Unlocking Solar Potential

How inverse modelling can contribute to uncovering plausible explanations for residential solar PV adoption dynamics in the Netherlands.

Noa Ommering

Delft University of Technology

Unlocking Solar Potential

How inverse modelling can contribute to uncovering plausible explanations for residential solar PV adoption dynamics in the Netherlands.

By

Noa Ommering

In partial fulfilment of the requirements for the degree of
Master of Science
In Complex Systems Engineering and Management (CoSEM)

at the Delft University of Technology
to be defended publicly on July 15th, 2024

First Supervisor:	Dr. Ir. E.J.L. (Emile) Chappin
Second Supervisor:	Dr. Y. (Yilin) Huang
Chair:	Dr. Ir. E.J.L. (Emile) Chappin
Advisor:	Lukas Schubotz
Student number:	4957601

TU Delft

Acknowledgements

I sincerely thank my supervisors, Emile, Yilin and Lukas, for their invaluable guidance and support throughout my thesis journey. Emile, our connection over the years has enriched my personal and academic life. I cannot thank you enough for not only being an academic mentor but also for teaching me the invaluable skill of “learning how to learn”. Embracing this journey was not always easy, but you showed me that discomfort is often the first step toward growth. Your ability to maintain a broader perspective and encourage growth greatly influenced my thesis work. Yilin, although the time we spent together was brief, your insightful questions sparked valuable reflections on my research, and for that, I am truly grateful. Lukas, as my daily supervisor, you have been a pillar of support, from brainstorming concepts to tackling technical issues. Your infectious enthusiasm and positivity always brought joy to the process.

To my friends, both within and outside TU Delft, thank you for always letting me see the bigger picture and for the eternal laughs. A special thanks to the other Lukas for your unwavering support through the ups and downs, especially during the challenges of living abroad. Your continuous encouragement meant the world to me. Indeed, “easier does not always mean better.”

Finally, I am deeply grateful to my family. My parents, who, despite not having the opportunity to study themselves, provided continuous support and all the resources necessary for my academic journey. For your love and devotion, I am eternally grateful. To my sister Babet, your support and lightheartedness have always brought laughter to my life.

Noa Ommering
Delft, June 2024

Executive Summary

The urgency of the transition from fossil to renewable energy sources is critical for the Netherlands, particularly under the National Climate Agreement, which aims to reduce CO_2 emissions by 95% by 2050 compared to 1990 levels. Local municipalities are at the forefront of this transition, tasked with implementing measures to decarbonise residential energy consumption, which accounts for approximately 20% of the total energy use in the Netherlands. One of the most promising technologies in this transition is solar photovoltaics (PV). Solar PV systems offer a practical solution for residential areas because they can integrate seamlessly into existing built environments. The Dutch government recognises this potential and has set a goal of generating at least 7 TWh of renewable energy production through small onshore solar projects by 2030. The adoption of residential solar PV has significantly exceeded expectations, with a capacity increase of 2600% over the past decade, achieving the goal of 7 TWh already by 2022.

However, the rapid adoption of solar PV presents several challenges for policymakers, particularly at the municipal level. These challenges include integrating solar power into the existing grid, developing financing mechanisms for solar PV, and ensuring equitable access to solar technology. Addressing these issues requires a deep understanding of the factors influencing solar PV adoption and the dynamics at play within different municipalities. Inverse modelling presents a methodology to explore these complexities. This technique involves working backwards from observed outcomes to identify and understand the underlying processes and parameters that generated those results. The application of inverse modelling is, however, very novel. Therefore, this thesis aims to lay the groundwork for future applications of inverse modelling by exploring the factors that influence solar PV adoption in Dutch municipalities. To achieve this goal, the following research question has been formulated:

How can inverse modelling contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities?

The methodology of this thesis involves a modelling approach that combines inverse modelling with agent-based modelling to model residential solar PV adoption. The study employs a multiple-case design to explore the adoption dynamics across different municipalities. This design allows for the identification of patterns, similarities and differences in solar PV adoption between these municipalities. Two machine learning algorithms are used to enable the inverse modelling process by exploring the parameter space. These algorithms are a Random Search algorithm and a Bayesian Search algorithm.

The results of this thesis reveal insights into the factors influencing solar PV adoption. Social factors emerge as the most critical determinants, often outweighing economic incentives. For instance, in Bloemendaal, despite its high income levels, social influences are more impactful than economic factors. This trend is consistent across the other municipalities, where social utility plays a crucial role in adoption rates. However, in Vaals, economic incentives are more significant than in the other municipalities, highlighting the variability in adoption drivers across the municipalities. Environmental impact, on the other hand, significantly influences adoption decisions in Westerveld, indicating that regional differences must be considered in policymaking.

The discussion chapter delves into the implications of these findings. It discusses the conclusiveness of the current results in light of data constraints and the reliance on a single ABM, which required extensive modifications. Despite these challenges, this thesis demonstrates the viability of inverse modelling as a promising approach to understanding solar PV adoption dynamics. However, it is also emphasised that further research is necessary to enhance the conclusiveness of the results. Furthermore, it questions the previously mentioned performance dilemma, which discusses whether individual results or general robustness is more important. Considering the purpose of inverse modelling, it is concluded that consistency and robustness seem more important in the inverse modelling process.

In terms of scientific relevance, this thesis makes substantial methodological contributions by demonstrating the application of inverse modelling in the socio-economic context of solar PV adoption in Dutch municipalities. The transparency of this work in addressing its limitations also sets a precedent for future

research on the topic. These limitations include the study's reliance on a heavily modified ABM, computational constraints, limited data points and the exploration of only four parameters, which may oversimplify the complex dynamics of solar PV adoption. This thesis provides an initial framework for future inverse modelling research. The study's social relevance is that understanding the dynamics of solar PV adoption can inform policy decisions and interventions aimed at promoting solar PV. The study underscores the importance of considering social dimensions in solar PV adoption, noting that financial incentives alone are insufficient to drive widespread adoption.

Finally, recommendations for future research include exploring alternative approaches for inverse modelling. Since the method outlined in this thesis is novel and experimental, alternative approaches are presented to enhance the inverse modelling framework. For instance, an alternative approach is the incorporation of multiple ABMs, which could enhance the robustness and cross-validate results, though this would increase complexity and demand more computational resources. Additionally, future inverse modelling research could either focus more on machine learning to increase model accuracy or focus less on machine learning to explore other methodologies. Exploring hybrid approaches to these recommendations that combine various techniques could also offer possibilities for future research.

Table of Contents

Acknowledgements ... **iii**
Executive Summary ... **iv**
Table of Contents ... **vi**
List of Figures ... **viii**
List of Tables ... **ix**
List of Abbreviations ... **xi**
1. Introduction ... **1**
1.1 Research Problem ... 1
1.2 Research Question ... 2
1.3 Systems Perspective and Relation to the CoSEM Programme ... 3
1.4 Introduction to the Research Approach ... 3
1.5 Research Outline ... 4
2. Theoretical Background ... **5**
2.1 Solar Photovoltaics (PV) Adoption ... 5
2.2 Inverse Modelling ... 6
2.3 Agent-Based Modelling ... 9
2.4 Relations Between Concepts ... 10
2.5 Literature Review ... 11
3. Methodology ... **16**
3.1 Research Approach ... 16
3.2 Research Framework ... 17
3.3 Scope of the Case Study ... 20
3.4 Machine Learning as a Means for Inverse Modelling ... 21
3.5 Chapter Summary ... 21
4. Data & Algorithms ... **23**
4.1 Data requisites ... 23
4.2 Machine Learning Algorithms ... 25
4.3 Chapter Summary ... 27
5. The Model ... **28**
5.1 NetLogo Model Elaboration ... 28
5.2 Model Conceptualisation Within the Inverse Modelling Process ... 33
5.3 Python Model Formalisation ... 36
5.4 Python Model Validation and Verification ... 39
5.5 Chapter Summary ... 40
6. Results ... **42**
6.1 The Municipality of Bloemendaal ... 42
6.2 The Municipality of Dantumadiel ... 47
6.3 The Municipality of Laren ... 52
6.4 The Municipality of Oegstgeest ... 56
6.5 The Municipality of Vaals ... 60
6.6 The Municipality of Westerveld ... 63
6.7 Inter-Municipal Comparative Analysis ... 66
6.8 The Municipality of Laren: Increased Splits and Iterations ... 71
6.9 Chapter Summary ... 76
7. Research Implications ... **78**
7.1 Performance Dilemma ... 78
7.2 Random Search vs Bayesian Search Performance ... 79
7.3 Conclusiveness of the Inverse Modelling Process ... 80
7.4 Reproducibility and Replicability ... 81
7.5 Alternative Approaches ... 82
7.6 Chapter Summary ... 83

8. Conclusion **85**
8.1 Answering the Research Question 85
8.2 Scientific Relevance 86
8.3 Societal Relevance 87
8.4 Limitations of the Study 87
8.5 Future Directions and Applications of Inverse Modelling Research 88
Bibliography **91**
Appendix **98**
A. 1st Python Script for NetLogo Model 99
B. 2nd Python Script for NetLogo Model 102
C. Income Distributions NetLogo 105
D. Initial PV share NetLogo 108
E. Agent State Variables NetLogo 109
F. Environmental State Variables NetLogo 110
G. Pseudo-Code Main Simulation NetLogo 111
H. Excluded Parameters for the IM process 112
I. Pseudo-code 115
J. Data Analysis Bloemendaal 119
K. Data Analysis Dantumadiel 122
L. Data Analysis Laren 125
M. Data Analysis Oegstgeest 128
N. Data Analysis Vaals 131
O. Data Analysis Westerveld 134
P. Laren Increased Splits and Iterations 137
Q. Inter-Municipal Analysis 140

List of Figures

Figure 1: Total installed residential PV capacity in the Netherlands between 2012 and 2020 2
Figure 2: Ways to study a system, highlighting the choices for this research 4
Figure 3: The development of installed capacity of household PV systems in the Netherlands 6
Figure 4: Visual presentation of Inverse Generative Social Science 7
Figure 5: Relations between the concepts 11
Figure 6: Rounds in the literature review.. 11
Figure 7: Prisma 2020 diagram of the literature review 12
Figure 8: Research Framework.. 19
Figure 9: Number of residential solar PV installations per municipality... 25
Figure 10: Number of solar PV installations per household... 25
Figure 11: The simulation process for the operationalisation of the Theory of Planned behaviour 30
Figure 12: Presentation of the interface of the model by Muelder & Filatova (2018).................................... 31
Figure 13: Conceptualisation of the inverse modelling process in this research.. 35
Figure 14: UML diagram of the basic model structure of the Python model.. 36
Figure 15: More detailed UML diagram of the Python model structure.. 38
Figure 16: Validation curve of the model, presenting the optimal amount of splits...................................... 41
Figure 17: Boxplot of the results of the inverse modelling process for the municipality of Bloemendaal 44
Figure 18: Plot of the real and the predicted solar PV adoption values for Bloemendaal............................. 47
Figure 19: Google Trends scores for the province of Friesland 48
Figure 20: Boxplot of the results of the inverse modelling process for the municipality of Dantumadiel....... 49
Figure 21: Plot of the real and the predicted solar PV adoption values for Dantumadiel.............................. 52
Figure 22: Google Trends score for the province of Noord-Holland 53
Figure 23: Boxplot of the results of the inverse modelling process for the municipality of Laren.................. 53
Figure 24: Plot of the real and the predicted solar PV adoption values for Laren 56
Figure 25: Boxplot of the results of the inverse modelling process for the municipality of Oegstgeest......... 57
Figure 26: Plot of the real and the predicted solar PV adoption values for Oegstgeest 59
Figure 27: Boxplot of the results of the inverse modelling process for the municipality of Vaals.................. 60
Figure 28: Plot of the real and the predicted solar PV adoption values for Vaals 63
Figure 29: Boxplot of the results of the inverse modelling process for the municipality of Westerveld......... 64
Figure 30: Plot of the real and the predicted solar PV adoption values for Westerveld................................ 66
Figure 31: Boxplots of the results for the municipality of Bloemendaal, Dantumadiel and Laren 67
Figure 32: Boxplots of the results for the municipalities of Oegstgeest, Vaals and Westerveld 68
Figure 33: Boxplot of the comparison between runs with the experimental setup in Laren.......................... 72
Figure 34: Plot of the real and predicted solar PV adoption values for Laren for the experimental setup 73
Figure 35: Boxplot of the comparison between runs with the experimental setup for Laren 74
Figure 36: Plot of the real and predicted solar PV adoption values for Laren for the experimental setup 75
Figure 37: Overview of areas with potential for methodological advancements in the IM process. 90

List of Tables

Table 1: Overview of Machine Learning categories ... 8
Table 2: Research gap identification ... 15
Table 3: Overview of the selected municipalities for the case study ... 21
Table 4: Training data for inverse modelling ... 24
Table 5: Attribute levels for environmental state variables that cannot be changed during the run ... 29
Table 6: Parameter Distributions for the Inverse Modelling Process in Python ... 32
Table 7: Mean, median and standard deviation of the Random Search results for Bloemendaal ... 44
Table 8: Summarised data analysis for the Random Search algorithm for Bloemendaal ... 45
Table 9: Mean, median and standard deviation of the Bayesian Search results for Bloemendaal ... 46
Table 10: Summarised data analysis for the Bayesian Search algorithm for Bloemendaal ... 46
Table 11: Mean, median and standard deviation of the Random Search results for Dantumadiel ... 49
Table 12: Summarised data analysis for the Random Search algorithm for Dantumadiel ... 50
Table 13: Mean, median and standard deviation of the Bayesian Search results for Dantumadiel ... 50
Table 14: Summarised data analysis for the Bayesian Search algorithm for Dantumadiel ... 51
Table 15: Mean, median and standard deviation of the Random Search results for Laren ... 54
Table 16: Summarised data analysis for the Random Search algorithm for Laren ... 54
Table 17: Mean, median and standard deviation of the Bayesian Search results for Laren ... 55
Table 18: Summarised data analysis for the Bayesian Search algorithm for Laren ... 55
Table 19: Mean, median and standard deviation of the Random Search results for Oegstgeest ... 57
Table 20: Summarised data analysis for the Random Search algorithm for Oegstgeest ... 58
Table 21: Mean, median and standard deviation of the Bayesian Search results for Oegstgeest ... 58
Table 22: Summarised data analysis for the Bayesian Search algorithm for Oegstgeest ... 59
Table 23: Mean, median and standard deviation of the Random Search results for Vaals ... 61
Table 24: Summarised data analysis for the Random Search algorithm for Vaals ... 61
Table 25: Mean, median and standard deviation of the Bayesian Search results for Vaals ... 62
Table 26: Summarised data analysis for the Bayesian Search algorithm for Vaals ... 62
Table 27: Mean, median and standard deviation of the Random Search results for Westerveld ... 64
Table 28: Summarised data analysis for the Random Search algorithm for Westerveld ... 65
Table 29: Mean, median and standard deviation of the Bayesian Search results for Westerveld ... 65
Table 30: Summarised data analysis for the Bayesian Search algorithm for Westerveld ... 65
Table 31: Average values of the parameters of the Random Search models ... 69
Table 32: Top 5 best-performing result values of the Random Search models ... 69
Table 33: Average values of the parameters of the Bayesian Search models ... 70
Table 34: Top 5 best-performing result values of the Bayesian Search models ... 71
Table 35: Comparison of the descriptive statistics between runs with the experimental or original setup ... 73
Table 36: Summarised data analysis of the experimental setup for the municipality of Laren ... 74
Table 37: Comparison of the descriptive statistics between runs with the experimental or original setup ... 75
Table 38: Summarised data analysis of the experimental setup for the municipality of Laren ... 76
Table 39: Income distribution for Bloemendaal ... 105
Table 40: Income distribution for Dantumadiel ... 105
Table 41: Income distribution for Laren ... 106
Table 42: Income distribution for Oegstgeest ... 106
Table 43: Income distribution for Vaals ... 106
Table 44: Income distribution for Westerveld ... 107
Table 45: The initial PV share per municipality, as used in the NetLogo model ... 108
Table 46: State variables of the agents of the ABM, modified to fit the case of this research ... 109
Table 47: State variables of the environment of the ABM, modified to fit the case of this research ... 110
Table 48: Excluded parameters for the IM process, including default value in the NetLogo model ... 112
Table 49: Data analysis results of the municipality of Bloemendaal for the Random Search algorithm ... 119
Table 50: Data analysis results of the municipality of Bloemendaal for the Bayesian Search algorithm ... 120
Table 51: Data analysis results of the municipality of Dantumadiel for the Random Search algorithm ... 122
Table 52: Data analysis results of the municipality of Dantumadiel for the Bayesian Search algorithm ... 123

Table 53: Data analysis results of the municipality of Laren for the Random Search algorithm 125
Table 54: Data analysis results of the municipality of Laren for the Bayesian Search algorithm 126
Table 55: Data analysis results of the municipality of Oegstgeest for the Random Search algorithm 128
Table 56: Data analysis results of the municipality of Oegstgeest for the Bayesian Search algorithm 129
Table 57: Data analysis results of the municipality of Vaals for the Random Search algorithm 131
Table 58: Data analysis results of the municipality of Vaals for the Bayesian Search algorithm 132
Table 59: Data analysis results of the municipality of Westerveld for the Random Search algorithm 134
Table 60: Data analysis results of the municipality of Westerveld for the Bayesian Search algorithm 135
Table 61: Data analysis results of the municipality of Laren for the Random Search algorithm 137
Table 62: Data analysis results of the municipality of Laren for the Bayesian Search algorithm 138
Table 63: Mean, median and standard deviations of the results for the Random Search algorithm 140
Table 64: Mean, median and standard deviations of the results for the Bayesian Search algorithm 140

List of Abbreviations

Abbreviation	Definition
ABM	Agent-Based Model
AI	Artificial Intelligence
DTW	Dynamic Time Warping
EU	European Union
IGSS	Inverse Generative Social Science
IM	Inverse Modelling
IQR	Interquartile Range
ML	Machine Learning
PV	Photovoltaic
PBC	Perceived Behavioural Control
TPB	Theory of Planned Behaviour

1 Introduction

This chapter introduces the research problem that will be addressed in this work. Furthermore, it emphasises the scientific relevance of the problem and the research question derived in Chapter 1.2. Subsequently, in Chapter 1.3 the systems perspective and relation to the master program are depicted, after which the research approach is presented in Chapter 1.4. Finally, the outline of the thesis is presented in Chapter 1.5.

1.1 Research Problem

The urgency of the energy transition has risen on the political agenda, as underscored by the unveiling of the National Climate Agreement by the Dutch Government in 2019 (Government of the Netherlands, 2023a). This agreement is a crucial component of the National Energy and Climate Plan (NCEP) and the Climate Plan required by the European Union (European Commission, 2023). The main ambition of this agreement is a CO_2 reduction target of 95% by 2050 in comparison to 1990, presenting a major sustainability challenge for the electricity sector (RVO, 2023b). Embedded within this overarching national initiative lies a significant focus on municipalities, underscoring the critical role local governance plays in achieving sustainability targets (Government of the Netherlands, 2023c). In the Netherlands, approximately 20% of the final energy consumption can be attributed to residential households, emphasising the significance of decarbonisation through the electrification of this sector as a key strategy on the journey towards achieving net-zero carbon emissions (Besagni et al., 2021; Zappa & van den Broek, 2018). This also offers a concrete pathway for municipalities to progress towards their objectives outlined in the National Climate Agreement.

Solar photovoltaics (PV) is expected to play a fundamental role in urban areas to meet the objectives of the National Climate Agreement as it offers the advantage of simple integration into the current built environment (Creutzig et al., 2017; Shafique et al., 2020). Therefore, the Dutch government has set the goal in the National Climate Agreement of at least 7 TWh renewable energy generation with small onshore solar energy projects of less than 15 kWp (Government of the Netherlands, 2023b). Over the past decade, residential solar PV capacity in the Netherlands has increased by over 2600%, as can be observed in Figure 1 below (Ballas et al., 2023). This transformation resonates deeply at the municipal level since this growth strongly advances sustainability goals. By the end of 2022, a total of 7,2 TWh of renewable energy generation with small onshore solar energy projects of less than 15 kWp had been realised, reaching goals more than 7 years ahead of projections (RVO, 2023a). This substantial growth can be attributed to increased cost competitiveness, technological advances and enhanced durability and reliability of solar panels, collectively leading to a notable shift in public opinion among Dutch citizens regarding solar energy (Statista, 2022; Kraaijvanger et al., 2023). However, this rapid change in Dutch citizens' attitudes towards solar energy poses a challenge for policymakers, particularly at the municipal level. Key concerns include ensuring the seamless integration of solar power into existing grids, establishing sustainable financing mechanisms to support infrastructure development and addressing social equity issues to ensure fair access to solar technologies (Kraaijvanger et al., 2023; Pierie et al., 2021). Policymakers must navigate these complexities to foster financial viability, inclusive solar PV adoption, and equitable benefit distribution.

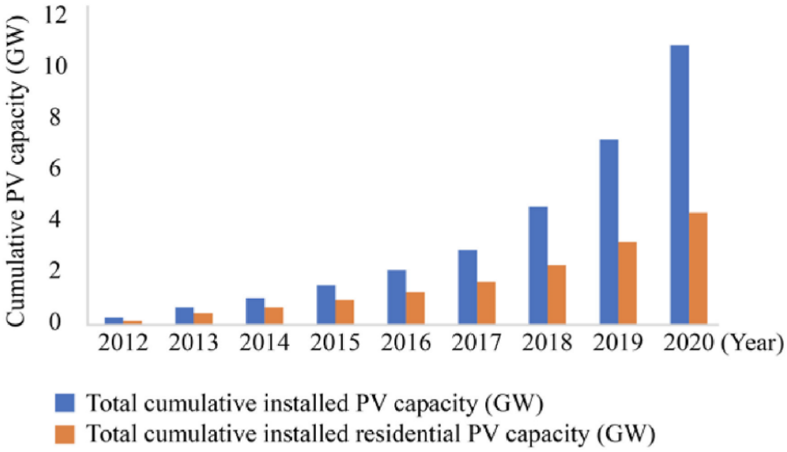


Figure 1: Total installed residential PV capacity in the Netherlands between 2012 and 2020 (Ballas et al., 2023)

Traditional ABMs (Agent-Based Models) are commonly used for analysing complex adaptive systems and transitions such as the energy transition (Schelling, 1971). They help researchers and policymakers understand complex system behaviour, predict future actions and address socio-technical challenges such as the energy transition (Turgut & Bozdag, 2023). However, the system's complexity and dynamic nature, combined with the constraints of observational data and modelling methodologies, may result in certain features going unnoticed.

Inverse modelling is a concept that could contribute to this challenge. IM employs programming to construct equation-based algorithms and decision trees for agents across simulation runs until the desired outcome is achieved (So et al., 2020). In doing so, inverse modelling offers a promising methodology for understanding the complexities of residential solar PV adoption dynamics. It can be used to uncover previously overlooked patterns in a system, which shows great potential for this research. In addition, it also shows potential for improving ABMs, increasing their adaptability during runs and thereby increasing their resemblance to reality. The application of inverse modelling is, however, very novel.

For this reason, this study aims to lay the groundwork for future applications of inverse modelling by exploring the factors that influence solar PV adoption in Dutch municipalities. Agent-based modelling and machine learning have an instrumental function towards this goal (Bogner et al., 2020). The research aims to employ IM techniques to reveal previously overlooked patterns and features influencing solar PV adoption, thereby enriching the understanding of adoption dynamics beyond traditional predictive accuracy metrics for solar PV adoption in the Netherlands. This also means that by leveraging IM to unravel the complex dynamics underlying solar PV adoption, this study aims to pave the way for future research and more practical applications of inverse modelling.

1.2 Research Question

Based on the research objective described above, the following main research question can be derived with the aim of providing new insights into the potential contribution of ML and IM to ABMs.

How can inverse modelling contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities?

Answering the question presented above can generate insights into the potential future role of incorporating inverse modelling in agent-based modelling for energy transition questions.

1.3 Systems Perspective and Relation to the CoSEM Programme

This thesis topic fits well into the Complex Systems Engineering and Management (CoSEM) study programme as the challenge can be characterised as a *Complex Adaptive System*: a dynamic network comprising numerous agents which operate in parallel, continuously engaging in actions and reactions to those of other agents (Holland, 2006). The collective behaviour of the system emerges from many decisions made by numerous individual agents at each moment. This definition can be applied to the case of residential solar PV capacity in the Netherlands since it is the actions and decisions of many agents (e.g. residents, governmental organisations, firms) that result in a certain amount of installed residential solar PV capacity. Consequently, *complex system properties* ought to be taken into consideration (e.g. emergent behaviour, self-organisation and path dependence) (Nikolic et al., 2013a).

On top of that, this topic requires a multidisciplinary approach as it encompasses both a *human* and a *physical subsystem*. The human subsystem comprises psychological, demographic and biophysical variables such as residents, whereas the physical subsystem consists of the technical solar PV systems and infrastructure (Hitchcock, 1993). Moreover, these human and physical subsystems converge with *institutional frameworks*, which refer to the policies, regulations, incentives and organisations that govern the adoption, installation and integration of residential solar PV systems (Gupta et al., 2008). On top of that, the use of artificial intelligence presents *ethical* concerns (Díaz-Domínguez, 2020). Some ethical concerns regarding machine learning may include bias, which can arise from the analysis of extensive datasets and the tendency to prioritise popular outcomes. Additionally, a potential lack of transparency, or opacity, poses a challenge as machine learning models operate as “black boxes”.

This study holds societal importance as it seeks to research the role of inverse modelling in uncovering plausible explanations for solar PV adoption dynamics in the Netherlands. These insights could be used in further research towards inverse modelling or influence interventions in decision-making processes related to the energy transition. This, in turn, facilitates more efficient and secure planning of energy infrastructure (Moglia et al., 2022). The study also leads to an improved and more accurate understanding of the system, which could result in more thoughtful policy formulation and even improved energy grid management in the future.

Thus, this challenge encompasses strong social, technological and institutional aspects and therefore fits well into the CoSEM study programme.

1.4 Introduction to the Research Approach

There are various methods to study the forecasting of systems such as residential solar PV capacity. In Figure 2 different ways to study a system are presented, where the components in the blue frame are based on Law & Kelton (1991), pages 2 and 109 specifically. The components in the red frame adhere to the simulation taxonomy used by for instance Borshchev & Filippov (2004) and Sumari et al. (2013). A *modelling* approach has been selected for this research because experimenting with an actual system is impossible when forecasting residential solar PV capacity. Physical models, such as the model presented by Liu & Zhang (2016), are very effective when forecasting solar PV capacity using technical and geographical data but overlook the influence of human behaviour on the system. Therefore, a *mathematical model* is chosen over a physical model for this research. Due to the complexity of system interactions in the case, a *simulation model* is chosen over an analytical solution.

To study solar PV adoption, a simulation modelling approach that is both generative and interactive is required (Nikolic et al., 2013b). Therefore *Agent-Based Modelling* is chosen to form the base of the inverse modelling process. The reason for this lies in the ability of ABMs to capture dynamic systems that are high in complexity, multidisciplinary and adaptive (Borshchev & Filippov, 2004). Moreover, using an ABM in this research allows for a bottom-up approach for modelling individual consumer behaviour towards solar PV (Zhang et al., 2022). Individual, heterogeneous actors can engage in interactions among themselves and with their surroundings. Simultaneously, the impacts of various organisational, institutional and temporal factors can be examined.

The development of a generic ABM proves, however, challenging due to the data precision that is required to forecast residential solar PV capacity. Consequently, a case study approach is selected to complement the modelling approach. This research will adopt a multiple-case design (Yin, 2012), which is particularly suitable for exploring complex phenomena and allowing for the investigation of patterns, commonalities, and differences across cases. Additionally, the embedded nature of this design enables the ability to capture in-depth knowledge of the case and its underlying complexity (Ragin, 2014). This means that a multiple-case study approach can aid in the formulation of new theories since an in-depth analysis of the relations between cases might unveil patterns or connections that contribute to theoretical progress. This aspect is important for this research, given that the application of inverse modelling to ABM is a new field. The exact research approach will be further elaborated in Chapter 3.

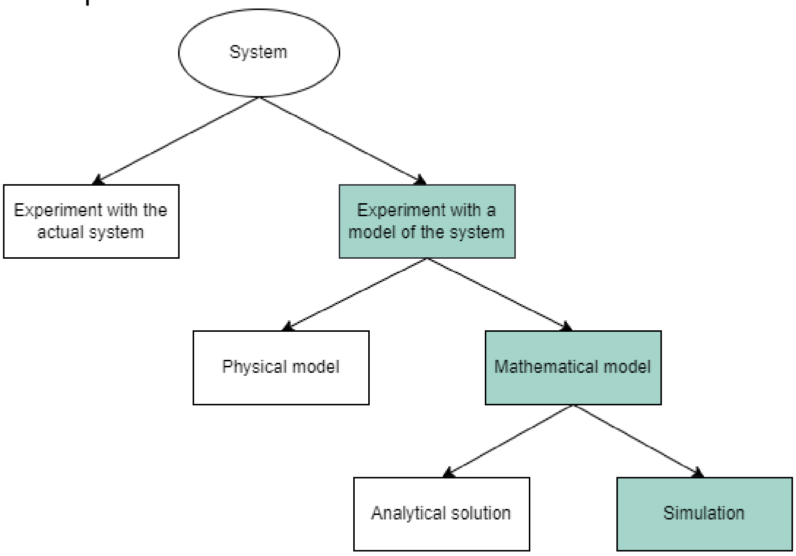


Figure 2: Ways to study a system, adapted from Law & Kelton (1991), highlighting the choices for this research

In the context of this research, inverse modelling is employed as a methodology for understanding the dynamics underlying residential solar PV adoption. It entails the *comparison of observed solar PV adoption data with model predictions and adjustment of model inputs to match real-world patterns.* This creates the potential for a deeper understanding of the factors influencing adoption rates across Dutch municipalities.

1.5 Research Outline

The structure of this thesis is as follows: after the introduction, the theoretical framework and relevant concepts for the research are elaborated in Chapter 2. On top of that, this chapter presents a literature review. Chapter 3 introduces the methodology for this research, including the research approach, research framework and the scope of the case study. Chapter 4 focuses on the data required for this research and the two algorithms that will be used to enhance an ABM on residential solar PV adoption. Then, Chapter 5 focuses on the model. This means that the current ABM that will be used for the IM process is presented along with the modifications that ought to be made for the mode to fit the case. This chapter also presents the model formalisation, software implementation and the model validation and verification. Chapter 6 presents the results of the study, including the data analysis of these results. Then, Chapter 7 focuses on the discussions that come forth from this research. Finally, Chapter 8 presents the conclusions that can be drawn from this research and the study's limitations. This chapter also includes the research questions' answers and the research's scientific and societal relevance.

2
Theoretical Background

To comprehend the complexity of the question presented in the introduction, it is essential to grasp relevant concepts. In this regard, this chapter adopts a top-down approach. The chapter begins by presenting the concept of solar photovoltaics in Chapter 2.1. Then, it introduces the method of inverse modelling in Chapter 2.2, and later, it delves into the principles of agent-based modelling in Chapter 2.3. The reasoning behind this order lies in the objective of this study. The exploration focuses mainly on plausible explorations for solar PV adoption and how inverse modelling can enable this. Therefore, these two topics are introduced first. Since agent-based modelling is instrumental to the aim of this research, these topics will be introduced after the topics of inverse modelling and solar PV adoption. These relations between the concepts are presented in Chapter 2.4. Finally, Chapter 2.5 presents the academic knowledge gap. To ensure that the research conducted in this thesis makes a meaningful contribution to science and avoids reinventing the wheel, establishing a knowledge gap is essential. This gap is identified through a PRISMA literature review, with the findings resulting in the formulation of the main research question.

2.1 Solar Photovoltaics (PV) Adoption

The first relevant concept is *solar photovoltaics (PV)* and their role in the energy transition of residential areas. It relates to the conversion of energy contained in sunlight, *solar energy*, into electricity. When this energy undergoes direct conversion into electricity utilising semiconductor-based devices, it is referred to as *photovoltaics* (Smets et al., 2016). Solar PV has emerged as a key technology in the global transition towards renewable energy sources, offering substantial potential for mitigating climate change and reducing dependence on fossil fuels. In the Netherlands, solar PV is expected to play a fundamental role on a municipal level in meeting the national renewable energy targets. This is for instance due to the fact that households account for almost 20% of the total energy consumption in the Netherlands (Klimaatmonitor, 2021). This highlights that decarbonization via the electrification of residences is a pivotal driver towards achieving national targets (Jägemann et al., 2013). On top of that, solar PV knows the advantage of relatively simple and seamless integration into the existing built environment which also makes solar PV an attractive solution for municipalities (Creutzig et al., 2017; Shafique et al., 2020).

In the Netherlands, installed solar PV capacity in urban areas has grown significantly in the past decade. This is believed to have had several reasons. For starters, solar PV technology knows strong technological advances, with advancements pushing the boundaries of conversion rates. For years, the efficiency of solar panels was roughly 17%, but recent developments are propelling efficiency rates to an anticipated 26% (TNO, 2023). These strides are achieved through innovations in materials and cell designs. These advancements do not only enhance the performance of solar PV systems, but they also drive down costs, making solar energy increasingly competitive and accessible for adoption on a residential level. On top of that, the durability and reliability of solar panels have improved over time due to improvements in manufacturing processes, materials and quality control measurements. These enhancements further increase the appeal and attractiveness of solar panels. Finally, the Dutch policy environment regarding solar energy has been pivotal (Osseweijer et al., 2018). According to Londo et al. (2020), the net metering policy is, next to feed-in tariffs, considered to be an important mechanism for improving the financial viability of households considering investing in solar PV. The Netherlands has had a net metering policy since 2004, which enables households with a PV system (referred to as 'prosumers') to utilise the electricity they generate at their discretion rather than solely at the moment of generation (Government of the Netherlands, 2004). This effectively equates the value of power generated by their PV system to the consumer tariff regardless of when it is produced or consumed. In combination with the substantial decrease in PV system costs, the net metering policy is considered a significant contributor to the rapid decrease in payback times for PV systems for Dutch households. Figure 3 presents the decrease in payback times and the corresponding growth in the capacity of PV systems in the first major adoption years in the Netherlands.

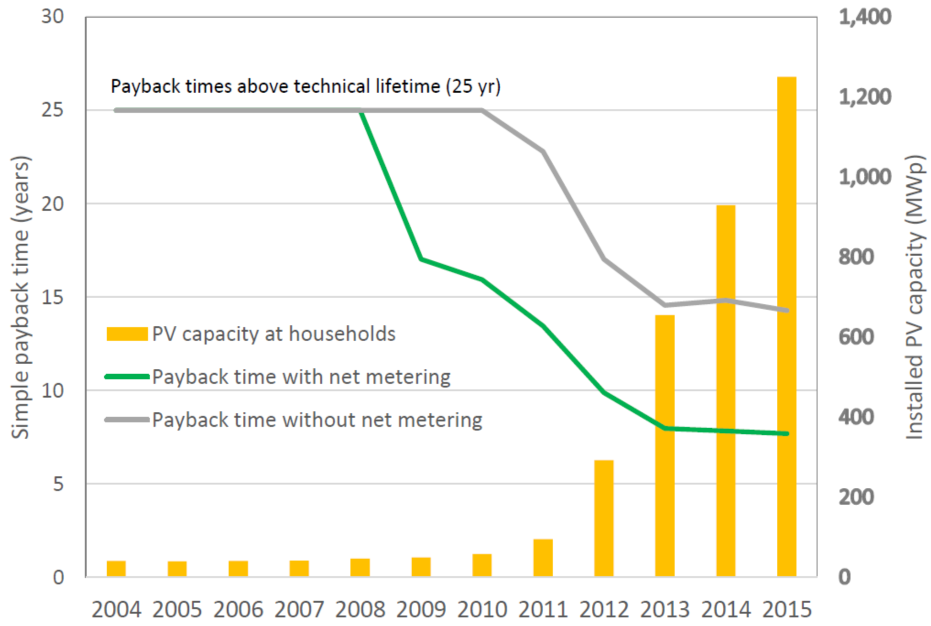


Figure 3: The development of installed capacity of household PV systems in the Netherlands (Londo et al., 2020)

As can be seen in both Figure 1 and Figure 3, the solar PV transition in the Netherlands is already well on its way, meaning that a wide variety of data is available on the topic. This makes a good case for inverse modelling, as relatively large datasets are required for this method.

2.2 Inverse Modelling

Inverse modelling is a method which can be derived from Inverse Generative Social Science (IGSS). IGSS represents a novel methodology within the domain of social science, wherein researchers use generative models to understand how social phenomena work. As elaborated before, ABMs are currently the primary tool for understanding how individual behaviours and interactions influence larger social patterns (Vu et al., 2019). However, until now, agents have been moved forward in iterations to produce explanations for certain phenomena (Epstein, 2012). This presents the *forward modelling problem*, which involves the prediction of the outcome of a system or phenomenon based on known inputs or parameters. This process does, however, not *explain* the system or phenomenon. Inverse Generative Social Science attempts to address this problem. The approach aims to understand the underlying processes that generate observed social data and can be used to make predictions or test hypotheses about social systems (Epstein, 1999). It is often used when the system is complex or poorly understood, and it requires iterative techniques to converge on the most likely values for unknown parameters. The link with inverse modelling becomes apparent in this methodology, as both share a common objective: the estimation of model parameters or inputs based on observed data and a deeper understanding of the system. Inverse Generative Social Science can utilise inverse modelling techniques as part of its methodology, which means that in this context, inverse modelling can be used to understand the complex dynamics underlying solar PV adoption. In other words, the forward modelling problem focuses on predicting outcomes based on known inputs of parameters whereas inverse modelling involves estimating unknown inputs or parameters based on observed outcomes. This potentially reveals new dynamics that may have been overlooked due to being counterintuitive or not immediately apparent (Greig et al., 2023). In the figure below, the process of IGSS is represented, demonstrating its efforts to reveal the potential spectrum of individual behaviours that contribute to an observed collective dynamic (Naumann-Woleske, 2021).

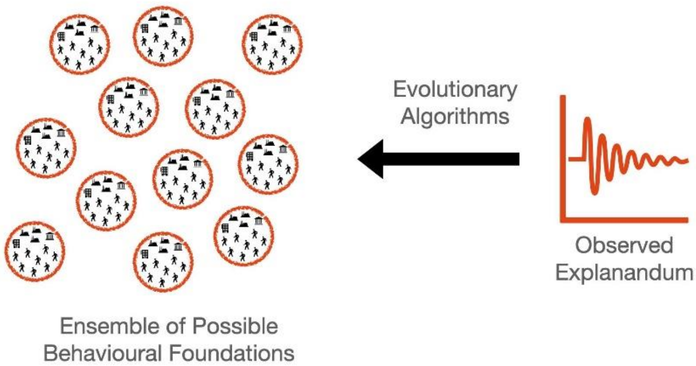


Figure 4: Visual presentation of Inverse Generative Social Science. Source: Naumann-Woleske (2021)

2.2.1. Challenges in Inverse Modelling

Inverse modelling is a promising method, however, it also comes with certain challenges. For one, the topic of inverse modelling is very novel and there is a scarcity of literature addressing the subject. This novelty of the concept of inverse modelling is a potential challenge for this research since it means that there is very limited experience and available documentation on its implementation.

On top of that, the challenge of data overfitting arises (Renzini et al., 2023). In inverse modelling, data overfitting can occur when a complex model fits 'noise' (random variations or fluctuations in the data that do not represent meaningful information or patterns) rather than underlying patterns due to limited data. This leads to inaccurate parameter estimates. Furthermore, according to Frank et al. (2022), a crucial consideration within inverse modelling revolves around determining when data is sufficient to determine model parameters. Insufficient data may result in unreliable parameter estimates, impairing the model's ability to truthfully represent the underlying system. Finally, the high sensitivity of both data quality and quantity poses a significant challenge in inverse modelling. This means that inaccuracies or uncertainties in the data can lead to biased or unreliable parameter estimates as well. This again hinders the model's ability to accurately represent the underlying system. Balancing the need for high-quality data and an adequate quantity of data is crucial for inverse modelling, and it highlights the importance of careful data collection and validation.

It is important to note that while these challenges remain valid, this research places a lesser emphasis on them. Instead, the focus lies on establishing a foundational understanding of inverse modelling. Thus, achieving perfection in addressing these challenges is not the primary goal. Rather, the research aims to provide a basis for future work in inverse modelling, acknowledging the importance of careful data collection and validation while still recognising that absolute perfection in modelling is not attainable (Box, 1976)

2.2.2. Machine Learning as a Means for Inverse Modelling

Inverse modelling can be enabled through machine learning (abbreviated as ML) (Epstein, 2023). This is a research field within computer science (specifically artificial intelligence) that is dedicated to researching algorithms and methodologies aimed at automating solutions for complex problems that prove challenging to address through traditional programming approaches (Rebala et al., 2019). It aims to enable computers to improve their performance over time and to learn without explicit programming (Bi et al. (2019); Murphy (2012)). Through fitting data, computers gain certain "experience" and can thereby improve their functioning. Machine learning possesses the unique ability to extract patterns from data and draw inferences even when encountering previously unseen data (Bogner et al., 2020). These capabilities offer a significant advantage in overcoming some of the previously mentioned challenges (Bashardoust et al., 2023).

The connection between machine learning and this research lies within its complementary and instrumental role in inverse modelling. According to Vu et al. (2019), machine learning has the potential to help explain human behaviour, potentially moving the modelling of complex systems into a new era. Inverse modelling helps refine the parameters of the ABM to better match observed data and understand the underlying system, while machine learning techniques aid in the analysis of real-world data to identify patterns and features that inform the parameter refinement process. Together, they can enable a deeper understanding of the factors influencing solar PV adoption dynamics. In other words: machine learning can enable inverse modelling. The application of machine learning to inverse modelling is, however, extremely novel.

The domain of machine learning can be categorised in various manners (Flach (2012) ; Morales & Escalante (2022)). This study follows a traditional categorisation for a broader perspective of ML, namely (1) supervised learning, (2) unsupervised learning and (3) reinforcement learning (Alpaydin, 2010). In the table below, an overview of these categories is presented together with their objectives (Ale Ebrahim Dehkordi et al., 2023).

Table 1: Overview of Machine Learning categories (Ale Ebrahim Dehkordi et al., 2023)

ML category	Data	Objective	Learning
Supervised	Labelled	Prediction	By mapping inputs to desired outputs
Unsupervised	Unlabelled	Identification of structures or patterns	By identifying the distribution or the structure of the input
Reinforcement	Interaction with environment	Optimisation	By rewarding good behaviour

At first sight, unsupervised learning seems to be more suitable for this research due to its ability to identify structures or patterns (Mohri et al., 2018). However, this study employs a different approach – reinforcement learning. Although the data on solar PV adoption includes labelled information, suggesting supervised learning (Chawla & Karakoulas, 2005), this research benefits from a RL framework, which allows the model to learn optimal strategies through trial and error. Furthermore, the exploitation vs exploitation balance is a central theme. As will be discussed in Chapter 7, managing this balance is crucial in the IM process, proving the suitability of RL.

As mentioned, using ML for IM is unprecedented in the literature. This novel approach poses a challenge due to the lack of experience and available documentation, making correct implementation difficult. However, this novelty adds value to any results obtained, contributing to future research. A second challenge in the use of ML for IM lies in its ethics. As our dependence on technology continues to expand, the significance of ethical considerations in machine learning will become more important. (Shadowen, 2017). One ethical issue is the 'black box' problem, concerning the lack of transparency in the decision-making processes of ML algorithms (Miller, 2019). Secondly, the application of ML can create ethical issues in terms of trustworthiness and fairness. In the context of computer systems, trustworthiness denotes the assurance that these systems operate in accordance with the assertions made by their designers. A third ethical consideration lies within the issue of 'learning without understanding'. It refers to the capability of machine learning models to make accurate predictions without truly comprehending the underlying concepts or relationships in the data or as described by Epstein (2019): "Artificial intelligence and machine learning are displacing humans, but not explaining them". This means that the application of ML to IM in itself does not explain the phenomena.

2.3 Agent-Based Modelling

Agent-based modelling (ABM) is the third relevant concept in this research and it describes a bottom-up approach for the simulation of human systems and behaviour (A. T. Crooks & Heppenstall, 2012). It is a computational modelling technique that is used to simulate complex systems by modelling individual agents and their interactions within an environment (Nikolic et al., 2013a). These agents can represent a wide variety of entities, such as individuals, organisations, or animals, each with unique attributes, behaviours, and decision-making processes. Typically, each agent follows a set of rules of behaviour based on its characteristics and interactions with other agents in the environment. As an ABM operates based on principles of bottom-up modelling, system-level behaviours emerge from interactions of individual agents.

The primary purpose of ABMs is to be able to understand and analyse the dynamics of complex systems by simulating the interactions and behaviours of the agents in these systems (Turgut & Bozdag, 2023). It is used to deduce the system's behaviour (macro-level) or emergent behaviour by modelling the agents' individual behaviour (micro-level) and interaction (Nan, 2011). By capturing the heterogeneity and autonomy of agents, ABMs can replicate real-world systems' complexity and emergent properties. Due to their flexibility, ABMs are valuable tools for exploring hypothetical scenarios, conducting experiments and testing theories in a controlled, virtual environment. This means that ABMs are increasingly being used by policymakers to inform decision-making processes (Belfrage et al., 2022).

2.3.1. Challenges in Agent-Based Modelling

However, despite its advantages, ABMs face certain challenging obstacles that constrain their applicability and resilience. For instance, it can be difficult to run complex models in an ABM framework because of the computational demands (Bashardoust et al., 2023). Parametrisation is a common challenge as it requires the specification of parameters that govern agent behaviour, interactions and environmental factors. Estimating these parameters (based on empirical data) can be a challenge. On top of that, ABMs often involve the modelling of a system with a large number of interacting agents with their own set of behaviours, attributes and decision-making processes which brings great complexity. Managing this complexity and ensuring that the model accurately reflects the real-world system it aims to simulate can be a challenge. Inverse Modelling could provide solutions for these challenges, as elaborated earlier in this chapter. ABM also knows the challenges related to its community. The primary criticism on this front that is often directed at ABMs is their tendency to be constructed as standalone models (Turgut & Bozdag, 2023). Conversely, researchers advocate for the sharing, extension and reuse of ABMs across various studies. This research aims to contribute to this challenge within the ABM community by reusing an existing ABM on solar PV adoption. This will be further elaborated on in Chapter 5. Finally, ABM often struggles with stochasticity, where randomness influences simulation outcomes. A fixed random seed ensures reproducible results, which is crucial for validating and comparing models. Without a fixed seed, each simulation run can yield vastly different outputs, complicating the analysis of the behaviour of the system. Identifying optimal parameters becomes challenging as each iteration produces different results, increasing computational complexity and uncertainty. Effective seed control can mitigate these issues by providing consistent results, simplifying the inverse modelling process, and facilitating robust comparisons. This issue with stochasticity becomes especially challenging when doing inverse modelling since this multiplies the issues. Thus, controlling stochasticity is essential for reliable and meaningful applications of IM when using ABMs.

2.3.2. ABM Development

Agent-based models have three primary elements: agents, environment, and time (Nikolic et al., 2013a). Agents are individual entities that interact with each other and the environment according to predefined rules or behaviours. It is the smallest element of an ABM. Each agent has a certain state, which refers to the parameter collection that belongs to an agent at a certain moment in time. It is specific to the agent and can change over time.

According to Jennings (2000) and Nikolic et al. (2013), agents have the following characteristics:

- Agents are encapsulated, possessing distinct identities with clearly defined boundaries.
- They operate within specific environments, receive information from and react to it.
- Agents are flexible, being able to adapt to changes and proactively responding to stimuli.
- They are autonomous, exercising control over their internal states and behaviours.
- Agents are goal-oriented, striving to fulfil objectives, solve problems or achieve defined goals.

Agents operate in an environment where they are provided with inputs and receive output from agent actions. Agents reside in this environment and it includes elements beyond the agent itself, such as other agents, making an agent's environment context-dependent (Nikolic et al., 2013a). Agents can influence and be influenced by the environment based on their action rules (Macal & North, 2009).

The final element in an agent-based model is time. In real-life systems, agents can interact concurrently and continuously (Nikolic et al., 2013a). However, computational limitations require modelling time discretely constraining parallel interactions based on software and hardware capabilities. It is crucial to acknowledge these constraints to develop models that effectively represent reality despite these limitations (Hammond, 2015).

According to Klügl (2009) and North & Macal (2007), developing an ABM goes as follows: firstly, theories regarding the modelled system should be sought based on empirical evidence. Then, a conceptual model should be built. This means agents need to be classified, the interaction between these agents and the environment needs to be defined, and agent behaviour needs to be defined. Thereafter, the ABM should be constructed using conceptualisation. Then, the conceptual model should be implemented in a computer using simulation software.

2.3.3. ABM Classification

Agent-based models can be classified by their intended objective. Klügl (2009) identifies three types of objectives: prediction, optimisation and understanding. ABMs that have *prediction* as their main objective aim to project future states or behaviours of the system being modelled. They are characterised by the capacity to foresee specific elements of data that are currently unknown, with a dependable level of accuracy, utilising a computational model (Ale Ebrahim Dehkordi et al., 2023). *Optimisation* is the second objective of ABMs and it aims to identify the most favourable or efficient configuration of system parameters to achieve certain objectives (Turgut & Bozdag, 2023). As ABMs can be used for the exploration of scenarios and parameter settings, optimal solutions that maximize desired outcomes (or minimise undesirable ones) can be determined. Finally, ABMs with the objective of *understanding* aim to improve insights into the underlying mechanisms and dynamics of complex systems. By simulating the interactions of individual agents and observing emergent patterns and behaviours, one can develop a deeper understanding of how the system functions and how its components influence one another. In this research, the focus will be on the final type of ABM with the objective of *understanding*. It is good to note that ABMs can be created for multiple objectives.

2.4 Relations Between Concepts

In Figure 5 below, a simple overview of the concepts is presented. Understanding solar PV adoption in Dutch municipalities is the main objective of this study. As mentioned earlier in this chapter, inverse modelling can improve this understanding. The basic structure of the inverse modelling process is presented as the green-dotted box. Within this process, parameters are selected using a Machine Learning algorithm. This ML algorithm enables a smart search of the parameter grid to select new parameters for each run. These parameters are then given to the agent-based model, which generates a certain output. This output is then compared to data (in this case, real solar PV adoption data), and the comparison is fed back to the algorithm, which suggests new parameters.

Agent-based modelling (in the forward modelling context) in itself also can improve understanding of solar PV adoption but in a different way. On the one hand, ABM allows for better exploration of emergent behaviour, scenario analysis and policy evaluation. Inverse modelling, on the other hand, allows for more data-driven insights based on empirical evidence and more targeted policy implementation. Neither one of these concepts is better than the other, they are simply different methodologies. ABM can, however, be a means for inverse modelling by iteratively refining the model parameters to match observed data. Machine learning can also serve as a means in this process since it can enable efficient parameter grid search.

