Factoring Fivana: Clientes Fugados

Alumnos:

- Ariel Sanabria
- Nicolás Morales

Julio 2022

fivana

Tabla de contenidos:

Presentación de la Empresa

Fivana está dedicada a realizar Factoring, que es una alternativa de financiamiento que se orienta de preferencia a pequeñas y medianas empresas. Consiste en un contrato mediante el cual una empresa traspasa el servicio de cobranza futura de los créditos y facturas existentes a su favor y a cambio obtiene de manera inmediata el dinero a que esas operaciones se refiere, aunque con un descuento.

El mercado de Factoring es muy competitivo, en donde las ofertas para captar clientes son muy fuertes y aún más para fidelizarlos. Para ello las estrategias comerciales y de marketing juegan un rol fundamental en la captación y retención de clientes.

Definición del problema

Uno de los tipos de clientes que más aquejan a la empresa son aquellos llamados como "Clientes Fugados", estos se definen como aquellos clientes que alguna vez se han financiado en la empresa pero que en los últimos tres meses no han realizado ninguna operación.

Para Fivana los Clientes Fugados son un problema importante ya que no tienen alguna forma de anticipar a que un cliente no realicé ninguna operación durante tres meses, por lo que no pueden generar campañas para evitar que el cliente se fuge.

Por lo tanto lograr una estimación de cliente fugado puede generar grandes beneficios para la empresa, sobre todo para obtener una cartera de clientes mucho más fidelizada y que frecuente negociaciones con Fivana.



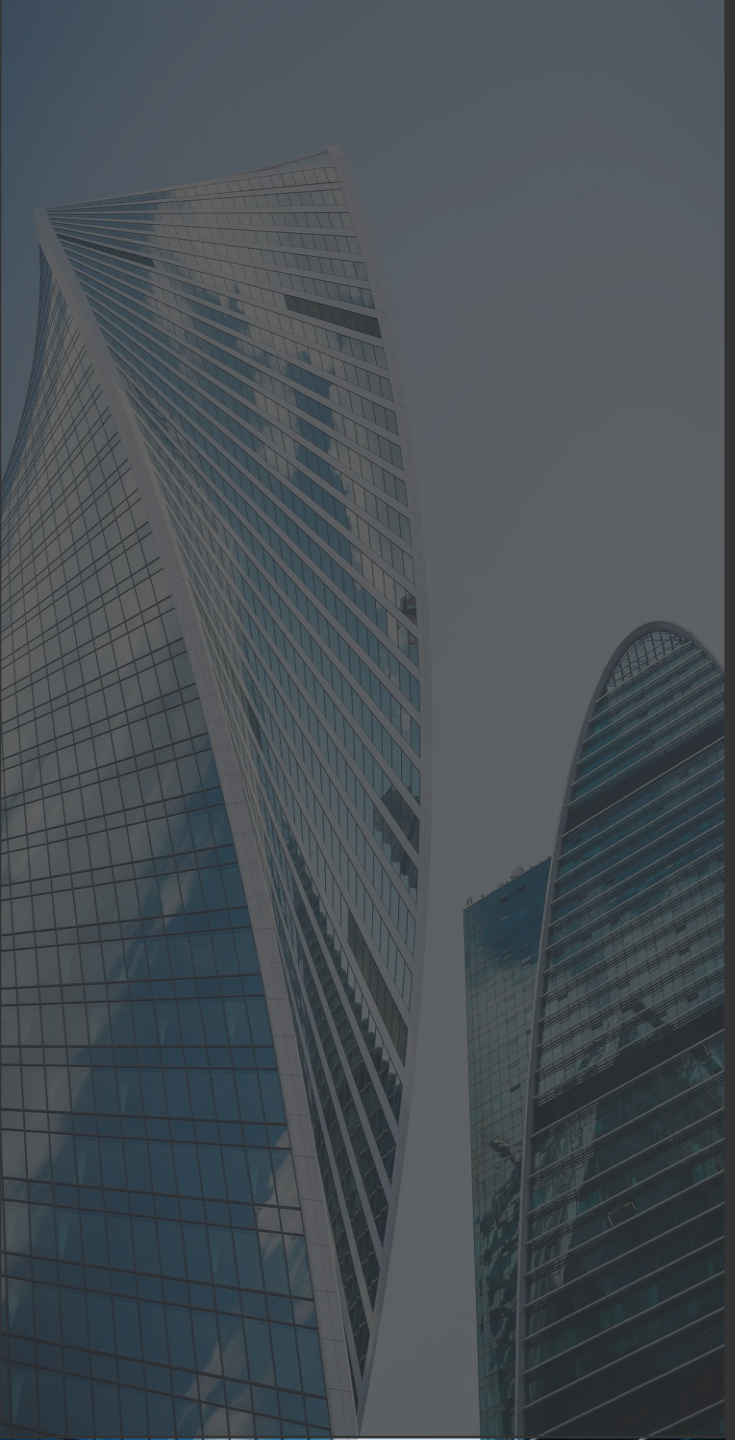

Objetivo y preguntas de la investigación

Objetivo

- Construir modelos de clasificación basados en datos para estimar si un cliente de Fivana se va fugar o no.

Preguntas de la investigación

- ¿Que datos son necesarios?
- ¿Que variables están correlacionadas?
- ¿Que variable va a definir si un cliente se fuga?



Data Acquisition

Se dio principal importancia a los datos de clientes, en torno a las características propias de cada cliente (datos cualitativos), comportamiento de facturación y comportamiento de operación, por lo tanto se definieron tres datasets que tienen como origen los sistemas de información de la empresa. Los cuales son los siguientes:

- Clientes: Se compone de toda la información categórica que posee la empresa acerca de un cliente.
- Operaciones: Es el comportamiento de los financiamientos que a obtenido el cliente con Fivana.
- Facturación: Es el comportamiento de facturación de un cliente a nivel mensual, detallando la cantidad de facturas emitidas y el monto total.

Dichos datasets se obtuvieron de los sistemas internos de información de la empresa.

