Activity_Course 3 TikTok project lab

February 1, 2025

1 TikTok Project

Course 3 - Go Beyond the Numbers: Translate Data into Insights

Your TikTok data team is still in the early stages of their latest project. So far, you've completed a project proposal and used Python to inspect and organize the TikTok dataset.

Orion Rainier, a Data Scientist at TikTok, is pleased with the work you have already completed and is requesting your assistance with some Exploratory Data Analysis (EDA) and data visualization. The management team asked to see a Python notebook showing data structuring and cleaning, as well as any matplotlib/seaborn visualizations plotted to help us understand the data. At the very least, include a graph comparing claim counts to opinion counts, as well as boxplots of the most important variables (like "video duration," "video like count," "video comment count," and "video view count") to check for outliers. Also, include a breakdown of "author ban status" counts.

Additionally, the management team has recently asked all EDA to include Tableau visualizations. Tableau visualizations are particularly helpful in status reports to the client and board members. For this data, create a Tableau dashboard showing a simple claims versus opinions count, as well as stacked bar charts of claims versus opinions for variables like video view counts, video like counts, video share counts, and video download counts. Make sure it is easy to understand to someone who isn't data savvy, and remember that the assistant director is a person with visual impairments.

You also notice a follow-up email from the Data Science Lead, Willow Jaffey. Willow suggests including an executive summary of your analysis to share with teammates.

A notebook was structured and prepared to help you in this project. Please complete the following questions.

2 Course 3 End-of-course project: Exploratory data analysis

In this activity, you will examine data provided and prepare it for analysis. You will also design a professional data visualization that tells a story, and will help data-driven decisions for business needs.

Please note that the Tableau visualization activity is optional, and will not affect your completion of the course. Completing the Tableau activity will help you practice planning out and plotting a data visualization based on a specific business need. The structure of this activity is designed to emulate the proposals you will likely be assigned in your career as a data professional. Completing this activity will help prepare you for those career moments.

The purpose of this project is to conduct exploratory data analysis on a provided data set. Your mission is to continue the investigation you began in C2 and perform further EDA on this data with the aim of learning more about the variables. Of particular interest is information related to what distinguishes claim videos from opinion videos.

The goal is to explore the dataset and create visualizations. *This activity has 4 parts:*

Part 1: Imports, links, and loading

Part 2: Data Exploration * Data cleaning

Part 3: Build visualizations

Part 4: Evaluate and share results

Be sure to complete this activity before moving on. The next course item will provide you with a completed exemplar to compare to your own work.

3 Visualize a story in Tableau and Python

4 PACE stages

Throughout these project notebooks, you'll see references to the problem-solving framework PACE. The following notebook components are labeled with the respective PACE stage: Plan, Analyze, Construct, and Execute.

4.1 PACE: Plan

Question: What methods are best for identifying outliers?

First we will find the mean and median of the different numerical variables, and if they are sufficiently different we can infer that there are outliers on one side of the distribution skewing the mean. Additionally, a boxplot can tell use a lot of additional information about the skew of the distribution. This is especially valuable if we make severable boxplots with different proportions of the data represented by the boxes on each sides of the mean. Then we can get a clear idea of where the data values are concentrated in the distribution.

Question: How do you make the decision to keep or exclude outliers from any future models?

It is often the case that we can generate a new value. This can be done by analyzing the rest of the data for the way this variable interacts with others, and then using the values of those other variables for our data point to predict and input a typical value. Of course, if the outliers are not problematic because we aren't using that variable in a model, or maybe we aren't even creating a model, then we can just keep them. Although we want to delete data as seldom as possible, if outliers are known to be incorrect inputs we can safely delete them.

4.1.1 Task 1. Imports, links, and loading

Go to Tableau Public The following link will help you complete this activity. Keep Tableau Public open as you proceed to the next steps.

Link to supporting materials: Public Tableau: https://public.tableau.com/s/. Note that the TikTok dataset can be downloaded directly from this notebook by going to "Lab Files" in the menu bar at the top of the page, clicking into the "/home/jovyan/work" folder, selecting `tiktok_dataset.csv`, and clicking "Download" above the list of files.

For EDA of the data, import the packages that would be most helpful, such as `pandas`, `numpy`, `matplotlib.pyplot`, and `seaborn`.

[1]:

```python
# Import packages for data manipulation
import pandas as pd
import numpy as np

# Import packages for data visualization
import matplotlib.pyplot as plt
import seaborn as sns
```

Then, load the dataset into a dataframe. Read in the data and store it as a dataframe object.

Note: As shown in this cell, the dataset has been automatically loaded in for you. You do not need to download the .csv file, or provide more code, in order to access the dataset and proceed with this lab. Please continue with this activity by completing the following instructions.

[2]:

```python
# Load dataset into dataframe
data = pd.read_csv("tiktok_dataset.csv")
```

4.2 PACE: Analyze

4.2.1 Task 2a: Data exploration and cleaning

The first step is to assess your data. Check the Data Source page on Tableau Public to get a sense of the size, shape and makeup of the data set.

Consider functions that help you understand and structure the data.

- `.head()`
- `.info()`
- `.describe()`
- `.groupby()`
- `.sort_values()`

Consider the following questions as you work:

What do you do about missing data (if any)?

Are there data outliers?

Start by discovering, using `.head()`, `.size`, and `.shape`.

```
[3]: # Display and examine the first few rows of the dataframe
     data.head(10)
```

```
[3]:
    # claim_status    video_id  video_duration_sec  \
0   1        claim  7017666017                  59
1   2        claim  4014381136                  32
2   3        claim  9859838091                  31
3   4        claim  1866847991                  25
4   5        claim  7105231098                  19
5   6        claim  8972200955                  35
6   7        claim  4958886992                  16
7   8        claim  2270982263                  41
8   9        claim  5235769692                  50
9  10        claim  4660861094                  45

                            video_transcription_text verified_status  \
0  someone shared with me that drone deliveries a…    not verified
1  someone shared with me that there are more mic…    not verified
2  someone shared with me that american industria…    not verified
3  someone shared with me that the metro of st. p…    not verified
4  someone shared with me that the number of busi…    not verified
5  someone shared with me that gross domestic pro…    not verified
6  someone shared with me that elvis presley has …    not verified
7  someone shared with me that the best selling s…    not verified
8  someone shared with me that about half of the …    not verified
9  someone shared with me that it would take a 50…        verified

  author_ban_status  video_view_count  video_like_count  video_share_count  \
0      under review          343296.0           19425.0              241.0
1            active          140877.0           77355.0            19034.0
2            active          902185.0           97690.0             2858.0
3            active          437506.0          239954.0            34812.0
4            active           56167.0           34987.0             4110.0
5      under review          336647.0          175546.0            62303.0
6            active          750345.0          486192.0           193911.0
7            active          547532.0            1072.0               50.0
8            active           24819.0           10160.0             1050.0
9            active          931587.0          171051.0            67739.0

   video_download_count  video_comment_count
0                   1.0                  0.0
1                1161.0                684.0
2                 833.0                329.0
3                1234.0                584.0
4                 547.0                152.0
```

```
5             4293.0              1857.0
6             8616.0              5446.0
7               22.0                11.0
8               53.0                27.0
9             4104.0              2540.0
```

