**Analysis of comparing different statistical learning models in predicting insurance claims**

1. Overview of the analysis

The goal of the analysis is to predict whether there are insurance claims associated with a policyholder according to data collected. The data of this analysis is the "Sample Insurance Claim Prediction Dataset" which based on “Medical Cost Personal Datasets” downloaded from Kaggle. The creditability of the data set is rather high as it is retrieved from the book Machine Learning with R by Brett Lantz. The source of the data can be found here: <https://www.kaggle.com/easonlai/sample-insurance-claim-prediction-dataset>. For this analysis, I preform the whole process using R language. The goals of the analysis are

1. Compare the classification process of different statistical learning models when applying to the dataset
2. Try to pursue for a higher prediction accuracy without overfitting
3. Elementary analysis of data

There are 1338 set of data available. For each set of data, there exist 8 distinct features plus one prediction variable which is the ‘insuranceclaim’. These variables have explicit descriptions of the policyholder respectively:

1) age : age of policyholder

2) sex: gender of policy holder (female=0, male=1)

3) bmi: body mass index, providing an understanding of body, weights that are relatively high or low relative to height, objective index of body weight (kg / m ^ 2) using the ratio of height to weight, ideally 18.5 to 25

4) steps: average walking steps per day of policyholder

5) children: number of children / dependents of policyholder

6) smoker: smoking state of policyholder (non-smoke=0;smoker=1)

7) region: the residential area of policyholder in the US (northeast=0, northwest=1, southeast=2, southwest=3)

8) charges: individual medical costs billed by health insurance

9) insuranceclaim: yes=1, no=0, the prediction variable

It can be see that all these features are common in everyday life and the collection of these data shouldn’t be very complicated. Therefore, I have a reason to believe that the accuracy for these data collected is relatively high in general, especially for age, sex, children, smoker and region. For bmi, steps, and charges, although the record of the data is not perfectly accurate, these variables are either calculated through certain formulas or measured in certain predefined method, hence I need a more flexible model to allow for relatively higher variance than the other variables. I will examine the accuracy of these data through data pre-process in later on section.

The dataset is small in general and the data dimension isn’t very high, and the prediction outcome is binary as well. Thus when considering about the models, I primarily focus on simple supervised models in this analysis.

1. Data pre-processing:

In this section, I mainly apply four steps:

*Data cleaning*

From the summary of the data, I observe no missing values. From the boxplot of the data, I observe no special pattern except a gargantuan number of outliers for the “charges” identified by the boxplot. I therefore further plot a histogram for double check if there is any noisy data included. From the graphs I can see that only charges that are approximately over 3000 is classified as outliers by boxplot, but this is in fact reasonable when I consider the meaning of the charge amounts. Therefore, I don’t further do any process to clean the data.



*Data integration*

For this dataset, I only have one excel file which contains data, so there is no need for this step.

*Data transformation*

I apply the normalization to all the features to scale the value of the features into standard scores. After this, I plot the histogram of each of the variables.



*Data reduction*

In terms of reduction of the data, several technics have been applied to examine the relationship within the features. First I applied the subset selection in the library *leaps*. This method fit a separate least squares regression for combination of features. Depending on the algorithm, there are multiple choice of the best subset. As there are only 1338 number of data, using forward or backward method of subset reduction will lose certain accuracy, therefore, I adopt the exhaustive subset selection for the data. The primary analysis of the subset shows that ‘children’ is the most significant feature and ‘sex’ and ‘region’ might not have a great impact on the prediction.

I further use a 10-fold cross validation to examine which subset is the best for prediction purpose. The results are showed as follows:



The subset selection method shows that feature ‘sex’ is better not included. However, this might have the potential downside effect brought by correlation among variables. Therefore, the principle component analysis was applied to review the data from another perspective. I first plot the correlation graph for the 8 features.



As shown, except that bmI and steps have an obvious negative correlation, smoke and charges share an obvious positive correlation, the other data are not or weakly mutually correlated. I further adopt principle component analysis(PCA) to the eight features.