Variables

Dataset: Facturación

- client_rut
- fecha_añomes_Emisión
- Cant_doc
- monto_facturado

Dataset: Facturación

- customer_rut
- fecha_ope
- Cant_op
- Monto_financiado
- prom_tasa

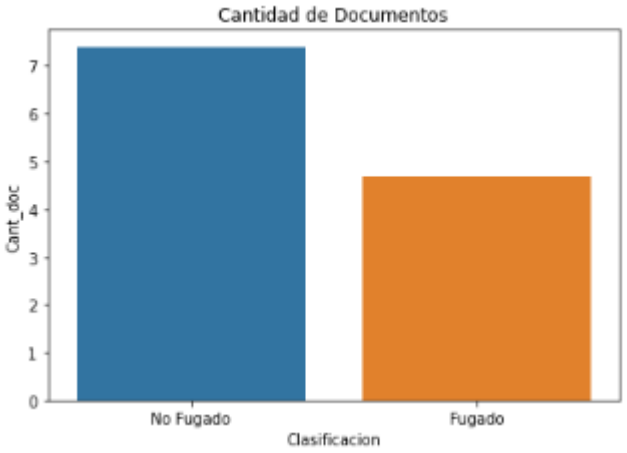
Dataset: Facturación

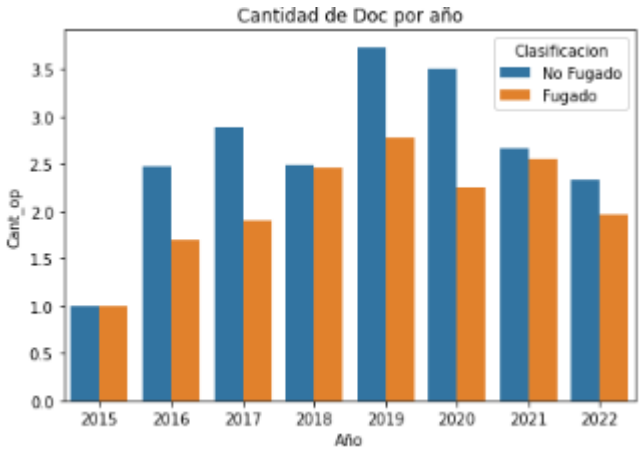
- Rut
- Razón Social
- Calif.
- Rep.Legal
- Contacto
- Registro
- Ejecutivo comercial
- Canal
- Monto financiado
- Verificado legalmente
- Estado
- Verificado por riesgo
- Habilitada cesión externa
- Disponible a fondo
- Ha operado
- Tasa preferencial
- SII
- Última liquidación
- Teléfono
- Correo
- Facturadores

Decisiones tomadas

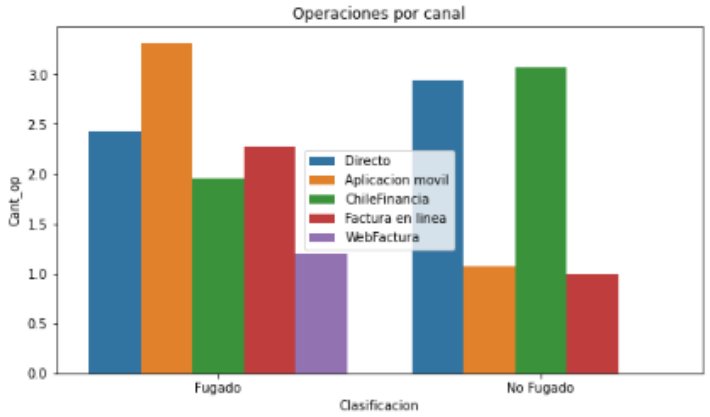
- Se eliminaron los registros de la base de Facturacion, que posean nulos dentro de los campos de fecha de facturación y montos facturados.
- Se eliminaron los valores de la base de Operaciones, que posean nulos dentro de los campos de fecha de cantidad de operaciones, monto financiado y tasa promedio.
- Se unieron todas los tres datasets y se creó una sola base para poder realizar los análisis correspondientes.
- Se agregó la variable target, identificado a los clientes fugados, según una clasificación por periodo.

Análisis Univariable





Se analizo la cantidad de documentos emitidos con la variable target de fuga. En donde podemos observar que la cantidad de documentos emitidos por los clientes fugados es menor al resto. Con esto podemos deducir que los clientes fugados tienen un comportamiento distinto de facturación.



En el gráfico se observa la cantidad de operaciones por canal con la variable de fuga. En la observación se distingue que la aplicación móvil en los fugados sobresale de los demás canales, con aquello podemos insinuar que la aplicación móvil no tiene una performance que satisfaga las necesidades del cliente

Análisis Bivariable

Tablas de contingencia

```
from scipy.stats import chi2_contingency
import numpy as np
value2 = tabla2.values
print(chi2_contingency(value2)[0:3])
```

```
(170.06702617675634, 1.8792690362198735e-08, 80)
```

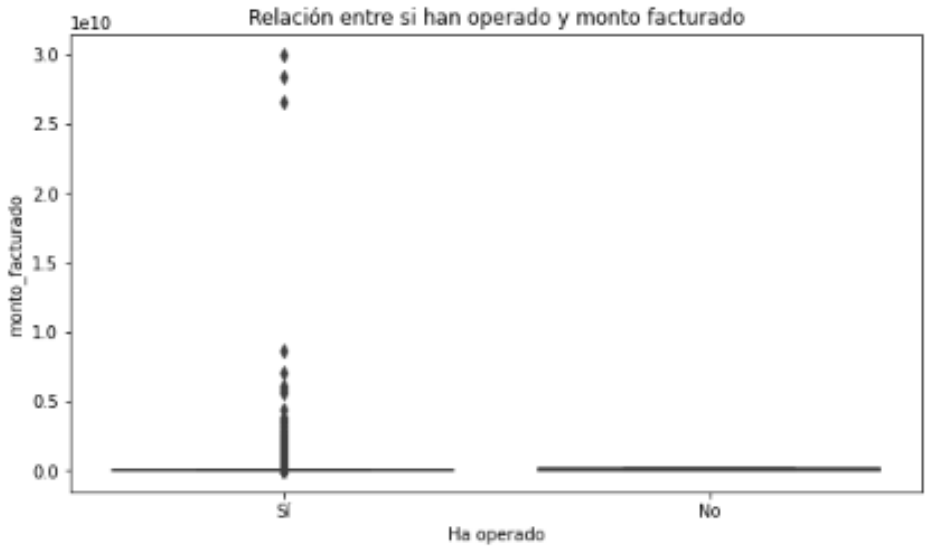
Como Conclusión de las tablas de contingencia, se puede apreciar que no existe una dependencia de las variables categóricas, ya que si usamos un nivel se significancia del 0,05 nuestro Pvalor para ambos casos sería mayor.

