Predictions and Unsupervised Analysis on Stocks Price Variations

Topics in Statistical Learning

Michele Bersani, Carlo Bosio, Federico Lusiani

Stocks Market Dataset

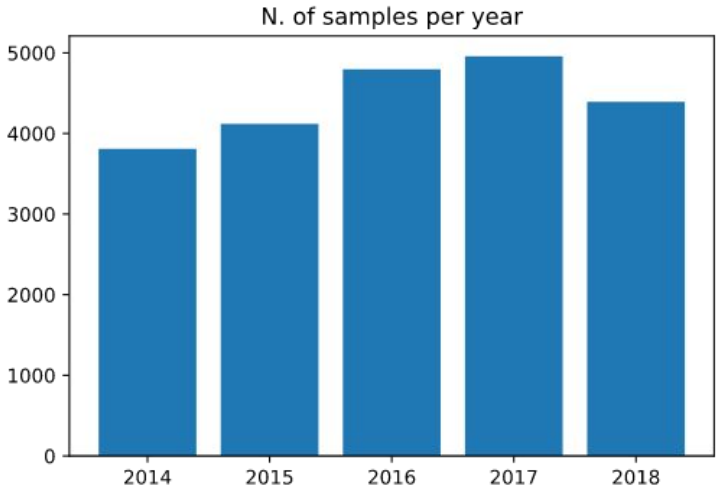
Source: 10-K filings of publicly traded companies

Subjects: ~4.000 financial profiles of publicly traded companies over the years 2014-2018, for a total of ~21.000 financial profiles

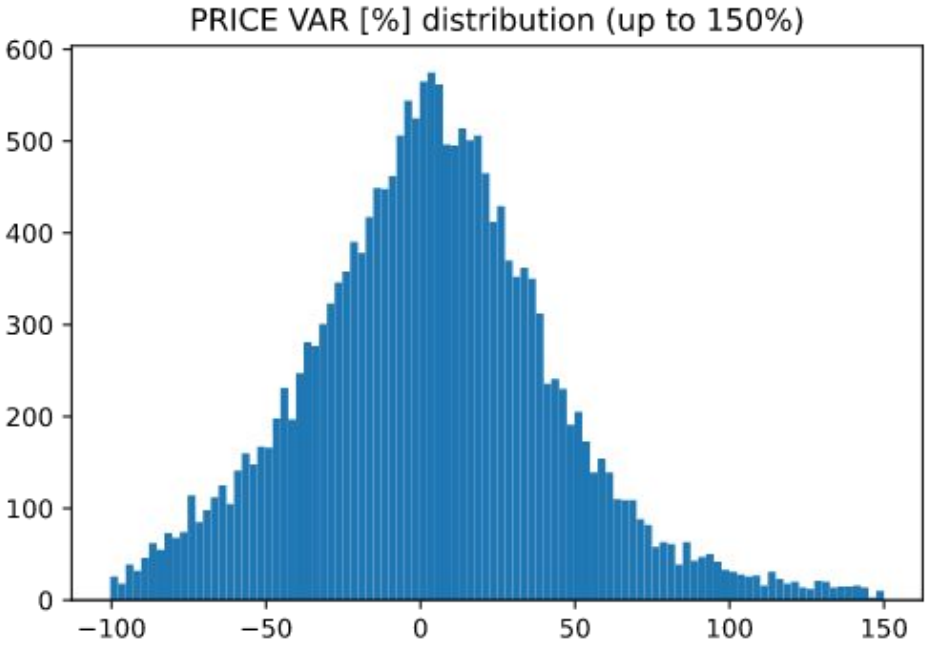
Measures: 200+ financial indicators for each company (in a given year)

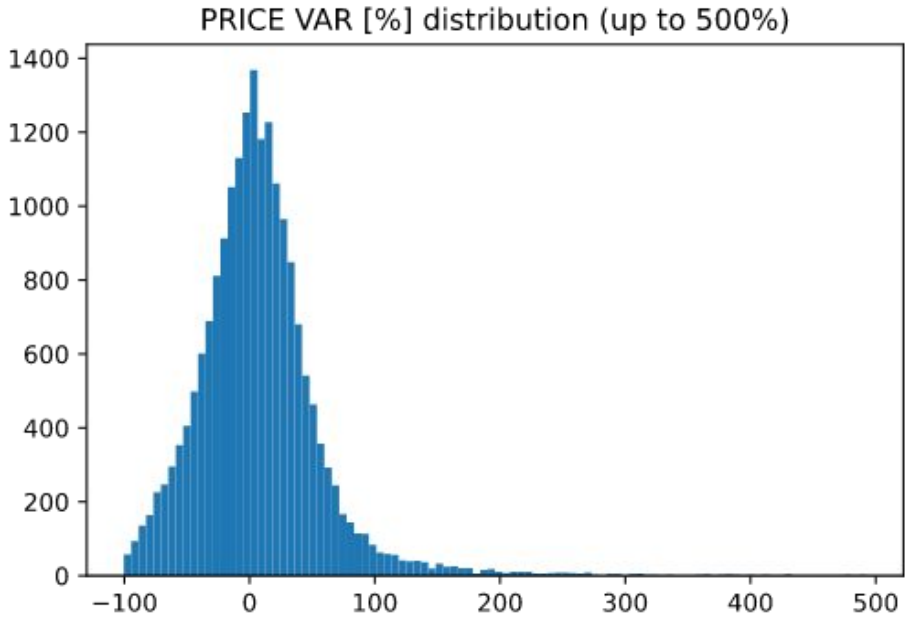
Goal A: Predict whether the price variation [%] of the company stocks will be positive in the next year

