Biometrics 000, 000–000 DOI: 000
000 0000

Supporting Information for Post-Selection Inference for Changepoint Detection Algorithms with Application to Copy Number Variation Data

Sangwon Hyun*

Department of Data Sciences and Operations, University of Southern California, Los Angeles, CA.

email: shyun@usc.edu

and

Kevin Z. Lin

Department of Statistics, University of Pennsylvania, Philadelphia, PA.

and

Max G'Sell and Ryan J. Tibshirani

Department of Statistics and Data Science, Carnegie Mellon University, Pittsburgh, PA.

This paper has been submitted for consideration for publication in *Biometrics*

Supplementary Materials

A *Appendix summary*

The code to perform estimation as well as saturated model tests are in https://github.com/robohyun66/binseginf, while the code to perform selected model tests are additionally in https://github.com/linnyKos/selectiveModel. The datasets in this article were obtained from the following places: for the Snijder analysis (Section 4.1), we used the data directly from the GLAD Bioconductor package (https://www.bioconductor.org/packages/release/bioc/html/GLAD.html), and for the Botton analysis (Section 4.3), we used the aCGH data directly from the CNVkit example GitHub package (https://github.com/etal/cnvkit-examples) while we preprocessed the BAM files in the same package using the CNVkit software (https://github.com/etal/cnvkit) to obtain the sequencing data, using the steps outlined in the CNVkit example package.

The following is a brief summary of the supporting information.

Appendix B contains a concise summary of the all the practicalities and extensions of our inferential tools mentioned in the main text. Appendix C reviews circular binary segmentation (CBS) and fused lasso (FL) and provides analogous results of the polyhedron (which is novel for CBS, and is a review of existing results from Hyun et al. (2018) for FL). Appendix D contains the algorithmic details for the selected model test sampler in the known and unknown σ^2 setting, as well as details for the marginalized variants of the saturated model tests. Appendix E contains numerous simulations results and details, including more details of our simulation demonstrating that our inferential tools has more unconditional power than sample splitting in Section 3.4. Appendix F contains a description of the procedure to choose k adaptively and its corresponding simulation results. Appendix G contains additional results on our CNV applications. Appendix H contains the proofs to our theoretical results.

B *Practicalities and extensions*

The sections in the main text formalize the mechanisms to perform post-selective inference with respect to the basic procedure highlighted in Section 1. However, as we've alluded to in the main text, especially in the applications in Section 4, there are many choices and variants the researcher can choose from which can affect the results in practice. To ease researchers into using our inferential tools, we summarize all the combination of choices mentioned in this work that the user faces based on the methods developed in the above sections and their practical impact.

B.1 *Practical considerations.* There are some practical choices that the user needs to make when implementing the procedure. Here, we summarize these choice, as alluded to in Section 1.

- **Algorithm (BS, WBS, CBS and FL)**: FL and BS have similar mechansims, but BS has a simpler mechanism and a less complex selection event, potentially giving higher post-selection conditional power. CBS is specialized for pairs of changepoints, and WBS specializes in localized changepoint detection compared to BS, but both have higher computational burden due to their more complex polyhedra.
- **Conditioning** (Plain or marginalized): Marginalizing over a source of randomness yields tests with higher power than plain inference, but at two costs: increased computational burden due to MCMC sampling being required, and worsened detection ability when using additive noise marginalization. Also, the marginalized p-values are subject to the sampling randomness, and the number of trials T needed to reduce the p-values' intrinsic variability scales with σ^2_{add}.
- **Number of estimated changepoints k (Fixed or data-driven)**: As currently described in Section 2.1, we described methods to find a fixed number of changepoints k. However, we can adopt stopping rules from Hyun et al. (2018) to adaptively choose k.

This increases the complexity of the polyhedra compared to those in Section 3.1, leading to lower statistical power than its fixed-k counterpart. This is shown in Appendix F.

- **Assumed null model (Saturated or selected)**: As mentioned in Section 2.2, selected model tests are valid under a stricter set of assumptions but often yield higher power. Computationally, saturated model tests are often simpler to perform than selected model tests due to the closed form expression of the tail probability.
- **Error variance σ^2 (Known or unknown)**: Saturated model tests require σ^2 to be known. In practice, we need to estimate it in-sample from a reasonable changepoint mean fitted to the same data, or estimated out-of-sample on left-out data. Selected model tests have the advantage of not requiring knowledge of σ^2, but require a larger computational burden, as mentioned in Section 3.2.

B.2 *Extensions.* As mentioned in Hyun et al. (2018), there are many practically-motivated extensions to the baseline procedure mentioned in Section 1 to either improve power or interpretability. We highlight these below. All of these extensions will still give proper Type-I error control under the appropriate null hypotheses.

- **Designing linear contrasts**: The user can make many types of contrast vectors v to fit their analysis, in addition to the segment test contrasts (2), as long as it measurable with respect to $M(y_{\text{obs}})$. One example is the spike test from (Hyun et al., 2018) of single location mean changes. For CNV analysis, it could be useful to test regions between an adjacent pair of changepoints away from the immediately surrounding regions. Also, a step-sign plot (a plot that shows the locations and direction of the changepoints, but not their magnitude) can help the user design contrasts (Hyun et al., 2018).
- **Post-processing the estimated changepoints (Decluttering and screening)**: Multiple detected changepoints too close to one another can hurt the power of segment tests. Post-processing the estimated changepoints based on decluttering (Hyun et al., 2018) or

screening (Lin et al., 2017) so the new set of changepoints are well-separated can lead to contrasts that yield higher power. We show empirical evidence of this improving power of the fused lasso, in Appendix E.7.

- **Pre-cutting**: We can also modify all the algorithms in Section 2.1 to start with an initial existing set of changepoints. This is useful in CGH analyses, when it is not meaningful to consider segments that start in one chromosome and end in another. By pooling information in this manner from separate chromosomal regions, the pre-cut analysis is an improvement over conducting separate analyses in individual chromosomes.

C *Circular binary segmentation and fused lasso*

Below, we review circular binary segmentation (CBS) and fused lasso (FL), as well as the analogous results for their respective polyhedra $\mathbf{\Gamma}$. To clarify, our results for CBS are novel, but our results for FL are taken directly from Hyun et al. (2018). We also include more discussion between all four methods – BS, WBS, CBS and FL.

We additionally note that all the algorithms we mention in our article is sometimes written differently in other changepoint work. Specifically, these algorithms (BS, WBS, and CBS) are sometimes described in the literature as recursively running until internally calculated statistics do not exceed a given threshold level τ. The reason that we choose to instead describe them as running until k steps is two-fold. First, we feel it is easier for a user to specify a priori a reasonable number of steps k, versus a threshold level τ. Second, we can use the method in Hyun et al. (2018) to adaptively choose the number of steps k and still perform valid inferences.

C.1 *Methods.*

The following descriptions are a continuation of the discussions in Section 2.1.

Circular binary segmentation (CBS). The k-step CBS algorithm (Olshen et al., 2004) specializes in detecting *pairs* of changepoints that have alternating directions. At a

step $\ell = 1, \ldots, k$, let $\widehat{\boldsymbol{a}}_{1:(\ell-1)}$, $\widehat{\boldsymbol{b}}_{1:(\ell-1)}$ be the changepoints estimated so far (with the pair a_j, b_j estimated at step j), and let $\boldsymbol{I}_j$, $j = 1, \ldots, 2\ell+1$ be the associated partition of $\{1, \ldots, n\}$. Intervals of length 2 are discarded. Let s_j and e_j denote the start and end index of $\boldsymbol{I}_j$. The next changepoint pair $\widehat{a}_\ell$ and $\widehat{b}_\ell$, and the maximizing interval index $\widehat{j}_\ell$, are found by

$$\{\widehat{j}_\ell, \widehat{a}_\ell, \widehat{b}_\ell\} = \underset{\substack{j \in \{1,\ldots,2(\ell-1)+1)\} \\ a,b \in \{s_j,\ldots,e_j-1\} \,:\, a<b}}{\operatorname{argmax}} \left|\boldsymbol{g}_{(s_j,a,b,e_j)}^T \boldsymbol{y}\right| \quad \text{where} \tag{1}$$

$$\boldsymbol{g}_{(s,a,b,e)}^T \boldsymbol{y} = \sqrt{\frac{1}{\frac{1}{|b-a|} + \frac{1}{|e-s-b+a|}}} \Big(\overline{y}_{(a+1):b} - \overline{y}_{\{s:a\} \cup \{(b+1):e\}} \Big). \tag{2}$$

As before, the new changepoint direction $\widehat{d}_\ell$ is defined based on the sign of the (modified) CUSUM statistic, $\widehat{d}_\ell = \operatorname{sign}(\boldsymbol{g}_{(s_j,a_{\ell+1},b_{\ell+1},e_j)}^T \boldsymbol{y})$ for $j = \widehat{j}_{\ell+1}(\boldsymbol{y})$.

Fused lasso (FL). The fused lasso estimator is defined by solving the convex optimization problem,

$$\min_{\theta \in \mathbb{R}^n} \sum_{i=1}^n (y_i - \theta_i)^2 + \lambda \sum_{i=1}^{n-1} |\theta_i - \theta_{i+1}|, \tag{3}$$

for a tuning parameter $\lambda \geqslant 0$. The fused lasso can be seen as a k-step algorithm by sweeping the tuning parameter from $\lambda = \infty$ down to $\lambda = 0$. Then, at given values of λ (called knots), the FL estimator sequentially introduces an additional changepoint into the solution of (3). See Hyun et al. (2018) for a more in-depth description.

C.2 *Polyhedral selection events for CBS and FL.* The following descriptions are a continuation of the discussions in Section 3.1. We prove the polyhedral selection event for CBS below, as well as review the existing result about the polyhedral selection event for FL.

