Supervised Learning Capstone

BY: FRANCES CUE

Background

I’ve acquired a Diabetic Readmission Data Set from kaggle. The dataset is composed of integrated data from 130 U.S. hospitals over the course of 10 years. It includes over 50 features such as length of stay, medications, lab results and primary diagnosis. In the healthcare field, determining the chances of a patient readmission is vital. If risks for readmission is known, better treatment plan can be created for patients.

Objective

To create various models to determine diabetes readmission prediction less than 30 days of previous admission.

Exploratory Data Analysis

- ❖Categorical values includes gender, race, age (which were group into counts of 10), medications, diagnosis via ICD9 codes, medical specialty of admitting doctor, medication changes and more.
- ❖Numerical columns included patient identifiers such as encounter id and patient number, admission type, number of times a patient was inpatient, outpatient, in emergency, procedures, time in hospital and number of medications.

Object Types

We have a mixture of 37 categorical values and 13 numerical/continuous values:

```
num_cols.dtypes
```

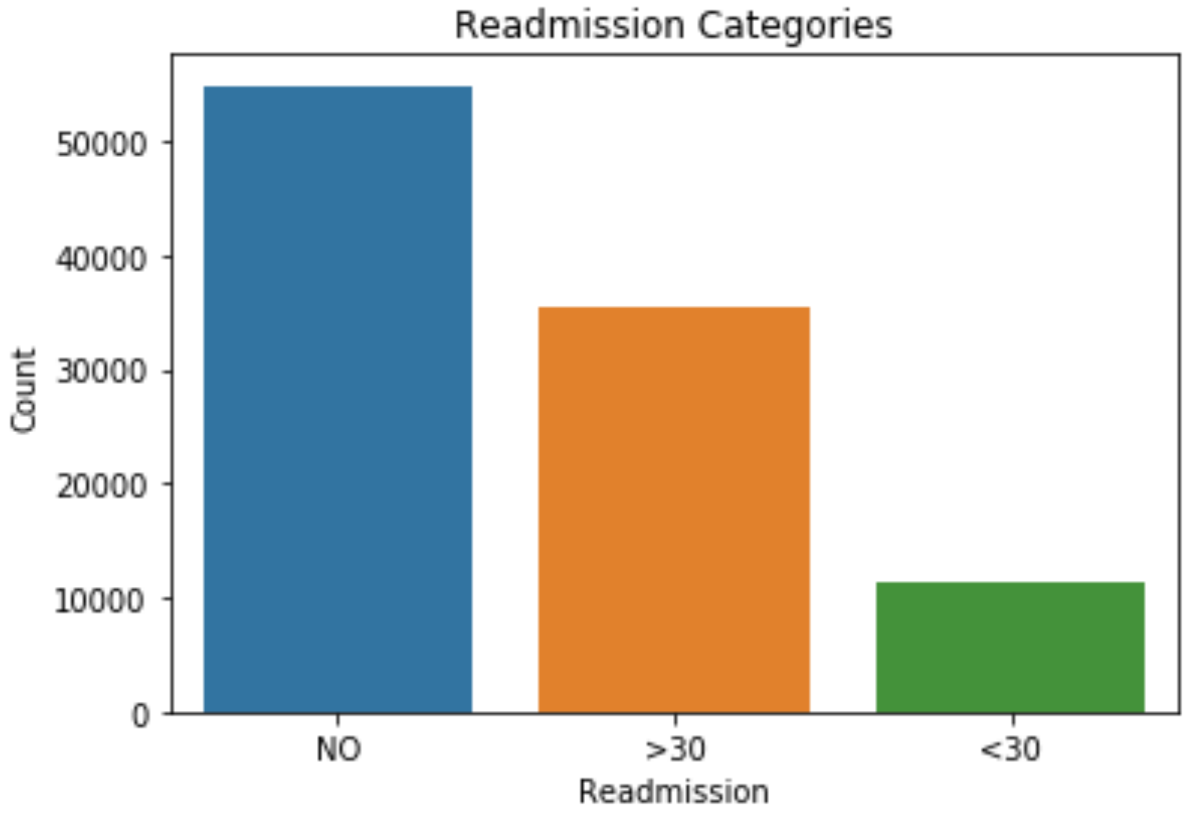
```
encounter_id                int64
patient_nbr                 int64
admission_type_id           int64
discharge_disposition_id    int64
admission_source_id         int64
time_in_hospital            int64
num_lab_procedures          int64
num_procedures              int64
num_medications             int64
number_outpatient           int64
number_emergency            int64
number_inpatient            int64
number_diagnoses            int64
target                      int64
age_group                   int64
dtype: object
```

```
cat_cols.dtypes
```

```
race                        object
gender                      object
age                         object
diag_1                      object
diag_2                      object
diag_3                      object
max_glu_serum               object
AlCresult                   object
metformin                   object
repaglinide                 object
nateglinide                 object
chlorpropamide              object
glimepiride                 object
acetohexamide               object
glipizide                   object
glyburide                   object
tolbutamide                 object
pioglitazone                object
rosiglitazone               object
acarbose                    object
miglitol                    object
troglitazone                object
tolazamide                  object
examide                     object
citoglipton                 object
insulin                     object
glyburide-metformin         object
glipizide-metformin         object
glimepiride-pioglitazone    object
metformin-rosiglitazone     object
metformin-pioglitazone      object
change                      object
diabetesMed                 object
readmitted                  object
dtype: object
```