Goal B: Predict the price variation [%] of the company stocks in the next year



Output distribution



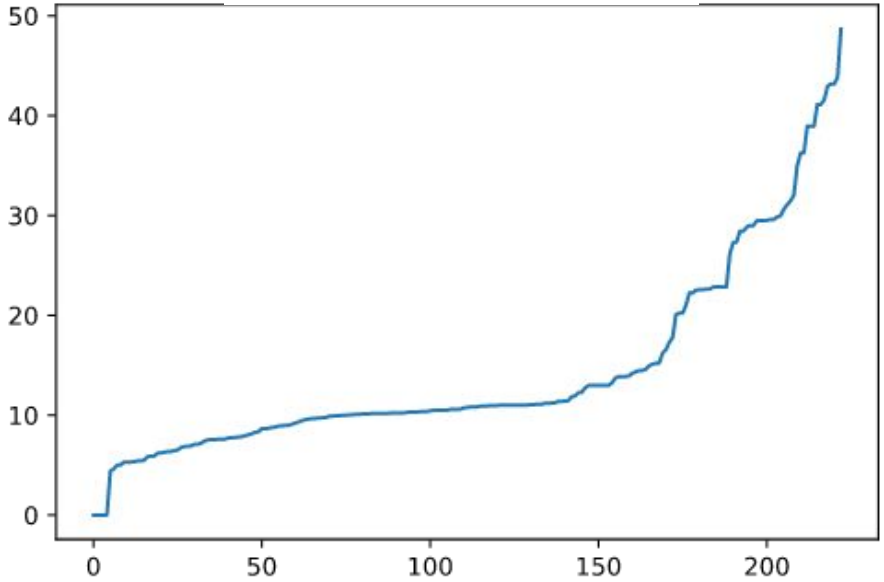


Preprocessing pipeline

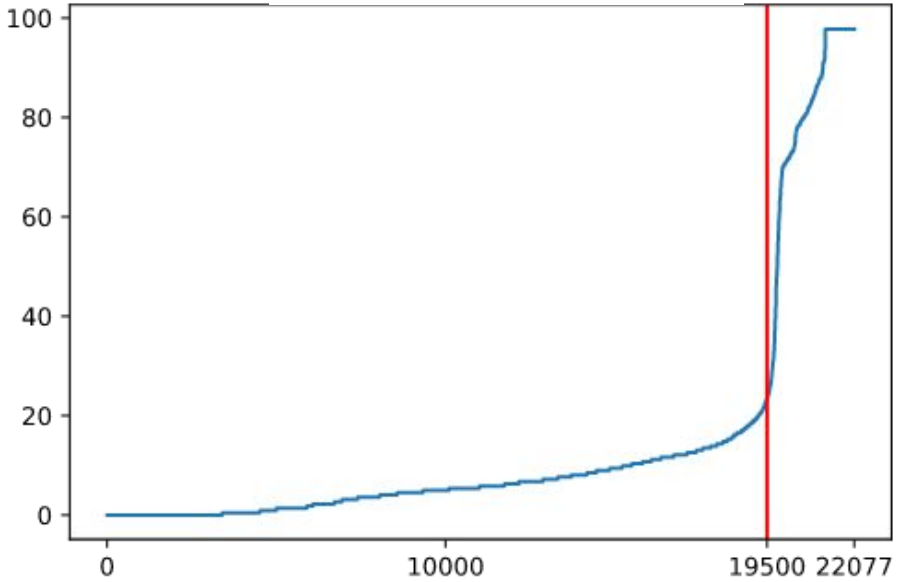
- Rows dropping if PRICE VAR > 150% or when exceeding NULLs
- Columns Normalization
- Logarithmic Transformation of Skewed Columns
- NULLs imputation with custom MICE
- Columns Normalization

NULLs distribution

% of NULLs per column



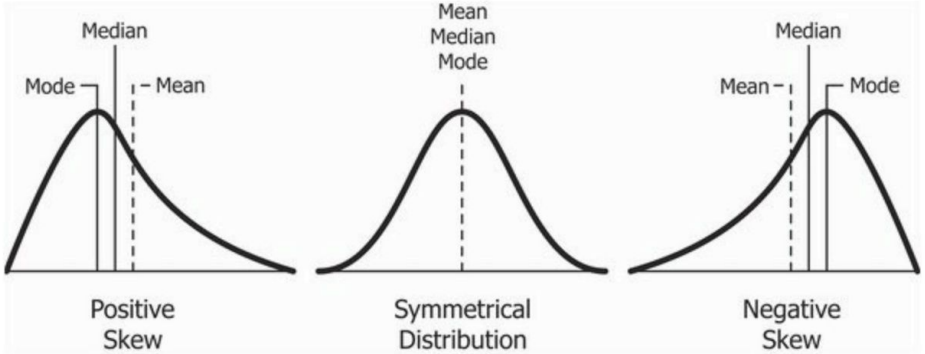
% of NULLs per row



Log Transformation of skewed columns

- Columns skewness evaluation:

$$s(X) = \frac{\mu(X) - \nu(X)}{\sigma(X)}$$



- Skewed distributions if $s(X) < -0.05 \quad \text{or} \quad s(X) > 0.05$

- If negative skewness: $X \leftarrow -X$

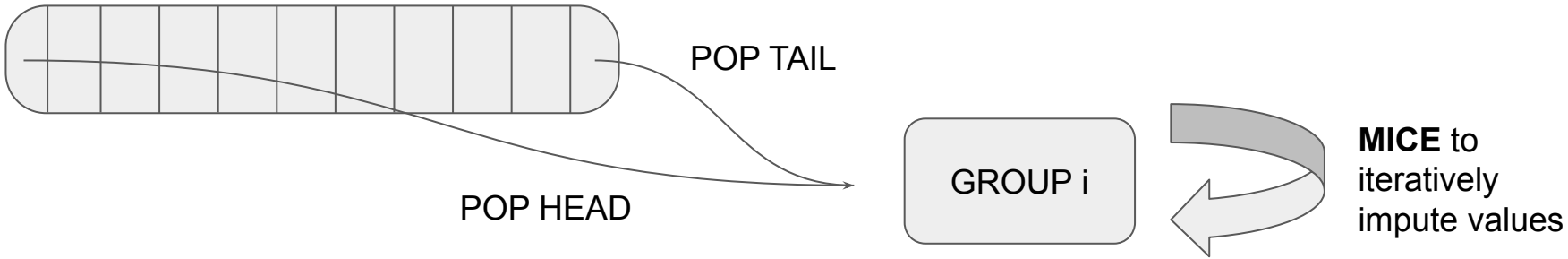
- $X \leftarrow \log(X - \min(X) + 0.01)$

