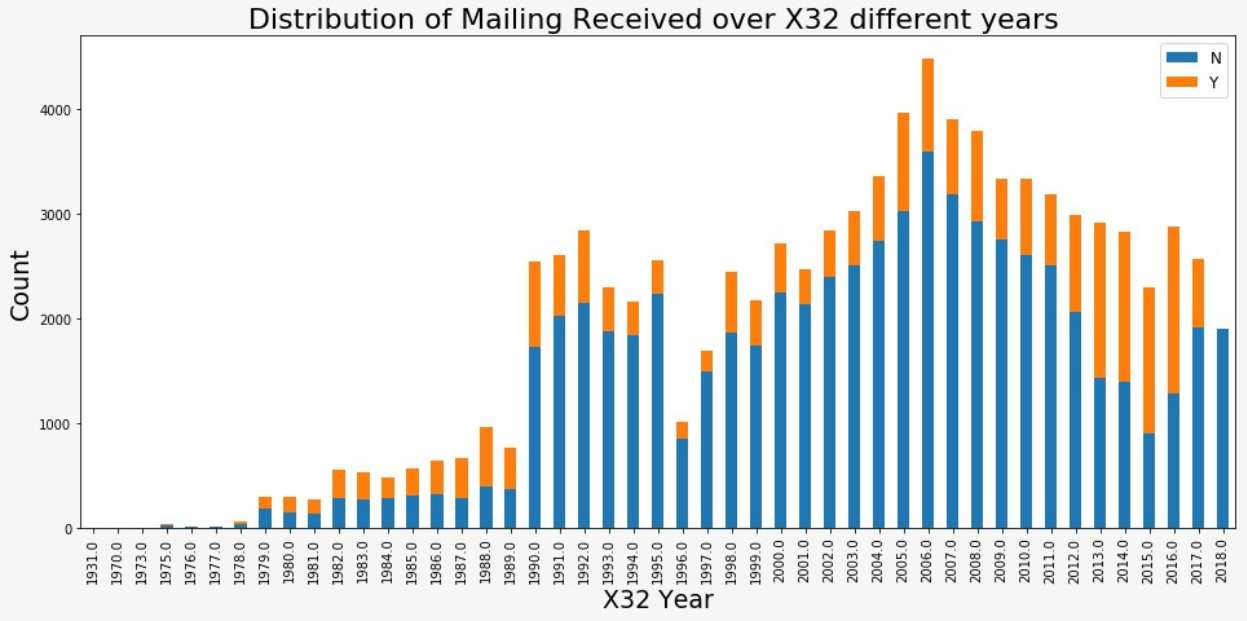
1)Objective:
Fundraising contributes towards a significant amount of money required by the Stony Brook University. Donors have contributed for different campaign priorities such as Scholarships, Research & Innovation, Endowed Faculty, Medicine, Economic Development, Facilities & Campus life, Stony Brook Annual Fund. The last fundraising campaign helped the university to get $630.7 million from 47,961 friends, corporations, foundations, faculty, staff and alumni. Achieving this target require to reach out more people who are willing to donate. So, the problem statement is to identify the potential donors to send the mails and also identify the users who responded to the emails with the fund. That means we need to predict the last two columns in the data set i.e "Mailing Received" and "Mailing Responded".

