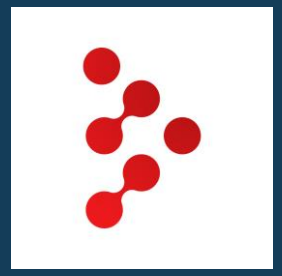


Intermediate Machine Learning

Review Linear & Logistic Regression

Objective & Outline

Objective & Outline

Objectives

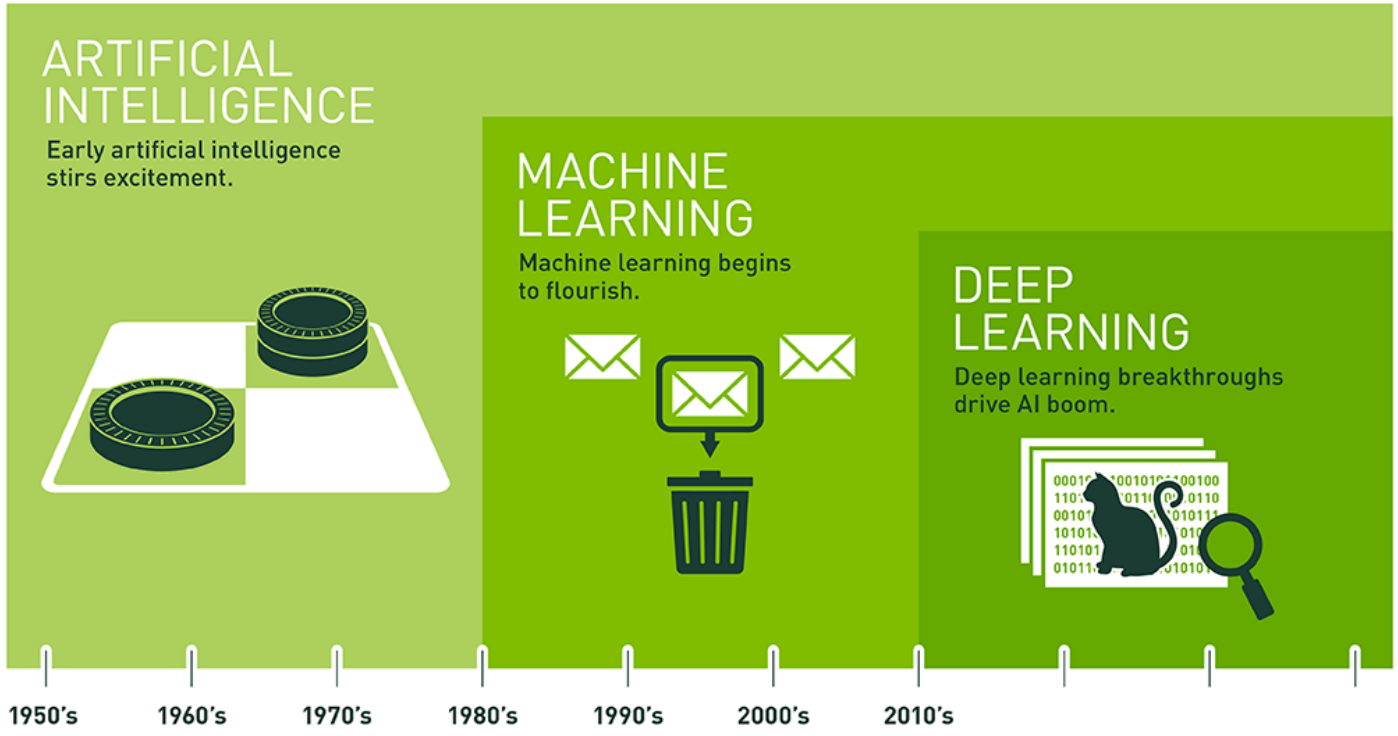
Memahami kembali konsep-konsep yang ada pada algoritma Linear Regression & Logistic Regression sebagai algoritma dasar di dunia Machine Learning

Outline

- AI, Machine Learning & Deep Learning
- Supervised and Unsupervised Learning
- Linear Regression Algorithm
- Logistic Regression Algorithm

