**Feature Importance and Ensemble Learning in Patient Readmission**

James Ho[[1]](#footnote-0), Haofeng Huang[[2]](#footnote-1), Yuran Pan[[3]](#footnote-2), Xuming Wang[[4]](#footnote-3), and Siyang Xiao[[5]](#footnote-4)

Department of Computer and Information Science, Fordham University, New York

**Abstract**

Hospitals and medical professionals hold the ability to predict patient readmission as an important research topic. For this reason, there is a good reason to provide and create a model which can successfully predict model readmittance. To address this problem, our research seeks to build a strong classifier to successfully classify patients who have a chance at being readmitted. To do this, we are using the following models: Logistic regression, K nearest neighbors, support vector machine, random forest in order to find insight.

**1. Introduction**

Patient readmission to hospitals is both time consuming and expensive for the patient and the hospital. Therefore, it can be important for physicians, and medical experts to identify key features to examine further as well as understand what causes for patient readmission to reduce costs to the patients and medical institutions. For this reason, in this project we seek to build models that use a variety of feature selection approaches as well ensemble learning techniques to measure the effectiveness of individual models and a combination of them through meta-learners such as voting.

To understand the dataset and topic further, we looked at previous research into the topic of patient readmission and their modeling approaches. From their findings specifically Strack et al (2013), we used some of their methods to do preprocessing which will be outlined below.

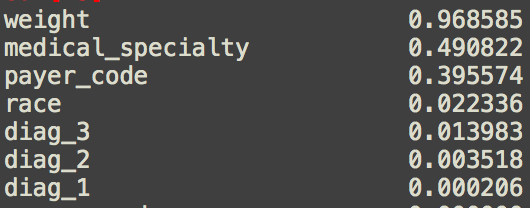
For the purpose of this project, we are exploring feature selection and importance measurements. As well as the role feature selection plays in model building and performance.

**2. Pre-Processing**

Before doing any feature selection we performed exploratory data analysis, handling missing values and encoding categories feature This will be elaborated on below.

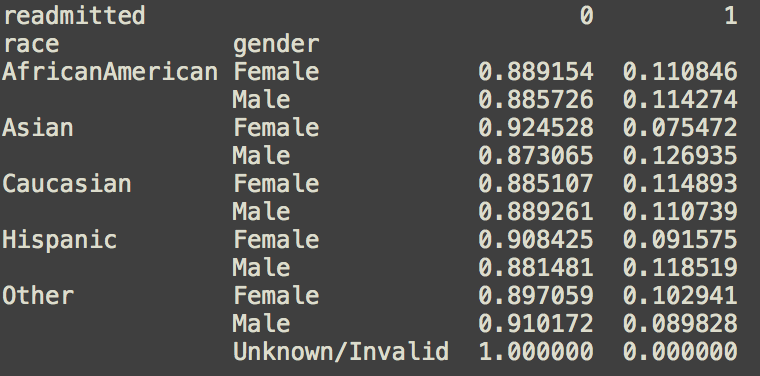
A. Dataset

The patient readmission dataset contains a total of 101,766 data points. The ratio within the dataset of patient readmission and no readmission is about 1:9. A total of 7 features contain missing data.



*Missing values by percent*

A cross-table on demographics indicates that Asian females have the lowest chance of being readmitted, while Asian males have the highest. However, there are only 641 (0.6%) patients that are Asians and this sample population size is not enough to support this finding.



*Readmitted (by race and gender)*

B. Handling Missing Data

In the data, we encounter some missing or unknown values such as 3 unknown gender points. That was removed along with the feature weight which is 97% missing. Payer\_code is dropped as it is irrelevant to our topic of interest. We dropped the entire records on patients with missing values (1485 counts) in diag\_1, diag\_2, and diag\_3. We impute the missing data in a race using the mode value of race feature, which is Caucasian. Although medical\_specialty has a high missing value percentage, we decided to include it because this information is medical related and may have an influence on the prediction. We filled in this feature based on the mode of each age group.