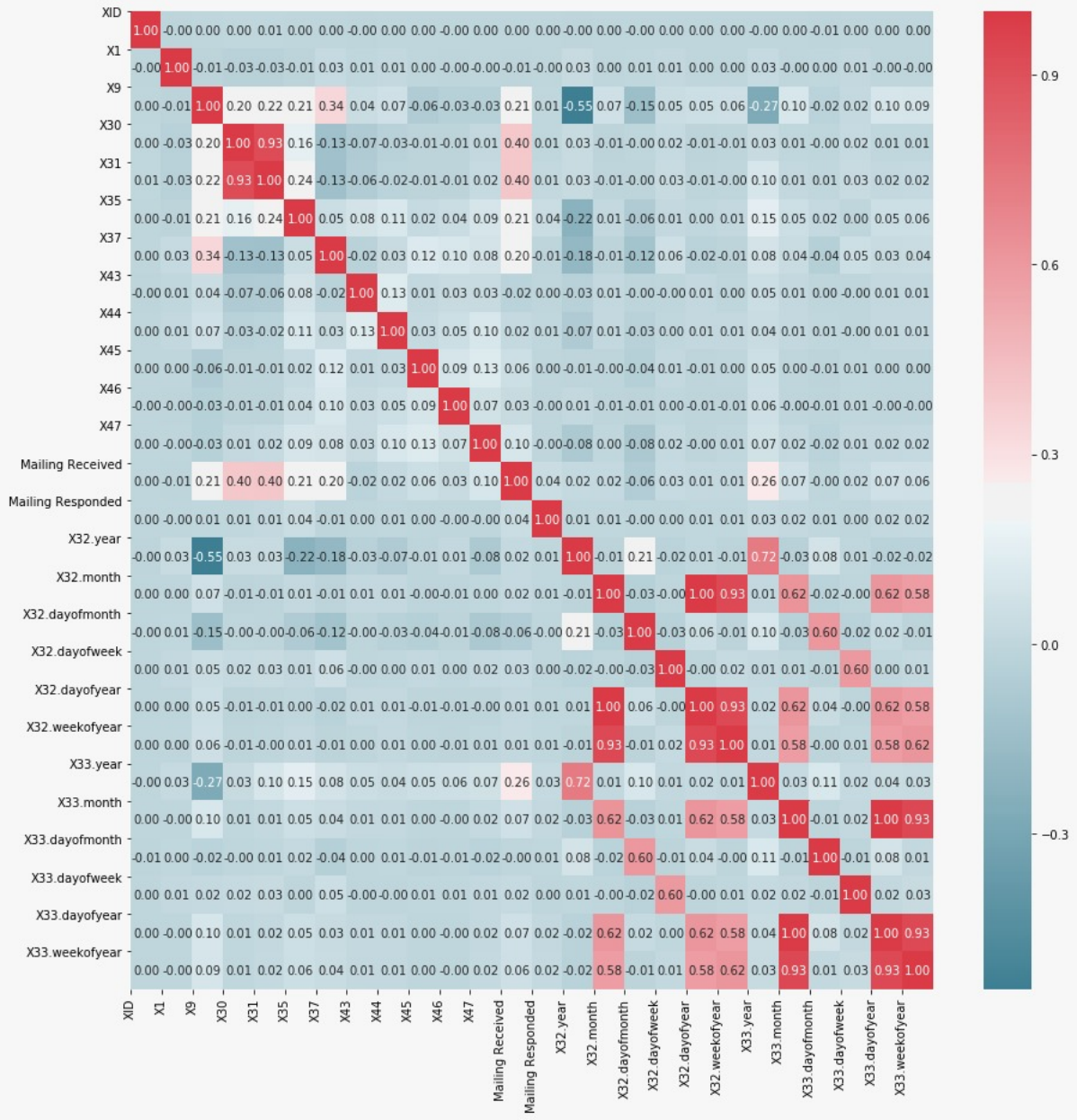
2)Background:
The discussion with the Stony Brook Fund Campaign Management team gave more insights regarding the last two columns and shed light on how we can tackle this particular anonymous data as they have clearly stated the part of the campaign's website is generated from the dataset provided to us and they gave a suggestion to guess the feature based on the information provided in the website and tweak them if it is going to help the prediction. They have given some examples to focus on the propensity of the person to donate based on certain features like relationship with the university and demographics, etc. As per the objective, we have focussed on predicting the column "Mailing Received" and then using that predicted column along with the other columns to predict "Mailing Responded".



3)Exploratory Data Analysis (EDA):
EDA is all about making sense of data in hand and gain some insight before using a model to predict. As mentioned in our proposal, we have followed the same process by performing the encoding to transform the categorical features to numerical features cleaned the data properly by analyzing the data. To discover the pattern, to spot the anomalies, to come up with the hypothesis which can work on your data, we have analyzed the datatype(integer, float, object), the distribution

of different features(minimum, maximum, mean and standard deviation) and come up with the summary statistics with the correlation plot given below. Moreover, you can observe that the output column “Mailing Received” is correlated with the columns “X9,X30,X31,X35,X37”. Here you can observe that the columns X30 and X31 are highly correlated with correlation of 0.93. X9 and X37 are correlated with the correlation coefficient 0.34.

You can observe that there is strong correlation between the columns generated from X32 and X33 which contains date. We can observe that the data set is imbalanced and we need to use the techniques discussed in class.



4)Anonymized Data:
To deal with the anonymous data, data insights and the relationship of data on the output plays a crucial role in improving the model. As per [6], inferred information: It may be possible for certain information to be inferred from anonymized data. For example, masking may hide the personal data, but it doesn't hide the length of the original data in terms of the number of characters. As per [5], two things to do with anonymized features: 1) Try to decode the features by guessing the true meaning of the feature. Features can be of different kinds- categorical, numeric, date-time etc. 2) Guess the feature type which will help in preprocessing the data for the model prediction. Since it is communicated to us that the information available in the campaign webpage, the webpage gives 27 options where a donor can support. By analyzing the data set, we got the columns X21, X26 with unique values close to 27. So we can guess with high probability that the columns X21, X26 may represent the reason "where they would like to support" such as "Student Brook Fund for Excellence", "Graduate School", etc.

