CMA IADCOM - UBA CENTRO DE INVESTIGACIÓN EN METODOS CUANTITATIVOS APLICADOS A LA ECONOMÍA Y LA GESTIÓN

Universidad de Buenos Aires Facultad de Ciencias Económicas

http://www.economicas.uba.ar/institutos_y_centros/revista-modelos-matematicos/

# EL USO DE PYTHON COMO ESTRATEGIA DIDÁCTICA EN UNA CLASE DE REGRESIÓN LINEAL SIMPLE

*Bianco, María José; Gache, Andrea y Lepera, Andrea*

*Universidad de Buenos Aires. Facultad de Ciencias Económicas. Ciudad Autónoma de Buenos Aires, República Argentina*

*mariajose.bianco@economicas.uba.ar andrealepera@economicas.uba.ar andreagache@gmail.com*

## Resumen

Recibido: 20-10-2022

Aceptado: 04-12-2022

**Palabras clave**

Regresión Lineal Simple.

Google Colaboratory.

Python.

Las nuevas tecnologías de la información producen un gran impacto y transformación en la sociedad, en la cultura y en la educación. Abren nuevas concepciones para la enseñanza y favorecen las actividades áulicas.

En particular, los alumnos futuros profesionales de las ciencias económicas necesitan comprender los conceptos estadísticos y sus aplicaciones, apreciar el poder de las herramientas estadísticas y, especialmente, reconocer situaciones en las cuales puede hacer uso efectivo de las mismas.

Estos avances conducen a un cambio de paradigma en las clases universitarias, desplazando el cálculo hacia la atenta selección de métodos que posibiliten la interpretación crítica de los resultados.

Debido a esto, se propone incorporar el lenguaje Python en el dictado de las clases como una herramienta complementaria de las actividades tradicionales. Su elección se debe a la facilidad de aprendizaje aún entre aquellos alumnos que no tienen conocimientos previos de programación. Asimismo, su versatilidad, su extensa comunidad de desarrollo, el hecho de ser código abierto y uno de los lenguajes más utilizados en el campo profesional, hacen de Python una estrategia didáctica adecuada.

En el presente trabajo se muestra la implementación en Python de un ejemplo empleado en el aula para introducir el tema Regresión Lineal Simple en la materia Estadística. Esta asignatura corresponde al segundo año de las carreras de la Facultad de Ciencias Económicas de la Universidad de Buenos Aires. Los alumnos no suelen tener conocimientos previos de lenguajes de programación, por lo cual se introduce Python mediante el entorno virtual Google Colaboratory a partir de un ejemplo sencillo desarrollado previamente en el aula con la intención de extender su uso a las distintas unidades del programa.



ISSN (En línea) 2362 3225

# THE USE OF PYTHON AS A DIDACTIC STRATEGY IN A SIMPLE LINEAR REGRESSION CLASS

## Abstract

**Keywords**

Linear Regression.

Google Colaboraty.

Python.

New information technologies produce a great impact and transformation in society, culture, and education. They develop new conceptions for teaching and encourage cultural activities.

In particular, future students in economics need to understand statistical concepts and their applications, appreciate the power of statistical tools, and especially recognize situations in which they can make effective use of them.

These advances lead to a paradigm shift in university classes, shifting calculation towards the careful selection of methods that enable the critical interpretation of the results.

Because of this, it is proposed to incorporate the Python language in the dictation of classes as a complementary tool to traditional activities. Its choice is due to the ease of learning even among those students who have no previous programming knowledge. Likewise, its versatility, its extensive development community, the fact that it is open source and one of the most used languages in the business professional field, make Python an adequate didactic strategy.

This paper shows the implementation in Python de an example used in the classroom to introduce the topic Simple Linear Regression in the subject Statistics. This subject corresponds to the second year of the careers of the Faculty of Economic Sciences of the University of Buenos Aires. Students usually do not have previous knowledge of programming languages, so Python is introduced through the virtual environment Google Colaboratory from a simple example previously developed in the classroom with the intention of extending its use to the different units of the program.



ISSN (En línea) 2362 3225

## INTRODUCCIÓN

El objetivo de este trabajo es introducir el lenguaje de programación Python en uno de los temas de la asignatura Estadística correspondiente al Segundo Tramo de todas las carreras de la Facultad de Ciencias Económicas de la Universidad de Buenos Aires.

Una de las últimas unidades de la asignatura corresponde al tema de Regresión Lineal Simple. En el desarrollo de la misma se inicia con la exposición de los conceptos teóricos del modelo, acompañados de un ejemplo sencillo con muy pocas observaciones a fin de facilitar los cálculos en el pizarrón. Finalmente se busca introducir el uso de un lenguaje de programación para obtener de manera más simple los mismos resultados del ajuste del ejemplo que fue desarrollado anteriormente en el aula. A su vez, una vez familiarizados con esta herramienta, puede abordarse el tratamiento de distintos ejemplos con mayor cantidad de observaciones, que permitan al alumno analizar violaciones de supuestos (de normalidad o de homocedasticidad), presencia de datos atípicos, diferencia de bondad de ajuste, etc.

