Hochschule Karlsruhe
University of
Applied Sciences

Fakultät für
Elektro- und
Informationstechnik

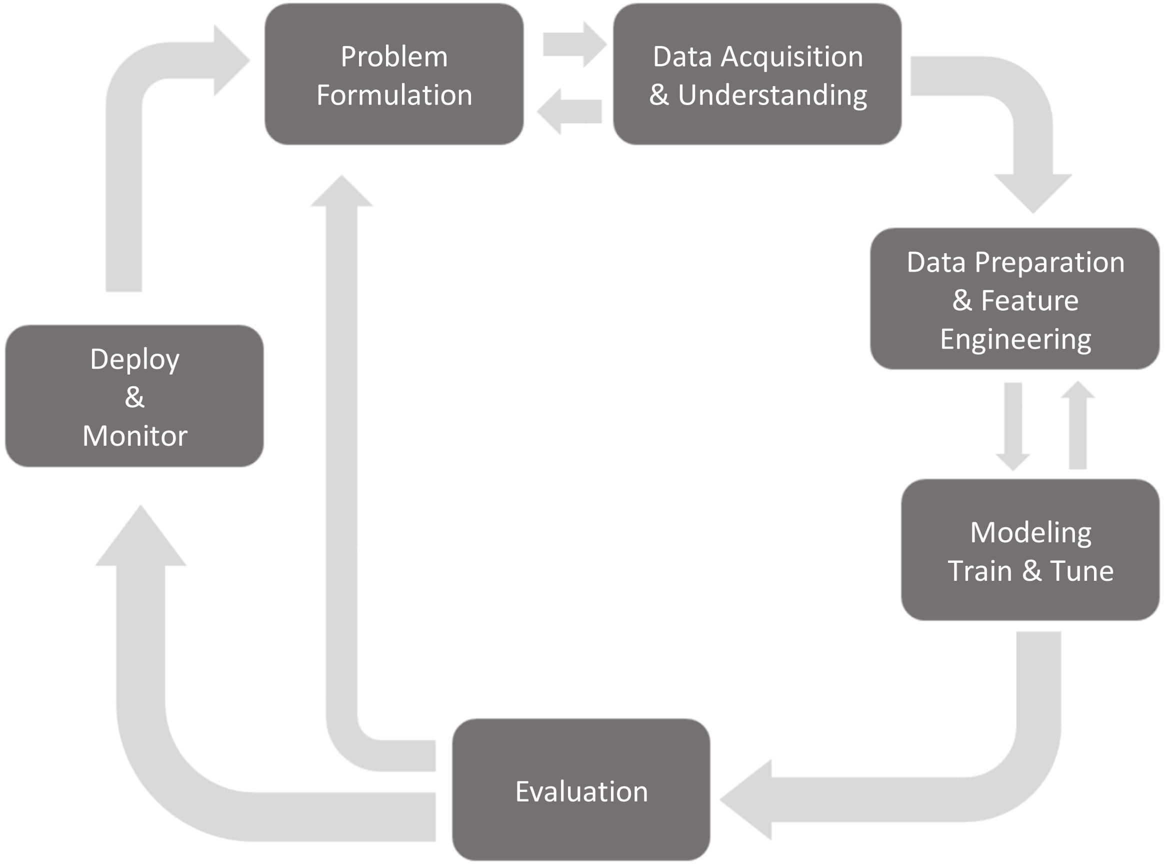
Data Preparation And Feature Engineering (2)

Source: DALL.E

Kawther Aboalam

ML Project Workflow - A Typical Procedure For A Machine Learning Project

- Projects start with understanding of the business case :
  - What? What for? Where? For whom?
- Data Acquisition, Data Understanding
- Data Preparation, Feature Engineering
- Designing a machine learning model:
  - train, validate, tune, validate , retune…retrain…..
- Model selection, model evaluation
- Evaluation:
  - Does the model meet the requirements?
  - Is the business case confirmed?
- Implementation of the model, model monitoring