5)Stratified Sampling:
Our first model is to predict the column "Mailing Received". Before building the model, we have the imbalanced data set which we need to optimize i.e only 14.77% of 1's in the above column which we need to predict. One more crucial thing which we need to focus is the split of the dataset for training and testing. There are so many sampling techniques discussed in class lecture[12] i.e Uniform Sampling, Random Sampling, Stratified, Random Sampling, Stream Sampling . Out of them, our objective is to select a sampling technique which works best for this particular data set without any kind of bias (for example cost sensitive bias). There are several techniques which were discussed in [8].Stratified sampling is the best technique for this kind of data set. So we have used the stratified shuffle split feature from sklearn to split the data[7] to ensure the data sampled in equal proportion from each group.

6)Model building:
As you can see that the accuracy is not a metric for optimization as the monkey can have 85% accuracy in the prediction of the column "Mailing Received" and more than 90% in the prediction of the column "Mailing Responded". Since, it is a binary classification problem, the metrics precision, recall and the average precision are the important metrics which we need to optimize.

$$precision = \frac{true\ positives}{true\ positives + false\ positives}$$

$$recall = \frac{true\ positives}{true\ positives + false\ negatives}$$

Average Precision is a single number used to summarize a Precision-Recall curve which is defined as the area under curve. One more important technique is that the area under the ROC curve(false positive rate and true positive rate). I have measured all these metrics for all the classifiers which we used to build the model. We have used the standard binary classifiers given in the literature i.e "Logistic Regression, SVM, Random Forest Classifier and Decision Tree Classifier" for predicting the model. Please find the reasons for which we used the above classifiers are:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees.The reason why I used Random Forest Classifier is that this is one of the classifiers which you can directly deal with the categorical data.

Logistic Regression is a discriminative classifier and is a widely used statistical model for binary classification that, in its basic form, uses a logistic function to model a binary dependent variable; many more complex extensions exist. But the problem with this logistic regression is that it doesn't behave good for non-linear seperable data.

Here comes, SVM into picture. One of the best classifiers for the non-linear seperable data, as it is using kernel trick by projecting the features to the different space and using a hyper plane to seperate the data and it comes up with different kernel like rbf kernel, chi-square kernel, gaussian kernel and C hyperparameters where we can use them to fit the model and predict.

Hyper parameter tuning:

Logistic Regression:

We have changed the value of C using GridsearchCV and the best value of C for our sampling of the data which we got is 9.

Random Forest Classifier:

We have tweaked the following hyper parameters to check whether we can optimize the model.

```
param_grid={
   'bootstrap': [True],
   'max_depth': [80, 90, 100, 110],
   'max_features': [2, 3],
   'min_samples_leaf': [3, 4, 5],
   'min_samples_split': [8, 10, 12],
   'n_estimators': [100, 200, 300, 1000]
}
```

We have come up with the above parameters by initially performing a RandomGridsearch and picking the best parameters from that search. This will try out 1 * 4 * 2 * 3 * 3 * 4 = 288 combinations of settings. We can fit the model, display the best hyperparameters, and evaluate performance[13],[14].

SVM:

As mentioned, we have performed the tuning of the parameters by changing the different kernels like "rbf kernel, chi-square kernel, gaussian kernel and the C hyperparameters"

Decision tree learning uses a decision tree to go from observations about an item to conclusions about the item's target value. It is one of the predictive modelling approaches used in statistics, data mining and machine learning.

We have tuned the hyper parameters for each model and presented the metrics below:

Before normalization	precision	recall	F score	Average precision	AUC_FTPR
Logistic_Regression	0.6662	0.4555	0.5411	0.6042	0.9179
Random_Forest_Classifier	0.7896	0.7313	0.7594	0.8031	0.9732
Linear_SVM	0.2810	0.9333	0.4320	0.2707	0.7930
Decision_Tree_Classifier	0.7295	0.7282	0.7289	0.5657	0.8445

Batch Normalization is one of the standard techniques which are employed in data science to get a most important features in the dataset and to improve the performance. Out of many normalization techniques, the technique $(\frac{x-\min(x)}{\max(x)-\min(x)})$ gaves better results.
Please find the results below after applying batch normalization.

After normalization	precision	recall	F score	Average precision	AUC_FTPR
Logistic_Regression	0.6738	0.4912	0.5682	0.6373	0.9288
Random_Forest_Classifier	0.7865	0.7408	0.7630	0.7993	0.9727
Linear_SVM	0.6933	0.4829	0.5693	0.4004	0.7259
Decision_Tree_Classifier	0.7339	0.7270	0.7304	0.5682	0.8443

Out of the techniques discussed in the class[8] to deal with the data imbalance, we have used employ here are "Weigh the minority class more heavily" which is implemented by Bayes Minimum Risk Classifier[9].

cost to fp	cost to fn	Precision	Recall
2	5	0.656	0.9439
10	10	0.7472	0.8279
2	10	0.656	0.9439
5	8	0.704	0.8931

This is a cost sensitive binary classifier where we can provide a cost matrix for false negatives and false positives and setting the cost to true positives and true negatives to 0 and works similar to the one discussed in the class. Here in this case, avoiding False-Negatives are more relevant than avoiding False-Positives. So, we can use this costcla Bayes Minimum Risk Classifier to improve the Recall.

