# *Proyecto Final*
# *Minería de datos*
# *Competencia Walmart*

**Equipo:**
Nancy Dira Martínez Guzmán
Marco Antonio Ramos
Santiago Battezzati

**Profesor**:
Juan Salvador Marmol Yahya

**Índice**



**Resumen**

El siguiente reporte resume lo presentado y realizado como parte del proyecto final de Minería de datos y la competencia de Walmart. Se presenta aquí una guía para comprender los documentos presentados así como el proceso que siguió el equipo en su realización. Cada una de las partes de este reporte remite a un documento presentado. Aquí se presenta alguna información adicional sobre cómo se llevó a cabo el proceso.

1. **Reproducibilidad del proceso**

Para hacer reproducible el proceso se debe proceder a:

1. Correr el archivo 0_reporte_eda_y_limpieza.Rmd
   Este archivo llama a su vez a los que se encuentra dentro de aux-R y genera los feathers de las bases de datos limpias y transformadas para los procesos subsiguientes.
2. Correr el archivo 1_modelado_walmart.ipynb
   Este archivo carga datos en Python, realiza selección de variables y corre magic loop para determinar el mejor modelo y sus mejores hiperparámetros.
3. Correr el archivo 2_modelo_final.ipynb
   Entrena los dos mejores modelos obtenidos por el magic loop, SVM y Random Forest y prepara los datos para subirse al concurso de Kaggle.

## 2. Comprensión del negocio

Se adjunta el documento 00_comprensión_negocio.Rmd

### 1.1. Antecedentes

Cada vez más empresas recaban información de sus procesos con el propósito de mejorar sus estrategias comerciales. En el caso de los supermercados, es muy importante conocer el tipo de clientes y el tipo de compras que realizan para poder mejorar el servicio al cliente, el atractivo de la marca y la competitividad de la empresa.

En este sentido, Walmart ha concentrado la información sobre compras realizadas por sus clientes en distintos viajes y con base en ello los ha categorizado de acuerdo a características en común. Por ejemplo, los clientes pueden hacer viajes de compra de la despensa semanal, viajes de compras muy especializadas (como el pago de servicios o la compra de medicamentos), algún viaje de temporada (como comprar de disfraces y dulces para Halloween), entre otros.

### 1.2. Determinación del objetivo

El reto del proyecto consta en poder recrear esta categorización con un acceso más limitado a los datos. Esto tiene el potencial de proponer nuevas y más robustas maneras de categorizar los viajes lo que puede traer muchas ventajas a la empresa, por ejemplo, en los modos de presentar sus productos a sus clientes en locales.

En este sentido, se proveen la base de datos train.csv que contiene la categorización. Con base en ella el proposito es predecir la categorización en la base de datos test.csv.

### 1.3. Determinación de criterio de éxito

El principal reto del proyecto es superar el accuracy que se podría haber logrado con la base de datos sin preprocesar. Ese será el baseline para los modelos que se buscarán desarrollar.

### 1.4. Plan del proyecto

Posterior a la comprensión de negocio y de acuerdo a la metodología CRISP-DM., para lograr el objetivo debemos en primer lugar hacer un análisis exploratorio de los datos para lograr en la mayor medida posible una comprensión integral de estos. En segundo y tercer lugar, con base en la comprensión del negocio y de los datos, proponer transformaciones y evaluar el desempeño de distintos modelos. Mencionamos segundo y tercer lugar porque estos procesos los debemos realizar en conjunto. Una vez elegido las transformaciones y el modelo final procedemos a la etapa de evaluación en la que reentrenanermos el modelo con datos de entrenamiento y prueba con hiperparámetros optimizados para posteriormente subir el modelo a kaggle. Finalmente, para la parte de despliege se desarrollara un web service en flask para predecir resultados a partir de nuevos datos y un reporte ejecutivo.

2. **Limpieza de datos, análisis exploratorio e ingeniería de características**

Se adjunta documento 0_reporte_eda_y_limpieza.Rmd

En esta instancia se buscó una primera comprensión de los datos.
Se comenzó cargando los datos para ver si había errores de carga. Se analizaron la descripciones de las columnas originales, presentadas en la documentación original.