[4]:
```
# Get the size of the data
data.size
```

[4]:
```
232584
```

[5]:
```
# Get the shape of the data
data.shape
```

[5]:
```
(19382, 12)
```

Get basic information about the data, using `.info()`.

[6]:
```
# Get basic information about the data
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19382 entries, 0 to 19381
Data columns (total 12 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   #                         19382 non-null  int64
 1   claim_status              19084 non-null  object
 2   video_id                  19382 non-null  int64
 3   video_duration_sec        19382 non-null  int64
 4   video_transcription_text  19084 non-null  object
 5   verified_status           19382 non-null  object
 6   author_ban_status         19382 non-null  object
 7   video_view_count          19084 non-null  float64
 8   video_like_count          19084 non-null  float64
 9   video_share_count         19084 non-null  float64
 10  video_download_count      19084 non-null  float64
 11  video_comment_count       19084 non-null  float64
dtypes: float64(5), int64(3), object(4)
memory usage: 1.8+ MB
```

Generate a table of descriptive statistics, using `.describe()`.

[7]:
```
# Generate a table of descriptive statistics
data.describe()
```

[7]:
```
                  #      video_id  video_duration_sec  video_view_count  \
count  19382.000000  1.938200e+04        19382.000000      19084.000000
```

```
mean    9691.500000  5.627454e+09           32.421732     254708.558688
std     5595.245794  2.536440e+09           16.229967     322893.280814
min        1.000000  1.234959e+09            5.000000         20.000000
25%     4846.250000  3.430417e+09           18.000000       4942.500000
50%     9691.500000  5.618664e+09           32.000000       9954.500000
75%    14536.750000  7.843960e+09           47.000000     504327.000000
max    19382.000000  9.999873e+09           60.000000     999817.000000

       video_like_count  video_share_count  video_download_count  \
count      19084.000000       19084.000000          19084.000000
mean       84304.636030       16735.248323           1049.429627
std       133420.546814       32036.174350           2004.299894
min            0.000000           0.000000              0.000000
25%          810.750000         115.000000              7.000000
50%         3403.500000         717.000000             46.000000
75%       125020.000000       18222.000000           1156.250000
max       657830.000000      256130.000000          14994.000000

       video_comment_count
count         19084.000000
mean            349.312146
std             799.638865
min               0.000000
25%               1.000000
50%               9.000000
75%             292.000000
max            9599.000000
```

4.2.2 Task 2b. Assess data types

In Tableau, staying on the data source page, double check the data types of the columns in the dataset. Refer to the dimensions and measures in Tableau.

Review the instructions linked in the previous Activity document to create the required Tableau visualization.

4.2.3 Task 2c. Select visualization type(s)

Select data visualization types that will help you understand and explain the data.

Now that you know which data columns you'll use, it is time to decide which data visualization makes the most sense for EDA of the TikTok dataset. What type of data visualization(s) would be most helpful? Consider the distribution of the data.

- Line graph
- Bar chart
- Box plot

- Histogram
- Heat map
- Scatter plot
- A geographic map

To understand the distribution of the data best, boxplots and histograms will be the right tools.

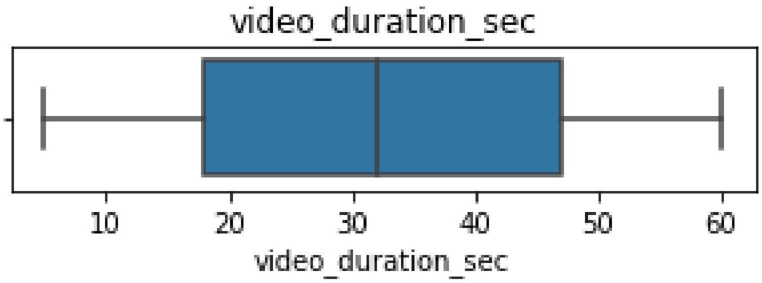
4.3 PACE: Construct

4.3.1 Task 3. Build visualizations

Now that you have assessed your data, it's time to plot your visualization(s).

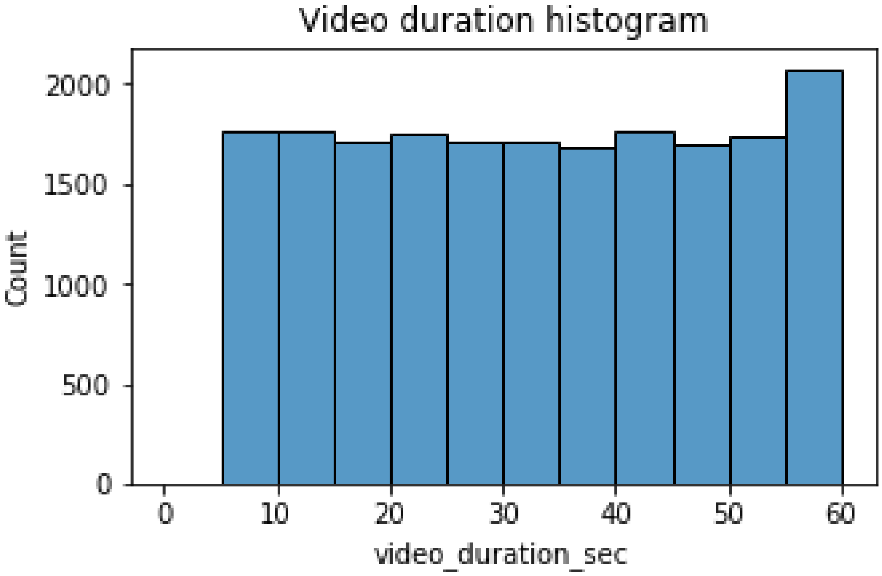
video_duration_sec Create a box plot to examine the spread of values in the `video_duration_sec` column.

```
[8]: # Create a boxplot to visualize distribution of `video_duration_sec'
     plt.figure(figsize=(5,1))
     plt.title('video_duration_sec')
     sns.boxplot(x=data['video_duration_sec']);
```



