

PROYECCIÓN DE VENTAS DE LOCALES



Versión 1.0

Autor:
Antonio Elvira García

CONTEXTO DEL PROYECTO



FG Burgers es una multinacional de comida rápida con origen en USA con más de 3000 locales distribuidos por todo el mundo, entre los países en los que tiene operación se encuentran UK, Francia, Alemania y España donde la marca cuenta con 22 locales en operación en 5 Comunidades Autónomas distintas (Comunidad de Madrid, Cataluña, Comunidad Valenciana, Andalucía y País Vasco).

En España la marca opera desde el año 2015.

El sector de la comida rápida es un sector muy competitivo solo en España existen más de 3000 locales de las diferentes marcas.

Si bien el sector de la comida rápida tiene generalmente unos altos ratios de facturación por local, es un sector donde la rentabilidad es bastante ajustada debido a los altos costos de inversión y de operación de los locales y se exige unos altos grados de gestión de los costos para así conseguir los niveles de rentabilidad esperados de los locales.

Parte esencial de esa gestión y de la consecución de los objetivos de rentabilidad viene dada por la optimizada proyección de ventas de los locales ya que los principales costos como son la mano de obra y coste de materia prima se ajustan en función de la proyección de ventas realizada.

OBJETIVOS

La marca FG Burgers buscaba una herramienta de proyección de ventas para poder implantar en sus locales en España.

Por lo tanto el objetivo de la investigación es realizar un modelo de proyección de ventas para poder predecir las ventas a futuro de los locales de la marca y también entender el comportamiento de estos en función de sus distintas variables.

Dentro de los objetivos de la investigación está resolver las siguientes preguntas;

01

¿Qué tipo de formato proporciona mayor facturación?

02

¿Qué Comunidad Autónoma es la que genera mayor facturación?

03

¿Afecta la época del año a las ventas?
¿Hay patrones de estacionalidad?

04

¿Hay relación entre la venta y el periodo del mes?

05

¿Afecta a las ventas que el día sea laborable o que haya algún festivo?

En base a estas preguntas se podrá definir un modelo de predicción de ventas que ayude a la marca a ajustar sus proyecciones de la manera más exacta para adaptar mejor la operación a la demanda de los locales.



FUENTE DEL DATASET Y CRITERIOS DE SELECCIÓN

Los datasets fueron conformados con datos provenientes de distintas fuentes de información de la empresa como son el programa de facturación de ventas de los locales y la base de datos de los locales.

Se decidió tomar estos Dataset porque contienen distinta información relevante de los locales actualmente abiertos por la marca para ser usada a la hora de realizar proyecciones de ventas.

Existen dos Dataset en formato .CSV uno para las ventas de los locales y otro para los datos generales de los locales.

VENTAS

	A	B	C	D	E	F	G	H
1	Local	Nombre_Loc	Fecha	Dia_Semana	Ventas	Clientes	Laborable	Festivo
2	1	Gran Via	31/03/2022	4	6856	476	1	0
3	2	Parquesur	31/03/2022	4	6449	413	1	0
4	3	Plaza Rio	31/03/2022	4	5599	385	1	0
5	4	La Gavia	31/03/2022	4	7110	468	1	0
6	5	La Maquinist	31/03/2022	4	6356	450	1	0
7	6	Granada	31/03/2022	4	6495	415	1	0
8	7	Plaza Mayor	31/03/2022	4	5436	368	1	0
9	8	Castellana	31/03/2022	4	6423	406	1	0
10	9	Sagrada Fam	31/03/2022	4	6973	437	1	0
11	10	Serrano	31/03/2022	4	6199	392	1	0
12	11	Diagonal Ma	31/03/2022	4	7546	500	1	0
13	12	Plaza Cataluñ	31/03/2022	4	6810	476	1	0

STORE

	A	B	C	D
1	Local	Nombre_Loc	Tipo_Local	Comunidad
2	1	Gran Via	In Store	1
3	2	Parquesur	Mall	1
4	3	Plaza Rio	Mall	1
5	4	La Gavia	Mall	1
6	5	La Maquinist	Mall	2
7	6	Granada	Mall	3
8	7	Plaza Mayor	In Store	1
9	8	Castellana	In Store	1
10	9	Sagrada Fam	In Store	2
11	10	Serrano	In Store	1

EXPLICACIÓN DE LOS DATOS DEL DATASET

VENTAS

1. Local: Es el Id de cada local
2. Nombre_Local: Es el nombre interno de cada local
3. Fecha: La fecha en la que se realizó la venta
4. Dia_Semana: Número de día de la semana relacionado con la fecha;
 - 1 = Lunes
 - 2 = Martes
 - 3 = Miércoles
 - 4 = Jueves
 - 5 = Viernes
 - 6 = Sábado
 - 7 = Domingo
5. Ventas: Ventas netas expresadas en euros realizadas por el local en ese día.
6. Clientes: Cantidad de clientes atendidos en el local en ese día.
7. Laborable: Indica si la fecha fue laborable;
 - 1 = Laborable
 - 0 = No laborable

Se consideran como no laborables a criterio de la marca los días festivos y los Sábados y Domingos.

