Deep Generative Design

A Deep Learning Framework for Optimized Shell Structures

P5

TU Delft

TU Delft

MSc Architecture, Urbanism & Building Sciences

Building Technology Track

Studio: Building Technology Sustainable Design Studio

Student

Stella Pavlidou

Student Number: 5385571

Mentors

Dr. Charalampos Andriotis, Structural Design & Mechanics

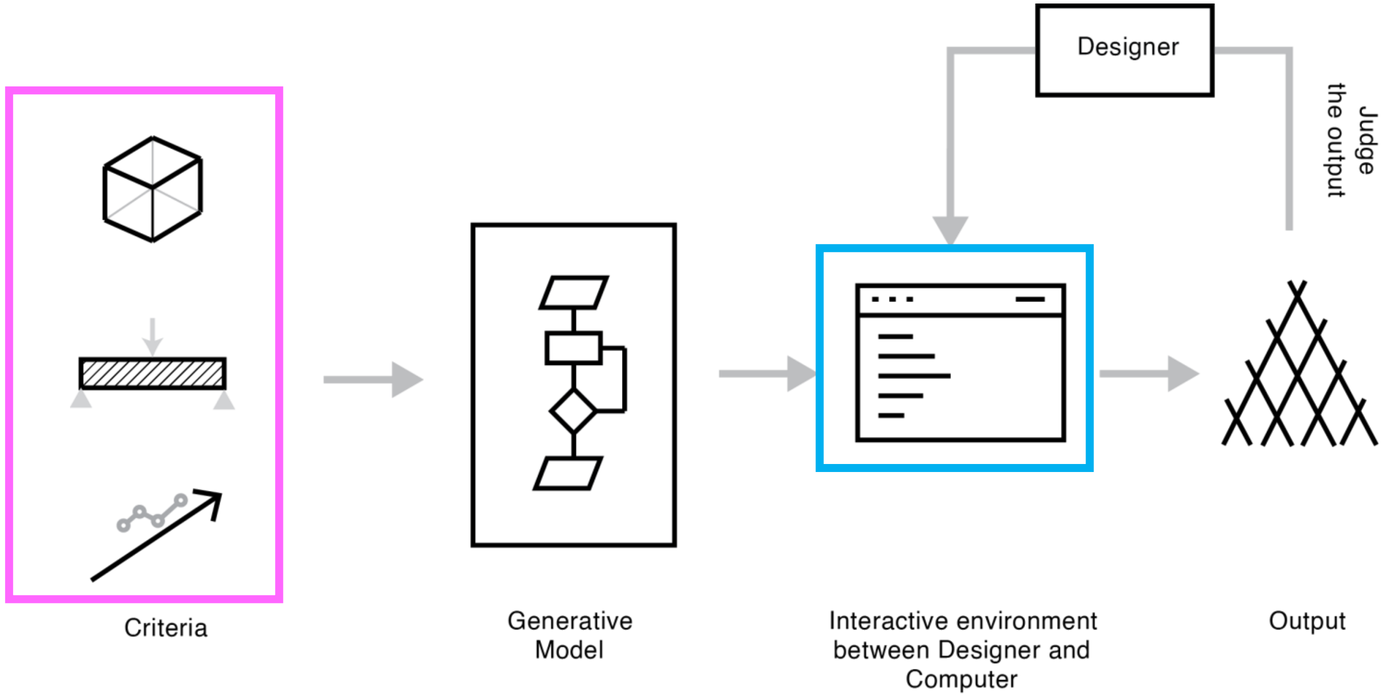
Dr. Michela Turrin, Design Informatics

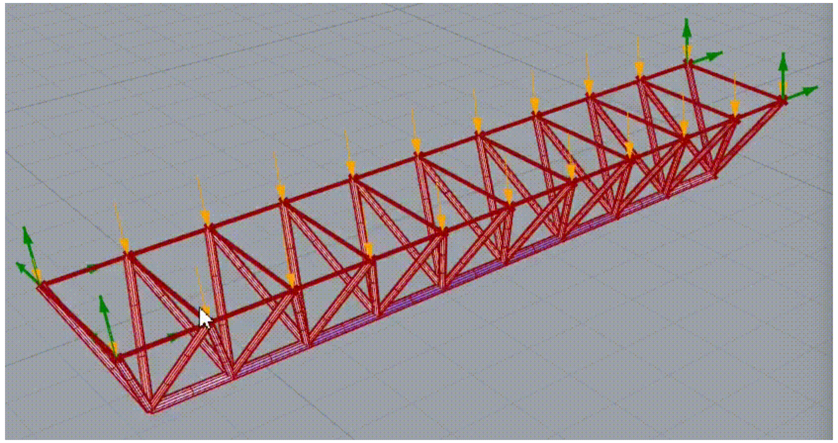
Delegate of the Board of Examiners

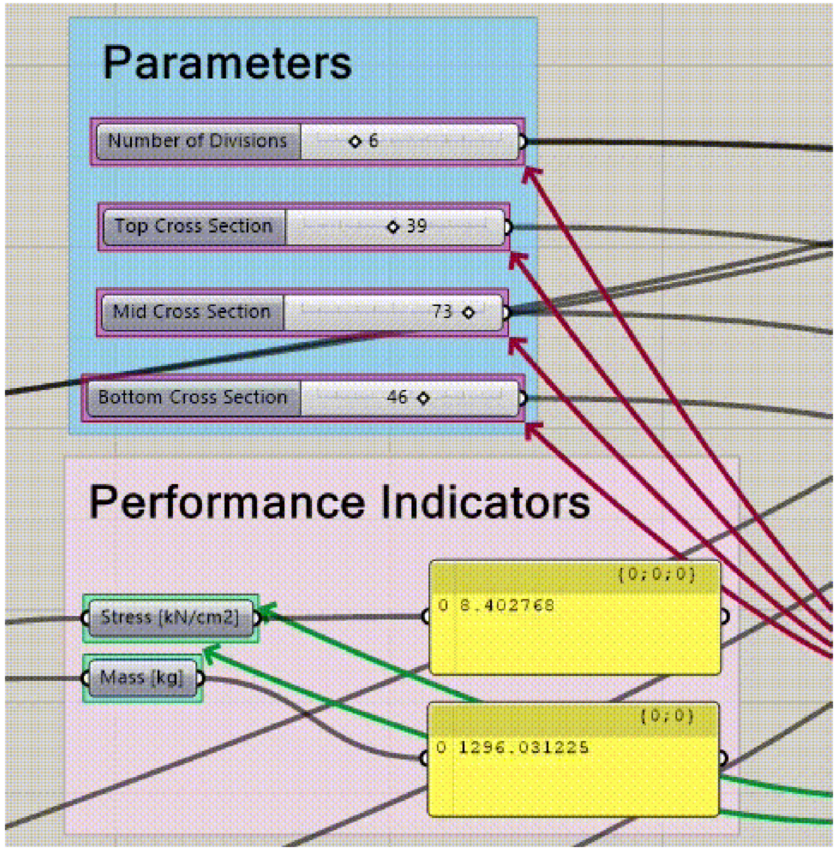
Herman de Wolff

Generative Design

Generative Design







Why Generative Design?

- Allows for a more **integrated** workflow between designer/engineer and computer.
- Facilitates the exploration of the **Design Space**.

Artificial Intelligence - Generative Design

Artificial Intelligence - Generative Design

Darth Vader cycling in Rotterdam
(Midjourney https://www.midjourney.com/app/)

Shell Structures

- Their topologies are explored by testing mesh tessellations.

- Topology affects:
  Aesthetics
  Structural Performance
  Cost
  Assembly Time

Figure 1.1. Robert and Arlene Kogod Courtyard

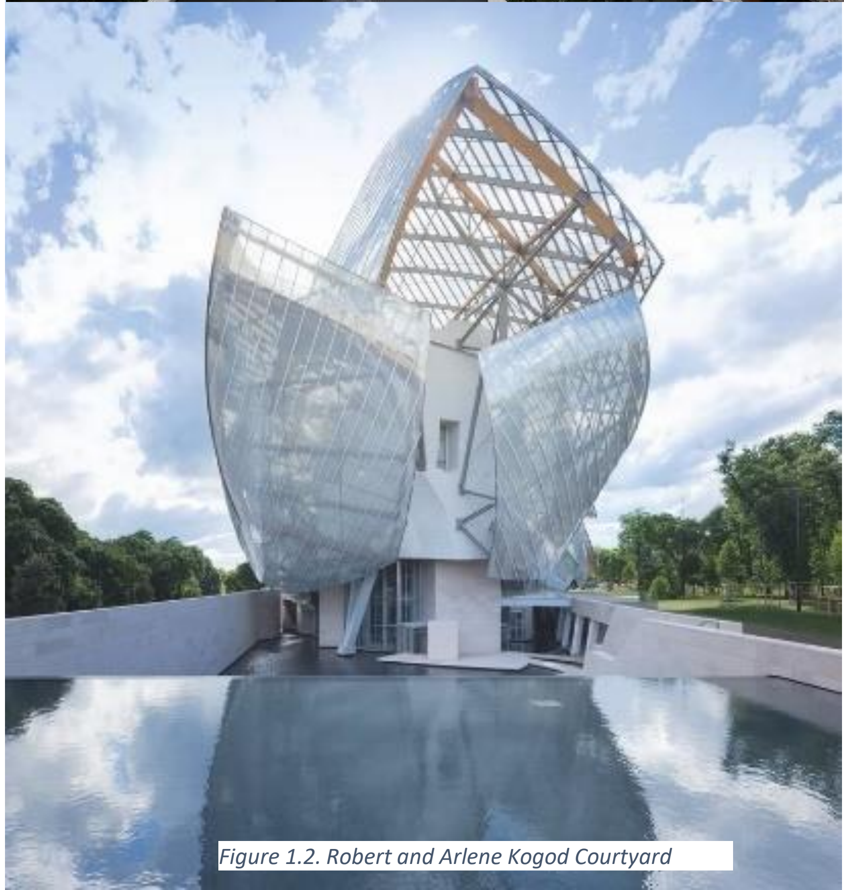
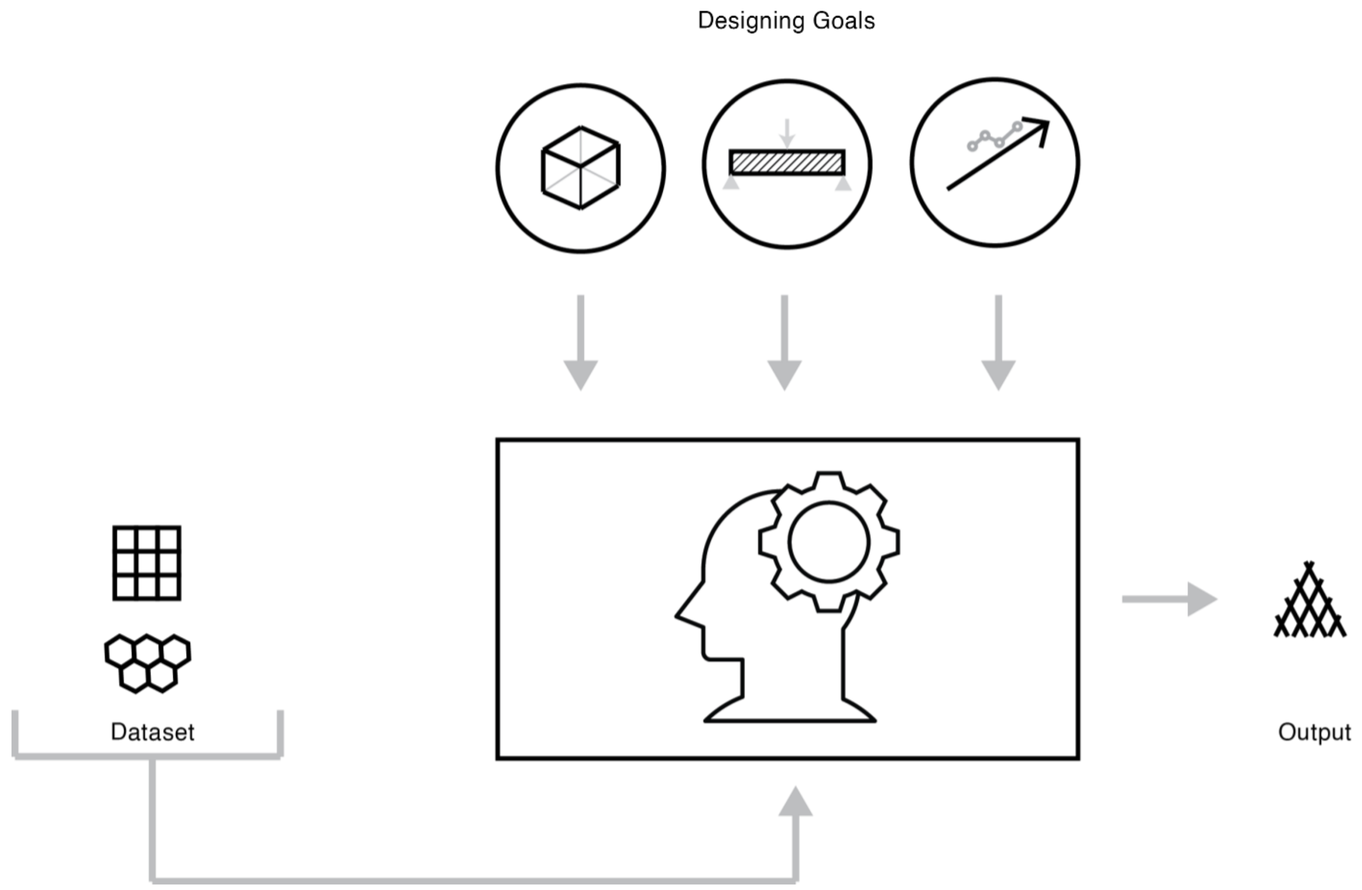
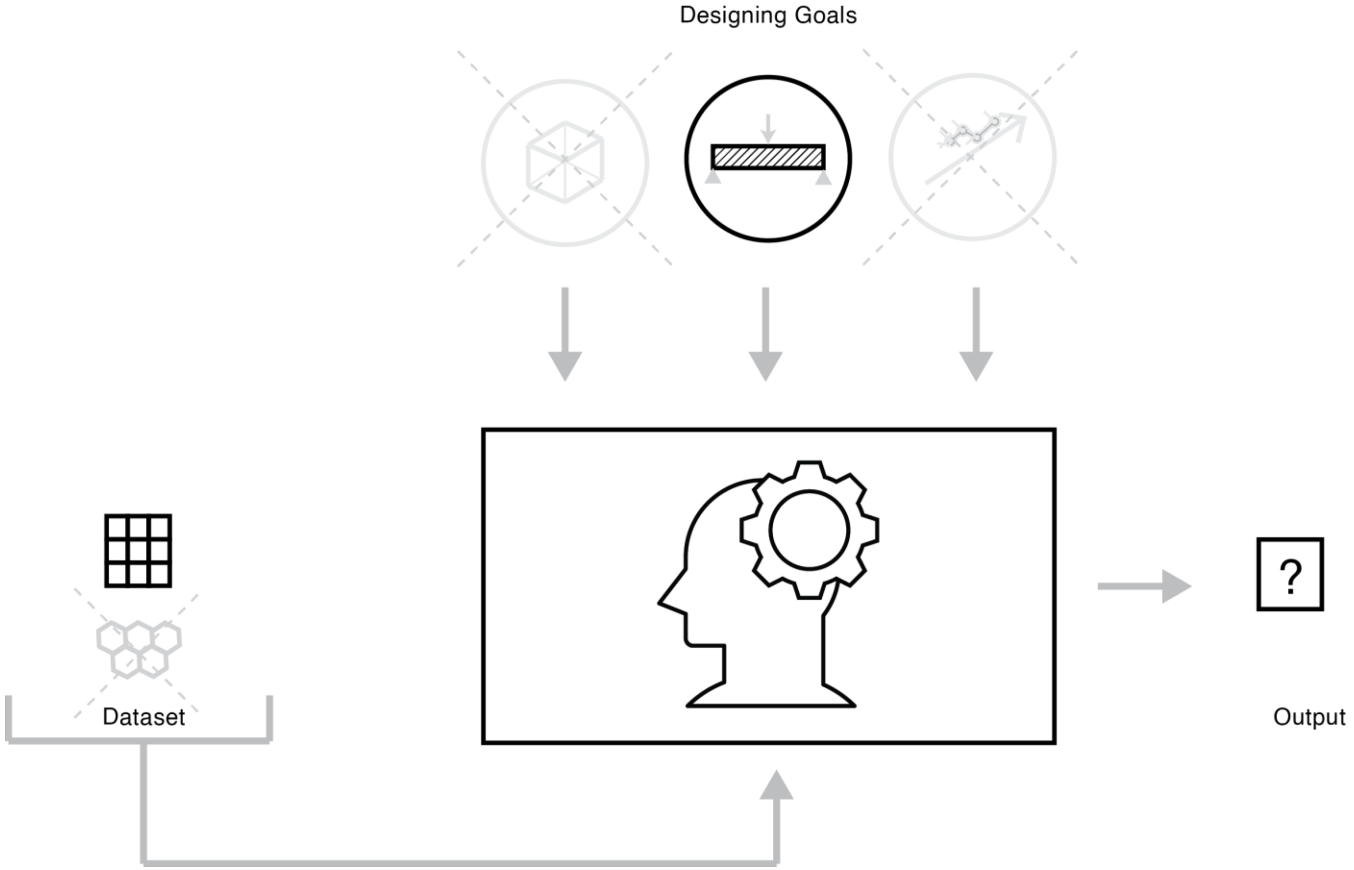


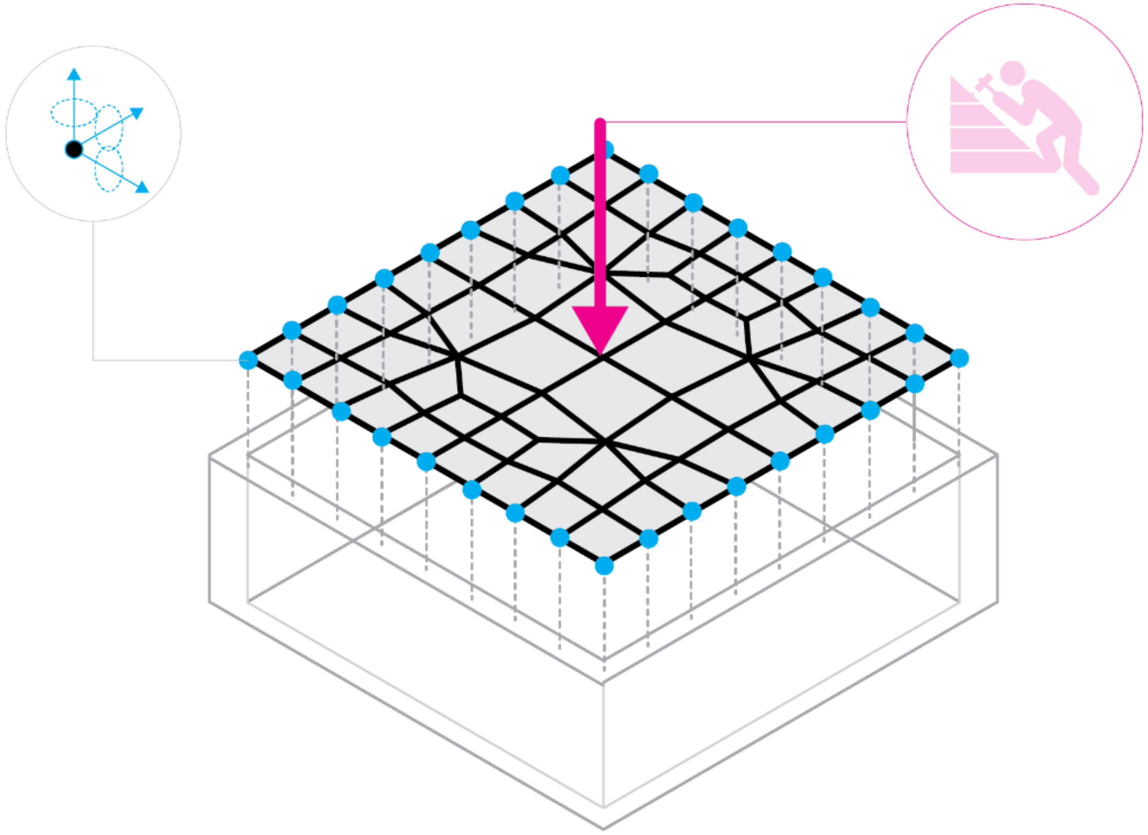
Figure 1.2. Robert and Arlene Kogod Courtyard

Artificial Intelligence - Generative Design

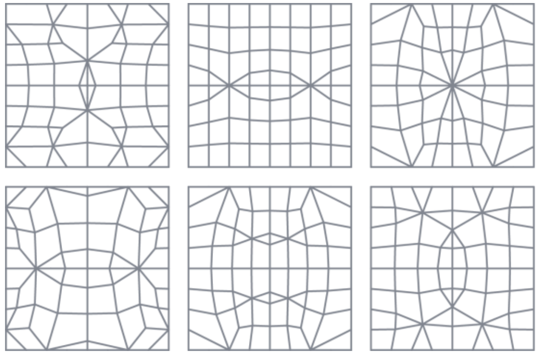


Artificial Intelligence - Generative Design





Creating the Dataset



Build an AI workflow

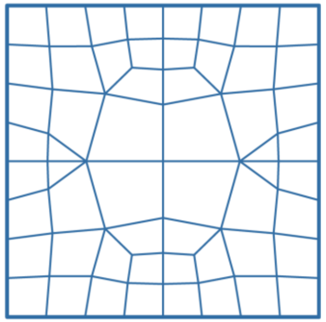
Conclusions

AI Workflow

- An AI Generative Model:
  Variational Autoencoder- VAE (Kingma & Welling, 2014)

- A model that predicts the structural performance:
  Surrogate Model that implements Regression with a deep neural network

- An Optimizer:
  A Gradient Descent Optimizer that searches the design space of the VAE for optimal solutions

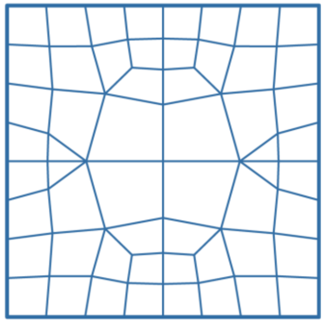


Generate a design!