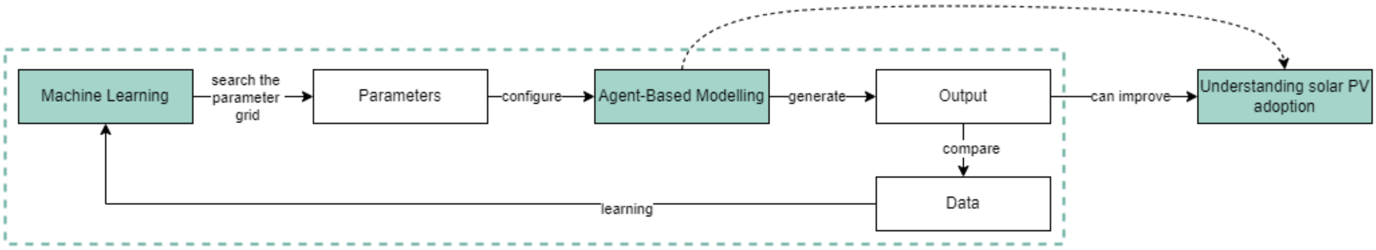


Figure 5: Relations between the concepts

2.5 Literature Review

To identify the research gap, a literature review is conducted following the PRISMA 2020 guidelines for systematic review reporting (Page et al., 2021). For this literature review, the search engine Scopus is used and only English articles are selected. On top of that, due to the novelty of the topics in this research, only articles that were published after 2016 that are accessible through the TU Delft institution are considered. This literature review consists of four rounds, each with its purpose. The purpose of the first round is to validate the lack of understanding of the complexity behind solar PV adoption. After this is validated, the potential for the use of inverse modelling for agent-based modelling will be further analysed. After validation of this second round, the third round tries to identify the potential for machine learning algorithms in ABMs for case studies on complex systems (such as solar PV adoption). Finally, the potential for using machine learning as a *means* for the application of inverse modelling to ABMs is validated.

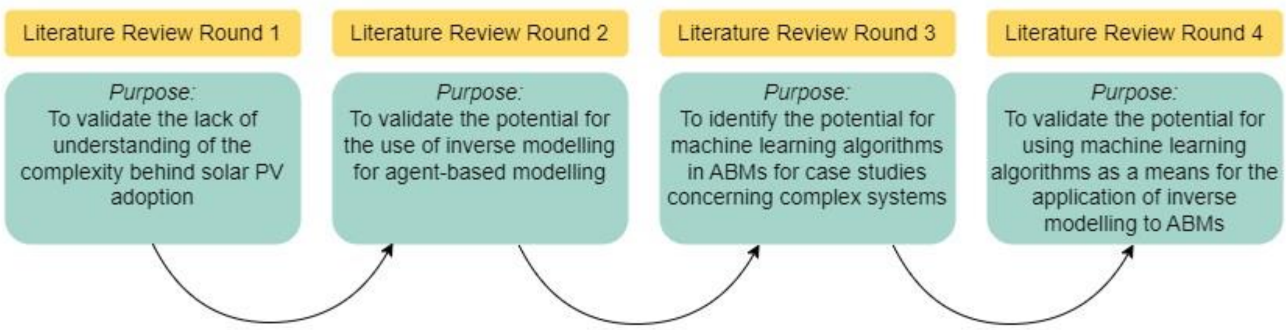


Figure 6: Rounds in the literature review

The first step of the literature review, the identification, resulted in a total of 89 hits on Scopus. Based on an initial screening of the abstracts of these researches, 59 articles were excluded due to irrelevancy. Thereafter, the remaining 30 articles were screened more thoroughly, which resulted in the exclusion of 17 more papers. The most important reasons for exclusion were a too-large focus on data science, authors using ABM in ML instead of ML for ABM, and using ML, IM, and ABMs in separate parts of the research. Finally, 2 more articles were excluded since they were duplicates between rounds. Additionally, 7 more articles were selected through snowballing, resulting in a final selection of 17 articles. The review was conducted on the 8th of March 2024.

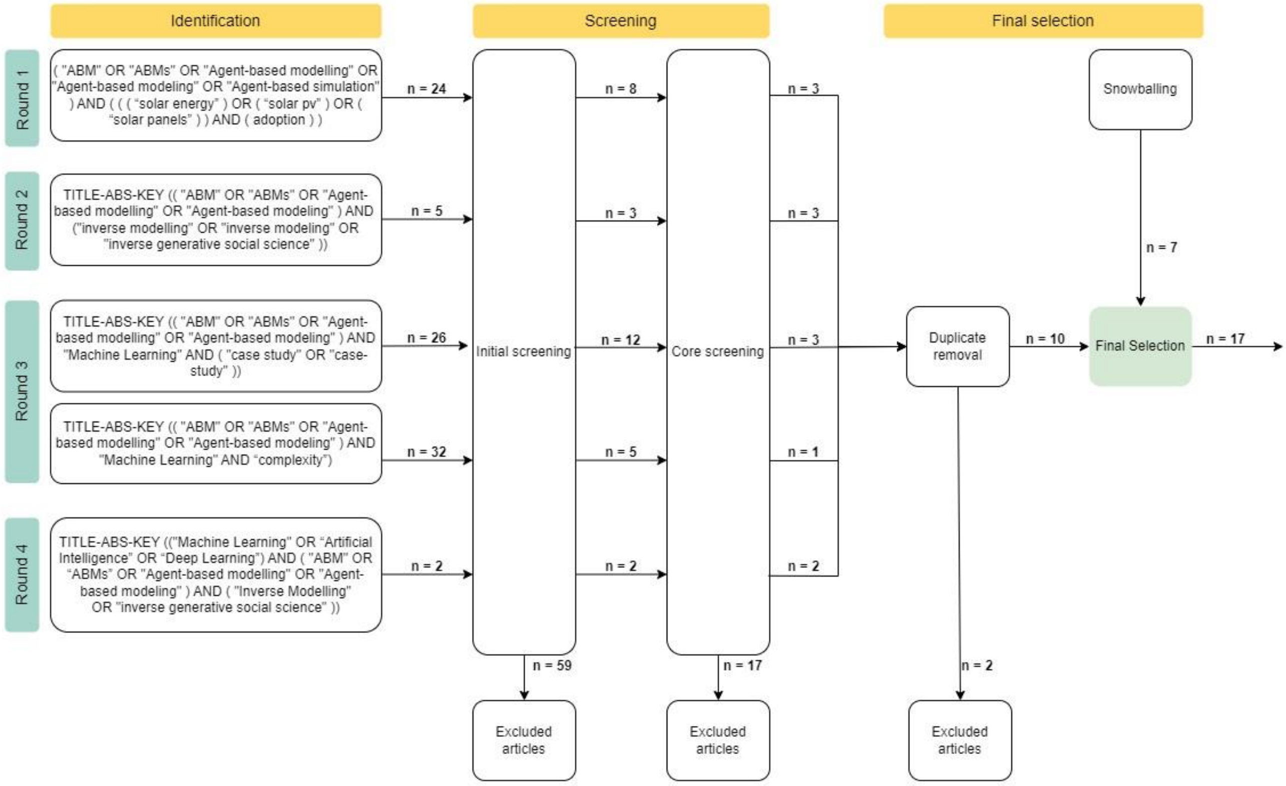


Figure 7: Prisma 2020 diagram of the literature review

The findings of the literature review are concretely described in Table 2. The academic knowledge gap is then identified based on the review, which results in the formulation of the research question.

Round 1:

The transition to renewable energy heavily depends on accurate residential solar PV adoption predictions. Ardila et al. (2022) assess the application of agent-based models in studying renewable energy transitions. This paper provides a broad overview of the complexities in renewable energy adoption, including solar PV, through ABMs. It emphasises the critical role of reliable adoption data in shaping energy policies and decision-making processes. It notes that the precision of such data can significantly influence the strategic planning of the energy transition.

Nuñez-Jimenez et al. (2023) delve into the uncertainties surrounding solar PV adoption, attributing them partly to the scarcity of case studies employing agent-based modelling (ABM) to simulate these dynamics. This lack of comprehensive ABM studies limits understanding of the complex factors influencing solar PV adoption, such as socioeconomics, demographic and geographic variables. N. Zhang et al. (2022) highlight the complexity behind consumer behaviour and decision-making in solar PV adoption. They argue for enhancing ABMs by integrating high-quality data sources, influencing Geographic Information Systems (GIS) data, and detailed information on housing and homeowner characteristics. This paper suggests that these improvements can lead to a deeper understanding of the patterns and drivers behind solar PV adoption.

Ardila et al. (2022) also critique existing ABMs for not fully capturing the inherent complexity of solar PV systems. They advocate for the introduction of additional dynamic layers to these models to reflect the real-world complexities of the energy transition. Akhatova et al. (2022) use agent-based modelling to analyse the adoption of solar PV systems. It examines various factors influencing adoption, using a case study approach to illustrate these dynamics and explores the complexities behind solar PV adoption. Macal's (2016) paper offers a comprehensive overview of methods and techniques used in agent-based modelling for simulating social systems. It covers the fundamentals of ABM and its applications in various domains, including energy systems. Furthermore, the paper's broad coverage also includes discussions relevant to integrating inverse modelling with ABMs. Both Akhatova et al. (2022) and Macal (2016) stress that this increasing complexity necessitates careful parametrisation and a thorough understanding of the system to ensure that ABMs can accurately simulate these systems.

Round 2:
Inverse modelling offers a promising approach to exploring the underlying dynamics of complex systems by adjusting model parameters to align with observed outcomes. The foundational paper by Epstein (2023) proposes the concept of Inverse Generative Social Science, where instead of simulating forward to see what might happen, models are used inversely to identify the initial conditions and parameters that lead to observed outcomes. This approach is particularly useful in agent-based modelling. According to this paper, inverse modelling can be used to gain deeper insights into complex social phenomena, which can particularly be beneficial for understanding the intricate factors driving residential solar PV adoption.

Vu et al. (2023) support Epstein's advocacy for a new era of model exploration aimed at uncovering the hidden mechanisms of complex social systems. They emphasise that inverse modelling could address the challenges identified in Round 1 by providing a more nuanced understanding of the adoption dynamics. Lastly, Greig et al. (2023) provide empirical support for the use of inverse modelling in ABMs. They apply Genetic Programming as as a method for it in ABMs, demonstrating its effectiveness in improving model accuracy and uncovering the underlying processes in complex systems. This case study underscores the practical utility of inverse modelling techniques in refining and validating ABMs.

Round 3:
The purpose of the third round of the literature is to prove the potential for machine learning algorithms in ABMs for case studies concerning complex systems. The reason for this is that if the literature proves that this is not possible, this has substantial consequences for the application of ML in IM for this case. The literature, however, proves that there is definitely potential for ML in ABMs.

Ale Ebrahim Dehkordi et al. (2023) provide a comprehensive review of the application of ML in ABMs. They categorise various ML techniques and their applications, illustrating how these technologies can be used to achieve different objectives, handle diverse data types and support various learning processes within ABMs. Bogner et al. (2020) discuss how ML can be used to enhance ABMs, focusing on improving model accuracy, scalability, and predictive power. It identifies challenges and opportunities in integrating ML with ABMs for various applications.

Turgut & Bozdag (2023) take this further by proposing a framework for integrating machine learning into ABMs. Their framework outlines how ML can be systematically employed to enhance the performance and capabilities of ABMs in simulating complex systems. This includes improving model precision, reducing computational costs and enabling real-time analysis. Nugroho & Uehara (2023) discuss the performance enhancements that ML can bring to ABMs. They highlight how ML can optimise the processing of large datasets and facilitate more accurate and efficient simulations. The paper by Y. Zhang et al. (2018) demonstrates the application of ML algorithms to ABMs in urban settings, which can be analogous to the complexities found in solar PV adoption. Harder et al. (2023) use ABM to examine the adoption of renewable energy technologies, including solar PV. They look into how economic, social and policy factors influence adoption decisions and diffusion patterns.
Both Y. Zhang et al. (2018) and Harder et al. (2023) recognise the transformative impact of ML on the scalability and prediction capabilities of ABMs.

Frank et al. (2022) research the combination of ABM and ML to analyse energy systems and thereby focus on how ML can improve the accuracy and efficiency of ABMs in predicting energy system behaviours and transitions. They also acknowledge the potential of inverse modelling but caution against the risk of overfitting, where models become too tailored to specific datasets, potentially losing their generalisability. This highlights the need for a balanced approach in applying inverse modelling to avoid compromising the model’s robustness and applicability across different scenarios.

W. Zhang et al. (2023) focus on the potential of ML in modelling complex systems. They argue that ML algorithms can manage the high-dimensional data and intricate interactions of such systems, making them ideal for enhancing the capabilities of ABMs. Finally, An et al. (2023) address the uncertainties and challenges associated with integrating ML into ABMs for urban energy systems. They emphasise the need for careful consideration of when and how to apply ML techniques and the types of ML most suitable for different modelling scenarios. This underscores the importance of the strategic integration of ML into ABMs.

Round 4:
The final round synthesises the insights from the previous rounds to explore the potential of using ML for inverse modelling in ABMs. This involves combining ML's advanced parameter estimation capabilities with ABM's sophisticated system simulations.

Greig et al. (2023) and Epstein (2023) both recognise the potential of ML as a tool for inverse modelling within ABMs. They argue that ML can enhance the accuracy and efficiency of inverse modelling processes by leveraging its powerful data processing and pattern recognition abilities. Epstein (2023) suggests that ML should not be seen merely as an end in itself but as a means to facilitate more effective inverse modelling. By using ML to refine the parameters and dynamics of ABMs, researchers can achieve a deeper and more accurate understanding of complex systems like solar PV adoption. Greig et al. (2023) provide a practical example of this synergy by using Genetic Programming to perform inverse modelling in ABMs. Their results demonstrate the significant improvements in the model’s ability to capture complex system behaviours that can be achieved through this integrated approach.

All in all, this round concludes that ML should not be seen as a prime focus point in this research; it should be seen as a tool to enable inverse modelling. The most important knowledge gap, however, that can be identified from this round is the novelty of the topic. The lack of application of inverse modelling to ABMs proves the essence of this research.

Through the identification of the knowledge gaps in the literature review, the following research question has been formulated to address these challenges:

How can inverse modelling contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities?

Table 2: Research gap identification

	Round *= snowballing	**Application of ML in inverse problems**	**Understanding of complex social phenomena with ABMs**	**Application of IM to real-world data**	**Application of ML algorithms in ABMs**	**Understanding solar PV adoption**
(Akhatova et al., 2022)	1					X
(Ale Ebrahim Dehkordi et al., 2023)	3		X		X	
(An et al., 2023)	*		X		X	
(Ardila et al., 2022)	1					X
(Bogner et al., 2020)	*		X		X	
(Epstein, 2023)	2 & 4	X				
(Greig et al., 2023)	2 & 4			X		
(Frank et al., 2022)	*			X		
(Harder et al., 2023)	3				X	
(Macal, 2016)	*		X		X	
(Nugroho & Uehara, 2023)	3		X		X	
(Nuñez-Jimenez et al., 2023)	*					X
(Turgut & Bozdag, 2023)	3		X		X	
(Vu et al., 2023)	2	X			X	
(N. Zhang et al., 2022)	1					X
(Y. Zhang et al., 2018)	*		X		X	
(W. Zhang et al., 2023)	*		X		X	

3 Methodology

Addressing the research question necessitates access to data, which is acquired through research methodologies. Selecting the appropriate method is crucial as incorrect choices could yield data that is unsuitable for addressing the main research question. This chapter delves into the research methodologies that will be employed during the study. The structure of this chapter is as follows: Firstly, the research approach will be presented, after which the research framework will be presented and discussed. Thereafter, the scope of the case study will be presented. Finally, this chapter will focus on how machine learning can be used as a means for inverse modelling.

3.1 Research Approach

To address the research gap identified in the literature review in Chapter 2, a state-of-the-art agent-based model that applies inverse modelling will be created. This approach addresses all identified research gaps as follows: **An existing ABM** on solar PV adoption will be modified so that it fits the case. This ABM can help **understand the complexity behind solar PV adoption**. As became clear in the literature review, inverse modelling can help in understanding complex dynamics underlying complex phenomena. Therefore, an **inverse modelling approach** will be used to understand these complexities behind solar PV adoption and **inverse modelling will be applied to the ABM.** Case-specific solar PV adoption curves will be integrated to serve as training data. This subchapter focuses on the modelling approach. Since the ABM that will be used requires more attention, this topic will be discussed separately. This means that this subchapter can be seen as more high-over. Further elaboration on the ABM can be found in Chapter 5.1. The general elaboration of machine learning as a means for inverse modelling can be found in this chapter, in Chapter 3.4. The actual application of inverse modelling can be found in Chapter 5.

This research employs a modelling approach whilst using a case study. By employing a modelling approach, the complex socio-technical system of solar PV adoption can be visualised and the impact of certain system interventions can be analysed, therewith enabling the opportunity for understanding the complex dynamics underlying the system (Nikolic et al.,2013). The advantage of using a modelling approach is the wide exploration of the impacts of different social, technical and institutional arrangements on solar PV adoption. This offers nearly limitless possibilities for visualising system interventions. However, the main disadvantage of taking a modelling approach is an inherent imperfection in models, as emphasised by Box (1976): “All models are wrong, but some are useful”. This means that models will never show exact similarities with reality. The accuracy of model assumptions determines its proximity to reality. This means also, that by employing a modelling approach, the results will not be perfect. This is a limitation that ought to be kept in mind.

For this specific research, an ABM on solar PV adoption has been selected from the Computational Model Library at CoMSES (2024). A quick research review on CoMSES found only 3 open-source models that included solar energy. One of these models, however, focused on the charging behaviour of EV (electric vehicle) drivers and only included solar PV minimally (van der Kam et al., 2019). Another model, by van der Kam et al. (2023), could be considered to be a good model for solar PV adoption, however, it relied too heavily on survey data. Therefore, this model could not be used for reproduction in this study. In conclusion, only 1 model could be selected, namely the model by Muelder & Filatova (2018). This model will be further elaborated on in Chapter 5.1.

The inverse modelling process includes changing the values of input parameters. Changing the values of these input parameters changes the output of the model. This model output can be compared with certain training data. In this case, the training data is equal to solar PV adoption data, which will be further elaborated on in Chapter 4. Since this training data resembles reality, a certain error value can be generated that describes how well the model predicts solar PV adoption. By applying inverse modelling, the input parameter values that generate the smallest error value can be found, creating more insight into the dynamics underlying solar PV adoption.

For the existing ABM, a variety of parameters will be included in the inverse modelling process. The parameter space of the ABM by Muelder & Filatova (2018) includes 4 parameters. The original ABM included more parameters, however, these have been excluded. The reason for exclusion is that this study aims to form a basis for future inverse modelling research. Therefore, the aim is not to make this study as perfect as possible (which can be achieved by including more parameters to vary) and including more parameters will not contribute to this ambition. The (exclusion of) parameters in the parameter space of the model will be further elaborated on in Chapter 5.1.

3.2 Research Framework

The research framework is presented in Figure 8. It summarizes the full research framework for studying the exploration of solar PV adoption using inverse modelling. This thesis relies on a structured approach that links its various components such as the main research question, research steps and outcomes such as certain data and chapters in the thesis. The research framework can be elaborated in the following process phases:

Problem Identification: The problem identification phase consists of three chapters within the thesis. This phase aims to identify the research problem, the research approach, the research gap and proposes a methodology. A literature review is conducted in this phase to improve the understanding of the concepts of inverse modelling, solar PV adoption and agent-based modelling. Scientific literature is used as input for this phase.

Model Conceptualisation & Formulation: The second phase starts with the process of finding data to compare the predictions of the ABM. On top of that, relevant ML algorithms are identified in this phase that will be used to search the parameter grid. Thereafter, the model conceptualization and formulation phase focuses on adapting the existing ABM to fit the case of this research. This means that certain adaptations in NetLogo (sometimes with the help of self-written Python scripts) need to be made so that the ABM in NetLogo is ready for further software implementation. On top of that, a conceptual model is created that schematically presents the next steps for the ABM model. This means that it presents the general steps within the Python model. Finally, the formal modelling step will take place. This includes writing the pseudo-code for the model.

Why Python?

There are various reasons why Python is used for further processing of the ABM in this research, and not NetLogo. These reasons are all mostly related to Python's adaptability and more extensive capabilities. Due to its versatility, integration of ABM with advanced data analysis and ML techniques is more efficient in Python. On top of that, Python offers a large number of libraries, facilitating a more efficient application of the inverse modelling process than NetLogo. Finally, Python is more compatible with other tools and workflows, which improves the reproducibility of this research.

Software Implementation: The software implementation phase is where the previous phases come together. The new knowledge about inverse modelling, ABM, machine learning techniques and solar PV adoption come together with the conceptual model and demographic data in the Python model.

This model enables the inverse modelling process. In the software implementation phase, programming takes place, and a schematic overview of the different code files, their inputs, outputs, and interdependencies will be presented.

Validation & Verification: As the title suggests, this phase involves validating and verifying the Python model. Verification takes place through static code analysis using Pylint and code review. Validation takes place through the creation of a model validation file in Python. This means that the output of this phase is a verified and validated model ready for data analysis.

Data Analysis: The fifth phase focuses on data analysis, which will be qualitative rather than quantitative. Using data analysis, the research results can be interpreted, and a better understanding of the role that inverse modelling can play in solar PV adoption dynamics can be gained. After this phase is completed, an answer to the research question can also be formulated.

Application: The insights from the data analysis phase can then be used in the final phase, namely application. This phase presents the policy implications of the results, a conclusion, and a critical discussion about this research. It also presents recommendations for future research and this thesis's scientific and societal relevance.

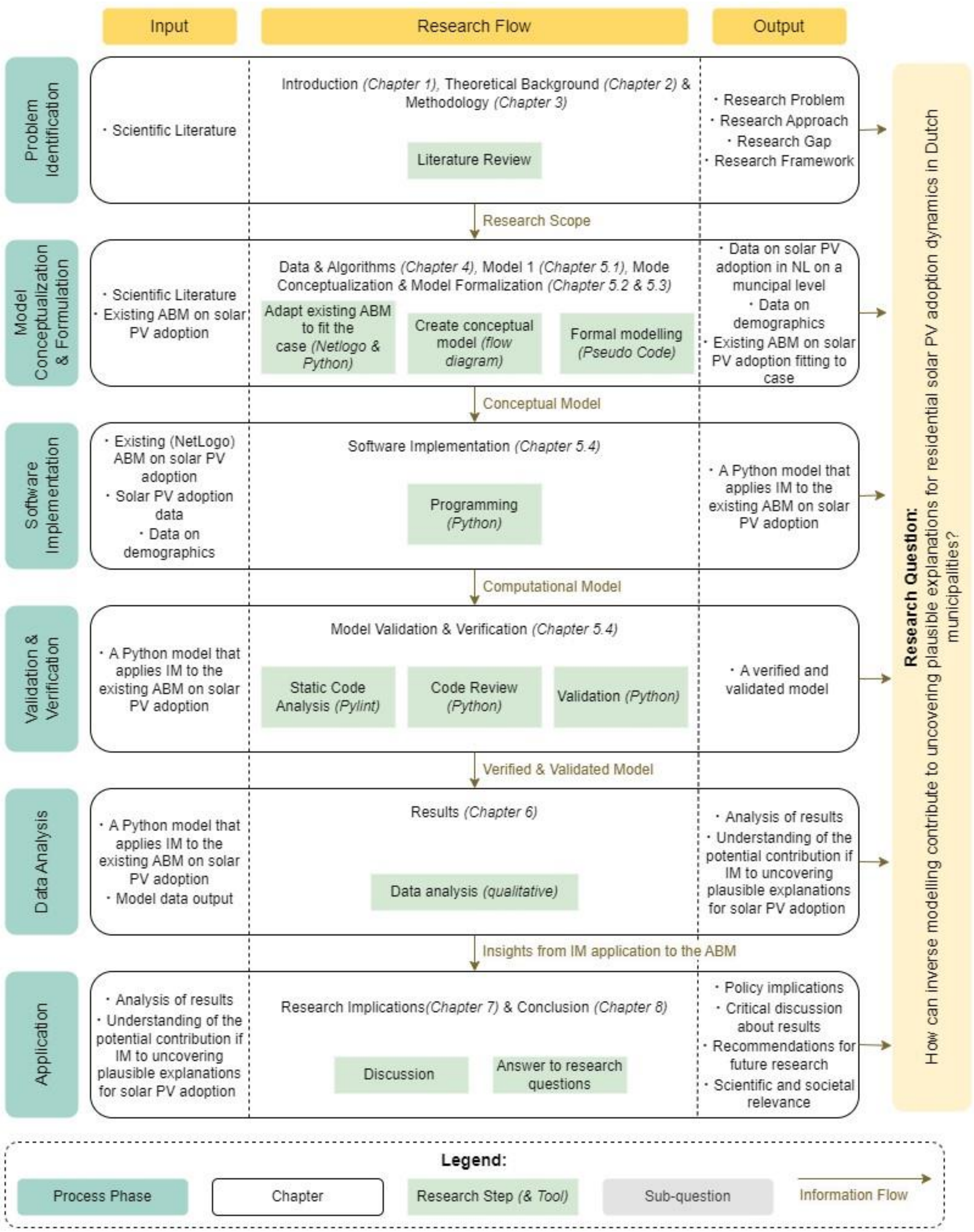


Figure 8: Research Framework

3.3 Scope of the Case Study

For the development of this thesis, a case study is chosen as a suitable research approach for the exploration of inverse modelling. According to Eisenhardt (1989), a case study is particularly applicable when little is known about a phenomenon. This applies well to the case of residential solar PV adoption in the Netherlands because the rapid shift in public opinion towards solar energy calls for further research on the underlying dynamics of this social phenomenon. On top of that, case studies also prove valuable when aiming to examine particular systems with strongly defined boundaries. As outlined by Flyvbjerg (2011), opting for a case study directs research towards an individual unit, thereby establishing the limits of what this unit encompasses. In the context of this study, the “unit” is limited to the geographical scope of the Netherlands, exemplified by a selected municipality sample.

Using a case study offers various advantages, such as improved empirical validity because it closely aligns theory-building with evidence (Eisenhardt, 1989). Furthermore, it presents the benefits of minimizing researcher bias and increasing the probability of uncovering novelty. This latter advantage is particularly crucial in this context since the focus of this study is explorative. Nonetheless, case studies also come with certain drawbacks, such as the potential for collecting irrelevant data and the necessity of fitting within theoretical frameworks (Crowe et al., 2011). These limitations can be addressed by justifying the choices made since this creates in-depth knowledge and allows unexpected issues to arise. Yin (1994) highlights the importance of contemporary events in case studies, although, in this research, much of the significant growth in solar PV in the Netherlands has already occurred. Balancing these strengths, a modelling approach employing a case study is considered appropriate, recognising the complementary nature of both methodologies.

As mentioned before, the scope of this research is solar PV adoption in the Netherlands. This scope will, however, be limited to a municipal level due to practical and data availability considerations. A total of 6 municipalities will be considered in this study. The selection of this number of municipalities as the focal point for the case study is primarily motivated by data considerations. Specifically, the necessity for highly specific geographic information and data about roof sizes (as is required for the model by Muelder & Filatova (2018)) dictates the choice. These essential datasets are obtainable from TUDelft3d (2024), which provides comprehensive information on every building in the Netherlands. The extensive scope of information contained within this website encompasses data for all buildings in the country, resulting in a substantial file size. To effectively manage the data and delineate clear system boundaries, the decision is made to select only municipalities that have less than 75.000 inhabitants and to select only 6 municipalities. This choice is guided by its manageable scale and feasible fit within the partitioned patches of the information map. By focusing on these aspects, the study can establish distinct and manageable parameters, facilitating a more systematic modelling approach.

In the process of selecting municipalities for this case study on residential solar PV adoption in the Netherlands, a variety of criteria can be considered. However, seeing the practical limitations of this study, only 3 criteria are used. The research by Stake (2005) provides insights into the process of case selection, emphasising the importance of considering socio-economic characteristics when choosing research cases. On top of that, the selection criteria that will be used for the case study should be parameters that can be changed in the model that will be used in the research. Considering these two factors, the following criteria emerge: income, population density and solar PV adoption per 1000 households. It should be noted that in the selection of the municipalities, only municipalities with a number of residents lower than 50.000 are considered due to computational abilities.

Income level is an important factor because it functions as a representative of household wealth and purchasing power, which substantially influences the affordability and accessibility of solar PV systems. For this criterion, the municipalities of Bloemendaal and Vaals are considered since these are the two municipalities having the highest (Bloemendaal) and the lowest (Vaals) median income out of all municipalities with less than 50.000 inhabitants (CBS, 2023a).

The second criterion to be considered is population density. The municipalities of Westerveld (least population-dense, excluding the Dutch islands) and Oegstgeest (most population-dense) will be used for the case study (CBS, 2023b).

The final criterion is the number of solar PV installations per 1000 households. The municipalities of Dantumadiel (with the most) and Laren (with the fewest) are selected as cases for this criterion.

All of these municipalities are presented in the table below. The median income presented in this table is based on CBS (2023a) and the number of residents is based on KadastraleKaart (2024). Appendix C presents an overview of the income distributions in these municipalities, which are later used in the ABM.

Table 3: Overview of the selected municipalities for the case study

Municipality	Province	Number of Residents	Median Income
Bloemendaal	Noord-Holland	23.159	€51.400
Dantumadiel	Friesland	19.135	€34.650
Laren	Noord-Holland	11.195	€48.250
Oegstgeest	Zuid-Holland	25.939	€47.750
Vaals	Limburg	10.190	€29.550
Westerveld	Drenthe	19.348	€38.750

3.4 Machine Learning as a Means for Inverse Modelling

As mentioned before, inverse modelling can be used to explore the dynamics underlying solar PV adoption. It involves the process of inferring the parameters of a model that best explains observed data, thereby providing insights into the factors driving solar PV adoption. One way to perform inverse modelling for solar PV adoption is by using machine learning techniques (Vu et al., 2023). This can work as follows: ML techniques, such as Bayesian Search and Random Search, can iteratively explore the parameter space which is defined by the solar PV adoption models. It samples parameter configurations, runs the model with these configurations, and evaluates the model's performance against observed data. In this sense, the objective function can be defined as the error between the (real-world) observed solar PV adoption data (training data) and the model predictions generated using the current parameter configuration. The goal is to minimise this error and, therefore, effectively improve the ability of the model to simulate real-world adoption dynamics. This process enables inverse modelling and can create better insights into the dynamics underlying solar PV adoption.

For instance, say, Bayesian Search is used as a ML algorithm. As the Bayesian Search progresses, it refines its estimate of the optimal parameter values based on the observed, real-world (training) data. This iterative process continues until convergence or until a predefined stopping criterion is met. In this case, two splits (see Chapter 5.4.2. for reasoning) and 15 iterations will be done for a total of 25 runs (see Chapter 5.4.2. for reasoning). The stopping criterion is therefore mostly based on computational abilities.

3.5 Chapter Summary

This methodology chapter outlines the research approaches and frameworks used to research solar PV adoption on a municipal level in the Netherlands. It begins with the research approach, detailing how an ABM will be employed and modified to fit the specific case study. This model uses inverse modelling to adjust input parameters, aiming to provide insights into the underlying dynamics. The research framework involves several structured phases. Initially, the problem is defined, and a literature review supports the methodology. Model Conceptualization adapts an existing ABM and identifies relevant machine learning algorithms. The Software Implementation phase integrates the model with Python, enhancing inverse modelling capabilities. The Validation and Verification phase ensures the model's accuracy, while the Data Analysis phase interprets results qualitatively to better understand solar PV adoption dynamics for the case. Importantly, these results are discussed as well. Finally, the Application phase draws conclusions and recommendations for future research.

This chapter also elaborates that the study focuses on six Dutch municipalities—Bloemendaal, Dantumadiel, Laren, Oegstgeest, Vaals, and Westerveld—selected based on income level, population density, and solar PV adoption rates. Two machine learning techniques, Bayesian Search and Random Search, are used for inverse modelling.

4
Data & Algorithms

This chapter comprehensively explores the foundational pillar of inverse modelling: data requisites. Understanding complex systems through modelling starts with collecting important data. Therefore, this chapter will first examine the key data necessary for inverse modelling and how it can be used. Thereafter, the machine learning algorithms that could be utilised to enable inverse modelling will be explored.

4.1 Data requisites

The data that is required for the inverse modelling process can roughly be divided into 2 categories: (1) necessary data for running the initial ABM model, and (2) necessary data for running the Python model. This section elaborates on this data. It should be noted, however, that the exact implementation will not be discussed in this chapter but rather in Chapter 5.

4.1.1. Necessary Data for Running the Model

The agent-based model that is used in this research is the NetLogo model by Muelder & Filatova (2018) which originally focuses on solar PV adoption in the municipality of Dalfsen in the Netherlands. This model has, however, several data limitations, which result in the model being unable to run. The reason for this is the fact that for the model datasheets, surveys and shapefiles were used that are not publicly available due to privacy reasons. Therefore, various adjustments need to be made to the model for it to run. However, not all data used in these datasheets, surveys, and shapefiles are essential for running the model. Therefore, the essential parameters that fully rely on these private data and that need to be adjusted have been identified. These are three parameters in total, namely *geolocation* of the households, *income* of the households and *roof sizes.* The exact modifications that have been done to the model will be discussed in Chapter 5. However, this chapter will go into further detail on the content of the data that is used to make the model run.

The first parameter is geolocation. In the original model, shapefiles have been used to identify the locations of houses in the municipalities. The spatial location and geographical distances between households in the model play a role when initialising the social network. This means that geolocation was the first adjustment that had to be fixed in the model. These geolocations for all houses were found via a webpage called 3DBag (TUDelft3d, 2024). Coincidentally, this page also contains the roof sizes of each house in the Netherlands. Due to its size, this webpage works with small patches that can be downloaded and that contain the required information. For the municipality of Laren for instance, a total of 8 tiles had to be downloaded. For the other municipalities, the number of tiles that had to be downloaded varied between 4 and 21. Each of the files of the patches contained, however, also a lot of unnecessary data that was not required for the implementation in the model of Muelder & Filatova (2018). Therefore, two Python scripts had to be written to extract the geolocation. The purpose of the first Python script was to extract the required data from the downloaded files from 3DBag, convert the geolocations to WGS84 form and convert this to a CSV and a shapefile. This script can be found in Appendix A. The purpose of the second Python script was to combine each of the individual patches into one larger patch. This, however, results in a larger amount of buildings than the number of municipalities. This is due to the fact that each of the municipalities is also home to several stores and company buildings. Therefore, it was implemented in the code to limit the amount of data points extracted to the number of households.

For the *roof sizes*, two Python scripts had to be written as well, serving the same purpose as the Python scripts for the geolocations. These can be found in Appendix A and B as well. By writing these scripts, one shapefile containing the geographical locations of the houses and one CSV file containing roof sizes were created which could be used in the model.

The final parameter that required new data is *income*. This data is acquired through CBS (2023a) and uses the median standardised income and the income distribution in each of the municipalities for 2022. These distributions can be found in Appendix C.

4.1.2. Data for Training The Model

Training data is essential for the process of inverse modelling. In the table below, the solar PV adoption data for the 6 municipalities is presented. The data that is used for training the model concerns the number of solar PV installations in residential homes throughout the years for each of these municipalities and is obtained from CBS (2020) and CBS (2023c). Data on the number of residential solar PV installations is only available from 2012 and is still incomplete for 2023, so only data from between 2012 and 2022 is used.

Table 4: Training data for inverse modelling

Number of solar PV installations (residential houses)											
Period	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**	**2021**	**2022**
Municipality											
Bloemendaal	75	136	203	300	378	453	612	794	1047	1447	1879
Dantumadiel	187	322	417	503	591	715	977	1360	2344	3228	4126
Laren	10	36	64	84	93	127	152	191	284	360	584
Oegstgeest	134	240	346	445	545	667	902	1407	1971	2461	3135
Vaals	0	108	173	217	255	439	534	734	916	1073	1238
Westerveld	141	296	427	920	1138	1706	2263	2717	3045	3440	4023

This training data is also presented in Figure 9 and Figure 10 below. Figure 9 is an exact visual representation of the data in Table 4. Figure 10, on the other hand, presents the number of solar PV installations per household for each municipality and, therefore, accounts for the sizes of the municipalities. From these figures, it becomes clear that most of these municipalities (especially Laren, Bloemendaal, Vaals and Oegstgeest) follow relatively similar curves, having a moderately exponential growth with varying degrees of increase over the years. The municipality of Dantumadiel follows this exponential curve as well, only steeper than for the previous four municipalities, indicating a slow start followed by a steep increase. The municipality of Westerveld presents a very interesting curve that is significantly different from the other municipalities, as it is sometimes sublinear and sometimes superlinear. This curvature presents a rapid increase in installations over the years, but the increase accelerates more gradually compared to the exponential growth of, for instance, Dantumadiel.

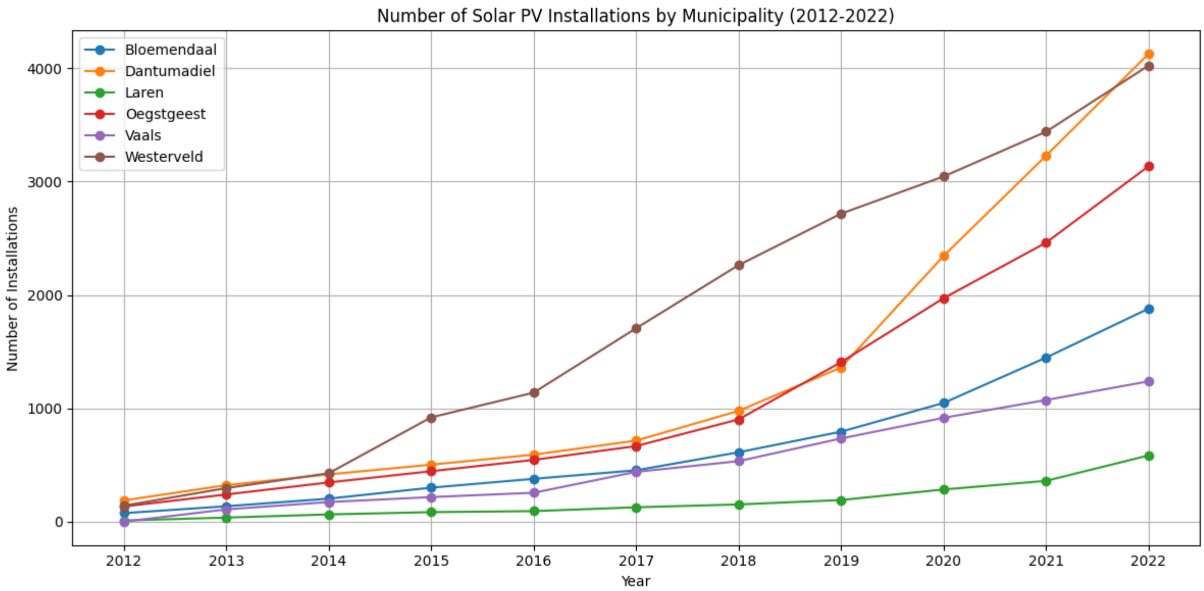


Figure 9: Number of residential solar PV installations per municipality (CBS (2020) & CBS (2023e))

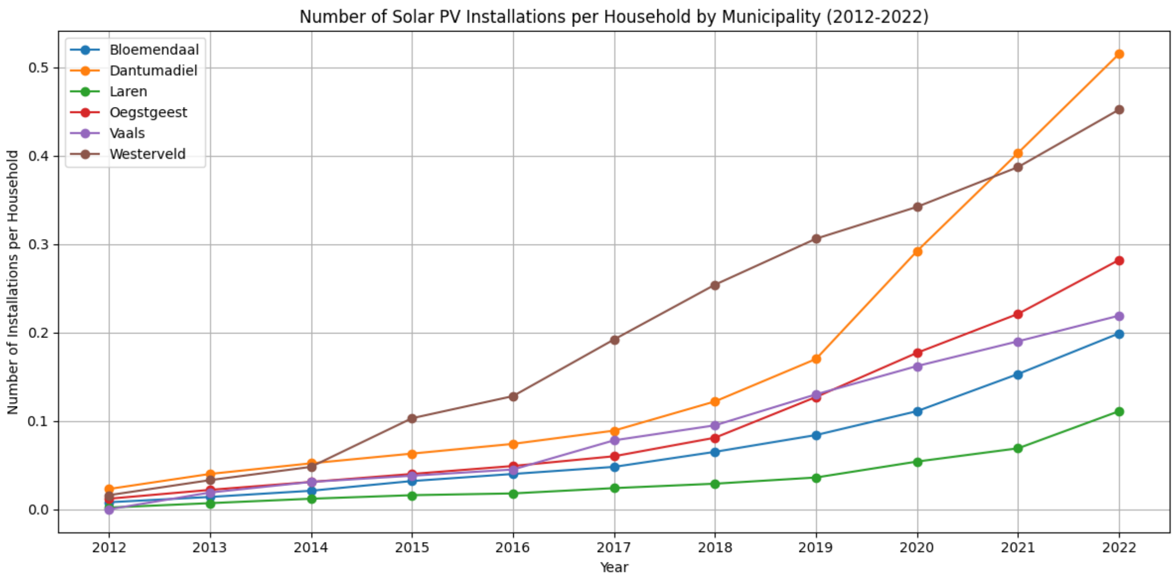


Figure 10: Number of solar PV installations per household (CBS (2020) & CBS (2023e))

4.2 Machine Learning Algorithms

As elaborated in Chapter 3, machine learning algorithms can be a means for inverse modelling in the pursuit of understanding the complex dynamics underlying solar PV adoption. These algorithms can be used to uncover patterns and insights with vast datasets. Among the diverse array of algorithms available, Random Search and Bayesian Search stand out for their versatility and applicability to inverse modelling tasks in this domain. These two algorithms will therefore be the two algorithms that will be applied to the models because of their ability to explore high-dimensional spaces while remaining computationally efficient (Bergstra & Bengio, 2012). Each of these two algorithms has different strengths and weaknesses. The Random Search algorithm for instance finds strengths in its simplicity, whereas Bayesian Optimisation finds strengths in its adaptability due to its ability to adapt the search based on previous evaluations. Below, the algorithms are elaborated more individually.

As mentioned before, many more algorithms could be applied to the model, such as random forests and neural networks. These algorithms might be better at capturing the complex patterns underlying the models. However, they are more complicated and less powerful in parameter optimisation. Since the focus of this thesis is socio-technical rather than purely within the domain of computer science, these other algorithms will not be further explored. They do, however, present intriguing opportunities for future research.

It should be mentioned again that the application of machine learning for this research is purely instrumental. This is another reason why only 2 algorithms are explored and not more. The data that is used for training the model is presented in Table 4.

4.2.1. Random Search

Random Search is a hyperparameter optimisation technique widely employed in machine learning to efficiently explore the hyperparameter space and identify optimal configurations for models (Bergstra & Bengio, 2012). Unlike exhaustive grid search, which evaluates every possible combination of hyperparameters, Random Search randomly samples from the parameter space θ (Andradóttir, 2006), making it computationally more efficient and suitable for high-dimensional problems like solar PV adoption modelling.

One of Random Search's key advantages lies in its ability to handle non-linear relationships and complex interactions between input features (Marmolejo et al., 2017). By systematically varying hyperparameters such as learning rates, regularisation strengths and network architectures, Random Search can uncover the most influential factors driving solar PV adoption and shed light on their interplay. Another advantage of Random Search algorithms is limited computational effort (Zabinsky, 2011). Random Search algorithms are rather popular due to their ability to swiftly and effortlessly yield a fairly good solution. Furthermore, the Random Search's stochastic nature allows it to explore diverse regions of the parameter space, thereby providing a more comprehensive understanding of the underlying dynamics. Through iterative experimentation and evaluation, researchers can refine their models and gain insights into the factors shaping solar PV adoption patterns.

The general structure of Random Search methods, as defined by Zabinsky (2011)**,** is as follows**:**

1. *Initialise*. Initialisation of the algorithm parameters (θ_{0}), the iteration index (k=0) and initial points ($X0 \subset S$), whereby S is a feasible region of *n* dimensions.
2. *Sample*. Generation of a set of candidate points ($V_{k+1} \subset S$) using a designated generator and its corresponding sampling distribution
3. *Simulate.* Update (X_{k+1}) using candidate points (V_{k+1}), previous iterations, and algorithmic parameters. Additionally, update algorithmic parameters to (θ_{k+1})
4. *Iterate*. If the stopping criterion is satisfied, terminate. Otherwise, increment *k* and go back to the first step.

Random Search involves sampling points from the hyperparameter space. Each point represents a specific configuration of hyperparameters. The idea is to explore the various regions of the hyperparameter space to find good configurations without exhaustively searching the entire space.
With a multiple-point generator, Random Search generates multiple candidate configurations in each iteration (Zabinsky, 2011). These configurations are then evaluated, and the best-performing one is selected for further exploration. Using multiple points in each iteration allows Random Search to explore the hyperparameter space more efficiently compared to a single-point generator, which evaluates only one configuration at a time.

In short, this means that the key strength of the Random Search algorithm lies mostly in its simplicity. The algorithm also knows, however, certain weaknesses. The first weakness of the algorithm is inefficiency. It may require a large number of evaluations to find the optimal parameters, especially in models with a high degree of dimensionality. The second weakness of the algorithm relates to a lack of guidance since this algorithm does not leverage information from previous evaluations to guide the search.

4.2.2. Bayesian Search

Bayesian Search can be seen as an extension of Random Search. It incorporates probabilistic models to guide the search process towards promising regions of the parameter space (Mortazavi, 2022). The technique relies on probabilistic surrogate models to estimate the behaviour of the objective function. By iteratively updating the surrogate model based on observed data, Bayesian Search intelligently explores the parameter space, which is the purpose of this research and therefore makes this algorithm very applicable.

One of Bayesian Search's notable strengths lies in its ability to adaptively adjust the search strategy based on previous evaluations, leading to more efficient exploration of the parameter space (Snoek et al., 2012). This adaptability is particularly advantageous in inverse modelling tasks, where the goal is to minimise the error between model predictions and observed data. Additionally, Bayesian Search provides uncertainty estimates for each evaluated point in the parameter space, enabling researchers to quantify the confidence in their model's predictions. This uncertainty quantification can be invaluable for robust decision-making and hypothesis testing in the context of solar PV adoption research. The final strength of the Bayesian Search Algorithm lies in its efficiency, especially relative to the Random Search Algorithm. Due to its ability to adapt the search based on previous evaluations and thereby focus on promising regions of the parameter space, it typically requires fewer evaluations compared to the Random Search algorithm to find good parameter values. The algorithm does, however, also know weaknesses. The first weakness is related to its implementation, which can be more complex compared to Random Search. Additionally, Bayesian Search is often more computationally intensive since it involves building and updating the model.

4.3 Chapter Summary

This chapter explores the data requisites that will play a fundamental role in the inverse modelling process. From the data requisite section, it becomes clear that for the agent-based model by Muelder & Filatova (2018) to run, three main parameters have to be adjusted, namely income, roof size and geolocation. Furthermore, the data that will be used to train these models has been identified. Solar PV adoption data is found for each of the municipalities that will be considered in the training process. Finally, the machine learning algorithms that will be considered to serve as a means for the inverse modelling process have been identified, namely: Random Search and Bayesian Search. The data that is used for training the model is presented in Table 4. In summary, it can be said that while Random Search is simple and easy to implement, Bayesian Search tends to be more efficient and effective. Therefore, a comparison between the results of both algorithms is interesting.

5
The Model

This chapter focuses on the modelling within this research. Firstly, the NetLogo model by Muelder & Filatova (2018) will be elaborated. Thereafter, the modifications that had to be done to this model in order to fit the objectives of this research will be disucussed. Finally, the model assumptions for this model will be discussed. Then, an explanation of the actual implementation of the entire inverse modelling process in Python will be given. This means that the existing NetLogo model by Muelder & Filatova (2018) is used as a basis for the inverse modelling process but that the IM process does not actually take place here since NetLogo does not have the capability to do efficient parameter calibration. Therefore, in order to make the IM process happen, a model in Python is created where parameter calibration is executed. To elaborate further on the relationship between the original NetLogo model and the new Python model, as well as the data and outcomes, a conceptualisation is presented in Chapter 6.2.1. This model conceptualisation, therefore, focuses on the entire modelling process within inverse modelling.

Finally, model validation and verification will be discussed in Chapter 5.4. It is good to note that in this chapter, the original model presented by Muelder & Filatova (2018) will be referred to as the 'original model', whereas the version of this model that has been adapted to fit the case of this specific research will be referred to as the 'modified model' or the ´Python model`.

5.1 NetLogo Model Elaboration

The model that is used in this research is the model presented in the work by Muelder & Filatova (2018). The study by Muelder & Filatova focuses on the transparency and consistency of ABMs in social science research. It highlights the challenge of translating qualitative social theories into quantitative models and the potential for varied interpretations. The study researches how different formalisations of a social theory, namely the Theory of Planned Behaviour, affect the outcomes of an ABM, using simulations focused on household solar PV investment decisions. This model specifically focuses on the municipality of Dalfsen in the Netherlands. The ABM that is presented in the study by Muelder & Filatova will be used as a basis for this research but it will be adapted to fit the case. It can collect data regarding the share of agents purchasing solar PVs and the total amount of energy generated by all installed PVs in the simulation. Therefore, this model can be used to apply inverse modelling to and hopefully gain a deeper understanding of the factors influencing adoption decisions.

5.1.1. NetLogo Model Modifications

In Chapter 4 it was discussed that initially, the model by Muelder & Filatova (2018) was not able to run. Therefore, the first modifications to the model are related to the agents' geolocation, income, and roof sizes. However, more errors were found in the model during the modelling process. These will be described in this chapter as well. This chapter will not dive deeper into these modifications since they have been elaborated on in the previous chapter.

Agents:
The model has one agent type, namely households. Each of these households has certain agent state variables and is characterised by a certain geographical location, household traits, involvement in a social network and decision-making processes. Appendix E includes these agent state variables from the modified model. This means that not all state variables are exactly equal to the original model. It should also be noted that the NetLogo code contains a lot more turtles-own variables than the amount of agent state variables listed in the appendix. The reason for this is that the original model lists many transfer variables as turtles-own. These transfer variables are not included for reasons of conciseness.
The income of the agents has been changed compared to the original model. Originally, a survey determined this income; however, this data is no longer available. Therefore, income data from CBS (2023a) was taken in the modified model. To balance the new number of households per income group, the income group thresholds were also modified. Originally these income group thresholds were €30.000 / year and €45.000 / year. However, this meant that the number of households in the income group > €45.000 a year was very large. Therefore, these income class thresholds were modified to €40.000 / year and €60.000 / year. As elaborated in Chapter 4, the roof sizes of the households have also been modified.
Finally, the initial PV share used in the model had to be adapted. Unfortunately, the way this parameter was utilised in the original model was not functional, so the code was changed to make it functional. Changes also had to be made to the setup and the go procedure for the model to work.

Environment:
The environment in this model is a municipality. So for instance for the municipality of Vaals, the environment is the municipality of Vaals. The interface, therefore, represents all households in the municipality at their geographical location and the links that are formed between these households (that form their social networks). Most of the environmental variables are presented in Appendix F. In Figure 12 the interface of this model is presented.

Again, several variables were left out since these were transfer variables. Some of the environmental state variables can, however, not be changed during the run. Their attribute levels are presented in Table 5. It should be noted that these attribute levels are based on their values as of 2012. Some of these variables, such as electricity prices, have changed a lot in the past decade. Furthermore, the PV costs are also considered stable during the runs, while in reality, these costs have decreased by over 80% since 2012 (IRENA, 2021). This forms a limitation to this study and would indicate that the NetLogo model likely underestimates the number of solar PV installations in a municipality.

