DATALEARNS24+7

Machine Learning
with PySpark

Big Data AI Engineer Training

Chapter 02

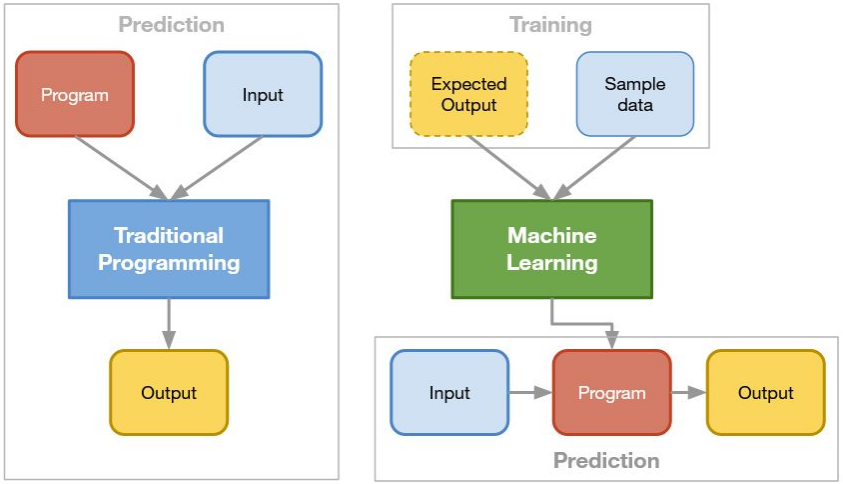
Machine Learning

What is Machine Learning?

The field of study that gives computers the ability to learn without being explicitly programmed.
— ***Arthur Samuel, 1959***

An approach to achieve ***artificial intelligence*** through system that can **learn** from experience to **find patterns** in a **set of data**
— ***Jason Mayes***

Bagian dari AI yang mempelajari kumpulan algoritma yang dapat belajar dari data, dan membuat prediksi terhadap data yang baru

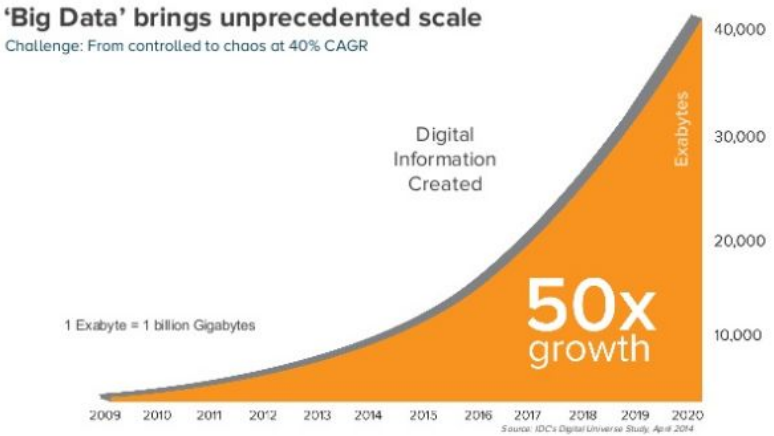


Mengapa Machine Learning Berkembang Pesat?

- Peningkatan ketersediaan data : jumlah dan label
- Peningkatan kemampuan komputasi : cloud, GPU, TPU, dll
- Peningkatan mutu teknik dan algoritma
- Dukungan industri : perusahaan besar maupun startup
- Komunitas riset dan open source yang aktif
- Penerapan dan dampak yang luas

"The amount of data created over the next three years will be more than the data created over the past 30 years, and the world will create more than three times the data over the next five years than it did in the previous five. "

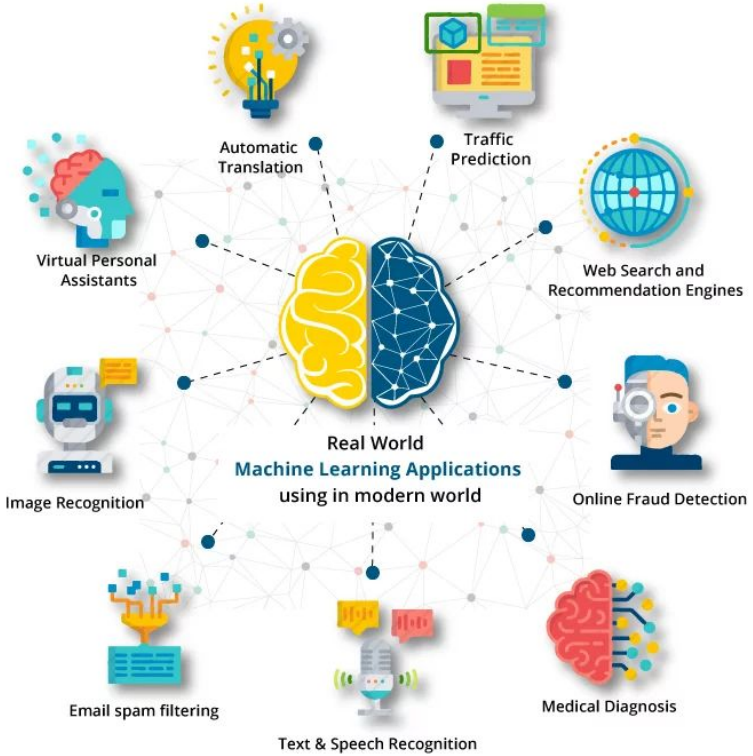
IDC Global DataSphere - May 2020



Kapan Machine Learning Dibutuhkan?

