Entendimiento_de_los_Datos_Muestra

May 4, 2025

[2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

1 Descripción inicial

[15]:
```
muestra = pd.read_excel('muestra1.xlsx')
muestra.head()
```

[15]:

	Orden	Sucursal	FechaID	HoraLLegada	TurnoID	Turno	TurnoTipo	TurnoHoraInicio	TurnoHoraFin	TurnoMinutosEspera	TAPRecepcionMinutos	TAPRecepcionCaja
0	316483	COYOACAN	20240301	6	41684208	N015	Solicitar Estudios	06:02:44	06:13:23	NaN	NaN	NaN
1	316499	COYOACAN	20240301	6	41684414	N025	Solicitar Estudios	06:09:35	06:19:43	10.13	NaN	NaN
2	316515	COYOACAN	20240301	6	41684824	N038	Solicitar Estudios	06:18:16	06:30:42	12.43	NaN	NaN
3	316531	COYOACAN	20240301	6	41684679	P004	Triage	06:25:09	06:27:01	1.87	NaN	NaN
4	316547	COYOACAN	20240301	6	41685173	C011	Citado	06:35:09	06:37:47	2.63	NaN	NaN

1.1 Número de observaciones y variables

[4]:
```
muestra.shape
```

[4]:
```
(59650, 12)
```

Esta base de datos cuenta con 12 variables y 59,650 observaciones.

1.2 Breve descripción de las variables

1.2.1 Tipo de datos:

[5]:
```
muestra.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 59650 entries, 0 to 59649
Data columns (total 12 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Orden                59650 non-null  int64
 1   Sucursal             59650 non-null  object
 2   FechaID              59650 non-null  int64
 3   HoraLLegada          59650 non-null  int64
 4   TurnoID              59650 non-null  int64
 5   Turno                59650 non-null  object
 6   TurnoTipo            59650 non-null  object
 7   TurnoHoraInicio      59650 non-null  object
 8   TurnoHoraFin         59650 non-null  object
 9   TurnoMinutosEspera   59649 non-null  float64
 10  TAPRecepcionMinutos  26316 non-null  float64
 11  TAPRecepcionCaja     26316 non-null  float64
dtypes: float64(3), int64(4), object(5)
memory usage: 5.5+ MB
```

Las variables *Orden, FechaId, HoraLLegada, TurnoID* son variables de tipo int. Las variables *Sucursal, Turno, TurnoTipo, TurnoHoraInicio, TurnoHoraFin* son variables tipo object y *TurnoMinutosEspera, TAPRecepcionMinutos, TAPRecepcionCaja* son tipo float.

1.2.2 Unidades de medida y significado de cada variable

Variable	Significado	Unidad de medida
`Orden`	Número de orden del registro	Entero (sin unidad física)
`Sucursal`	Nombre de la sucursal	Texto (nombre)
`FechaID`	Fecha en formato numérico (tipo AAAAMMDD)	Entero (fecha codificada)
`HoraLLegada`	Hora en la que el cliente llegó (tipo H)	Entero
`TurnoID`	Identificador único del turno asignado	Entero
`Turno`	Código del turno asignado (tipo N####)	Texto
`TurnoTipo`	Tipo de turno (por ejemplo: que estudios se van a realizar)	Texto
`TurnoHoraInicio`	Hora en la que comenzó el turno (formato de hora: HH:MM:SS)	Tiempo
`TurnoHoraFin`	Hora en la que terminó el turno (formato de hora: HH:MM:SS)	Tiempo

Variable	Significado	Unidad de medida
TurnoMinutosEspera	Tiempo de espera entre llegada y atención (TurnoHoraFin-TurnoHoraInicio)	Minutos
TAPRecepcionMinutos	Tiempo en caja de atención	Minutos
TAPRecepcionCaja	ID de la caja en la que fue atentidido	Entero (sin unidad física)

1.3 Variable categóricas

[17]:
```
muestra['Sucursal'].unique()
```

[17]:
```
array(['COYOACAN', 'CULIACAN', 'CULIACAN CAÑADAS',
       'CULIACAN COLEGIO MILITAR', 'CULIACAN LA CONQUISTA'], dtype=object)
```

