Machine Learning Engineer Nanodegree

Capstone Project

Nimish Kale
December 1st, 2017

I. Definition

Project Overview

Advertising is a major upcoming application and numerous tech industry giants are heavily invested in this. The involved parties in advertising are the producers which are the companies selling some product, consumers which are the buyers of the product and the advertising medium provider. Since so much finances are revolving around advertising, more focused approach will be beneficial to all the involved parties. Consumers can get more relevant sale pitches, sellers can target more relevant audience and with smartness involved the advertising medium providers can leverage it's resources effectively. That being said, several studies have been done to focus on the identifying the target audience, but not many studies have been done in identifying the conditions or environment suitable for a particular advertisement. For instance, a beverage provider can attract more customers with a hot beverage advertisement instead of a cold beverage late in the night, and make better impact by advertising in opposite fashion during noon time. With this intelligence the advertisement medium providers can improve the impact of reaching the audience.

Internet revolution has significantly increased the number of daily internet users and this has seen an exponential rise in advertisement through medium of internet. Lot of webcrawlers are hovering the internet on number of websites to understand what the user is accessing and make predictions on that basis. The social network usage also has increased significantly with the advent of numerous social networks providing different features. According to a recent study [Statista Dossier, 2014] social network users will increase to 2.44 billion users in 2018. Such an enormous increase in social network usage provides an excellent and easy opportunity for product based companies to reach potential customers. With social network as an easy platform, advertising can reach a wider audience and make more impact for the individual or the organization.

On a social network, users share their likings or interests which is visible to other users based on some criterion. Several product sellers have their profiles on these social networks and try to reach users by being connected to them. As of now, very few studies were found to have been done for approaches that effectively predict the impact of a particular post on a social network before it's actual publication. A system able to predict the impact of individual published posts can provide a valuable advantage when deciding to communicate through social media, tailoring the promotion of products and services [Moro, 2016]. Companies can leverage this information and make informed decisions regarding the posts to optimize the impact.

Problem Statement

This project is an attempt to predict the impact of numerous posts published by a cosmetic company on social networking site Facebook. There are numerous factors that can impact the effectiveness of an advertising post on a social platform like post date, time, other characteristics etc. We will try to mine these characteristics and try to categorize the impact of each post. The impact will be the total reach of the company product based on interactions of the Facebook users with the post of known set of characteristics.

The focus of this project is as follows:

1. To study and mine the characteristics that influence the impact of a post.
2. Create a model that can efficiently predict from the characteristics of the posts.
3. Measure the correctness of the created model and measure the difference between the predicted and the real values.

Dataset

The dataset is publicly available for research in UCI repository located here: https://archive.ics.uci.edu/ml/datasets/Facebook+metrics#.

The majority of input parameters (type, category, post hour, .. etc.) are discrete variables however the majority of output parameters (lifetime post reach, post consumers, likes, shares etc.) are continuous variables. As we are targeting in evaluation how to improve the reach of a post, exact value estimation of these output variables does not add too much value. Considering this we will use binning approach and split these output variables into fixed number of bins and the convert the problem into a classification analysis. We perform the analysis for Lifetime Post Total Reach, Lifetime Post Total Impressions, Lifetime Engaged Users, Lifetime Post Consumers, Lifetime Post Consumptions, Lifetime Post Impressions by people who have liked your Page, Lifetime

Post reach by people who like your Page, Lifetime People who have liked your Page and engaged with your post, Total Interactions as the target variables.

Metrics

For this problem we use **accuracy score** and **mean percent error** as the metrics to evaluate the performance of the model created. We evaluate over multiple trials these scores for each trial and report a mean average score.

Accuracy can be explained as a score of the total input samples correctly mapped to the right output features. Higher the accuracy score better the model. Apart from accuracy, we also calculate the mean percent error, where lower mean percent error signifies that the number of misclassified samples are low.

For classification problems, a high number of correct classification of samples and a low number of misclassifications is a good justification of the model. So, accuracy and mean percent error together should be a good judge for the analysis of this problem. Reference: https://en.wikipedia.org/wiki/Mean_percentage_error

II. Analysis

Data Exploration

The dataset contains 500 samples each with 19 features. The dataset is a complete dataset, with no missing data. The details of the features are as described below:

1. Page total likes: Number of likes the page had before the post was published. This is a continuous integer value.
2. Type: This is a discrete feature explaining the type of content like a link, photo, video or status.
3. Category: Manual categorization of the campaign for which the post was published. This is also a discrete feature with values as Action (special offers and contests), product (direct advertisement, explicit brand content) and inspiration (non-explicit brand related content)
4. Post Month: Month the post was published which is a discrete variable with values in range from 1 to 12.
5. Post Weekday: Discrete feature variable suggesting the day of the week when the post was published having values from 1 to 7 (Monday-Sunday).
6. Post Hour: Hour of the day having values from 1 to 24.

