Stepwise Regression with Negatively Correlated Covariates

Riley Ashton

Contents

1 Introduction

David Hamilton's 1987 (Hamilton 1987) paper showed that correlated variables are not always redundant and that $R^2 > r_{yx_1}^2 + r_{yx_2}^2$. In Section 3 of Hamilton's paper, an extreme example was shown when two variables were negatively correlated and neither did a good job in explaining the response on their own, but together they explained nearly all the variance ($R^2 \approx 1$).

Forward selection is a greedy algorithm that only examines one variable at a time. In cases of two highly negatively correlated variables, it is possible that neither variable would be added, since it is possible that neither variable would be significant on their own.

Algorithms that consider adding highly negatively correlated variables together, and additionally any variables correlated with those are discussed and tested.

1.1 Notation used

- p: number of covariates/predictors
- q: number of nonzero covariates/predictors (i.e. covariates in the model)
- n: number of observations
- AIC: Akaike information criterion
- BIC: Bayes information criterion
- Step: name used for traditional stepwise regression in this report
- $\mathcal{O}$ refers to Big O notation. Formally $f(n) = \mathcal{O}(g(n))$ is defined as

$$\exists k > 0\,, \exists n_0\,, \forall n > n_0\,, |f(n)| \leq k \cdot g(n)$$

2 Algorithm

2.1 Step2

2.1.1 Algorithm Idea

The first algorithm is called Step2. The purpose of Step2 improve upon the traditional stepwise algorithm (in this report called Step), by examining the correlation between covariates.

In the case of highly negatively correlated covariate pairs, Step2 will consider the traditional choices of adding a single variable, but also the option of adding both negatively correlated covariates to the model (refered to as a block). What constitutes a highly negatively block is determined by the parameter `cor_cutoff`. So a cutoff of -0.5 would mean all covariate pairs with a correlation of less than -0.5 (e.g. -0.7) would be additionally considered together. They would also be considered as singles, as is every other variable not yet included in the model. In this report a `cor_cutoff` of -0.5 is used. This choice of -0.5 for `cor_cutoff` was arbitrary.

While both Step and Step2 are greedy algorithms, Step2 looks two steps ahead at highly negatively correlated covariate pairs.

AIC or BIC is used as the metric for deciding which of the possible models is the best at each step. This is controlled by the parameter `k`. `k` $= 2$ means AIC is used and `k` $= log_e(n)$ means BIC by Step2. In this report $k = \log_e(n)$ or BIC is used.

2.1.2 Pseudocode

Note this pseudocode uses the R notion of formulas, which are manipulated similar to strings. In other languages a list or set of data matrix indices may be stored and appended to and removed from.

```
let pairs = covariate pairs (x,y) such that cor(x,y) > cor_cutoff,
let current_info_crit = information criterion of starting formula
let current_formula = starting formula
let next_formulas = empty list
let past_formulas = empty lookup container
loop
    if(direction is "forward" or "both",
        and the number of observations > the number of covariates in current_formula)

        append (current_formula + y) to next_formulas where y is
        a covariate not yet in current_formula

        append (current_formula + x + y) to next_formulas where (x,y) is in
            pairs and neither x nor y already appear in current_formula
    end if

    if(direction is "backwards" or "both"
        and the number of observations > the number of covariates in current_formula)

        remove y from current_formula and append it to next_formulas
          where y is a covariate already in current_formula

        append (current_formula + x + y) to next_formulas where (x,y) is in
            pairs and both x and y already appear in current_formula
    end if

    if(length of next_formulas is 0)
        return the model corresponding to the current_formula
    end if

    let X = fit a glm to the data for each formula in next_formulas
    let min_info_crit = the minimum information criterion for the models in X
    let min_formula = the formula corresponding to the model with the min_info_crit

    if(min_formula in past_formulas)
        a cycle is present, return an error
    endif

    else if(min_info_crit <= current_info_crit)
        add current_formula to past_formulas
        current_info_crit = min_info_crit
        current_formula = min_formula
      next_formulas = empty list
    end elseif

    else if(min_info_crit > current_info_crit)
        return the model corresponding to the current_formula
    endif
end loop
```

2.1.3 Time and Space Complexity vs Step

The time complexity of the algorithm depends heavily on the choice of `cor_cutoff`. The tradeoff is between model that is potentially more accurate and a longer run time. The returns in model accuracy, to `cor_cutoff` closer to zero, decrease rapidly. Step rarely has a problem with only slightly negative correlated variables, so there is little for Step2 to improve on in those situations.

The time complexity of Step2 is generally within a constant of Step. At worst it will be `p` times greater, where `p` is the number of parameters. This is since Step chooses from at most `p` models to fit at each stage, where Step2 choose from at most `p` choose 2 models.

The space complexity is largely unchanged from the traditional algorithm, since only the AIC or BIC or each model is stored.

2.2 Step3

2.2.1 Algorithm Idea

Step3 does everything that Step2 does, but it also considers recursing on the pairwise negatively correlated covariates, i.e. the blocks of Step2, and considering anything highly correlated with them and then anything highly correlated with those, etc until it reaches a block size of `max_block_size`. Additionally, Step3 with a `max_block_size` of 2 is equivalent to Step2. The depth used in this report is set at 3, so the algorithm will, at most, consider including three covariates at a time. This choice was arbitrary.

What is classified as a highly correlated variable for the purposes of recursion are set by the two parameters `recursive_cor_positive_cutoff` and `recursive_cor_negative_cutoff`. In this report 0.5 and -0.5 was used for `recursive_cor_positive_cutoff` and `recursive_cor_negative_cutoff`, respectively. The choice of 0.5 and -0.5 was arbitrary.

- Let $\wedge$ represent logical AND
- Let $\vee$ represent logical OR
- Let CC be a short-form for `cor_cutoff`
- Let RCN be a short-form for `recursive_cor_negative_cutoff`
- Let RCP be a short-form for `recursive_cor_positive_cutoff`
- Let $r(x_1, x_2)$ denote the sample correlation between x_1 and x_2
- Let S_1 denote the set of covariates
- Then S_2 denoting the set of highly negatively correlated pairs (x_1, x_2) is defined as

$$S_2 = \{(x_1, x_2) \,|\, x_1 \in S_1 \wedge x_2 \in S_1 \wedge x_1 \neq x_2 \wedge r(x_1, x_2) < \text{CC}\}$$

- Then S_3 denoting the set of triples (x_1, x_2, x_3) is defined as

$$S_3 = \{(x_1, x_2, x_3) \,|\, (x_1, x_2) \in S_2 \wedge x_3 \in S_1 \wedge x_3 \notin (x_1, x_2) \wedge (\text{block } x_1\, x_2\, x_3)\}$$

  where

$$\text{block } x\, y\, z = (r(x, z) < \text{RCN}) \vee (r(x, z) > \text{RCP}) \vee (r(y, z) < \text{RCN}) \vee (r(y, z) > \text{RCP})$$

- Likewise this pattern continues
- Then, in the genral case, S_n denoting the set of length n, $(x_1, x_2, \ldots, x_n)$ is defined as

  $S_n = \{(x_1, \ldots, x_n) \mid (x_1, \ldots, x_{n-1}) \in S_{n-1} \wedge x_n \in S_1 \wedge x_n \notin (x_1, \ldots, x_{n-1}) \wedge (\text{block2 } (x_1, \ldots, x_{n-1})\, x_n)\}$

  where

$$\text{block2 } D\, c = \exists a, b \in D \,|\, a \neq b \wedge \text{block } a\, b\, c$$

2.2.2 Pseudocode Changes from Step2

The core of the algorithm is the same as Step2, except for computing pairs and adding or removing pairs to current formula. This is because pairs are now generalized to lists of size ≥ 2. In the case of adding or removing pairs to current formula, Step3 appends the objects in lists to the current formula if none are currently in the current formula and it removes them if all are currently in the current formula.

