# Midterm report

Statistical Learning

**Adin Pablo Bartoli**

Midterm assessment of the course
Statistical Learning 2024/25 (MSIAM)



University Grenoble Alpes
Grenoble-INP ENSIMAG
Date of submission: 18 November 2024
Professor: IOUDITSKI Anatoli

# Contents

**1 Introduction** **2**

**2 Exercise E.1** **2**
2.1 First part . . . 2
2.1.1 QUESTION 1 . . . 2
2.1.2 QUESTION 2 . . . 3
2.1.3 QUESTION 3 . . . 3
2.1.4 QUESTION 4 . . . 4
2.1.5 QUESTION 5 . . . 5
2.2 Second part . . . 7
2.2.1 QUESTION 1 . . . 7
2.2.2 QUESTION 2 . . . 7
2.2.3 QUESTION 3 . . . 7
2.2.4 OTHER METHODS . . . 9

**3 Exercise E.2** **12**
3.1 QUESTION 1 . . . 12
3.2 QUESTION 2 . . . 12
3.3 QUESTION 3 . . . 12

**4 Exercise E.3 (Dantzig Selector)** **20**
4.1 QUESTION 1 . . . 20
4.2 QUESTION 2 . . . 20
4.3 QUESTION 3 . . . 21
4.4 QUESTION 4 . . . 22

# 1 Introduction

All the code can be found in the code folder of this assignment.

# 2 Exercise E.1

## 2.1 First part

### 2.1.1 QUESTION 1

First, in order to use the tools presented in the lectures, we setup the model in matrix form. Given $J = \{i_1, i_2, ..., i_s\}$, where for each segment $[i_l, i_{l+1})$ (where $i_{s+1} = n + 1$) $f(t)$ is constant with value $c_l$ on each segment. We want to have a model of the form $f = X\theta$. We define:

- $X$ as a $n \times n$-matrix with each column representing an indicator vector for a segment (taking values of 1 on a specific segment and 0 otherwise):

$$X_{j,l} = \begin{cases} 1 & \text{if } i_l \leq j < i_{l+1} \\ 0 & \text{otherwise} \end{cases}$$

- $\theta = (c_1, ..., c_s)^T$ as the vector of constant for each segment.

We want to find $\hat{\theta}$, which is the least-squares estimator of $\theta^*$. We have that:

$$\hat{\theta} = \arg\min_{\theta} \|y - X\theta\|_2^2 = (X^T X)^{-1} X^T y.$$

We can now compute $X^T X$ and $X^T y$:

- Each entry of $X^T X$ represents the inner product of indicator vectors for different segments. Thus, $X^T X$ is a diagonal matrix where the $(l, l)$-entry is the number of points in the segment $[i_l, i_{l+1})$ which is $i_{l+1} - i_l$:

$$X^T X = diag(i_2 - i_1, i_3 - i_2, ...., i_{s+1} - i_s)$$

- We have $(X^T y)_l = \sum_{j=i_l}^{i_{l+1}-1} y_j$, thus the vector $X^T y$ contains the sum of $y_i$ values within each segment $[i_l, i_{l+1})$.
- As $X^T X$ is diagonal is inverse is trivial.
- We thus find $\hat{\theta}_l = \frac{1}{i_{l+1} - i_l} \sum_{j=i_l}^{i_{l+1}-1} y_j$, which are the average values in the segment $[i_l, i_{l+1})$.

Now that we have established $\hat{\theta}$, we can express the least-squares prediction $\hat{f}_s = X\hat{\theta}$ as:

$$\hat{f}_s = \sum \bar{y}_{i_l, i_{l+1}} E_{i_l}^{i_{l+1}}$$

We want the mean of each segment to only contribute to its segment of $f$ in $\hat{f}$. This is why the terms $E_{i_l}^{i_{l+1}}$ (indicators of segment $l$) are in the product.

### 2.1.2 QUESTION 2

We know that $\hat{\theta}^{BIC} \implies \hat{f}^{BIC}$ as $\hat{f}^{BIC} = X\hat{\theta}^{BIC}$. So the question is just about observing that we optimize on the same sets of values. Indeed, instead of searching over all possible subsets $J$ of varying sizes, we can first find the minimum error $\delta_n^2(s)$ for each $s$, then add the penalty term $\tau^2 s$, and choose the $s$ that minimizes the expressions.

We have:

$$J = \bigcup_{s=0}^{n} J_s$$

and consequently:

$$\arg\min J = \arg\min \bigcup_{s=0}^{n} J_s = \arg\min \left( \bigcup_{s=0}^{n} \arg\min J_s \right)$$

As the penalty function is constant when $s$ is fixed the minimum argument over $J_s$ yields the best fit. Then we minimize over the $s$ values. Thus we do have that the final estimator $\hat{f}^{BIC}$ may be found as the minimizer of the penalized least-squares criterion $\delta_n^2(s)+\tau^2 s$ among $\hat{f}_{\hat{J}_S}$ for $s = 0, ..., n$.

### 2.1.3 QUESTION 3

- $\delta_n(s)$ represents the minimum possible error for a model with exactly $s$ jumps, where we seek to partition the data $y_i$ into $s+1$ segments.
- $R^2(k, j)$ sums the squared deviations of $y_i$ from its segment mean.
- $\hat{J}_s$ is the set of indices $\{i_1, ..., i_s\}$ that gives this minimum error $\delta_n(s)$

As shown in the previous questions, each segment is approximated with its mean $\bar{y}_{i_l, i_{l+1}}$. Then the squared error for this segment is $R^2(i_l, i_{l+1})$. The total squared error for a fixed set of jump points $\{i_1, ..., i_s\}$ is the sum of squared errors over the segments:

$$||y - f_J||_2^2 = N^2(i_1) + \sum_{l=1}^{s} R^2(i_l, i_{l+1})$$

where $N^2(i_1)$ represents the sum of squares before the first jump $i_1$ and $\sum_{l=1}^{s} R^2(i_l, i_{l+1})$ represents the residuals sum of squares within each interval $[i_l, i_{l+1})$.

To find the optimal configuration of jump points, we must minimize the total squared error over all choices of $\{i_1, ..., i_s\}$, which is exactly as looking for the minimizer set of the quantity $N^2(i_1) + \sum_{l=1}^{s} R^2(i_l, i_{l+1})$.

We conclude that minimizing the total squared error over all possible configurations of $s$ jump points leads to the expressions for $\delta_n(s)$ and $\hat{J}_s$:

$$\hat{J}_s \in \arg \min_{1 \le i_1 \le ... \le i_s \le n} N^2(i_1) + \sum_{l=1}^{s} R^2(i_l, i_{l+1})$$

and:

$$\delta_n(s) = \min_{1 \le i_1 \le \ldots \le i_s \le n} N^2(i_1) + \sum_{l=1}^{s} R^2(i_l, i_{l+1})$$

### 2.1.4 QUESTION 4

We aim to prove that the minimum error with $s$ jumps can be expressed by the errors of $s-1$ jumps. If the $s$-th jump occurs at position $i$, then the total error is the sum of :

- The min error with $s-1$ jumps up to $i-1$: $\delta^2_{i-1}(s-1)$
- The residual error from $i$ to $n+1$: $R^2(i, n+1)$

To find $\delta^2_n(s)$ we minimize this total error over all possible positions of $i$ from $s$ to $n$.

We establish the result by induction on $s$.

**Base case:** $s = 2$

$$\begin{aligned}
\delta^2_n(2) &= \min_{i=2,\ldots,n} \left\{\delta^2_{i-1}(1) + R^2(i, n+1)\right\} \\
&= \min_{i=2,\ldots,n} \left\{N^2(i-1) + R^2(i-1, i) + R^2(i, n+1)\right\} \\
&= \min_{i=2,\ldots,n} \left\{N^2(i-1) + R^2(i-1, n+1)\right\}
\end{aligned}$$

Which indeed is the minimum error for 2 jumps.

