Learning the link between architectural form and structural efficiency: a supervised machine learning approach

S. Pooyan Kazemi S. [1*], Aldo Ghisi [2] and Stefano Mariani [3,]

[1] PhD student at Politecnico di Milano, Department of Civil and Environmental Engineering, Piazza Leonardo da Vinci, 32, 20133, Milano, Italy; seyedpooyan.kazemi@polimi.it
[2] Associate professor in Politecnico di Milano 2; aldo.ghisi@polimi.it.
[3] Associate professor in Politecnico di Milano 2; stefano.mariani@polimi.it

POLITECNICO
MILANO 1863

DIPARTIMENTO DI
INGEGNERIA CIVILE E AMBIENTALE

Abstract

In this work, we exploit supervised machine learning (ML) to investigate the relationship between architectural form and structural efficiency under seismic excitations. We inspect a small dataset of simulated responses of tall buildings, differing in terms of base and top plans within which a vertical transformation method is adopted (tapered forms). A diagrid structure with members having a tubular cross-section is mapped on the architectural forms, and static loads equivalent to the seismic excitation are applied. Different ML algorithms, such as kNN, SVM, decision tree, ensemble, discriminant, Naïve Bayes are next trained, to classify the seismic response of each form on the basis of a specific label. Results to be presented rely upon the drift of the building at its top floor, though the same procedure can be generalized and adopt any performance characteristic of the considered structure, like e.g., the drift ratio, total mass, or expected design weight. The classification algorithms are all tested within a Bayesian optimization approach; it is then found that the decision tree classifier provides the highest accuracy, linked to the lowest computing time. This research activity puts forward a promising perspective for the use of ML algorithms to help architectural and structural designers during the early stages of conception and control of tall buildings.

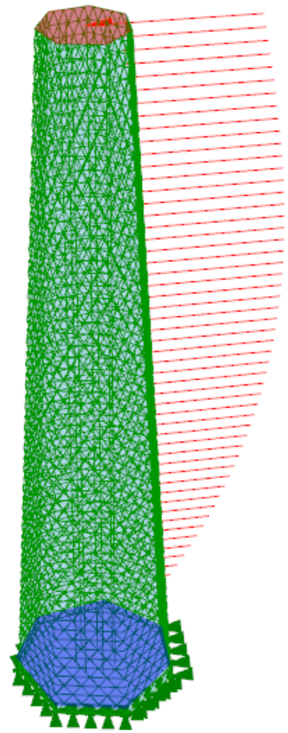
Keywords: Supervised Machine Learning; Classification;
Tall Building; Architectural Form; Structural Efficiency

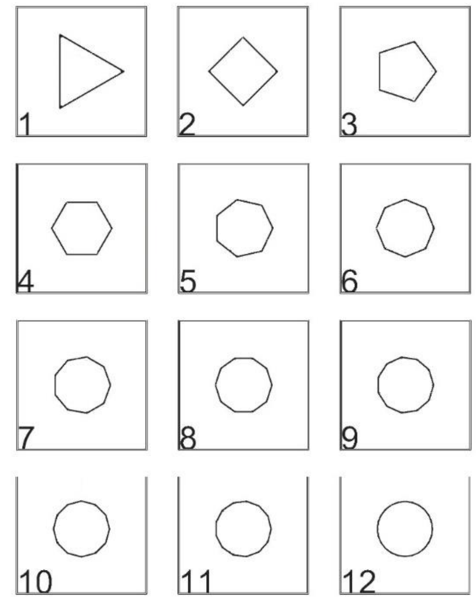
IOCA 2021

Introduction

Problem definition

- Investigate the effect of architectural form (top and bottom plan) on structural efficiency of tall buildings by means of ML





The importance

- Generating new data needs a huge amount of effort for structural and architectural modelling, ML could make it simpler!
- Early-stage design phase plays a critical role in tall buildings; it will be aided by powerful ML tools

Source: https://en.wikipedia.org

Introduction

Objectives

- Comparison between efficiency of some classification algorithms
- Investigate the structural efficiency and architectural form relation by ML tools

Function

- Propose an advanced workflow for architects and engineers in tall building design

