A Sparse Additive Model for High-Dimensional Interactions with an Exposure Variable by Sahir R Bhatnagar, Tianyuan Lu, Amanda Lovato, David L Olds, Michael S Kobor, Michael J Meaney, Kieran O'Donnell, Yi Yang, Celia MT Greenwood.

Response to the reviewers

We thank the reviewers for their constructive comments, which we believe has significantly improved our manuscript. In this document, we reproduce the reviewers comments, and provide our response to each of them below.

Reviewer 1

The paper proposes a method sail for detecting non-linear interactions with an environmental or exposure variable in high-dimensional settings where the strong or weak heredity constraints hold. The asymptotic properties are proved. To estimate unknown parameters, a computationally efficient algorithm with automatic tuning parameter selection is provided. In particular, an R package sail is made available on CRAN. Experiments conducted with simulations and real data validate the effectiveness of the proposed method.

In my opinion, the topic discussed in the paper is interesting and suitable to publish in CSDA. The organization and English quality of the paper is well-written. The theoretical and empirical studies seem to be convincing to show the effectiveness of the proposed method Sail. However, the following questions and comments may be useful for the authors to further improve the paper.

Reviewer Point P 1.1 — In the body part, the main steps of several algorithms are provided. However, the explanation and issues that should be paid attention to use these algorithms are not presented but postponed in appendix. I propose to add some descriptions of the algorithms in the body part.

Reply: Thank you for pointing this out. We have moved the more detailed description of the Algorithm from the appendix to the main text.

Reviewer Point P 1.2 — The statements and symbols used in the paper are not very rigorous. For example, $Y_1, Y_2, \cdots, Y_n$ at the beginning of subsection 1.1 are in fact n observations of the outcome variable. But the authors state $Y = (Y_1, Y_2, \cdots, Y_n) \in \mathbb{R}^n$ be a continuous outcome variable. The environmental variable X_E has the similar problem.

Reply: Thank you for bringing this up. We have fixed this issue and changed the wording as follows:

> Let $Y \in \mathbb{R}$ be a continuous outcome variable, $E \in \mathbb{R}$ a binary or continous environment/exposure vector of known importance, and $X \in \mathbb{R}^p$ a vector of additional predictors, possibly high-dimensional. Assume that we have n observations of each quantity denoted by $Y = (Y_1, \ldots, Y_n) \in \mathbb{R}^n$, $X_E = (E_1, \ldots, E_n) \in \mathbb{R}^n$, and $\mathsf{X} = (X_1^\top, \ldots, X_p^\top) \in \mathbb{R}^{n \times p}$.

Reviewer Point P 1.3 — In the paragraph under the formula (3), it was mentioned that w_E, w_j, w_{JE} are non-negative penalty factors. I consider that this is not rigorous since they are actually some weight parameters to reflect the relative importance of each part in the penalty term. Here, lambda is the penalty factor that should be tuned by some technique. But the authors haven't explain the role of it in the paragraph. In addition, how can the parameters w_E, w_j, w_{JE} be determined? Do they also be chosen based on cross-validation?

Reply: Thanks for this question. The penalty factors are what allow for the adaptive `sail` approach detailed in Section 2.3 which is similar to the adaptive lasso technique (Zou, 2006). More specifically, Algorithm 2 details how these parameters (w_E, w_j, w_{JE}) are determined. Briefly, this is done in three main steps:

1. For a decreasing sequence $\lambda = \lambda_{max}, \ldots, \lambda_{min}$ and fixed α run the algorithm and use cross-validation or a data splitting procedure to determine the optimal value for the tuning parameter: $\lambda^{[opt]} \in \{\lambda_{max}, \ldots, \lambda_{min}\}$. Let $\widehat{\beta_E}^{[opt]}$, $\widehat{\boldsymbol{\theta}}_j^{[opt]}$ and $\widehat{\boldsymbol{\tau}}_j^{[opt]}$ for $j = 1, \ldots, p$ be the coefficient estimates corresponding to the model at $\lambda^{[opt]}$. Note that in this first step, the penalty factors (w_E, w_j, w_{JE}) are all set to be 1.

