Statistical Learning

Dawid Dieu

Assignment 2

Prediction error and information criteria

tags: SL

Generate the design matrix $X_{1000\times 950}$ such that its elements are iid random variables from $N\left(0, \sigma = \frac{1}{\sqrt{1000}}\right)$. Then generate the vector of the response variable according to the model

$$Y = X\beta + \epsilon \ ,$$

where $\beta = (3, 3, 3, 3, 3, 0, ..., 0)^T$ and $\epsilon \sim N(0, I)$.

Task 1

In this task we will examine different calculations of the Prediction Error of models of various sizes.

Subtask 1: For each of the considered models

a)

Estimate β with the Least Squares method and calculate residual sum of squares and the true expected value of the prediction error.

- As one can see looking at the table below, RSS is getting smaller with the model size. This is as expected because the model is able to overfit to the data.

b)

Use the residual sum of squares to estimate PE assuming that σ is known and replacing σ with its regular unbiased estimator.

- When calculating PE for known σ I used the following formula: $PE=RSS+2*\sigma*p$
- When calculating PE for unknown σ I used the following formula: $PE=RSS+2*MSE*p$

PE for unknown σ performed much better, because it wasn't shrinking. And we know that including more than 5 first features introduces only redundant variables to our model. Therefore we would expect the PE to stay at the same level for number of columns greater than 5.

c)

Estimate PE using leave-one-out crossvalidation (do not perform analysis 1000 times but apply the formula for leave-one-out cross-validation error provided in class).

- When calculating PE using CV I used the following formula: $PE = CV = \sum_{ni=1}(Y_i-\hat{Y}_i1-M_{ii})^2$

Here we can see that the errors were cumulating when the model size was growing. This method becomes pretty unstable for bigger models (> 500 columns in our case).

columns	RSS	PE true	PÊ known σ	PÊ unknown σ	PÊ using CV
2	1047.795	1000.998	1047.799	1051.995	1054.167
5	1029.658	1000.995	1029.668	1040.006	1042.402
10	1025.875	1000.99	1025.895	1046.6	1048.79
100	895.2547	1000.9	895.4547	1094.2	1107.85
500	461.5009	1000.5	462.5009	1384.503	1862.233
950	36.20127	1000.05	38.10127	1411.849	15838.43

Subtask 2: Select the optimal model using two versions of AIC: for known and unknown σ.

Here I was experimenting with two solvers `stepwise` and `fast_forward`. The latter one perfomed much faster, especially when building model from 950 features.

I used two variations of AIC function.

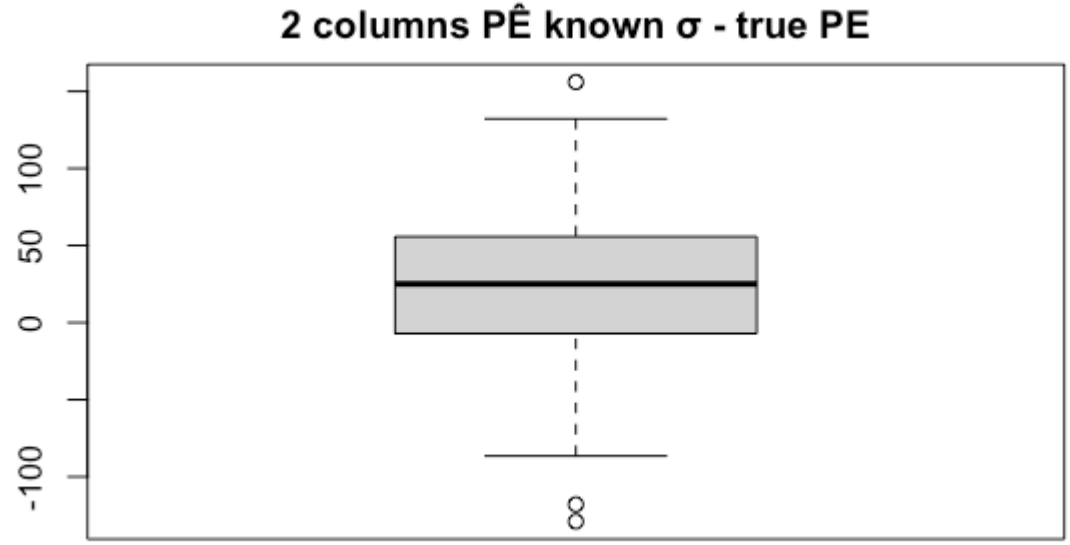
1. Known sigma: AIC = $n * e^{-2*loglikn} + 2 * k * 1$, and the 1 is sigma of epsilon
2. Unknown sigma: AIC = $-2 * loglik + 2 * k$

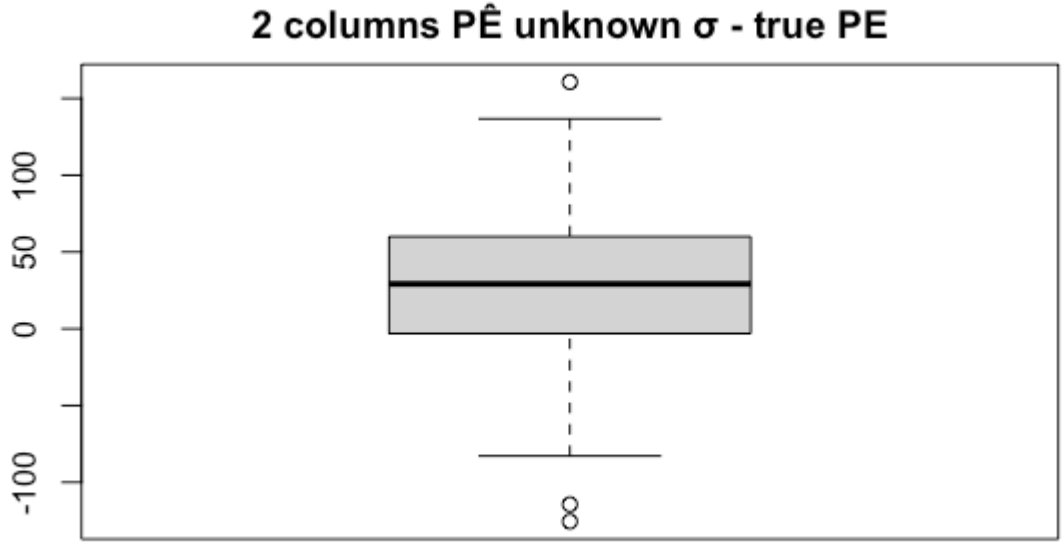
columns	Known σ selected features	Known σ selected features
2	“2”	“2”
5	“2” “4” “5” “6”	“2” “4” “5” “6”
10	“2” “4” “5” “6” “8”	“2” “4” “5” “6” “8”
100	18 features	18 features
500	70 features	70 features
950	70 features	70 features

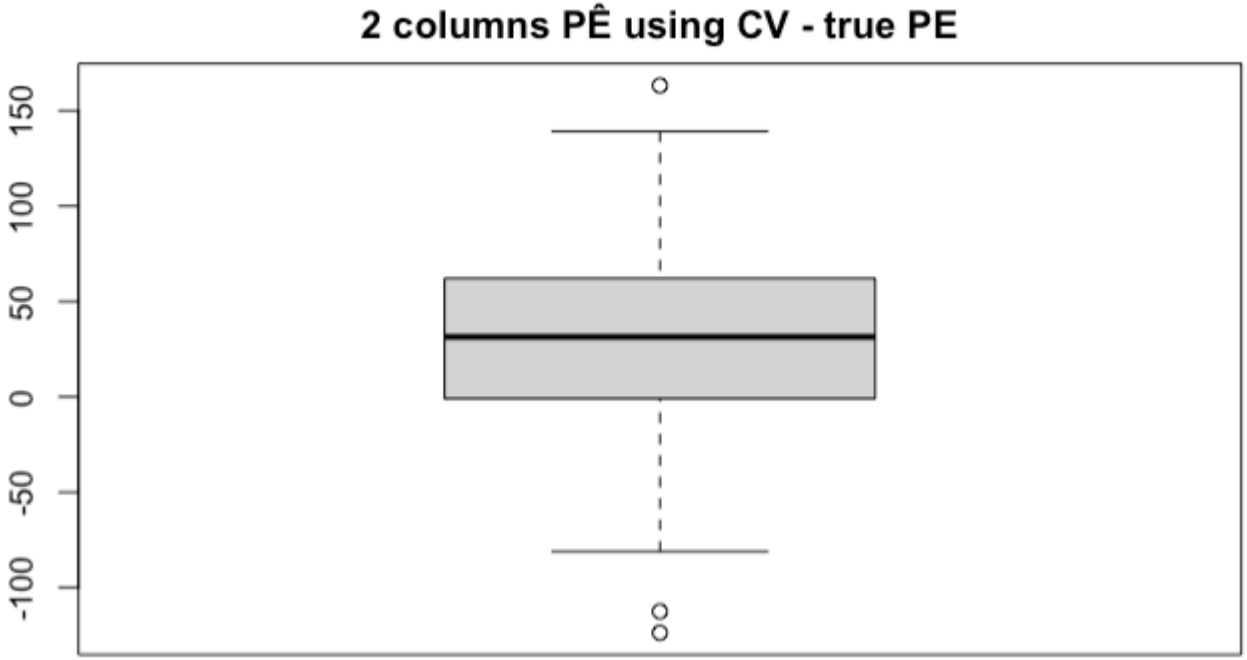
Subtask 3: Repeat the above calculations 100 times and:

- for each of the considered models compare the boxplots of ^PE – PE for three estimates of PE, mentioned above.

