

# Análisis de la Demanda de Energía Eléctrica

## *London Smart Meters*

Autores:  
-Panizza, Camila  
-Ron, Juan Ignacio  
-Sirai, Juan Bautista



# Contenido

---

<b>1- Resumen Ejecutivo.....</b>	<b>P.3</b>
<b>2- Datos Utilizados.....</b>	<b>P.5</b>
<b>3- Análisis Exploratorio.....</b>	<b>P.6</b>
<b>4- Problema 1: Estimación de Demanda Mediano / Largo Plazo.....</b>	<b>P.13</b>
<b>    4.1- Modelo Lineal.....</b>	<b>P.15</b>
<b>    4.2- Modelo con trasformación Logarítmica y Estacionalidad Mensual....</b>	<b>P.16</b>
<b>    4.4- Modelo con trasformación Logarítmica y Estacionalidad Mensual y Condiciones Climáticas.....</b>	<b>P.17</b>
<b>    4.5- Modelo ARIMA y análisis de Residuos.....</b>	<b>P.18</b>
<b>    4.6- Modelo con Datos de Panel.....</b>	<b>P.20</b>
<b>    4.7- Comparativa de modelos.....</b>	<b>P.22</b>
<b>5- Problema 2: Estimación de Demanda Puntual.....</b>	<b>P.23</b>
<b>    5.1- Selección de un subset de datos.....</b>	<b>P.24</b>
<b>    5.2- Data wrangling.....</b>	<b>P.25</b>
<b>    5.3- Evaluación de modelos de Machine Learning.....</b>	<b>P.26</b>
<b>    5.4- Escalabilidad del modelo.....</b>	<b>P.34</b>
<b>    5.5- Optimización: generación de un nuevo modelo.....</b>	<b>P.40</b>
<b>6- Conclusiones.....</b>	<b>P.45</b>



# Resumen Ejecutivo

Dentro del marco de un conjunto de acciones liderada por la Unión Europea para hacer un uso más eficiente de energía y combatir el cambio climático, se lanzó desde el gobierno del Reino Unido una iniciativa para que las prestatarias del servicio de distribución eléctrica instalen medidores inteligentes en todos los hogares de Inglaterra, Gales y Escocia. Los mismos permiten tener información precisa acerca del consumo diario de cada punto de suministro

Dentro del marco de este informe, utilizaremos la información provista por dichas mediciones para abordar las siguientes problemáticas:

## 1. ¿Se puede predecir la demanda global de energía en el mediano/largo plazo?

### Estimación de la demanda como serie temporal

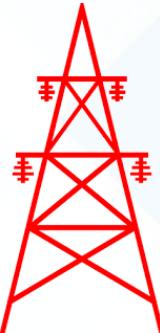


Las empresas prestatarias del servicio de energía eléctrica operan en el Mercado Eléctrico como Distribuidoras, esto quiere decir que compran energía a un precio determinado para luego venderla a los usuarios, contemplando un Valor Agregado de Distribución que deberá tanto cubrir los costos de operación, como asegurar una determinada rentabilidad. De allí se desprende que predecir la Demanda de Energía a mediano y largo plazo, es el primer paso que debe realizar cualquier Distribuidora en el proceso de planificación estratégica y elaboración de su plan de negocios.



# Resumen Ejecutivo

## 2. ¿Podemos estimar el consumo de un usuario o conjunto de usuarios en un momento dado?



### Estimación puntual de demanda de energía

Poder estimar cuánto consume determinado usuario o conjunto de usuarios en un momento dado del tiempo sería útil para evaluar inversiones en la red eléctrica, pero también permitiría encontrar un momento/lugar óptimos para la interrupción del servicio para realizar reparaciones, o alimentar otras zonas de mayor demanda. Incluso podría servir para evaluar potenciales casos de fraude, si un consumidor presenta en la realidad un consumo muy diferente al estimado para él o para los consumidores de perfil similar en un momento dado.

Para abordar este problema trabajamos con la misma base que utilizamos para el análisis por series de tiempo pero con un mayor grado de detalle: tomamos las mediciones de consumo cada una hora. Esto nos permitirá realizar una predicción más acotada en el tiempo y quizás más útil para la toma de decisiones del negocio.

Por su complejidad, dividimos al problema en dos partes:

Primero buscamos al mejor algoritmo para resolver el problema. Para ello, seleccionamos al azar una pequeña porción de los datos e implementamos distintos modelos de regresión que pudieran predecir el comportamiento de la variable objetivo: modelos lineales, modelos de k-vecinos y modelos de árbol. El modelo CatBoost, basado en árboles de decisión, fue el que tuvo la mejor performance en términos de R2, error y velocidad de procesamiento.

Luego de identificar el mejor algoritmo para estudiar el problema, analizamos si era posible entrenar un modelo que pudiera predecir con eficacia para la generalidad de los datos. En primer lugar aplicamos nuestro primer modelo Catboost, lo que nos dio buena performance en conjuntos de usuarios de perfil similar, pero muy pobre desempeño a nivel general.

Luego probamos distintas variantes para entrenar el modelo: tomar aleatoriamente un conjunto más amplio de datos, tomar una muestra estratificada por perfil socioeconómico, tomar la muestra estratificada e incorporar el perfil a las variables regresoras. Finalmente, el modelo que mejor performance logró fue aquel entrenado con una muestra estratificada por el clúster de consumo promedio diario y su variabilidad.

A partir de los resultados obtenidos, concluimos esta sección destacando que es posible generar un modelo a tales fines, y que las variables climáticas, pero sobre todo las del perfil de consumo tienen una importancia fundamental. Nuestros modelos son susceptibles de ser mejorados, pero ello a costa de una demanda mucho mayor de capacidad de cómputo. Fuera del alcance de este trabajo quedó la implementación de modelos en tecnologías de procesamiento distribuido, que podrían trabajar con todo el cuerpo de datos de una manera más dinámica y potencialmente lograr resultados aún mejores.



# Datos Utilizados

---

El primer paso para el análisis y modelado de datos, es la carga de los Datasets. Los mismos se encuentran disponibles de manera gratuita en sitios como Kaggle e incluso en el sitio oficial del gobierno londinense. A continuación, detallamos las principales características de las bases utilizadas:

## Dataset de Mediciones Diarias:

*Dimensiones iniciales: (3.510.433 ; 9 )*

En este conjunto de datos, encontraremos una versión refactorizada de los datos de Londres, que contiene las lecturas de consumo de energía de una muestra de 5.567 hogares londinenses que participaron en el proyecto Low Carbon London liderado por UK Power Networks entre noviembre de 2011 y febrero. 2014.

Los principales campos que utilizaremos son:

- **LCLid:** Identificador único del punto de medición.
- **Day:** día de la medición.
- **Energy\_sum:** total consumido en kWh en el día.

## Dataset Clima:

*Dimensiones iniciales: (882 ; 32 )*

Contiene datos referidos a las condiciones climáticas diarias para el período de tiempo de análisis:

- **temperature:** Temperatura
- **windBearing:** Dirección del viento.
- **dewPoint:** Punto de rocío
- **cloudCover:** Nubosidad
- **windSpeed:** Velocidad del viento
- **pressure:** Presión
- **apparentTemperature:** Sensación térmica
- **visibility:** Visibilidad
- **uvIndex:** índice UV
- **moonPhase:** Fase Lunar

## Datasets de Usuarios:

*Dimensiones iniciales: (5.566 ; 5)*

Contiene información que permite caracterizar a cada hogar o punto de suministro:

- **LCLid:** Identificador único del punto de medición.
- **Stdor / ToU:** Tarifa estándar o por hora de uso (Time of Use).
- **Acorn:** Es una segmentación geodemográfica para residentes de UK. Clasifican cada código postal del país en 62 categorías, las cuales se pueden resumir en 18 grupos, que pueden a su vez englobarse dentro de 6 categorías mayores.
- **Acorn Grouped:** categoría mayor a la que corresponde cada Acorn.

## Datset UK Bank Holidays:

*Dimensiones iniciales: (25 ; 2 )*

Contiene las fechas y detalle de cada feriado en UK para el período de referencia:

- **Bank holiday:** Fecha de feriado
- **type:** Descripción de feriado.

Link de acceso al set de datos: <https://www.kaggle.com/jeanmidev/smart-meters-in-london>



# Análisis Exploratorio

Debido a que el proyecto de Smart Meters se implementó de manera paulatina en la ciudad de Londres, la distribución de mediciones no se reparte de manera homogénea durante el período de referencia.

Es así como durante los primeros años de análisis, la cantidad de mediciones fue muy baja alcanzando una estabilidad recién entre fines de 2012 y 2014.

Figura 1: Cantidad de mediciones diarias



Teniendo ello en consideración, para que el análisis de los patrones de consumo no se vea afectado por cuestiones relativas a la cantidad de mediciones, utilizaremos para esta sección una transformación de nuestro Dataset original, agrupándolo por Día, Acorn y Tarifa, y calculando el consumo como el promediando energy\_sum por la cantidad de observaciones de cada categoría.

Figura 2: Muestra de los primeros 3 registros de nuestro Dataset

day	Acorn_grouped	ACORN-Q	Std	energy_sum	temperatureMean	humidity	visibility	cloudCover	apparent_temperatureMean	pressure	uvIndex	dewPoint	
0	2011-11-23	Adversity	ACORN-Q	Std	5.79800	7.085	0.93	8.06	0.36	6.27	1027.12	1.0	6.29
1	2011-11-23	Comfortable	ACORN-F	Std	4.76125	7.085	0.93	8.06	0.36	6.27	1027.12	1.0	6.29
2	2011-11-23	Comfortable	ACORN-F	ToU	3.03600	7.085	0.93	8.06	0.36	6.27	1027.12	1.0	6.29

\* Por cuestiones de practicidad, solo se visualizan las primeras columnas



Antes de continuar con nuestro análisis creemos importante explicar un concepto que hasta el momento mencionamos muy sutilmente, y es la definición de “Acorn”,  
El concepto de Acorn, desarrollado por CACI Limited en Londres, es una herramienta de segmentación que clasifica a la población del Reino Unido en tipos demográficos. Se construyó en base al análisis de factores sociales y el comportamiento de la población para proporcionar información precisa y una mejor comprensión de los diferentes tipos de personas y comunidades en todo el Reino Unido. De esta manera, los Acorn segmentan hogares, códigos postales y vecindarios en 6 categorías principales, 18 grupos y 62 tipos.

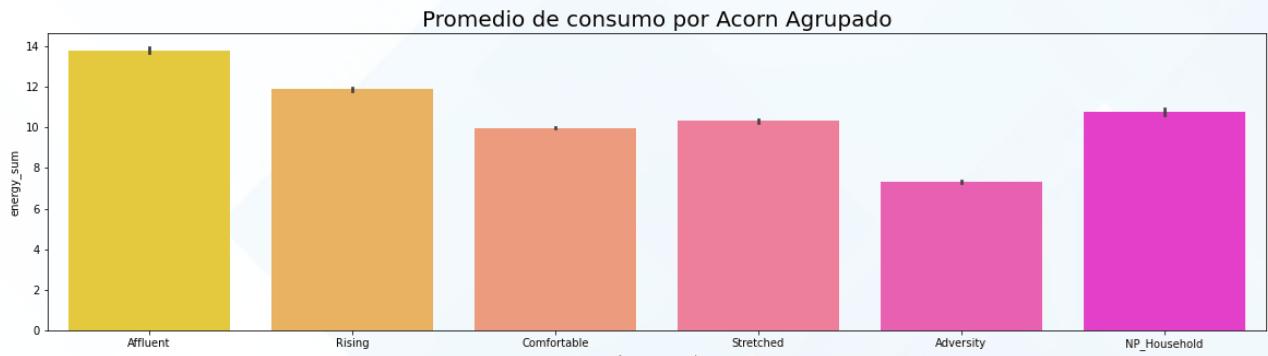
Acorn Grouped	Acorn	Acorn Grouped	Acorn
Affluent Achievers	Lavish Lifestyles	Financially Stretched	Student Life
	Executive Wealth		Modest Means
	Mature Money		Striving Families
Rising Prosperity	City Sophisticates	Urban Adversity	Poorer Pensioners
	Career Climbers		Young Hardship
Comfortable Communities	Countryside Communities	Struggling Estates	Struggling Estates
	Successful Suburbs		Difficult Circumstances
	Steady Neighbourhoods		Not Private Households
	Comfortable Seniors	Not Private Households	Not Private Households
	Starting Out		

Como hipótesis preliminar, creemos que los patrones de consumo pueden estar condicionados con ciertas características socio demográficas del consumidor. Por ejemplo, no es lo mismo el nivel de consumo eléctrico que realizará un joven estudiante que recién se emancipa (dado que posiblemente tenga menos electrodomésticos en su hogar, y pase mayor tiempo en otros lugares tales como Universidad o trabajo) que el que pueda realizar una familia con hijos y un nivel de equipamiento mayor.

Con esta hipótesis en mente, en las siguientes secciones realizaremos un breve análisis del consumo de energía eléctrica, esta vez diferenciado por Acorn al que pertenece el suministro.



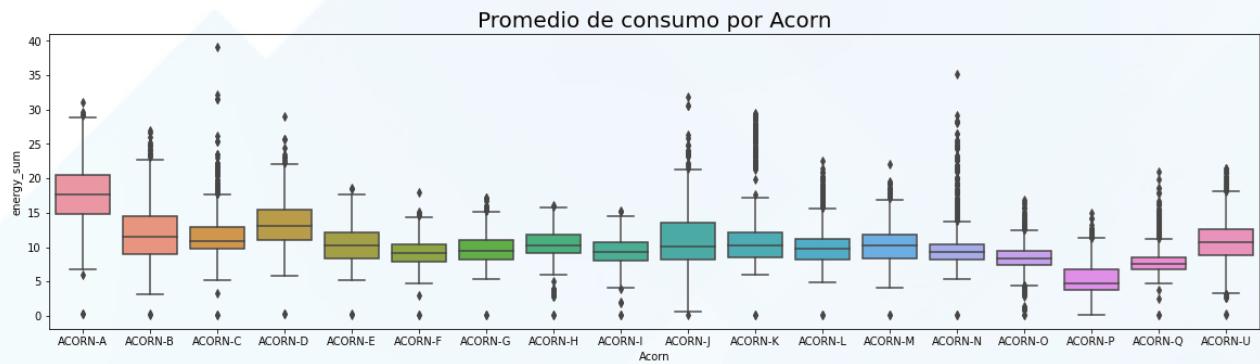
**Figura 3: Promedio diario de KwH consumidos por Acorn Grouped**



En línea con nuestra hipótesis previa, el grupo con mayor poder adquisitivo y estabilidad socio económica es quien en promedio posee un mayor consumo de energía eléctrica.

Esta misma conclusión se desprende también al analizar cada Acorn dentro de los grupos generales, vemos como el Acorn A es quien en promedio presenta un mayor patrón de consumo durante el período de referencia.

**Figura 4: Promedio diario de KwH consumidos por Acorn**



Para aquellos lectores que no estén familiarizados con el gráfico de la figura 4, nos tomaremos unos breves renglones a fin de clarificar los principales conceptos.

El Diagrama de Caja y bigotes (Box Plot en inglés) es un tipo de gráfico que muestra un resumen de una gran cantidad de datos en cinco medidas descriptivas, permitiéndonos identificar valores atípicos y comparar distribuciones.

Las dimensiones de la caja está determinada por la distancia del rango intercuartílico (diferencia entre cuartil 1 y 3), mientras que el segmento que divide la caja en dos partes es la mediana, que facilitará la comprensión de si la distribución es simétrica o asimétrica.

La continuación de dos segmentos en la caja se denominan bigotes, que determina el límite para la detección de valores atípicos. Los bigotes deben tener una longitud máxima, que es 1,5 veces el rango intercuartílico. Cualquier observación que se encuentre por encima de ese valor (sea para el límite superior o inferior), se considerará “outlier” o valor atípico.



## Demanda de energía como serie temporal

Hasta el momento, hemos realizado un análisis estático de la demanda de energía eléctrica. Pudimos observar como en promedio, para todo el período de referencia, distintas agrupaciones de usuarios presentan distintas características y patrones de consumo.

Ahora bien, dado que nuestra principal variable de análisis se origina a partir de mediciones diarias (en el Dataset original se reportan mediciones cada media hora, a efectos de simplificar el análisis tomamos un promedio diario), y no menos importante, contamos con dicha información, creemos oportuno abordar el problema a partir de un análisis de series temporales.

Por serie de tiempo nos referimos a datos estadísticos que se recopilan, observan o registran en intervalos de tiempo regulares (diario, semanal, semestral, anual, entre otros). El término serie de tiempo se aplica por ejemplo a datos registrados en forma periódica que muestran, por ejemplo, la cotización de una acción en el mercado, la evolución del PBI, o en nuestro caso, el consumo diario de energía eléctrica en Reino Unido.

Lo interesante de las series temporales, es que permiten encontrar patrones de interacción con otras variables, ya sea macroeconómicas, climáticas, sociodemográficas, entre otras, que pueden ser estudiadas también a partir de sus propias series.

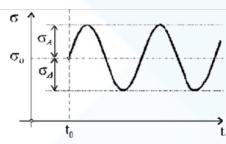
A grandes rasgos, podemos identificar en las series de tiempo cuatro tipos de variación, que actuando de manera conjunta le dan a la misma sus valores y forma característica:



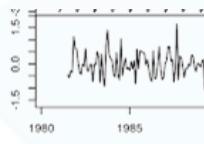
**1. Tendencia :** La tendencia secular o tendencia a largo plazo de una serie es por lo común el resultado de factores a largo plazo. En términos intuitivos, la tendencia de una serie de tiempo caracteriza el patrón gradual y consistente de las variaciones de la propia serie, que se consideran consecuencias de fuerzas persistentes que afectan el crecimiento o la reducción de la misma.



**2. Estacionalidad:** Esta variación corresponde a los movimientos de la serie que recurren año tras año en los mismos meses (o en los mismos trimestres) del año poco más o menos con la misma intensidad.



**3. Variación cíclica:** Largas desviaciones de la tendencia debido a factores diferentes de la estacionalidad. Los ciclos por lo general se producen durante un intervalo de tiempo extenso, y los tiempos que transcurren entre los picos o valles sucesivos de un ciclo no necesariamente son iguales.

