



Equipo 3

SISTEMA PROACTIVO DE APOYO A LA TOMA DE DECISIONES INTELIGENTE (IDSS)

Avance 1

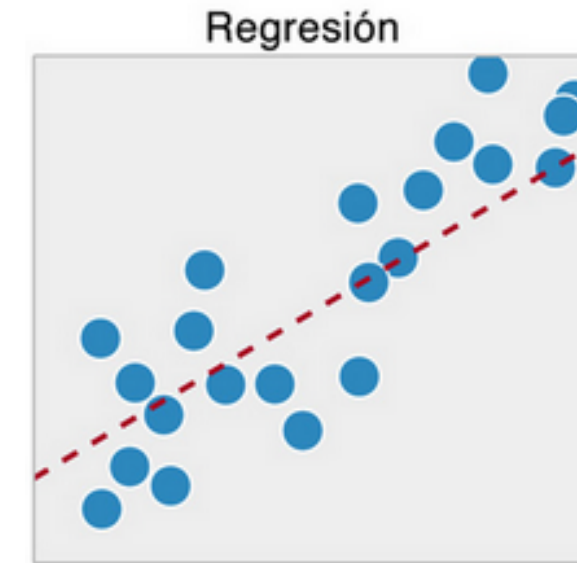
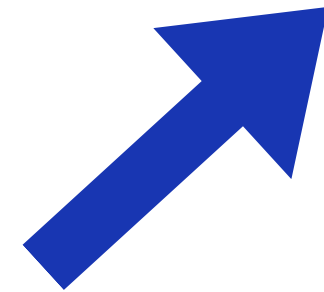


Manuel Antonio Acevedo Ham	A01410910
Jorge Daniel Amezola Gutiérrez	A01793759
Ana Clemencia Aristizábal Londoño	A01795433
Alan Samael Arriaga Castillo	A01153355
Kevin Balderas Sánchez	A01795149

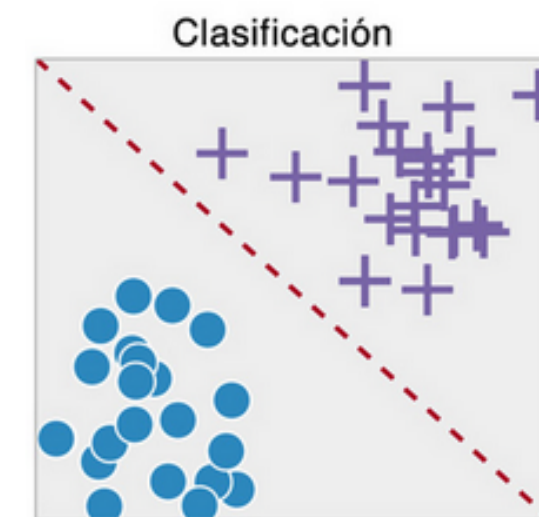
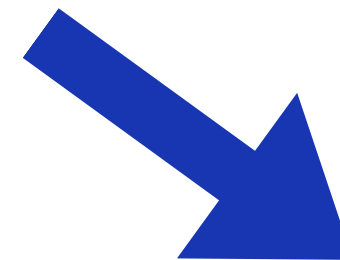
Acercamiento inicial del problema

Predecir popularidad de publicación en una red social

1400



1312.3



$1400 < \textit{Umbral}$

Acercamiento inicial del problema

Utilizando herramientas de colaboración que permitirán una integración del proyecto final, logrando evaluar en todo momento la ruta crítica; se mencionan a continuación:

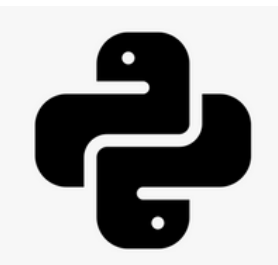


GitHub

<https://github.com/A01795433-AnaAristizabal/MLOPSGrupo3>



Project template para Python



Python



VS Code



Jupyter Notebooks



Data Version Control



Videos

https://drive.google.com/drive/folders/1XgbRPmMgcQyO6gszxAiOtmi5G8fMY0_6?usp=sharing

Actividades por Rol

Durante la aplicación de la metodología MLOps (Machine Learning Operations) para la elaboración del ejercicio, cada integrante del equipo ha asumido un rol con un conjunto específico de actividades para el desarrollo e implementación de los modelos de machine learning que luego serán llevados a producción.



