Práctica 2

Flores Gomez Laura Carelly
Galindo Reza Miguel
Garduño Baldazo Cristian
Martínez Hernández Elias Helaman
Nájera Balderas Jorge Gabriel
Zetina Martinez Angélica

March 13, 2025

Contents

1 Infraestructura

El primer paso en la práctica fue la creación de una base de datos en Amazon RDS (Relational Database Service) y una instancia virtual EC2 para facilitar la conexión a dicha base de datos desde el equipo local.

Creación de la base de datos en RDS Se configuró una base de datos PostgreSQL en AWS utilizando Amazon RDS. Esta base de datos se diseñó para permitir la escalabilidad y facilitar la gestión de los datos. Amazon RDS automatiza tareas como la instalación, aplicación de parches, copias de seguridad y recuperación ante desastres. Se estableció el endpoint de la base de datos como:

`practica-ec2-db.cdck04ywgz89.us-east-2.rds.amazonaws.com`

Configuración de la instancia EC2 Se configuró una instancia EC2 (Elastic Compute Cloud) en la misma región que el RDS para crear un entorno seguro y controlado de conexión. La instancia EC2 actúa como un puente seguro para que el equipo local pueda acceder a la base de datos RDS. Se configuró con un grupo de seguridad que permite conexiones SSH y tráfico únicamente desde la dirección IP del equipo local.

Puente SSH para la conexión Para permitir el acceso desde el equipo local a la base de datos RDS, se estableció un túnel SSH utilizando el siguiente comando:

```
ssh -i "C:\Users\migue\Downloads\prueba.pem" -L 5432:
    practica-ec2-db.cdck04ywgz89.us-east-2.rds.amazonaws.com
    :5432 ubuntu@18.118.30.188
```

Este comando cumple con los siguientes elementos técnicos: - El flag ‘-i‘ indica la ruta del archivo PEM (clave privada) necesario para autenticar la conexión SSH. - La opción ‘-L‘ crea un túnel local que redirige el puerto 5432 del equipo local al puerto 5432 del endpoint de RDS, permitiendo que las conexiones PostgreSQL accedan como si estuvieran en el entorno local. - La dirección ‘ubuntu@18.118.30.188‘ especifica el usuario y la dirección IP pública de la instancia EC2, que sirve como intermediario para la conexión segura.

Conexión a la base de datos PostgreSQL Una vez establecido el túnel SSH, se conectó a la base de datos con el siguiente comando:

```
psql –h localhost –U postgres –d Proyecto
```

Este comando permite conectarse directamente al endpoint del RDS utilizando ‘localhost‘ gracias al túnel SSH previamente creado.

2 Introducción

El análisis se basa en un conjunto de datos que recopila información sobre factores que pueden influir en la productividad diaria de las personas. Este conjunto de datos incluye 5000 registros y 15 columnas que contienen información diversa como hábitos de sueño, niveles de ejercicio, consumo de cafeína y productividad.

Las principales variables del conjunto de datos son:

'Date' Fecha del registro.

'Person$_ID$' Identificador único de cada persona.

'Age' Edad del individuo.

'Gender' Género del individuo.

'Sleep Start Time' y **'Sleep End Time'**: Horas en que inicia y finaliza el ciclo de sueño.

'Total Sleep Hours' Total de horas dormidas por noche.

'Sleep Quality' Puntaje de calidad del sueño (escala del 1 al 10).

'Exercise (mins/day)' Minutos dedicados a la actividad física diaria.

'Caffeine Intake (mg)' Cantidad de cafeína consumida diariamente.

'Screen Time Before Bed (mins)' Minutos de exposición a pantallas antes de dormir.

'Work Hours (hrs/day)' Horas dedicadas al trabajo diariamente.

'Productivity Score' Puntuación que mide el nivel de productividad diaria.

'Mood Score' Puntuación que evalúa el estado de ánimo del individuo.

'Stress Level' Nivel de estrés reportado.

El análisis se centra en identificar patrones y correlaciones entre estas variables y el nivel de productividad, utilizando técnicas de exploración de datos, preprocesamiento y modelado predictivo.

