Preprocessing

Iker Meneses Sales

20-11-2023

Tras haber analizado la base de datos de forma estricta y haber aplicado diferentes métodos de clusterización, el siguiente paso será aplicar todo lo visto a una base de datos real. Como bien se sabe, en una base de datos de un banco es normal encontrar que la gran mayoría de clientes no sean morosos, ya que sino la salud de la que gozaría el banco sería pésima. Así pues, ahora se trabajará con una base de datos donde el número de clientes morosos no sea cercano al 50%, sino más bien cercano al 0. Para ello, se ha seleccionado una base de datos desbalanceada.

Así pues, será necesario preprocesar la base de datos nuevamente. Para ello, será óptimo seguir los pasos propuestos por Karina Gibert con el objetivo de desarrollar correctamente el KDD y, así, obtener conclusiones óptimas a partir de nuestros datos.

Para ello, seguiremos 4 grandes bloques:

- Limpieza de datos y estandarización de formato
- Detección y tratamiento de missings
- Detección y tratamiento de outliers
- Feature Engineering

Limpieza de datos y estandarización de formato

Una vez se ha realizado la descriptiva preprocessing y se ha identificado el número de valores missing en nuestra base de datos, es óptimo analizar todas las variables una a una, así como algunas variables categóricas a las cuales se les puede reducir el número de categorías.

Para empezar, se puede apreciar que la variable `OCCUPATION_TYPE` tiene un total de 18 categorías:

Table 1: Distribución inicial de la variable OCCUPATION TYPE

Categoría	Frecuencia
Accountants	163
Cleaning staff	80
Cooking staff	114
Core staff	440
Drivers	301
High skill tech staff	169
HR staff	7
IT staff	8
Laborers	863
Low-skill Laborers	29
Managers	380
Medicine staff	130
Private service staff	46
Realty agents	18
Sales staff	514
Secretaries	26
Security staff	112
Waiters/barmen staff	30
NA	1570

Una buena idea sería combinar algunas categorías con el objetivo de reducir el número de categorías y, además, aumentar el número de individuos por categoría. Seguidamente, se muestran los cambios realizados, donde se han agrupado todos los individuos en 5 categorías en función del capital humano empleado para su puesto:

- Low skill laborers: Engloba las categorías de "security staff", "cooking staff", "cleaning staff", "drivers", "low skill laborers", "waiters staff".
- Low-mid skill laborers: Engloba las categorías de "secretaries", "private service staff" y "laborers".
- Mid skill laborers: Engloba las categorías de "accountants", "HR staff" y "sales staff".
- Mid-high skill laborers: Engloba las categorías de "IT staff", "realty agents" y "core staff".
- High skill staff: Engloba las categorías de "high skill tech staff", "managers" y "medicine staff".

Table 2: Distribución final de la variable OCCUPATION TYPE

Categoría	Frecuencia
High skill laborers	679
Low-mid skill laborers	935
Low skill laborers	666
Mid-high skill laborers	466
Mid skill laborers	684
NA	1570

Este proceso lo repetiremos con la variable `ORGANIZATION_TYPE`:

Table 3: Distribución inicial de la variable ORGANIZATION TYPE

Categoría	Frecuencia
Advertising	8
Agriculture	46
Bank	33
Business Entity Type 1	91
Business Entity Type 2	153
Business Entity Type 3	1118
Cleaning	6
Construction	107
Culture	11
Electricity	8
Emergency	20
Government	152
Hotel	11
Housing	38
Industry: type 1	19
Industry: type 10	2
Industry: type 11	44
Industry: type 12	4
Industry: type 13	2
Industry: type 2	2
Industry: type 3	47
Industry: type 4	12
Industry: type 5	10
Industry: type 6	2
Industry: type 7	25
Industry: type 8	2
Industry: type 9	55
Insurance	7
Kindergarten	119
Legal Services	4
Medicine	191
Military	44
Mobile	4
Other	280
Police	50
Postal	32
Realtor	10
Religion	2
Restaurant	36
School	127
Security	49
Security Ministries	36
Self-employed	617
Services	28
Telecom	8
Trade: type 1	4
Trade: type 2	42
Trade: type 3	53
Trade: type 5	1
Trade: type 6	10
Trade: type 7	135
Transport: type 1	5
Transport: type 2	38
Transport: type 3	13
Transport: type 4	78

Como se puede apreciar, en este caso disponemos de muchísimas categorías, pero es de destacar la categoría XNA, la cual deberíamos sustituir a NA, para después poder imputarle algún valor. Así pues, se ha agrupado cada categoría profesional en función del sector al que se dedica el individuo. Así, la distribución final es la siguiente:

Table 4: Distribución final de la variable ORGANIZATION TYPE

Categoría	Frecuencia
Business and bank	1402
Education	265
Industry and construction	333
Medicine	191
Other	419
Personal services	146
Public services	310
Self-employed	617
Trade and telecom	253
Transport	134

Ahora, esta variable pasa a tener 10 categorías, las cuales representan los diferentes sectores presentes en la economía presente hoy en día.

Así pues, el resto de variables tienen una uniformidad evidente: se puede apreciar cómo las variables categóricas presentan un número de categorías pequeño y, por parte de las variables numéricas, todas están expresadas en las mismas unidades, de forma que no habrá problemas con la manipulación de éstas.

Detección y tratamiento de missings

Para este apartado, trataremos de identificar aquellos valores desconocidos y valorar sobre su aleatoriedad para, posteriormente, imputar valores. Para empezar, es de destacar cómo hay 47 individuos con un coche de 64 años y 11 con un coche de 65. Si nos fijamos en la distribución de esta variable, es muy extraño que haya tantos individuos con valores atípicos, ya que el siguiente valor máximo es 46. Así, se potará por imputar valores nulos a estos individuos.

Seguidamente, pasaremos a imputar diferentes valores a aquellas variables donde hay observaciones sobre las cuales se desconocen sus valores reales. Este paso es necesario, ya que el hecho de disponer de valores desconocidos (también conocidos como NA) dificulta el análisis posterior de la variable.

Una vez hemos recategorizado todas aquellas variables que presentaban problemas, el número de NA por variables es el siguiente:

Table 5: Missings por variable

Categoría	Frecuencia
AMT_INCOME_TOTAL	0
AMT_CREDIT	0
AMT_ANNUITY	0
DAYS_BIRTH	0
OWN_CAR_AGE	3363
AMT_GOODS_PRICE	2
CNT_FAM_MEMBERS	0
CODE_GENDER	0
NAME_INCOME_TYPE	0
NAME_EDUCATION_TYPE	0
NAME_FAMILY_STATUS	0
OCCUPATION_TYPE	1570
ORGANIZATION_TYPE	930
REGION_RATING_CLIENT	0
TARGET	0

Una vez tenemos identificados todos los valores missing de nuestra base de datos, será necesario identificar si éstos son completamente aleatorios (MCAR), aleatorios (MAR), o no aleatorios (MNAR). Para ello, realizaremos el test de Little, el cual indica si los missings disponibles en la base de datos son fruto del azar o si siguen un patrón.

Para este test, diremos que los datos no siguen un patrón si no se rechaza hipótesis nula o, alternativamente, si no encuentra patrones entre los missings. Así pues, este es el resultado:

Table 6: Test de Little

statistic	df	p.value	missing.patterns
3321.21225527829	79	0	7

Como se puede apreciar, el algoritmo ha detectado 7 patrones entre los valores missing, de forma que no se puede decir que hay un patrón aleatorio, de forma que calificaremos nuestros valores missing como MNAR.

Seguidamente, imputaremos los valores por los tres métodos de imputación conocido, pero antes de imputar los valores numéricos, será necesario pasar los NA a categoría `unknown`.