Kevin Balderas

Data Analyst



Ana Aristizábal

Data Scientist



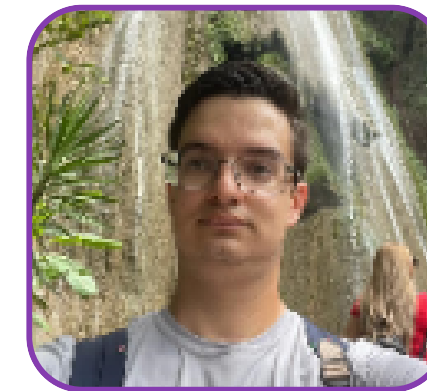
Manuel Acevedo

ML Engineer



Alan Arriaga

Software Engineer



Daniel Amezola

MLOps Engineer

Kevin Balderas, analista de datos, ha proporcionado la base de datos que ha impulsado el modelo de machine learning (ML). Su trabajo de preparación y limpieza de datos ha asegurado que la Data Scientists Ana comenzara la construcción de los modelos de ML sobre datos fiables y de calidad.

La Data Scientists Ana, ha creado y experimentado con los algunos modelos (son la base del ciclo de vida de machine learning en MLOps) Su trabajo está alineado con las fases de entrenamiento y evaluación dentro del ciclo. en las siguientes fases del ejercicio trabajará en estrecha colaboración con el MLOps Engineer Daniel para trasladar los modelos entrenados a producción de forma eficiente y repetible.

Manuel, ML Engineer se ha ocupado de convertir el modelo desarrollado por la Data Scientists Ana en un sistema que para la siguientes fases se espera que se más robusto, reproducible y escalable, automatizando pipelines y en la integración de los modelos dentro del flujo de producción

Finalmente, Alan Software Engineer y Daniel MLOps Engineer asegurarán que los modelos se integren en sistemas de software y que puedan ser desplegados de manera segura y escalable

Análisis del Problema

Características del problema

Se trata de un conjunto de datos que describe la popularidad de artículos publicados por Mashable durante un período de dos años.

El objetivo principal de este conjunto de datos es predecir la cantidad de veces que se compartirá un artículo en las redes sociales (shares), lo que lo convierte en un recurso útil para tareas relacionadas con la influencia de las redes sociales, el rendimiento del contenido y el modelado predictivo.

Puntos clave:

- Período: dos años.
- Variable objetivo: la cantidad de veces que se compartió un artículo en las redes sociales (popularidad).
- Número de registros: 39 644 artículos.
- Número de características: 61 atributos.











Características

- Valores numéricos -Valores booleanos - Valores categóricos

Objetivo:

- La variable objetivo es la cantidad de veces que se compartió el artículo, que se puede usar como una tarea de regresión o clasificación según cómo se plantee el problema.

