



Instituto Tecnológico y de Estudios Superiores de Monterrey

Maestría en Inteligencia Artificial Aplicada

Proyecto Integrador

Aplicación de modelos de aprendizaje supervisado para el cálculo de probabilidad de pérdida de empleo en clientes de nómina

Avance 1. Análisis Exploratorio de Datos

Grupo 10:

Abraham Cabanzo Jiménez - A01795355

Kevin Alejandro Ayón Payán - A01740679

Pedro Ulises Meléndez Ortega - A00716301

Doctora: Grettel Barceló Alonso

26 de enero de 2025

Exploratory Data Analysis (EDA)

Muchos autores definen el análisis exploratorio de datos (EDA) como el punto de partida para cualquier elaboración de algún modelo o proyecto en el que se esté implementando un conjunto de datos. Su propósito principal, sobre todo, es identificar aquellas relaciones, patrones, errores, cualidades o características de la base de datos para poder detectar, corregir o sobreponer anomalías, valores atípicos o, en medida de lo posible, controlar y estructurar los datos de la manera más eficiente posible para el trabajo posterior de estos.

El análisis exploratorio de datos puede efectuarse de muchas maneras posibles, la más común es en el manejo y manipulación de los datos por medio de códigos de limpieza y estructuración para evitar ciertos errores o problemas en la estructura de los datos en procesos más avanzados del proyecto.

Algunas de las características a identificar al implementar el análisis exploratorio de datos son las siguientes:

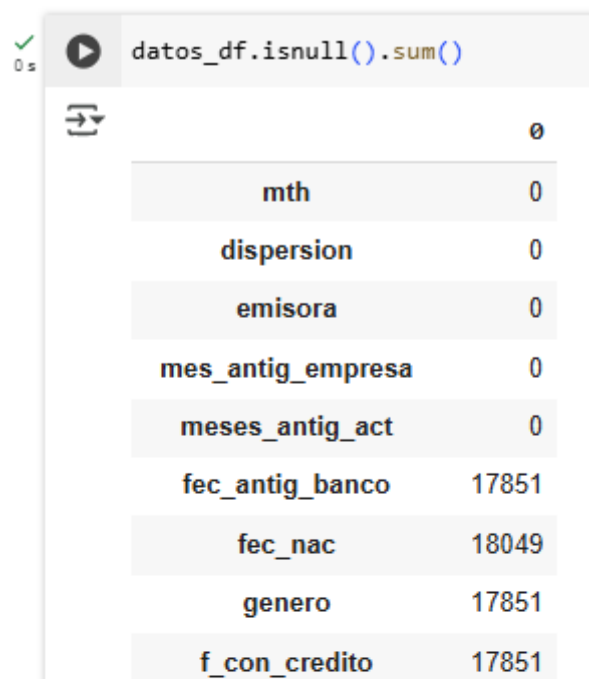
- Descripción general de los datos: siendo este el punto de partida para el análisis, se debe de identificar y detectar la descripción de las variables que componen el conjunto de datos. ¿Son variables categóricas? ¿y las variables booleanas? ¿Cuál es el tamaño de la base? ¿Cuáles son todos los tipos de datos que integran la base?, entre estas y muchas preguntas descriptivas se contestan en este punto.
- Valores atípicos y valores faltantes: Se debe identificar la presencia (o no) de valores atípicos dentro de la base para su posterior control, manejo y rellenado de estos para evitar posibles sesgos en los análisis. Además, se requiere rellenar o eliminar todo el registro (dependiendo del usuario y el tipo de dato) los datos faltantes que existen dentro de la base.
- Estadísticas descriptivas: Se realizan las estadísticas descriptivas del conjunto de datos y de los campos que integran la base. Valores como media, mediana, moda, desviación estándar, etc son algunos ejemplos.
- Tendencia y distribución: Se debe observar e identificar la tendencia y distribución de los datos. De esta manera, será más eficiente el uso de distintos modelos estadísticos para poder ejecutar de manera óptima los análisis y obtener resultados y conclusiones más exactas.

Preguntas a contestar:

¿Hay valores faltantes en el conjunto de datos? ¿Se pueden identificar patrones de ausencia?