Nulls imputation with custom MICE

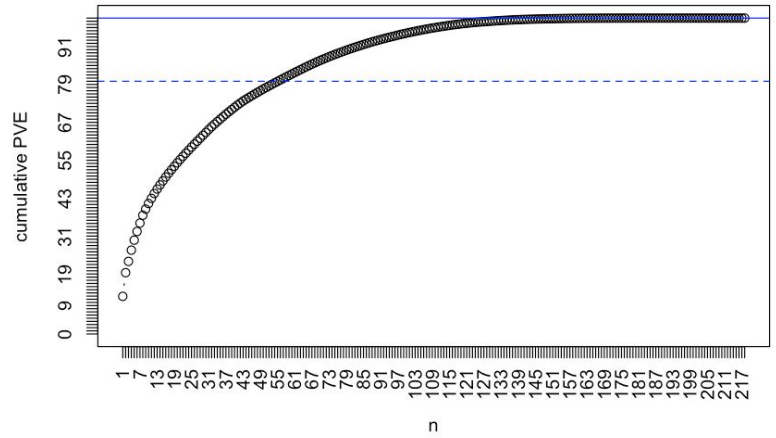
Obs: the nulls are sparsely distributed all over columns and rows

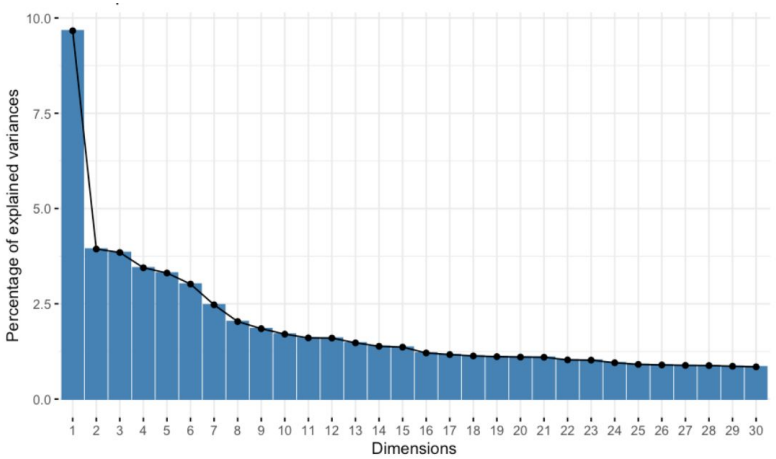
Grouping columns:

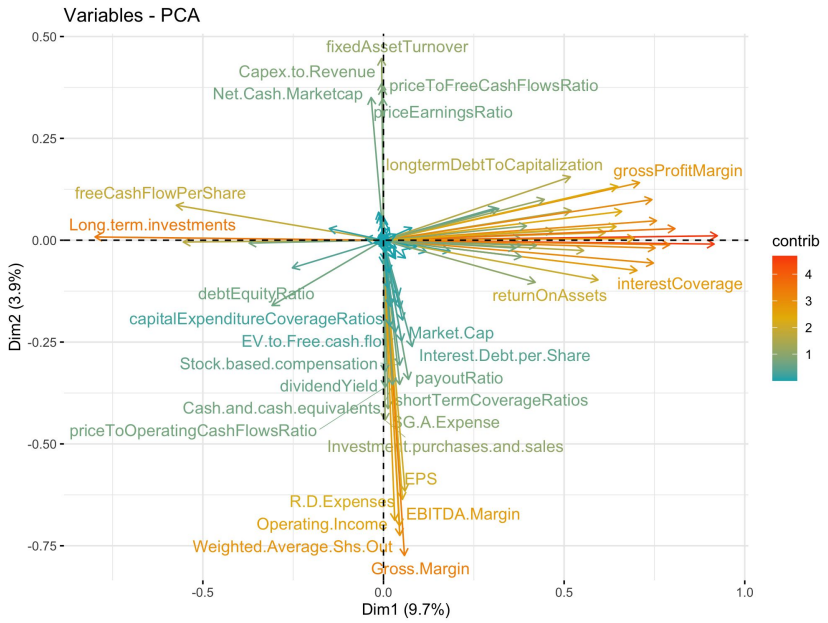
- Store columns in a deque ordered by number of nulls
- Create groups of 10 columns, where each group pops 5 columns from the head and 5 from the tail