**Induction Hypothesis**

We assume that :

$$\delta^2_n(s-1) = \min_{i=2,\ldots,n} \left\{\delta^2_{i-1}(s-2) + R^2(i, n+1)\right\}$$

**Induction**

$$\begin{aligned}
\min_{i=s,\ldots,n} \left\{\delta^2_{i-1}(s-1) + R^2(i, n+1)\right\} &= \min_{i=2,\ldots,n} \left\{\min_{j=s-1,\ldots,i-1} \left\{\delta^2_{j-1}(s-2) + R^2(j, i)\right\} + R^2(i, n+1)\right\} \\
&= \min_{i=s,\ldots,n} \left\{\delta^2_{l-1}(s-2) + R^2(l, i+1) + R^2(i, n+1)\right\} \\
&= \min_{i=s,\ldots,n} \left\{\delta^2_{l-1}(s-2) + R^2(l, n+1)\right\}
\end{aligned}$$

where $l$ is optimal for the placement of the $s-1$ jumps. Iterating this result $s-2$ times we indeed find the enounced recursive formula.

### 2.1.5 QUESTION 5

- Complexity of $N^2(k)$ for $1 \leq k \leq j \leq n+1$: We only go once through the data, thus the complexity is $\mathcal{O}(n)$.
- Complexity of $R^2(k,j)$ for $1 \leq k \leq j \leq n+1$ : We compute the mean for each pair of segments $(k,j)$ with $k \leq j$ which means that we have to go through $\frac{n(n-1)}{2}$ pairs, using cumulative sums for $\sum_m^i y_m$ and $\sum_m^i y_m^2$ with a cost of $\mathcal{O}(n)$, as shown just above, we have a cost $\mathcal{O}(1)$ for each pair, thus $\mathcal{O}(n^2)$ for all pairs.

The following algorithm computes the minimum error $\delta_n^2(s)$ achievable with $s$ jumps for data of size $n$, along with the corresponding jump positions $J_s$. It uses a dynamic programming approach with cumulative sums to optimize performance.

**Input:** Observations $y_1, y_2, \ldots, y_n$; Penalty parameter $\tau^2$; Maximum number of jumps $s_{\max}$.
**Output:** $J_s$: Optimal jump points for the best number of jumps $s$.
`/* Step 1: Precompute` $N_2(k)$ `and` $R_2(k,j)$ `*/`
Compute cumulative sums:

$$S_y(k) = \sum_{m=1}^{k} y_m, \quad S_{y^2}(k) = \sum_{m=1}^{k} y_m^2 \quad \text{for } k = 1, \ldots, n.$$

Use the cumulative sums to compute:

$$N_2(k) = \sum_{m=1}^{k-1} y_m^2, \quad R_2(k,j) = S_{y^2}(j-1) - S_{y^2}(k-1) - \frac{(S_y(j-1) - S_y(k-1))^2}{j-k}.$$

`/* Step 2: Initialize for` $s = 1$ `*/`
For $k = 1, \ldots, n$:

$$\delta[k, 1] = N_2(k) + R_2(k, n+1).$$

`/* Step 3: Dynamic programming for` $s \geq 2$ `*/`
**for** $s = 2$ **to** $s_{\max}$ **do**
  **for** $k = s$ **to** $n$ **do**
    Compute:

$$\delta[k,s] = \min_{i=s,\ldots,k} \{\delta[i-1, s-1] + R_2(i, k+1)\}.$$

    Store the minimizing $i$ in $J[k,s]$.
  **end**
**end**
`/* Step 4: Compute BIC for each` $s$ `*/`
For $s = 1, \ldots, s_{\max}$:

$$\text{BIC}(s) = \delta[n,s] + \tau^2 \cdot s.$$

`/* Step 5: Select the best` $s$ `and recover jumps */`
Find the optimal $s$ as:

$$s_{\text{best}} = \arg\min_s \text{BIC}(s).$$

Use traceback to recover the jump points $J_{s_{\text{best}}}$.

**Algorithm 1:** Dynamic Programming Algorithm for $\delta_n^2(s)$ and $J_s$

**Complexity:**

- Precomputing $R_2(i,j)$ for all pairs: $O(n^2)$,
- Dynamic programming to compute $\delta[k,s]$: $O(n^3)$,
- Traceback to recover $J_s$: $O(n)$.

Overall complexity: $O(n^3)$.

## 2.2 Second part

### 2.2.1 QUESTION 1

After writing the subroutine, we simulate different kinds of data by varying the parameters as shown in figure 1. Directly from these graphs, we observe a large spectrum of data shapes, with a strong dependence on the parameters. In figure 2 we observe the same graphics with true values of function $f$ without the noise and with the jump spots in red. These data are the data with which we test the algorithm from section 1.

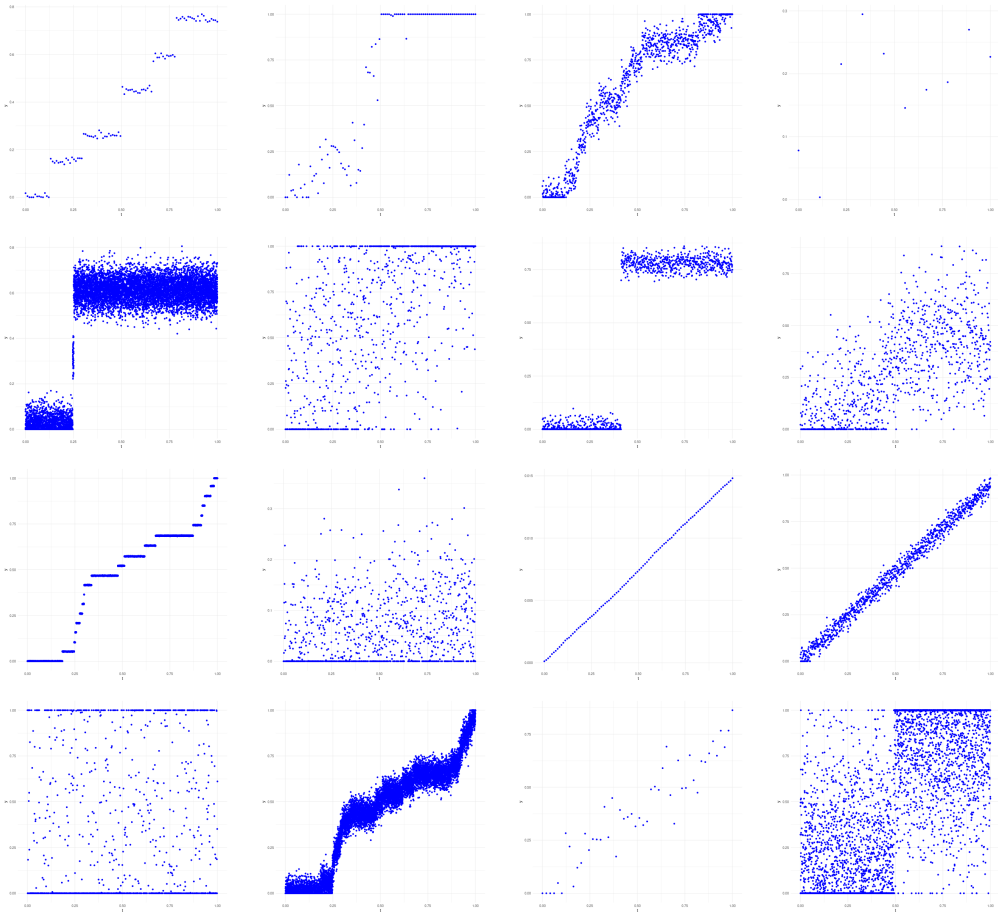

**Table 1:** Data simulation with various parameters.

### 2.2.2 QUESTION 2

Implementation of the functions and the routine for selection of $\hat{J}_s$, are in the EXERCISE2.R file.

