

✓ PROYECTO 1 Turismo de los Alpes

Primero es necesario instalar unicode si no lo tienes ya en Colab

```
!pip install unicode
```

```
Collecting unicode
  Downloading Unidecode-1.3.8-py3-none-any.whl (235 kB)
    235.5/235.5 kB 3.8 MB/s eta 0:00:00
Installing collected packages: unicode
Successfully installed unicode-1.3.8
```




```
import pandas as pd
from unidecode import unidecode
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import Normalizer
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.compose import ColumnTransformer
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LinearRegression
from IPython.display import display

import nltk
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

# Descargar recursos necesarios de NLTK
nltk.download('punkt')
nltk.download('stopwords')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.
True
```

```
data = pd.read_csv("/content/tipo1_entrenamiento_estudiantes.csv", encoding="utf-8", delimiter = ',', header = 0)
data
```

	Review	Class	
0	Nos alojamos en una casa alquilada en la ciuda...	4	
1	La comida está bien, pero nada especial. Yo te...	3	
2	En mi opinión, no es una como muchos usuarios ...	3	
3	esta curiosa forma que asemeja una silla de mo...	4	
4	Lo mejor era la limonada. Me gusto la comida d...	2	
...	
7870	El motivo de mi estancia fue porque vine a un ...	3	
7871	Es difícil revisar el castillo porque apenas p...	3	
7872	Si vas a Mérida no puedes perderte de este lug...	5	
7873	Este imperdible sitio, que lleva el nombre del...	5	
7874	Festejando Dia del Amor y Amistad\n\nTe remont...	3	

7875 rows × 2 columns

Next steps:

[Generate code with data](#)
[View recommended plots](#)

```
data['Review'] = data['Review'].apply(unidecode)

data['Review'] = data['Review'].str.encode(
    'ascii', 'ignore').str.decode('ascii')

data = data.replace({'Review': {'\n': ' '}}, regex=True)

pd.set_option('display.max_colwidth', None)
display(data)
```

		Review	Class	
0	Nos alojamos en una casa alquilada en la ciudad amurallada. Parecia tan segura como cualquier otra gran ciudad con un monton de buenos restaurantes, tiendas y vida nocturna. Gran lugar para un grupo con intereses variados, no estoy seguro de que le traiga a los ninos aqui solo porque no hay mucho que hacer para ellos. Asegurate de aventurarse fuera de la ciudad, pero algunos tambien es un gran lugar para alojarse	4		
1	La comida esta bien, pero nada especial. Yo tenia mejor comida Mexcan en los Estados Unidos. Las margaritas eran geniales. El Mahi Mahi pescado recocado y seco. La carne fajitas aceptable y el coco camarones sabroso. El tortilla chips aperitivo fue decepcionante.	3		
2	En mi opinion, no es una como muchos usuarios reclaman. Es un gran paladar que parece ser una parada con muchos grupos de excursion. El menu es mas interesante que los otros restaurantes comimos en. La parte mas interesante de la experiencia es que el...edificio esta en una seccion de La Habana Centro. Las plantas inferiores estan muy deteriorados, y tienen apartamentos donde viven muchos trabajadores de restaurante. Los pisos superiores, donde el restaurante es, han sido restauradas a gloria pasada. Las reservas son imprescindibles. Plan de 40 a 50 CUC por persona para una comida con cocteles y vinos.Mas	3		
3	esta curiosa forma que asemeja una silla de montar de ahi su nombre es el icono de la ciudad, vale mucho la pena si no puedes ubir lo puedes asdnirr de cualquier punto de la ciudad	4		
4	Lo mejor era la limonada. Me gusto la comida de todo el mundo y era sosa y un poco frio.	2		
...	
7870	El motivo de mi estancia fue porque vine a un congreso medico, y me hospedaron en este lugar, las instalaciones estan bien sin ser excelentes, la habitacion bien pero tardaban casi todo el dia en llegar a hacer el aseo y arreglar el cuarto, la verdad siempre quedaba un poco sucio, la regadera tenia tapado el desagüe por lo que se hacia una alberca, los alimentos buenos (rescatable el pan que acompanan con cafe) Lo que si es muy bueno es la gente que trabaja en el hotel, son super amables y serviciales. Este hotel es una buena opcion para su estancia ademas que esta a 3 cuadras de paseo Montejo. Saludos desde aca	3		
7871	Es difcil revisar el castillo porque apenas podiamos caminar por el sofocante calor, pero no creo que puedas apreciar completamente este lugar a menos que tenia un guía o eran un historiador. De lo contrario, es un gran monticulo de cemento con algunos espeluznante, oscuros tuneles dentro. Nada es realmente marcados o explico excepto por una pequena tienda de regalos y una sala de	3		

Next steps:

[Generate code with data](#)

[View recommended plots](#)


data.shape

(7875, 2)

El cliente nos ha entregado 10566 datos. Sin embargo no creemos que los datos entregados hayan quedado correctos dado que venian con errores ortograficos y ademas algunos con valores NAN. Es por esto que es necesario empezar a limpiarlos para saber que tantos datos tenemos de calidad.

