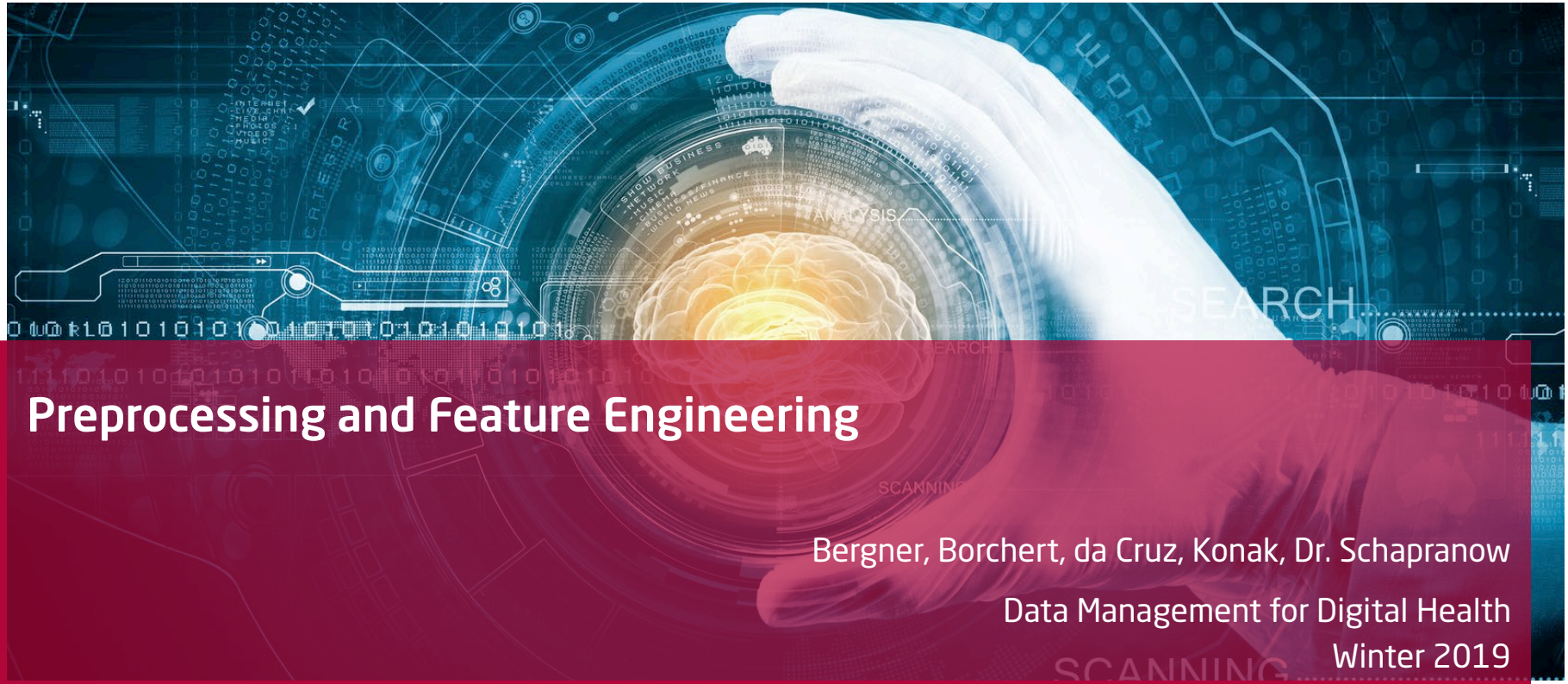
Hasso Plattner Institut
Digital Engineering · Universität Potsdam



Preprocessing and Feature Engineering

Bergner, Borchert, da Cruz, Konak, Dr. Schapranow

Data Management for Digital Health

Winter 2019

Agenda

Hasso Plattner Institut

Medical Use Cases

Biology Recap

Oncology

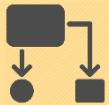
Nephrology and Intensive Care

Technology Foundation

Data Sources

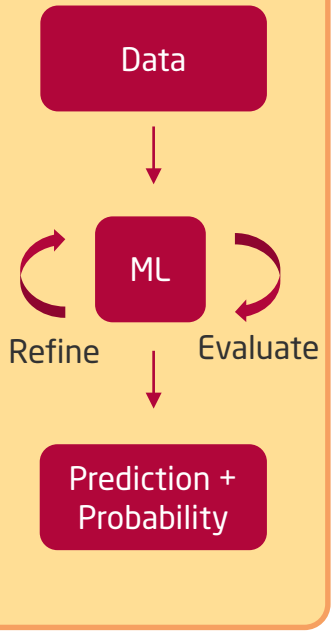
Data Formats

Processing and Analysis



Software Architectures

Machine Learning



Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

Agenda

Hasso Plattner Institut

Medical Use Cases

Biology Recap

Oncology

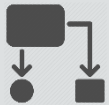
Nephrology and Intensive Care

Technology Foundation

Data Sources

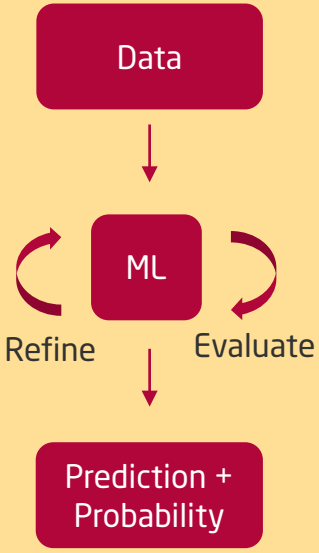
Data Formats

Processing and Analysis



Software Architectures

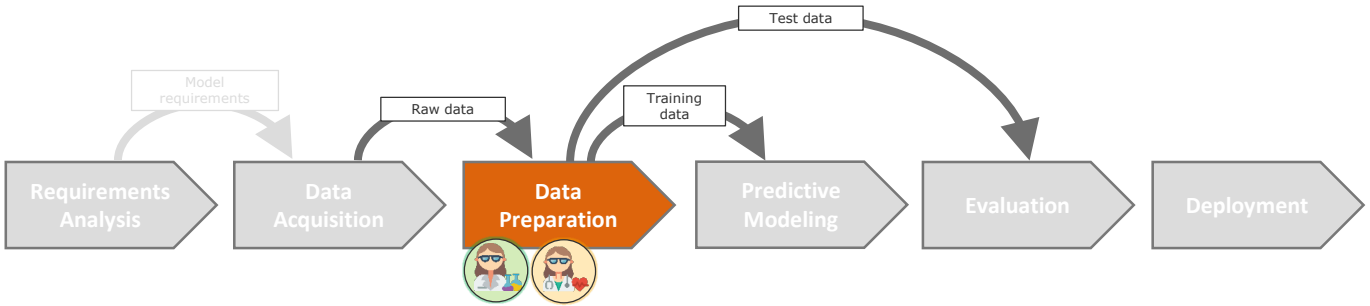
Machine Learning



Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

Data Preparation



Data Preparation

- Exploration
- Quality assessment
- Cleansing
- Labeling
- Imputation
- Feature engineering

Roles: Data Scientist, Domain Expert, (Data) Engineer

Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

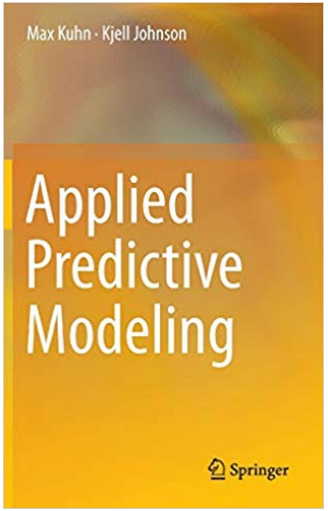
What Is Data Preparation

Data preparation can make or break the predictive ability of your model

According to Kuhn and Johnson data preparation is the process of **addition**, **deletion** or **transformation** of training set data

Sometimes, preprocessing of data can lead to unexpected improvements in model accuracy

Data preparation is an important step and you should experiment with data pre-processing steps that are appropriate for your data to see if you can get that desirable boost in model accuracy



Data Preparation Importance Motivation

Data in Healthcare → sparse and incomplete

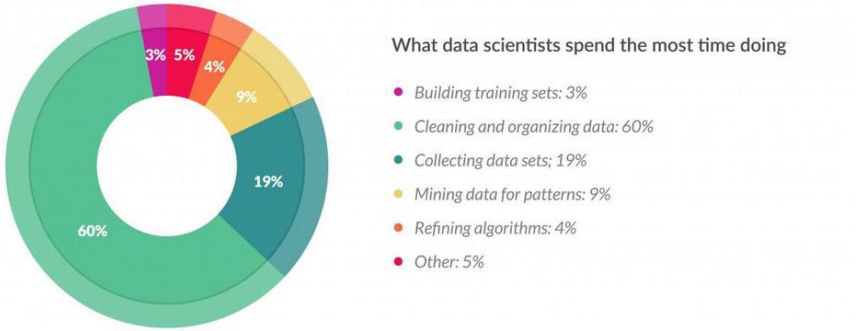
Preparing the proper input dataset, compatible with the machine learning algorithm requirements

Integral step in Machine Learning

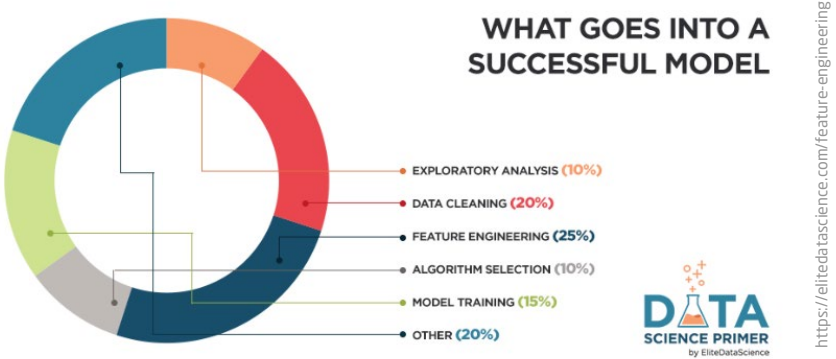
Directly affects the ability of our model to learn

Make sure that it is in a useful scale, format and even that meaningful features are included

Improving the performance of machine learning models



Source: https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/



Why Data Preparation Is so Important in Digital Health

Hasso Plattner Institut

algorithms exist but selecting the best classifier surely improves the accuracy of the predictions. The preprocessing methods selected in this study are Multiple Imputation, k-means for missing values treatment, Discretization to change in discrete values, Standard scaler, Min-Max scalar for feature scaling and, Random Forest (RF) for feature selection. For the classification Logistic Regression (LR), Artificial Neural Network (ANN), Support Vector Machine (SVM) and Random Forest (RF) are used. To evaluate the performance of model accuracy, sensitivity, specificity are used. This study compares the model performance with and without preprocessed data and has proved that the selected preprocessing methods significantly improves the model performance.

have used two different datasets of healthcare sector for predicting the type 2 diabetes onset. Firstly, we started

Table 5: Outcome of classifiers before and after applying pre-processing techniques on PIDDs datasets

Preprocessing Techniques	Classifier	Accuracy	Sensitivity	Specificity
No Preprocessing	LR	0.75	0.57	0.86
With Preprocessing	**LR**	**0.77**	**0.56**	**0.88**
No Preprocessing	ANN	0.67	0.25	0.89
With Preprocessing	**ANN**	**0.80**	**0.65**	**0.88**
No Preprocessing	SVM	0.65	0	1
With Preprocessing	**SVM**	**0.79**	**0.82**	**0.78**
No Preprocessing	Random Forest	0.74	0.48	0.89
With Preprocessing	**Random Forest**	**0.78**	**0.67**	**0.84**

Table 6: Outcome of classifiers before and after applying pre-processing techniques on LUDB2 datasets

Preprocessing Techniques	Classifier	Accuracy	Sensitivity	Specificity
No Preprocessing	LR	0.98	0.96	1.0
With Preprocessing	**LR**	**0.98**	**1.0**	**0.96**
No Preprocessing	ANN	0.94	0.88	1.0
With Preprocessing	**ANN**	**0.96**	**0.92**	**1.0**
No Preprocessing	SVM	0.74	1.0	0.48
With Preprocessing	**SVM**	**1.0**	**1.0**	**1.0**
No Preprocessing	Random Forest	1.0	1.0	1.0
With Preprocessing	**Random Forest**	**1.0**	**1.0**	**1.0**

2nd INTERNATIONAL CONFERENCE ON ADVANCED COMPUTING AND SOFTWARE ENGINEERING (ICACSE-2019)

Impact of Preprocessing Methods on Healthcare Predictions

Puneet Misra[1] and Arun Singh Yadav[2]

Abstract—Machine learning (ML) is now a day gaining immense importance and is becoming a key technology as the rapid growth of quality of medical data and information. But the early and accurate detection of disease is still a challenge due to the complex, incomplete and multidimensional healthcare data. Data preprocessing is an essential step of ML whose primary goal is to provide processed data to improve the prediction accuracy. This study summarizes the popular data preprocessing steps based on their usages, popularity and literature. After that the selected preprocessing methods is applied on the raw data which is then used by classifiers for predictions. In this experiment we have taken diabetes classification problem. Type II diabetes mellitus (T2DM) is a major disease with high penetrance in human around the world and still rising. This may cause other serious complications like kidney failure, heart failure, blindness etc. The early detection and diagnosis help to identify and may avoid these complications. Several classification algorithms exist but selecting the best classifier surely improves the accuracy of the predictions. The preprocessing methods selected in this study are Multiple Imputation, k-means for missing values treatment, Discretization to change in discrete values, Standard scaler, Min-Max scalar for feature scaling and, Random Forest (RF) for feature selection. For the classification Logistic Regression (LR), Artificial Neural Network (ANN), Support Vector Machine (SVM) and Random Forest (RF) are used. To evaluate the performance of model accuracy, sensitivity, specificity are used. This study compares the model performance with and without preprocessed data and has proved that the selected preprocessing methods significantly improves the model performance.