### 2.2.3 QUESTION 3

**Remark:** There is a problem on the implementation of the algorithm, on how it gives more credits to the jumps at the late data points. I have not had time to fix this problem, but I reckon the later

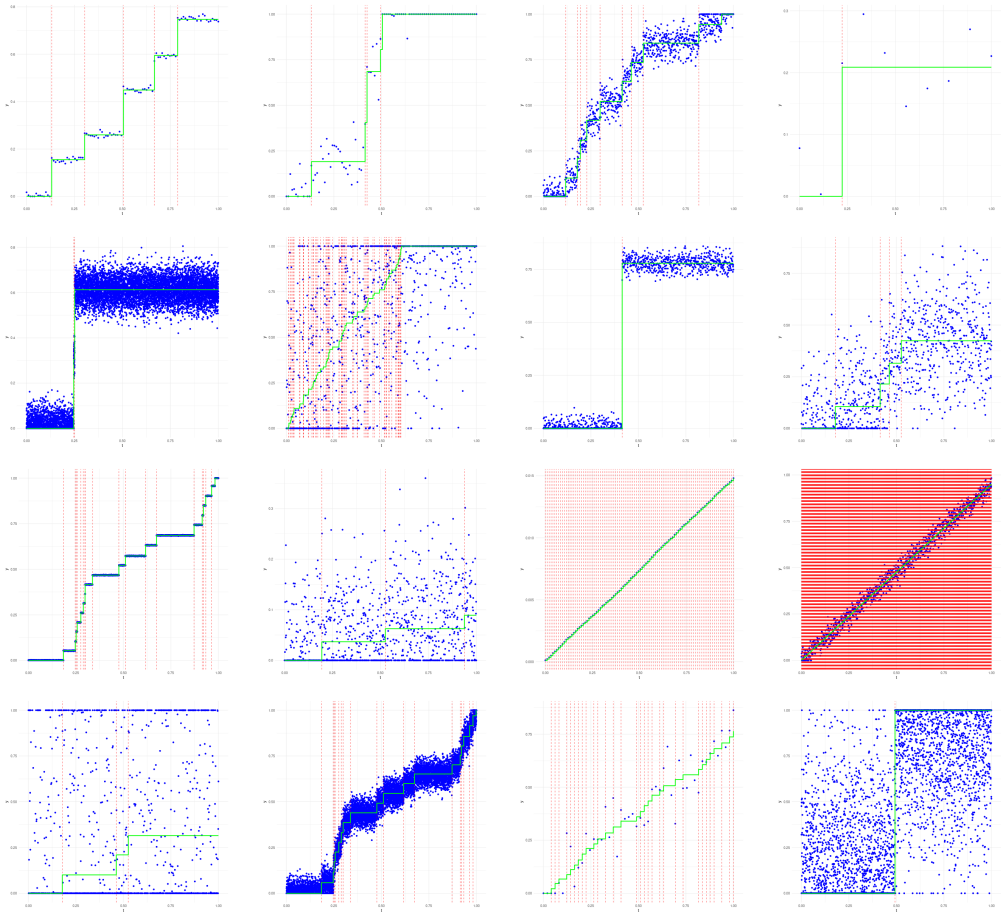

**Table 2:** Data simulation with various parameters, jump spot (in dashed red) and true value of $f$ in green.

interpretation of the graphs (which are general) should be altered by this fact. Also some jumps happen negatively (which is impossible for the true function), I should have corrected that.

We can now run experiments on the previously simulated data. We show the results for different choices of $\tau$. First in figure 8 we use $\tau = \log(n)$, which is a strong penalty, which should hence work well when the jumps are rare. Then we try a milder penalty in figure 4 which is $\tau = \log(n)/n$, which should allow for many more jumps.

I also wanted to try something for fixing $\tau$ by taking into account the volatility present in the data. Assuming the noise in the data is within a certain range with high probability $\mathbb{P}(\xi \in [a, b]) = 0.99$ we can evaluate the variability inside of the data and fix $\tau$ proportionally. We now have $\tau = var(y)/mean(y)$. We can observe the results in figure 5, when the noise is contained (not to large) the results are better than with the two other metrics for $\tau$, and the red step-wise function fits better to the data.

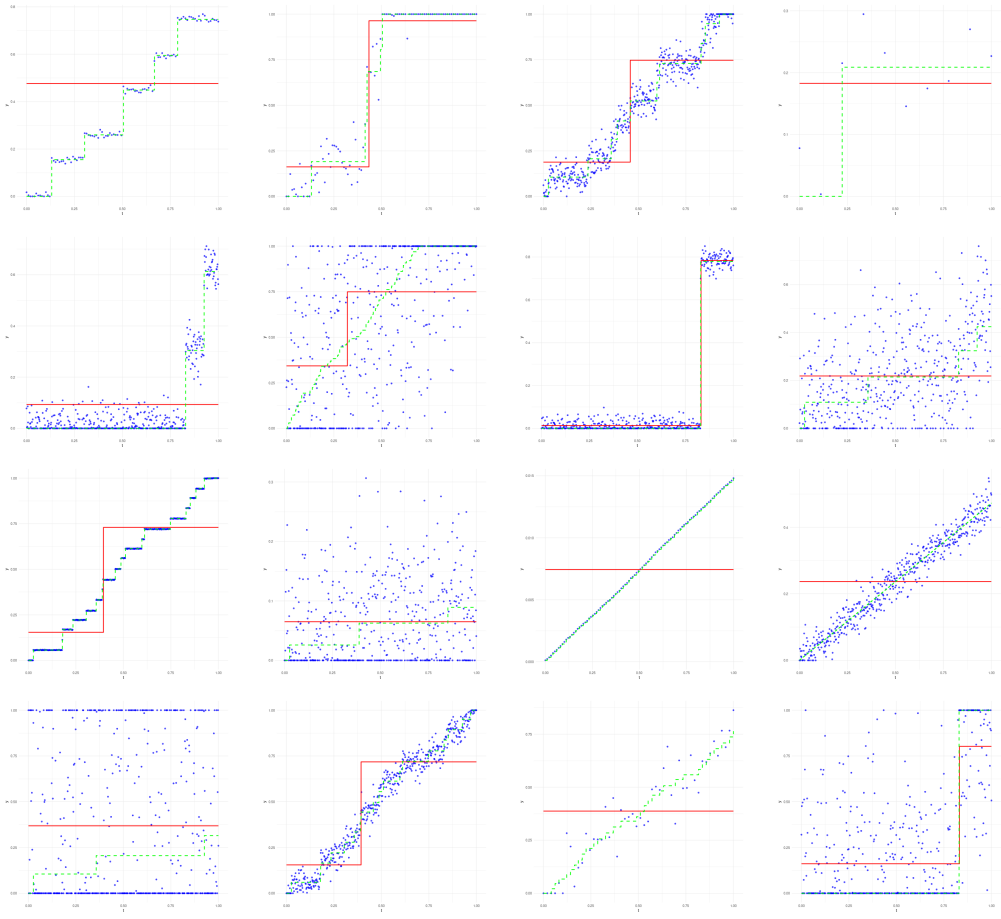

**Table 3:** Prediction of $f$ (in red) using $\tau = \log(n)$, true value in green.

### 2.2.4 OTHER METHODS

I think that when facing this problem, we could try some other approaches such as clustering on the $x, y$ position of the points and learning-based methods.

