The Effect of Domain Shift on Learning Curve Extrapolation

Max Soeters[1]
Supervisor(s): Tom Viering[1]**, Cheng Yan**[1]**, Sayak Mukherjee**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Max Soeters
Final project course: CSE3000 Research Project
Thesis committee: Tom Viering, Cheng Yan, Sayak Mukherjee, Matthijs Spaan

Abstract

Domain shift is when the distribution of data differs between the training of a model and its testing. This can happen when the conditions of training are slightly different from the conditions that will happen when a model is tested or used. This is a problem for generalizability of a model. Learning curves are widely used in machine learning to predict how much data is needed when training a model. This paper will explore how domain shift impacts learning curve extrapolation using Learning Curve Prior Fitted Networks. We will explore the effect of domain shift on the performance of models while comparing different learners and groups of learners, thereby showing that domain shift is relevant to learning curve extrapolation and has a statistically significant impact on the accuracy of such extrapolations. We will also discuss how patterns like well-behavedness have an impact on this effect of domain shift, while also showing that is it not the full solution to predicting the effect.

1 Introduction

How much data do you need to train a model? This is a question that needs to be answered when you want to train a model to accomplish a machine learning task. Sample wise learning curves portray the expected performance of the model based on the amount of data used during training. They are different from epoch wise learning curves, which show performance of a model during training.

Sample wise learning curves are a necessary tool to estimate how much data you need to successfully complete training a model to a certain prediction accuracy. This is supported by the fact that Dimensional research (2019) found that 51% of companies have problems with not having enough data when training their models. In other cases data can be expensive or time consuming to acquire, which means learning curves are useful to know how much time or money must be spent (Viering and Loog, 2022).

To find how much data is needed for a model during training, we can use curve extrapolation. This is a technique where we use the known points of a learning curve to predict the rest of the curve. Using that we can see how much data we would need to train until the wanted accuracy is reached (Mohr and Rijn, 2024).

To extrapolate learning curves, we use Learning Curve Prior Fitted Networks (LCPFN). These networks use Bayesian inference to extrapolate learning curves (Adriaensen et al., 2023). Viering et al. (2024) later modified this technique to work for sample wise learning curves and based on real data, and verified it to perform better than previous versions of LCPFNs.

For previous extrapolation methods it has been shown that taking the correct machine learning model into account is important if you want an accurate estimate of how much data is needed for training (Viering and Loog, 2022). Therefore it is a relevant question if taking the correct machine learning model into account is also important when extrapolating learning curves using LCPFNs.

To find if it is important to take the correct model into account for curve extrapolation using LCPFNs, we will study the effect of domain shift. Domain shift is when the distribution of data differs between training and testing of models. This means that we will train a model on curves of one type of machine learning algorithm, learner A, and evaluate it on curves from another, learner B. If learner A and B were to have similar curves, the predictions should be relatively accurate when this process happens. If the curves of A and B are not very similar, there should be a large error when evaluating on the other group.

The aim of this paper is to answer the following research question: What is the effect of domain transfer on learning curve extrapolation?

This can be broken down into two sub-questions:

1. Is there a trend between groups of learners in the effect of the domain transfer?
2. Does domain transfer over single learners impact the accuracy of PFNs?

2 Related Work

The following section will cover the background of this research and explain Learning curves, LCPFNs, Domain Shift and the Learning Curve DataBase.

Learning curves

Learning curves plot the performance of a machine learning model as a function of a resource used during the process of training. (Mohr and Rijn, 2024) Some examples of resources are the amount of training data or the amount of time spent training or iterations. For this paper we will look specifically into sample wise curves, which compare the performance of the model to the amount of data used for training. (Viering et al., 2024) Mentions of learning curves throughout this work mean sample wise learning curves unless otherwise specified.

When talking about learning curves, we are interested in the true learning curve. However these are impossible to attain in practice. Therefore we have to rely on empirical learning curves. Empirical learning curves are sets of estimates of the true curve at different points along the curve, called anchors. The estimates are then averaged leads to the empirical curve. (Mohr et al., 2023)

Learning curves are sometimes ill-behaved, as described by Viering and Loog (2022). While one would expect error to always decrease with more data, learning curves sometimes peak and dip, meaning that the learning curves are not monotone and convex. Monotone is when a model always improves when more data is added. Convex means that the more data we already have, the less a model will improve from adding a set amount of extra training data. (Yan et al., 2025) Peaking is when there is suddenly a maximum in the error when more data gets added, after which the error goes down again. Dipping is described as the performance deteriorating when adding more samples after a certain point, after which it never recovers to the same performance again. (Viering and Loog, 2022)

LCPFN

Learning Curve Prior Fitted Networks were designed to extrapolate learning curves using a Bayesian approach, meaning it makes use of Bayes' theorem for extrapolation of the curves. The intrinsically hard to predict behaviour that some curves portray makes this Bayesian approach show great potential. However, previous Bayesian approaches were either very restrictive or computationally expensive. (Adriaensen et al., 2023) The PFN presented in that work was shown to be way faster than previous methods.

Viering et al. (2024) extends the PFNs to also work for sample size curves. That paper verifies that the PFNs perform better in most cases in comparison to parametric model based priors that can't deal with deviating behaviour like peaking and dipping. The models were trained using real curves from the Learning Curve Database, which were fed into the model. An important part of training their models was the use of data augmentation. This is a technique used to prevent overfitting in models. They apply random linear transformations to slightly alter the curves fed into the model while training.

Domain Shift

Domain shift is a challenge in machine learning where the distribution of data is not the same between training and testing. This means that the environment, conditions, or features on can differ between training and the target task. (Wang et al., 2025) This is a big problem for the generalizability of machine learning models.

An example of the impact of domain shift is in self-driving cars, where models that perform great during daytime struggle during nighttime (Salem, 2024). Domain shift being an issue was also highlighted during the 2011 Fukushima nuclear disaster, where models failed to account for composite disasters because they were trained on historical data. This lead to an underestimate of risk, and thus inefficient safety measures were taken. This is how domain shift can impact models and lead to inaccurate predictions, sometimes with big consequences. (Wang et al., 2025)

There are lots of techniques being explored to try and avoid this problem, most prominently domain adaptation. (Wu et al., 2023) Domain adaptation techniques try to handle the mismatch between the source domains and target domain (Sun et al., 2015). They try to adapt at either feature level, instance level or model level to get closer to the target domain. (Wang et al., 2025)

This shows that domain shift is a very relevant problem currently, and understanding the effect it has with the domain of interest for a project can help give insight in considerations that need to be taken when

training a model.

Learning Curve Database

In this research project we will use the Learning Curve Database (LCDB).