- Data bervolume besar
- Permasalahan kompleks
- Lingkungan dengan pola yang dinamis atau tidak jelas
- Personalisasi
- Otomasi pekerjaan : misal pekerjaan yang berulang-ulang, atau yang beresiko tinggi untuk manusia
- Mencari pola tersembunyi
- Mendukung pengambilan keputusan berbasis data

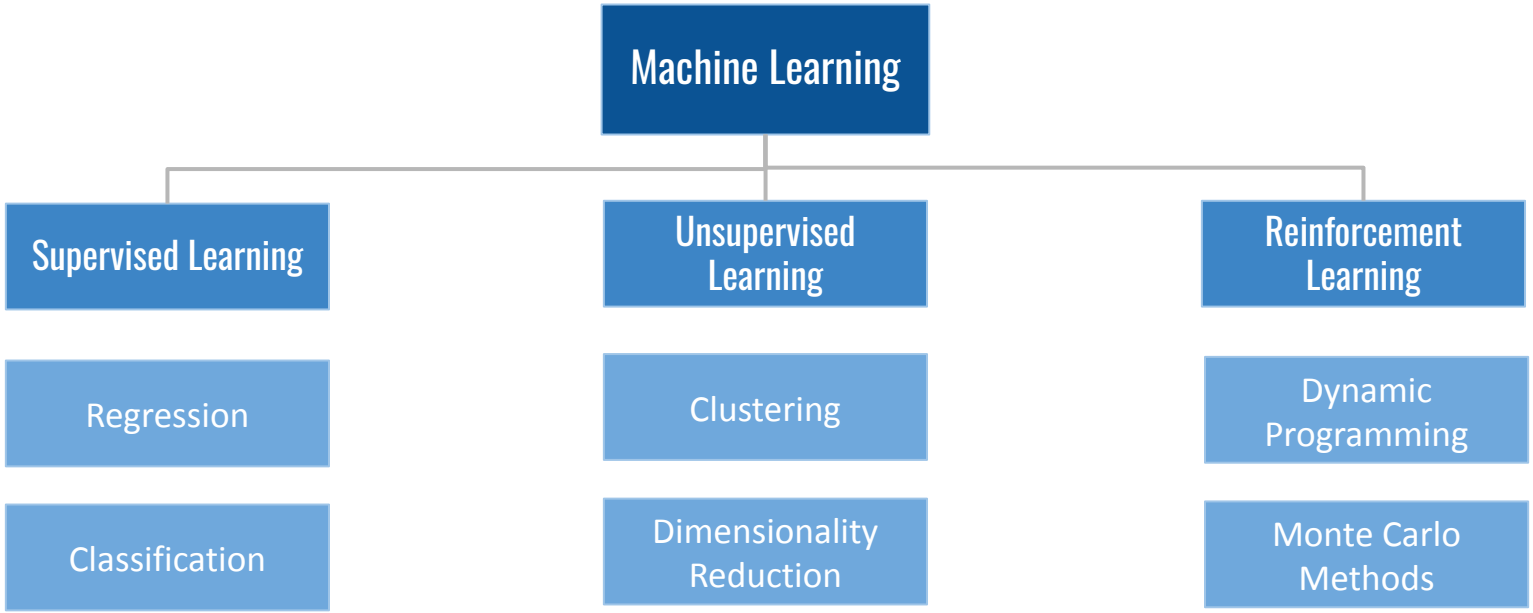
Pemanfaatan Machine Learning



Kapan Machine Learning Tidak Diperlukan?

- Data terbatas : jumlah dan kualitas
- Masalah sederhana
- Memerlukan pemahaman dan penilaian mendalam
- Resiko bias dan etika
- Pertimbangan biaya dan sumber daya vs. manfaat

