bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Testing for dependence on tree structures

Merle Behr[a,], **M. Azim Ansari**[b], **Axel Munk**[c,d], **and Chris Holmes**[b,1]

[a]Department of Statistics, University of California at Berkeley; [b]Department of Statistics, and Wellcome Centre for Human Genetics, University of Oxford; [c]Institute for Mathematical Stochastics, University of Goettingen; [d]Max Planck Institute for Biophysical Chemistry, Goettingen

This manuscript was compiled on December 18, 2019

Tree structures, showing hierarchical relationships and the latent structures between samples, are ubiquitous in genomic and biomedical sciences. A common question in many studies is whether there is an association between a response variable measured on each sample and the latent group structure represented by some given tree. Currently this is addressed on an ad hoc basis, usually requiring the user to decide on an appropriate number of clusters to prune out of the tree to be tested against the response variable. Here we present a statistical method with statistical guarantees that tests for association between the response variable and a fixed tree structure across all levels of the tree hierarchy with high power, while accounting for the overall false positive error rate. This enhances the robustness and reproducibility of such findings.

Subgroup detection | Hypothesis testing | Tree structures | Change-point detection

In the era of big data where quantifying the relationship between samples is difficult, tree structures are commonly used to summarize and visualize the relationship between samples and to capture latent structure. The hierarchical nature of trees allows the relationships between all samples to be viewed in a single framework and this has lead to their widespread usage in genomics and biomedical science. Examples are phylogenetic trees built from genetic data, hierarchical clustering based on distance measures of features of interest (for example gene expression data with thousands of markers measured in each sample), evolution of human languages, and more broadly in machine learning where clustering and unsupervised learning is a fundamental task (1–7).

Often samples have additional response measurements y_i (e.g., phenotypes) and a common question is whether there is a relation between the sample's latent group structure captured by the tree T and the outcome of interest y_i, i.e., whether the distribution of y_i depends on its relative location among the leaves of the tree T. Testing for all possible combinations of groupings on the tree is practically impossible as it grows exponentially with sample size. Currently users typically decide on the number of clusters on an ad hoc basis (e.g., after plotting the response measurement on the leaves of the tree and deciding visually which clusters to choose). These clusters are then tested for association with the outcome of interest. This lack of rigorous statistical methodology has limited the translational application and reproducibility of these methods.

Here, we present a statistical method and accompanying R package, `treeSeg`, that given a significance level α, tests for dependence of the response measurement distribution on all levels of hierarchy in a given tree while accounting for multiple testing. It returns the most likely segmentation of the tree such that each segment has a distinct response distribution, while controlling the overall false positive error. This is achieved by embedding the tree segmentation problem into a change-point detection setting (8–13).

`treeSeg` does not require any assumptions on the generation process of the tree T. It treats T as given and fixed, testing the response of interest against the given tree structure. Every tree T, independent of how it was generated, induces some latent ordering of the samples. `treeSeg` tests whether for this particular ordering, the distribution of the independent observations y_i depends on their locations on the tree.

`treeSeg` is applicable to a wide-range of problems across many scientific disciplines where the association between a tree structure and a response variable is under investigation, such as phylogenetic studies, molecular epidemiology, metagenomics, gene expression studies, etc. (3, 5, 14–16). The only inputs needed are the tree structure T and the outcome of interest y_i for the leaves of the tree. We demonstrate the sensitivity and specificity of `treeSeg` using simulated data and its application to a cancer gene expression study (1).

1. Results

For ease of presentation we restrict to discrete binary response measurements $y_i \in \{0, 1\}$. However, the procedure is equally applicable to continuous and other observation types (see Methods). If there is no association between the tree T and the response measurement y_i, then the observed responses y_i would be randomly distributed on the leaves of the tree, independent of the tree structure T. However, if the distribution of responses is associated with the tree structure, we may observe clades in the tree with distinct response distributions. The power to detect segments with distinct distributions depends on the size of the clade and the change in response probability, $p_i = \mathbf{P}(Y_i = 1) = 1 - \mathbf{P}(Y_i = 0)$, which means one can only make statistical statements on the *minimum* number of clades

Significance Statement

Tree like structures are abundant in the empirical sciences as they can summarize high dimensional data and show latent structure among many samples in a single framework. Prominent examples include phylogenetic trees or hierarchical clustering derived from genetic data. Currently users employ ad hoc methods to test for association between a given tree and a response variable, which reduces reproducibility and robustness. In this paper, we introduce treeSeg, a simple to use and widely applicable methodology with high power for testing between all levels of hierarchy for a given tree and the response while accounting for the overall false positive rate. Our method allows for precise uncertainty quantification and therefore increases interpretability and reproducibility of such studies across many fields of science.

The authors declare that they have no competing interests.

[1]Correspondence and requests for materials should be addressed to Chris Holmes (email: holmes@stats.ox.ac.uk).

