Logística de envíos: ¿Cuándo llega?

Práctico: Análisis exploratorio y curación de datos

***Objetivos generales***

* Realizar un estudio exploratorio del dataset para extraer información útil sobre el problema a resolver.
* Desarrollar visión crítica en relación a la problemática para llevar a cabo el procedimiento de ciencia de datos.
* Desarrollar habilidades de comunicación de la información obtenida a partir de los datos de manera clara y sencilla.

***Objetivos específicos***

* Realizar un análisis profundo de los datos para identificar posibles problemas en la información proporcionada (por ejemplo datos duplicados).
* Trabajar con diferentes encodings en las variables categóricas de manera de poder usar ciertos algoritmos de machine learning que solo aceptan datos numéricos.
* Estudiar y analizar de qué manera puede predecirse la cantidad de días en que un envío llegará a destino y qué variables tienen mayor peso en dicha predicción.

***Desarrollo del trabajo***

**Inspección de los datos**

Lo primero que se realizó en este trabajo fue el análisis de las variables con las que cuenta el data set, el cual podemos describir a continuación:

1. el data set tiene un total de 5000 entradas o registros
2. cuenta con variables mencionadas a continuación:

**- Sender\_state, Receiver\_state,Shipment\_type,Service, Status:** son variables categóricas.

**- Sender\_zipcode, Receiver\_zipcode, Quantity, Shipment\_days:** sonvariables enteras.

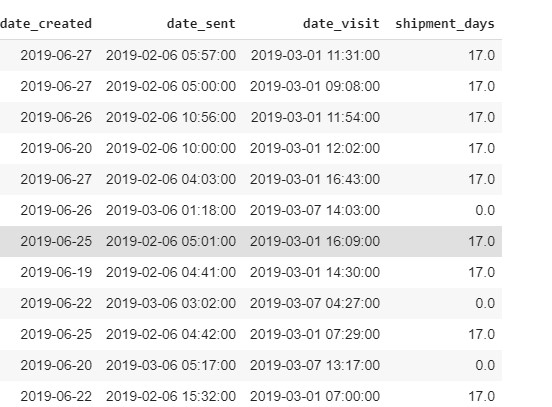
**- Date\_created, Date\_sent, Date\_visit:** son variables de tipo fecha.

En un análisis previo se estudió la variable shipment\_days y se pudo ver mediante un estudio de cajas, con sus valores outliers respectivos, que en el data set se encontraban valores negativos en dicha variable. Se tomó la decisión de eliminar estos valores, dado que no representan valores reales y que la proporción de estos datos dentro de todo el conjunto es muy chica.

Luego a partir de este data set se analizaron si existía información duplicada, y como se confirmó procedimos a eliminarla. Dado que todos los valores de las variables eran las mismas.

Durante la primera inspección de los datos, se observó que la variable date\_created no considera horas a diferencia de las otras variables de fechas. Creemos que para resolver nuestro problema, el día de creación del envío resulta mucho más importante que la hora en sí misma con lo cual lo observado anteriormente no sería un problema. Además, considerando que la respuesta del problema en un principio debe consistir en dar un intervalo de días, las franjas horarias podrían no ser tenidas en cuenta a priori.

También en dicha variable se encontraron datos fuera del rango esperado, en cuanto a que la fecha de creación debe ser anterior a la fecha de envío.



La inspección de los datos de arriba nos inclina a pensar que los datos con fecha de creación en junio de 2019 en realidad corresponden a datos del mes de enero del mismo año. Esta suposición se basa en que las fechas de las variables **date\_sent** y **date\_visit** muestran una cierta coherencia entre sí y con la variable **shipment\_days**.

Esta hipótesis plantea la disyuntiva de si esta información debe ser eliminada o si las fechas de creación de junio deben ser modificadas poniendo en su lugar enero de 2019.

Si uno se guía por los datos la primera opción sería eliminar estas filas (ya que la cantidad de las mismas no afectaría a la cantidad total). Sin embargo podemos modificar nuestros datos subsanando de esta manera un posible error producido durante la carga de los datos. Creemos que para un análisis en un futuro cercano podemos imputar los datos de creación de junio poniéndoles fecha de enero.

En relación a la variable **date\_visit** no contamos con información en los meses del año 2018 (recordemos que el rango es Octubre 2018 hasta Abril 2019) lo cual nos llama poderosamente la atención.

