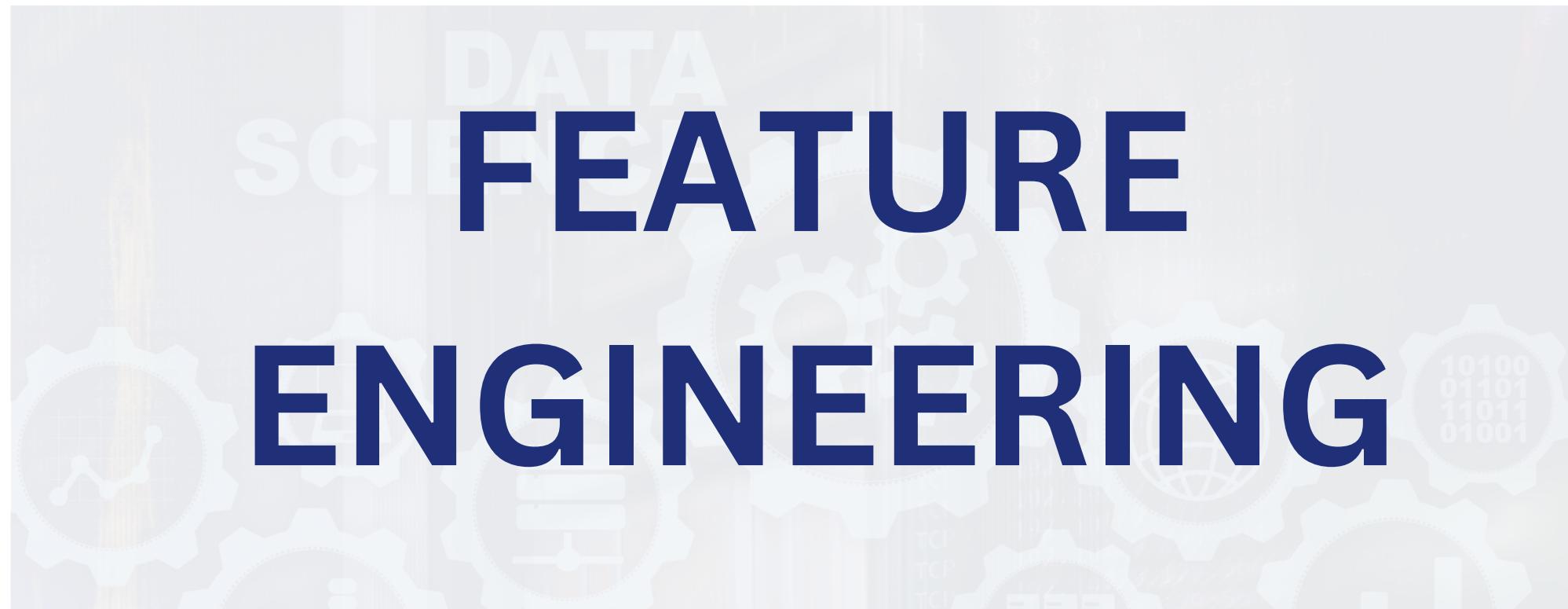
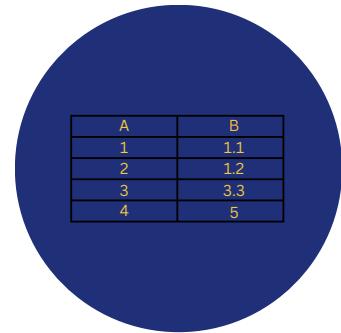


FEATURE ENGINEERING



Feature Engineering



In this section, we will cover **Feature Engineering**, reviewing essential concepts from statistics as well as key aspects of Feature Engineering

TOPICS WE'LL COVER:

Types of Variable

Feature Encoding

Feature Scaling

Cross Validation

GOALS FOR THIS SECTION:

- Review the different data types in statistics
- Discuss feature transformation techniques
- Explore feature extraction methods
- Examine approaches to feature selection