Seguidamente, toca imputar los NA disponibles en las variables numéricas de nuestros datos. Para ello, utilizaremos tres métodos distintos: kNN, MiMMi y MICE. Posteriormente, se comparará la imputación entre estos métodos y se seleccionará el método que resulte una distribución más parecida a la original antes de imputar.

Imputación por criterios estadísticos

En este caso, el objetivo será imputar en función de criterios estadísticos básicos. Para ello, se procederá a imputar valores en función de la media estadística o algún otro estadístico central de distribución.

Imputación por kNN

El algoritmo K-Nearest Neighbors (KNN), es un método de clasificación supervisada, que utiliza la proximidad para hacer clasificaciones o predicciones sobre un punto de datos desconocido. El algortimo, utiliza

un hiperparámetro llamado "k", que representa el número de vecinos más cercanos y el cual se ha obtenido mediante el cálculo de $k = \sqrt{n}$.

A continuación, se crean dos objetos: `fullVariables`, que corresponde a las variables que no presentan ningún dato faltante y `uncompleteVars`, que guarda las variables con missings.

Como se puede observar, se obtiene la imputación de los valores faltantes en el dataframe `df_knn` utilizando el algoritmo descrito previamente.

Imputación por MiMMi

La imputación por MiMMi se realiza utilizando un enfoque basado en clústeres y se utiliza la distancia de Gower como métrica de distancia para medir la similitud entre observaciones.

La función uncompleteVar se define para verificar si hay valores faltantes (representados como NA) en un vector dado.

La función Mode se define para calcular la moda de un vector. Esta función se utiliza más adelante para imputar valores faltantes en variables categóricas.

Se define la función MiMMi.

Se usa la función MiMMi y se obtienen los resultados imputados.

Imputación por MICE

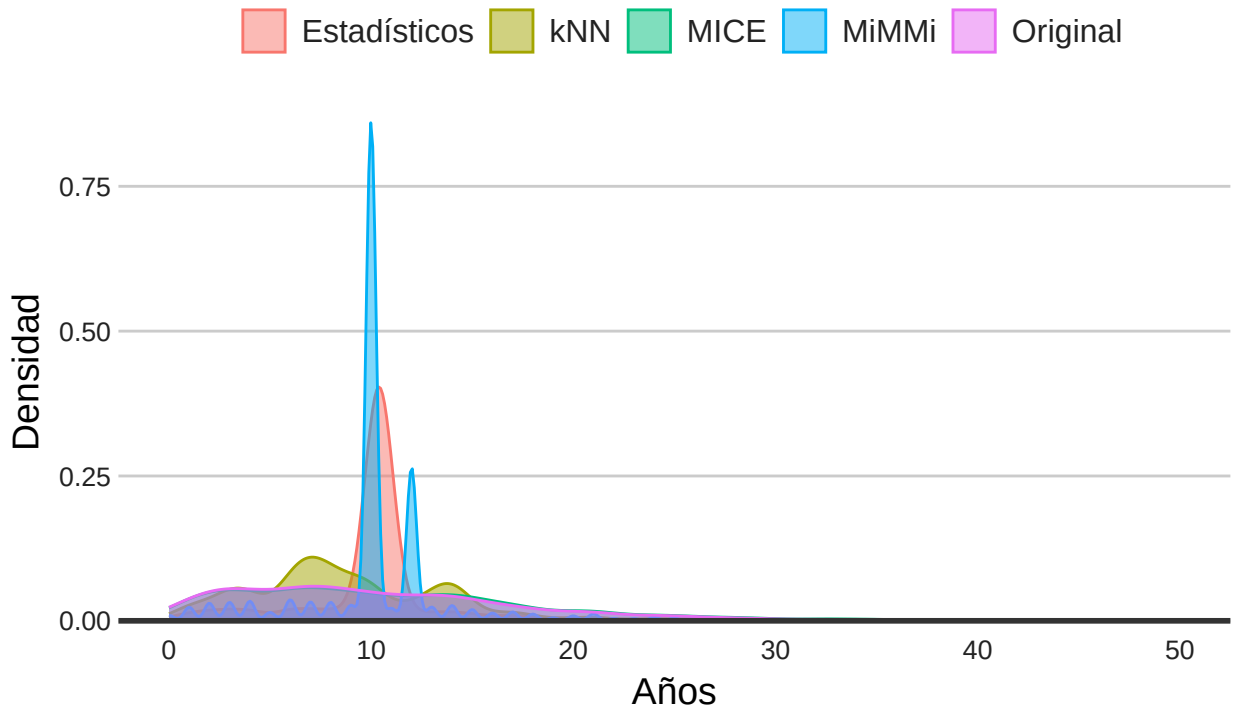
Por último, se recurrirá a imputar a través del MICE como último método de imputación de valores numéricos. El MICE (Multiple Imputation by chained Equations) se basa en un método iterativo a partir del cual se resuelven ecuaciones consecutivamente con el objetivo de imputar valores de la forma más aproximada posible. Así pues, es momento de imputarlo:

Decisión del método de imputación elegido

Llegados a este punto, en el momento de seleccionar el método de imputación elegido para el método de imputación final. En nuestro caso, como únicamente disponemos de dos variables numéricas con missings, podemos comparar la función de densidad de los datos originales contra los imputados por cada método. Así pues, vamos a mirar variable por variable:

OWN_CAR_AGE

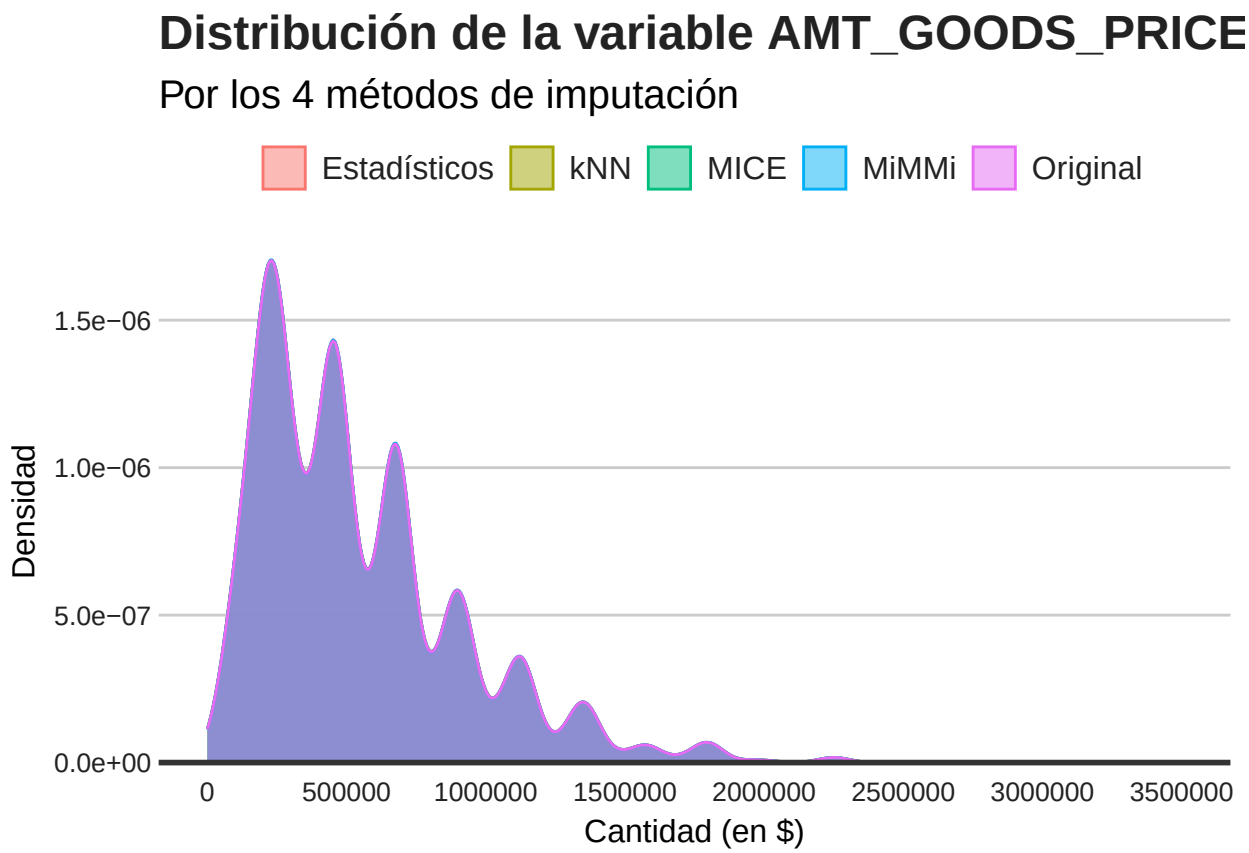
Esta variable es la que presenta más valores no disponibles en nuestra base de datos, de forma que se acepta un mayor margen de error en cuanto a la imputación de valores se refiere. Así, la densidad resultante para cada método es la siguiente:



Como se puede apreciar, hay tres métodos de imputación que claramente se alejan mucho de la distribución inicial de los datos: criterios estadísticos, kNN y MiMMi. Así pues, se puede apreciar como el MICE es el algoritmo que aproxima la densidad de los datos a los originales, de forma que este será el método escogido.

AMT_GOODS_PRICE

Como se ha visto previamente en al descriptiva preprocessing, esta variable únicamente presentaba 3 NA, de forma que la densidad en todos los métodos será muy similar:



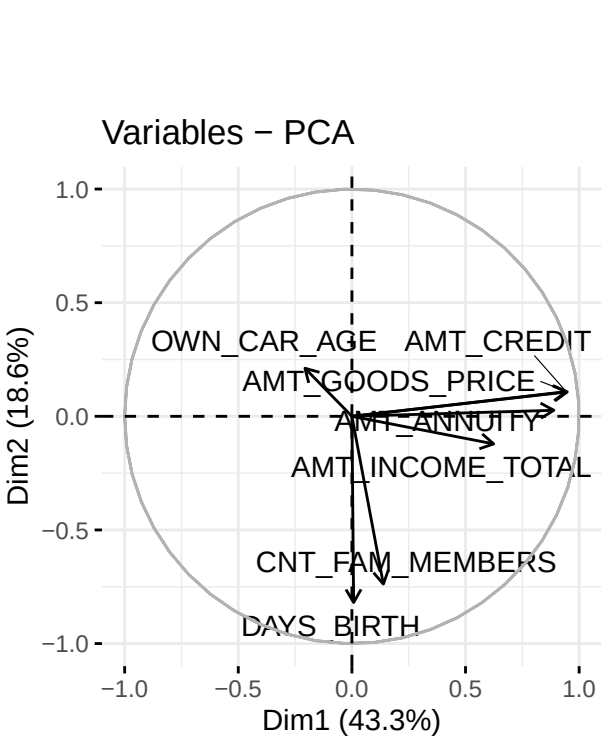
Como se puede apreciar, todos los métodos retornan una estimación similar de la densidad, por lo que se podría decir que es indiferente escoger un método en concreto. De esta forma, se decide usar el MICE como método de imputación final seleccionado.

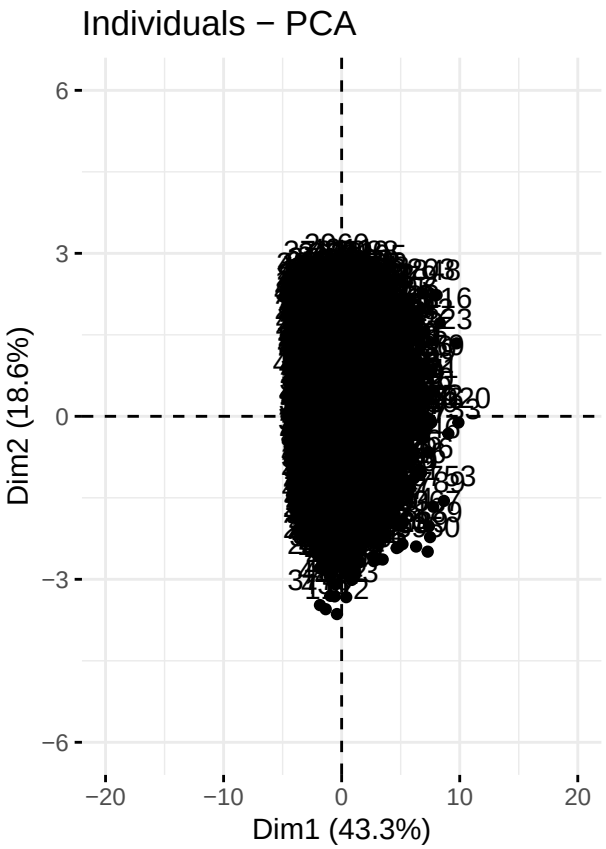
He aquí una tabla resumen sobre los resultados obtenidos acerca de cuál es el mejor criterio de imputación:

	OWN_CAR_AGE	AMT_GOODS_PRICE
Estadísticos	No	Yes
kNN	No	Yes
MICE	Yes	Yes
MiMMi	No	Yes