En la primera parte de este trabajo se explicarán los conceptos básicos del modelo de Regresión Lineal Simple y se describirá luego la implementación en Python del ejemplo mencionado utilizando el entorno de trabajo de Google Colaboratory. No se incluirán ni el desarrollo de los cálculos del ejemplo que suelen realizarse en el pizarrón ni el análisis en Python de aquellos ejemplos con violaciones de supuestos o con observaciones atípicas, pues la finalidad de este artículo es simplemente mostrar cómo acercar a los alumnos a este lenguaje de programación, en el contexto de una clase tradicional.

La elección de la plataforma de Google Colaboratory se fundamenta en su disponibilidad libre y gratuita para cualquier persona que cuente con una cuenta de Google, facilita el acceso a nuevos usuarios sin necesidad de instalación de ningún *software*. Se parte del supuesto de que los alumnos, en su mayoría, no cuentan con conocimientos previos de programación en Python o en otro lenguaje.

Con esta actividad en la clase, se busca que el estudiante valore el ahorro de tiempo en el análisis de datos y en la resolución de ejercicios y a la vez se propicia la interpretación de resultados, por encima de los cálculos.

Se intenta con esto actualizar la forma de dictado de la materia, procurando que las clases fomenten la importancia de la Estadística en los futuros profesionales del área de las Ciencias Económicas, herramienta esencial para la toma de decisiones.

## 1. DESCRIPCIÓN DE LA PROPUESTA

Hoy en día existen numerosas herramientas para el análisis de datos como Python, R, SPSS, entre otras. La falta de conocimientos previos de programación por pate del alumnado dificulta la introducción de un lenguaje como Python para el análisis estadístico. Sin embargo, dado que se trata de un lenguaje eficiente, ampliamente empleado en el campo de las ciencias económicas, que se encuentra en constante desarrollo y cuenta con una gran cantidad de bibliotecas integradas, su implementación como estrategia didáctica resulta sumamente interesante.

Al ser un lenguaje de código abierto al que se puede acceder gratuitamente, es accesible a aquellos alumnos que no tienen conocimientos previos del mismo. Se utiliza en este trabajo el entorno virtual Colaboratory de Google, libre y gratuito, que permite escribir y ejecutar código Python sin requerimientos de configuración ni instalación de *software* en la computadora del usuario, y con la posibilidad de compartir contenido de una forma sencilla facilitando el trabajo colaborativo de los estudiantes.

## 2. REGRESIÓN LINEAL SIMPLE

La regresión es una de las herramientas principales de la estadística inferencial cuyo objetivo es describir la relación entre una variable métrica, denominada variable dependiente o de respuesta, y otro conjunto de variables, métricas o no, denominadas variables independientes o predictoras (Uriel Jiménez, E., y Aldás Manzano J., 2005). En particular la regresión lineal presupone que dicha relación entre las $p$ variables predictoras $(X_1, \cdots, X_p)$ y la variable dependiente $(Y)$ se produce mediante una expresión lineal en los parámetros o coeficientes beta $(\beta_0, \cdots \beta_p)$ de la forma:

$$\boldsymbol{E\big(Y|X = (x_1, \cdots, x_p)\big) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p} \quad (1)$$

O equivalentemente,

$$\boldsymbol{Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon} \qquad \text{CON } \boldsymbol{E(\epsilon) = 0} \quad (2)$$

Cuando solamente se introduce una variable predictora, es decir $p = 1$, el modelo recibe el nombre de Regresión Lineal Simple.

$$\boldsymbol{E(Y|X = x) = \beta_0 + \beta_1 x} \quad (3)$$

O de manera equivalente:

$$\boldsymbol{Y = \beta_0 + \beta_1 X + \epsilon} \qquad \text{CON } \boldsymbol{E(\epsilon) = 0} \quad (4)$$

El término $\epsilon$ se denomina perturbación o término de error y refiere a todas aquellas variables que

afectan colectivamente a $Y$ pero fueron omitidas en el modelo[1] (Gujarati, D. , 2022)

**2.1. Estimación de parámetros por Cuadrados Mínimos**

Para la estimación de los parámetros o coeficientes beta, se considera una muestra aleatoria de tamaño $n$ o datos de entrenamiento[2] de la forma $(x_1, y_1), \cdots, (x_n, y_n)$ con lo que, de acuerdo con la Ecuación (4) para cada observación se tendrá:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \qquad (i = 1, \cdots, n) \quad (5)$$

Si se denotan por $\hat{\beta}_0$ y $\hat{\beta}_1$ los estimadores de los parámetros $\beta_0$ y $\beta_1$ del modelo, sería esperable que las predicciones $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ estuvieran "cerca" en algún sentido de las observaciones $y_i$. Se buscará, entonces, que todos los residuos $\epsilon_i = y_i - \hat{y}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$ estén lo más cerca posible del 0.