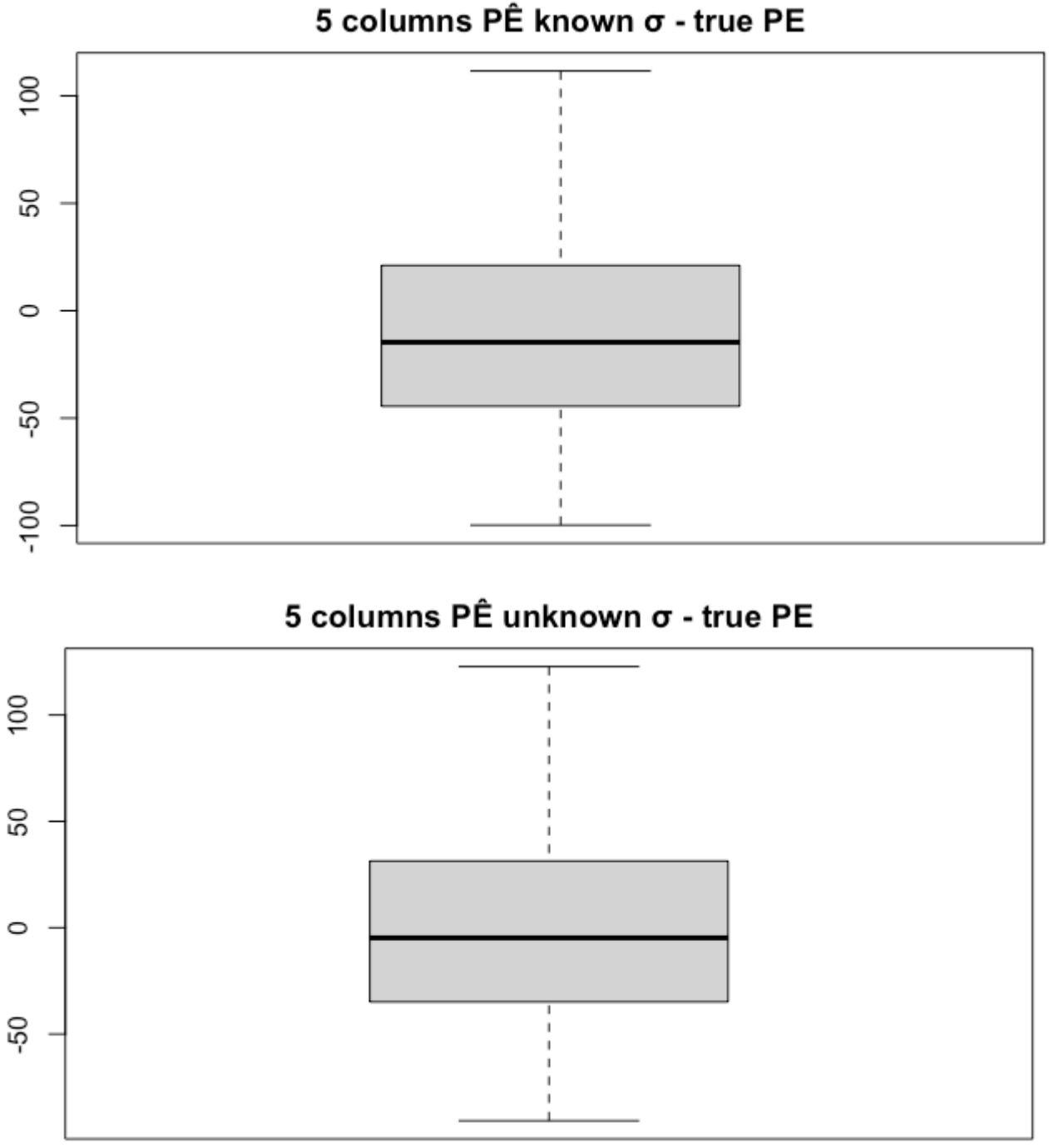
a) 2 columns

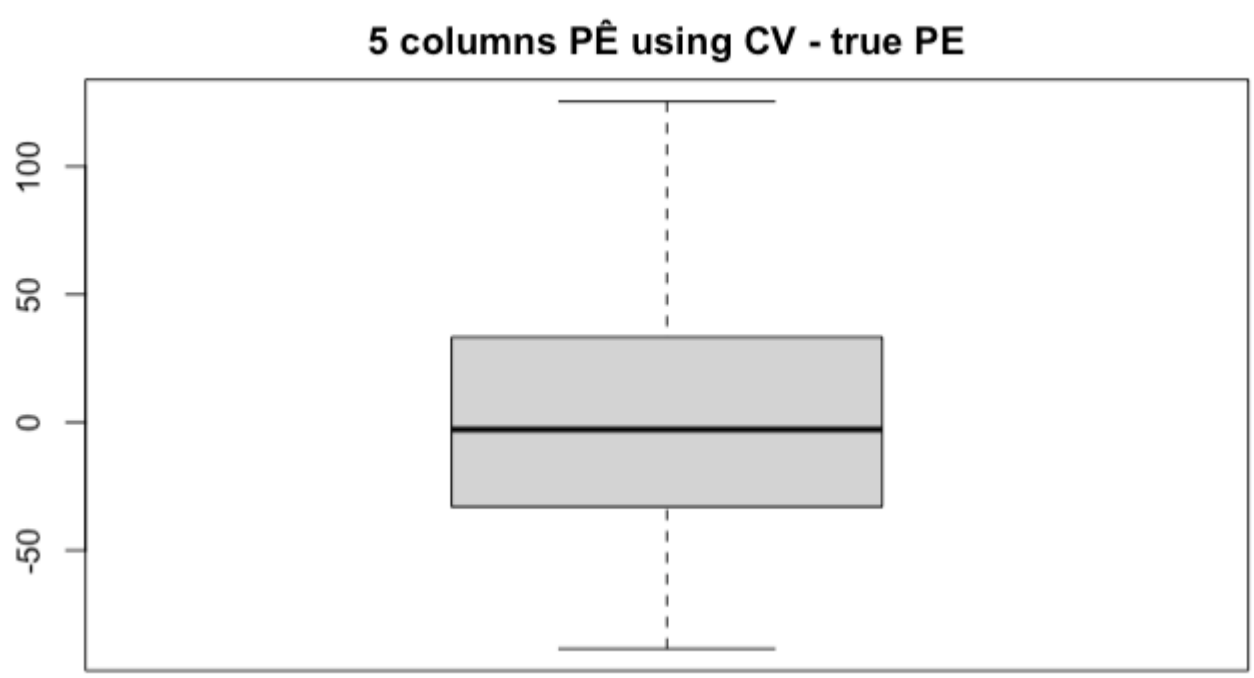

2 columns PÊ known σ - true PE
100
50
0
-100


2 columns PÊ unknown σ - true PE
100
50
0
-100

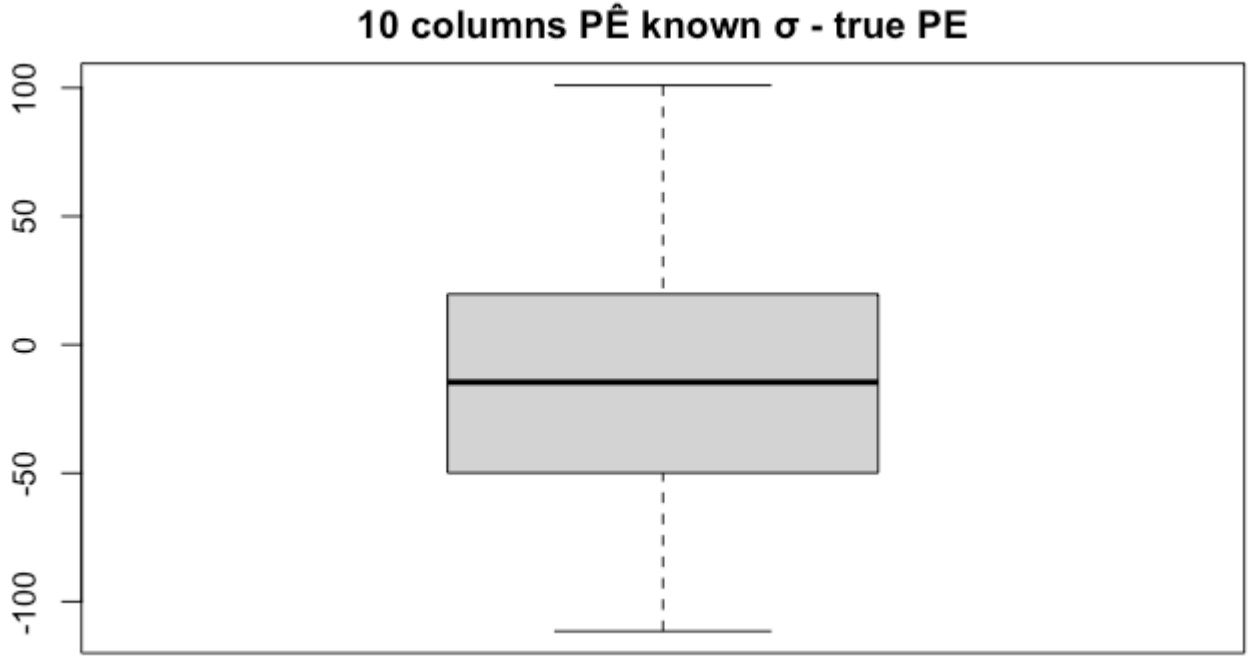

2 columns PÊ using CV - true PE
150
100
50
0
-50
-100

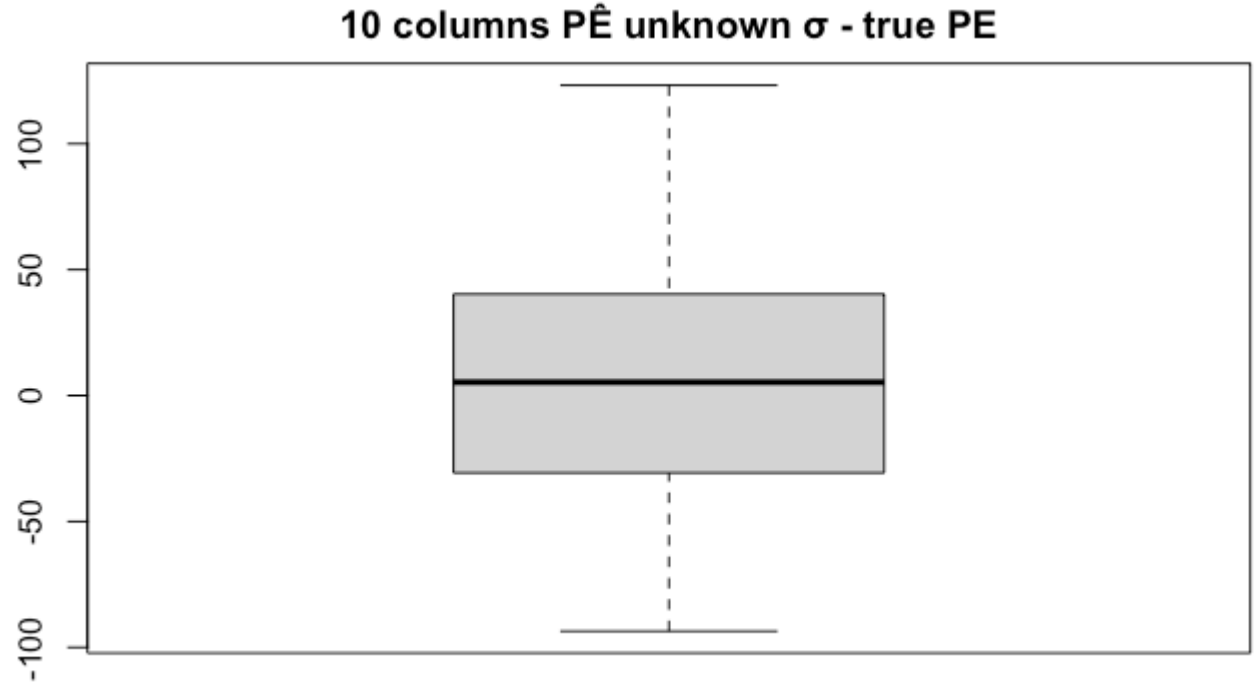
b) 5 columns

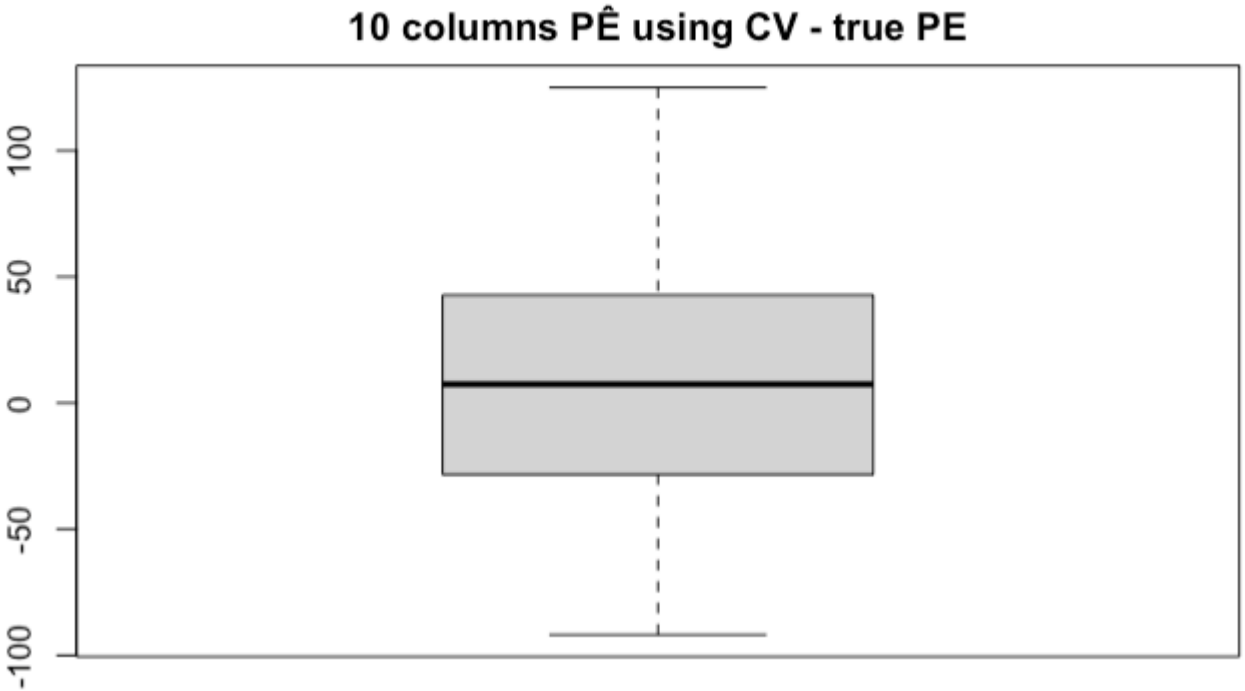

5 columns PÊ known σ - true PE
100
50
0
-50
-100
5 columns PÊ unknown σ - true PE
100
50
0
-50


