MACHINE LEARNING

LINEAR CLASSIFICATION

AGENDA

01 Introduction

Classification problem and linear regression

02 Binary classification

Logistic regression and Newton´s method

03 Generalized Linear Models

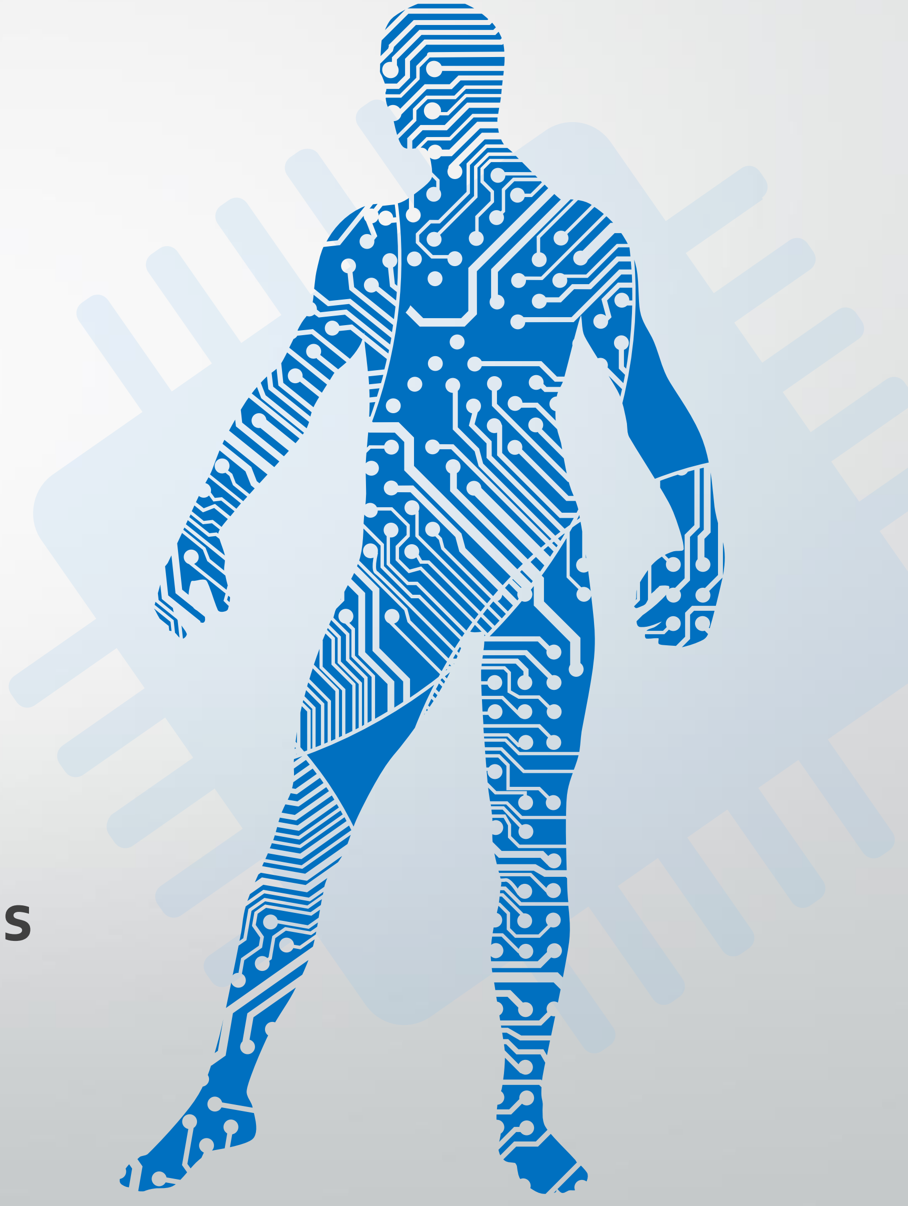
Exponential Family, Building GLMs, Softmax

04 Discriminative and generative models

Differences, Gaussian Analysis, Bayes Classifier

05 Binary evaluation metrics

Sensitivity, Specificity, F1 Score, ROC Curve



AI

INTRODUCTION

INTRODUCTION

CLASSIFICATION PROBLEM

The problem is the same as that of **regression**: predicting a set of **output variables** y given a set of **input data** X.

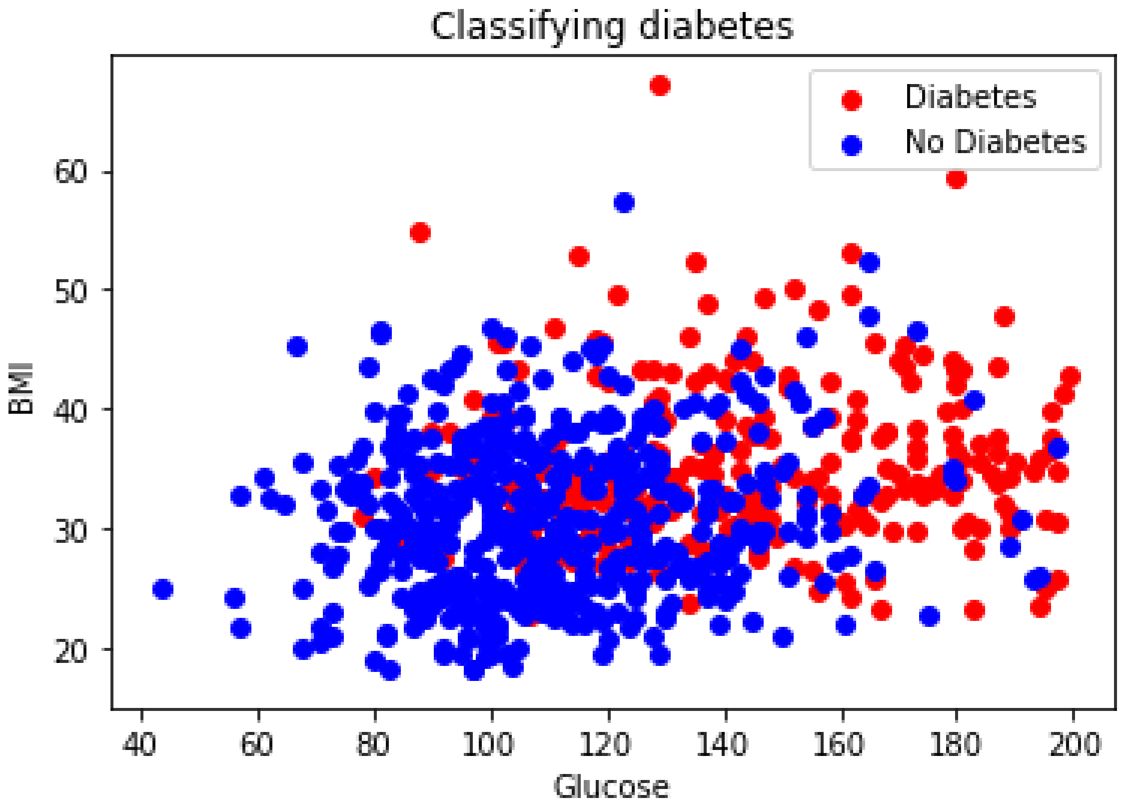
The only **difference** is that the **values** of y take on a set of **discrete values**.

x_1=Glucose [mg/dl]	x_2=IMC	y = Presence of diabetes
148	**33.6**	**1**
85	**26.6**	**0**
183	**23.3**	**1**
89	**28.1**	**0**
137	**43.1**	**1**
⋮	⋮	⋮

INTRODUCTION

CLASSIFICATION PROBLEM

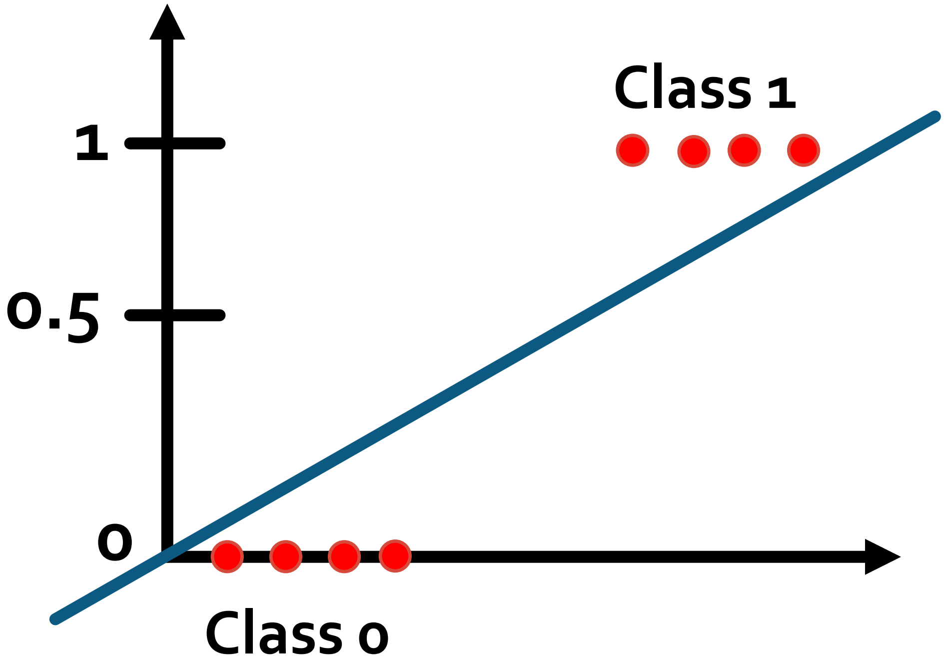
We plot 752 data points (patients) using Python:



INTRODUCTION

LINEAR REGRESSION

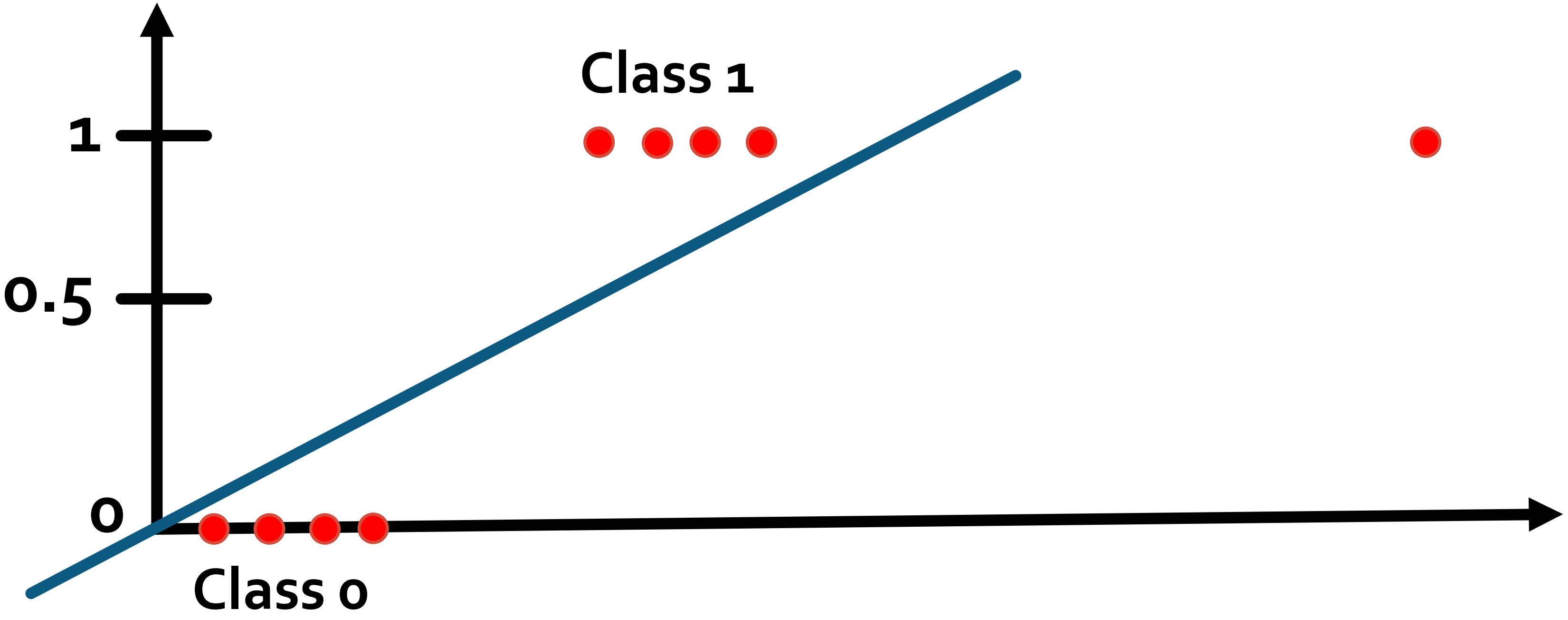
WHY NOT APPLY LINEAR REGRESSION TO CLASSIFY?