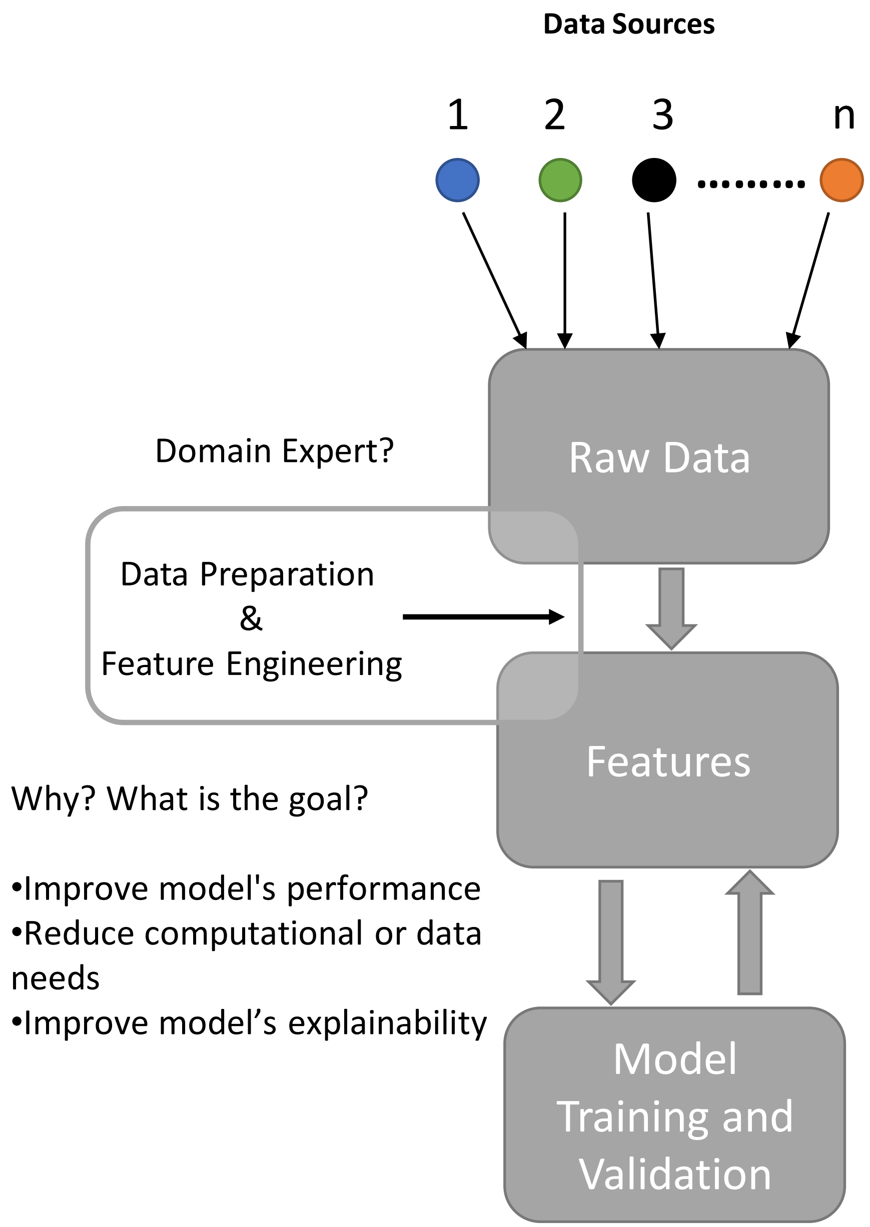

Problem Formulation
Data Acquisition & Understanding
Data Preparation & Feature Engineering
Modeling Train & Tune
Evaluation
Deploy & Monitor

Modified by me: Source: successfactory management coaching gmbh

Data Preparation and Feature Engineering

Data Flow in Machine Learning

- Machine learning projects use data generated from different sources, they often have non-uniform formats and structures, requiring preparation before use in machine learning processes.
- Steps in data preparation:
  - Data exploration
  - Quality assessment/Data cleaning
  - Imputation
  - Feature engineering: (creating the relevant features)
    - Features Selection for ML model
    - Combination, transformation, create a new feature
    - Features encoding
  - After building the models, testing if the selected features achieve the desired outcomes
  - Repeating the preparation process if necessary



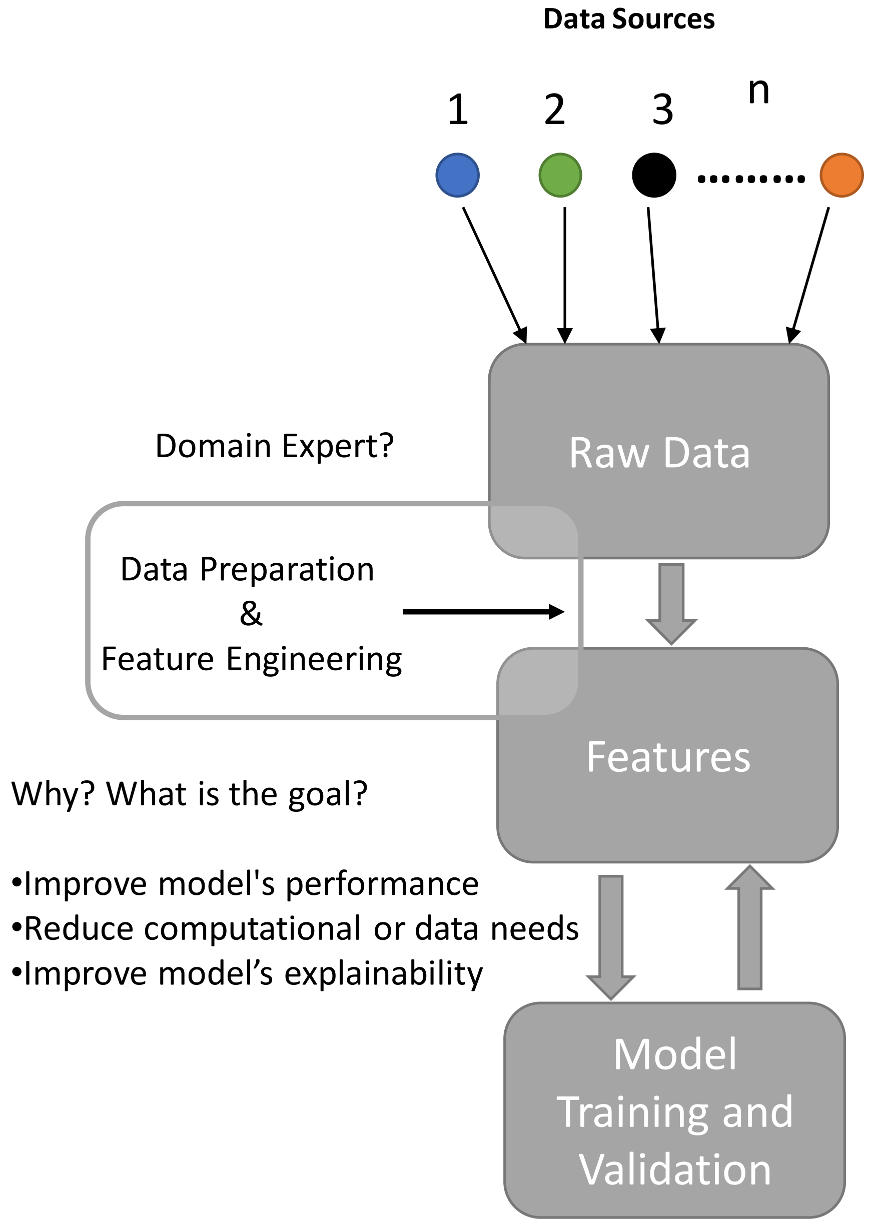
Why? What is the goal?

