Proyecto modelos de Machine Learning aplicado al cálculo de probabilidad de refacturas de clientes

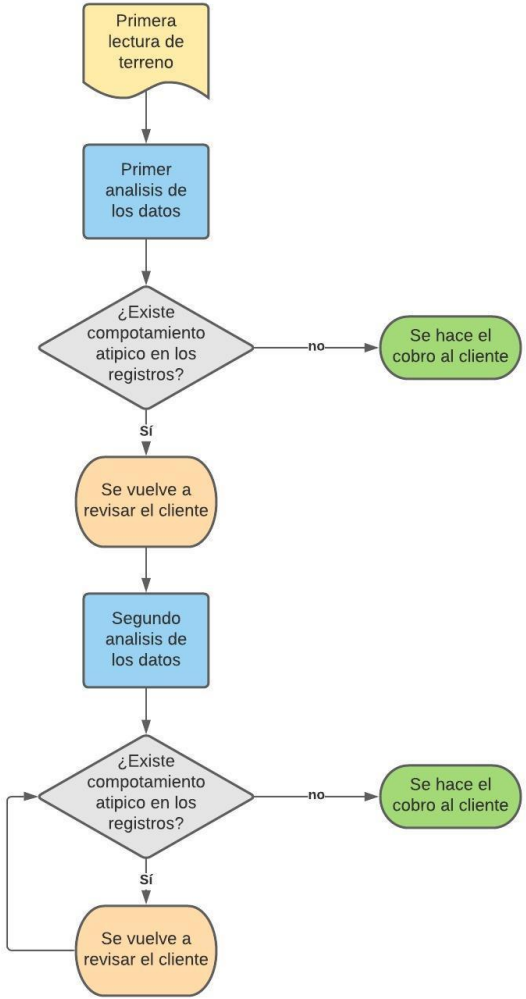
Joaquín Farias Muñoz
Practicante Esval S.A.

Índice

Proceso de análisis de cobros y refacturas

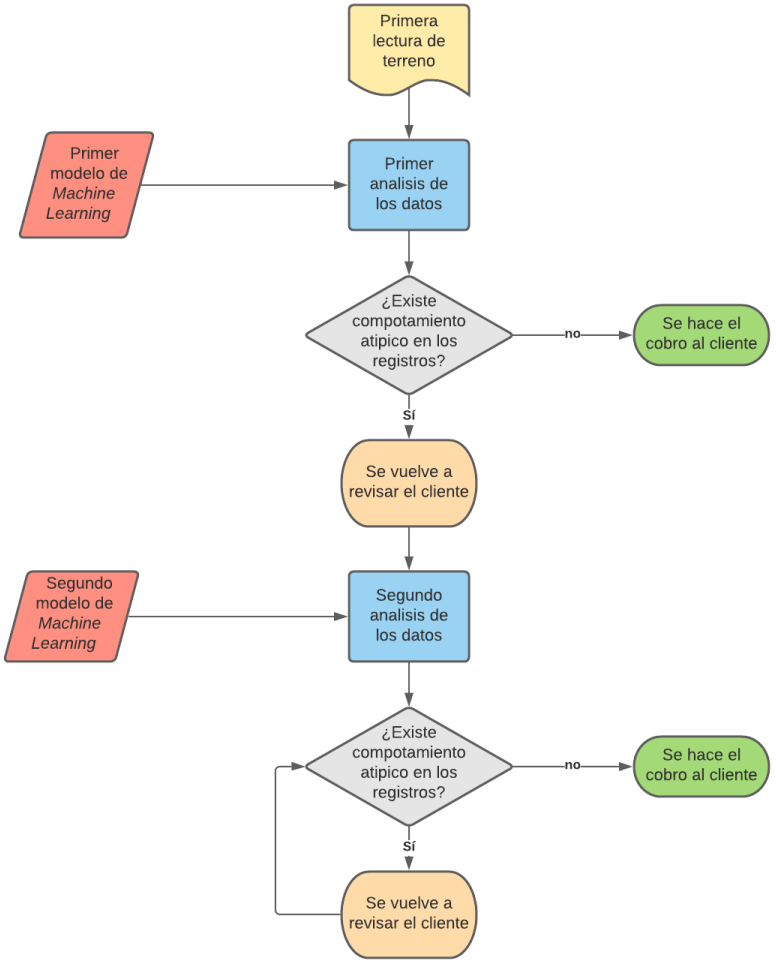
- Existen varios pasos de extracción de los datos y validación dentro de los cobros de los clientes.

- A pesar de esto aun existen lecturas erróneas que terminan en refacturas, lo que conlleva un costo para la empresa.



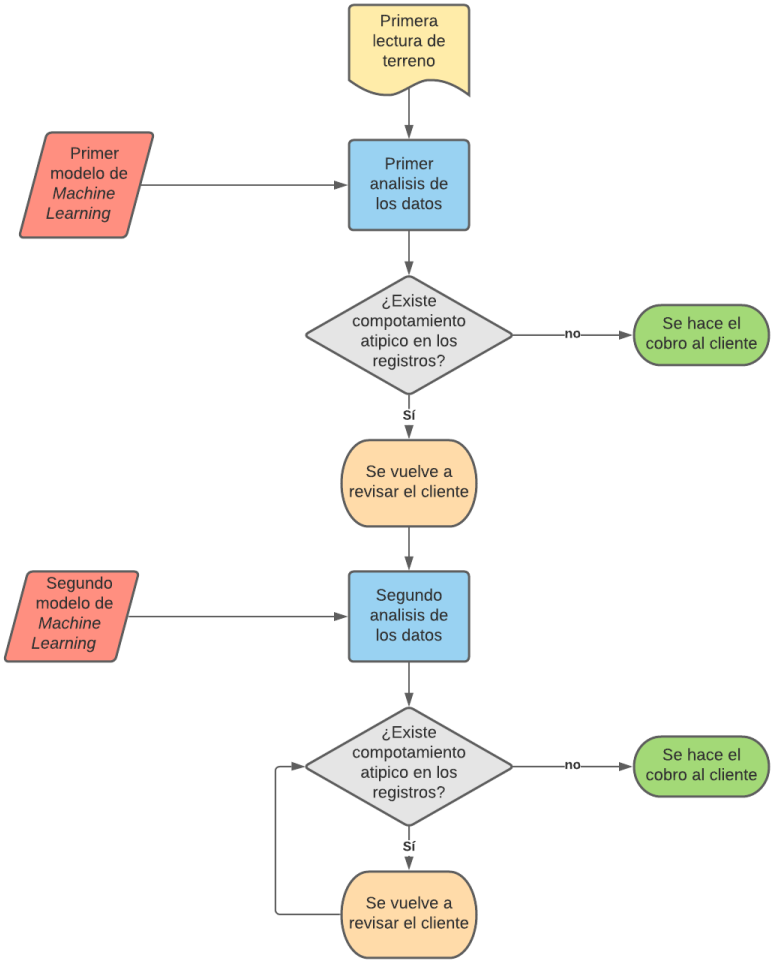
Proceso de análisis de cobros y refacturas

- Como solución al problema de refacturas se propone entrenar **modelos de machine learning**, para así disminuir lo mas posible las lecturas erróneas y las refacturas a los clientes.



Proceso de análisis de cobros y refacturas

- Se pretende ocupar modelos en dos instancias:
  - Después del primer análisis de datos solo con las lecturas de terreno.
  - Después del segundo análisis de datos con los datos entregados por lo analistas.



Índice

- Definición de problema
- **Exploración de datos**
- Preprocesamiento de datos, algoritmo y métricas
- Modelos finales, resultados y tiempos
- Discusiones y conclusiones

Descripción de datos y valores nulos

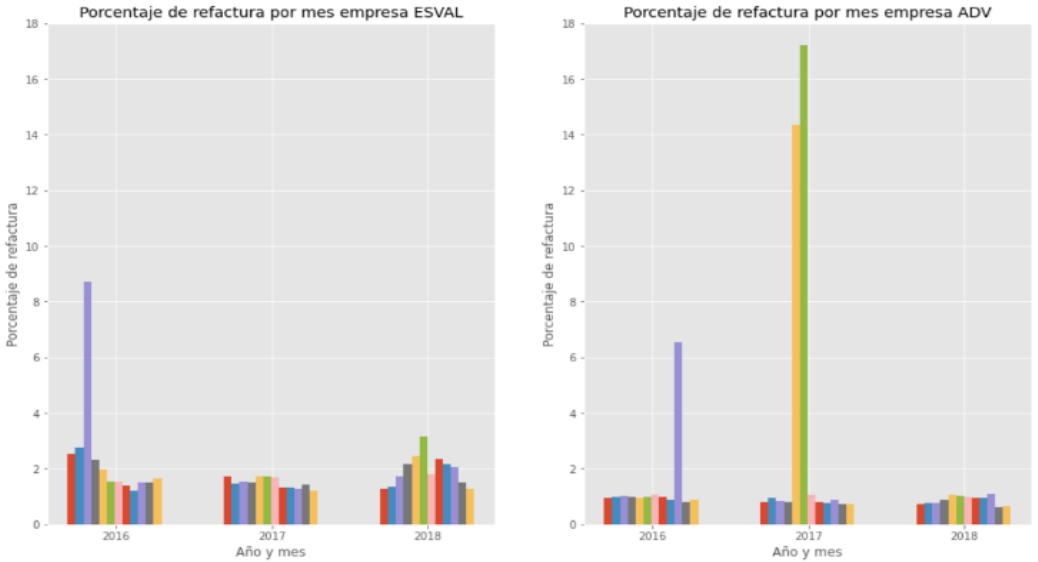
- Los datos se extraen del archivo “2016-2018v3.csv” y que contiene 28 columnas con datos entre los años 2016 y 2018.
- Los valores nulos categóricos se llevaran a la mayoría de cada variable y los numéricos se eliminaran.
- Nuestra etiqueta es “TIENE_REFA” que contiene la información de si el cliente tuvo refactura o no en el mes actual.