7)Feature Engineering:
As we wish to generate insights from the data which will help us in building a better model, we spent more time on the data and added additional features by taking sum of the all the features i.e by adding "add" column and the square of "add" to introduce the non-linear features where we can introduce the non-linear dependencies such as x_2x_3, x_1x_2, x_1^2. Here you can observe that the Logistic Regression and Linear SVM are behaving better after adding those features. Since all the features are normalized, we can directly pick the top 10 coefficients by performing the custom sort

on the features and coefficients to generate the best 10 features i.e ['X6', 'X42', 'X29', 'X27', 'X33.year', 'X34', 'X35', 'X36', 'X31','X37'].

Please find the results below:

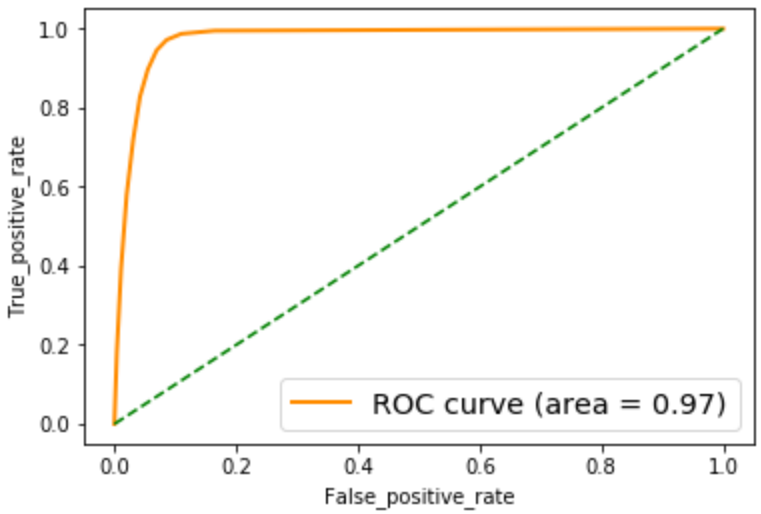
After Features	precision	recall	F score	Average precision	AUC_FTPR
Logistic_Regression	0.6741	0.4918	0.5687	0.6372	0.9289
Random_Forest_Classifier	0.7845	0.7372	0.7601	0.8080	0.9738
Linear_SVM	0.6905	0.4835	0.5687	0.3994	0.7260
Decision_Tree_Classifier	0.7331	0.7298	0.7315	0.5693	0.8456

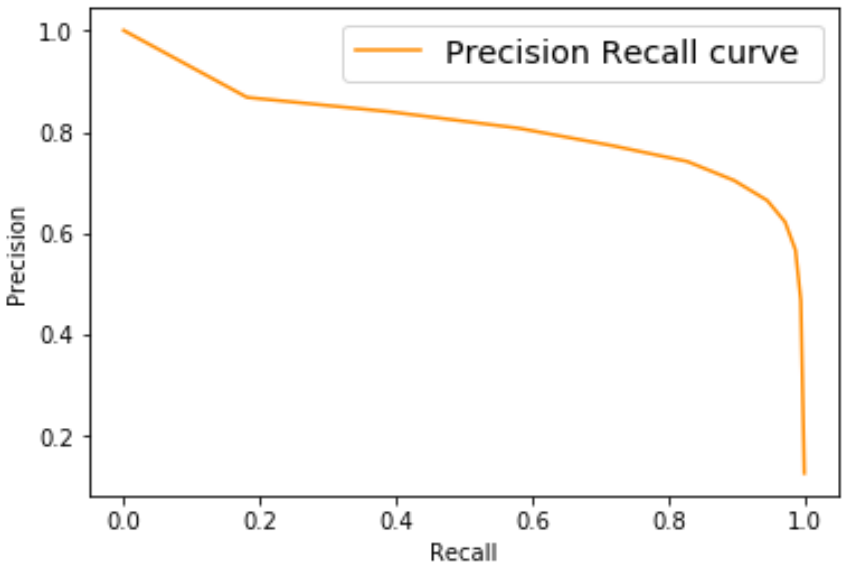
Since our data set is imbalanced, we have used oversampling on our data using SMOTE. We have separated the test data set and applied SMOTE only on the training data set, to achieve the results.