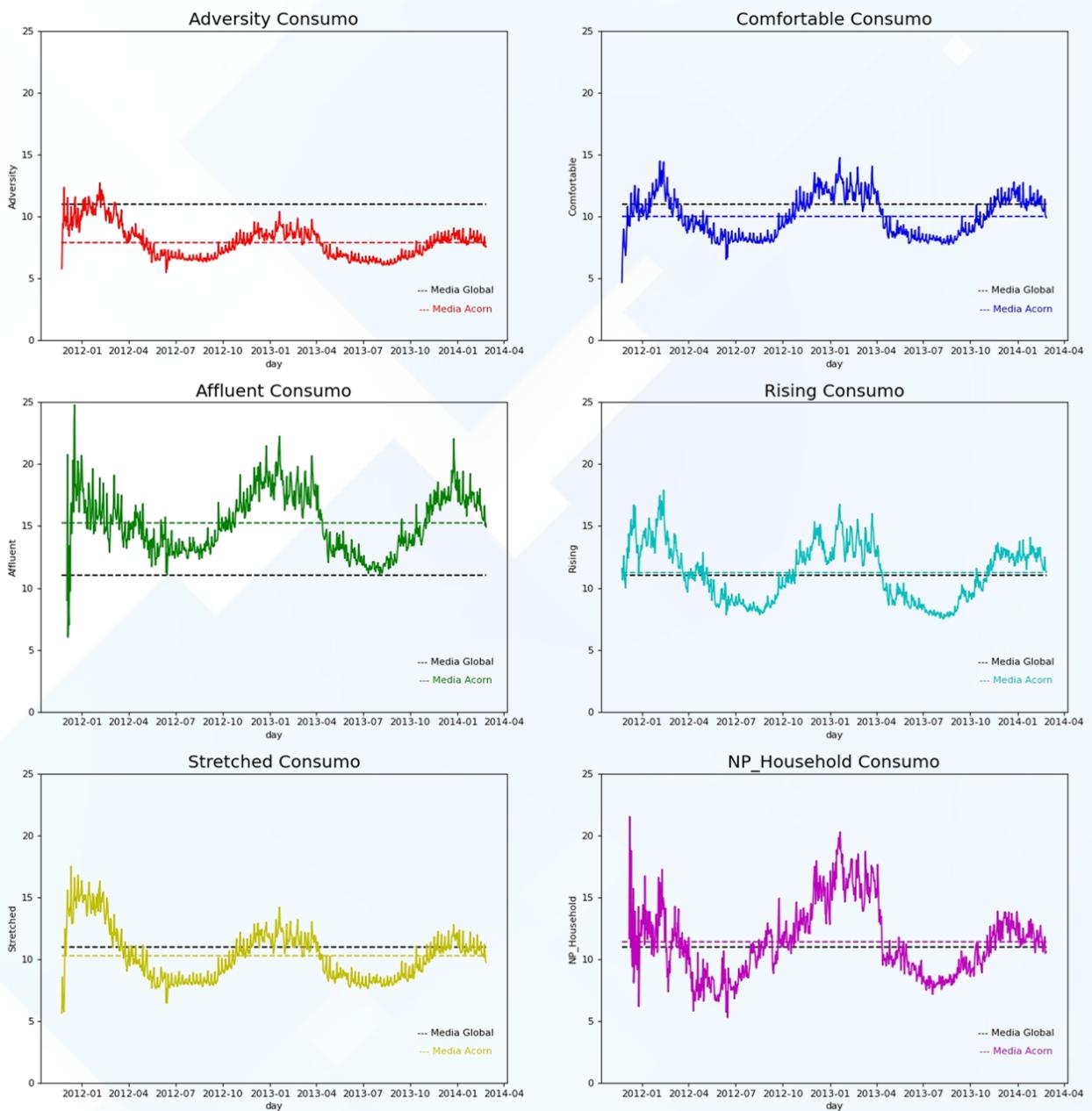


**4. Variación Irregular:** Esta se debe a factores a corto plazo, imprevisibles y no recurrentes que afectan a la serie de tiempo. Este movimiento que queda después de explicar los movimientos de tendencia, estacionales y cíclicos; ruido aleatorio o error en una serie de tiempo.



Luego del breve repaso de series temporales, podemos procederemos a graficar el consumo de energía eléctrica, por cada grupo de Acorn, a lo largo de todo el período de análisis.

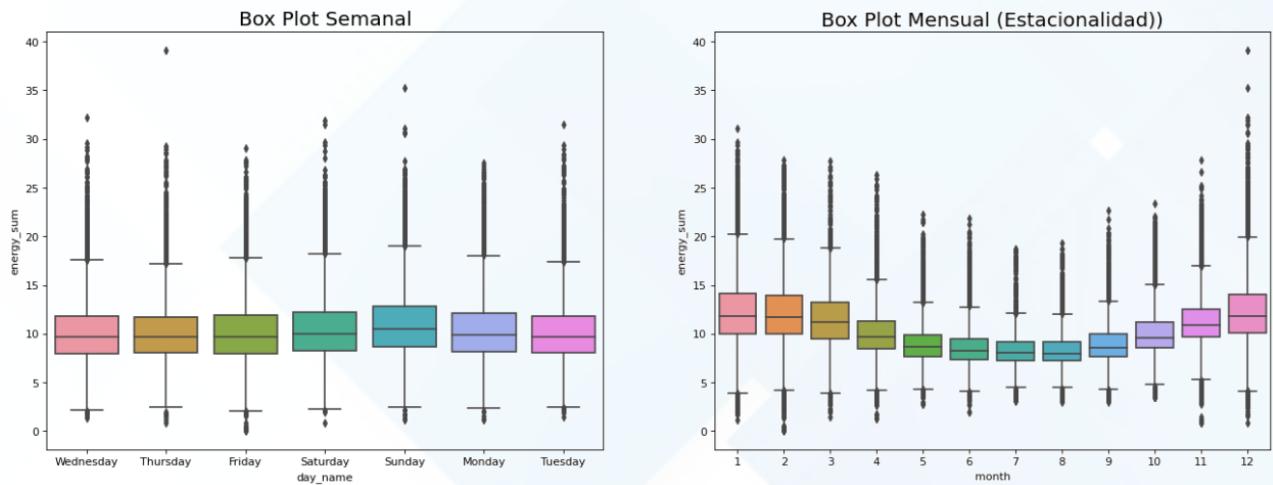
**Figura 5: Consumo diario de energía eléctrica, por grupo de Acorn**



Dado que nuestra serie es relativamente corta, es esperable que no podamos observar comportamientos de Tendencia o Ciclos, que por lo general se aprecian en el largo plazo. Sin embargo, podemos apreciar claramente la componente estacional con picos y valles en períodos regulares, y también volvemos a reforzar el insight de que distintos tipos de consumidores reflejan distintos patrones y niveles de consumo.



Figura 6: Consumo de energía diario y mensual



Analizando más en detalle el consumo de energía en función del tiempo, podemos identificar picos de consumo en determinados días de la semana como los domingos, y en los meses entre noviembre y marzo.

El primer caso, es debido a que estamos analizando la demanda de energía residencial, con lo cual tendría sentido que durante los fines de semana, la gente tiene más tiempo para estar en su casa y hacer uso de las instalaciones.

Respecto al segundo hallazgo, creemos que merece la pena detenernos un poco más en la relación entre otras series de tiempo presentes en nuestro dataset, y el consumo de energía.

## Energía y Variables Climáticas

Si por un momento nos ponemos a pensar en el consumo de energía en nuestros hogares, generalmente podremos identificar dos notorios picos de consumo:

- En el invierno, con la utilización de estufas eléctricas, aires acondicionados y otros electrodomésticos para calentar los hogares (de manera simplificada, podemos decir que los artefactos que generan calor poseen un consumo energético considerable).
- En el verano, con la mayor utilización de aires acondicionados, que en los últimos años comenzaron a encontrarse en muchas residencias.

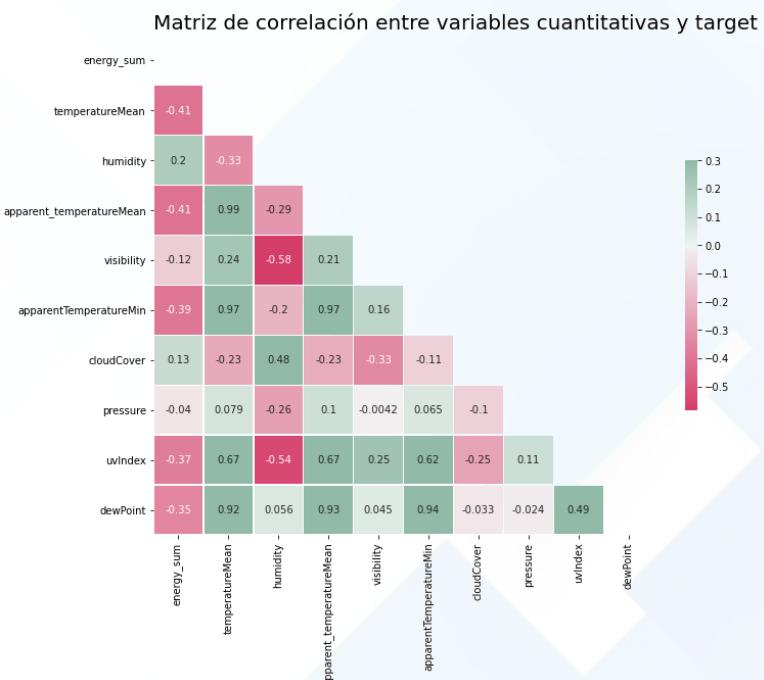
Volviendo a la región de donde provienen las mediciones, caeremos en la cuenta de que tenemos un solo pico de consumo bien marcado, entre los meses de noviembre y marzo.

Ello es porque en ese rango de fechas nos encontramos en pleno invierno, con temperaturas que no superan los 6° y baja sensación térmica debido a los vientos y abundantes lluvias, mientras que si bien en los meses de junio a septiembre nos encontramos en verano, las temperaturas no son mucho mayores a los 20°, con lo cual el consumo energético para climatizar no se hace tan necesario como en otros hemisferios.



Finalizaremos nuestra sección descriptiva, analizando las relaciones entre nuestras variables climáticas y nuestro target .

### Figura 7: Matriz de Correlación

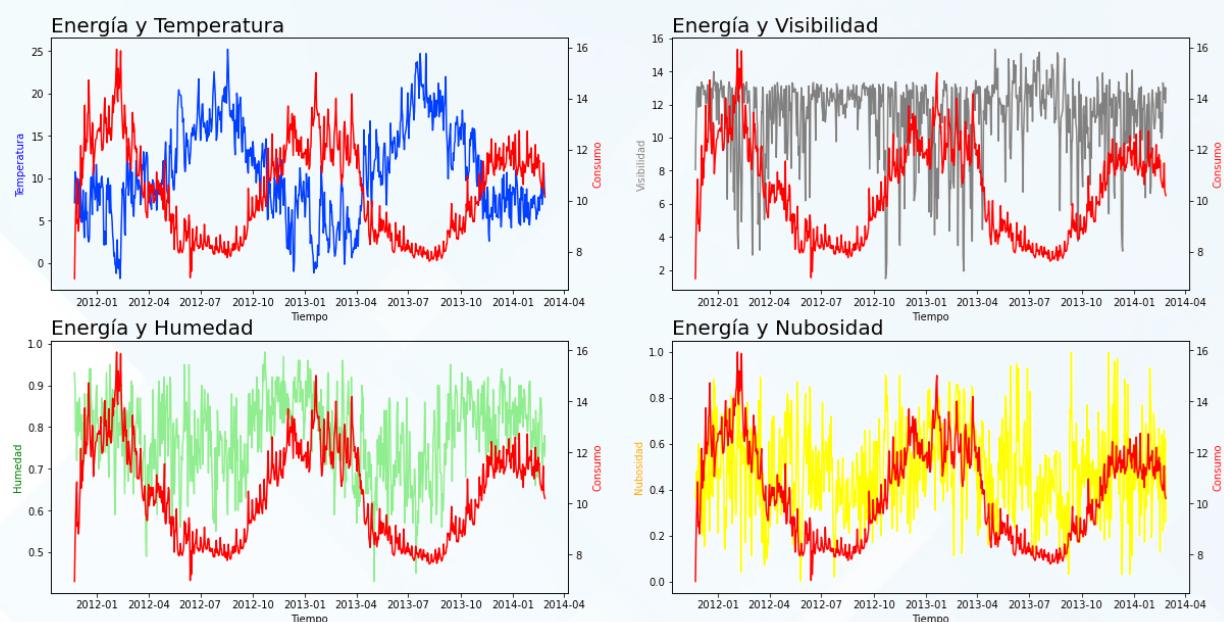


La matriz de correlación, nos muestra de una manera rápida y visual, la intensidad y el sentido de la relación entre cada una de nuestras variables cuantitativas. La intensidad, viene dado por el valor numérico de la correlación, a mayor valor absoluto, más fuerte es la relación entre las mismas.

En sentido de la relación está marcado por el signo, correlaciones positivas indican variables que crecen o decrecen en conjunto, mientras que negativas nos muestran un comportamiento inverso.

Como podremos observar a continuación en la figura 8, la temperatura posee una correlación negativa fuerte con el consumo de energía: valores elevados de consumo suelen presentarse con temperaturas bajas, y viceversa.

### Figura 8: Series climáticas y consumo



\* Algo importante a destacar, es que correlación no implica causalidad. Para decir que una variable causa un efecto en otra, se requiere otro tipo de análisis.



# Problema 1:

## Estimación de la demanda como serie temporal

---

Comenzaremos con un modelo simplificado de predicción trabajando con el subset de datos agrupados por día y condiciones climáticas, sin considerar tipo de tarifa ni Acorn.

Optamos por el **consumo promedio** como nuestra variable objetivo, ya que al tener distinta cantidad de mediciones a lo largo del tiempo, la demanda agregada de energía no es un valor completamente limpio, ya que se ve afectado por la cantidad de mediciones realizadas en cada período. Según la documentación, esto puede deberse a que la implementación de este proyecto de Smart Meters se realizó de manera gradual.

Dado que en todo modelo, la información que brindemos será determinante del resultado que nos arroje, realizamos algunas transformaciones y limpieza previa a nuestro sub set.

No es nuestro objetivo detenernos mucho más en aspecto de Limpieza y Data Wrangling, pero básicamente luego de realizar el primer Group By para quedarnos con promedio de consumo diarios, realizamos una limpieza de outliers (valores atípicos), transformamos variables de fecha a un formato DateTime y creamos nuevas features tales como un índice de tiempo, y variables dummy para los meses y días.

A grandes rasgos, nuestros nuevos datos se ven de la siguiente mantera:

day	energy_sum	apparent_temperatureMean	apparentTemperatureMin	humidity	visibility	cloudCover	pressure	uvIndex	dewPoint	temperatureMin	...	December	February
2013-06-24	8.41986	13.36	10.62	0.66	12.89	0.52	1021.79	5.0	7.04	10.62	...	0	0
2013-06-25	8.02719	15.01	11.26	0.57	14.24	0.42	1028.61	6.0	6.54	11.26	...	0	0

Podrán notar como las variables Dummy incrementaron nuestra cantidad de columnas, pero ello nos permitirá hacerle entender a nuestro modelo la presencia o ausencia de determinada variable categórica.

Una variable Dummy toma valores binarios, en donde el cero representa la carencia del atributo, mientras que el uno la presencia.

En nuestro caso, cuando transformamos de esta manera la columna “mes”, estamos creando 12 columnas nuevas, una para cada mes del año, en donde la columna representativa del mes en curso tendrá valor uno, mientras que las restantes un cero.

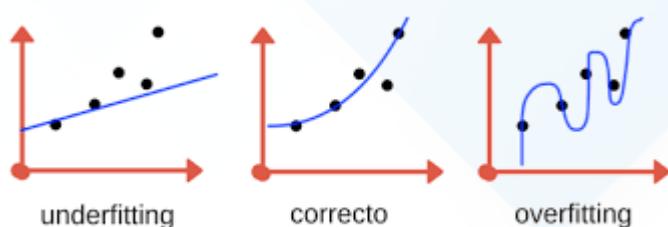
Debido a que solo once columnas nos alcanzan para darnos cuenta en que mes estamos, y a efectos de evitar la multicolinealidad en nuestras variables, al generar las dummy siempre descartamos una de las columnas generadas.



## Separación en datos de Entrenamiento y Testeo

Cuando desarrollamos y configuramos un algoritmo de machine learning, el siguiente paso es brindarle nuestros datos para que “aprenda” las características de los mismos, y en base a ello pueda arrojarnos un output acertado.

Ahora bien, puede ocurrir el caso que nuestro modelo aprenda tan bien de nuestros datos, que genere un overfitting o sobre ajuste, que se produce cuando el algoritmo aprende incluso del ruido de nuestro dataset, con lo cual pierde capacidad predictiva sobre datos nuevos.



En el caso del overfitting, vemos como la curva se ajusta exactamente a cada dato, pero pierde capacidad predictiva frente a valores nuevos (justamente lo opuesto a nuestro objetivo)

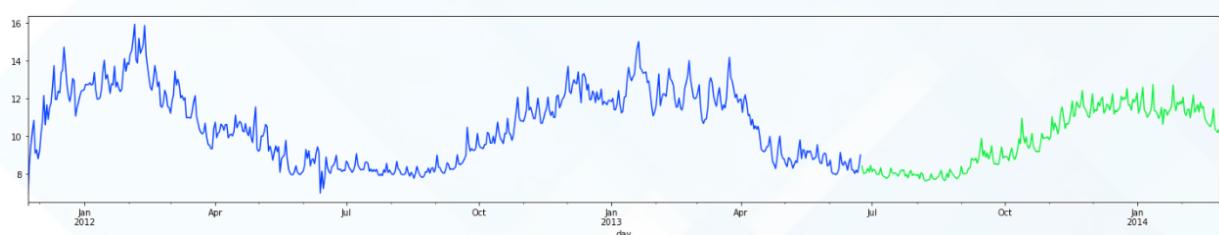
Una operación que es común en todos los modelos de aprendizaje supervisado es la división de nuestro conjunto de datos en (al menos) dos partes: una parte Train, que corresponderá a la mayor parte de nuestro dataset y que usaremos para entrenar nuestro modelo y un parte Test, de menor tamaño, sobre la que evaluaremos nuestro modelo entrenado.

La idea central entonces, es que nuestro algoritmo entrene y aprenda en los datos de Train (en esta etapa podremos hacer todas las pruebas y ajustes que queramos), y luego lo probaremos una única vez con los datos de testeо, datos nunca antes vistos.

Si comparamos alguna métrica de error o de performance, es esperable que el desempeño en los datos de testeо sea ligeramente inferior (ya que son desconocidos!), pero no debería caer demasiado.

Hay diversas maneras de realizar la separación de datos, pero dado que nosotros estamos trabajando con series de tiempo, no queremos que datos del futuro se mezclen con los del pasado, ya que estaríamos generando Data Leakage o filtrado de datos: nuestro modelo no debería ver mientras entrena, el valor del target que va a predecir.

