**A logo of a university

Description automatically generated**

**Department of Mathematics and Statistics**

**STA 301 – Foundations of Statistics for Data Science**

**Online News Popularity Prediction and Analysis**

**Abstract**

This project aims to determine which independent variables affected the shares of an article the most. In the first section, we explain each of the variables included in the dataset. We analyze the summary statistics such as the mean, median and outliers in the data as well as construct histograms and boxplots to analyze the distribution of shares. We use pie charts to help analyze the distribution of categorical predictors. We also conduct a correlation matrix to analyze the correlation between independent variables. Furthermore, we conduct inferential statistics like Chi-square test, confidence intervals, one way anova and two way anova to determine the correlation between the independent variables. Finally, we build a logistic regression model to determine which variables predict shares of an article. We also build a regression model where shares is a continuous variable which results in producing a higher r-squared value of 13.3%.

**Section 1: Data Set Information**

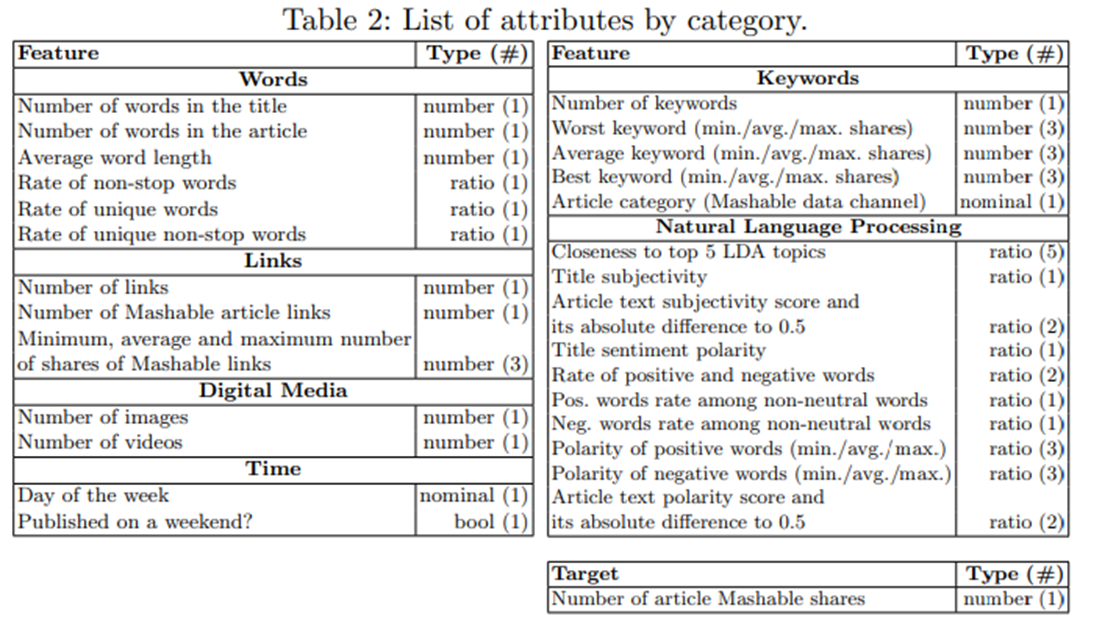
The project's main objective is to determine the factors contributing to the popularity of online news articles so that such factors will be prioritized when writing articles in the future. The data comprises 39,644 news articles from Mashable's online website. The OnlineNewsPopularity dataset summarizes features and statistics about articles published over two years – 2013 and 2014. Each article consists of statistics for different parameters that may be useful in determining the article's popularity which is measured through the number of shares. "Shares" is the target variable in the data set and is a continuous numerical variable.

Additionally, there are 60 other attributes which include two non-predictive features (URL and the Days between the article publication and the dataset acquisition. There are 43 continuous variables, which fall under groups such as Descriptions of Keywords, Words, LDAs, Polarities, Global processing, Links, and Digital Media. Few variables have negative values, so there needs to be further analysis for this. While there are 15 categorical variables, which are the genres, which day the article was posted, and if it was a weekend. These variables have binary inputs of '0' or '1.' For example, if the article is published on Monday, there would be one in that column, while the rest of the days (the rest six columns) have the value '0.' For further calculations, these columns may be combined to better understand the link of shares to the day of publishing. This means these categorical variables only focus on the '1,' as those values give more information about the variable.

As the dataset is larger, each feature needs to be analyzed to see the relation of the predictors to the target variable. The features have different ranges, so standardizing the data is required before any analysis. Additionally, the two non-predictors will be removed. As the dataset is large, 50% of the data would be taken for analysis. This means the dataset would now consist of 19822 observations. This would be further split into testing (80%) and training (20%). The analysis will be done on the training data, while in the end, the testing data will be used for prediction.

In conclusion, the data will be analyzed concerning the target variable: the number of shares with the rest of the predictors. Regression models are used in Minitab software to study the connection between different variables. Regression offers several techniques, like stepwise and best subset, to help determine which variables are essential for the model. Additionally, focusing on the feature selection, These techniques would compare the features and find the variables that impact the outcome of shares most.

The descriptive report of the variables are summarized in the following table. Due to the long list of variables, we have added detailed descriptions in the appendix.



**Section 2**

**Subsection 2.1**: *Exploratory data analysis: Graphs and Summary Statistics*

Training Data:

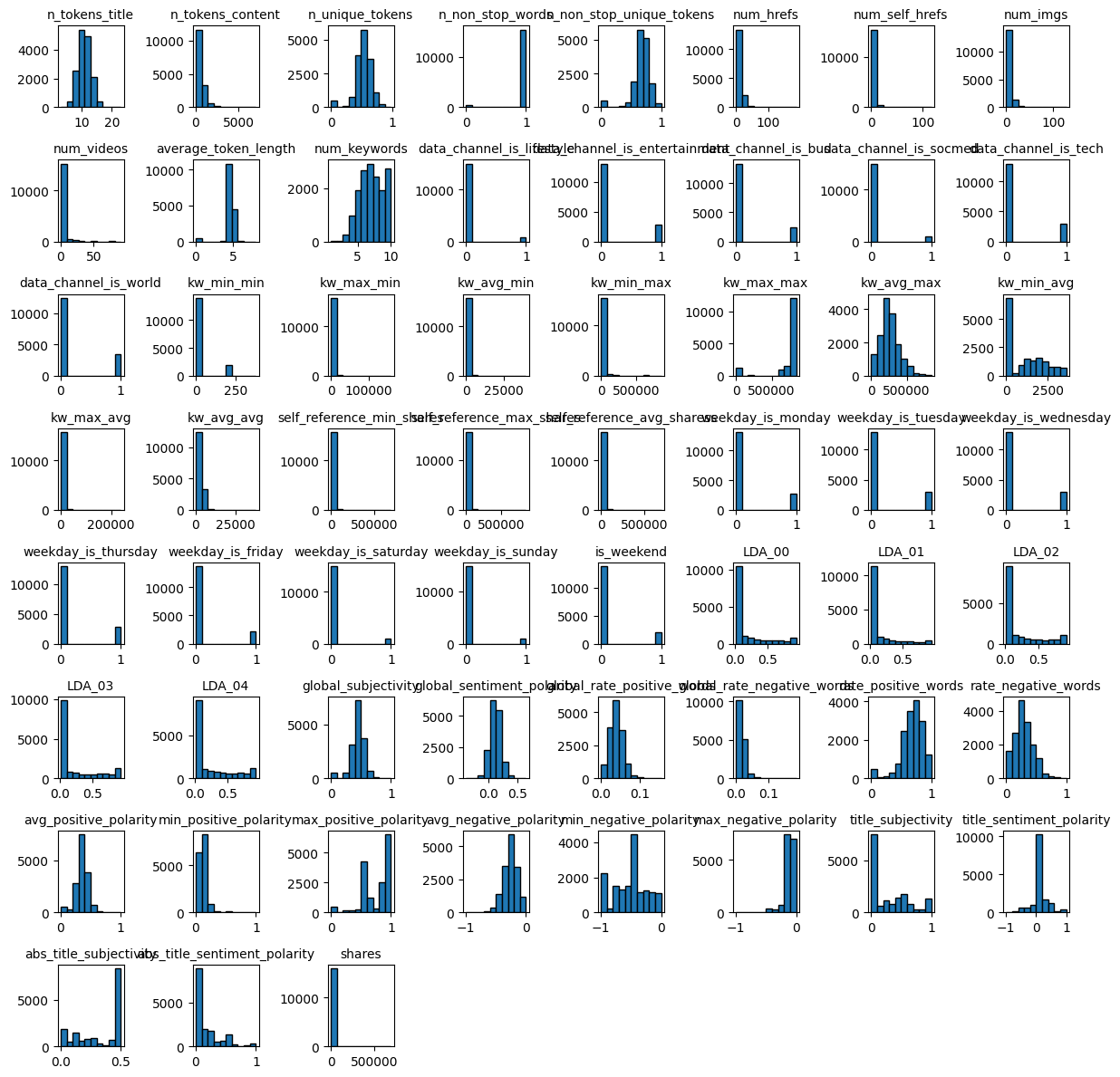
A number of steps were done to preprocess this data, this was needed to proceed with building a model:

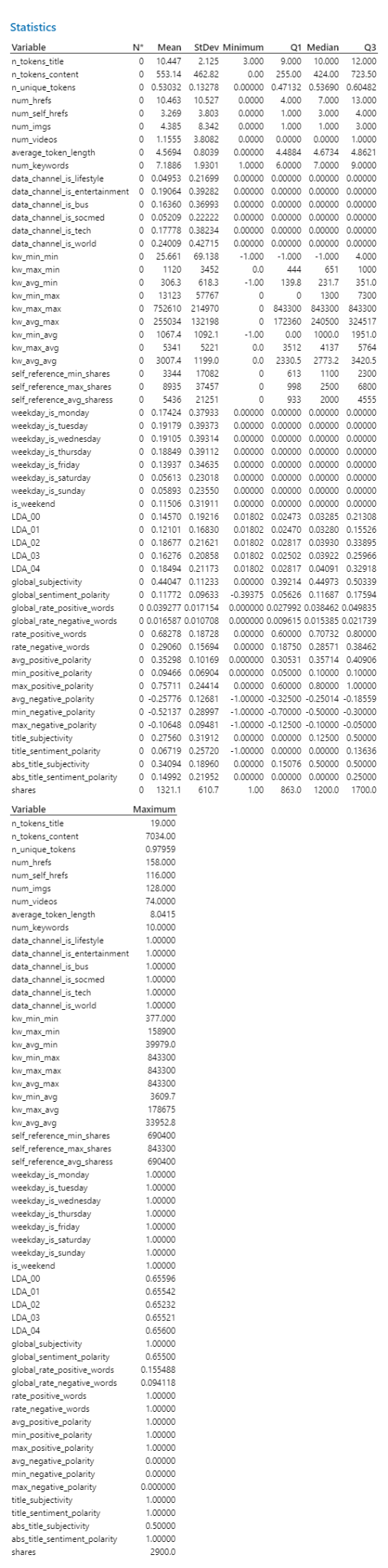
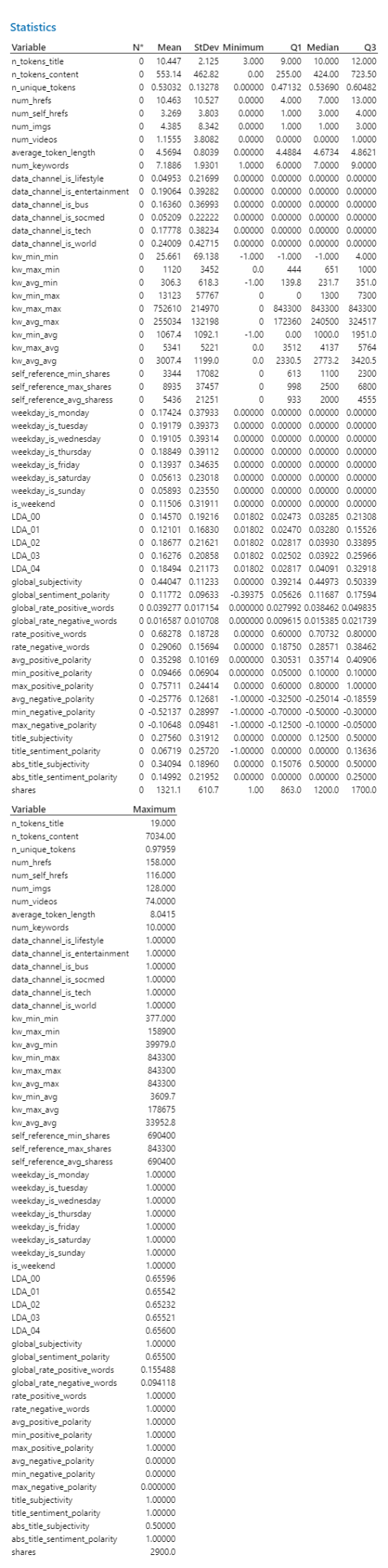
* The URL and time delta columns have been omitted as they are non-predictor variables mentioned in the dataset itself. They are meta-data and can't be a feature
* The is\_weeked would be removed later on, as there are columns that already mention weekday\_is\_sunday and weekday\_is\_saturday. There is duplicated information
* NaN, or Null rows need to be removed, but the columns didn't have null values.
* As all variables have different ranges, it is needed to standardize the data.