Create a histogram of the values in the `video_duration_sec` column to further explore the distribution of this variable.

```
[9]: # Create a histogram
     plt.figure(figsize=(5,3))
     sns.histplot(data['video_duration_sec'], bins=range(0,61,5))
     plt.title('Video duration histogram');
```

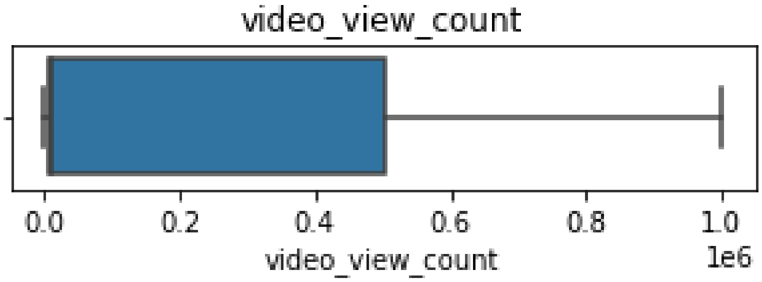


Question: What do you notice about the duration and distribution of the videos?

The duration of videos is largely distributed uniformly for video durations less than a minute.

video_view_count Create a box plot to examine the spread of values in the `video_view_count` column.

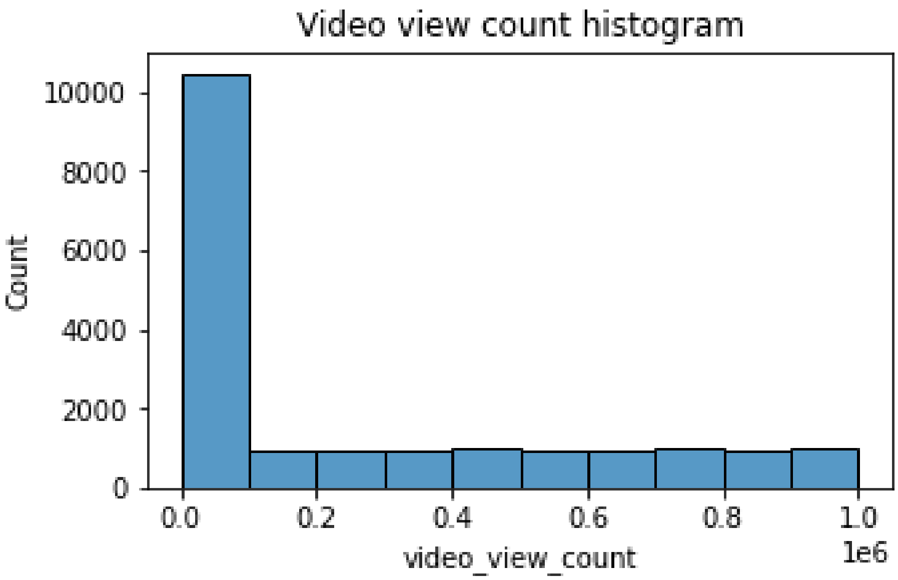
[10]:
```
# Create a boxplot to visualize distribution of `video_view_count`
plt.figure(figsize=(5, 1))
plt.title('video_view_count')
sns.boxplot(x=data['video_view_count']);
```



Create a histogram of the values in the `video_view_count` column to further explore the distribution of this variable.

[11]:

```
# Create a histogram
plt.figure(figsize=(5,3))
sns.histplot(data['video_view_count'], bins=range(0,(10**6+1),10**5))
plt.title('Video view count histogram');
```



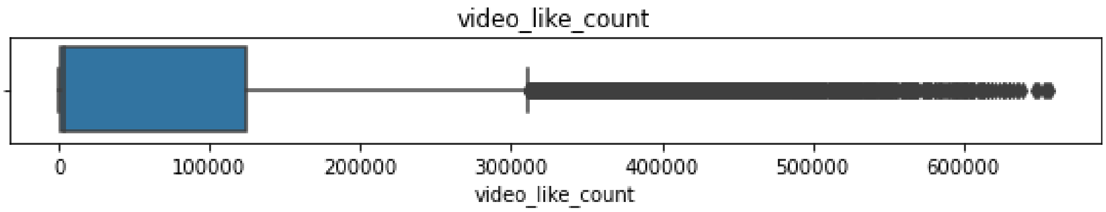
Question: What do you notice about the distribution of this variable?

Dramatically right skewed to videos with less than 100,000 views.

video_like_count Create a box plot to examine the spread of values in the `video_like_count` column.

[12]:

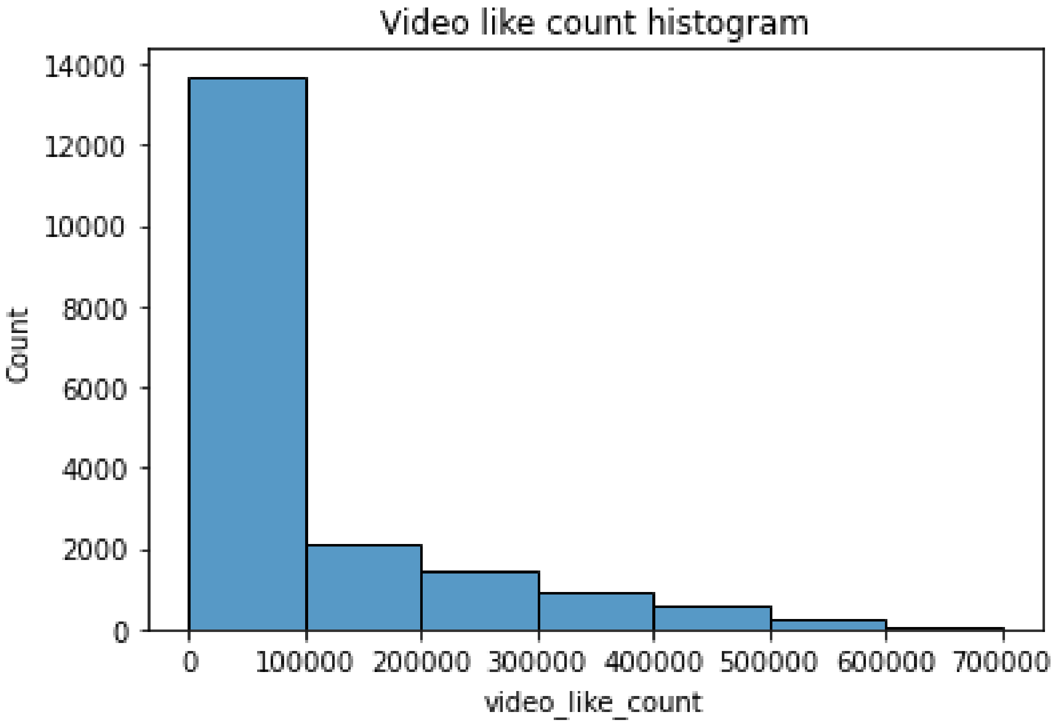
```
# Create a boxplot to visualize distribution of `video_like_count`
plt.figure(figsize=(10,1))
plt.title('video_like_count')
sns.boxplot(x=data['video_like_count']);
```



Create a histogram of the values in the `video_like_count` column to further explore the distribution of this variable.

[13]:

```
# Create a histogram
ax = sns.histplot(data['video_like_count'], bins=range(0,(7*10**5+1),10**5))
labels = [0] + [str(i) + 'k' for i in range(100, 701, 100)]

plt.title('Video like count histogram');
```



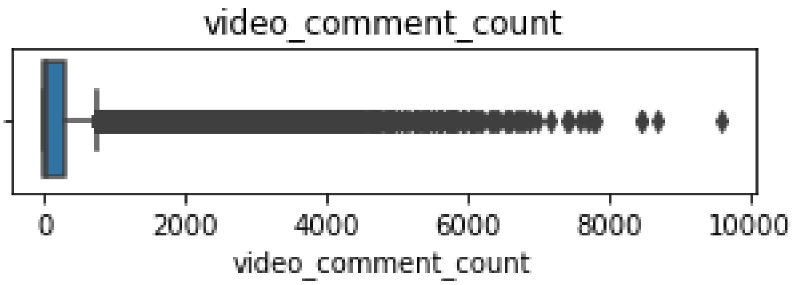
Question: What do you notice about the distribution of this variable?

Dramatically right skewed to videos with less than 100,000 likes.

video_comment_count Create a box plot to examine the spread of values in the `video_comment_count` column.

[14]:

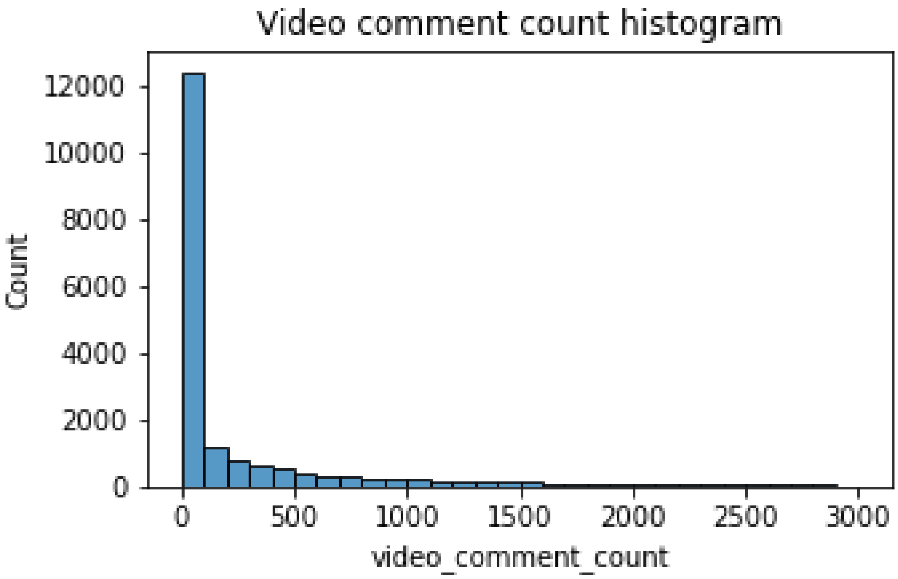
```
# Create a boxplot to visualize distribution of `video_comment_count`
plt.figure(figsize=(5,1))
plt.title('video_comment_count')
sns.boxplot(x=data['video_comment_count']);
```



Create a histogram of the values in the `video_comment_count` column to further explore the distribution of this variable.

[15]:

```
# Create a histogram
plt.figure(figsize=(5,3))
sns.histplot(data['video_comment_count'], bins=range(0,(3001),100))
plt.title('Video comment count histogram');
```



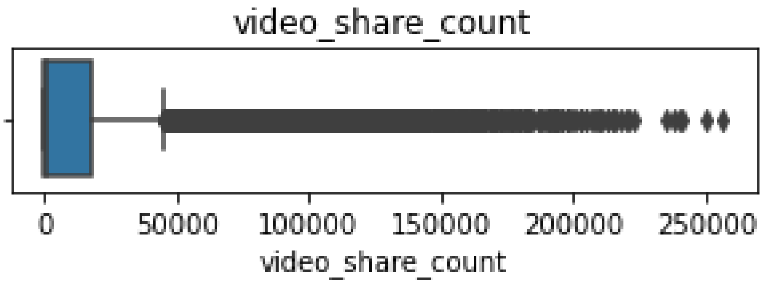
Question: What do you notice about the distribution of this variable?

Dramatically right skewed to videos with less than 100 comments.

video_share_count Create a box plot to examine the spread of values in the `video_share_count` column.