Es por ello que realizamos un ordenamiento cronológico de nuestras observaciones, y dividimos en datos de entrenamiento al 70% más antiguo, y en datos de testeо al 30% restante, quedando de la siguiente manera:

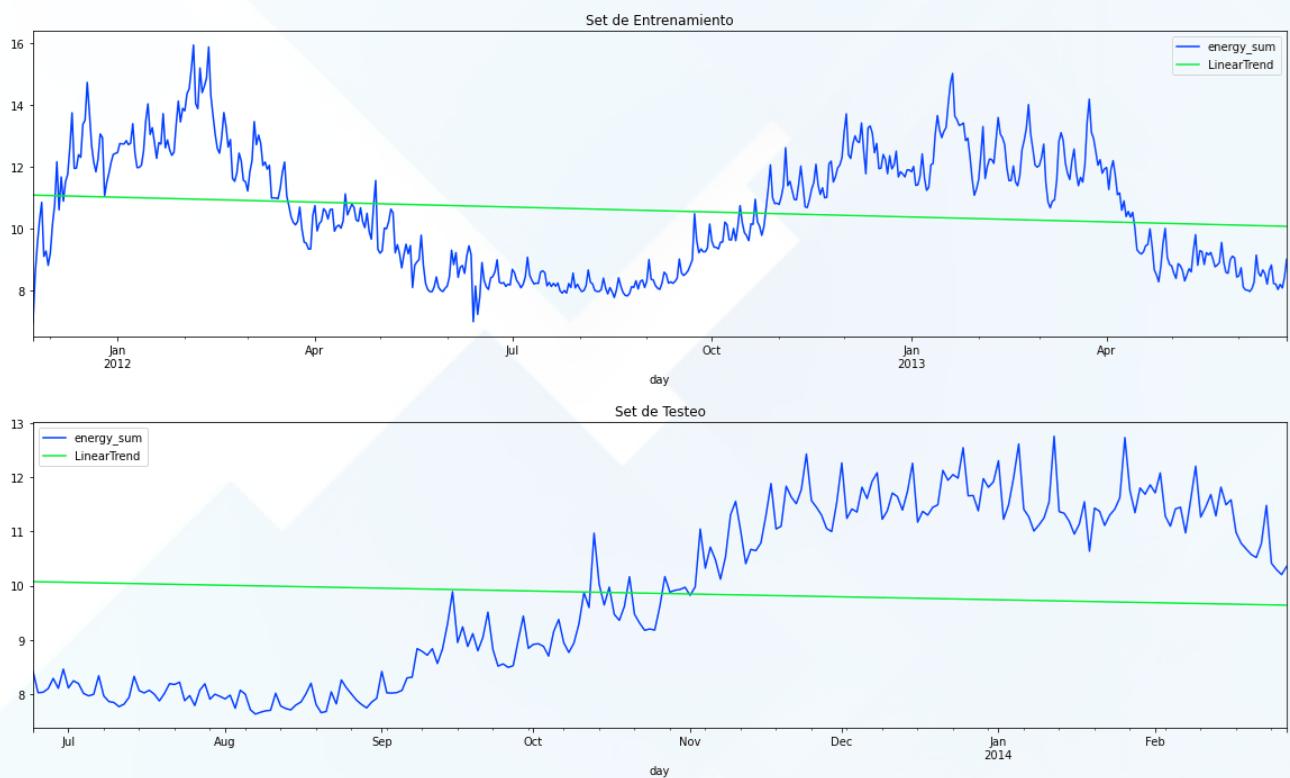




## Modelo de tendencia lineal:

Un primer abordaje que podamos plantear, es la de suponer que podemos hallar una recta que minimice el error cuadrático medio.

Sin embargo, como ya hemos visto en la sección introductoria, debido a que nuestra variable target no es lineal, un modelo de este estilo será incapaz de lograr una predicción acertada



En el recuadro superior se grafica la variable real versus nuestra predicción en el set de entrenamiento, mientras que en recuadro inferior en el set de testeo.

Si bien la recta de color verde es la que minimiza el error cuadrático medio, lejos está de captar la génesis de nuestra variable objetivo, arrojando como consecuencia una pobre predicción.

En algunos casos, cuando nuestra variable objetivo posee una varianza que no es constante, sino que se va incrementando, puede ser interesante realizar una trasformación logarítmica a la misma, entrenar el modelo y predecir con dicha variable transformada, y luego transformar las predicciones nuevamente a su escala original.

Luego de realizada dicha transformación con el modelo lineal simple, el rendimiento de nuestro modelo fue ligeramente superior, con lo cual decidimos mantenerlo para los siguientes abordajes (a efectos de no extender demasiado esta sección, mostraremos solo los modelos más interesantes desde el punto de vista de los resultados arribados).

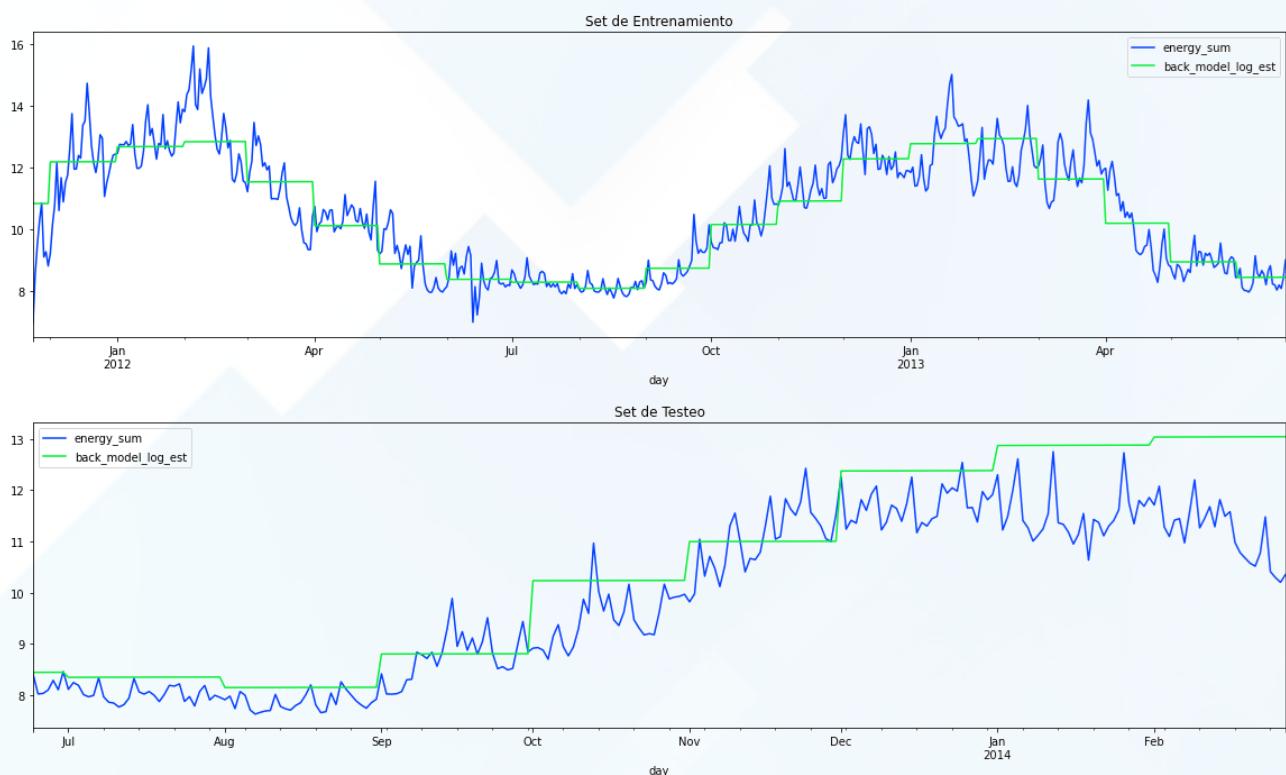


## Modelo con transformación logarítmica y estacionalidad:

Gracias a el primer análisis exploratorio que nos permitió entender un poco mejor a nuestra variable, una idea interesante es modelar el efecto de la estacionalidad.

Gracias a las variables Dummy previamente creadas, podemos entrenar y testear nuestro modelo con información relativa a que mes y día de la semana es en cada medición.

Si miramos nuevamente la figura de entrenamiento y testeо, podemos apreciar como nuestras predicciones tienen ahora una forma “escalonada”, y comienza a ajustarse un poco mejor a la estructura de nuestra variable target.



Si bien en el set de entrenamiento nuestras predicciones parecen seguir, con notable error, la variable original, en el set de testeо la performance se ve empobrecida. Sobre los valores finales de nuestro dataset, nuestra predicción continúa creciendo, pese a que los valores reales de consumo disminuyen.

Algo nos indica que aún podemos mejorar nuestro modelo, dotándolo de mayor complejidad.



### Modelo con transformación logarítmica, estacionalidad y variables climáticas:

El momento que todos estábamos esperando ha llegado, es hora de introducir las variables climáticas!

Como demostramos en la primera parte de este documento, tanto la sensación térmica, como la humedad, visibilidad, y otras condiciones climáticas presentan cierta relación con el consumo de energía.

Al introducir entonces dichas features a nuestro modelo para que entrene, podemos notar como las predicciones mejoran notablemente, y los saltos que al principio parecían caprichosos, guardaban cierta relación con las condiciones climáticas de cada día en particular.



Incluso ya en el set de testeo, las predicciones respetan el patrón real de consumo sobre el final del período, señal de que nuestro modelo aprendió “un poco más” de la variable objetivo durante el entrenamiento.



## Modelo ARIMA y análisis de residuos

El modelo Arima es una metodología econométrica basada en modelos dinámicos que utiliza datos de series temporales.

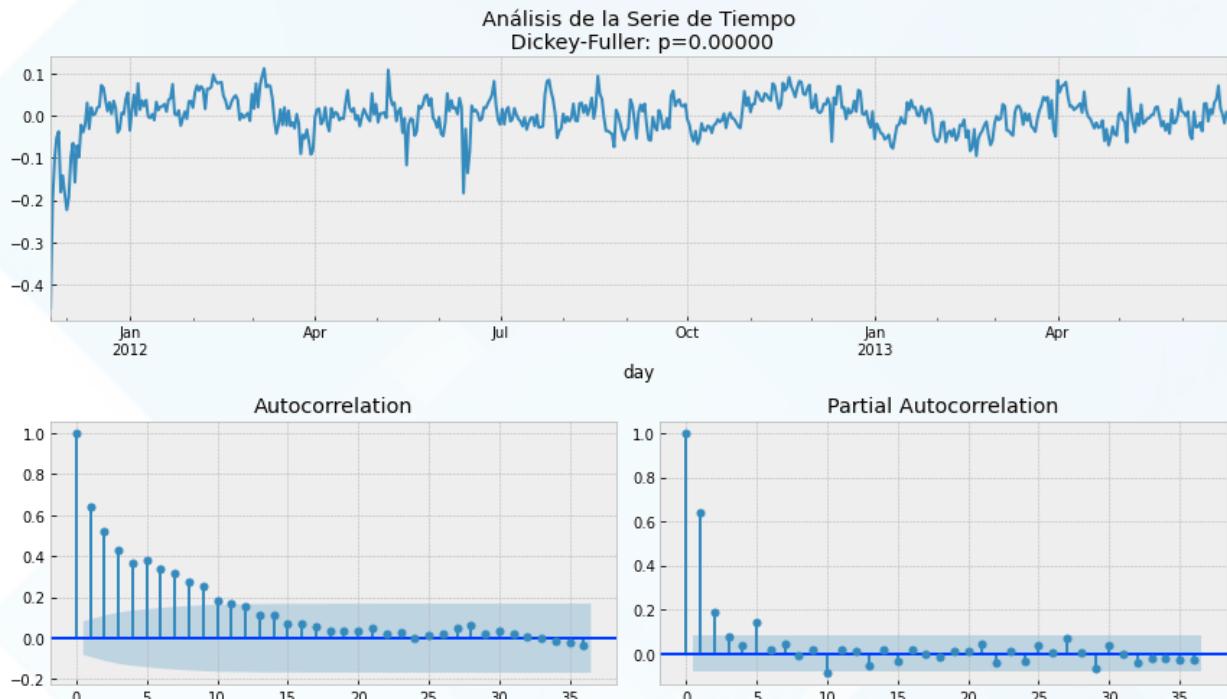
Para identificar cual es el proceso ARIMA que ha generado una determinada serie temporal es necesario que los datos sean estacionarios, es decir, no pueden presentar tendencia creciente o decreciente (si presentan tendencia habría que diferenciar la serie porque la serie no es estacionaria en media), ni tampoco pueden presentar fluctuaciones de diferente amplitud. Si la dispersión no se mantiene constante entonces la serie no es estacionaria en varianza y habría que transformarla siendo, la transformación logarítmica la más habitual.

Una vez que la serie es estacionaria es necesario obtener las funciones de autocorrelación simple y parcial muestrales para determinar el proceso ARIMA( $p,d,q$ ) más adecuado que haya podido generar la serie estacionaria.

En los modelos ARIMA( $p,d,q$ ),  $p$  representa el orden del proceso autorregresivo,  $d$  el número de diferencias que son necesarias para que el proceso sea estacionario y  $q$  representa el orden del proceso de medias móviles.

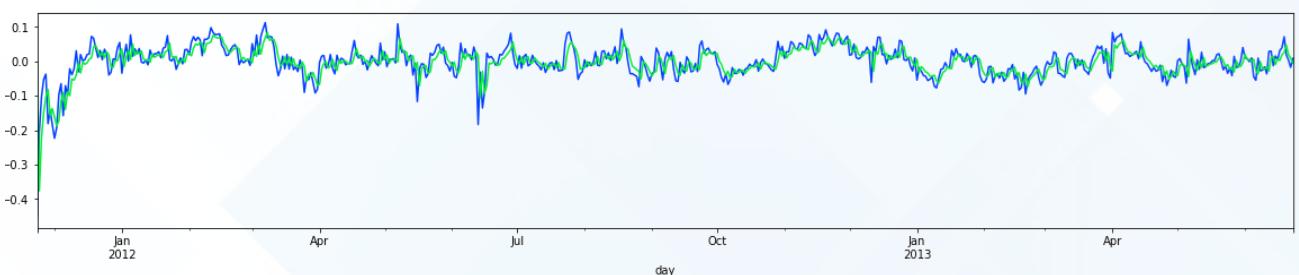
Una vez estimado y validado el modelo Arima se puede utilizar para obtener valores futuros de la variable objeto de estudio

Para comenzar, analizaremos la estacionariedad de los residuos de nuestra serie de tiempo, y a su vez su autocorrelación y autocorrelación parcial.



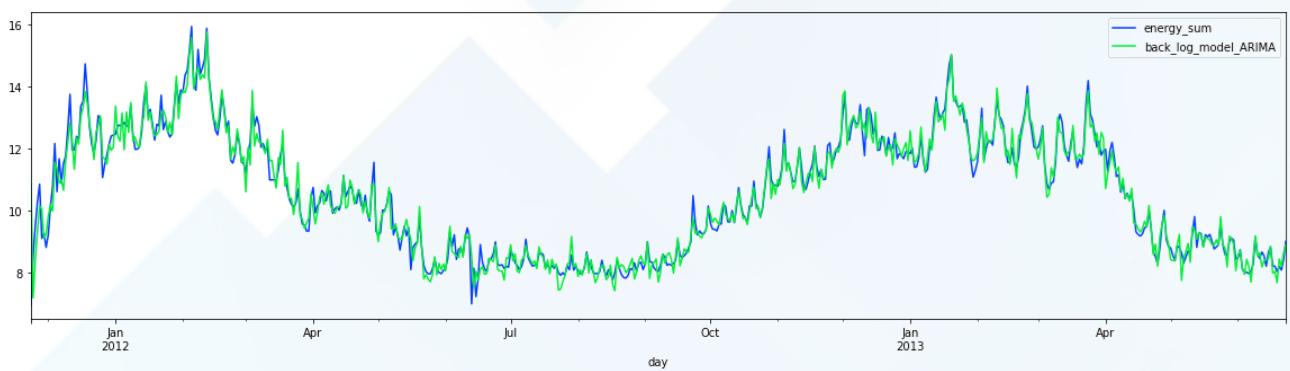


Con los parámetros seleccionados, entreno el modelo y realizamos nuestra predicción, según como vemos en la figura inferior.

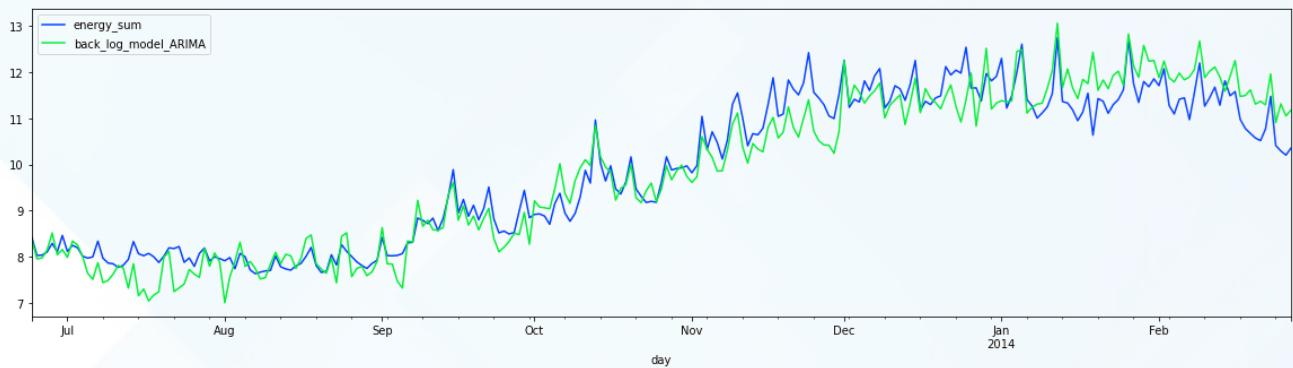


Finalmente, incorporamos este nuevo input a nuestro modelo con variables climáticas, a fin de mejorar nuestra predicción:

### Train Set



### Test Set





## Modelo con datos de panel

Como vimos, el último modelo con la transformación logarítmica, y la incorporación de dummy de Meses, Días y Variables climáticas obtuvo una performance respetable. Luego, al analizar los residuos de este modelo, pudimos avanzar en la mejora de la capacidad predictiva, a partir de la utilización de ARIMA.

