Variable Selection Methods Comparison

Hongjie Liu, Jiajun Tao, Shaohan Chen

2023-02-27

Outline

- Background
- Statistical methods to be studied
- Objective
- Simulation
- Experiment Settings and Scenario
- Model Evaluation
- Missing Weak Predictor Analysis

Background

- Variable selection methods help to optimize models in high-dimensional settings where we need to select predictors that balance fitness and complexity.
- The presence of weak predictors is a problem that plagues traditional variable selection methods.

Statistical Methods to be Studied

Step-wise forward method

- Starts with the empty model, and iteratively adds the variables that best improve the model fit. That is often done by sequentially adding predictors with the largest reduction in AIC. For linear models,

$$AIC = n \log\left(\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2/n\right) + 2p.$$

Automated LASSO regression

- Estimates model parameters by optimizing a penalized loss function:

$$\min_{\beta} \frac{1}{2n}\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i\beta)^2 + \lambda\sum_{k=1}^{p}|\beta_k|.$$

Objectives

(1) Evaluate the effectiveness of both methods in identifying weak and strong predictors.

(2) Examine how the absence of "weak" predictors affects parameter estimations.

Types of Signals

- Strong signals

$$S_{strong} = \{j : |\beta_j| > c\sqrt{\log(p)/n} \text{ for some } c > 0,\ 1 \le j \le p\}$$

- Weak-but-correlated (WBC) signals

$$S_{WBC} = \{j : 0 < |\beta_j| \le c\sqrt{\log(p)/n} \text{ and } \mathrm{corr}(X_j, X_j') \neq 0 \text{ for some } j' \in S_1,\ 1 \le j \le p\}$$

- Weak-and-independent (WAI) signals

$$S_{WAI} = \{j : 0 < |\beta_j| \le c\sqrt{\log(p)/n} \text{ and } \mathrm{corr}(X_j, X_j') = 0 \text{ for all } j' \in S_1,\ 1 \le j \le p\}$$

- Null signals: $S_{null} = \{j : \beta_j = 0,\ 1 \le j \le p\}$

Types of Signals

Thus, p predictors can be partitioned as

$$\{1, \cdots, p\} = S_{strong} \cup S_{WBC} \cup S_{WAI} \cup S_{null}.$$

- We assume that $|S_{strong}| = p_S$, $|S_{WBC}| = p_{WBC}$, $|S_{WAI}| = p_{WAI}$.
- The number of true predictors $p_S + p_{WBC} + p_{WAI}$ should be less than n.

Data Generation

- Normality assumption

$$\mathbf{y} \sim N(\mathbf{X}\beta, \sigma^2\mathbf{I})$$

- For $j \in S_{strong}$, $\beta_j = 20$; For $j \in S_{WBC} \cup S_{WAI}$, $\beta_j = 0.5$.
- We choose $c = 20$ so that $0.5 \leq c\sqrt{\log p/n} < 20$ for all scenarios to be investigated.
- For the error term, we set $\sigma = 8$.

Data Generation - Design Matrix

- We assume
$$\mathbf{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$
- All predictors are standardized. Then we have $\boldsymbol{\mu} = \mathbf{0}$ and $\Sigma_{i,i} = 1$ for all i.
- We set $p_{WBC} \geq p_{strong}$. For each strong predictor (except one of them), we set $[p_{WBC}/p_{strong}]$ WBC predictors to be correlated with it. Each WBC predictor is set to be correlated with one and only one strong predictor.
- All other elements of $\boldsymbol{\Sigma}$ are 0.

We use the `MASS::mvrnorm` function to generate data following a multivariate normal distribution.

Data Generation - Simulation Code

```
corr_matrix = matrix(rep(0, len = p^2), nrow = p)
corr_num = pwbc %/% ps
for (i in 1:(ps - 1)) {
  for (j in (ps + 1 + (i - 1)*corr_num):(ps + i*corr_num))
    corr_matrix[i, j] = corr
    corr_matrix[j, i] = corr
  }
}
for (j in (ps + 1 + (ps - 1)*corr_num):(ps + pwbc)) {
    corr_matrix[ps, j] = corr
    corr_matrix[j, ps] = corr
}
diag(corr_matrix) = 1
X = MASS::mvrnorm(n, mu = rep(0, p), Sigma = corr_matrix)
beta = c(rep(20, ps), rep(0.5, pwbc + pwai),
         rep(0, p - ps - pwbc - pwai))
Y = X %*% beta + rnorm(n, mean = 0, sd = 8)
```

Investigation Settings and Scenarios

Fixed

- Number of parameters: $p = 100$
- Ratio of true and null signals: $2 : 3$
- Correlation between strong and WBC: $corr = 0.4$

Unfixed

- Number of observations:
  $n = 100$ (high dimensional), $500, 2000$
- Ratio of strong, WBC, and WAI signals:
  $p_{strong} : p_{WBC} : p_{WAI} = 1 : 4 : 5$ and $3 : 3 : 4$

Evaluation Metrics

Define true predictors as positive and null predictors as negative

Signal Identification

- **Complexity:** Number of Selected Parameters
- **Sensitivity:** $\frac{TP}{TP+FN}$
- **Specificity:** $\frac{TN}{TN+FP}$
- **F1-score:** $\frac{2 \cdot \text{sensitivity} \cdot \text{precision}}{\text{sensitivity} + \text{precision}}$
- **Accuracy:** $\frac{TP+TN}{TP+TN+FP+FN}$

