



UNIVERSIDAD SANTO TOMÁS

PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA

T U N J A

VIGILADA MINEDUCACIÓN - SNIES 1732

ACREDITACIÓN
INSTITUCIONAL
DE ALTA CALIDAD
MULTICAMPUS

Res. MEN No. 01456 del 29 de enero de 2016

Vigencia por seis años



Formando personas que transforman

Bicentenario de la Independencia Nacional





UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Faculty: systems engineer

Course: Deep Learning

Topic: Introducción a la Visión Artificial

Professor: Luis Fernando Castellanos Guarín

Email: Luis.castellanosg@usantoto.edu.co

Phone: 3214582098

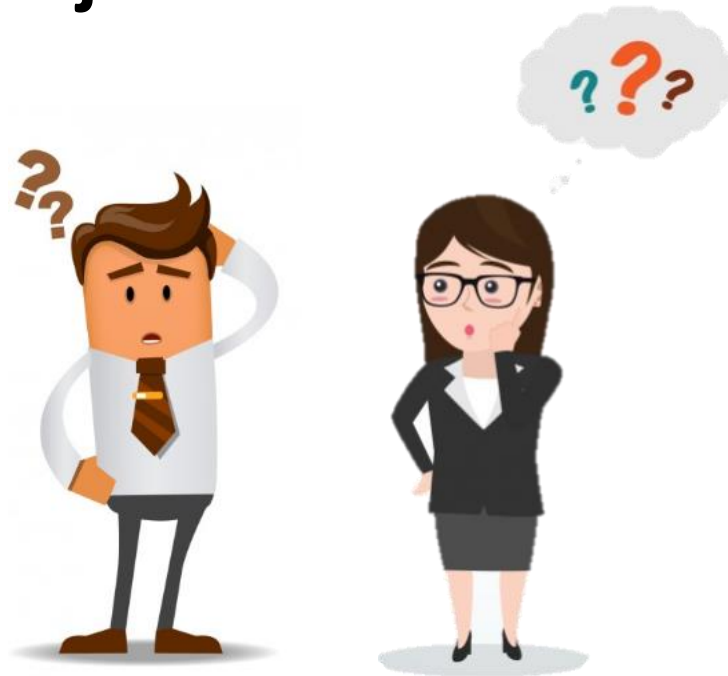
Formando personas que transforman

Bicentenario de la Independencia Nacional

Deep learning - requerimientos

Requerimientos de Software

¿Cual es el mejor lenguaje de programación para trabajar “Machine learning”?



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Deep learning - requerimientos

Requerimientos de Software

¿Cual es el mejor lenguaje de programación para trabajar “Machine learning”?

- R
- LISP
- PROLOG
- JAVA
- C++
- PYTHON
- MATLAB



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Deep learning - requerimientos

Requerimientos de Software

¿Cual es el mejor lenguaje de programación para trabajar “Machine learning”?

- **R:** Está diseñado para análisis estadísticos y visualizaciones, se usa con frecuencia para desbloquear los patrones en grandes bloques de datos
- **LISP:** trabaja con expresiones simbólicas y prototipado, herramientas útiles en el campo del Machine Learning. Además, se utiliza en proyectos cuyo objetivo es permitir a las aplicaciones basadas en IA ejecutar razonamientos similares a los humanos.
- **PROLOG:** se basa en estructuras de datos arbóreas que facilitan la búsqueda de patrones. Todas estas características combinadas convierten este lenguaje en uno de los más flexible



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Deep learning - requerimientos

Requerimientos de Software

¿Cual es el mejor lenguaje de programación para trabajar “Machine learning”?