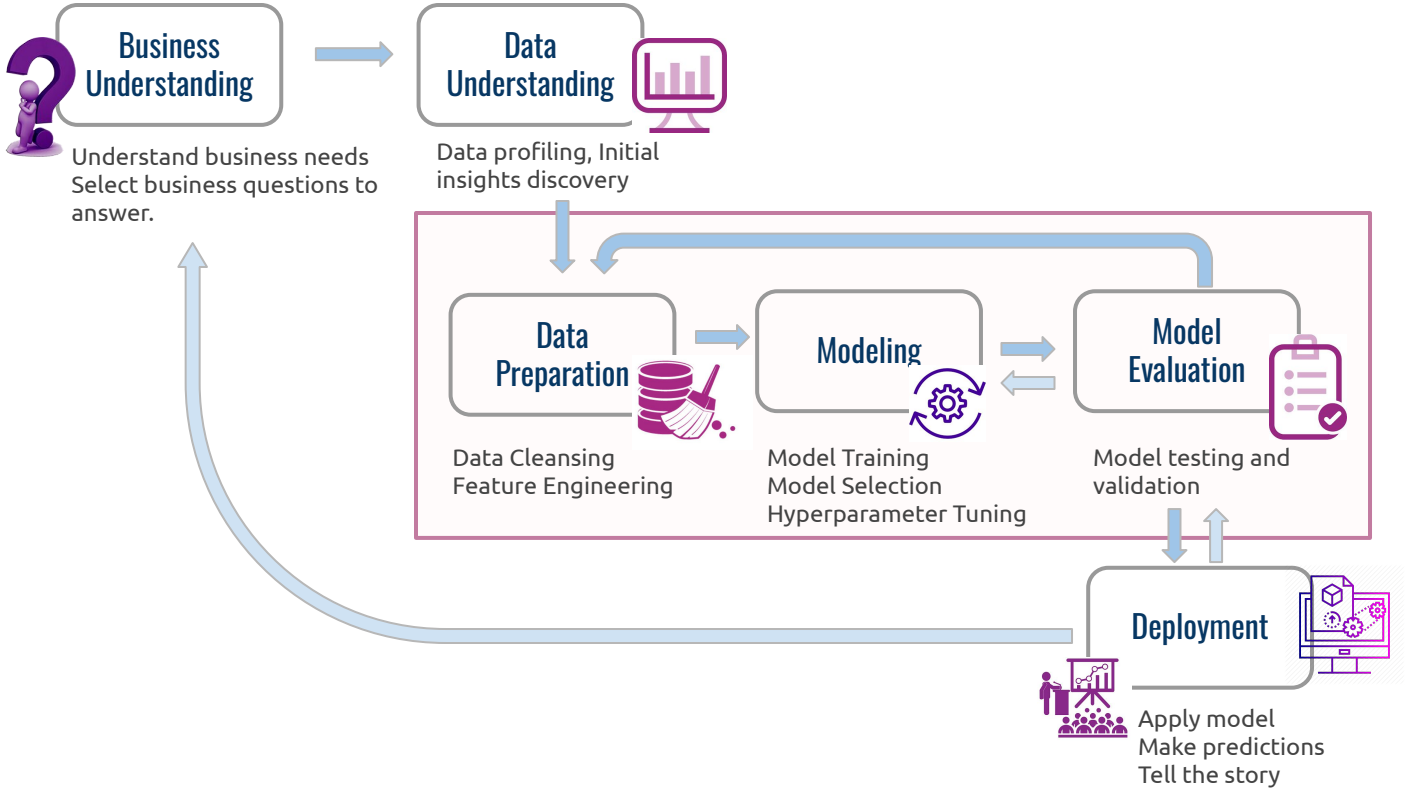
Kategori Algoritma Machine Learning



Kategori Algoritma Machine Learning

	Supervised	Unsupervised	Reinforcement
Training input	Data berlabel	Data tidak berlabel	Menggunakan interaksi agen dengan lingkungan untuk mendapatkan feedback dalam bentuk reward atau penalti.
Tujuan	Memprediksi atau mengklasifikasikan data baru berdasarkan pola yang dipelajari dari data	Mengidentifikasi struktur atau pola yang tersembunyi dalam data tanpa panduan label.	Menemukan strategi optimal untuk menyelesaikan pekerjaan dalam lingkungan yang kompleks.
Contoh penerapan	Klasifikasi email spam, prediksi harga saham, pengenalan wajah	Pengelompokan pelanggan, deteksi fraud, analisis jaringan sosial.	AlphaGo, DeepMind, self-driving car
Contoh algoritma	Regresi linier, decision tree, svm	K-means clustering, PCA	Q-Learning, Deep Q-Networks (DQN)

Machine Learning Pipeline



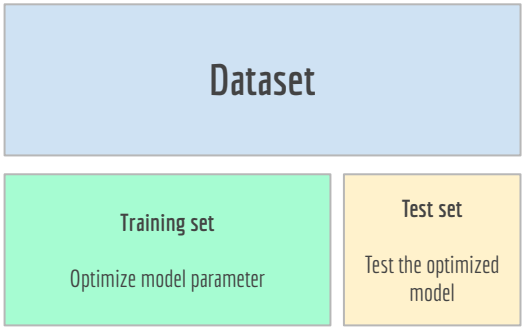
Instance, Features dan Labels

- Instance: sebuah baris dalam data, disebut juga *observation*
- Feature: sebuah kolom dalam data, disebut juga *attribute* atau *variable*
  - predictors atau input variable: features yang digunakan untuk melakukan prediksi
  - label : output dari model, features yang akan diprediksi

sepalLength	sepalWidth	petalLength	petalWidth	species
5.3	3.7	1.5	0.2	setosa
6.0	3.4	4.5	1.6	versicolor
6.5	3.0	5.5	1.8	virginica
6.5	3.2	5.1	2.0	virginica

Model Evaluation : Training & Testing Set

- Inti machine learning adalah **generalisasi** → sebaik apa kinerja sebuah model dalam menangani **unseen data**
- Untuk mensimulasikan unseen data → split data menjadi training dan testing set
- Data training dan testing hendaknya :
  - Cukup besar → rasio yg umum 80:20 atau 70:30
  - Representatif (mencakup semua kasus dalam dataset) → gunakan metode split yang sesuai (randomized, stratify, etc.)