[19]:
```
muestra['TurnoTipo'].unique()
```

[19]:
```
array(['Solicitar Estudios', 'Triage', 'Citado', 'Citados sin folio',
       'Folio Pagado', 'Cotizacion', 'Examen de la Vista',
       'Atención Empresas', 'Abono a Lentes', 'Estudio Pendiente',
       'Entrega de Resultados'], dtype=object)
```

2 Exploración de los datos

2.1 Estadísticas descriptivas

[20]:
```
muestra.describe()
```

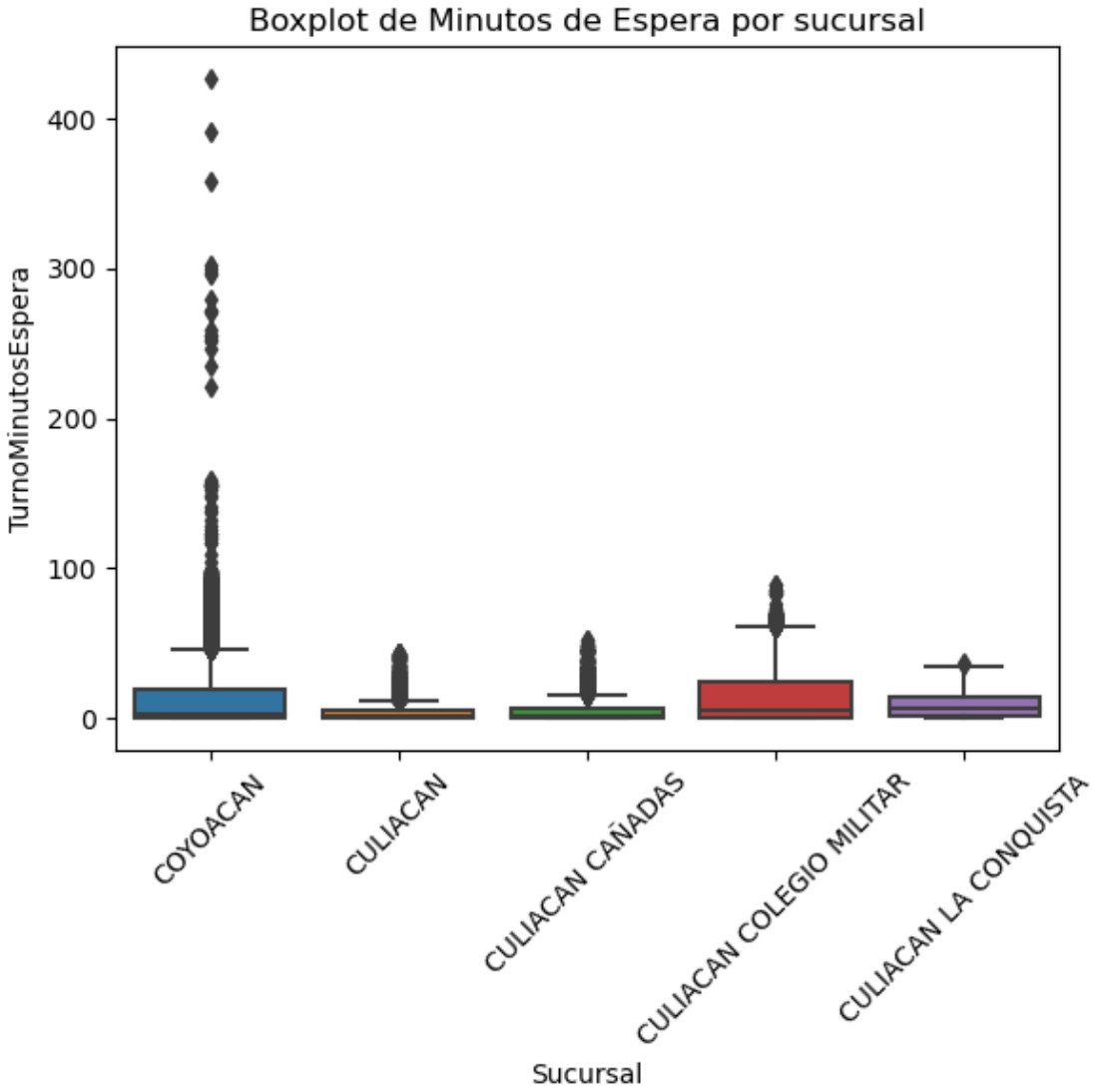
[20]:
```
                Orden       FechaID   HoraLLegada       TurnoID  \
count    59650.000000  5.965000e+04  59650.000000  5.965000e+04
mean    386877.426035  2.024031e+07      9.992406  4.259338e+07
std      55158.491105  7.809021e+00      3.154808  5.238362e+05
min     316468.000000  2.024030e+07      0.000000  4.168380e+07
25%     331380.250000  2.024031e+07      7.000000  4.213963e+07
50%     425463.500000  2.024031e+07      9.000000  4.259950e+07
75%     440375.750000  2.024032e+07     12.000000  4.304578e+07
max     455288.000000  2.024033e+07     21.000000  4.349509e+07

       TurnoMinutosEspera  TAPRecepcionMinutos  TAPRecepcionCaja
count        59649.000000         26316.000000      26316.000000
mean            10.039047             3.518992       2218.669897
std             17.460828             2.051851       1577.809132
min              0.000000             0.000000       1134.000000
25%              0.250000             2.220000       1137.000000
50%              2.330000             3.180000       1202.000000
75%             11.550000             4.430000       3758.000000
```

```
max          426.830000          64.850000        7033.000000
```