1. Festivo: Indica si la fecha fue festivo.
 - 1 = Festivo
 - 0 = No Festivo

Los festivos tomados son los festivos nacionales de España

STORE

1. Local: Es el Id de cada local
2. Nombre_Local: Es el nombre interno de cada local
3. Tipo_Local: Es el tipo de local, puede ser de dos tipos;
 - In Store: Local ubicado a pie de calle
 - Mall: Local ubicado dentro de un Centro Comercial
4. Comunidad: Indica a que Comunidad Autónoma (región) pertenece cada local;
 - 1 = Comunidad de Madrid
 - 2 = Cataluña
 - 3 = Andalucía
 - 4 = País Vasco
 - 5 = Comunidad Valenciana

SELECCIÓN DEL ALGORITMO

DEBIDO A LA FLEXIBILIDAD Y AL TIPO DE DATOS CON ALTA ESTACIONALIDAD DENTRO DE LAS SERIES DE TIEMPO EL ALGORITMO SELECCIONADO FUE PROPHET.

PROPHET

SOBRE PROPHET

Prophet es un procedimiento para pronosticar datos de series temporales basados en un modelo aditivo donde las tendencias no lineales se ajustan a la estacionalidad anual, semanal y diaria, además de los efectos de las vacaciones. Funciona mejor con series temporales que tienen fuertes efectos estacionales y varias temporadas de datos históricos. Prophet es robusto a los datos faltantes y los cambios en la tendencia, y generalmente maneja bien los valores atípicos.

Prophet es un software de código abierto lanzado por el equipo de Core Data Science de Facebook. Está disponible para su descarga en CRAN y PyPI.

CARACTERÍSTICAS DE PROPHET

PROPHET

Rápido y preciso

Prophet se utiliza en muchas aplicaciones de Facebook para producir pronósticos confiables para la planificación y el establecimiento de objetivos. Hemos encontrado que funciona mejor que cualquier otro enfoque en la mayoría de los casos. Ajustamos modelos en Stan para que obtenga pronósticos en solo unos segundos.

Totalmente automático

Obtenga un pronóstico razonable sobre datos desordenados sin esfuerzo manual. Prophet es robusto a los valores atípicos, los datos faltantes y los cambios dramáticos en sus series temporales.

Previsiones ajustables

El procedimiento prophet incluye muchas posibilidades para que los usuarios modifiquen y ajusten los pronósticos. Puede usar parámetros interpretables por humanos para mejorar su pronóstico agregando su conocimiento del dominio y el negocio al que se está aplicando.

Disponible en R y Python

Los desarrolladores han implementado el procedimiento Prophet en R y Python, pero comparten el mismo código Stan subyacente para el ajuste. Por lo que se puede utilizar cualquier lenguaje de programación con el que te sientas cómodo para obtener pronósticos.

<https://facebook.github.io/prophet/>

DATA ADQUISITION Y DATA WRANGLING



- Al definir la misión, la visión y los objetivos, es mejor hacerlo de forma directa y concisa.
- La misión debe ser describir claramente tu propósito y lo que deseas que logre tu empresa.
- Usar palabras clave es de suma importancia en este punto, ya que puede ayudarte a resumir los objetivos y a destacar el enfoque.
- Al resumir la visión, siempre debes hacerlo en un tono positivo.
- Esto te permite explicar el concepto que tienes en mente y cómo se puede beneficiar la empresa a partir de esto.
- Por último, al establecer los objetivos, no olvides incluir un cronograma. Esto vincula la misión y la visión, y resume lo que deseas lograr.

A medida que el proyecto avanza, notarás que no se siguen al pie de la letra todas las cosas que anotas en el punto de partida. En esta etapa, planifica cómo gestionarás los cambios al alcance del proyecto, al presupuesto o al cronograma con un plan de gestión de cambios.

Esto asegura que cualquier cambio propuesto pueda pasar por un proceso de revisión y aprobación sistemático por parte de todos los involucrados antes de su implementación. Asegúrate de definir quién gestionará y evaluará las solicitudes de modificaciones, y de acordar un cronograma para la aplicación de los cambios solicitados. Una vez que un cambio se aprueba, el gestor de proyecto debe incorporar las actualizaciones en el documento de referencia.

DEFINICIÓN DE LA VARIANTE TARGET

El principal objetivo del proyecto era poder crear un modelo para predecir las ventas de los locales de la marca por lo que claramente nuestra variante Target a utilizar en este caso será la columna VENTAS del dataset ventas.

1 df_ventas

	Local	Nombre_Local	Fecha	Dia_Semana	Ventas	Clientes	Laborable	Festivo
0	1	Gran Via	31/03/2022	4	6856	476	1	0
1	2	Parquesur	31/03/2022	4	6449	413	1	0
2	3	Plaza Rio	31/03/2022	4	5599	385	1	0
3	4	La Gavia	31/03/2022	4	7110	468	1	0
4	5	La Maquinista	31/03/2022	4	6356	450	1	0
...
17505	18	Diagonal	01/01/2020	3	9306	520	0	1
17506	19	Valencia	01/01/2020	3	8947	497	0	1
17507	20	Paseo de Gracia	01/01/2020	3	12417	700	0	1
17508	21	Xanadu	01/01/2020	3	11439	655	0	1
17509	22	La Vaguada	01/01/2020	3	12692	705	0	1