MLCanvas

<div>Decisions</div> <div></div> <div><p>El punto clave que se debe tomar es qué tan bien se puede predecir la cantidad de "shares" en redes sociales utilizando diversas características de los artículos.</p><p>Comprender qué características impulsan la popularidad de los artículos ayudará a los equipos editoriales a priorizar la creación de contenido, optimizar el SEO y mejorar las estrategias de participación de la audiencia.</p></div>	<div>ML Task</div> <div></div> <div><p>Tarea: se trata de una tarea de regresión (predicción sobre la cantidad de veces que se comparte un artículo) o puede tratarse como una tarea de clasificación (por ejemplo, popularidad baja, media o alta).</p><p>Objetivo: crear un modelo predictivo que calcule la cantidad de veces que se comparte un artículo en función de sus atributos.</p></div>	<div>Value Propositions</div> <div></div> <div><p>Para el negocio: Predecir la popularidad del contenido podría mejorar la estrategia de contenido, lo que generaría más tráfico y mejoraría los ingresos por publicidad.</p><p>Identificando de manera eficiente qué características (por ejemplo, tema, día de la semana) impulsan la participación permite tomar decisiones editoriales optimizadas.</p></div>	<div>Data Sources</div> <div></div> <div><p>Conjunto de datos: El conjunto de datos de noticias en línea consta de artículos publicados por Mashable y contiene 61 características como: Cantidad de imágenes, videos, recuento de palabras. Características basadas en procesamiento del lenguaje natural, como polaridad de sentimientos. Características basadas en el tiempo, como el día de la semana. Tipo de datos: Datos estructurados en formato CSV.</p></div>	<div>Collecting Data</div> <div></div> <div><p>Datos existentes: conjunto de datos recopilados de Mashable que abarca dos años de publicación de artículos y sus características correspondientes y el rendimiento en las redes sociales.</p><p>Recopilación de datos futuros: canales automatizados para recopilar datos similares de otras fuentes o continuar extrayendo datos de Mashable para obtener nuevos artículos.</p></div>
<div>Making Predictions</div> <div></div> <div><p>Resultado: el modelo genera una cantidad prevista de veces que se comparte cada artículo en las redes sociales.</p><p>Aplicación en el mundo real: las predicciones podrían ayudar a optimizar la programación de contenido, identificar qué temas funcionan mejor en las redes sociales o marcar artículos con probabilidades de volverse virales.</p></div>	<div>Offline Evaluation</div> <div></div> <div><p>Se realice una evaluación sin conexión mediante cross-validation y un conjunto de validación de reserva para medir el rendimiento.</p><p>Métricas: Para la regresión: (MSE), R cuadrado.</p><p>Herramientas: Usando bibliotecas de Python como scikit-learn para realizar evaluaciones sin conexión y comparar el rendimiento del modelo.</p></div>	<div>Para los usuarios:</div> <div>Los editores y creadores de contenido pueden usar estos conocimientos para priorizar la creación de contenido que tenga éxito entre los lectores y se comparta bien en las plataformas sociales.</div>	<div>Features</div> <div></div> <div><p>Tipos de características: Características de contenido: cantidad de palabras, cantidad de imágenes, cantidad de enlaces, polaridad de sentimientos.</p><p>Características contextuales: día de publicación, canal (negocios, tecnología, estilo de vida).</p><p>Interacción: cantidad de veces que se comparte, popularidad del tema.</p></div>	<div>Building Models</div> <div></div> <div><p>Selección de modelos:</p><p>Regresión: Regresión Lineal, Random Forest, XGBoost para predecir la cantidad de shares.</p><p>Entrenamiento de modelos: Dividiendo el conjunto de datos en conjuntos de entrenamiento y prueba.</p><p>Ajuste los hiperparámetros utilizando técnicas como GridSearchCV.</p><p>Evaluar los modelos utilizando métricas como MSE para regresión.</p><p>Herramientas: ubibliotecas de Python scikit-learn, XGBoost.</p></div>
<div>Live Evaluation and Monitoring</div> <div></div> <div><p>Monitoreo en tiempo real: implementando el modelo para realizar predicciones en tiempo real sobre nuevos artículos.</p><p>Métricas clave: monitoreando la precisión de las predicciones en curso, la latencia de las predicciones y la coherencia entre las acciones previstas y reales en producción.</p><p>Ciclo de retroalimentación: volviendo a entrenar el modelo periódicamente para tener en cuenta las tendencias cambiantes en el comportamiento de los usuarios y la popularidad de los artículos.</p></div>				

