Final Project Report

Yu Lu, Zhongyu Wang,Yuting Wei
Peking University

June 24, 2013

Abstract

In this report we study the classification of Percutaneous coronary intervention (PCI). The main difficulty of this problem is the large amount of missing data. We impute the missing value by K-Nearest-Neighbour(KNN) method, which focuses on the local structure of distribution of all variables.

Then we try 4 different methods in classification , which are best subset selection logistic regression, elastic net logistic regression, K-Nearest Neighbour and Support vector machine. On Anzhen data set, the best subset selection logistic regression performs best and it achieves 92.6% prediction accuracy. As for Chaoyang301 data set and the whole data set, KNN is the best one and it achieves 88.4% and 87.9% prediction accuracy separately.

Finally, we point out that variable ”PCI术中钙拮抗剂” alone can achieve 89.4% prediction accuracy on Anzhen data while variable ”随机血糖” has 86.4% prediction accuracy on Chaoyang301 data. Therefore, those two variables are most important. By considering the prediction accuracy and the interpretation of the model, we conclude that one variable linear model is the best for this problem.

Key word: classification, data imputation, K-Nearest-Neighbour, PCI

1 Problem Description

Percutaneous coronary intervention (PCI) is a non-surgical procedure used to treat the stenotic (narrowed) coronary arteries of the heart found in coronary heart disease. These stenotic segments are due to the build up of the cholesterol-laden plaques that form due to atherosclerosis. If no-reflow happens after PCI, the treatment is useless to patients. As a result Judging whether no-flow will happen after PCI by the medical data that hospitals are able to get from patients is a valuable work. We proposed several algorithms for the problem. In this project, we mainly deal with dataset got from Anzhen Hospital and 301 Hospital in Beijing.

The dataset have several charateristics that we should lay stress on.

First, the datasets have many missing values, which is often the case of medical data. Although every efforts should be made towards avoiding sparse data collection of medical data, almost always there are some missing data which must be handled carefully. Missing data can't be used on some datasets and it might affect the prediction accuracy. So it is reasonable to first impute the missing values or take other measures.

Another problem worth noticing is that the two hospitals tends to collect different medical of patients. This can be seen in figure1. As a result, the algorithm my choose different features and get different accuracy in the two datasets.

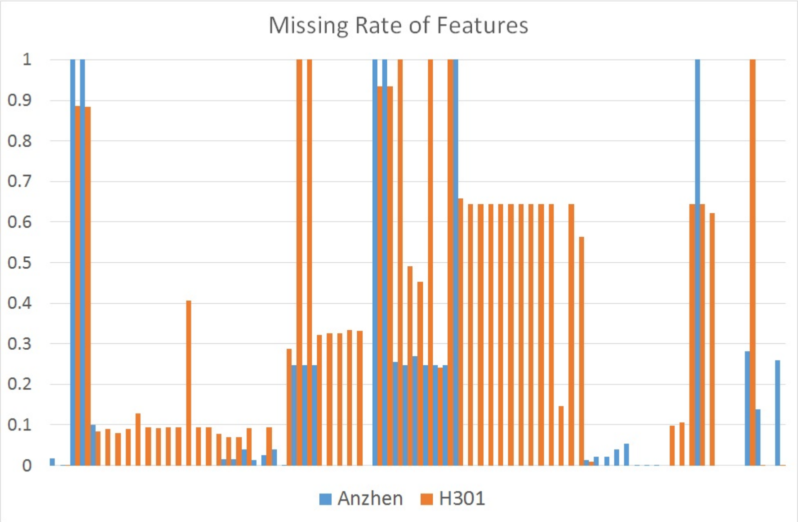

Figure 1: missing rates in different datasets

2 Data Augmentation

As is shown in figure 1, there is a large proportion of missing data. In order to make statistical inference, we should impute the missing values first. The best way to impute those missing values is by the joint distribution of p features . Unfortunately, there is no appropriate common parametric distribution to describe this data set, so we have to fit a nonparametric distribution. In fact, in order to impute a specific missing value, we only need to fit the distribution locally. Therefore, we use **K-Nearest-Neighbour Imputation** method.

Nearest Neighbour Imputation is a frequently used method in dealing with missing values. But it works only if at least one feature of all data are not missing. This assumption are not true in our problem.

A remedy is first using mean imputation or median imputation to pre-fill all missing values. But this method might be problematic. All missing value of a feature are filled with the same value and their distribution to the distance of those examples are zero. This might lead to a underestimation of the distance between examples with the same missing feature.