Con esto, la estimación $\hat{\beta}_0$ y $\hat{\beta}_1$ por mínimos cuadrados de los parámetros $\beta_0$ y $\beta_1$ del modelo de regresión lineal surge de minimizar la diferencia cuadrática de los residuos, es decir:

$$\min_{\hat{\beta}_0, \hat{\beta}_1} \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right)^2 \qquad (6)$$

Puede demostrarse que los estimadores de mínimos cuadrados resultan ser:

$$\hat{\beta}_1 = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(xi - \bar{x})^2} \qquad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \qquad (7)$$

Donde $\bar{y}$ y $\bar{x}$ denotan, respectivamente, la media muestral de las variables $Y$ y $X$.

Una vez estimados los parámetros del modelo, se puede realizar predicciones de la variable $Y$ basándose en la denominada Recta de Regresión Muestral $\hat{y}(x) = \hat{\beta}_0 + \hat{\beta}_1 x$, para un valor de $X = x_0$

$$\hat{y}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0 \qquad (8)$$

Las hipótesis del modelo son conocidas bajo el nombre de Supuestos de Gauss Markov y se presentan a continuación.

- $\epsilon_i$ es independiente de $x_i$, $(\forall i \in \{1, \cdots, n\})$
- $\epsilon_i$ es independiente de $\epsilon_j$ $(\forall i \neq j)$
- Homocedastacidad de los residuos, es decir $Var(\epsilon_i) = \sigma^2$ constante $\forall i$

[1] La omisión puede deberse a vaguedad de la teoría, no disponibilidad de la información, variables proxy inadecuadas, entre otras. (Gujarati, 2003)

[2] El nombre de "datos de entrenamiento" surge de la terminología usada en los modelos de Aprendizaje Automático. En el caso de Aprendizaje Supervisado se intenta predecir el valor de alguna variable a partir de una muestra. Para ello se busca que los algoritmos "aprendan" de un conjunto de datos muestrales denominados "datos de entrenamiento" pero teniendo una buena *performance* frente a datos que no fueron usados previamente, denominados "datos de prueba".

- Normalidad de los residuos $\epsilon_i \sim N(0, \sigma)$

**2.2. Coeficiente de correlación lineal muestral de Pearson.**

Dadas dos variables aleatorias $X, Y$ con valores esperados $\mu_X, \mu_Y$ y varianzas $\sigma_X^2, \sigma_Y^2$ respectivamente, se definen la covarianza $cov(X, Y)$ y el Coeficiente de Correlación Lineal $\rho_{XY}$ como:

$$\boldsymbol{cov(X,Y) = E[(X - \mu_X)(Y - \mu_Y)]} \quad (9)$$

$$\boldsymbol{\rho = \rho_{XY} = \frac{cov(X,Y)}{\sigma_X \sigma_Y}} \quad (10)$$

Entre las propiedades del coeficiente de correlación lineal se tienen las siguientes:

- $-1 \leq \rho \leq 1$
- $\rho_{XY} = \rho_{YX}$
- Es invariante frente al cambio de escala de las variables, por lo cual no se ve afectado por cambio de unidades de medida de las variables $X, Y$
- Si las variables $X$ e $Y$ son estocásticamente independientes, la covarianza y el coeficiente de correlación lineal son nulos. La recíproca no es cierta. Cuando $\rho = 0$ decimos que las variables están incorrelacionadas o no correlacionadas, es decir que no tienen relación lineal (pero esto no implica que no tengan relación alguna).
- Cuando una de las variables es una función lineal de la otra, por ejemplo si $Y = a + bX$, el coeficiente de correlación lineal toma sus valores extremos -1 o 1. En particular, si la pendiente $b > 0, \rho = 1$ y si $b < 0, \rho = -1$.

A partir de una muestra aleatoria de tamaño $n$, los respectivos estimadores de la covarianza y el coeficiente de correlación lineal son los denominados covarianza muestral y coeficiente de correlación lineal muestral o de Pearson, que se definen como:

$$\boldsymbol{\widehat{cov}(X,Y) = \frac{\sum(x_i - \bar{x})(y_i - \overline{y})}{n-1}} \quad (11)$$

$$\boldsymbol{\hat{\rho} = \hat{\rho}_{XY} = \frac{\widehat{cov}(X,Y)}{S_X \cdot S_Y}} \quad (12)$$

donde $S_X, S_y$ denotan, respectivamente, el desvío muestral de $X$ e $Y$.

Puede interpretarse la asociación lineal entre los valores de $X$ e $Y$ observados a partir del coeficiente de correlación muestral de Pearson de manera que:

- si $\hat{\rho}$ toma valor absoluto cercano a 1, la asociación es fuerte.
- si $\hat{\rho}$ toma valor absoluto cercano a 0, la asociación es débil.
- si $\hat{\rho} > 0$ la asociación es directa.
- si $\hat{\rho} < 0$ la asociación es inversa.