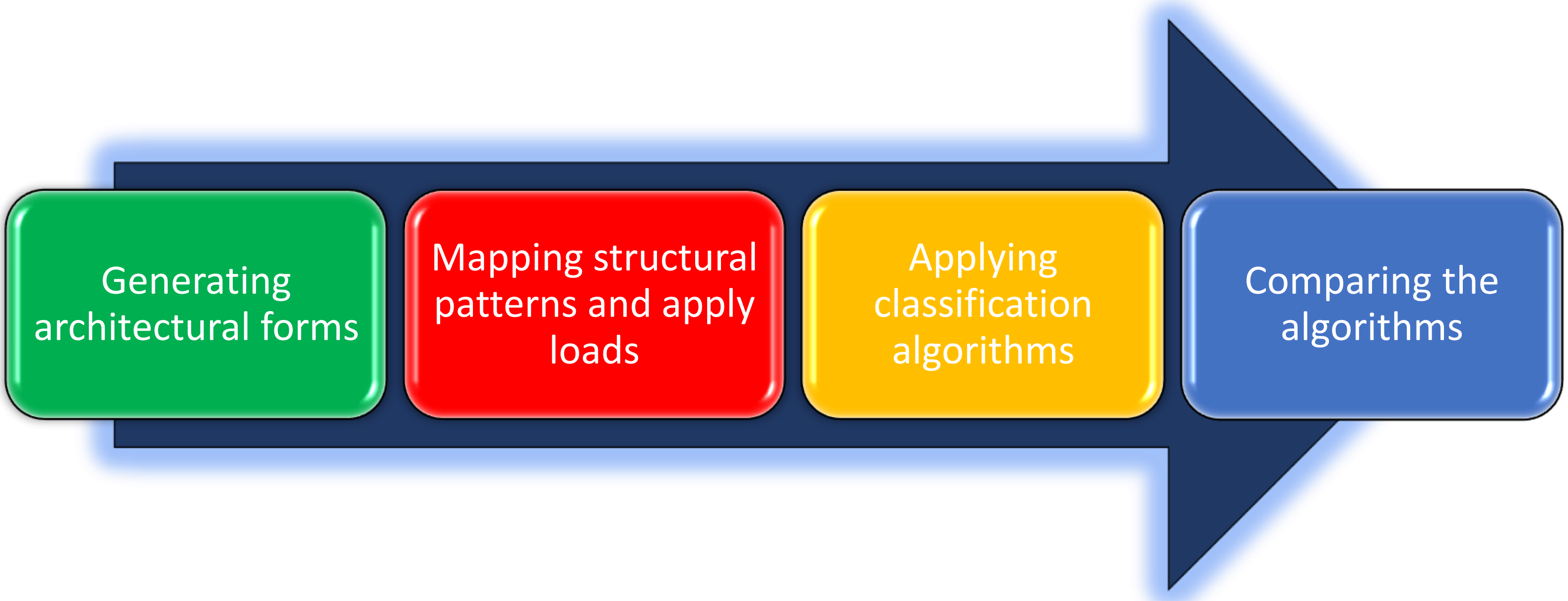
Tools

- Parametric modelling for architectural and structural analysis: Rhino, Grasshopper, Karamba
- Matlab for ML (classification learner app)