NRO_SUMINISTRO	0
COD_DIAMETRO	1
LECTURA	0
LECTURA_TERRENO	0
CLAVE_TERRENO	0
LECTURA_ANT	0
CONSUMO_BASE	1
CONSUMO_PROM	0
CATEGORIA	2172
TIP_DOCUMENTO	0
COD_LECTOR	935
COD_OBS	936
COD_LOCALIDAD	0
RECORR1	0
RECORR2	0
TIE_REMAR	5197537
ID_RELACION	0
COD_EMPRESA	0
CLAVE_LECTURA	0
CLAVE_TERRENO_MES_ANT	756369
CONS_BASE_MES_ANT	1
COD_OBS_MES_ANT	756854
CLAVE_TERRENO_MISMO_MES_ANNO_ANT	756369
CONS_BASE_MISMO_MES_ANNO_ANT	0
COD_OBS_MISMO_MES_ANNO_ANT	757302
ANNO	0
MES	0
TIENE_REFA	0
dtype: int64	

Columnas y valores nulos.

Datos atípicos

- existen 5 meses de distintos años que presentan un comportamiento atípico respecto a los demás.
- En principio se intento filtrar estos meses cc datos atípicos, pero finalmente se decidió por **eliminarlos**.
- Lo anterior porque los modelos tienen mejor desempeño sin los meses atípicos que con ellos filtrados.



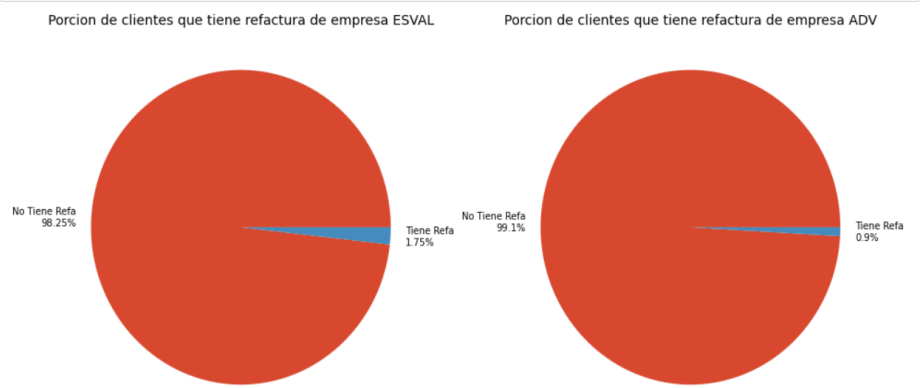
Refacturas por mes y año de empresa ESVAL y Aguas del valle.

Correlaciones

- existe poca correlación entre las variables a excepción de algunas puntuales.
- Estas excepciones son " LECTURA_TERRENO ", que corresponde a la lectura del mes actual con "LECTURA_ANT" y "LECTURA". Y, “CONSUMO_BASE” con “CONSUMO_BASE_MES_ANT” y “CONS_BASE_MISMO_MES_ANNO_ANT”.
- Las correlaciones son esperables por la forma de los datos.

Matriz de correlación.

Graficas de proporción de refactura

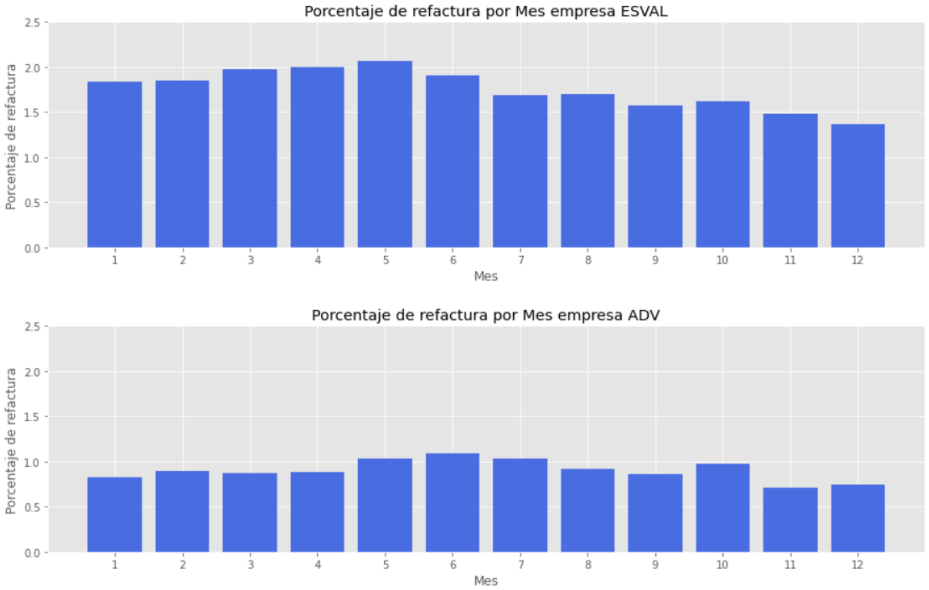


Proporción de refacturas por empresa.

- El porcentaje es bastante bajo comparado con el total de registros de los datos, esto nos asegura que trabajamos con una base de datos desbalanceada.
- Lo anterior debe tomarse muy en cuenta, ya que deben tomarse acciones especiales para que los algoritmos tengan un buen funcionamiento y se llegue a resultados óptimos.

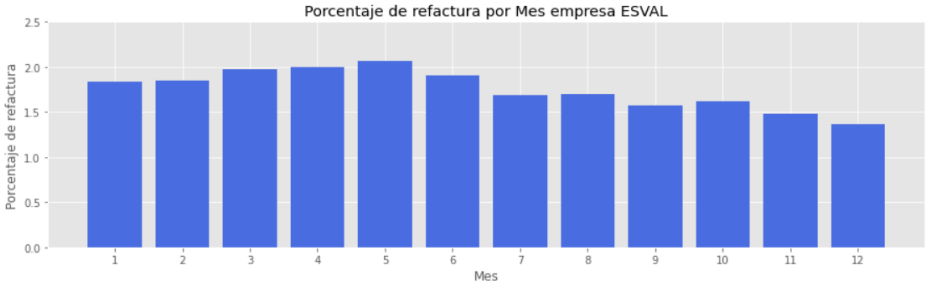
Graficas de proporción de refactura

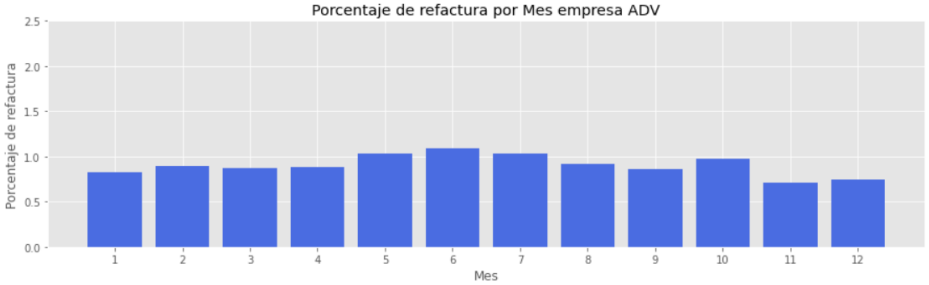
- Existe un cierto aumento en los primeros meses del año que va disminuyendo a medida que este avanza.
- Esto ya se ha notado antes y se atribuye a las segundas viviendas o residencias de vacaciones que solo son ocupadas durante la estación de verano. .
- Existe algunos clientes con estacionalidad contraria, es decir, se ocupan los inmuebles solo en durante el invierno y muy poco en verano, ejemplos de estos clientes podrían ser los colegios, las universidades, etc.



Proporción de refacturas por mes de cada empresa.

Graficas de proporción de refactura

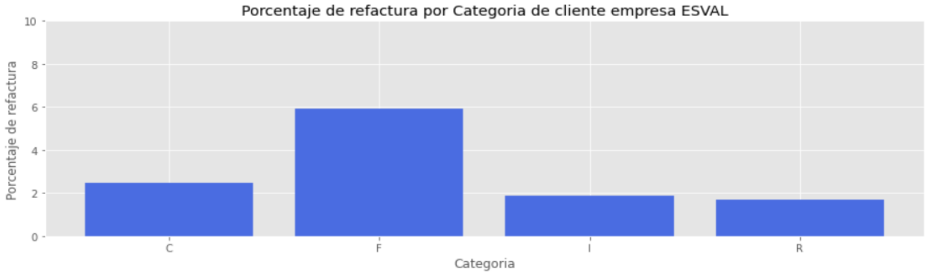




Proporción de refacturas por mes de cada empresa.