- **JAVA:** Además, dispone de interfaces de datos muy atractivas para mejorar la experiencia del usuario. Si tenemos alguna duda, dispone de una nutrida comunidad de usuarios que pueden ayudarnos
- **C++:** Es el lenguaje de programación más rápido del mundo y en IA eso es vital.
- **PYTHON:** es un éxito entre los principiantes que son nuevos en Machine Learning. Python viene con librerías específicas como SciPy que contiene NumPy y Pandas, que permiten a la computadora aprender álgebra lineal y métodos kernel (respaldo por Google)
- **MATLAB:** Es un algoritmo rápido, estable y seguro que asegura la matemática compleja. Considerado como un lenguaje de núcleo duro para matemáticos y científicos que se ocupan de sistemas complejos, encuentra un camino en muchas aplicaciones



Deep learning - requerimientos

Requerimientos de Software

¿Cual es el mejor lenguaje de programación para trabajar “Machine learning”?



Instagram, Youtube, Google, Facebook, Netflix, vaya hasta la misma Nasa utiliza **Python**.

El potencial del lenguaje es sin duda increíble, con él, podemos desarrollar prácticamente lo que deseemos.

Python es utilizado en muchas áreas:

- videojuego,
- páginas web,
- la ciencia de datos,
- machine learning,
- Blockchain

En todas estas áreas, y muchas más, es posible utilizar Python



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Deep learning - requerimientos

Requerimientos de Software

Necesitaremos un computador con software como:

- Python 3
- scikit learn
- Tensorflow/pytorch/keras
- OpenCv



Deep learning - requerimientos

Requerimientos de hardware

Necesitaremos un computador con:

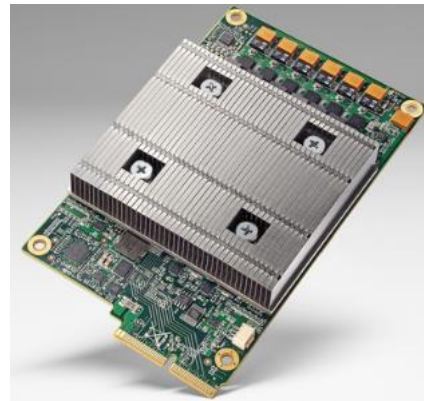
- Una buena CPU: mínimo I5 séptima generación



- GPU



- TPU (Tensor Processing Unit)



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



O podemos usar

Google colabory

Donde no tendremos que invertir dinero en hardware ni tiempo instalando software

<https://colab.research.google.com/notebooks/welcome.ipynb?hl=es>

Colaboratory es una herramienta de investigación para la educación y la investigación de aprendizaje automático. Es un entorno de notebook Jupyter que no requiere configuración para usar.



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Segundo ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

1. Ingresar a Google colabatory

1. Clic en Archivo-> Nuevo cuaderno de Python 3

1. Definimos el nombre al archivo así: PIT1_PYTHON_BASIC_XXXX

- Donde XXXX son las iniciales de su nombre

2. Diseñar un programa en PYTHON que debe tomar como argumento tres números y devolver el mayor de ellos.

1. Una vez realice el programa en la parte superior de la ventana deben darle “compartir” y agregar mi correo electrónico : Luis.castellanosg@usantoto.edu.co



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Segundo ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

P1T1: Realice un programa en Python que determine el mayor de tres números:

Método clásico

```
var1=int(input("ingresar el valor de la 1° variable:"))
var2=int(input("ingresar el valor de la 2° variable:"))
var3=int(input("ingresar el valor de la 3° variable:"))
if (var1>var2 and var1>var3):
    print("el mayor es :"+str(var1))
elif (var2>var1 and var2>var3):
    print("el mayor es :"+str(var2))
else:
    print("el mayor es :"+str(var3))
```

Método Python

```
var1, var2, var3= input ("Ingrese los tres números separados por coma:").split(",")
print("El número máximo es: "+ str(max(var1, var2, var3)))
```



Segundo ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

1. Diseñar un programa en PYTHON donde se ingresen N cantidad de números y me imprima la suma

```
#Método clásico
tot=int(input("Ingrese la cantidad N de números: "))
suma=0
for i in range(0, tot):
    suma+=int(input("ingrese el valor del num "+str(i)+": "))
print("El total es: "+str(suma))
```

```
#Método minimalista de Python
suma=0
for i in range(0, int(input("Ingrese la cantidad N de números: "))):
    suma+=int(input("ingrese el valor del num "+str(i)+": "))
print("El total es: "+str(suma))
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Segundo ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

Diseñe un programa (máximo 5 líneas) que me visualice únicamente los números pares existentes entre dos números.

(RECUERDE hacerlo en dos partes (como usted cree que se hace.....y como debe hacerse en Python..minimo de lineas)



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Segundo ejercicio en Python (P1T3_PYTHON_BASIC_XXXX)