Mohr et al. (2023) first introduced the LCDB 1.0 to make help deepen our knowledge of learning curves. It is meant to be able to be used as a tool for selection of classifier, as it focusses on supervised learners. It contains multiple splits of data from which the empirical learning curve can be derived. It contains learning curves from 20 classifiers for 246 datasets. The datasets are largely from AutoML or published benchmarks to assure reproducibility. This work also compiles statistics on behaviours like monoticity, convexity, and on if learners will cross (meaning learner A starts worse than learner B, but ends better). A figure in that report that describes the crossing in detail is figure 3, which will be referenced as Mohr's figure below.

Yan et al. (2025) improves on the LCDB by publishing version 1.1. This version fixes some missing curves in LCBD 1.0, which is briefly mentioned. The paper further focusses on statistics about the curves and showing that the curves are more ill-behaved than previously thought. It states that around 14% of all curves are ill-behaved, but it varies which learner's curves show this behaviour. This is illustrated in figure 1, which is a coloured version of table 6 from that paper about the full database without feature scaling, which is the specific version of the LCDB 1.1 used in this paper. It is coloured in to better show patterns and assist in showing differences between the groups of learners that we have created based on mechanism of learning and on the SciKitLearn library (1. Supervised Learning, n.d.).

	Missing	Flat	Monotone	Convex	Mono & Conv	Peaking	Dipping
SVM_Linear	0.38	3.4	93.21	92.83	91.7	1.51	2.64
SVM_Poly	0.38	17.36	80.75	79.62	78.49	0.38	1.89
SVM_RBF	0.38	18.49	80	73.58	73.21	0	0.38
SVM_Sigmoid	0.38	19.25	36.98	49.81	32.45	23.4	42.26
Decision Tree	0.38	4.53	94.72	94.34	94.34	0.38	0.75
ExtraTree	0.38	3.77	94.72	95.85	94.72	0	0.38
ens.ExtraTrees	0.38	9.06	88.68	89.06	87.92	0.75	1.89
ens.RandomForest	0.38	9.06	89.06	89.43	88.3	0.38	1.13
ens.GradientBoosting	0.38	3.4	95.09	96.23	95.09	0	0.75
LogisticRegression	0.38	6.79	91.32	89.06	88.68	1.13	1.51
PassiveAggressive	0.38	5.66	86.42	90.19	85.66	1.89	3.4
Perceptron	0.38	3.02	94.34	95.47	93.96	0.75	1.51
RidgeClassifier	0.38	7.17	78.11	77.74	76.23	10.94	4.91
SGDClassifier	0.38	2.26	95.09	96.98	95.09	0.38	2.26
MLP	0.38	4.91	74.34	73.58	67.55	9.81	3.77
LDA	0.38	3.77	63.77	63.02	58.87	24.53	6.42
QDA	0.38	3.77	63.02	60	51.32	19.62	26.79
BernoulliNB	0.38	26.42	66.79	62.64	60.38	4.91	5.66
MultinomialNB	30.57	9.06	56.98	55.09	54.72	3.02	4.53
ComplementNB	30.57	8.3	55.09	56.23	54.34	4.15	6.79
GaussianNB	0.38	4.53	73.21	80.38	71.32	12.83	24.53
KNN	0.38	10.94	87.17	87.55	86.79	0.75	2.26
NearestCentroid	0.38	10.94	81.51	84.15	81.13	4.15	11.32
DummyClassifier	0.38	69.06	28.3	27.17	26.42	3.02	0

Figure 1: Coloured version of curve statistics table 6 (LCDB1.1 Full no FS) from Yan et al. (2025)

From the statistics provided, these are some notable facts. One learner that stands out in Mohr's figure in the SVC group is the sigmoid, which has a large probability to be overtaken by the other learners. In figure 1 this learner also deviates from the other SVCs. This may make the PFN less accurate as one type deviates a lot from the rest.

For the tree based learners, they seem to have similar percentages of monotony and convexness, with little peaking and dipping throughout all of them. This will probably lead to a good score for this PFN. However, apart from one learner the perceptron, SGD Classifier, KNN, and the SVM_Linear this behaviour is not seen in other learners. This could lead to the PFN trained on trees to perform badly on most subgroups as the behaviour distribution is only seen in 4 other learners.

QDA does start off worse than almost all other classifiers, which could make it a tad more difficult. Apart from those behaviours the groups seem pretty similar so I think that the domain shift between these groups could be smaller. While these features in the DA classifiers might share a few features with others, there is a lot of peaking and dipping which might make it hard to predict in general.

When comparing the naïve bayes classifiers to the DA classifiers, QDA seems to have a relatively similar distribution to the gaussianNB when you look at peaking and dipping, and LDA seems similar to the others, except for the peaking. This means that they might be similarly distributed groups.

3 Methodology

This section broadly outlines the experiments that are done in this thesis. A more in-depth description can be found in the experimental setup chapter.

3.1 Experiment 1: Comparing subgroups of Learners

The first experiment looks at the first research question to see if there is a trend in the subgroups of learners. This is done by grouping the learners available to us by their mechanism of learning like Support Vector, Trees or Discriminant Analysis. This is in line with the way they are grouped in the SciKitLearn library (1. Supervised Learning, n.d.).

PFNs are trained on every group and evaluate on their own groups to create a baseline for the model without a domain shift. Then the effect of the domain shift is shown by comparing the baseline to the performance of models on other groups.

Doing this for the groups of learners shows us if there is a pattern in the effect of domain shift, thus answering one of our research questions. It will also reveal what groups are of interest for further investigation when looking at individual learners. This grouped experiment is done first as an exploratory experiment to see which learners should be the focus of experiments that look at research question 2, as training models for all 24 learners individually is too time-consuming.

3.2 Experiment 2: Investigation of the Discriminant Analysis Learners

Experiment 2 looks into single learners that were part of a group that showed anomalous behaviour in experiment 1, to see if we can answer research question 2. It will look into the Discriminant Analysis group, which turned out to be the worst performing group in general in the previous experiment. It seemed to be a messy group, causing the domain shift to have a very big effect from all other groups. When looking at figure 1 the percentage of peaking and dipping is high in this group, meaning it warrants further investigation. We can see which of these learners causes the biggest domain shift.

The learners of the DA group, the Linear Discriminant Analysis and Quadratic Discriminant Analysis learners will be compared to the trees group, which was the best performing group in the previous experiment. This was the biggest domain shift present in experiment 1, meaning it is warrants investigation.

3.3 Experiment 3: Well-Behaved Curves Compared to Ill-Behaved Curves

The previous experiments indicated the fact that the well-behavedness of curves of a learner could be a factor in the effect of domain shift. Experiment 3 will explore this further by comparing some well-behaved learners, some ill-behaved learners, and a middle of the line learner. Finding out if this pattern actually exists will help answer research question 1 and 2 as it looks for a trend while also looking at domain shift along individual learners.