Parameter Estimation

- **MSE:** $\frac{1}{p}\sum_{i=1}^{p}(\hat{\beta}_i - \beta_i)^2$

Signal Identification Performance - Complexity

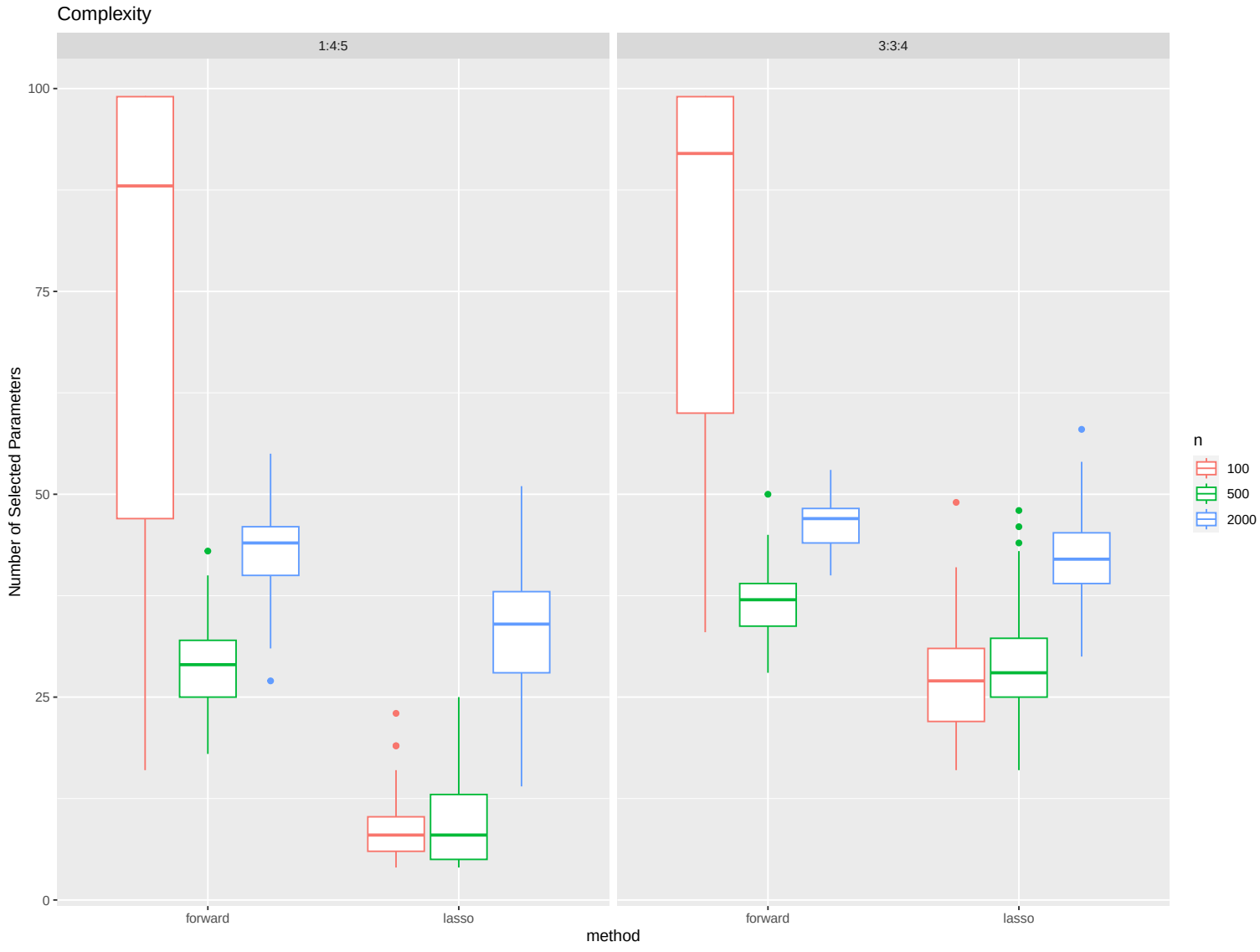


Figure 1: Model Complexity

Signal Identification Performance - Complexity Exploration

- When in high dimensional scenario, forward selection tends to select nearly all of the predictors but Lasso does not
- Lasso tends to select much fewer predictors than forward selection, and the parameters it select will increase as n increases
- As n increases, the predictors that two models select are more precise(closer to true predictor number 40)