AI Workflow

- An AI Generative Model:
  Variational Autoencoder- VAE (Kingma & Welling, 2014)

- A model that predicts the structural performance:
  Surrogate Model that implements Regression with a deep neural network



What is its structural performance?

- An Optimizer:
  A Gradient Descent Optimizer that searches the design space of the VAE for optimal solutions

AI Workflow

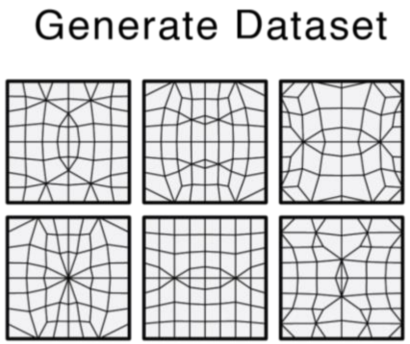
- An AI Generative Model:
  Variational Autoencoder- VAE (Kingma & Welling, 2014)

- A model that predicts the structural performance:
  Surrogate Model that implements Regression with a deep neural network

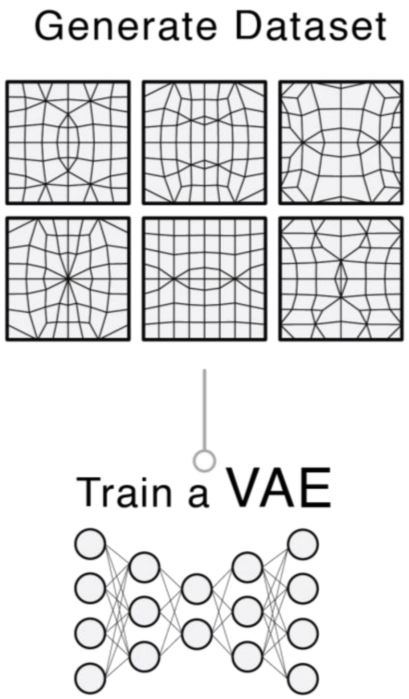
- An Optimizer:
  A Gradient Descent Optimizer that searches the design space of the VAE for optimal solutions

→ **Optimize the design**

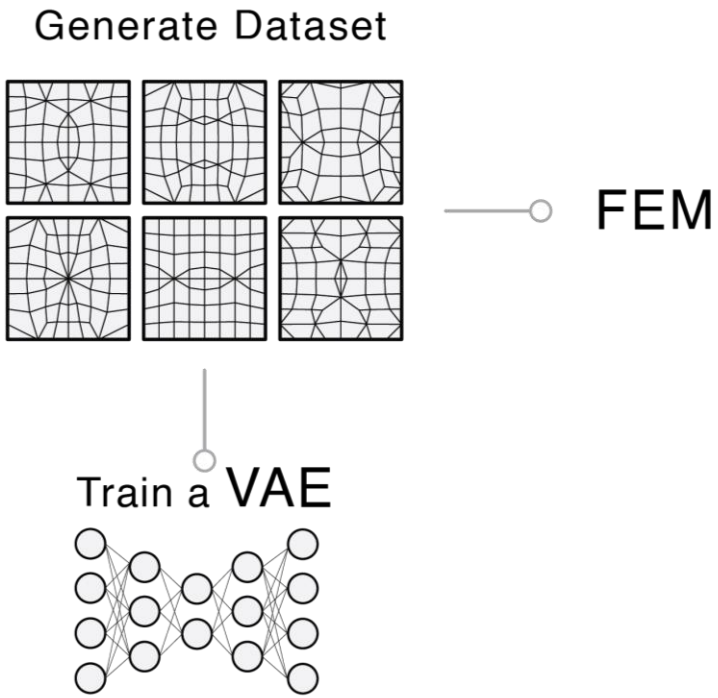
Generate the Dataset



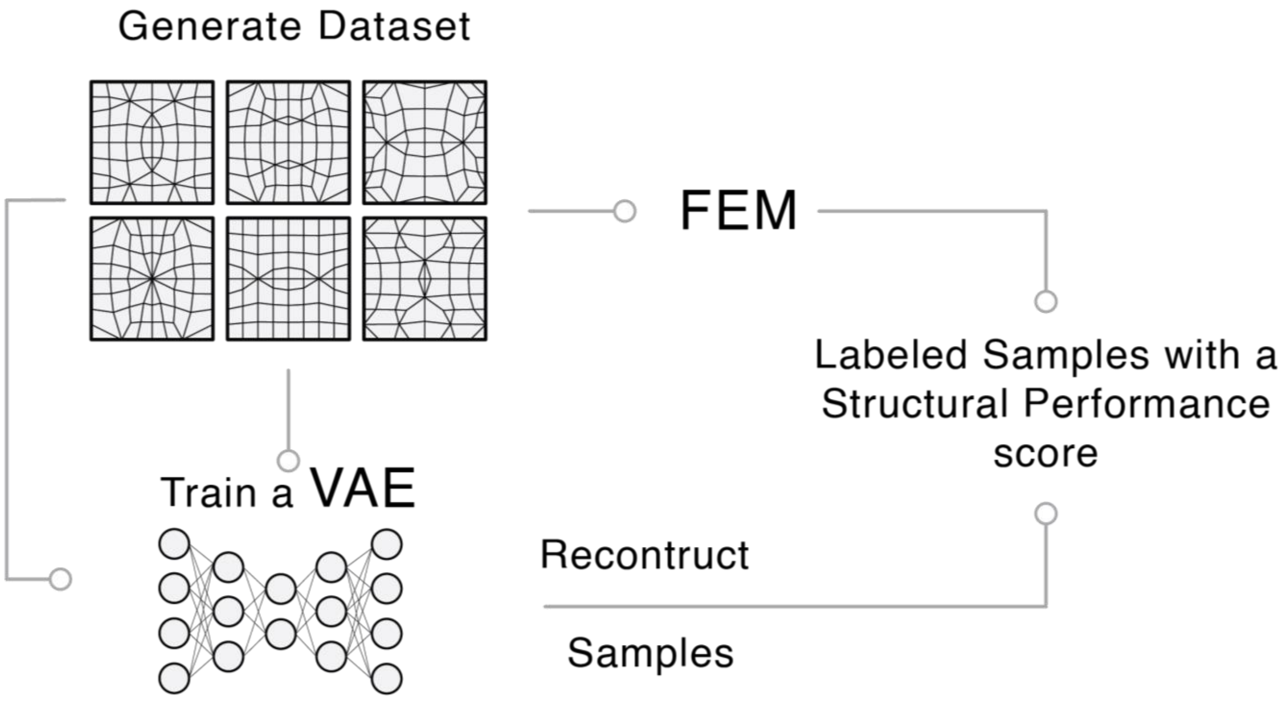
Generate the Dataset



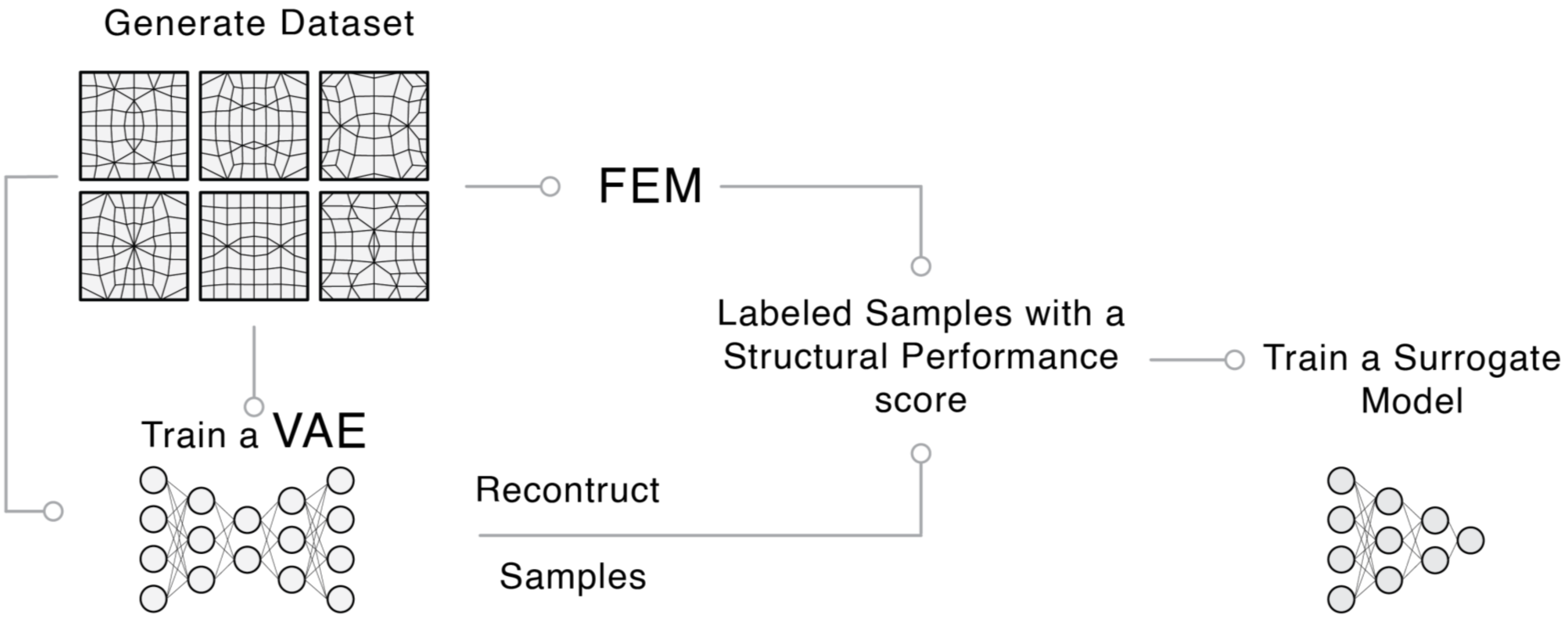
Generate the Dataset



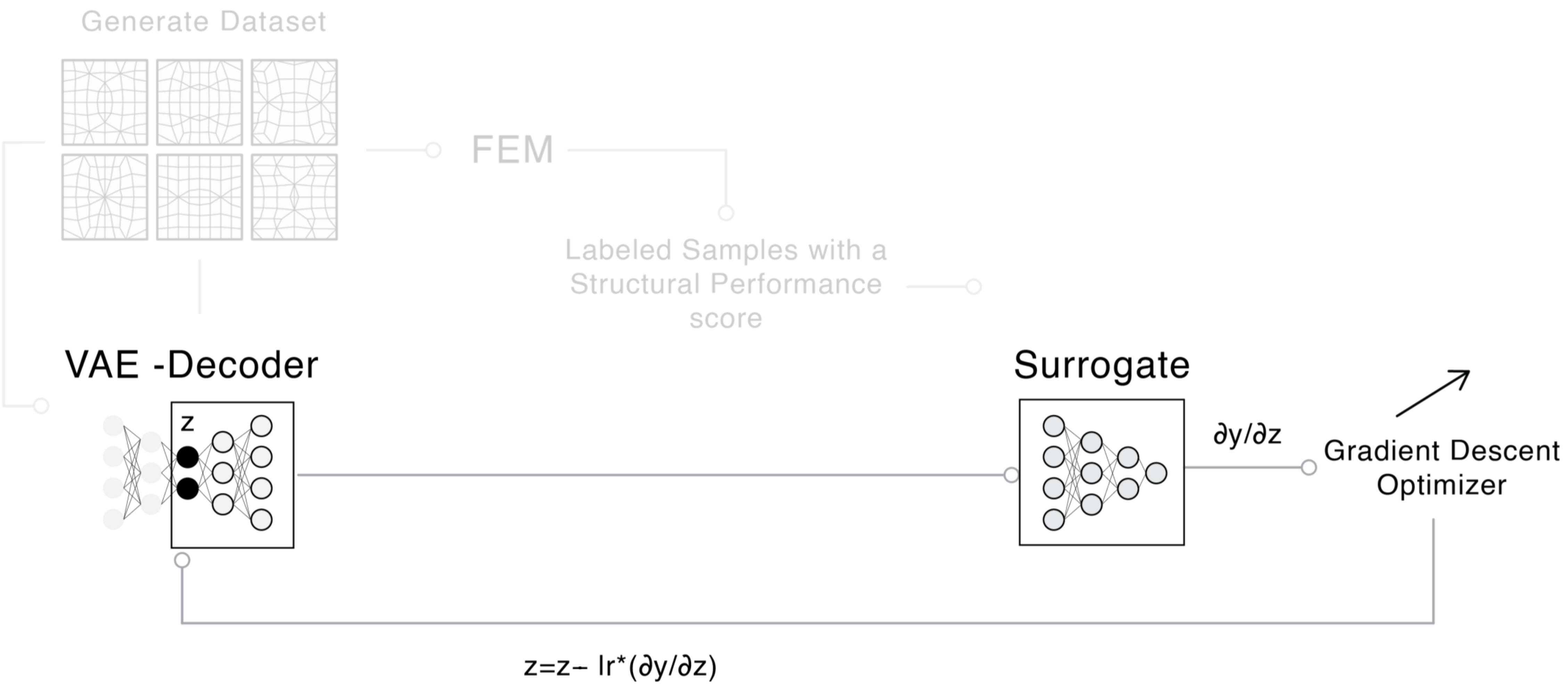
Generate the Dataset

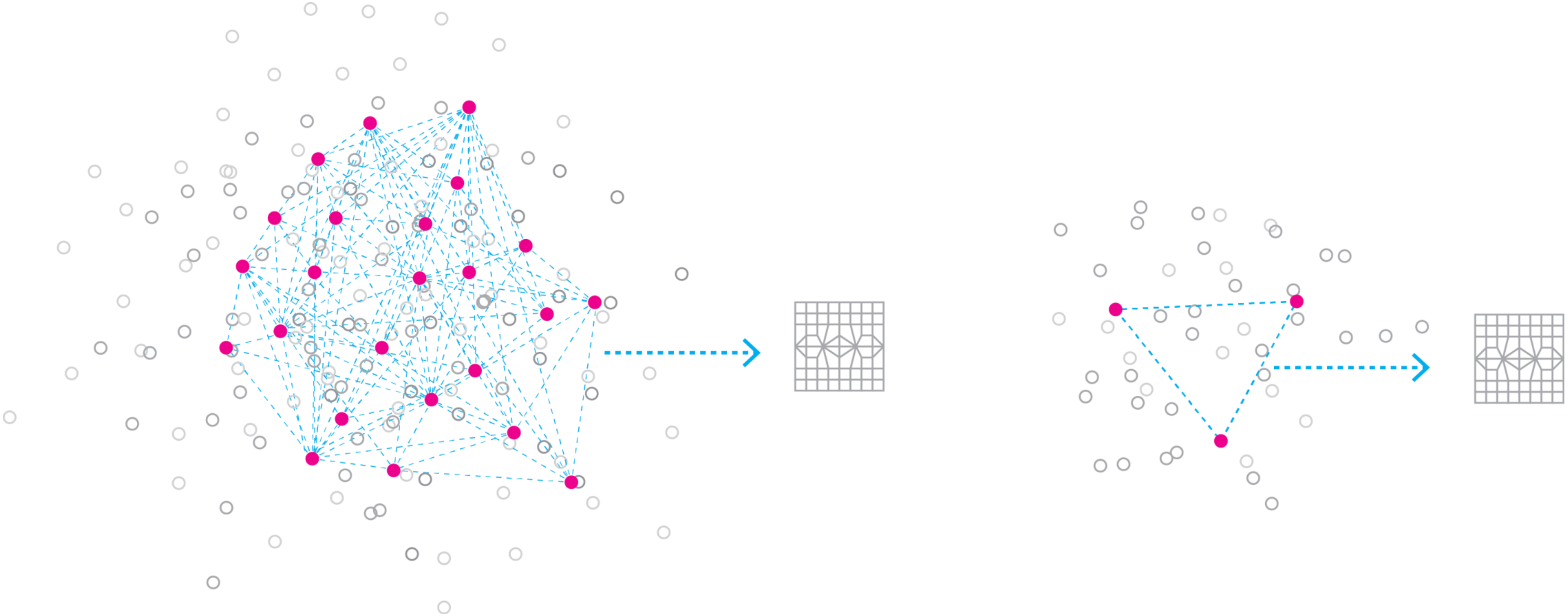


Generate the Dataset



Generate the Dataset





Main Research Question?

- Can an **AI** based framework generate **new structurally effective** solutions, in relation to the dataset that was used for training? This would prove that AI can be a powerful creative assistant for designers and engineers and could potentially help expand the possibilities of **Generative Design**.

Sub-questions

- Can a Variational Autoencoder be trained to generate mesh tessellations from which shell structures occur?
- What form of data can be used to train a Variational Autoencoder to generate mesh tessellations?
- Can a surrogate model learn to predict the structural performance of decoded graph networks that represent mesh tesselations?
- Can a Gradient Descent Optimizer propagate back to encoded data to search for optimum solutions?

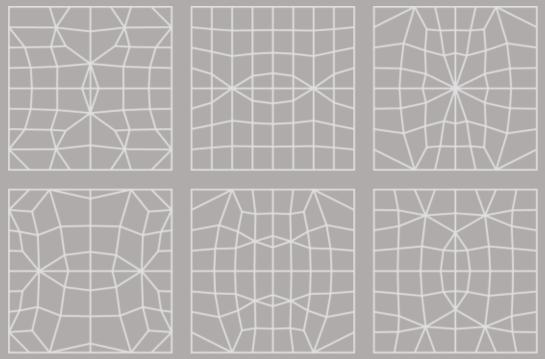
Objectives

- **Generate** a novel dataset of at least **1000 samples**.
- **Pre-process** the dataset's samples to create data appropriate to be used for training AI models.
- Develop an **appropriate architecture** for a generative model (**VAE**).
- Develop an **appropriate architecture** for a **surrogate** model.

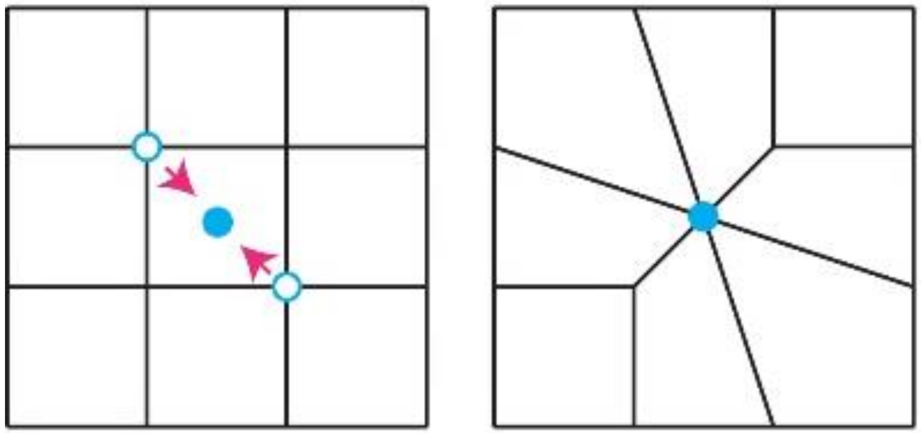
Boundary Conditions

- The designs of the dataset will be restricted in terms of their **shape and pattern**.
- The performance indicator of the workflow is only **structural performance**.
- The generative model that will be used is that of the **VAE**.