Readmission Categories

The target chosen for this project is to predict readmission in less than 30 days after previous admission



Looking for missing values

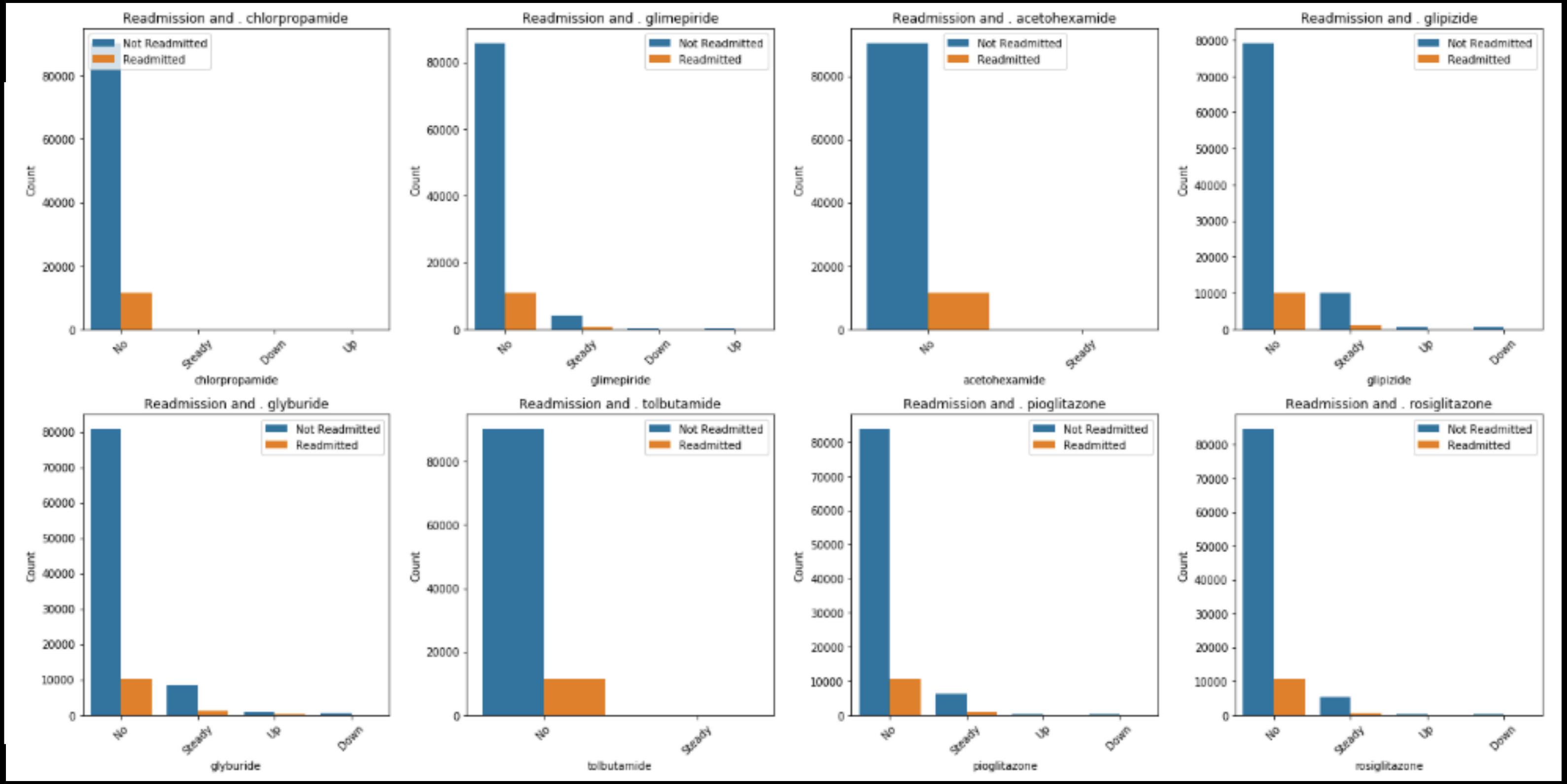
The dataset contained no null values upon first examination. However, it had many ‘?’ values that represent missing values.