```
#Método clásico
inicio=int(input("Número minimo: "))
fin=int(input("Número final: "))
for i in range(inicio,fin):
    if ((i%2)==0):
        print("El número par :" + str(i))
```

```
#Método Python minimalista
for i in range(int(input("Número minimo: ")),int(input("Número final: "))) :
    if( (i%2)==0 ): print("El número par :" + str(i))
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Manejo de Listas / vectores

En una lista puedo guardar “*lo que se me de la xyz voluntad*”, pero tiene sus desventajas...cuales son?

ejemplo:

- lista1=["a","b","c"]
- lista2=[1, 2, "papa","yuca"]
- lista3=[lista1, lista2]

Como imprimir o visualizar la información:

- for x in range(len(lista1)): print (lista1[x])
- print(*lista1)
- print(' '.join(lista1))
- print('\n'.join(map(str, lista1)))



Tercer ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

Diseñe un programa que me visualice dos listado concatenados:

- Uno donde los números sean de 1 al 12.
- Otro listado donde los números sean del 12 al 1

Ejemplo: $[1,2,3,4,5,6,7,8,9,10,11,12], + [12,11,10,9,8,7,6,5,4,3,2,1]$



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Tercer ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

Diseñe un programa que me visualice dos listado concatenados:

- Uno donde los números sean de 1 al 12.
- Otro listado donde los números sean del 12 al 1

Ejemplo: $[1,2,3,4,5,6,7,8,9,10,11], + [12,11,10,9,8,7,6,5,4,3,2,1]$

```
#Método clásico
rango1=list(range(1,13))
rango2=list(range(12,0,-1))
print(rango1+rango2)
```

```
#Método Python minimalista
lista=list(range(1,12))
print(lista+lista[::-1])
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Cuarto ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

P1T5: Diseñe un programa que me visualice un listado de 10 números aleatorios entre 1 y 100

- Ejemplo: *25,12,80,32,1,7,98,23,56,10*

Usando librerías:

```
import random
Var1= random.randrange(101)
print(Var1)
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Cuarto ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

Diseñe un programa que me visualice un listado de 10 números aleatorios entre 1 y 100:

- Ejemplo: *25,12,80,32,1,7,98,23,56,10*

```
#Método clásico
import random
lista= [ random.randrange(10) for i in range(10) ]
print(lista)
```

```
#Método Python minimalista
for i in range(10): print(random.randrange(100))
```



Cuarto ejercicio en Python (P1T6_PYTHON_BASIC_XXXX)

P1Tx: Diseñe un programa que me genere un listado con las edades de N habitantes de una ciudad, donde la cantidad de habitantes sea un valor randomico entre 1000 y 50000 y la edad de cada persona también debe ser un randomico entre 1 y 100 años. Al final del el programa me debe indicar cuanto es la población total y el porcentaje de cuantos son mayores de edad (≥ 18) y cuantos no.



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



El objeto array numpy

(computación científica)



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



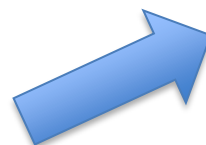
Numpy

Numpy es :

- Uno de los módulos más importantes de Python
- **Añade** toda la capacidad matemática y vectorial a Python haciendo posible operar con cualquier dato numérico o array (posteriormente veremos qué es un array).
- **Incorpora** operaciones tan básicas como la suma o la multiplicación u otras mucho más complejas como la transformada de Fourier o el álgebra lineal.
- **Incorpora** herramientas que nos permiten incorporar código fuente de otros lenguajes de programación como C/C++ o Fortran lo que incrementa notablemente su compatibilidad e implementación.