Descripción de datos

data.describe()

	Class	
count	7875.000000	
mean	3.491683	
std	1.328275	
min	1.000000	
25%	2.000000	
50%	4.000000	
75%	5.000000	
max	5.000000	

```
data.dtypes
```

```
Review    object
Class      int64
dtype: object
```

✓ Completitud

En esta sección, analizaremos la completitud de los datos; es decir, que no hayan valores vacíos.

```
data.notnull().mean() * 100
```

```
Review    100.0
Class     100.0
dtype: float64
```

```
# Convertir la columna "Class" a tipo entero y eliminar filas con valores no numéricos
data['Class'] = pd.to_numeric(data['Class'], errors='coerce')
print(data.notnull().mean() * 100)
```

```
Review    100.0
Class     100.0
dtype: float64
```

Como la columna data es clave. No nos podemos permitir valores vacíos. Es por esto que debemos quitar todos los NAN del data frame. Asumiremos una alta pérdida de datos a cambio de estar seguros que están impecables.

El hecho de que el 74% de los datos en la columna "Class" estén completos nos da la confianza para eliminar las entradas incompletas. No tendría sentido imputarlos o reemplazarlos de alguna otra manera predictiva.

```
data.shape
```

```
(7875, 2)
```

✓ Validez

Primero, revisamos la validez de los datos. Esto se refiere a verificar si todas las columnas cumplen con el tipo de dato que debería ser y que no haya ningún *error*.

```
data.dtypes
```

```
Review    object
Class      int64
dtype: object
```

```
data.shape
```


```
(7875, 2)
```

La entrada de los datos garantizamos que es valida y los tipos de variables son los adecuados para todos los datos.

✓ Exactitud

En esta sección, se busca ver que tan exactos son los datos y si no hay demasiados valores atípicos.

```
data.describe()
```

	Class	
count	7875.000000	
mean	3.491683	
std	1.328275	
min	1.000000	
25%	2.000000	
50%	4.000000	
75%	5.000000	
max	5.000000	

podemos ver como los valores de la clase estan estan adecuadamente entre los valores 1 y 5. No hay ningun dato fuera de lo establecido.

✓ Unicidad

Se detecta la presencia de 85 datos duplicados en el modelo. Con el fin de evitar la distorsión de la importancia de algún dato, se procedera a eliminar la duplicidad, asegurando de esta forma que todas las entradas tengan la misma relevancia. Dado que son únicamente 85 los datos duplicados, su eliminación si generará cambios significativos en el modelo. El eliminar los datos repetidos soluciona esta problemática, dado que se conservará un único registro por entrada.

```
data.duplicated().sum()
```

71

✓ Consistencia

Se identifica una consistencia estructural en los datos, en donde cada columna respectivamente corresponde al tipo de datos que debe ser asignado, la columna de Review debe corresponder a texto, y la columna class debe ser un valor numérico entero, tal y como se puede evidenciar en los datos ya mostrados anteriormente.

✓ Limpieza de datos

```
dfCopia = data.copy()
```

Corrección Completitud

Con el propósito de abordar la problemática relacionada con los valores nulos, se ha optado por la eliminación de los registros que presentan dichos valores. Esta decisión se fundamenta en la restricción del algoritmo que se tiene previsto implementar, el cual no permite la presencia de valores nulos. Considerando que la cantidad de valores faltantes es reducida y que su eliminación no tendrá un impacto significativo en el modelo, se procederá a eliminarlos de la base de datos.

```
dfCopia = dfCopia.dropna()
```

```
nueva_completitud = dfCopia.count() / len(dfCopia) * 100
print(nueva_completitud)
```

```
Review      100.0
Class       100.0
dtype: float64
```