Columns with over 90% missing were dropped. Other columns that provide no insight such as insurance payer codes were dropped as well.

```python
#looking for ? values.
for col in diabetes.columns:
    if diabetes[col].dtype == object:
        print(col,diabetes[col][diabetes[col] == '?'].count())
```

```
race 2273
gender 0
age 0
weight 98569
payer_code 40256
medical_specialty 49949
diag_1 21
diag_2 358
diag_3 1423
max_glu_serum 0
A1Cresult 0
metformin 0
repaglinide 0
nateglinide 0
chlorpropamide 0
```

There were many medications (27) included in the dataset. Visualization helped determine relationship between target.



Feature Engineering

```
{'[0-10)':0,
 '[10-20)':10,
 '[20-30)':20,
 '[30-40)':30,
 '[40-50)':40,
 '[50-60)':50,
 '[60-70)':60,
 '[70-80)':70,
 '[80-90)':80,
 '[90-100)':90}
'age_group'] = diabete
```

- New feature was created for age since it was grouped by 10s. This in turn created a new numerical value for age.
- With discharge dispositions, patients who have expired or was sent to hospice were dropped because there will be no chance for readmission.

Grouped diagnosis by counts/occurrence

	Diagnosis	ICD_code	count
0	Heart Failure	428	6862
1	Other forms of Chronic Heart Disease	414	6581
2	Symptoms involving respiratory system and othe...	786	4016
3	Myocardial infarcation	410	3614
4	Pneumonia	486	3508
5	Cardiac Dysrythmias	427	2766
6	Emphysema	491	2275
7	Osteoarthritis	715	2151
8	Cellulitis	682	2042
9	General Symptoms	434	2028

	2nd_diagnosis	ICD_code	count
0	Disorders of fluid electrolyte and acid-base b...	276	6752
1	Heart Failure	428	6662
2	Diabetes mellitus without mention of complicat...	250	6071
3	Cardiac Dysrythmias	427	5036
4	Essential Hypertension	401	3736
5	Chronic airway obstruction, not elsewhere clas...	496	3305
6	Other disorders of urethra and urinary tract	599	3288
7	Hypertensive chronic kidney disease	403	2823
8	Other forms of chronic ischemic heart disease	414	2650
9	Other acute and subacute forms of ischemic hea...	411	2566

	Other_diagnosis	ICD_code	count
0	Diabetes mellitus without mention of complicat...	250	11555
1	Essential Hypertension\t	401	8289
2	Disorders of fluid electrolyte and acid-base ...	276	5175
3	Heart Failure	428	4577
4	Cardiac Dysrythmias	427	3955
5	Other forms of chronic ischemic heart disease	414	3664
6	Chronic airway obstruction, not elsewhere clas...	496	2605
7	Hypertensive chronic kidney disease	403	2357
8	Chronic Kidney Disease	585	1992
9	Disorders of lipoid metabolism	272	1969