Signal Identification Performance - Sensitivity

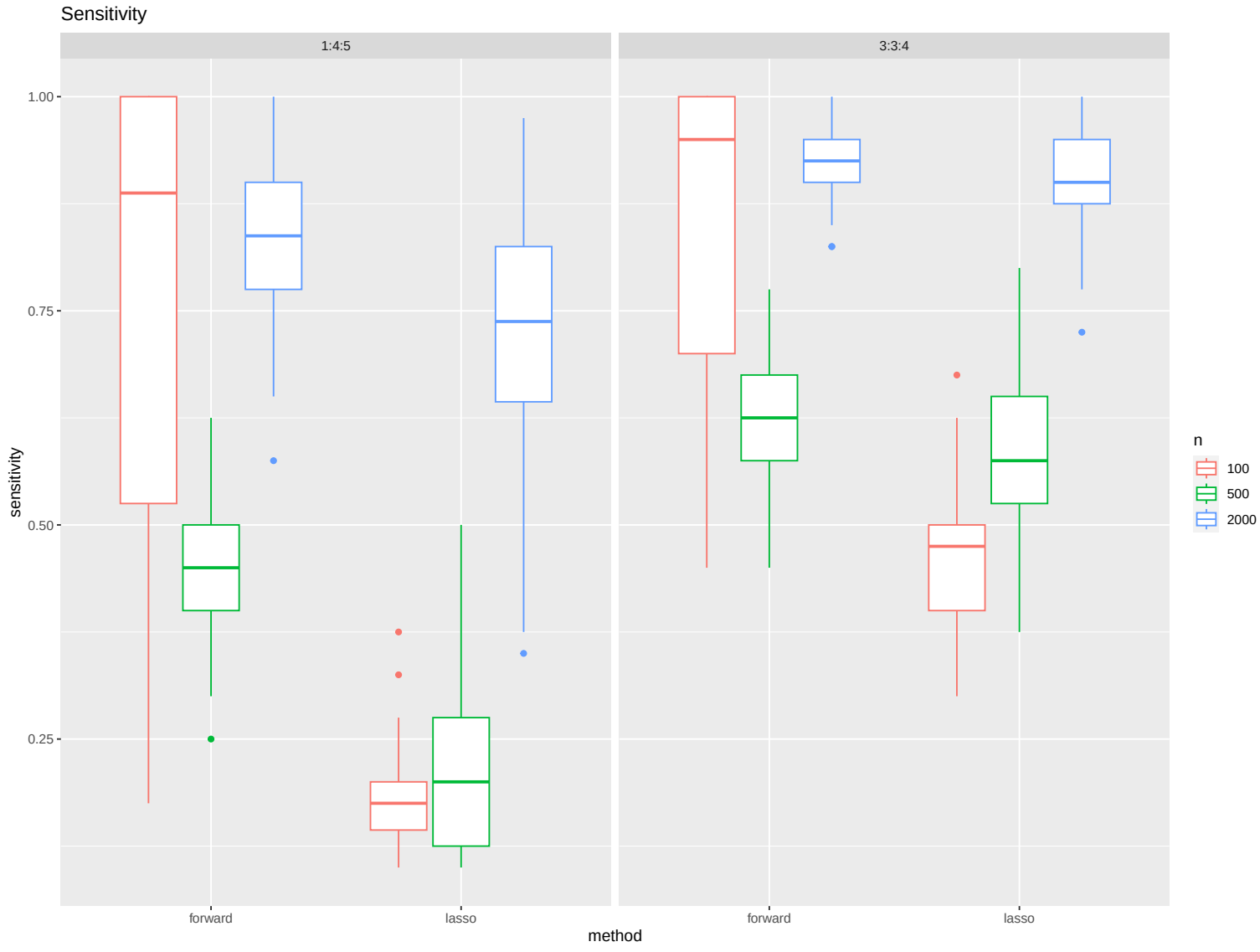


Figure 2: Sensitivity Performance

Signal Identification Performance - Sensitivity

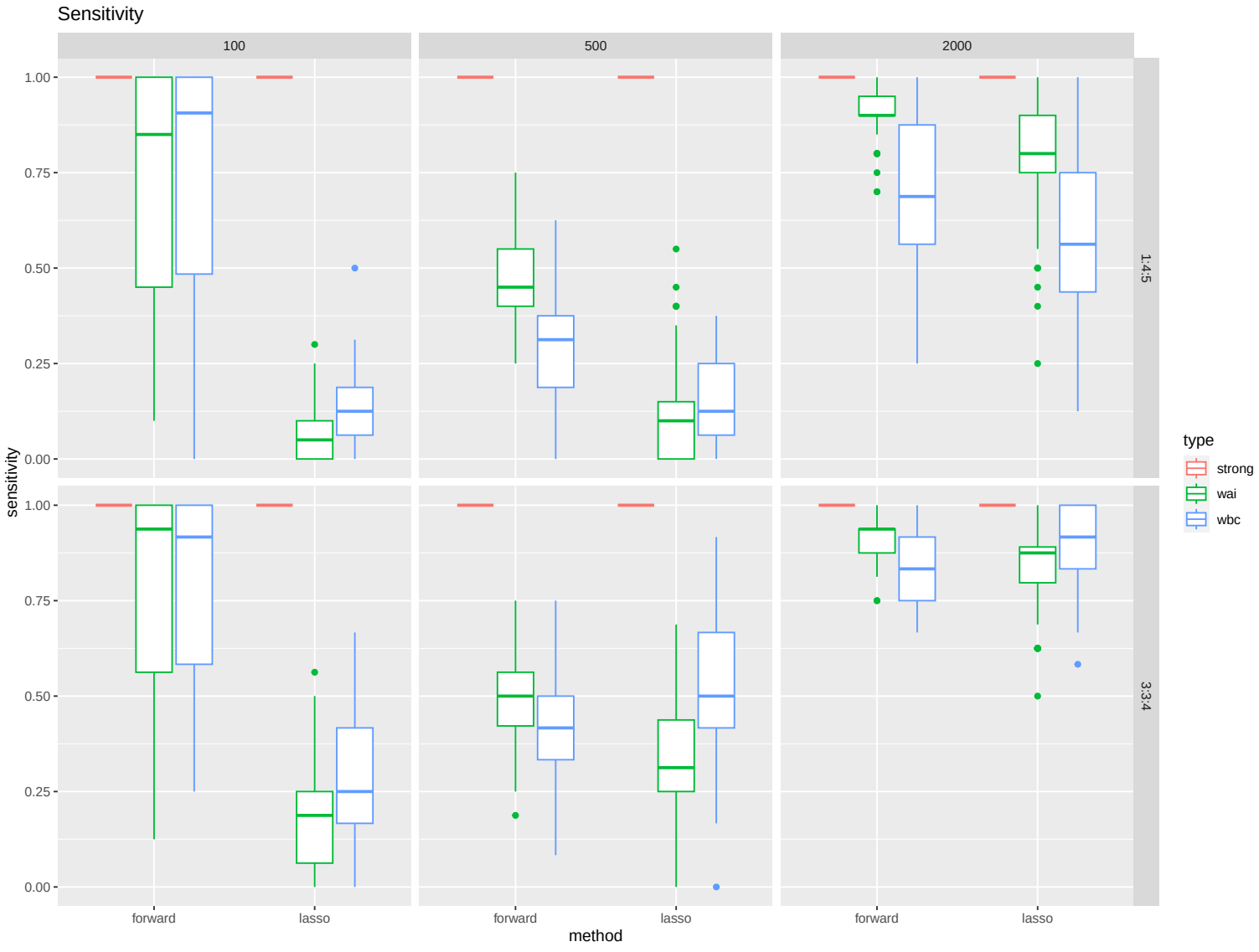


Figure 3: Sensitivity of True Signals

Signal Identification Performance - Sensitivity Exploration

- Forward selection are highly sensitive in high dimensional case(n=100) while Lasso does not
- Overall, the sensitivity of two models increases as n increases
- Both models are sensitive in selecting strong signals. But when it comes to weak predictors, Lasso is much less sensitive in high dimensional scenario than forward selection.
- When n increases, the sensitivity discrepancy of selecting weak predictors between two models becomes smaller. But still, forward selection is overall more sensitive than Lasso
- The sensitivity discrepancy between two models are smaller when the ratio of strong predictors becomes larger

Signal Identification Performance - Specificity

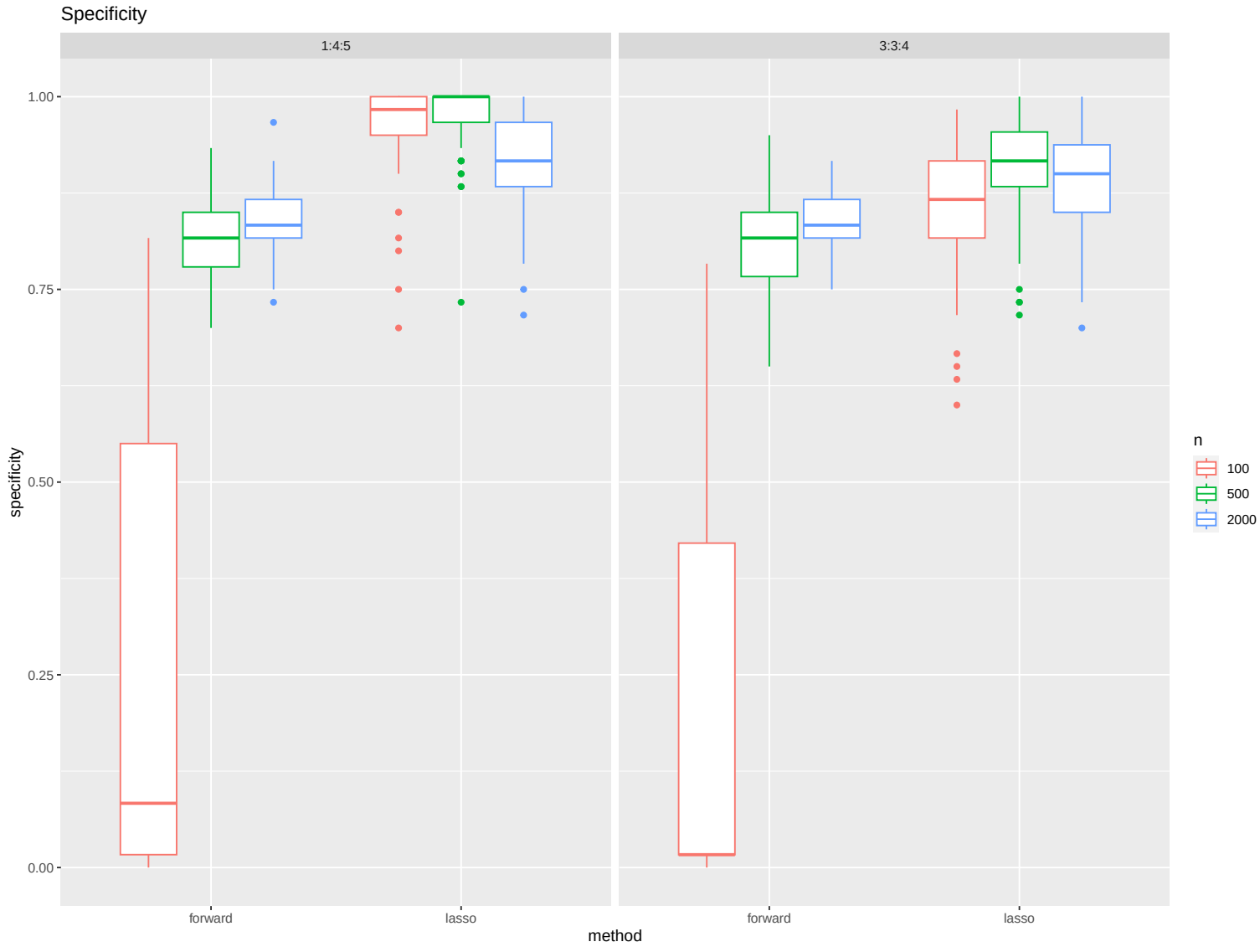


Figure 4: Specificity Performance

Signal Identification Performance - Specificity Exploration

- In high dimensional scenario, the specificity of forward selection is near 0, which means it almost does not identify any null predictor, but Lasso in turn has high specificity
- In high dimensional scenario, forward selection is very assertive and tends to identity all 100 predictors as true, leading to extremely high sensitivity but low specificity
- In high dimensional scenario, Lasso is very conservative and tends to identity most 100 predictors as null,leading to low sensitivity but high specificity
- As n increases, forward selection has higher specificity. And overall, specificity are higher when the ratio of strong predictors are lower