2. Set the weights to be

   $$w_E = \left(|\widehat{\beta_E}^{[opt]}| + 1/n\right)^{-1}, \; w_j = \left(\|\widehat{\boldsymbol{\theta}}_j^{[opt]}\|_2 + 1/n\right)^{-1}, \; w_{jE} = \left(\|\widehat{\boldsymbol{\tau}_j}^{[opt]}\|_2 + 1/n\right)^{-1} \text{ for } j = 1, \ldots, p$$

   The $1/n$ is added to avoid dividing by zero.

3. Run the `sail` algorithm again, but this time with the penalty factors defined in step 2), and use cross-validation or a data splitting procedure to choose the optimal value of λ.

We have added some text in the paragraph under the formula (3) to point the reader to the adaptive `sail` algorithm for further details on how these weights are estimated.

Reviewer Point P 1.4 — On page 14, it is mentioned that the environment variable is centralized and then the method sail is applied. I wonder why the environmental variable also needed to be centralized. If it is a binary variable, should it be also centralized?

Reply: Thanks for pointing this out. Since we place no restriction on X_E (i.e. it can be binary or continuous), we always center it. We agree that binary variables need not be centralized and this will not have an impact on the estimates for β_E. We centered the binary environment variable because our computational algorithm assumes that all variables have been centered by their mean, irrespective of the variable type. This centering will subsequently affect the estimate of the intercept. We have added the text below to the computation Section 2 to clarify why we center all variables prior to fitting the `sail` method:

> We assume that Y, $\boldsymbol{\Psi}_j$, X_E and $X_E \circ \boldsymbol{\Psi}_j$ have been centered by their sample means $\overline{Y}$, $\overline{\boldsymbol{\Psi}}_j$, $\overline{X}_E$, and $\overline{X_E \circ \boldsymbol{\Psi}_j}$, respectively. Here, $\overline{\boldsymbol{\Psi}}_j \in \mathbb{R}^{m_j}$ and $\overline{X_E \circ \boldsymbol{\Psi}_j} \in \mathbb{R}^{m_j}$ represent the column means of $\boldsymbol{\Psi}_j$ and $X_E \circ \boldsymbol{\Psi}_j$, respectively. Since the intercept (β_0) is not penalized and all variables have been centered, we can omit it from the loss function and compute it once the algorithm has converged for all other parameters.

Reviewer Point P 1.5 — In the last paragraph of section 3, it is mentioned that ϕ_m^{init} is an initial sqrt(n)-consistent estimate of $\phi *_m$. I wonder how we can obtain this type of ϕ_m^{init} in practice. Does this estimate have large influence on the performance of the whole method?

Reply: Thanks for this question. This is directly related to the comment about the penalty factors (w_E, w_j, w_{JE}). In practice, we use the `sail` algorithm to obtain the ϕ_m^{init} as described in Algorithm 2 in the main paper (we also restate the algorithm in our response to comment 1.3 above). In Lemma 1 and Theorem 1, we prove the existence of a local minimizer of our objective function, and that the `sail` estimator is indeed $\sqrt{n}$-consistent.

Reviewer Point P 1.6 — In simulations and real-data experiments, it is mentioned that the tuning parameters are selected by 10-fold cross-validation. I wonder what type of metric is used in cross-validation in each situation. In section 5.2, AUC is used to determine the optimal tuning parameters. But in simulations and subsection 5.1, I guessed that prediction accuracy is utilized according to the contexts. Why different metrics are used in different situations? Does it have some special reasons and benefits to do in this manner?

Reply: We agree that there are many choices of selecting the optimal tuning parameter. In the simulation studies, we used a sample splitting approach (last paragraph of Section 4.2), where the training set was used to fit the model and the validation set was used to select the optimal tuning parameter corresponding to the minimum prediction mean squared error (MSE). This is a common technique in simulation studies since we have the luxury of creating as much data as we want. This has two benefits: 1) it reduces computational cost since we only have to fit the model once for a sequence of tuning parameters and 2) we obtain an unbiased estimate of the prediction error (see for example Hastie et al. (2020); Haris et al. (2016)). This is also the approach we took in the second real data analysis (Section 5.2) because we had enough observations ($n = 8,873$). In the first real-data analysis (Section 5.1), we use cross-validation because we don't have enough data for the sample splitting approach ($n = 189$). For the metric used to select the tuning parameter, we follow Friedman et al. (2010), who use the mean squared prediction error for continuous responses and the AUC for binary responses.