5 columns PÊ using CV - true PE
100
50
0
-50

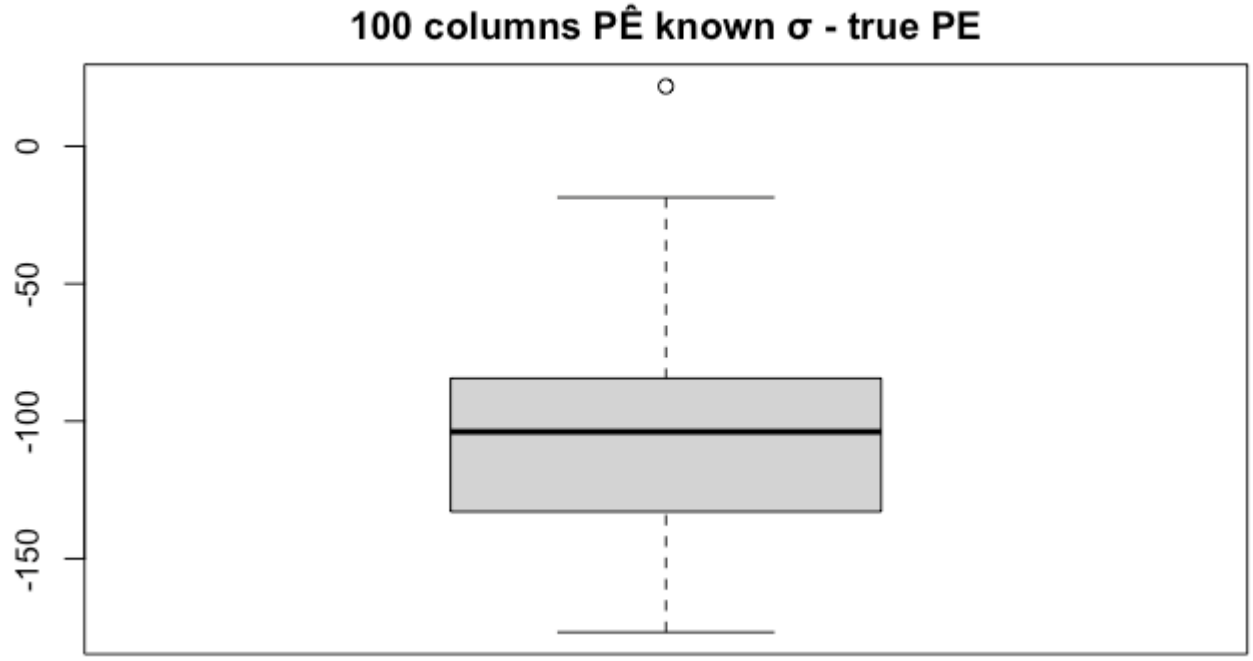
c) 10 columns

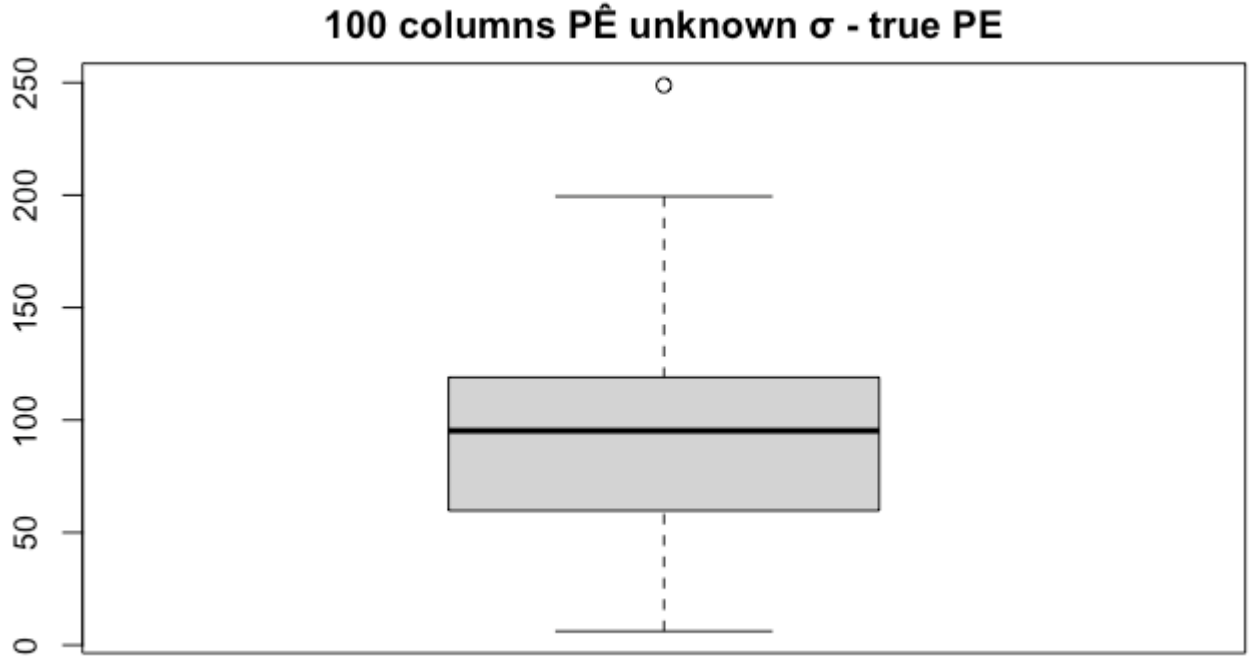

10 columns PÊ known σ - true PE
100
50
0
-50
-100

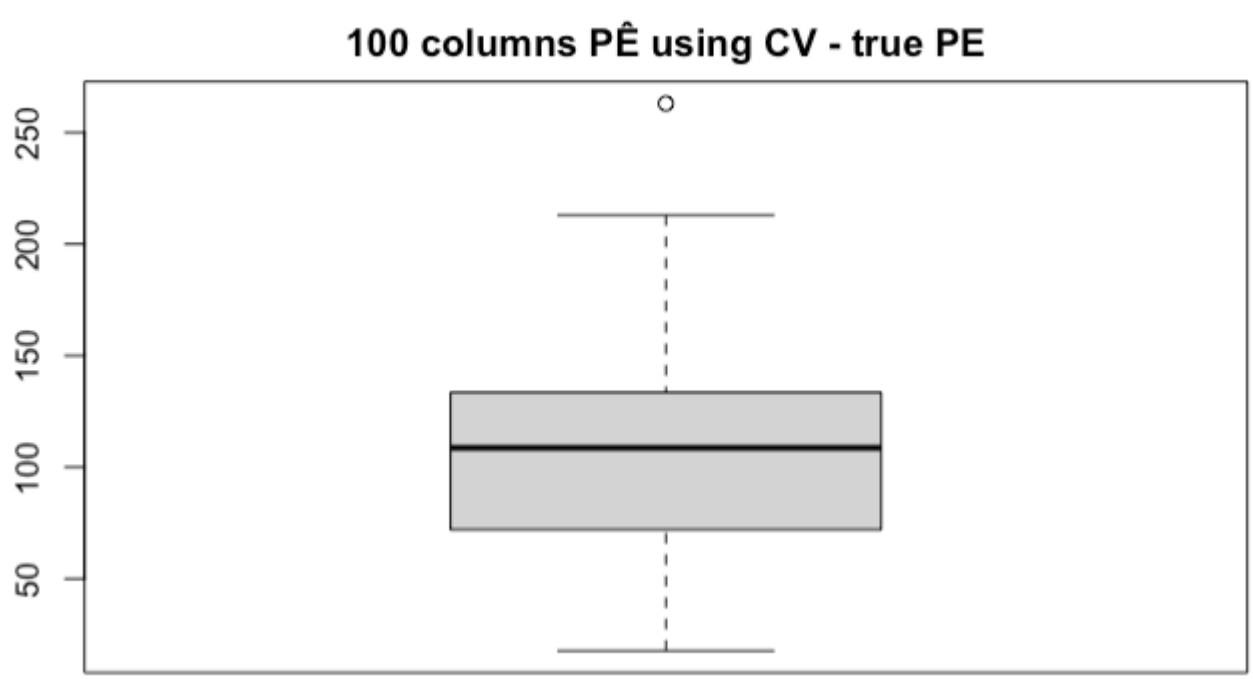

10 columns PÊ unknown σ - true PE
100
50
0
-50
-100


10 columns PÊ using CV - true PE
100
50
0
-50
-100

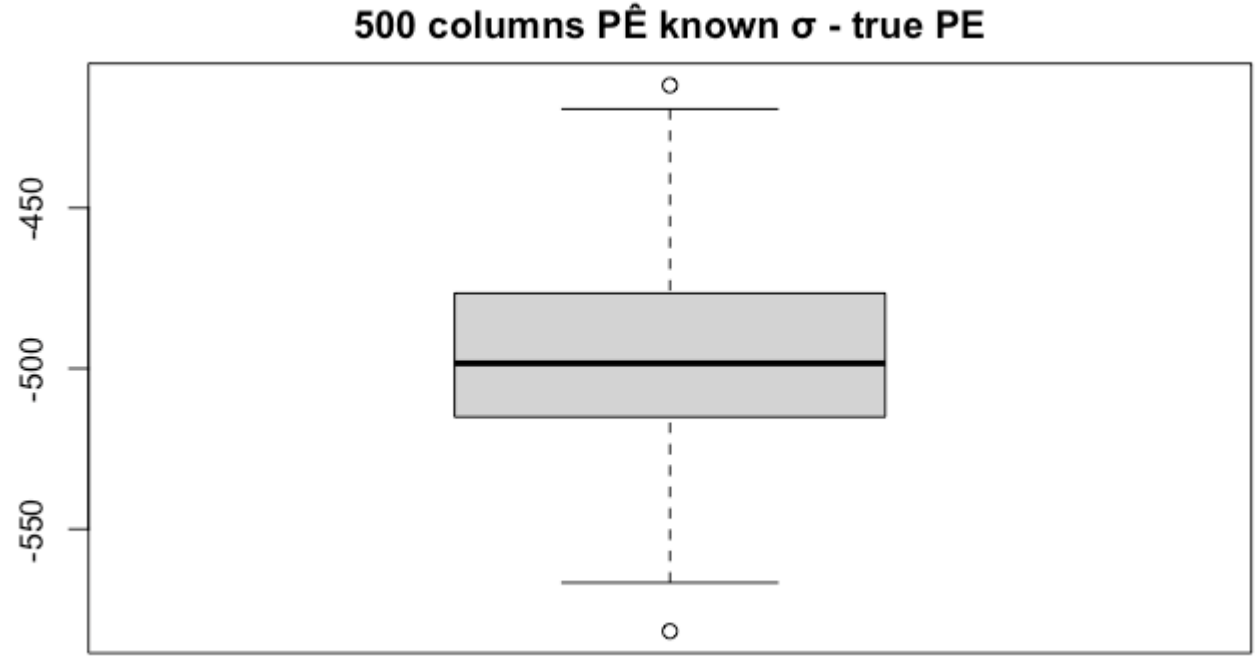
d) 100 columns

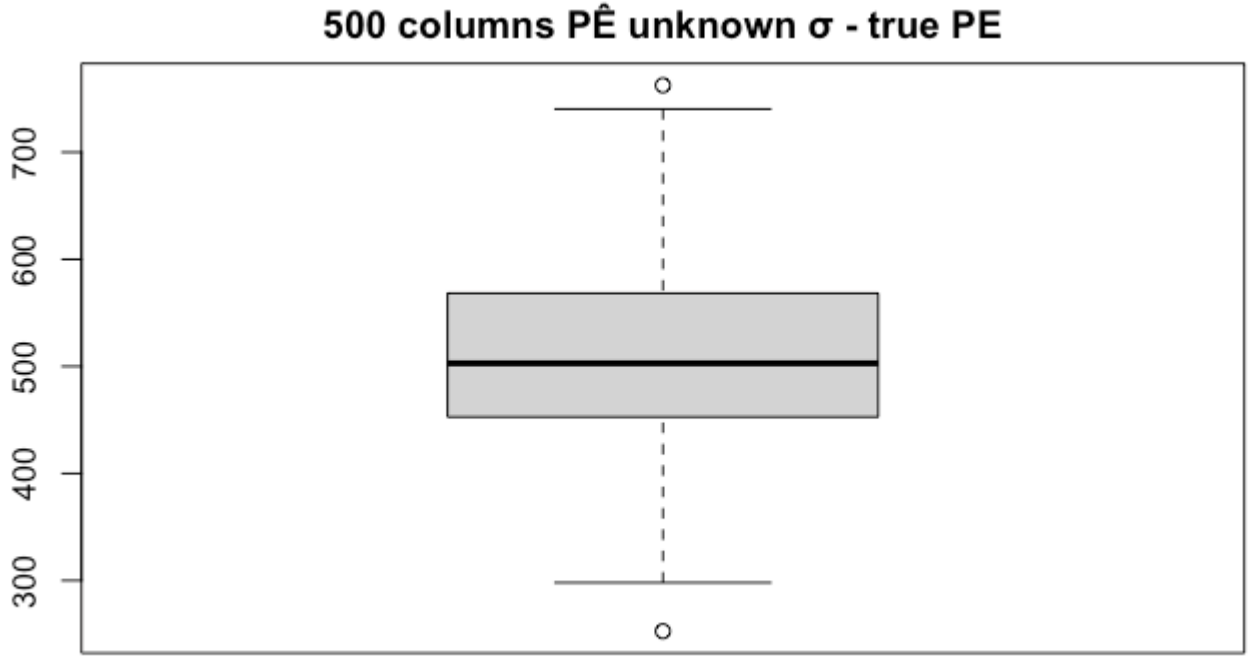


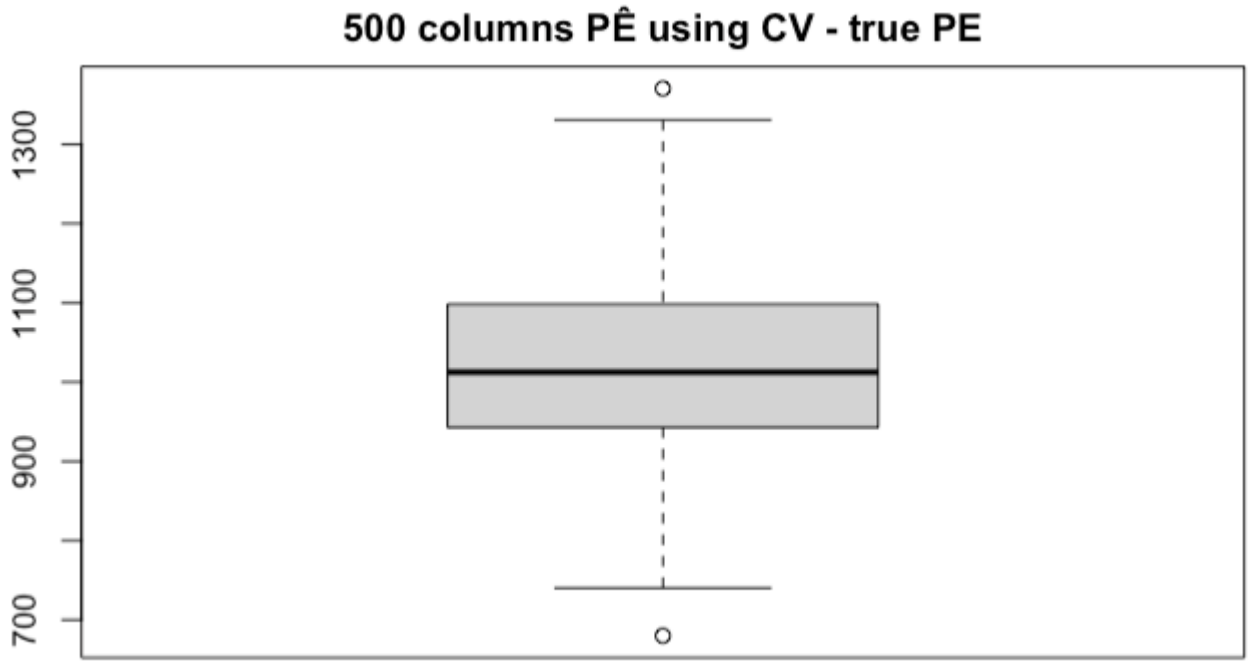



