HOSPITAL READMISSION PATTERN DISCOVERY USING SVM

Anurag Chatterjee (A0178373U)[1], Bhujbal Vaibhav Shivaji (A0178321H)[1], Chan Yi Jie Kelvin (A0178430E)[1], Koh Lam Seng (A0179666H)[1], Lim Pier (A0178254X)[1]

ABSTRACT

Support vector machine (SVM) algorithm is well known to the machine learning community for its good practical results In this exercise, we investigate how SVM [1] performs on a clinical database of about 100,000 inpatient diabetes records. Our goal is to discover readmission patterns of hospitalized diabetic patients.

1. INTRODUCTION

The dataset was collected from the UCI machine learning repository which represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. It has also been used to determine the impact of Hemoglobin A1c (HbA1c) on hospital diabetic patients' readmission rates [2].

It is beneficial to use this data to derive new insights that may prove useful for medical research. The inherent challenges of clinical data are missing values, incomplete or inconsistent records, and high dimensionality understood not only by number of features but also their complexity.

HbA1c, which is known as a predictor for type-2 diabetes was investigated in conjunction with other variables garnered from this data like time in hospital, glucose serum test result and number of outpatient visits in the year to determine whether it resulted in readmission in less than 30 days (<30), more than 30 days (>30) or no readmission (NO) using SVM.

1.1. Data pre-processing

We first explored the data and removed original dataset columns which had a lot of missing values, for e.g. 'Weight' has 97% values missing, 'Payer code' and 'Medical Specialty' have more than 50% values missing. Then we removed null values like '?' and 'Unknown/Invalid' in other columns.

Next, we converted string values to categorical format required for SVM, for example 'Glucose serum test result' has values ">200," ">300," "normal," "none", and the target variable 'Readmitted' has values "<30" if the patient was readmitted in less than 30 days, ">30" if the patient was readmitted in more than 30 days, and "No" for no record of readmission.

Examples of variables for conversion to Categorical Codes	
Glucose serum test result: Indicates the range of the result or if the test was not taken.	>200
	>300
	normal
	none
Readmitted: Whether the patient was readmitted in less than 30 days, more than 30 days or no	<30
	>30
	No

Figure 1 Some variables for conversion to categorical codes

The following variables 'time_in_hospital', 'num_lab_procedures','num_procedures',’num_medications',' number_outpatient','number_emergency', number_inpatient', 'number_diagnoses' are not converted to categorical format to preserve their numerical properties.

Examples of Numerical Variables	
Time in hospital	Integer number of days between admission and discharge
Number of lab procedures	Number of lab tests performed during the encounter

Table 1 Some numerical features

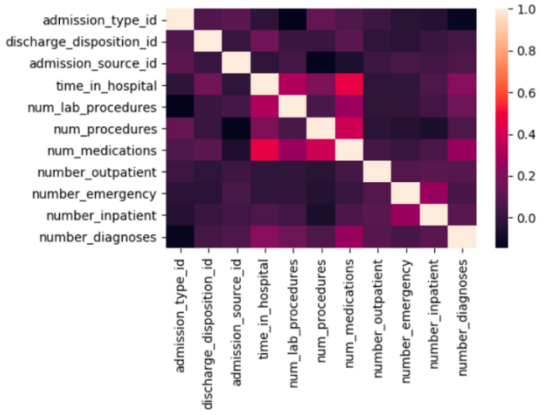

Figure 1 Correlation Matrix for the Numerical Values

[1] Graduate students, Master of Technology in Knowledge Engineering, KE-30, Institute of Systems Science, National University of Singapore

Training and test data was split 75-25 with random number seed 0 and fitted on i7 4770 CPU with 8GB memory.

1.2. Data Exploration

It was found that the target variable for the multi-class classification in the training set had the below distribution:

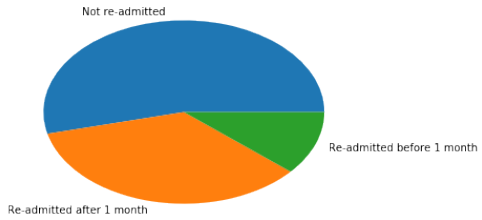

Figure 2 Distribution of target variable

There were very few patients that were re-admitted before 1 month.
Next the distribution of the patients with respect to the age groups was visualized.

