1. OBJETIVO

Describir los pasos realizados para lograr una aproximación a futuro de los posibles valores de demanda de energía activa en la zona de Gálvez, Santa Fe a partir de los datos de consumo de la misma brindados desde 09/2018 al 02/2022.

2. ALCANCE

El siguiente informe está destinado a detallar el procedimiento empleado con Prophet respecto al análisis de datos correspondientes a una serie de tiempo, modelar los mismos, graficar sus principales características e intentar obtener un pronóstico sobre lo que puede ocurrir en el futuro.

3. GLOSARIO

4. Elección del modelo de predicción

De entre los varios modelos empleados para el análisis y modelado de series de tiempo (diferenciándose unos de otros según la manera que tienen de aproximarse mediantes distintos cálculos matemáticos y estadísticos) se pueden enumerar los siguientes:

- Holt-Winters.
- Prophet.
- SARIMA.
- LSTM.

De los mencionados, el modelo "Prophet" destaca por permitir al usuario configurar de manera manual parámetros como la frecuencia de estacionalidad, flexibilidad en la tendencia, ajuste en los días feriados y porcentaje de incertidumbre sin necesidad de tener una base de conocimiento fuerte en matemáticas, permitiendo también el ajuste automático de alguno de sus parámetros con solo ingresar los datos a modelar.

La función matemática que lo representa es la siguiente:

$$y(t) = g(t) + s(t) + h(t) + e(t)$$

- $g(t) : factor\ de\ tendencia$
- $s(t) : componente\ de\ estacionalidad$
- $h(t) : componente\ de\ feriados$
- $e(t) : componente\ de\ incertidumbre$

El uso del modelo es bastante directo: Se importan las librerías necesarias, se cargan los datos a un data-frame haciendo uso de "Pandas", se estudia la información y prepara (análisis de características, filtro de información, y gráficas) de la manera adecuada y se comienza con el modelado.

5. DESCRIPCIÓN DEL PROCEDIMIENTO

5.1 Fuente de datos

Se nos ha provisto de datos sobre el consumo de demanda activa de la ciudad de Gálvez en la provincia de Santa Fe desde las fechas 11/09/2018 – 13hs hasta 1/02/2022 – 16.15hs. La información está medida diariamente con mediciones realizadas cada 15 minutos.
La elección de los medidores trifásicos se realizó en base a presentar un movimiento estacionario y constante a lo largo de los años de medición, lo cual se constató con los gráficos de demanda en Mr. Dims:

- DIGA00032694
- DIGA00039133
- DIGA00039136
- DIPA00006955

La información brindada sobre el clima corresponde a las fechas 12/04/2021 – 8:43hs hasta 1/02/2022 – 16:58hs e incluye la temperatura del día medida con intervalos de aproximadamente 15'.

5.2 Lectura y limpieza de los datos

Se seleccionó el medidor DIGA00039136 para comenzar con el estudio de los datos.
Se levanta el archivo .csv y se le asignó a un dataframe (*objeto bi-dimensional conformado por filas y columnas sobre el que se puede seleccionar, agregar, eliminar, concatenar, y renombrar elementos*):

	terminal	fechahora	demanda_activa
0	DIGA00039136	2017-08-18 09:15:00.000	0
1	DIGA00039136	2018-09-11 13:00:00.000	12160
2	DIGA00039136	2018-09-11 13:15:00.000	36400
3	DIGA00039136	2018-09-11 13:30:00.000	37280
4	DIGA00039136	2018-09-11 13:45:00.000	37920

Fig 1: Primeras 5 filas del archivo.

El tamaño total de mediciones realizadas en el periodo de tiempo brindado es de 118743.

A las columnas "fechahora" y "demanda_activa" del dataframe se las renombre como "datetime" y "y[kW]" respectivamente por cuestiones de orden en el desarrollo, mientras que la columna "terminal" es eliminada por no aportar nada.

Se elimina la primer fila y, de existir, aquellos datos duplicados solo reteniendo la última medición.

Se convierte al dataframe en una serie de tiempo (para posterior análisis temporal) moviendo la columna "datetime" como index (filas). En el proceso se verifica que dicha serie de tiempo tenga una frecuencia establecida (en este caso de 15 minutos), en caso

de no ser así, se completa la línea de tiempo indexando las fechas y/u horarios faltantes e interpolando para completar con valores de energía activa.

	y[kW]
2018-09-11 13:00:00	12160.0
2018-09-11 13:15:00	36400.0
2018-09-11 13:30:00	37280.0
2018-09-11 13:45:00	37920.0
2018-09-11 14:00:00	36000.0

Fig 2: Primeras 5 filas de la serie de tiempo.

Antes de la siguiente instancia se aplica un filtro digital de suavización "Savitzky-Golay" sobre los datos a lo largo del tiempo, con el propósito de incrementar la precisión de los resultados sin alterar la tendencia.

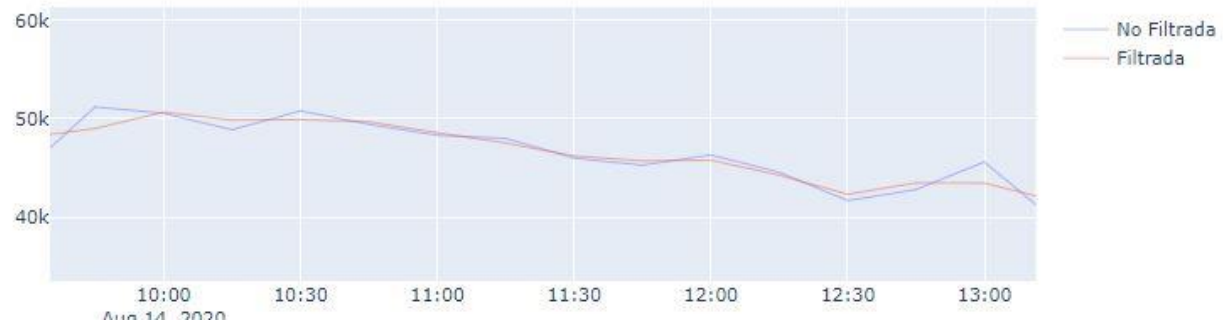

Fig 3: Filtro Savitzky-Golay

5.3 Análisis exploratorio y gráfico de los datos: EDA y VDA

La siguiente imagen corresponde al gráfico de la demanda en el tiempo:

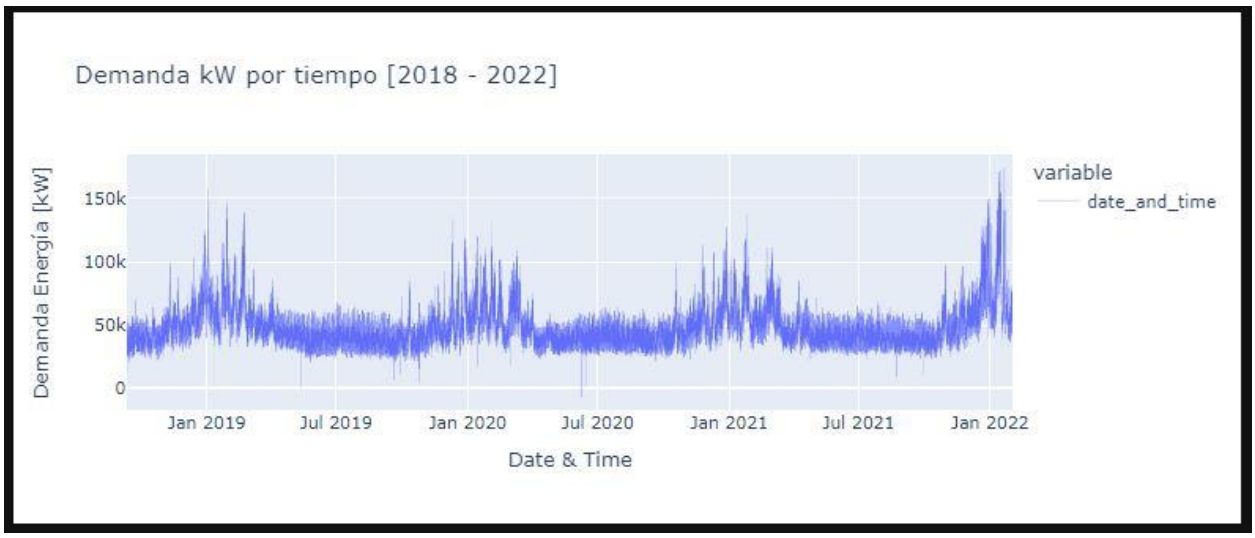

Fig 4: Energía vs Tiempo

En ella se puede observar un comportamiento estacional en los meses de primavera y verano con una marcada llanura en los meses de otoño e invierno.

En la siguiente imagen se representa mediante gráficas los patrones de demanda por año:

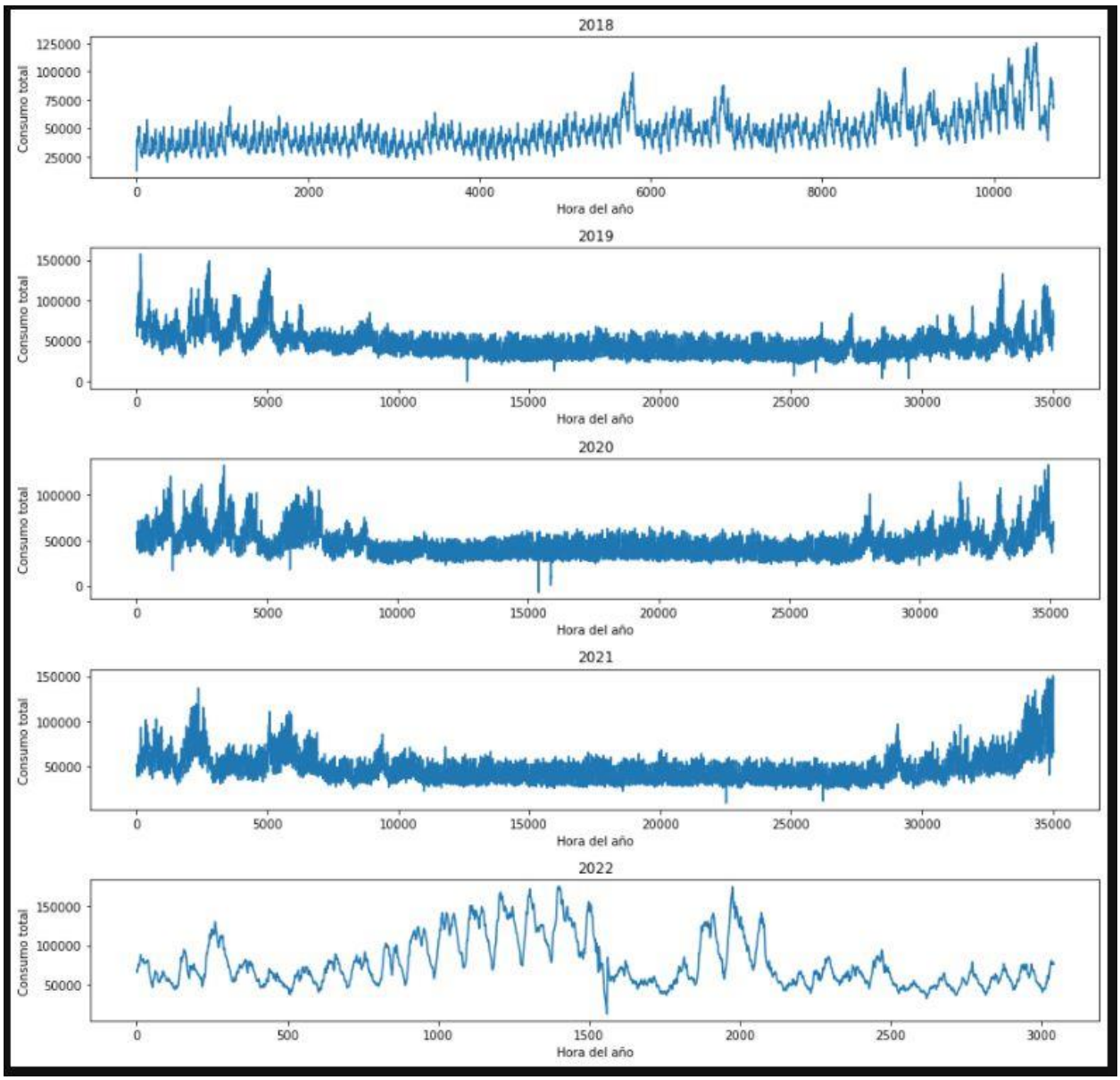

Fig 5: Energía vs Tiempo (anual)

La imagen 6 muestra un gráfico de tipo "Box-Plot" (usando los percentiles 25%, 50% mediana y 75%) que muestra que tan distribuidos (en cuanto a dispersión o densidad) están los datos respecto a los meses del año. Esto es útil porque de esta forma se puede observar una tendencia que ha de ser aprendida por el modelo para luego poder recrear valor futuros sin incertidumbre estacional.