2.2.3 Time and Space Complexity vs Step

With a terrible choice of cutoff parameters (e.g. correlation < 1 or > -1) and unlimited recursion depth, Step3 will preform all subsets regression. All subsets regression has an exponential time complexity in the number of covariates.

The worst case space and time complexity for a `max_block_size` of mbs is $\mathcal{O}\left(p^{(\text{mbs}-1)}\right)$ times that of Step. With proper choices of cutoffs the time and space complexity is expected to be within a constant of Step.

Note that the expected proportional increase in running time could be computed before running any simulations for any given data set and algorithm parameters. A possible future improvement could involve the algorithm guaranteeing similar performance to Step by tuning the algorithm parameters for a given data set.

3 Simulations

3.1 Linear model

- $\mathbf{Y} = \beta_0 + \beta_1 \mathbf{X}_1 + \beta_2 \mathbf{X}_2 + \cdots + \epsilon$
- $\mathbf{X}_i$ are correlated, centred random normal variables
- Intercept ($\beta_0 = 9$)
- $\epsilon \sim \mathcal{N}(\mu = 0, \sigma = 1)$
- 1000 Simulations

3.2 Generating Correlated Centred Random Normal Variables

The variables are generated according the the formula

$$\mathbf{X} = \mathbf{CZ}$$

This generates a vector of correlated random normal variables $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu} = 0, \boldsymbol{\Sigma})$

where $\mathbf{C}$ is a $p \times p$ matrix that can be solved from the given covariate matrix, $\boldsymbol{\Sigma}$, such that

$$\boldsymbol{\Sigma} = \mathbf{CC^T} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{1p} & \sigma_{2p} & \cdots & \sigma_p^2 \end{bmatrix}$$

$\mathbf{C}$ is found using Cholesky decomposition

Where $\mathbf{Z}$ is a vector of random normal variables

$$\mathbf{Z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{bmatrix}$$

Where $Z_i \sim \mathcal{N}(\mu = 0, \sigma = 1)$

3.3 Model 1 : Two negatively correlated

This model is designed to show the advantage of Step2 over Step

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & -0.8 & 0 & 0 & 0 & 0 \\ -0.8 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{\beta} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$n = 100$

3.4 Model 2: Three Negatively Correlated

This model is designed to show the advantage of Step3 over Step2 and Step

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & -0.8 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.8 & 1 & -0.75 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & -0.75 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \boldsymbol{\beta} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$n = 100$

3.5 Model 3 : Big p

This model is designed to show the advantages of Step2 and Step3 over Step when $p >> n$. Note the covariates V6 through V100 are independent and do not contribute to the response ($\beta_i = 0$ for $i \in [6..100]$).

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & -0.8 & 0.25 & 0 & 0 & 0 & \cdots & 0 \\ -0.8 & 1 & -0.5 & 0 & 0 & 0 & \cdots & 0 \\ 0.25 & -0.5 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & -0.75 & 0 & \cdots & 0 \\ 0 & 0 & 0 & -0.75 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \qquad \boldsymbol{\beta} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$n = 50$

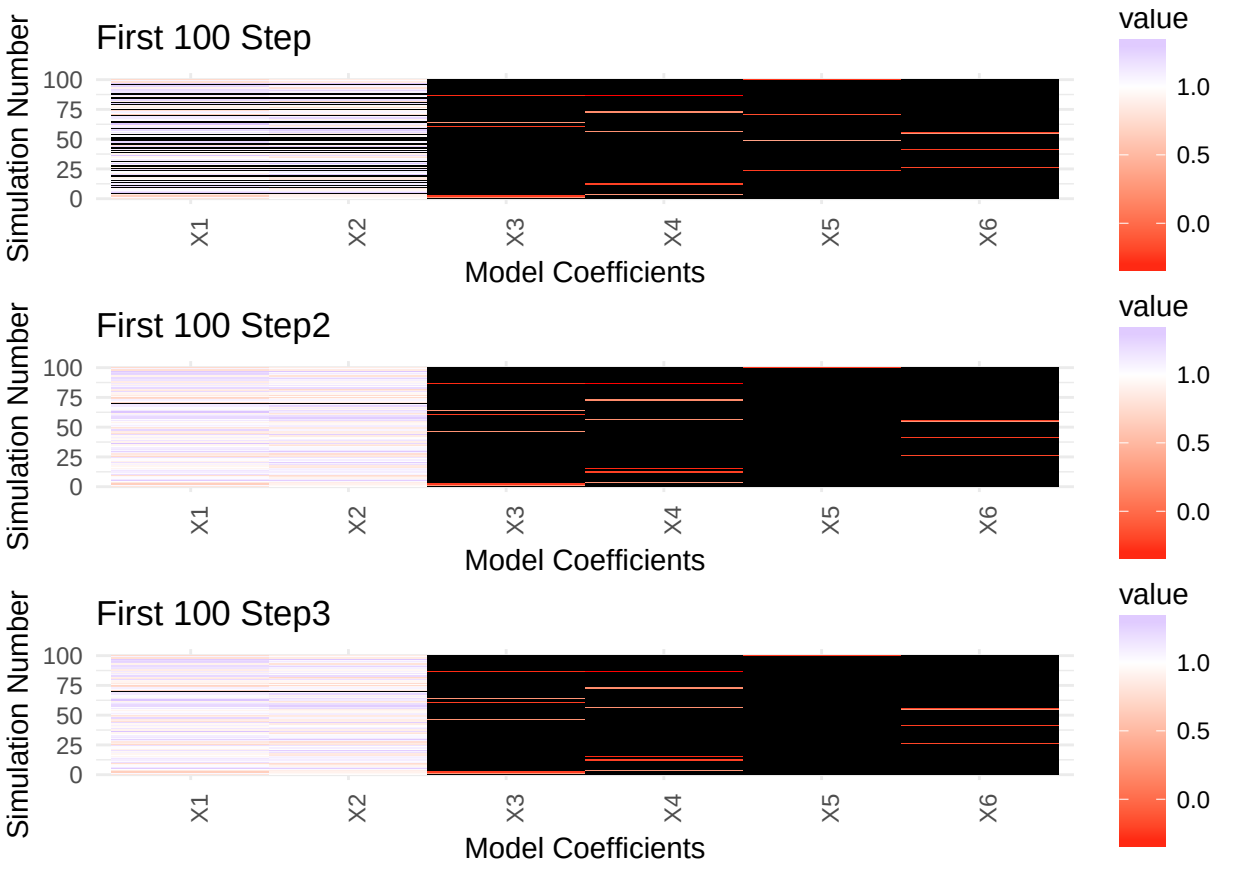
4 Results

4.1 Model 1 : Two Negatively Correlated

In this case Step3 did not end up selecting any more covariates to add to its block inclusion via its recursion, so it was identical to Step2. They were identical since only two covariates were high negatively correlated (X_1 and X_2) and neither X_1 nor X_2 were highly correlated with any other covariates. This meant that Step2 and Step3 considered including X_1 and X_2 in the same step, but did not consider adding any other variables in the same step.

Both Step2 and Step3 outperformed Step, since Step often ($\approx 30\%$) selected no covariates to include in the model (i.e. the intercept only model). This is due to the highly negative sample correlation ($r \approx -0.8$) between X_1 and X_2.

Since Step did not include X_1 and X_2 about 40% of the time, while Step2 and Step3 included them nearly all the time, the variance of the coefficient estimates for X_1 and X_2 were much higher in the Step simulations than in Step2 or Step3.



Here the heat map for the first 100 simulations and the values of the model coefficients are shown. Note that NA values are shown as black, meaning that coefficient was not selected in that simulation. Step often misses both X1 and X2, while Step2 and Step3 nearly always include both of these variables.

Table 1: % Simulations Including Covariates

	Step	Step2	Step3
X1	0.659	0.998	0.998
X2	0.659	0.998	0.998
X3	0.035	0.037	0.037
X4	0.040	0.039	0.039
X5	0.042	0.044	0.044
X6	0.035	0.036	0.036

Table 1 shows the proportion of times that each algorithm selected a given covariate. Step did not select X1 or X2 over a third of the time. Step2 and Step3 nearly always select X1 and X2.