17510 rows x 8 columns

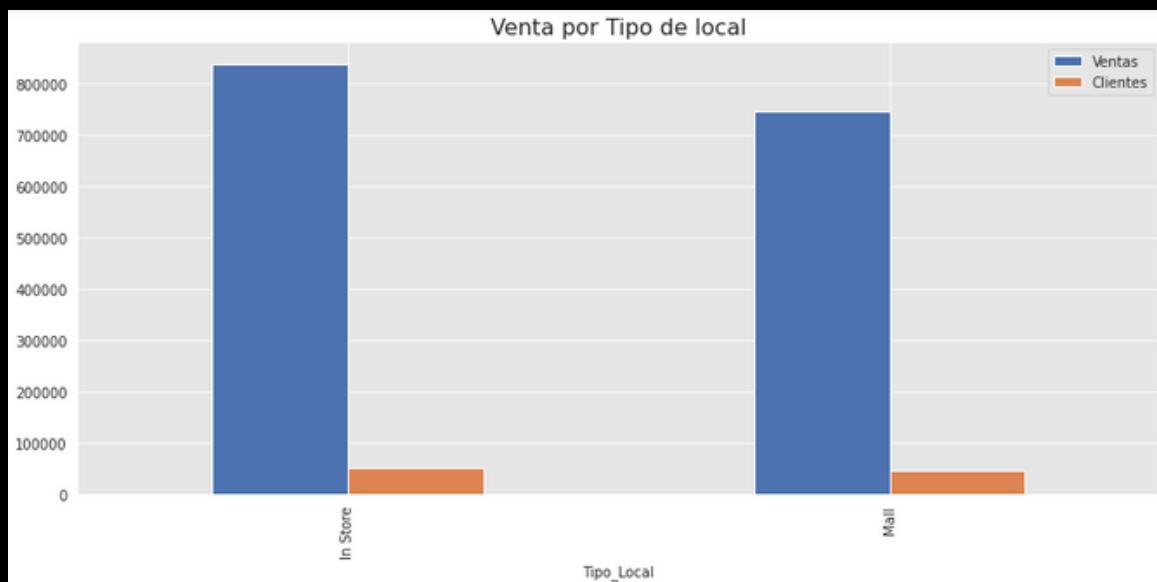
ANÁLISIS Y CONCLUSIONES INICIALES

Uno de los objetivos del proyecto era poder responder a una serie de preguntas relacionadas con las ventas de los locales, para responder a estas preguntas se realizaron distintos tipos de análisis univariado, bivariado y multivariado de los datos disponibles.

01

¿Qué tipo de formato proporciona la mayor facturación?

Tipo_Local	Ventas	Clientes
In Store	839752.3	52928.0
Mall	748218.5	47139.1



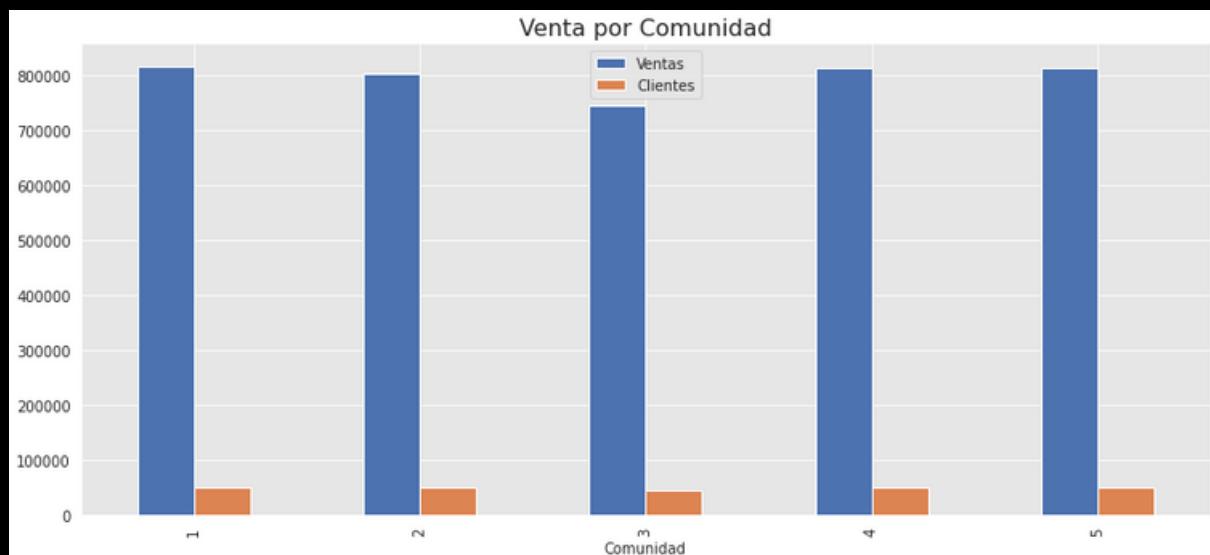
Podemos observar que si bien no hay una gran diferencia entre ambos formatos si se puede afirmar que el promedio de ventas y clientes de los locales del tipo In Store son mayores que las de los locales del tipo Mall.

ANÁLISIS Y CONCLUSIONES INICIALES

02

¿Qué Comunidad Autónoma es la que genera mayor facturación?

Comunidad	Ventas	Clients
1	815192.0	51368.2
2	801462.0	50512.9
3	744566.6	46946.6
4	811352.7	51016.0
5	812078.4	51156.2