Chapter 03

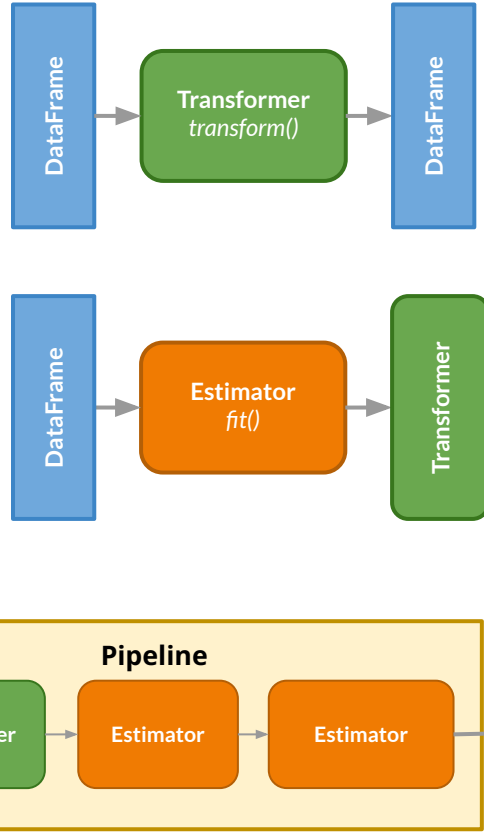
Sekilas Spark MlLib

Introduction

- MLlib adalah pustaka machine learning Spark
- Mulai Spark 2.0 MlLib menggunakan DataFrame API
- Dokumentasi MLlib dapat dilihat di https://spark.apache.org/docs/latest/ml-guide.html

Machine Learning Pipelines

- MLlib menstandarisasi API, memudahkan untuk menggabungkan berbagai algoritma dalam sebuah workflow/pipeline
- Komponen utama MlLib :
  - **Transformer**: algoritma yang mentransformasi sebuah *DataFrame* menjadi *DataFrame* yang lain
  - **Estimator**: algoritma yang melakukan **fitting** sebuah *DataFrame*, untuk menghasilkan *Transformer*
  - **Pipeline** merangkai beberapa Transformer dan Estimator untuk membentuk sebuah workflow



Data Representation (Feature Encoding)

- Model machine learning memerlukan input dan menghasilkan output numerik
- Data kategorik harus di-encode menjadi angka sebelum dimasukkan ke dalam model
- Beberapa metode encoding : one-hot encoding, dummy encoding, hash encoding, learned embedding, etc.
- Yang paling populer dan akan kita gunakan dalam pelatihan ini adalah : label, one-hot, dan dummy encoding

features	target
[0,1, 2, .12]	1
[1, 2, 4, .22]	0
[0, 1, 1, .32]	2
[3, 1, 2, .01]	4
[5, 2, 0, .31]	1
[2, 1, 2, .45]	2
[0, 1, 0, .6]	3
[3, 1, 2, .2]	4

Data structure for MLlib : features vectors and target vector

One Hot and Dummy Encoding

- **One hot encoding** : konversi fitur kategorik menjadi vektor biner dengan salah satu kolom bernilai 1. Fitur dengan N jenis (kardinalitas N) diwakili oleh N kolom
- **Dummy encoding** : mirip dengan one-hot, namun menggunakan N-1 kolom untuk N jenis

name	qty
apple	5
banana	6
cherry	7
banana	8
apple	9

One-Hot Encoded

var1	var2	var3	qty
1	0	0	5
0	1	0	6
0	0	1	7
0	1	0	8
1	0	0	9

Dummy Encoded

var1	var2	qty
1	0	5
0	1	6
0	0	7
0	1	8
1	0	9

name	var1	var2	var3
apple	1	0	0
banana	0	1	0
cherry	0	0	1

name	var1	var2
apple	1	0
banana	0	1
cherry	0	0

Hands-On 01

MlLib Introduction

MLlib Basics

In this lab we will learn about :

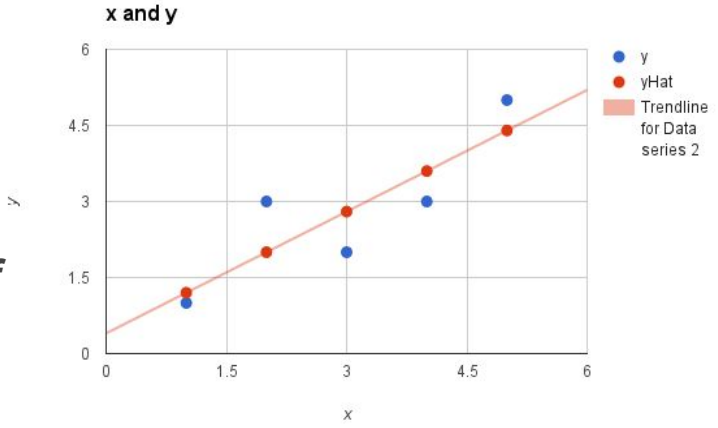
- The basic of MLlib
- How to perform feature engineering using some of MLlib transformers :
  a. StringIndexer
  b. OneHotEncoder
  c. VectorAssembler
- How to use Pipeline

Chapter 04

Linear Regression

Introduction

- Mencari pola dalam data dan menggunakannya untuk membuat prediksi
- Memprediksi nilai numerik kontinyu
- Simple linear regression : ***fitting a line into a set of data***
- Use cases: prediksi harga barang, sales and demand forecasting, prediksi harga saham, dll.