→ Los clientes que ocupan estas viviendas estacionales muy **a menudo se encuentran con facturas que sospechan tienen errores**, con lo que hacen un reclamo que termina en una refactura.

Graficas de proporción de refactura



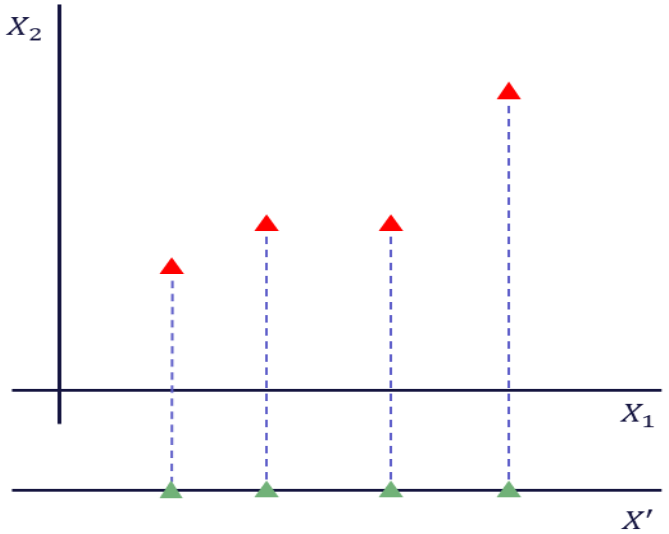
Porcentaje de refacturas por categoría de cliente y empresa.

- En las dos empresas el tipo de cliente que más tiene refacturas son los fiscales que corresponden al índice “F”.
- Generalmente se le hace consideraciones especiales a la hora de los cobros, y por eso tienen un mayor porcentaje de refacturas.
- Se opto por **eliminar los clientes de este tipo**, ya que los datos estarían sesgados gracias a estas consideraciones especiales y el porcentaje de estos clientes respecto al total es ínfimo.

Análisis APC

- El análisis de componentes principales APC, es una proyección de las distintas variables, que conllevan distintas dimensiones, disminuyendo así las dimensiones hasta un numero donde puedan ser graficadas.

- En la figura se puede apreciar un ejemplo del el análisis de componentes principales de los datos desde 2 dimensiones (X_1 y X_2) reduciéndolos a 1 dimensión (X').

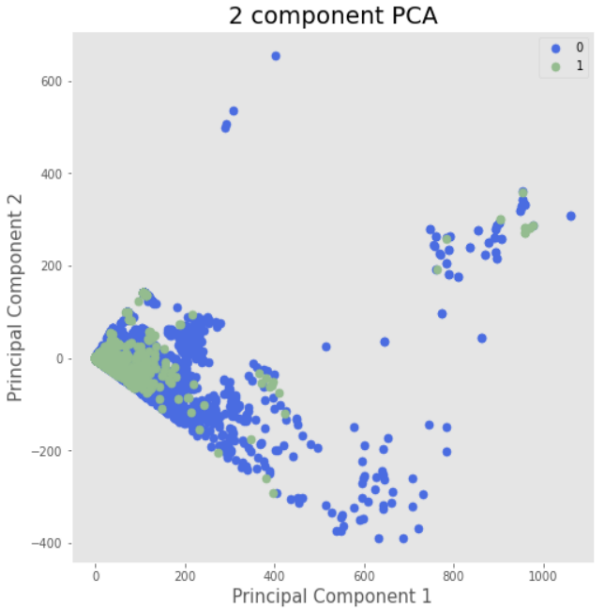


Ejemplo APC.

Análisis APC

- En la figura se puede apreciar el análisis de componentes principales de los datos reduciéndolos a 2 dimensiones.

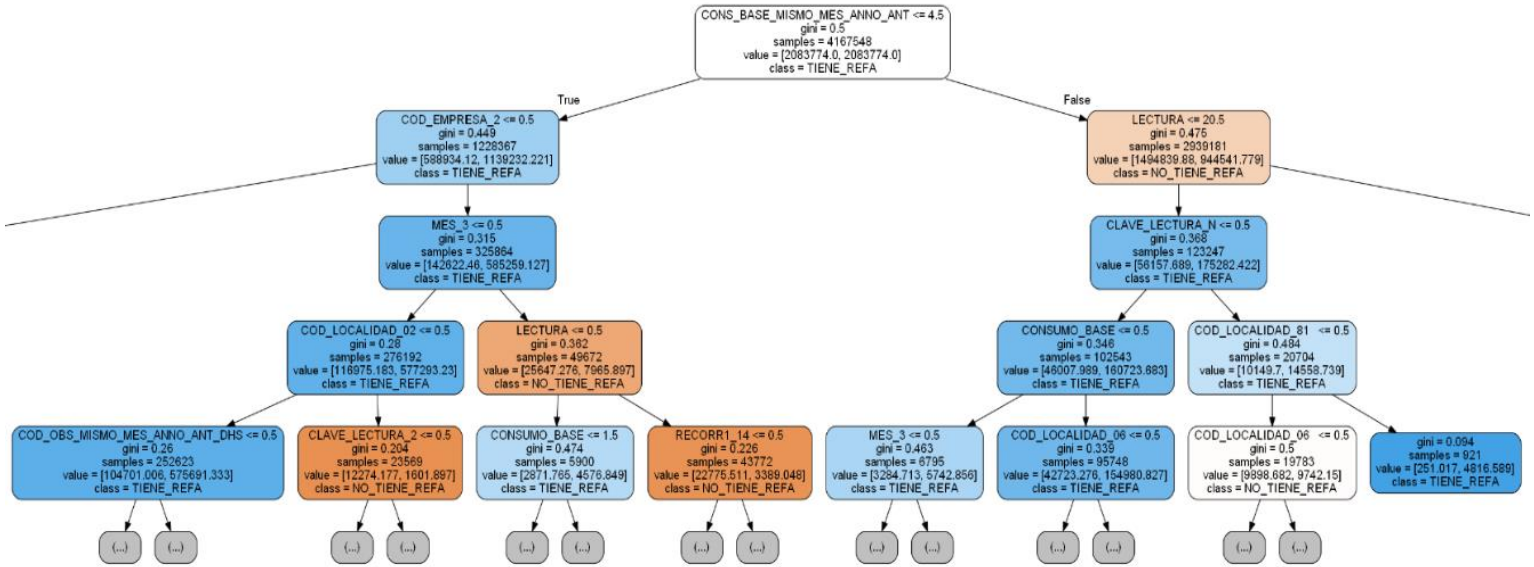
- Se puede notar que los datos no tienen una separación líneal clara, esto hace prever que cuando se clasifiquen los clientes con refactura también se incurrirá en errores clasificando clientes sin refactura como si la tuvieran porque están dentro o muy cerca de los límites de los datos de clientes con refacturas



APC para datos. En verde los clientes con refacturas, en azul los sin refacturas

Árbol de decisión explicativo

El árbol de decisión toma un registro y según si es mayor o menor a un umbral de ciertas variables decide a que nodo de decisión pasa a continuación, lo que finalmente lleva a las hojas finales donde se decide si el registro tiene refactura o no.

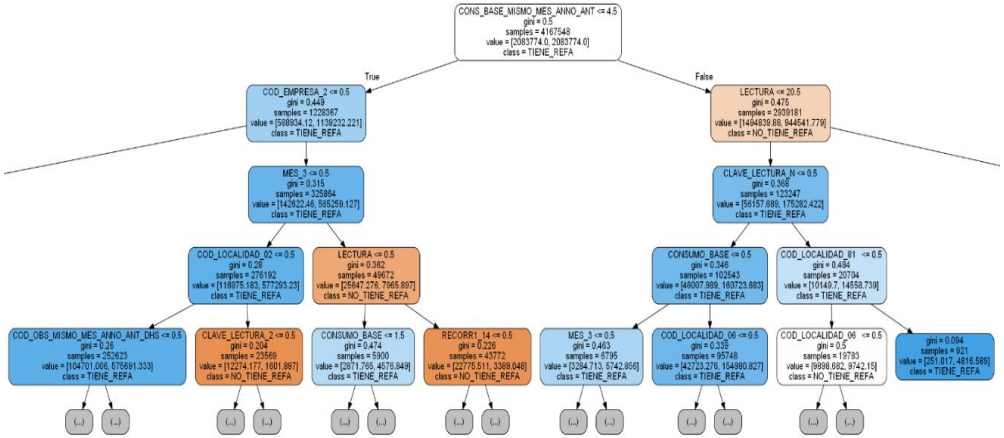


Árbol de decisión explicativo con todos los datos.

Árbol de decisión explicativo

- se puede notar que muestra una hoja o nodo final después de cuatro niveles.