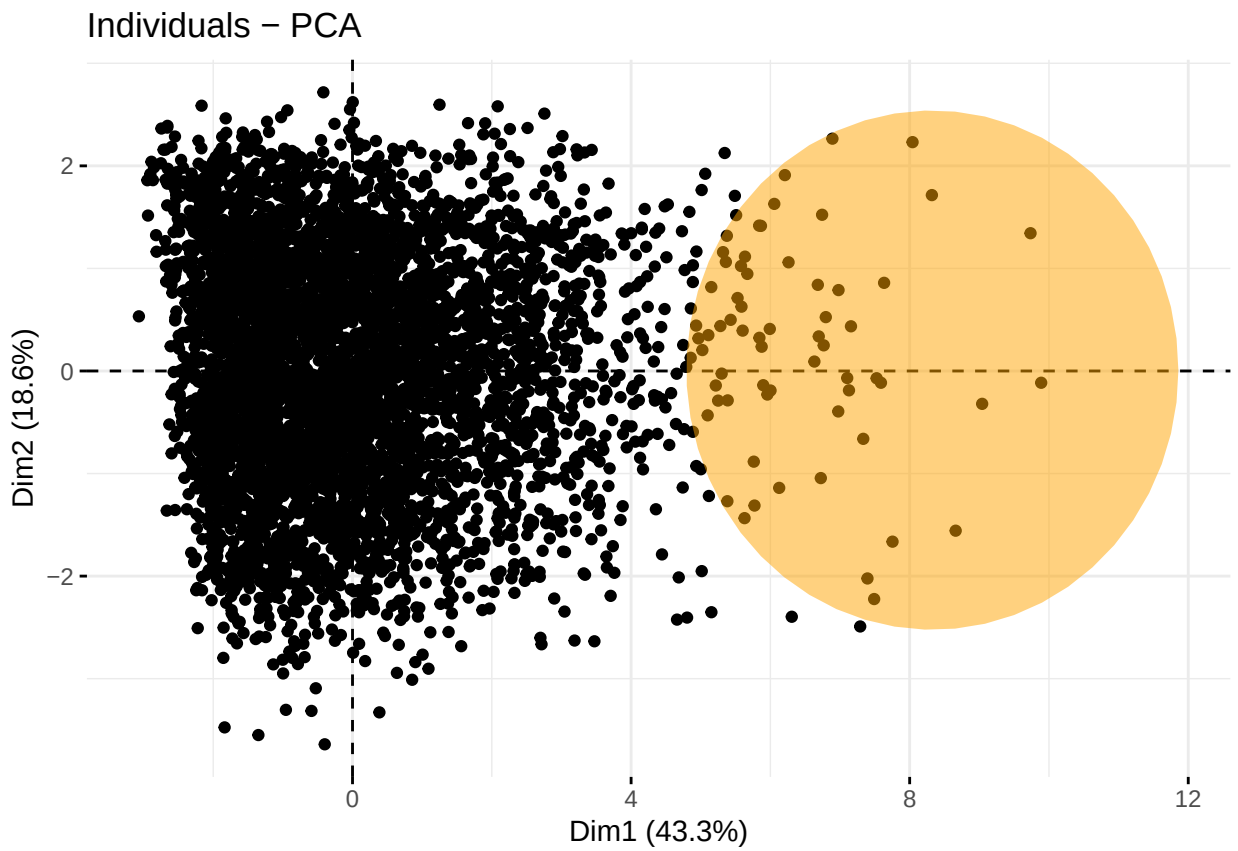
Detección y tratamiento de outliers

En este apartado se tratará de visualizar aquellas observaciones extremas y, además, discernir sobre si deben ser corregidas o no, dependiendo de la naturaleza de la variable. Para ello, se utilizarán métodos multivariantes, como el análisis de componentes principales (PCA). Así, se procede a representar la proyección de los individuos en los primeros planos factoriales para así observar cuáles se alejan del resto de puntos:





Como se puede apreciar, la combinación de las dos primeras dimensiones del PCA acumulan un total del 60% de la inercia total explicada, de forma que es un método de detección bastante fiable en nuestro caso. Identificamos, especialmente, un punto que sobresale del segundo plano factorial, mientras que podemos catalogar una decena de grupos realmente alejados del grupo en la primera dimensión:



Tras haber realizado el PCA correspondiente, vemos claramente un grupo de outliers. Así pues, pasamos a analizar los que son valores extremos por la dimensión 1. Como se puede apreciar, el primer plano factorial viene dado por las variables referidas a cantidad de dinero de nuestra base de datos. Así pues, los outliers presentes son personas con unos ingresos muy altos y que, además, realizaron préstamos por una cantidad de dinero muy superior al que cobran. Así pues, se trata de personas ricas, las cuales existen en nuestra sociedad, de forma que se quedan en la base de datos tal y como aparece. Más adelante, se aplicará alguna transformación que pueda permitir corregir estos valores tan extremos.

Feature engineering

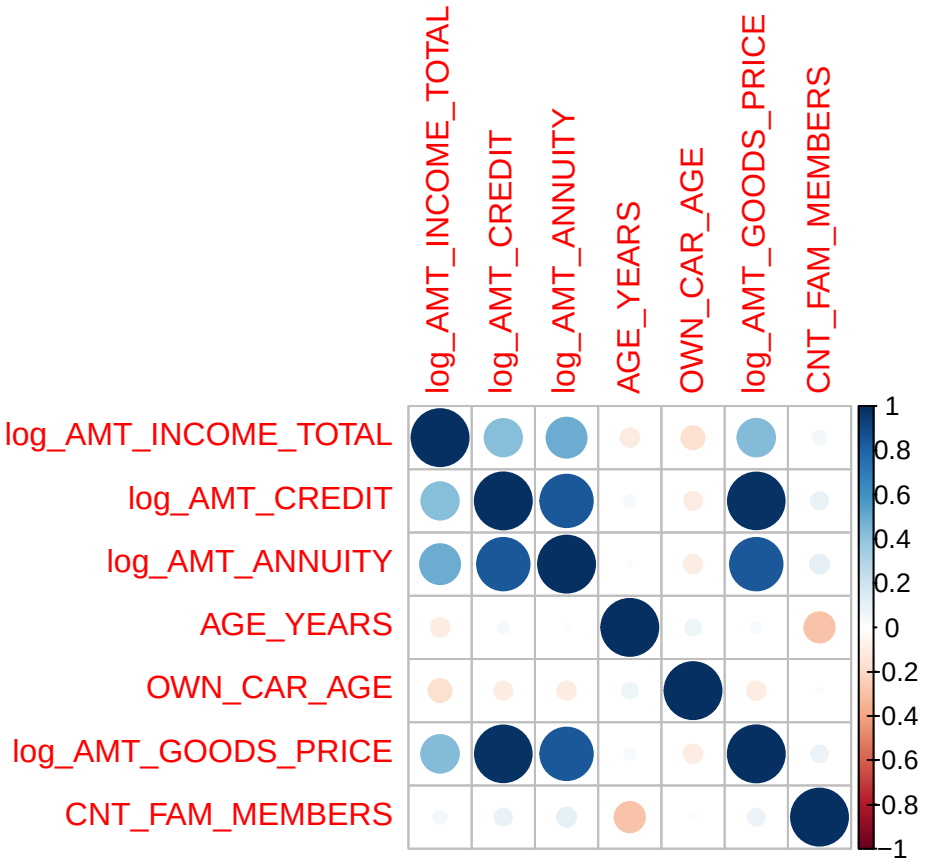
Por último, realizaremos la selección de variables final para nuestra base de datos, así como aplicar transformaciones correctas a nuestras variables para que cumplan algunas hipótesis, como normalidad o heteroscedasticidad. Para este apartado se hace una disección de cada variable una a una.

En primer lugar, se resolverán problemas relacionados con las variables numéricas. Como tenemos variables relacionadas con cantidades monetarias (salario, cantidad prestada...), tal vez sería mejor aplicar una transformación logarítmica:

Así pues, esta transformación debería resolver problemas relacionados con la normalidad de estas variables. Otro cambio a realizar es el respectivo a la variable `DAYS_BIRTH`, la cual muestra el número de días que lleva vivo el individuo. Sin embargo, el hecho de que esta variable esté en negativo y expresada en días (cuando normalmente se hace en años) hace que su interpretación sea complicada. De esta forma, se harán los cambios permanentes para encontrar la edad de los clientes, guardándola en una variable llamada `AGE_YEARS`.