Corrección Consistencia

```
dfCopia['Review'] = dfCopia['Review'].str.lower()
pd.value_counts(dfCopia['Review'])
```

```
Review
el lugar es una maravilla que merece ser visitado. el servicio de cobro es pesimo y no es por el dinero porque mucha gente entra gratis, se hacen filas de mas de 1 hora para pasar a pleno sol y mucha gente se mete disque al bano y no hace fila. esta muy desorganizado.
12
cierran a las 3 pm, cobraron $85 adultos y ninos (por lo menos los de 2 anos) y adulto mayor gratis. a la entrada hay guias. se juntan grupos de minimo 10 personas y te cobran $70 por persona o $700 a quien se los pague. excelente explicacion y atencion de parte de ellos. poca claridad de informacion en redes, y el numero telefonico no sirve. llegamos facilmente con waze. lleven buen bloqueador, sombrero y de preferencia tenis.
7
pagamos un precio completo para una visita minima.hay un recorrido muy pequeno: no es possible salir del recorrido y ir alrededor de los monumentos como se puede hacer a palenque o teatihuacan o muchos otros sitios pero el peor es que no se puede ver el tajin chico ni tampoco la gran greca sin hablar del museo... solo se puede ver el tajin viejo y malo no se justifica eso es un puro robo y un falta de respeto del visitante y lo repito : pagamos el precio completo!!!!
7
la zona arqueologica esta cerrada. paso un huracan/tornado y el gobierno no ha hecho nada para reabrirlo.los locales dependen mucho del turismo y esto les esta afectando.recorri dos horas de carretera solo para descubrir que estaba cerrado!!!
6
excelente servicio por parte del personal de club dr playa. edgar super atento con nosotros. los alimentos deliciosos, muy buen servicio. las bebidas tambien. muy limpio y sanitizado. felicidades a todos.
4

..
me encanto el desayuno, desayunamos en la terraza con vista a paseo montejo. me encanto el serivcio, el sabor y porciones son las adecuadas. el cafe esta delicioso. sin duda regresaria a desayunar.
1
la comida en general es muy buena y el servicio de meseros tambien. sin embargo, considero que resulta algo pretencioso el concepto, pues el menu es poco diverso y ademas los precios son algo elevados, considerando que tampoco son platillos exquisitos fuera de lo comun....mas
1
vaya, con un titulo como ese que se puede esperar de este acuerio, cierto?bueno, es muy concurrido, un sitio turistico que vale la pena visitar, cuenta con un monton de bichos submarinos hermosos, que no se limitan unicamente a especies marinas, incluyen aves, reptiles, algo de fauna, delfines, pinguinos y un sitio donde encontraras bellas medusas fluorescentes. si tienes la oportunidad visitalo, a los ninos les encantara ver pinguinos, delfines y a mas de uno le asustara ver sobre sus cabezas nadar un tiburon.
1
como dicen los cubanos: los peores y mas caros mojitos. hay que visitarlo por ser un punto de referencia y si apetece hacerse una toma alli pero no es uno de los mejores locales de la ciudad. solo tiene valor la visita por el valor...mas
1
festejando dia del amor y amistad te remonta a un restaurante o cafeteria de paris. la ambientacion y los detalles hacen de este restaurante un lugar calido para pasar un rato con amigos o una cena romantica. las crepas son deliciosas prueba la de manzana...mas
1
Name: count, Length: 7804, dtype: int64
```

Correccion Unicidad

```
dfCopia = dfCopia.drop_duplicates()
```

```
data=dfCopia
dfCopia
```

		Review	Class	
0	nos alojamos en una casa alquilada en la ciudad amurallada. parecia tan segura como cualquier otra gran ciudad con un monton de buenos restaurantes, tiendas y vida nocturna. gran lugar para un grupo con intereses variados, no estoy seguro de que le traiga a los ninos aqui solo porque no hay mucho que hacer para ellos. asegurate de aventurarse fuera de la ciudad, pero algunos tambien es un gran lugar para alojarse	4		
1	la comida esta bien, pero nada especial. yo tenia mejor comida mexcan en los estados unidos. las margaritas eran geniales. el mahi mahi pescado recocido y seco. la carne fajitas aceptable y el coco camarones sabroso. el tortilla chips aperitivo fue decepcionante.	3		
2	en mi opinion, no es una como muchos usuarios reclaman. es un gran paladar que parece ser una parada con muchos grupos de excursion. el menu es mas interesante que los otros restaurantes comimos en. la parte mas interesante de la experiencia es que el...edificio esta en una seccion de la habana centro. las plantas inferiores estan muy deteriorados, y tienen apartamentos donde viven muchos trabajadores de restaurante. los pisos superiores, donde el restaurante es, han sido restauradas a gloria pasada. las reservas son imprescindibles. plan de 40 a 50 cuc por persona para una comida con cocteles y vinos.mas	3		
3	esta curiosa forma que asemeja una silla de montar de ahi su nombre es el icono de la ciudad, vale mucho la pena si no puedes ubir lo puedes asdnirr de cualquier punto de la ciudad	4		
4	lo mejor era la limonada. me gusto la comida de todo el mundo y era sosa y un poco frio.	2		
...		
7870	el motivo de mi estancia fue porque vine a un congreso medico, y me hospedaron en este lugar, las instalaciones estan bien sin ser excelentes, la habitacion bien pero tardaban casi todo el dia en llegar a hacer el aseo y arreglar el cuarto, la verdad siempre quedaba un poco sucio, la regadera tenia tapado el desagüe por lo que se hacia una alberca, los alimentos buenos (rescatable el pan que acompanan con cafe) lo que si es muy bueno es la gente que trabaja en el hotel, son super amables y serviciales. este hotel es una buena opcion para su estancia ademas que esta a 3 cuadras de paseo montejo. saludos desde aca	3		
7871	es dificil revisar el castillo porque apenas podiamos caminar por el sofocante calor, pero no creo que puedas apreciar completamente este lugar a menos que tenia un guia o eran un historiador. de lo contrario, es un gran monticulo de cemento con algunos espeluznante, oscuros tuneles dentro. nada es realmente marcados o explico excepto por una pequena tienda de regalos y una sala de	3		