- En lenguaje común esto quiere decir que si el consumo base del mismo mes del año anterior es mayor a 4.5, la lectura es menor que 2.5, la clave de lectura es normal (“N”) y la localidad es 81 tiene una **probabilidad del 0.29* de que el cliente tendrá una refactura**, o mejor dicho el modelo clasifica esos datos como clientes con refacturas

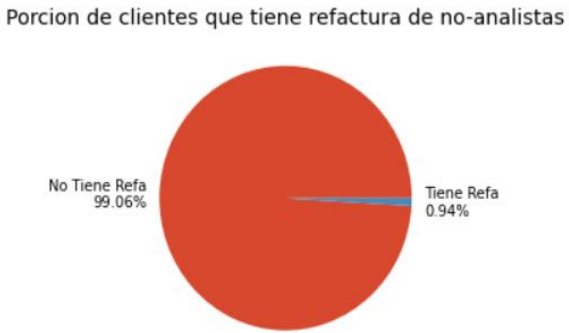


Árbol de decisión explicativo con todos los datos.

* Esto significa que dentro de todos los datos con esas condiciones el 29% tiene refactura, esto es alto porque en general el porcentaje de refacturas de los datos es de menos del 2%.

Datos con y sin responsabilidad del analista

***Los datos analista tienen un 6,1% de los registros de los dato no-analista.**



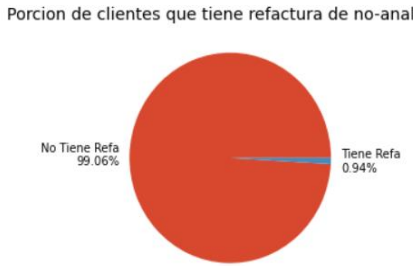
Proporción de refacturas de datos analista y no-analista*.

→ Se acordó que el modelo será usado en dos instancias.

→ Estas instancias tienen variables que están o no disponibles.

Datos con y sin responsabilidad del analista

- La primera instancia es en la que los datos llegan después de la primera lectura de terreno, dentro de esta no están aún las variables “LECTURA” y “CLAVE_LECTURA”. Los datos de esta instancia son llamados **datos no-analista**.



Proporción de refacturas de datos analista y no-analista.

- La segunda instancia es en la que los analistas ya han modificado o no consumos y las variables incluyen todas las columnas de los datos, pero solo los registros que tengan las categorías: "P", "A", "B", "C", "G", "M", "O", "W", y “Z” dentro de la variable “CLAVE_LECTURA”. Lo anterior, porque según los conversado con los expertos de la empresa estas claves son las que reflejan un cambio en el consumo de parte de los analistas. Estos datos son llamados **datos analista**.

Índice

- Definición de problema
- Exploración de datos
- **Preprocesamiento de datos, algoritmo y métricas**
- Modelos finales, resultados y tiempos
- Discusiones y conclusiones

Codificación de datos

→ Para los datos numéricos se ocupo la codificación minimax. Esta se define como:

$$minmax(x) = \frac{x - min}{max - min}$$

→ Para los datos categóricos se ocupa la codifica en caliente (One hot encoder). Que se explica en la figura.

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

Ejemplo de codificación One hot encoder para datos categóricos.

Eliminación de columnas

- Se eliminaron algunas columnas que no tiene sentido incluir en los algoritmos porque no aportan información sobre la probabilidad que un cliente tenga una refactura.
- También se eliminarán las columnas "**CLAVE_LECTURA**" y "**LECTURA**" en los datos de no-analista.

Columnas eliminadas:

- "**NRO_SUMINISTRO**": se eliminó porque es solo un identificador de cliente, pero no aporta información para la clasificación.
- "**COD_LECTOR**": se eliminó porque es solo un identificador de lector, pero no aporta mucha información solo la clasificación.
- "**ANNO**": se eliminó porque solo aporta información respecto al año de la lectura, pero en el futuro el modelo se ocupará en años que no están dentro de los datos de entrenamiento de los modelos.

División de datos

Pero, las únicas divisiones que tuvieras bueno resultados fue por **código de diámetro** y por **temporada de consumo**.

→ Se probo el desempeño de los algoritmo propuesto con varias divisiones de datos como:

- División por categoría de cliente.
- División por empresa.
- División por mes.
- División por tipo de remarcador.
- División por tarifa.
- División por localidad

División de datos

Pero, las únicas divisiones que tuvieras bueno resultados fue por **código de diámetro** y por **temporada de consumo**.

→ Para hacer la división por temporada de consumo se debió crear una nueva variable, para esto se realizó con la siguiente formula para todos los clientes:

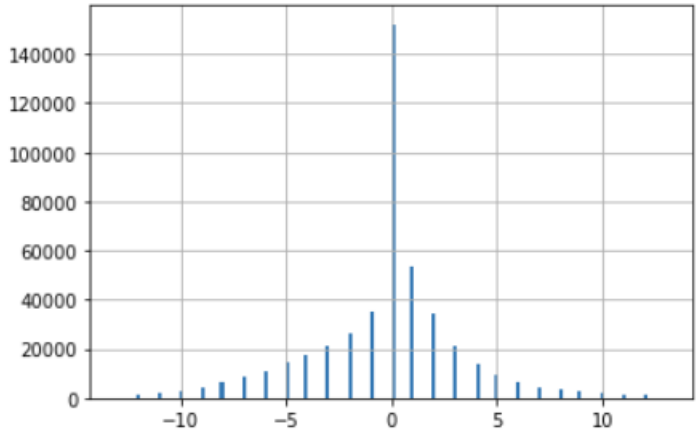
$$diff_{consumo} = \left(\frac{Dic + Ene + Feb + Mar}{4}\right) - \left(\frac{Abr + May + Jun + Jul + Ago + Sep + Oct + Nov}{8}\right)$$

División de datos

Pero, las únicas divisiones que tuvieras bueno resultados fue por **código de diámetro** y por **temporada de consumo**.

→ Con esto se hizo un histograma y se dividió los clientes en tres grupos:

- Los clientes que tienen más consumos en la temporada de verano.
- Los clientes que tienen más consumos en la temporada del resto de años.
- Los clientes que tienen el mismo consumo en ambas temporadas



Histograma con nueva columna “DIFF_CONSUMOS”.

Algoritmos de Machine Learning

- Se emplearon cuatro algoritmos de machine learning para intenta solucionar el problema.
- De estos se obtuvieron los que tienen un mejor rendimiento en distintos escenarios.

Regresión logística.

Árbol de decisión.

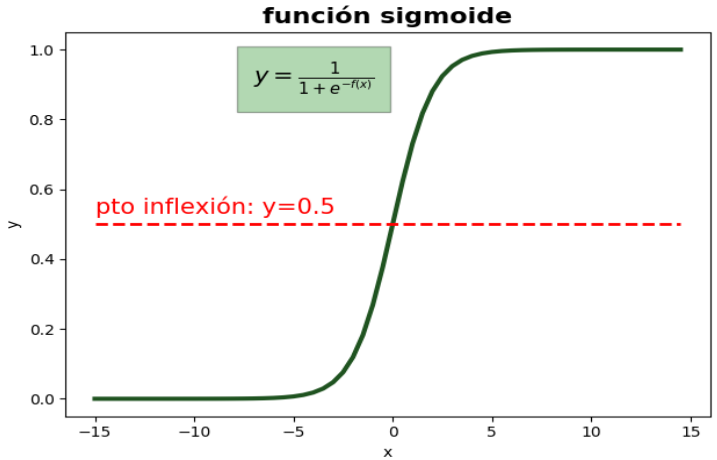
Redes neuronales artificiales.

XGBoost.

Algoritmos de Machine Learning

Regresión logística.

- Este algoritmo funciona de forma iterativa: primero calculando el error de clasificación de los datos dependiendo de una función de costo, luego de pasar por una función de decisión que en general es la función logística (o sigmoide).
- Con el error calculado intenta ajustar los parámetros con el gradiente descendiente para así mejorar la eficacia de la clasificación.
- En general, se le da un número máximo de iteraciones para no caer en sobreajuste de los modelos.



Algoritmos de Machine Learning

Árbol de decisión.

- Es un algoritmo que dependiendo de una condición en un nodo va decidiendo como se clasificando los datos o como se van repartiendo a lo largo del árbol hasta llegar a los nodos o hojas finales.
- Cabe destacar que dependiendo de cómo se ordenen las condiciones y las variables en un árbol puede tener resultados distintos

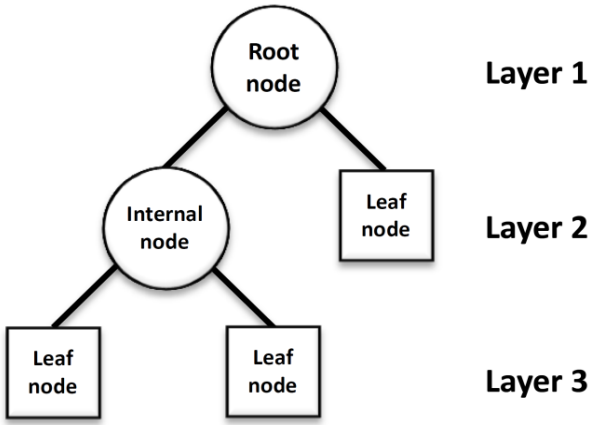


Diagrama simple de árbol de decisión.

Algoritmos de Machine Learning

XGBoost.