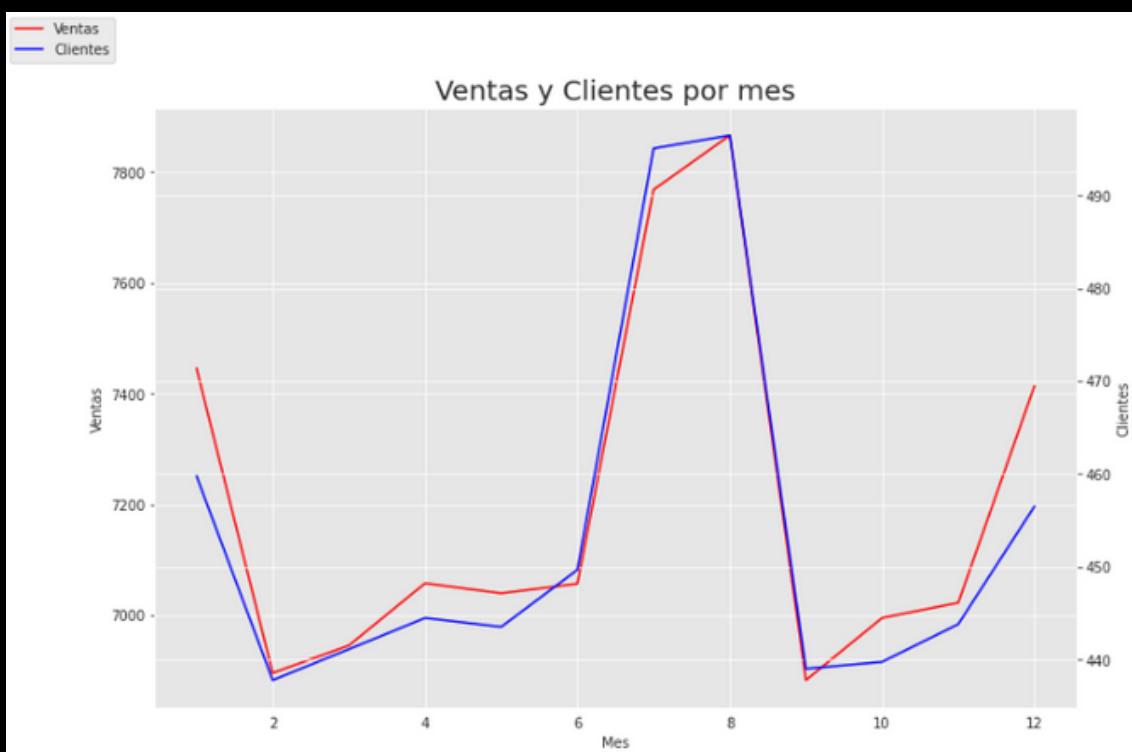
Podemos observar que no hay casi relación entre la Comunidad Autonómica donde estan situados los locales y el nivel de facturación y clientes que tienen los mismos.

ANÁLISIS Y CONCLUSIONES INICIALES

03

¿Afecta la época del año a las ventas?

¿Hay patrones de estacionalidad?



- Podemos observar que hay un incremento de clientes y ventas durante los periodos de junio a septiembre y noviembre a enero
- Estos periodos coinciden con el verano y las Navidades, períodos en los que los consumidores tienen más vacaciones

Por lo que podemos afirmar que efectivamente si afecta la época del año y que si hay un patrón de estacionalidad

ANÁLISIS Y CONCLUSIONES INICIALES

04

¿Hay relación entre la venta y el periodo del mes?



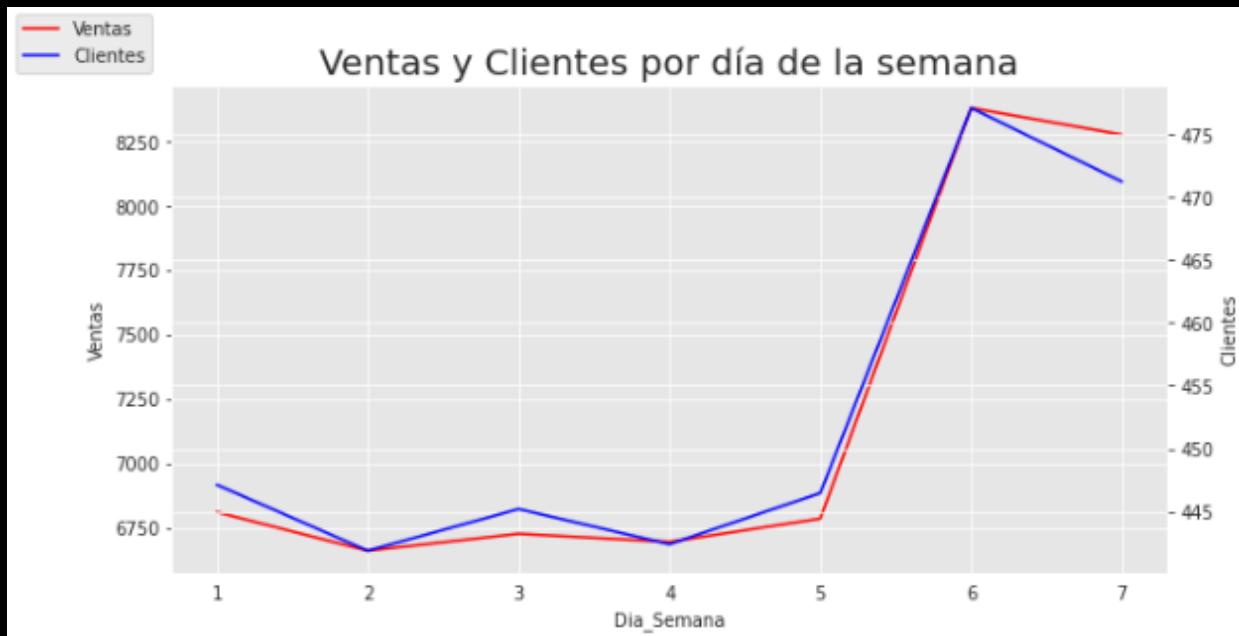
- Podemos observar que hay un incremento de clientes y ventas durante la última y la primera semana del mes.
- Estas semanas suelen coincidir con las fechas de cobro de los salarios por lo que esto puede aumentar al nivel de afluencia de los locales

Por lo que podemos afirmar que efectivamente si afecta el día del mes en el que nos encontramos al nivel de ventas y clientes.