4 Experimental Setup

This section will expand on the method by going more into the details of how the experiments are conducted. It will also briefly discuss the procedure I use for significance testing of my results.

4.1 Experiment 1: Comparing groups of Learners

The groups made for the first experiment are:

- Support Vector Classifiers: 'SVC_linear', 'SVC_poly', 'SVC_rbf' and 'SVC_sigmoid'
- Tree based classifiers: 'DecisionTree', 'ExtraTree', 'ens.ExtraTrees', 'ens.RandomForest' and 'ens.GradientBoosting'

- Naïve Bayes Classifiers: 'BernoulliNB', 'MultinomialNB', 'ComplementNB' and 'GaussianNB'
- Nearest Neighbours Classifiers: 'KNN' and 'NearestCentroid'
- Discriminant Analysis Classifiers: 'LDA' and 'QDA'
- Linear models: 'LogisticRegression', 'PassiveAggressive', 'Perceptron', 'RidgeClassifier' and 'SGDClassifier'
- Neural networks: 'MLPClassifier'
- Dummy: 'DummyClassifier'

The learners are grouped based on the method of learning, following the SciKitLearn documentation (1. Supervised Learning, n.d.).

To evaluate the accuracy of models, I will use the mean absolute error. It is a commonly used metric that does not use any scaling, where mean squared error would artificially make smaller errors less prevalent since the error will always be below 1 in this experiment.

Another metric we use is the confidence interval, which is a range of values likely to contain the true value. (Hazra, 2017) If the 90% confidence interval of the network's predictions is very big, it is a sign of the network being very unsure about the prediction it made. This can be because the data it trained on diverges a lot, meaning the curves are very messy.

Three models will be trained on every group, using differing splits of data. This is to make sure that the results are not based on a model that is good or bad because of an unlikely fluke. This makes the results more reliable. Doing this for 24 learners would be a very strenuous process. This approach allows more focus on areas of interest after I see where interesting behaviour happens.

The dummy classifier is a classifier that make predictions that ignore the input. Therefore it never learns and will be a good baseline for the domain shift. An effect on performance should be prevalent as the learning curve is not a true learning curve since it never learns anything.

4.2 Experiment 2: Further Investigation of the Discriminant Analysis Learners

Experiment 1 revealed that the networks that were not performing too well suffered from a lower amount of curves being present. Therefore we got an improved version of the training procedure for the models, which uses data augmentation. This makes sure that we do not run into the problem of too little data, as this was shown to be a problem in the previous experiment. This augmentation algorithm modifies every curve fed into the training so it keeps getting 'new' curves as input. This improves performance of the models considerably when there is not a lot of data. The augmentation was done with the procedure outlined in Appendix A3 of Viering et al. (2024)

I have also randomly chosen data from the training sets to make sure all PFNs are trained on an equal amount of data, as this was a critique on my previous experiment. I was unable to apply this to my first experiment due to the time constraints of the project. The rest of the procedure is the same as the first experiment.

4.2.1 Verification of similarity of procedures

Since the training procedure between experiment 1 and 2 was adjusted, it is necessary that we verify that similar trends occur when using both training procedures.

To verify if the different training procedures have similar results a supplementary experiment was run to see how the different models would perform on the domain shift tested in experiment 2. So for PFN 1 the training used in experiment 2 is used, and for the second PFN augmentation is not used. I do cut down the training set sizes to ensure equal amounts of data.

I concluded that while augmentation significantly improves the performance of the models and the magnitude of effect the domain shift had, it does not alter the general pattern of the results for this experiment. Therefore I believe experiment 1 still gives us valid results for finding the pattern between groups.

The full supplementary experiment can be found in Appendix A

4.3 Experiment 3: Well-Behaved Curves Compared to Ill-Behaved Curves

The third experiment looks into both research question 1 and 2. Experiment 1 gave a hint that the pattern that might be most influential for the performance after doing domain shift is if the curves are well or ill behaved, and Experiment 2 confirmed that QDA, which is the more ill-behaved of the 2 learners in the DA group was still the worst evaluated. I plan to test this idea more by testing the domain transfer among single learners, which will be a mix of well-behaved and ill-behaved curves.

This experiment tests the following learners, basing on figure 1:

- SVM_Sigmoid: This learner has very high percentage of dipping, a high percentage of peaking, and has a low percentage of monotone and convex curves.
- LDA: This curve has a lot of peaking, but lower dipping. The peaking is about the same percentage of peaking as SVM_Sigmoid, so the performance during the domain shift between these can give an indication about the effect of dipping. While compared against well-behaved curves gived an idea of the effect of peaking.
- NearestCentroid: This curve has average percentages in every category, so it is a nice curve to evaluate as we expect different behaviour towards the well-behaved curve and ill-behaved curves.
- GradientBoosting: This seems to be one of the most well-behaved learners, with the most Monoticity, Convexness, and combination of those 2. It also has little peaking and dipping.
- ExtraTree: Another very well-behaved curve like the previous one.

These curves should give a coherent picture of what the contribution is of the well-behaved ness of curves on the effect domain transfer

4.4 LCPFN Training and Evaluation Parameters

The performance of the PFNs is very subject to the parameters given during training and evaluation. The PFNs were trained using learning curves of length 80, with 300 epochs, 3 transformer layers, an embedding size of 128, a batch size of 20 and a learning rate of 0.0001. The evaluation is only done on the last 20 anchors, meaning that only the PFN will only try to predict the last 20 points of the learning curve.

For experiment 1 all curves available by splitting the data 80-20 for training and testing was used for training. In later experiments, this was cut down to 5300 curves during training to ensure equal amounts of data, as that was the lowest amount of curves any learner or group had individually.

More elaborate information on the exact setup can be found in the repository in the training notebooks, the link to which is in the responsible research section.

4.5 Significance testing

When looking at the results of an experiment, a significance test is used to find if the distributions of the evaluation of a model's own group is statistically significantly different to the evaluation after shifting the domain.

To determine what type of test should be used to find the significance it is needed to determine the distribution of data per domain shift. To determine this, the data is standardized by getting the z-score of all data points. This means that if the data was normally distributed, it would adhere to a standard normal distribution after getting the z-score. Then a Kolmogorov-Smirnov test is used to see if they adhere to a normal distribution.

For all distributions in the experiments, the p value was below 0.05. The conclusion from that is that the data is not normally distributed. This was also run on the data from the individual models, instead of

the aggregate, to make sure that that data is also not following a normal distribution. Knowing that, Mann-Whitney is the best statistical test for looking at the significance of my results compared to an alternative like a t-test. The code showing this is in the repository.

5 Results

This section will compile the results of all experiments done throughout this project.