### 2.3. Inferencia en el modelo de regresión

El cuadrado medio residual CMres es un estimador de la varianza $\sigma^2$ del término de error $\epsilon$. Se define en función de la suma de los cuadrados de los residuos:

$$\boldsymbol{CM_{Res}} = \frac{\sum(y_i - \hat{y}_i)^2}{n-2} \qquad (13)$$

De la Ecuación 4 se deriva que el valor de la variable $Y$ puede expresarse en función de la variable $X$ como $Y = \beta_0 + \beta_1 \cdot X + \epsilon$. Pero si el parámetro $\beta_1$ tomara valor nulo, la variable regresora $X$ no tendría efecto en la variable $Y$. Por lo cual es usual realizar un test de significación de este parámetro cuyas hipótesis son $H_0: \beta_1 = 0$ vs. $H_1: \beta_1 \neq 0$. El rechazo de la hipótesis nula indicaría que la regresión es significativa.

El estadístico de prueba a emplearse en este caso es $\mathcal{E} = \frac{\beta_1}{\sqrt{\frac{CM_{Res}}{\sum(xi-\bar{x})^2}}}$ que, bajo el supuesto de que $H_0$ es verdadera, tiene distribución T de Student con n-2 grados de libertad ($\mathcal{T}_{n-2}$). Cabe señalar, también, que es usual utilizar el P-valor a fin de realizar un test de hipótesis[3].

Asimismo, luego de la estimación de los parámetros, son de interés la construcción de intervalos de confianza para la media poblacional $E(Y|X = x_k)$ e intervalos de predicción $Y_k$ para la variable $Y$ cuando $X = x_k$.

Si se denota por $\hat{y}_k = \hat{\beta}_0 + \hat{\beta}_1 \cdot x_k$ a la predicción para $X = x_k$, la expresión de estos intervalos con nivel de confianza $1 - \alpha$, se muestra a continuación:

Intervalo de Confianza para $E(Y|X = x_k)$ :

$$\left(\boldsymbol{\hat{y}_k} \pm \boldsymbol{t_{n-2,\alpha/2}} \cdot \sqrt{\boldsymbol{CM_{Res}} \cdot \left(\frac{1}{n} + \frac{(x_k - \bar{x})^2}{\sum(x_i - \bar{x})^2}\right)}\right) \qquad (14)$$

Intervalo de Predicción para $Y_k$:

$$\left(\boldsymbol{\hat{y}_k} \pm \boldsymbol{t_{n-2,\alpha/2}} \cdot \sqrt{\boldsymbol{CM_{Res}} \cdot \left(\mathbf{1} + \frac{1}{n} + \frac{(x_k - \bar{x})^2}{\sum(x_i - \bar{x})^2}\right)}\right) \qquad (15)$$

### 2. 4. Coeficiente de Determinación

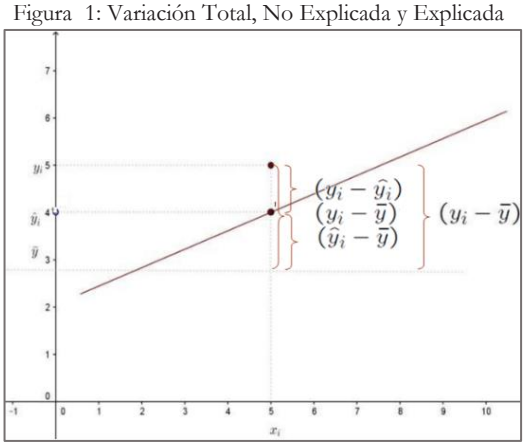
[3] El P-valor indica el máximo nivel de significación que nos lleva a un rechazo en un test de Hipótesis. En el caso de los tests de significatividad de los coeficientes beta de la regresión, la hipótesis nula es que dicho coeficiente es nulo.

Se denomina coeficiente de Determinación y se denota por $R^2$ al coeficiente que indica la proporción de variación de la variable $Y$ que queda explicada a través del comportamiento de X.

Su definición surge de la descomposición de la Variación Total de $Y$ en Variación No Explicada y Variación Explicada por el modelo.

Como se puede observar en la Figura 1, para cada observación muestral, la diferencia entre $y_i$ y la media muestral $\bar{y}$ puede obtenerse como la suma del residuo $y_i - \hat{y}_i$ más la diferencia entre la predicción $\hat{y}_i$ e $\bar{y}$. Al considerarse la totalidad de las observaciones y si se elevan al cuadrado estas diferencias se obtienen, respectivamente, la Variación Total, la Variación No Explicada y la Variación Explicada por el modelo.