ANÁLISIS Y CONCLUSIONES INICIALES

05

¿Afecta a las ventas que el día sea laborable o que haya algún festivo?



- Podemos observar que hay un incremento de clientes y ventas durante el fin de semana (de viernes a domingo) siendo el día más fuerte el sábado
- Estos días coinciden con los días no laborables

Por lo que podemos afirmar que efectivamente si hay una relación directa entre las ventas y clientes y si el día es laborable o no.

PARAMETRIZACIÓN Y CREACIÓN DEL MODELO

Tras el análisis de los datos y tal y como indicamos con anterioridad se decide elegir para este proyecto el método Prophet ya que claramente los datos tienen una gran estacionalidad, antes de la creación del modelo hay que realizar ciertas transformaciones y adecuar los datos a los parámetros marcados en la documentación.

El primer paso es crear un dataframe llamado 'holidays' este dataframe es necesario para que el modelo tenga en cuenta la estacionalidad durante los días que son festivos o en los que hay vacaciones, como en nuestro dataset ya teníamos indicado que en la columna 'Festivos', los días que tenían valor = 1 eran festivos, simplemente adaptamos esto a lo requerido por Prophet.



```
1 holidays = pd.DataFrame({'ds':df[df.Festivo==1].Fecha, 'holiday':'Festivo'})  
2 holidays
```

	ds	holiday
84	2022-01-06	Festivo
89	2022-01-01	Festivo
96	2021-12-25	Festivo
113	2021-12-08	Festivo
115	2021-12-06	Festivo
...
17409	2020-04-10	Festivo
17410	2020-04-09	Festivo
17431	2020-03-19	Festivo
17504	2020-01-06	Festivo
17509	2020-01-01	Festivo
601 rows × 2 columns		

PARAMETRIZACIÓN Y CREACIÓN DEL MODELO

El siguiente paso es crear un dataframe con la variable a predecir (target) y con las series de tiempo, en este caso añadimos la variable laborable que nos ayudará a ajustar de manera más eficiente el modelo.

Es importante saber que en Prophet la variable a predecir (target) siempre tiene que tener como nombre 'Y' y que las series de tiempo siempre deben ser llamadas 'ds'.

```
[61] 1 store_df = df[['Fecha', 'Ventas', 'Laborable']].rename(columns={'Fecha':'ds', 'Ventas':'y'})
```

	ds	y	Laborable
0	2022-03-31	8290	1
1	2022-03-30	6958	1
2	2022-03-29	8099	1
3	2022-03-28	8542	1
4	2022-03-27	8917	0
...
17505	2020-01-05	8887	0
17506	2020-01-04	11305	0
17507	2020-01-03	7924	1
17508	2020-01-02	9285	1
17509	2020-01-01	13580	0
17510 rows × 3 columns			

PARAMETRIZACIÓN Y CREACIÓN DEL MODELO

Creamos el modelo, para ello definimos una función con toda la parametrización del modelo y así poder elegir la proyección para cada tipo de local.

```
1 def forecast_sales(Local, df, periods):
2     print(f'Sales forecast for Store nº{Local}')
3     store_df = df[df['Local']==Local]
4     store_df = store_df[['Fecha', 'Ventas', 'Laborable']].rename(columns={'Fecha':'ds', 'Ventas':'y'})
5     model= Prophet(
6         yearly_seasonality=True,
7         weekly_seasonality=True,
8         daily_seasonality=False,
9         holidays = holidays,
10        interval_width=1,
11        mcmc_samples = 0,
12        growth='linear',
13        changepoint_prior_scale = 0.1,
14        seasonality_prior_scale = 0.1)
15     model.fit(store_df)
16     future = model.make_future_dataframe(periods=periods)
17     forecast = model.predict(df=future)
18
19 #figure1 = model.plot(forecast, xlabel='date', ylabel='sales')
20 #plt.title(f'Sales forecast for Store nº{store_id}')
21
22 return model, forecast
```

RESULTADOS DEL MODELO

Tras indicarle al modelo sobre que local queremos realizar la predicción de ventas y la cantidad de periodos a predecir, el modelo nos va a retornar un dataframe con distintas variables de tendencia etc. para ver la predicción de ventas nos vamos a quedar con las variables 'ds' que son las series de tiempo y la variable 'yhat' que es la predicción de ventas.

```
[66] 1 model, forecast = forecast_sales(12, df, periods=21)
      2 forecast.tail(21)
```

```
1 forecast.tail(21)[['ds', 'yhat']]
```

	ds	yhat
821	2022-04-01	7344.836135
822	2022-04-02	9132.329351
823	2022-04-03	9177.233543
824	2022-04-04	7260.074380
825	2022-04-05	7208.888895
826	2022-04-06	7181.499432
827	2022-04-07	7211.020844
828	2022-04-08	7352.716609
829	2022-04-09	9167.463624
830	2022-04-10	9238.738068
831	2022-04-11	7346.412515
832	2022-04-12	7317.910445
833	2022-04-13	7310.497983

RESULTADOS DEL MODELO

Podemos hacer un ploteo interactivo con nuestras predicciones de ventas realizadas con el modelo.