Signal Identification Performance - F1-Score

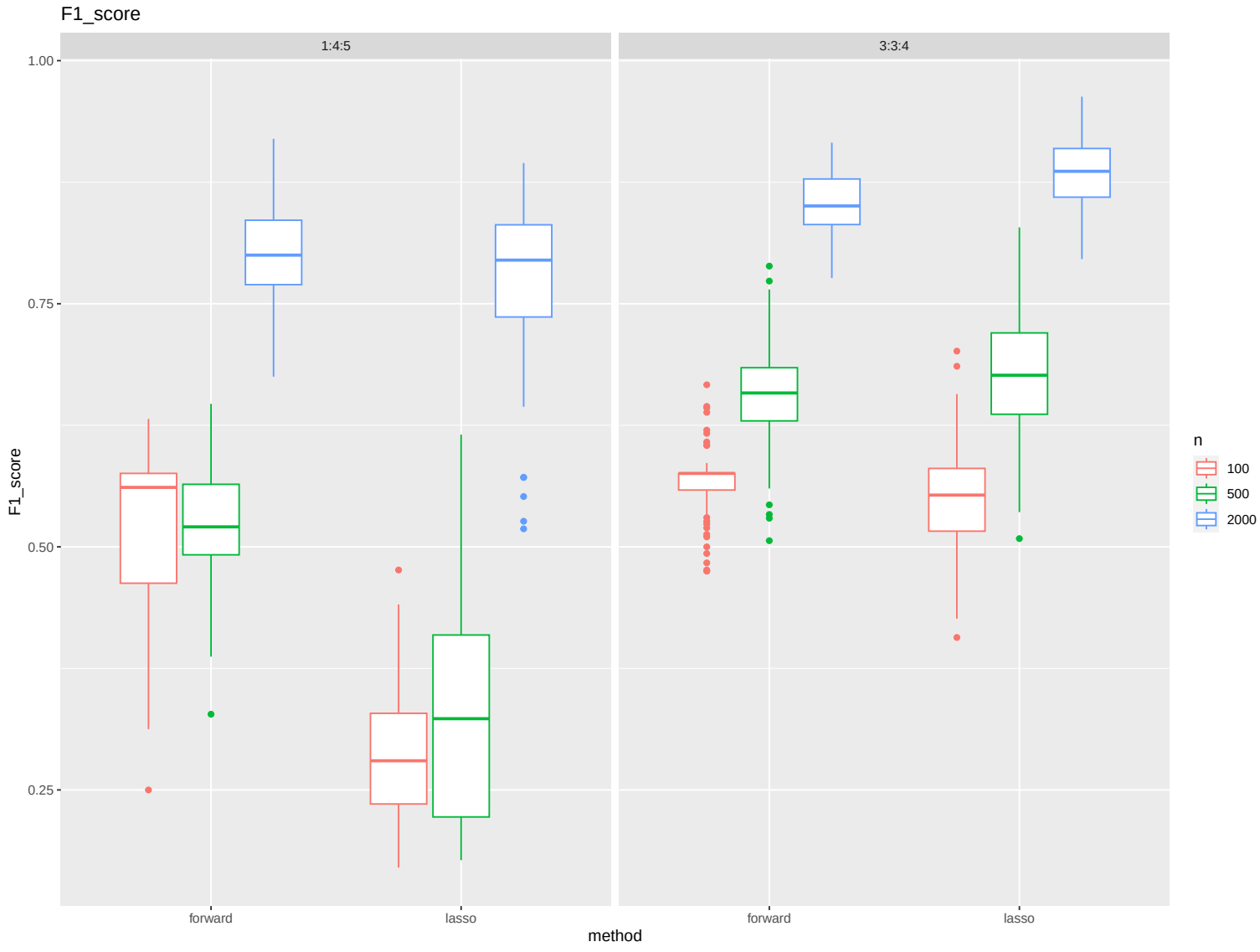


Figure 5: F1-Score Performance

Signal Identification Performance - F1-Score

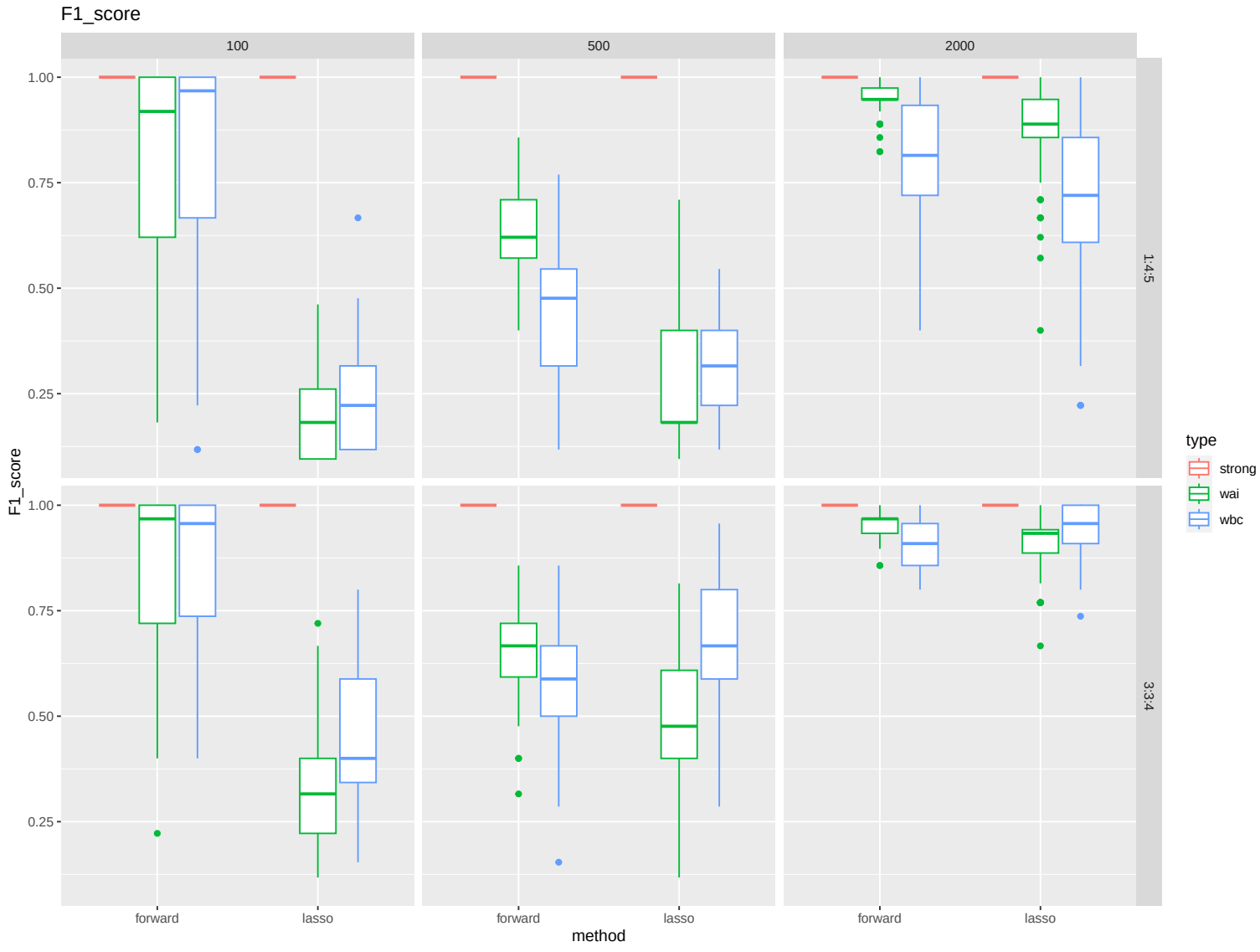


Figure 6: F1-Score of True Signals

Signal Identification Performance - F1-score Exploration

- Lasso has lower F1-Score when n is not large. When n=2000, both models have similarly high F1-score
- F1-score will increase significantly for both models, when the ratio of strong predictors are larger. And F1-score is also higher when the ratio of strong predictors are larger
- Strong predictors have F1-Score=1 for each scenario, and weak predictors have higher F1-score when n increases for Lasso