Figura 1: Variación Total, No Explicada y Explicada




Fuente: Elaboración propia

La variación total de $Y$ o "Suma de Cuadrados Total" ($SCT$) es el numerador de la varianza muestral de $Y$. Puede demostrarse que la misma puede descomponerse en la "Suma de los cuadrados del Error" ($SCE$) y la "Suma de cuadrados de la Regresión" ($SCR$), tal como se muestra a continuación:

$$\underbrace{\sum(y_i - \bar{y})^2}_{SCT} = \underbrace{\sum(y_i - \hat{y}_i)^2}_{SCE} + \underbrace{\sum(\hat{y}_i - \bar{y})^2}_{SCR} \qquad (16)$$

La Suma de Cuadrados del Error ($SCE$) guarda relación con los residuos y refiere a lo que no pudo ser explicado por el modelo.

De tal forma que el cociente $\frac{SCE}{SCT}$ brinda la proporción de Variación No Explicada por el modelo, mientras $\frac{SCR}{SCT} = 1 - \frac{SCE}{SCT}$ la proporción de Variación Explicada.

Así el coeficiente de determinación $R^2$ se define por:

$$R^2 = \frac{SCR}{SCT} \qquad (17)$$

Al tratarse de una proporción, se cumple que $0 \leq R^2 \leq 1$; este coeficiente tomará valores más cercanos a 1 cuanto mayor sea la proporción de variabilidad de $Y$ que pueda explicarse por el modelo. De tal forma que $R^2$ resulta ser un indicador de la Bondad del Ajuste.

Puede demostrarse que el Coeficiente de Determinación es igual al cuadrado del coeficiente de correlación, o sea:

$$R^2 = \hat{\rho}^2_{XY} \qquad (18)$$

Es importante destacar que los dos coeficientes utilizados en el análisis de regresión difieren en cuanto a su interpretación, tal como se muestra en la Figura 2, el coeficiente de Determinación $R^2$ representa el porcentaje de la variación de $Y$ que es explicado por la regresión, mientras que el coeficiente de correlación muestral de Pearson $\hat{\rho}_{XY}$ nos permite clasificar la relación lineal en "fuerte" o "débil" (según el valor absoluto $|\hat{\rho}_{XY}|$ se encuentre cercano a 1 o a 0) y el sentido de la relación (directa o inversa) entre las variables según el signo positivo o negativo de $\hat{\rho}_{XY}$ (Bacchini, 2018).

Figura 2: Coeficiente de Correlación Lineal de Pearson y Coeficiente de Determinación

| Coeficiente de Correlación Lineal de Pearson | Coeficiente de Determinación |
|---|---|
| $\hat{\rho}_{XY}$ | $R^2$ |
| $-1 \leq \hat{\rho}_{XY} \leq 1$ | $0 \leq R^2 \leq 1$ |
| Mide la asociación lineal entre X e Y | Mide la proporción de variación explicada por el modelo |
| $\lvert\hat{\rho}_{XY}\rvert = \sqrt{R^2}$ | $R^2 = \hat{\rho}^2_{XY}$ |

Fuente: Elaboración propia

## 3. EJEMPLO DE APLICACIÓN CON PYTHON

El ejemplo utilizado en el aula para el cálculo de la recta de regresión, las predicciones, coeficientes de correlación de Pearson y de Determinación e inferencia en el modelo intenta explicar la relación entre la ganancia (en miles de dólares) de ciertas heladerías (variable "Ganancia") y su inversión en publicidad (en miles de pesos) (variable "Inversión")

- ✓ Variable Independiente ($X$): Inversión en Publicidad (miles de pesos)
- ✓ Variable Dependiente ($Y$): Ganancia (miles de dólares)

Los datos observados o muestra aleatoria se vuelcan en la tabla a continuación:

| Inversión ($1000) | Ganancia (U$S1000) |
|---|---|
| 1 | 5 |
| 2 | 7 |
| 3 | 9 |
| 4 | 10 |
| 5 | 14 |

A partir de aquí, se irá mostrando la sintaxis y salidas de Python utilizando el entorno de Google Colaboraty, que permiten realizar el ajuste al modelo de los datos del ejemplo que ya fuera trabajado en el aula.

### 3.1 Importación de librerías

Para poder iniciar el análisis es necesario importar las siguientes bibliotecas, cada una con su correspondiente alias mediante el comando "import librería as alias".