Baseline Models

Scores for measurements

AUC

Accuracy

Recall

Precision

Specificity

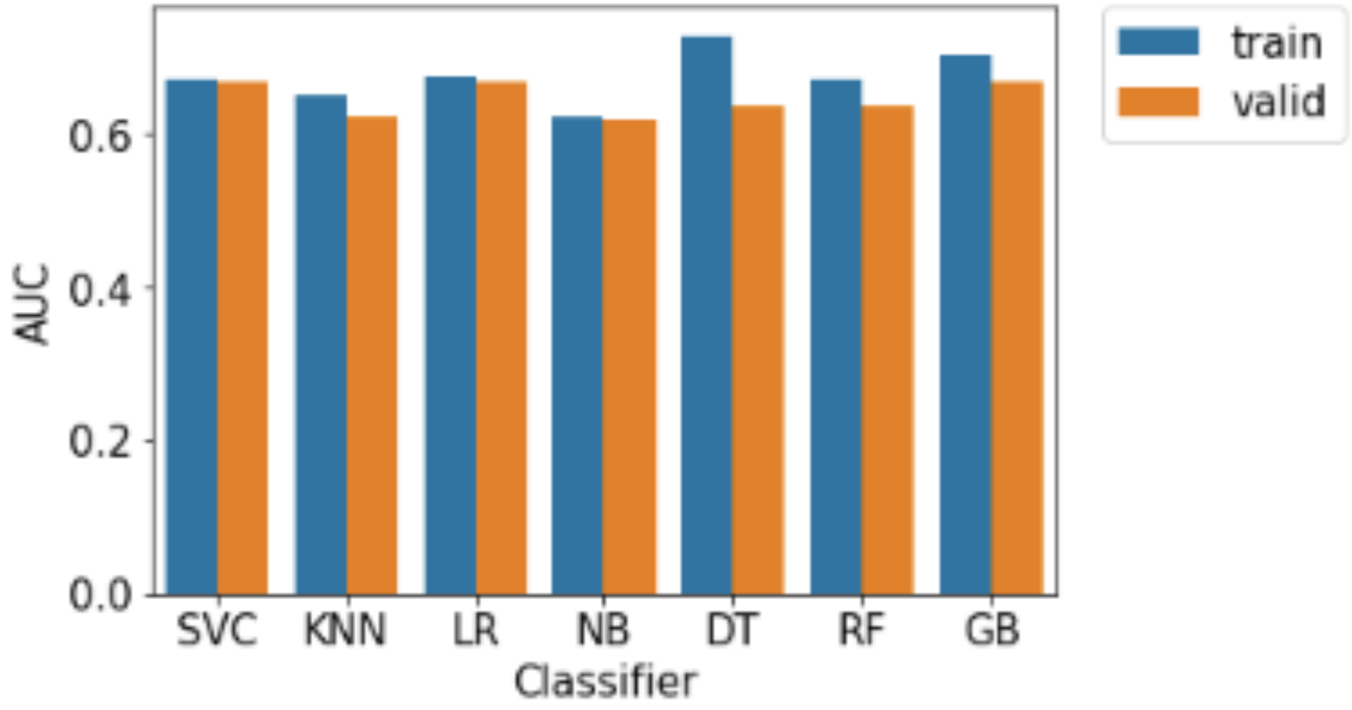
```python
def calc_specificity(y_actual, y_pred, thresh):
    # calculates specificity
    return sum((y_pred < thresh) & (y_actual == 0)) /sum(y_actual ==0)

def print_report(y_actual, y_pred, thresh):

    auc = roc_auc_score(y_actual, y_pred)
    accuracy = accuracy_score(y_actual, (y_pred > thresh))
    recall = recall_score(y_actual, (y_pred > thresh))
    precision = precision_score(y_actual, (y_pred > thresh))
    specificity = calc_specificity(y_actual, y_pred, thresh)
    print('AUC:%.3f'%auc)
    print('accuracy:%.3f'%accuracy)
    print('recall:%.3f'%recall)
    print('precision:%.3f'%precision)
    print('specificity:%.3f'%specificity)
    print('prevalence:%.3f'%calc_prevalence(y_actual))
    print(' ')
    return auc, accuracy, recall, precision, specificity
#setting threshold to 50% since our data is now balanced
thresh = 0.5
```

