



**Tecnológico
de Monterrey**

Análisis de datos Huawei
Segundo Avance del Proyecto

Laboratorio de diseño y optimización de operaciones
Profesora: Ana Luisa Masetto Herrera

Gerardo Gutiérrez Aguirre	A01368146
Juan Ramón Rodríguez Márquez	A01365002
José Manuel González Guadarrama	A01364681
Kevin Alberto Sorzano Mongel	A01368603
Mario Naime Monroy	A01365605
José Alberto Naime Monroy	A01365892

Introducción

Tras realizar el planteamiento del problema basándonos en la información de nuestra base de datos, entender la o las problemáticas y plasmar los objetivos principales de nuestro proyecto. Pudimos orientar nuestro proyecto de una manera en la cual podemos tomar los datos más relevantes ya previamente filtrados y enfocarlos a nuestro objetivo de una manera concisa. De este modo podemos determinar las limitaciones de nuestro proyecto y asimismo utilizar las herramientas aprendidas para maximizar nuestras fortalezas de la mano con llegar a los objetivos.

Procederemos ahora al análisis de datos tomando como primer paso la limpieza de datos para consecuentemente realizar y análisis para poder construir una codificación que nos permita proseguir con la etapa de modelado de nuestra investigación y análisis de resultados

Etapa III

La limpieza de datos se llevó a cabo con la ayuda de Jupyter Notebook el cual utiliza lenguaje de programación Python. Empezamos con la lectura de datos, la cual fue primordial realizar la instalación de la paquetería Pandas y abrir la librería sklearn para poder utilizar la función “mean_absolute_error”, para después poder importar nuestro archivo con los datos limpios.

```
LECTURA DE DATOS

In [1]: #Instalar paquetería Pandas
        #!pip install pandas
        import pandas as pd

In [2]: #De la librería sklearn sólo importar la función mean_abs_err
        from sklearn.metrics import mean_absolute_error

In [3]:
        import matplotlib.pyplot as plt

In [4]: #Importar Base de Datos
        datos = pd.read_csv("datos_completos.csv")
```

Posteriormente, mandamos a llamar los primeros 10 renglones de nuestra base de datos, así como la extensión de ésta para poder ver que, efectivamente, íbamos a trabajar con el archivo correcto.

Lunes 18 de octubre de 2021

```
In [5]: # Primeros 10 renglones de la Base de datos
datos.head(10)

Out[5]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	ventas_totales_en_tienda_de_cada_mes	ventas_promedio_en_tienda_de_cada_mes	ventas_tot
0	1	0	1	1	0	29	0.852941	
1	1	1	1	0	0	29	0.852941	
2	1	2	1	0	0	30	0.882353	
3	1	3	1	0	0	25	0.735294	
4	1	4	1	0	0	21	0.617647	
5	1	5	1	0	0	32	0.941176	
6	1	6	1	0	0	16	0.470588	
7	1	7	1	0	0	10	0.294118	
8	1	8	1	0	0	18	0.529412	
9	1	0	2	1	1	29	0.852941	

10 rows x 24 columns

```
In [6]: # num de reng y col
datos.shape

Out[6]: (99999, 24)
```

Visualizamos los encabezados de las columnas para posteriormente poder descartar aquellas que no nos sean de utilidad en el modelo de predicción.

```
In [7]: # datos de columnas
datos.columns

Out[7]: Index(['pdv_id', 'mes_id', 'sku_id', 'ventas_totales',
              'y_ventas_siguiente_mes', 'ventas_totales_en_tienda_de_cada_mes',
              'ventas_promedio_en_tienda_de_cada_mes',
              'ventas_totales_en_tienda_de_cada_sku',
              'ventas_promedio_en_tienda_de_cada_sku', 'ventas_totales_1_mes_pasado',
              'ventas_totales_2_meses_pasados', 'ventas_totales_3_meses_pasados',
              'ventas_totales_tienda_y_mes_del_mes_pasado',
              'ventas_totales_tienda_y_mes_2_pasado',
              'ventas_totales_tienda_y_mes_3_pasado',
              'ventas_promedio_tienda_y_mes_del_mes_pasado',
              'ventas_promedio_tienda_y_mes_2_pasado',
              'ventas_promedio_tienda_y_mes_3_pasado',
              'ventas_totales_tienda_y_sku_del_mes_pasado',
              'ventas_totales_tienda_y_sku_2_pasado',
              'ventas_totales_tienda_y_sku_3_pasado',
              'ventas_promedio_tienda_y_sku_del_mes_pasado',
              'ventas_promedio_tienda_y_sku_2_pasado',
              'ventas_promedio_tienda_y_sku_3_pasado'],
              dtype='object')