Uso:

```
import numpy as np
from random import randint
matriz = np.array([[i+1, randint(10000,70000)] for i in range(10)])
print(matriz)
```



```
[[ 1 51360]
 [ 2 12247]
 [ 3 11167]
 [ 4 50145]
 [ 5 60036]
 [ 6 26505]
 [ 7 40667]
 [ 8 19105]
 [ 9 62079]
 [10 45023]]
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1702



Formando personas que transforman



Cuarto ejercicio en Python (P1T7_PYTHON_BASIC_XXXX)

P1Tx: Diseñe un programa (máximo 5 líneas) donde 6 candidat@s se disputan la alcaldía (ayuntamiento municipal), los candidat@s de los siguientes partidos:

- 1 – socio-democratico
- 2 – Ultra-derechista
- 3 – primero los ricos
- 4 – Centro demoniaco
- 5 – Cambio invertido
- 6 – Alianza queremos más pobres

Los votos de cada candidato son generados de forma aleatoria entre 1 y 25000 votos máximo, el software debe determinar quién fue el ganador. (utilicé la librería numpy)



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Cuarto ejercicio en Python (P1Tx_PYTHON_BASIC_XXXX)

Diseñe un programa (máximo 5 líneas) donde 6 candidat@s se disputan la alcaldía (ayuntamiento municipal), los candidat@s de los siguientes partidos:

- 1 – socio-democratico
- 2 – Ultra-derechista
- 3 – primero los ricos
- 4 – Centro demoniaco
- 5 – Cambio invertido
- 6 – Alianza queremos más pobres

Los votos de cada candidato son generados de forma aleatoria entre 1 y 25000 votos máximo, el software debe determinar quién fue el ganador. (utilicé la librería numpy)

```
import numpy as np
from random import randint
matriz = np.array([[i+1,randint(1,20000)] for i in range(6)])
ganador = np.where(matriz[:,1] == matriz[:,1].max())
print("las votaciones:\n"+str(matriz)+"\n El ganador es: "+str(matriz[ganador[0]]))
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNEs 1732



Formando personas que transforman



Ejercicio en Python (PYTHON_BASIC_XXXX)

Diseñe un programa donde 6 candidat@s se presentan a la disputa para la Gobernación del departamento (estado), los candid@tos de los siguientes partidos:

- 1 – socio-democratico
- 2 – Ultra-derechista
- 3 – primero los ricos
- 4 – Centro demoniaco
- 5 – Cambio invertido
- 6 – Alianza queremos más pobres

Los votos de cada candidat@s son generados de forma aleatoria entre 1 y 4500 votos máximo para cada uno de los 123 municipios del departamento, el software debe determinar quién fue ganador@.

Nota: Creamos una matriz llamada votos, compuesta por 123 filas (municipios) y 6 columnas (candidat@s) y cuyo valor de cada celda sea un randomico entre 1 y 3000



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Ejercicio en Python (PYTHON_BASIC_XXXX)

P1Tx: Votaciones para la Gobernación de departamento en el 20xx:

1. Recorremos las columnas de la matriz: *for candidato in range(len(votos_20xx[0]))*
2. Creamos una nueva matriz llamada *tot_votos* de tipo numpy donde la primera columna sea el *candidato+1* y la segunda columna sea el total de votos obtenidos: *votos_20xx[:,candidato].sum()*
3. Con la nueva matriz *tot_votos* determinamos el valor máximo de la segunda columna: *tot_votos[:,1].max()*
4. y por ultimo visualizamos todo el registro donde se encuentra el valor máximo

```
import numpy as np; from random import randint
votos_20xx = np.random.randint(3000,size=(123,6))
tot_votos = np.array([[candidato+1,votos_20xx[:,candidato].sum()] for candidato in range(len(votos_20xx[0]))])
ganador = np.where(tot_votos == tot_votos[:,1].max())
print("El candidato ganador fue el: "+str(tot_votos[ganador[0]])+"\n El total de votos por candidato es \n"+str(tot_votos))
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE's 1732



Formando personas que transforman



Ejercicio en Python (PYTHON_BASIC_XXXX)

P1Tx: Crear un programa en PYTHON para simular una carrera de formula uno, en donde:
Existen 5 vehículos

1. Renault
2. Ferrari
3. Mercedes,
4. Ford
5. Red bull

La carrera esta diseñada para 4 vueltas donde el tiempo mínimo es 8.2 segundos y el máximo es 10 segundos.

Diseñe la matriz de 5 x 4 donde determine cual es el vehículo ganador donde los tiempos de cada vehículo por vuelta se un randomico.



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



Ejercicio en Python (PYTHON_BASIC_XXXX)

P1Tx: Crear un programa en PYTHON para simular una carrera de formula uno, en donde existen 5 vehículos

- Renault
- Ferrari
- Mercedez
- Ford
- Red bull

La carrera esta diseñada para 4 vueltas donde el tiempo mínimo es 8.2 segundos y el máximo es 10 segundos. Diseñe la matriz de 5 x 4 donde determine cual es el vehículo ganador donde los tiempos de cada vehículo por vuelta se un randomico.