TYPES OF VARIABLES

Variables

Numerical

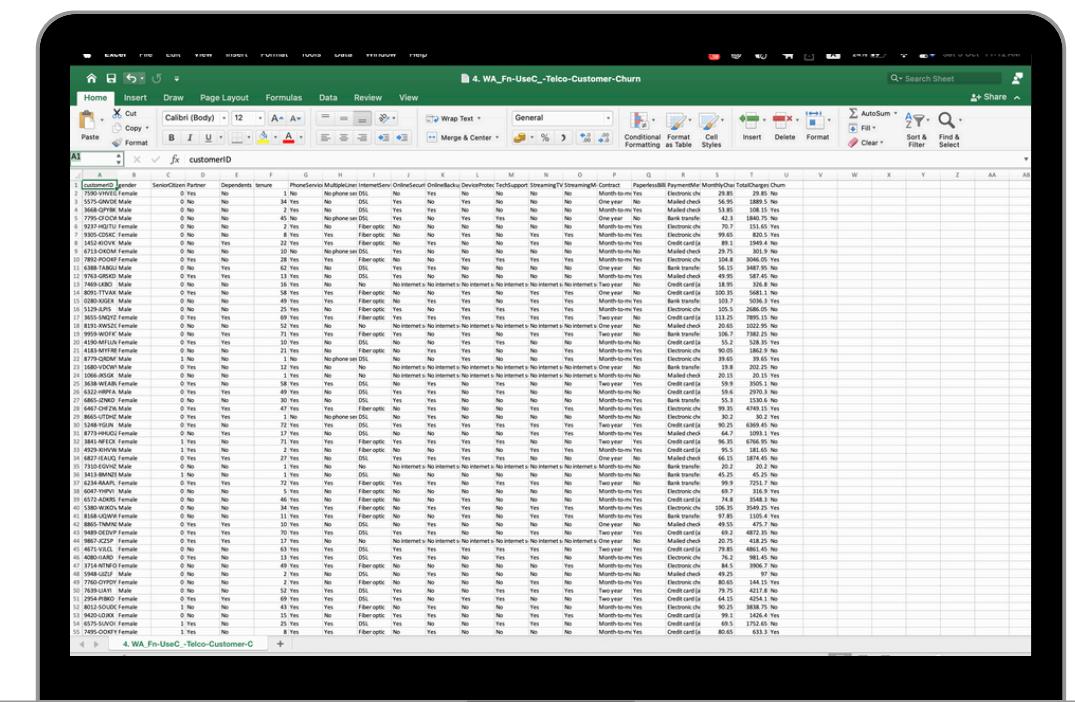
Categorical

NUMERICAL:

customer_id	age	tenure	balance	products_nur	credit_card	active_memk	estimated_sa	churn
15634602	42	2	0	1	1	1	101348.88	1
15647311	41	1	83807.86	1	0	1	112542.58	0
15619304	42	8	159660.8	3	1	0	113931.57	1
15701354	39	1	0	2	0	0	93826.63	0
15737888	43	2	125510.82	1	1	1	79084.1	0
15574012	44	8	113755.78	2	1	0	149756.71	1
15592531	50	7	0	2	1	1	10062.8	0
15656148	29	4	115046.74	4	1	0	119346.88	1
15792365	44	4	142051.07	2	0	1	74940.5	0
15592389	27	2	134603.88	1	1	1	71725.73	0
15767821	31	6	102016.72	2	0	0	80181.12	0
15737173	24	3	0	2	1	0	76390.01	0
15632264	34	10	0	2	1	0	26260.98	0
15691483	25	5	0	2	0	0	190857.79	0
15600882	35	7	0	2	1	1	65951.65	0
15643966	45	3	143129.41	2	0	1	64327.26	0
15737452	58	1	132602.88	1	1	0	5097.67	1
15788218	24	9	0	2	1	1	14406.41	0
15661507	45	6	0	1	0	0	158684.81	0
15568982	24	6	0	2	1	1	54724.03	0
15577657	41	8	0	2	1	1	170886.17	0
15597945	32	8	0	2	1	0	138555.46	0
15699309	38	4	0	1	1	0	118913.53	1
15725737	46	3	0	2	0	1	8487.75	0
15626047	29	5	0	1	1	1	197616.16	0

CATEGORICAL:

country	gender
France	Female
Spain	Female
France	Female
France	Female
Spain	Female
Spain	Male
France	Male
Germany	Female
France	Male
France	Male
Spain	Male
France	Female
France	Female
France	Male
France	Male
Spain	Male
France	Female
France	Female



Feature Encoding

Feature (categorical) Encoding refer to replacing categorical values (such as strings or text) with a numerical representation.

- To build predictive features from categories



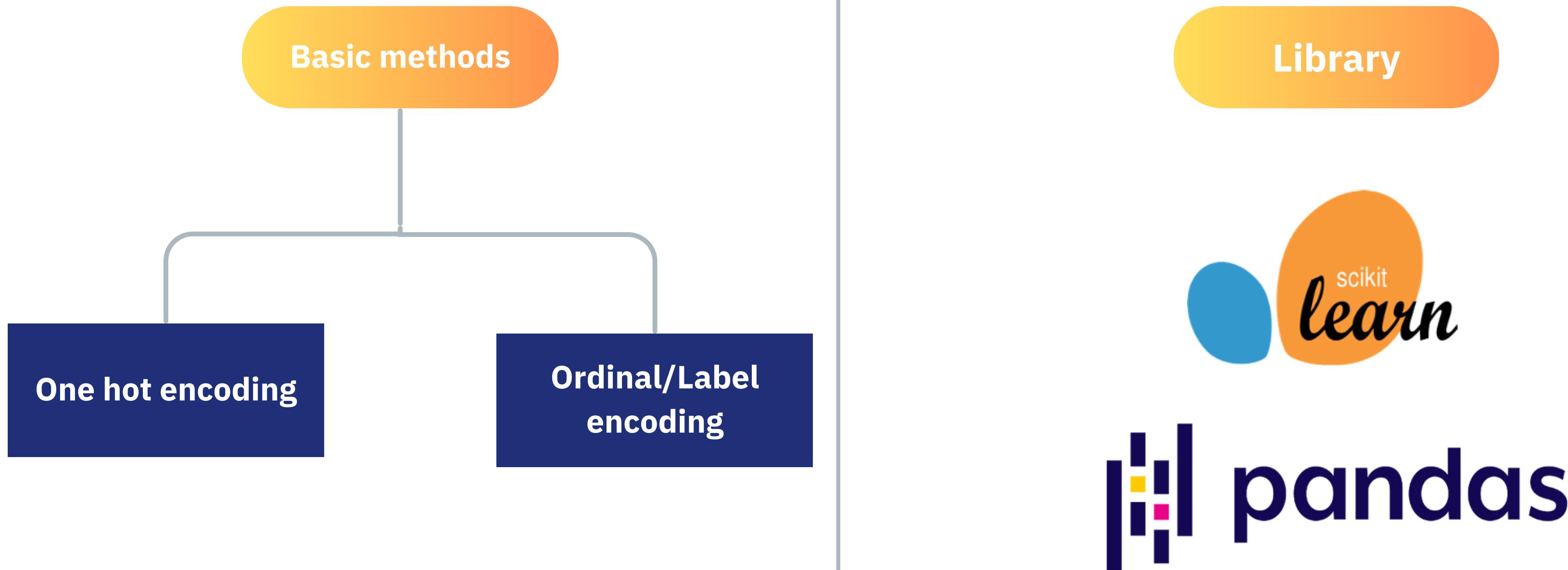
CATEGORICAL:

country	gender
France	Female
Spain	Female
France	Female
France	Female
Spain	Female
Spain	Male
France	Male
Germany	Female
France	Male
France	Male
France	Male
Spain	Male
France	Female
France	Female
Spain	Female

