Data Journalism

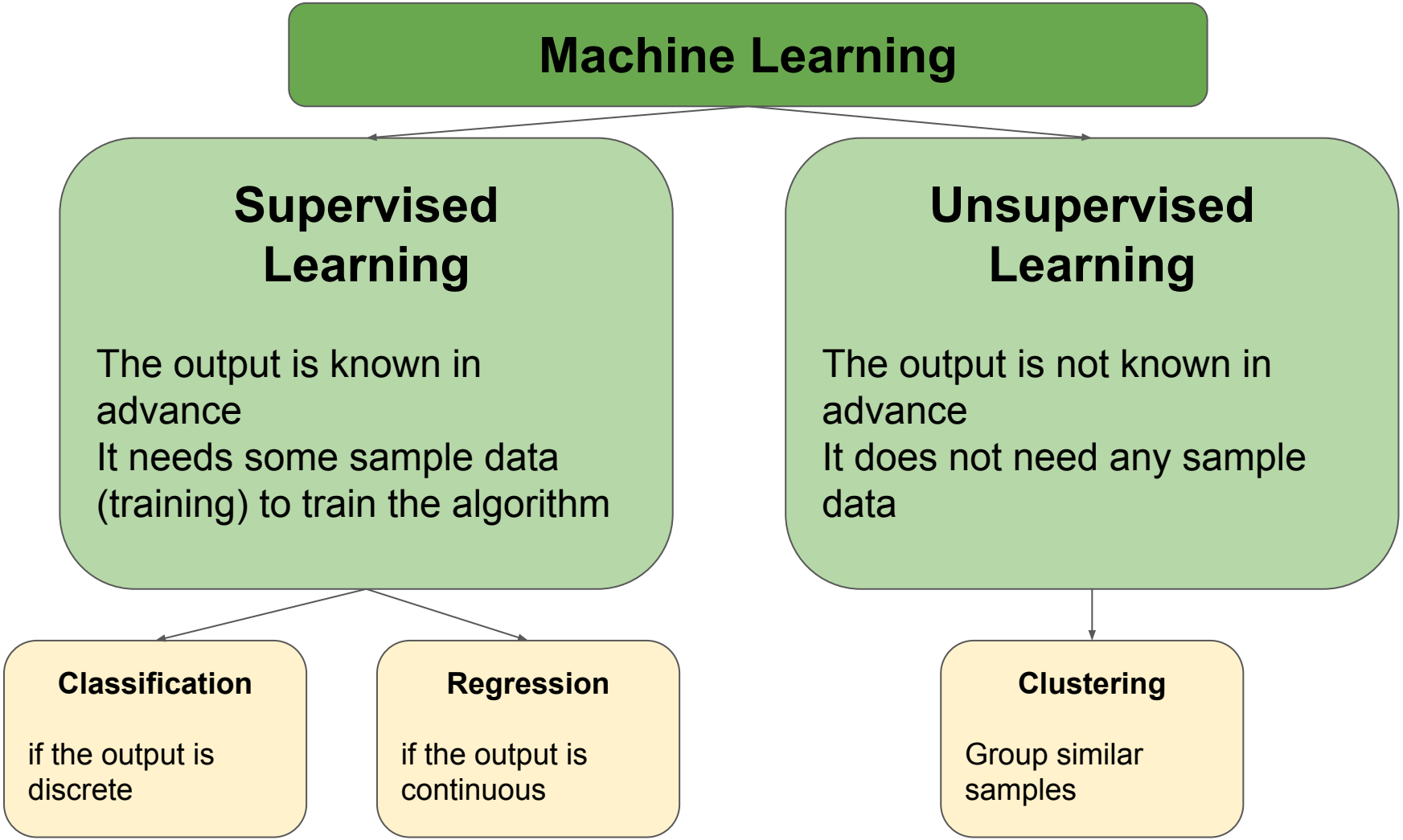
Data Analysis

Part 2

Angelica Lo Duca
angelica.loduca@iit.cnr.it

Machine Learning is the study of computer algorithms that improve automatically through experience and by the use of data*

