Selection by Hypothesis Testing

Alexander Maslev, Hanxiong Wang, and Minyoung Lee

2018/05/13

1. Introduction

When dealing with model selection, one frequently used method is selection by hypothesis testing in the scope of stepwise forward regression. In this report, our goal is to implement selection by hypothesis testing on simulated data, as well as on a real high-dimensional dataset (where the number of covariates is much greater than the number of observations). To do so, we rely on the *selectiveInference* package that has already been created in R.

The *selectiveInference* package implements forward stepwise regression (function *fs*) and allows us to conduct selective inference on the data by computing p-values for each of the variables (function *fsInf*). With the resulting list of p-values, we can perform model selection by using one of three stopping criteria: 1) the ForwardStop rule, 2) Benjamini and Hochberg's rule and 3) the StrongStop rule.

The first of these has already been programmed in *selectiveInference*, while our objective is to develop algorithms to apply the latter two stopping rules. Then, we check the performance of the three stopping criteria as we vary α, our significance level, and determine the statistical power of each rule using our simulation experiment. Finally, we will apply each rule on a real high-dimensional dataset and compare the results with a measure of prediction error.

This report will first begin with an exploration of the fundamental ideas of selection by hypothesis testing. Then, we will detail our results with the simulated data and present a chart of the power of the tests. Next, we discuss the application of our algorithms to the real dataset and its comparison to prediction error. To conclude, we will summarize the results of our analysis and their implications.

2. Basic Concept of Hypothesis Testing

Stepwise forward regression begins by ordering the covariates from the greatest to the least relevance to the dependent variable. In doing so, it results in a list of sequential p-values that could be used for a series of hypothesis tests using our aforementioned stopping rules. The goal of all this is to control the false discovery rate (fdr):

$$\text{fdr} = \frac{V_p}{R_p(\alpha)}$$

Here, V_p is the number of false discoveries (incorrectly rejected null hypotheses) and $R_p(\alpha)$ is the total number of rejected hypotheses among p possible hypotheses. Similarly, the power of the test is defined as:

$$power = P(reject\ H_0 | H_1\ true)$$

We can see that the ideal algorithm minimizes the fdr and maximizes the power of the test.

With hypothesis testing, each of our potential covariates is assigned a p-value associated to our null hypotheses, that being $\beta_j = 0$ for $j=1,\ldots\ldots,p$. These p-values are not independent, but rather they are dependent on the relationship between the covariate at hand and the proposed model. In other words, they are generated sequentially. All of this is done with the *fs and fsInf* functions in the *selectiveInference* package. In particular, at each step, *fs* adds the predictor that results in the greatest correlation between the predictors and the residual. *fsInf* is subsequently used to calculate the sequential p-values.

If one of these p-values becomes too large (based on the given stopping rule), the variable being looked at and all subsequent variables will not be included in the final model. We decide to set the cutoff point there and have determined our optimal model. Below, we have shown our stopping rules in detail.

ForwardStop:

$$\hat{k}_F = \max\left\{ k \in \{1, \ldots, p\} \middle| -\frac{1}{k}\sum_{j=1}^{k} \log(1 - p_j) \leq \alpha \right\}$$

StrongStop:

$$\hat{k}_S = \max\left\{ k \in \{1, \ldots, p\} \middle| \exp\left\{ \sum_{j=k}^{p} \frac{\log(p_j)}{j} \right\} \leq \frac{k\alpha}{p} \right\}$$

Benjamini and Hochberg:

reject H_0 if $p_k < \alpha_k$, where
$\alpha_k = \alpha(k+1)/p$ and p_k is the k^{th} p-value

The first two rules stop the algorithm when $\hat{k}$ is reached. The Benjamini and Hochberg rule stops it when $p_k > \alpha_k$, and $\hat{k}$ is thus selected as k-1.

Overall, that is the general idea of selection by hypothesis testing, and it usually runs smoothly in conjunction with the R package. However, high-dimensional data presents specific problems that need to be dealt with by modifying the general algorithm and using a form of dimension reduction. We will discuss this in the next section.

3. Simulation

We simulate the high dimensional data to evaluate performance of the different stopping rules. we generate data from multivariate normal distribution X ~ N(0, I) with p = 1000 and n = 100. We can estimate $\hat{y}$ by the relation $y = X_{n*p}\beta + \epsilon$ where ϵ ~N(0,0.5). We choose the generating vector $\beta = [5,7,0,2,4,6, rep(0, p-6)]$. So here we have v1 v2 v4 v5 v6 which is not equal to 0. Later we are going to use this result to test whether the result will fit our simulation.

