UNIVERSIDAD SERGIO ARBOLEDA

Escuela de Ciencias Exactas e Ingeniería - Maestría en Matemáticas Aplicadas
Técnicas Avanzadas de Minería de Datos y Machine Learning
Taller grupal (2 personas máx.)
2021-I

Doc. Luz Stella Gómez Fajardo

Estudiantes:
Carlos Mauricio Moreno Rojas
Miller Alexander Quiroga Campos.

1. Utilizando la Base de Datos Universo (CrunchBase):

    1. Cuánto capital se ha invertido en LaTAM durante el último año. Desagregue gráficamente por país.
    2. Haga una comparación entre Colombia con cada uno de los otros países. Analice.
    3. ¿Cuáles son los fondos que más invierten en Colombia? Haga un análisis descriptivo de cada uno de ellos.

        3.1 ¿Cuál es la tesis de inversión de cada uno de estos fondos?

    4. Muestre gráficamente los exits de capital privado en Colombia por deal size.
    5. Muestre el crecimiento porcentual mensual de ingresos por inversión en Colombia en comparación con los demás países.
    6. ¿De acuerdo con los hallazgos, qué le hace falta a Colombia para lograr más inversión?

2. Con la unión de las bases de datos, luego de etiquetar con 1 para coincidencias y 0 en caso contrario:

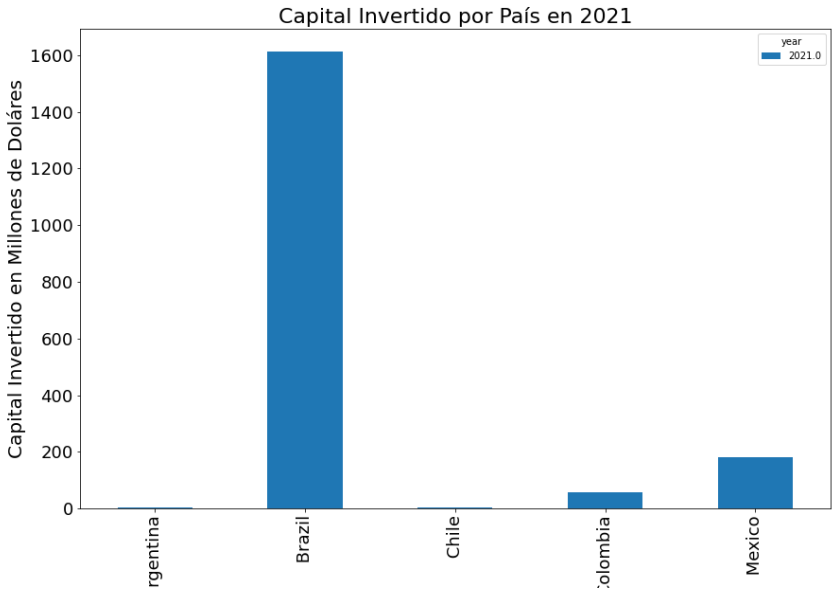
Construir el data warehouse
Validar gráficamente y eliminar aquellas variables que no afectan la variable de respuesta.
Realizar una regresión logística para determinar las características que hacen exitosa una startup para obtener inversión. Hacer el análisis correspondiente.

Solución:

1. Base de datos CrunchBase

Se realiza la limpieza de la data Colombia, y se concatena con las datas de los 10 paises del archivo comprimido.

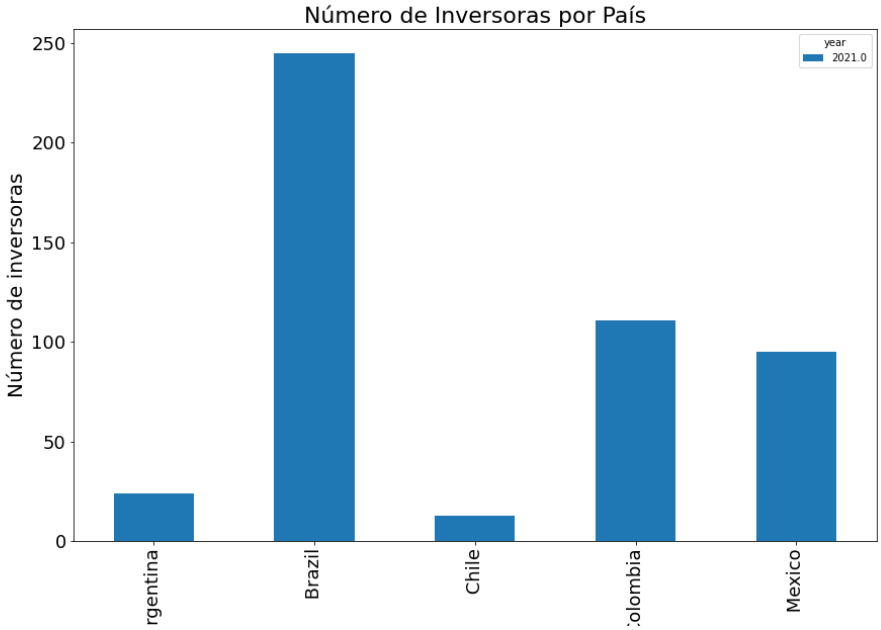
1. Cuánto capital se ha invertido en LaTAM durante el último año. Desagregue gráficamente por país.