INTRODUCTION

LINEAR REGRESSION

WHY NOT APPLY LINEAR REGRESSION TO CLASSIFY?



The function $w^T X$ does not describe the expected value of $Y \in \{0, 1\}$

BINARY CLASSIFICATION

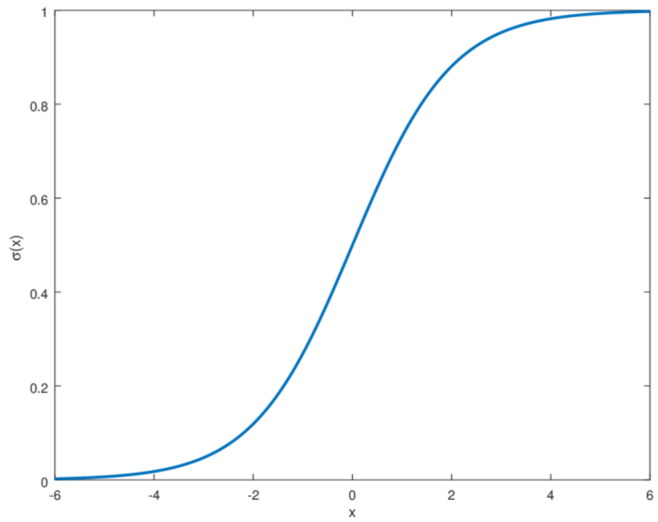
LOGISTIC REGRESSION

HYPOTHESIS

The **hypothesis** that we use for **linear regression** is not adequate to solve our **classification problem**.

A **hypothesis** that **better fits** the problem is proposed

$$h_w(x) = g\left(w^T X\right) = \frac{1}{1 + e^{-w^T X}}$$



This function is called the **sigmoid** or **logistic function.**

It is **guaranteed** that $g(z) \in \{0, 1\}$**.**

LOGISTIC REGRESSION

SIGMOID FUNCTION

HOMEWORK

Prove that:

$$g'(z) = g(z)(1 - g(z))$$

LOGISTIC REGRESSION
ASSUMPTIONS

Now, we define the **same problem** of finding the best combination of **weights** that **fits** the **classification problem**.

The following **assumptions** are applied:

1. We assume that the **hypothesis** defines a **probability measure** (**Bernoulli**):

$$P(y = 1/x; w) = h_w(x)$$

$$P(y = 0/x; w) = 1 - h_w(x)$$

$$P(y/x; w) = h_w(x)^y (1 - h_w(x))^{1-y}$$

$$L(w) = P(y/x; w)$$

LOGISTIC REGRESSION

ASSUMPTIONS

2. Assuming $\boldsymbol{m}$ **samples** are taken, we have that the **likelihood** can be written as:

$$\boldsymbol{L(w) = \prod_{i=1}^{m} P\left(y^{(i)}/x^{(i)}; w\right)}$$

$$\boldsymbol{L(w) = \prod_{i=1}^{m} h_w\left(x^{(i)}\right)^{y^{(i)}} \left(1 - h_w\left(x^{(i)}\right)\right)^{1-y^{(i)}}}$$

LOGISTIC REGRESSION

LOGARITHMIC LOSS

The **logarithmic loss** can be written as:

$$l(w) = log \prod_{i=1}^{m} P\left(y^{(i)} / x^{(i)}; w\right)$$

$$l(w) = \sum_{i=1}^{m} y^{(i)} log\, h_w\left(x^{(i)}\right) + \left(1 - y^{(i)}\right) log \left(1 - h_w\left(x^{(i)}\right)\right)$$

$$l(w) = \sum_{i=1}^{m} y^{(i)} log \left(\frac{1}{1 + e^{-w^T X}}\right) + \left(1 - y^{(i)}\right) log \left(1 - \frac{1}{1 + e^{-w^T X}}\right)$$

LOGISTIC REGRESSION

COST FUNCTION

The **cost function** can be expressed as:

$$\boldsymbol{J(w) = -l(w)}$$

$$\boldsymbol{J(w) = -\sum_{i=1}^{m} y^{(i)} log\left(\frac{1}{1+e^{-w^T X}}\right) + \left(1 - y^{(i)}\right) log\left(1 - \frac{1}{1+e^{-w^T X}}\right)}$$

LOGISTIC REGRESSION

COST FUNCTION

Interpreting the **cost function**:

$$J(w) = -\sum_{i=1}^{m} y^{(i)} log\, h_w\left(x^{(i)}\right) + \left(1 - y^{(i)}\right) log\left(1 - h_w\left(x^{(i)}\right)\right)$$

If $y = 0$	If $y = 1$
$J(y = 0, \hat{y}) = -log(1 - \hat{y})$	$J(y = 1, \hat{y}) = -log(\hat{y})$
$\lim_{\hat{y} \to 0} log(1 - \hat{y}) \to 0$	$\lim_{\hat{y} \to 0} log(\hat{y}) \to \infty$
$\lim_{\hat{y} \to 1} log(1 - \hat{y}) \to \infty$	$\lim_{\hat{y} \to 1} log(\hat{y}) \to 0$

LOGISTIC REGRESSION

OPTIMIZATION

HOW TO FIND THE BEST WEIGHTS W?

LOGISTIC REGRESSION

GRADIENT DESCENT

Finding the best combination of **weights** $\boldsymbol{w}$ by **gradient descent** fits the **classification problem**.

$$\boldsymbol{w} := \boldsymbol{w} - \boldsymbol{\alpha}\,\boldsymbol{\nabla_w J(w)}$$

HOMEWORK

Show that the **gradient** with respect to a **single training data** $(\boldsymbol{x}, \boldsymbol{y})$ and **weight** $\boldsymbol{w_j}$ is given by:

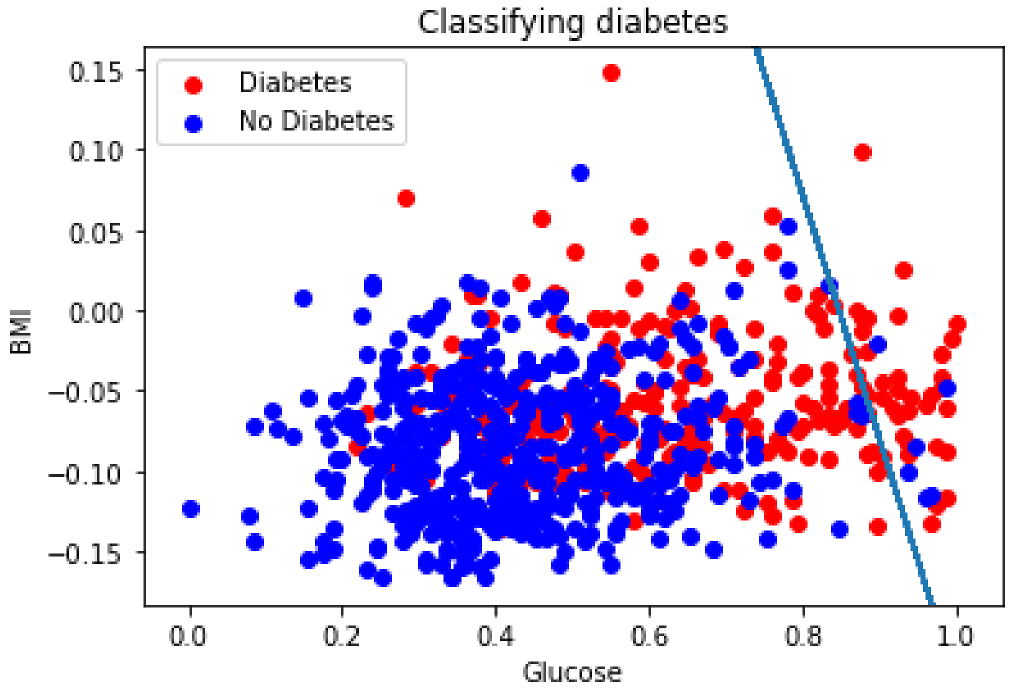
$$\frac{\partial}{\partial w_j} J(w) = (y - h_w(x))x_j$$

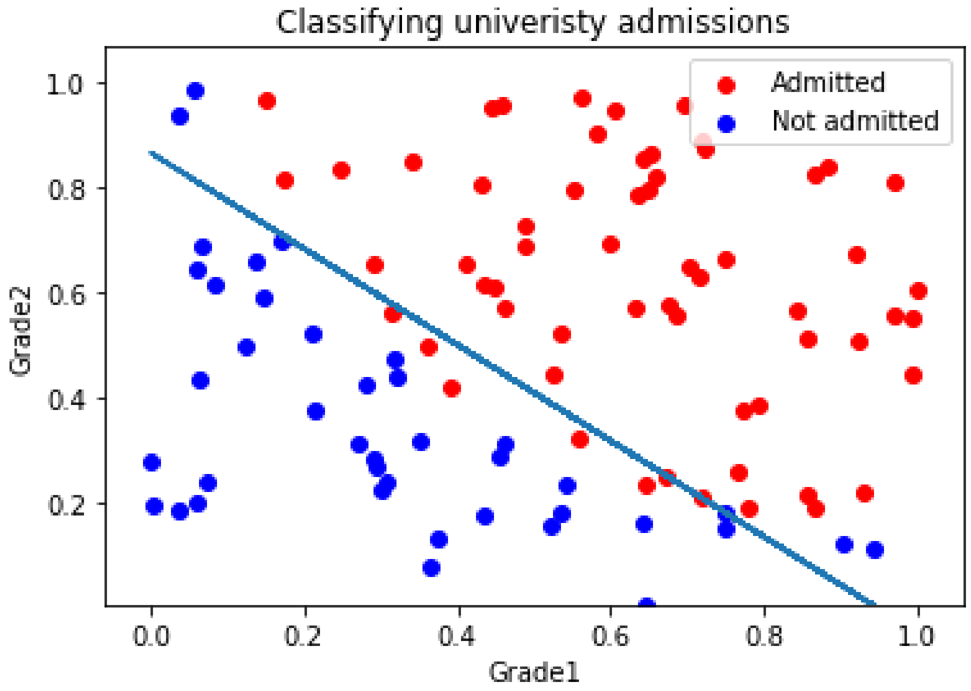
Where $\boldsymbol{x}, \boldsymbol{w_j} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \{\mathbf{0}, \mathbf{1}\}$