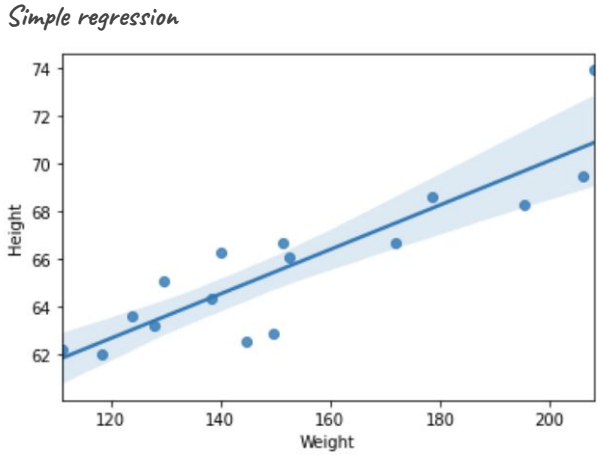
Types of Linear Regression

- Simple / univariate linear regression : 1 variabel input untuk memprediksi 1 variabel output

$$y = w_0 + w_1 x$$

- Multiple / multivariate linear regression : lebih dari 1 variabel input untuk memprediksi 1 variabel output

$$y = w_0 + w_1 x_1 + w_2 x_2 + .. + w_n x_n$$

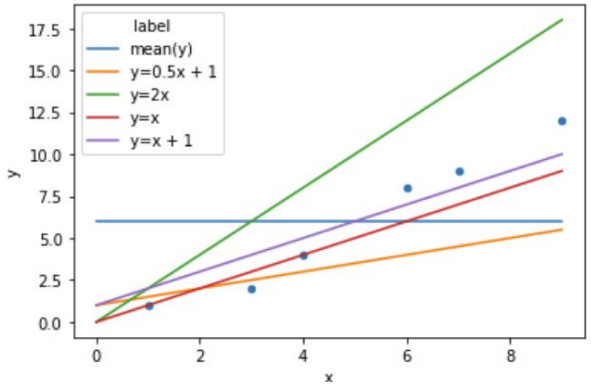


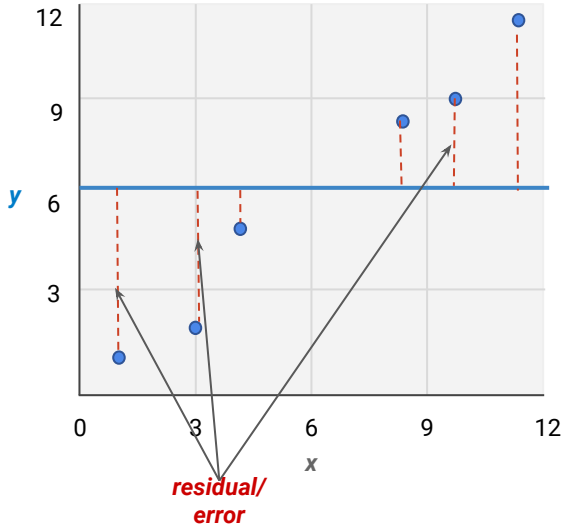
The Best Line

- Garis/persamaan dengan **jumlah error terkecil**
- **Error / residual** : selisih antara hasil prediksi dengan nilai sebenarnya → jarak antara titik data dengan garis

$$e = y - \hat{y}$$

- Sum of Squared Error (SSE) : $\mathrm{SSE} = \sum_i^n (y_i - \hat{y}_i)^2$
- Mean of Squared Error (MSE) : $\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$





Evaluasi Model Regresi

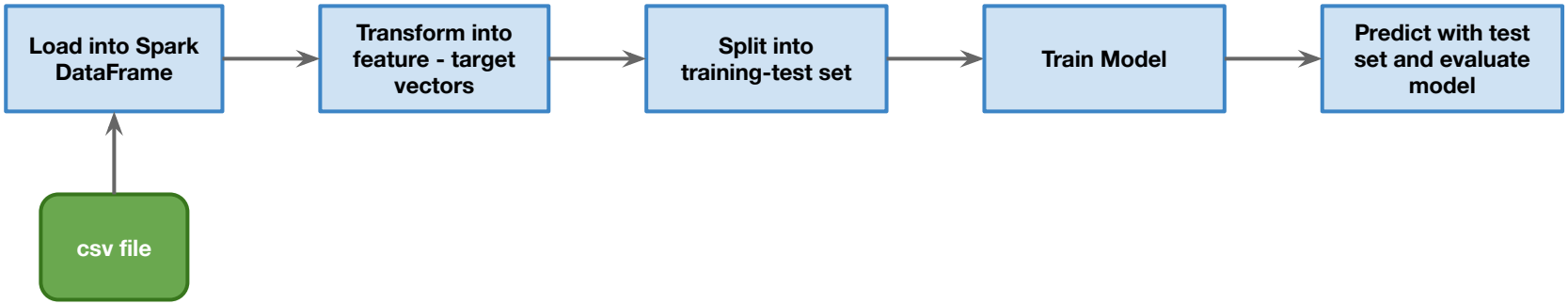
- Formula untuk to mengukur error :
  - R^2 (R-Squared) $R^2 = 1 - \frac{SSE}{SST}$
  - MAE (Mean Absolute Error) $MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$
  - RMSE (Root Mean Squared Error) $RMSE = \sqrt{\sum_{i=1}^{n}\frac{(\hat{y}_i - y_i)^2}{n}}$