After these few pointers, there are 45 continuous numeric values and 13 categorical variables. To understand the distribution of the variables, we can look at the histograms.

* In the diagram, Shares, num\_hrefs, num\_self\_hrefs, num\_imgs, num\_videos are skewed to the right. While n\_toklen\_title, n\_unique\_tokens seems to have a general normal.
* The LDA’s are also highly right-skewed.
* The global\_rate\_positive\_words, global\_rate\_negative\_words and rate\_negative\_words variables are also right skewed.
* The n\_non\_stop words variable and rate\_positive\_words variable are highly left-skewed.

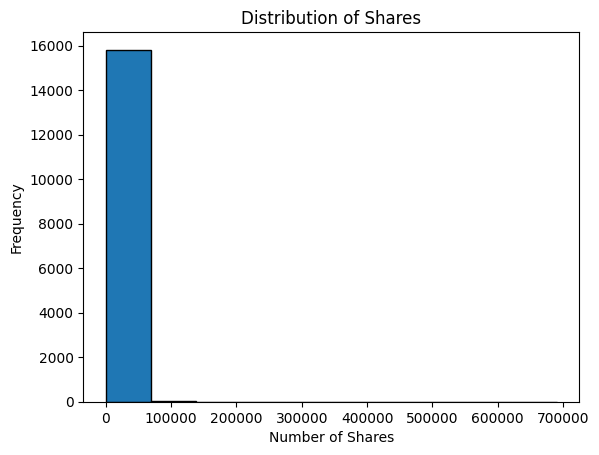
Additionally, this shows that there may be outliers visible for every variable. This can be combated if there is some sort of transformation for the variables. As the data contains negative, zero as well as positive values the transformation would vary from predictor to another.



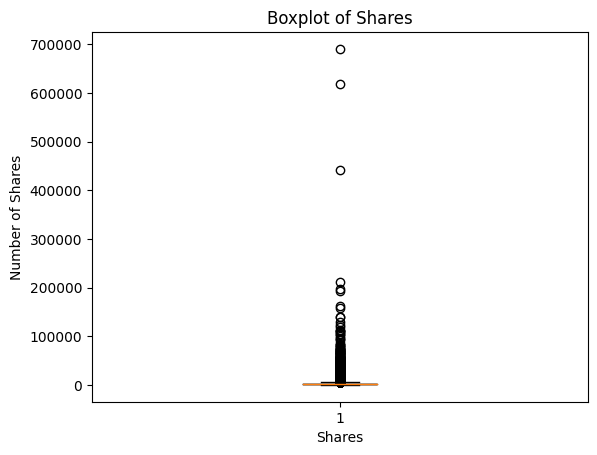


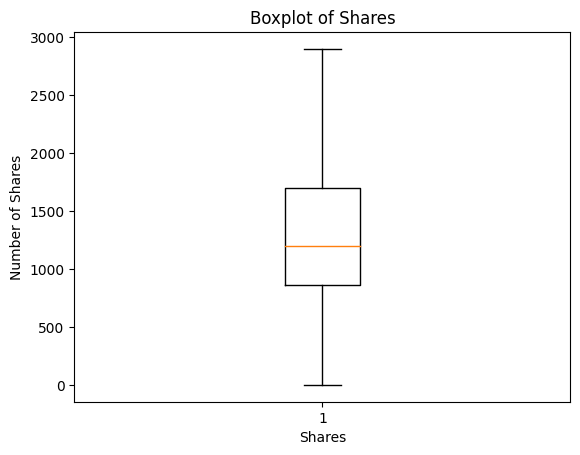
Target Variable (Number of shares):

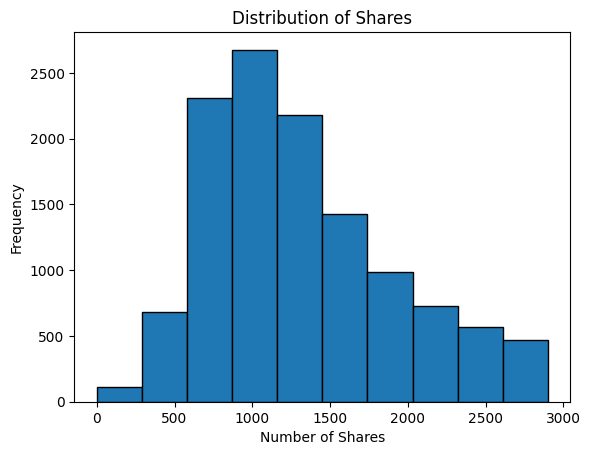
The target variable was examined while using a histogram:

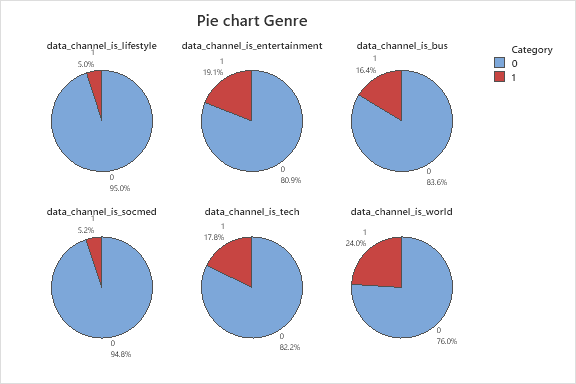


This picture indicates that the “shares” is highly skewed. This can be combated if the “shares” is filtered out.

****

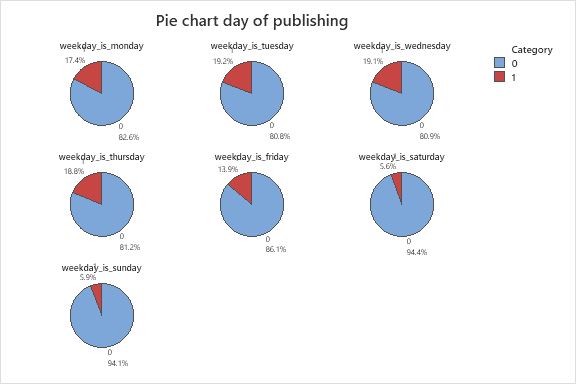
There are many outliers in shares, so there can be a cut-off. By observing the feature, there are many outliers, to be able to create a threshold, we need to review the description of shares. Based on the boxplot, the most optimum cut-off was 3000. This provided a better spread of shares. 



Additionally, this reduced the training dataset from 15858 to 12133 observations. The following analysis would be done on this data. 

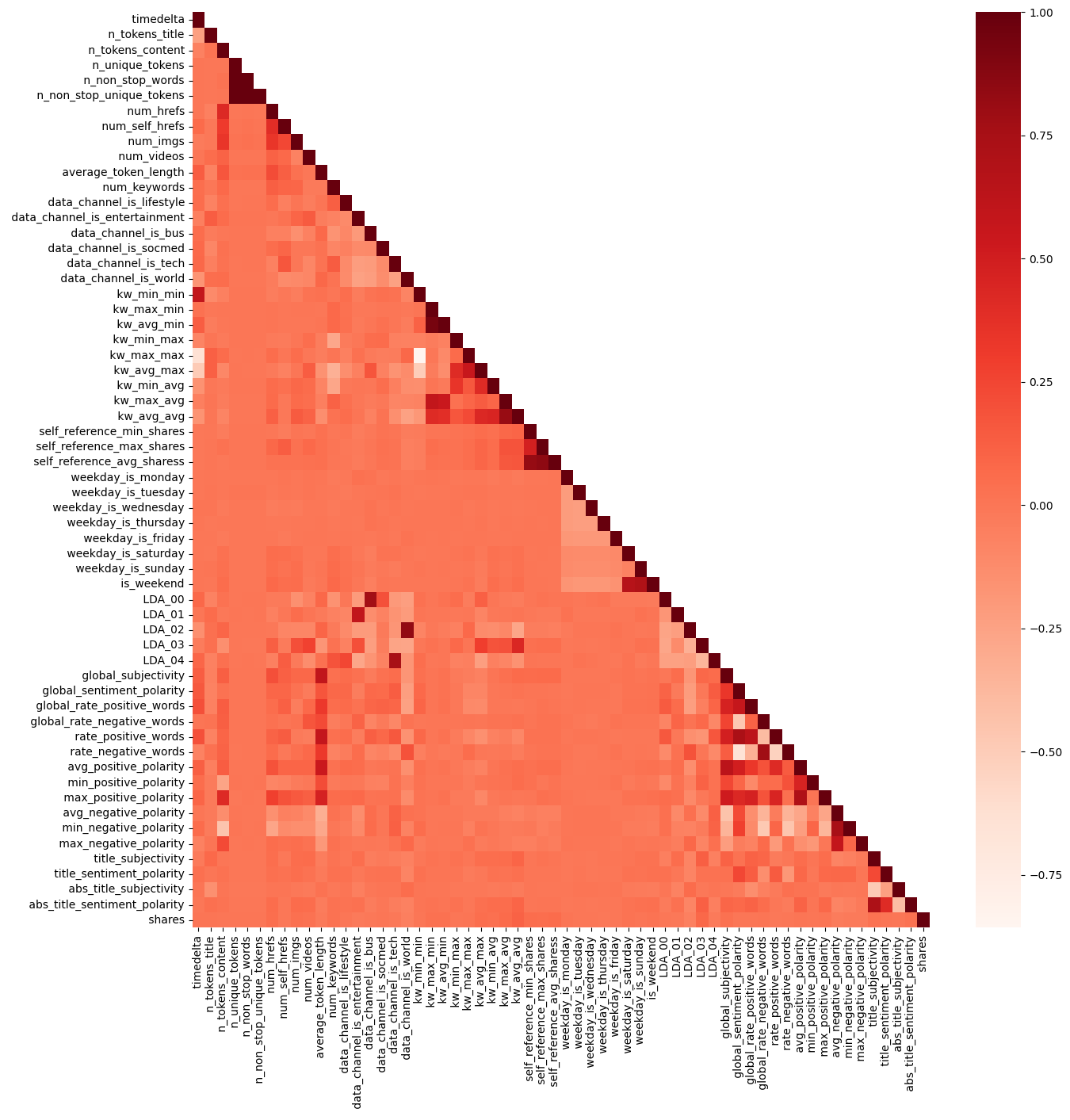
Categorical Data:

Based on the categorical data, the genre of the article can be represented in pie charts. The articles about the world are popular among other topics. While entertainment, business, and technology topics are also popular. The 1 represents 12133 articles, those articles are under that topic, while the 0 represents 12133 articles that are not under that topic. This is represented as different columns presented as “Data\_channel\_is.”