NUMERICAL:

country	gender	
France	Female	0
Spain	Female	
France	Female	
France	Female	
Spain	Female	
Spain	Male	1
France	Male	
Germany	Female	
France	Male	
France	Male	
France	Male	
Spain	Male	
France	Female	
France	Female	
Spain	Female	

Basic categorical encoding methods



One hot Encoding

One Hot Encoding consist in encoding categorical variables with binary set which take values 0 and 1 .

CATEGORICAL:

Fruit	Categorical value of fruit	Price
apple	1	5
apple	2	5
Banana	1	5
orange	3	20

NUMERICAL:

Fruit_apple	Fruit_Banana	Fruit_orage	price
1	0	0	5
1	0	0	5
0	1	0	5
0	0	1	20



One hot Encoding

One hot ending with **k-1** dummy

CATEGORICAL:

Fruit	Categorical value of fruit	Price
apple	1	5
apple	2	5
Banana	1	5
orange	3	20
orange	3	20

NUMERICAL:

Fruit_apple	Fruit_orage	price
1	0	5
1	0	5
1	0	5
0	1	20
0	1	20



One hot Encoding intro k-1 variables

- More generally, a categorical variable be encoded by creating $k-1$ binary variables, where k is the number of distinct categories
- In the case of binary variable, like gender where $k=2$ (male/female) we need to create only $1 (k-1) = 1$ binary variable.
- One hot encoding into **$k-1$** binary variables takes into account that we can use 1 less dimension and still represent the whole information.
- if the observation is 0 in all the binary variables, then it must be 1. in the final (not present) binary variable.

Two hot Encoding

Two Hot Encoding Consist in encoding categorical variables with banary set which take values 0 and 1 .

CATEGORICAL:

Fruit	Categorical value of fruit	Price
apple	1	5
apple	2	5
apple	1	5
orange	3	20

NUMERICAL:



Fruit	price
1	5
1	5
1	5
0	20

Ordinal/Label/integer/encoding

Ordinal encoding consist in replacing the categories by digit from **1** to **n** (or **0** to **n-1**) , depending the implementation).

CATEGORICAL:

Size	Categorical value	Price
Small	1	5
Medium	2	5
Large	1	5
Large	3	20

NUMERICAL:



Fruit	price
0	5
1	5
2	5
2	20

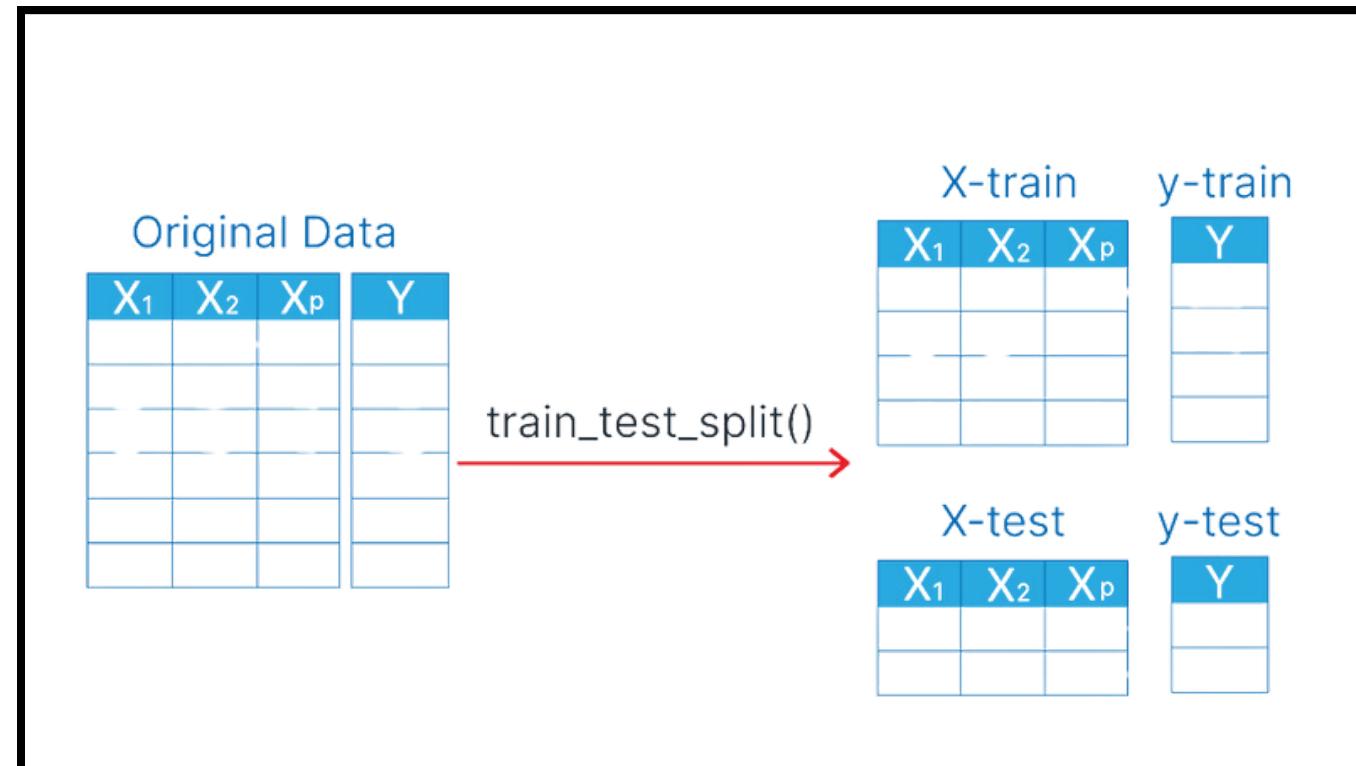
CROSS VALIDATION



Split Data

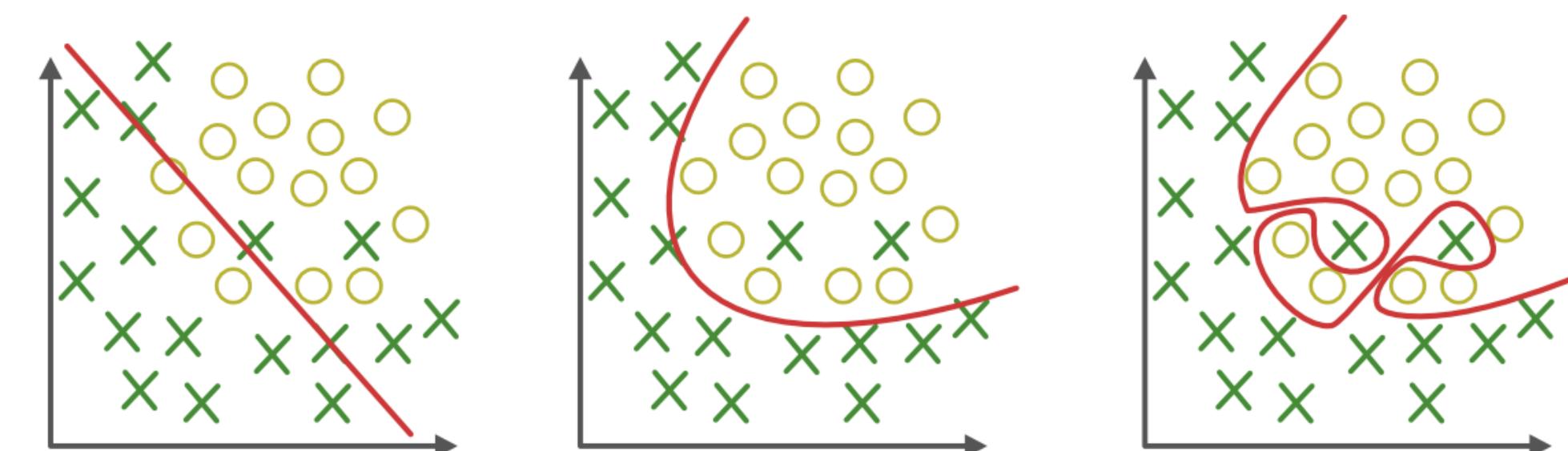
Performance Evaluation

Train Test Split



- The **training data** set is used to train and develop models.
 - The **testing data** set is used after the training is done.