5.1 Experiment 1: Comparing subgroups of Learners

In experiment 1, three models were trained on every subgroup with 3 different separations of test and train data. The shown statistics are averages of all 3 models, unless otherwise mentioned. The statistics were calculated by aggregating all results of the 3 models and running the evaluation on those combined data points.

We can see that the PFN's are not very good overall. This seems especially prevalent when looking at the model trained on the network. However, all groups that had only 2 or less classifiers perform worse in the evaluation. Following that result we have been given a new way to train models for experiment 2.

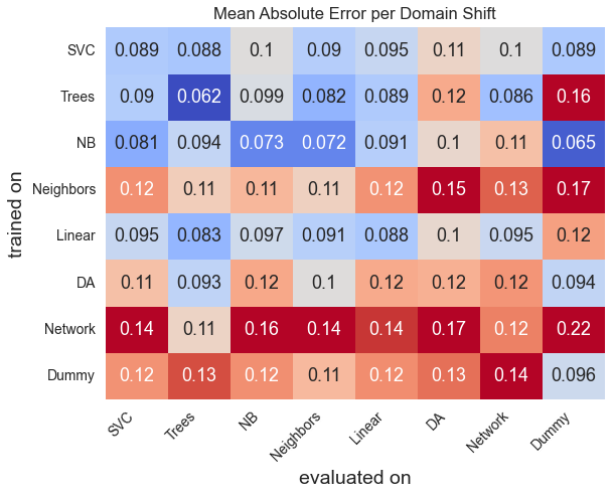

Figure 2: MAE per domain shift. Every cell represents the MAE when using a model trained using data from the group on the y axis, and evaluation on the group on the x axis.

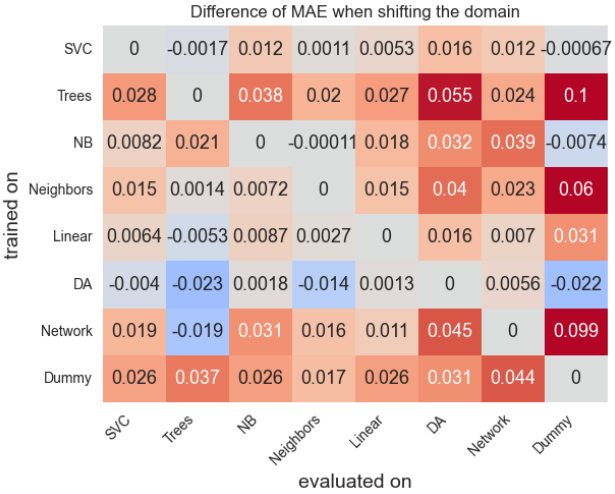

Figure 3: Difference of MAE when shifting domain computed by subtracting the error from the baseline of a model from the performance on group on x axis

Looking at figure 3 It shows that the Discriminant analysis model does better on other groups than on it's own on average, if you look at the corresponding row. One can see that the DA trained model actually performs better on half of other subgroups, and very marginally worse for the others, where we would typically expect a model to be best in its own group. Figure 4 reveals from the three times where it does not do better on other groups, the result is not actually significant twice. Therefore there is only 1 domain shift that made the DA network perform significantly worse.

The DA group also has a significant effect on the evaluation of other models when those models evaluate on that group. The biggest effect of the domain shifts is when using the PFN trained on Trees to evaluate on the Discriminant Analysis group. The DA group also has the largest confidence interval as shown in figure 5.

In the same table, it is shown that tree-based classifiers have the smallest confidence intervals, ignoring the Dummy. From all models evaluating their own group, Trees are have the smallest average MAE. On average the trees group is also evaluated better by other models then the group they trained on.

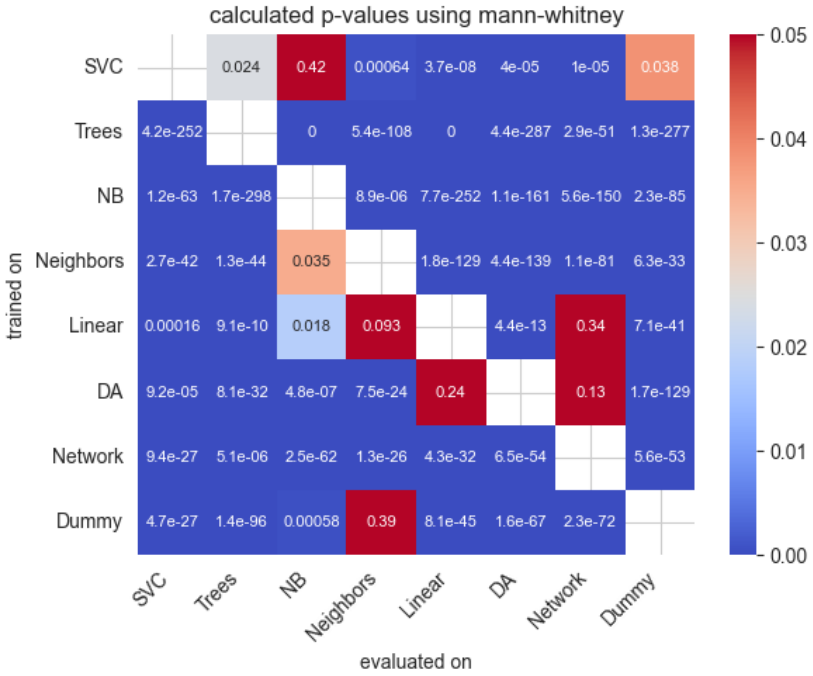

Figure 4: Significance test p-values for experiment 1. If a cell is dark red it means that p-value exceeds 0.05, meaning that domain shift does not have a significant impact on performance of the model.

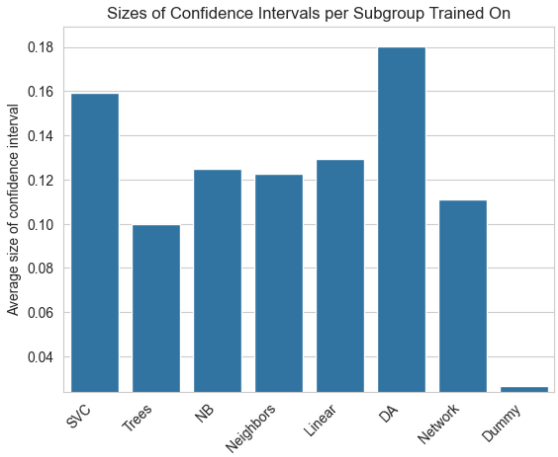

Figure 5: Average size of confidence intervals over evaluation of own group and all other learners.

An interesting takeaway from the p-values in figure 4 is that the shift to the neighbours group from the dummy classifier is not significant.