Similarly, the other categorical variable is “weekday\_is\_.” All the days of the week were observed from Sunday to Saturday. Also, a variable shows that the article was published on the weekend. If we consider Sunday and Saturday as a weekend, those are the days with the lowest number of articles published. While Tuesday has the highest number of articles published. But the pie charts look similar for the weekdays. The 1 represents the 12133 articles, the number of articles published on that particular day, while the 0 represents among 12133 articles the number of articles that weren't published on that day. 

Correlation Matrix:

Heat maps are data visualization tools that represent values in a matrix or table format using colors to indicate the intensity of the values. Each cell or data point in the matrix is assigned a color in a heat map based on its value. Typically, a gradient color scheme is used, where the highest values are represented by darker red, and the lowest values are represented by lighter orange.



Here there are a few spots that represent a darker red. This means that they may be correlated. These include:

* Number of unique tokens and the Number of non-stop words
* Number of non-stop words and nonstop unique tokens.
* Keyword avg-min and Keywords max-min
* Rate of positive words and global sentiment polarity
* Rate of negative words and the global rate of negative words
* Avg negative polarity and min negative polarity
* Title subjectivity and abs title sentiment polarity

**Subsection 2.2:** *Inferential Statistics*

**Chi-square test:**

(See Appendix 3 for calculations) We conduct Pearson’s Chi-squared tests to test whether there are any associations between variables. In this particular case, we want to explore whether there are any associations between the group of categorical variables, data\_channel, and weekday. The hypotheses are defined as follows:

H0: The variables are independent

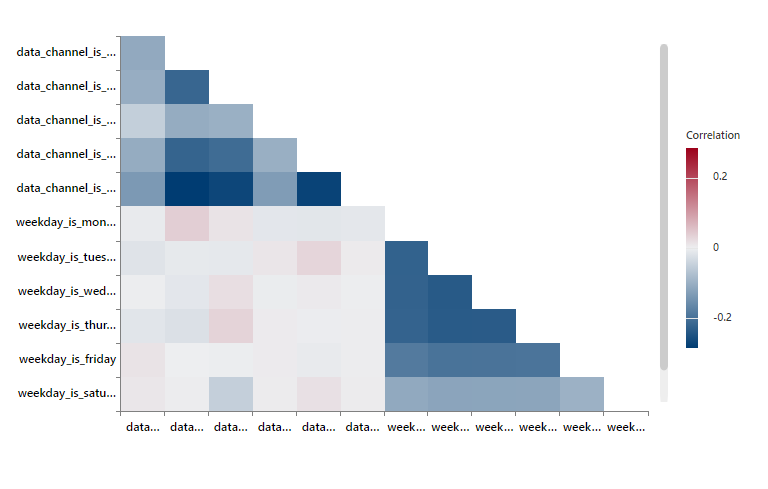
Ha: The variables are dependent

Confidence level = 95%

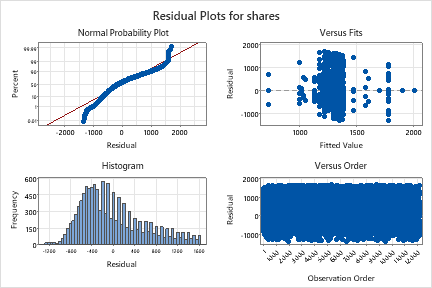
At 95% confidence level, we can reject the null hypothesis for the following combinations as p-value<0.05:

* Data\_channel\_is\_lifestyle X weekday\_is\_sunday
* Data\_channel\_is business X weekday\_is\_thursday, wednesday, saturday and sunday.
* Data\_channel\_is\_tech X weekday\_is\_tuesday and sunday
* Data\_channel\_is\_socmed X weekday\_is\_monday and saturday
* Data\_channel\_is\_entertainment X weekday\_is\_monday and thursday

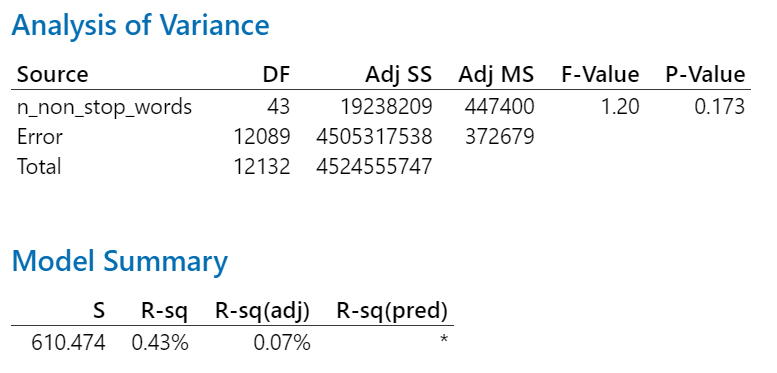
This means that the variables in each of these combinations are dependent and associated with each other. Since these variables are dependent, we can also conclude that these variables may have multicollinearity.

So, to make sure that there is no multicollinearity, we need to check the correlation between these variables. In the below graph, there is no value that exceeds 0.7. So we don't remove any variables.

**One-Way ANOVA**



*Between ‘shares’ and ‘n\_non\_stop\_tokens’*



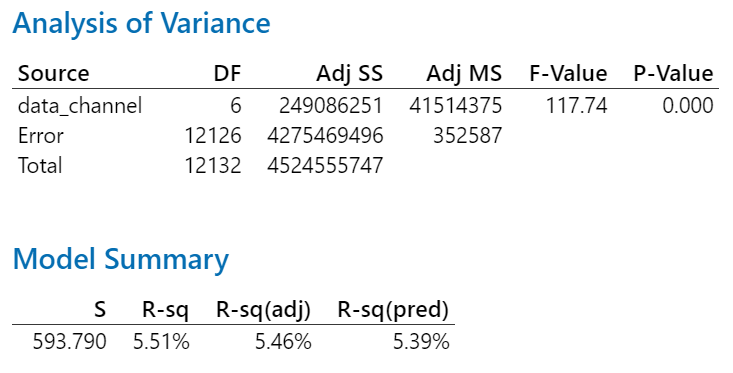
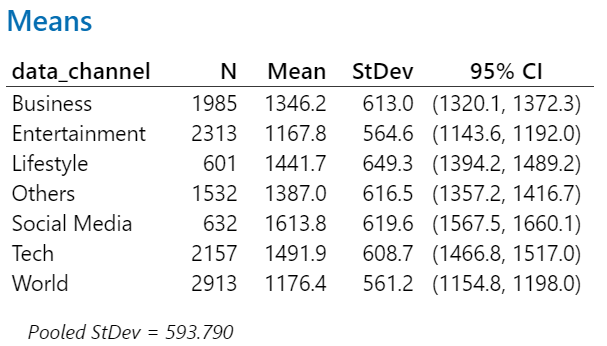
Hypothesis Test:

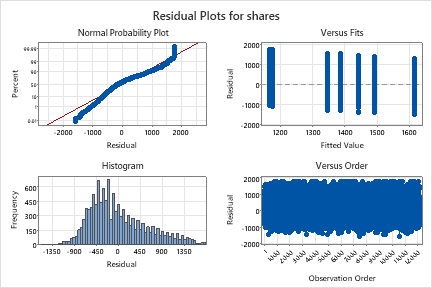
Ho : µ1 = µ2

Ha: at least one of the means is different from the others

Since the p-value (0.173) is more than 0.05, we fail to reject the null hypothesis. That means that at the 5% level of significance, the variable ‘n\_non\_stop\_tokens’ is insignificant and can be removed from the model. The normality plot shows a relatively normal distribution with a few outliers. The versus fits plot shows that there is no pattern in the data.

*Between ‘shares’ and ‘data\_channel’*





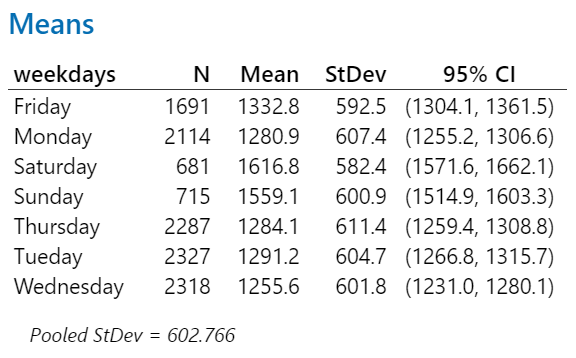
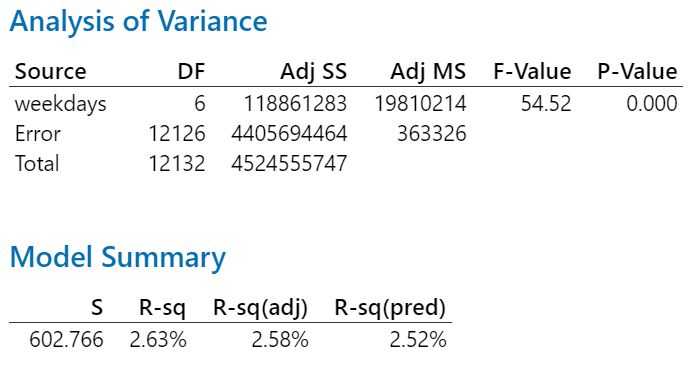
Hypothesis Test Method:

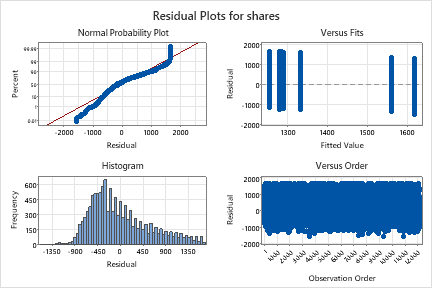
Ho : µ1 = µ2= µ3= µ4= µ5= µ6 =µ7

Ha: at least one of the means is different from the others

Since the p-value (0.000) is less than 0.05, we reject the null hypothesis. That means that at the 5% level of significance, the variable ‘data\_channel’ is significant and must remain in the model. Similar to the previous analysis, the normality plot shows a relatively normal distribution with a few outliers. The versus fits plot; however, seems to show a slight negative linear relationship.

*Between ‘shares’ and ‘weekdays’*





Hypothesis Test:

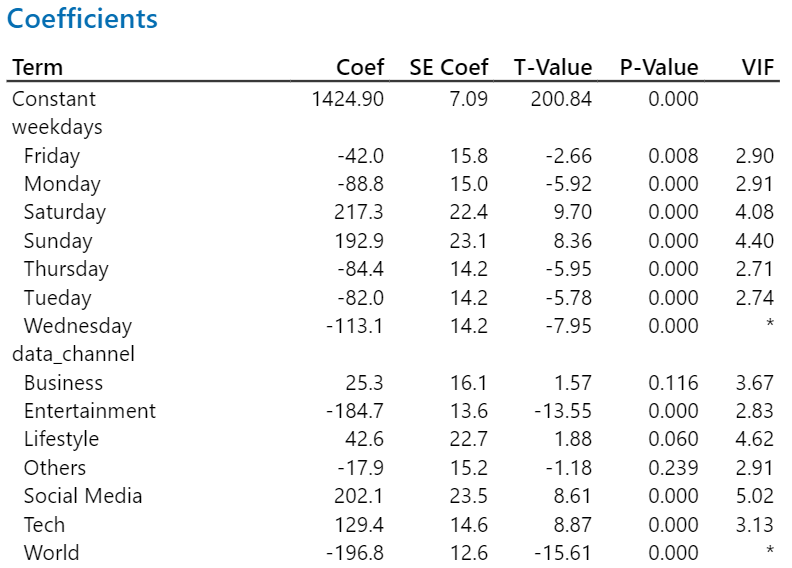
Ho : µ1 = µ2= µ3= µ4= µ5= µ6 =µ7

Ha: at least one of the means is different from the others

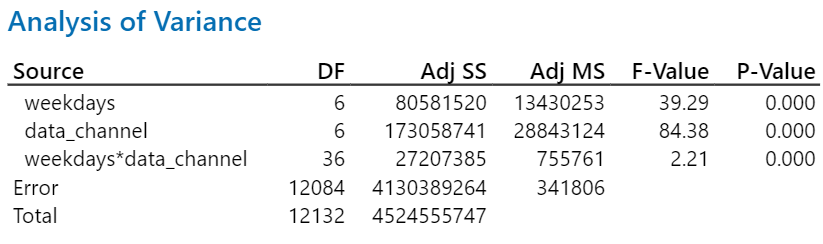
Since the p-value (0.000) is less than 0.05, we reject the null hypothesis. That means that at the 5% level of significance, the variable ‘weekdays’ is significant and must remain in the model.