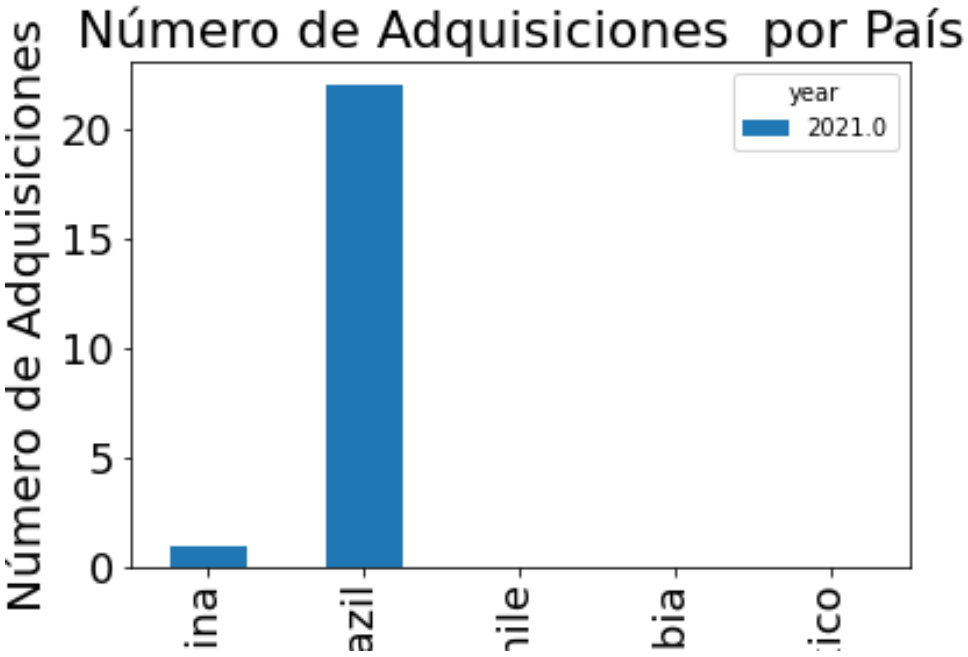


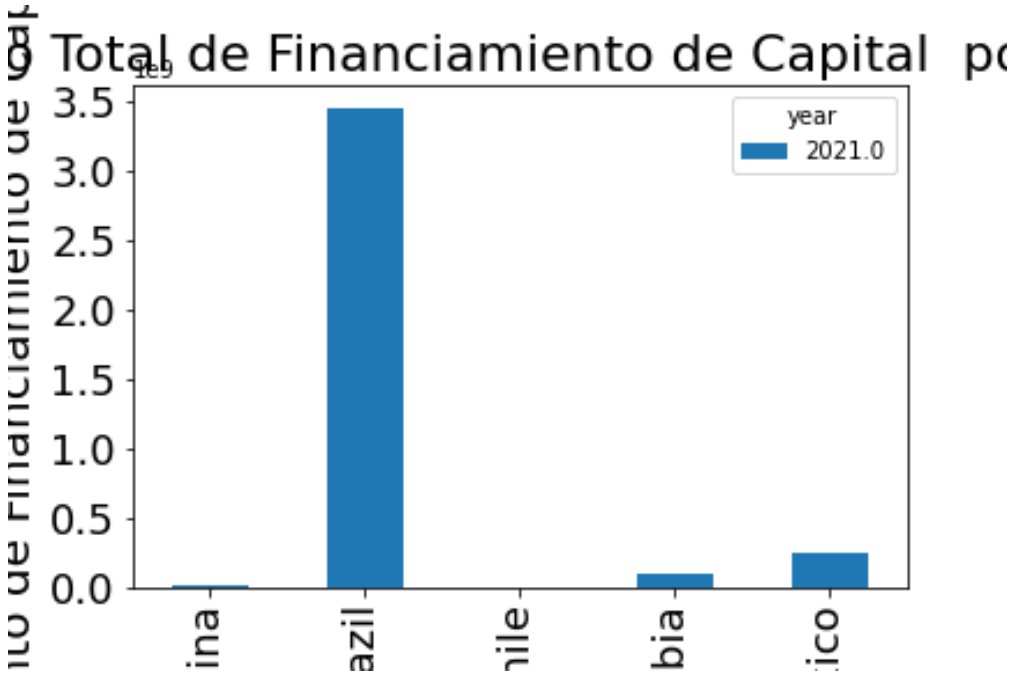
De acuerdo con el gráfico obtenido de los países de Latinoamérica se puede observar que la mayoría del capital invertido es el de Brasil, después México, Colombia y un poco sobresalen argentina y Chile Y Uruguay no aparece en este caso.

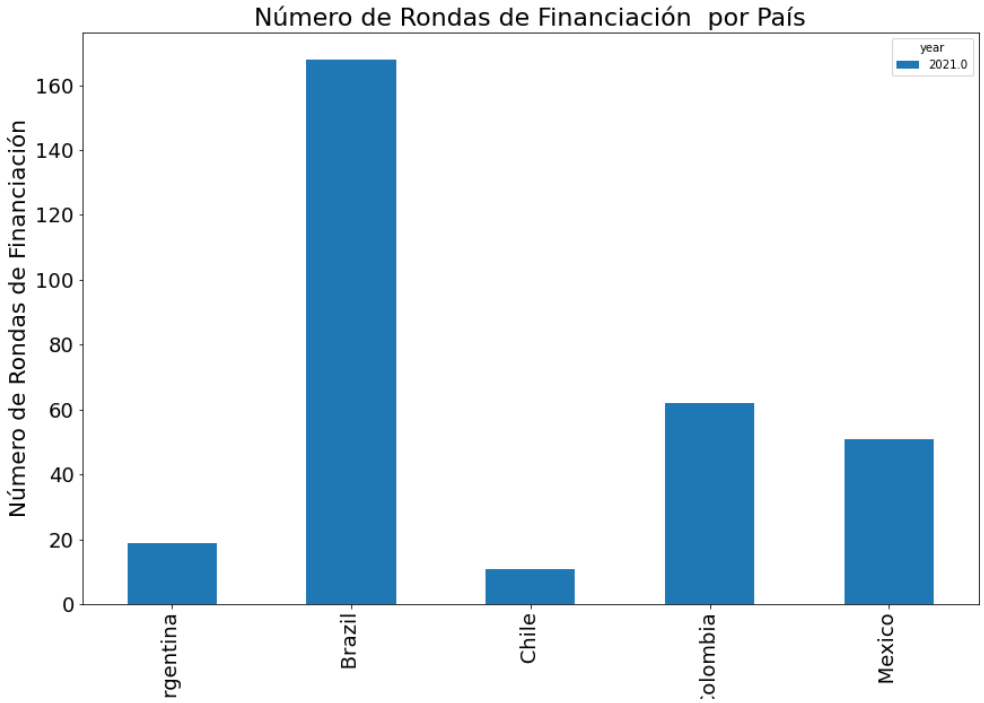
Se puede observar que el nivel de desarrollo de un país como Brasil es dado por la cantidad de capital que se invierte, Colombia sobresale sobre países como Chile y Argentina.

