import plotly.express as px

import pandas as pd

# Load the CSV file

df = pd.read\_csv('Tweets.csv')

# Create a new column called `count` and populate it with the number of tweets for each sentiment label

df['count'] = df['airline\_sentiment'].value\_counts()

# Create a bar chart of the overall sentiment distribution

fig = px.bar(df, x='airline\_sentiment', y='name', color='airline\_sentiment', color\_continuous\_scale='Viridis')

fig.update\_layout(title='Overall Sentiment Distribution', xaxis\_title='Sentiment Label', yaxis\_title='Number of Tweets')

fig.show()

# Create a pie chart of the overall sentiment distribution

fig = px.pie(df, values='count', names='airline\_sentiment', hole=0.5)

fig.update\_layout(title='Overall Sentiment Distribution')

fig.show()

# Create a word cloud of the most common words used in the tweets

words = df['text'].str.split().sum()

fig = px.wordcloud(words, max\_font\_size=50, max\_words=200)

fig.update\_layout(title='Word Cloud of Most Common Words')

fig.show()

# Create a map of the locations from which the tweets were sent

fig = px.scatter\_mapbox(df, lat='tweet\_latitude', lon='tweet\_longitude', color='airline\_sentiment', size\_max=15, zoom=3)

fig.update\_layout(mapbox\_style="open-street-map")

fig.update\_layout(title='Map of Tweet Locations')

fig.show()

# Create a scatter plot of the sentiment score vs. retweet count

fig = px.scatter(df, x='retweet\_count', y='airline\_sentiment', color='airline\_sentiment')

fig.update\_layout(title='Scatter Plot of Sentiment Score vs. Retweet Count', xaxis\_title='Retweet Count', yaxis\_title='Sentiment Score')

fig.show()

# Create a box plot of the sentiment score for each airline

fig = px.box(df, x='airline\_name', y='airline\_sentiment', color='airline\_name')

fig.update\_layout(title='Box Plot of Sentiment Score for Each Airline', yaxis\_title='Sentiment Score')

fig.show()

# Create a line chart of the average sentiment score over time

df\_sentiment\_over\_time = df.groupby('tweet\_created')['airline\_sentiment'].mean()

fig = px.line(df\_sentiment\_over\_time, x=df\_sentiment\_over\_time.index, y=df\_sentiment\_over\_time.values)

fig.update\_layout(title='Line Chart of Average Sentiment Score Over Time', xaxis\_title='Date', yaxis\_title='Sentiment Score')

fig.show()