# Machine Learning: In-depth Predictive Analysis Report

Machine learning is a very powerful way to analyze data in order to perform predictions or to discover hidden insights. It allows for the production of models that fit to a dataset and generalizes to new unseen data. Machine learning can be used in regression analysis or classification problems. During the inferential statistics portion of this capstone project a simple linear regression analysis was performed on a video game's global sales value based on its critic score. We will take a look at using machine learning models to feed additional features to it and increase its predictive power as well as framing the problem in a classification context.

In order to feed additional features into the model the features first need to be transformed into a format that the scikit-learn library accepts. This means that we will need to encode the categorical features. Some of them will also be turned into dummy variables through a method called one hot encoding. One hot encoding is a way to prevent bias that comes from ordinal values formed from the encoding of categorical features. This turns the category values into additional features that have binary values indicating whether or not the record contains that category. We also need to be sure not to add too many features to the model. Having too many features, or dimensions, may cause what is known as the curse of dimensionality which can lead to various problems that occur in analyzing data in a high dimensional space.

Looking at the linear regression model first, we added three new columns: log of global sales, count of publishers, and pub class. The log of global sales was added because it is generally advisable to take the log whenever you are working with money. One of the features of a video game is the publisher. There is a larger amount of publishers in this dataset and performing one hot encoding on them could potentially lead to the curse of dimensionality. In order to circumvent that situation, we could create a new column that categorizes a video game based on the amount of times the publisher appears in the dataset. This is where the count of publishers column comes in. We gave a video game a class of 0 if its publisher had a count less than 50, a class of 1 if the count was between 50-149, a class of 2 for counts between 150-199, and a class of 3 for 200+. Now that we have a more usable representation of the publisher class we can create a “lean” dataset that only contains the features that will be fed into the model as well as the target variable.

With a clean dataset we can now perform machine learning in order to predict the log of global sales for a video game. This was done by splitting the dataset into a training set and a hold-out set, cross-validating the model, fitting it on the training set, predicting on the testing set, and then calculating metrics for the model performance. The reason why the dataset is split into a training set and a hold-out set is because we want to fit the model to the training set and then use the hold-out set to see how well the model performs to unseen data. Cross-validation was used to observe how the model would fit to different samples of the training data. The performance metrics that were calculated for the linear regression model were the adjusted R-squared and the root mean squared error values. The adjusted R-squared value is the R-squared value that takes into account the addition of more features. The root mean squared error is a value that is analogous to the standard deviation of the predicted values.

The cross-validation had produced 5 values for the adjusted R-squared that ranged from 0.28-0.32. This indicates that the model had a stable performance for different sections of the dataset. After training the entire model on the training set and testing it with the hold-out set the adjusted R-squared and the root mean squared error(RMSE) values were calculated to be

0.299 and 1.2769, respectively. For the simple linear regression model of critic scores and log global sales, we had an R-squared value of about 17%. Introducing the additional features of a video game increased the R-squared value by about 13%. The range of values for log of global sales ranged from about -4 to 4. The calculated RMSE value shows that the errors of the prediction were on average about 1.28, which is a relatively large error given the range. This indicates that the linear regression model may not be able to predict the global sales of a video game very accurately. It seems that the linear regression model has room for improvement. Let's explore the prediction of global sales in the light of classification.

In order to turn this prediction problem into classification the dataset would need to be labeled. To keep it simple, each video game was given a binary label of whether or not it was a high selling video game. The range of values for the global sales was between 0.01-82 million units sold. It was decided that any video game that sold more than 1 million units would be high selling. The dataset for this classification problem will be very similar to the one used in the linear regression model. However, the target variable will be the high selling labels instead of the log of global sales. Some additional features, such as user scores and count of games per publisher, were also added to the dataset.

Just like with the linear regression model the categorical features, as well as the target variable, were encoded and the dataset was split into a testing and a hold-out set. The testing set was used in some initial classifier modeling as well as hyperparameter tuning through grid search cross validation. Grid search cross validation is a method of obtaining the most optimal hyperparameters for a machine learning model by fitting a model with various sets of hyperparameters and then cross validating it to produce a score. The hyperparameters that produce the highest scores were chosen as the most optimal hyperparameters for the model, given the dataset.

The first classifier model that was looked at was logistic regression. Without tuning any of the hyperparameters and fitting and testing it with the training set, it was able to produce an accuracy score of about 79%, out-of-the-box. The accuracy of a classifier model shows the percentage of predictions the model correctly produced. A few other models such as random forest and SVM were also ran out-of-the-box and produced scores of 79% and 76%, respectively. The logistic regression, SVM , and random forest classifiers had their hyperparameters optimized and scored 79%, 79%, and 80%, respectively. From these results it was seen that logistic regression and random forest did not change very much, but the SVM classifier was brought up to the same level as the other two in terms of accuracy score.

One down side to accuracy is that it is not robust to unbalanced data. For example, say we were trying to classify whether or not a plane would crash. There are far more occurrences of planes not crashing than crashing. Let's just use an arbitrary ratio of 1 crash for every 100 flights. Even if the classifier misclassified that 1 crash, the accuracy score would be 99%, which is not very representative of the data. Therefore some other metrics that takes this situation into account are precision, recall, and f-1. The precision of a model represents how many of the predicted positives are actually true. The recall represents the proportion of actual positives being predicted correctly. There will always be a trade-off between precision and recall. The f-1 score is the harmonic mean of precision and recall. Which metric to use for scoring the performance of a classifier depends on the dataset and the problem being solved.

For this problem of classifying video games as being high sales or not, it was determined that the precision would be the most suitable metric for scoring the model. Using the best hyperparameters obtained from the grid search cross validation as well as the testing set for scoring, it was seen that all three models performed very similarly to each other. But it looks like the random forest model scored a 73% in predicting the positive class correctly and outscored both the logistic regression and SVM models by about 2%. One final way to judge how well a model performed is by looking at an ROC curve.

A Receiver Operator Characteristic (ROC) curve is a way to visualize the performance of a binary classifier model. It looks at the ratio of true positives and false negatives given varying threshold values. A threshold determines whether or not the classifier will classify a record as a certain class, given its probability of being that class. The model curve that is closer to the upper left portion of the plot indicates a good classifier. A curve that is towards the diagonal middle of the plot indicates that the classifier does no better than randomly guessing the class. The area under the curve (AUC) is a way to represent these curves numerically and allows for comparisons of performance between different models.

After plotting the curve for the three classifiers the AUCs were calculated to be 0.81, 0.80, and 0.84 for logistic regression, SVM, and random forest, respectively. Given all the results that have been observed, it looks as though the best classifier for prediction given this dataset is the random forest model. The random forest classifier tied in best out-of-box performance, scored the highest after tuning the hyperparameters, had the highest precision in classifying the positive class, and had the highest AUC for the ROC curve. The logistic regression model had also performed well for this dataset and was a strong contender for the best classifier model.

Machine learning is an extremely powerful tool in providing predictive and analytical solutions to problems in data. Given the endless types and amount of data, many machine learning algorithms have been developed in order to tackle these problems. Some of these algorithms were applied to this dataset of video games in order to predict how well a video game would perform in sales, globally. It was seen that by applying machine learning and performing linear regression on many different features of a video game, the R-squared value was able to be increased from the 17% seen in the simple linear regression of critic scores and log of global sales to 30%. However, the root mean squared value of 1.28 shows that the linear regression model has room for improvement. It was also seen that by converting this problem into a classification one, a video game was able to predicted as having a global sales value larger than 1 million units with an accuracy of about 79%. The classifier that produced the best model was random forest which had a precision of about 73% and an AUC of about 84%.