In our method, we overcome this drawback by defining distance between two samples. The definition is:

$$D(x, y) = sqrt(\Sigma_{i=1}^{p} d(x_i - y_i))$$

d(x,y)is defined as:

- If both x_i and y_i are not missing, $d(x_i - y_i) = (x_i - y_i)^2$
- if both x_i and y_i are missing $d(x_i - y_i) = E(x_i - y_i)^2 = Var(feature\ i) * 2$
- If only y_i is missing, $d(x_i - y_i) = E(x_i - y_i|x_i)^2 = Var(feature\ i) + (x_i - mean(feature\ i))^2$

The definition is derived by assuming all features are independent and all examples are identically distributed. the mean and variance are replaced by its estimation on given dataset. Besides, in order to make p futures distance addable, the dataset is normalized to zero mean and unit variance. Also, we shouldn't expect a feature be predictive if we have to impute too many missing values there, so we delete features whose missing rate are larger than 50%.

A drawback of this definition of distance is it doesn't treat discrete and continuous variable separately. Since all the discrete variables in this problem preserve order or binary variables, it is meaningful to define distance between them.

The procedure of k-nearest neighbour imputation is in Algorithm 1.

We should note that then k-nearest neighbour imputation is actually convergence. As is shown in figure 2, we apply k Nearest Neighbour Imputation iteratively, the data set at last convergence in approximate 60 steps.

Algorithm 1 k Nearest Neighbour Imputation

```
procedure kNN IMPUTATION
    for all $i \in features$ do
        for all $j \in examples$ do
            if $X_{ji}$ is missing in original data then
                $X_j^1, X_j^2 ... X_j^k$ ←k examples nearest with $X_j$ whose feature j is not missing,
                $X_{ji} \leftarrow \frac{\sum_{i=1}^{k} X_j^i}{k}$
            end if
        end for
    end for
```

3 Model fitting and prediction

After augmentation, we try 4 different methods to fitting data and make predictions. They are 2 linear methods Logistic Regression(elastic net) and Logistic Regression(Best subset selection), and two non-linear methods, Nearest Neighbour and Support Vector Machine(radial kernel).

We use 5-fold cross validation to determine tuning parameter in different methods. And we compare those methods according to their 5-fold cross-validation prediction accuracy. Besides, we use those methods on all 3 data sets: Anzhen Hospital's PCI data, 301 Hospital's PCI data and their combination.

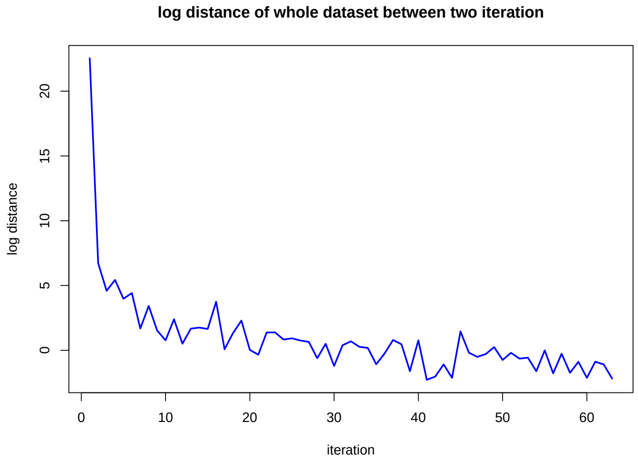

Figure 2: The convergence of k Nearest Neighbour Imputation

3.1 Logistic Regression(Best subset selection)

It is common to use logistic model in dealing with binary response. In this problem, p=69¡¡n, so the traditional regression model works out. In order to achieve best prediction accuracy, we use best subset selection to fit the data. We first separate data set into five parts, utilize four parts to do best-subset selection to do variable selections(the forward method is applied). Then we use the rest part and selected-variables to run a logistic regression model and make predictions. The best result in 3 datasets is respectively 92.6%, 85.3%, 85.8% when numbers of features are respectively 11, 4, 17.

3.2 Logistic Regression(elastic net)

In order to select variables automatically, penalized methods can be used to fit the logistic regression model. Here we use the elastic net(a combination of Lasso and Ridge regression) to fit the model. By using R package glmnet, this methods can be used easily and results are 90.2%,80.0%,84.6% in three data sets and the corresponding average number of variables selected in cross-validation are 13, 17 and 14.