DRAFT

bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

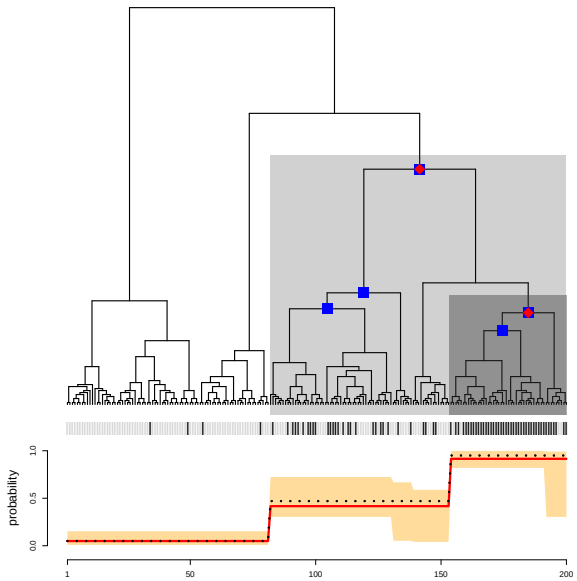

Fig. 1. Illustration of the `treeSeg` method. Binary tree with 200 leaves and three segments with distinct response distributions indicated by dark gray, light gray and white backgrounds. Outcomes for each sample is shown on the leaves of tree as gray or black vertical lines. Leaf responses were simulated such that the black line has probability of $0.95, 0.47$ and 0.05 for each of the dark gray, light gray and white background sections respectively. Using $\alpha = 0.1$, `treeSeg` has estimated three segments on the tree with distinct response distributions indicated by the red dots on the nodes of the tree. Blue dots constitute a 90% confidence set for the nodes of the tree associated with the change in response distribution. The bottom panel shows the simulation response probabilities (black dotted line) and the `treeSeg` estimate (red line) and its 90% confidence bands in orange color.

with distinct distributions on the tree and not the *maximum*.

Figure 1 illustrates our method and its output for a simulated dataset. The responses y_i are displayed on the leaves of the tree as black and gray lines. The tree T is made of three segments with distinct distributions over the responses indicated by dark gray, light gray, and white backgrounds. Given a confidence level $1-\alpha$ (e.g., $1-\alpha = 0.9, 0.95$), the `treeSeg` procedure estimates the most likely segmentation of the tree into regions of common response distributions, such that the true number of segments is at least as high as the estimated number of segments with probability of $1-\alpha$.

Our method employs many likelihood ratio statistics simultaneously to test for changes in the response distribution on all levels of tree hierarchy and estimates at what level, if any, there is a change. The multiple testing procedure of `treeSeg` is based on a multiscale change-point methodology (11), tailored to the tree structure. The significance levels of the individual tests are chosen in such a way that the overall significance level is the pre-specified α. Theoretical proofs are provided in the Methods section and the online supplementary information. As well as the maximum likelihood estimate, our method also provides confidence sets (at $1-\alpha$ level) for the nodes of the tree associated with the change in response distribution and a confidence band for the response probabilities p_i over the segments.

In the example of Figure 1, using $\alpha = 0.1$, `treeSeg` estimates three segments in the tree T, indicated with red dots on the nodes of the tree, recovering the true simulated changes in response distributions. In Figure 1, blue dots on the tree indicate the $1-\alpha$ confidence set for the nodes on the tree associated with the change in responses y_i. The red line, in the bottom plot, shows the maximum likelihood estimate of the response probabilities p_i for each segment, which accurately recovers the true simulated probabilities shown as black dotted line. The orange band shows the $1-\alpha$ confidence band of the response probabilities.

The `treeSeg` method can handle missing data and to make response predictions on new samples. Computationally, `treeSeg` scales well with sample size, for example a test simulation for a tree with 100,000 samples (number of leaves in the tree) and no response association took around 110 minutes to run on a standard laptop. For details on `treeSeg`'s implementation see online supplementary text.

A. Simulation Study. We confirmed the statistical calibration and robustness of `treeSeg` using simulation studies. We found that for reasonable minimal clade sizes and changes in response distribution, `treeSeg` is able to detect association between response and tree structure reliably (Supplementary Figures S11 - S16). More importantly, `treeSeg` almost never detects segments which are not present (it can be mathematically proven that the inclusion of a false positive segment only happens with probability $\leq \alpha$, see Methods) and the nominal guarantee of $1-\alpha$ is even exceeded in most cases (see online supplementary information). For example, in 1,000 randomly generated trees (200 leaves) with no changes in response distribution, `treeSeg` (using $\alpha = 0.1$) correctly assigns no association between response and tree structure in 98.6% of the runs.

The `treeSeg` algorithm incorporates a fixed ordering of the leaf nodes according to the tree structure T. In principle, any such ordering is equally valid, as long as this is made a-priori, independent of the response variables. In our online implementation we apply a standardized pre-ordering of the nodes, so that `treeSeg`'s output is independent of any user specified ordering. Furthermore, we provide simulation results showing that `treeSeg` is very robust to random changes in node ordering and consistently infers the correct number of segments on the tree (Supplementary Figures S17 - S25). In Section E in the online supplement we also discuss an approach based on aggregating results among several random leaf orderings for the given tree. For more details on simulation studies refer to the online supplementary information.