```
import numpy as np
matriz = np.random.uniform(8.2,10,size=(4,5))
tot_vueltas = np.array([[corredor+1,matriz[:,corredor].sum()] for corredor in range(len(matriz[0]))])
ganador=np.where(matriz == matriz[:,1].min())
print ("Resultados de la carrera:\n"+str(matriz)+"\nEl Piloto ganador fue el: "+str(tot_vueltas[ganador[0]]))
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE'S 1732



Formando personas que transforman



Scipy

open-source software
for mathematics, science, and engineering

Release:	1.2.1
Date:	February 10, 2019



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman



SciPy es:

- una colección de algoritmos matemáticos y funciones de conveniencia construidas en la extensión Numpy de Python.
- Agrega un poder significativo a la sesión interactiva de Python al proporcionarle al usuario comandos y clases de alto nivel para manipular y visualizar datos.
- Un entorno de procesamiento de datos y prototipos de sistemas en Python que compite con sistemas como MATLAB, IDL, Octave, R-Lab y SciLab.



SciPy está organizado en subpaquetes que cubren diferentes dominios de computación científica. Estos se resumen en la siguiente tabla

Subpaquete	Descripción
cluster	Algoritmos de agrupamiento
constants	Constantes físicas y matemáticas.
fftpack	Rutinas de transformada rápida de Fourier
integrate	Integración y resolución de ecuaciones diferenciales ordinarias.
interpolate	Interpolación y alisado de splines.
io	Entrada y salida
linalg	Álgebra lineal
ndimage	Procesamiento de imágenes en N dimensiones.
odr	Regresión ortogonal a distancia
optimize	Optimización y rutinas de búsqueda de raíces.
signal	Procesamiento de la señal
sparse	Matrices dispersas y rutinas asociadas.
spatial	Estructuras de datos espaciales y algoritmos.
special	Funciones especiales
stats	Distribuciones estadísticas y funciones.



matplotlib

biblioteca de graficos 2D de Python

Puede generar gráficos:

- histogramas,
- espectros de potencia,
- de barras,
- de error,
- de dispersión, etc.,