REMEMBER THE GRADIENT DESCENT IN LINEAR REGRESSION→ SAME RESULT

LOGISTIC REGRESSION

REAL EXAMPLE



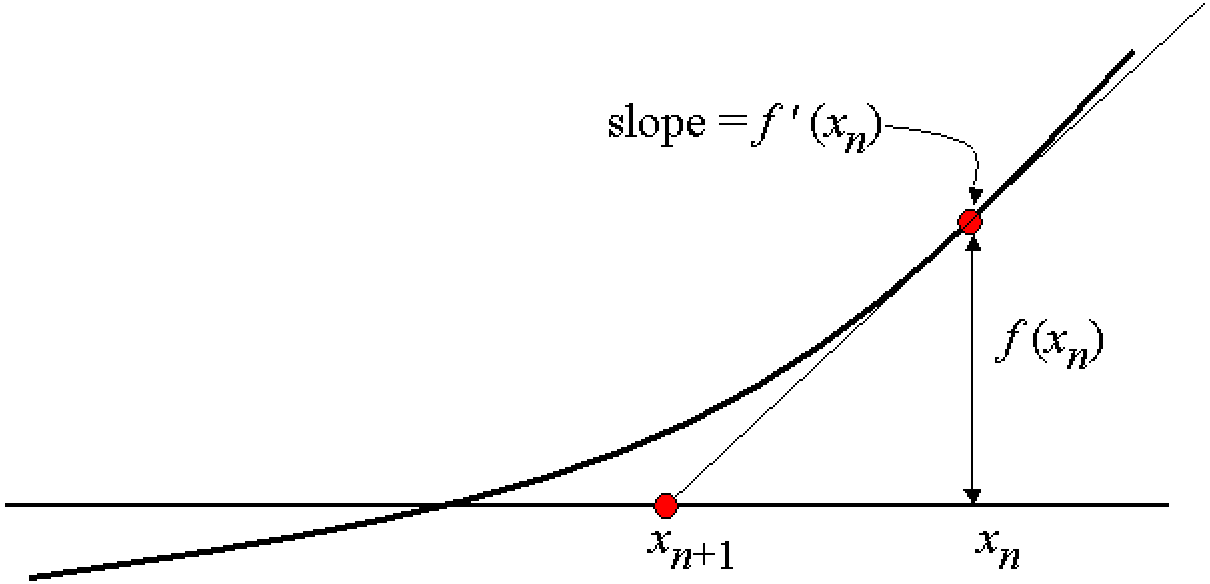


NEWTON'S METHOD

OPTIMIZATION

We **recall** from numerical methods, **Newton's method**, where the value of $\boldsymbol{w}$ is found for which $\boldsymbol{f}\,(\boldsymbol{w}) = \mathbf{0}$:

$$\boldsymbol{w} \coloneqq \boldsymbol{w} - \frac{\boldsymbol{f}(\boldsymbol{w})}{\boldsymbol{f}'(\boldsymbol{w})}$$



NEWTON'S METHOD

OPTIMIZATION

If we want to find the **minimum** of the **cost function** $\boldsymbol{J}\ (\boldsymbol{w})$, this will **correspond** to finding the **points** where $\boldsymbol{\nabla_w J(w) = 0}$, meaning that we can use **Newton's method** (much faster):

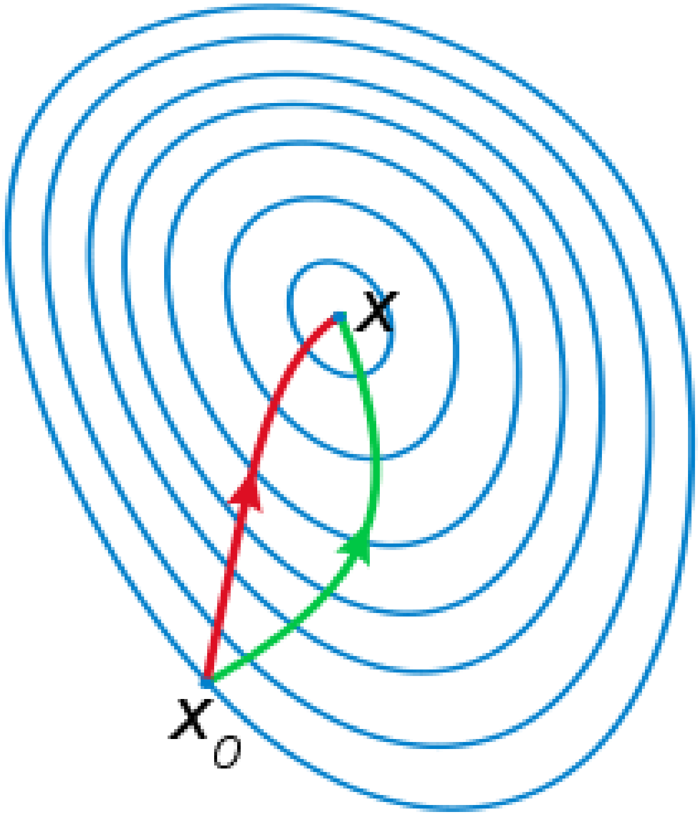
$$\boldsymbol{w \coloneqq w - H^{-1} \nabla_w J(w)}$$

where $\boldsymbol{H^{-1}}$ is the Hessian matrix, and $\boldsymbol{\nabla_w}$ is the gradient.

$$\boldsymbol{H} = \begin{bmatrix} \dfrac{\partial^2 J(w)}{\partial {w_1}^2} & \cdots & \dfrac{\partial^2 J(w)}{\partial w_1 \partial w_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial^2 J(w)}{\partial w_n \partial w_1} & \cdots & \dfrac{\partial^2 J(w)}{\partial {w_n}^2} \end{bmatrix}$$

NEWTON´S METHOD

OPTIMIZATION



Gradient descent
(Linear convergence)

Newton´s method
(Quadratic convergence)

NEWTON'S METHOD

OPTIMIZATION

WHY NOT USE NEWTON'S METHOD INSTEAD OF GRADIENT DESCENT?

GENERALIZED LINEAR MODELS

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY

We **recall** that we made the **following assumptions**:

LEAST SQUARES

$\boldsymbol{y} \in \mathbb{R} \sim \boldsymbol{Gaussian}$ (Linear regression)

LOGISTIC REGRESSION

$\boldsymbol{y} \in \{\mathbf{0}, \mathbf{1}\} \sim \boldsymbol{Bernoulli}$ (Binary classification)

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY

We say that a **probability distribution** belongs to the **exponential family** if it can be written as follows:

$$p(y;\eta) = b(y)e^{\left(\eta^T T(y) - a(\eta)\right)}$$

Where:

- η is named as the **natural** or **canonical parameter** of the distribution.
- $T(y)$ **sufficient statistic** (statistic that summarizes the complete information of a sample).
- $a(\eta)$ the **partition logistics function**.

NOTE: $e^{-a(\eta)}$ works as a **normalization constant** to ensure that $p\,(y;\,\eta)$ **integrates 1**.

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY

$$\boldsymbol{p(y;\eta) = b(y)e^{\left(\eta^T T(y) - a(\eta)\right)}}$$

If we **fix** $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{T}$, then we can say that we have a **distribution parametrized** only by $\boldsymbol{\eta}$, whereby varying $\boldsymbol{\eta}$ gives us **different distributions**.

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY EXAMPLES

We prove that the **Bernoulli distribution** is part of the **exponential family**:

$$\boldsymbol{p(y;\phi) = \phi^{y}(1-\phi)^{1-y}}$$

By **varying** $\boldsymbol{\phi}$ we obtain **different distributions** $\boldsymbol{p(y;\phi)}$.

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY EXAMPLES

We find:

$$T(y) = y$$

$$b(y) = 1$$

$$\eta = log\left(\frac{\phi}{1-\phi}\right)$$

$$a(\eta) = -log(1-\phi)$$

$$p(y;\phi) = e^{\left(log\left(\frac{\phi}{1-\phi}\right)y + log(1-\phi)\right)}$$

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILIY EXAMPLES

We prove that the **Gaussian distribution** (where $\boldsymbol{\sigma}$ **does not matter**) is part of the **exponential family** (so $\boldsymbol{\sigma^2 = 1}$**):**

$$\boldsymbol{p(y;\mu) = \frac{1}{\sqrt{2\pi}} e^{\frac{(y-\mu)^2}{2}}}$$

HOMEWORK 1

GENERALIZED LINEAR MODELS

CONSTRUCTING THE MODELS

We are going to **make** the **following assumptions**:

1. The output variable $\boldsymbol{y} \,/\, \boldsymbol{X}; \boldsymbol{w} \sim \boldsymbol{FamExp}\ (\boldsymbol{\eta})$

2. **Objective:** given a matrix of characteristics $\boldsymbol{X}$**,** calculate $\boldsymbol{E}(\boldsymbol{T}(\boldsymbol{y}); \boldsymbol{\eta})$ referred to as a **canonical response function**.

$$\boldsymbol{h}(\boldsymbol{\eta}) = \boldsymbol{E}(\boldsymbol{T}(\boldsymbol{y}); \boldsymbol{\eta})$$

3. The **relationship** between $\boldsymbol{\eta}$, $\boldsymbol{w}$ and $\boldsymbol{X}$ is defined to be **linear** (only if $\boldsymbol{\eta} \in \mathbb{R}$):

$$\boldsymbol{\eta} = \boldsymbol{w}^{\boldsymbol{T}} \boldsymbol{X}$$

GENERALIZED LINEAR MODELS

CONSTRUCTING THE MODELS

EXAMPLE: Bernoulli

For fixed values $\boldsymbol{X}$ and $\boldsymbol{w}$ the **objective** is to **calculate**:

$$\boldsymbol{h(w) = E(T(y);\eta)}$$

But we know that $\boldsymbol{T(y) = y}$ so the **expected value** of a variable that is **distributed** as **Bernoulli** is:

$$\boldsymbol{E(y;\ \eta) = P(y = 1;\ \eta)}$$

Similarly, we know from the exponential family that:

$$\boldsymbol{\phi = P(y = 1;\ \eta)}$$

GENERALIZED LINEAR MODELS

EXPONENTIAL FAMILY EXAMPLES

It follows from the **definition** of the **exponential family** that the **natural parameter** $\boldsymbol{\eta}$ is expressed as:

$$\boldsymbol{\eta} = \boldsymbol{log}\left(\frac{\boldsymbol{\phi}}{\mathbf{1}-\boldsymbol{\phi}}\right)$$

By **clearing** $\boldsymbol{\phi}$ from the previous equality:

$$\boldsymbol{\eta} = \boldsymbol{log}\left(\frac{\boldsymbol{\phi}}{\mathbf{1}-\boldsymbol{\phi}}\right)$$

The **following** is **obtained**:

$$\boldsymbol{\phi} = \frac{\mathbf{1}}{\mathbf{1}+\boldsymbol{e}^{-\boldsymbol{\eta}}}$$

GENERALIZED LINEAR MODELS

CONSTRUCTING THE MODELS

So the **expected value** is:

$$\boldsymbol{E(y;\ \eta) = \phi = \frac{1}{1 + e^{-\eta}}}$$

But thanks to the third assumption $\boldsymbol{\eta = w^T X}$

$$\boldsymbol{E(y; w, X) = \phi = \boxed{\frac{1}{1 + e^{-W^T X}}}}$$

SIGMOID FUNCTION

CONSTRUCTING THE MODELS

EXAMPLE: Gaussian

Prove that the canonical response function is equivalent to $\boldsymbol{W^T X}$ assuming that $\boldsymbol{y \sim Gaussian}$ and that $\boldsymbol{\eta} = \boldsymbol{W^T X}$

$$\boldsymbol{h(w) = E(T(y);\, \eta) = w^T X}$$

HOMEWORK 2

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

The following problem is defined, where the **output variable** $\boldsymbol{y} \in \{\mathbf{1}, \ldots, \boldsymbol{k}\}$ is distributed as a **multinomial function** $\boldsymbol{y} \sim \boldsymbol{Multinomial}$.