Signal Identification Performance - Accuracy

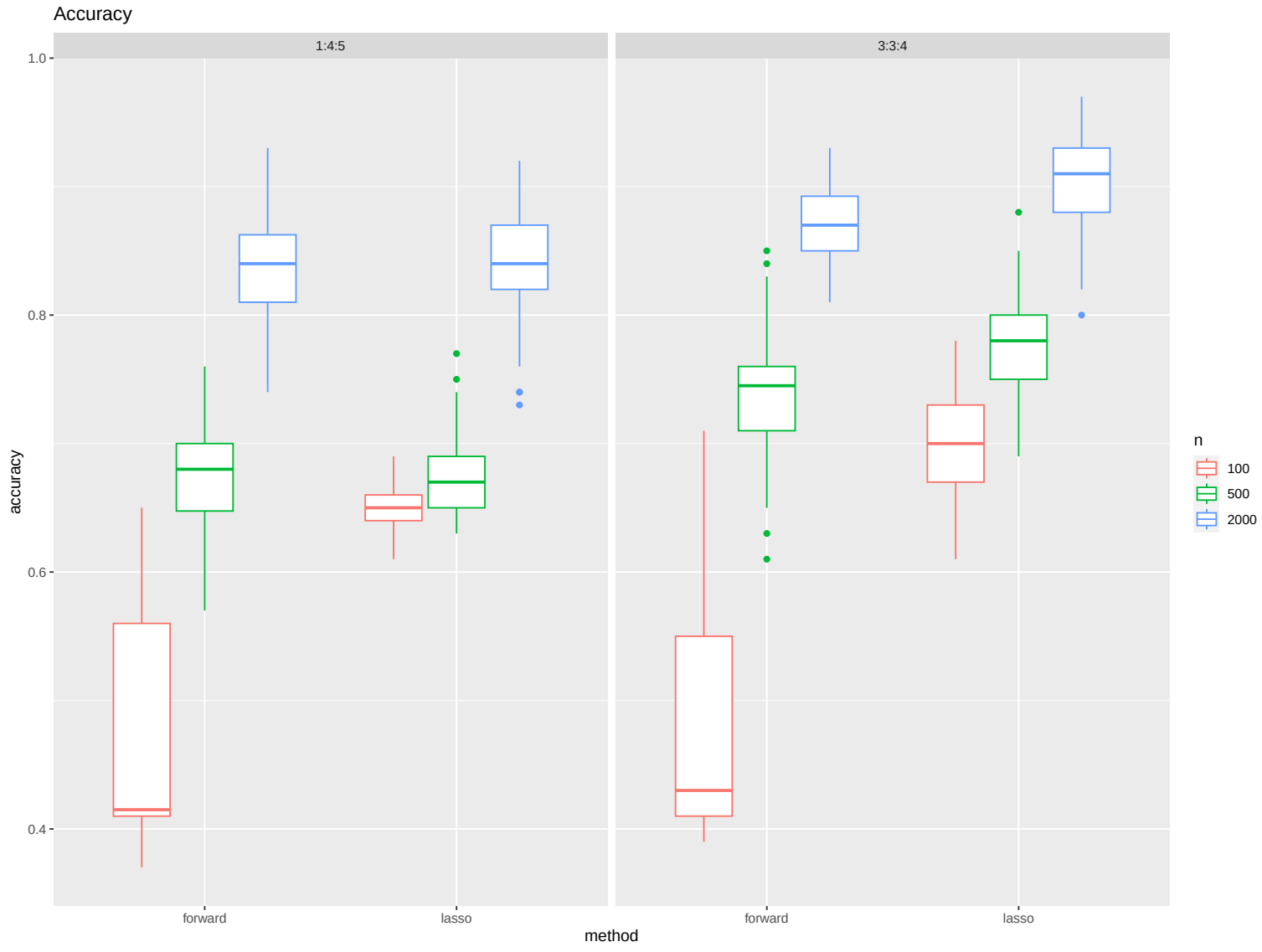


Figure 7: Accuracy Performance

Signal Identification Performance - Accuracy

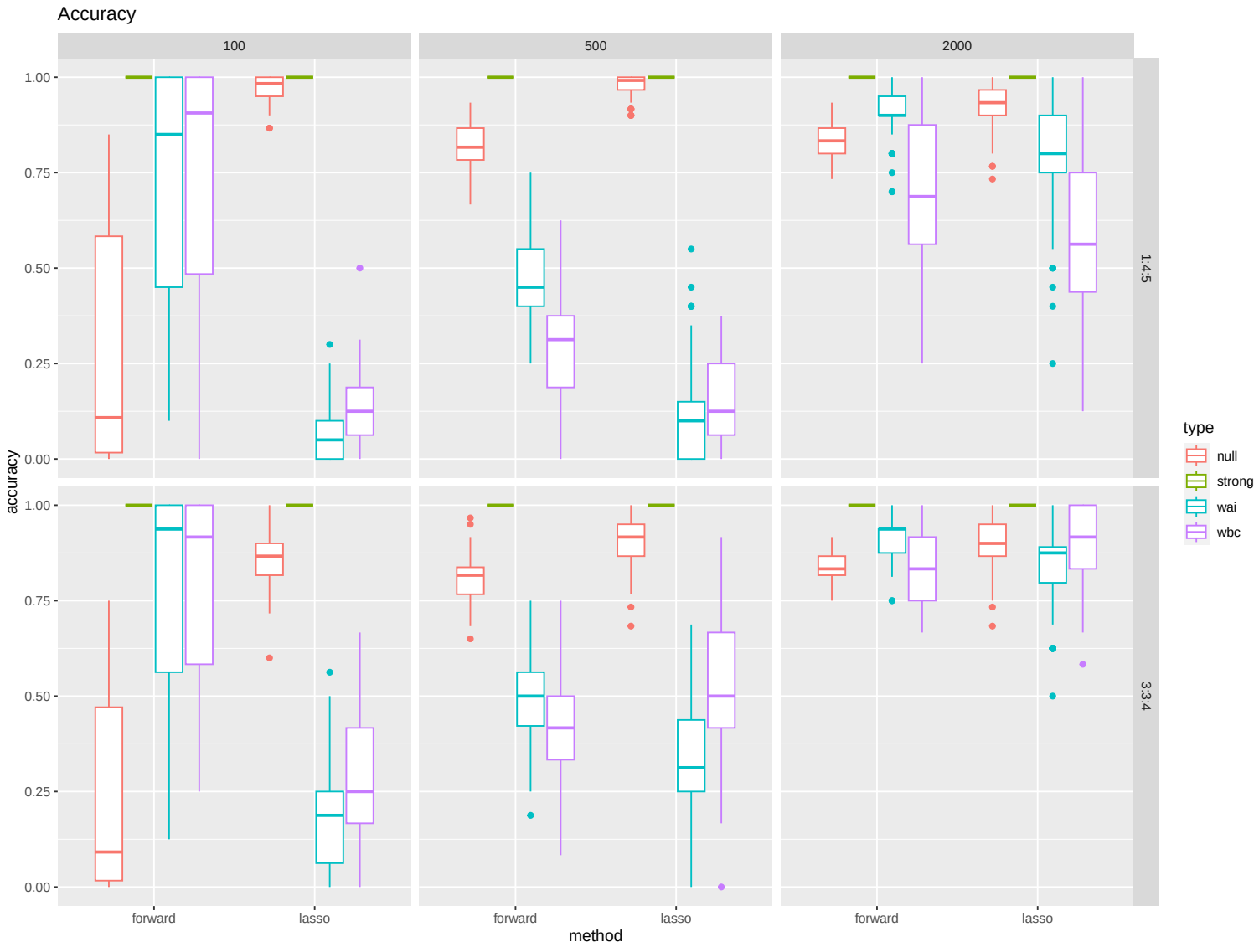
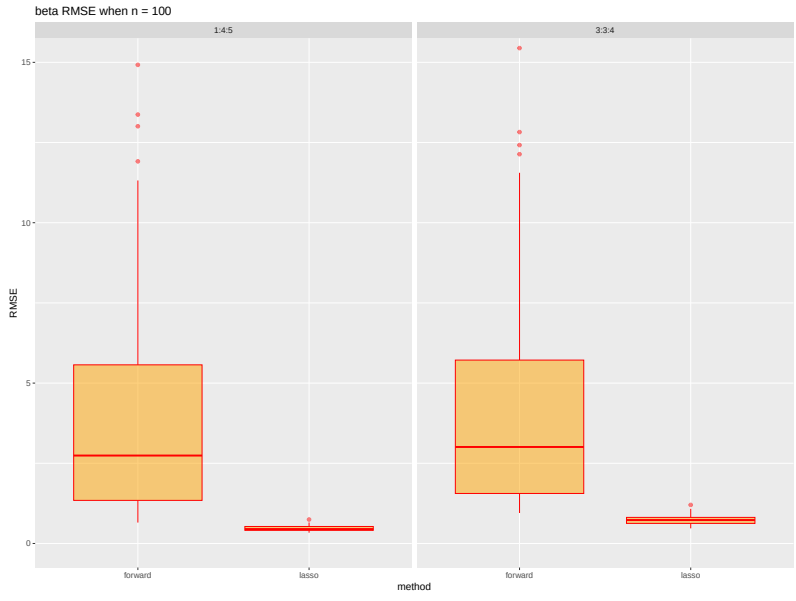