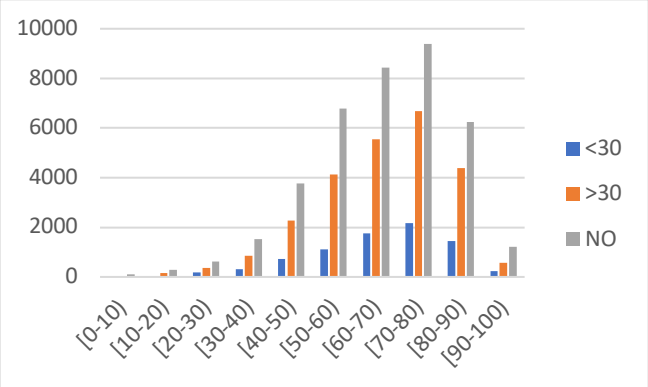

Figure 3 Distribution of number of patients with age and readmission pattern

As expected, a huge chunk of the patient population consisted of the elderly generation with a sizable number of people in the 50 to 90 age range. The decreasing trend from an age of 80 upwards would be due to lower number of individuals in the age range since the average human life expectancy in the US is approximately 80 years [4].
Also, the above graph shows that across all the age groups the number of re-admissions within a month is less than the number of re-admissions after a month which is less than no re-admissions.

2. BASELINE APPROACH

2.1. Linear SVM with Raw data

We used linear kernel with default parameters and raw input data which gave a test accuracy **0.524**.

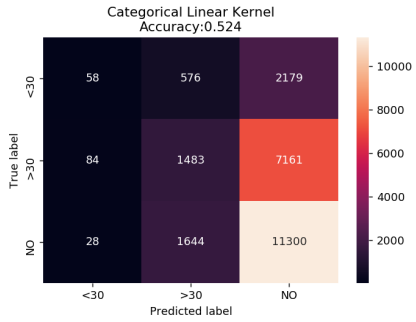

Figure 4 Default linear SVM confusion matrix

We experimented with normalized input data via min max scaling from the next section onwards.

3. PROPOSED APPROACHES

3.1. Linear kernel SVM

We performed grid search on SVM with linear kernel. Among the penalty parameters of the error term C= [0.1, 1, 10, 100, 1000], the best parameter C=10 gave a test accuracy of **0.558**, which outperformed the baseline linear SVM.

	mean_test_score	mean_train_score	param_C	params	rank_test_score
2	0.562069	0.562640	10	{'C': 10}	1
1	0.561879	0.562463	1	{'C': 1}	2
0	0.560859	0.561539	0.1	{'C': 0.1}	3

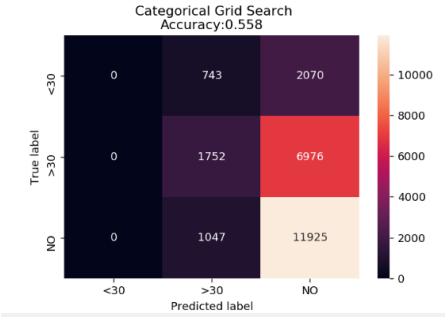

Figure 5 Linear SVM confusion matrix

Setting the dual parameter on the LinearSVM Scikit-Learn module to False to solve the primal optimization method for the SVM and re-running the SVM operation allowed us to obtain a test accuracy of **0.57**. Using the primal optimization method is recommended when the number of samples is more than the number of features [1].

We also explored setting the class_weight parameter to 'balanced' in order to use the class frequencies to adjust the weights for the different classes inversely. However, this did not result in better accuracy.

3.2. Polynomial kernel SVM

Polynomial kernel with best parameters {'C': 5, 'degree': 3} (test accuracy **0.571**) has similar accuracy compared with the enhanced linear kernel (test accuracy **0.57**)

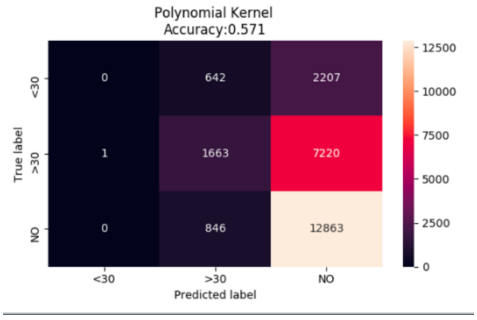

Figure 6 Polynomial kernel SVM confusion matrix

3.3. Sigmoid kernel SVM

Sigmoid kernel with best parameters {'C': 1000, 'gamma': 0.001} (test accuracy 0.557) gave similar accuracy compared with linear kernel (test accuracy 0.558)

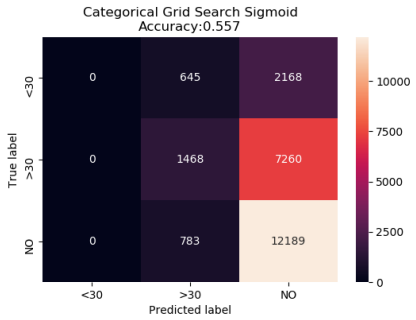

Figure 7 Sigmoid kernel SVM confusion matrix

3.4. RBF kernel SVM

We experienced issues training Radial Basis Function (RBF) kernel SVM, possibly due to the huge number of training records [3].