- Improve model's performance
- Reduce computational or data needs
- Improve model’s explainability

Data Preparation and Feature Engineering

Data Flow in Machine Learning

- Machine learning projects use data generated from different sources, they often have non-uniform formats and structures, requiring preparation before use in machine learning processes.
- Steps in data preparation:
  - Data exploration
  - Quality assessment/Data cleaning
  - Imputation
  - Feature engineering: (creating the relevant features)
    - Features Selection for ML model
    - Combination, transformation, create a new feature
    - Features encoding
  - After building the models, testing if the selected features achieve the desired outcomes
  - Repeating the preparation process if necessary



Why? What is the goal?

- Improve model's performance
- Reduce computational or data needs
- Improve model’s explainability

Data Preparation and Feature Engineering

Missing Values Imputaion

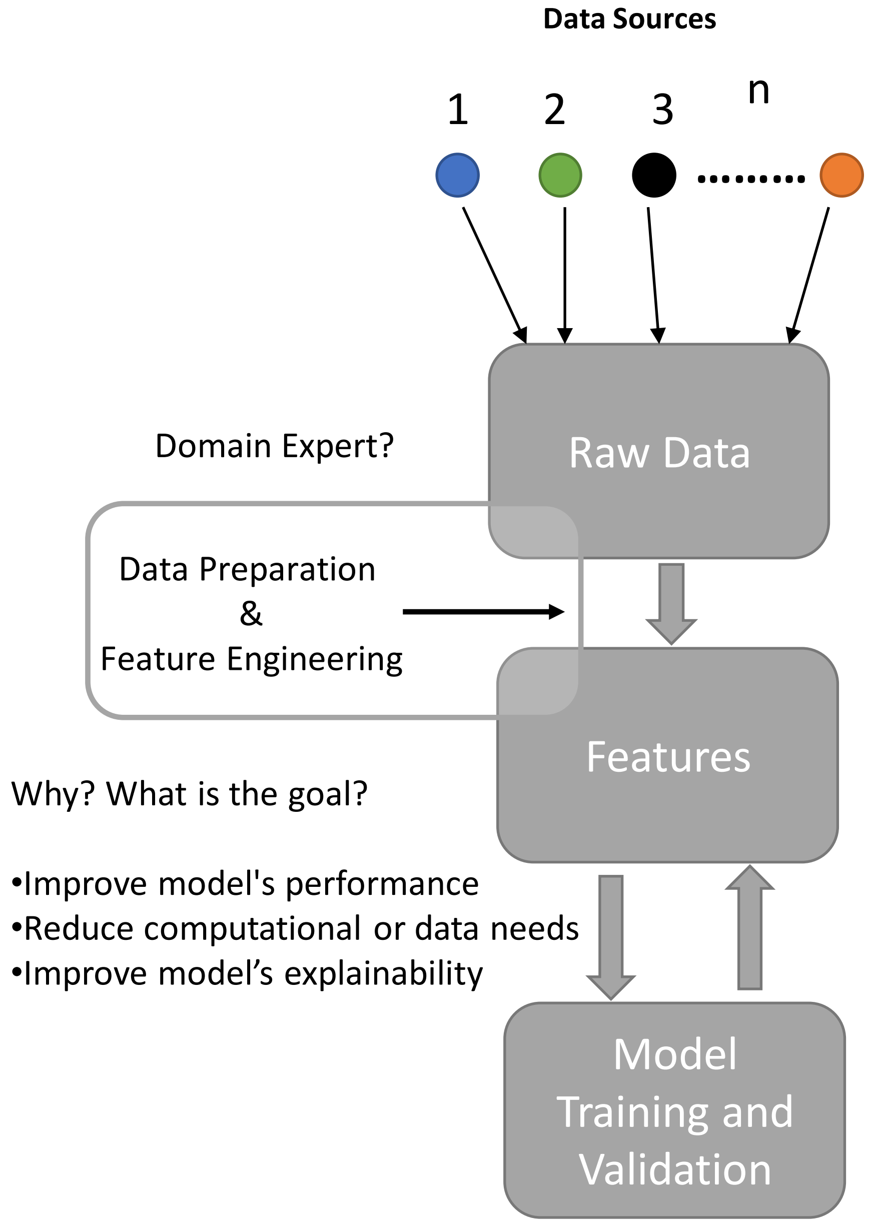
- In Early Project Stage
  - Missing quantitative data can be replaced using simple methods such as a fixed value, the mean of the data series, or zero. For ordered sequences or time series, the missing value can be imputed using the predecessor or successor value.
  - For qualitative (categorical) data, missing entries can be replaced with a designated "missing" category or the most frequently occurring value in the dataset.
  - It's important to note that imputing missing values alters the dataset and can significantly influence the results of subsequent analyses—especially when dealing with many missing entries or when those entries are located near each other in the sequence.