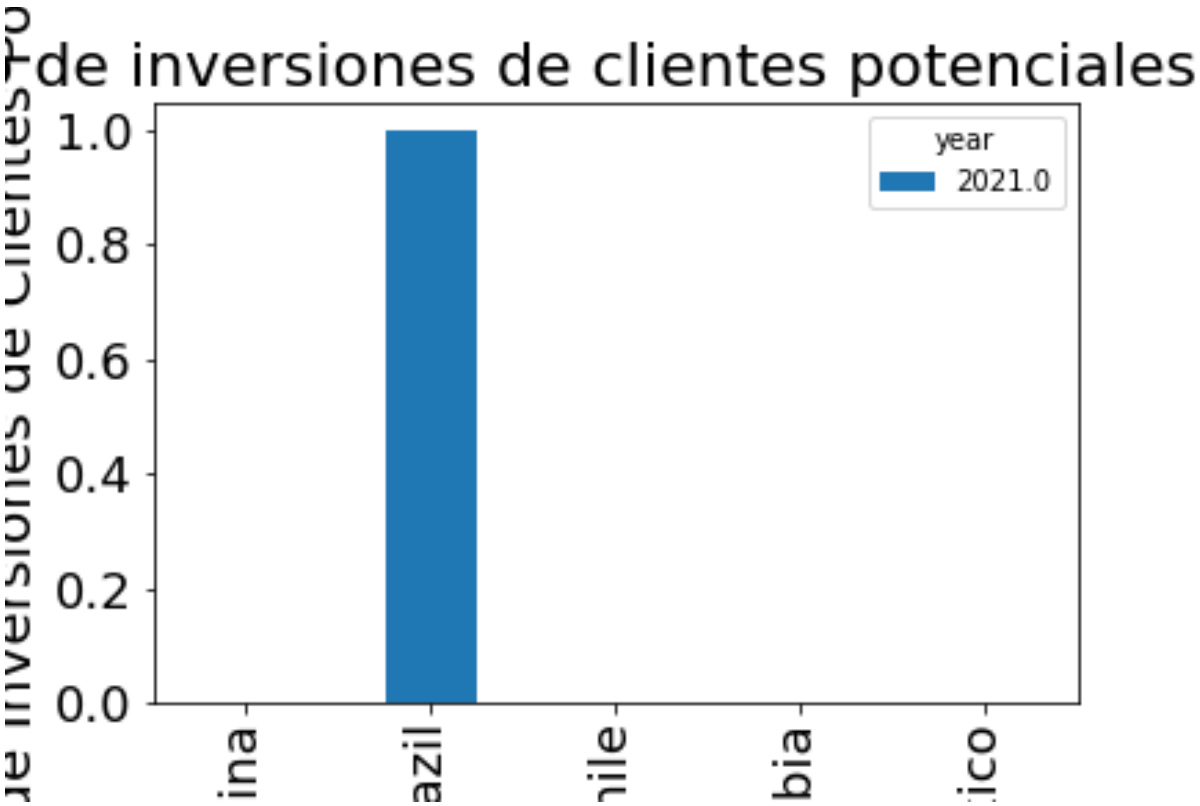
2. Haga una comparación entre Colombia con cada uno de los otros países. Analice.






Total de Financiamiento de Capital
1e9
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
year
2021.0
ina
azil
hile
bia
ico

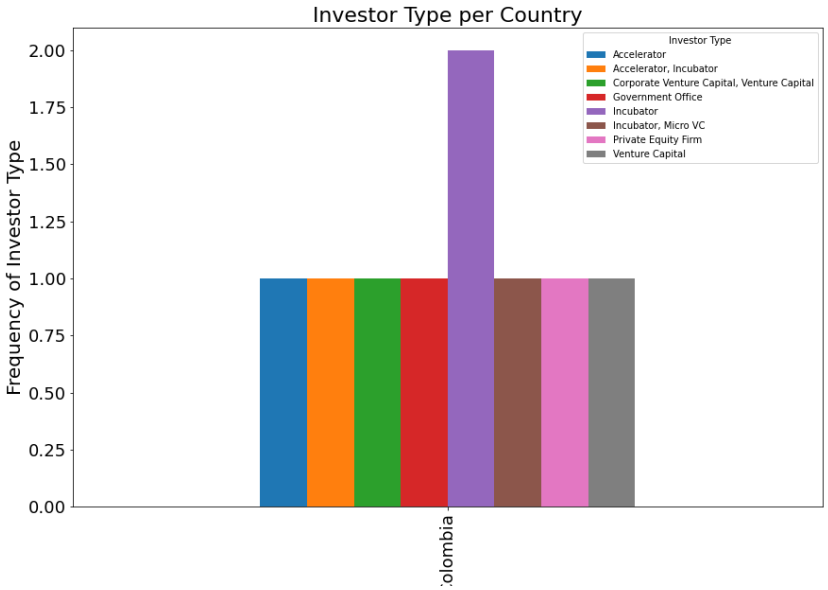

Número de Rondas de Financiación por País
Número de Rondas de Financiación
160
140
120
100
80
60
40
20
0
year
2021.0
rgentina
Brazil
Chile
olombia
Mexico