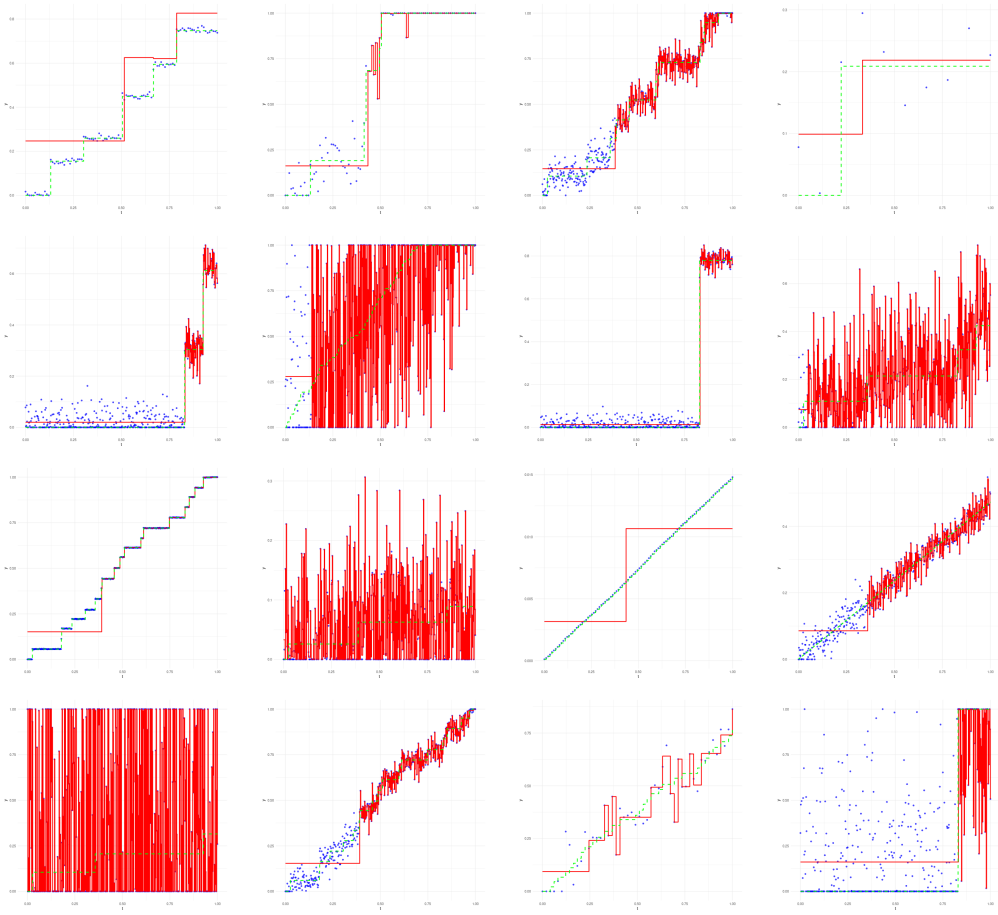

**Table 4:** Prediction of $f$ (in red) using $\tau = \log(n)/n$, true value in green.

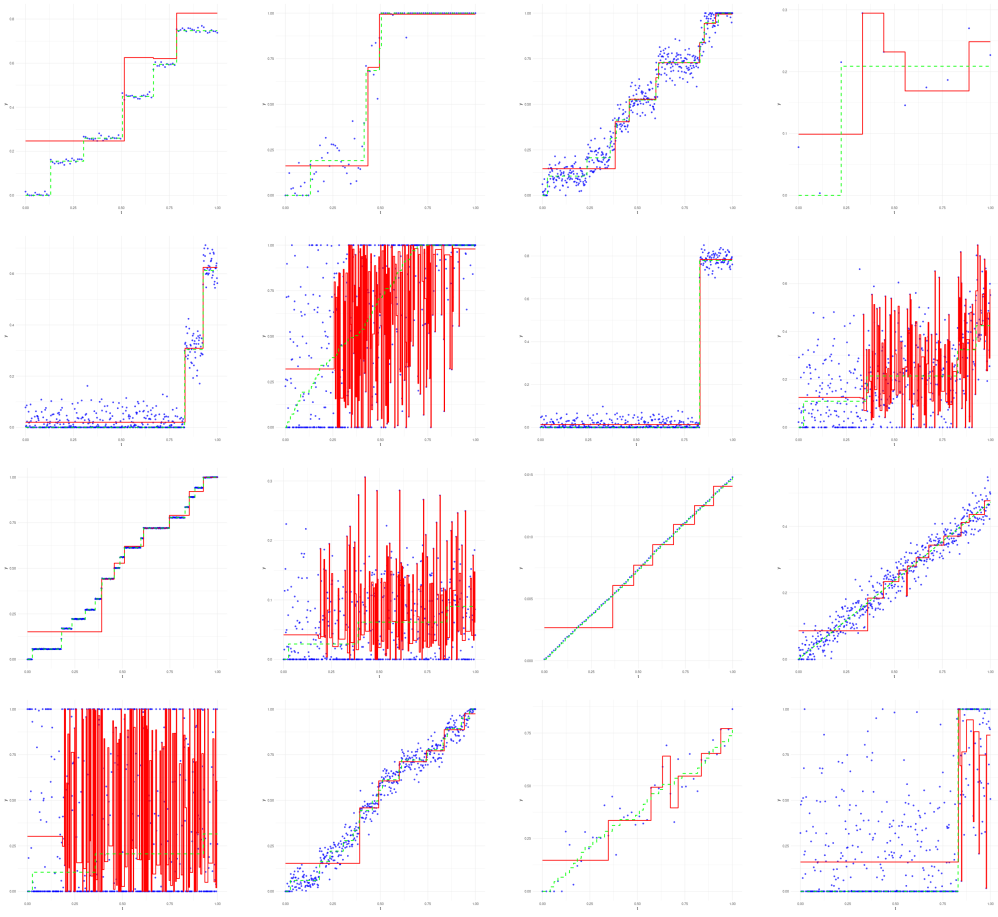

**Table 5:** Prediction of $f$ (in red) using $\tau = var(y)/\bar{y}$, true value in green

# 3 Exercise E.2

## 3.1 QUESTION 1

Our goal is to write a subroutine implementing the BIC estimator utilizing the forward-backward search algorithm. We define convergence in different manners:

- Close enough: If the BIC criterion improvement is smaller than a certain threshold.
- Fix point: If the algorithm is not moving $J_i = J_{i-1}$.
- Time limit convergence: If we reach a certain number of iterations.

Examples of these criteria utilization have been put in the R code, and it is also possible to play with the parameter $\tau$.

## 3.2 QUESTION 2

We want to express the observation model as a linear regression of form $y = X\theta + \xi$. In our case, the model is define as : $y_i = f(t_i) + \xi_i, i = 1, ..., n$ with $f(t) = \sum_{i=1}^{n} \theta_j^* \rho_j(t)$. For $f(t)$ to be expressed as a linear combination of basis functions we use the trigonometric basis $\rho_j(t)$(given in the subject).

For each $t_i, i = 1, ..., n$ we evaluate the basis function to form the design matrix $X$. Each row corresponds to a point $t_i$ and each column corresponds to a basis function $\rho_j(t)$. Thus we have $X_{ij} = \rho_j(t_i)$ and the model becomes $y = X\theta^* + \xi$.

The routine implemented in R includes:

- Random or deterministic (fixed) selection of the support set $J$ of $\theta$
- Random or deterministic (regular) grid of points $t_i \in [0, 1]$
- Different choices of parameters:
  - Variance $\sigma^2$.
  - $p$, $n$, and $s$ of the random signal.
  - The value of the norm $\|\theta^*\|_2$ of the signal.
  - The value $r$ of the smallest amplitude $|\theta_j^*|, j \in J$.

## 3.3 QUESTION 3

We present results of the simulations, all done with R seed(123). All the graphics or tables present in this section can thus be generated by running the provided code.

First we build tables to analyze the results. Such large tables are not extremely convenient but necessary in order to allow the parameters to vary and better grasp the problem.