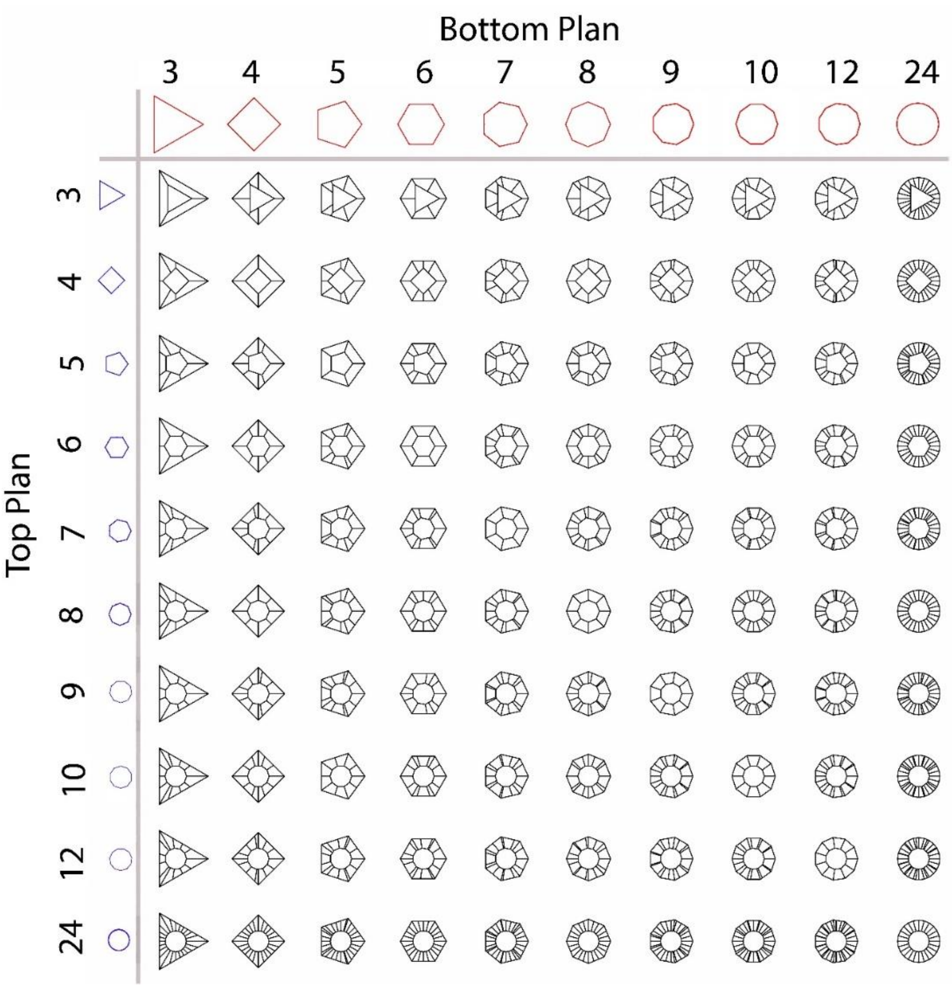
Source: https://en.wikipedia.org

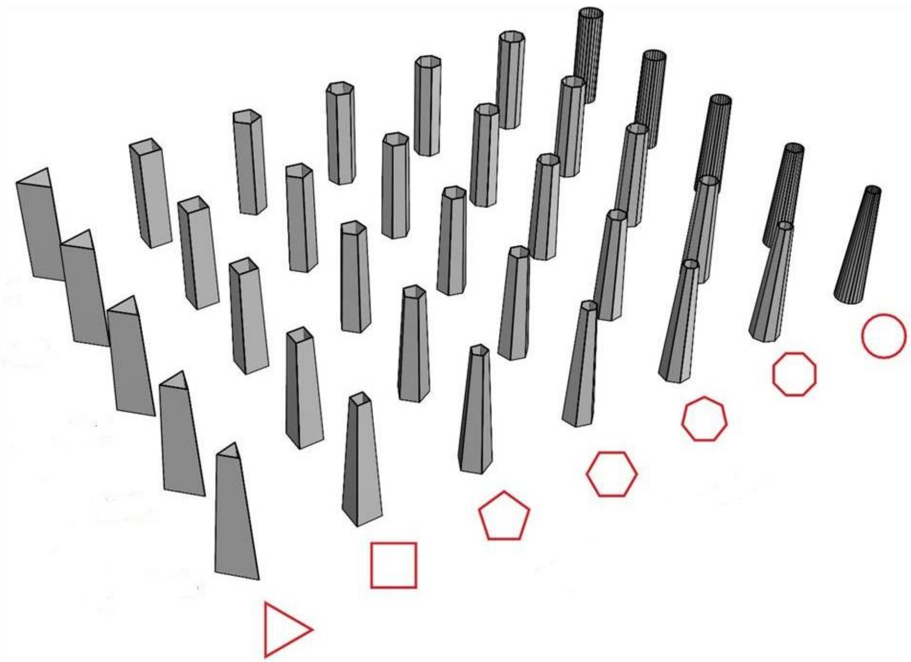
IOCA 2021

Methodology



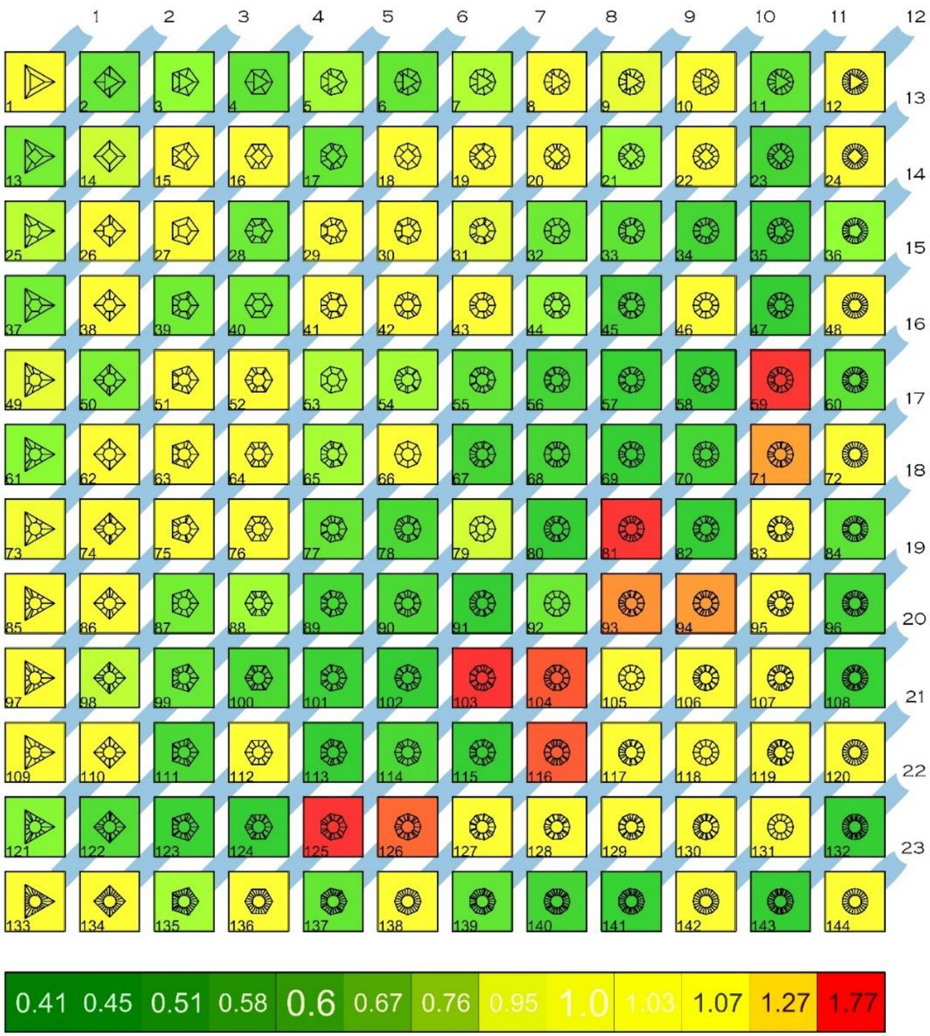
Architectural forms 12*12 (144 forms)





IOCA 2021

Drift diagrams for 144 models