7. Paid: Boolean valued feature with value 1 if the post was paid post by the company, and 0 otherwise.
8. Lifetime Post Total Reach: Total number of people that viewed the post which is a continuous valued feature.
9. Lifetime Post Total Impressions: Number of times a post from a page was displayed as an update. People may see multiple impressions of the same post. For example, someone might see a Page update in News Feed once, and then a second time if a friend shares it.
10. Lifetime Engaged Users: Number of unique users that clicked on the post
11. Lifetime Post Consumers: Number of users that clicked anywhere on the post.
12. Lifetime Post Consumptions: Total number of clicks
13. Lifetime Post Impressions by people who have liked your Page: Number of impressions who have liked the company page
14. Lifetime Post reach by people who like your Page: Number of people who saw the post.
15. Lifetime People who have liked your Page and engaged with your post: Number of unique users that have liked the page and clicked on the post.
16. Comment: Number of "Comments" on the post
17. Likes: Number of "Likes" on the post
18. Shares: Number of "Shares" of the post
19. Total Interactions: Sum of all likes comments and shares on the post.

From the above list of features, 8. onwards will be considered as output or to be predicted features and remaining are inputs. As seen from the above list, all the considered input features are discrete variables and the output variables are majority continuous values.

Extracting input features, the distributions of the features is as follows:

There are 4 input values for feature Type, where the distribution is as Photo (426), Link (22), Status (45), Video (7). So, the number of input samples for Photo are superseding others. For category, there are 3 possible input values and the distribution is as Action (215), Product (130) and Inspiration (155), which close to an even distribution. Of the total samples, there are 139 unpaid posts and remaining 361 posts are paid. Post Month, Post Week and Post Hour have a nearly even distribution.

Exploratory Visualization

We plot all the output variables with the input variables and try to establish a relation between the input and output features. Following are the visualization for output feature '*Lifetime Post total Reach*'. (All Plots included)

In the (**Fig-1**) *category* vs *Lifetime Post Total Reach*, Product(1) seems to be having a larger impact on the lifetime post total reach than Inspiration(3) and Action(0).

In the (**Fig-2**) *page total likes* vs *Lifetime Post Total Reach*, there seems to be heavier concentration of the post total reach when page total likes are high.

From the (**Fig-3**) *paid* vs *Lifetime Post Total Reach*, we can clearly see that paid posts have higher reach.

From the (**Fig-4**) *post hour* vs *Lifetime Post Total Reach*, post hour from 1 to 14 earns higher reach.

From the (**Fig-5**) *post month* vs *Lifetime Post Total Reach*, post month from January to July earns higher reach.