B. Application to Cancer data example. We illustrate the application of the method on a breast cancer gene expression study (1) where data are publicly available. Following the original study, we used correlation of gene expression data as a distance measure between samples to build a hierarchical clustering tree. In the original study, based on visual inspection, the authors divided the samples into two clusters, observing differences in the distributions of various clinical responses between the two clusters.

In contrast, `treeSeg` only requires a significance level α as input and searches for associations between responses and the tree on all levels of hierarchy, while accounting for multiple testing. Our results are shown in Figure 2. Using an $\alpha = 0.05$, for one of the responses, `treeSeg` delinated the tree into two clusters with distinct response distribution as in the original study. However, `treeSeg` reports different patterns of association between the tree and the other five responses, including one, which has no association with tree structure.

DRAFT

bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

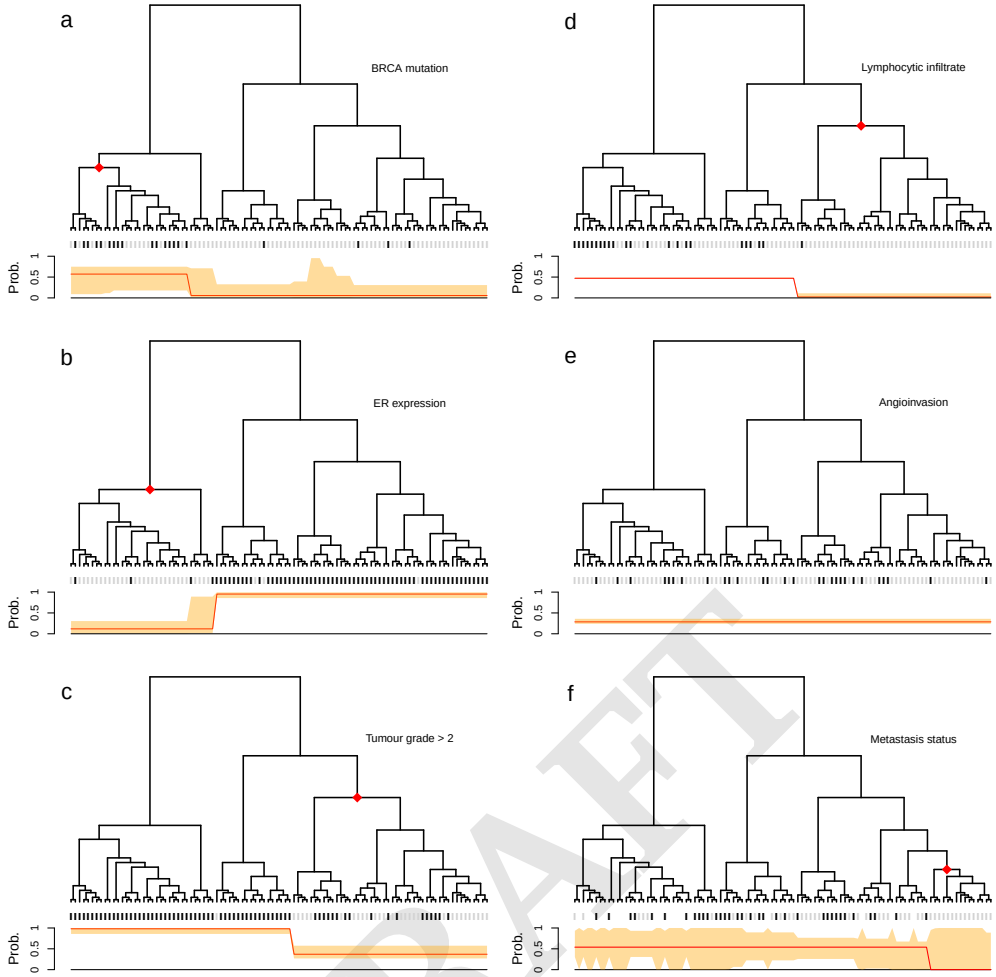

Fig. 2. Application of `treeSeg` to a cancer gene expression study (1). Gene expressions for 98 breast cancer samples were clustered based on correlation between samples. Six clinical responses were collected for the samples, BRCA mutation, oestrogen receptor (ER) expression, histological grade, lymphocytic infiltration, angioinvasion and development of distant metastasis within 5 years which are represented in panels (a-f) respectively. In each panel, the `treeSeg` estimation (at $\alpha = 0.05$) for clades with distinct response distribution and their probabilities are indicated by the red dots on the tree and the red line below the tree. The orange band shows the 95% confidence band for the response probabilities p_i for the estimated segments. In panel (e) there is no association between the tree and the response (angioinvasion) and in panel (f) some of the samples have missing observations for the response (distant metastasis within 5 years).

DRAFT

`treeSeg`'s application is not restricted to trees generated using hierarchical clustering. For an application to a maximum likelihood phylogenetic tree generated from pathogen sequence data see Section 3 in the online supplementary material. Indeed, `treeSeg` is completely independent of the particular way the tree was generated, with all results holding true conditioned on the specific tree T under consideration (see Section F in the online supplementary materials).