Classification parameters

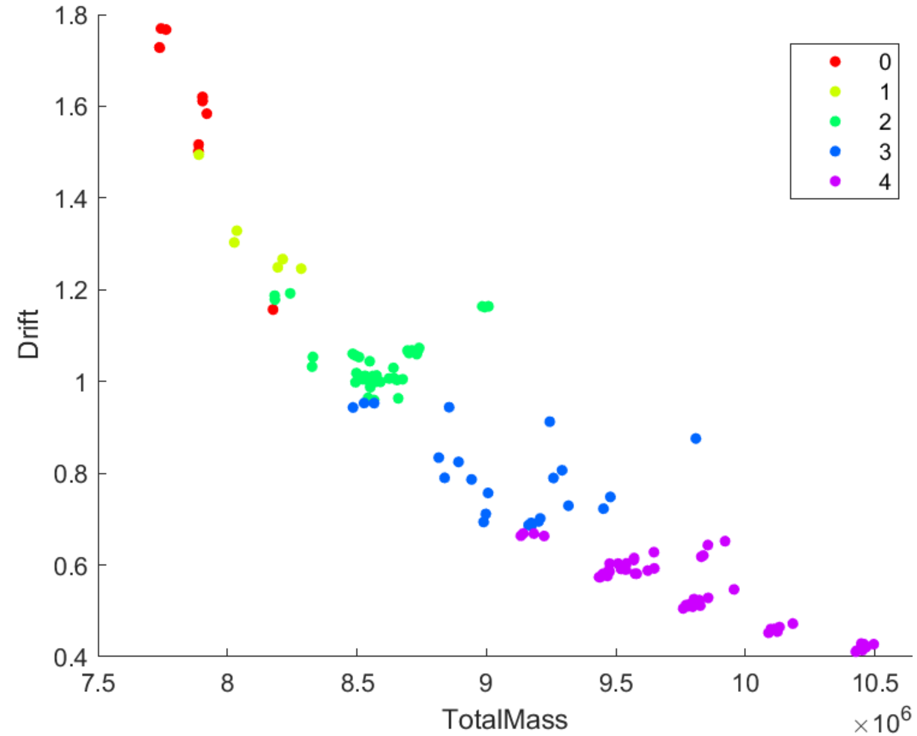
Number of observations	Number of predictors	Label
144	8	Drift response 0-5

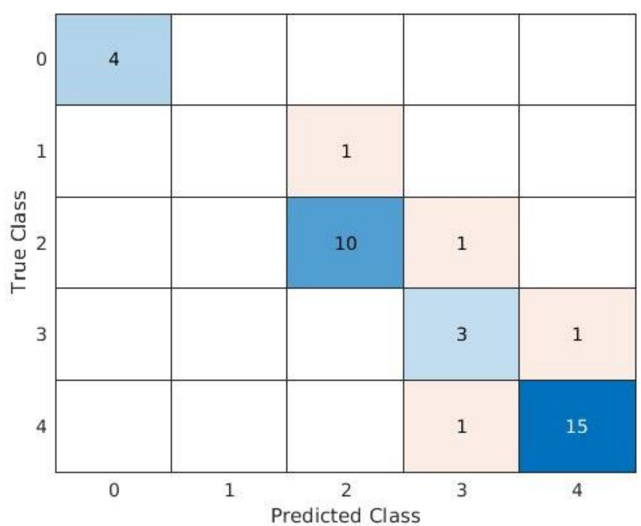
Very Good	**4**
Good	**3**
Mediocre	**2**
Bad	**1**
Very Bad	**0**