Como en todo gran conjunto de datos, es muy difícil no encontrar variables vacías o nulas que pueden interferir con el cálculo o elaboración de un análisis. Para este caso, se encontraron diversos datos faltantes en las bases de datos, los cuales pudimos identificar gracias a códigos de identificación de datos nulos en python.

Se pudieron identificar las siguientes ausencias según el campo que se realiza el conteo de datos faltantes:



The screenshot shows a Jupyter Notebook interface. At the top, there is a code cell with the text `datos_df.isnull().sum()`. Below the code cell, there is an output cell displaying a table with two columns: the variable names and their corresponding counts of missing values. The table is as follows:

	0
mth	0
dispersion	0
emisora	0
mes_antig_empresa	0
meses_antig_act	0
fec_antig_banco	17851
fec_nac	18049
genero	17851
f_con_credito	17851

En el código subido en github se observan todos los campos

¿Cuáles son las estadísticas resumidas del conjunto de datos?

Primeramente, antes de revisar las estadísticas de la base de datos, se realiza una exploración inicial de los datos para identificar si son valores numéricos o categóricos:

```
datos_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 504549 entries, 0 to 504548
Data columns (total 27 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   mth                                    504549 non-null object
1   dispersion                            504549 non-null float64
2   emisora                              504549 non-null int64
3   mes_antig_empresa                    504549 non-null object
4   meses_antig_act                      504549 non-null int64
5   fec_antig_banco                      486698 non-null object
6   fec_nac                              486500 non-null object
7   genero                               486698 non-null object
8   f_con_credito                        486698 non-null float64
9   saldo_promedio                       486698 non-null float64
10  creditos                             486698 non-null float64
11  mes_ult_disp                          504549 non-null object
12  meses_para_perdida                   504549 non-null int64
13  empleados                            504423 non-null float64
14  dispersion_empleados                 504423 non-null float64
15  dispersion_prom                      504423 non-null float64
16  porc_penetracion                     454257 non-null float64
17  saldo_total                          454257 non-null float64
18  vencido                              69315 non-null float64
19  castigos                             18012 non-null float64
20  rotacion_emisora                     488338 non-null float64
21  segmento                             504377 non-null object
22  tipo_gob                             171856 non-null object
23  target                              504549 non-null int64
24  edad_ingreso                         486500 non-null float64
25  edad_perdida_emp                     486500 non-null float64
26  uuid                                 504549 non-null object
```

Una vez identificados los tipos de datos, se pueden realizar las estadísticas descriptivas iniciales para cada uno de ellos, como las siguientes:

Numéricos:

[7] datos_df.describe()

	dispersion	emisora	meses_antig_act	f_con_credito	saldo_promedio	creditos	meses_para_perdida	empleados	dispersion_empleados	dispersion_prom	porc_penetracion	saldo_total	vencido	castigos	rotacion_emisora
count	5.045490e+05	5.045490e+05	504549.000000	486698.000000	4.866980e+05	486698.000000	504549.000000	504423.000000	5.044230e+05	5.044230e+05	454257.000000	4.542570e+05	69315.000000	1.801200e+04	488338.000000
mean	1.481075e+04	1.907819e+06	35.632995	0.170656	1.037922e+04	0.204182	4.097963	2510.878556	3.161451e+07	1.312134e+04	0.395954	2.329331e+07	74783.416151	1.521678e+04	0.190637
std	2.988375e+04	1.815919e+06	33.965709	0.376209	9.187785e+04	0.498292	1.157223	4380.164621	4.920886e+07	1.220007e+04	0.250817	3.153545e+07	29141.675310	7.053149e+04	0.151305
min	1.000000e-02	1.000000e+00	0.000000	0.000000	0.000000e+00	0.000000	0.000000	1.000000	4.640000e+02	2.000000e+02	0.000745	2.500000e+02	797.110000	-5.180000e+00	0.000000
25%	7.467840e+03	2.643420e+05	8.000000	0.000000	2.317400e+02	0.000000	4.000000	114.000000	1.173872e+06	8.625624e+03	0.194915	2.378103e+05	67629.480000	3.568200e+02	0.072846
50%	1.066304e+04	1.454291e+06	24.000000	0.000000	8.668000e+02	0.000000	4.000000	530.000000	6.880417e+06	1.157581e+04	0.376439	3.858012e+06	84905.070000	1.376450e+03	0.159402
75%	1.692122e+04	2.938144e+06	56.000000	0.000000	2.979113e+03	0.000000	5.000000	3302.000000	4.370084e+07	1.475329e+04	0.594675	3.304573e+07	90981.880000	1.718624e+04	0.259009
max	1.337450e+07	6.024120e+06	103.000000	1.000000	1.515353e+07	10.000000	5.000000	18402.000000	2.130181e+08	1.487693e+06	3.000000	9.149855e+07	246982.770000	1.623741e+06	1.000000

