

# Logistic and Linear regression

## Using python.

Iván Andrés Trujillo Abella  
Dayana Ibeth Castro Guevara

Facultad de Ingenieria  
Pontificia Universidad Javeriana

**`trujilloiv@javeriana.edu.co`**

# Preamble

For this lesson you need remember some concepts as **Probability Distribution Function(PDF)**, **joint distribution function**, **expectation** and **variance** of random variables please check the following material:

## Lecture

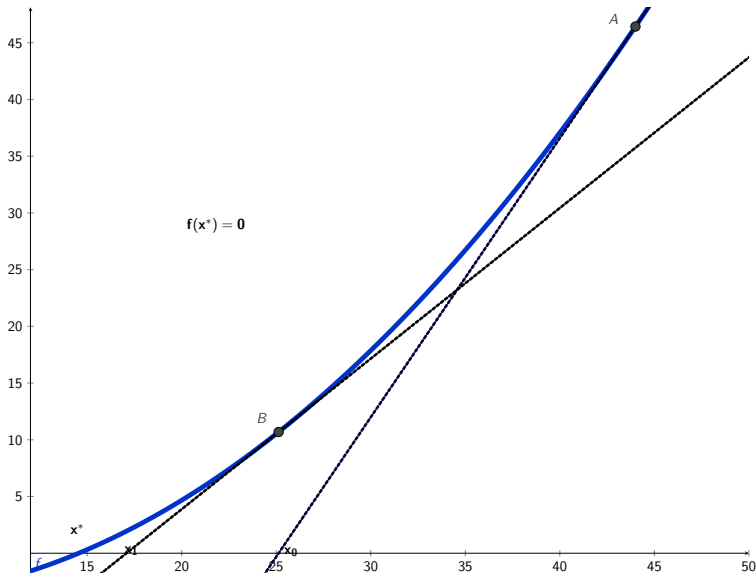
- *Notes of probability and inferential statistics as background to linear regression analysis using python.* **(Click here)**

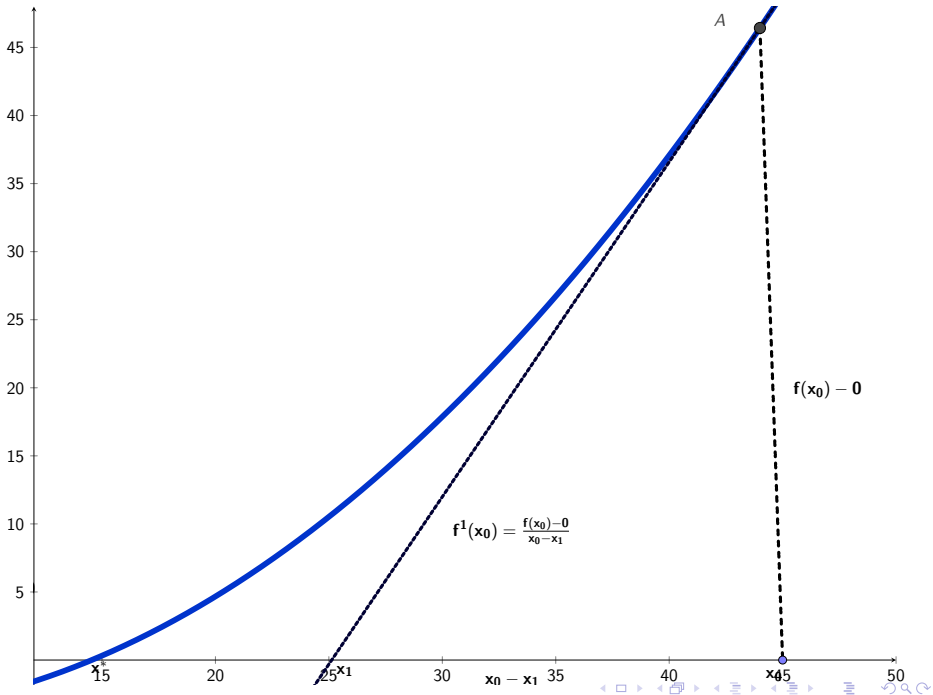
# Introduction

Logistic regression is used broadly in empirical works, it is used in economics, engineering, epidemiology and clinical research. In a simplified way the logistic regression is used to binary problems.

If we take a point near to the change in concavity the method could produce divergence.