Selection event for CBS. We define the model for the k-step CBS estimator as

$$M_{1:k}^{\mathrm{CBS}}(\boldsymbol{y}_{\mathrm{obs}}) = \big\{\widehat{\boldsymbol{a}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}}),\ \widehat{\boldsymbol{b}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}}),\ \widehat{\boldsymbol{d}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}})\big\},$$

where now $\widehat{\boldsymbol{a}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}})$ and $\widehat{\boldsymbol{b}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}})$ are the pairs of estimated changepoint locations, and $\widehat{\boldsymbol{d}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}})$ are the changepoint directions, as described in Section 2.1.

PROPOSITION 1: *Given any fixed $k \geqslant 1$ and $\{\boldsymbol{a}_{1:k}, \boldsymbol{b}_{1:k}, \boldsymbol{d}_{1:k}\}$, we can explicitly construct*

$\boldsymbol{\Gamma}$ *where*

$$\big\{\boldsymbol{y} : M^{\mathrm{CBS}}_{1:k}(\boldsymbol{y}, \boldsymbol{w}) = \{\boldsymbol{a}_{1:k}, \boldsymbol{b}_{1:k}, \boldsymbol{d}_{1:k}\}\big\} = \big\{\boldsymbol{y} : \boldsymbol{\Gamma}\boldsymbol{y} \geqslant \boldsymbol{0}\big\}.$$

Let $\boldsymbol{I}^{(\ell)}_j$ *denote the* j*th interval of* $B(\ell)$ *intervals remaining for an intermediate step* $\ell \in \{1, \ldots, k\}$*, and let* $C(x, 2) = \binom{x}{2}$*. Then* $\boldsymbol{\Gamma}$ *has a number of rows equal to*

$$2\sum_{\ell=1}^{k}\Big\{\sum_{j=1}^{B(\ell)} C(|\boldsymbol{I}^{(\ell)}_j| - 1, 2) - 1\Big\}.$$

Selection events for FL, and a brief comparison. The model for the k-step FL estimator is

$$M^{\mathrm{FL}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}}) = \big\{\widehat{\boldsymbol{b}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}}),\ \widehat{\boldsymbol{d}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}}),\ \widehat{\boldsymbol{R}}_{1:k}(\boldsymbol{y}_{\mathrm{obs}})\big\},$$

where $\widehat{\boldsymbol{b}}_{1:k}(\boldsymbol{y})$ and $\widehat{\boldsymbol{d}}_{1:k}(\boldsymbol{y})$ are changepoint locations and directions, and $\widehat{\boldsymbol{R}}_{\ell}(\boldsymbol{y}) \in \mathbb{R}^{n-\ell}$ for $\ell = 1, \ldots, k$ whose elements represent signs of a certain statistic $h_i(\boldsymbol{y})$ calculated at location i in competition for maximization with $\widehat{\boldsymbol{b}}_{\ell}$ at step ℓ. These statistics $h_i(\boldsymbol{y})$ are weighted mean differences at location i and are analogous to CUSUM statistics in BS. Hyun et al. (2018) makes this representation more explicit, proving that for any fixed $k \geqslant 1$ and $\boldsymbol{b}_{1:k}, \boldsymbol{d}_{1:k}, \boldsymbol{R}_{1:k}$, we can explicitly construct $\boldsymbol{\Gamma}$ such that

$$\big\{\boldsymbol{y} : M^{\mathrm{FL}}_{1:k}(\boldsymbol{y}) = \{\boldsymbol{b}_{1:k}, \boldsymbol{d}_{1:k}, \boldsymbol{R}_{1:k}\}\big\} = \{\boldsymbol{y} : \boldsymbol{\Gamma}\boldsymbol{y} \geqslant \boldsymbol{0}\},$$

where $\boldsymbol{\Gamma}$ has the same number of rows as a k-step BS event.

D *Additional algorithmic details*

In this section, we describe additional algorithmic details for selected model tests (from Section 3.2) and marginalized saturated tests (from Section 3.3).

D.1 *Selected model tests, hit-and-run sampling for known* σ^2*.* The following is the hit-and-run sampler to estimate the tail probability of the law of (5). This is for the known σ^2 setting, which differs from the setting described in the main text in Section 3.2. This was briefly described in Fithian et al. (2015) but the authors have later implemented it in ways not

originally described in the above works to make it more efficient. We do not claim novelty for the following algorithm, but simply state it for completion. The original code can be found the repository https://github.com/selective-inference, and we reimplemented it to suite our coding framework and simulation setup.

We specialize our description to test the null hypothesis $H_0 : \boldsymbol{v}_j^T \boldsymbol{\theta} = 0$ against the one-sided alternative $H_1 : \boldsymbol{v}_j^T \boldsymbol{\theta} > 0$. There are some notation to clarify prior to describing the algorithm. Let $\boldsymbol{v}_j \in \mathbb{R}^n$ denote the vector such that

$$\boldsymbol{v}_j^T \boldsymbol{y} = \overline{y}_{\boldsymbol{I}_{j+1}} - \overline{y}_{\boldsymbol{I}_j}.$$

Let $\boldsymbol{A} \in \mathbb{R}^{k \times n}$ denote the matrix such that the last k equations in the above display are satisfied if and only if $\boldsymbol{AY} = \boldsymbol{A}\boldsymbol{y}_{\text{obs}}$. Based on Section 3.1, observe that our goal reduces to sampling from the n-dimensional distribution

$$\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_n), \quad \text{conditioned on} \quad \boldsymbol{\Gamma Y} \geqslant \boldsymbol{0},\ \boldsymbol{AY} = \boldsymbol{A}\boldsymbol{y}_{\text{obs}}. \tag{4}$$

where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix. (Observe that we can set the mean of the above Gaussian distribution to any vector $\boldsymbol{\theta}$ as long as $\boldsymbol{\theta}$ satisfies the null hypothesis. We choose $\boldsymbol{\theta} = \boldsymbol{0}$ for convenience here.)

The first stage of the algorithm *removes the nullspace* of $\boldsymbol{A}$ in the following sense. Construct any matrix $\boldsymbol{B} \in \mathbb{R}^{n \times n}$ such that it has full rank and the last k rows are equal to $\boldsymbol{A}$. Then, consider the following n-dimensional distribution.

$$\boldsymbol{Y}' \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{B}^T \boldsymbol{B}), \quad \text{conditioned on} \quad \boldsymbol{\Gamma B}^{-1} \boldsymbol{Y}' \geqslant \boldsymbol{0},\ (\boldsymbol{Y}')_{(n-k+1):n} = \boldsymbol{A}\boldsymbol{y}_{\text{obs}}. \tag{5}$$

Note that $\boldsymbol{B}^{-1}\boldsymbol{Y}'$ has the same law as (8). Observe that the above distribution is a conditional Gaussian, meaning we can remove the last conditioning event. Towards that end, let $\boldsymbol{\Gamma}''$ denote the first $n-k$ columns of the matrix $\boldsymbol{\Gamma B}^{-1}$, and let $\boldsymbol{u}''$ denote the last k columns of $\boldsymbol{\Gamma B}^{-1}$ left-multiplying $\boldsymbol{A}\boldsymbol{y}_{\text{obs}}$. Also, consider the following partitioning of the matrix $\boldsymbol{B}^T\boldsymbol{B}$,

$$\sigma^2 \cdot \boldsymbol{B}^T \boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_{11} & \boldsymbol{B}_{12} \\ \boldsymbol{B}_{12}^T & \boldsymbol{B}_{22} \end{bmatrix},$$

where $\boldsymbol{B}_{11}$ is a $(n-k)\times(n-k)$ submatrix, $\boldsymbol{B}_{12}$ is a $(n-k)\times k$ submatrix, and $\boldsymbol{B}_{22}$ is a $k\times k$ submatrix. Then, consider the following $n-k$-dimensional distribution.

$$\boldsymbol{Y}'' \sim \mathcal{N}\Big(\boldsymbol{B}_{12}\boldsymbol{B}_{22}^{-1}(\boldsymbol{A}\boldsymbol{y}_{\text{obs}}),\ \boldsymbol{B}_{11}-\boldsymbol{B}_{12}\boldsymbol{B}_{22}^{-1}\boldsymbol{B}_{12}^{T}\Big), \quad \text{conditioned on} \quad \boldsymbol{\Gamma}''\boldsymbol{Y}'' \geqslant -\boldsymbol{u}''. \tag{6}$$

Note that $\boldsymbol{Y}''$ has the same law as the first $n-k$ coordinates of (5).

The next stage of the algorithm *whitens* the above distribution so its covariance is the identity. Let $\boldsymbol{\mu}''$ and $\boldsymbol{\Sigma}''$ denote the mean and variance of the unconditional form of the above distribution (6). Let $\boldsymbol{\Theta}$ be the matrix such that $\boldsymbol{\Theta}\boldsymbol{\Sigma}''\boldsymbol{\Theta}^{T}=\boldsymbol{I}_n$. This must exist since $\boldsymbol{\Sigma}''$ is positive definite. Consider the following $n-k$ dimensional distribution,

$$\boldsymbol{Z} \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{I}_n), \quad \text{conditioned on} \quad \boldsymbol{\Gamma}''\boldsymbol{\Theta}^{-1}\boldsymbol{Z} \geqslant -\boldsymbol{u}''-\boldsymbol{\Gamma}''\boldsymbol{\mu}''. \tag{7}$$

Note that $\boldsymbol{\Theta}^{-1}\boldsymbol{Z}+\boldsymbol{\mu}''$ has the same law as (6). Hence, we have constructed linear mappings F and G between (8) and (7) such that $F(\boldsymbol{Y}) \overset{d}{=} \boldsymbol{Z}$, and $G(\boldsymbol{Z}) \overset{d}{=} \boldsymbol{Y}$ (i.e., have the same distribution).