In the appendix is the proportion of times that each algorithm selected the correct model.

Table 2: Fitted Coefficients Bias

	Step	Step2	Step3
(Intercept)	-0.007	-0.004	-0.004
X1	-0.319	0.002	0.002
X2	-0.321	0.001	0.001
X3	0.001	0.001	0.001
X4	-0.002	-0.003	-0.003
X5	-0.002	-0.002	-0.002
X6	0.002	0.002	0.002

Table 3: Fitted Coefficients Variance

	Step	Step2	Step3
(Intercept)	0.012	0.011	0.011
X1	0.258	0.031	0.031
X2	0.256	0.030	0.030
X3	0.002	0.002	0.002
X4	0.003	0.003	0.003
X5	0.003	0.003	0.003
X6	0.002	0.002	0.002

The bias of the fitted coefficients $\text{bias}_{\beta_j} = \left(\frac{1}{n}\sum_{i=1}^{n}\beta_{ij}\right) - \left(\frac{1}{n}\sum_{i=1}^{n}\hat{\beta}_{ij}\right)$, shown in Table 2, is greatly reduced for X1 and X2 under Step2 and Step3 compared to Step.

The variance of the fitted values for X1 and X2, shown in Table 3 is also reduced since Step included X1 and X2 about 60% of the time

---

Table 4 shows the 5 most commonly occuring orders that each variable was included, by each stepwise algorithm. The full tables are available in the appendix. Note that the character | delineates the steps of the selection algorithm and || denotes the termination of the selection algorithm.

For example, || denotes the algorithm terminated without selecting any variables, while X1X2 | X3 || means the algorithm selected both X1 and X2 in the first step, X3 in the second step and then terminated.

Table 4: Inclusion Order for Step, Step2, Step3 Respectively

Order Step	# Step	Order Step2	# Step2	Order Step3	# Step3
||	304	X1X2 ||	852	X1X2 ||	852
X1 | X2 ||	289	X1X2 | X5 ||	39	X1X2 | X5 ||	39
X2 | X1 ||	265	X1X2 | X3 ||	35	X1X2 | X3 ||	35
X2 | X1 | X4 ||	14	X1X2 | X4 ||	31	X1X2 | X4 ||	31
X1 | X2 | X5 ||	12	X1X2 | X6 ||	31	X1X2 | X6 ||	31

Step2 and Step3 are identical, as explained at the start of the section. Step did not select any variables over a quarter of the time. In nearly all cases, Step2 and Step3 selected X1 and X2 together. It is ability to include highly negatively correlated covariates together that makes Step2 and Step3 ideal in this type of case with high pairwise negative correlations.

$$\begin{bmatrix} X & -0.89/-0.64 & -0.3/0.33 & -0.31/0.28 & -0.29/0.33 & -0.32/0.33 \\ -0.89/-0.64 & X & -0.26/0.3 & -0.35/0.37 & -0.31/0.28 & -0.27/0.3 \\ -0.3/0.33 & -0.26/0.3 & X & -0.31/0.27 & -0.29/0.28 & -0.31/0.33 \\ -0.31/0.28 & -0.35/0.37 & -0.31/0.27 & X & -0.29/0.3 & -0.27/0.33 \\ -0.29/0.33 & -0.31/0.28 & -0.29/0.28 & -0.29/0.3 & X & -0.32/0.32 \\ -0.32/0.33 & -0.27/0.3 & -0.31/0.33 & -0.27/0.33 & -0.32/0.32 & X \end{bmatrix}$$

The above matrix shows the sample pairwise correlation matrix of all the simulations. looking at the first off diagonal entry of $-0.89/-0.67$ means that the minimum sample pairwise correlation between $X1$ and $X2$ was -0.89, while the maximum was -0.67. This shows that, for all simulations, Step2 and Step3 would have the option of including $X1$ and $X2$ together, but would not have the option to do so for any other variables.

Table 5: Test SSE

	Min	Max	Mean	Median
Step	66.767	205.301	117.903	114.020
Step2	66.098	169.813	104.488	103.787
Step3	66.098	169.813	104.488	103.787

Table 6: Training SSE

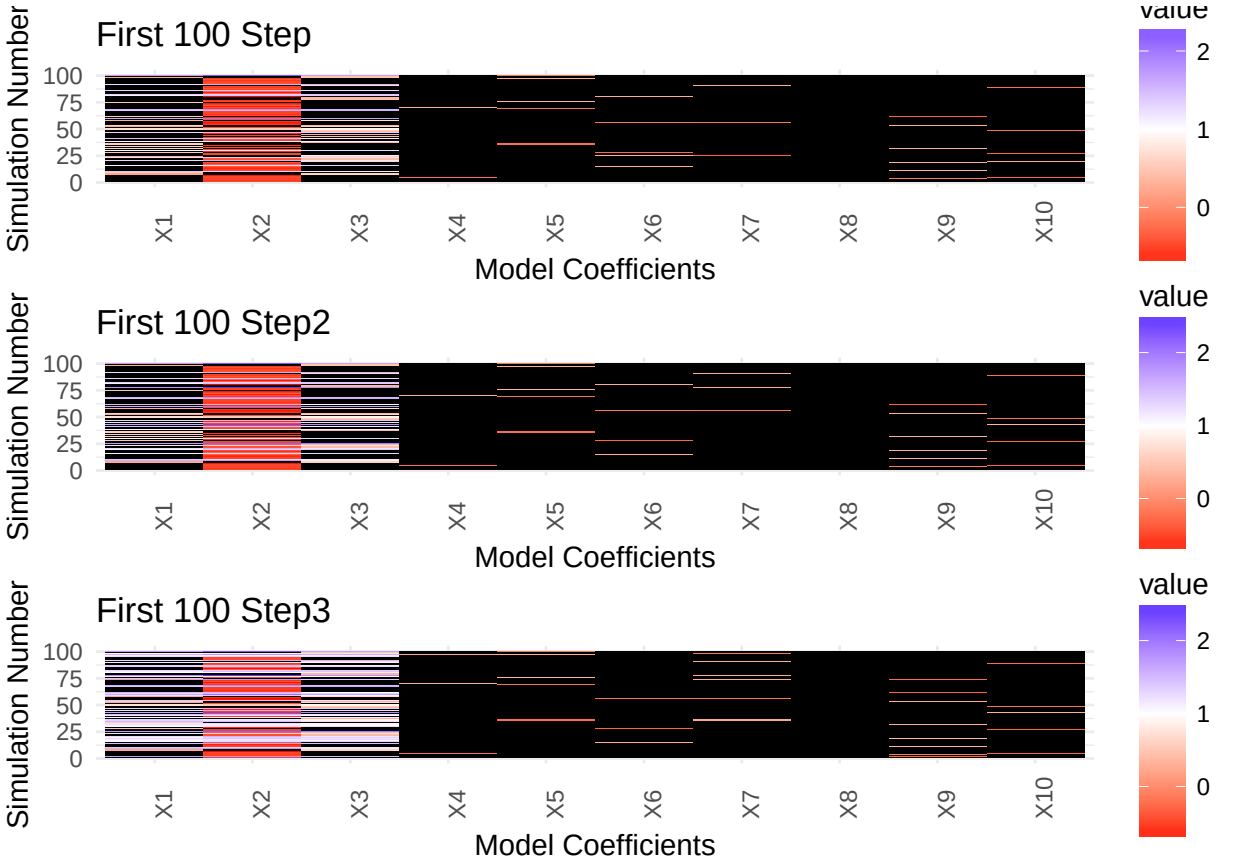
	Min	Max	Mean	Median
Step	62.028	197.897	108.396	103.517
Step2	59.685	140.377	96.765	96.000
Step3	59.685	140.377	96.765	96.000

Table 5 and Table 6 show the test and training SSE from each algorithm. Step2 and Step3 are, again, identical. Step2 and Step3 reduced the training SSE and test SSE compared to Step by 10.7% and 11.4%, respectively.