100 columns PÊ using CV - true PE
250
200
150
100
50

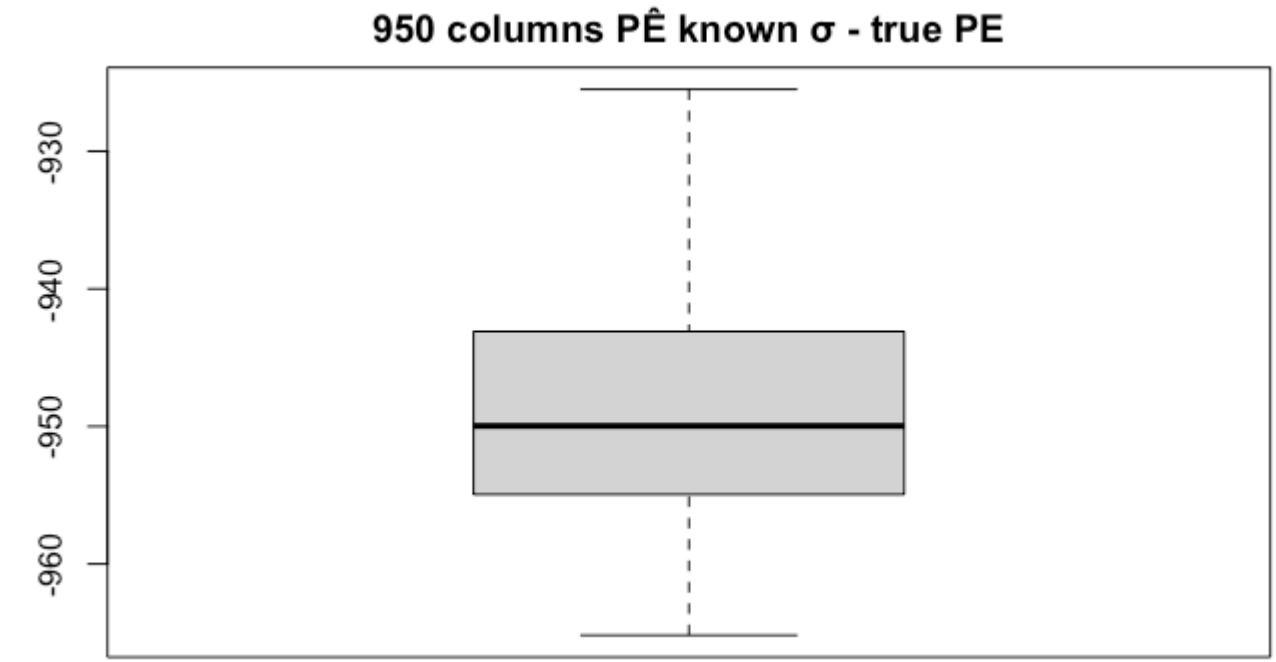
e) 500 columns

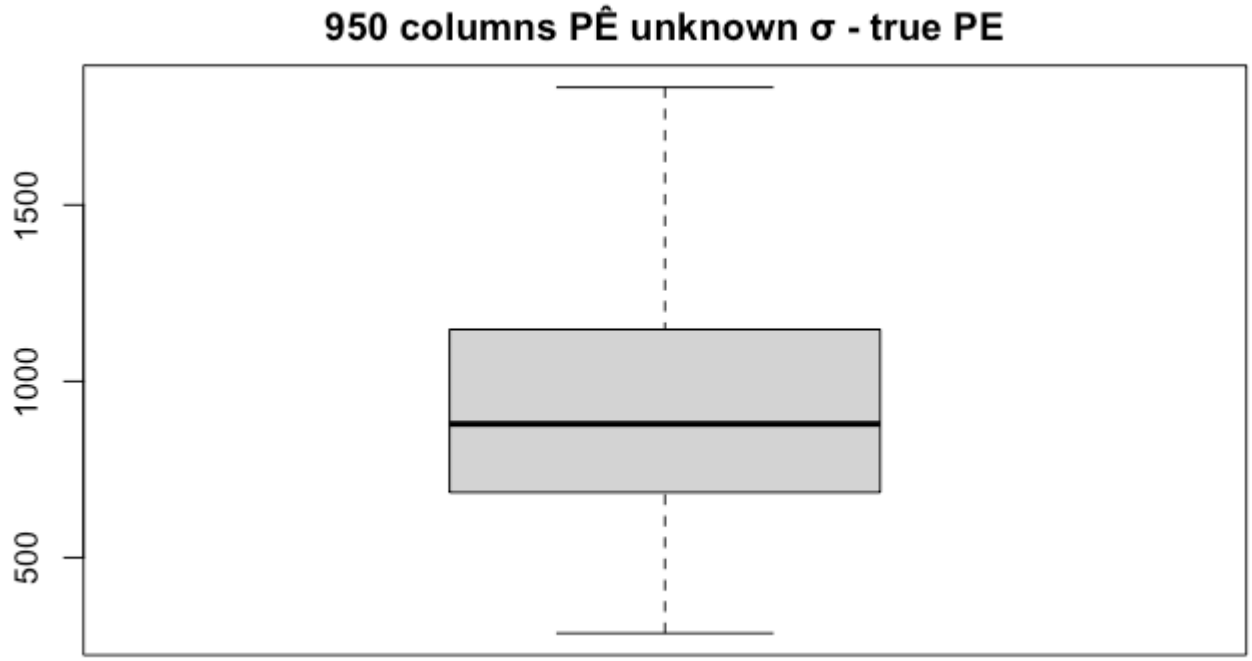


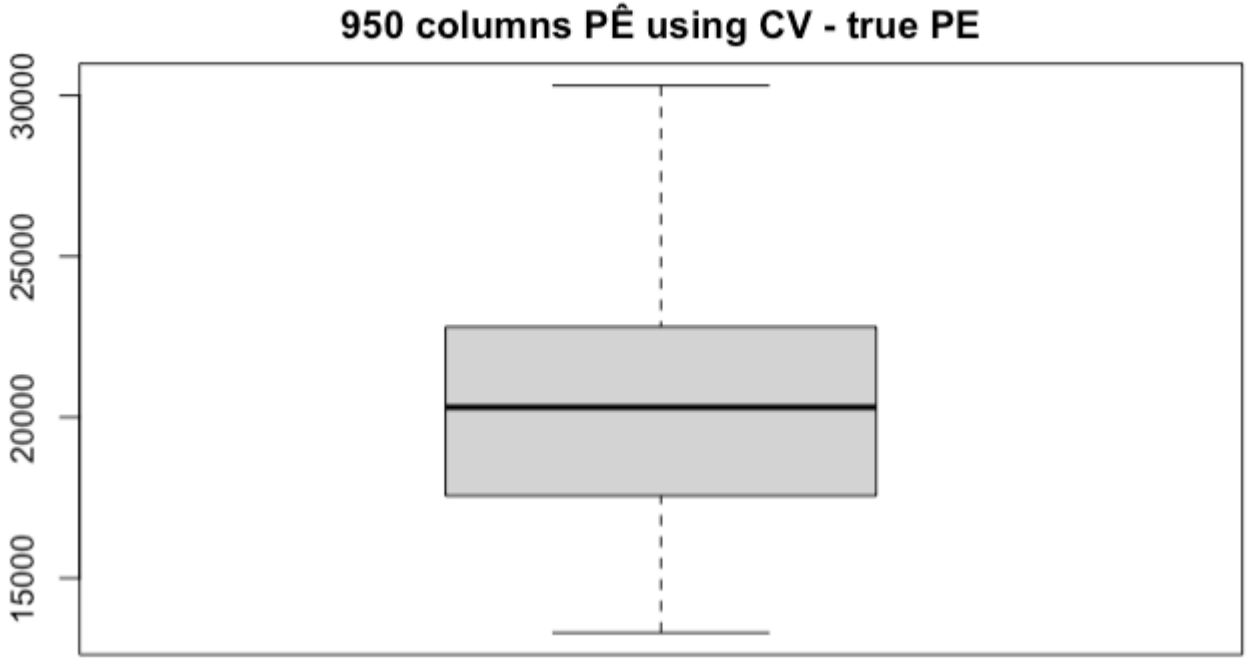



500 columns PÊ using CV - true PE
1300
1100
900
700

f) 950 columns





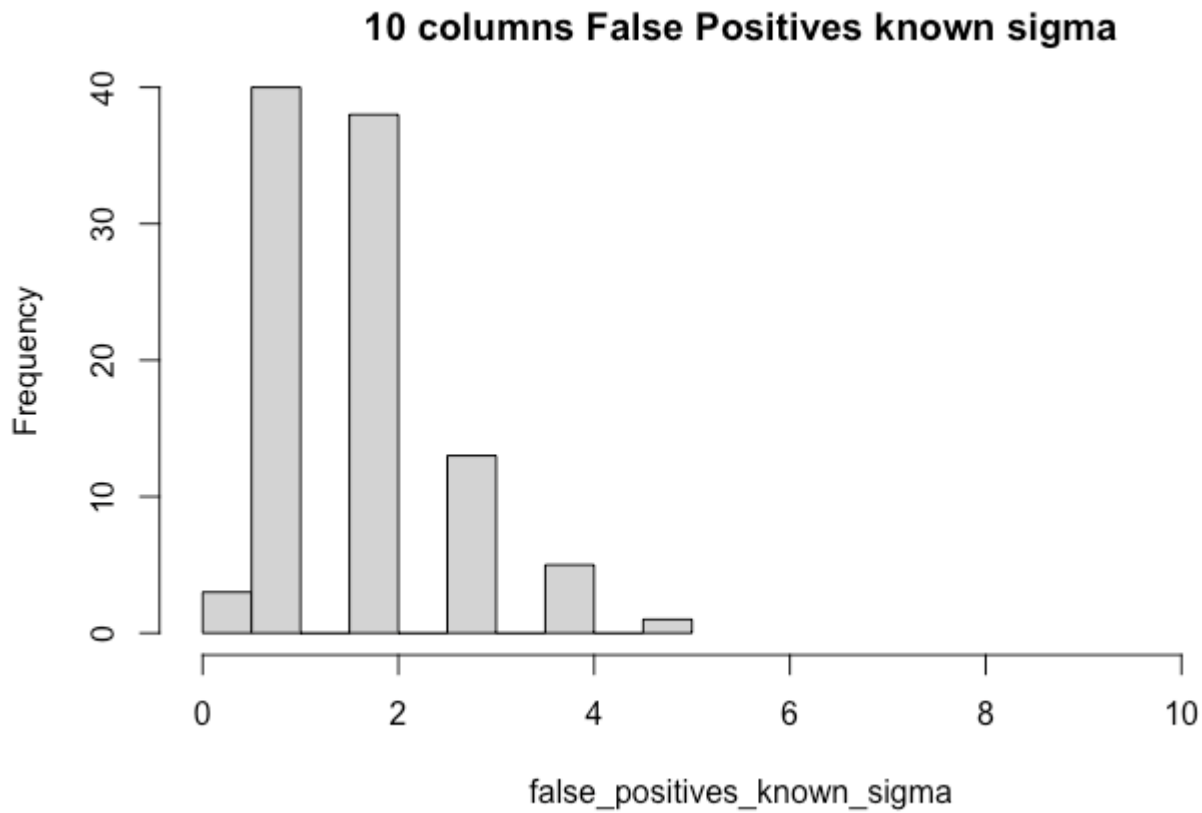


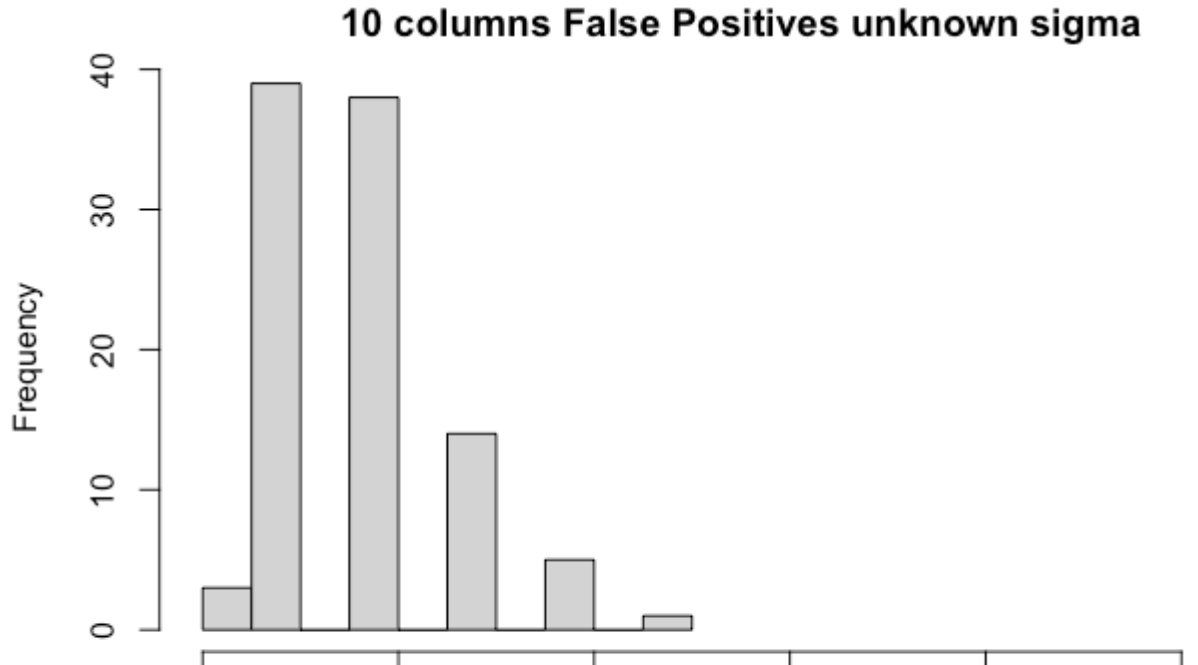
We can clearly see with number of columns bias and deviation from true PE grows. As describe above PE estimated using CV is most unstable.