Baseline Model Scores

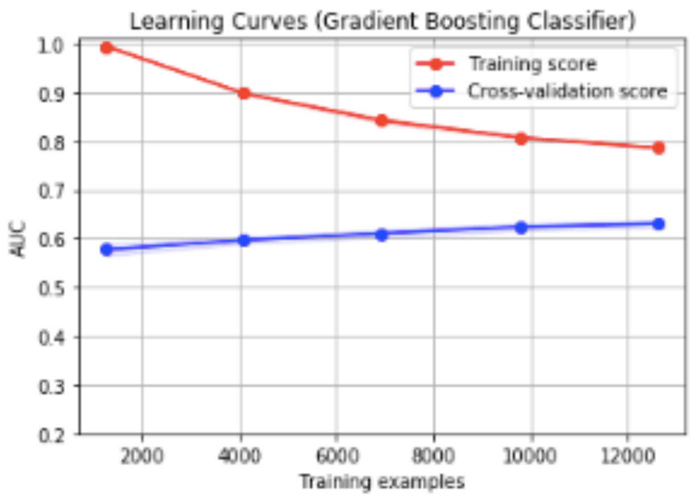
	classifier	data_set	auc	accuracy	recall	precision	specificity
0	SVC	train	0.672487	0.621527	0.527337	0.649734	0.693010
1	SVC	valid	0.667575	0.685256	0.542433	0.189116	0.682203
2	KNN	train	0.652335	0.604782	0.518331	0.626687	0.658125
3	KNN	valid	0.624185	0.650426	0.504451	0.162711	0.636577
4	LR	train	0.675219	0.623050	0.550298	0.644002	0.695801
5	LR	valid	0.666753	0.668210	0.571513	0.185728	0.680539
6	NB	train	0.623045	0.572371	0.315489	0.648839	0.829253
7	NB	valid	0.619656	0.769076	0.326409	0.192577	0.825515
8	DT	train	0.729287	0.664595	0.590131	0.693397	0.738298
9	DT	valid	0.637278	0.663647	0.539466	0.176676	0.678874
10	RF	train	0.670875	0.623811	0.586325	0.633845	0.661296
11	RF	valid	0.635109	0.617677	0.586944	0.165109	0.621595
12	GB	train	0.704446	0.694342	0.670303	0.704158	0.718381
13	GB	valid	0.669533	0.613986	0.585757	0.163384	0.617585