3.3 Nearest Neighbour Method.

Nearest Neighbour method can also been used in classification, as it captures local distribution of the design matrix. If two samples are near in the sense of distance defined in section 2, the response variable should also be the same. However, not all p variables are related to the response variable, so we should do variable selection first to find useful feature.

Marginalized logistic regression is done on each feature. Then we pick k features with largest β value got in marginalized logistic regression and weighted by their exact β value.The top 20 features ranked by β value in 3 datasets are in table3: It can be seen that the most distinctive features in 3 datasets are quite different.

In classification period, k nearest neighbour method is used. the number of neighbors are the same as it in KNN imputation period. The algorithm is tested on 3 different datasets:. We tested different number of chosen features and number of neighbors. And the best prediction accuracy we get is 91.1%, 88.4% and 87.9% respectively, while the corresponding number of features select is

40, 1 and 7.

3.4 Support Vector Machine(radial kernel)

Support Vector Machine(SVM) is a classical method used in classification. It can be linear or non-linear by using different kernels. In this problem we find the radial kernel perform best. SVM also involves variable selection and perhaps the best way is also using best subset selection. However, it is very time consuming so that we cannot finish it on our computer. Here we only use best subset selection up to 4 variables.

For Anzhen dataset, the variable we finally select are "随机血糖"、"罪犯血管血栓数量"、"PCI术中2b3a"、"PCI术中钙拮抗剂", and we can achieve 89.6% prediction accuracy. For Chaoyang301 dataset, the only variable we select is "随机血糖",and the prediction accuracy is 86.8%. For all dataset, we finally select the variable the same in Anzhen dataset, and the corresponding prediction accuracy is 87.6%.

Dataset	Anzhen	Chaoyang 301	All
Logistic(Best subset)	0.926(11)	0.853(4)	0.858(7)
Logistic(Elastic Net)	0.902(13)	0.800(17)	0.846(14)
Nearest Neighbour	0.911(40)	0.884(1)	0.879(7)
SVM(Radial Kernel)	0.876(4)	0.868(1)	0.876(4)

Table 1: Best prediction accuracy of 4 methods on 3 data sets.

4 Result Analysis

We conclude the results in table 1. The data in bracket is the number of variables select. For example, 0.926(11) means we the best subset select 11 variables to achieve maximum prediction accuracy on Anzhen data. As the standard deviation of all those methods in prediction is quite small(< 0.05), so we use number of variables selected rather than standard deviation to compare different methods.

The best subset logistic regression achieve best prediction accuracy on Anzhen data set, while nearest neighour method perform best on Chaoyang301 and the All data set. In this problem, linear methods do not perform worse than non-linear method, all those 4 method's prediction accuracy are close. As for the interpretation of the model, SVM is the best as it use least variables in the model. As for computation, Elastic Net Logistic is the best, as it is the fastest method.

It is interesting to note that one variable is enough for among 2 data sets separately by using naive classification. As for the Chaoyang301 data set, the variable "随机血糖" alone can achieve 86.4% prediction accuracy. The method is very simply. For those "随机血糖(after normalization)> 0.12", let it Y=1, and otherwise Y=0. But this variable is not effective in the Anzhen data set. On the other hand, the binary variable "PCI术中钙拮抗剂" can achieve 89.4% prediction accuracy on Anzhen data set by predicting Y equals to the indicator of “PCI术中钙拮抗剂”. So we may conclude that concerning the prediction accuracy, the computation and the interpretation of the model, best method of this problem is actually the simplest method by using one variable.

The reason why “PCI术中钙拮抗剂” fails to achieve such high prediction accuracy on the Chaoyang301 data set is its value are almost all missing. A nature idea is that if we can impute missing “PCI术中钙拮抗剂” value in Chaoyang301 data set, perhaps we can make predictions on whole data set easily. However, the correlation between“PCI术中钙拮抗剂” and other variables is low, so we fail to make good imputation of missing “PCI术中钙拮抗剂” value. The reason why ”随机血糖” does not preform well on the Anzhen data set remains unclear, and we think it deserves further study.