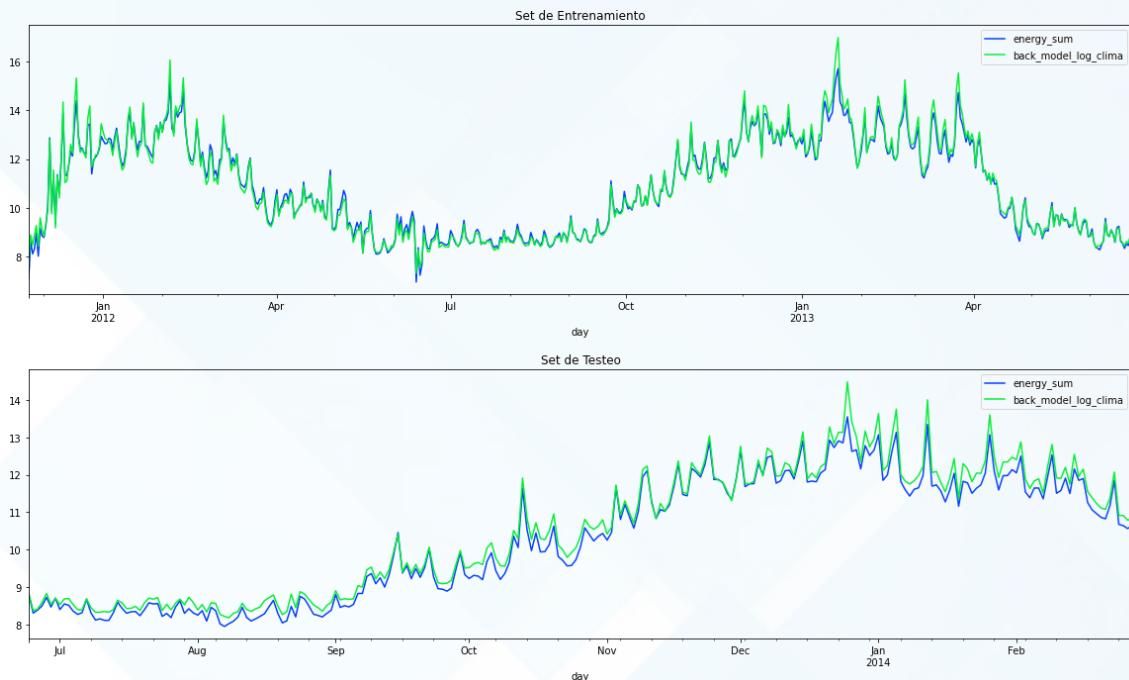
La problemática es tanto compleja como motivante, y podríamos dedicar mucho más tiempo a intentar perfeccionar nuestro modelo. Pero a los fines de este trabajo, creímos conveniente dar un punto y aparte, para pasar a aplicar de forma rápida y solo a efectos de hacer un primer acercamiento, otros tipo de modelos y técnicas que tuvimos el agrado de investigar.

En los modelos anteriores, si bien pudimos introducir una gran cantidad de variables con poder explicativo del consumo de energía, al haber transformado nuestro dataset a un promedio de consumo de energía diario, perdimos información acerca de qué tipo de usuario consumió cada unidad de energía.

Es por ello, que una alternativa es realizar una aproximación mediante la utilización de datos de panel. Con ese objetivo, reorganizaremos ligeramente nuestra estructura de datos, conservando esta vez las features demográficas, en un Dataset Multi Index.

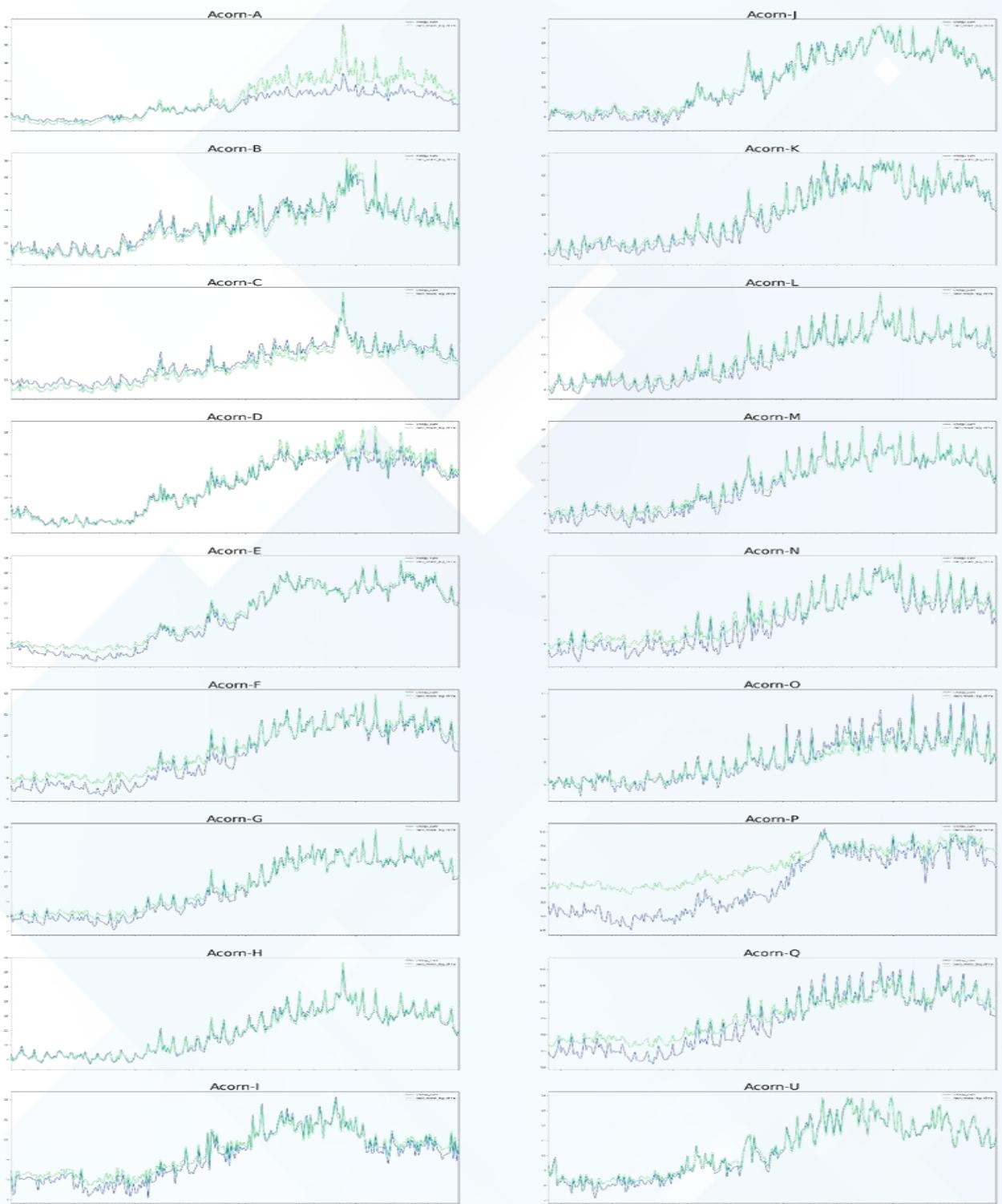
En resumidas cuentas, los datos con los que trabajaremos para este último modelo de series temporales, tendrán dos índices, el primero representativo de la fecha de consumo, y el segundo relativo al Grupo de Acorn. Luego, nuestra variable objetivo se interpreta como el promedio de energía consumida diaria, por cada Grupo de Acorn.

Como en los casos anteriores, procederemos a entrenar y testear nuestro modelo a partir de la incorporación de esta nueva feature:





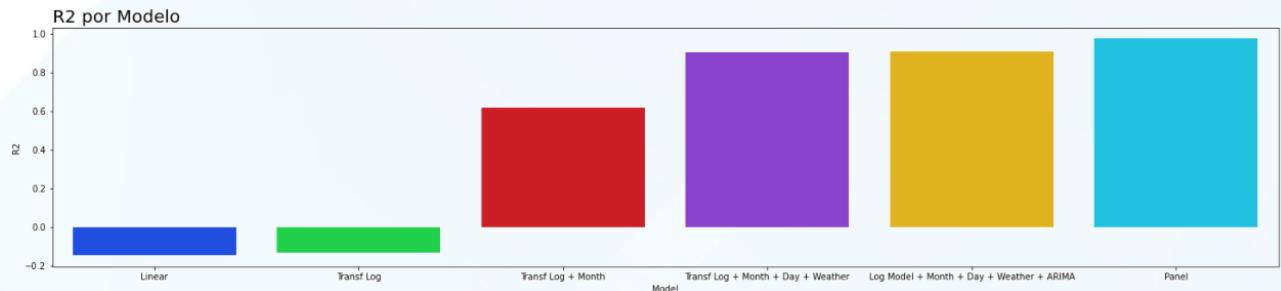
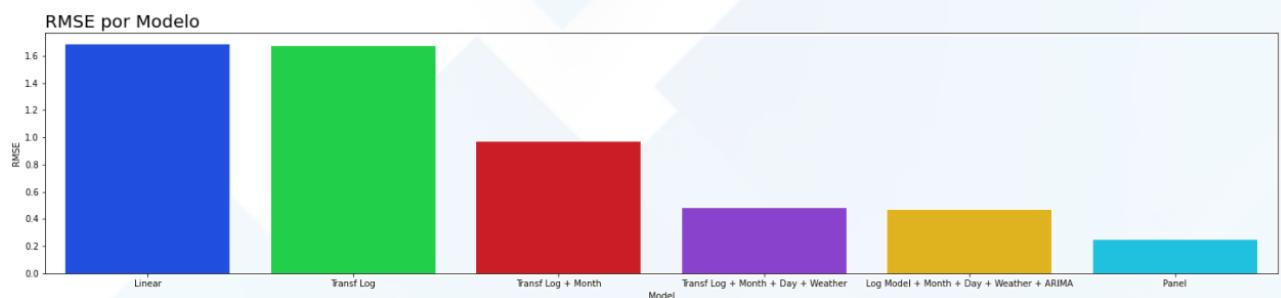
Como podremos ver en el siguiente gráfico, no para todos los Acorn nuestro modelo pudo realizar correctas predicciones. Sin embargo, no caben dudas que agregar esta información mejoró nuestra estimación general de los valores promedio de consumo.





Antes de pasar a la siguiente sección, resumiremos las principales medidas de error y de score que fuimos recopilando a través de los distintos modelos.

Model	RMSE	R2
Linear	1.68338	-0.149072
Transf Log	1.67298	-0.134916
Transf Log + Month	0.970193	0.618322
Transf Log + Month + Day + Weather	0.481693	0.905915
Log Model + Month + Day + Weather + ARIMA	0.467491	0.911381
Panel	0.246516	0.97664





## Problema 2: Estimación puntual de demanda de energía

---

El segundo problema que indagaremos podría resumirse a partir de la siguiente pregunta: ¿es posible construir un modelo que prediga la demanda de un determinado consumidor en un momento dado en el tiempo?

Esta cuestión nos parece interesante sobre todo desde el punto de vista técnico, y podría ser importante para la toma de decisiones en distintas áreas y para distintos objetivos del negocio. Poder estimar cuánto consume determinado usuario o conjunto de usuarios en un momento dado del tiempo sería útil para evaluar inversiones en la red eléctrica, pero también permitiría encontrar un momento/lugar óptimos para la interrupción del servicio para realizar reparaciones, o alimentar otras zonas de mayor demanda. Incluso podría servir para evaluar potenciales casos de fraude, si un consumidor presenta en la realidad un consumo muy diferente al estimado para él o para los consumidores de perfil similar en un momento dado.

Para este problema utilizaremos la misma base de datos que utilizamos anteriormente pero con una complejidad mayor, ya que contamos con **los mismos datos de consumo desagregados cada media hora**. Esto nos permitirá darle un mayor grado de detalle a nuestro modelo, que también tendrá el resto de componentes vistos: variables climatológicas, estacionales y socioeconómicas.

Ahora bien, esta complejidad presenta un cierto inconveniente, dada la gran masividad de datos que nuestro modelo debería manipular y con ello el gran poder de cómputo requerido para poder ejecutarlo. Para tener un dimensionamiento del inconveniente: en lugar de contar con 3,5 millones de registros de demanda de energía concentrados en un único archivo, **ahora lidiaremos con un total de 168 millones de registros repartidos en 106 archivos .csv diferentes**, y eso sólo en lo que se refiere a datos de consumo.

Para poder abordar este problema dividimos el trabajo en distintas etapas:

1. Selección de un subset de datos.
2. Data wrangling, o modelado de datos para su manipulación posterior.
3. Evaluación de distintos modelos de Machine Learning y definición del modelo “óptimo” para el problema.
4. Evaluación de la escalabilidad del modelo.
5. Optimización del modelo.
6. Conclusiones



## 1. Selección de un subset de datos

Para abordar el problema decidimos recortar el análisis a un número de registros reducido y que sea manipulable a través de las herramientas de procesamiento centralizado. Fuera del alcance de este trabajo está la aplicación de métodos de procesamiento distribuido que podrían manejar el volumen total de datos de una sola vez y sin mayor inconveniente.

Para ello, en una primera instancia **seleccionaremos al azar un único archivo** de los 106 existentes: el .csv correspondiente al **block 16**. Cada .csv contiene los registros de una manzana determinada para todo el periodo, detallado a nivel usuario. Por otro lado, se observó que las manzanas tienen una composición relativamente homogénea en términos de Acorn, y en la mayoría de los casos todos los usuarios de una manzana pertenecen a una misma categoría.

Figura 5: Detalle de cantidad de usuarios, mediciones y Acorn por archivo

block	Acorn	LCLid	measures	block	Acorn	LCLid	measures	block	Acorn	LCLid	measures
block_0	ACORN-A	48	24793	block_17	ACORN-E	50	30723	block_36	ACORN-E	50	32052
block_1	ACORN-A	50	31727	block_18	ACORN-E	50	31783	block_37	ACORN-E	50	31844
block_10	ACORN-D	50	31413	block_19	ACORN-E	50	32608	block_38	ACORN-E	50	30709
block_100	ACORN-Q	50	31681	block_2	ACORN-A	50	31132	block_39	ACORN-E	50	32528
block_101	ACORN-Q	50	30080	block_20	ACORN-E	50	36105	block_4	ACORN-C	50	30794
block_102	ACORN-Q	50	29230	block_21	ACORN-E	50	33845	block_40	ACORN-E	50	31857
block_103	ACORN-Q	50	31806	block_22	ACORN-E	50	33296	block_41	ACORN-E	50	31378
block_104	ACORN-Q	50	32771	block_23	ACORN-E	50	29836	block_42	ACORN-E	50	31924
block_105	ACORN-Q	50	32027	block_24	ACORN-E	50	32707	block_43	ACORN-E	44	28204
block_106	ACORN-Q	50	31101	block_25	ACORN-E	50	31975		ACORN-F	6	3929
block_107	ACORN-Q	50	31029	block_26	ACORN-E	50	30048	block_44	ACORN-F	50	33749
block_108	ACORN-Q	50	33303	block_27	ACORN-E	50	33433	block_45	ACORN-F	50	33180
block_109	ACORN-Q	48	30103	block_28	ACORN-E	50	35951	block_46	ACORN-F	50	34518
block_11	ACORN-D	50	32711	block_29	ACORN-E	50	33291	block_47	ACORN-F	50	34660
block_110	ACORN-Q	17	10247	block_3	ACORN-A	9	5421	block_48	ACORN-F	50	33630
	ACORN-U	32	18447		ACORN-B	25	14462	block_49	ACORN-F	50	31915
block_111	ACORN-U	16	9794		ACORN-C	16	8967	block_5	ACORN-C	50	29554
block_12	ACORN-D	27	17712	block_30	ACORN-E	50	34641	block_50	ACORN-F	50	32348
	ACORN-E	23	15427	block_31	ACORN-E	50	32206	block_51	ACORN-F	50	33045
block_13	ACORN-E	50	32865	block_32	ACORN-E	50	31345	block_52	ACORN-F	50	31459

\* A modo de ejemplo se muestran el detalle de los primeros 55 archivos.

Sabemos que este modo de proceder representará un limitante a nuestro modelo, que estará entrenado con un determinado perfil de usuario, el de la manzana seleccionada, y que probablemente no sea representativo del universo de usuarios. Más específicamente, el modelo aprenderá el comportamiento del consumo en relación a un único “prototipo” de usuario, pero en la realidad esta relación o función podría cambiar para cada subset de datos. Trataremos de salvar esta omisión más adelante, para enfocarnos ahora en evaluar los distintos modelos que se pueden aplicar para resolver el problema.



## 2. Data Wrangling

Una vez seleccionado el subset de datos para entrenar nuestro modelo, realizaremos las tareas de data wrangling necesarias, es decir, las transformaciones para poder manipular los datos y dejarlos disponibles para el modelado.

Sin entrar en mayores detalles de limpieza, en este proceso seleccionamos los registros posteriores al año 2012, que tal como habíamos visto en la primera parte, es el momento a partir del cual la cantidad de registros se estabilizó para todos los usuarios. Creamos las variables referidas a la fecha, como representativas de la estacionalidad, agrupamos las mediciones por hora, lo cual redujo la cantidad de registros a la mitad, y sumamos los registros de todos los usuarios para obtener un **consumo total por manzana**. Esta variable ('energy') será la variable target u objetivo de nuestros modelos.

**Figura 6: Datos para modelar**

	year	month	week	day_name	hour	visibility	windBearing	temperature	dewPoint	pressure	apparentTemperature	windSpeed	precipType	humidity	summary	is_holiday	energy
0	2013	1	1	Friday	0	13.33	255.0	10.79	8.64	1036.97	10.79	4.76	rain	0.87	Mostly Cloudy	True	16.728
1	2013	1	1	Friday	1	13.50	257.0	10.28	8.24	1036.82	10.28	4.75	rain	0.87	Overcast	False	14.653
2	2013	1	1	Friday	2	13.45	260.0	10.20	8.10	1036.95	10.20	4.41	rain	0.87	Overcast	False	10.838
3	2013	1	1	Friday	3	13.50	261.0	9.82	7.67	1037.02	7.54	4.56	rain	0.86	Overcast	False	10.626
4	2013	1	1	Friday	4	14.00	262.0	9.76	7.61	1037.03	7.29	4.97	rain	0.86	Overcast	False	10.387

\* A modo de ejemplo se muestran el detalle de los primeros 5 filas del dataset

En el listado de variables independientes o predictoras tenemos:

- Temporales o estacionales: año, mes, semana, día de la semana, hora.
- Climatológicas: visibilidad, sentido del viento, velocidad del viento, temperatura, punto de rocío, presión, sensación térmica, tipo de precipitación, humedad, resumen climatológico.
- Otras variables: día feriado.

En este primer análisis descartaremos las variables relacionadas al perfil del consumidor (Acorn y Acorn grouped).

Como resultado final del proceso tenemos una tabla con 10.153 registros y 113 dimensiones o columnas, incluida la variable target.



### **3. Evaluación de modelos de Machine Learning.**

Una vez que preparamos el dataset, estandarizamos o normalizamos sus variables, creamos dummies y sepáramos los sets de entrenamiento y de validación o testeo. El paso siguiente es evaluar distintos modelos de Machine Learning para la resolución del problema.

Sucesivamente iremos probando distintas variantes dentro de distintas familias de modelos, evaluando su performance en los sets de entrenamiento y validación, y evaluando en ambos casos las principales métricas de rendimiento: error medio absoluto, error medio cuadrático, raíz del error medio cuadrático, y el R2.

**Figura 7: Métricas para modelos de regresión**

Métrica	Nombre	Significado
MAE	Error absoluto medio	Promedio de los errores en términos absolutos.
MSE	Error cuadrático medio	Promedio de los errores al cuadrado.
RSME	Raíz del error cuadrático medio	Raíz cuadrada del promedio de errores al cuadrado.
R2	Coeficiente de determinación	Proporción de la variabilidad de los datos explicada por el modelo (valor máximo = 1).