Keywords: Machine learning, Disease prediction, Classification, Preprocessing, Multiple Imputation, k-NN, Standard Scaler, Min-Max scalar, RFE, LR, ANN, SVM, RF

I. Introduction

Machine Learning (ML) is gaining importance day by day due to its capability work with heterogeneous data set. ML algorithms directly learn from the data, produce the hidden insights and can predict or forecast the future outcomes on the basis of its learning[1]. Predictions can be done either by classification or regression approach. The classification accuracy of the predictions depends on the quality of the data. Data generated from various sources may have missing values, noisy, inconsistent, voluminous and class imbalanced[2]. This imperfect data requires data preparation stage to clean and prepare the data[3] for further analysis. To get quality of data, machine learning provides one of the most meaningful steps called data preprocessing. This step usually takes the significant amount of time[4] and should be implemented carefully to improve the overall model performance.

The data preprocessing task includes certain steps like data preparation, integration, cleaning, normalization, scaling and data reduction techniques to reduce complexity, to noisy and irrelevant elements using feature selection and discretization etc. After this the outcome expected a final dataset for further analysis using ML

The main objective of this paper to study the impact of selected preprocessing techniques on prediction value and justify that data preprocessing using intelligent techniques significantly improves the model performance. Another objective is to study various preprocessing methods and select the best among them and this selection has been done on the basis of their usages, popularity and should have been widely cited by research community.

We have started with the study of the most influential data preprocessing algorithms under various section of it and select the widely used among them. In next section we discussed about the dataset one public and another real-world. The experimental setup and proposed framework is explained in next section. We further described the model evaluation. The detailed discussion on the result and performance of the model before and after applying data processing techniques is covered in next section and the last section covered the conclusion and future work.

II. Study of Data Processing Algorithms

A. Data Cleaning

The data taken from the real-world problem is seldom clean and complete, specially the healthcare field. The

https://www.researchgate.net/publication/332436103_Impact_of_Preprocessing_Methods_on_Healthcare_Predictions

Data Preparation Steps

How do I clean up the data? → Data Cleaning

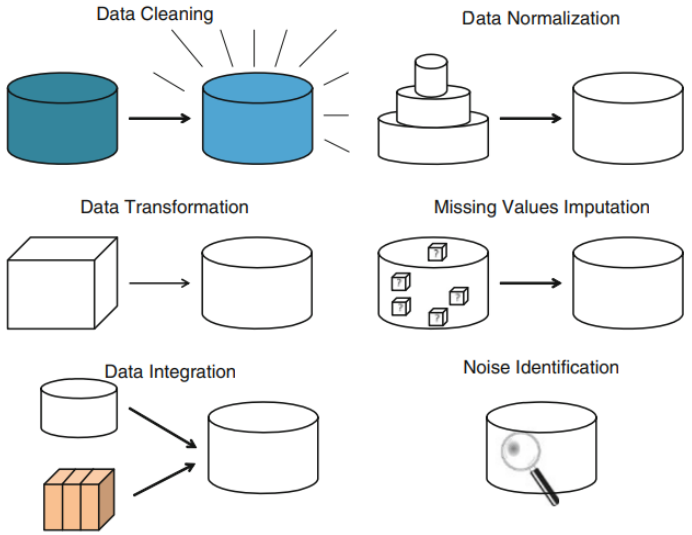
How do I provide accurate data? → Data Transformation

How do I incorporate and adjust data? → Data Integration

How do I unify and scale data? → Data Normalization

How do I handle missing data? → Missing Data Imputation

How do I detect and manage noise? → Noise Identification

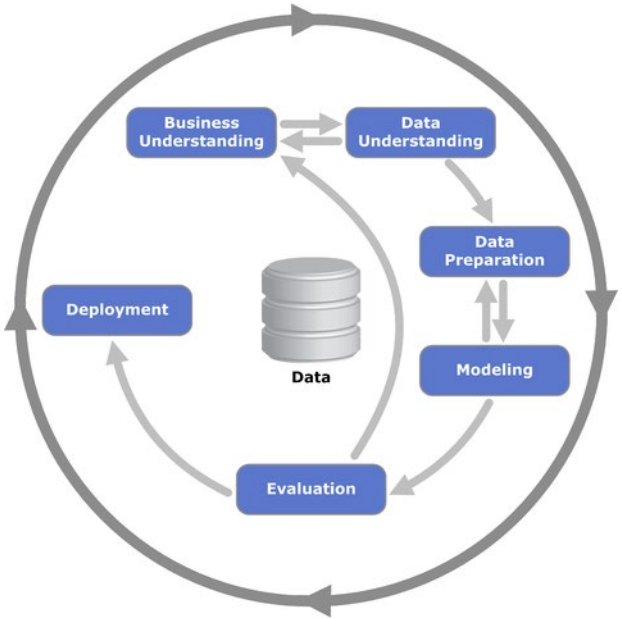


Data Preparation Process

Process for getting data ready for a machine learning algorithm can be summarized

- **Step 1**: Select Data
- **Step 2**: Preprocess Data
- **Step 3**: Transform Data

Follow this process in a linear manner



https://statistik-dresden.de/archives/1128

Select Data

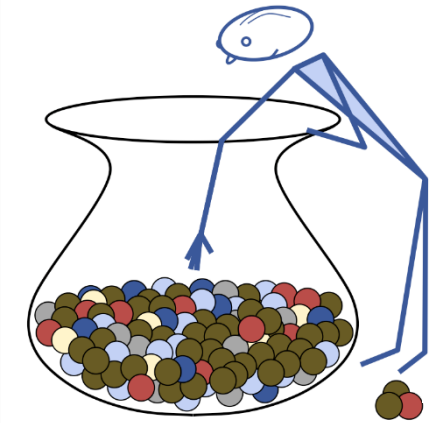
Hasso Plattner Institut

There is always a strong desire for including all data that is available, that the maxim "more is better" will hold. This may or may not be true

Consider what data you actually need to address the question or problem you are working on

Questions to help you think:

- What is the extent of the data you have available?
- What data is not available that you wish you had available?
- What data don't you need to address the problem?



http://uniquerecall.com/

Preprocess Data
Better Data > Fancier Algorithms

Hasso Plattner Institut

Formatting: Selected data may not be in a suitable format

Cleaning: Removal or fixing of missing data

- Incomplete and do not carry the data to address the problem
- Sensitive information → anonymized or removed
- Identifying incomplete, incorrect, inaccurate, irrelevant parts of the data

Sampling: More selected data available than needed

- Longer running times for algorithms
- Larger computational and memory requirements
- Take smaller representative sample before considering the whole dataset

https://www.flickr.com/photos/marc_smith/1473557291/sizes/l/

Dummy Variables

Transforming categorical attribute to numerical attribute

Each attribute will have value either 0 or 1

Full Dummy Variables: Represent n categories using n dummy variables, one variable for each level

Dummy Variables with Reference Group: Represent the categorical variable with n categories using n-1 dummy variables

Dummy Variables for Ordered Categorical Variable with Reference Group: Assume mathematical ordering Small < Medium < Large. To indicate the ordering, use more 1s for higher categories

	X_0	X_1	X_2
Small	1	0	0
Medium	0	1	0
Large	0	0	1

Also known as One-Hot Encoding!

	X_1	X_2
Small	0	0
Medium	1	0
Large	0	1

← Reference Group

	X_1	X_2
Small	0	0
Medium	1	0
Large	1	1

https://de.mathworks.com/help/stats/dummy-indicator-variables.html

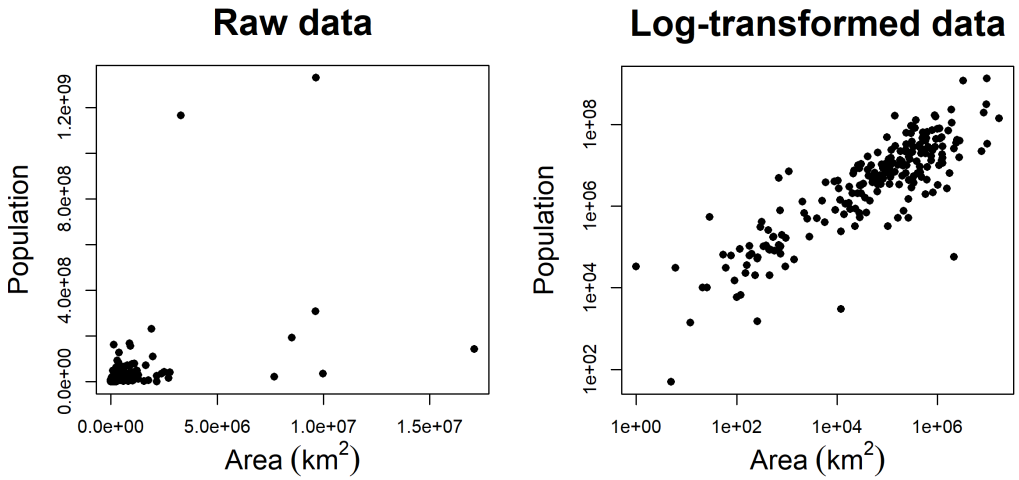
Transformed Attributes

Hasso Plattner Institut

Data transformation changes relative differences among individual values

Types of transformation:

- Linear: By adding constant or multiplying by constant
- Non-linear: log-transformation, square-root transformation etc.

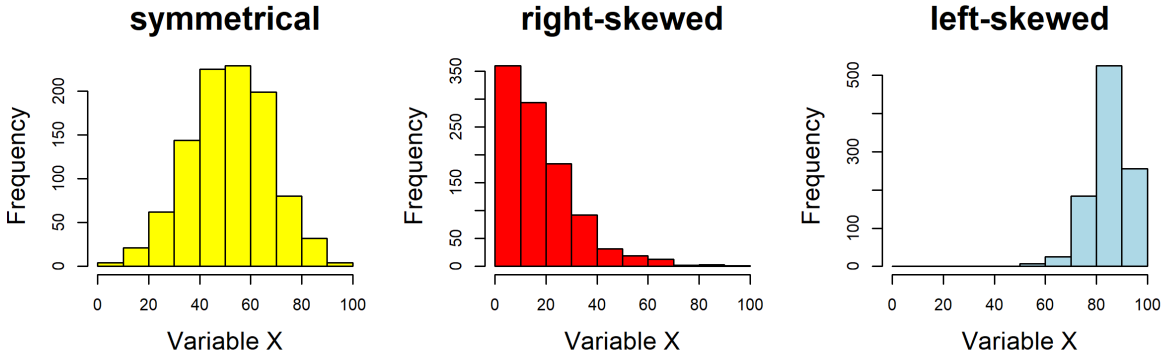


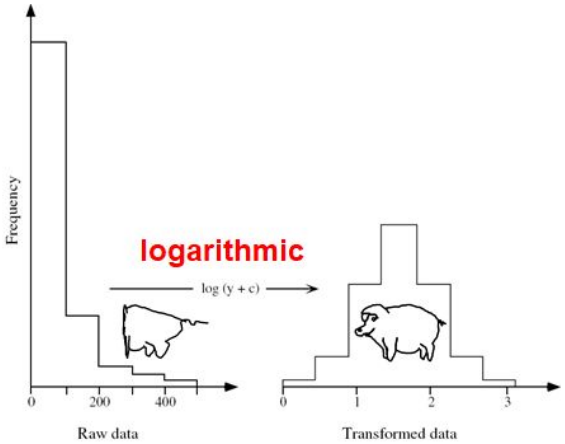
https://www.davidzeleny.net/anadat-r/doku.php/en:data_preparation

