AIMS | African Institute for Mathematical Sciences
NEXT EINSTEIN INITIATIVE

GAUSSIAN DISCRIMINANT ANALYSIS (GDA)

Nancy, Ornela, Avotra, Allassan, Fenosoa

April 22, 2022

Overview

1. Introduction

2. Gaussian Discriminant Analysis

3. Sentimental Analysis using GDA

4. Conclusion

Introduction

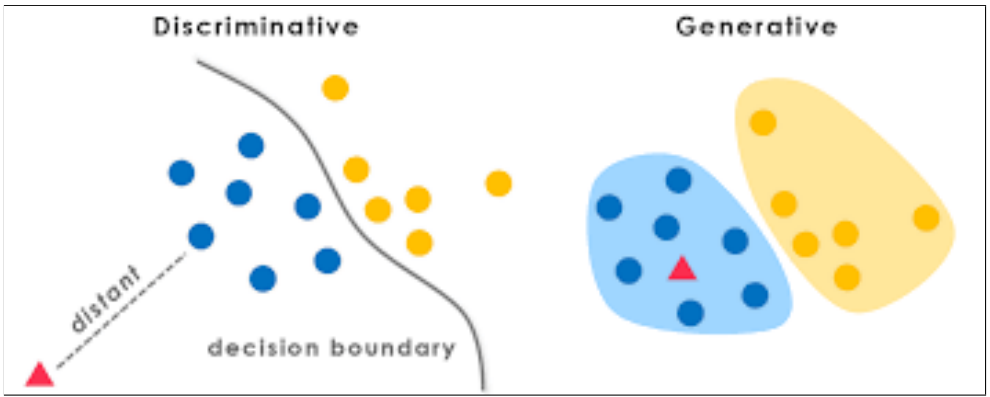
Introduction

The main aim of this project is to implement Gaussian Discriminative Analysis. Machine Learning models can be classify as discriminative or generative models.

- Discriminative Models
  We model probability distribution $\mathbb{P}(\boldsymbol{y}|\boldsymbol{x})$ with the training data.

- Generative Models
  We This is a part of unsupervised learning task in

Machine Learning Models



Driscriminative models Vs Generative models

Gaussian Discriminant Analysis

Gaussian Discriminant Analysis

GDA is a generative learning algorithm.

- Input features are continuous random variables
- Assume $\mathbb{P}(\boldsymbol{x}|\boldsymbol{y})$ and $\mathbb{P}(\boldsymbol{y})$ are distributed according to multivariate normal distribution and Bernuolli distribution respectively.

Gaussian Discriminant Analysis

Assumptions

$$\mathbb{P}[\boldsymbol{y}_i; \phi] = \phi^{\boldsymbol{y}_i}(1-\phi)^{1-\boldsymbol{y}_i},\ \boldsymbol{y}_i \in \{0, 1\}. \tag{1}$$

$$\mathbb{P}\big[\boldsymbol{x}_i | \boldsymbol{y}_i = 0; \boldsymbol{\mu_0}, \boldsymbol{\Sigma}\big] = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{\mu_0})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu_0}) \right) \tag{2}$$

$$\mathbb{P}\big[\boldsymbol{x}_i | \boldsymbol{y}_i = 1; \boldsymbol{\mu_1}, \boldsymbol{\Sigma}\big] = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{\mu_1})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu_1}) \right) \tag{3}$$

Gaussian Discriminant Analysis

From Bayes rule , we can compute $\mathbb{P}(\boldsymbol{y}|\boldsymbol{x})$.

$$\mathbb{P}(\boldsymbol{y}|\boldsymbol{x}) = \frac{\mathbb{P}(\boldsymbol{x}|\boldsymbol{y}) \times \mathbb{P}(\boldsymbol{y})}{\mathbb{P}(\boldsymbol{x})}$$

$$\text{argmax}_{\boldsymbol{y}}\mathbb{P}(\boldsymbol{y}|\boldsymbol{x}) = \text{argmax}_{\boldsymbol{y}}\frac{\mathbb{P}(\boldsymbol{x}|\boldsymbol{y}) \times \mathbb{P}(\boldsymbol{y})}{\mathbb{P}(\boldsymbol{x})}$$

$$\text{argmax}_{\boldsymbol{y}}\mathbb{P}(\boldsymbol{y}|\boldsymbol{x}) \propto \text{argmax}_{\boldsymbol{y}}\mathbb{P}(\boldsymbol{x}|\boldsymbol{y}) \times \mathbb{P}(\boldsymbol{y})$$