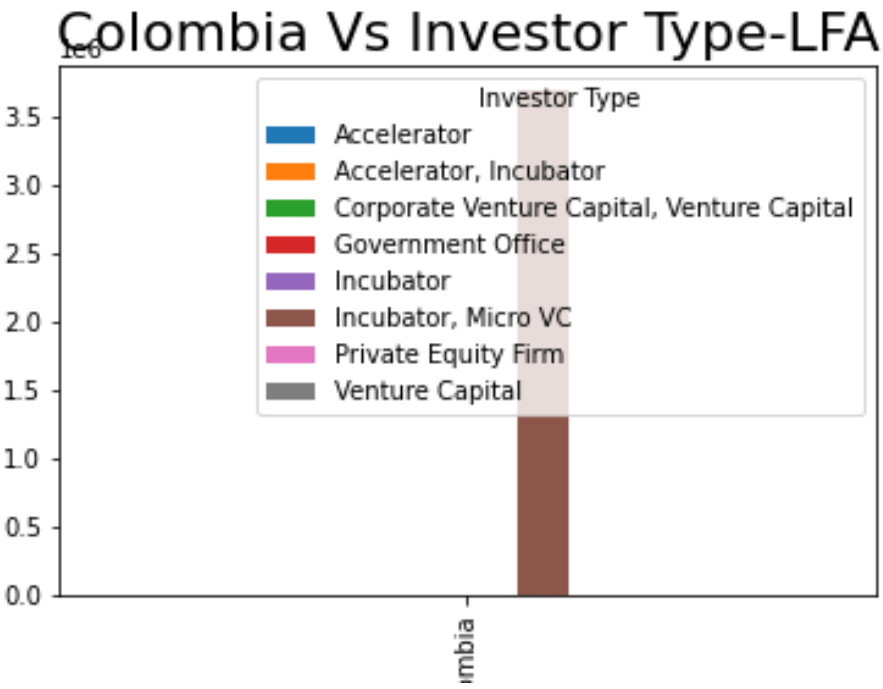


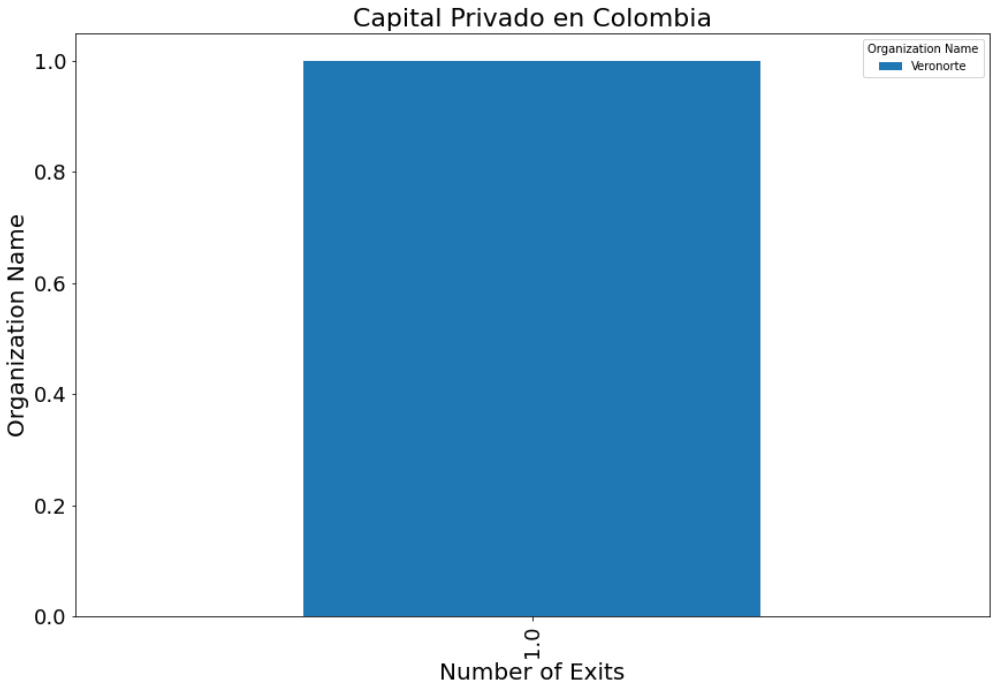
Se realizó el análisis de algunas de las variables dadas dentro de la data dada, para ellos e tomaron los demás países con respecto a Colombia pudiendo observar que Brasil es el país que sobresale en todos los casos, observándose la relevancia que existe en la inversión sobre él, Colombia en la región se puede ver que es un país que es interesante para la inversión, claro que está muy lejos de Brasil. Se puede observar como Colombia se encuentra arriba de México en ciertos casos y como siempre supera a países como Chile y Argentina los cuales tiene un nivel de desarrollo más alto.

3. ¿Cuáles son los fondos que más invierten en Colombia? Haga un análisis descriptivo de cada uno de ellos.

En el gráfico se pueden observar cada uno de los fondos que más invierten en Colombia entre ellos hay uno que es el que sobresale en especial que es el Incubator, Micro VC.