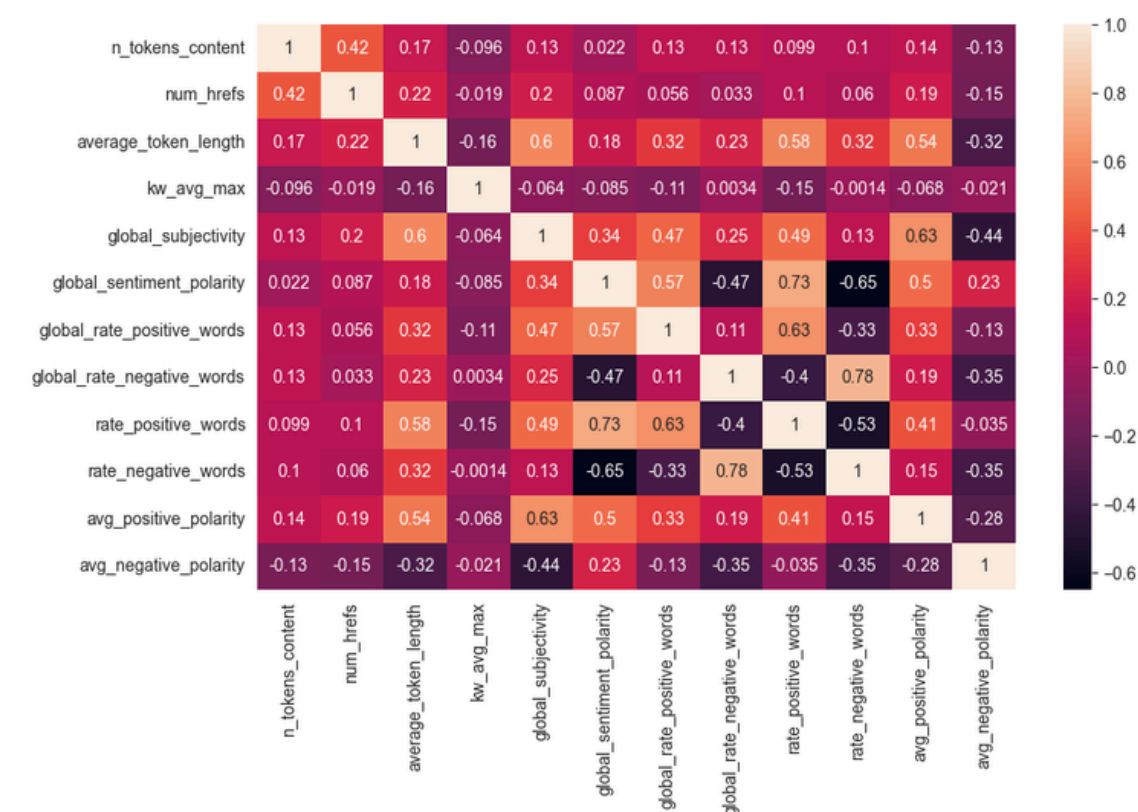
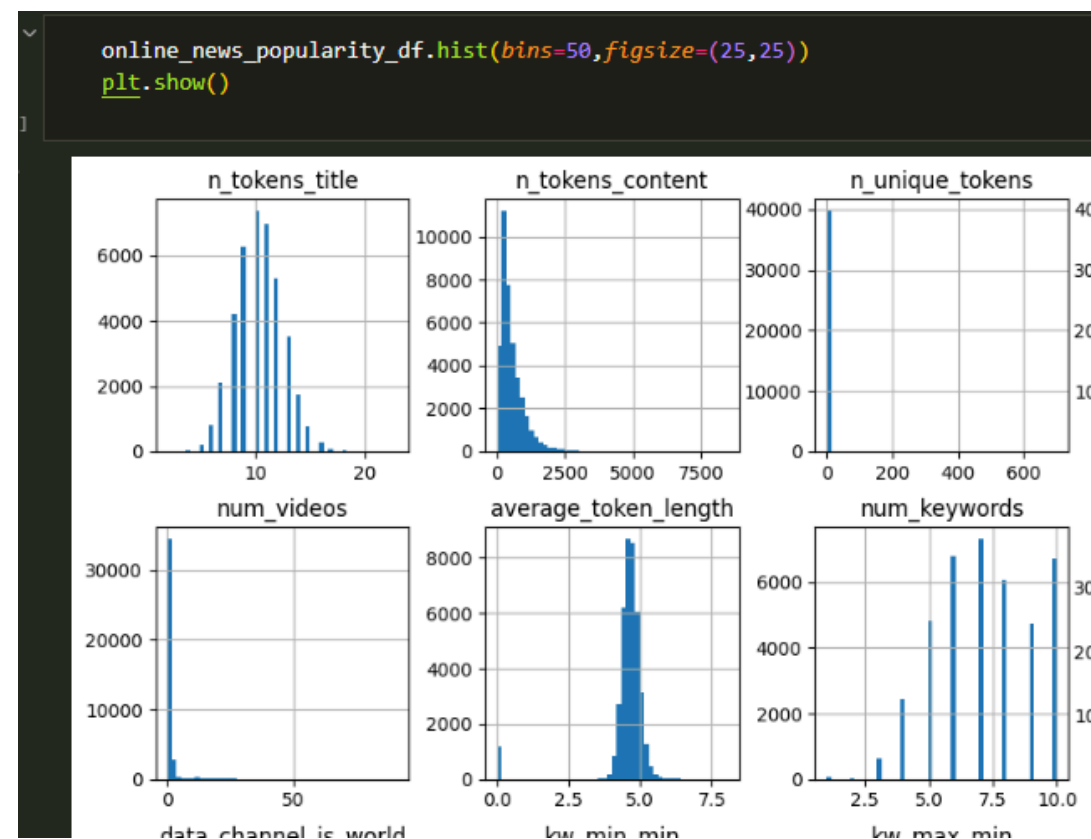
Métodos y técnicas a Utilizar