Para la evaluación además utilizamos un gráfico que muestra o compara los valores reales con los predichos, y otro gráfico que muestra la distribución de los errores de predicción. Llegado el momento explicaremos con mayor grado de detalle su utilidad.

A continuación, los distintos grupos de modelos evaluados.

#### **3.1 Modelos Lineales**

Estos modelos se caracterizan por su facilidad de implementación y por su explicabilidad. En términos generales, suponen una relación lineal entre la variable objetivo y las variables predictoras. Los modelos ajustan una recta a los distintos puntos ubicados en las n dimensiones existentes, aquella recta que minimiza su error cuadrático, esto es la distancia entre la predicción y el valor observado.

Dentro de los modelos lineales evaluamos las siguientes alternativas.

1. Modelo lineal base.
2. Modelo lineal con penalización de norma L1 – Regresión Lasso.
3. Modelo lineal con penalización de norma L2 – Regresión Ridge.
4. Elastic-Net.
5. Modelo lineal aplicando Análisis de Componentes Principales (PCA).
6. Modelo lineal con eliminación recursiva de variables (RFE).



**Figura 8: Rendimiento de los modelos lineales**

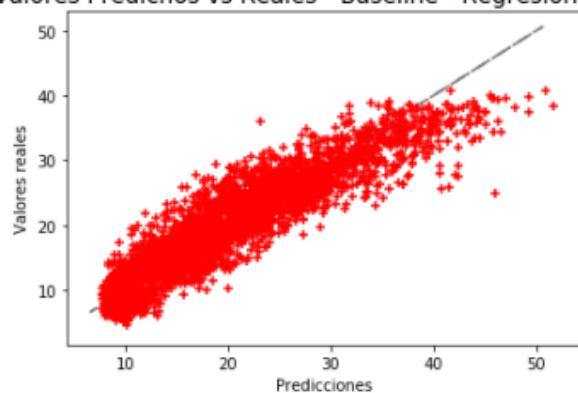
Model	Training MAE	Training MSE	Training RMSE	Training R2	Testing MAE	Testing MSE	Testing RMSE	Testing R2
Baseline - Regresión Lineal	2.61696	11.2299	3.3511	0.835922	2.64751	11.7768	3.43173	0.832197
Regresión Lasso	2.61723	11.23	3.35112	0.83592	2.64768	11.777	3.43176	0.832194
Regresión Ridge	2.61723	11.2298	3.35109	0.835923	2.64762	11.7764	3.43167	0.832203
Elastic-Net	2.61721	11.2304	3.35118	0.835914	2.64777	11.7779	3.43189	0.832182
Regresión Ridge + PCA	5.19007	44.6641	6.68312	0.347419	5.091	44.4301	6.66559	0.366933
Regresión Ridge + RFE	3.84904	24.9262	4.99261	0.635807	9.49063	150.517	12.2685	-1.14466

En esta familia de modelos encontramos una performance bastante aceptable en términos de raíz del error cuadrático medio y de R2, tanto en el set de validación como en el de testeo.

En aquellos modelos que realizan cierto tipo de penalización sobre las variables para evitar el sobreajuste u overfitting (Lasso, Ridge y Elastic-Net) su grado de penalización fue optimizado por validación cruzada, y en ninguno de los casos la penalización es significativa. La performance de los modelos más complejos no difiere de la del modelo base y las métricas no sugieren que se esté produciendo overfitting.

**Figura 9: Validación del modelo de regresión lineal**

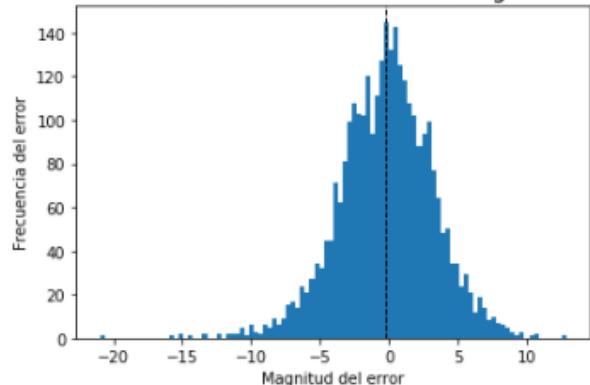
Valores Predichos vs Reales - Baseline - Regresión Lineal



A la izquierda podemos ver los gráficos previamente mencionados, referidos al rendimiento de nuestro modelo lineal base en el set de validación.

El gráfico de arriba muestra la relación entre los valores reales y las predicciones que realiza el modelo. La situación ideal sería que los puntos se ubiquen sobre la recta diagonal, lo cual indicaría que el valor predicho es idéntico al valor real que se está tratando de predecir.

Distribución de los errores - Baseline - Regresión Lineal



En este caso la relación no es perfecta, pero se observa que el modelo puede capturar algo de la esencia de cómo se comporta nuestra variable objetivo respecto a las variables predictoras o independientes. La métrica del R2 (0.83 en este modelo base) resume esta capacidad, la cual definimos como un valor más que aceptable.

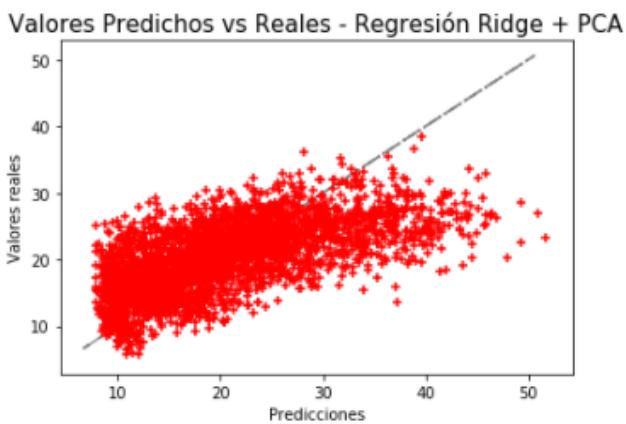


El gráfico de la parte inferior muestra la distribución de los errores del modelo. Podemos observar que el error promedio es cercano a cero y la mayoría de los errores se encuentra en el intervalo [-5,5]. La métrica de RMSE con un resultado de 3.43 es un buen indicador. Obviamos los gráficos correspondientes a los otros modelos lineales dado que su performance es similar.

Sí haremos un comentario sobre los últimos dos modelos implementados. Para la regresión Ridge + PCA se realizó un análisis de componentes principales para detectar las 30 variables más importantes a la hora de explicar la variabilidad de los datos. Tomamos esas 30 variables y las utilizamos para intentar predecir el consumo de energía. Por otro lado, para la regresión Ridge + RFE realizamos un proceso de eliminación recursiva de features. En este proceso, se comienza implementando el modelo con la totalidad de las variables y luego se van eliminando de manera sucesiva por su (baja) significancia, hasta llegar al número deseado. En este caso, 30 variables.

En estos modelos la performance decae de manera significativa. Entendemos que ello se debe a que en ambos casos estamos eliminando variables importantes: días, meses, años, factores climáticos, y ello no es posible sin perder capacidad predictiva. Esta caída en la performance ocurre tanto en el set de entrenamiento como en el de testeo, por lo cual podemos concluir que todas las variables resultan significativas en mayor o menor medida, y no es posible eliminar o simplificar las n dimensiones sin que el modelo pierda robustez.

**Figura 10: Validación del modelo Ridge + PCA**



A modo ilustrativo, para contrastar con lo que ocurría con el resto de los modelos lineales, mostramos los gráficos correspondientes a la regresión Ridge + PCA.

Podemos observar cómo la simplificación o reducción en la dimensionalidad del dataset llevó a una peor performance del modelo. Los puntos no se acercan a la recta como ocurría anteriormente, e incluso están más dispersos.

Lo mismo puede decirse sobre la distribución de los errores del modelo. La dispersión en torno a la media es mucho mayor, lo cual se puede corroborar a partir de las métricas expuestas anteriormente.

Un comportamiento similar se observa para el modelo Ridge + RFE., que obviamos presentar.



### 3.2 Modelos de K-vecinos

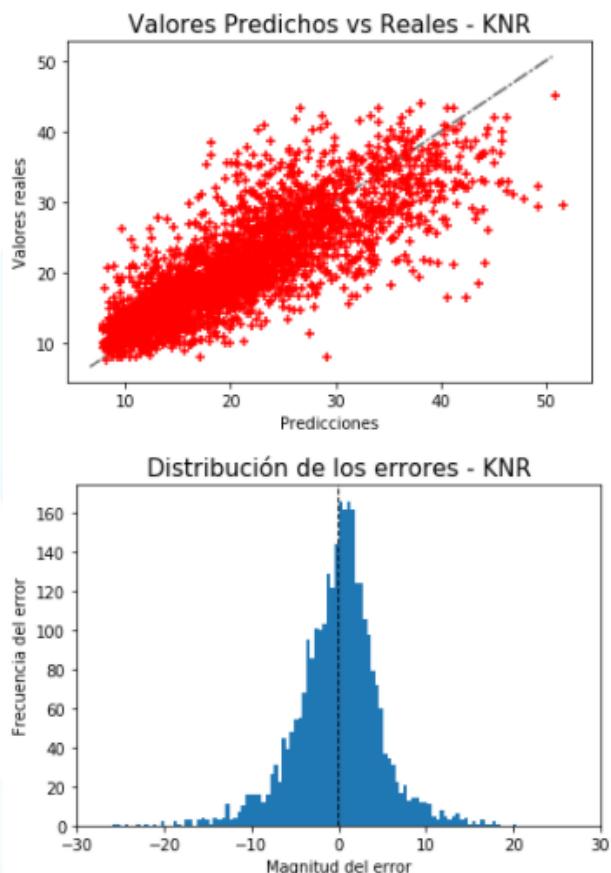
Estos modelos , utilizados tanto para problemas de clasificación como de regresión, tratan de predecir en función de la ubicación espacial de los puntos en las n dimensiones, y predicen los valores en función de una medida de cercanía o distancia entre los puntos existentes.

Los modelos de K-vecinos operan minimizando esta medida de distancia, y no son paramétricos, en el sentido de que no intentan optimizar un parámetro determinado o función de costo, sino que guardan en memoria la ubicación de cada punto en el espacio n-dimensional, y asocian los nuevos puntos según su cercanía a los puntos ya “aprendidos”. Por ello son susceptibles de caer en el sobreajuste u overfitting, su entrenamiento con cierto grupo de datos puede volverse obsoleto cuando se presentan casos no vistos anteriormente.

Figura 11: Rendimiento de los modelos de K-Vecinos

Model	Training MAE	Training MSE	Training RMSE	Training R2	Testing MAE	Testing MSE	Testing RMSE	Testing R2
KNR	0	0	0	1	3.52563	23.5681	4.8547	0.664187
KNR + PCA	0	0	0	1	3.78686	27.1611	5.21163	0.612992

Figura 12: Validación del modelo KNR



Con este ejercicio logramos comprobar la característica de los modelos de K-vecinos: el rendimiento en el set de entrenamiento es perfecto, literalmente de error cero, pero en el set de validación la performance decae. No es un mal resultado, pero está por debajo del modelo lineal más básico, y el poder de cómputo que requiere para su procesamiento es muchísimo mayor. Cabe aclarar que la cantidad de vecinos y la medida de distancia fueron optimizadas a través de **validación cruzada** y **GridSearch**.

La alternativa que exploramos fue aplicar este modelo en el caso del set simplificado por PCA. En este caso, la performance obtenida fue mejor que la de su equivalente lineal, y no difiere mucho del rendimiento de KNR con el set completo. No obstante, el poder de cómputo requerido por el modelo lo sigue haciendo poco eficiente.



### 3.3 Modelos de árbol

Los modelos basados en árboles de decisión tienen distintas variantes: de ensamble, random forests, modelos adaptativos, etc. En esta sección probaremos algunas de esas alternativas.

Si bien el pormenor conceptual de estos modelos escapa el alcance de este trabajo, mencionaremos lo mínimo indispensable en cada caso. En términos generales, podemos mencionar que todos ellos se basan en la estructura de un árbol de decisión.

**Figura 13: Árboles en problemas de regresión**

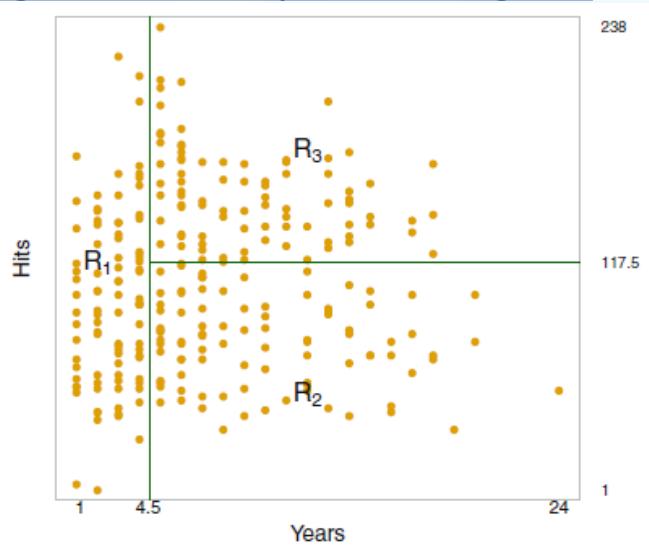


Imagen extraída de “An Introduction To Statistical Learning” de G. James, D. Witten, T. Hastie y R. Tibshirani. Capítulo 8, pág. 316

En el caso de los problemas de regresión, los modelos toman el valor de la feature más significativa, aquella que mejor divide el cuerpo de datos, parten el espacio de datos según ese criterio y repiten el proceso generando nuevas regiones que se vuelven a dividir entre sí. De esta manera el árbol va generando nodos de los cuales se desprenden nuevas ramas que a su vez se convierten en nodos de otras ramas.

Este proceso se repite de manera sistemática hasta que la ganancia de realizar la división es menor a su “costo”. O bien, se especifica en sus hiperparámetros la cantidad de niveles que debería tener (técnica conocida como *pruning*).

Las variantes más sencillas de estos modelos son fácilmente interpretables, pero cuentan con un problema: el overfitting. El criterio que se utiliza para realizar la división de los datos es específico del set de entrenamiento, y puede no ser representativo para nuevas observaciones, por lo cual puede que no sean del todo efectivos para la generalización en otro conjunto de datos.

**Figura 14: Rendimiento de los modelos de árbol**

Model	Training MAE	Training MSE	Training RMSE	Training R2	Testing MAE	Testing MSE	Testing RMSE	Testing R2
Bagging	0.927976	1.80322	1.34284	0.973653	2.31534	10.0722	3.17367	0.856485
Random Forest	0.818034	1.25048	1.11825	0.981729	2.17818	9.07353	3.01223	0.870715
ExtraTrees	9.59232e-12	1.59983e-19	3.99978e-10	1	2.11842	8.6523	2.94148	0.876717
Gradient Boosting	2.55085	11.2296	3.35106	0.835926	2.63102	12.6279	3.55358	0.820069
AdaBoost	5.45556	42.0472	6.48438	0.385654	5.42686	42.0255	6.48271	0.401195
LightGBM	1.70728	5.01005	2.23832	0.926799	2.10947	8.13903	2.8529	0.88403
CatBoost	1.46727	3.58854	1.89434	0.947568	1.98361	7.17251	2.67815	0.897802

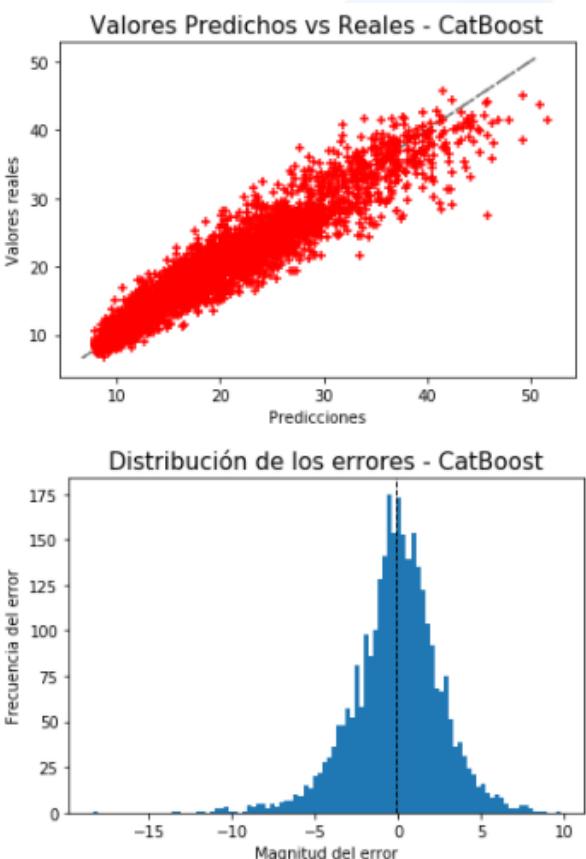


Como respuesta surgen los **modelos de ensamble**. Estos toman una serie de árboles (pueden ser cientos o miles) que, partiendo de distintos subsets de datos, se agregan o promedian para llegar a un único resultado final. Estos subsets de datos a su vez pueden ser reales y seleccionados al azar, o bien generados artificialmente (técnica conocida como bootstrapping). Tomar distintos sets de datos y distintas variables para hacer las agrupaciones reduce entonces el potencial overfitting del modelo, alcanzando un resultado que puede ser generalizable. En este grupo tenemos el modelo de **Bagging**.

Un paso adicional lo dan los **Random Forests**, que además de tomar subconjunto de datos para cada árbol, toman un subset de features para hacer la partición en cada nodo en cada árbol. Esto le da mayor robustez al modelo, ya que si existe un feature que prevalece sobre los demás, los árboles tenderán a parecerse entre sí. De esta manera se evita este problema. El modelo **ExtraTrees** es una variante dentro de este grupo.

Finalmente tenemos los **modelos basados en boosting**. Esto agregan un condimento adicional: además de hacer esta selección de datos y variables, readaptan los árboles entrenados a nuevos datos y nuevas variables de manera incremental, perfeccionando el modelo. Dentro de este grupo tenemos **Gradient Boosting**, **AdaBoost** y llegamos a las variantes más complejas, pero rápidamente optimizables, como son **LightGBM** y **CatBoost**.

**Figura 15: Validación del modelo CatBoost**



Tal como se observa en la [figura 14](#), en todos estos casos, la performance tanto sobre el set de entrenamiento como en el de validación fueron muy buenas, e incluso superiores a los modelos lineales (salvo el caso de AdaBoost), y ello en términos de R2 como en las métricas de error.

En los gráficos de la izquierda se puede analizar la performance de nuestro mejor modelo, logrado a través de CatBoost, con un R2 de 0.897 y un RMSE de 2.67.