AI, Machine Learning & Deep Learning

AI, Machine Learning & Deep Learning



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

AI, Machine Learning & Deep Learning

Machine Learning Algorithms

Support Vector Machine

Bayesian Network

Logistic Regression

Naive Bayes

K-Nearest Neighbour

Decision Tree

K-Means

Linear Regression

Artificial Neural Network

Perceptron

Random Forest

AI, Machine Learning & Deep Learning

Deep Learning Algorithms

Deep Fully Neural Network

Convolutional Neural Network

Recurrent Neural Network

YOLO

Deep Boltzmann Machine

Single-Shot Detector

Fully Convolutional Network

Deep Belief Network

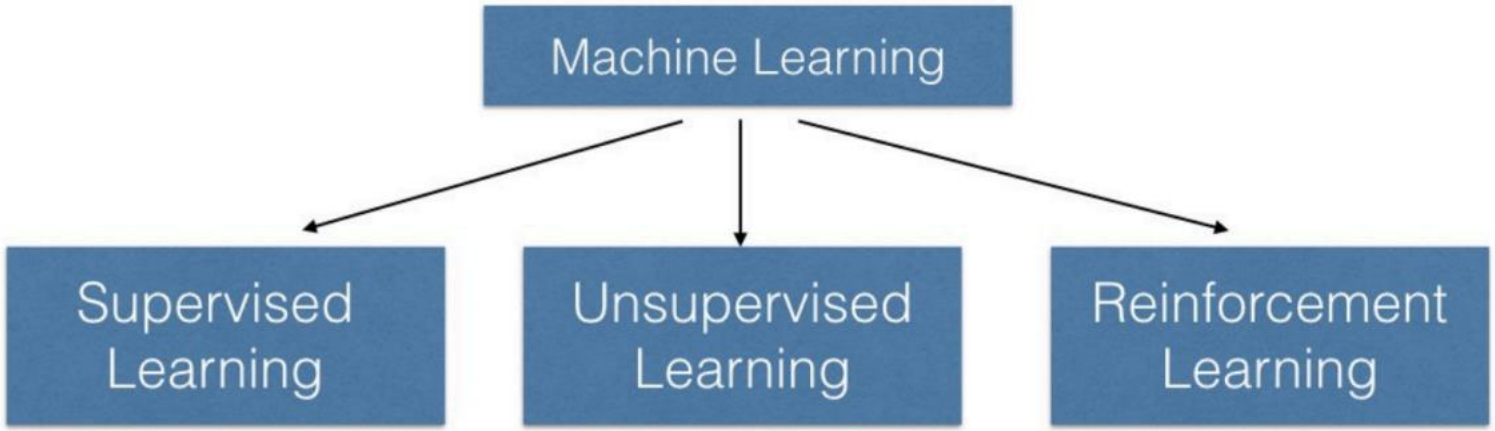
Deep Reinforcement Learning

Auto-Encoders

U-Net

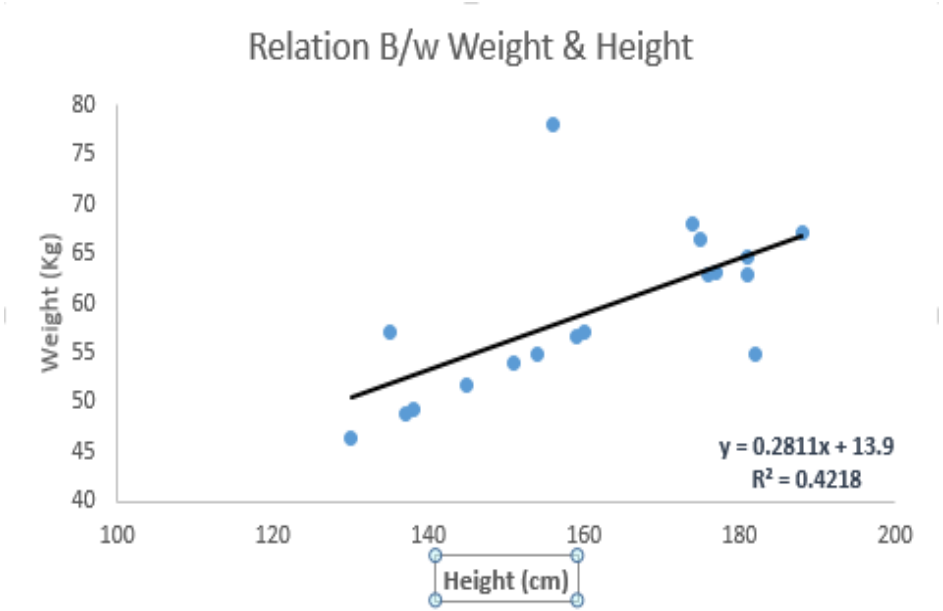
Supervised & Unsupervised Learning

Supervised and Unsupervised Learning



Linear Regression Algorithm

Linear Regression Algorithm

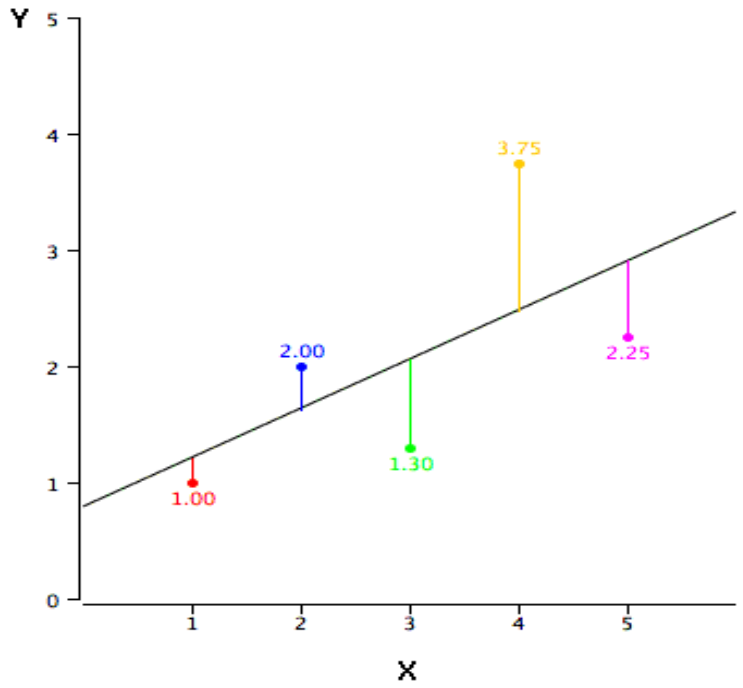


- Algoritma Linear Regression menjadi algoritma yang pertama kali dipilih oleh mereka untuk memahami algoritma Machine Learning

- Pada teknik ini:
  - Dependent Variable bersifat continuous
  - Memiliki garis regresi yang bersifat linear