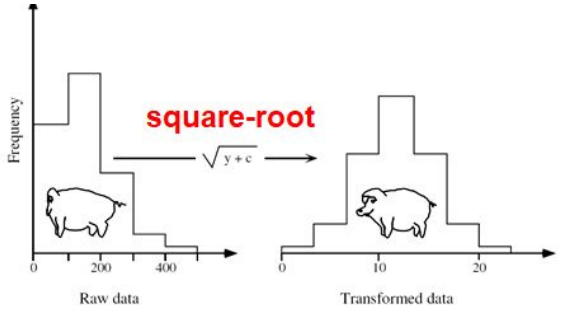
Transformed Attributes
Box-Cox

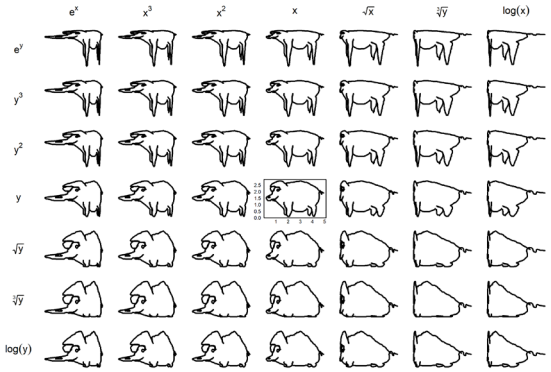
Hasso Plattner Institut

log transformation is suitable for strongly right-skewed data, sqrt transformation is suitable for slightly right-skewed data









https://www.davidzeleny.net/anadat-r/doku.php/en:data_preparation

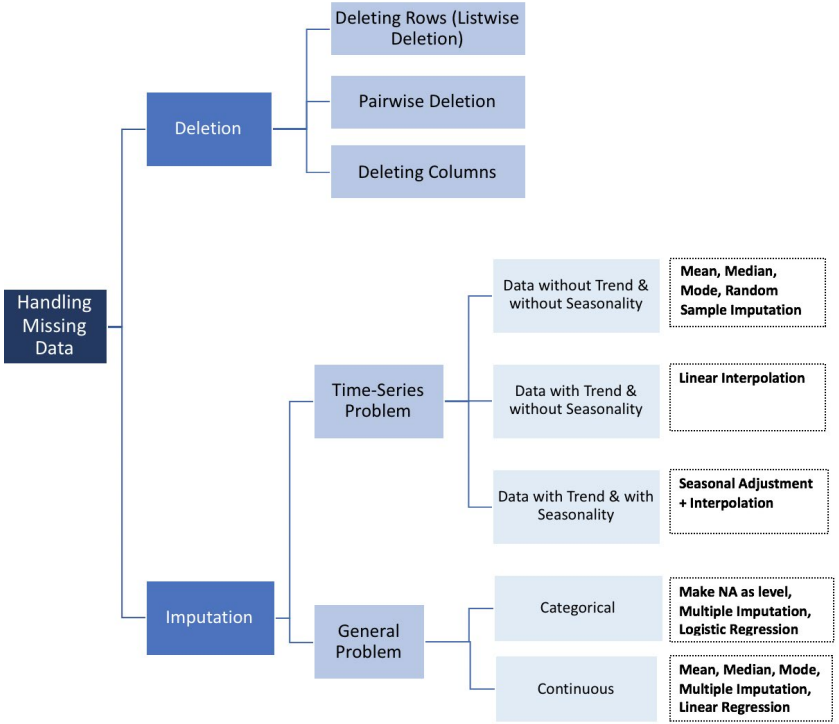
How to Handle Missing Data

Hasso Plattner Institut

There is NO good way to deal with missing data!

Different solutions for data imputation depending on the kind of problem – Time series Analysis, ML, Regression etc.

No general solution



https://towardsdatascience.com/how-to-handle-missing-data-8646b18db0d4

Data Imputation (Mean/Median) Values

Hasso Plattner Institut

Calculating the mean/median of the non-missing values in a column

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

mean()

	col1	col2	col3	col4	col5
0	2.0	5.0	3.0	6.0	7.0
1	9.0	11.0	9.0	0.0	7.0
2	19.0	17.0	6.0	9.0	7.0

Pros	Cons
Easy and fast	Doesn't factor the correlations between features. It only works on the column level
Works well with small numerical datasets	Will give poor results on encoded categorical features (do NOT use it on categorical features)
	Not very accurate
	Doesn't account for the uncertainty in the imputations

Data Imputation (Most Frequent) or (Zero/Constant) Values

Hasso Plattner Institut

Most Frequent statistical strategy to impute missing values

Replacing missing data with the most frequent values within each column

Pros	Cons
Works well with categorical features	It also doesn't factor the correlations between features
	It can introduce bias in the data

Zero or Constant imputation replaces the missing values with either zero or any constant value you specify

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	NaN
1	9	NaN	9.0	0	7.0
2	19	17.0	NaN	9	NaN

df.fillna(0)

	col1	col2	col3	col4	col5
0	2	5.0	3.0	6	0.0
1	9	0.0	9.0	0	7.0
2	19	17.0	0.0	9	0.0

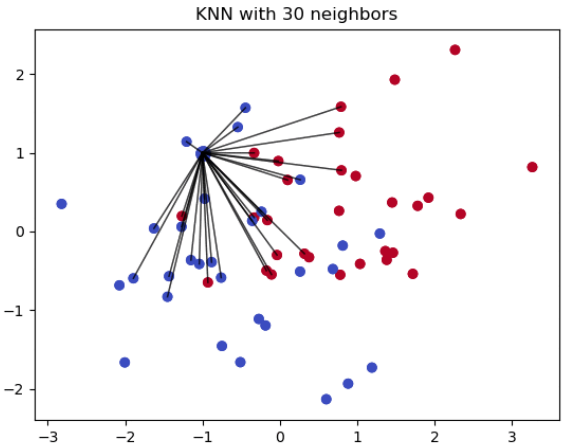
Data Imputation
k-NN

Hasso Plattner Institut

k nearest neighbours is an algorithm that is used for simple classification

Algorithm uses 'feature similarity' to predict the values of any new data points

New point is assigned a value based on how closely it resembles the points in the training set



Pros	Cons
Can be much more accurate than the mean, median or most frequent imputation methods (It depends on the dataset)	Computationally expensive. KNN works by storing the whole training dataset in memory
	K-NN is quite sensitive to outliers in the data (unlike SVM)

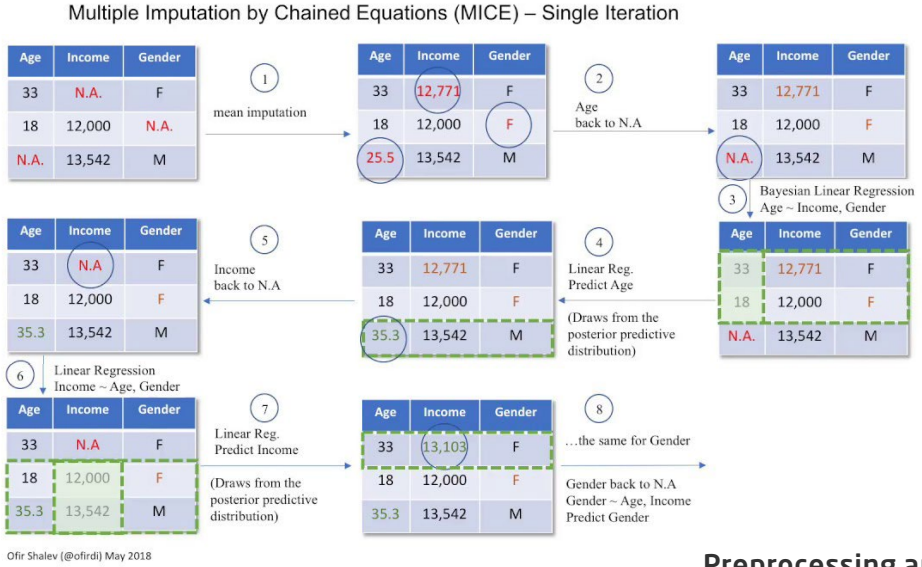
Data Imputation
Multivariate Imputation

Hasso Plattner Institut

Filling the missing data multiple times

Multiple Imputations (MIs) are much better than a single imputation as it measures the uncertainty of the missing values in a better way

Chained equations approach is also very flexible and can handle different variables of different data types



https://www.youtube.com/watch?v=zX-pacwVyvU

Data Reduction

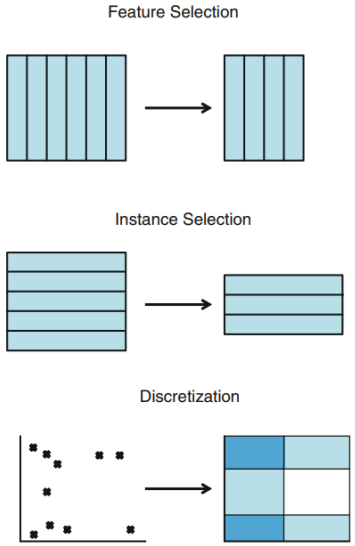
Hasso Plattner Institut

How do I reduce the dimensionality of data? → Feature Selection (FS)

How do I remove redundant and/or conflictive examples? → Instance Selection (IS)

How do I simplify the domain of an attribute? → Discretization

How do I fill in gaps in data? → Feature Extraction and/or Instance Generation



Projection
Principal Component Analysis (PCA)

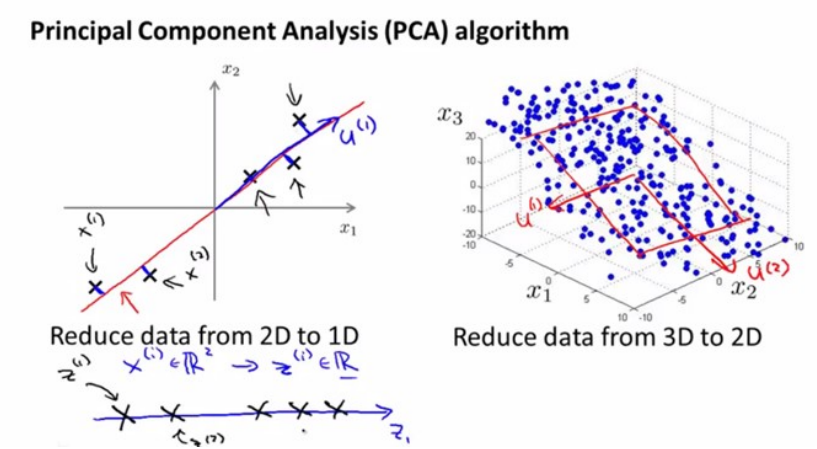
Hasso Plattner Institut

As the amount of data grows in the world, the size of datasets available for ML development also grows

Dimensionality reduction involves the transformation of data to new dimensions in a way that facilitates discarding of some dimensions without losing any key information

Large-scale problems bring about several dimensions that can become very difficult to visualize

Some of such dimensions can be easily dropped for a better visualization

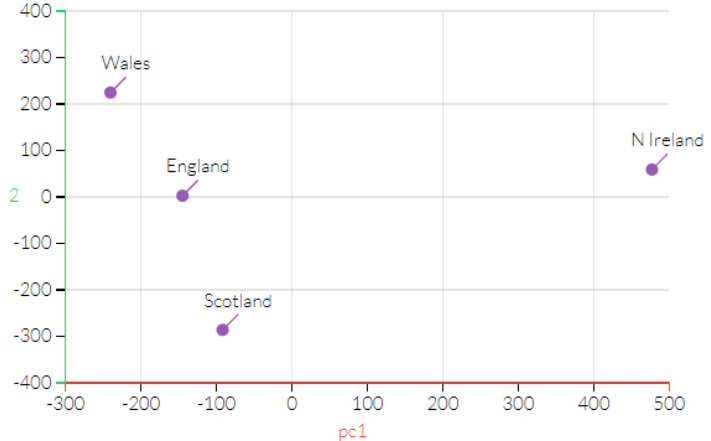


https://www.dezyre.com/data-science-in-python-tutorial/principal-component-analysis-tutorial