- In Later Project Stage
  - In later stages, as the dataset becomes better understood, more sophisticated imputation methods can be applied based on detected relationships between variables.
  - K-Nearest Neighbors (KNN) Imputation: Estimates missing values by calculating the mean or median of the k most similar data points.
  - Multivariate Imputation: Uses regression or other statistical models to predict missing values based on other features in the dataset.

Data Preparation and Feature Engineering

Data Flow in Machine Learning

- Machine learning projects use data generated from different sources, they often have non-uniform formats and structures, requiring preparation before use in machine learning processes.
- Steps in data preparation:
  - Data exploration
  - Quality assessment/Data cleaning
  - Imputation
  - Feature engineering: (creating the relevant features)
    - Features Selection for ML model
    - Combination, transformation, create a new feature
    - Features encoding
  - After building the models, testing if the selected features achieve the desired outcomes
  - Repeating the preparation process if necessary



Data Sources

Domain Expert?

Data Preparation & Feature Engineering

Why? What is the goal?

- Improve model's performance
- Reduce computational or data needs
- Improve model’s explainability

Data Preparation and Feature Engineering

Feature Engineering – Feature Selection

- Why?
  - Speeds up training by reducing the number of input variables
  - Reduces model complexity, making results easier to interpret
  - Improves model accuracy, when the most relevant features are chosen
  - Decreases the risk of overfitting by eliminating irrelevant or redundant features

- Feature Selection Approaches:
  - Filter Approach: Selecting features based on statistical measures of relevance to the target variable (e.g., correlation)
  - Wrapper Approach: Selecting a subset of features, training a model with them, and then adjusting the subset based on the model's performance, either by adding or removing features.
  - Embedded Approach: Feature selection is built into the model training process. An Example is **Lasso Regression.**

Data Preparation and Feature Engineering

Feature Engineering - Combination, transformation, create a new feature

+ Features can contribute more directly to a model’s prediction through functional mappings, interactions, or logical linkages between variables.
+ When predicting process behavior, especially in technical or physical systems, linear models are often used. These rely on a linear combination of input features to estimate the outcome.
+ However, many real-world systems exhibit nonlinear behavior, where linear combinations are insufficient. In such cases, feature engineering is necessary to introduce:
  - Feature interactions (combinations)

    In predictive maintenance for rotating machinery, combining features like vibration amplitude and temperature reveals critical stress conditions that may not be detectable when analyzing each feature separately.
  - Nonlinear transformations

    To predict vehicle behavior under braking, applying a nonlinear transformation like speed^2 captures the quadratic relationship in braking distance, based on the formula $d = \frac{v^2}{2a}$
  - Derived features that reflect underlying system dynamics

    In predicting cardiovascular risk, pulse pressure—derived from the raw features systolic and diastolic blood pressure by subtraction—reflects the force per heartbeat and offers important insight into arterial health.

Data Preparation and Feature Engineering

Feature Engineering - Combination, transformation, create a new feature

- Categorical data is typically encoded using binary values (0 and 1).
- Logical links between features (e.g., AND/OR conditions) can be constructed using simple operations, but linear models cannot inherently capture these logical interactions.
- If such relationships are important, they should be explicitly included as new features.
- Linking features is especially useful when domain knowledge suggests meaningful logical or functional connections.

x_1	x_2	x_1 and x_2	x_1 or x_2
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