[16]:

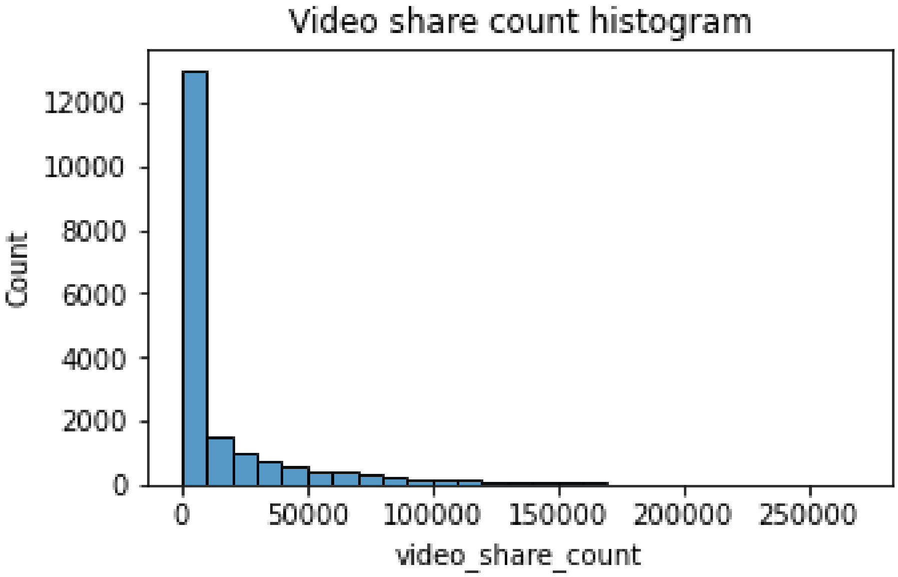
```
# Create a boxplot to visualize distribution of `video_share_count`
plt.figure(figsize=(5,1))
plt.title('video_share_count')
sns.boxplot(x=data['video_share_count']);
```



Create a histogram of the values in the `video_share_count` column to further explore the distribution of this variable.

[17]:

```
# Create a histogram
plt.figure(figsize=(5,3))
sns.histplot(data['video_share_count'], bins=range(0,(270001),10000))
plt.title('Video share count histogram');
```



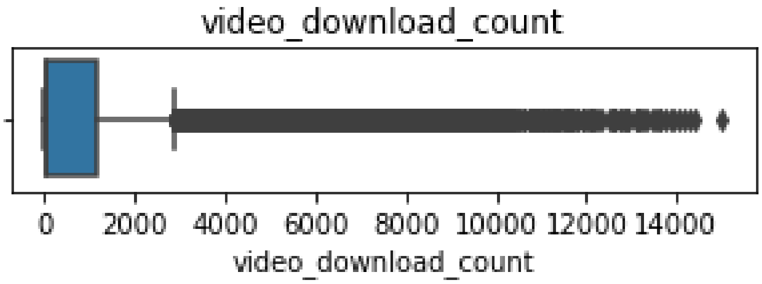
Question: What do you notice about the distribution of this variable?

Dramatically right skewed to videos with less than 10,000 shares.

video_download_count Create a box plot to examine the spread of values in the `video_download_count` column.

[18]:

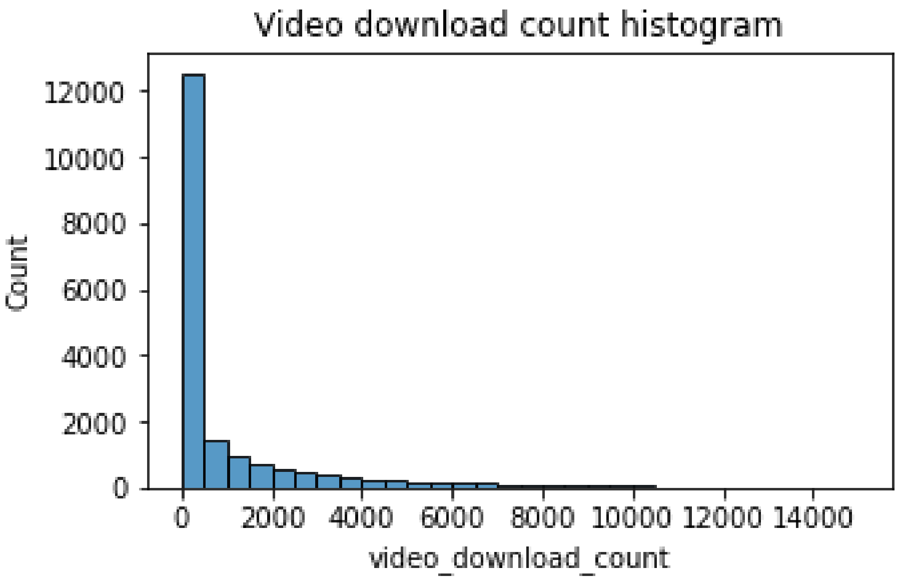
```
# Create a boxplot to visualize distribution of `video_download_count`
plt.figure(figsize=(5,1))
plt.title('video_download_count')
sns.boxplot(x=data['video_download_count']);
```



Create a histogram of the values in the `video_download_count` column to further explore the distribution of this variable.

[19]:

```
# Create a histogram
plt.figure(figsize=(5,3))
sns.histplot(data['video_download_count'], bins=range(0,(15001),500))
plt.title('Video download count histogram');
```