Wikipedia

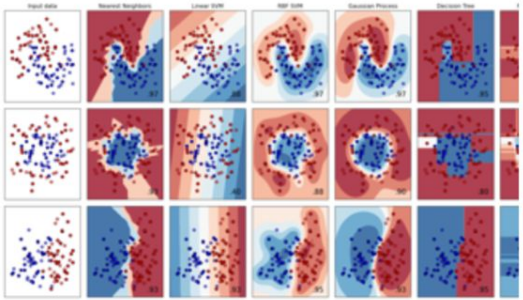

Machine Learning
Supervised Learning
The output is known in advance
It needs some sample data (training) to train the algorithm
Unsupervised Learning
The output is not known in advance
It does not need any sample data
Classification
if the output is discrete
Regression
if the output is continuous
Clustering
Group similar samples

Scikit-Learn - Python Library for Machine Learning

Classification

Identifying which category an object belongs to.

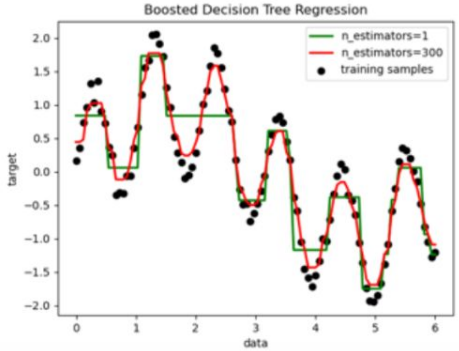
Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...



Regression

Predicting a continuous-valued attribute associated with an object.

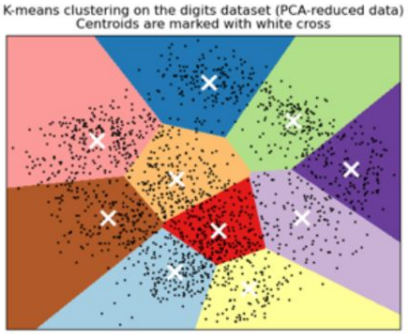
Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...



Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...



(Input) Features

A set of attributes (columns) in a dataset provided as input to a Machine Learning Algorithm.

Features: X, Y, Z

Numeric features: X,Y

Categorical features: Z

X	Y	Z
3.4	2.3	A
4.5	3.4	B

Types of Classification

Output classes must be numbers.

Binary Classification

- only two output classes: 0 - 1

Multiclass Classification

- many output classes: 0,1, … N

Example - Classification

Given an email, recognize if it is spam or not.

Discrete Output
Yes/No

Training Data
the output is already known

Text	Sender	Spam
Dear customer, this is a nice job offer for you.	Customer Service	Yes
Hi Angelica, how are you?	Giulia	No
This product is for you.	Product Sellers	Yes
Dear Angelica, I would like to inform you that tomorrow there will be a party.	Maria	???

