Machine learning handbook

Classifier

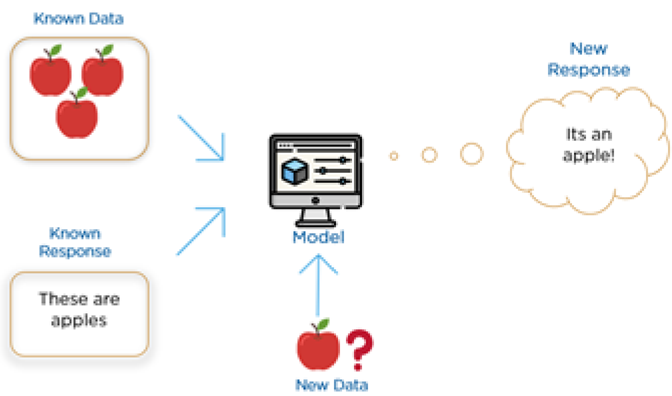
Taeyang Yang

Update: July 13, 2018

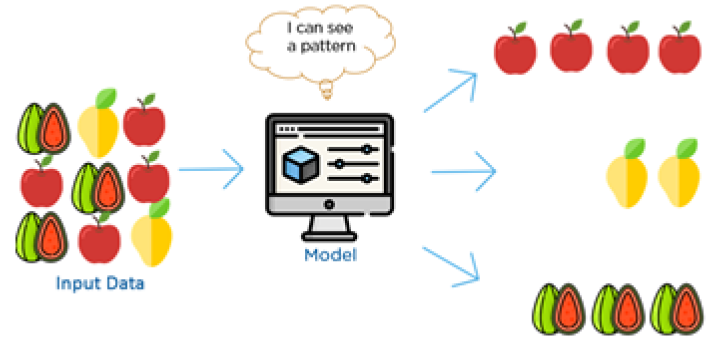
BCILAB, UNIST

Supervised vs. unsupervised learning

Supervised learning

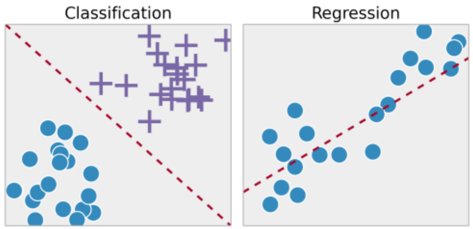


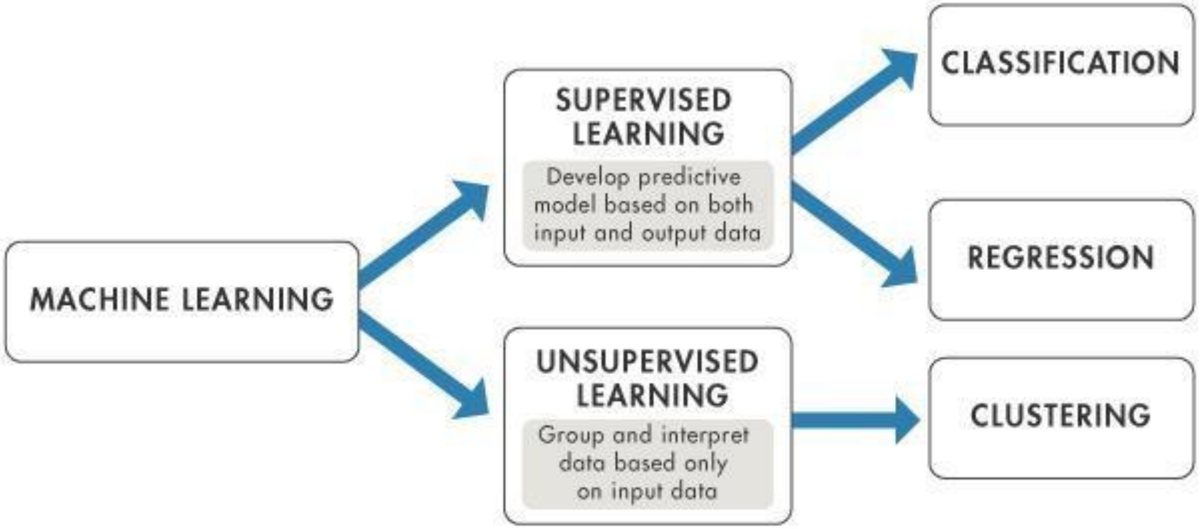
Unsupervised learning



https://www.quora.com/What-is-the-difference-between-supervised-and-unsupervised-learning-algorithms

Supervised vs. unsupervised learning





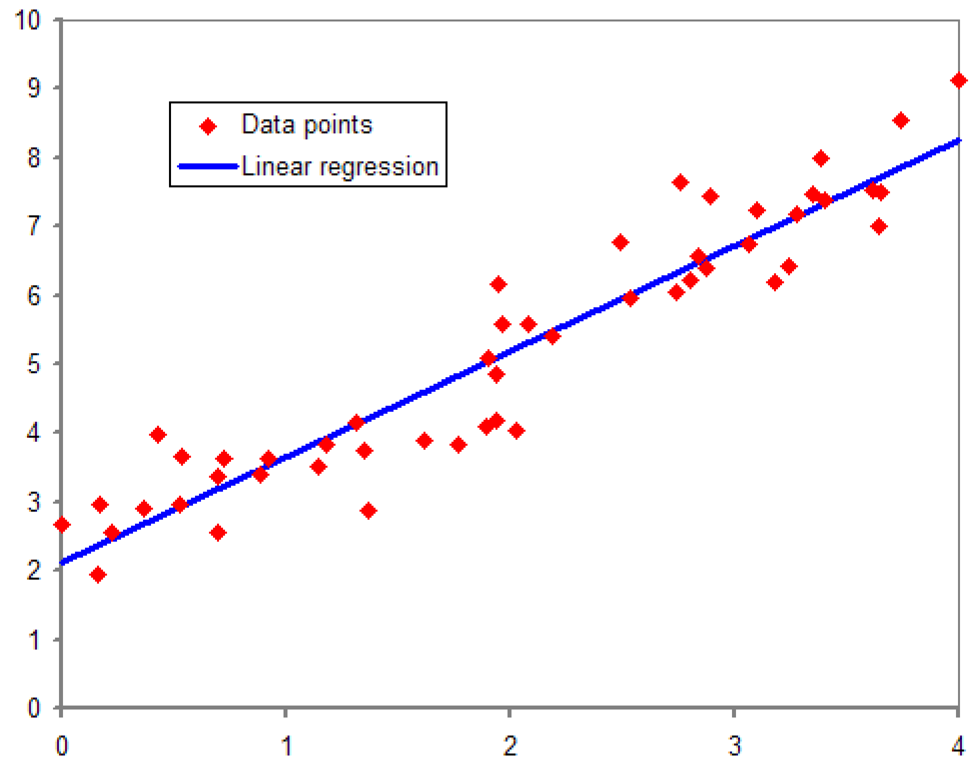
https://www.researchgate.net/figure/Supervised-Learning-versus-Unsupervised-Learning-Mathworks-nd_fig1_319937079
https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d

1

Supervised learning

- Linear Regression
- Naïve Bayes (NB) classifier
- Linear Discriminant Analysis (LDA)
- Support Vector Machine (SVM)

Linear Regression (overview)



$$\hat{y} = ax + b$$

Linear Regression

- Same expression !

$$\boldsymbol{y} = \boldsymbol{a}\boldsymbol{x} + \boldsymbol{b}$$

$$\hat{y} = X\beta + \varepsilon$$

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_n^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{x_{11}} & \cdots & \boldsymbol{x_{1p}} \\ \boldsymbol{x_{21}} & & \boldsymbol{x_{2p}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{x_{n1}} & \cdots & \boldsymbol{x_{np}} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