Reviewer Point P 1.7 — In subsection 4.1, I wonder how lasso and adaptive lasso is applied to other situations, like linear interactions and non-linear main effects. Should we consider all interactions between each pair of covariates plus the original covariates as the input variables of lasso or adaptive lasso?

Reply: Thanks for this question. The model you are describing is sometimes referred to in the literature as the *all-pairs* lasso (Bien et al., 2013). The input design matrix is indeed the main effects along with all the interactions $[X_1, X_2, \ldots, X_p, X_E, X_1 : X_E, X_2 : X_E, \ldots, X_p : X_E]$. In Section 1.1 we state that the issue with this approach is that since no constraint is placed on the structure of the model, it is possible that an estimated interaction term is non-zero while the corresponding main effects are zero. The same principle could be applied to non-linear effects. That is, we could use the input design matrix $[f(X_1), f(X_2), \ldots, f(X_p), X_E, f(X_1) : X_E, f(X_2) : X_E, \ldots, f(X_p) : X_E]$, where $f(\cdot)$ is a non-linear transformation of the main effects. The same issue arises however, in that the heredity property is violated. We did not explore this option in our simulations because the lasso was not intended for this purpose. We instead focused on specialized approaches intended for interaction selection while satisfying the heredity property.

Reviewer Point P 1.8 — Please the authors to check the information of each reference to make sure that the provided information is complete and correct. I doubt that the page numbers of Ning et al.(2018), Zou and Zhang(2019) are incorrect.

Reply: Thanks for pointing this out. We have fixed the references.

Reviewer 2

In this paper, the authors introduced a sparse version of allowing a model with non-linear interactions. Primarily, only one key exposure variable is allowed in the proposed model. From the numerical study, when the true model follows the “strong heredity principle", the performance is best.

Reviewer Point P 2.1 — I think that the introduction seems to be too long.

Reply: Thanks for pointing this out. We have removed an entire paragraph from the introduction. We found it difficult to remove more, without significantly affecting the flow of the paper.

Reviewer Point P 2.2 — The notation of sections 1, 2 is not consistent. For example, in section 2.2(weak heredity), the reparametrization α_j should be τ_j because equation (2) has no α_j terms. Also, Equations (4) and (6) have similar parameterization repeatedly. A summarized table about models, reparametrization, penalty terms helps understand the proposed method under strong/weak heredity principles.

Reply: Thank you for pointing out this issue. We have fixed the error. We have also added a summary table in the main text which shows the model types, reparametrization and penalty as follows:

Table 1: Summary of reparametrization and penalty terms for strong and weak heredity `sail` model. Note that the penalty terms are identical for both model types, i.e., the reparametrization only affects the likelihood term of the objective function.

Model	Reparametrization	Penalty
Strong heredity	$\boldsymbol{\tau}_j = \gamma_{jE}\beta_E\boldsymbol{\theta}_j$	$\lambda(1 - \alpha)\left(w_E\lvert\beta_E\rvert + \sum_{j=1}^{p} w_j\lVert\boldsymbol{\theta}_j\rVert_2\right) + \lambda\alpha\sum_{j=1}^{p} w_{jE}\lvert\gamma_{jE}\rvert$
Weak heredity	$\boldsymbol{\tau}_j = \gamma_{jE}(\beta_E \cdot \mathbf{1}_{m_j} + \boldsymbol{\theta}_j)$	$\lambda(1 - \alpha)\left(w_E\lvert\beta_E\rvert + \sum_{j=1}^{p} w_j\lVert\boldsymbol{\theta}_j\rVert_2\right) + \lambda\alpha\sum_{j=1}^{p} w_{jE}\lvert\gamma_{jE}\rvert$

Reviewer Point P 2.3 — In measuring the variable section’s ability, selections of other variables are affected significantly by the selection of X_E. How to evaluate the effect of X_E in TPR? For example, "TPR" of X_E only. Small simulations or some discussions are necessary.

