

Práctica Monitoreo de Modelos.
Manejo de Data Drift
Instrucciones:

Generación de Datos:

- Genera dos conjuntos de datos con valores numéricos. Estos conjuntos pueden representar frecuencias o probabilidades y deben sumar 1 (si son probabilidades).
- Por ejemplo, podrías tener dos listas de números que representen histogramas o distribuciones de probabilidad.

```
import numpy as np  
data_distribution_1 = np.array([0.2, 0.3, 0.1, 0.4])  
data_distribution_2 = np.array([0.1, 0.4, 0.2, 0.3])
```

Cálculo de la Distancia de Wasserstein:

- Utiliza la función `wasserstein_distance` de la biblioteca `scipy.stats` para calcular la distancia de Wasserstein entre los dos conjuntos de datos.

```
from scipy.stats import wasserstein_distance
```

```
wasserstein_dist = wasserstein_distance(data_distribution_1,  
data_distribution_2)  
print(f"Distancia de Wasserstein: {wasserstein_dist}")
```

Cálculo de la Distancia de Jensen-Shannon:

- Implementa una función para calcular la distancia de Jensen-Shannon entre los dos conjuntos de datos.

```
def jensen_shannon_distance(p, q):  
    m = 0.5 * (p + q)  
    js_distance = 0.5 * (wasserstein_distance(p, m) + wasserstein_distance(q,  
m))  
    return js_distance
```

```
jensen_shannon_dist = jensen_shannon_distance(data_distribution_1,  
data_distribution_2)  
print(f"Distancia de Jensen-Shannon: {jensen_shannon_dist}")
```

Análisis e Interpretación:

- ***Compara las distancias obtenidas. ¿Cómo interpretarías estos resultados? ¿Qué implicaciones tiene para la similitud o diferencia entre las distribuciones?***