Boxplots of test and training SSE are located in the appendix

4.2 Model 2 : Three Negatively Correlated

In addition to examing singles, like Step, Step2 and Step3 would examine pairs of X1, X2 and X3. Finally Step3 would additionally consider adding X1, X2 and X3 all together. This results in Step2 having better results than Step and Step3 having better results than Step2.



Examining the heat map of fitted coefficients, all algorithms have difficulty selecting all the correct covariates. This is an especially difficult correlation structure. However Step3 does a considerably better job at selecting all of X1, X2 and X3 compared to Step and Step2.

Table 7: % Simulations Including Covariates

	Step	Step2	Step3
X1	0.226	0.273	0.566
X2	0.778	0.853	0.870
X3	0.317	0.341	0.631
X4	0.034	0.034	0.039
X5	0.032	0.031	0.031
X6	0.036	0.035	0.035
X7	0.038	0.038	0.041
X8	0.025	0.026	0.029
X9	0.033	0.033	0.035
X10	0.040	0.042	0.044

Table 7 shows the proportion of times that each algorithm selected a given covariate. Step had very poor performance in selecting X1 and X3, about a fifth and a third of the time respectively. Step2 offered slght improvements with these two variables, but Step3 showed a marked improvement selecting both X1 and X3 compared to Step or Step2.

It is also worth noting that Step2 and Step3 performed better in selecting X2 over Step. This suggests that X2 was not always significant by itself.

In the appendix is the proportion of times that each algorithm selected the correct model.

Table 8: Fitted Coefficients Bias

	Step	Step2	Step3
(Intercept)	0.000	0.000	-0.001
X1	-0.789	-0.705	-0.326
X2	-1.164	-1.049	-0.479
X3	-0.731	-0.661	-0.306
X4	0.000	0.000	0.001
X5	0.002	0.002	0.001
X6	0.000	-0.001	-0.001
X7	-0.001	-0.001	-0.001
X8	-0.001	-0.001	-0.001
X9	0.004	0.004	0.003
X10	-0.003	-0.003	-0.004

Table 9: Fitted Coefficients Variance

	Step	Step2	Step3
(Intercept)	0.011	0.010	0.010
X1	0.206	0.284	0.416
X2	0.494	0.703	0.906
X3	0.223	0.294	0.366
X4	0.002	0.002	0.003
X5	0.002	0.002	0.002
X6	0.002	0.002	0.002
X7	0.002	0.002	0.002
X8	0.002	0.002	0.002
X9	0.002	0.002	0.003
X10	0.003	0.003	0.003

Step2 reduces the bias of X1, X2 and X3 (Table 8), but suffers greater fitted coefficient variance compared to Step (Table 9). Step3 further reduces the bias of X1, X2 and X3, but suffers greater fitted coefficient variance compared to Step2. The increase in variance is most likely due to the fact that the proportion that X1 and X3 are selected approaches 50% with Step3 (Table 7). Additionally the proportion that Step3 selected at least the correct covariates (Table 21), or the correct model (Table 20) is far closer to 50% compared to Step. These two tables, Table 20 and Table 21, are in the appendix.

Table 10: Inclusion Order for Step, Step2, Step3 Respectively

Order Step	# Step	Order Step2	# Step2	Order Step3	# Step3
X2 ||	498	X2 ||	498	X1X2X3 ||	382
X3 ||	98	X3 | X1X2 ||	94	X2 ||	271
X3 | X1 | X2 ||	74	X3 ||	63	X3 ||	63
X1 ||	40	X1 | X2X3 ||	32	X1X2X3 | X10 ||	23
X3 | X1 ||	31	X2X3 | X1 ||	30	X1X2X3 | X7 ||	23
X2 | X6 ||	19	X3 | X1 ||	26	X1 ||	22
X2 | X9 ||	19	X1 ||	22	X1X2X3 | X4 ||	19
X2 | X4 ||	18	X2 | X6 ||	19	X3 | X1 ||	18
X1 | X3 | X2 ||	17	X2 | X9 ||	19	X1X2X3 | X6 ||	17
X2 | X10 ||	17	X2 | X4 ||	18	X2X3 ||	13

In Table 10, Step3 shows its recursive ability here by including X1, X2 and X3 together in one single step.

$$\begin{pmatrix}
X & -0.89/-0.61 & -0.05/0.5 & -0.31/0.35 & -0.32/0.33 & -0.31/0.33 & -0.32/0.26 & -0.33/0.29 & -0.32/0.32 & -0.34/0.32 \\
-0.89/-0.61 & X & -0.85/-0.58 & -0.38/0.37 & -0.36/0.34 & -0.3/0.33 & -0.28/0.25 & -0.27/0.33 & -0.35/0.31 & -0.32/0.3 \\
-0.05/0.5 & -0.85/-0.58 & X & -0.34/0.32 & -0.38/0.31 & -0.3/0.32 & -0.34/0.32 & -0.3/0.32 & -0.3/0.28 & -0.36/0.27 \\
-0.31/0.35 & -0.38/0.37 & -0.34/0.32 & X & -0.35/0.29 & -0.26/0.35 & -0.32/0.27 & -0.29/0.35 & -0.31/0.36 & -0.31/0.29 \\
-0.32/0.33 & -0.36/0.34 & -0.38/0.31 & -0.35/0.29 & X & -0.28/0.31 & -0.32/0.32 & -0.29/0.31 & -0.34/0.36 & -0.34/0.31 \\
-0.31/0.33 & -0.3/0.33 & -0.3/0.32 & -0.26/0.35 & -0.28/0.31 & X & -0.31/0.36 & -0.35/0.27 & -0.31/0.28 & -0.3/0.31 \\
-0.32/0.26 & -0.28/0.25 & -0.34/0.32 & -0.32/0.27 & -0.32/0.32 & -0.31/0.36 & X & -0.31/0.27 & -0.28/0.29 & -0.3/0.29 \\
-0.33/0.29 & -0.27/0.33 & -0.3/0.32 & -0.29/0.35 & -0.29/0.31 & -0.35/0.27 & -0.31/0.27 & X & -0.34/0.34 & -0.31/0.27 \\
-0.32/0.32 & -0.35/0.31 & -0.3/0.28 & -0.31/0.36 & -0.34/0.36 & -0.31/0.28 & -0.28/0.29 & -0.34/0.34 & X & -0.31/0.29 \\
-0.34/0.32 & -0.32/0.3 & -0.36/0.27 & -0.31/0.29 & -0.34/0.31 & -0.3/0.31 & -0.3/0.29 & -0.31/0.27 & -0.31/0.29 & X
\end{pmatrix}$$

Examining the sample correlation matrix, shown above, Step3 always had the option to include X1, X2 and X3 together.

Table 11: Test SSE

	Min	Max	Mean	Median
Step	68.55	171.91	113.804	112.274
Step2	68.55	171.91	112.838	111.249
Step3	66.98	171.91	110.119	108.872

Table 12: Training SSE

	Min	Max	Mean	Median
Step	56.453	167.015	102.996	102.008
Step2	56.453	167.015	102.082	101.120
Step3	56.453	143.900	97.779	96.817

As evidenced by Table 11 and Table 12, Step2 and Step had similar performance. Step2 had only slightly better test and training SSE than Step. The similar performance, between Step and Step2, is most likely due to the complex correlation structure of the covariates. Step2 is only designed to handle high levels of correlation between 2 covariates at a time, but the correlation structure here involves 3 covariates (X1, X2 and X3), so Step2's performance is less than optimal. Step3, being designed to handle complex correlation structures with its recursive ability, improved upon the performance of Step and Step2.

Step3 reduced the training SSE and test SSE compared to Step by 5.1% and 3.2%, respectively.