Table 5: Attribute levels for environmental state variables that cannot be changed during the run

Variable	Attribute	Attribute levels	Reference
PV costs per m2	The costs of solar panels per m2	€1000 / m2	(IEA, 2020)
PV peak power	The maximum power in kW/m2	1 kW/m2	(Huynh et al., 2013)
Sunshine hours	Total sunshine hours per year	2209 hours	(Statista, 2022a)(Zonneplan, 2012)
Performance ratio	The overall performance efficiency of the PV system	0.6	(Ghazali M. & Abdul Rahman, 2012)
PV lifetime	The assumed lifetime of a PV system in years	25 years	(Tan et al., 2022)
Grid electricity costs	The electricity price	€0.26/kWh	(CBS, 2012)

Processes:
At the heart of the model lies the household's decision-making process, where they decide whether to invest in solar PV or not. This process occurs in every time step and for each of the households. Following each time step, the attributes of the households are adjusted and the cumulative outcomes are projected for the entire municipality. Moreover, the model offers three alternatives for depicting

the decision-making process, each based on different interpretations of the Theory of Planned Behaviour (TPB) (Muelder & Filatova, 2018). However, due to computational limitations, only one of these interpretations will be used in this research, namely the RR model presented in the consecutive studies by Rai & Robinson (2015) and Robinson & Rai (2015). In their studies, household decisions regarding the adoption of solar PV installations and the diffusion of PVs in a residential area in Texas in the USA were investigated. Within this analytical framework, household agents were conceived as rational actors, weighing various determinants including environmental consequences, economic viability and social influences.

Economic considerations are operationalised in the model through metrics such as payback period estimation, system profitability and anticipated monthly electricity bill savings. In the ABM instantiation, economic utility is introduced to encapsulate this aspect. Environmental impact includes dimensions such as CO_2 emission savings and households' environmental consciousness. Social influence is explained via metrics like the prevalence of PVs in the neighbourhood and interactions with PV owners. In the original papers by Rai & Robinson (2015), a comfort utility related to the psychological and aesthetic dimensions was not included. Muelder & Filatova (2018), do include this utility in their NetLogo model.

In the model, income is firstly compared to payback assessments through:

$$th_{inc} > u_{eco} \qquad (1)$$

If this PBC barrier is passed, households assess their potential utility of a PV investment decision using multi-attribute utility through:

$$U_{RR} = w_{eco} * u_{eco} + w_{env} * u_{env} + w_{soc} * u_{soc} + w_{cof} * u_{cof} \qquad (2)$$

The decision of the household to install PV or not is taken by a comparison between the result of the multi-attribute utilities associated with solar PV (U_{RR}) against a predetermined threshold value which is called "RR_{sia}" in the model. In Figure 11, this process is visualised.

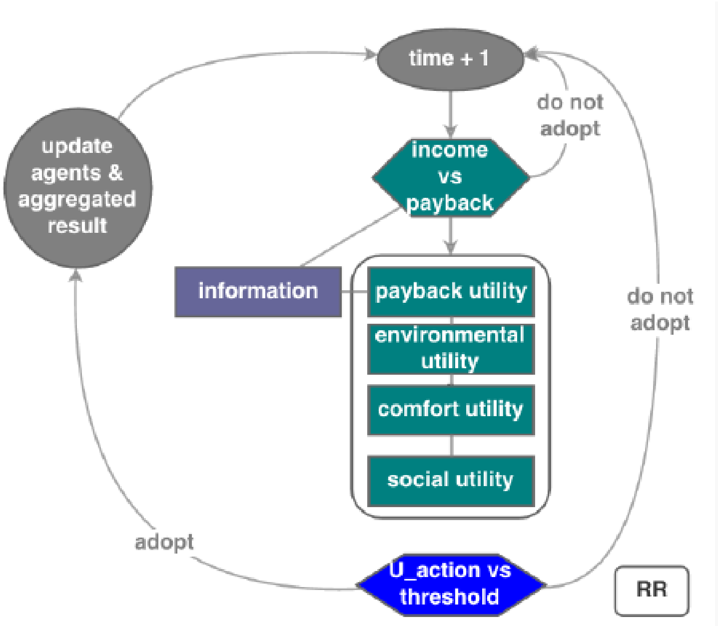


Figure 11: The simulation process for the operationalisation of the Theory of Planned behaviour by Robinson & Rai (2015)

Design Concepts:
In each time step, a variety of data is collected, starting with the share of agents purchasing solar PVs. Furthermore, the number of PV installations categorised by income class and the average environmental significance of households with PV installations are categorised by income class. Some data, however, requires calculations. For instance, the total electricity production aggregated across all installed PV systems can be calculated through the following formula whereby $e_{max} = PV$ peak power, $t_{sun} = sunshine\ hours, p = performance\ ratio\ of\ the\ PV$ and $a = roof\ size$.

$$E_{tot}(t_{pv}) = e_{max} * t_{sun} * p * a \qquad (3)$$

The overall financial savings for households that result from solar PV installations are calculated based on the total revenue generated by the PV system (r_{tot}). This comprises the summation of the power generated by the PV system (E_{tot}) multiplied by the electricity costs (C_e) over the PV system's lifetime (t_{PV}), subtracted by the total cost per square meter (C_{PV}) and roof size (a):

$$S_{mon} = r_{tot}(t_{PV}) - (C_{PV} * a)$$
$$r_{tot} = \sum_{t=1}^{t_{PV}} E_{tot}(t) * C_e \qquad (4)$$

Finally, the total reduction in CO_2 emissions attributed to solar PV installations can be calculated by multiplying the total electricity production from PVs $(E_{tot}(t_{PV}))$ by the average CO_2 savings per kWh (S_{CO2}):

$$S_{CO2} = E_{tot}(t_{PV}) * S_{CO2} \qquad (5)$$

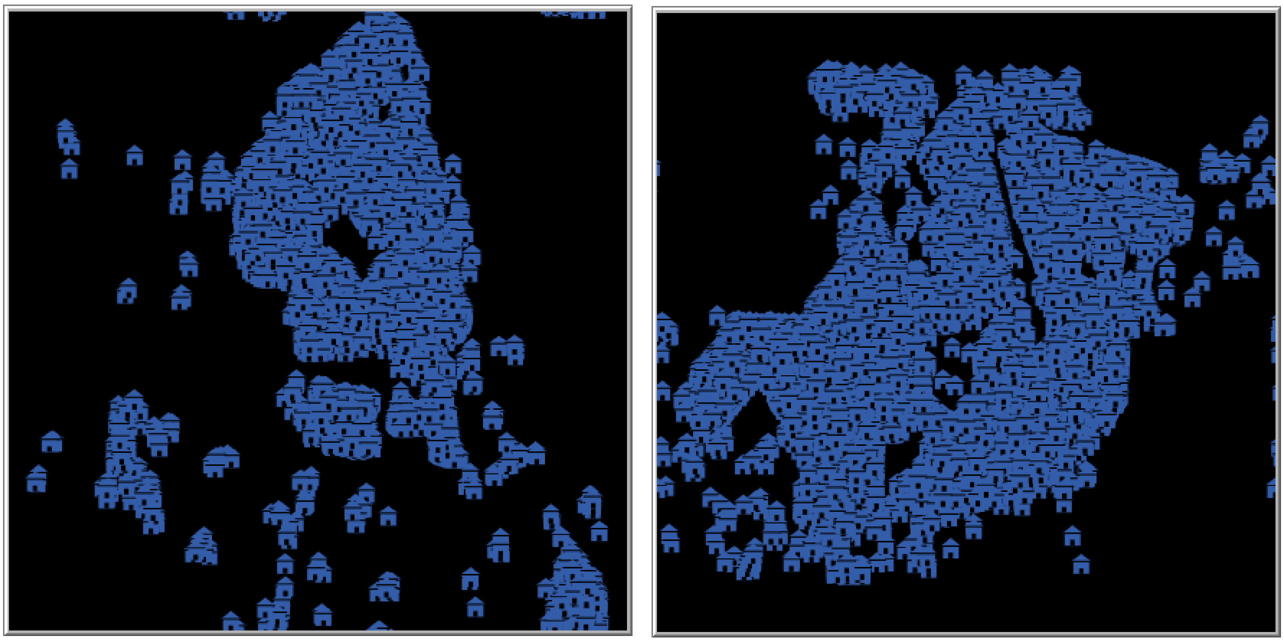


Figure 12: Presentation of the interface of the model by Muelder & Filatova (2018). In this case, the municipalities of Vaals (left) and Oegstgeest (right) have been taken.

5.1.2. Model Assumptions

Several assumptions have been made for the model. While it is impossible to name all assumptions, the most important are the following:

- The attribute levels of environmental state variables that cannot be changed during the run are based on their values in 2012. Some of these values changed drastically throughout the past decade, however. Therefore, this is an assumption that presents a limitation to the study.

- The model focuses mainly on the behaviour of households and does not include policy measurements to a large degree.

5.1.3. Fixed Parameters for Inverse Modelling

Each of the municipalities has certain demographical characteristics that are fixed. These characteristics are therefore also set as fixed parameters in the inverse modelling process: they cannot be changed. Table 6 below presents an overview of these parameters and their values for each municipality. Firstly, the number of households is set as a fixed parameter. The second of these fixed parameters is the initial PV share, which relates to the share of households owning solar panels at t=0. Note that these are not a percentage but rather a fraction of the total population (amount of households). Furthermore, the income distribution is set as a fixed parameter. These distributions per municipality are presented in Appendix C. Finally, the roof sizes and geolocations are also set as fixed parameters. These are based on data from TUDelft3d (2024). These fixed parameters per municipality for the model can be found in Appendix D.

5.1.4. Variable Interface Parameters for Inverse Modelling

In the table below, an overview is given of the parameters that will be used in the inverse modelling process and the range and steps by which they will be changed during this process. In total, 5 variable parameters were chosen. The reason behind this number is a combination of computational abilities (more parameters slowing down the model due to a larger search area) and processibility in the data analysis. Note that when a distribution is (0, 1, 0.1) this means that the parameter will have a discrete distribution where 0 is the minimum, 1 is the maximum and the steps are 0.1.

The economic utility weight reflects the significance that households attribute to financial factors when making decisions about purchasing solar PV systems. Similarly, the environmental utility weight indicates the importance households place on the environmental impact of solar PV. The comfort utility pertains to the significance households assign to the psychological comfort derived from solar PV systems, such as potential concerns regarding aesthetic disruption or other forms of discomfort unrelated to financial aspects. Lastly, social utility refers to the importance households place on familiarity or shared experiences with PV systems within their social circle (Muelder & Filatova, 2018).

Some of the parameters are fixed in Python, such as the RandomSeed and the fact that only the RR model will be used. Since the number of households is based on a shapefile, this parameter is not defined. On top of that, the initial PV share is also fixed since this is based on real data (CBS, 2020).

Table 6: Parameter Distributions for the Inverse Modelling Process in Python

Parameter	Description	Distribution
Discrete Parameters		
Weight_eco	The economic utility	(0, 1, 0.01)
Weight_env	The environmental utility.	(0, 1, 0.01)
Weight_cof	The comfort utility.	(0, 1, 0.01)
Weight_soc	The social utility.	(0, 1, 0.01)

5.1.5. Excluded Parameters in the IM Process

Due to computational limitations (specifically, overflow errors in the Python file), not all of the parameters in the NetLogo model can be considered. Therefore, several parameters have been set to a default value in the NetLogo model, and will therefore not be used in the search space during the inverse modelling process. This list of excluded parameters has been added in Appendix H. This chapter will briefly elaborate on the reasoning behind excluding these parameters in the inverse modelling process, and their default value in the NetLogo model. Note: this does not mean that these parameters are not considered at all. It only means that *one* value is considered, instead of the entire value space for the parameter.

The first exclusion decision is related to the type of TPB that is used. The purpose of the original model was to test different code implementations of the Theory of Planned Behaviour in this specific model on solar PV adoption, namely MF (based on Muelder & Filatova (2018)), SE (based on Schwarz & Ernst (2009)) and RR (based on Robinson et al. (2007)). However, since this is not the objective of this study, only one type of TPB operationalisation will be sufficient. For this study, the RR model will be considered due to it being the earliest implementation of the TPB and it being the one resulting in the least computational limits. For this reason, all of the parameters related to the MF and the SE model are non-applicable. The RR_sensitivity_barrier is also not included since this parameter is not included in the original code of the model by default: it needs to be uncommented for it to be included. Considering the computational limits, it was therefore decided not to uncomment this section in the code.

The Uncertainty switch relates to whether or not uncertainty is included in financial aspects, such as the payback time of solar panels. Since this will always be the case in reality, this switch is excluded from the inverse modelling process and set to "True" by default. The Financial_Information and the Probability_Financial_information parameters are excluded because of internal modelling mistakes, which means that the Financial_Information switch does not impact the model results. The Information_Threshold switch is therefore set to "False" by default. The exclusion of the Random_links parameter also relates to an internal modelling mistake: including this parameter sometimes results in an error in the model, therefore it was chosen not to include it in the inverse modelling process and set its default value to 0. The weight distribution is set to homogenous by default because the heterogenous option makes use of survey results of the weights of the economic, environmental, social and comfort utility, whereas this research aims to consider these parameters in the model as well. The Information_distribution that Is used for the calculation of the uncertainty level for different information distributions is by default set to empirical. The reason for this is that this is the only option within this parameter that makes use of scientific literature for the reasoning behind this distribution. (Rai & McAndrews (2012) ; Rai et al. (2016)).

The Ajadv and Ajsoc parameters were not included in the inverse modelling process due to their limited influence and were therefore set to a value of 0.02 by default. The true/false parameters Visibility, Sparking_events, Info_Costs, Info_Costs_revenue, Infor_Costs_Income and Information_Threshold were not included due to their relatively low impact in the model. These parameters have been set to True by default. Furthermore, the discrete parameters Interest rate and PV_SDE_premium are not included because the values of these parameters do not differ per municipality: they are fixed on a national level. The interest rate is set to 0.06 by default and the PV_SDE_premium is set to 0.10 by default. The Influence_cost_time and the information_threshold_value are both not included due to their limited impact and are set to a value of 0.25 by default. The close_links parameter has been set to 2 by default.

Finally, the RR_sia parameter was not included. Even though this parameter would have been interesting to research since it dives deeper into the adoption barrier for solar PV adoption, the influence of the value of this parameter was too large for this research. This meant that even a minor change in this parameter would result in completely different values for all other parameters. For this reason, it was decided that the RR_sia parameter is too unstable to use in the data analysis. By default, this value has been set to 0.15.

5.2 Model Conceptualisation Within the Inverse Modelling Process

This subchapter introduces the conceptual framework of the model and presents a UML diagram illustrating its general structure. The aim is to clearly understand the proposed model's theoretical basis and organisational layout. It is important to note that even though this chapter is titled "Model Conceptualisation," it encompasses two independent models: the NetLogo model previously presented and a Python model. However, these two models cannot be seen individually within the IM process (they are both required) and operate as a unity. Therefore, any reference to "the model" in this work pertains to the entire modelling process, which includes both the NetLogo and the Python model. Specific references to either the NetLogo or Python model will be explicitly stated.

To understand the conceptual model, one must start with the real world. Real-world data provides information on demographics such as income distribution, initial pv-share and roof sizes mentioned earlier in this chapter. This data forms the 'fixed parameters' or parameters that are fixed for a certain municipality and that will not change during the inverse modelling process. On top of that, certain variable parameters also form inputs for the NetLogo model. These variable parameters have also been presented earlier in this chapter. Together, these fixed and variable parameters form the inputs for the NetLogo model for a certain municipality. This NetLogo model is then able to generate a certain output. This output is then used in the Python model. With the output of the NetLogo model, the Python model can generate predictions on solar PV adoption for that municipality using machine learning algorithms. These predictions are then compared with the *real* data on solar PV adoption, as presented in Chapter 4. These predictions are the number of solar PV installations installed for that municipality during the years 2012 – 2022 (so 1 value per year, totalling 11 predicted values). The predictions made by the model and the real data (which is also in the shape of installed solar PV installations for that municipality during the years 2012 - 2022) are then compared, and a certain error value can be calculated. This error value gives an indication of how good (or bad) the prediction is. This means that the goal is to *minimise* this error value. By changing the variable parameters, different predictions can be made which results in different error values and corresponding values for the variable parameters for each iteration. This is the inverse modelling process. After a certain number of iterations, a number of error values and associated parameters are generated, and the lowest error value represents the *best* prediction. This means that the model will have *one* minimum error value and associated variable parameters for each run. Adjusting the model parameters to match the simulation outputs as closely as possible to real-world data is also called parameter calibration.

This process can be executed for each of the municipalities, resulting in a total of *6* different outputs for each of the models. These outputs can then be analysed. The goal hereby is to research if, given the fixed values, the same or different values for the variable parameters lead to good (or bad) predictions between the municipalities. In doing so, one can explore the potential interactions between fixed and variable parameters that might explain regional differences in solar PV adoption and analyse how fixed parameters contribute to regional variations in solar PV adoption rates. In Chapter 6, the data analysis process will be further elaborated.

Figure 13 presents the conceptual model of the inverse modelling process visually. Note that this conceptual model visualises the process for *3* municipalities instead of 6. This conceptual model presents the relationships between the NetLogo models, the Python models, and the real world. This relationship and feedback to the real world are essential in the inverse modelling process since this is the part that improves the understanding of the system.

This conceptual model also clearly defines the difference between parameter calibration and inverse modelling, two terms that can easily be confused. Figure 13 shows that parameter calibration focuses on adjusting model parameters to optimise the model's fit to the observed, real-world data, aiming for accurate predictions. Conversely, inverse modelling seeks to understand the driving forces and dynamics shaping solar PV adoption rather than solely optimising model parameters for prediction accuracy. This means that inverse modelling, in a sense, goes deeper. Whereas optimising the model parameters is the end goal for parameter calibration, for inverse modelling, parameter calibration can be seen as a means to get to the end goal of actually explaining a particular phenomenon.

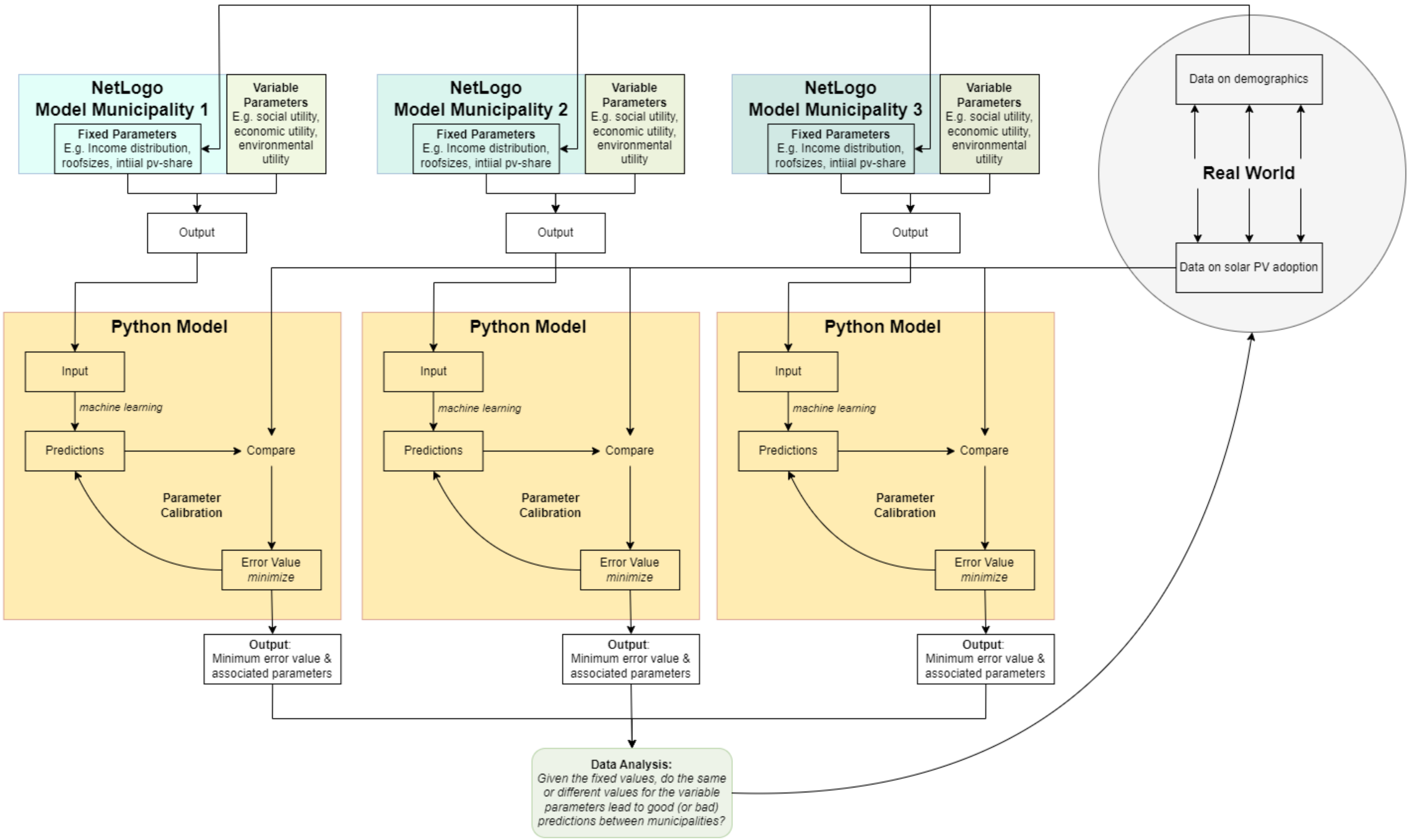


Figure 13: Conceptualization of the inverse modelling process in this research

5.3 Python Model Formalisation

After the model conceptualisation, the model formalisation process takes place. In this subchapter, a comprehensive overview of the structure of the model and the pseudocode is presented to clarify and further interpret the conceptual model. This is a crucial phase in the model development because it marks the transition from abstract conceptualisation to concrete implementation.

5.3.1. UML Diagram of the Basic Model Structure

The first step in the model formalisation process is to zoom in on the Python model and create a basic UML diagram of the model structure. From this conceptual model, one can zoom in on the Python model and create a basic UML diagram of the basic model structure. The model can be run from the main file. This main file requires, however, certain other files. Firstly, the main file requires training data. This data is generated in the right format in the *generate_data* file. On top of that it makes use of a custom cross-validator. This cross-validator is used to evaluate the performance of the model by partitioning the dataset into subsets, training the model and then evaluating it on the remaining subsets. This helps in preventing overfitting. The *plot_fit* file is not directly used for running the main file, but rather for visualisation purposes. However, since the *plot_fit* file cannot exist without using the main file, the relationship between these two files is a composition. The *main* file can create one or more plot fits, depending on the inputs of the *main* file. Furthermore, the *main* file makes use of an optimise file. In this file, a custom model is defined that is based on the NetLogo model (defined in the *model* file), along with utility functions for hyperparameter tuning and evaluation. The model uses parameters to configure its behaviour and makes predictions using the *error* file. In the *erro*r file, an error function is defined and used to determine how good or bad the model's prediction is. Therefore, the predictions and training data are compared in this file. This process is repeated over and over again (see the model fit process) until the optimal parameters of the model that minimise the error metric are found. The basic UML diagram of this model structure can be found below. The main file creates one optimal error value. This one optimal error value comes, however, forth from the *optimise* file. Using the *optimise* file, a variety of error values and their belonging parameter values are generated. The one optimal error value is the smallest of these error values.

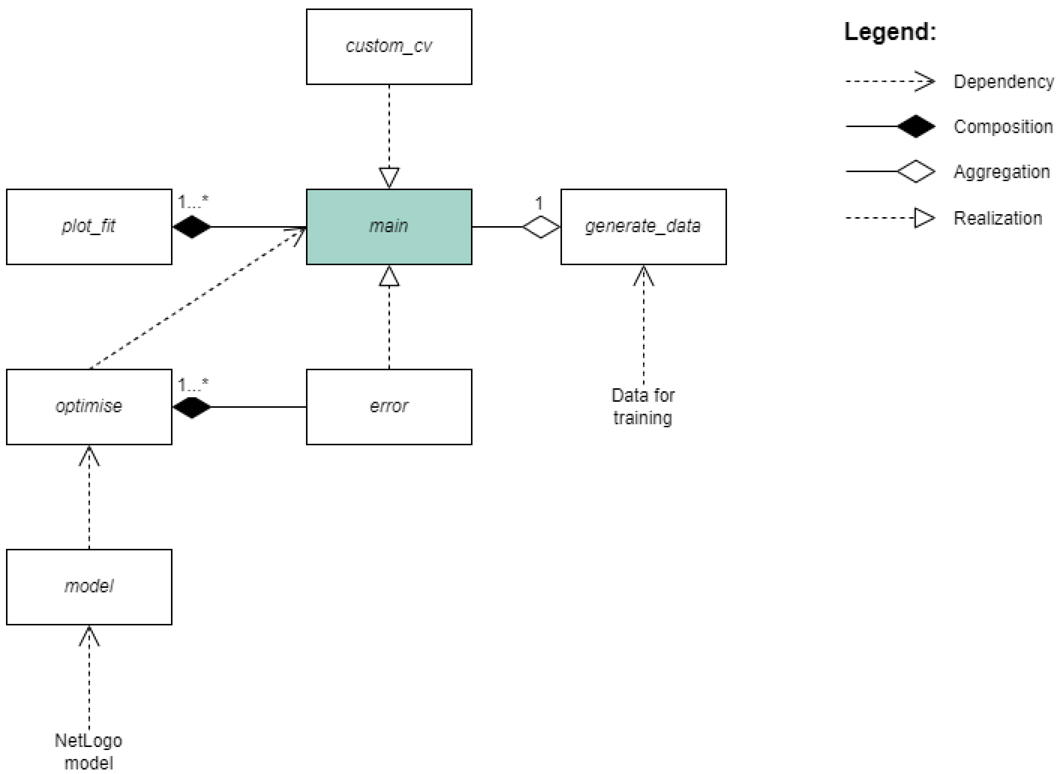


Figure 14: UML diagram of the basic model structure of the Python model used for parameter calibration within the inverse modelling process

5.3.2. Detailed Model Structure

This subchapter aims to provide a more detailed overview of the structure of the Python model presented in Figure 15 and elaborate on it so that it provides a basis for understanding the pseudocode. This detailed model structure begins with the NetLogo model. This NetLogo model is loaded into the *model* file, meaning that it serves as a bridge between Python and the NetLogo model. Pynetlogo is used as an interface to access NetLogo from Python (Jaxa-Rozen & Kwakkel, 2018). In this file, a NetLogo instance is started and the model is loaded. Then, a function called *run_model_with_parameters* is defined. In this function, the input parameters are set and the model is set up. The model runs 11 ticks, whereby one tick equals one year (each year between 2012 and 2022). Then the model is run and the number of solar PV installations is counted per tick. Finally, a data frame is created with the data per tick. The function returns this data frame.
The second file is the *generate data* file. Its goal is to take the training data and convert it to a CSV file.

The third file is the *plot_fit* file. This file defines the function of showing a plot containing the predictions, the results, and potentially the error and parameters.
In the fourth file, called the *error* file, the error function is defined. The CSV containing the training data that was generated in the *generate_data* file is read and the data from the municipality is extracted. Then, a custom distance metric function is defined. This function computes the distance between the model data and the target data, considering the standard deviation. It first calculates the median and mean of the model data and checks if the absolute difference between the model's median or mean and the target data is within 10% of the specified standard deviation. If it is, the function returns 0; otherwise, it returns the absolute difference. Thereafter, the error function is defined. This function calculates the Dynamic Time Warping (DTW) distance between the predictions and the actual data on solar PV adoption for the municipality using the custom distance metric defined before. Dynamic Time Warping is a technique that can be used for measuring the similarity between two sequences (Salvador & Chan, 2007; Berndt & Clifford, 1994). The advantage of using DTW is that it is relatively robust to temporal variations which makes it useful for time series data such as the data in this research. Finally, the distance is returned.

The fifth file is the *optimise* file. In this file, a custom model class is defined that serves as a wrapper for the machine learning algorithm. This class initialises various parameters that define the behaviour of the model. They are set with default values and stored in a dictionary, which makes them easily accessible throughout the class. Then, the fit method is defined, which is later used to fit the machine learning algorithm in the *main* file. Thereafter, the prediction method is defined which generates predictions. For scoring the model, the *score* method computes the model's performance score using the *error* function defined earlier. This score makes sure that the smallest error value is selected, so the parameter configuration with the smallest margin between the predictions and the real data.

The sixth file is the *custom_cv* file. This file defines a custom cross-validation strategy. The purpose of the base-cross validator is to define the common interface and behaviour for all cross-validation strategies. It assesses the performance of the predictive model by splitting the dataset into multiple subsets, training the model, and then evaluating it on the remaining subsets.

The seventh file is the *main* file. This is the file that aims to optimise the model using the machine learning algorithm. The file firstly imports all necessary libraries, just like in the other files. Thereafter, it sets the number of runs for the optimisation process. Thereafter, it defines a function to load the data from the CSV file and a function that defines the custom discrete function for generating search spaces. A function that saves the error values and the parameter values to a CSV file is also created. Then, the training data is loaded and the required data is extracted. This means that first the years are extracted from the data, then the training data and the data for only the municipality. Due to the required data formats of the algorithm, the training data ought to be transposed. Then, the model is initialised, and the parameter distributions are defined. This creates the search space for the ML algorithm. Then, the randomised search for hyperparameter tuning is performed for each run and the fit method, defined in the optimise file, is applied. The best model outcome is then used to make predictions on the training data. Finally, the error value between the predictions and the actual values is computed. These error values (one for each run) and corresponding parameters are saved to a CSV file. The pseudocode for each of these files can be found in Appendix I.

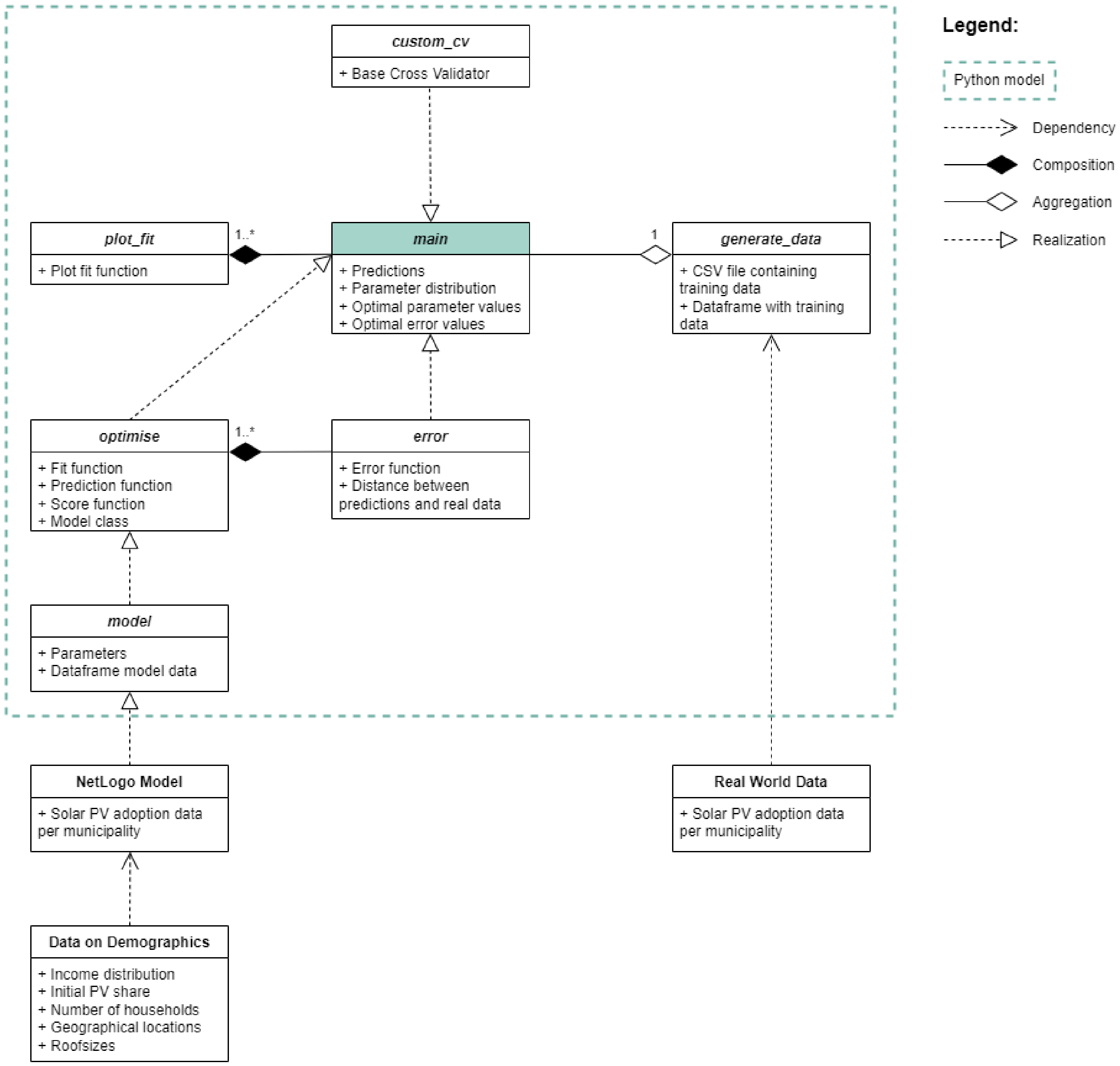


Figure 15: More detailed UML diagram of the Python model structure

5.3.3. Machine Learning Application

The machine learning algorithm serves as a means for the inverse modelling process in this model. The file utilises the Random Search algorithm. This subchapter elaborates on the algorithm's application.

To begin with, the (NetLogo) model is imported. This model represents the algorithm that will be optimised throughout the process. The optimisation process revolves around tuning hyperparameters. These hyperparameters are defined in a parameter grid and consist of all the variable parameters presented earlier in this chapter. The core of the optimisation takes place within a loop where the algorithm is employed. The algorithm is initialised with certain parameters, such as the model, the parameter grid, the cross-validation strategy (including the number of splits), the verbosity level, the number of parallel jobs and the number of parameter settings to sample. Hereby, the verbose controls the verbosity of the output during the process, and the number of jobs specifies the number of parallel jobs to run during model training (this will be set to -1 to ensure that all available CPU cores will be utilised for the process) and the number of iterations refers to the number of times the optimisation process is repeated. Then, the Randomised Search or Bayesian Search instance is fitted to the training data. During this process, it randomly samples combinations of hyperparameters from the parameter grid and evaluates the model's performance using cross-validation. Once fitted, the best model found during the search is obtained. This is done by using the *best_estimator_* attribute of the Randomised Search or Bayesian Search instance. The best model is then used to make predictions on the training data. The error value and the corresponding best parameters found are stored in lists during each iteration of the optimisation loop.

5.4 Python Model Validation and Verification

In the methodology in Chapter 2, the process of verifying and validating the model was shortly highlighted to ensure its accuracy and intended functionality. Within this process, three key stages can be delineated for ABMs, namely: model verification, model validation and sensitivity analysis (Cooley & Solano, 2011). For this research, however, the sensitivity analysis stage will not be considered. This is due to the fact that within a sensitivity analysis, one tries to evaluate the robustness and reliability of the model by assessing the sensitivity of its outcomes to changes in input variables. However, since this research focuses on modifying the input parameters, a sensitivity analysis will not be executed.

5.4.1. Model Verification

Verification and validation are often conflated. In the context of this research, however, refers to assessing whether the model's logic is sound. This means ensuring that both the programming logic and the formal logic of the model are correct (Cooley & Solano, 2011). Hereby, it is essential to ensure that the model behaves as anticipated.

Various verification techniques were employed throughout the programming process based on the techniques presented by Whitner & Balci (1989). Firstly, *informal analysis* was continuously employed during the coding process. This approach hinges on the modeller's understanding of the code's logic and the influence of modelling decisions on the model's outcomes. Additionally, *desk checking* was regularly conducted. This involves reviewing the code to confirm that the sequence of commands and the underlying logic adhered to the structure outlined in the conceptual model.

Syntax analysis is an inherent feature of both the NetLogo and Python environments. Therefore, this type of analysis occurs automatically each time the code is compiled. Furthermore, the entire code is divided into distinct functions, each serving a specific purpose. This organisation, called structural analysis, ensures a structured codebase, facilitating error identification and enabling the testing of individual functions.

The primary focus during the development of the model was dynamic analysis, emphasising the examination of the model's behaviour during execution. As previously mentioned, the code was structured into separate functions, each with its distinct role. Upon completion of a new function or new part in the code, it underwent independent testing to explore its capabilities, considering its reliance on inputs from other functions. It was also during this phase that it was discovered that the original NetLogo model contained several functions that were either not functioning or functioning wrongly (e.g. the *initial PV share)*, as mentioned before in Chapter 5.1. These functions were then changed so that they could work. In the event of errors, debugging and execution tracing methodologies were employed to pinpoint and resolve issues.

Finally, *pylint* was used. Pylint is a code analysis tool for Python that can help improve code quality by identifying programming errors, adherence to coding standards and potential issues in Python code. It analyses the code and provides feedback on various aspects such as coding standards, syntax errors, unused variables, etc. In doing so, it contributes to model verification. Pylint was used during the programming process. However, not all of the feedback from pylint was always implemented. For instance, pylint often gave recommendations for making the code more compact. This feedback was not always implemented since it was thought to reduce the readability and reproducibility of the code. Suggestions often made the code more 'black box' which is not desired. Therefore, all code files except the *main* and *optimise* files achieved a pylint score of 9.00/10.00 or higher.

5.4.2. Model Validation

Validation, on the other hand, concerns the degree to which the model accurately mirrors the system it represents (Darvishi & Ahmadi, 2014). However, determining if a model is valid is not a simple binary answer (A. Crooks et al., 2008). Validity assessment often entails comparing model outputs with empirical data. The validity of the model was checked in two ways. Firstly, *face validity* (Cooley & Solano (2011) ; A. T. Crooks & Heppenstall (2012)) was performed by consulting with advisor Lukas Schubotz. This led to a continuous review of the model, improving its validity.

On top of that, the model's validity was checked by creating a separate Python script to check its validity. This Python script performs a validation curve analysis to determine the optimal number of splits for cross-validation in the modelling task. The script begins by loading the municipality's real solar PV adoption data. Then, cross-validation is performed using a custom cross-validation method, namely the one described in the *custom_cv* file. The dataset is split into training and testing sets based on the specified number of splits. The model is trained on the training data and then evaluated on the testing data using the error function specified in the *error* file. The cross-validation process is repeated for different numbers of splits. Due to computational limitations, the maximum number of splits was set to 5.

The mean cross-validated error and its standard deviation are computed for each number of splits. These values are then plotted on a validation curve, with the number of splits on the x-axis and the cross-validated error on the y-axis. The script identifies the optimal number of splits as the value that minimises the mean cross-validated error. It prints out this optimal value along with the corresponding error and standard deviation. In Appendix I.8. the pseudo-code for this file is presented. In Figure 16, the validation curve is presented. Note that the validation curve is run with random input parameter values, therefore the error value itself is not representative of the error value in the future, only the relationship between the error values for each number of splits is relevant. From this graph, it can be concluded that 2 is the optimal amount of splits. Therefore, the model will be run with 2 splits in the future. Not that the model validation is based on the Random Search algorithm. To facilitate the best comparison between the two algorithms, the number of splits for the Bayesian Search algorithm is also configured to be 2. This is a limitation of the study since the Bayesian Search algorithm might require more splits to find the optimal search area. The number of iterations was set to 15, resulting in 30 fits executed for 25 runs. This configuration was carefully chosen to balance the reliability and trustworthiness of the results with computational constraints. Increasing the number of iterations and splits typically enhances the robustness and accuracy of the results by providing a more comprehensive exploration of the parameter space and a better estimate of the model's performance. However, this comes at the cost of increased computational demand and processing time. This careful consideration ensures that the results are credible and obtained within a reasonable timeframe, making the research rigorous and feasible.

5.5 Chapter Summary

This chapter thoroughly explores the model conceptualisation, formalisation, validation and verification process. Firstly, the chapter presents an exploration of the NetLogo model's conceptualisation. This part presents a foundation for understanding the design and functionality of the NetLogo model, which serves as a fundamental component for bridging real-world data with the simulation, as presented in Chapter 5.2. The chapter highlights the significance of incorporating both fixed parameters, such as demographic data, and variable inputs, which are essential for driving the model's predictive capabilities. The following section, Chapter 5.2, provides a deeper understanding of the conceptual framework. A UML model is used to visually present the model's basic format and data flows. Chapter 5.3 describes the transition of the model conceptualisation to implementation. It outlines the various model files, including NetLogo integration, data generation, error computation and optimisation. Finally, the validation and verification of the model are elaborated on in Chapter 5.4. Verification techniques such as informal analysis, desk checking, syntax analysis and the use of Pylint are employed to confirm the correctness of the model's logic and implementation. The validation, on the other hand, focuses on aligning the model with empirical data through face validity and empirical data analysis, which are concluded in a validation curve analysis to optimise cross-validation parameters.

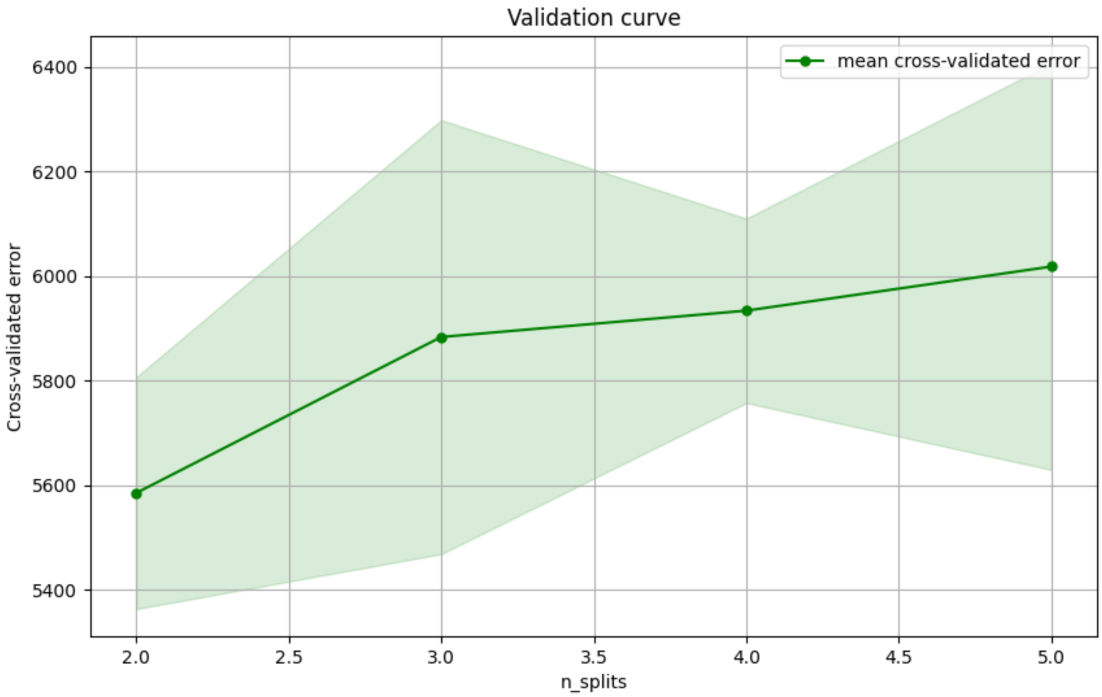


Figure 16: Validation curve of the model, presenting the optimal amount of splits

6 Results

This chapter focuses on the results that come forth from running the model. In total, 12 different versions of the model have been produced, namely 2 for each municipality (one for the Random Search algorithm, one for the Bayesian Search algorithm). In this chapter, the results from each of these models will be analysed qualitatively. For organisational purposes, this chapter is structured as follows: Firstly, a municipality-specific analysis will be conducted for each of the municipalities in Chapters 6.1 – 6.6. This means that the results will be analysed individually for each municipality. The municipality-specific analysis starts with research on general demographic information on the municipalities, a Google Trends analysis and a short analysis of news articles concerning solar PVs in each of the municipalities. Thereafter, the solar PV adoption curve for each municipality will be discussed briefly. After this general discussion about each municipality, the results from the inverse modelling process will be presented and discussed. This part will also discuss the fitness and scatter of the model results, and each subchapter ends with a short discussion of future projections based on the IM results. Chapter 6.7 focuses on an inter-municipal analysis. In this part, the outcomes for each of the municipalities in Chapters 6.1-6.6 will be compared and checked for patterns and similarities between them. Finally, Chapter 6.8 focuses on an experimental change in the setup, whereby the number of splits and iterations is increased for one municipality, namely Laren.

It is essential to note that throughout this chapter, terms such as ´social factor´, 'environmental factor', ´economic weight´ or even simply `comfort´ are used (to give examples). It is important to keep in mind that these terms, though they might not initially seem related to the model, are always based on the results of the model and relate to its parameters (so *weight_eco, weight_env, weight_soc* and *weight_cof*). This means that they should not be read as self-contained statements but rather in the context of the model. So instead of *´social factor´,* one could read *´the social parameter weight_soc of the model shows XYZ´.* This has been done to improve the readability of the text.

6.1 The Municipality of Bloemendaal

The municipality of Bloemendaal, situated in the province of Noord-Holland, has held the title of the wealthiest municipality in the Netherlands in recent years, boasting a median income of €51.400 and an average income of even €90.300 (CBS, 2023c). The municipality has 23.782 inhabitants and 9945 households as of 2024 (AlleCijfers, 2024a). The municipality has a population density of 602 inhabitants per km^2 (CBS, 2023b). This data is also specified in Appendix D, along with the demographic data of the other municipalities. In Figures 9 and 10, the solar PV adoption curve of the municipality is presented, also in relation to the adoption curves of the other municipalities. The adoption curve in this graph shows a steady increase in the number of installations over the years. This indicates a growing interest and adoption of solar energy within the municipality. On top of that, the rate of adoption seems to accelerate over time. The increase in the number of installations appears to be more pronounced in recent years (e.g. from 2017 to 2022). While Google Trends can provide insights into residents' interest in solar PVs, its regional monitoring capabilities are limited to the province level, such as Noord-Holland. However, it does offer the ability to gauge general interest within individual municipalities over time. Scores are assigned on a scale from 0 to 100, with 100 indicating the highest relative popularity of a search term compared to total searches within the area.

For example, a score of 50 represents proportionally half the popularity. Notably, the municipality of Drenthe exhibits the highest relative popularity nationwide, while Noord-Holland scores 41, the lowest among all regions. Within Noord-Holland, Bloemendaal ranks fifth lowest with a score of 32. Thus, it can be inferred that residents of Bloemendaal infrequently search for information on solar panels, potentially impacting the adoption of solar PV systems within the municipality.

Additionally, between 2012 and 2022, there were 34 news articles on Google featuring the Dutch term for solar panels combined with the municipality name of Bloemendaal. The majority of these articles revolved around the development of a solar carport equipped with 5,000 solar panels within the municipality. However, it is worth noting that this carport cannot be classified as residential solar PV. Furthermore, other news articles were not directly related to residential solar PV in Bloemendaal.

From these observations, it can be concluded that despite its affluent status, Bloemendaal demonstrates a relatively low level of interest in adopting solar PV technology. This becomes evident from not only the solar PV adoption curve but also from the limited searches for information on solar panels, suggesting a slower uptake of residential solar PV installations compared to other regions. Furthermore, news coverage within Bloemendaal regarding solar VP appears limited.

6.1.1. Inverse Modelling Results for Bloemendaal

As mentioned, the inverse modelling process uses a Random Search and a Bayesian Search algorithm. The results of the inverse modelling process of each of these algorithms are described and compared below. Furthermore, a boxplot with the inverse modelling results is created. The boxplot for the municipality of Bloemendaal can be found in Figure 17. Note that the boxplot colours are chosen based on the colours in the solar PV adoption curves in Figures 9 and 10. In those graphs, Bloemendaal has, for instance, the colour blue, Dantumadiel orange, Laren green, etc. For uniformity purposes, these colours have also been kept for the boxplots. Only the colour of the Bayesian Search algorithm has been selected as a lighter shade of the original colour to distinguish between the algorithms.

Running the Random Search model took roughly 4 hours and 24 minutes, while the Bayesian Search algorithm's runtime was 4 hours and 35 minutes. This indicates that the Bayesian Search model took roughly 25 additional seconds per run. This additional time is due to Bayesian Search building and updating the model to guide its search. Each iteration involves fitting this model and optimising to select the next set of parameters, which adds computational load compared to the Random Search model that does not include these steps.

Boxplot elaboration:
In the figure below, the boxplots of the results of the inverse modelling process are presented. This graph visually compares the distributions of the error values generated by the inverse modelling process for the municipality of Bloemendaal for both the Random Search algorithm and the Bayesian Search algorithm. A boxplot is used to present these results since this type of plot is useful for summarising the key characteristics of each of the datasets. Note that since the error value measures the distance between the predicted and the real (actual) solar PV adoption values in the municipality, a lower error value presents a higher fitness.

In the boxplot, each box represents the interquartile range (IQR), which contains the middle 50% of the data. This range is defined by the first quartile (Q1) and the third quartile (Q3). Inside each box, a horizontal line indicates the median or the second quartile (Q2). For the Random Search algorithm, this median line is coloured blue, while for the Bayesian Search algorithm, it is coloured light blue. These median lines provide a clear visual indication of the central tendency of each dataset. The whiskers extend from the boxes to the smallest and largest value within 1.5 times the IQR from the lower and upper quartiles. These whiskers give a sense of the overall spread of the data, excluding any outliers. Outliers, data points that fall significantly outside the typical range, are plotted as individual dots beyond the whiskers. This helps identify extreme values that might indicate unusual cases or potential errors in the data collection process.

The means of the datasets are marked by triangles within the boxes. For the Random Search dataset, the mean marker is coloured blue, matching the colour of its median, while the marker for the Bayesian Search dataset is again light blue. These mean markers provide an additional measure of central tendency, offering insight into the average performance of each method. Furthermore, black dots are scattered to the left of each box, representing the individual data points of each dataset. These dots provide a more granular view of the data distribution. By showing each data point, the scatter plot complements the summary statistics provided by the box and whiskers.

Note that the y-axis of the plot is set to a range from 0 to 30,000. This ensures that all data points from the municipalities, including the most extreme outliers, are clearly visible and comparable.

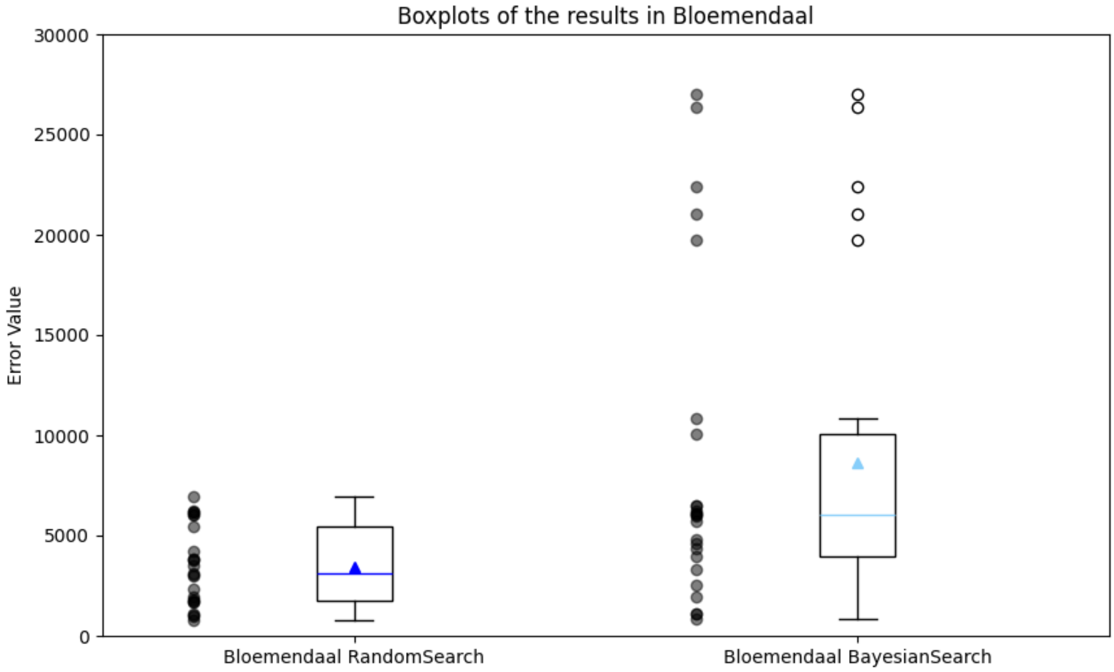


Figure 17: Boxplot of the results of the inverse modelling process for the municipality of Bloemendaal

Random Search Results:
The inverse modelling process is first executed using the Random Search algorithm. The results from this process can be found in Appendix J. The run with the best fit and, therefore, greatest accuracy is a run with an error value of 788. This run is visualised in the boxplot as the lowest black dot for the Random Search boxplot. Furthermore, this run is presented in Figure 18, visualising both the real solar PV adoption values of the municipality and the adoption value predictions made by the Python model. The median (the blue line in the boxplot) has a value of 3090, and the mean (the blue triangle) has a value of 3458. One can see in the boxplot that the mean and the median are located relatively close to one another. When the mean and median are close to one another, it indicates that the data is symmetrically distributed and not heavily skewed, indicating robustness against outliers and consistent estimates of typical values. This can also be noticed in the plot: no outliers are identified. The whiskers of the boxplot are relatively short, which suggests a smaller spread with data points clustered closer to the median. The standard deviation is equal to 1933, as also presented in Table 7.