The graph of principle component analysis pve shows a descending trend for separate principle components and an increasing trend for cumulative principle components. However, there doesn’t exist an obvious ‘elbow’ for the individual pve graph as usual, only a sharp decrease for the 4th and 7th principle components. Considering the cumulative effects brought by all of these principle components, I eventually decided to use all 8 components.

As the above two methods indicate separate methods, I record both datasets and evaluate them for all of the models that I are about to test. The original data set has also been saved. In later on models, different data reduction method is applied to different models according to the characteristics of the models.

1. Model setup

In this analysis, I evaluated the performance of the following models: K nearest-neighbor, Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Partial Least Square, Principle Component Regression, Ridge, Lasso, Support Vector Machine and shallow neural network.

I first set up a comparing matrix to store the performance of different models. In this analysis, I mainly focus on the RSS and accuracy two perspectives. For some models, I include AIC, BIC, R square and other related indicators to evaluate the models.

The flow of the analysis starts from evaluating each of the models separately, followed by a cross validation of the performances of all the models.

* 1. K nearest-neighbor(kNN)

As a non-parametric method performed in this analysis, the kNN method has an advantage of disregarding the assumptions of the dataset. However, it might take a long time to process for its non-parametric nature. As the dataset is relatively a small one, it is reasonable to set this as the very first approach.

*Data*

To examine the effect of the curse of dimentionality, I reserve the original data, data after subset selection, data after PCA for comparison. In addition, the training/test split is chosen to be 0.75.

*Method*

The I first use 10-fold cross-validation to select the value for k. As there are 1338 data to be categorized into 2 categories, I first set a boundary to let k be 1 to 100. I accord the training set and test set data and calculate the prediction accuracy for comparison. The results is shown as follow:



It can be shown that the training accuracy is descending. The test set accuracy fluctuates within the 1~20 region, then it goes down graudually. From this, I have reasons to believe that the best k lies in the 1~20 region. Thus, I further narrows the cross-validation region for k to be in 1~20. Moreover, I applied 30-fold to make the results more convincing.



The data shows that when k = 1, the accuracy of training set is basically 1, while the accuracy of test set is around 0.9. This indicates a high variance situation. On the other hand, when k>1, the training set accuracy starts to go below the test set accuracy, especially when k>4, the difference between training set and test set accuracy is more than 5%. This demonstrates high bias of the model. I therefore consider k = 3 is the best parameter setting, as under this situation, the two lines are quite close while remain at a relatively high level.

After this, same analysis was applied to the other two datasets (after subset reduction and PCA), except that the region for k narrows to 1~20 directly. The results are presented as follows:



From these 2 graphs, it is clear that the result of data after PCA shows no significant improve compared to the original data. On the other hand, the data after subset selection obviously outperforms the other 2. For all three datasets, k =3 is always the best k for selection.

The final step is to cross-validate the RSS and accuracy of the data. As kNN algorithm in R doesn’t consume a lot of time, I measure the mean and variance of RSS and accuracy using a 30-fold cross validation.

A close up of a sign

Description automatically generated

*Conclusion*

From this section, I set up a benchmark of how well the parametric method should perform. It appears that kNN works not bad towards the dataset I have.

* 1. Logistic Regression

Then, I applied a logistic regression using the glm function. This is the most widely used method in the industry and give very intuitive interpretation. However, the logistic regression may tend to have a higher bias due to its simple nature

*Data*

Same as the kNN approach, I use all three of the datasets I obtain to compare results. Again the training/test split is chosen to be 0.75

*Method*

I again cross validate the bias and variance information given the glm function.

As glm is a relatively fast method, I choose k = 50 fold to make sure that the result is less deviant. I store the cross validation result in 3 matrices corresponding to the 3 datasets. The results are as follows

A screenshot of text

Description automatically generated

I observe a minor difference among the three results, and within each row, the test and training error difference doesn’t seem to be large. Combined with the results I yield in the previous kNN method, it does seem that the logistic regression doesn’t result in a very high bias or variance situation. Moreover, all three outcomes have a standard deviation of around 1%, which is reasonably acceptable. Therefore, it might be trivial to select from one of these three results, but since data after subset selection takes the less memory space and get a tiny better result, I choose it to be the data for logistic regression.

After the above steps, I calculate the 30-fold mean and variance of RSS and accuracy.

