d31b4867-8870-44f1-a923-d728048022d7

December 1, 2024

Comentario general del revisor Status del proyecto: Aprobado

¡Hola! Soy **Francisco Cortés**, estoy contento de revisar tu proyecto y ser parte de tu proceso de aprendizaje. A lo largo del texto, haré algunas observaciones sobre mejoras en el código y también haré comentarios sobre tus percepciones sobre el tema. Si existe algún error en el código, no te preocupes, estoy aquí para ayudarte a mejorarlo, en la primera iteración te lo señalaré para que tengas la oportunidad de corregirlo, pero si aún no encuentras una solución para esta tarea, te daré una pista más precisa en la próxima iteración y también algunos ejemplos prácticos. Estaré abierto a retroalimentación y discusiones sobre el tema. Encontrarás mis comentarios a continuación - **por favor no los muevas, modifiques o borres**. Revisaré cuidadosamente tu código para comprobar que se han cumplido con los requisitos y te proporcionaré mis comentarios en cajas verdes, amarillas o rojas como esta:

Comentario del revisor

Si la ejecución fue perfecta succesfully.

Comentario del revisor

Si existe alguna recomendación para que tu código mejore.

Comentario del revisor

Si existen correcciones necesarias para cumplir con los requisitos. El trabajo no puede ser aceptado si hay alguna caja roja.

Puedes responderme de la siguiente manera:

Respuesta del estudiante.

[1]:
```
#PASO 4

## 4.1: Análisis exploratorio de datos (EDA)

import pandas as pd

# Cargar los datos
df_company_trips = pd.read_csv('/datasets/project_sql_result_01.csv')
df_neighborhood_trips = pd.read_csv('/datasets/project_sql_result_04.csv')

# Ver las primeras filas de ambos archivos
print(df_company_trips.head())
```

```
print(df_neighborhood_trips.head())
```

```
                     company_name  trips_amount
0                       Flash Cab         19558
1        Taxi Affiliation Services         11422
2                Medallion Leasing         10367
3                      Yellow Cab          9888
4  Taxi Affiliation Service Yellow          9299
  dropoff_location_name  average_trips
0                  Loop   10727.466667
1           River North    9523.666667
2         Streeterville    6664.666667
3             West Loop    5163.666667
4                O'Hare    2546.900000
```

[2]:
```
## 4.2 Estudiar los datos

# Revisa el tipo de datos y las primeras filas
print(df_company_trips.info())
print(df_neighborhood_trips.info())

# Revisa las estadísticas descriptivas
print(df_company_trips.describe())
print(df_neighborhood_trips.describe())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64 entries, 0 to 63
Data columns (total 2 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   company_name  64 non-null     object
 1   trips_amount  64 non-null     int64 
dtypes: int64(1), object(1)
memory usage: 1.1+ KB
None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 94 entries, 0 to 93
Data columns (total 2 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   dropoff_location_name  94 non-null     object 
 1   average_trips          94 non-null     float64
dtypes: float64(1), object(1)
memory usage: 1.6+ KB
None
       trips_amount
```

```
count     64.000000
mean    2145.484375
std     3812.310186
min        2.000000
25%       20.750000
50%      178.500000
75%     2106.500000
max    19558.000000
       average_trips
count      94.000000
mean      599.953728
std      1714.591098
min         1.800000
25%        14.266667
50%        52.016667
75%       298.858333
max     10727.466667
```

Comentario del revisor

Correcto!

Buena manera de leer los datos

[3]:
```
## 4.3 Asegurar de que los tipos de datos sean correctos:

# Convertir las fechas (si hay columnas de fechas)
# df['start_ts'] = pd.to_datetime(df['start_ts'])

# Asegúrate de que las columnas numéricas sean tipo numérico
df_company_trips['trips_amount'] = pd.
 ↪to_numeric(df_company_trips['trips_amount'], errors='coerce')
df_neighborhood_trips['average_trips'] = pd.
 ↪to_numeric(df_neighborhood_trips['average_trips'], errors='coerce')
```

Comentario del revisor

Correcto!

Buena manera de asegurar que el tipo de dato en las columnas sea correcto

[5]:
```
## 4.4 Identificar los 10 principales barrios en términos de finalización del␣
 ↪recorrido:

top_neighborhoods = df_neighborhood_trips.nlargest(10, 'average_trips')
print(top_neighborhoods)
```