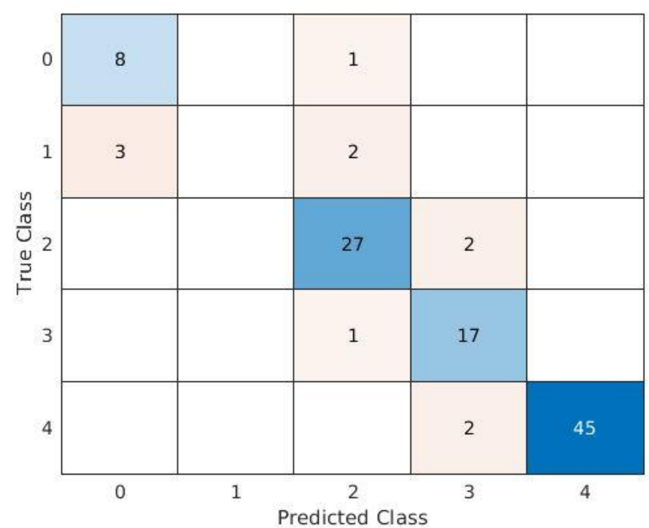
IOCA 2021

Training & Testing

- Total data: 144 models
- Training: 108   Testing: 36 (75% to 25%)
- Randomization
- Response: labels according to drift (0-4)
- Predictors: 7
- First Testing on kNN algorithm
- Number of neighbors=5
- Accuracy rate training: 91.67
- Accuracy rate testing: 83.33







IOCA 2021

Classification Algorithms

KNN

SVM

Decision Tree

Ensemble

Discriminant

Naïve Bayes

Model specification
• Label: drift (5 classes) • 5 fold cross validation • 108 observations (75% of 144 models) • 8 predictors