Boxplots test and training SSE located in appendix

4.3 Model 3 : Big p

In this simulation $p >> n$ and the curse of dimensionality is present in the results

Table 13: % Simulations that Selected Correct Model

	x
Step	0.000
Step2	0.001
Step3	0.001

There was so much noise that Step2 and Step3 selected the correct model only once and Step never selected it (Table 13).

Table 14: % Simulations Selected at Minimum All Correct Covariates

	x
Step	0.188
Step2	0.817
Step3	0.852

When judging algorithms on including at least the covariates of the correct model, Step3 did the best followed by Step2 and then Step (Table 14).

Table 15: Test SSE

	Min	Max	Mean	Median
Step	66.55	1198.447	276.442	252.603
Step2	39.07	884.075	226.345	210.094
Step3	39.07	884.075	222.081	207.934

Table 16: Training SSE

	Min	Max	Mean	Median
Step	0	115.590	7.046	0
Step2	0	75.324	3.882	0
Step3	0	75.324	3.660	0

Table 15 and 16 show the test and training SSE results. The median value of 0 for training SSE is not surprising, since it is easy to fit a perfect training model in the case of $p >> n$.

Step2 reduced the training SSE and test SSE compared to Step by 44.9% and 18.1%, respectively.

Step3 reduced the training SSE and test SSE compared to Step by 48.1% and 19.7%, respectively.

Boxplots of test and training SSE are located in the appendix

5 Conclusion

Step2 and Step3 offer improvements to the traditional stepwise algorithm in the case of highly negatively correlated covariates, in terms of model fit. They can suffer from longer runtimes, especially when their algorithm parameters are chosen without regard for runtime.

6 Appendix

6.1 Future Inquiries

- A general recommendation for setting the algorithm parameters of Step2 or Step3
- Using crossvalidation to set the algorithm parameters
- Avoiding algorithm parameters choices that cause the asymptotic runtime to become greater than the traditional stepwise selection algorithm

6.2 Additional Tables & Plots

6.2.1 Model 1

Table 17: % Simulations that Selected Correct Model

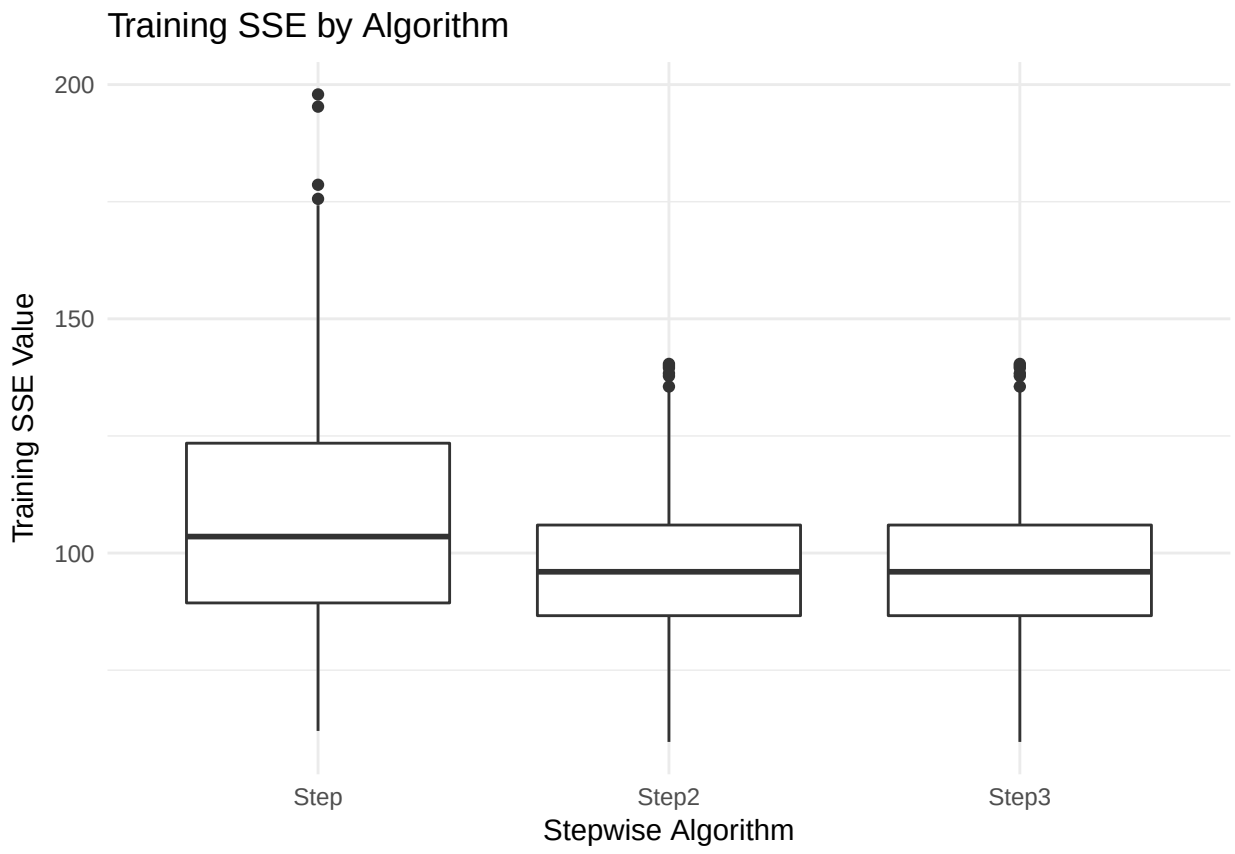
	x
Step	0.554
Step2	0.852
Step3	0.852

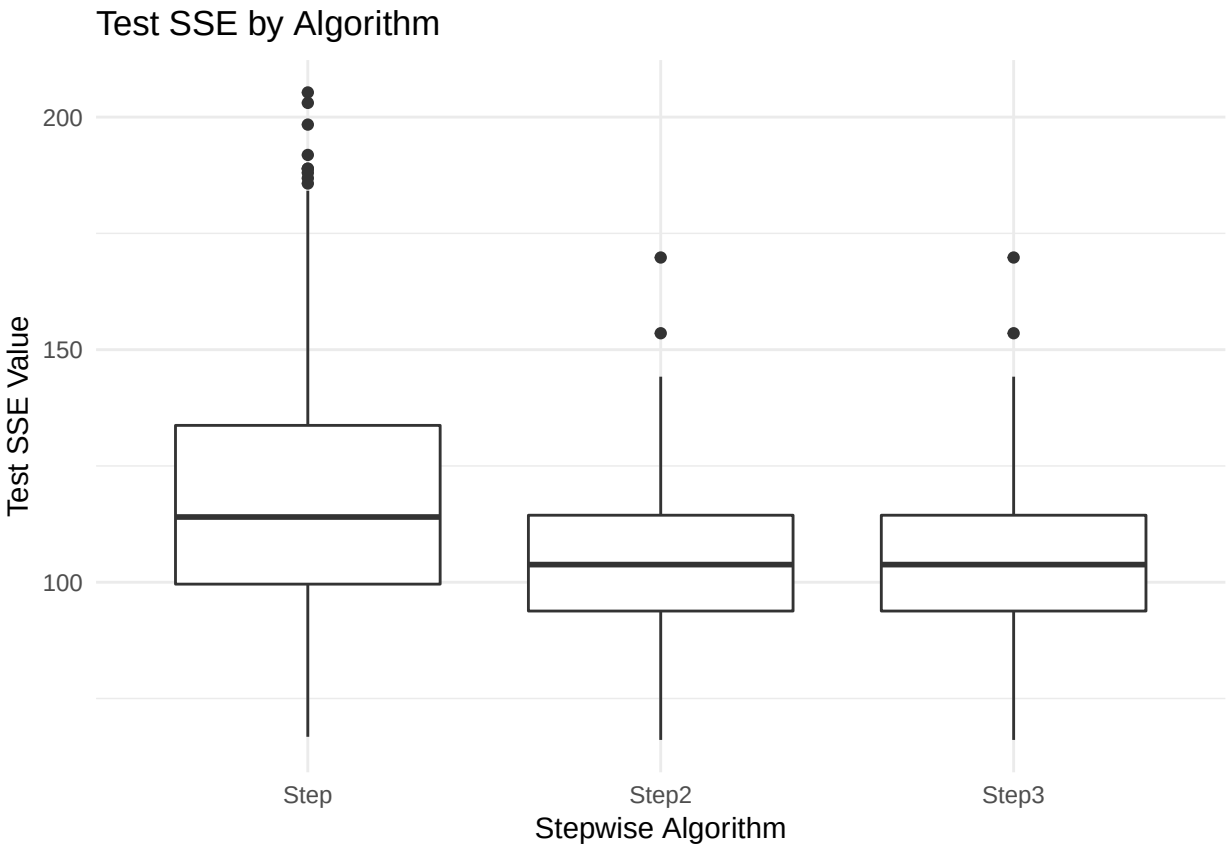
Table 18: % Simulations Selected at Minimum All Correct Covariates