- **Cost (Loss) function**

$$\begin{aligned} J(\beta) = MSE &= \frac{1}{2m}\sum_{i=1}^{m}\left(\hat{y}^{(i)} - y^{(i)}\right)^2 \\ &= \frac{1}{2m}\sum_{i=1}^{m}\left(X^{(i)}\beta + \varepsilon - y^{(i)}\right)^2 \end{aligned}$$

Linear Regression

- Cost (Loss) function

$$J(\beta) = MSE = \frac{1}{2m}\sum_{i=1}^{m}\left(X^{(i)}\beta + \varepsilon - y^{(i)}\right)^2$$

- Derivate

$$\frac{\partial J}{\partial \varepsilon} = \frac{1}{m}\sum_{i=1}^{m}\left(X^{(i)}\beta + \varepsilon - y^{(i)}\right)$$

$$\frac{\partial J}{\partial \beta} = \frac{1}{m}\sum_{i=1}^{m}\left(X^{(i)}\beta + \varepsilon - y^{(i)}\right)X^{(i)}$$

Linear Regression

- Solution 1: Normal equation, **derivate=0**

$$\varepsilon m + \beta \sum_{i=1}^{m} X^{(i)} = \sum_{i=1}^{m} y^{(i)}$$

$$\varepsilon \sum_{i=1}^{m} X^{(i)} + \beta \sum_{i=1}^{m} {X^{(i)}}^2 = \sum_{i=1}^{m} y^{(i)} X^{(i)}$$

- **Matrix expression**

$$\begin{bmatrix} \varepsilon \\ \beta \end{bmatrix}^T \begin{bmatrix} m & \sum_{i=1}^{m} X^{(i)} \\ \sum_{i=1}^{m} X^{(i)} & \sum_{i=1}^{m} {X^{(i)}}^2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{m} y^{(i)} \\ \sum_{i=1}^{m} y^{(i)} X^{(i)} \end{bmatrix}$$

Linear Regression

- Trick

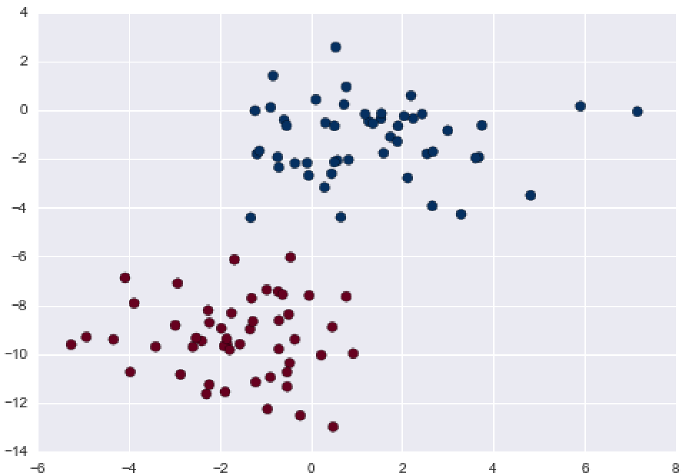
$$\boldsymbol{w} = \begin{bmatrix} \varepsilon \\ \beta \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} 1 & \boldsymbol{x}_1^T \\ 1 & \boldsymbol{x}_2^T \\ & \vdots \\ 1 & \boldsymbol{x}_n^T \end{bmatrix}, \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