In [8]: #Quitar todas estas variables
datos_ma = datos.drop(['ventas_totales_en_tienda_de_cada_mes',
                      'ventas_promedio_en_tienda_de_cada_mes',
                      'ventas_totales_en_tienda_de_cada_sku',
                      'ventas_promedio_en_tienda_de_cada_sku', 'ventas_totales_1_mes_pasado',
                      'ventas_totales_2_meses_pasados', 'ventas_totales_3_meses_pasados',
                      'ventas_totales_tienda_y_mes_del_mes_pasado',
                      'ventas_totales_tienda_y_mes_2_pasado',
                      'ventas_totales_tienda_y_mes_3_pasado',
                      'ventas_promedio_tienda_y_mes_del_mes_pasado',
                      'ventas_promedio_tienda_y_mes_2_pasado',
                      'ventas_promedio_tienda_y_mes_3_pasado',
                      'ventas_totales_tienda_y_sku_del_mes_pasado',
                      'ventas_totales_tienda_y_sku_2_pasado',
                      'ventas_totales_tienda_y_sku_3_pasado',
                      'ventas_promedio_tienda_y_sku_del_mes_pasado',
                      'ventas_promedio_tienda_y_sku_2_pasado',
                      'ventas_promedio_tienda_y_sku_3_pasado'], axis=1)

In [9]: #Mostrar los datos sin esas columnas
datos_ma.head(10)

Out[9]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes
0	1	0	1	1	0
1	1	1	1	0	0
2	1	2	1	0	0
3	1	3	1	0	0
4	1	4	1	0	0
5	1	5	1	0	0
6	1	6	1	0	0
7	1	7	1	0	0
8	1	8	1	0	0
9	1	0	2	1	1

Para poder empezar con el modelo de promedios móviles (Etapa IV) es indispensable cumplir con los pasos anteriores, contar con una base de datos limpia y conocer el funcionamiento de los modelos que queremos aplicar.

Etapas IV

“Los promedios móviles son promedios calculados a partir de subgrupos artificiales de observaciones consecutivas”.

El pronóstico de promedio móvil es óptimo para patrones de demanda aleatorios o nivelados donde se pretende eliminar el impacto de los elementos irregulares históricos mediante un enfoque en períodos de demanda reciente.

```
In [10]: # Crear nueva columna para pedir lo del mes anterior
datos_ma['m1_pedir_lo_del_mes_pasado']=datos_ma['ventas_totales']

In [11]: datos_ma.head(16)
Out[11]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	m1_pedir_lo_del_mes_pasado
0	1	0	1	1	0	1
1	1	1	1	0	0	0
2	1	2	1	0	0	0
3	1	3	1	0	0	0
4	1	4	1	0	0	0
5	1	5	1	0	0	0
6	1	6	1	0	0	0
7	1	7	1	0	0	0
8	1	8	1	0	0	0
9	1	0	2	1	1	1
10	1	1	2	1	1	1
11	1	2	2	1	0	1
12	1	3	2	0	0	0
13	1	4	2	0	0	0
14	1	5	2	0	0	0
15	1	6	2	0	0	0

Usamos el promedio de observaciones para los dos meses anteriores para generar los subgrupos.

```
In [12]: # MA: Pedir un promedio de los 2 meses anteriores
# Agrupar por punto de venta y producto, toma una ventana de 2 y saca el promedio
datos_ma['m2_promedio_de_dos_meses_anteriores'] = datos_ma.groupby(['pdv_id', 'sku_id']).rolling(2)['ventas_totales']

In [13]: datos_ma.head(16)
Out[13]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	m1_pedir_lo_del_mes_pasado	m2_promedio_de_dos_meses_anteriores
0	1	0	1	1	0	1	NaN
1	1	1	1	0	0	0	0.5
2	1	2	1	0	0	0	0.0
3	1	3	1	0	0	0	0.0
4	1	4	1	0	0	0	0.0
5	1	5	1	0	0	0	0.0
6	1	6	1	0	0	0	0.0
7	1	7	1	0	0	0	0.0
8	1	8	1	0	0	0	0.0
9	1	0	2	1	1	1	NaN
10	1	1	2	1	1	1	1.0
11	1	2	2	1	0	1	1.0
12	1	3	2	0	0	0	0.5
13	1	4	2	0	0	0	0.0
14	1	5	2	0	0	0	0.0
15	1	6	2	0	0	0	0.0