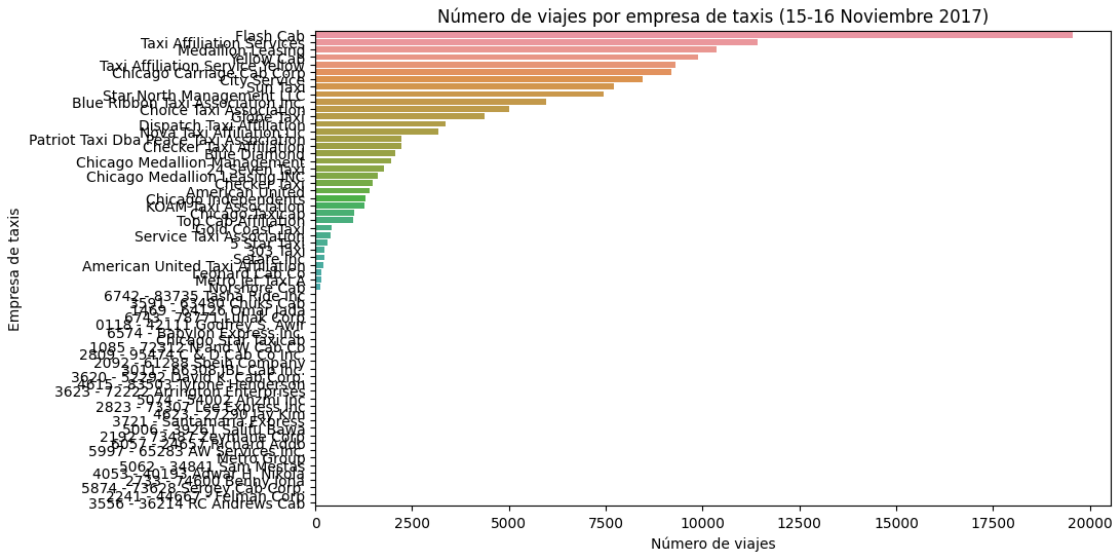
```
  dropoff_location_name  average_trips
0                  Loop   10727.466667
1           River North    9523.666667
2         Streeterville    6664.666667
```

```
3              West Loop    5163.666667
4                 O'Hare    2546.900000
5              Lake View    2420.966667
6             Grant Park    2068.533333
7         Museum Campus    1510.000000
8            Gold Coast    1364.233333
9    Sheffield & DePaul    1259.766667
```

[6]:
```
## 4.5 Gráfico 1:

import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 6))
sns.barplot(x='trips_amount', y='company_name', data=df_company_trips.
 ↪sort_values('trips_amount', ascending=False))
plt.title('Número de viajes por empresa de taxis (15-16 Noviembre 2017)')
plt.xlabel('Número de viajes')
plt.ylabel('Empresa de taxis')
plt.show()
```



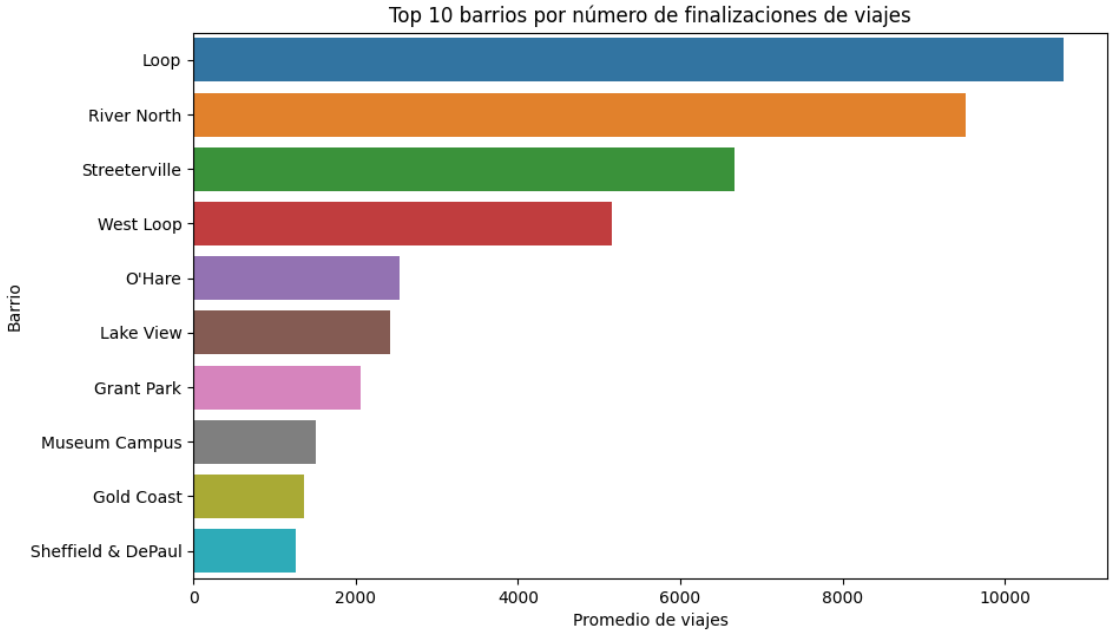
Comentario del revisor

Correcto!

Buena manera de mostrar el resultado, se puede observar claramente cual compañía es la que tiene más viajes agendados

[7]:
```
## 4.6 Gráfico 2:
```

```
plt.figure(figsize=(10, 6))
sns.barplot(x='average_trips', y='dropoff_location_name',␣
 ↪data=top_neighborhoods)
plt.title('Top 10 barrios por número de finalizaciones de viajes')
plt.xlabel('Promedio de viajes')
plt.ylabel('Barrio')
plt.show()
```



Comentario del revisor

Correcto!

