Hands on
Machine Learning

2019 AI summer program in Asia University

Yueh-Lin Tsai

2019 / 07

About me

- Yueh-Lin Tsai

- Education
  - National Cheng Kung University, M.S., Psychology (2013-2015)
  - National Cheng Kung University, B.S., Psychology (2009-2013)

- Present
  - AI Engineer in Taiwan AI Academy

What is Artificial Intelligence ?

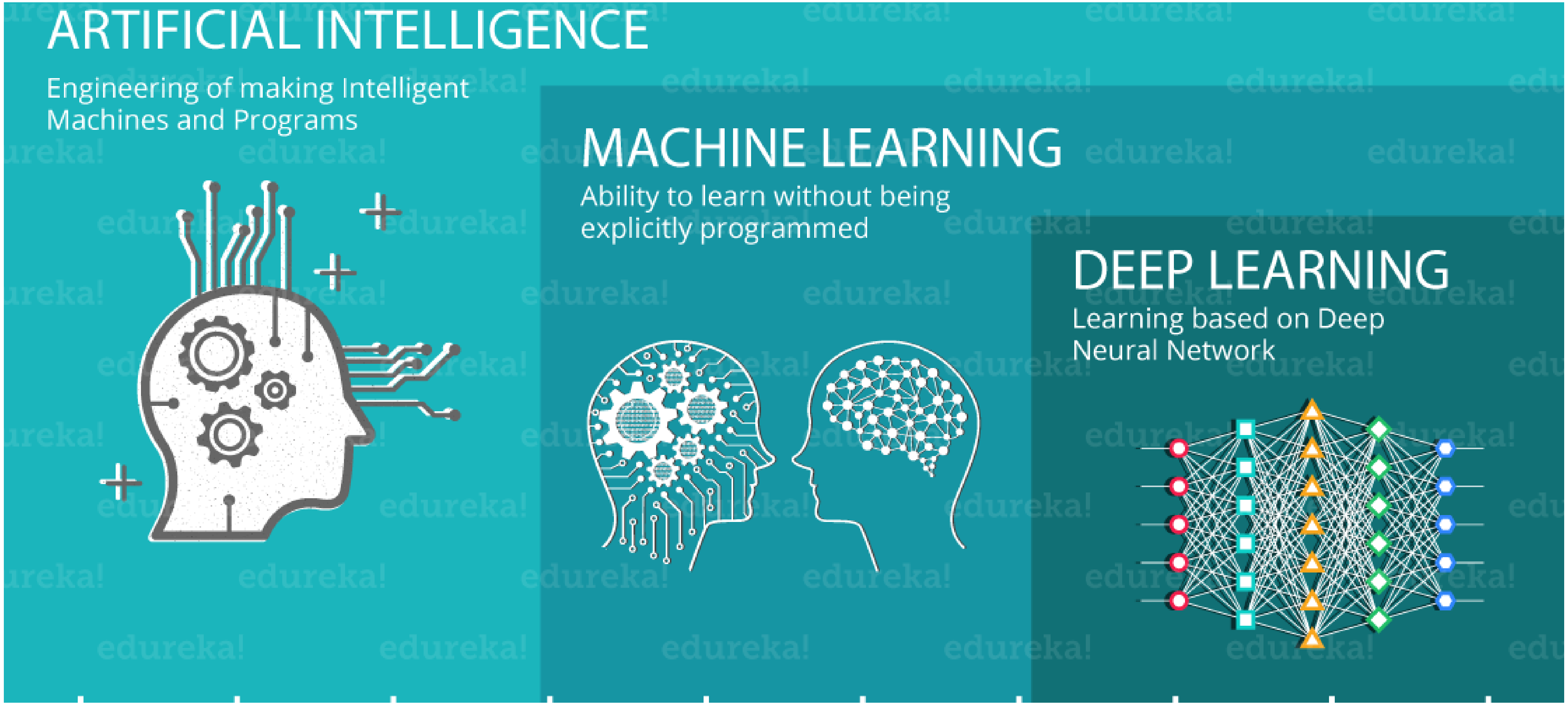
Artificial Intelligence

- Definition : Intelligence demonstrated by machine

- How ?

Expertise system	Machine learning
Explicit rule from human	Get knowledge from data