Capital Privado en Colombia
Organization Name
Veronorte
1.0
0.8
0.6
0.4
0.2
0.0
Organization Name
1.0
Number of Exits

2. Con la unión de la base de datos:

- ColombiaCB-5March21.csv
- Top100Startups- Colombia.xlsx
- Empresas Unicorn - Contactos.xlsx

Al realizar la unión de los datos y realizar limpieza, se encuentra que las compañías que se encuentran en común en los 3 archivos son 12:

Value
LA HAUS
CHIPER
LIFTIT
UBITS
LAIKA
TRUORA
ADDI
VOZY
HABI
AFLORE
LENTESPLUS
FRUBANA

Y se crea una variable de bandera, donde estas compañías adoptan un valor de 1, las cuales se encuentran en las 3 datas, y 0 donde no.

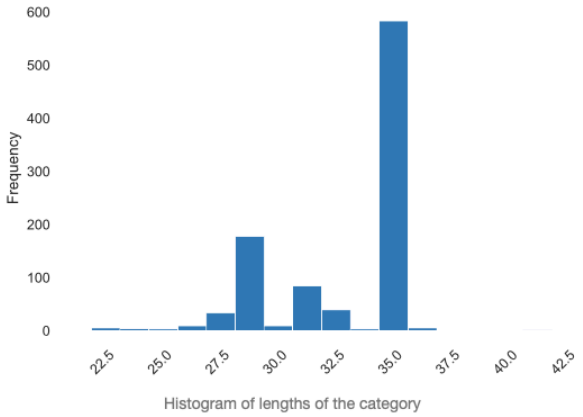
```
In [18]:
    ...: count_int = len(df[df['Intersección']==0])
    ...: #Cantidad de empresas que están en las 3 datas
    ...: count_int2 = len(df[df['Intersección']==1])
    ...:
    ...: pct_int = count_int / (count_int + count_int2)
    ...: pct_int2 = count_int2 / (count_int + count_int2)
    ...:
    ...: print('Porcentaje de Empresas, No estan en las 3 data', pct_int*100)
    ...: print('Porcentaje de Empresas en las 3 data', pct_int2*100)
Porcentaje de Empresas, No estan en las 3 data 98.7460815047022
Porcentaje de Empresas en las 3 data 1.2539184952978055
```

Notamos que la variable de decisión que es la misma tiene un porcentaje de 1,25% con respecto al 98,75%,

Al realizar un report profile, obtenemos como se comportan las varibles:

Overview | Categories | Words | Characters

Value	Count	Frequency (%)
Bogotá, Distrito Especial, Colombia	579	60.5%
Medellín, Antioquia, Colombia	158	16.5%
Cali, Valle del Cauca, Colombia	76	7.9%
Barranquilla, Atlantico, Colombia	24	2.5%
Envigado, Antioquia, Colombia	8	0.8%
Bucaramanga, Santander, Colombia	8	0.8%
Perú, Valle del Cauca, Colombia	8	0.8%
Armenia, Quindio, Colombia	7	0.7%
Manizales, Caldas, Colombia	7	0.7%
Canadá, Cundinamarca, Colombia	7	0.7%
Other values (42)	75	7.8%

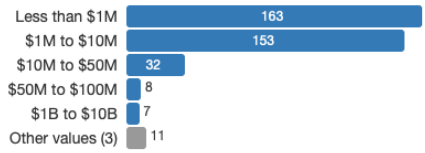


Estimated Revenue Range

Categorical

HIGH CORRELATION
MISSING

Distinct	8
Distinct (%)	2.1%
Missing	583
Missing (%)	60.9%
Memory size	7.6 KiB



Operating Status

Categorical

HIGH CORRELATION

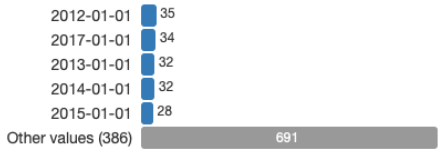
Distinct	2
Distinct (%)	0.2%
Missing	0
Missing (%)	0.0%
Memory size	7.6 KiB

Founded Date

Categorical

HIGH CARDINALITY
MISSING

Distinct	391
Distinct (%)	45.9%
Missing	105
Missing (%)	11.0%
Memory size	7.6 KiB

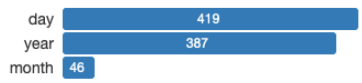


Founded Date Precision

Categorical

HIGH CORRELATION
MISSING

Distinct	3
Distinct (%)	0.4%
Missing	105
Missing (%)	11.0%
Memory size	7.6 KiB



Company Type

Categorical

HIGH CORRELATION
MISSING

Distinct	2
Distinct (%)	0.2%
Missing	17
Missing (%)	1.8%
Memory size	7.6 KiB

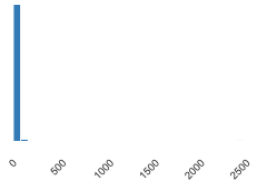
Number of Articles

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

Distinct	32
Distinct (%)	12.5%
Missing	702
Missing (%)	73.4%
Infinite	0
Infinite (%)	0.0%

Mean	16.75294118
Minimum	1
Maximum	2524
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB

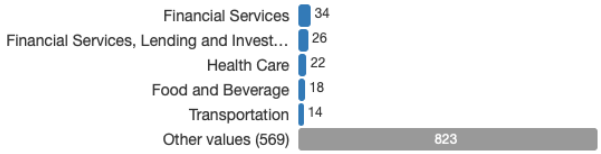


Industry Groups

Categorical

HIGH CARDINALITY
MISSING

Distinct	574
Distinct (%)	61.3%
Missing	20
Missing (%)	2.1%
Memory size	7.6 KiB



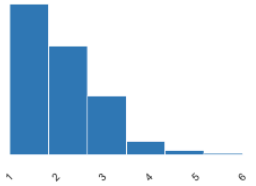
Number of Founders

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

Distinct	6
Distinct (%)	1.4%
Missing	533
Missing (%)	55.7%
Infinite	0
Infinite (%)	0.0%