2. Discussion

The only tuning parameter for the `treeSeg` method is a significance level α. Depending on the application, the user can decide which value of α is appropriate, or screen through several values of α, e.g., $\alpha = 0.01, 0.05, 0.1, 0.5$. A small α gives a higher confidence that all detected associations are, indeed, present in the data (with probability of at least $1 - \alpha$). A larger α allows to detect more clusters, but increases the risk of including false positive clusters.

The confidence statement for detected clades and response probabilities that accompany `treeSeg`'s segmentation, account for multiple testing at the level of $1 - \alpha$. This allows for precise uncertainty quantification when detecting associations between tree structure and the responses. We highlighted `treeSeg`'s potential with an example from a gene expression study, but note its ubiquitous applicability in various settings. Our method `treeSeg` is implemented as an R package, available on GitHub (https://github.com/merlebehr/treeSeg) and accompanied by a detailed Jupyter notebook with reproductions of all figures in this text, and has the potential to be used across many fields of science.

3. Methods

A. Model assumptions. For illustration purposes, we focus on binary traits $Y_i \in \{0, 1\}$. Please see online supplementary text Section B on how `treeSeg` generalizes for arbitrary continuous and discrete data. We assume a fixed given rooted tree T with n leaves, that captures some neighborhood structure of interest. For the n samples (the leaves of the tree) independent

bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

(a) Off(i) **(b)** $T(i)$

Fig. 3. Illustration of notation used in the paper. The respective subset of leaf nodes Off(i) and sub-tree $T(i)$ are shown in red.

binary traits Y_i, $i = 1, \ldots, n$, with success probability p_i, are observed, that is,

$$Y_i \sim \mathrm{Bern}(p_i) \quad \Leftrightarrow \quad \mathbf{P}(Y_i = 1) = 1 - \mathbf{P}(Y_i = 0) = p_i, \quad [1]$$

independently for $i = 1, \ldots, n$, where Bern(p) denotes a Bernoulli distribution with success probability p. The aim is to estimate the underlying success probabilities $p_1, \ldots, p_n$ from the observations Y_i. Without any additional structural information on the success probabilities we cannot do better than estimating $p_i = Y_i$. However, taking into account the tree structure, we can assume that the success probabilities are associated with the tree such that samples on the same clade of the tree may have the same success probabilities. Our methodology is based on a testing problem, where the null model assumes that all isolates have the same success probability, say p_0, and the alternative model assumes that some of the clades on the tree have different success probabilities ($p_0 + c \in [0, 1]$). In the following, we denote an internal node (which demarcates a clade on the tree) with a distinct success probability as an *active node*.

For simplicity, we will assume in the following that the tree T is binary. Extensions to arbitrary trees are straight forward. We use the following notation. For a binary, rooted tree $T = (V, E)$ we assume vertices $V = \{1, \ldots, n, n+1, \ldots, 2n-1\}$ and edges $E = \{(i, j) \ : \ i, j \in V \text{ with } i, j \text{ connected}\}$. The leaves are labeled $V_L = \{1, \ldots, n\}$, the inner nodes are labeled $V_I = \{n + 1, \ldots, 2n - 1\}$, and the root is labeled $2n - 1$. For a node $i \in V$ its set of offspring leaves in V_L is denoted as Off(i). For a node $i \in V$ the subtree of T with root i is denoted as $T(i)$. An illustrative example for this notation is shown in Figure 3. Moreover, for an inner node $i \in V_I$ with offspring leaves $\mathrm{Off}(i) = \{i_1, \ldots, i_m\} \subset \{1, \ldots, n\}$ and for some $\epsilon \in (0, 1)$, we denote the left ϵ-leaf-neighborhood of i as $\mathrm{N_L}(i, \epsilon) = \{i_1 - \lfloor n\epsilon \rfloor, i_1 - \lfloor n\epsilon \rfloor + 1, \ldots, i_1 + \lfloor n\epsilon \rfloor - 1, i_1 + \lfloor n\epsilon \rfloor\}$ and, analog, the right ϵ-leaf-neighborhood of i as $\mathrm{N_R}(i, \epsilon) = \{i_m - \lfloor n\epsilon \rfloor, i_m - \lfloor n\epsilon \rfloor + 1, \ldots, i_m + \lfloor n\epsilon \rfloor - 1, i_m + \lfloor n\epsilon \rfloor\}$.

We consider the following statistical model.

