Report

Team ID: 20

Department	ID	Name
CS	20201700772	محمود طارق محمد البدري
CS	20201700781	محمود محمد محمد عبدالحميد
CS	20201700792	مروان احمد محمد علي
CS	20201700924	نصوح نبيل نصوح شوقل
CS	20201700719	محمد علي محمود محمد
CS	20201700071	احمد محمد جلال سليم

Processing Technique

1. **Prepare Dataset:**
    a. We drop the first three columns (URL, id, name) and column 5(icon URL) because it never affects the model.
    b. We drop subtitles and in-app purchase columns because it has the most null values.
    c. We fill Language column to make the data balanced and it have a few null values.
    d. Replaces the '+' character in the 'Age Rating' column with an empty string, and then converts the resulting strings to integers.
    e. Extract the year feature from columns (Original Release Date, Current Version Release Date) and replace them with original columns.
2. **Feature Scaling:**
    a. Scaling all features with StandardScaler except for column ‘Size’ scale it with MinMaxScaler.
    b. Reshape target and scale it with MinMaxScaler.
3. **Feature Encoding:**
    a. Encode columns (Developer, Primary Genre) with LableEncoder.
    b. Encode columns (Languages, Genres) by encode unique values then get sum of every single row.

Feature Selection

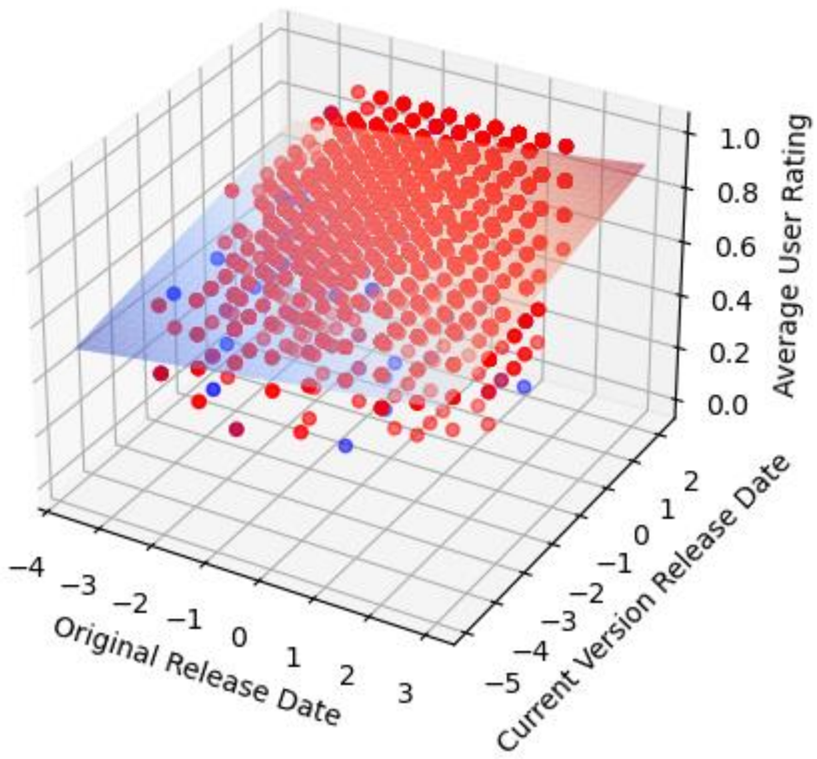
- Performs feature selection using the ANOVA F-test as the scoring function and the SelectKBest method from scikit-learn. The function first checks for constant features and removes them from the dataset. Then, it selects the top k features with the highest F-test scores.