- Parameters: $\phi_1, \phi_2, \ldots, \phi_{k-1}$
- $P(y = i) = \phi_i$
- $\phi_k = 1 - (\phi_1 + \phi_2 + \cdots + \phi_{k-1})$

The **sufficient statistic** $\boldsymbol{T}\ (\boldsymbol{y})$ is defined as a **vector**:

$$T(1) = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} T(2) = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} \ldots\ T(k-1) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} T(k) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{k-1}$$

Indicator function:

$$1\{True\} = 1 \qquad 1\{False\} = 0$$

$$T(y)_i = 1\{y = i\}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

The **multinomial distribution** is **expressed** in the form of the **exponential family**:

$$P(y) = \phi_1^{1\{y=1\}} \phi_2^{1\{y=2\}} \dots \phi_k^{1\{y=k\}}$$

$$P(y) = \phi_1^{T(y)_1} \phi_2^{T(y)_2} \dots \phi_k^{T(y)_{k-1}} \phi_k^{1-\sum_{j=1}^{k-1} T(y)_j}$$

$$P(y) = e^{\left[T(y)_1 \log(\phi_1) + T(y)_2 \log(\phi_2) + \dots + \left(1-\sum_{j=1}^{k-1} T(y)_j\right) \log(\phi_k)\right]}$$

$$P(y) = e^{\left[T(y)_1 \log\left(\frac{\phi_1}{\phi_k}\right) + T(y)_2 \log\left(\frac{\phi_2}{\phi_k}\right) + \dots + T(y)_{k-1} \log\left(\frac{\phi_{k-1}}{\phi_k}\right) + \log(\phi_k)\right]}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

It is **compared** to the **exponential family form**:

$$P(y) = e^{\left[T(y)_1 \log\left(\frac{\phi_1}{\phi_k}\right) + T(y)_2 \log\left(\frac{\phi_2}{\phi_k}\right) + \ldots + T(y)_{k-1} \log\left(\frac{\phi_{k-1}}{\phi_k}\right) + \log(\phi_k)\right]}$$

$$\boldsymbol{p(y;\eta) = b(y)e^{\left(\eta^T T(y) - a(\eta)\right)}}$$

Therefore:

$$\boldsymbol{b(y) = 1}$$

$$\boldsymbol{a(\eta) = -log(\phi_k)}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

$$P(y) = e^{\left[T(y)_1 \log\left(\frac{\phi_1}{\phi_k}\right) + T(y)_2 \log\left(\frac{\phi_2}{\phi_k}\right) + \ldots + T(y)_{k-1} \log\left(\frac{\phi_{k-1}}{\phi_k}\right) + \log(\phi_k)\right]}$$

$$\boldsymbol{p(y;\eta) = b(y)e^{\left(\eta^T T(y) - a(\eta)\right)}}$$

Finally:

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{log\left(\frac{\phi_1}{\phi_k}\right)} \\ \vdots \\ \boldsymbol{log\left(\frac{\phi_{k-1}}{\phi_k}\right)} \end{bmatrix}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

The **canonical response function** would be:

$$\eta_i = log\left(\frac{\phi_i}{\phi_k}\right)$$

$$\eta_k = log\left(\frac{\phi_k}{\phi_k}\right) = 0$$

$$\phi_k e^{\eta_i} = \phi_i$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

The **sum of probabilities** must be **equal** to **1**:

$$\sum_{i=1}^{k} \phi_k e^{\eta_i} = \sum_{i=1}^{k} \phi_i = 1$$

$$\phi_k \sum_{i=1}^{k} e^{\eta_i} = \sum_{i=1}^{k} \phi_i = 1$$

$$\phi_k = \frac{1}{\sum_{i=1}^{k} e^{\eta_i}}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

Substituting $\phi_k = \frac{1}{\sum_{i=1}^{k} e^{\eta_i}}$ in the canonical response function

$$\phi_k e^{\eta_i} = \phi_i$$

$$\boxed{\phi_i = \frac{e^{\eta_i}}{\sum_{j=1}^{k} e^{\eta_j}}}$$

SOFTMAX FUNCTION

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

Using the **third assumption** that $\boldsymbol{\eta_i = w_i^T X}$**:**

$$\boldsymbol{\phi_i = \frac{e^{w_i^T X}}{\sum_{j=1}^{k} e^{w_j^T X}}}$$

Knowing that the objective is to calculate $\boldsymbol{h(w) = E(T(y);\ \eta)}$:

$$\boldsymbol{E\left(\begin{bmatrix} T(y)_1 \\ T(y)_2 \\ \vdots \\ T(y)_{k-1} \end{bmatrix}\right) = E\left(\begin{bmatrix} 1\{y = 1\} \\ 1\{y = 2\} \\ \vdots \\ 1\{y = k-1\} \end{bmatrix}\right) = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{k-1} \end{bmatrix}}$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

The hypothesis $\boldsymbol{h(w)}$ is **expressed** as a **vector**:

$$\boldsymbol{h(w)} = \begin{bmatrix} \dfrac{e^{w_1^T X}}{\sum_{j=1}^{k} e^{w_j^T X}} \\ \dfrac{e^{w_2^T X}}{\sum_{j=1}^{k} e^{w_j^T X}} \\ \vdots \\ \dfrac{e^{w_{k-1}^T X}}{\sum_{j=1}^{k} e^{w_j^T X}} \end{bmatrix} = \begin{bmatrix} P(y=1) \\ P(y=2) \\ \vdots \\ P(y=k-1) \end{bmatrix}$$

which has as **components** the **probability** of **each class** $\boldsymbol{i}$: $\boldsymbol{P(y=i)}$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

We calculate the **logarithmic loss** and **cost function** for a **single training data**:

$$l(w) = \sum_{l=1}^{k} 1(y = l) log(p(y/x)) = \sum_{i=1}^{k} 1(y = l) log\left(\frac{e^{w_l^T X}}{\sum_{j=1}^{k} e^{w_j^T X}}\right)$$

$$J(w) = -l(w)$$

GENERALIZED LINEAR MODELS

SOFTMAX REGRESSION

We calculate the **logarithmic loss** and **cost function** for $\boldsymbol{m}$ **training data**:

$$l(w) = \sum_{i=1}^{m} \sum_{l=1}^{k} 1\left(y^{(i)} = l\right) log \left(\frac{e^{w_l^T x^{(i)}}}{\sum_{j=1}^{k} e^{w_j^T x^{(i)}}} \right)$$

$$J(w) = -l(w)$$

THE NEWTON OR GRADIENT DESCENT METHOD MAY BE USED

DISCRIMINATIVE AND GENERATIVE MODELS

DISCRIMINATIVE AND GENERATIVE MODELS
DIFFERENCES

Discriminative models:

- The models that have been seen so far are called **discriminative models**, where learning about the **probability distribution** $\boldsymbol{p}\ (\boldsymbol{y}\ /\ \boldsymbol{x})$ is done **directly**.

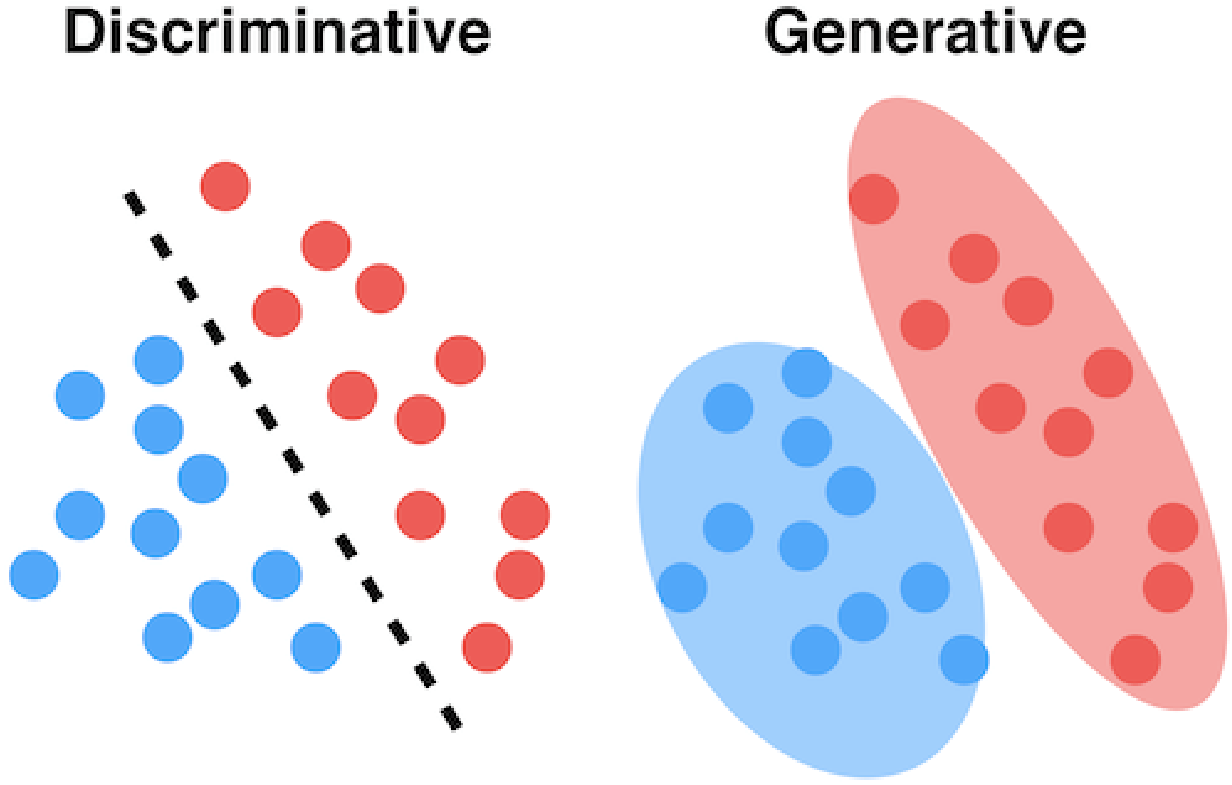
- **Examples**: logistic regression and least squares.

Generative models:

- Models that try to learn $\boldsymbol{p}\ (\boldsymbol{x}\ /\ \boldsymbol{y})$ and $\boldsymbol{p}\ (\boldsymbol{y})$. For the case of **binary classification** the algorithm would learn **three different distributions** $\boldsymbol{p}\ (\boldsymbol{x}\ /\ \boldsymbol{y} = \mathbf{0})$, $\boldsymbol{p}\ (\boldsymbol{x}\ /\ \boldsymbol{y} = \mathbf{1})$ and $\boldsymbol{p}\ (\boldsymbol{y})$.

- Thus, a new **data** would be **classified** by **comparing** it with both **distributions** $\boldsymbol{p}\ (\boldsymbol{x}\ /\ \boldsymbol{y} = \mathbf{0})$ and $\boldsymbol{p}\ (\boldsymbol{x}\ /\ \boldsymbol{y} = \mathbf{1})$.

DISCRIMINATIVE AND GENERATIVE MODELS
DIFFERENCES



GENERATIVE MODELS

DESCRIPTION

A **generative model** models the **characteristics** that are **conditioned** by the **response variable** $\boldsymbol{p}(\boldsymbol{x} / \boldsymbol{y})$.