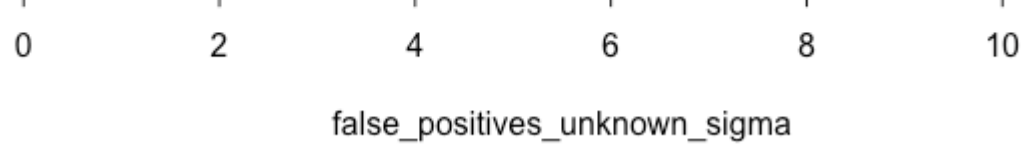
- Provide histograms of the number of false negatives and false positives produced by both versions of AIC (with known and unknown σ).

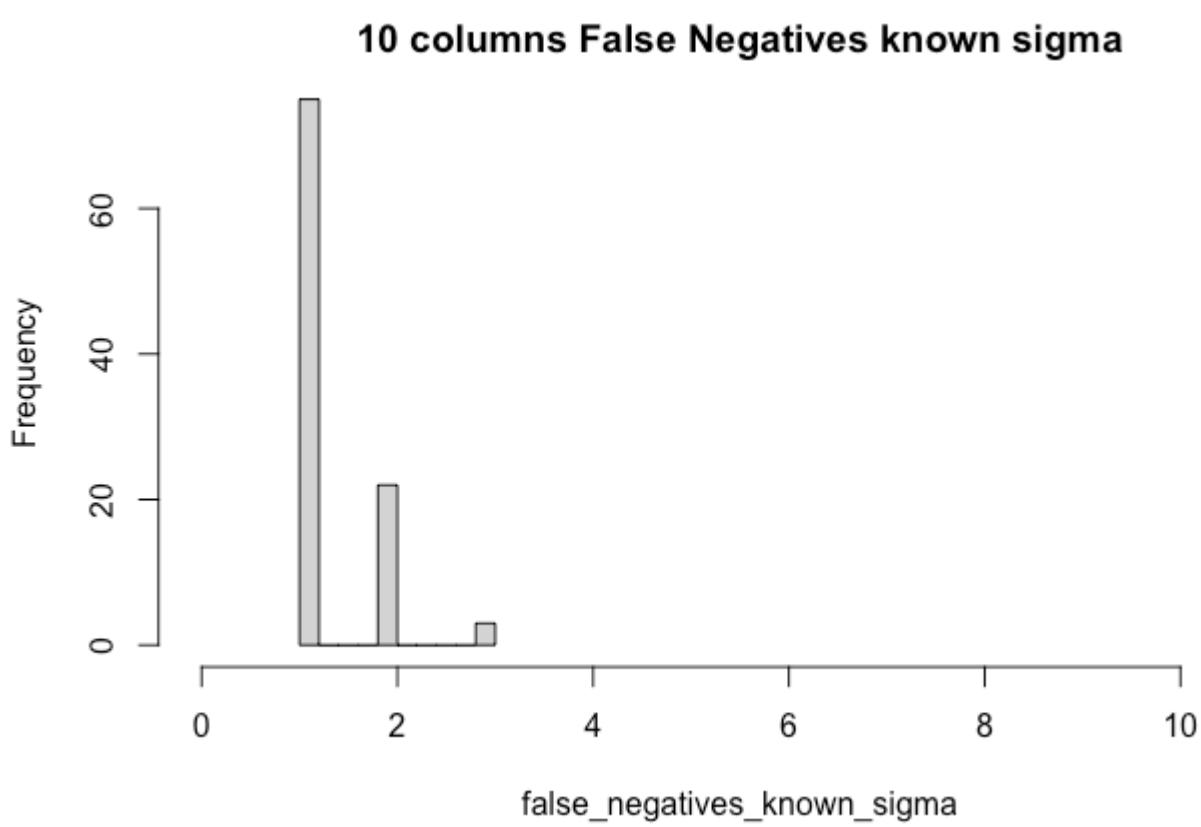
I skipped 2 and 5 columns charts because they were not that interesting.

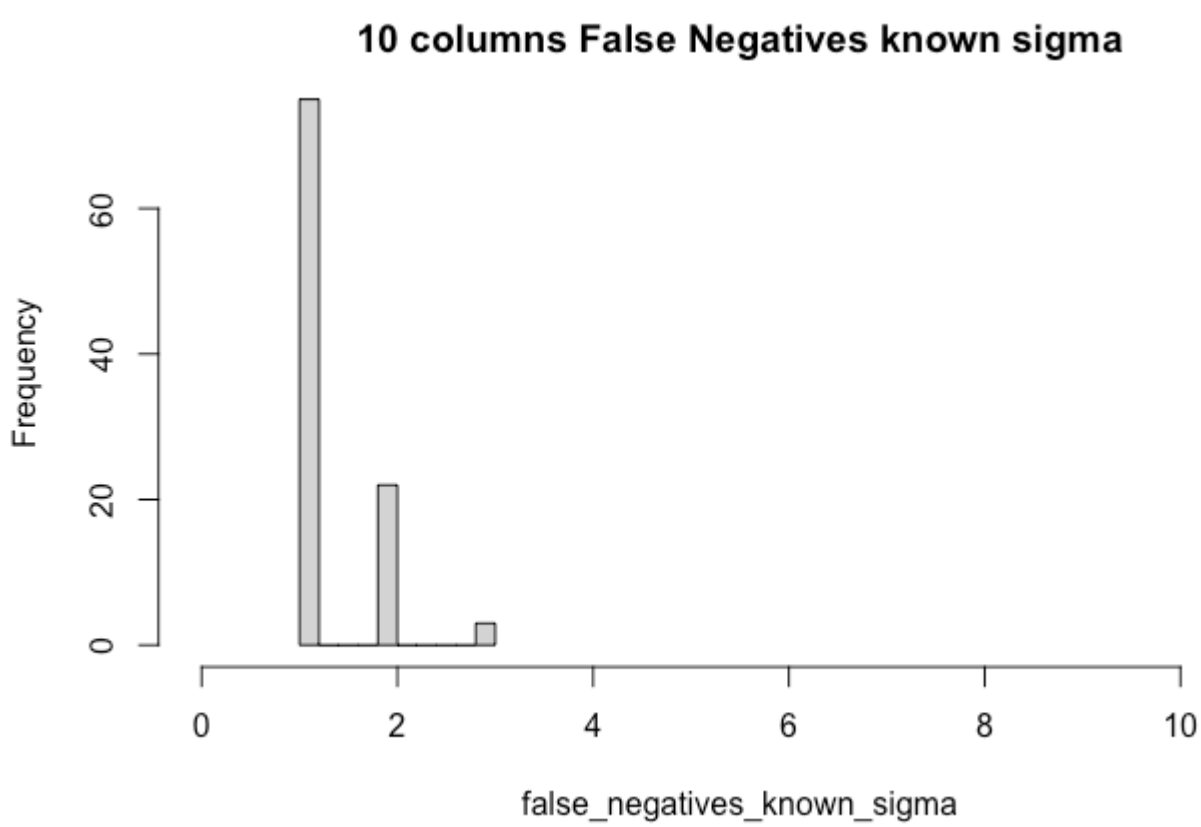
a) 10 columns


10 columns False Positives known sigma
Frequency
0
10
20
30
40
0
2
4
6
8
10
false_positives_known_sigma