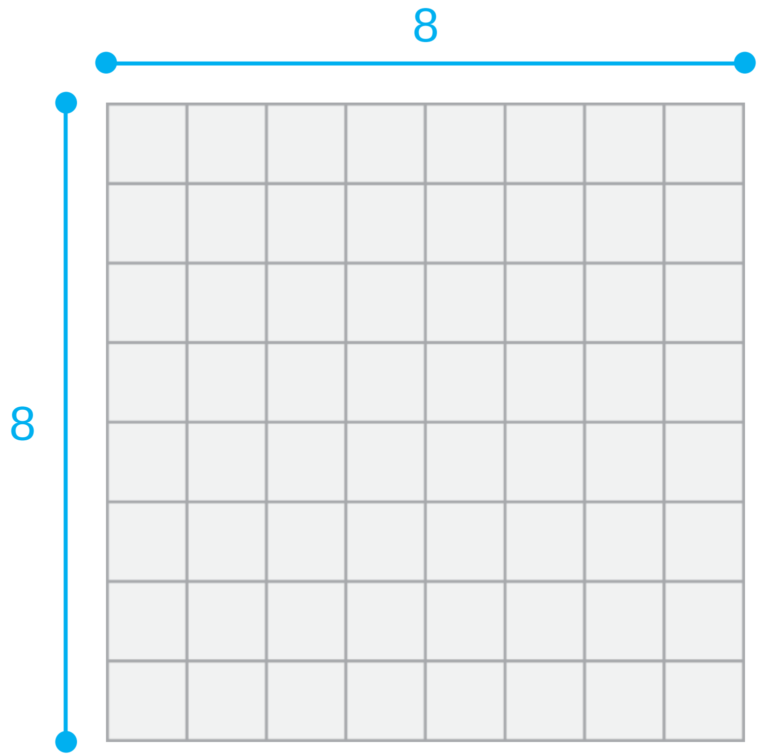
Dataset Generation



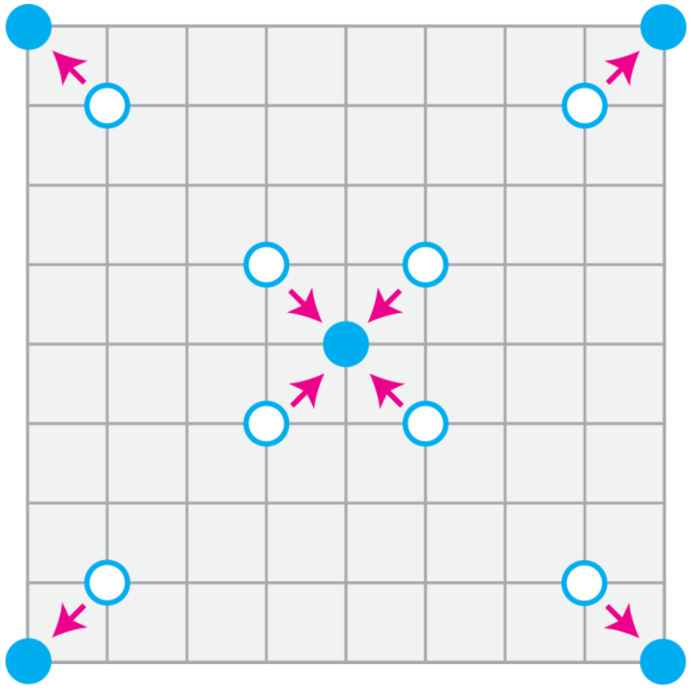
Dataset Creation



Dataset Creation

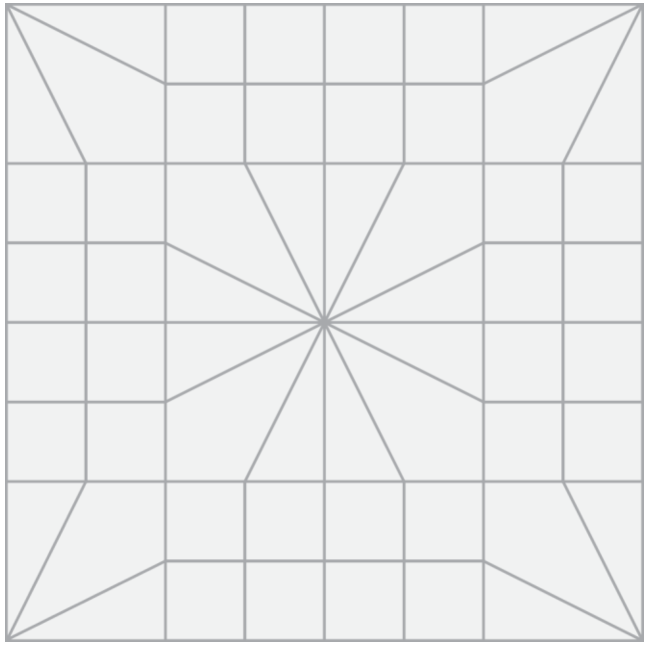


Initial Mesh

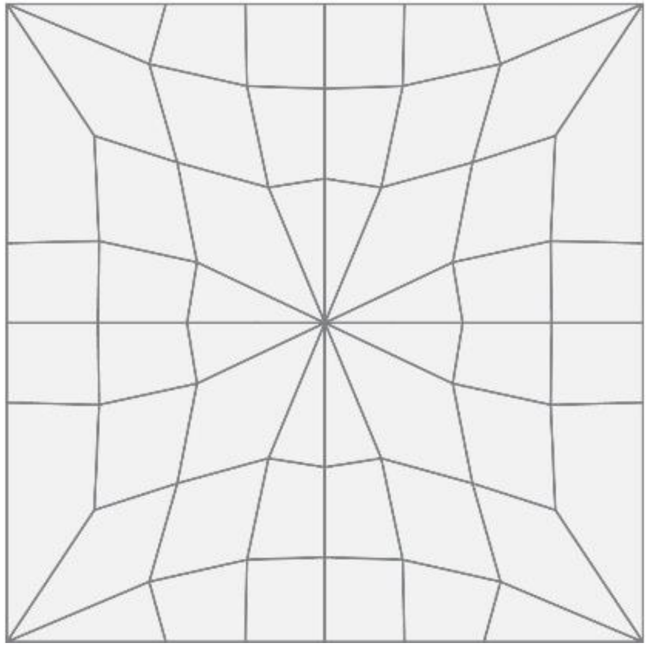


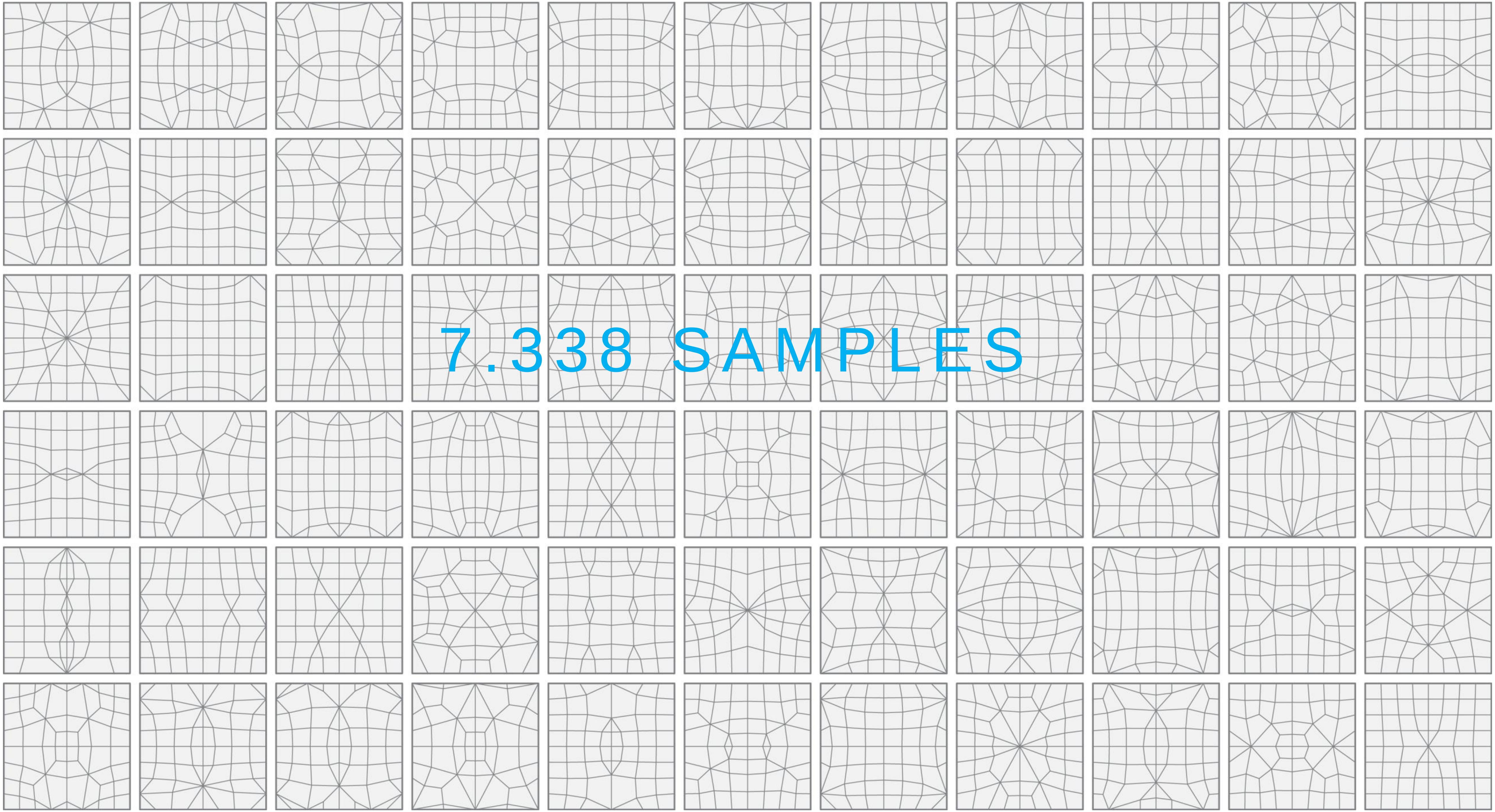
Vertices to join

Dataset Creation



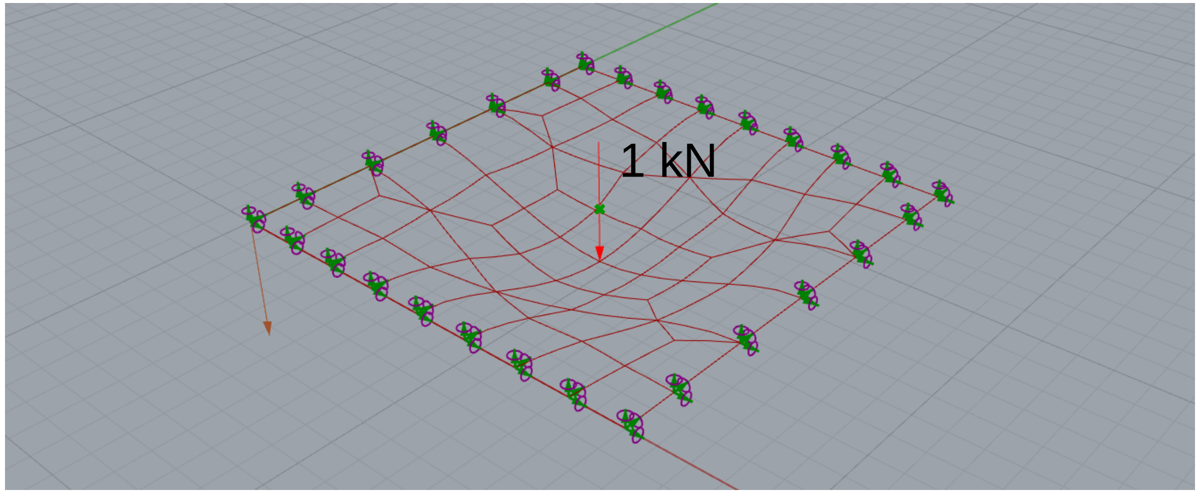
Dataset Creation




7.338 SAMPLES

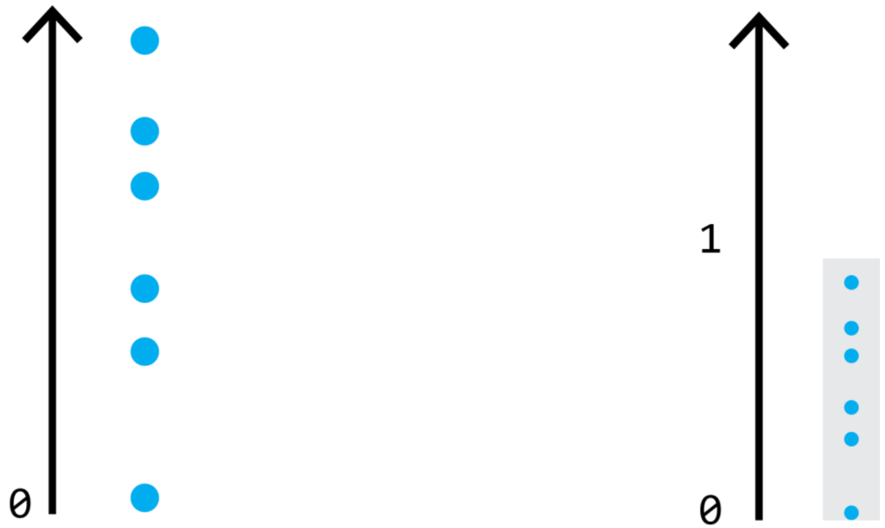
Simulate with FEM

Labelling the dataset



FEM simulation with Karamba3D

1. The **Maximum Displacement** in cm.
2. The **Maximum Utilization** (ratio between the tensile or compressive strength and the maximum allowable stress)
3. The **Mass** of the structure in kg.



Normalization

Labelling the dataset

$$\textbf{Performance} =$$

$$0,4 \times \textit{Normalized Displacement} + 0,4 \times \textit{Normalized}\ \textbf{Utilization} + 0,2 \times \textit{Normalized}\ \textbf{Mass}$$

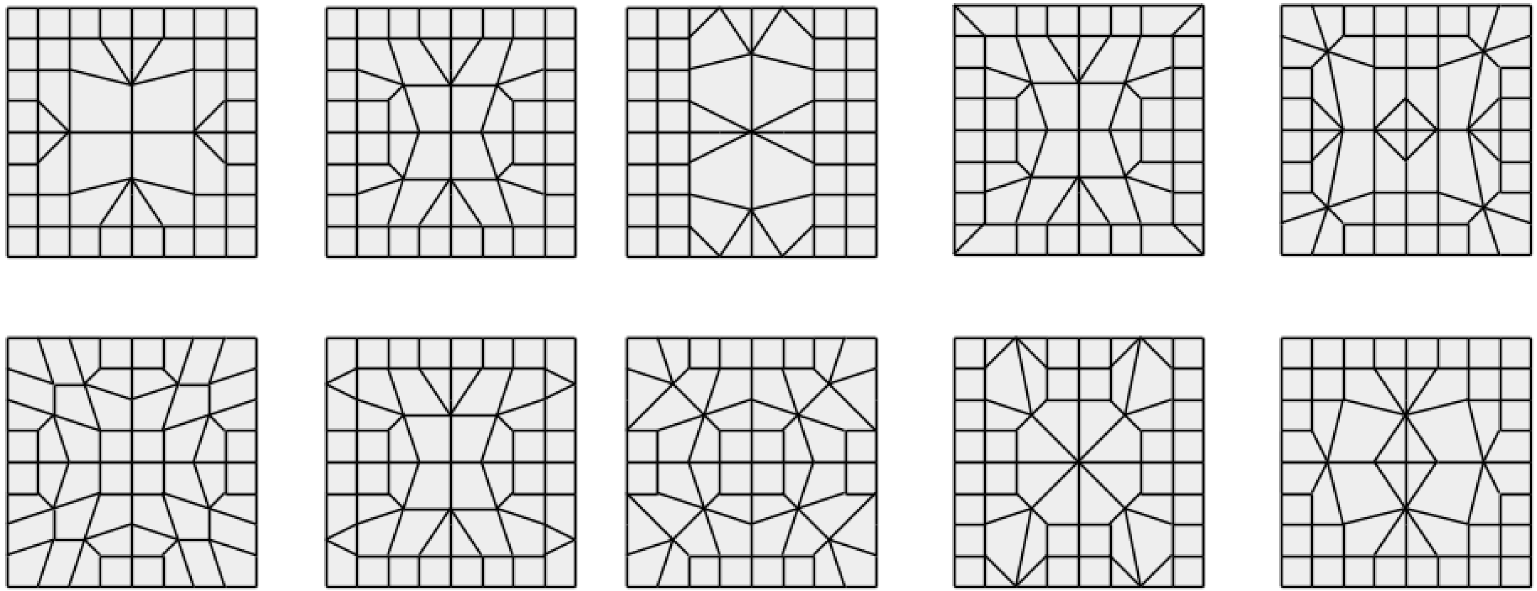
$$\downarrow$$

$$\textit{Normalized Performance}$$

Maximum_displacement[cm]	Utilization	Mass[kg]	Norm_dis	Norm_Util	Norm_Mass	Performance	Norm_Performance
8.176643	0.460882	1650.8257	0.4015813	0.211992	0.603486	0.3661266	0.453239807
8.230162	0.465895	1659.1931	0.4129745	0.218017	0.627971	0.3779906	0.474080792
8.552545	0.597416	1595.0081	0.4816043	0.376071	0.44015	0.4311002	0.56737568
7.333896	0.448868	1647.0965	0.222175	0.197555	0.592573	0.2864065	0.313199618
8.075146	0.512321	1632.9862	0.3799743	0.273809	0.551283	0.3717699	0.463153143
7.958961	0.396704	1630.4362	0.3552405	0.134867	0.543821	0.3048072	0.345523125
8.04153	0.419472	1637.4375	0.372818	0.162228	0.564309	0.3268802	0.384297709
7.274894	0.539231	1671.1759	0.2096145	0.306148	0.663035	0.338912	0.405433289
7.343729	0.818736	1630.2022	0.2242683	0.642041	0.543136	0.4551511	0.609624679

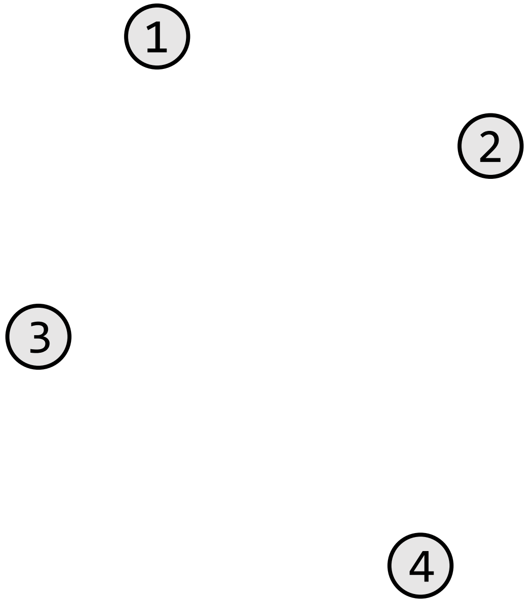
Excluding best performing designs

Mesh Index	Norm Performance
1592	0
916	0.085279428
2871	0.099995339
3178	0.101466942
585	0.101741484
2448	0.114673073
468	0.119237033
1093	0.11999031
3374	0.124247914
2286	0.126232537
3487	**0.12844538**
3659	0.129979758
3143	0.132732184
370	0.136648056
3401	0.137413625
131	0.140570862