In order to set up a hit-and-run sampler, generate p unit vectors $\boldsymbol{g}_1,\ldots,\boldsymbol{g}_p$. (The choice of p is arbitrary, and the specific method of generating these p vectors is also arbitrary.) Our hit-and-run sampler will move in the linear directions dictated by $\boldsymbol{g}_1,\ldots,\boldsymbol{g}_p$. We are now ready to describe the hit-and-run sampler in Algorithm 1, which leverages many of the same calculations in (8) and (9) (from Section 3.2). The similarity arises since conditional slices of a multivariate Gaussian still yield Gaussian distributions (loosely speaking), and $\Pi^{\perp}_{\boldsymbol{g}_i}(\boldsymbol{Z}+\boldsymbol{g}_i)=\Pi^{\perp}_{\boldsymbol{g}_i}\boldsymbol{Z}$ (by definition of projections).

The computational efficiency of Algorithm 1 comes from the fact that very few multiplication operations need to be done with the polyhedron matrix $\boldsymbol{\Gamma}''\boldsymbol{\Theta}^{-1}$, a potentially huge matrix. $\boldsymbol{U}$ and $\boldsymbol{\rho}_1,\ldots,\boldsymbol{\rho}_p$ (each vectors of the same length to be defined in the algorithm below) carry all the information needed about polyhedron throughout the entire procedure of generating M samples.

Algorithm 1: MCMC hit-and-run algorithm for selected model test with known σ^2

Choose a number M of iterations.

Set $\boldsymbol{z}^{(0)} = F(\boldsymbol{y}_{\text{obs}})$, as described in the text.

Generate p unit directions $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_p$, each vector of length n.

Compute $\boldsymbol{U} = \boldsymbol{\Gamma}''\boldsymbol{\Theta}^{-1}\boldsymbol{z}^{(0)} + \boldsymbol{u}'' + \boldsymbol{\Gamma}''\boldsymbol{\mu}''$, which represents the "slack" of each constraint.

Compute the p vectors, $\boldsymbol{\rho}_i = \boldsymbol{\Gamma}''\boldsymbol{\Theta}^{-1}\boldsymbol{g}_i$ for $i \in \{1, \ldots, p\}$.

for $m \in \{1, \ldots, M\}$ **do**

Select an index i uniformly from 1 to p.

Compute the truncation bounds

$$\mathcal{V}_{\text{lo}} = \boldsymbol{g}_i^T\boldsymbol{z}^{(m-1)} - \min_{j:(\boldsymbol{\rho}_i)_j > \boldsymbol{0}} U_j/(\boldsymbol{\rho}_i)_j, \quad \text{and} \quad \mathcal{V}_{\text{up}} = \boldsymbol{g}_i^T\boldsymbol{z}^{(m-1)} - \max_{j:(\boldsymbol{\rho}_i)_j < \boldsymbol{0}} U_j/(\boldsymbol{\rho}_i)_j.$$

Sample $\boldsymbol{\alpha}^{(m)}$ from a Gaussian with mean $\boldsymbol{g}_i^T\boldsymbol{z}^{(m-1)}$ and variance 1, truncated to lie between $\mathcal{V}_{\text{lo}}$ and $\mathcal{V}_{\text{up}}$.

Form the next sample

$$\boldsymbol{z}^{(m)} = \boldsymbol{z}^{(m-1)} + \boldsymbol{\alpha}^{(m)}\boldsymbol{g}_i, \quad \text{and} \quad \boldsymbol{y}^{(m)} = G(\boldsymbol{z}^{(m)}).$$

Update the slack variable,

$$\boldsymbol{U} \leftarrow \boldsymbol{U} + \boldsymbol{\alpha}^{(m)}\boldsymbol{\rho}_i.$$

Return the approximate for the tail probability of (6), $\sum_{m=1}^{M} \mathbb{1}[\boldsymbol{v}^T\boldsymbol{y}^{(m)} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}]/M$.

D.2 *Selected model tests, hit-and-run sampling for unknown σ^2.* Below in Algorithm 2, we explicitly describe the hit-and-run sampler we developed to perform selected model tests for unknown σ^2, described in Section 3.2. Similar to the previous subsection, for notational convenience, observe that the last k constraints in (10) can be rewritten as $\boldsymbol{AY} = \boldsymbol{A}\boldsymbol{y}_{(\text{obs})}$ for some matrix $\boldsymbol{A} \in \mathbb{R}^{k\times n}$. Recall that our goal in this setting is to sample from the distribution

$$\boldsymbol{Y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n), \quad \text{conditioned on} \quad \boldsymbol{\Gamma Y} \geqslant \boldsymbol{0},\ \boldsymbol{AY} = \boldsymbol{A}\boldsymbol{y}_{\text{obs}},\ \|\boldsymbol{Y}\|_2 = \|\boldsymbol{y}_{\text{obs}}\|_2. \tag{8}$$

Similar to the above subsection, observe that the mean vector $\boldsymbol{\theta}$ of the Gaussian distribution is arbitrary as long as the null hypothesis is satisfied (and hence we denote $\boldsymbol{\theta} = \mathbf{0}$ for convenience) and the covariance matrix $\boldsymbol{\Sigma}$ of the Gaussian distribution is also arbitrary as long as it is a diagonal matrix since we are conditioning on the event $\|\boldsymbol{Y}\|_2 = \|\boldsymbol{y}_{\text{obs}}\|_2$ (and hence we denote $\boldsymbol{\Sigma} = \boldsymbol{I}_n$ for convenience). As described in the paper, since we are conditioning on all the sufficient statistics under the null hypothesis, sampling from the above distribution is equivalent to sampling uniformly from the set

$$\Big\{\boldsymbol{Y} : \boldsymbol{\Gamma Y} \geqslant \mathbf{0},\ \boldsymbol{AY} = \boldsymbol{A}\boldsymbol{y}_{\text{obs}},\ \|\boldsymbol{Y}\|_2 = \|\boldsymbol{y}_{\text{obs}}\|_2\Big\},$$

which is the goal of the hit-and-run sampler below.

Algorithm 2: MCMC hit-and-run algorithm for selected model test with unknown σ^2

Choose a number M of iterations and set $\boldsymbol{y}^{(0)} = \boldsymbol{y}_{\text{obs}}$.

for $m \in \{1, \ldots, M\}$ **do**

Uniformly sample two unit vectors $\boldsymbol{s}$ and $\boldsymbol{t}$ in the nullspace of $\boldsymbol{A}$.

Compute the set $\mathcal{I} \subseteq [-\pi/2, \pi/2]$ that intersects the set

$$\Big\{\boldsymbol{y} \ :\ \boldsymbol{y} = \boldsymbol{y}^{(m-1)} + r(\omega)\sin(\omega)\cdot\boldsymbol{s} + r(\omega)\cos(\omega)\cdot\boldsymbol{t} \quad \text{for any } \omega \in [-\pi/2, \pi/2]\Big\},$$

for the radius function $r(\omega) = -2(\boldsymbol{y}^{(m-1)})^T(\sin(\omega)\cdot\boldsymbol{s} + \cos(\omega)\cdot\boldsymbol{t})$, with the polyhedral set implied by model selection event,

$$\Big\{\boldsymbol{y} \ :\ \boldsymbol{\Gamma y} \geqslant \mathbf{0}\Big\}.$$

Uniformly sample $\omega^{(m)}$ from $\mathcal{I}$ and form the next sample

$$\boldsymbol{y}^{(m)} = \boldsymbol{y}^{(m-1)} + r(\omega^{(m)})\sin(\omega^{(m)})\cdot\boldsymbol{s} + r(\omega)\cos(\omega^{(m)})\cdot\boldsymbol{t}.$$

Return the approximate for the tail probability of (7), $\sum_{m=1}^{M}\mathbb{1}[\boldsymbol{v}^T\boldsymbol{y}^{(m)} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}]/M$.

Observe that the set $\mathcal{I}$ in each iteration of the above algorithm can be a disjoint set of closed intervals in $[-\pi/2, \pi/2]$.

D.3 *Marginalization over additive noise or over WBS intervals.* In this section, we explicitly describe the algorithms to perform marginalized saturated model tests developed in Section 3.3. The two methods, marginalization over additive noise or over WBS intervals, are strikingly similar. In this section, we use the notation in Section 3.3, where

$$k(\boldsymbol{w}_{\text{obs}}) = \Phi\big(\mathcal{V}_{\text{up}}/\tau\big) - \Phi\big(\boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}/\tau\big)$$
$$g(\boldsymbol{w}_{\text{obs}}) = \Phi\big(\mathcal{V}_{\text{up}}/\tau\big) - \Phi\big(\mathcal{V}_{\text{lo}}/\tau\big),$$

and in this marginalized setting,

$$\mathcal{V}_{\text{lo}} = \boldsymbol{v}^T\big(\boldsymbol{y}_{\text{obs}} + \boldsymbol{w}_{\text{obs}}\big) - \min_{j:\rho_j>0}\big\{\boldsymbol{\Gamma}(\boldsymbol{y}_{\text{obs}} + \boldsymbol{w}_{\text{obs}}\big\}_j/\rho_j,$$
$$\mathcal{V}_{\text{up}} = \boldsymbol{v}^T\big(\boldsymbol{y}_{\text{obs}} + \boldsymbol{w}_{\text{obs}}\big) - \max_{j:\rho_j<0}\big\{\boldsymbol{\Gamma}(\boldsymbol{y}_{\text{obs}} + \boldsymbol{w}_{\text{obs}})\big\}_j/\rho_j,$$

and $\tau = \sigma^2\|\boldsymbol{v}\|_2^2$ and $\boldsymbol{\rho} = \boldsymbol{\Gamma}\boldsymbol{v}/\|\boldsymbol{v}\|_2^2$.

First, in Algorithm 3, we describe the saturated model tests, marginalized over additive noise.