There are 2.2% (1506 counts) missing value in the race, we merge it to Caucasian which is the mode.

C. Merging

Due to large count of unique category in some feature, we merge some categories to reduce the cardinality. These are:

|  |  |  |
| --- | --- | --- |
| **Feature** | **Value** | **Count** |
| admission\_source\_id | Admitted because of physician/clinic referral | 29460 |
|  | Admitted from emergency room | 54853 |
|  | Others | 12553 |
| Admission\_type\_ID | Discharged to home | 59005 |
|  | Others | 37861 |
| Age | 30-60 | 29530 |
|  | 60-100 | 65230 |
|  | <30 | 2006 |
| Medical\_speciality | Cardiology | 5129 |
|  | Family/GeneralPractice | 6953 |
|  | InternalMedicine | 61765 |
|  | Others | 18279 |
|  | Surgery | 4740 |
| diag\_1 | Circulatory | 29166 |
|  | Respiratory | 7954 |
|  | Digestive | 9128 |
|  | Diabetes | 4918 |
|  | Injury | 6646 |
|  | Musculoskeletal | 4793 |
|  | Genitourinary | 3061 |
|  | Neoplasms | 17508 |
|  | Other | 13692 |
| diag\_2 | Circulatory | 30315 |
|  | Respiratory | 11900 |
|  | Digestive | 4018 |
|  | Diabetes | 8060 |
|  | Injury | 2337 |
|  | Musculoskeletal | 1748 |
|  | Genitourinary | 2307 |
|  | Neoplasms | 25918 |
|  | Other | 10263 |
| diag\_3 | Circulatory | 28946 |
|  | Respiratory | 16749 |
|  | Digestive | 3817 |
|  | Diabetes | 6380 |
|  | Injury | 1874 |
|  | Musculoskeletal | 1875 |
|  | Genitourinary | 1649 |
|  | Neoplasms | 28647 |
|  | Other | 6929 |