	x
Step	0.659
Step2	0.998
Step3	0.998

Table 19: Inclusion Order for Step, Step2, Step3 Respectively

Order Step	# Step	Order Step2	# Step2	Order Step3	# Step3
||	304	X1X2 ||	852	X1X2 ||	852
X1 | X2 ||	289	X1X2 | X5 ||	39	X1X2 | X5 ||	39
X2 | X1 ||	265	X1X2 | X3 ||	35	X1X2 | X3 ||	35
X2 | X1 | X4 ||	14	X1X2 | X4 ||	31	X1X2 | X4 ||	31
X1 | X2 | X5 ||	12	X1X2 | X6 ||	31	X1X2 | X6 ||	31
X2 | X1 | X3 ||	12	||	2	||	2
X2 | X1 | X5 ||	11	X1X2 | X4 | X6 ||	2	X1X2 | X4 | X6 ||	2
X5 ||	11	X1X2 | X5 | X4 ||	2	X1X2 | X5 | X4 ||	2
X2 | X1 | X6 ||	10	X1X2 | X6 | X4 ||	2	X1X2 | X6 | X4 ||	2
X6 ||	9	X1X2 | X3 | X5 ||	1	X1X2 | X3 | X5 ||	1
X1 | X2 | X3 ||	8	X1X2 | X4 | X3 ||	1	X1X2 | X4 | X3 ||	1
X4 ||	8	X1X2 | X4 | X5 ||	1	X1X2 | X4 | X5 ||	1
X3 ||	7	X1X2 | X5 | X6 ||	1	X1X2 | X5 | X6 ||	1
X1 | X2 | X6 ||	6				
X1 | X2 | X4 ||	4				
X5 | X2 | X1 ||	4				
X3 | X2 | X1 ||	3				
X4 | X1 | X2 ||	3				
X4 | X2 | X1 ||	3				
X6 | X1 | X2 ||	3				
X3 | X1 | X2 ||	2				
X6 | X2 | X1 ||	2				
X1 | X2 | X4 | X5 ||	1				
X1 | X2 | X4 | X6 ||	1				
X2 | X1 | X4 | X6 ||	1				
X3 | X1 | X2 | X5 ||	1				
X3 | X6 ||	1				
X4 | X2 | X1 | X5 ||	1				
X4 | X3 | X1 | X2 ||	1				
X4 | X5 | X2 | X1 ||	1				
X6 | X2 | X1 | X4 ||	1				
X6 | X4 ||	1				


Training SSE by Algorithm
200
150
100
Training SSE Value
Step
Step2
Step3
Stepwise Algorithm


Test SSE by Algorithm
Test SSE Value
200
150
100
Step
Step2
Step3
Stepwise Algorithm