Qué método es el más adecuado y Porqué



Como la tarea principal es el aprendizaje supervisado, se trata de un problema de regresión, en el que el objetivo, es predecir el número de “shares” de los artículos, aplicamos lo siguiente:

1 Análisis exploratorio EDA



Métodos y técnicas a Utilizar



Como la tarea principal es el aprendizaje supervisado, se trata de un problema de regresión, en el que el objetivo, es predecir el número de “shares” de los artículos, aplicamos lo siguiente:

2 Ingeniería de características

```
##Preprocessing and feature engineering

def transformaciones_1(data):
    print("*****Transformaciones*****")
    cols_names_numeric = columns_numeric()
    variables_a_transformar = cols_names_numeric[:11]
    n = len(variables_a_transformar)
    misdatos = data

    sbn.set(rc={'figure.figsize':(17,12)})
    fig, axes = plt.subplots(5, n)

    for k in range(0,n):
        # Datos originales -----
        plt.subplot(5,n,k+1+(n*0))

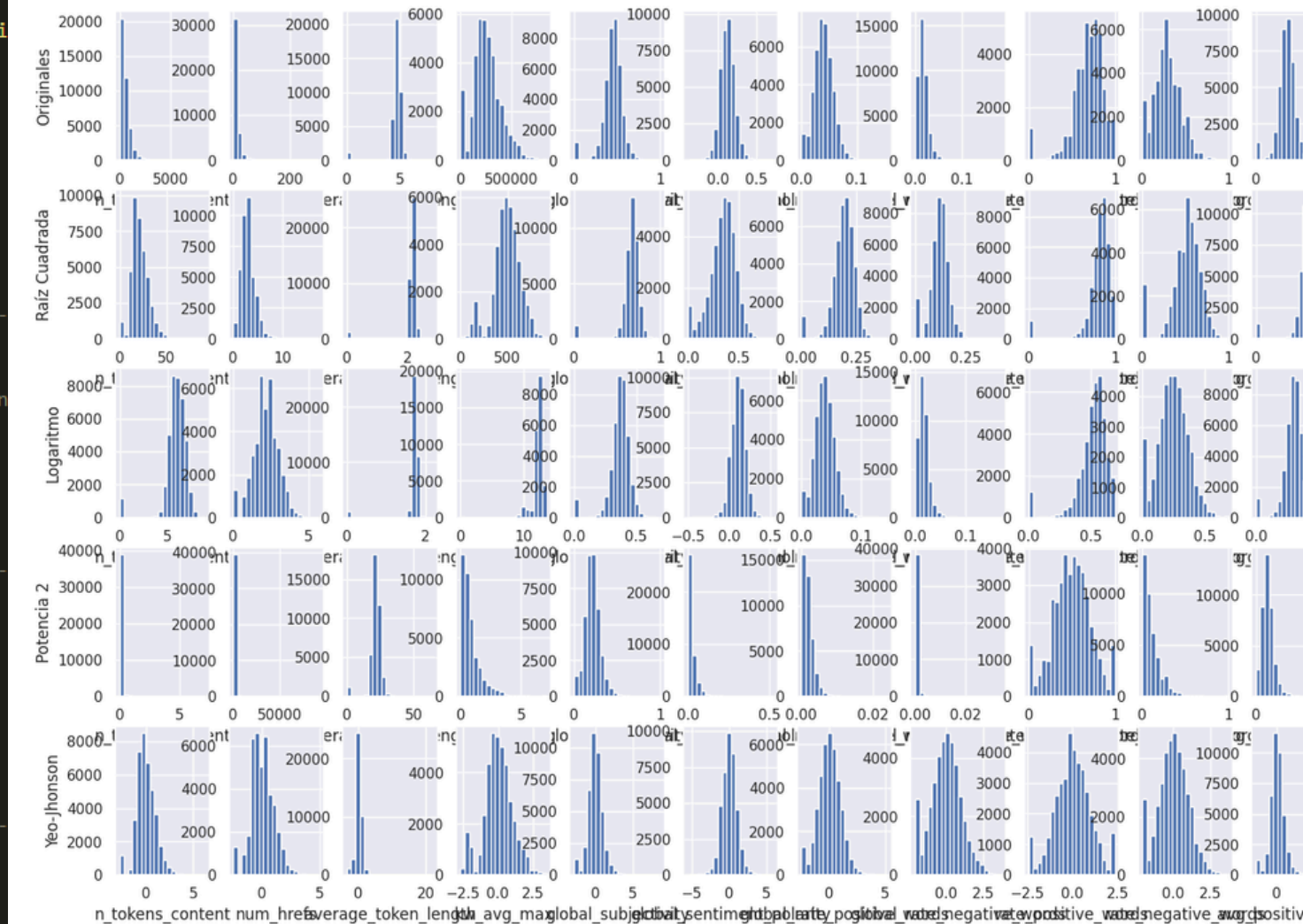
        Transf0 = misdatos[variables_a_transformar[k]] # En esta línea
        plt.hist(Transf0,bins=20) # En esta línea agrega el comando
        plt.xlabel(variables_a_transformar[k])
        if k==0:
            plt.ylabel('Originales')

        # Datos transformados con raíz cuadrada -----
        plt.subplot(5,n,k+1+(n*1))

        Transf1 = np.sqrt(misdatos[variables_a_transformar[k]])
        plt.hist(Transf1,bins=20) # En esta línea
        plt.xlabel(variables_a_transformar[k])
        if k==0:
            plt.ylabel('Raíz Cuadrada')

        # Datos transformados con logaritmo natural -----
        plt.subplot(5,n,k+1+(n*2))

        Transf2 = np.log1p(misdatos[variables_a_transformar[k]])
        plt.hist(Transf2,bins=20) # En esta línea
        plt.xlabel(variables_a_transformar[k])
        if k==0:
            plt.ylabel('Logaritmo')
```