Modern Artificial Intelligence

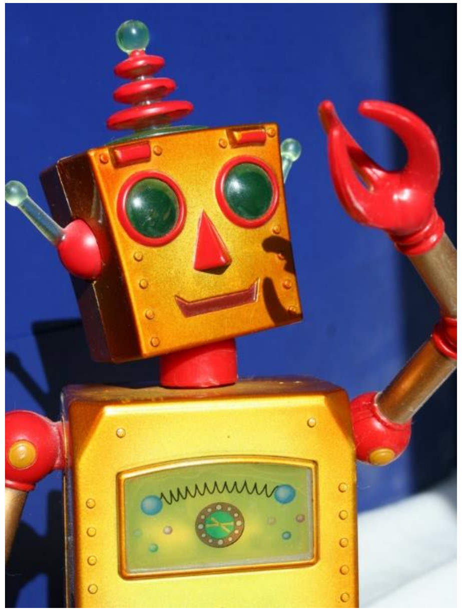


Source

Machine learning

- Extract relations/patterns from data automatically

- Apply those rules to unseen data



$$f(x) = y$$

Type of machine learning

- Supervised learning
  - Regression
  - Classification

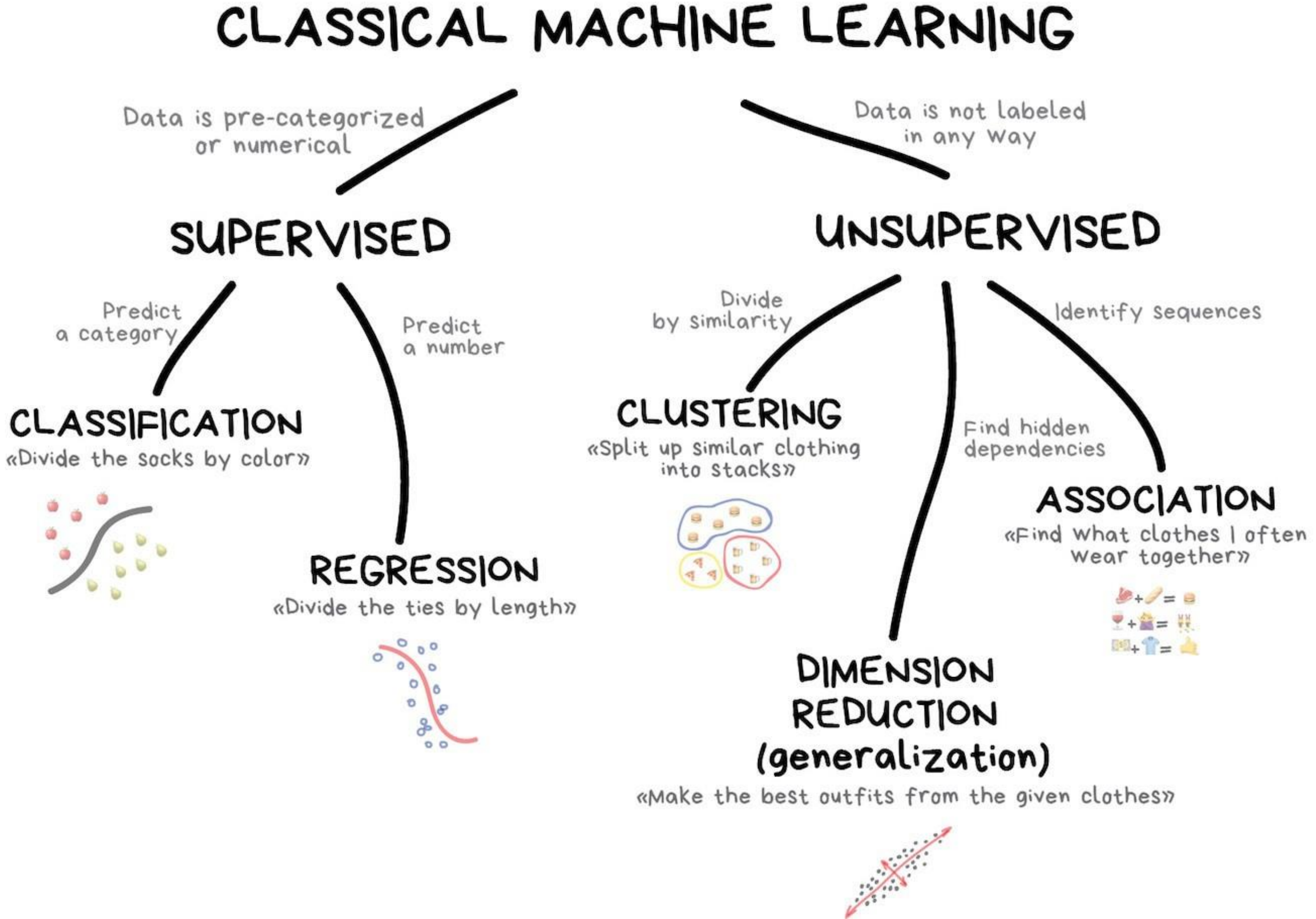
Problems with answer

- Unsupervised learning
  - Cluster
  - Dimension reduction

Problems without answer

- Reinforcement learning

Problems with fuzzy metric


CLASSICAL MACHINE LEARNING
Data is pre-categorized or numerical
Data is not labeled in any way
SUPERVISED
UNSUPERVISED
Predict a category
Predict a number
Divide by similarity
Identify sequences
CLASSIFICATION
«Divide the socks by color»
REGRESSION
«Divide the ties by length»
CLUSTERING
«Split up similar clothing into stacks»
Find hidden dependencies
ASSOCIATION
«Find what clothes I often wear together»
DIMENSION REDUCTION (generalization)
«Make the best outfits from the given clothes»

Source

python™

+

Time for practice

Familiar with colab and python

Machine learning workflow

1. Problem definition
2. Data collection
3. Data exploration / preprocessing
4. Build model
5. Model evaluation

Cheat sheet – scikit learn

Machine Learning with Python

Collect Data

- BeautifulSoup
- Lxml
- Requests
- Pandas

Preprocessing and EDA

- Numpy
- Pandas
- Scikit-learn
- Matplotlib
- NLTK

Analysis and Modeling

- Statsmodels
- Scikit-learn
- Tensorflow
- Keras
- Pytorch

Machine Learning with Python

Collect Data

- BeautifulSoup
- Lxml
- Requests
- Pandas

Preprocessing and EDA

- Numpy
- Pandas
- Scikit-learn
- Matplotlib
- NLTK

Analysis and Modeling

- Statsmodels
- Scikit-learn
- Tensorflow
- Keras
- Pytorch

Build up your ML model in one slide

```python
import pandas as pd
from sklearn import preprocessing, linear_model, model_selection, metrics

data = pd.read_csv('example_data.csv')

data_y = data['target']
data = data.drop('target', axis = 1, inplace = True)

one_hot_data = pd.get_dummies(data)

ss = preprocessing.StandardScaler()
scale_data = ss.fit_transform(data)

train_x, test_x, train_y, test_y = model_selection.train_test_split(data, data_y, test_size = 0.2, random_state = 99)

model = linear_model.LinearRegression() # LogisticRegression()
model.fit(train_x, train_y)

test_prediction = model.predict(test_x)
print('r-square of linear regression : {:.3f}'.format(metrics.r2_score(test_prediction, test_y)))
```

Machine learning workflow

1. Problem definition
2. Data collection
3. **Data exploration / preprocessing**
4. Build model
5. Model evaluation

[Cheat sheet – scikit learn]

Exploration and preprocessing

Data exploration

- Get to know your dataset
  - How many data do I have ?
  - Which column/feature do I have ?
  - Statistics and relations between columns ?
  - Outlier or missing data ?