Using the **Bayes' Theorem**, $\boldsymbol{p}\ (\boldsymbol{y} / \boldsymbol{x})$ can be calculated.

$$\boldsymbol{p}(\boldsymbol{y}/\boldsymbol{x}) = \frac{\boldsymbol{p}(\boldsymbol{x}/\boldsymbol{y})\boldsymbol{p}(\boldsymbol{y})}{\boldsymbol{p}(\boldsymbol{x})}$$

We have that the **denominator** can be **calculated** as follows (**binary classification**):

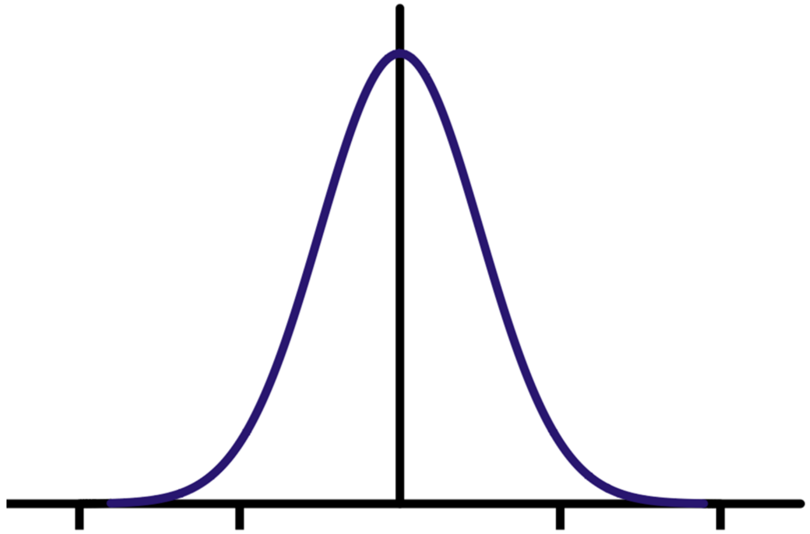
$$\boldsymbol{p}(\boldsymbol{x}) = \boldsymbol{p}(\boldsymbol{x}/\boldsymbol{y} = \mathbf{1})\boldsymbol{p}(\boldsymbol{y} = \mathbf{1}) + \boldsymbol{p}(\boldsymbol{x}/\boldsymbol{y} = \mathbf{0})\boldsymbol{p}(\boldsymbol{y} = \mathbf{0})$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

In a **Gaussian discriminative analysis** we assume that:

$$p(x/y) \sim Gaussian$$



GENERATIVE MODELS

MULTIVARIATE GAUSSIANS

A **Gaussian distribution** of $\boldsymbol{d}$ **dimensions** is **parametrized** by:

- **Mean vector**: $\mu \in \mathbb{R}^d$
- **Covariance matrix**: $\Sigma \in \mathbb{R}^{dxd}$

where Σ is **symmetric** and **positive semi-definite**. Its density can be calculated as:

$$\boldsymbol{p(x;\mu,\Sigma)=\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}e^{\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)}}$$

Properties:

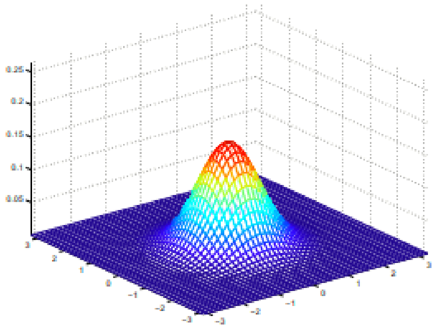
$$\boldsymbol{E[X]=\int_x x\,p(x;\mu,\Sigma)\,dx=\mu}$$

$$\boldsymbol{Cov(X)=E[(X-E[X])(X-E[X])^T]=\Sigma}$$

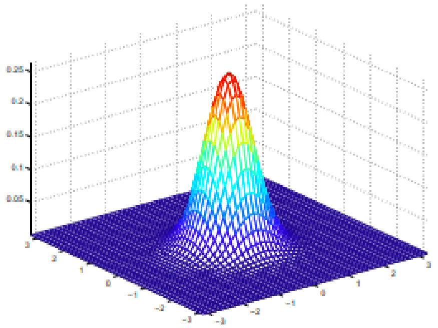
GENERATIVE MODELS

MULTIVARIATE GAUSSIANS

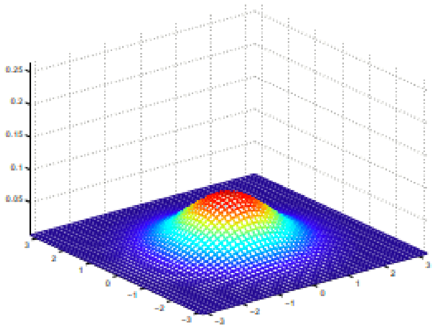
Example: Varying Σ.



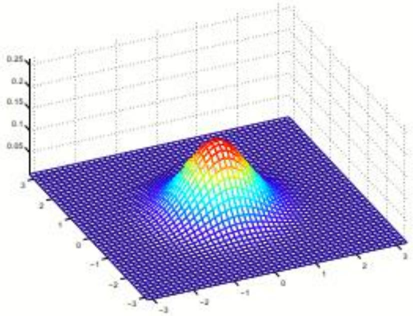
$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



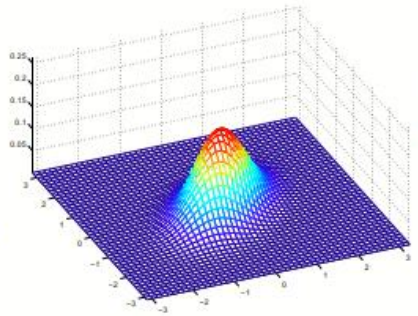
$$\boldsymbol{\Sigma} = \begin{bmatrix} 0.6 & 0 \\ 0 & 0.6 \end{bmatrix}$$



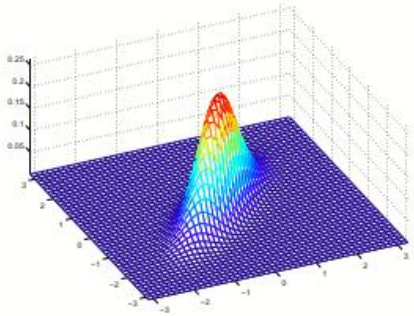
$$\boldsymbol{\Sigma} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$



$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$



$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

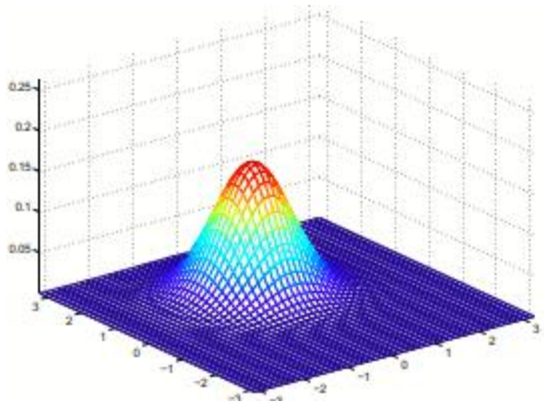


$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$$

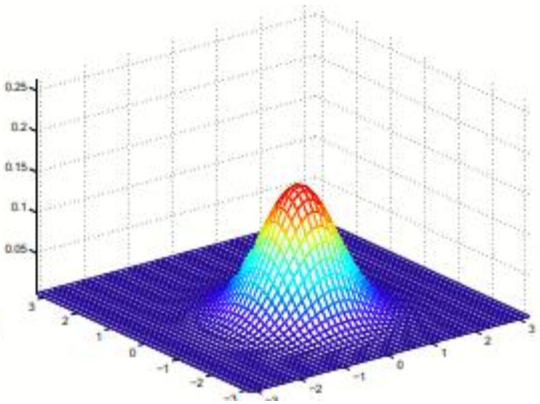
GENERATIVE MODELS

MULTIVARIATE GAUSSIANS

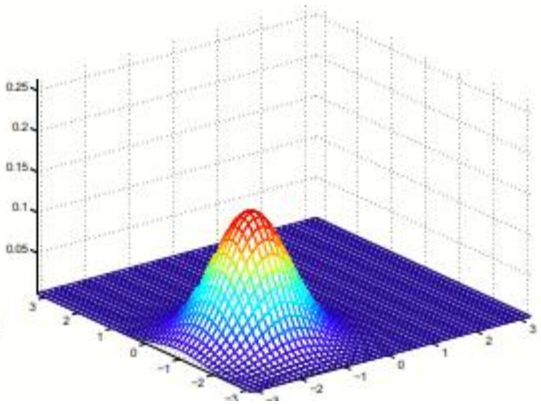
Example: Varying μ.



$\boldsymbol{\mu} = \begin{bmatrix} \mathbf{1} \\ \mathbf{0} \end{bmatrix}$



$\boldsymbol{\mu} = \begin{bmatrix} -\mathbf{0.5} \\ \mathbf{0} \end{bmatrix}$



$\boldsymbol{\mu} = \begin{bmatrix} -\mathbf{1} \\ -\mathbf{1.5} \end{bmatrix}$

G E N E R A T I V E   M O D E L S

GAUSSIAN DISCRIMINATIVE ANALYSIS

Gaussian discriminative analysis solves the **binary classification problem** where:

$$\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}$$
$$\boldsymbol{y} \in \{\mathbf{0}, \mathbf{1}\}$$

The **assumptions** of the **model** are:

$$y \sim Bernoulli(\emptyset)$$

$$x \;/\; y = 0 \sim N(\mu_0, \Sigma)$$

$$x \;/\; y = 1 \sim N(\mu_1, \Sigma)$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

The **distributions** are given by:

$$
\begin{aligned}
p(y) &= \phi^y (1-\phi)^{1-y} \\
p(x|y=0) &= \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)\right) \\
p(x|y=1) &= \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right)
\end{aligned}
$$

The **parameters** of the **distributions** would be: $\boldsymbol{\phi}, \boldsymbol{\mu_0}, \boldsymbol{\mu_1}, \boldsymbol{\Sigma}$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

The **joint likelihood** of the data would be given by:

$$l(\phi, \mu_k, \Sigma) = \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}; \phi, \mu_k, \Sigma)$$

$$l(\phi, \mu_k, \Sigma) = \prod_{i=1}^{m} p(x^{(i)}/y^{(i)};\ \mu_k, \Sigma)p(y^{(i)}; \phi)$$

$$\log l(\phi, \mu_k, \Sigma) = \log \prod_{i=1}^{m} p(x^{(i)}/y^{(i)}; \mu_k, \Sigma)p(y^{(i)}; \phi)$$

$$\log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \log(p(x^{(i)}/y^{(i)}; \mu_k, \Sigma) + \log p(y^{(i)}; \phi)$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

The **joint likelihood** of the data would be given by:

$$\boldsymbol{log\, l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} log(p(x^{(i)}/y^{(i)}; \mu_k, \Sigma) + \ log\, p(y^{(i)}; \phi)}$$

$$\boldsymbol{log\, l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \mathbf{log}\left(\frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}}\right) + log\left[e^{\left(-\frac{1}{2}(x^{(i)}-\mu_k)^T \Sigma^{-1}(x^{(i)}-\mu_k)\right)}\right] + \ \mathbf{log}(\Phi^{y^{(i)}}(1-\Phi)^{1-y^{(i)}})}$$

$$\boldsymbol{log\, l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} log(\frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}}) - \frac{1}{2}\left(x^{(i)} - \mu_k\right)^T \Sigma^{-1}\left(x^{(i)} - \mu_k\right) + y^{(i)}\, log(\phi) + \left(1 - y^{(i)}\right)\mathbf{log}((1 - \Phi))}$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