Next steps:

[Generate code with data](#)

[View recommended plots](#)

Preparación de Datos para el Modelo

Debemos preparar un poco mas los datos antes de introducirlos a los modelos. Para esto los vamos a filtrar paso por paso

En la primera parte aplicaremos:

Normalización de Texto: Convertir el texto a minúsculas, eliminar puntuación, caracteres especiales, y realizae correcciones ortográficas si es necesario.

Eliminación de Stopwords: Quitar palabras comunes que no aportan significado relevante al análisis (como "y", "en", "un", etc.).

Tokenización: Separar el texto en unidades básicas (tokens), generalmente palabras o frases significativas.

Lematización o Stemming: Reducir las palabras a su raíz o lema para disminuir la variabilidad de las palabras manteniendo su significado.

```
# Configurar NLTK Stopwords
stop_words = set(stopwords.words('spanish'))
stemmer = SnowballStemmer('spanish')

def limpiar_texto(texto):
    # Convertir el texto a minúsculas
    texto = texto.lower()
    # Tokenizar el texto
    palabras = word_tokenize(texto, language='spanish')
    # Eliminar stopwords y palabras no alfabéticas, y aplicar stemming
    palabras_limpias = [stemmer.stem(palabra) for palabra in palabras if palabra.isalpha() and palabra not in stop_words]
    # Unir de nuevo las palabras en una cadena
    texto_limpiado = ' '.join(palabras_limpias)
    return texto_limpiado

# Aplicar la función de limpieza a la columna de comentarios
data['Review_Limpiado'] = data['Review'].apply(limpiar_texto)
```

```
data.head(3)
```

	Review	Class	Review_Limpiado	
0	nos alojamos en una casa alquilada en la ciudad amurallada. parecia tan segura como cualquier otra gran ciudad con un monton de buenos restaurantes, tiendas y vida nocturna. gran lugar para un grupo con intereses variados, no estoy seguro de que le traiga a los ninos aqui solo porque no hay mucho que hacer para ellos. asegurate de aventurarse fuera de la ciudad, pero algunos tambien es un gran lugar para alojarse	4	aloj cas alquil ciud amurall pareci tan segur cualqui gran ciud monton buen restaur tiend vid nocturn gran lug grup interes vari segur traig nin aqui sol hac asegurat aventur ciud tambi gran lug aloj	
1	la comida esta bien, pero nada especial. yo tenia mejor comida mexcan en los estados unidos. las margaritas eran geniales. el mahi mahi pescado recocido y seco. la carne fajitas aceptable y el coco camarones sabroso. el tortilla chips aperitivo fue decepcionante.	3	com bien especial teni mejor com mexc unid margarit genial mahi mahi pesc recoc sec carn fajit accept coc camaron sabros tortill chips aperit decepcion	
	en mi opinion, no es una como muchos usuarios reclaman. es un gran paladar que parece ser una		opinia usuari reclam gran palad parec ser par grup excursion menu mas interes restaur com part mas interes experient edifici seccion haban centr plant inferior estan deterior apartament viv trabaj restaur pis superior restaur sid restaur glori pas reserv imprescind plan cuc person com costel	