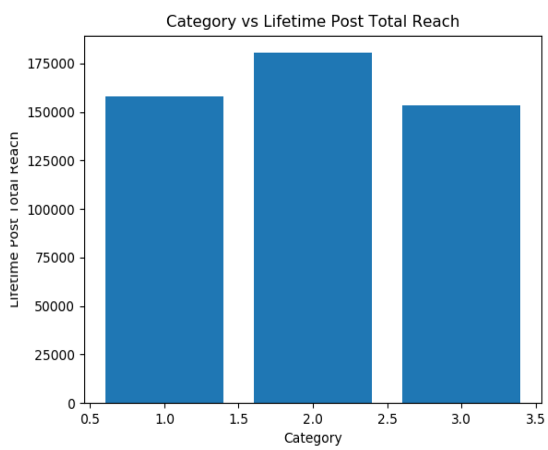


Fig-1

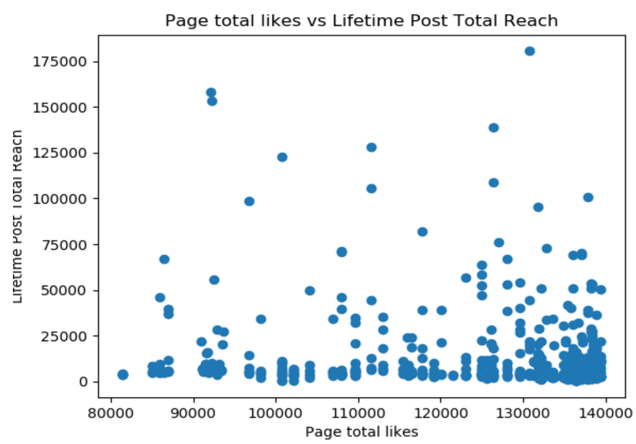


Fig-2

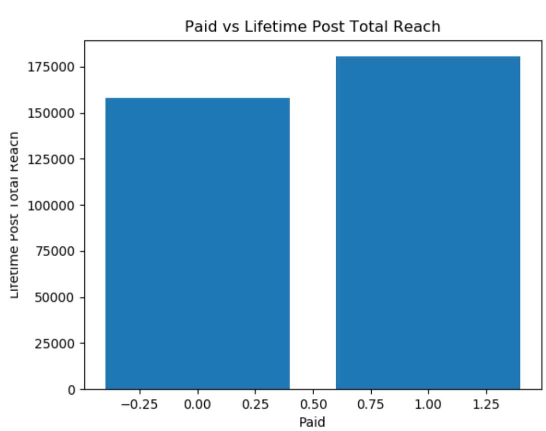


Fig-3

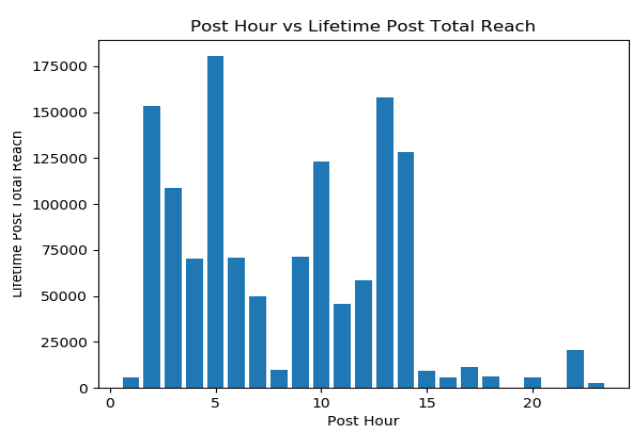


Fig-4

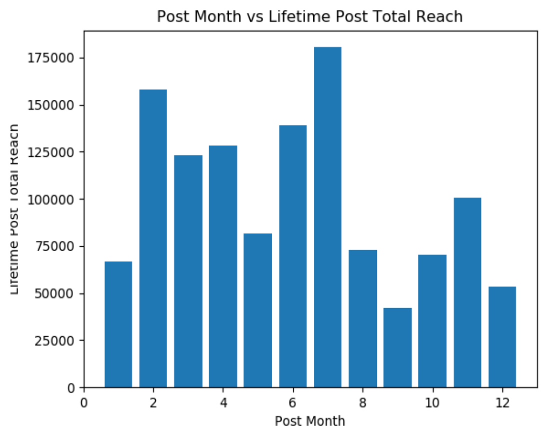


Fig-5

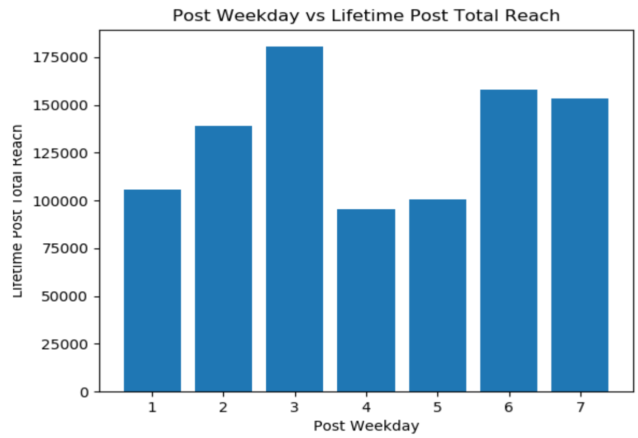


Fig-6

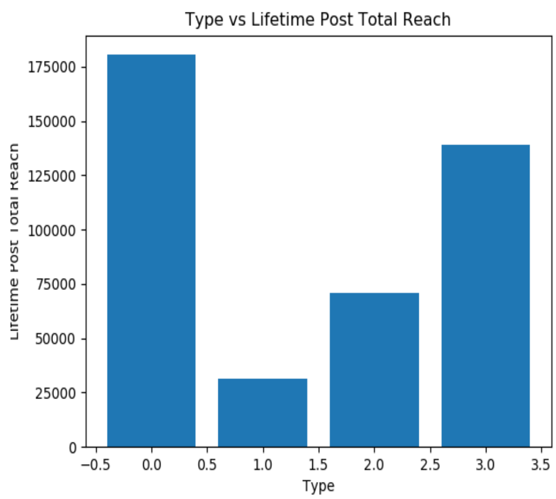
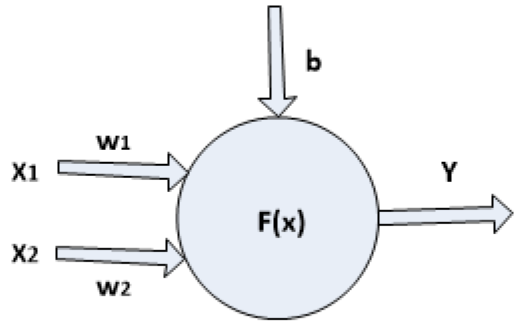


Fig-7

Algorithms and Techniques