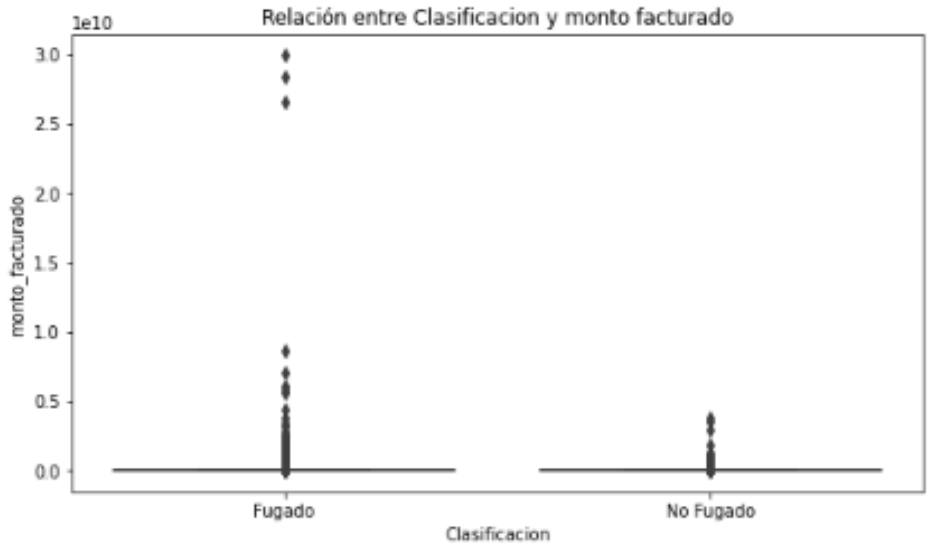
Canal Estado	Aplicacion movil	ChileFinancia	Contador	Directo	Facto	Factura en linea	Laudus	OpenDTE	WebFactura
Activo	0.001878	0.073662	0.000939	0.649940	0.000134	0.002147	0.000134	0.000134	0.000671
Inactivo: Cliente insatisfecho (Enojado)	0.000000	0.000134	0.000000	0.001744	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: El cliente no esta facturando	0.000000	0.001744	0.000000	0.010197	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: Linea de credito	0.000000	0.000537	0.000000	0.003891	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: NA	0.000000	0.000134	0.000000	0.162351	0.000000	0.001208	0.000000	0.000000	0.000403
Inactivo: No califica cliente	0.000000	0.001342	0.000000	0.008185	0.000000	0.000134	0.000000	0.000000	0.000000
Inactivo: No califican deudores	0.000000	0.000134	0.000000	0.011002	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: No esta interesado	0.000000	0.000805	0.000000	0.008587	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: No necesita financiamiento ahora	0.000000	0.003220	0.000000	0.024420	0.000000	0.000268	0.000000	0.000000	0.000000
Inactivo: Opera con competencia	0.000000	0.000939	0.000000	0.010734	0.000000	0.000000	0.000000	0.000000	0.000000
Inactivo: Prejudicial - Judicial	0.000134	0.001476	0.000000	0.016503	0.000000	0.000134	0.000000	0.000000	0.000000