Principal Component Analysis

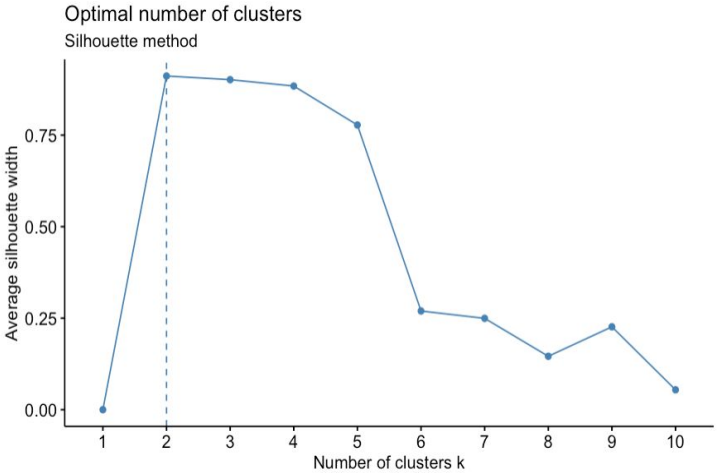




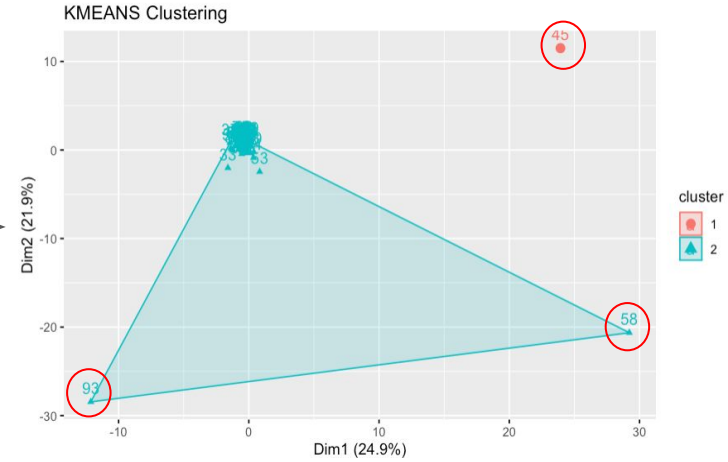


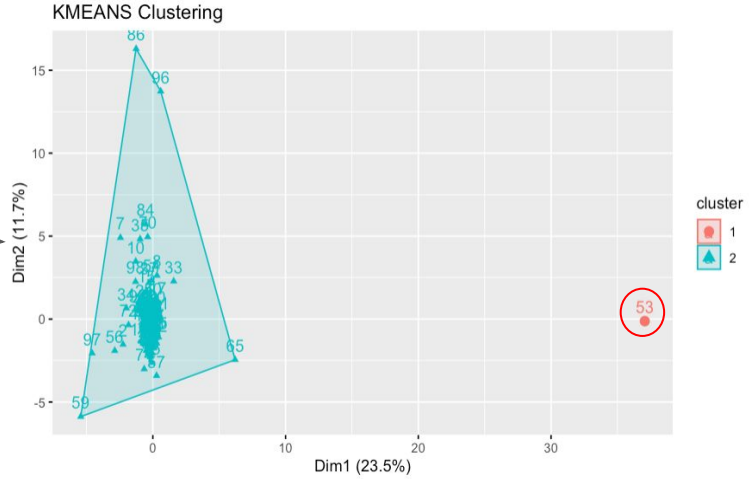
K-MEANS clustering on 100 patterns

Clustering before dropping the outliers



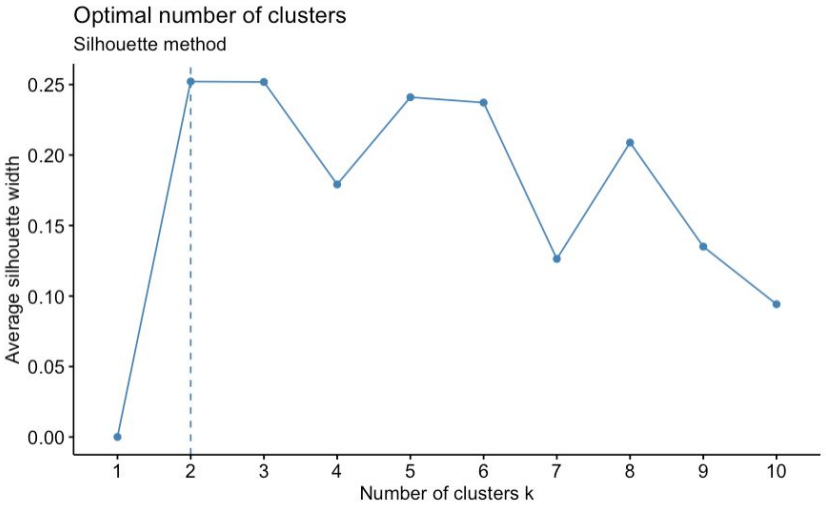
Initial *nbclust*

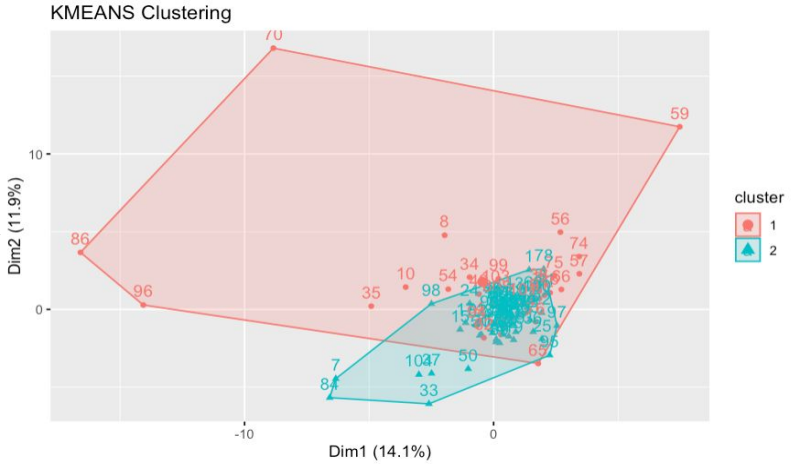


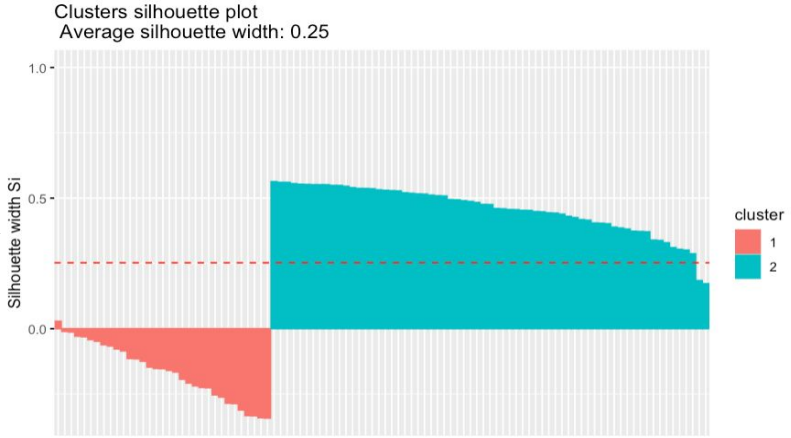


K-MEANS clustering on 100 patterns

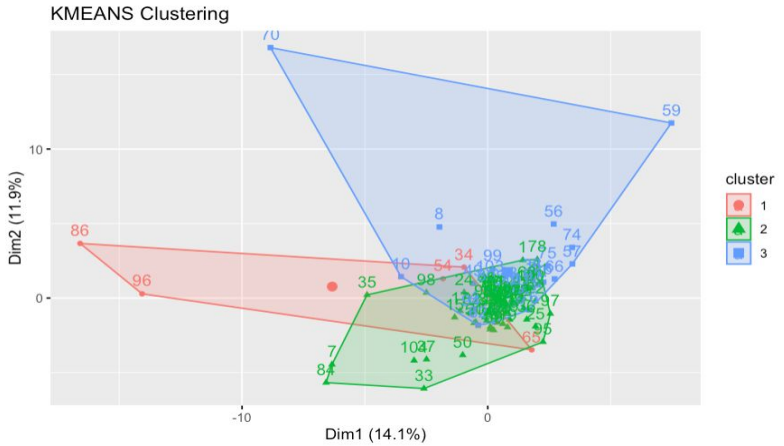
Clustering after dropping the outliers (2 clusters)