Data Preparation and Feature Engineering

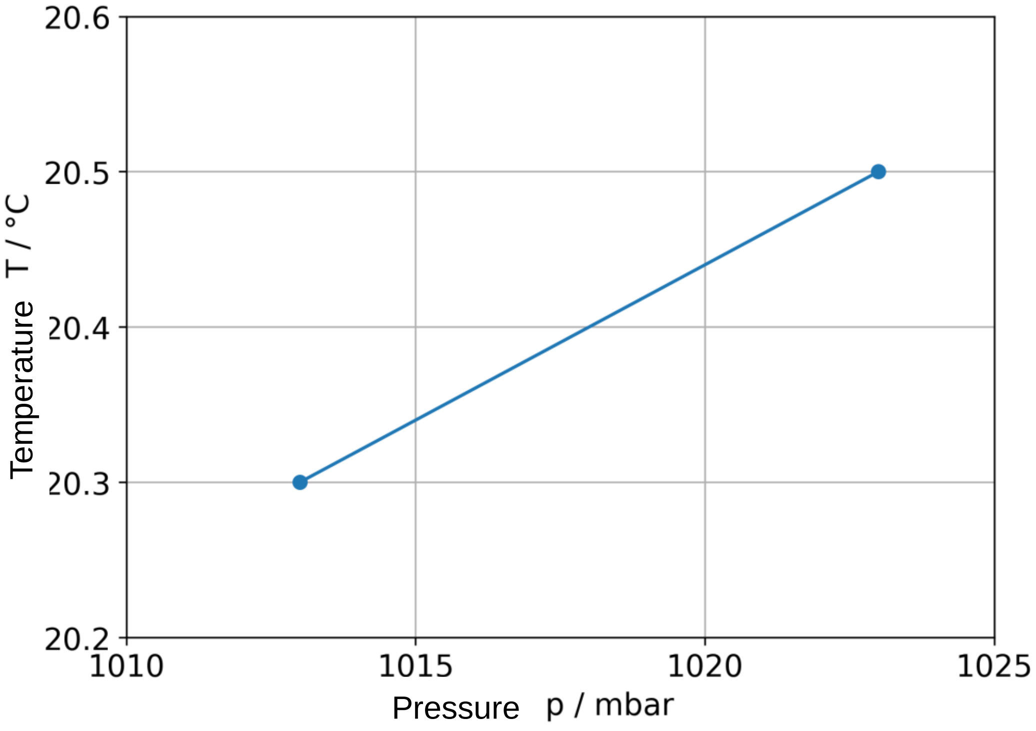
Features Encoding – Numerical Features

- Numerical values like 1013 and 1.013 can both indicate the barometric pressure, once in mbar and once in bar, unit can be freely selected.
- Distance of two multidimensional samples, however, depends on the unit of each dimension
  - Pressure in mbar and the temperature in degrees Celsius

$$D_1 = \sqrt{10^2 + 0.2^2} = 10.002$$

  - Pressure in bar and the temperature in degrees Celsius

$$D_2 = \sqrt{0.01^2 + 0.2^2} = 0.2002$$



Data Preparation and Feature Engineering

Features Encoding – Numerical Features

- Distance D_2 has a smaller amount and is significantly characterized by the temperature difference
- To give all quantities a comparable meaning, quantitative quantities are standardized or normalized (scaling)
  - Data Standardization: It scales the data such that the mean $(\bar{x})$ is 0 and the standard deviation (s) is 1

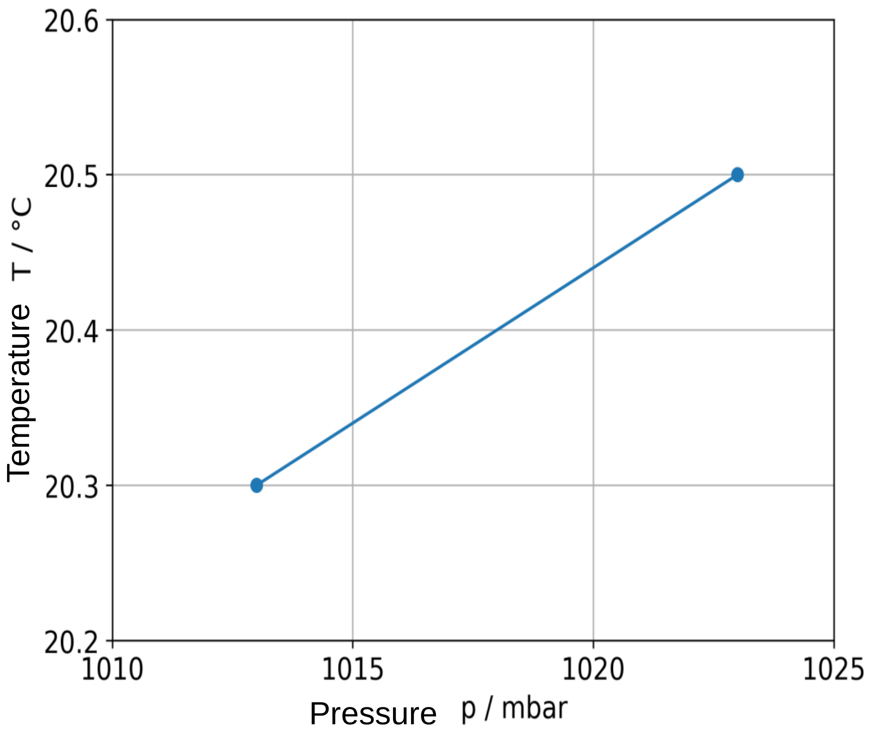
$$x_S = \frac{x - \bar{x}}{s}$$

  - Data Min-Max Normalization: It scales the data with a known range [a,b]

