**Proyecto de evaluación Actividad 1**

Logotipo

Descripción generada automáticamente

**Estudiante: Juan Carlos González Ibarra**

**Correo:** [**jcarlos0284@gmail.com**](mailto:jcarlos0284@gmail.com)

**Fecha: 17 de mayo, 2024**

**Contenido**

1. Tema
   1. Planteamiento del problema
   2. Objetivos
2. Metodología de Desarrollo
3. Extracción de Datos

3.1 Fuentes de extracción de datos

3.2 Que es la extracción de datos

3.3 Extracción de datos con Python

1. Data Clean
2. Almacenamiento base de datos
3. Algoritmos de Aprendizaje Supervisado y No Supervisado
4. Modelado de Datos
5. Análisis exploratorio de Datos (AED)
6. Resultados de AED
7. Visualización de datos
8. Conclusión

**1. Tema**

**1.1 Planteamiento del problema**

La industria de los videojuegos ha experimentado un crecimiento significativo, impulsado por la popularidad de los videojuegos como formas de entretenimiento digital y el aumento en el número de jugadores. Sin embargo, este crecimiento presenta desafíos importantes. La constante evolución de los videojuegos, impulsada por avances tecnológicos y cambios en los hábitos de consumo, ha creado incertidumbre sobre cómo anticipar y adaptarse a las demandas cambiantes del mercado.

Además, el lapso de la pandemia y las medidas de confinamiento incremento esta tendencia, con un aumento notable en el número de jugadores que utilizan plataformas como Nintendo, Xbox y Playstation. Este cambio en el comportamiento de los consumidores ha llevado a un aumento del 6,3% en el número de usuarios de consolas digitales en Estados Unidos. La proliferación de smartphones y el desarrollo de tecnologías como la realidad virtual han ampliado aún más el acceso a los videojuegos, atrayendo a una nueva generación de jugadores.

En este contexto de intensa competitividad y demanda creciente de innovación, las empresas del sector enfrentan el desafío de mantenerse al día con las expectativas del público y encontrar formas innovadoras de ofrecer experiencias de juego únicas y atractivas. La idea del “metaverso” surge como una posible solución, ofreciendo un mundo virtual en el que los usuarios pueden crear y compartir sus propias experiencias.

Por lo anterior, el proyecto plantea desarrollar una herramienta de predicción que pueda identificar con precisión qué estilos de videojuegos tienen el potencial de éxito.

**1.2 Objetivos**

* Plantear un método de desarrollo para la extracción, limpieza, almacenamiento y análisis de datos de videojuegos exitosos y no exitosos, considerando variables como género, plataforma de lanzamiento, reseñas de usuarios, ventas, entre otros.
* Diseñar un modelo predictivo basado en machine learning que pueda predecir el potencial de éxito de un videojuego en función de sus características específicas.
* Evaluar la eficacia del modelo de predicción mediante la comparación de las predicciones con el desempeño real de los videojuegos.

**2. Metodología de Desarrollo**

Para realizar el proyecto se desarrollarán módulos de código para cada uno de los puntos planteados en los objetivos para que se pueda permitir **establecer las variables del tipo de Genero y/o Plataforma del Videojuego a desarrollar y poder determinar (experimentalmente) la predicción de ventas de ese videojuego tomando como referencia la variable copia de ventas del videojuego**, así que, se va trabajar cada tarea de la siguiente forma:

1. Investigar las fuentes donde se puede obtener los datos para el proyecto.
2. Extraer los datos vía Web Scraping
3. Data Clean:
   * Carga de Datos
   * Preprocesar los datos.
   * Eliminar variables irrelevantes.
   * imputar valores faltantes.
   * Limpiar los nombres de los atributos que no se vayan a utilizar.
4. Almacenar los datos en un sistema de base de datos
5. Definición de la(s) *Variable(s) Objetivo(s).*
6. Generar un Modelo Predictivo Supervisado con Algoritmo de:
   * Regresión: Este algoritmo puede predecir un valor continuo (como las ventas globales de un videojuego). Se trata de encontrar una relación entre las variables de entrada y la variable de salida que puede ser representada como una ecuación matemática. Los algoritmos de regresión son: regresión lineal, la regresión polinomial y la regresión de árbol de decisión.
   * Clasificación: Este algoritmo puede predecir una categoría o clase (como el género de un videojuego). Se trata de encontrar una frontera de decisión que separe las diferentes clases en el espacio de las variables de entrada. Los algoritmos de clasificación son: regresión logística, los árboles de decisión, las máquinas de vectores de soporte y las redes neuronales.
   * Predicción: Este algoritmo se utiliza para predecir. La diferencia radica en el tipo de variable de salida que se trata de predecir.
7. Evaluar el modelo.
8. Implementar el modelo.
9. Realizar un Análisis Exploratorio de los Datos para entender la distribución de las variables para conseguir el objetivo del proyecto:
   * Análisis de Correlación de Ventas Globales por Consola y Genero.
   * Análisis de Ventas Globales de Videojuegos por Plataforma.
   * Análisis de Ventas Globales por Género.
   * Análisis de Ventas Globales por Compañía de Desarrollo.