Las columnas originales son :

**TripType:** a categorical id representing the type of shopping trip the customer made. This is the ground truth that you are predicting. TripType_999 is an "other" category.
**VisitNumber:** an id corresponding to a single trip by a single customer
**Weekday:** the weekday of the trip
**Upc:** the UPC number of the product purchased
**ScanCount:** the number of the given item that was purchased. A negative value indicates a product return.
**DepartmentDescription:** a high-level description of the item's department
**FinelineNumber:** a more refined category for each of the products, created by Walmart

Se analizan distintos aspectos como los valores faltantes, que en una primera instancia se decide no imputar. Se cambian nombres de las variables (para respetar convenciones) y se adapta la lectura de datos para que el tipo de variables sean categóricas.

**En el análisis univariado y bivariado** se detectan algunas características principales de los datos, presentadas en el documento. Algunas de ellas son

con los tipos de productos más comprados, el tipo de trip type más común, los días más comunes para hacer las compras (fines de semana), entre otros.
También descubrimos una relación en los datos faltantes: vemos que si la observación está imcompleta, la compra suele provenir de PHARMACY RX. Por lo tanto, se decide realizar una imputación.

No encontramos fuerte correlación entre variables numéricas.
(Se adjunta shinny app en la que se pueden observar las gráficas bivariadas en caso de que se busque mayor profundidad, aunque las principales están presentadas en el EDA).

Se escriben los datos en feather, para poder seguir el proceso en Python.

**Ingeniería de características**

Se adjunta el documento **04-to-wider** en la carpeta aux-R, donde se hace la ingeniería de características. Sin embargo ese archivo es llamado a su vez por el reporte general de esta sección, más arriba mencionado.

La decisión principal consistió en convertir a cada viaje en la unidad de observación. Para eso se tomaron las variables de días de la semana y department description y se las convirtió a dummies. Y se hizo una suma de cada columna, agrupada por viaje. De este modo, se construyó una tabla en wide en la que cada observación no es ya un objeto comprado (como en la original) si no un viaje entero. También se crearon las variables regreso y variedad, que contabilizan cantidad de ítems regresados y variedad de departamentes en cada viaje.

## 3. Modelado y Evaluación

**Se adjuntan los archivos** 1_modelado_walmart.ipynb y 2_modelo_final.ipynb donde se lleva a cabo modelado y evaluación.

Con los datos preparados, se procedió a analizar algunos modelos. Esta parte se realizó en Python, cargando los datos en feather. Se dividió en test y train, y se hizo un proceso de selección de características a partir de un Extra Trees Classifier. A continuación, se realizó un magic loop para elegir de entre una serie de modelos y encontrar sus mejores hiperparámetros.
Una vez obtenidos los dos mejores modelos, los cuales resultaron SVM y Random Forest se los volvió a entrenar y se procedió a su evaluación.

Se subieron los resultados a Kaggle para comprobar el score obtenido.
Dada la falta de tiempo, no pudimos ahondar en estrategias alternativas. Esto se debió también a que el proceso requiere muchas horas, sobre todo el magic loop.
Sin embargo, de haber tenido más tiempo, hubiera correspondido regresar a la etapa de transformación de los datos y probar el magic loop nuevamente, con los datos dispuestos de otra manera ya que CRISP es un método iterativo.

Una de las alternativas que intentamos probar fue la de disponer los datos en una matriz rala en wide, que contuviera one hot encoding de los días de la semana, el description department y el fineline number, para luego hacer una selección de las características (en total, se convierten en más de 5000 de este modo). La búsqueda de esta prueba puede verse aquí: modelos_ok_prueba_sparse.ipynb
Aunque no se consiguió terminar, por falta de tiempo, se obtuvo la matriz rala y se preparó todo para realizar una selección de características, que luego permitiera insertar en el magic loop y seleccionar modelos nuevamente.