# illustration





$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \quad (1)$$

Thus in  $i$  iteration we have:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \quad (2)$$

We select the point  $A$  the tangent line cross the x-axis in  $x_0$  and again in  $f(x_0)$  and again the tangent line of this point ( $B$ ) cross the x-axis in  $x_1$  each step is also known as a iteration. The algorithm converge to the the value  $x^*$  think that this could be used to *optimization problems*.

# Newton-raphson program

---

```
def quadratic(a,b,c, x):  
    return a*x**2 + b*x + c  
def dfQuadratic(a,b,x):  
    return a*x**2 + b*x  
a,b,c ,x0 = 1,-3,-4,8  
def raphsonQuadratic(a,b,c,x0, error_max=0.0000015,  
    iteration_max=100):  
    xi = x0  
    iter, error = 0, 100  
    data = []  
    while (iter < iteration_max) and (error > error_max):  
        xj = xi - quadratic(a,b,c,xi) / dfQuadratic(a,b,xi)  
        error = abs(xj - xi)  
        iter += 1  
        data.append((xj,xi,error,iter))  
        xi = xj  
    return data  
raphsonQuadratic(a,b,c,4)[-1]
```



# Gradient descent

## Guided laboratory

Construct a program given the process explained by the instructor.

# Prior, Posterior and Likelihood

Assume that  $\Theta$  is unknown parameter and  $\mathbf{X}$  it is a set of features or information collected. Therefore will be defined:

- $P(\Theta)$  prior distribution
- $P(\Theta | \mathbf{X})$  posterior distribution
- $P(\mathbf{X} | \Theta)$  likelihood

# Max a posteriori

A set of features  $\mathbf{X} = \{x_1, \dots, x_n\}$  assuming a distribution  $P(\mathbf{X}, \Theta)$  where  $\Theta$  is a parameter (a random variable).

$$\Theta_{ma} = \max P(\Theta | \mathbf{X}) \quad (3)$$

the last equation must be compared regarding the maximum likelihood estimation. that establish the  $\max P(\mathbf{X} | \Theta)$ .

# Problem

## Problem

We need find the probability that we draw exactly 3 damaged computers of 5 if in the batch there are 12 computers and 5 of them fail.

# Generalization of the problem

**The problem:** Find the probability of get  $(i)$  in  $(j)$  where  $(m)$  fill a property in a total of  $(N)$ .

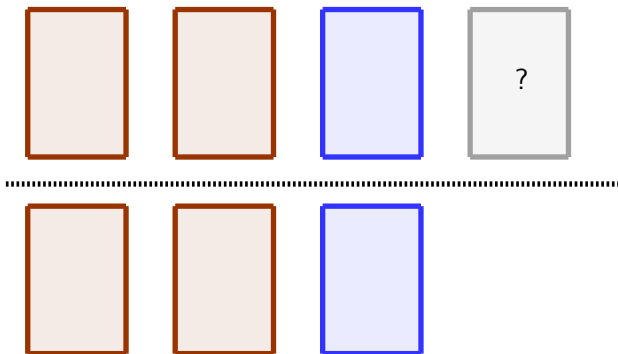
deriving

there are  $\binom{m}{i}$  possible ways of get  $i$  among  $m$  and by each one of the possible combination there are  $\binom{N-m}{j-i}$  in a total of  $\binom{N}{j}$  combinations, therefore the probability will be:

$$\frac{\binom{m}{i} \binom{N-m}{j-i}}{\binom{N}{j}} \quad (4)$$

(??) is also known as **Hypergeometric distribution**.

# Freund illustration



# Estimate $\Theta$ by maximum likelihood

The **Method of maximum likelihood** consist in estimate the parameter  $\Theta$  the number of *red cards* that maximize the probability of see the **data** (three red cards and a blue car), in this case  $\Theta$  could be 2 or 3.

$$\frac{\binom{3}{2} \binom{1}{1}}{\binom{4}{3}} > \frac{\binom{2}{2} \binom{2}{1}}{\binom{4}{3}} \quad (5)$$

in Scipy the notation is:

$$p(k, M, n, N) = \frac{\binom{n}{k} \binom{M-n}{N-k}}{\binom{M}{N}} \quad (6)$$

---

```
from scipy.stats import hypergeom
k,M,n,N = 2,4,3,3
print(hypergeom.pmf(k,M,n,N))
k, M,n,N = 2,4,2,3
print(hypergeom.pmf(k,M,n,N))
```

---



# Maximum Likelihood Estimation (MLE)

Suppose that your data is generated by a theoretical distribution, the inverse problem is to determine the most probable parameter that generated the data.