The likelihood function L is given by

$$L(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma}, \phi) = \mathbb{P}(\boldsymbol{x}|\boldsymbol{y}; \boldsymbol{\mu_k}, \boldsymbol{\Sigma})\mathbb{P}(\boldsymbol{y}; \phi) \text{ where } k = \mathbb{1}_{\{\boldsymbol{y}_i=1\}}$$

$$\phi = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{y}_i = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{\boldsymbol{y}_i=1\}}$$

$$\boldsymbol{\mu_0} = \frac{\sum_{i=1}^{n} \mathbb{1}_{\{\boldsymbol{y}_i=0\}}\boldsymbol{x}_i}{\mathbb{1}_{\{\boldsymbol{y}_i=0\}}}$$

$$\boldsymbol{\mu_1} = \frac{\sum_{i=1}^{n} \mathbb{1}_{\{\boldsymbol{y}_i=1\}}\boldsymbol{x}_i}{\mathbb{1}_{\{\boldsymbol{y}_i=1\}}}$$

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n} \left(\boldsymbol{x}_i - \boldsymbol{\mu_k}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu_k}\right)^T \text{ where } k = \mathbb{1}_{\{\boldsymbol{y}_i=1\}}$$

Sentimental Analysis using GDA

Sentimental Analysis using GDA

Preprocessing

Steps

- lower case
- remove stopwords
- remove punctuations
- remove apostrophe
- tokenize
- lemmatization
- convert numbers

TF -IDF

- Term- Frequency

$$TF = \frac{\text{Nbr of repetitions of word in a doc}}{\text{\# of words in a doc}}$$

- Inverse Document Frequency

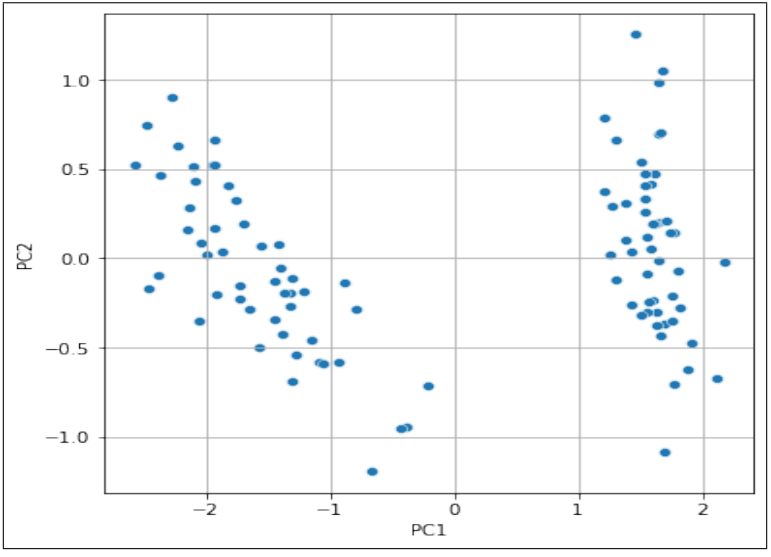
$$IDF = \log\left[\frac{\text{\# Number of docs}}{\text{Nbr of docs containing the word}}\right]$$

Sentimental Analysis using GDA

The estimated parameters from the training dataset

$$
\begin{aligned}
\phi^* &= 0.75 \\
\boldsymbol{\mu_0}^* &= (-0.0262559, -0.01121153)^T \\
\boldsymbol{\mu_1}^* &= (0.00595905, 0.0021744)^T \\
\boldsymbol{\Sigma}^* &= \begin{pmatrix} 4.62948277 \cdot 10^{-4} & -9.65664189 \cdot 10^{-6} \\ -9.65664189 \cdot 10^{-6} & 1.49865708 \cdot 10^{-3} \end{pmatrix}
\end{aligned}
$$

The accuracy after prediction is 50%.



Visualization of the Iris Dataset using PCA

Conclusion

In conclusion,

- GDA model does not need any hyperparameter.
- GDA models works perfectly with small dataset.
- Time complexity is expensive


THANK
YOU