Multi-layer Perceptron (MLP) is a supervised learning algorithm that learns a function by training on a dataset.



The figure above explains the functionality of a basic neuron. X_1 and X_2 are inputs to the network where w_1 and w_2 are the weights associated with each input. There is bias input with value 1 and weight b. The output Y is computed using function F(x), where

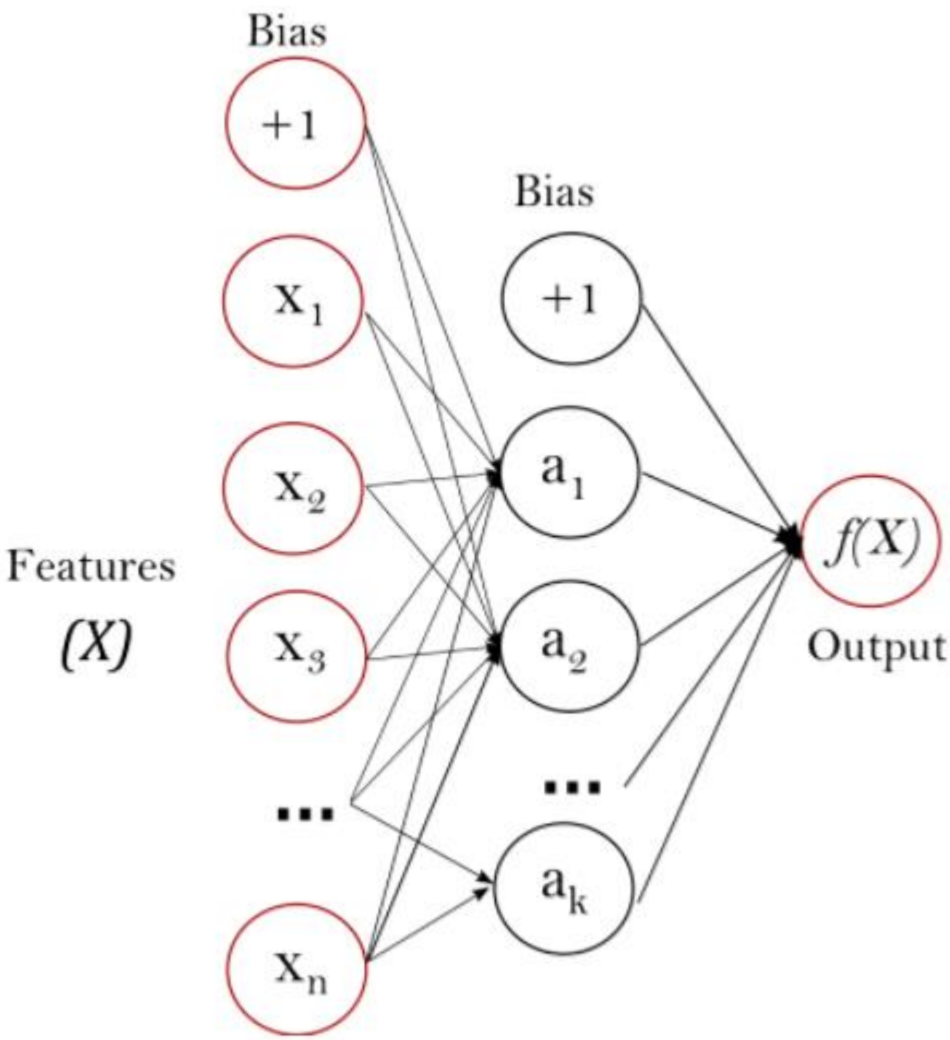
$$F(x) = \dot{f}(X_1 * w_1 + X_2 * w_2 + b)$$