5.2 Experiment 2: Investigation of the Discriminant Analysis Learners

For experiment 2, we can see in figure 6 that there is less Mean Absolute Error for the Trees group, which is still the same as the previous experiment. The mean absolute error going down to 0.038 from 0.062 is a substantial gain, verified to be significant with a Mann-Whitney test. The second observation from this

figure is that QDA is evaluated badly by every model, while the other 2 groups are evaluated relatively well.

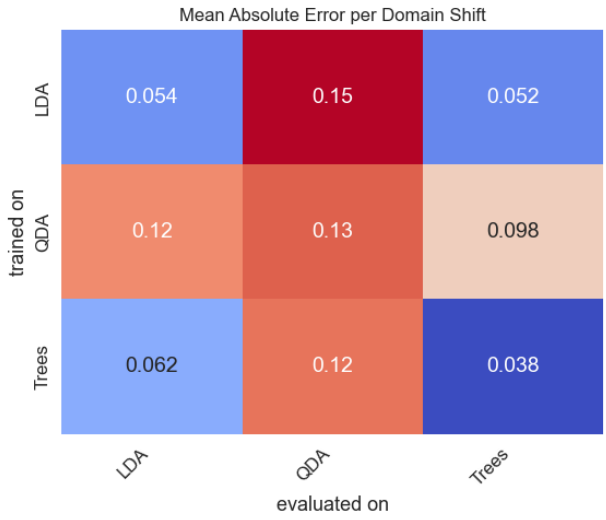

Figure 6: MAE per domain shift. Every cell represents the MAE when using a model trained using data from the group on the y axis, and evaluation on the group on the x axis.

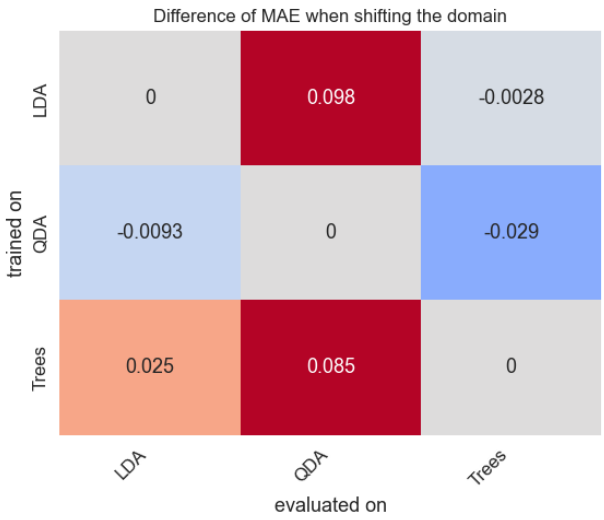

Figure 7: Difference of MAE when shifting domain computed by subtracting the error from the baseline of a model from the performance on group on x axis.

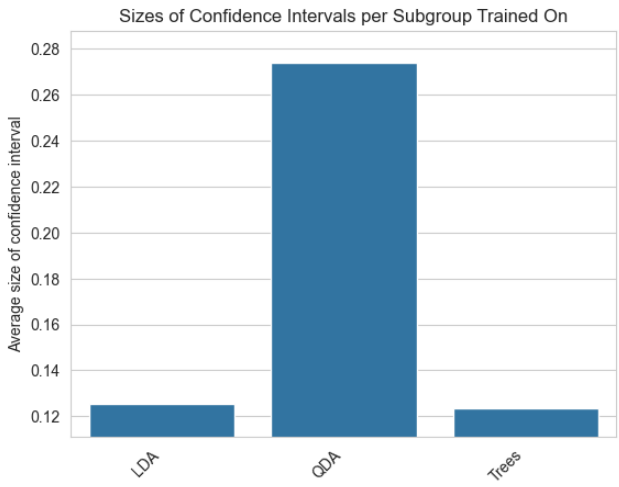

Figure 8: Average size of confidence intervals over evaluation of own group and all other learners.

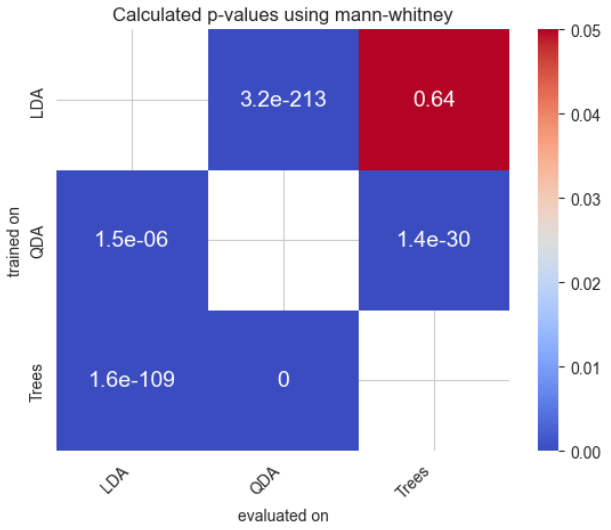

Figure 9: Significance test p-values for experiment 2. If a cell is dark red it means that p-value exceeds 0.05, meaning that domain shift does not have a significant impact on performance of the model.

Figure 7 shows that the performance of the QDA models improves when evaluating other models. The LDA model seems to do slightly better when evaluating trees, but way worse when evaluating QDA. The confidence interval of QDA is also the biggest according to figure 8.

In figure 9 we can see that almost all shifts have a significant effect on the Mean Absolute error. The one

that is not significant is LDA to trees. While it did slightly better compared to evaluating its own learner, the difference in average MAE was only 0.0028.

5.3 Experiment 3: Well-Behaved Curves Compared to Ill-Behaved Curves

Figure 10: MAE per domain shift. Every cell represents the MAE when using a model trained using data from the group on the y axis, and evaluation on the group on the x axis.

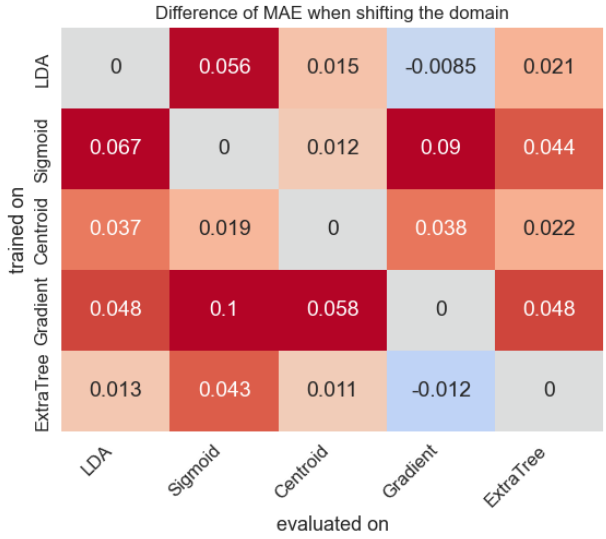

Figure 11: Difference of MAE when shifting domain computed by subtracting the error from the baseline of a model from the performance on group on x axis.

