Predict the Churn Risk Rate

As I’m new in Machine Learning and Data Science field, It was a challenging task for me but at the same time it was quite interesting. I’ve learnt a lot for this challenge. I’ve used certain procedure for this task which are mentioned below:

* The first thing was as usual loading the data(train.csv).
* After that I’ve checked the entire data whether it consist missing values, outliers, special characters etc.
* Then I started feature engineering in which I’ve replaced missing value with mean of that particular feature in case of continuous data.
* In case of categorical data, I used mode to replace missing values.
* For special character like “?”, I used mode because it was present in categorical feature column.
* There were many outliers in terms of negative values, I simply replaced them with 0. And in target feature column, I’ve used absolute function.
* After cleaning the data, I’ve dropped few irrelevant features.
* Then to make useful, I’ve created dummies of categorical feature columns and then dropped original one. And finally created a new data frame by concatenating old data frame with all the dummies.
* Next step was to standardize the complete all the feature columns except target feature column.
* After that I’ve split the data in training and testing perspective for the model.
* I’ve used 2 models; one was Logistic Regression and other was Random Forest Classifier for that. And in terms of overall accuracy the Random Forest Classifier performed better. So, I’ve chose that as final model and deployed it.
* In final process, I’ve created another notebook where I’ve loaded the test.csv and done same feature engineering and then for prediction I’ve loaded the pickled model and got the prediction and made a new data frame as mentioned in task. And also created a csv file from that data frame.

For this challenge, I’ve used jupyter notebook with python. I’ve created 2 notebooks; one for train.csv and another for test.csv. For code, I’ve used my knowledge and also refer the python documentations.