A close up of a sign

Description automatically generated

*Conclusion*

Compare to the kNN result, the stability of pure Logistic regression is higher, whereas the accuracy level appears to be lower.

* 1. Linear Discriminant Analysis (LDA) & Quadratic Discriminant Analysis (QDA)

Next I examine the performance of LDA and QDA. Compared to the logistic regression, the LDA and QDA are more used when dealing with more than 2 classes. These methods also outperforms logistic regression when the separation of the classes are very obvious. When it comes to the difference between LDA and QDA, although they both assume the features are drawn from a multivariate Gaussian distribution, QDA also assumes that the covariance matrix be different while LDA assumes the same. Therefore, QDA appears to be more flexible while LDA is more stable with its results.

*Data*

Again, I utilized the 3 datasets I obtained since LDA and QDA both don’t contain data reduction and feature selection method.

*Method*

Just as the previous two methods, I examine the 30-fold cross validation results of the three datasets. The results are shown as follows:

A blue and white text

Description automatically generated

It can be seen that LDA outperforms QDA in both bias and variance in any of the circumstances (LDA 12% vs QDA 17% for mean RSS). It appears that the LDA might do a better job in classifying this dataset. Comparing the datasets, again the outcomes are very similar, indicating that data reduction technic isn’t very necessary to this data. I choose data after subset selection as it contains fewer dimensions.

A picture containing bottle, text, book

Description automatically generated

From the graph I can see the result of the accuracy level and the RSS of the chosen model.

*Conclusion*

As the results yield from the logistic regression is quite stable, it is expected that the separation of the two classes are not very distant, thus the LDA and QDA method doesn’t outperforms the logistic regression.

* 1. Partial Least Square(PLS) & Principle Component Regression(PCR)

These two methods are closely connected to the PCA method mention above. Specifically, the PLS algorithm emphasize on data reduction subject to the prediction context, while principle component regression use a number of principle components for regression analysis. Both of the methods can be served as a feature selection tool as well as classifier.

*Data*

As principle component analysis can be only used once, otherwise producing the same principle components, I don’t consider the usage of dataset after PCA for this model. Therefore, I only compare the original dataset and the data set after subset selection.

*Method*

I first examine the PLS method. For both of the datasets, I first summarize the PLS analysis and plot the mean square error of prediction.

A blue and white sign

Description automatically generated A close up of a sign

Description automatically generated



It appears that for the original dataset, using 7 or 8 component is the best choice. For the dataset after subset selection, using 6 or 7 components is the optimal option. Since the dataset is relatively small, we can use all the components, but if the dataset is larger, we can use the ‘elbow’ point (which is 4 or even 2 at this case) to exchange a little bit of bias for a significant computational expense improve. Eventually, I use a 30-fold model to cross validate these 4 models.

A blue and white text

Description automatically generated

In overall, the results doesn’t demonstrate much difference among the eight models in terms of both mean and variance. This outcome is within expectation, as the previous several models indicate very similar results. From the bias/variance perspective, all 8 models don’t indicate a high bias or variance.

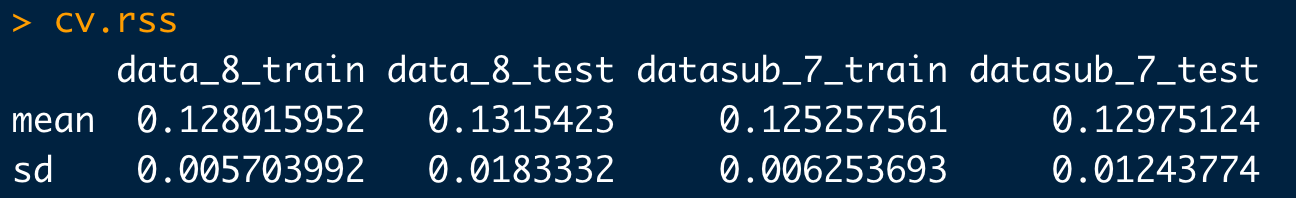
When focusing on different datasets, it is clear that 7 or 8 principle components work out same result for original dataset. On the other hand, for the dataset after subset selection, 7 components perform slightly better than 6 components.