Comparándolo con el rendimiento de nuestro mejor modelo lineal, observamos que su capacidad para predecir valores es superior, sobre todo en los casos de menor consumo, y el desvío observado en los de mayor consumo es menor (aunque imperfecto). En términos de error, la distribución se observa más acotada que el primer caso analizado, tal como reflejan las métricas.



### 3.4 Comparativa

A continuación mostramos el cuadro que resume la totalidad de las performances de los modelos, a partir de las métricas de RMSE y R2. La línea puntuada representa la mejor marca en ambas métricas

**Figura 16: Comparativa de modelos por score R2 y RMSE**

Tabla de Scores Final - R2 en set de validación

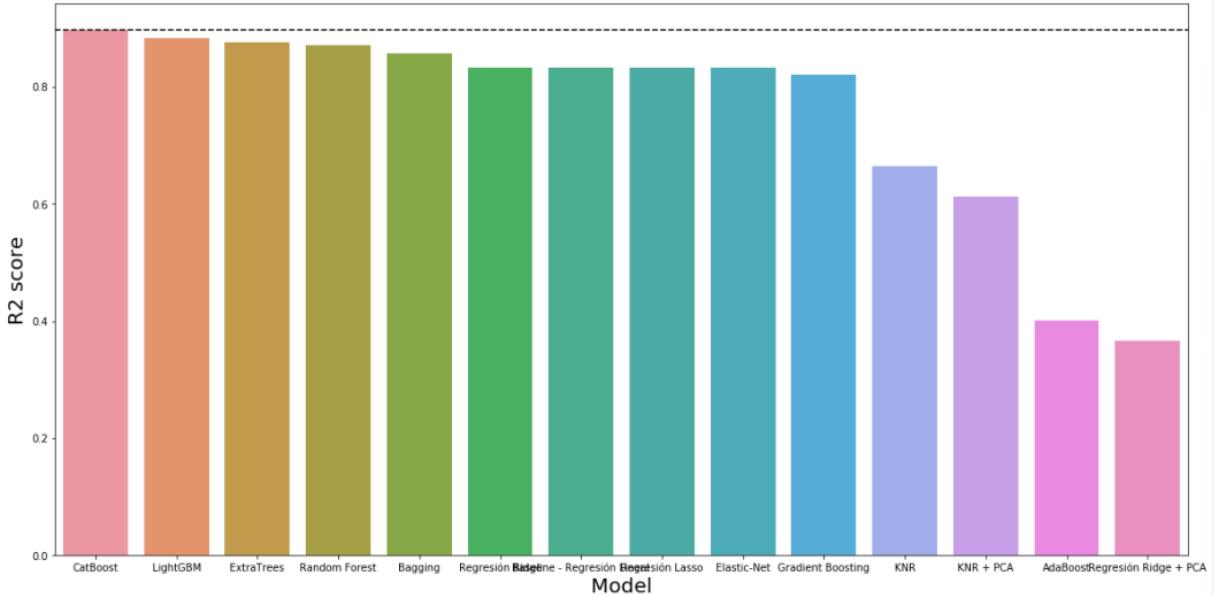
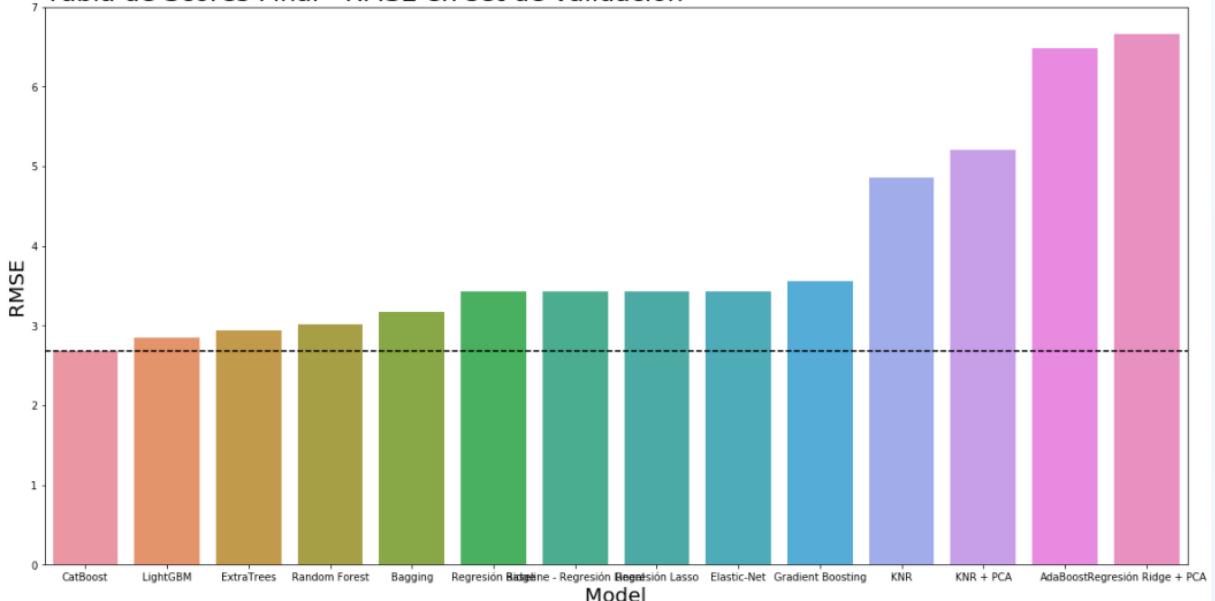


Tabla de Scores Final - RMSE en set de validación



Dados los resultados observados, la velocidad de cómputo y la facilidad de implementación, **tomamos a CatBoost como el modelo óptimo para la resolución del problema**. No obstante de ello, nunca debemos perder de vista el problema de su interpretabilidad, diríamos que incluso es clave para su implementación efectiva en el negocio, de acuerdo a lo establecido por los autores Marco Tulio Ribeiro, Sameer Singh y Carlos Guestrin<sup>1</sup>.

<sup>1</sup> Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin: '[Why Should I Trust You?](#)': Explaining the Predictions of Any Classifier.

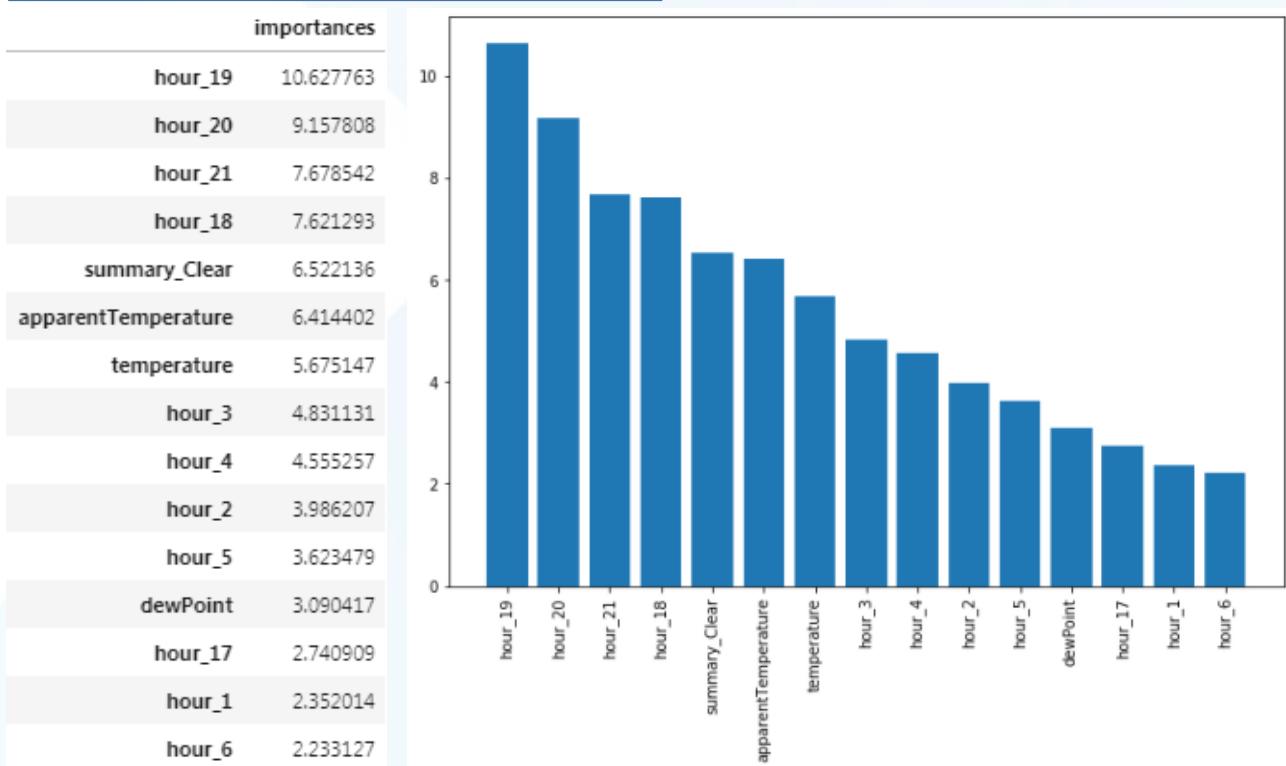


### **3.5 Importancia de las variables**

Un primer paso para reducir esa barrera de la intrepretabilidad es indagar sobre la importancia de las features o variables del modelo. Los modelos de árbol tienen la facilidad de poder brindarnos esa información, aquellas variables que mejor sirven para segmentar a los datos modelados.

A continuación mostramos las veinte principales variables que tiene nuestro modelo para la realización de las predicciones.

**Figura 17: Modelo CatBoost - Feature importance**



Podemos realizar los siguientes comentarios:

- La importancia que tiene la franja horaria de 18 a 21 hs. Entendemos que ello puede deberse a que es el horario donde se produce el pico de consumo, ya que las familias vuelven a sus hogares y además la temperatura empieza a bajar.
- En segundo lugar vienen dos variables climáticas: el clima despejado y la sensación térmica. Por las razones descriptas en el análisis preliminar, entendemos la relación que existe entre clima y temperatura respecto al consumo de energía eléctrica.
- En un tercer escalon viene la franja horaria de menor consumo.



## 4. Escalabilidad del modelo

Una vez definido el modelo óptimo para la resolución del problema cabe hacerse la siguiente pregunta: ¿Qué tan válido puede ser para predecir valores nunca vistos? ¿La relación que aprendió el modelo entre las variables dependientes y la variable objetivo es válida para generalizar o sólo es útil para los datos seleccionados? ¿Estamos ante un caso de overfitting?

Otra cuestión a considerar es que, de la manera que procedimos, al agrupar la información de una única manzana, estamos desecharando todo el valor que podrían aportar los datos socioeconómicos para predecir el consumo de energía. ¿Es entonces el modelo susceptible de ser mejorado? A continuación exploraremos la primera de las preguntas:

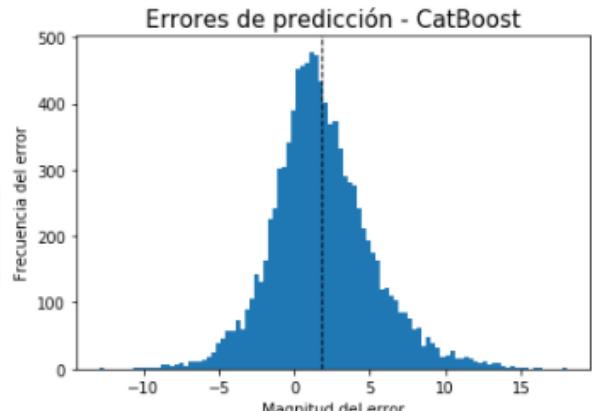
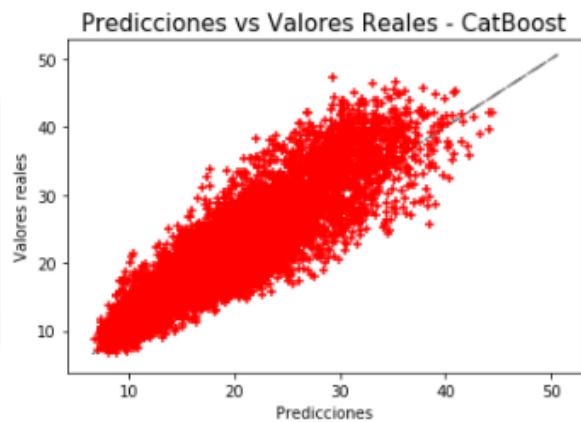
### 4.1 Aplicación del modelo a un nuevo set de datos

Una primera forma de indagar en esta cuestión es tomar al azar un .csv, realizar las mismas transformaciones que se hicieron para el set de entrenamiento, e intentar predecir sus valores. A continuación mostramos los resultados obtenidos en una manzana tomada al azar (**block 44**)

**Figura 18: Validación del modelo CatBoost para nuevos datos**

```
model_evaluation(y_test_2, cat_testing_predictions_2)
MAE: 2.854483345498464
MSE: 14.29563670934513
RMSE: 3.7809571155125696
R^2: 0.6971072984410509
```

```
model_graph(y_test_2, cat_testing_predictions_2, 'CatBoost')
```



Observamos el rendimiento del algoritmo decayó un poco. Concretamente, el R2 se redujo en un 22%, mientras que el RMSE se incrementó en un 41%.

No obstante, debemos decir que esta caída en la performance es esperable. Después de todo, las características sociodemográficas varían de un caso al otro. Incluso podemos estas obviando otro tipo de variables que pueden ser significativas para el fenómeno.

Dado que la cuestión no es menor y requiere un mayor análisis, haremos un breve paréntesis para estudiar con mayor profundidad la relación entre el consumo y las distintas manzanas, así como del consumo y los distintos Acorn existentes, ya no de manera agregada y como una serie de tiempo como lo hicimos anteriormente, sino tomando las mediciones cada una hora e independientemente de su relación temporal.



## 4.2 Estudio del consumo de energía por Clustering

Para esta parte aplicamos algoritmos de aprendizaje no supervisado. A diferencia de los algoritmos de aprendizaje supervisado que buscan asignar etiquetas o valores a las observaciones en función de lo aprendido -y que dichas predicciones pueden ser correctas o incorrectas- estos algoritmos buscan aprender de la relación subyacente entre los distintos registros de datos, sin ser necesariamente correcta o incorrecta. Anteriormente, cuando aplicamos el algoritmo de PCA (principal component analysis) para reducir la dimensionalidad del dataset, estábamos aplicando este tipo de modelos sin comentarlo.

En este caso, sin entrar en mayores detalles técnicos, aplicaremos el algoritmo K-Means. Este utiliza una medida de distancia y en una cantidad K (predefinida) de vecinos para determinar si un punto pertenece o no a una agrupación o clúster. Es decir que el algoritmo recorre la totalidad de los datos y los va asignando a distintas agrupaciones en base a esa medida de distancia. De esta manera, el conjunto de observaciones queda asignado a una cantidad K de clústeres, donde para cada observación la distancia respecto al centro de su clúster es óptima en relación a la distancia respecto a los centroides de las otras agrupaciones.

### Clustering de consumo por Acorn

Figura 19: Métricas de similaridad para Clustering por Acorn

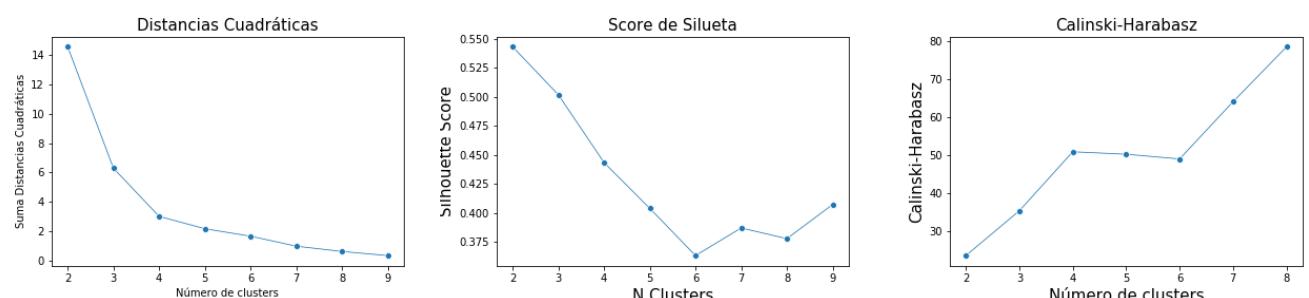
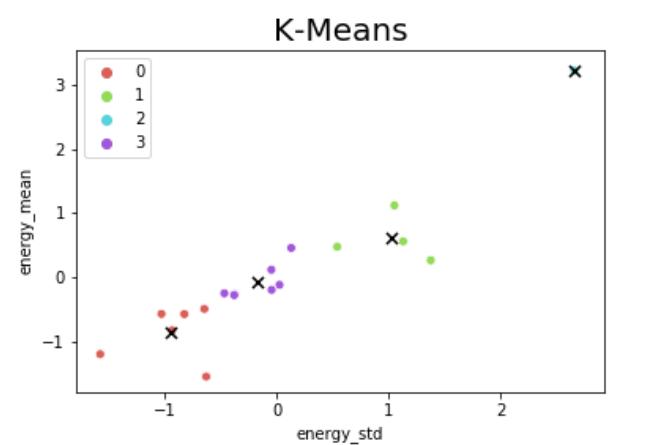


Figura 20: K-Means para Clustering por Acorn

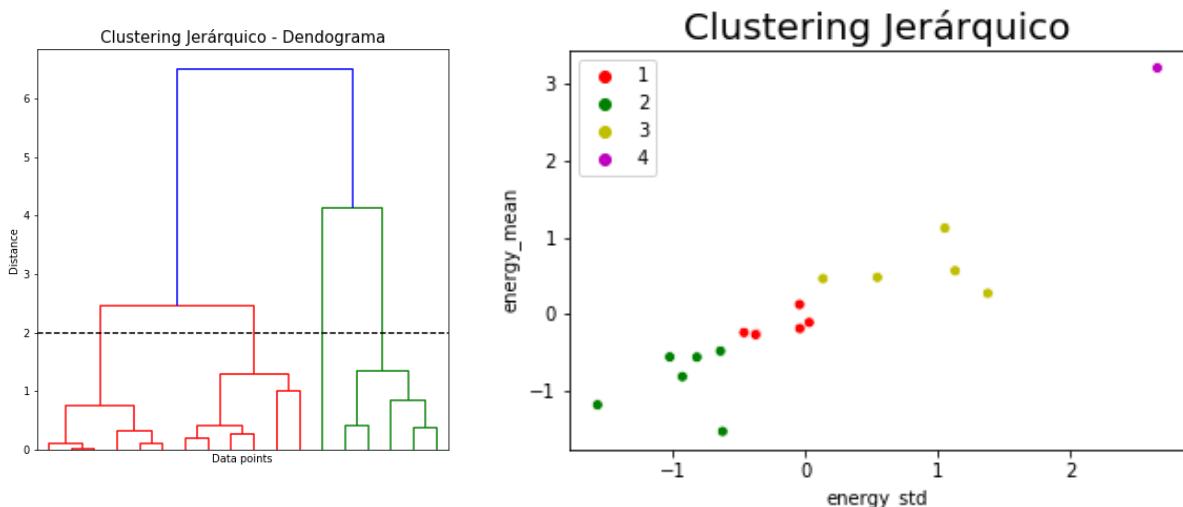


En una primera instancia, aplicamos K-Means al consumo diario de los distintos Acorn, más precisamente sobre dos variables: energía total y variabilidad ('energy\_mean' y 'energy\_std'). Para ello, además del trabajo de preprocessamiento necesario de normalizar las variables (el algoritmo se basa en una medida de distancia, por lo cual es totalmente necesario), buscamos optimizar la cantidad de clústeres a través de las métricas distancia cuadrática, score de silueta y Calinski-Harabasz.



