Es importante conocer y entender el negocio, así como también los datos con los que estamos trabajando por lo que es una buena práctica hacer una introducción explicando de que se trata el dataset y describir que contiene cada columna de los datos.

**INTRODUCCION**

Este dataset contiene el top 100 de videojuegos por cada categoría en Google play Store, con información de las calificaciones y otros datos como precio y numero de instalaciones.

COLUMNAS

* Rank: Posición del videojuego en una categoría particular
* Title: Nombre del juego
* Total ratings: Número total de calificaciones.
* Installs: Numero de instalaciones aproximado
* Average rating: Calificación promedio hasta 5
* Growth (30 days): Crecimiento porcentual en 30 días
* Growth (60 days): Crecimiento porcentual en 60 días
* Price: Precio en dólares
* Category: Categoria del videojuego
* 5 star ratings: Numero de calificaciones de 5 estrellas.
* 4 star ratings: Numero de calificaciones de 4 estrellas.
* 3 star ratings: Numero de calificaciones de 3 estrellas.
* 2 star ratings: Numero de calificaciones de 2 estrellas.
* 1 star ratings: Numero de calificaciones de 1 estrellas.
* Paid: Es un videojuego pago (True) o gratis (False)

ACLARACIONES

* El crecimiento porcentual se calcula teniendo en cuenta la cantidad total de instalaciones y la cantidad total de calificaciones, con ello encontrar el porcentaje de crecimiento promedio.
* Cada puesto en el top 100 fue registrado en un día del año, en total 100 dias.

EMPEZAMOS

IMPORTAMOS LAS LIBRERIAS

#Procesamiento

import numpy as np

import pandas as pd

#Para manejo de fechas

import datetime as dt

#Manejo de valores nulos

import missingno as msn

#Visualizacion

import matplotlib.pyplot as plt

import seaborn as sns

import plotly.graph\_objects as go

CARGAMOS LOS DATOS EN UNA VARIABLE y LEEAMOS LAS 5 PRIMERAS LINEAS CON HEAD

ag= pd.read\_csv('../datasets/android-games.csv')

ag.head()

A picture containing text, monitor, black, screen

Description automatically generated

También podemos observar las ultimas filas de los datos para ver su comportamiento

ag.tail()

A picture containing text, monitor, black, displayed

Description automatically generated

O podemos visualizar una muestra aleatoria

ag.sample(3)

A picture containing text, monitor, black, scoreboard

Description automatically generated

TAMAÑO DEL DATAFRAME

Es importante que sepamos la cantidad de datos que estamos usando si son muchos o pocos dependiendo del análisis que queramos hacer.

ag.shape

Graphical user interface, application

Description automatically generated

Es importante hacer estos pasos previos a la limpieza para conocer el comportamiento de los datos.

**ANALISIS EXPLORATORIO DE LOS DATOS**

Con el siguiente método se mostrarán las columnas y el número de datos que tenemos en cada una teniendo en cuenta que no se contabiliza los datos nulos o basura. Como sabemos, el total de nuestros datos es 1736 por lo que si en alguna columna tenemos menos de ese valor podemos identificar que hay datos basura en la columna correspondiente.

Además, nos da información del tipo de dato en cada columna.

ag.info()

A picture containing text, plaque

Description automatically generated

* Vemos que avarage rating solo tiene 1728 datos y paid 1729 por lo que podemos decir que nos faltan datos en estas columnas.
* El tipo de dato de la fecha, categoría e instalaciones es object.

Python nos ofrece el tipo de dato fecha para un mejor manejo por lo es preferible modificar el tipo de dato de date.

Vamos a ver los datos estadísticos

ag.describe()

Graphical user interface, text

Description automatically generated

Usando T vamos a hacer la transformada de la tabla para tener una mejor visualización

ag.describe().T

Text

Description automatically generated

* GROWTH 30 y 60 tiene algunos datos atípicos observando el valor mínimo es de 0 y que los 75% de los datos llegan hasta 4 pero sus máximos se encuentran en valores de 6 cifras.

Estas observaciones previas se hacen para luego tenerlas en cuenta.

VISUALIZACIONES

Vamos a realizar un histograma respecto a la columna average rating para ver la frecuencia con la que aparecen los valores

#Matplotlib

plt.hist(ag['average rating'])

Chart

Description automatically generated

Como vemos la mayoría de los videojuegos tienen una calificación media 4 estrellas.