Análisis Bivariable

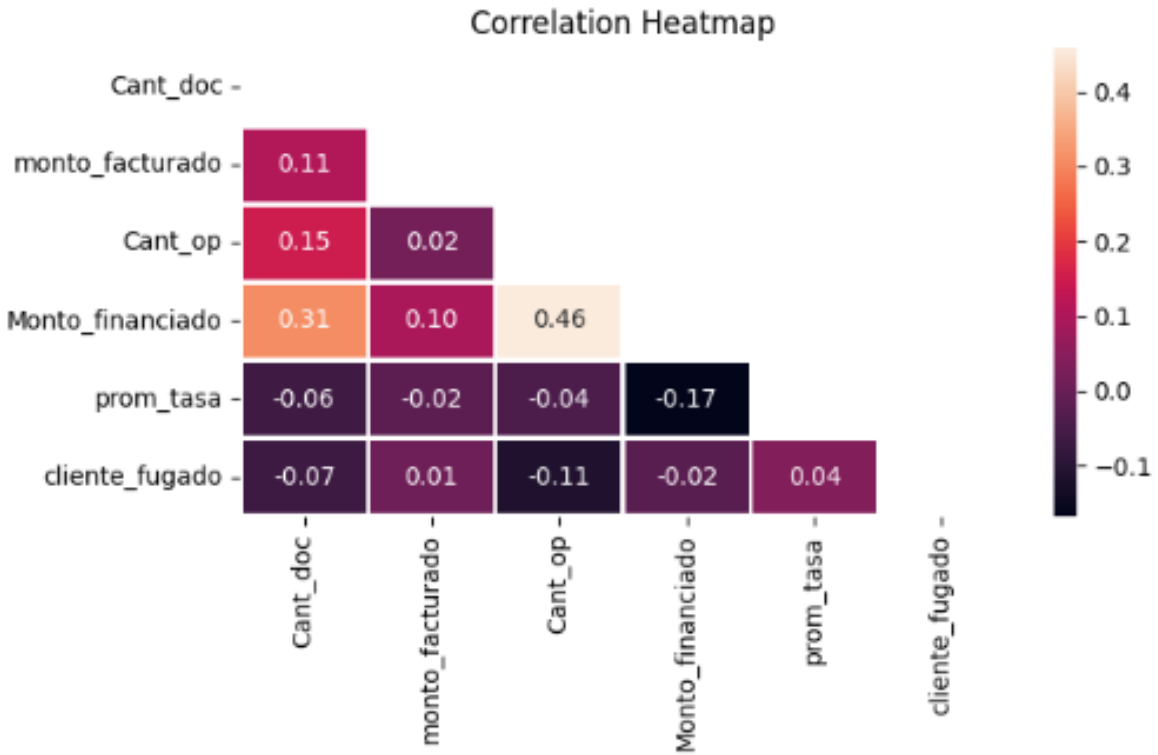
BOXPLOT

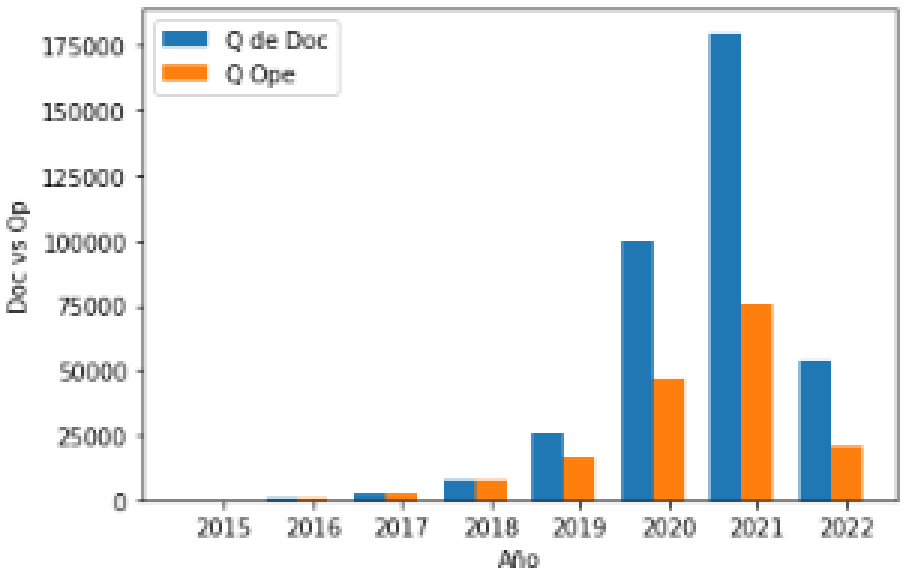
Aquellos clientes que no han operado tienen mayor dispersión en su monto facturado, lo que nos da como conclusión que la empresa apunta a clientes con una facturación más acotada.





Análisis Multivariable



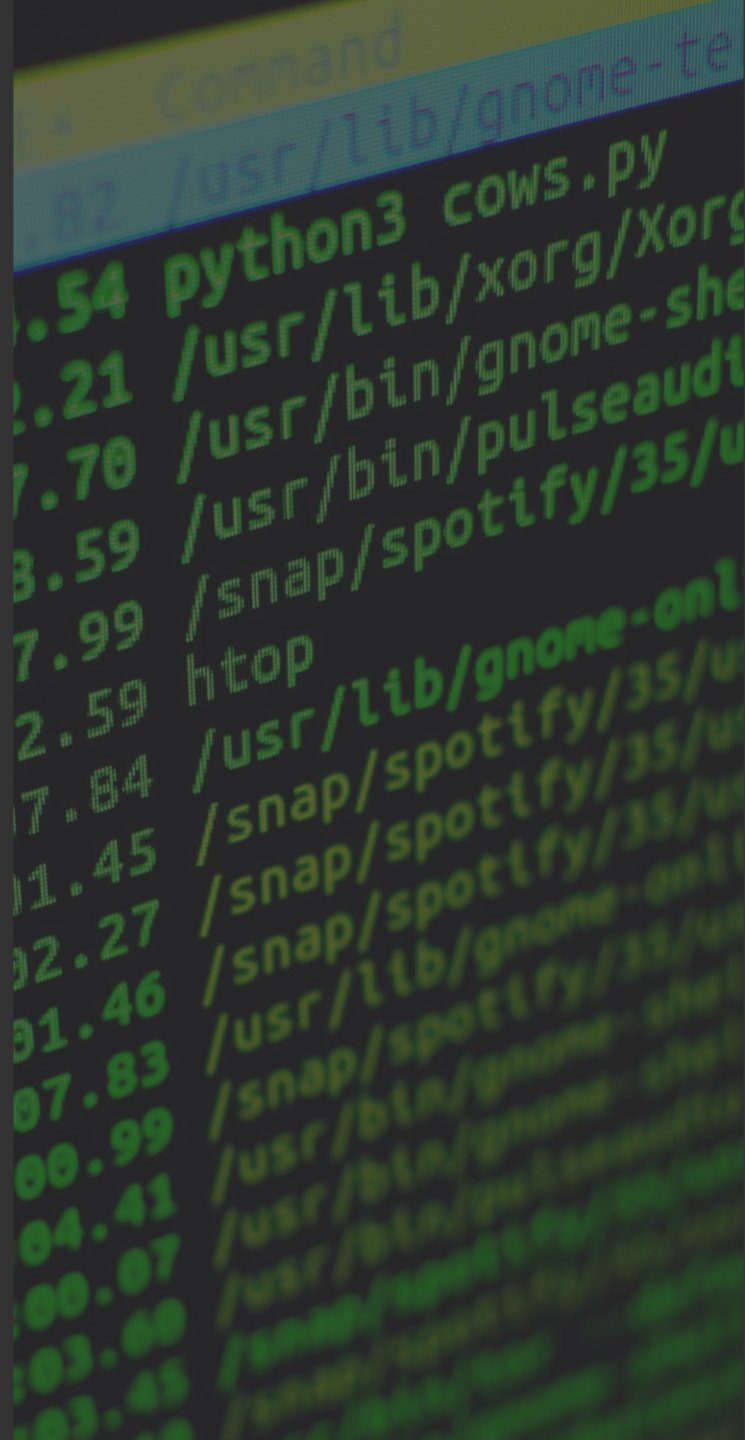


Algoritmos Tentativos

Árbol de decisión

- Se toma un 20% para test y 80% para train.