Categoricos:

[8] datos_df.describe(include="object")

	mth	mes_antig_empresa	fec_antig_banco	fec_nac	genero	mes_ult_disp	segmento	tipo_gob	uuid
count	504549	504549	486698	486500	486698	504549	504377	171856	504549
unique	2	104	104	22583	3	6	8	4	279198
top	01JUL2024	01FEB2016	01JAN2016	30AUG1994	M	01DEC2024	PYME	MUNICIPAL	ffe95d04-b8c9-9644-aff0-1b936a500a08
freq	266309	59901	71598	72	280612	429972	199632	98245	2

Así podemos ir interpretando las estadísticas que estaremos esperando para el trabajo de la base de datos, así como posibles valores atípicos y promedios.

¿Hay valores atípicos en el conjunto de datos?

Sí existen valores atípicos en el conjunto de datos, principalmente observaremos como ejemplo el campo de edad, el cual se identificaron ciertos valores atípicos y posteriormente se realizó una estrategia de eliminación de estos registros para no ensuciar el análisis y evitar errores en el proyecto:

Identificación:

```
datos_df["edad_ingreso"].describe()
```

	edad_ingreso
count	486500.000000
mean	37.966865
std	12.753063
min	-7.000000
25%	27.000000
50%	37.000000
75%	47.000000
max	220.000000

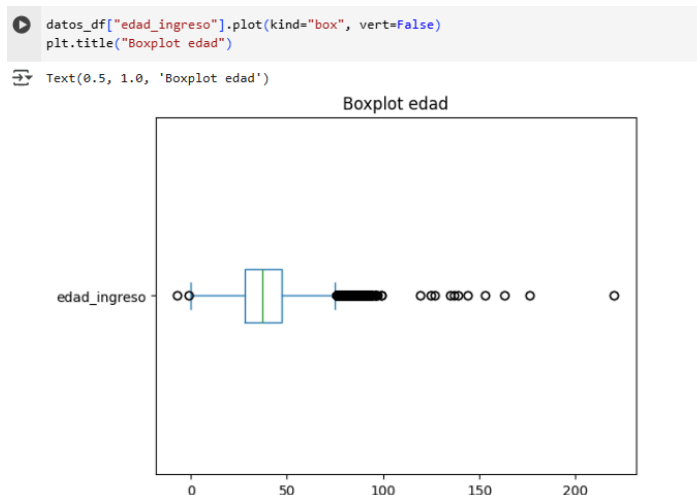
Aquí se muestra como claramente la edad tiene presencia de datos atípicos y que en un análisis en la vida real son datos que no tienen coherencia. Este tipo de registros será eliminado:

```
[13] faltedad= datos_df["edad_ingreso"].isnull().sum()  
faltedad
```

```
18049
```

```
faltedad= datos_df["edad_ingreso"].isnull().sum()  
  
poredad = (faltedad/len(datos_df))*100  
  
print("El porcentaje de los valores faltantes de edad es ", poredad)
```

```
El porcentaje de los valores faltantes de edad es 3.577254141817742
```



El código para dropear estos valores atípicos será de la siguiente manera:

```
#Creando el dataframe
outliers.df = pd.DataFrame(outliers)
outliers.df
```

	edad_ingreso
908	82.0
909	82.0
1032	78.0
1033	78.0
1081	76.0
...	...
495357	76.0
498102	82.0
500925	77.0
501940	79.0
502729	77.0

2270 rows × 1 columns

```
datos_df = datos_df.merge(outliers.df, how='left', indicator=True)
datos_df = datos_df[datos_df['_merge'] == 'left_only']

datos_df.drop(columns=['_merge'], inplace=True)
datos_df
```

#Pasaron de ser 1.667.540 filas a 1.654.630 filas debido a la eliminación de los

Los códigos se observan de mejor manera en código python en GitHub

¿Cuál es la cardinalidad de las variables categóricas?