附录

A The result of Best subset selection

number of variables	An Zhen	Chao Yang301	Together
1	0.9012	0.8309	0.8311
2	0.9012	0.8235	0.8272
3	0.9012	0.8382	0.8175
4	0.8971	**0.8529** ‡	0.8272
5	0.8971	0.8382	0.8252
6	0.9012	0.8382	0.8369
7	0.9053	0.8419	0.8369
8	0.9136	0.8529	0.8427
9	0.9218	0.8529	0.8388
10	0.9218	0.8529	0.8408
11	**0.9259** †	0.8419	0.8388
12	0.9259	0.8456	0.8447
13	0.9259	0.8419	0.8447
14	0.9259	0.8419	0.8427
15	0.9218	0.8419	0.8447
16	0.9177	0.8419	0.8466
17	0.9177	0.8529	**0.8583** §
18	0.9136	0.8529	0.8563
19	0.9177	0.8529	0.8524
20	0.9012	0.8529	0.8485

†: 病变支数、甘油三酯、症状到PCI时间、PCI前ARB、术前TIMI血流 、罪犯血管狭窄程度、罪犯血管、肌酐、killip分级、舒张压、既往CA拮抗剂

‡: 病变支数、甘油三酯、症状到PCI时间、入院诊断

§: 病变支数、入院诊断、PCI前ARB、术前TIMI血流、罪犯血管血栓数量、PCI史、吸烟史、PCI术中硝酸、既往利尿剂、性别、预扩张、既往ACEI、PCI前他汀、PCI前ADP拮抗剂、HDLC、侧支循环分级、身高

Table 2: prediction results using different data set

B The result of K-Nearest-Neighour

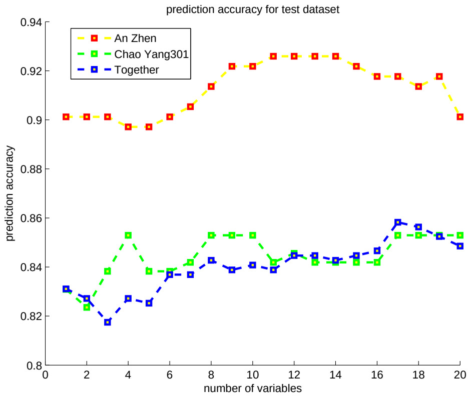

Figure 3: Testing results with different neighbor numbers

number of variables	All	Anzhen	Chaoyang 301
1	PCI前ARB	HDLC	随机血糖
2	HDLC	PCI前ARB	罪犯血管狭窄程度
3	PCI术中钙拮抗剂	PCI术中钙拮抗剂	罪犯血管血栓数量
4	罪犯血管血栓数量	白蛋白	术前TIMI血流
5	随机血糖	PCI术中0b3a	killip分级
6	PCI术中0b3a	罪犯血管血栓数量	年龄
7	IABP	PCI术中血栓抽吸	收缩压
8	术前TIMI血流	PCI术中硝酸酯	PCI前CK
9	PCI术中血栓抽吸	PCI前他汀	入院诊断
10	killip分级	IABP	舒张压
11	PCI术中硝酸酯	随机血糖	中性粒细胞
12	入院诊断	术前TIMI血流	后扩张
13	LDLC	PCI前低分子肝素	吸烟史
14	年龄	入院诊断	支架数量
15	PCI前CK	侧枝循环分级	脑梗塞史
16	侧枝循环分级	症状到PCI时间	甘油三酯
17	中性粒细胞	LDLC	侧枝循环分级
18	收缩压	PCI前ACEI	最大扩张压力
19	PCI前他汀	心率	PCI术中硝酸酯
20	PCI前2b3a拮抗剂	PCI前 β 阻滞剂	性别

Table 3: Top 20 Features Ranked by Beta Value.

N	knum=20			knum=10			knum=40		
	Anzhen	H301	All	Anzhen	H301	All	Anzhen	H301	All
1	0.8031	0.8838	0.7948	0.7784	0.8742	0.7948	0.8031	0.8786	0.7948
2	0.8031	0.8757	0.7948	0.7784	0.8757	0.7777	0.8031	0.8683	0.7948
3	0.8912	0.8713	0.8337	0.8764	0.8639	0.824	0.8912	0.8647	0.8317
4	0.8912	0.8661	0.829	0.8904	0.8573	0.8232	0.8912	0.8669	0.8282
5	0.8813	0.8676	0.869	0.8542	0.8654	0.8679	0.8887	0.8544	0.8628
6	0.8912	0.847	0.8655	0.8962	0.8558	0.8605	0.9036	0.8441	0.8632
7	0.8747	0.847	0.8741	0.8986	0.8455	0.8791	0.9011	0.8426	0.8601
8	0.897	0.8492	0.8737	0.8904	0.8367	0.8733	0.9036	0.8338	0.8558
9	0.8986	0.8492	0.8581	0.8896	0.8367	0.8636	0.9036	0.836	0.8589
10	0.8995	0.8382	0.8519	0.8929	0.8419	0.8675	0.9028	0.8242	0.8473
15	0.9011	0.8264	0.8512	0.8962	0.8272	0.8558	0.9052	0.8125	0.8488
20	0.9077	0.8323	0.8496	0.8945	0.8272	0.8457	0.9036	0.8169	0.8492
30	0.9085	0.836	0.8512	0.9003	0.8272	0.8496	0.906	0.81985	0.8477
40	0.911	0.8382	0.8523	0.9011	0.8286	0.8473	0.9044	0.8227	0.8465
50	0.9093	-	0.8535	0.9052	-	0.8504	0.9069		0.8457

Table 4: Test Results
N is the number of chosen features.