Data pre-process

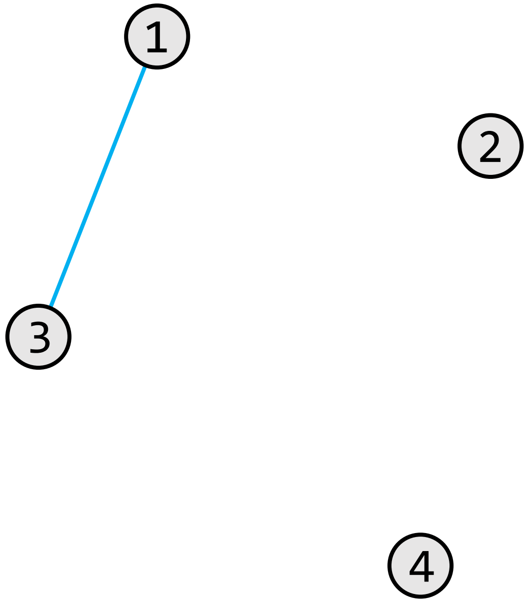
Data pre-process



	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0

Adjacency Matrix

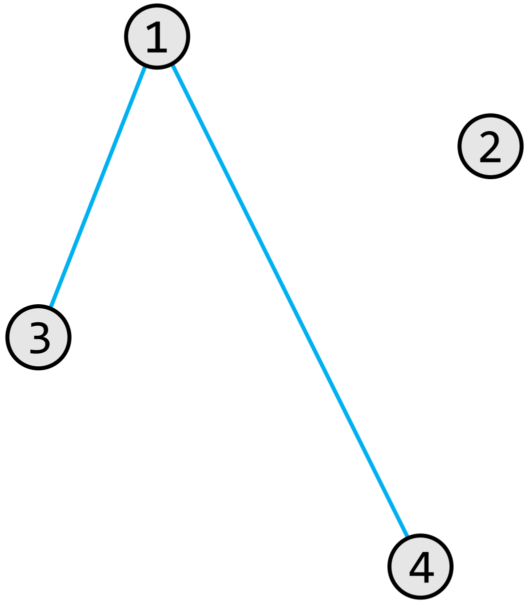
Data pre-process



	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	0	0	0
3	0	1	0	0	0
4	0	0	0	0	0

Adjacency Matrix

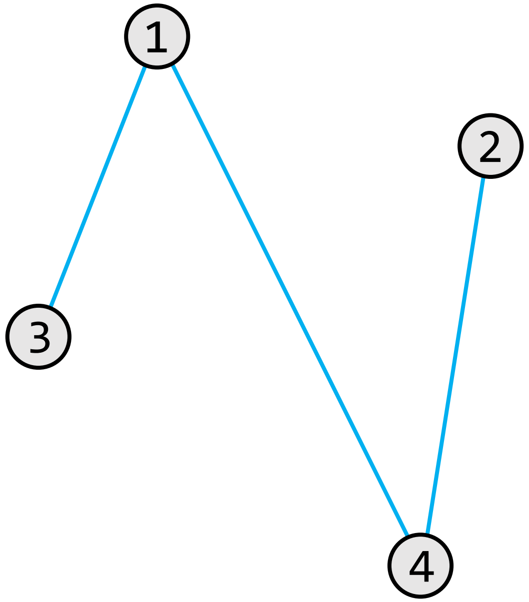
Data pre-process



	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	0	0	0
3	0	1	0	0	0
4	0	1	0	0	0

Adjacency Matrix

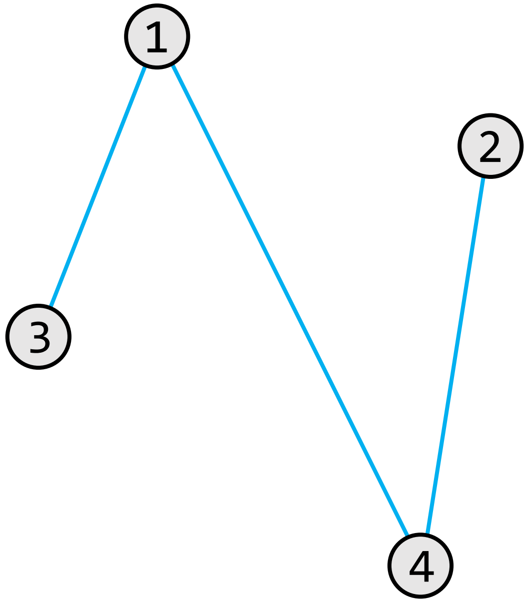
Data pre-process



	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	0	0	1
3	0	1	0	0	0
4	0	1	1	0	0

Adjacency Matrix

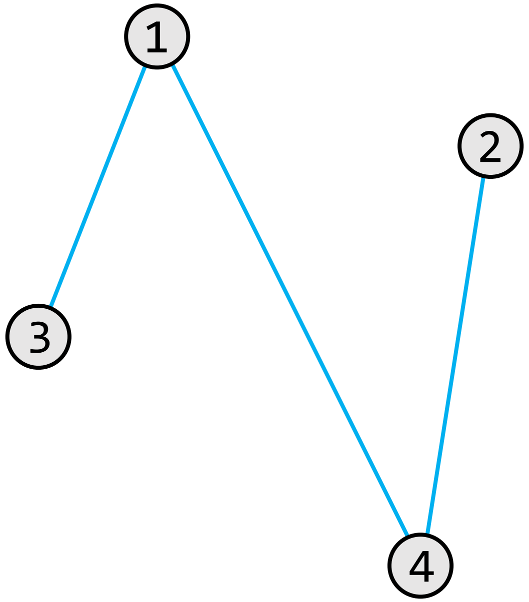
Data pre-process



	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	0	0	1
3	0	1	0	0	0
4	0	1	1	0	0

Adjacency Matrix

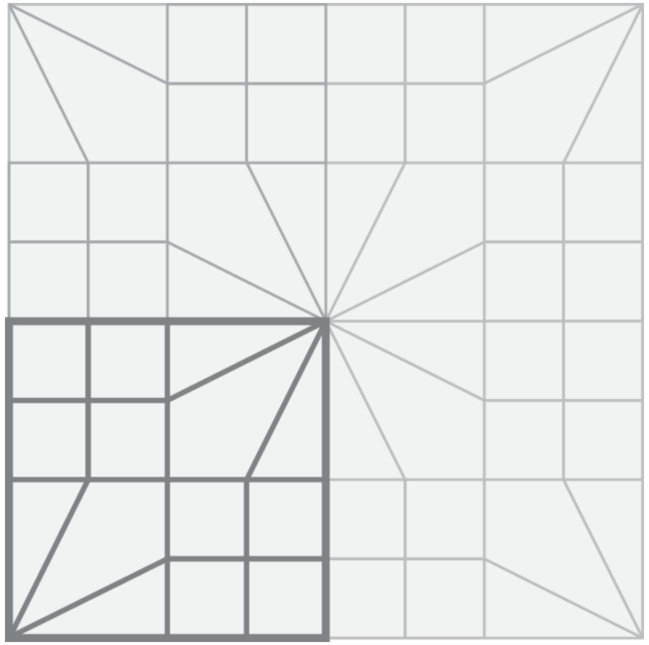
Data pre-process



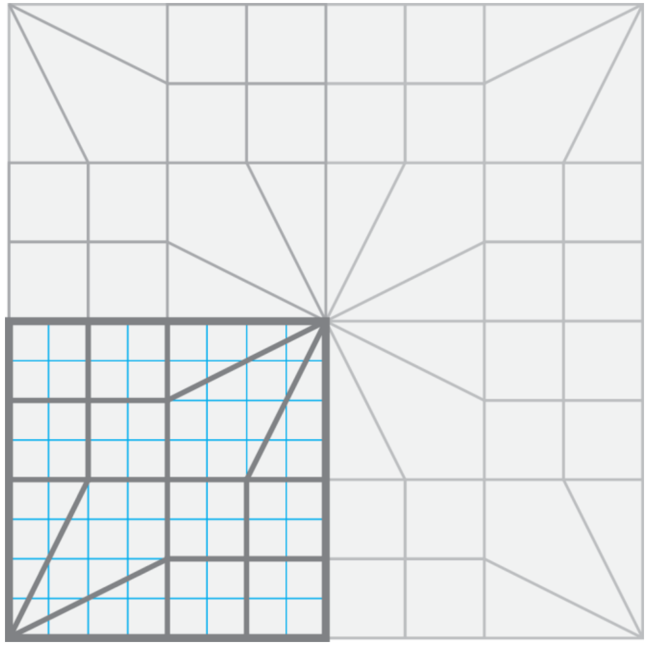

```
[
 0,
 0,0,
 0,0,0,
 0,1,0,0
 0,1,1,0,0
]
```

Tensor

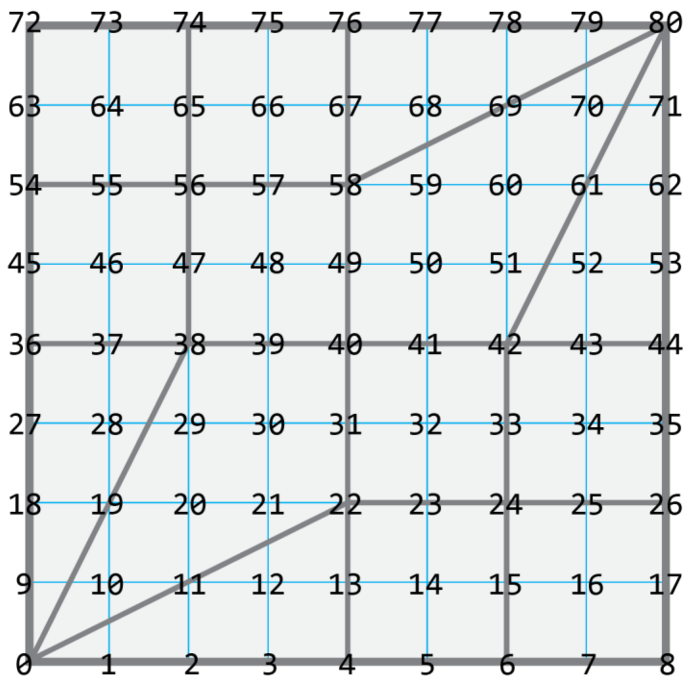
Data pre-process



Data pre-process



Data pre-process



Number of Vertices in the 1/4 Mesh: **81**

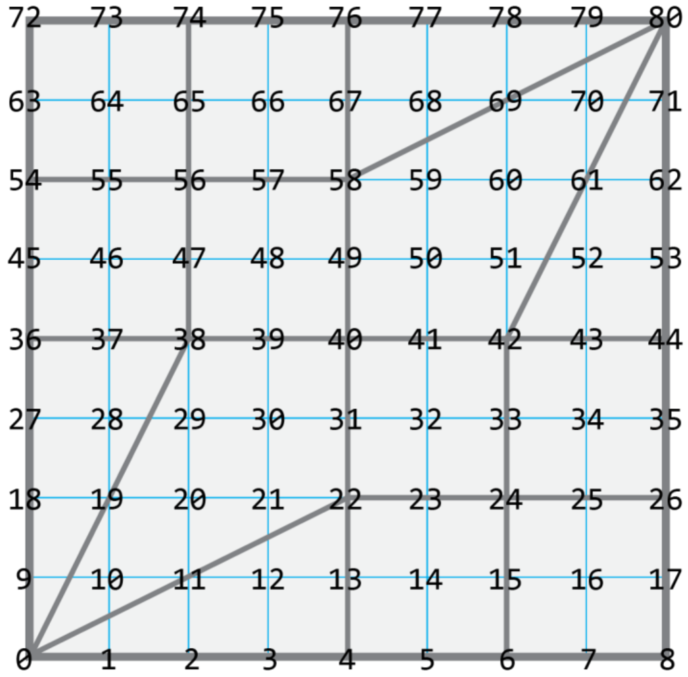
81

81

	0	1	2	…	78	79	80
0	0	1	0		0	0	0
1	1	0	1	…	0	0	0
2	0	1	0	…	0	0	0
…	…	…	…	…	…	….	….
78	0	0	0	…	0	1	0
79	0	0	0	…	1	0	1
80	0	0	0	…	0	1	0

Shape : (81,81)

Data pre-process



Number of Vertices in the 1/4 Mesh: **81**

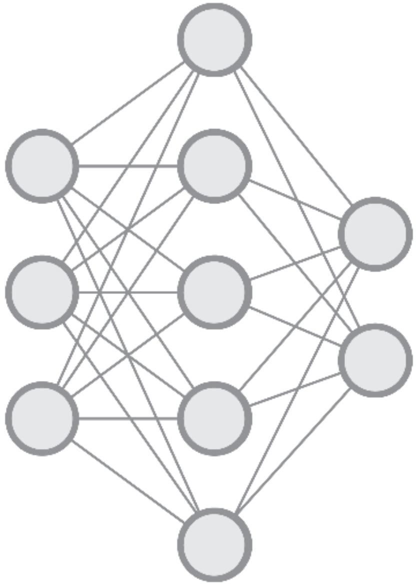
	0	1	2	...	78	79	80
0	0	1	0		0	0	0
1	1	0	1	...	0	0	0
2	0	1	0	...	0	0	0
...	...	...	...	...	...	....	....
78	0	0	0	...	0	1	0
79	0	0	0	...	1	0	1
80	0	0	0	...	0	1	0

[0,1,0,0,1,0,...,0,1,0]

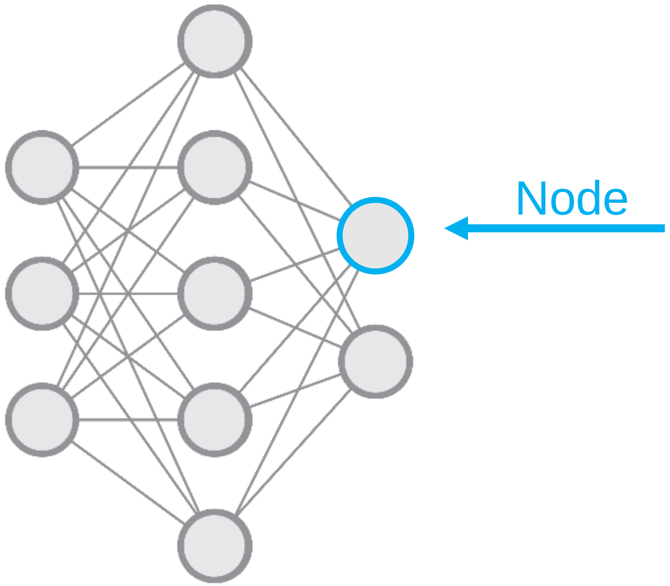
Tensor Shape: (3240)