Maximizing with respect to ϕ:

$$\nabla_{\emptyset} \log l(\phi, \mu_k, \Sigma) = \nabla_{\phi} \sum_{i=1}^{m} \boldsymbol{y}^{(\boldsymbol{i})} \log\left(\frac{\phi}{1-\phi}\right) + \log((1-\Phi)) = 0$$

$$\nabla_{\emptyset} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \left(\frac{\boldsymbol{y}^{(\boldsymbol{i})}}{\phi(1-\phi)} - \frac{1}{1-\phi}\right) = 0$$

$$\sum_{i=1}^{m} \frac{\boldsymbol{y}^{(\boldsymbol{i})}}{\phi(1-\phi)} = \frac{m}{1-\phi}$$

$$\boxed{\nabla_{\emptyset} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \frac{\boldsymbol{y}^{(\boldsymbol{i})}}{m} = \sum_{i=1}^{m} \frac{1(\boldsymbol{y}^{(\boldsymbol{i})} = \boldsymbol{k})}{m} = \phi = \frac{\textit{Number of data in class k}}{\textit{Total data}}}$$

GAUSSIAN DISCRIMINATIVE ANALYSIS

Maximizing with respect to $\boldsymbol{\mu_k}$:

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \nabla_{\mu_k} \sum_{i=1}^{m} -\frac{\mathbf{1}}{\mathbf{2}} \left(\boldsymbol{x}^{(\boldsymbol{i})} - \boldsymbol{\mu_k}\right)^{\boldsymbol{T}} \boldsymbol{\Sigma}^{-\mathbf{1}} \left(\boldsymbol{x}^{(\boldsymbol{i})} - \boldsymbol{\mu_k}\right) = 0$$

Applying the property $\nabla_x x^T A x = 2Ax$:

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \boldsymbol{\Sigma}^{-\mathbf{1}} \left(\boldsymbol{x}^{(\boldsymbol{i})} - \boldsymbol{\mu_k}\right) = 0$$

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \boldsymbol{\Sigma}^{-\mathbf{1}} \sum_{i=1}^{m} \left(\boldsymbol{x}^{(\boldsymbol{i})} - \boldsymbol{\mu_k}\right) = 0$$

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \boldsymbol{x}^{(\boldsymbol{i})} - \sum_{i=1}^{m} \boldsymbol{\mu_k} = 0$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

We are **only interested** in the $\boldsymbol{x}^{(i)}$ that belong to **class k**:

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \boldsymbol{x}^{(i)} - \sum_{i=1}^{m} \boldsymbol{\mu_k} = 0$$

$$\nabla_{\mu_k} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} 1\left(\boldsymbol{y}^{(i)} = \boldsymbol{k}\right)\boldsymbol{x}^{(i)} - \sum_{i=1}^{m} 1(\boldsymbol{y}^{(i)} = \boldsymbol{k})\boldsymbol{\mu_k} = 0$$

$$\boxed{\boldsymbol{\mu_k} = \frac{\sum_{i=1}^{m} 1(\boldsymbol{y}^{(i)} = \boldsymbol{k})\boldsymbol{x}^{(i)}}{\sum_{i=1}^{m} 1(\boldsymbol{y}^{(i)} = \boldsymbol{k})} = \frac{\textit{Sum of x that belong to class k}}{\textit{Class k data number}}}$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

Maximizing with respect to $\boldsymbol{\Sigma^{-1}}$:

$$\nabla_{\boldsymbol{\Sigma}^{-1}} \log l(\phi, \mu_k, \Sigma) = \nabla_{\boldsymbol{\Sigma}^{-1}} \sum_{i=1}^{m} \boldsymbol{log}\left(\frac{\mathbf{1}}{(\mathbf{2}\boldsymbol{\pi})^{d/2}|\boldsymbol{\Sigma}|^{1/2}}\right) - \frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)^{\boldsymbol{T}} \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right) = 0$$

$$\nabla_{\boldsymbol{\Sigma}^{-1}} \log l(\phi, \mu_k, \Sigma) = \nabla_{\boldsymbol{\Sigma}^{-1}} \sum_{i=1}^{m} \boldsymbol{log}\left(\frac{\mathbf{1}}{(\mathbf{2}\boldsymbol{\pi})^{d/2}|\boldsymbol{\Sigma}|^{1/2}}\right) - \frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)^{\boldsymbol{T}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)\boldsymbol{\Sigma}^{-\boldsymbol{T}} = 0$$

Because Σ is symmetrical to $\boldsymbol{\Sigma^{-1}} = \boldsymbol{\Sigma^{-T}}$:

$$\nabla_{\boldsymbol{\Sigma}^{-1}} \log l(\phi, \mu_k, \Sigma) = \nabla_{\boldsymbol{\Sigma}^{-1}} \sum_{i=1}^{m} -\frac{\boldsymbol{d}}{\mathbf{2}}\boldsymbol{log}(\mathbf{2}\boldsymbol{\pi}) + \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{log}\left(\left|\boldsymbol{\Sigma}^{-1}\right|\right) - \frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)^{\boldsymbol{T}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)\boldsymbol{\Sigma}^{-1} = 0$$

$$\nabla_{\boldsymbol{\Sigma}^{-1}} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \nabla_{\boldsymbol{\Sigma}^{-1}} \frac{\mathbf{1}}{\mathbf{2}}\boldsymbol{log}\left(\left|\boldsymbol{\Sigma}^{-1}\right|\right) - \nabla_{\boldsymbol{\Sigma}^{-1}} \frac{\mathbf{1}}{\mathbf{2}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)^{\boldsymbol{T}}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)\boldsymbol{\Sigma}^{-1} = 0$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

Applying the property $\nabla_x b^T x = b$:

$$\nabla_{\Sigma^{-1}} \log l(\phi, \mu_k, \Sigma) = \frac{\mathbf{1}}{\mathbf{2}} \sum_{i=1}^{m} \nabla_{\Sigma^{-1}} \left[\boldsymbol{log}\left(\left|\boldsymbol{\Sigma}^{-\mathbf{1}}\right|\right)\right] - \left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)^{\boldsymbol{T}} = 0$$

Applying the property $\nabla_x \log(|X|) = X^{-T}$:

$$\nabla_{\Sigma^{-1}} \log l(\phi, \mu_k, \Sigma) = \sum_{i=1}^{m} \left(\boldsymbol{\Sigma}^{-\mathbf{1}}\right)^{-T} - \left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)^{\boldsymbol{T}} = 0$$

$$m\boldsymbol{\Sigma}^{\boldsymbol{T}} - \sum_{i=1}^{m} \left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)^{\boldsymbol{T}} = 0$$

$$\boldsymbol{\Sigma}^{\boldsymbol{T}} = \frac{1}{m} \sum_{i=1}^{m} \left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_{\boldsymbol{k}}\right)^{\boldsymbol{T}}$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

Because Σ is symmetrical to $\boldsymbol{\Sigma}^{T} = \boldsymbol{\Sigma}$:

$$\boxed{\Sigma = \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu}_k\right)^{T}}$$

VARIANCE OF K-CLASS DATA

Remembering that it is an **outer product:** $\Sigma \in \mathbb{R}^{nxn}$

$$\Sigma = \begin{bmatrix} \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{1}} - \boldsymbol{\mu}_k\right)^2 & \cdots & \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{1}} - \boldsymbol{\mu}_k\right)\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{n}} - \boldsymbol{\mu}_k\right) \\ \vdots & \ddots & \vdots \\ \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{n}} - \boldsymbol{\mu}_k\right)\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{1}} - \boldsymbol{\mu}_k\right) & \cdots & \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)}{}_{\boldsymbol{n}} - \boldsymbol{\mu}_k\right)^2 \end{bmatrix}$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS

Once we have **found** the **parameters**:

$$\phi = \sum_{i=1}^{m} \frac{1(\boldsymbol{y}^{(i)} = \boldsymbol{k})}{m}$$

$$\boldsymbol{\mu_k} = \frac{\sum_{i=1}^{m} 1(\boldsymbol{y}^{(i)} = \boldsymbol{k})\boldsymbol{x}^{(i)}}{\sum_{i=1}^{m} 1(\boldsymbol{y}^{(i)} = \boldsymbol{k})}$$

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu_k}\right)\left(\boldsymbol{x}^{(i)} - \boldsymbol{\mu_k}\right)^{\boldsymbol{T}}$$

We can **start making predictions**.

GAUSSIAN DISCRIMINATIVE ANALYSIS

To make a prediction we calculate the **posterior probability** $\boldsymbol{P}$ ($\boldsymbol{y}$ = **1** / $\boldsymbol{x}$) **with** the **new data** $\boldsymbol{x}$:

$$\boldsymbol{p(y=1/x)=\frac{p(x/y=1)p(y=1)}{p(x)}}$$

We assign a threshold: if $\boldsymbol{P}$ ($\boldsymbol{y}$ = **1** / $\boldsymbol{x}$) ≥**0.5** then it belongs to class 1, otherwise it belongs to class 0. That is, we are **maximizing the probability**:

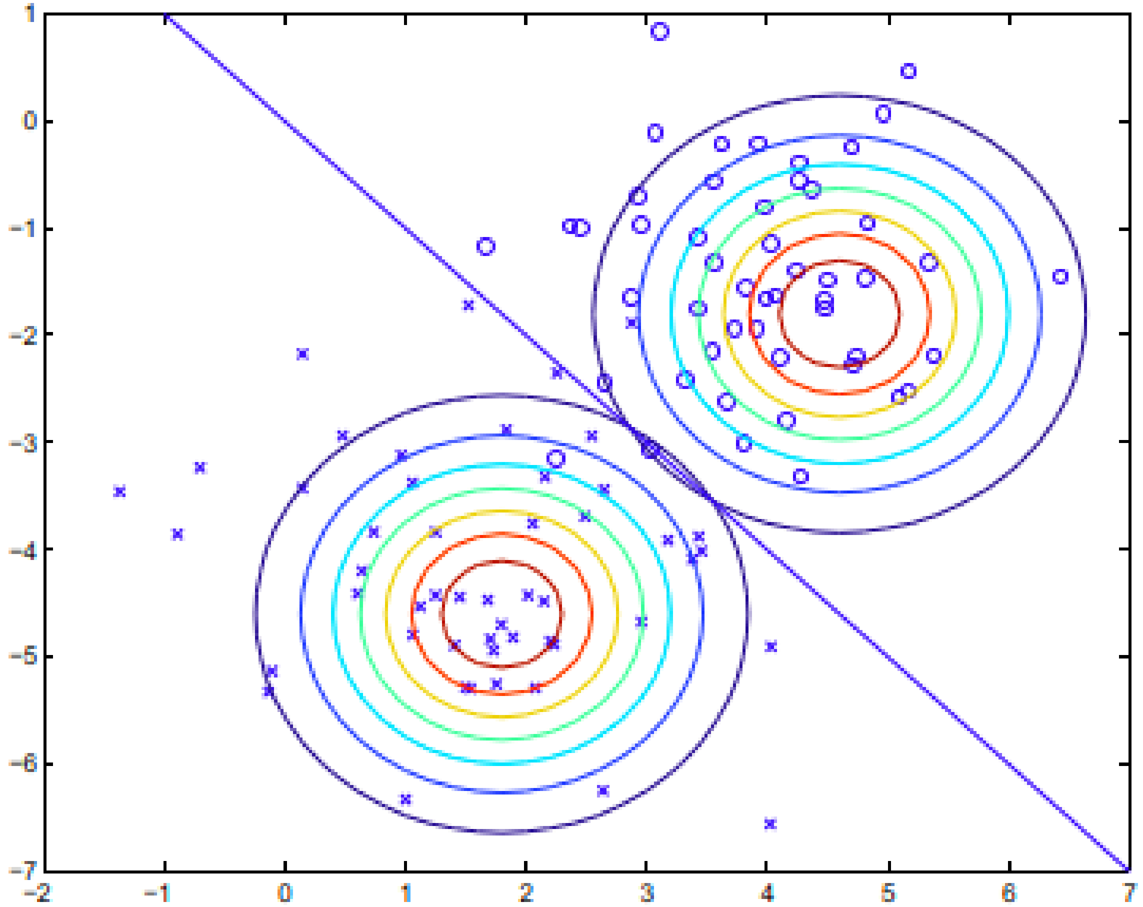
$$\boldsymbol{argmax_y\, p(y/x)=argmax_y \frac{p(x/y)p(y)}{p(x)}}$$

