Predicting Mobile Game Success

MileStone1 and MileStone2

TEAM ID 19

Preprocessing:

Step (1) Feature Selection:

- Exclude columns that have unique strings.

Step (2) Data Cleaning:

1. Remove all rows with missing data in Label column.

Step (3) Data Cleaning:

2. Encoding for features: Give weights to strings.
    - Categorical Fields are split up into columns, in which each columns feature is represented by 1 for existence of this feature and 0 for non-existence of this feature.
    - Date Strings are encoded by the value of the date's year.

Step (4) Data Cleaning:

3. Empty cells are filled in each feature column by the average value represented by this column.

Step (5) Data Scaling:

- Scale all data to be ranged between (0, 1). Using Min-Max-Scalar.
- Min-Max-Scalar has proven higher efficiency than Standard-Scalar.
- Label column is not scaled, in order to keep a meaningful and a representing accuracy output.

Analysis:

Testing:

- Testing Size is equal to 20% of the dataset size.
- Training Size is equal to 80% of the dataset size.

Prediction

Label column: ‘Average User Rate’.
Label columns values ranges between 0 and 5.

Data Set:

Regression Techniques:

1. Multivariable Linear Regression.
   According to our dataset we have more than one factor that influences the ‘Average User Rate’ output. The model describes how a single response variable ‘Average User Rate’ depends linearly on several predictor variables.

2. Polynomial Linear Regression.
   Polynomial regression is about improving the model’s closeness to the data by increasing the order.

Model 1: Multivariable Linear Regression Run Output

Model 2: Polynomial Linear Regression Run Output

- Polynomial degree of 2:
- Polynomial degree of 3:

Conclusion:

Our intuition was that Polynomial linear regression would better, but it was disproved by the error of the output.

Multivariable linear regression has proven to be better than Polynomial linear regression according to the features that we have selected.

Classification

Label Column: 'Rate'.
Classifying between 3 classes: Low, Intermediate, High.
Classes are converted into numeric values: Low = 1, Intermediate = 2, High = 3.

Data Set:

Classification Techniques:

1. Logistic Regression.
2. SVM.
3. KNN.
4. Decision Tree.

Hyperparameters tuning:

	max_iter	C
Logistic Regression	**100**	**1**
	1000	**5**
	1000	**0.1**

	max_iter	kernel
SVM	**100**	**sigmoid**
	1000	**poly degree 3**
	1000	**'poly' degree 5**

	k_neighbors	weights
KNN	**10**	**uniform**
	20	**distance**
	30	**distance**

	creiterion	max_depth
Decision Tree	**Gini**	**10**
	Entropy	**10**
	Entropy	**50**

Dimensionality Reduction:

PCA	**n_components**
	2
	3
	5

Accuracy, time train, time test (Before PCA):

Accuracy, time train, time test (After PCA):

Conclusion:

Our intuition for Logistic Regression was proved to be a bad classifier for our data.

On the other side, our intuition for KNN was proved to be a good classifier for our data.

Our intuition for SVM to be a good classifier