**3. Extracción de datos.**

**3.1 Fuentes de extracción de datos**

En este proyecto se cuenta con un conjunto de datos obtenidos a través de Web Scraping de las páginas web:

* https://www.mobygames.com
* https://www.kaggle.com/datasets/sidtwr/videogames-sales-dataset
* https://www.kaggle.com/datasets/gregorut/videogamesales

**3.2 Que es la extracción de datos**

La extracción de datos es un proceso crucial para cualquier organización que desee aprovechar al máximo su información para la toma de decisiones. Permite recuperar datos de diversas fuentes y transformarlos en un formato utilizable para su posterior análisis o almacenamiento. Los datos pueden provenir de bases de datos, hojas de cálculo, sitios web, archivos PDF, correos electrónicos, entre otros.

La extracción de datos se puede utilizar para descubrir patrones o relaciones ocultas dentro de los datos para tomar decisiones informadas o crear modelos predictivos.

La metodología de la extracción de datos implica los siguientes pasos:

* **Identificación de fuentes de datos**: Determinar de dónde provienen los datos necesarios y qué método utilizar para cada fuente, como consultas SQL, APIs o software de **web scrapping**.
* **Establecimiento de conexiones**: Establecer conexiones con las fuentes de datos seleccionadas utilizando métodos específicos según el tipo de fuente.
* Consulta o recuperación de datos: Recuperar datos específicos utilizando consultas SQL, extracción de texto mediante OCR u otras técnicas según sea necesario.
* **Transformación y carga de datos**: **Transformar los datos extraídos** para cumplir con el formato requerido, incluyendo **la limpieza** de datos, la **normalización** y el **enriquecimiento**, y **guardarlos** en un destino, como **archivos** **planos** o **bases** **de** **datos**.

Para este paso se utilizará la técnica de **web** **scraping** con el lenguaje de programación **Python** versión 3.

El web scraping es una técnica utilizada para extraer datos de sitios web de manera automatizada. Permite recopilar información de páginas web de forma sistemática, convirtiendo los datos no estructurados en un formato estructurado y utilizable.

Para realizar web scraping en Python, hay varias bibliotecas útiles disponibles son:

* **Beautiful Soup:** Es una biblioteca Python que se utiliza para extraer datos de archivos HTML y XML. Proporciona métodos simples para navegar y buscar en el árbol de análisis, lo que la hace muy útil para la extracción de datos de páginas web.
* **Requests**: Aunque no es específicamente una biblioteca de web scraping, Requests es una biblioteca HTTP para Python que permite realizar solicitudes HTTP de manera sencilla. Es muy útil para descargar el contenido HTML de una página web antes de aplicar técnicas de web scraping con Beautiful Soup.
* **Selenium:** Es una herramienta poderosa para automatizar navegadores web. Puede ser útil para interactuar con páginas web que requieren JavaScript para cargar o realizar acciones dinámicas. Selenium puede controlar un navegador web y simular la interacción del usuario.

**3.3 Extracción de datos con Python**

1. Para utilizar estas **bibliotecas** se ejecutarán los siguientes comandos para instalarlos en el entorno del lenguaje Python:

!pip install pandas

!pip install html5lib

!pip install beautifulsoup4

!pip install requests

!pip install selenium

Texto

Descripción generada automáticamente con confianza baja

1. Declarar las bibliotecas

from selenium import webdriver

from bs4 import BeautifulSoup

import pandas as pd

import time

1. Se crea un dataframe para almacenar la información que se extraiga.

#Crear un DataFrame vacío para almacenar los datos

all\_data = pd.DataFrame()

1. Se define el tipo de driver de web browser que se utilizara para obtener la información.

# Especifique la ubicación del driver de Firefox

driver = webdriver.Firefox()

1. **Se crea un bucle que iterara a través del rango de números de página para extraer datos del sitio web MobyGames.** Para cada número de página en el rango especificado, el código construye una URL correspondiente a esa página, realiza una solicitud GET al sitio web utilizando Selenium, espera a que el JavaScript genere la tabla de datos, parsea el contenido HTML de la página utilizando BeautifulSoup, busca la tabla que contiene los datos, y luego se guarda en el DataFrame de Pandas \*\*all\_data\*\* a partir de esta tabla. Una vez que itera a través de todas las páginas, cierra el driver de Selenium.