Reply: We agree with this assessment. The premise of our method is built upon the fact that there is an exposure/environment variable with known importance. As such, all of our data generating mechanisms include an important exposure effect. We have evaluated the variable selection rates of the strong heredity sail method and reported the results in a figure shown in Supplemental Section C (Figure 8). We have also added a discussion of these results in the simulation section results (Section 4.3). Figure 1 is one example of the figure we have included in the supplemental section. It represents the model selection results across all simulation replications for scenario 1a). For example, we see that the set of variables $\{X1, X3, X4, E, X3 : E, X4 : E\}$ was selected in 131 of the replications. We can see that the environment variable is always selected across all replications.

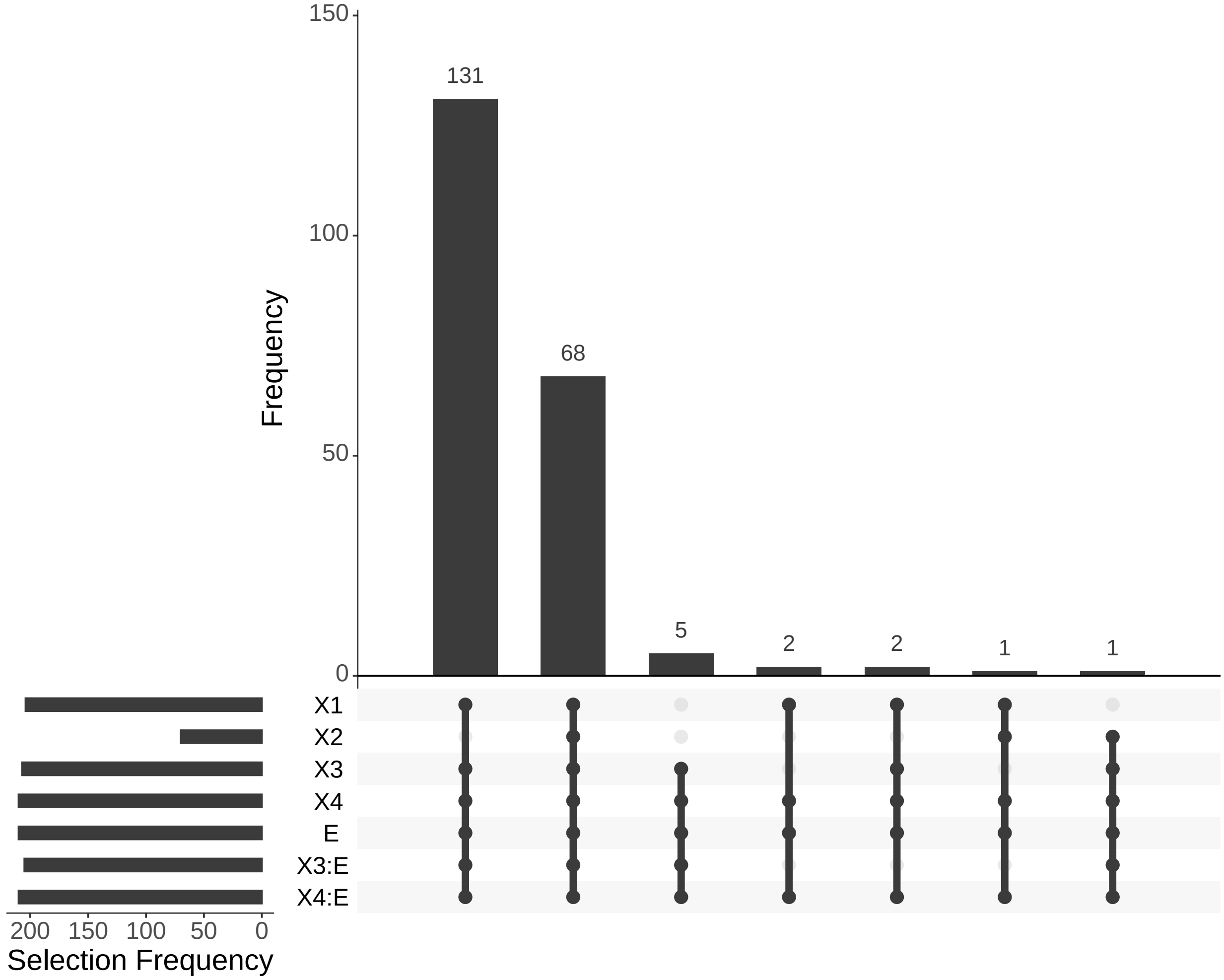

Figure 1: Model selection results for strong heredity sail method. Shown are the selected models and their frequencies. We can see that the environment variable is always selected across all simulation scenarios and replications.

