Capstone Project Proposal

Avazu Click Through Rate Prediction

(https://www.kaggle.com/c/avazu-ctr-prediction)

Domain Background

In domain of web advertisement Click Through Rate (CTR) is defined as the ratio of total number of people clicking on an ad and total number of people visiting the webpage. Similarly, we can compute click through rate of any component of our webpage.

Click through rate is one of the important metrics for companies that have online products. It helps them to analyze their products which leads to better design, increased customer satisfaction and revenue growth.

Due to its importance the task of click through rate prediction has been tackled by many Data Scientists in past years. The current state of the art is achieved by using ensemble methods which use techniques like boosting to iteratively learn multiple classifiers (decision trees) to make prediction [1] and online learning to accommodate the online nature of the problem [2].

Problem Statement

A very important challenge for an online company is to estimate the click through rate of the web component (an ad or a link). This estimate can help them evaluate the effectiveness of the component and a model for estimation can also be used to design future components in such a way that chances of their effectiveness increases.

Take example of an ad space on a web page. If you can effectively model that a user will click or show interest in one of the ads displayed on your webpage, you can estimate how much money that ad will make for you. From there you can design recommendation systems to display ads that maximizes your advertisement revenue.

For this project I would try to create a machine learning model (supervised classification) which can predict if a user will click on an advertisement campaign run by Avazu an Advertisement Platform. The data consist of 11 days of user activity (approximately 40 million impressions/page views). The data was uploaded on Kaggle by Avazu for an online competition 2 years ago.

Datasets and Input

The dataset provided is already divided in train and test set. Train set contains approximately 35 million rows (data for 10 days) and test set contains approximately 5 million rows (data for 1 day). The set of fields are as follows:

- **id: ad identifier**
- **click: 0/1 for non-click/click**
- **hour: format is YYMMDDHH, so 14091123 means 23:00 on Sept. 11, 2014 UTC.**
- **C1 -- anonymized categorical variable**
- **banner_pos**
- **site_id**
- **site_domain**
- **site_category**
- **app_id**
- **app_domain**
- **app_category**
- **device_id**
- **device_ip**
- **device_model**
- **device_type**
- **device_conn_type**
- **C14-C21 -- anonymized categorical variables**

In the dataset, the field click is our target variable which is a binary variable determining if the user clicked on the ad or not. Rest of the fields are features that will be used by model to predict the target variable. Some of the fields are anonymized categorical variable representing some feature of user or web-platform.

Solution Statement

As the size of dataset is large (approximately 7 GB) the challenge would be to select a model which can perform online learning (learn by examples iteratively instead of batch learning). I intend to explore various online learning algorithms and determine their efficiency in modelling the user behavior.
For online learning I am planning to use scikit-learn which provides many implementations of online learning algorithms like:

- MultinomialNB
- BernoulliNB
- Perceptron
- SGDClassifier
- PassiveAggressiveClassifier

All these implementation provides a partial fit method which allows to train a model one (subset) of data points at a time.

If time would permit I would also like to explore if deep neural networks would be efficient in modelling the user click behavior. The reason for considering deep learning models is that they have shown good promise is some complex classification problems like Image classification. But I am not giving priority because deep learning methods learns hierarchal features and I am not sure if the problem has any such structure.

Benchmark Model

I plan to use a simple logistic regression classifier as my benchmark model.

Evaluation Metric

I am planning to use log loss and F1-score as metrics to evaluate the performance of the model.

The log loss metric is good for models which provide a probabilistic score for their classification. In those models the log loss allows to penalize the confident false predictions. And hence provide a better evaluation metric for the probabilistic models than simple accuracy percentage.

The dataset provided seems very imbalanced hence I intend to also evaluate the model using F1-score as it allows to balance the score using precision and recall and provides the true picture of the model accuracy for each class.

Project Design

I am planning to use python2, pandas, ipython (jupyter), scikit-learn, numpy and matplotlib to perform various tasks required for this project.

I intend to divide my project in following sections:

1. **Exploratory Data Analysis:** I am planning to perform single variable EDA like frequency distribution charts and multiple variable EDA like correlation matrix to check the interaction between variables.

2. **Data Processing**
    a. Missing data: for learning I intend to remove the data points with the missing data. The reason is as we have huge amount of data and I intend to use an online learning algorithm the strategy to use standard deviation of the feature might not work as by adding the new features the standard deviation would change.

3. **Feature selection, feature scaling, feature transformation. In feature transformation**: I intend to create new features from existing features based on my understanding of the problem. Feature scaling and feature selection are difficult as most

of the methods are for batch learning and we are trying to use online learning. Though there has been some recent research for performing feature selection in online learning algorithms [3][4]. If time permits I would seek out an implementation or implement these methods myself to perform the feature selection. If not, I will select a sample of the dataset and perform the feature selection based on the sampled dataset.

4. **Model selection, validation and parameter tuning:** I intend to try various online learning algorithms present in the scikit-learn package like:

- MultinomialNB
- BernoulliNB
- Perceptron
- SGDClassifier
- PassiveAggressiveClassifier

For model selection and hyper-parameter tuning I would use forward chaining cross validation which is mostly used in time series modelling.

5. **Model Evaluation:** For evaluation I intend to use log loss and/or F1 score to check the performance of the model. Plot learning curves to see how models are performing over increasing number of data points.

6. **Conclusions**

References:

1. H. B. McMahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, D. Golovin, S. Chikkerur, D. Liu, M. Wattenberg, A. M. Hrafnkelsson, T. Boulos, and J. Kubica. Ad click prediction: a view from the trenches. In KDD, 2013.

2. H. Brendan McMahan, Gary Holt, D. Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, Sharat Chikkerur, Dan Liu, Martin Wattenberg, Arnar Mar Hrafnkelsson, Tom Boulos, Jeremy Kubica, Ad Click Prediction: a View from the Trenches at Google, Inc.

3. Haichuan Yang, Ryohei Fujimaki, Yukitaka Kusumura, Ji Liu, Online Feature Selection: A Limited-Memory Substitution Algorithm and Its Asynchronous Parallel Variation

4. Jialei Wang, Peilin Zhao, Steven C.H. Hoi, and Rong Jin, Online Feature Selection and Its Applications, TKDE2014