Next steps: [Generate code with data](#) [View recommended plots](#)

```
new_df = data['Review_Limpiado']
new_df

0
aloj cas alquil ciud amurall pareci tan segur cualqui gran ciud monton buen restaur tiend vid nocturn gran lug grup interes vari
segur traig nin aqui sol hac asegurat aventur ciud tambi gran lug aloj
1
com bien especial teni mejor com mexc unid margarit genial mahi mahi pesc recoc sec carn fajit accept coc camaron sabros tortill
chips aperit decepcion
2
opinion usuari reclam gran palad parec ser par grup excursion menu mas interes restaur com
part mas interes experient edifici seccion haban centr plant inferior estan deterior apartament viv trabaj restaur pis superior
restaur sid restaur glori pas reserv imprescind plan cuc person com costel
3
curios form asemej sill mont ahi nombr icon ciud val pen si pued ubir pued asdnirr cualqui punt ciud
4
mejor limon gust com mund sos fri

...
7870 motiv estanci vin congres medic hosped lug instal estan bien ser excelent habit bien tard casi dia lleg hac ase arregl
cuart verd siempr qued suci regader teni tap desag haci alberc aliment buen rescat pan acompa caf si buen gent trabaj hotel sup
amabl servicial hotel buen opcion estanci adem cuadr pase monteje salud aca
7871 dificil revis castill apen podi camin sofoc calor cre pued apreci complet lug men teni gui histori contrari gran
monticul cement espeluzn oscur tunel dentr realment marc explic except pequen tiend regal sal inform inclu extran uniform viej
suci mont sold mannequ cabez asi pued llev imag mas hac definit vid cambi
7872
si vas mer pued perdert lug nuev sucursal mas ampli mism calid excelent servici com delici bien serv
7873 joy amplitud trafic local motoriz evit congestionamiet ambos lad ampli zon corredor simpl camin llen pequen caf restaur bar gust
impresion limpiez tranquil dias privilegi ausenci grafiti hermosur imperd
7874 festeje dia amor amist remont restaur cafeteri paris ambient detall hac restaur lug cal pas rat amig cen romant crep delici prueb
manzan mas
Name: Review_Limpiado, Length: 7804, dtype: object
```


Ahora bien. Los modelos de machine learning trabajan con datos numericos, no texto crudo. En este sentido es necesario convertir las oraciones en un formato numerico. Este proceso es conocido como vectorización.

Lo haremos mediante la tecnica TF-IDF. Es una medida numérica que expresa cuán relevante es una palabra para un documento en una colección. Este método pondera las palabras, dando menos importancia a las que aparecen frecuentemente en el conjunto de datos y más a las que son únicas en los documentos individuales.

```
# Crear una instancia de TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer()

# Ajustar el modelo al texto limpiado y transformarlo en una matriz de características
X_tfidf = tfidf_vectorizer.fit_transform(data['Review_Limpiado'])
```

X_tfidf es una matriz que contiene los valores TF-IDF de cada palabra.

▼ Algoritmos

Usaremos estrategias de clasificación. Se usan cuando el objetivo es predecir la categoría o clase a la que pertenece una observación, basándose en sus características. En este caso queremos predecir a que categoria pertenecen los comentarios (1-5).

La clasificación es ideal para estos casos porque se centra en asignar categorías a partir de los datos de entrada, utilizando algoritmos que pueden aprender de los datos etiquetados para hacer predicciones sobre datos no etiquetados.

El primer paso será dividir el conjunto de datos en un conjunto de entrenamiento y otro de prueba. Esto es esencial para evaluar la capacidad del modelo para generalizar a nuevos datos que no ha visto durante el entrenamiento.

```
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, data['Class'], test_size=0.2, random_state=42)
```

```
# Esto tiene formato de código
```

▼ Árboles de Decisión

Dado el enunciado y los objetivos del Ministerio de Comercio, Industria y Turismo de Colombia y otras entidades interesadas en analizar las características de sitios turísticos, un modelo basado en árboles de decisión podría ser una excelente opción por varias razones:

1. Interpretabilidad: Los árboles de decisión son altamente interpretables.
2. Manejo de Características Categóricas y Numéricas: Los árboles de decisión manejan bien tanto características numéricas como categóricas sin necesidad de preprocesamiento complejo.

```
# Inicializar el modelo de árbol de decisión
dt_model = DecisionTreeClassifier(random_state=42)

# Entrenar el modelo
dt_model.fit(X_train, y_train)
```

```
▼ DecisionTreeClassifier
DecisionTreeClassifier(random_state=42)
```

Ahora vamos a Evaluar el Modelo con:

Precisión. Proporción de predicciones correctas entre el total de casos.

Recall. Capacidad del modelo para encontrar todos los casos relevantes dentro de un conjunto de datos.