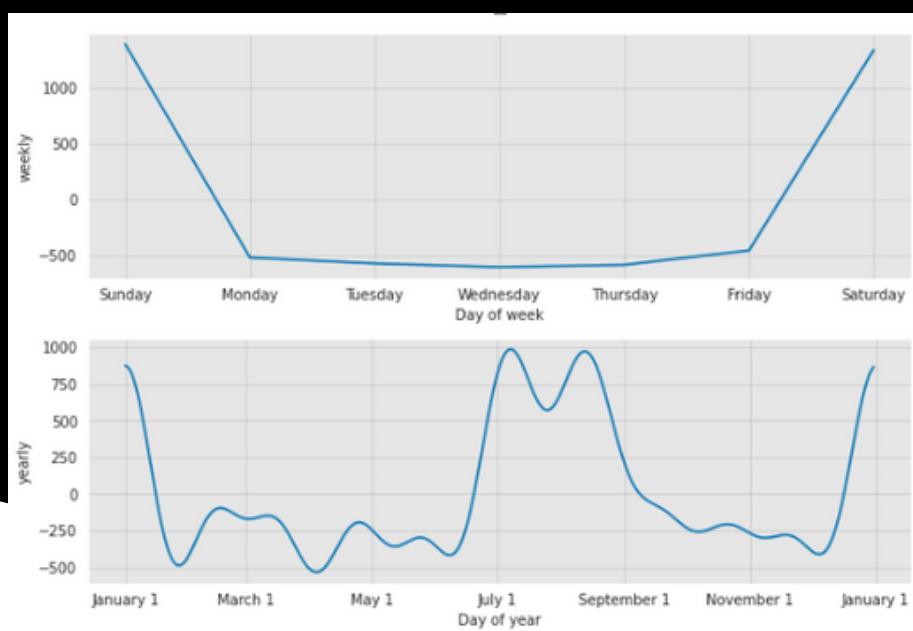
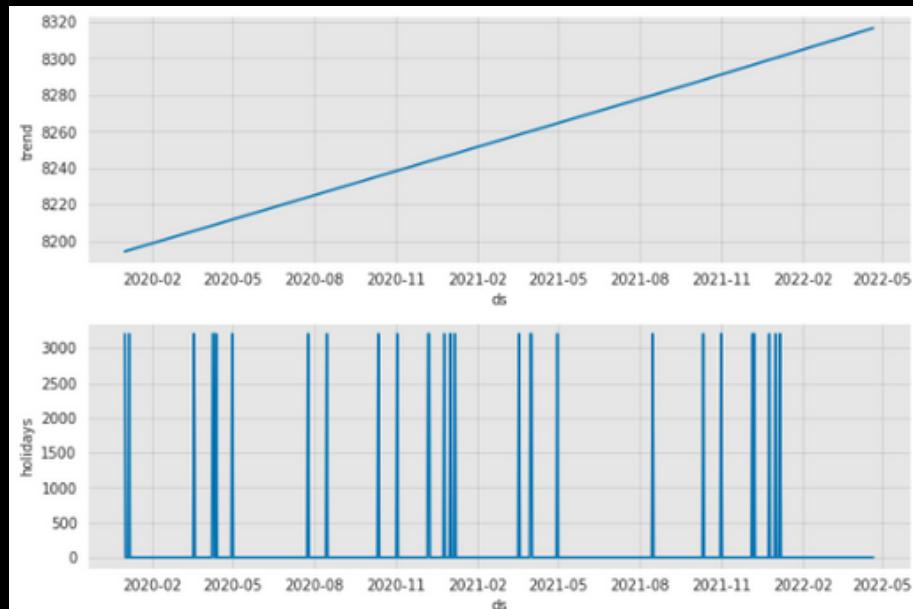


RESULTADOS DEL MODELO

Mediante el metodo plot.components podemos ver si nuestro modelo tiene estacionalidad y como se comportan los valores proyectados.



```
1 figure2= model.plot_components(forecast)
```