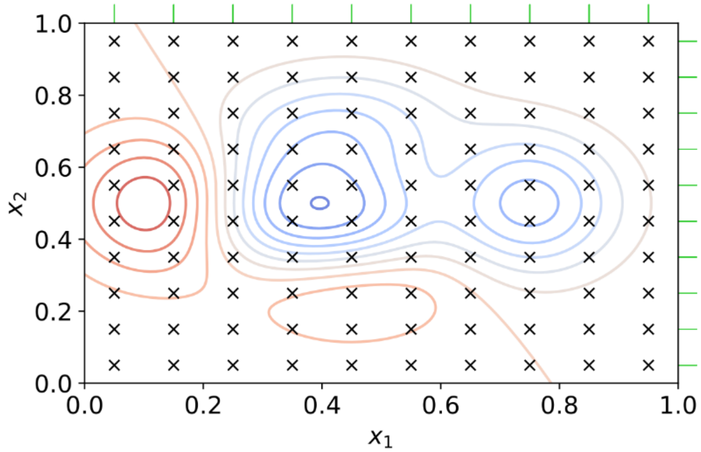
IOCA 2021

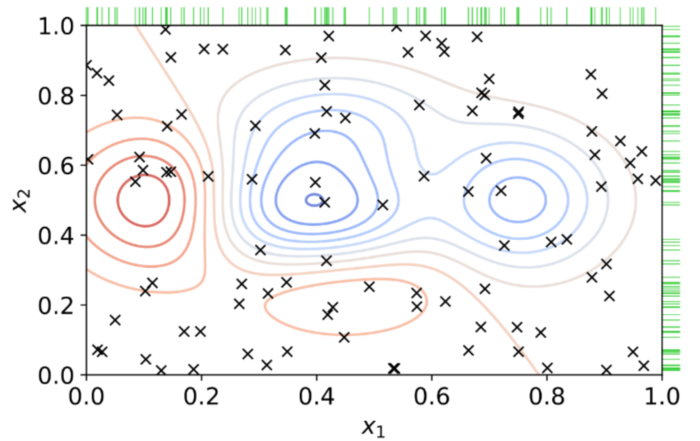
Hyperparameter optimization

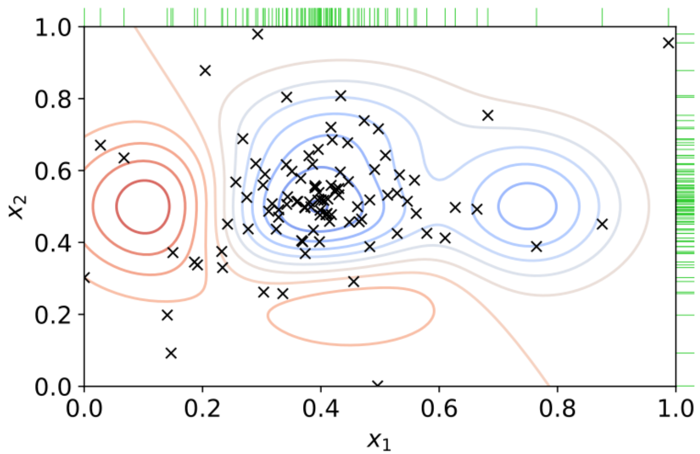
Grid search

Random search

Bayesian optimization







IOCA 2021

Optimizable k-Nearest Neighbor

Hyperparameter search range

Number of Neighbors	Distance Metric	Distance Weight	Standardize Data
1-54	City block Chebyshev Correlation Cosine Euclidean Hamming Jaccard Mahalanobis MinkowsKi (cubic) Spearman	Equal Inverse Squared inverse	True False