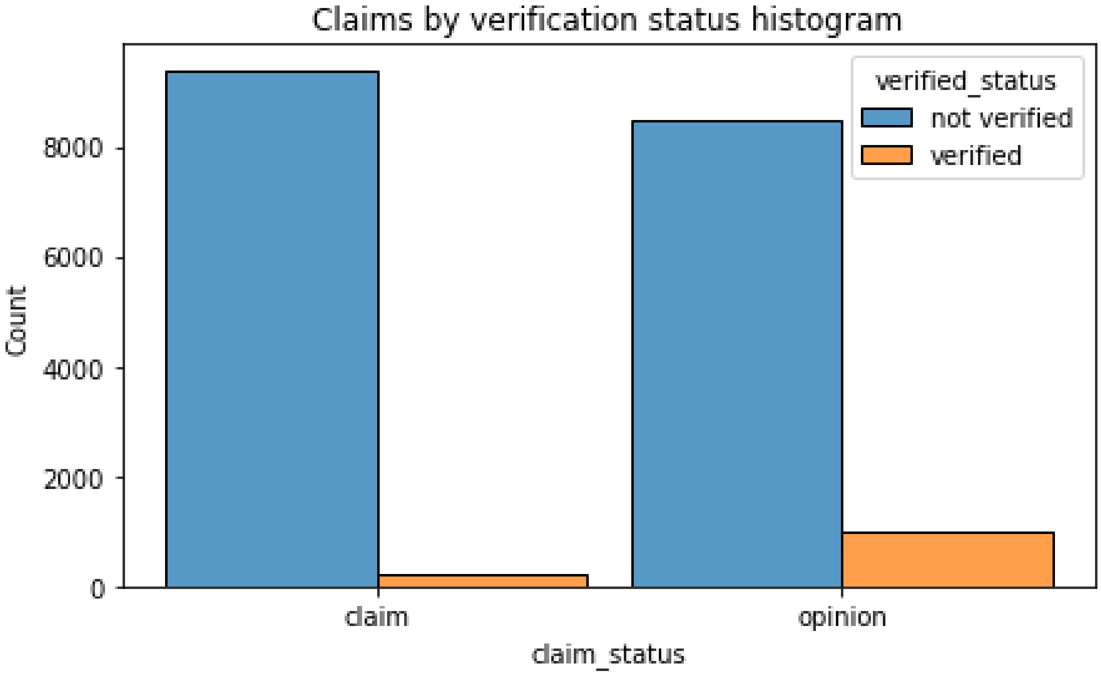


Question: What do you notice about the distribution of this variable?

Dramatically right skewed to videos with less than 500 downloads.

Claim status by verification status Now, create a histogram with four bars: one for each combination of claim status and verification status.

[20]:
```python
# Create a histogram
plt.figure(figsize=(7,4))
sns.histplot(data=data,
             x='claim_status',
             hue='verified_status',
             multiple='dodge',
             shrink=0.9)
plt.title('Claims by verification status histogram');
```



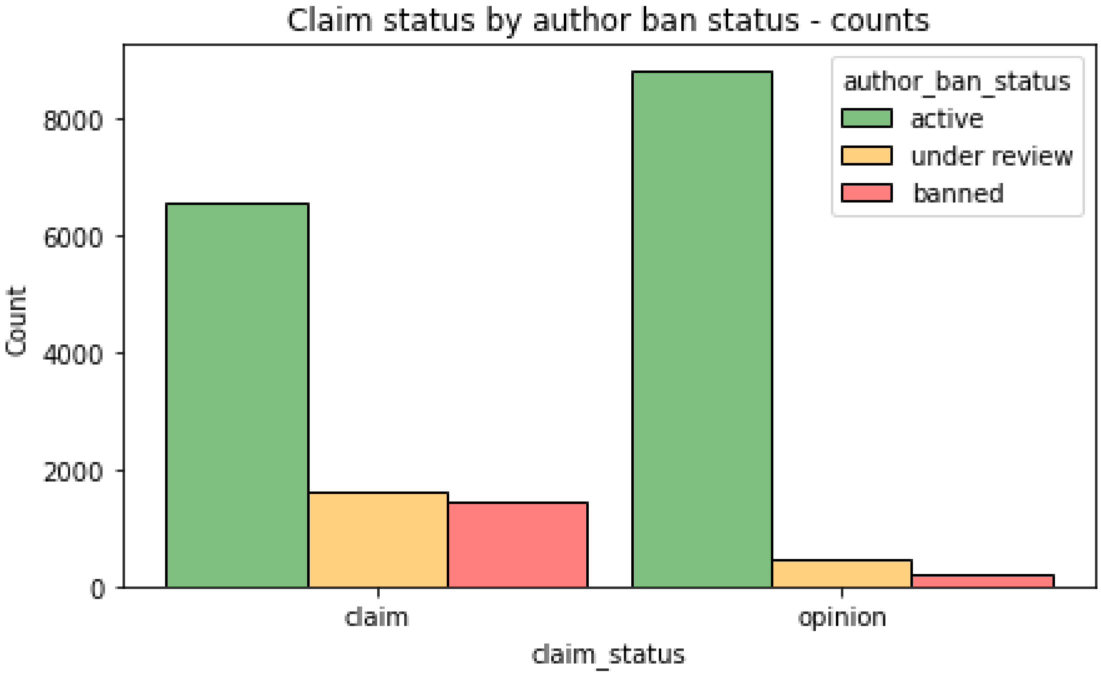
Question: What do you notice about the number of verified users compared to unverified? And how does that affect their likelihood to post opinions?

The ratio of not verified/verified is much smaller for opinions than claims.

Claim status by author ban status The previous course used a `groupby()` statement to examine the count of each claim status for each author ban status. Now, use a histogram to communicate the same information.

[21]:
```python
# Create a histogram
fig = plt.figure(figsize=(7,4))
```

```
sns.histplot(data, x='claim_status', hue='author_ban_status',
             multiple='dodge',
             hue_order=['active', 'under review', 'banned'],
             shrink=0.9,
             palette={'active':'green', 'under review':'orange', 'banned':
 ↪'red'},
             alpha=0.5)
plt.title('Claim status by author ban status - counts');
```

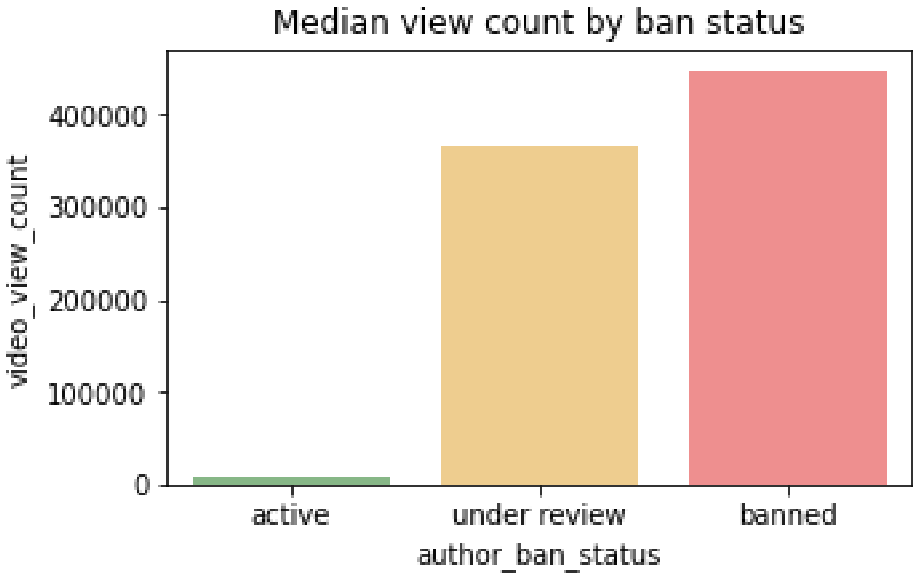


Question: What do you notice about the number of active authors compared to banned authors for both claims and opinions?

