

# Actividad 4

Jose Pablo Salazar Velazquez

Febrero 2016

## 1 Introducción

En el área de la estadística, Mínimos cuadrados es una técnica de análisis numérico enmarcada dentro de la optimización matemática, en la que, dados un conjunto de pares ordenados: variable independiente, variable dependiente, y una familia de funciones, se intenta encontrar la función continua, dentro de dicha familia, que mejor se aproxime a los datos (un “mejor ajuste”), de acuerdo con el criterio de mínimo error cuadrático

La técnica de los mínimos cuadrados es parte de ambas secciones en el análisis de la regresión, pues se puede utilizar tanto en la regresión lineal como en el no lineal. En la actividad de hoy, se utilizaron ambos métodos para poder ajustar funciones del tipo lineal y exponencial a dos conjuntos de datos distintos con una herramienta que nos proporciona el paquete Scipy de Python.

## 2 Actividad

En esta practica se realizara un ajuste lineal de temperaturas medias de invierno en Nueva York y un ajuste exponencial de la presión atmosférica con respecto a la altura, utilizando los paquetes de *Scipy: curve\_fit; leastsq*.

### 2.1 Programa 1: AjLineal.py

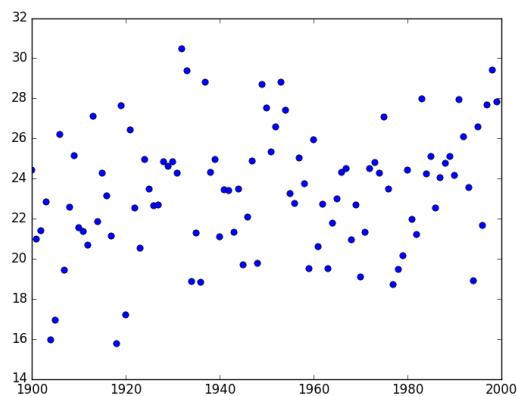
En este programa se lee el documento con los datos de temperaturas medias de invierno en Nueva York, y se realiza un ajuste lineal.

### problema 1

```
import matplotlib.pyplot as plt
import numpy as np

datos = np.loadtxt("lineal.txt")
x=datos[:,0]
y=datos[:,1]
print (x)
print (y)

plt.plot(x,y, 'bo')
plt.show()
```



## 2.2 Programa 2

En este programa se lee el documento con los datos de la presión atmosférica con respecto a la altura, y se realiza un ajuste exponencial.

### Código

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import optimize

#Leyendo el archivo con los datos
datos = np.loadtxt('lineal.txt')
x=datos[:,0]
```

```

y=datos[:,1]

print (x)
print (y)

#Escribiendo con la forma de  $y=mx+b$ 
fitfunc = lambda p, x: p[0]*x + p[1]

#Distancia del punto al ajuste lineal
errfunc = lambda p, x, y: fitfunc(p, x) - y

#Parametros patito
p0 = [1, 1]

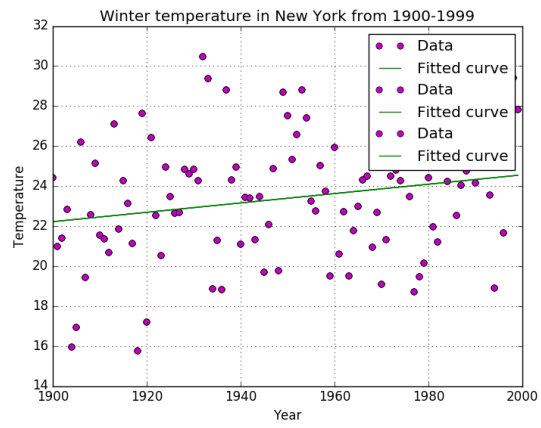
#Optimizando la recta
p1, success = optimize.leastsq(errfunc, p0[:], args=(x, y))

#Para la grafica
time = np.linspace(x.min(), x.max(), 100)
plt.plot(x, y, "mo", label="Data")
plt.plot(time, fitfunc(p1, time), "g-", label="Fitted curve")

plt.title("Winter temperature in New York from 1900-1999")
plt.grid()
plt.legend()
plt.xlabel("Year")
plt.ylabel("Temperature")

```

Lo que resulta de este código es lo siguiente:



## References

- [1] Wikipedia, *Mínimos cuadrados*. Recuperado de: [https://es.wikipedia.org/wiki/M%C3%ADnimos\\_cuadrados](https://es.wikipedia.org/wiki/M%C3%ADnimos_cuadrados)
- [2] Lizárraga, C. (2016) *Actividad 4 (2016-1)*. Recuperado de: [http://computacional1.pbworks.com/w/page/105016164/Actividad%204%20\(2016-1\)](http://computacional1.pbworks.com/w/page/105016164/Actividad%204%20(2016-1))
- [3] Qiku programación, (2013) *Cómo utilizar leastsq función desde scipy.optimize en python*. Recuperado de: <http://goo.gl/giZ7rO>