We begin with a first table where the penalty ($\tau$) is a constant (which hence does not depend on $n$).

| $n$ | $p$ | $s$ | $\tau$ | **selected_vars** | **correct_vars** | **bic_value** |
|---|---|---|---|---|---|---|
| 10 | 9 | 3 | 1 | 1 | 0 | -285.22 |
| 24 | 9 | 3 | 1 | 3 | 3 | -246.02 |
| 32 | 9 | 3 | 1 | 4 | 3 | -193.75 |
| 40 | 9 | 3 | 1 | 3 | 3 | -178.80 |
| 48 | 9 | 3 | 1 | 3 | 3 | -117.63 |
| 56 | 9 | 3 | 1 | 3 | 3 | -133.75 |
| 10 | 12 | 3 | 1 | 4 | 0 | -407.54 |
| 24 | 12 | 3 | 1 | 3 | 3 | -216.82 |
| 32 | 12 | 3 | 1 | 3 | 2 | -191.00 |
| 40 | 12 | 3 | 1 | 4 | 3 | -176.21 |
| 48 | 12 | 3 | 1 | 3 | 3 | -154.34 |
| 56 | 12 | 3 | 1 | 3 | 3 | -142.87 |
| 10 | 15 | 3 | 1 | 7 | 3 | -960.34 |
| 24 | 15 | 3 | 1 | 3 | 3 | -220.52 |
| 32 | 15 | 3 | 1 | 3 | 3 | -206.04 |
| 40 | 15 | 3 | 1 | 3 | 3 | -183.64 |
| 48 | 15 | 3 | 1 | 3 | 3 | -145.35 |
| 56 | 15 | 3 | 1 | 3 | 3 | -134.01 |
| 10 | 18 | 3 | 1 | 4 | 1 | -359.82 |
| 24 | 18 | 3 | 1 | 3 | 3 | -197.50 |
| 32 | 18 | 3 | 1 | 3 | 3 | -236.50 |
| 40 | 18 | 3 | 1 | 3 | 3 | -167.74 |
| 48 | 18 | 3 | 1 | 3 | 3 | -121.46 |
| 56 | 18 | 3 | 1 | 3 | 3 | -122.35 |
| 10 | 9 | 7 | 1 | 4 | 3 | -349.78 |
| 24 | 9 | 7 | 1 | 6 | 6 | -133.87 |
| 32 | 9 | 7 | 1 | 7 | 7 | -117.35 |
| 40 | 9 | 7 | 1 | 3 | 3 | 8.88 |
| 48 | 9 | 7 | 1 | 7 | 7 | -71.44 |
| 56 | 9 | 7 | 1 | 7 | 7 | -1.01 |
| 10 | 12 | 7 | 1 | 5 | 3 | -276.46 |
| 24 | 12 | 7 | 1 | 7 | 7 | -147.22 |
| 32 | 12 | 7 | 1 | 7 | 7 | -146.83 |
| 40 | 12 | 7 | 1 | 2 | 2 | 15.29 |
| 48 | 12 | 7 | 1 | 6 | 6 | -83.82 |
| 56 | 12 | 7 | 1 | 6 | 6 | -48.36 |
| 10 | 15 | 7 | 1 | 5 | 1 | -347.40 |
| 24 | 15 | 7 | 1 | 7 | 7 | -185.12 |
| 32 | 15 | 7 | 1 | 7 | 7 | -139.31 |
| 40 | 15 | 7 | 1 | 2 | 2 | -24.01 |
| 48 | 15 | 7 | 1 | 7 | 7 | -67.64 |
| 56 | 15 | 7 | 1 | 7 | 7 | -80.24 |
| 10 | 18 | 7 | 1 | 4 | 3 | -410.30 |

Continued on next page

| **n** | **p** | **s** | **tau** | **selected_vars** | **correct_vars** | **bic_value** |
|---|---|---|---|---|---|---|
| 24 | 18 | 7 | 1 | 7 | 7 | -166.12 |
| 32 | 18 | 7 | 1 | 7 | 6 | -227.29 |
| 40 | 18 | 7 | 1 | 6 | 6 | -92.29 |
| 48 | 18 | 7 | 1 | 6 | 6 | -111.60 |
| 56 | 18 | 7 | 1 | 7 | 7 | -90.38 |

**Table 6:** Summary of results with different values of $n$, $p$, $s$ and fixed $\tau$.

Observing the results in table 6, it is clear that the number of correct variables depends on whether $n > p$ or not. If $p > n$ there seems to be difficulty to identify the true variables.

The same setup is used in table 7 but with fixing different values of $\tau$ as a function of $n$, $\tau = f(n)$, and we look especially at the case $\tau = \log(n)$ which is the canonical choice for $\tau$. In practice, this canonical value provides a strong baseline but may not be optimal for all datasets, especially when dealing with high-dimensional data (i.e., $p > n$) or sparse signals.

| $n$ | $p$ | $s$ | $\tau$ | **selected_vars** | **correct_vars** | **bic_value** |
|---|---|---|---|---|---|---|
| 20 | 10 | 2 | 3.00 | 2.50 | 2.00 | -245.54 |
| 20 | 20 | 2 | 3.00 | 2.00 | 0.50 | -248.89 |
| 20 | 10 | 5 | 3.00 | 5.00 | 4.50 | -257.08 |
| 20 | 20 | 5 | 3.00 | 6.00 | 4.00 | -214.60 |
| 20 | 10 | 8 | 3.00 | 7.00 | 7.00 | -256.91 |
| 20 | 20 | 8 | 3.00 | 4.50 | 3.50 | -103.43 |
| 50 | 10 | 2 | 3.91 | 1.50 | 1.50 | -181.95 |
| 50 | 20 | 2 | 3.91 | 2.00 | 2.00 | -177.43 |
| 50 | 10 | 5 | 3.91 | 5.00 | 5.00 | -124.34 |
| 50 | 20 | 5 | 3.91 | 4.00 | 4.00 | -145.76 |
| 50 | 10 | 8 | 3.91 | 7.50 | 7.50 | -61.08 |
| 50 | 20 | 8 | 3.91 | 7.50 | 7.50 | -72.63 |

**Table 7:** Summary of results with $\tau = \log(n)$

In table 7 we observe better results when $\tau = \log(n)$ and $n > p$. We see that in comparison of to table 6, the number of selected variables seems to be closer to the real number of variables.

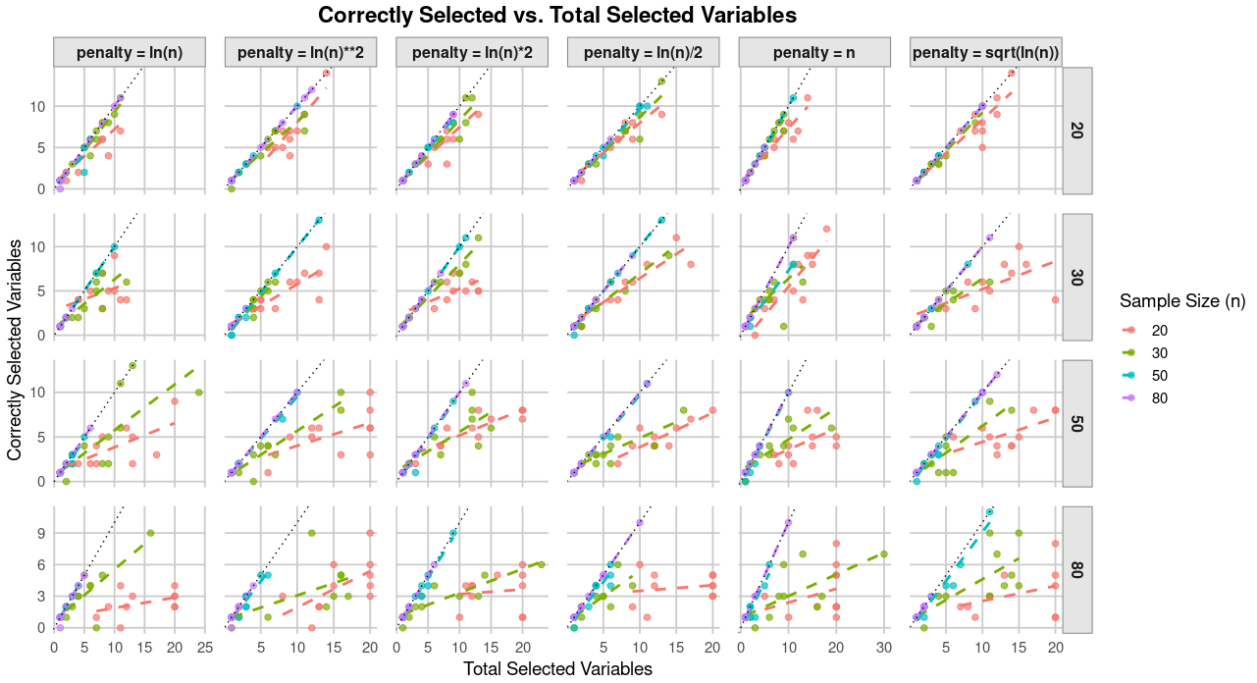


**Figure 1:** Success rate of selected variables for different functions $\tau$ and for different values of $p$ (each row) and sample sizes.

