ESSENTIALS OF DATA SCIENCE

Theory Activity No. 1

Name: Mahesh Dattatray Jadhav

Division: CS3

Batch: C33

Roll No.: 63

PRN: 202401040286

10 Problem Statements Using NumPy:

1. Find total tweets per sentiment category.

Solution:

```
sentiment_counts = df['airline_sentiment'].value_counts().to_dict()

sentiments = np.array(list(sentiment_counts.keys()))

counts = np.array(list(sentiment_counts.values()))
```

2. Find the proportion of each sentiment.

Solution:

```
total = counts.sum()

proportions = counts / total
```

3. Find tweets per airline.

Solution:

```
airline_counts = df['airline'].value_counts().to_dict()

airlines = np.array(list(airline_counts.keys()))

tweet_counts = np.array(list(airline_counts.values()))
```

4. Find negative sentiment distribution per airline.

Solution:

negatives = df[df['airline_sentiment'] == 'negative']

negative_counts = negatives['airline'].value_counts().reindex(airlines, fill_value=0).values

5. Find the percentage of negative tweets per airline.

Solution:

negative_percentage = negative_counts / tweet_counts * 100

6. Find the most common reason for negative sentiment.

Solution:

negative_reasons = negatives['negativereason'].dropna()

reason_counts = negative_reasons.value_counts()

top_reason = reason_counts.idxmax()

top_reason_count = reason_counts.max()

7. Find the airline with the highest positive sentiment ratio.

Solution:

positives = df[df['airline_sentiment'] == 'positive']

positive_counts = positives['airline'].value_counts().reindex(airlines, fill_value=0).values

positive_ratio = positive_counts / tweet_counts

best_airline_idx = np.argmax(positive_ratio)

best_airline = airlines[best_airline_idx]

8. Find the standard deviation of negative tweet counts per airline.

Solution:

std_dev_negatives = np.std(negative_counts)

9. Find the airline with the most consistent sentiment distribution (smallest range).

Solution:

pivot = df.pivot_table(index='airline', columns='airline_sentiment', aggfunc='size', fill_value=0)

sentiment_range = pivot.max(axis=1) - pivot.min(axis=1)

most_consistent_airline = sentiment_range.idxmin()

10. Find the trend of tweets over time (daily average tweet count).

Solution:

df['tweet_created'] = pd.to_datetime(df['tweet_created'])

df['date'] = df['tweet_created'].dt.date

daily_counts = df.groupby('date').size()

daily_avg = np.mean(daily_counts.values)

10 Problem Statements Using Pandas:

1. What is the overall sentiment distribution?

Solution:

sentiment_distribution = df['airline_sentiment'].value_counts()

print(sentiment_distribution)

2. Which airline received the most negative tweets?

Solution:

most_negative_airline = df[df['airline_sentiment'] == 'negative']['airline'].value_counts()

print(most_negative_airline)

3. What are the top 5 reasons for negative sentiment?

Solution:

top_negative_reasons = df['negativereason'].value_counts().head(5)

print(top_negative_reasons)

4. What is the sentiment breakdown for each airline?

Solution:

airline_sentiment_breakdown = df.groupby(['airline', 'airline_sentiment']).size().unstack().fillna(0)

print(airline_sentiment_breakdown)

5. Which day had the highest number of tweets?

Solution:

df['tweet_created'] = pd.to_datetime(df['tweet_created'])

```
df['date'] = df['tweet_created'].dt.date

most_active_day = df['date'].value_counts().idxmax()

tweet_count = df['date'].value_counts().max()

print(f"{most_active_day} had the most tweets: {tweet_count}")
```

6. What percentage of tweets for each airline are negative?

Solution:

```
airline_counts = df['airline'].value_counts()

negative_counts = df[df['airline_sentiment'] == 'negative']['airline'].value_counts()

negative_percentage = (negative_counts / airline_counts * 100).fillna(0).round(2)

print(negative_percentage)
```

7. What is the average number of tweets per user?

Solution:

```
avg_tweets_per_user = df['name'].value_counts().mean()

print(avg_tweets_per_user)
```

8. Which airline had the most positive feedback?

Solution:

```
most_positive_airline = df[df['airline_sentiment'] == 'positive']['airline'].value_counts()

print(most_positive_airline)
```

9. Which users posted the most tweets?

Solution:

```
top_users = df['name'].value_counts().head(5)

print(top_users)
```

10. How many unique negative reasons are there and how are they distributed?

Solution:

```
unique_reasons = df['negativereason'].nunique()

reason_distribution = df['negativereason'].value_counts()
```

```
print(f"Unique reasons: {unique_reasons}")

print(reason_distribution)
```