UDACITY MACHINE LEARNING NANODEGREE CAPSTONE PROJECT PROPOSAL

A PREPRINT

Thomas Rogenhofer
Udacity Machine Learning Nanaodegree
Udacity
Mountain View, CA 94040

April 27, 2019

ABSTRACT

In this paper, I outline the key points that encompass the capstone project of the Udacity Machine Learning Nanodegree. In the beginning, I introduce the domain background of the project. This provides the reader with sufficient information to follow the problem statement. Once the problem is discussed, I describe the datasets and inputs and propose a solution to the problem. At the end of this project, I present a benchmark model and evaluation metrics for measuring how well the solution is performing.

***K*eywords** Capstone Project Proposal · Google Analytics Customer Revenue Prediction

1 Project Domain Background

The Pareto Principle states that for many events, about 80% of the effects come from 20% of the causes [1]. This is an observation that holds true for many e-commerce platforms, as only a small percentage of customers generates most of the revenue. With this observation in mind, marketing managers must answer three questions in the context of a marketing campaign: how often to make an offer (frequency), when to make an offer (timing) and whom to contact (target group selection) [2].

This capstone project addresses the question of target group selection for a Google Merchandise Store, also known as GStore. In this store, customers can buy all sorts of Google related merchandise like T-shirts, bags, drinkware and other items. The GStore can be found at

`https://shop.googlemerchandisestore.com/`

2 Problem Statement

As discussed in the previous chapter, marketing managers of the GStore must identify those customers that generate most of the revenue. These customers can then be contacted in future marketing campaigns, or in real-time when visiting the GStore. Learning the user intent in real time has become a vivid area of research and personal visits to online stores often prove to be rather unsatisfactory in terms of customer experiences [3].

For this project, we need to predict the natural log of the sum of all transactions for a given customer. This means that for every user, the target is defined as follows:

$$y_{user} = \sum_{i=1}^{n} transaction_{user_i} \tag{1}$$

$$target_{user} = \ln(y_{user} + 1) \tag{2}$$

This formal description of the problem fulfills the requirement that the solution must be quantifiable, measurable and replicable. In addition, with this formal description in place, we can evaluate a number of different machine learning algorithms, in order to find the best solution to our problem.

3 Data Sets and Inputs

The datasets for this capstone project is based on the datasets provided by the kaggle competition "Google Analytics Customer Revenue Prediction", that can be found at

`https://www.kaggle.com/c/ga-customer-revenue-prediction/data`

The dataset contains three different types of files and their purpose is as follows.

- train_v2.csv: a dataset for training the machine learning algorithms. It contains user transactions from August 1st, 2016 to April 30th, 2018.
- test_v2.csv: a dataset for testing the machine learning algorithms. It contains user transactions from May 1st, 2018 to October 15th, 2018.
- sample_submission_v2.csv: a list of customers for which a prediction of the sum of transaction is required in the context of the kaggle competition.

The participation in the kaggle competition is not within the scope of this project. Therefore, I will only use the first two datasets from the list above.

Both train_v2.csv and test_v2.csv contain the columns as listed in the table below. The dataset has a number of characteristics that are important to note. First, each row in the dataset represents one visit of a customer to the store. Second, there are multiple columns in the dataset that contain JSON objects.

The target value "transactionRevenue" that I am going to predict is contained within the JSON object "total". This will require data preprocessing prior to any rounds of model training and optimization.

Table 1: Data Fields contained in the training and testing datasets

Column Name	Description
fullVisitorId	A unique identifier for each user of the Google Merchandise Store
channelGrouping	The channel via which the user came to the Store
date	The date on which the user visited the Store
device	The specifications for the device used to access the Store
geoNetwork	This section contains information about the geography of the user
socialEngagementType	Engagement type, either "Socially Engaged" or "Not Socially Engaged"
totals	This section contains aggregate values across the session
trafficSource	This section contains information about the Traffic Source from which the session originated
visitId	An identifier for this session This is part of the value usually stored as the _utmb cookie.
visitNumber	The session number for this user. If this is the first session, then this is set to 1
visitStartTime	The timestamp (expressed as POSIX time)
hits	This row and nested fields are populated for any and all types of hits.
customDimensions	This section contains any user-level or session-level custom dimensions that are set for a session.
totals	This set of columns mostly includes high-level aggregate data

4 Solution Statements

As a first step towards a solution, I will perform a number of steps in the area of data preprocessing, as some data fields contain JSON objects of various depth-levels. This will be followed by a thorough analysis of missing values and intensive data exploration.