In figure 1 we observe the results of proportion of correctly predicted variables for different functions of $n$ for $\tau$. The closer the colored lines to the $f(x) = x$ line the more accurate the predictions. We observe that when the sample size ($n = 20$) is low and the dimensionality $p$ increases, the prediction starts to be wrong and far from reality.

In figure 2 we observe the same results as in figure 1 but presented in a different manner, and with random selection of $n$, $p$ and $s$. These graphs need to be interpreted in this way: the closer the curve to 1 the more accurately predicted the variables.

When dealing with high-dimensional settings where $p > n$ (i.e., the number of predictors is greater than the number of observations), selecting an appropriate penalty parameter $\tau$ for BIC can be challenging; the canonical BIC choice of $\tau$ may not be optimal in these settings because it can lead to over-penalization, failing to capture relevant predictors, especially when the true signal is sparse. Our goal is to adjust $\tau$ based on dimensionality. We can try to include the dimension of $p$ to fix $\tau$. This choice incorporates both the sample size $n$ and the dimensionality $p$. The $\log(p)$ term helps adjust for the larger model space, reducing the risk of over-penalization. We introduce a new criterion, the EBIC :

$$EBIC = n \ln(\frac{RSS}{n}) + s \ln(n) + 2\gamma s \log(p)$$

This criterion balances model fit with a stronger penalty for large $p$, effectively controlling for the potential over-selection in high-dimensional settings.

We compare the two criteria in figures 3 and 8; it seems that indeed when $p >> n$ the EBIC criterion behaves better than the BIC criterion.