In the third experiment, the 5 selected learners had some interesting results. When looking at figure 10 we can see that the sigmoid model has the highest mean absolute error on it's own group compared to other models evaluating their own groups. The model trained on the sigmoid learners also has the highest MAE on average (taken along the rows).

When looking at figure 11 we can see that the model trained on sigmoid and the model trained on gradient have the most loss when evaluating other learners, all differences in error above 0.05 except for 1 belonging to those to models. The other big error was from the LDA model evaluating the sigmoid group.

Figure 12 gives us a different view about the consistency of the learners. LDA seems to be cleaner than ExtraTree and Centroid, while the percentages in figure 1 states otherwise.

Two of the performed domain shifts showed the behaviour where the model does better than on the group it was trained on. This occured with the LDA to Gradient shift and the ExtraTree to Gradient shift.

If we look at figure 13 we can see that almost all domain shifts have a significant effect. The two that are not seen as significant are domain shifts to the centroid learners.

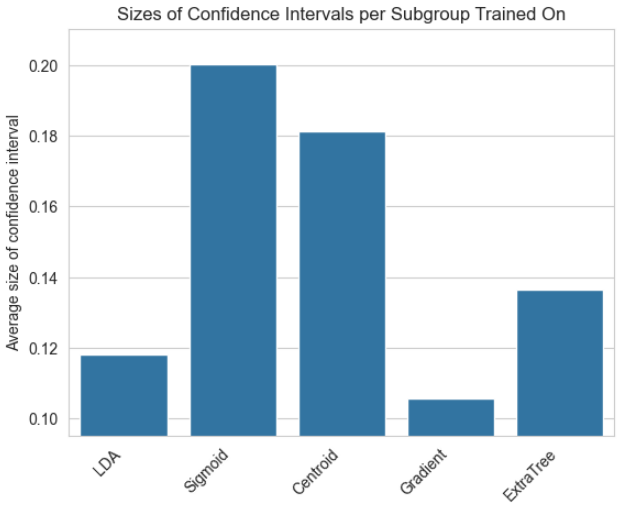

Figure 12: Average size of confidence intervals over evaluation of own group and all other learners.

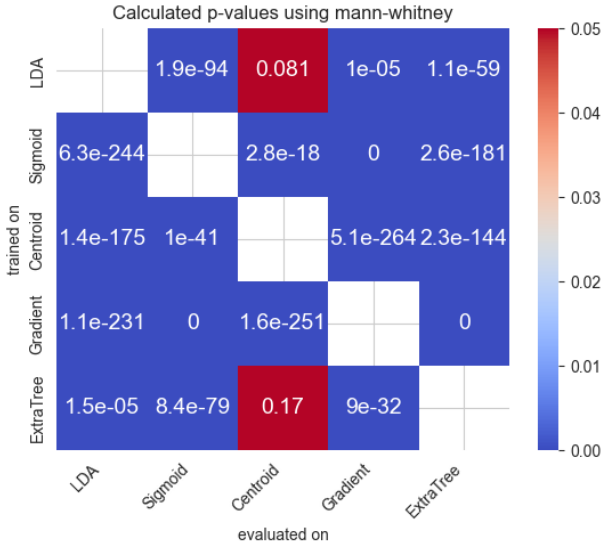

Figure 13: Significance test p-values for experiment 3. If a cell is dark red it means that p-value exceeds 0.05, meaning that domain shift does not have a significant impact on performance of the model.

6 Discussion

In experiment 1, the results showed the fact that the DA subgroup has very messy curves. This is in agreement with previous work shown in figure 1, where they are shown to have a lot of peaking and dipping. Trees seem to be the most well-behaved group when looking at the statistics, they also have the lowest MAE and the smallest confidence intervals. Combining that with the fact that other models do better on trees than on their own groups on average and the DA model performing better after a domain shift, indicates that the trend of domain shift between subgroups could be partly caused by how well-behaved curves in the subgroup are.

The fact that the dummy to neighbors domain shift does not have a significant effect is a very surprising result, as this would imply a PFN trained on dummy learning curves (which do not actually learn and thus are flat or random learning curves) does not perform significantly worse when evaluating the Neighbors group learning curves. This could be because the setup only evaluates on the last 20 anchors of each learning curve. At the end of the learning curve, the curve is generally flatter, so it could be that the neighbors group has curves that are very flat at the end, which could make the Dummy classifier model perform well on that group. Figure 1 does show that both learners of the neighbors group have more flat curves than normal, so it is definitely possible that these curves tend to be flatter at the end. This does show that some results seem to be heavily dependant on the setup of the experiment.

Experiment 2 heavily indicates that QDA is the group that caused most of the messiness of the DA group in experiment 1. This aligns with our hypothesis based on figure 1.

From figures 6, 7, and 8 we can conclude that QDA is difficult to predict, as it exhibits the same behaviours as the Discriminant Analysis group in experiment 1. It has the biggest confidence interval, is evaluated way worse by the other groups and the model performs better on average when evaluating the other groups. While LDA has some peaking according to figure 1, QDA has the messier set of curves with both peaking and dipping having a high percentage.

Experiment 2 also showed some interesting behaviour with significance where the shift from LDA to trees had a difference of 0.00028. It is logical that this is not seen as significant. However, the QDA to LDA difference was also small and is considered very significant. With the amount of samples, over a thousand

per distribution, it is possible that a change like that makes a big difference in significance.

In experiment 3, Sigmoid was the worst evaluated. This is consistent with what we have seen so far in experiments 1 and 2 when concerning messy curves. However, ExtraTree has the second lowest accuracy, while it is one of the well-behaved curves. This is a surprising fact that we would have expected to be different to fully align with the previous indications of the pattern.

The fact that the domain shift from LDA to sigmoid has a big effect is interesting as the difference in distribution between the groups is mostly in the dipping percentage and flatness, with the percentage of peaking being similar. This means that the dipping and flatness is certainly a major factor of the effect of domain transfer as the model that has the least loss evaluating the sigmoid group is the centroid model. This is interesting because the centroid learner also has a moderate percentage of dipping and flatness, which the other groups do not have.

The positive effect of domain shift on the performance in the LDA to Gradient and the ExtraTree to Gradient shifts is also of interest. Gradient is the most well-behaved group of the ones in this experiment according to figure 1. This does show that when this positive effect happens in the experiments, it is often towards a more well-behaved group or learner. This is very interesting and definitely warrants further investigation.