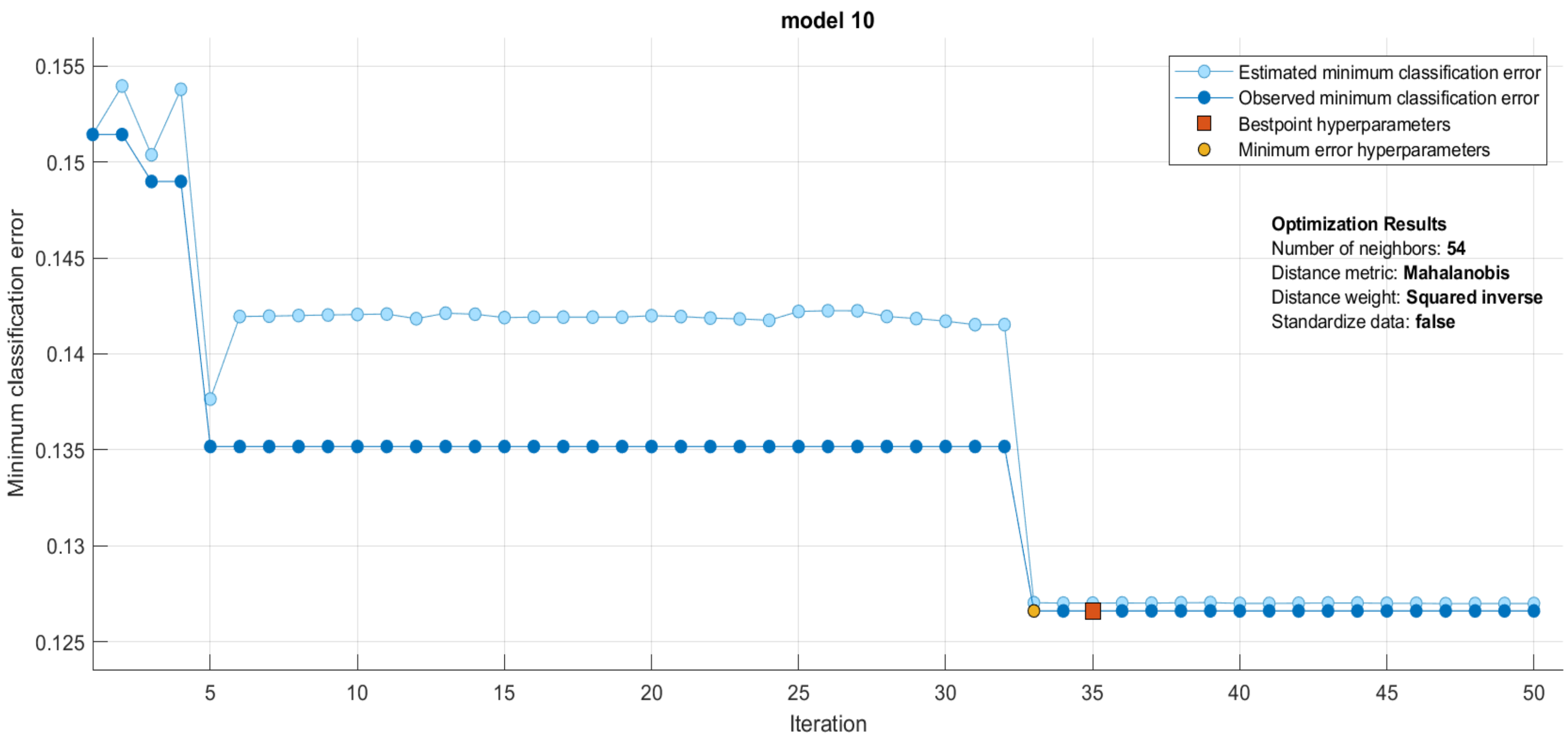
IOCA 2021

Optimizable k-Nearest Neighbor

Model No.	Accuracy Training	Accuracy Testing	Total Misclass. cost	Training time	Number of Neighbors	Distance Metric	Distance Weight	Standardize Data	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
1	91.7	94.4	9	38.1	2	City block	Squared inverse	True	Bayesian	Expected Improvement per second plus	30	all	Disabled
3	91.7	94.4	9	15.9	2	City block	inverse	True	Random search		30	all	Disabled
8	91.7	97.2	9	64.5	9	City block	Squared inverse	True	Bayesian	Expected Improvement per second plus	50	all	Disabled
9	91.7	94.4	9	121.13	1	City block	inverse	True	Grid search		Grid Div.=10	all	Disabled
10	88	97.2	13	55.9	54	Mahalanobis	Squared inverse	false	Bayesian	Expected Improvement per second plus	50	all	95% Variance 1 kept off
16	88	97.2	13	54.3	15	Euclidean	Squared inverse	false	Bayesian	Expected Improvement per second plus	50	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	95% Variance 1 kept off
17	96.3	97.2	4	46.3	3	City block	Squared inverse	True	Bayesian	Expected Improvement per second plus	50	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	Disabled

IOCA 2021

Bayesian optimization sample



IOCA 2021

Optimizable Support Vector Machine

Hyperparameter search range

Kernel Function	Kernel Scale	Box Constraint Level	Multi Class Method	Standardize Data
Gaussian Linear Quadratic Cubic	0.001-1000	0.001-1000	One-vs-All One-vs-One	True false

IOCA 2021

Optimizable Support Vector Machine

Model No.	Accuracy Training	Accuracy Testing	Total Misclass. cost	Training time	Kernel Function	Kernel Scale	Box Constraint Level	Multi Class Method	Standardize Data	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
1	95.4	94.4	5	134.6	Gaussian	15.96	984.48	One-vs-One	True	Bayesian	Expected Improvement per second plus	40	all	Disabled
2	95.4	94.4	5	2030	Linear	10		One-vs-One	True	Grid search		Grid Div.=10	all	Disabled
3	95.4	97.2	5	223.5	Linear	2.14		One-vs-One	True	Random Search		40	all	Disabled
4	95.4	94.4	5	139.8	Linear	301.08		One-vs-One	True	Bayesian	Expected Improvement per second plus	40	all	Disabled
5	95.4	97.2	5	139.8	Quadratic	133.56		One-vs-One	True	Bayesian	Expected Improvement per second plus	40	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	Disabled
6	86.1	97.2	15	137.8	Gaussian	.12	966.23	One-vs-All	True	Bayesian	Expected Improvement per second plus	40	all	95% Variance 1 kept off
7	91.7	97.2	9	152.9	Linear	99.79		One-vs-One	True	Bayesian	Expected Improvement per second plus	40	all	5 kept off

IOCA 2021

Optimizable Decision Tree

Hyperparameter Search Range

Maximum Number of Splits	Split Criterion
1-107	Gini's diversity index Twoing rule Maximum deviance reduction

IOCA 2021

Optimizable Decision Tree

Model No.	Accuracy Training	Accuracy Testing	Total Misclassification cost	Training time	Maximum Number of Split	Split Criterion	Surrogate Decision Split	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
2	93.5	100.0	7	29.7	5	Twoing rule		Bayesian	Expected Improvement per second plus	40	all	Disabled
5	93.5	100.0	7	17.4	4	Gini's diversity index	On, using max 10 surrogate	Random Search		40	all	Disabled
6	93.5	100.0	7	18.8	7	Gini's diversity index	Find all	Random Search		40	all	Disabled
9	86.1	77.8	15	30.76	42	Gini's diversity index	Find all	Random Search		40	all	95% Variance 1 kept off
11	93.5	100.0	7	45.0	5	Gini's diversity index	off	Random Search		40	1. Model No	Disabled

IOCA 2021

Optimizable Ensemble

Hyperparameter Search Range

Ensemble method	Maximum number of splits	Number of learners	Learning rate	Number of predictors to sample
Bag AdaBoost RUSBoost	1-107	10-500	0.001-1	1-8