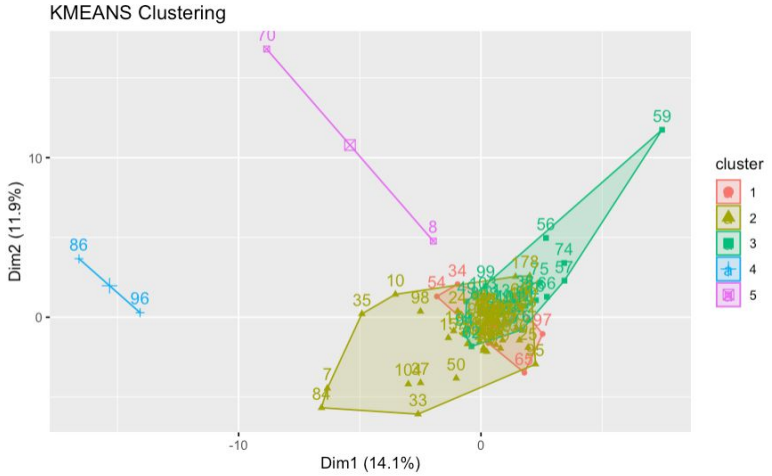


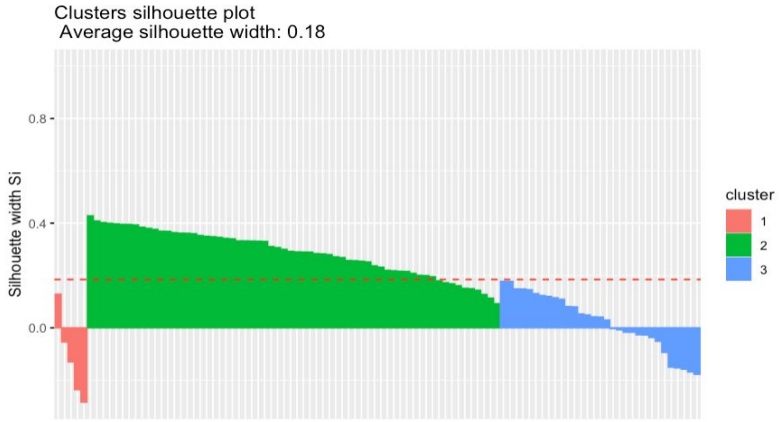


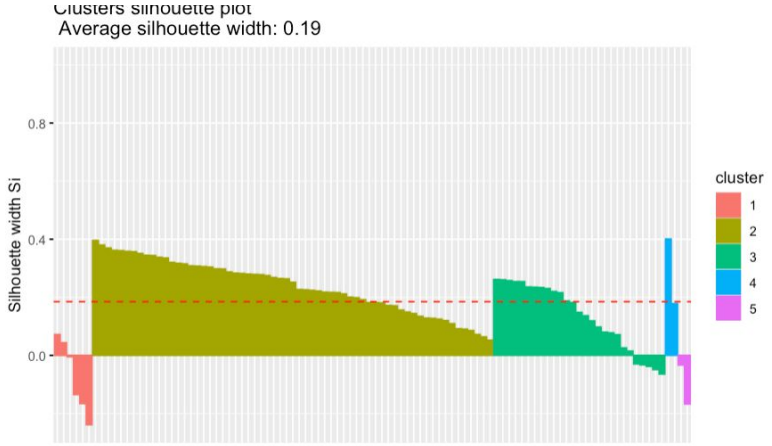


K-MEANS clustering on 100 patterns



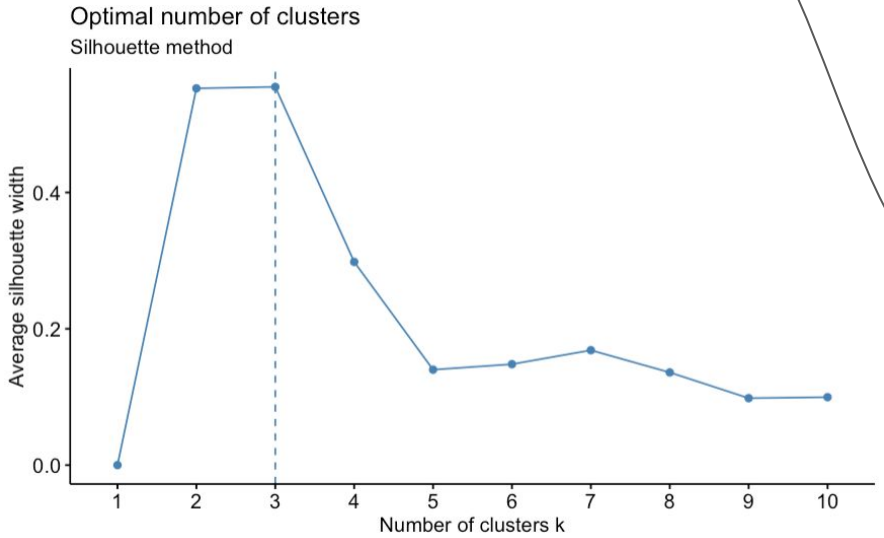


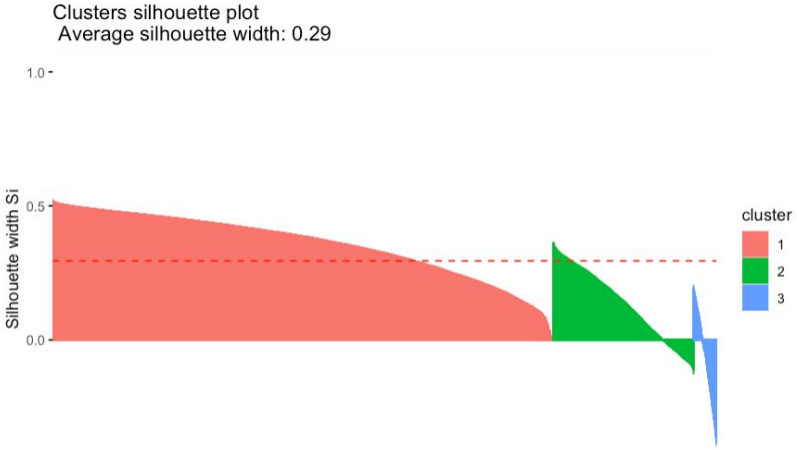




K-MEANS after Log Scaling

Clustering





Stocks dataset - Predicting

Model: **Logistic Regression**

Target: **PRICE VAR [%] > 0 ?**

Error measurement and hps tuning: **5-fold cross-validation**

- Log-transform improved error rate by **~3%**
- Mean validation error rate: **~39%**