AI-WORKFLOW

Neural Networks

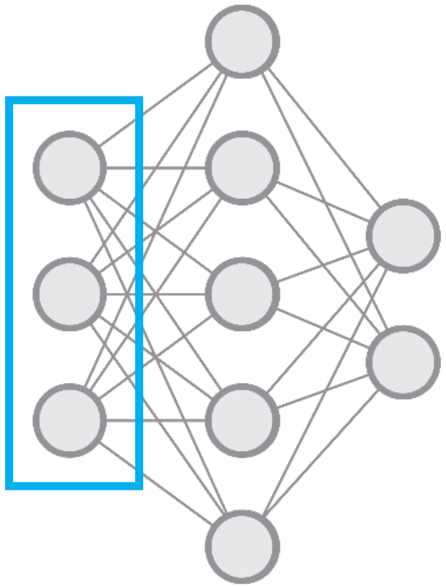


Neural Networks



Neural Networks

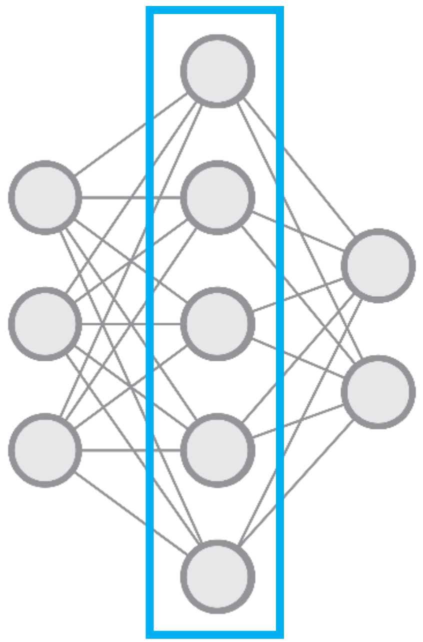
Simple Neural Network



Input Layer

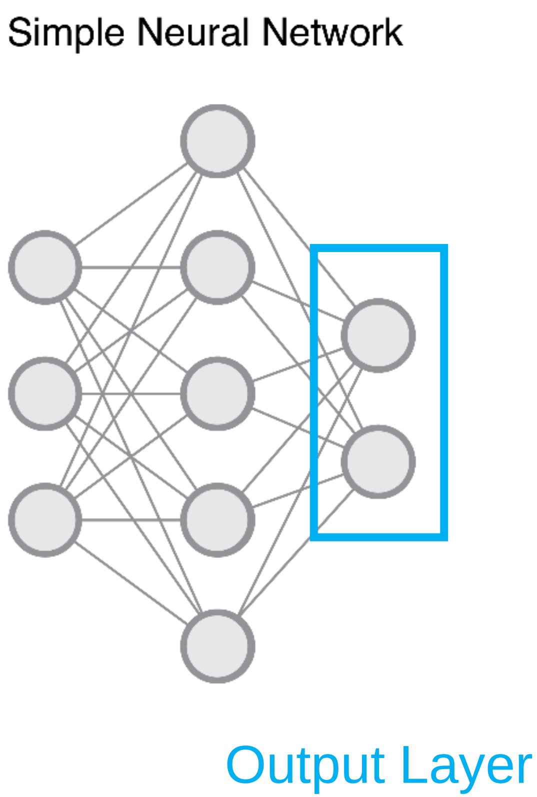
Neural Networks

Simple Neural Network

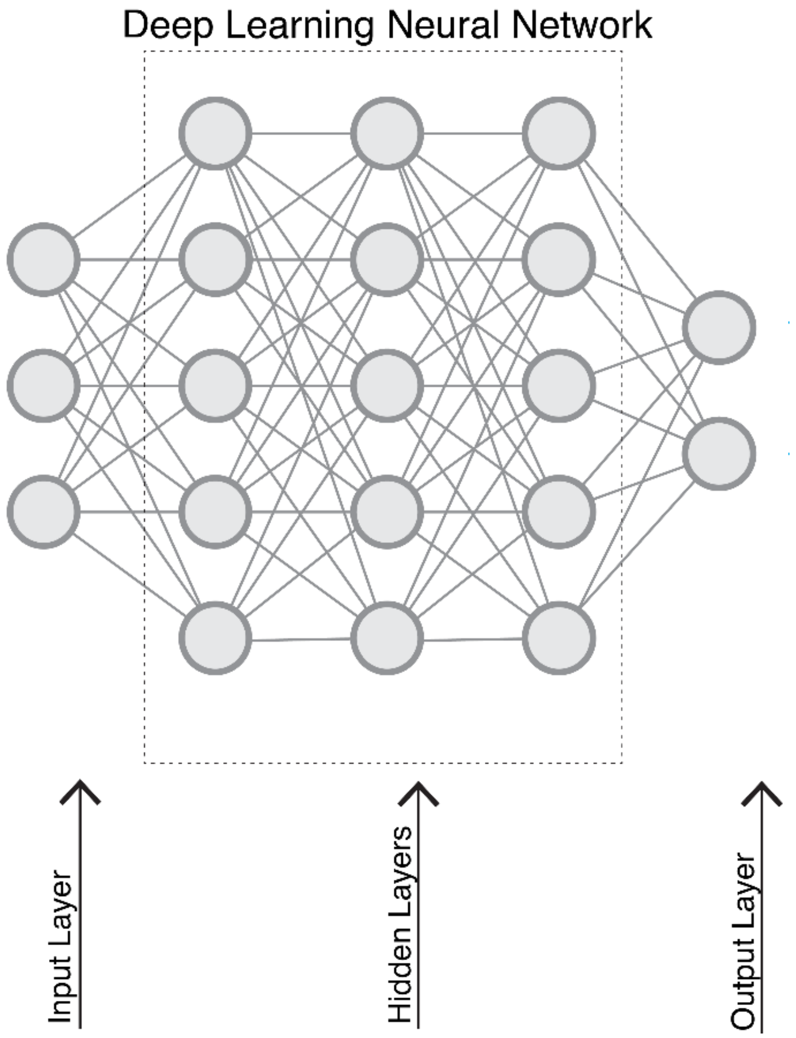


Hidden Layer

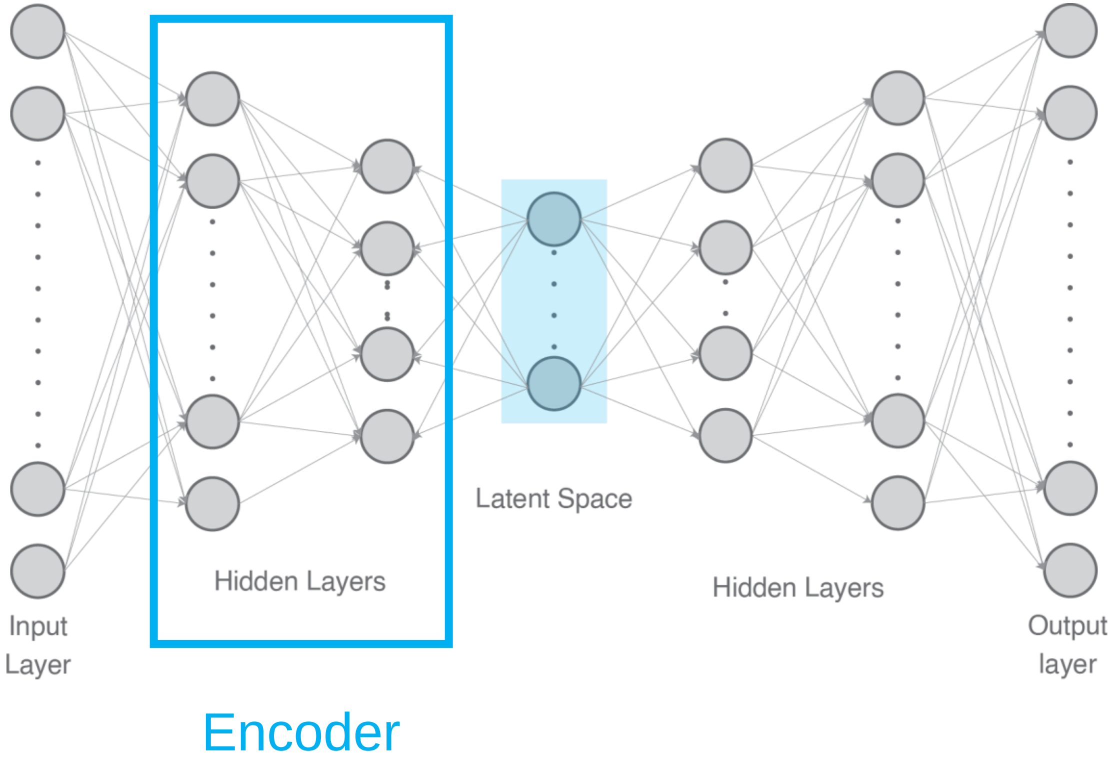
Neural Networks



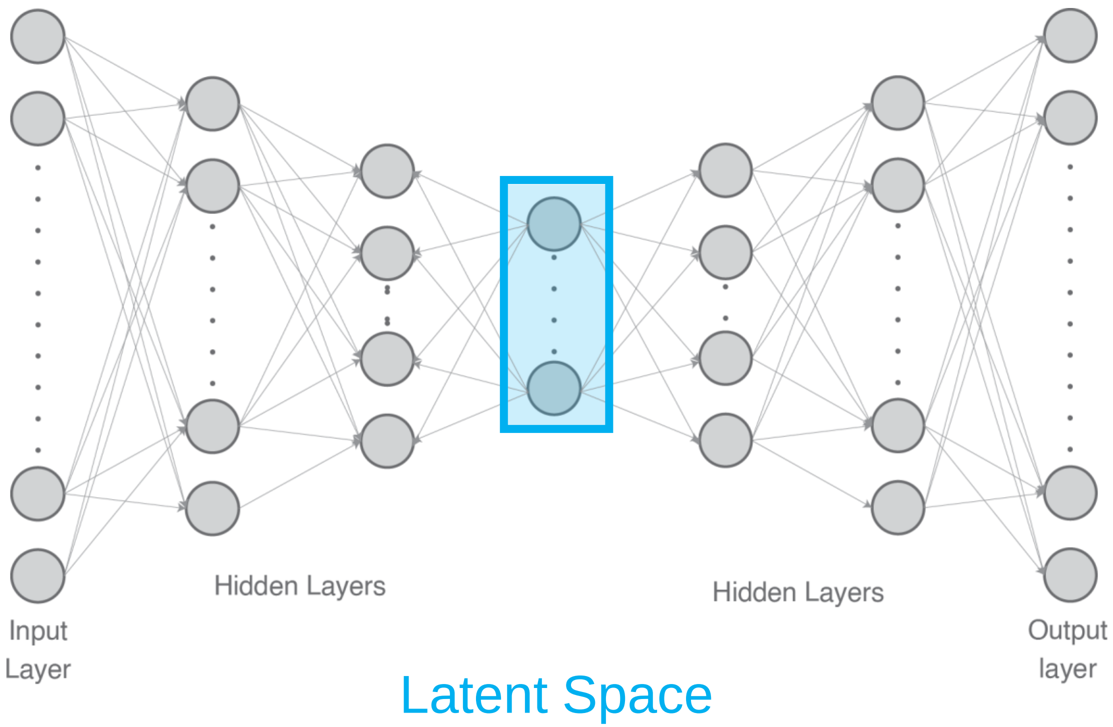
Neural Networks



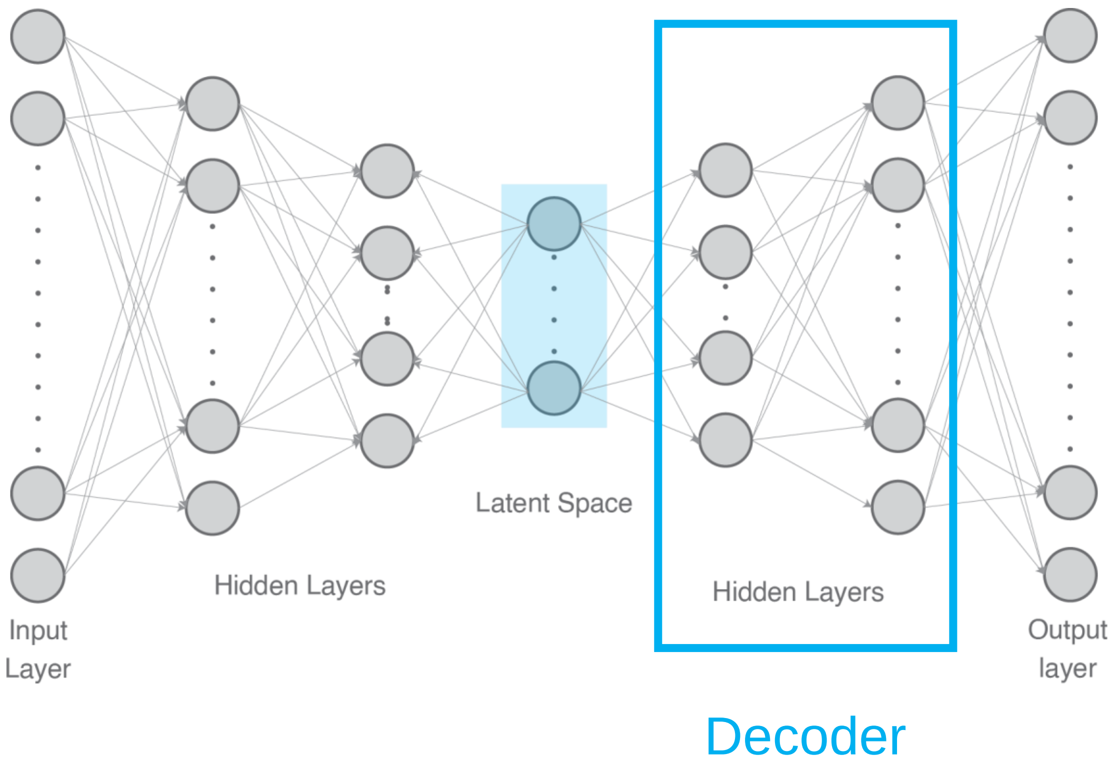
Generative Model : Variational Autoencoder (VAE)



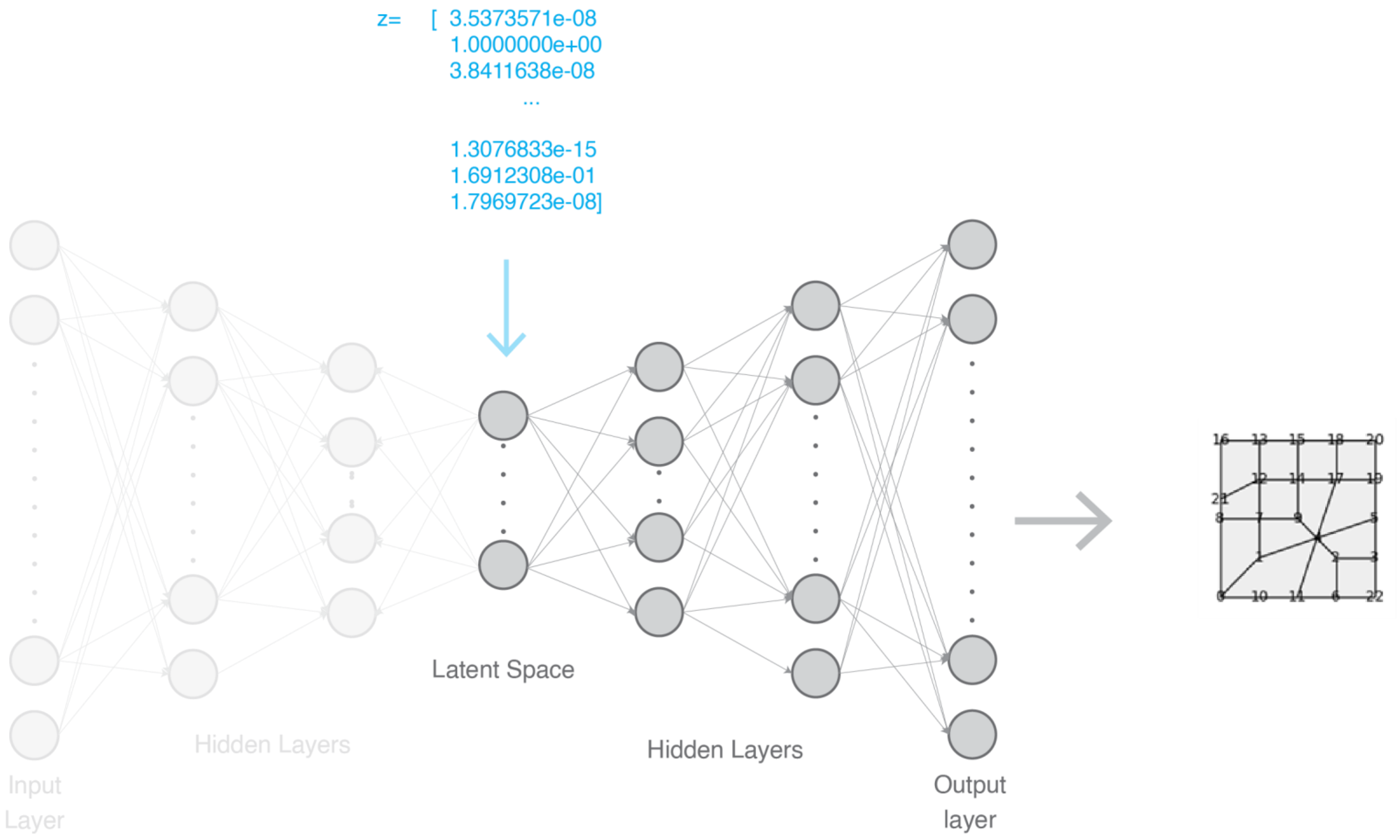
Generative Model : Variational Autoencoder (VAE)



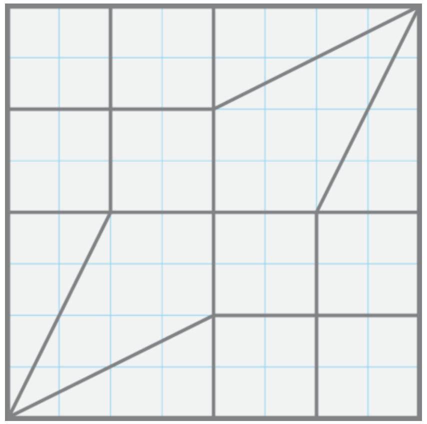
Generative Model : Variational Autoencoder (VAE)



Generative Model : Variational Autoencoder (VAE)

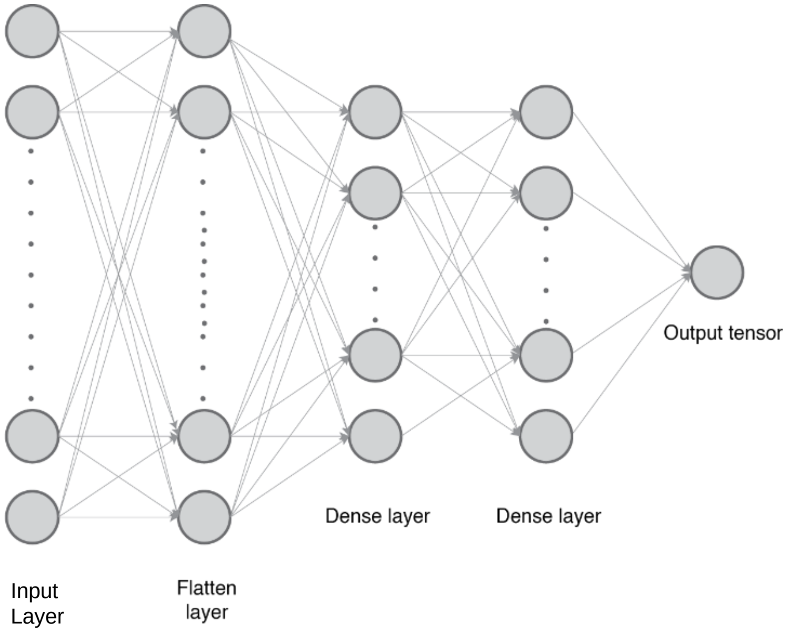


Surrogate Model



[0,1,0,0,1,0,...,0,1,0]

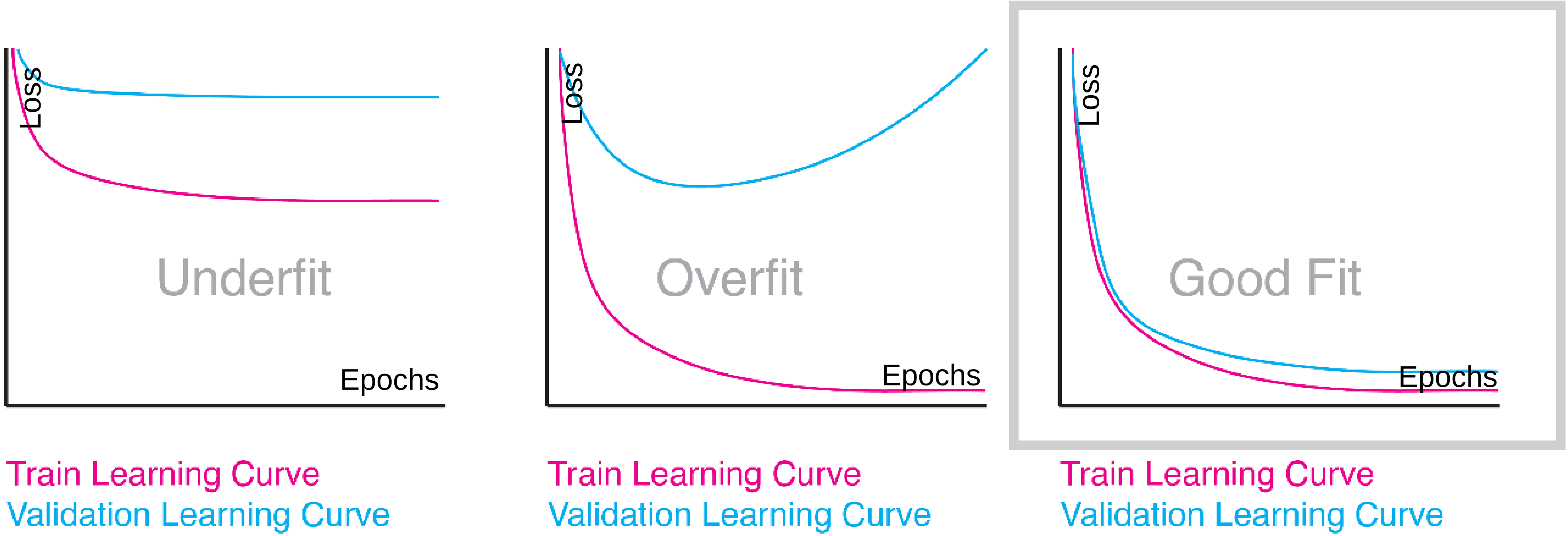
Tensor Shape
(3240)