Model 1. For a given binary, rooted tree $T = (V, E)$ as above, assume one observes for each of the leaves $i \in V_L$ independent Bernoulli random variables $Y_i \sim B(p_i)$, $1 \le i \le n$, where the vector of success probabilities $p = (p_1, \ldots, p_n)$ is an element of

$$\mathcal{S} = \mathcal{S}(T) := \Bigg\{ \left(p_0 + \sum_{j=1}^{k} c_j \mathbb{1}_{i \in \mathrm{Off}(v_j)} \right)_{1 \le i \le n} \in [0, 1]^n : \ v_j \in V, \ p_0, c_j \in \mathbb{R}, \ 0 \le k \le 2n - 1 \Bigg\}.$$

For an element $p \in \mathcal{S}$ we denote the set of nodes $V(p) := \{v_1, \ldots, v_k\}$ as a set of *active nodes* and $k(p) = k$ as the number of active nodes. To ensure identifiability of active nodes, we further assume that for each active node v_j, $j = 1, \ldots, k$, there exists at least one offspring leaf $i \in 1, \ldots, n$ that has the same success probability as v_j. This just excludes the trivial case where the influence of one active node (or the root) is completely masked by other active nodes. Equivalently, this means that we assume $\#\mathrm{supp}(p) = \#\{p_i \ : \ i = 1, \ldots, n\} = k + 1$. We provide a simple example in the supplementary material Figure S27.

We stress that the set $V(p)$ is not necessarily unique (see Figure S26 in the supplement for an example). That is, for a given vector $p \in \mathcal{S}$ there may exists two (or more) sets of active nodes $\{v_1, \ldots, v_k\}$ and $\{v'_1, \ldots, v'_k\}$ such that

$$p_i = p_0 + \sum_{j=1}^{k} c_j \mathbb{1}_{i \in \mathrm{Off}(v_j)} = p'_0 + \sum_{j=1}^{k} c'_j \mathbb{1}_{i \in \mathrm{Off}(n'_j)}.$$

To overcome this ambiguity we will implicitly associate with each $p \in \mathcal{S}(T)$ a set of active nodes $V(p)$ of size $k(p)$. When a specific vector $p \in \mathcal{S}(T)$ has more than one possible sets of active nodes $V', V'', \ldots$ we assume several copies of p in $\mathcal{S}(T)$, one associated with each of the sets $V', V'', \ldots$. In the following, we explore the tree structure to estimate the underlying success probabilities p_i and hence, their segmentation into groups of leaves where observations within a group have the same success probability and observations between different groups have different success probabilities.

B. Multiscale segmentation. The procedure we propose extends (11) from totally ordered structures to trees and is a hybrid method of estimating and testing. A fundamental observation is that one can never rule out an additional active node. This is because a node could be active but changed the success probability of its offspring nodes only by an arbitrarily small amount. On the other hand, if in a subtree successes are much more common than in the remaining tree, it is possible to significantly reject the hypothesis that all leaves in the tree have the same success probability.

For a given candidate vector $\tilde{p} \in \mathcal{S}$ our procedure employs on each subtree $T(i)$ where $\tilde{p}$ is constant, with $\tilde{p} \equiv \tilde{p}(\mathrm{Off}(i))$, a likelihood ratio (LR) test for the hypothesis that the corresponding observations all have the same success probability $\tilde{p}(\mathrm{Off}(i))$. The levels of the individual tests are chosen in such a way that the overall level of the multiple test is α, for a given pre-specified $\alpha \in (0, 1)$. A statistical hypothesis test can always be inverted into a confidence statement and vice versa. Therefore, we can derive from the above procedure a confidence set for the vector of success probabilities $p = (p_1, \ldots, p_n)$. We require our final estimate $\hat{p} = \hat{p}_{1-\alpha}$ to lie in this confidence set. That is, we require whenever $\hat{p}$ has constant success probability on a subtree, the respective LR test accepts. Within all vectors $p \in \mathcal{S}$ which lie in this confidence set we choose one which comes from a minimum number of active nodes and within this set, we choose the maximum likelihood solution. Thereby, our procedure does not only provide an estimate, but also confidence statement for all quantities (11). More precisely, the following asymptotic confidence statements hold true.

1. With probability at least $1 - \alpha$ the true underlying signal $p \in \mathcal{S}$ originates from at least $\hat{k}$ active nodes, where $\hat{k}$ is the number of active nodes of $\hat{p}$, see Theorem 1.

DRAFT

bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

2. `treeSeg` yields a set of nodes $\mathcal{C}_{1-\alpha}$, such that the active nodes of p, $V(p)$, are contained with probability at least $1-\alpha$ in $\mathcal{C}_{1-\alpha}$, see Corollary 4.

3. `treeSeg` yields a confidence band for the underlying signal p, denoted as $\underline{p}_{1-\alpha}$ and $\overline{p}_{1-\alpha}$, such that with probability at least $1-\alpha$ it holds that $\underline{p}_{1-\alpha} \leq p \leq \overline{p}_{1-\alpha}$ simultaneously for all $i = 1, \ldots, n$, see Theorem 3.

Moreover, the coverage probability of the confidence sets allows to derive (up to log factors) optimal convergence rates of the `treeSeg` estimator as the sample size n increases (11). In particular, we show the following.

1. For fixed overestimation bound $\alpha \in (0,1)$, the probability that `treeSeg` underestimates the number of active nodes k vanishes exponentially as n increases, see Theorem 2.

2. The localization of the estimated active nodes is optimal up to a leaf node set of order $\log(n)$, see Theorem 5.

In the following we will give details of the method and of the statements 1–5. The proofs of Theorem 1, 2, 3, 5 and Corollary 4 are similar to the totally structured setting (11). Necessary modifications are outlined in the online supplementary text.