Note: Using PCA for features extraction does not improve the error, but it makes the training faster (since it's less data) with an error that is only slightly worse

Stocks dataset - Predicting

Model: **Linear Regression**

Target: **PRICE VAR [%]**

Error measurement and hps tuning: **5-fold cross-validation**

- dim-reduction using PCA did not improve error
- Log-transform improved error by **~2**
- Mean validation absolute error: **~30**

Note: Using the linear regressor as a binary predictor yields the same error rate as the logistic regressor, but is about 30x faster to train

Dataset 2: Pima Indians Diabetes

Source: National Institute of Diabetes and Digestive and Kidney Diseases

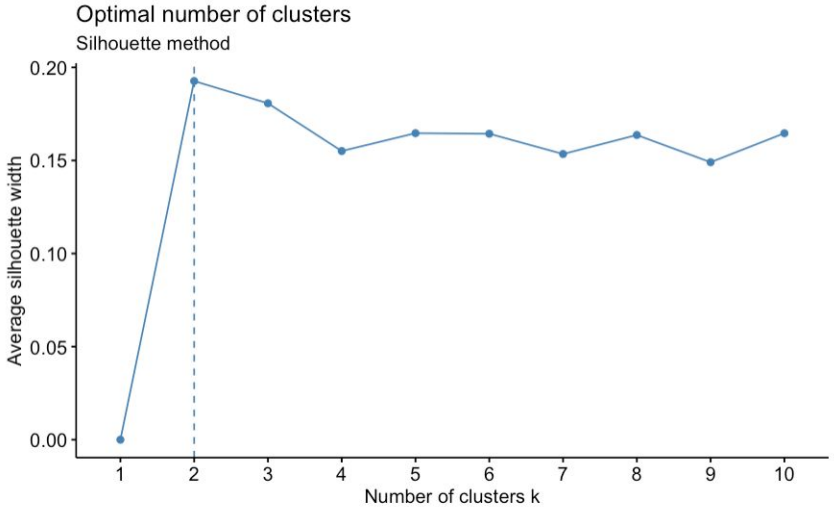
Subjects: 768 patients, all females at least 21 years old of Pima Indian heritage

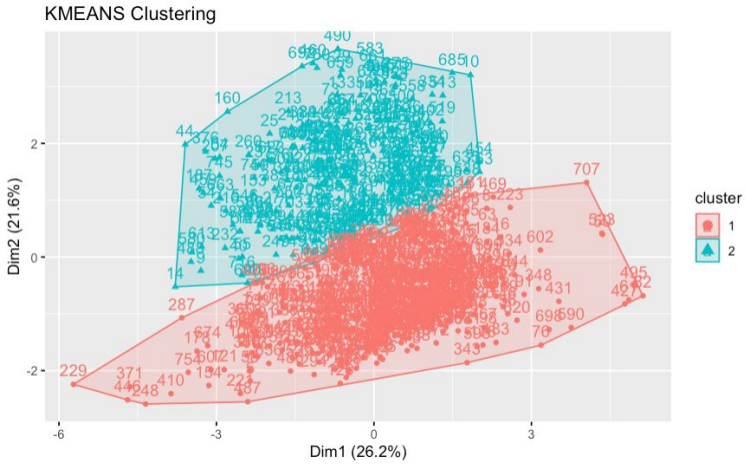
Measures per patient: N. of pregnancies, Glucose Level, Blood Pressure, Skin Thickness, Insulin Level, BMI, Diabetes Pedigree Function, Age

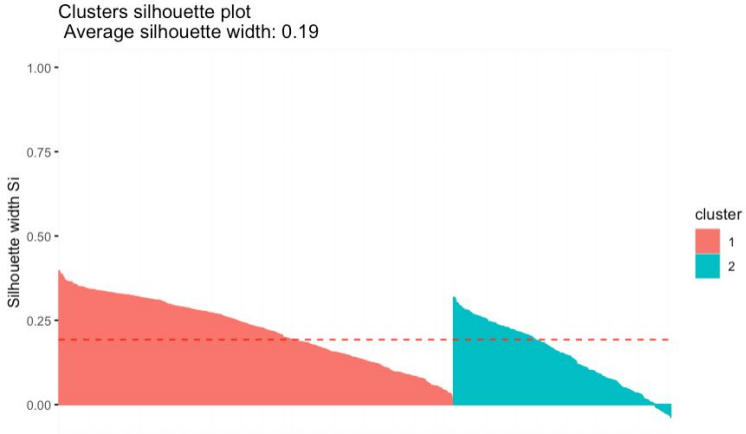
Goal: predict whether the patient suffers from diabetes