Algorithm 3: Marginalizing over additive noise

Choose a number T of trials.

for $t \in \{1, \dots, T\}$ **do**

Sample the additive noise $\boldsymbol{w}_j$ from $\mathcal{N}(\boldsymbol{0}, \sigma^2_{\text{add}}\boldsymbol{I}_n)$.

Compute $k(\boldsymbol{w}_t)$ and $g(\boldsymbol{w}_t)$.

Return the approximate for the tail probability (12), $\sum_{t=1}^T k(\boldsymbol{w}_t)/\sum_{t=1}^T g(\boldsymbol{w}_t)$.

Next, in Algorithm 4, describe the saturated model tests, marginalized over WBS intervals.

Algorithm 4: Marginalizing over random intervals

Choose a number T of trials.

for $t \in \{1, \ldots, T\}$ **do**

Sample the non-maximizing intervals $\boldsymbol{w}_\ell = (s_\ell, \ldots, e_\ell)$ for $\ell \in \{1, \ldots, B\} \backslash \{\widehat{j}_{1:k}\}$ where s_ℓ, e_ℓ are uniformly drawn from 1 to n and $s_\ell < e_\ell$.

Check to see that $\{\widehat{j}_{1:k}\}$ are still the indices of the maximizing intervals. If not, return to the previous step.

Compute $k(\boldsymbol{w}_t)$ and $g(\boldsymbol{w}_t)$.

Return the approximate for the tail probability (12), $\sum_{t=1}^{T} k(\boldsymbol{w}_t) / \sum_{t=1}^{T} g(\boldsymbol{w}_t)$.

E *Simulations*

In this section, we show simulation examples to demonstrate properties of the segmentation post-selection inference tools presented in the current article.

E.1 *Data generating process.* Let the mean $\boldsymbol{\theta}$ consist of two alternating-direction change-points of size δ in the middle as in (9), chosen to be a realistic example of mutation phenomena as observed in aCGH datasets (Snijders et al., 2001). Specifically, let the sample size be set to be $n = 200$, chosen to be in the scale of the chromosomal data. Then, we model this using the equation below,

$$\textbf{Middle mutation: } \text{for } i \in 1, \ldots, n: \quad y_i \sim \mathcal{N}(\theta_i, 1), \quad \theta_i = \begin{cases} \delta & \text{if } 101 \leqslant i \leqslant 140 \\ 0 & \text{if otherwise,} \end{cases} \tag{9}$$

for the signal size $\delta \in \{0, 0.25, 0.5, 1, 2, 4\}$ with noise level $\sigma^2 = 1$.

E.2 *Methodology.* In the following simulations, we consider the following four estimators (BS, WBS, CBS and FL), each run for two steps. We perform saturated model tests on each estimator, but only perform selected model tests on BS and FL for simplicity, for both known and unknown noise parameter σ^2. We use the basis procedure outlined in Section 1 with a significance level of $\alpha = 0.05$. Throughout the entire simulation suite, the empirical

standard deviation in each of the power curves and detection probabilities is less than 0.02. For each method, for each signal-to-noise size δ, we run more than 250 trials.

E.3 *Type-I error control verification.* First, we examine all our statistical inferences under the global null where $\boldsymbol{\theta} = \mathbf{0}$ to demonstrate their validity – uniformity of null p-values, or type I error control. Specifically, any simulations from the no-signal regime $\delta = 0$ from the middle mutation (9) can be used. When there is no signal, the null scenario $\boldsymbol{v}^T\boldsymbol{\theta} = 0$ is always true so we expect all p-value to be uniformly distributed between 0 and 1. We verify this expected behavior in Figure 7. We notice that the methods that require MCMC (marginalized saturated and selected model tests) requires more trials to converge towards the uniform distribution compared to their counterparts that have exact calculations.

[Figure 1 about here.]

E.4 *Calculating power.* After verifying that our inferential tools have valid Type-I control, we now want to investigate their power – how often they correctly deem an estimated changepoint as significant when it is near a true changepoint. Since the tests are performed only when a changepoint is selected, it is necessary to separate the detection ability of the estimator from power of the test. To this end, we define the following quantities,

$$\text{Conditional power} = \frac{\#\text{ correctly detected \& rejected}}{\#\text{ correctly detected}} \tag{10}$$

$$\text{Detection probability} = \frac{\#\text{ correctly detected}}{\#\text{ tests conducted}} \tag{11}$$

$$\text{Unconditional power} = \text{Detection} \times \text{Conditional power} \tag{12}$$

The overall power of an inference tool can only be assessed by examining the conditional and unconditional power together. We consider a detection to be correct if it is within ± 2 of the true changepoint locations.

E.5 *Power comparison across signal sizes δ.* For saturated model tests, we perform additive-noise inferences using Gaussian $\mathcal{N}(0, \sigma^2_{\text{add}})$ with $\sigma_{\text{add}} = 0.2$ for BS, FL, and CBS. For WBS, we employ the randomization scheme as described in Section 3.3 with $B = n$. With the metrics in (11)-(12), we examine the performance of the four methods. The solid lines in Figure 8 show the "plain" method where model selection based on $M(y_{\text{obs}})$. The dotted lines show the marginalized counterparts where the model selection is $M(y_{\text{obs}}, W)$, margnialized over W.

We see in Figure 8 that WBS and CBS have higher conditional and unconditional power than BS. This is expected since the former two are more adept for localized change-points of alternating directions. FL noticeably under-performs in power compared to segmentation methods. This is partially caused by FL's detection behavior, and can be explained by examining alternative measures of detection and improved with post-processing. This investigation is deferred to Appendix E.7. The marginalized versions of each algorithm have noticeably improved power, but almost unnoticeably worse detection than their non-randomized, plain versions (middle panel of Figure 8). Combined, in terms of unconditional power, marginalized inferences clearly dominate their plain counterparts.

Selected model inference simulations are shown in Figure 9. Surprisingly, there is an almost inconceivable drop in power from unknown σ^2 to known σ^2. Compared to the saturated model tests in Figure 8, there is smaller power gap between FL and BS. Also, selected model tests appear to have higher power than saturated model tests. In general however, it is hard to compare the power of saturated and selected models due to the clear difference in model assumptions.

[Figure 2 about here.]

[Figure 3 about here.]

E.6 *More details about sample splitting.* Here, we discuss the details used to generate Figure 2. As mentioned in the main text, sample splitting is another valid inference technique. After splitting the dataset in half based on even and odd indices, we run a changepoint algorithm on one dataset and conduct classical one-sided t-test on the other. This is the most comparable test, as it does not assume σ^2 is known and conducts a one-sided test of the null $H_0 : \boldsymbol{v}^T\boldsymbol{\theta} = 0$. Instead of ± 2 slack used for calculating detection in post-selective inference (dotted and dashed lines), ± 1 was used for sample splitting inference (solid line). The loss in detection accuracy in the middle panel of Figure 2 shows the downside of halving data size for detection. Unconditional power for marginalized saturated model tests and selected model tests are noticeably higher than the other two.

We note that the results in Figure 2 were based on approximate detection. This choice of approximate detection is somewhat arbitrary, and it is informative to see if the results would change if we considered only exact detection. We can see from Figure 10 that randomized TG p-values have comparable power with sample splitting inferences, among tests that are regarding exactly the right changepoints.

[Figure 4 about here.]

E.7 *Power comparison using unique detection.* FL was appeared to have a large drop in power compared to segmentation algorithms. In addition to these three measures shown in Appendix E.4, for multiple changepoint problems like middle mutations it is useful to measure performance using an alternative measure of detection called unique detection. This is useful because some algorithms – mainly FL, but to also BS to some extent, primarily in later steps – admit "clumps" of nearby points. If this clumped detection pattern occurs in early steps, the algorithm requires more steps than others to fully admit the correct changepoints. In this case, detection alone is not an adequate metric, and unique detection

can be used in place.

$$\text{Unique detection probability} = \frac{\#\text{changepoints which were approximately detected}}{\#\text{number of true changepoints.}} \quad (13)$$

In plain words, unique detection is measuring how many of the true changepoint locations have been approximately recovered.

We present a simple case study. In addition to a 2-step FL, imagine using a 3-step FL, but with post-processing. For post-processing, declutter by centroid clustering with maximum distance of 2, and test the $k_0 < 3$ changepoints, pitting the resulting segment test p-values against $0.05/k_0$. A 2-step FL's detection does not reach 1 even at high signals ($\delta = 4$) because of the aforementioned clumped detection behavior. The resulting segment tests are also not powerful, since the segment test contrast vectors consist of left and right segments which do not closely resemble true underlying piecewise constant segments in the data. However, when detection is replaced with unique detection, two things are noticeable. First, decluttered FL's detection performance is noticeably improved when going from 2 to 3 steps. Also, when unconditional power is calculated using unique detection, BS does not have as large of an advantage over the the several variants of fused lasso. This is shown in Figure 11. We see from the right figure (compared to the left) that the a "decluttered" version of 2- or 3-step FL has much closer unconditional power to BS.

[Figure 5 about here.]

E.8 *Power comparison with different mean shape.* The synthetic mean discussed here consists of a single upward changepoint piece-wise constant mean, as shown in (14) and Figure 12 (right). This is chosen to be another realistic example of the mutation phenomenon as observed in aCGH datasets from Snijders et al. (2001), in addition to the case shown in the main text. We focus on the *duplication* mutation scenario, but the results apply similarly to deletions. As before, the sample size $n = 200$ was chosen to be in the scale of the data length in a typical aCGH dataset in a single chromosome. For saturated model tests, WBS

no longer outperforms BS in power. This is expected since there is only a single changepoint not accompanied by opposing-direction changepoints.

$$\textbf{Edge mutation:} \quad y_i \sim \mathcal{N}(\theta_i, 1), \;\; \theta_i = \begin{cases} \delta & \text{if } 161 \leqslant i \leqslant 200 \\ 0 & \text{if otherwise} \end{cases} \tag{14}$$

[Figure 6 about here.]