The function F(x) is called as activation function and it converts the input variable based on the weights and bias to output values. The purpose of the activation function is to introduce non-linearity into the output of a neuron. This is important because most real world data is non linear and we want neurons to learn these non linear representations. The activation functions that exists are

- Sigmoid: Input values are converted to range 0 to 1, using exponential functions
- Tanh: Input values are converted to range -1 to 1
- ReLU: Rectified Linear Unit

The bias values adds a specific constant value to the activation functions which helps in fitting the data.

A Multilayer Perceptron is a model containing one or more hidden layers apart from input and output layers. The figure below shows an example of multilayer perceptron with a single hidden layer



Input Layer: The Input layer has *n* nodes X_1 to X_n. The Bias node has a value of 1. The other nodes take X_1, X_2 .. X_n as external inputs which are numerical values depending upon the input dataset. No computation is performed in the Input layer, so the outputs from nodes in the Input layer are 1, X_1, X_2 … X_n respectively, which are fed into the Hidden Layer.

Hidden Layer: The Hidden layer has n nodes with the Bias node having an output of 1. The output of the other two nodes in the Hidden layer depends on the outputs from the Input layer as well as the weights associated with each input. Figure The function ***f*** refers to the activation function. These outputs of the hidden layer are then fed forward to the nodes in the Output layer.

Output Layer: The Output layer has one node which take inputs from the Hidden layer and perform similar computations as hidden node. The output value calculated as a result of these computations act as outputs of the Multilayer Perceptron.

Given a set of features **X = (X_1, X_2, ... X_n)** and a target **Y**, a Multilayer Perceptron can learn the relationship between the features and the target, for either classification or regression. Considering this, the multilayer perceptron is a good candidate to develop a model based on discrete input features and classify output features.

References: https://en.wikipedia.org/wiki/Artificial_neural_network https://link.springer.com/chapter/10.1007/978-3-319-67340-0_1

Benchmark

The previous study on this dataset (Moro et al., 2016) mentions about using SVM techniques for predicting the performance. The mean absolute percentage errors achieved by SVM technique is ranging from minimum of 26.9% for Lifetime people who have liked your page and engaged with your post to 69.3% for Lifetime post total impressions.

III. Methodology

Data Preprocessing

The purpose of the model building is to provide the seller with appropriate input feature parameters to reach larger audience and promote sales of the products. The output features are continuous valued variables and building a model to predict exact value does not add value.

Transformation to categorical values: We categorize the continuous output variables into categorical values and build a classification model to predict the correct classes. Data binning or bucketing is a data pre-processing technique used to reduce the effects of minor observation errors. The original data values which fall in a small interval, are replaced by a value representative of that interval, often the central value. It is a form of quantization. For our dataset we divide each feature into 10 bins and come up with input schemes for predicting higher numbered bins.

Scaling and Normalizing: Standardize features by removing the mean and scaling to unit variance. Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the training set. Mean and standard deviation are then stored to be used on later data using the transform method. Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual feature do not more or less look like standard normally distributed data. We perform scaling and normalizing on the continuous variables before converting them to discrete variables.

Implementation

The following flow was implemented for the analysis to solve the problem:

1. Data preprocessing
    - We scale and normalize the continuous features without which it becomes difficult to train and have balanced model.
    - https://en.wikipedia.org/wiki/Feature_scaling
    - Then we transform the continuous output variables into categorical values and assign numeric values for the categories.
2. Split in training and testing data
    - Before training a model it is important to have a testing data to verify the correctness of the developed model.
    - Since we have one dataset without any distinction of training and testing data, we randomly divide the dataset to verify our model.
    - This is iterative process to add variations to our training and testing data.
3. Creation of the MLP model
    - Using the earlier split training data, develop a Neural Network based model to predict the output values.
    - Sklearn provides a tool called grid search, that performs an exhaustive search over specified parameters. Using this tool we extract the optimized configuration.
        - Reference: http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html
    - Over multiple iterations, for the MLPClassifier, the lbfgs (quasi-Newton methods) solver always outperformed other solvers. Lbfgs stands for limited memory Broyden–Fletcher–Goldfarb–Shanno algorithm, which is a optimized version of bfgs algorithm for parameter estimations.
        - Reference:
            1. https://en.wikipedia.org/wiki/Limited-memory_BFGS
            2. https://en.wikipedia.org/wiki/Broyden%E2%80%93Fletcher%E2%80%93Goldfarb%E2%80%93Shanno_algorithm

