1. **Preprocessing techniques**
2. Dropped any row that contains any blank values
3. Dropped any row that has videos that have errors or removed
4. Dropped any row that have videos with no name id (#NAME?)
5. Handled time format in ‘trending-date’ to be transformed to proper format ‘%y.%d.%m’ using pd.to\_datetime function
6. Handled date-time format in ‘publish\_time’ to be transformed to proper format ‘%Y-%m-%dT%H:%M:%S.%fZ’ using pd.to\_datetime function
7. Splitted ‘publish\_time’ into two columns to separate date and time in different columns
8. There were multiple columns that had string values and needed to be transformed using feature\_encoder

Columns:

* Video\_id
* Title
* Channel\_title
* Comments\_disabled
* Ratings\_disabled
* Video\_error\_or\_removd
* Tags

1. **Analysis on dataset and how features affect each other**
2. Made correlation matrix to help us in features selection using .corr() function

Graphical user interface, text

Description automatically generated

1. Get features with more than 50% correlation with likes using heatmap

Graphical user interface, application

Description automatically generated

1. We observed that ‘views’ and ‘comment\_count’ had the most correlation effect on likes
2. **Regression techniques used (final conclusion)**
3. Multiple Linear regression

* Features used:
  + Views
  + Comments\_count

Graphical user interface, text, application, email

Description automatically generated

1. Polynomial regression (approximate training time 45 ms)

* Features used:
  + Views
  + Comments\_count
* Final Polynomial degree: 6

Graphical user interface, text, application, email

Description automatically generated

* We tried different other polynomial degrees bigger and smaller than 6 so we can find the best polynomial degree
  + Degree 4

Graphical user interface, text, application, email

Description automatically generated

* + Degree 5

Graphical user interface, text, application, email

Description automatically generated

* + Degree 7

Graphical user interface, text, application, email

Description automatically generated

* + Degree 8

Graphical user interface, text, application, email

Description automatically generated

1. **Features used**
2. From the correlation matrix and heatmap we figured that views and comments\_count will have the best affect on our model
3. We tried to experiment with adding other features like channel\_title but it didn’t affect our linear regression much

Graphical user interface, text

Description automatically generated

1. We also tried adding tags but we didn’t observe huge difference without adding it

Text

Description automatically generated

1. **Size of our training and testing data**
2. Our dataset size after dropping values was 36980
3. We splitted our training and testing data to 70:30
4. **Improvements on our models**
5. We used normalization so we can decrease our mean square error.
   * Polynomial regression model before normalizing the features (Degree = 6)

Graphical user interface, text, application, email

Description automatically generated

* + Linear regression model before normalizing the output

Graphical user interface, text, application, email

Description automatically generated

* + Polynomial regression model before normalizing the output (Degree = 6)

Graphical user interface, text, application, email

Description automatically generated

* We noticed that after normalizing the features and output the MSE decreases for linear and polynomial regression models.
* We normalized our features and output between 0-100

1. **Conclusion**

• Features used based on the result of correlation matrix, heatmap and experimenting:

* + Views
  + Comments\_count
* Models:
  + Multiple linear regression
  + Polynomial Regression (degree = 6)
* The problem was how to predict the number of video likes, we used two regression models the first model was multiple linear regression (line) and the another one is polynomial regression (curve) but First we used normalization and data preprocessing to try to decrease our MSE.
* Our opinion is that the polynomial regression would be better and this opinion is proved with MSE which is equal to 1.5855 was less than the multiple linear regression model MSE which was equal to 2.3017