- Pertanyaan penting yaitu *"Bagaimana mendapatkan the best fit line?"*

Linear Regression Algorithm

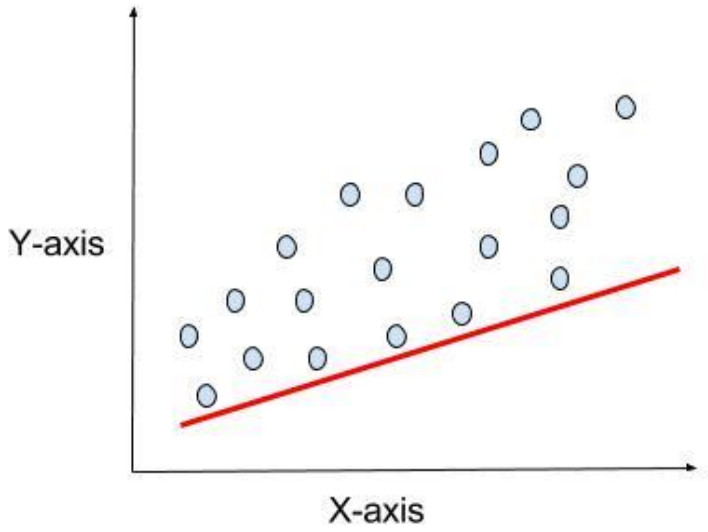


$y = a + bx$

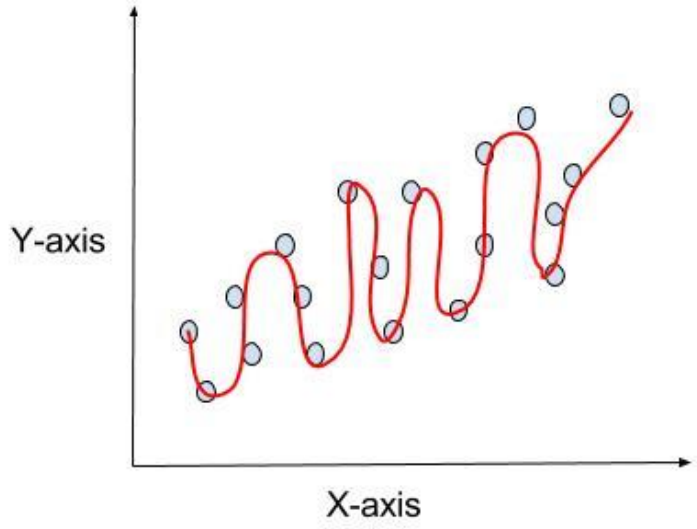
- Bagaimana mendapatkan the best fit line (nilai a dan b)?
- Untuk mendapatkan itu bisa menggunakan kalkulasi *Least Square Method* untuk meminimalisir nilai error
- Pada akhirnya the best fit line bisa dievaluasi menggunakan metric R-square

Linear Regression Algorithm

- The best fit line (model) yang baik akan menentukan posisi dengan kalkulasi error yang paling minimal
- No underfitting, no overfitting



Underfitting



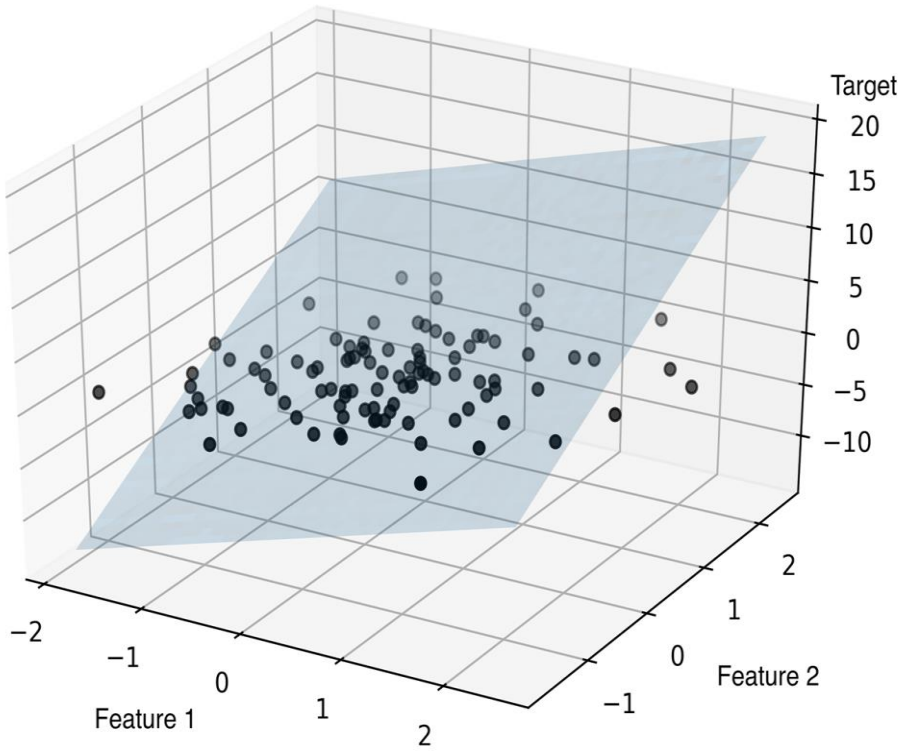
Overfitting

Linear Regression Algorithm

- **Simple linear regression**: Relasi *one-to-one* antara variabel input dan variabel output

- **Multiple linear regression**: Relasi *many-to-one*, tidak hanya menggunakan satu variabel input, melainkan menggunakan lebih dari satu input

$$y = w_0 x_0 + w_1 x_1 + \ldots + w_m x_m = \sum_{i=0}^{m} w_i x_i = w^T x$$



Linear Regression Algorithm

- **Fit_intercept** : boolean, optional, default True
  - Whether to calculate the intercept for this model. If set to False, no intercept will be used in calculations
- **Normalize** : boolean, optional, default False
  - This parameter is ignored when fit_intercept is set to False. If True, the regressors X will be normalized before regression by subtracting the mean and dividing by the l2-norm. If you wish to standardize, please use sklearn.preprocessing.StandardScaler before calling fit on an estimator with normalize=False

class `sklearn.linear_model.` **`LinearRegression`** (*fit_intercept=True*, *normalize=False*, *copy_X=True*, *n_jobs=1*)