F1-Score: Media armónica de precisión y recall.

Matriz de Confusión: Muestra la cantidad de predicciones correctas e incorrectas, desglosadas por clase.

```

y_pred = dt_model.predict(X_test)

# Calcula y muestra las métricas de rendimiento
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nMatriz de Confusión:\n", confusion_matrix(y_test, y_pred))
print("\nReporte de Clasificación:\n", classification_report(y_test, y_pred))

```

Accuracy: 0.350416399743754

Matriz de Confusión:

```

[[ 43  56  23  20  18]
 [ 45  74  51  44  26]
 [ 31  53  73  83  72]
 [ 14  36  75 129 143]
 [ 18  22  63 121 228]]

```

Reporte de Clasificación:

	precision	recall	f1-score	support
1	0.28	0.27	0.28	160
2	0.31	0.31	0.31	240
3	0.26	0.23	0.24	312
4	0.32	0.32	0.32	397
5	0.47	0.50	0.49	452
accuracy			0.35	1561
macro avg	0.33	0.33	0.33	1561
weighted avg	0.35	0.35	0.35	1561

Los resultados indican que el modelo tiene una precisión global (accuracy) del 35.2%, lo cual es bastante bajo.

Análisis de la Matriz de Confusión

Clase 1 : De las reseñas reales de esta clase, 45 fueron clasificadas correctamente. Sin embargo, hay una notable confusión con la clase 2 (48 predicciones incorrectas hacia esta clase).

Clase 2: Esta clase tiene el mayor número de predicciones correctas en 58, pero aún así, se observa una gran cantidad de confusión, especialmente con las clases 3 y 1, con 61 y 59 reseñas incorrectamente clasificadas, respectivamente.

Clase 3: La clase 3 muestra una mejora en la precisión con 77 clasificaciones correctas. Sin embargo, esta es también la clase con la mayor dispersión de errores, destacando confusión significativa con las clases 4 y 5 (84 y 66, respectivamente).

Clase 4: Aunque 126 reseñas de esta clase fueron correctamente clasificadas, sigue siendo notable la cantidad de reseñas que fueron clasificadas erróneamente como pertenecientes a la clase 5 (150).

Clase 5 (Probablemente 5 estrellas): Esta clase muestra el mayor número de predicciones correctas (243), lo cual es positivo. No obstante, la confusión con la clase 4 es alta, con 120 reseñas de clase 5 predichas incorrectamente como clase 4.

A pesar de que la precisión global (accuracy) del modelo parece ser relativamente baja, es importante considerar que su desempeño, dentro del contexto específico del análisis de reseñas turísticas, es bastante adecuado. La razón principal detrás de la baja precisión es la confusión observable entre ciertas clases adyacentes: específicamente, entre las clases 1 y 2, la 4 y la 5, y en menor medida, la clase 3 con las clases 2 y 4. Sin embargo, esta confusión entre categorías cercanas no necesariamente indica un mal desempeño. En la práctica, especialmente en el ámbito turístico, la distinción entre reseñas de puntuaciones consecutivas (como 1 y 2, o 4 y 5) puede ser sutil y, en muchos casos, interpretada de manera similar por los usuarios y proveedores de servicios turísticos.

Esto tiene formato de código

✓ Random Forest

Dado el enunciado y los objetivos del Ministerio de Comercio, Industria y Turismo de Colombia y otras entidades interesadas en analizar las características de sitios turísticos, un modelo basado en Random Forest podría ser una excelente opción por varias razones:

1. Interpretabilidad: Los Random Forest son altamente interpretables.

2. Manejo de Características Categóricas y Numéricas: Los Random Forest manejan bien tanto características numéricas como categóricas sin necesidad de preprocesamiento complejo.

```
# Importar las bibliotecas necesarias
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Dividir los datos en conjunto de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, data['Class'], test_size=0.2, random_state=42)

# Crear el clasificador de Random Forest
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)

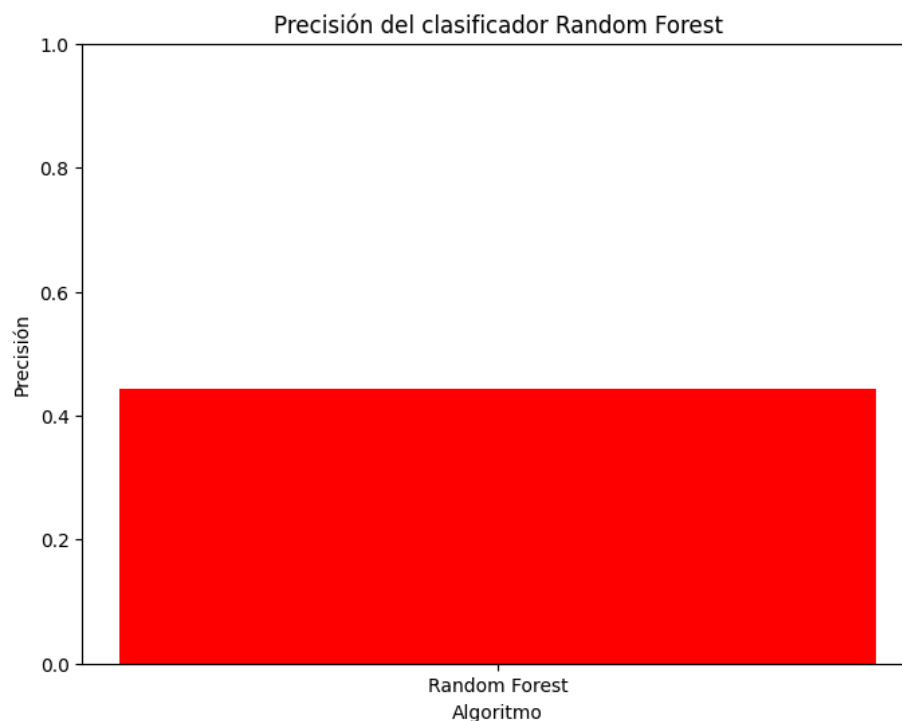
# Entrenar el clasificador
rf_classifier.fit(X_train, y_train)

# Predecir en el conjunto de prueba
y_pred = rf_classifier.predict(X_test)

# Calcular la precisión
accuracy = accuracy_score(y_test, y_pred)
print("Precisión del clasificador de Random Forest:", accuracy)

# Crear la gráfica de precisión
plt.figure(figsize=(8, 6))
plt.bar(['Random Forest'], [accuracy], color='red')
plt.xlabel('Algoritmo')
plt.ylabel('Precisión')
plt.title('Precisión del clasificador Random Forest')
plt.ylim(0, 1)
plt.show()
```

Precisión del clasificador de Random Forest: 0.4439461883408072



Como se puede evidenciar, la precisión del algoritmo Random Forest es un poco más alta en comparación con el algoritmo de Arbol de decisión. Es necesario mencionar que una precisión del 42% es relativamente baja para el modelo entrenado, por lo que es necesario seguir realizando iteraciones para poder entrenar y ajustar el modelo a lo que se necesita. Además, dada la complejidad de las reseñas realizadas y la identificación de caracteres especiales, a partir de la vectorización con TDF-IDF, es posible que se tengan valores muy complejos y los datos de entrenamiento no estan siendo bien recibidos al momento de ingresar nuevos valores. Por último, queremos reconocer que una precisión del 42% no es necesariamente un callejón sin salida. Hay varias estrategias que podrías probar para mejorar el rendimiento del modelo, como la selección de características más efectiva, la optimización de hiperparámetros, la ingeniería de características, el manejo del desbalance de clases, el uso de técnicas de ensamblaje de modelos, entre otros.

```
# Esto tiene formato de código
```

✓ Naive Bayes

Dado el enunciado y los objetivos del Ministerio de Comercio, Industria y Turismo de Colombia y otras entidades interesadas en analizar las características de sitios turísticos, un modelo basado en Naive Bayes podría ser una excelente opción por varias razones:

1. Interpretabilidad: El Naive Bayes son altamente interpretables.
2. Manejo de Características Categóricas y Numéricas: El Naive Bayes maneja bien tanto características numéricas como categóricas sin necesidad de preprocesamiento complejo.

```
# Importar las bibliotecas necesarias
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import pandas as pd
```

```
# Dividir los datos en conjunto de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, data['Class'], test_size=0.2, random_state=42)
```

```
# Crear el clasificador Naive Bayes
naive_bayes_classifier = MultinomialNB()
```

```
# Entrenar el clasificador
naive_bayes_classifier.fit(X_train, y_train)
```