The ratio of active/banned authors is a lot larger for opinions than claims.

Median view counts by ban status Create a bar plot with three bars: one for each author ban status. The height of each bar should correspond with the median number of views for all videos with that author ban status.

[22]:
```
# Create a bar plot
ban_status_counts = data.groupby(['author_ban_status']).median(
    numeric_only=True).reset_index()

fig = plt.figure(figsize=(5,3))
sns.barplot(data=ban_status_counts,
            x='author_ban_status',
            y='video_view_count',
            order=['active', 'under review', 'banned'],
            palette={'active':'green', 'under review':'orange', 'banned':'red'},
            alpha=0.5)
plt.title('Median view count by ban status');
```

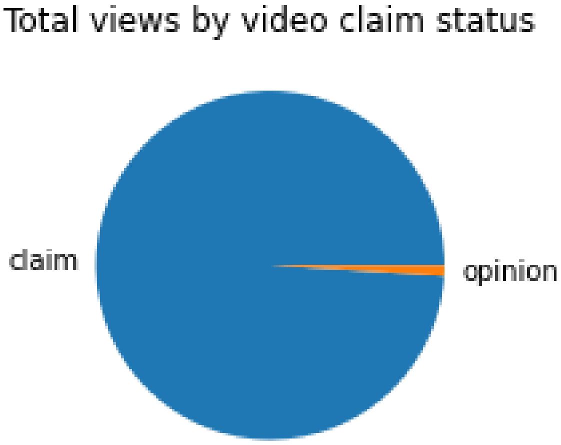


[23]:
```
# Calculate the median view count for claim status
data.groupby('claim_status')['video_view_count'].median()
```

[23]:
```
claim_status
claim      501555.0
opinion      4953.0
Name: video_view_count, dtype: float64
```

Total views by claim status Create a pie graph that depicts the proportions of total views for claim videos and total views for opinion videos.

[24]:
```
# Create a pie graph
fig = plt.figure(figsize=(3,3))
plt.pie(data.groupby('claim_status')['video_view_count'].sum(),␣
 ↪labels=['claim', 'opinion'])
plt.title('Total views by video claim status');
```



Question: What do you notice about the overall view count for claim status?

The vast majority of video views go to tiktoks considered claims.

4.3.2 Task 4. Determine outliers

When building predictive models, the presence of outliers can be problematic. For example, if you were trying to predict the view count of a particular video, videos with extremely high view counts might introduce bias to a model. Also, some outliers might indicate problems with how data was captured or recorded.

The ultimate objective of the TikTok project is to build a model that predicts whether a video is a claim or opinion. The analysis you've performed indicates that a video's engagement level is strongly correlated with its claim status. There's no reason to believe that any of the values in the TikTok data are erroneously captured, and they align with expectation of how social media works: a very small proportion of videos get super high engagement levels. That's the nature of viral content.

Nonetheless, it's good practice to get a sense of just how many of your data points could be considered outliers. The definition of an outlier can change based on the details of your project, and it helps to have domain expertise to decide a threshold. You've learned that a common way to determine outliers in a normal distribution is to calculate the interquartile range (IQR) and set a threshold that is 1.5 * IQR above the 3rd quartile.

In this TikTok dataset, the values for the count variables are not normally distributed. They are heavily skewed to the right. One way of modifying the outlier threshold is by calculating the **median** value for each variable and then adding 1.5 * IQR. This results in a threshold that is, in this case, much lower than it would be if you used the 3rd quartile.

Write a for loop that iterates over the column names of each count variable. For each iteration: 1. Calculate the IQR of the column 2. Calculate the median of the column 3. Calculate the outlier threshold (median + 1.5 * IQR) 4. Calculate the numer of videos with a count in that column that exceeds the outlier threshold 5. Print "Number of outliers, {column name}: {outlier count}"

```
Example:
Number of outliers, video_view_count: ___
Number of outliers, video_like_count: ___
Number of outliers, video_share_count: ___
Number of outliers, video_download_count: ___
Number of outliers, video_comment_count: ___
```

[25]:

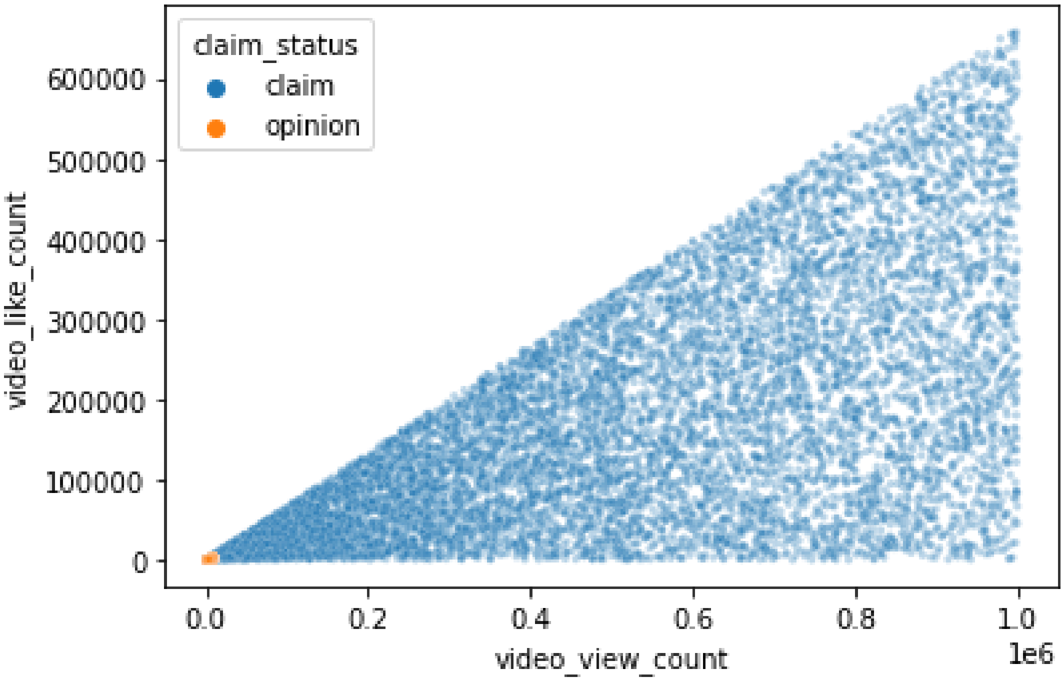
```
count_cols = ['video_view_count', # Grab only columns for tiktok count␣
 ↪statistics
              'video_like_count',
              'video_share_count',
              'video_download_count',
              'video_comment_count',
              ]

for column in count_cols:
    q1 = data[column].quantile(0.25)
    q3 = data[column].quantile(0.75)
    iqr = q3 - q1
    median = data[column].median()
    outlier_threshold = median + 1.5*iqr

    # Count the number of values that exceed the outlier threshold
    outlier_count = (data[column] > outlier_threshold).sum()
    print(f'Number of outliers for {column}:', outlier_count)
```

```
Number of outliers for video_view_count: 2343
Number of outliers for video_like_count: 3468
Number of outliers for video_share_count: 3732
Number of outliers for video_download_count: 3733
Number of outliers for video_comment_count: 3882
```

Scatterplot

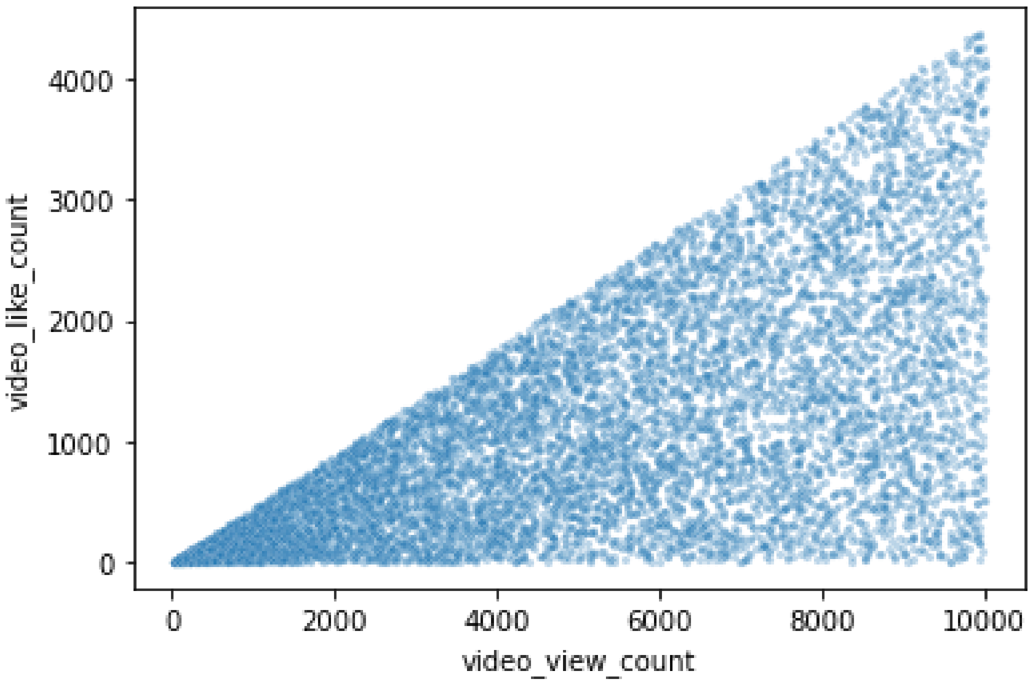
[26]:

```
# Create a scatterplot of `video_view_count` versus `video_like_count`␣
 ↪according to 'claim_status'
sns.scatterplot(x=data["video_view_count"], y=data["video_like_count"],
                hue=data["claim_status"], s=10, alpha=.3)
plt.show()
```



[27]:

```
# Create a scatterplot of ``video_view_count` versus `video_like_count` for␣
 ↪opinions only
opinion = data[data['claim_status']=='opinion']
sns.scatterplot(x=opinion["video_view_count"], y=opinion["video_like_count"],
                s=10, alpha=.3)
plt.show()
```



You can do a scatterplot in Tableau Public as well, which can be easier to manipulate and present. If you'd like step by step instructions, you can review the instructions linked in the previous Activity page.

4.4 PACE: Execute

4.4.1 Task 5a. Results and evaluation

Having built visualizations in Tableau and in Python, what have you learned about the dataset? What other questions have your visualizations uncovered that you should pursue?

Pro tip: Put yourself in your client's perspective, what would they want to know?

Use the following code cells to pursue any additional EDA. Also use the space to make sure your visualizations are clean, easily understandable, and accessible.

Ask yourself: Did you consider color, contrast, emphasis, and labeling?

Yes, it is essential to use colors to differentiate distinct variables in each visualization, and to choose accessible colors; avoiding color combinations like green/red.

4.4.2 Task 5b. Conclusion

Make it professional and presentable

You have visualized the data you need to share with the director now. Remember, the goal of a data visualization is for an audience member to glean the information on the chart in mere seconds.

Questions to ask yourself for reflection: Why is it important to conduct Exploratory Data Analysis? What other visuals could you create?

EDA is important because missing data and outliers can dramatically skew your understanding of a system if you include missing data without understanding why it is missing or use the wrong statistics on outliers.

Visualizations helped me understand correlations between variables such as: video_view_count, video_like_count, video_share_count, author_ban_status, and video_download_count with claim_status.

You've now completed a professional data visualization according to a business need. Well done! Be sure to save your work as a reference for later work in Tableau.

Congratulations! You've completed this lab. However, you may not notice a green check mark next to this item on Coursera's platform. Please continue your progress regardless of the check mark. Just click on the "save" icon at the top of this notebook to ensure your work has been logged.