Luego de aplicar el algoritmo K-means utilizamos otra herramienta, **Clustering Jerárquico**, con el que obtuvimos resultados similares.

**Figura 21: Clustering Jerárquico por Acorn**



Sobre la izquierda podemos ver un dendrograma, que muestra cómo se van particionando los datos hasta llegar a las agrupaciones de menor orden. Al nivel de umbral elegido, podemos observar cuatro clústeres distintos, lo mismo que pudimos observar con K-Means.

**Figura 22: Composición de los Clústeres de Consumo por Acorn**

Acorn	energy_std	energy_mean	cluster	Acorn_grouped
4 ACORN-E	0.182968	0.217671	1	Rising
6 ACORN-G	0.180855	0.213401	1	Comfortable
7 ACORN-H	0.180787	0.230448	1	Comfortable
12 ACORN-M	0.170734	0.209023	1	Stretched
10 ACORN-K	0.168083	0.210506	1	Stretched
11 ACORN-L	0.170871	0.209796	1	Stretched
8 ACORN-I	0.162683	0.197343	2	Comfortable
15 ACORN-P	0.163179	0.140144	2	Adversity
14 ACORN-O	0.153973	0.179153	2	Adversity
13 ACORN-N	0.157285	0.192991	2	Stretched
5 ACORN-F	0.151094	0.193131	2	Comfortable
16 ACORN-Q	0.134552	0.159074	2	Adversity
17 ACORN-U	0.216370	0.254442	3	NP_Household
3 ACORN-D	0.214019	0.284808	3	Rising
2 ACORN-C	0.198575	0.249801	3	Affluent
1 ACORN-B	0.186137	0.248845	3	Affluent
9 ACORN-J	0.223836	0.238423	3	Comfortable
0 ACORN-A	0.262627	0.398735	4	Affluent

Si bien nuestro primer análisis (unidimensional, en términos de consumo promedio) sólo destacaba la superioridad del consumo de los Acorn de mayor poder adquisitivo, ampliar el análisis sobre el consumo promedio y su variabilidad nos da otro tipo de respuesta, más compleja.

Observamos dos grandes grupos, que a su vez se dividen en otros dos grupos respectivamente. Y dentro del clúster de mayor consumo, encontramos únicamente (y muy por sobre el resto) al Acorn A, con mayor consumo diario y mayor variabilidad en el consumo diario que el resto.

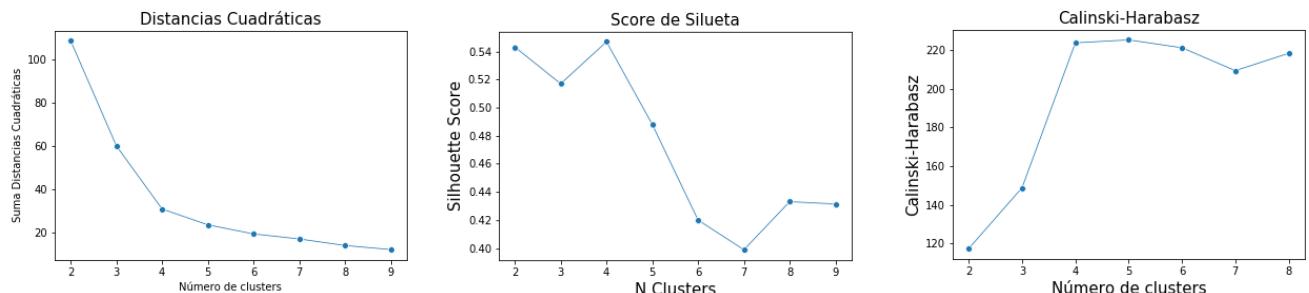
En los clústeres de menor consumo observamos una mezcla de Acorns, lo que apunta a que la relación en realidad no es tan lineal como suponíamos.



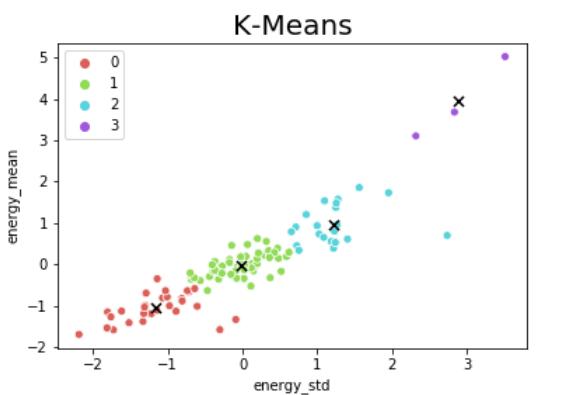
## Clustering de consumo por block

Cuando realizamos el mismo análisis pero a nivel de block o manzana, la respuesta es igual de contundente. Encontramos cuatro clústeres distintos, de los cuales uno es muy superior al resto. El procedimiento para optimizar los hiperparámetros fue el mismo que en el primer caso.

**Figura 23: Métricas de similaridad para Clustering por block**

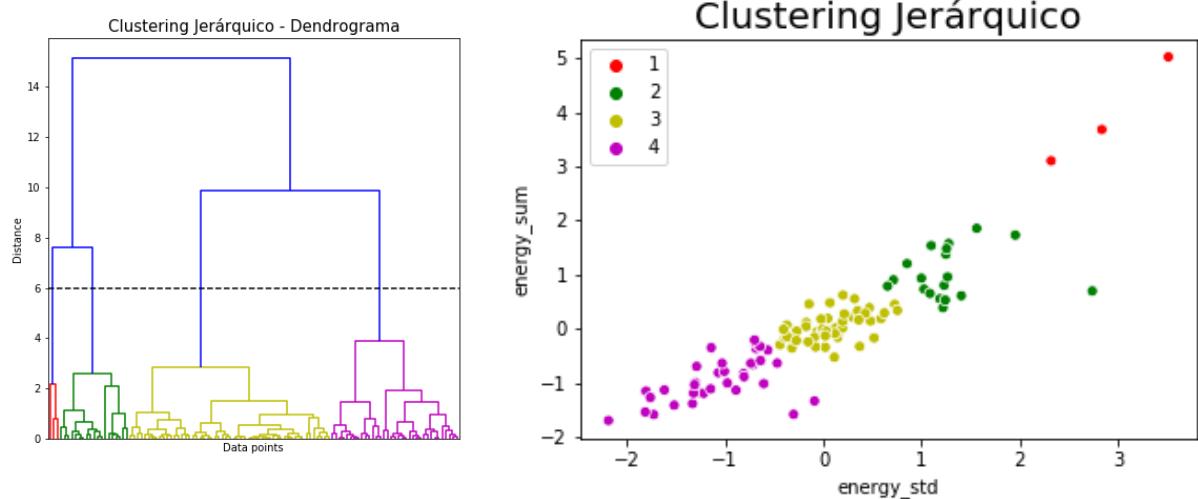


**Figura 24: K-Means para Clustering por block**



Los resultados observados son en cierta forma consistentes con los observados en el clustering por Acorn. Las manzanas de mayor poder adquisitivo son las que presentan un mayor consumo, y una mayor variabilidad en el consumo diario. No obstante, en el resto de los clústeres, la relación entre clúster y Acorn se vuelve más difusa: no es una relación lineal, y no se observa que una mejor condición económica implique necesariamente mayor consumo.

**Figura 25: Clustering Jerárquico por block**



Utilizaremos esta información del agrupamiento por consumo y variabilidad diarios por manzana para aplicar nuestro modelo predictivo.



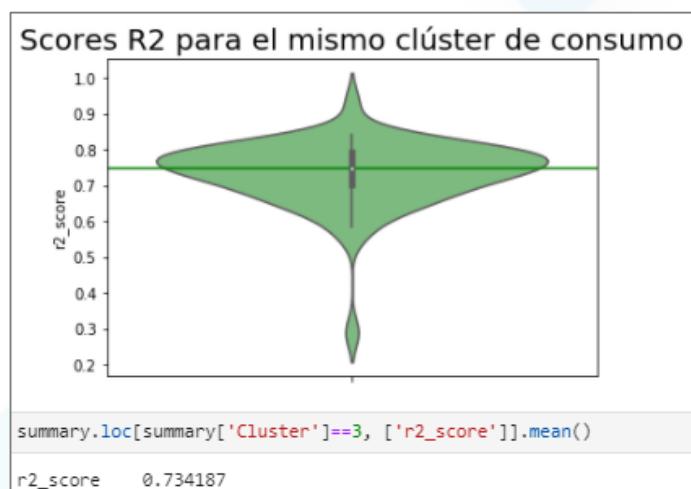
#### 4.3 Aplicación del modelo en condiciones similares

Una vez estudiada con mayor detalle como se comporta el consumo diario por block, tomaremos esa información para identificar candidatos para realizar nuestras predicciones.

Tomaremos entonces aquellos usuarios que se encuentren dentro del mismo clúster de consumo diario que los que se utilizaron para entrenar el modelo. Sabemos que el cuerpo de datos se comporta de manera similar, al menos en lo que se refiere al consumo diario por lo que a priori debería tener una performance aceptable.

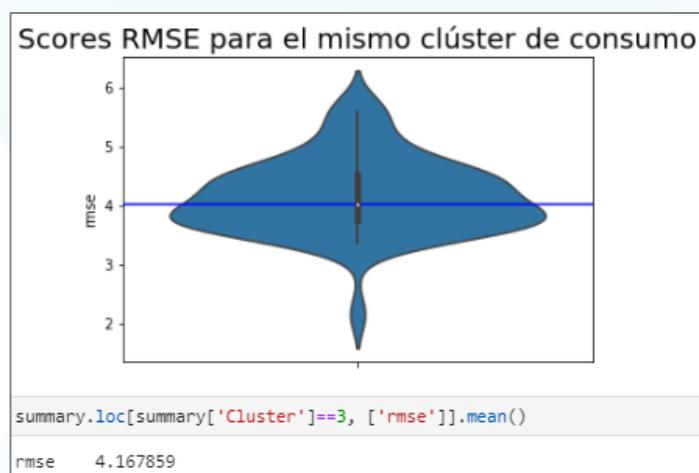
Procedemos entonces de la siguiente manera: tomaremos los archivos de esos blocks, realizaremos las mismas tareas de preprocesamiento y data wrangling, utilizaremos todos los datos para predecir los valores, y evaluaremos el desempeño de nuestro modelo a partir de sus métricas.

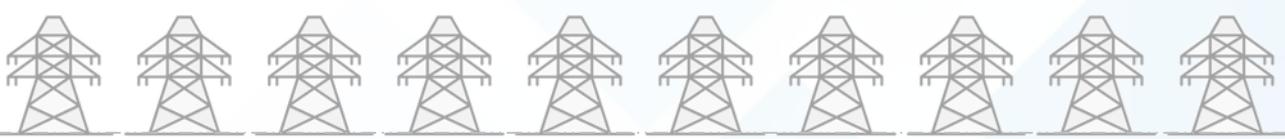
Figura 26: Distribución de los scores R2 para el mismo clúster de consumo.



Observamos que la performance es especialmente buena en algunos casos, pero en otros decae de manera significativa. En los extremos tenemos casos donde el modelo puede predecir de manera perfecta, mientras que en otros el rendimiento es muy pobre, por debajo del 0.3.

El resultado promedio es un R2 de 0,73, denotado por la línea azul. Esta es una medida que consideramos más que aceptable en esta instancia.

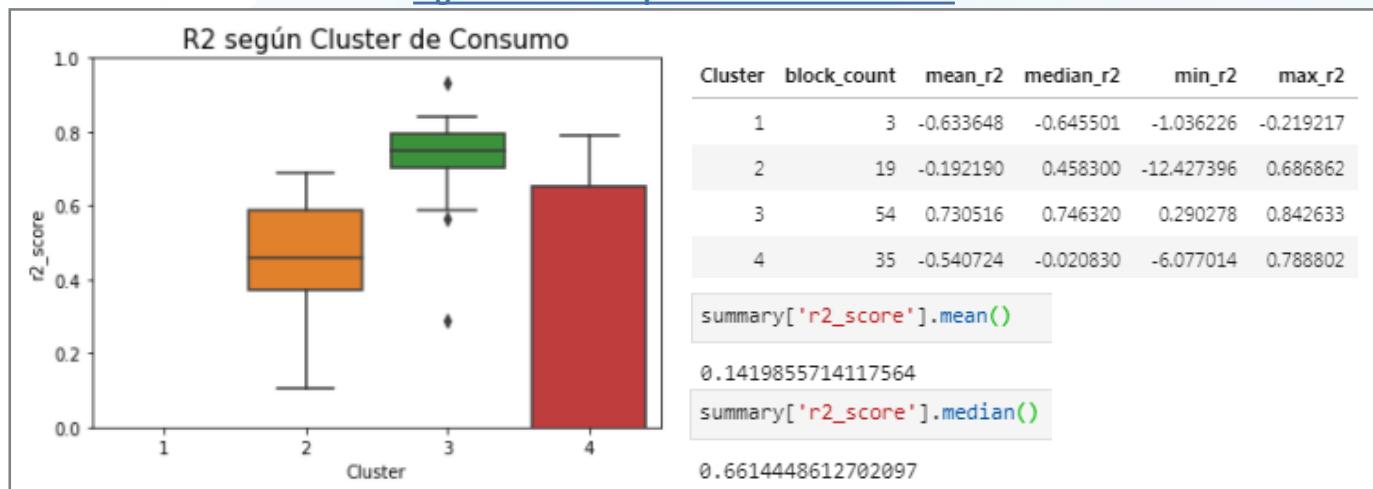




#### 4.4 Aplicación del modelo en todo el universo de datos

Una vez analizado el rendimiento en los datos de características similares a priori (cabe destacar que el consumo promedio diario y su variabilidad no es equivalente a observar el consumo por hora), aplicamos el modelo sistemáticamente sobre todo el universo de datos. Este procesamiento es altamente demandante en términos de cómputo, pero nos sirve para evaluar la validez general de nuestro modelo.

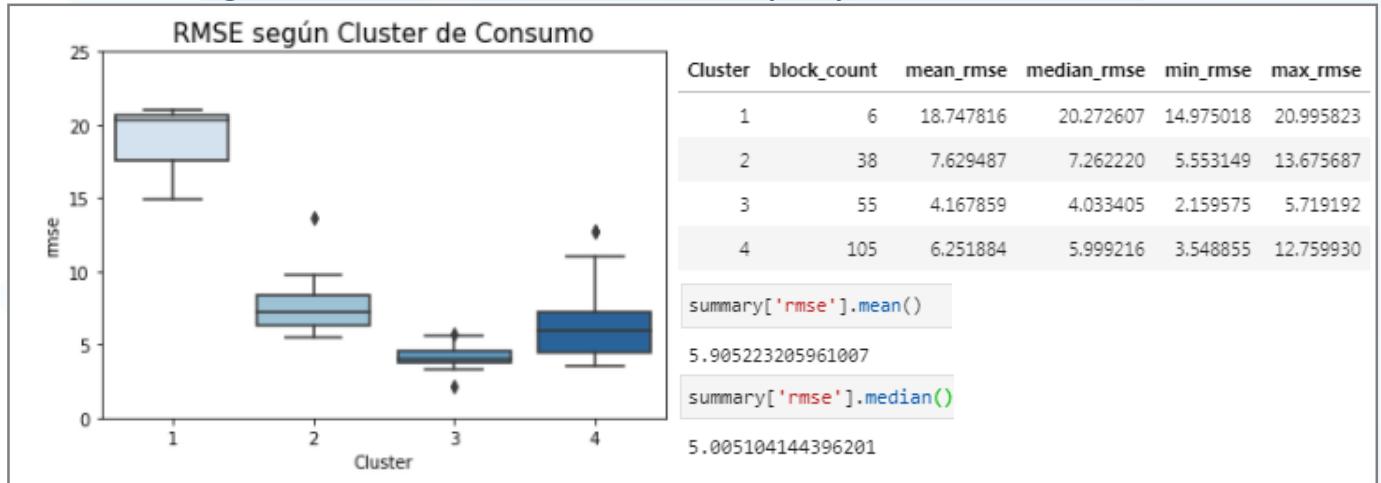
**Figura 26: Scores por clúster de consumo**



De nuevo, los resultados obtenidos son interesantes. Observamos que el modelo puede lograr una buena performance en blocks pertenecientes a otros clústeres de consumo, incluso con rendimiento similar al inicial. Basta con ver el rendimiento máximo en los clústeres 2 y 4 (el clúster 1 es claramente muy diferente). No obstante, en otros casos el resultado es tan pobre que echa por tierra la performance promedio. El R2 promedio es de 0.14, mientras que la mediana es de 0.66. Esto es una señal de cómo los blocks “atípicos” para afectan la predicción general.

Pero no todo está perdido. Recordemos que cuando formulamos este modelo, tomamos al azar un único block, y además descartamos todos los datos asociados a su condición socioeconómica. Entonces **hay muchas oportunidades para enriquecer el modelo y hacerlo más adaptable a los distintos escenarios posibles**.

**Figura 27: Distribución de los scores RMSE para por clúster de consumo**





## **5. Optimización: generación de un nuevo modelo**

A continuación probamos distintas alternativas. En primer lugar cabe hacer una aclaración: nuestro modelo inicial estaba entrenado con un total de 7.107 registros y 113 dimensiones o variables. Al incorporar más blocks la cantidad de registros para el entrenamiento se incrementa exponencialmente. Por otra parte, para los modelos más complejos que pasaremos a evaluar tomaremos nuevas variables que serán transformadas en dummy, por lo cual también se incrementa la dimensionalidad del dataset.

Para poder resolver este inconveniente en términos de procesamiento requerido y poder evaluar la capacidad predictiva de nuestros modelos procedimos de la siguiente manera:

1. Seleccionamos de manera aleatoria una muestra de archivos, de acuerdo a ciertos criterios predefinidos (totalmente al azar, o segmentando la población en distintos estratos)
2. Realizamos las mismas transformaciones que hicimos anteriormente. En aquellos casos donde se incorporan nuevas variables, las mismas se incorporan como dummy.
3. Con el nuevo cuerpo de datos entrenamos un modelo CatBoost y probamos su rendimiento en el set de validación.
4. Con el modelo entrenado y validado, de una manera iterativa y secuencial probamos el rendimiento sobre cada uno de los archivos .csv, de a uno a la vez, y guardamos las métricas obtenidas (R2 y RMSE) en una tabla aparte.