Métodos y técnicas a Utilizar



Como la tarea principal es el aprendizaje supervisado, se trata de un problema de regresión, en el que el objetivo, es predecir el número de “shares” de los artículos, aplicamos lo siguiente:

3 Selección de modelos

DecisionTreeRegressor

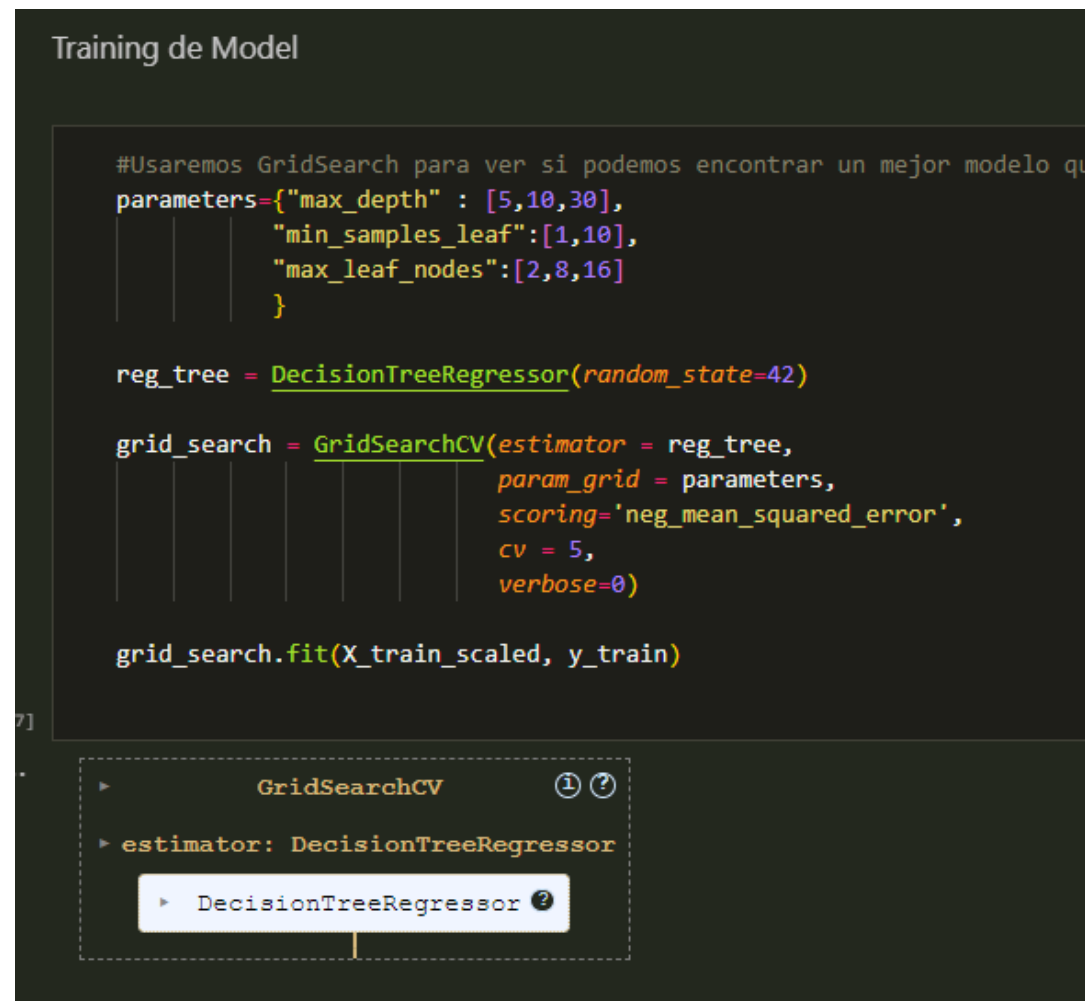
```
Training de Model

#Usaremos GridSearch para ver si podemos encontrar un mejor modelo que
parameters={"max_depth" : [5,10,30],
            "min_samples_leaf": [1,10],
            "max_leaf_nodes": [2,8,16]
            }

reg_tree = DecisionTreeRegressor(random_state=42)

grid_search = GridSearchCV(estimator = reg_tree,
                           param_grid = parameters,