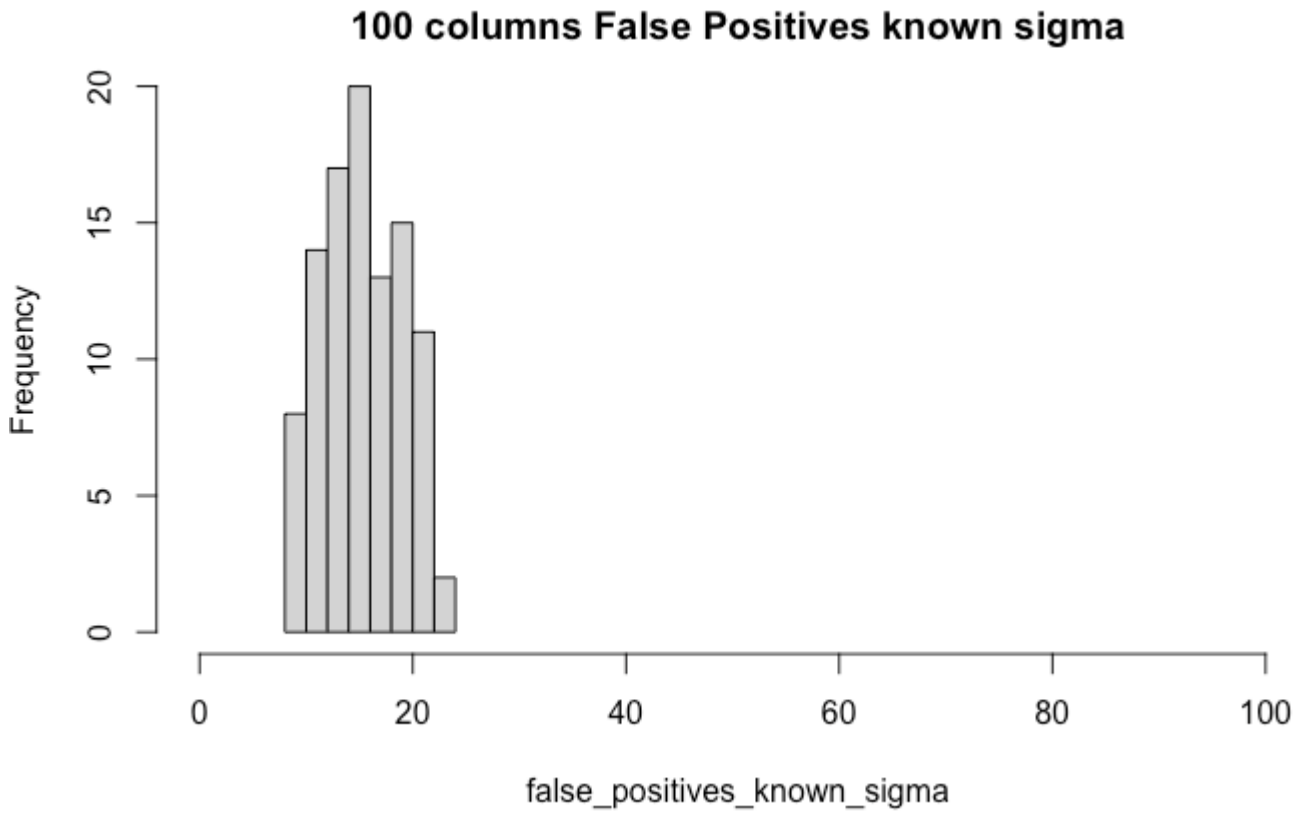

10 columns False Positives unknown sigma
Frequency
0
10
20
30
40

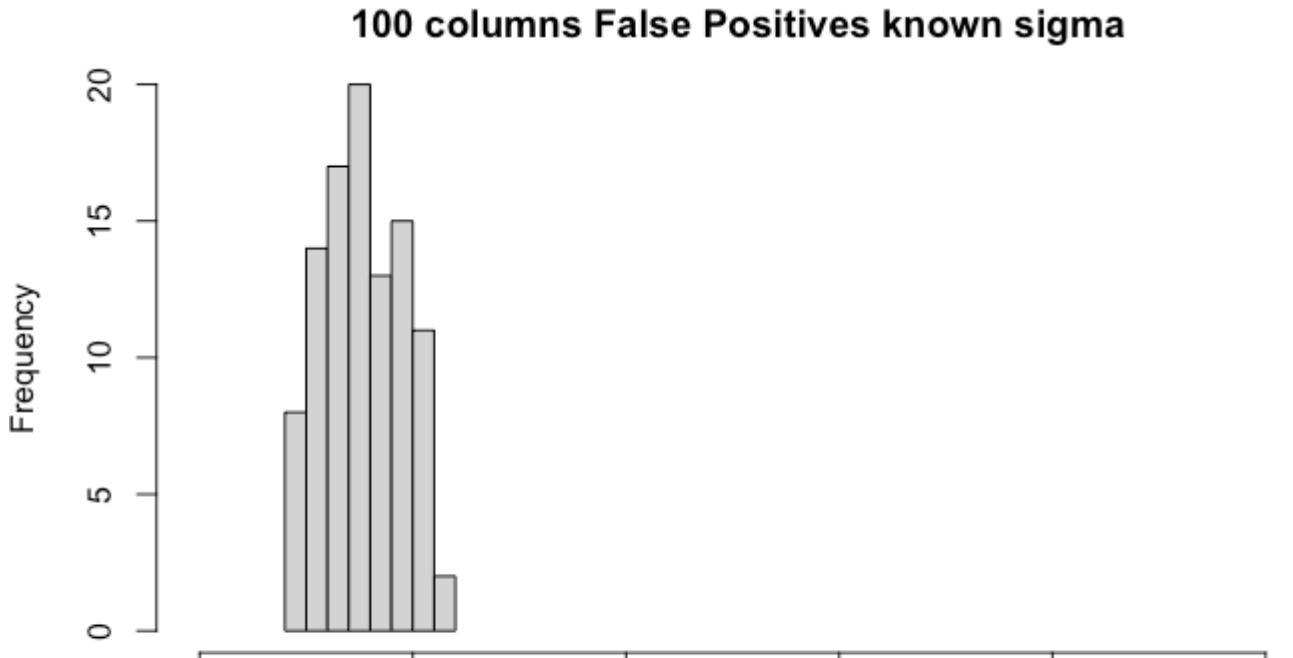

0
2
4
6
8
10
false_positives_unknown_sigma

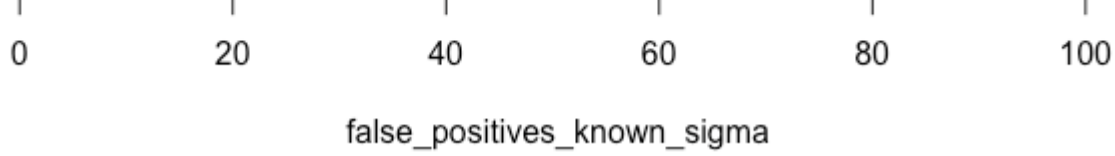

10 columns False Negatives known sigma
Frequency
0
20
40
60
0
2
4
6
8
10
false_negatives_known_sigma

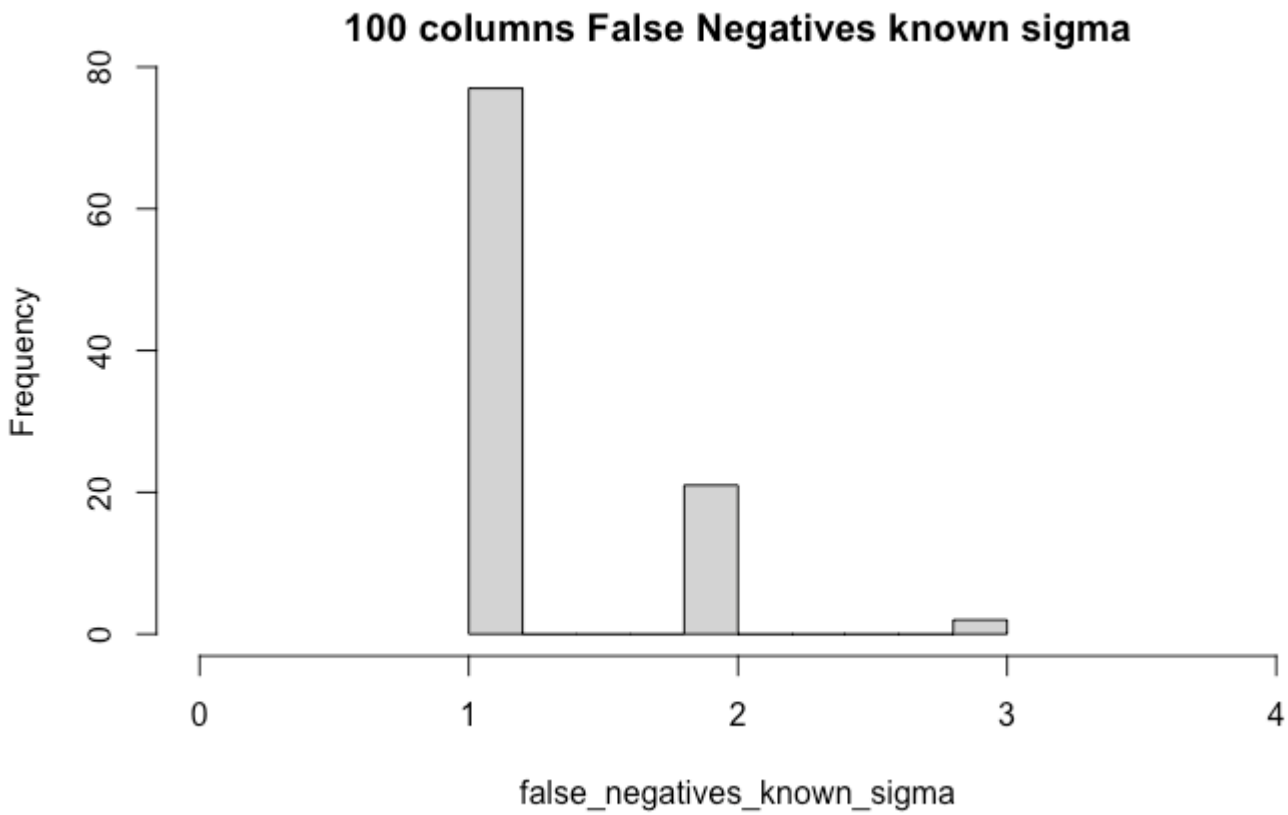

10 columns False Negatives known sigma
Frequency
0
20
40
60
0
2
4
6
8
10
false_negatives_known_sigma