Si bien este procedimiento es engorroso y lleva un tiempo considerable de procesamiento, nos permite tener una visión general de la performance, y poder comparar el rendimiento del modelo en base a distintas hipótesis de trabajo, siempre buscando mejorar el rendimiento base obtenido en el primer acercamiento. Estas variantes del modelo son:

- A) Seleccionar una muestra más grande de archivos para entrenar nuestro modelo.
- B) Seleccionar una muestra de archivos estratificando por Acorn.
- C) Seleccionar una muestra estratificada por Acorn e incorporar la información socioeconómica como variable.
- D) Seleccionar una muestra de archivos estratificando por clúster de consumo e incorporarlo como variable.

La diferencia entre los pasos b y c nos permitirá además evaluar el valor de la información socioeconómica para el modelo. Si tomamos los mismos archivos para el entrenamiento, la diferencia en sus performance (y de por sí, la significatividad de las variables) nos dará un mejor indicio de la importancia de estas variables para la regresión.



## A) Seleccionando una muestra de archivos para entrenar el modelo

Tomamos al azar un total de 18 archivos distintos seleccionados al azar para entrenar nuestro modelo. Realizamos la operación una serie de veces y mostramos los mejores resultados a continuación.

**Figura 28: lista de archivos seleccionados al azar**

block_23	block_67	block_6
block_73	block_78	block_83
block_20	block_15	block_71
block_40	block_19	block_75
block_40	block_3	block_108
block_59	block_104	block_45

En este primer caso, la cantidad de registros de entrenamiento se incrementa de 7.107 a 113.715 (un 1500%), mientras que la cantidad de dimensiones se mantiene constante. El proceso definido en la introducción de la sección nos permite evaluar el rendimiento del modelo sobre la totalidad de archivos en aproximadamente media hora, con el poder de cómputo disponible. A continuación mostramos los resultados obtenidos:

**Figura 29: Scores por clúster de consumo**



Observamos que la incorporación de nuevos archivos para el entrenamiento no logró mejorar la performance general del modelo. En términos generales, nuestra mediana de R2 (para despejar el efecto de los valores atípicamente bajos) bajó ligeramente de un 0.66 a un 0.65, y lo mismo ocurrió con el RMSE, que pasó de 5 a 4.73, aunque en ese sentido es algo positivo.

A nivel clúster, se observa que los resultados obtenidos en los clústeres de consumo 2 y 3 han mejorado, se observa en el boxplot una distribución mucho más compacta de los datos, por lo cual podemos decir que el modelo logró captar mejor el comportamiento de estos grupos. Por otro lado, el rendimiento para el clúster 4 empeoró, y el clúster 1 sigue siendo difícil de predecir. Si observamos la lista de archivos de entrenamiento, ninguno de los blocks pertenecientes a éste último grupo ha sido utilizado, y teniendo en cuenta además su carácter muy distinto al resto en términos de consumo, esto tiene sentido.



## B) Seleccionando una muestra estratificada por Acorn

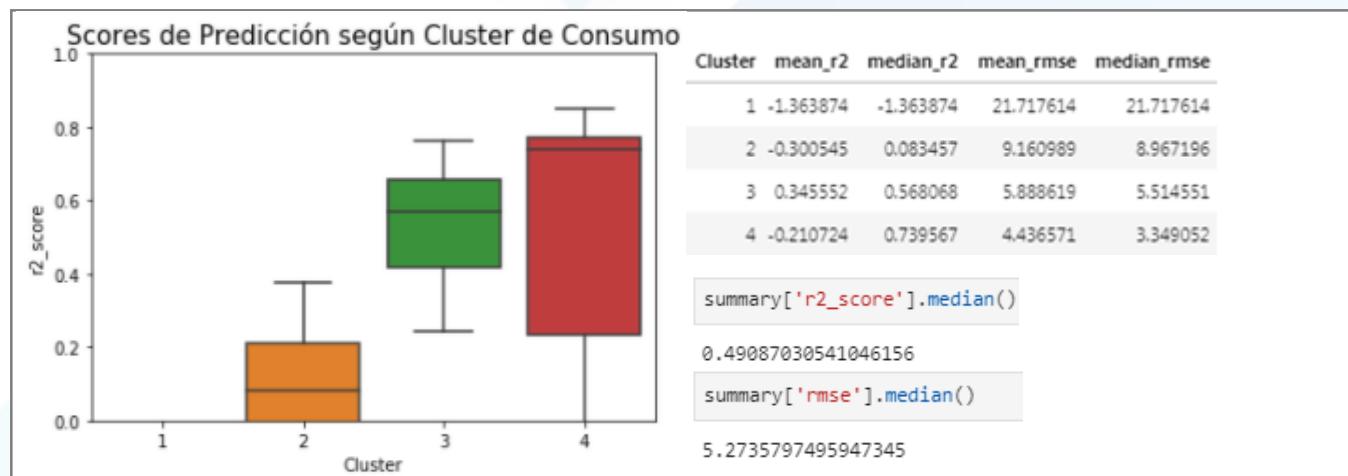
En esta instancia seleccionamos al azar un block por cada Acorn existente, siendo un total de 18 los archivos seleccionados. Mostraremos a continuación los resultados obtenidos en la primera muestra:

**Figura 29: lista de archivos seleccionados al azar**

block_1	block_60	block_85
block_3	block_67	block_88
block_5	block_71	block_90
block_10	block_73	block_92
block_13	block_76	block_97
block_45	block_77	block_110

La cantidad de registros de entrenamiento se incrementa de a 163.474 (un 2200%) y la cantidad de variables permanece constante.

**Figura 30: Scores por clúster de consumo**



Observamos que realizar una selección aleatoria estratificada por Acorn no logró superar el rendimiento del modelo con selección aleatoria pura, e incluso en términos de R2 y RMSE fue inferior al rendimiento del modelo inicial, entrando con un solo archivo. A nivel clúster de consumo, todas las métricas empeoraron, exceptuando al clúster 4, de consumo más bajo, que tuvo una mediana de R2 superior a las anteriores. Los resultados se repitieron en tres nuevos ensayos.

**Figura 31: Evaluación de Resultados en 4 muestras aleatorias**

sample	mean_r2	median_r2	mean_rmse	median_rmse
sample 1	0.146983	0.306873	7.020207	6.669449
sample 2	0.054804	0.134288	7.596109	7.176568
sample 3	0.022628	0.490870	6.268726	5.273580
sample 4	0.154153	0.460203	6.392613	5.679655

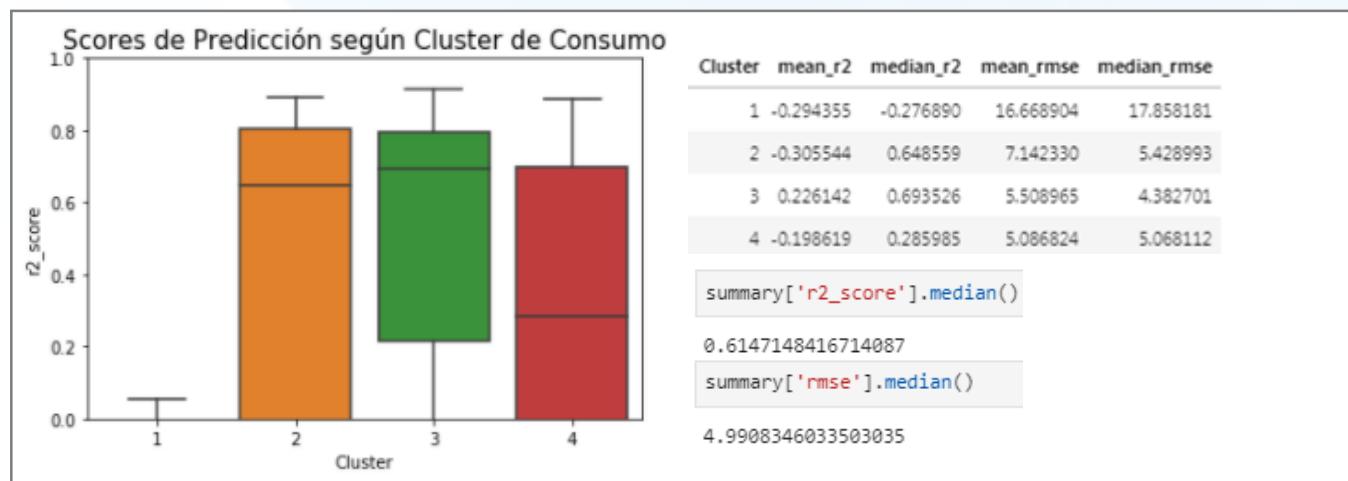


### C) Seleccionando una muestra estratificada y utilizando el Acorn como predictor

Ahora incorporamos el dato del Acorn como una variable predictora más, pasada como dummy. De esta manera, la cantidad de registros de entrenamiento se mantiene igual (163.474), pero la cantidad de dimensiones se incrementa de 113 a 130.

La muestra de archivos seleccionada es la misma que utilizamos en el punto B), para que los resultados puedan ser comparables.

Figura 32: Scores por clúster de consumo



En este caso observamos que incorporar la variable socioeconómica para realizar la regresión permitió mejorar la performance general del modelo, tanto en términos de su R2 como de su error. Lo mismo se observa en todos los clústeres de consumo, excepto para el clúster 4, donde se observa que el R2 promedio mejoró, pero la mediana decayó significativamente. A continuación, los resultados obtenidos en las cuatro muestras anteriores:

Figura 33: Evaluación de Resultados en 4 muestras aleatorias

sample	mean_r2	median_r2	mean_rmse	median_rmse
sample 1	-0.019261	0.421587	6.672226	5.962692
sample 2	-0.010735	0.614715	5.953062	4.990835
sample 3	-0.010735	0.614715	5.953062	4.990835
sample 4	0.036295	0.662424	5.882583	4.532184

A partir de las muestras seleccionadas, podemos decir que en los cuatro casos tanto el R2 como el RMSE mejoran a partir de la inclusión del Acorn como variable, es decir que tenemos un modelo más certero para realizar las predicciones. No obstante, sólo la última de las alternativas pudo alcanzar la capacidad de predicción del modelo inicial y del modelo entrenado con blocks al azar.



## D) Seleccionando una muestra estratificada por clúster de consumo

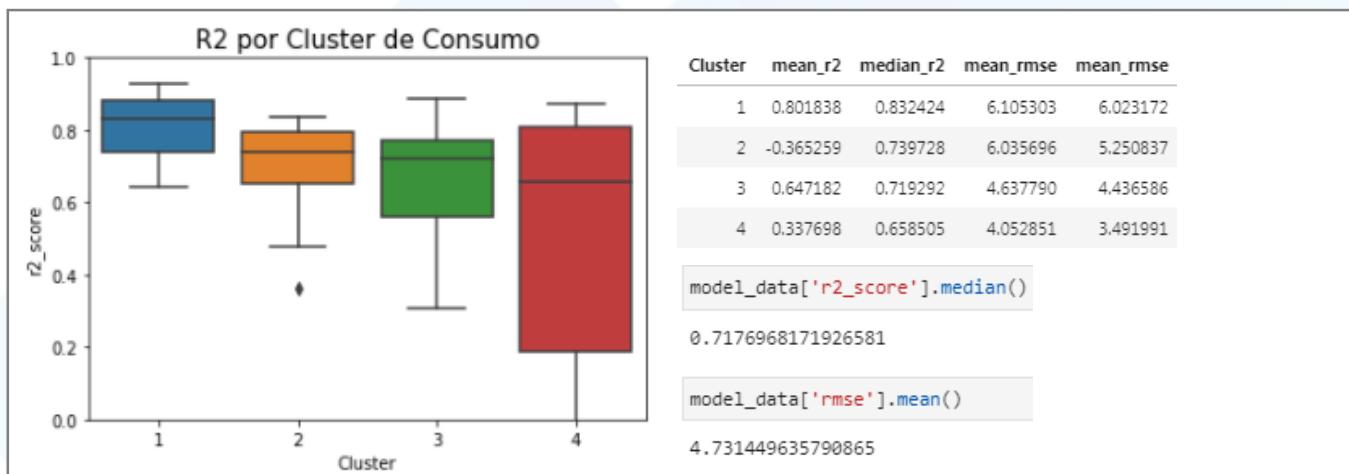
En esta oportunidad descartaremos el dato de Acorn, para incorporar como variable el clúster de consumo al cual pertenece cada block. Dado que la cantidad de clústeres es menor y el tamaño de cada clúster varía, tomamos al azar entre 1/3 y 1/5 de cada uno. Además, incrementamos el tamaño total de la muestra al máximo factible, obteniendo la siguiente distribución.

[Figura 34: lista de archivos seleccionados al azar](#)

	block	cluster		block	cluster		block	cluster
1	block_1	1	9	block_19	3	17	block_43	4
2	block_7	2	10	block_59	3	18	block_109	4
3	block_29	2	11	block_16	3	19	block_105	4
4	block_30	2	12	block_59	3	20	block_105	4
5	block_3	2	13	block_81	3	21	block_90	4
6	block_35	3	14	block_4	3	22	block_55	4
7	block_33	3	15	block_15	3	23	block_107	4
8	block_60	3	16	block_17	3			

Incorporamos como variable el clúster de consumo, por lo cual la cantidad de dimensiones se incrementa a 116.

[Figura 35: Distribución de los scores R2 para por clúster de consumo](#)



Los resultados obtenidos por el modelo superan a todos los anteriores. Si bien la información socioeconómica del consumidor (condensada en su categoría de Acorn) enriquecía el modelo, no es tan potente como introducir la tendencia o estrato de consumo/variabilidad diarios. Por otro lado, el método de muestreo para el entrenamiento del modelo ha mostrado ser más representativo del conjunto, logrando una mejor capacidad predictiva en general.

[Figura 33: Evaluación de Resultados en 4 muestras aleatorias](#)

sample	mean_r2	median_r2	mean_rmse	median_rmse
sample 1	0.382857	0.717697	4.731450	4.370494
sample 2	-1.043637	0.674085	7.200766	5.275988
sample 3	-0.243302	0.634952	5.950958	4.603695
sample 4	-0.017468	0.676285	5.436096	4.317186

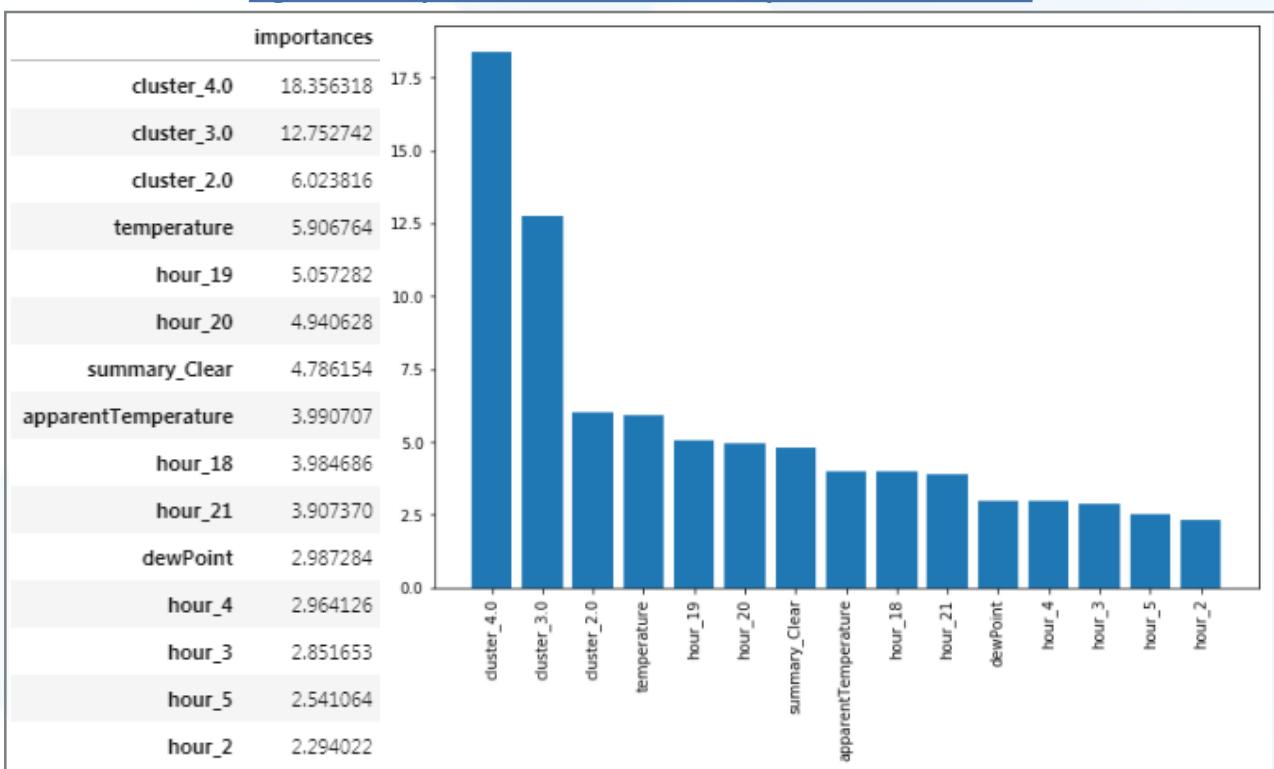


## **6. Conclusión**

A partir de los resultados obtenidos, concluimos esta sección destacando que es factible generar un modelo a que pueda predecir el consumo de un determinado conjunto de usuarios (en este caso un ‘block’ o manzana) en determinado momento en el tiempo. A tales fines, son los modelos basados en árboles de decisión aquellos que brindan un mejor rendimiento en términos de R2, métricas de error y tiempo de procesamiento.

Para el desarrollo de nuestro modelo ha sido necesario incorporar las dimensiones climáticas y temporales, pero son las variables asociadas del perfil de consumo las que tienen una importancia fundamental. Incorporar la información del clúster de consumo promedio diario y su variabilidad (como representativo del perfil de consumidor) aportó un valor diferencial, por encima del que aportaba la información socioeconómica, condensada en el ‘Acorn’.

**Figura 34: Importancia de las variables para el modelo final**



Analizando el *feature importance*, o importancia de las variables, observamos cómo la ubicación en uno u otro clúster es más significativa para la predicción que la temperatura e incluso los horarios pico de consumo.

Nuestros modelos son susceptibles de ser mejorados, fundamentalmente a partir de la inclusión de más datos para el entrenamiento, pero está claro que ello es a costa de una demanda mucho mayor de capacidad de cómputo. Fuera del alcance de este trabajo quedó la implementación de modelos en tecnologías de procesamiento distribuido, que podrían trabajar con todo el cuerpo de datos de una manera más dinámica y potencialmente lograr resultados aún mejores.