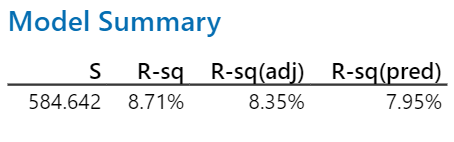
The normality plot shows a relatively normal distribution with a few outliers. The versus fits plot; however, seems to show a slight negative linear relationship.

**Two-Way ANOVA**



*Between ‘weekdays’ and ‘data\_channel’*



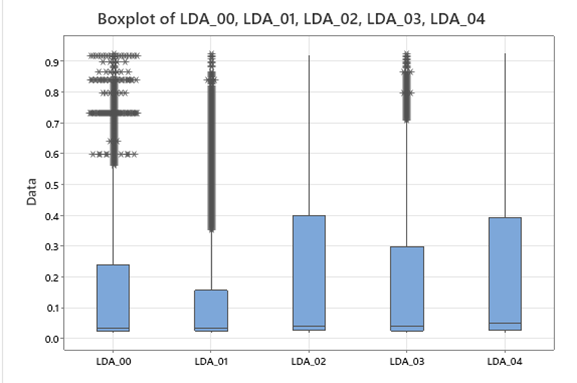


Hypothesis Test:

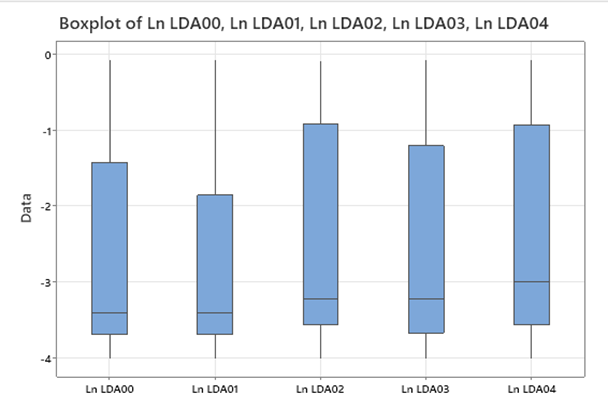
Ho : µ1 = µ2= µ3

Ha: at least one of the means is different from the others

Since the p-value (0.000) is less than 0.05, we reject the null hypothesis, which means that the 2 factors ‘weekdays’ and ‘data\_channel’, as well as their interaction, are statistically significant at the 5% significance level. Additionally, the VIFs of the factors are relatively low (threshold = 5), which indicates that there’s low multicollinearity between the two variables. Regarding the interaction variable, we decided to not include it in the model because it gave us a lower r-squared value compared to the final model and is a lot more complex.

**LDA:** LDA had many outliers in the box plot. An Ln was taken to find a better representation of LDA, as the histograms for the LDA weren’t in the form of the normal bell curve. Also, there were many outliers even after removing the outliers in the shares. 

These graphs are the stats on the LDA without the transformation part. So through the ln, there is an expectation that the data spread would be better. It would make the box plots as “normal” as possible so that the statistical analysis results from this data become more valid.

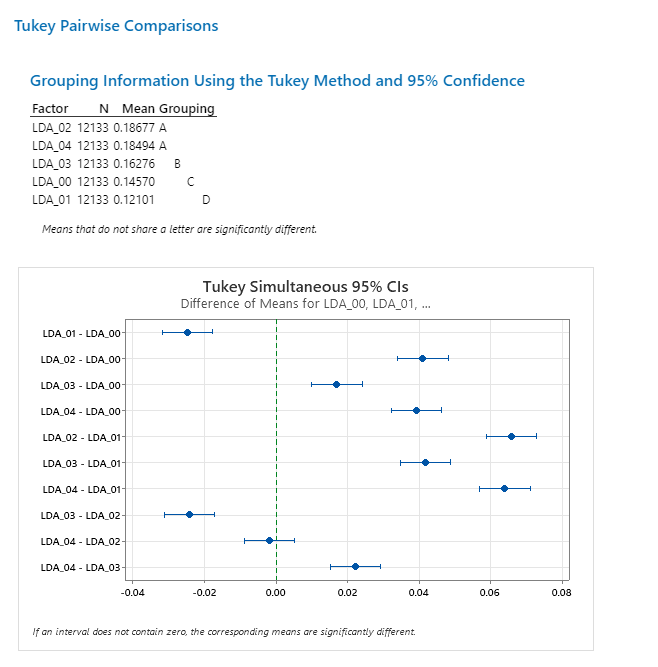


Based on Tukey Method - Hypothesis Test Method:

Ho: µ1 = µ2= µ3= µ4= µ5

Ha: at least one of the means is different from the others

With the p-value being less than 0.05 we reject the null hypothesis. So at the 5% significance level, we can say that at least one of the variables has different means.



The pairwise analysis resulted in LDA\_04 - LDA\_02 having a mean value of 0, indicating that these two means are similar. There is a slight relation between LDA 2 and LDA 4.

**Section 3: Model Building**

**Logistic Regression**

(See Appendix 2 )

Logistic regression is used to predict the categorical dependent variable given a set of predictor variables (features). For this purpose, we created a binary dummy y variable from our dependent variable ‘shares’ where if shares are 1500 and above, it is coded as ‘1’ and if it is below 1500, it is coded as ‘0’.

Our reference event, ‘1’ occurred 4,178 times whereas ‘0’ occurred 7,955 times. Looking at the p-values, we can observe that many variables like num\_imgs, and num\_videos have a high p-value.

The odd ratio for continuous predictors indicates that for every increase in that specific variable, the probability of an article being popular increases. One odd ratio that stands out among the continuous variables is global\_rate\_negative\_words. For every increase in the global\_rate\_negative\_words variable, the probability of an article being popular changes by a factor of 78. This means that the more negative words are included in the article, the more likely the article becomes popular. This can be explained by the notion that negative news gets more attention.

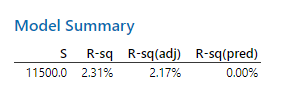
Among the categorical predictors for data channels and weekdays, the highest odd ratios indicate that an article published on Saturday has a higher chance of being popular than an article published on Monday. Similarly, an article has a higher chance of getting more shares if the genre is social media compared to articles where the genre is entertainment.

The goodness of fit test indicates that we have a good model as the p-value<0.05. This means that the data fits the model. The r-squared value is only 7.13%. But in logistic regression, the area under the ROC curve is a better metric as this metric evaluates the discriminatory power of the logistic regression model. It measures the ability of the model to distinguish between the two classes (e.g., binary outcomes). Higher values, closer to 1, indicate better discrimination. The area under ROC is 0.6815, which indicates a good discrimination between the binary outcomes.

**Regression**

Regression (Training Data Directly):

Before any preprocessing of the training data:



The obtained R-squared value of 2.31% indicates a relatively weak relationship between the variables under investigation. This finding is consistent with what has been reported in numerous other research papers. The low value can be attributed to several factors. Firstly, outliers within the dataset may have influenced the overall relationship. Secondly, establishing strong relationships between the variables can make it more challenging if the data exhibits high variability or does not follow a normal distribution. And lastly, there is no well-defined theory or understanding of the variables and the relation between the variables,

There was little preprocessing and manipulation needed to help increase the prediction of the regression model. The following steps:

Step 1: Removal of URL and timedelta variables. These are non-predictive variables.

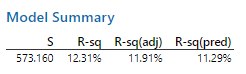
Step 2: Eliminate “Share” values exceeding a threshold of 3000. This step helped create a relatively normal distribution for shares and removed extreme values in millions.

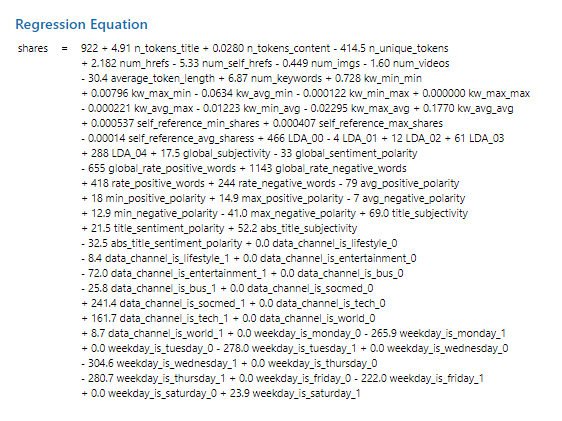
Step 3: Exploration of potential transformations to help spread the data. It was observed that LDA needed it the most and helped with normality.

Step 4: Remove variables "n\_non\_stop\_words" and "n\_non\_stop\_unique\_tokens" due to high correlation.

Step 5: Conduct an analysis of variance (ANOVA) to assess the equality of means and subsequent removal of variables based on correlation findings.

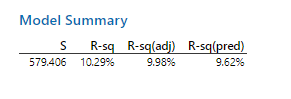
Regression (After these 5 steps):

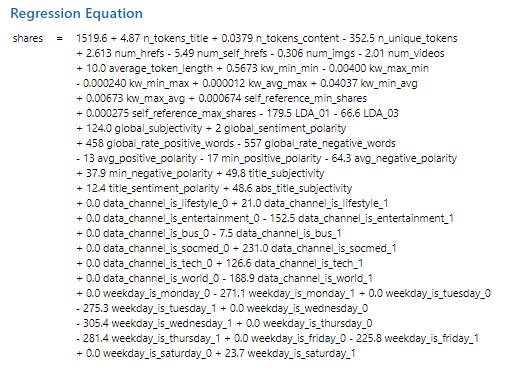




TRIAL 1: Removing correlated variables randomly is not a safe approach due to the potential loss of important information. However, if we systematically remove highly correlated variables, defined as those with a correlation coefficient above 0.7, the following variables were removed from the analysis: ['LDA\_00', 'LDA\_02', 'LDA\_04', 'abs\_title\_sentiment\_polarity', 'kw\_avg\_avg', 'kw\_avg\_min', 'kw\_max\_max', 'max\_positive\_polarity,' 'min\_negative\_polarity', 'rate\_negative\_words', 'rate\_positive\_words', 'self\_reference\_avg\_sharess']. By applying this criterion, we aimed to reduce multicollinearity and enhance the interpretability and stability of the regression model.