3.5. One Hot Encoding of some categorical features

We explored one hot encoding as an alternative data processing method for categorical values. For linear kernel, the test accuracy improved slightly from 0.558 to **0.576**.

We observed that one hot encoded input took approximately 50% longer to train (from 4 min 24s to 6min 21s). The increased training time became more apparent when we used non-linear kernels.

	mean_test_score	mean_train_score	param_C	params	rank_test_score
0	0.576633	0.598574	0.1	{'C': 0.1}	1
1	0.573913	0.602966	1	{'C': 1}	2
2	0.559499	0.585397	10	{'C': 10}	3

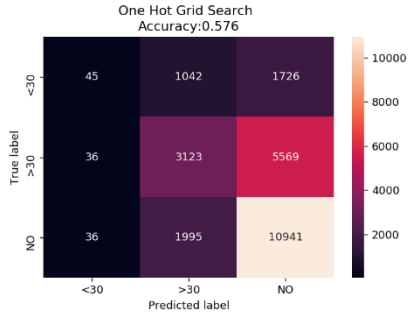

Figure 8 One Hot Encoded Linear SVM confusion matrix

4. ALTERNATIVE APPROACHES

4.1. Modelling by age-groups

Health patterns vary across age groups as seen during the data exploration. We split the data into various age groups to create models for each of these groups and observe better results.

4.1.1. Age 0-30

The dataset has 2509 samples for this age group which is around 2.5% of the data.

Even with this small subset, the initial grid search with all parameters above parallelized on 4 cores did not complete after 3 hours, possibly due to the number of combinations. For example, polynomial kernel alone had 5x7x4 = 140 combinations of the hyper-parameters.

We dropped gamma parameters for sigmoid and polynomial kernels to improve run time for this and subsequent age groups. The best parameters {'C': 1000, 'gamma': 0.01, 'kernel': rbf} gave test accuracy of **0.630**:

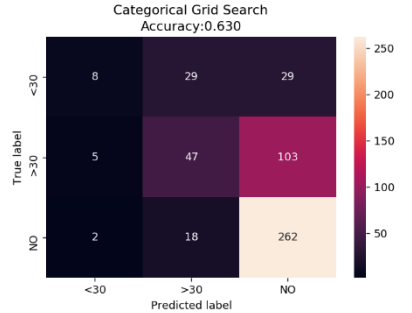

Figure 9 Age 0-30 model confusion matrix

4.1.2. Age 30-50

Linear kernels gave the best results for this age group.

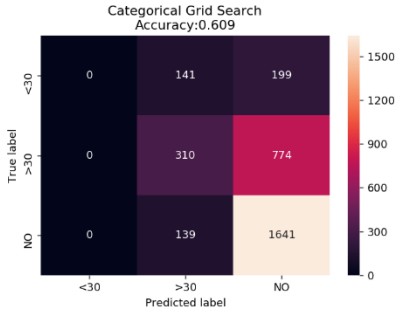

Figure 10 Age 30-50 confusion matrix

4.1.3. Age 50-70

RBF kernels gave the best results for this age group.

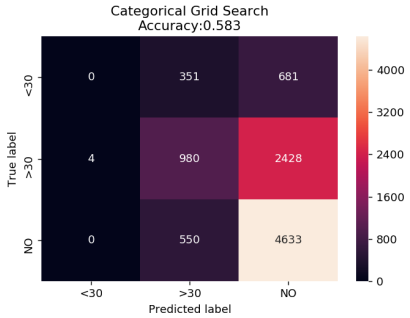

Figure 11 Age 50-70 confusion matrix

4.1.4. Age 70-100

Grid search with best parameters {'C': 100, 'gamma': 0.05, 'kernel': 'rbf'} gave test accuracy 0.544

