Machine Learning

Ainura Zakirova
ai.zakirova@innopolis.university

1 Motivation

Cloud gaming is one of the most popular developing areas in the entertainment business. It has great advantages, allowing any user to enjoy some of the best games. But cloud gaming faces some challenges along the way of adoption. One of these problems is the latency. Of course, many factors affect the latency. These factors can be both on the part of the user (such as unstable Internet or an inability to handle the video stream) and on the part of the provider itself. As for the latter, the reasons may be different. Specifically, in our work, we considered the factors affecting the bitrate in the first task and the stream quality in the second task, which is directly related to the quality of the clouding games. For example, by building a regression model to predict the bitrate value, we can figure out what has a greater impact on the bitrate and what has a lesser one. Based on this information, we can predict what the next bitrate value will be. Evaluating the quality of a stream is no less important, since it is important for us to track at which characteristics the quality drops and on which characteristics it depends most. All this can help us improve the service; hence, it means attracting more users, which is one of the main goals.

2 Data

Regression task: we have nine features and predictor - bitrate. These features are **FPS**, **RTT**, **Dropped Frames**, **mean** and **standard deviation** of **bitrate**:
FPS - Frames per second is responsible for picture's smoothness;
RTT - Round-trip time is the amount of time it takes for a data packet to be sent to the server and for acknowledgement of that packet having been received by the client;
Dropped Frames are lost video frames in the process of data transfer.
The regression dataset is based by taking mean value and standard deviation for every feature, and also max for Dropped Frames.
Classification task: eleven features and target - stream quality (takes value 0 or 1). In this dataset, we have the same features as in regression dataset and some new ones:
FPS lag is the result of a general slowdown in graphics;
Auto bitrate state is a mode on how a bitrate is calculated;
Auto FEC is an Automatic Forward Error Correction mode.

3 Exploratory data analysis

In regression dataset, we have only numeric values and no missing values. All features have different scales, therefore it's necessary to apply a scaling method. *MinMAxScaler* was used for this purpose.
By building *pairplot* from *seaborn* library, the pairwise visualization for every feature was shown. The correlation between the majority of features with target is small (< 0.1). The features "bitrate_mean", "bitrate_std" and "fps_mean" have the highest correlation with target features.
In classification task, we have two categorical features, which we encoded to transform them to numeric type by using *LabelEncoder*. Also, the dataset doesn't have any missing values. As features have different scales, *MinMAxScaler* was applied.
Classification task requires more preprocessing, as it faces with the imbalanced problem. A special method was used to solve this problem (Section 6).

4 Task

We are faced with two tasks: regression and classification. The objective of these tasks is to estimate such function f, which will define the relationship between features and target in the best way:

$$f : X \longrightarrow Y$$

And we want the loss to be as small as possible, that is:

$$AVG[(y - f)^2] \longrightarrow min$$

4.1 Regression

In regression task we applied different regression models, but firstly, we used feature selection methods, such as Lasso and SelectKBest to remain only meaningful features. Lasso model assigned zero weights to almost all features except of three: "bitrate_mean", "bitrate_std", and "fps_mean". The same result we got by using SelectKBest method, so we remain only these three features as our dataset.
Further, four regression models were applied: Linear regression model, Lasso regression, Ridge regression and Polynomial regression.
As performance measurement metrics, Root Mean Squared Error(RMSE), Mean Absolute Error (MAE), and R^2-score were used.

4.2 Classification

In Classification, feature selection was also applied by using Lasso model. It removed three features, and we kept working only with eight remaining. Then a logistic regression model with different parameter "penalty" was tested. 'Penalty' parameter defines different types of regularization, namely L1, L2.

As performance measurement metrics Accuracy, Precision, Recall and F1-Score were chosen.

5 Results

The results of models for regression task on test data is represented in Table 1.:

Table 1. Regression models on Test set

Model	RMSE	MAE	R^2-score
Linear regr.	1949.7	1076.9	0.893
Lasso	1949.6	1077.7	0.893
Ridge	1949.7	1077.1	0.893
Polynomial(d = 4)	1947.3	1051.9	0.894

All models are pretty similar in their results, but Polynomial regression with degree 4 is better than others.
The results of models on train set is represented in Table 2.:

Table 2. Regression models on Train set

Model	RMSE	MAE	R^2-score
Linear regr.	1975.9	1097	0.894
Lasso	1977.6	1097.9	0.894
Ridge	1977.6	1097.2	0.894
Polynomial(d = 4)	1959.9	1064.6	0.896

Our results for Train and Test set are not much different, what means that we deal with good models, which are neither underfit nor overfit. To avoid overfitting or underfitting we should choose the model properly. In our case, a more complex model gives better performance, because our task is not simple.
Also, we applied cross-validation on training data to compare models on average metrics.
Classification was done on data with outliers and without it. In Table 3. we can see the results of models with outliers on test dataset:

Table 3. Classification models on Test set

Model	Acc.	Recall	Precision	F1-score
Log.(pen.='none')	0.94	0.136	0.704	0.228
Log.(pen.='L1')	0.94	0.132	0.708	0.222
Log.(pen.='L2')	0.94	0.130	0.725	0.221

Table 4. represents the results of classification model on training dataset. In the classification task, we used regularization methods to prevent overfitting. And comparing with test set, difference between metrics is insignificant. Thus, our classifications models don't overfit or underfit.

Table 4. Classification models on Train set

Model	Acc.	Recall	Precision	F1-score
Log.(pen.='none')	0.945	0.240	0.835	0.370
Log.(pen.='L1')	0.945	0.239	0.84	0.373
Log.(pen.='L2')	0.945	0.235	0.850	0.368

F1-Score is mostly used for comparing models, because it considers the both Precision and Recall at the same time. Then the best model is logistic regression without regularization. But also we have to define, what metric is more important in our particular context to improve it further.
Precision can be considered as more important metric with respect to stream quality, therefore logistic regression with L2 regularization is the best regarding Precision.
In Table 5. is represented comparison between logistic regression with and without outliers:

Table 5. Comparison, based on outliers

Model	Acc.	Recall	Precision	F1-score
Log. with outliers	0.94	0.133	0.708	0.223
Log. without outliers	0.94	0.139	0.693	0.233

We can notice, that F1-Score and Recall are increased, while Precision is decreased, because we cannot increase Precision and Recall simultaneously, we should find some tradeoff.

6 Data Imbalance

Classification task data is imbalanced. We have 7% of one class and 93% of another class.
In this work we used undersampling method - RandomUnderSampler and obtained the results represented in Table 6.

Table 6. Undersampling model

Model	Acc.	Recall	Precision	F1-score
Logistic	0.883	0.506	0.276	0.357

Although F1-Score is increased, but Precision is decreased, which is unwanted for our task.

7 Conclusion

Bitrate and stream quality are very important aspects of cloud gaming. Therefore, obtained results could be useful from the service improvement point of view.