

```
# -*- coding: utf-8 -*-
"""Untitled0.ipynb
```

```
Automatically generated by Colab.
```

```
Original file is located at
https://colab.research.google.com/drive/1\_Mth9TG2AAQr63HMorRq2mkodUzVtAWb
"""
```

```
import numpy as np
```

```
T = np.array([
    [0.85, 0.05, 0.075],
    [0.05, 0.85, 0.075],
    [0.1, 0.1, 0.85]
```

```
])

#  $T^3$ 
T_3 = np.linalg.matrix_power(T, 3)
print('T_3', T_3)
```

```
#Estado inicial
x0 = np.array([0.60, 0.20, 0.20])
```

```
# Distribución después de tres campañas
x3 = np.dot(T_3, x0)
print('Distribución después de tres campañas', x3)
```

```
valor_inicial = 0.60
valor_final = 0.4446
```

```
variacion_porcentual = ((valor_final - valor_inicial) / valor_inicial) * 100
```

```
print('variacion_porcentual de A', variacion_porcentual)
```

```
import numpy as np
```

```
P = np.array([
    [0.85, 0.05, 0.075],
    [0.05, 0.85, 0.075],
    [0.10, 0.10, 0.85]
```

```
])

# Valores propios y vectores propios
eigenvalues, eigenvectors = np.linalg.eig(P)
```

```
# Vector propio asociado al valor propio  $\lambda = 1$ 
index = np.argmax(eigenvalues)
stationary_vector = eigenvectors[:, index]
```

```
# Normalizar el vector propio
stationary_vector_normalized = stationary_vector / np.sum(stationary_vector)
```

```
print("Valores propios:", eigenvalues)
print("Vector propio asociado a  $\lambda=1$  (no normalizado):", stationary_vector)
print("Vector propio normalizado (distribución a largo plazo):", stationary_vector_normalized)
```

```
import numpy as np
```

```
# modelo matricial
transition_matrix = np.array([
    [0.5, 2, 1],
    [0.3, 0.6, 0],
    [0, 0.4, 0.9]
```

```
])

initial_state = np.array([100, 50, 20])
```

```
# Estado de la población después de 4 meses
state_after_1_month = np.dot(transition_matrix, initial_state)
state_after_2_months = np.dot(transition_matrix, state_after_1_month)
state_after_3_months = np.dot(transition_matrix, state_after_2_months)
state_after_4_months = np.dot(transition_matrix, state_after_3_months)
```

```
#Población después de 4 meses
print('Población después de 4 meses', state_after_4_months)
```

```
import numpy as np
```

```
# modelo matricial
P = np.array([
    [0.5, 2, 1],
```

```
[0.3, 0.6, 0],  
[0, 0.4, 0.9]
```

```
1)
```

```
# Población inicial  
initial_population = np.array([100, 50, 20])  
  
# Calcular la matriz de transición después de 4 meses  
P_4_months = np.linalg.matrix_power(P, 4)  
  
# Comportamiento de la población a largo plazo  
population_4_months = initial_population @ P_4_months  
print('Comportamiento de la población a largo plazo', population_4_months)
```