Buena manera de crear una grafica que nos muestra los lugares más frecuentados

0.1 4.6 Conclusiones:

Gráfico 1: Empresas de taxis y número de viajes Este gráfico nos muestra cuántos viajes realizó cada empresa de taxis entre el 15 y el 16 de noviembre de 2017. Al observar el gráfico, podrías identificar:

Empresas con mayor número de viajes: Si alguna empresa tiene barras más largas, significa que esa empresa de taxis realizó una gran cantidad de viajes en esos días específicos. Empresas con menos viajes: Las barras más cortas indican empresas que realizaron menos viajes durante este periodo. Esto puede sugerir que esas empresas tienen una menor cuota de mercado o simplemente operan menos frecuentemente durante estos días. Comparación entre empresas: Este análisis nos permite ver si hay alguna empresa dominante o si las empresas tienen una distribución más equitativa en términos de viajes.

Gráfico 2: Los 10 barrios principales por número de finalizaciones de viajes Este gráfico ilustra

cuáles son los barrios más populares en Chicago donde terminan los viajes. Al observar los datos, podrías concluir:

Barrio con más viajes finalizados: El barrio con la barra más alta tiene la mayor cantidad de viajes que terminaron allí. Este podría ser un área con alta actividad de taxis, como zonas comerciales, turísticas o de gran densidad poblacional. Comparación de barrios: Al observar los 10 barrios principales, puedes ver qué áreas tienen más finalizaciones de viajes en comparación con otras. Esto puede reflejar factores como la accesibilidad, la demanda de transporte o la proximidad a puntos de interés (por ejemplo, aeropuertos, centros comerciales, etc.). Zonas menos populares: Los barrios con barras más cortas podrían indicar menos demanda de taxis en esas áreas. Esto podría ser debido a que son zonas residenciales, menos accesibles, o simplemente menos transitadas en comparación con los barrios más grandes. Conclusiones Generales: Empresas más populares: Podría haber una o dos empresas dominantes en el mercado de taxis durante estos días, lo que sugiere que las empresas con más viajes pueden tener mayores recursos o una red de clientes más amplia. Áreas de alta demanda: Los barrios con más viajes pueden indicar zonas de mayor actividad económica o turística. Estos datos son útiles para que las empresas de taxis puedan optimizar sus recursos y concentrar sus esfuerzos en áreas con mayor demanda. El análisis visual de estos gráficos te ayuda a entender tanto la distribución de los viajes entre diferentes empresas de taxis como la demanda en varias ubicaciones de la ciudad.

Comentario del revisor

Correcto!

Las observaciones que haces me parecen bastante acertadas

[8]:
```
##PASO 5 Prueba de hipótesis

from scipy import stats

# Cargar el archivo de los viajes desde el Loop hasta O'Hare
df_trips = pd.read_csv('/datasets/project_sql_result_07.csv')

# Filtrar los viajes en días lluviosos y no lluviosos
rainy_day_trips = df_trips[df_trips['weather_conditions'] ==␣
 ↪'Bad']['duration_seconds']
good_day_trips = df_trips[df_trips['weather_conditions'] ==␣
 ↪'Good']['duration_seconds']

# Realizar la prueba t para comparar las medias
t_stat, p_value = stats.ttest_ind(rainy_day_trips, good_day_trips,␣
 ↪nan_policy='omit')

# Imprimir los resultados
print(f'T-statistic: {t_stat}')
print(f'P-value: {p_value}')

# Decidir si rechazar o no la hipótesis nula
alpha = 0.05
```

```
if p_value < alpha:
    print("Rechazamos la hipótesis nula: la duración de los viajes cambia los␣
 ↪sábados lluviosos.")
else:
    print("No se rechaza la hipótesis nula: no hay diferencia significativa en␣
 ↪la duración de los viajes.")
```

```
T-statistic: 6.946177714041499
P-value: 6.517970327099473e-12
Rechazamos la hipótesis nula: la duración de los viajes cambia los sábados
lluviosos.
```

Comentario del revisor

Bien hecho!

Es correcto rechazar la hipotesis nula, debido a que sí hay notables cambios entre un día de buenas condiciones y uno con malas condiciones

##Interpretacion:

R= Si el p-valor es menor que el nivel de significación (0.05), se rechaza la hipótesis nula, lo que indicaría que la duración de los viajes varía los sábados lluviosos.

0.2 Comentario general del revisor

Comentario del revisor Has realizado un buen trabajo, me doy cuenta de que has aplicado los conocimientos que has adquirido durante el curso, los procedimientos realizados son correctos, este es un ejercicio que nos ayuda a entender y comprobar las hipotesis con procesos estadisticos.

Continúa con el buen trabajo y mucho éxito en el siguiente Sprint!