María Ana Ortiz - Álvaro Pabón - Jorge López

CLASIFICACIÓN DE CLIENTES: CASO DE UNA FINANCIERA DE CRÉDITOS DE LIBRANZAS

I. Resumen.

El objetivo de este caso de estudio es generar una segmentación de los clientes actuales de una compañía financiera que ofrece créditos de libranzas.
A pesar de que las libranzas son un instrumento de crédito relativamente estandarizado y bien reconocido en la industria financiera, el perfil de las personas que los solicitan es, por decirlo menos, poco homogéneo. Derivado de la necesidad de poder ofrecer productos de crédito que se adapten cada vez más a las características y necesidades de los clientes, resulta indispensable segmentar eficientemente a clientes potenciales.

II. Introducción.

¿Cómo puede una compañía de financiamiento tener una mayor penetración de mercado haciendo uso de la analítica de datos?
Con de la crisis económica producto de la pandemia por Covid-19, la entidad de financiamiento "Tu Libranza" ha tenido complicaciones para regresar al volumen previo de colocación de créditos. Visto que el sector financiero ha tenido una importante reactivación económica, la Junta Directiva de la entidad se pregunta si sus productos estandarizados se ajustan realmente a las características de su población objetivo.
"Tu Libranza" ha determinado que a través de una segmentación clara de sus clientes potenciales podría generar una oferta de productos más atractiva que conduciría, finalmente, a lograr una mayor penetración de mercado.
El problema que se pretende resolver corresponde a una clasificación de clientes que utilizará herramientas y algoritmos de aprendizaje no supervisado.

III. Materiales y Métodos.

Descripción de los datos:
La información con la cual se dispone para el estudio corresponde a la base de clientes actuales de la financiera. En total son 30.776 registros de créditos de libranza con 37 variables cada uno, de cuales, en su mayoría corresponden a variables de información de crédito.

Las variables, al ser la mayoría tan específica, se vuelven complejas de describir una a una. Por esta razón se decidió describir únicamente las variables cuya interpretación es más sencilla de entender:

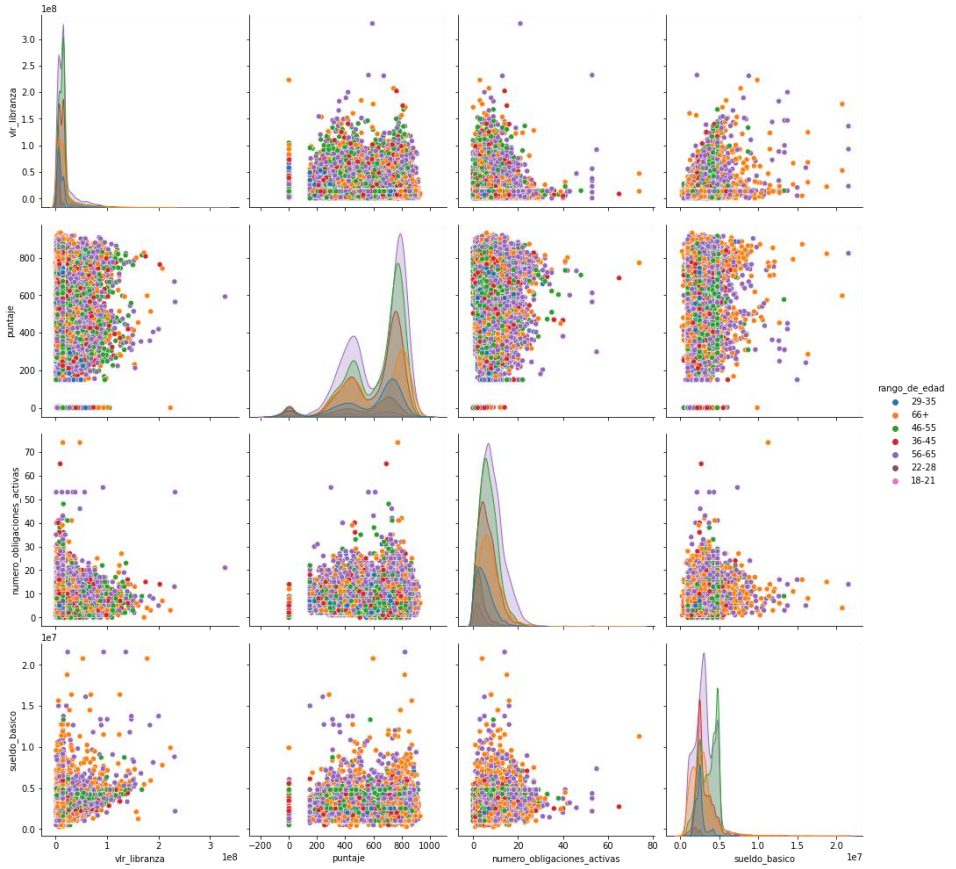
Valor de libranza / Puntaje de crédito / Número de obligaciones / Género / Edad / Sueldo