# Insights about MLE

we are going to say in a general term that  $f(x_i)$  is PDF or PMF of a random variable.

# Problem

you have  $n$  balls in a bag where there are  $j$  reds and  $k$  black thus  $n = j + k$ . However you do not know the really proportion of each one color. if you draw balls and both are different what is the proportion of black balls  $\Theta$ .

## MLE insight

Then we choose a  $\Theta$  among all posible values that maximize the probability of seen the data.

# Problem

In a formal way we can assume that  $x \sim B(n, \Theta)$  and we have seen the data results  $\{x_1, x_2, x_3, \dots, x_n\}$

$$P(X = x_1 \cap X = x_2, \dots, \cap X = x_n) \quad (7)$$

The variables are i.i.d and therefore the joint probability is the result of multiply the marginal probabilities.

$$P(X = x_1 \cap X = x_2, \dots, \cap X = x_n) = \prod_{i=1}^n P(x_i) \quad (8)$$

# Problem

$$\begin{aligned}\prod_{i=1}^n P(x_i) &= \prod_{i=1}^n \binom{m}{x_i} \Theta^{x_i} (1 - \Theta)^{m-x_i} \\ &= \prod \binom{m}{x_i} \prod \Theta^{x_i} \prod (1 - \Theta)^{m-x_i} \\ &= \prod \binom{m}{x_i} \Theta^{\sum x_i} (1 - \Theta)^{\sum (m-x_i)}\end{aligned}\tag{9}$$

$$\ln(\prod P(x_i)) = \ln\left(\binom{m}{x_i}\right) + \sum x_i \ln(\Theta) + (nm - \sum x_i) \ln(1 - \Theta)$$

$$\begin{aligned}
 L(\Theta) &= \ln\left(\prod P(x_i)\right) \\
 \frac{dL(\Theta)}{d\Theta} &= \frac{\sum x_i}{\Theta} - \frac{nm - \sum x_i}{1 - \Theta} = 0 \\
 \frac{1}{\Theta} &= \frac{nm - \sum x_i}{\sum x_i} + 1 \\
 \Theta^* &= \frac{\sum x_i}{nm}
 \end{aligned} \tag{10}$$

Now to proof that  $L(\Theta^*)$  is the maximum probability we must show that  $\frac{d^2L(\Theta)}{d\Theta^2} \big|_{\Theta=\Theta^*} < 0$ .



# About estimators

The quality of estimators:

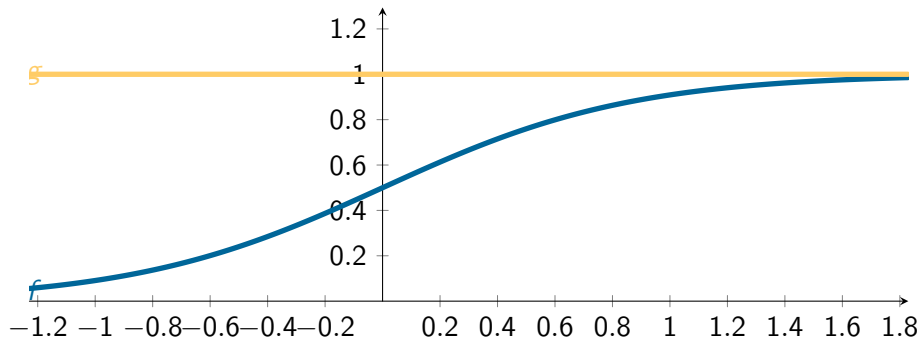
- **Unbiased estimator** : The mean of the estimator is equal to the mean of parameter.
- **Variance**: the dispersion of the estimations regarding the mean value of the same.
- **quadratic value mean**: offer information about another two.



Suppose that do you have a function  $f(x, y) = x^2 + y^2$ , then we can define the gradient as follow:

$$\nabla f(x, y) = \begin{bmatrix} \frac{df}{dx} \\ \frac{df}{dy} \end{bmatrix} \quad (11)$$

# Logistic equation



# logistic equation

$$f(x) = \frac{\kappa}{1 + e^{-\alpha(x-x_0)}} \quad (12)$$

Where  $\kappa$  it is the maximun value.