Linear Regression Algorithm

- **Fit** : Estimates the best representative function for the the data points. With that representation, you can calculate new data points

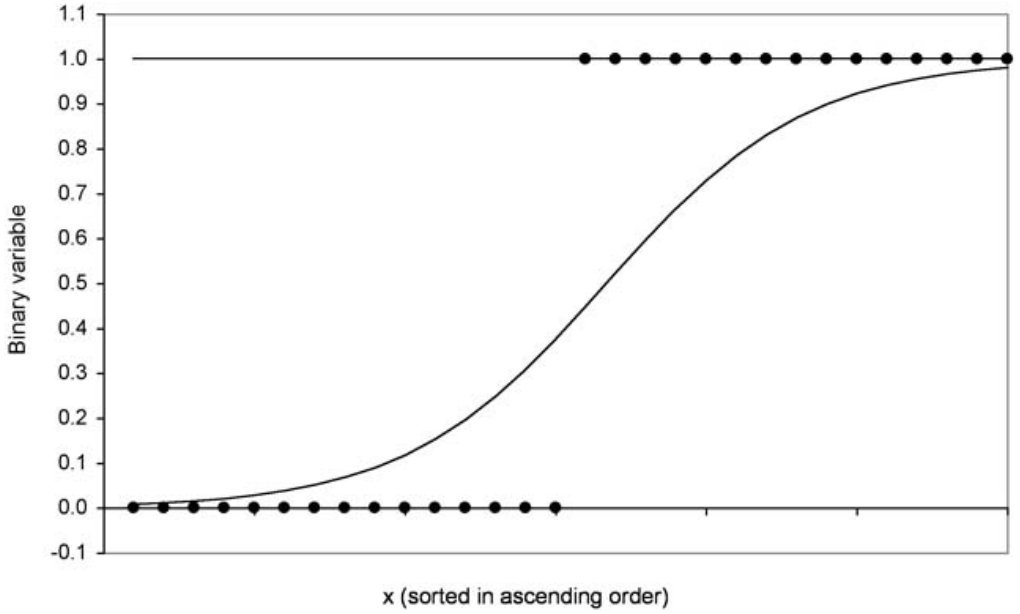
- **Predict** : Utilizing incoming data points to find the new output based on model representation from the fit method

- **Score:** Returns the coefficient of determination R^2 of the prediction.

class `sklearn.linear_model.` **`LinearRegression`** (*fit_intercept=True*, *normalize=False*, *copy_X=True*, *n_jobs=1*)