Data preprocessing

- Handle missing data
  - Delete data which have missing values (row or column)
  - Missing imputation
- Handle outliers
  - Distribution transformation
  - Replace outliers

Data preprocessing

```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

- Convert categorical data to numerical data
  - Label encoding
  - One-hot encoding

Name	Score
Amy	78
Bob	90
Chris	65
Amy	86
Chris	67

Name_label	Score
1	78
2	90
3	65
1	86
3	67

Label encoding

Amy_oh	Bob_oh	Chris_oh	Score
1	0	0	78
0	1	0	90
0	0	1	65
1	0	0	86
0	0	1	67

One-hot encoding

Data preprocessing

```
from sklearn.preprocessing import StandardScaler, MinMaxScaler
```

- Normalize data
  - Standard scale
  - Min-max scale

$$x_{standard} = \frac{x - \mu}{\sigma}$$

$$x_{minmax} = \frac{x - Min(x)}{Max(x) - Min(x)}$$

Name	Score
Amy	78
Bob	90
Chris	65
Amy	86
Chris	67

Name	Score
Amy	0.0719
Bob	1.1509
Chris	-1.0969
Amy	0.7912
Chris	-0.9171

Standard scale

Name	Score
Amy	0.48
Bob	0
Chris	1
Amy	0.16
Chris	0.92

Min-max scale

Data preprocessing