The solution to this capstone project requires a model that uses historical customer data to make a prediction on future customer spending. As we are going to predict a continuous quantity, our solution can be classified as "regression predictive modeling". Generally speaking, this the task of approximating a mapping function (f) from input variables (X) to a continuous output variable (y). At this stage of the capstone project, it is not clear, which machine learning algorithm will be best suited for this predictive regression modeling task. Therefore, I will select four machine learning algorithms that have proven effective for predictive regression modeling.

- LinearRegression
- AdaBoostRegressor
- SVR
- XGBoost

5 Benchmark Model

I will select XGBoost as benchmark model for this capstone project, because of the following characteristics [4].

- Execution Speed. XGBoost is really fast when compared to other gradient boosting.
- Model Performance. XGBoost performs very well on regression predictive modeling problems.

I will improve the benchmark with hyper-parameter tuning. If the hyper-parameter tuning should not improve the performance score, I will apply ensemble methods.

6 Evaluation Metrics

For this capstone project, I will use three model evaluation techniques [5].

- Supplied Test Set. I will train the models on the entire training data-set and use the separate test-set to evaluate the performance of the model.
- Percentage Split. I will randomly split the entire dataset into a training and a testing partition for each round of model evaluation.
- Cross Validation. I will split the dataset into so-called k-partitions or folds. Then I will train the model on all of the partitions except one that is held back as the test-set. This procedure is repeated over the number of all k-partitions. At the end, I will calculate the average performance of all k models.

I will apply the model performance measure of "Root Mean Squared Error" (RMSE) after each round of training and testing. RMSE provides the average amount of error made on the test-set in the units of the output variable and can be formalized as follows.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(y_i - \hat{y_i}\Big)^2}, \tag{3}$$

where y hat is the natural log of the predicted revenue for a customer and y is the natural log of the actual summed revenue value plus one.

7 Project Design

For this capstone project, I propose the following theoretical workflow for approaching a solution.

- Data Exploration
  - Setup technical project infrastructure.
  - Analysis of the training data.
  - Analysis of the test data.
  - Analysis of data dimensions.
  - Statistical data summary.
- Data Pre-Processing and Data Cleaning
  - Identification of feature and target columns.

  - Pre-processing of feature columns.
  - Cleaning of data.
  - Splitting of training and validation data according to model evaluation techniques.
  - Feature scaling.
- Machine Learning Algorithm Evaluation
  - Specify models.
  - Evaluation of model with validation data set.
  - Evaluation of feature importance and feature selection.
  - Evaluation of model performance with RSME.
  - Selection of best performing model.
- Model Tuning
- Conclusion

References

[1] Stan Lipovetsky, Pareto 80/20 law: derivation via random partitioning, In *International Journal of Mathematical Education in Science and Technology,* pages 271–277., 2009.

[2] Stefan Lessmann, Kristof Coussement, Koen W. De Bock, Johannes Haupt. Targeting customers for profit: An ensemble learning framework to support marketing decision making, *SSRN Electronic Journal*, 2018.

[3] Amy Wenxuan Ding, Shibo Li, Patrali Chatterjee, Learning User Real-Time Intent for Optimal Dynamic Web Page Transformation, In *Information Systems Research,* 2015.

[4] Jason Brownlee, A Gentle Introduction to XGBoost for Applied Machine Learning, Accessed on *https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/* 2016.

[5] Jason Brownlee, How To Estimate The Performance of Machine Learning Algorithms in Weka, Accessed on *https://machinelearningmastery.com/estimate-performance-machine-learning-algorithms-weka/* 2016.