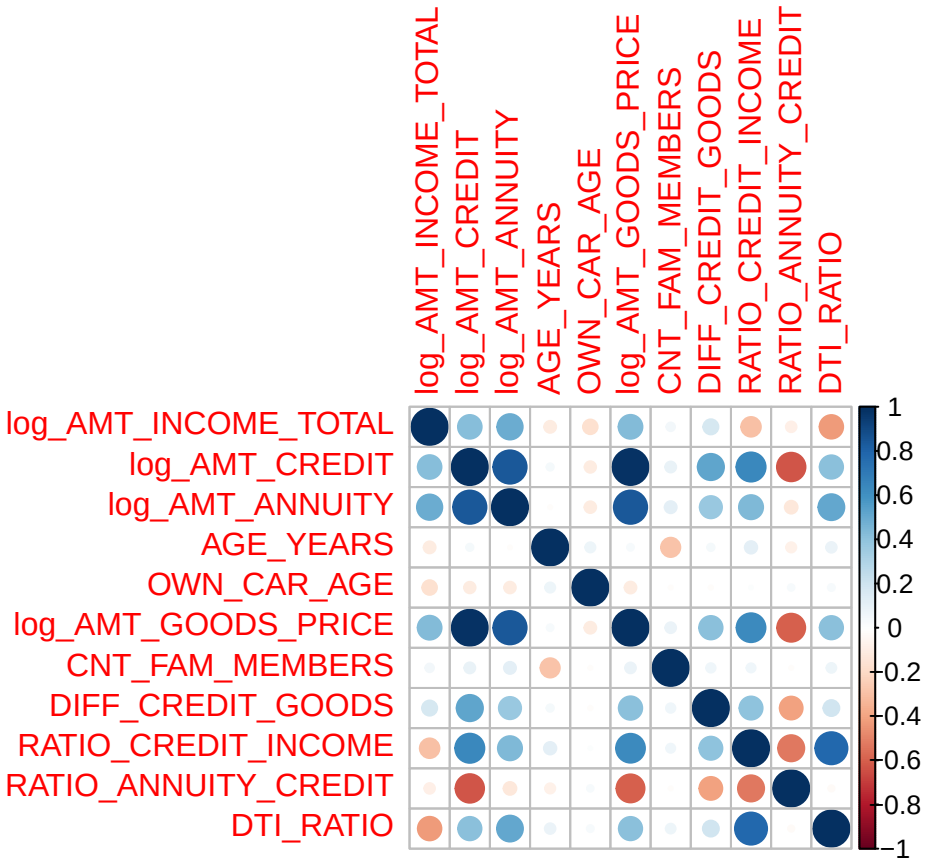
Ahora, vamos a unir aquellas variables ya preprocesadas con el objetivo de tener el dataset preparado para crear nuevas variables.

Antes de avanzar, haremos un correlograma para ver los pares de variables con un mayor coeficiente de correlación de Pearson:



Como se puede apreciar y como era de esperar, hay 3 variables que presentan una gran autocorrelación entre ellas: `log_AMT_CREDIT`, `log_AMT_GOODS_PRICE` y `log_AMT_ANNUITY`. de esta forma, sería ideal nuevas variables a partir de éstas con las cuales se pueda resolver este problema, ya que explican exactamente lo mismo. Para ello, será necesario basarse en la teoría económica y en qué se fijan las entidades de crédito para conceder préstamos. Así, el siguiente objetivo será crear ratios y variables que pretendan controlar y relacionar dinero prestado con capacidad del cliente para retornarlo:

- DIFF_CREDIT_GOODS: Diferencia entre el crédito pedido y el valor del bien para el que se quiere usar
- RATIO_CREDIT_INCOME: Ratio entre el crédito pedido y el salario anual del prestatario. También se puede contar como el número de años que se tarda en devolver el crédito
- RATIO_ANNUITY_CREDIT: Ratio entre la anuidad del préstamo y el crédito total solicitado
- DTI_RATIO: El DTI (Debt-to-income) ratio mide la capacidad del cliente para pagar la annuity de su préstamo en relación con sus ingresos



Se puede apreciar que, ahora, las nuevas variables creadas no presentan tanta correlación entre ellas como anteriormente había. Se puede apreciar, además, que las correlaciones entre las variables donde había problemas siguen teniéndolos y, como se aprecia en el PCA sencillo realizado antes, será necesario descartar alguna variable, ya que explican cosas similares en las mismas dimensiones. Así, en el PCA se deberá realizar el descarte adecuado de variables en función de su aportación al PCA resultante.

Eliminación de variables a través del PCA

Se proceden a eliminar, primeramente, aquellas variables para las cuales ya existe su transformación logarítmica. Esto se hace para no contar con variables que contengan la misma capacidad explicativa (y así evitar colinealidad). También se elimina la variable DAYS_BIRTH, ya que se cuenta con AGE_YEARS, que es una transformación de la inicial, debido a que DAYS_BIRTH no tenia una clara interpretación.

Teniendo en cuenta que la inercia equivale a la proporción de la variabilidad de los datos, se sabe que con un 80% de inercia se puede obtener casi toda la información o variabilidad de la base de datos original. Con ello, vemos que el 80% de la inercia acumulada se logra con 5 planos factoriales, pero aún se pueden eliminar algunas variables.

Observamos la tabla de rotaciones:

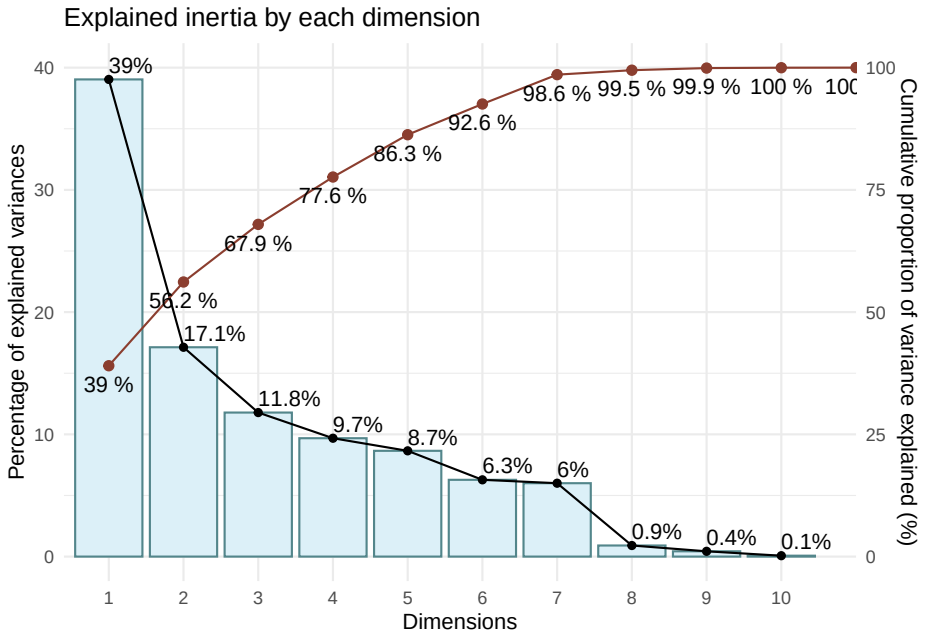


Figure 1: Porcentaje de inercia explicado por dimensión

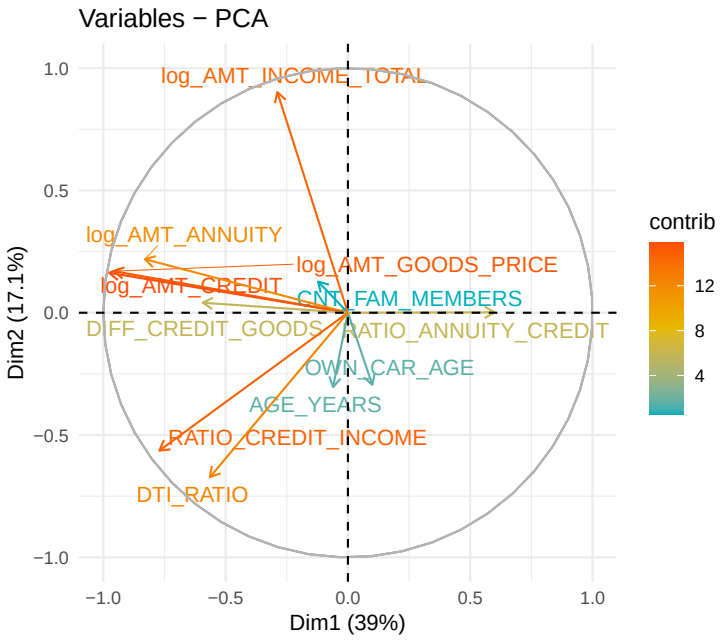


Figure 2: Proyección de variables en los dos primeros planos factoriales

Table 8: Correlación de cada variable con cada plano factorial

	PC1	PC2	PC3	PC4	PC5
OWN_CAR_AGE	0.0482030	-0.2147590	0.0769556	0.2124352	-0.9297935
CNT_FAM_MEMBERS	-0.0588179	0.0929658	-0.6331317	0.3395058	-0.1036261
log_AMT_INCOME_TOTAL	-0.1392241	0.6571139	0.0972753	-0.1405196	-0.1763198
log_AMT_CREDIT	-0.4712020	0.1189920	0.0510834	-0.0062252	-0.0347199
log_AMT_ANNUITY	-0.4005658	0.1596318	-0.1510783	-0.4216955	-0.1777113
log_AMT_GOODS_PRICE	-0.4617666	0.1237915	0.0364116	-0.0533605	-0.0298206
AGE_YEARS	-0.0293842	-0.2216182	0.6059015	-0.2585720	-0.0779069
DIFF_CREDIT_GOODS	-0.2862735	0.0298434	0.1153661	0.3103286	-0.0469760
RATIO_CREDIT_INCOME	-0.3720815	-0.4105490	-0.0503004	0.0927487	0.1203650
RATIO_ANNUITY_CREDIT	0.2893721	0.0013840	-0.3120685	-0.6076862	-0.1927285
DTI_RATIO	-0.2722575	-0.4898101	-0.2768696	-0.3132092	0.0085247

En el grafico vemos que las flechas de **log_AMT_GOODS_PRICE** y **log_AMT_CREDIT** se solapan entre ellas, eso queire decir que las dos variables explican el mismo plano factorial. Vemos en la tabla de rotaciones que **log_AMT_CREDIT** contribuye más a explicar el primer plano factorial, y además las correlaciones entra cada una de las variables y cada dimensión son muy similares. Por esta razón eliminamos **log_AMT_GOODS_PRICE**.

Nos quedamos con una variable menos, por tanto tenemos 10 variables numéricas.

De vuelta, verificamos el porcentaje de inercia por cada componente principal y la acumulada:

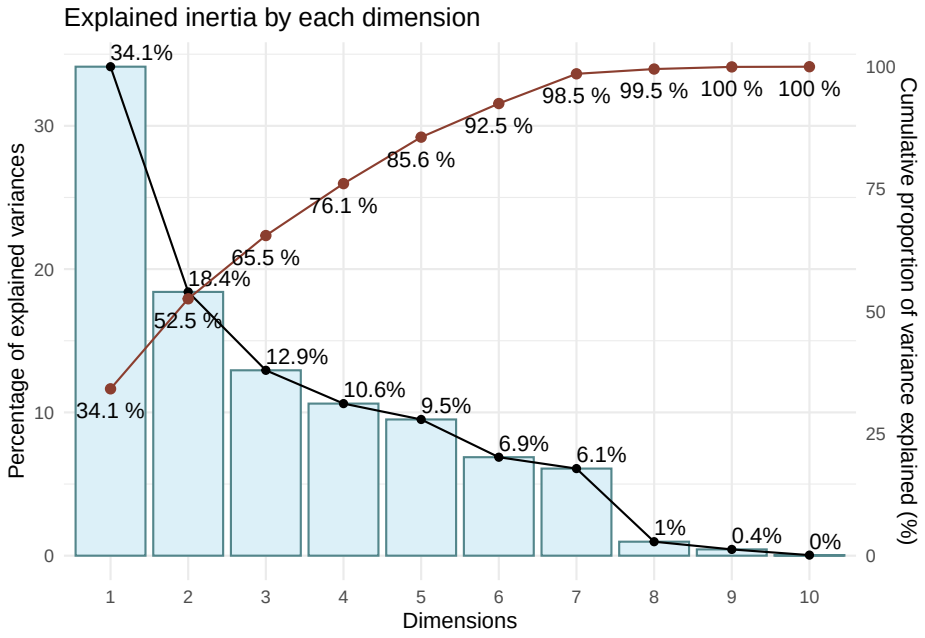


Figure 3: Porcentaje de inercia explicado por dimensión

Como se puede ver, seguimos teniendo 5 dimensiones que acumulan el 80% de la varianza.

Vemos que las variables **CNT_FAM_MEMBERS**, **AGE_YEARS** y **OWN CAR AGE** no explican las dos primeras componentes pero si nos fijamos en la tabla de rotaciones vemos que sí tienen importancia a la hora de explicar las otras tres dimensiones:

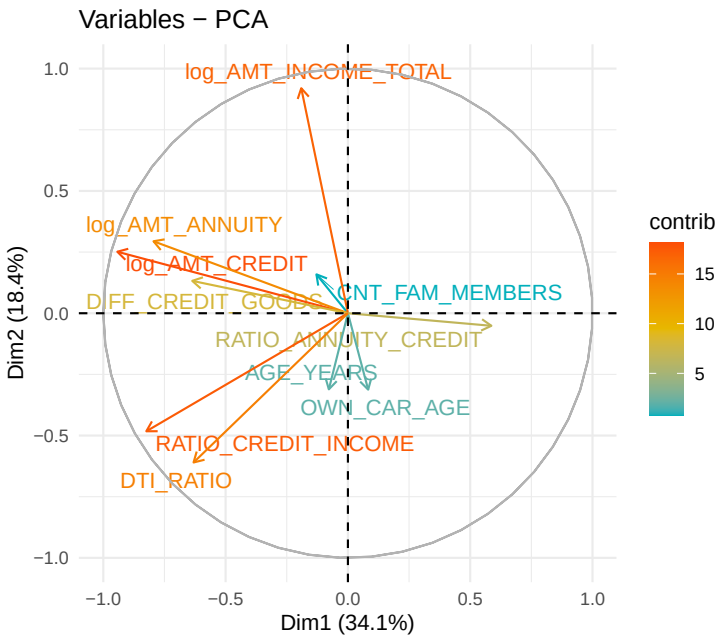


Figure 4: Proyección de variables en los dos primeros planos facoriales

Table 9: Correlación de cada variable con cada plano factorial

	PC1	PC2	PC3	PC4	PC5
OWN_CAR_AGE	0.0448564	-0.2302757	0.0682728	0.2374868	-0.9204698
CNT_FAM_MEMBERS	-0.0696389	0.1170965	-0.6253548	0.3446156	-0.0975813
log_AMT_INCOME_TOTAL	-0.1032219	0.6780034	0.1127291	-0.1578039	-0.1887548
log_AMT_CREDIT	-0.5102972	0.1852910	0.0669570	-0.0244020	-0.0436793
log_AMT_ANNUITY	-0.4302495	0.2168397	-0.1402859	-0.4375011	-0.1955742
AGE_YEARS	-0.0422045	-0.2293068	0.6006725	-0.2659287	-0.0869647
DIFF_CREDIT_GOODS	-0.3445047	0.0975371	0.1404796	0.2727512	-0.0669711
RATIO_CREDIT_INCOME	-0.4460967	-0.3556622	-0.0452077	0.0854911	0.1192847
RATIO_ANNUITY_CREDIT	0.3175459	-0.0378105	-0.3257004	-0.5977281	-0.2028508
DTI_RATIO	-0.3415601	-0.4496669	-0.2805932	-0.3134270	0.0015825

Por ejemplo, en el caso de **OWN CAR AGE** se puede ver en la tabla anterior que, se podría decir que no es la que mejor explica las primeras componentes, pero vemos que explica casi toda la componente 5.

Otra observación se podria hacer de las variables **log_AMT_CREDIT** y **log_AMT_ANNUITY**, donde se puede apreciar que tienen correlaciones similares con la primera y segunda dimensión. Teniendo en cuenta que esas dos primeras dimensiones (PC1 y PC2) són las más importantes, ya que acumulan la mayoría de la inercia (en total un 52.2%), parece una decisión sensata eliminar una de ellas, en este caso **log_AMT_ANNUITY**.