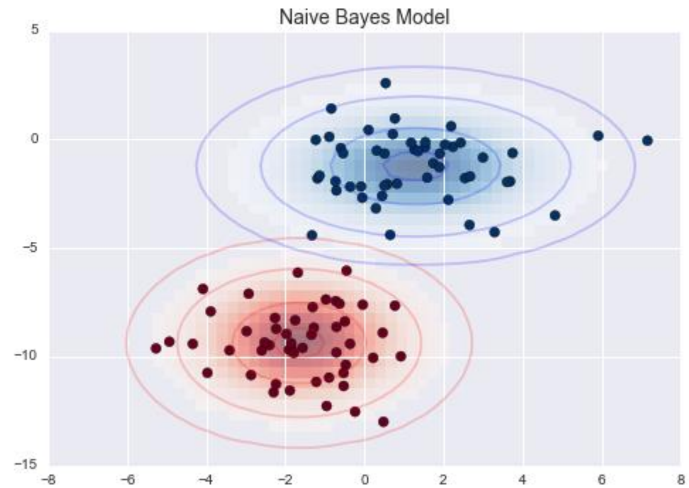
$$\boldsymbol{X}^T\boldsymbol{X} = \begin{bmatrix} m & \sum_{i=1}^m x_i \\ \sum_{i=1}^m x_i & \sum_{i=1}^m x_i^2 \end{bmatrix}, \boldsymbol{X}^T\boldsymbol{y} = \begin{bmatrix} \sum_{i=1}^m y_i \\ \sum_{i=1}^m y_i x_i \end{bmatrix}$$

- Estimated weight

$$wX^TX = X^Ty$$
$$\boldsymbol{w} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-\mathbf{1}}\boldsymbol{X}^T\boldsymbol{y}$$

Naïve Bayes classifier (overview)





https://medium.com/@gp_pulipaka/applying-gaussian-na%C3%AFve-bayes-classifier-in-python-part-one-9f82aa8d9ec4

Naïve Bayes classifier

- **Bayes theorem-based model**
  - Bayes' theorem

Likelihood *Class prior probability*

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

Posterior probability *Predictor prior probability*

Naïve Bayes classifier

$$p(C_k|x) = \frac{p(x|C_k)p(C_k)}{p(x)}$$

- **Joint probability**: chain rule of conditional probability

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\ &= p(x_1|x_2, \dots, x_n, C_k)p(x_2, \dots, x_n, C_k) \\ &= p(x_1|x_2, \dots, x_n, C_k)p(x_2|x_3, \dots, x_n, C_k)p(x_3, \dots x_n, C_k) \\ &= \cdots \\ &= p(x_1|x_2, \dots, x_n, C_k)p(x_2|x_3, \dots, x_n, C_k) \dots p(x_n|C_k)p(C_k) \end{aligned}$$

- **Assumption of "naïve conditional independence"** :
  - Each x_i is conditionally independent of every other features x_j for $j \neq i$ given the category C_k

$$p(x_i|x_{i+1}, \dots, x_n, C_k) = p(x_i|C_k)$$

- **Posterior**

$$\begin{aligned} p(C_k|x_1, \dots x_n) \propto p(C_k, x_1, \dots, x_n) &= p(C_k)p(x_1|C_k)p(x_2|C_k) \dots p(x_n|C_k) \\ &= p(C_k)\prod_{i=1}^{n} p(x_i|C_k) \end{aligned}$$

Naïve Bayes classifier

- Maximum a posterior (MAP)

$$\hat{y} = \underset{k\in\{1,\dots,K\}}{\operatorname{argmax}}\, p(C_k) \prod_{i=1}^{n} p(x_i|C_k)$$

- What is $\boldsymbol{p(x_i|C_k)}$?
  - parameter estimation based on assumption of likelihood
    - Gaussian naïve Bayes → μ, σ

$$p(x = v|C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(v-\mu_k)^2}{2\sigma_k^2}}$$

    - Multinomial naïve Bayes → p_{ki}

$$p(X|C_k) = \frac{(\sum_i x_i)!}{\prod_i x_i!} \prod_i p_{ki}^{x^i}$$

Linear Discriminant Analysis (LDA)

Shrinkage Linear discriminant analysis

Support Vector Machine (SVM)

2

Unsupervised learning

- Random forest
- K-nearest neighborhood
- K-means clustering

Random forest

K-nearest neighborhood

K-means clustering