Detecting Overfitting and Underfitting



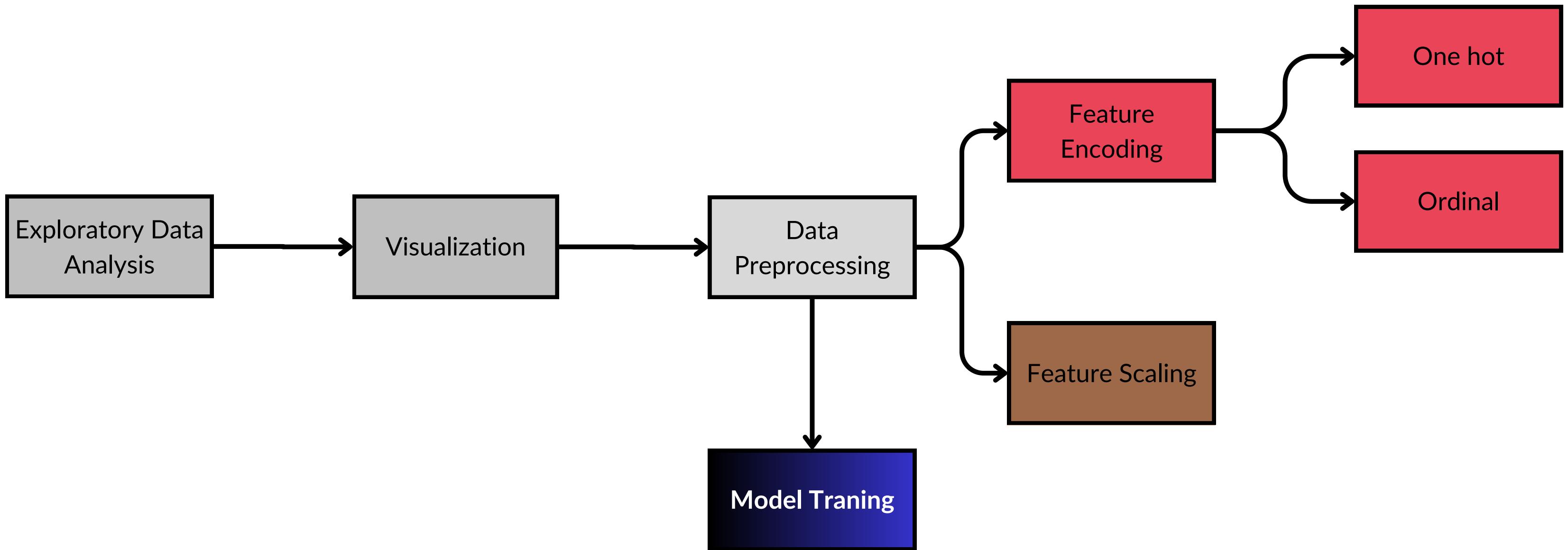
FEATURE SCALING



What is Feature Scaling

Feature scaling refer to the methods used to normalize the range of values of independent variables.

- In other word, the methods to set the feature value range within a similar scale.



Why Feature Scaling

Regression Coefficient and Scale

The size of the regression coefficient depends on the scale of the variable. Larger numbers in data can overpower smaller ones when building a regression model.

Bigger Numbers Dominate

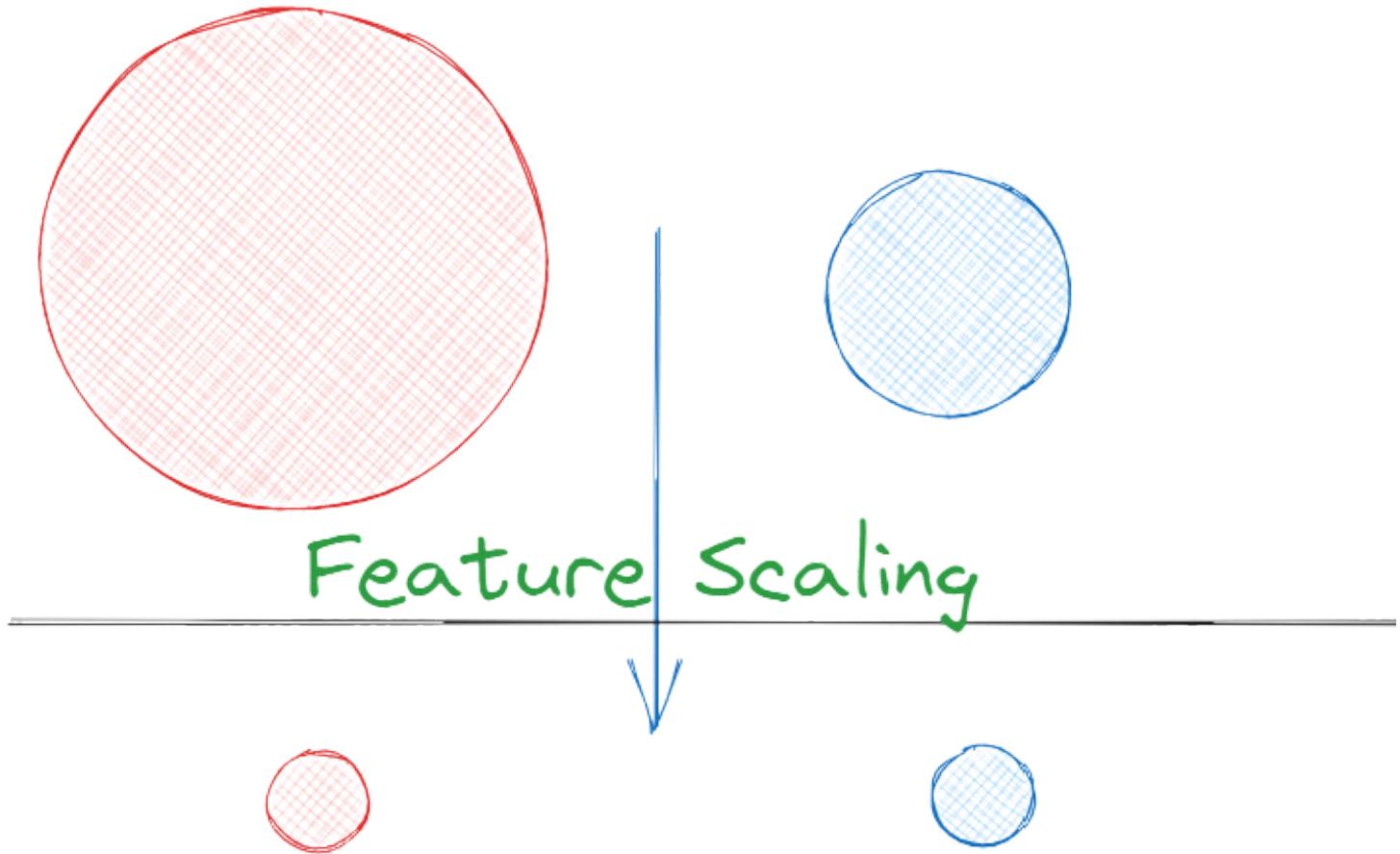
If one variable has values in thousands and another has values in single digits, the larger numbers will have more influence in the model, even if the smaller numbers are just as important.