[Figure 7 about here.]

F *Model size selection using information criteria – choosing k adaptively*

Throughout the article we assume that the number of algorithm steps k is fixed. Hyun et al. (2018) introduces a stopping rule based on information criteria (IC) which can be characterized as a polyhedral selection event. The IC for the sequence of models $M_{1:\ell}$, for $\ell = 1, \ldots, n-1$ is

$$J(M_{1:\ell}) = \|\boldsymbol{y} - \widehat{\boldsymbol{y}}_{M_{1:\ell}(\boldsymbol{y})}\|_2^2 + p\big(M_{1:\ell}(\boldsymbol{y})\big). \tag{15}$$

We omit the dependency on $\boldsymbol{y}$ when obvious. We use the BIC complexity penalty $p(M_k) = \sigma^2 \cdot k \cdot \log(n)$ for this article. Also define $S_\ell(\boldsymbol{y}) = \text{sign}\left(J(M_{1:\ell}) - J(M_{1:(\ell-1)})\right)$ to be the sign of the difference in IC between step $\ell - 1$ and ℓ. This is a $+1$ for a rise and -1 for a decline. A data-dependent stopping rule $\widehat{k}$ is defined as

$$\widehat{k}(y) = \min\{k : S_k(y) = S_{k+1}(y) = \ldots = S_{k+q}(y) = 1\} \tag{16}$$

which is a local minimization of IC, defined as the first time q consecutive rises occur. As discussed in Hyun et al. (2018), $q = 2$ is a reasonable choice for the changepoint detection. To carry out valid post-selective inference, we condition on the selection event $\mathbb{1}[S_{1:(k+q)}(\boldsymbol{y}) = S_{1:(k+q)}(\boldsymbol{y}_{\text{obs}})]$, which is enough to determine $\widehat{k}$. A k-step model for k chosen by (16) can be understood to be $M_{1:\widehat{k}}(\boldsymbol{Y}) = M_{1:k}(\boldsymbol{y}_{\text{obs}})$. The corresponding selection event $P_{M_{1:\widehat{k}}}$ is with the additional halfspaces, as outlined in Hyun et al. (2018). Simulations in Figure 14 show

that introducing IC stopping is valid, by controlled type-I error, but comes at the cost of considerable power loss.

[Figure 8 about here.]

G *Additional results to CNV analyses in Section 4*

G.1 *Additional details to Snijders analysis in Section 4.1.* In this section, we provide additional details associated with results in Section 4.1. As part of the preprocessing, we also remove single outliers which are at least three standard deviations away from its surrounding points. Afterwards, we set σ^2 for all the saturated model tests in that section for a particular cell line by computing the empirical variance after fitting a pre-cut 10-step WBS across said cell line.

G.2 *Additional details and results to follow-up analysis of Snijders dataset in Section 4.2.* In this section, we provide additional details and results associated with the heavy-tail study performed in Section 4.2. Throughout all the simulations in that section, we set σ^2 for all the saturated model tests by computing the empirical variance after fitting a pre-cut 10-step WBS across the entire cell line GM01750.

We had mentioned the bootstrap substitution method proposed by Tibshirani et al. (2018) in Section 4.2, and we contrast the performance of our bootstrapped variant (shown in Figure 4D) to the variant originally proposed in Tibshirani et al. (2018). First, let $\boldsymbol{\beta}$ denote $\overline{\theta}$, the grand mean of $\boldsymbol{\theta}$ (i.e., a 0-changepoint model). Then, the main idea in Tibshirani et al. (2018) is to approximate the law of $\boldsymbol{v}^T\boldsymbol{Y}$ used to construct the TG statistic (8) with the bootstrapped distribution of $\boldsymbol{v}^T(\boldsymbol{Y}-\boldsymbol{\beta})$ by bootstrapping the residuals, $\boldsymbol{y}-\overline{y}\cdot\mathbb{1}_n$. Here, the empirical grand mean $\overline{y}\cdot\mathbb{1}_n$ represents the simplest model with no changepoints for a length-n vector. While this estimate will usually ensure the resulting p-values has valid Type-I error control, we see that it produces overly conservative p-values in practice if there exist *any*

changepoints (Figure 15). Hence, as mentioned in Section 4.2, we suggest researchers to use the bootstrapping variant we proposed over the original method in Tibshirani et al. (2018), since it yields more powerful p-values and does not seem to violate the Type-I error control in practice.

[Figure 9 about here.]

G.3 *Additional details of Botton analysis in Section 4.3.* In this section, we provide additional details associated with the results in Section 4.3. The data for these analyses originated from `https://github.com/etal/cnvkit-examples`, the GitHub repository associated with Talevich et al. (2016). In our experience, our inferential tools work well on sequencing data as long as it is appropriately preprocessed. Specifically, we preprocessed our sequencing data using CNVkit according to the scripts within the CNVkit GitHub repository, which converts the BAM file containing the counts into the desired $\log_2$ copy number ratio, and applies sophisticated processing to correct for coverage biases. We then additionally filtered out outliers based on whether or not a particular data point lied outside of 1.5 times the interquantile range (IQR) of the median within a local window. This is done, as suggested by the documentation to the CNVkit pipeline. Since the resulting aCGH data is originally more than 7 times the length of the sequencing data (in terms of the number of probes used), we performed a down-sampling on the aCGH data by averaging each non-overlapping consecutive set of $\log_2$ values from 7 probes into one $\log_2$ value. This ensured that the number of samples for each chromosome was roughly comparable between sequencing and aCGH datasets. To use our inferential tools, we set σ^2 for all analyses on sequencing data by computing the standard deviation of the residuals after fitting a 5-step WBS model on each chromosome aside from chromosome 5 and 10. Similarly, we set σ^2 for all analyses on aCGH data by computing the standard deviation of the residuals after fitting a 0-changepoint model on each chromosome aside from chromosome 5 and 10. We used a 0-changepoint model for

the aCGH data since our diagnostics (similar to those done in Section 4.2) showed that the results were slightly heavy-tailed. Hence, as discussed in Appendix G.2, by fitting a 0-changepoint model, this empirically retains the Type-I error control at the cost of more conservative inference.

H *Proofs of results*

H.1 *Proof of Proposition 1, (BS).*

Proof. When $k = 1$, $2(n-2)$ linear inequalities characterize the single changepoint model $\{b_1, d_1\}$:

$$d_1 \cdot \boldsymbol{g}_{(1,b_1,n)}^T \boldsymbol{y} \geqslant \boldsymbol{g}_{(1,b,n)}^T \boldsymbol{y}, \quad \text{and} \quad d_1 \cdot \boldsymbol{g}_{(1,b_1,n)}^T \boldsymbol{y} \geqslant -\boldsymbol{g}_{(1,b,n)}^T \boldsymbol{y}, \quad b \in \{1, \ldots, n-1\} \backslash \{b_1\}.$$

Now by induction, assume we have constructed a polyhedral representation of the selection event up through step $k-1$. All that remains is to characterize the kth estimated changepoint and direction $\{b_k, d_k\}$ by inequalities that are linear in y. This can be done with $2(n-k-1)$ inequalities. To see this, assume without a loss of generality that the maximizing interval is $j_k = k$; then $\{b_k, d_k\}$ must satisfy the $2(|\boldsymbol{I}_k| - 2)$ inequalities

$$d_k \cdot \boldsymbol{g}_{(s_k,b_k,e_k)}^T \boldsymbol{y} \geqslant \boldsymbol{g}_{(s_k,b,e_k)}^T \boldsymbol{y} \quad \text{and} \quad d_k \cdot \boldsymbol{g}_{(s_k,b_k,e_k)}^T \boldsymbol{y} \geqslant -\boldsymbol{g}_{(s_k,b,e_k)}^T \boldsymbol{y}, \quad b \in \{s_k, \ldots, e_k - 1\} \backslash \{b_k\}.$$

For each interval $\boldsymbol{I}_\ell$, for $\ell = 1, \ldots, k-1$, we also have $2(|\boldsymbol{I}_\ell| - 1)$ inequalities

$$d_k \cdot \boldsymbol{g}_{(s_k,b_k,e_k)}^T \boldsymbol{y} \geqslant \boldsymbol{g}_{(s_\ell,b,e_\ell)}^T \boldsymbol{y} \quad \text{and} \quad d_k \cdot \boldsymbol{g}_{(s_k,b_k,e_k)}^T \boldsymbol{y} \geqslant -\boldsymbol{g}_{(s_\ell,b,e_\ell)}^T \boldsymbol{y}, \quad b \in \{s_\ell, \ldots, e_\ell - 1\}.$$

The last two displays together completely determine $\{b_k, d_k\}$, and as $\sum_{\ell=1}^k |\boldsymbol{I}_\ell| = n$, we get our desired total of $2(n-k-1)$ inequalities.

H.2 *Proof of Proposition 2, (WBS).*

Proof. The construction of $\boldsymbol{\Gamma}$ is basically the same as that for BS in Proposition 1; the only difference is that, at step k, the inequalities defining the new rows of $\boldsymbol{\Gamma}$ are based on the intervals w_{j_k} and w_ℓ, $\ell \in J_k \backslash \{j_k\}$, instead of $\boldsymbol{I}_{j_k}$ and $\boldsymbol{I}_\ell$, $\ell \neq j_k$, respectively. To compute the

upper bound on the number of rows m, observe that in step $\ell \in \{1, \ldots, k\}$, there are at most $B - \ell + 1$ intervals remaining. Among these, the interval j_k contributes $p - 2$ inequalities, and the remaining $B - \ell$ intervals contributes $p - 1$ inequalities.