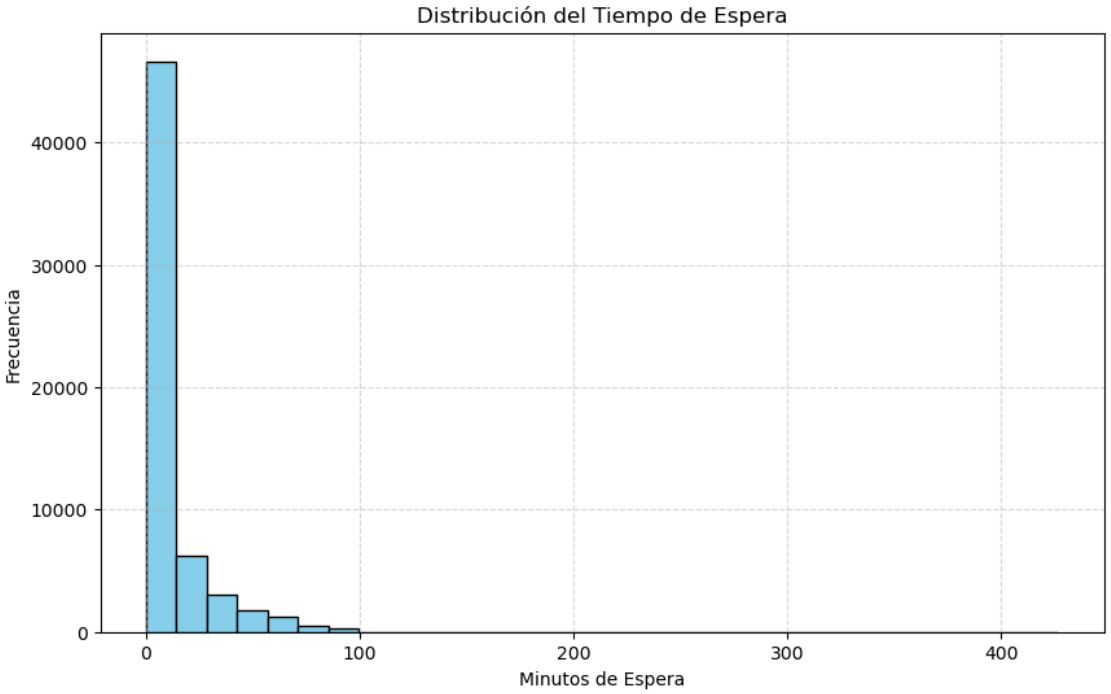
2.2 Gráficas exploratorias

[21]:
```
sns.boxplot(x='Sucursal', y='TurnoMinutosEspera', data=muestra)
plt.xticks(rotation=45)
plt.title('Boxplot de Minutos de Espera por sucursal')
plt.show()
```