Applications of PCA

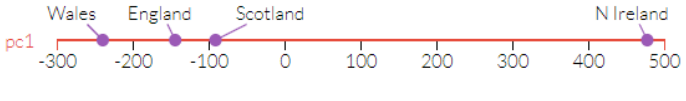
Hasso Plattner Institut

Pros	Cons
Removes Correlated Features	Independent variables become less interpretable
Improves Algorithm Performance	Data standardization is must before PCA
Reduces Overfitting	Information Loss
Improves Visualization	



	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

http://setosa.io/ev/principal-component-analysis/



Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

Fourier Transformation

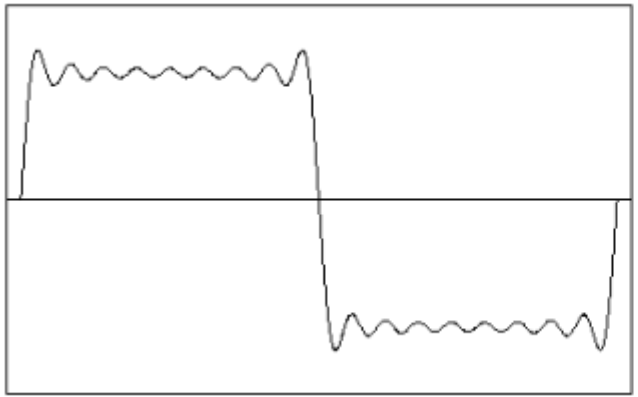
Hasso Plattner Institut

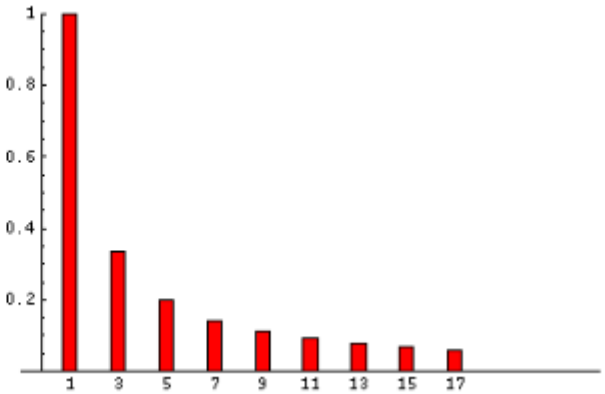
Fourier showed that any periodic signal $s(t)$ can be written as a sum of sine waves with various amplitudes, frequencies and phases

$$s(t) = a_0 + a_1 \sin(\omega t + \phi_1) + a_2 \sin(2\omega t + \phi_2) + a_3 \sin(3\omega t + \phi_3) + \cdots$$

For example, the Fourier expansion of a square wave can be written as

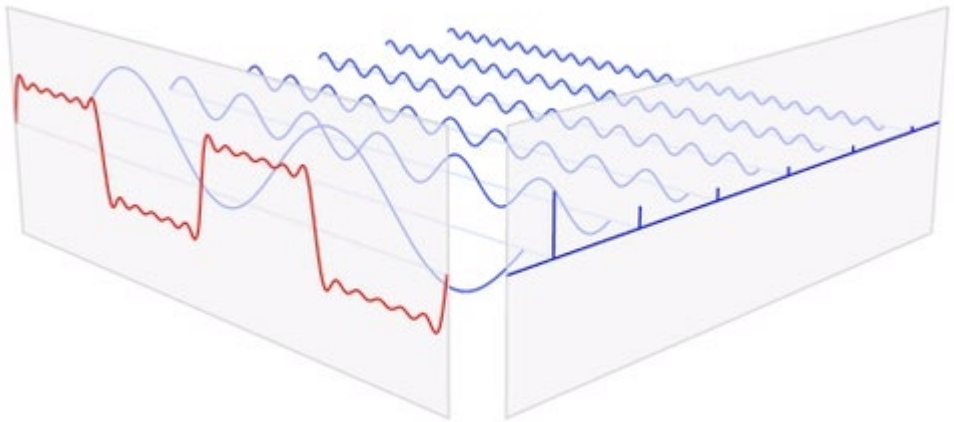
$$s(t) = \sin(\omega t) + \frac{1}{3}\sin(3\omega t) + \frac{1}{5}\sin(5\omega t) + \frac{1}{7}\sin(7\omega t) + \cdots$$





http://mriquestions.com/fourier-transform-ft.html

Fast Fourier Transform

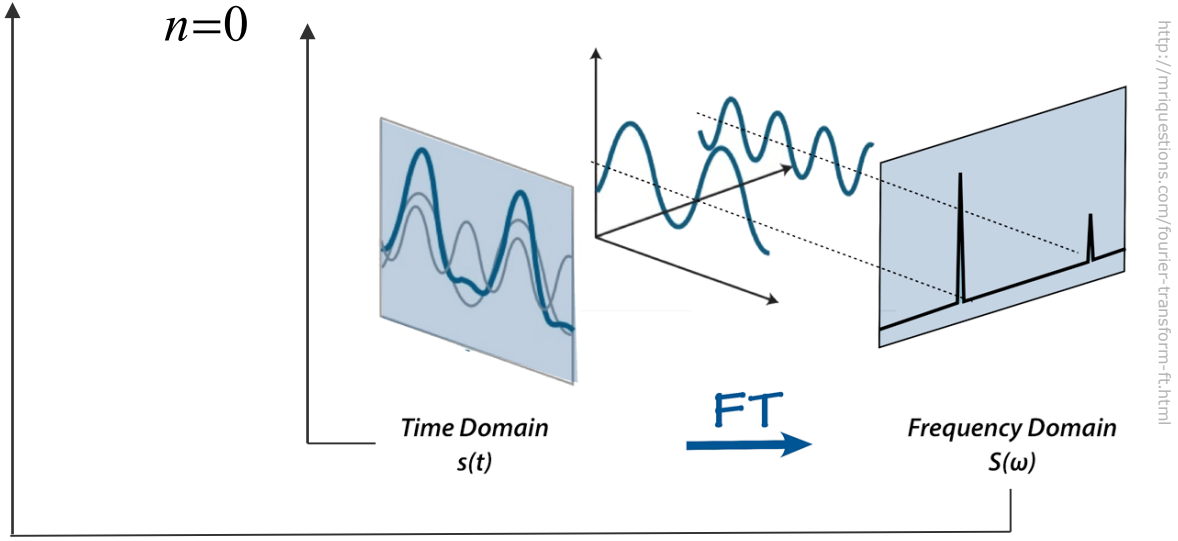


https://giphy.com/gifs/fourier-transform-Km4XeiMqFNCDK

Discrete Fourier Transform

Hasso Plattner Institut

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i 2\pi \frac{k}{N} n}$$

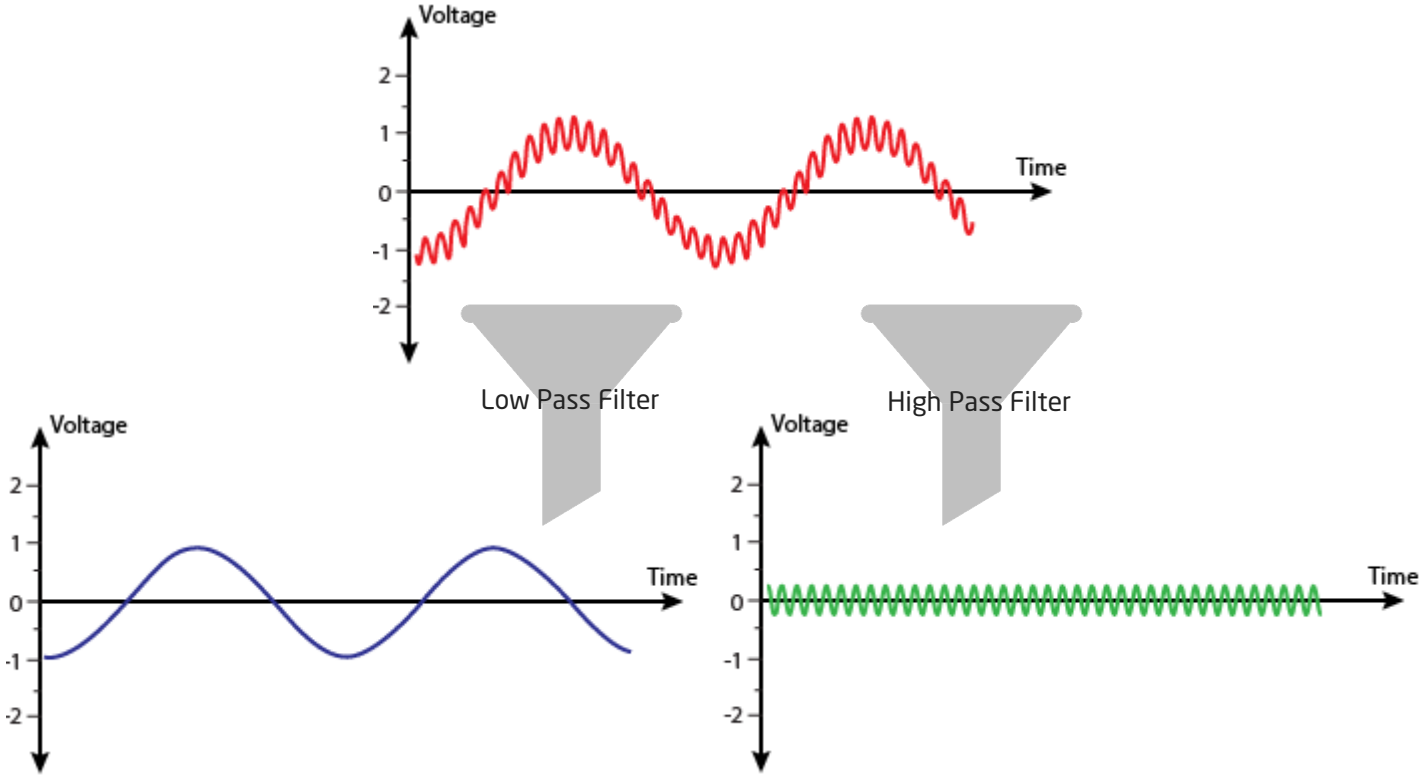


http://mriquestions.com/fourier-transform-ft.html

Fourier series in 1822

https://de.wikipedia.org/wiki/Joseph_Fourier

Filter



https://www.adinstruments.com/tips/data-quality

Fourier Transformation

Hasso Plattner Institut

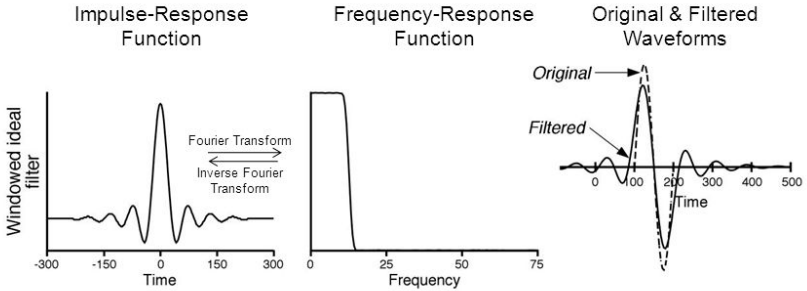
Important signal processing tool

Used to decompose a signal into its sine and cosine components

Output of the transformation represents the signal in the Fourier or frequency domain

Apply mathematical operations to eliminate certain frequency domains very easily

Applying the inverse Fourier transform to recover the original time signal



https://slideplayer.com/slide/4173668/

Correlation

Hasso Plattner Institut

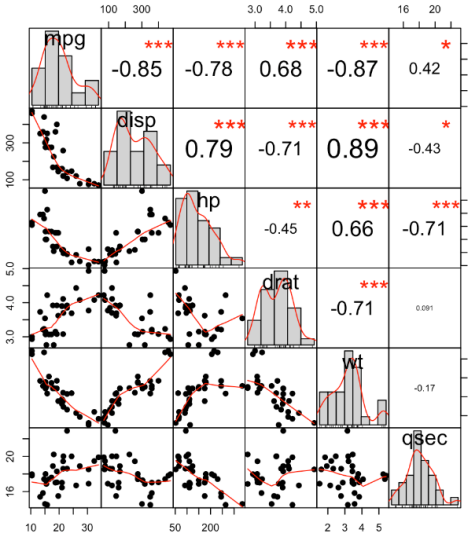
Way to understand the relationship between multiple variables and attributes in your dataset

Using Correlation, you can get some insights such as:

- One or multiple attributes depend on another
- One or multiple attributes are associated with other attributes

Can help in predicting one attribute from another (great way to impute missing values)