IOCA 2021

Optimizable Ensemble

Model No.	Accuracy	Total Misclassification cost	Training time	Ensemble method	Maximum number of splits	Number of learners	Learning rate	Number of predictors to sample	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
1	98.1	2	123.3	RUSBoost	38	71	.94	Select All	Bayesian	Expected Improvement per second plus	40	all	Disabled
2	97.2	3	128.3	RUSBoost	23	324	.46	Select All	Grid Search		Grid Div.= 10	all	Disabled
3	98.1	2	72.0	AdaBoost	3	20	.10	Select All	Random Search	Expected Improvement per second plus	40	all	Disabled
4	96.3	4	80.8	RUSBoost	14	12	.98	Select All	Bayesian	Expected Improvement per second plus	40	1. Model No	Disabled
5	98.1	2	105.9	RUSBoost	25	65	.85	Select All	Bayesian	Expected Improvement per second plus	40	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	Disabled
6	85.2	16	129.4	RUSBoost	106	124	0.07	Select All	Bayesian	Expected Improvement per second plus	40	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	95% Variance 1 kept off
7	88	13	99.2	Bag	66	61	1		Bayesian	Expected Improvement per second plus	40	all	95% Variance 1 kept off

IOCA 2021

Optimizable Naïve Bayes

Hyperparameter search range

Distribution names	Kernel type:
Gaussian Kernel	Gaussian Box Epanechnikov Triangle

IOCA 2021

Optimizable Naïve Bayes

Model No.	Accuracy	Total Misclassification cost	Training time	Distribution names	Kernel type:	Support	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
1	90.7	10	85.5	Kernel	Gaussian	unbounded	Bayesian	Expected Improvement per second plus	30	all	Disabled
2	92.6	8	109.7	Kernel	Box	Positive	Bayesian	Expected Improvement per second plus	30	all	Disabled
4	92.6	8	13.7	Kernel	Box	Positive	Grid Search		Grid Div.= 10	all	Disabled
5	82.4	19	36.6	Gaussian	Triangle	Positive	Bayesian	Expected Improvement per second plus	30	all	95% Variance 1 kept off
7	91.7	9	66.3	Kernel	Gaussian	unbounded	Bayesian	Expected Improvement per second plus	30	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	Disabled

IOCA 2021

Optimizable Discriminant

Hyperparameter search range

Discriminant type:
Linear Quadratic Diagonal Linear Diagonal Quadratic

IOCA 2021

Optimizable Discriminant

Model No.	Accuracy	Total Misclassification cost	Training time	Discriminant type:	Optimizer	Acquisition Function	Iteration	Feature selection	PCA
1	93.5	7	47.8	Linear	Bayesian	Expected Improvement per second plus	40	all	Disabled
2	85.2	16	46.8	Diagonal Quadratic	Bayesian	Expected Improvement per second plus	40	all	95% Variance 1 kept off
3	94.4	6	45.3	Linear	Bayesian	Expected Improvement per second plus	40	1. Model No	Disabled
4	94.4	5	47.9	Linear	Bayesian	Expected Improvement per second plus	40	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	Disabled
5	85.2	16	48.7	Diagonal Quadratic	Bayesian	Expected Improvement per second plus	40	1. Model No 2. Total length of Diagrid Members 3. Max Normal Force	95% Variance 1 kept off

IOCA 2021

Conclusion

Model	Train Acc.	Test Acc.	Model	Train Acc.	Test Acc.	Model	Train Acc.	Test Acc.
KNN			SVM			Tree		
model 1	91.7	94.4	1	95.4	94.4	2	93.5	100
3	91.7	94.4	2	95.4	94.4	5	93.5	100
8	91.7	97.2	3	95.4	94.4	6	93.5	100
9	91.7	94.4	4	95.4	94.4	9	86.1	78
10	88.0	97.2	5	95.4	91.7	11	93.5	100
16	88.0	97.2	6	86.1	91.7			
17	96.3	97.2	7	91.7	91.7			

Model	Train Acc.	Test Acc.	Model	Train Acc.	Test Acc.	Model	Train Acc.	Test Acc.
Naïve Bayes			Discriminant			Ensemble		
1	90.7	91.7	1	93.5	86.1	1	98.1	100.0
2	92.6	88.9	2	85.2	83.3	2	97.2	97.2
4	92.6	88.9	3	94.4	91.7	3	98.1	100.0
5	82.4	83.3	4	95.4	91.7	4	96.3	97.2
7	91.7	88.9	5	85.2	83.3	5	98.1	100.0
						6	85.2	94.4
						7	88.0	94.4

It was proved that supervised machine learning can be successfully applied to this case study. Moreover, between six classification algorithms, each of them provided some advantages and disadvantages. Namely, the **ensemble** and the **decision tree** classifier algorithm could achieve the best results.

IOCA 2021

Thank you