Splitting Data

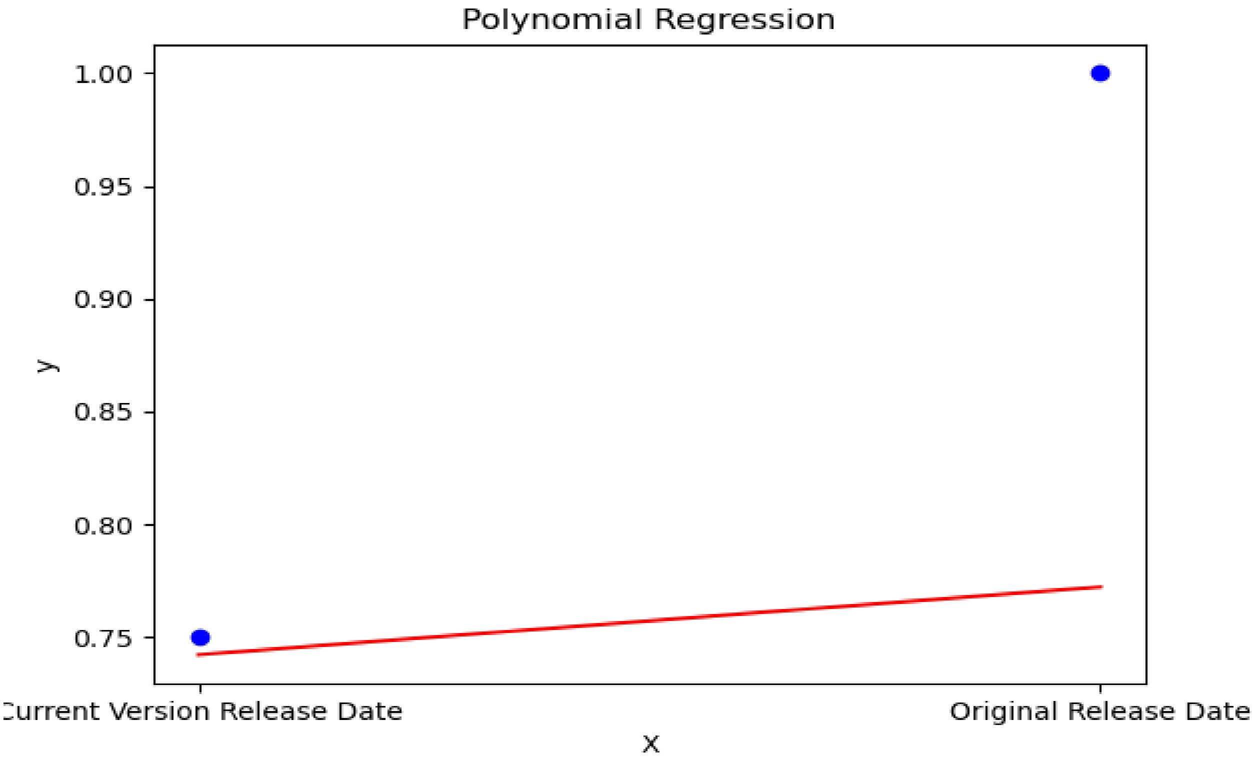
- Splitting the data into training and testing sets using the train_test_split to 20% testing set and 80% training set and shuffle data before splitting.

Regression Techniques

- **Multiple Regression:** performs a multiple linear regression on the input features X and output variable y using scikit-learn's LinearRegression () model. It then predicts the output values for X_test and returns the predicted values y_pred.