1) Hypothesis testing with full data set

In this part, we apply Strongstop and Benjamini and Hochberg to select variable after the ordering process with fs function in r. We use data generated in the setting above.

```
##     Strong Stop Benjamini and Hochberg
## 0           0                  0
## 0.05        2                  0
## 0.1         2                  0
## 0.3         2                  0
## 0.5         2                  0
## 0.7         2                  0
## 1           2                  0
```

As shown above, we choose only two variables in Strong stop and we don't choose any variable in Benjamini and Hockberg when $\alpha = 0.01$. Even though we increase the alpha, they don't include anymore variables. Thus power of the Strong Stop is just 0.2 and it is zero for Benjamini and Hockberg for all the alpha.

```
p_valus_order_variable[1:10,]

##      order_variable      p_all
##  [1,]              2 0.009374754
##  [2,]              6 0.638385321
##  [3,]              1 0.291800602
##  [4,]              5 0.275337675
##  [5,]              4 0.602617309
##  [6,]            541 0.857022569
##  [7,]            143 0.326932100
##  [8,]            829 0.185105645
##  [9,]             58 0.421983543
## [10,]            353 0.091444155
```

As we can see above, from the second step, we have very large p value that leads us to exclude variable from the step.

2) Hypothesis testing with reduced data set

As we saw in the previous part, we have problem of inflation of the p value if we do not reduce the dimension first and order the variable using fs function. Also there is warning

that we need p < n/2 for the function. Thus, we reduce the dimension of the data set to p = n/2 based on the correlation with y and X's to avoid problem.

```
##      order_variable       p_all
##  [1,]              1 0.000000e+00
##  [2,]              2 0.000000e+00
##  [3,]              5 0.000000e+00
##  [4,]              6 0.000000e+00
##  [5,]             25 3.888639e-88
##  [6,]              8 6.243045e-01
##  [7,]             38 6.942853e-01
##  [8,]             13 8.177296e-01
##  [9,]             39 3.573905e-01
## [10,]              14 1.464169e-01
```

If we compare the p value of the result of full data set and reduced data set, we can clearly see that p value of the reduced data set is decreased.

First, we can see the result of "Strong Stop" below.

```
##   order_variable       p_all var_name
## 1              1 0.000000e+00       V2
## 2              2 0.000000e+00       V6
## 3              5 0.000000e+00       V1
## 4              6 0.000000e+00       V5
## 5             25 3.888639e-88       V4
```

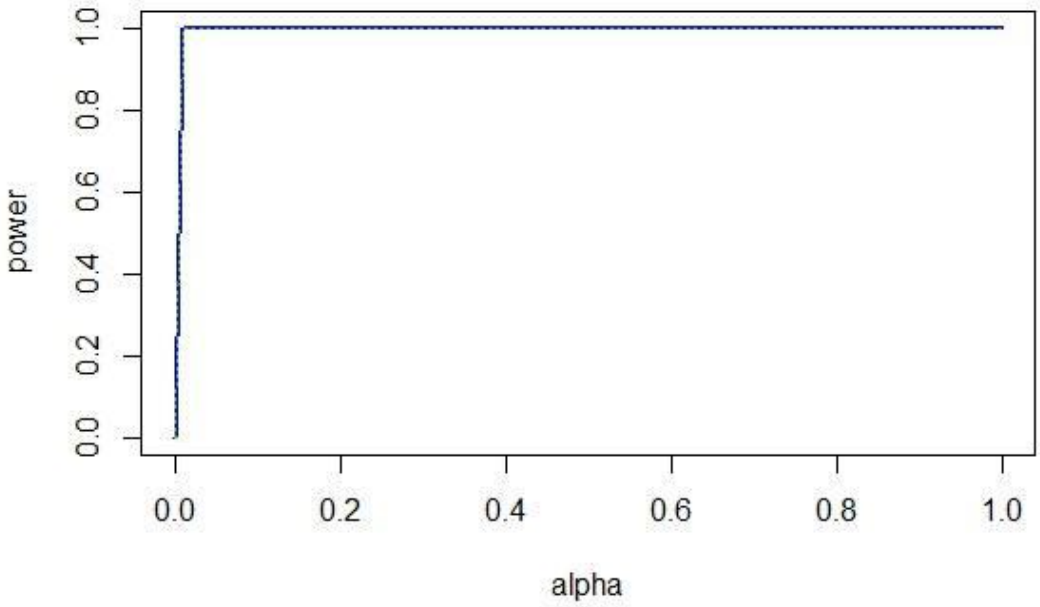
We select the variables "v1 v2 v5 v6" with Strong stop when alpha is 0.05 where the true beta that is not equal to zero are "v1 v2 v4 v5 v6". We choose 4 of true value over 5 candidate which is good compare to the previous result.