Output Classes Yes/No must be converted to 0/1

Example - Regression

Understand the relationship between drug dosage and blood pressure of patients

Training Data
the output is already known

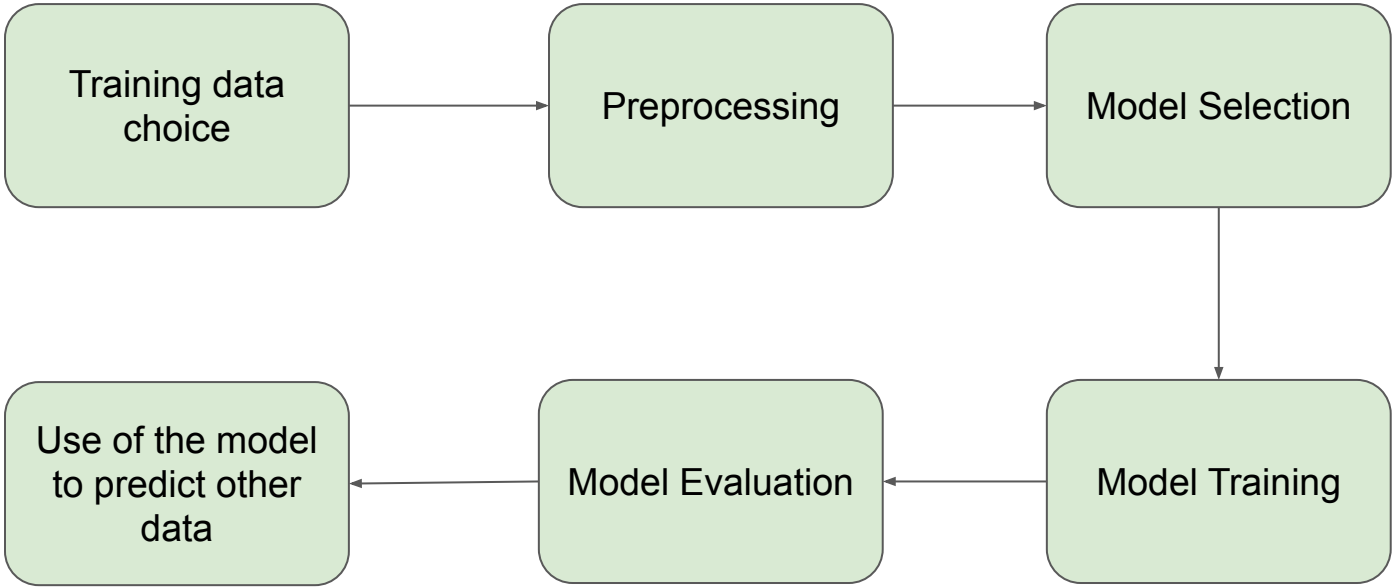
Blood Pressure	Age	Drug dosage
100	76	70%
80	22	35%
95	76	32%
44	76	???

Example - Clustering

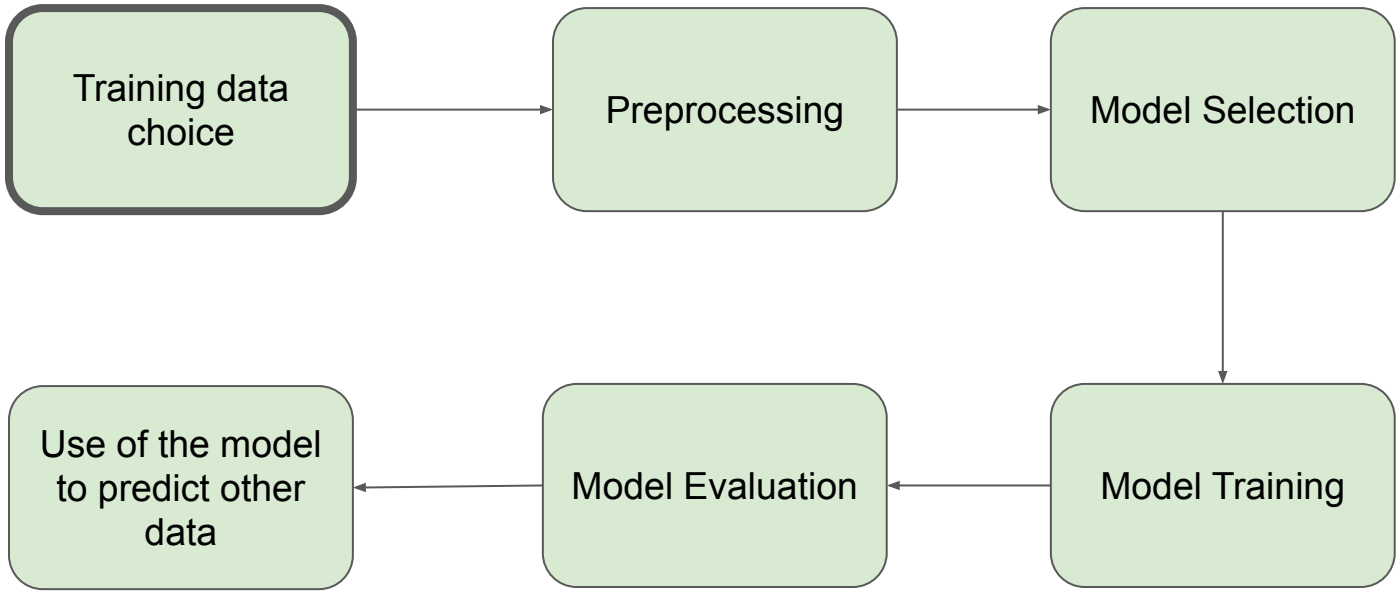
Discover the votes distribution in a classroom of students