For an arbitrary given test vector $\tilde{p}$ (which may depend on Y) we define the multiscale statistic (11, 17–19)

$$T_n(\tilde{p}, Y) = \max_{\substack{1 \leq i \leq j \leq n \\ \tilde{p}|_{[i,j]} \text{ const.}}} \sqrt{2T_i^j(Y_i^j, \tilde{p}_i)} - pen(j-i+1), \quad [2]$$

where $Y_i^j = (Y_i, ..., Y_j)$ and $pen(x) := \sqrt{2\log(e/x)}$. Here, T_i^j is the local log-LR test statistic (20) for the testing problem

$$H: \; p_i = \ldots = p_j = \tilde{p}|_{[i,j]} \quad \text{vs.} \quad K: \; p_i = \ldots = p_j \neq \tilde{p}|_{[i,j]}.$$

The calibration term $pen(\cdot)$ serves as a balancing of the different scales in a way that the maximum in [2] is equally likely attained on all scales (11, 17) and guarantees certain optimality properties of the statistic [2] (17). Assuming a minimal segment scale $\lambda \in (0,1)$ of the underlying success probability vector p, that is,

$$\mathcal{S}_\lambda = \{p \in \mathcal{S} \text{ with const. segments' length at least } n\lambda\}, \quad [3]$$

it can be shown that $T_n(p, Y)$ converges in distribution to a functional of the Brownian motion (11) which is stochastically bounded by

$$M := \sup_{0 \leq s < t \leq 1} \left(\frac{B(t) - B(s)}{\sqrt{t-s}} - \sqrt{2\log\frac{e}{t-s}} \right). \quad [4]$$

Thereby, the minimal scale λ may depend on n such that $n\lambda/log(n)^3 \to \infty$ as $n \to \infty$, see (11). As the distribution of M does not depend on the true underlying signal p, its quantiles can be obtained by Monte Carlo simulations and are, in the following denoted as $q_{1-\alpha}$, that is,

$$\lim_{n\to\infty} \sup_{p \in \mathcal{S}_\lambda} \mathbf{P}(T_n(p, Y) > q_{1-\alpha}) \leq \mathbf{P}(M > q_{1-\alpha}) = \alpha. \quad [5]$$

For a given confidence level $\alpha \in (0,1)$ or, equivalently, a threshold value $q = q_{1-\alpha}$ in [5], we first define an estimator for the number of active nodes k in Model 1 via

$$\hat{k}(q) := \min_{p \in \mathcal{S}} k(p) \quad \text{s.t. } T_n(p, Y) \leq q. \quad [6]$$

Once the number of active nodes k is estimated, we estimate p as the constrained maximum likelihood estimator

$$\hat{p}(q) := \text{argmax}_{p \in \mathcal{H}(q)} \sum_{i=1}^n l_{p_i}(Y_i), \quad [7]$$

where $l_p(y)$ is the log-likelihood function of the binomial distribution and

$$\mathcal{H}(q) := \{p \in \mathcal{S} : k(p) = \hat{k}(q) \text{ and } T_n(p, Y) \leq q\}. \quad [8]$$

Note that the maximum likelihood (ML) solution in [7] is not necessarily unique. On the one hand, this is due to the non-uniqueness of the active nodes. On the other hand, this might happen with positive probability by the discreteness of the Bernoulli observations Y. In that case, `treeSeg` just reports the first available solution, with all other equivalent solutions listed in the confidence set. Clearly, if we choose q as in [5] for some given confidence level $\alpha \in (0,1)$ the estimator $\hat{k}(q_{1-\alpha})$ asymptotically controls the probability to overestimate the number of active nodes, as summarized in the following theorem.

Theorem 1. For fixed minimal scale $\lambda > 0$ and significance level $1-\alpha \in (0,1)$, let $\mathcal{S}_\lambda$ be as in [3], $q_{1-\alpha}$ as in [5], and $\hat{k}(q)$ `treeSeg`'s estimated number of active nodes in [6]. Then it holds that

$$\lim_{n\to\infty} \sup_{p \in \mathcal{S}_\lambda} \mathbf{P}(\hat{k}(q_{1-\alpha}) > k(p)) \leq \alpha. \quad [9]$$

We stress that in Theorem 1 it is possible to let λ go to zero as n increases (11), recall the paragraph after equation [4]. In particular, from the construction of $\hat{k}$ in [6] it follows that $T_n(p, Y) \leq q$ implies $\hat{k}(q) \leq k$ and thus, for the set $\mathcal{H}(q_{1-\alpha})$ in [8] one obtains that

$$\mathbf{P}\left(p \in \mathcal{H}(q_{1-\alpha})\right) \geq \mathbf{P}\left(T_n(p, Y) \leq q_{1-\alpha}\right) - \mathbf{P}\left(\hat{k}(q_{1-\alpha}) < k\right). \quad [10]$$

By [5] it follows that the first term on the right hand side, $\mathbf{P}\left(T_n(p, Y) \leq q_{1-\alpha}\right)$, is asymptotically lower bounded by $1-\alpha$. Moreover, as we show in the following Theorem 2, the underestimation error $\mathbf{P}\left(\hat{k}(q_{1-\alpha}) < k\right)$ vanishes exponentially fast as sample size n increases. From this it follows that the set $\mathcal{H}(q_{1-\alpha})$ constituents an asymptotically honest confidence set (11) for the whole vector p from which confidence sets as in statements 2 and 3 follow, see Theorem 3 and Corollary 4.