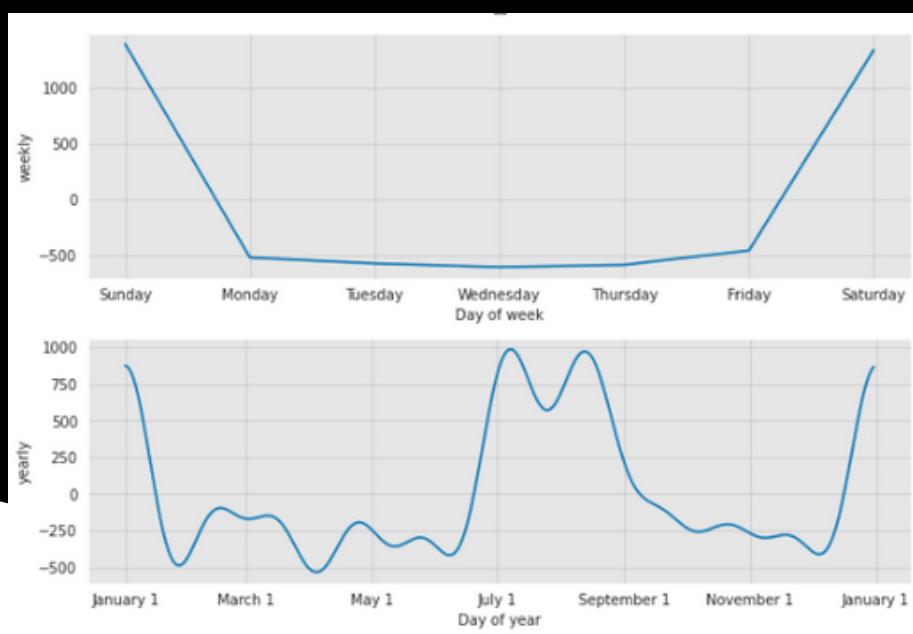
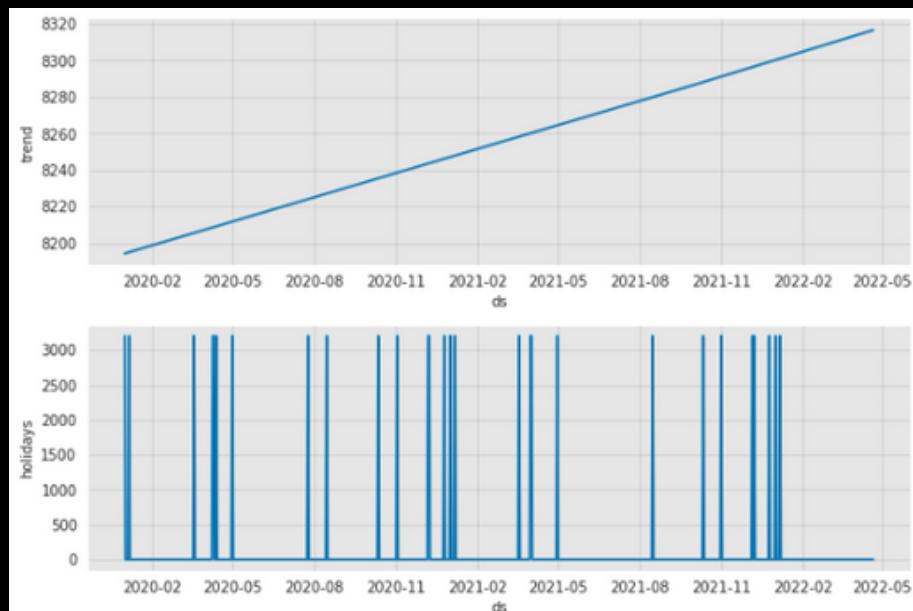


RESULTADOS DEL MODELO

Mediante el metodo plot.components podemos ver si nuestro modelo tiene estacionalidad y como se comportan los valores proyectados.



```
1 figure2= model.plot_components(forecast)
```



VALIDACIÓN DEL MODELO

Para la validación del modelo utilizamos el método cross validation del módulo diagnostics de Prophet, para esto definimos una función con los parámetros que queremos utilizar para la validación.

```
1 from fbprophet.diagnostics import cross_validation
2
3 df_cv = cross_validation(
4     model=model,
5     initial='540 days',
6     period='90 days',
7     horizon = '180 days'
8 )
9
10 df_cv
```

Esto nos genera un dataframe en el que vamos a tener 'y' que es la venta real y la variable 'yhat' que es la predicción realizada por el modelo.

	ds	yhat	yhat_lower	yhat_upper	y	cutoff
0	2021-07-05	8696.915650	4688.216674	11548.624410	9392	2021-07-04
1	2021-07-06	8598.081228	5668.690030	11356.981991	9990	2021-07-04
2	2021-07-07	8593.763544	5307.888841	11411.864339	9357	2021-07-04
3	2021-07-08	8617.667265	5735.158045	11093.021622	9030	2021-07-04
4	2021-07-09	8849.016958	5828.754064	11951.165686	9485	2021-07-04
...
355	2022-03-27	9239.033981	6608.369205	12451.291999	9253	2021-10-02
356	2022-03-28	7327.781826	4261.256987	10349.842248	6464	2021-10-02
357	2022-03-29	7254.962660	4639.809169	9790.439575	6528	2021-10-02
358	2022-03-30	7148.608996	4216.203993	9764.477237	8169	2021-10-02
359	2022-03-31	7188.841515	4289.688223	10138.496417	7926	2021-10-02

Podemos observar que el modelo realiza una predicción bastante certera de las ventas.

VALIDACIÓN DEL MODELO

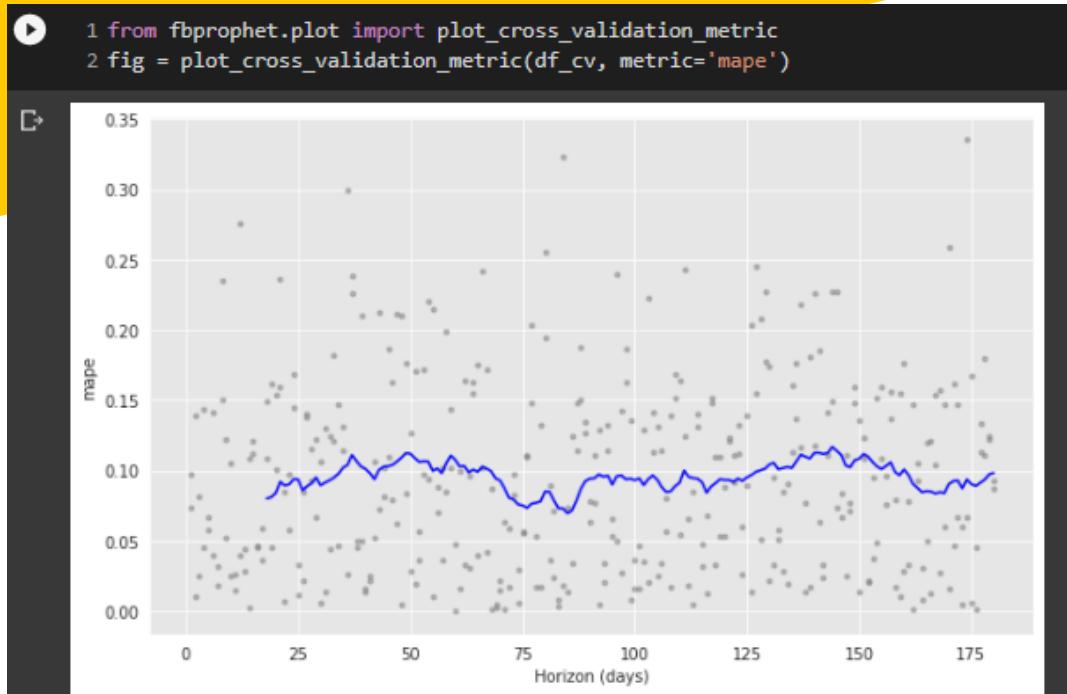
Observemos ahora las métricas del performance del modelo para ver si realmente se ajusta a lo que queremos y si es preciso.