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.formula.api as smf
```

Cada una de ellas tiene su utilidad y la asignación de un alias facilita la sintaxis.

- Numpy (alias np): permite realizar cálculos numéricos.
- Pandas (alias pd): especializada en manejo de bases de datos
- El módulo pyplot de matplotlib permite elaborar la mayoría de los gráficos
- Statsmodel[4] permite trabajar con modelos estadísticos y la api formula permite una sintaxis similar a la usada en el lenguaje R.

### 3.2 Diagrama de Dispersión

Es usual que antes de ajustar un modelo de regresión lineal simple, se visualicen los datos mediante un diagrama de dispersión o *scatterplot.*

En primer lugar, se ingresan los datos correspondientes a las observaciones realizadas en formato de *Data Frame*, mediante la biblioteca Pandas.

[4] Statsmodels es un paquete de Python que permite a los usuarios explorar datos, estimar modelos estadísticos y realizar pruebas estadísticas. Una extensa lista de estadísticas descriptivas, pruebas estadísticas, funciones de trazado y estadísticas de resultados están disponibles para diferentes tipos de datos y cada estimador. Complementa el módulo de estadísticas de SciPy. Statsmodels es parte de la pila científica de Python que está orientada al análisis de datos, la ciencia de datos y la estadística. Statsmodels se basa en las bibliotecas numéricas NumPy y SciPy, se integra con Pandas para el manejo de datos y utiliza Patsy para una interfaz de fórmula similar a R. Las funciones gráficas se basan en la biblioteca Matplotlib.

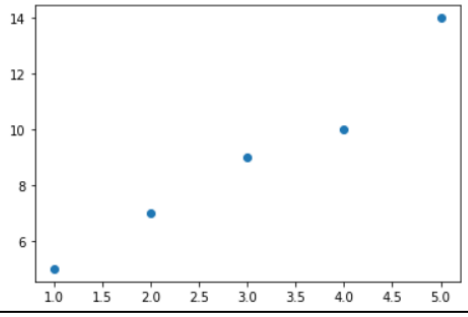
Línea de Código Python para cargar los datos

```
datos=pd.DataFrame({'x':[1,2,3,4,5], 'y':[5,7,9,10,14]})
```

La sintaxis para elaborar el diagrama de dispersión entre la variable X (*inversión*) y la variable Y (*ganancia*) y la correspondiente salida de Python es la siguiente:

Línea de Código Python y salida del Scatterplot.

```
[ ]  plt.scatter(datos['x'],datos['y'])
     plt.show()
```




### 3.3 Estimación de los parámetros

Para realizar el ajuste del modelo $Y = \beta_0 + \beta_1 \cdot X + \epsilon$ se utiliza la biblioteca statsmodels.formula.api.

La misma permite realizar el ajuste directamente mediante el método[5] "fit"(ajuste) de "ols" (mínimos cuadrados ordinarios); dicho ajuste será guardado una variable que denominamos "lm" para luego poder acceder fácilmente a múltiple información del modelo.

Línea de Código Python para realizar el ajuste por mínimos cuadrados

```
lm=smf.ols('y~x', data=datos).fit()
```

Por ejemplo para acceder a los parámetros estimados $\hat{\beta}_0$ y $\hat{\beta}_1$ y con ellos construir la recta de regresión muestral, se emplea el comando "lm.params"

Línea de Código Python para obtener la estimación de parámetros del modelo

```
[ ]  lm.params

     Intercept    2.7
     x            2.1
     dtype: float64
```

De esta forma la recta de regresión muestral es:

$$\hat{y}(x) = 2.1 + 2.7 \cdot x \qquad (19)$$

Es importante resaltar la interpretación de los parámetros que han sido estimados:

[5] En Python, un método es una función específica de un objeto.

- El coeficiente $\hat{\beta}_0$ representa la ganancia esperada cuando la inversión publicitaria es nula.
- El coeficiente $\hat{\beta}_1$ representa el cambio esperado (aumento si $\hat{\beta}_1 > 0$ o disminución si $\hat{\beta}_1 < 0$)[6] de la variable $Y$ (en sus unidades) cuando la variable $X$ aumenta una unidad.

En particular en este ejemplo de aplicación, considerando que las unidades de las variables $Y$ (Ganancia) y $X$ (Inversión en Publicidad) son, respectivamente, miles de dólares y miles de pesos, el valor del coeficiente $\hat{\beta}_1 = 2{,}1$ indica que por cada $1000 adicionales de Inversión en Publicidad de la heladería, se espera que su Ganancia aumente U$S 2100.

A partir del ajuste, pueden obtenerse tanto las predicciones $y_i$ para los valores $x_i$ de la muestra como los residuos $e_i = y_i - \hat{y}_i$.

Generamos ambos con las siguientes líneas de código:

Línea de Código Python para realizar predicciones y residuos

```
Y_pred=2.7+2.1* datos['x']

residuos=datos['y']-Y_pred
```

Armamos un *Data Frame* a fin de visualizar en un formato de tabla los valores de $X, Y, Y_{pred}, e$, cuyo código y salida se muestran a continuación:

Línea de Código Python para armar una Tabla con las observaciones, predicciones y residuos

```
Tabla_resultados=pd.DataFrame({

    "Invers": datos['x'],

    'Gcia':datos['y'],

    'Pred': Y_pred,

    'e': residuos})
```

[6] Si $\hat{\beta}_1 = 0$ la variable $X$ no tendría efecto sobre la variable $Y$.

Salida del código Python de la Figura anterior