```
from sklearn.preprocess import train_test_split
```

- Data splitting
  - Training set
  - Validation set
  - Testing set

- Cross validation

Time for practice

Data exploration and preprocessing

Machine learning workflow

1. Problem definition
2. Data collection
3. Data exploration / preprocessing
4. **Build model**
5. Model evaluation

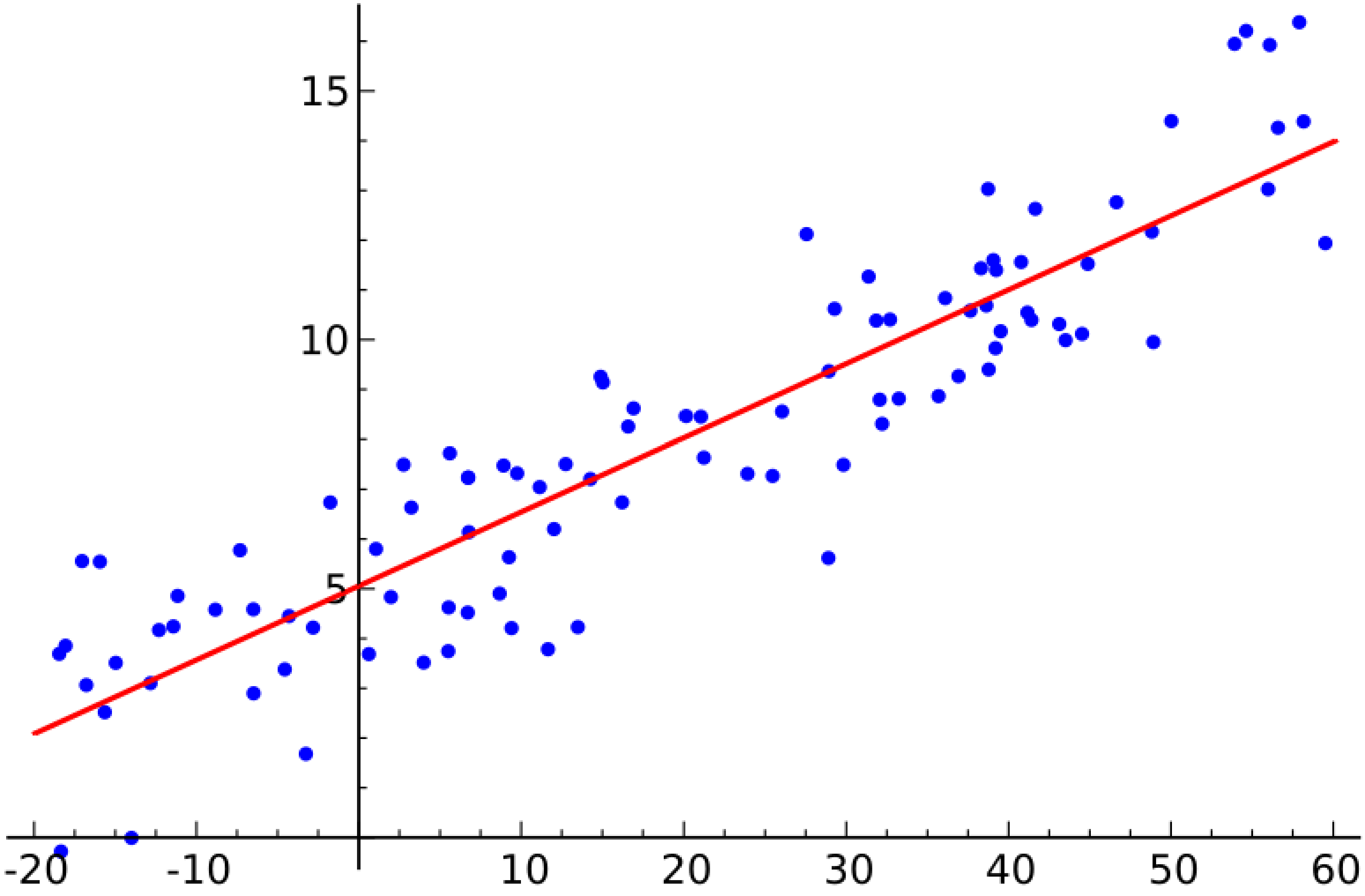
[Cheat sheet – scikit learn]

Build model

Build model

For regression problem

- Linear model
  - Linear regression