	horizon	mse	rmse	mae	mape	mdape	coverage
0	18 days	6.870550e+05	828.887824	669.726037	0.080075	0.058364	1.000000
1	19 days	6.875874e+05	829.208921	666.142052	0.081069	0.054975	1.000000
2	20 days	6.999398e+05	836.624067	685.874822	0.083993	0.058364	1.000000
3	21 days	8.296082e+05	910.828278	748.947051	0.092047	0.062810	1.000000
4	22 days	7.986374e+05	893.665138	726.238258	0.089335	0.062810	1.000000
...
158	176 days	9.127280e+05	955.367976	706.348489	0.088977	0.066849	0.972222
159	177 days	9.371011e+05	968.039826	730.045324	0.091230	0.072496	0.972222
160	178 days	1.008640e+06	1004.310836	762.477077	0.093631	0.085038	0.972222
161	179 days	1.061253e+06	1030.171446	796.024663	0.097430	0.104708	0.972222
162	180 days	1.048674e+06	1024.048062	801.795569	0.098310	0.098783	0.972222

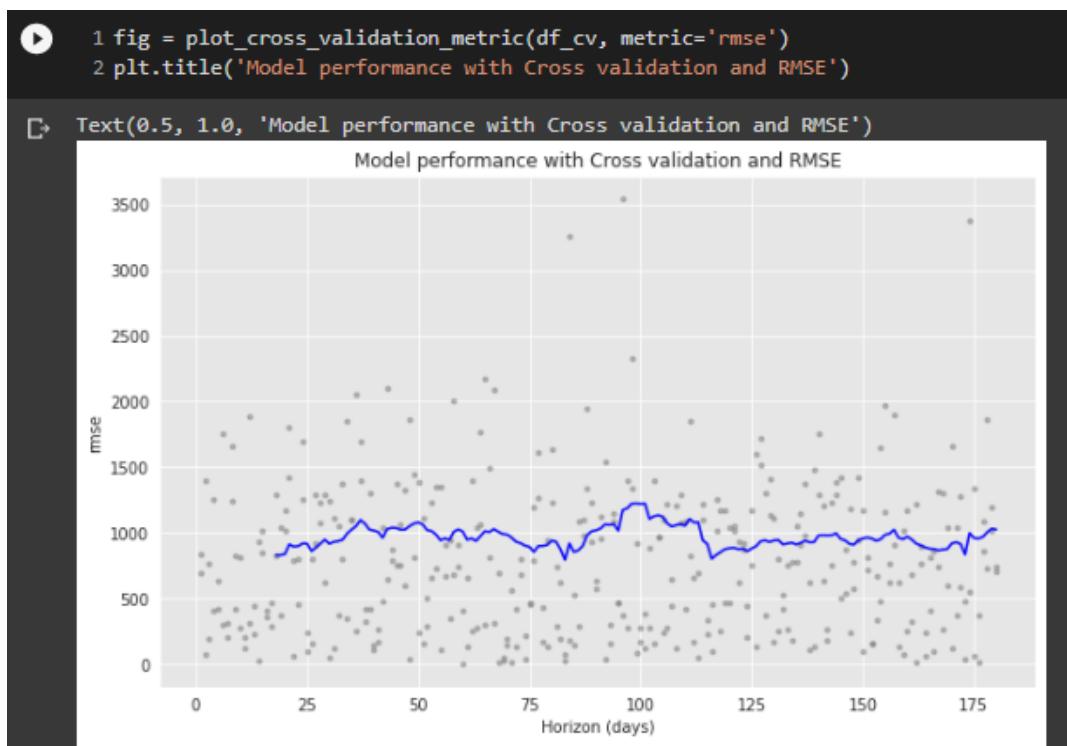
Podemos observar que el modelo es más preciso en horizontes de tiempos más cercano y que pierde algo de efectividad en horizontes de proyección más lejanos.

VALIDACIÓN DEL MODELO

Observemos el mape del modelo como evoluciona con el horizonte de proyección de 180 días que utilizamos.



Grafiquemos ahora el RMSE en el mismo horizonte para ver como se comporta el modelo.



LÍDER DEL PROYECTO

ANTONIO ELVIRA GARCÍA

Tengo 35 años soy español y actualmente resido entre España y Argentina y me desempeño como gerente de expansión de la marca Mostaza, teniendo la responsabilidad la apertura de nuevos locales de la marca en distintos países. También a lo largo de los años trabajé en otras empresas del sector como FIVE GUYS y McDonald's.

Mi formación principal fue en administración y operacional pero actualmente he incursionado en el mundo del análisis de datos lo que me lleva a cursar la carrera de Data Scientist en CoderHouse conocimientos que pretendo aplicar a mis responsabilidades actuales.