The discrepancy between the stats of LDA and ExtraTree in figure 1 and the sizes of their confidence intervals in figure 12 is noteworthy. ExtraTree having a large confidence interval while the statistics say it should be very well-behaved shows the statistics about ill behaviour in figure 1 may not give the complete picture as to what causes a group or learner to be messy and hard to predict for the PFNs.

The fact that the domain shift towards the centroid learner is not significant twice makes sense as it was included as the middle of the line learner with percentages of peaking and dipping that are not too good and not too bad, making it closer to both well and ill behaved groups.

The results of experiment 3 further support the fact that well-behavedness and behaviour like peaking, dipping, and flatness are definitely a factor in the effect of domain shift, but also indicate that there are other factors that also contribute.

The significance analysis on experiment 1 revealed that 86% of effects of domain shifts between the groups were statistically significant. Between single learners in experiments 2 and 3 88% of the domain shifts had a significant effect. This does clearly show that the effect of domain shift in general is relevant and significant.

Regrettably, the experiments performed in this research were not perfect. For the first experiment I was unable to rerun it with the improved training procedure as I needed to train a lot of models for that and the time limits for this project did not allow for that while also doing other experiments. I did verify that similar patterns would occur with both training methods, but re-running that experiment could be of interest.

Another experiment that was unfortunately unattainable due to time constraints was a comparison of all individual learners on their own. This would answer research question 2 in much greater depth, and revealing the effect of domain shift on a broader scale.

One thing in the training of the PFN's that could be improved is that we have only trained on curves of length 80. This is something that the whole group used, and was given to us at the beginning of the project. Being able to use more curves that were too long now would mean better training in general. This is something that could have been improved in the experiments.

Another point of discussion is the fact that I predicted the last 20 anchors. Only focussing on this last part of the curve can possibly have an effect on the domain shift as well, some differences between curves of different learners may happen in other parts of the curve. Comparing the domain shift using different anchors can also be useful.

Lastly, there are setups available for the LCPFN's that perform even better than what was used here. However, it was unattainable for this research to use those settings due to the amount of models that had to be trained. This is because those settings would take around 75 times longer to train based on my discussion with teammates. This is simply not feasible with the time constraints of this project.

7 Conclusions and Future Work

Conclusion

This thesis has analysed the effect that domain shift has on learning curve extrapolation. The research question that was investigated in this paper is: What is the effect of domain transfer on learning curve extrapolation? This was split into the questions: "Is there a trend between groups of learners in the effect of the domain transfer?" and "Does domain transfer over single learners impact the accuracy of PFNs?".

With the experiments that were done, the research questions can be answered in the following way:

- **Research Question 1**: There is a trend in the effect of domain shift between groups of learners present. Experiment 1 concluded that it seems to be at least partly dependant of how well-behaved the group or learners are, because some domain shifts resulted in models improving performance. Experiment 2 and 3 also indicated a similar pattern. However, we have definitely not uncovered the full pattern behind the effect of domain shift, as experiment 3 did have some results that did not fully adhere to this pattern, and results like the confidence intervals suggest that the well-behavedness statistics were not a full predictor for how messy a group is.
- **Research Question 2**: The domain transfer definitely impacts the accuracy of the LCPFNs between single learners. With 88% of the effects of domain transfers being significant with single learners, we can see that most domain shifts have a significant effect, but not all.

Therefore we can answer the main question of this thesis with certainty that domain shift has a significant effect on performance of learning curve extrapolation when applying LCPFNs. Overall, the different domain shifts had a statistically significant effect in around 87% of the cases investigated.

Possibly the most interesting result of this research was that sometimes the domain shift had a positive effect, and the model did better on data it was not trained on. This occurrence happened multiple times and definitely warrants more investigation to see if the well-behavedness pattern that seemed to occur is the explanation behind that or if there are other factors at play as well.

Future Work

This research gives a small insight into what could be of importance to the effect of domain transfer on learning curve extrapolation. However there are also lots of aspects still to discover.

Fistly, as hinted at in the discussion the experiment using all individual learners available should be a complete picture of what the effect of domain shift truely is. In the experiments in this paper we have convincingly proven that domain shift is definitely relevant and has a significant effect on the extrapolation of learning curves, but the complete picture of all learners against each other can definitely be valuable to get a broader and more precise idea of when it is significant.

That will also help in determining the patterns, about which the current conclusion of our experiments is that well-behavedness is definitely a factor, however it should be investigated further. The experiments suggest that there are other factors at play, and further experimentation on different groupings of learners can figure that out, since peaking and dipping percentages do not seem to be the complete pattern. This is an interesting aspect of the research to pursue further as finding other factors will make the full pattern even more clear, making it a promising direction for future work.

The discussion also talked about the infeasibility of using the best performing settings for this experiment due to time constraints. An interesting aspect of future work is the seeing if the better settings impact the domain shift, and the patterns we see in the results. The supplementary experiment showed that the patterns were similar, but it is interesting to see how that evolves when the performance improves even further.

8 Responsible Research

This research was conducted with the TU Delft code of conduct in mind. (Roeser and Copeland, n.d.) It outlines the core values, known as DIRECT, which stands for Diversity, Integrity, Respect, Engagement, Courage, and Trust. For research, the Integrity is very important of these values. Part of the Integrity value is that research is carried out responsibly.

To ensure that this research is carried out responsibly, seeding is used for both randomness control and data splitting in the code. This was verified using code from Check If Models Have Same Weights (2017). My models are trained with the random seed 42 initialized at the tops of the notebook. To make sure the models are trained on different data every time the data split states are rotated for every model. For every group or learner trained on, split states 1, 8 and 42 were used to divide the data into the training and test splits. Then the results are averaged for the results, to get a more representative result and accounting for variance in performance of models. All models I have trained are available in the GitHub repository (https://github.com/Maxmwoan/Research-Project), so the extraction of data can be done without first having to train the model to verify findings. Models can also be retrained by using the correct seeds in the repository.

An ethical issue which this research affects is the understanding of learning curved. When learning curves are better understood, this can speed up training of models. Speeding up model training can accelerate the use of AI, reduce costs, and lower the environmental impact of training models. While my research will not make a big impact on its own, it can be a step in the right direction to improving these ethical problems with AI.

8.1 Use of Large Language Models

The integrity value of the TU Delft code of conduct also references transparency. Therefore this section will disclose how Large Language Models were used during this project.

Task	**Yes**	**No**
Help with formatting LaTeX Documents	✓	
Assistance while gathering research papers	✓	
Assistance while generating figures for report	✓	
Designing experiment structure		✓
Formulating research questions		✓
Running experiments		✓
Writing/helping improve the writing in the report		✓

Table 1: Use of LLMs for this project

For each item I have answered yes to in table 1 I will elaborate on the context of AI usages.

