Prediction

I have reviewed the data of activity.csv and target.csv. The target csv is my goal and that's why I labeled the target csv as 1. I concatenate both csv after taking them to the pandas dataframe. After taking the data into pandas dataframe which data matched with both activity and target data's dates and customers those are labeled as 1 and others as 0.

> Feature Selection

There are 5 columns in the newly merged activity_new dataframe which are date, customer, activity_type, activity_count, and label. My task is predicting the data for the closed deals. And what I saw that there is only one feature which is 'activity_type' for predicting the closing deals. The other columns are not important features. So what I did is written in the section below.

> Pipeline for training data

1. At first I downsampled the majority label 0 data, close as label 1 data.
2. Dropped the unnecessary features which are 'date', 'customer', 'activity_count'.
3. I used LabelEncoder of scikit learn as it helps to convert the categorical data to numeric data.
4. Then I split the data for training and validation.
5. Then I applied Standard scaling which helped me to get standardized distributed data with a zero mean and standard deviation of one.
6. Then used the data in different algorithms for classifying whether it is a closed deal or not.
7. For that I used Random Forest with 300 trees, Logistic regression with tol parameter 0.001, SVM(Support Vector Machine) with tol=0.0001, Gradient Boost Classifier with learning rate 0.1 and max_depth=9, then MLP Classifier with hidden layer size (25, 25, 25) and 200 iterations.

> Accuracy for each algorithm

Algorithms	Training Accuracy	Precision	Recall	F1-score	Confusion Matrix
Random Forest	0.66 0.0 – 0.678 1.0 – 0.636	0.0 – 0.73 1.0 – 0.58	0.0 – 0.68 1.0 – 0.64	0.0 – 0.70 1.0 – 0.60	TN - 4073 FP - 1927 FN - 1492 TP - 2617

Logistic Regression	0.59 0.0 – 1.000 1.0 – 0.000	0.0 – 0.59 1.0 – 0.00	0.0 – 1.00 1.0 – 0.00	0.0 – 0.74 1.0 – 0.00	TN - 6000 FP - 0 FN - 4109 TP - 0
SVM	0.64 0.0 – 0.651 1.0 – 0.629	0.0 – 0.72 1.0 – 0.55	0.0 – 0.65 1.0 – 0.63	0.0 – 0.68 1.0 – 0.59	TN - 3908 FP - 2092 FN - 1523 TP - 2586
Gradient Boost	0.66 0.0 – 0.678 1.0 – 0.636	0.0 – 0.73 1.0 – 0.58	0.0 – 0.68 1.0 – 0.64	0.0 – 0.70 1.0 – 0.60	TN - 4073 FP - 1927 FN - 1492 TP - 2617
MLP Classifier	0.66 0.0 – 0.713 1.0 – 0.582	0.0 – 0.71 1.0 – 0.58	0.0 – 0.71 1.0 – 0.58	0.0 – 0.71 1.0 – 0.58	TN - 4278 FP - 1722 FN - 1717 TP - 2392

As you can see Logistic Regression has become biased on label 0. Gradient Boost and Random Forest almost gave the same result and overall their accuracy is better than other algorithms like SVM and MLP Classifier. Even I tried to fit the data in the XGBoost Classification algorithm but it's also giving the same result like Random Forest and Gradient Boost. I chose these algorithms because I have only one feature and these machine learning algorithms work well in such types of data.

> Future planning for getting good result

1. The labels are imbalanced so if I get better data with a balanced dataset or at least close to a balanced dataset then I think we'll get a better result using classification algorithms.
2. If the dataset is not that much balanced then as I used downsample, that time I would try to upsample the dataset so that the information will stay as it is.
3. Here I am getting only one feature. So, if I get more features then I will try to fit the data in a deep neural network, which will probably give me good results in prediction.