    - We tuned the following parameters of MLPClassifier:

Parameters	alpha	Learning_rate	Max_iter	solver
Values	0.1, 0.001, 0.0001	Constant, invscaling, adaptive	200, 400, 750, 1000	Lbfgs, adam

- Alpha penalty parameter:
    1. Increasing alpha may fix overfitting problems by encouraging smaller weight variables. Similarly, decreasing alpha may fix underfitting problems by encouraging larger weights variables.
    2. We tried values of 0.1, 0.001, 0.0001 and 0.00001 for the penalty parameter.
- learning_rate:
    1. Learning rate configuration represents the process by which the weights of the neurons are updated. We iterated with different learning rate options of constant, invscaling and adaptive.
    2. Adaptive learning rate gave optimized error and accuracy scores. With constant learning rate the number of iterations needed to converge the model were high and invscaling learning rate reported higher error rates.
- max_iter.
    1. Maximum number of iterations until the solver converges. The default iterations for MLP is 200, which were not sufficient for the algorithm to provide improved results. For most of the output features the number of iterations needed to converge were in excess of 400. A max_iter=1000 configuration provided optimized results. With multiple iterations and fine tuning the parameters the model provided optimized results as mentioned below.

- The **Final model** has following parameters:

Parameters	alpha	Learning_rate	Max_iter	solver
Values	0.0001	adaptive	1000	Lbfgs

- After model is generated, absolute error and mean average error over multiple iterations for each output feature is evaluated.

- As we will be randomly dividing the dataset multiple times, the error rate will be averaged over all the iterations.

IV. Results

Model Evaluation and Validation

Multiple trials (10) are performed and the mean of percent errors and f-score is reported for each of the output features. In each of the trial we randomly divide the available dataset into training and testing data in ration 1:2.

The table below shows the reported mean Absolute Percent Error and F-score for each of the output features.

Output Feature	**Mean Percent Error**	**Mean Accuracy Score**
Lifetime Post Total Reach	3.81%	0.78
Lifetime Post Total Impressions	3.33%	0.94
Lifetime Engaged Users	1.03%	0.76
Lifetime Post Consumers	3.57%	0.8
Lifetime Post Consumptions	2.24%	0.83
Lifetime Post Impressions by people who have liked your Page	0.91%	0.98
Lifetime Post reach by people who like your Page	4.67%	0.67
Lifetime People who have liked your Page and engaged with your post	13.09%	0.53
Total Interactions	2.3%	0.94

The advantage of dividing the continuous dataset into equal sized bins, is that it is insensitive to minor changes or errors. These errors can be human or sensor errors which unknowingly happens in field. Scaling and normalizing dataset somewhat helps in handling outliers.

For evaluating robustness of the model, we calculate the variances in the percent error and accuracy scores for each of the output variable. The observed variances for accuracy scores for all output features are less than 0.1, and that for percent error rates vary from 0.4 for Total Interactions to high of 34% for Lifetime Post Total Reach.

Accuracy scores over 10 trials:

Output Feature	**Accuracy Scores**
Lifetime Post Total Reach	[0.80, 0.81, 0.78, 0.80, 0.83, 0.73, 0.85, 0.78, 0.88, 0.64]
Lifetime Post Total Impressions	[0.95, 0.95, 0.95, 0.92, 0.97, 0.95, 0.90, 0.90, 0.93, 0.92]
Lifetime Engaged Users	[0.68, 0.81, 0.85, 0.76, 0.78, 0.68, 0.76, 0.69, 0.80, 0.76]
Lifetime Post Consumers	[0.78, 0.76, 0.78, 0.73, 0.85, 0.83, 0.81, 0.76, 0.81, 0.90]
Lifetime Post Consumptions	[0.76, 0.71, 0.83, 0.85, 0.88, 0.80, 0.76, 0.88, 0.71, 0.85]
Lifetime Post Impressions by people who have liked your Page	[0.97, 1.0, 0.97, 0.93, 1.0, 1.0, 0.97, 1.0, 1.0, 1.0]
Lifetime Post reach by people who like your Page	[0.68, 0.68, 0.68, 0.69, 0.71, 0.62, 0.73, 0.73, 0.71, 0.68]
Lifetime People who have liked your Page and engaged with your post	[0.59, 0.47, 0.46, 0.46, 0.54, 0.57, 0.57, 0.54, 0.54, 0.54]
Total Interactions	[0.97, 0.93, 0.95, 0.95, 0.90, 0.95, 0.97, 0.97, 0.90, 0.95]