                           scoring='neg_mean_squared_error',
                           cv = 5,
                           verbose=0)

grid_search.fit(X_train_scaled, y_train)
```



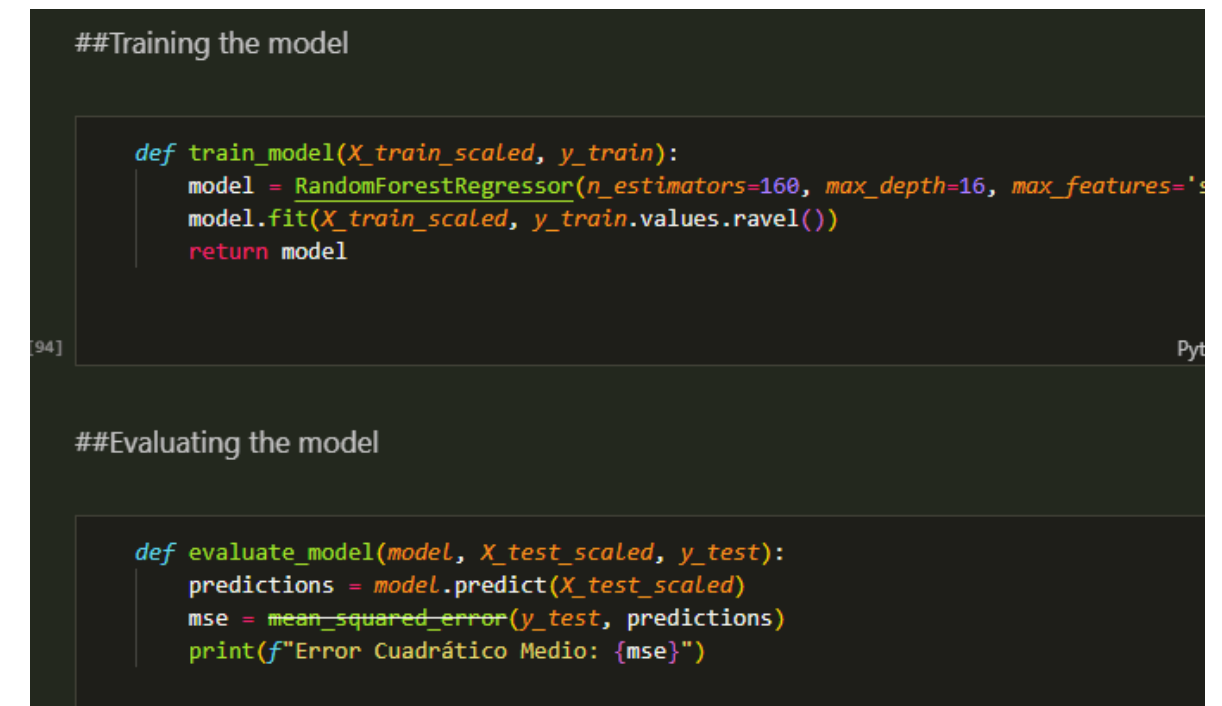
RandomForestRegressor

```
##Training the model

def train_model(X_train_scaled, y_train):
    model = RandomForestRegressor(n_estimators=160, max_depth=16, max_features='sqrt')
    model.fit(X_train_scaled, y_train.values.ravel())
    return model

##Evaluating the model

def evaluate_model(model, X_test_scaled, y_test):
    predictions = model.predict(X_test_scaled)
    mse = mean_squared_error(y_test, predictions)
    print(f"Error Cuadrático Medio: {mse}")
```