Del total de 30,776 registros, el 51% son mujeres, el 35% son hombres y el 14% carece de información de género. Sin embargo, independientemente del género, las variables se distribuyen de una forma muy similar.

Adicionalmente, se realizó un análisis por 7 grupos etarios, para los cuales la proporción de registros es la siguiente:

Grupo Etario	Proporción	Grupo Etario	Proporción
18-21	1.5%	**46-55**	23.9%
22-28	4.4%	**56-65**	31.4%
29-35	7.4%	**66+**	14.0%
36-45	17.4%		

Más allá de la diferencia por el número de registros en cada grupo etario, se puede observar que el grupo con un mayor sueldo básico es aquel de personas entre 46 y 55 años:



La información cuenta con las siguientes 37 variables:

ID: Cédula anonimizada
vlr_cuota: Valor cuota del crédito
vlr_libranza: Valor presente del crédito
Puntaje: Score de bureau de crédito
numero_obligaciones_activas: Cantidad de obligaciones financieras son saldo mayor a cero
genero: Femenino o Masculino
rango_de_edad: Edad de la persona en rangos
cartera_bancaria: pendiente por identificar
cartera_bancaria_numero_creditos: Cantidad total de créditos en el sector financiero.
cartera_bancaria_valor_inicial: Valor total inicial de créditos en el sector financiero.
cartera_bancaria_valor_saldo: Valor saldo total de créditos en el sector financiero.
cartera_bancaria_valor_cuotas: Valor total cuotas de créditos activos en el sector financiero.
cartera_bancaria_valor_mora: Valor mora total de créditos activos en el sector financiero.
informacion_tdc_numero_tdc: Cantidad total de tarjeta de créditos activas.
informacion_tdc_valor__cupos: Cupo total de tarjeta de créditos activas.
informacion_tdc_valor__utilizado: Saldo utilizado en la tarjeta de crédito.
informacion_tdc_valor__cuotas: Cuota total a pagar de las tarjetas de crédito activas.
informacion_tdc_valor_mora: Valor total en mora de las tarjetas de crédito.
sector__real_numero_creditos: Cantidad de créditos activos en el sector real.
sector__real_valor_inicial: Valor total inicial de créditos activos en el sector cooperativo.