Estimated
Performance

Learning Curves

```
loss function = error(output, expected output)
```

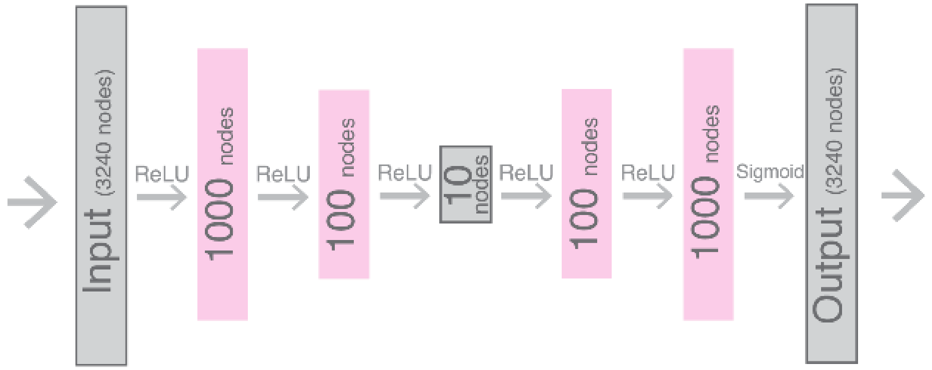


Variational Autoencoder

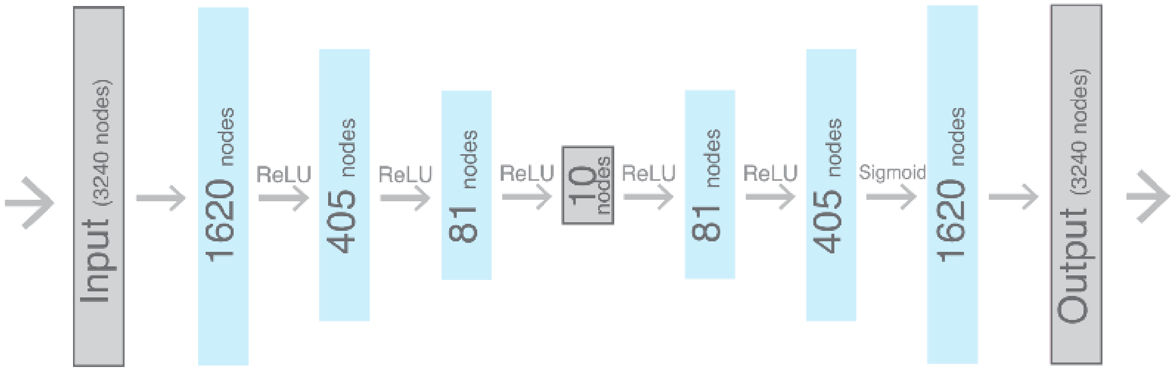
Variational Autoencoder

- Epochs: 500
- Batch size: 64
- Adam optimizer with learning rate 0.001

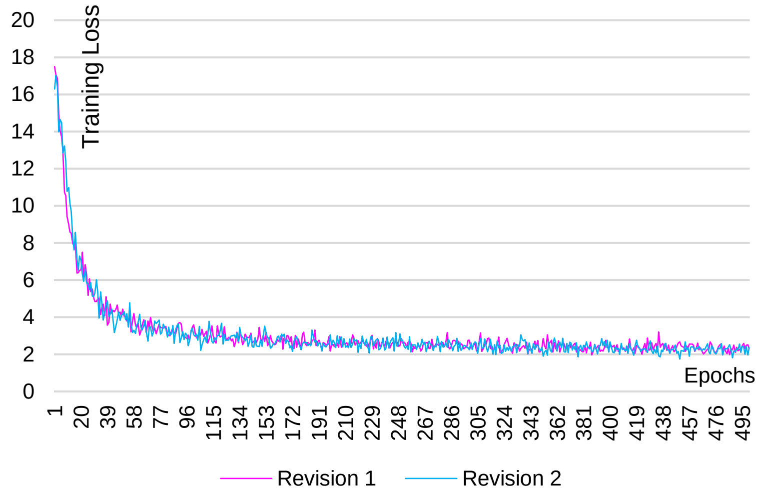
Revision 1



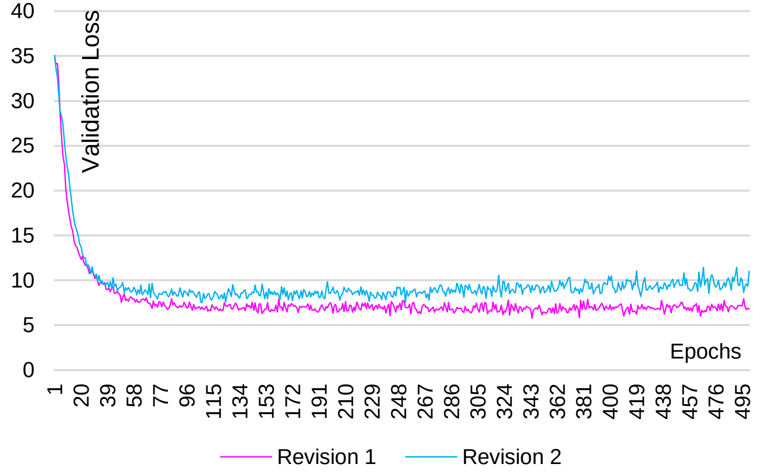
Revision 2



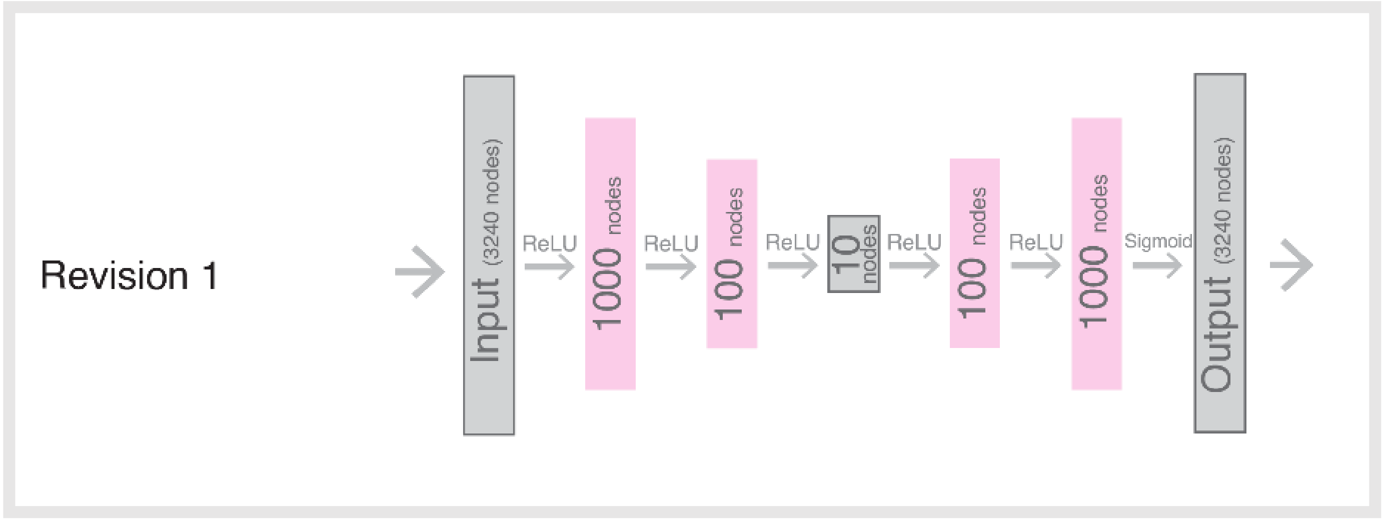
Variational Autoencoder

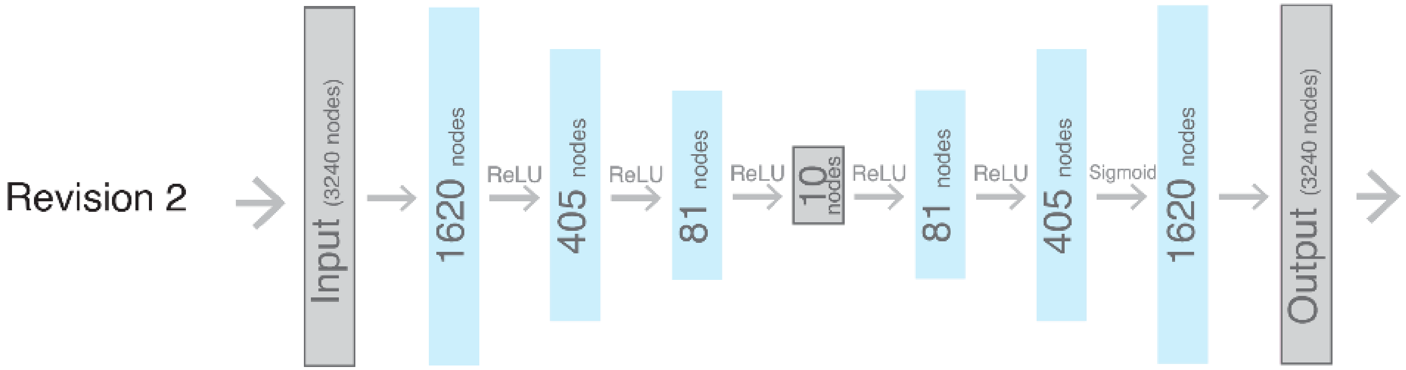


Training Loss after 500 epochs

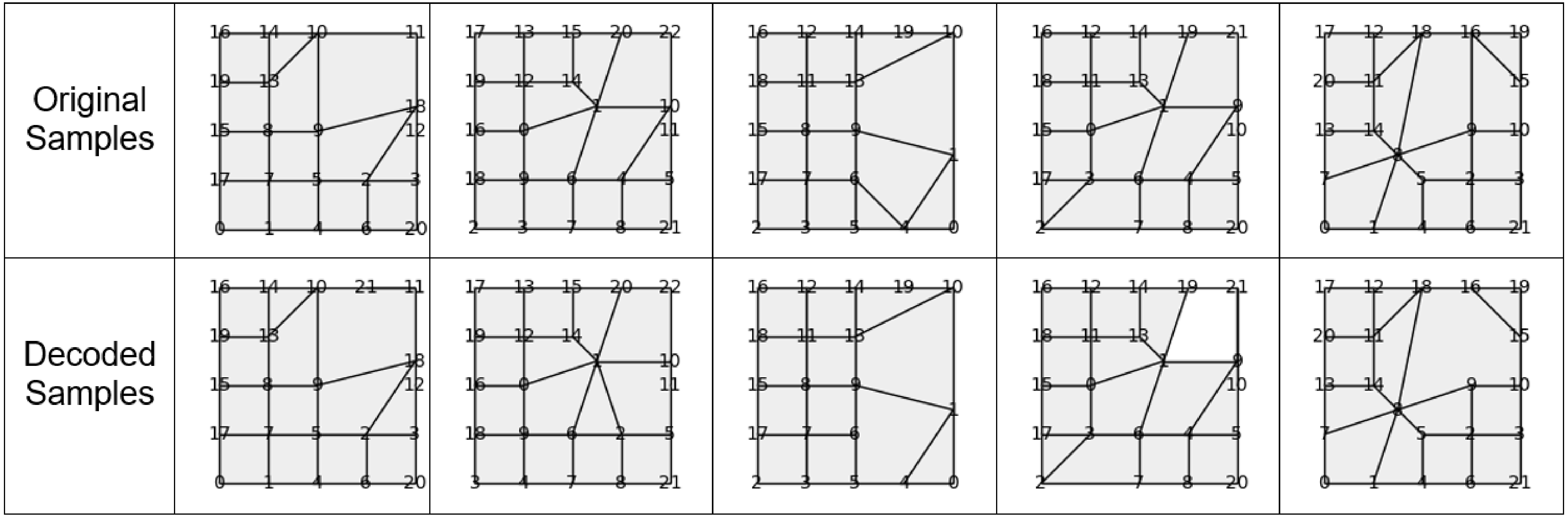


Validation Loss after 500 epochs

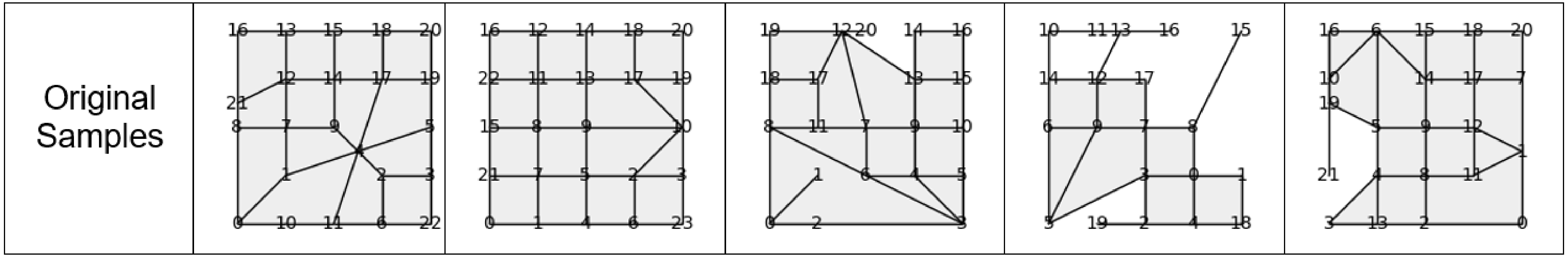




Variational Autoencoder



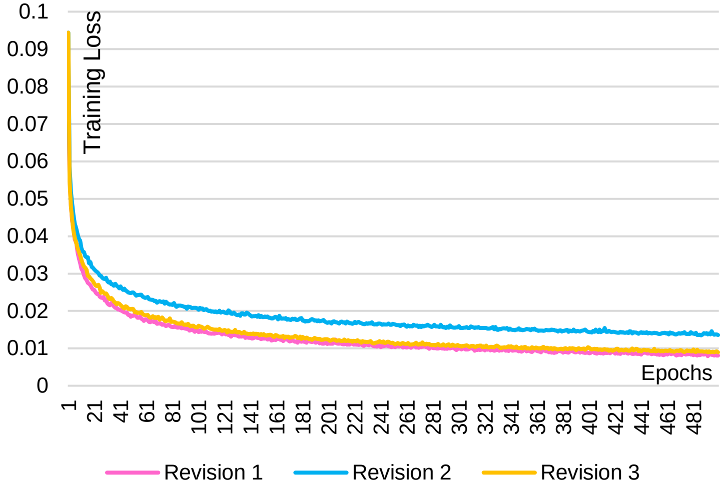
Some of the best performed samples and their decoded result