Reviewer Point P 2.4 — Is only the linear effect of the main exposure variable X_E allowed? Otherwise, the simulation can include the non-linear effect of X_E.

Reply: This is indeed a limitation of our current approach. We mentioned this issue in our discussion (Section 6). sail can currently only handle $X_E \cdot f(X)$ or $f(X_E) \cdot X$, i.e., only one of the variables in the interaction can have a non-linear effect. The main challenge is in the computational algorithm.

Allowing both terms to have a non-linear effect would prevent us from using the block coordinate descent algorithm. This is a good starting point however for future directions. In our method development and simulations, we focused on $X_E \cdot f(X)$ because in our real data applications, the exposure was a categorical variable. Our goal was therefore to identify non-linear interactions of the X variable by exposure status.

Reviewer Point P 2.5 — How to generate X_E in the simulations?

Reply: Thank you for pointing this out. The exposure variable (X_E) is generated from a standard normal distribution truncated to the interval [-1,1]. We have added this information to the Simulation Design (Section 4.2).

Reviewer Point P 2.6 — In the theorem, the assumptions in the current manuscript appear to be incomplete. For example, it is necessary to describe 3rd differentiability or boundedness for non-linear functions of covariates clearly.

Reply: The third order differentiability is given in regularity condition C3. This condition is used in the proof for Theorem 1. We note that this is a common regularity condition used in the proofs of several of the papers we have cited on proving the oracle property for penalized regression models Fan and Li (2001); Choi et al. (2010).

Reviewer Point P 2.7 — From the first figure of Figure 1, the estimate of the intercept in the model seems to be biased. The actual function $5x_1$ of the first variable is not balanced. That is, the center is not around zero. Other functions have a balanced mean.

Reply: Thank you for pointing out this issue. We have revised the estimation of the intercept. Since the intercept (β_0) is not penalized and all variables have been centered, we can omit it from the loss function and compute it once the algorithm has converged for all other parameters. We have updated the main text, including the algorithms and computation section. We have also redone all simulation studies to reflect this change. Figure 2 below show the updated results. As we can see, the estimation of the intercept is correct. We would however like to point out that the estimation of all parameters will be biased due to the penalty term in the objective function. This is the reason we are seeing an attenuation effect in the figures. A future direction would be to create a de-biased version of the `sail` estimator as proposed by van de Geer et al. (2014).