Table 7: Mean, median and standard deviation of the Random Search results for Bloemendaal

Mean	3458
Median	3090
Standard deviation	1933

In Table 8, a summary of the data analysis for the Random Search algorithm for Bloemendaal is presented. When analysing these results, one can notice that the top 5 best performing runs have a significantly higher value for the *comfort* factor than the other runs, namely 0.52 compared to 0.30 on average. The *environmental* factor, on the other hand, is actually a lot lower. The *social* factor is always very high. This is an interesting observation because it indicates that the social factor ought to be high to be selected as the best performer *within* a run. However, it also indicates that *between* the runs, the *comfort* factor actually seems to be a determining factor when it comes to how good a prediction is.

When looking at these results, one can see a discrepancy between the results for the best-scoring run and the average results. Whereas the best-scoring results suggest a high importance for the *social* and *comfort* factors, the average results suggest a high importance for the *social* and *environmental* factors. This presents a dilemma for data interpretation, which centres on prioritising top-performing results or considering overall consistency in an inverse modelling process. Should the focus be on top-performers' specific strengths or on the average values for broader insights? This dilemma of handling the differences between performances during runs adds complexity. On the one hand, focusing on the top-performing results, one may uncover specific parameter combinations that lead to optimal outcomes. This approach allows for identifying promising scenarios that achieve low error values, potentially providing valuable insights into effective strategies or conditions for solar PV adoption. However, this approach may overlook the broader variability and dynamics present in the dataset. It may not fully capture the range of possible outcomes or the robustness of the model across different parameter settings. On the other hand, focusing on the average values of all results provides a more comprehensive perspective on the model's overall performance across different parameter settings. It considers the variability and distribution of outcomes, which offers insights into the general trends and patterns. However, while this approach offers a broader view, it may also obscure the specific parameter combinations that lead to optimal outcomes.

To solve this dilemma, one ought to take in mind the general objective of the inverse modelling process, which is to uncover explanations for complex phenomena or residential solar PV adoption dynamics in this case. Considering this goal of inverse modelling, a more nuanced approach is necessary. Therefore, the more robust results coming forth from the second approach seem more appropriate for this research, and the average results seem more appropriate for data analysis. For that reason, the average results will be used mainly for the data analysis for the rest of the municipalities. To still make comparisons between good and bad runs, the top 5 best and worst runs will be considered.

Table 8: Summarised data analysis for the Random Search algorithm for Bloemendaal

	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight sum**
Top 5 best runs					
	0.26	0.36	0.52	0.68	1.82
Top 5 worst runs					
	0.14	0.52	0.18	0.68	1.52
Average values over all runs					
	0.33	0.49	0.30	0.68	1.8

Bayesian Search Results:

For the Bayesian Search algorithm, the best-scoring run has an error value of 863, surpassing that of the Random Search algorithm. Both the actual and the predicted solar PV adoption values for this best run are presented in Figure 18. Seeing the data interpretation discussion above, this would indicate an inferior performance to the Random Search algorithm if one would place emphasis on optimal outcomes. However, as also described above, this does not align with the objectives of inverse modelling in this research, and therefore, this higher error value does not yet indicate a better or worse performance of the model. The median (the light blue line in the boxplot) has a value of 6057, and the mean (the light blue triangle) has a value of 8609 (see Table 9). When the mean and median in a boxplot are far apart, as is the case here, it indicates that the data is skewed. The mean being greater than the median indicates that some large values in the dataset are pulling the mean upwards, suggesting the presence of outliers. These outliers can also be observed in the boxplot, where a total of 5 outliers can be seen. This suggests a reduced robustness compared to the Random Search results. Furthermore, one can notice the length of the whiskers in the boxplot, with the bottom whisker being significantly longer than the top whisker. Longer whiskers typically indicate a larger spread or variability in the data, suggesting that the range of values extends further from the median. When the whisker is shorter at the top than at the bottom, it therefore suggests asymmetry in the distribution of data. Specifically, it indicates that the dataset is right-skewed, meaning that there are more data points clustered towards the lower end of the distribution, resulting in a shorter upper whisker.

Table 9: Mean, median and standard deviation of the Bayesian Search results for Bloemendaal

Mean	8609
Median	6057
Standard deviation	7827

When analysing the results of the Bayesian Search algorithm further, it can be noticed that the *social* factor again scores the highest when looking at the average values of all results, with a value of 0.65. When one would solely look at the top-performing runs (which is only one run in this case), this result is exaggerated with a value for the *social* factor of 0.9. For the top 5 best-performing runs, this social weight is 0.76, which is still higher than average. When considering the five worst-performing runs, on the other hand, the *social* factor is actually very low, with a value of 0.38. The *comfort* factor, on the other hand, seems to be the least important when considering the average values of all runs. This indicates that *social* utility is important in the decision-making process for residential solar PV in this municipality.

Table 10: Summarised data analysis for the Bayesian Search algorithm for Bloemendaal

	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight sum**
Top 5 best runs					
	0.56	0.54	0.34	0.76	2.20
Top 5 worst runs					
	0.60	0.54	0.30	0.38	1.82
Average values over all runs					
	0.45	0.47	0.29	0.65	1.86

6.1.2. Added Value of Inverse Modelling for Bloemendaal

Considering both the standard deviation and the minimum error value, the Random Search model performs better than the Bayesian Search model. Chapter 7 will discuss this observation more thoroughly.

The inverse modelling process has provided valuable insights into the dynamics of solar PV adoption in the municipality of Bloemendaal. The traditional analyses, such as the Google Trends analysis, provide surface-level insights. However, the inverse modelling process offered additional explanatory power by uncovering some underlying mechanisms driving solar PV adoption in Bloemendaal. For instance, the observation that social factors significantly influence adoption decisions according to this model is the added value provided by the inverse modelling process. It highlights the importance of social networks and community dynamics in shaping residents' attitudes towards solar PV technology. This observation also resonates with the limited search data, indicating a preference of inhabitants for social utility over financial incentives (which are more often searched online). On top of that, despite the high income level of residents in Bloemendaal, the inverse modelling results suggest that economic factors do not play a predominant role in driving solar PV adoption. This implies that while residents may have the financial means to invest in solar PV systems, other factors, such as social dynamics and comfort, may influence their adoption decisions more.

When considering the actual adoption values for Bloemendaal, a pronounced increase in the number of solar PV installations over the years is evident. This trend suggests a likely continuation of rapid growth in the near future. However, predictions from the best runs of the two algorithms employed reveal some interesting differences.

Future Projections:
The Random Search algorithm's best predictive run initially aligns closely with the actual adoption curve but begins to diverge slightly after 2019, although the growth rate remains steep. In contrast, the Bayesian Search algorithm predicts a more linear trajectory rather than an exponential one, leading to a higher error value in its predictions (see Figure 18).

Considering these differences between the algorithms, future projections (so after 2022) could indicate that according to the Random Search algorithm, the number of solar PV installations in Bloemendaal will continue to grow stronger and more exponential compared to the predictions made by the Bayesian Search model.

The summarised data analysis for each algorithm, presented in Tables 8 and 10, highlights the underlying reasons for these differences. The Random Search algorithm attributes more weight to the *comfort* factor in its best runs, while the Bayesian Search algorithm places greater emphasis on *economic, environmental* and *social* factors.

Given that the Random Search algorithm predicts a sharper increase in solar PV adoption, it suggests that future policies to encourage residential solar PV adoption should focus more on enhancing the ease and convenience of purchasing and owning solar PV systems. By prioritising these comfort-related factors, policies can better align with the drivers that lead to steeper adoption growth, as identified by the Random Search model.

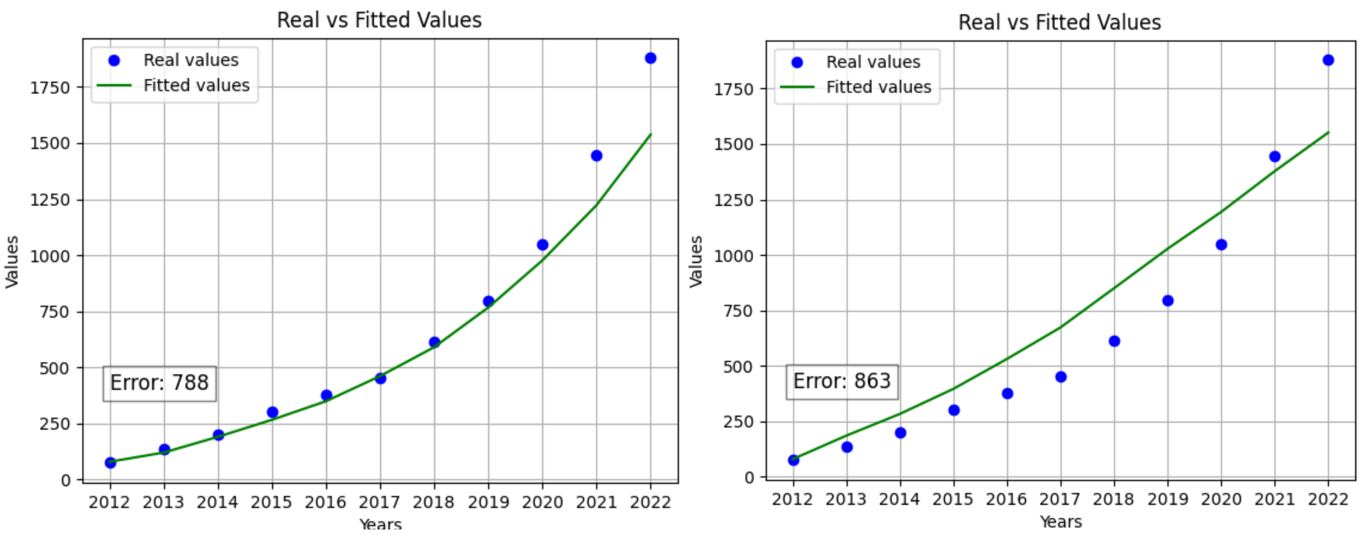


Figure 18: Plot of the real and the predicted solar PV adoption values for Bloemendaal using the Random (left) and Bayesian (right) Search algorithm

6.2 The Municipality of Dantumadiel

The municipality of Dantumadiel in the province of Friesland emerges as a compelling case for the case study. In terms of demographics, Dantumadiel accommodates a population of 19.135 inhabitants distributed across 8.016 households (AlleCijfers, 2024b). The solar PV adoption curve of this municipality can be found in Figures 9 and 10 in Chapter 4. Firstly, the curve depicts a pattern of sustained growth in solar PV installations over time, reaching 4.126 installations by 2022. It is also the municipality with the most solar PV installations out of all municipalities in the case study selection. The first acceleration in adoption rates can especially be observed from 2017 onwards, whereafter a second, even steeper, acceleration can be observed from 2019 onwards. While the exact reason for this timing is unknown, a confluence of factors such as policy incentives (e.g. SDE+ introduction in 2013 (RVO, 2013)), technological advances and increasing public awareness. Noteworthy is the socio-economic backdrop against which this transition unfolds. The median income in the municipality is €34.650, marginally below the national median income of €39.100 (CBS, 2023a). The municipality has a relatively low population density, namely 227 inhabitants per km2 (CBS, 2023b). The demographic information on the municipality of Dantumadiel is also presented in the table in Appendix D. The search interest of inhabitants towards solar energy in the province of Friesland, as presented in Figure 19, is notable, with a score of 98 on Google Trends. Remarkable is the observation that the search interest increased significantly after hitting a low in the fall of 2016, before the start of the acceleration in adoption rates from 2017 onwards. This steep curve ends in 2018, before the start of the second acceleration in adoption rates in 2019.

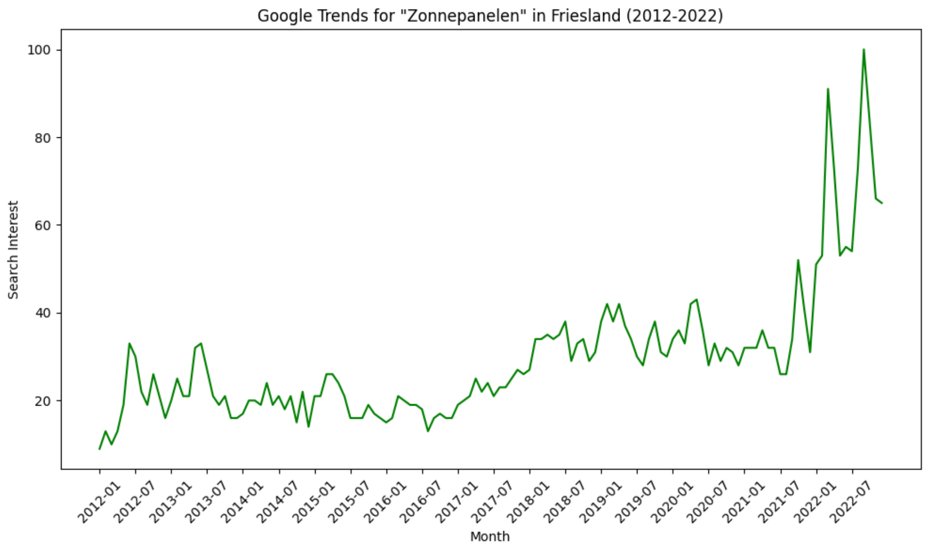


Figure 19: Google Trends scores for the province of Friesland

When analysing news articles about solar PV in Dantumadiel, it is noticeable that the municipality is named as one of the municipalities in the country with the most solar panels three times. These articles, however, were published in 2022 or even later and, therefore, cannot be related to the adoption curve for explanatory purposes.

6.2.1. Inverse Modelling Results for Dantumadiel

This section provides the results of the inverse modelling process for the municipality of Dantumadiel for both the Random Search and the Bayesian Search algorithms. The boxplot of these results is presented in the figure below. Figures 31 and 32 in Chapter 6.7 show the boxplots of the municipality of Dantumadiel in combination with the boxplots of the other municipalities. Running the Dantumadiel Random Search model took 3 hours and 44 minutes. For the Bayesian Search model, running the model took 3 hours and 55 minutes.

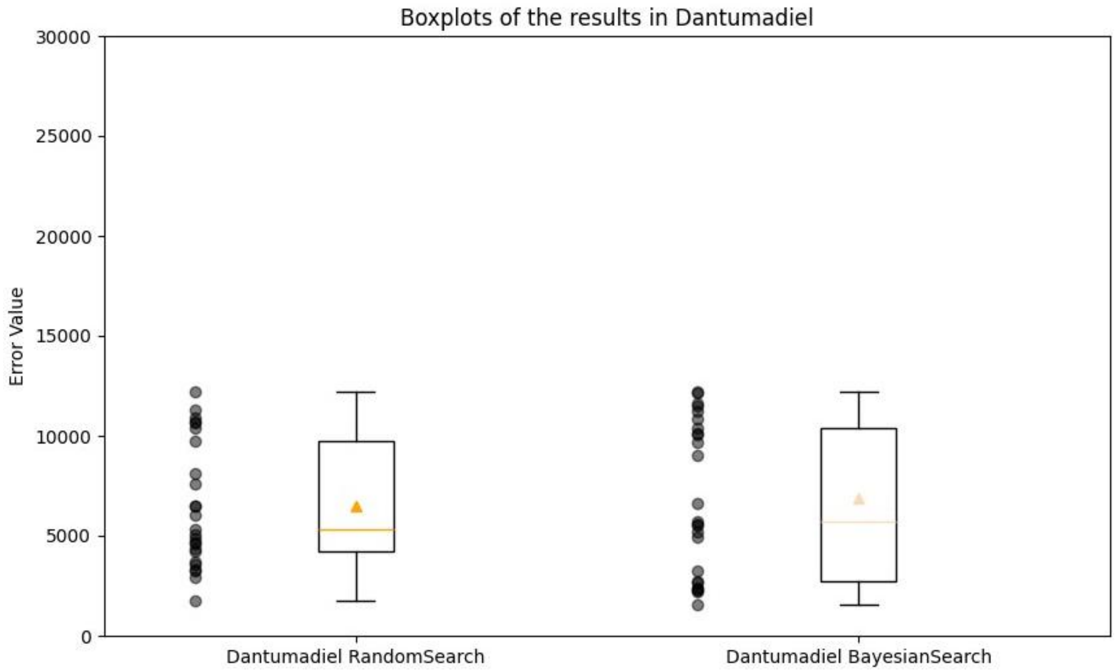


Figure 20: Boxplot of the results of the inverse modelling process for the municipality of Dantumadiel

Random Search Results:
On the left side of the figure above, the boxplot of the results of the inverse modelling process for the Random Search model is visualised. The orange line represents the median, which is equal to 5327 in this case. The mean, as represented by the orange triangle, is equal to 6497. This means that the mean of the dataset is noticeably higher than the median. This hints at a slight right-skew in the distribution, where a few higher values pull the mean upwards. These values, although not outliers, can be observed in the boxplot. It can be seen that between an error value of roughly 8000 and 9800, no individual data points are observed. The data points above 9800 are pulling the mean upwards. Despite this skew, the whiskers of the boxplot are about equally long, stretching from the edges of the box to the minimum and maximum values (1740 and 12236, respectively). This balance in the whiskers' length suggests that the data's spread is relatively even on both sides, even though the right skew is present. The standard deviation is equal to 3071, as can be seen in the table below.

Table 11: Mean, median and standard deviation of the Random Search results for Dantumadiel

Mean	6497
Median	5327
Standard deviation	3071

When analysing the average values of the parameters in the runs in Table 12, the high value of 0.75 for the *social* factor is noticeable, indicating that households place high importance on familiarity with solar PV systems within their social circle. The *comfort* and *economic* parameters seem to be the least important. However, when analysing the top 5 best scoring runs, one can notice that the values for the *social* parameter are relatively similar in all of the runs. The *economic* parameter, on the other hand, shows a much larger value for the top 5 best scoring runs than for the top 5 worst scoring runs or even the average values. Considering the fact that Dantumadiel is a municipality with high solar PV adoption rates, this indicates that the *economic* factor plays a significant role in households' decision to purchase solar PV systems. Because the *social* factor is high for each of the runs, it indicates that this factor is mostly important for deciding the best fit *within* a run.

Table 12: Summarised data analysis for the Random Search algorithm for Dantumadiel

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.52	0.56	0.40	0.76	2.24
Top 5 worst runs					
	0.24	0.36	0.28	0.84	1.72
Average values over all runs					
	0.36	0.50	0.34	0.75	1.94

Bayesian Search Results:

For the Bayesian Search model results, one can first notice that the box in the boxplot is bigger. This indicates a more extensive interquartile range (IQR), which signifies greater variability or dispersion among the central 50% of the data points. Furthermore, one can notice the light orange line in the box, indicating the median with a value of 5694. The mean value, on the other hand, is equal to 6872. The difference between the mean and the median is still present, indicating a slight right skew in the data distribution (with a few higher values pulling the mean upwards). This right skew is, however, not as strong as for the Random Search model. The whiskers in the boxplot are relatively short, shorter than for the Random Search model. This indicates that, despite the inconsistency in the middle range, the values outside the IQR do not deviate much from the quartiles. Another interesting observation is the fact that three strong areas can be observed when looking at the individual data points. Firstly, there is an area between error values of roughly 1500 and 3200, then a small area with error values between roughly 4900 and 6600. Finally, there is a large area with error values between roughly 9000 and 12.000. The exact reason behind these areas is unknown. One can speculate, however, that it could be related to the algorithm's inherent ability to search in certain areas to look for the optimal solution (see Chapter 4.2). The standard deviation of this model is equal to 3754, which is higher than that of the Random Search model. The minimum error value of the model is 1534, which is, on the other hand, lower than that of the Random Search model. This best run is also visually presented in Figure 21, along with the municipality's real solar PV adoption values and the adoption values predicted by the Python model.

Table 13: Mean, median and standard deviation of the Bayesian Search results for Dantumadiel

Mean	6872
Median	5694
Standard deviation	3754

When looking at the top 5 best- and worst-scoring runs, as presented in Table 14, the most noticeable thing is the difference between the sum of the weights of each parameter value. For the top 5 best scoring runs, this sum equals 2.38, whereas this sum equals 1.62 for the five worst-scoring runs. Furthermore, one can notice that especially the *economic* parameter and the *comfort* parameter have a lot higher values compared to the five worst scoring runs, indicating that for this algorithm and this municipality, these two factors play an important role in the decision process for solar PV adoption between runs.

When considering the average parameter values of all runs, one can notice how the *social* and *environmental* parameters especially have high values. This indicates that even though these two parameters are always considered the most important during the runs for this model, the *economic* and the *comfort* parameters make the most difference in determining a good run since these values differentiate significantly between the best- and worst-scoring runs. In contrast, the *social* and *environmental* factors always remain high.

Table 14: Summarised data analysis for the Bayesian Search algorithm for Dantumadiel

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.5	0.6	0.54	0.74	2.38
Top 5 worst runs					
	0.2	0.48	0.14	0.8	1.62
Average values over all runs					
	0.33	0.55	0.34	0.76	1.97

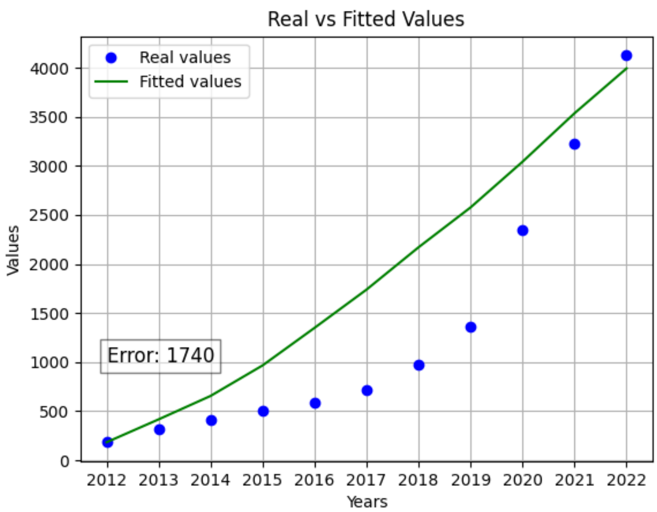
6.2.2. Added Value of Inverse Modelling for Dantumadiel

The added value of inverse modelling for this municipality can be found in the fact that through inverse modelling, it can be identified that, according to this research, the model does a better job at predicting solar PV adoption when the weight of each of the parameters is higher. This indicates that, based on this model, solar PV adoption in this municipality is higher when inhabitants attribute high importance to each factor. On top of that, insight can be gained that even though the *social* and *environmental* factors might seemingly play the most important role for households when making adoption decisions, the *economic* and *comfort* parameters make the strongest difference in the decision whether or not to adopt solar PV. This insight can be gained when comparing the average values with the values for the best and worst scoring runs.

At the same time, it should be said that these results have a very large room for improvement. When considering the standard deviation of either one of the models for this municipality, it can be said that the scatter of the results is still extremely high, presenting a lack of uniformity and high variability in the results. The scatter in this municipality is very high, especially when comparing the standard deviations of Dantumadiel with those of the previous municipality, Bloemendaal. The main goal of inverse modelling is to gain a deeper understanding of solar PV adoption dynamics, so robust results are extremely important. Therefore, as described above, the added value of inverse modelling for Dantumadiel is subject to validation and should be approached with caution.

Future Projections:
From the best runs of each algorithm presented in Figure 21 it can be noticed that the Random Search algorithm follows a more linear curve over the last years, whereas the Bayesian Search algorithm follows a slightly more exponential curve. This would indicate that if this curve were to be continued after 2022, the Bayesian Search model would result in higher adoption rates. As can be concluded from the error values, the Bayesian Search model also generates a more accurate prediction, closer resembling the exponential solar PV adoption curve of Dantumadiel.

Considering Tables 12 and 14, it can be said that for the top 5 best runs for each of the algorithms, most parameter weights only differ by a few hundredths. The only factor that differs more strongly is again the *comfort* factor. This factor is higher for the Bayesian Search algorithm for Dantumadiel than for the Random Search algorithm.
Given that the Bayesian Search algorithm predicts a sharper increase in solar PV adoption and more closely resembles the real values, it suggests that future policies to encourage residential solar PV adoption should focus more on enhancing the comfort behind purchasing and owning solar PV systems.



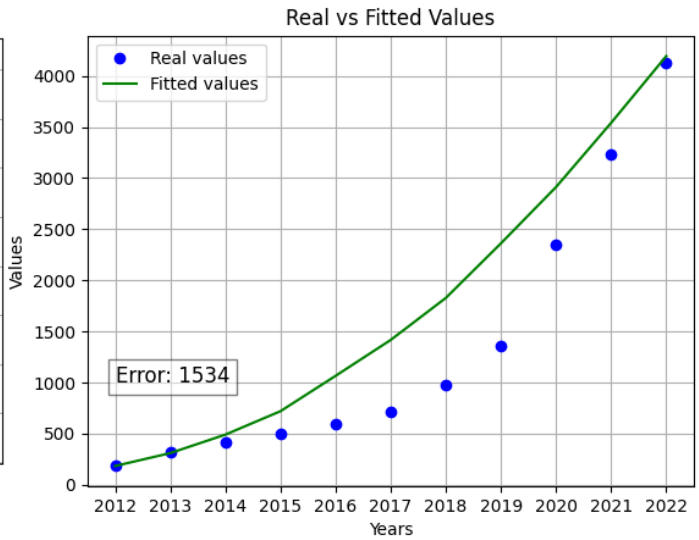


Figure 21: Plot of the real and the predicted solar PV adoption values for Dantumadiel using the Random (left) and Bayesian (right) Search Algorithm

6.3 The Municipality of Laren

Located in the province of Noord-Holland, the municipality of Laren distinguishes itself with a high median income of €48.250 (CBS, 2023a). The population density of the municipality is 944 inhabitants per km^2 and is inhabited by 11.195 residents, distributed across 5.255 households (AlleCijfers (2024c) ; CBS (2021)). In the context of solar PV adoption, the municipality demonstrates an adoption curve that can be described as "slow and steady". Laren started with a relatively low number of solar PV installations compared to other municipalities in 2012, with only 10 installations registered. Over the years, there has been a steady increase in solar VP adoption in the municipality, with the only stronger acceleration taking place in 2021. Laren has fewer residential solar PV installations than the other municipalities in the case study, even when population size is considered.

The acceleration in solar PV adoption in Laren can also be noticed in the search interest on Google Trends, as presented in Figure 22. Particularly noteworthy is the steep increase in the number of searches by residents of the province, notably starting in late 2020. Being situated in the province of Noord-Holland, Laren registers a relatively modest search score of 35.
Notably, the news coverage concerning solar PV in Laren is relatively sparse, with only six articles related to solar panels published between 2012 and 2022. These articles, mostly only remotely related to solar PV adoption, offer limited insight. Notably, one article reported a significant house fire attributed to solar panels in 2021. However, this incident does not seem to have had an apparent impact on solar PV adoption within the municipality.

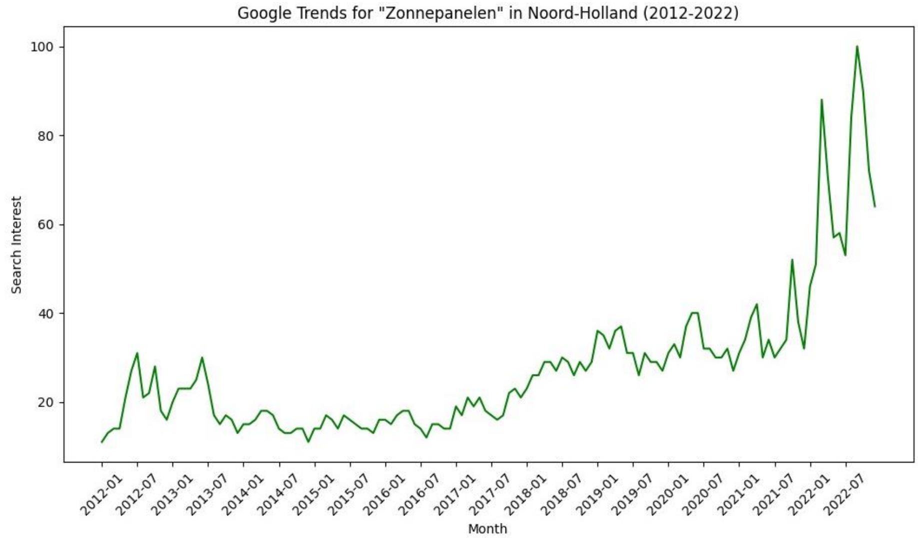


Figure 22: Google Trends score for the province of Noord-Holland

6.3.1. Inverse Modelling Results for Laren

In the boxplot below, the inverse modelling results for the municipality of Laren are presented. The Random Search model for the municipality of Laren took in total 2 hours and 26 minutes to run. For the Bayesian Search model, this time was equal to 2 hours and 36 minutes. One can note that it took a lot less time to run the models for Laren compared to the previous two municipalities. The reason for this lies in the amount of data that had to be reloaded for each run. Since Laren only has far fewer households, a lot less data ought to be loaded for every run. This process takes a lot of time, even in the original NetLogo model already.

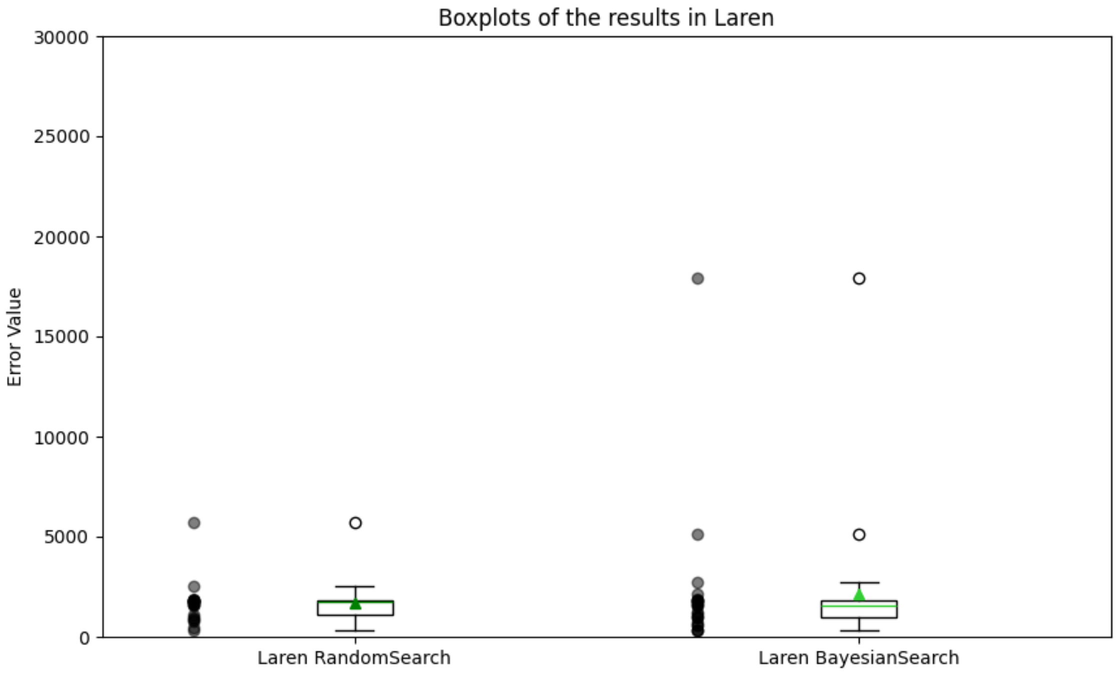


Figure 23: Boxplot of the results of the inverse modelling process for the municipality of Laren

Random Search Results:

The inverse modelling process for the Random Search algorithm provides fairly good results for Laren. As can be noticed in the graph above, the boxplot for the model is very narrow. This typically indicates that the data values in the results have low variability or dispersion. If the IQR is small, the majority of the data points are clustered closely together. When analysing the median of the boxplot, one can notice that the line for the median is almost equal to the top end of the box and has a value of 1738. This indicates that the distribution of the data is negatively skewed because most of the data points are concentrated towards the lower end of the distribution. The fact that the median is close to the top end of the box indicates that there are very few high values and that the bulk of the results lie below the median. One can also notice that the mean is less than the median, with a value of 1668. This indicates again that the distribution is negatively skewed. Furthermore, the boxplot only shows one outlier, with a value of 5735. Compared to the previous two municipalities, even the outlier of this result has a relatively low value. The whiskers of the boxplot are short as well, which indicates that the range of the data is relatively small. The best result has an error value of 314, closely predicting the number of solar PV installations in the municipality in 2022 at 545, only 39 installations below the actual count. This best run is also visually presented in Figure 24. The combination of a small boxplot (a low scatter) and low error values indicates a robust prediction performance.

Table 15: Mean, median and standard deviation of the Random Search results for Laren

Mean	1668
Median	1738
Standard deviation	974

When looking at the average values of the results in Table 16, one can notice that the *social* factor is again the most important, with a value of 0.8. The *environmental* factor is the second most important in the model, and the *economic* and *comfort* factors seem least important to inhabitants when considering the model results. Interestingly enough, when analysing the parameter values of the top 5 best performing runs, one can notice that these deviate less from the average values than what was the case in the previous two municipalities. The only factor that deviates more is the *economic* parameter. This indicates that, indeed, the *social* and *environmental* parameters are important for households when deciding to adopt solar PV. However, this also indicates that the *economic* parameter is more important for households in the decision-making process for runs that are better at predicting the actual solar PV adoption curve.

Table 16: Summarised data analysis for the Random Search algorithm for Laren

	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight sum**
Top 5 best runs					
	0.42	0.48	0.4	0.78	2.08
Top 5 worst runs					
	0.16	0.58	0.32	0.78	1.84
Average values over all runs					
	0.36	0.42	0.31	0.8	1.89

Bayesian Search Results:

For the Bayesian Search model, it can be noticed that the box in the boxplot is slightly larger, indicating that there is slightly more variability within the middle 50% of the data compared to the box of the Random Search model. The median is 1578, which is slightly lower than that of the Random Search model. The mean, on the other hand, has a value of 2136, which is higher than that of the Random Search model. When looking closer at the boxplot, one can notice that the median is lower than the mean. This indicates a positively skewed distribution, suggesting that there are some high values pulling the mean upwards. The standard deviation is equal to 3359, which is relatively high compared to that of the Random Search model. This standard deviation is highly influenced by the two outliers that can be seen in the boxplot. The first outlier, with a value of 5111, is still considerable. However, the second outlier has an error value of 17903, which is very high. This extreme outlier pulls up not only the standard deviation but also the mean of the error values. Without these outliers, this model would actually perform better than the Random Search model.

When looking at the best-performing run, with an error value of 316 (presented in Figure 24), it can be noticed that the final prediction of this run is less accurate (with 357 installations in 2022, compared to 545 installations for the Random Search algorithm). However, as can be observed in Figure 24, the predicted solar PV adoption curve by the Bayesian Search model aligns better with the general trajectory of the actual solar PV adoption curve of the municipality. Therefore, the final error value of this model is actually very similar to that of the Random Search model.

Table 17: Mean, median and standard deviation of the Bayesian Search results for Laren

Mean	2136
Median	1578
Standard deviation	3359

When looking at the parameter values of the results in Table 18, one can first notice that the *social* factor is again the most important when looking at the average parameter values. The *environmental* factor seems to be the least important, with a score of only 0.24. When comparing these average parameter results, however, to the top 5 best-performing results, one can notice a large difference in the parameter value of the *comfort* parameter. Whereas the *comfort* parameter value is equal to 0.37 for the average of all runs, it is equal to 0.68 for the top 5 best-performing runs. This difference is a lot stronger in this municipality than in the previous two municipalities. The importance of the *social* parameter also increased for the top-performing runs compared to the average runs.

Table 18: Summarised data analysis for the Bayesian Search algorithm for Laren

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.38	0.24	0.68	0.82	2.12
Top 5 worst runs					
	0.54	0.38	0.18	0.58	1.68
Average values over all runs					
	0.39	0.24	0.37	0.69	1.68

6.3.2. Added value of Inverse Modelling for Laren

When assessing the value added by the inverse modelling process for Laren, valuable insights and potential areas for questions are encountered. This will be elaborated on below.

Firstly, the traditional and inverse modelling analyses offer valuable insights into Laren's solar PV adoption dynamics. Firstly, analysing news articles about solar PVs in Laren revealed relatively sparse coverage. Despite the low media attention, the inverse modelling outcomes emphasise the importance of social considerations in adoption decisions, hinting at a disconnect between media discourse and the actual influencing factors. In the context of this specific ABM, this could also explain the low adoption of solar PV in the municipality. Moreover, the Google Trends analysis depicts residents' increasing interest in solar PVs, which has been particularly notable from late 2020 onwards. However, the inverse modelling results in the context of the ABM by (Muelder & Filatova, 2018) also unveil the prominence of social factors. This suggests that residents prioritise familiarity or shared experiences with PV systems within their social circles, a dimension not fully captured by the Google Trends analysis.

However, the insights derived from the inverse modelling process for this municipality may also warrant scrutiny. Discrepancies exist between the results of the Random Search algorithm and the Bayesian Search algorithm. While both algorithms recognise the importance of the social factor, the Random Search algorithm assigns significantly greater importance to the economic factor than the Bayesian Search algorithm. Consequently, the reliability of these findings may be subject to questioning.

It is important to note that so far, the municipality of Laren shows the most promising boxplot for the IM process due to its relatively low scatter and low minimum error values. Chapter 7 will discuss which of these two factors is more important for IM.

Future Projections:
When looking at the best run of each of the algorithms in Figure 24, it can be noted that the Random Search algorithm follows a more linear curve, whereas the Bayesian Search model follows a more exponential curve. Due to their different strengths, the algorithms have relatively similar minimum error values for these runs. This means that when considering future projections, it is likely that in the near future, the Random Search algorithm will yield higher solar PV adoption rates. However, due to the exponential nature of the result of the Bayesian Search algorithm, this run might actually yield higher solar PV adoption rates after a few years.

Considering the summarised data analyses for both algorithms in Table 16 and Table 18, it can be said that the top 5 best runs for the Random Search algorithm place a higher emphasis on the *economic* and *environmental* factors than the Bayesian Search algorithm. The top 5 best runs for the Bayesian search, on the other hand, place a higher emphasis on the *comfort* factor and the *social* factor. This would indicate that future policies aimed at increasing residential solar PV should focus on the economic and environmental factors in the short term, whereas the focus could shift to comfort and social factors in the long term.

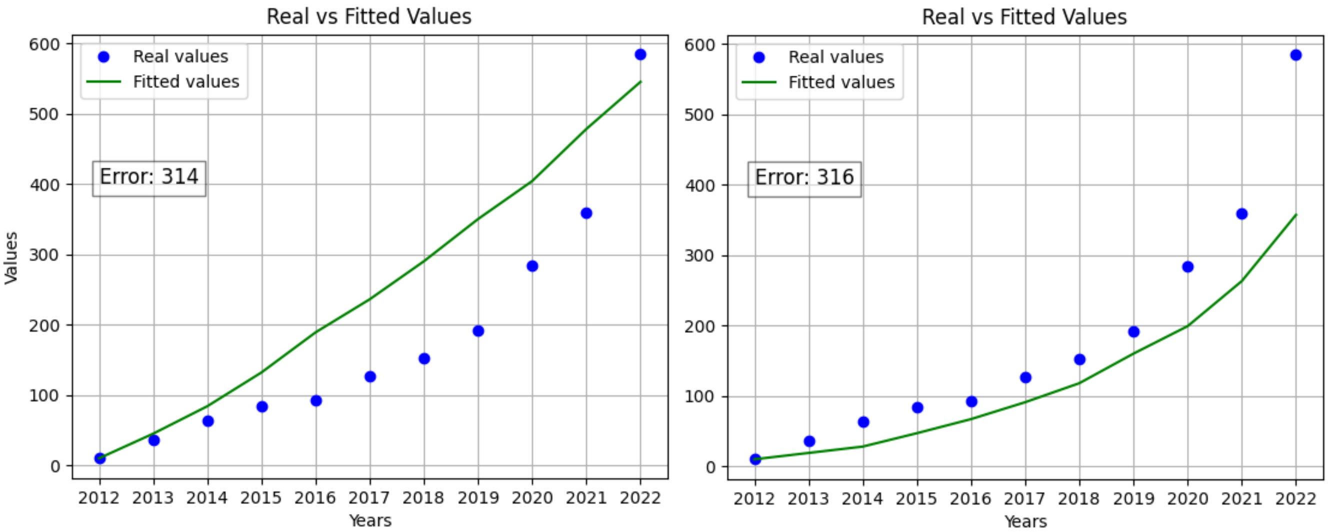


Figure 24: Plot of the real and the predicted solar PV adoption values for Laren using the Random (left) and Bayesian (right) Search Algorithm

6.4 The Municipality of Oegstgeest

The municipality of Oegstgeest, located in the province of Zuid-Holland, is home to 25.939 inhabitants distributed over 11.115 households (AlleCijfers, 2024d). The municipality has a median income of €47.750, which is higher than the national median income in the Netherlands (CBS, 2023a). Its population density is relatively high, namely 3530 residents per km^2 (CBS, 2021).
The solar PV adoption trend in Oegstgeest commenced with a moderate number of installations in 2012. Since then, Oegstgeest has witnessed consistent growth in solar PV adoption, with installations increasing annually. Overall, the municipality’s adoption curve can be characterised as moderate to slightly above average compared to other municipalities. Notably, Oegstgeest experienced more pronounced growth, particularly from 2017 to 2018.

When analysing Google Trends results, it should first be noted that the province of Zuid-Holland has the second-lowest search interest in solar PV, with a score of 53. Within the province, Oegstgeest gets a score of 37, which is relatively low. In terms of news articles, 23 news articles featuring solar panels in Oegstgeest were published on Google between 2012 and 2022. Most of the articles feature the installation of solar panels on various public buildings.

6.4.1. Inverse Modelling Results for Oegstgeest

The inverse modelling results for the municipality of Oegstgeest are presented in the boxplot below. Due to the number of households (and, therefore, the increased amount of data), the models of this municipality took the longest to run. For the Random Search model, the running process took 5 hours and 11 minutes, whereas the Bayesian Search model took 5 hours and 21 minutes. Similarly to the runtimes of the models of the previous municipalities, the Bayesian Search model took around 25 seconds per run longer. Appendix M presents the inverse modelling results for the municipality of Oegstgeest in more detail.

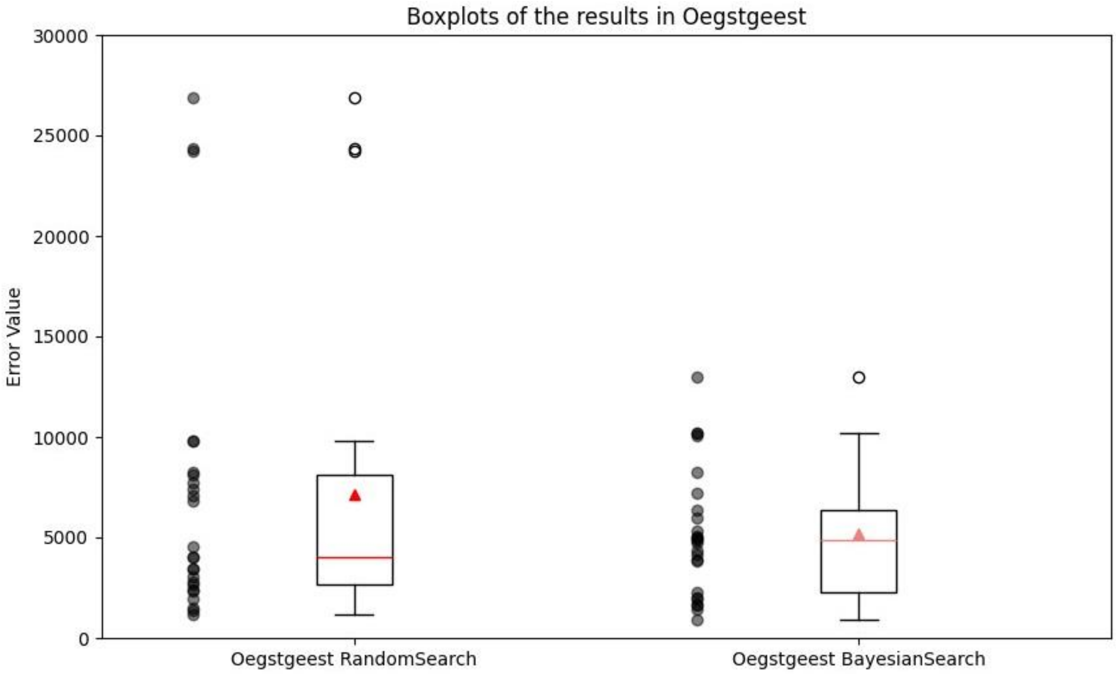


Figure 25: Boxplot of the results of the inverse modelling process for the municipality of Oegstgeest

Random Search Results:

On the left side of the boxplot presented above, one can see the results of the Random Search model for the municipality of Oegstgeest. One can first notice the size of the box, which is relatively large, indicating that there is a wide IQR in the dataset. This suggests that there is a substantial variability in the data and that the central data points are spread out over a wide range of values. Furthermore, the mean value (indicated by the red triangle) is 7160, which is significantly higher than the median value of 4052. This discrepancy indicates a right-skewed (or positively skewed) distribution. A right-skewed distribution occurs because the higher values pull the mean upward more than they affect the median. This implies that the majority of values are relatively low but that there are a few extremely high values that skew the distribution. This can also be noticed in the plot above. Three very strong outliers can be identified (note that two of them have almost the same value, so it looks as if there are only two outliers when there are, in fact, 3), with values of over 24.000. As a result, the standard deviation of the Random Search model is high, with a value of 7152. Without the three outliers, the standard deviation would have had a value of roughly 2700. The smallest error value that is measured for this model is equal to 1139.

Table 19: Mean, median and standard deviation of the Random Search results for Oegstgeest

Mean	7160
Median	4052
Standard deviation	7152

From the average parameter values, it can be noticed that the *social* parameter in the model is of the highest importance to households. Thereafter, the *comfort* factor is the most important, and the *economic* factor seems to be the least important in the model. When looking at the top 5 worst-scoring runs, one can notice that the *environmental* and *social* parameters differ, especially from the average values. On the one hand, the social parameter has a much lower value for the five worst-scoring runs than for the average. The *environmental* factor, on the other hand, scores a lot lower. For the top 5 best-scoring runs, it can be noticed that the *environmental* factor is actually higher than average, indicating that environmental considerations are relatively important in this municipality for good predictions in the model.

Table 20: Summarised data analysis for the Random Search algorithm for Oegstgeest

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.32	0.46	0.38	0.54	1.7
Top 5 worst runs					
	0.4	0.3	0.4	0.48	1.58
Average values over all runs					
	0.33	0.42	0.47	0.66	1.88

Bayesian Search Results:

For the Bayesian Search model, one can first notice in the boxplot that the box is smaller than the box of the Random Search model. This indicates a smaller IQR and, thus, a smaller data variability. The standard deviation is equal to 3109, which is much lower than that of the Random Search model. The main reason for this is the outliers. For the Bayesian Search model, only one outlier can be identified, with a value of 12977. This means that this model does not only have fewer outliers than the Random Search model, but this outlier also is less strong. Therefore, the standard deviation is significantly lower. Furthermore, one can also notice that the mean and the median are much closer together. The mean has a value of 5165, and the median has a value of 4885, as presented in Table 21. This still indicates a slight positive skew, but much less strong than for the Random Search model. This slight right skew can also be identified by looking at the whiskers of the boxplot. Since the top whisker is longer than the bottom, this indicates the existence of a few higher values in comparison to more concentrated data on the lower end. The minimum error value equals 904, which is better than that of the Random Search model. In Figure 26, the best runs of each of the models are plotted. In these graphs, one can nicely see the difference between the best runs. Even though the model's predicted total amount of installed solar PV (3137 installations) in Oegstgeest is exceptionally close to the total amount of installed solar PV in the municipality in reality (3135 installations), this run still gets a higher error value (and therefore worse fitness) than the best run of the Bayesian Search model. This is all related to the fact that this last model follows the total solar PV adoption curve much better, generating a lower error value overall.

Table 21: Mean, median and standard deviation of the Bayesian Search results for Oegstgeest

Mean	5165
Median	4885
Standard deviation	3109

When analysing the average parameter values of all runs, one can notice that, again, the *social* parameter is the most important, followed by the environmental factor (with values of 0.72 and 0.57, respectively). The *economic* factor is, on the other hand, the least important in this model. When considering only the top 5 best runs, the *social* and *environmental* factors are still the most important in the model. Their value is, however, lower than the average of the runs. This indicates that, even though these two factors are still considered the most important in this municipality, their significance becomes less for the better-performing runs.

Table 22: Summarised data analysis for the Bayesian Search algorithm for Oegstgeest

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.40	0.52	0.32	0.56	1.80
Top 5 worst runs					
	0.22	0.4	0.4	0.72	1.74
Average values of all runs					
	0.36	0.57	0.39	0.72	2.04

6.4.2. Added Value of Inverse Modelling for Oegstgeest

When considering the inverse modelling results for the municipality of Oegstgeest, one can say that the Bayesian Search model outperforms the Random Search model since it has both a lower scatter and a lower minimum error value. When considering the average values of both models, it can be said that the *social* parameter is the most important in the model. On the other hand, the economic parameter is considered the least important parameter in each of the models. Because of the inverse modelling process, this finding could be identified. This is an interesting observation, especially because, as found in the demographic analysis, the municipality of Oegstgeest ranks 6th on the list of municipalities in the Netherlands with the highest median income. Based on these inverse modelling results, one could potentially see a relation between the income level of the municipality and the significance that households attribute to financial factors when making decisions about purchasing solar PV systems.

Future Projections:

In Figure 26, the best runs for both the Random Search algorithm and the Bayesian Search algorithm are presented. Both predictions follow a similar curve, with the Bayesian Search curve being slightly more exponential, indicating that this algorithm could result in higher adoption values in the future. When looking at the data analysis for this algorithm in Table 22 and comparing it to the data analysis for the Random Search algorithm in Table 20, it can be noticed that both algorithms have relatively similar values for each parameter for the top 5 runs. Generally, the values for the Bayesian Search algorithm are slightly higher, except for the *comfort* factor. The *social* and *environmental* factors score the highest in the best runs for the Bayesian Search algorithm, indicating that future policies aiming at increasing solar PV adoption rates should focus mostly on these two factors. These results are, however, still inconclusive.