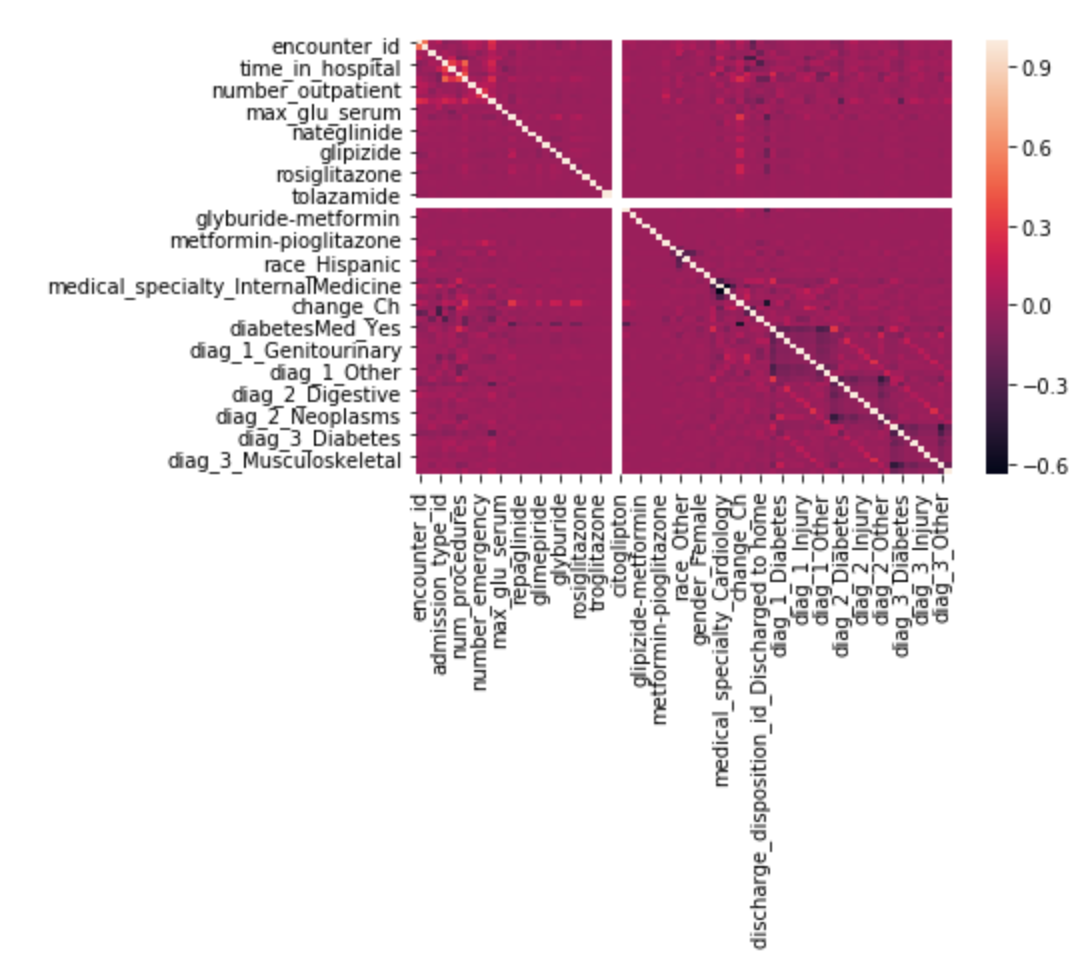
D. Features and merging

Discharge\_disposition\_id: We dropped patients that indicate death (3415 counts) to avoid indivdiuas who would not be readmitted.The following codes indicated dead or hospice care from the code sheet are ['11','12','13','14','19','20','21','25','26'])

We drop encounter\_id and patient\_nbr because we create a new feature that takes in the information from both of them.

We then drop the features below: 'acetohexamide',glimepiride-pioglitazone’,'metformin-rosiglitazone','metformin-pioglitazone','chlorpropamide','tolbutamide','acarbose','miglitol','troglitazone', 'tolazamide' , 'examide', 'citoglipton', 'glyburide-metformin','glipizide-metformin'

This is because these features are either very sparse or hold a correlation of zero.



*Correlation Matrix between(partial) features before cleaning*

E. Encoding

We apply one-hot encoding method, a process by which categorical variables are converted into new features that are binary. This process allows machine learning algorithms and feature scoring algorithms to identify key features that are more important and specific rather than something general. Thus we use sklearn package,skearn.preprocessing.OneHotEncoder to achieve this goal.

F. Feature Engineering

While there are duplicates of patient\_Nbr which indicates the person, the encounter\_id is unique which can be treated as when a patient came to the hospital. Instead of dropping the records with duplicated Patient\_Nbr, we retain all the records and create a new feature called “Return” from using Patient\_Nbr and encounter\_id. We treated a patient encounter\_id which is lower as a first visit in order to come up with return feature. 0 indicates that this is the patient’s first time of visiting the hospital, and 1 indicates the patient has been enrolled in the hospital before.

G. Summary of the Feature Processing

After processing the data we ended up with data points which are 96,866 records and 87 features. While we did not remove returning patients like Strack et al (2013) in order to deal with the data point independence problem for logistic regression. We instead treat each encounter as the independent values and for this reason, Logistic Regression’s independence is not violated.

**3. Techniques Used**

In order to identify feature importance in model building, we look into various approaches one can take to the selection procedure.

A. Imbalance Data and SMOTE (Synthetic Minority Oversampling Technique)

As stated before, our dataset is imbalance and subject to problems for measuring the accuracy and performance of the model. More important, since the data is imbalanced we must also take into account the problem of dealing with the imbalance in our training step. In order to prevent creating a bias in our machine learning models. Our data mostly consists of individuals who are not readmitted to the hospital and a minority of people who are readmitted we proposed the use of SMOTE on the training set in order to better train the models.