Figure 8: Accuracy of Different Signals

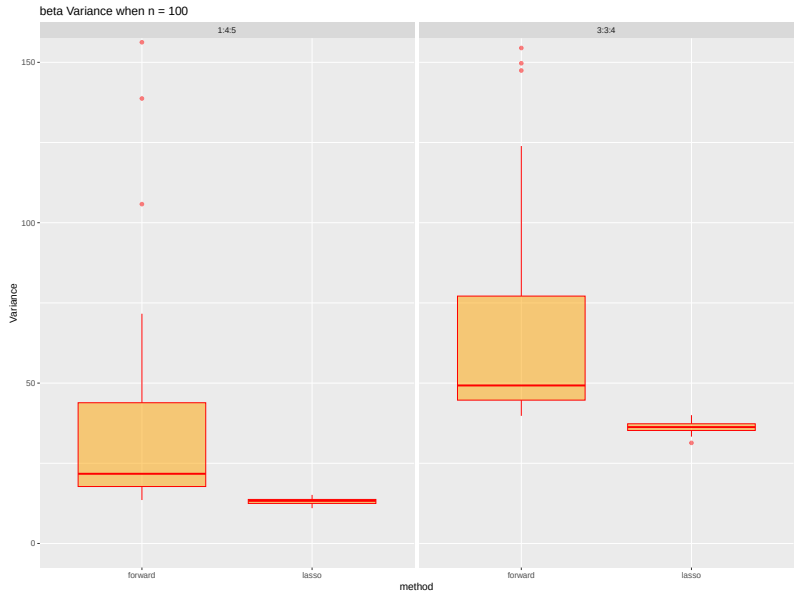
Signal Identification Performance - Accuracy Exploration

- Accuracy is low in high dimensional scenario, especially forward selection
- Accuracy increases for both models when n increase
- Accuracy is higher when the ratio of strong predictors is higher, and Lasso has overall higher accuracy than forward
- In high dimensional, forward has higher accuracy for weak predictors than Lasso