```
[ ] Tabla_resultados
```

| | Invers | Gcia | Pred | e |
|---|---|---|---|---|
| **0** | 1 | 5 | 4.8 | 0.2 |
| **1** | 2 | 7 | 6.9 | 0.1 |
| **2** | 3 | 9 | 9.0 | 0.0 |
| **3** | 4 | 10 | 11.1 | -1.1 |
| **4** | 5 | 14 | 13.2 | 0.8 |

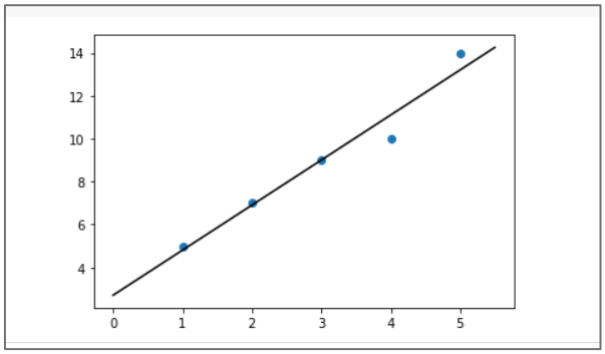
### 3.4 Representación Gráfica de la Recta de Regresión

Si se desea graficar la recta de regresión muestral en el diagrama de dispersión, puede emplearse código que se expone a continuación en la, que pasamos a explicar:

En las dos primeras líneas se consideran valores equiespaciados en el eje de las abscisas y sus correspondientes ordenadas en la recta de regresión. Luego se solicita el diagrama de dispersión y a continuación el gráfico (plot) de la recta. Finalmente con el método "show( )" de la biblioteca con alias "plt", se solicita la salida del gráfico.

Línea de código Python para diagrama de dispersión y Recta de Regresión Muestral y su salida correspondiente

```
[ ] x_linea=np.linspace(start=0, stop=max(datos['x'])+0.5, num=1000)
    y_linea=lm.params[0]+lm.params[1]*x_linea
    plt.scatter(datos['x'],datos['y'])
    plt.plot(x_linea, y_linea, 'black')
    plt.show()
```




### 3.5 Coeficiente de Correlación Lineal Muestral de Pearson

Recordando la definición, el Coeficiente de Correlación Lineal Muestral de Pearson es el cociente entre la covarianza y los desvíos de ambas variables. La matriz $S$ de varianzas y covarianzas de un conjunto de observaciones de $X$ e $Y$ se obtiene mediante el método "cov" de la biblioteca Numpy (con alias np). Como la covarianza se encuentra en la fila 1 columna 2 de S, debemos acceder a

dicha posición a través de dichos índices[7]. Vemos que el coficiente $\hat{\rho}_{XY}$ guardado bajo el nombre $R$ en el código, resulta ser un valor positivo cercano a 1, lo que muestra que las variables "Inversion" y "Ganancias" tienen una fuerte correlación lineal directa.

Línea de Código Python para calcular el Coeficiente de Correlación Muestral de Pearson y su correspondiente salida

```
[ ]  R=np.cov(datos['x'],datos['y'])[0][1]/(datos['x'].std()*datos['y'].std())
     R.round(4)

     0.9791
```

### 3.7 Proporción de variación explicada (coeficiente de determinación) $R^2$

Dado que el coeficiente de Determinación es el cuadrado del coeficiente de Correlación calculado recién y guardado bajo el nombre $R$, solamente hace falta realizar la potencia 2 del mismo y obtenemos una proporción 0.9587 o un porcentaje del 95,87% de variación explicada por el modelo, lo que indicaría un buen ajuste.

Línea de Código Python para calcular el Coeficiente de Determinación a partir del de Correlación de Pearson y su correspondiente salida

```
[ ]  (R**2).round(4)

     0.9587
```

### 3.8 Resumen del Ajuste mediante la biblioteca específica

Se muestra en este apartado el análisis del ajuste obtenido mediante la biblioteca "statsmodels.formula.api". Al guardar el ajuste en la variable "lm", como hicimos, podemos acceder a distintos aspectos del ajuste mediante los correspondientes métodos de este objeto de Python.

Algunos de estos métodos o funciones son:

- *Summary( )*: nos brinda un resumen completo del ajuste, incluyendo la estimación de los parámetros, el P-valor que indica si los mismos son significativos o no, el coeficiente de Determinación $R^2$, etc.

[7] En Python la posición de una lista o iterable se inicializa en 0, por lo cual si se desea acceder a la 1ra fila 2da columna de una matriz S, se debe indicar con los índices 0 y 1 respectivamente.

Línea de Código Python de un resumen o *Summary* del ajuste y su correspondiente salida