Ahora conservamos 9 variables numéricas.

De forma igual que anteriormente, comprobamos el porcentaje de inercia para cada componente principal y la acumulada:

Como se puede comprobar, las 5 dimensiones siguen siendo las necesarias para acumular el 80% de la varianza.

Observamos tambien la tabla de rotaciones para verificar si se puede eliminar alguna variable más:

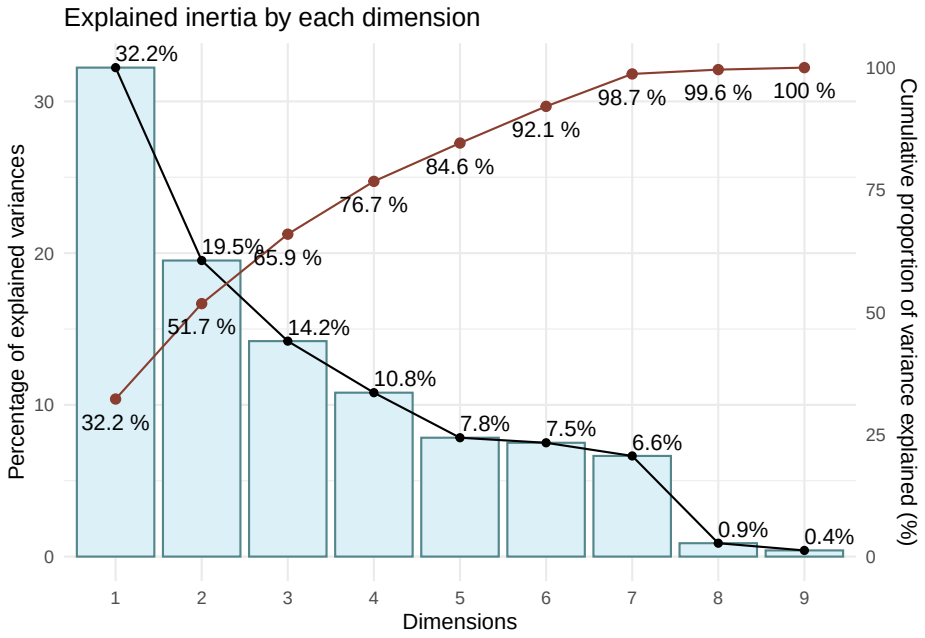


Figure 5: Porcentaje de inercia explicado por dimensión

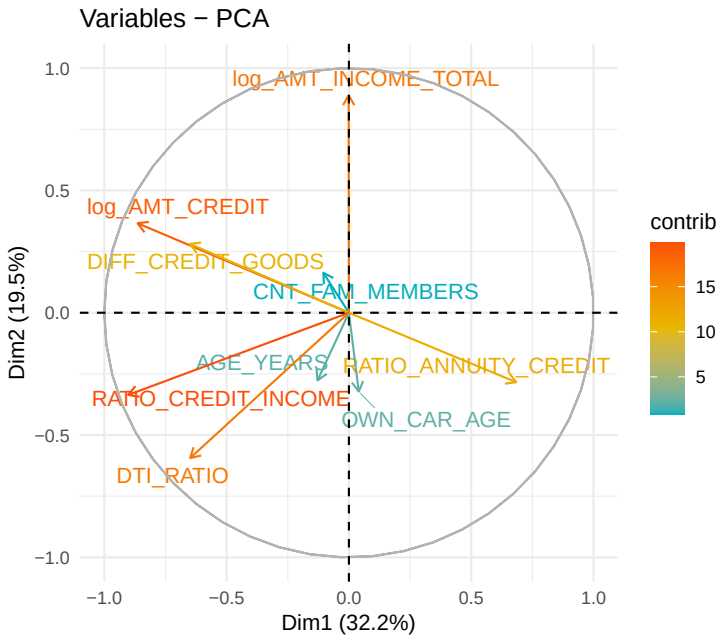


Figure 6: Proyección de variables en los dos primeros planos factoriales

Table 10: Correlación de cada variable con cada plano factorial

	PC1	PC2	PC3	PC4	PC5
OWN_CAR_AGE	0.0236578	-0.2431871	0.0856094	0.9304912	-0.0313202
CNT_FAM_MEMBERS	-0.0613547	0.1231404	-0.6851660	0.2130058	-0.3479517
log_AMT_INCOME_TOTAL	-0.0005400	0.6707252	0.1004662	0.0480525	-0.3636840
log_AMT_CREDIT	-0.5071695	0.2760046	0.0327814	-0.0071521	-0.1447726
AGE_YEARS	-0.0753880	-0.2086722	0.6522356	-0.0256320	-0.5333714
DIFF_CREDIT_GOODS	-0.3816334	0.2124758	0.0616461	0.1733418	-0.1028635
RATIO_CREDIT_INCOME	-0.5297879	-0.2534602	-0.0712543	-0.0719465	0.0797531
RATIO_ANNUITY_CREDIT	0.4015872	-0.2142091	-0.1817931	-0.1360986	-0.5891711
DTI_RATIO	-0.3810247	-0.4481384	-0.2114547	-0.1790276	-0.2759971

Si nos fijamos en el gráfico que incluye los dos primeros planos factoriales (PC1 y PC2), resulta fácil ver que **log_AMT_CREDIT** y **DIFF_CREDIT_GOODS** se solapan en su proyección, teniendo **log_AMT_CREDIT** más contribución dado que el vector es más largo. De aquí se entiende que las correlaciones de ambas variables en los dos primeros planos factoriales son muy similares, motivo por el cual solapan. En la tabla de correlaciones anterior se puede comprobar como efectivamente, estas correlaciones son similares. Incluso la correlación en ambas variables con la tercera dimensión (PC3) es baja, de forma parecida. Por tanto, se procede a eliminar aquella con menos contribución en PC1 y PC2, esta siendo **DIFF_CREDIT_GOODS**.

Ahora se conservan 8 variables numéricas.

Se vuelven a ejecutar todos los pasos anteriores para volver a verificar si hace falta eliminar más variables:

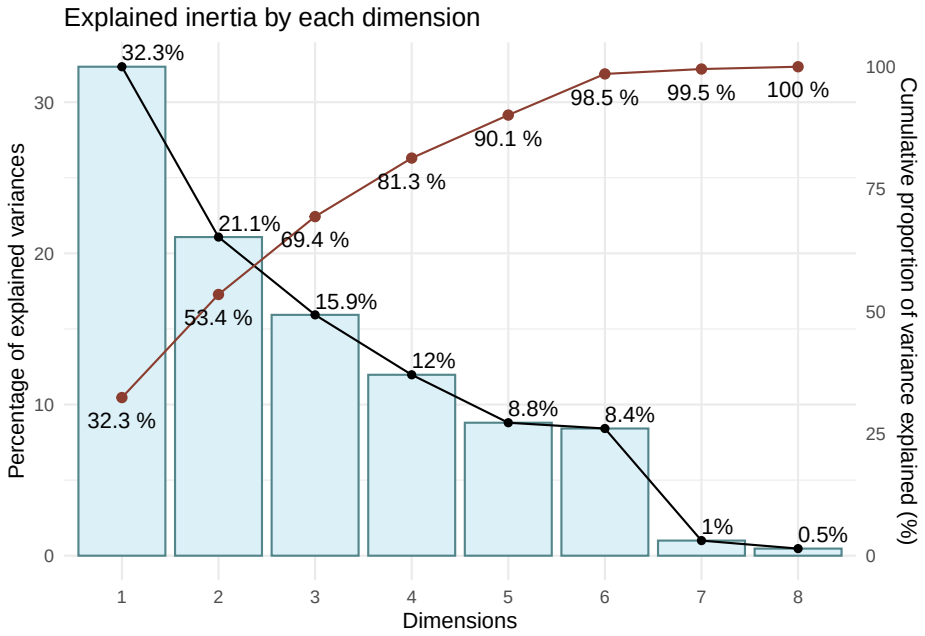


Figure 7: Porcentaje de inercia explicado por dimensión

Se aprecia como la eliminación de **DIFF_CREDIT_GOODS** ha modificado el número de dimensiones necesarias para alcanzar el 80% de inercia acumulada, pasando de 5 a 4 dimensiones.