Mean	1.870283019
Minimum	1
Maximum	6
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB

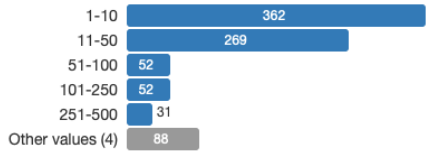


Number of Employees

Categorical

HIGH CORRELATION
MISSING

Distinct	9
Distinct (%)	1.1%
Missing	103
Missing (%)	10.8%
Memory size	7.6 KiB

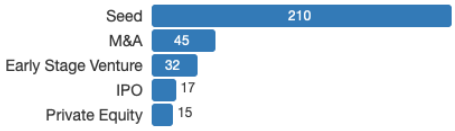


Funding Status

Categorical

HIGH CORRELATION
MISSING

Distinct	6
Distinct (%)	1.9%
Missing	635
Missing (%)	66.4%
Memory size	7.6 KiB

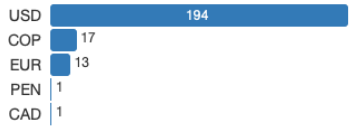


Last Funding Amount Currency

Categorical

HIGH CORRELATION
MISSING

Distinct	5
Distinct (%)	2.2%
Missing	731
Missing (%)	76.4%
Memory size	7.6 KiB

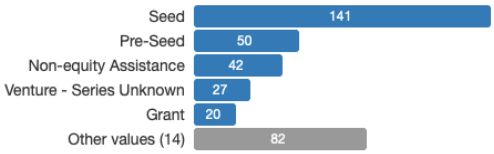


Last Funding Type

Categorical

HIGH CORRELATION
MISSING

Distinct	19
Distinct (%)	5.2%
Missing	595
Missing (%)	62.2%
Memory size	7.6 KiB



Last Equity Funding Amount Currency

Categorical

HIGH CORRELATION
MISSING

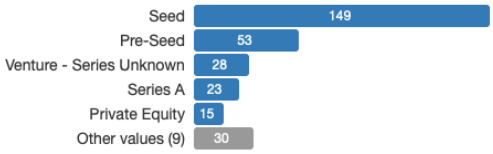
Distinct	4
Distinct (%)	2.0%
Missing	755
Missing (%)	78.9%
Memory size	7.6 KiB

Last Equity Funding Type

Categorical

HIGH CORRELATION
MISSING

Distinct	14
Distinct (%)	4.7%
Missing	659
Missing (%)	68.9%
Memory size	7.6 KiB



Total Equity Funding Amount Currency

Categorical

HIGH CORRELATION
MISSING

Distinct	5
Distinct (%)	2.3%
Missing	736
Missing (%)	76.9%
Memory size	7.6 KiB

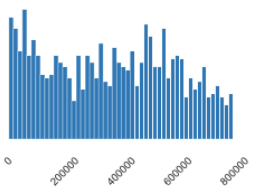
CB Rank (Organization)

Real number ($\mathbb{R}_{\geq 0}$)

HIGH CORRELATION
UNIQUE

Distinct	957
Distinct (%)	100.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%

Mean	363341.2905
Minimum	1575
Maximum	793737
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB



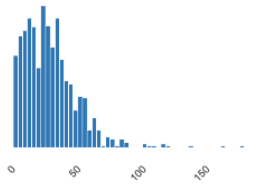
BuiltWith - Active Tech Count

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

Distinct	95
Distinct (%)	10.4%
Missing	41
Missing (%)	4.3%
Infinite	0
Infinite (%)	0.0%

Mean	30.0360262
Minimum	1
Maximum	183
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB



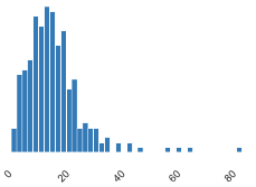
G2 Stack - Total Products Active

Real number ($\mathbb{R}_{\geq 0}$)

MISSING

Distinct	42
Distinct (%)	15.0%
Missing	677
Missing (%)	70.7%
Infinite	0
Infinite (%)	0.0%

Mean	16.025
Minimum	1
Maximum	84
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB



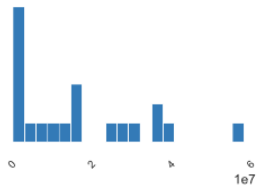
Total Funding Amount_right

Real number ($\mathbb{R}_{\geq 0}$)