Any bound on the underestimation error necessarily must depend on the minimal segment scale λ in [3], as well as a minimal pairwise difference $\delta \in (0,1)$ of success probabilities in different active segments. That is, for $p \in \mathcal{S}_\delta$ we assume $\delta < \min_{p_i \neq p_j} |p_i - p_j|$ and let

$$\mathcal{S}_{\delta,\lambda} = \mathcal{S}_\lambda \cap \mathcal{S}_\delta. \quad [11]$$

With this, one obtains that the underestimation probability decreases exponentially (11) in n (for fixed δ, λ and significance level $1-\alpha \in (0,1)$) as the following theorem shows.

Theorem 2. For fixed minimal scale $\lambda > 0$, minimal success probability difference $\delta > 0$, and significance level $1-\alpha \in (0,1)$, let $\mathcal{S}_{\lambda,\delta}$ be as in [11], $q_{1-\alpha}$ as in [5], and $\hat{k}(q)$ `treeSeg`'s estimated number of active nodes in [6]. Then it holds that

$$\sup_{p \in \mathcal{S}_{\lambda,\delta}} \mathbf{P}(\hat{k}(q_{1-\alpha}) < k(p)) \leq C_1\, e^{-C_2\, n}, \quad [12]$$

DRAFT

bioRxiv preprint doi: https://doi.org/10.1101/622811; this version posted January 15, 2020. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

where C_1 and C_2 are positive constants, which only depend on α, λ, δ.

Again, it is possible to let α, λ, and δ go to zero as the sample size n increases (11). The proof of Theorem 2 is similar as for totally ordered structures (11). We outline necessary modifications in the online supplementary text. From Theorem 2 and [10] we directly obtain that $\mathcal{H}(q_{1-\alpha})$, indeed, constitutes a $1-\alpha$ asymptotic confidence set for the segmentation p.

Theorem 3. For fixed minimal scale $\lambda > 0$, minimal success probability difference $\delta > 0$, and significance level $1-\alpha \in (0,1)$, let $\mathcal{S}_{\lambda,\delta}$ be as in [11], $q_{1-\alpha}$ as in [5], and $\mathcal{H}(q_{1-\alpha})$ as in [8], then

$$\lim_{n\to\infty} \sup_{p \in \mathcal{S}_{\lambda,\delta}} \mathbf{P}\left(p \in \mathcal{H}(q_{1-\alpha})\right) \geq 1-\alpha.$$

As a corollary, we also obtain a confidence set of the active nodes.

Corollary 4. For fixed minimal scale $\lambda > 0$, minimal success probability difference $\delta > 0$, and significance level $1-\alpha \in (0,1)$, let $\mathcal{S}_{\lambda,\delta}$ be as in [11], $q_{1-\alpha}$ as in [5], and $\mathcal{H}(q_{1-\alpha})$ as in [8], then

$$\lim_{n\to\infty} \sup_{p \in \mathcal{S}_{\lambda,\delta}} \mathbf{P}\left(V(p) \subset \{v \in V(\tilde{p}) \ : \ \tilde{p} \in \mathcal{H}(q_{1-\alpha})\}\right) \geq 1-\alpha.$$

Theorem 1 and 2 reveal `treeSeg`'s ability to accurately estimate the number of active nodes in Model 1. For any (arbitrarily small) $\alpha \in (0,1)$ we can control the overestimation probability by $1-\alpha$, see Theorem 1. Simultaneously, as the sample size n increases, the underestimation error probability vanishes exponentially fast, see Theorem 2. The next theorem shows that `treeSeg` does not just estimate the number of active nodes correctly with high probability, but that it also estimates the location of those active nodes with high accuracy. To this end, note that for any active node $v \in V(p)$ and any $\epsilon \geq 1/n$ the leaf nodes of its left ϵ-leaf-neighborhood $\mathrm{N_L}(v,\epsilon)$ have non-constant success probability. The same is true for the right ϵ-leaf-neighborhood $\mathrm{N_R}(v,\epsilon)$. Now assume that `treeSeg` estimates the number of active nodes correctly $\hat{k} = k$, which is the case with high probability by Theorem 1 and 2. Then, $\hat{p}$ being non-constant on both, $\mathrm{N_R}(v,1/n)$ and $\mathrm{N_L}(v,1/n)$, for any true active nodes $v \in V(p)$, implies a perfect segmentation. Thus, the following theorem shows that `treeSeg`, indeed, yields such a perfect segmentation up to a leaf node set of size $\mathcal{O}(\log(n))$. That is, conditioned on the correct model dimension $\hat{k} = k$, `treeSeg`'s segmentation is perfect up to at most an order of $\log(n)$ miss-classified leaf nodes.