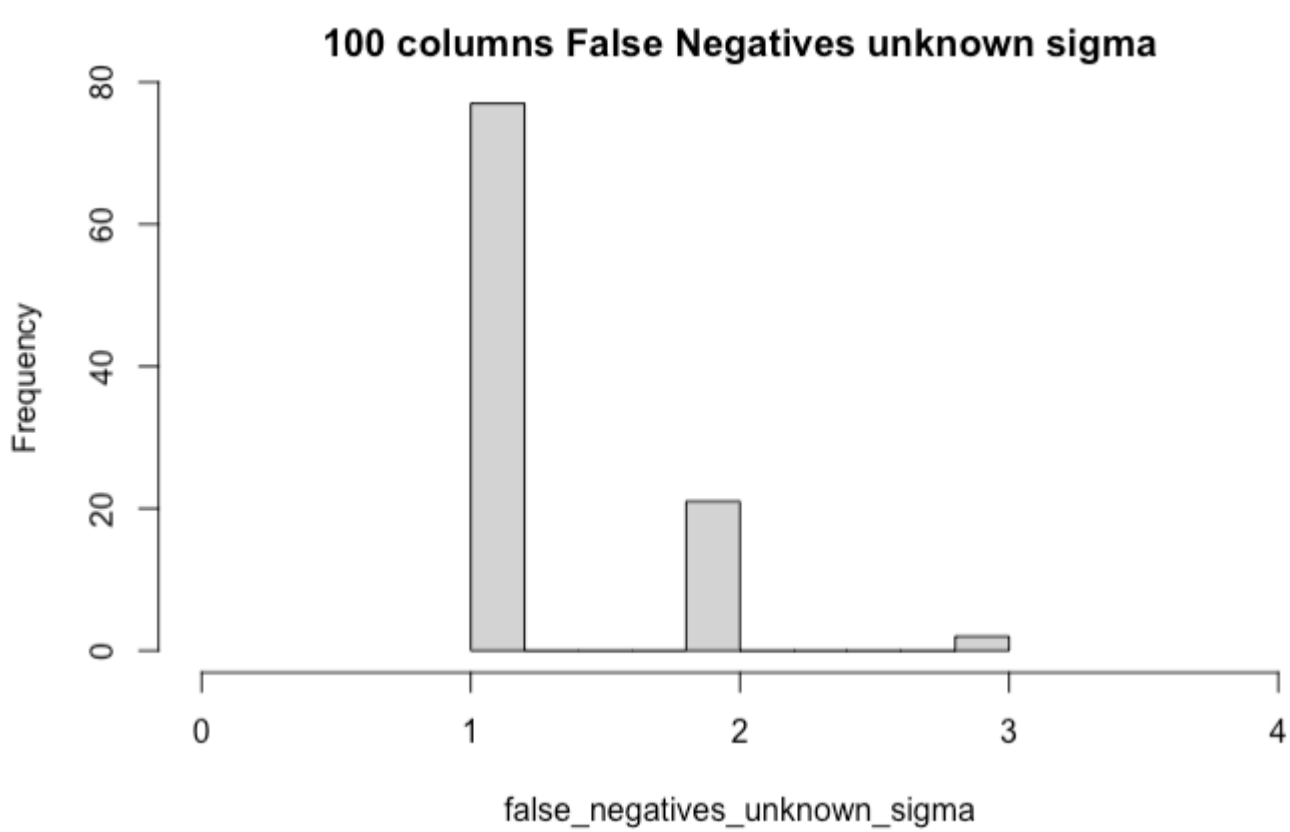
a) 100 columns





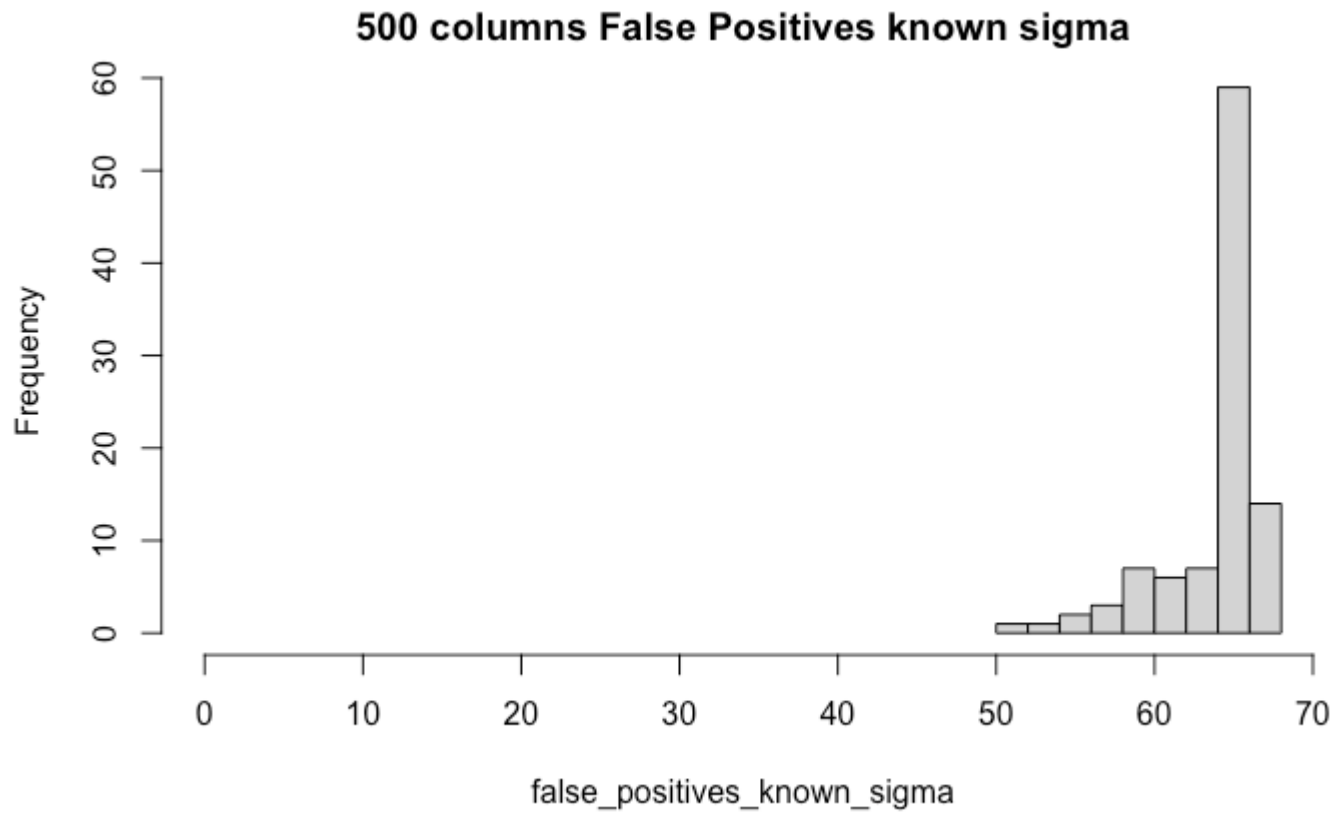

0
20
40
60
80
100
false_positives_known_sigma

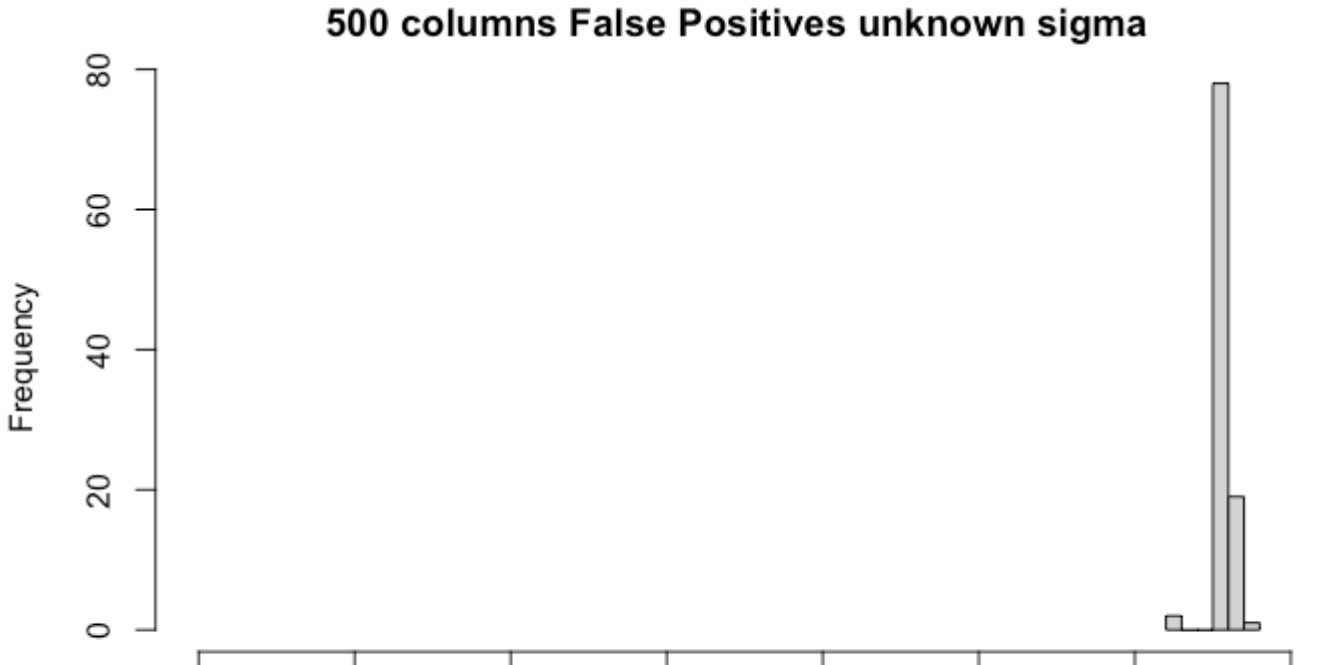

100 columns False Negatives known sigma
Frequency
0
20
40
60
80
0
1
2
3
4
false_negatives_known_sigma



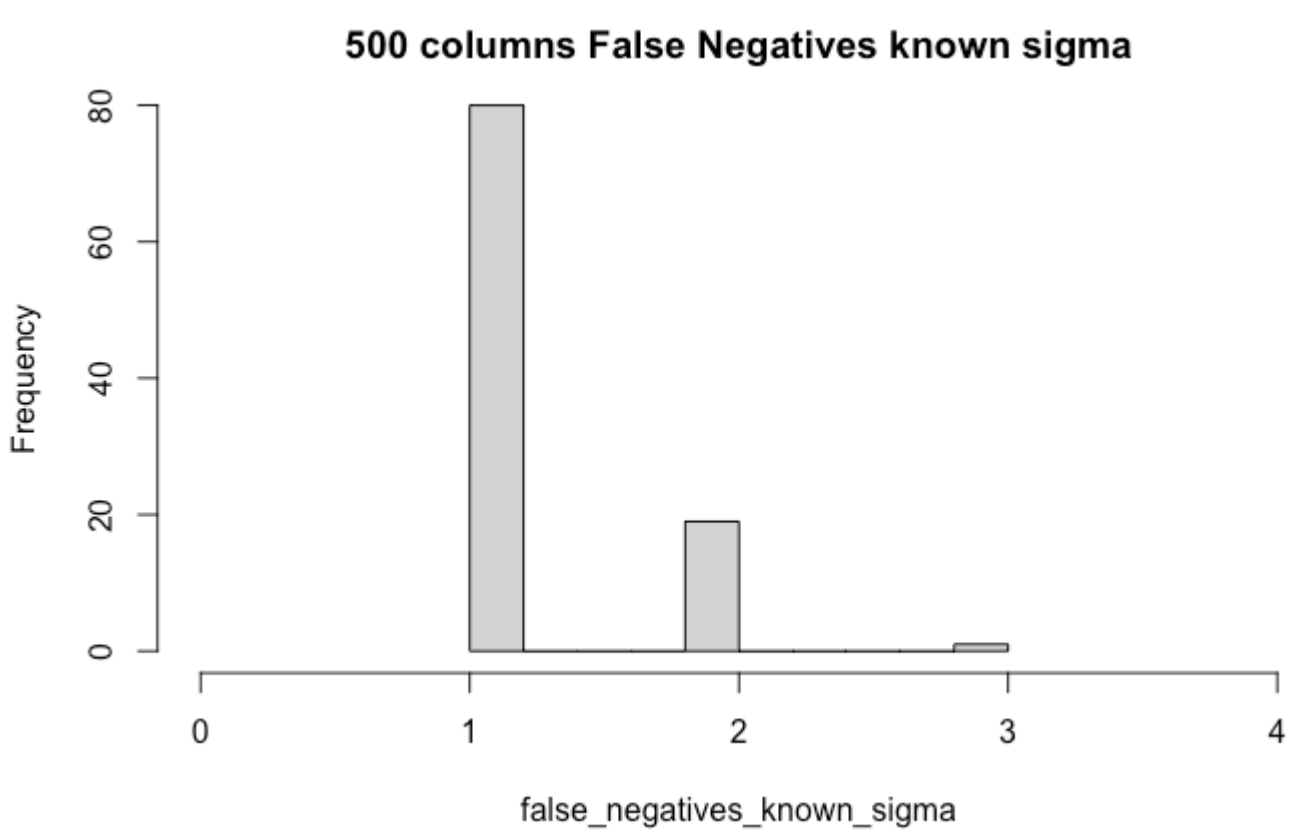
Here model chooses around 17 features, almost always the significant ones.

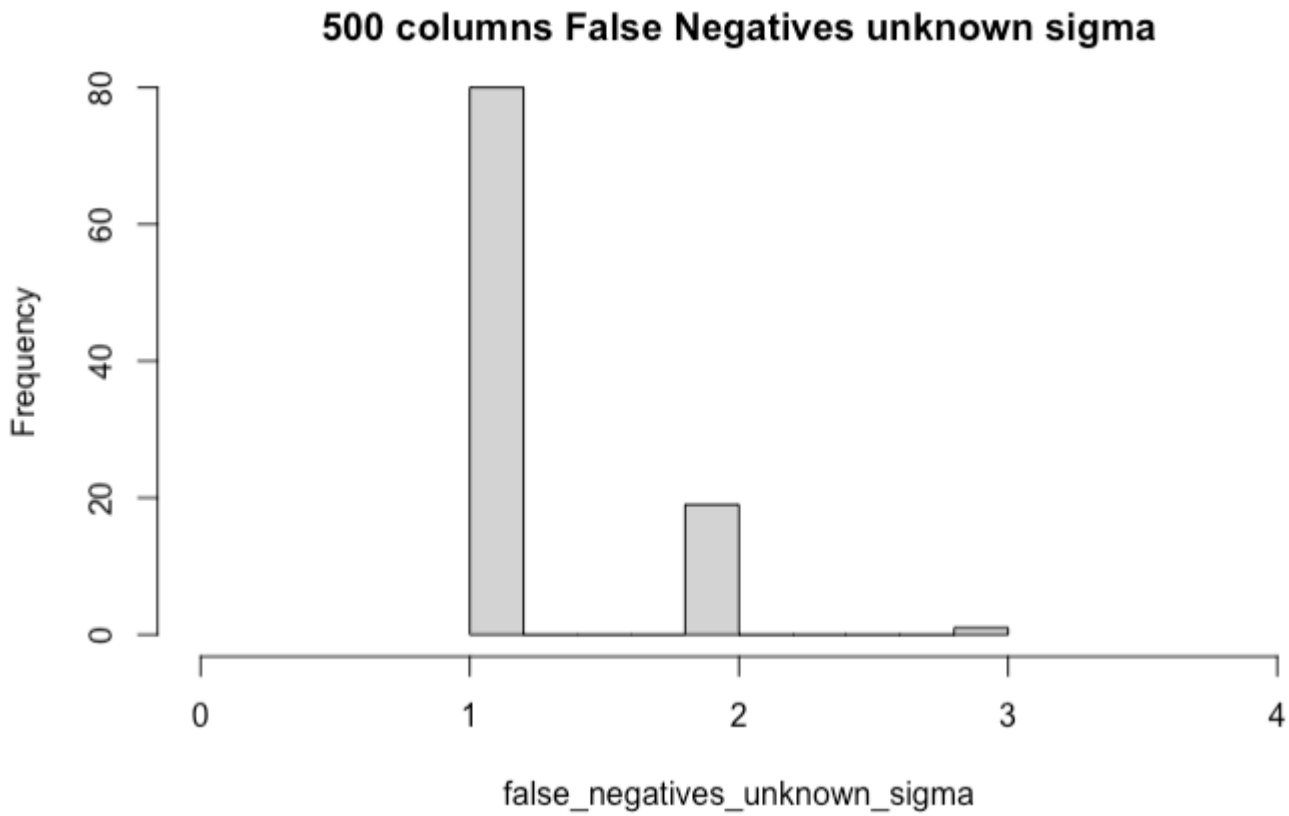
a) 500 columns





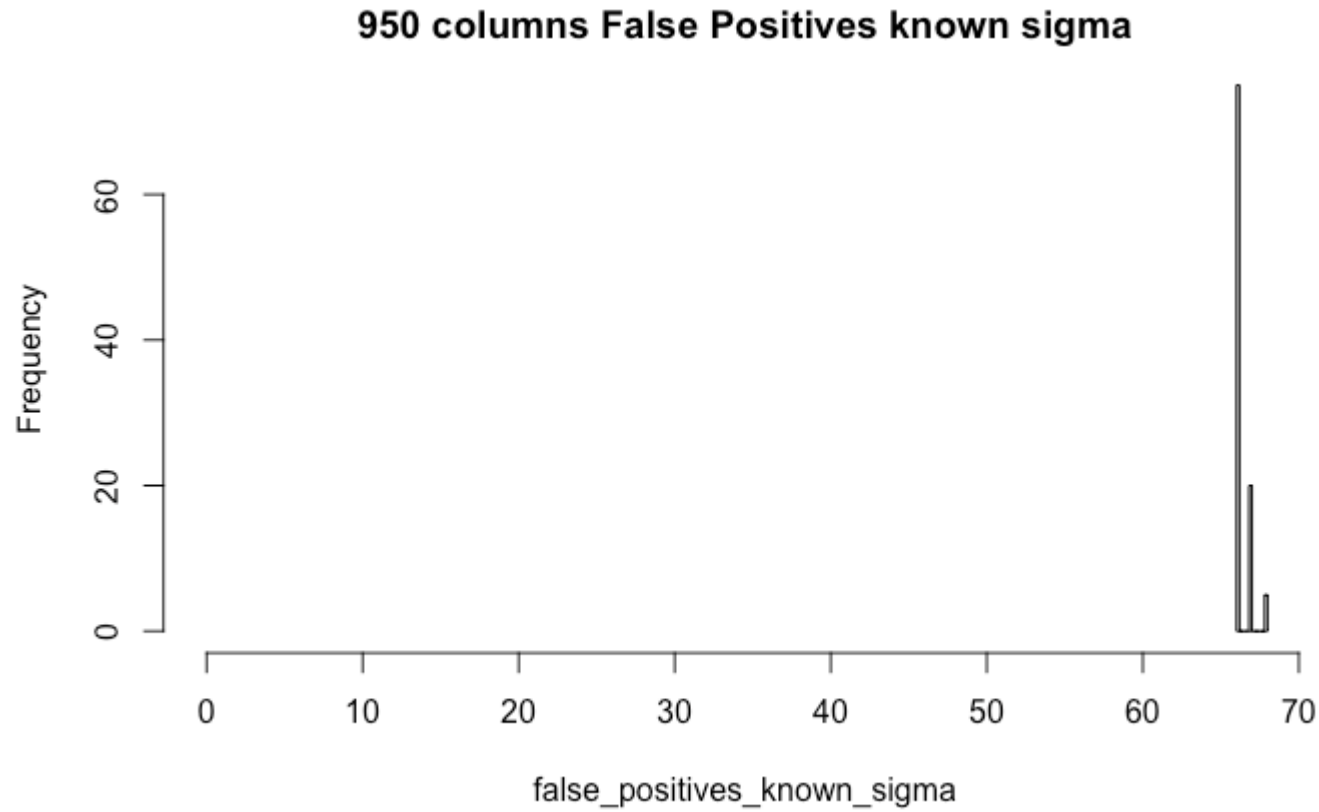
false_positives_unknown_sigma

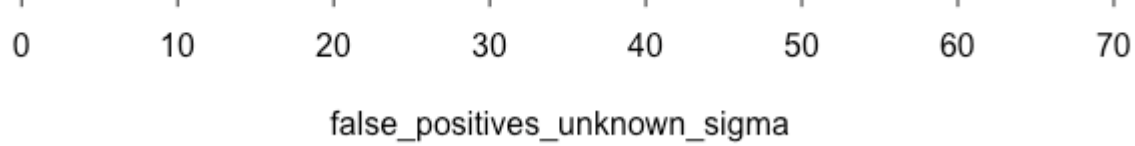

500 columns False Negatives known sigma
Frequency
0
20
40
60
80
0
1
2
3
4
false_negatives_known_sigma