Logistic Regression Algorithm

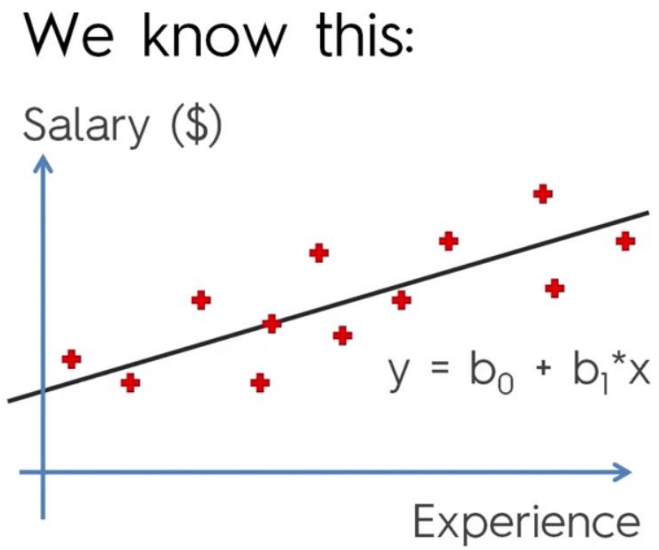
Logistic Regression Algorithm

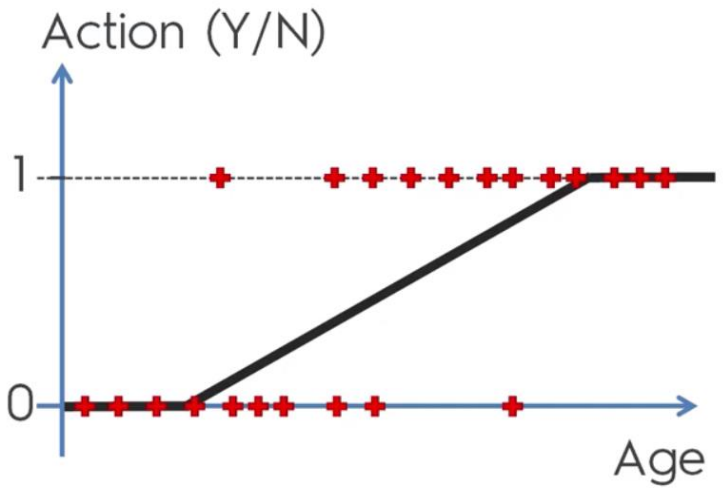


- Logistic Regression bukan termasuk algoritma regresi melainkan algoritma klasifikasi
- Logistic regression memprediksi probabilitas dari satu kejadian (*occurrences of event*) berdasarkan data yang ada dengan menentukan *logit function* menggunakan *sigmoid function*

Logistic Regression Algorithm

Kenapa tidak menggunakan Linear Regression saja?





Logistic Regression Algorithm

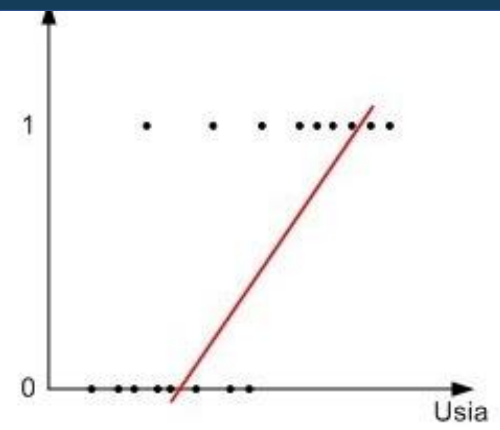
Dengan penggunaan sigmoid function pada algoritma logistic regression maka akan didapatkan probabilitas dari satu kejadian.

Think of it as probabilities!