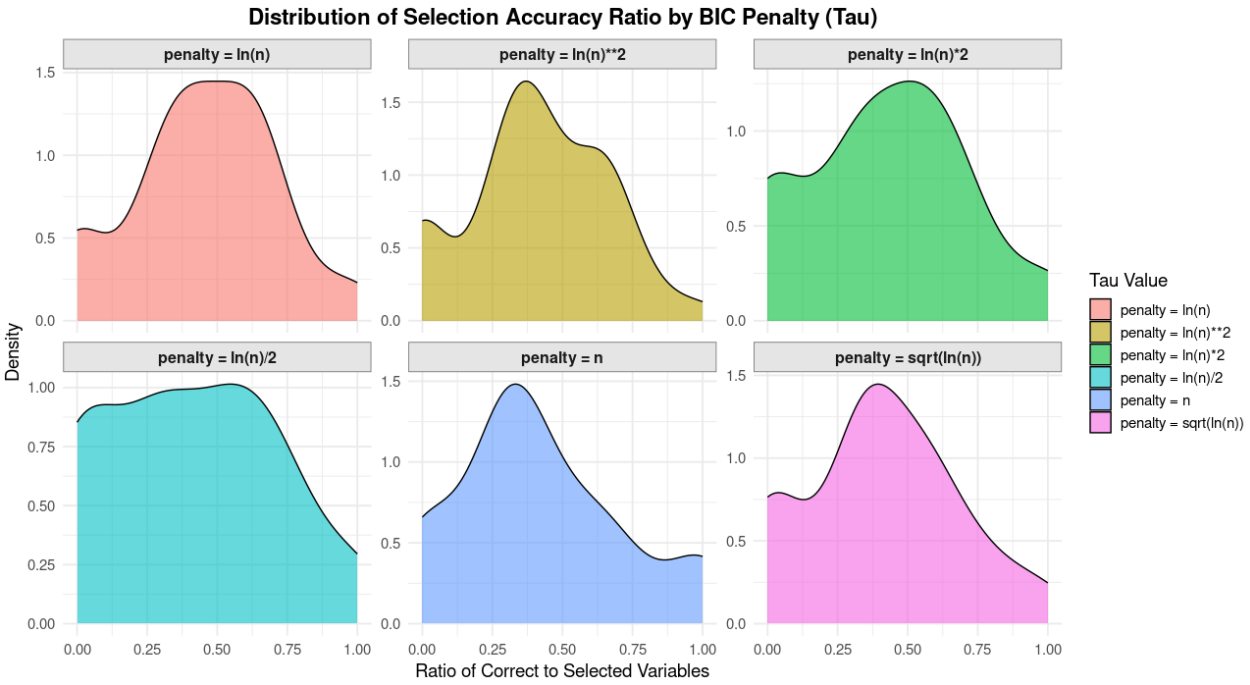


**Figure 2:** Density of the success ratio of number of selected variable with respect to true number of variable.

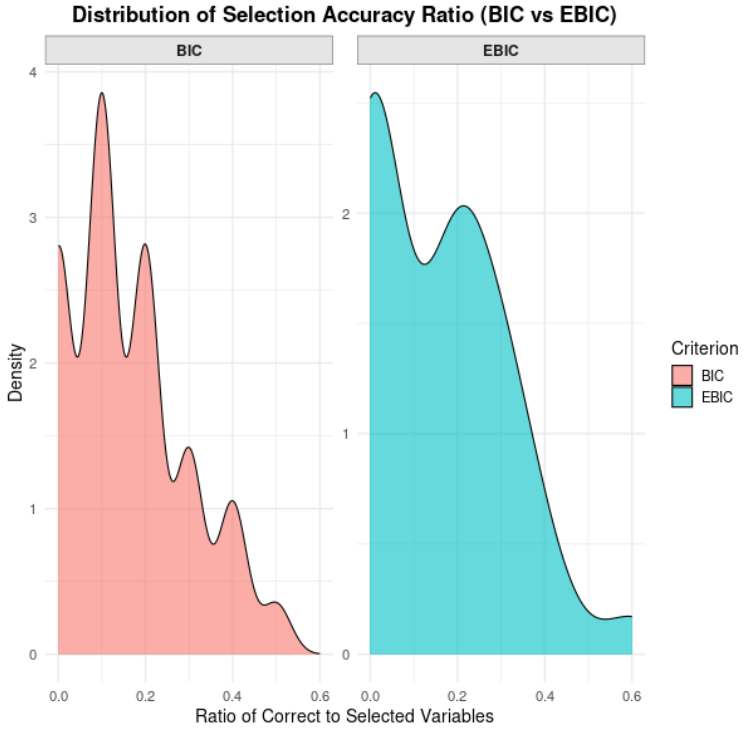


**Figure 3:** Comparison of the BIC and EBIC criteria in the high-dimensional setting (i.e., $p >> n$).

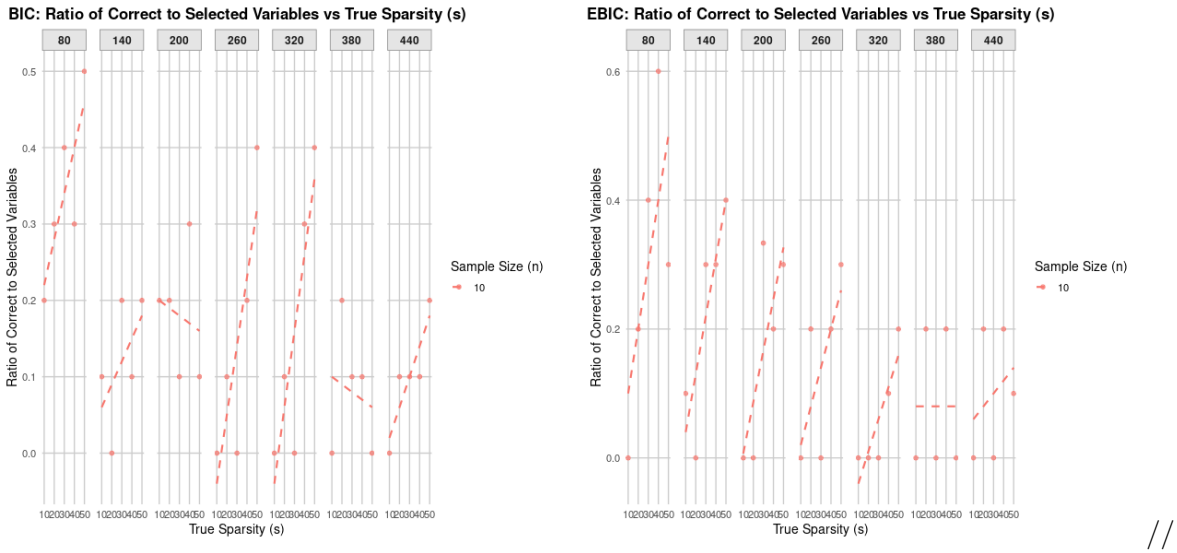


**Table 8:** BIC and EBIC comparison in high-dimensionality setting.

We can also look at what happens when we vary the convergence criterion. We want to check if varying the convergence criterion lead to different results. For different settings of $n\Omega p$ with $\Omega \in \{>, =, <<\}$, the result can be seen in figures 4, 5 and 6.

Case $n > p$. In this dimensionality configuration the BIC criterion seems to work perfectly regardless of the convergence criterion. Thus to still run test we increase the variance $\sigma$ of the selected variables.

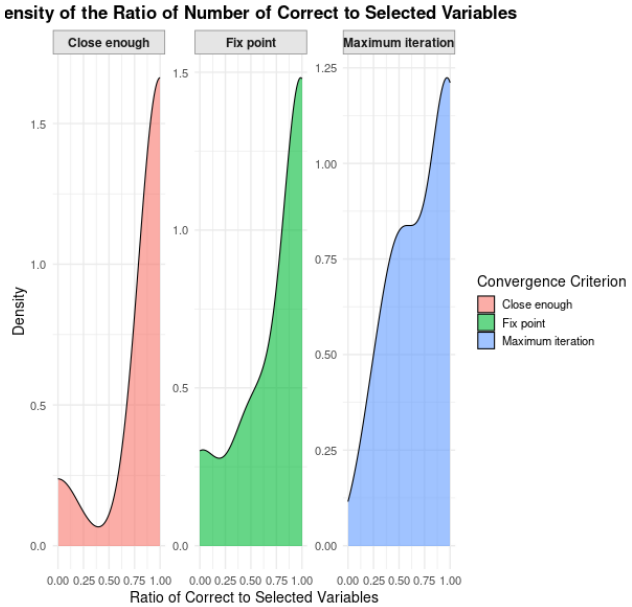


**Figure 4:** Comparison of the different convergence criterion when $n > p$.

Case $n = p$.

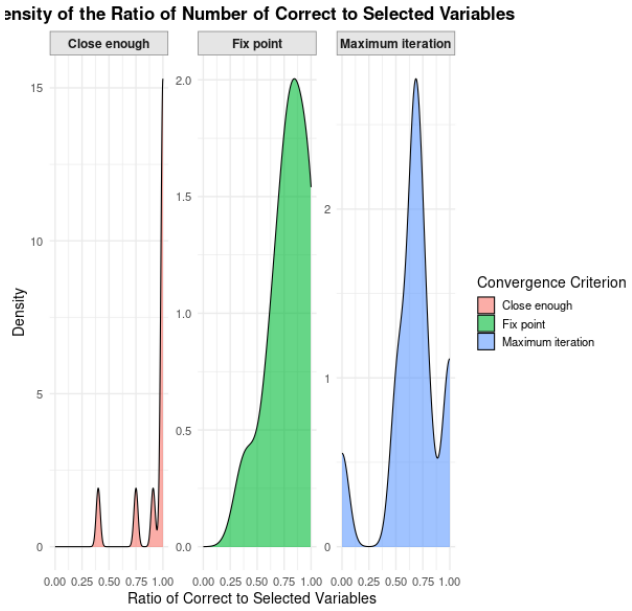


**Figure 5:** Comparison of the different convergence criterion when $n = p$

Case $n << p$.

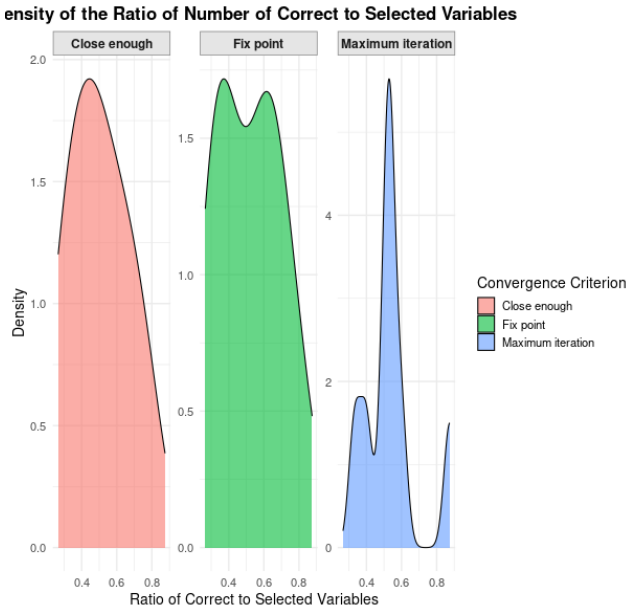


**Figure 6:** Comparison of the different convergence criterion when $n << p$

The other metric to take into account when comparing convergence methods is the computation

time until convergence.

**Note:** In the setting used for this experiments, the criterion close enough is by far the one taking the longest time.

**Remark:** Nevertheless these conclusions are to be taken lightly; indeed these criteria depend on hyper-parameters (i.e., the maximum number of iterations, and $\epsilon$ for close the enough method) which are here fixed and thus results could change with changed these hyper-parameters.

# 4 Exercise E.3 (Dantzig Selector)

## 4.1 QUESTION 1

We have :

$$\begin{aligned}
||\xi^T X||_\infty &= ||X^T \xi||_\infty \\
&= \max_j |x_{.,j}^T \xi| \\
&= \max_j ||x_{.,j}^T||_2 \frac{|x_{.,j}^T \xi|}{||x_{.,j}^T||_2} \\
&\leq \bar{x} \max_j |v_j^T \xi|
\end{aligned}$$

where, $\bar{x} = \max_j ||x_{.,j}^T||_2$ and $v_j = \frac{|x_{.,j}^T \xi|}{||x_{.,j}^T||_2}$ .

We have that $\forall j, v_j^T \xi \sim SG(v_j^T 0, v_j^T \sigma^2 I v_j) \sim SG(0, \sigma^2)$. By o.r.t of the $v_j$ family.

Thus :

$$\begin{aligned}
\mathbb{P}(||\xi^T X||_\infty \geq \rho) &\leq \mathbb{P}(\bar{x} \max_j |v_j^T \xi| \geq \rho) \\
&= \mathbb{P}(\bar{x} \max_j |v_j^T \xi| \geq \sigma \sqrt{(} 2ln(2p/\epsilon)) \\
&\leq \sum_j \mathbb{P}(|v_j^T \xi| \geq \sigma \sqrt{(} 2ln(2p/\epsilon)) \\
&\leq \sum_j 2e^{-ln(2p/\epsilon)} \\
&= \sum_j 2\epsilon/2p \\
&\leq \epsilon
\end{aligned}$$

which establishes the first result.