It’s again very trivial to select between the two subsets, but will use 7 components after subset selection data.

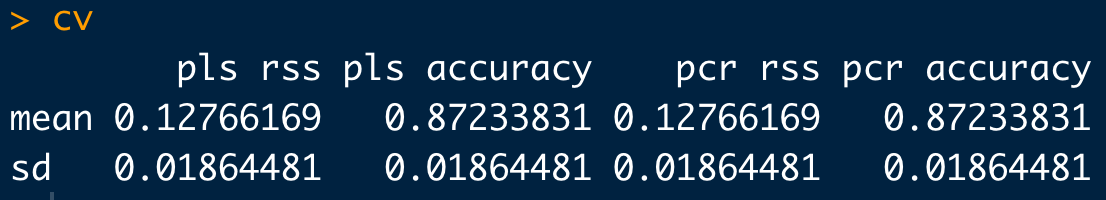
The same procedure is processed with the PCR approach.

A blue and white sign

Description automatically generatedA screenshot of a cell phone

Description automatically generated

The outcomes are very similar as PLS, due to the fact that the principle components with regarding to y or without it are pretty much the same. Choose 7 components after subset selection data.



The results are exactly the same! This somehow illustrates that both PCR and PLS selects very similar 7 components.

*Conclusion*

The results support the descriptions of these methods at the beginning, that PCR and PLS classify data by calculating components. Due to the small dataset, the effect of prediction outcome doesn’t play an important rule in determining which the most important components are.

* 1. Ridge and Lasso Regression

The ridge and lasso regression are two regularization approach. By adding a term with lambda at the very end, these methods can re-weighted the significance of feature in logistic regression. Comparing the two models internally, the ridge regression won’t diminish the weight of any variables but only shrink them, while the lasso regression can.

*Data*

As these two methods adopt the feature selection method in their modelling, it is not necessary to consider the effect brought by subset selection. Therefore, we only compare the original dataset and PCA dataset.

*Method*

I applied the same procedure for ridge and lasso regression with the two chosen datasets. In each program, I first cross validate the choice of lambda using 30-fold, then select the best lambda that minimize the cross validation error. After that, the training and test error is calculated and the coeficients are shown. The results are presented as follows:

|  |  |
| --- | --- |
| Ridge\_original | Ridge\_PCA |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| Lasso\_original | Lasso\_PCA |
| A close up of text on a white background  Description automatically generated | A close up of text on a black background  Description automatically generated |

Figure

The four results don’t suffer from high bias or variance, and they are very similar at the same time. Horizontally, Lasso regression will penalize those variables that are the least significant, such as sex. In Lasso regression of original dataset, the weight of sex even diminishes to 0, and its weight is approximately separated to the other features compared to the ridge. Vertically, the two datasets gain very similar results. This is predictable as both Ridge and Lasso have already include feature selection in their process. The result also justify that after PCA, the original information is well preserved. We therefore consider all of these four methods when comparing accuracies.

*Conclusion*

As regularization method of Logistic regression, the two methods don’t have significant improvement. This is partly because the logistic regression itself doesn’t suffer from overfitting.

* 1. Support Vector Machines with different types of kernel

The support vector machines focus on finding a standard to best separate classes. To achieve better prediction outcome, kernels are usually used. The most commonly used kernels are linear kernel, polynomial kernel and radial kernel. The e1071 package in R also contain 5 svm method, respectively named ‘C-classification’, ‘nu-classification’, ‘eps-regression’, ‘nu-regression’, ‘one-classification’(for novelty detection). This analysis will mainly focus on the ‘eps-regression’(classification by gaining inference from regression).

Albeit the accuracy and stability of svm is usually high, it also requires a long time to process, hence in terms of cross validation, I use a relatively smaller k value.