Can (sometimes) indicate the presence of a causal relationship

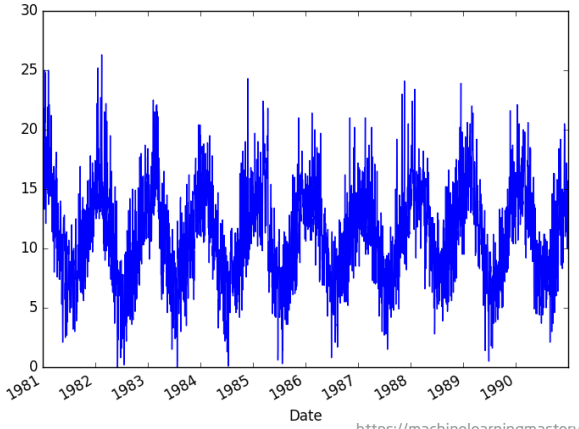


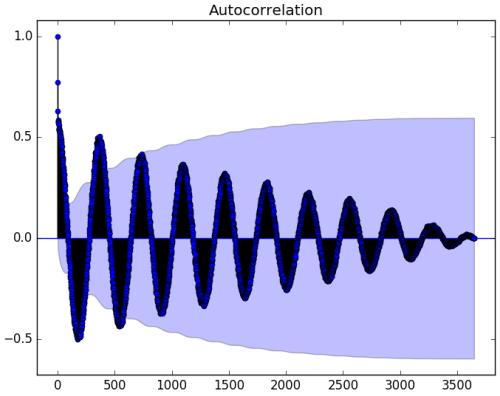
http://www.sthda.com/english/wiki/correlation-analyses-in-r

Autocorrelation

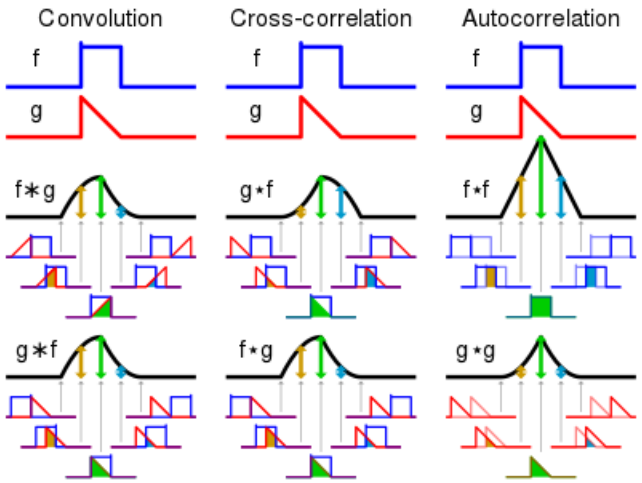
Hasso Plattner Institut

- Heavily used in time series analysis and forecasting
- Measure of the correlation between the lagged values of a time series
- Uncover hidden patterns in data
- Identify seasonality and trend in our time series data





https://machinelearningmastery.com/gentle-introduction-autocorrelation-partial-autocorrelation/



https://en.wikipedia.org/wiki/Autocorrelation

Transform Data

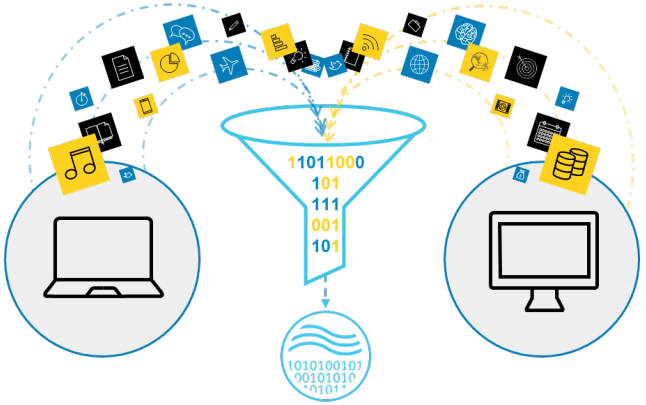
Hasso Plattner Institut

Scaling: The preprocessed data may contain attributes with a mixtures of scales for various quantities. Many machine learning methods like data attributes to have the same scale

Decomposition: There may be features that represent a complex concept that may be more useful to a machine learning method when split into the constituent parts

¡ Example → Date

Aggregation: There may be features that can be aggregated into a single feature



https://blog.dellemc.com/en-us/digital-transformation-just-got-easier-with-analytic-insights/

Standardization (Variance Scaling)

Hasso Plattner Institut

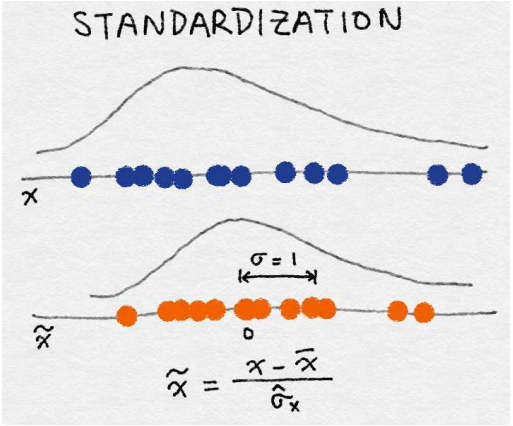
$$\tilde{x} = \frac{x - \text{mean}(x)}{\text{sqrt}(\text{var}(x))}$$

It subtracts off the mean of the feature (over all data points) and divides by the variance

It can also be called *variance scaling*

resulting scaled feature has a mean of 0 and a variance of 1

If the original feature has a Gaussian distribution, then the scaled feature does too



Feature Engineering for Machine Learning
Principles and Techniques for Data Scientists
Alice Zheng and Amanda Casari, O'Reilly, 2018

Min-Max Scaling

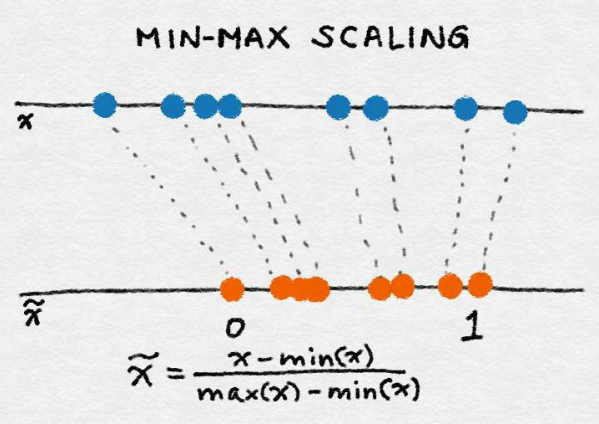
Hasso Plattner Institut

$$\tilde{x} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Let x be an individual feature value (i.e., a value of the feature in some data point)

$\min(x)$ and $\max(x)$, respectively, be the minimum and maximum values of this feature over the entire dataset

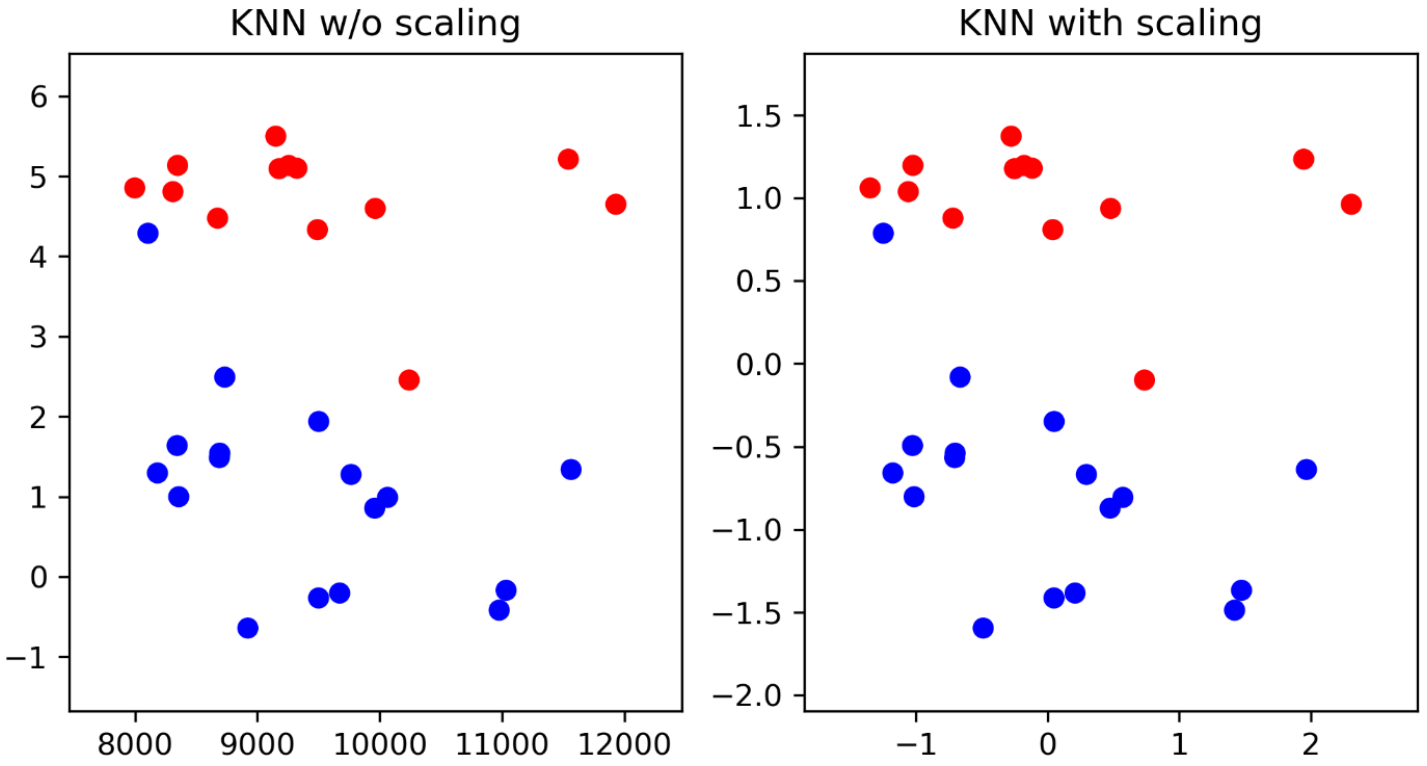
Min-max scaling squeezes (or stretches) all feature values to be within the range of [0, 1]



Feature Engineering for Machine Learning
Principles and Techniques for Data Scientists
Alice Zheng and Amanda Casari, O'Reilly, 2018

Why Scaling?

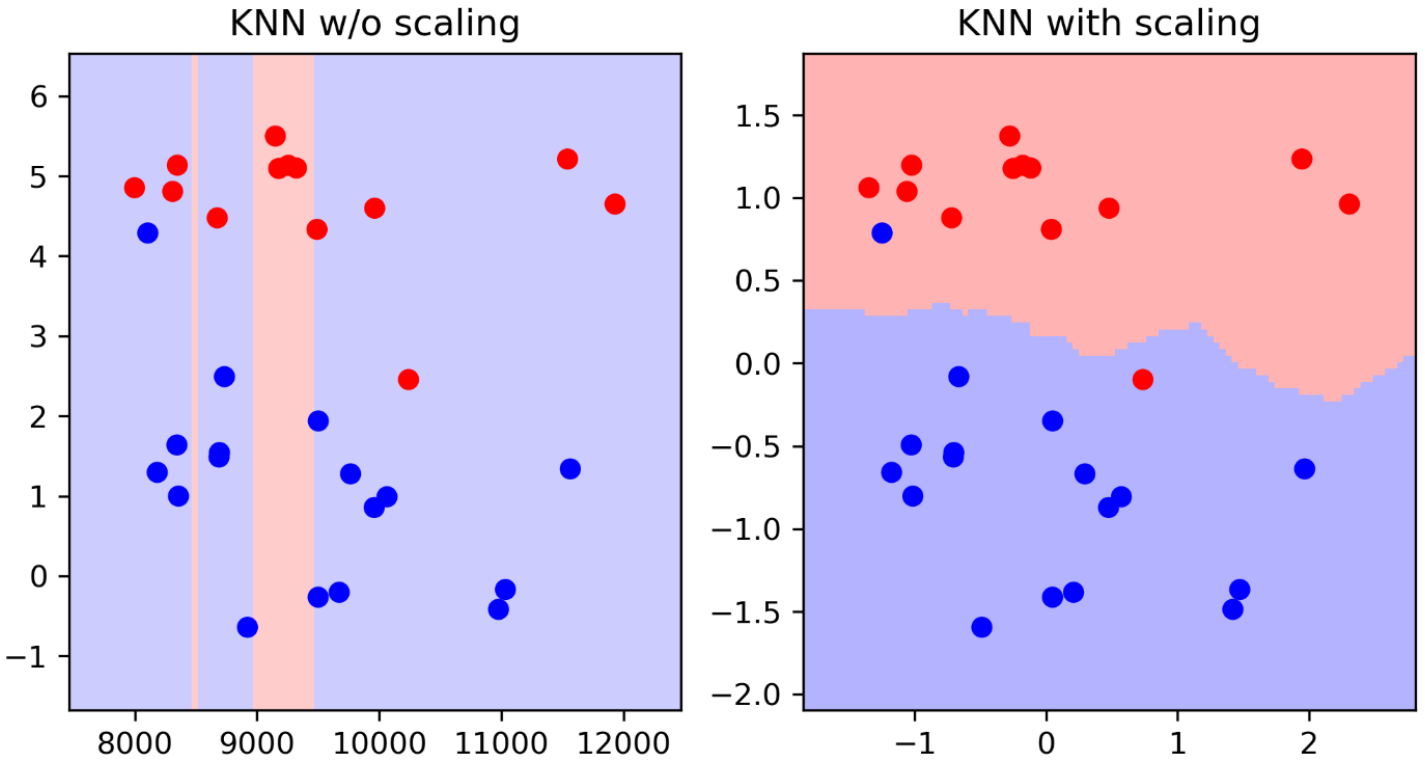
Hasso Plattner Institut



https://blog.dellemc.com/en-us/digital-transformation-just-got-easier-with-analytic-insights/

Why Scaling?