SMOTE proposed by Chawla et al. (2002) is a method to deal with imbalance data by oversampling the minority. Instead of replicating the minority classifier it synthetically oversamples the minority classifier until a ratio set by the experimenter is reached usually 1:1 using the KNN algorithm with K set to 5 by default.

The importance of SMOTE for the rest of the project is to deal with the data imbalance. This implementation was done for both our Feature Selection and Tuning approach though there are other methods such as bagging and undersampling. We used the package from Imbalance Learn to implement SMOTE. The use of SMOTE is important to our Feature Selection as certain classifiers specifically KNN requires a balanced dataset to trained properly which is what is purported by Blagus and Lusa (2013).

B. Feature Selection: Filter Method  
Feature selection is a research area with a variety of approaches taken in order to determine which set of features should be used in model building. As outlined by Guyon and Elisseeff (2003), one should use expert opinion if possible to select features in the model building. They also outline the use of other feature approaches given no expert opinion which is the filter, wrapper, and embedded approach. Both of these methods are used as workarounds to the problem of (2^n) – 1 test of every feature. In this project, we used some of the more common approaches derived from filtering and wrapper as outlined by Guyon and Elisseeff (2003).

For this project, we used a global feature filter approach, using the scoring mechanisms of different models and cross-validation with k-fold = 5. It is important to note that the use of filter methods usually involves a manual set cut off choosing where to cut off the number of features. In our case, we set the cutoff to k = 10 features. We saw that in our calculation for the global unweighted f1 score around 10 features gave the better performance for most models. Using the 89 features, we scored them using different techniques that are present in the sklearn and scipy library. This included the use of Correlation, Chi-Squared, Mutual Information, the mean of these three measurements as well as the rank of these three measurements.

Regarding the use of SMOTE, its effects on the models and feature selection methods. We must note that it is purported by Blagus and Lusa (2013) that SMOTE affects the correlation between samples from the minority class and does not affect the correlation between variables. Therefore our EDA analysis still holds as it was done before the SMOTE step. They also purport that SMOTE creates problems for classifiers that assume independence amongst samples such as logistic regression. Lastly certain feature selection methods are affected by SMOTE as there is an assumption of independence.

C. Feature Selection: Wrapper Method

In addition to implementing a filter feature selection, we also implemented a wrapper forward feature selection method with SMOTE. Wrapper methods are computationally expensive as they are a greedy algorithm as outlined by Guyon and Elisseeff (2003). Wrapper methods pick features based on a predictability score and often perform better but take more time to calculate for. For this project, instead of using the sklearn’s overall accuracy measurement to pick features, we pick features sequentially through the f1 score. This ensures that we pick features that optimize for a balanced score rather than just an overall accuracy as the data is imbalanced. Within wrapper approach there are two methods, one is called sequential forward selection and sequential backward selection. Both of these processes involve a base calculation and taking subsets that give optimal results in order to choose the set of features. We first implemented sequential forward selection and these were the results we obtained for each model:

D. Modeling and Tuning

In the model tuning parts, we have tried the parameter below to reach the best performance:

Logistic Regression  
C = [ 0.01, 0.1, 1, 10, 100]

Perceptron

alpha = [0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0, 10, 100, 1000]  
  
Linear SVM  
parameters\_lin\_svm = {'linearsvc\_\_C': C}

Random Forest

c45\_param = {'randomforestclassifier\_\_n\_estimators': [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000]}

K-nearest neighbors  
k = [1, 3, 5, 11, 21, 41, 61, 81]

At the end,we got a collection of tuned classifiers for ensemble.