K-MEANS clustering on all the patterns

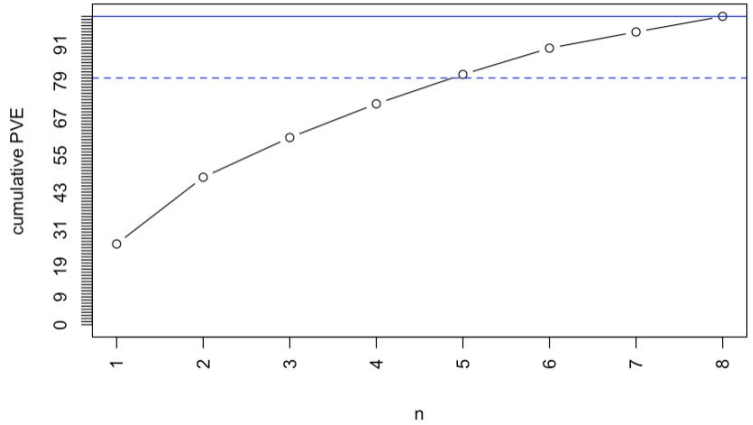
Original components space

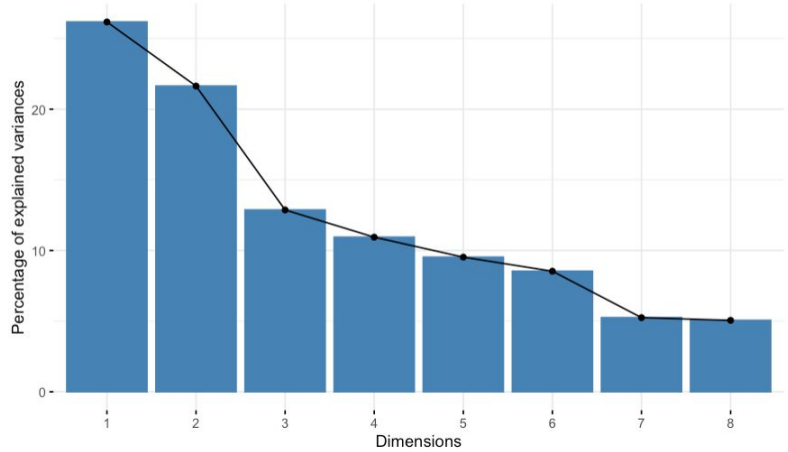


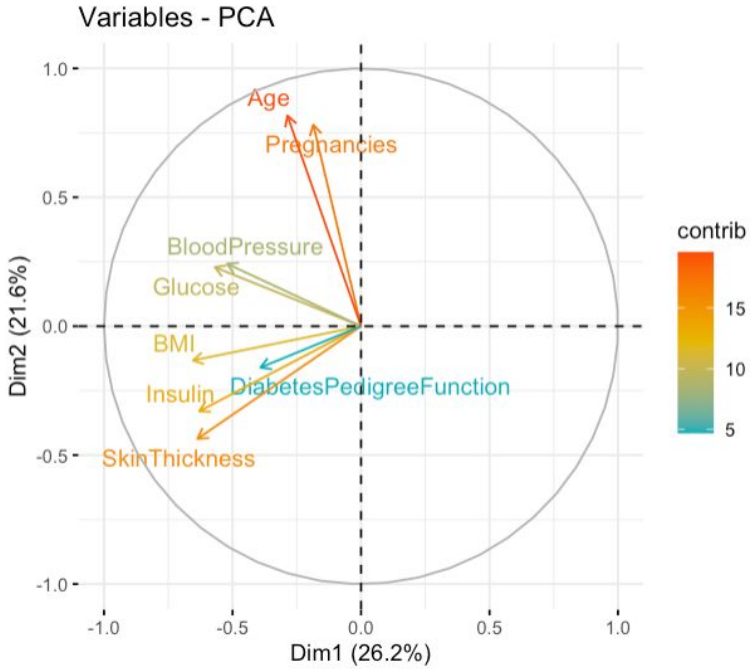




Principal Component Analysis

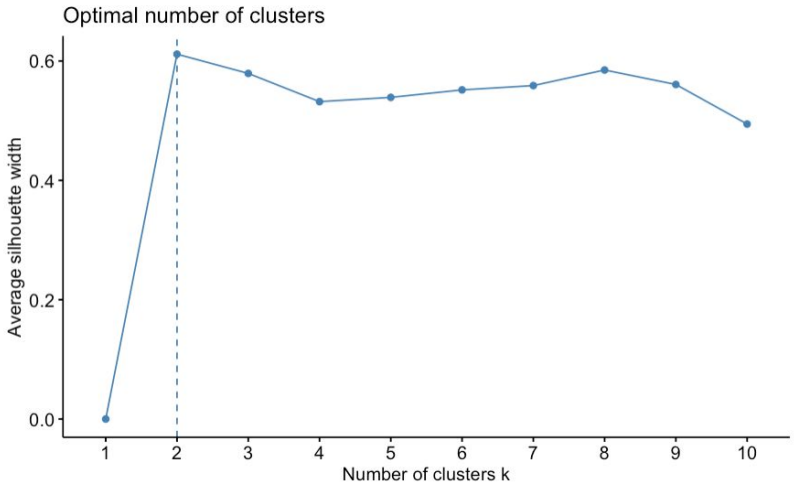


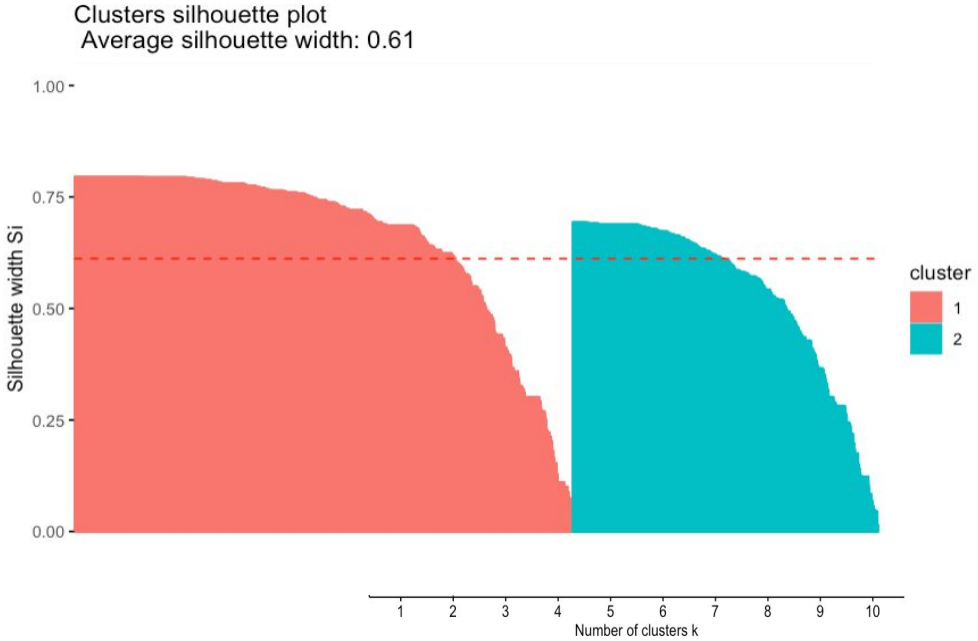




K-MEANS clustering on all the patterns

Principal components space



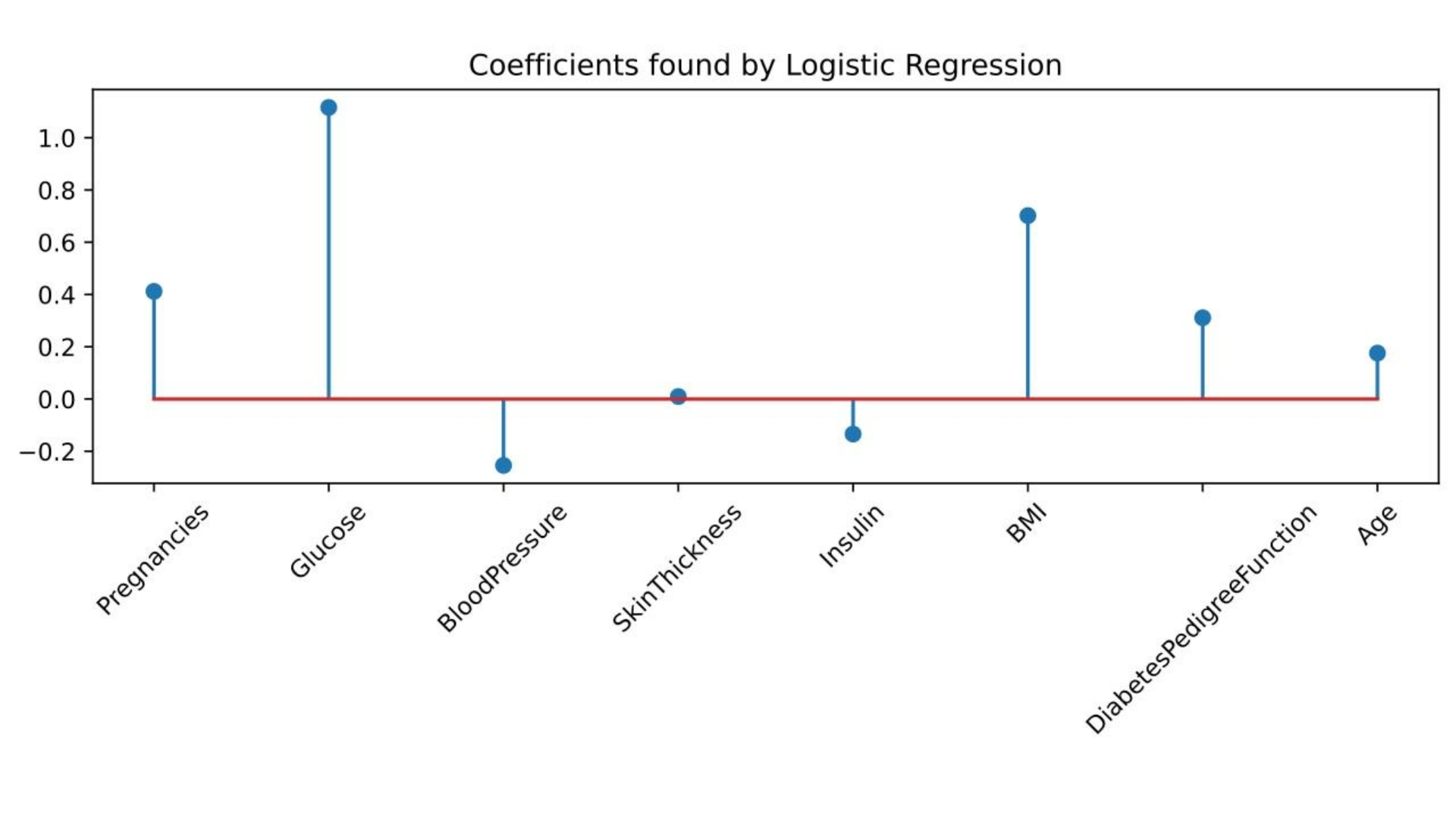


Diabetes Dataset - Predicting

Model: **Logistic Regression**

Error measurement and hps tuning: **5-fold cross-validation**

- Log-transform did not improve accuracy
- Mean validation error rate: **~22%**


Coefficients found by Logistic Regression
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
Pregnancies
Glucose
BloodPressure
SkinThickness
Insulin
BMI
DiabetesPedigreeFunction
Age

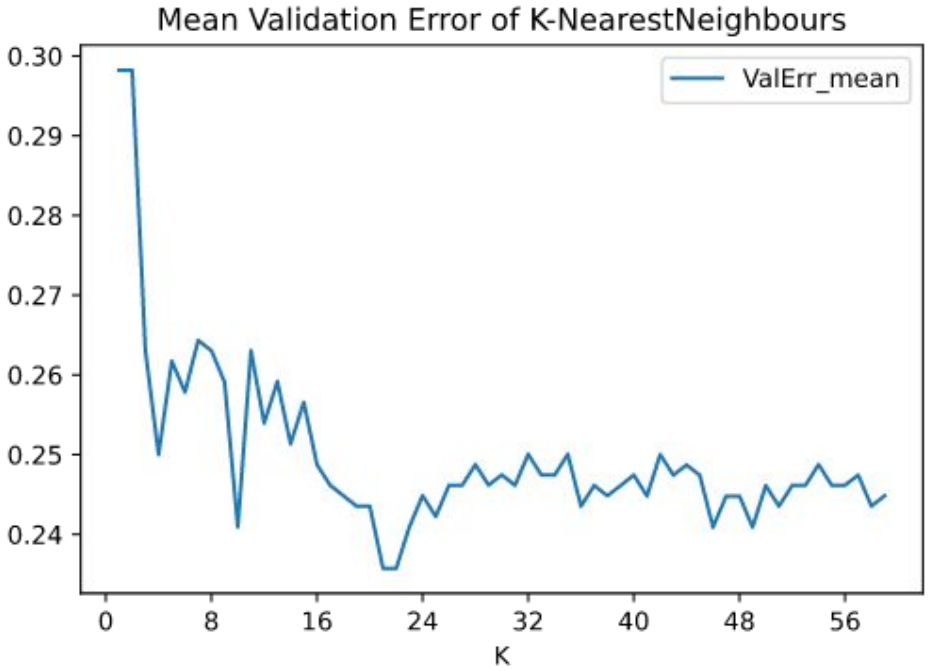
Diabetes Dataset - Predicting

Model: **K-Nearest Neighbours**

Error measurement and hps tuning:

5-fold cross-validation

- **Optimal K found**: 21
- mean validation error rate: **~23,5%**



Predicting using Neural Networks

For both dataset, we have also applied a neural network model (error validation and hyper-parameters tuning using 5-fold cross-validation).

The neural model did not perform considerably better than the logistic regressor (for the Diabetes Dataset) and the linear regressor (for the Stocks Dataset).

This suggests that the models used already have enough expressivity to capture the information present in both datasets. Therefore, to achieve better results, one or more of this factors might be needed:

- **samples**
- **meaningful features**
- **other forms of data-preprocessing**

Thank you for your attention :D

Questions?