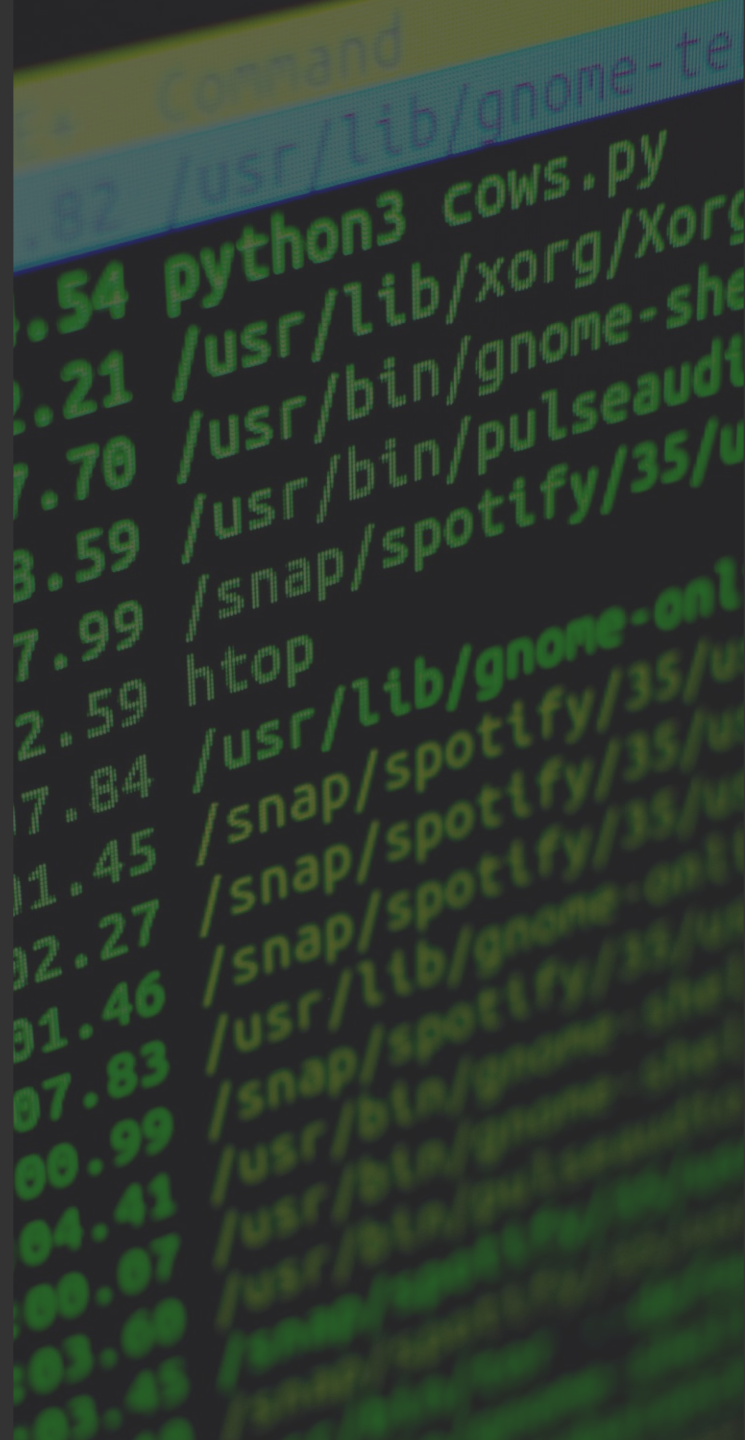
	precision	recall	f1-score	support
0	0.71	0.27	0.39	4907
1	0.68	0.93	0.78	8092
accuracy			0.68	12999
macro avg	0.69	0.60	0.59	12999
weighted avg	0.69	0.68	0.63	12999



Regresión logística

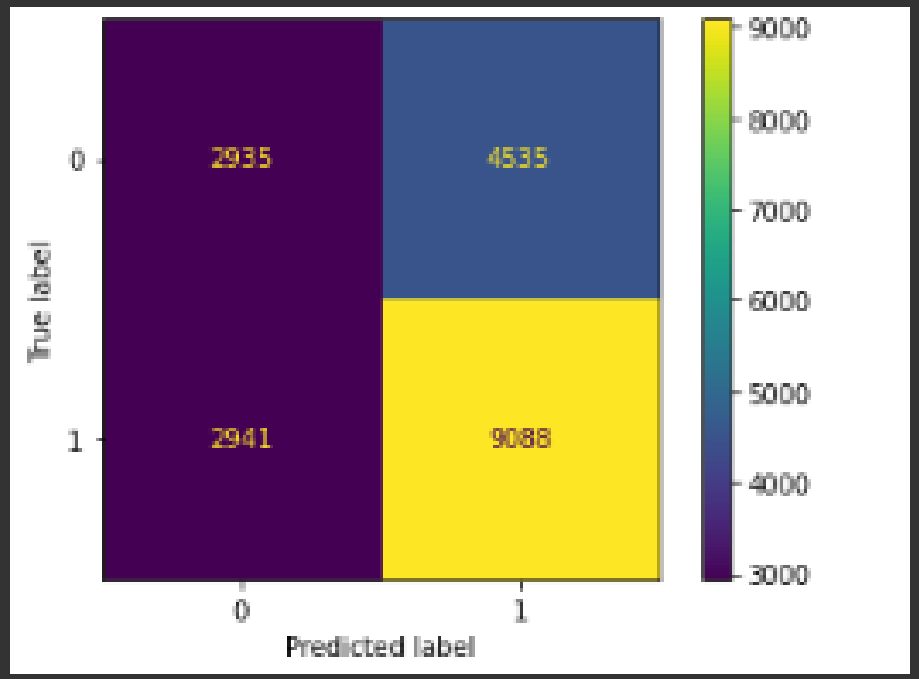
- Se toma un 20% para test y 80% para train.