Con solo unas pocas líneas de código



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



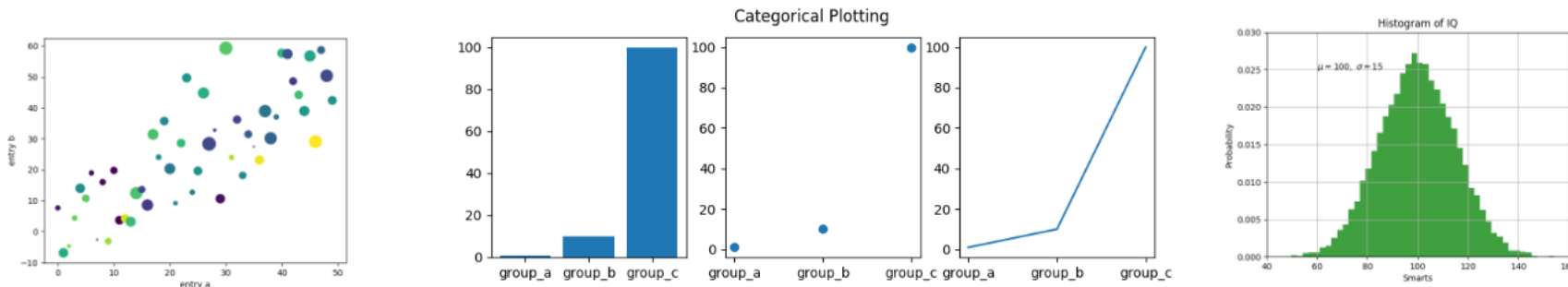
Formando personas que transforman



Matplotlib.pyplot

Para un trazado simple, el módulo **pyplot** proporciona una interfaz similar a **MATLAB**, particularmente cuando se combina con IPython.

Para un usuario avanzado, el tiene el control total de los estilos de línea, las propiedades de fuente, las propiedades de los ejes, etc., a través de una interfaz orientada a objetos o mediante un conjunto de funciones familiares para los usuarios de MATLAB.