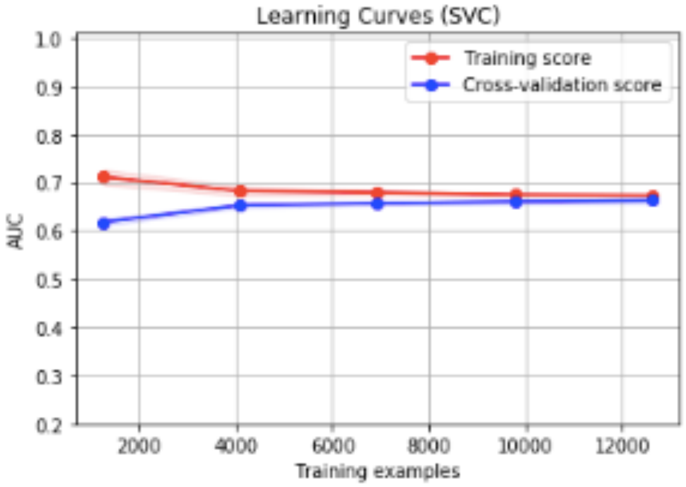


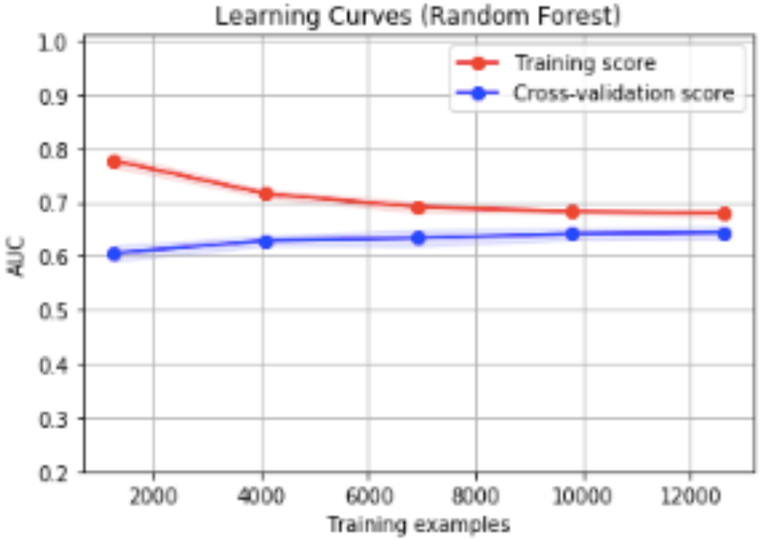
We can choose our best models based on this graph.

Learning Curve

Gradient boosting classifier had better scores than others but the learning curve shows high bias/ underfitting. Other models show less gap but poorer scores. We will use hyperparameter tuning to alleviate some of these issues.







Feature Importance

Feature Importance Random Forest

	importance
number_inpatient	0.183701
time_in_hospital	0.098817
number_emergency	0.093810
discharge_disposition_id_22	0.077494
num_medications	0.057466
num_lab_procedures	0.052790
number_diagnoses	0.045715
number_outpatient	0.028754
number_outpatient	0.023830
insulin_No	0.023613