$$x_N = \frac{(x - x_{min})(b - a)}{x_{max} - x_{min}} + a$$

X_{min} and X_{max} are the minimum and maximum values of X, respectively

For a range of [0,1]: $$x_N = \frac{(x - x_{min})}{x_{max} - x_{min}}$$

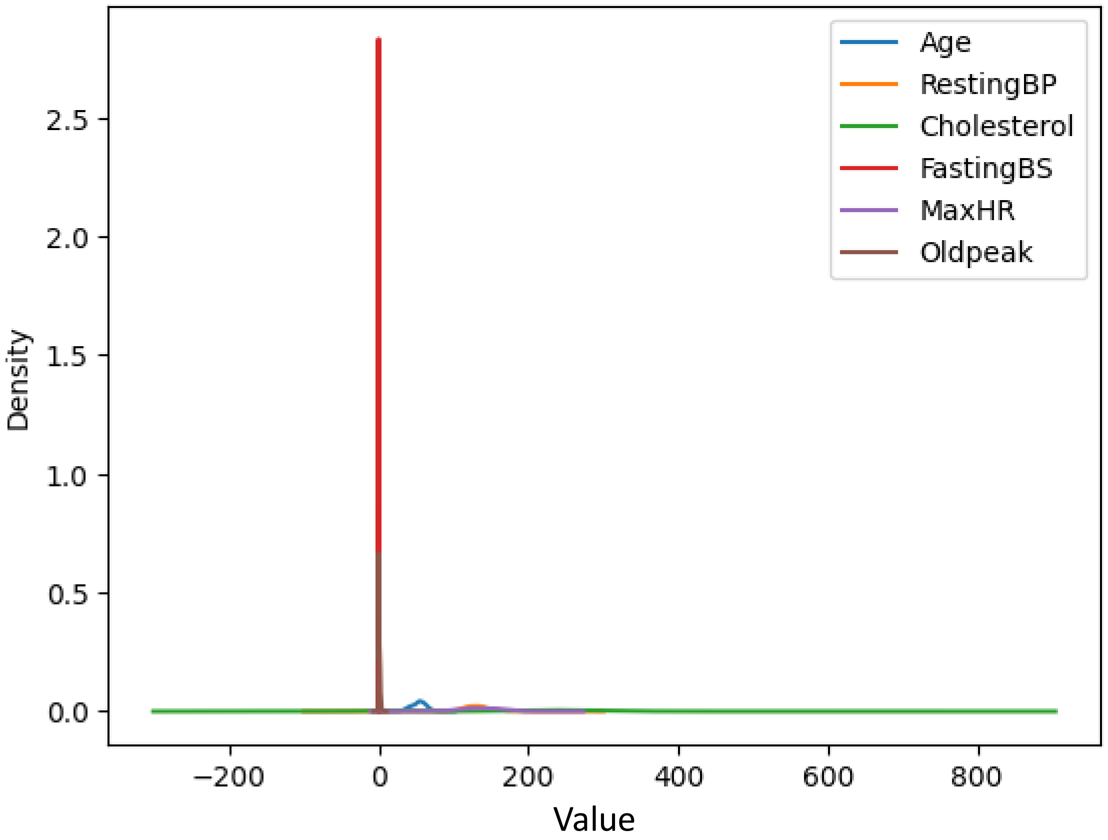


Data Preparation and Feature Engineering

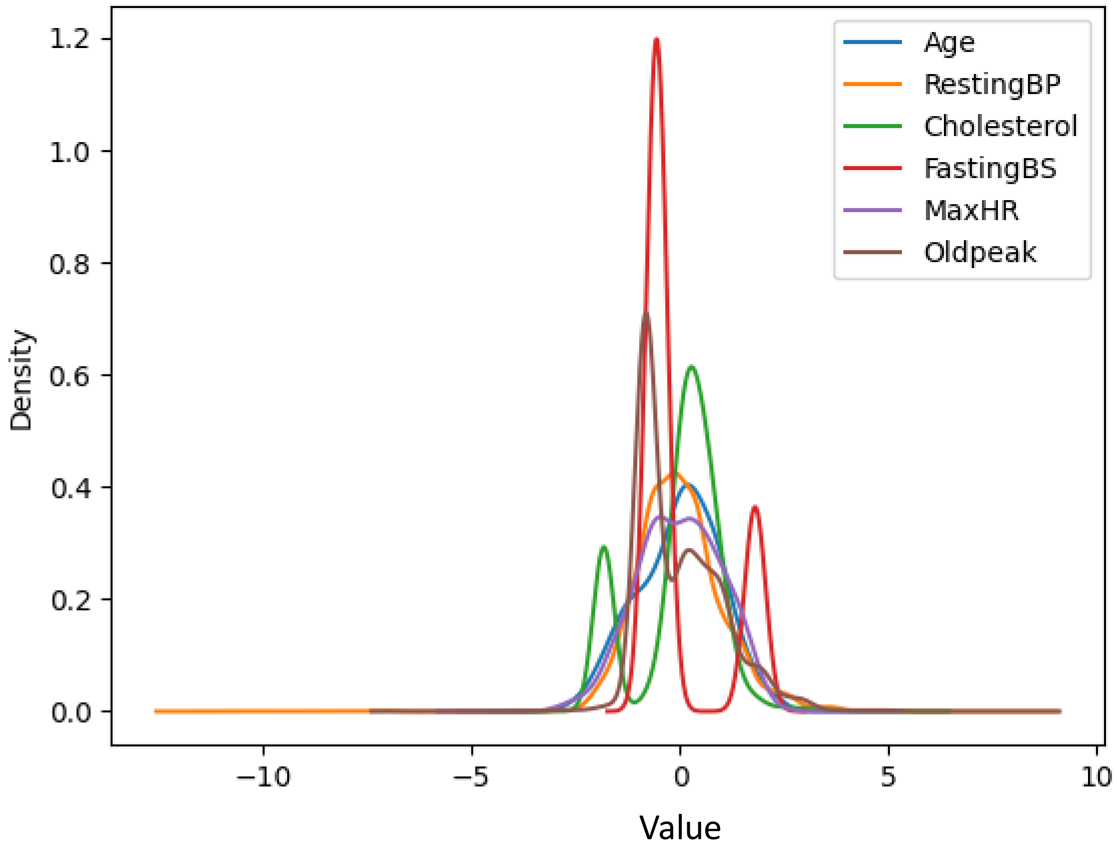
Features Encoding – Numerical Features

Numerical features of Heart Failure Prediction Dataset

Before scaling



After scaling (Standarization)