Resultados

Areas a mejorar en el análisis del caso

Después de aplicar la metodología ML, evaluamos los modelos mediante la métricas de: Error Cuadrático Medio (MSE)

```
##Evaluating the model
```

```
def evaluate_model(model, X_test_scaled, y_test):  
    predictions = model.predict(X_test_scaled)  
    mse = mean_squared_error(y_test, predictions)  
    print(f"Error Cuadrático Medio: {mse}")
```

Error Cuadrático Medio (MSE)

```
tree_Reg = RandomForestRegressor()  
tree_Reg.fit(X_train_scaled, y_train)  
  
preds = tree_Reg.predict(X_train_scaled)  
mse = mean_squared_error(y_train, preds)  
print(f"Error Cuadrático Medio: {mse}")
```

16]

```
.. Error Cuadrático Medio: 0.22702768615784408
```

```
preds = tree_Reg.predict(X_test_scaled)  
mse = mean_squared_error(y_test, preds)  
print(f"Error Cuadrático Medio: {mse}")
```

17]

```
.. Error Cuadrático Medio: 1.6618155539436172
```

DVC



Para el versionamiento del dataset, habilitamos el uso de DVC el cual se configuro para utilizar AWS S3 como repositorio de información.

Politica IAM

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Resource": "arn:aws:s3:::test-dbt-01aws"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": "arn:aws:s3:::test-dbt-01aws/*"
    }
  ]
}
```

Accesos

Configurar accesos

Les comparto por google drive estas credenciales.

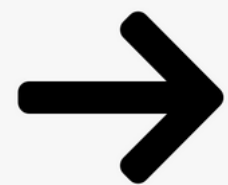
```
dvc remote modify --local storage \
                    access_key_id 'mysecret'
```

```
dvc remote modify --local storage \
                    secret_access_key 'mysecret'
```

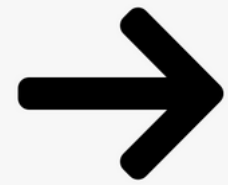

Conclusiones

Áreas a mejorar en el análisis del caso

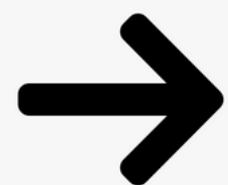
Estrategias empleadas en el análisis y solución del caso



La tarea de predecir la cantidad de veces que se comparte un artículo en las redes sociales requirió una comprensión profunda del conjunto de datos, sus características y los objetivos. Identificamos con éxito este problema como un problema de regresión supervisada.



Implementación y monitoreo de modelos: iniciar con la implementación y el monitoreo de modelos para garantizar que sigan siendo útiles en entornos dinámicos. Reconocemos que desarrollar una sólida cadena de procesos para mecanismos de monitoreo y retroalimentación (MLOps) es fundamental y podría mejorarse en proyectos futuros.



Proceso de implementación de Pipeline: trabajar en la incorporación de los proyectos futuros un enfoque más integral en la implementación de modelos y el monitoreo de la desviación de datos y el deterioro de los modelos para garantizar el éxito a largo plazo.