Gradient Descent and Scaling

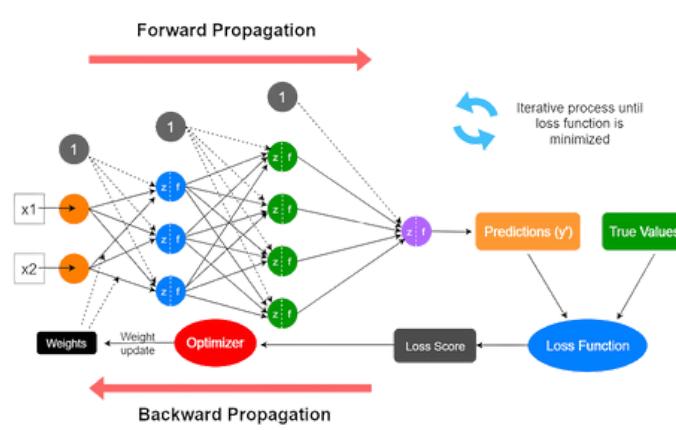
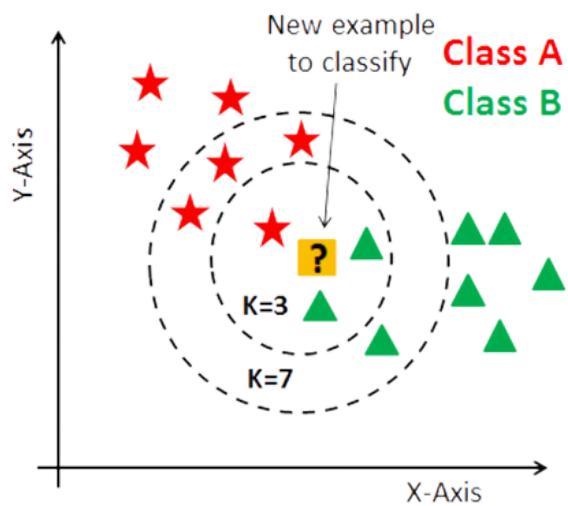
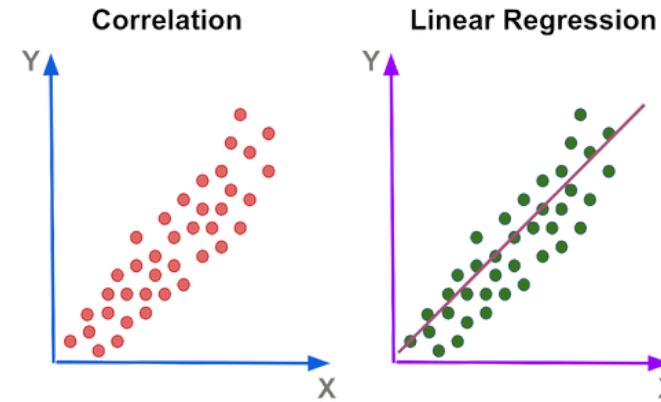
When all features are on a similar scale (e.g., 0–1), gradient descent, a method to train models, works faster and finds the best solution more quickly.

Scaling for SVM

Scaling features also speeds up the process of finding the support vectors in Support Vector Machines (SVMs), making the algorithm more efficient.



Why Feature Scaling



The Machine Learning models affected by the magnitude of the feature:

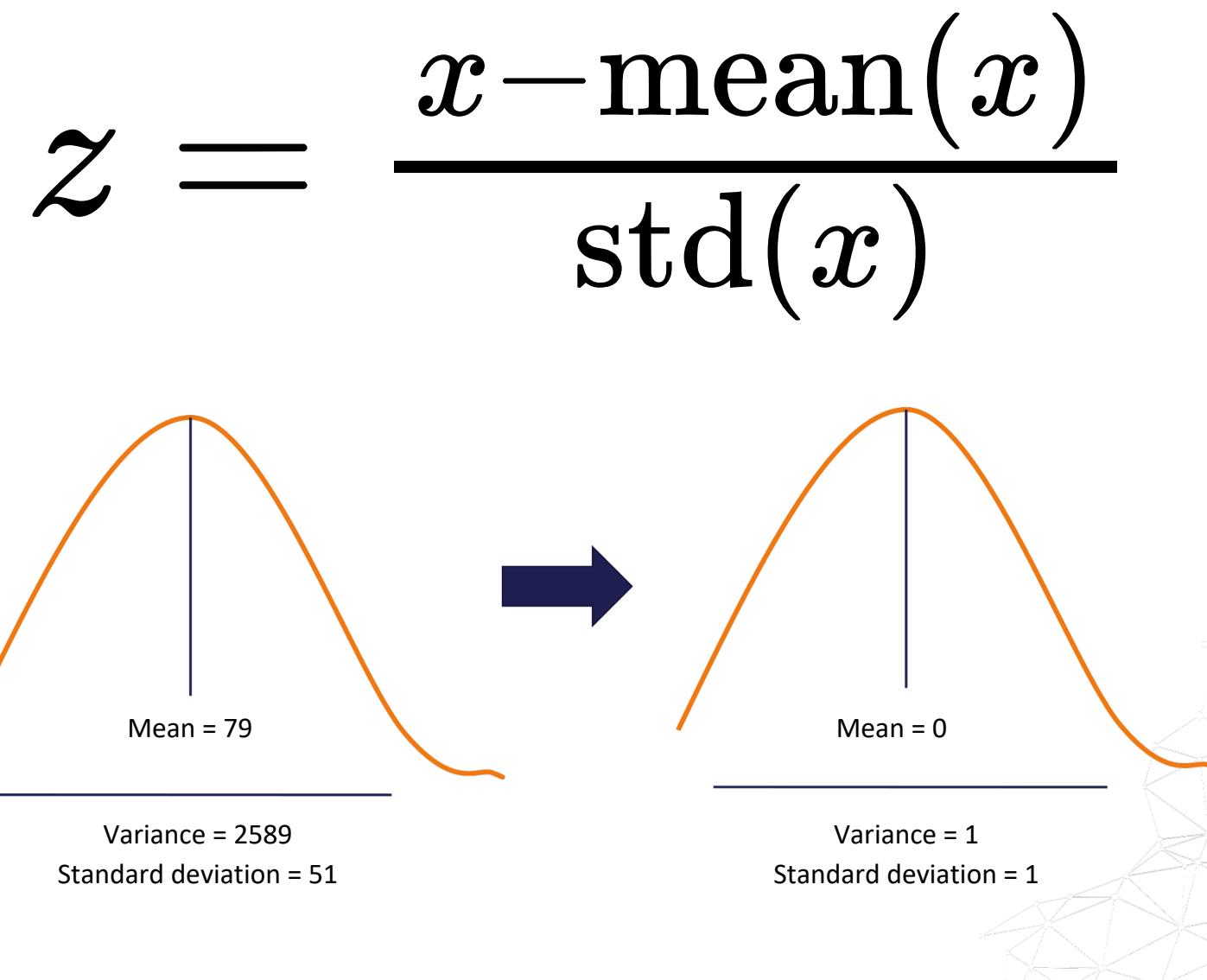
- **Linear and Logistics Regression**
- **Neural Networks**
- **KNN**
- **K-means clustering**
- **SVMs**
- **Linear Discriminant Analysis (LDA)**
- **Principal Component Analysis (PCA)**