	precision	recall	f1-score	support
0	0.00	0.00	0.00	24669
1	0.62	1.00	0.77	40325
accuracy			0.62	64994
macro avg	0.31	0.50	0.38	64994
weighted avg	0.38	0.62	0.48	64994



KNN

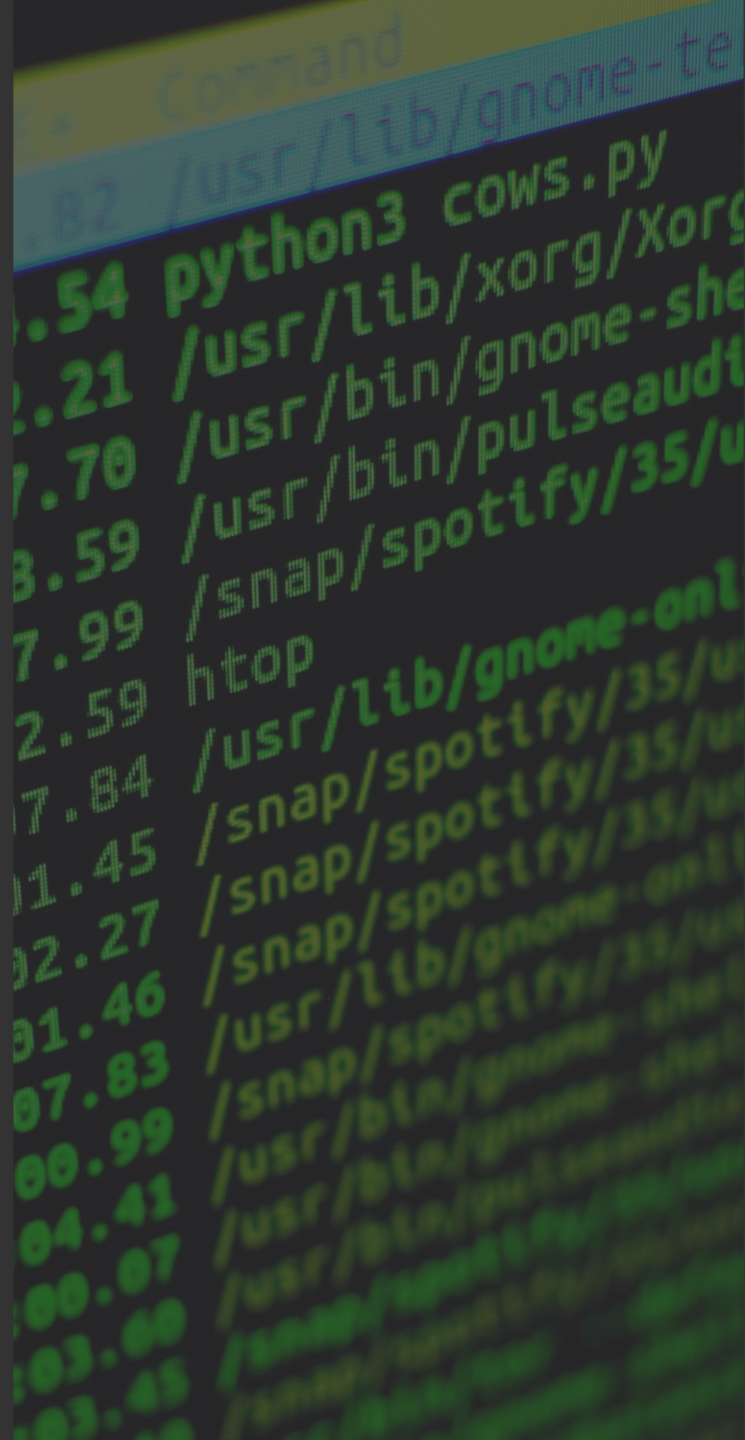
- Se toma un 30% para test y 70% para train.



Random Forest

- Se toma un 20% para test y 80% para train.

	precision	recall	f1-score	support
0	0.87	0.71	0.78	4907
1	0.84	0.94	0.89	8092
accuracy			0.85	12999
macro avg	0.86	0.82	0.83	12999
weighted avg	0.85	0.85	0.85	12999