Regression (After TRIAL 1):





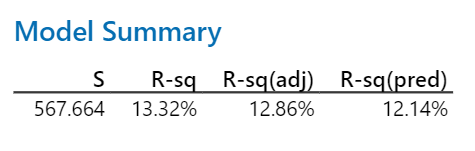
Regression (After TRIAL 2):

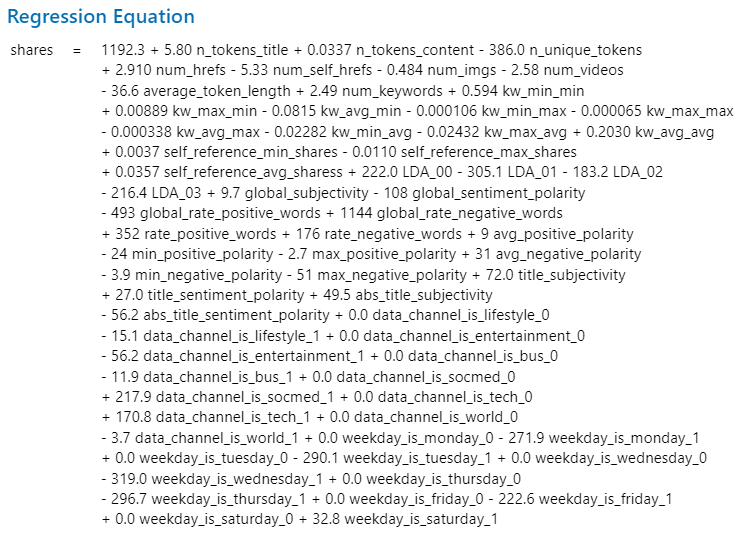
TRAIL 2: Every variable has an outlier, however, these three variables have the most.

df=df[df['self\_reference\_min\_shares'] < 10000]

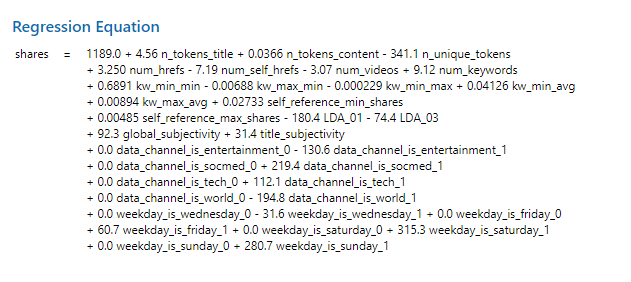
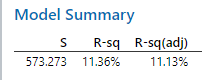
df=df[df['self\_reference\_max\_shares'] < 10000]

df=df[df['self\_reference\_avg\_shares'] < 10000]

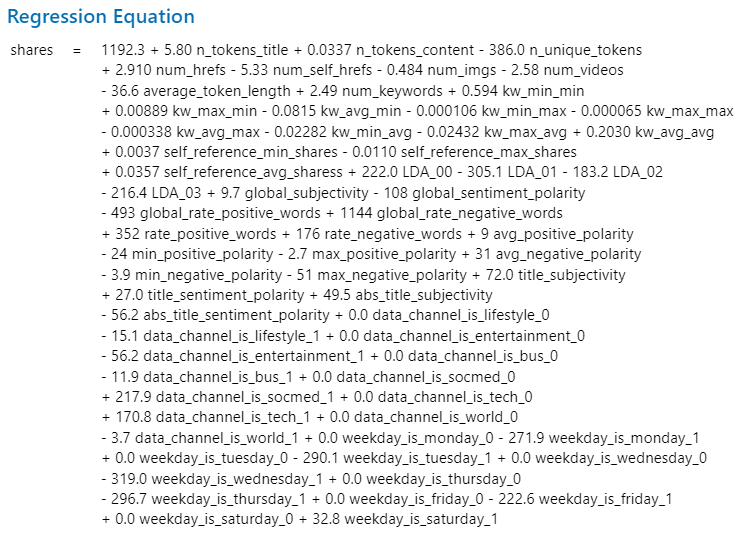




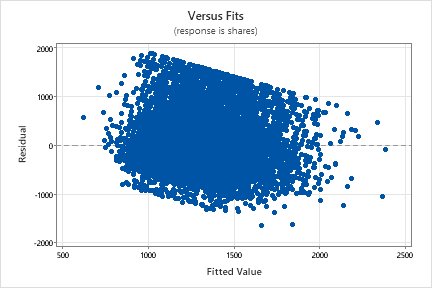
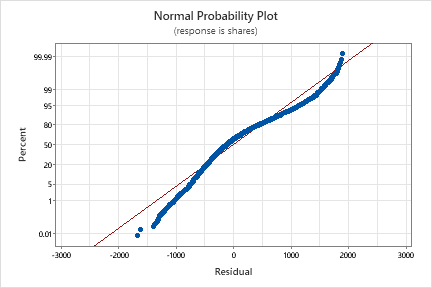
Regression (After combining TRAIL 1 AND TRAIL 2):



**Section 4: Model validation**

BEST MODEL: 

In this section, the final model built in Section 3 is applied to the testing data, and the accuracy of the prediction is calculated.

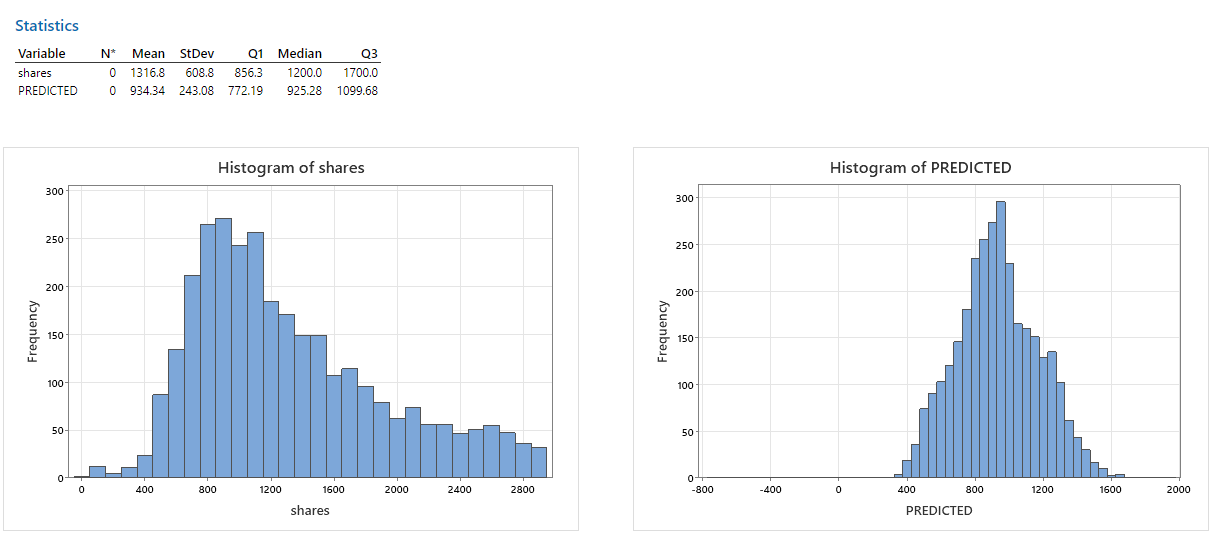


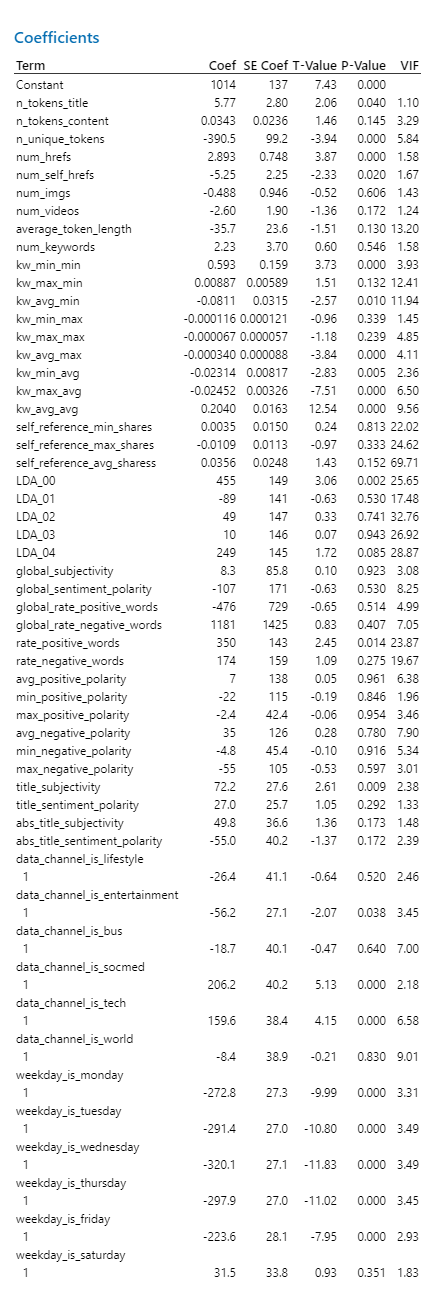
Constant variance: There is no equality of variance. Since the residuals on the scatterplot don't exhibit a relationship, they are normally distributed. It doesn't Violate the constant variance.

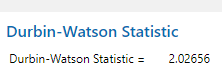
Normal: The graph resulted in a normal distribution. Additionally, zero means relatively distributing the data evenly above and below the zero line.

There are 2 outlier points in the normality graph.

The extra column represents the Predicted Values when compared to the actual values. As the model isnt the best in prediction, the values are similar. But this shows the distribution of the data when compared to predicted and actual.







**Section 5: Discussion and Conclusions**

The first step in section 2 is to use bar graphs to find any relationship between the categorical variables and their factors. We focused on transformations and saw LDA needed a transformation and shares required to get cut-off. We removed variables such as "n\_non\_stop\_words" and "n\_non\_stop\_unique\_tokens" due to high correlation.

As for the continuous variables, we used pie charts to demonstrate their behaviors and how commonly they are viewed. The next step was to generate the general descriptive statistics, which concluded that our data set contained many outliers and every variable resulted in it. So to analyze it, we found the variables that give the worst division of data which were Self\_references min, max, and avg. The matrix plot generated concluded that few variables show correlation. However, as we need to know the relationship between the variables and the definitions, removing them randomly isn't safe as we might lose data. Moving along, Minitab generated the Chi-squared tests, ANOVA, and 2-way ANOVA. However, there were only a few things that needed to be changed.

After some trial and error, we found that regression by removing absurd values of self-references and having a threshold of 3000 in shares gave the best R^2 value.

Difficulties:

* Lots of outliers in every predictor, and there was no trend between them
* The dataset creator should have mentioned how few of the variables were created. This was needed as it was difficult to know if the variables correlated because they had similar meanings.
* The transformation of predictors using Ln or central means didn't help with the dataset, so we still had many outliers.
* The number of observations was very high
* Even after removing correlated variables and p-values that are high, it didn't help with the R^2

Even though we were able to increase the r-squared value by approximately 10%, it remains relatively low. Therefore, we can conclude that there are several other factors that affect the number of shares that were not taken into account when compiling this dataset. That leaves room for further research and data gathering to find all other relevant variables.

Our primary obstacle throughout the project revolved around our dataset. We encountered unrealistic and illogical values for the predictor variables, which aligned with similar challenges faced by other researchers, as evident in online papers. Despite this challenge, the project served as an invaluable learning experience.