*Data*

As dataset will have a huge impact on the prediction accuracy of SVM, I applied all three datasets for examination

*Method*

I first process different types of kernels for the three datasets respectively. At this stage, I didn’t apply any cross validation but to gain an initial understanding of the results.

|  |  |  |  |
| --- | --- | --- | --- |
|  | data | data\_sub | data\_pca |
| radial | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| linear | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| sigmoid | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |

From the naïve analysis, we can see that the SVM with linear or sigmoid kernel preforms not as good as the radial kernel. Moreover, the results SVM linear kernels yield is somewhat similar to the logistic regression of the previous analysis gain. I therefore only focus on the radial kernel. It kind of make sense that the SVM model with radial kernel attains the highest accuracy, as the radio kernel is in a sense the weighted nearest neighbor method which shares certain level of similarity as kNN, thus the two methods are expected to have similar accuracy level.

The next step is to shrink the range for the parameter selection (gamma and c in this case). The initial guess is 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100 for both of the parameters. As the model is very sensitive to the parameter selection, I try as many values as possible. First I try on the original dataset. Each cell of this grid represent the corresponding accuracy level to certain parameter setting.

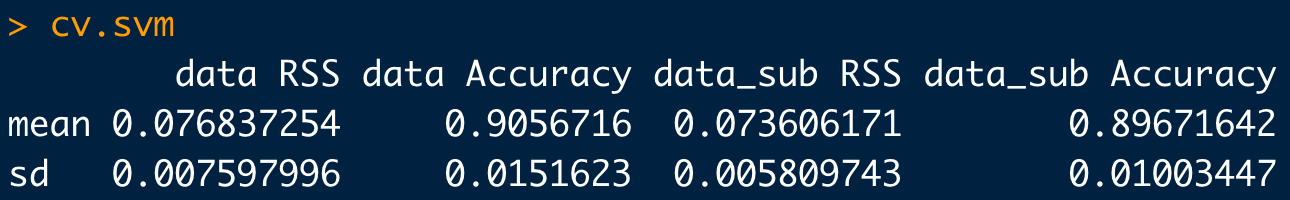


By setting a maximum allowed training/test difference to be around 5%, I figure out the region that might attain the most accurate model. Same steps are applied to the other two datasets.





It is discovered that the data after pca manipulation tends to be darker than the other two, suggesting that it might not be so good a predictor. I therefore conclude the svm model by using only the original dataset and the dataset after subset selection. A 5-fold cross validation is applied to the model.



*Conclusion*

The SVM model does outperforms the Logistic regression related model, however, it is much slower and it requires a lot of parameter settings. This analysis has a limit that it doesn’t examine the impact of all type of svm models.

* 1. Neural Network(NN)

Among all the methods, the neural network might be the one hardest for people to understand intuitively. At the same time, there are a lot of factors to test about. However, the accuracy of Neural Network is usually very high, due to the fact that it continuously decompose the prediction problem to smaller sets of prediction problem. It also takes a lot of memory. I don’t consider large fold of cross validation for this approach. In addition, I don’t consider deep learning in this case, only considering the case of one hidden layer, as one hidden layer can usually capture most of the features and save space and time.

*Data*

Since neural network does contain certain types of feature selection, and plus the dataset is a small one, I only attempt to use the original dataset for evaluation.

*Method*

Although the number of layer isn’t a concern, the number of units inside the hidden layer still needs to be considered. Instead of choosing k-fold cross validation which is not very feasible in R, I use the holdout validation instead. So I divide the dataset to be 0.6/0.2/0.2 for training, development and test respectively. I use the training and development set to validate the best hidden layer number. Accuracy is selected to be the measurement standard.



From the first figure it isn’t hard to see that the accuracy of training set goes up to a high level and fluctuates after approximately 10 hidden layers, whereas the development always fluctuates steadily. From the regression line it can be shown that the accuracy level is increasing with more hidden layers, but ineffectively. The second figure shows that the model suffers from overfitting approximately after 10. We therefore zoom in and observe the pattern between 1 to 10.



Although the graph varies for different choice of training set, it is still observed that the model suffers from higher and higher chance of overfitting issue but gaining accuracy at the same time. It is hard to determine which number is the best, but since the R studio doesn’t support large k-fold cross validation, we choose 3 neurons. After choosing the best hidden neuron number to be , I applied a 8 fold cross validation to evaluate the mean and standard deviation of its variance.

A close up of a sign

Description automatically generated

*Conclusion*

Albeit its high prediction power, neural network contains a lot of parameter setting to adjust and is less user friendly.