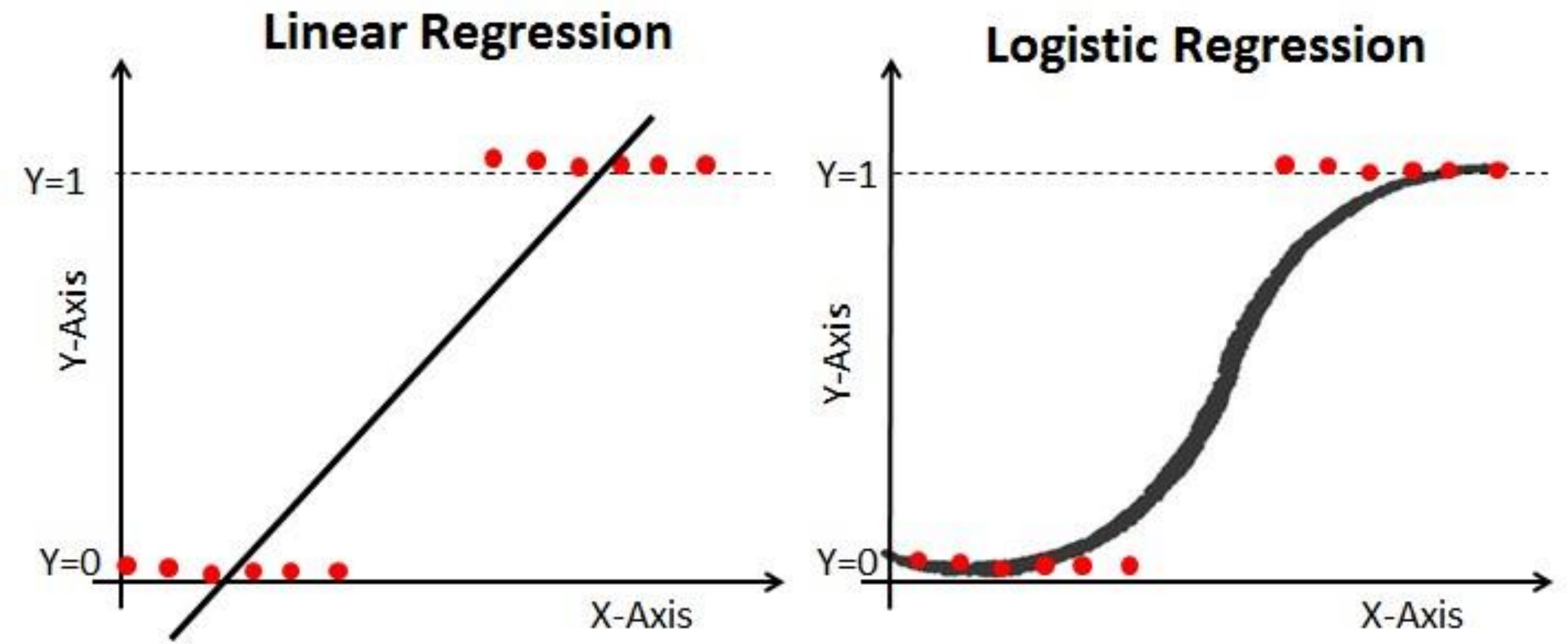


Source

Build model

For classification problem

- Linear model – logistic regression

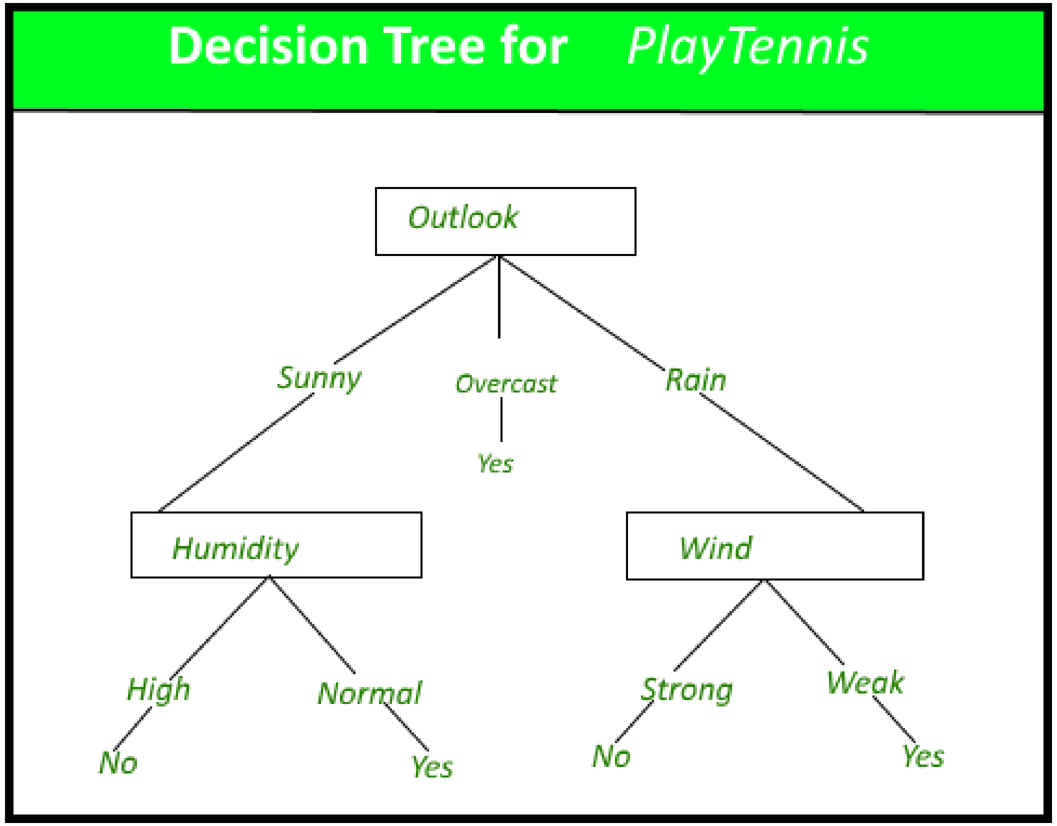


[Source]

Build model

For regression / classification problem

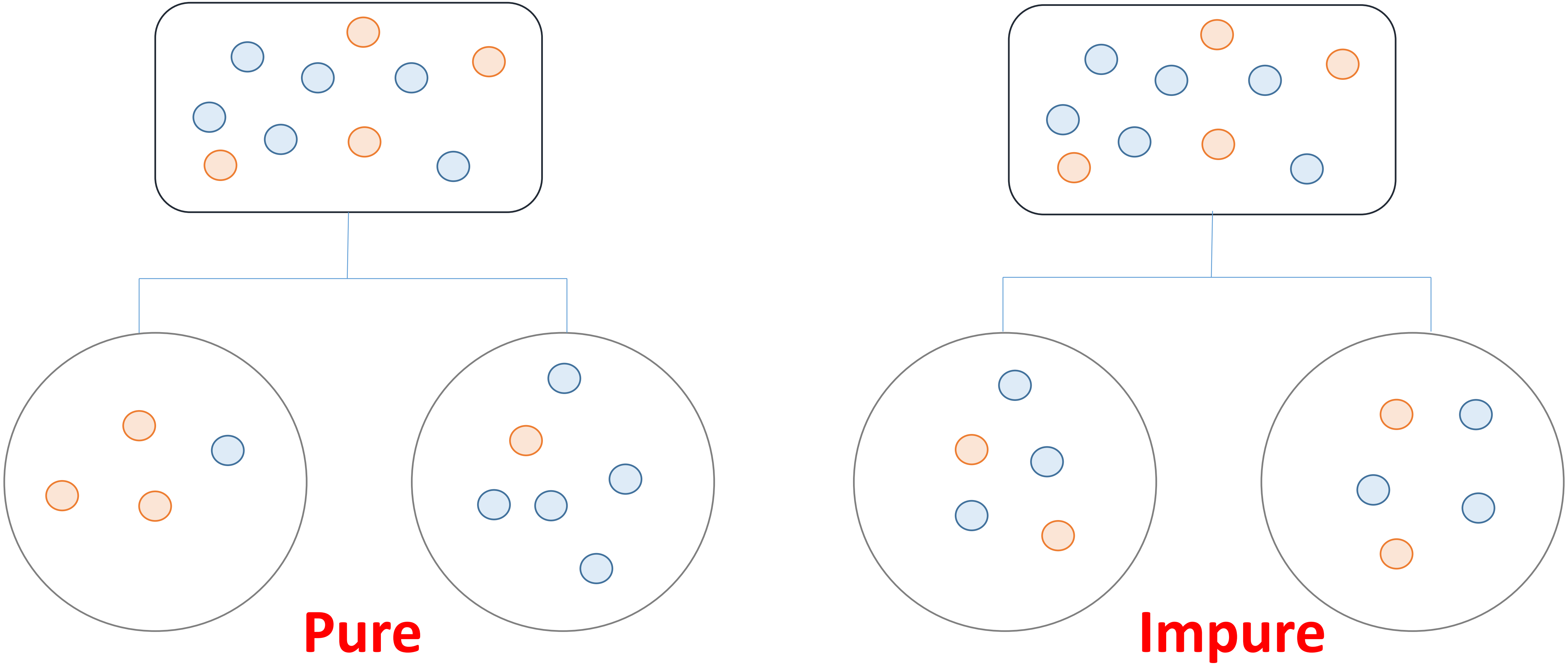
- Tree based model
  - Decision tree



Source

Build model

For regression / classification problem



Machine learning workflow

1. Problem definition
2. Data collection
3. Data exploration / preprocessing
4. Build model
5. Model evaluation

[Cheat sheet – scikit learn]