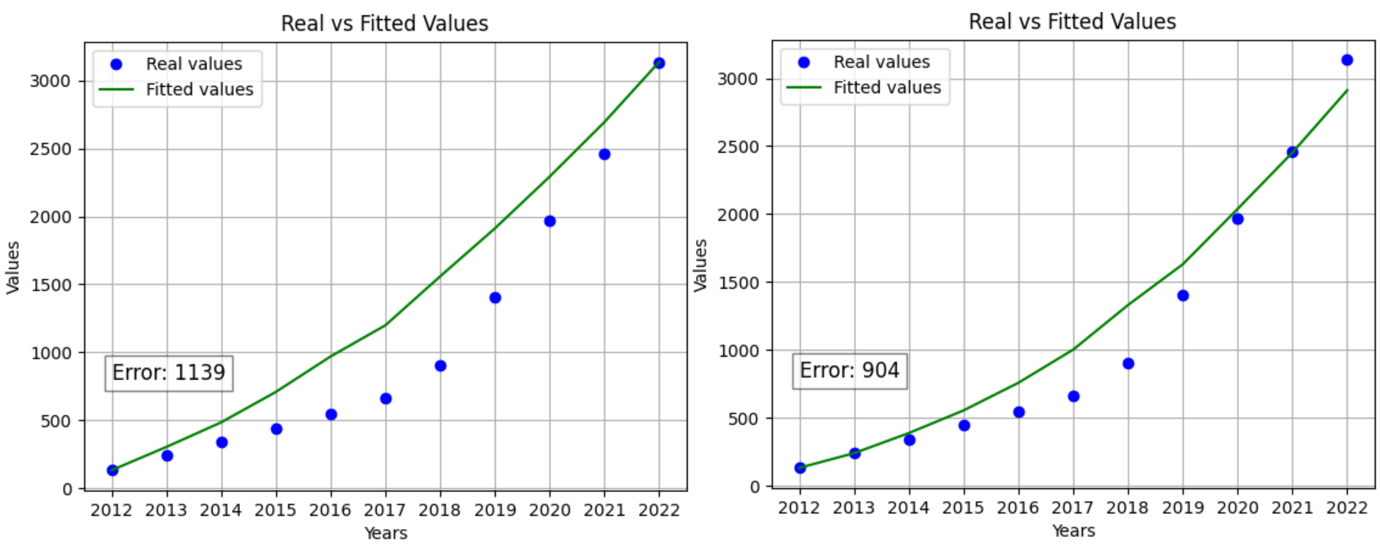


Figure 26: Plot of the real and the predicted solar PV adoption values for Oegstgeest using the Random (left) and Bayesian (right) Search Algorithm

6.5 The Municipality of Vaals

Vaals is located in the south of the Netherlands in the province of Limburg. It is home to 10.092 inhabitants distributed over 5.642 households (AlleCijfers, 2024e). The median income in the municipality is equal to €29.550, which is notably lower than the national median (CBS, 2023a). With 427 inhabitants per km^2, the municipality is sparsely populated by Dutch means. In 2012, Vaals had no recorded solar PV installations. However, starting in 2013, there has been a steady rise in solar PV adoption, with the number of installations increasing each year. A significant acceleration in adoption occurred in 2016 and persisted from 2018 onwards. Despite this, Vaals shows slightly lower solar PV adoption rates compared to the average of other municipalities.
According to Google Trends, the province of Limburg scores average in terms of interest in solar PV, obtaining a score of 88. The municipality of Vaals scores, however, very low within the province, with a score of 0. In addition, only one article was published in 2021, and it was not related to residential solar PV.

6.5.1. Inverse Modelling Results for Vaals

The inverse modelling results for Vaals are presented in the boxplot below. Running the Random Search model took 2 hours and 38 minutes, and the Bayesian Search model took 2 hours and 49 minutes.

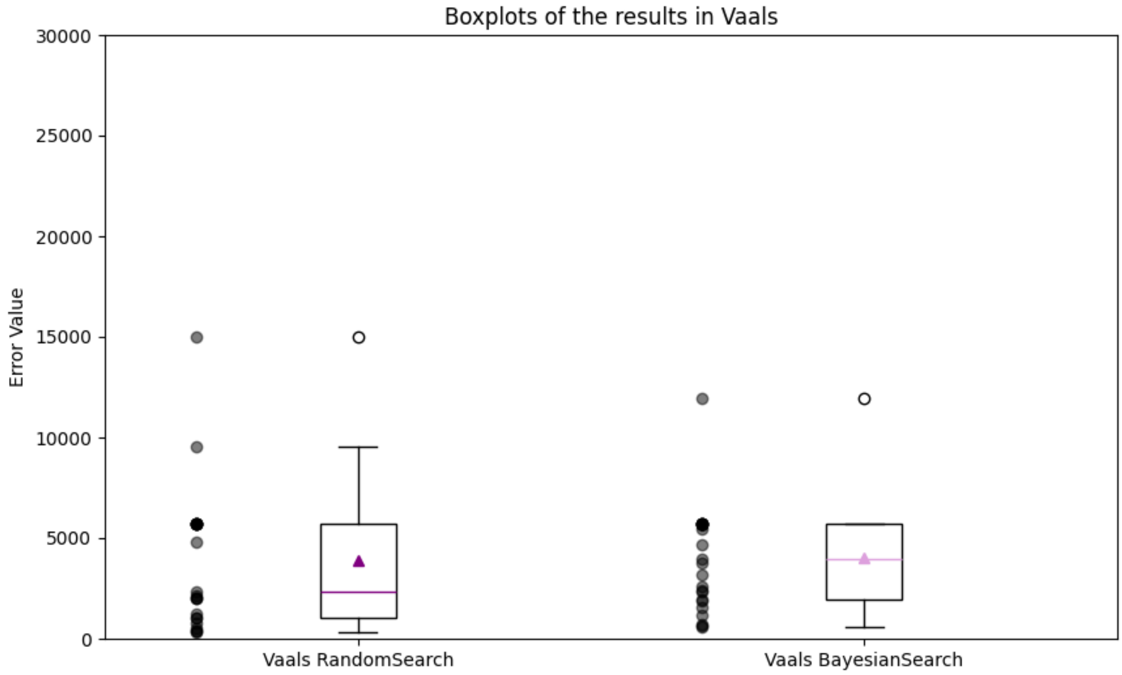


Figure 27: Boxplot of the results of the inverse modelling process for the municipality of Vaals

Random Search Results:
The inverse modelling process is again first executed using the Random Search algorithm. The results from this process can be found in Appendix N. The most prominent aspect of the boxplot of the Random Search model lies in the proportion between its upper and lower whiskers. The top whisker is significantly longer than the bottom whisker, indicating that the upper end of the data distribution is spread over a wider range of values than the lower end. This can also be noticed in the plot when looking at the individual data points. Most individual data points are located relatively close together at the lower end of the error value spectrum. This discrepancy suggests greater variability among the higher data points compared to the lower ones. The positive skew of the data can also be noticed when looking at the median and the mean of the data. The median is equal to a value of 2331, whereas the mean is equal to 3873. As elaborated before, a higher value of the mean compared to the median also indicates right-skewedness. One outlier can be identified, with a value of 14994. The best-performing run has an error value of 320. In Figure 28, the real and predicted adoption values

for this best run are presented. The standard deviation of this model is equal to 3358, presented in Table 23.

Table 23: Mean, median and standard deviation of the Random Search results for Vaals

Mean	3873
Median	2331
Standard deviation	3358

When further analysing the average values of the run parameters, as presented in Table 24, the *social* parameter is again identified as the most important, with a value of 0.66, followed by the *economic* factor. When considering the top 5 best-performing runs, the *economic* factor becomes more important, indicating that financial factors play a significant role in prediction accuracy for this model and this municipality.

Another striking observation is that out of 25 runs, nine runs have the same error value. The reason for this is that for each of these runs, the predicted amount of solar PV installations in the municipality is 0, meaning that no solar PV is adopted at all. This observation can, so far, only be made for this municipality. The average scores of each of the four parameters for these nine runs are further analysed to gain insight into why no solar PV is adopted for these runs. In doing so, it is very noticeable that the sum of all weights attributed to the four parameters is much lower than it would be when considering the average sum of all weights of all the other runs. For comparison: the sum of all weights attributed to the four parameters of the nine runs with the same error value is equal to 1.38, whereas the sum of all weights attributed to all the *other* runs (note that this is the same as the average of *all* runs, as presented in Appendix N, is equal to 2.1. For each of these nine runs, at least one of the parameters' values is equal to 0, meaning that no significance is attributed to this factor by households at all. When considering each of the parameter values individually, one can notice that only the *social* parameter does not have a lower value compared to the average. To sum up, solar PV adoption in the municipality of Vaals for the Random Search model is not present when households in the model place low importance on the different factors that influence solar PV adoption.

Table 24: Summarised data analysis for the Random Search algorithm for Vaals

	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight sum**
Top 5 best runs					
	0.68	0.48	0.3	0.74	2.2
Top 5 worst runs					
	0.44	0.22	0.28	0.62	1.56
Average values of all runs					
	0.51	0.45	0.21	0.66	1.84

Bayesian Search Results:

In the case of the Bayesian Search algorithm, the box of the boxplot is smaller than that of the Random Search algorithm, indicating a smaller variability in error values. The data is hardly skewed, as indicated by the mean and median values (4000 and 3958, respectively). Furthermore, a striking observation can be made when looking at the whiskers of the boxplot, namely the fact that no top whisker is visible. This indicates that the upper quartile range falls within the IQR, indicating a limited variability for the higher data points. One can, however, notice the presence of an outlier, with a value of 11.919. The value of this outlier is too extreme to be represented by the upper whisker in the boxplot because it is located too far from the upper quartile. The minimum error value equals 608, which is higher than that of the Random Search model. The real and predicted solar PV adoption values for this best run are presented in Figure 28. This difference can also be seen in the plots of the minimum error values of each model, as presented in Figure 28. In these graphs, it can be noticed that not only does the Bayesian Search model estimate the final amount of solar PV installations in the municipality less good, it also predicts the general solar PV adoption curve less accurately. However, the Bayesian Search model's standard deviation is lower than that of the Random Search model, with a value of

2490 (see Table 25). Chapter 7 will go into more detail about the importance of the scatter compared with the minimum error value for the IM process.

Table 25: Mean, median and standard deviation of the Bayesian Search results for Vaals

Mean	4000
Median	3958
Standard deviation	2490

When analysing the parameter values of the average of all runs in Table 26, it can be noticed that again the *social* factor is the most important, followed by the *environmental* factor this time (with values of 0.68 and 0.5 respectively). This differs from the Random Search model, where the *economic* factor was more important. When comparing the top 5 best-performing runs with the average parameter values, it becomes clear that the *environmental* and *economic* parameters differ from the average values. This indicates that these two parameters significantly impact the accuracy of the predictions made by the model.

For the Bayesian Search model, it can again be noticed that several runs (8 this time) result in the same error value of 5687, indicating no solar PV adoption at all. These runs have again in common that their sum of parameter weights is relatively low (1.49 compared to 2 for the average of the other runs). This time, however, the most predominant change exists for the *environmental* factor, which only has a value of 0.14 on average for these runs (compared to 0.66 for all other runs). The importance of the *economic* factor is also significantly lower.

Table 26: Summarised data analysis for the Bayesian Search algorithm for Vaals

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.54	0.7	0.2	0.66	2.1
Top 5 worst runs					
	0.48	0.3	0.28	0.66	1.72
Average values of all runs					
	0.38	0.5	0.28	0.68	1.84

6.5.2. Added Value of Inverse Modelling for Vaals

Several key insights emerge when examining the added value of inverse modelling for the municipality of Vaals. Firstly, both traditional analyses and inverse modelling offer valuable perspectives on solar PV adoption dynamics in Vaals. The sparse media coverage of solar PVs in Vaals, as revealed by news article analysis, contrasts with the prominence of social factors emphasised by inverse modelling. This suggests that while media attention may be limited, social considerations significantly influence adoption decisions, potentially contributing to the lower adoption rates observed in Vaals.

However, the insights from inverse modelling also warrant scrutiny. While both Random Search and Bayesian Search algorithms identify social factors as significant, discrepancies arise in the importance attributed to economic factors. The Random Search algorithm underscores the importance of economics in decision-making. Conversely, the Bayesian Search algorithm places less emphasis on economic factors. This raises questions about the consistency and reliability of inverse modelling results, particularly in capturing the complex interplay of factors influencing solar PV adoption in Vaals.

Moreover, an intriguing finding emerged from both models: the absence of solar PV installations in several runs. The analysis of the IM outcomes highlights that this phenomenon could be linked to instances where parameters were accorded lower weights. This suggests that when households assign diminished significance to each factor, solar PV adoption does not occur. However, it should be noted that this observation could only be made for this municipality specifically. Thus, it underscores that the collective weighting of parameters is not the sole determinant. One plausible explanation is the municipality's low-income levels, as revealed by the demographic analysis. It is conceivable that due to these low-income levels, numerous households initially fail to meet the *income vs payback threshold* outlined in Chapter 5, leading to the absence of solar PV adoption in these instances.

Future Projections:
When looking at the best predictions in Figure 28, it can be noticed that the Random Search algorithm performs significantly better than the Bayesian Search algorithm for Vaals. The Random Search algorithm follows a more exponential curve than the Bayesian Search algorithm. This indicates that in the future, solar PV adoption rates will continue to rise strongly in Vaals. Considering the top 5 best runs in the summarised data analysis for the Bayesian Search algorithm for Vaals, it can be argued that future policies aimed at increasing residential solar PV adoption rates should focus on *environmental* and *social* aspects rather than *comfort* factors. The *economic* factor also remains important.

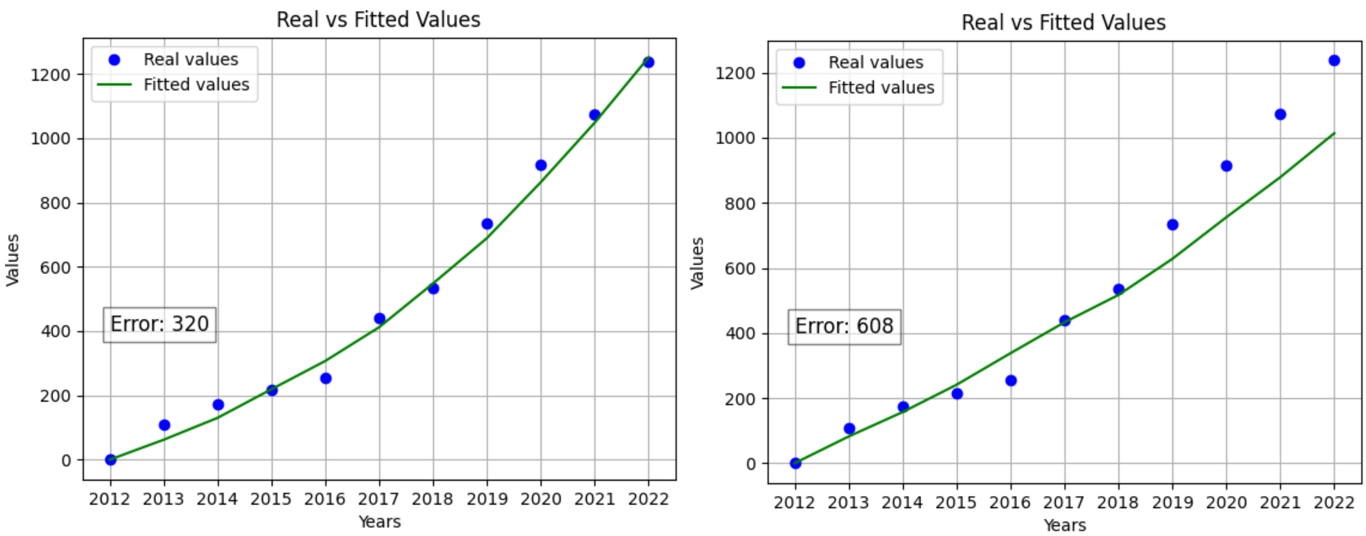


Figure 28: Plot of the real and the predicted solar PV adoption values for Vaals using the Random (left) and Bayesian (right) Search Algorithm

6.6 The Municipality of Westerveld

Finally, the municipality of Westerveld is home to 19.348 inhabitants distributed over 8892 households (AlleCijfers, 2024f). The municipality is very lightly populated, with a population density of 71 persons per km^2. The median income is equal to €38.750, which is very close to the Dutch national median. The municipality of Westerveld exhibits a compelling adoption trend, distinguished by a blend of rapid and sustained growth, setting it apart from neighbouring municipalities. The trajectory begins relatively early, with an initial surge in solar PV adoption evident in 2014. Subsequently, in 2015, the pace steadied for a year before undergoing another strong curve from 2016 onwards. This upward momentum remained stable until around mid-2017, after which a third acceleration occurred in 2021. Overall, it is evident that solar PV adoption in Westerveld is strong and shows a prominent upward trajectory.

According to Google Trends, the province of Drenthe shows a very high interest in solar PV, with a score of 100. Unfortunately, no Google Trends data is available specifically for the municipality of Westerveld. Most news articles published on Google about solar PVs in Westerveld are related to public buildings. However, Westerveld also appears in the national news due to its high solar PV adoption per inhabitant.

6.6.1. Inverse Modelling Results for Westerveld

The Random Search model took 4 hours and 8 minutes to run, compared to 4 hours and 19 minutes compared to the Bayesian Search model. This again means that the Bayesian Search model required roughly 25 seconds of extra time per run, proving the consistency of the additional runtime of the Bayesian Search model for each of the municipalities. In the plot below, the inverse modelling results for the municipality of Westerveld are presented.

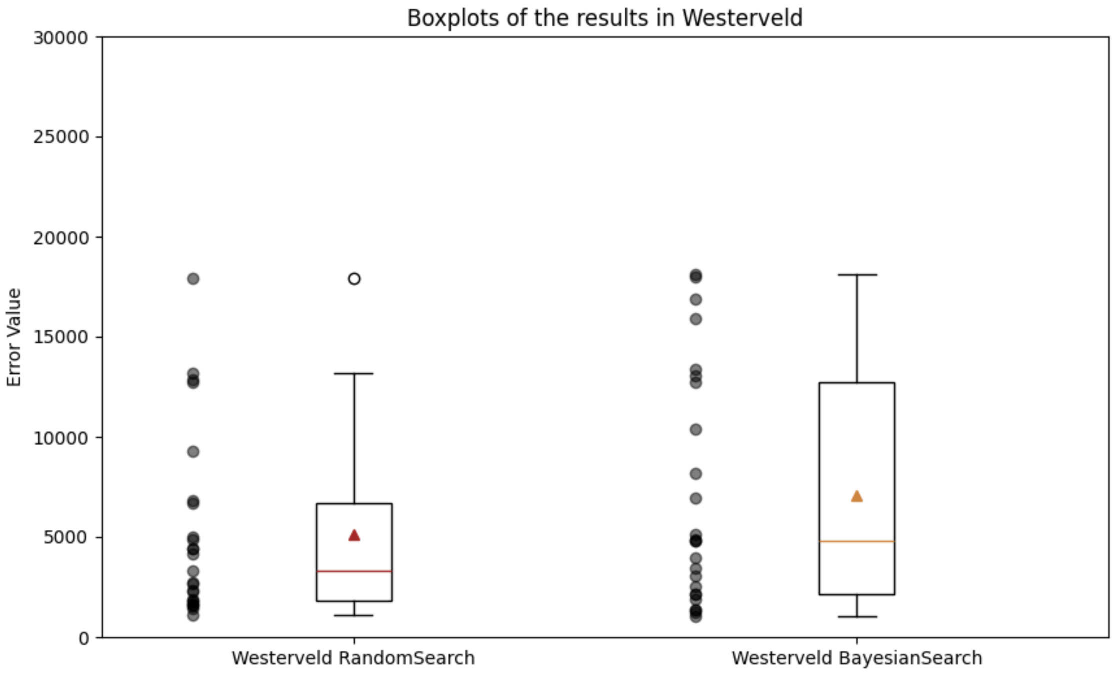


Figure 29: Boxplot of the results of the inverse modelling process for the municipality of Westerveld

Random Search Results:

From the final Random Search boxplot, the first aspect that catches the eye is the length of the upper whisker. This indicates that the upper quartile of the dataset is spread over a wider range of values compared to the lower quartile, suggesting a greater variability among the higher data points. This can also be observed when looking at the data points individually. Up to an error value of roughly 5000, the variability of the data points is significantly smaller than those above that. The data is, therefore, positively skewed. This positive skew can also be observed when comparing the mean value of 5135 with the median value of 3292. This shows that the mean value is higher than the median value, indicating a right skew. The standard deviation equals 4485, and one outlier with a value of 17.942 can be identified. The lowest error value is equal to 1075 and is presented in Figure 30, along with its real and predicted adoption values.

Table 27: Mean, median and standard deviation of the Random Search results for Westerveld

Mean	5135
Median	3292
Standard deviation	4485

From the summarised data analysis in Table 28, it becomes clear that the *social* parameter is, again, the most important, with a value of 0.69. It is followed by the *environmental* and *economic* parameters with values of 0.52 and 0.46, respectively. Furthermore, when looking at the five best-performing parameters, one can notice that the *environmental* parameter becomes slightly less important and has a value of 0.46, tied with the *economic* parameter. *Comfort* and *social* weight also become less important.

Table 28: Summarised data analysis for the Random Search algorithm for Westerveld

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.46	0.46	0.26	0.6	1.78
Top 5 worst runs					
	0.38	0.50	0.38	0.72	1.98
Average values of all runs					
	0.46	0.52	0.38	0.69	2.05

Bayesian Search Results:
For the Bayesian Search model, the box size in the boxplot is very evident as it is significantly larger than that of the Random Search model. This indicates greater variability in the dataset. Again, The top whisker is larger than the bottom, indicating a positive skew. This positive skew can also be identified when looking at the individual data points, showing a much greater variability in the higher data points compared to the lower data points. Both the median, with a value of 4825, and the mean, with a value of 7089, are much larger for the Bayesian Search model than for the Random Search model. The standard deviation is also larger, with a value of 5746. This larger variability is caused by a greater number of high error values: a total of 8 runs (out of 25) have an error value larger than 10.000. The minimum error value, on the other hand, is equal to 1011, which is smaller than that of the Random Search model and is presented in Figure 30.

Table 29: Mean, median and standard deviation of the Bayesian Search results for Westerveld

Mean	7098
Median	4825
Standard deviation	5746

When analysing the average parameter values over all runs, as presented in Table 30, it becomes evident that the *social* and *environmental* parameters are the most important. The *comfort* factor is again of the least importance. An interesting observation can be made when considering the top 5 best-performing runs: the *environmental* factor specifically becomes much more important with a value of 0.72. This indicates that predictions with higher accuracy are made when households value environmental aspects more in the model. This observation becomes even more evident when also considering the top 5 worst-performing runs. For these runs, the average *environmental* factor has a value of 0.2, significantly lower than the value observed for the average value of all runs.

Table 30: Summarised data analysis for the Bayesian Search algorithm for Westerveld

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.58	0.72	0.32	0.7	2.32
Top 5 worst runs					
	0.32	0.2	0.18	0.64	1.34
Average values of all runs					
	0.44	0.48	0.33	0.66	1.91

6.6.2. Added Value of Inverse Modelling for Westerveld

The inverse modelling process indicates that the *social* and *environmental* parameters are especially important for prediction accuracy in this municipality. Considering that the municipality of Westerveld has a relatively high solar PV adoption rate, it is especially interesting to see this relation between the adoption rates and the importance of the *environmental* parameter, which is considerably higher compared to its value in the other municipalities. It should be noted, though, that the relatively high scatter of data in this municipality raises questions about whether these IM results are conclusive. This discussion will continue in Chapter 7.

Future Projections:
The best runs of each algorithm, as presented in Figure 30, show that the algorithms perform relatively similarly. The Bayesian algorithm shows a slightly more linear curve than the Random Search algorithm. These graphs indicate that even though solar PV adoption in Westerveld will continue to grow, this growth is not as exponential as in the other municipalities (e.g. Dantumadiel). It should also be noted that even though the Random Search and the Bayesian Search algorithms show very similar minimum error values for the best runs, their values for the top 5 best runs in the summarised data analysis in Tables 28 and 30 are quite different. Both algorithms present the *environmental* and *social* factors as the most influential in the decision-making process. However, the weights for the Bayesian Search algorithm are stronger. Due to the inconclusiveness of the results, it is difficult to provide information on the potential focus areas of future policymaking.

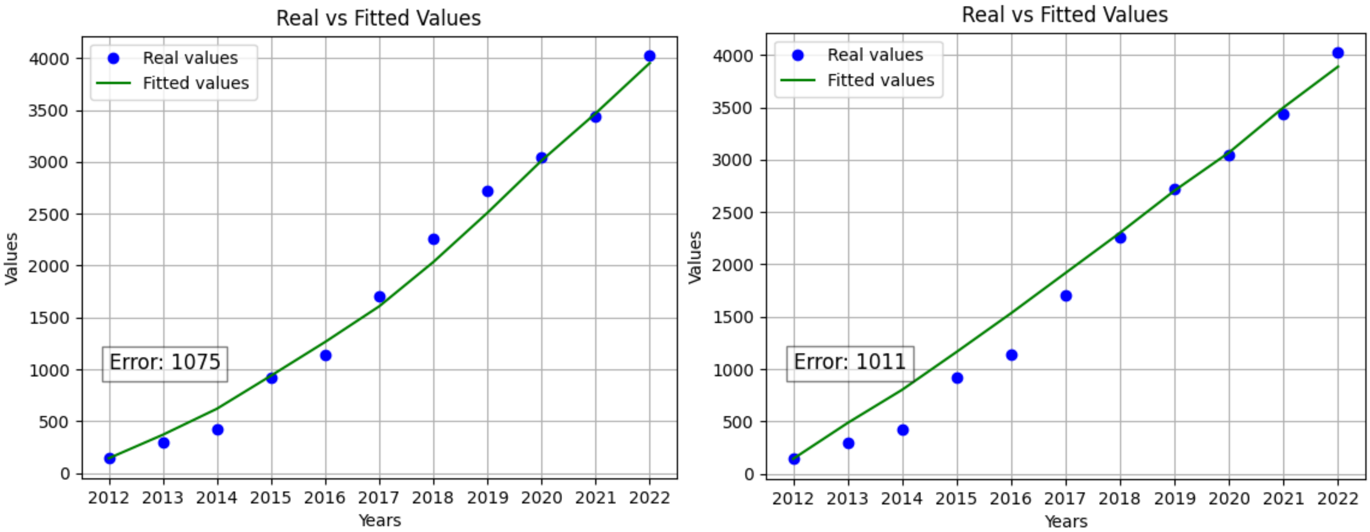


Figure 30: Plot of the real and the predicted solar PV adoption values for Westerveld using the Random (left) and Bayesian (right) Search Algorithm

6.7 Inter-Municipal Comparative Analysis

In the inter-municipal analysis, the results from the six municipalities are compared. First, the boxplots of each municipality's results are compared. Thereafter, the actual parameter values of the outcomes are compared.

6.7.1. Boxplot Comparison:

In Figures 25 and 26 below, the boxplots of the municipalities are presented side by side so that they can be compared more easily. Firstly, the Random Search models and the Bayesian Search models are compared. In terms of scatter, the Random Search model performs better for 4 out of 6 municipalities (namely Bloemendaal, Dantumadiel, Laren and Westerveld). This is evident from the size of the boxes in the boxplots: a smaller box indicates lower scatter and, therefore, less variability. The lower scatter of the Random Search model can also be proven by taking the average standard deviation of all municipalities for each of the algorithms. The average standard deviation for Random Search equals 3496 and 4381 for the Bayesian Search algorithm. Additionally, regarding the lowest minimum error value, both algorithms produce equally good outcomes. For 3 out of 6 municipalities, Random Search yields a lower minimum error value (for Bloemendaal, Laren and Vaals). In comparison, Bayesian Search achieves a lower minimum error value for the other three municipalities (Dantumadiel, Oegstgeest and Westerveld). The discussion on whether scatter or minimum error value is a more important determinant of the quality of the IM model is presented in Chapter 7. In Appendix Q, the mean, median and standard deviations of the results of the inter-municipal comparative analysis are presented.

Furthermore, it can be observed that the Bayesian Search model generally produces more outliers, with 9 in total compared to 6 outliers for the Random Search model. Additionally, the median error value for each municipality is lower with the Random Search algorithm than with the Bayesian Search algorithm, indicating that the Random Search algorithm more accurately predicts solar PV adoption curves. On the other hand, the mean values do not significantly favour either algorithm. This is because mean values are more sensitive to outliers and extreme values (both positive and negative), making them a less robust measure of performance. Interestingly enough, these observations indicate a slightly better performance of the Random Search algorithm compared to the Bayesian Search algorithm. This is surprising because the Bayesian Search algorithm should theoretically produce better outcomes. A discussion of the performance between the two algorithms is presented in Chapter 7. An important note is that the minimum error values cannot be accurately compared across different municipalities. This is because the error value is based on the difference between the predicted and actual number of solar PV installations in a municipality. In larger municipalities, the absolute difference between predicted and actual installations is generally smaller compared to larger municipalities. This results in a distorted view of the results since the relative or percentage differences between the municipalities might be similar. This limitation arises from how the error value is calculated. Although theoretically, this error value could account for the size of the municipality, it is not desirable because the original ABM already factors in the number of households in the municipality, meaning this variable would be considered twice.

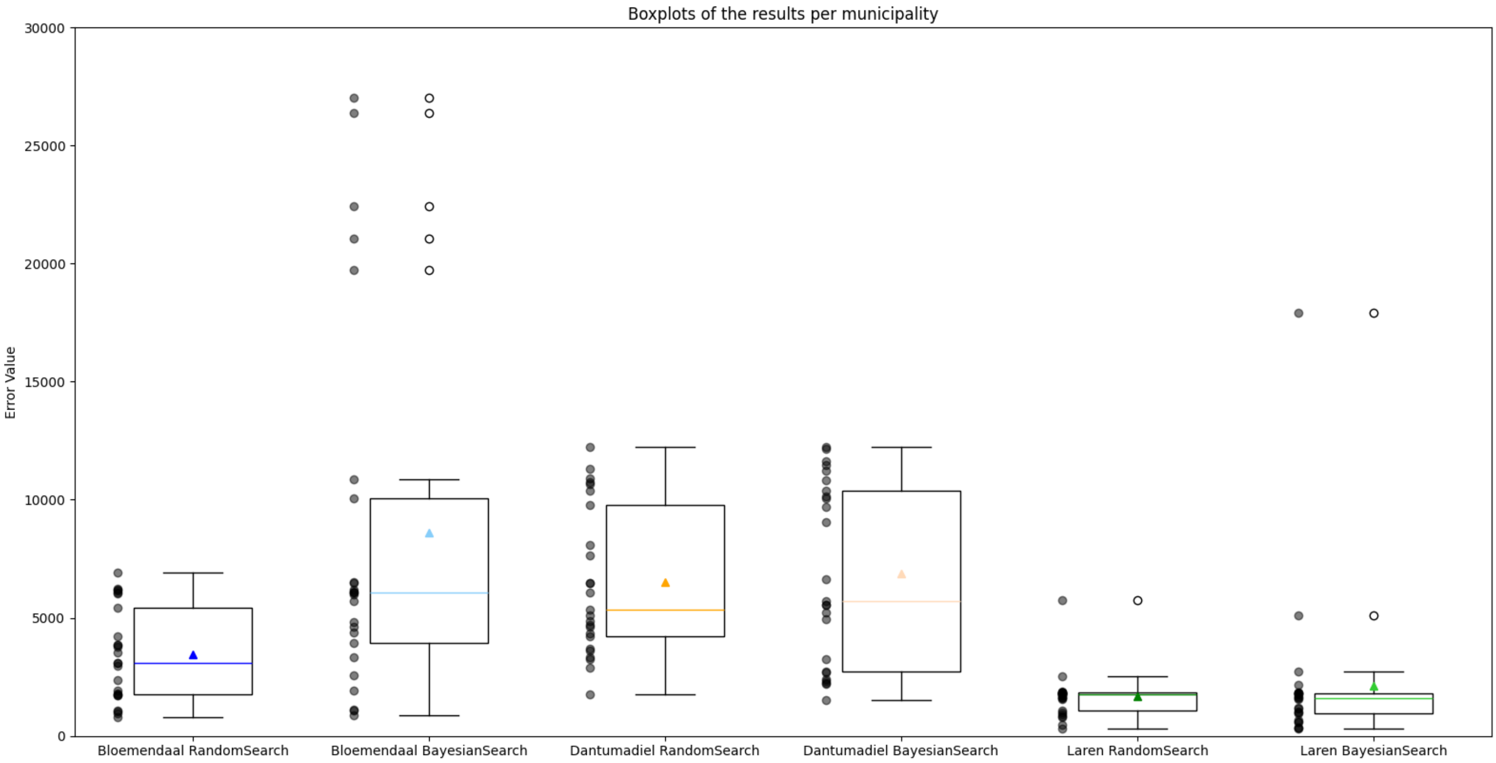


Figure 31: Boxplots of the results for the municipality of Bloemendaal, Dantumadiel and Laren

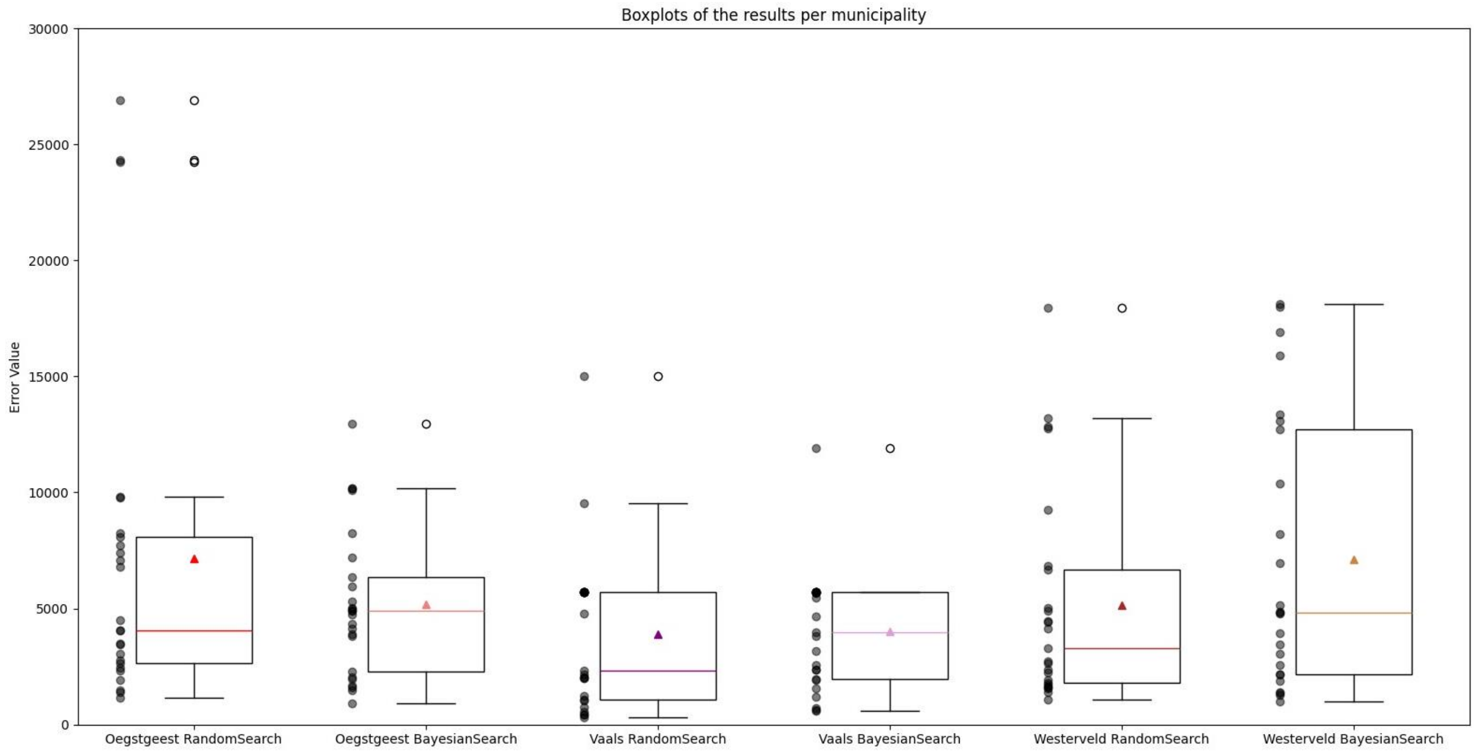


Figure 32: Boxplots of the results for the municipalities of Oegstgeest, Vaals and Westerveld

6.7.2. Parameter Value Comparison

The actual parameter values between the municipalities are differentiated between the two algorithms. This is because the two algorithms could potentially have different optimal parameter values for the municipalities, making it difficult to compare them.

Random Search models: average values

When analysing the average values of the parameters of the Random Search models in Table 31, the first parameter to consider is the *economic* factor or *weight_eco*. Within this parameter, Vaals emerges with the highest weighting (of 0.51) despite being associated with the lowest income among the municipalities. This suggests that, for this specific ABM, households in Vaals attribute high importance to financial considerations when deciding on solar PV system purchases. On the other hand, the *weight_eco* parameter exhibits its lowest values in Bloemendaal and Oegstgeest (with values of 0.33 each), two of the municipalities with the highest income levels. This observation hints at a pattern between income and the significance of financial factors in solar PV decision-making. However, this pattern does not hold for Dantumadiel and Westerveld, which share similar income levels but display notable differences in values for the *economic* parameter.

Dantumadiel and Westerveld do, however, display the highest values for the *environmental* parameter (0.50 and 0.52, respectively). Interestingly enough, these two municipalities are also the two municipalities with the highest solar PV adoption rate, indicating that within this model, households in municipalities with high adoption rates place higher importance on the environmental impact of solar PV. This relation is, however, not evident for the other municipalities. The municipality of Laren, having the lowest adoption rates, also displays a low value for the *weight _env* parameter. The municipality of Oegstgeest also has a low value for the *environmental* parameter, but its scores are above average for solar PV adoption.

The *comfort* parameter, or *weight_cof*, has the highest parameter value for the municipality of Oegstgeest. The lowest value can be observed for the municipality of Vaals. No specific patterns across municipalities for this parameter could be found.

The social parameter has the highest value of all parameters, which indicates that it significantly impacts the model's performance. The social parameter is highest for the municipality of Laren and lowest for the municipalities of Oegstgeest and Vaals. No specific patterns across municipalities could be found for this parameter.

Table 31: Average values of the parameters of the Random Search models

Random Search Models: Average of All Runs					
	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight_sum**
Bloemendaal	0.33	0.49	0.30	0.68	1.80
Dantumadiel	0.36	0.50	0.34	0.75	1.94
Laren	0.36	0.42	0.31	0.80	1.89
Oegstgeest	0.33	0.42	0.47	0.66	1.88
Vaals	0.51	0.45	0.21	0.66	1.84
Westerveld	0.46	0.52	0.38	0.69	2.05

Random Search models: best scoring runs

When considering the *economic* parameter of the top 5 best-scoring runs of the Random Search models, the municipality of Vaals displays the highest value again and Bloemendaal the lowest value. Furthermore, the weights are exaggerated because Vaals displays an even higher value of 0.68) than for the average of all runs, and Bloemendaal has an even lower value (0.26). However, the pattern between income and the economic parameter is much more visible for the five best-performing runs: the three wealthiest municipalities (Bloemendaal, Laren and Oegstgeest) display the three lowest values for the *weight_eco* parameter.

For the *environmental* parameter, the values for most municipalities are actually in a very similar range (between 0.46 and 0.48). Only Dantumadiel (with a value of 0.56) and Bloemendaal (with a value of 0.36) display different values. Contrary to the results with the average values, no strong pattern can be recognised here for the *environmental* parameter. The same holds for the *comfort* parameter, where Bloemendaal displays the highest value of 0.52, and Westerveld displays the lowest value of 0.26. For this parameter, no specific patterns across municipalities could be found.

Finally, the values of the *social* parameter are again the highest of all parameters, indicating a significant impact on the model's performance. The value of the social parameter is slightly higher and slightly lower for some municipalities than for the average runs. The parameter value remains the highest for Laren (0.78) and the lowest for Oegstgeest.

Table 32: Top 5 best-performing result values of the Random Search models

Random Search Models: Top 5 Best-Performing Runs					
	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight_sum**
Bloemendaal	0.26	0.36	0.52	0.68	1.82
Dantumadiel	0.52	0.56	0.40	0.76	2.24
Laren	0.42	0.48	0.40	0.78	2.08
Oegstgeest	0.32	0.46	0.38	0.54	1.70
Vaals	0.68	0.48	0.30	0.74	2.20
Westerveld	0.46	0.46	0.26	0.60	1.78

Bayesian Search models: average values

For the Bayesian Search models, the average values for the *economic* parameters are quite different compared to the average values for this parameter in the Random Search model. Whereas for the Random Search model, a slight pattern between income and the weight of the *economic* parameter could be found, this pattern is not present for the Bayesian Search models. In this case, the *economic* parameter has the highest values for the municipalities of Bloemendaal and Westerveld and the lowest for Dantumadiel and Oegstgeest.

The municipality of Laren is the only municipality that attributes significantly less weight to the *environmental* parameter than the other municipalities. The municipality of Oegstgeest, on the other hand, attributes the most weight to this parameter. No specific patterns across municipalities for this parameter could be found.

The *comfort* parameter holds relatively low weights across all municipalities. Interestingly, this characteristic is also evident in Random Search models. It suggests that while the *comfort* parameter has a relatively strong impact on the model's performance, its effect is inverse compared to the *social* parameter. Unlike the *social* parameter, which benefits from high values for optimal results (yielding low error values), the *comfort* parameter seems to require lower values, though not excessively low ones. The *social* parameter is again significantly higher for each of the municipalities compared to the other parameters. However, no specific patterns across the municipalities could be found.

A final interesting remark can be made regarding the sum of all the parameter weights for each municipality. From this sum, a pattern can be recognised related to the solar PV adoption rate of each municipality. Interestingly, the three municipalities with the highest solar PV adoption rates also have the highest sum of parameter weights. On the other hand, the sum of parameter weights of the municipality with lower adoption rates is lower. This indicates that a pattern exists between the significance that households attribute to all different aspects that influence solar PV adoption and the actual adoption rates.

Table 33: Average values of the parameters of the Bayesian Search models

Bayesian Search Models: Average of All Runs					
	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight_sum**
Bloemendaal	0.45	0.47	0.29	0.65	1.86
Dantumadiel	0.33	0.55	0.34	0.76	1.97
Laren	0.39	0.24	0.37	0.69	1.68
Oegstgeest	0.36	0.57	0.39	0.72	2.04
Vaals	0.38	0.50	0.28	0.68	1.84
Westerveld	0.44	0.48	0.33	0.66	1.91

Bayesian Search models: best scoring runs

Finally, the top 5 best-scoring runs for the Bayesian Search models are considered, as presented in Table 34. For the *economic* parameter, the highest value is displayed for the municipality of Westerveld and the lowest for Laren. This would initially suggest a pattern between the significance that households attribute to financial aspects in the decision-making process for solar PV and their actual adoption behaviour since Westerveld has one of the highest adoption rates and Laren has the lowest. However, when further analysing the values of the other municipalities for this parameter, one can notice that this pattern does not hold. One interesting observation, though, is that the average of the weight_eco values for the top 5 best-performing runs (an average value of 0.49) is significantly higher than for the average of all runs (an average value of 0.39). This would indicate that this parameter value seems to have quite an impact on the performance of the model, indicating that higher values for the *economic* parameter lead to better predictions.

For the *environmental* parameter, the municipality of Westerveld displays the highest value, and the municipality of Laren portrays the lowest value. The low value for Laren should not be noticed, as was also already apparent for the average values. Again, no specific patterns across the municipalities could be found, though.

For the *comfort* parameter, the highest value was found for the municipality of Laren (0.68) and the lowest for Vaals (0.20). However, the analysis does not indicate any patterns across the municipalities.

Finally, the *social* parameter is again the highest of all parameters on average. The highest value was found for Laren, and the lowest value was found for Oegstgeest. It should be noted that the previously mentioned pattern for the sum of all parameter weights for each municipality cannot be recognised for the top 5 best-performing runs.

Table 34: Top 5 best-performing result values of the Bayesian Search models

Bayesian Search Models: Top 5 Best-Performing Runs					
	Weight_eco	**Weight_env**	**Weight_cof**	**Weight_soc**	**Weight_sum**
Bloemendaal	0.56	0.54	0.34	0.76	2.20
Dantumadiel	0.50	0.60	0.54	0.74	2.38
Laren	0.38	0.24	0.68	0.82	2.12
Oegstgeest	0.40	0.52	0.32	0.56	1.80
Vaals	0.54	0.70	0.20	0.66	2.10
Westerveld	0.58	0.72	0.32	0.70	2.32

Concluding remarks:

When considering the results of the IM process of the different municipalities, the question of what predicts a good run and whether this is generic or municipality-specific can be raised. The IM process has shown that the *social* parameter values are consistently the highest for the top 5 best-performing runs, indicating that this parameter value seems to have the highest impact on the performance of the model. Furthermore, the IM process has shown that a high sum of weighted values consistently leads to better fits with the observed data. This indicates that for this ABM, a good run can generally be predicted by looking at the *social* parameter value and the sum of weighted values. This is true not just for a single municipality but for all of them, indicating that the *social* parameter and the sum of weighted values are generic indicators for the performance of a run.

Several promising patterns emerge from the data, particularly regarding the influence of economic parameters and municipal income in the Random Search models. While varying across municipalities, these patterns suggest unique indicators for good runs specific to a particular area. This indicates that they are, contrary to the *social* parameter and the sum of weighted values, municipality-specific instead of generic. Although these findings currently show some variability and require cautious interpretation, they point to the strong potential of inverse modelling to deeply explore and understand the dynamics underlying solar PV adoption. Enhancing the robustness of these municipality-specific patterns will be key to fully harnessing this potential. Chapters 7 and 8 will discuss the factors contributing to this variability, including what factors predict a good run per municipality, and outline strategies for achieving more reliable results in future studies.

6.8 The Municipality of Laren: Increased Splits and Iterations

When considering the results of each municipality, one can distinguish differences between the results of *all* runs and the *well-scoring* runs (e.g. the top 5 highest-performing runs). This observation suggests that not all runs are created equal and that some configurations consistently yield better results. This insight leads to a critical question in the research: Does allowing the model to run for more iterations and splits increase the likelihood of identifying these high-quality runs, or is the occurrence of good runs more a matter of chance? If extending the search time leads to more frequent identification of high-performing runs, it would underscore the value of persistent and extended searches in optimising model outcomes in the IM process. Conversely, if good runs appear randomly, irrespective of the number of iterations, it might suggest a need for different optimisation strategies or model refinements.

To explore this, an experiment using the municipality of Laren as a case study is proposed. The experiment involves increasing the number of runs and iterations significantly to observe whether a more extended search leads to the discovery of more good runs. Specifically, 30 iterations over five splits will be executed. The reason behind the number of splits lies in the search strategy behind the Bayesian Search algorithm: an increased number of splits increases the number of search areas for the Bayesian algorithm. The number of iterations is doubled compared to before, resulting in a total of 150 fits per run instead of 30 fits before. The municipality of Laren is chosen for this experiment because its existing good results compared to the other municipalities (see Figure 31) make an excellent benchmark to test if extended search durations can uncover even better solutions. Furthermore, its small size requires less computational power, allowing for more efficient experimentation.

6.8.1. Random Search Results:

Figure 33 presents a comparative boxplot of the Random Search algorithm's inverse modelling process results. On the left, one can find the outcomes from the experimental setup, where the model executed 30 iterations and five splits across 25 runs. The 'original' inverse modelling results are displayed on the right, as detailed in Chapter 6.3. These were obtained using 15 iterations and two splits over 25 runs. These results are plotted side by side to facilitate a clear comparison between the outcomes of the experiment and the initial results for the municipality of Laren.

To better highlight the details of the data, the y-axis range has been adjusted to 0 to 6000, a narrower scale compared to the 0 to 30,000 range used in previous boxplots. This change enhances the visibility of the graphical details. The total time to run this experiment was 12 hours and 9 minutes, whereas the original setup took significantly less time, at 2 hours and 26 minutes. This stark contrast in time underscores the considerably higher computational effort required when running the model with more iterations and splits.

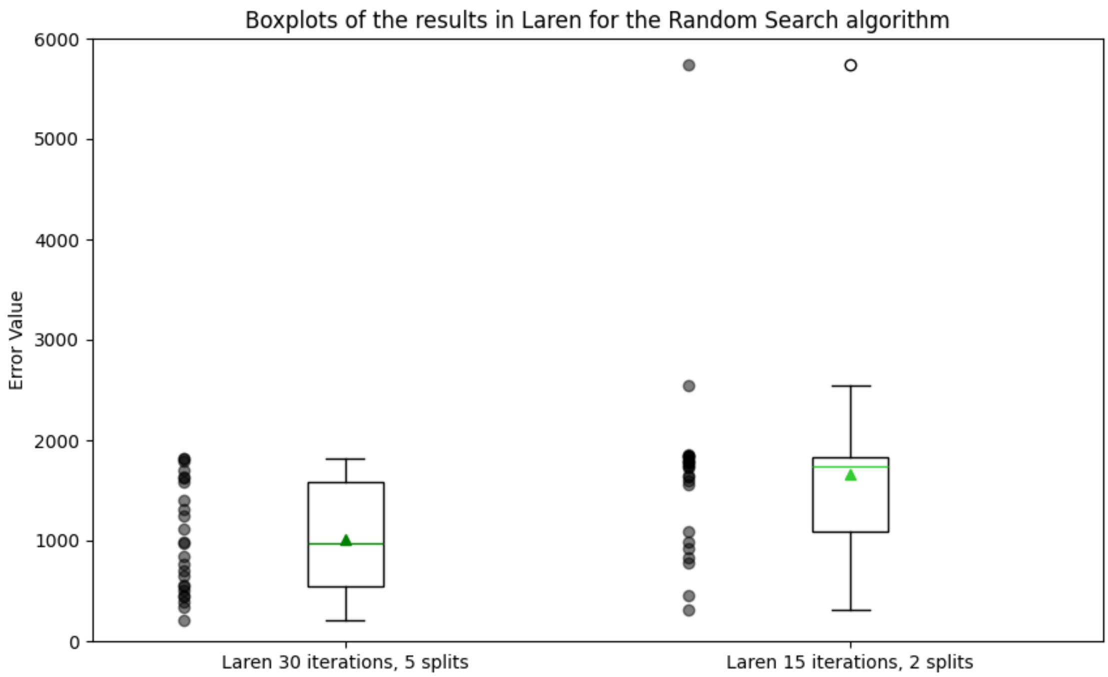


Figure 33: Boxplot of the comparison between runs with the experimental (left) or original (right) setup for the inverse modelling process in Laren for the Random Search algorithm

When analysing the boxplots of the experimental and original setups, several differences in the data distribution become apparent. Firstly, the median, represented by the horizontal line in the box, is noticeably lower for the experimental setup compared to the original setup, with median values of 979 and 1738, respectively. This median line reflects the central value of the data. Similarly, the mean, indicated by the triangle, is also lower for the experimental setup, with a mean of 1016 versus 1668 for the original setup, as detailed in Table 35. Furthermore, the differences in the size of the boxes can be noticed. The box for the experimental setup is larger, indicating a broader interquartile range (IQR). This suggests that the middle 50% of the data in the experimental setup is more spread out.