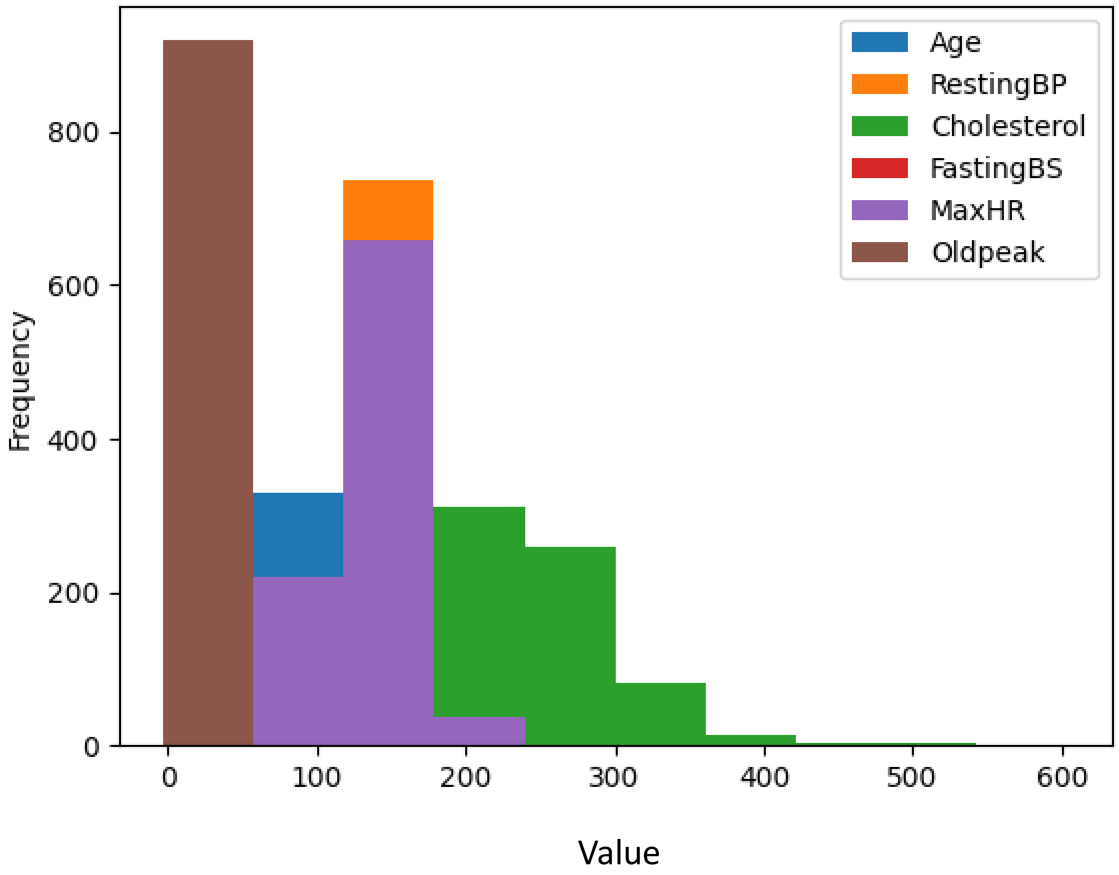


Data Preparation and Feature Engineering

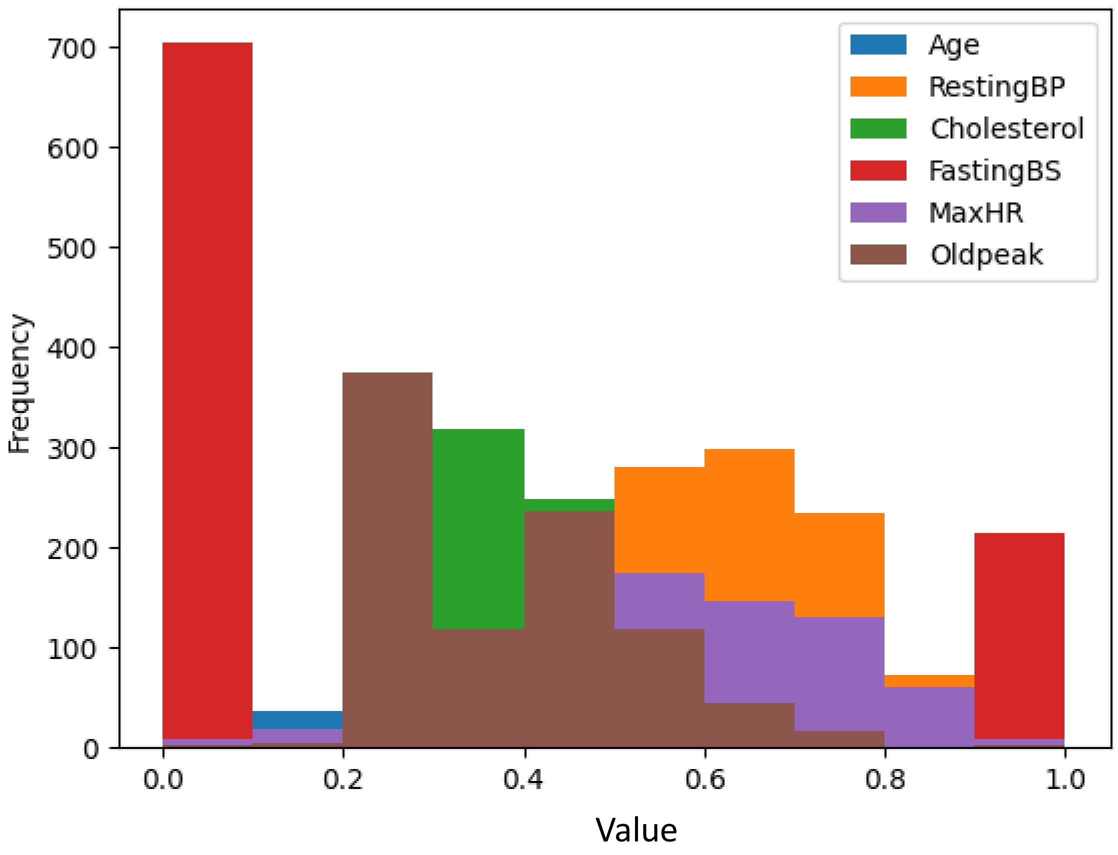
Features Encoding – Numerical Features

Numerical features of Heart Failure Prediction Dataset

Before scaling



After scaling (Min-Max Normalization)



Data Preparation and Feature Engineering

Features Encoding – Categorical Features

- One-Hot Encoding creates a new binary column for each category. Each column has a value of 1 if the observation belongs to that category, and 0 otherwise.
- This can significantly increase the number of columns in the dataset, especially with categorical features containing many levels. Some columns may contain very few non-zero entries, making them sparse and less informative.
- The resulting columns are often redundant—for example, if there are three supplier categories (A, B, and C), knowing the values of two allows inference of the third.

$$A = 1 - B - C$$

- To avoid redundancy and multicollinearity, one column is typically removed during one-hot encoding (a technique known as drop-first)

Part	Feature	Coding supplier		
		A	B	C
1	Supplier A	1	0	0
2	Supplier B	0	1	0
3	Supplier C	0	0	1
4	Supplier B	0	1	0
5	Supplier C	0	0	1

Data Preparation and Feature Engineering

Features Encoding – Categorical Features

- With Ordinal Encoding, unique numbers are assigned to all categories
  - Grouping features have no natural order
  - Ordinal features have a natural order, which is also taken into account during coding
- Quality evaluation as an example of ordinal encoding