3 Importación de Librerías y Carga de Datos

El primer paso fue la importación de librerías esenciales para el análisis de datos:

Listing 1: Importación de librerías

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
```

pandas y **numpy** Se utilizan para manipulación y análisis de datos, facilitando el manejo de estructuras como DataFrames y matrices.

matplotlib y **seaborn** Librerías para la visualización de datos que permiten obtener gráficos claros y efectivos.

scikit-learn Se emplea para realizar la partición del conjunto de datos y para calcular la precisión del modelo predictivo.

Posteriormente, se realizó la carga del conjunto de datos:

Listing 2: Carga de datos

```
df = pd.read_csv("datos.csv")
df.head()
```

Este código carga un archivo CSV denominado 'datos.csv' en un DataFrame de pandas. El comando 'df.head()' muestra las primeras 5 filas del conjunto de datos para una inspección visual rápida.

4 Análisis Exploratorio de Datos (EDA)

El análisis exploratorio se centra en comprender la distribución y las relaciones entre las variables del conjunto de datos.

Listing 3: Análisis exploratorio de datos

```
print(df.describe())
sns.pairplot(df)
plt.show()
```

El método 'df.describe()' Proporciona estadísticas descriptivas como la media, desviación estándar, valores mínimos y máximos, entre otros. estadísticas descriptivas como la media, desviación estándar, valores mínimos y máximos, entre otros.

‘sns.pairplot(df)‘ Genera un gráfico de pares que permite observar las relaciones entre las variables numéricas. Esto es útil para identificar posibles correlaciones y patrones visuales. un gráfico de pares que permite observar las relaciones entre las variables numéricas. Esto es útil para identificar posibles correlaciones y patrones visuales.

5 Preprocesamiento de Datos

El preprocesamiento es una fase crítica que asegura que los datos estén limpios y listos para el modelo predictivo. Incluye:

Listing 4: Preprocesamiento de datos

```
df.fillna(df.mean(), inplace=True)
df = pd.get_dummies(df, drop_first=True)
```

Gestión de valores nulos ‘df.fillna(df.mean(), inplace=True)‘ rellena cualquier valor nulo utilizando el valor medio de cada columna numérica. Esta técnica es común cuando los valores ausentes son aleatorios.

Codificación de variables categóricas ‘pd.get$_d$*ummies*()‘*conviertevariablescategóricasenvariablesdumn* *True*‘*eliminaunadelascategoríasparaevitarcolinealidad.*

6 Modelo Predictivo

Se implementó un modelo de regresión lineal para predecir la variable objetivo:

Listing 5: Modelo predictivo

```
from sklearn.linear_model import LinearRegression

X = df.drop("target", axis=1)
y = df["target"]
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

model = LinearRegression()
model.fit(X_train, y_train)

predictions = model.predict(X_test)
print(f"Precisi n␣del␣modelo:␣{accuracy_score(y_test,␣
    predictions)}")
```

División de datos `train_test_split` divide el conjunto de datos en conjuntos de entrenamiento (80%) y prueba (20%) para evaluar el rendimiento del modelo.

Modelo de regresión lineal Se crea una instancia del modelo `LinearRegression` y se entrena con los datos de entrenamiento mediante `model.fit()`..

Predicción y evaluación El modelo predice los valores de la variable objetivo en el conjunto de prueba. La precisión se calcula utilizando `accuracy_score()`. (Nota: En el caso de regresión lineal, sería más apropiado utilizar una métrica como el RMSE o R2 en lugar de `accuracy_score()`).. (Nota: En el caso de regresión lineal, sería más apropiado utilizar una métrica como el RMSE o R2 en lugar de 'accuracy$_s core$').

7 Interpretación de Resultados

El modelo utilizado fue 'RandomForest', un algoritmo de aprendizaje supervisado basado en la creación de múltiples árboles de decisión que trabajan en conjunto para obtener una predicción más precisa.

Los resultados mostraron las siguientes métricas:

MSE (Error cuadrático medio) Esta métrica mide el promedio de los errores al cuadrado entre los valores reales y las predicciones. En este caso, se obtuvo un valor de **8.5402**, indicando que el modelo tuvo errores moderados en sus predicciones.

RMSE (Raíz del error cuadrático medio) El valor obtenido fue **2.9216**, lo que representa el error promedio en las mismas unidades que la variable objetivo.

MAE (Error absoluto medio) Con un valor de **2.533**, indica que, en promedio, el modelo se desvía por aproximadamente 2.53 unidades en sus predicciones.

R2 Score Este valor fue de **-0.0224**, lo que indica que el modelo no logra explicar significativamente la variabilidad de los datos. Un valor negativo del R2 sugiere que el modelo es peor que simplemente predecir el valor medio de la variable objetivo.

Importancia de Variables El gráfico de importancia de características mostró que algunas variables tuvieron una mayor influencia en las predicciones del modelo. En particular, la variable 'Work Hours (hrs/day)' fue la más relevante, seguida por 'Total Sleep Hours' y 'Caffeine Intake (mg)'. Estas tres variables tuvieron un mayor impacto en la predicción del modelo, destacando que el número de horas de trabajo influye significativamente en la variable respuesta.