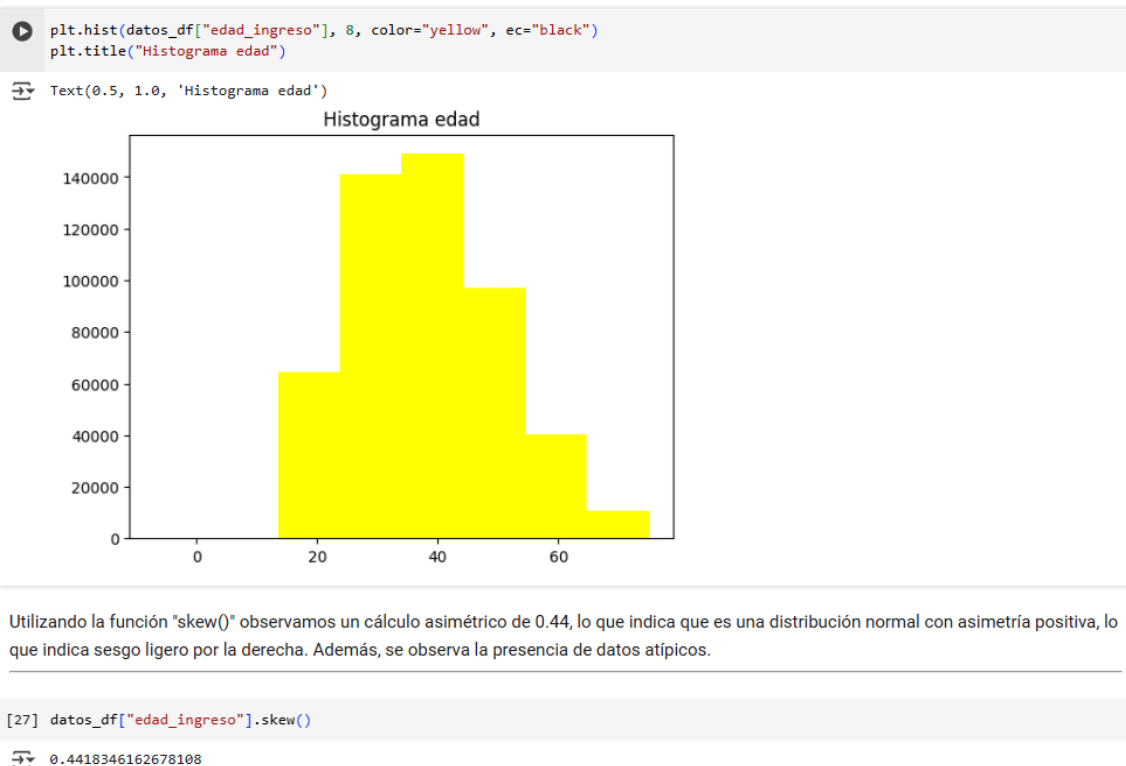
```
[8] datos_df.describe(include="object")
```

	mth	mes_antig_empresa	fec_antig_banco	fec_nac	genero	mes_ult_disp	segmento	tipo_gob	uuid
count	504549	504549	486698	486500	486698	504549	504377	171856	504549
unique	2	104	104	22583	3	6	8	4	279198
top	01JUL2024	01FEB2016	01JAN2016	30AUG1994	M	01DEC2024	PYME	MUNICIPAL	ffe95d04-b8c9-9644-aff0-1b936a500a08
freq	266309	59901	71598	72	280612	429972	199632	98245	2

La cardinalidad en las variables categóricas se refiere al registro único en cada uno de los campos. En la imagen mostrada con anterioridad, se muestra el campo “Unique” que representa el número único de datos en cada una de las variables categóricas.

¿Existen distribuciones sesgadas en el conjunto de datos? ¿Necesitamos aplicar alguna transformación no lineal?