Model evaluation

Regression problem

Model evaluation

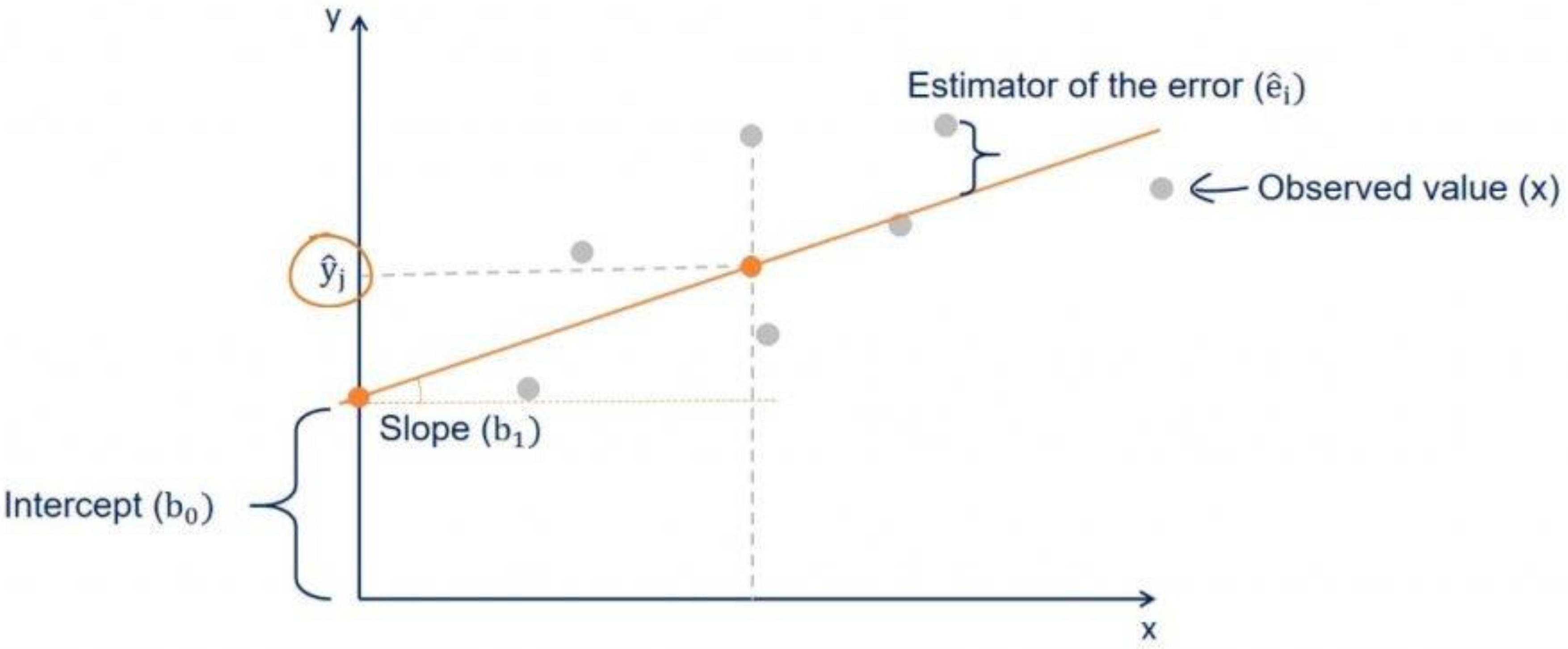
```
from sklearn.metrics import mean_squared_error,
mean_absolute_error, r2_score
```

- Regression problem
  - Mean Squared Error, MSE
  - Mean Absolute Error, MAE
  - R – square, R^2

Model evaluation

Linear regression model. Geometrical representation

$$\hat{y}_i = b_0 + b_1 x_i$$



Source

Model evaluation

$$MSE = \frac{1}{n}\sum\underbrace{\left(y-\hat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