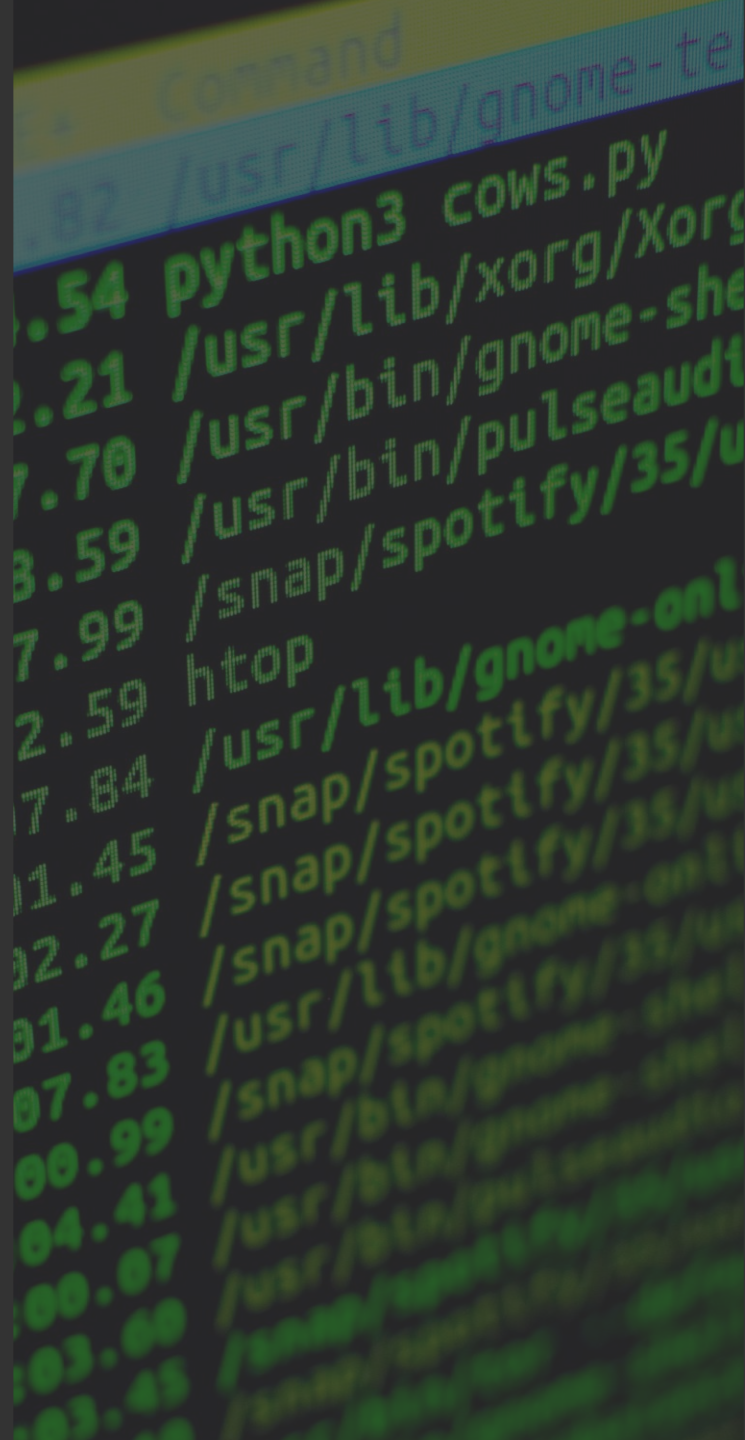


Boostings Models

AdaBoost Classifier

- Se toma un 30% para test y 70% para train.

```
[ ]  print("Accuracy:",metrics.accuracy_score(y_test, y_pred))

     Accuracy: 0.6478959919993845
```

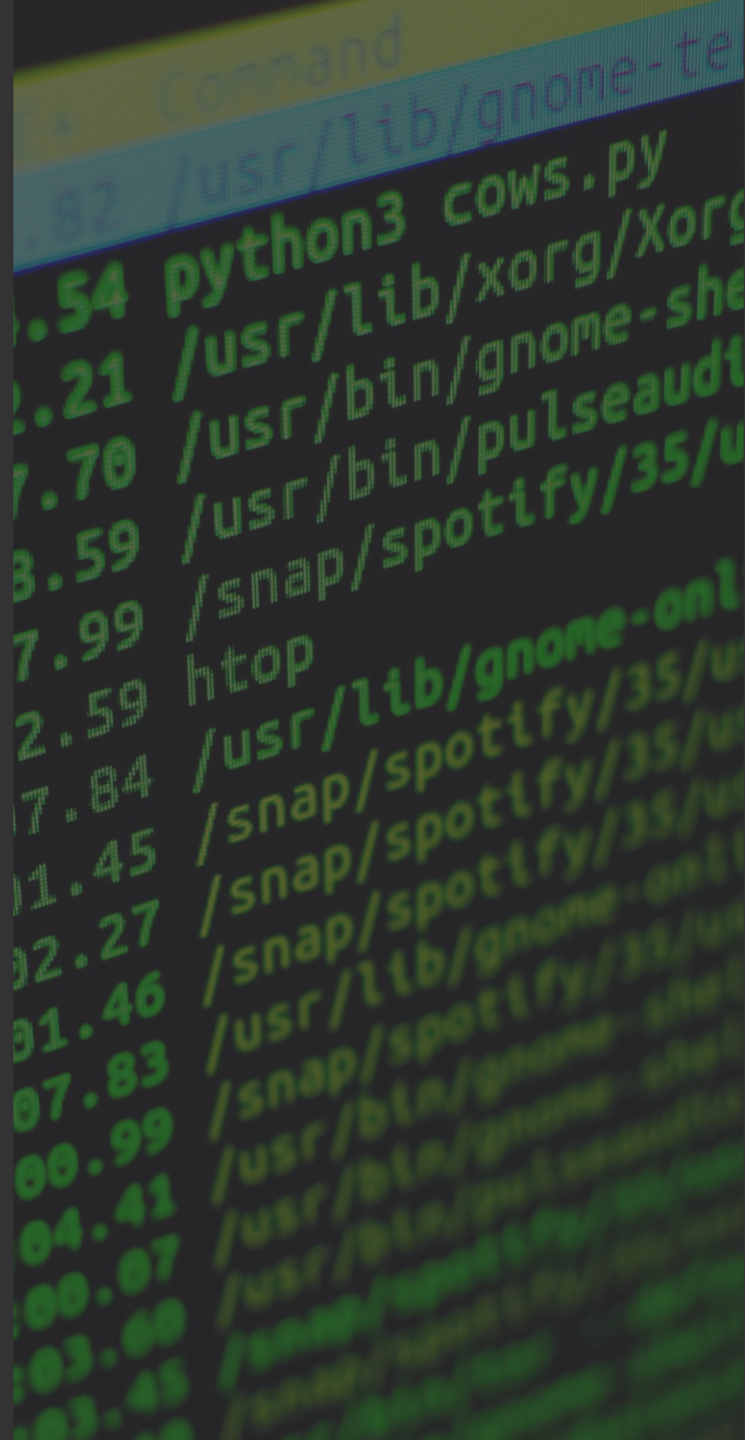


XGBoost

- Se toma un 20% para test y 80% para train.

```
from sklearn.metrics import accuracy_score

#Calculo el accuracy en Train
train_accuracy = accuracy_score(y_train, y_train_pred)

#Calculo el accuracy en Test
test_accuracy = accuracy_score(y_test, y_test_pred)

print('% de aciertos sobre el set de train:', train_accuracy)
print('\n % de aciertos sobre el set de test:',test_accuracy)
```

```
% de aciertos sobre el set de entrenamiento: 0.9032214636022694

 % de aciertos sobre el set de evaluación: 0.8536041233941072
```