sector__real_valor_saldo: Valor saldo total de créditos activos en el sector real.
sector__real_valor_cuotas: Valor total cuotas de créditos activos en el sector real.
sector__real_valor_mora: Valor mora total de créditos activos en el sector real.
sector__cooperativo_numero_creditos: Cantidad de créditos activos en el sector cooperativo.
sector__cooperativo_valor_inicial: Valor total inicial de créditos activos en el sector cooperativo.
sector__cooperativo_valor_saldo: Valor saldo total de créditos activos en el sector cooperativo.
sector__cooperativo_valor_cuotas: Valor total cuotas de créditos activos en el sector cooperativo.
sector__cooperativo_valor_mora: Valor mora total de créditos activos en el sector cooperativo.
obligaciones_mora_30: Número de obligaciones en mora de 30
obligaciones_mora_60 Número de obligaciones en mora de 60
obligaciones_mora_90 Número de obligaciones en mora de 90
obligaciones_mora_120 Número de obligaciones en mora de 120
ultimo_ano_moras_30: Cantidad de productos en mora 30 en el último año.
ultimo_ano_moras_60 Cantidad de productos en mora 60 en el último año.
ultimo_ano_moras_90 Cantidad de productos en mora 90 en el último año.
ultimo_ano_moras_120 Cantidad de productos en mora 120 en el último año.
sueldo_basico: Salario de la persona

Proceso de limpieza:
Contamos con 8,925 variables perdidas en sueldo básico, con una primera iteración eliminamos las columnas de cédula y sueldo básico, posteriormente eliminamos las 284 filas de rango_de_edad que contiene valores perdidos. Las variables categóricas género y rango de edad las convertimos a dummies. Revisamos que el puntaje fuera superior a 7 por que debajo de ese número significa que no tiene o no califica, creamos nuevas columnas a partir de otras "total_mora", "mora_baja" y "mora_alta", desechamos las columnas con las cuales se construyó las nuevas.
Aplicar logaritmo sobre las variables nos ayudó a mejorar la segmentación y finalmente, estandarizamos los datos. Como resultado obtenemos un *dataframe* con 29,470 filas y 38 columnas (los datos iniciales tenían 30,766 filas y 37 columnas).

Transformación y Limpieza de Datos

```
datos2 = datos.drop(['cedula', 'sueldo_basico'], axis = 1).dropna()
datos3 = datos2[datos2['puntaje'] > 7]
datos3 = pd.get_dummies(datos3)
datos3['total_mora'] = datos3['cartera_bancaria_valor_mora'] + datos3['sector__real_valor_mora']
+ datos3['sector__cooperativo_valor_mora'] + datos3['informacion_tdc_valor_mora']
datos3['mora_baja'] = np.where((datos3['obligaciones_mora_30'] + datos3['obligaciones_mora_60']) > 0, 1, 0)
datos3['mora_alta'] = np.where((datos3['obligaciones_mora_90'] + datos3['obligaciones_mora_120']) > 0, 1, 0)

datos3 = datos3.drop(['cartera_bancaria_valor_mora', 'sector__real_valor_mora', 'sector__cooperativo_valor_mora',
                      'informacion_tdc_valor_mora', 'obligaciones_mora_30', 'obligaciones_mora_60',
                     'obligaciones_mora_90', 'obligaciones_mora_120'], axis = 1)

datos4 = datos3.applymap(lambda x: 0.1 if x ==0 else x)

datos5 = datos4.applymap(lambda x: np.log(x))
datos6 = StandardScaler().fit_transform(datos5)
```

Utilizamos primero un análisis de componentes principales De acuerdo a la siguiente grafica podemos analizar que con 8 componentes principales alcanzamos una variabilidad del 90%:

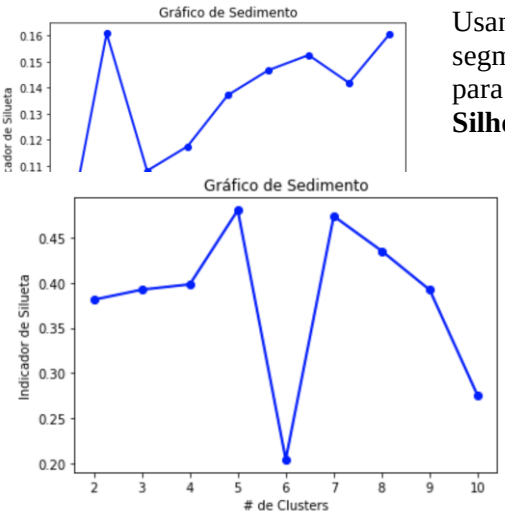
Variables con más variabilidad según componentes principales:

	Pesos PC1
numero_obligaciones_activas	0.319407
informacion_tdc_numero_tdc	0.278461
informacion_tdc_valor__utilizado	0.270422
informacion_tdc_valor__cupos	0.267157
cartera_bancaria_numero_creditos	0.260740
informacion_tdc_valor__cuotas	0.258082
cartera_bancaria_valor_inicial	0.252594
cartera_bancaria_valor_saldo	0.243974
cartera_bancaria_valor_cuotas	0.238226
sector__real_numero_creditos	0.208626

	Pesos PC2
total_mora	0.340239
ultimo_ano_moras_60	0.329197
ultimo_ano_moras_90	0.326266
mora_alta	0.320283
ultimo_ano_moras_120	0.302532
ultimo_ano_moras_30	0.294781
cartera_bancaria	0.185964
mora_baja	0.180912
sector__real_numero_creditos	0.121335
sector__real_valor_cuotas	0.112704

	Pesos PC3
sector__cooperativo_valor_cuotas	0.458383
sector__cooperativo_valor_saldo	0.458381
sector__cooperativo_valor_inicial	0.457840
sector__cooperativo_numero_creditos	0.453568
ultimo_ano_moras_30	0.051278
ultimo_ano_moras_60	0.050280
rango_de_edad_46-55	0.043954
ultimo_ano_moras_90	0.043898
mora_baja	0.041850
mora_alta	0.039817

	Pesos PC4
cartera_bancaria_valor_cuotas	0.276081
cartera_bancaria_valor_saldo	0.274453
cartera_bancaria_valor_inicial	0.261757
cartera_bancaria_numero_creditos	0.247626
cartera_bancaria	0.196115
ultimo_ano_moras_60	0.131632
ultimo_ano_moras_90	0.128749
ultimo_ano_moras_30	0.115788
ultimo_ano_moras_120	0.088168
mora_alta	0.085626



Usando estos 8 componentes principales, realizamos segmentación a través de KMedoids desde 2 hasta 10 clusters para determinar el mejor indicador.
Silhouette Score: 0.1608 con 3 clusters es el mejor resultado.

Realizamos otra prueba para mejorar el indicador teniendo en cuenta las columnas de **'vlr_libranza', 'total_mora', 'cartera_bancaria_valor_saldo', 'sector__real_valor_saldo', 'sector__cooperativo_valor_saldo','mora_alta', 'mora_baja'** teniendo en cuenta la lectura de importancia en PCA, pero al correr el kmediods se hizo sin componentes principales:
Con 5 grupos de clientes logramos mejorar el Silhouette Score: 0.4805.