6.2.2 Model 2

Table 20: % Simulations that Selected Correct Model

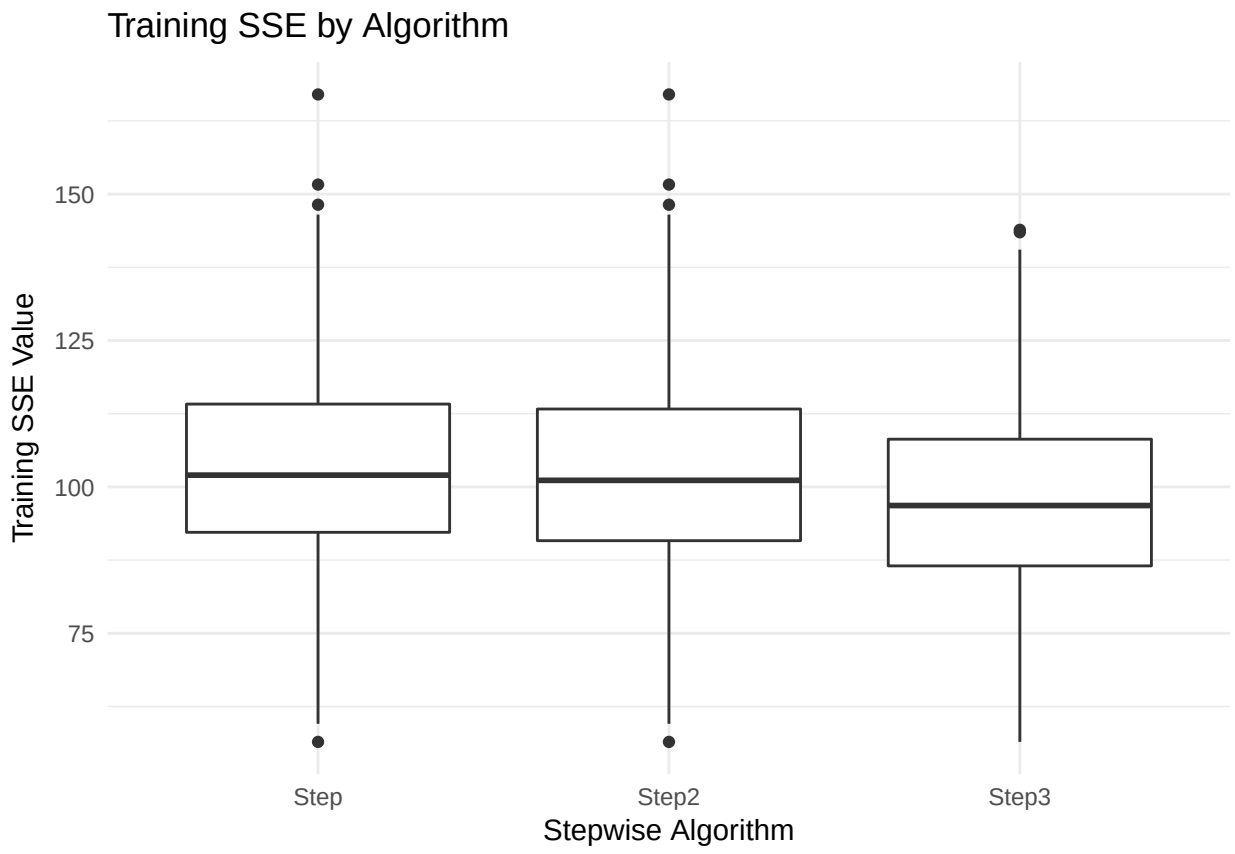
	x
Step	0.104
Step2	0.159
Step3	0.382

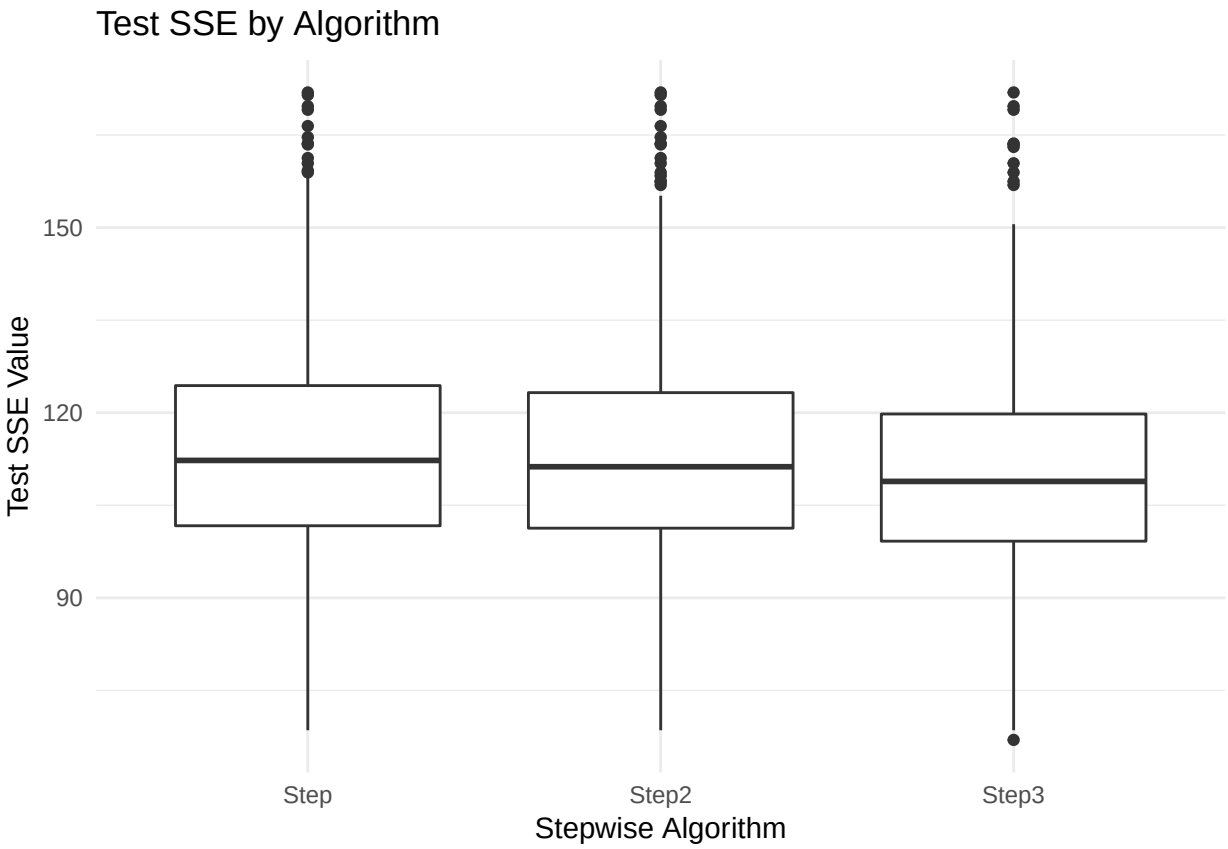
Table 21: % Simulations Selected at Minimum All Correct Covariates

	x
Step	0.134
Step2	0.209
Step3	0.515

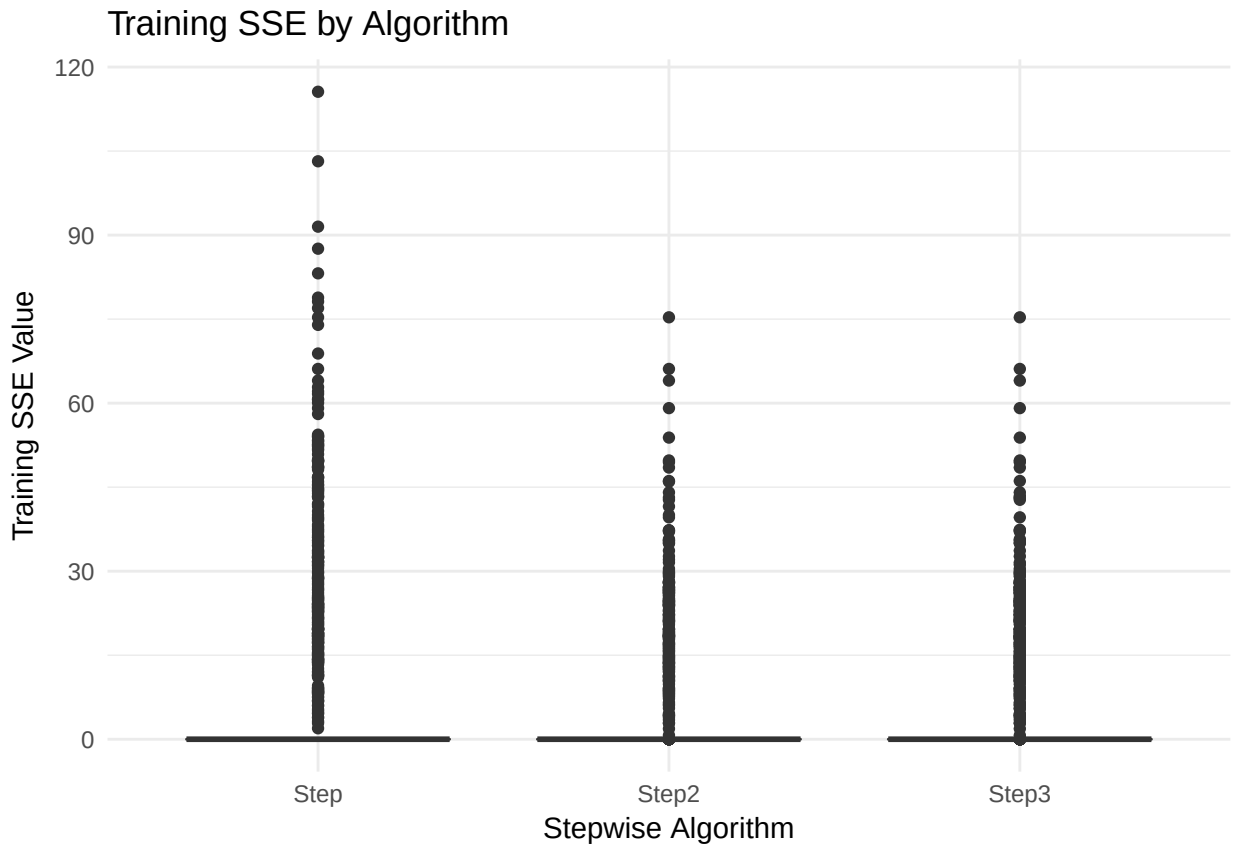
Table 22: Inclusion Order for Step, Step2, Step3 Respectively