Al igual que en el anterior, usamos el promedio de observaciones pero esta vez para los tres meses anteriores para generar los subgrupos.

```
In [14]: # Lo mismo pero ventana de 3
# MA: Pedir un promedio de los 3 meses anteriores
# Agrupar por punto de venta y producto, toma una ventana de 3 y saca el promedio
datos_ma['m3_promedio_de_tres_meses_anteriores'] = datos_ma.groupby(['pdv_id', 'sku_id']).rolling(3)['ventas_totales']

In [15]: datos_ma.head(16)
Out[15]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	m1_pedir_lo_del_mes_pasado	m2_promedio_de_dos_meses_anteriores	m3_promedio_de_tre
0	1	0	1	1	0	1		NaN
1	1	1	1	0	0	0	0	0.5
2	1	2	1	0	0	0	0	0.0
3	1	3	1	0	0	0	0	0.0
4	1	4	1	0	0	0	0	0.0
5	1	5	1	0	0	0	0	0.0
6	1	6	1	0	0	0	0	0.0
7	1	7	1	0	0	0	0	0.0
8	1	8	1	0	0	0	0	0.0
9	1	0	2	1	1	1	1	NaN
10	1	1	2	1	1	1	1	1.0
11	1	2	2	1	0	1	1	1.0
12	1	3	2	0	0	0	0	0.5
13	1	4	2	0	0	0	0	0.0
14	1	5	2	0	0	0	0	0.0
15	1	6	2	0	0	0	0	0.0

De cada columna para cada mes, guardamos el mes para más adelante identificar los datos en el análisis

```
In [16]: #De los datos tomar columna mes id julio y guardarlo en esa variable
error_julio = datos_ma[datos_ma.mes_id == 0]
error_julio.head(5)

Out[16]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	m1_pedir_lo_del_mes_pasado	m2_promedio_de_dos_meses_anteriores	m3_promedio_de_tre
0	1	0	1	1	0	1		NaN
9	1	0	2	1	1	1	1	NaN
18	1	0	3	3	3	3	3	NaN
27	1	0	4	2	4	2		NaN
36	1	0	5	1	0	1		NaN

```
In [17]: #De los datos tomar columna mes id agosto y guardarlo en esa variable
error_agosto = datos_ma[datos_ma.mes_id == 1]
error_agosto.head(5)

Out[17]:
```

	pdv_id	mes_id	sku_id	ventas_totales	y_ventas_siguiente_mes	m1_pedir_lo_del_mes_pasado	m2_promedio_de_dos_meses_anteriores	m3_promedio_de_tre
1	1	1	1	0	0	0	0	0.5
10	1	1	2	1	1	1	1	1.0
19	1	1	3	3	3	3	3	3.0
28	1	1	4	4	3	4	4	3.0
37	1	1	5	0	0	0	0	0.5

Cálculos:

- Aplicar la fórmula para el error, que en este caso es la variación del mes actual y la diferencia respecto al mes anterior (o subgrupo de meses)

```
In [26]: #cálculo manual
#sum(abs(error_julio['y_ventas_siguiente_mes'] - error_julio['m1_pedir_lo_del_mes_pasado']))/64600

In [27]: error_m1_julio = mean_absolute_error(error_julio['y_ventas_siguiente_mes'], error_julio['m1_pedir_lo_del_mes_pasado'])
error_m2_julio = None
error_m3_julio = None

In [35]: error_m1_marzo = mean_absolute_error(error_marzo['y_ventas_siguiente_mes'], error_marzo['m1_pedir_lo_del_mes_pasado'])
error_m2_marzo = mean_absolute_error(error_marzo['y_ventas_siguiente_mes'], error_marzo['m2_promedio_de_dos_meses_anteriores'])
error_m3_marzo = mean_absolute_error(error_marzo['y_ventas_siguiente_mes'], error_marzo['m3_promedio_de_tres_meses_anteriores'])
```

```
In [36]: errores_mae = [['Julio', error_m1_julio, error_m2_julio, error_m3_julio],
                        ['Agosto', error_m1_agosto, error_m2_agosto, error_m3_agosto],
                        ['Septiembre', error_m1_septiembre, error_m2_septiembre, error_m3_septiembre],
                        ['Octubre', error_m1_octubre, error_m2_octubre, error_m3_octubre],
                        ['Noviembre', error_m1_noviembre, error_m2_noviembre, error_m3_noviembre],
                        ['Diciembre', error_m1_diciembre, error_m2_diciembre, error_m3_diciembre],
                        ['Enero', error_m1_enero, error_m2_enero, error_m3_enero],
                        ['Febrero', error_m1_febrero, error_m2_febrero, error_m3_febrero],
                        ['Marzo', error_m1_marzo, error_m2_marzo, error_m3_marzo]]

mae = pd.DataFrame(errores_mae, columns = ['Mes', 'mae_pedir_anterior', 'mae_promedio_2_meses_anteriores', 'mae_promedio_3_meses_anteriores'])
mae
```

```
Out[36]:
```