# logistic equation

## Population growth

$$\frac{dy}{dt} = ry\left(1 - \frac{y}{k}\right) \quad (13)$$

# python implementation

## Statsmodels

---

```
formula = 'died~studytime + C(drug) ' # logit died studytime
        i.drug
model = smf.logit(formula= formula, data=df)
results = model.fit()
print(results.summary())
coefs = pd.DataFrame({
    'coef': results.params.values,
    'odds ratio': np.exp(results.params.values),
    'name': results.params.index
})
coefs
```

---

# Solve

with newton method solve the following:

$$e^x = 4 - x^2 \quad (14)$$

las dos soluciones son:  $x_1 = 1.05$   $x_2 = -1.96$

# Contingency table

		Diagnose	
		Disease	No-Disease
Risk Factor	Smoke	a	b
	Not Smoke	c	d

From this table we can calculate the odds, increase the probability of a diagnose smoke, or not?

$$odds_{smoke} = \frac{a}{b} = \frac{P(D \mid smoke)}{P(ND \mid smoke)} \quad (15)$$

e

# Odds ratio

We can calculate the odds ratio as:

$$OR = \frac{odds_{smoke}}{odds_{NoSmoke}} \quad (16)$$

according to the information of the table, we can calculate the odds ratio as  $OR = \frac{\frac{a}{b}}{\frac{c}{d}}$ .

How we can interpret this  $OR$ ?



# From odds to probability

$$\frac{P(A)}{P(A^c)} + 1 = odds + 1 \quad (17)$$

$$\frac{P(A) + P(A^c)}{P(A^c)} = odds + 1 \quad (18)$$

$$\frac{1}{P(A^c)} = odds + 1 \quad (19)$$

$$\frac{odds}{odds + 1} = P(A) \quad (20)$$

# Odds Ratio

---

```
testing = pd.DataFrame({  
    'smoke': [1,1,0,1,1,0,1,1,0,1,0,0,1,1,1,0,0,1,1,1],  
    'diagnose': [1,0,1,1,0,0,1,1,0,0,0,1,1,0,0,0,0,0,1,0]})  
pd.crosstab(testing['smoke'], testing['diagnose'])
```

---

```
# Odds ratio  
oddsnum = 6/7  
oddsdem = 2/5  
print(oddsnum, oddsdem, oddsnum/oddsdem)
```

---

# Logistic regression

# Using statsmodels

---

```
import statsmodels.formula.api as smf
model_logit = smf.logit(formula="diagnose ~ smoke", data=testing)
res = model_logit.fit()
# How we can convert to dummy variable??
res.summary()
np.exp(res.params) # This give us the odds ratio getted
                    previously.
```

---

# ORs lessert than 1

$$(1 - OR) * 100 \quad (21)$$

# Linear regression

When it is useful the linear regression?

for instance, the relation among the income and compsuition, body mas index and levels of microalbum, etc, the wage and years of experience, unlike logistic regression the indepent variable not is dicotomy.

# A linear model

Where are going consider that our prediction  $\hat{y}_i$  differ from the real value  $y_i$  in  $u_i$ :

$$y_i - \hat{y}_i = u_i \quad (22)$$

and we are going to model  $\hat{y}_i = \beta_0 + \beta_1 x_i$ , therefore:

$$y_i = \beta_0 + \beta_1 x_i + u_i \quad (23)$$

Where  $y_i$  is our independent term,  $\beta_0$  is a constant.

# MLE

We have a collection of  $\vec{y} = y_1, y_2, \dots, y_n$ , that our objective is build a model that allow us to find the value of  $\beta_0, \beta_1$  and maximize the probability of seen the data( $\vec{y}$ ).



Assume that

$$y_i = \beta_0 + \beta_1 x_i + u_i \quad (24)$$

and

$$u_i \sim N(0, \sigma^2) \quad (25)$$

Therefore the  $E(y_i) = \beta_0 + \beta_1 x_i$  and  $\text{var}(y_i) = \sigma^2$  (why?).

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2) \quad (26)$$

If we make distributional assumptions we could make inference and hypothesis testing.