Hasso Plattner Institut



https://blog.dellemc.com/en-us/digital-transformation-just-got-easier-with-analytic-insights/

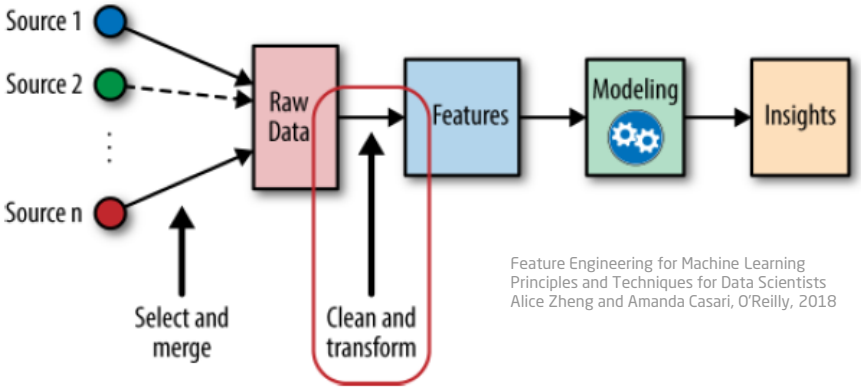
Feature Engineering

Hasso Plattner Institut

Coming up with features is difficult, time-consuming, requires expert knowledge. "Applied machine learning" is basically feature engineering. ~ Andrew Ng

The features you use influence more than everything else the result. No algorithm alone, to my knowledge, can supplement the information gain given by correct feature engineering. ~ Luca Massaron

Good data preparation and feature engineering is integral to better prediction ~ Marios Michailidis (KazAnova), Kaggle GrandMaster, Kaggle #3, former #1



Feature Engineering for Machine Learning
Principles and Techniques for Data Scientists
Alice Zheng and Amanda Casari, O'Reilly, 2018

Feature Engineering

Hasso Plattner Institut

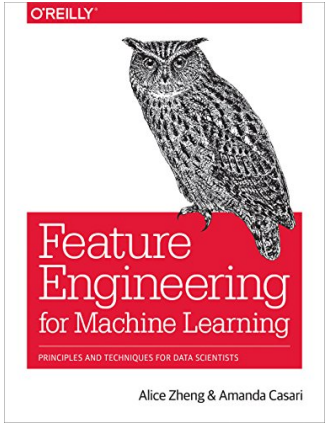
Data may be hard to understand and process

Conduct feature engineering to make reading of the data easier for our machine learning models

Feature Engineering is a process of transforming the given data into a form which is easier to interpret

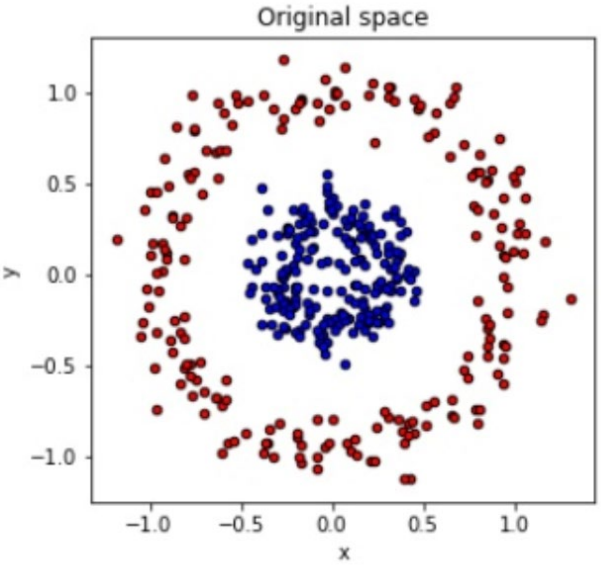
In general: Features can be generated so that the data visualization prepared for people without a data-related background can be more digestible

Different models often require different approaches for the different kinds of data



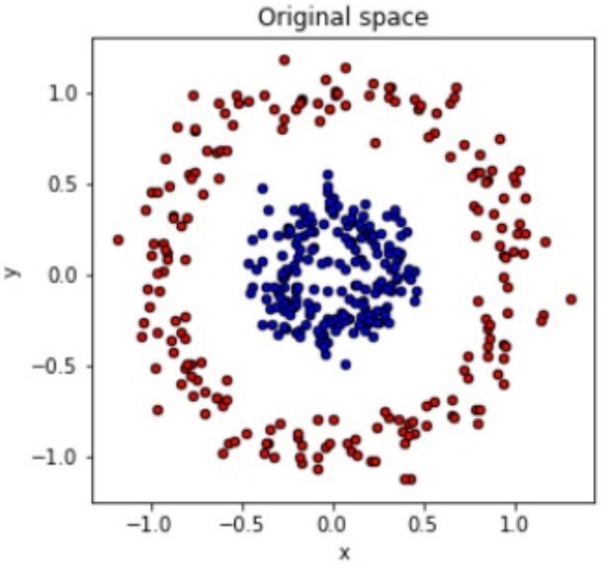
Feature Engineering for Machine Learning
Principles and Techniques for Data Scientists
Alice Zheng and Amanda Casari, O'Reilly, 2018

Feature Engineering
Example: Coordinate Transformation

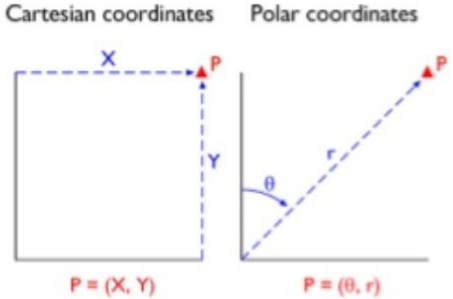


Not possible to seperate using linear classifier

Feature Engineering
Example: Coordinate Transformation

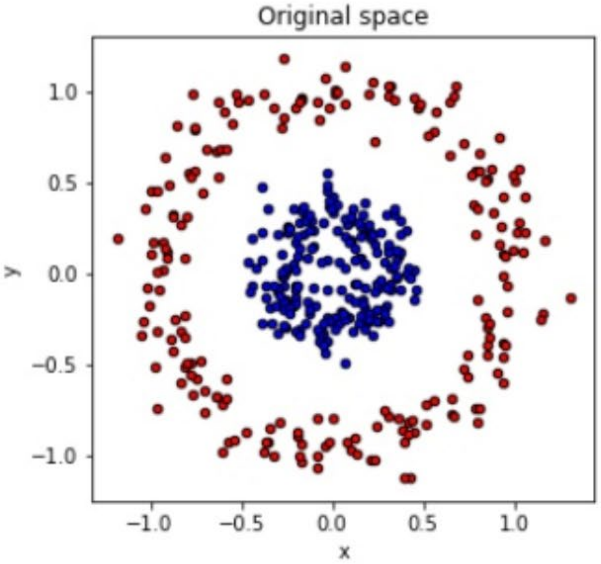


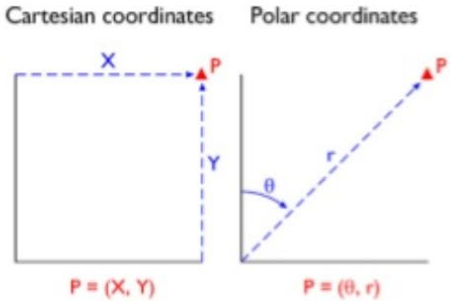
What if you use polar Coordinates instead?

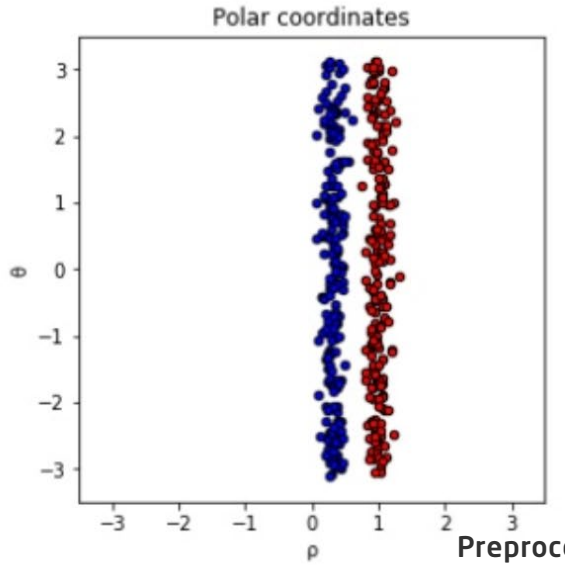


Feature Engineering
Example: Coordinate Transformation

Hasso Plattner Institut







Iterative Process of Feature Engineering

Brainstorm features: Really get into the problem, look at a lot of data, study feature engineering on other problems and see what you can steal

Devise features: Depends on your problem, but you may use automatic feature extraction, manual feature construction and mixtures of the two

Select features: Use different feature importance scorings and feature selection methods to prepare one or more “views” for your models to operate upon

Evaluate models: Estimate model accuracy on unseen data using the chosen features

Aspects of Feature Engineering

Hasso Plattner Institut

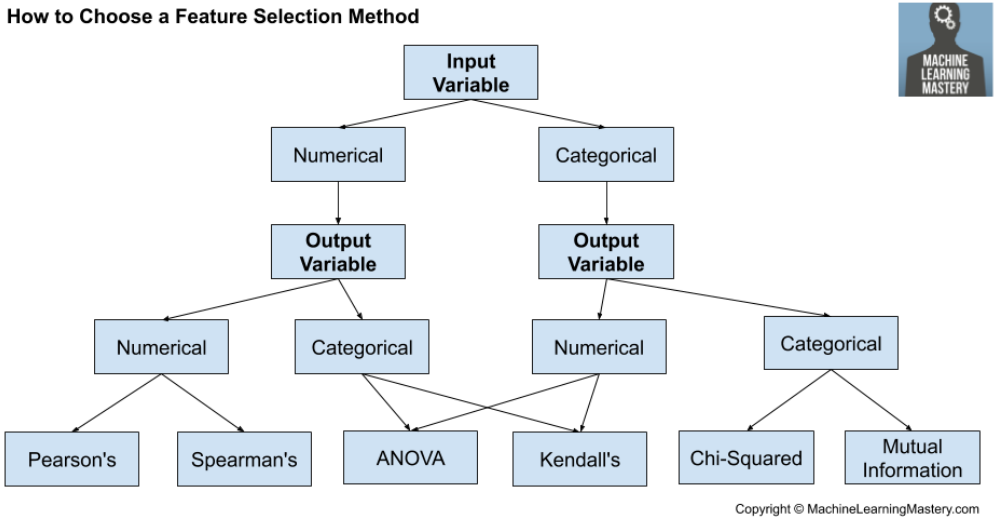
Feature Engineering	
Feature Selection	Most useful and relevant features are selected from the available data
Feature Extraction	Existing features are combined to develop more useful ones
Feature Addition	New features are created by gathering new data
Feature Filtering	Filter out irrelevant features to make the modeling step easy

Feature Selection

Hasso Plattner Institut

Process where you automatically select those features in your data that contribute most to the prediction variable or output in which you are interested

Having irrelevant features in your data can decrease the accuracy of many models, especially linear algorithms like linear and logistic regression