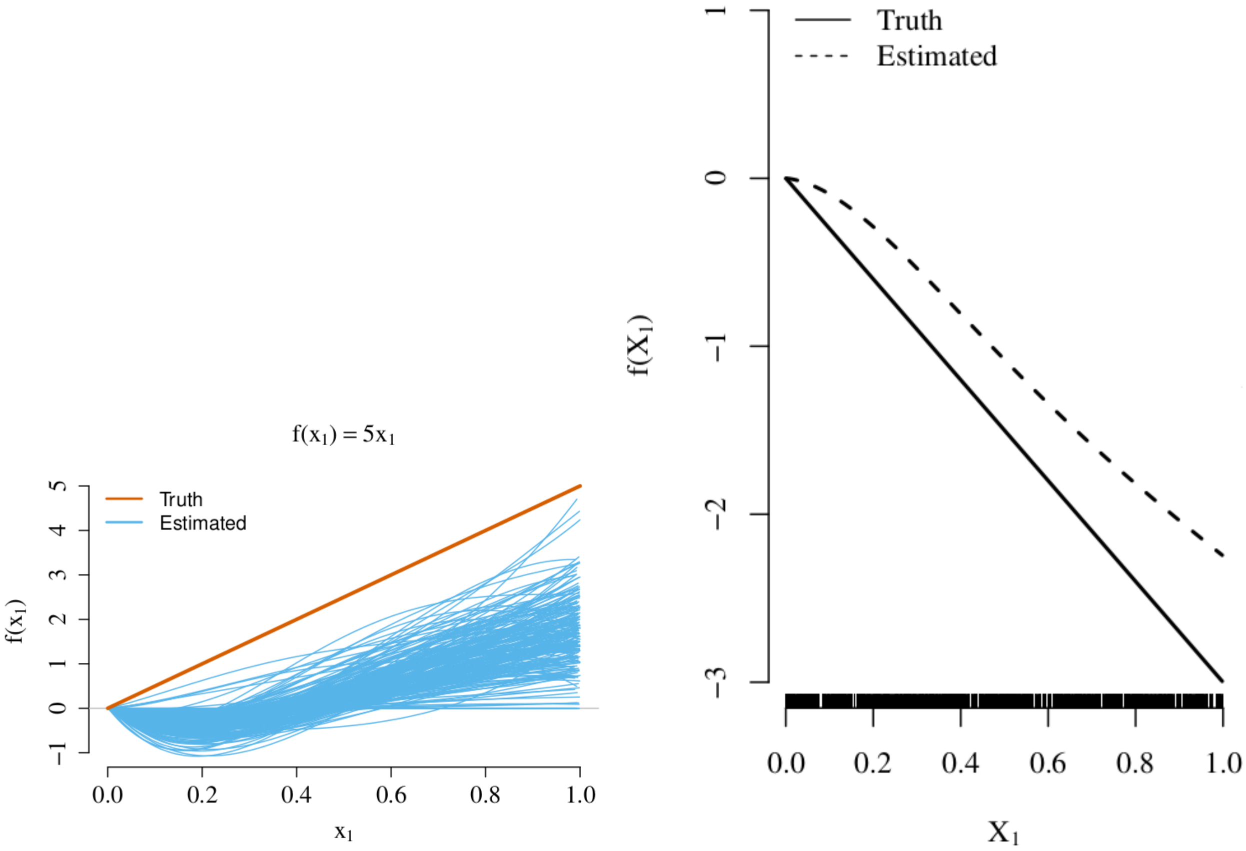

Figure 2: **Left**: True and estimated main effects across 200 replications for strong heredity `sail` in simulation scenario 1a). **Right**: True and estimated main effects for toy example given in section 1.3 of the main text.

Reviewer Point P 2.8 — In the result("(2) Non-linearity simulation scenario"), the performance of the proposed method is not good(not stable) compared to others. It is necessary to check the estimation of intercept.

Reply: As mentioned in our response to point 2.7, we have revised the estimation of the intercept. The performance of our proposed method in the "(2) Non-linearity simulation scenario" has now improved significantly, as shown in Figure 3.

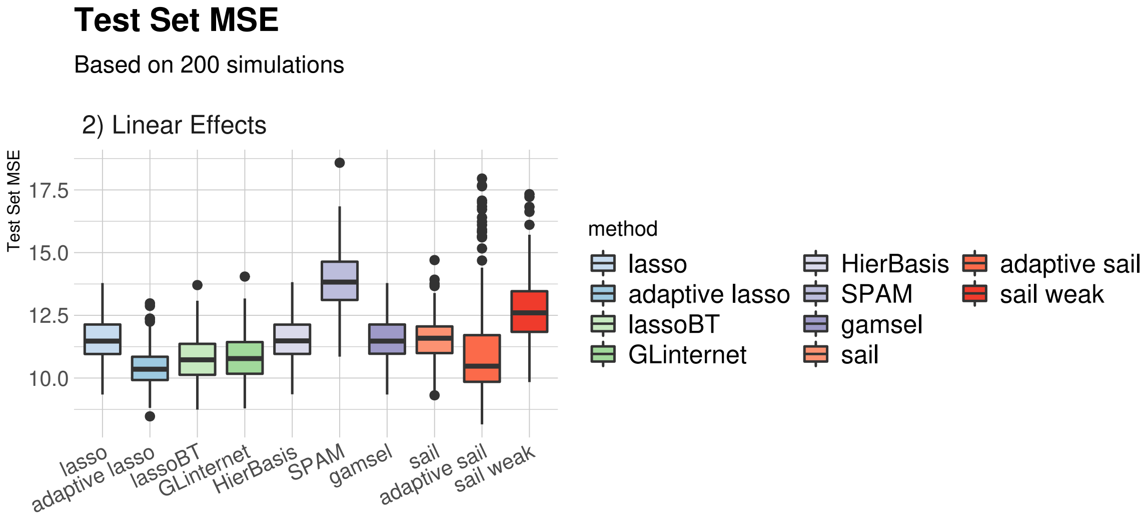

Figure 3: Simulation results across 200 replications for the (2) Non-linearity simulation scenario.