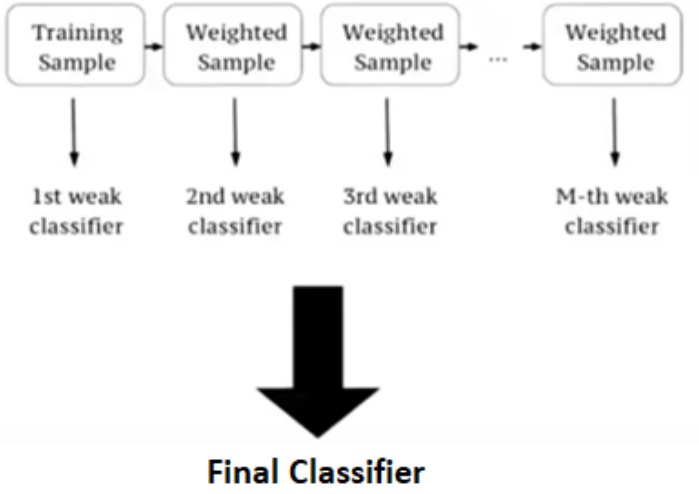


Diagrama simple de XGBoost.

→ Es un algoritmo de booster, esto quiere decir que es un algoritmo que tomar varios algoritmos simples y toma una decisión dependiendo lo que diga la mayoría de estos algoritmos por separado. En este caso, XGBoost crea varios árboles de decisión débiles para clasificar los datos.

Algoritmos de Machine Learning

Redes neuronales artificiales.

- Es un algoritmo que intenta imitar el comportamiento de las neuronas de nuestro cerebro. Este algoritmo es iterativo: donde luego de definirse cuantas capas y neuronas tendrá se hace pasar todos los datos por el modelo para luego propagar el error hacia atrás e ir cambiando los parámetros de cada neurona (que serían los pesos de cada variable).

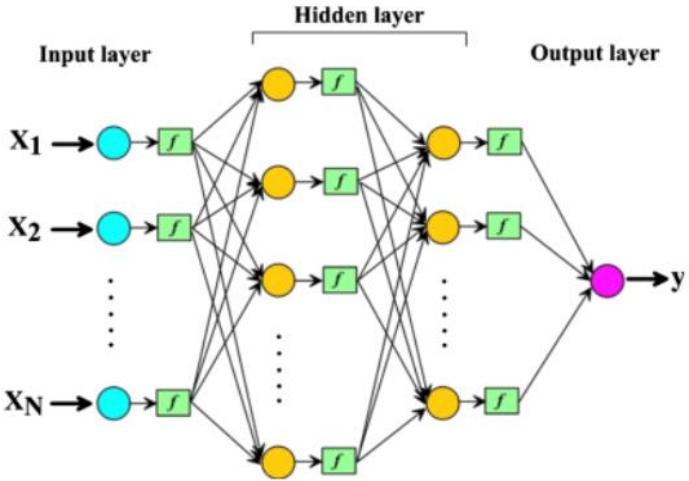


Diagrama simple redes neuronales artificiales.

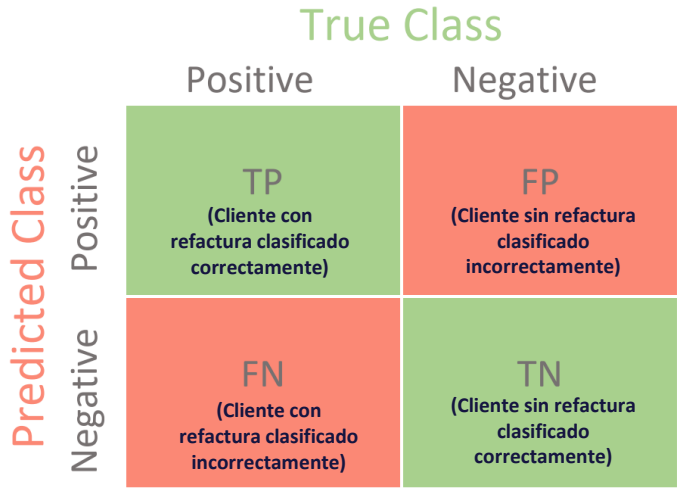
Métricas de evaluación

La matriz de confusión para un caso binario es una matriz de dimensiones 2x2, donde se la clasificación de Verdaderos positivos, Falsos positivos, Verdaderos negativos y Falsos negativos.

En nuestro caso la clase positiva seria “no tiene refactura” y la negativa “tiene refactura”, solo por un tema de orden de variables se debe tomar en cuenta a la hora de los cálculos.

Verdaderos positivos (VP o TP): son la cantidad de datos de la clase positiva (tiene refactura) que se clasificación bien.

- **Falsos positivos (FP):** son la cantidad de datos de la clase positiva (tiene refactura) que se clasificación mal.
- **Verdaderos negativos (VN o TN**): son la cantidad de datos de la clase negativa (no tiene refactura) que se clasificación bien.
- **Falsos negativos (FN):** son la cantidad de datos de la clase negativa (no tiene refactura) que se clasificación mal.



Métricas de evaluación

El *Recall* mide el acierto de la clase positiva comparado con todos los clasificados positivos.

$$\text{Recall} = \frac{VP}{VP+FN}$$

Métricas de evaluación

La *precision* mide el acierto de la clase positiva comparado con todas las predicciones de la clase positiva.

$$\text{Precision} = \frac{VP}{VP+FP}$$

Métricas de evaluación

Se nota que no se puede diferenciar que modelos es mejor utilizando ninguna de estas dos métricas por separado.

El *Recall* es alto cuando se detecta bien la clase, es decir cuando los FN no son muchos. Pero, no toma en cuenta los FP.

¿Cómo solucionamos esto?

Se proponen dos métricas: F1-score y G-mean

La *precision* es alta cuando se clasifica la clase positiva con mas precisión, es decir cuando los FP no son mucho. Pero, no toma en cuenta los FN.

Métricas de evaluación

El *F1-score da un promedio entre los valores de* precisión *y* recall.

$$\text{F1} - score = \frac{2 * (precision * recall)}{precision + recall}$$

Pero, en el caso de *F1-score* ¿realmente importa la precisión y el recall de la misma manera*?*

Y, en el caso de *G-mean* ¿realmente importan las dos clases de la misma manera?

El *G-mean* da un promedio de la buena clasificación tomando en cuenta cada clase.

$$G - mean = \sqrt{\frac{VP}{VP + FN} x \frac{VN}{VN + FP}}$$

Métricas de evaluación

El *F-measura* (F_β) da una medida de la importancia entre el *recall* y la *precisión* controlado por la constante β.

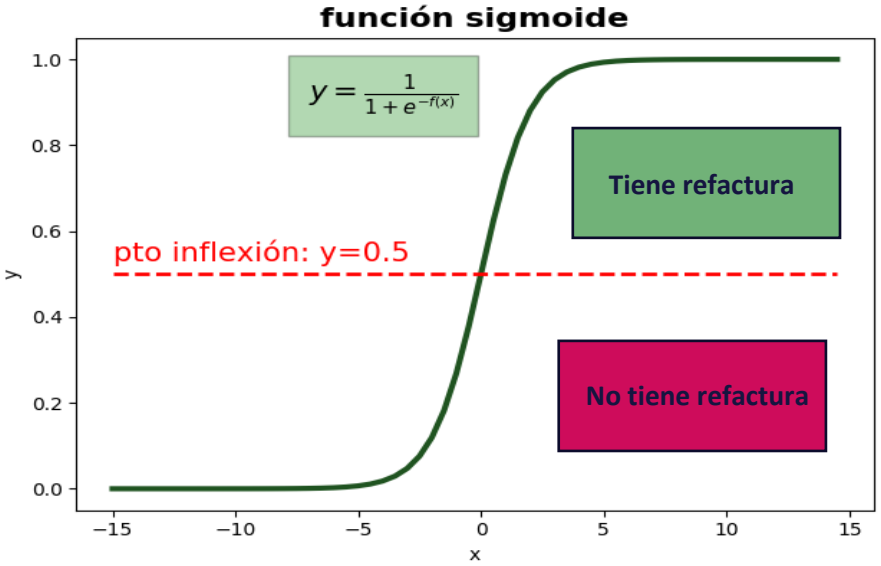
$$F_\beta = \frac{(1+\beta)^2 * (recall * precision)}{\beta^2 * recall + precision}$$

Si el valor de β es mayor que 1 se le da mas importancia al *recall* (es decir, no importa que hayan mucho positivos mal clasificados mientras no haya muchos negativos mal clasificados), mientras que si β es menor que 1 se da mas importancia a la *precision* (es decir, no importa que haya varios negativos mal clasificados, mientras que los positivos estén bien clasificados) .

Métricas de evaluación

¿Cómo decide la regresión logística y las redes neuronales?

La función sigmoide



Si $y > 0.5$ la predicción de ese registro es positivo, es decir tiene refactura (y=1)

Si $y < 0.5$ la predicción de ese registro es negativa, es decir no tiene refactura (y=0)

Se podría subir el umbral de decisión para evitar que se clasificaran tantos valores negativos como positivos.

Métricas de evaluación

La curva ROC es un grafico que muestra el rendimiento de un modelo de clasificación en todos los umbrales de clasificación.