H.3 *Proof of Proposition 1, (CBS).*

Proof. The proof follows similarly to the proof of Proposition 1. Observe that for any $k' < k$, the model $M^{\text{CBS}}_{1:k'}(\boldsymbol{y}_{\text{obs}})$ is strictly contained in the model $M^{\text{CBS}}_{1:k}(\boldsymbol{y}_{\text{obs}})$. Hence, we can proceed using induction, and let b_i for $i \in \{1, \ldots, k\}$ denote $\widehat{b}_i$ for simplicity, and do the same for a_i, d_i and j_i. Let $C(x, 2) = \binom{x}{2}$ for simplicity as well.

For $k = 1$, the following $2 \cdot (C(n-1, 2) - 1)$ inequalities characterize the selection of the changepoint model $\{a_1, b_1, d_1\}$,

$$d_1 \cdot \boldsymbol{g}^T_{(1,a_1,b_1,n)}\boldsymbol{y} \geqslant \boldsymbol{g}^T_{(1,r,t,n)}\boldsymbol{y}, \quad \text{and} \quad d_1 \cdot \boldsymbol{g}^T_{(1,a_1,b_1,n)}\boldsymbol{y} \geqslant -\boldsymbol{g}^T_{(1,r,t,n)}\boldsymbol{y},$$

for all $r, t \in \{1, \ldots, n-1\}$ where $r < t$, $r \neq a_1$ and $t \neq b_1$.

By induction, assume we have constructed the polyhedra for the model, $M^{\text{CBS}}_{1:(k-1)}(\boldsymbol{y}_{\text{obs}}) = \{\boldsymbol{a}_{1:(k-1)}, \boldsymbol{b}_{1:(k-1)}, \boldsymbol{d}_{1:(k-1)}\}$. To construct $M^{\text{CBS}}_{1:k}(\boldsymbol{y}_{\text{obs}})$, all that remains is to characterize the kth parameters $\{a_k, b_k, d_k\}$. To do this, assume that j_k corresponds with the interval $\boldsymbol{I}_k$ having the form $\{s_k, \ldots, e_k\}$. Within this interval, we form the first $2 \cdot (C(|\boldsymbol{I}_{j_k}| - 1, 2) - 1)$ inequalities of the form,

$$d_k \cdot \boldsymbol{g}^T_{(s_k,a_k,b_k,e_k)}\boldsymbol{y} \geqslant \boldsymbol{g}^T_{(s_k,r,t,e_k)}\boldsymbol{y} \quad \text{and} \quad d_k \cdot \boldsymbol{g}^T_{(s_k,a_k,b_k,e_k)}\boldsymbol{y} \geqslant -\boldsymbol{g}^T_{(s_k,r,t,e_k)}\boldsymbol{y}$$

for all $r, t \in \{s_k, \ldots, e_k - 1\}$ where $r < t$ and $r \neq a_k$ and $t \neq b_k$. The remaining inequalities originate from the remaining intervals. For each interval $\boldsymbol{I}_\ell$, for $\ell \in \{1, \ldots, 2k-1\} \backslash \{j_k\}$, let $\boldsymbol{I}_\ell$ have the form $\{s_\ell, \ldots, e_\ell\}$. We form the next $2 \cdot C(|\boldsymbol{I}_\ell| - 1, 2)$ inequalities of the form

$$d_k \cdot \boldsymbol{g}^T_{(s_k,a_k,b_k,e_k)}\boldsymbol{y} \geqslant \boldsymbol{g}^T_{(s_\ell,r,t,e_\ell)}y \quad \text{and} \quad d_k \cdot \boldsymbol{g}^T_{(s_k,a_k,b_k,e_k)}\boldsymbol{y} \geqslant -\boldsymbol{g}^T_{(s_\ell,r,t,e_\ell)}\boldsymbol{y}$$

for all $r, t \in \{s_\ell, \ldots, e_\ell - 1\}$ where $r < t$.

H.4 *Proof of Proposition 3, (Marginalization).*

Proof. For concreteness, we write the proof where $\boldsymbol{W}$ represents additive noise, but the proof generalizes to the setting where $\boldsymbol{W}$ represents random intervals easily. First write $T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v})$ as an integral over the joint density of $\boldsymbol{W}$ and $\boldsymbol{Y}$,

$$\begin{aligned} T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}) &= P(\boldsymbol{v}^T\boldsymbol{Y} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}} | M(\boldsymbol{Y}+\boldsymbol{W}) = M(\boldsymbol{y}_{\text{obs}}+\boldsymbol{W}), \Pi_{\boldsymbol{v}}^{\perp}\boldsymbol{Y} = \Pi_{\boldsymbol{v}}^{\perp}\boldsymbol{y}_{\text{obs}}) \\ &= \int \mathbb{1}(\boldsymbol{v}^T\boldsymbol{y} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}) f_{\boldsymbol{W},\boldsymbol{Y}|E_1,E_2}(\boldsymbol{w},\boldsymbol{y}) d\boldsymbol{w} d\boldsymbol{y}. \end{aligned} \tag{17}$$

Then the joint density $f_{\boldsymbol{W},\boldsymbol{Y}|E_1,E_2}(\boldsymbol{w},\boldsymbol{y})$ partitions into two components, whose latter component (a probability mass function) can be rewritten using Bayes rule. For convenience, denote $g(\boldsymbol{w}) = \mathbb{P}(E_1|\boldsymbol{W}=\boldsymbol{w}, E_2)$.

$$\begin{aligned} f_{\boldsymbol{W},\boldsymbol{Y}|E_1,E_2}(\boldsymbol{w},\boldsymbol{y}) d\boldsymbol{y} d\boldsymbol{w} &= f_{\boldsymbol{Y}|\boldsymbol{W}=\boldsymbol{w},E_1,E_2}(\boldsymbol{y}) \cdot f_{\boldsymbol{W}|E_1,E_2}(\boldsymbol{w})\ d\boldsymbol{y}\ d\boldsymbol{w} \\ &= f_{\boldsymbol{Y}|\boldsymbol{W}=\boldsymbol{w},E_1,E_2}(\boldsymbol{y}) \cdot \frac{\mathbb{P}(E_1|\boldsymbol{W}=\boldsymbol{w},E_2) f_{\boldsymbol{W}|E_2}(\boldsymbol{w})}{\mathbb{P}(E_1|E_2)}\ d\boldsymbol{y}\ d\boldsymbol{w} \\ &= f_{\boldsymbol{Y}|\boldsymbol{W}=\boldsymbol{w},E_1,E_2}(\boldsymbol{y}) \cdot \frac{g(\boldsymbol{w}) f_{\boldsymbol{W}}(\boldsymbol{w})}{\int g(\boldsymbol{w}') f_{\boldsymbol{W}}(\boldsymbol{w}') d\boldsymbol{w}'}\ d\boldsymbol{y}\ d\boldsymbol{w}, \end{aligned}$$

where we used the independence between W and E_2 in the last equality. With this, $T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v})$ from (17) becomes:

$$T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}) = \int \mathbb{1}(\boldsymbol{v}^T\boldsymbol{y} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}) \cdot g(\boldsymbol{w}) \cdot \frac{f_{\boldsymbol{W}|E_2}(\boldsymbol{w})}{\int g(\boldsymbol{w}') f_{\boldsymbol{W}}(\boldsymbol{w}') d\boldsymbol{w}'} \cdot f_{\boldsymbol{Y}|\boldsymbol{W}=\boldsymbol{w},E_1,E_2}(\boldsymbol{y})\ d\boldsymbol{y}\ d\boldsymbol{w}.$$

Now, rearranging, we get:

$$\begin{aligned} T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}) &= \int \underbrace{\left\{ \int \mathbb{1}(\boldsymbol{v}^T\boldsymbol{y} \geqslant \boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}) \cdot f_{\boldsymbol{Y}|\boldsymbol{W}=\boldsymbol{w},E_1,E_2}(\boldsymbol{y}) d\boldsymbol{y} \right\}}_{T(\boldsymbol{y}_{\text{obs}},\boldsymbol{v},\boldsymbol{w})} \underbrace{\frac{g(\boldsymbol{w})}{\int g(\boldsymbol{w}') f_{\boldsymbol{W}}(\boldsymbol{w}') d\boldsymbol{w}'}}_{a(\boldsymbol{w})} f_{\boldsymbol{W}}(\boldsymbol{w}) d\boldsymbol{w} \\ &= \int T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}, \boldsymbol{w}) a(\boldsymbol{w})\ f_{\boldsymbol{W}}(\boldsymbol{w})\ d\boldsymbol{w}. \end{aligned} \tag{18}$$

This proves the first equality in Proposition 3. To show what the weighting factor $a(\boldsymbol{w})$ equals, observe that by applying Bayes rule to the numerator of $a(\boldsymbol{w}_{\text{obs}})$, and rearranging:

$$\begin{aligned} a(\boldsymbol{w}) &= \frac{g(\boldsymbol{w})}{\int g(\boldsymbol{w}') f_W(\boldsymbol{w}')\ d\boldsymbol{w}'} = \frac{\mathbb{P}(E_1|E_2, \boldsymbol{W}=\boldsymbol{w})}{P(E_1|E_2)} = \frac{\mathbb{P}(\boldsymbol{W}=\boldsymbol{w}|E_1,E_2)}{\mathbb{P}(\boldsymbol{W}=\boldsymbol{w}|E_2)} \\ &= \frac{\mathbb{P}(\boldsymbol{W}=\boldsymbol{w}|E_1,E_2)}{\mathbb{P}(\boldsymbol{W}=\boldsymbol{w})}. \end{aligned}$$

Finally, to show the second equality in Proposition 3, observe that we can also represent