Reviewer Point P 2.9 — Overall, many typos are very confusing to understand the proposed method in the current manuscript. Mainly, it is hard to follow the equations in the appendix. There are also many typos and inconsistent notations. For example, in equation (1) of the appendix, what is λ_m^θ?

Reply: Thank you for raising this issue. λ_m^θ is a typo and should be λ_m. We have fixed this in the proofs given in Supplemental Section A. We have also re-checked the proofs and notation to ensure that it is indeed correct. The original notation in our proof was in fact more complicated and confusing. We tried to simplify the notation as much as possible. We summarize the original notation and the corresponding simplified notation in Table 2. We have also added this Table to the Supplemental Section A.

β_E^*	$\boldsymbol{\theta}_1^{*\top}$	$\boldsymbol{\theta}_2^{*\top}$	$\cdots$	$\boldsymbol{\theta}_p^{*\top}$	γ_{1E}^*	γ_{2E}^*	$\cdots$	γ_{pE}^*
$\boldsymbol{\phi}_1^{*\top}$	$\boldsymbol{\phi}_2^{*\top}$	$\boldsymbol{\phi}_3^{*\top}$	$\cdots$	$\boldsymbol{\phi}_{p+1}^{*\top}$	$\boldsymbol{\phi}_{p+2}^{*\top}$	$\boldsymbol{\phi}_{p+3}^{*\top}$	$\cdots$	$\boldsymbol{\phi}_{2p+1}^{*\top}$
$\lambda(1-\alpha)w_E$	$\lambda(1-\alpha)w_2$	$\lambda(1-\alpha)w_3$	$\cdots$	$\lambda(1-\alpha)w_{p+1}$	$\lambda\alpha w_{p+2,E}$	$\lambda\alpha w_{p+3,E}$	$\cdots$	$\lambda\alpha w_{2p+1,E}$
λ_1	λ_2	λ_3	$\cdots$	λ_{p+1}	λ_{p+2}	λ_{p+3}	$\cdots$	λ_{2p+1}

Table 2: Correspondence between parameters used to simplify the notation in the proofs. The first row shows the actual parameters used in the loss function. The second row shows the corresponding parameters in the simplified notation. The third row shows the actual tuning parameters used in the penalty function. The fourth row shows the corresponding tuning parameters in the simplified notation. This correspondence greatly simplifies the notation used in the proofs.

This notation then allows us to write down the `sail` estimates as

$$\widehat{\mathbf{\Phi}}_n = \arg\min_{\mathbf{\Phi}} Q_n(\mathbf{\Phi}) = -L_n(\mathbf{\Phi}) + n\lambda_m \sum_{m=1}^{2p+1} \|\boldsymbol{\phi}_m\|_2 \,, \tag{1}$$

References

Bien, J., Taylor, J., Tibshirani, R., et al. (2013). A lasso for hierarchical interactions. *The Annals of Statistics*, 41(3):1111–1141. 3

Choi, N. H., Li, W., and Zhu, J. (2010). Variable selection with the strong heredity constraint and its oracle property. *Journal of the American Statistical Association*, 105(489):354–364. 6

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360. 6

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1. 3

Haris, A., Witten, D., and Simon, N. (2016). Convex modeling of interactions with strong heredity. *Journal of Computational and Graphical Statistics*, 25(4):981–1004. 3

Hastie, T., Tibshirani, R., and Tibshirani, R. (2020). Best subset, forward stepwise or lasso? analysis and recommendations based on extensive comparisons. *Statistical Science*, 35(4):579–592. 3

van de Geer, S., Bühlmann, P., Ritov, Y., and Dezeure, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 42(3):1166–1202. 6

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429. 2