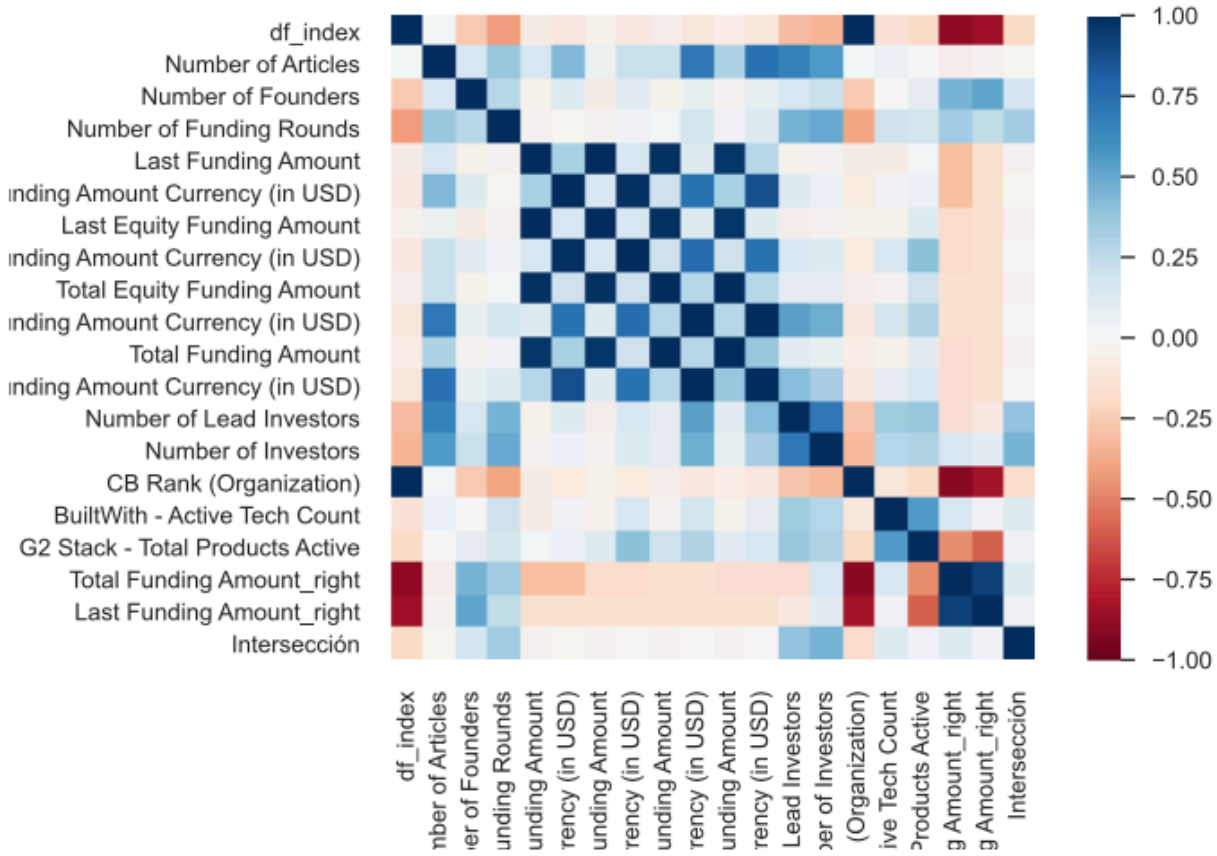
HIGH CORRELATION
MISSING

Distinct	20
Distinct (%)	95.2%
Missing	936
Missing (%)	97.8%
Infinite	0
Infinite (%)	0.0%

Mean	16535000
Minimum	150000
Maximum	58200000
Zeros	0
Zeros (%)	0.0%
Memory size	7.6 KiB



Con su gráfico de correlaciones:



Y se realiza la limpieza de nombres y se elimina variables que no afectan a la variable de respuesta:

```
df = df.drop(['Contact Email'], axis=1) # elimina columna Contact Email
df = df.drop(['Phone Number'], axis=1)  # elimina columna Phone Number
df = df.drop(['Full Description'], axis=1) # elimina columna Full Description
df = df.drop(['Transaction Name URL'], axis=1) # elimina columna Transaction
df = df.drop(['Acquired by URL'], axis=1) # elimina columna Acquired by URL
df = df.drop(['Exit Date'], axis = 1)
df = df.drop(['Exit Date Precision'], axis = 1)
df = df.drop(['Closed Date'], axis = 1)
df = df.drop(['Website'], axis = 1)
df = df.drop(['Twitter'], axis = 1)
df = df.drop(['Facebook'], axis = 1)
df = df.drop(['LinkedIn'], axis = 1)
df = df.drop(['Hub Tags'], axis = 1)
df = df.drop(['Investor Type'], axis = 1)
df = df.drop(['Investment Stage'], axis = 1)
df = df.drop(['Number of Portfolio Organizations'], axis = 1)
df = df.drop(['Number of Investments'], axis = 1)
df = df.drop(['Number of Lead Investments'], axis = 1)
df = df.drop(['Number of Exits'], axis = 1)
df = df.drop(['Number of Exits (IPO)'], axis = 1)
df = df.drop(['Accelerator Program Type'], axis = 1)
df = df.drop(['Accelerator Duration (in weeks)'], axis = 1)
df = df.drop(['School Type'], axis = 1)
df = df.drop(['School Program'], axis = 1)
df = df.drop(['Number of Enrollments'], axis = 1)
df = df.drop(['Number of Founders (Alumni)'], axis = 1)
df = df.drop(['Number of Acquisitions'], axis = 1)
df = df.drop(['Acquisition Status'], axis = 1)
df = df.drop(['Transaction Name'], axis = 1)
df = df.drop(['Acquired by'], axis = 1)
df = df.drop(['Announced Date'], axis = 1)
df = df.drop(['Announced Date Precision'], axis = 1)
```

```
df = df.drop(['Price'], axis = 1)
df = df.drop(['Price Currency'], axis = 1)
df = df.drop(['Price Currency (in USD)'], axis = 1)
df = df.drop(['Acquisition Type'], axis = 1)
df = df.drop(['Acquisition Terms'], axis = 1)
df = df.drop(['IPO Date'], axis = 1)
df = df.drop(['Delisted Date'], axis = 1)
df = df.drop(['Delisted Date Precision'], axis = 1)
df = df.drop(['Money Raised at IPO'], axis = 1)
df = df.drop(['Money Raised at IPO Currency'], axis = 1)
df = df.drop(['Money Raised at IPO Currency (in USD)'], axis = 1)
df = df.drop(['Valuation at IPO'], axis = 1)
df = df.drop(['Valuation at IPO Currency'], axis = 1)
df = df.drop(['Valuation at IPO Currency (in USD)'], axis = 1)
df = df.drop(['Stock Symbol'], axis = 1)
df = df.drop(['Stock Symbol URL'], axis = 1)
df = df.drop(['Stock Exchange'], axis = 1)
df = df.drop(['Last Leadership Hiring Date'], axis = 1)
df = df.drop(['Number of Events'], axis = 1)
df = df.drop(['Apptopia - Number of Apps'], axis = 1)
df = df.drop(['Apptopia - Downloads Last 30 Days'], axis = 1)
df = df.drop(['IPqwery - Patents Granted'], axis = 1)
df = df.drop(['IPqwery - Trademarks Registered'], axis = 1)
df = df.drop(['IPqwery - Most Popular Patent Class'], axis = 1)
df = df.drop(['IPqwery - Most Popular Trademark Class'], axis = 1)
df = df.drop(['Aberdeen - IT Spend'], axis = 1)
df = df.drop(['Aberdeen - IT Spend Currency'], axis = 1)
df = df.drop(['Aberdeen - IT Spend Currency (in USD)'], axis = 1)
df = df.drop(['School Method'], axis = 1)
df = df.drop(['No'], axis = 1)
df = df.drop(['Em'], axis = 1)
df = df.drop(['NIT'], axis = 1)
df = df.drop(['CORREO ELECTRONICO'], axis = 1)
df = df.drop(['TELÉFONO '], axis = 1)
```

```
df = df.drop(['Unnamed: 16'], axis = 1)
df = df.drop(['Ciudad'], axis = 1)
df = df.drop(['Closed Date Precision'], axis = 1)
df = df.drop(['Organization Name URL'], axis = 1)
df = df.drop(['Headquarters Regions'], axis = 1)
df = df.drop(['Founded Date_right'], axis = 1)
df = df.drop(['Last Funding Date_right'], axis = 1)
```