Divide by the total number of data points

Predicted output value

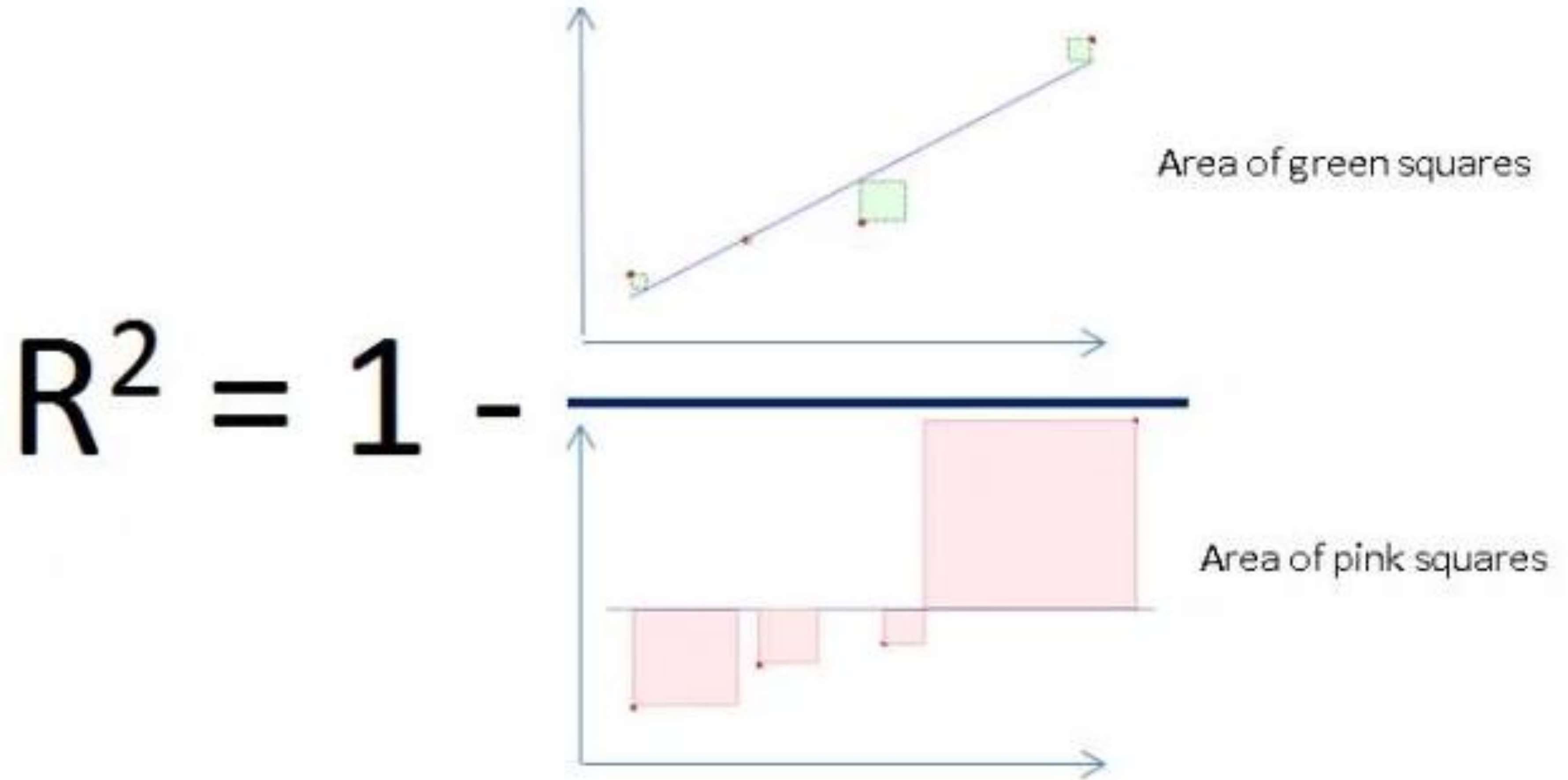
Actual output value

$$MAE = \boxed{\frac{1}{n}}\underbrace{\sum}_{\text{Sum of}}\underbrace{\left|\boxed{y}-\boxed{\hat{y}}\right|}_{\text{The absolute value of the residual}}$$

[Source]

Model evaluation

Residual Variance



Total Variance

Source

Time for practice

Build your first machine learning model with python

Model evaluation

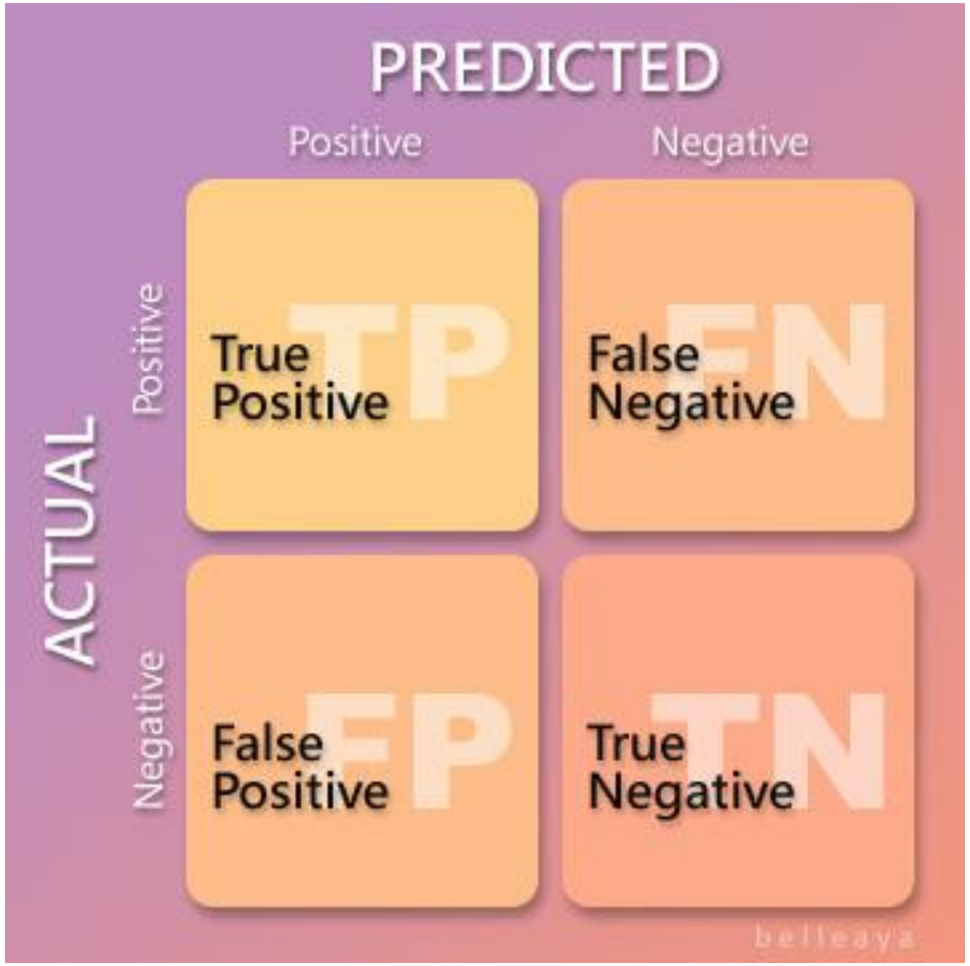
Classification problem

Model evaluation