	Mes	mae_pedir_anterior	mae_promedio_2_meses_anteriores	mae_promedio_3_meses_anteriores
0	Julio	0.568356	NaN	NaN
1	Agosto	0.545945	0.504500	NaN
2	Septiembre	0.519305	0.567546	0.557976
3	Octubre	0.582396	0.610116	0.671257
4	Noviembre	0.749347	0.746152	0.784268
5	Diciembre	0.776888	0.708397	0.726577
6	Enero	0.673387	0.780578	0.762428
7	Febrero	0.610206	0.702457	0.825008
8	Marzo	0.615066	0.608496	0.636186

Calculamos el error absoluto medio para los tres diferentes pronósticos y de esta manera poder basar nuestro análisis en esta métrica comparativa.

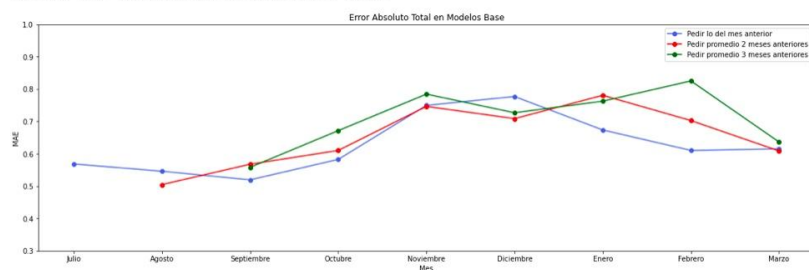
Es una medida de la diferencia entre dos variables, sirve para cuantificar la precisión de la técnica de medición comparando los valores pronosticados frente a las observaciones reales.

Se aplicará el modelo con el error absoluto medio (MAE) más pequeño.

Graficamos los tres modelos. Para nuestro análisis respecto al MAE, concluimos que la línea azul la cual representa al promedio móvil con 1 mes es el indicado para los pronósticos.

```
In [38]: # GRÁFICA
plt.figure(figsize=(20,6)) #impresión de la figura
plt.plot(mae.Mes, mae.mae_pedir_anterior, '-o', color="royalblue")
plt.plot(mae.Mes, mae.mae_promedio_2_meses_anteriores, '-o', color="red")
plt.plot(mae.Mes, mae.mae_promedio_3_meses_anteriores, '-o', color="darkgreen")
plt.legend(['Pedir lo del mes anterior', 'Pedir promedio 2 meses anteriores', 'Pedir promedio 3 meses anteriores'])
plt.ylim(0.3, 1)
plt.ylabel('MAE')
plt.xlabel('Mes')
plt.title('Error Absoluto Total en Modelos Base')
```

```
Out[38]: Text(0.5, 1.0, 'Error Absoluto Total en Modelos Base')
```