Feature Importance from Logistic Regression

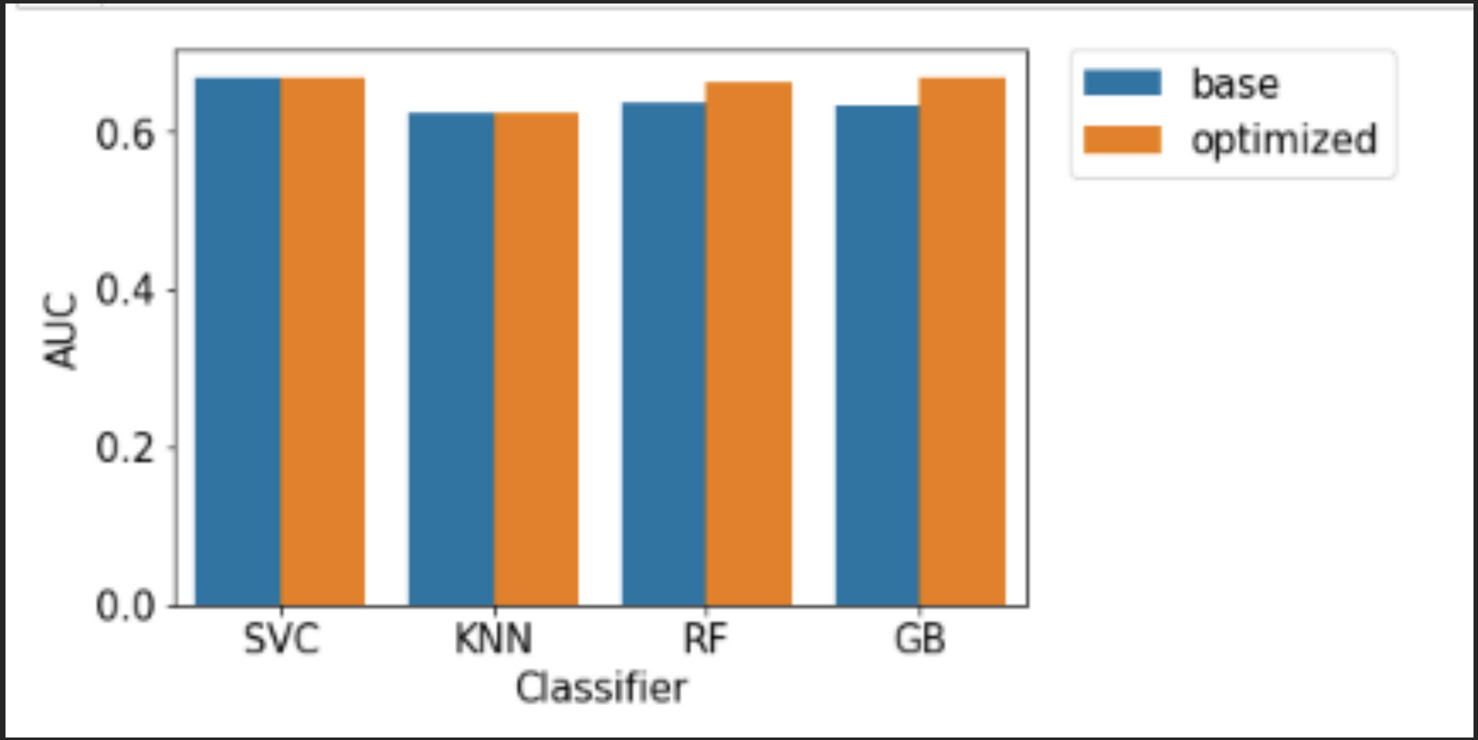
	importance
number_inpatient	0.357715
discharge_disposition_id_22	0.188475
rosiglitazone_No	0.180301
repaglinide_No	0.173352
repaglinide_Steady	0.156140
rosiglitazone_Steady	0.134689
diabetesMed_Yes	0.120780
discharge_disposition_id_3	0.119310
discharge_disposition_id_28	0.110808
discharge_disposition_id_5	0.109521

```
['discharge_disposition_id_22',
 'discharge_disposition_id_3',
 'number_diagnoses',
 'number_inpatient',
 'number_emergency',
```

Chi-Squared Top 5 features

Hyperparameter Tuning Results

	classifier	data_set	auc
0	SVC	base	0.667564
1	SVC	optimized	0.667575
2	KNN	base	0.624185
3	KNN	optimized	0.624185
4	LR	base	0.666753
5	LR	optimized	0.667887
6	RF	base	0.635109
7	RF	optimized	0.635109
8	GB	base	0.632191
9	GB	optimized	0.669533