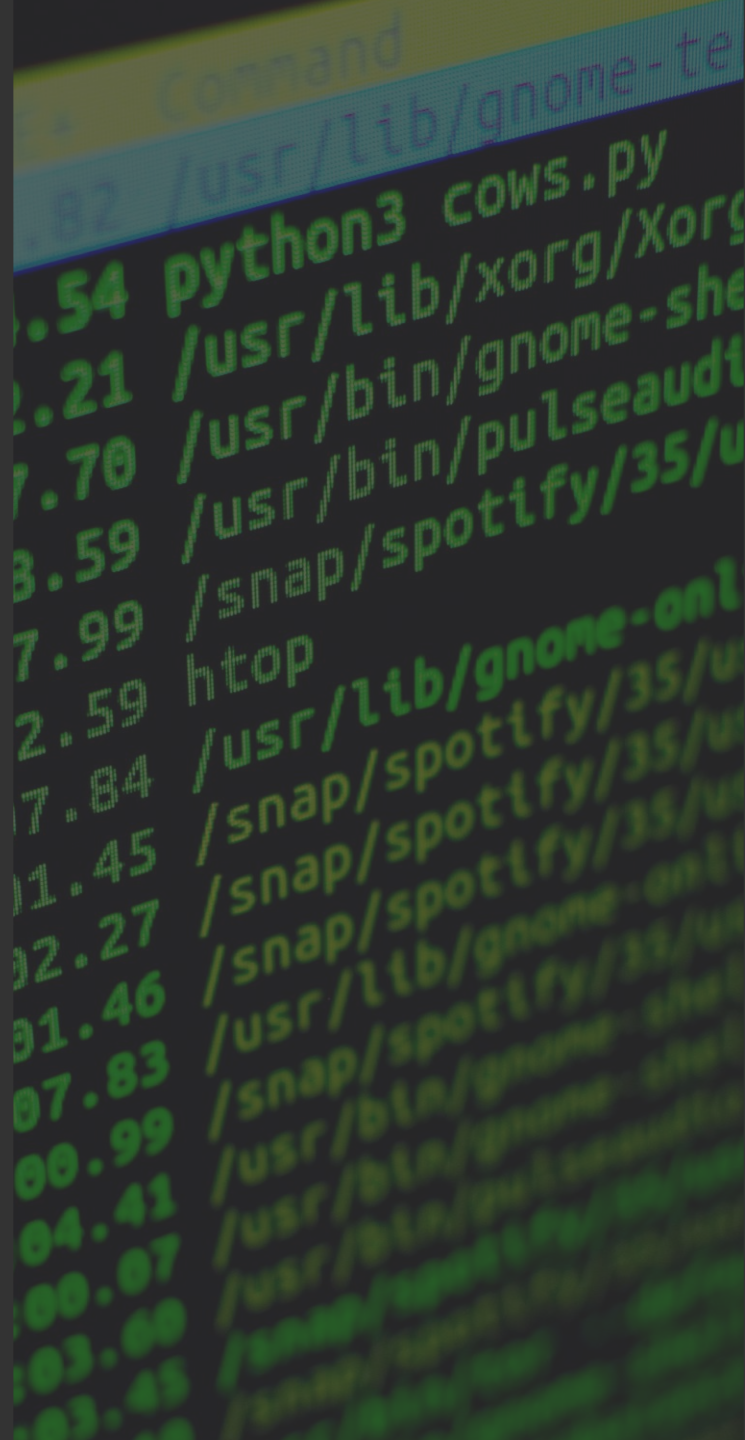


LightGBM

- Se toma un 20% para test y 80% para train.

```
#Accuracy
from sklearn.metrics import accuracy_score
accuracy=accuracy_score(y_pred, y_test)
print('LightGBM Model accuracy score: {0:0.4f}'.format(accuracy_score(y_test, y_pred)))
```

```
LightGBM Model accuracy score: 0.6556
```



Evaluación de resultados

- Dados los resultados de los algoritmos tentativos de clasificación y Boostings se tomó la decisión de realizar las mejoras a Random Forest que tuvo un accuracy 0.85 y a XGBoost con un accuaracy del 0.85

Mejoras de Algoritmos

Random Forest

- Stratified- K fold (clasificacion) - Radom Forest

```
Iteracion:  10  Accuracy: 0.8601323280504694

Accuracy promedio: 0.8591716169351498
```

- Randomized Search CV - Radom Forest

```
[ ]  print (f'Train Accuracy - : {rf_RandomGrid.score(X_train,y_train):.3f}')
     print (f'Test Accuracy - : {rf_RandomGrid.score(X_test,y_test):.3f}')

     Train Accuracy - : 0.998
     Test Accuracy - : 0.860
```

XGBoost

- Stratified- K fold (clasificacion) - Radom Forest

```
Mejores parametros {'subsample': 1.0, 'min_child_weight': 1, 'max_depth': 20, 'gamma': 0.5, 'criterion': 'entropy', 'colsample_bytree': 0.6}
Mejor score de CV 0.8979901571308091
Accuracy del modelo = 0.9013
```

Conclusiones

Los resultados luego de las mejoras de los algoritmos fueron muy gratificantes para ambos modelos. En Random Forest, el resultado es de un 86%, el cual consideramos que es aceptable para el objetivo de la investigación pero por otro lado se tiene el XGBoos que luego de las mejoras alcanzo el 90% de predicción de clientes fugados, es por ello que recomendamos dicho modelo para definir estrategias comerciales mucho más especificas.

Futuras lineas

Tenemos la convicción que el resultado de la investigación podría generar una mejor performance de las campañas de marketing y decisiones comerciales.

Creemos que la empresa tiene una gran fuente de información, que podría alimentar distintos algoritmos parar diferentes problemas de negocio, esperamos que esta investigación de pie a muchos modelos más de machine learning para la empresa.