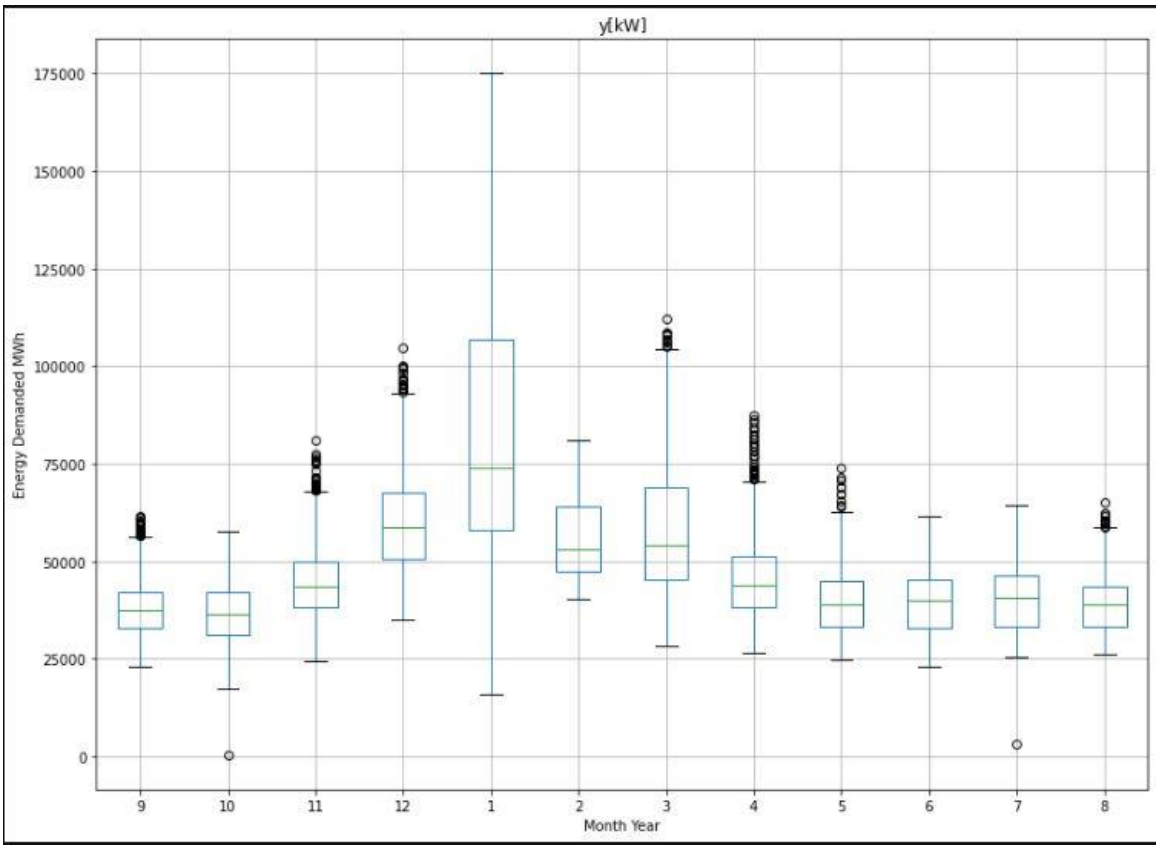

Fig 6: Box Plot respecto a los meses del año.

Se puede observar una gran cantidad de datos se acumulan en el mes de enero con bastante distribución.

Para conocer el comportamiento de los usuarios, se realizan dos gráficas con el uso de la mediana de la energía (porque a diferencia del promedio, esta no se ve tan afectada por factores externos a la medición o valores que están fuera del rango normal):

Fig 7: Mediana de consumo por hs por día semanal

El consumo por la mañana es bastante similar en los siete días pero luego comienza a notarse una clara diferencia entre el resto y el domingo que es el día de menor consumo por semana. Respecto al horario, el pico se da por la noche cuando las actividades en el hogar alcanzan su máximo.

Fig 8: Mediana de consumo por hs por estación.

Respecto a lo que ocurre en las estaciones del año, es clara la diferencia del verano respecto al resto. La mayor demanda ocurre en esa época y en las gráficas anteriores se puede constatar.

Como conclusión al análisis gráfico realizado se observa que las estaciones junto a los días de semana ocurren los patrones de comportamiento de los datos y por tanto es necesario reflejarlos como funciones que luego han de ser adheridas al modelo de predicción. También se pueden concluir sus comportamientos no solo estacionarios sino una tendencia constante a lo largo del tiempo, con comportamientos casi idénticos lo cual se traduce como variable "aditiva" (y no multiplicativa) del modelo y de comportamiento linear.

5.4 Aplicación del modelo

Realizado el análisis en el tiempo para comprender el comportamiento de los datos, se continúa con la partición de los datos para preparar los dataframe de "entrenamiento" y de "test".

La partición suele corresponder con un 20% a 25% para el test, pero esto varía según el proyecto. En este caso, el 25% parece encontrar un buen resultado.

```
Training shape: (88460, 13)
 Testing shape: (30498, 13)

Proporción del train-test: 25.64%

Las fechas de entrenamiento son: 2018-09-11 13:00:00 & 2021-03-20 23:45:00
Las fechas de test son: 2021-03-21 00:00:00 & 2022-02-01 16:15:00
```

Fig 9: Partición de los datos.

Esto gráficamente se refleja como sigue:

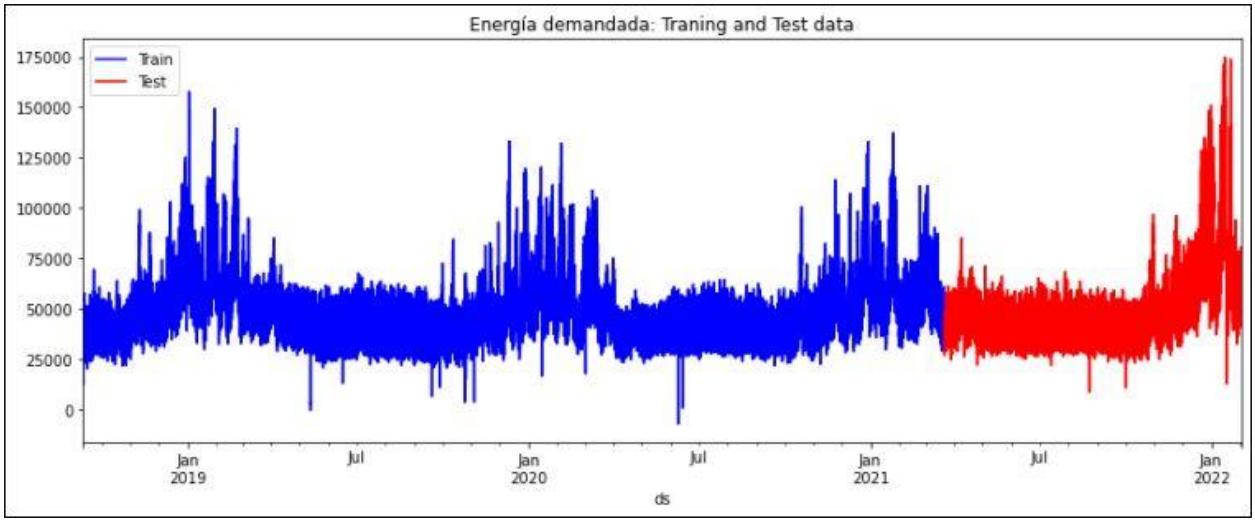

Fig 10: Test al 25.64%.

Para que Prophet haga su trabajo correctamente es necesario hacer unos cambios al dataframe que se había formado con anterioridad (train y test). Esto es porque es de requisito obligatorio que el dataframe a procesar contenga la columna "**ds**":"**datestamp**" correspondiente a las fechas en formato: YYYY-MM-DD HH:MM:SS, como también la columna "**y**" que ha de ser numérica y representa la variable a pronosticar.

	ds	y
0	2018-09-11 13:00:00	12523.428571
1	2018-09-11 13:15:00	34946.285714
2	2018-09-11 13:30:00	39460.571429
3	2018-09-11 13:45:00	37460.571429
4	2018-09-11 14:00:00	36240.000000

Fig 11: Primeras cinco filas del test_prophet

Recordando que la estacionalidad puede ser modelada por sus efectos semanales en las semanas de verano y sus patrones diarios de consumo, se deben modelar como funciones lo que se conoce como "*Condiciones Estacionales"*. Esto se refleja en la siguiente imagen:

	ds	y	is_spring	is_summer	is_autumn	\
0	2018-09-11 13:00:00	12523.428571	False	False	True	
1	2018-09-11 13:15:00	34946.285714	False	False	True	
2	2018-09-11 13:30:00	39460.571429	False	False	True	
3	2018-09-11 13:45:00	37460.571429	False	False	True	
4	2018-09-11 14:00:00	36240.000000	False	False	True	

	is_winter	is_weekend	is_weekday
0	False	False	True
1	False	False	True
2	False	False	True
3	False	False	True
4	False	False	True

Fig 12: Condiciones estacionales añadidas a "train" y "test".