Parameter Estimation Performance: n=100



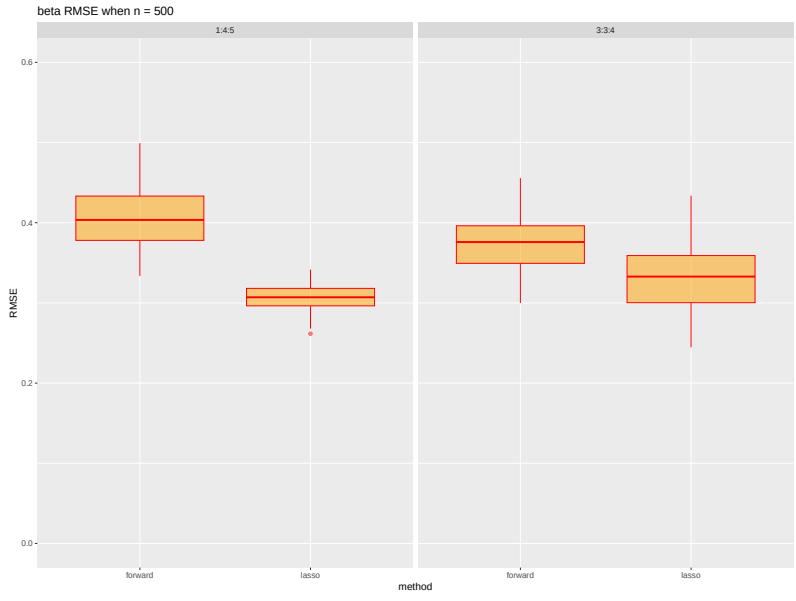
(a) Beta RMSE when n=100



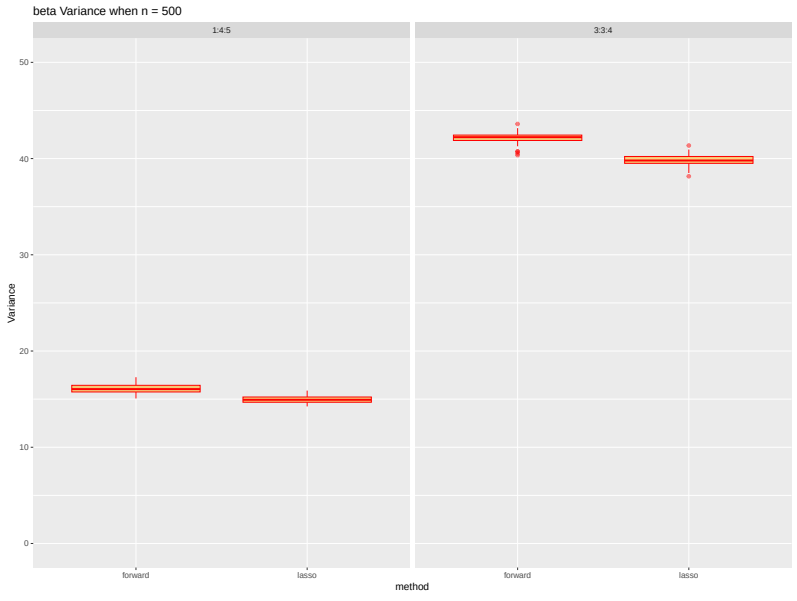
(b) Beta Variance when n=100

Figure 9: Distribution of the datasets

Parameter Estimation Performance: n=500



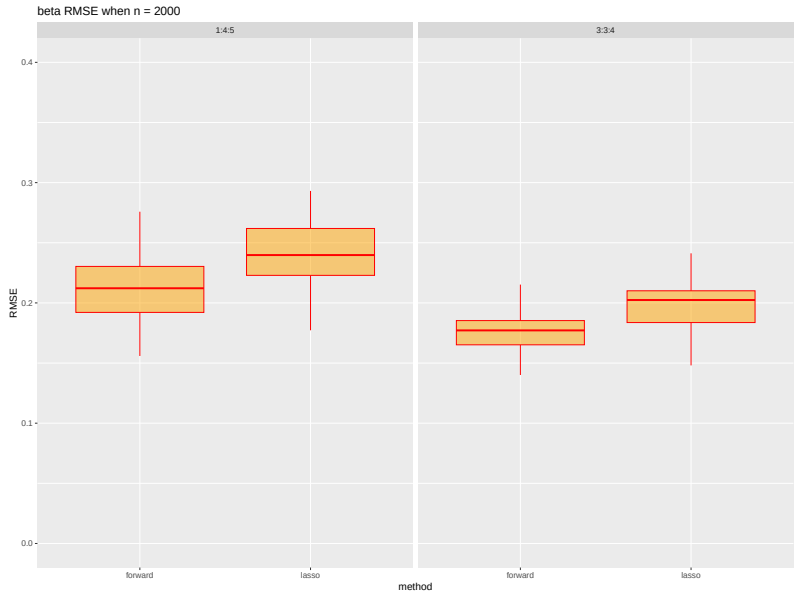
(a) Beta RMSE when n=500



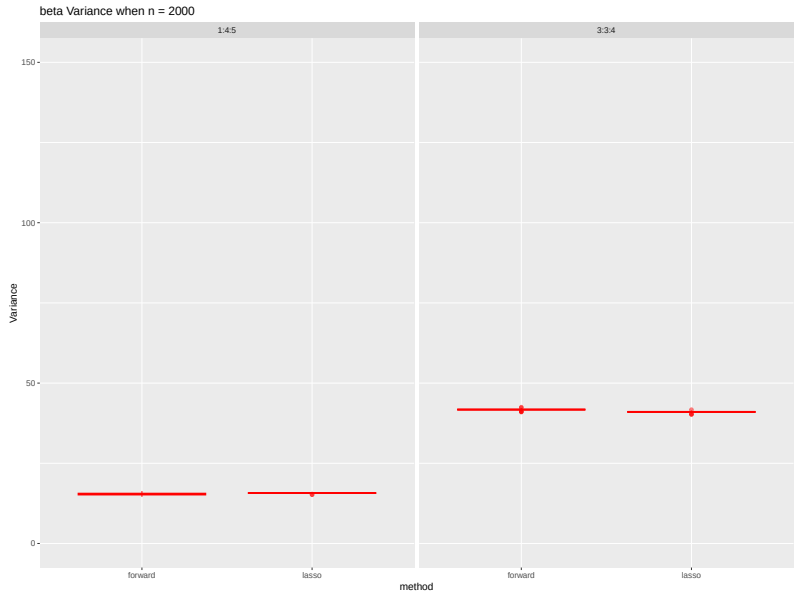
(b) Beta Variance when n=500

Figure 10: Distribution of the datasets

Parameter Estimation Performance: n=2000



(a) Beta RMSE when n=2000



(b) Beta Variance when n=2000

Figure 11: Distribution of the datasets

Parameter Estimation Performance - Exploration

- In high-dimensional scenario, Lasso performs much better than forward
- Lasso performs better than forward when n=500, but worse than forward when n=2000

Predictors Identification Conclusions

High Dimensional Scenario(n=100)

- Forward selection tend to be assertive and better at identifying weak signals, with extremely high sensitivity, low specificity and lots of selected predictors
- Lasso tend to be conservative and better at identifying null signals, with extremely high specificity, low sensitivity and few selected predictors
- Both identify strong predictors perfectly

Predictors Identification Conclusions

High Dimensional Scenario(n=100)

- Forward selection performs better than Lasso on F1-score but worse on overall accuracy
- Lasso performs better on parameter estimation than forward selection
- Both models seem to be too radical in predictor identification

Predictors Identification Conclusions

Normal Scenario(n=500, 2000)

- Forward selection tends to select more predictors than Lasso, but both get closer to actual number as n increases
- Lasso is better at identifying null predictors than forward selection, but poorer at identifying other weak predictors (under-screening). Both models nicely identify strong predictors.
- Lasso performs better on parameter estimation than forward when n is not large, and conversely as n increases

Predictors Identification Conclusions

Normal Scenario(n=500, 2000)

- When there are more strong predictors, weak-but-correlated predictors become easier to be identified, especially for Lasso. However, parameter estimation also becomes more unstable
- Identification differences of all metrics are narrowed down with n increasing

