A Review on Best Subset, Forward Stepwise or Lasso? Analysis and Recommendations Based on Extensive Comparisons

Advanced Statistical Computing, Fall 2021

Myungjun Kim, Seungyeop Hyun and Taeyoung Chang

December 15, 2021

1 Introduction

In recent work, Bertsimas, King and Mazumder (2016) suggested a *Mixed Integer Optimization* (MIO) approach to solve the best subset selection problem,

$$\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \quad \|Y - X\beta\|_2^2 \quad \text{subject to} \quad \|\beta\|_0 \leq k.$$

Using recent advances in MIO algorithms, they demonstrated that best subset selection can now be solved at much larger problem sizes than what was thought possible.

1.1 Is Best Subset the Holy Grail?

Hastie, Tibshirani and Tibshirani (2020) pointed out that neither best subset nor the lasso uniformly dominate the other over the wide range of signal-to-noise ratio (SNR). When there is an observational noise like real world dataset, whether the best subset gives a better estimator than others is a subtle question. Different procedures have different operating characteristics, that is, give rise to different bias-variance tradeoffs as tuning parameters vary.

1.2 What Is a Realistic Signal-to-Noise Ratio?

Let $y_0 = f(x_0) + \epsilon_0$ where x_0 and ϵ_0 are independent. The SNR and the proportion of variance explained (PVE) are defined as

$$SNR = \frac{Var(f(x_0))}{Var(\epsilon_0)} \quad \text{and} \quad PVE(f) = 1 - \frac{Var(\epsilon_0)}{Var(y_0)} = \frac{SNR}{1 + SNR}.$$

A PVE of 0.5 (SNR = 1) is rare for noisy observational data, and 0.2 (SNR = 0.25) may be more typical. A PVE of 0.86 (SNR = 6) seems unrealistic. Bertsimas, King and Mazumder (2016) considered SNRs in the range of about 2 to 8 in low-dimensional cases, and about 3 to 10 in high-dimensional cases. Hastie et al. (2020) also have doubts about this choice of SNR range.

The goal of this paper is *not* about 1) What is the best prediction algorithm? 2) What is the best variable selector? or 3) Empirically validating theory for ℓ_0 and ℓ_1 penalties. Rather, this paper *is* about the relative merits of the three most canonical forms for sparse estimation in a linear model: ℓ_0, ℓ_1 and forward stepwise selection.

2 Algorithms

2.1 Best Subset Selection

The best subset problem is written by

$$\text{minimize}_{\beta} \|Y - X\beta\|_2^2 \quad \text{subject to} \quad \|\beta\|_0 \leq k$$

Best subset finds the k predictors that produces the best fit in terms of squared error. It is non-convex problem and is known to be NP-hard. A mixed integer optimization (MIO) formulation for the best subset problem is suggested (Bertsimas, King and Mazumder, 2016).

2.1.1 MIO Formulations for the Best Subset Problem

The best subset problem can be structured as the following MIO formulation:

$$\begin{aligned} \underset{\beta, \mathbf{z}}{\text{minimize}} \quad & \|Y - X\beta\|_2^2 \\ \text{subject to} \quad & \beta_i(1 - z_i) = 0, \quad \forall\, i = 1, \cdots, p \\ & z_i \in \{0, 1\}, \quad \forall\, i = 1, \cdots, p \\ & \sum_{i=1}^{p} z_i \leq k. \end{aligned}$$

Adding problem-dependent constants M_U and M_ℓ, a more structured rerpresentation can be given as

$$\begin{aligned} \underset{\beta, \mathbf{z}}{\text{minimize}} \quad & \frac{1}{2}\beta^T(X^TX)\beta - \langle X^Ty, \beta\rangle + \frac{1}{2}\|y\|_2^2 \\ \text{subject to} \quad & \beta_i(1 - z_i) = 0, \quad \forall\, i = 1, \cdots, p \\ & z_i \in \{0, 1\}, \quad \forall\, i = 1, \cdots, p \\ & \sum_{i=1}^{p} z_i \leq k, \\ & \|\beta\|_\infty \leq M_U \quad \text{and} \quad \|\beta\|_1 \leq M_\ell. \end{aligned}$$

Utilizing these bounds typically leads to improved performance of MIO. For $n < p$ case, we add another optimization variable $\xi \in \mathbb{R}^n$ with constraints $\xi = X\beta$.

2.1.2 Obtaining Warmstart for the Optimization

Our situation can be viewed as

$$\begin{aligned} \underset{\beta}{\text{minimize}} \quad & g(\beta) \\ \text{subject to} \quad & \|\beta\|_0 \leq k, \end{aligned}$$

where $g(\beta) = \frac{1}{2}\|y - X\beta\|_2^2$. Note that g is convex and has Lipshcitz continuous gradient with Lipshcitz constant $\ell = \lambda_{\max}(X^TX)$. For such convex function $g(\beta)$, with any $L \geq \ell$, we have

$$g(\eta) \leq Q_L(\eta, \beta) = g(\beta) + \frac{L}{2}\|\eta - \beta\|_2^2 + \langle \nabla(\beta), \eta - \beta\rangle.$$

We want to find $\text{argmin}_{\|\eta\|_0 \leq k} Q_L(\eta, \beta)$ with given β for getting close to the minimizer of $g(\beta)$.

2.1.3 Projected Gradient Method

$\text{argmin}_{\|\eta\|_0 \le k} Q_L(\eta, \beta)$ has a closed form solution which is

$$H_k\left(\beta - \frac{1}{L}\nabla g(\beta)\right),$$

where $H_k(\mathbf{c})$ denotes the projection to the coordinates having k largest (in absolute value) elements of $\mathbf{c}$. By updating

$$\beta_{m+1} \in H_k\left(\beta_m - \frac{1}{L}\nabla g(\beta_m)\right),$$

we can find a stationary point of the main problem. We exploit this value as a warmstart for the optimization of MIO problem using solver.

2.2 Forward Stepwise Selection

Forward stepwise is less ambitious version of best subset. It starts with the empty model and iteratively adds the variable that best improves the fit. Formally, the procedure starts with an empty active set $\mathcal{A}_0$ and for each step $k = 1, \cdots, \min\{n, p\}$, we select the variable indexed by

$$j_k = \text{argmin}_{j \notin \mathcal{A}_{k-1}} \|y - P_{\mathcal{A}_{k-1} \cup \{j_k\}} y\|_2^2.$$

It means that it chooses the variable that leads to the lowest squared error when added to $\mathcal{A}_{k-1}$. Equivalently, it adds the variable which achieves the maximum absolute correlation with y after we project out the contributions from $X_{\mathcal{A}_{k-1}}$

$$\begin{aligned}
&\text{minimize} \quad \|y - P_{\mathcal{A}_{k-1} \cup \{j_k\}} y\|_2^2 \\
&\Leftrightarrow \text{maximize} \quad \|P_{A_{k-1} \cup \{j_k\}} y\|_2^2 \quad \because \text{ Pythagorean Law} \\
&\Leftrightarrow \text{maximize} \quad \|P_{(I - P_{\mathcal{A}_{k-1}})X_{j_k}} y\|_2^2 \quad \because \ \|P_{\mathcal{A}_{k-1} \cup \{j_k\}} y\|_2^2 = \|P_{\mathcal{A}_{k-1}} y\|_2^2 + \|P_{(I - P_{\mathcal{A}_{k-1}})X_{j_k}} y\|_2^2 \\
&\Leftrightarrow \text{maximize} \quad \frac{|\langle\ (I - P_{\mathcal{A}_{k-1}})X_{j_k}\ ,\ y\ \rangle|}{\|(I - P_{\mathcal{A}_{k-1}})X_{j_k}\|_2}
\end{aligned}$$

The forward stepwise selection is highly structured and this greatly aids its computation. Suppose that we have maintained a QR decomposition of active submatrix $X_{\mathcal{A}_{k-1}}$ of predictors and the orthogonalized remaining predictors with respect to $X_{\mathcal{A}_{k-1}}$. Then we find one of remaining predictor which has maximum absolute correlation with y. To update

$$X_{\mathcal{A}_{k-1}} = Q_{k-1} R_{k-1} \quad \text{to} \quad X_{\mathcal{A}_k} = Q_k R_k$$

with selected variable X_{j_k}, we shall take advantage of modified Gram-Schimidt algorithm.

Using MGS, we can derive k-th column of Q_k and k-th column of R_k

$$\begin{aligned}
\mathbf{v}_k &= \mathbf{x}_{j_k} - P_{\text{span}(\{\mathbf{q}_1, \cdots, \mathbf{q}_{k-1}\})}(\mathbf{x}_{j_k}) = \mathbf{x}_{j_k} - \sum_{j=1}^{k-1} \langle \mathbf{q}_j, \mathbf{x}_k \rangle \cdot \mathbf{q}_j \\
&= \mathbf{x}_k - \sum_{j=1}^{k-1} \left\langle \mathbf{q}_j, \mathbf{x}_{j_k} - \sum_{i=1}^{j-1} \langle \mathbf{q}_i, \mathbf{x}_{j_k} \rangle \mathbf{q}_i \right\rangle \cdot \mathbf{q}_j \\
\mathbf{q}_k &= \mathbf{v}_k / \|\mathbf{v}_k\|_2
\end{aligned}$$

Orthogonalizing the remaing predictors with respect to the one just included can be done using Q_k since $I - P_{\mathcal{A}_k} = I - Q_k Q_k^T$.

2.3 The Lasso

The lasso problem is written by

$$\underset{\beta \in \mathbb{R}^p}{\text{minimize}} \quad \frac{1}{2}\|Y - X\beta\|_2^2 + \lambda\|\beta\|_1, \quad \lambda \geq 0.$$

It solves a convex relaxtion of best subset problem where we replace the ℓ_0 norm by the ℓ_1 norm.

2.3.1 Pathwise Coordinate Descent

R package `glmnet` solves the lasso problem using *pathwise coordinate descent*. It computes the solutions for a decreasing sequence of λ,

$$\lambda_{\max} = \|X^T Y\|_\infty > \cdots > \lambda_{\min} = \epsilon\lambda_{\max},$$

typically with $\epsilon = 0.001$ and $K = 100$ values of λ on the log scale. Starting at $\lambda_{\max}$, where all coefficients of the solution $\hat{\beta}$ are zero, we use *warm starts* in computing the solutions at the sequence of λ, i.e., $\hat{\beta}(\lambda_k)$ is used as an initial value for λ_{k+1}.

2.3.2 Active Set Strategy

After one or several cycles through p variables, we store the nonzero coefficient in the active set $\mathcal{A}$. Further iterations of coordinate descent restricted to $\mathcal{A}$ continues till convergence. One more cycle through all variables is necessary to check KKT optimality conditions:

$$\begin{aligned} \left|\langle\, x_j\,,\, y - X\hat{\beta}(\lambda)\,\rangle\right| &= \lambda, \quad \forall j \in \mathcal{A}, \\ \left|\langle\, x_j\,,\, y - X\hat{\beta}(\lambda)\,\rangle\right| &\leq \lambda, \quad \forall j \notin \mathcal{A}. \end{aligned}$$

If there were a variable violating the conditions, then add it in $\mathcal{A}$ and go back to the previous step.

2.3.3 Screening Rule

For some problems, screening rules can be used in combination with coordinate descent to further wittle down the active set. For the lasso, Tibshirani (2012) suggested the *sequential strong rules* which discards the jth predictor from the optimization problem at λ_k if

$$\left|\langle\, x_j\,,\, y - X\hat{\beta}(\lambda_{k-1})\,\rangle\right| < 2\lambda_k - \lambda_{k-1}.$$

Motivation for the strong rules comes with KKT conditions. If we assume that we can bound the amount that $c_j(\lambda) = \langle\, x_j\,,\, y - X\hat{\beta}(\lambda)\,\rangle$ changes as we move from λ to another $\tilde{\lambda}$, i.e.,

$$|c_j(\lambda) - c_j(\tilde{\lambda})| \leq |\lambda - \tilde{\lambda}| \quad \forall\, \lambda\,,\, \tilde{\lambda}, \quad \text{and} \quad \forall j = 1, \cdots, p$$

then $|c_j(\lambda_{k-1})| < 2\lambda_k - \lambda_{k-1}$ (which satisfying strong rule) implies

$$\begin{aligned} |c_j(\lambda_k)| &\leq |c_j(\lambda_k) - c_j(\lambda_{k-1})| + |c_j(\lambda_{k-1})| \\ &< (\lambda_{k-1} - \lambda_k) + (2\lambda_k - \lambda_{k-1}) = \lambda_k. \end{aligned}$$

so that $\hat{\beta}_j(\lambda_k) = 0$ by the KKT conditions. The sequential strong rule can mistakenly discard active predictors, so it must be combined with a check of the KKT conditions.

2.3.4 Algorithm for lasso implemented by `glmnet`

Using both *ever-active* set of predictors $\mathcal{A}(\lambda)$ and the strong set $S(\lambda)$ which is the set of the indices of the predictors that survive the screening rule can be advantageous.

1. Set $\mathcal{E} = \mathcal{A}(\lambda)$.
2. Solve the problem at value λ by using only the predictors in $\mathcal{E}$.
3. Check the KKT conditions at this solution for all predictors in $S(\lambda)$. If violated, then add these violating predictors into $\mathcal{E}$ and go back to previous step using the current solution as a warm start.
4. Check the KKT conditions at all predictors. No violations means we are done. Otherwise, add these violators into $\mathcal{E}$, recompute $S(\lambda)$ and go back to the first step using the current solution as a warm start.

Note that violations in the third step are faily common whereas those in the fourth step are rare. Hence the fact that the size of $S(\lambda)$ is very much less than p makes this an effective strategy.

2.3.5 A (Simplified) Relaxed Lasso

A simplified version of the relaxed lasso estimator is

$$\hat{\beta}^{\text{relax}}(\lambda, \gamma) = \gamma \hat{\beta}^{\text{lasso}}(\lambda) + (1 - \gamma)\hat{\beta}^{\text{LS}}(\lambda),$$

where $\lambda \leq 0$ and $\gamma \in [0, 1]$. Here, $\mathcal{A}_\lambda$ is the active set of $\hat{\beta}^{\text{lasso}}(\lambda)$ and $\hat{\beta}^{\text{LS}}(\lambda)$ is the full-sized version of $\hat{\beta}^{\text{LS}}_{\mathcal{A}_\lambda} = (X^T_{\mathcal{A}_\lambda} X_{\mathcal{A}_\lambda})^{-1} X^T_{\mathcal{A}_\lambda} Y$, padded with zeros. The relaxed lasso tries to undo the shrinkage inherent in the lasso estimator. If γ is away from 1, then the relaxed lasso estimator withdraws the shrinkage.

We have implemented the best subset selection, the forward stepwise selection, the lasso and the relaxed lasso in Julia. In `test.ipynb`, we compare the results of our functions with `bestsubset` package in R.

3 Simulations

3.1 Setup

- Define coefficients $\beta_0 \in \mathbb{R}^p$ according to s (sparsity level) and the beta-type.
- The predictor matrix $X \in \mathbb{R}^{n \times p}$ i.i.d. from $N_p(0, \Sigma)$ where $\Sigma_{ij} = \rho^{|i-j|}$ and $\rho \in \{0, 0.35, 0.70\}$.
- The response vector $Y \in \mathbb{R}^n$ from $N_n(X\beta_0, \sigma^2 I)$ with σ^2 defined to meet the desired SNR level ν, i.e., $\sigma^2 = \beta_0^T \Sigma \beta_0 / \nu$.

$$\nu = \frac{Var(f(x_0))}{Var(\epsilon_0)} = \frac{Var(x_0^T \beta_0)}{Var(\epsilon_0)} = \frac{\beta_0^T \Sigma \beta_0}{\sigma^2}$$

- Run the lasso, relaxed lasso, forward stepwise, and best subset on the data over a wide range of parameters, and choose the parameter by minimizing prediction error on a validation set.
- Record several metrics of interest and repeat total of 10 times, and average the results.

3.2 Coefficients

We chose a sparse β_0 with the following various types.

- beta-type 1: β_0 has s components equal to 1, occuring at equally-spaced indices between 1 and p, and the rest equal to 0.
- beta-type 2: β_0 has its first s components equal to 1, and the rest equal to 0.
- beta-type 3: β_0 has its first s components taking nonzero values equally-spaced between 10 and 0.5, and the rest equal to 0
- beta-type 5: β_0 has its first s components equal to 1, and the rest decaying exponentially to 0, specifically, $\beta_{0i} = 0.5^{i-s}$, for $i = s+1, \ldots, p$.

3.3 Configurations

We considered the following four problem settings.

Setting	n	p	s
Low	100	10	5
Medium	500	100	5
High-5	50	1000	5
High-10	100	1000	10

In each setting, we considered ten values for the SNR ranging from 0.05 to 6 on a log scale.

SNR	0.05	0.09	0.14	0.25	0.42	0.71	1.22	2.07	3.52	6.00
PVE	0.05	0.08	0.12	0.20	0.30	0.42	0.55	0.67	0.78	0.86

3.4 Evaluation Metrics

- Relative risk:
$$RR(\hat{\beta}) = \frac{\mathbb{E}(x_0^T\hat{\beta} - x_0^T\beta_0)^2}{\mathbb{E}(x_0^T\beta_0)^2} = \frac{(\hat{\beta}-\beta_0)^T\Sigma(\hat{\beta}-\beta_0)}{\beta_0^T\Sigma\beta_0}$$
A perfect score is 1 (if $\hat{\beta} = \beta_0$) and the null score is 1 (if $\hat{\beta} = 0$).
- Relative test error:
$$RTE(\hat{\beta}) = \frac{\mathbb{E}(y_0 - x_0^T\hat{\beta}^2)}{\sigma^2} = \frac{(\hat{\beta}-\beta_0)^T\Sigma(\hat{\beta}-\beta_0) + \sigma^2}{\sigma^2}$$
The null score is $(\beta_0^T\Sigma\beta_0 + \sigma^2)/\sigma^2 = SNR + 1$.
- Proportion of variance explained:
$$PVE(\hat{\beta}) = 1 - \frac{\mathbb{E}(y_0 - x_0^T\hat{\beta}^2)}{Var(y_0)} = 1 - \frac{(\hat{\beta}-\beta_0)^T\Sigma(\hat{\beta}-\beta_0) + \sigma^2}{\beta_0^T\Sigma\beta_0 + \sigma^2}$$
A perfect score is $SNR/(1+SNR)$ and the null score is 0.

Since the coefficient β_0 is sparse, it is also important to classify whether each β_j is zero or not. Hence it is worth scoring number of nonzeros and F-score of each model.

- Number of nonzeros: $\|\hat{\beta}\|_0 = \sum_{i=1}^{p} 1\{\hat{\beta}_i \neq 0\}$
- F-Score: the harmonic mean of recall and precision

3.5 Results

3.5.1 Computation Time

Setting	n	p	s	BS	FS	Lasso	RLasso
Low	100	10	5	0.313	0.003	0.002	0.002
Medium	500	100	5	76.8 hr	0.890	0.013	0.154
High-5	50	1000	5	44.2 hr	0.123	0.014	0.159
High-10	100	1000	10	61.7 hr	0.254	0.024	0.158

Table 1: Time in seconds for one path of solutions for each method

Setting	n	p	s	BS	FS	Lasso	RLasso
Low	100	10	5	2.20	0.026	0.0006	0.0009
Medium	500	100	5	4634	1.801	0.004	0.056
High-5	50	1000	5	4896	0.127	0.003	0.018
High-10	100	1000	10	4905	0.454	0.010	0.038

Table 2: Reproduced time in seconds for one path of solutions for each method

The above table is from the original paper, and the below is our computation time table. The lasso and the relaxed lasso solve the problems in less than 0.01 seconds, which outperforms the forward stepwise and the best subset selection. Different time-limit for the best subset selection gives quite different computation time, but the best subset selection still spends much more time than other methods.

3.5.2 Effective Degrees of Freedom

In the context of bias-variance tradeoff, some methods such as the forward stepwise selection or the best subset selection find an estimator which minimizes the bias error rather than the variance error. On the other hand, another model, e.g. the lasso, focuses on the low-variance, and one can say it is less *aggressive* than the former ones. The effective degrees of freedom measures the aggressiveness of model. If a model uses a low-bias and high-variance estiamtor, then the fitted value $\hat{Y}_i$ tends to be close to the true value Y_i so that $Cov(Y_i, \hat{Y}_i)$ increases.

The left panel is original and the right panel is reproduced via Monte Carlo evaluation of the covariance with 500 replications. The relaxed lasso located in the middle because it tries to undo the shrinkage inherent of the lasso.

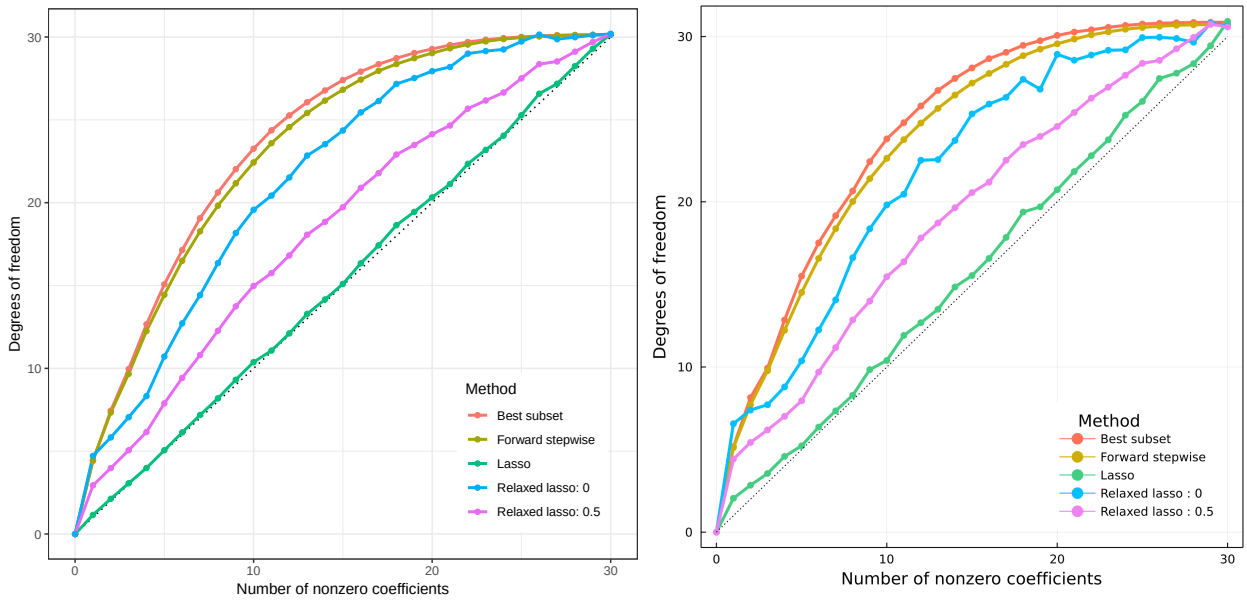


Figure 1: Degrees of freedom $\sum_{i=1}^{n} Cov(Y_i, \hat{Y}_i)/\sigma^2$ for the lasso, forward stepwise, best subset and the relaxed lass with $\gamma = 0.5$ and $\gamma = 0$.

3.5.3 Accuracy Metrics

We display a slice of the accuracy results, focusing for concreteness on the case in which the predictor correlation level is $\rho = 0.35$, and the population coefficients follow the beta-type 2 pattern except for Figure 5. The left figures are from the paper, and the right figures are reproduced by our simulations. Except for the relative error, all figures exhibit the relative test error, PVE, number of nonzero coefficients and F-score as functions of the SNR level for the corresponding setting. Figures 2, 3 and 4 show for the low, medium and high-5 setting. Figure 5 shows the same for the high-5 setting and beta-type 1. We average the metrics for the four methods over 10 repititions. In the relative test error plots, the dotted line represents the null score, $SNR + 1$; in the PVE plots, it is the perfect score SNR / (1 + SNR); in the number of nonzeros plots, it is the true number of nonzeros s.

In Figures 2 and 3, the lasso has less relative test error than the forward stepwise and the best subset, and the situation is reversed around $\nu = 1.22$ and $\nu = 0.25$, respectively. Thanks to its flexibility to catering the shrinkage inherent, the relaxed lasso gives the least relative test error over the entire SNR range. All four methods deliver almost perfect PVEs for all SNR levels. In the high-5 setting, however, the results are quite different. In the relative test error plot, there is no difference among the null model ($\hat{\beta} = 0$) and four methods at low SNR levels. For this reason, the PVEs are close to zero at low SNR levels. Especially, the forward stepwise and the best subset have poor accuracy metrics. Only in Figure 5, the best subset selection consistently has the best accuracy.

The lasso shows revealing behavior in the number of nonzeros plots. Due to the shrinkage, the lasso brings much denser solutions than the forward stepwise and the best subset. Although the number of nonzeros coeffients are different, they achieve the similar relative test error using opposite ways. The lasso prefers low-variance estimates; the best subset and the forward stepwise favors low-bias estimates. Also, it is noticeable that the lasso and the relaxed lasso have the different number of nonzeros coefficients. By introducing another tuning parameter γ, the relaxed lasso may achieve the optimal value at some λ at which the lasso does not. For more details, see `NumNonzeroLassoRelax.ipynb`. Since the relaxed lasso chooses larger λ than the lasso through the tuning (which can be balanced by taking weighted average with active set restriced LSE), we can understand why does the relaxed lasso have the smaller number of nonzero coefficients.

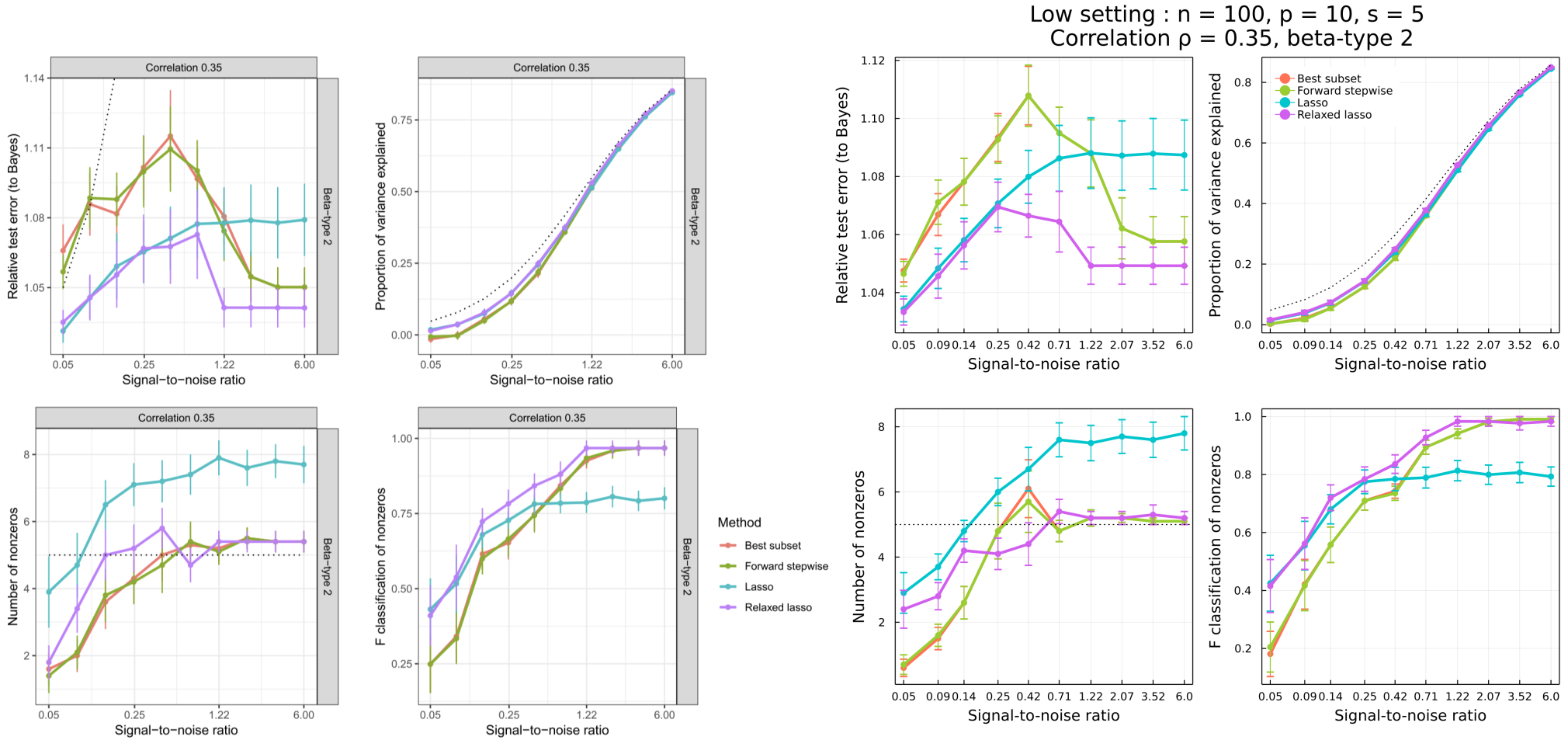


Figure 2: RTE, PVE, number of nonzero coefficients and F-score as functions of SNR with $n = 100, p = 10, s = 5, \rho = 0.35$ and beta-type 2.

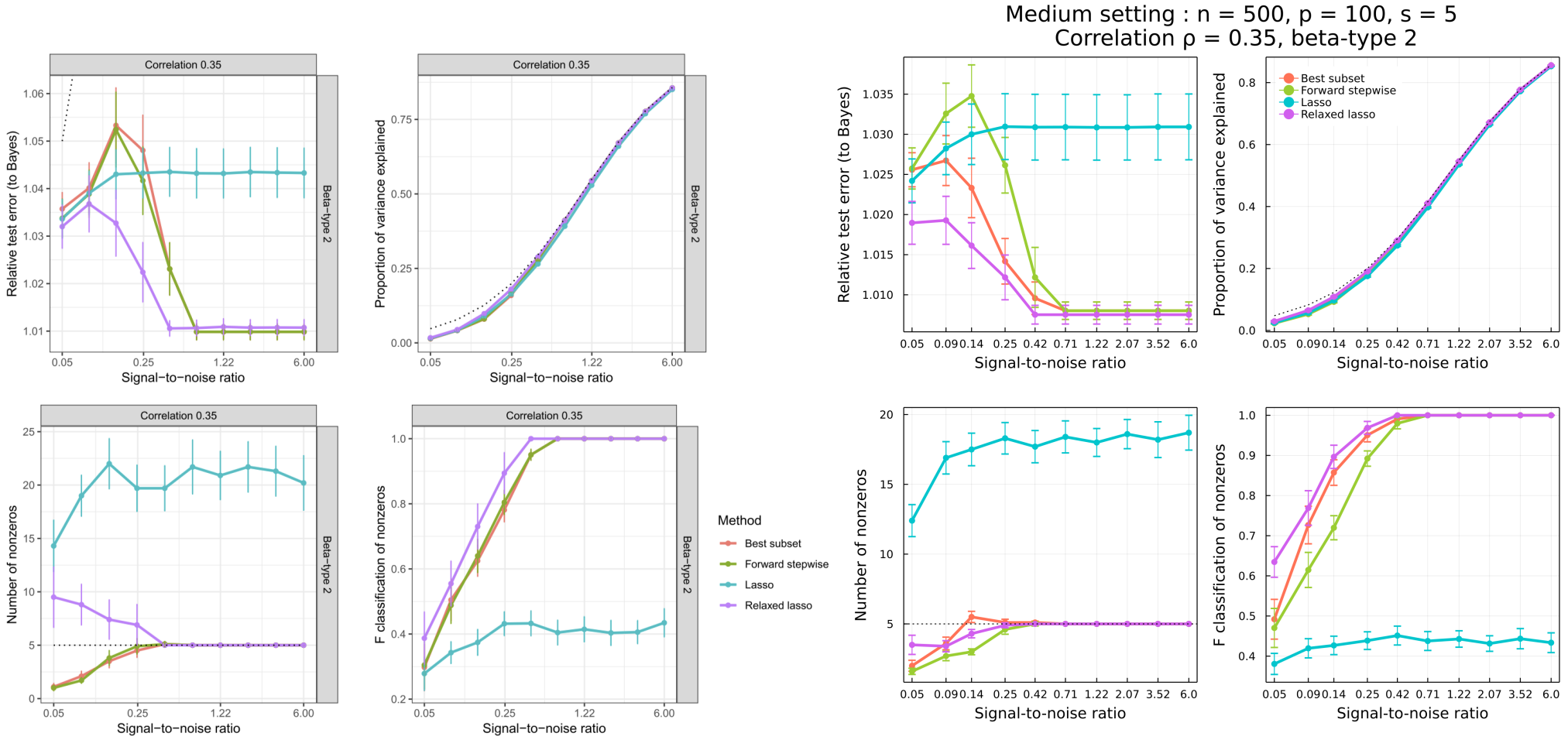


Figure 3: RTE, PVE, number of nonzero coefficients and F-score as functions of SNR with $n = 500$, $p = 100$, $s = 5$, $\rho = 0.35$ and beta-type 2.

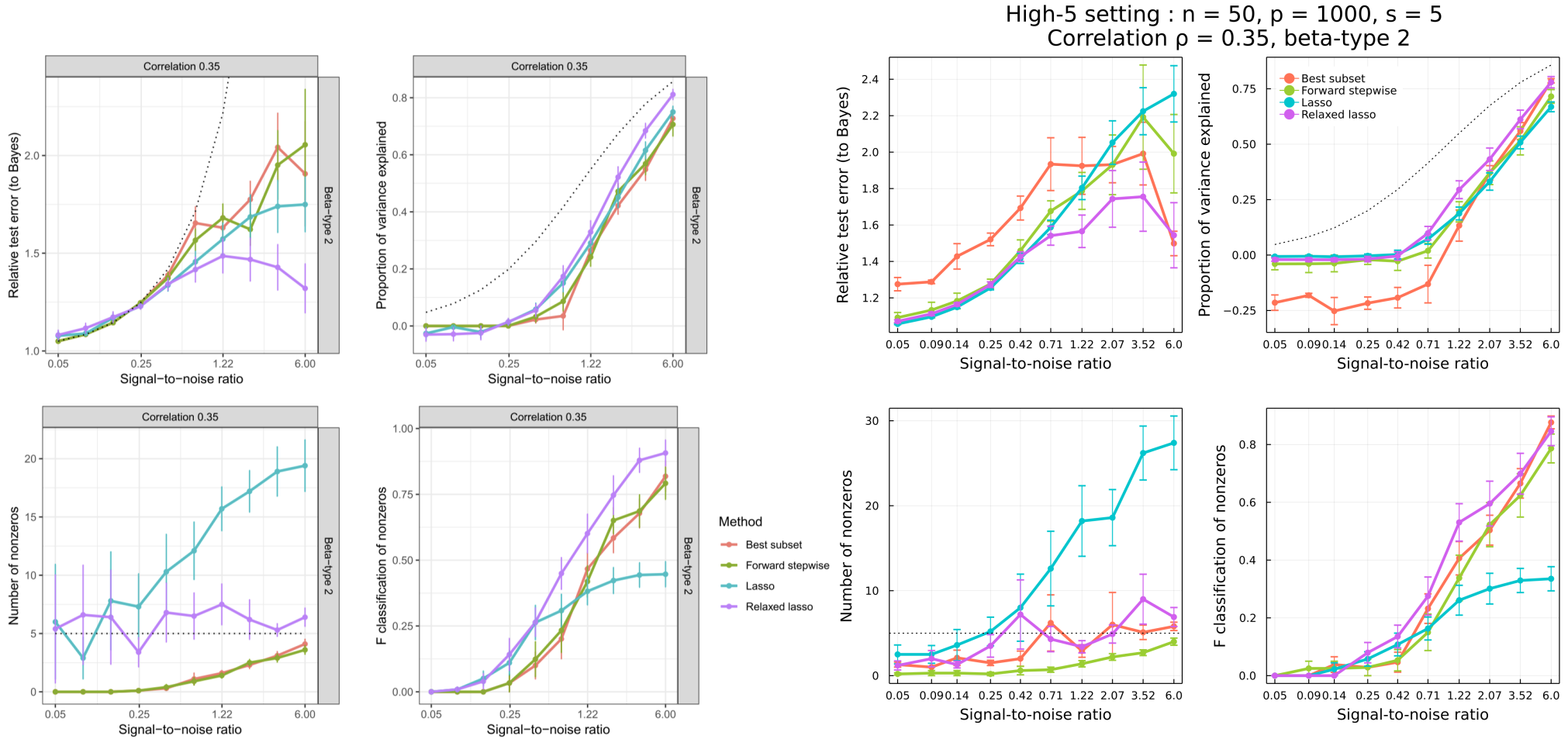


Figure 4: RTE, PVE, number of nonzero coefficients and F-score as functions of SNR with $n = 50, p = 1000, s = 5, \rho = 0.35$ and beta-type 2.

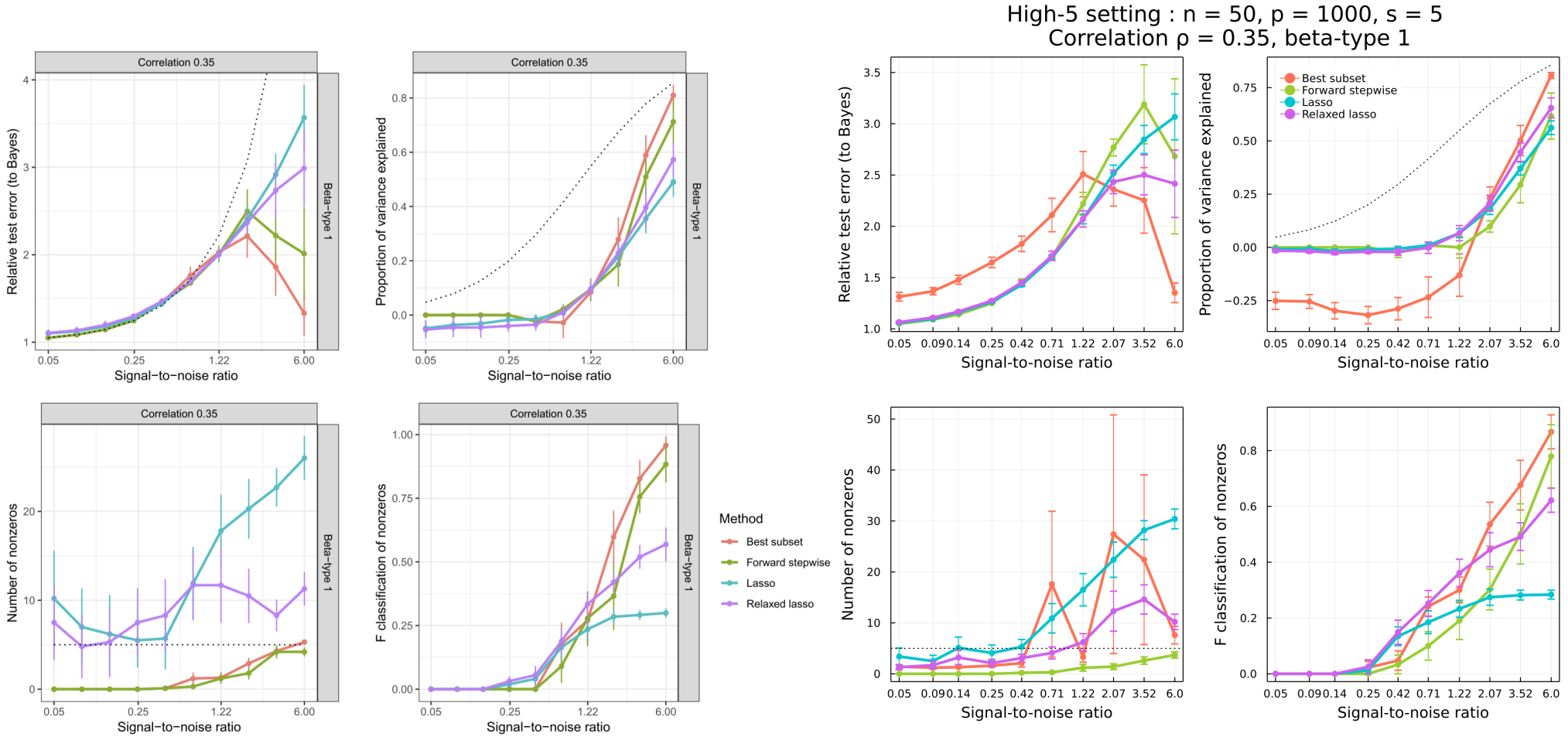


Figure 5: RTE, PVE, number of nonzero coefficients and F-score as functions of SNR with $n = 100, p = 1000, s = 10, \rho = 0.35$ and beta-type 2.

4 Summary of Results

- The lasso and relaxed lasso are very fast and forward stepwise is also fast, though not quite as fast as the lasso.
- In high-5 and high-10 setting, best subset selection gives very poor accuracy because of time-limit.
- Forward stepwise selection and best subset selection perform quite similarly over all settings, but the former one is much faster. This does not agree with the results for forward stepwise in Bertsimas et al (2016).
- In the low SNR range, the lasso outperforms the best subset selection while it has worse accuracy than best subset selection in the high SNR range.
- The relaxed lasso performs better than other methods over all settings.

References

[1] Trevor Hastie, Robert Tibshirani, and Ryan Tibshirani. Best Subset, Forward Stepwise or Lasso? Analysis and Recommendations Based on Extensive Comparisons. *Statistical Science*, 35(4):579 – 592, 2020.

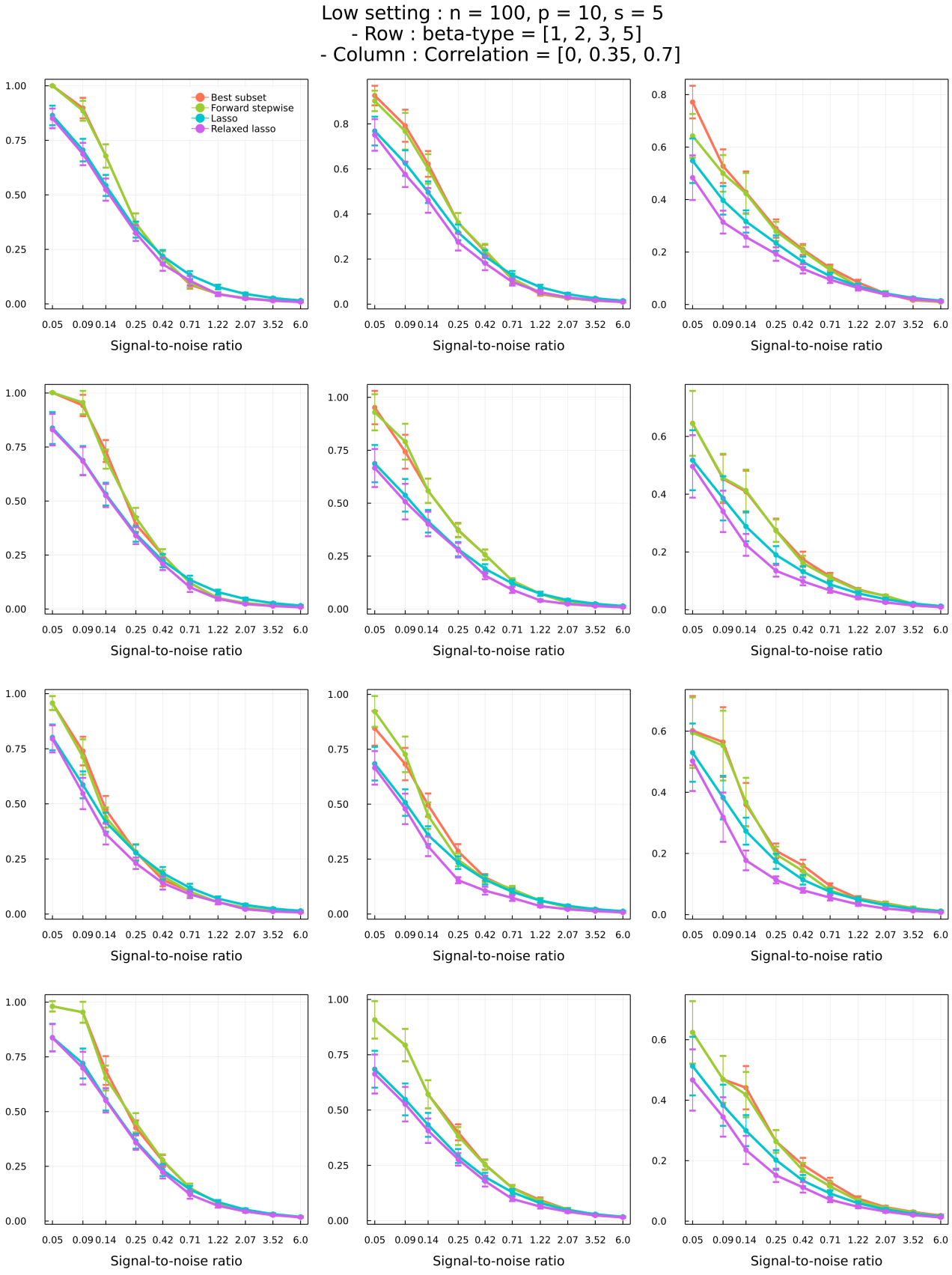
[2] Dimitris Bertsimas, Angela King, and Rahul Mazumder. Best subset selection via a modern optimization lens, 2015.

[3] Robert Tibshirani, Jacob Bien, Jerome Friedman, Trevor Hastie, Noah Simon, Jonathan Taylor, and Ryan J. Tibshirani. Strong rules for discarding predictors in lasso-type problems, 2010.

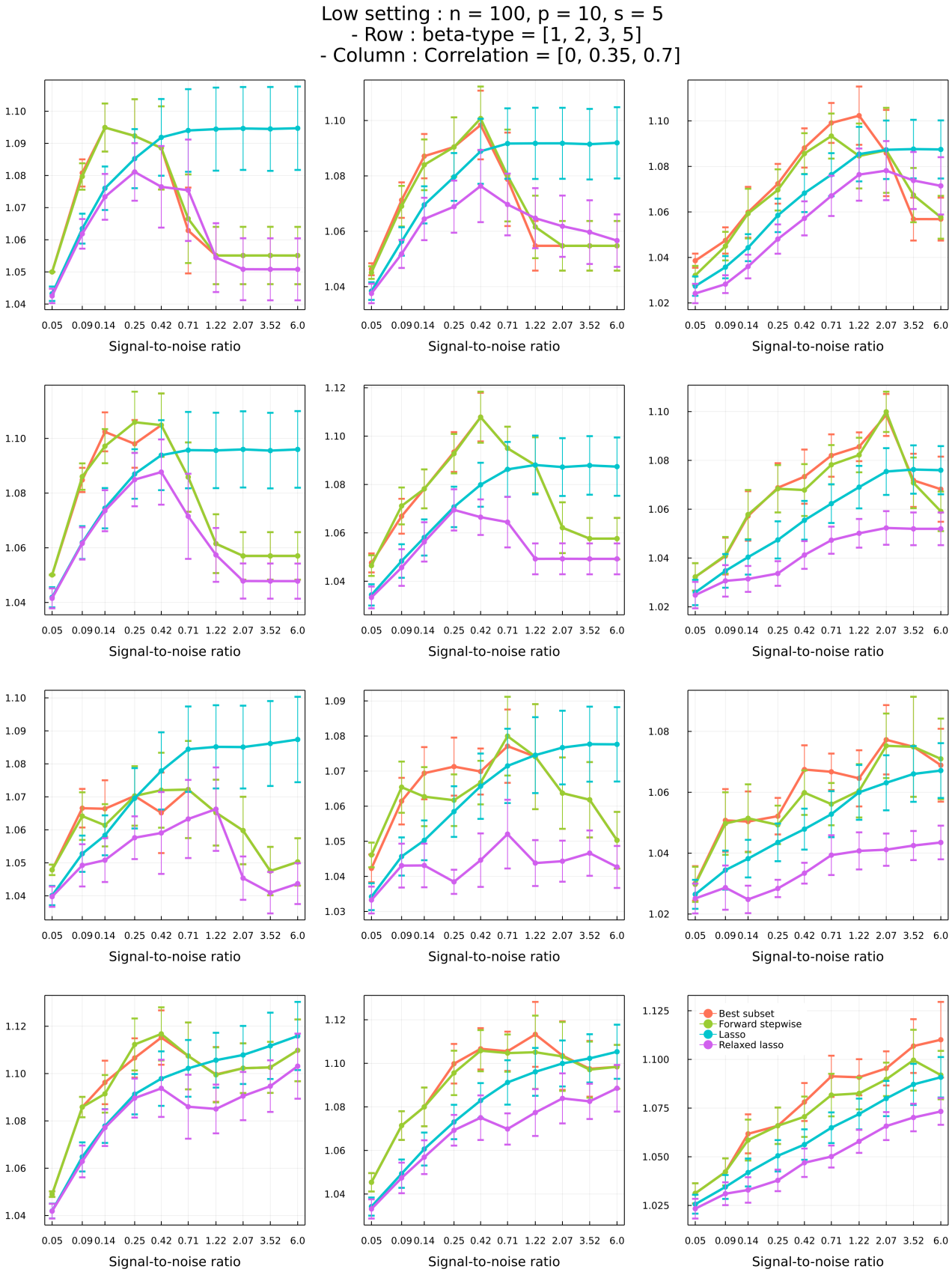
[4] Jerome Friedman, Trevor Hastie, Holger Höfling, and Robert Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302 – 332, 2007.

Appendix

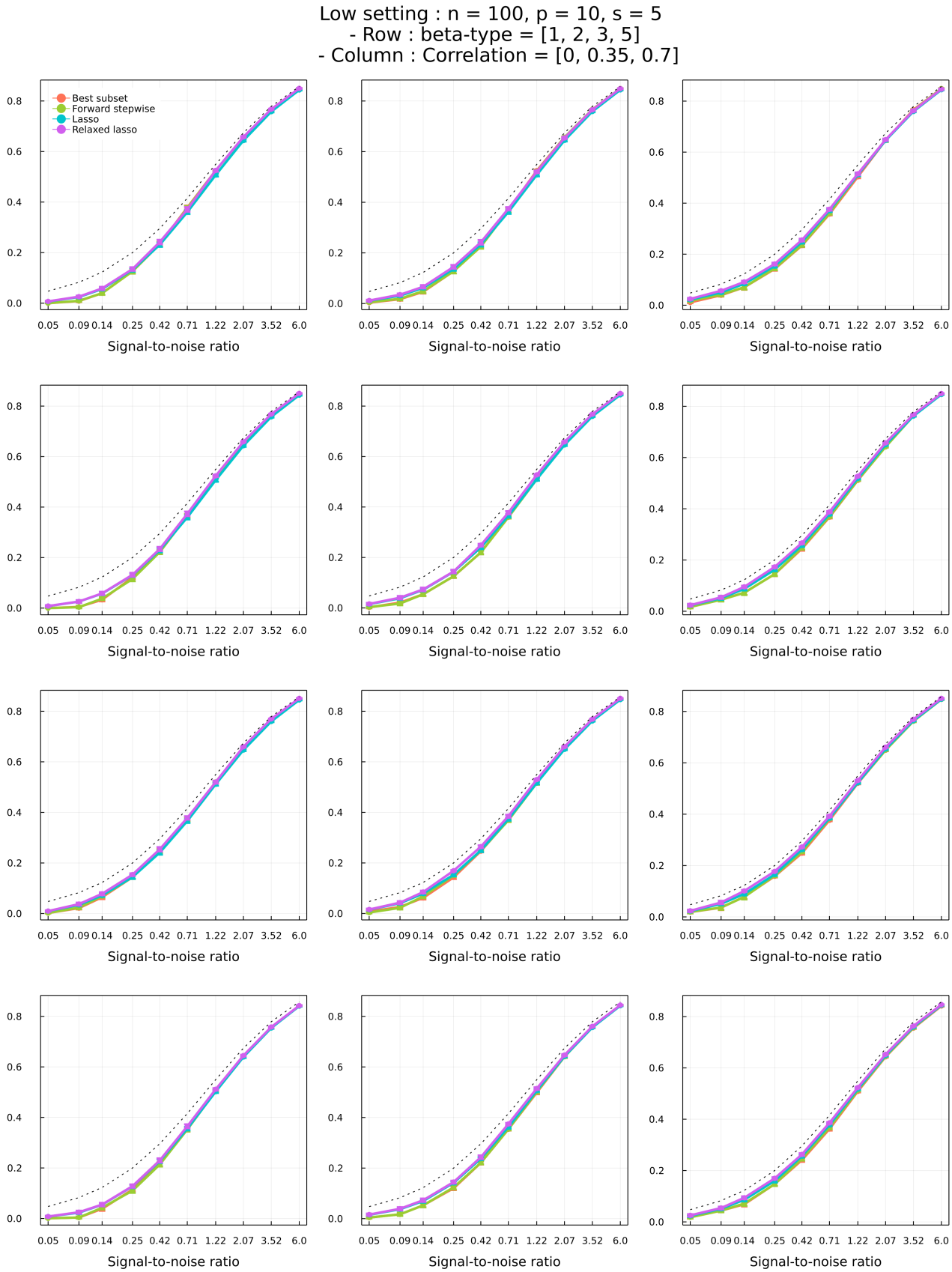
Relative risk (to null model)



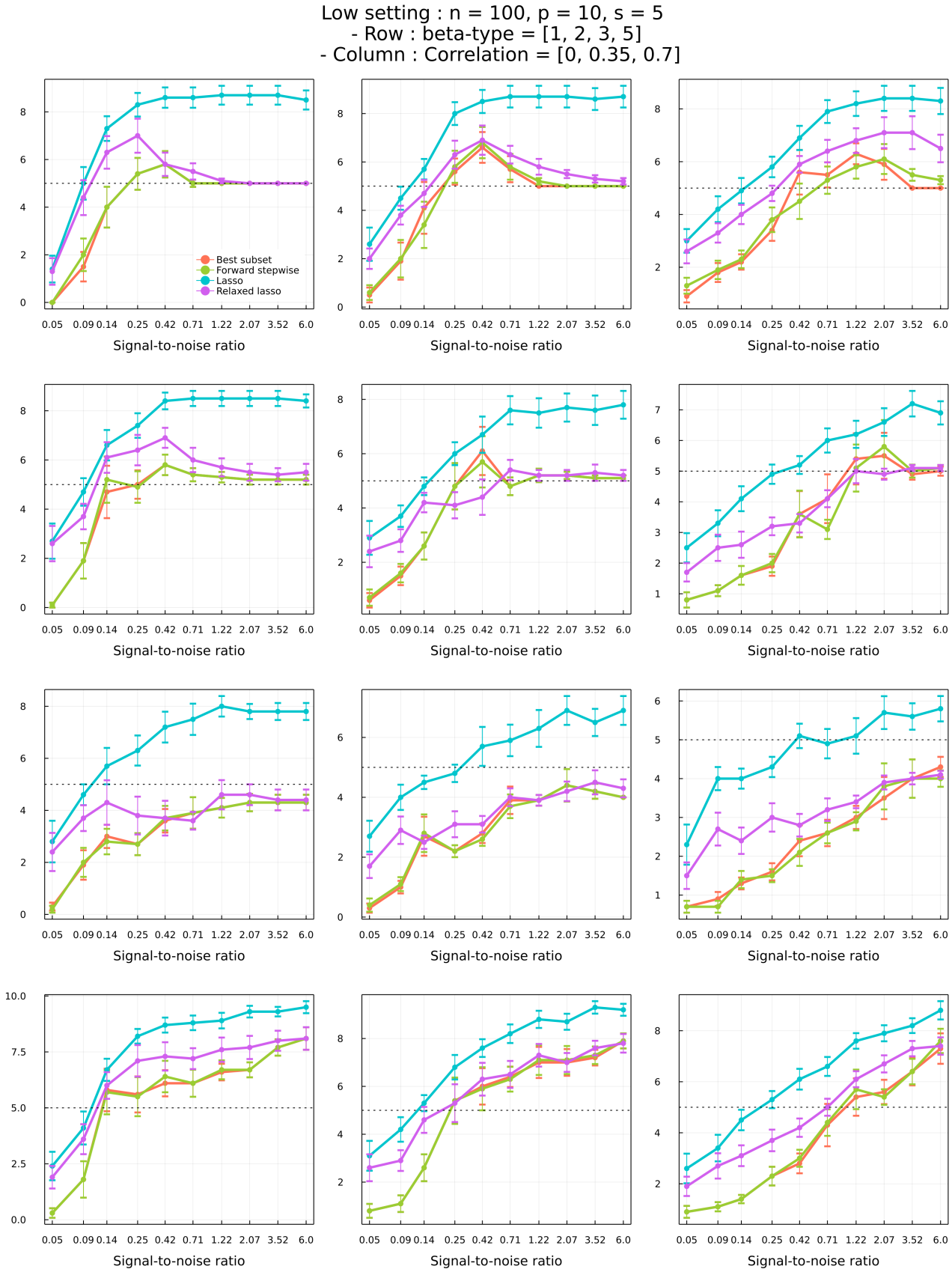
Relative test error (to Bayes)



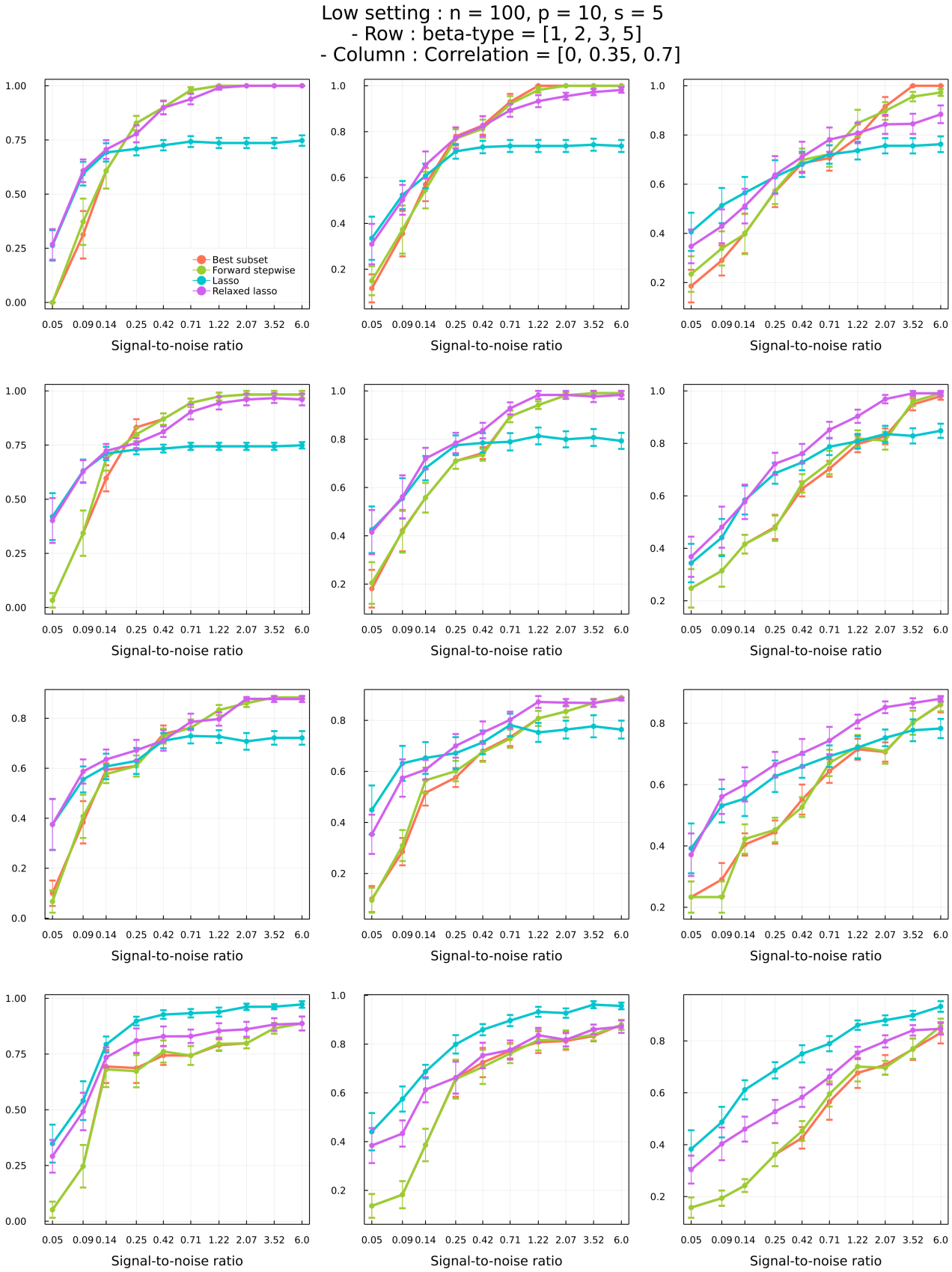
Proportion of variance explained



Number of nonzero coefficients



F-score


Low setting : n = 100, p = 10, s = 5
- Row : beta-type = [1, 2, 3, 5]
- Column : Correlation = [0, 0.35, 0.7]
Best subset
Forward stepwise
Lasso
Relaxed lasso
0.05 0.09 0.14 0.25 0.42 0.71 1.22 2.07 3.52 6.0
Signal-to-noise ratio