Con los datos de entrenamiento preparados, se genera una instancia de la clase "Prophet", se cargan los datos a entrenar y se gráfica el resultado.

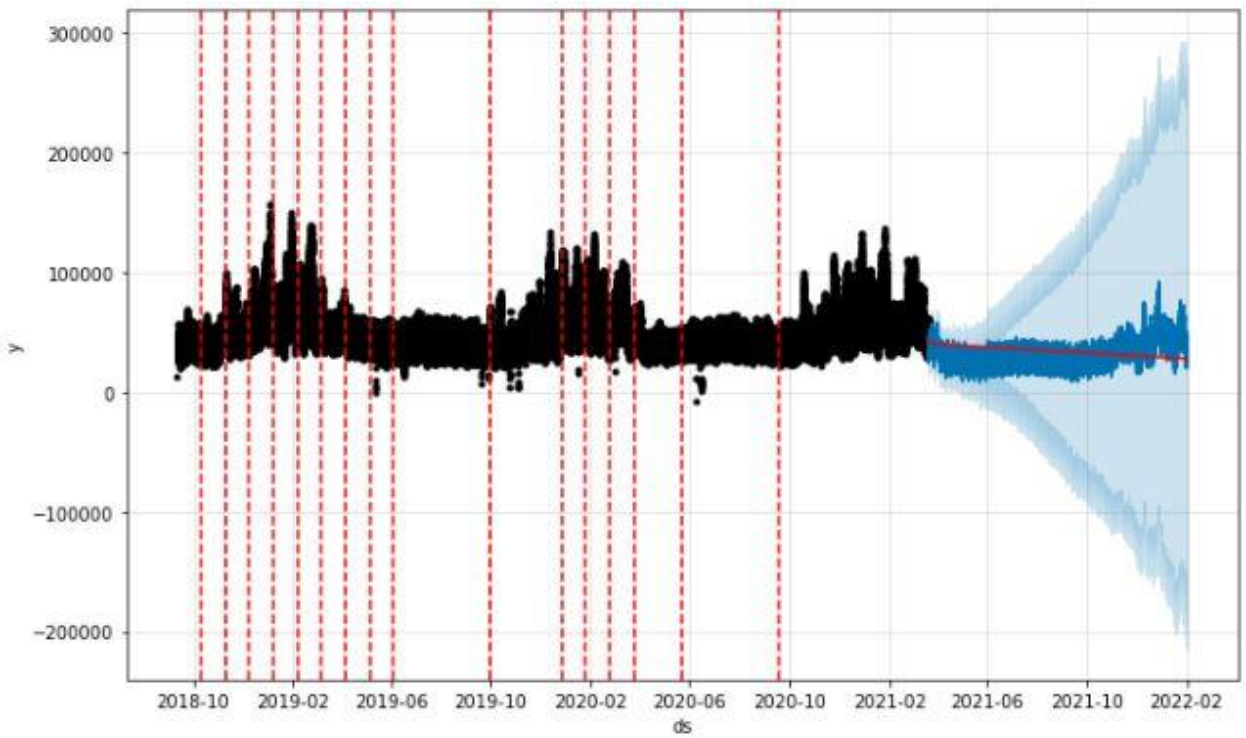

Fig 13: Modelo de predicción sin ajustes.

En la imagen se pueden apreciar los siguientes elementos:

1) Las líneas rojas, indican en su conjunto, "changepoints" que el modelo si no se le especifica toma de manera natural 25 de ellos dentro del 80% del set de entrenamiento. Estos sirven para que el modelo comprenda el comportamiento de los datos.

2) En la parte azul de la gráfica se puede observar una fina línea roja de pendiente negativa como resultado de la tendencia que el programa cree que se ha de tener. Esto no corresponde con el movimiento del conjunto de datos.
3) La sombra azul corresponde a la "flexibilidad" que tiene la curva de tendencia a adaptarse a distintos puntos. Como puede observarse, en una primera aproximación tiene un amplio abanico y esto debe ajustarse para cerrarse.

Las gráficas de comparación entre el conjunto de datos para test y los valores predichos, puede verse a continuación:

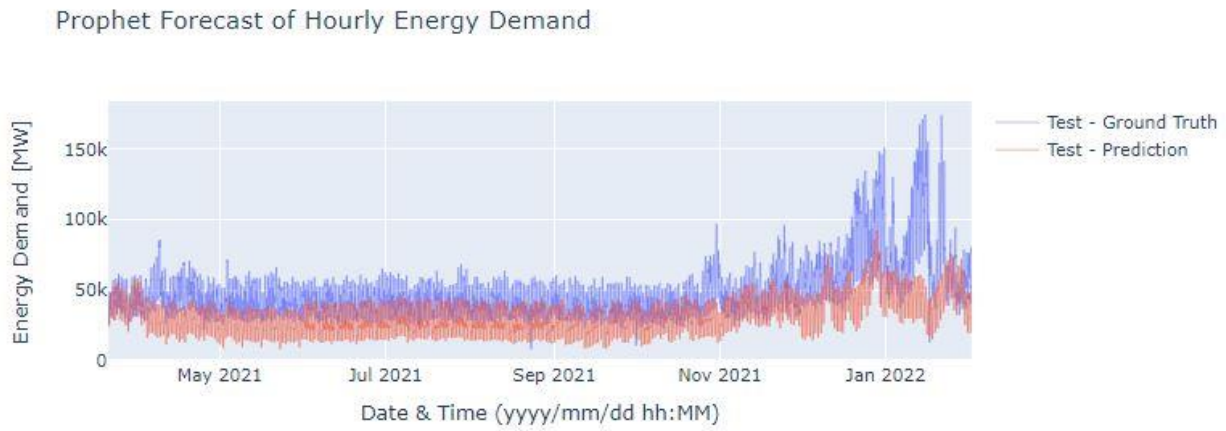

Fig 14: Comparasión Test vs Predicción (naranja)

El Error Porcentual Absoluto Medio (MAPE) que dicta que tan lejos está un punto de predicción respecto a su punto correspondiente de test fue: **MAPE 35%.**

En un primer acercamiento al modelo, sin ajuste de sus parámetros internos, el error es alto. Se puede observar que el movimiento del comportamiento es correspondido pero los niveles de energía obtenidos no son un reflejo correcto.