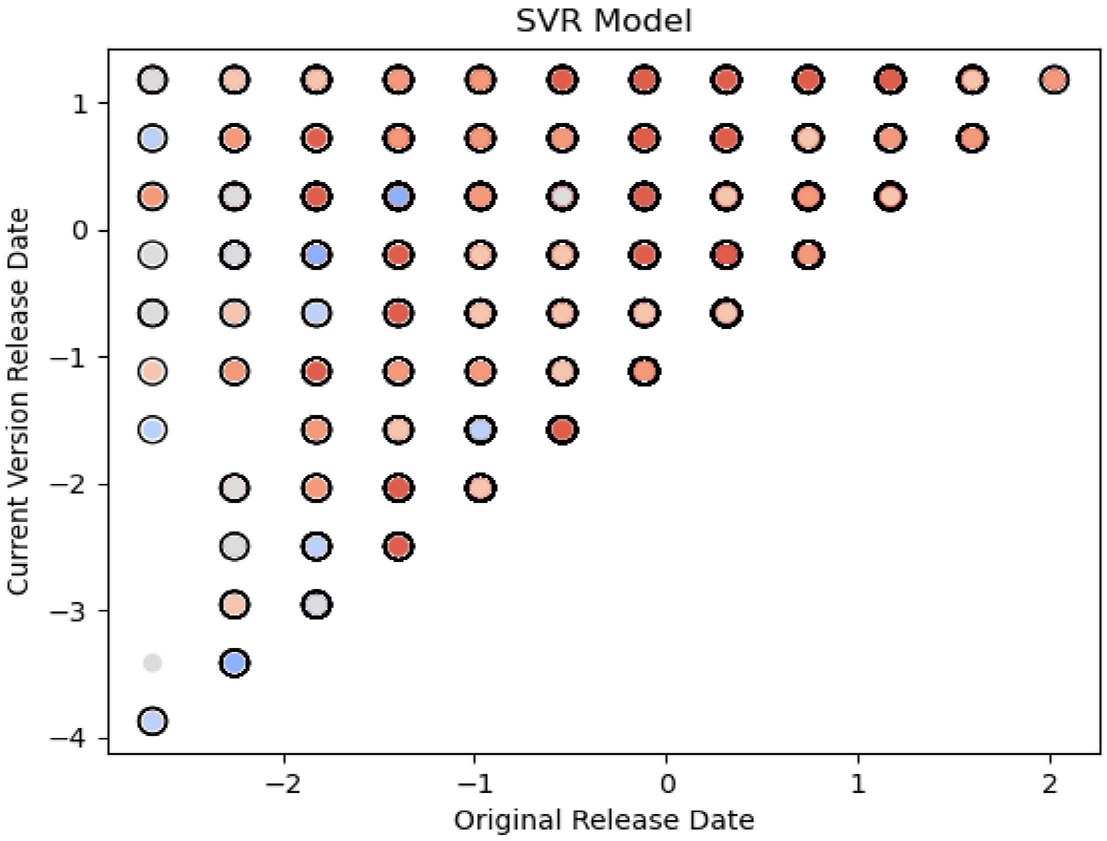


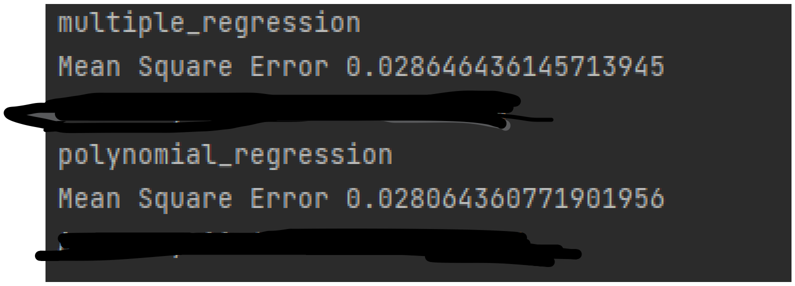
- **Polynomial Regression:** create polynomial features up to the specified degree. Then it fits a linear regression model to the training data with the polynomial features and makes predictions on the test data using the same polynomial features. The predicted values are returned



as y_pred.

- **Support Vetor Machine:** using the default hyperparameters of the SVR () class from scikit-learn. Given the training set X_train and target variable Y_train, it fits the model to the training data and returns the predicted values for the test set X_test.





```
multiple_regression
Mean Square Error 0.028646436145713945
[redacted]
polynomial_regression
Mean Square Error 0.028064360771901956
[redacted]
```

Conclusion

- Based on the mean squared error and accuracy scores, it seems that all three models performed similarly, with the polynomial regression model performing slightly better than the other two.