$a(\boldsymbol{w})$ as

$$a(\boldsymbol{w}) = \frac{g(\boldsymbol{w})}{\mathbb{E}[g(\boldsymbol{w})]} \tag{19}$$

by definition, where the denominator is the expectation taken with respect to the random variable $\boldsymbol{W}$. Leveraging the geometric theorems in works like Tibshirani et al. (2018), it can be shown that

$$g(\boldsymbol{w}) = P\Big(M(\boldsymbol{Y}+\boldsymbol{W}) = M(\boldsymbol{y}_{\text{obs}}+\boldsymbol{W}) \mid \Pi_{\boldsymbol{v}}^{\perp}\boldsymbol{Y} = \Pi_{\boldsymbol{v}}^{\perp}\boldsymbol{y}_{\text{obs}}\Big) = \Phi(\mathcal{V}_{\text{up}}/\tau) - \Phi(\mathcal{V}_{\text{lo}}/\tau). \tag{20}$$

Also from the same references as well as stated in Section 3.3, we know that

$$T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}, \boldsymbol{w}) = \frac{\Phi(\mathcal{V}_{\text{up}}/\tau) - \Phi(\boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}/\tau)}{\Phi(\mathcal{V}_{\text{up}}/\tau) - \Phi(\mathcal{V}_{\text{lo}}/\tau)} \tag{21}$$

Putting (19), (20) and (21) together into (18), we complete the proof by obtaining

$$T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}) = \frac{\int T(\boldsymbol{y}_{\text{obs}}, \boldsymbol{v}, \boldsymbol{w}) g(\boldsymbol{w}) f_{\boldsymbol{W}}(\boldsymbol{w}) d\boldsymbol{w}}{\int g(\boldsymbol{w}) f_{\boldsymbol{W}}(\boldsymbol{w}) d\boldsymbol{w}} = \frac{\int \Phi(\mathcal{V}_{\text{up}}/\tau) - \Phi(\boldsymbol{v}^T\boldsymbol{y}_{\text{obs}}/\tau) f_{\boldsymbol{W}}(\boldsymbol{w}) d\boldsymbol{w}}{\int \Phi(\mathcal{V}_{\text{up}}/\tau) - \Phi(\mathcal{V}_{\text{lo}}/\tau) f_{\boldsymbol{W}}(\boldsymbol{w}) d\boldsymbol{w}}.$$

References

Fithian, W., Taylor, J., Tibshirani, R., and Tibshirani, R. J. (2015). Selective sequential model selection. arXiv: 1512.02565.

Hyun, S., G'Sell, M., and Tibshirani, R. J. (2018). Exact post-selection inference for the generalized lasso path. *Electronic Journal of Statistics* pages 1053–1097.

Lin, K., Sharpnack, J. L., Rinaldo, A., and Tibshirani, R. J. (2017). A sharp error analysis for the fused lasso, with application to approximate changepoint screening. In *Advances in Neural Information Processing Systems*, pages 6884–6893.

Olshen, A., Seshan, V. E., Lucito, R., and Wigler, M. (2004). Circular binary segmentation for the analysis of array-based DNA copy number data. *Biostatistics* **5,** 557–572.

Snijders, A. M., Nowak, N., Segraves, R., Blackwood, S., Brown, N., Conroy, J., Hamilton, G., Hindle, A. K., Huey, B., Kimura, K., et al. (2001). Assembly of microarrays for genome-wide measurement of DNA copy number. *Nature genetics* **29,** 263–264.

Talevich, E., Shain, A. H., Botton, T., and Bastian, B. C. (2016). CNVkit: Genome-wide copy number detection and visualization from targeted DNA sequencing. *PLoS computational biology* **12,**.

Tibshirani, R. J., Rinaldo, A., Tibshirani, R., and Wasserman, L. (2018). Uniform asymptotic inference and the bootstrap after model selection. *Ann. Statist.* **46,** 1255–1287.

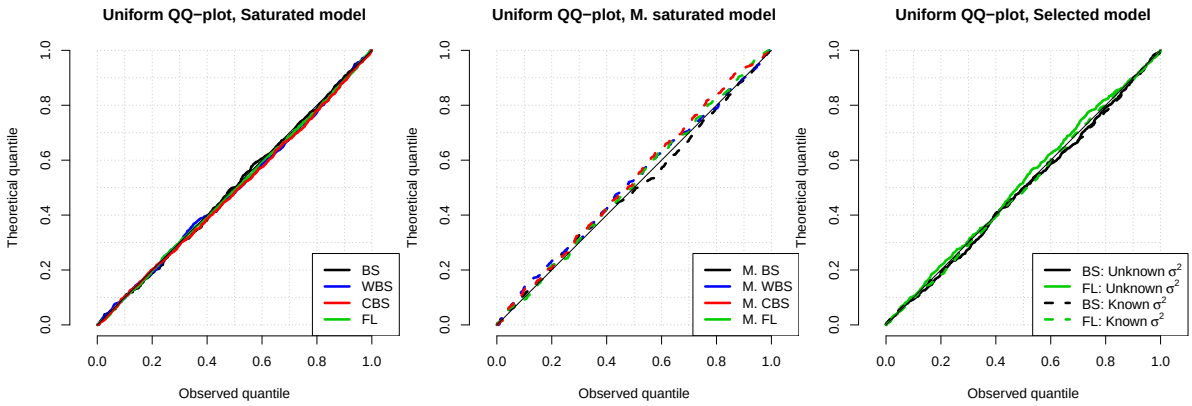

Figure 6: *All plots showing the p-values of various statistical inferences under the global null. (Left): Saturated model tests, specifically BS (black), WBS (blue), CBS (red) and FL (green). (Middle): Marginalized variants of the left plot. (Right): Selected model tests, specifically BS (black) and FL (green), either with unknown* σ^2 *(solid) or known* σ^2 *(dashed). Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

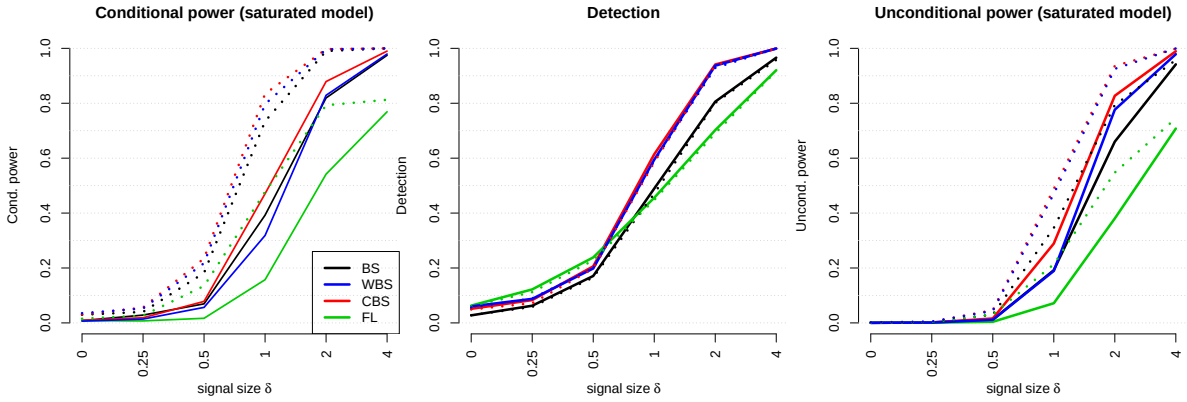

Figure 7: *Data was simulated from two settings over signal size* $\delta \in (0, 4)$ *with* $n = 200$ *data points. Several two-step algorithms (WBS, SBS, CBS, FL) were applied, and post-selection segment test inference was conducted on the resulting two detected changepoints from each method. The dotted lines are the marginalized versions of each test. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

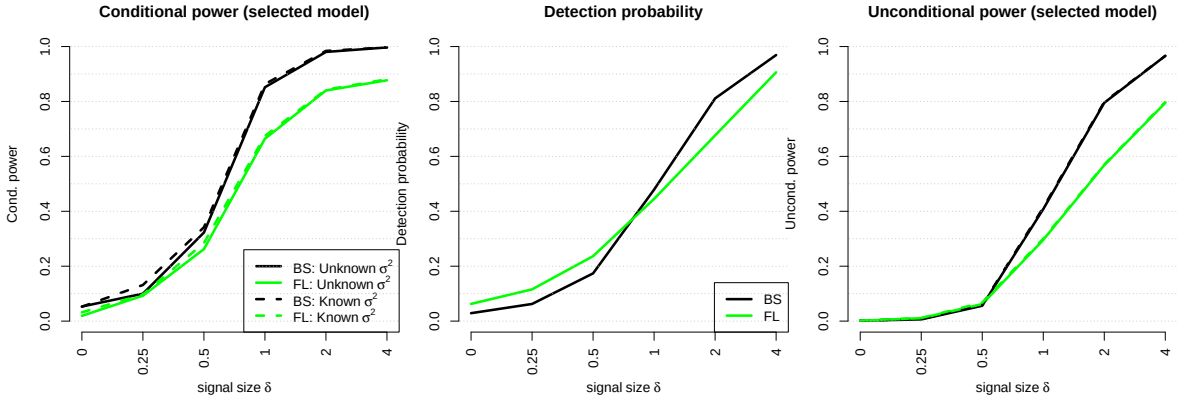

Figure 8: *Setup similar to Figure 8 but for selected model tests. Only BS (black) and FL (green) are shown. but the selected model test is applied to both known (dashed line) and unknown noise parameter σ^2 (solid line). Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