## 4.2 QUESTION 2

- We have $\xi \in \Theta$, thus,

$$\begin{aligned}
||X^T X v||_\infty &= ||X^T X(\hat{\theta} - \theta^*)||_\infty \\
&= ||X^T(\hat{(}y) - y + \xi)||_\infty \\
&\leq ||X^T(\hat{(}y) - X\theta||_\infty + ||X^T \xi||_\infty \\
&\leq \tau + \rho
\end{aligned}$$

because $||X^T(\hat{(}y) - X\theta||_\infty \leq \tau$ by definition, and $||X^T\xi||_\infty \leq \rho$ because $\xi \in \Theta$.

- $||X^T(y - X\theta^*)||_\infty = ||X^T\xi||_\infty \leq \rho \leq \tau$ We deduce from the definition of $\hat{\theta}$ : $||\hat{\theta}||_1 \leq ||\theta^*||_1$.
- We now evaluate the error of the estimate $\hat{\theta}$

$$\begin{aligned} ||\hat{\theta}||_1 &= \sum_{j\in J} |\hat{\theta}_j| + \sum_{j\notin J} |\hat{\theta}_j| \leq \sum_{j\in J} |\theta^*_j| + 0 \\ &=> \sum_{j\notin J} |\hat{\theta}_j| \leq \sum_{j\in J} |\hat{\theta}_j - \theta^*_j| \\ &\text{And we have } \sum_{j\in J} |\hat{\theta}_j - \theta^*_j| = ||v_{J^c}|| \leq ||v_J|| \end{aligned}$$

## 4.3 QUESTION 3

- We have $||v||_1 = ||v_{J_c}||_1 + ||v_J||_1 \leq 2||v_{J_c}||_1$ , by question 2.
- Condition $Q_\infty(\beta, k, s)$ implies that :

$$\begin{aligned} 2||v_{J_c}||_1 &\leq 2(\beta s||X^TXv||_\infty + k||v||_1) \\ &=> ||v||_1 - 2k||v||_1 \leq 2\beta s(\tau + \rho) \\ &\text{And finally, } => ||v||_1 \leq \frac{2\beta s(\tau+\rho)}{(1-2k)} \text{ because, } 1 - 2k > 0. \end{aligned}$$

- Next, we compute a bound for the error prediction.

$$\begin{aligned} ||Xv||_2^2 = (Xv)^T(Xv) &=< v|X^TXv > \\ &= \sum_{i=1}^{p} |v_i||(X^TXv)_i| \\ &\leq \sum_{i=1}^{p} |v_i| max_j |(X^TXv)_j| \\ &= ||v||_1||X^TXv||_\infty \\ &\leq \frac{2\beta s(\tau+\rho)^2}{(1-2k)} \text{by latest inequality and definition of } ||X^TXv||_\infty \end{aligned}$$

- Introducing $v^1$ the vector obtained from $v$ by zeroing all but $s$ entries of the maximal amplitude. We get :

$$
\begin{aligned}
||v^1||_2 = ||v||_{s,2} &\leq \sqrt(s)||X^TXv||_\infty + \frac{k}{\sqrt{s}}||v||_1 \\
&\leq \sqrt{s}(\tau+\rho) + \frac{k}{\sqrt{s}}\frac{2\beta s(\tau+\rho)}{(1-2k)} \text{ by, } Q_\infty(\beta,k,s) \\
&\leq \sqrt{s}(\tau+\rho)(1+\frac{2k}{1-2k}) \\
&\text{Thus, we have : } ||v^1||_2 \leq \frac{2\beta s(\tau+\rho)}{(1-2k)}
\end{aligned}
$$

- By definition of $v^1$, if $j_0 \notin J$ is such that $|v_{j_0}| = max_{j\notin J} = max_j|v_j^1|$, then $v_{j_0}^2 \leq v_j^2, \forall j in J$. Which implies that :

$$
\begin{aligned}
sv_{j_0} &\leq \sum_{j\in J} v_j^2, \qquad\qquad Card(J) = s \\
&=> |v_{j_0}| \leq \frac{1}{\sqrt{s}}\sqrt{\sum_{j\in J} v_j^2} = \frac{||v||_{s,2}}{\sqrt{s}} = \frac{||v^1||_2}{\sqrt{s}}
\end{aligned}
$$

  Hence, by Holder inequality $||v-v^1||_2^2 \leq ||v-v^1||_\infty||v-v^1||_1 \leq |v_{j_0}|||v-v^1||_1$.

  And, we get : $||v-v^1||_2^2 \leq \frac{||v^1||_2}{\sqrt{s}}||v-v^1||_1 \leq \frac{||v^1||_2}{\sqrt{s}}||v||_1$

- We deduce : $||v-v^1||_2^2 \leq s(\frac{2\beta(\tau+\rho)}{1-2k})^2$, and $||v||_2^2 \leq s(\frac{2\beta(\tau+\rho)}{1-2k})^2$.

  Hence, $||v||_2 = \sqrt{||v^1||_2^2 + ||v-v^1||_2^2} \leq \frac{4\beta\sqrt{s}(\tau+\rho)}{1-2k}$

## 4.4 QUESTION 4

Using the same process used in the proof of property 4.2 we define $v^1, ..., v^q$ which are $d$-sparse and such that $v = v^1 + ... + v^q$ with $< v^i, v^j >= 0, \forall i \neq j$. We also have for all $j \geq 2$, $||v^j||_\infty \leq \frac{||v^{j-1}||}{\sqrt{d}}$, $||v^j|| \leq ||v^{j-1}||_1$ and $||v^j||_2 \leq \sqrt{||v^j||_\infty||v^j||_1} \leq \frac{||v^j||}{\sqrt{d}}$.

Since $X$ is $RI(\delta, 2d)$ for every $d$-sparse vectors $u, v$ with overlapping support, we have :

$$
|u^TX^TXv| \leq \delta||u||_2||v||_2 \tag{1}
$$

By Holder inequality, $||v^1||_1||X^TXv||_\infty \geq (v^1)^TX^TXv$, but, $(v^1)^TX^TXv = ||Xv^1||_2^2 + \sum_{i=2}^q (v^1)^TX^TXv^i$ and since $X$ is $RI(\delta, 2d)$, we use (1) and then :

$$\begin{aligned}(v^1)^T X^T X v &\geq ||Xv^1||_2^2 - \delta ||v^1||_2 \sum_{i=2}^{q} ||v^i||_2 \\ &\geq ||Xv^1||_2^2 - \delta s^{-1/2} ||v^1||_2 \sum_{i=2}^{q} ||v^{i-1}||_1 \\ &\geq ||Xv^1||_2^2 - \delta s^{-1/2} ||v^1||_2 ||v^{|}|_1\end{aligned}$$

Thus $||Xv^1||_2^2 \leq ||v^1||_1 ||X^T X v||_\infty + \delta s^{-1/2} ||v^1||_2 ||v^{|}|_1$. Hence we have :

- $||v^1||_2 = \frac{||v^1||_2}{||Xv^1||_2^2} ||Xv^1||_2^2 \leq \frac{||v^1||_2}{||Xv^1||_2^2} ||v^1||_1 ||X^T X v||_\infty + \delta \frac{||v^1||_2^2}{||Xv^1||_2^2} ||v||_1$
- $||v^1||_1 = \sqrt{s} (\frac{||v^1||_2}{||Xv^1||_2})^2 ||X^T X v||_\infty + \frac{\delta}{\sqrt{s}} \frac{||v^1||_2}{||Xv^1||_2} ||v||_1$
- $\frac{||v^1||_2}{||Xv^1||_2} \leq \frac{1}{\sqrt{1-\delta}} \leq \sqrt{s} \frac{||X^T X v||_\infty}{1-\delta} + \frac{\delta}{\sqrt{s}(1-\delta)} ||v||_1$

We proved $X$ verifies $Q_\infty(\beta, k, s)$ with $\beta = \frac{1}{1-\delta}$ and $k = \frac{\delta}{1-\delta}$