Sex	Vote
F	30
M	22
M	28
M	18

Supervised Learning Workflow



Supervised Learning Workflow



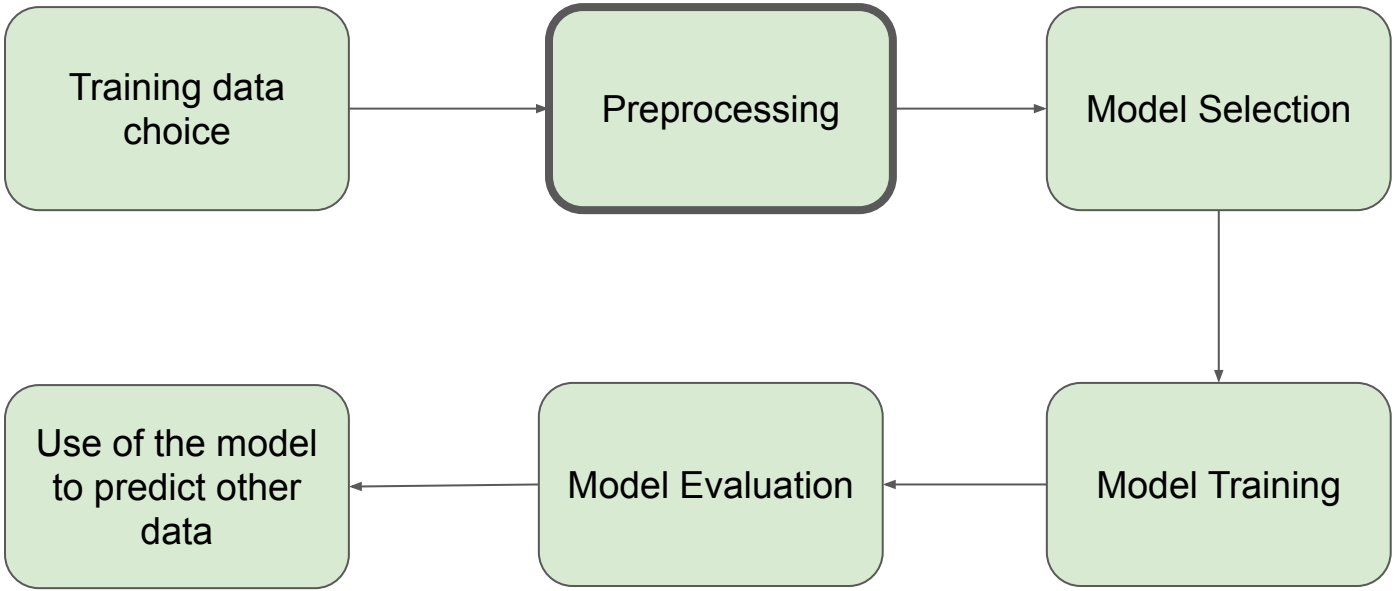
Training Data Choice

Use a subset of all the available dataset

Choose training data randomly or use a specific criterium

```
import numpy as np
from sklearn.model_selection import train_test_split

X, y = np.arange(10).reshape((5, 2)), range(5)
X_train, X_test, y_train, y_test = train_test_split( X, y,
test_size=0.33, random_state=42)
```

Supervised Learning Workflow



Preprocessing

Data Cleaning

Features Selection

Data Normalization

Data Balancing

Features Selection

Data contain some features that are either redundant or irrelevant, and thus can be removed.

Features selection is the process of selecting a subset of relevant features

Features Selection VS Features Extraction

Feature extraction creates new features from functions of the original features

Feature selection returns a subset of the features.

Example of Features Extraction: from a text, extract the TFIDF (Term Frequency Inverse Document Frequency) of each word in the text, i.e. how much is important the word in the text.