Help with formatting LaTeX Documents: While formatting the research paper I have used LLMs to debug errors I got, implement features like referenceing correctly, and help create the format of tables/figures.

An example of a prompt is: "<Example figure 1 code> <Example figure 2 code>. Can you make these figures appear next to each other on the page using LaTeX?"

Assistance while gathering research papers: At some points finding the correct type of papers or in one case I was unable to find a reference from another paper. Therefore I used an LLM to help search for papers. Any papers that were returned were carefully reviewed to determine if they fit the research I am doing.

Example prompts: "Can you find some scientific papers about the concept of domain shift or domain transfer?" and "<Reference from a paper I was unable to find>. Can you find a link to this reference?"

Assistance while generating figures for report: When designing the figures for this report, I used LLM's to help me figure out how to make certain improvements to my figures.

Example prompts: "If I want to add <feature> to a seaborn heatmap, how do I do that?", "Is it possible to add <feature> to <type of figure>?", and "How do I fix <Issue with figure> in a <type of figure> in python?"

References

1. Supervised learning. (n.d.). Scikit-Learn. Retrieved 8 May 2025, from https://scikit-learn/stable/supervised_learning.html

Adriaensen, S., Rakotoarison, H., Müller, S., & Hutter, F. (2023). Efficient Bayesian Learning Curve Extrapolation using Prior-Data Fitted Networks (No. arXiv:2310.20447). arXiv. https://doi.org/10.48550/arXiv.2310.20447

Check if models have same weights. (2017, June 27). PyTorch Forums. https://discuss.pytorch.org/t/check-if-models-have-same-weights/4351

Hazra, A. (2017). Using the confidence interval confidently. Journal of Thoracic Disease, 9(10). https://doi.org/10.21037/jtd.2017.09.14

Mohr, F., & Rijn, J. N. van. (2024). Learning Curves for Decision Making in Supervised Machine Learning: A Survey. Machine Learning, 113(11–12), 8371–8425. https://doi.org/10.1007/s10994-024-06619-7

Mohr, F., Viering, T. J., Loog, M., & van Rijn, J. N. (2023). LCDB 1.0: An Extensive Learning Curves Database for Classification Tasks. In M.-R. Amini, S. Canu, A. Fischer, T. Guns, P. Kralj Novak, & G. Tsoumakas (Eds.), Machine Learning and Knowledge Discovery in Databases (pp. 3–19). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-26419-1_1

Roeser, S., & Copeland, S. M. (n.d.). TU Delft Code of Conduct: Why What Who How. Delft University of Technology. https://doi.org/10.4233/UUID:704E72B8-6B14-4CF1-A931-9C0F93C50152

Salem, H. B. (2024, October 23). Tackling Domain Shift in AI: A Deep Dive into Domain Adaptation. Medium. https://medium.com/@bensalemh300/tackling-domain-shift-in-ai-a-deep-dive-into-domain-adaptation-c2debd758edd

Sun, S., Shi, H., & Wu, Y. (2015). A survey of multi-source domain adaptation. Information Fusion, 24, 84–92. https://doi.org/10.1016/j.inffus.2014.12.003

Viering, T. J., Adriaensen, S., Rakotoarison, H., & Hutter, F. (2024). From Epoch to Sample Size: Developing New Data-driven Priors for Learning Curve Prior-Fitted Networks.

Viering, T., & Loog, M. (2022). The Shape of Learning Curves: A Review (No. arXiv:2103.10948). arXiv. https://doi.org/10.48550/arXiv.2103.10948

Wang, Y., Zhang, Z., Gong, D., & Xue, G. (2025). Mitigating domain shift problems in data-driven risk assessment models. Reliability Engineering & System Safety, 263, 111263. https://doi.org/10.1016/j.ress.2025.111263

Wu, S., Shu, L., Song, Z., & Xu, X. (2023). SFDA: Domain Adaptation With Source Subject Fusion Based on Multi-Source and Single-Target Fall Risk Assessment. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 31, 4907–4920. https://doi.org/10.1109/TNSRE.2023.3337861

Yan, C., Mohr, F., & Viering, T. (2025). LCDB 1.1: A Database Illustrating Learning Curves Are More Ill-Behaved Than Previously Thought (No. arXiv:2505.15657). arXiv. https://doi.org/10.48550/arXiv.2505.15657

A Supplementary experiment comparing the training methods

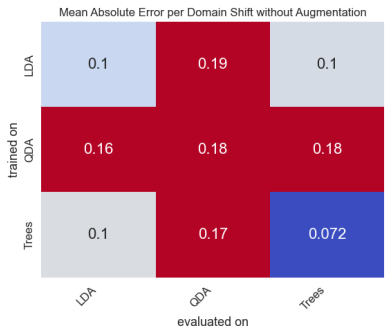

Figure 14: MAE per domain shift without Augmentation

Figure 15: Difference of MAE when shifting domain.

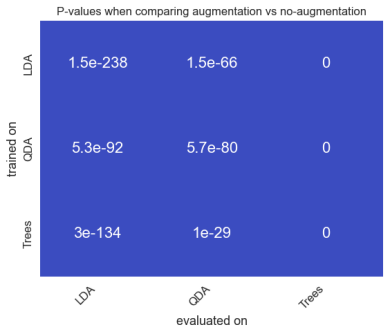

Figure 16: Comparison of evaluations of the different training techniques

When comparing the different training methods, we can see that there is a significant difference between the populations when comparing them using a Mann-Whitney test as in figure 16. The errors MAE is worse than when using augmentation, which is no surprise, but there does not seem to be a pattern in the difference in MAE when shifting the domain when comparing figures 7 and 15. There are some that differences that became bigger, with the differences became bigger when trained on trees.

One important thing to note is that the trend we discovered in experiment 2 still occurs where QDA model is still the worst and the group is also evaluated the worst by other models. This means both methods manage to reveal the grand pattern we see.

From the p-values for the domain shifts in figure 17 we can see that all results are significant, which especially surprising when looking at trained on QDA and evaluating on Trees. But is it only just about significant.

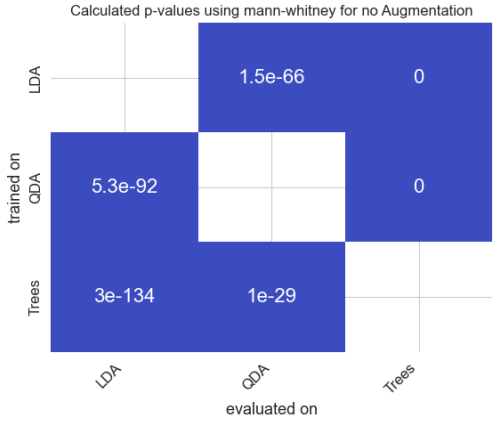

Figure 17: P-values for each domain shift