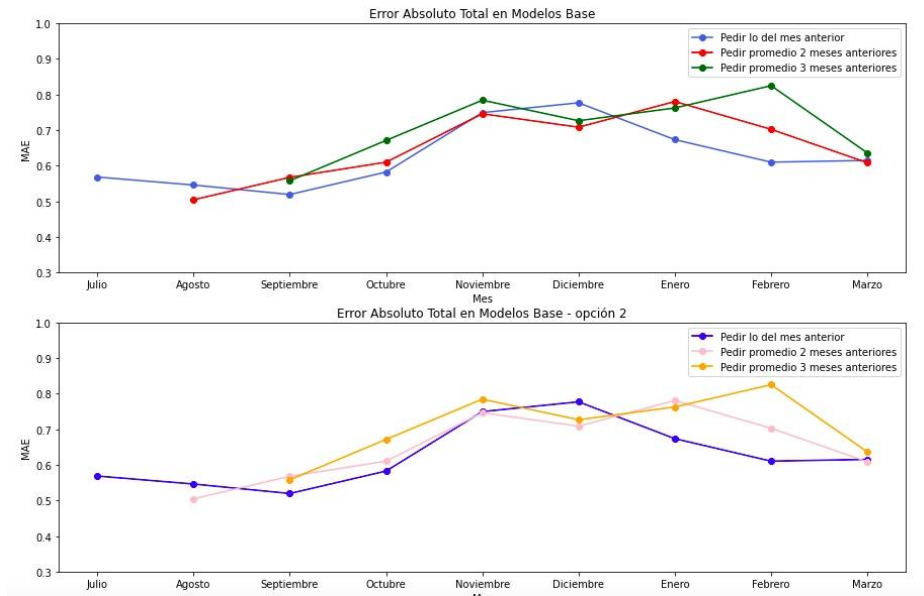


Respecto a las gráficas de error absoluto medio, hicimos los gráficos del *Modelo Base* y el *Modelo Base - opción 2*. En ambos casos se obtuvieron los mismos resultados donde el modelo de pronósticos a utilizar es el que el subgrupo está dado solo por el dato del mes anterior.

```
In [40]: plt.figure(figsize=(15,10))

plt.subplot(2, 1, 1)
plt.plot(mae.Mes, mae.mae_pedir_anterior, '-o', color="royalblue")
plt.plot(mae.Mes, mae.mae_promedio_2_meses_anteriores, '-o', color="red")
plt.plot(mae.Mes, mae.mae_promedio_3_meses_anteriores, '-o', color="darkgreen")
plt.legend(['Pedir lo del mes anterior', 'Pedir promedio 2 meses anteriores', 'Pedir promedio 3 meses anteriores'])
plt.ylim(0.3, 1)
plt.ylabel('MAE')
plt.xlabel('Mes')
plt.title('Error Absoluto Total en Modelos Base')

plt.subplot(2, 1, 2)
plt.plot(mae.Mes, mae.mae_pedir_anterior, '-o', color="blue")
plt.plot(mae.Mes, mae.mae_promedio_2_meses_anteriores, '-o', color="pink")
plt.plot(mae.Mes, mae.mae_promedio_3_meses_anteriores, '-o', color="orange")
plt.legend(['Pedir lo del mes anterior', 'Pedir promedio 2 meses anteriores', 'Pedir promedio 3 meses anteriores'])
plt.ylim(0.3, 1)
plt.ylabel('MAE')
plt.xlabel('Mes')
plt.title('Error Absoluto Total en Modelos Base - opción 2')
```



Conclusiones

Gracias a la correcta realización de las etapas anteriores del proyecto fuimos capaces de aplicar un modelo de promedios móviles para la predicción de los datos y evaluarlo por medio de herramientas de programación. Es muy importante darle seguimiento y realizar una correcta limpieza de datos y definición de la pregunta central para conocer el problema que queremos resolver.

Referencias

- DESCONOCIDO. (-). Introduction to Time Series Analysis. Engineering Statistics Handbook, de Octubre 2021 Sitio web: <https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4.htm>
- SUPPORT MINITAB. (2019). ¿Qué es un promedio móvil? Octubre 2021, de MINITAB Sitio web: <https://support.minitab.com/es-mx/minitab/18/help-and-how-to/modeling-statistics/time-series/supporting-topics/moving-average/what-is-a-moving-average/>

- Tokens24 2018. (2018). ¿Qué son los promedios móviles y cómo se utilizan en el comercio?. OCTUBRE 2021, de TOKENS 24 Sitio web:
<https://www.tokens24.com/es/cryptopedia/trading/que-son-los-promedios-moviles-y-como-se-utilizan-en-el-comercio>
- SUPPORT MINITAB. (2019). Interpretar todos los estadísticos y gráficas para Promedio móvil. Octubre 2021, de MINITAB Sitio web:
<https://support.minitab.com/es-mx/minitab/18/help-and-how-to/modeling-statistics/time-series/how-to/moving-average/interpret-the-results/all-statistics-and-graphs/>