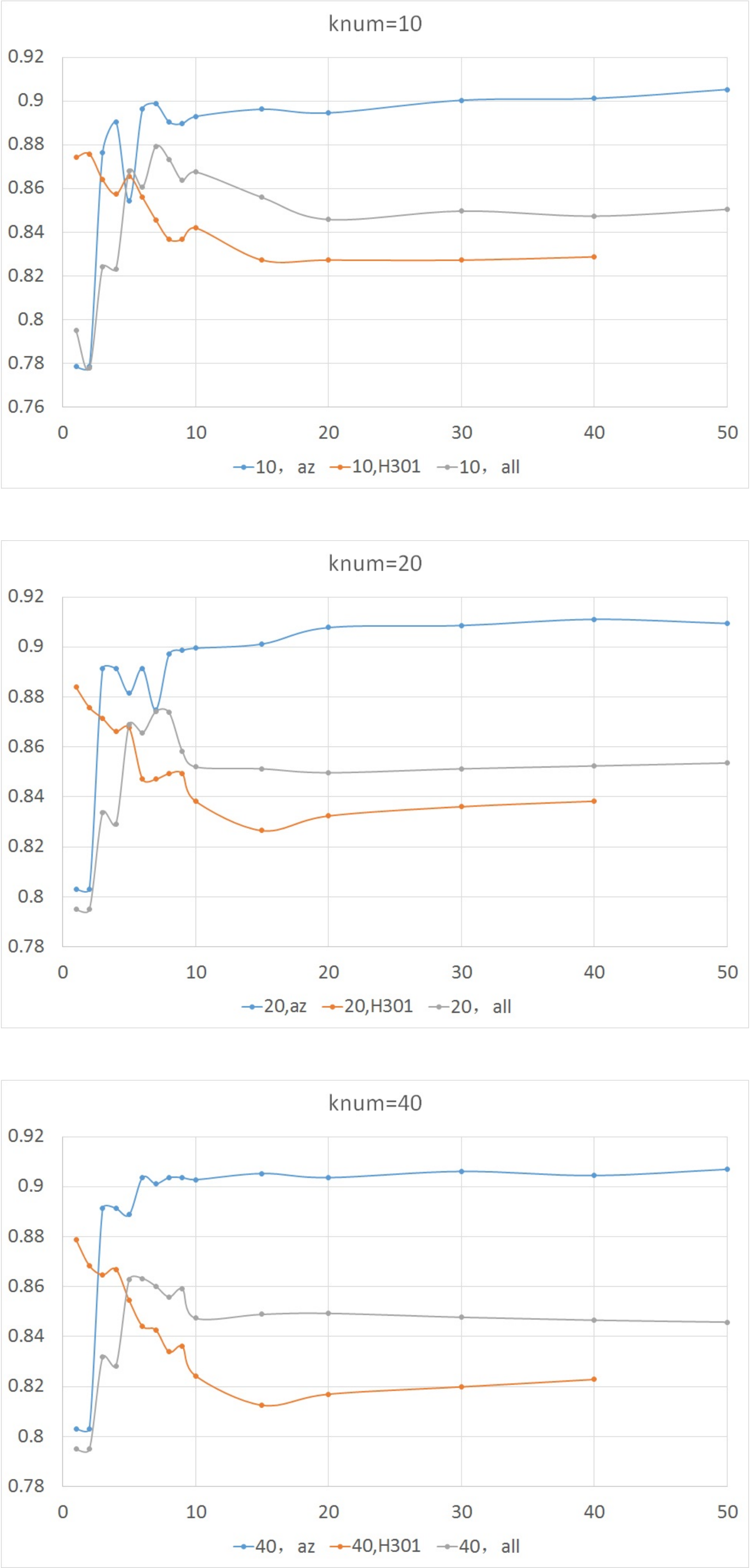

Figure 4: Testing results with different neighbor numbers