[25]:
```
plt.figure(figsize=(10, 6))
plt.hist(muestra['TurnoMinutosEspera'], bins=30, color='skyblue',␣
↪edgecolor='black')
```

```
plt.title('Distribución del Tiempo de Espera')
plt.xlabel('Minutos de Espera')
plt.ylabel('Frecuencia')
plt.grid(True, linestyle='--', alpha=0.5)
plt.show()
```



2.3 Hallazgos importantes

2.3.1 Minutos de espera por turno por sucursal

[26]:
```
muestra.groupby('Sucursal')['TurnoMinutosEspera'].describe()
```

[26]:

Sucursal	count	mean	std	min	25%	50%	75%	max
COYOACAN	29071.0	13.321135	21.501693	0.0	0.5000	2.82	18.43	426.83
CULIACAN	12723.0	3.633254	5.892265	0.0	0.1000	1.13	4.51	43.77
CULIACAN CAÑADAS	8260.0	4.833964	7.473694	0.0	0.1300	1.75		
CULIACAN COLEGIO MILITAR	7739.0	14.147076	18.078719	0.0	0.5900	4.48		
CULIACAN LA CONQUISTA	1856.0	8.578524	8.046691	0.0	0.9775	6.55		

```
CULIACAN CAÑADAS           6.10   51.68
CULIACAN COLEGIO MILITAR  24.53   88.65
CULIACAN LA CONQUISTA     14.57   37.03
```

2.3.2 Tiempo promedio de atención por Sucursal

[27]:
```
muestra.groupby('Sucursal')['TAPRecepcionMinutos'].mean()
```

[27]:
```
Sucursal
COYOACAN                       NaN
CULIACAN                  4.061060
CULIACAN CAÑADAS          3.025481
CULIACAN COLEGIO MILITAR  3.306680
CULIACAN LA CONQUISTA     2.793902
Name: TAPRecepcionMinutos, dtype: float64
```

2.3.3 Tiempo promedio de espera por Sucursal

[28]:
```
muestra.groupby('Sucursal')['TurnoMinutosEspera'].mean()
```

[28]:
```
Sucursal
COYOACAN                  13.321135
CULIACAN                   3.633254
CULIACAN CAÑADAS           4.833964
CULIACAN COLEGIO MILITAR  14.147076
CULIACAN LA CONQUISTA      8.578524
Name: TurnoMinutosEspera, dtype: float64
```

2.4 Tiempos de espera mayores a 1 hora 40 min

[33]:
```
muestra[muestra['TurnoMinutosEspera'] > 100]['Sucursal'].unique()
```

[33]:
```
array(['COYOACAN'], dtype=object)
```

Aquí podemos ver que la sucursal de Coyoacán tiene los tiempos de espera más grandes. Esto indica una tendencia en la Surcursal de Coyoacán al esperar ser atendido en caja.

[35]:
```
muestra[muestra['TurnoMinutosEspera'] > 100].describe()
```

[35]:

	Orden	FechaID	HoraLLegada	TurnoID	\
count	47.000000	4.700000e+01	47.000000	4.700000e+01	
mean	329748.553191	2.024031e+07	8.574468	4.254708e+07	
std	5977.661848	5.593293e+00	2.849444	3.766064e+05	
min	321525.000000	2.024030e+07	0.000000	4.202955e+07	
25%	321685.500000	2.024030e+07	7.000000	4.202970e+07	
50%	332282.000000	2.024032e+07	8.000000	4.271335e+07	
75%	332334.000000	2.024032e+07	10.000000	4.271818e+07	
max	342998.000000	2.024032e+07	15.000000	4.335566e+07	

	TurnoMinutosEspera	TAPRecepcionMinutos	TAPRecepcionCaja
count	47.000000	0.0	0.0
mean	197.987872	NaN	NaN
std	81.815428	NaN	NaN
min	104.750000	NaN	NaN
25%	134.300000	NaN	NaN
50%	156.030000	NaN	NaN
75%	257.100000	NaN	NaN
max	426.830000	NaN	NaN

2.5 Variables relevantes

Para el problema a solucionar, las variables importantes son `TurnoMinutosEspera` y `TAPRecepcionMinutos`. También nos importan variables como tipo de prioridad y número de cajas por sucursal. Nuestra variable dependiente es `TurnoMinutosEspera` que depende del tipo de prioridad, número de cajas por sucursal, número de personas que llegan a la sucursal por hora y `TAPRecepcionMinutos`.