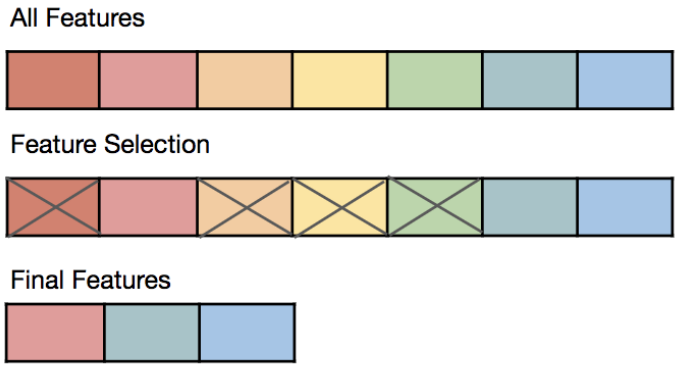
https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data

Feature Selection

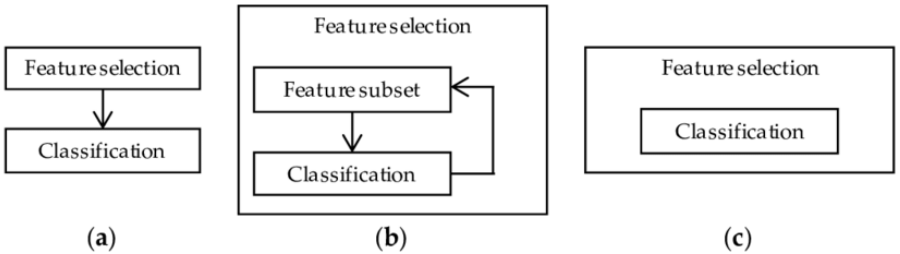
Hasso Plattner Institut

Three benefits of performing feature selection before modeling your data are:

- **Reduces Overfitting**: Less redundant data means less opportunity to make decisions based on noise
- **Improves Accuracy**: Less misleading data means modeling accuracy improves
- **Reduces Training Time**: Less data means that algorithms train faster



https://quantdare.com/what-is-the-difference-between-feature-extraction-and-feature-selection/



https://towardsdatascience.com/feature-selection-techniques-1bfab5fe0784

Feature Extraction

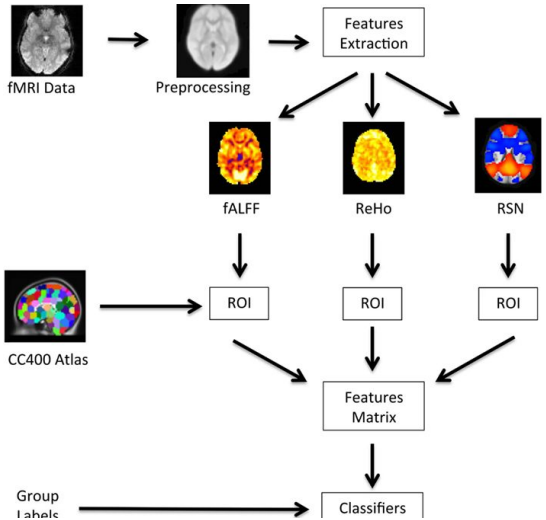
Hasso Plattner Institut

Aims to reduce the number of features in a dataset by creating new features from the existing ones (and then discarding the original features)

New reduced set of features should then be able to summarize most of the information contained in the original set

Create some interaction (e.g., multiply or divide) between each pair of variables → lengthy process

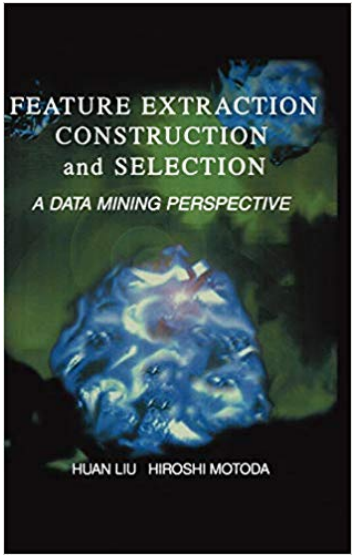
Deep feature synthesis (DFS) is an algorithm which enables you to quickly create new variables with varying depth

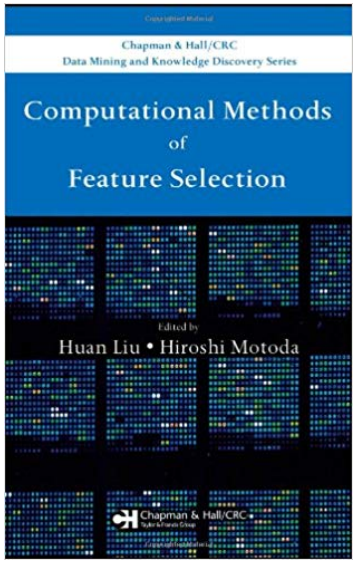


https://matlab1.com/feature-extraction-image-processing/

To Know More

Hasso Plattner Institut





Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

To Know More

Here are some generally relevant papers:

- JMLR Special Issue on Variable and Feature Selection

Here are some generally relevant and interesting slides:

- Feature Engineering (PDF), Knowledge Discover and Data Mining 1, by Roman Kern, Knowledge Technologies Institute
- Feature Engineering and Selection (PDF), CS 294: Practical Machine Learning, Berkeley
- Feature Engineering Studio, Course Lecture Slides and Materials, Columbia
- Feature Engineering (PDF), Leon Bottou, Princeton

And a video for some good practical tips:

- Feature Engineering

Time Series
Let's Compare ECG Signals

Preprocessing and Feature Engineering

Data Management for Digital Health, Winter 2019

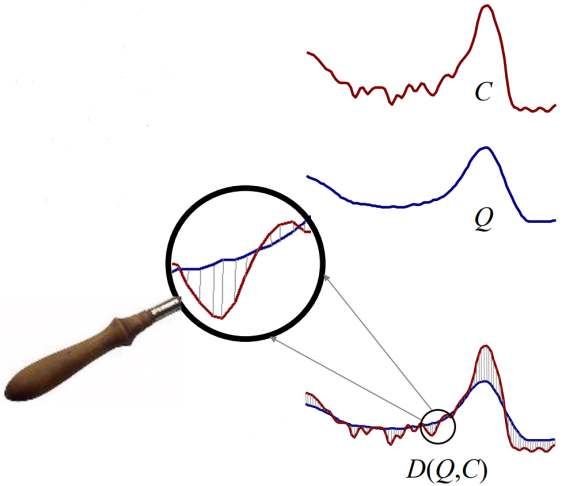
Euclidean Distance Metric
Comparing to Time Series

Let's assume we want to compare two time series

Given two time series:

$$Q = q_1 \ldots q_n$$
$$C = c_1 \ldots c_n$$

$$D(Q, C) \equiv \sqrt{\sum_{i=1}^{n} (q_i - c_i)^2}$$



https://en.wikipedia.org/wiki/Professor_Frink

About 80% of published work in data mining uses Euclidean distance

http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf

Data Preparation Time Series

If we naively try to measure the distance between two "raw" time series, we may get very unintuitive results

https://en.wikipedia.org/wiki/Professor_Frink

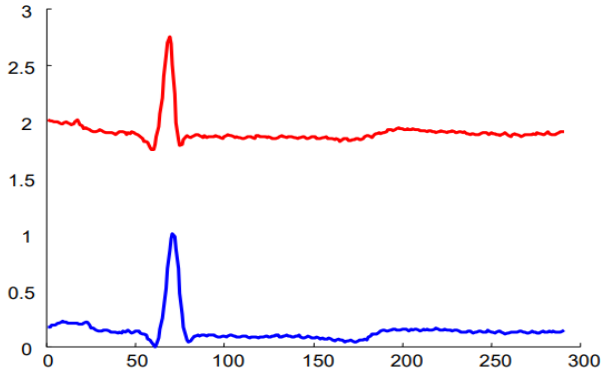
4 most common distortions

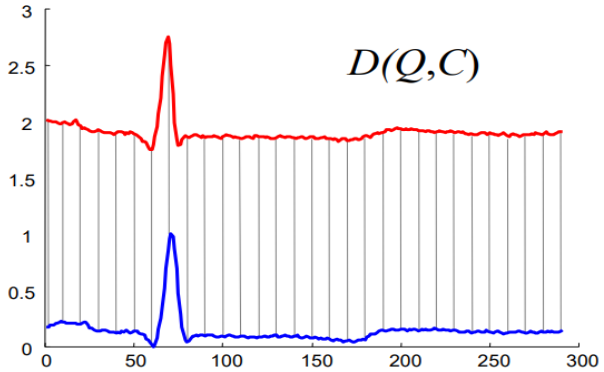
- Offset Translation
- Amplitude Scaling
- Linear Trends
- Noise

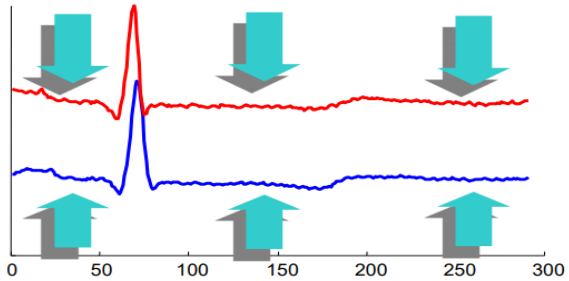
Euclidean distance is very sensitive to some "distortions" in the data. For most problems these distortions are not meaningful → should remove them

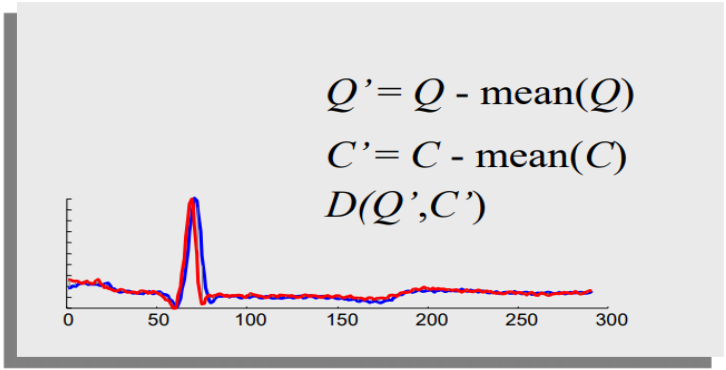
https://en.wikipedia.org/wiki/Dr._Nick

Preprocessing the Data
Offset Translation



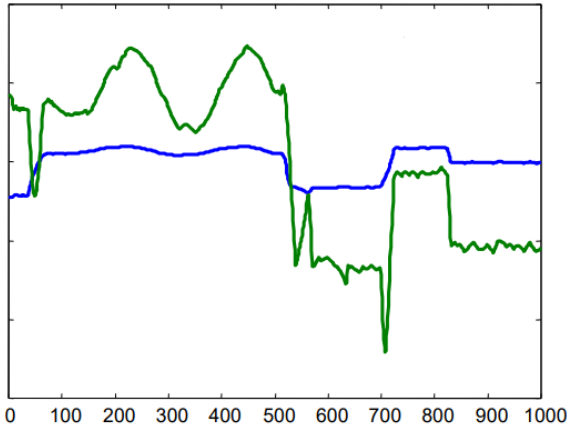


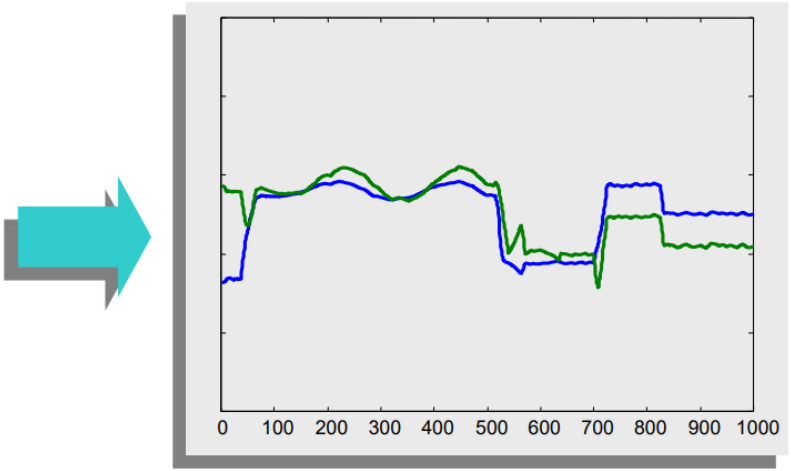




http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf

Preprocessing the Data
Amplitude Scaling





Zero-mean

Unit-variance

Widely used for normalization in many machine learning algorithms

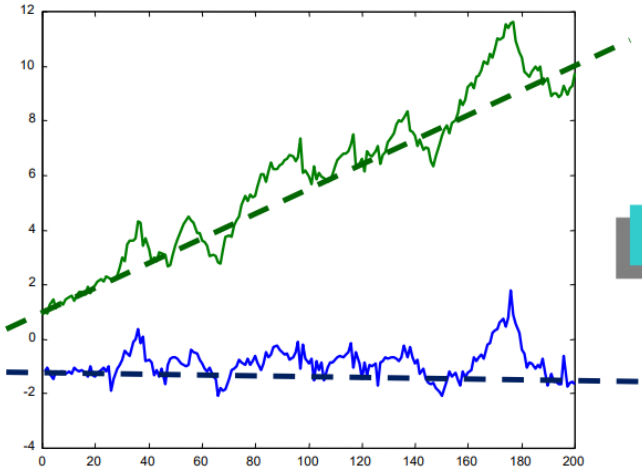
$$Q'' = (Q - \text{mean}(Q)) \,/\, \mathbf{std}(\boldsymbol{Q})$$