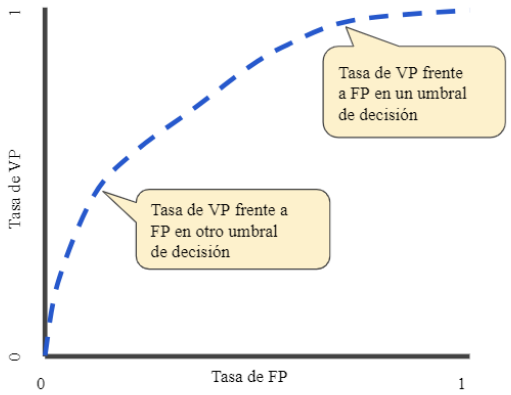
La métrica AUC es el área bajo la curva ROC, donde un área de 1 es un clasificador perfecto y 0 uno que clasifica todo mal.

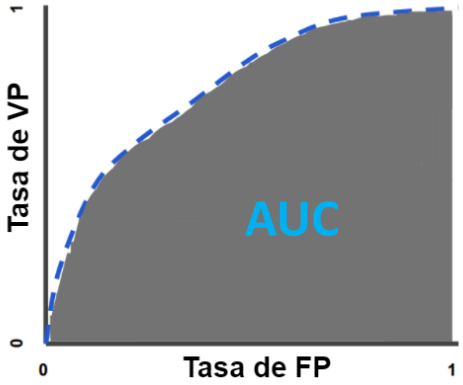
→ **Tasa de verdaderos positivos (TPR):**

$$TPR = \frac{VP}{VP + FP}$$

→ **Tasa de falsos positivos (FPR):**

$$FPR = \frac{FP}{FP + VN}$$

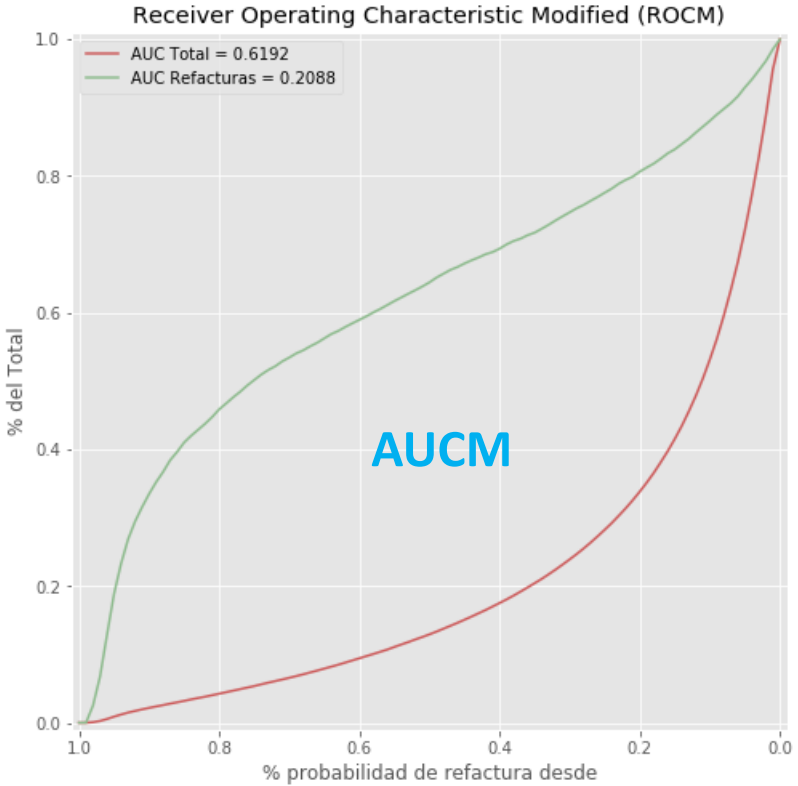




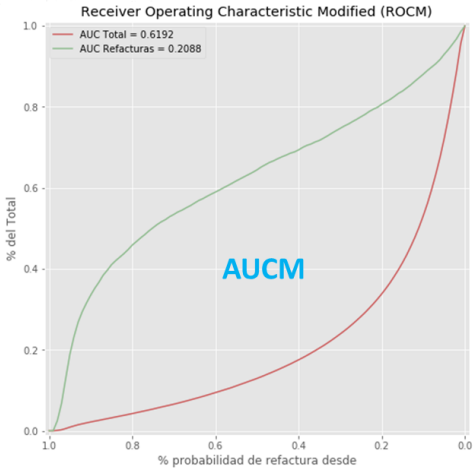
Métricas de evaluación

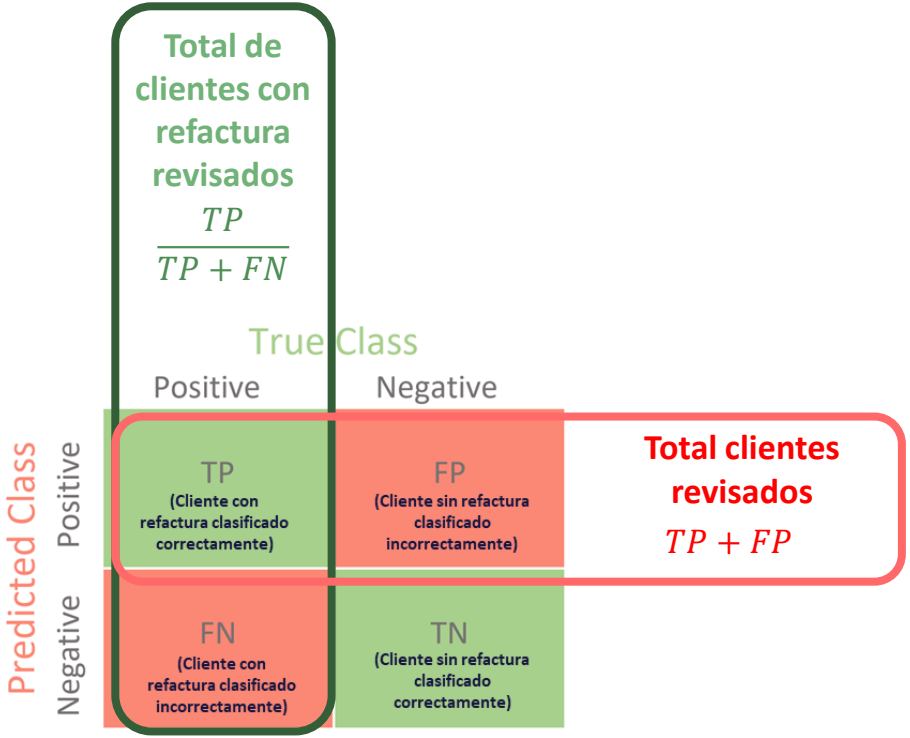
La línea verde representa clientes que tienen refactura que son revisados en cada umbral de decisión.

La línea roja representa el total de clientes que son revisados con cada umbral de decisión.