Techniques for Features selection

Filter Methods

Wrapper Methods

Embedded Methods

Filter Methods

Find relations among features and, on the basis of a score, decide if to discard a feature or not.

- *Numerical Features* - use the correlation coefficient, such as the Pearson Coefficient and the ANOVA test
- *Categorical Features* - use the chi-square test

Filter methods are time consuming.

Wrapper Methods

The selection of features is done while running the model.

1. Select a subset of features
2. Train a predictive model on selected features
3. Score the model performance
4. Start again from point 1

As wrapper methods train a new model for each subset, they are very **computationally intensive**, but usually provide the best performing feature set for that particular type of model or typical problem.

Wrapper techniques

- **RFE** (Recursive Feature Elimination)
- **Stepwise regression** - adds the best feature or deletes the worst feature at each round

Embedded Methods

Different regularization methods are used.

The most common methods are **Ridge Regression** and **Lasso Regression**.

Features Selection links

Scikit-Learn Features Selection

Data Vedas

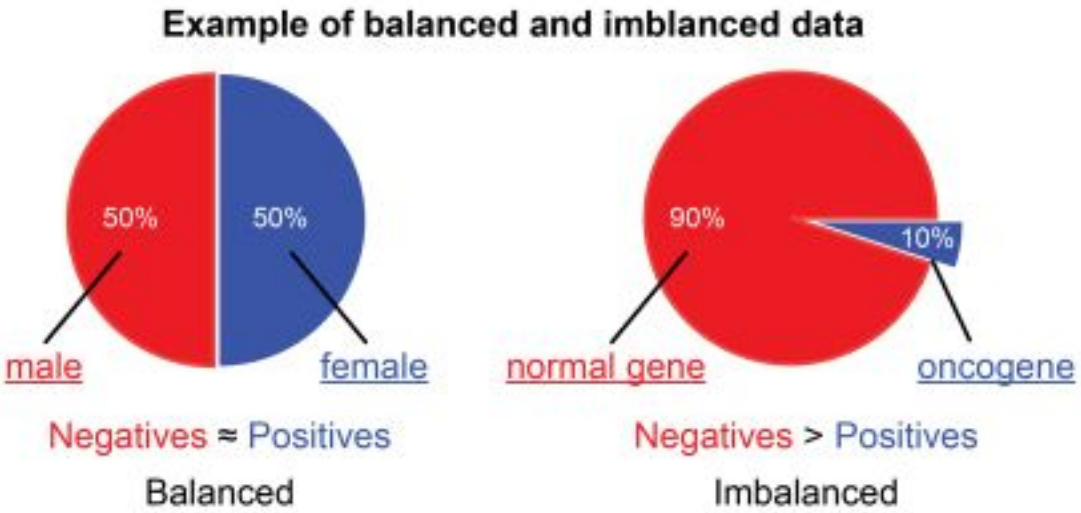
Data Normalization

Already seen with Data Cleaning

All the input features must fall in the same range intervals, e.g. [0,1]