Second, we can see the result of "Benjamini and Hochberg" below.

```
##   order_variable       p_all var_name
## 1              1 0.000000e+00       V2
## 2              2 0.000000e+00       V6
## 3              5 0.000000e+00       V1
## 4              6 0.000000e+00       V5
## 5             25 3.888639e-88       V4
```

Similar to the result of "Strong Stop", we select four true variables out of five when alpha is 0.05. Which is big improvement compare to the previous result that we have no variable selected even when alpha is one.

Finally, We plot the power of the test.



We can see from the plot above that the power of the test reaches to one when alpha is 0.01. It means they select all the true non-zero variables among the candidate after the dimension reduction of the data set.

4. Real data

We will demonstrate the application of our model selection algorithms using the Gordon (2002) lung cancer data set. The data set was used to help the researchers pathologically distinguish malignant pleural mesothelioma (MPM) from adenocarcinoma (ADCA) of the lung. The Gordon data set contains a sample size of n=181 tissue samples divided into the two classes, 31 MPM and 150 ADCA. For each of our 181 samples, we have p=12,533 measurements of various gene expression levels. Thus, this is a truly high-dimensional data set, with p>>n. The authors' goal was to determine if a small number of genes can be used to classify the samples, and we aim to do the same using our hypothesis testing algorithms.

As we saw from the simulation study, if we use fs function with high dimensional data set, it will produce inflated p value which will cause the poor selection of the variables. Thus we first reduce dimension of the data set based on the correlation of the covariates with response variable, then we apply different stopping rules.

First, We use the Benjamini and Hochberg rule to select the variable of the data set.

```
##   order_variable       p_all var_name
## 1              1 5.080946e-302    V3332
## 2             15  1.000526e-46    V7997
## 3             49  1.872329e-95    V9219
## 4              9  5.687249e-04    V8362

##   order_variable       p_all var_name
## 1              1 5.080946e-302    V3332
## 2             15  1.000526e-46    V7997
## 3             49  1.872329e-95    V9219
## 4              9  5.687249e-04    V8362
```

We set the alpha to the 0.05 and 0.1 to compare the result. As shown above, we select four variables which are V3332, V7997, V9219 and V8362 for the given alpha.

Second, we use the Strong rule to select the variable of the data set.

```
##   order_variable       p_all var_name
## 1              1 5.080946e-302    V3332
## 2             15  1.000526e-46    V7997
## 3             49  1.872329e-95    V9219

##   order_variable       p_all var_name
## 1              1 5.080946e-302    V3332
## 2             15  1.000526e-46    V7997
## 3             49  1.872329e-95    V9219
```

We set the alpha to the 0.05 and 0.1 to compare the result. As shown above, we select three variables which are V3332, V7997 and V9219 for the given alpha. Compare to the Benjamini and Hochberg, We select one less variable which is V8362.
To compare the two different rules, we build a model with selected variables to see the performance of the model.

```
## Benjamini and Hochberg          Strong Stop
##              0.7454545            0.6000000
```

As we can see the result above, the model based on the Benjamini and Hochberg rule performs much better than the Strong Stop rule. The difference of the two method is selection of the variable “8362”. We don’t know the property of the variable but we can assume that it is very important variable for the predicting the mesothelioma or the adenocarcinoma.

5. Conclusion

Overall, we can see that no model selection method by hypothesis testing is perfect, but it seems that the algorithm implementing the Benjamini and Hochberg rule performs the best for our specific dataset. In other cases, it may be that the StrongStop rule has the best predictions, or even that none of them perform well.

Of course, we have seen how it is essential to perform dimension reduction before utilizing the algorithms; otherwise there is a tendency to select far too few covariates. The *selectiveInference* package itself is not built to handle well datasets with $p>n/2$, and high-dimensional datasets used in their raw form seem to create confusion.

We must be careful in using correlations for dimension reduction, as there is a possibility that we eliminate some important variables before the data is even run through our algorithms. Unfortunately, it is necessary due to the inherent structure of the R code. On the whole, selection by hypothesis testing within the framework of stepwise forward regression can be a powerful tool to extract information from seemingly overwhelming or hopeless datasets. If we are careful in how we approach the analysis, we can achieve plenty of useful results such as the ability to aid in medical diagnosis.

Bibliography

Victoria-Feser, M.-P. (2018). *A Lecture in Model Selection in High Dimensions*, Research Center for Statistics, GSEM, University of Geneva, Switzerland.

Gordon G.J., Jensen R.V., Hsiao L.L., Gullans S.R., Blumenstock J.E., Ramaswamy S., Richards W.G., Sugarbaker D.J., Bueno R.
Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma
Cancer Res., 62 (17) (2002), pp. 4963-4967