Hands-On 02

Linear Regression

Linear Regression in MlLib

- Learn to train and test MlLib Linear Regression model
- We will try to process a very simple linear regression. First with 1 numeric input variable, and with a numeric and categorical input variable.

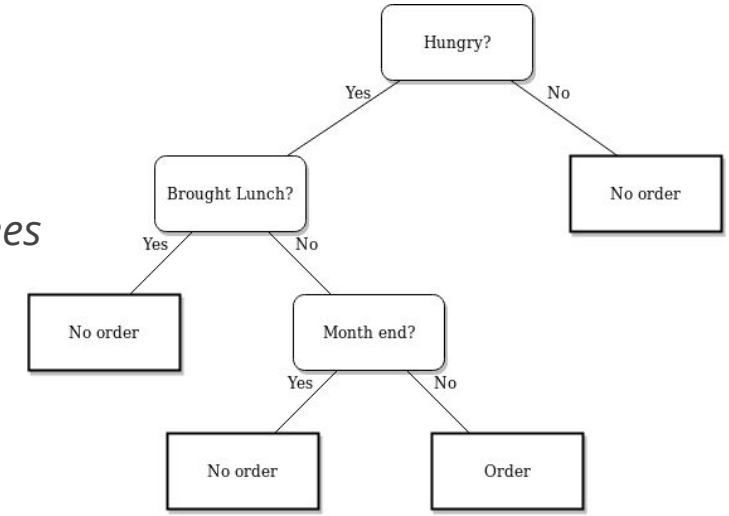


Chapter 05

Decision Tree

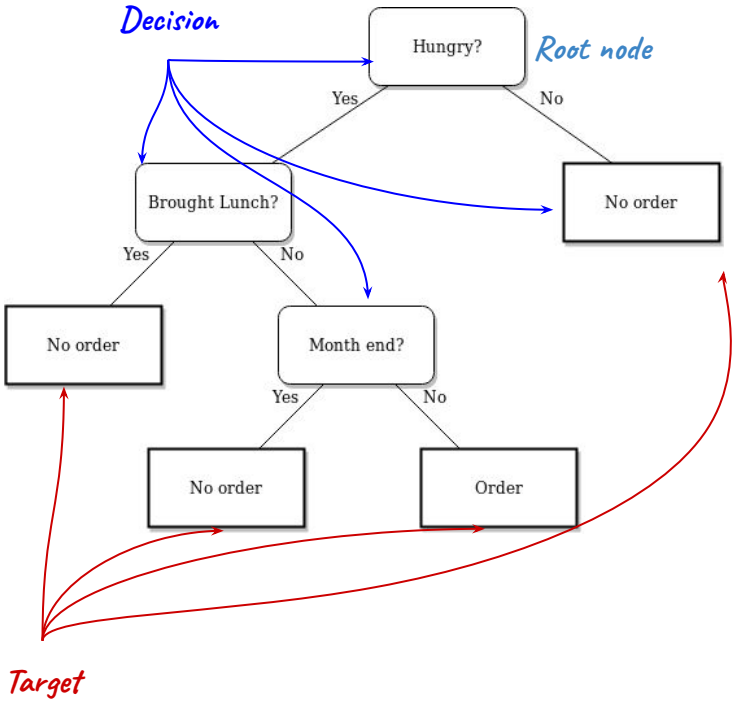
Decision Tree : Introduction

- Metode supervised learning untuk **klasifikasi** dan **regresi**
- Mempelajari aturan keputusan sederhana dari dataset
- Merupakan dasar untuk algoritma-algoritma lain, misalnya *Random Forests* dan *Boosted Decision Trees* semacam as *XGBoost*
- Kelebihan :
  - Mudah difahami dan divisualisasikan
  - Dapat menangani prediksi numerik maupun kategorik (regresi dan klasifikasi)