Y se convierte algunas columnas en numéricas que tiene registros con string como las comas, y algunos sin información NAN,

```python
numericColumnsNames = [
    "CB Rank (Company)",
    "Number of Articles",
    "Number of Founders",
    "Number of Funding Rounds",
    "Last Funding Amount",
    "Last Funding Amount Currency (in USD)",
    "Last Equity Funding Amount",
    "Last Equity Funding Amount Currency (in USD)",
    "Total Equity Funding Amount",
    "Total Equity Funding Amount Currency (in USD)",
    "Total Funding Amount",
    "Total Funding Amount Currency (in USD)",
    "Number of Lead Investors",
    "Number of Investors",
    #"Number of Events",
    "CB Rank (Organization)",
    "BuiltWith - Active Tech Count",
    "G2 Stack - Total Products Active"
]

for columnName in numericColumnsNames:
    df1[columnName] = df1[columnName].fillna(0)
```

Se realiza la regresión logística con un r cuadrado de 0.68

```
In [73]: print(log_reg.summary())
                           Logit Regression Results
==============================================================================
Dep. Variable:                      y   No. Observations:                  957
Model:                          Logit   Df Residuals:                      944
Method:                           MLE   Df Model:                           12
Date:                Sat, 10 Apr 2021   Pseudo R-squ.:                 0.6876
Time:                        02:30:06   Log-Likelihood:                -20.141
converged:                      False   LL-Null:                       -64.471
Covariance Type:            nonrobust   LLR p-value:                8.968e-14
==============================================================================================
                                 coef    std err          z      P>|z|      [0.025      0.975]
----------------------------------------------------------------------------------------------
CB Rank (Company)                0.0314      0.016      1.965      0.049    8.78e-05       0.063
Number of Articles              -0.1872      0.114     -1.639      0.101      -0.411       0.037
Number of Founders               0.1919      0.433      0.443      0.658      -0.657       1.041
Number of Funding Rounds         0.0585      0.263      0.222      0.824      -0.457       0.574
Last Funding Amount Currency (in USD)  -8.267e-08    1.2e-06     -0.069      0.945   -2.44e-06    2.28e-06
Last Equity Funding Amount Currency (in USD)   7.977e-08    1.2e-06      0.066      0.947   -2.28e-06    2.44e-06
Total Equity Funding Amount Currency (in USD)  3.381e-08   1.45e-07      0.233      0.816   -2.51e-07    3.19e-07
Total Funding Amount            -4.486e-05      0.129     -0.000      1.000      -0.253       0.253
Total Funding Amount Currency (in USD)   4.483e-05      0.129      0.000      1.000      -0.253       0.253
Number of Investors              0.2124      0.106      1.997      0.046       0.004       0.421
CB Rank (Organization)          -0.0304      0.015     -1.973      0.048      -0.061      -0.000
BuiltWith - Active Tech Count    0.0610      0.024      2.502      0.012       0.013       0.109
G2 Stack - Total Products Active -0.0226      0.039     -0.574      0.566      -0.100       0.055
==============================================================================================
```

Y se realiza una matriz de confusión donde me indica la cantidad de 1 y 0 que predice el modelo.

```
In [74]: yhat = log_reg.predict(df0)#entrenamiento

In [75]: prediction = list(map(round, yhat))

In [76]:
    ...:
    ...: cm = confusion_matrix(Y, prediction)
    ...: print ("Confusion Matrix : \n", cm)
Confusion Matrix :
 [[942   3]
 [  5   7]]
```

La Matrix de confusión me predice 942 ceros y 7 unos, lo cual se deja de predecir 3 ceros y 5 unos.

Sería bueno que la variable a predecir tuviera un p