En lo que respecta a las primeras 168hs (primera semana) de comparación, se obtuvo el siguiente resultado:

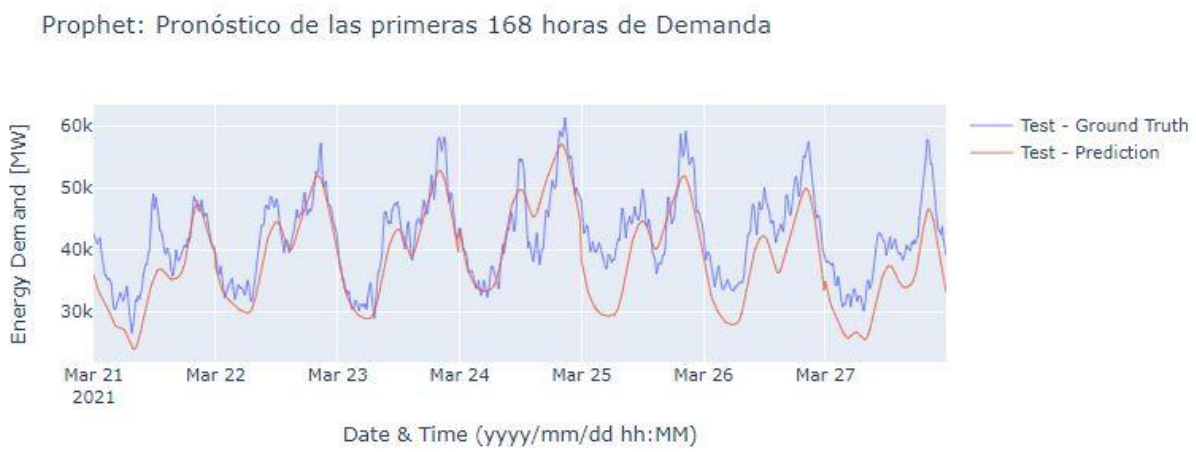

Fig 15: Primeras 168hs de predicción.

Con un **MAPE 10%.** El comportamiento es seguido, como así también como gran parte de los valores en energía, en consecuencia el error bajo.

En lo correspondiente a la última semana:

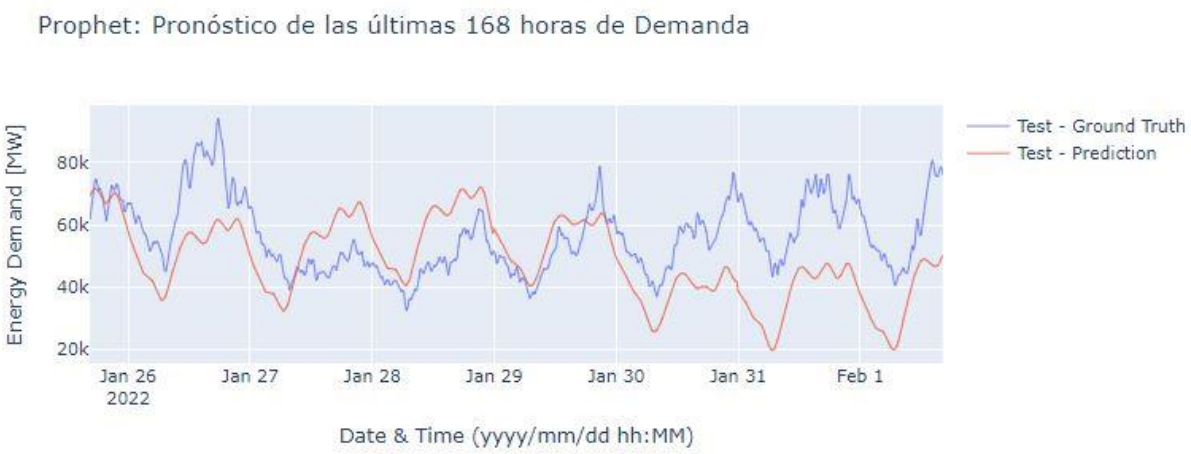

Fig 16: Últimas 168hs.

Con un **MAPE 25%.** Un error mayor respecto a lo visto en la primera semana, se refleja en la falta de seguimiento del comportamiento y valores de energías.

Las proyecciones de 168hs son realizadas con la intención de observar puntos que son críticos si se desea expandir la curva de tiempo para tiempos futuros donde aún no hay mediciones pero que se desean predecir.

5.5 Ajuste de parámetros

Vale la pena mencionar los siguientes parámetros que han sido ajustados:

- *'Growth' donde se indica el tipo de crecimiento lineal.*
- *'n_changepoints' indicando una mayor cantidad de puntos de cambios p/aprendizaje.*
- *'seasonality_mode' de tipo aditiva.*
- *'changepoint_prior_scale' para reducir la flexibilidad de la tendencia.*

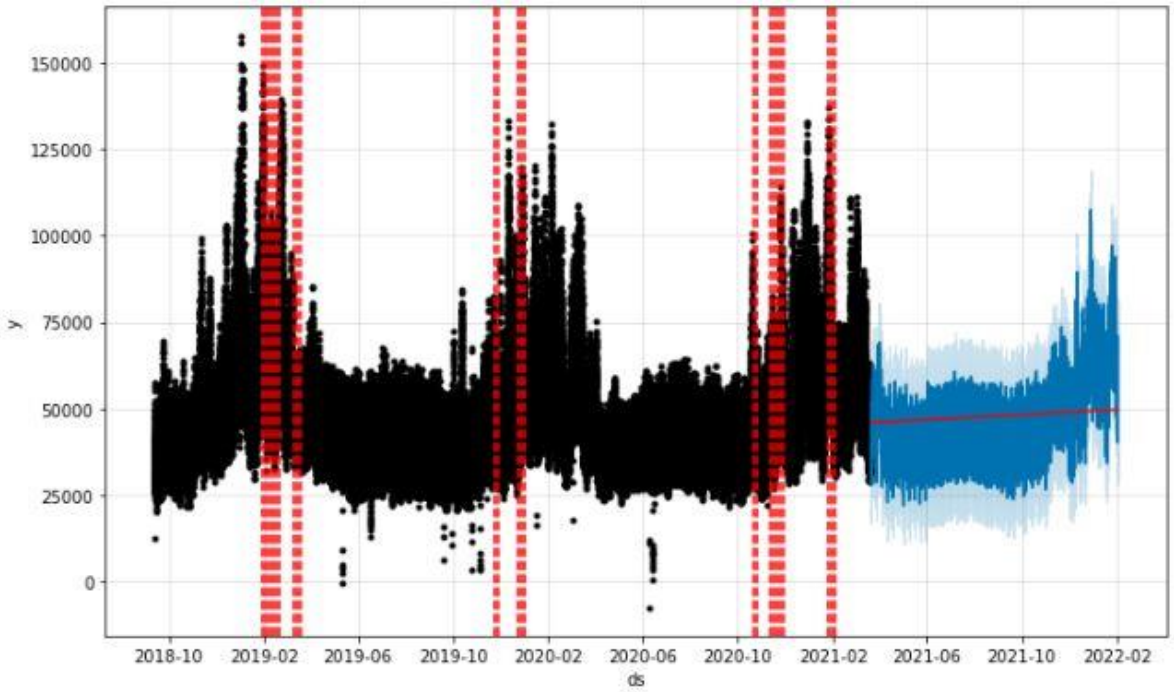

Fig 17: Señal resultante de ajustes.