Métricas de evaluación





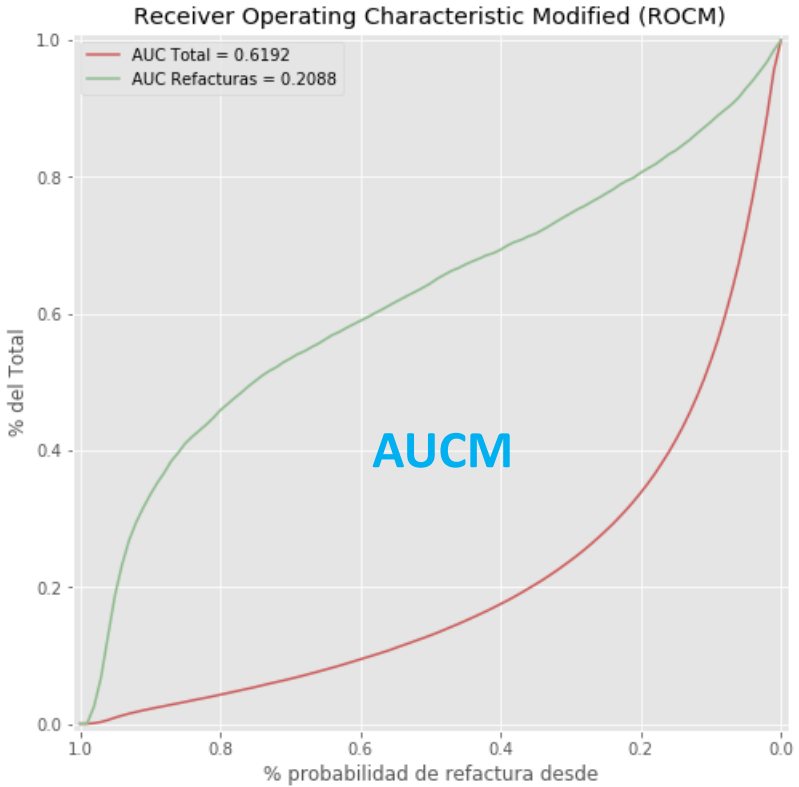
Métricas de evaluación

→ **Area Under Curve Modified (AUCM):**

$$\text{AUCM} = AUC_{refacturas} - AUC_{total}$$

→ **En el caso del ejemplo**

$$AUCM = 0{,}4104$$

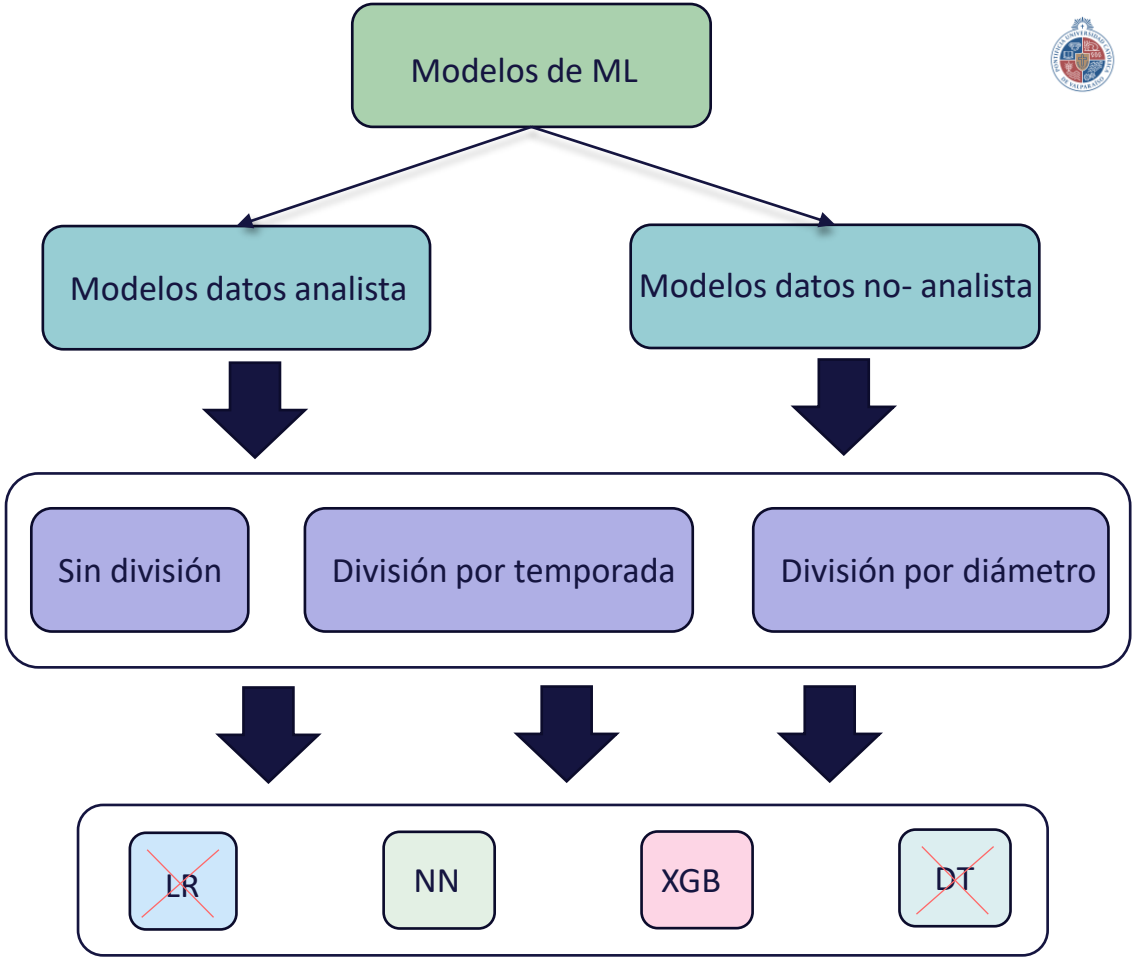


Índice

- Definición de problema
- Exploración de datos
- Preprocesamiento de datos, algoritmo y métricas
- **Modelos finales, resultados y tiempos**
- Discusiones y conclusiones

Modelos

- Como se dijo anteriormente se harán modelos para las dos instancias en que se ocuparan los algoritmos.
- De estos modelos solo se enfoco en las divisiones que tuvieron mejores resultados con los datos de prueba.



Modelos

- Se hizo un estudio de los mejores algoritmos para el problema planteado comparándolos en un caso de aplicación con los datos analista.
- Los mejores resultados los obtuvieron los algoritmos de XGBoost y de redes neuronales.

Comparación de resultados de los algoritmos.

Modelos	F1-score	G-mean	F-measure	AUCM	Porcentaje de refacturas revisadas
LR	0,06813	0,73277	0,650685	0,31941	0,47849
DT	0,065785	0,745445	0,673345	**0,397995**	0,42707
NN	0,10587	**0,75179**	**0,68664**	0,388445	**0,533805**
XGB	**0,114915**	0,66629	0,548015	0,35829	0,485605

Resultados de modelos no-analista

- Se nota que solo se muestran los algoritmo XGBoost y redes neuronales, esto es porque son los que tuvieron mejores resultados.
- Los mejores resultados se obtiene con el algoritmo de redes neuronales, mas no en todos los sectores de facturación.

Sin división.

sector	XGBoost sin división		Neural Networks sin división	
	Umbral decisión	% refacturas encontradas	Umbral decisión	% refacturas encontradas
1	0,39	0,50	0,59	**0,76**
2	0,49	**0,57**	0,75	0,53
3	0,38	**0,49**	0,67	0,46
4	0,64	0,33	0,65	**0,44**
5	0,58	0,38	0,69	**0,41**
6	0,68	0,49	0,68	**0,62**
7	0,74	0,45	0,69	**0,59**
8	0,65	0,34	0,57	**0,41**
9	0,49	**0,60**	0,7	0,50
10	0,42	0,41	0,67	**0,41**
11	0,66	0,34	0,72	**0,49**
12	0,49	0,32	0,64	**0,38**
13	0,51	0,36	0,65	**0,36**
14	0,34	0,69	0,56	**0,70**
15	0,56	**0,56**	0,65	0,51
16	0,48	0,22	0,62	**0,37**
17	0,42	**0,47**	0,66	0,45
18	0,44	0,64	0,58	**0,64**
19	0,51	0,61	0,72	**0,68**
20	0,28	0,41	0,66	**0,48**
Promedio		0,46		**0,51**
Máximo		0,69		**0,76**
Mínimo		0,22		**0,36**

Resultados de modelos no-analista

- Con esta división se obtiene mejores resultados que con los datos sin dividir.
- Los mejores resultados se obtiene con el algoritmo de redes neuronales, mas no en todos los sectores de facturación.

División por temporada.

sector	XGBoost por temporada		Neural Networks por temporada	
	Umbral decisión	% refacturas encontradas	Umbral decisión	% refacturas encontradas
1	0,52	0,54	0,61	**0,73**
2	0,6	0,59	0,72	**0,60**
3	0,54	0,41	0,62	**0,53**
4	0,75	0,33	0,61	**0,45**
5	0,65	0,45	0,67	**0,48**
6	0,75	0,56	0,71	**0,61**
7	0,78	0,41	0,7	**0,45**
8	0,76	0,32	0,6	**0,43**
9	0,61	0,52	0,73	**0,70**
10	0,69	0,38	0,59	**0,46**
11	0,74	0,38	0,76	**0,43**
12	0,71	0,33	0,65	**0,51**
13	0,62	0,34	0,59	**0,38**
14	0,45	0,66	0,56	**0,75**
15	0,69	0,57	0,71	**0,57**
16	0,64	0,34	0,7	**0,48**
17	0,59	0,43	0,71	**0,48**
18	0,58	0,63	0,61	**0,65**
19	0,64	0,55	0,66	**0,74**
20	0,54	0,41	0,63	**0,49**
Promedio		0,46		**0,55**
Máximo		0,66		**0,75**
Mínimo		0,32		**0,38**

Resultados de modelos no-analista

- Con esta división se obtiene mejores resultados que con los datos sin dividir.
- Los mejores resultados se obtiene con el algoritmo de redes neuronales, mas no en todos los sectores de facturación.

División por diámetro.

sector	XGBoost por diámetro		Neural Networks por diámetro	
	Umbral decisión	% refacturas encontradas	Umbral decisión	% refacturas encontradas
1	0,49	0,52	0,6	**0,79**
2	0,59	**0,71**	0,72	0,61
3	0,44	**0,58**	0,6	0,51
4	0,75	0,34	0,6	**0,47**
5	0,53	**0,51**	0,7	0,50
6	0,72	0,59	0,72	**0,65**
7	0,81	0,39	0,68	**0,48**
8	0,76	0,39	0,6	**0,48**
9	0,7	**0,80**	0,66	0,56
10	0,55	0,45	0,65	**0,46**
11	0,77	0,31	0,72	**0,47**
12	0,39	0,40	0,65	**0,47**
13	0,59	0,34	0,62	**0,39**
14	0,35	0,66	0,54	**0,79**
15	0,82	0,42	0,74	**0,6**
16	0,78	0,27	0,71	**0,39**
17	0,77	0,38	0,71	**0,55**
18	0,7	0,63	0,64	**0,66**
19	0,44	0,63	0,59	**0,77**
20	0,37	0,42	0,6	**0,49**
Promedio		0,49		**0,55**
Máximo		**0,80**		0,79
Mínimo		0,27		**0,39**

Resultados de modelos no-analista

Comparación de modelos data no-analista.