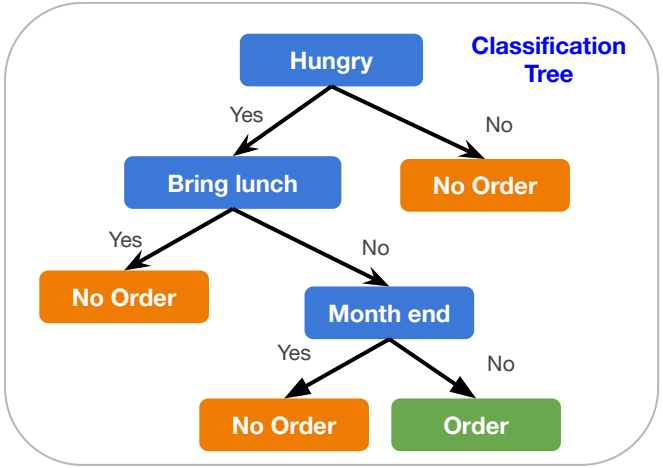
Decision Tree : Terminology

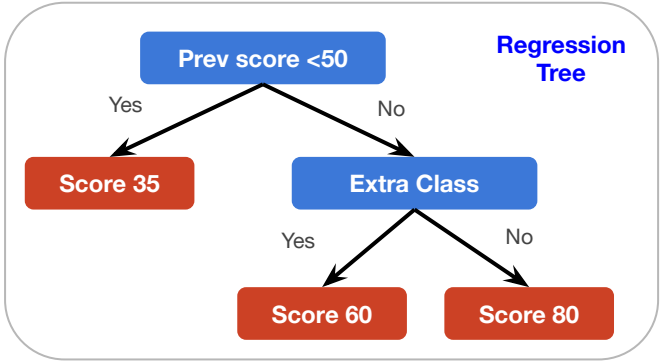
- **Root node** : node awal
- **Leaf node** : node akhir
- **Internal** atau **intermediate node** : nodes selain root dan leaf
- **Splitting** : proses memecah sebuah node
- **Pruning** : kebalikan splitting, yaitu membuang beberapa aturan untuk mengurangi kompleksitas
- Proses keputusan ada di root dan internal node



Classification vs Regression Tree

	Classification Tree	Regression Tree
Target variable	Categorical	Numerical
Prediction	Mode	Mean
Cost function	Gini score, information gain	SSE

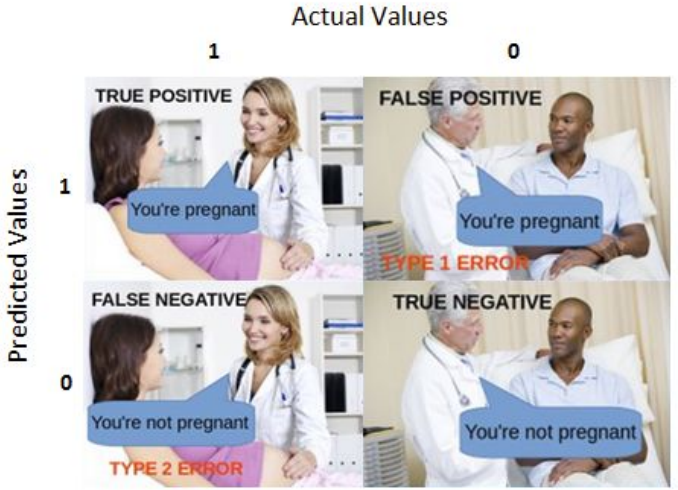




Evaluation : Confusion Matrix

- We can evaluate classification result using confusion matrix
- Some metrics derived from the matrix below are : accuracy, precision, recall, and F-measure

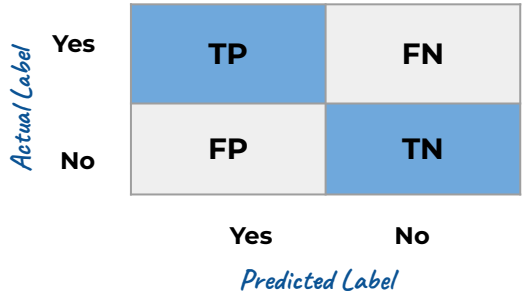
Actual Label \ Predicted Label	Yes	No
Yes	True Positive (**TP**)	False Negative (**FN**)
No	False Positive (**FP**)	True Negative (**TN**)