En la imagen 17 se observan mayor cantidad de changepoints como una tendencia resultante positiva con menor flexibilidad y más adaptada a la curva de test.

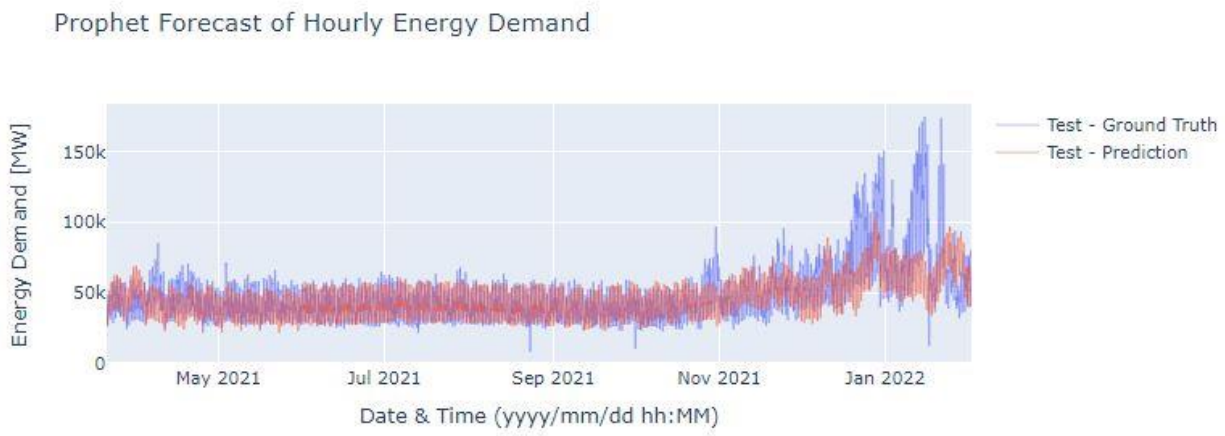

Fig 18: Resultados de predicción con ajustes.

Con **MAPE 13.52% (una reducción del 22%).**

Del resultado de la imagen, con los ajustes se logró una reducción del error de precisión, el movimiento de la curva predicha sigue bastante bien a la de test, pero falla en los picos de alto valor.

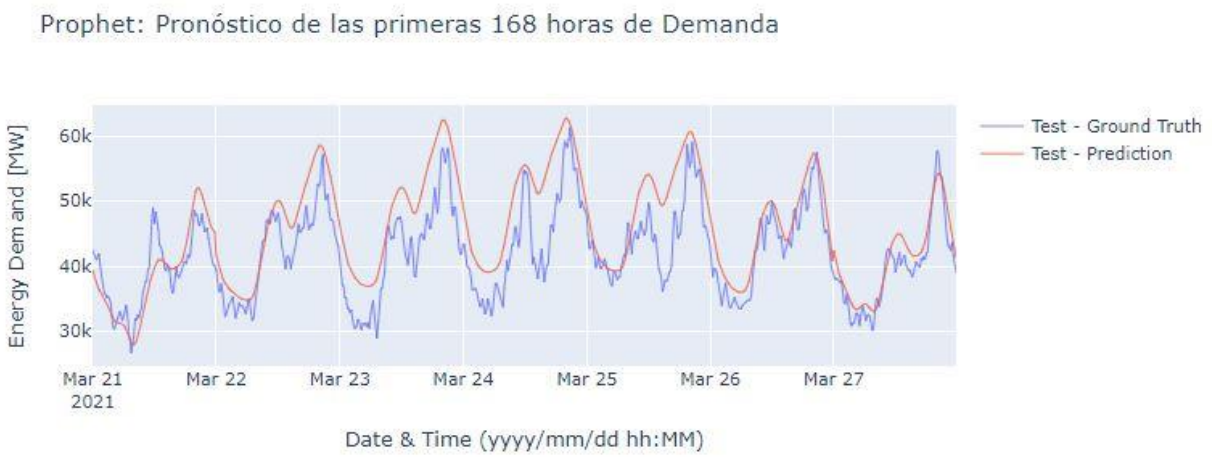

Fig 19: Primeras 168hs.

MAPE 10.11%. Si bien la diferencia de error con el primer resultado no es sustancial, el seguimiento que tiene la curva predicha es mucho mejor y se adapta a los niveles de cambio de la energía de una mejor manera.

Fig 20: Últimas 168hs.

MAPE 32.44%. El error respecto al primer resultado es mucho mayor como consecuencia de tener una curva más adaptada (lo que ajusto los valores y subio el nivel general) por ello se ve el manto de la curva predicha bastante más arriba que la curva de test.

5.6 Apartado clima

En lo que respecta al clima, los datos brindados corresponden desde 12/04/2018 – 13hs hasta 1/02/2022 – 16.15hs, lo cual hace a un total de 27213 datos totales de medición de clima, cuya curva es la siguiente representada:

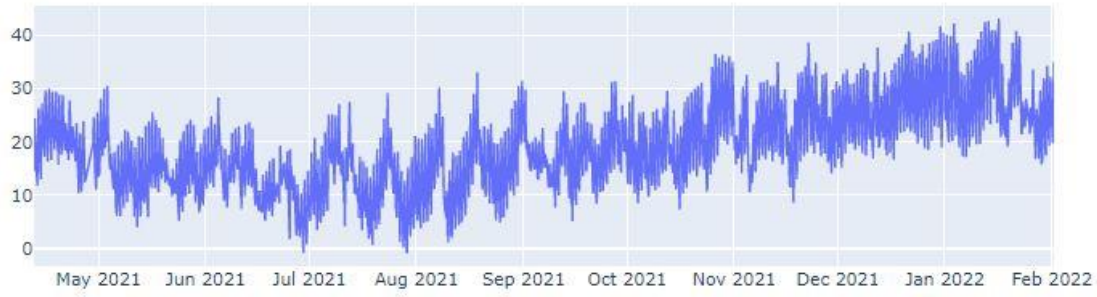

Fig 21: Curva de clima.

Teniendo en cuenta que de los datos de energía se tienen aproximadamente 4 años, esta cantidad debe ser recortada para igualar la cantidad de datos (porque es obligatorio que todas las variables a ser sometidas al modelo de predicción **deban tener la misma longitud digital**).

Por lo tanto la curva de energía correspondida a la misma línea en tiempo es:

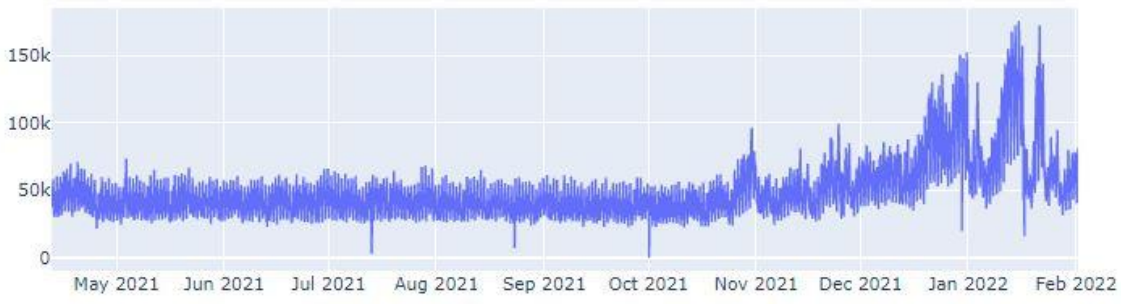

Fig 22: Curva de energía.