```
lm.summary()
```

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | y | **R-squared:** | 0.959 |
| **Model:** | OLS | **Adj. R-squared:** | 0.945 |
| **Method:** | Least Squares | **F-statistic:** | 69.63 |
| **Date:** | Mon, 05 Dec 2022 | **Prob (F-statistic):** | 0.00361 |
| **Time:** | 14:28:23 | **Log-Likelihood:** | -4.6757 |
| **No. Observations:** | 5 | **AIC:** | 13.35 |
| **Df Residuals:** | 3 | **BIC:** | 12.57 |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | 2.7000 | 0.835 | 3.235 | 0.048 | 0.044 | 5.356 |
| **x** | 2.1000 | 0.252 | 8.345 | 0.004 | 1.299 | 2.901 |

| | | | |
|---|---|---|---|
| **Omnibus:** | nan | **Durbin-Watson:** | 2.547 |
| **Prob(Omnibus):** | nan | **Jarque-Bera (JB):** | 0.432 |
| **Skew:** | -0.692 | **Prob(JB):** | 0.806 |
| **Kurtosis:** | 2.598 | **Cond. No.** | 8.37 |

En esta salida podemos observar lo siguiente:[8]

1. Se usaron 5 observaciones.
2. Las estimaciones de los parámetros son: 2.7 el *intercept* $\hat{\beta}_0$ y 2.1 el coeficiente $\hat{\beta}_1$ de la variable denominada X. Como los P-valores son respectivamente 0.048 y 0.004 , ambos parámetros $\beta_0$ y $\beta_1$ a nivel poblacional pueden considerarse no nulos.
3. El coeficiente de determinación $R^2$ toma el valor 0.959.

- *params*: para obtener los parámetros estimados

Línea de Código Python de los parámetros estimados y su correspondiente salida

```
[ ] lm.params

    Intercept    2.7
    x            2.1
    dtype: float64
```

- *pvalues*: para obtener los P-valores que permiten testear si los parámetros son significativamente no nulos.

Línea de Código Python de los P-valores de los coeficientes beta y su correspondiente salida

```
[ ] lm.pvalues

    Intercept    0.048039
    x            0.003608
    dtype: float64
```

- *rsquared*: arroja el valor del coeficiente de Determinación, el cual redondeamos a 4 decimales.

Línea de Código Python de del coeficiente $\boldsymbol{R^2}$ y su correspondiente salida.

[8] Solamente se mencionan los aspectos que interesan en este trabajo. Pero también se incluye en el *summary* el R Cuadrado Ajustado (para modelos de regresión múltiples), el test F, el índice de Akaike, los coeficientes de asimetría y kurtosis, tests de normalidad, etc.

```
[ ]  lm.rsquared.round(4)

     0.9587
```

- *predict*: permite hacer predicciones para un *Data Frame* con valores de variables regresoras $x$

Línea de Código Python de las predicciones para el conjunto de datos observados 'x' y su correspondiente salida

```
[ ]  Y_pred=lm.predict(pd.DataFrame(datos['x']))
```

```
   Y_pred
0     4.8
1     6.9
2     9.0
3    11.1
4    13.2
```

## CONCLUSIONES

Dada la importancia y necesidad de implementar las nuevas tecnologías en el dictado de las clases universitarias se propicia el empleo en la materia Estadística del Segundo Tramo de la Facultad de Ciencias Económicas de la Universidad de Buenos Aires el uso de Python por ser un *software* libre y abierto, de fácil manejo y con una rápida curva de aprendizaje.

A su vez, al permitir al alumno quitar el foco de los cálculos para centrar la atención en los conceptos, promueve el espíritu crítico respecto a los datos estadísticos. Brinda también un entorno dinámico que favorece la motivación y el desarrollo de una nueva experiencia formativa durante la clase. Se genera de esta manera un ambiente activo, favorable a la reflexión y a la motivación de los estudiantes durante la clase acerca de los conceptos estadísticos fundamentales (Carlson, K.A, y Winquist, J.R., 2011).

Asimismo, beneficia la mejora continua del proceso de enseñanza-aprendizaje tanto por parte de los alumnos de grado en los cursos de Estadística.

Sin embargo, es de suma importancia comprender que el uso de estos recursos no reemplaza el desarrollo teórico y formal de los temas, sino que lo complementa. Dada la intencionalidad meramente pedagógica, esta herramienta se convierte en un facilitador de cálculos con complejas operaciones matemáticas.

Se nos plantea, así, el reto de implementar esta estrategia didáctica en otras unidades temáticas de la asignatura como, por ejemplo, Estadística Descriptiva o Probabilidad.

**REFERENCIAS**

Bacchini, D.; Vázquez, L,; Bianco, M J; Casparri, M T. (2018). *Introducción a la Probabilidad y a la Estadística*. Recuperado: http://bibliotecadigital.econ.uba.ar/download/libros/Bacchini_Introduccion-a-la-probabilidad-y-a-la-estadistica-2018.pdf

Carlson, K. A., & Winquist, J. R. (2011). *Evaluating an active learning approach to teaching introductory statistics: A classroom workbook approach*. Journal of Statistics Education, 19(1).

Uriel Jiménez, E., y Aldás Manzano J. (2005). *Análisis Multivariado Aplicado*. Thomson, Madrid.— *WRIGHT*, 377-399.

Gujarati, D. M. (2022). *Gujarati: Basic Econometrics*. McGraw-hill.