- Ordinal encoding does not change the number of columns, data set remains clear

Quality Evaluation	Ordinal coding
very good	1
good	2
bad	3
Very bad	4

Data Preparation and Feature Engineering

Features Encoding – Comparison for Categorical Features

- Encoding has an influence on the distance of two qualitatively described records
- Distance of a very fast speed and a slow speed is calculated from the difference of the ordinal coding

$$D_{O13} = 3 - 1 = 2$$

- Distance of a fast speed and a slow speed is calculated from the difference of the ordinal coding

$$D_{O23} = 3 - 2 = 1$$

Where O refers to ordinal

Speed	Ordinal Encoding
very fast	1
fast	2
slow	3
Very slow	4

Data Preparation and Feature Engineering

Features Encoding – Comparison for Categorical Features

- In One Hot Encoding the distance is calculated with the feature vectors, for distance of a very fast and a slow speed results in

$$D_{H13} = \sqrt{\Delta x_1^2 + \Delta x_2^2 + \Delta x_3^2 + \Delta x_4^2} = \sqrt{1+0+1+0} = \sqrt{2}$$

- And for distance of a fast and a slow speed results in

$$D_{H23} = \sqrt{\Delta x_1^2 + \Delta x_2^2 + \Delta x_3^2 + \Delta x_4^2} = \sqrt{0+1+1+0} = \sqrt{2}$$

- Distance dimensions have not changed, no statement about processing speed possible
- One Hot Encoding is especially useful for grouping features, while Ordinal Encoding is useful for ordinal features

Feature	One Hot Encoding			
	very fast	fast	slow	Very slow
very fast	1	0	0	0
fast	0	1	0	0
slow	0	0	1	0
Very slow	0	0	0	1

Hochschule Karlsruhe
University of
Applied Sciences

Fakultät für
Elektro- und
Informationstechnik

www.h-ka.de