If $\boldsymbol{p(y=1)=p(y=0)=0.5}$ we have:

$$\boldsymbol{argmax_y p(y/x)=argmax_y\, p(x/y)}$$

GENERATIVE MODELS

GAUSSIAN DISCRIMINATIVE ANALYSIS



GAUSSIAN DISCRIMINATIVE ANALYSIS

SUMMARY:

1. Compute the parameters for each Gaussian (in binary classification it would be two curves) from the data.

$$\phi, \mu, \Sigma$$

2. Calculate the probabilities

$$P(x/y = k) \;\; P(y) \;\; P(x)$$

3. Use Bayes' Theorem to make a new prediction.

$$\boldsymbol{p(y = k/x) = \frac{p(x/y = k)p(y = k)}{p(x)}}$$

GENERATIVE MODELS

COMPARISON DISCRIMINATIVE MODELS

If we analyze the probability $p\ (y = 1\ /\ x)$, in **Bayes' Theorem**, we can realize that we obtain the **same logistic regression curve**.

$$p(y=1/x) = \frac{p(x/y=1)p(y=1)}{p(x)}$$

$$p(y=1/x) = \frac{p(x/y=1)p(y=1)}{p(x/y=1)p(y=1)+p(x/y=0)p(y=0)}\left(\frac{\frac{1}{p(x/y=1)p((y=1)}}{\frac{1}{p(x/y=1)p((y=1)}}\right)$$

$$p(y=1/x) = \frac{1}{1+\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)}}$$

GENERATIVE MODELS

COMPARISON DISCRIMINATIVE MODELS

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\frac{1}{2}\left(x^{(i)}-\mu_0\right)^T \Sigma^{-1}\left(x^{(i)}-\mu_0\right)+\frac{1}{2}\left(x^{(i)}-\mu_1\right)^T \Sigma^{-1}\left(x^{(i)}-\mu_1\right)\right)}\left(\frac{1-\phi}{\phi}\right)$$

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\frac{1}{2}\left(x^{(i)}-\mu_0\right)^T \Sigma^{-1}\left(x^{(i)}-\mu_0\right)+\frac{1}{2}\left(x^{(i)}-\mu_1\right)^T \Sigma^{-1}\left(x^{(i)}-\mu_1\right)\right)}\left(e^{\log\left(\frac{1-\phi}{\phi}\right)}\right)$$

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\frac{1}{2}\Sigma^{-1}\left[\left(x^{(i)}-\mu_0\right)^T\left(x^{(i)}-\mu_0\right)-\left(x^{(i)}-\mu_1\right)^T\left(x^{(i)}-\mu_1\right)\right]+\log\left(\frac{1-\phi}{\phi}\right)\right)}$$

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\frac{1}{2}\Sigma^{-1}\left[\sum_{j=1}^{n}\left(x^{(i)}{}_j-\mu_0\right)^2-\left(x^{(i)}{}_j-\mu_1\right)^2\right]+\log\left(\frac{1-\phi}{\phi}\right)\right)}$$

GENERATIVE MODELS

COMPARISON DISCRIMINATIVE MODELS

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\frac{1}{2}\Sigma^{-1}\left[\sum_{j=1}^{n} -2\mu_0 x^{(i)}{}_j + \mu_0{}^2 + 2\mu_1 x^{(i)}{}_j - \mu_1{}^2\right] + \log\left(\frac{1-\phi}{\phi}\right)\right)}$$

Because $x^{(i)}{}_0 = 1$

$$\frac{p(x/y=0)p(y=0)}{p(x/y=1)p((y=1)} = e^{\left(-\left[\sum_{j=1}^{n} \frac{1}{2}\Sigma^{-1}(\mu_0-\mu_1)\left(x^{(i)}{}_j\right)\right] + \left[\frac{1}{2}\Sigma^{-1}\left(\mu_0{}^2-\mu_1{}^2\right) + \log\left(\frac{1-\phi}{\phi}\right)\right]\left(x^{(i)}{}_0\right)\right)}$$

Comparing $w^T X$:

$$w^T = \begin{bmatrix} -\frac{1}{2}\Sigma^{-1}\left(\mu_0{}^2 - \mu_1{}^2\right) - \log\left(\frac{1-\phi}{\phi}\right) \\ \frac{1}{2}\Sigma^{-1}(\mu_0 - \mu_1) \\ \vdots \\ \frac{1}{2}\Sigma^{-1}(\mu_0 - \mu_1) \end{bmatrix}$$

GENERATIVE MODELS

COMPARISON DISCRIMINATIVE MODELS

Therefore:

$$p(y=1/x)=\frac{1}{1+e^{-\left(\left[\sum_{j=1}^{n}\frac{1}{2}\Sigma^{-1}(\mu_0-\mu_1)\left({x^{(i)}}_j\right)\right]+\left[-\frac{1}{2}\Sigma^{-1}\left({\mu_0}^2-{\mu_1}^2\right)+\log\left(\frac{1-\phi}{\phi}\right)\right]\left({x^{(i)}}_0\right)\right)}}=\boxed{\frac{1}{1+e^{-W^TX}}}$$

Where w^T :

$$w^T=\begin{bmatrix}-\frac{1}{2}\Sigma^{-1}\left({\mu_0}^2-{\mu_1}^2\right)-\log\left(\frac{1-\phi}{\phi}\right)\\ \frac{1}{2}\Sigma^{-1}(\mu_0-\mu_1)\\ \vdots\\ \frac{1}{2}\Sigma^{-1}(\mu_0-\mu_1)\end{bmatrix}$$

GENERATIVE MODELS

COMPARISON DISCRIMINATIVE MODELS

In a general fashion:

Likelihood function

$$x/y \sim ExponentialFamily(\eta)$$

$$\Downarrow \;\; \not\Uparrow$$

Posterior Distribution

$$P(y/x) = sigmoid\ function$$

THAT'S WHY LOGISTIC REGRESSION IS USED→ WORKS FOR MANY TYPES OF ASSUMPTIONS

COMPARISON DISCRIMINATIVE MODELS

A **generative model** will have **better performance** than a discriminative one, if the assumption that we made of the shape of the distribution $P(x / y)$ holds for the real data (more information is provided to the algorithm).

Otherwise, if the data does not behave as we assumed, the **discriminative model** will have **better results**, because even though our assumptions were not as accurate, **logistic regression** works for many different assumptions.

GENERATIVE	DISCRIMINATIVE
Better performance when the P (x / y) distribution is known	Better performance when P (x / y) is unknown (robust to incorrect assumptions)
Asymptotic efficiency	There may be a better model
Needs little data	Needs a lot of data

DISCRIMINATIVE AND GENERATIVE MODELS

Naive Bayes Classifier

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

BERNOULLI MULTIVARIATE EVENT MODEL:

For the case of the **Gaussian discriminative analysis**, it was the case that $\boldsymbol{x^{(i)}}_{\boldsymbol{j}} \in \mathbb{R}$**,** that is, they were **continuous values**.

For cases, such as **text classification** (“desired mail" and “spam mail") the **x vectors** can have a very **high dimensionality** (the number of words is immense).

$$\boldsymbol{x} \in \{\mathbf{0}, \mathbf{1}\}^{\mathbf{5000}}$$

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \begin{array}{l} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \text{buy} \\ \vdots \\ \text{zygmurgy} \end{array}$$

$\mathbf{2^{5000}}$ **parameters** would be needed.

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

SOLUTION: naive Bayes assumption (very strong)

We suppose that $\boldsymbol{x^{(i)}}_{\boldsymbol{j}}$ are **conditionally independent** given $\boldsymbol{y}$.

That is, if a text is known to be "**spam**" $\boldsymbol{y} = \mathbf{1}$, the fact that the word x_{2087} = "buy" appears in the text does not affect beliefs about the appearance of any other word, such as x_{39831} = "price".

$$\boldsymbol{p}(\boldsymbol{x_1}, \ldots \boldsymbol{x_j}, \ldots, \boldsymbol{x_n}/\boldsymbol{y}) = \boldsymbol{p}(\boldsymbol{x_1})\boldsymbol{p}(\boldsymbol{x_2}/\boldsymbol{x_1}, \boldsymbol{y})\boldsymbol{p}(\boldsymbol{x_3}/\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{y}) \ldots = \boldsymbol{p}(\boldsymbol{x_1}/\boldsymbol{y})\boldsymbol{p}(\boldsymbol{x_2}/\boldsymbol{y}) \ldots = \prod_{j=1}^{n} \boldsymbol{p}(\boldsymbol{x_j}/\boldsymbol{y})$$

EVEN WHEN THE ASSUMPTION IS NOT TRUE, THE ALGORITHM PERFORMS GOOD IN MANY APPLICATIONS

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

As the **Bayes classifier** is a **generative model**, it is of interest to model the distributions $\boldsymbol{p(x_j/y)}$ and $\boldsymbol{p(y)}$. In particular, we want to **maximize the joint likelihood**:

$$l\left(\phi_y, \phi_{j/y=0}\phi_{j/y=1}\right) = \prod_{i=1}^{m} p\left(x^{(i)}, y^{(i)}\right)$$

Where the **assumptions** are:

$$p(x/y) = p\left(x_1, \dots x_j, \dots, x_n/y\right) = \prod_{j=1}^{n} p(x_j/y)$$

$$x_j \,/\, y = 0 \sim Bernoulli(\phi_{j/y=0})$$

$$x_j \,/\, y = 1 \sim Bernoulli(\phi_{j/y=1})$$

$$y \sim Bernoulli(\phi_j)$$

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

The **result** of the **maximum likelihood** estimation gives:

$$\phi_{j/y=1} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 1\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}} = \frac{\#times\ the\ word\ j\ is\ repeated\ in\ "spam"}{total\ \text{"spam"}\ mail}$$

$$\phi_{j/y=0} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}} = \frac{\#times\ the\ word\ j\ is\ repeated\ in\ "desired"}{total\ \text{"desired"}\ mail}$$

$$\phi_y = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}{m} = \frac{total\ spam\ mail}{total\ mail}$$

If the **characteristics $\boldsymbol{x_j^{(i)}}$** take **more values**, they can be **modeled as multinomials**.

If the **characteristics $\boldsymbol{x_j^{(i)}}$** take **continuous values**, they are **discretized in intervals**.

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

To make a **new prediction** we use **Bayes' theorem**:

$$p(y=1/x) = \frac{p(x/y=1)p(y=1)}{p(x)}$$

Where:

$$p(x/y=1) = \prod_{j=1}^{n} p(x_j/y=1) = \prod_{j=1}^{n} (\phi_j/y=1)$$

$$p(y) = \phi_y = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1\}}{m}$$

$$p(x) = \prod_{j=1}^{n} p\left(x_j/y=1\right) p(y=1) + \prod_{j=1}^{n} p\left(x_j/y=0\right) p(y=0)$$

What happens if a word appears in an email that the algorithm has never seen in another email?