Percent Errors reported over the 10 trials:

Output Feature	Accuracy Scores
Lifetime Post Total Reach	[6.0, 0.0, 14.0, 2.0, 0.0, 14.0, 0.0, 0.0, 0.0, 0.0]
Lifetime Post Total Impressions	[0.0, 0.0, 0.0, 0.0, 10.0, 0.0, 0.0, 0.0, 0.0, 8.0]
Lifetime Engaged Users	[0.0, 2.0, 0.0, 12.0, 2.0, 4.0, 0.0, 2.0, 14.0, 6.0]
Lifetime Post Consumers	[2.0, 0.0, 6.0, 0.0, 8.0, 0.0, 2.0, 6.0, 2.0, 0.0]
Lifetime Post Consumptions	[0.0, 0.0, 10.0, 10.0, 0.0, 10.0, 2.0, 0.0, 10.0, 2.0]
Lifetime Post Impressions by people who have liked your Page	[18.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 10.0, 0.0, 0.0]
Lifetime Post reach by people who like your Page	[0.0, 2.0, 6.0, 12.0, 2.0, 4.0, 2.0, 16.0, 4.0, 2.0]
Lifetime People who have liked your Page and engaged with your post	[16.0, 18.0, 4.0, 22.0, 8.0, 24.0, 6.0, 16.0, 6.0, 4.0]
Total Interactions	[0.0, 6.0, 0.0, 2.0, 0.0, 0.0, 0.0, 18.0, 0.0, 0.0]

Additionally, we induce gaussian noise in the dataset and rerun our analysis to understand the performance with noisy data. The accuracy scores and percent errors reported are comparable to the one reported without the noise induced, confirming the robustness of the model generated. The accuracy scores for the output features with noise induced is as shown below:

Output Feature	Mean Percent Error with noise	Mean Accuracy Score with noise
Lifetime Post Total Reach	6.81%	0.78
Lifetime Post Total Impressions	7.33%	0.91

Lifetime Engaged Users	4.03%	0.76
Lifetime Post Consumers	5.57%	0.78
Lifetime Post Consumptions	4.24%	0.79
Lifetime Post Impressions by people who have liked your Page	0.64%	0.98
Lifetime Post reach by people who like your Page	4.67%	0.67
Lifetime People who have liked your Page and engaged with your post	19.09%	0.53
Total Interactions	3.5%	0.94

Justification

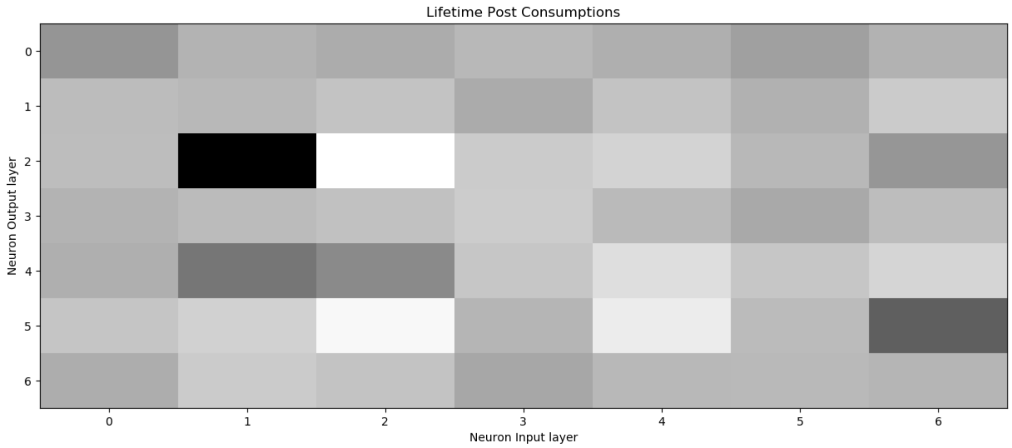
The benchmark model given by Moro et al., 2016 mentions about using SVM techniques for predicting the performance. However, the model is developed for predicting exact values, unlike the MLP model discussed here which divides the data and predict the right bucket for each data entry. Due to the bucketing, the model achieves significantly better results than the reported values in the benchmark model.

