Shadi.com Case Study

Saruk Shaikh

Understanding the data:

- The data train data has 101180 and 24 columns
- The data set doesn't have any null value
- It has two columns F15 and F16 which consist of Dates
- The features consist different units so standardization is required
- The C column has binary value (0 and 1)
- The ratio of 0 and 1 is 76353:24827 or 3:1
- The data is imbalanced

Pre-processing:

- As the data is imbalanced I used Up-sampling technique which is used to duplicate observation from minority class.
- I can us class weight or Smote for this data set.
- I have scaled the data
- The date and time column is not randomly converted to some factor type, I used datetime package and represented the number from day 1 year 1.

Machine learning models:

- I have used several models which were fluctuating with accuracy
- First I used the logistic without upsampling but it gave only 40% accuracy
- Then using upsampling the train and test accuracy went to 67%
- Xgboost was a good choice because its main advantage is it works good with imbalanced data, but it accuracy was around 68-70%, after using class weight also to tackle imbalance problem.
- Then I used thought of NN to read and understand the data by the model

- Neural Network characteristics:
    - 1. Input layer: Input_dim is set to 22 because each of our input samples has 22 features. So we have 22 input neurons at input layer.
    - 2. Hidden layer: Next is a Dense layer which is fully connected layer with 1024 neurons.
    - 3. Dropout layer: Dropout layer which disables fraction of inputs to reduce over fitting. Here I have set dropout to 5%.
    - 4. Activation function: I have used 'relu' as a activation function for hidden layer and 'sigmoid' at the output layer.
    - 5. Output layer: Output layer is again a dense layer with single neuron as we are dealing with binary classification which has only one output 0 or 1.
    - 6. Model optimizer: It is the search technique used to update weights in the model. I have used RMSprop which is an adaptive learning rate optimization method.
    - 7. Model Loss Function: 'binary crossentropy' as it is binary classification problem.
    - 8. Model training: The model is trained on NumPy arrays using the fit() function.

Evaluation:

- I used Stratified kfold cross validation because it samples the data by taking equal number of classes so it would be good for validation
- Data splitting: Here we separate portion of data into validation dataset and evaluate the performance of model on that validation dataset each epoch. Here I have split 20% data into validation dataset.
- Confusion matrix: It is a table which is used to describe the performance of classification model on set of test data.

```
[[ 8848  6495]
 [  807 14392]]
```

The classifier made a total of 30542 predictions. Out of those 30542 cases, the classifier predicted "1" 14883 times, and "0" 15659 times. In reality, 15199 persons belong to class “1”, and 15342 to class “0”.

	precision	recall	f1-score	support
0	0.92	0.58	0.71	15343
1	0.69	0.95	0.80	15199
avg / total	0.80	0.76	0.75	30542