Persamaan regresi sederhana $Y = a_0 + a_1 X$

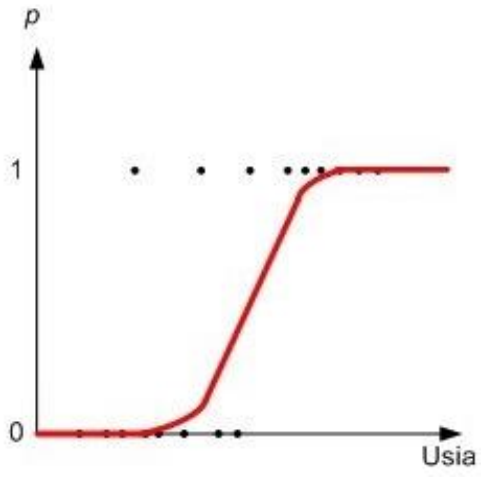
+

Fungsi sigmoid $p = \frac{1}{1 + e^{-y}}$

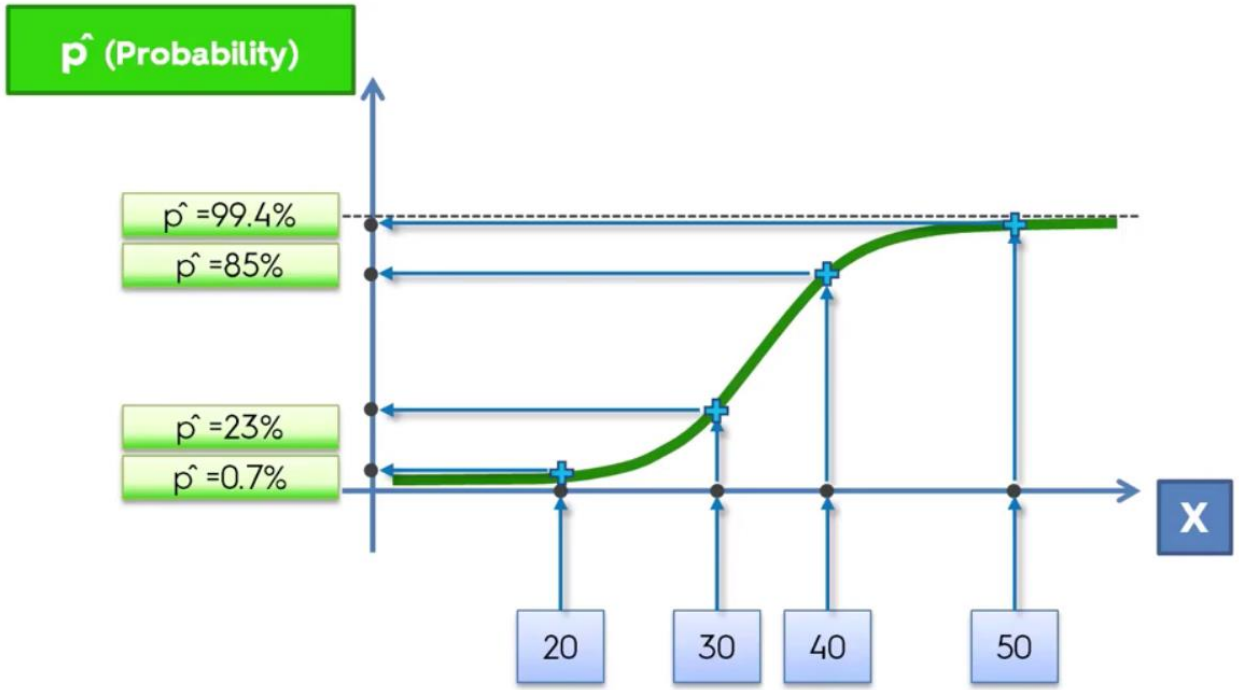


STEP 2

Persamaan regresi logistik $\ln\left(\frac{p}{1-p}\right) = a_0 + a_1 X$

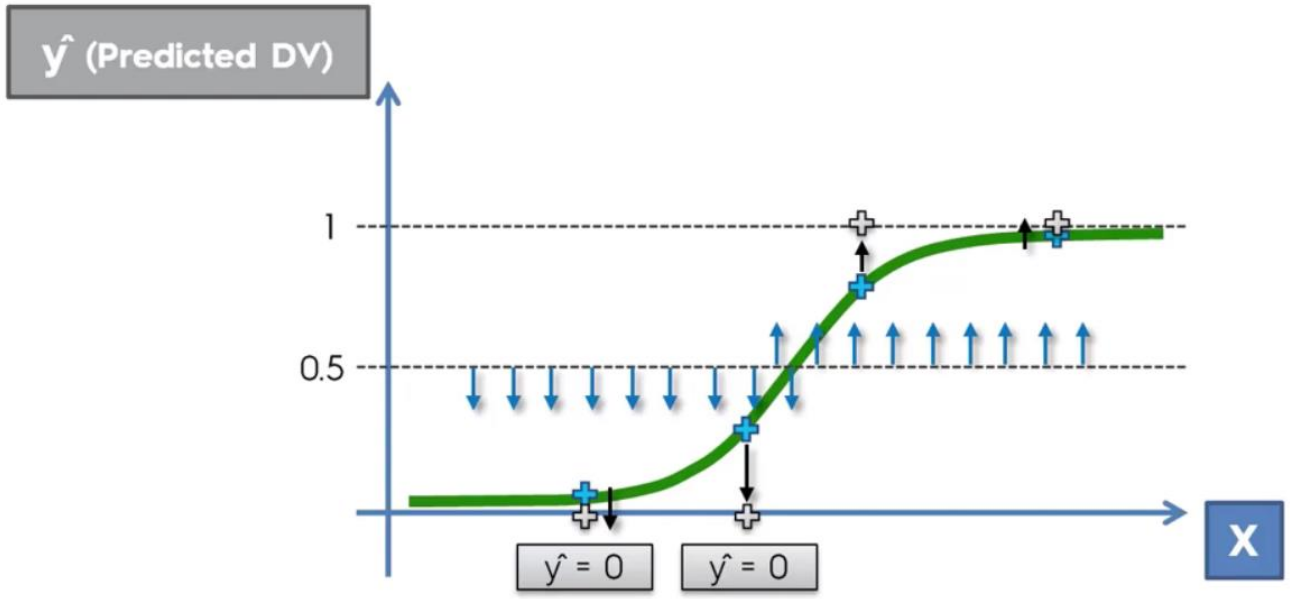


Logistic Regression Algorithm



Algoritma Logistic regression jika dilihat dari sudut pandang probabilitas kejadian

Logistic Regression Algorithm



- Threshold akan menentukan apakah data tertentu termasuk pada kelas A atau kelas B
- Teknik ini menjadi cikal bakal lahirnya konsep Deep Learning

Thanks!

Team Indonesia AI