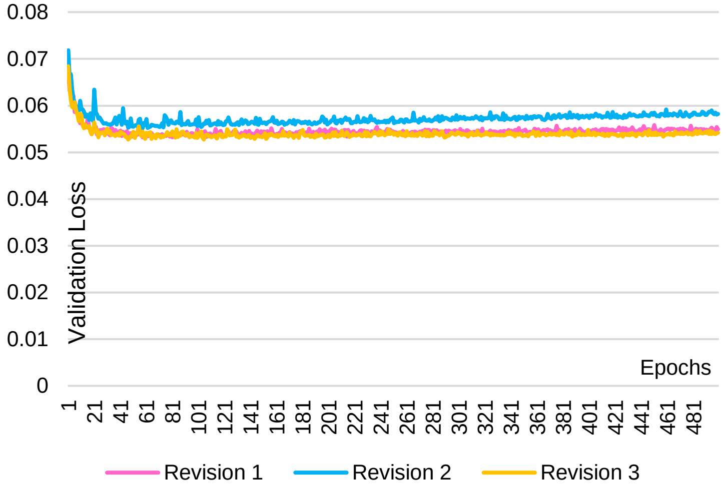
Random AI generated meshes

Surrogate Model

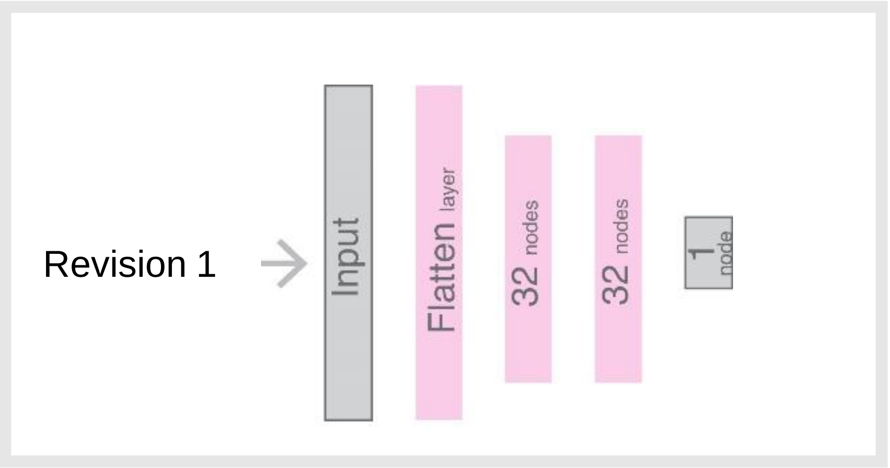
Surrogate Model

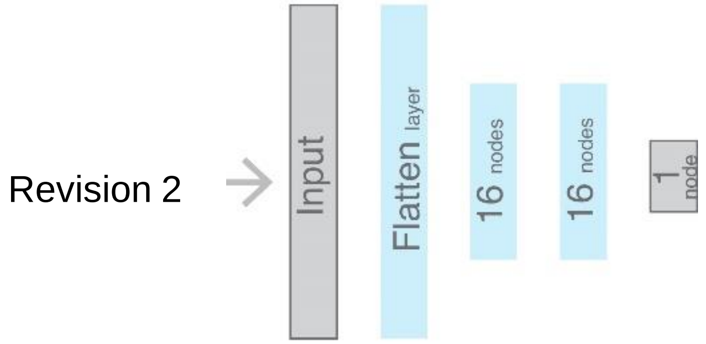


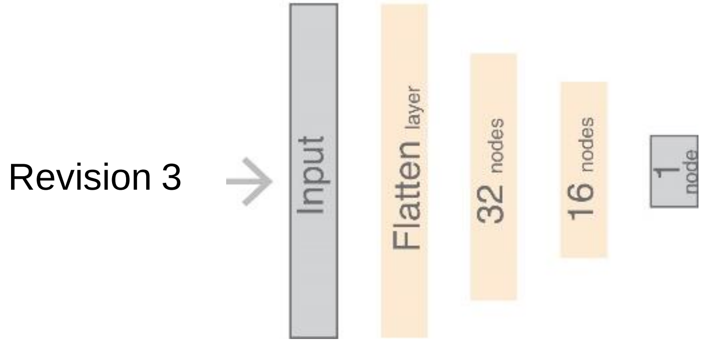
Loss after 500 epochs



Validation Loss after 500 epochs

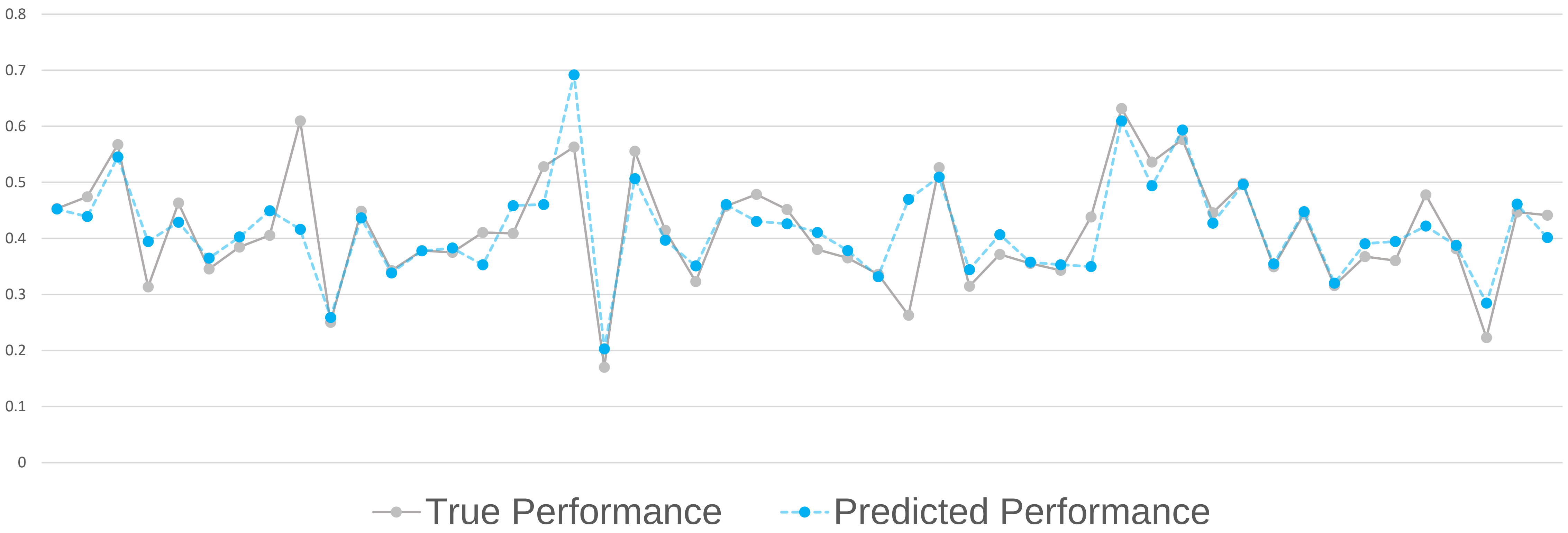




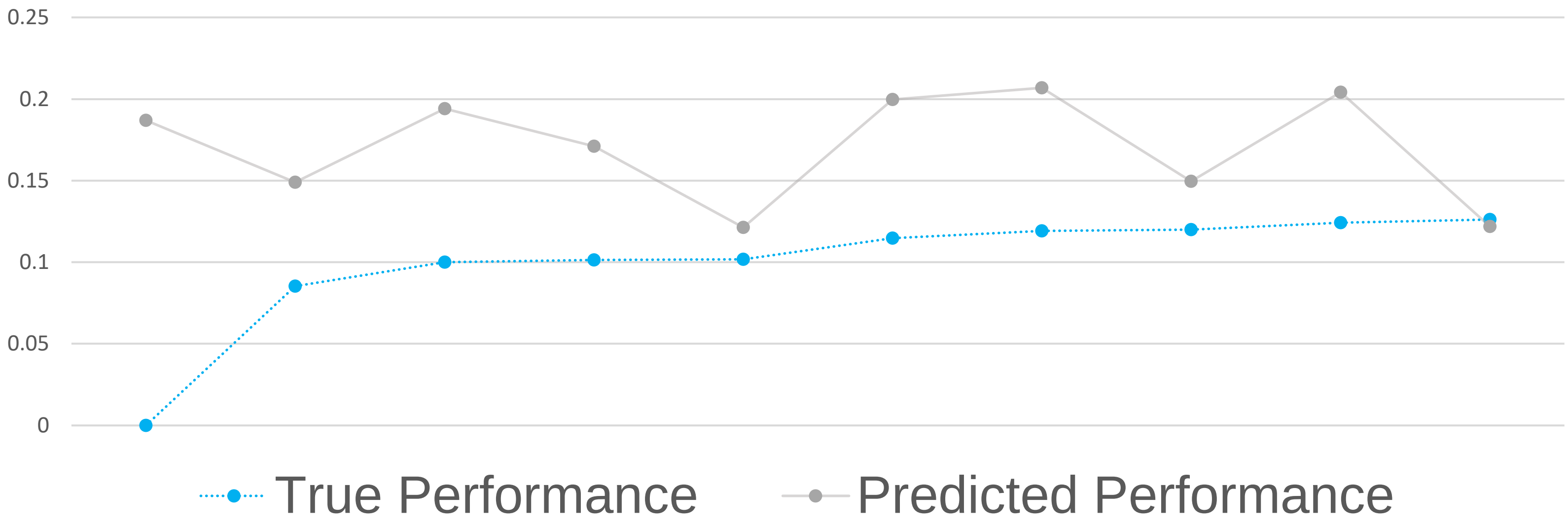


Surrogate Model

Evaluation of 50 samples excluded from training



Surrogate Model Evaluation of 10 best performing samples excluded from training



Gradient Descent Optimizer

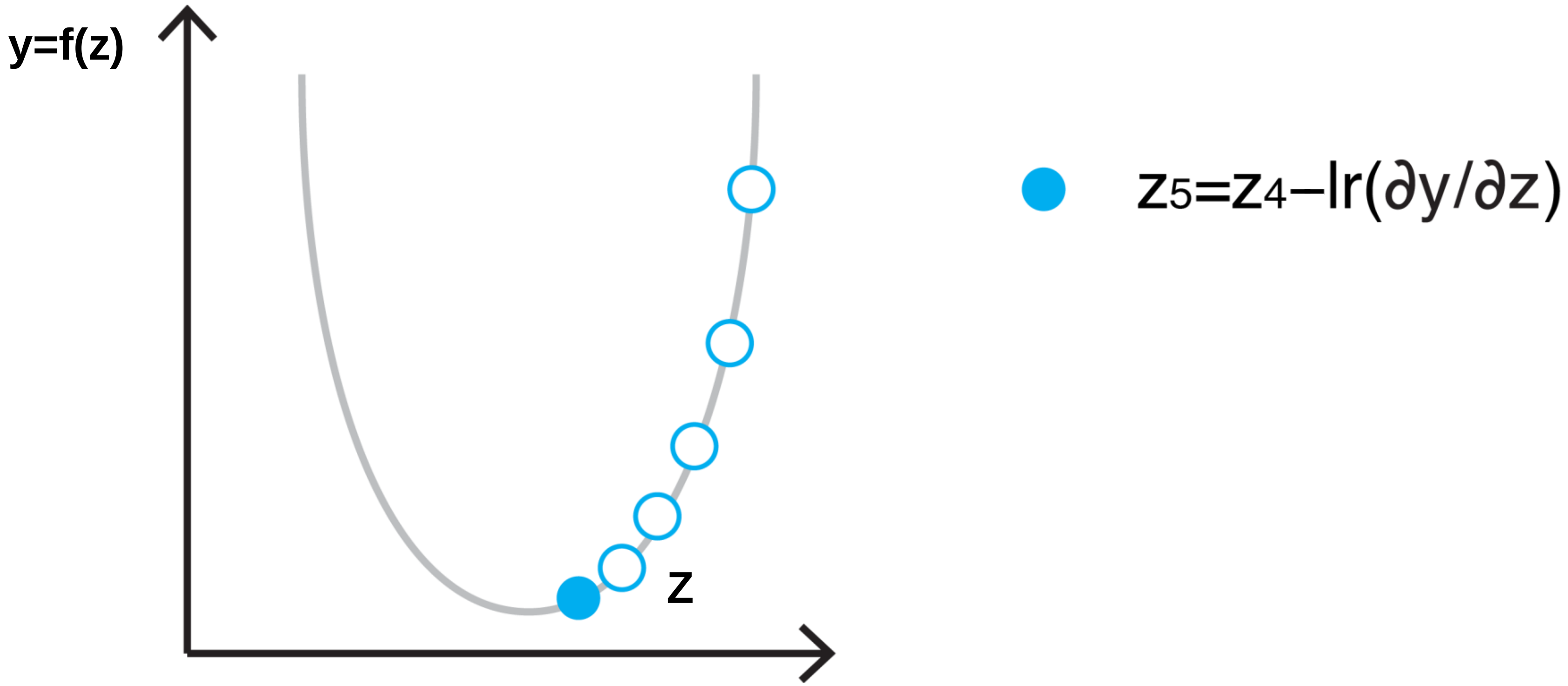
Gradient Descent Optimizer

$$Z = Z - lr\,\frac{\partial y}{\partial_z}\,(Z_0,\ Z_{n,})$$

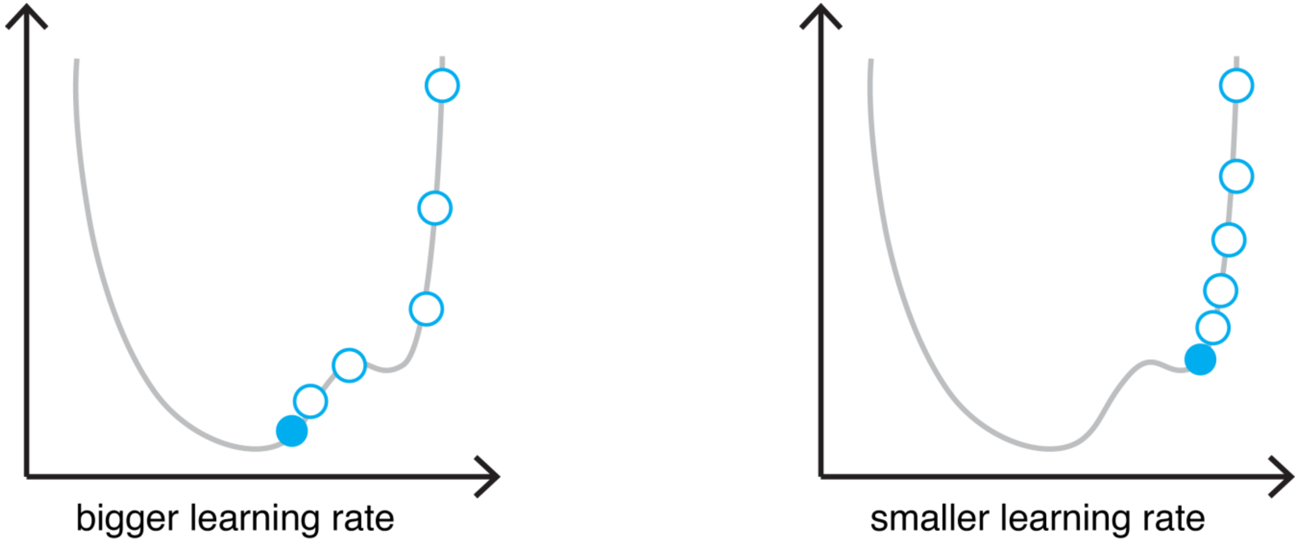
lr : Learning rate that determines how large the update or moving step is.

Z: The latent's space z vector to be updated

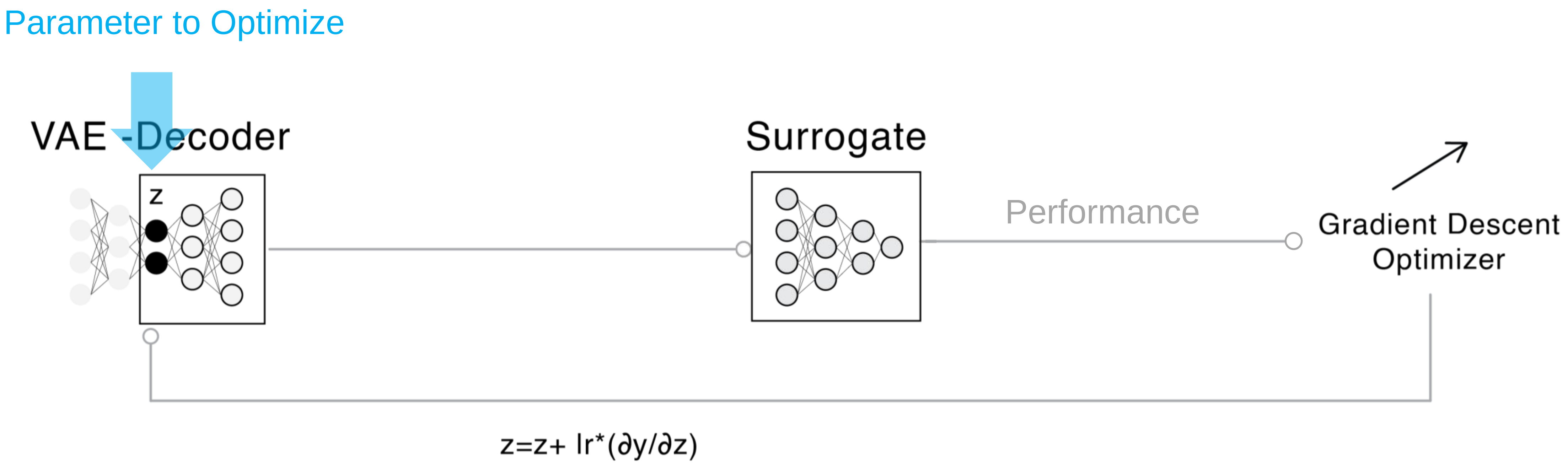
Y: Structural Performance



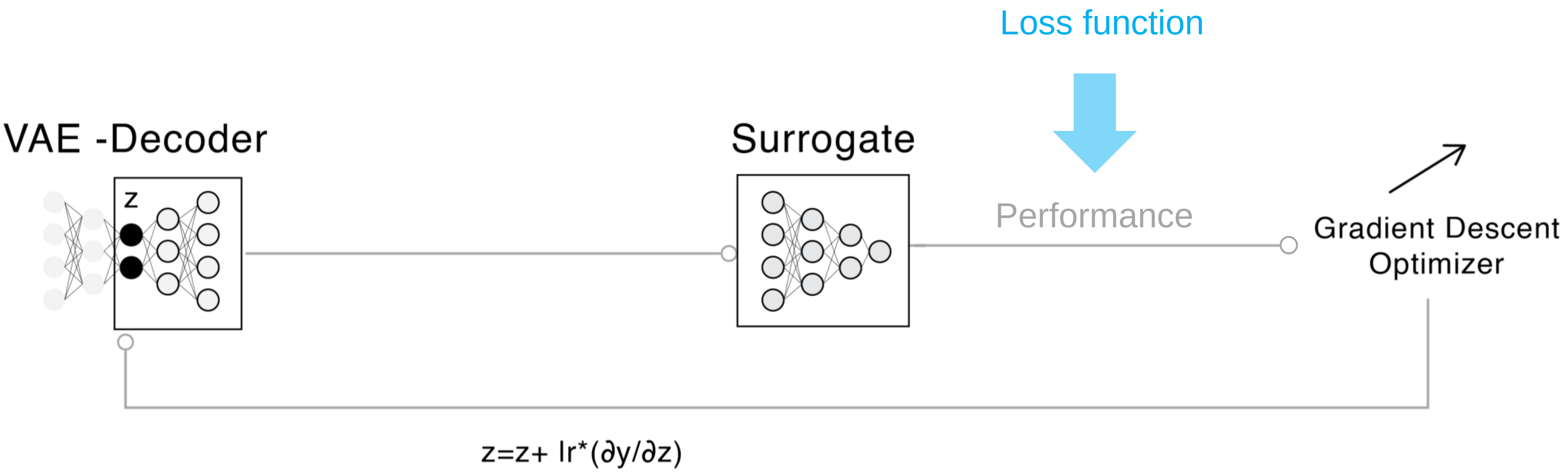
Gradient Descent Optimizer



Gradient Descent Optimizer

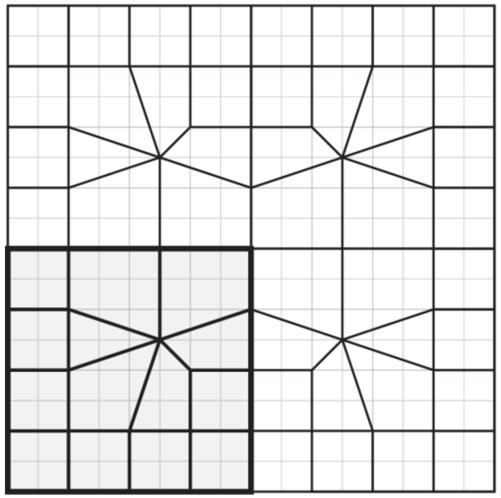


Gradient Descent Optimizer

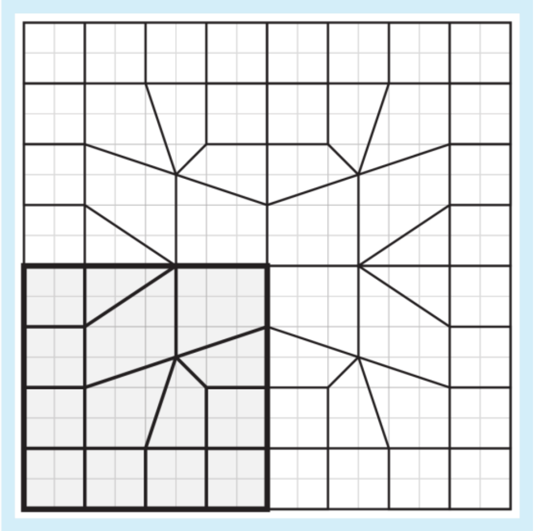


Results

Initial Mesh



Optimized Mesh



Learning rate: **0.5**

Number of iterations: **1000**

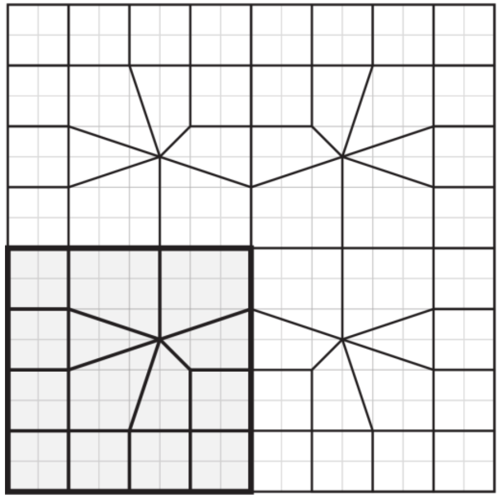
Performance Score of Starting Mesh: **0.17705911**

Estimated Performance Score of Optimized Shell: **0.1425786**

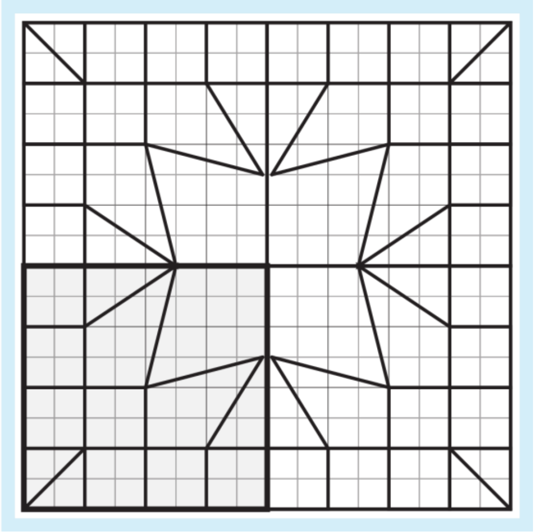
Real Performance Score of Optimized Shell: **0.085932**

Novel Design optimized by 206%

Results



Initial Mesh



Optimized Mesh

Learning rate: **2.5**

Number of iterations: **1000**

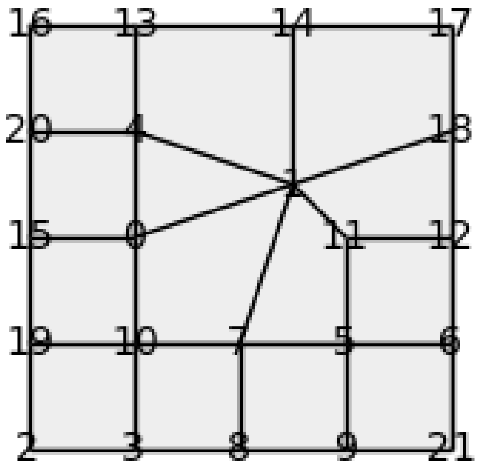
Performance Score of Starting Mesh: **0.17705911**

Estimated Performance Score of Optimized Shell: **0.16579011**

Real Performance Score of Optimized Shell: **0.044936816**

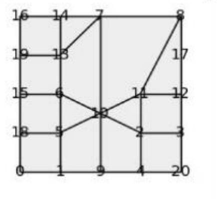
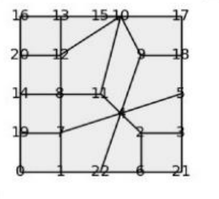
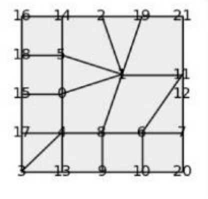
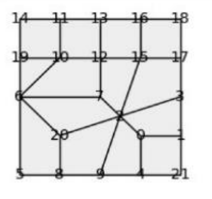
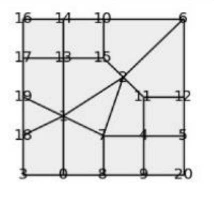
Novel Design optimized by 394%

Results

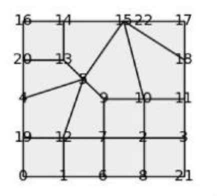
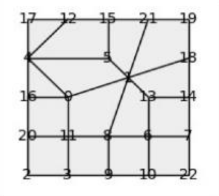
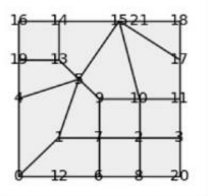
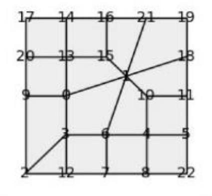
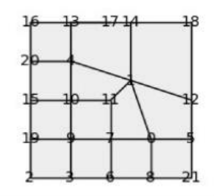


Starting Design

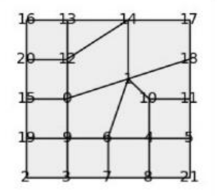
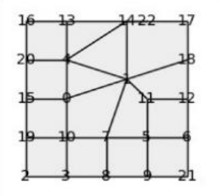
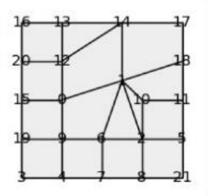
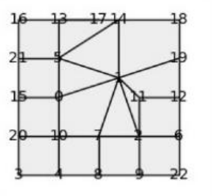
Learning Rate 5

					
Predicted Perfomance	0.18982741	0.17134124	0.14119774	0.10963255	0.18809715
Novelty	no	yes	yes	yes	no
Iterations	500	500	500	500	500

Learning Rate 2.5

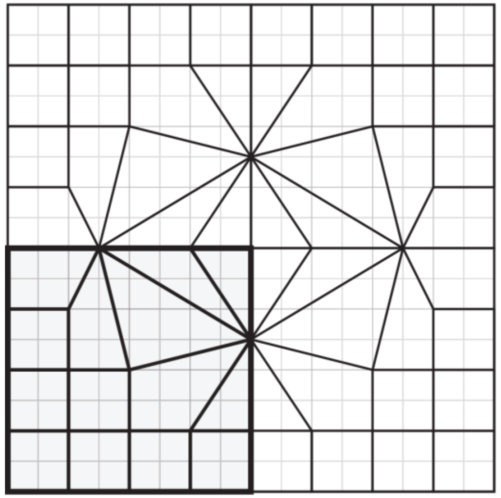
					
Predicted Perfomance	0.16329785	0.17972642	0.1726144	0.19679457	0.21983096
Novelty	no	no	yes	no	no
Iterations	500	500	500	500	500

Learning Rate 0.5

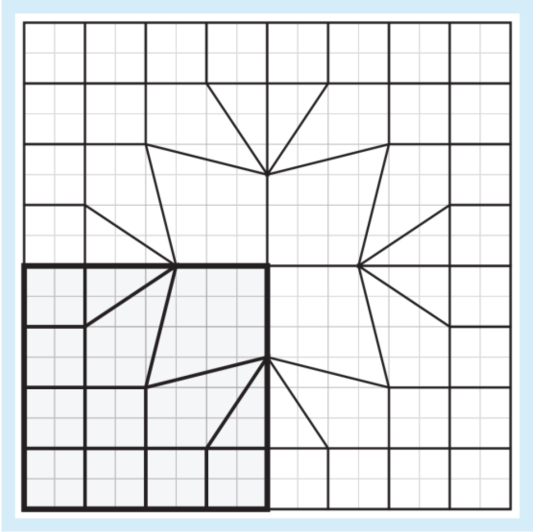
				
Predicted Perfomance	0.14270523	0.12407419	0.14612302	0.1620233
Novelty	yes	yes	yes	yes
Iterations	500	500	500	500

Results

Initial Mesh



Optimized Mesh



Optimized by 2591%
(Similar to the best performing mesh excluded from training)

Learning rate: **2.5**

Number of iterations: **1000**

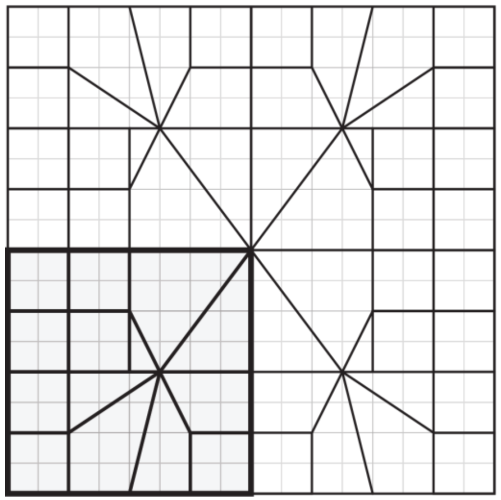
Performance Score of Starting Mesh: **0.4659505**

Estimated Performance Score of Optimized Shell: **0.15925622**

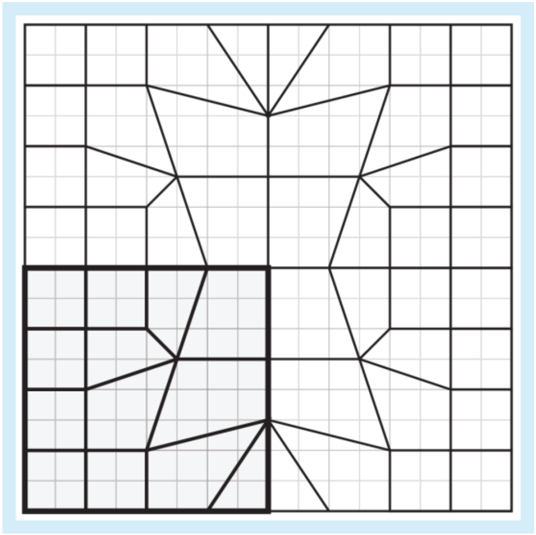
Real Performance Score of Optimized Shell: **0.01798574**