2.6 Transformaciones prometedoras

Primero, para la variable `TAPRecepcionMinutos` se hará una imputación simple de la media por sucursal, ya que esta variable es importante pero la mitad de sus datos son datos faltantes. Algunas transformaciones prometedoras podrían ser simulaciones de llegada de pacientes por sucursal, por día y por hora. Esto para poder evaluar la solución.

3 Evaluación de la calidad de los datos

3.1 Detección y cuantificación de valores faltantes, valores extremos (outliers), inconsistencias o errores.

A partir de los análisis exploratorios realizados, observamos la presencia de valores inusualmente altos en los tiempos de espera para ser atendido. Si los ponemos en contexto, estos tiempos resultan extremos considerando el servicio que se ofrece. Al consultar con el Socio Formador, nos comentó que algunos usuarios obtienen su turno y posteriormente se retiran del lugar, regresando incluso hasta siete horas después para ser atendidos.

Dado que estamos analizando interacciones humanas dentro de un sistema de atención, es importante considerar el significado detrás de estos valores atípicos. No necesariamente reflejan un fallo del sistema, sino comportamientos específicos de los usuarios. Sin embargo, para efectos analíticos, consideramos que un tiempo de espera mayor a 2 horas (120 minutos) puede clasificarse como un outlier, ya que supera por mucho el comportamiento general observado en los datos.

[44]:
```
columna = muestra['TurnoMinutosEspera']
# Calcular Q1, Q3 y IQR
Q1 = columna.quantile(0.25)
Q3 = columna.quantile(0.75)
IQR = Q3 - Q1
```

```
# Límites inferior y superior
lim_inf = Q1 - 1.5 * IQR
lim_sup = Q3 + 1.5 * IQR

# Identificar outliers
outliers = columna[(columna < lim_inf) | (columna > lim_sup)]


print("Outliers analíticos encontrados:")
print(f"Mínimo : {min(outliers)}\nMáximo : {max(outliers)}")
print(f"Cantidad de outliers analíticos: {len(outliers)}")
```

```
Outliers analíticos encontrados:
Mínimo : 28.45
Máximo : 299.77
Cantidad de outliers analíticos: 6836
```

[45]:
```
# Elimine una muestra donde la persona espero más de 5 horas según el registro
muestra.drop(muestra[muestra['TurnoMinutosEspera'] > 120].index, inplace=True)
```

3.2 Discusión sobre la calidad general de los datos y posibles problemas que podrían afectar el análisis futuro

En general, la calidad de los datos es razonablemente buena, ya que la mayoría de las variables clave no presentan problemas graves de completitud o consistencia. No obstante, existen algunos aspectos que podrían afectar la validez y precisión del análisis, especialmente en lo que respecta a la estimación de los tiempos de espera (TurnoMinutosEspera), nuestra variable dependiente. Uno de los principales retos identificados es el alto número de valores faltantes en la variable TAPRecepcionMinutos, que representa una métrica importante en el flujo del servicio. De los 59,650 registros totales, esta variable solo está disponible para 26,316 observaciones (aproximadamente el 44%). Dado su potencial valor explicativo, se optó por realizar una imputación simple utilizando la media por sucursal. Aunque esto permite conservar la variable en el análisis, es importante reconocer que este tipo de imputación introduce supuestos que podrían suavizar o distorsionar relaciones reales presentes en los datos originales.

Además, variables relevantes como el número de cajas por sucursal y el número de personas que llegan por hora no se encuentran directamente en los datos actuales, lo que requerirá un proceso de transformación o simulación para su incorporación. Esta necesidad puede limitar la rapidez del análisis o introducir complejidad adicional al modelado.