```
from sklearn.metrics import confusion_matrix, accuracy_score,
precision_score, recall_score
```

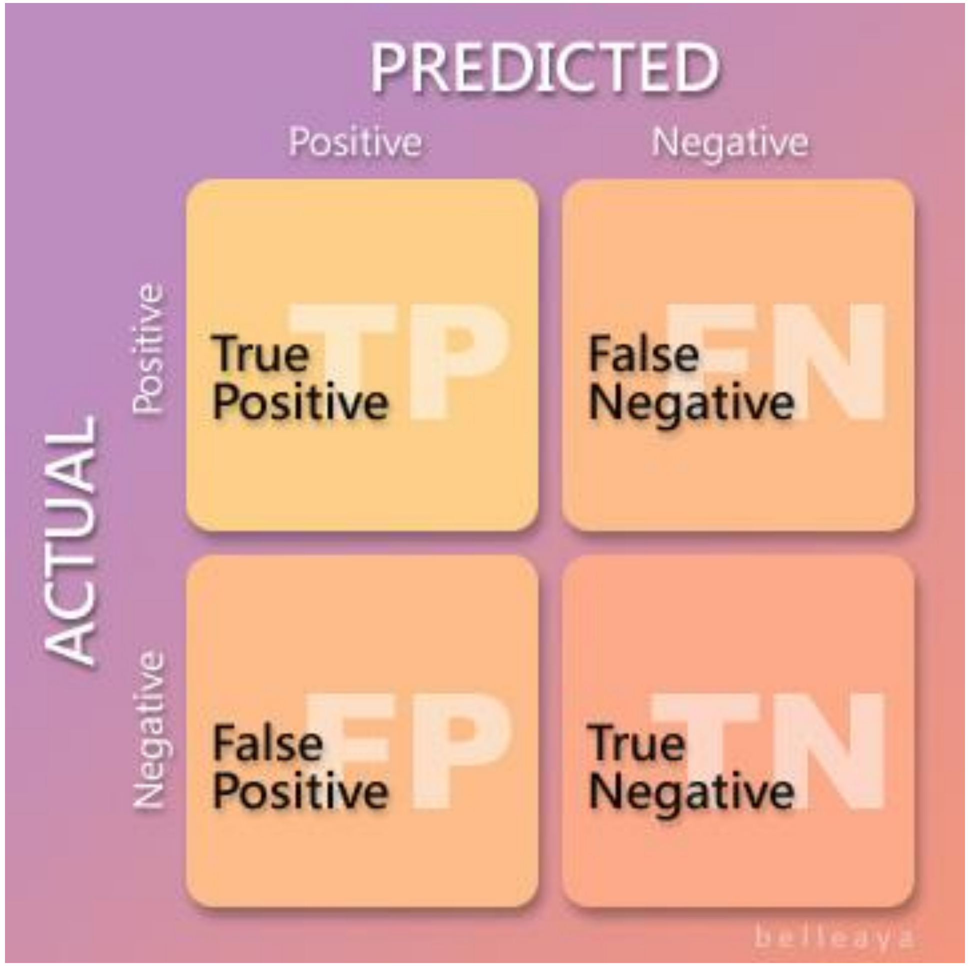
- Classification problem
    - Confusion matrix
    - Accuracy
    - Precision, recall

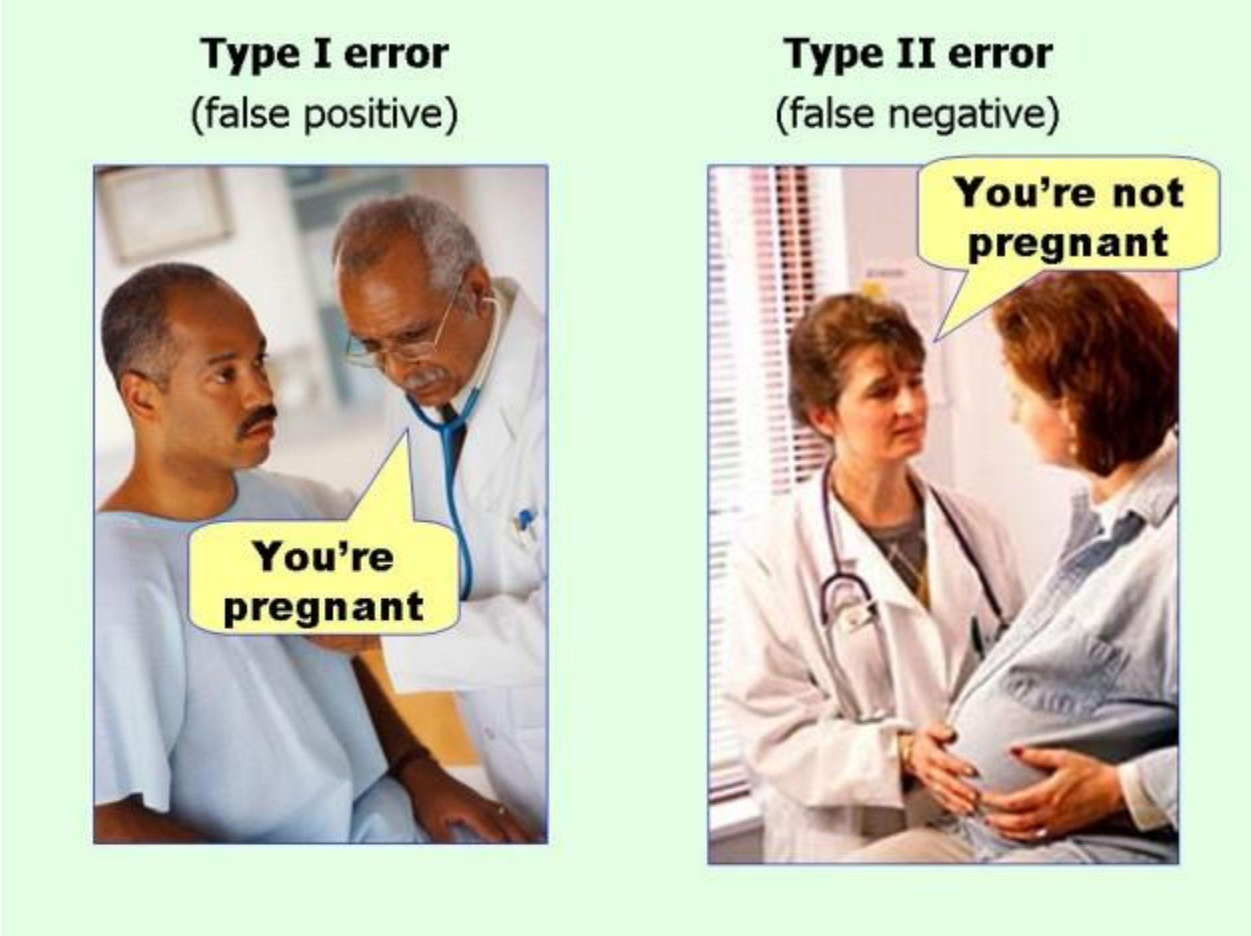
Model evaluation



Source

Model evaluation





Source

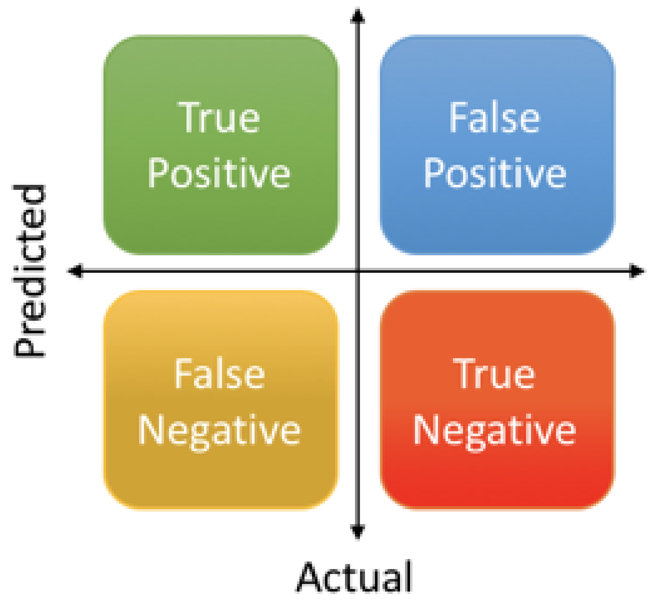
Model evaluation

- Precision & recall

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

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

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



Source

Time for practice

Build your first machine learning model with python

Summary

1. Types of machine learning

2. Machine learning workflow

3. Machine learning model with python

Advanced topics about ML

Feature engineering

- Generate new feature
  - Domain knowhow
  - Data exploration

Feature selection

- Feature selection
  - Correlation
  - Lasso, Ridge regression
  - Index of feature importance

Model selection

- Linear model
  - Focus on global information
  - Have data hypothesis

Encoding matters

Normalization is needed

- Tree-based model
  - Clear rules provided by model
  - Focus on local information

No need to normalize data

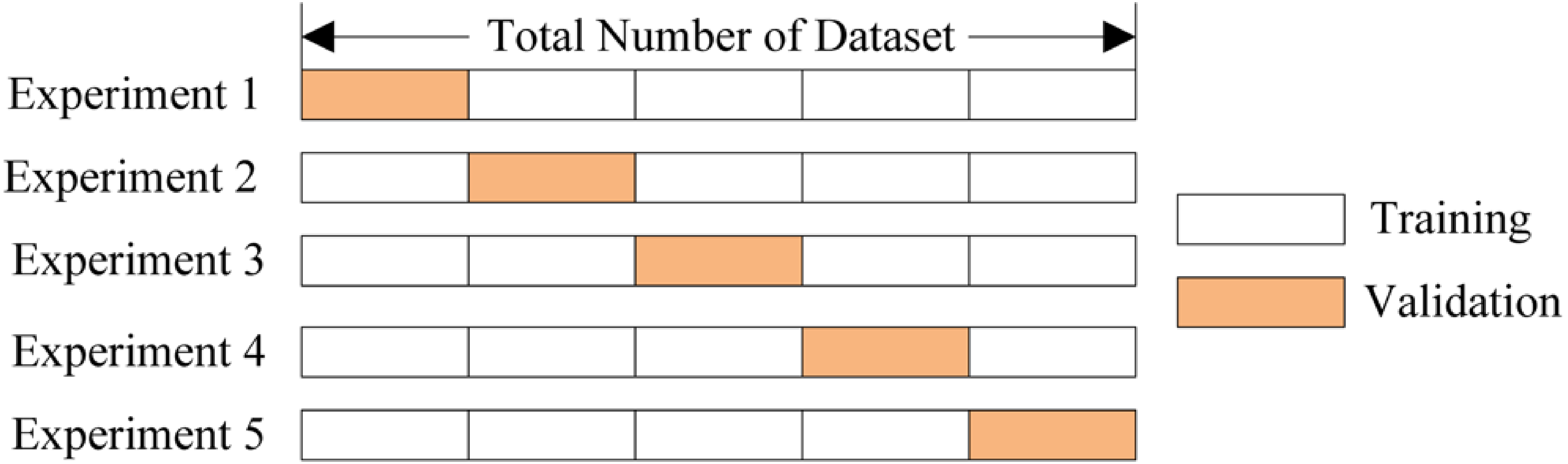
Other ML models

- Other machine learning models
  - Support Vector Machine
  - K-Nearest Neighbor
  - Naïve-bayes
  - Neural network

Other ML models

- Ml models that usually showed on ml competitions
  - Bagging : Random Forest
  - Boosting : XGBoost, LightGBM, CatBoost
  - Neural Network
  - Stacking model

Cross validation



Source