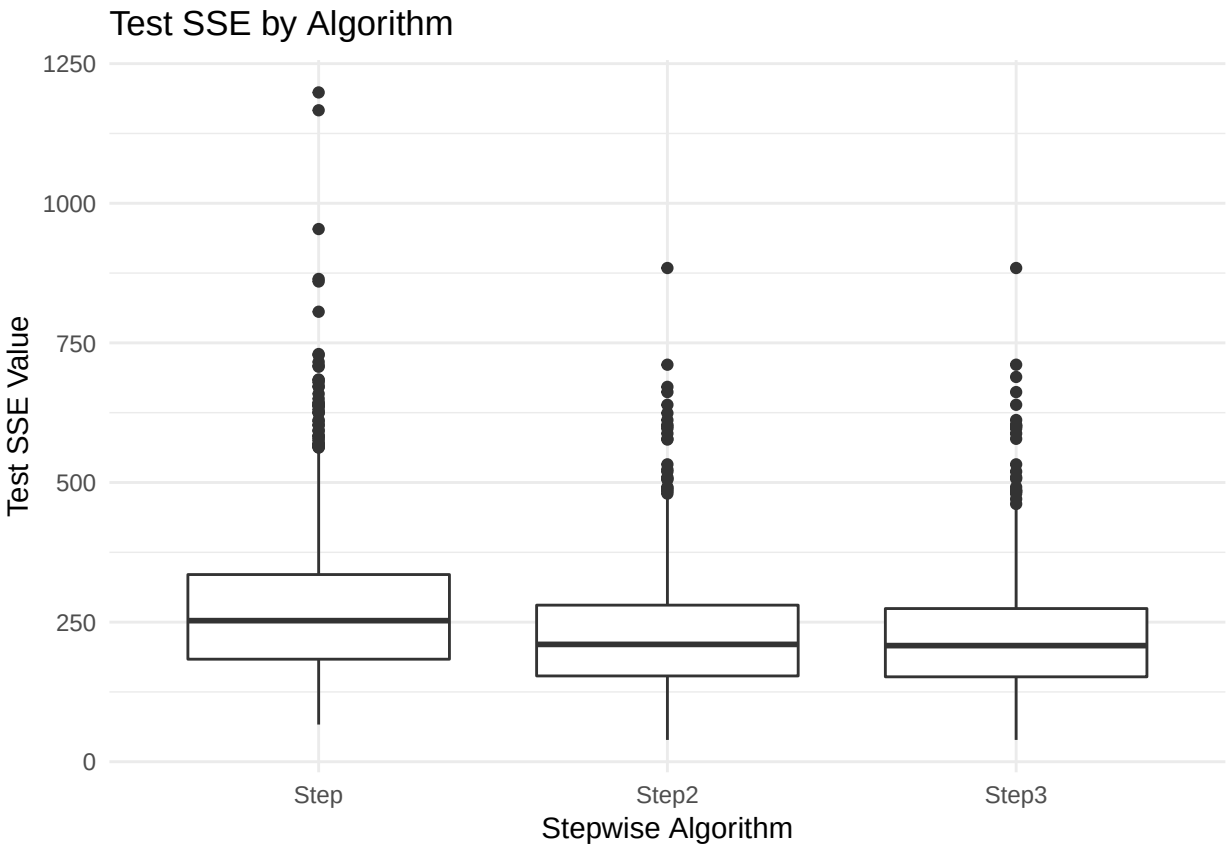
Order Step	# Step	Order Step2	# Step2	Order Step3	# Step3
X2 ||	498	X2 ||	498	X1X2X3 ||	382
X3 ||	98	X3 | X1X2 ||	94	X2 ||	271
X3 | X1 | X2 ||	74	X3 ||	63	X3 ||	63
X1 ||	40	X1 | X2X3 ||	32	X1X2X3 | X10 ||	23
X3 | X1 ||	31	X2X3 | X1 ||	30	X1X2X3 | X7 ||	23
X2 | X6 ||	19	X3 | X1 ||	26	X1 ||	22
X2 | X9 ||	19	X1 ||	22	X1X2X3 | X4 ||	19
X2 | X4 ||	18	X2 | X6 ||	19	X3 | X1 ||	18
X1 | X3 | X2 ||	17	X2 | X9 ||	19	X1X2X3 | X6 ||	17
X2 | X10 ||	17	X2 | X4 ||	18	X2X3 ||	13
X2 | X7 ||	17	X2 | X10 ||	17	X1X2X3 | X5 ||	12
X2 | X5 ||	15	X2 | X7 ||	17	X1X2X3 | X9 ||	12
X2 | X3 ||	12	X2 | X5 ||	14	X2 | X10 ||	10
X2 | X8 ||	12	X2X3 ||	13	X2 | X8 ||	10
X2 | X3 | X1 ||	11	X2 | X8 ||	12	X2 | X9 ||	10
X3 | X10 ||	7	X3 | X1X2 | X10 ||	7	X2 | X5 ||	9
X3 | X1 | X2 | X7 ||	6	X3 | X1X2 | X7 ||	7	X2 | X6 ||	9
X3 | X5 ||	5	X3 | X1X2 | X6 ||	6	X2 | X4 ||	8
X3 | X7 ||	5	X3 | X1X2 | X5 ||	4	X2 | X7 ||	7
X3 | X8 ||	5	X3 | X7 ||	4	X1X2X3 | X8 ||	6
X1 | X9 ||	4	X3 | X8 ||	4	X1X2X3 | X4 | X8 ||	4
X3 | X9 ||	4	X1X2 | X3 ||	3	X3 | X7 ||	4
X3 | X6 | X1 | X2 ||	3	X3 | X10 ||	3	X3 | X8 ||	4
X1 | X3 ||	2	X3 | X5 ||	3	X3 | X9 ||	3
X1 | X4 | X3 | X2 ||	2	X3 | X9 ||	3	X1 | X9 | X2X3 ||	2
X1 | X5 ||	2	X1 | X2X3 | X7 ||	2	X1X2X3 | X6 | X5 ||	2
X2 | X1 | X3 ||	2	X1 | X3 ||	2	X1X2X3 | X8 | X4 ||	2
X2 | X10 | X4 ||	2	X1 | X4 | X2X3 ||	2	X3 | X10 ||	2
X3 | X1 | X2 | X10 ||	2	X1 | X5 ||	2	||	1
X3 | X1 | X2 | X4 ||	2	X1 | X9 | X2X3 ||	2	X1 | X10 ||	1
X3 | X1 | X2 | X6 ||	2	X2 | X10 | X4 ||	2	X1 | X3 | X9 ||	1
X3 | X4 ||	2	X2X3 | X1 | X10 ||	2	X1 | X3 ||	1
X3 | X6 ||	2	X2X3 | X1 | X4 ||	2	X1 | X4 ||	1
||	1	X2X3 | X7 ||	2	X1 | X5 ||	1
X1 | X10 | X6 | X3 ||	1	X3 | X1X2 | X4 | X8 ||	2	X1 | X6 | X3 ||	1
X1 | X10 ||	1	X3 | X1X2 | X4 ||	2	X1 | X6 ||	1
X1 | X3 | X6 | X2 ||	1	X3 | X1X2 | X8 ||	2	X1 | X7 | X8 ||	1
X1 | X3 | X9 ||	1	X3 | X6 ||	2	X1 | X8 ||	1
X1 | X4 ||	1	||	1	X1X2 ||	1
X1 | X6 | X3 ||	1	X1 | X10 | X6 | X3 ||	1	X1X2X3 | X10 | X4 ||	1
X1 | X6 ||	1	X1 | X10 ||	1	X1X2X3 | X10 | X5 ||	1
X1 | X7 | X5 ||	1	X1 | X2X3 | X10 ||	1	X1X2X3 | X5 | X4 ||	1
X1 | X7 | X8 ||	1	X1 | X2X3 | X9 ||	1	X1X2X3 | X5 | X7 ||	1
X1 | X8 ||	1	X1 | X3 | X6 | X2 ||	1	X1X2X3 | X5 | X9 ||	1
X2 | X1 ||	1	X1 | X3 | X9 ||	1	X1X2X3 | X6 | X10 ||	1
X2 | X10 | X3 | X1 ||	1	X1 | X4 ||	1	X1X2X3 | X6 | X9 ||	1


Training SSE by Algorithm
Training SSE Value
150
125
100
75
Step
Step2
Step3
Stepwise Algorithm


Test SSE by Algorithm
Test SSE Value
150
120
90
Step
Step2
Step3
Stepwise Algorithm

6.2.3 Model 3


Training SSE by Algorithm
120
90
60
30
0
Training SSE Value
Step
Step2
Step3
Stepwise Algorithm


Test SSE by Algorithm
1250
1000
750
500
250
0
Test SSE Value
Step
Step2
Step3
Stepwise Algorithm

6.3 Algorithm Source Code URL

- View source code at https://github.com/riley-ashton/Selection/tree/master/R
- View report code at https://github.com/riley-ashton/Selection-Report

References

Hamilton, David. 1987. "Sometimes R 2> R 2 Yx 1+ R 2 Yx 2: Correlated Variables Are Not Always Redundant." *The American Statistician* 41 (2). Taylor & Francis Group: 129–32.