En resumen, aunque los datos disponibles permiten realizar un análisis significativo, es fundamental tener en cuenta estas limitaciones:

1. Alta proporción de datos faltantes en variables clave.
2. Comportamientos atípicos no controlados en los tiempos de espera.
3. Variables explicativas necesarias que no están explícitamente disponibles y requieren ser derivadas.

Estos factores deberán considerarse cuidadosamente al interpretar los resultados de cualquier modelo predictivo o causal, así como en la toma de decisiones basadas en los datos.

3.3 Estrategias preliminares para tratar los problemas identificados.

1. Tratamiento de valores faltantes Variable `TAPRecepcionMinutos`: Se aplicará una imputación simple utilizando la media por sucursal. Esta imputación localizada permite conservar la variable en el análisis sin perder una gran cantidad de registros, respetando las diferencias operativas entre sedes.
2. Manejo de outliers en TurnoMinutosEspera Se establecerá un umbral de 120 minutos (2 horas) para identificar y tratar como outliers los tiempos de espera excesivos. Este criterio se fundamenta tanto en el comportamiento observado como en la retroalimentación del Socio Formador, quien indicó que algunos usuarios pueden ausentarse por varias horas después de solicitar un turno.
3. Generación de variables derivadas Algunas variables clave, como el número de personas que llegan por hora o el número de cajas por sucursal, no están explícitamente presentes en los datos. Se propone:
    - Calcular la carga horaria por sucursal a partir de la agrupación por Sucursal, FechaID y HoraLLegada.
    - Incorporar información adicional, si está disponible, sobre el número de cajas operando por sucursal.
    - Simular llegadas usando modelos de procesos estocásticos como Poisson, en caso de escenarios incompletos.
4. Segmentación por sucursal Dado que el comportamiento de las variables varía significativamente entre sucursales, tanto el análisis exploratorio como la construcción del modelo se realizarán por sucursal. Esta decisión permitirá capturar de manera más precisa las diferencias operativas y de comportamiento entre sedes, evitando generalizaciones que podrían distorsionar los resultados.

4 Primeros insights

4.1 Identificación de patrones o correlaciones iniciales relevantes

De lo descubierto en esta exploración inicial, identificamos las siguientes cosas:

- Tiempo de espera y sucursal: Los tiempos de espera extremadamente altos (>150 minutos) se concentran en una sola sucursal (Coyoacán) y en tres días específicos (5, 15 y 25 de marzo). Esto sugiere un patrón temporal y geográfico muy localizado.
- Tendencia general en los tiempos de espera: La mayoría de los datos se encuentra en un rango razonable (15–60 minutos), lo que permite identificar outliers claros y posibles condiciones normales vs. anómalas.
- Día del mes y carga de trabajo: La aparición recurrente de los días 5, 15 y 25 podría estar relacionada con comportamientos cíclicos en la demanda (por ejemplo, pagos, cortes de quincena, etc.).

- Sucursal y saturación: Solo Coyoacán muestra saturación grave, lo que podría indicar una falla en la capacidad operativa o una demanda anómala.

4.2 Reflexión crítica

Por último, de todo lo anterior se pueden generar diversas propuestas para guiar el enfoque analítico o de modelación posterior:

1. Modelado por sucursal: dado que el comportamiento de Coyoacán difiere tanto del resto, se podría modelar las sucursales de manera separada o incluir interacciones con la variable "Sucursal". Usar un modelo jerárquico podría capturar mejor estas diferencias estructurales.
2. Análisis temporal detallado: los patrones en fechas específicas indican que incluir variables temporales (día del mes, semana, etc.) podría mejorar el poder explicativo del modelo. También se justifica probar efectos no lineales o estacionales.
3. Posibles soluciones:

   No algorítmicas:

   a) Sistema de prioridad dinámica. Salud Digna tiene un sistema de prioridd estática. Se p

   b) Modelar como una cola con envejecimiento. Implementar un mecanismo de envejecimiento e

   Algorítmicas:

   a) Cola de prioridad con función de utilidad.

   b) Algoritmo Round Robin con prioridad.

   c) Shortest Expected Wait Time First (SEWTF)

   d) Time-To-Live Queue (TTLQ)