**Appendix: Variable Explanation**

Detailed explanation of each variable present in the dataset:

url: URL of the article (non-predictive)  
timedelta: Days between the article publication and the dataset acquisition (non-predictive)  
n\_tokens\_title: Number of words in the title - Quantitative  
n\_tokens\_content: Number of words in the content - Quantitative   
n\_unique\_tokens: Rate of unique words in the content - Quantitative  
n\_non\_stop\_words: Rate of non-stop words in the content - Quantitative  
n\_non\_stop\_unique\_tokens: Rate of unique non-stop words in the content - Quantitative  
num\_hrefs: Number of links - Quantitative  
num\_self\_hrefs: Number of links to other articles published by Mashable - Quantitative  
num\_imgs: Number of images - Quantitative  
num\_videos: Number of videos - Quantitative  
average\_token\_length: Average length of the words in the content - Quantitative  
num\_keywords: Number of keywords in the metadata - Quantitative

data\_channel\_is\_lifestyle: Is data channel 'Lifestyle'? - Categorical  
data\_channel\_is\_entertainment: Is data channel 'Entertainment'? - Categorical  
data\_channel\_is\_bus: Is data channel 'Business'? - Categorical  
data\_channel\_is\_socmed: Is data channel 'Social Media'? - Categorical  
data\_channel\_is\_tech: Is data channel 'Tech'? - Categorical  
data\_channel\_is\_world: Is data channel 'World'? - Categorical

kw\_min\_min: Worst keyword (min. shares) - Quantitative  
kw\_max\_min: Worst keyword (max. shares) - Quantitative  
kw\_avg\_min: Worst keyword (avg. shares) - Quantitative  
kw\_min\_max: Best keyword (min. shares) - Quantitative  
kw\_max\_max: Best keyword (max. shares) - Quantitative  
kw\_avg\_max: Best keyword (avg. shares) - Quantitative  
kw\_min\_avg: Avg. keyword (min. shares) - Quantitative  
kw\_max\_avg: Avg. keyword (max. shares) - Quantitative  
kw\_avg\_avg: Avg. keyword (avg. shares)- Quantitative

self\_reference\_min\_shares: Min. shares of referenced articles in Mashable - Quantitative  
self\_reference\_max\_shares: Max. shares of referenced articles in Mashable - Quantitative  
self\_reference\_avg\_sharess: Avg. shares of referenced articles in Mashable - Quantitative

weekday\_is\_monday: Was the article published on a Monday? - Categorical  
weekday\_is\_tuesday: Was the article published on a Tuesday? - Categorical  
weekday\_is\_wednesday: Was the article published on a Wednesday? - Categorical  
weekday\_is\_thursday: Was the article published on a Thursday? - Categorical  
weekday\_is\_friday: Was the article published on a Friday? - Categorical  
weekday\_is\_saturday: Was the article published on a Saturday? - Categorical  
weekday\_is\_sunday: Was the article published on a Sunday? - Categorical  
is\_weekend: Was the article published on the weekend? - Categorical

LDA\_00: Closeness to LDA topic 0  
LDA\_01: Closeness to LDA topic 1  
LDA\_02: Closeness to LDA topic 2  
LDA\_03: Closeness to LDA topic 3  
LDA\_04: Closeness to LDA topic 4

global\_subjectivity: Text subjectivity; Looks into how subjective the article is - Quantitative  
global\_sentiment\_polarity: Text sentiment polarity - Quantitative  
global\_rate\_positive\_words: Rate of positive words in the content - Quantitative  
global\_rate\_negative\_words: Rate of negative words in the content - Quantitative

rate\_positive\_words: Rate of positive words among non-neutral tokens - Quantitative  
rate\_negative\_words: Rate of negative words among non-neutral tokens - Quantitative  
Avg\_positive\_polarity: Avg. polarity of positive words - Quantitative  
min\_positive\_polarity: Min. polarity of positive words - Quantitative  
max\_positive\_polarity: Max. polarity of positive words - Quantitative  
avg\_negative\_polarity: Avg. polarity of negative words - Quantitative  
min\_negative\_polarity: Min. polarity of negative words - Quantitative  
max\_negative\_polarity: Max. polarity of negative words - Quantitative

title\_subjectivity: Title subjectivity - Quantitative  
title\_sentiment\_polarity: Title polarity – Quantitative  
abs\_title\_subjectivity: Absolute values for title\_subjectivity - Quantitative  
abs\_title\_sentiment\_polarity: Absolute values for title\_sentiment\_polarity - Quantitative  
shares: Number of shares (target) – Quantitative

**Appendix: R or Python codes used in the analysis**

**Appendix 1**

df.columns=df.columns.str.replace(" ","")

df=df.drop('url',axis=1)

df=df.drop('timedelta',axis=1)

df=df.drop('is\_weekend',axis=1)

df=df.drop('n\_non\_stop\_unique\_tokens',axis=1)

df=df.drop('n\_non\_stop\_words',axis=1)

import numpy as np

df['LDA\_00'] = np.log1p(df['LDA\_00'])

df['LDA\_01'] = np.log1p(df['LDA\_01'])

df['LDA\_02'] = np.log1p(df['LDA\_02'])

df['LDA\_03'] = np.log1p(df['LDA\_03'])

df['LDA\_04'] = np.log1p(df['LDA\_04'])

from sklearn.preprocessing import StandardScaler

object = StandardScaler()

object.fit\_transform(df)

import pandas as pd

from sklearn.model\_selection import train\_test\_split

train\_data, test\_data = train\_test\_split(df, test\_size=0.5, random\_state=42, shuffle=True)

# Display the shape of the new datasets

print('Training data shape:', train\_data.shape)

train\_data, test\_data = train\_test\_split(train\_data, test\_size=0.8, random\_state=42, shuffle=True)

#@title CHECK CORRELATION

import pandas as pd

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sns

#Using Pearson Correlation

corrmat = df.corr()

fig, ax = plt.subplots()

fig.set\_size\_inches(11,11)

sns.heatmap(corrmat)

def correlation(dataset, threshold):

col\_corr = set() # Set of all the names of correlated columns

corr\_matrix = dataset.corr()

for i in range(len(corr\_matrix.columns)):

for j in range(i):

if abs(corr\_matrix.iloc[i, j]) > threshold: # we are interested in absolute coeff value

colname = corr\_matrix.columns[i] # getting the name of column

col\_corr.add(colname)

return col\_corr

corr\_features = correlation(df, 0.7)

len(set(corr\_features))

# Remove rows with shares greater than 3000 (inplace)

df=df[df['shares'] < 3000]

import matplotlib.pyplot as plt

# Access the 'shares' column in the DataFrame

shares\_data = df['shares']

# Plotting the boxplot

plt.boxplot(shares\_data)

# Adding labels and title

plt.xlabel('Shares')

plt.ylabel('Number of Shares')

plt.title('Boxplot of Shares')

# Display the boxplot

plt.show()

import matplotlib.pyplot as plt

# Access the 'shares' column in the DataFrame

shares\_data = df['shares']

# Plotting the histogram

plt.hist(shares\_data, bins=10, edgecolor='black')

# Adding labels and title

plt.xlabel('Number of Shares')

plt.ylabel('Frequency')

plt.title('Distribution of Shares')

# Display the histogram

plt.show()

from sklearn.linear\_model import LinearRegression

from sklearn.feature\_selection import SelectFromModel

# Create your target variable column 'target\_column\_name' in the DataFrame 'df'

target = df['shares']

# Drop the target variable column from the DataFrame

features = df.drop('shares', axis=1)

# Create a linear regression model

reg\_model = LinearRegression()

# Perform feature selection

selector = SelectFromModel(reg\_model)

selector.fit(features, target)

# Get the ranked features

ranked\_features = features.columns[selector.get\_support()]

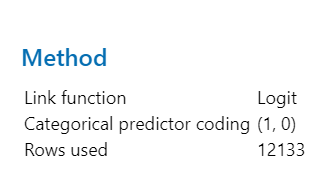
# Print the ranked features

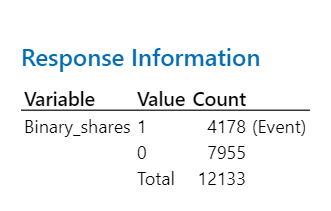
print("Ranked Features:")

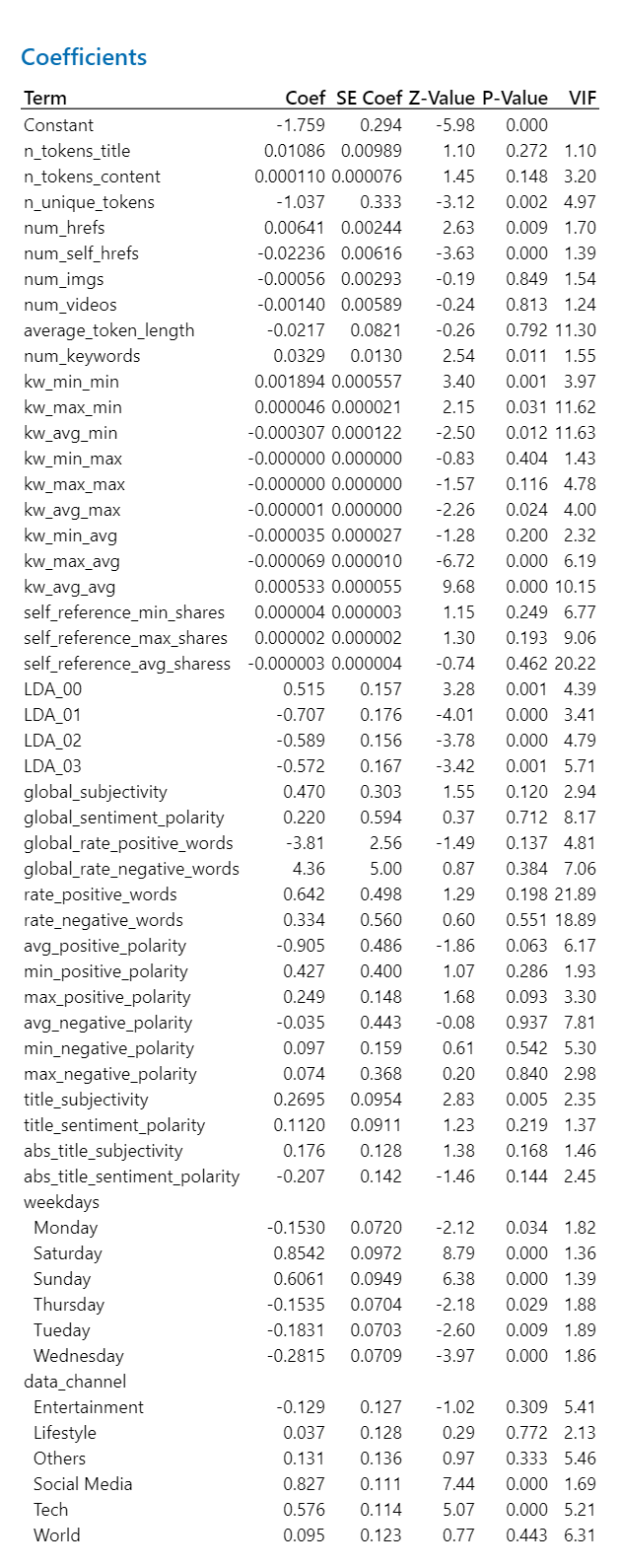
for feature in ranked\_features:

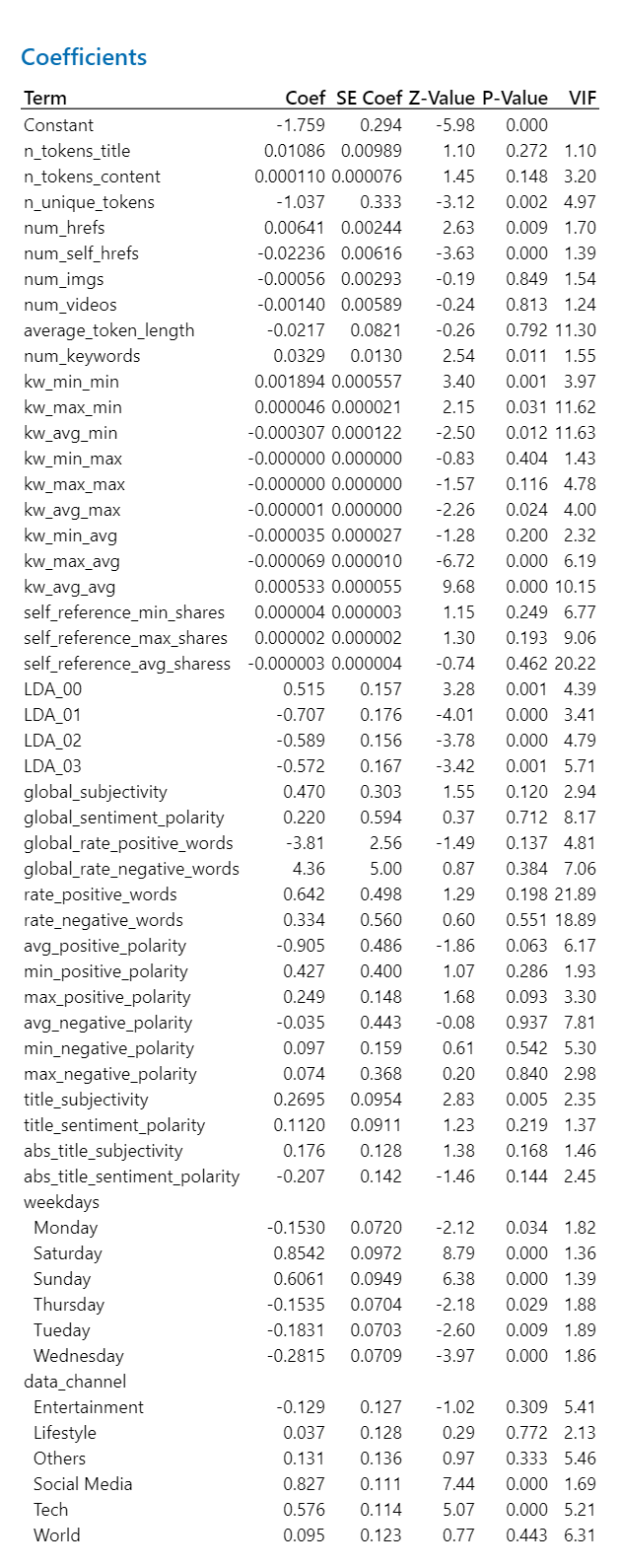
print(feature)

**Appendix 2: Logistic Regression**



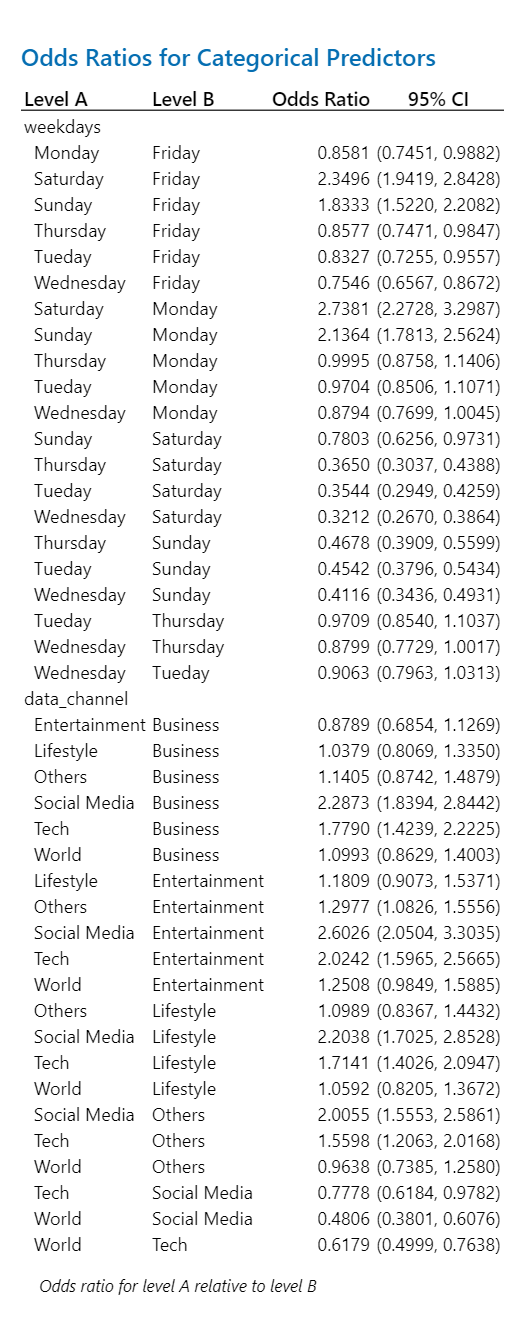


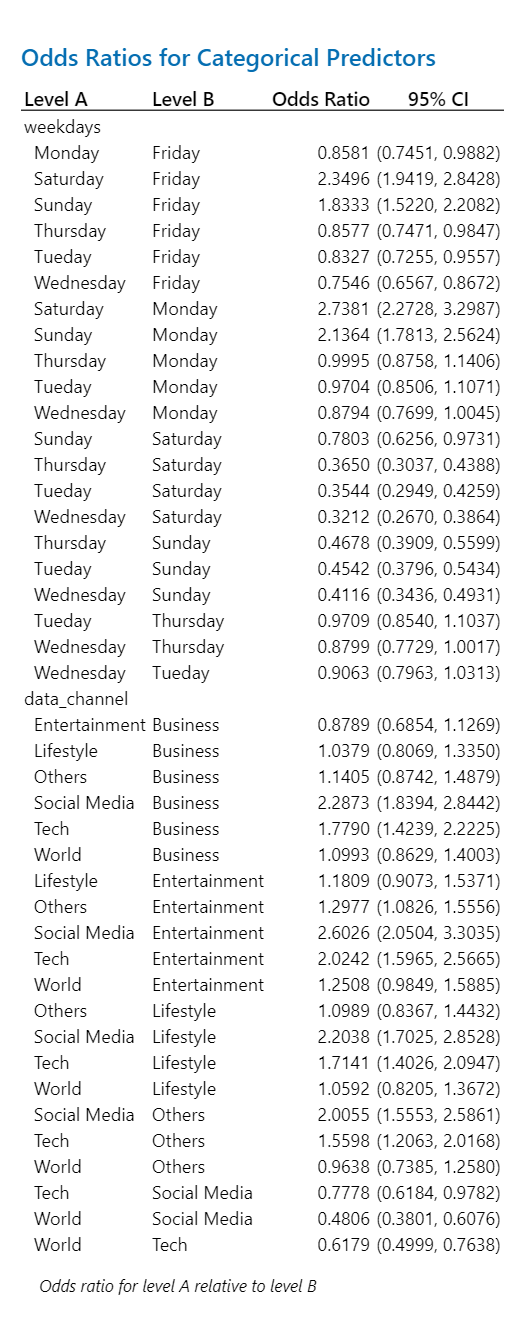


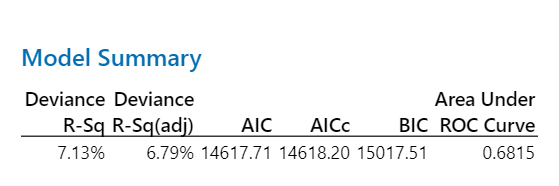


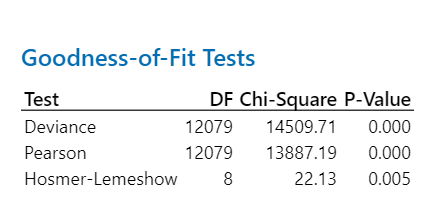


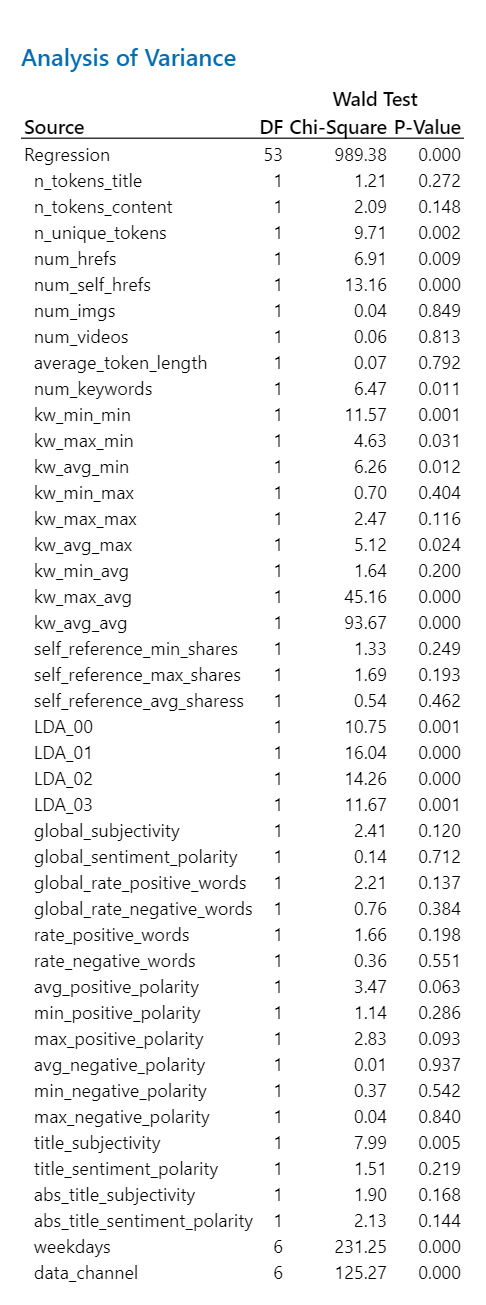
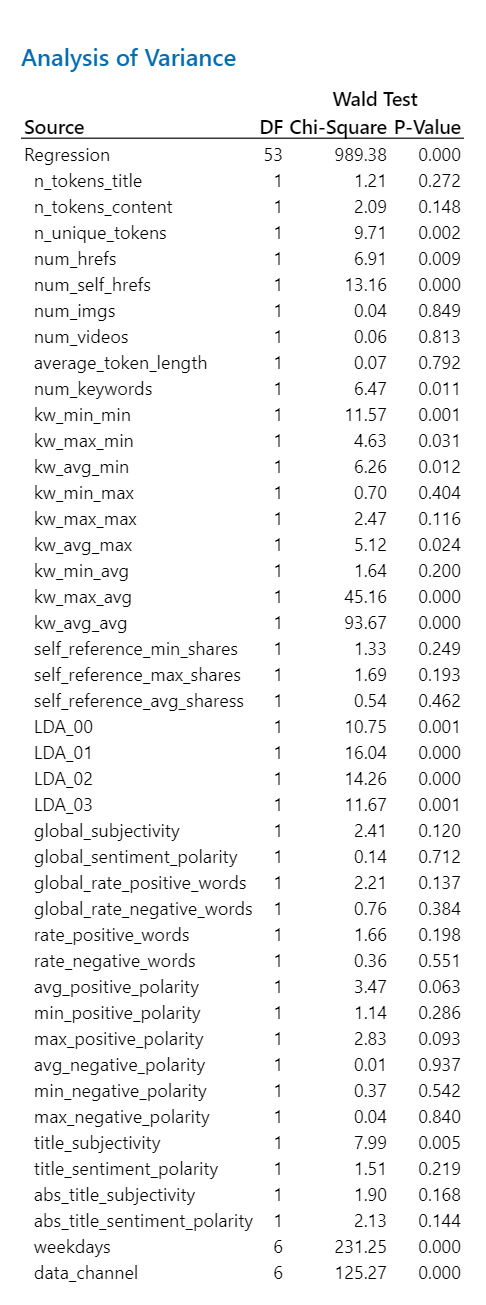