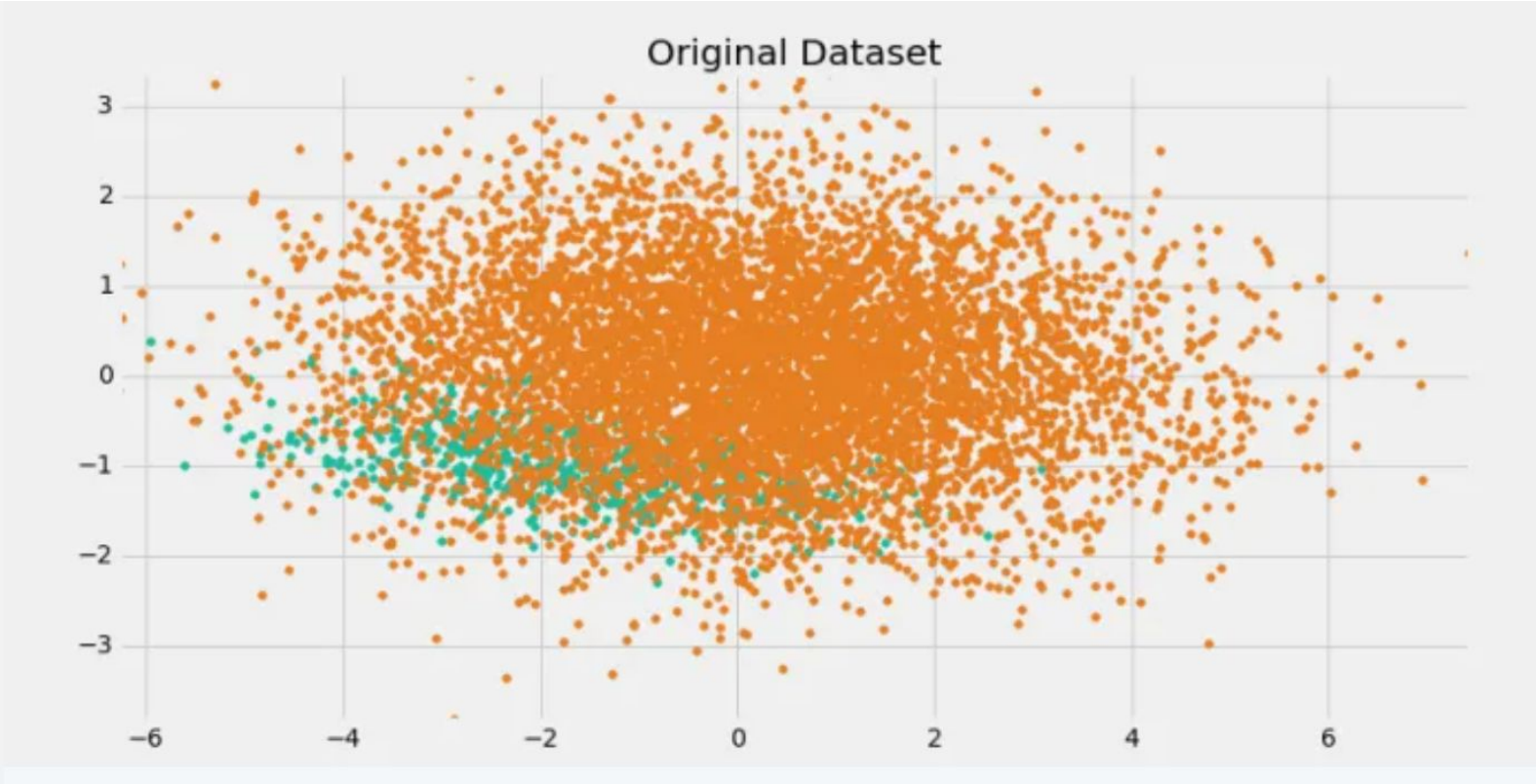
Balancing

Imbalanced dataset = a dataset where the distribution of output class is not fair.



Source from https://medium.com/analytics-vidhya/what-is-balance-and-imbalance-dataset-89e8d7f46bc5

Balancing (cont.)