Ahora hacemos un gráfico de barras distinto a un histograma

#Seaborn

sns.countplot(x='paid',data=ag)

Chart, bar chart

Description automatically generated

* La mayoría de los juegos son gratuitos.

Ahora vamos a determinar que porcentaje corresponde a juegos pagos y gratis, a partir de un gráfico de barras

plot =(100 \* ag['paid'].value\_counts() / len(ag['paid'])).plot (kind='bar', title='% videojuegos pagos y gratis')

Chart, bar chart

Description automatically generated

Comentando un poco el código, utilizamos value.counts() que sirve para contar la cantidad de valores únicos y a esos valores lo multiplicamos por 100 y dividimos por la cantidad total para así obtener los porcentajes.

Si solo ejecutamos

ag['paid'].value\_counts()

Graphical user interface

Description automatically generated with medium confidence

Vemos la cantidad que existe de cada valor

Con el siguiente comando vamos a analizar las dos variables que presentaban datos atípicos y esto lo hacemos a través de graficas

#Plotly

fig = go.Figure()

fig.add\_trace(go.Box(y=ag['growth (30 days)']))

fig.add\_trace(go.Box(y=ag['growth (60 days)']))

fig.show()

Primero me salió el siguiente error

[ValueError: Mime type rendering requires nbformat>=4.2.0 but it is not installed](https://stackoverflow.com/questions/66557543/valueerror-mime-type-rendering-requires-nbformat-4-2-0-but-it-is-not-installed)

Y se solucionó con el siguiente comando:

pip install --upgrade nbformat

Graphical user interface, application, Teams

Description automatically generated

Este grafico interactivo nos permite obtener además de una visión de los valores atípicos, también, permite visualizar valores de probabilidad tales como:

* Media
* Mediana
* Desviación estándar

A picture containing graphical user interface

Description automatically generated

Chart

Description automatically generated with medium confidence

**DESCRIPTIVAS**

Descripción de variables categóricas:

ag.describe(include='O')

Utilizamos el parámetro include=’O’ para que solo busque las variables que son de tipo objecto o que son cadenas de caracteres y sacar información de:

* La cantidad de datos disponibles
* Los valores únicos
* Top que seria como la moda el dato que más se repite
* Freq que es la cantidad de veces que aparece el top.

Graphical user interface, application, website

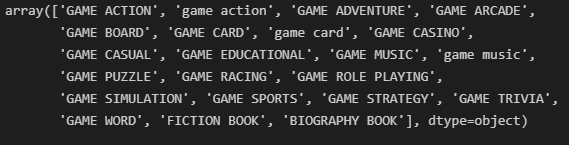
Description automatically generated

Para la fecha vemos que tenemos 121 valores, pero en realidad estos deben ser 100 ya que se realizo un registro por cada una de las posiciones del top en un día diferente y todos de manera consecutiva.

En la columna installs vemos que en top el valor es tipo cadena cuando es una cadena, pero es preferente que sea de tipo número.

Valores únicos de las categorías

ag['category'].unique()



Vemos que tenemos para la misma categoría distintos formatos (minúsculas y mayúsculas). Y es el caso de:

* Game action
* Game card

Luego vemos que fiction book y biography book no corresponde con la categoría de juegos.

Se hace estas observaciones para luego realizar la limpieza.

Ahora hacemos lo mismo con la fecha

ag['Date'].unique()

Text

Description automatically generated

Al comienzo vemos que el formato de la fecha es DD/MM/AAAA y cada valor es de tipo número, pero lo últimos datos están en formato MM/DD/AAAA además que el mes es de tipo String.

Con pandas.to\_datetime vamos a indicar que el tipo de datos que estamos leyendo es de tipo fecha y vamos a utilizar datetime.strftime para dar un mismo formato de fecha.

ag['Date']=pd.to\_datetime(ag['Date']).dt.strftime("%m/%d/%y")

Verificamos

Text

Description automatically generated

Y por ultimo

ag['installs'].unique()



Los valores M y K tienen un valor determinado lo cual podrían modificarse y así tener un tipo de dato integer para un mejor análisis de este atributo

* M = 10000000
* K = 1000

OBSERVACIONES

* Valores nulos en las variables paid y average rating
* El tipo de dato de fecha es Object
* Variable installs es categoría
* Variables growth (30 days) y growth (60 days) contiene datos atipicos.