Si bien el modelo de predicción puede ser aplicado, con lo desarrollado anteriormente puede observarse que hay un faltante de comportamiento estacionario y como tratamiento de información de energía no refleja gran parte de sus condiciones.

Se unió ambos segmentos en un solo .csv con las columnas de tiempo, demanda y clima:

	datetime	y[kW]	temp
0	2021-04-12 08:45:00	42160	14.44
1	2021-04-12 09:00:00	37920	15.56
2	2021-04-12 09:15:00	43360	16.67
3	2021-04-12 09:30:00	42960	17.22
4	2021-04-12 09:45:00	49440	17.78

Fig 23: Set de datos (primeros cinco datos).

En lo que respecta al tratamiento de los datos es idéntico a la sección anterior que solo incluía demanda, solo que cambia el agregado de la temperatura.

Se genera un set de datos para entrenamiento:

- Para el presente caso, se decidió tomar por completo como entrenamiento al archivo .csv disponible para mitigar la falta de datos representativos de los cambios al recortar para generar un conjunto de datos que sirvan como test.

Recordando que Prophet necesita que existan las columnas "ds" e "y", el set de datos de entrenamientos (cinco primeras filas) es el siguiente:

	ds	y	temp	is_spring	is_summer	is_autumn	is_winter	is_weekend	is_weekday
0	2021-04-12 08:45:00	42160	14.44	True	False	False	False	False	True
1	2021-04-12 09:00:00	37920	15.56	True	False	False	False	False	True
2	2021-04-12 09:15:00	43360	16.67	True	False	False	False	False	True
3	2021-04-12 09:30:00	42960	17.22	True	False	False	False	False	True
4	2021-04-12 09:45:00	49440	17.78	True	False	False	False	False	True

Fig 24: Train_Prophet

Donde la columna de "temp" representa los datos de temperatura, y el resto de columnas marcan la diferencia temporal entre las estaciones del año y los días de la semana.

- La misma estructura de la figura 24 presenta el set de datos para test.

Los resultados del modelo con los parámetros en ajustes por defecto y el agregado del clima son los siguientes:

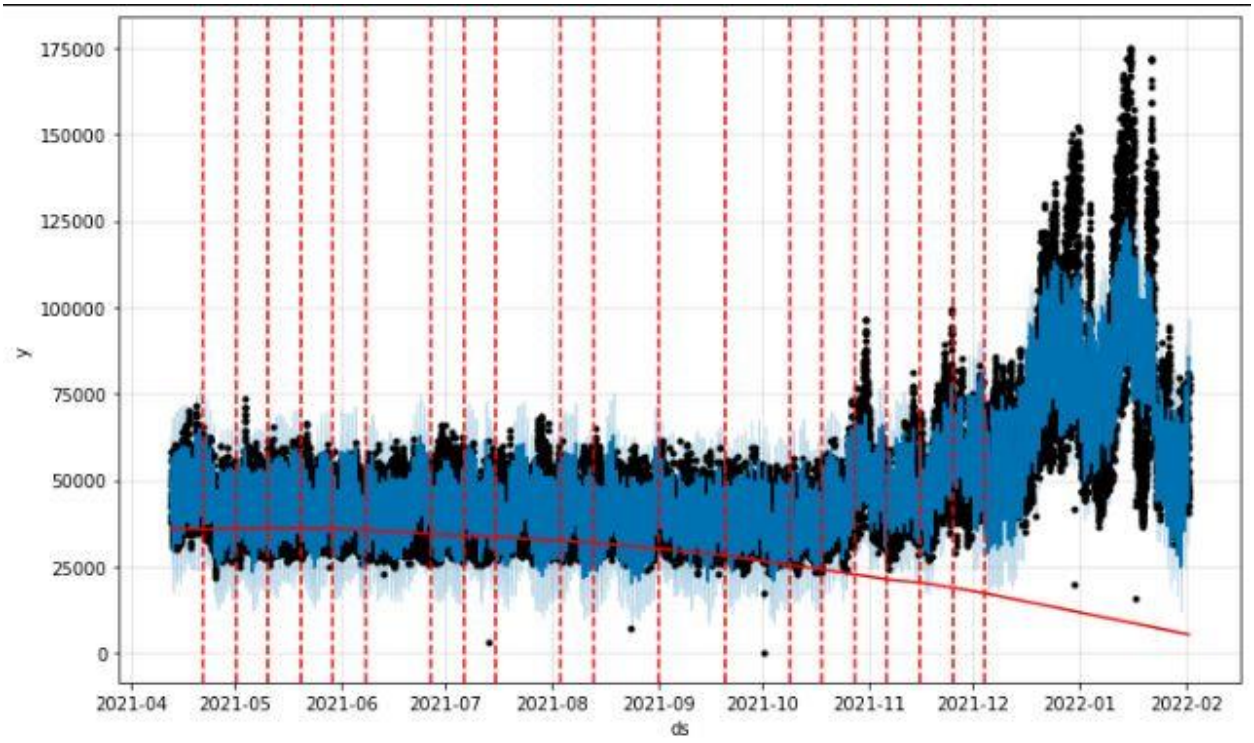

Fig 25: Resutlados por defecto.

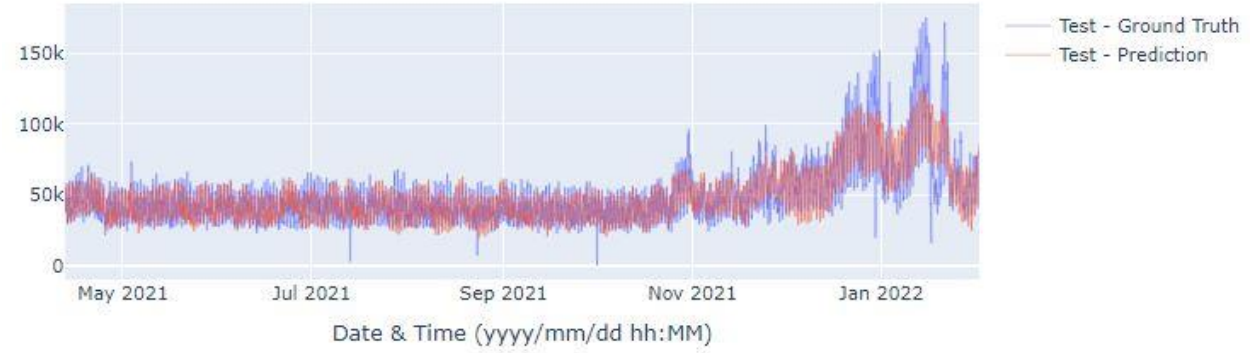

Fig 26: Comparación test-predicción.