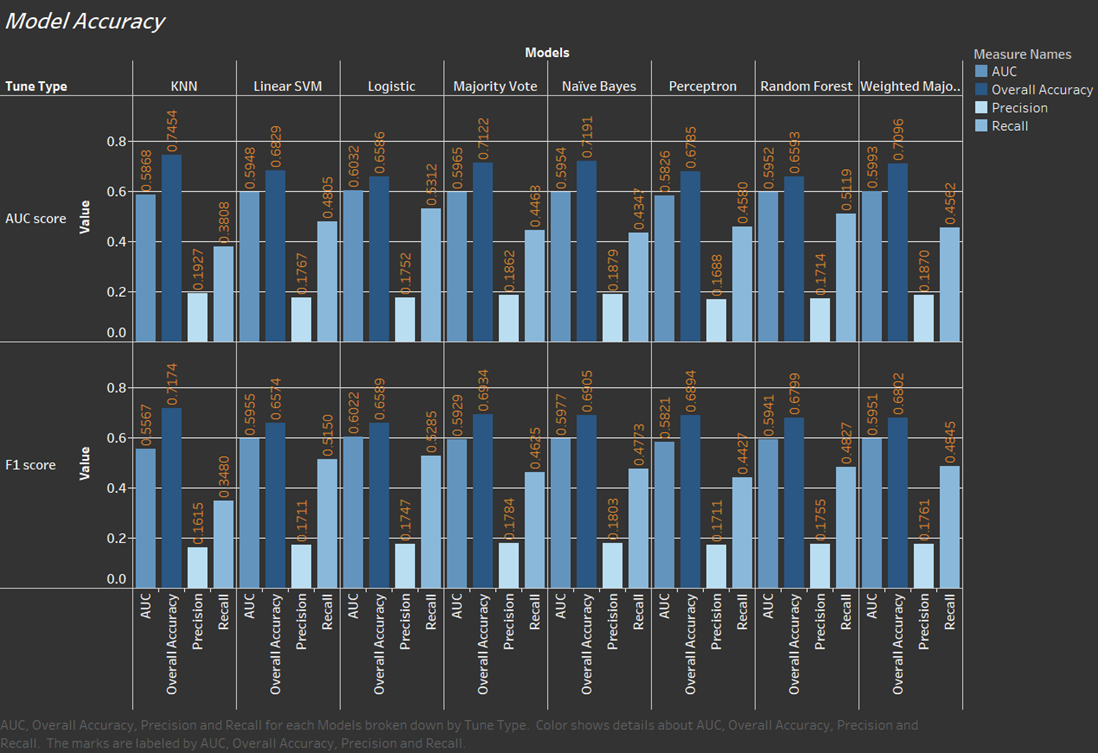
**5. Ensemble of Models**

For our voting approach, we did a simple vote system where all six models had equal say in prediction. We also implemented a weighted vote for the models where we used the cross cross-validated f1 score to determine the weights of each classifier. This was done by normalizing F1 to sum to 1 which gives the models each weight which can add to 1.

**6. Evaluation of Processes**

A. Overall Accuracy

Below we can see some results of our model. It outlines the scores two models Wrapper features selected based on F1 score improvement and tuned differently. One is tuned to F1 score and the other is tuned to AUC score from ROC.