The mean absolute percentage errors achieved by benchmark SVM model is ranging from minimum of 26.9% for Lifetime people who have liked your page and engaged with your post to 69.3% for Lifetime post total impressions. The mean absolute percentage errors achieved by the MLP model ranges from 0.91% for Lifetime Post Impressions by people who have liked your Page to 13.09% for Lifetime People who have liked your Page and engaged with your post.

V. Conclusion

Free-Form Visualization

We plot the gray scale of the model's input layer coefficients to outputs firing values. For output feature Lifetime Post Consumptions, the plot is shown in the figure below



On gray scale large negative numbers are black, large positive numbers are white, and numbers near zero are gray. For MLPClassifier each neuron is taking it's weighted input and fits the input data, which outputs 0 for inputs much less than 0 and outputs 1 for inputs much greater than 0. So, for instance, if a particular weight is large and negative it means that neuron is having its output strongly pushed to zero by the input from neuron of the underlying layer. In the above figure if a particular box is gray then that means that neuron isn't very sensitive to the output of neuron in the layer below it.

The benchmark model as specified earlier, reports 26.9% to 69.3% mean percent errors. These large error rates reported are justified since we have a small set of features available to build the model. However, exact prediction of the continuous variables for this problem is not adding value. The motive behind the problem was to identify a set of input features for which the reach of the post to the public can be maximized. For this, converting continuous valued dataset into discrete values helps build a stronger model with smaller error rate and higher accuracy. The important point this problem makes is that, the dataset should be transformed into one which suits the intended motive and maximize the use of the available inputs to assist and be a stepping stone in building a better model.

Reflection

This research focused in modeling performance metrics extracted from posts published in a company's Facebook page through the usage of data mining. Moreover, the MLP technique was employed by feeding it with seven input features, all provided by Facebook's page, except a content specific categorization provided by the page's manager. Nine models for each of the output variable, were implemented with these input features, from which the two target variables achieved the best performance modeled the “Lifetime Post Impressions by people who have liked your Page” and the “Lifetime Engaged Users” output features, with a mean absolute percentage error of 0.91% and of 1.03%, respectively.

On running the MLP model with continuous values, the results were comparable to the benchmark SVM model with percent errors ranging from 24% to 69% for the output variables. This analysis by transforming the values into categorical variables achieved improved error rates with continuing to achieve the intended goal of creating an assisting model for targeted marketing.

Improvement

The model can be enriched with several other context features (e.g., if the product is being advertised elsewhere) for enhancing its performance. If additional information like with exact text is available, then text mining methods could be employed for extracting additional information and enhance the overall model. Additionally, having the text of the comments of each post, user sentiment analysis could also be employed to add the feelings each post is generating and make better predictions.

References

- (Moro et al., 2016) S. Moro, P. Rita and B. Vala. Predicting social media performance metrics and evaluation of the impact on brand building: A data mining approach. Journal of Business Research, Elsevier, In press.
- Ballantine, P.W., Lin, Y., & Veer, E. (2015). The influence of user comments on perceptions of Facebook relationship status updates. Computers in Human Behavior, 49, 50–55
- Habibi, M. R., Laroche, M., & Richard, M. O. (2014). The roles of brand community and community engagement in building brand trust on social media. Computers in Human Behavior, 37, 152–161
- https://en.wikipedia.org/wiki/Artificial_neural_network
- https://link.springer.com/chapter/10.1007/978-3-319-67340-0_1
- http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html
- Ballantine, P.W., Lin, Y., & Veer, E. (2015). The influence of user comments on perceptions of Facebook relationship status updates. Computers in Human Behavior, 49, 50–55.
- Bianchi, C., & Andrews, L. (2015). Investigating marketing managers' perspectives on social media in Chile. Journal of Business Research, 68(12), 2552–2559.

- Cortez, P. (2010). Data mining with neural networks and support vector machines using the R/rminer tool. Advances in data mining. Applications and theoretical aspects (pp. 572–583). Berlin Heidelberg: Springer.
- Cortez, P., & Embrechts, M. J. (2013). Using sensitivity analysis and visualization techniques to open black box data mining models. Information Sciences, 225, 1–17. http://dx.doi.org/10.1016/j.ins.2012.10.039.
- Culnan,M. J.,McHugh, P. J., & Zubillaga, J. I. (2010). Howlarge US companies can use Twitter and other social media to gain business value. MIS Quarterly Executive, 9(4), 243–259.