Assignment 1

Mark Vishnevskiy
m.vishnevskiy@innopolis.university

1 Motivation

These days, the gaming industry is becoming more and more popular, and top IT companies are trying to take their place in this industry, as many users need quality service. Thus, the problem of optimization and analysis of transmitted data is becoming increasingly important, as it can significantly improve the quality of services provided. Cloud gaming, unlike other activities on the Internet, requires a flawless connection, since data transfer is carried out both ways, and even a small delay can ruin the user experience. Thus, the prediction of connection quality based on known data will help service providers to better understand existing problems and improve the quality of services provided.

2 Data

We have some features that demonstrate user's settings and connection quality:
RTT: Round-trip time is the amount of time it takes for a data packet to be sent to the server, plus the amount of time it takes for acknowledgement of that packet having been received by the client
Dropped Frames: Lost video frames in the process of data transfer. Dropped Frames indicates that the connection to the remote server isn't stable, or you can't keep up with your set bitrate. If too many frames are dropped, the client may be disconnected from the streaming server
FPS: Frames per second is the frequency of individual images that are displayed on a video device per second. FPS can be described as the picture's smoothness – the higher the FPS, the smoother the final picture
Adaptive bitrate: it is a mode on how a bitrate is calculated. (categorical feature)
Auto FEC: Automatic Forward Error Correction mode (categorical feature)
Classification target: the stream target to be predicted
Regression target: the bitrate to be predicted

3 Exploratory data analysis

The dataset for the classification problem contains some categorical features. Those features are easily converted by `OneHotEncoder`, but I decided to add an ordinal encoding column for `auto_bitrate_state` since its values can be represented in ascending order. With the help of the pandas profiler, the characteristics of the features were obtained. In particular, many data have numerous outliers that need to be carefully removed. At the same time, the volume of the positive class in the classification is so small that the standard function considers it to be an outlier, so it is necessary to first separate the features and the target. To obtain the correct metrics, it is necessary not to normalize the target accidentally in the regression task. Normalization was carried out with the usual `MinMaxScaler` since after removing the outliers, the data fit well. It is also important to note the need to transform the test data on encoders trained on the train set.
The regression task did not require a large amount of data preprocessing, but removing some features and reducing the dimension made it possible to display a graph that clearly demonstrates the linear dependence of the target on the parameters.
The task of classification required more careful preparation of the data. In particular, an increase in the size of the minor class made it possible to significantly (up to three times) increase the recall metric. Removing uninteresting features and lowering the dimension made it possible to plot a three-dimensional graph, showing the almost complete absence of dependence.

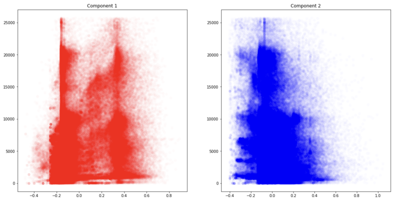

Figure 1. Regression visualization

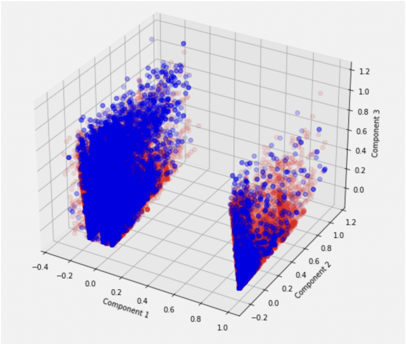

Figure 2. Classification visualization with PCA and three components

Figure 2 shows the distribution of the target variable for classification task over the dataset, processed by PCA algorithm with number of components equal to three.

4 Task

4.1 Regression

To solve the regression task, I used the following algorithms: Linear Regression, Linear Regression with Ridge (L2) regularization and Bayesian Ridge.

Linear regression is a parametric and supervised machine learning approach, which approximates the target variable as a linear combination of features.

One of the disadvantage of classical linear regression is the tendency to have imbalanced values of weights, which can lead to overfitting. Due to this fact, the regularization techniques are widely applied to machine learning. There are two of the most common ways to for regularization - Lasso (L1) and Ridge (L2) regularization. L1 regularization minimizes the sum of the first normal form of the weights, while L2 uses the sum of squares. From these two options, I have selected the L2 regularization, because the number of features in the dataset as well, as the number of weights is relatively low.

For the third algorithm for regression task, I had a choice between Support Vector Regression (SVR) and Bayesian approach. The SVR algorithms requires significantly more computation resources. Therefore, I have decided to use the well-known statistical approach for data analysis - Bayesian methods, in particular — Bayesian Ridge. The Bayesian techniques are based on the Bayes rule from the probability theory:

$$\begin{cases} p(y_{new} = c | x_{new}, X, y) = \frac{p(x_{new}|c) \cdot p(c)}{p(x_{new})} \\ p(x_{new}) = \sum_{c=1}^{C} p(x_{new}|c) p(c) \end{cases} \tag{1}$$

where $p(y_{new} = c | x_{new}, X, y)$ is a posterior class probability, $p(x_{new}|c)$ is the likelihood, $p(c)$ is the prior class probability, and $p(x_{new})$ is the marginal likelihood. This theorem is the base for `BayesianRidge` and `GaussianNB`, which I used in this assignment.

4.2 Classification

For the classification task, I have chosen Logistic regression algorithm with L1 and L2 regularization. Additionally, I have also applied the `GaussianNB` classifier.

Logistic regression is an extension of the linear regression algorithm for classification. This approach separates the classes with sigmoid-like function. The `GaussianNB` was described in Subsection 4.1.

5 Results

The regression metrics may seem dramatically large at first, but detailed analysis changes the view. The peculiarity of the regression dataset is that the target value varies from 0 to approximately, 30000 (after the removed outliers). Thus, the average absolute error is only about 13%, which is a pretty good result. It is also important to note that the difference between the metrics for the training and test set is negligible. If the training MAE was significantly larger, this would indicate overfitting, but I avoided this problem by properly preparing the data.

To describe classification metrics, it is necessary to understand what is Precision and Recall: Precision is the fraction of relevant instances among the retrieved instances, while recall is the fraction of relevant instances that were retrieved.

The values obtained as a result of the classification indicate that there is a trade-off between precision and recall, which means that we cannot raise both metrics, and we have to select one important metric based on context, and then adjust it by augmenting the dataset.

Table 1. Regression test score

Score	LinReg	Ridge	BayesianRidge
MAE	4485.921	4483.772	4484.983
MSE	4.077e+07	3.869e+07	3.884e+07
RMSE	6385.112	6220.069	6231.96
Root Squared	0.378506	0.291424	0.298463

Table 2. Regression train score

Score	LinReg	Ridge	BayesianRidge
MAE	4251.013	4251.031	4251.026
MSE	2.795e+07	2.795e+07	2.795e+07
RMSE	5286.895	5286.912	5286.910
Root Squared	0.254703	0.254691	0.254692

Table 3. Classification test score

Score	LogReg L1	LogReg L2	NaiveBaies
Accuracy	0.919748	0.919416	0.895938
Precision	0.381070	0.379338	0.309092
Recall	0.393526	0.394227	0.498025

Table 4. Classification train score

Score	LogReg L1	LogReg L2	NaiveBaies
Accuracy	0.844624	0.844299	0.840273
Precision	0.647508	0.644196	0.556261
Recall	0.208410	0.207883	0.344118

6 Conclusion

Summarizing the work done, I gained an invaluable experience in analyzing and preparing real data for classification and regression problems, and also learned how to work with metrics and process the results.