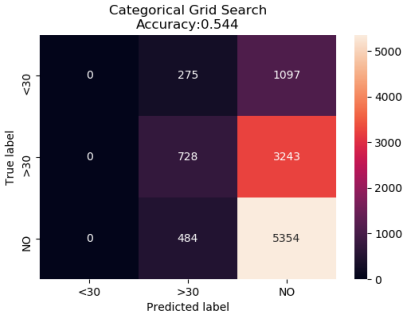

Figure 12 Age 70-100 confusion matrix

4.2. Binarization of target

We found from previous experiments that SVM performed relatively poor for Multi-class classification. So, we decided to binarize the target variable. We categorized target variable as the patients who got readmitted before and after 30 days which comprises of the categories '<30' and '>30' and the patients who never got re-admitted which comprises of category 'NO'. We applied LinearSVM for this. The result analysis was as follows:

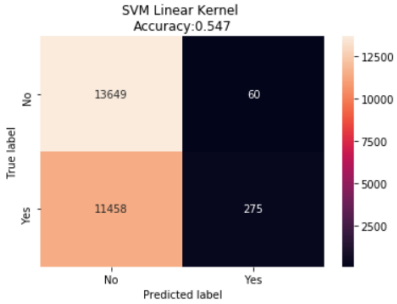

Figure 13 Confusion matrix for binary classification

Furthermore, we took out the numerical columns along with age column which is categorized into 3 categories and did linear SVM for the same. We saw that accuracy got increased for the same. The result analysis can be given as follows:

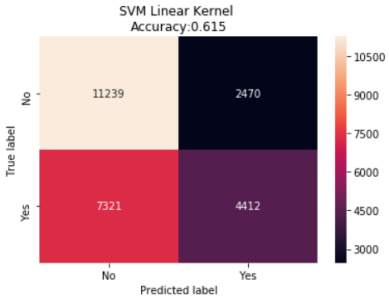

Figure 14 Confusion matrix for binary classification with feature selection

Here 'Yes' corresponds to people who got readmitted again while 'No' corresponds to people who did not get readmitted.
The below image shows the relative importance of the various features for predicting the target as obtained using [5].

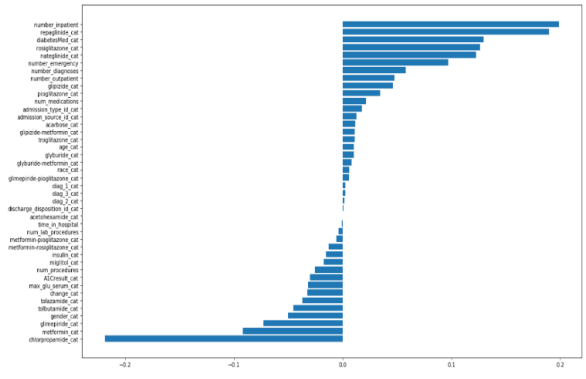

Figure 15 Feature importance according to SVM coefficients

5. EXPERIMENTAL RESULTS

We get an average accuracy of approximately **0.57** using the SVM classifiers, which is substantially better than the random probability of 33% which one would expect from a three-class classification problem. However, we see that the number of instances that are correctly predicted for the "<30" class is consistently less. Technically this can be attributed to the imbalance in the data where there is much fewer records of this class as could be seen during the data exploration. Also, one would expect readmissions within 30 days as emergency cases without any fixed patterns and so the features could not be learnt by the SVM algorithms.

6. CONCLUSION

SVM is primarily a binary classifier. Hence, when running a multi-class classification problem, Scikit-Learn's SVC function defaults to using a one-vs-one scheme. This means that one trains K(K-1) / 2 binary classifiers for a K-way multi-class problem. Each binary classifier receives the samples of a pair of classes from the original training set, and must learn to distinguish between these two classes. At prediction time, a voting scheme is applied [5]. The issue with this is that the fit time complexity is more than quadratic with the number of samples, making it hard to scale above couple of 10,000 samples.

As a result, it is not very practical compute time-wise to train multi-class classification problems using non-linear SVM models for large datasets. Nevertheless, in the spirit of learning, we did carry out the tests on our large dataset. Our dataset comprised of about 100,000 samples, resulting in long training times.

REFERENCES

[1] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 1w2, pp. 2825-2830, 2011.

[2] Beata Strack, Jonathan P. DeShazo, Chris Gennings, et al., "Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records," BioMed Research International, vol. 2014, Article ID 781670, 11 pages, 2014. https://doi.org/10.1155/2014/781670.

[3] Hsu, Chih-wei & Chang, Chih-chung & Lin, Chih-Jen. (2003). A Practical Guide to Support Vector Classification Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin.

[4] WHO, "Wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/List_of_countries_by_life_expectancy. [Accessed May 2018].

[5] "sklearn.svm.LinearSVC," [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html.