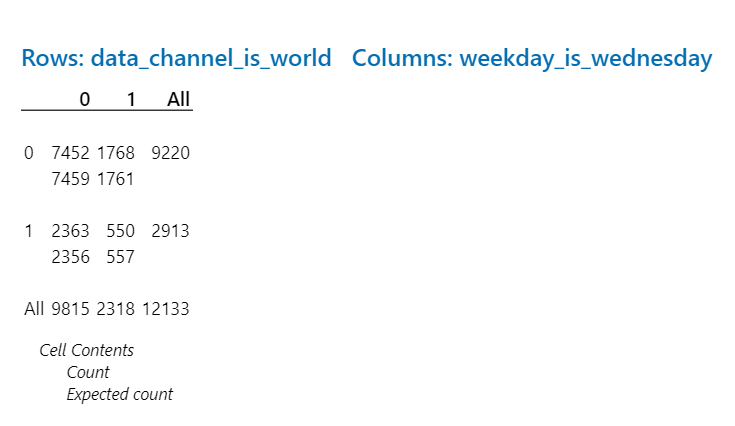


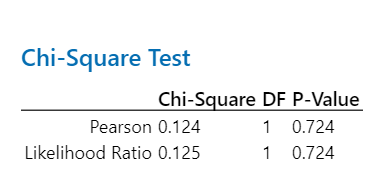


**Appendix 3: Chi-Square**

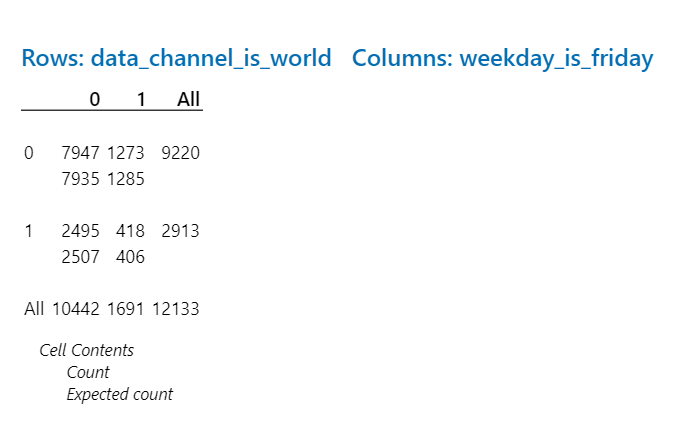
Some associations with the variables that produce high p values:

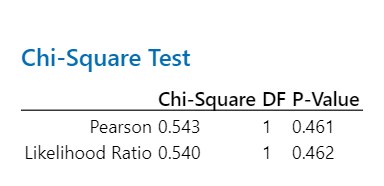
* World X Wednesday



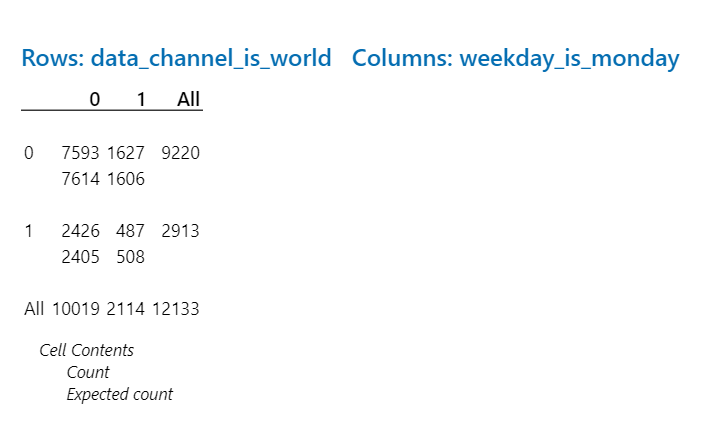


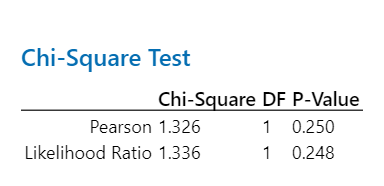
* World X Friday



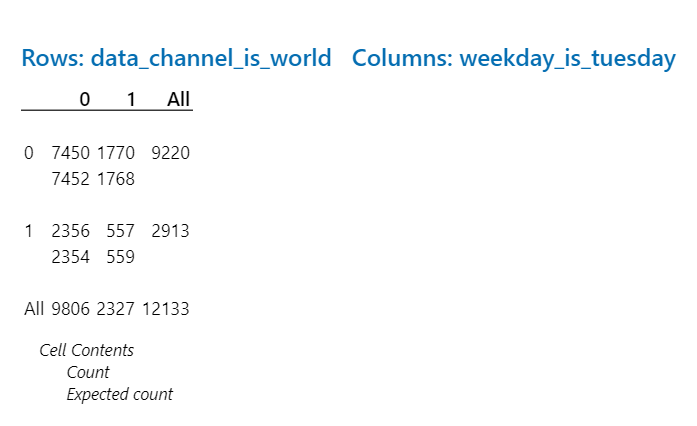


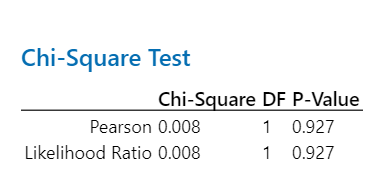
* World X Monday



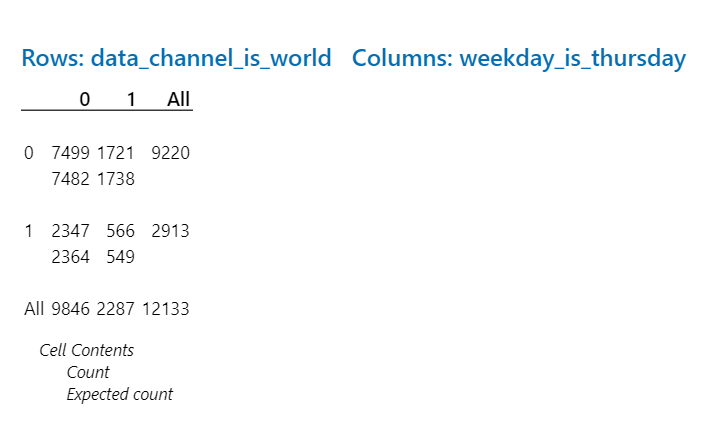


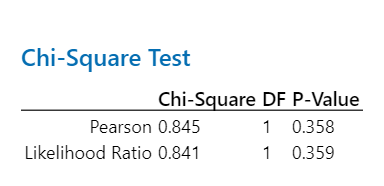
* World X Tuesday



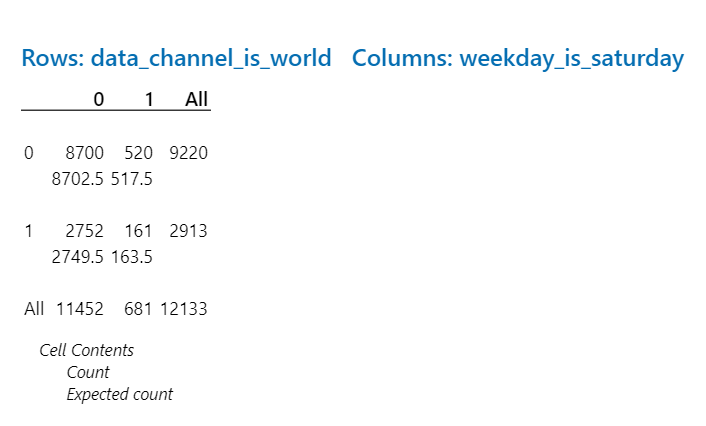


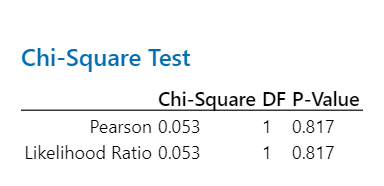
* World X Thursday



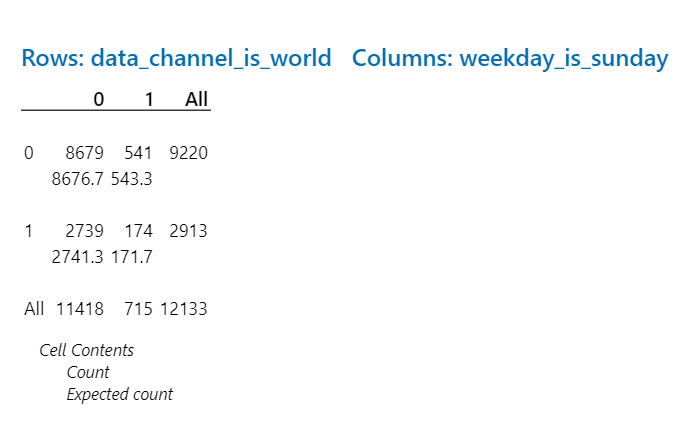


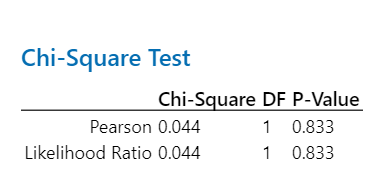
* World X Saturday





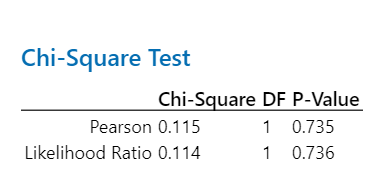
* World X Sunday



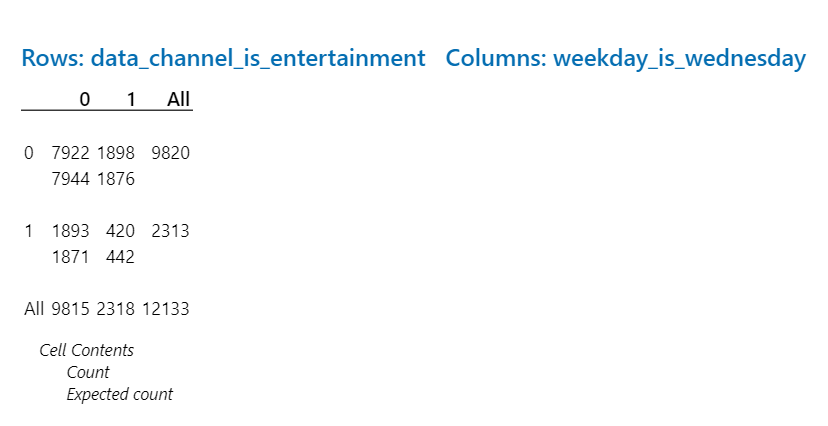


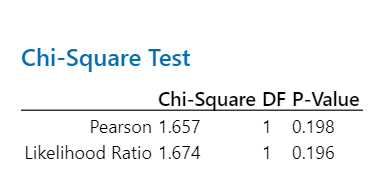
Wednesday X Data\_channel\_is lifestyle



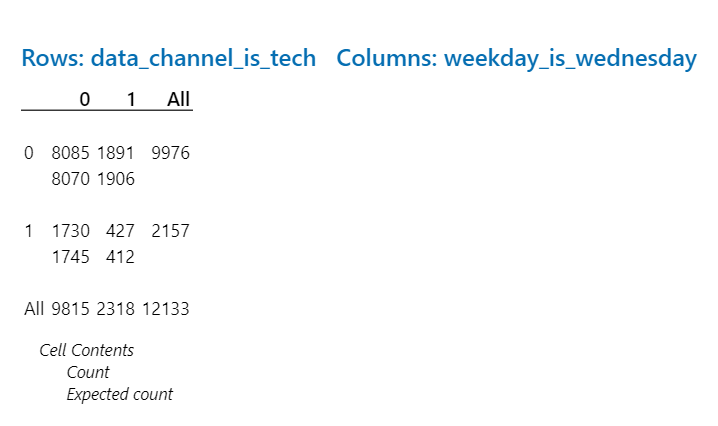


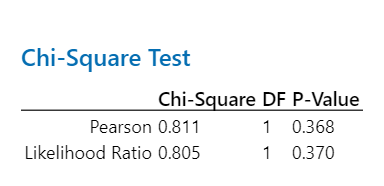
Wednesday X data\_channel\_is\_entertainment





Wednesday X Tech





Wednesday X Social Media