However, the whiskers for the experimental setup are shorter, implying that the data points beyond the quartiles are less spread out compared to the original setup. This indicates that the majority of the data is closer to the quartiles, with fewer extreme values or outliers. Consequently, with its larger box and shorter whiskers, the experimental setup exhibits greater variability within its central values but is more compact overall. This is corroborated by the standard deviation, which is 524 for the experimental setup, significantly lower than the 974 for the original setup, indicating less overall variability. Additionally, the original setup has an outlier, which is not present in the experimental setup.

Table 35: Comparison of the descriptive statistics between runs with the experimental (left) or original setup (right)

	30 iterations, five splits	**15 iterations, two splits**
Mean	1016	1668
Median	979	1738
Standard deviation	524	974

The best run for the experimental setup achieves an error value of 207, while the best run for the original setup has a higher error value of 314. In Figure 34, these best runs for each setup are displayed, showing how the fitted values from the experimental setup align more closely with the actual solar PV adoption values in the municipality of Laren.

Overall, the experimental setup demonstrates superior performance in resembling the real solar PV adoption data. This is evidenced by its lower mean, median and standard deviation compared to the original setup, indicating more accurate and consistent results. Moreover, the experimental setup's best-performing run not only achieves a lower error value but also delivers a more precise fit than the original model.

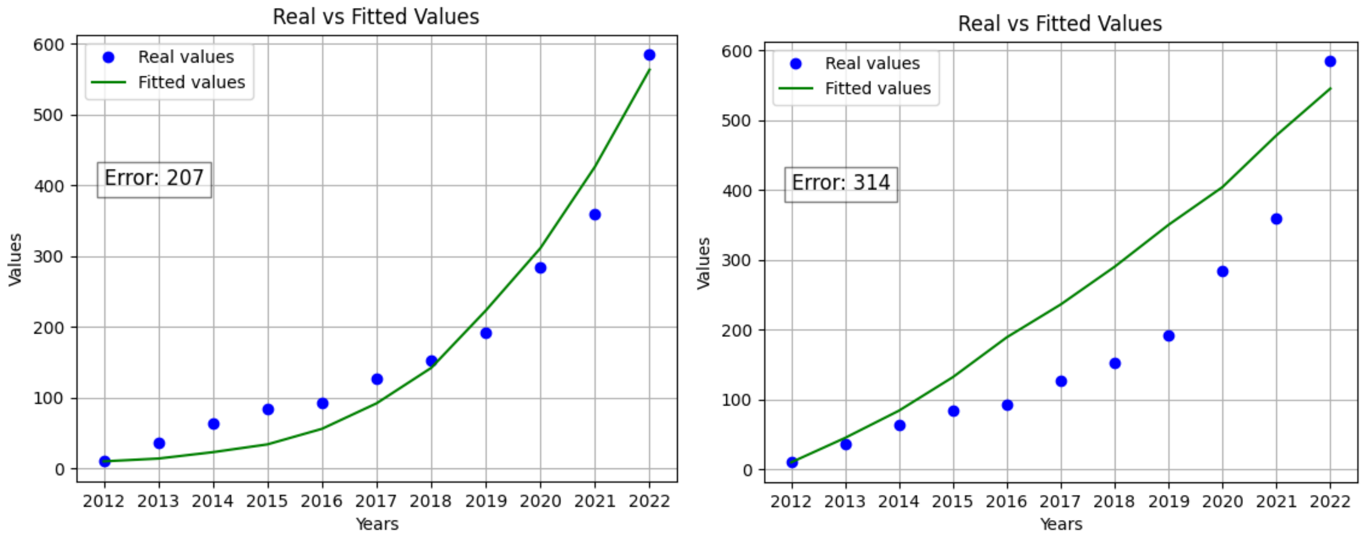


Figure 34: : Plot of the real and the predicted solar PV adoption values for Laren for the experimental (left) and the original (right) setup

When considering the weights of the different factors in the inverse modelling process for the environmental setup, it can first be noticed that social weight is still the most important overall. The *environmental* factor, on the other hand, appears to be the least important, with a value of 0.28. An interesting observation hereby is the fact that the average values of all runs (for instance, when looking at the sum of weights) always seem to score higher than either the best or the worst runs. This would indicate that the performance of the model does not necessarily depend as much on the strength of the weights themselves but more so on their relations to each other.

When comparing the weights of the different parameters of the experimental setup to the original setup, it can be noticed that these weights are generally lower, except for the *comfort* factor. This indicates that when increasing the performance of the model for Laren, the weight of the *comfort* factors becomes more important while the weight of the other parameters becomes less important. This trend appears to be the most apparent for the *environmental* factor.

Table 36: Summarised data analysis of the experimental setup for the municipality of Laren

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.34	0.34	0.40	0.68	1.76
Top 5 worst runs					
	0.10	0.36	0.40	0.66	1.52
Average values of all runs					
	0.45	0.28	0.34	0.72	1.78

6.8.2. Bayesian Search Results

In Figure 35, the comparative boxplot of the Bayesian Search algorithm's IM results is presented. On the left, the outcomes from the experimental setup can be found, where the model executed 30 iterations and five splits across 25 runs. The original results are displayed on the right. These original results can also be found in Chapter 6.3 and are obtained using 15 iterations and two splits over 25 runs. These results are plotted side by side to facilitate a clear comparison of the outcomes of the experiment and the initial results. The y-axis range has again been adjusted, this time to 0 to 18000, due to the presence of an outlier in the original results. The total time to run this experiment was 12 hours and 21 minutes, whereas the original setup took 2 hours and 36 minutes.

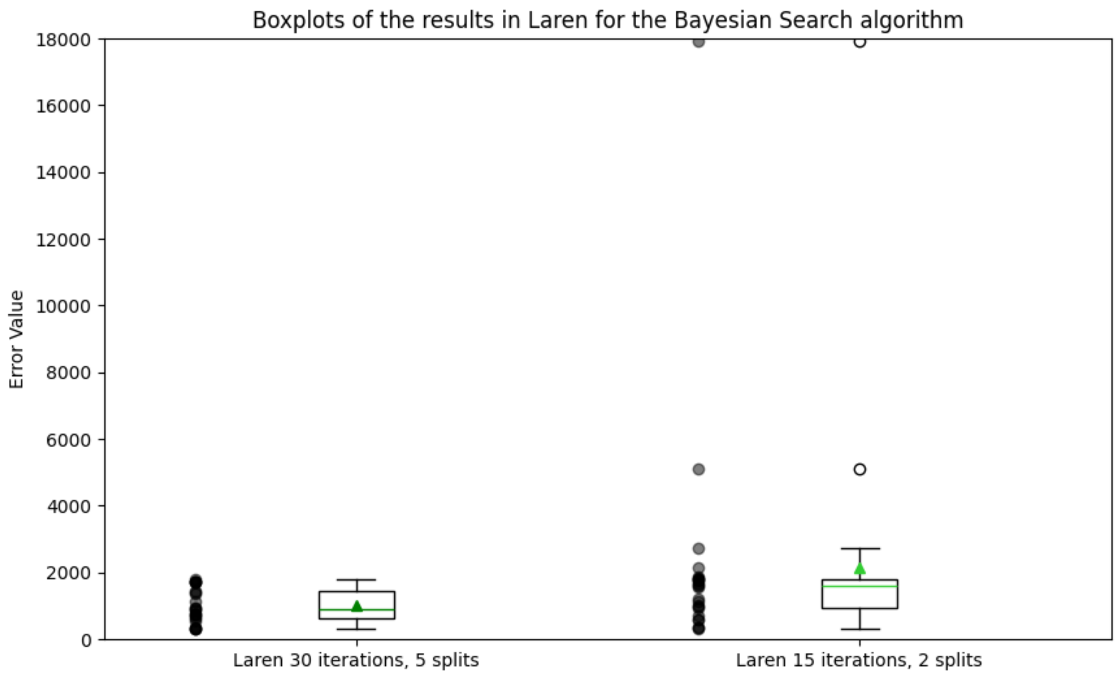


Figure 35: Boxplot of the comparison between runs with the experimental (left) or original (right) setup for the inverse modelling process in Laren for the Bayesian Search algorithm

When analysing the boxplots in the figure above, it can first be noticed that the original setup has two outliers that were not present in the experiment. The size of the boxes is relatively similar, indicating that the interquartile range did not change significantly in the experiment. The whiskers, on the other hand, are noticeably smaller for the experimental setup. All in all, this indicates that the data for the experimental setup is less spread out. Furthermore, one can notice how, in the experimental setup, the mean and median are relatively close to each other, with values of 999 and 907, respectively. For the original setup, these values were equal to 2136 and 1578, respectively (see Table 37). The standard deviation of the experimental setup is equal to 516, which is significantly smaller than the standard deviation observed for the original setup, which was equal to 3359. All in all, this indicates that the experimental setup performs better than the original setup since it produces generally lower error values and a lower spread as well.

When comparing the descriptive statistics of the experimental setup of the Bayesian Search algorithm (see Table 37) with the Random Search algorithm (see Table 35), it can be noticed how the Bayesian Search algorithm now performs better in every aspect than the Random Search algorithm. The mean, median and standard deviation are all lower in the experimental setup for the Bayesian Search algorithm. This is an interesting observation, especially considering the fact that for the original setup, the Random Search algorithm performed better than the Bayesian Search algorithm. This would indicate that the performance of the Bayesian Search algorithm strongly improves when the number of splits and iterations is increased.

Table 37: Comparison of the descriptive statistics between runs with the experimental (left) or original setup (right)

	30 iterations, five splits	**15 iterations, two splits**
Mean	999	2136
Median	907	1578
Standard deviation	516	3359

The best run for the experimental setup achieved a minimum error value of 304, while the best run for the original setup achieved a minimum error value of 316, again indicating superior performance for the experimental setup. In Figure 36, the plots of the real and predicted adoption values for Laren for both the experimental and the original setup for the Bayesian Search algorithm are presented. It should be noted, though, that the experimental setup for the Random Search algorithm resulted in a minimum error value of 207. This indicates that the Random Search algorithm still performs better than the Bayesian Search algorithm in the experiment when it comes to the best run. However, as presented by the boxplots and descriptive statistics, the Bayesian Search algorithm now performs better overall.

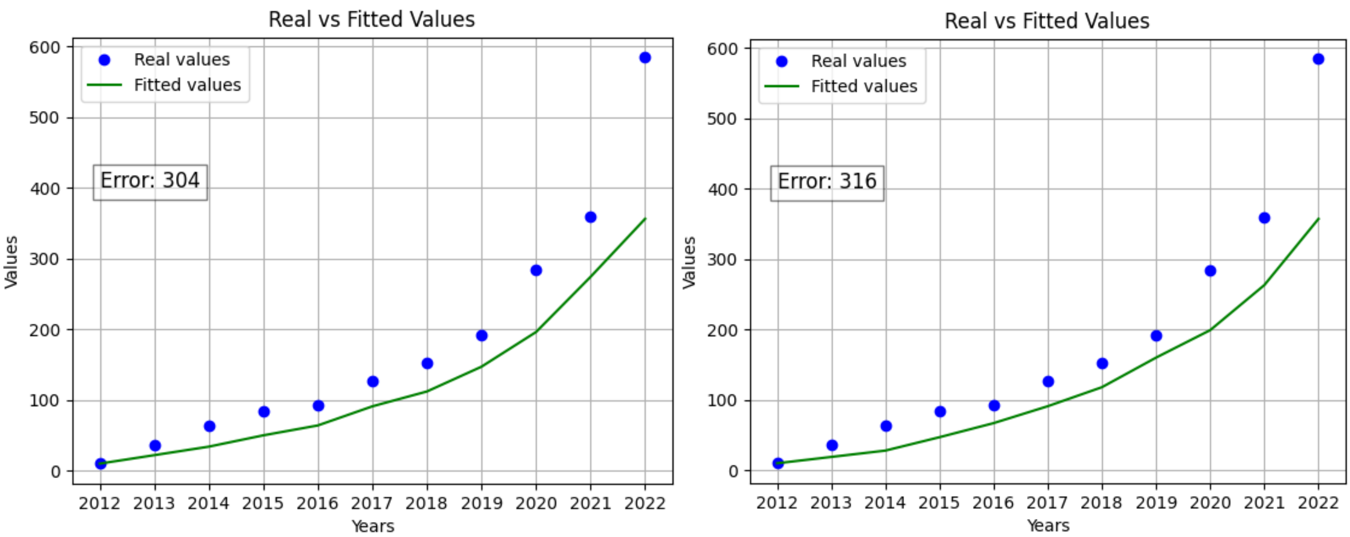


Figure 36: Plot of the real and the predicted solar PV adoption values for Laren for the experimental (left) and the original (right) setup

Table 38 presents a summarised overview of the data analysis of the experimental setup for the municipality of Laren. When considering the average values of all runs, it should first be noted that the *social* weight is again very high, whereas the *environmental* weight is actually very low. Both economic and social weights score average. When considering the top 5 best runs, however, the weight for the *environmental* factor becomes even lower. This indicates that in the runs that resemble reality the closest, households attribute a low weight to the *environmental* factor. When considering the top 5 worst runs, on the other hand, the *environmental* weight becomes slightly higher. Furthermore, the *economic* weight strongly decreases for the top 5 worst runs, indicating that in the runs that resemble reality the least, households attribute a low weight to this factor. Finally, it should be noted that the best-performing runs have a higher sum of weights compared to the worst-performing runs.

When comparing the weights of the factors of the experimental setup with those of the original setup (as presented in Table 18), it can first be noted that the weighted sum of all factors is higher for the average values of all runs in the experimental setup. Furthermore, when considering the best-performing runs, it can be noticed that the *environmental* factor has become less important in the experimental results. The same is true for the *comfort* factor, which now has a value of 0.46 compared to a value of 0.68 in the original setup. For the worst-performing runs, the opposite is true: the weights of the *environmental* and *comfort* factors have actually increased compared to the original setup.

Table 38: Summarised data analysis of the experimental setup for the municipality of Laren

	Weight_eco	Weight_env	Weight_cof	Weight_soc	Weight sum
Top 5 best runs					
	0.50	0.16	0.46	0.70	1.82
Top 5 worst runs					
	0.22	0.28	0.34	0.72	1.56
Average values of all runs					
	0.46	0.22	0.40	0.73	1.80

6.9 Chapter Summary

This chapter details the results from running 12 different versions of the Python model, using both Random Search and Bayesian Search algorithms for each of the six municipalities. The chapter first provides a municipality-specific analysis, followed by a comparative inter-municipal analysis.

For each municipality, the analysis begins with general demographic information, a Google Trends analysis, and a brief review of news articles concerning solar PVs in that municipality. This sets the stage for a discussion regarding the solar PV adoption curve within each municipality, addressing trends and adoption rates over time. Following this, the results of the inverse modelling process are presented, with a focus on the fitness and the scatter of the results. For the municipality of Bloemendaal, it was found that despite its high income levels, economic factors do not play a predominant role in solar PV adoption. Instead, social factors and community dynamics are more significant, indicating a preference for social utility over financial incentives. For Dantumadiel, the analysis presented a similar trend where social influences are more impactful than economic considerations in driving solar PV adoption. Laren and Oegstgeest also reflect the trend seen in Bloemendaal and Dantumadiel, with social dynamics playing a crucial role in adoption rates. For Vaals, however, the economic incentives are significantly more important than for the other municipalities. Finally, in Westerveld, a notable influence of environmental awareness on adoption decisions can be observed.

The inter-municipal analysis compares the outcomes from each municipality to identify patterns and similarities. It was observed that social factors generally have the most significant impact across all municipalities. The patterns identified in the analysis were, however, not very strong, indicating that further research is needed to assess their reliability. This indicates that IM has strong potential to explore the dynamics of solar PV adoption and that continued research could yield more robust and conclusive insights.

Lastly, an experiment is performed where the model for the municipality of Laren is run with five splits and 30 iterations instead of the two splits and 15 iterations in the original setups. The aim of this experiment is to determine if increasing the number of iterations and splits in the modelling process leads to better results. Key findings show that extended runs yield better results with lower error values, more consistent results, and fewer outliers for both algorithms. Whereas the Random Search algorithm performed significantly better for the original setup, the Bayesian Search now performs often better in the experimental setup. The Random Search algorithm only performs better for the minimum error value of the best run.

7
Research Implications

This chapter discusses the research implications of the results obtained so far. Chapter 7 is an essential part of this research since this subchapter discusses the results, which is crucial for a topic as novel as inverse modelling. Various topics will be discussed, such as the performance dilemma, the results of both the Random Search and the Bayesian Search models, and the model's reliability. Finally, the potential implications of alternative approaches will be discussed.

7.1 Performance Dilemma

The inverse modelling process has provided valuable insights into the dynamics of solar PV adoption in the municipalities. On the other hand, it has also raised dilemmas. Firstly, the data analysis of the municipality of Bloemendaal revealed a discrepancy between the top-performing runs and the average results, posing a dilemma in data interpretation. Top-performing results, for instance, placed much higher weight on the *economic* factor compared to the average results. This dilemma questions whether to focus on the specific strengths of top performers or broader insights from average values. To solve this dilemma, the general objective of inverse modelling should be kept in mind: to uncover explanations for complex phenomena or residential solar PV adoption dynamics in this case. Considering this goal, a more nuanced approach is necessary. Therefore, the more robust results coming from the second approach (to focus on broader insights from average values) seem more appropriate for this research.

However, this dilemma raises a second dilemma similar to the first, but on a different level. Whereas the first dilemma focuses on data interpretation, one can also pose this dilemma on a model level. For instance, when looking at the results of the municipality of Dantumadiel, one can notice how the standard deviation is lower for the Random Search model (namely, 3071 instead of 3754 for the Bayesian Search model). However, the minimum error value is lower for the Bayesian Search model (1534 instead of 1740 for the Random Search model). This would, on the one hand, indicate that the Random Search model performs better due to the lower standard deviation. On the other hand, it could indicate that the Bayesian Search model performs better since it can create predictions closer to reality.

The decision between attaining one highly accurate prediction amidst higher scatter or achieving lower scatter with higher error values is highly dependent on the context of inverse modelling. On the one hand, having one highly accurate prediction amidst higher scatter demonstrates that the model can achieve highly accurate prediction under the right conditions. This can be useful for showcasing the potential of the model and demonstrating that the model can, in principle, capture the underlying dynamics very well. On top of that, the optimal run could serve as a benchmark for future improvements and for understanding the conditions under which the model performs best. However, a higher scatter indicates that the model’s performance is susceptible to parameter variation. This can imply that the model lacks robustness, which is problematic when trying to generalise findings or when the goal is to understand underlying dynamics reliably. Furthermore, suppose the model performs well in only one specific scenario. In that case, it may not be easy to interpret the result as representative of the broader phenomenon, which can limit the model's explanatory power.

A lower scatter with slightly higher error values, on the other hand, suggests that the model performs consistently across different parameter settings. This robustness is valuable for understanding the system's general behaviour and ensuring that the model's insights are reliable. On top of that, a model with consistently good (even if not optimal) performance across various scenarios can provide more generalisable and actionable insights about solar PV adoption. This aligns well with the goal of inverse modelling to uncover underlying explanations. Disadvantages include, however, that consistently higher error values, even if they are only slightly higher, mean that the model is not performing at its best. Furthermore, by focusing on robustness, there might be a risk of not fully exploiting conditions under which the model can perform optimally.

Considering the arguments for each side of the dilemma stated above, prioritising robustness and generalisability is more valuable, given the goal of inverse modelling (to understand and explain complex phenomena like solar PV adoption). By prioritising robustness, inverse modelling aims to capture the fundamental dynamics of the studied system. A model that performs consistently well across different scenarios indicates a deeper understanding of the underlying processes driving the observed phenomena, which is crucial in the inverse modelling process. Generalisability, on the other hand, is essential for extending the insights gained from inverse modelling to new situations or contexts, which is eventually the goal.

7.2 Random Search vs Bayesian Search Performance

When looking at the results of all of the municipalities (see Chapter 6.7), a general trend can be observed for the Random Search models to perform better in the municipalities of Bloemendaal, Dantumadiel, Laren and Westerveld, and Bayesian in Oegstgeest and Vaals in terms of scatter. On the other hand, when looking solely at the minimum error values, both algorithms seem to perform equally well, with Random Search performing better for Bloemendaal, Laren and Vaals, and Bayesian Search performing better for Dantumadiel, Oegstgeest and Westerveld.

This initially seems surprising since Bayesian optimisation should logically perform better than Random Search algorithms because it intelligently explores the parameter space based on past evaluations instead of exploring randomly. This raises the question of *why there is an oscillation in the search quality of the Bayesian Search models.*

The performance of the Bayesian Search algorithm can be explained by the low number of splits (folds) and iterations (2 and 15, respectively) due to computational limitations. Bayesian optimisation relies on prior knowledge or assumptions about the distribution of the objective function. However, with insufficient prior information, which is likely given the low number of splits and iterations, the effectiveness of Bayesian optimisation can be hindered. This limitation restricts the exploration of the parameter space, making it challenging to find the optimal parameter values. In fact, the low amount of splits can hinder finding the optimal parameter values.

This explanation of the performance of the Bayesian Search algorithm can be supported by the results of Chapter 6.8, which performs an experiment where the number of splits and iterations is increased. From this experiment, it became clear that when the number of splits and iterations is increased, the performance of the Bayesian Search model does indeed significantly improve. Both the scatter and the minimum error values decreased significantly, and the Bayesian Search model even started to perform better than the Random Search model in some areas. This experiment, therefore, supports the hypothesis that the inferior performance of the Bayesian Search models is related to the number of splits and iterations.

In conclusion, these findings suggest that the performance difference between the algorithms is not municipality-specific but rather a meta-level discussion that reflects broader methodological challenges. This underscores the importance of considering meta-level factors, such as the number of splits and iterations, when evaluating the algorithms' efficacy.

These results mean that for inverse modelling, Random Search would be a better algorithm for IM performance, considering the presented number of splits and iterations (two and 15, respectively). However, relying on randomly generated parameter configurations is inefficient and costly in the long term. Therefore, an adaptive strategy, such as a hybrid optimisation approach, could be considered. In such an approach, the process would start with a Random Search to explore a broad range of parameter space. Subsequently, leveraging the insights gained from the Random Search phase, a Bayesian Search process could be informed. Using this initial data, Bayesian Search can then make more informed decisions. Alternatively, augmenting the number of splits and iterations becomes viable with sufficient computational resources. This expansion would facilitate broader exploration of the parameter grid, enhancing the optimisation process.

7.3 Conclusiveness of the Inverse Modelling Process

This subchapter delves into the conclusiveness of the inverse modelling results and process within this study, a crucial aspect for accurately explaining the dynamics behind residential solar PV adoption. Conclusiveness in inverse modelling ensures that the findings are definitive, accurately reflecting the real-world phenomena they intend to model.

The entire inverse modelling process begins with the model developed by Muelder & Filatova (2018). This agent-based model (ABM) relies heavily on survey data, which was unavailable due to privacy concerns. Consequently, modifications to the ABM in NetLogo were necessary. Initially, these adjustments pertained only to the geolocation of households, their incomes, and roof sizes. However, over time, additional components of the model were found to be malfunctioning. For example, the function describing the initial amount of solar PV systems in the municipality did not operate correctly. This function was critical for the research and required manual adjustments. Ultimately, numerous changes were necessary to ensure the ABM's proper functionality. This necessity for modifications suggests potential issues with the reliability of the original ABM. It should also be noted that the original ABM had not been peer-reviewed, indicating that peer review could provide additional validation and support for its reliability. Moreover, as discussed in Chapter 2, a certain degree of uncertainty is inherent in ABMs. Despite this, repeated simulations can stabilise the model's randomness, enhancing its conclusiveness by producing more consistent results.

The next part of the inverse modelling process takes place in the Python model. By incrementally building up this model, validating and verifying every component, its clarity and reliability are improved. Although this approach may not yield a perfectly efficient model, it contributes significantly to the overall trustworthiness. Thus, this stage of the inverse modelling process is regarded as conclusive.

The data analysis constitutes the final phase of the IM process. As discussed earlier in this chapter, comprehensive data analysis requires consideration of a wide variety of factors and dilemmas. By focusing on the mean, median, scatter and average values, some of the most critical aspects have been carefully examined, thereby enhancing the reliability of the data analysis process. However, this does not imply that the analysis is complete, as there is a risk that certain relations may have been overlooked. For instance, the patterns identified in the parameter value comparison analysis have been observed visually without any formal correlation analysis. Consequently, some patterns may have been overlooked.

In summary, while the outcomes affirm the existence and viability of inverse modelling, they also underscore limitations within this specific study. The results from the IM process, while indicative, are still inconclusive with regard to the dynamics underlying solar PV adoption in these municipalities. This inconclusiveness can be addressed by the improvement of the original ABM or by running the model with more iterations. This study remains valuable as an exploratory inverse modelling framework for future research, suggesting areas for improvement such as refining modelling techniques, enhancing data analysis methods and exploring alternative approaches as outlined in Chapter 7.5 and Chapter 8.

7.4 Reproducibility and Replicability

Reproducibility and replicability are cornerstone principles in scientific research, serving as the foundation upon which credible and reliable findings are built (National Academies of Sciences, 2019). Reproducibility refers to the capacity for others to duplicate the results of a study using the same methods and data (Boylan, 2016). It ensures that the research process is transparent and verifiable. Replicability, on the other hand, involves obtaining consistent results when the study is conducted with different datasets or slightly altered methodologies, thereby validating the robustness of the conclusions across varying conditions. This section delves into the reproducibility and replicability of this work and the specific application of the NetLogo model by Muelder and Filatova, a crucial component of this study.

In this research, reproducibility is supported by the continuous documentation of every aspect of the research process. The first aspect of reproducibility relates to data availability. All of the data used in this research is drawn from publicly available sources. For instance, the residential solar PV adoption data on a municipal level in the Netherlands comes from CBS, a governmental institution that gathers statistical information about the Netherlands. Data on geolocations, roof sizes, electricity prices, etc., have all been acquired through publicly available sources. All preprocessing steps, from data cleaning and transformation to integration, are thoroughly documented to facilitate replication. This transparency extends to the methodological framework, where each phase of the IM process is elaborated upon, including the selection and calibration of parameters and the choice of ML algorithms for parameter space exploration. These detailed descriptions ensure that other researchers can follow the exact steps taken and achieve the same result.

Software tools and computational codes are another vital aspect of reproducibility in this study. The agent-based model, developed by Muelder & Filatova (2018) in NetLogo, was initially not very reproducible due to its heavy reliance on survey data that was not publicly available. This NetLogo model has, however, been altered so that it now solely relies on publicly available data. To promote reproducibility, the altered NetLogo model, along with the steps that ought to be taken for data implementation, can be found on GitHub and in Appendix A and B. For conciseness reasons, only the code for the municipality of Laren is on GitHub. Additionally, the Python scripts used throughout the research process are provided, offering a clear blueprint of the computational workflow. To further ensure the model's correctness and validity, extensive verification and validation techniques are employed. Cross-validation, sensitivity analysis, and comparisons with empirical data are conducted, reinforcing the credibility of the model's outputs. These steps not only enhance the reproducibility of the research but also strengthen confidence in the reliability and accuracy of the results.

Despite these efforts, certain challenges to reproducibility are recognised. The high computational resources required for running multiple simulations and parameter searches may pose a barrier to replication. The complexity of the modifications applied to the NetLogo model might also complicate replication despite extensive documentation. Furthermore, since the NetLogo model has a strong municipal focus, modelling every single household, it is less suitable for reproducibility on, for instance, a regional or national level.

Replicability is equally crucial, ensuring that the findings of this study hold true for varying conditions and methodologies. Applying the research framework to different geographical areas or time periods would, for instance, test the study's external validity. For instance, using the inverse modelling approach to analyse solar PV adoption in other countries or different periods can help confirm the robustness of the conclusions drawn in this study. The sensitivity of the NetLogo model, especially to different parameter settings, is, however, a crucial aspect of its replicability. The NetLogo model has specific random inputs (e.g. the exact income of a household), which means that no run is precisely the same as the next. This problem of stochasticity is, however, inherent to ABMs, as elaborated in Chapter 2.3. This study attempts to address this issue by doing an as large number of runs of the model as possible and by experimenting with different model configurations.

7.5 Alternative Approaches

As this study aims to establish an exploratory inverse modelling framework, it is important to consider alternative approaches. Given the novelty of the topic of inverse modelling, the methodology outlined in this research is inherently innovative. However, this novelty also implies that the IM process delineated in this study may not represent the optimal approach. Therefore, this section also addresses alternative approaches. Exploring alternative approaches is essential for the question of how inverse modelling can contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities because it allows for refining and improving the IM process. These alternative approaches do not only serve as suggestions for future research but also as valuable suggestions for refining and enhancing the inverse modelling process.

The first alternative approach originates in the dilemma presented in the previous subchapter regarding the reliability of the original ABM. Of course, initially, picking an ABM that is peer-reviewed and has a better setup will improve the trustworthiness of the IM process. However, the possibility still exists that this ABM does not contain the parameters that are actually relevant for the dynamics underlying solar PV adoption in the Netherlands. This presents a strong limitation for inverse modelling. One could consider the alternative approach of including multiple ABMs in the IM process to address this limitation. Including multiple ABMs in the inverse modelling process can offer several advantages. For instance, it could enhance the robustness of the analysis. Comparing the results across different models allows for the identification of more consistent patterns. Compared to using one ABM, the advantage is that it reduces reliance on the assumptions of a single model, which increases the credibility of the findings. On top of that, including multiple ABMs in the IM process allows for cross-validation, where one model can be used to validate the results obtained from another. This also enhances the trustworthiness of the results. Finally, different ABMs might capture different aspects of the system, leading to a more comprehensive understanding of the underlying dynamics, which is essential when considering the goal of inverse modelling. This diversity in modelling could reveal new insights and patterns that a single model might overlook.
Incorporating multiple ABMs might seem like an appealing solution to enhance robustness and cross-validate results. However, in reality it is more of an “easy fix”: while suggesting this approach may appear straightforward, its actual implementation is highly complex. Aligning various models involves significant challenges, such as ensuring that they share consistent assumptions, scales and inputs. Inconsistencies between the ABMs in terms of assumptions, structures and setups can complicate the interpretation of results. Furthermore, models might interpret data differently, leading to inconsistencies in results and increased computational demands. This means that including multiple ABMs comes with significantly increased complexity, which demands more sophisticated expertise and substantial computational resources. If one were to try this alternative approach, a systematic approach for comparing (and contrasting) the different ABMs is essential. However, by carefully managing these disadvantages, incorporating multiple ABMs can significantly enrich the inverse modelling process.

A hybrid step between using a single model and deploying multiple ABMs could be gradually adding complexity. One could start with a simpler base model and progressively add complexity by introducing new modules. This method helps understand the incremental impact of each additional component and ensures manageability. Furthermore, one could think of implementing a modular design, whereby a single ABM is built with a flexible architecture where different modules or components can be swapped in and out. One could, for instance, have separate modules for different adoption behaviours, policy interventions, or market conditions.

The second alternative approach originates from the dilemma presented earlier regarding the performance of the Random Search versus the Bayesian Search model. This dilemma raises the question of what would happen if the focus in the IM process were either *more* or actually *less* on machine learning algorithms. Increasing the focus on machine learning (ML) could, on the one hand, increase the accuracy of the models, especially if better ML techniques, such as neural networks, are used. It could lead to more efficiently optimised model parameters and could uncover additional relationships between various factors that influence adoption rates that were previously overlooked. These factors could enrich the overall understanding of solar PV adoption dynamics. Focussing more strongly on ML can, on the other hand, also limit the exploration of other tools that could be used for the IM process. The potential benefits of other approaches (e.g., qualitative research and statistical modelling) may be overlooked by prioritising ML techniques. Therefore, this narrow focus on ML might result in a less comprehensive understanding of solar PV adoption dynamics. This means that less focus on ML techniques could be beneficial as well. Additionally, an overreliance on ML may lead to challenges with model interpretability and transparency.

When looking at the different strengths and weaknesses of both the Random Search and the Bayesian Search models, one could also apply a hybrid form whereby the strengths of both algorithms are combined. Random Search, on the one hand, is more focused on *exploration*. It samples parameters randomly across the entire search space, which allows it to explore a wide range of possibilities. Bayesian Search, on the other hand, has a stronger emphasis on *exploitation*. It uses a probabilistic model to predict the performance of different parameter sets. It prioritises sampling in areas where it expects to find optimal solutions, thus focusing more on promising regions of the search space. Combining Random Search and Bayesian Search can efficiently optimise parameters by balancing exploration and exploitation. Initially, Random Search could be applied to explore the parameter space broadly. This phase samples a wide range of parameter combinations, providing diverse insights without bias. After a predefined number of iterations, one could transition to Bayesian Search. Using the data from the Random Search, Bayesian Search focuses on exploiting the promising regions identified earlier while occasionally exploring uncertain areas to avoid local optima. This hybrid approach leverages the wide-ranging exploration of Random Search and the focused sampling of Bayesian Search, leading to more efficient and reliable parameter optimisation.

7.6 Chapter Summary

This chapter explores the research implications that can be derived from the results of the IM process. This chapter is crucial as it discusses the noel aspects and potential of IM in understanding the dynamics of solar PV adoption while also addressing limitations and dilemmas encountered during the process.

The first dilemma presented is the “performance dilemma”, highlighting the discrepancies between top-performing runs and the average results. These discrepancies raise a dilemma on whether to prioritise the specific strengths of top-performing runs or broader insights from average values. The goal of IM is to uncover explanations for complex phenomena, which suggests that a focus on broader insights is more appropriate.

Another dilemma discusses the comparison of results between the Random Search and Bayesian Search models. The Random Search models showed lower standard deviations, while the Bayesian Search sometimes achieved lower minimum error values, suggesting it can produce highly accurate predictions under the right conditions. This trade-off between consistency and potential accuracy needs careful consideration. However, given the goal of IM of reliably understanding system dynamics, lower standard deviations (and therefore the Random Search model) appear more appropriate. It is also noted that the seemingly poorer results of the Bayesian Search models are likely the result of the low number of splits and iterations used.

The chapter also addresses the conclusiveness of the results. It concludes that while IM shows potential, the current findings remain inconclusive regarding the dynamics of solar PV adoption in these municipalities. To address this inconclusiveness, improvements to the original ABM or additional model iterations are necessary. The results do indicate that social factors significantly influence solar PV adoption, a valuable insight from the inverse modelling process.

Lastly, alternative approaches to improve the inverse modelling framework are proposed. Incorporating multiple ABMs can enhance the robustness and cross-validate results, though it also introduces complexity and demands more (computational) resources. Furthermore, future IM research could either focus more or less on machine learning, each with its advantages and disadvantages. A hybrid approach could be conducted, using Random Search's broad exploration of the parameter space and Bayesian Search's focused exploitation of promising regions to efficiently optimise model parameters. All in all, these suggestions aim to refine and enhance the IM process for future studies, leading to more reliable and comprehensive insights.

8 Conclusion

This final chapter delves into the practical applications of the research findings, focusing on how inverse modelling can provide deeper insights into residential solar PV adoption dynamics in Dutch municipalities. Chapter 8.1 focuses on answering the main research question, as presented in Chapter 1. After that, the study's scientific and societal relevance is discussed. Finally, the limitations and recommendations for future research are considered.

8.1 Answering the Research Question

The research question guiding this study is:

How can inverse modelling contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities?

To address this question, the contributions of inverse modelling have been researched through the lens of agent-based modelling. Inverse modelling has shown considerable potential in exploring the complex dynamics of solar PV adoption. The methodology allows for estimating model input parameters based on observed outcomes, revealing new dynamics that may have been previously overlooked. The methodology is particularly valuable in the context of solar PV adoption, where many factors influence individual and collective decisions. In this thesis, inverse modelling was applied to six Dutch municipalities, revealing significant, though preliminary, patterns in the adoption dynamics.

The key finding from the inverse modelling application is the substantial influence of social factors on solar PV adoption. Across all studied municipalities, social dynamics often outweighed economic incentives. For instance, in Bloemendaal, where economic factors were expected to be more dominant due to high-income levels, social influences proved to be more critical. This indicates that, based on the ABM used by Muelder & Filatova (2018), familiarity and shared experience with PV systems within social circles play a crucial role in the decision to adopt solar PV systems. The consistently high values of the social parameter also indicate that this parameter value significantly impacts the model's performance. Furthermore, the inverse modelling process has shown that a higher sum of weighted values consistently leads to better fits with the observed data and that the role of economic incentives, while significant, varies strongly in importance.

The comparison of Random Search and Bayesian Search algorithms highlighted a trade-off between consistency and potential accuracy. While the Random Search algorithm showed lower standard deviations, indicating more consistent results, the Bayesian Search sometimes achieved lower minimum error values, suggesting it could produce highly accurate predictions under the right conditions. The discrepancy presents the need for careful consideration in future studies to balance these aspects. Furthermore, it can be concluded that the reliability of the findings can be improved by addressing data constraints and reducing the reliance on a single ABM that requires extensive modifications in future research. Despite these challenges, this thesis demonstrates that inverse modelling is a viable approach with significant potential for understanding solar PV adoption.

So, to answer the research question, inverse modelling can contribute to uncovering plausible explanations for residential solar PV adoption dynamics in Dutch municipalities by providing a systematic approach to understanding the underlying factors of the system and their intentions. Through IM, the key determinants that drive solar PV adoption can be inferred by analysing observed patterns and behaviours within the municipalities. This approach is additionally powerful because it does not rely solely on pre-existing theories or assumptions, as with forward modelling. Instead, it derives insights directly from the data.

Through IM, the influence of model parameters could be revealed (in this case, mainly the influence of social factors over economic incentives, highlighting the importance of community engagement and peer effects). It allows for adjusting model parameters to match observed outcomes, thereby identifying the interplay between various factors (economic, social, environmental and comfort factors). Patterns between demographic data and these factors can be identified as well, though the patterns found in this research are not conclusive as a result of some challenges in the initial ABM. This indicates that inverse modelling has a strong potential to uncover plausible explanations for solar PV adoption, especially when the challenges mentioned in this work are addressed in future research.

8.2 Scientific Relevance

Though this thesis's scientific relevance is mostly methodological, it is pivotal in demonstrating the potential of inverse modelling for understanding residential solar PV adoption dynamics in Dutch municipalities. The relevance revolves around the application of IM, the comparison of search algorithms, and the critical assessment of model reliability.

Firstly, inverse modelling has been innovatively applied to the socio-economic context of solar PV adoption. This methodology has so far not yet been applied. This thesis advances scientific methods within the social sciences domain by applying inverse modelling. By exploring a methodology that has not yet been employed, the gap identified in the literature review is (partially) filled, and the groundwork for future research on this topic has been laid.

This thesis also contributes methodologically by comparing the effectiveness of Random Search and Bayesian Search algorithms in inverse modelling. The findings suggest that while Bayesian Search can achieve higher accuracy under optimal conditions, Random Search provides more consistent results, which is of higher importance considering the purpose of IM. The comparison of the algorithms demonstrates their potential in refining models to reflect real-world data better and provides a methodological basis for future research. This groundwork allows for further refinement and improvement of the application of these algorithms. In an initial experiment for the municipality of Laren, it was already shown that increasing the number of splits for the Bayesian Search algorithm significantly improves its results. Therefore, refinement and improvement of the application of these algorithms could focus on further exploration of increasing the number of splits and iterations. The insights gained from this comparison can guide future research on the topic in selecting and applying appropriate algorithms.

Finally, this work critically addresses the limitations associated with data availability and model reliability. By acknowledging the constraints due to limited data points and the original ABM's lack of reliability, this thesis underscores the necessity for more high-quality data and robust models for future research. This transparency in the limitations is scientifically very relevant as it highlights areas for improvement and guides future research efforts towards IM.

Importantly, the reproducibility and replicability of this study are addressed as well. Reproducibility in this work is ensured through comprehensive documentation of the research process, data availability from publicly accessible sources, and sharing of computational codes and model adjustments on GitHub. This transparency supports the reliability of findings and facilitates future replication in scientific work.

In conclusion, this study's methodological contributions are substantial. This study can serve as an initial exploratory framework for future inverse modelling research, improving the substantive contributions of future IM research.

8.3 Societal Relevance

For the assessment of the societal relevance of this work, the assessment criteria presented by Bornmann (2013) are used. These criteria are: social impact, cultural impact, environmental impact and economic impact.

The first criterion refers to the *social impacts* of the work by contributing to challenges that address social issues that influence policymaking or enhance public debates. The social impact of this thesis can be found in its ability to address social issues, influence policymaking and enhance public debates regarding residential solar PV adoption. By employing inverse modelling, this thesis sheds light on the underlying social dynamics that drive solar PV adoption, such as the influence of community engagement and peer effects (the social parameter) or the influence of financial considerations (the economic parameter). These insights can inform policymakers about the importance of each of these factors in promoting solar PV adoption, potentially leading to more effective and socially inclusive energy policies. Moreover, this thesis' findings on the importance of social dynamics in the decision-making process can inspire a more community-driven approach towards solar PV adoption, contributing to the social impact.

The second criterion refers to how the research enhances *cultural capital.* The cultural impact of this research lies in its potential to improve cultural capital by creating a greater understanding of the social and cultural dynamics related to solar PV adoption. By revealing how different factors play a role in the decision-making process of solar panels on a residential level, inverse modelling can contribute to broader cultural awareness of solar PV adoption. By highlighting the significance of municipal contexts, this research also promotes cultural preservation by valuing differences in the energy transition. Furthermore, this research relates to households in the Netherlands, which is quite advanced in residential solar PV adoption. Insights into the dynamics underlying this system in the Netherlands can contribute to the transition in countries with less advanced solar PV adoption dynamics.

The *environmental impact* of this thesis lies in its aim to provide insights that can drive increased adoption of residential solar PV systems, thereby contributing to the energy transition. Of course, the research results alone are not reliable enough to make this contribution. However, this research lays the foundation for future IM research to create more robust insights. By understanding the factors influencing solar PV adoption, policymakers and stakeholders can develop more targeted strategies to promote solar PV adoption.

Finally, the *economic impact* criterium refers to how the work contributes to a country's economic capital, encompassing cost-related and value-creating aspects. This impact is multifaceted for this thesis. By further uncovering the dynamics behind residential solar PV adoption, IM can provide valuable information to help optimise incentive programs and financial support mechanisms for solar PV, making the transition more cost-effective.

8.4 Limitations of the Study

The results of this research are subject to limitations. The first limitation is model dependence. This research relies heavily on *one* agent-based model, namely the ABM by Muelder & Filatova (2018). The need for extensive modifications to this ABM due to unavailable survey data and some

malfunctioning components raises concerns about the reliability of the original ABM. Additionally, this original ABM was not peer-reviewed, diminishing its trustworthiness.
This first limitation also relates to a second limitation concerning the modelling approach applied in this research. While the modelling approach provides many advantages, it also inherently contains certain imperfections due to their inherent imperfection. This fundamental limitation means that the results will always be approximate and may not accurately reflect real-world dynamics.
Another limitation is related to computational constraints. A balance had to be made between computational ability and the feasibility of the model results for the number of splits, iterations, and runs. In the end, it was decided to do two splits, 15 iterations, and 25 runs. However, these constraints limit the Bayesian Search models significantly. This restriction affects the exploration of the parameter space and, therefore, the effectiveness of the optimisation process. Chapter 6.8's experiment, which focused on increasing splits and iterations for a single municipality, proves this effectiveness and underscores the significance of this limitation. On top of that, a limitation of this study lies in the data. The study faced significant data constraints, particularly the availability of training data. The limited number of data points (11 points) reduces the statistical power of the findings since IM relies heavily on high-quality and sufficient data. Insufficient data can lead to overfitting and unreliable parameter estimates, which impact the ability of the model to accurately represent the underlying system.

Moreover, model incompleteness presents another challenge. Despite careful analysis, there is a risk that some relations were overlooked due to the lack of a formal correlation analysis. This could mean that certain dynamics underlying solar PV adoption remain unexplored.
Furthermore, a limitation arises from the integration of machine learning. As mentioned in Chapter 7, ML can enhance model accuracy, but it may also complicate model interpretability and transparency. A seventh limitation can be found in the exclusion of parameters for simplicity. In the final IM process, only four different parameters were explored, even though more parameters were present in the ABM. The advantage of doing so is that it simplifies the optimisation process, reduces computational complexity and enhances interpretability since the focus is only on key parameters. However, it also risks oversimplification and overlooking interactions among fixed parameters that could influence model behaviour. This constraint leads to the final limitation of the study, namely the assumptions in the ABM concerning parameter values. The parameters not included in the IM process have been given a standard value. This also risks oversimplification and can make results less reliable.

8.5 Future Directions and Applications of Inverse Modelling Research

As inverse modelling emerges as a novel approach to understanding and predicting solar PV adoption, identifying the right next steps is crucial for advancing its methodological robustness and expanding its practical applications. This section aims to outline the future directions and practical uses of inverse modelling using the experience obtained during this work, alongside some final remarks about using inverse modelling as a toolkit.

8.5.1. Methodological Advancements

In order to draw strong implications, some methodological advancements can be made, as emerged from this research, to optimise the inverse modelling framework. As mentioned in Chapter 7.4, the methodology applied in this research is novel and experimental, indicating that improvements can be made. Chapter 7.4 presented some alternative approaches. These approaches can also be seen as recommendations for future research or potential methodological advancements for the inverse modelling framework. This section briefly presents different areas with potential for these methodological advancements.

Hyperparameters
Exploring the inverse modelling framework could, for instance, include experimentation with hyperparameters, such as the learning rate or the world size. The alternative approach from Chapter 7.5, relating to the degree of focus on ML, is an example of this. Placing either a larger or a smaller emphasis on machine learning impacts model accuracy and understanding, highlighting the trade-

offs between enhanced optimisation and potential limitations in comprehensiveness and interpretability.

Furthermore, combining Random Search and Bayesian Search offers a hybrid approach that optimises parameters by balancing broad *exploration* with focused *exploitation.* This method would begin with a Random Search to explore the parameter space widely and then transition to a Bayesian Search, leveraging data to exploit promising regions, thereby enhancing efficiency and reliability in parameter optimisation.

Structural Optimisation:
Moreover, exploring the inverse modelling framework could also include structural optimisation, which focuses on refining the model's architecture or structure to improve its performance. Structural optimisation can be applied to both the original ABM and the Python model. One could, for instance, consider another type of implementation of the Theory of Planned Behaviour in the original ABM (e.g. MF or SE, as elaborated by Muelder & Filatova (2018)), consider different sequences of factors, or try different model architectures.

Furthermore, integrating multiple agent-based models could significantly improve the robustness and generalisability of IM results. However, as described in Chapter 7.5, it also introduces significant complexities, such as aligning consistent assumptions, scales and inputs, which require expertise and substantial computational resources. Therefore, exploring hybrid approaches that combine different modelling techniques may also yield valuable insights.

Computational Resources
Another opportunity for future research is to optimise computational resources. Augmenting the number of splits and iterations, especially for the Bayesian Search models, would facilitate a broader exploration of the parameter grid, enhancing optimisation and accuracy. Essentially, this is part of exploring the hyperparameters. However, since it appeared to have such an impactful implication in this study, it is discussed in this separate section.

The effectiveness of this opportunity is shown in Chapter 6.8, where the number of splits and iterations is increased for the municipality of Laren. This experiment presented significant improvement in the performance of both algorithms. Another recommendation lies in the expansion of training data. Increasing the training data is crucial for strengthening the robustness of the findings. More data would lead to better validation and more reliable conclusions. For this research, limited training data was available due to the scope (municipal) of the study and the fact that residential solar PV adoption only started mainly happening in the last 12 years.

Parameters:
Increasing the number of parameters in the inverse modelling process (currently four) can significantly enhance the model’s ability to capture complex dynamics and provide more nuanced insights into solar PV adoption patterns. This expansion allows for a more detailed representation of influencing factors. However, it also introduces pitfalls, such as increased computational demand, the potential for parameter redundancy and multicollinearity. Balancing parameter complexity and model performance is, therefore, crucial.

In the image below, the graph (Figure 5, see Chapter 2.4) of the relationships between the different concepts is revisited. This time, areas for potential methodological advancements, as outlined in this section, are highlighted in red. These highlights indicate where these improvements can be implemented in the inverse modelling process.

Structural optimisation is applicable to both the Agent-Based Model and the Python model. For the Python model, adjustments to the structural architecture can directly impact the hyperparameters, which in turn affect the model’s performance. Exploring these hyperparameters is crucial, as it allows for further methodological advancements that can refine the model’s accuracy and efficiency.

Additionally, enhancements in computational resources can significantly influence the machine learning components, facilitating more robust and comprehensive model optimisation.

It is important to note that the red-highlighted areas in this figure represent only the specific areas discussed in this chapter. Methodological advancements, however, are not limited to these regions alone; they can arguably be applied to every step in the IM process. For instance, one might opt to change the machine learning algorithm used in the IM framework or substantially increase the number of data points for model validation. Such modifications can profoundly enhance the model's robustness and predictive capability.

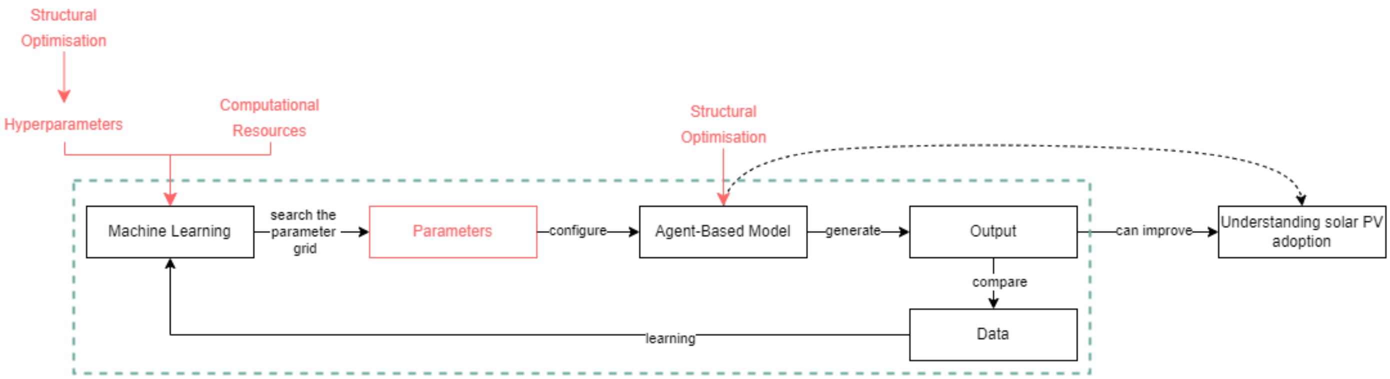


Figure 37: Overview of the relations between concepts in the inverse modelling process. Areas with potential for methodological advancements, as outlined in this chapter, are highlighted in red.

8.5.2. Inverse Modelling as a Toolkit

When inverse modelling, it is crucial to maintain perspective and avoid getting lost in the process itself. Therefore, this section briefly outlines key considerations.