Model for Test Set

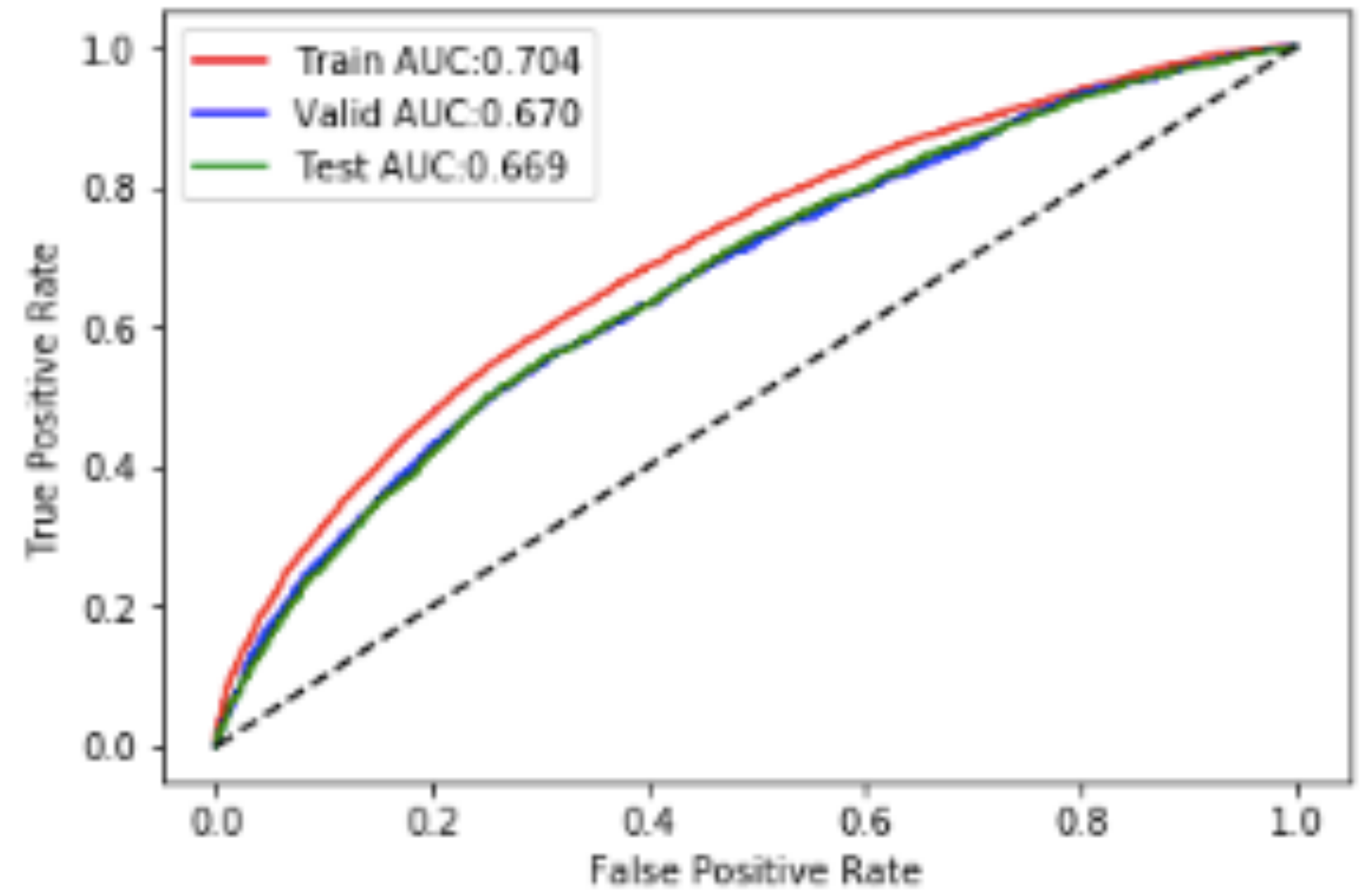
```
Training:
AUC:0.704
accuracy:0.646
recall:0.595
precision:0.662
specificity:0.696
prevalence:0.500

Validation:
AUC:0.670
accuracy:0.650
recall:0.582
precision:0.178
specificity:0.658
prevalence:0.113

Test:
AUC:0.669
accuracy:0.644
recall:0.589
precision:0.183
specificity:0.652
prevalence:0.117
```

Gradient Boosting Classifier was chosen due to it's higher scores.

ROC CURVE


Train AUC:0.704
Valid AUC:0.670
Test AUC:0.669
1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate

Conclusion

What have we learned from exploring this dataset?

1. There is a correlation between number of inpatient visits and being readmitted less than 30 days.

2. A patient being discharged to the rehab or a subacute facility has a higher chance of readmission in less than 30 days.

3. Since many patients have a primary diagnosis of heart related conditions, it is worth looking at studying readmission rates for this population. Diabetes and heart disease have a known correlation, how does this affect readmission rates?

4. If the intention is to truly predict readmission for diabetic patients, it may be helpful to look at diabetes as a primary diagnosis. According to the ICD 10, primary diagnosis requires the most serious attention and is resource intensive while secondary and tertiary diagnosis could be diseases that co-exist during admission or develop thereafter admission.

5. Additional information may be needed for this dataset. Information such as procedures and certain blood work could provide more insight into readmission.

Reference:

Dataset acquired from: https://www.kaggle.com/brandao/diabetes

ICD 10: https://www.icd10watch.com/blog/clearing-confusion-between-principal-and-primary-diagnoses

ICD 9 Codes : http://www.icd9data.com/