Machine Learning model insensitive to feature magnitude are the are the ones based on tree:

- **Classification and Regression Trees**
- **Random Forests**
- **Gradient Boosted Trees**

Standardisation

Mean the variables at 0 and sets the variance to 1



Standardisation: example

Mean = 79
Standard dev = 51

Obs. -Mean

Standard dev

Price
100
90
50
40
20
100
50
60
120
40
200

Price

Price
0.41
0.22
-0.57
-0.76
-1.16
0.41
-0.57
-0.37
0.80
-0.76
2.37

Min-Max Scaing

Scales the variable between 0 and 1

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

MinMaxScaling: example

Price
100
90
50
40
20
100
50
60
120
40
200

Max = 79
Min = 20
Range = 180



Price
0.44
0.39
0.17
0.11
0.00
0.44
0.17
0.22
0.56
0.11
1.00

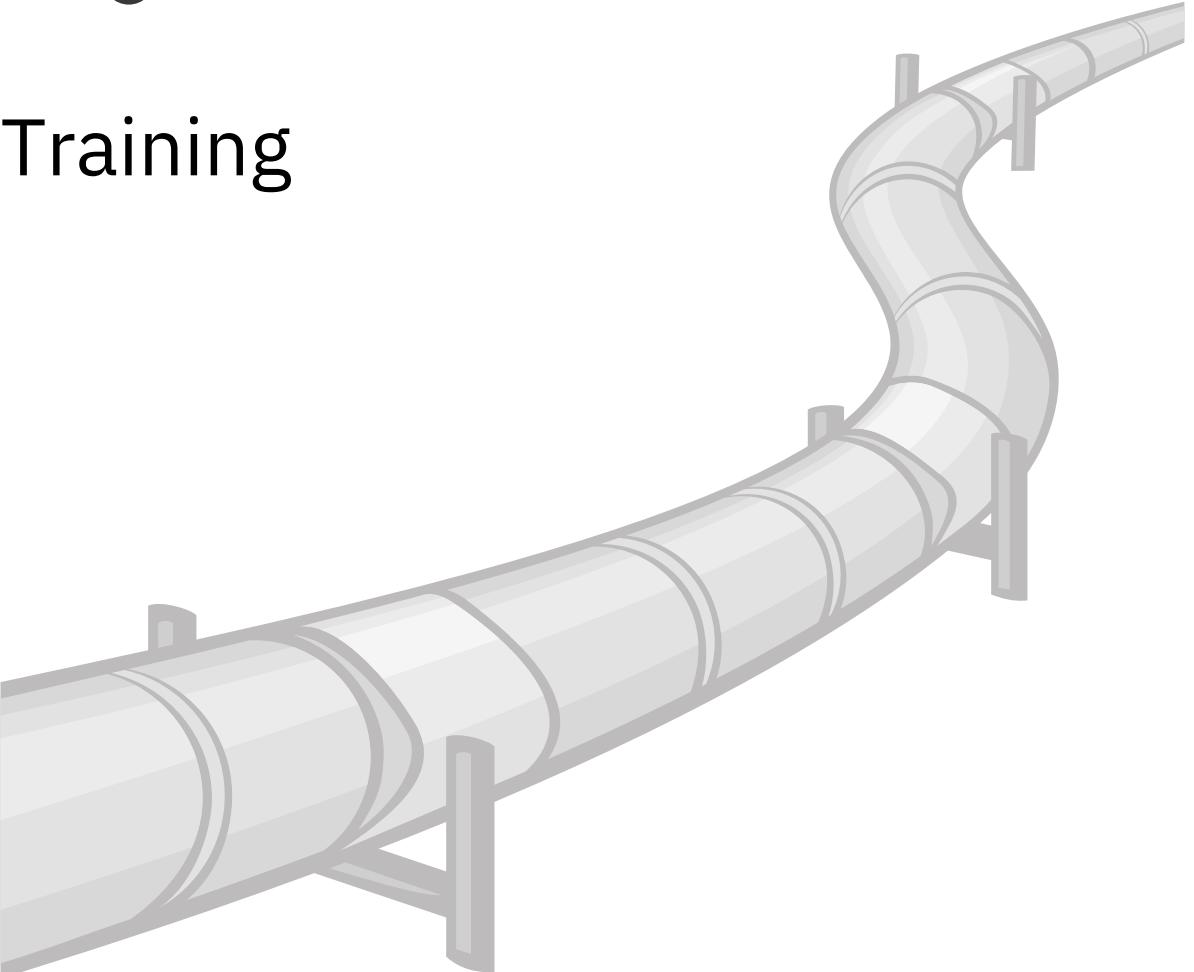
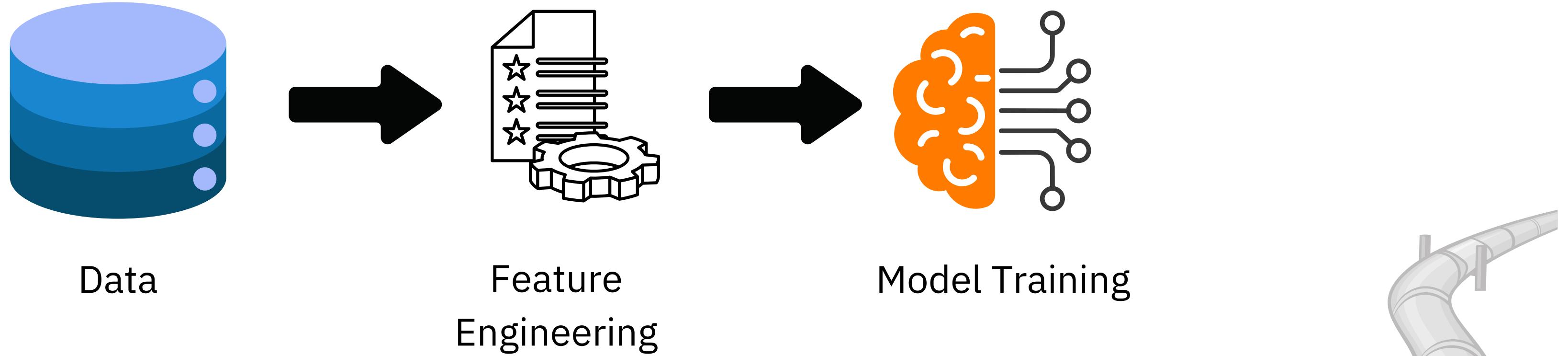
Min-Max vs. Standardisation