Balancing Techniques

All the output class must be equally represented

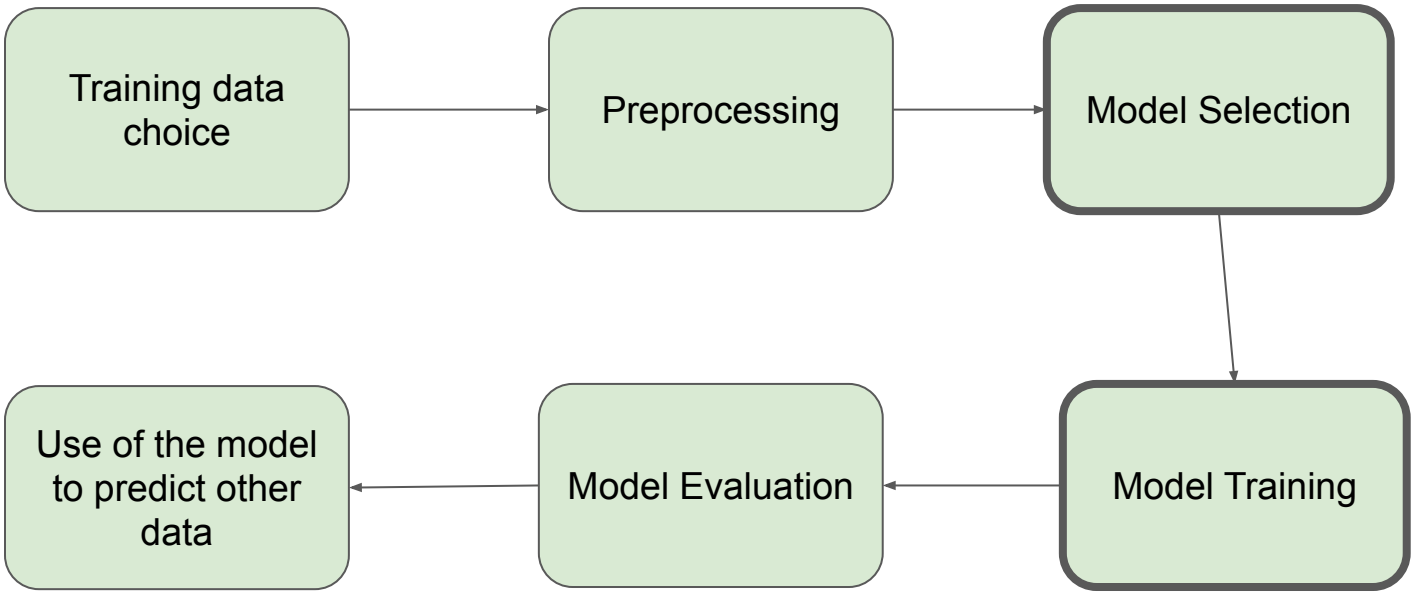
- **oversample the minority class** - create synthetic data to represent the minority class
  -
- **undersample the majority class** - remove data to reduce the number of samples in the majority class

Balancing in Python

Python library Imbalanced Learn

```
pip install -U imbalanced-learn
```

Supervised Learning Workflow



Model Selection and Training

All the models follow this structure:

1. build the model
2. fit the model on training data
3. use the model to predict the output of new data

Example:

```
from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n_neighbors=3)
model.fit(X, y)
model.predict([[1.1]])
```

Parameters Tuning

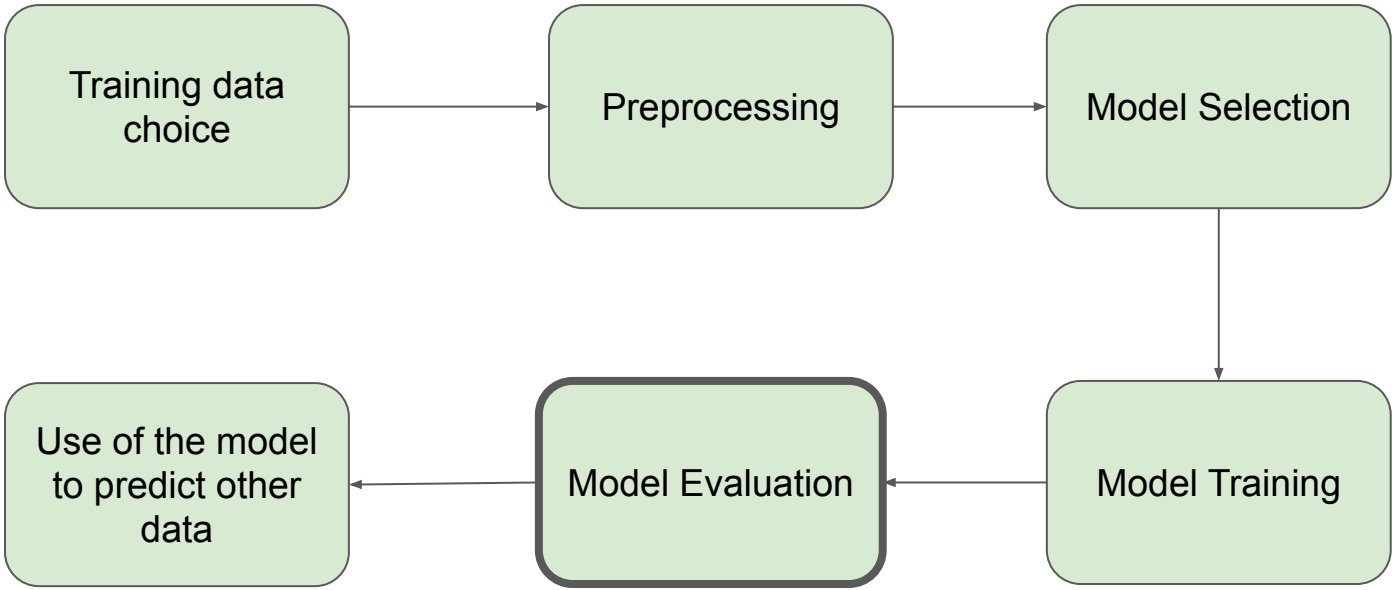
Select the best combination of parameter values for a given model.