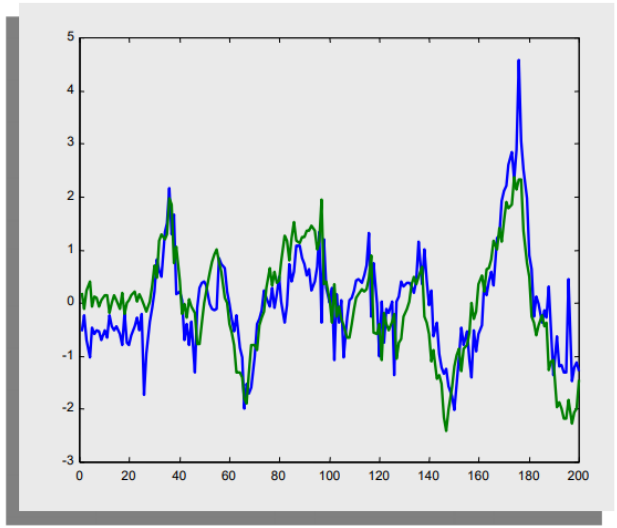
$$C'' = (C - \text{mean}(C)) \,/\, \mathbf{std}(\boldsymbol{C})$$

$$D(Q'', C'')$$

http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf

Preprocessing the Data
Offset Translation





Removing linear trend:

- Fit the best fitting straight line to the time series, then
- subtract that line from the time

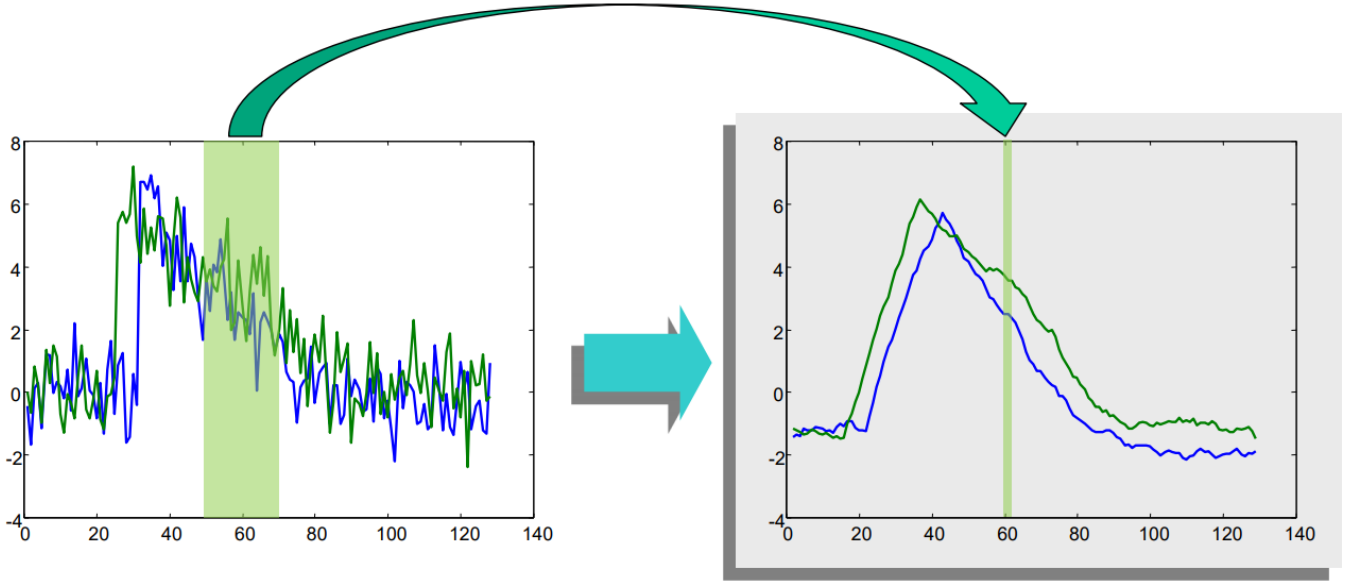
Remove linear trend

Removed offset translation

Removed amplitude scaling

http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf

Preprocessing the Data
Noise



The intuition behind removing noise is ...

Average each data points value with its neighbors

$$Q' = \text{smooth}(Q)$$

$$C' = \text{smooth}(C)$$

$$D(Q', C')$$

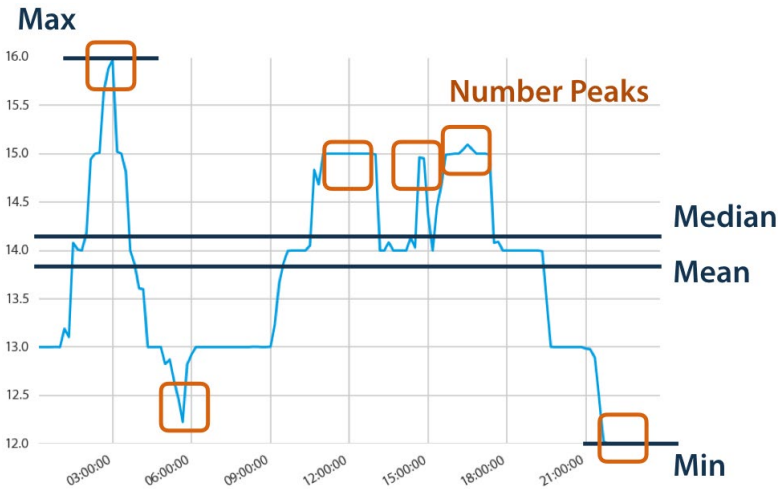
http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf

Feature Engineering for Time Series

Date Time Features: These are components of the time step itself for each observation

Lag Features: These are values at prior time steps

Window Features: These are a summary of values over a fixed window of prior time steps



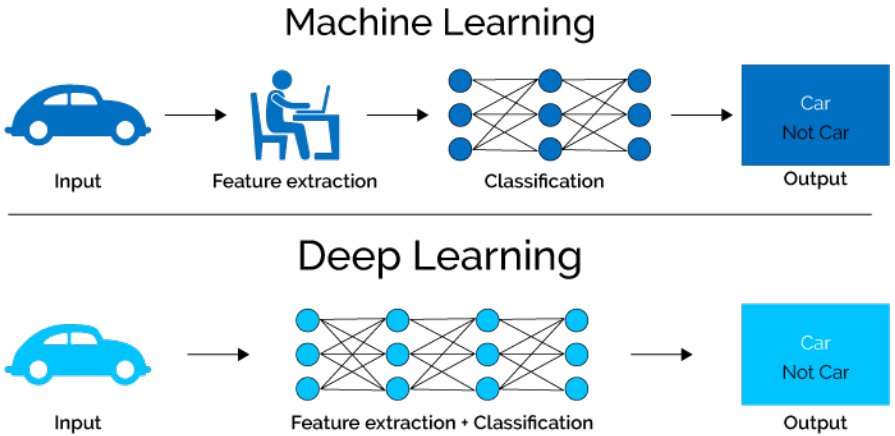
https://tsfresh.readthedocs.io/en/latest/text/introduction.html

Automated Feature Engineering
Why Do It?

Hasso Plattner Institut

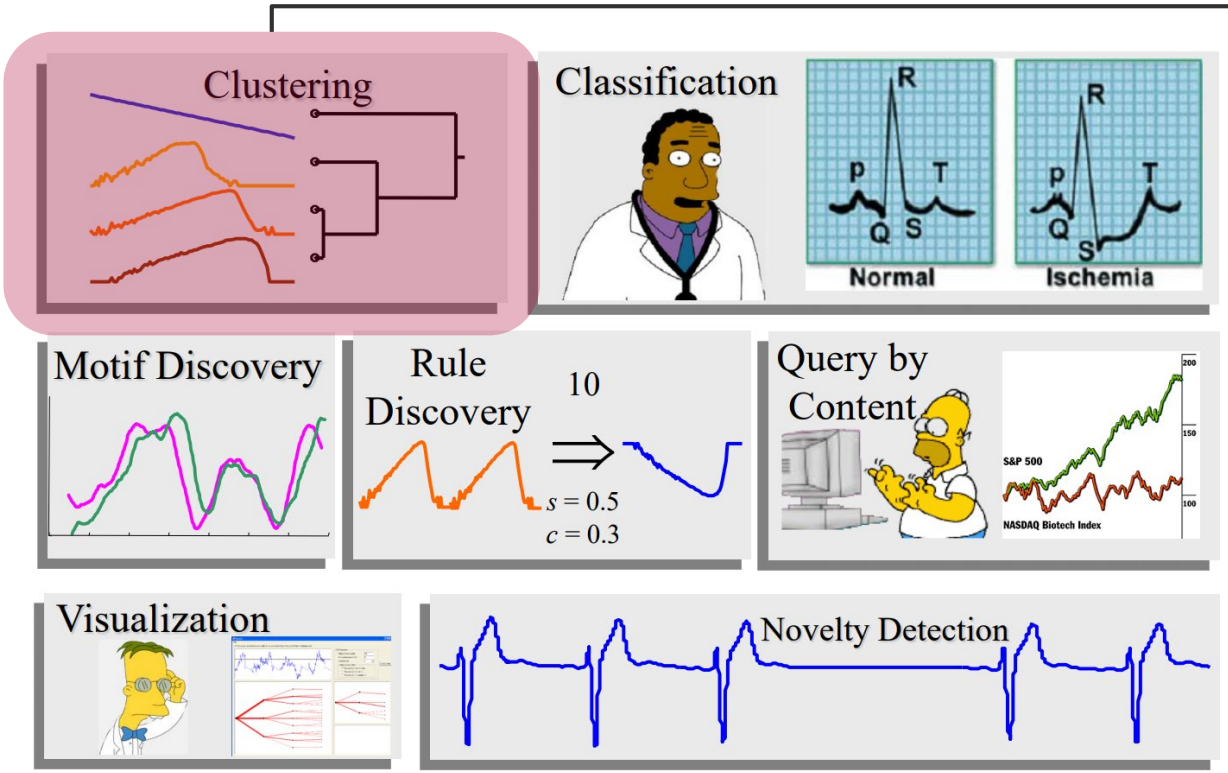
We're interested in *features*—we want to know which are relevant. If we fit a model, it should be **interpretable**

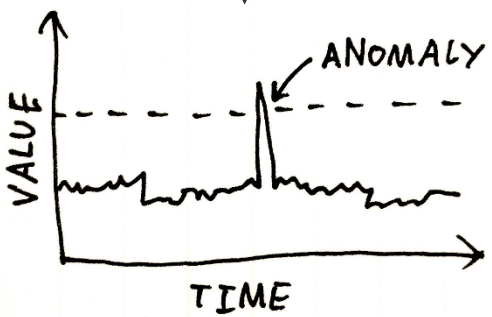
- What causes lung cancer?
  - Features are aspects of a patient's medical history
  - Binary response variable: did the patient develop lung cancer?
  - Which features best predict whether lung cancer will develop? Might want to legislate against these features.



https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063

What Next?





http://didawiki.cli.di.unipi.it/lib/exe/fetch.php/dm/time_series_2017.pdf
http://amid.fish/anomaly-detection-with-k-means-clustering

What to Take Home?

Data preparation allows simplification of data to make it ready for Machine Learning and involves data selection, preprocessing, and transformation

Step 1: Data Selection Consider what data is available, what data is missing and what data can be removed

Step 2: Data Preprocessing Organize your selected data by formatting, cleaning and sampling from it

Step 3: Data Transformation Transform preprocessed data ready for machine learning by engineering features using scaling, attribute decomposition and attribute aggregation