MAPE 11.88%: En la figura 25 se puede observar en negro los datos reales siendo la sombra azul lo predicho. El error que se distingue es la tendencia (línea roja) la cual indica un decaimiento siendo que los datos no representan tal cosa. Los puntos de cambios (líneas rojas) no logran tomar la totalidad de los datos por lo que se pierde en el aprendizaje los picos de cambios en la última etapa, lo cual deriva en una tendencia negativa. En la figura 26 se representa lo mismo pero de manera más prolija.

	yhat	ds	y	temp
0	43237.340553	2021-04-12 08:45:00	42160	14.44
1	44955.207274	2021-04-12 09:00:00	37920	15.56
2	46586.491843	2021-04-12 09:15:00	43360	16.67
3	47568.194613	2021-04-12 09:30:00	42960	17.22
4	48446.867638	2021-04-12 09:45:00	49440	17.78
...	...	...	...	...
28322	85512.935721	2022-02-01 15:15:00	75040	34.28
28323	84902.225694	2022-02-01 15:30:00	76320	33.94
28324	85666.210417	2022-02-01 15:45:00	79440	35.02
28325	84836.387262	2022-02-01 16:00:00	77760	34.50
28326	84826.448060	2022-02-01 16:15:00	76320	34.80

Fig 27: Test-predicción tabular.

En la figura 27 se muestra en pantalla los resultados previos en los gráficos pero de manera numérica y tabular, donde "yhat" es el pronóstico.

Los resultados del programa con los parámetros ajustados al modelo de datos se muestran a continuación:

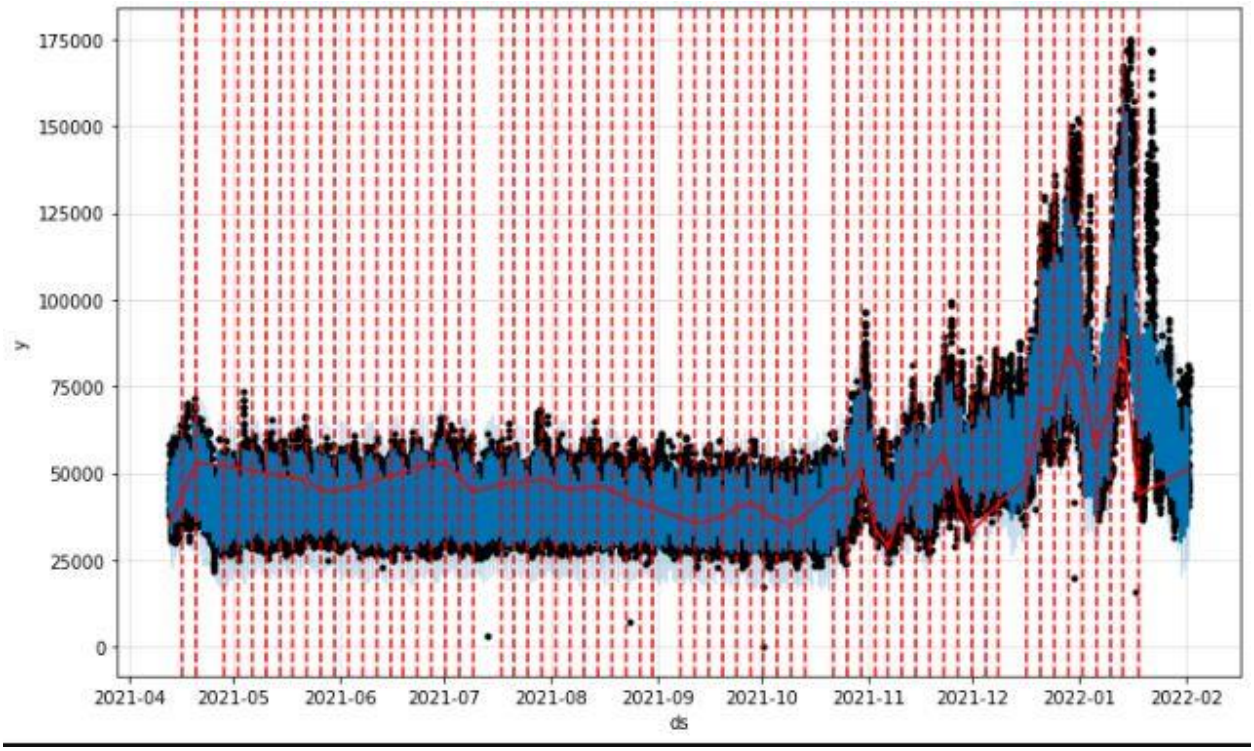

Fig 28: Resultados prophet con ajustes.

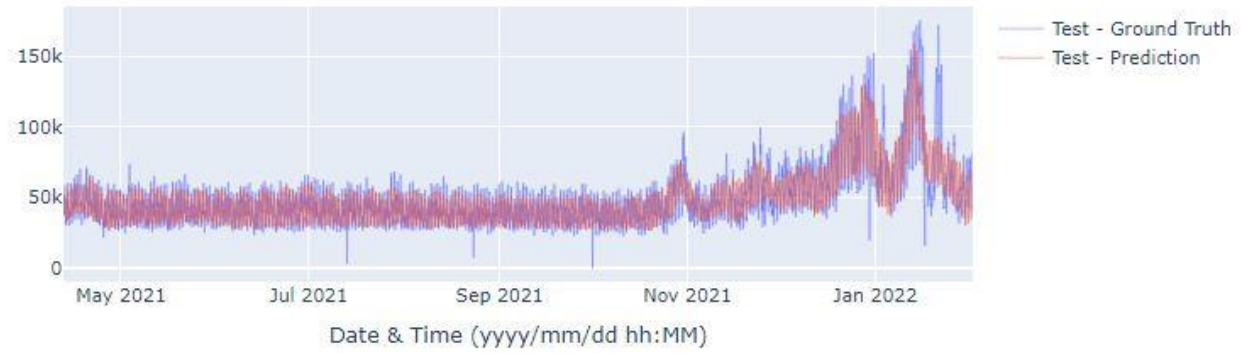

Fig 29: Comparación test - predicción con ajustes.

MAPE 9.1%: Se ajustó la cantidad de puntos de cambios y también su distribución sobre el conjunto de datos para que capture mejor los cambios. Como resultado se observa una mejor tendencia, más ajustada a los cambios reales.

En cuanto a los datos tabulares se puede observar que los valores pronosticados se acercan más a los valores reales por lo que el error es menor.

	yhat	ds	y	temp
0	39681.419462	2021-04-12 08:45:00	42160	14.44
1	40410.790085	2021-04-12 09:00:00	37920	15.56
2	41109.795330	2021-04-12 09:15:00	43360	16.67
3	41763.212430	2021-04-12 09:30:00	42960	17.22
4	42356.244678	2021-04-12 09:45:00	49440	17.78
...	...	...	...	...
28322	68423.443979	2022-02-01 15:15:00	75040	34.28
28323	68155.167269	2022-02-01 15:30:00	76320	33.94
28324	67830.750977	2022-02-01 15:45:00	79440	35.02
28325	67465.262493	2022-02-01 16:00:00	77760	34.50
28326	67073.124018	2022-02-01 16:15:00	76320	34.80

Fig 30: Datos tabulados.