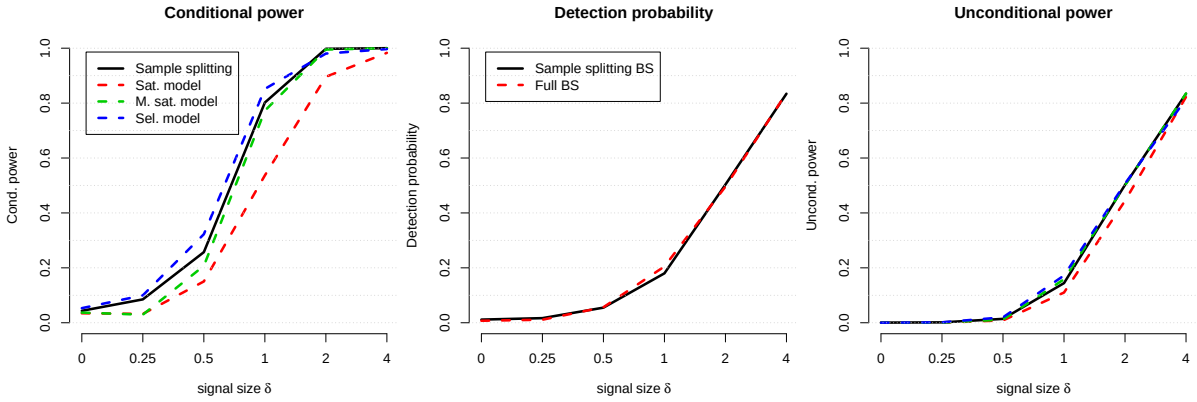

Figure 9: *The same setup as in Figure 2 but with exact detection. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

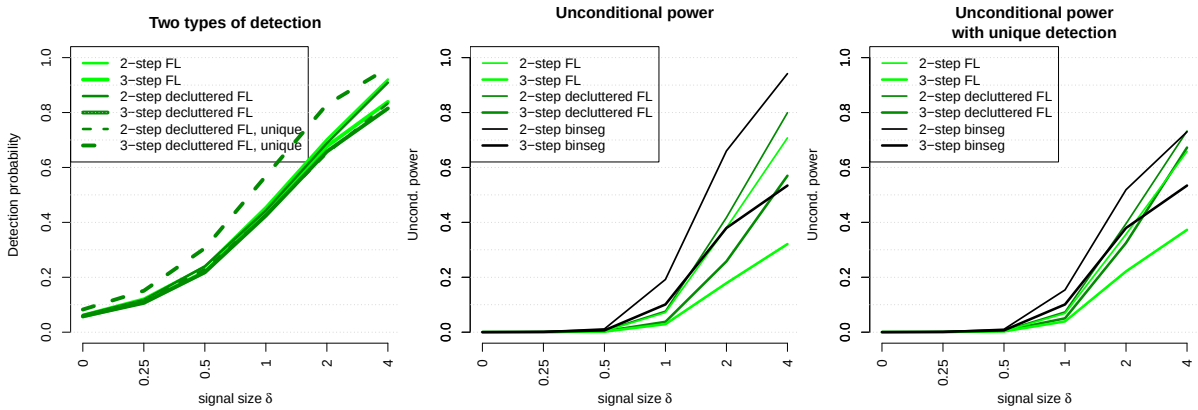

Figure 10: *(Left): Various detections for FL, either using 2 or 3 steps, and either using decluttering or not. (Middle): The unconditional power of various segmentation algorithms. (Right): The unconditional power, but defined as the conditional power multiplied by the unique detection probability. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

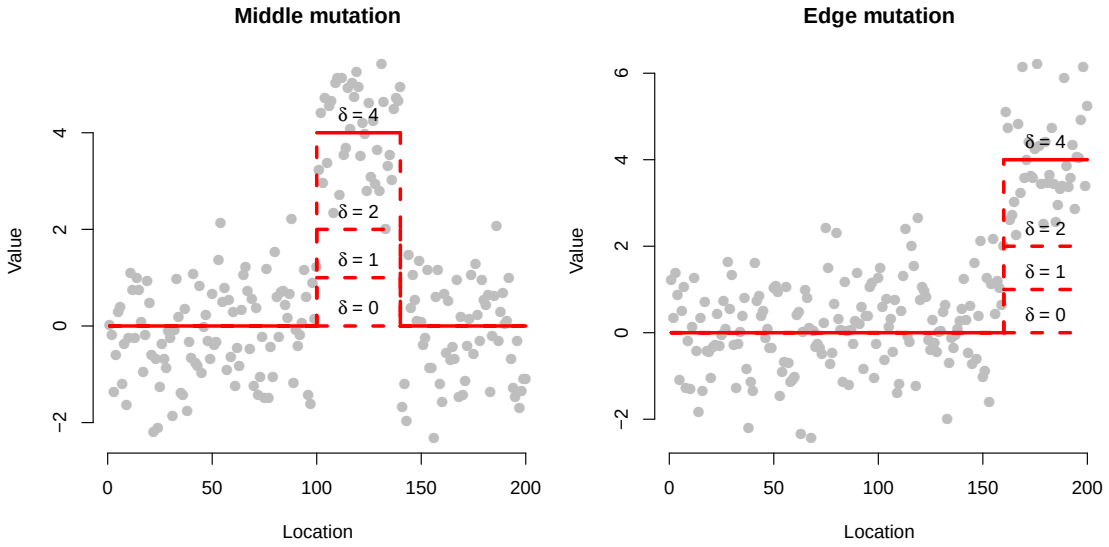

Figure 11: *(Left) Example of simulated Gaussian data for middle mutation as defined in* (9) *with* $\delta = 4$*, with data length* $n = 200$ *and noise level* $\sigma = 1$*. The possible mean vectors* θ *for* $\delta = 0, 1, 2$ *are also shown. (Right) Analogous to the left figure, but representing edge mutations defined in* (14)*. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

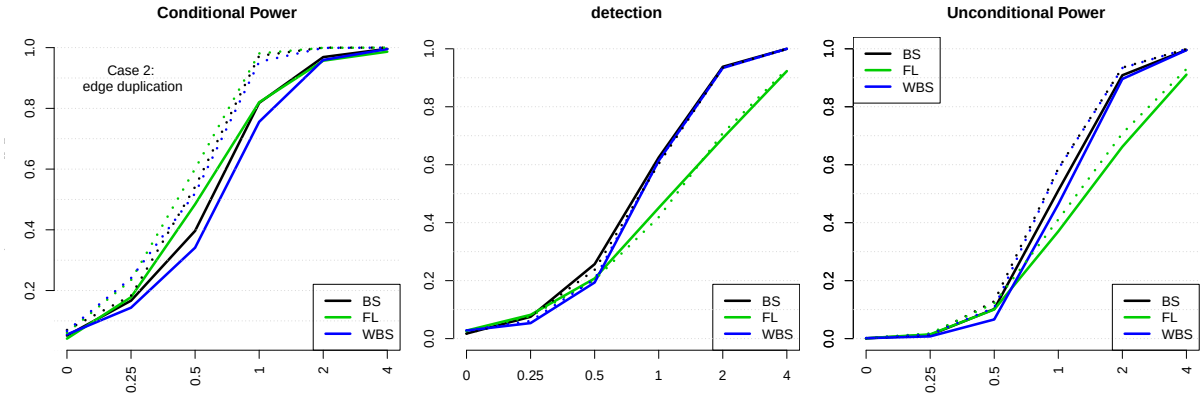

Figure 12: *Same setup as Figure 8 but for edge-mutation data. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*

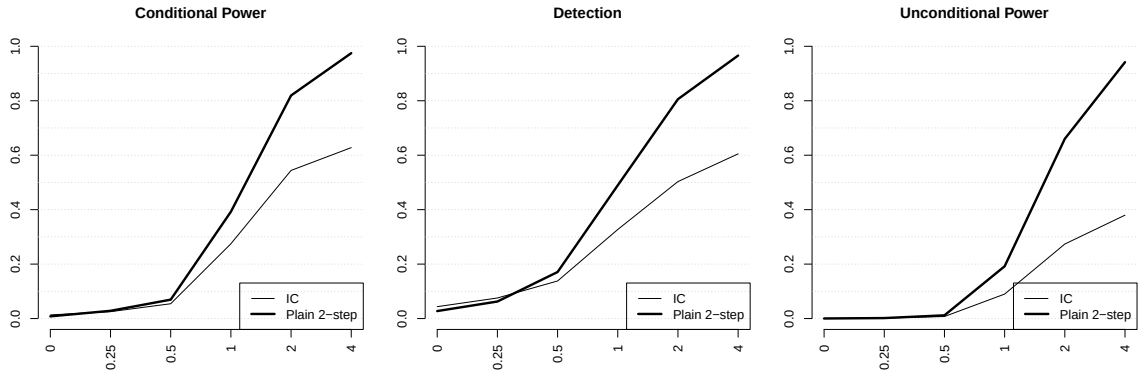

Figure 13: *Similar setup as Figure 8. In the middle-mutation data example from (9). IC-stopped BS inference (bold line) is compared to a fixed 2-step BS inference (thin line). We can see that the power and detection are considerably lower. The average number of steps taken per each* δ *on x-axis ticks are* $1.34, 1.86, 3.02, 3.64, 3.77, 3.72$*, respectively.*

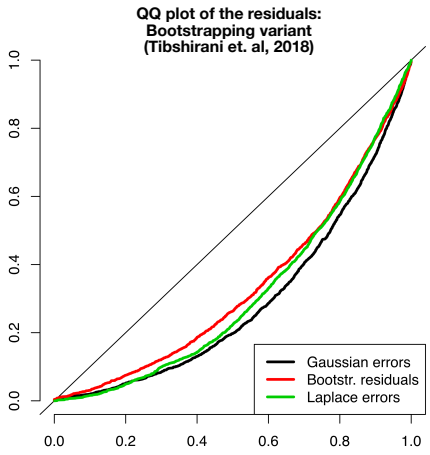

Figure 14: *QQ-plot of p-values derived from the bootstrap substitution method developed in Tibshirani et al. (2018). These results are presented in a similar fashion to Figure 4 C and D. Note: this figure appears in color in the electronic version of this article, and any mention of color refers to that version.*