Grid Search Cross Validation is a technique which permits to define a set of possible values we wish to try for the given model and it trains on the data and identifies the best estimator from a combination of parameter values.

Parameters Tuning in Scikit-learn

```
from sklearn.model_selection import GridSearchCV
from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier()

param_grid = {
    'n_neighbors': [3, 30],
    'algorithm' : ['auto', 'ball_tree', 'kd_tree', 'brute']
}

grid = GridSearchCV(model, param_grid = param_grid)
grid.fit(X_train, y_train)


best_estimator = grid.best_estimator_
```

Supervised Learning Workflow



Model Evaluation

Predicted \ Actual class	P	N
P	TP	FP
N	FN	TN

$$false\ positive\ rate = \frac{false\ positives}{false\ positives + true\ negatives}$$

$$true\ positive\ rate = \frac{true\ positives}{true\ positives + false\ negatives}$$

Precision, Recall and Accuracy

$$\textbf{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

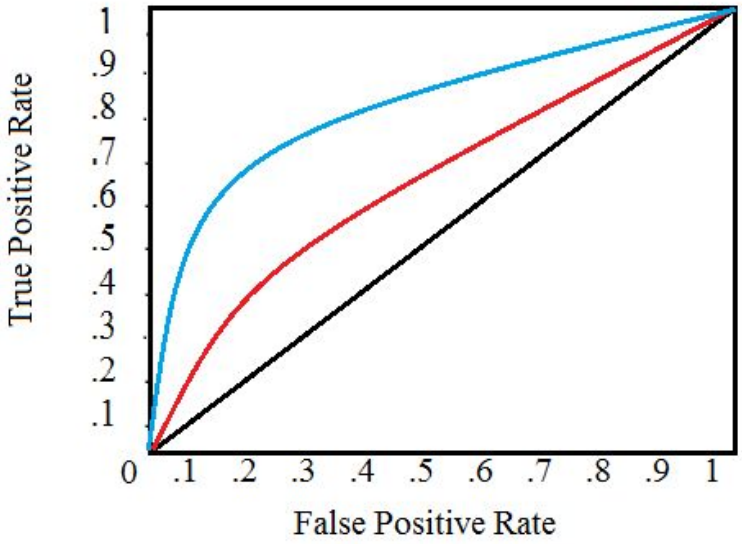
$$\textbf{Recall} = \frac{\text{True Positive}}{\text{Predicted Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

$$\textbf{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{Total}}$$

Receiver Operating Characteristic (ROC)

Le curve ROC passano per i punti (0,0) e (1,1), avendo inoltre due condizioni che rappresentano due curve limite:

- una che taglia il grafico a 45°, passando per l'origine. Questa retta rappresenta il caso del classificatore casuale (linea di «nessun beneficio»), e l'area sottesa AUC è pari a 0,5.
- la seconda curva è rappresentata dal segmento che dall'origine sale al punto (0,1) e da quello che congiunge il punto (0,1) a (1,1), avendo un'area sottesa di valore pari a 1, ovvero rappresenta il classificatore perfetto.



From [Wikipedia]