Inverse modelling is a powerful methodology designed to delve deeply into the underlying mechanisms driving observed phenomena, such as the adoption of solar PV systems in this research. It seeks to understand the driving forces and dynamics shaping solar PV adoption. This means that solely optimising model parameters for prediction accuracy is not the end goal of the inverse modelling process; *understanding* is. Inverse modelling is, therefore, a tool that can help social scientists explain social phenomena. It provides a structured approach to understanding complexities and going beyond prediction. Most existing ABMs do forward modelling to produce explanations for certain phenomena involving the prediction of the outcome of a system based on known inputs or parameters. This process does, however, not explain the system or phenomenon. IM tries to address this challenge. It is important to note, though, that inverse modelling does not try to replace forward modelling. They can simply be seen as different methodologies.
Several requirements must be met to effectively use inverse modelling. The model needs to replicate a well-defined outcome, a preliminary model structure representing the system being studied, and sufficient computational resources to handle the numerous simulations and adjustments required. Effective optimisation algorithms are also essential for exploring the parameter space.

While social scientists are the primary audience of inverse modelling, this approach also appeals to other disciplines. Environmental scientists, for instance, could benefit from IM as it enables the exploration of complex interactions between human activities and environmental outcomes, such as carbon emissions. Interdisciplinary collaboration could, therefore, also be an interesting recommendation for future IM research. It could bring more diverse perspectives, enriching the research findings and creating a more profound understanding of, in this case, solar PV adoption dynamics.

Bibliography

Akhatova, A., Kranzl, L., Schipfer, F., & Heendeniya, C. B. (2022). Agent-Based Modelling of Urban District Energy System Decarbonisation—A Systematic Literature Review. *Energies*, *15*(2), 554. https://doi.org/10.3390/en15020554

Ale Ebrahim Dehkordi, M., Lechner, J., Ghorbani, A., Nikolic, I., Chappin, É., & Herder, P. (2023). Using Machine Learning for Agent Specifications in Agent-Based Models and Simulations: A Critical Review and Guidelines. *Journal of Artificial Societies and Social Simulation*, *26*(1). https://doi.org/10.18564/jasss.5016

AlleCijfers. (2024a). *Statistieken gemeente Bloemendaal*. https://allecijfers.nl/gemeente/bloemendaal/

AlleCijfers. (2024b). *Statistieken gemeente Dantumadiel*. https://allecijfers.nl/gemeente/dantumadiel/

AlleCijfers. (2024c). *Statistieken gemeente Laren*. https://allecijfers.nl/gemeente/laren/

AlleCijfers. (2024d). *Statistieken gemeente Oegstgeest*. https://allecijfers.nl/gemeente/oegstgeest/

AlleCijfers. (2024e). *Statistieken gemeente Vaals*. https://allecijfers.nl/gemeente/vaals/

AlleCijfers. (2024f). *Statistieken gemeente Westerveld*. https://allecijfers.nl/gemeente/westerveld/#:~:text=Er wonen 8.892 huishoudens in,18 wijken en 45 buurten.

Alpaydin, E. (2010). *Introduction to Machine Learning* (2nd ed.). Massachusetts Institute of Technology.

An, L., Grimm, V., Bai, Y., Sullivan, A., Turner, B. L., Malleson, N., Heppenstall, A., Vincenot, C., Robinson, D., Ye, X., Liu, J., Lindkvist, E., & Tang, W. (2023). Modeling agent decision and behavior in the light of data science and artificial intelligence. *Environmental Modelling & Software*, *166*, 105713. https://doi.org/10.1016/j.envsoft.2023.105713

Andradóttir, S. (2006). An Overview of Simulation Optimization via Random Search. In *Handbooks in Operations Research and Management Science* (Vol. 13, pp. 617–631). North Holland.

Ardila, L., Franco, C. J., Cadavid, L., & Torres, J. P. (2022). Modeling the technological adoption of solar energy neighborhoods: The case of Chile. *Journal of Cleaner Production*, *363*, 132620. https://doi.org/10.1016/j.jclepro.2022.132620

Ballas, D., Liu, X., & Zhang, J. (2023). Neighbourhood-level spatial determinants of residential solar photovoltaic adoption in the Netherlands. *Renewable Energy*, *206*, 1239–1248. https://doi.org/10.1016/j.renene.2023.02.118

Bashardoust, A., Safaei, N., Haki, K., & Shrestha, Y. R. (2023). Employing Machine Learning to Advance Agent-based Modeling in Information Systems Research. *ICIS 2023 Proceedings. 3.*

Belfrage, M., Lorig, F., & Davidsson, P. (2022). Making Sense of Collaborative Challenges in Agent-based Modelling for Policy-Making. *CEUR: AMPM 2022, 2nd Workshop in Agent-Based Modeling & Policy-Making*.

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, *13*(1), 281–305.

Berndt, D. J., & Clifford, J. (1994). Using Dynamic Time Warping to Find Patterns in Time Series. *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining (KDD-94)* , 359–370.

Besagni, G., Premoli Vilà, L., Borgarello, M., Trabucchi, A., Merlo, M., Rodeschini, J., & Finazzi, F. (2021). Electrification pathways of the Italian residential sector under socio-demographic constrains: Looking towards 2040. *Energy*, *217*, 119438. https://doi.org/10.1016/j.energy.2020.119438

Bi, Q., Goodman, K. E., Kaminsky, J., & Lessler, J. (2019). What is Machine Learning? A Primer for the Epidemiologist. *American Journal of Epidemiology*. https://doi.org/10.1093/aje/kwz189

Bogner, K., Müller, M., Pyka, A., Ebersberger, B., Berger, T., & Dahlke, J. (2020). *Is The Juice Worth The Squeeze? Machine Learning In And For Agent-Based Modelling*. https://doi.org/10.13140/RG.2.2.16497.02403

Bornmann, L. (2013). What is societal impact of research and how can it be assessed? a literature

survey. *Journal of the American Society for Information Science and Technology*, *64*(2), 217–233. https://doi.org/10.1002/asi.22803
Borshchev, A., & Filippov, A. (2004). From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. *Proceedings of the 22nd International Converence of the System Dynamics Society*, 25–29.
Box, G. E. P. (1976). Science and Statistics. *Journal of the American Statistical Association*, *71*(356), 791. https://doi.org/10.2307/2286841
Boylan, J. E. (2016). Reproducibility. *IMA Journal of Management Mathematics*, *27*(2), 107–108. https://doi.org/10.1093/imaman/dpw003
CBS. (2012). *Aardgas en elektriciteit; gemiddelde tarieven 1996-2012*. https://www.cbs.nl/nl-nl/cijfers/detail/37359
CBS. (2020). *Zonnestroom; vermogen bedrijven en woningen, regio(indeling 2018),2012-2018*. https://opendata.cbs.nl/#/CBS/nl/dataset/84518NED/table?ts=1710165208033
CBS. (2021). *Bevolkingsontwikkeling; Regio per maand*. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/37230ned/table?ts=1614531439313
CBS. (2023a). *Inkomen van huishoudens; huishoudenskenmerken; regio (indeling 2023)*. https://opendata.cbs.nl/statline#/CBS/nl/dataset/85711NED/table?ts=1708688624270
CBS. (2023b). *Inwoners per gemeente*. https://www.cbs.nl/nl-nl/visualisaties/dashboard-bevolking/regionaal/inwoners
CBS. (2023c). *Vermogen van huishoudens opnieuw gestegen*. https://www.cbs.nl/nl-nl/nieuws/2023/44/vermogen-van-huishoudens-opnieuw-gestegen#:~:text=De Nederlandse gemeente met het,meest vermogende gemeente van Nederland
CBS. (2023d). *Zonnestroom; vermogen en vermogensklasse, bedrijven en woningen, regio*. https://opendata.cbs.nl/#/CBS/nl/dataset/85005NED/table?ts=1710165524465
CBS. (2023e). *Zonnestroom; vermogen en vermogensklasse, bedrijven en woningen, regio*. https://opendata.cbs.nl/#/CBS/nl/dataset/85005NED/table
Chawla, N. V., & Karakoulas, G. (2005). Learning From Labeled And Unlabeled Data: An Empirical Study Across Techniques And Domains. *Journal of Artificial Intelligence Research*, *23*, 331–366. https://doi.org/10.1613/jair.1509
CoMSES. (2024). *Computational Model Library*. https://www.comses.net/codebases/?query=solar
Cooley, P., & Solano, E. (2011). Agent-Based Model validation considerations. *Proceedings of the 3rd International Conference on Adances in System Simulation*, 134–139.
Creutzig, F., Agoston, P., Goldschmidt, J. C., Luderer, G., Nemet, G., & Pietzcker, R. C. (2017). The underestimated potential of solar energy to mitigate climate change. *Nature Energy*, *2*(9), 17140. https://doi.org/10.1038/nenergy.2017.140
Crooks, A., Castle, C., & Batty, M. (2008). Key challenges in agent-based modelling for geo-spatial simulation. *Computers, Environment and Urban Systems*, *32*(6), 417–430. https://doi.org/10.1016/j.compenvurbsys.2008.09.004
Crooks, A. T., & Heppenstall, A. J. (2012). Introduction to Agent-Based Modelling. In *Agent-Based Models of Geographical Systems* (pp. 85–105). Springer Netherlands. https://doi.org/10.1007/978-90-481-8927-4_5
Crowe, S., Cresswell, K., Robertson, A., Huby, G., Avery, A., & Sheikh, A. (2011). The case study approach. *BMC Medical Research Methodology*, *11*(1), 100. https://doi.org/10.1186/1471-2288-11-100
Darvishi, M., & Ahmadi, G. (2014). Validation techniques of agent based modelling for geospatial simulations. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XL-2/W3*, 91–95. https://doi.org/10.5194/isprsarchives-XL-2-W3-91-2014
Díaz-Domínguez, A. (2020). How Futures Studies and Foresight Could Address Ethical Dilemmas of Machine Learning and Artificial Intelligence. *World Futures Review*, *12*(2), 169–180. https://doi.org/10.1177/1946756719894602
Eisenhardt, K. M. (1989). Building Theories from Case Study Research. *The Academy of Management Review*, *14*(4), 532. https://doi.org/10.2307/258557
Epstein, J. M. (1999). Agent-based computational models and generative social science. *Complexity*, *4*(5), 41–60. https://doi.org/https://doi.org/10.1002/(SICI)1099-0526(199905/06)4:5%3C41::AID-CPLX9%3E3.0.CO;2-F

Epstein, J. M. (2012). *Generative Social Science: Studies in agent-based computational modeling*. University Press Princeton.
Epstein, J. M. (2019). Inverse Generative Social Science: What Machine Learning can do for Agent-Based Modeling. In *Social-Behavioral Modeling for Complex Systems*. Wiley.
Epstein, J. M. (2023). Inverse Generative Social Science: Backward to the Future. *Journal of Artificial Societies and Social Simulation*, *26*(2). https://doi.org/10.18564/jasss.5083
European Commission. (2023). *National Energy and Climate Plans*. https://commission.europa.eu/energy-climate-change-environment/implementation-eu-countries/energy-and-climate-governance-and-reporting/national-energy-and-climate-plans_en
Flach, P. A. (2012). *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press.
Flyvbjerg, B. (2011). Case Study. In *The Sage Handbook of Qualitative Research* (4th ed., pp. 301–316). Sage.
Frank, A., Bartels, E., Grissom, A. R., Blake, J. S., Tarini, G., Eusebi, K. E., Mait, J., Bienenstock, E. J., Parker, A. W., Geist, E., Davis, P. K., Popper, S. W., Connable, B., WInkelman, Z., Axtell, R. L., Grana, J., Lempert, R. J., Klima, K., Turner, S., … Marler, R. T. (2022). *Adaptive Engagement for Undergoverned Spaces: Concepts, Challenges, and Prospects for New Approaches: Chapter 16*. RAND Corporation. https://doi.org/10.7249/RRA1275-1
Ghazali M., A., & Abdul Rahman, A. M. (2012). The Performance of Three Different Solar Panels for Solar Electricity Applying Solar Tracking Device under the Malaysian Climate Condition. *Energy and Environment Research*, *2*(1). https://doi.org/10.5539/eer.v2n1p235
Government of the Netherlands. (2004). *Kamerstuk*. https://zoek.officielebekendmakingen.nl/kst-29372-45.html
Government of the Netherlands. (2023a). *Climate Policy*. https://www.government.nl/topics/climate-change/climate-policy
Government of the Netherlands. (2023b). *Overheid bevordert groei zonne-energie*. https://www.rijksoverheid.nl/onderwerpen/duurzame-energie/zonne-energie#:~:text=Deze regelingen zijn nodig om,dan 11%2C5 miljoen huishoudens.
Government of the Netherlands. (2023c). *Ruim € 1 miljard voor uitvoering klimaatbeleid gemeenten en provincies*. https://www.rijksoverheid.nl/actueel/nieuws/2023/02/27/ruim-€1-miljard-voor-uitvoering-klimaatbeleid-gemeenten-en-provincies
Greig, R., Major, C., Pacholska, M., Bending, S., & Arranz, J. (2023). Learning Interpretable Logic for Agent-Based Models from Domain Independent Primitives. *Journal of Artificial Societies and Social Simulation*, *26*(2). https://doi.org/10.18564/jasss.5087
Gupta, J., Termeer, K., van den Brink, M., & Klostermann, M. (2008). *Institutions for adaptation: the capacity and ability of the Dutch institutional framework to adapt to climate change*.
Hammond, R. A. (2015). *Assessing the Use of Agent-Based Models for Tobacco Regulation* (R. Wallace, A. Geller, & V. Ayano (eds.)). Washington (DC) National Academies Press.
Harder, N., Qussous, R., & Weidlich, A. (2023). Fit for purpose: Modeling wholesale electricity markets realistically with multi-agent deep reinforcement learning. *Energy and AI*, *14*, 100295. https://doi.org/10.1016/j.egyai.2023.100295
Hitchcock, G. (1993). An integrated framework for energy use and behaviour in the domestic sector. *Energy and Buildings*, *20*(2), 151–157. https://doi.org/10.1016/0378-7788(93)90006-G
Holland, J. H. (2006). Studying Complex Adaptive Systems. *Journal of Systems Science and Complexity*, *19*(1), 1–8. https://doi.org/10.1007/s11424-006-0001-z
Huynh, D. C., Nguyen, T. A. T., Dunnigan, M. W., & Mueller, M. A. (2013). Maximum power point tracking of solar photovoltaic panels using advanced perturbation and observation algorithm. *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, 864–869. https://doi.org/10.1109/ICIEA.2013.6566488
IEA. (2020). *Evolution of solar PV module cost by data source, 1970-2020*. https://www.iea.org/data-and-statistics/charts/evolution-of-solar-pv-module-cost-by-data-source-1970-2020
IRENA. (2021). *Renewable Power Generation Costs in 2020*.
Jägemann, C., Fürsch, M., Hagspiel, S., & Nagl, S. (2013). Decarbonizing Europe’s power sector by 2050 — Analyzing the economic implications of alternative decarbonization pathways. *Energy Economics*, *40*, 622–636. https://doi.org/10.1016/j.eneco.2013.08.019

Jaxa-Rozen, M., & Kwakkel, J. H. (2018). PyNetLogo: Linking NetLogo with Python. *Journal of Artificial Societies and Social Simulation*, *21*(2). https://doi.org/10.18564/jasss.3668

Jennings, N. R. (2000). On agent-based software engineering. *Artificial Intelligence*, *117*(2), 277–296. https://doi.org/10.1016/S0004-3702(99)00107-1

KadastraleKaart. (2024). *Kadastrale Kaart*. https://kadastralekaart.com/gemeenten/vaals-GM0981

Klimaatmonitor. (2021). *Klimaatmonitor*. https://klimaatmonitor.databank.nl/

Klügl, F. (2009). *Agent-Based Simulation Engineering*. University of Würzburg.

Kraaijvanger, C. W., Verma, T., Doorn, N., & Goncalves, J. E. (2023). Does the sun shine for all? Revealing socio-spatial inequalities in the transition to solar energy in The Hague, The Netherlands. *Energy Research & Social Science*, *104*, 103245. https://doi.org/10.1016/j.erss.2023.103245

Law, A. M., & Kelton, D. M. (1991). *Simulation Modeling and Analysis* (1st ed., Vol. 1). McGraw-Hill Inc. .

Liu, Z., & Zhang, Z. (2016). Solar forecasting by K-Nearest Neighbors method with weather classification and physical model. *2016 North American Power Symposium (NAPS)*, 1–6. https://doi.org/10.1109/NAPS.2016.7747859

Londo, M., Matton, R., Usmani, O., van Klaveren, M., Tigchelaar, C., & Brunsting, S. (2020). Alternatives for current net metering policy for solar PV in the Netherlands: A comparison of impacts on business case and purchasing behaviour of private homeowners, and on governmental costs. *Renewable Energy*, *147*, 903–915. https://doi.org/10.1016/j.renene.2019.09.062

Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, *10*(2), 144–156. https://doi.org/10.1057/jos.2016.7

Macal, C. M., & North, M. J. (2009). Agent-based modeling and simulation. *Proceedings of the 2009 Winter Simulation Conference (WSC)*, 86–98. https://doi.org/10.1109/WSC.2009.5429318

Marmolejo, J. A., Velasco, J., & Selley, H. J. (2017). An adaptive random search for short term generation scheduling with network constraints. *Plos One*, *12*(2). https://doi.org/10.1371/journal.pone.0172459

Miller, S. (2019). Machine Learning, Ethics and Law. *Australasian Journal of Information Systems*, *23*. https://doi.org/10.3127/ajis.v23i0.1893

Moglia, M., Nygaard, C. A., Glackin, S., Cook, S., & Tapsuwan, S. (2022). Hybrid Approach for Modelling the Uptake of Residential Solar PV Systems, with Case Study Application in Melbourne, Australia. *Journal of Artificial Societies and Social Simulation*, *25*(4). https://doi.org/10.18564/jasss.4921

Mohri, M., Rostamizadeh, A., & Ameet, T. (2018). *Foundations of Machine Learning* (2nd ed.). The MIT Press.

Morales, E. F., & Escalante, H. J. (2022). A brief introduction to supervised, unsupervised, and reinforcement learning. In *Biosignal Processing and Classification Using Computational Learning and Intelligence* (pp. 111–129). Elsevier. https://doi.org/10.1016/B978-0-12-820125-1.00017-8

Mortazavi, A. (2022). Interactive fuzzy Bayesian search algorithm: A new reinforced swarm intelligence tested on engineering and mathematical optimization problems. *Expert Systems with Applications*, *187*, 115954. https://doi.org/10.1016/j.eswa.2021.115954

Muelder, H., & Filatova, T. (2018). One Theory - Many Formalizations: Testing Different Code Implementations of the Theory of Planned Behaviour in Energy Agent-Based Models. *Journal of Artificial Societies and Social Simulation*, *21*(4). https://doi.org/10.18564/jasss.3855

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.

Nan. (2011). Capturing Bottom-Up Information Technology Use Processes: A Complex Adaptive Systems Model. *MIS Quarterly*, *35*(2), 505. https://doi.org/10.2307/23044054

National Academies of Sciences, E. and M. (2019). *Reproducibility and Replicability in Science*. National Academies Press. https://doi.org/10.17226/25303

Naumann-Woleske, K. (2021, June 18). *Introducing Inverse Generative Social Science*.

Nikolic, I., Dam, K. H., & Lukszo, Z. (2013a). *Agent-Based Modelling of Socio-Technical Systems* (Chapter 2.5, Vol. 9). Springer Netherlands. https://doi.org/10.1007/978-94-007-4933-7

Nikolic, I., Dam, K. H., & Lukszo, Z. (2013b). *Agent-Based Modelling of Socio-Technical Systems*.

Springer Netherlands. https://doi.org/10.1007/978-94-007-4933-7
North, M. J., & Macal, C. M. (2007). *Managing business complexity: discovering strategic solutions with agent-based modeling and simulation*. Oxford University Press.
Nugroho, S., & Uehara, T. (2023). Systematic Review of Agent-Based and System Dynamics Models for Social-Ecological System Case Studies. *Systems*, *11*(11), 530. https://doi.org/10.3390/systems11110530
Nuñez-Jimenez, A., Mehta, P., & Griego, D. (2023). Let it grow: How community solar policy can increase PV adoption in cities. *Energy Policy*, *175*, 113477. https://doi.org/10.1016/j.enpol.2023.113477
Osseweijer, F. J. W., van den Hurk, L. B. P., Teunissen, E. J. H. M., & van Sark, W. G. J. H. M. (2018). A comparative review of building integrated photovoltaics ecosystems in selected European countries. *Renewable and Sustainable Energy Reviews*, *90*, 1027–1040. https://doi.org/10.1016/j.rser.2018.03.001
Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., … Moher, D. (2021). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, n71. https://doi.org/10.1136/bmj.n71
Pierie, F., van Someren, C. E. J., Kruse, S. N. M., Laugs, G. A. H., Benders, R. M. J., & Moll, H. C. (2021). Local Balancing of the Electricity Grid in a Renewable Municipality; Analyzing the Effectiveness and Cost of Decentralized Load Balancing Looking at Multiple Combinations of Technologies. *Energies*, *14*(16), 4926. https://doi.org/10.3390/en14164926
Ragin, C. C. (2014). *The Comparative Method: Moving Beyond Qualitative and Quantitative Strategies (With a New Introduction)* (1st ed., Vol. 1). University of California Press.
Rai, V., & McAndrews, K. (2012). Decision-making and behaviour change in residential adopters of solar pv. *Proc. of the World Renewable Energy Forum.*
Rai, V., Reeves, D. C., & Margolis, R. (2016). Overcoming barriers and uncertainties in the adoption of residential solar PV. *Renewable Energy*, *89*, 498–505. https://doi.org/10.1016/j.renene.2015.11.080
Rai, V., & Robinson, S. A. (2015). Agent-based modeling of energy technology adoption: Empirical integration of social, behavioral, economic, and environmental factors. *Environmental Modelling & Software*, *70*, 163–177. https://doi.org/10.1016/j.envsoft.2015.04.014
Rebala, G., Ravi, A., & Churiwala, S. (2019). Machine Learning Definition and Basics. In *An Introduction to Machine Learning* (pp. 1–17). Springer International Publishing. https://doi.org/10.1007/978-3-030-15729-6_1
Renzini, F., Debernardi, C., Bianchi, F., Cremonini, M., & Squazzoni, F. (2023). *The New Frontiers of Social Simulation in the Data Science Era: An Introduction to the Proceedings* (pp. 1–10). https://doi.org/10.1007/978-3-031-34920-1_1
Robinson, D. T., Brown, D. G., Parker, D. C., Schreinemachers, P., Janssen, M. A., Huigen, M., Wittmer, H., Gotts, N., Promburom, P., Irwin, E., Berger, T., Gatzweiler, F., & Barnaud, C. (2007). Comparison of empirical methods for building agent-based models in land use science. *Journal of Land Use Science*, *2*(1), 31–55. https://doi.org/10.1080/17474230701201349
Robinson, S. A., & Rai, V. (2015). Determinants of spatio-temporal patterns of energy technology adoption: An agent-based modeling approach. *Applied Energy*, *151*, 273–284. https://doi.org/10.1016/j.apenergy.2015.04.071
RVO. (2013). *SDE+ 2013*. https://www.rvo.nl/sites/default/files/Digitale brochure SDE+ 2013 (kleurenversie) def incl monitoring.pdf
RVO. (2023a). *Monitor zon-PV 2023*. https://www.rijksoverheid.nl/binaries/rijksoverheid/documenten/rapporten/2023/10/09/monitor-zon-pv-2023-in-nederland/Monitor+zon-PV+2023+definitief+correctie+2023+10+06.pdf
RVO. (2023b). *Beleid Zonne-energie*. https://www.rvo.nl/onderwerpen/zonne-energie/beleid
Salvador, S., & Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, *11*(5), 561–580. https://doi.org/10.3233/IDA-2007-11508
Schelling, T. C. (1971). Dynamic models of segregation†. *The Journal of Mathematical Sociology*, *1*(2), 143–186. https://doi.org/10.1080/0022250X.1971.9989794

Schwarz, N., & Ernst, A. (2009). Agent-based modeling of the diffusion of environmental innovations — An empirical approach. *Technological Forecasting and Social Change*, *76*(4), 497–511. https://doi.org/10.1016/j.techfore.2008.03.024
Shadowen, A. N. (2017). *Ethics and Bias in Machine Learning: A Technical Study of What Makes Us “Good.”* John Jay College of Criminal Justice.
Shafique, M., Luo, X., & Zuo, J. (2020). Photovoltaic-green roofs: A review of benefits, limitations, and trends. *Solar Energy*, *202*, 485–497. https://doi.org/10.1016/j.solener.2020.02.101
Smets, A., Jäger, K., Isabella, O., van Swaaij, R., & Zeman, M. (2016). *Solar Energy: The physics and engineering of photovoltaic conversion, technologies and systems*. UIT Cambridge Limited.
Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian Optimization of Machine Learning Algorithms. *Advances in Neural Information Processing Systems*, *4*.
So, S., Badloe, T., Noh, J., Bravo-Abad, J., & Rho, J. (2020). Deep learning enabled inverse design in nanophotonics. *Nanophotonics*, *9*(5), 1041–1057. https://doi.org/10.1515/nanoph-2019-0474
Stake, R. E. (2005). Qualitative Case Studies. In N. K. Denzin & Y. S. Lincoln (Eds.), *The Sage handbook of qualitative research* (3rd ed., pp. 443–466). Sage Publications Ltd.
Statista. (2022a). *Hours of sunshine in the Netherlands from 1990 to 2022*. https://www.statista.com/statistics/1012949/hours-of-sunshine-in-the-netherlands/
Statista. (2022b). *In your opinion, which energy sources should the Netherlands mainly use for electricity production?* https://www.statista.com/statistics/731682/opinions-on-energy-source-usage-in-the-future-in-the-netherlands/
Sumari, S., Ibrahim, R., Zakaria, N. H., & Ab Hamid, A. H. (2013). Comparing Three Simulation Model Using Taxonomy: System Dynamic Simulation, Discrete Event Simulation and Agent Based Simulation. *International Journal of Management Excellence*, *1*(3), 54. https://doi.org/10.17722/ijme.v1i3.9
Tan, V., Dias, P. R., Chang, N., & Deng, R. (2022). Estimating the Lifetime of Solar Photovoltaic Modules in Australia. *Sustainability*, *14*(9), 5336. https://doi.org/10.3390/su14095336
TNO. (2023). *Rendement zonnepanelen stijgt*. https://www.tno.nl/nl/duurzaam/hernieuwbare-elektriciteit/zonnemodules-massamaatwerk/rendement-zonnepanelen-stijgt/
TUDelft3d. (2024). *3dbag*. https://3dbag.nl/en/viewer
Turgut, Y., & Bozdag, C. E. (2023). A framework proposal for machine learning-driven agent-based models through a case study analysis. *Simulation Modelling Practice and Theory*, *123*, 102707. https://doi.org/10.1016/j.simpat.2022.102707
van der Kam, M., Lagomarsino, M., Rahmani Azad, Z., Parra, D., & Hahnel, U. J. J. (2023). Co-adoption pathways toward a low-carbon energy system. *IScience*, *26*(10), 107815. https://doi.org/10.1016/j.isci.2023.107815
van der Kam, M., Peters, A., van Sark, W., & Alkemade, F. (2019). Agent-Based Modelling of Charging Behaviour of Electric Vehicle Drivers. *Journal of Artificial Societies and Social Simulation*, *22*(4). https://doi.org/10.18564/jasss.4133
Vu, T. M., Buckley, C., Duro, J. A., Brennan, A., Epstein, J. M., & Purshouse, R. C. (2023). Can Social Norms Explain Long-Term Trends in Alcohol Use? Insights from Inverse Generative Social Science. *Journal of Artificial Societies and Social Simulation*, *26*(2). https://doi.org/10.18564/jasss.5077
Vu, T. M., Probst, C., Epstein, J. M., Brennan, A., Strong, M., & Purshouse, R. C. (2019). Toward inverse generative social science using multi-objective genetic programming. *Proceedings of the Genetic and Evolutionary Computation Conference*, 1356–1363. https://doi.org/10.1145/3321707.3321840
Whitner, R. B., & Balci, O. (1989). Guidelines for selecting and using simulation model verification techniques. *Proceedings of the 21st Conference on Winter Simulation*, 559–568.
Yin, R. K. (1994). *Case Study Research: Design and Methods* (2nd ed.). Sage Publications Inc.
Yin, R. K. (2012). A (very) brief refresher on the case study method. In *Applications of Case Study Research* (3rd ed., Vol. 34). Sage Publications Inc. https://us.sagepub.com/en-us/nam/applications-of-case-study-research/book235140
Zabinsky, Z. B. (2011). Random Search Algorithms. In *Wiley Encyclopedia of Operations Research and Management Science*. Wiley. https://doi.org/10.1002/9780470400531.eorms0704
Zappa, W., & van den Broek, M. (2018). Analysing the potential of integrating wind and solar power

in Europe using spatial optimisation under various scenarios. *Renewable and Sustainable Energy Reviews*, *94*, 1192–1216. https://doi.org/10.1016/j.rser.2018.05.071

Zhang, N., Lu, Y., Chen, J., & Hwang, B.-G. (2022). An agent-based diffusion model for Residential Photovoltaic deployment in Singapore: Perspective of consumers' behaviour. *Journal of Cleaner Production*, *367*, 132793. https://doi.org/10.1016/j.jclepro.2022.132793

Zhang, W., Valencia, A., & Chang, N.-B. (2023). Synergistic Integration Between Machine Learning and Agent-Based Modeling: A Multidisciplinary Review. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(5), 2170–2190. https://doi.org/10.1109/TNNLS.2021.3106777

Zhang, Y., Grignard, A., Lyons, K., Aubuchon, A., & Larson, K. (2018). Real-time machine learning prediction of an agent-based model for urban decision-making . *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, 2171–2173.

Zonneplan. (2012). *Zonnepanelen in Heerde*. https://www.zonneplan.nl/zonnepanelen-in-heerde

Appendix

A

1st Python Script for NetLogo Model

A.1. Script for extraction of the geolocations

```
import json
import csv
import geopandas as gpd
from pyproj import Transformer
from shapely.geometry import Point

# Specify the path to your JSON file
json_file_path = r'C:\Filepath\Patch 1.json'

# Load JSON data from the file into a Python dictionary
with open(json_file_path, 'r') as file:
    data = json.load(file)

# Extract x_max and y_max coordinates from "geographicalExtent" data for each city
object
geographical_extent_data_2d = []

for key, value in data['CityObjects'].items():
    # Access the 'geographicalExtent' field within each object
    geographical_extent = value.get('geographicalExtent')
    if geographical_extent is not None and len(geographical_extent) == 6:
        # Extract x_max and y_max coordinates (ignoring z)
        _, _, _, x_max, y_max, _ = geographical_extent
        # Append the 2D geographical extent data to the list
        geographical_extent_data_2d.append([x_max, y_max])

# Define the original CRS (RD New) and the target CRS (WGS84)
original_crs = 'EPSG:28992'
target_crs = 'EPSG:4326'  # WGS84

# Initialise the transformer
transformer = Transformer.from_crs(original_crs, target_crs)
```

```
# Transform the coordinates to WGS84
household_coordinates_Patch1 = [transformer.transform(x, y) for x, y in
geographical_extent_data_2d]

# Define the CSV file path to save the transformed data
csv_file_path_wgs84 = r'C:\Filepath\Patch 1. csv'

# Write the WGS84 geographical extent data to a CSV file
with open(csv_file_path_wgs84, 'w', newline='') as csvfile:
    writer = csv.writer(csvfile)
    # Write the header row
    writer.writerow(['longitude', 'latitude'])
    # Write the WGS84 geographical extent data
    writer.writerows(household_coordinates_Patch1)

print("Geographical extent data has been transformed to WGS84 and saved to CSV:",
csv_file_path_wgs84)

# Create a GeoDataFrame from the transformed coordinates
geometry = [Point(lon, lat) for lon, lat in household_coordinates_Patch1]
gdf = gpd.GeoDataFrame(geometry=geometry, columns=['geometry'], crs=target_crs)

# Define the shapefile path to save the transformed data
household_coordinates_Patch1 = r'C:\Filepath\Patch 1.shp'

# Save the GeoDataFrame as a shapefile
gdf.to_file(household_coordinates_Patch1)

print("Geographical extent data has been transformed to WGS84 and saved to
Shapefile:", household_coordinates_Patch1)
```

A.2. Script for extraction of roof sizes

```
import json
import csv
import os

# Specify the path to your JSON file
json_file_path = r'C:\Filepath\Patch 1.json'

# Load JSON data from the file into a Python dictionary
with open(json_file_path, 'r') as file:
    data = json.load(file)

# Extract "b3_opp_dak_schuin" values as an array
b3_opp_dak_schuin_values = []
```

```python
for key, value in data['CityObjects'].items():
    # Access the 'attributes' dictionary within each object
    attributes = value.get('attributes', {})

    # Extract the value of 'b3_opp_dak_schuin' and append it to the list
    b3_opp_dak_schuin = attributes.get('b3_opp_dak_schuin')
    if b3_opp_dak_schuin is not None:
        b3_opp_dak_schuin_values.append(b3_opp_dak_schuin)

# Define the path for the CSV file
csv_file_path = os.path.join(os.path.dirname(json_file_path),
'b3_opp_dak_schuin_values.csv')

# Write the b3_opp_dak_schuin_values array to a CSV file
with open(csv_file_path, 'w', newline='') as csv_file:
    writer = csv.writer(csv_file)
    writer.writerow(['b3_opp_dak_schuin_values'])  # Write header
    writer.writerows(map(lambda x: [x], b3_opp_dak_schuin_values))

print(f"CSV file saved at: {csv_file_path}")
```

*Note that these Python scripts are for 1 of the total number of used patches. For conciseness purposes, the scripts of the other patches have not been included.

B

2nd Python Script for NetLogo Model

B.1. Script for patch combination for the geolocations

```
import json
import csv
import geopandas as gpd
from pyproj import Transformer
from shapely.geometry import Point
import random

# List of paths to your JSON files
json_file_paths = [
    r'C:\filepath\Patch 1.json',
    r'C:\filepath\Patch 2.json',
    r'C:\filepath\Patch 3.json',
    r'C:\filepath\Patch 4.json'
]

# Initialise empty lists to store transformed data
all_geographical_extent_data_2d = []

# Load and process each JSON file
for json_file_path in json_file_paths:
    with open(json_file_path, 'r') as file:
        data = json.load(file)

    # Extract x_max and y_max coordinates from "geographicalExtent" data for each
city object
    for key, value in data['CityObjects'].items():
        geographical_extent = value.get('geographicalExtent')
        if geographical_extent is not None and len(geographical_extent) == 6:
            _, _, _, x_max, y_max, _ = geographical_extent
            all_geographical_extent_data_2d.append([x_max, y_max])

# Shuffle the list of geographical extent data
```

```
random.shuffle(all_geographical_extent_data_2d)

# Select the first 9445 items from the shuffled list
selected_coordinates = all_geographical_extent_data_2d[:9445]

# Define the original CRS (RD New) and the target CRS (WGS84)
original_crs = 'EPSG:28992'
target_crs = 'EPSG:4326'  # WGS84

# Initialise the transformer
transformer = Transformer.from_crs(original_crs, target_crs)

# Transform the coordinates to WGS84
all_coordinates_wgs84 = [transformer.transform(x, y) for x, y in
selected_coordinates]

# Define the CSV file path to save the transformed data
csv_file_path = r'C:\filepath\combined_household_coordinates.csv'

# Write the WGS84 geographical extent data to a CSV file
with open(csv_file_path, 'w', newline='') as csvfile:
    writer = csv.writer(csvfile)
    # Write the header row
    writer.writerow(['longitude', 'latitude'])
    # Write the WGS84 geographical extent data
    writer.writerows(all_coordinates_wgs84)

print("Geographical extent data has been transformed to WGS84 and saved to CSV:",
csv_file_path)

# Create a GeoDataFrame from the transformed coordinates
geometry = [Point(lon, lat) for lon, lat in all_coordinates_wgs84]
gdf = gpd.GeoDataFrame(geometry=geometry, columns=['geometry'], crs=target_crs)

# Define the shapefile path to save the transformed data
shapefile_path = r'C:\filepath\combined_household_coordinates.shp'

# Save the GeoDataFrame as a shapefile
gdf.to_file(shapefile_path)

print("Geographical extent data has been transformed to WGS84 and saved to
Shapefile:", shapefile_path)
```

B.2. Script for patch combination for the roofsizes

```
import json
import csv
import os
```

```python
import random

# List of paths to your JSON files
json_file_paths = [
    r'C:\filepath\Patch 1.json',
    r'C:\filepath\Patch 2.json',
    r'C:\filepath\Patch 3.json',
    r'C:\filepath\Patch 4.json'
]

# Initialise empty list to store all b3_opp_dak_schuin values
all_b3_opp_dak_schuin_values = []

# Load and process each JSON file
for json_file_path in json_file_paths:
    with open(json_file_path, 'r') as file:
        data = json.load(file)

    # Extract "b3_opp_dak_schuin" values as an array
    b3_opp_dak_schuin_values = []
    for key, value in data['CityObjects'].items():
        attributes = value.get('attributes', {})
        b3_opp_dak_schuin = attributes.get('b3_opp_dak_schuin')
        if b3_opp_dak_schuin is not None:
            b3_opp_dak_schuin_values.append(b3_opp_dak_schuin)

    all_b3_opp_dak_schuin_values.extend(b3_opp_dak_schuin_values)

# Shuffle the list of b3_opp_dak_schuin values
random.shuffle(all_b3_opp_dak_schuin_values)

# Select the first 9445 values from the shuffled list
selected_values = all_b3_opp_dak_schuin_values[:9445]

# Define the path for the CSV file
csv_file_path = os.path.join(os.path.dirname(json_file_paths[0]),
'combined_b3_opp_dak_schuin_values.csv')

# Write the combined b3_opp_dak_schuin_values to a CSV file
with open(csv_file_path, 'w', newline='') as csv_file:
    writer = csv.writer(csv_file)
    writer.writerow(['b3_opp_dak_schuin_values'])  # Write header
    writer.writerows(map(lambda x: [x], selected_values))

print(f"CSV file saved at: {csv_file_path}")
```

**Note that only the first 9445 items are selected. This is based on the municipality of Bloemendaal and its number of households. For each of the municipalities, this number will therefore be different.*

C

Income Distributions NetLogo

C.1. Income Distribution Municipality 1 (Bloemendaal)

Table 39: Income distribution for Bloemendaal (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	5%
€10.001 - €20.000	6%
€20.001 - €30.000	6%
€30.001 - €40.000	6%
€40.001 - €50.000	6%
€50.001 - €60.000	7%
€60.001 - €70.000	8%
€70.001 - €80.000	9%
€80.001 - €90.000	13%
€90.001 - €100.000	34%

C.2. Income Distribution Municipality 2 (Dantumadiel)

Table 40: Income distribution for Dantumadiel (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	7%
€10.001 - €20.000	13%
€20.001 - €30.000	13%
€30.001 - €40.000	13%
€40.001 - €50.000	12%
€50.001 - €60.000	11%
€60.001 - €70.000	10%
€70.001 - €80.000	8%
€80.001 - €90.000	7%
€90.001 - €100.000	5%

C.3. Income Distribution Municipality 3 (Laren)

Table 41: Income distribution for Laren (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	7%
€10.001 - €20.000	7%
€20.001 - €30.000	6%
€30.001 - €40.000	6%
€40.001 - €50.000	7%
€50.001 - €60.000	7%
€60.001 - €70.000	7%
€70.001 - €80.000	9%
€80.001 - €90.000	12%
€90.001 - €100.000	32%

C.4. Income Distribution Municipality 4 (Oegstgeest)

Table 42: Income distribution for Oegstgeest (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	6%
€10.001 - €20.000	5%
€20.001 - €30.000	6%
€30.001 - €40.000	6%
€40.001 - €50.000	7%
€50.001 - €60.000	8%
€60.001 - €70.000	10%
€70.001 - €80.000	11%
€80.001 - €90.000	15%
€90.001 - €100.000	25%

C.5. Income Distribution Municipality 5 (Vaals)

Table 43: Income distribution for Vaals (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	14%
€10.001 - €20.000	15%
€20.001 - €30.000	14%
€30.001 - €40.000	12%
€40.001 - €50.000	10%
€50.001 - €60.000	8%
€60.001 - €70.000	8%
€70.001 - €80.000	7%
€80.001 - €90.000	6%
€90.001 - €100.000	6%

C.6. Income Distribution Municipality 6 (Westerveld)

Table 44: Income distribution for Westerveld (CBS, 2023a)

Income range	Percentage of households
€0 - €10.000	6%
€10.001 - €20.000	9%
€20.001 - €30.000	9%
€30.001 - €40.000	12%
€40.001 - €50.000	11%
€50.001 - €60.000	11%
€60.001 - €70.000	11%
€70.001 - €80.000	11%
€80.001 - €90.000	11%
€90.001 - €100.000	10%

*The data does not include incomes over €100.000. This is a limitation of this study, especially for the municipalities with a higher income.

D

Initial PV share NetLogo

Table 45: The initial PV share per municipality, as used in the NetLogo model (CBS, 2020)

Municipality	Number of households	Initial number of PV installations	Initial PV share	Median income	Population density (# inhabitants per km^2)
Bloemendaal	9.445	75	0.008	€51.400	602
Dantumadiel	8.016	187	0.023	€34.650	227
Laren	5.255	10	0.002	€48.250	944
Oegstgeest	11.115	134	0.012	€47.750	3530
Vaals	5.642	0	0.000	€29.550	427
Westerveld	8.892	141	0.016	€38.750	71

E

Agent State Variables NetLogo

Table 46: State variables of the agents of the ABM, modified to fit the case of this research

Name	**Description**	**Domain**	**Static?**
General			
Esystem	The energy system of the household. The list item equals 1 if the household has PV installed and 0 if not.	{0, 1}	No
Income	The income of the household. Distribution based on CBS (2023a). See Appendix C.	[0, 100.000]. *Depends on municipality.*	No
Income class	Household income class. It is equal to 1 when household income is less than €40.000 / year, 2 when it is between €40.000 and €60.000, and 3 when it is higher than €60.000.	{1, 2, 3}	No
Roof size	The roof size of a household. Based on TUDelft3d (2024).	[0, 10.000]. *Depends on municipality.*	Yes
Number of neighbours	Relates to the size of the social network of the household. Depends on the number of links that a household forms, which is an interface parameter.	[0, 10]	Yes
Weight economic utility	Determines consumer preferences utility function	[0, 1]	Yes
Weight of comfort utility	Determines consumer preferences utility function	{0, -1}	Yes
Weight of social utility	Determines consumer preferences on social norm	[0, 1]	No
Weight environmental utility	Determines consumer preferences on environmental attitude	[0, 1]	Yes
Comfort utility	Aesthetics of PV per household	{0, 1}	Yes
Economic utility	Payback utility	{0, 1}	No
Environmental utility	Environmental utility, dependent on social network	{0, 1}	No
Social utility	Utility for influencing social network	{0, 1}	No

F

Environmental State Variables NetLogo

Table 47: State variables of the environment of the ABM, modified to fit the case of this research

Name	**Description**	**Domain**	**Static?**
General			
PV share	Percentage of households that installed solar PV	[0, 1]	No
PV-no, PV-yes	Values used to calculate both utility for and against PV	{0, 1}	No
Household fields	Variable related to the geographical positioning of households	-	Yes
PV costs per m2	External constant for PV utility calculations related to the costs of PV panels per m2	€ / m2	Yes
PV peak power	External constant for PV utility calculation related to the peak power	kW/m2	Yes
Sunshine hours	External constant for PV utility calculation related to the peak power	hours	Yes
Performance ratio	External constant for PV utility calculation related to the performance ratio of the solar panels	%	Yes
PV lifetime	External constant for PV utility calculation related to the assumed lifetime of PV in years	years	Yes
Grid electricity costs	External constant for PV utility calculation related to the electricity price	€/kWh	Yes
Avoided cost per kWh	Variable calculated from external constants for PV utility calculations	€	Yes
CO_2 per kWh	Variable calculated from external constants for PV utility calculations	kg/kWh	Yes
Average CO_2	Variable calculated from external constants for PV utility calculations	kg/kWh	Yes
Visibility value	Variable calculated from external constants for PV utility calculations	[0, 1]	No
Number of turtles	Total number of turtles (households)	*Depends on municipality*	Yes
Initial PV share	The initial PV share at t=0	[0, 1] *Depends on municipality.*	Yes

G

Pseudo-Code Main Simulation NetLogo

```
Initialisation
      Load household data and setup social network links
      Initialise financial, environmental and social variables
      Calculate utilities and barriers for each household
      Determine visibility and income barriers for PV installations

For each timestep
      Update price
      Update life-cycle
      Update Greenhouse gas emissions
      Update age of solar panels

      For each agent
            Evaluate adoption decision for solar panels
            Check for adoption barriers
            Adopt technology if decision meets the threshold
            Set opinion on adopted technology and share with neighbours
            Update emotions on technology for each neighbour
```

H

Excluded Parameters for the IM process

Table 48: Excluded parameters for the IM process, including default value in the NetLogo model

Parameter	Description	Distribution	Default
True / False Parameters			
MF_Income	Includes income as a measure of PBC as a probabilistic barrier, giving households with a higher income a higher chance to consider solar PV than households with a lower income.	[True, False]	N / A
MF_Income_Barrier	It includes income as a measure, but it is attached to a threshold instead of probabilistic behaviour.	[True, False]	N / A
Financial_Information	Switch on whether households are informed on the finances by the installer.	[True, False]	[“False”]
Visibility	Switch of the visibility barrier of the technology, dependent on market share and advertisement	[True, False]	[“True”]
Sparking_Events	Switch for making the visibility barrier dependent on the social network as well	[True, False]	[“True”]
Info_Costs	Switch on the implementation of the information on financial aspects, as this will increase the initial investment costs dependent on the amount of time spent searching for information and the probability of information on economics already provided by the installers.	[True, False]	[“True”]
Info_Costs_Revenue	Switch to including PV's potential revenue in the calculation of the costs of the time spent on information search.	[True, False]	[“True”]
Info_Costs_Income	Switch to include the household’s income in the calculation of the costs of the time that is spent on information search.	[True, False]	[“True”]
Information_Threshold	Switch whether or not households who did not invest a certain amount of time in searching for	[True, False]	[“False”]

	information is not continuing the decision-making process.		
Uncertainty	Switch on the implementation of the information on financial aspects as inaccuracy of the calculation of the economic utility.	[True, False]	[“True”]
Option Parameters			
Weight_distribution	The weight distribution for the household, which can be set to either homogenous or heterogenous.	[“heterogenous”, “homogenous”]	[“homogenous”]
Information_distribution	The probability distribution of the costs/inaccuracy of information over all households for Uncertainty and Info_Costs.	["empirical", "uniform", "normal", "poisson"]	[“normal”]
TPB_operationalisation	Determine the model (MF, SE or RR) used for the Theory of Planned Behaviour. Due to computational limitations (overload errors), his parameter is fixed to the MF model.	[“MF”, “SE”, “RR”]	[“RR”]
Ajadv	The strength of influence of advertisement on the visibility	[“0.02”, “0”]	[“0.02””]
Ajsoc	The strength of influence of social networks on the visibility	[“0.02”, “0”]	[“0.02””]
Discrete Parameters			
Random_links	The probability for an agent to have a link based on the geographical distance between two agents	(0, 0.1, 0.01)	0
Close_links	The number of links an agent’s social network can have is set depending on an agent’s income class.	(0, 5, 1)	2
MF_income_barrier_value	The threshold value that MF_Income_barrier is attached to.	(0, 1, 0.01)	N / A
Probability_financial_information	The percentage of households that are informed on the finances by the installer.	(0, 1, 0.01)	N / A
SE_importance_control	The importance of the control parameter in the utility function for the SE model.	(0, 1, 0.01)	N / A
SE_importance_attitide	The importance of the attitude parameter in the utility function for the SE model.	(0, 1, 0.01)	N / A
RR_sensitivity_barrier	Can be included in case one wants to test the PBC barrier, comparing payback to income.	(0, 1 0.01)	N/A
RR_sia	The threshold value against which a household’s utility for the PV installation is compared. If the threshold is smaller than the utility, PV is installed.	(0, 0.25, 0.01)	0.15
InputRandomSeed	Random seed for all stochastic variables in this model	(0,100,10)	10
Influence_cost_time	The percentage of monthly revenue and/or monthly income that is used to calculate the costs	(0, 1, 0.01)	0.25

	of information of financial aspects of solar PV.		
Information_threshold_value	The value for the information threshold is as mentioned above.	(0, 1, 0.01)	0.25
Interest rate	The interest rate of the solar PV investment	(0, 0.1, 0.02)	0.06
PV_SDE_premium	Subsidy by the government is to be used for the financing of solar panels.	(0, 0.3, 0.05)	0.10

I

Pseudo-code

I.1. Pseudo-code for the *model* file

```
Import necessary libraries

Start a NetLogo instance

Load the model

Define function run_model_with_parameters:
      Set the input parameters. Do not give them an initial value.
      Setup the model
      Double-check that the model stops after 11 ticks
      Run the model
            Count the number of solar PV installations per tick
      Create a data frame with data per tick
      Convert the data frame to the right data format
      Return the data frame
```

I.2. Pseudo-code for the *generate_data* file

```
Import necessary libraries

Load the data from the Excel file

Filter out the header rows

Extract the years from the first row

Extract the municipalities and corresponding installation data

Create a dictionary to store the installation data for each municipality

Convert the dictionary to a data frame

Save the data frame to a CSV file called “solar_pv_training_data”
```

I.3. Pseudo-code for the *plot_fit* file

```
Import necessary libraries

Define function plot_fit:
      Calculate predictions
      Convert predictions to the right data format
      Plot the results
      Add error to plot (if provided)
      Add parameters to plot (if provided)
      Present the legend
      Show the plot
```

I.4. Pseudo-code for the *error* file

```
Import necessary libraries

Read the “solar_pv_training_data.csv” file

Extract the data from the municipality

Define the custom distance metric function
      Calculate model statistics (median and mean)
      Check if any of the values fall within 10% of the model_std
            Within the range? Return 0
            Not within the range? Return absolute difference

Define the error function, using the predictions and real municipality solar pv
adoption data
      Convert municipality data to a 1D array if it is a DataFrame
      Define model_std as the standard deviation of the predictions array
      Calculate the distance using fastdtw with the custom metric.
      Return the distance
```

I.5. Pseudo-code for the *optimise* file

```
Import necessary libraries

Define a custom estimator class
      Initialise the class with default parameters
            Set the parameters
                  Store the parameters in a dictionary
      Define the fit method
      Define the prediction method
            Run the model with the stored parameters
      Define the score method to evaluate model performance
            If predictions are already? Use them
            Predictions not yet available? Generate them
```

```
        Calculate the error using the error function
              Make sure to use a negative sign because RandomisedSearchCV tries
              to maximise the score!
```

I.6. Pseudo-code for the *custom_cv* file

```
Import necessary libraries

Define a custom cross-validator class
      Initialise the class with the number of splits
      Return the number of splits
      Generate indices to split data into training and test sets
            Calculate the number of samples
            Iterate over fold sizes to yield training and test indices
                  Test indices for the current fold
                  Training indices are everything else
```

I.7. Pseudo-code for the *main* file

```
Import necessary libraries

State the number of runs

Define a function to load data from the CSV file
      Load the training data

Define a custom discrete function for generating search spaces

Define a function to save error values and parameter values to a CSV file
      Write data for each run

Load the training data

Extract the years from the data

Extract the solar PV adoption data for all municipalities

Transpose the original matrix of the solar PV adoption data for all municipalities

Extract the data for only the municipality

Initialise the model

Define the parameter distributions

Create lists to store error values and corresponding parameter values for each run

For each run:
```

```
      Perform randomised search or bayesian search for hyperparameter tuning
      Apply the fit method defined in the optimise file

Use the best model to make predictions on the training data

Compute the error value between the predictions and the actual values

Append error value and corresponding parameters to the lists

Save the error values and parameter values to a CSV file

Plot the fit of the best estimator for the last run
```

I.8. Pseudo-code for the *validation_curve* file

```
Import necessary libraries

Load the data

Define the custom_cross_validation_score function
      Initialise custom cross-validation object
      Iterate over train-test splits
            Split data into training and testing sets
            Set model parameters
            Train the model
            Generate predictions
            Compute error if municipality data is available
      Return the scores

Define the main script
      Define the file path
      Set working directory
      Load the data
      Initialise the model object
      Define the range for number of splits in cross validation
      Create lists to store the mean cross-valiated score for each number of splits
      Define default model parameters
      Loop over the range for the number of splits
      Plot the validation curve
      Find the optimal number of splits
```

The actual code for this thesis can be found here.