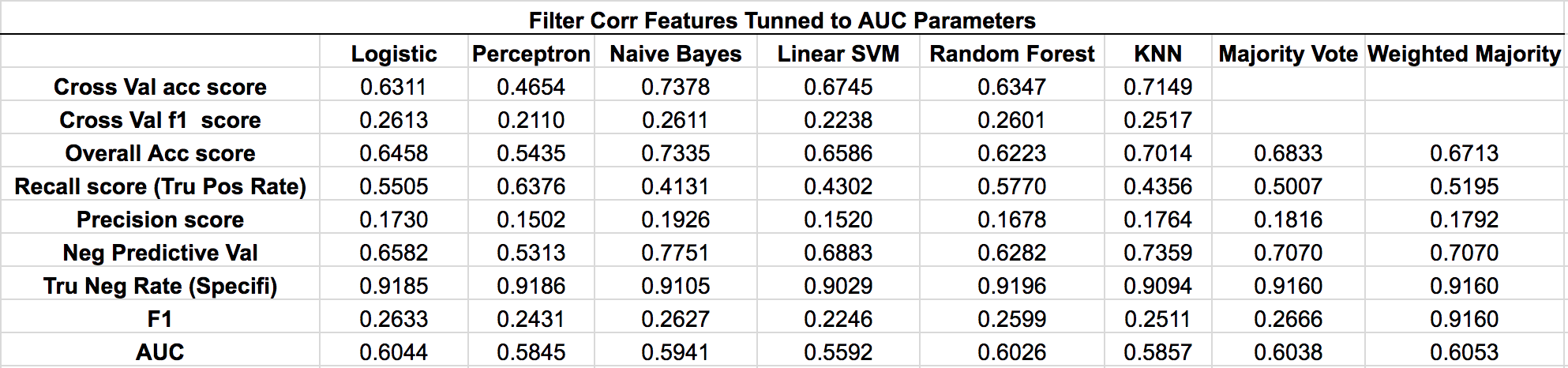


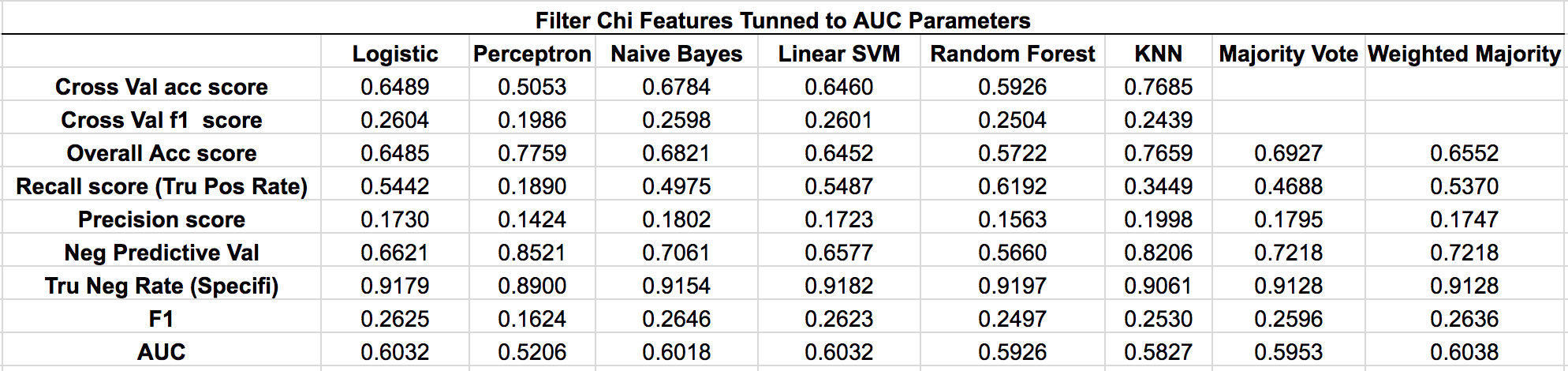
*B. Precision, Recall (True Positive Rate), Selectivity, and Specificity (True Negative Rate)*