### **¿Cuándo llega?: Features más importantes**

Uno de los objetivos principales de nuestro análisis es predecir (dar un número aproximado de días hábiles) de cuanto va a tardar en llegar un envío desde que el cliente realiza la compra a través de la plataforma de Mercado Libre hasta que el paquete llega a su casa.

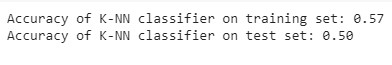
Esto nos lleva a pensar que dicha predicción se parecerá a los valores de la variable shipment\_days con, quizás, 1 o 2 días más de diferencia. Pues este último feature cuenta el intervalo de días entre date\_sent y date\_visit y nosotros necesitamos dar la aproximación a partir del momento de la compra.

Frente a este problema consideramos que el estado de Brasil que recibe el envío (receiver\_state) es un feature importante a tener en cuenta (mientras más lejos esté de San Pablo probablemente demore más). Por otro lado, el tipo de correo (service) también puede ser una variable importante a tener en cuenta porque la calidad de servicio impactará directamente en los días que tarda en llegar el pedido.

Para ello aplicamos un procedimiento de clustering knn, donde la variable shipment\_days previamente fue separada en clases.

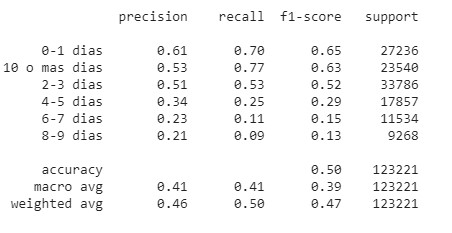
La primera clasificación que se realizó fue con los features de inputs: **service** y **receiver\_zipcode.**

Encontramos un accuracy de :



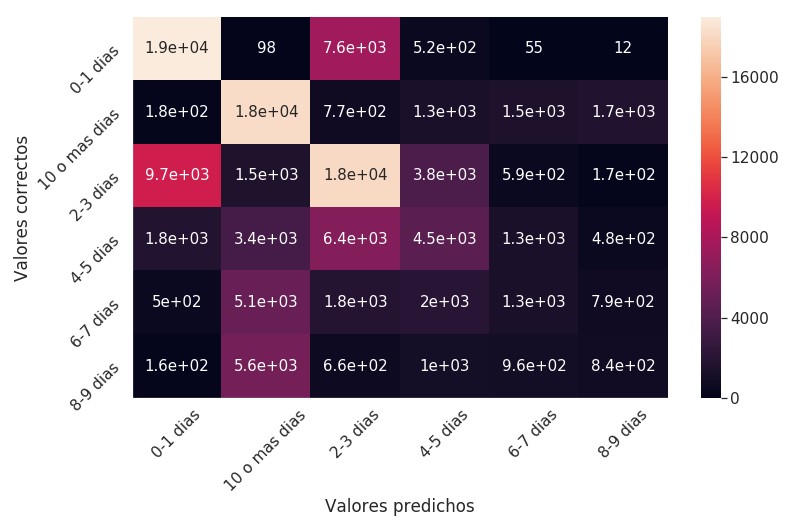
Recordemos que el accuracy indica la proporción de datos que fueron clasificados como se esperaba.

A continuación mostramos el reporte de clasificación:



En el mismo podemos ver, por cada clase la precisión obtenida por el clasificador. Y el accuracy promedio macro y weighted en 0.41 y 0.46 respectivamente.

A continuación veamos la matriz de confusión:



La matriz de confusión la podemos leer como por ejemplo:

En la clase 10 o más días se predijieron 98 valores que corresponden a la clase 0-1 dias.

En la clase 6-7 días se predijieron 55 valores que en realidad pertenecen a la clase 0-1 dias.

En la clase 8-9 días se predijieron 12 valores, que corresponden a la clase 0-1dias.

Los valores de la diagonal de la matriz de confusión muestran la cantidad de predicciones correctas de cada una de las clases.

En un problema de clasificación de clases es un buen signo tener valores altos en la diagonal de la matriz de confusión y bajos fuera de ella, pues esto significa que la predicción de clases es buena..

También se realizaron otros experimentos con otras selección de features, pero no se logró mejorar el accuracy.

### **Conclusión**

A pesar de utilizar distintos features en el clustering de knn obtuvimos un bajo accuracy, lo cual podemos concluir que si bien knn es un clasificador automático excelente para nuestro conjunto de datos no podemos obtener una clasificación esperada. En un futuro podemos utilizar otros clasificadores que mejor se adapten a nuestra problemática.