	Standardisation	Min-Max
Range	Mean = 0, Std. Dev = 1	Less sensitive
Sensitivity to outliers	Scaled to [0, 1] (or other specified range)	Highly sensitive
Preferred for	Distance-based algorithms (e.g., PCA, SVM)	Neural Networks, kNN, K-Means



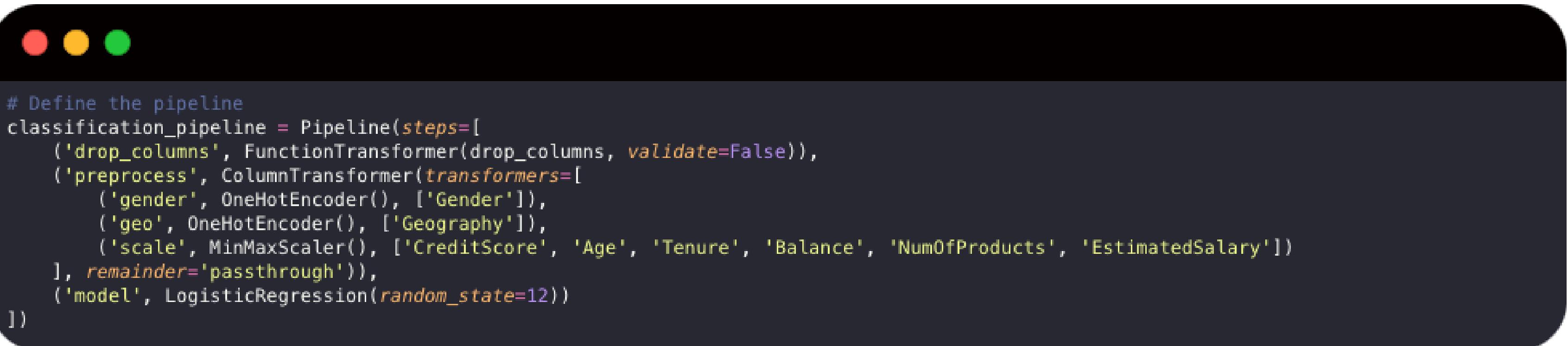
Feature Engineering Pipeline

Putting it all together



Feature Engineering Pipeline

A **machine learning (ML) pipeline** is a series of steps that automate the process of building, training, evaluating, and deploying ML models.



```
# Define the pipeline
classification_pipeline = Pipeline(steps=[
    ('drop_columns', FunctionTransformer(drop_columns, validate=False)),
    ('preprocess', ColumnTransformer(transformers=[
        ('gender', OneHotEncoder(), ['Gender']),
        ('geo', OneHotEncoder(), ['Geography']),
        ('scale', MinMaxScaler(), ['CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'EstimatedSalary'])
    ], remainder='passthrough')),
    ('model', LogisticRegression(random_state=12))
])
```