Theorem 5. For fixed minimal scale $\lambda > 0$, minimal success probability difference $\delta > 0$, and significance level $1-\alpha \in (0,1)$, for any $p \in \mathcal{S}_{\lambda,\delta}$ it holds true that

$$\hat{p}|_{\mathrm{N_L}\left(v, \frac{C_3 \log(n)}{n}\right)} \text{ and } \hat{p}|_{\mathrm{N_R}\left(v, \frac{C_3 \log(n)}{n}\right)}$$

are not constant, for all $v \in V(p)$, with probability at least $1 - C_1\, e^{-C_2\, n}$, where C_1, C_2, C_3 are positive constants which only depend on α, λ, δ.

Theorem 5 follows directly by translating change-point location estimation accuracy results for the totally ordered case (11) to the tree setting.

A natural question is whether the localization rate in Theorem 5 is optimal. In particular, one can compare this result to the totally ordered setting, where the minimax optimal change-point estimation rate is known to be of the same order (possibly up to $\log(n)$-factors). One would expect that the additional tree structure leads to a strictly better segmentation rate. It turns out, however, that, without making further assumptions on the particular tree, the rate in Theorem 5 cannot be improved in general, see Theorem 6 in the supplement. On the other hand, when one imposes additional structural assumptions on the tree, for example, for perfect trees, it can be shown that `treeSeg` yields a perfect segmentation with high probability, see Theorem 7 in the supplement. Thus, `treeSeg` efficiently leverages the tree structure to overcome the minimax lower bound from a simple change-point estimation problem, whenever the tree allows this. We provide more details in Section D in the supplement.

ACKNOWLEDGMENTS. MB was supported by DFG postdoctoral fellowship BE 6805/1-1. Moreover, MB acknowledges funding of DFG-GRK 2088. AM was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2067/1- 390729940. AM and MB acknowledge support of DFG-SFB 803 Z02. The authors thank Laura Jula Vanegas for help with parts of the implementation.

1. LJ van 't Veer, et al., Gene expression profiling predicts clinical outcome of breast cancer. *Nature* **415**, 530–536 (2002).
2. RD Gray, QD Atkinson, Language-tree divergence times support the Anatolian theory of Indo-European origin. *Nature* **426**, 435–439 (2003).
3. PB Eckburg, Diversity of the Human Intestinal Microbial Flora. *Science* **308**, 1635–1638 (2005).
4. T Hastie, R Tibshirani, JH Friedman, *The Elements of Statistical Learning*, Springer Series in Statistics. (Springer New York), (2009).
5. NR Faria, et al., The early spread and epidemic ignition of HIV-1 in human populations. *Science* **346**, 56–61 (2014).
6. MA Ansari, X Didelot, Bayesian inference of the evolution of a phenotype distribution on a phylogenetic tree. *Genetics* **204**, 89–98 (2016).
7. J Fukuyama, et al., Multidomain analyses of a longitudinal human microbiome intestinal cleanout perturbation experiment. *PLOS Comput. Biol.* **13**, e1005706 (2017).
8. NR Zhang, DO Siegmund, A modified Bayes information criterion with applications to the analysis of comparative genomic hybridization data. *Biometrics* **63**, 22–32 (2007).
9. R Killick, P Fearnhead, IA Eckley, Optimal detection of changepoints with a linear computational cost. *J. Am. Stat. Assoc.* **107**, 1590–1598 (2012).
10. J Sharpnack, A Singh, A Rinaldo, Changepoint detection over graphs with the spectral scan statistic in *Artificial Intelligence and Statistics.* pp. 545–553 (2013).
11. K Frick, A Munk, H Sieling, Multiscale change point inference. *J. Royal Stat. Soc. Ser. B (Statistical Methodol.* **76**, 495–580 (2014).
12. H Chen, N Zhang, Graph-based change-point detection. *The Annals Stat.* **43**, 139–176 (2015).
13. C Du, CLM Kao, SC Kou, Stepwise signal extraction via marginal likelihood. *J. Am. Stat. Assoc.* **111**, 314–330 (2015).
14. MA Ansari, et al., Genome-to-genome analysis highlights the effect of the human innate and adaptive immune systems on the hepatitis C virus. *Nat. Genet.* **49**, 666–673 (2017).
15. J Lu, et al., MicroRNA expression profiles classify human cancers. *Nature* **435**, 834–838 (2005).
16. SG Earle, et al., Identifying lineage effects when controlling for population structure improves power in bacterial association studies. *Nat. Microbiol.* **1**, 16041 (2016).
17. L Dümbgen, V Spokoiny, Multiscale testing of qualitative hypotheses. *The Annals Stat.* **29**, 124–152 (2001).
18. L Dümbgen, G Walther, Multiscale inference about a density. *The Annals Stat.* **36**, 1758–1785 (2008).
19. M Behr, C Holmes, A Munk, Multiscale blind source separation. *The Annals Stat.* **46**, 711–744 (2018).
20. D Siegmund, B Yakir, Tail probabilities for the null distribution of scanning statistics. *Bernoulli* **6**, 191 (2000).

DRAFT