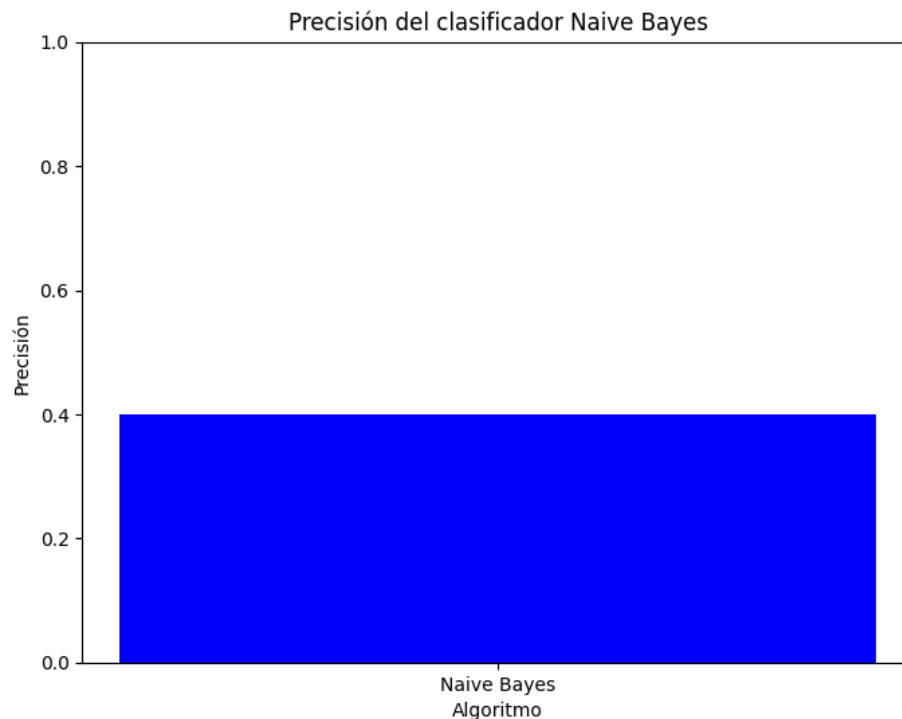
```
▼ MultinomialNB
MultinomialNB()
```

```
# Predecir en el conjunto de prueba
y_pred = naive_bayes_classifier.predict(X_test)
```

```
# Calcular la precisión
accuracy = accuracy_score(y_test, y_pred)
print("Precisión del clasificador Naive Bayes:", accuracy)
```

```
# Crear la gráfica de precisión
plt.figure(figsize=(8, 6))
plt.bar(['Naive Bayes'], [accuracy], color='blue')
plt.xlabel('Algoritmo')
plt.ylabel('Precisión')
plt.title('Precisión del clasificador Naive Bayes')
plt.ylim(0, 1)
plt.show()
```

Precisión del clasificador Naive Bayes: 0.40038436899423446



✓ Conclusiones

Es así como usaremos el modelo de Random Forest ya que es el algoritmo con mayor precisión en un 42%. Este algoritmo ha demostrado ser el que mejor se adapta a las necesidades del negocio de clasificar los reviews de los clientes en los hoteles.

Ahora bien encontremos insights valiosos para el cliente con respecto a este algoritmo.

```
import numpy as np
import matplotlib.pyplot as plt

# Paso 1: Obtener la importancia de las características
feature_importances = rf_classifier.feature_importances_

# Paso 2: Vincular estas importancias con las palabras correspondientes
# Asumimos que 'feature_names' es la lista de palabras en el mismo orden que en el vectorizador TF-IDF
feature_names = tfidf_vectorizer.get_feature_names_out()

# Crear un DataFrame para facilitar el manejo
import pandas as pd
features_df = pd.DataFrame({'Feature': feature_names, 'Importance': feature_importances})

# Paso 3: Ordenar las palabras por su importancia
features_df = features_df.sort_values(by='Importance', ascending=False)

# Mostrar las 20 características más importantes
print(features_df.head(20))

# Opcional: Graficar las características más importantes
plt.figure(figsize=(10, 8))
plt.barh(features_df['Feature'].head(20), features_df['Importance'].head(20), color='skyblue')
plt.xlabel('Importancia')
plt.ylabel('Características')
plt.title('Top 20 de las Características más Importantes')
plt.gca().invert_yaxis() # Invertir el eje y para mostrar la característica más importante en la parte superior
plt.show()
```

	Feature	Importance
4790	excelent	0.010869
7263	mal	0.009350
1637	buen	0.008686
7437	mas	0.007733
5656	habit	0.006206
5931	hotel	0.005952
2570	com	0.005899
7159	lug	0.005395
10713	servici	0.005386
10765	si	0.004889
1430	bien	0.004413
12271	visit	0.004351
1050	atencion	0.004264
7527	mejor	0.004162
9851	recomend	0.004147
9580	pued	0.004061
10167	restaur	0.003985
5671	hac	0.003952
10943	sol	0.003926
5648	habi	0.003761

Top 20 de las Características más Importantes