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE'S 1732

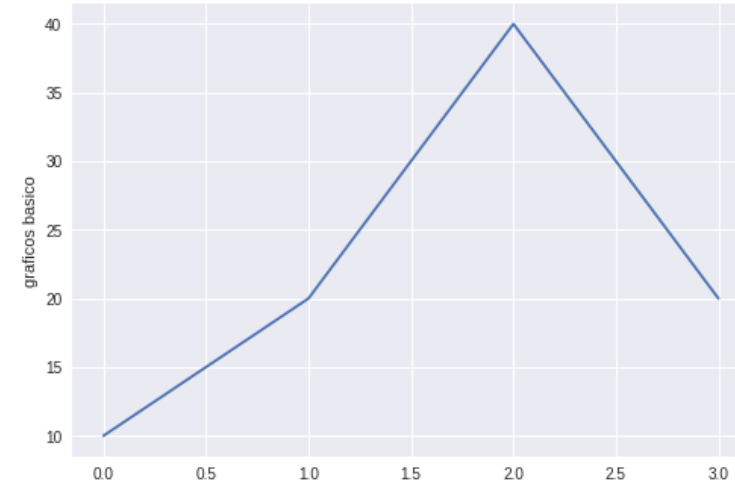


Formando personas que transforman

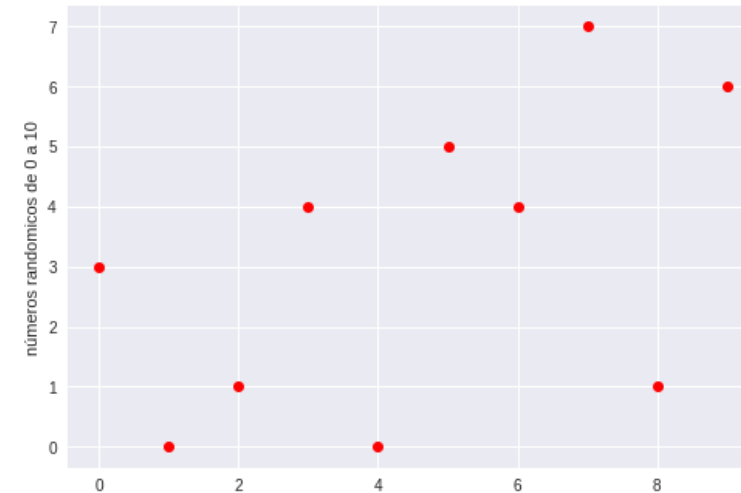


Matplotlib.pyplot

```
import matplotlib.pyplot as plt  
plt.plot([10, 20, 40, 20])  
plt.ylabel("graficos basico")  
plt.show()
```



```
import matplotlib.pyplot as plt  
import random  
lista= [ random.randrange(10) for i in range(10)]  
plt.plot(lista,'ro')  
plt.ylabel("números randomicos de 0 a 10")  
plt.show()
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE's 1732



Formando personas que transforman



P1Tx Matplotlib.pyplot

P1Tx: visualice en gráficos de dispersión [pyplot (scatter)] o de barras tomando los valores obtenidos de los ejercicios:

- P1Tx candidat@s a la alcaldía (ayuntamiento) - barras
- P1Tx (carrera de formular 1) – dispersión (colores por cada vehículo)



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1732



Formando personas que transforman

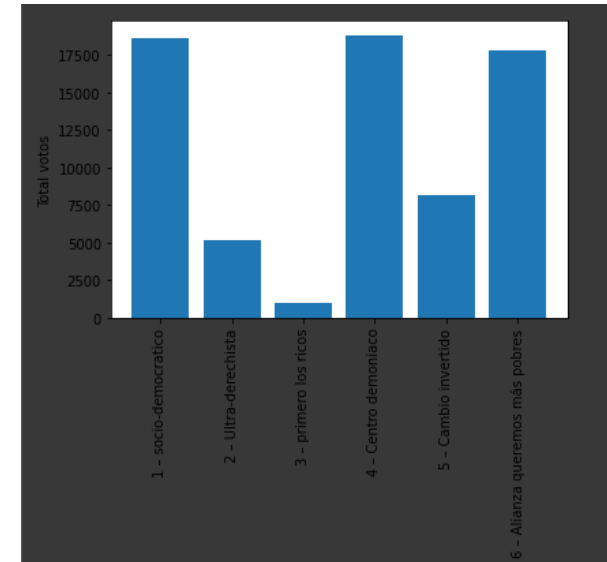


P1Tx Matplotlib.pyplot

P1Tx: visualice en gráficos de dispersión [pyplot (scatter)] o de barras tomando los valores obtenidos de los ejercicios:

- P1Tx (candidat@s a la alcaldía del ayuntamiento - barras

```
import numpy as np ; from random import randint; import matplotlib.pyplot as plt
matriz = np.array([[i+1, randint(1,20000)] for i in range(6)])
ganador= np.where(matriz == matriz[:,1].max())
#Diagrama de Barras
fig = plt.figure('Gráfica de barras') #Generacion de ventana
ax = fig.add_subplot(1,1,1) #Un eje coordinado
nombres = ['1 - socio-democratico','2 - Ultra-derechista','3 - primero los ricos','4 - Centro demoniaco','5 - Cambio invertido','6 - Alianza queremos más pobres']#Arreglo con los partidos
ax.bar(range(len(matriz)),matriz[:,1])#Funcion var para diagramar, con parametros de votos y puntos a ubicar
ax.set_xticks(range(len(matriz[:,1]))) #Numero de puntos a ubicar
ax.set_xticklabels(nombres,rotation='vertical') #Agregar etiquetas a cada punto ubicado en el eje x
plt.ylabel("Total votos");plt.xlabel("Candidatos")
plt.show()
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE'S 1732



Formando personas que transforman



P1T9 Matplotlib.pyplot

P1Tx: visualice en gráficos de dispersión [pyplot (scatter)] o de barras tomando los valores obtenidos de los ejercicios:

- P1Tx (carrera de formular 1) – dispersión (colores por cada vehículo)

```
import numpy as np
matriz = np.random.uniform(8.2,10,size=(4,5))
tot_vueltas = np.array([[corredor+1,matriz[:,corredor].sum()] for corredor in range(len(matriz[0]))])
ganador = np.where(matriz == matriz[:,1].min())
print(" El tiempo total por cada corredor fue: " + "\n" + str(tot_vueltas) + "\n" + " El corrdor ganador fue el numero: " + str(tot_vueltas[ganador[0]]))
#Diagrama de dispersion
x = tot_vueltas[:,1]
y = tot_vueltas[:,0]
fig = plt.figure()
fig.add_subplot(111)
plt.xlabel('Corredores')
plt.ylabel('Total tiempo')
plt.scatter(y, x, c=x)
plt.show()
```



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNE'S 1732



Formando personas que transforman



Final



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA
T U N J A
VIGILADA MINEDUCACIÓN - SNIES 1733



Formando personas que transforman