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

The **probability** that it will assign, of seeing the word in **either of the two emails** will be **0**.

$$\phi_{j/y=1} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 1\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}} = \frac{\#times\ the\ word\ j\ is\ repeated\ in\ "spam"}{total\ \text{"spam"}\ mail} = \frac{0}{m_{neg}}$$

$$\phi_{j/y=0} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}} = \frac{\#times\ the\ word\ j\ is\ repeated\ in\ "desired"}{total\ \text{"desired"}\ mail} = \frac{0}{m_{pos}}$$

When making the **prediction** we will have an **incongruity**:

$$\boldsymbol{p(y = 1/x) = \frac{p(x/y = 1)p(y = 1)}{p(x)} = \frac{0}{0}}$$

STATISTICALLY IT IS NOT ADEQUATE TO SAY THAT THE PROBABILITY OF AN EVENT IS ZERO JUST BECAUSE YOU HAVE NOT SEEN IT IN YOUR DATA

NOTE: LAPLACE AND THE SUN

LAPLACE SMOOTHING:

The **solution** is to **change our estimate** (**general multinomial case** where y = {1, 2,…, k}):

$$p(y=1) = \frac{(\#\,\text{"1"s} + 1)}{(\#\text{"0"s} + 1) + (\#\,\text{"1"s} + 1)}$$

$$p(y=j) = \phi_y = \frac{1 + \sum_{i=1}^{m} 1\{y^{(i)} = j\}}{m + k}$$

When calculating the **other estimators** we have:

$$\phi_{j/y=1} = \frac{1 + \sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 1\}}{2 + \sum_{i=1}^{m} 1\{y^{(i)} = 1\}}$$

$$\phi_{j/y=0} = \frac{1 + \sum_{i=1}^{m} 1\{x_j^{(i)}, y^{(i)} = 0\}}{2 + \sum_{i=1}^{m} 1\{y^{(i)} = 0\}}$$

NAIVE BAYES CLASSIFIER

MULTINOMIAL EVENT MODEL:

Let's see the case when $\boldsymbol{x_j} = \{\mathbf{1}, \mathbf{2}, \dots, \boldsymbol{k}\}$.

For example, now the value of $\boldsymbol{x_j}$ represents the **position** of the **word in** the **dictionary**, and the index $\boldsymbol{j}$ indicates the **position** of the **word in** the **mail**.

So $\boldsymbol{n}$ now represents the **length of the email** (and varies according to each email).

The **assumptions** would be:

$$p(x/y) = p\left(x_1, \dots x_j, \dots, x_n/y\right) = \prod_{j=1}^{n} p(x_j/y)$$

$$x_j \,/\, y = 0 \sim Multinomial(\phi_{j/y=0})$$

$$x_j \,/\, y = 1 \sim Multinomial(\phi_{j/y=1})$$

$$y \sim Bernoulli(\phi_j)$$

GENERATIVE MODELS

NAIVE BAYES CLASSIFIER

The **parameters**, when calculating the **maximum likelihood** would be (with **Laplace smoothing**):

$$\phi_{k/y=1} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\{x_j^{(i)} = k, y^{(i)} = 1\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}\, d_i + n_i + k} = \frac{\#\ of\ times\ that\ the\ word\ k\ appears\ in\ mails\ ND}{total\ words\ in\ mails\ ND}$$

$$\phi_{k/y=0} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n_i} 1\left\{x_j^{(i)} = k, y^{(i)} = 0\right\} + 1}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}\, d_i + n_i + k} = \frac{\#\ of\ times\ that\ the\ word\ k\ appears\ in\ mails\ D}{total\ words\ in\ mails\ D}$$

$$\phi_y = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\} + 1}{m}$$

BINARY EVALUATION METRICS

BINARY EVALUATION METRICS

DEFINITIONS

False positive (error type I):

- **Statistics:** A **true null hypothesis** is **incorrectly rejected.**
- **Machine learning:** the model **predicts** that the **class of a training data p (y / x) = 1** is **positive,** when the **reality** is that the **data belonged** to the **negative class y = 0.**

BINARY EVALUATION METRICS

DEFINITIONS

False negative (error type II):

- **Statistics:** The **false null hypothesis** is **incorrectly accepted.**
- **Machine learning:** the model **predicts** that the **class** of a **training data** $\boldsymbol{p}$ **(**$\boldsymbol{y}$ **/** $\boldsymbol{x}$**) = 0** is **negative,** when the **reality** is that the **data belonged** to the **positive class** $\boldsymbol{y}$ **= 1.**

BINARY EVALUATION METRICS

DEFINITIONS

True positive:

- **Statistics:** The **false null hypothesis** is **correctly accepted.**
- **Machine learning:** the model **predicts** that the **class** of a **training data $p\ (y\ /\ x) = 1$ is positive,** when the **reality** is that the **data belonged** to the **positive class $y = 1$.**

BINARY EVALUATION METRICS

DEFINITIONS

True negative:

- **Statistics:** The **true null hypothesis** is **correctly accepted.**
- **Machine learning:** the model **predicts** that the **class** of a **training data $\boldsymbol{p}$ ($\boldsymbol{y}$ / $\boldsymbol{x}$) = 0 is negative,** when the **reality** is that the **data belonged** to the **negative class $\boldsymbol{y}$ = 0.**

BINARY EVALUATION METRICS

CONFUSION MATRIX

		PREDICTIVE VALUES	
		POSITIVE (1)	NEGATIVE (0)
ACTUAL VALUES	POSITIVE (1)	TP	FN
	NEGATIVE (0)	FP	TN

BINARY EVALUATION METRICS

SENSITIVITY

Sensitivity (“**true positive rate**”, “**recall**”): measures the **performance** of the model to **predict** the **positive class** $y = 1$.

It is **difficult** for a **sensitive algorithm** to make a **mistake** in **predicting** the **positive class**.

Still, **high sensitivity** can be **accompanied** by many **false positives**.

$$\boldsymbol{Sens = \frac{TP}{TP + FN}}$$

BINARY EVALUATION METRICS

SPECIFICITY

Specificity (“**true negative rate**”): measures the **performance** of the model to **predict** the **negative class** $\boldsymbol{y}$ = 0.

It is **difficult** for a **specitive algorithm** to make a **mistake** in **predicting** the **negative class**.

Still, **high specitivity** can be **accompanied** by many **false negatives**.

$$\boldsymbol{Spec = \frac{TN}{TN + FP}}$$

BINARY EVALUATION METRICS

A C C U R A C Y

Accuracy (**not to be confused** with "**positive predictive value**" or "**precision**") measures the **performance** of the **model** in a **general way**. How much the **predictions deviate** from the **actual values**.

$$\boldsymbol{Acc = \frac{TP + TN}{TP + TN + FN + FP}}$$

BINARY EVALUATION METRICS

POSITIVE PREDICTIVE VALUE

The "**positive predictive value**" or "**precision**" measures the **fraction of positives** that were **correctly predicted**.

$$\boldsymbol{PPV = \frac{TP}{TP + FP}}$$

BINARY EVALUATION METRICS

F1 SCORE

The **F1 score** is the **harmonic mean** (average for rates) **between** the **recall** and the **PPV**:

$$\boldsymbol{Sens = \frac{TP}{TP + FN}}$$

$$\boldsymbol{PPV = \frac{TP}{TP + FP}}$$

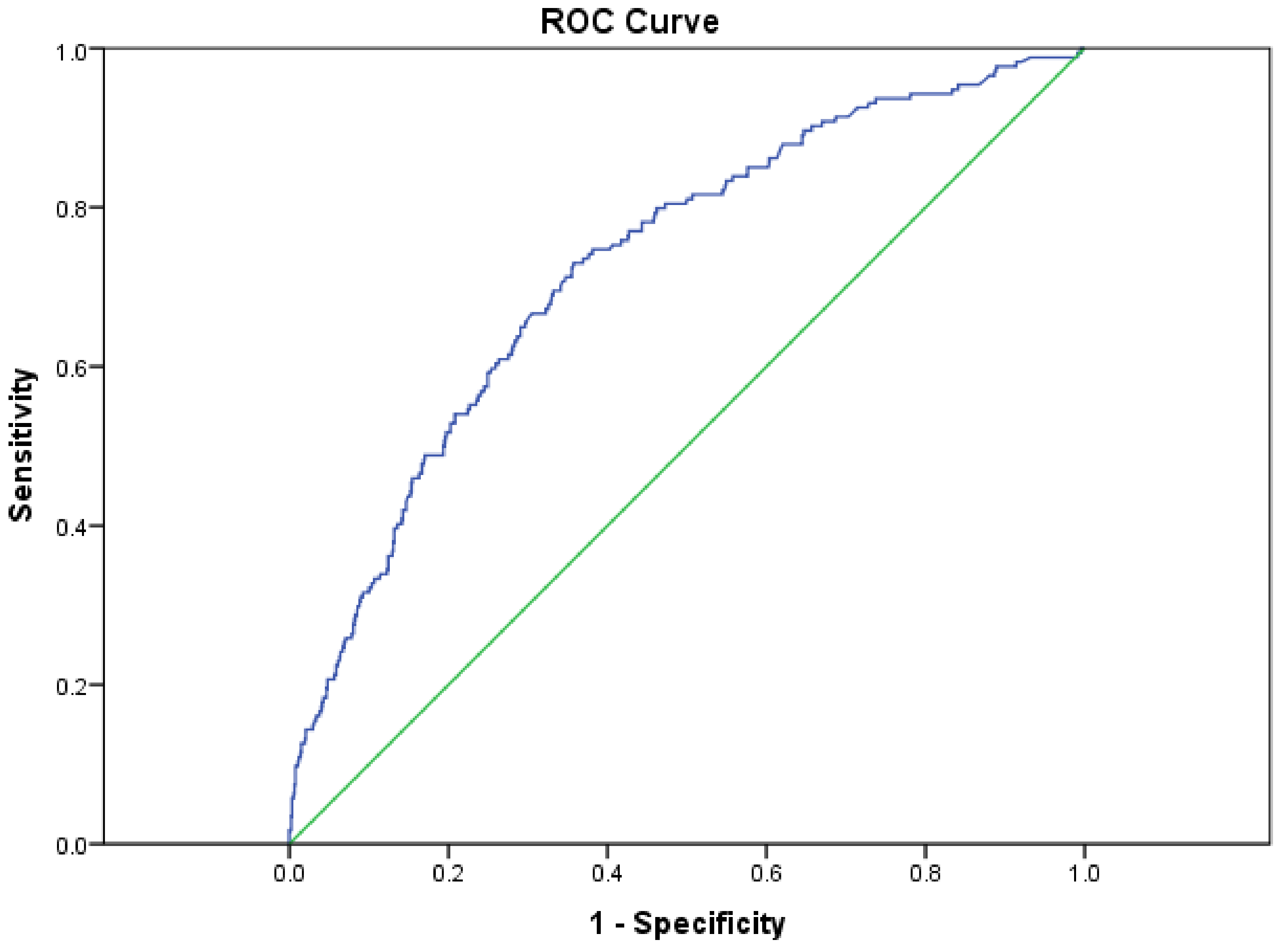
$$\boldsymbol{F1 = \frac{2}{Sens^{-1} + PPV^{-1}}}$$

BINARY EVALUATION METRICS

ROC CURVE

FPR vs TPR

The threshhold is varied



Diagonal segments are produced by ties.

BINARY EVALUATION METRICS

ROC CURVE