# PMF normal distribution

The PMF of a normal variable is:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x - \mu}{\sigma}\right)^2\right) \quad (27)$$

where  $\mu$  is mean. Therefore for our problem.

$$f(y_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y_i - \beta_0 - \beta_1 x_i}{\sigma}\right)^2\right) \quad (28)$$

# Likelihood function

$$L(\beta_0, \beta_1, \sigma^2) = \prod_{i=1}^n f(y_i) \quad (29)$$

$$\prod_{i=1}^n f(y_i) = \prod_{i=1}^n \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{y_i - \beta_0 - \beta_1 x_i}{\sigma}\right)^2\right) \quad (30)$$

Now remember that  $e^x \cdot e^y = e^{x+y}$  therefore the *likelihood function* is rewritten as:

$$\frac{1}{(\sigma\sqrt{2\pi})^n} \exp\left(\frac{-1}{2\sigma^2} \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2\right) \quad (31)$$

# Log likelihood function

Apply natural logarithm to the likelihood function

$$\ln(L(\beta_0, \beta_1, \sigma^2)) = -n \ln(\sigma \sqrt{2\pi}) - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2 \quad (32)$$

now we need establish the partial derivatives for  $\beta_0$  and  $\beta_1$ . Notice that to maximize the expression we need minimize  $\sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2$ .

$$\frac{\partial \ln(L)}{\partial \beta_0} = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) = 0 \quad (33)$$

$$\frac{\partial \ln(L)}{\partial \beta_1} = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) x_i = 0 \quad (34)$$

We are going to reach the same equations minimizing the error among the prediction and the real value of  $y_i$ .

# Solving the problem

We have an analytical solution!.

# Ordinary least squares

$$u = \sum (y_i - \hat{y}_i)^2 \quad (35)$$

note that  $\hat{y}_i = \beta_0 + \beta_1 x_i$ .

$$u^2 = \sum (y_i - \beta_0 - \beta_1 x_i)^2 \quad (36)$$

the first order condition require  $\frac{\partial u^2}{\partial \beta_0} = 0$ ,  $\frac{\partial u^2}{\partial \beta_1} = 0$ .

# Chain rule

to get

$$\frac{\partial u^2}{\partial \beta_0} = \frac{\partial u^2}{\partial u} \frac{\partial u}{\partial \beta_0} \quad (37)$$

$$\frac{\partial u^2}{\partial \beta_1} = \frac{\partial u^2}{\partial u} \frac{\partial u}{\partial \beta_1} \quad (38)$$

Remember that  $\frac{d \sum g_i(x)}{dx} = \sum \frac{dg_i(x)}{dx}$ . Therefore  $\frac{\partial u^2}{\partial u} = 2u$  and  $\frac{\partial u}{\partial \beta_0} = -1$ ,  
 $\frac{\partial u}{\partial \beta_1} = -x_i$ .



# Partial derivatives

$$\frac{\partial u^2}{\partial \beta_0} = -2 \sum (y_i - \beta_0 - \beta_1 x_i) \quad (39)$$

$$\frac{\partial u^2}{\partial \beta_1} = -2 x_i \sum (y_i - \beta_0 - \beta_1 x_i) \quad (40)$$

note that this will be zero if we know the exactly parameters.

# Solving

## Analytically

$$\hat{\beta}_1 = \frac{\sum (y_i - \bar{y})(x_i - \bar{x})}{\sum (x_i - \bar{x})^2} = \frac{\text{cov}(x, y)}{\text{var}(x)}. \quad (41)$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \quad (42)$$

To see the full derivation of ?? and ?? **click here**

# Causality

Correlation not imply cuasality!!

# What is the correct interpretation?

# Multiple Linear regression

$$y_i \sim N(\beta^\top x_i, \sigma^2) \quad (43)$$

The likelihood function  $\beta, \mathbf{x} \in \mathbf{R}^d$ .

$$\prod_{i=1}^n \frac{1}{(\sigma\sqrt{2\pi})^n} \exp\left(\frac{-1}{2\sigma^2} \sum_{i=1}^n (y_i - \beta^\top x_i)^2\right) \quad (44)$$

$$\frac{1}{(\sigma\sqrt{2\pi})^n} \exp\left(\frac{-1}{2\sigma^2} \sum_{i=1}^n (y_i - \beta^\top x_i)^2\right) \quad (45)$$

You can check the appendix in matricial operations

$$\frac{1}{(\sigma\sqrt{2\pi})^n} \exp\left(\frac{-1}{2\sigma^2} (\mathbf{y} - \beta^\top \mathbf{x})^\top (\mathbf{y} - \beta^\top \mathbf{x})\right) \quad (46)$$