# Iterar a través de un rango de números de página

# Supongamos que hay 10 páginas

for i in range(1, 30):

    # Construir la URL para la página actual

    url = f"https://www.mobygames.com/game/sort:moby\_score/page:{i}/"

    # Hacer una petición GET al sitio web

    driver.get(url)

    # Espera que el JavaScript genere la tabla (puede que necesites ajustar el tiempo de espera)

    time.sleep(2)

    # Parsear el contenido de la página con BeautifulSoup

    soup = BeautifulSoup(driver.page\_source, 'html.parser')

    # Buscar la tabla en el contenido de la página

    table = soup.find('table')

    # Crear un DataFrame de Pandas a partir de la tabla

    df = pd.read\_html(str(table))[0]

    # Agregar los datos al DataFrame total

    all\_data = pd.concat([all\_data, df])

# Cerrar el driver de Selenium

driver.quit()

1. Visualizar los datos que contiene el Data Frame **all\_data**

print(all\_data)

Captura de pantalla de un celular con texto

Descripción generada automáticamente

1. Se guardan los datos que contiene el Data Frame all\_data en un archivo con extensión csv llamado "consolas.csv".

all\_data.to\_csv('consolas.csv', index=False)

1. Se realiza este mismo proceso con las otras páginas que están como fuentes para tener más de un banco de datos que pueda ayudar a realizar la herramienta de predicción.

Se tiene como resultados los archivos csv:

* Captura de pantalla de computadora

  Descripción generada automáticamente'juegos\_mas\_vendidos.csv'
* 'consolas.csv'
* 'consolas2.csv'
* 'moby\_score.csv'
* 'vgsales.csv'
* 'video\_games.csv

estos archivos contienen diferente información que llevara a cabo el proceso **Data Clean** para eliminar, normalizar y posterior guardar los datos con las variables necesarias para el análisis exploratorio de los datos.

**8. Bibliográfia**

* Documentación oficial de Python: https://docs.python.org/3/
* Documentación oficial de Pandas: https://pandas.pydata.org/docs/
* Documentación oficial de Matplotlib: <https://matplotlib.org/stable/contents.html>
* Documentación oficial de Seaborn: <https://seaborn.pydata.org>
* Documentación oficial de Scikit-learn: https://scikit-learn.org/stable/
* Documentación oficial de Numpy: https://numpy.org/doc/stable/
* Analisis Exploratorio de Datos https://www.aprendemachinelearning.com/analisis-exploratorio-de-datos-pandas-python/
* Conceptos en Python <https://www.geeksforgeeks.org>
* Dudas <https://stackoverflow.com/questions/tagged/pandas+python>
* Analisis de datos <https://ocw.uc3m.es/course/view.php?id=230>
* Diccionarios de datos en data frame <https://github.com/nsheikh23/COVID_StockMarket_Analysis/blob/master/52_Week.ipynb>
* Procesamiento de data frames en pandas https://barcelonageeks.com/eliminar-una-o-varias-columnas-de-pyspark-dataframe/
* Data Clean <https://github.com/mramshaw/Data-Cleaning>
* Ploteo de datos <https://github.com/tomimester/python-histogram/blob/master/plot-histogram-python-pandas.ipynb>
* Data Cleaning in Python: the Ultimate Guide (2020) <https://towardsdatascience.com/data-cleaning-in-python-the-ultimate-guide-2020-c63b88bf0a0d>
* Regresión lineal en Python: un análisis detallado <https://www.cienciadedatos.net/documentos/py10-regresion-lineal-python.html>
* Aplicando Machine Learning para predecir éxitos de videojuegos   
  <https://www.saturdays.ai/projects/Videojuegos.html>
* Machine Learning para la predicción de ventas https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/
* McKinney, Wes. Python for Data Analysis. O'Reilly Media, Inc, 2017.
* VanderPlas, Jake. Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media, Inc, 2016.