Siguiendo con el ejemplo de edad, se muestra una distribución ligeramente sesgada hacia la derecha, esto se logró construyendo un histograma y calculando el sesgo con la función skew() en python para validar la presencia o no de este. Esto se logró como ejemplo con la siguiente imagen:

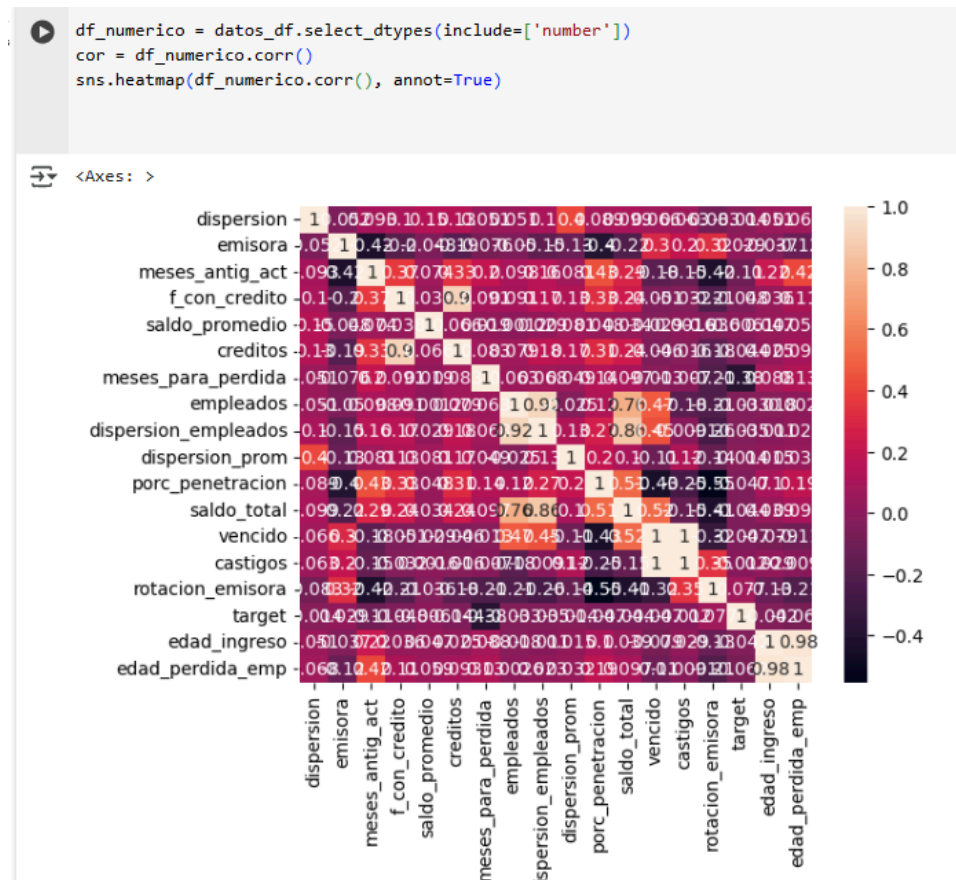


¿Se identifican tendencias temporales? (En caso de que el conjunto incluya una dimensión de tiempo).

Por el momento, cómo está construida la base de datos y la manera en que se construye el modelo para el proyecto, no se indagará aún en las tendencias temporales. Los campos de fechas serán manejados con posterioridad en el proyecto.

¿Hay correlación entre las variables dependientes e independientes?

Para valores prácticos, se realizó la correlación de ejemplo solo a variables numéricas, la cual dio como resultado la siguiente matriz:



La cual indica las variables que tienen correlación (y no) en el conjunto de datos.

¿Cómo se distribuyen los datos en función de diferentes categorías? (análisis bivariado)

Para este caso, se observa el siguiente ejemplo para observar la distribución de los datos entre diferentes categorías. Se tomó como ejemplo las variables “GENERO” y “SEGMENTO” para fines prácticos:

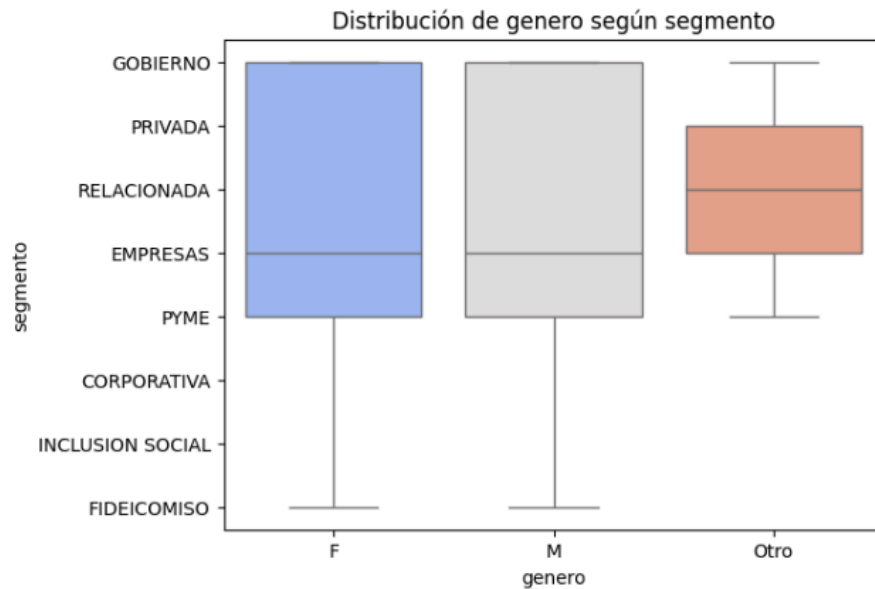

```
sns.boxplot(x='genero', y='segmento', data=datos_df, palette='coolwarm')
plt.title("Distribución de genero según segmento")
plt.show()
```



<ipython-input-35-f878cc5db638>:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign

```
sns.boxplot(x='genero', y='segmento', data=datos_df, palette='coolwarm')
```



¿Hay desequilibrio en las clases de la variable objetivo?

Si hay un desbalanceo entre clases debido a que los registros de pérdida de empleo en el periodo de observación son apenas el 4.4% de la muestra, esto lo identificamos con la variable calculada target con valor 1.

Conclusiones generales del análisis exploratorio de datos.

El data set armado cuenta con 26 columnas, de las cuales 18 son numéricas, 4 categóricas, el ID anonimizado del cliente y 3 variables de tiempo (fecha).

Es necesario transformar los datos de fecha.

De entrada observamos que debemos revisar el porcentaje de rotación, ya que al parecer existen registros que marcan 100% de rotación (puede ser por empresas de 1 empleado).

También observamos que se tienen que limpiar/revisar la edad de ingreso y la edad de pérdida de empleo porque nos marca un valor máximo de 220 y 228 años respectivamente y tenemos casos con edad negativa lo cual claramente es atípico y puede ser un problema de origen de datos. En este sentido, observamos que debemos limpiar en este caso eliminando todos aquellos datos con una edad menor a 18 años y superior a 100 años.

El análisis exploratorio de los datos sugerían que mantengamos, de acuerdo al análisis de cuartiles, las edades entre 18 y 62 años. Sin embargo creemos que se deben eliminar las edades menores a 18 años y crear una regla de eliminación que solo suprimirá los registros cuando la edad de pérdida de empleo sea mayor a los 70 años, ya que el préstamo a pensionados o jubilados tiene un tratamiento de riesgo completamente diferente. Otra opción es eliminar el registro cuando exista una discrepancia entre la edad calculada con base en la fecha de nacimiento y la edad reportada en la fecha de pérdida de empleo.

En el caso de datos nulos en edad, estos corresponden al 3.5% de los datos, por lo que se pueden descartar estos registros.

Los datos nulos no representan problema en el caso de la variable vencidos y castigos. Debido a que son registros que no perdieron su empleo o no se tiene el dato del saldo insoluto que el banco tuvo que pagar.

Para las variables categóricas destacamos que existen 8 segmentos para el caso de iniciativa privada y 4 segmentos para administración pública.

Una variable que sí se debe revisar por la alta cantidad de nulos en el tipo de gobierno al que pertenece ya que tiene demasiados nulos, (municipal, estatal, federal). El problema es que son demasiados datos nulos como para imputar datos, por lo que la alternativa puede ser agrupar esta variable como un solo dato como por ejemplo administración pública o simplemente gobierno. Es decir, agrupar como variable categórica, gobierno y los giros privados que sí vienen especificados.