Please find the results below:

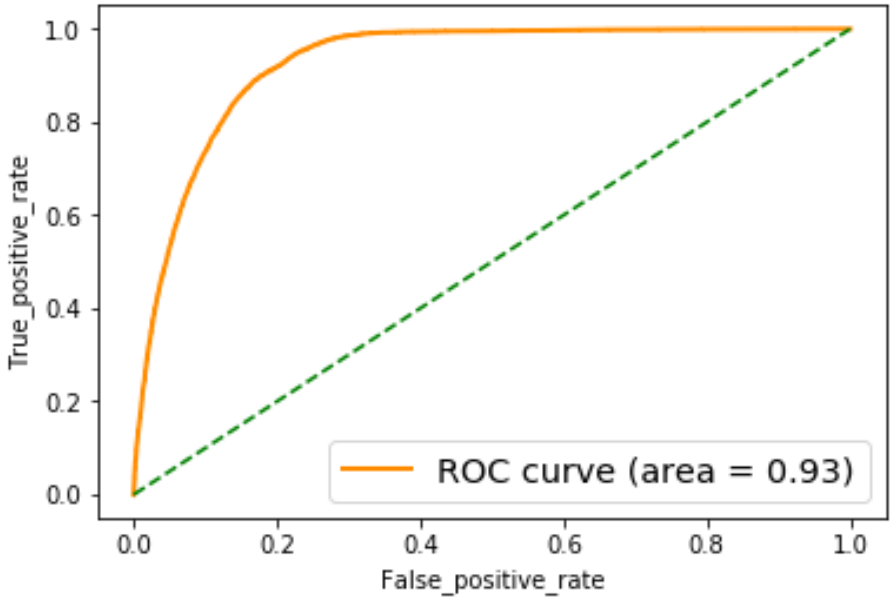
Smote	precision	recall	F score	Average precision	AUC_FTPR
Logistic_Regression	0.4409	0.8788	0.5872	0.6077	0.9274
Random_Forest_Classifier	0.7417	0.8279	0.7825	0.7844	0.9736
Linear_SVM	0.4389	0.8850	0.5868	0.4030	0.8600
Decision_Tree_Classifier	0.7112	0.7466	0.7285	0.5632	0.8512

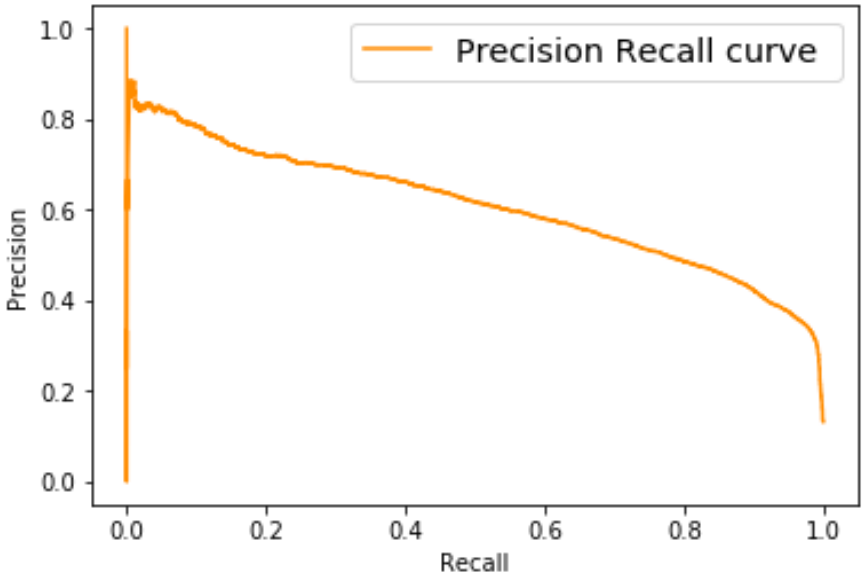
Please find the ROC curve and the precision recall curve for the Random Forest Classifier below:





Please find the ROC curve and the precision recall curve for the Logistic Regression below:

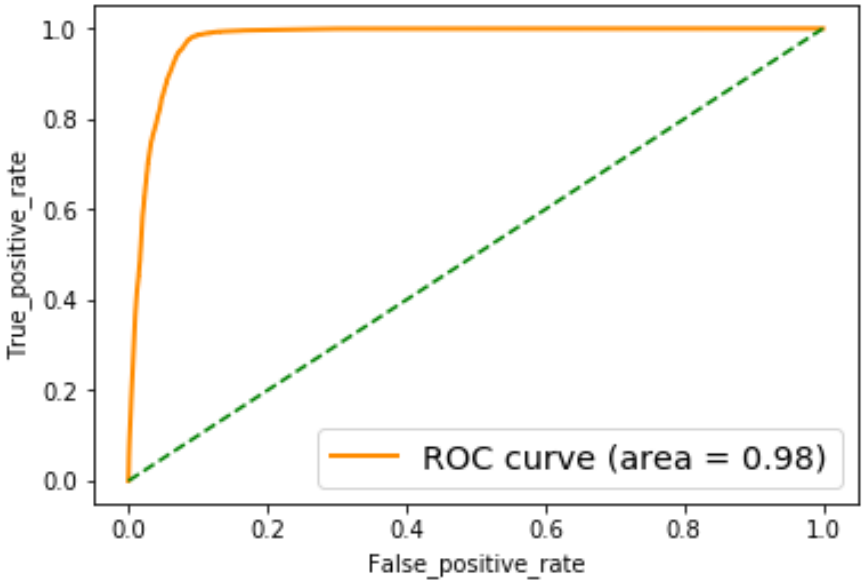


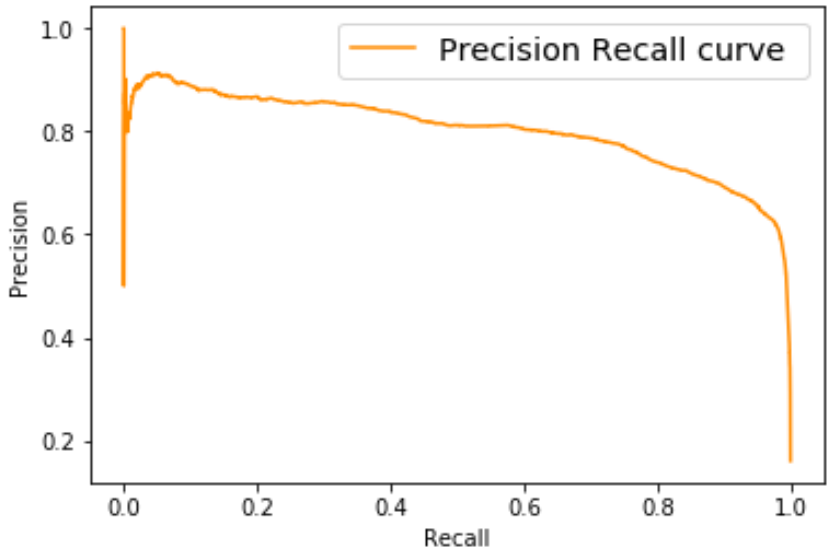


No Free Lunch Theorem:
It basically says that, if you look at all possible problems that you might apply optimization algorithms to, then, on average, any algorithm is just as good or bad as any other. In other words, the theorem says there cannot be a single universal machine learning algorithm that will solve all problems. No free lunch in search and optimization. So, we have tried other classifiers as well like Gaussian Naïve Bayes Classifier, Extreme Gradient Boosting Classifier, Light GBM (Gradient Boosting) classifiers. Out of these four, Extreme Gradient Boosting Classifier is doing better for the prediction of the column i.e "Mailing Received" with good recall.

Imbalance parameter	precision	recall	F score	Average precision	AUC_FTPR
6.8182	0.6118	0.9797	0.7533	0.7821	0.9736

Imbalance parameter is calculated as the number of 0's to the number of 1's and it is close to 6.8 in the 49th column. After having discussion with the professor, we have compensated the model for better precision and better ROC curve area. Please find the ROC curve area below:

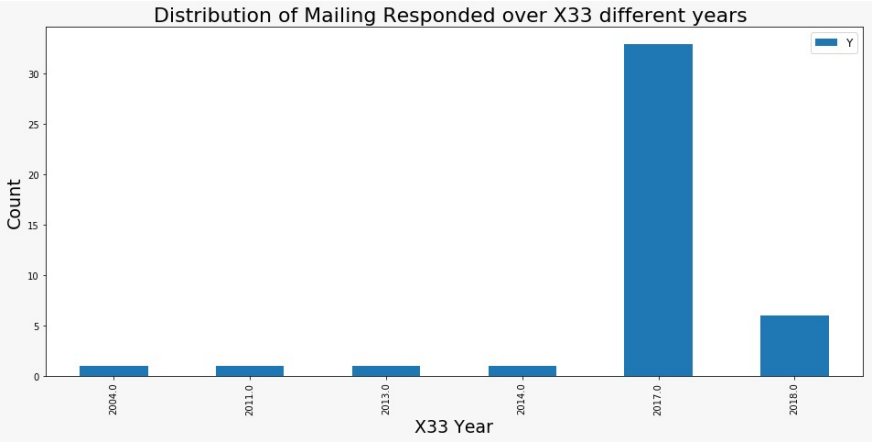


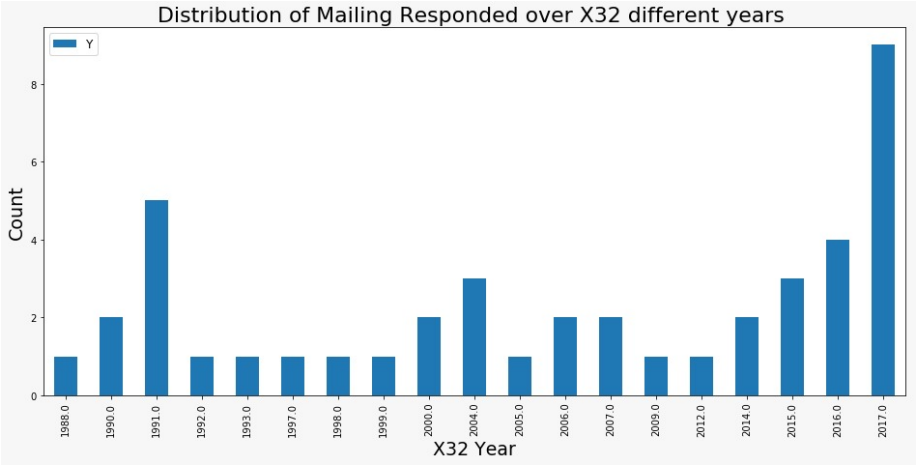


Issues faced:

When we tried to predict the column “Mailing Received” and tried to check the count of 1’s in the “Mailing Responded”, we got only 26 out of 44. So, this will perform badly on the prediction of “Mailing Responded”. So, we worked on the data preprocessing and has removed some features to achieve better model for the final result as it should atleast predict 1’s that correspond to those present in the “Mailing Responded”

Prediction of the 50th column:
When we try to use the best model to predict the last column "Mailing responded" by using the values of the predicted column "Mailing received", it is not behaving that great as the imbalance ratio is very bad.





If you can observe that the column "Mailing responded" in the above graphs, it contains only 45 1's out of 1.7 lakh values, this we should consider this as the anomaly detection.
Techniques we tried to deal with imbalanced dataset are:

1) Discard elements from the bigger class.
2) Replicate members of the smaller class, ideally with random perturbation.

We have tried employing the oversampling and undersampling techniques[10].

K-cross validation:
In general when you have very less data, the , k cross validation is performed to select the hyperparameters or the optimization algorithm. As you can observe the count of 1's in the column "Mailing Responded" as 45, this is exactly same as they have been only 45 U.S. presidents, so any analysis you can do on them represents very small sample statistics.

Similar issues arise in medical trials, which are very expensive to run, potentially yielding data on well under a hundred patients. If I'm using the linear classifier, then the
average performance of these k classifiers trained in stands in as the presumed accuracy
for the full model. We have used K-cross validation using for model selection and error estimation of a model. We have used K=5 and trained the model on the three models "Random Forest Classifier", "Extreme Gradient Boosting", and "lightGBM" using Stratified K fold sampling[15]. Out of these five models, we got better recall and precision values for "Extreme Gradient Boosting". Extreme Gradient Boosting model utilizes the concept of model averaging of weak classifiers and it is performing well for the two columns. Since these are nonlinear classifier and each fold gives rise to different hypothesis altogether, we cannot average the models to achieve better accuracy.

Experiments to deal with the imbalance in data set:
So, we have stick with the "Extreme Gradient Boosting Classifier" and ran five different experiments.

a) Prediction of the column "Mailing Responded" without using all the values in "Mailing received"
b) Prediction of the column "Mailing Responded" using all the values in "Mailing received".
c) Splitting the data set into train and test sets and applying SMOTE analysis (an oversampling method to introduce random perturbations to get the equal proportion of 1's and 0's and used for imbalanced data sets) only on the training data. Prediction of the column "Mailing Responded" using this model.
d) Judicious usage of 1's in the column "Mailing Received" . As you can observe that the data set in the column "Mailing Responded" is dependent on 1's in the column "Mailing Received" and not on 0's. So we have restricted our data set to get the 1's in the column "Mailing Received" and tried to predict the column "Mailing Responded" as we can get better imbalance.
e) SMOTE analysis with the judiciary use of 1's in the column "Mailing Received".

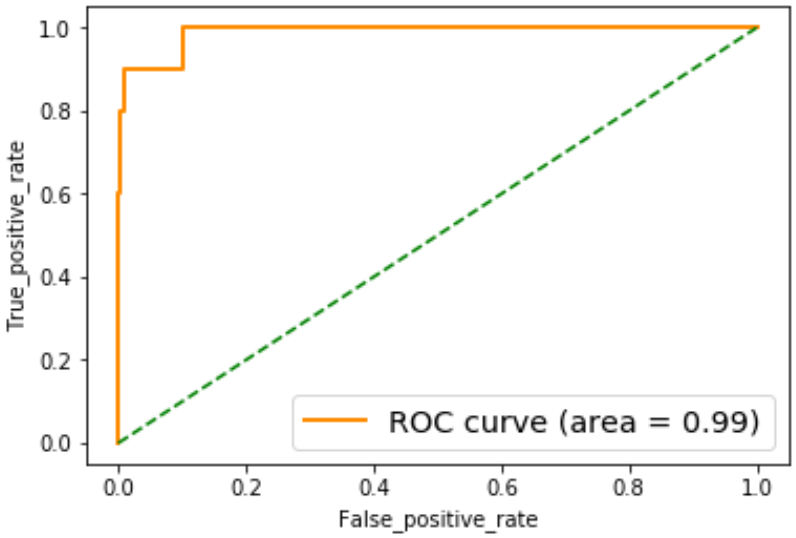
We have tuned the hyper parameter "scale_pos_weight" to deal with the imbalance when we are not applying SMOTE. Please find the results for the above analysis.