1. Discussion

After evaluating all of the above models, The accuracy summary of the models are summarized as follows:

A screenshot of a cell phone

Description automatically generated

*Feature Selection*

Throughout the whole analysis, we compare three datasets in total, namely the original dataset, the dataset after subset selection and the sate set after PCA. Except for kNN which obtains a minor improvement after applying subset selection, the other methods are basically immune to feature selection process. There are two underlying reasons. First of all, the dataset used in this analysis is relatively small and the aim of the analysis is to predict better accuracy instead of gaining more understanding of the data, and thus I didn’t trade-off bias to gain more variance. Secondly, a lot of methods including PLS, PCR, Ridge, Lasso, SVM, NN already have their parameter tuning method built in within the model, the feature selection doesn’t help much with the accuracy. However, if the dataset gets bigger, it’s worth trying to consider the two feature selection method in advance.

*Prediction Model*

In terms of standard deviation, all models maintain at level around 1%, which is stable. In terms of the mean of accuracy, it appears that these models can be stratified into two groups: Logistic, LDA, PLS&PCR, Ridge&Lasso within one group that can achieve approximately 87% accuracy, and kNN, SVM (RBF), NN (one hidden layer) that attain 90% accuracy. The first group of models are basically derived from and closely related with Logistic Regression. As the Logistic Regression attain this accuracy level without high variance and the data isn’t separate enough, all of these models get similar results. Within the second group, SVM and NN yield higher accuracy is because the intrinsic classification function is not linear anymore. These two methods trade-off a bit of variance and gain higher accuracy. This is actually the case for a insurance dataset, since some variables (such as charges) can vary a lot given some other variables fixed (such as sex). Under this circumstance, a non-linear model is intuitively better than a straight linear one. As for kNN, there is an intuitive understanding: if a group of people with similar characteristics (such as very high bmi and few steps) issues claim, than very probably the next people with these characteristics also reports claim. However, kNN is very subjective to features included. An inappropriate data cleaning will effectively impact kNN.

In overall, this analysis suggest that kNN be the best model to predict for this dataset. The model is not only more accurate than some of the models, but also run much quicker than SVM and NN. However, if the dataset starts to grow, the effectiveness of kNN will probably reduce for its non-parametric nature. In that case, we could consider SVM and NN. We can also gain a rough understanding of the data by using Logistic regression related method at an early stage.

Despite the current models used, there are also many other possible solutions. Decision tree and random forest including bagging and boosting related methods, for an example, are very intuitive ways to present the significance level of different features. However, due to the fact that the package *tree*, *randomForest* and *gbm* in R can only accept certain levels in the features input, as well as the high computational expense to produce a similar outcome as the other methods, this analysis doesn’t include these technics as a comparison. A deeper neural network can also be considered with other packages.

1. Appendix
   1. Excel files of raw data



* 1. R program

Execution order of files:

Data pre-processing

-> kNN -> kNN\_sub -> kNN\_pca

-> Logistic regression -> Logistic Regression Overall ->

-> LDA&QDA -> LDA&QDA\_sub -> LDA&QDA\_pca -> LDA&QDA Overall

-> PLS pre-> PLS -> PCR pre -> PCR ->PLS&PCR Overall

-> Ridge -> Ridge\_pca -> Lasso -> Lasso\_pca->Ridge & Lasso Overall

-> SVM -> SVM\_sub -> SVM\_pca -> SVM\_test(change the dataset and the range for multiple times)->SVM Overall

->NN -> NN Overall

-> Overall

* 1. References

<https://www.kaggle.com/easonlai/sample-insurance-claim-prediction-dataset>

An Introduction to Statistical Learning with Applications in R

需要改善的地方有：

1. 用更好的package来进行data pre-processing和绘图(ggplot2)
2. 分成training/development/test， 分配比例如何选择？
3. 加入更多的变量如R^2, AIC, BIC, Decile Plot, Double Lift Chart以选择模型
4. 是否可以加入decision tree？只要对charges进行一些加工
5. 如何改善training set的随机生成？
6. 从regression inference 到 classification， 中间取值怎么判定？