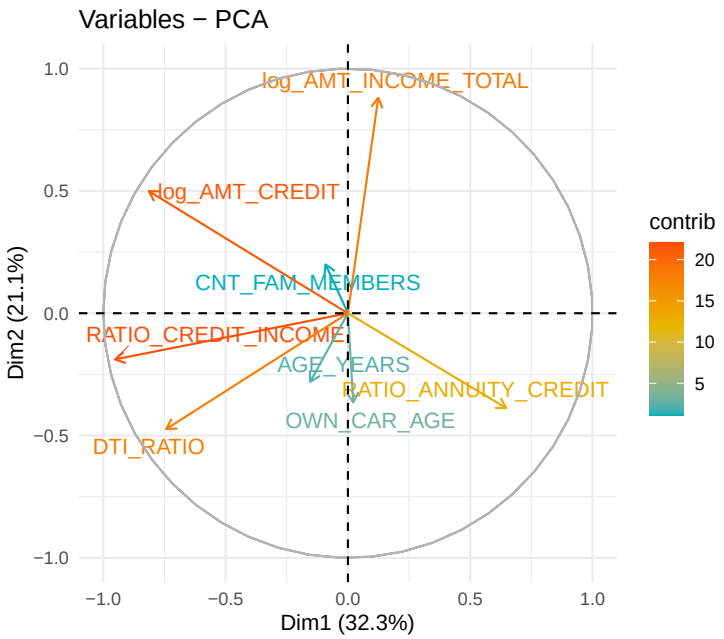


Figure 8: Proyección de variables en los dos primeros planos factoriales

Table 11: Correlación de cada variable con cada plano factorial

	PC1	PC2	PC3	PC4
OWN_CAR_AGE	0.0136015	-0.2805660	0.0566802	0.9279813
CNT_FAM_MEMBERS	-0.0569020	0.1537128	-0.6815960	0.2127134
log_AMT_INCOME_TOTAL	0.0770885	0.6782457	0.1283026	0.0953268
log_AMT_CREDIT	-0.5061279	0.3844741	0.0672871	0.0618092
AGE_YEARS	-0.0958336	-0.2158479	0.6478801	-0.0145592
RATIO_CREDIT_INCOME	-0.5916707	-0.1453620	-0.0575363	-0.0408077
RATIO_ANNUITY_CREDIT	0.4024628	-0.2986119	-0.2097241	-0.2139054
DTI_RATIO	-0.4617731	-0.3645929	-0.2101949	-0.1818331

Comprobando el gráfico de las dos primeras dimensiones, y analizando las correlaciones, parece ser que ya no hace falta eliminar más variables. Por tanto, conservamos 8 variables numéricas.

Las variables eliminadas han sido: - **AMT_INCOME_TOTAL**, **AMT_CREDIT**, **AMT_ANNUITY**, **AMT_GOODS_PRICE**, todas ellas con motivo de que ya se habia creado otra variable a partir de su transformación logarítmica. - **DAYS_BIRTH**, ya que la variable **AGE_YEARS** es una transformación de ella. - **log_AMT_GOODS_PRICE** - **log_AMT_ANNUITY** - **DIFF_CREDIT_GOODS**

Balanceo de los datos

Por último, una vez ya se ha limpiado la base de datos y ésta es óptima para realizar un análisis de datos, es necesario balancearla. Para ello, entre todas las alternativas posibles que se han buscado, la opción que proporciona una base de datos balanceada más parecida a la base de datos original en cuanto a la estructura se refiere es la proporcionada por la función `ovun.sample()`, del paquete `ROSE`, usando la opción `method = "both"`. Esta opción hace una mezcla entre oversampling y undersampling, manteniendo la estructura

original y retornando una nueva base de datos con observaciones artificiales ya balanceada. Así pues, se presenta un `summary()` de la nueva base de datos balanceada sobre la que se trabajará:

```
library(ROSE)
```

```
## Warning: package 'ROSE' was built under R version 4.2.3
```

```
## Loaded ROSE 0.0-4
```

```
data_factor = select_if(data, is.factor)
data_desbalanceada = data.frame(data_factor, data_numeric)

x = names(data_desbalanceada)[!(names(data_desbalanceada) %in% c("TARGET"))]
y = "TARGET"


data_balanceada = ovun.sample(reformulate(x,y), data = data_desbalanceada, method = "both", p = 0.5, se

summary(data_balanceada)
```

```
##  CODE_GENDER             NAME_INCOME_TYPE
##  F:3004      Commercial associate:1144
##  M:1996      Pensioner           : 728
##              State servant       : 335
##              Student             :   1
##              Unemployed          :   0
##              Working             :2792
##
##                     NAME_EDUCATION_TYPE            NAME_FAMILY_STATUS
##  Academic degree              :   1     Civil marriage      : 481
##  Higher education             :1053     Married             :3214
##  Incomplete higher            : 192     Separated           : 336
##  Lower secondary              :  53     Single / not married: 782
##  Secondary / secondary special:3701     Widow               : 187
##
##
##                 OCCUPATION_TYPE                ORGANIZATION_TYPE
##  High skill laborers    : 632   Business and bank        :1559
##  Low-mid skill laborers :1126   Unknown                  : 728
##  Low skill laborers     : 827   Self-employed            : 686
##  Mid-high skill laborers: 414   Other                    : 424
##  Mid skill laborers     : 715   Industry and construction: 336
##  Unknown                :1286   Public services          : 287
##                                 (Other)                  : 980
##  REGION_RATING_CLIENT TARGET    OWN_CAR_AGE    CNT_FAM_MEMBERS
##  1: 441               0:2521   Min.   : 0.00   Min.   :1.000
##  2:3655               1:2479   1st Qu.: 5.00   1st Qu.:2.000
##  3: 904                        Median :10.00   Median :2.000
##                                Mean   :10.99   Mean   :2.199
##                                3rd Qu.:15.00   3rd Qu.:3.000
##                                Max.   :45.00   Max.   :6.000
##
```

```
##  log_AMT_INCOME_TOTAL log_AMT_CREDIT    AGE_YEARS     RATIO_CREDIT_INCOME
##  Min.   :10.27        Min.   :10.81   Min.   :21.00   Min.   : 0.2308
##  1st Qu.:11.63        1st Qu.:12.55   1st Qu.:32.00   1st Qu.: 2.0800
##  Median :11.91        Median :13.14   Median :41.00   Median : 3.2920
##  Mean   :11.92        Mean   :13.07   Mean   :42.16   Mean   : 3.9068
##  3rd Qu.:12.22        3rd Qu.:13.57   3rd Qu.:52.00   3rd Qu.: 5.0000
##  Max.   :13.60        Max.   :14.74   Max.   :68.00   Max.   :20.9091
##
##  RATIO_ANNUITY_CREDIT   DTI_RATIO
##  Min.   :0.02638      Min.   :0.01605
##  1st Qu.:0.03770      1st Qu.:0.11440
##  Median :0.05000      Median :0.16606
##  Mean   :0.05330      Mean   :0.17932
##  3rd Qu.:0.06149      3rd Qu.:0.22572
##  Max.   :0.12443      Max.   :0.77475
##
```