Modelos	F1-score	G-mean	F-measure	AUCM	Porcentaje de refacturas revisadas
NN sin división	0,10587	0,75179	0,68664	0,388445	0,533805
XGB sin división	**0,114915**	0,66629	0,548015	0,35829	0,485605
NN por temporada	0,100485	**0,760595**	**0,69987**	**0,40925**	0,55133
XGB por temporada	0,086355	0,69713	0,589485	0,37274	0,46323
NN por diámetro	0,100985	0,759235	0,697555	0,412245	**0,55905**
XGB por diámetro	0,101465	0,71887	0,632615	0,3934	0,490805

→ Se nota que el mejore modelo es con los datos divididos por diámetro usando el algoritmo de rede neuronales.

Resultados de modelos analista

- En esto modelos no tiene sentido la métrica del porcentaje de refacturas encontrada, así que se utilizan dos métricas que también son veraces.
- De estos modelos solo se enfoco en las divisiones que tuvieron mejores resultados con los datos de prueba.

Sin división.

sector	XGboost sin división		Neural Networks sin división	
	F-Measure	AUCM	F-Measure	AUCM
1	0,66	0,35	**0,77**	**0,45**
2	**0,68**	0,36	0,60	**0,38**
3	0,49	0,30	**0,50**	**0,32**
4	0,53	0,27	**0,77**	**0,35**
5	0,50	0,23	**0,97**	**0,47**
6	0,63	0,35	**0,71**	**0,38**
7	**0,67**	0,30	0,65	**0,36**
8	0,66	0,38	**0,74**	**0,43**
9	**0,99**	0,53	0,89	**0,52**
10	**0,60**	**0,39**	0,52	0,36
11	0,72	0,39	**1,01**	**0,52**
12	**0,79**	**0,45**	0,67	0,37
13	0,46	0,19	**0,58**	**0,32**
14	0,59	0,38	**0,65**	**0,40**
15	0,70	0,37	**0,73**	**0,38**
16	0,45	0,30	**0,70**	**0,29**
17	0,47	0,23	**0,51**	**0,30**
18	0,55	0,34	**0,67**	**0,42**
19	**0,69**	0,34	0,67	**0,36**
20	**0,56**	**0,30**	0,50	0,29
Promedio	0,62	0,34	**0,69**	**0,38**
Máximo	0,99	**0,53**	**1,01**	0,52
Mínimo	0,45	0,19	**0,50**	**0,29**

Resultados de modelos analista

- Con esta división se obtiene mejores resultados que con los datos sin dividir.
- Los mejores resultados se obtiene con el algoritmo de redes neuronales, mas no en todos los sectores de facturación.

División por temporada.

sector	XGBoost por temporada		Neural Networks por temporada	
	F-Measure	AUCM	F-Measure	AUCM
1	0,7	0,36	**0,75**	**0,48**
2	**0,63**	0,38	0,62	**0,40**
3	**0,51**	0,23	0,50	**0,33**
4	0,64	0,27	**0,74**	**0,35**
5	0,59	0,24	**0,78**	**0,39**
6	**0,69**	0,34	0,66	**0,37**
7	**0,70**	**0,32**	0,59	0,32
8	**0,74**	0,32	0,61	**0,38**
9	0,86	0,42	**0,97**	**0,59**
10	**0,60**	**0,38**	0,45	0,28
11	**0,88**	0,40	0,74	**0,44**
12	**0,74**	0,36	0,58	**0,39**
13	**0,62**	0,26	0,59	**0,37**
14	**0,78**	**0,46**	0,73	0,46
15	0,71	0,37	**0,71**	**0,43**
16	**0,65**	0,33	0,61	**0,34**
17	**0,55**	0,21	0,54	**0,26**
18	0,61	0,29	**0,69**	**0,44**
19	**0,80**	**0,34**	0,60	0,27
20	**0,66**	**0,36**	0,55	0,35
Promedio	**0,69**	0,33	0,65	**0,38**
Máximo	0,88	0,46	**0,97**	0,59
Mínimo	**0,51**	0,21	0,45	0,26

Resultados de modelos analista

- Con esta división se obtiene mejores resultados que con los datos sin dividir.
- Los mejores resultados se obtiene con el algoritmo de redes neuronales, mas no en todos los sectores de facturación.

División por diámetro.

sector	XGBoost con división		Neural Networks con división	
	F-Measure	AUCM	F-Measure	AUCM
1	0,80	**0,43**	**0,81**	0,42
2	0,65	0,42	**0,76**	**0,42**
3	0,62	0,38	**0,67**	**0,44**
4	0,57	0,27	**0,67**	**0,36**
5	0,65	0,28	**0,97**	**0,53**
6	0,76	0,38	**0,86**	**0,50**
7	0,61	0,34	**0,65**	**0,35**
8	0,77	0,45	**0,85**	**0,55**
9	0,93	0,46	**0,99**	**0,55**
10	**0,55**	**0,38**	0,52	0,35
11	0,77	0,35	**0,82**	**0,48**
12	**0,82**	**0,47**	0,69	0,41
13	0,55	0,31	**0,69**	**0,47**
14	0,62	0,36	**0,82**	**0,51**
15	0,70	0,41	**0,77**	**0,49**
16	0,64	0,37	**0,67**	**0,38**
17	0,59	0,27	**0,69**	**0,35**
18	0,61	0,38	**0,82**	**0,55**
19	**0,8**	**0,40**	0,76	0,44
20	0,67	0,36	**0,84**	**0,43**
Promedio	0,681	0,37	**0,77**	**0,45**
Máximo	0,93	0,47	**0,99**	**0,55**
Mínimo	**0,55**	0,27	0,52	**0,35**

Resultados de modelos analista

Comparación de modelos data analista.

Modelos	F1-score	G-mean	F-measure	AUCM
NN sin división	0,080095	0,74661	0,69451	0,38848
XGB sin división	0,08668	0,71008	0,62426	0,342465
NN por temporada	0,09056	0,72765	0,654135	0,38718
XGB por temporada	0,0697	0,74292	0,690475	0,337605
NN por diámetro	**0,106875**	**0,791235**	**0,77162**	**0,45403**
XGB por diámetro	0,071045	0,741	0,68931	0,37827

→ Se nota que el mejore modelo es con los datos divididos por diámetro usando el algoritmo de rede neuronales.

Tiempos de entrenamiento y ejecución

- Los tiempos de ejecución en cuanto a entrenamientos y prueba de los modelos es importante, ya que la idea de este proyecto es poder reentrenarlo cada cierto tiempo para mejorar su desempeño y tener datos, y por lo tanto patrones, más actualizados.

- Los tiempos de prueba son los tiempos promedio que demora el modelo en clasificar los datos de un sector de facturación del mes de diciembre del año 2018.

Comparación de tiempos modelos datos no-analista.

Modelos	Tiempo entrenamiento (hh:mm:ss)	Tiempo prueba (hh:mm:ss)
NN sin división	12:03:54	00:01:46
XGB sin división	01:15:53	00:00:35
NN por temporada	15:44:36	00:00:20
XGB por temporada	01:57:19	00:00:15
NN por diámetro	37:04:13	00:00:08
XGB por diámetro	02:02:32	00:00:17

Tiempos de entrenamiento y ejecución

- Se nota que en los dos casos las redes neuronales tienen los mayores tiempos de entrenamiento.

- También se nota que los tiempos de ejecución o pruebas son bastante bajos.

Comparación de tiempos modelos datos analista.

Modelos	Tiempo entrenamiento (hh:mm:ss)	Tiempo prueba (hh:mm:ss)
NN sin división	00:26:59	00:00:02
XGB sin división	00:04:29	00:00:09
NN por temporada	00:33:47	00:00:35
XGB por temporada	00:01:05	00:00:04
NN por diámetro	00:34:23	00:00:02
XGB por diámetro	00:06:27	00:00:01

Índice

- Definición de problema
- Exploración de datos
- Preprocesamiento de datos, algoritmo y métricas
- Modelos finales, resultados y tiempos
- **Discusiones y conclusiones**

Conclusiones

- **Se hizo un recorrido por las distintas aristas del problema** de refacturas, dejando claro los costos que tiene para la empresa.
- **Se crearon modelos de machine learning** para solucionar el problema mencionado. Se hicieron pruebas con modelos de árboles de decisión, regresión logística, redes neuronales y XGBoost.
- **Se probo con varias divisiones de los datos** para aislar posibles patrones que pudieron ser encontrados por los algoritmos de aprendizaje. Finalmente, después de varias pruebas se noto que las mejores divisiones de los datos son por temporada y por código de diámetro.
- **Se estudiaron los tiempos de ejecución** del entrenamiento y prueba de los modelos, llegando a la conclusión que se podrían entrenar perfectamente dos modelos de datos analista y dos modelos de datos no-analista.

Propuestas y trabajos futuros

- **Se implementará en la nube** para un funcionamiento más eficaz
- Se nota que a pesar de que los tiempos de entrenamiento de los modelos son elevados y necesitan de una implementación en la nube, los tiempos de prueba son bastante bajos por lo que se **hace posible implementarlos en computadoras personales**.
- Como propuesta más futura, se espera poder idear una **implementación en teléfonos celulares** para que así los analistas tengan aún más a mano la asistencia de los modelos.
- **Se propone que se ocupen varios modelos en la ejecución** del sistema día a día, pero que solo arroje un resultado. Con esto se pretende acotar la complejidad de la decisión del analista y pueda apoyar de mejor manera en el sistema de aprendizaje de máquina.

¡Gracias!

¿Alguna pregunta?

jfarias.practica@esval.cl