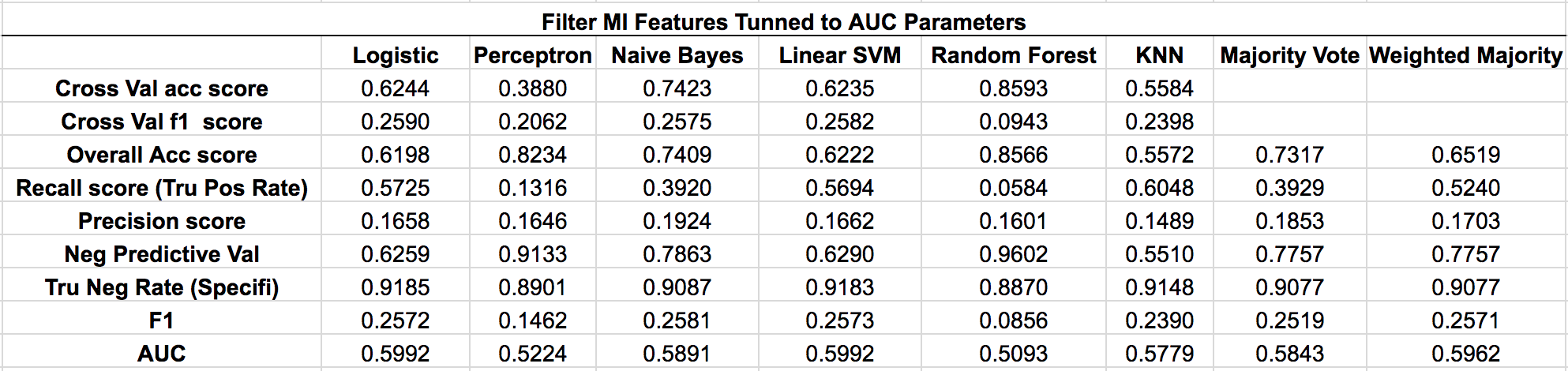
For each model that was created, we evaluated them using the following scoring methods. Each of which evaluates the ability of it to classify labels.