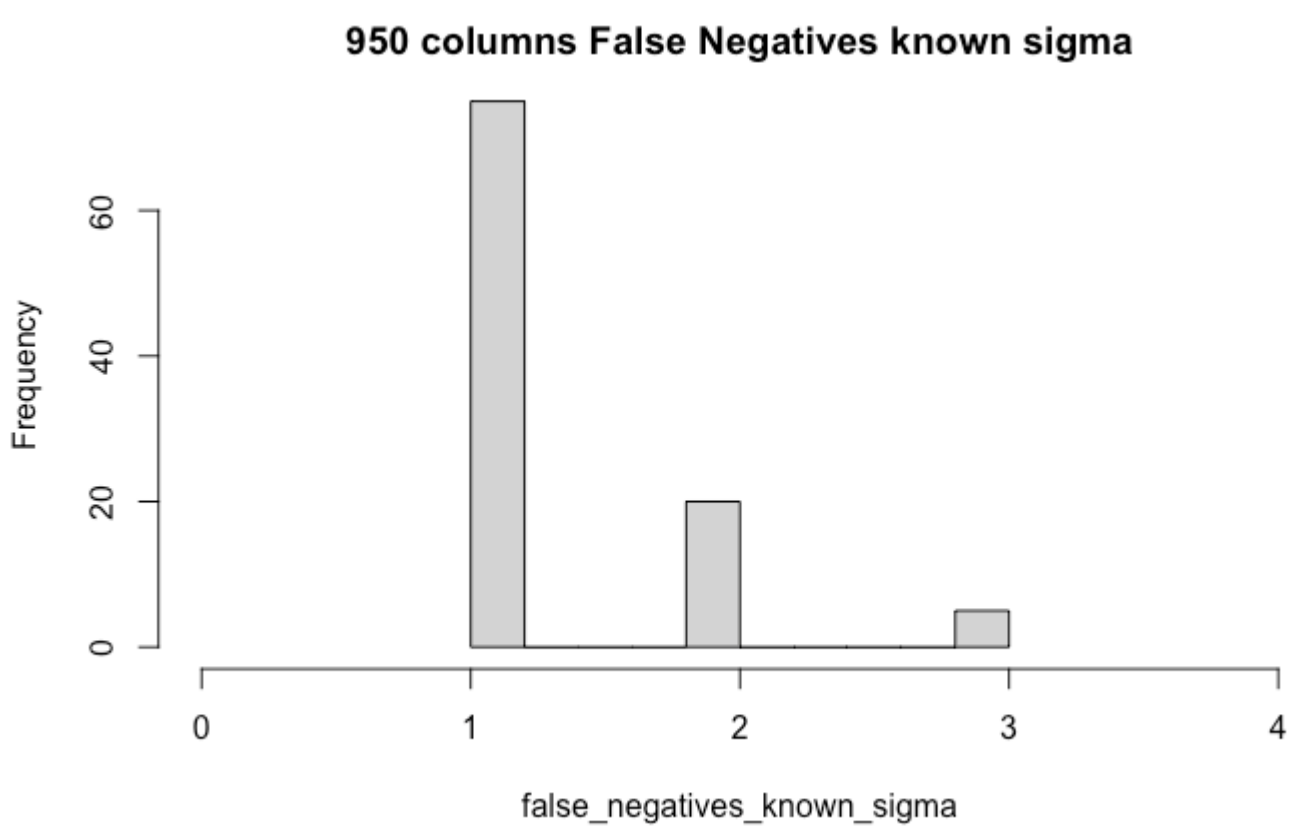


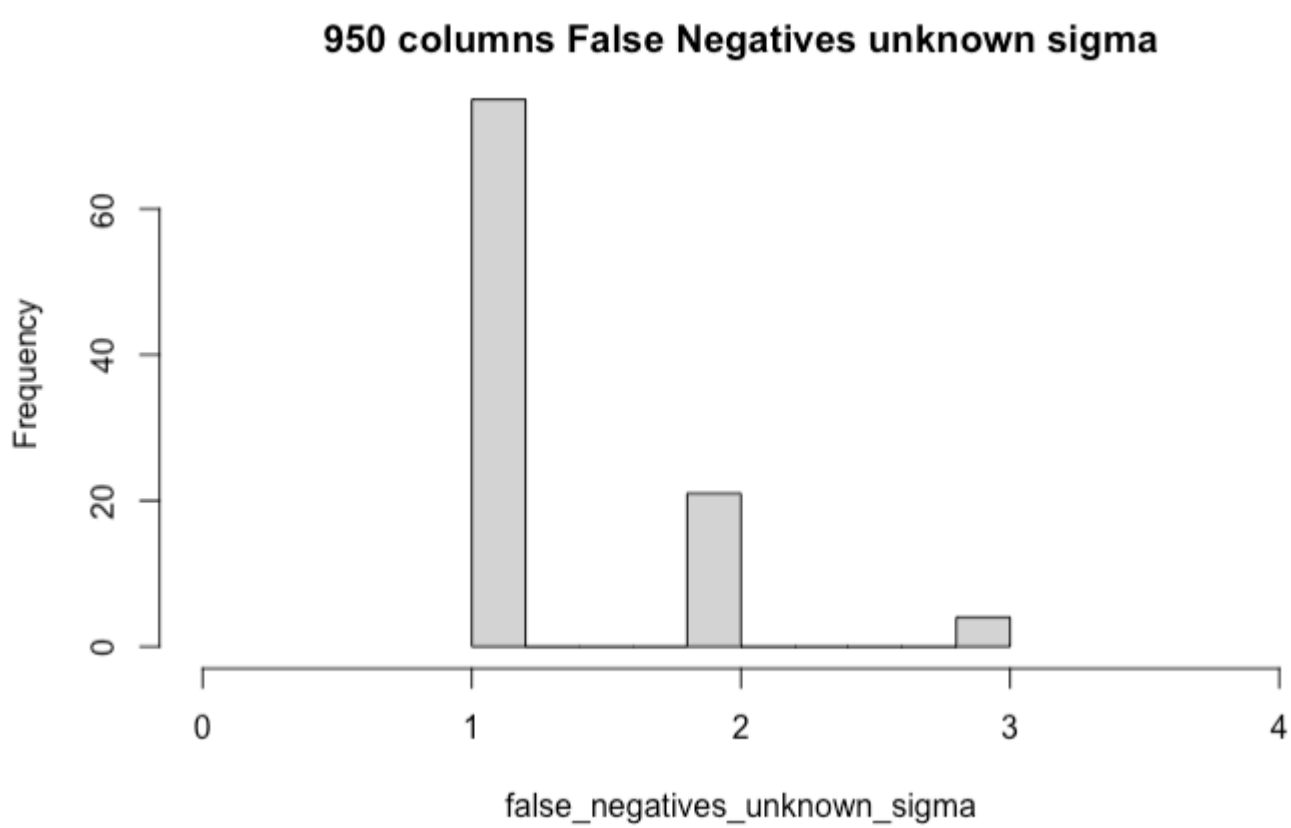
For unknown sigma the std is visibly bigger.
But we still fail to discover all of the significant features.

a) 950 columns




0
10
20
30
40
50
60
70
false_positives_unknown_sigma


950 columns False Negatives known sigma
Frequency
0
20
40
60
0
1
2
3
4
false_negatives_known_sigma



For bigger models the number of false positives grows quite a lot. But we can see clear benefit of using AIC. For example for model with 500 or 950 columns we were able to reduce their number to only 65-70 on average. On the other hand we sometimes discard up to 3 out of 5 significant variables.

Task 2

In this task we will use different criterions like BIC, AIC, RIC, mBIC i mBIC2 to identify important covariates when the search is performed over the data base date consisting of:

i) 20 first variables

ii) 100 first variables

iii) 500 first variables

iv) all 950 variables

a)

Report the number of false and true discoveries and the square error of the estimation of the vector of expected values of Y: RSS = $||X\beta - \hat{Y}||^2$

b)

Repeat point a) 100 times and report the estimated power, FDR and mean squared error of the estimation of expected values of Y for all criteria considered above. Critically summarize the results.

I will show mean results for tests for different number of columns as it's the most interesting part of this task.

20 columns

Criterion	RSS	#False Positives	#True Positives	FDR	POWER
BIC	998.87	0.77	2.45	0.239	0.49
AIC	983.6	3.34	3.72	0.47	7.44
RIC	996.7	0.86	2.67	0.24	0.53
mBIC	1005	0.48	1.64	0.22	0.328
mBIC2	1001.2	0.71	1.86	0.27	0.372

- AIC was the best here (winning with the highest Power), on average it was able to identify 3.72 out of 5 significant columns, but at a cost of higher number of False Positives.

100 columns

Criterion	RSS	#False Positives	#True Positives	FDR	POWER
BIC	1001.57	1.39	2.39	0.367	0.478
AIC	947.874	15.92	3.77	0.808	0.754
RIC	1007.139	0.57	1.76	0.2446	0.352
mBIC	991.098	0.27	0.91	0.2288	0.182
mBIC2	988.86183	0.34	1.02	0.2499	0.204

- AIC still has the greatest power, but the average number of False Positives grew even more. In my opinion BIC looks much better and more stable due to low number of False Positives.

500 columns

Criterion	RSS	#False Positives	#True Positives	FDR	POWER
BIC	959.49	4.91	2.54	0.659	0.508
AIC	777.1632	66.28	3.72	0.9468	0.744
RIC	956.4986	0.58	1.17	0.33142	0.234
mBIC	712.502	0.12	0.48	0.1999	0.096
mBIC2	711.6984	0.17	0.48	0.2615	0.096

- AIC is still the winner in terms of Power, but its number of False Positives is clearly outstanding now.

- mBIC and mBIC2 have the lowest FDR, but they on average discover only 0.5 column.

950 columns

Criterion	RSS	#False Positives	#True Positives	FDR	POWER
BIC	934.690	9.49	2.52	0.790	0.504
AIC	787.997	66.21	3.79	0.9458	0.758
RIC	789.351	0.53	0.83	0.3897	0.166
mBIC	469.97	0.15	0.4	0.272	0.08
mBIC2	468.75	0.22	0.4	0.35483	0.08

- As AIC continues to have the biggest Power and making the biggest number of False Positives, mBIC and mBIC2 somehow managed to get the lowest RSS despite having quite low number of average True discoveries.

It's worth mentioning that Power for AIC and BIC doesn't depend on the number of the columns as we can see in their formulas. This is exactly what we observed in the experiment.

Other three criterions depend on the number of columns that's why the Power is getting lower and lower.

Task 3

Data set realdata.Rdata contains the expression levels of 3221 genes for 210 individuals.

a)

Randomly select 30 individuals for the test.

b)

Use the training set (180 individuals) to construct the regression model explaining the expression level of gene 1 (first column in the data set) as the function of expression levels of other genes. Select explanatory variables using AIC, BIC, mBIC and mBIC2.
Test the accuracy of your model predictions on the test set. Which criterion yielded the best prediction results.

criterion	MSE model without intercept	MSE model with intercept
AIC	14.90958	0.09592228
BIC	9.213101	0.03969493
mBIC	37.55673	0.03516365
mBIC2	20.90261	0.03272206

I used `fast_forward` method to build models according to different criterions. When I used only ˆβ vector without intercept the best was clearly BIC.
But when I allowed for intercept the errors became very small and mBIC2 yielded the smallest MSE.