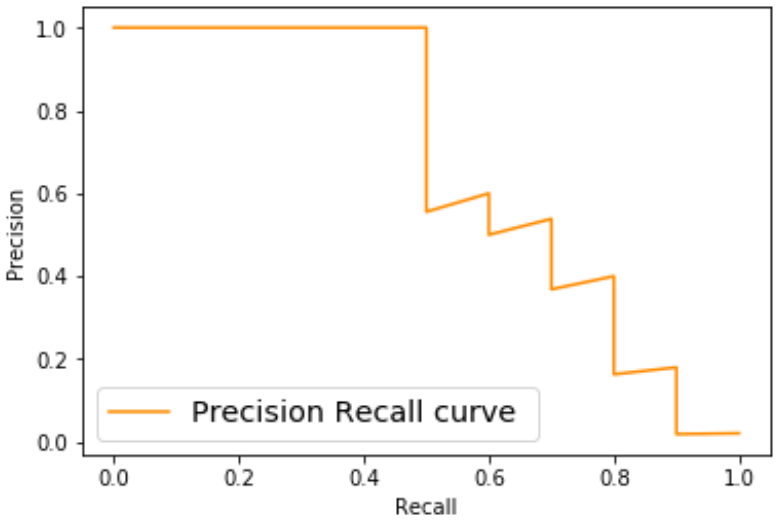
	precision	recall	F score	Average precision	AUC_FTPR
a	0.0189	0.6154	0.0366	0.0248	0.9827
b	0.0560	0.7778	0.1045	0.1156	0.9915
c	0.0753	0.7778	0.1373	0.3528	0.9964
d	1.0000	0.4444	0.6154	0.6386	0.9766
e	0.1765	0.6667	0.2791	0.5795	0.9555

To come up with the best method for these five experiments, we have used the K cross validation and got the above results. Out of all these five, we have picked the method d as it is decent with the recall and performing great with the precision. After that we tuned the hyper parameters of this using GridsearchCV and achieved the best results using the following hyperparameters i.e "learning_rate=0.2, depth = 5". Please find the final results for the column "Mailing Responded" below:

precision	recall	F score	Average precision	AUC_FTPR
0.7143	0.5	0.5882	0.6739	0.9883

Please find the ROC plot and precision recall plot below:





Conclusion:

This project has helped us to come up with the various techniques to deal with the imbalanced data set which is the prevalent in different fields. We started with the standard binary classifiers and then checked non-linear classifiers i.e Extreme Gradient Boosting which takes advantage of several weak classifiers and played very judicially by tweaking the hyper parameters to get the higher recall without comprising much for the precision.

9)References:

[1] https://www.stonybrook.edu/campaign/

[2]https://datascience.stackexchange.com/questions/32818/train-test-split-of-unbalanced-dataset-classification

[3] https://stats.stackexchange.com/questions/254710/is-it-better-to-compute-a-roc-curve-using-predicted-probabilities-or-distances-f

[4] https://www.coursera.org/lecture/competitive-data-science/visualizations-zoIx3

[5]https://www.coursera.org/lecture/competitive-data-science/exploring-anonymized-data-qJHOb

[6]https://www.pdpc.gov.sg/-/media/Files/PDPC/PDF-Files/Other-Guides/Guide-to-Anonymisation_v1-(250118).pdf

[7]https://scikitlearn.org/stable/modules/generated/sklearn.model_selection.StratifiedShuffleSplit.html#sklearn.model_selection.StratifiedShuffleSplit

[8]https://docs.google.com/presentation/d/1oXGmFxOTNS0lRrNeDp5HHUtCPIxMxpGWRosy4LqlMw/edit#slide=id.g13a73706d1_1_49

[9] http://albahnsen.github.io/CostSensitiveClassification/BayesMinimumRiskClassifier.html

[10] https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis

[11] https://beckernick.github.io/oversampling-modeling/

[12]https://docs.google.com/presentation/d/17wpU8sWcH8u7Bhfpnaw2TKLrvRzCGgjPyqzGB4G0SM/edit#slide=id.g13cba946af_0_15

[13]https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

[14]https://medium.com/all-things-ai/in-depth-parameter-tuning-for-random-forest-d67bb7e920d

[15] https://stats.stackexchange.com/questions/52274/how-to-choose-a-predictive-model-after-k-fold-cross-validation

[16]https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold