Penalized Regression Methods for Interaction and Mixed-Effects Models with Applications to Genomic and Brain Imaging Data

Sahir Rai Bhatnagar

Doctor of Philosophy

Department of Epidemiology, Biostatistics and Occupational Health

McGill University
Montréal, Québec, Canada
July 2018

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Doctor of Philosophy

Chapter 1

A General Framework for Variable Selection in Linear Mixed Models with Applications to Genetic Studies with Structured Populations

Sahir Rai Bhatnagar[1,2], Karim Oualkacha[3], Yi Yang[4], Marie Forest[2], Celia MT Greenwood[1,2]

[1]Department of Epidemiology, Biostatistics and Occupational Health, McGill University

[2]Lady Davis Institute, Jewish General Hospital, Montréal, QC

[3]Département de Mathématiques, Université de Québec À Montréal

[4]Department of Mathematics and Statistics, McGill University

Abstract

Complex traits are known to be influenced by a combination of environmental factors and rare and common genetic variants. However, detection of such multivariate associations can be compromised by low statistical power and confounding by population structure. Linear mixed effect models (LMM) can account for correlations due to relatedness but have not been applicable in high-dimensional (HD) settings where the number of fixed effect predictors greatly exceeds the number of samples. False positives can result from two-stage approaches, where the residuals estimated from a null model adjusted for the subjects' relationship structure are subsequently used as the response in a standard penalized regression model. To overcome these challenges, we develop a general penalized LMM framework called `ggmix` that simultaneously, in one step, selects variables and estimates their effects, while accounting for between individual correlations. Our method can accommodate several sparsity-inducing penalties such as the lasso, elastic net and group lasso, and also readily handles prior annotation information in the form of weights. We develop a blockwise coordinate descent algorithm which is highly scalable, computationally efficient and has theoretical guarantees of convergence. Through simulations, we show that `ggmix` leads to correct Type 1 error control and improved variance component estimation compared to the two-stage approach or principal component adjustment. `ggmix` is also robust to different kinship structures and heritability proportions. Our algorithms are available in an R package (https://github.com/greenwoodlab).

1.1 Introduction

Genome-wide association studies (GWAS) have become the standard method for analyzing genetic datasets owing to their success in identifying thousands of genetic variants associated with complex diseases (https://www.genome.gov/gwastudies/). Despite these impressive findings, the discovered markers have only been able to explain a small proportion of the phenotypic variance; this is known as the missing heritability problem (Manolio et al., 2009). One plausible explanation is that there are many causal variants that each explain a small amount of variation with small effect sizes (J. Yang et al., 2010). Methods such GWAS, which test each variant or single nucleotide polymorphism (SNP) independently, may miss these true associations due to the stringent significance thresholds required to reduce the number of false positives (Manolio et al., 2009). Another major issue to overcome is that of confounding due to geographic population structure, family and/or cryptic relatedness which can lead to spurious associations (Astle et al., 2009). For example, there may be subpopulations within a study that differ with respect to their genotype frequencies at a particular locus due to geographical location or their ancestry. This heterogeneity in genotype frequency can cause correlations with other loci and consequently mimic the signal of association even though there is no biological association (Marchini et al., 2004; Song et al., 2015). Studies that separate their sample by ethnicity to address this confounding suffer from a loss in statistical power.

To address the first problem, multivariable regression methods have been proposed which simultaneously fit many SNPs in a single model (Hoggart et al., 2008; Li et al., 2010). Indeed, the power to detect an association for a given SNP may be increased when other causal SNPs have been accounted for. Conversely, a stronger signal from a causal SNP may weaken false signals when modeled jointly (Hoggart et al., 2008).

Solutions for confounding by population structure have also received significant attention in the literature (Eu-Ahsunthornwattana et al., 2014; Kang et al., 2010; Lippert et al., 2011;

Yu et al., 2006). There are two main approaches to account for the relatedness between subjects: 1) the principal component (PC) adjustment method and 2) the linear mixed model (LMM). The PC adjustment method includes the top PCs of genome-wide SNP genotypes as additional covariates in the model (Price et al., 2006). The LMM uses an estimated covariance matrix from the individuals' genotypes and includes this information in the form of a random effect Astle et al. (2009).

While these problems have been addressed in isolation, there has been relatively little progress towards addressing them jointly at a large scale. Region-based tests of association have been developed where a linear combination of p variants is regressed on the response variable in a mixed model framework (Oualkacha et al., 2013). In case-control data, a stepwise logistic-regression procedure was used to evaluate the relative importance of variants within a small genetic region (Cordell & Clayton, 2002). These methods however are not applicable in the high-dimensional setting, i.e., when the number of variables p is much larger than the sample size n, as is often the case in genetic studies where millions of variants are measured on thousands of individuals.

There has been recent interest in using penalized linear mixed models, which place a constraint on the magnitude of the effect sizes while controlling for confounding factors such as population structure. For example, the LMM-lasso (Rakitsch et al., 2013) places a Laplace prior on all main effects while the adaptive mixed lasso (Wang et al., 2011) uses the L_1 penalty (Tibshirani, 1996) with adaptively chosen weights (Zou, 2006) to allow for differential shrinkage amongst the variables in the model. Another method applied a combination of both the lasso and group lasso penalties in order to select variants within a gene most associated with the response (Ding et al., 2014). However, these methods are normally performed in two steps. First, the variance components are estimated once from a LMM with a single random effect. These LMMs normally use the estimated covariance matrix from the individuals' genotypes to account for the relatedness but assumes no SNP main effects

(i.e. a null model). The residuals from this null model with a single random effect can be treated as independent observations because the relatedness has been effectively removed from the original response. In the second step, these residuals are used as the response in any high-dimensional model that assumes uncorrelated errors. This approach has both computational and practical advantages since existing penalized regression software such as `glmnet` (Friedman et al., 2010) and `gglasso` (Y. Yang & Zou, 2015), which assume independent observations, can be applied directly to the residuals. However, recent work has shown that there can be a loss in power if a causal variant is included in the calculation of the covariance matrix as its effect will have been removed in the first step (Oualkacha et al., 2013; J. Yang et al., 2014).

In this paper we develop a general penalized LMM framework called `ggmix` that simultaneously selects variables and estimates their effects, accounting for between-individual correlations. Our method can accommodate several sparsity inducing penalties such as the lasso (Tibshirani, 1996), elastic net (Zou & Hastie, 2005) and group lasso (Yuan & Lin, 2006). `ggmix` also readily handles prior annotation information in the form of a penalty factor, which can be useful, for example, when dealing with rare variants. We develop a blockwise coordinate descent algorithm which is highly scalable and has theoretical guarantees of convergence to a stationary point. All of our algorithms are implemented in the `ggmix` R package hosted on GitHub with extensive documentation (`http://sahirbhatnagar.com/ggmix/`). We provide a brief demonstration of the `ggmix` package in Appendix **??**.

The rest of the paper is organized as follows. Section 2 describes the `ggmix` model. Section 3 contains the optimization procedure and the algorithm used to fit the `ggmix` model. In Section 4, we compare the performance of our proposed approach and demonstrate the scenarios where it can be advantageous to use over existing methods through simulation studies. Section 5 discusses some limitations and future directions.

1.2 Penalized Linear Mixed Models

1.2.1 Model Set-up

Let $i = 1, \ldots, N$ be a grouping index, $j = 1, \ldots, n_i$ the observation index within a group and $N_T = \sum_{i=1}^{N} n_i$ the total number of observations. For each group let $\boldsymbol{y}_i = (y_1, \ldots, y_{n_i})$ be the observed vector of responses or phenotypes, $\mathbf{X}_i$ an $n_i \times (p+1)$ design matrix (with the column of 1s for the intercept), $\boldsymbol{b}_i$ a group-specific random effect vector of length n_i and $\boldsymbol{\varepsilon}_i = (\varepsilon_{i1}, \ldots, \varepsilon_{in_i})$ the individual error terms. Denote the stacked vectors $\mathbf{Y} = (\boldsymbol{y}_i, \ldots, \boldsymbol{y}_N)^T \in \mathbb{R}^{N_T \times 1}$, $\boldsymbol{b} = (\boldsymbol{b}_i, \ldots, \boldsymbol{b}_N)^T \in \mathbb{R}^{N_T \times 1}$, $\boldsymbol{\varepsilon} = (\boldsymbol{\varepsilon}_i, \ldots, \boldsymbol{\varepsilon}_N)^T \in \mathbb{R}^{N_T \times 1}$, and the stacked matrix
$\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_N)^T \in \mathbb{R}^{N_T \times (p+1)}$. Furthermore, let $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^T \in \mathbb{R}^{(p+1)\times 1}$ be a vector of fixed effects regression coefficients corresponding to $\mathbf{X}$. We consider the following linear mixed model with a single random effect (Pirinen et al., 2013):

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{b} + \boldsymbol{\varepsilon} \tag{1.1}$$

where the random effect $\boldsymbol{b}$ and the error variance $\boldsymbol{\varepsilon}$ are assigned the distributions

$$\boldsymbol{b} \sim \mathcal{N}(0, \eta\sigma^2\boldsymbol{\Phi}) \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, (1-\eta)\sigma^2\mathbf{I}) \tag{1.2}$$

Here, $\boldsymbol{\Phi}_{N_T \times N_T}$ is a known positive semi-definite and symmetric covariance or kinship matrix calculated from SNPs sampled across the genome, $\mathbf{I}_{N_T \times N_T}$ is the identity matrix and parameters σ^2 and $\eta \in [0, 1]$ determine how the variance is divided between $\boldsymbol{b}$ and $\boldsymbol{\varepsilon}$. Note that η is also the narrow-sense heritability (h^2), defined as the proportion of phenotypic variance attributable to the additive genetic factors (Manolio et al., 2009). The joint density of $\mathbf{Y}$ is

therefore multivariate normal:

$$\mathbf{Y}|(\boldsymbol{\beta}, \eta, \sigma^2) \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \eta\sigma^2\boldsymbol{\Phi} + (1-\eta)\sigma^2\mathbf{I}) \tag{1.3}$$

The LMM-Lasso method (Rakitsch et al., 2013) considers an alternative but equivalent parameterization given by:

$$\mathbf{Y}|(\boldsymbol{\beta}, \delta, \sigma_g^2) \sim \mathcal{N}(\mathbf{X}\boldsymbol{\beta}, \sigma_g^2(\boldsymbol{\Phi} + \delta\mathbf{I})) \tag{1.4}$$

where $\delta = \sigma_e^2/\sigma_g^2$, σ_g^2 is the genetic variance and σ_e^2 is the residual variance. We instead consider the parameterization in (1.3) since maximization is easier over the compact set $\eta \in [0, 1]$ than over the unbounded interval $\delta \in [0, \infty)$ (Pirinen et al., 2013). We define the complete parameter vector as $\boldsymbol{\Theta} := (\boldsymbol{\beta}, \eta, \sigma^2)$. The negative log-likelihood for (1.3) is given by

$$-\ell(\boldsymbol{\Theta}) \propto \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\log\left(\det(\mathbf{V})\right) + \frac{1}{2\sigma^2}\left(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\right)^T \mathbf{V}^{-1}\left(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\right) \tag{1.5}$$

where $\mathbf{V} = \eta\boldsymbol{\Phi} + (1-\eta)\mathbf{I}$ and $\det(\mathbf{V})$ is the determinant of $\mathbf{V}$.

Let $\boldsymbol{\Phi} = \mathbf{U}\mathbf{D}\mathbf{U}^T$ be the eigen (spectral) decomposition of the kinship matrix $\boldsymbol{\Phi}$, where $\mathbf{U}_{N_T \times N_T}$ is an orthonormal matrix of eigenvectors (i.e. $\mathbf{U}\mathbf{U}^T = \mathbf{I}$) and $\mathbf{D}_{N_T \times N_T}$ is a diagonal matrix of eigenvalues Λ_i. $\mathbf{V}$ can then be further simplified (Pirinen et al., 2013)

$$\begin{aligned}
\mathbf{V} &= \eta\boldsymbol{\Phi} + (1-\eta)\mathbf{I} \\
&= \eta\mathbf{U}\mathbf{D}\mathbf{U}^T + (1-\eta)\mathbf{U}\mathbf{I}\mathbf{U}^T \\
&= \mathbf{U}\eta\mathbf{D}\mathbf{U}^T + \mathbf{U}(1-\eta)\mathbf{I}\mathbf{U}^T \\
&= \mathbf{U}\left(\eta\mathbf{D} + (1-\eta)\mathbf{I}\right)\mathbf{U}^T \\
&= \mathbf{U}\widetilde{\mathbf{D}}\mathbf{U}^T
\end{aligned} \tag{1.6}$$

where

$$\widetilde{\mathbf{D}} = \eta\mathbf{D} + (1-\eta)\mathbf{I} \tag{1.7}$$

$$= \eta \begin{bmatrix} \Lambda_1 & & & \\ & \Lambda_2 & & \\ & & \ddots & \\ & & & \Lambda_{N_T} \end{bmatrix} + (1-\eta) \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1+\eta(\Lambda_1-1) & & & \\ & 1+\eta(\Lambda_2-1) & & \\ & & \ddots & \\ & & & 1+\eta(\Lambda_{N_T}-1) \end{bmatrix}$$

$$= \operatorname{diag}\left\{1+\eta(\Lambda_1-1), 1+\eta(\Lambda_2-1), \ldots, 1+\eta(\Lambda_{N_T}-1)\right\} \tag{1.8}$$

Since (1.7) is a diagonal matrix, its inverse is also a diagonal matrix:

$$\widetilde{\mathbf{D}}^{-1} = \operatorname{diag}\left\{\frac{1}{1+\eta(\Lambda_1-1)}, \frac{1}{1+\eta(\Lambda_2-1)}, \ldots, \frac{1}{1+\eta(\Lambda_{N_T}-1)}\right\} \tag{1.9}$$

From (1.6) and (1.8), $\log(\det(\mathbf{V}))$ simplifies to

$$\begin{aligned} \log(\det(\mathbf{V})) &= \log\left(\det(\mathbf{U})\det\left(\widetilde{\mathbf{D}}\right)\det(\mathbf{U}^T)\right) \\ &= \log\left\{\prod_{i=1}^{N_T}(1+\eta(\Lambda_i-1))\right\} \\ &= \sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i-1)) \end{aligned} \tag{1.10}$$

since $\det(\mathbf{U}) = 1$. It also follows from (1.6) that

$$\begin{aligned}\mathbf{V}^{-1} &= \left(\mathbf{U}\widetilde{\mathbf{D}}\mathbf{U}^T\right)^{-1} \\ &= \left(\mathbf{U}^T\right)^{-1}\left(\widetilde{\mathbf{D}}\right)^{-1}\mathbf{U}^{-1} \\ &= \mathbf{U}\widetilde{\mathbf{D}}^{-1}\mathbf{U}^T \end{aligned} \tag{1.11}$$

since for an orthonormal matrix $\mathbf{U}^{-1} = \mathbf{U}^T$. Substituting (1.9), (1.10) and (1.11) into (1.5) the negative log-likelihood becomes

$$-\ell(\boldsymbol{\Theta}) \propto \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i - 1)) + \frac{1}{2\sigma^2}\left(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\right)^T \mathbf{U}\widetilde{\mathbf{D}}^{-1}\mathbf{U}^T\left(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}\right) \tag{1.12}$$

$$\begin{aligned} &= \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i - 1)) + \frac{1}{2\sigma^2}\left(\mathbf{U}^T\mathbf{Y} - \mathbf{U}^T\mathbf{X}\boldsymbol{\beta}\right)^T \widetilde{\mathbf{D}}^{-1}\left(\mathbf{U}^T\mathbf{Y} - \mathbf{U}^T\mathbf{X}\boldsymbol{\beta}\right) \\ &= \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i - 1)) + \frac{1}{2\sigma^2}\left(\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\boldsymbol{\beta}\right)^T \widetilde{\mathbf{D}}^{-1}\left(\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\boldsymbol{\beta}\right) \\ &= \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i - 1)) + \frac{1}{2\sigma^2}\sum_{i=1}^{N_T}\frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p}\widetilde{X}_{ij+1}\beta_j\right)^2}{1+\eta(\Lambda_i - 1)} \end{aligned} \tag{1.13}$$

where $\widetilde{\mathbf{Y}} = \mathbf{U}^T\mathbf{Y}$, $\widetilde{\mathbf{X}} = \mathbf{U}^T\mathbf{X}$, $\widetilde{Y}_i$ denotes the i^{th} element of $\widetilde{\mathbf{Y}}$, $\widetilde{X}_{ij}$ is the i, j^{th} entry of $\widetilde{\mathbf{X}}$ and $\mathbf{1}$ is a column vector of N_T ones.

1.2.2 Penalized Maximum Likelihood Estimator

We define the $p+3$ length vector of parameters $\boldsymbol{\Theta} \coloneqq (\Theta_0, \Theta_1, \ldots, \Theta_{p+1}, \Theta_{p+2}, \Theta_{p+3}) = (\boldsymbol{\beta}, \eta, \sigma^2)$ where $\boldsymbol{\beta} \in \mathbb{R}^{p+1}, \eta \in [0,1], \sigma^2 > 0$. In what follows, $p+2$ and $p+3$ are the indices in $\boldsymbol{\Theta}$ for η and σ^2, respectively. In light of our goals to select variables associated with the response in high-dimensional data, we propose to place a constraint on the magnitude of the regression coefficients. This can be achieved by adding a penalty term to the likelihood

function (1.13). The penalty term is a necessary constraint because in our applications, the sample size is much smaller than the number of predictors. We define the following objective function:

$$Q_\lambda(\boldsymbol{\Theta}) = f(\boldsymbol{\Theta}) + \lambda \sum_{j \neq 0} v_j P_j(\beta_j) \tag{1.14}$$

where $f(\boldsymbol{\Theta}) \coloneqq -\ell(\boldsymbol{\Theta})$ is defined in (1.13), $P_j(\cdot)$ is a penalty term on the fixed regression coefficients $\beta_1, \ldots, \beta_{p+1}$ (we do not penalize the intercept) controlled by the nonnegative regularization parameter λ, and v_j is the penalty factor for jth covariate. These penalty factors serve as a way of allowing parameters to be penalized differently. Note that we do not penalize η or σ^2. An estimate of the regression parameters $\widehat{\boldsymbol{\Theta}}_\lambda$ is obtained by

$$\widehat{\boldsymbol{\Theta}}_\lambda = \arg\min_{\boldsymbol{\Theta}} Q_\lambda(\boldsymbol{\Theta}) \tag{1.15}$$

This is the general set-up for our model. In Section 1.3 we provide more specific details on how we solve (1.15).

1.3 Computational Algorithm

We use a general purpose block coordinate gradient descent algorithm (CGD) (Tseng & Yun, 2009) to solve (1.15). At each iteration, we cycle through the coordinates and minimize the objective function with respect to one coordinate only. For continuously differentiable $f(\cdot)$ and convex and block-separable $P(\cdot)$ (i.e. $P(\boldsymbol{\beta}) = \sum_i P_i(\beta_i)$), Tseng and Yun Tseng & Yun (2009) show that the solution generated by the CGD method is a stationary point of $Q_\lambda(\cdot)$ if the coordinates are updated in a Gauss-Seidel manner i.e. $Q_\lambda(\cdot)$ is minimized with respect to one parameter while holding all others fixed. The CGD algorithm has been successfully applied in fixed effects models (e.g. Meier et al. (2008), Friedman et al. (2010)) and linear mixed models with an ℓ_1 penalty Schelldorfer et al. (2011). In the next section we provide some brief details about Algorithm 1. A more thorough treatment of the algorithm is given

in Appendix **??**.

We emphasize here that previously developed methods such as the LMM-lasso (Rakitsch et al., 2013) use a two-stage fitting procedure without any convergence details. From a practical point of view, there is currently no implementation that provides a principled way of determining the sequence of tuning parameters to fit, nor a procedure that automatically selects the optimal value of λ. To our knowledge, we are the first to develop a CGD algorithm in the specific context of fitting a penalized LMM for population structure correction with theoretical guarantees of convergence. Furthermore, we develop a principled method for automatic tuning parameter selection and provide an easy-to-use software implementation in order to promote wider uptake of these more complex methods by applied practitioners.

Algorithm 1: Block Coordinate Gradient Descent

Set the iteration counter $k \leftarrow 0$, initial values for the parameter vector $\boldsymbol{\Theta}^{(0)}$ and convergence threshold ϵ;
for $\lambda \in \{\lambda_{max}, \ldots, \lambda_{min}\}$ **do**
  repeat

$$\boldsymbol{\beta}^{(k+1)} \leftarrow \arg\min_{\boldsymbol{\beta}} Q_\lambda\left(\boldsymbol{\beta}, \eta^{(k)}, \sigma^{2\ (k)}\right)$$
$$\eta^{(k+1)} \leftarrow \arg\min_{\eta} Q_\lambda\left(\boldsymbol{\beta}^{(k+1)}, \eta, \sigma^{2\ (k)}\right)$$
$$\sigma^{2\ (k+1)} \leftarrow \arg\min_{\sigma^2} Q_\lambda\left(\boldsymbol{\beta}^{(k+1)}, \eta^{(k+1)}, \sigma^2\right)$$

    $k \leftarrow k + 1$
  until *convergence criterion is satisfied:* $\left\|\boldsymbol{\Theta}^{(k+1)} - \boldsymbol{\Theta}^{(k)}\right\|_2 < \epsilon$;
end

1.3.1 Updates for the β parameter

Recall that the part of the objective function that depends on $\boldsymbol{\beta}$ has the form

$$Q_\lambda(\boldsymbol{\Theta}) = \frac{1}{2}\sum_{i=1}^{N_T} w_i \left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1}\beta_j\right)^2 + \lambda \sum_{j=1}^{p} v_j|\beta_j| \tag{1.16}$$

where

$$w_i \coloneqq \frac{1}{\sigma^2\left(1+\eta(\Lambda_i-1)\right)} \tag{1.17}$$

Conditional on $\eta^{(k)}$ and $\sigma^{2\,(k)}$, it can be shown that the solution for β_j, $j=1,\ldots,p$ is given by

$$\beta_j^{(k+1)} \leftarrow \frac{\mathcal{S}_\lambda\left(\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}\left(\widetilde{Y}_i - \sum_{\ell\neq j}\widetilde{X}_{i\ell}\beta_\ell^{(k)}\right)\right)}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{ij}^2} \tag{1.18}$$

where $\mathcal{S}_\lambda(x)$ is the soft-thresholding operator

$$\mathcal{S}_\lambda(x) = \text{sign}(x)(|x|-\lambda)_+$$

$\text{sign}(x)$ is the signum function

$$\text{sign}(x) = \begin{cases} -1 & x<0 \\ 0 & x=0 \\ 1 & x>0 \end{cases}$$

and $(x)_+ = \max(x,0)$. We provide the full derivation in Appendix **??**.

1.3.2 Updates for the η paramter

Given $\boldsymbol{\beta}^{(k+1)}$ and $\sigma^{2\,(k)}$, solving for $\eta^{(k+1)}$ becomes a univariate optimization problem:

$$\eta^{(k+1)} \leftarrow \arg\min_{\eta} \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i-1)) + \frac{1}{2\sigma^{2\,(k)}}\sum_{i=1}^{N_T}\frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p}\widetilde{X}_{ij+1}\beta_j^{(k+1)}\right)^2}{1+\eta(\Lambda_i-1)} \tag{1.19}$$

We use a bound constrained optimization algorithm (Byrd et al., 1995) implemented in the `optim` function in `R` and set the lower and upper bounds to be 0.01 and 0.99, respec-

tively.

1.3.3 Updates for the σ^2 parameter

Conditional on $\boldsymbol{\beta}^{(k+1)}$ and $\eta^{(k+1)}$, ${\sigma^2}^{(k+1)}$ can be solved for using the following equation:

$$ {\sigma^2}^{(k+1)} \leftarrow \arg\min_{\sigma^2} \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{N_T}\frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p}\widetilde{X}_{ij+1}\beta_j\right)^2}{1+\eta(\Lambda_i - 1)} \tag{1.20} $$

There exists an analytic solution for (1.20) given by:

$$ {\sigma^2}^{(k+1)} \leftarrow \frac{1}{N_T}\sum_{i=1}^{N_T}\frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p}\widetilde{X}_{ij+1}\beta_j^{(k+1)}\right)^2}{1+\eta^{(k+1)}(\Lambda_i - 1)} \tag{1.21} $$

1.3.4 Regularization path

In this section we describe how determine the sequence of tuning parameters λ at which to fit the model. Recall that our objective function has the form

$$ Q_\lambda(\boldsymbol{\Theta}) = \frac{N_T}{2}\log(\sigma^2) + \frac{1}{2}\sum_{i=1}^{N_T}\log(1+\eta(\Lambda_i - 1)) + \frac{1}{2}\sum_{i=1}^{N_T} w_i\left(\widetilde{Y}_i - \sum_{j=0}^{p}\widetilde{X}_{ij+1}\beta_j\right)^2 + \lambda\sum_{j=1}^{p} v_j|\beta_j| \tag{1.22} $$

The Karush-Kuhn-Tucker (KKT) optimality conditions for (1.22) are given by:

$$ \begin{aligned} \frac{\partial}{\partial\beta_1,\ldots,\beta_p}Q_\lambda(\boldsymbol{\Theta}) &= \mathbf{0}_p \\ \frac{\partial}{\partial\beta_0}Q_\lambda(\boldsymbol{\Theta}) &= 0 \\ \frac{\partial}{\partial\eta}Q_\lambda(\boldsymbol{\Theta}) &= 0 \\ \frac{\partial}{\partial\sigma^2}Q_\lambda(\boldsymbol{\Theta}) &= 0 \end{aligned} \tag{1.23} $$

The equations in (1.23) are equivalent to

$$\begin{aligned}
\sum_{i=1}^{N_T} w_i \widetilde{X}_{i1} \left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1} \beta_j \right) &= 0 \\
\frac{1}{v_j} \sum_{i=1}^{N_T} w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1} \beta_j \right) &= \lambda \gamma_j, \\
\gamma_j \in \begin{cases} \text{sign}(\hat{\beta}_j) & \text{if} \quad \hat{\beta}_j \neq 0 \\ [-1, 1] & \text{if} \quad \hat{\beta}_j = 0 \end{cases}, &\qquad \text{for } j = 1, \ldots, p \\
\frac{1}{2} \sum_{i=1}^{N_T} \frac{\Lambda_i - 1}{1 + \eta(\Lambda_i - 1)} \left(1 - \frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1} \beta_j \right)^2}{\sigma^2 (1 + \eta(\Lambda_i - 1))} \right) &= 0 \\
\sigma^2 - \frac{1}{N_T} \sum_{i=1}^{N_T} \frac{\left(\widetilde{Y}_i - \sum_{j=0}^{p} \widetilde{X}_{ij+1} \beta_j \right)^2}{1 + \eta(\Lambda_i - 1)} &= 0
\end{aligned} \tag{1.24}$$

where w_i is given by (1.17), $\widetilde{\mathbf{X}}_{-1}^T$ is $\widetilde{\mathbf{X}}^T$ with the first column removed, $\widetilde{\mathbf{X}}_1^T$ is the first column of $\widetilde{\mathbf{X}}^T$, and $\boldsymbol{\gamma} \in \mathbb{R}^p$ is the subgradient function of the ℓ_1 norm evaluated at $(\hat{\beta}_1, \ldots, \hat{\beta}_p)$. Therefore $\widehat{\boldsymbol{\Theta}}$ is a solution in (1.15) if and only if $\widehat{\boldsymbol{\Theta}}$ satisfies (1.24) for some γ. We can determine a decreasing sequence of tuning parameters by starting at a maximal value for $\lambda = \lambda_{max}$ for which $\hat{\beta}_j = 0$ for $j = 1, \ldots, p$. In this case, the KKT conditions in (1.24) are equivalent to

$$\begin{aligned}
\frac{1}{v_j} \sum_{i=1}^{N_T} \left| w_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \widetilde{X}_{i1} \beta_0 \right) \right| &\leq \lambda, \quad \forall j = 1, \ldots, p \\
\beta_0 &= \frac{\sum_{i=1}^{N_T} w_i \widetilde{X}_{i1} \widetilde{Y}_i}{\sum_{i=1}^{N_T} w_i \widetilde{X}_{i1}^2} \\
\frac{1}{2} \sum_{i=1}^{N_T} \frac{\Lambda_i - 1}{1 + \eta(\Lambda_i - 1)} \left(1 - \frac{\left(\widetilde{Y}_i - \widetilde{X}_{i1} \beta_0 \right)^2}{\sigma^2 (1 + \eta(\Lambda_i - 1))} \right) &= 0 \\
\sigma^2 &= \frac{1}{N_T} \sum_{i=1}^{N_T} \frac{\left(\widetilde{Y}_i - \widetilde{X}_{i1} \beta_0 \right)^2}{1 + \eta(\Lambda_i - 1)}
\end{aligned} \tag{1.25}$$

We can solve the KKT system of equations in (1.25) (with a numerical solution for η) in

order to have an explicit form of the stationary point $\widehat{\boldsymbol{\Theta}}_0 = \left\{\hat{\beta}_0, \mathbf{0}_p, \hat{\eta}, \widehat{\sigma}^2\right\}$. Once we have $\widehat{\boldsymbol{\Theta}}_0$, we can solve for the smallest value of λ such that the entire vector $(\hat{\beta}_1, \ldots, \hat{\beta}_p)$ is 0:

$$\lambda_{max} = \max_j \left\{ \left| \frac{1}{v_j} \sum_{i=1}^{N_T} \hat{w}_i \widetilde{X}_{ij} \left(\widetilde{Y}_i - \widetilde{X}_{i1} \hat{\beta}_0 \right) \right| \right\}, \quad j = 1, \ldots, p \tag{1.26}$$

Following Friedman et al. Friedman et al. (2010), we choose $\tau\lambda_{max}$ to be the smallest value of tuning parameters λ_{min}, and construct a sequence of K values decreasing from λ_{max} to λ_{min} on the log scale. The defaults are set to $K = 100$, $\tau = 0.01$ if $n < p$ and $\tau = 0.001$ if $n \geq p$.

1.3.5 Warm Starts

The way in which we have derived the sequence of tuning parameters using the KKT conditions, allows us to implement warm starts. That is, the solution $\widehat{\boldsymbol{\Theta}}$ for λ_k is used as the initial value $\boldsymbol{\Theta}^{(0)}$ for λ_{k+1}. This strategy leads to computational speedups and has been implemented in the `ggmix` R package.

1.3.6 Prediction of the random effects

We use an empirical Bayes approach (e.g. Wakefield (2013)) to predict the random effects $\boldsymbol{b}$. Let the maximum a posteriori (MAP) estimate be defined as

$$\widehat{\boldsymbol{b}} = \arg\max_{\boldsymbol{b}} f(\boldsymbol{b}|\mathbf{Y}, \boldsymbol{\beta}, \eta, \sigma^2) \tag{1.27}$$

where, by using Bayes rule, $f(\boldsymbol{b}|\mathbf{Y}, \boldsymbol{\beta}, \eta, \sigma^2)$ can be expressed as

$$\begin{aligned} f(\boldsymbol{b}|\mathbf{Y}, \boldsymbol{\beta}, \eta, \sigma^2) &= \frac{f(\mathbf{Y}|\boldsymbol{b}, \boldsymbol{\beta}, \eta, \sigma^2)\pi(\boldsymbol{b}|\eta, \sigma^2)}{f(\mathbf{Y}|\boldsymbol{\beta}, \eta, \sigma^2)} \\ &\propto f(\mathbf{Y}|\boldsymbol{b}, \boldsymbol{\beta}, \eta, \sigma^2)\pi(\boldsymbol{b}|\eta, \sigma^2) \\ &\propto \exp\left\{-\frac{1}{2\sigma^2}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b})^T\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b}) - \frac{1}{2\eta\sigma^2}\boldsymbol{b}^T\boldsymbol{\Phi}^{-1}\boldsymbol{b}\right\} \\ &= \exp\left\{-\frac{1}{2\sigma^2}\left[(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b})^T\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b}) + \frac{1}{\eta}\boldsymbol{b}^T\boldsymbol{\Phi}^{-1}\boldsymbol{b}\right]\right\} \end{aligned} \tag{1.28}$$

Solving for (1.27) is equivalent to minimizing the exponent in (1.28):

$$\widehat{\boldsymbol{b}} = \arg\min_{\boldsymbol{b}}\left\{(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b})^T\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b}) + \frac{1}{\eta}\boldsymbol{b}^T\boldsymbol{\Phi}^{-1}\boldsymbol{b}\right\} \tag{1.29}$$

Taking the derivative of (1.29) with respect to $\boldsymbol{b}$ and setting it to 0 we get:

$$\begin{aligned} 0 &= -2\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta} - \boldsymbol{b}) + \frac{2}{\eta}\boldsymbol{\Phi}^{-1}\boldsymbol{b} \\ &= -\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \left(\mathbf{V}^{-1} + \frac{1}{\eta}\boldsymbol{\Phi}^{-1}\right)\boldsymbol{b} \\ \widehat{\boldsymbol{b}} &= \left(\mathbf{V}^{-1} + \frac{1}{\widehat{\eta}}\boldsymbol{\Phi}^{-1}\right)^{-1}\mathbf{V}^{-1}(\mathbf{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}) \\ &= \left(\mathbf{U}\widetilde{\mathbf{D}}^{-1}\mathbf{U}^T + \frac{1}{\widehat{\eta}}\mathbf{U}\mathbf{D}^{-1}\mathbf{U}^T\right)^{-1}\mathbf{U}\widetilde{\mathbf{D}}^{-1}\mathbf{U}^T(\mathbf{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}) \\ &= \left(\mathbf{U}\left[\widetilde{\mathbf{D}}^{-1} + \frac{1}{\widehat{\eta}}\mathbf{D}^{-1}\right]\mathbf{U}^T\right)^{-1}\mathbf{U}\widetilde{\mathbf{D}}^{-1}(\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\widehat{\boldsymbol{\beta}}) \\ &= \mathbf{U}\left[\widetilde{\mathbf{D}}^{-1} + \frac{1}{\widehat{\eta}}\mathbf{D}^{-1}\right]^{-1}\mathbf{U}^T\mathbf{U}\widetilde{\mathbf{D}}^{-1}(\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\widehat{\boldsymbol{\beta}}) \end{aligned}$$

where $\mathbf{V}^{-1}$ is given by (1.11), and $(\widehat{\boldsymbol{\beta}}, \widehat{\eta})$ are the estimates obtained from Algorithm 1.

1.3.7 Choice of the optimal tuning parameter

In order to choose the optimal value of the tuning parameter λ, we use the generalized information criterion (Nishii, 1984) (GIC):

$$GIC_\lambda = -2\ell(\widehat{\boldsymbol{\beta}}, \widehat{\sigma}^2, \widehat{\eta}) + a_n \cdot \widehat{df}_\lambda \tag{1.30}$$

where $\widehat{df}_\lambda$ is the number of non-zero elements in $\widehat{\boldsymbol{\beta}}_\lambda$ (Zou et al., 2007) plus two (representing the variance parameters η and σ^2). Several authors have used this criterion for variable selection in mixed models with $a_n = \log N_T$ (Bondell et al., 2010; Schelldorfer et al., 2011), which corresponds to the BIC. We instead choose the high-dimensional BIC (Fan & Tang, 2013) given by $a_n = \log(\log(N_T)) * \log(p)$. This is the default choice in our `ggmix` R package, though the interface is flexible to allow the user to select their choice of a_n.

Appendix A

Supplemental Methods and Simulation Results for Chapter ??

A.1 Description of Topological Overlap Matrix

Starting with a similarity measure $s_{ij} = |cor(i,j)|$ between node i and node j, one could apply a hard threshold to determine if this pair is considered connected or not resulting in an un-weighted network (a matrix of 0's and 1's). Instead, Zhang and Horvath (Zhang & Horvath, 2005) propose a soft thresholding framework that assigns a connection weight to each gene pair using a power adjacency function $a_{ij} = |s_{ij}|^{\beta}$. The parameter β determines the sensitivity and specificity of the pairwise connection strengths e.g. a larger β will result in fewer connected nodes which can reduce noise in the network but can also eliminate signal if too large. A measure of similarity is then derived using the symmetric and non-negative topological overlap matrix (Ravasz et al., 2002) (TOM) $\Omega = [\omega_{ij}]$:

$$\omega_{ij} = \frac{l_{ij} + a_{ij}}{min\{k_i, k_j\} + 1 - a_{ij}} \tag{A.1}$$

where $l_{ij} = \sum_u a_{iu}a_{uj}$, $k_i = \sum_u a_{iu}$ is the node connectivity, and the index u runs across all nodes of the network. Basically, ω_{ij} is a measure of similarity in terms of the commonality of the nodes they connect to. If i and j are unconnected and do not share any neighbors then $\omega_{ij} = 0$. An $\omega_{ij} = 1$ means that i and j are connected, and the neighbors of the node with fewer connections are also neighbors of the other node.

A.2 Binary Outcome Simulation Results

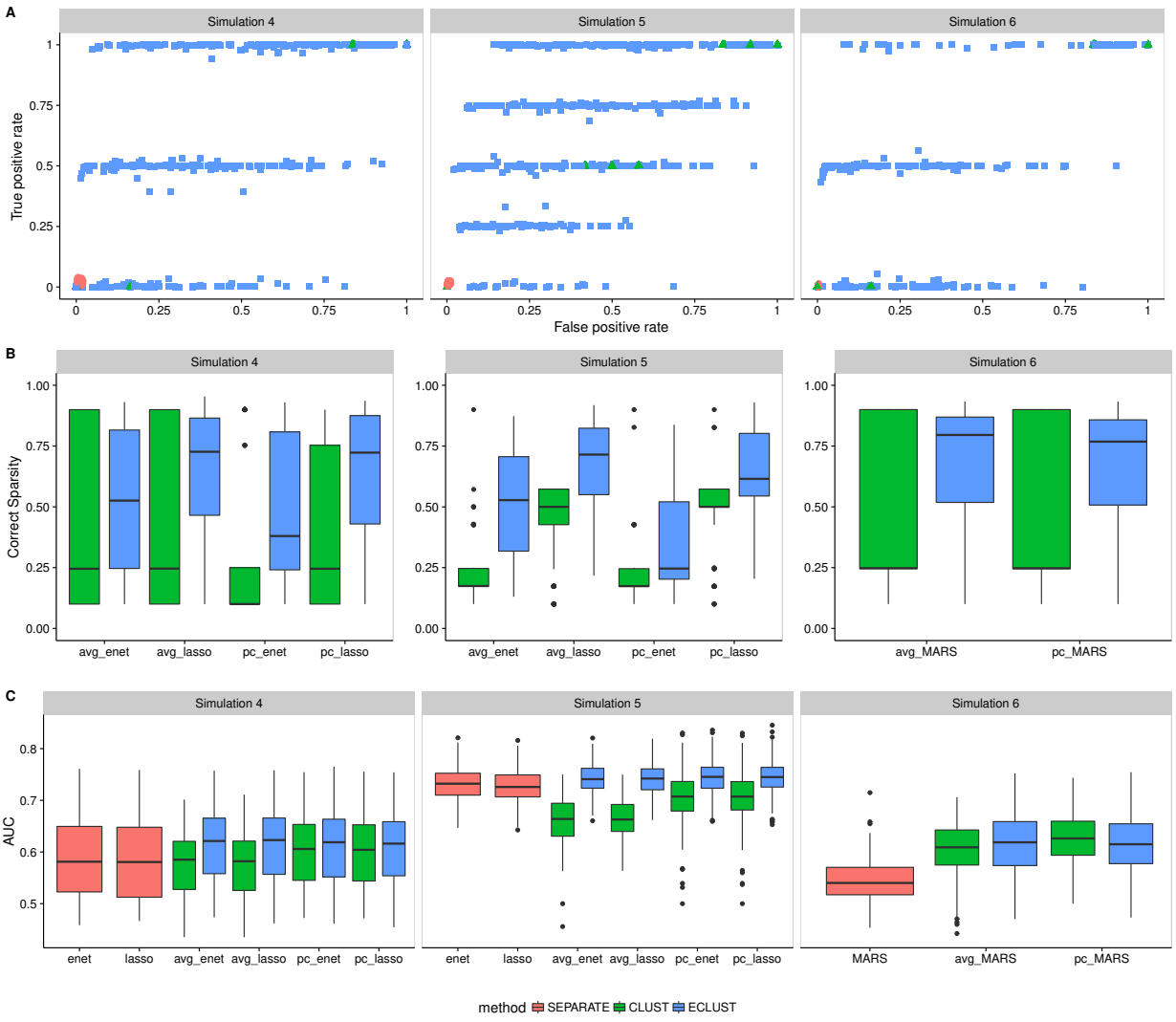


Figure A.1: Model fit results from simulations 4, 5 and 6 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

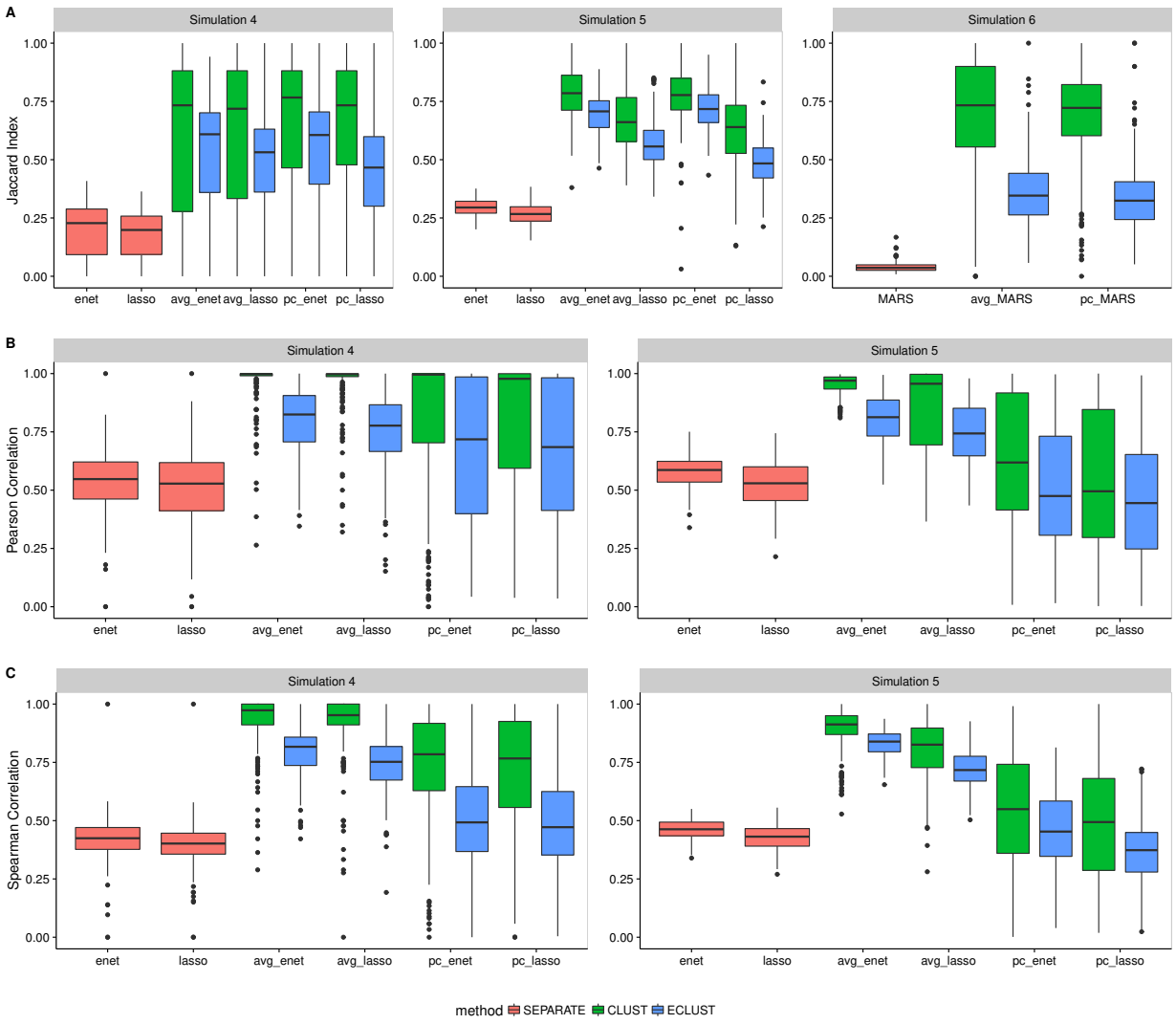


Figure A.2: Stability results from simulations 4, 5 and 6 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

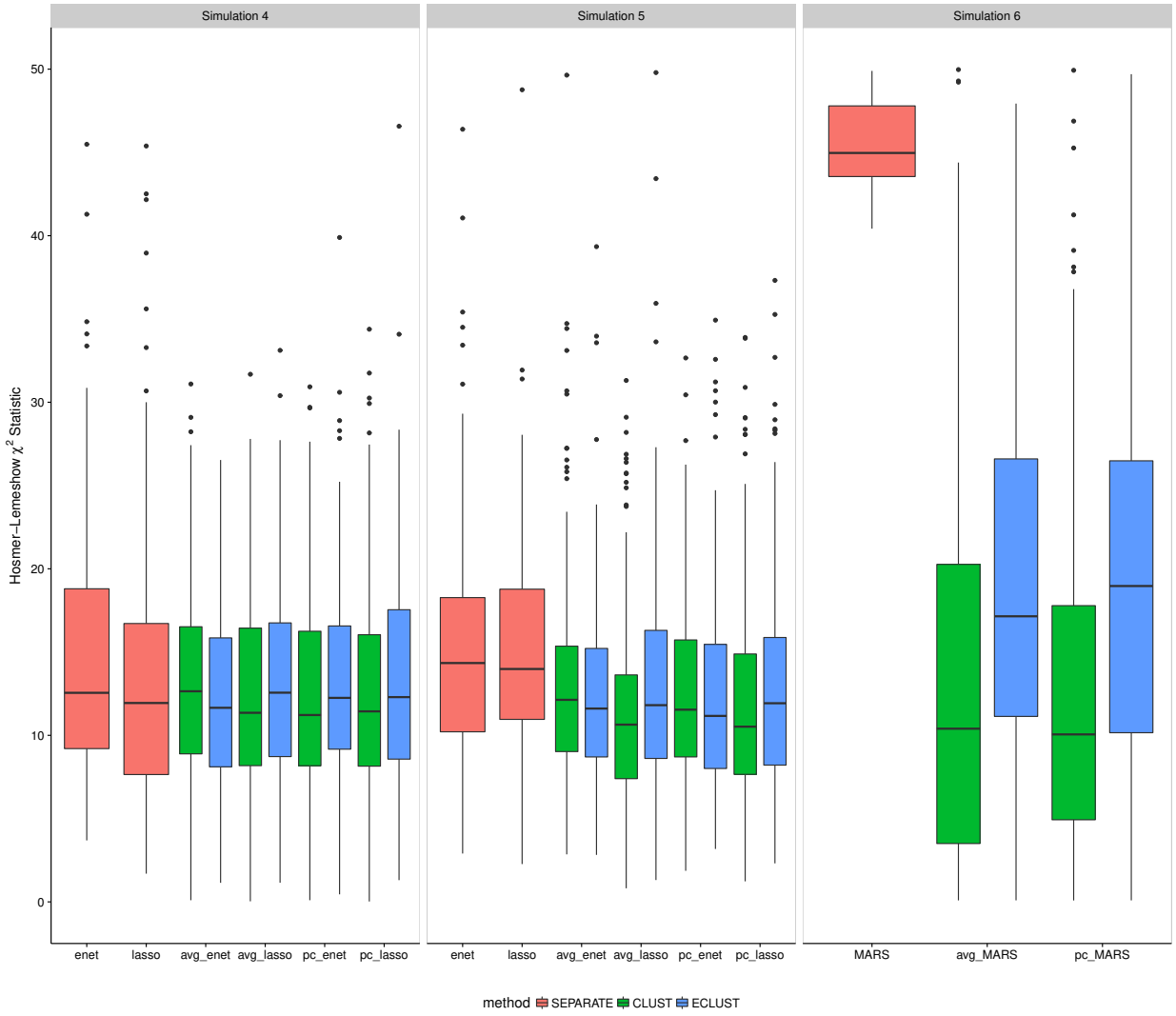


Figure A.3: Hosmer-Lemeshow statistics from simulations 4, 5 and 6 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

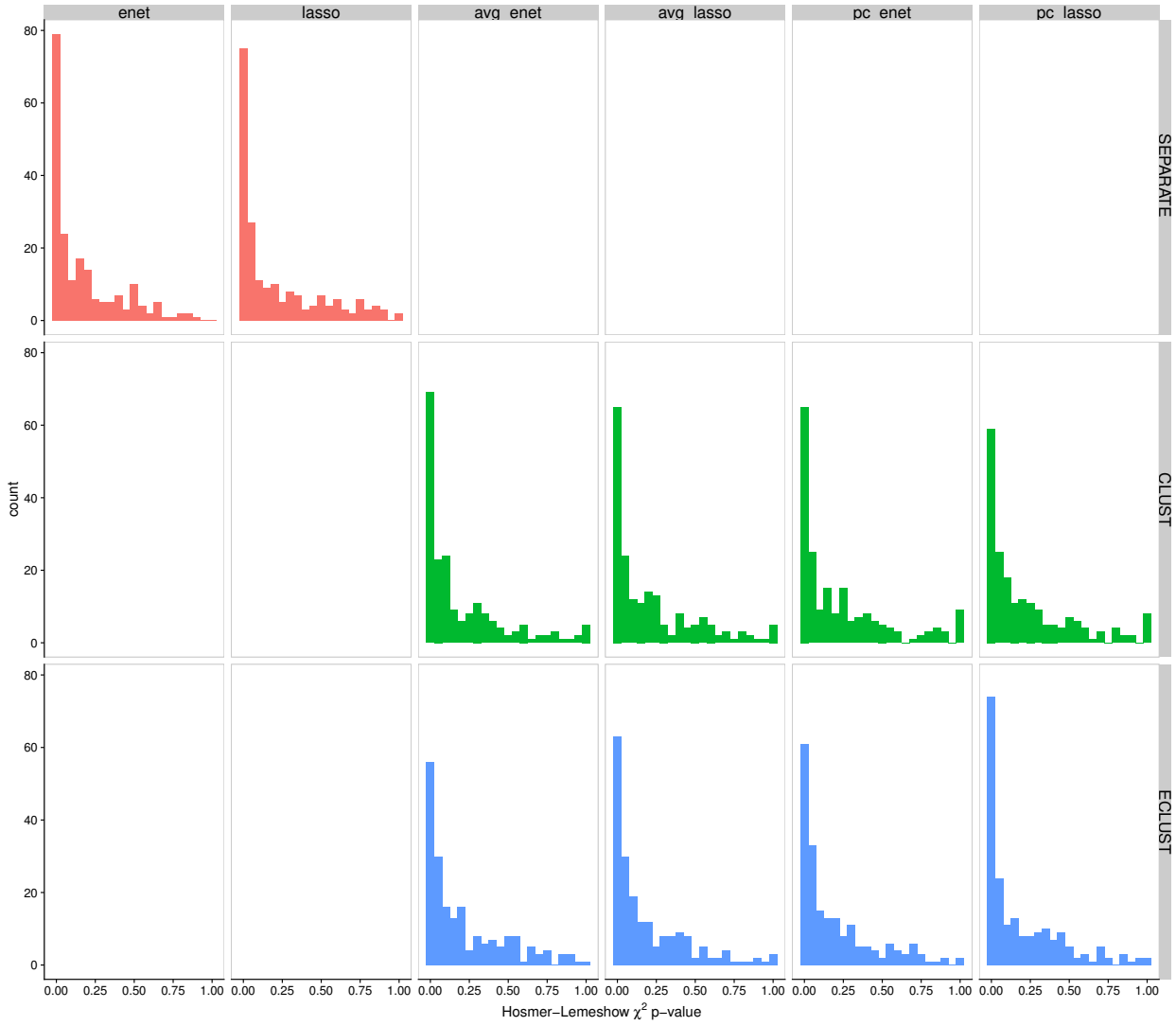


Figure A.4: Hosmer-Lemeshow p-values from simulation 4 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

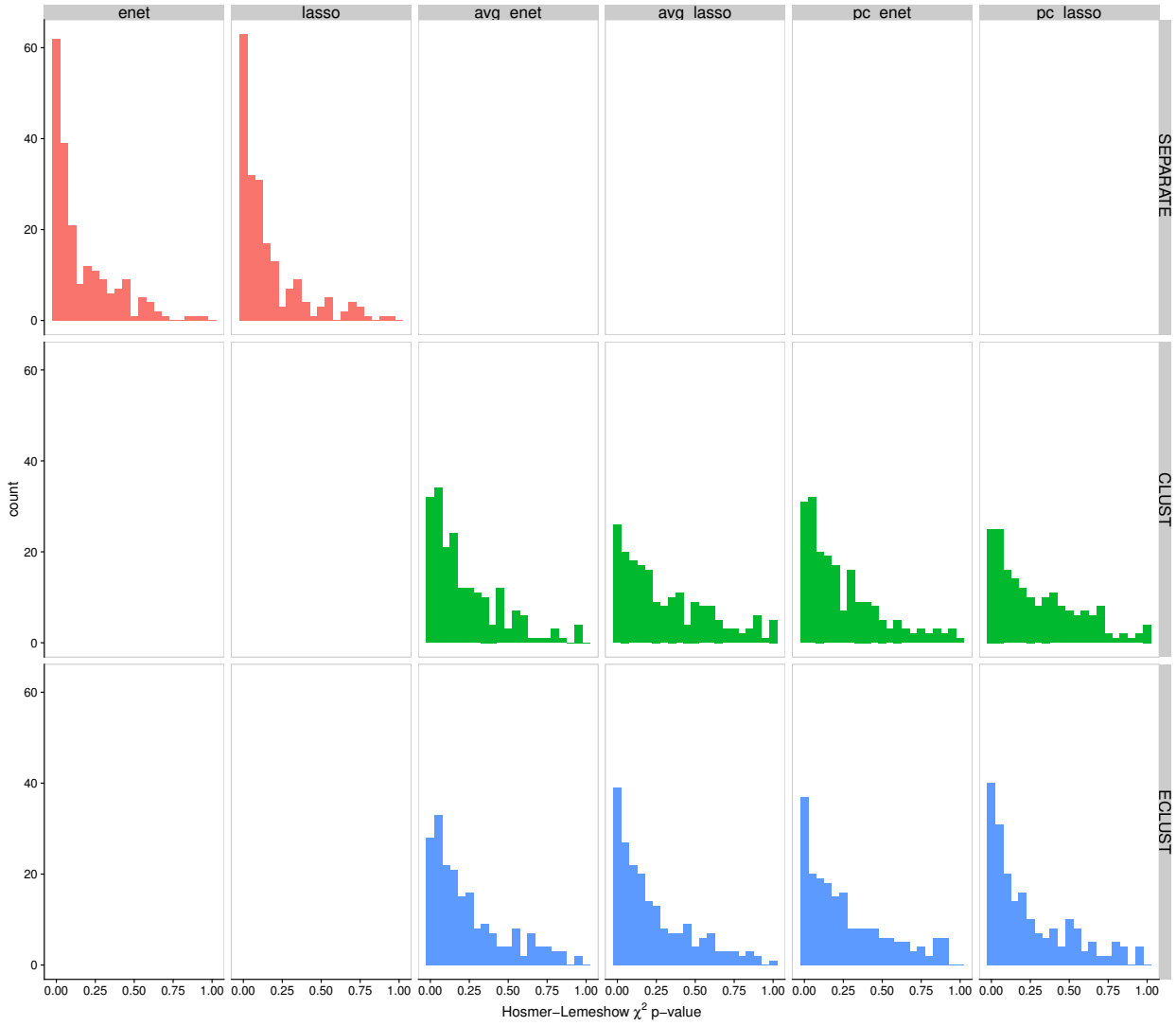


Figure A.5: Hosmer-Lemeshow p-values from simulation 5 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

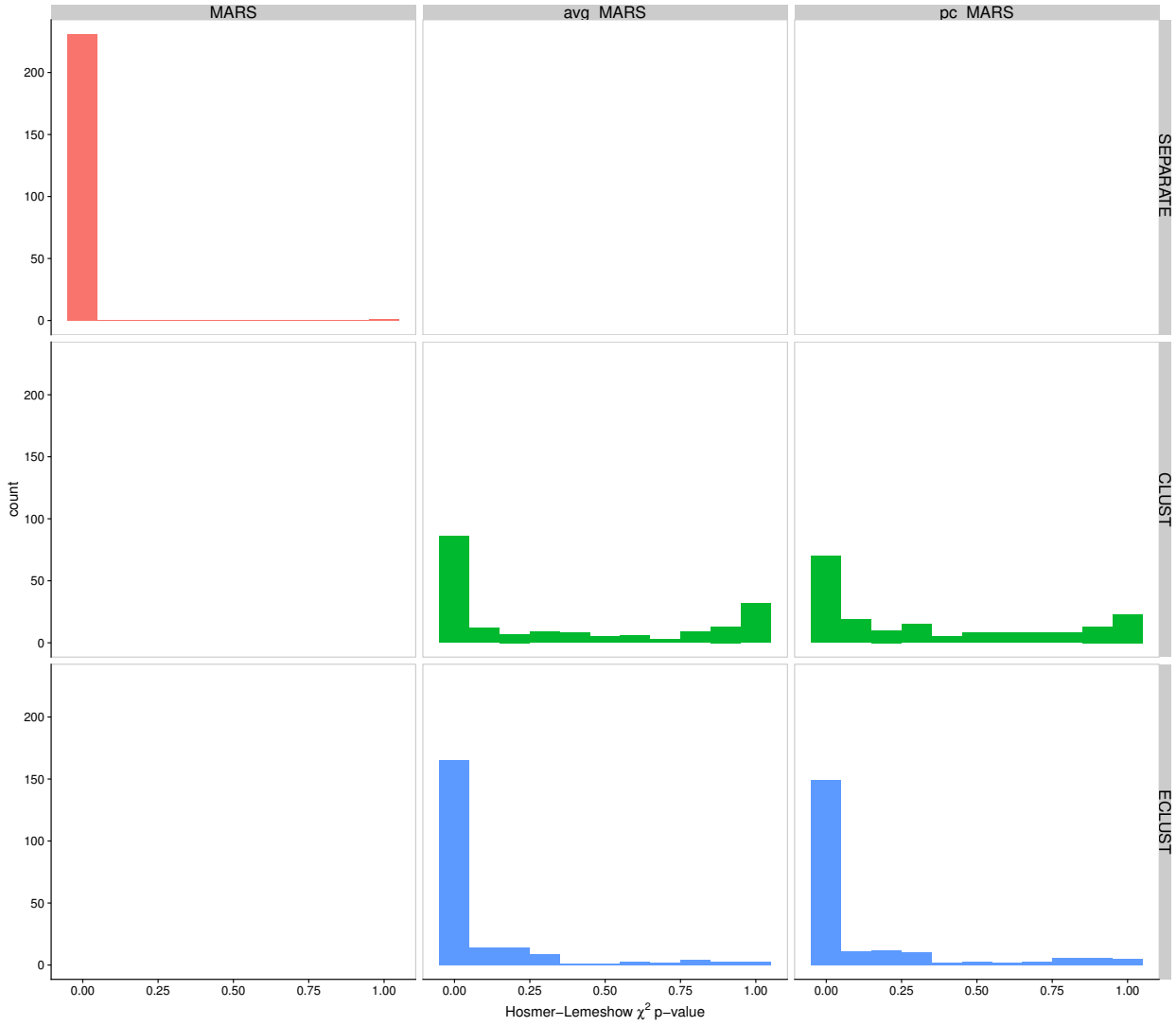


Figure A.6: Hosmer-Lemeshow p-values from simulation 6 for $SNR = 1$, $\rho = 0.9$, and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. SEPARATE results are in pink, CLUST in green and ECLUST in blue.

A.3 Analysis of Clusters

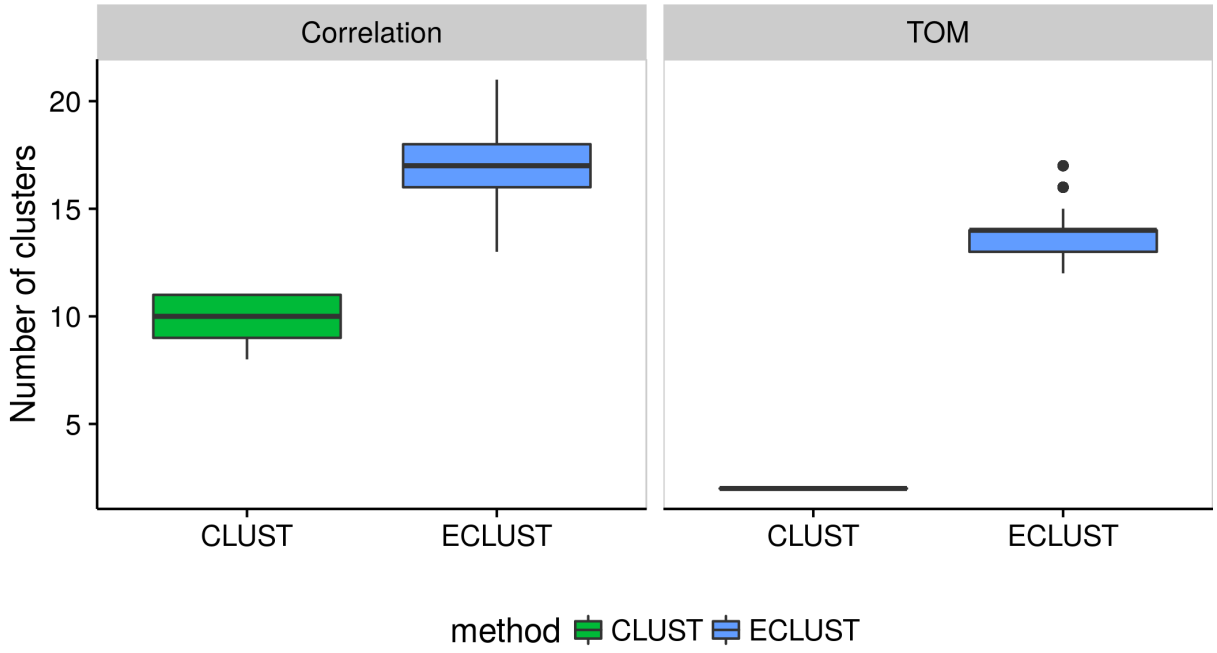


Figure A.7: Number of estimated clusters from applying the `dynamicTreeCut` algorithm to hierarchical clustering of the dissimilarity matrix with average linkage. Left panel: CLUST uses $1 - Cor(X_{all})$ and ECLUST uses the euclidean distance of $Cor(X_{\text{diff}})$ as measures of dissimilarity. Right panel: CLUST uses $1 - TOM(X_{all})$ and ECLUST uses the euclidean distance of $TOM(X_{\text{diff}})$ as measures of dissimilarity. Empirical distributions based on 200 simulation runs.

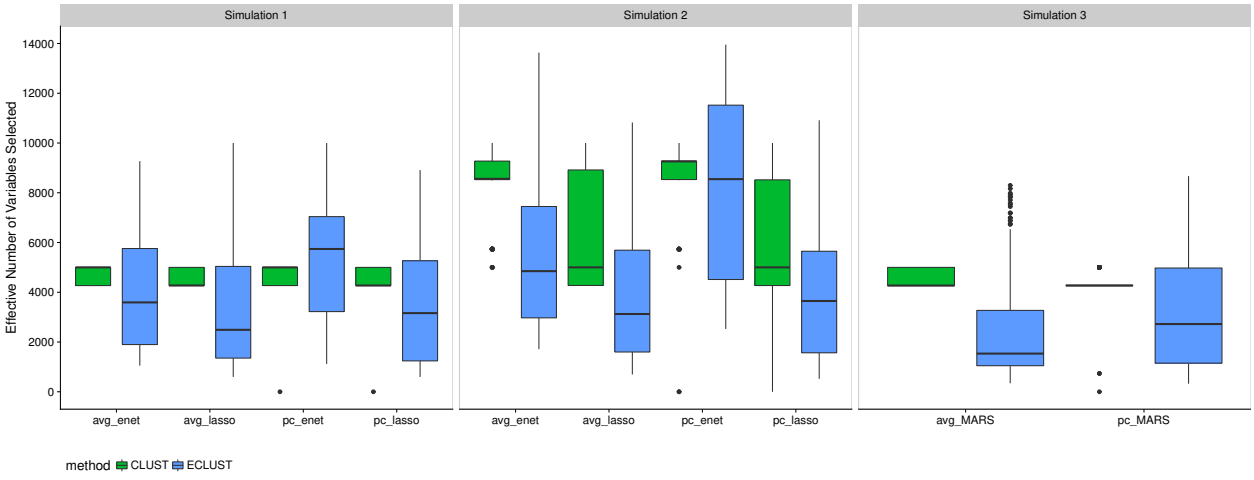


Figure A.8: Effective number of selected variables for simulations 1-3 for $SNR = 1, \rho = 0.9$. A variable was considered "selected" if its corresponding cluster representative was selected.

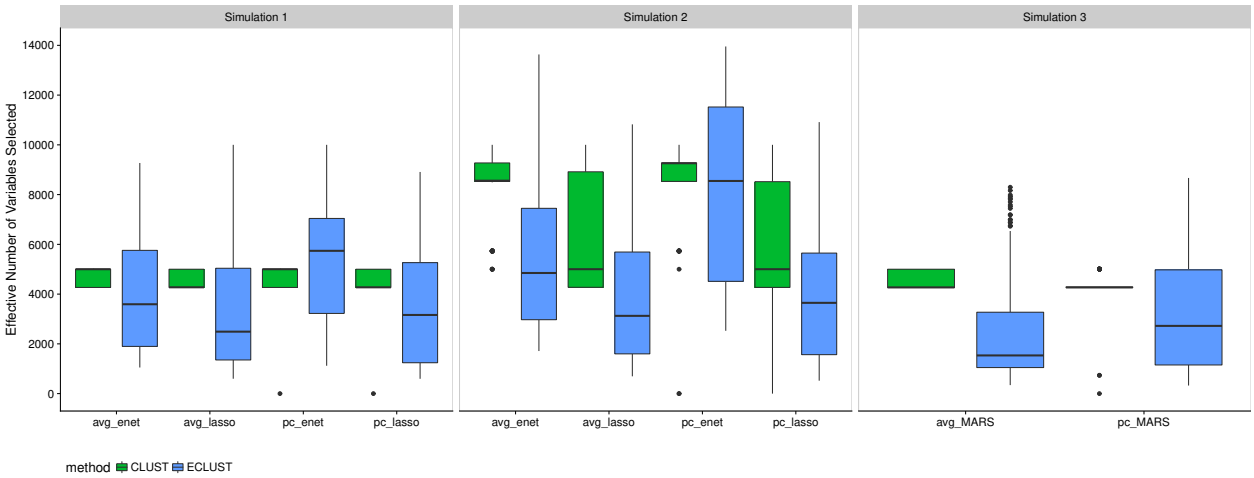


Figure A.9: Effective number of selected variables for simulations 4-6 for $SNR = 1, \rho = 0.9$ and $\alpha_j \sim \text{Unif}\left[\log(1.9), \log(2.1)\right]$. A variable was considered "selected" if its corresponding cluster representative was selected.

A.4 Simulation Results Using TOM as a Measure of Similarity

Simulation 1

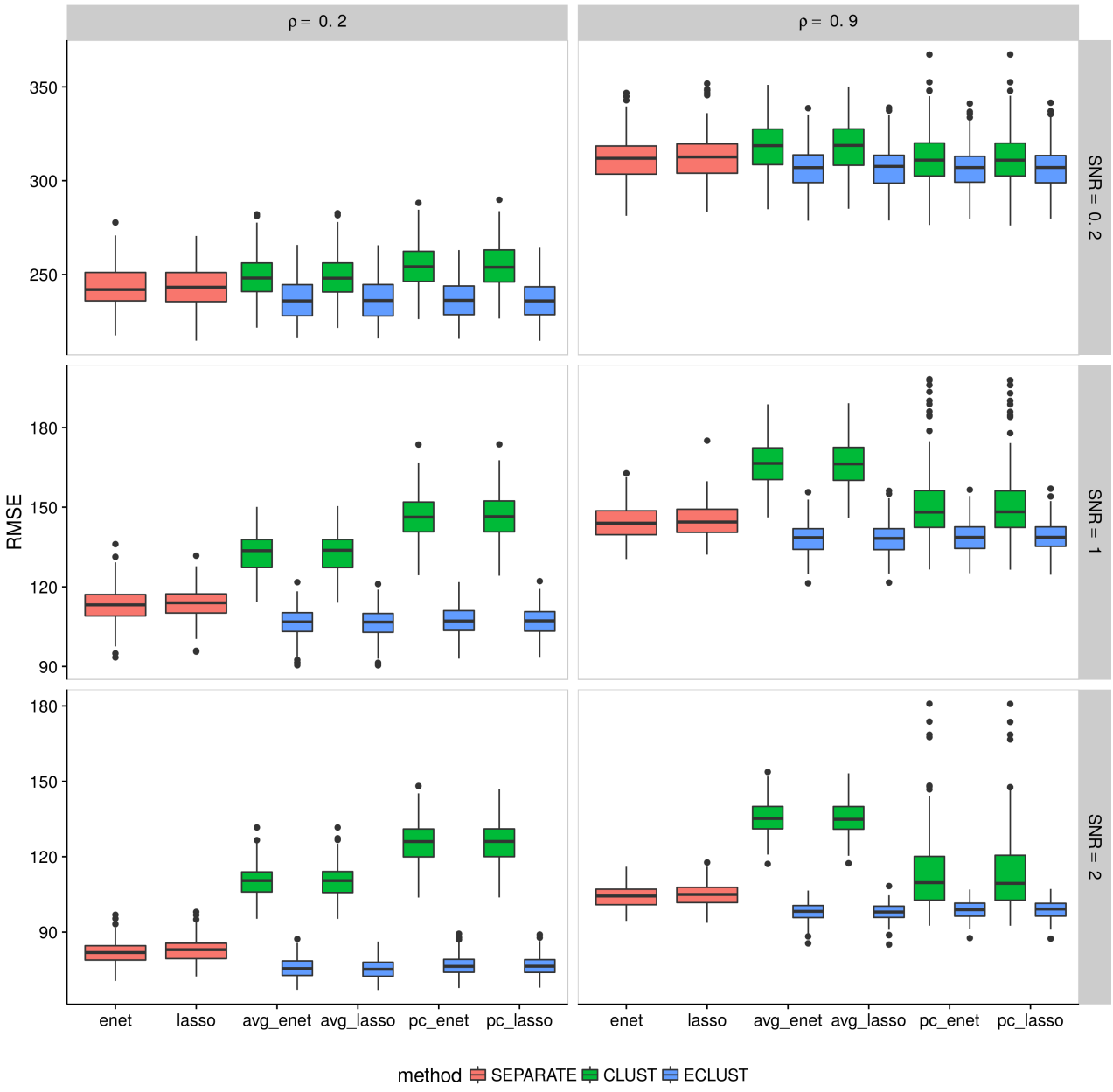


Figure A.10: Simulation 1 – Root mean squared error on an independent test set using the TOM as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

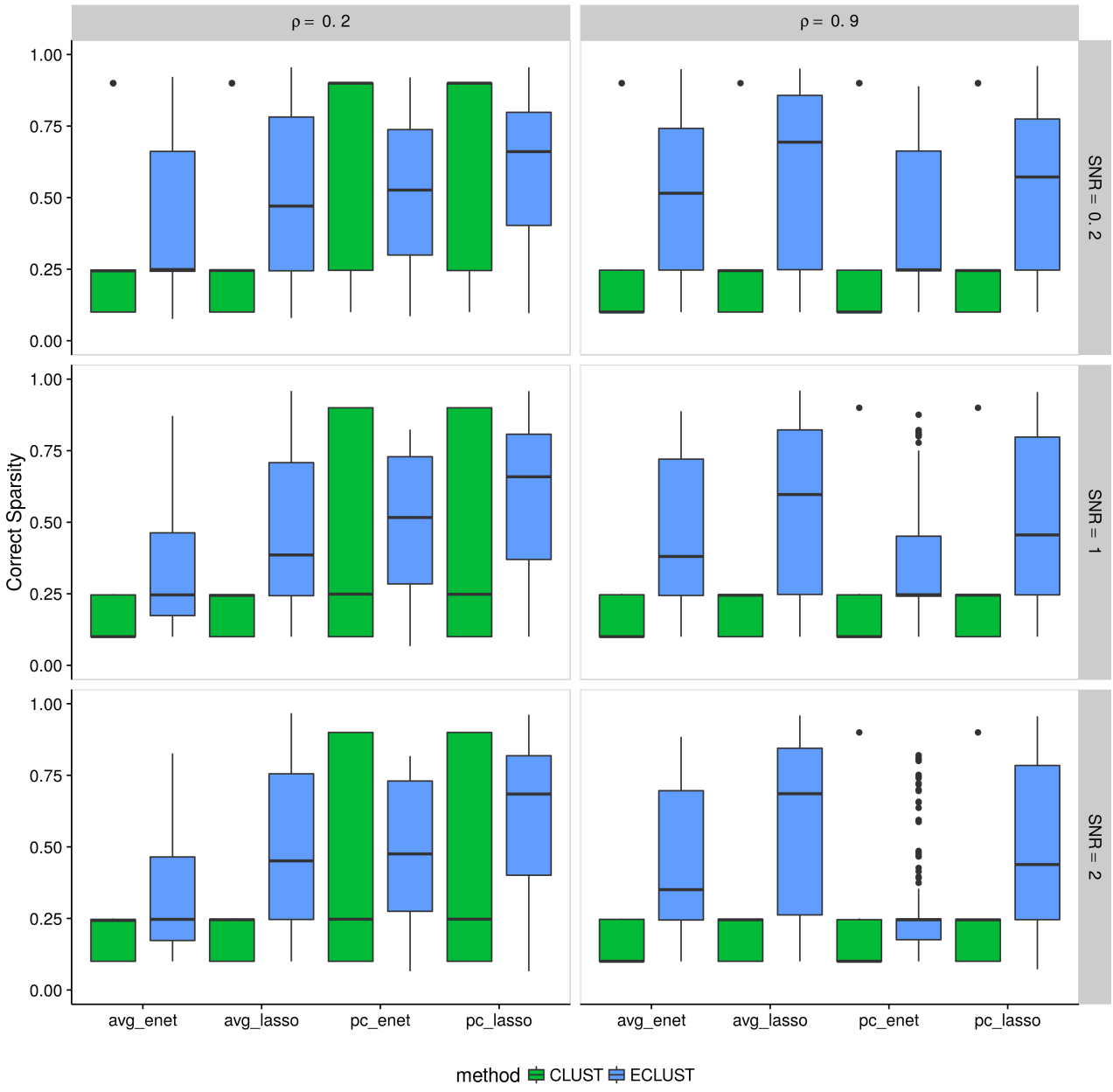


Figure A.11: Simulation 1 – Correct Sparsity based on the training set using the TOM as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

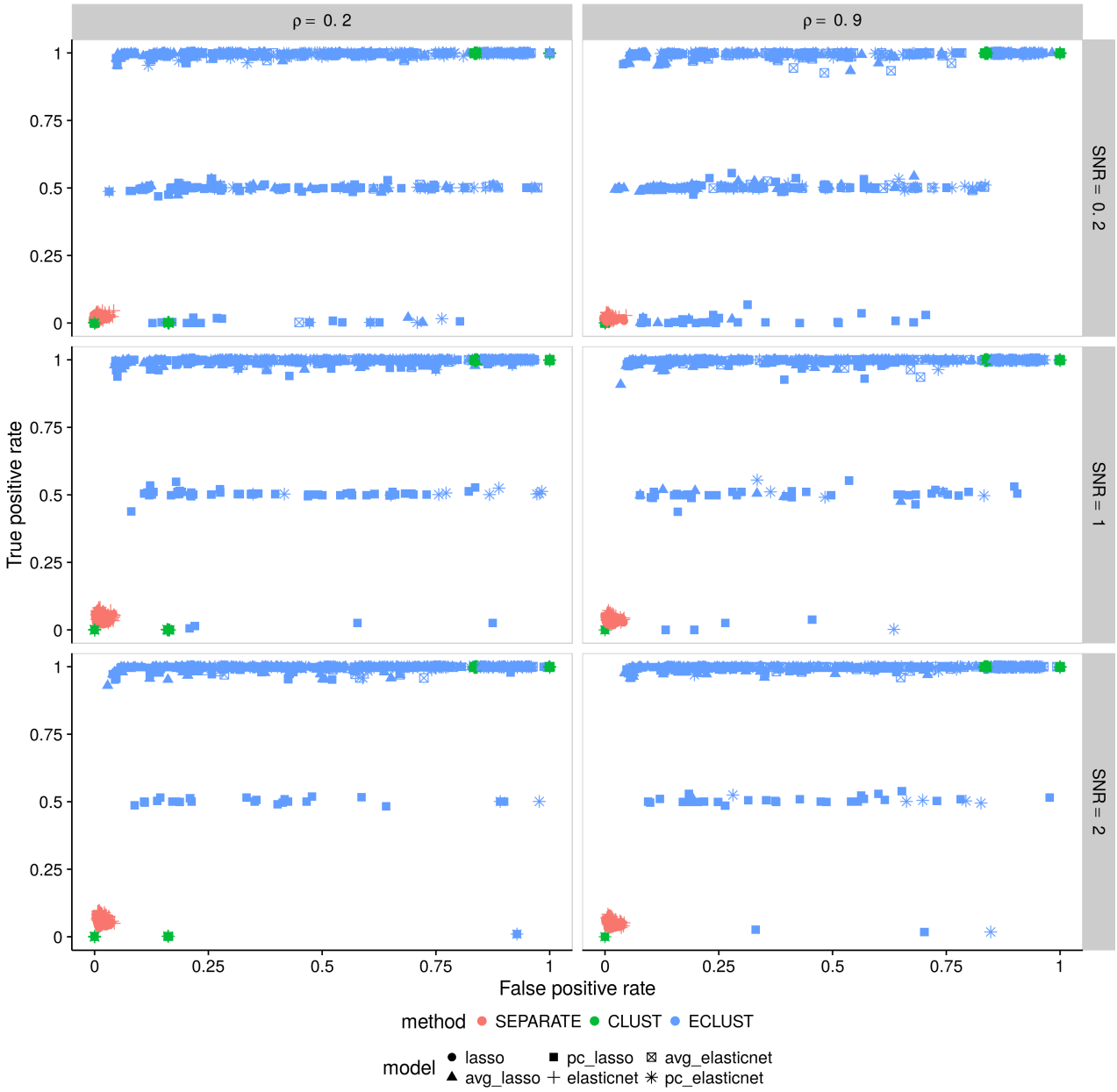


Figure A.12: Simulation 1 – True positive rate vs. false positive rate based on the training set using the TOM as a measure of similarity. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

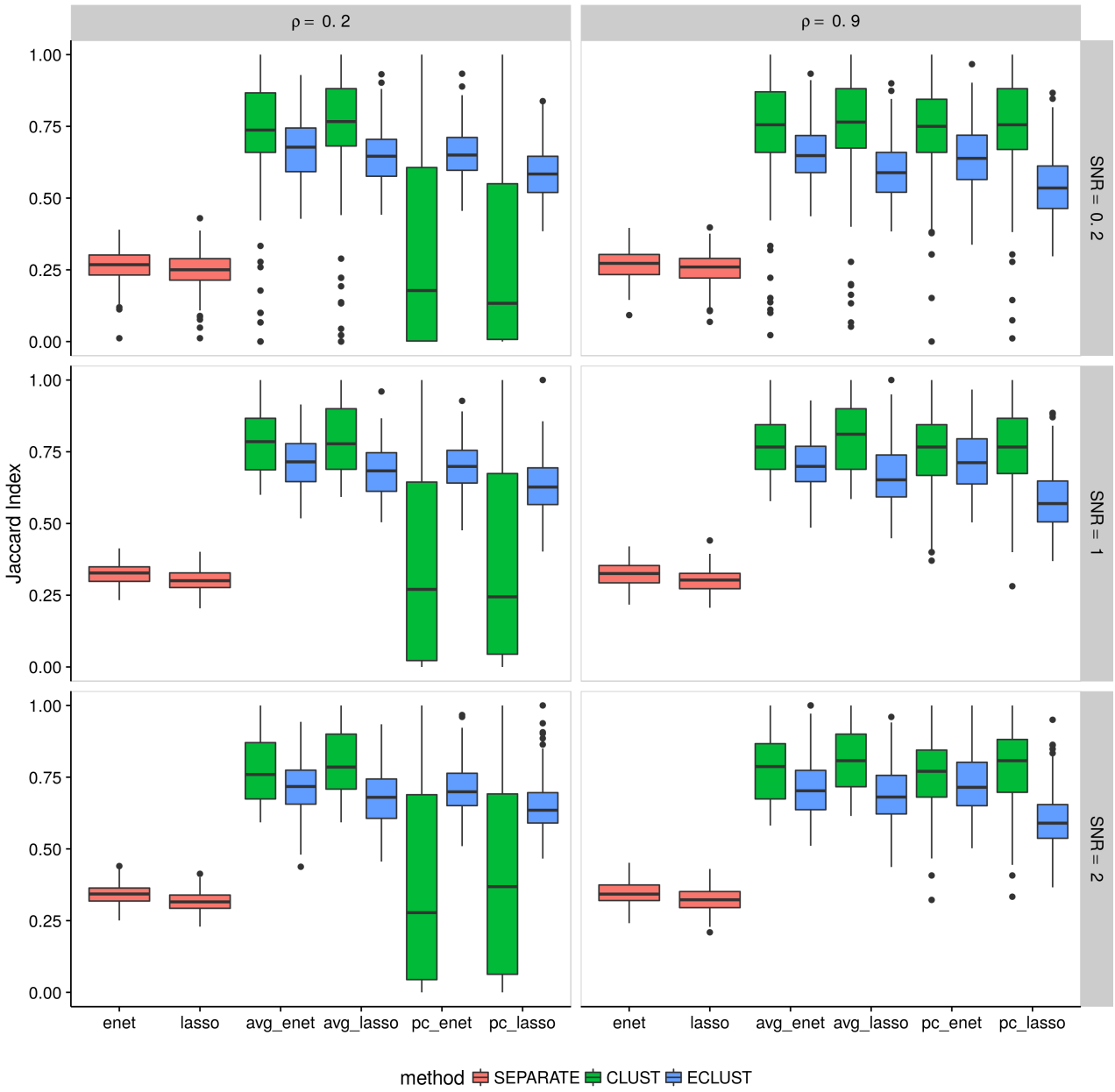


Figure A.13: Simulation 1 – Average Jaccard Index from 10 CV folds of the training set using the TOM as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

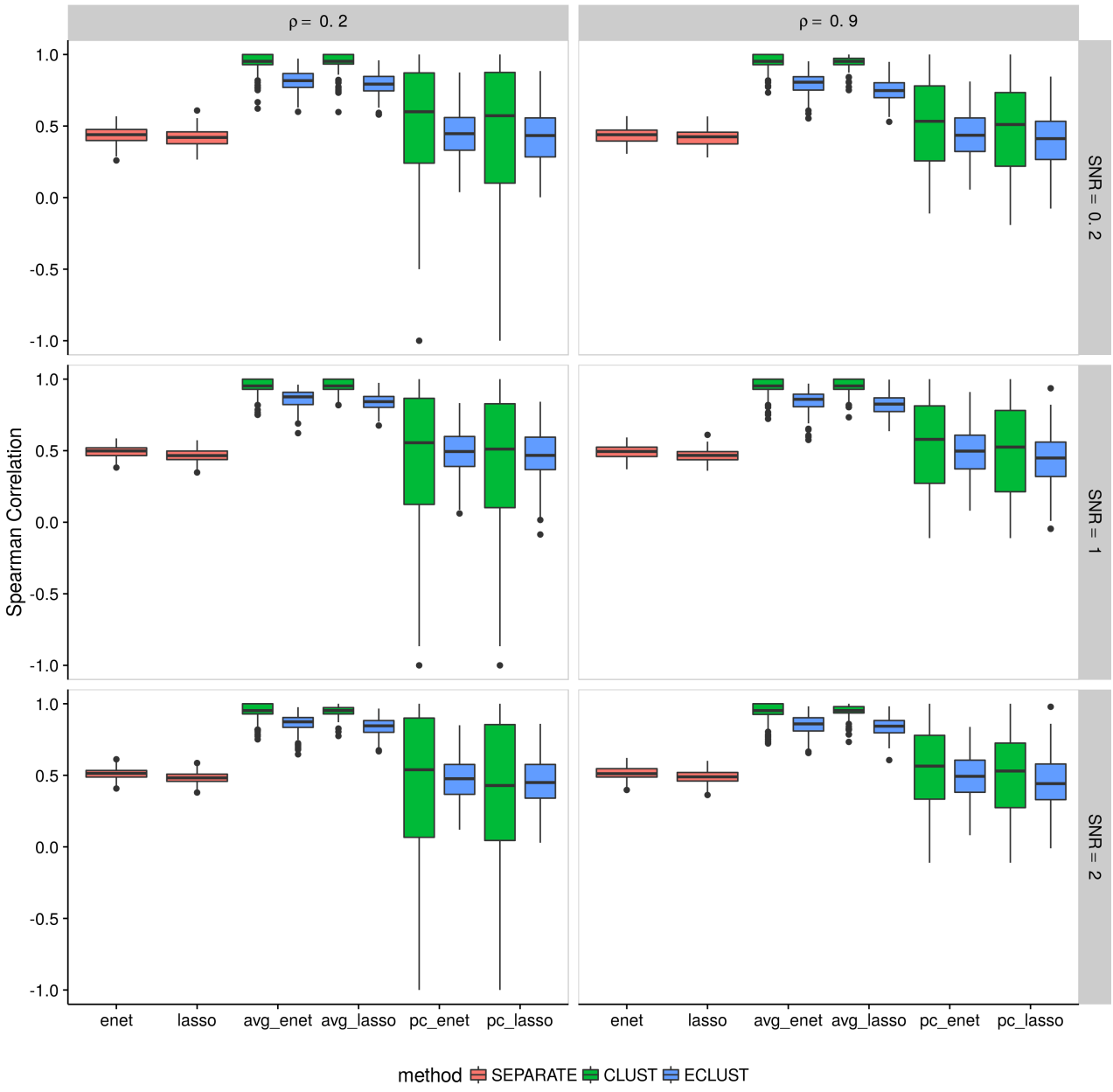


Figure A.14: Simulation 1 – Average Spearman correlation from 10 CV folds of the training set using the TOM as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Spearman correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

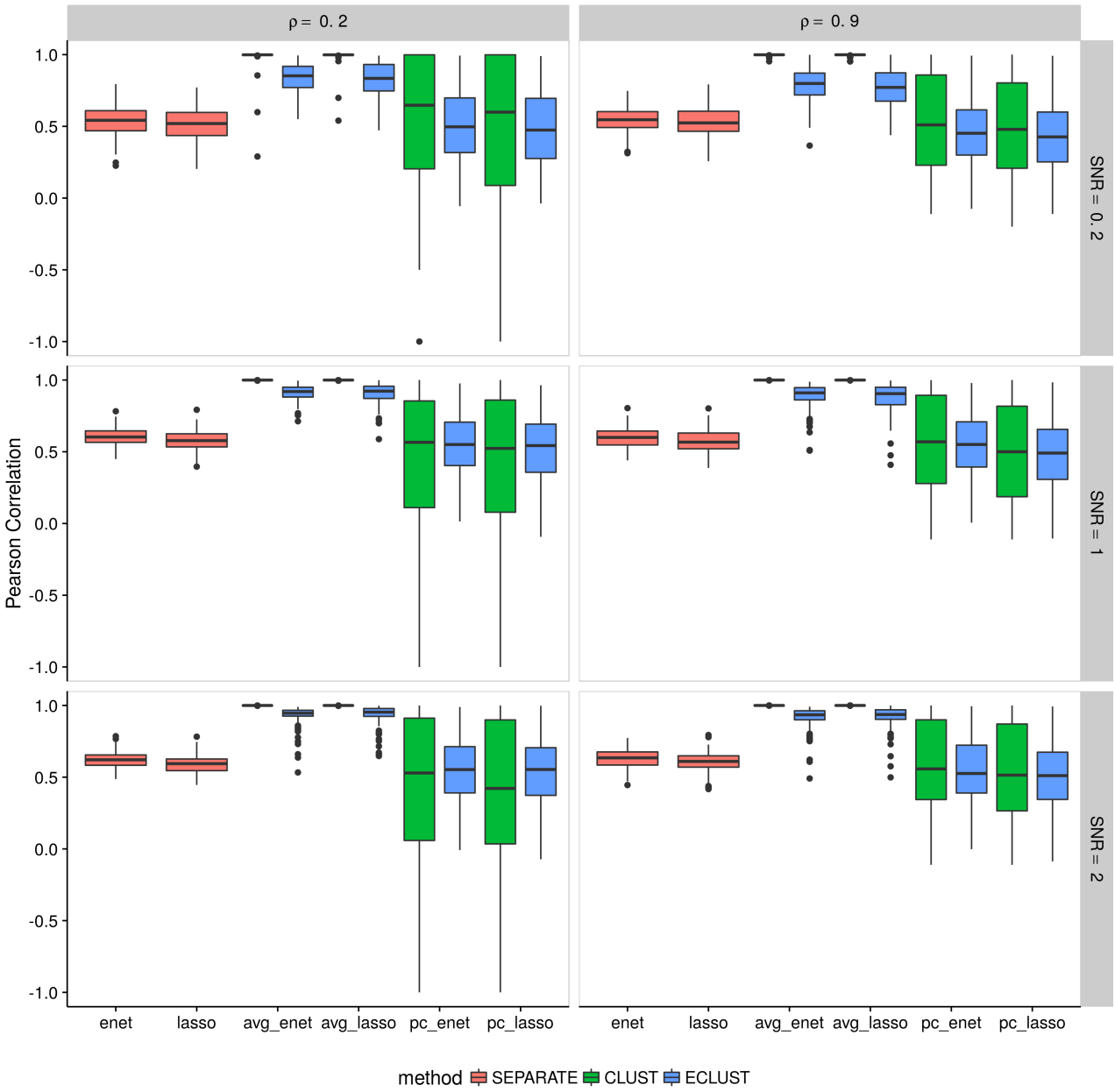


Figure A.15: Simulation 1 – Average Pearson correlation from 10 CV folds of the training set using the TOM as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Pearson correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

Simulation 2

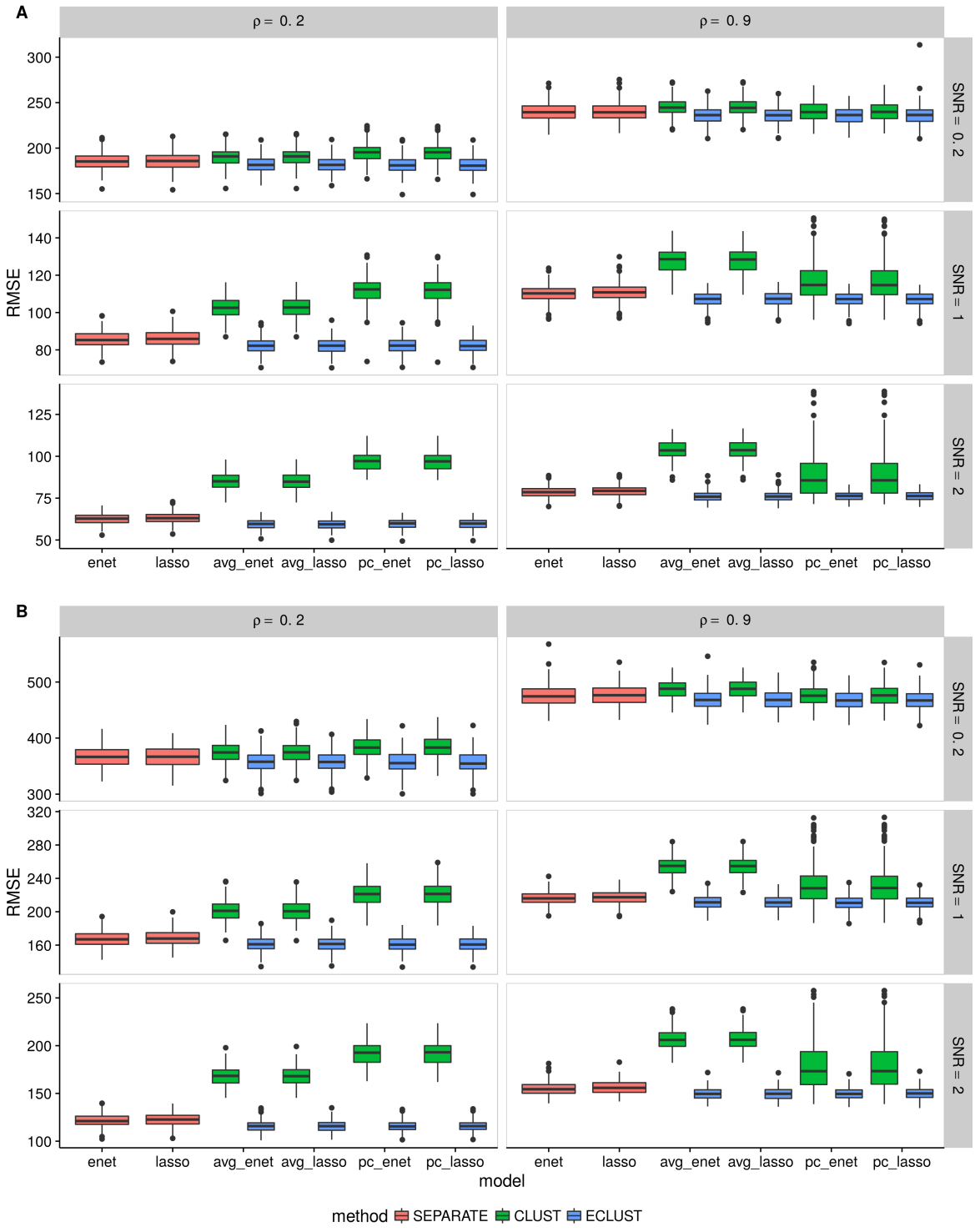


Figure A.16: Simulation 2 – Root mean squared error on an independent test set using the TOM as a measure of similarity from 200 simulation runs. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

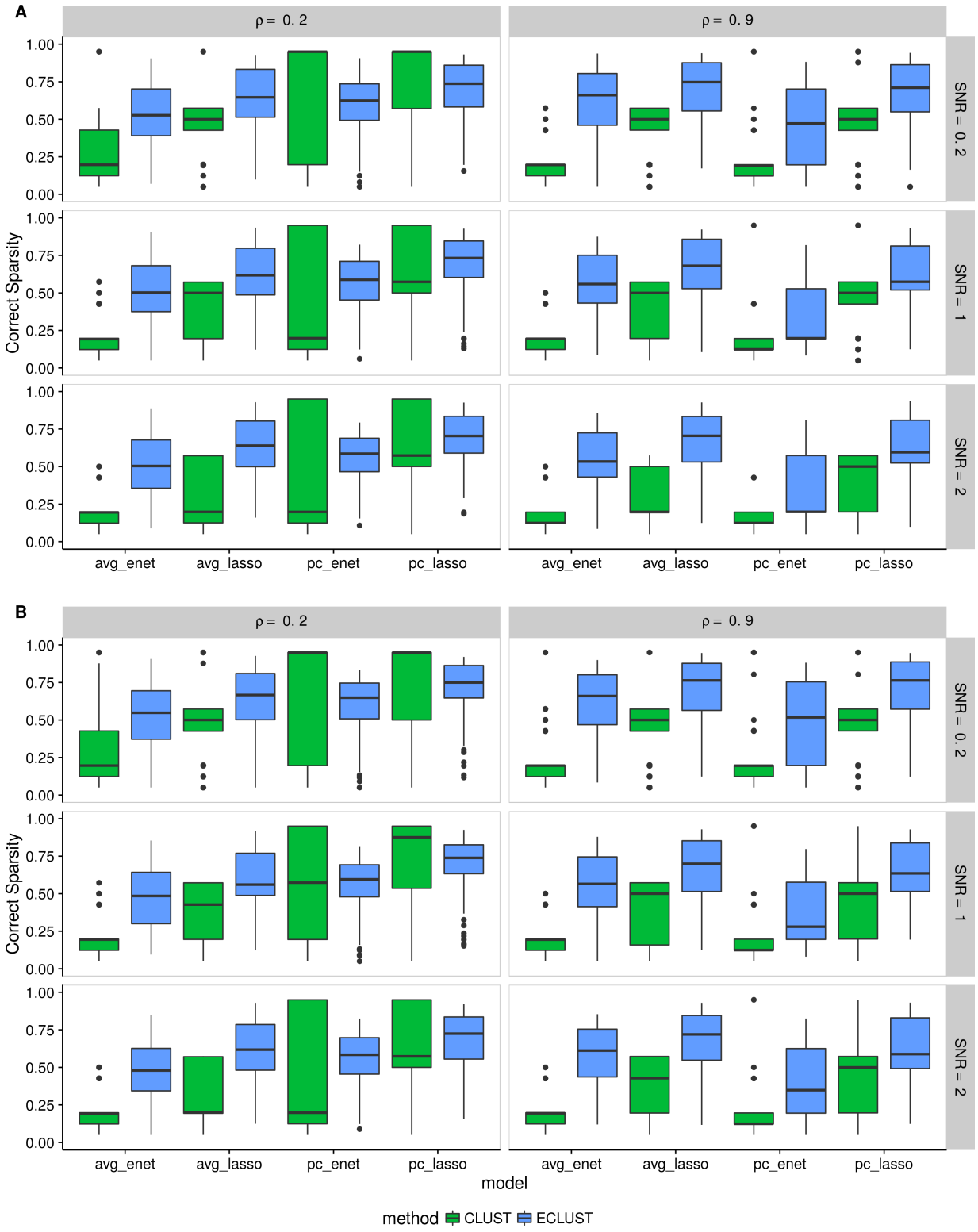


Figure A.17: Simulation 2 – Correct Sparsity based on the training set using the TOM as a measure of similarity from 200 simulation runs. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

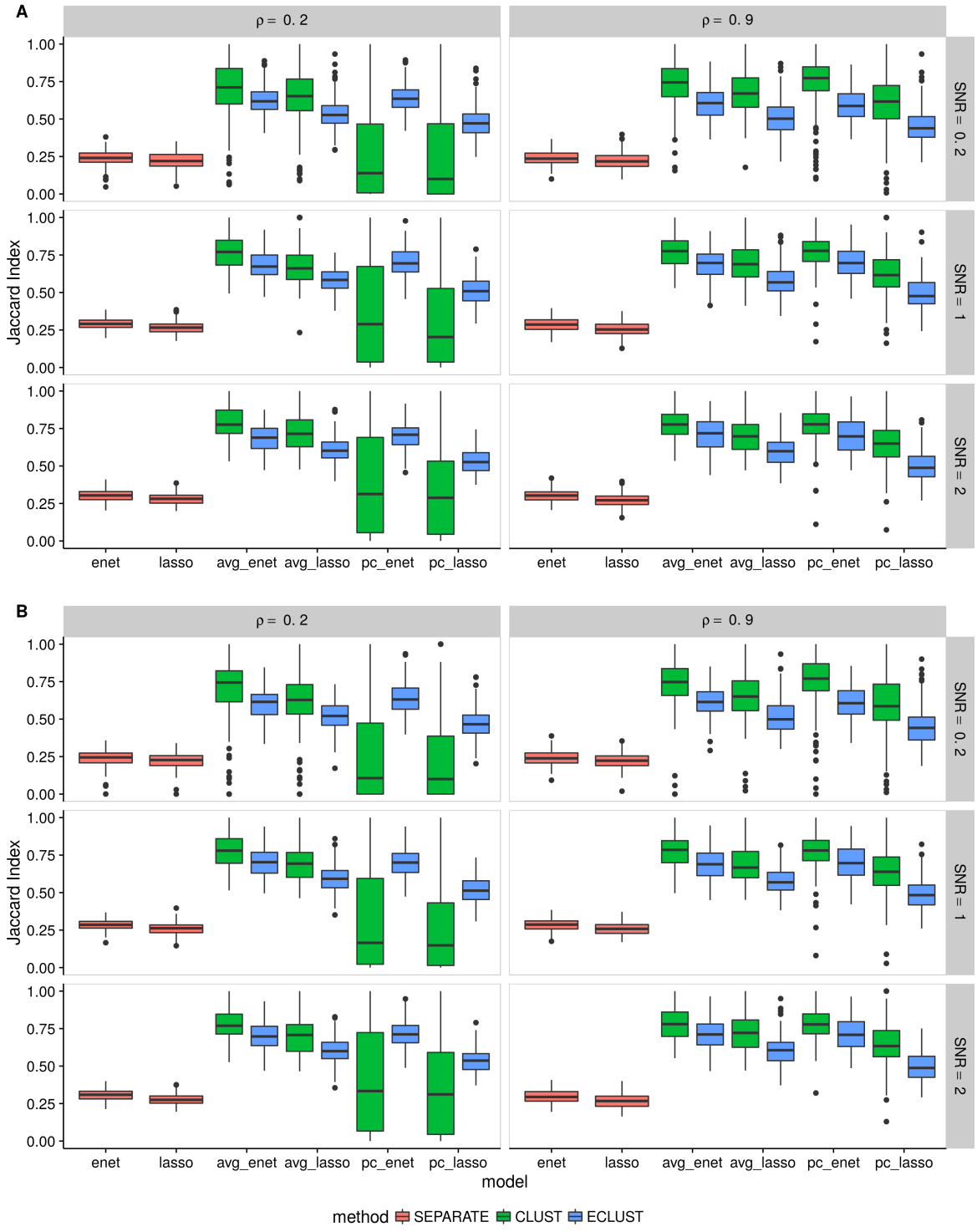


Figure A.19: Simulation 2 – Average Jaccard Index from 10 CV folds of the training set using the TOM as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

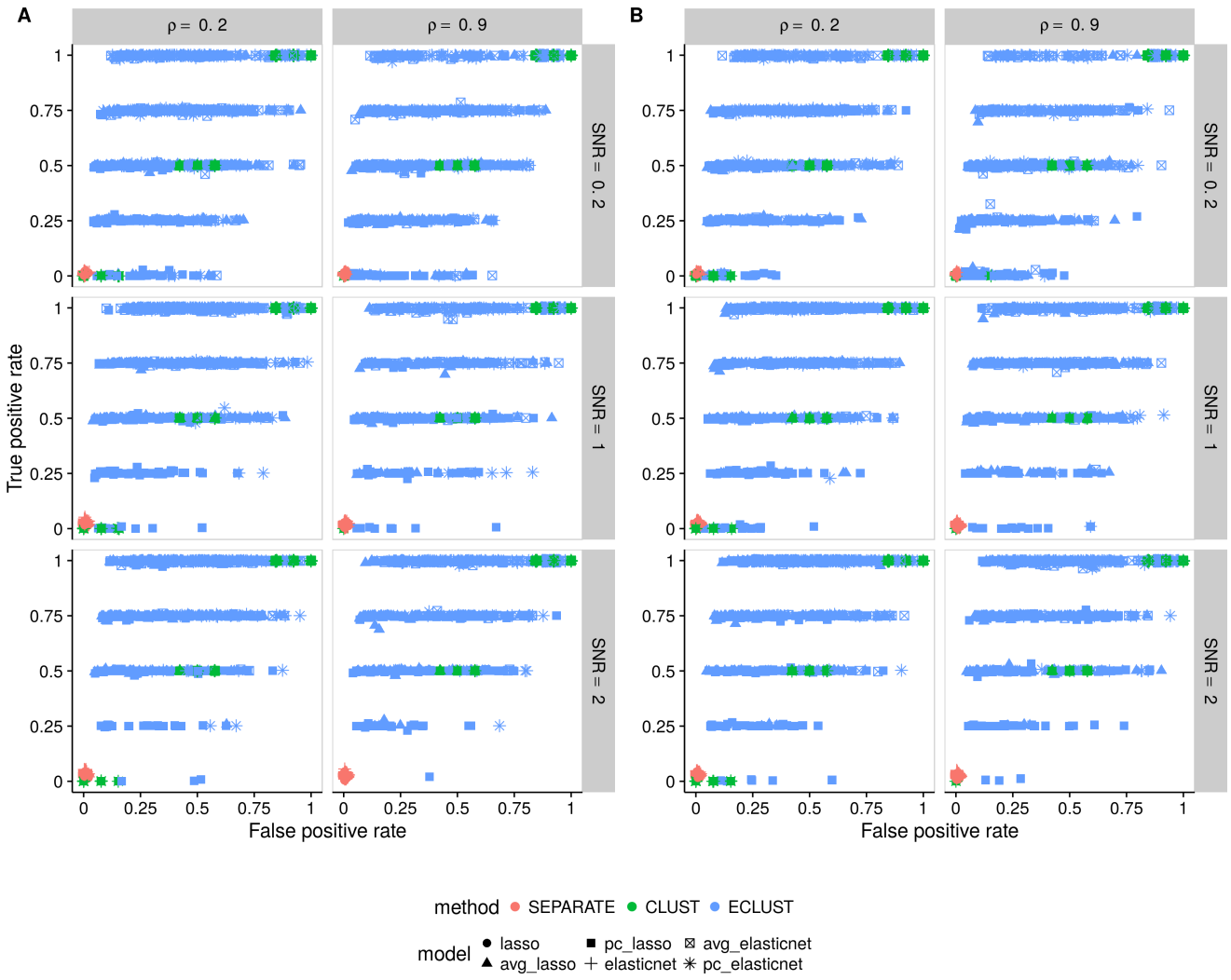


Figure A.18: Simulation 2 – True positive rate vs. false positive rate based on the training set using the TOM as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

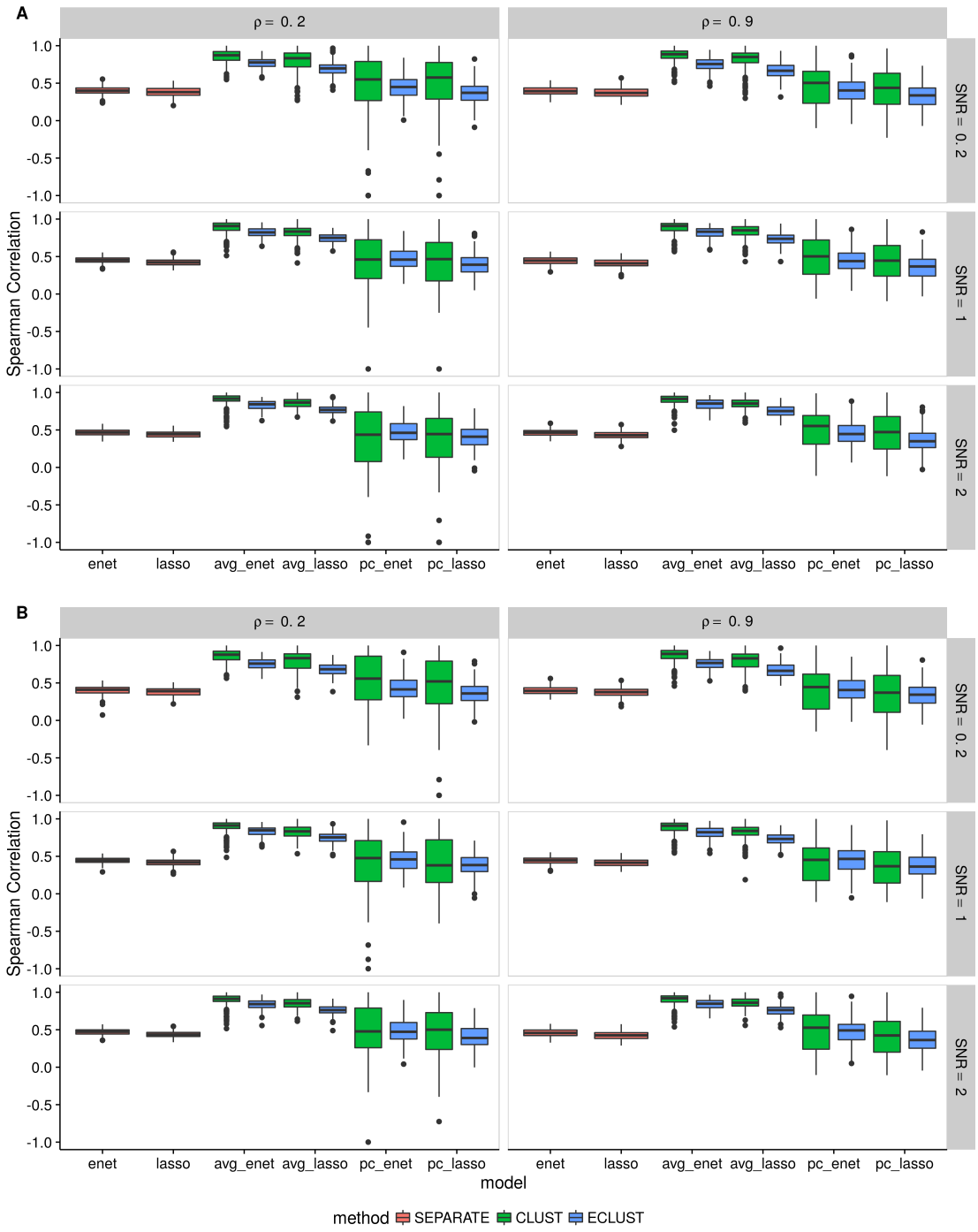


Figure A.20: Simulation 2 – Average Spearman correlation from 10 CV folds of the training set using the TOM as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Spearman correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

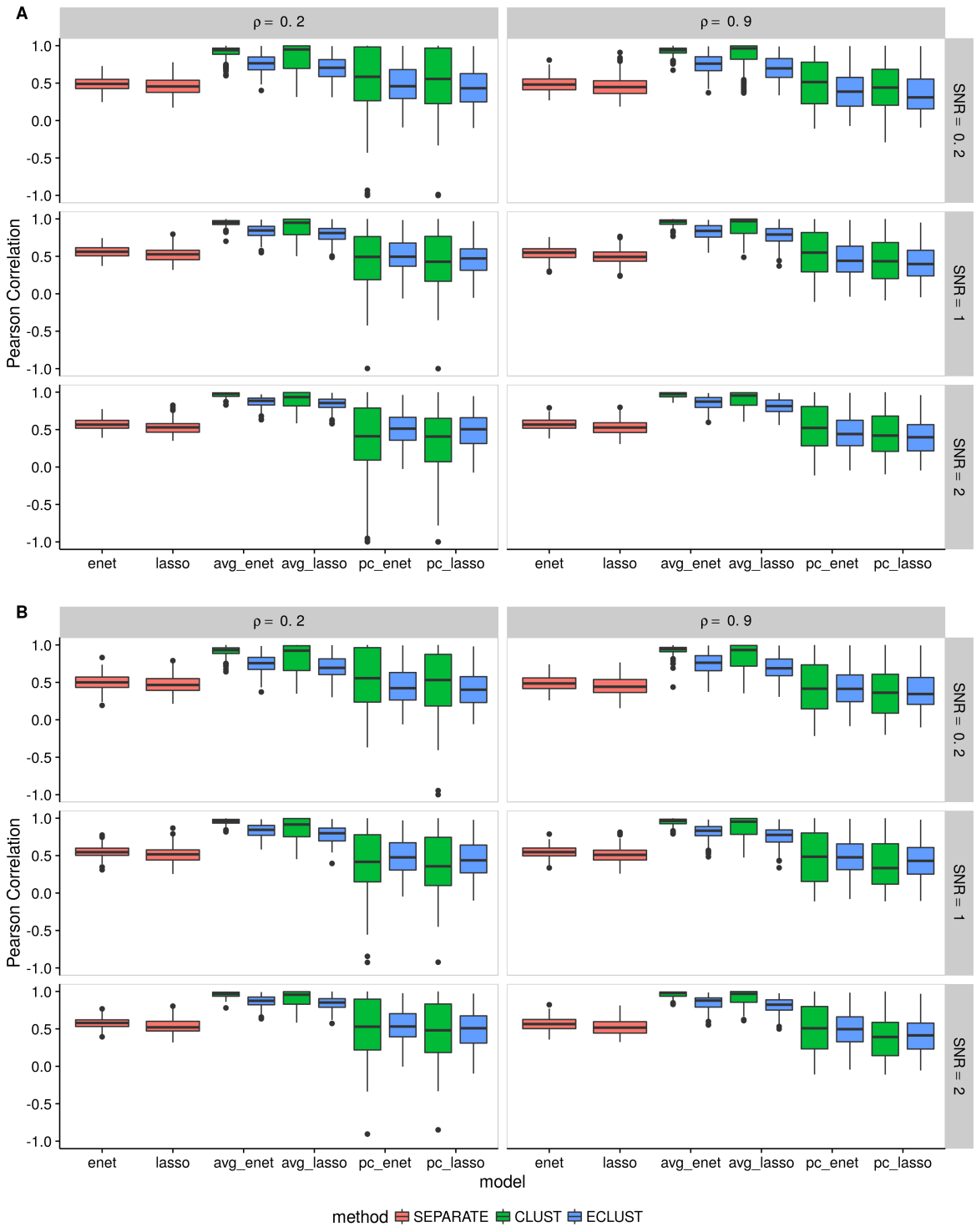


Figure A.21: Simulation 2 – Average Pearson correlation from 10 CV folds of the training set using the TOM as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Pearson correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

Simulation 3

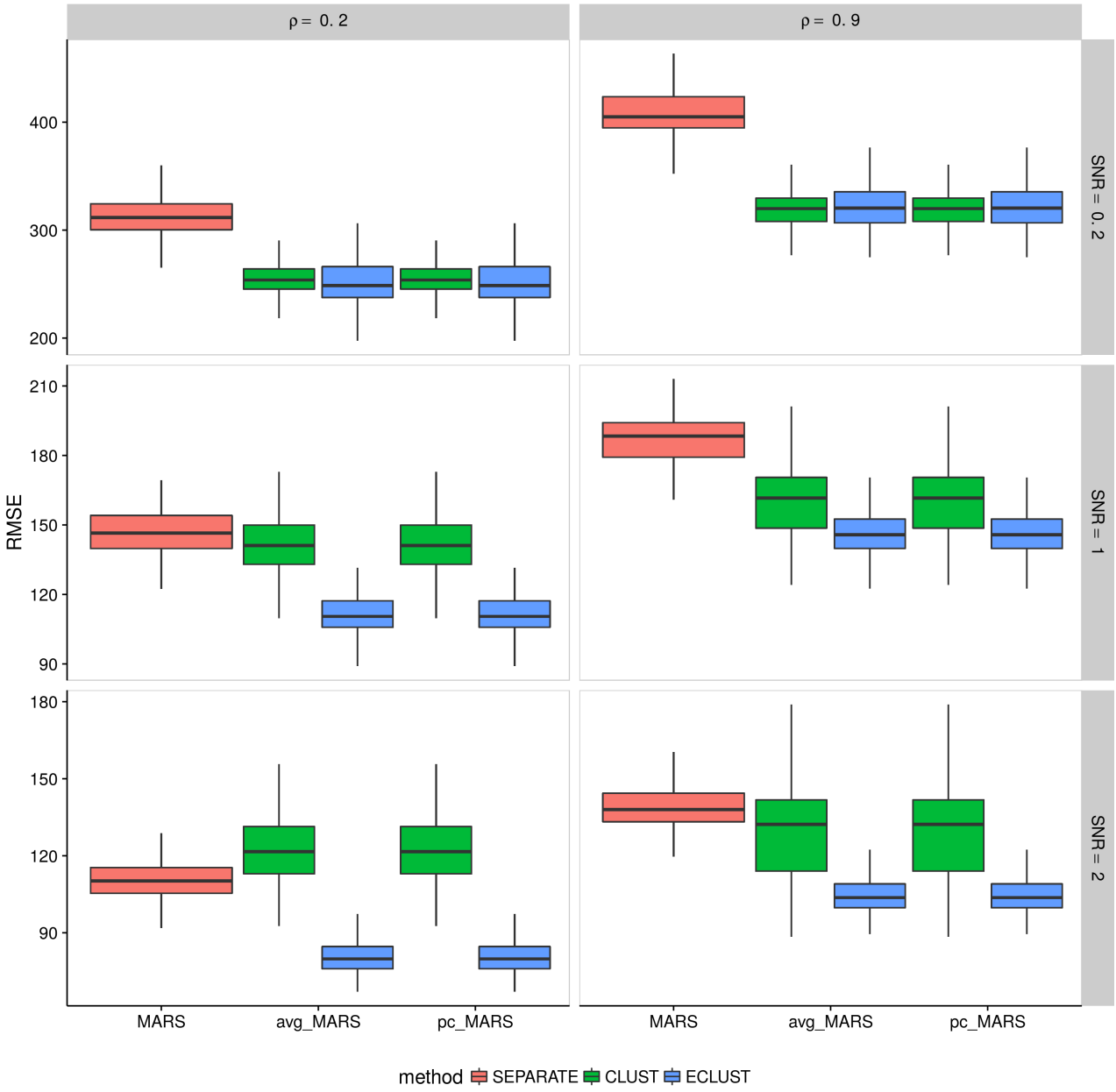


Figure A.22: Simulation 3 – Root mean squared error on an independent test set using the TOM as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

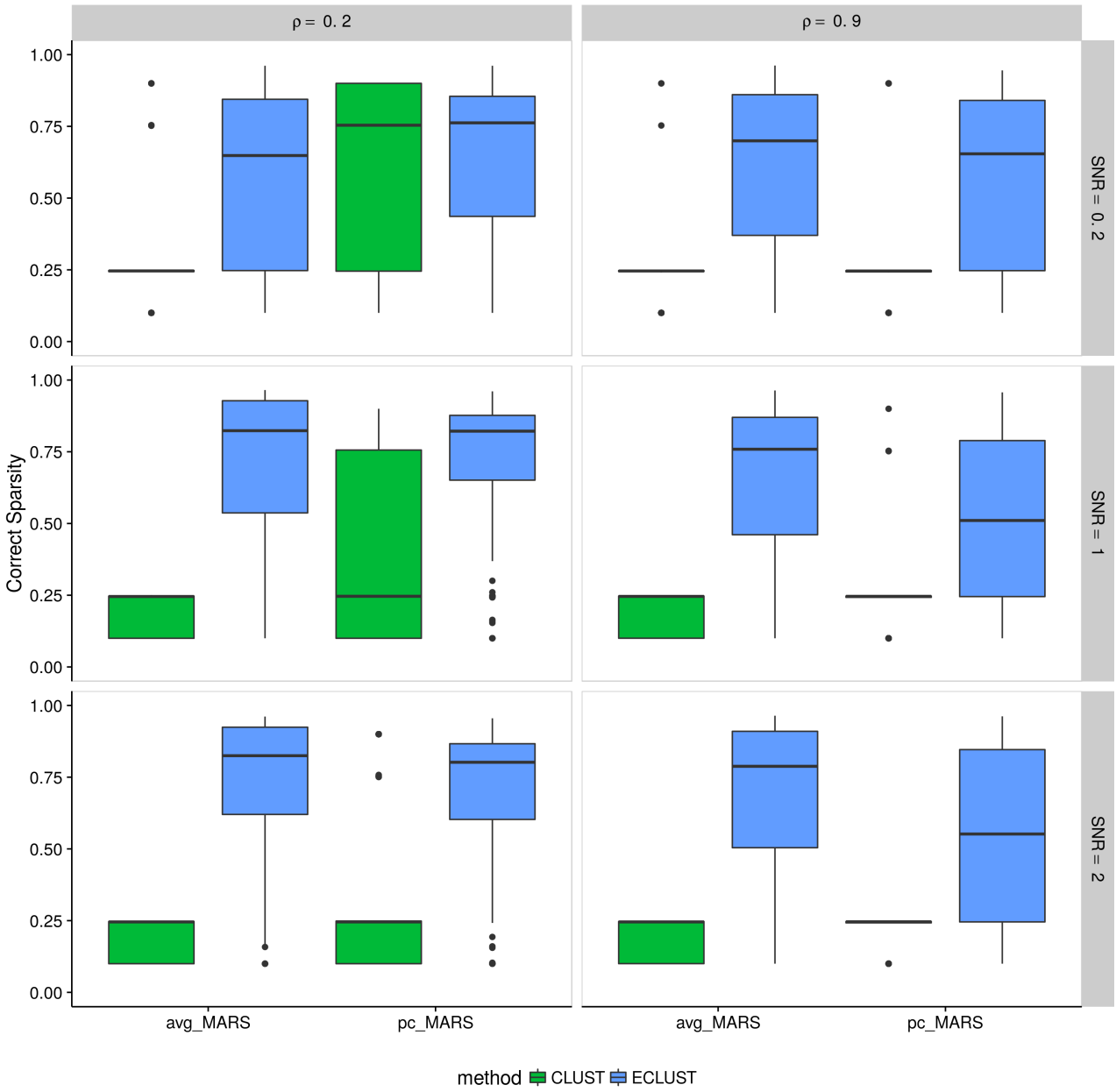


Figure A.23: Simulation 3 – Correct Sparsity based on the training set using the TOM as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

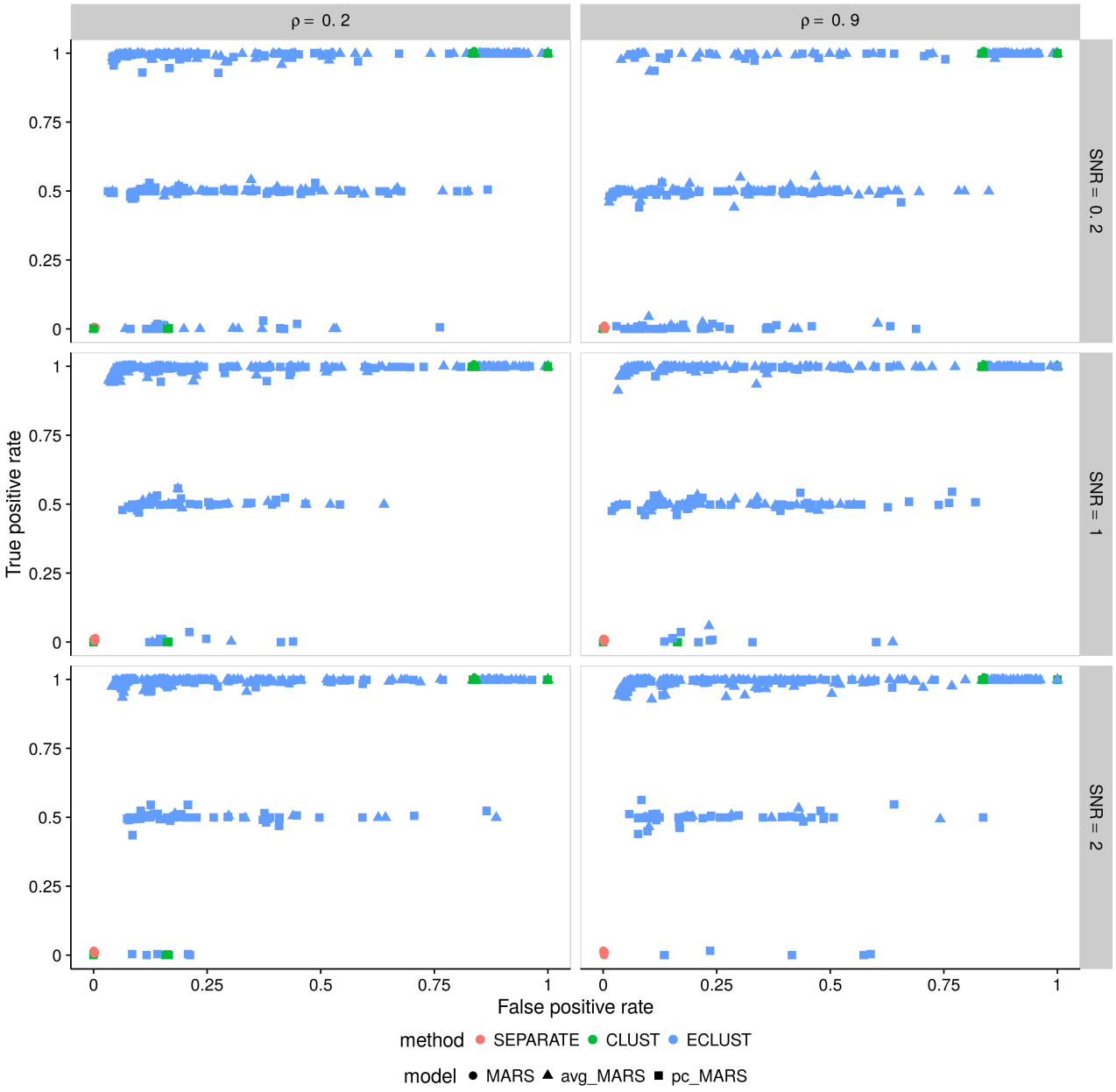


Figure A.24: Simulation 3 – True positive rate vs. false positive rate based on the training set using the TOM as a measure of similarity. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

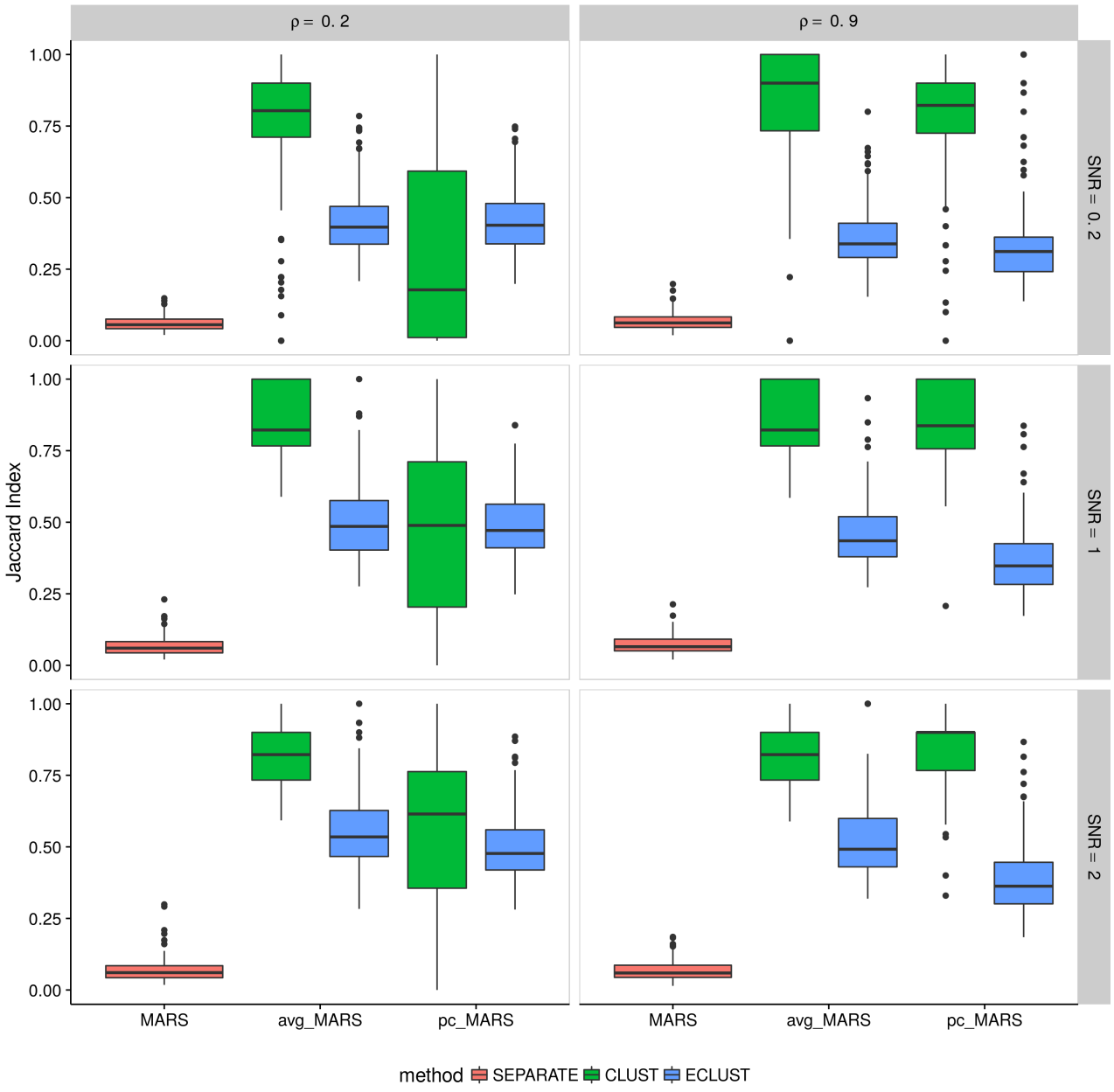


Figure A.25: Simulation 3 – Average Jaccard Index from 10 CV folds of the training set using the TOM as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

A.5 Simulation Results Using Pearson Correlations as a Measure of Similarity

Simulation 1

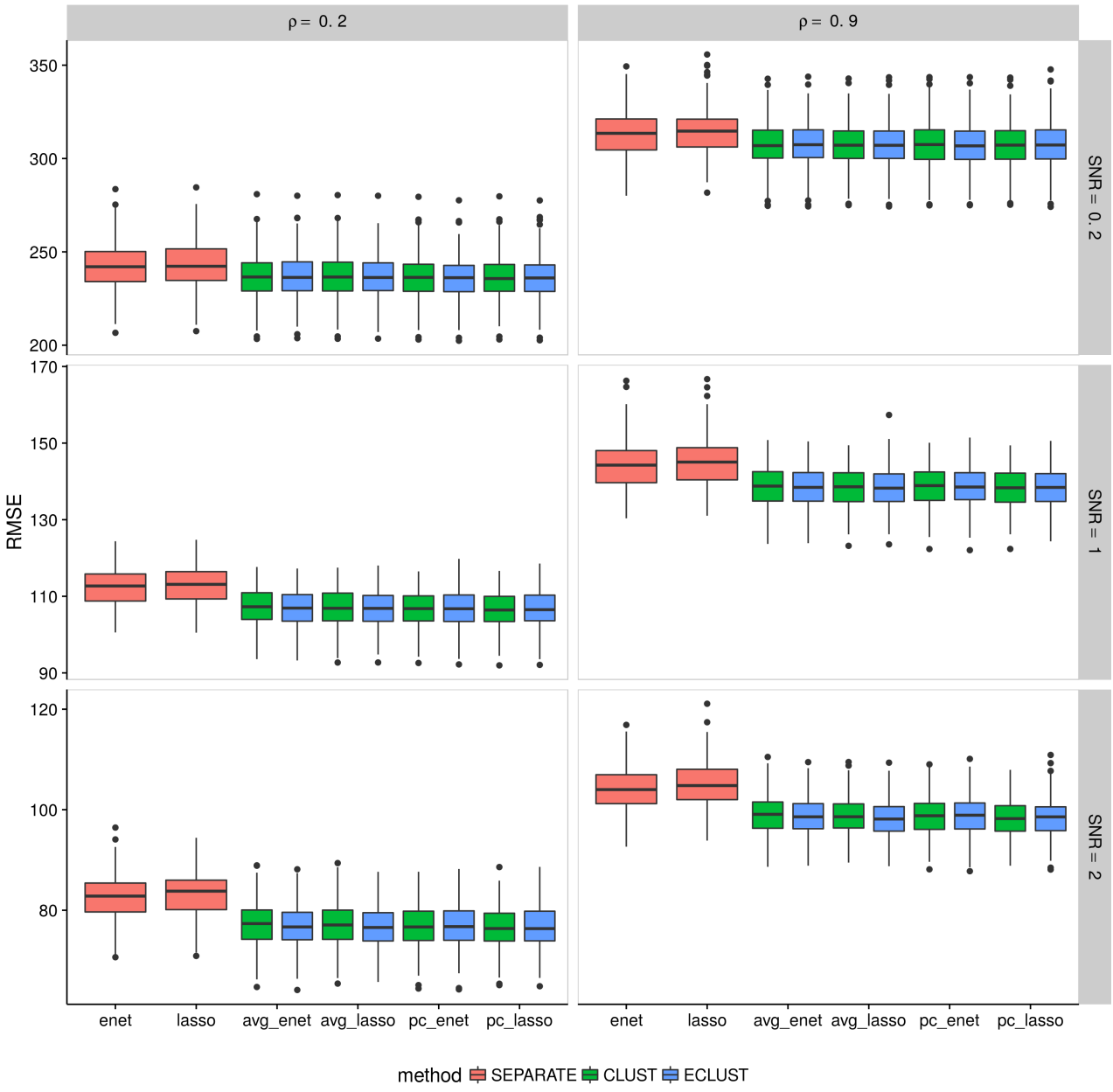


Figure A.26: Simulation 1 – Root mean squared error on an independent test set using the Correlation as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

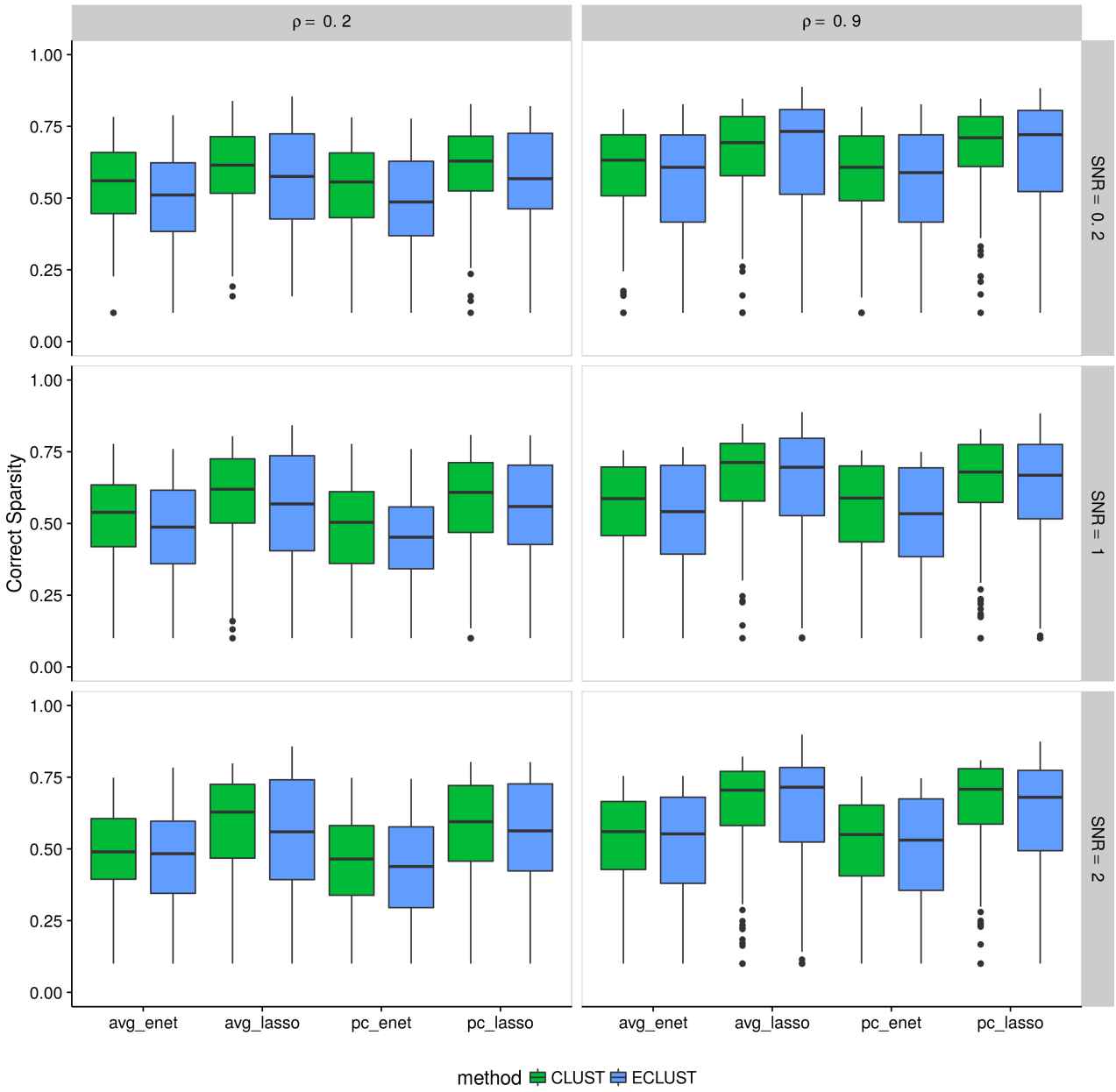


Figure A.27: Simulation 1 – Correct Sparsity based on the training set using the Pearson correlation as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

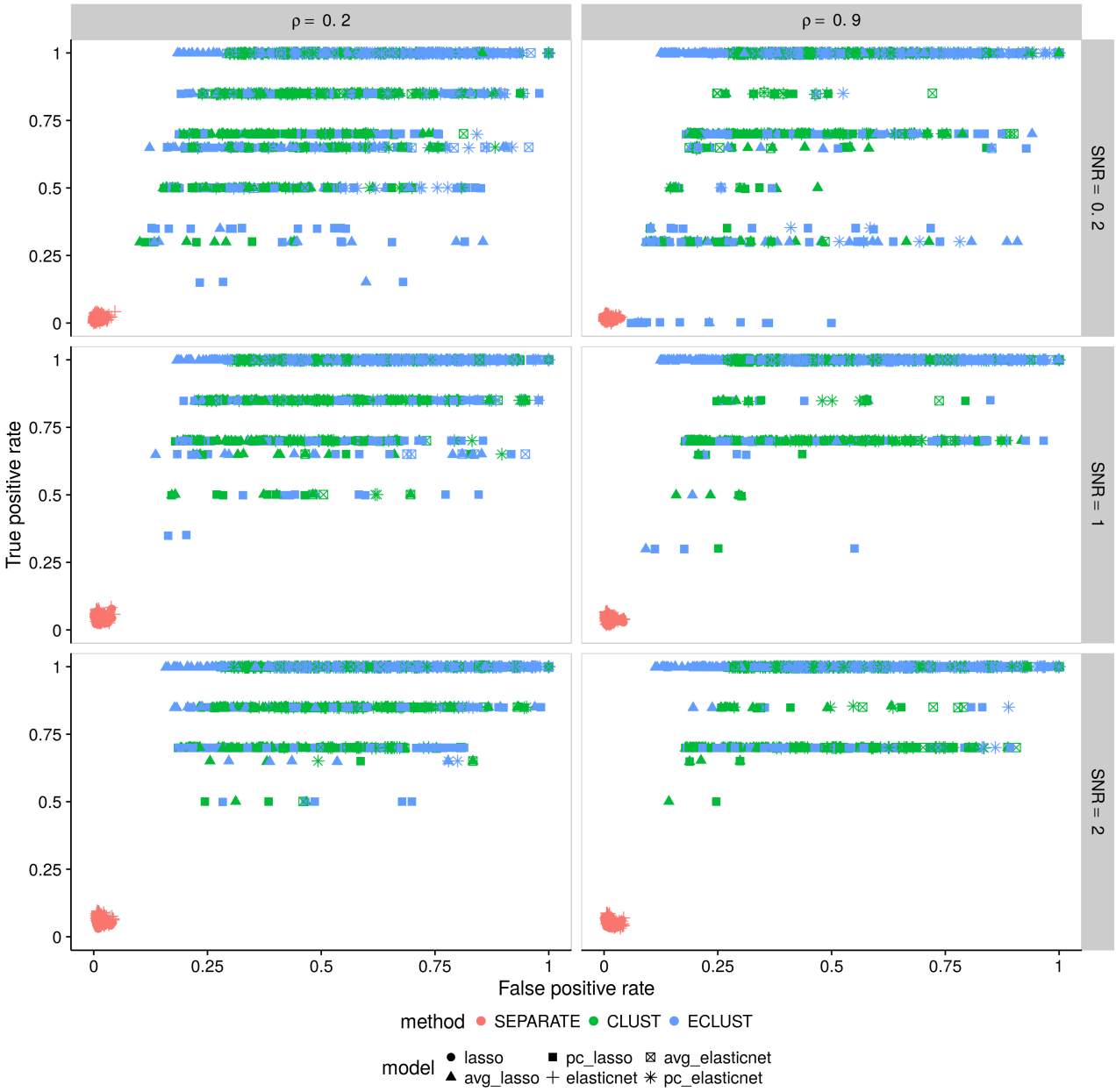


Figure A.28: Simulation 1 – True positive rate vs. false positive rate based on the training set using the Pearson correlation as a measure of similarity. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

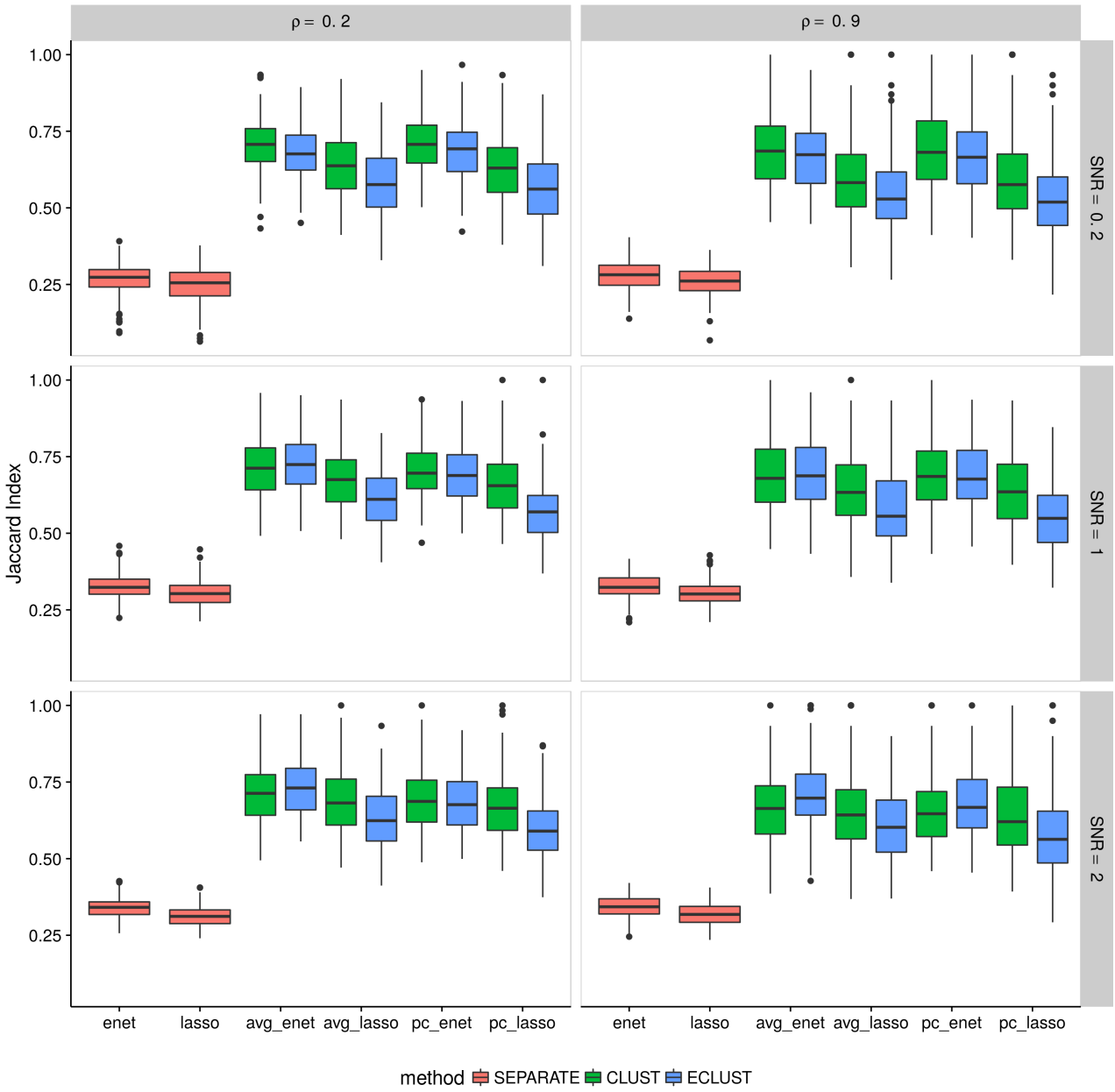


Figure A.29: Simulation 1 – Average Jaccard Index from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

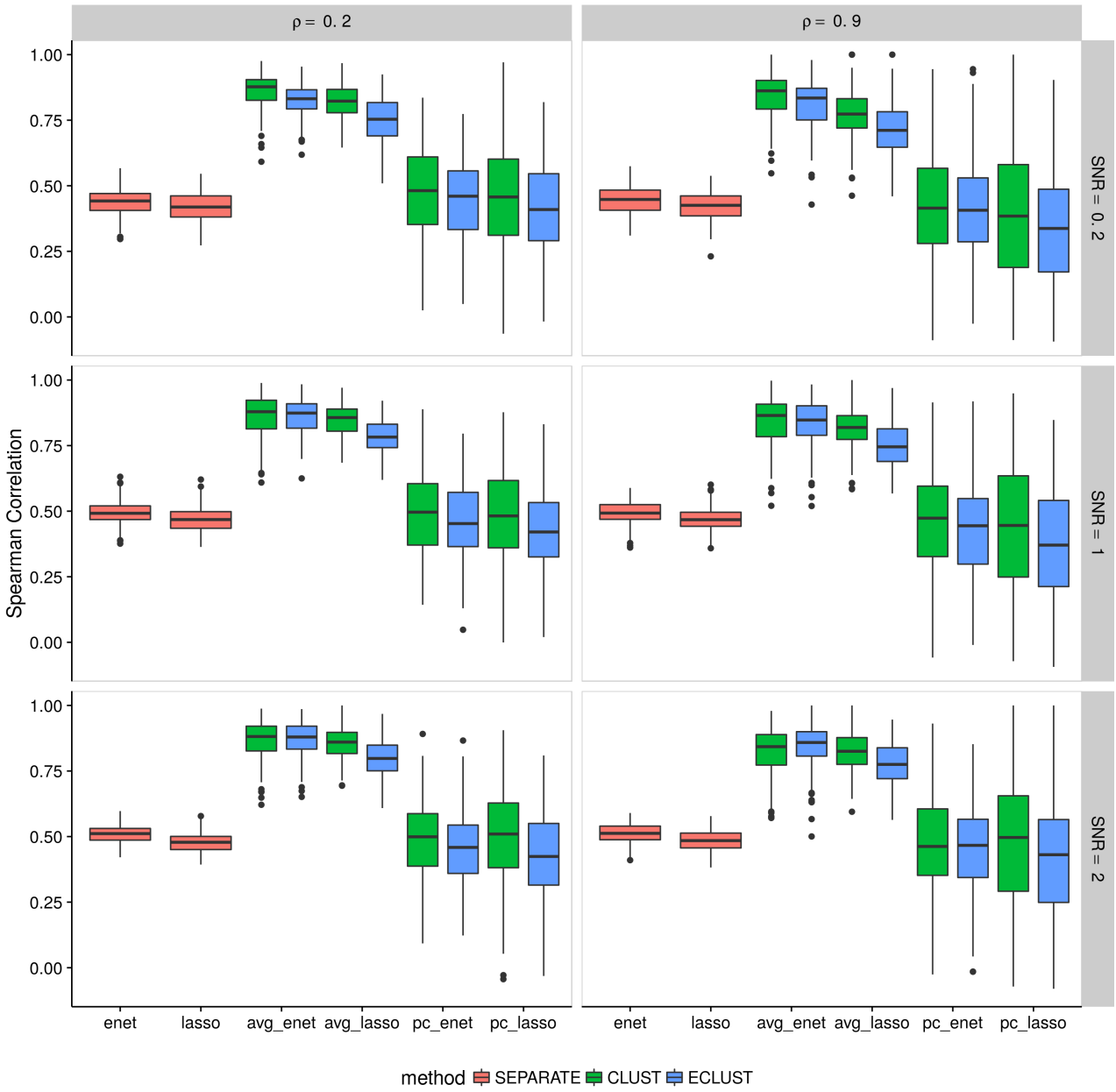


Figure A.30: Simulation 1 – Average Spearman correlation from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Spearman correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

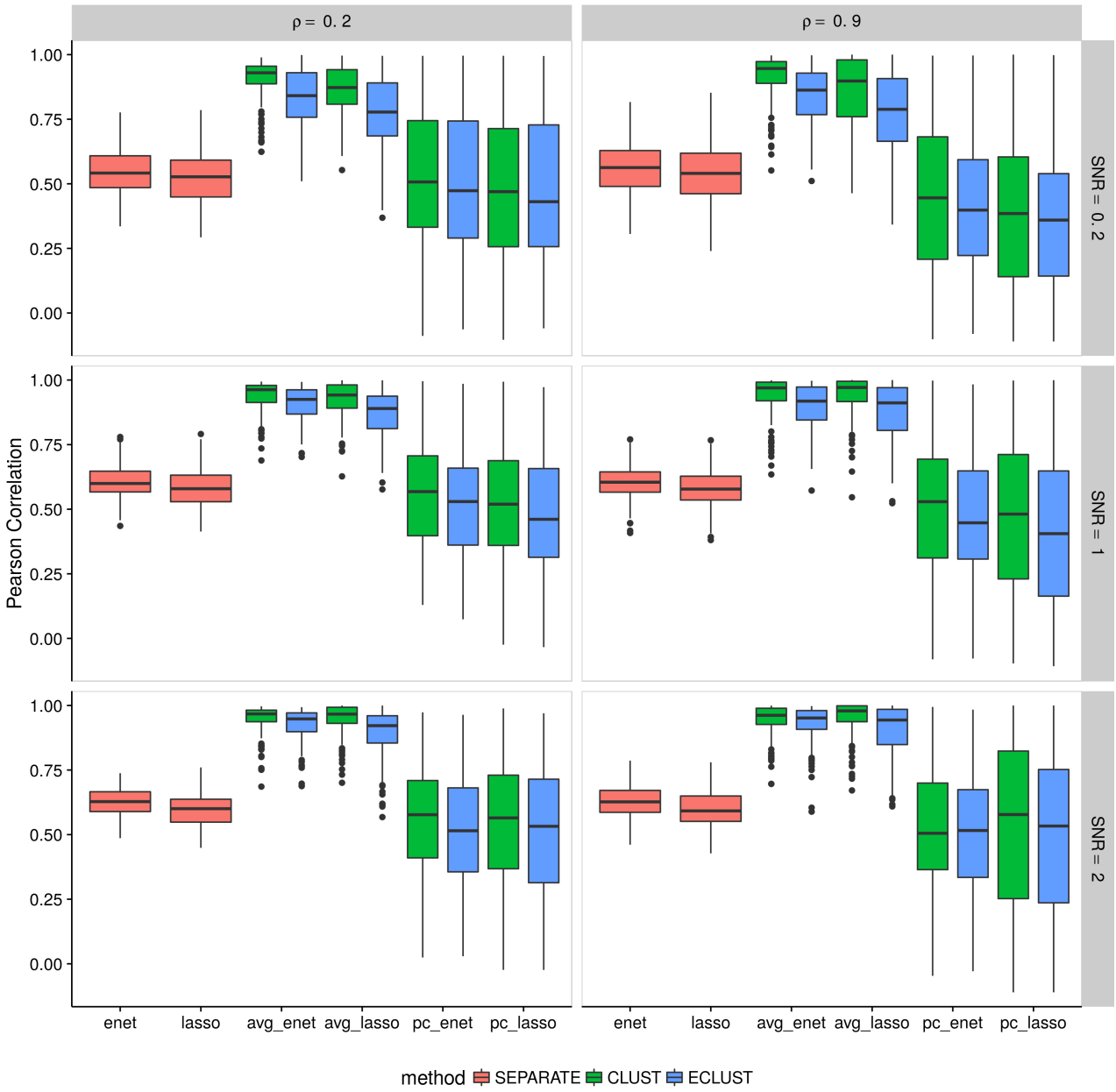


Figure A.31: Simulation 1 – Average Pearson correlation from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Pearson correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

Simulation 2

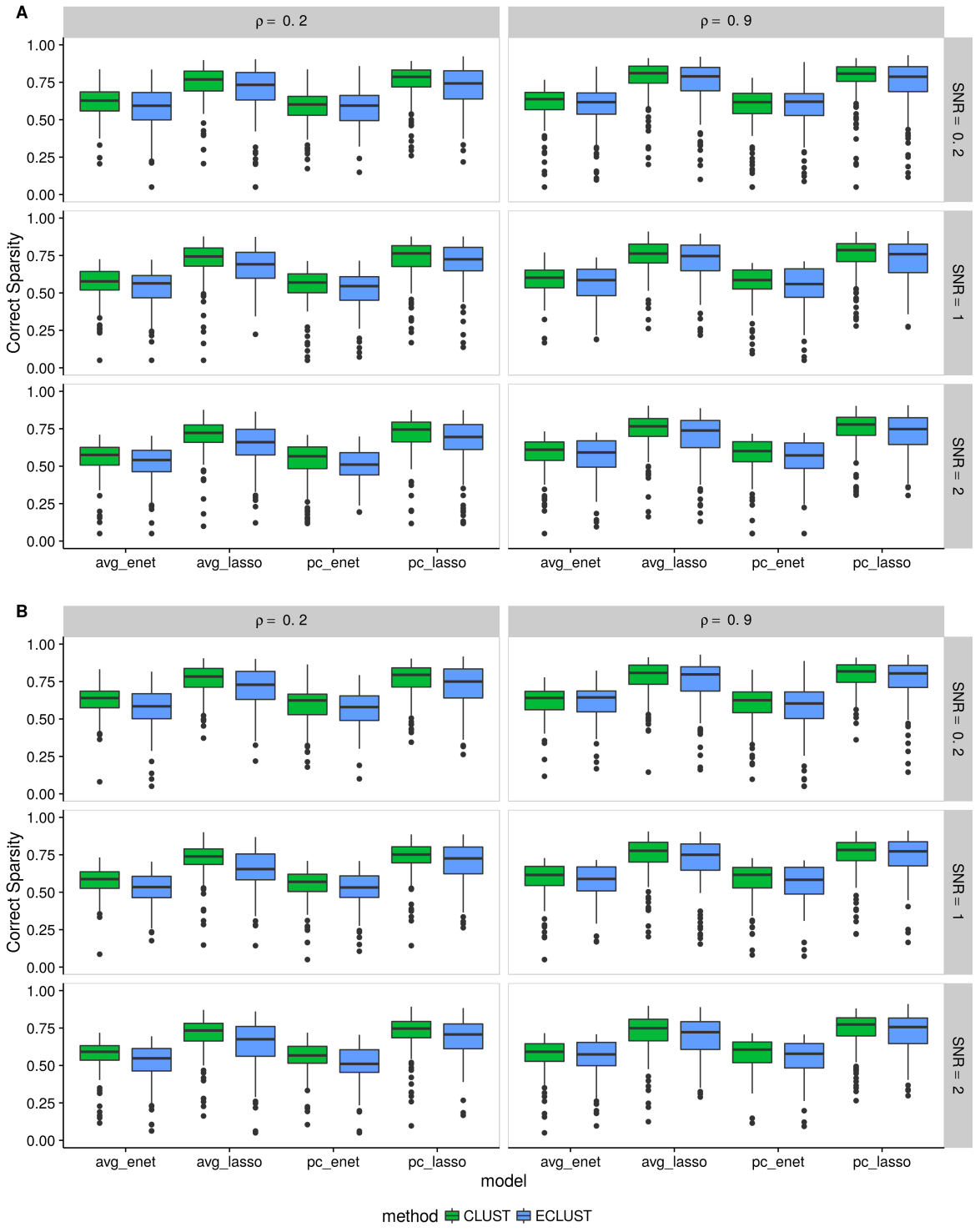


Figure A.33: Simulation 2 – Correct Sparsity based on the training set using the Pearson correlation as a measure of similarity from 200 simulation runs. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

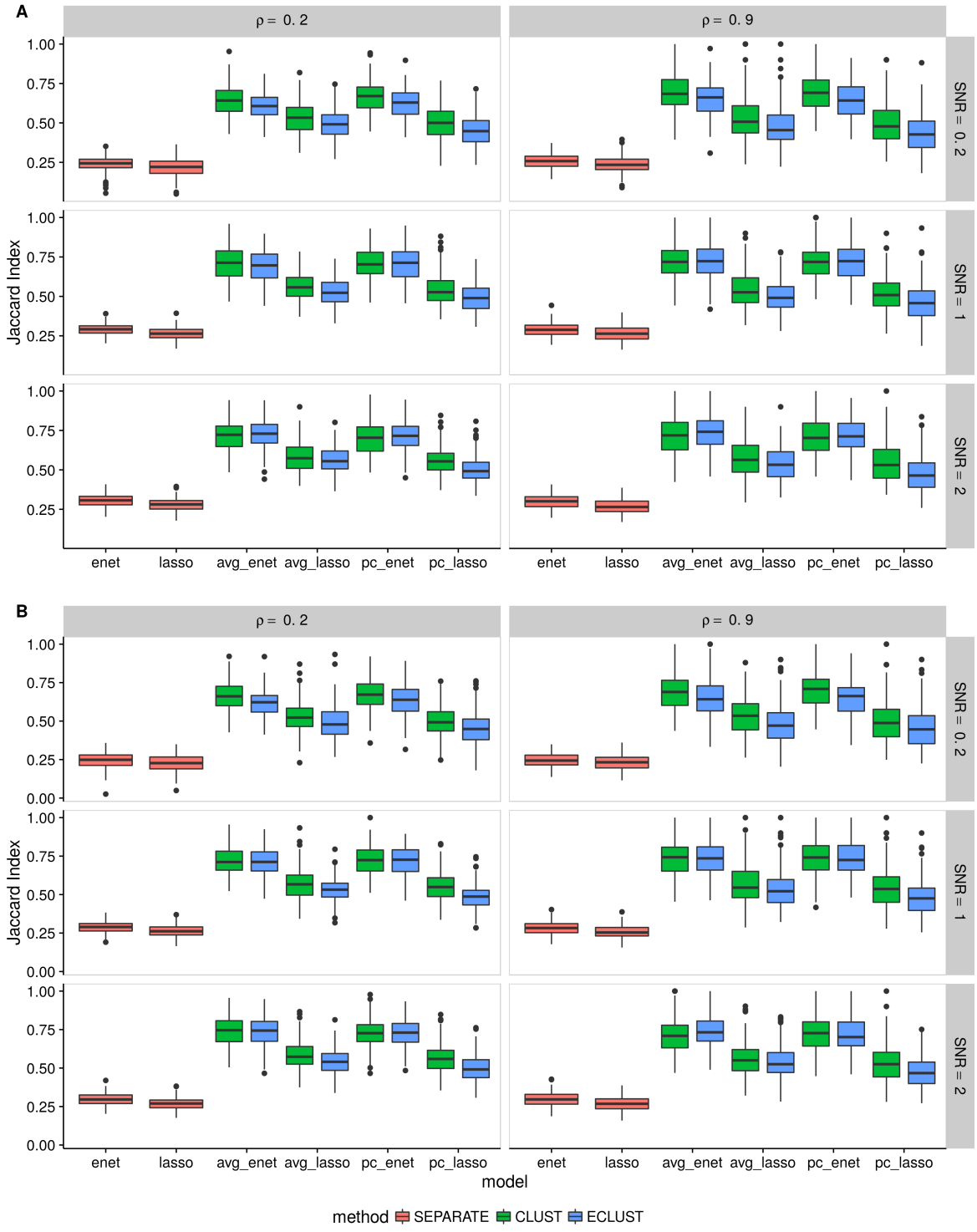


Figure A.35: Simulation 2 – Average Jaccard Index from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

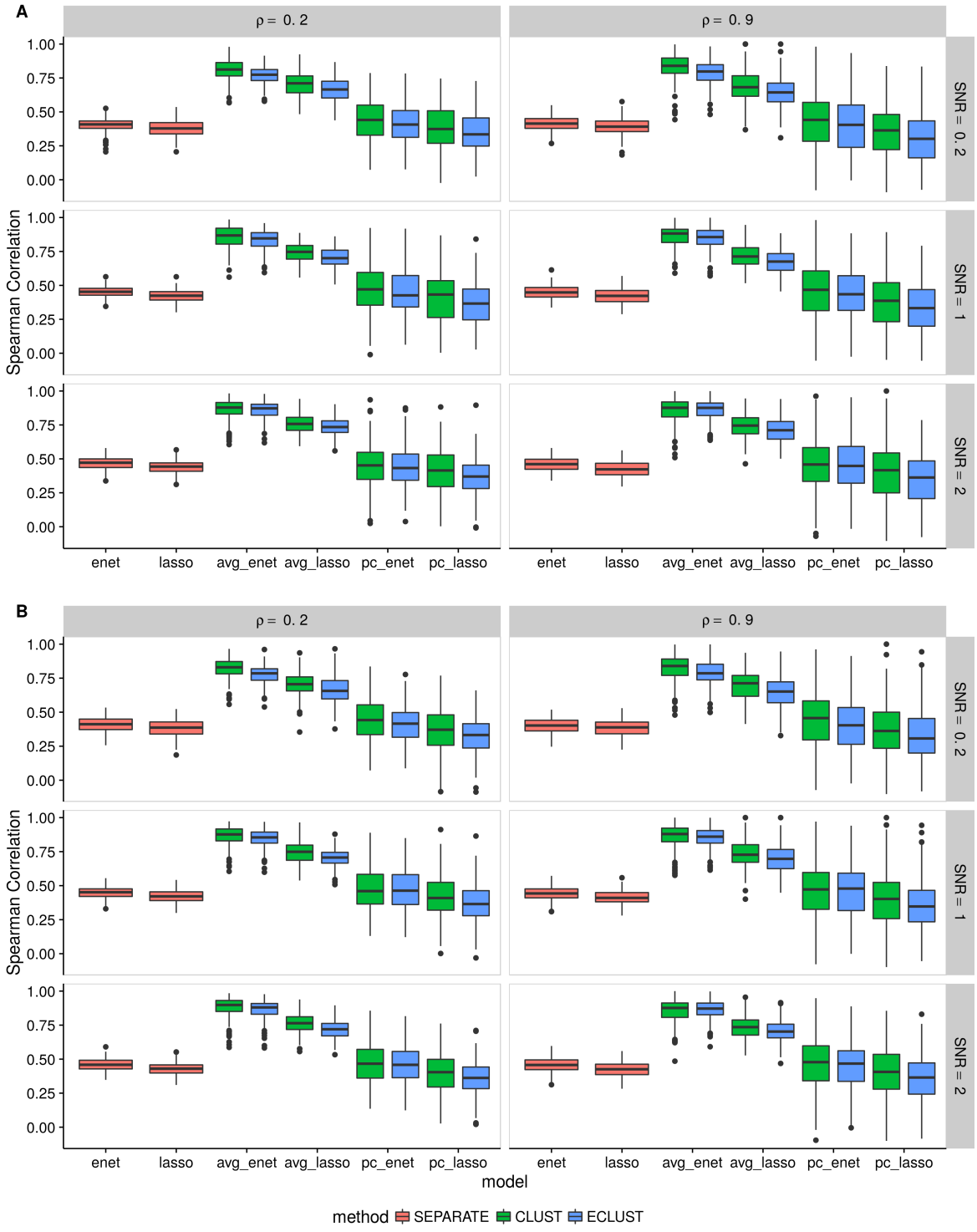


Figure A.36: Simulation 2 – Average Spearman correlation from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Spearman correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

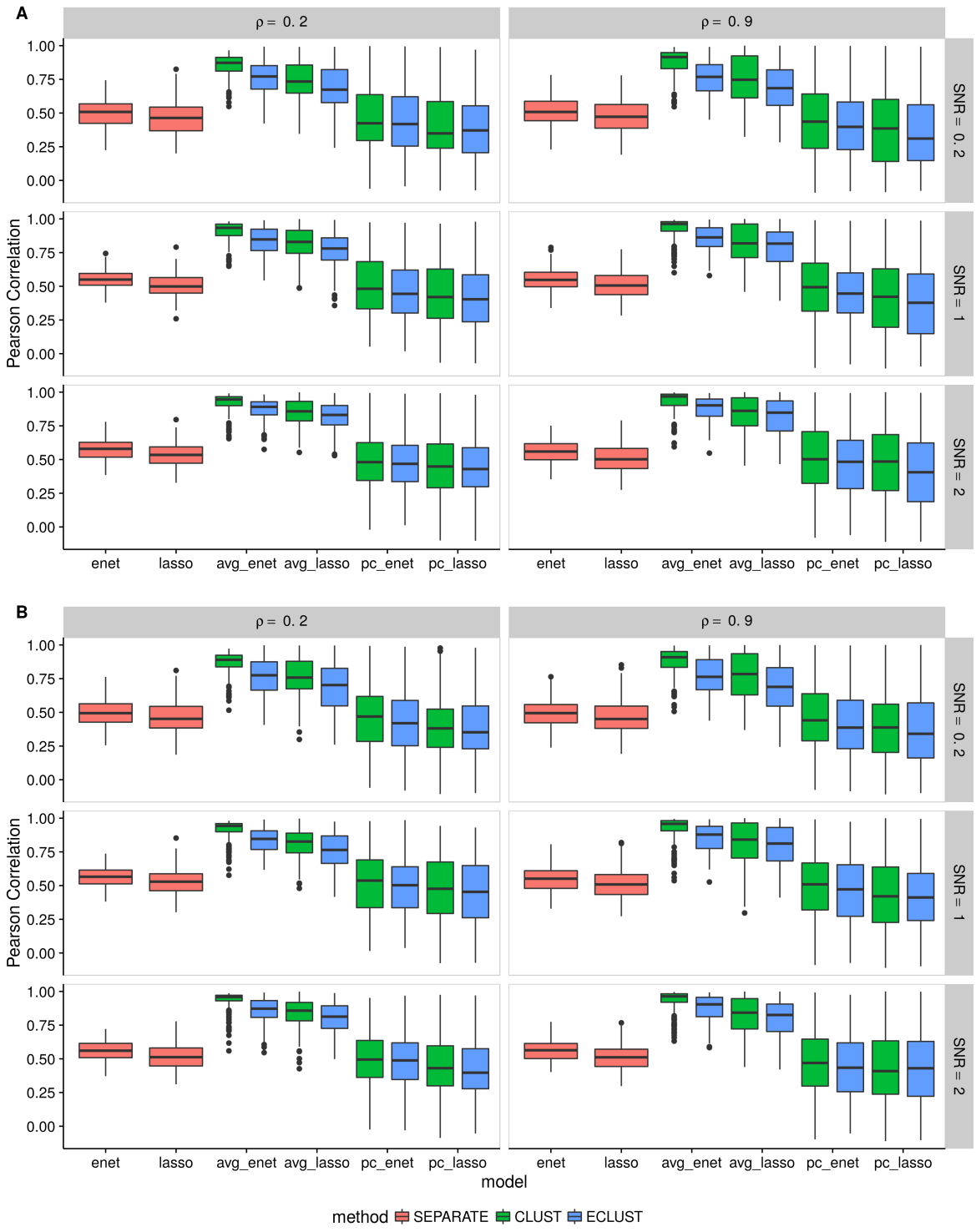


Figure A.37: Simulation 2 – Average Pearson correlation from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. We fit the model to each of the 10 CV folds resulting in 10 sets of estimated regression coefficients. We then calculate the Pearson correlation between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

Simulation 3

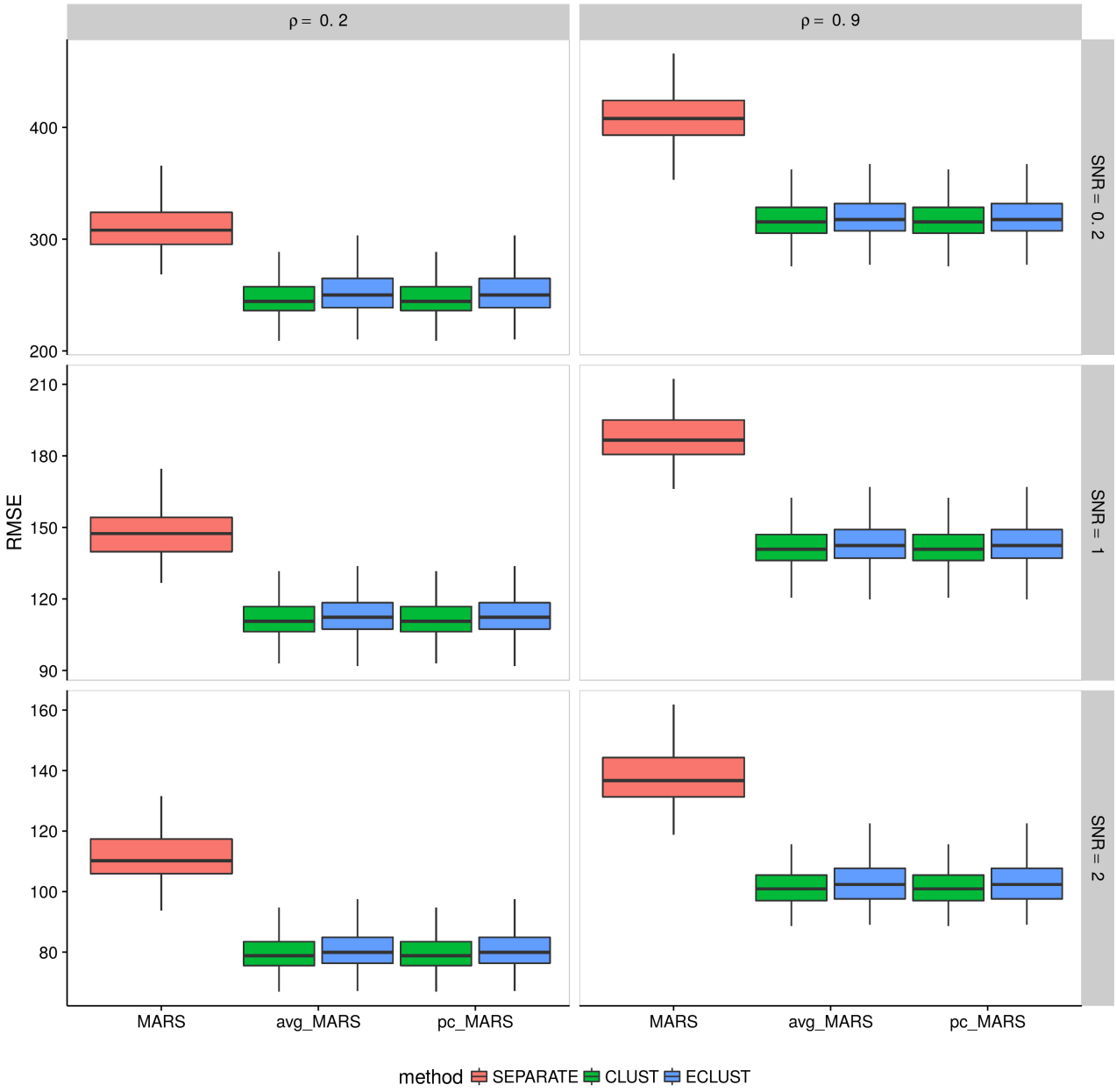


Figure A.38: Simulation 3 – Root mean squared error on an independent test set using the Pearson correlation as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

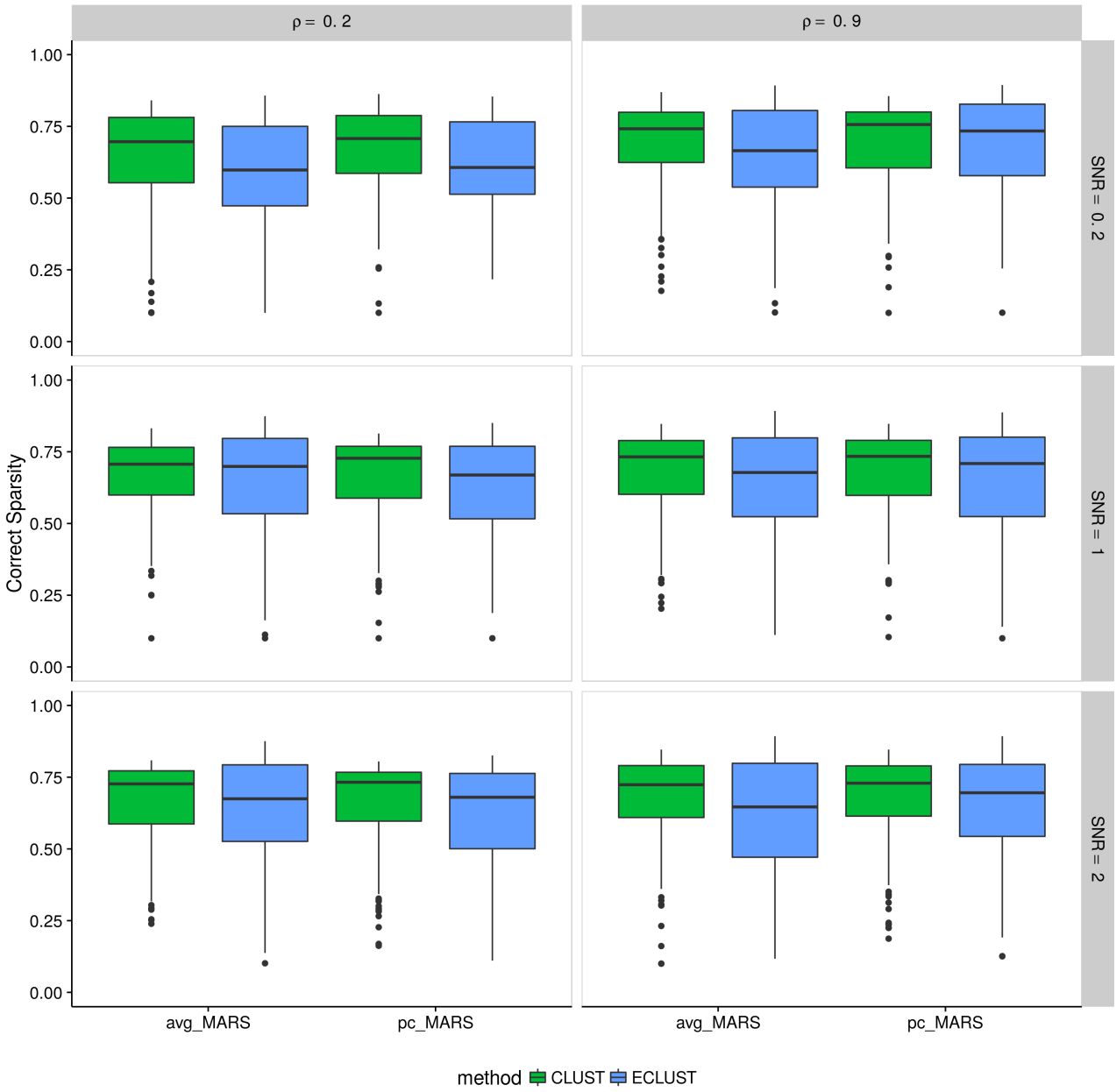


Figure A.39: Simulation 3 – Correct Sparsity based on the training set using the Pearson correlation as a measure of similarity from 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

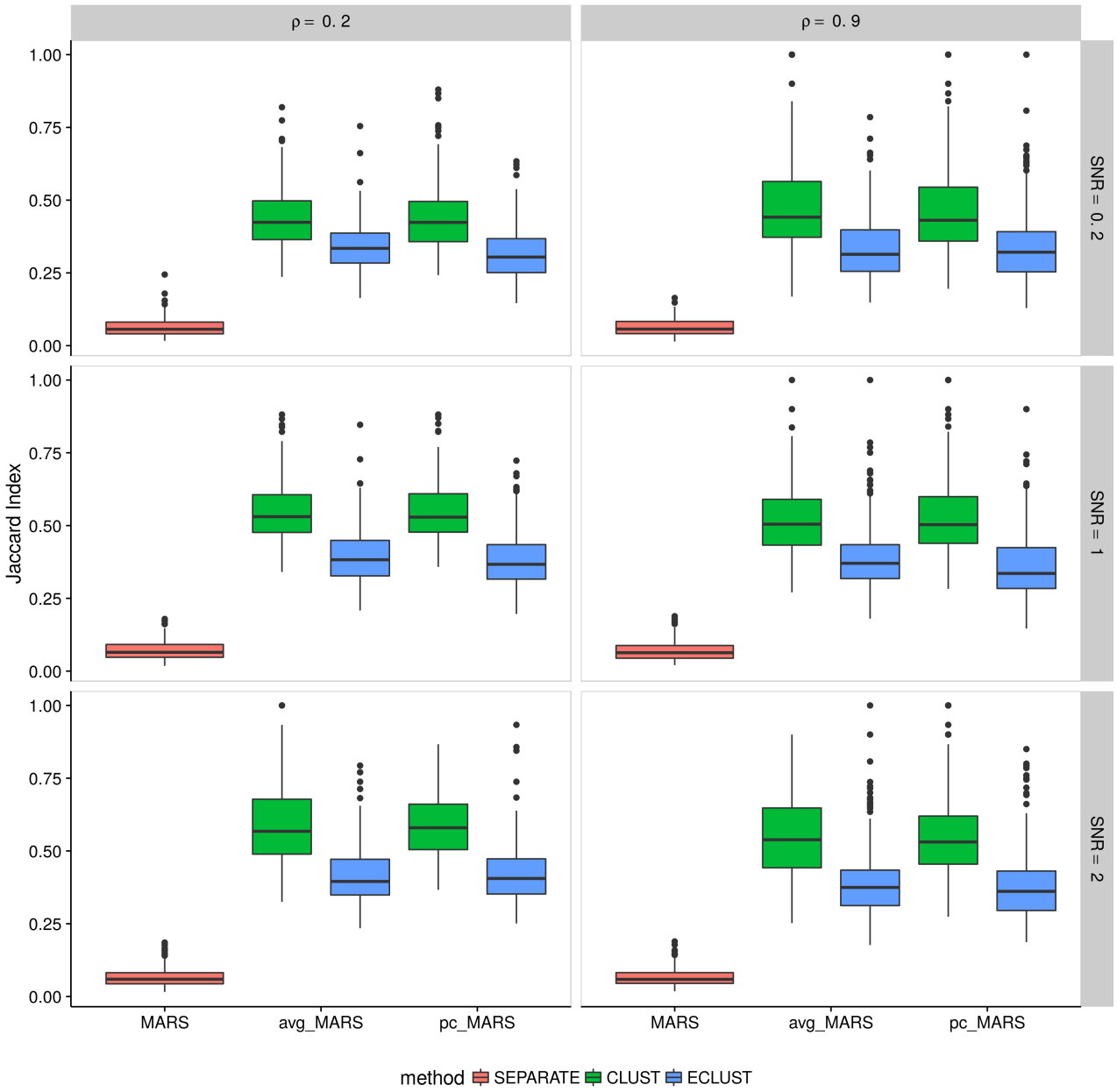
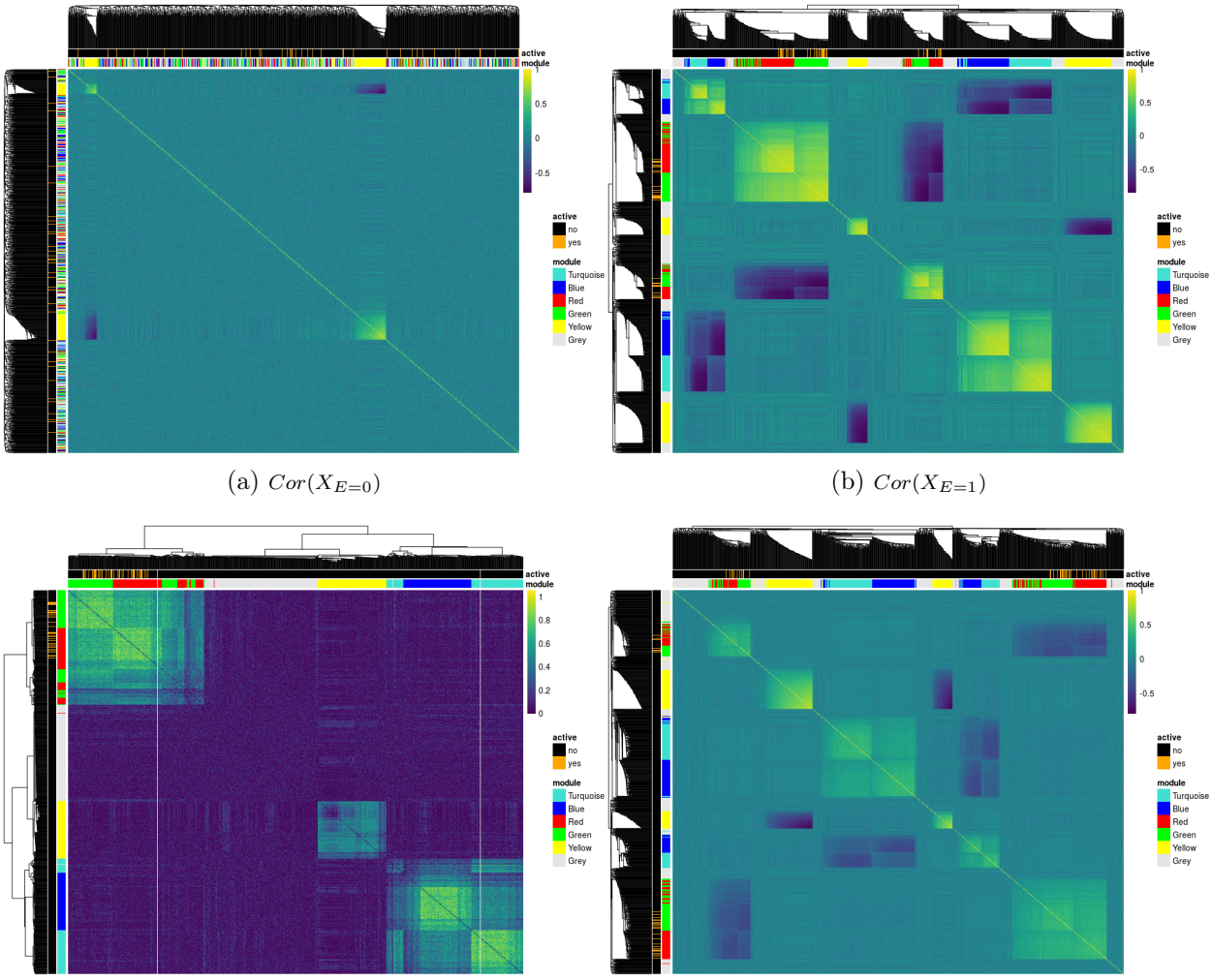


Figure A.41: Simulation 3 – Average Jaccard Index from 10 CV folds of the training set using the Pearson correlation as a measure of similarity. We fit the model to each of the 10 CV folds resulting in 10 sets of selected predictors. We then calculate the Jaccard Index between all $\binom{10}{2}$ possible combinations of these sets and take the average. This process is repeated for each of the 200 simulation runs. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

A.6 Visual Representation of Similarity Matrices

Pearson Correlation Matrix



(a) $Cor(X_{E=0})$

(b) $Cor(X_{E=1})$

(c) $|Cor(X_{E=1}) - Cor(X_{E=0})|$

(d) $Cor(X_{\text{all}})$

Figure A.42: Pearson correlation matrices of simulated predictors based on subjects with (a) $E = 0$, (b) $E = 1$, (c) their absolute difference and (d) all subjects. Dendrograms are from hierarchical clustering (average linkage) of one minus the correlation matrix for a, b, and d and the euclidean distance for c. The *module* annotation represents the true cluster membership for each predictor, and the *active* annotation represents the truly associated predictors with the response.

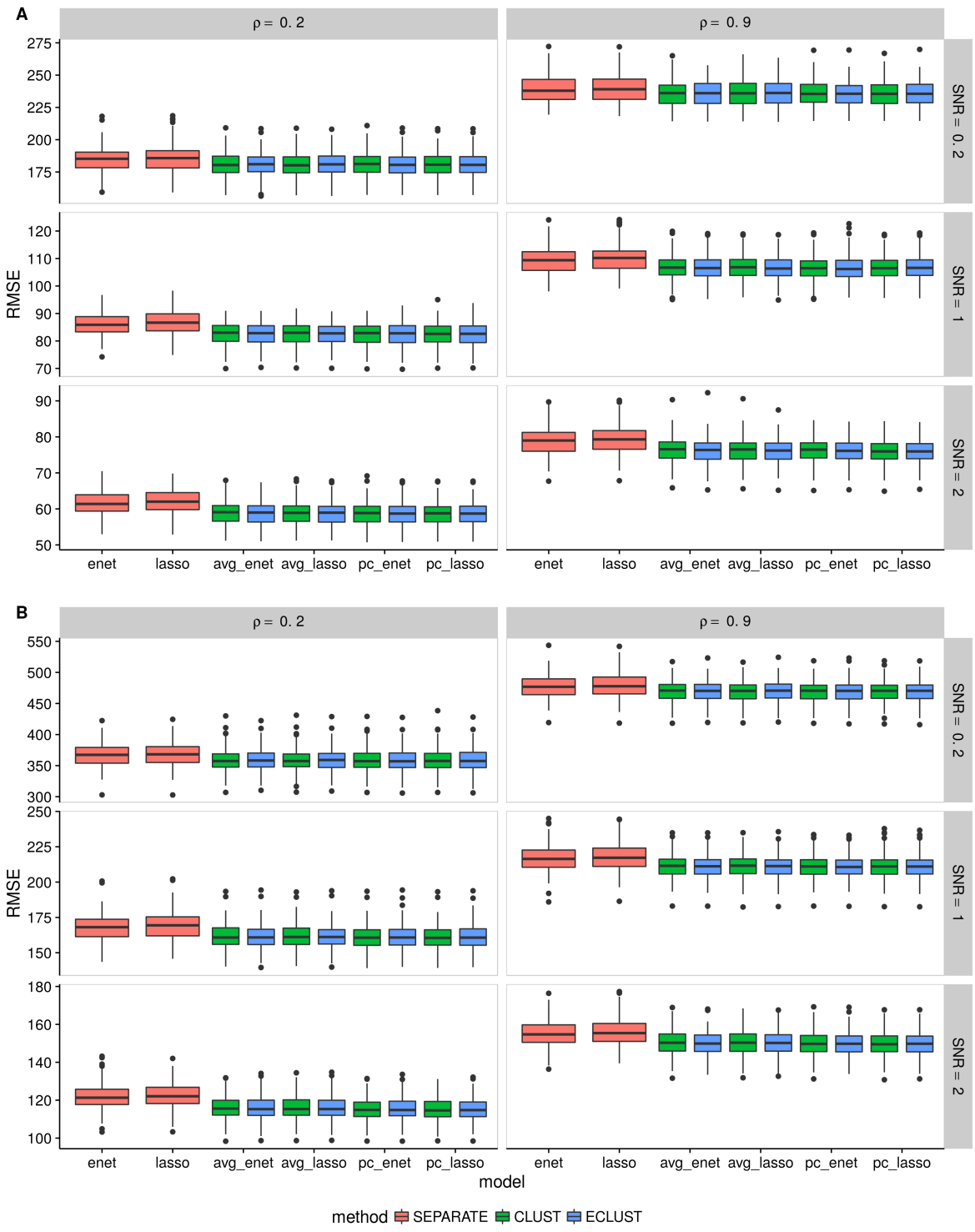


Figure A.32: Simulation 2 – Root mean squared error on an independent test set using the Pearson correlation as a measure of similarity from 200 simulation runs. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

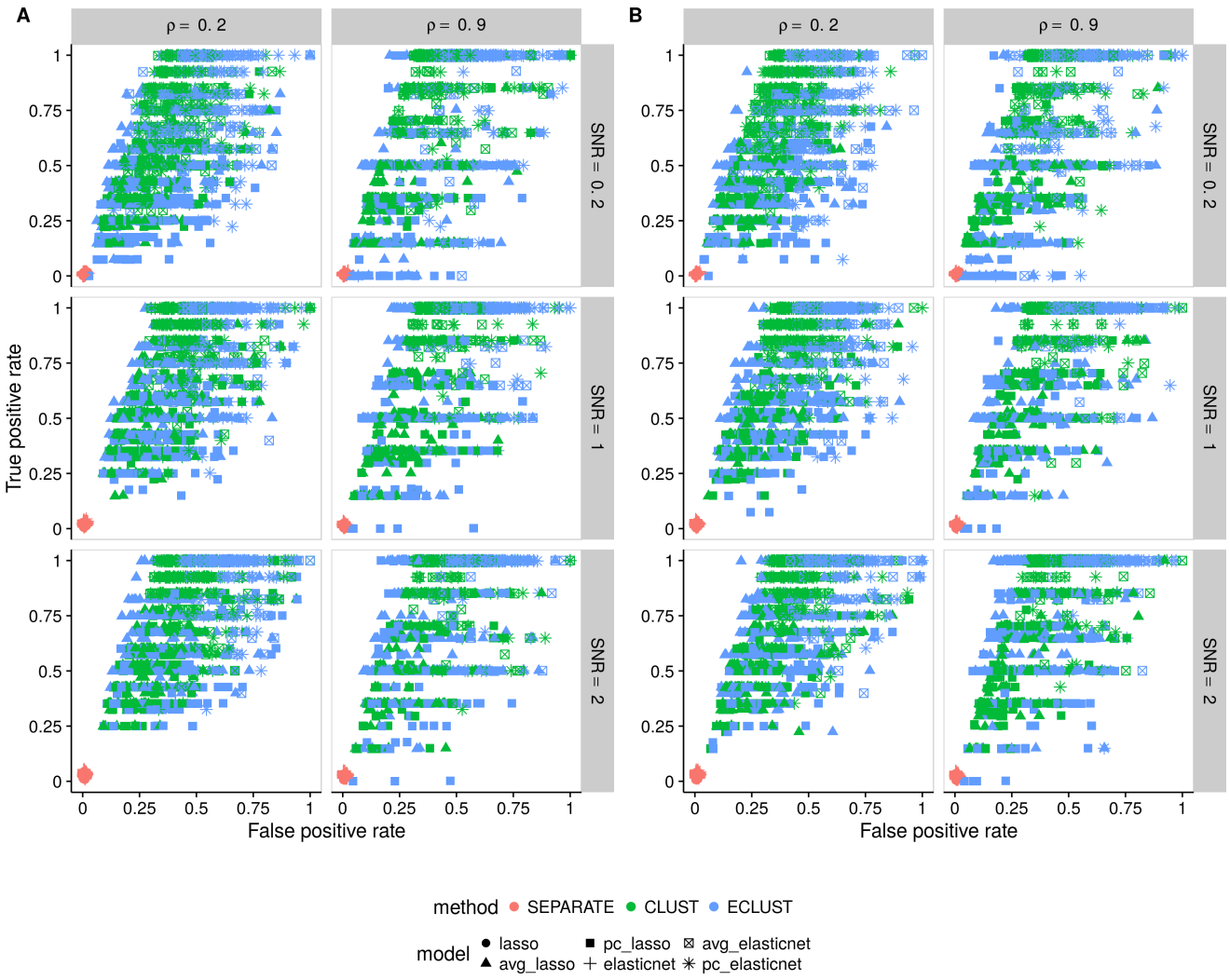


Figure A.34: Simulation 2 – True positive rate vs. false positive rate based on the training set using the Pearson correlation as a measure of similarity. (A) $\alpha_j \sim \text{Unif}\,[0.4, 0.6]$, (B) $\alpha_j \sim \text{Unif}\,[1.9, 2.1]$. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

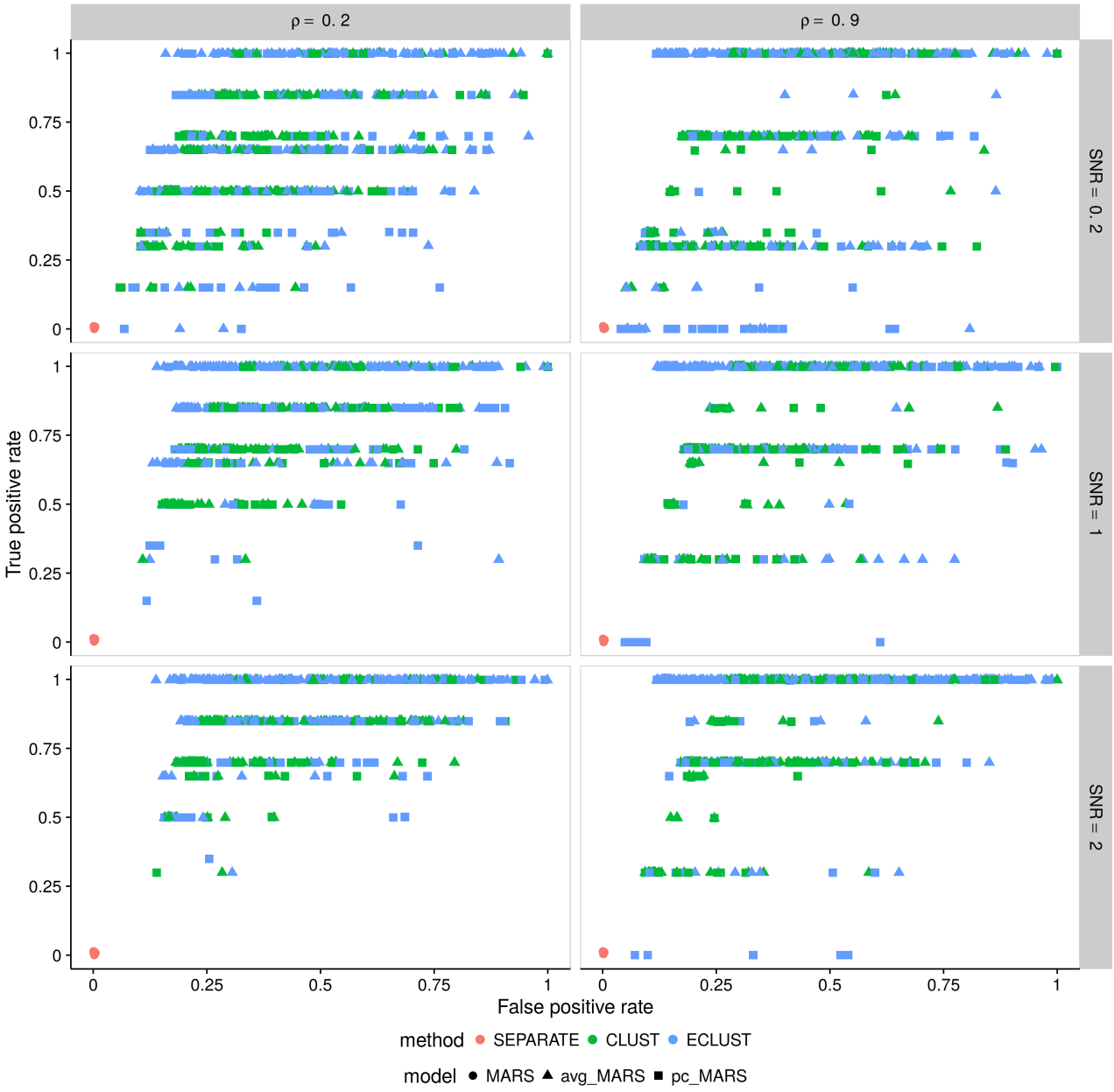


Figure A.40: Simulation 3 – True positive rate vs. false positive rate based on the training set using the Pearson correlation as a measure of similarity. Each point represents 1 simulation run (there are a total of 200 simulation runs). Vertical panels represent varying correlation between active clusters. Horizontal panels represent different signal-to-noise ratios.

References

Astle, W., Balding, D. J., et al. (2009). Population structure and cryptic relatedness in genetic association studies. *Statistical Science*, *24*(4), 451–471.

4, 5

Bondell, H. D., Krishna, A., & Ghosh, S. K. (2010). Joint variable selection for fixed and random effects in linear mixed-effects models. *Biometrics*, *66*(4), 1069–1077.

Byrd, R. H., Lu, P., Nocedal, J., & Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, *16*(5), 1190–1208.

Cordell, H. J., & Clayton, D. G. (2002). A unified stepwise regression procedure for evaluating the relative effects of polymorphisms within a gene using case/control or family data: application to hla in type 1 diabetes. *The American Journal of Human Genetics*, *70*(1), 124–141.

5

Ding, X., Su, S., Nandakumar, K., Wang, X., & Fardo, D. W. (2014). A 2-step penalized regression method for family-based next-generation sequencing association studies. In *Bmc proceedings* (Vol. 8, p. S25).

5

Eu-Ahsunthornwattana, J., Miller, E. N., Fakiola, M., Jeronimo, S. M., Blackwell, J. M., Cordell, H. J., ... others (2014). Comparison of methods to account for relatedness in genome-wide association studies with family-based data. *PLoS Genet*, *10*(7), e1004445.
4

Fan, Y., & Tang, C. Y. (2013). Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *75*(3), 531–552.

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, *33*(1), 1.
6, 11, 16

Hoggart, C. J., Whittaker, J. C., De Iorio, M., & Balding, D. J. (2008). Simultaneous analysis of all snps in genome-wide and re-sequencing association studies. *PLoS genetics*, *4*(7), e1000130.
4

Kang, H. M., Sul, J. H., Zaitlen, N. A., Kong, S.-y., Freimer, N. B., Sabatti, C., ... others (2010). Variance component model to account for sample structure in genome-wide association studies. *Nature genetics*, *42*(4), 348.
4

Li, J., Das, K., Fu, G., Li, R., & Wu, R. (2010). The bayesian lasso for genome-wide association studies. *Bioinformatics*, *27*(4), 516–523.
4

Lippert, C., Listgarten, J., Liu, Y., Kadie, C. M., Davidson, R. I., & Heckerman, D. (2011). Fast linear mixed models for genome-wide association studies. *Nature methods*, *8*(10), 833–835.

4

Manolio, T. A., Collins, F. S., Cox, N. J., Goldstein, D. B., Hindorff, L. A., Hunter, D. J., ... others (2009). Finding the missing heritability of complex diseases. *Nature*, *461*(7265), 747–753.

4, 7

Marchini, J., Cardon, L. R., Phillips, M. S., & Donnelly, P. (2004). The effects of human population structure on large genetic association studies. *Nature genetics*, *36*(5), 512.

4

Meier, L., Van De Geer, S., & Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *70*(1), 53–71.

Nishii, R. (1984). Asymptotic properties of criteria for selection of variables in multiple regression. *The Annals of Statistics*, 758–765.

Oualkacha, K., Dastani, Z., Li, R., Cingolani, P. E., Spector, T. D., Hammond, C. J., ... Greenwood, C. M. (2013). Adjusted sequence kernel association test for rare variants controlling for cryptic and family relatedness. *Genetic epidemiology*, *37*(4), 366–376.

5, 6

Pirinen, M., Donnelly, P., Spencer, C. C., et al. (2013). Efficient computation with a linear mixed model on large-scale data sets with applications to genetic studies. *The Annals of Applied Statistics*, *7*(1), 369–390.

Price, A. L., Patterson, N. J., Plenge, R. M., Weinblatt, M. E., Shadick, N. A., & Reich, D. (2006). Principal components analysis corrects for stratification in genome-wide association studies. *Nature genetics*, *38*(8), 904.

5

Rakitsch, B., Lippert, C., Stegle, O., & Borgwardt, K. (2013). A lasso multi-marker mixed model for association mapping with population structure correction. *Bioinformatics*, *29*(2), 206–214.

5, 8, 12

Ravasz, E., Somera, A. L., Mongru, D. A., Oltvai, Z. N., & Barabási, A.-L. (2002). Hierarchical organization of modularity in metabolic networks. *science*, *297*(5586), 1551–1555.

19

Schelldorfer, J., Bühlmann, P., DE, G., & VAN, S. (2011). Estimation for high-dimensional linear mixed-effects models using l1-penalization. *Scandinavian Journal of Statistics*, *38*(2), 197–214.

Song, M., Hao, W., & Storey, J. D. (2015). Testing for genetic associations in arbitrarily structured populations. *Nature genetics*, *47*(5), 550–554.

4

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 267–288.

5, 6

Tseng, P., & Yun, S. (2009). A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, *117*(1), 387–423.

Wakefield, J. (2013). *Bayesian and frequentist regression methods*. Springer Science & Business Media.

Wang, D., Eskridge, K. M., & Crossa, J. (2011). Identifying qtls and epistasis in structured plant populations using adaptive mixed lasso. *Journal of agricultural, biological, and environmental statistics*, *16*(2), 170–184.

5

Yang, J., Benyamin, B., McEvoy, B. P., Gordon, S., Henders, A. K., Nyholt, D. R., ... others (2010). Common snps explain a large proportion of the heritability for human height. *Nature genetics*, *42*(7), 565.

4

Yang, J., Zaitlen, N. A., Goddard, M. E., Visscher, P. M., & Price, A. L. (2014). Advantages and pitfalls in the application of mixed-model association methods. *Nature genetics*, *46*(2), 100.

6

Yang, Y., & Zou, H. (2015). A fast unified algorithm for solving group-lasso penalize learning problems. *Statistics and Computing*, *25*(6), 1129–1141.

6

Yu, J., Pressoir, G., Briggs, W. H., Bi, I. V., Yamasaki, M., Doebley, J. F., ... others (2006). A unified mixed-model method for association mapping that accounts for multiple levels of relatedness. *Nature genetics*, *38*(2), 203.

5

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*(1), 49–67.

6

Zhang, B., & Horvath, S. (2005). A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, *4*(1).

19

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, *101*(476), 1418–1429.

5

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *67*(2), 301–320.

6

Zou, H., Hastie, T., Tibshirani, R., et al. (2007). On the "degrees of freedom" of the lasso. *The Annals of Statistics*, *35*(5), 2173–2192.