Missing Weak Predictors Analysis - Introduction

How missing "weak" predictors impacts the parameter estimations

Definition: missing weak predictors = true weak predictors but estimated as null

How to value parameter estimations: RMSE

Missing Weak Predictors Analysis - Methods

How to value parameter estimations: RMSE

Most missing: simulations that have the least non-null estimations

Least missing: simulations that have the most non-null estimations

Middle: in between

Missing Weak Predictors Analysis - Result: n=100

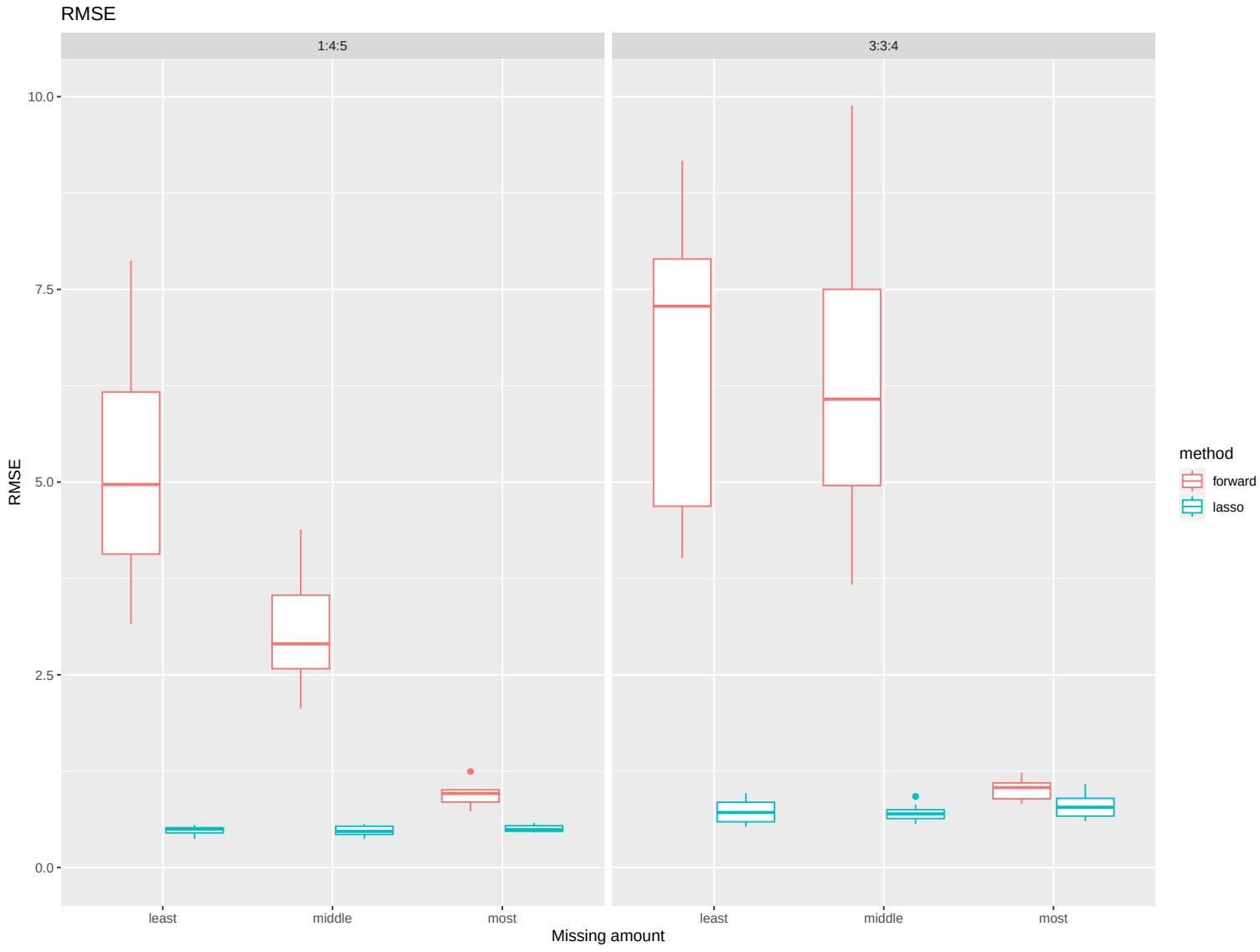


Figure 12: RMSE when n=100

Missing Weak Predictors Analysis - Result: n=500

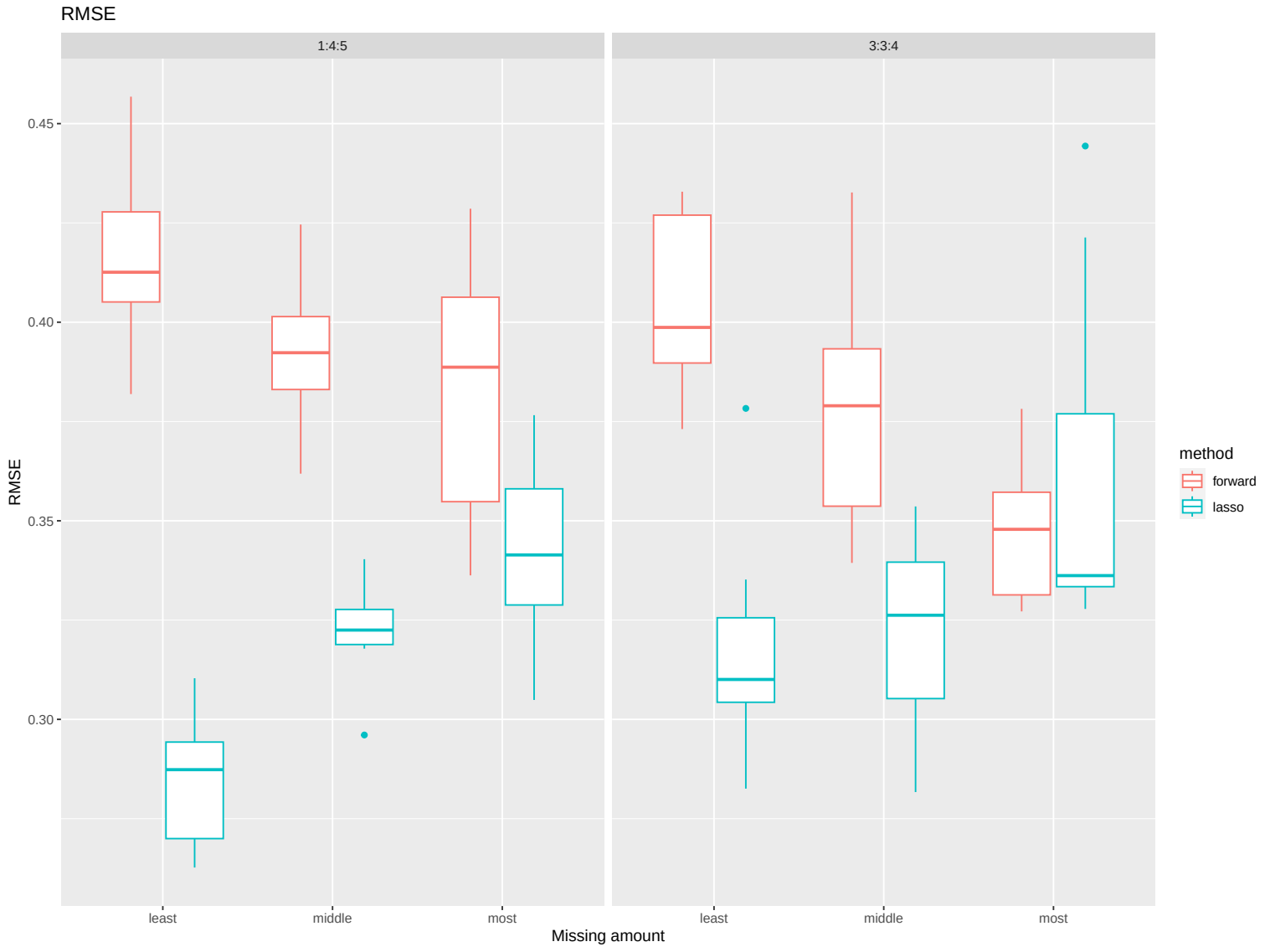


Figure 13: RMSE when n=500

Missing Weak Predictors Analysis - Result: n=2000

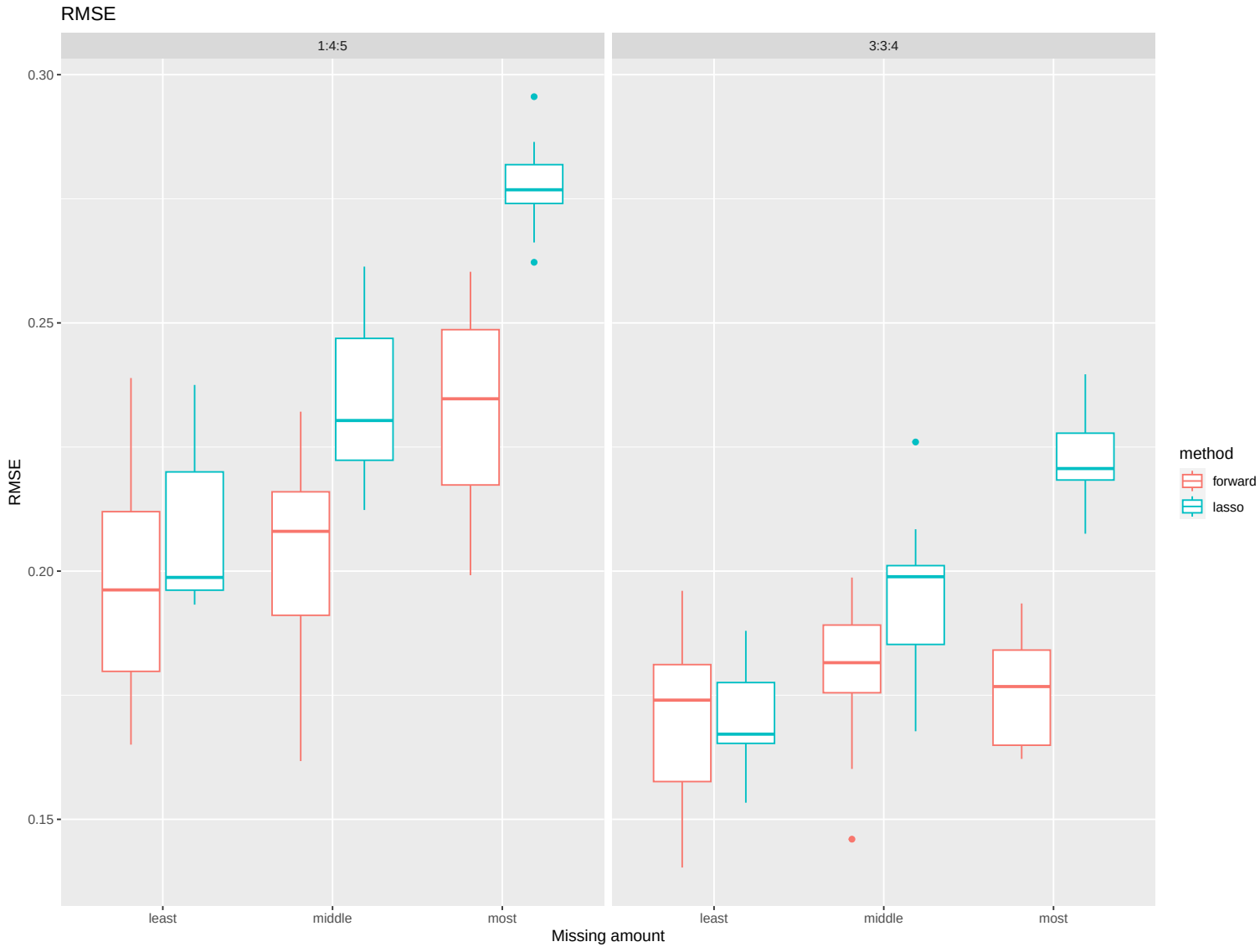


Figure 14: RMSE when n=2000

Missing Weak Predictors Analysis - Discussion

- No apparent patterns between different ratios
- In high-dimensional scenarios, Lasso performs much better than forward selection according to RMSE, no matter how much missing.
- When n is large enough, RMSE of both methods become small.
- When $n = 500$, Lasso is slightly better than forward selection, however, when $n = 2000$, just the reverse.
- In Lasso, RMSE seems to increase if the missing amount increases, but in forward selection, RMSE decreases when missing amount increases.

Limitations

- Correlation matrix Pattern
- Number of observations n and number of parameters p
- Ratio of Strong, Wbc, and Wai predictors

Future Work

- Larger number of initial samples, narrower coverage window
- Increased sample size, changes in bootstrap performance?
- Changes in treatment propensity model
- Non-normal distributions of covariates