J

Data Analysis Bloemendaal

The table below presents the results of the data analysis conducted for the municipality of Bloemendaal. Each row corresponds to a single run in the model. A run consists of 30 fits comprising 2 splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

J.1. Results using the Random Search Algorithm

Table 49: Data analysis results of the municipality of Bloemendaal for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[75. 136. 203. 300. 378. 453. 612. 794. 1047. 1447. 1879]				
788	0.2	0	0.7	0.7	[79. 120. 190. 266. 349. 461. 590. 767. 977. 1223. 1538.]
972	0.2	0.8	0.4	0.7	[79. 225. 390. 559. 745. 908. 1105. 1270. 1438. 1621. 1774.]
1031	0.7	0.1	0.6	0.9	[79. 121. 172. 245. 342. 429. 562. 731. 912. 1144. 1394.]
1085	0.1	0	0.4	0.4	[79. 114. 169. 218. 309. 383. 479. 628. 810. 980. 1227.]
1712	0.1	0.9	0.5	0.7	[79. 288. 550. 785. 1068. 1295. 1518. 1753. 1977. 2188. 2398.]
1715	0.1	0.8	0.1	0.5	[79. 203. 316. 412. 507. 613. 709. 804. 880. 949. 1028.]
1753	0.9	0.5	0	0.8	[79. 132. 206. 285. 378. 463. 551. 638. 751. 843. 945.]
1787	0.4	0.1	0.3	0.5	[79. 117. 157. 195. 261. 326. 402. 498. 620. 775. 933.]
1921	0.5	0.5	0.5	0.9	[79. 144. 215. 286. 349. 417. 484. 586. 671. 783. 894.]
2344	0.3	0.7	0.1	0.6	[79. 149. 230. 320. 395. 475. 550. 607. 677. 730. 776.]
2967	0.7	0.3	0.6	0.9	[79. 214. 407. 611. 854. 1126. 1463. 1847. 2257. 2738. 3188.]

3072	0.3	0.9	0.2	0.8	[79. 125. 179. 220. 271. 329. 372. 422. 487. 540. 588.]
3090	0	0.9	0	0.3	[79. 774. 1261. 1595. 1785. 1955. 2055. 2119. 2168. 2193. 2210.]
3511	0.2	0.9	0.2	0.4	[79. 416. 722. 1070. 1324. 1627. 1892. 2167. 2433. 2659. 2871.]
3755	0.6	0.4	0	0.5	[79. 301. 555. 828. 1091. 1388. 1720. 2069. 2448. 2828. 3234.]
3840	0.6	0.3	0.4	0.8	[79. 99. 133. 162. 189. 217. 257. 288. 326. 374. 427.]
3855	0.3	0.2	0.6	0.8	[79. 108. 127. 154. 181. 204. 248. 285. 321. 368. 425.]
4231	0.8	0.4	0.3	0.9	[79. 96. 127. 152. 175. 197. 227. 256. 291. 322. 366.]
5426	0.4	0.3	0.3	0.7	[79. 89. 107. 124. 137. 149. 160. 173. 183. 194. 205.]
6018	0.2	0.7	0.3	0.8	[79. 88. 92. 98. 104. 113. 119. 126. 128. 133. 138.]
6071	0.2	0.9	0.2	0.9	[79. 88. 94. 100. 107. 109. 114. 117. 125. 131. 134.]
6162	0.1	0	0.4	0.9	[79. 90. 96. 103. 104. 108. 111. 114. 115. 118. 118.]
6189	0.1	0.7	0.1	0.7	[79. 90. 93. 96. 100. 105. 109. 111. 112. 113. 117.]
6219	0.2	0.2	0	0.6	[79. 88. 91. 94. 97. 103. 106. 106. 110. 111. 112.]
6928	0.1	0.8	0.2	0.3	[79. 513. 978. 1364. 1737. 2106. 2422. 2717. 3005. 3265. 3517.]

J.2. Results using the Bayesian Search Algorithm

Table 50: Data analysis results of the municipality of Bloemendaal for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[75. 136. 203. 300. 378. 453. 612. 794. 1047. 1447. 1879]				
863	0.4	0.6	0.6	0.9	[79. 185. 283. 396. 531. 673. 850. 1028. 1194. 1378. 1552.]
1080	0.5	0.7	0.1	0.6	[79. 216. 335. 466. 589. 730. 858. 996. 1159. 1294. 1431.]
1133	0.2	0.5	0.5	0.7	[79. 164. 256. 370. 508. 648. 776. 921. 1072. 1211. 1365.]
1917	0.9	0.7	0.3	0.9	[79. 267. 448. 654. 883. 1113. 1364. 1652. 1977. 2284. 2575.]
2560	0.8	0.2	0.2	0.7	[79. 111. 148. 194. 244. 295. 360. 434. 524. 603. 705.]
3333	0.1	0.5	0.3	0.4	[79. 341. 580. 835. 1111. 1420. 1730. 2065. 2402. 2693. 3001.]
3932	0.2	0.2	0.8	0.9	[79. 110. 135. 168. 192. 231. 266. 298. 337. 375. 413.]
4362	0.4	0.1	0.2	0.5	[79. 99. 119. 141. 164. 188. 223. 255. 287. 317. 348.]
4621	0.6	0.6	0.1	0.5	[79. 343. 608. 919. 1210. 1538. 1914. 2288. 2652. 3023. 3383.]
4808	0.4	0.5	0.5	0.9	[79. 95. 122. 143. 167. 188. 206. 223. 246. 260. 285.]
5717	0.6	0.3	0.1	0.8	[79. 93. 103. 114. 126. 132. 145. 150. 155. 160. 170.]

5986	0.4	0	0	0.6	[79. 97. 112. 120. 125. 129. 130. 133. 135. 135. 135.]
6057	0.2	0.2	0.2	0.6	[79. 85. 93. 98. 106. 110. 116. 121. 125. 129. 135.]
6078	0.2	0.6	0	0.6	[79. 89. 100. 104. 112. 115. 119. 122. 124. 128. 130.]
6116	0.5	0.5	0	0.9	[79. 86. 94. 97. 99. 108. 116. 120. 121. 123. 125.]
6206	0.2	0.5	0.1	0.8	[79. 84. 89. 94. 95. 102. 107. 109. 112. 112. 113.]
6463	0	0	0.1	0.9	[79. 79. 79. 79. 79. 79. 79. 79. 79. 79. 79.]
6498	0.7	0.8	0.6	0.9	[79. 330. 601. 953. 1293. 1693. 2101. 2503. 2997. 3432. 3870.]
10040	0.3	0.8	0.3	0.4	[79. 541. 1030. 1471. 1919. 2387. 2810. 3204. 3588. 3968. 4312.]
10867	0.7	0.7	0.7	0.9	[79. 410. 779. 1195. 1660. 2176. 2715. 3292. 3816. 4361. 4868.]
19738	0.9	0.5	0.7	0.9	[79. 627. 1238. 1922. 2636. 3301. 3913. 4542. 5141. 5703. 6201.]
21054	0.4	0.5	0.1	0.2	[79. 792. 1489. 2267. 2923. 3549. 4118. 4689. 5208. 5668. 6141.]
22423	0.9	0.6	0.2	0.5	[79. 769. 1473. 2206. 2914. 3577. 4225. 4885. 5481. 6089. 6571.]
26365	0	0.9	0.1	0	[79. 1729. 2773. 3521. 4055. 4456. 4715. 4909. 5062. 5201. 5293.]
27000	0.8	0.2	0.4	0.3	[79. 1034. 1894. 2680. 3519. 4184. 4795. 5399. 5938. 6450. 6885.]

K

Data Analysis Dantumadiel

In the table below, the results of the data analysis conducted for the municipality of Dantumadiel are presented. Each row corresponds to a single run in the model. A run consists of 30 fits comprising two splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

K.1. Results using the Random Search Algorithm

Table 51: Data analysis results of the municipality of Dantumadiel for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[187. 322. 417. 503. 591. 715. 977. 1360. 2344. 3228. 4126.]				
1740	0.4	0.3	0.7	0.8	[184. 417. 656. 967. 1350. 1740. 2171. 2577. 3043. 3537. 3994.]
2901	0.3	0.9	0.3	0.6	[184. 466. 739. 1018. 1342. 1631. 1926. 2227. 2509. 2797. 3062.]
3258	0.9	0.4	0.2	0.8	[184. 334. 526. 707. 910. 1139. 1422. 1694. 1999. 2316. 2644.]
3330	0.9	0.8	0.2	0.9	[184. 382. 597. 826. 1061. 1306. 1582. 1865. 2131. 2449. 2745.]
3592	0.1	0.4	0.6	0.7	[184. 308. 444. 630. 784. 980. 1209. 1488. 1769. 2106. 2451.]
3709	0.8	0.4	0.1	0.7	[184. 315. 479. 659. 850. 1070. 1319. 1591. 1864. 2173. 2480.]
4200	0.4	0.6	0.1	0.5	[184. 396. 606. 794. 994. 1218. 1418. 1640. 1865. 2111. 2314.]
4341	0.5	0.6	0	0.6	[184. 386. 584. 791. 986. 1185. 1367. 1567. 1784. 2005. 2218.]
4638	0.1	0.8	0.5	0.8	[184. 322. 510. 683. 869. 1043. 1236. 1438. 1652. 1856. 2044.]
4691	0.7	0.5	0.7	0.8	[184. 722. 1252. 1821. 2377. 2992. 3542. 4071. 4538. 4996. 5430.]
4866	0.2	0.6	0	0.4	[184. 434. 694. 922. 1123. 1317. 1498. 1671. 1827. 1944. 2079.]

5090	0.1	0.7	0	0.4	[184. 446. 705. 936. 1128. 1315. 1467. 1616. 1741. 1861. 1943.]
5327	0	0.5	0.6	0.6	[184. 612. 1166. 1729. 2343. 2970. 3573. 4200. 4779. 5316. 5805.]
6063	0.2	0.2	0.7	0.8	[184. 246. 315. 406. 490. 594. 716. 834. 981. 1150. 1347.]
6468	0.5	0.6	0.4	0.9	[184. 270. 354. 445. 540. 647. 785. 903. 1018. 1140. 1281.]
6490	0	0.6	0.7	0.9	[184. 258. 338. 432. 525. 634. 741. 864. 987. 1118. 1254.]
7625	0.8	0.3	0.3	0.9	[184. 243. 309. 352. 432. 493. 568. 647. 726. 835. 948.]
8090	0.2	0.8	0.4	0.9	[184. 256. 314. 372. 439. 505. 562. 635. 708. 779. 847.]
9767	0	0.1	0.5	0.7	[184. 241. 289. 335. 374. 414. 448. 470. 501. 522. 544.]
10388	0.6	0.5	0.1	0.8	[184. 219. 256. 279. 308. 336. 362. 389. 413. 437. 465.]
10673	0.1	0.6	0.2	0.6	[184. 218. 241. 262. 292. 316. 340. 364. 384. 409. 432.]
10738	0.3	0	0.4	0.9	[184. 218. 247. 269. 301. 321. 349. 367. 389. 408. 423.]
10904	0.3	0.6	0.3	0.9	[184. 211. 230. 252. 276. 291. 316. 343. 369. 385. 406.]
11308	0.3	0.1	0.4	0.9	[184. 206. 228. 251. 265. 288. 301. 316. 329. 341. 358.]
12236	0.2	0.5	0.1	0.9	[184. 196. 205. 212. 222. 226. 232. 238. 240. 244. 250.]

K.2. Results using the Bayesian Search Algorithm

Table 52: Data analysis results of the municipality of Dantumadiel for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[187. 322. 417. 503. 591. 715. 977. 1360. 2344. 3228. 4126.]				
1534	0.2	0.1	0.9	0.8	[184. 310. 491. 720. 1066. 1417. 1828. 2362. 2913. 3541. 4194.]
2203	0.7	0.8	0.4	0.7	[184. 566. 958. 1363. 1794. 2211. 2632. 3084. 3491. 3899. 4291.]
2253	0.8	0.5	0.4	0.8	[184. 461. 791. 1145. 1515. 1899. 2309. 2727. 3100. 3526. 3918.]
2323	0.6	0.8	0.5	0.8	[184. 513. 860. 1241. 1602. 1956. 2335. 2728. 3127. 3510. 3915.]
2397	0.2	0.8	0.5	0.6	[184. 593. 1021. 1419. 1854. 2343. 2792. 3236. 3686. 4118. 4506.]
2693	0.3	0.9	0.7	0.8	[184. 533. 947. 1371. 1845. 2363. 2862. 3365. 3861. 4350. 4755.]
2726	0.7	0.8	0.2	0.7	[184. 445. 709. 974. 1262. 1601. 1886. 2211. 2519. 2856. 3216.]
3263	0.1	0.2	0.7	0.7	[184. 287. 399. 533. 694. 900. 1165. 1445. 1772. 2142. 2610.]
4951	0.1	0.9	0.2	0.6	[184. 385. 565. 760. 951. 1124. 1305. 1475. 1612. 1761. 1906.]
5204	0.1	0.9	0.6	0.6	[184. 755. 1327. 1960. 2564. 3119. 3688. 4182. 4648. 5115. 5527.]
5533	0	0	0.9	0.9	[184. 288. 397. 508. 622. 732. 856. 1003. 1147. 1323. 1543.]

5552	0.2	0.9	0.1	0.6	[184. 343. 494. 642. 790. 941. 1081. 1216. 1343. 1462. 1594.]
5694	0.7	0.9	0.1	0.9	[184. 297. 410. 546. 677. 813. 950. 1083. 1242. 1403. 1519.]
6623	0.1	0.4	0.3	0.5	[184. 266. 351. 450. 578. 684. 786. 886. 997. 1124. 1237.]
9043	0.6	0.2	0.4	0.9	[184. 223. 260. 315. 356. 405. 450. 499. 549. 617. 661.]
9686	0.7	0	0	0.8	[184. 254. 303. 340. 368. 405. 446. 478. 503. 528. 557.]
10049	0	0.6	0.5	0.9	[184. 237. 283. 321. 357. 397. 425. 444. 466. 487. 504.]
10144	0.5	0.8	0	0.9	[184. 225. 263. 299. 331. 365. 391. 421. 446. 476. 494.]
10400	0.6	0.1	0.1	0.7	[184. 223. 245. 272. 308. 338. 365. 384. 412. 435. 465.]
10819	0	0.8	0.2	0.7	[184. 213. 245. 280. 311. 336. 355. 374. 386. 397. 412.]
11223	0.3	0.7	0.2	0.9	[184. 208. 233. 250. 267. 285. 298. 322. 336. 357. 368.]
11479	0.2	0.7	0.2	0.9	[184. 204. 225. 251. 267. 281. 291. 303. 316. 326. 340.]
11644	0.1	0.7	0.2	0.8	[184. 203. 229. 251. 266. 282. 297. 301. 307. 314. 323.]
12156	0.1	0	0.1	0.5	[184. 194. 206. 215. 219. 230. 243. 247. 250. 256. 258.]
12215	0.3	0.3	0	0.9	[184. 196. 208. 213. 221. 229. 238. 240. 247. 248. 251.]

L

Data Analysis Laren

The table below presents the results of the data analysis conducted for the municipality of Laren. Each row corresponds to a single run in the model. A run consists of 30 fits comprising two splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

L.1. Results using the Random Search Algorithm

Table 53: Data analysis results of the municipality of Laren for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[10. 36. 64. 84. 93. 127. 152. 191. 284. 360. 584.]				
314	0.9	0.6	0.1	0.9	[10. 45. 84. 132. 189. 236. 290. 350. 404. 478. 545.]
456	0.3	0.3	0.4	0.6	[10. 53. 94. 153. 214. 275. 347. 430. 538. 641. 756.]
779	0.5	0.2	0.7	0.9	[10. 16. 25. 30. 38. 55. 69. 89. 108. 147. 178.]
826	0.2	0.7	0.5	0.8	[10. 70. 132. 195. 282. 375. 466. 554. 659. 777. 891.]
918	0.2	0.6	0.3	0.7	[10. 24. 35. 49. 59. 72. 83. 99. 109. 127. 143.]
989	0.7	0	0.2	0.6	[10. 11. 13. 21. 25. 36. 49. 65. 82. 98. 126.]
1091	0.1	0.1	0.9	0.9	[10. 16. 25. 32. 38. 47. 54. 67. 83. 98. 110.]
1564	0.5	0.9	0.1	0.7	[10. 116. 219. 320. 417. 517. 619. 727. 826. 915. 1029.]
1602	0.4	0.3	0.6	0.9	[10. 11. 14. 18. 20. 23. 23. 30. 35. 39. 44.]
1642	0.8	0.4	0	0.8	[10. 15. 19. 22. 26. 27. 31. 33. 36. 37. 39.]
1645	0.7	0.6	0	0.8	[10. 10. 12. 14. 19. 24. 27. 30. 31. 35. 38.]

1733	0.5	0	0.4	0.8	[10. 13. 16. 17. 19. 21. 21. 24. 25. 26. 30.]
1738	0.7	0.1	0.1	0.9	[10. 13. 18. 19. 21. 22. 25. 26. 27. 27. 27.]
1751	0.2	0.2	0.7	0.9	[10. 12. 14. 15. 16. 17. 18. 21. 22. 24. 27.]
1773	0	0.9	0	0.7	[10. 16. 18. 19. 20. 21. 21. 21. 22. 22. 22.]
1780	0	0.3	0.5	0.8	[10. 10. 12. 16. 16. 17. 19. 20. 21. 22. 22.]
1791	0.7	0.5	0	0.9	[10. 10. 12. 15. 16. 16. 19. 20. 20. 20. 20.]
1811	0	0.5	0.4	0.8	[10. 13. 15. 16. 17. 17. 17. 17. 17. 17. 17.]
1830	0.2	0.5	0.2	0.8	[10. 12. 12. 12. 12. 13. 14. 15. 15. 16. 16.]
1844	0.6	0	0	0.9	[10. 11. 11. 11. 11. 12. 13. 13. 13. 14. 14.]
1845	0	0.7	0.2	0.9	[10. 11. 11. 12. 12. 13. 13. 13. 13. 14. 14.]
1849	0	0.6	0.1	0.7	[10. 11. 11. 11. 12. 13. 13. 13. 13. 13. 13.]
1859	0.3	0	0.1	0.6	[10. 10. 11. 11. 11. 11. 12. 12. 12. 12. 12.]
2542	0	0.7	0.6	0.8	[10. 122. 232. 349. 479. 627. 747. 876. 1008. 1129. 1268.]
5735	0.5	0.9	0.6	0.9	[10. 163. 335. 505. 704. 903. 1136. 1330. 1533. 1775. 1988.]

L.2. Results using the Bayesian Search Algorithm

Table 54: Data analysis results of the municipality of Laren for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[10. 36. 64. 84. 93. 127. 152. 191. 284. 360. 584.]				
316	0.3	0.1	0.7	0.7	[10. 19. 28. 47. 67. 91. 118. 160. 199. 263. 357.]
336	0.4	0	0.8	0.8	[10. 12. 21. 34. 46. 71. 102. 147. 225. 315. 437.]
342	0.1	0.9	0.5	0.9	[10. 46. 88. 146. 214. 270. 322. 388. 438. 502. 567.]
583	0.4	0.2	0.8	0.9	[10. 13. 22. 32. 44. 59. 80. 105. 139. 183. 242.]
591	0.7	0	0.6	0.8	[10. 14. 16. 21. 27. 43. 66. 96. 133. 175. 244.]
690	0.5	0	0.1	0.4	[10. 15. 19. 25. 38. 56. 78. 103. 126. 151. 200.]
943	0	0.2	0.9	0.9	[10. 16. 32. 41. 45. 53. 66. 81. 103. 120. 137.]
989	0.6	0.2	0.1	0.5	[10. 42. 90. 166. 249. 346. 446. 584. 691. 826. 976.]
1026	0	0	0.6	0.5	[10. 17. 30. 56. 94. 146. 218. 346. 527. 822. 1155.]
1137	0.1	0.4	0.7	0.8	[10. 13. 21. 27. 38. 47. 51. 66. 79. 92. 103.]
1203	0.5	0	0.5	0.8	[10. 15. 21. 25. 31. 39. 50. 56. 67. 76. 94.]

1575	0.2	0	0.7	0.8	[10. 15. 17. 18. 21. 23. 28. 35. 41. 46. 47.]
1578	0.7	0.1	0.1	0.6	[10. 11. 12. 15. 16. 20. 25. 27. 30. 33. 46.]
1647	0.9	0.2	0.2	0.9	[10. 13. 15. 18. 20. 22. 24. 25. 29. 36. 40.]
1683	0.4	0.1	0.2	0.5	[10. 10. 14. 16. 19. 20. 23. 24. 27. 31. 36.]
1761	0.3	0.2	0.4	0.8	[10. 15. 15. 17. 18. 20. 21. 21. 23. 24. 26.]
1800	0.2	0.4	0.1	0.6	[10. 11. 14. 14. 14. 16. 17. 18. 18. 19. 20.]
1801	0.2	0.6	0.1	0.8	[10. 12. 14. 16. 16. 17. 17. 19. 19. 19. 19.]
1811	0.2	0	0.3	0.6	[10. 10. 11. 12. 14. 14. 16. 16. 17. 18. 18.]
1834	0.3	0.4	0	0.7	[10. 10. 11. 13. 14. 14. 14. 14. 14. 14. 15.]
1856	0.3	0.4	0	0.6	[10. 10. 10. 10. 11. 12. 12. 12. 12. 12. 12.]
2151	0.6	0.7	0	0.7	[10. 124. 246. 352. 476. 578. 692. 813. 941. 1056. 1172.]
2726	0.7	0.2	0.2	0.6	[10. 64. 148. 257. 391. 540. 694. 880. 1051. 1281. 1495.]
5111	0.4	0.5	0.5	0.7	[10. 134. 287. 447. 602. 809. 1000. 1228. 1474. 1732. 2011.]
17903	0.7	0.1	0.2	0.3	[10. 544. 1036. 1458. 1842. 2197. 2554. 2836. 3113. 3392. 3610.]

M

Data Analysis Oegstgeest

The table below presents the results of the data analysis conducted for the municipality of Oegstgeest. Each row corresponds to a single run in the model. A run consists of 30 fits comprising two splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

M.1. Results using the Random Search Algorithm

Table 55: Data analysis results of the municipality of Oegstgeest for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[134. 240. 346. 445. 545. 667. 902. 1407. 1971. 2461. 3135.]				
1139	0.3	0.4	0.5	0.6	[133. 304. 485. 709. 970. 1199. 1559. 1912. 2293. 2694. 3137.]
1376	0.1	0.3	0.5	0.5	[133. 278. 456. 691. 967. 1278. 1650. 2058. 2510. 2976. 3548.]
1462	0.3	0.9	0.3	0.5	[133. 394. 659. 945. 1219. 1486. 1775. 2061. 2318. 2622. 2902.]
1916	0.6	0	0.1	0.4	[133. 184. 258. 366. 494. 678. 917. 1180. 1477. 1801. 2181.]
2307	0.3	0.7	0.5	0.7	[133. 273. 442. 615. 792. 980. 1162. 1405. 1627. 1874. 2118.]
2431	0.4	0.8	0.4	0.7	[133. 281. 431. 618. 789. 979. 1177. 1382. 1600. 1851. 2063.]
2652	0.2	0	0.9	0.9	[133. 191. 270. 380. 497. 641. 798. 1001. 1271. 1547. 1878.]
2781	0.3	0.8	0.1	0.4	[133. 334. 533. 715. 888. 1060. 1242. 1412. 1595. 1749. 1911.]
3039	0	0.9	0.4	0.6	[133. 329. 531. 712. 889. 1067. 1231. 1404. 1534. 1672. 1796.]
3447	0.3	0	0.6	0.7	[133. 189. 261. 348. 457. 570. 715. 872. 1065. 1275. 1535.]
3471	0.3	0.5	0.5	0.7	[133. 221. 317. 438. 553. 698. 844. 1002. 1169. 1359. 1545.]

4043	0.4	0.5	0.4	0.7	[133. 224. 312. 414. 524. 636. 768. 906. 1050. 1208. 1370.]
4052	0	0.2	0.6	0.6	[133. 197. 276. 348. 451. 570. 703. 836. 972. 1151. 1352.]
4513	0.3	0.3	0.8	0.9	[133. 181. 229. 308. 378. 463. 581. 708. 864. 1035. 1214.]
6787	0.3	0.1	0.6	0.8	[133. 179. 215. 261. 323. 372. 417. 472. 537. 605. 671.]
7081	0.3	0.9	0.7	0.7	[133. 507. 961. 1407. 1939. 2549. 3185. 3793. 4399. 5031. 5640.]
7400	0.6	0.8	0.2	0.9	[133. 176. 219. 261. 301. 348. 408. 452. 490. 531. 563.]
7708	0.5	0.3	0.5	0.9	[133. 162. 194. 222. 259. 295. 341. 380. 420. 471. 519.]
8091	0.3	0.4	0.6	0.9	[133. 168. 211. 237. 270. 302. 348. 374. 402. 429. 461.]
8234	0.5	0.2	0.5	0.9	[133. 166. 192. 227. 259. 290. 320. 347. 375. 403. 445.]
9776	0.3	0.1	0.2	0.6	[133. 149. 164. 175. 194. 209. 214. 230. 238. 248. 257.]
9821	0	0.2	0.4	0.7	[133. 156. 182. 194. 206. 214. 227. 236. 242. 246. 251.]
24234	0.9	0.8	0.3	0.1	[133. 1374. 2496. 3433. 4251. 5020. 5737. 6350. 6844. 7319. 7716.]
24335	0.6	0.4	0.8	0.7	[133. 1125. 2109. 3075. 3954. 4836. 5615. 6372. 7046. 7658. 8241.]
26904	0.2	0	0.3	0.3	[133. 1253. 2344. 3313. 4265. 5152. 5975. 6751. 7408. 8038. 8508.]

M.2. Results using the Bayesian Search Algorithm

Table 56: Data analysis results of the municipality of Oegstgeest for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[134. 240. 346. 445. 545. 667. 902. 1407. 1971. 2461. 3135.]				
904	0.7	0.2	0.5	0.8	[133. 241. 389. 556. 759. 1003. 1331. 1631. 2038. 2453. 2911.]
1458	0.4	0.1	0.2	0.4	[133. 227. 345. 485. 667. 888. 1144. 1438. 1776. 2158. 2553.]
1594	0.6	0.7	0.4	0.7	[133. 364. 642. 898. 1211. 1536. 1880. 2269. 2685. 3096. 3525.]
1697	0	0.7	0.5	0.6	[133. 366. 622. 874. 1137. 1386. 1611. 1872. 2095. 2353. 2586.]
1972	0.3	0.9	0	0.3	[133. 601. 1053. 1438. 1814. 2159. 2446. 2725. 2955. 3203. 3454.]
2041	0.9	0.3	0.4	0.9	[133. 228. 327. 445. 613. 761. 967. 1218. 1494. 1789. 2136.]
2266	0.2	0.4	0.5	0.6	[133. 243. 359. 510. 653. 830. 1014. 1207. 1475. 1754. 2065.]
3816	0.3	0.9	0.6	0.9	[133. 258. 363. 478. 606. 741. 888. 1024. 1186. 1321. 1466.]
3889	0.5	0.9	0.3	0.5	[133. 496. 904. 1304. 1733. 2172. 2639. 3087. 3555. 4013. 4427.]
4151	0.6	0.7	0.2	0.7	[133. 249. 365. 479. 594. 713. 840. 958. 1101. 1230. 1356.]
4320	0.8	0	0.2	0.7	[133. 177. 230. 313. 400. 491. 603. 747. 911. 1083. 1290.]

4755	0.7	0.1	0.5	0.9	[133. 170. 223. 278. 366. 473. 586. 682. 816. 957. 1139.]
4885	0	0.9	0.4	0.7	[133. 271. 405. 522. 638. 734. 851. 946. 1030. 1094. 1155.]
4907	0	0.9	0.4	0.7	[133. 286. 426. 560. 682. 777. 865. 955. 1026. 1094. 1151.]
4971	0	0.7	0.6	0.8	[133. 245. 334. 434. 552. 651. 767. 850. 948. 1013. 1080.]
5033	0.3	0.7	0.5	0.8	[133. 209. 303. 399. 498. 600. 727. 806. 910. 1002. 1107.]
5299	0	0.9	0.6	0.6	[133. 520. 935. 1399. 1925. 2407. 2903. 3417. 3950. 4512. 5058.]
5960	0.9	0.7	0.1	0.9	[133. 204. 273. 342. 402. 481. 550. 623. 705. 780. 844.]
6350	0.5	0.9	0.3	0.9	[133. 193. 259. 313. 382. 450. 506. 569. 633. 688. 757.]
7212	0.3	0.7	0.5	0.9	[133. 186. 238. 293. 332. 376. 418. 469. 518. 560. 596.]
8252	0.2	0.4	0.9	0.8	[133. 424. 746. 1104. 1573. 2155. 2860. 3624. 4461. 5272. 6077.]
10090	0.1	0.3	0.3	0.6	[133. 149. 164. 174. 180. 192. 200. 205. 209. 220. 227.]
10167	0.5	0.7	0	0.9	[133. 149. 165. 173. 180. 191. 199. 203. 207. 210. 212.]
10170	0.3	0.6	0.2	0.8	[133. 141. 151. 170. 177. 189. 196. 201. 205. 206. 209.]
12977	0	0	0.6	0.5	[133. 307. 579. 974. 1555. 2363. 3359. 4468. 5617. 6638. 7497.]

N

Data Analysis Vaals

The table below presents the results of the data analysis conducted for the Vaals municipality. Each row corresponds to a single run in the model. A run consists of 30 fits comprising two splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

N.1. Results using the Random Search Algorithm

Table 57: Data analysis results of the municipality of Vaals for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[0. 108. 173. 217. 255. 439. 534. 734. 916. 1073. 1238.]				
320	0.9	0.3	0.3	0.8	[0. 62. 130. 219. 307. 412. 548. 689. 863. 1048. 1251.]
409	0.3	0.6	0.4	0.6	[0. 92. 178. 281. 391. 533. 688. 818. 971. 1123. 1277.]
410	0.8	0.3	0	0.6	[0. 51. 126. 202. 298. 395. 516. 666. 849. 1060. 1282.]
544	0.8	0.6	0.4	0.9	[0. 67. 148. 236. 328. 450. 570. 683. 813. 960. 1119.]
760	0.6	0.6	0.4	0.8	[0. 61. 139. 219. 301. 384. 486. 589. 718. 843. 963.]
1050	0.2	0.5	0.3	0.5	[0. 60. 138. 219. 288. 371. 447. 527. 645. 726. 825.]
1063	0.4	0.5	0.3	0.6	[0. 56. 120. 198. 272. 342. 434. 513. 620. 717. 835.]
1252	0.9	0.6	0.1	0.7	[0. 118. 233. 360. 500. 671. 858. 1061. 1302. 1520. 1795.]
1999	0.6	0.7	0	0.7	[0. 77. 142. 190. 248. 306. 356. 399. 442. 517. 563.]
2006	0.2	0.9	0.2	0.6	[0. 69. 129. 174. 233. 295. 348. 394. 466. 517. 563.]
2035	0.5	0.7	0.2	0.7	[0. 43. 82. 130. 194. 246. 310. 369. 436. 493. 559.]

2160	0.9	0.8	0.1	0.7	[0. 146. 307. 494. 685. 868. 1085. 1331. 1561. 1786. 2042.]
2331	0.3	0.2	0.2	0.4	[0. 24. 53. 87. 123. 177. 221. 279. 331. 409. 483.]
4786	0.9	0.9	0.3	0.7	[0. 221. 451. 702. 969. 1241. 1518. 1801. 2119. 2380. 2677.]
5687	0.8	0.5	0	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.5	0.8	0	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.7	0	0.5	0.8	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0	0.7	0.2	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.1	0	0	0.2	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.2	0	0	0.5	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.8	0	0	0.6	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0	0.1	0.1	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.3	0	0.1	0.3	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
9531	0.8	0.8	0.5	0.7	[0. 340. 687. 1084. 1454. 1811. 2170. 2500. 2807. 3095. 3399.]
14994	0.3	0.2	0.7	0.6	[0. 442. 883. 1377. 1866. 2378. 2860. 3302. 3671. 4007. 4310.]

N.2. Results using the Bayesian Search Algorithm

Table 58: Data analysis results of the municipality of Vaals for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[0. 108. 173. 217. 255. 439. 534. 734. 916. 1073. 1238.]				
608	0.7	0.6	0.2	0.7	[0. 82. 157. 241. 338. 433. 517. 628. 756. 879. 1014.]
621	0.9	0.5	0.2	0.8	[0. 55. 134. 201. 290. 403. 498. 613. 748. 880. 1017.]
704	0.7	0.8	0.1	0.7	[0. 79. 152. 228. 312. 400. 497. 620. 734. 853. 975.]
1194	0.4	0.7	0.1	0.4	[0. 148. 312. 480. 620. 808. 961. 1144. 1301. 1462. 1637.]
1572	0	0.9	0.4	0.7	[0. 72. 157. 233. 309. 377. 436. 509. 570. 627. 670.]
1914	0.7	0.4	0.2	0.7	[0. 45. 81. 131. 186. 240. 300. 373. 440. 508. 579.]
1970	0.6	0.5	0	0.6	[0. 49. 87. 140. 194. 246. 302. 378. 437. 498. 568.]
2350	0.1	0.9	0.2	0.6	[0. 58. 113. 177. 229. 268. 312. 363. 409. 452. 481.]
2379	0.7	0.5	0.4	0.9	[0. 29. 55. 100. 143. 192. 240. 295. 357. 412. 485.]
2565	0.2	0.5	0.5	0.7	[0. 30. 63. 105. 154. 208. 248. 299. 347. 398. 444.]
3156	0.1	0.7	0.4	0.7	[0. 30. 56. 82. 117. 148. 187. 226. 262. 291. 331.]

3794	0.3	0.9	0	0.7	[0. 29. 60. 89. 114. 137. 162. 178. 200. 215. 223.]
3958	0.2	0.8	0	0.6	[0. 31. 65. 96. 118. 144. 166. 176. 185. 195. 200.]
4654	0.4	0.8	0.5	0.6	[0. 205. 433. 707. 991. 1275. 1531. 1798. 2061. 2338. 2597.]
5474	0	0.3	0.9	0.8	[0. 0. 0. 3. 4. 5. 6. 12. 16. 26. 31.]
5662	0.1	0.8	0.5	0.5	[0. 240. 483. 774. 1069. 1349. 1647. 1937. 2253. 2559. 2851.]
5687	0.5	0	0.1	0.8	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0	0	0.7	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.3	0.1	0.3	0.9	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.1	0.3	0	0.5	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.2	0.1	0.6	0.8	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.2	0.4	0.2	0.6	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.9	0.2	0.2	0.8	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
5687	0.3	0	0.2	0.7	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
11919	0.8	0.8	0.2	0.4	[0. 424. 848. 1278. 1722. 2076. 2498. 2834. 3158. 3431. 3697.]

O

Data Analysis Westerveld

The table below presents the results of the data analysis conducted for the Westerveld municipality. Each row corresponds to a single run in the model. A run consists of 30 fits comprising two splits and 15 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasing the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

O.1. Results using the Random Search Algorithm

Table 59: Data analysis results of the municipality of Westerveld for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[141. 296. 427. 920. 1138. 1706. 2263. 2717. 3045. 3440. 4023.]				
1075	0.4	0.4	0.6	0.8	[142. 369. 620. 939. 1264. 1608. 2035. 2510. 3013. 3469. 3954.]
1401	0.6	0.4	0.4	0.7	[142. 464. 786. 1167. 1591. 2038. 2515. 2989. 3435. 3909. 4352.]
1549	0.4	0.8	0	0.6	[142. 561. 990. 1396. 1750. 2114. 2422. 2764. 3061. 3347. 3599.]
1606	0.8	0.3	0.2	0.7	[142. 363. 599. 862. 1182. 1515. 1918. 2345. 2778. 3180. 3618.]
1651	0.1	0.4	0.1	0.2	[142. 688. 1208. 1708. 2191. 2608. 3014. 3395. 3762. 4068. 4391.]
1736	0.6	0.1	0	0.4	[142. 234. 364. 577. 855. 1238. 1729. 2349. 3062. 3770. 4517.]
1812	0.1	0.6	0.8	0.9	[142. 386. 704. 1062. 1436. 1863. 2323. 2851. 3432. 3970. 4535.]
1908	0.8	0.7	0.2	0.8	[142. 417. 739. 1054. 1385. 1731. 2086. 2461. 2831. 3212. 3607.]
2242	0.2	0.5	0.5	0.7	[142. 370. 591. 839. 1122. 1438. 1767. 2097. 2450. 2844. 3260.]
2366	0.9	0.4	0.5	0.9	[142. 535. 939. 1367. 1840. 2336. 2855. 3354. 3820. 4307. 4770.]
2647	0.8	0.3	0.6	0.9	[142. 531. 943. 1411. 1947. 2443. 2946. 3438. 3974. 4446. 4892.]

2722	0.2	0.5	0.6	0.8	[142. 325. 519. 732. 973. 1237. 1519. 1815. 2185. 2567. 2924.]
3292	0.9	0.6	0.3	0.7	[142. 628. 1150. 1673. 2216. 2747. 3243. 3750. 4258. 4736. 5151.]
4125	0.8	0.7	0.1	0.6	[142. 651. 1163. 1730. 2285. 2838. 3440. 4007. 4537. 5041. 5579.]
4419	0	0.6	0.2	0.5	[142. 412. 672. 926. 1183. 1409. 1636. 1880. 2111. 2317. 2485.]
4441	0.6	0.2	0.7	0.8	[142. 617. 1150. 1721. 2316. 2915. 3492. 4035. 4606. 5179. 5678.]
4902	0.2	0.9	0.5	0.6	[142. 750. 1332. 1913. 2535. 3097. 3611. 4118. 4606. 5076. 5506.]
5005	0.4	0.7	0.5	0.6	[142. 682. 1226. 1816. 2380. 2951. 3547. 4121. 4686. 5196. 5696.]
6660	0.5	0.6	0.4	0.5	[142. 735. 1359. 1978. 2610. 3225. 3835. 4428. 4948. 5440. 5851.]
6831	0.2	0.9	0.4	0.9	[142. 280. 446. 617. 791. 954. 1145. 1347. 1516. 1719. 1893.]
9271	0.9	0.8	0.4	0.4	[142. 1046. 1885. 2647. 3381. 4024. 4602. 5165. 5595. 5988. 6355.]
12750	0.2	0.3	0.7	0.9	[142. 193. 242. 305. 375. 453. 518. 603. 702. 807. 909.]
12840	0.6	0.1	0.2	0.6	[142. 189. 243. 289. 347. 413. 495. 572. 677. 764. 893.]
13186	0	0.7	0.6	0.9	[142. 191. 249. 327. 394. 456. 527. 609. 686. 765. 848.]
17942	0.2	0.6	0	0.8	[142. 168. 180. 192. 198. 204. 208. 210. 216. 217. 218.]

O.2. Results using the Bayesian Search Algorithm

Table 60: Data analysis results of the municipality of Westerveld for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[141. 296. 427. 920. 1138. 1706. 2263. 2717. 3045. 3440. 4023.]				
1011	0.5	0.9	0.4	0.8	[142. 485. 802. 1164. 1534. 1919. 2303. 2706. 3070. 3501. 3891.]
1262	0.7	0.7	0.2	0.7	[142. 492. 823. 1188. 1554. 1976. 2357. 2772. 3188. 3600. 4002.]
1338	0.7	0.4	0.6	0.9	[142. 453. 825. 1205. 1625. 2075. 2543. 3021. 3476. 3942. 4434.]
1376	0.7	0.8	0.2	0.7	[142. 497. 854. 1245. 1606. 2017. 2425. 2837. 3225. 3609. 4022.]
1865	0.3	0.8	0.2	0.4	[142. 699. 1164. 1669. 2174. 2625. 3032. 3428. 3863. 4241. 4593.]
2161	0.4	0.5	0	0.5	[142. 415. 730. 1029. 1348. 1667. 2007. 2329. 2644. 2958. 3232.]
2168	0.6	0.3	0	0.5	[142. 435. 781. 1180. 1563. 2056. 2553. 3104. 3677. 4262. 4813.]
2545	0.1	0.9	0.1	0.5	[142. 563. 969. 1333. 1665. 1959. 2221. 2469. 2689. 2888. 3090.]
3047	0.8	0.8	0.1	0.9	[142. 366. 592. 856. 1117. 1388. 1659. 1951. 2238. 2528. 2846.]
3450	0.6	0.5	0.2	0.7	[142. 337. 552. 793. 1027. 1314. 1584. 1835. 2126. 2421. 2732.]
3953	0.6	0.6	0.4	0.6	[142. 649. 1190. 1740. 2322. 2899. 3432. 3980. 4479. 4958. 5414.]

4792	0.4	0.7	0.6	0.7	[142. 620. 1122. 1709. 2309. 2924. 3566. 4163. 4706. 5232. 5714.]
4825	0.1	0.2	0.8	0.8	[52. 208. 305. 453. 626. 804. 990. 1247. 1544. 1899. 2363.]
4863	0.2	0.1	0.4	0.4	[142. 563. 1057. 1601. 2174. 2797. 3439. 4094. 4716. 5276. 5837.]
5148	0.4	0.8	0.4	0.5	[142. 714. 1278. 1921. 2527. 3079. 3639. 4170. 4687. 5133. 5562.]
6956	0.5	0.7	0.2	0.8	[142. 302. 443. 618. 788. 955. 1141. 1316. 1502. 1694. 1862.]
8198	0.1	0.4	0.7	0.8	[142. 226. 316. 424. 525. 649. 804. 973. 1161. 1388. 1642.]
10382	0.4	0	0.5	0.7	[142. 197. 261. 321. 398. 483. 582. 704. 868. 1033. 1257.]
12726	0.4	0.3	0.5	0.8	[142. 198. 246. 315. 375. 443. 513. 596. 690. 802. 914.]
13082	0.8	0.7	0.9	0.5	[142. 1123. 2059. 2974. 3760. 4480. 5148. 5755. 6235. 6683. 7056.]
13381	0.3	0.1	0.6	0.8	[142. 209. 243. 285. 333. 401. 469. 549. 618. 717. 815.]
15910	0.7	0.5	0	0.9	[142. 192. 236. 263. 293. 325. 353. 381. 415. 441. 456.]
16900	0.4	0.1	0.1	0.5	[142. 167. 187. 210. 226. 240. 261. 279. 297. 319. 347.]
17979	0.2	0.1	0.1	0.5	[142. 147. 156. 173. 176. 185. 191. 196. 205. 213. 219.]
18129	0	0.2	0.1	0.5	[142. 154. 167. 177. 182. 185. 187. 190. 191. 192. 194.]

Mean	7098
Median	4825
Standard deviation	5746

P

Laren Increased Splits and Iterations

The table below presents the results of the data analysis conducted for the Laren municipality. Each row corresponds to a single run in the model. A run consists of 150 fits comprising five splits and 30 iterations each. The values provided in each row represent the outcome of the best-performing fit within that run. Consequently, the lowest error values in the table signify the highest fitness. In summary, each row contains the optimal performance achieved in a single run, showcasting the predictions generated by this best fit, the associated parameter values and the error value that comes forth from these predictions.

P.1. Results using the Random Search Algorithm

Table 61: Data analysis results of the municipality of Laren for the Random Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[10. 36. 64. 84. 93. 127. 152. 191. 284. 360. 584.]				
207	0.4	0	0.7	0.7	[10. 14. 23. 34. 56. 92. 142. 223. 311. 426. 563.]
343	0.3	0.6	0.4	0.8	[10. 43. 72. 95. 125. 167. 202. 237. 281. 333. 373.]
390	0.8	0	0.3	0.7	[10. 16. 25. 36. 50. 65. 101. 141. 190. 249. 336.]
448	0.1	0.6	0.4	0.7	[10. 39. 68. 93. 125. 152. 177. 201. 240. 260. 303.]
450	0.1	0.5	0.2	0.5	[10. 44. 70. 104. 144. 163. 193. 215. 247. 282. 313.]
512	0.6	0.7	0.2	0.9	[10. 31. 47. 77. 98. 127. 154. 178. 207. 239. 271.]
550	0.7	0	0.6	0.8	[10. 19. 29. 35. 49. 66. 91. 112. 141. 199. 251.]
552	0	0	0.9	0.8	[10. 15. 27. 35. 46. 62. 77. 107. 142. 190. 250.]
647	0.7	0	0.6	0.9	[10. 14. 19. 32. 44. 59. 77. 100. 135. 172. 223.]
697	0.7	0.5	0.3	0.9	[10. 34. 49. 61. 76. 96. 112. 137. 154. 178. 200.]
767	0.5	0	0.4	0.6	[10. 14. 22. 26. 33. 46. 61. 79. 109. 133. 185.]

850	0.7	0.1	0	0.5	[10. 12. 19. 27. 30. 41. 55. 68. 89. 117. 160.]
979	0.6	0.2	0.5	0.8	[10. 13. 17. 22. 29. 38. 46. 59. 79. 97. 130.]
983	0.7	0.7	0.1	0.9	[10. 20. 31. 39. 54. 63. 73. 89. 101. 116. 130.]
1112	0.5	0.2	0.4	0.7	[10. 15. 22. 27. 35. 44. 52. 66. 74. 84. 109.]
1245	0.7	0	0.1	0.6	[10. 19. 27. 33. 39. 44. 50. 56. 64. 75. 87.]
1312	0.9	0.4	0	0.8	[10. 15. 22. 23. 28. 38. 45. 54. 67. 72. 79.]
1400	0.6	0.2	0.3	0.7	[10. 14. 18. 21. 24. 26. 29. 36. 47. 58. 68.]
1579	0.9	0	0.1	0.8	[10. 12. 15. 16. 22. 27. 30. 33. 37. 37. 45.]
1625	0.2	0.5	0	0.5	[10. 11. 15. 21. 24. 29. 30. 33. 35. 36. 40.]
1634	0.1	0.4	0.6	0.8	[10. 11. 12. 15. 19. 22. 24. 26. 28. 31. 41.]
1707	0.3	0.2	0.7	0.9	[10. 13. 15. 21. 22. 24. 26. 28. 29. 30. 32.]
1789	0.1	0.3	0	0.3	[10. 11. 15. 17. 18. 19. 19. 19. 20. 20. 20.]
1813	0	0.5	0.4	0.7	[10. 11. 12. 13. 15. 17. 17. 17. 17. 17. 17.]
1816	0	0.4	0.3	0.6	[10. 10. 11. 11. 13. 14. 16. 16. 16. 17. 17.]

P.2. Results using the Bayesian Search Algorithm

Table 62: Data analysis results of the municipality of Laren for the Bayesian Search algorithm

Error	Weight_eco	Weight_env	Weight_cof	Weight_soc	Predictions
Real Data	[10. 36. 64. 84. 93. 127. 152. 191. 284. 360. 584.]				
304	0.5	0	0.9	0.9	[10. 22. 34. 50. 64. 91. 112. 147. 196. 274. 356.]
307	0.7	0.1	0.5	0.8	[10. 16. 27. 40. 61. 86. 109. 151. 217. 291. 371.]
324	0.3	0	0	0.2	[10. 20. 34. 51. 78. 116. 174. 245. 343. 477. 649.]
326	0.5	0.7	0.3	0.9	[10. 32. 61. 90. 120. 156. 197. 243. 282. 324. 370.]
333	0.5	0	0.6	0.7	[10. 16. 25. 48. 68. 95. 127. 160. 228. 302. 389.]
558	0.3	0.8	0.2	0.8	[10. 34. 63. 90. 116. 135. 154. 175. 197. 227. 249.]
624	0.7	0	0.4	0.7	[10. 15. 22. 29. 35. 51. 63. 86. 133. 172. 232.]
670	0.4	0	0.6	0.7	[10. 17. 27. 32. 43. 60. 81. 102. 125. 172. 214.]
731	0.4	0.8	0.1	0.8	[10. 34. 47. 62. 82. 95. 116. 128. 146. 159. 183.]
745	0.5	0.2	0.7	0.9	[10. 17. 20. 28. 39. 51. 70. 93. 117. 157. 186.]
748	0.7	0	0.1	0.5	[10. 12. 14. 17. 24. 34. 47. 59. 87. 130. 187.]

855	0.8	0	0.1	0.6	[10. 16. 24. 30. 33. 36. 53. 66. 86. 117. 157.]
907	0.7	0.1	0.5	0.8	[10. 11. 14. 21. 29. 36. 49. 67. 94. 116. 145.]
921	0.1	0.3	0.9	0.9	[10. 14. 21. 29. 38. 47. 54. 71. 95. 117. 144.]
938	0.7	0	0.6	0.9	[10. 12. 17. 23. 36. 41. 50. 62. 78. 98. 140.]
1123	0.6	0.2	0.3	0.7	[10. 17. 19. 24. 33. 42. 52. 57. 72. 87. 108.]
1364	0.2	0.1	0.5	0.6	[10. 11. 14. 19. 26. 27. 33. 41. 50. 62. 71.]
1415	0.3	0.1	0.4	0.6	[10. 13. 18. 29. 30. 31. 39. 43. 49. 56. 65.]
1444	0.5	0.2	0.5	0.8	[10. 16. 19. 21. 26. 27. 33. 38. 48. 53. 63.]
1696	0.9	0.4	0	0.8	[10. 13. 16. 18. 20. 21. 24. 26. 30. 33. 34.]
1704	0.6	0.1	0.4	0.9	[10. 11. 12. 13. 17. 20. 22. 23. 27. 30. 32.]
1712	0	0	0.4	0.5	[10. 13. 14. 19. 20. 21. 25. 26. 27. 30. 32.]
1714	0	0.6	0.4	0.7	[10. 12. 16. 21. 24. 26. 26. 28. 28. 29. 29.]
1731	0	0.6	0.3	0.6	[10. 12. 15. 16. 17. 19. 20. 22. 25. 26. 30.]
1782	0.5	0.1	0.2	0.9	[10. 15. 16. 17. 17. 20. 21. 21. 21. 21. 22.]

Q

Inter-Municipal Analysis

Table 63: Mean, median and standard deviations of the results for the Random Search algorithm

Random Search Models			
	Mean	**Median**	**Standard deviation**
Bloemendaal	3458	3090	1933
Dantumadiel	6497	5327	3071
Laren	1668	1738	974
Oegstgeest	7160	4052	7152
Vaals	3873	2331	3358
Westerveld	5135	3292	4485
Average	4632	3305	3496

Table 64: Mean, median and standard deviations of the results for the Bayesian Search algorithm

Bayesian Search Models			
	Mean	**Median**	**Standard deviation**
Bloemendaal	8609	6057	7827
Dantumadiel	6872	5694	3754
Laren	2136	1578	3359
Oegstgeest	5165	4885	3109
Vaals	4000	3958	2490
Westerveld	7098	4825	5746
Average	5647	4500	4381