Results

Initial Mesh



Optimized Mesh



Learning rate: **2.5**

Number of iterations: **1000**

Performance Score of Starting Mesh: **0.18231553**

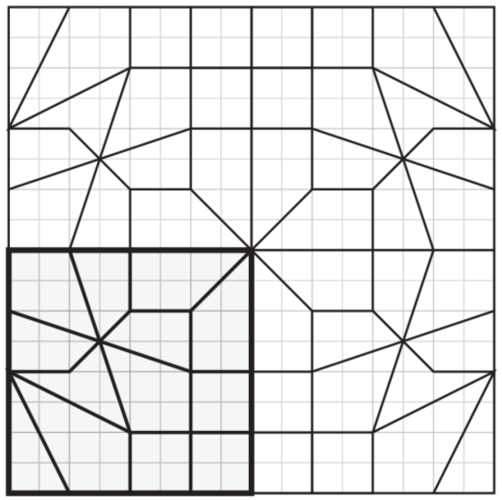
Estimated Performance Score of Optimized Shell: **0.12000429**

Real Performance Score of Optimized Shell: **0.113239**

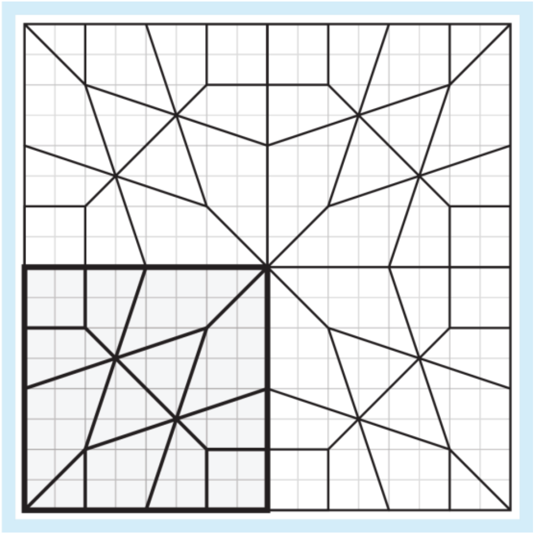
Novel Design optimized by 161%

Results

Initial Mesh



Optimized Mesh



Learning rate: **5**

Number of iterations: **1000**

Performance Score of Starting Mesh: **0.3534903**

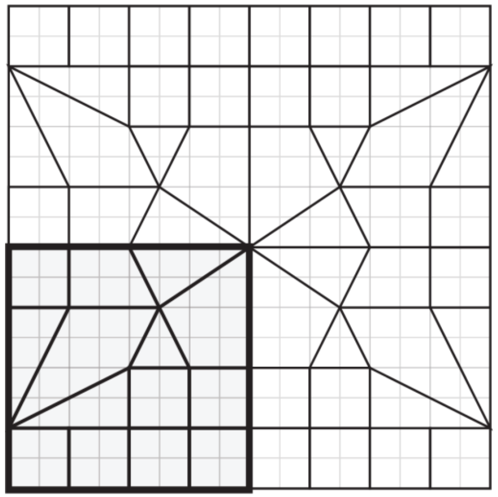
Estimated Performance Score of Optimized Shell: **0.1897085**

Real Performance Score of Optimized Shell: **0.115608**

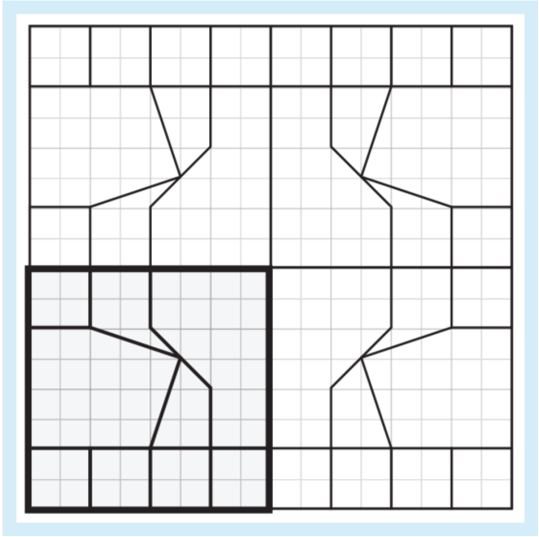
Novel Design optimized by 306%

Results

Initial Mesh



Optimized Mesh



Invalid Design

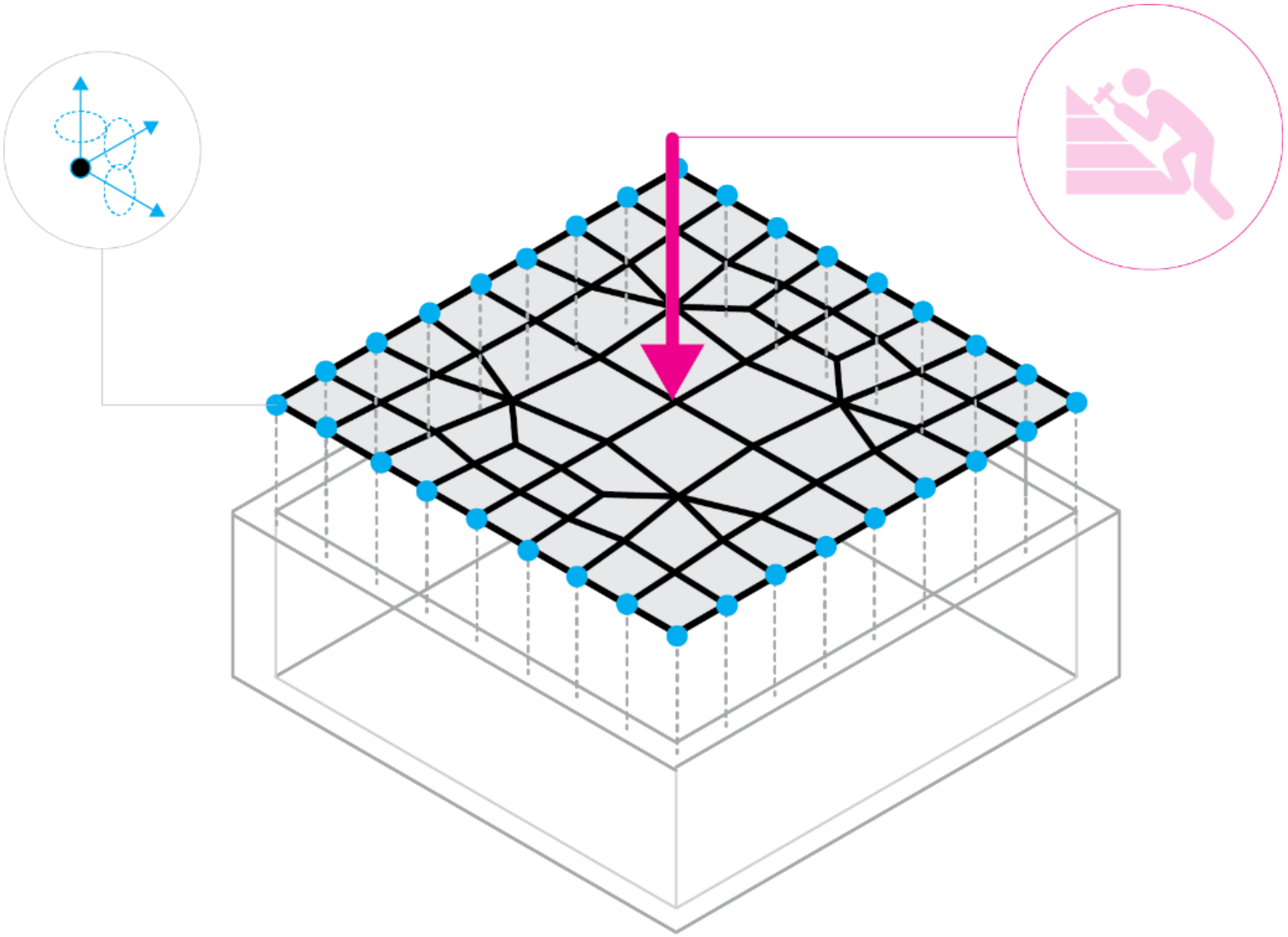
Learning rate: **0.5**

Number of iterations: **1000**

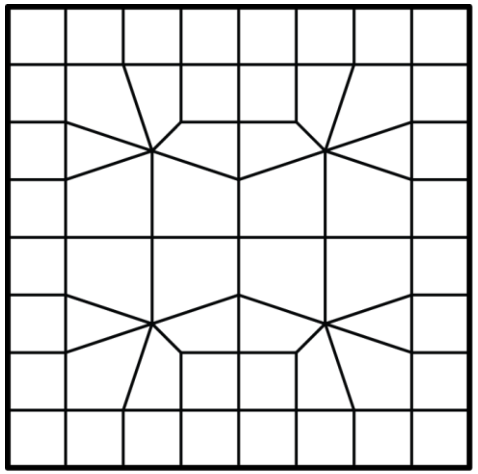
Performance Score of Starting Mesh: **0.415089337**

Estimated Performance Score of Optimized Shell: **0.1145393**

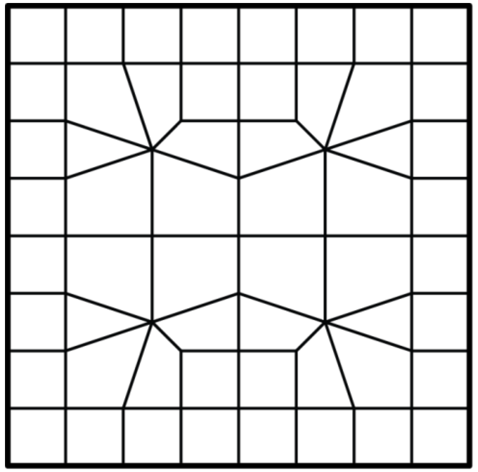
Case Study



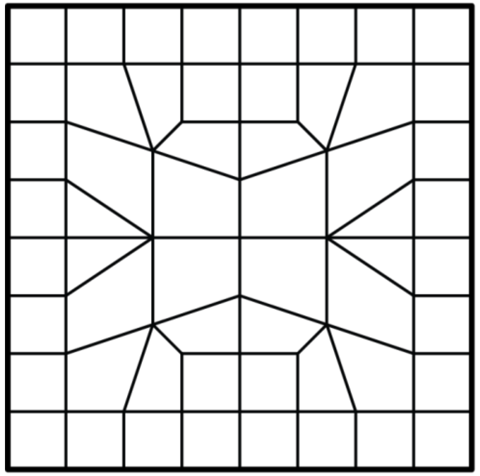
Starting Performance Score
0.177059



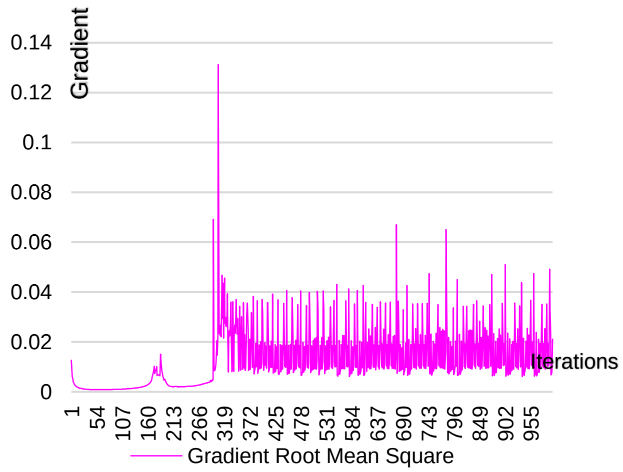
Starting Performance Score
0.177059



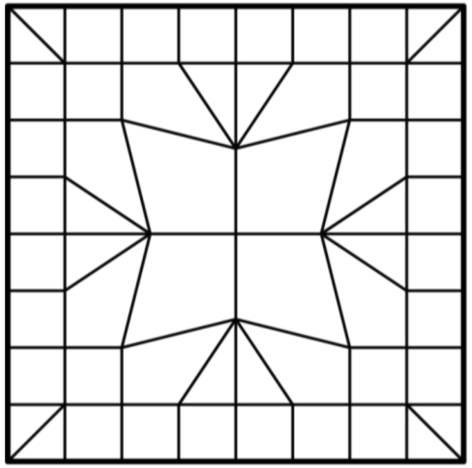
Performance score
0.085932
Optimized by **206%.**



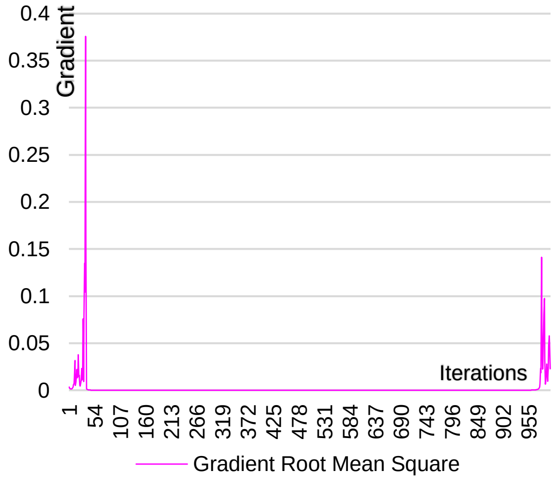
Learning rate: 0.5



Performance score
0.044936
Optimized by **394%.**



Learning rate: 2.5



The final design using the AI output result for a learning rate of 0.5

The final design using the AI output result for a learning rate of 2.5

Conclusions

Main Question

Can an AI based framework generate new and structurally effective solutions?

- The Gradient Descent Optimizer was able to converge to structurally better performing designs than those existed in provided dataset.
- An AI workflow can indeed expand the capabilities of Generative Design and reveal novel and structurally effective solutions.

Sub questions

- **Can a Variational Autoencoder be trained to generate mesh tessellations?**
  Yes, the VAE can generate novel solutions.

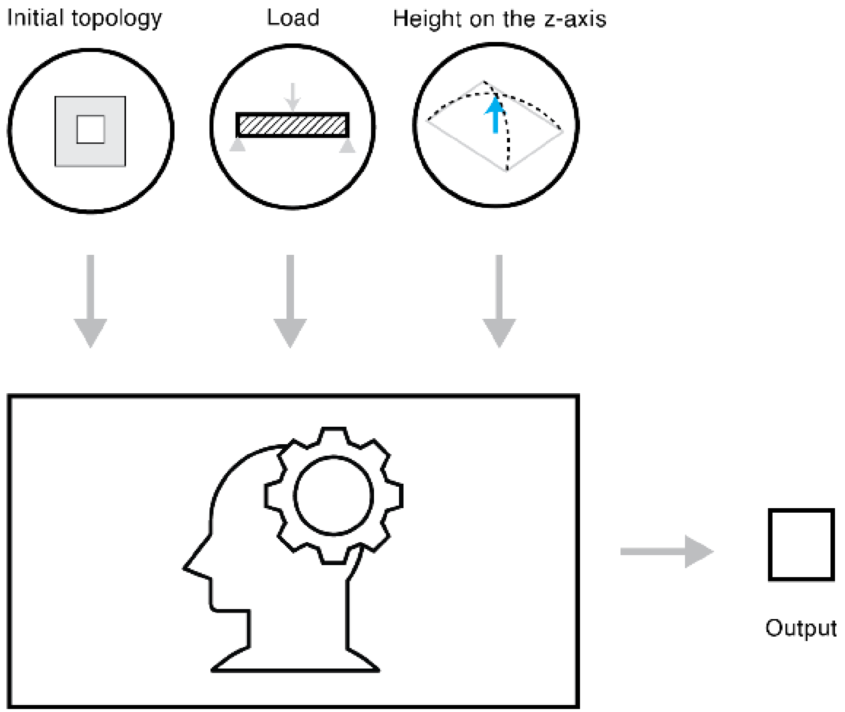
- **What form of data can be used to train a Variational Autoencoder to generate mesh tessellations?**
  - Adjacency matrices can be used successfully.
  - A flattened and simplified product, resulting from the adjacency matrices, can be used.

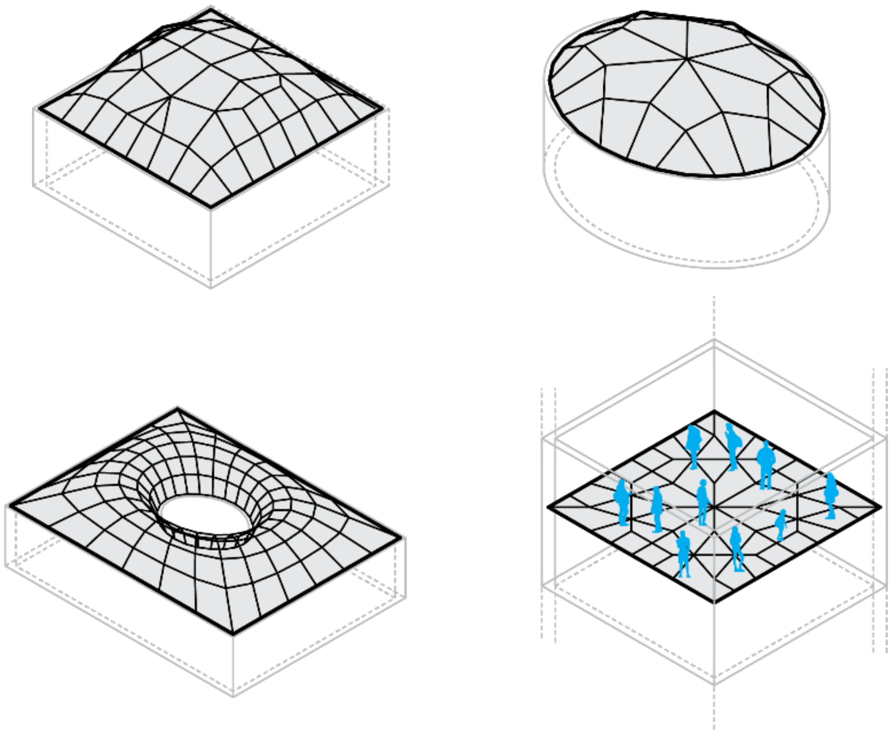
- **Can a surrogate model learn to predict the structural performance of decoded graph networks?**

  Yes, if the loss of the VAE is low it can

- **Can a Gradient Descent Optimizer propagate back to encoded data to search for optimum solutions?**
  Yes, The Gradient Descent was able to optimize mesh tessellations and discover novel solutions. However, in many cases invalid designs were produced. This is due to two main problems:
  - The VAE often generates invalid samples.
  - The surrogate model has not yet been trained to predict the performance of invalid tessellations

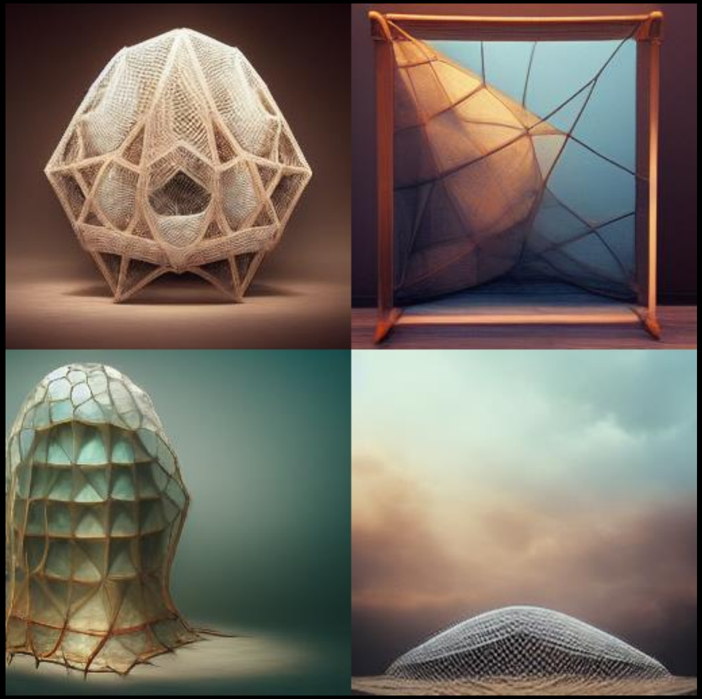
Future Development





Future Development

- Other generative models like GANs and Graph Variational Autoencoders could produce better results.
- Dataset augmentation with penalized samples for training the surrogate model to score negatively invalid meshes.
- Dataset augmentation with further pattern exploration, extrusion height, boundaries, etc.
- Training the workflow based on some other criteria qualitive and quantitative criteria (different load cases, similarity, number of singularities, maximum length of edges,etc).



Mesh shell structure
(Midjourney , https://www.midjourney.com/app/)