IV. Resultados y Discusión

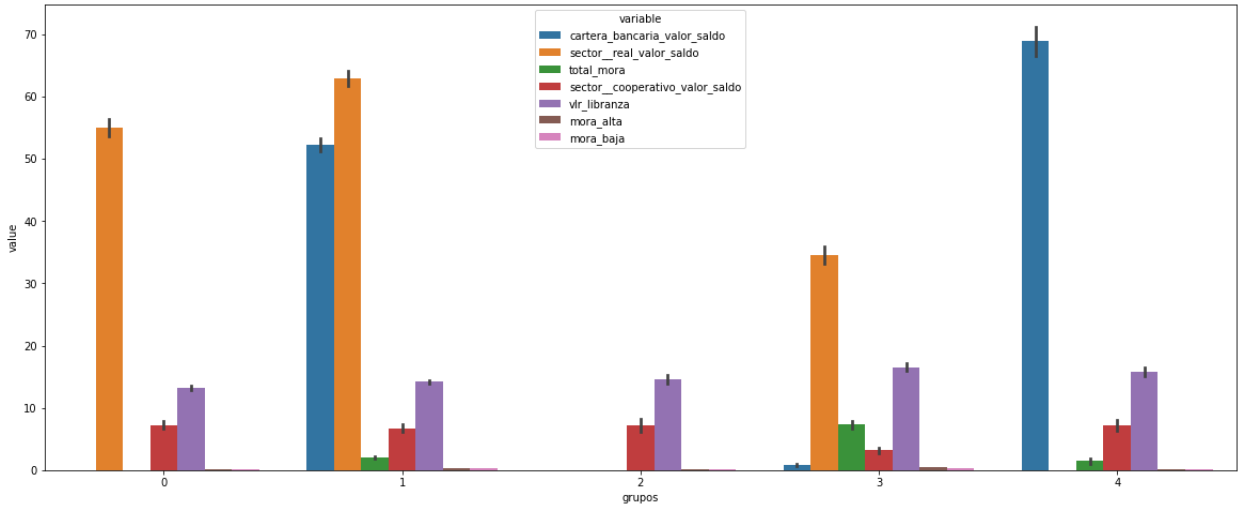
A partir de la selección de variables y análisis de cantidad de clusters para el método de KMedoids y el grafico de sedimento, se decidió también buscar la asignación de clusters por medio del Algoritmo KMeans.
Para el algoritmo KMeans se realizó con 5 clusters donde el Silhouette Score fue de 0.45, y la varianza intra clusters fue de 3274391.19.
Al comparar las dos métricas obtenidas por los algoritmos, se determinó que los clusters resultantes de KMedoids tiene menor valor de intra-varianza lo que hace que sus clusters sean más similares internamente.

Los clusters obtenidos se ven en la siguiente tabla:

grupos	vlr_libranza	total_mora	cartera_bancaria_valor_saldo	sector__real_valor_saldo	sector__cooperativo_valor_saldo	mora_alta	mora_baja
0	13.160050	0.000000	0.000000	55.028592	7.242888	0.141687	0.148635
1	14.219267	2.081429	52.222735	63.011505	6.667153	0.245479	0.246646
2	14.575838	0.001049	0.000000	0.000000	7.222028	0.150979	0.135559
3	16.472839	7.292345	0.785093	34.524213	3.172952	0.514149	0.319728
4	15.772622	1.450573	68.930516	0.000000	7.142192	0.175430	0.183166

Como resultado podemos ver que tenemos 5 grupos de clientes:
0: Clientes al día, con un perfil de clientes que toma prestado principalmente en el sector real y algo en el sector cooperativo.
1: Clientes con perfil principalmente bancario y del sector real por sus saldos.
2: Clientes con muy bajo endeudamiento en todos los sectores
3: Clientes que presentan moras (alto riesgo)
4: Clientes con perfil bancario principalmente

En esta grafica se puede observar cómo los clusters varían en cada variable, una forma más fácil de notar las características anterior mente enunciadas.



V. Conclusión

Los segmentos encontrados para “Tu Libranza” permitirán que pueda ofrecer y desarrollar productos por tipo de cliente, ya que cada uno tiene características que se lograron diferenciar entre los demas.
El algoritmo de implementado de selección de variables y de clustering permitió obtener los segmentos, principalmente la unión de variables para acumular valores.
Finalmente se considera que los segmentos se podrían mejorar si se tuviera más información del comportamiento bancario, ejemplo que tan seguido hacen un aporte a su deuda.

VI. Bilbliografia

“Sklearn_extra.cluster.kmedoids,” scikit. [Online]. Available: https://scikit-learn-extra.readthedocs.io/en/stable/generated/sklearn_extra.cluster.KMedoids.html. [Accessed: 25-Sep-2022].
“sklearn.cluster.KMeans” scikit. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html [Accessed: 25-Sep-2022].