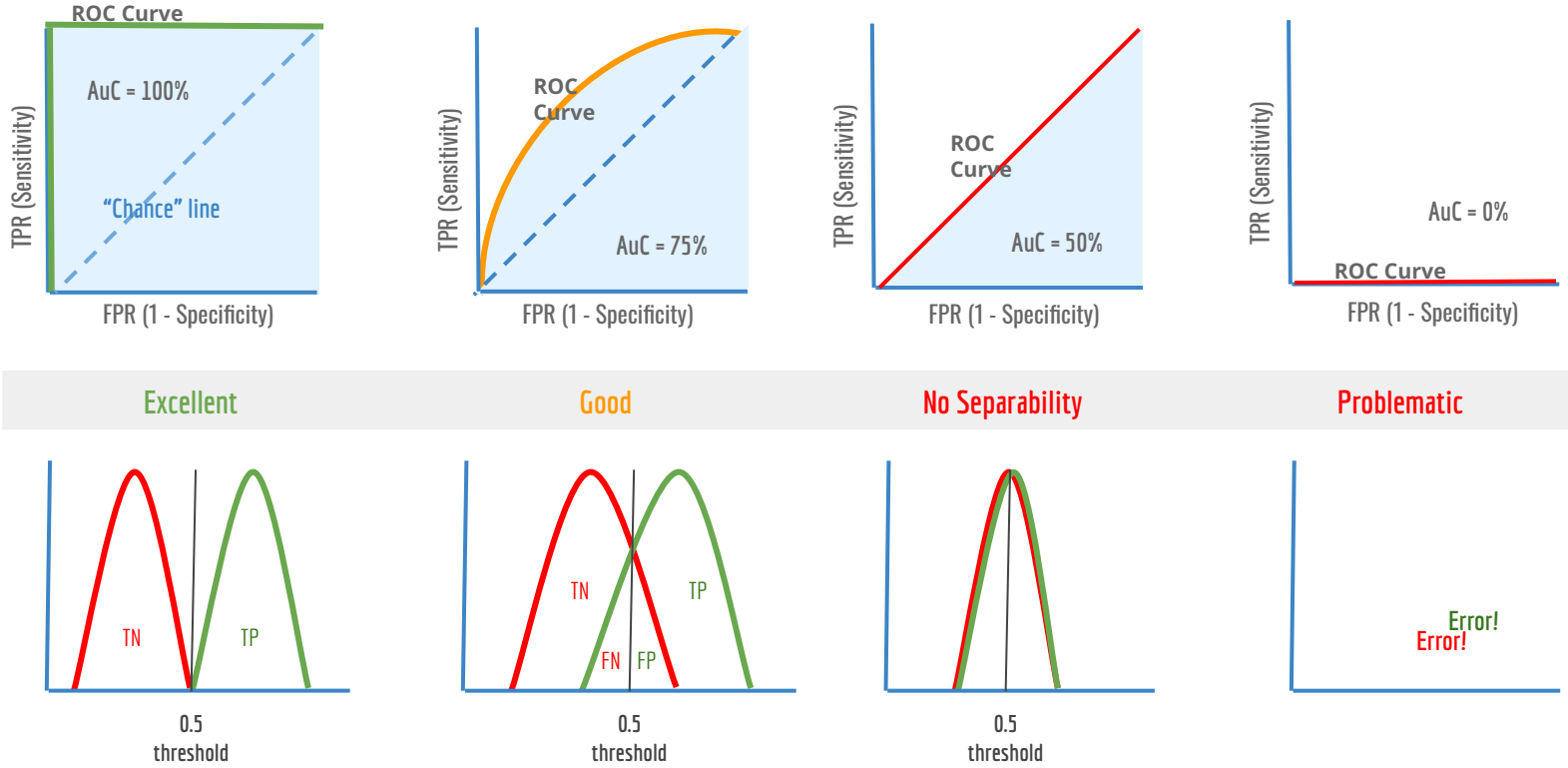
Pic from : https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Evaluation Metrics

Accuracy	Recall
Ratio of correctly predicted observation to the total observations $\frac{(TP + TN)}{\text{All Observations}}$	Ratio of correctly predicted positive observations to all actual positive observations $\frac{TP}{(TP + FN)}$
Precision	**F-Measure**
Ratio of correctly predicted positive observations to all predicted positive observations. $\frac{TP}{(TP + FP)}$	Provides a way to combine both precision and recall into a single measure $\frac{2 * (Precision * Recall)}{(Precision + Recall)}$



Evaluation : Area Under ROC Curve

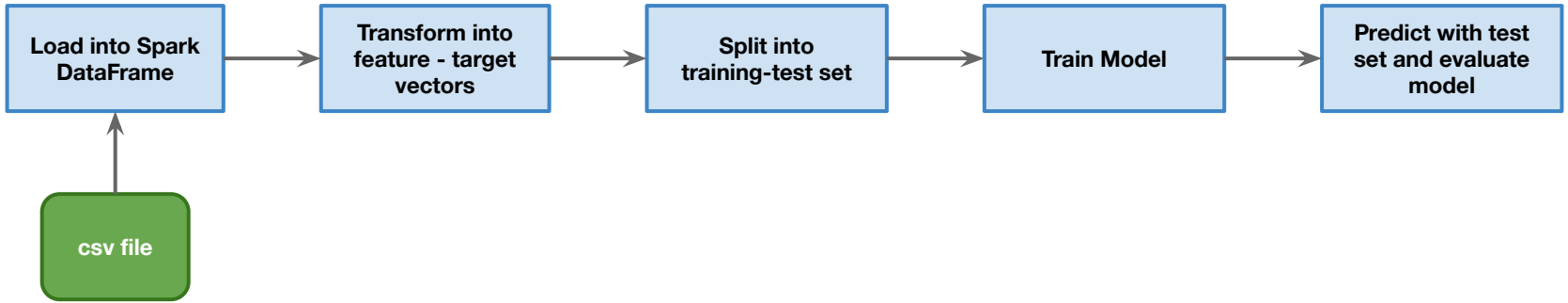


Hands-On 03

Decision Tree

Decision Tree in MlLib

Learn to train and test MlLib Decision Tree model



References

Next-Generation Machine Learning with Spark, *Butch Quinto*
Spark MlLib Online Documentation : https://spark.apache.org/docs/latest/ml-guide.html