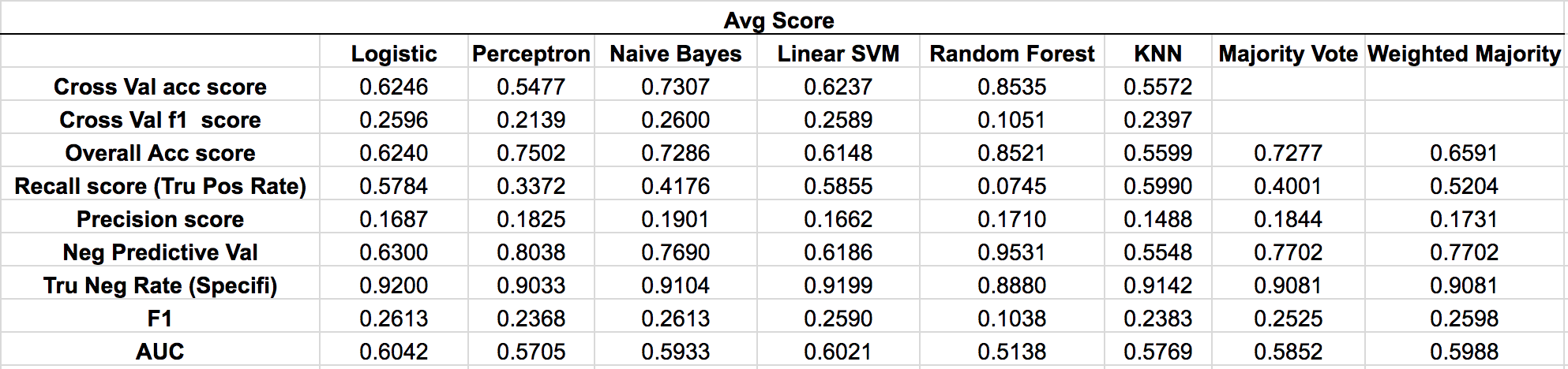
C. Results

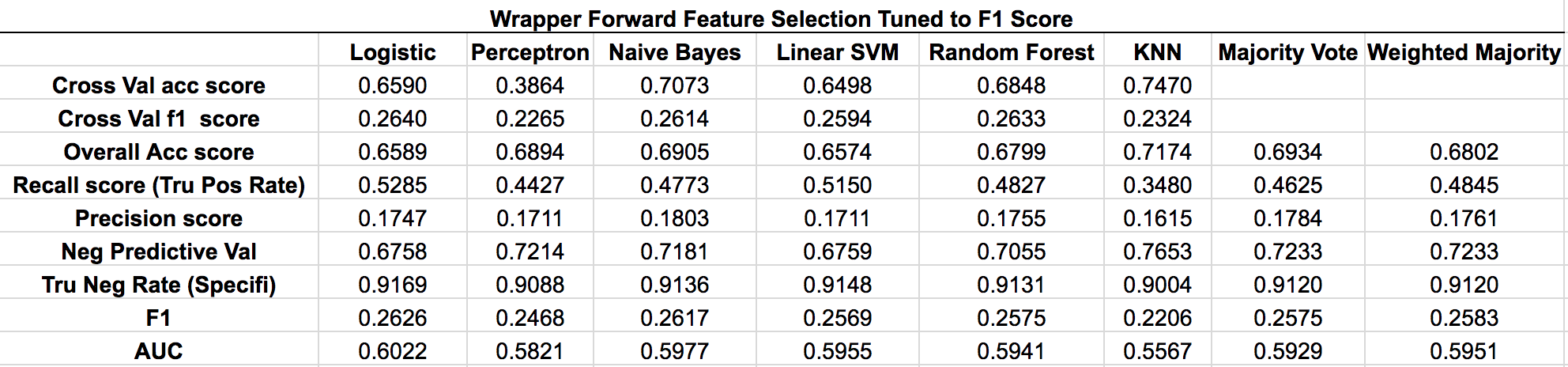
Below are results based on the formulas above:











**7. Discussion, Summary, and Conclusion**

A. Unaddressed Problems

SMOTE’s effectiveness in dealing with imbalance data can be questioned. Although it is the method we choose to use there are alternatives to combat imbalance data such as undersampling, hybrid sampling, and bagging.

The problem of feature selection in relation to SMOTE. While SMOTE can be seen as the minimal answer to the question of feature selection in imbalance data we affect certain scoring algorithms like correlation in addition to this there have been other proposed methods for imbalanced data feature selection such as the one outlined in Yin et al. 2013 (Feature selection for high-dimensional imbalanced data) which propose the use of pseudo subclasses during feature selection. Lastly, SMOTE has been argued to only aid in increasing performance of KNN which was argued in Blagus and Lusa 2013 (SMOTE for high-dimensional class-imbalanced data) and may only serve as a band-aid to the problem

Although a wrapper technique was applied in one set of features.There exists scenarios were our scoring algorithms did not capture relationships between the target and the features. Our scoring algorithms only captured relationships between single features and target y.

Interoperability of the model should be questioned as there is some multicollinearity which is also observed in our different filter scoring methods. Similar features are selected which are originally from the same feature which was converted in one hot.

B. Model Limitation

The models we implemented are linear classifiers, which indicates the data might not be linearly separable. Based on the result of the ensemble which doesn’t have significant improvement, all algorithms we implemented might have similar classification results.

**9. Future Work**

A. Feature Selection and Tuning

Using a backward wrapper method to generate different feature selection results, and then tuned them using that implementation.In regards to tuning a higher CV k fold should have been implemented given more time and computational power. The consideration for using principal component and linear discriminant analysis instead of feature selection and apply all the algorithms and ensemble method to selected PCs and important features.

B. Lineararity Problem and Non Linear Classification

Focus on nonlinear approaches should be looked at again through Kernels in SVM and a less discriminative approach through generative neural networks which may be able to better separate our labels.

C. Enssemble Question

We can revisit the use of different boosting methods which address imbalance data such as SMOTEBoost, Random Undersampling Boosting, and Undersampling Boosting. In addition to across model boosting like combining generative models which may better perform.

1. [jho9@fordham.edu](mailto:jho9@fordham.edu) [↑](#footnote-ref-0)
2. [hhuang96@fordham.edu](mailto:hhuang96@fordham.edu) [↑](#footnote-ref-1)
3. [ypan73@fordham.edu](mailto:ypan73@fordham.edu) [↑](#footnote-ref-2)
4. [xwang423@fordham.edu](mailto:xwang423@fordham.edu) [↑](#footnote-ref-3)
5. [sxiao14@fordham.edu](mailto:sxiao14@fordham.edu) [↑](#footnote-ref-4)