UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Scienze Statistiche
Corso di Dottorato di Ricerca in Scienze Statistiche
Ciclo XXXIV

Resampling-based methods for multiple testing on high-dimensional data

Coordinatore del Corso: Prof. Nicola Sartori

Supervisore: Prof. Livio Finos

Co-supervisore: Prof. Jelle J. Goeman

Dottorando/a: Anna Vesely

12 January 2022

Abstract

We consider the problem of testing multiple hypotheses in high-dimensional settings, arguing that more tools are needed to support an exploratory approach, where researchers may test many subsets of hypotheses and make a selection post hoc. We focus on resampling-based methods, that rely on minimal assumptions and tend to be more powerful than parametric approaches, especially in presence of multiple hypotheses. In this framework, we provide two general and flexible procedures: a method to make confidence statements on the proportion of true discoveries (TDP), and a method to make inference on predictor variables in linear regression.

First, we propose a general closed testing procedure for sum-based global tests. It provides lower confidence bounds for the TDP, simultaneously over all subsets of hypotheses; these simultaneous inferences come for free, i.e., without any adjustment of the α-level, whenever a global test is used. Our method allows for an exploratory approach, as simultaneity ensures control of the TDP even when the subset of interest is selected post hoc. It adapts to the unknown joint distribution of the data through permutation testing. Any sum test may be employed, depending on the desired power properties. We present an iterative shortcut for the closed testing procedure, based on the branch and bound algorithm, which converges to the full closed testing results, often after few iterations; even if it is stopped early, it controls the TDP. We compare the properties of different choices for the sum test through simulations, then we illustrate the feasibility of the method for high dimensional settings on brain imaging data.

Subsequently, we propose a multiple testing method for hypotheses on coefficients in high-dimensional linear regression. It allows to construct asymptotically valid resampling-based tests for any subset of hypotheses, which can be used in closed testing procedures, including the above-mentioned shortcut. The approach is presented in two ways: an

exact method, and an approximate method that is less computationally intensive. We show that, to build test statistics for any set of hypotheses, it is sufficient to define test statistics for individual hypotheses, relying on a variable selection procedure, and then combine them through a suitable function. The resulting method is extremely flexible, allowing different selection procedures and several combining functions. The performance of the proposed exact and approximate methods is illustrated through the analysis of simulated data and real gene expression data.

Sommario

Nel contesto dei test multipli su dati ad alta dimensionalità, sono necessari nuovi strumenti per supportare un approccio esplorativo, in cui i ricercatori possano testare diversi sottoinsiemi di ipotesi e selezionare l'insieme di interesse post hoc. In questo manoscritto ci concentriamo sui test di permutazione, che richiedono assunzioni minime e sono generalmente più potenti degli approcci parametrici, soprattutto quando si considerano ipotesi multiple. Proponiamo due metodi generali e flessibili per dare un insieme di confidenza per la proporzione di veri positivi (*true discovery proportion*, TDP) e per fare inferenza sui predittori nella regressione lineare.

In primo luogo, proponiamo una procedura basata sul closed testing per test globali definiti tramite somme. Questa permette di calcolare limiti inferiori di confidenza per il TDP, simultaneamente rispetto a tutti i sottoinsiemi di ipotesi. Per qualsiasi test globale, tali inferenze simultanee sono disponibili senza aggiustare il livello di significatività. Il metodo proposto permette un approccio esplorativo, in quanto la simultaneità dei limiti di confidenza controlla il TDP anche quando l'insieme di interesse è selezionato post hoc. Inoltre, il metodo si adatta alla distribuzione dei dati tramite permutazioni. Si può utilizzare qualsiasi test basato sulle somme, a seconda delle proprietà desiderate. Il metodo è presentato come una scorciatoia iterativa per la procedura di closed testing, che sfrutta un algoritmo branch and bound e che converge al closed testing, spesso dopo poche iterazioni. La procedura controlla il TDP anche se interrotta prima di giungere a convergenza. Dopo aver confrontato le proprietà di diversi test globali tramite simulazioni, mostriamo che il metodo è adatto a dati ad alta dimensionalità analizzando un dataset di immagini cerebrali.

Successivamente, proponiamo una procedura per testare ipotesi multiple sui coefficienti di una regressione lineare ad alta dimensionalità. Il metodo costruisce test di

permutazione asintoticamente validi per ogni sottoinsieme di ipotesi. Tali test possono essere poi utilizzati all'interno di approcci basati sul closed testing, compresa la scorciatoia definita precedentemente. Proponiamo il metodo in due versioni, una esatta e un'approssimazione che richiede minori tempi computazionali e minore memoria. Mostriamo che, per calcolare delle statistiche test per qualsiasi insieme di ipotesi, è sufficiente definire delle statistiche per le singole ipotesi, sfruttando una procedura per la selezione di variabili; queste statistiche vengono poi combinate tramite funzioni con determinate caratteristiche. Ne risulta un metodo estremamente flessibile, che permette di usare diverse procedure di selezione e diverse funzioni per la combinazione. Illustriamo il comportamento del metodo esatto e di quello approssimato con l'analisi di dati simulati e dati di genetica.

To Valerio,

the most annoying

- and yet the loveliest -
flatmate ever

Acknowledgements

This manuscript would have never been possible without the immense support of Professors Livio Finos and Jelle Goeman. Many researchers may have great expertise and enthusiasm, but I truly believe it is not so common to find supervisors who are also this understanding, kind, and fun to work with.

I extend my gratitude to Dr. Pierre Neuvial and Dr. Jesse Hemerik for taking the time to review the thesis and for providing many interesting and stimulating insights.

During my visiting period at LUMC I was really lucky to meet many people who made me feel welcome and at home since the first day. I want to thank Prof. Wouter Weeda and Dr. Xu Chen in particular for letting me attend their meetings and helping me understand (a little bit of) fMRI analysis.

Dr. Angela Andreella, thank you for your friendship and the invaluable help you gave me. I cannot count the times I ran to your office with problems and questions, and I cannot think of one single time when you complained about it. I also wish to thank the other PhD students and researchers I spent my time with in Padua. I am glad for the mutual support and the great deal of fun that we had together - even if some of you may have unintentionally tried to sabotage my PhD by making me drink too much.

Contents

List of Figures

List of Tables

Introduction

Overview

In a large variety of fields, such as neuroimaging, genomics and psychometrics, data has a high-dimensional structure, meaning that the number of features under study is potentially much larger than the sample size. In this framework, interest often does not lie in studying single features, but in detecting subsets of features that are statistically significant. A typical example is functional Magnetic Reasonance Imaging (fMRI) data, where brain activation is measured as the correlation between a sequence of cognitive stimuli and blood oxygenation levels. The resulting brain image comprises approximately 300,000 volume units called voxels, each of which may be studied for significant neural activity. This huge multiple testing problem, consisting of around 300,000 statistical tests, is made even more complex by the fact that researchers are generally interested in brain regions, i.e., subsets of voxels. The goal is detecting, and possibly localizing and quantifying, significant activation within the brain.

This type of problems calls for new statistical tools, which must address the desired research questions while dealing with the difficulties that arise from high-dimensional data, including the heavy computational burden.

Furthermore, when testing multiple hypotheses an issue arises from the dependence structure of the data. Indeed, it has an impact on the behavior of the tests, and so must be taken into account, but is usually unknown. A suitable solution to this problem is provided by resampling-based methods (Fisher, 1936; Ernst, 2004), that rely on minimal assumptions (Hemerik and Goeman, 2018a) and generally offer an improvement in power over the parametric approach, especially when multiple hypotheses are considered (Westfall and Young, 1993; Pesarin, 2001; Hemerik and Goeman, 2018b; Hemerik *et al.*, 2019).

Therefore in this manuscript we focus on resampling-based procedures. We provide two general and flexible methods to perform multiple testing on high-dimensional data. First, we give a method to make confidence statements on the proportion of

true discoveries (TDP) within subsets of hypotheses, valid even under post-doc selection. Subsequently, we introduce a method to make inference on predictor variables for high-dimensional linear regression.

Main contributions of the thesis

Permutation-based true discovery guarantee by sum tests

Chapter 1 provides a new perspective on the age-old subject of global testing, with a focus on sum tests. This is a broad class of global tests that aggregate signal from multiple features through sums, including many p-value combinations and other popular multiple testing methods (e.g., see Pesarin, 2001; Goeman *et al.*, 2006).

Using results from the closed testing framework (Genovese and Wasserman, 2006; Goeman and Solari, 2011; Goeman *et al.*, 2019), we argue that all global tests automatically come with an inbuilt selective inference method, through which we can make many additional inferences without paying a price in terms of the global test's α-level. This allows not just to provide p-values, that only infer the presence of some discoveries, but also to make confidence statements on the TDP; this is considerably more informative, as it leads to quantify the proportion of these discoveries. Moreover, such statements come not just for the full testing problem, but also simultaneously over all subsets of hypotheses. This way, the procedure is not compromised by post-hoc selection, and so researchers can postpone the choice of subsets of interest until after seeing the data.

The main challenge of this framework is the computational complexity, which is exponential in the total number of hypotheses, and quickly grows to an infeasible size when many hypotheses are considered.

We propose a general closed testing procedure for sum global tests, which provides lower confidence bounds for the TDP, simultaneously over all subsets of hypotheses. The procedure is presented as an iterative shortcut for the full closed testing method. The shortcut is based on the branch and bound algorithm, and converges to the full closed testing results, often after few iterations; even if it is stopped early, it still provides valid confidence bounds for the TDP. Moreover, it adapts to the a-priori unknown joint distribution of the data through permutation testing. The resulting method is exact and extremely flexible, as it applies to any sum test and adapts to any data correlation structure.

We show through simulations how different choices of the sum test come with very different power properties. The advantages of the method, and in particular its feasibility

in high-dimensional settings, are illustrated through the analysis of real fMRI data, studying supra-threshold clusters with the goal of quantifying the proportion of active voxels.

Resampling-based inference for high-dimensional regression

Chapter 2 investigates the problem of making inference in the context of high-dimensional linear regression, where the number of predictor variables is greater than the sample size.

We build on the approach of Meinshausen *et al.* (2009) that assigns statistical significance to predictors by means of a data-splitting procedure. The main intuition is that p-values can be constructed for each variable by repeatedly splitting the data into two subsets. For each split, the first subset is used to reduce the number of variables through classical variable selection techniques such as the Lasso (Tibshirani, 1996); then the second subset is used to calculate p-values for the selected variables. Finally, the resulting p-values are adjusted and suitably aggregated over the different splits. Under some theoretical conditions, the aggregated p-values can be employed for asymptotic control of both the family-wise error rate (FWER) and the false discovery rate (FDR).

Based on the previous framework and the permutation test introduced in Hemerik *et al.* (2020), we propose a novel procedure to define permutation test statistics for each predictor in high-dimensional regression. We prove that these individual statistics can be combined and used as input in different multiple testing methods such as the maxT-method (Westfall and Young, 1993), closed testing procedures (Genovese and Wasserman, 2006; Goeman and Solari, 2011; Goeman *et al.*, 2019), as well as the shortcut introduced in Chapter 1. The procedure is asymptotically exact and flexible, as different selection methods and combinations of the individual test statistics can be used, with different power properties.

We exploit the same simulation settings as Meinshausen *et al.* (2009) to show the superiority of the new resampling-based method. Then we embed the method into the shortcut of Chapter 1 to study gene expression data.

Chapter 1

Permutation-based true discovery guarantee by sum tests

1.1 Introduction

In high-dimensional data analysis, researchers are often interested in detecting subsets of features that are associated with a given outcome. For instance, in functional magnetic resonance imaging (fMRI) data the objective may be to identify a brain region that is activated by a stimulus; in genomics data one may want to find a biological pathway that is differentially expressed. In this context, global tests allow to make meaningful statements at the set level. A diverse range of global tests has been proposed in literature: well-known examples are p-value combinations, described and compared in Pesarin (2001), Loughin (2004), Won *et al.* (2009) and Pesarin and Salmaso (2010); other popular methods are Simes' test (Simes, 1986), the global test of Goeman *et al.* (2006), the sequence kernel association test (Wu *et al.*, 2011) and higher criticism (Donoho and Jin, 2015). Their main characteristic is the ability to aggregate signal from multiple features. A substantial proportion, including many of the above-mentioned methods, is sum-based, meaning that the global test statistic may be written as a sum of contributions per feature. In this chapter we restrict to such sum-based tests.

The probability distribution of a global statistic depends not only on the marginal distributions of the data, but also on the joint distribution; for this reason, many sum tests only have a known null distribution under independence. Approaches that deal with the a-priori unknown joint distribution are worst-case distributions, defined either generally or under restrictive assumptions (Vovk and Wang, 2020), and nonparametric permutation testing (Fisher, 1936; Ernst, 2004). As worst-case distributions tend to be very conservative, the latter approach is preferable; it relies on minimal assumptions

(Hemerik and Goeman, 2018a), and generally offers an improvement in power over the parametric approach, especially when multiple hypotheses are considered (Westfall and Young, 1993; Pesarin, 2001; Hemerik and Goeman, 2018b; Hemerik *et al.*, 2019).

Rejecting a null hypothesis, however, gives little information on the corresponding set. A significant p-value only indicates that there is at least one true discovery, i.e., one feature associated with the outcome, but does not give any information on the proportion of true discoveries (TDP), nor their localization. This becomes problematic especially for large sets (Woo *et al.*, 2014). Moreover, since interest is usually not just in the set of all features, but in several subsets, a multiple testing procedure is necessary (Nichols, 2012; Meijer and Goeman, 2016). Finally, when researchers do not know a priori which subsets they are interested in, they may want to test many and then make the selection post hoc. The case for the use of TDPs in large-scale testing problems was argued by Rosenblatt *et al.* (2018) in neuroimaging and by Ebrahimpoor *et al.* (2020) in genomics.

This chapter presents a general approach for inference on the TDP. The method allows any sum-based test, requiring only that critical values are determined by permutations. It provides TDPs not only for the full testing problem, but also simultaneously for all subsets, allowing subsets of interest to be chosen post hoc.

We will solve this problem using the closed testing framework (Marcus *et al.*, 1976), which allows to construct confidence sets for the TDP simultaneously over all possible subsets (Genovese and Wasserman, 2006; Goeman and Solari, 2011; Goeman *et al.*, 2019). These additional simultaneous inferences on all subsets come for free, i.e., without any adjustment of the α-level, whenever a global test is applied. Simultaneity ensures that the procedure is not compromised by post-hoc selection, therefore researchers can postpone the choice of the subset until after seeing the data, while still obtaining valid confidence sets; used in this way, closed testing allows a form of post-hoc inference. Furthermore, closed testing has been proven to be the optimal way to construct multiple testing procedures, as all family-wise error rate (FWER), TDP and related methods are either equivalent to or can be improved by it (Goeman *et al.*, 2021). The main challenge of this framework is the computational complexity, which is extremely high when the multiple testing problem consists of many hypotheses, and when using many permutations. Closed testing for the TDP so far mostly focused on Simes-based test procedures, while sum-test-based closed testing was done under independence or with worst-case distributions (Vovk and Wang, 2020; Wilson, 2019; Tian *et al.*, 2021), simpler because critical values depend only on the size of the subset.

We propose a general closed testing procedure for sum-based permutation tests,

which provides simultaneous confidence sets for the TDP of all subsets of the testing problem. We develop two shortcuts to make this procedure feasible for large-scale problems. First, we develop a quick shortcut that approximates closed testing and has worst-case complexity of order $m \log^2(m)$ in the number m of individual hypotheses, and linearithmic in the number of permutations. Next, we embed this shortcut within a branch and bound algorithm, obtaining an iterative procedure that converges to full closed testing, often after few iterations; even if it is stopped early, it still controls the TDP. The resulting procedure is exact and extremely flexible, as it applies to any sum test and adapts to the correlation structure of the data. It can be scaled up to high-dimensional problems, such as fMRI data, whose typical dimension is of order 10^5. Finally, we show that particular choices of the sum test statistic, namely statistics based on truncation, result in faster procedures.

The structure of the chapter is as follows. First, we introduce sum tests in Section 1.2, then we review the properties of permutation testing and closed testing in Sections 1.3 and 1.4. We derive the single-step shortcut in Section 1.5, and characterize when it is equivalent to closed testing in Section 1.6. In Section 1.7 we define the iterative shortcut, and finally in Section 1.8 we introduce truncation-based statistics. In the remaining sections we compare the properties of different sum tests through simulations, explore an application to fMRI data, and discuss results. Proofs and some additional results are postponed to the Appendices (Sections 1.11, 1.12 and 1.13).

1.2 Sum tests

We start with a general definition of a sum test statistic. Let X be data, taking values in a sample space $\mathcal{X}$. Assume we are interested in studying m univariate hypotheses $H_1, \dots, H_m$, having indices in $M = \{1, \dots, m\}$, with significance level $\alpha \in [0, 1)$. Let $N \subseteq M$ be the unknown subset of true hypotheses. A generic subset $S \subseteq M$, with size $|S| = s$, defines an intersection hypothesis $H_S = \bigcap_{i \in S} H_i$, which is true if and only if $S \subseteq N$. In the particular case of $S = \emptyset$, we take $H_\emptyset$ as usual to be a hypothesis that is always true.

For each univariate hypothesis H_i, let $T_i : \mathcal{X} \to \mathbb{R}$ denote a test statistic. The general form of a sum test statistic for H_S is

$$T_S = g\left(\sum_{i \in S} f_i(T_i)\right),$$

where $f_i : \mathbb{R} \to \mathbb{R}$ are generic functions, and $g : \mathbb{R} \to \mathbb{R}$ is strictly monotone. Usually the

functions f_i are also taken as monotone, so that high values of T_S give evidence against H_S. Moreover, as f_i may depend on i, the contributions $f_i(T_i)$ may have different distributions, as in the case of weighted sums. Examples include p-value combinations such as Fisher (1925), Pearson (1933), Liptak/Stouffer (Liptak, 1958), Lancaster (1961), Edgington (1972), and Cauchy (Liu and Xie, 2020). We mention especially the generalized mean family (Vovk and Wang, 2020) with $f_i(x) = x^r$ and $g(y) = (y/s)^{1/r}$, where $r \in \mathbb{R}$, for which Wilson (2019) studied the harmonic mean ($r = -1$).

Since we can always re-write $\tilde{T}_i = f_i(T_i)$ and $\tilde{T}_S = g^{-1}(T_S)$, without loss of generality we can assume that f_i and g are the identity, so that

$$T_S = \sum_{i \in S} T_i.$$

In particular, for the empty set we obtain $T_\emptyset = 0$. Furthermore, we assume that the signs are chosen in such a way that high values of T_i, and therefore T_S, correspond to evidence against H_i and H_S, respectively.

1.3 Permutation testing

To find the critical value for the test statistic T_S we will use permutations. Let $\mathcal{P}$ be a collection of transformations $\pi : \mathcal{X} \to \mathcal{X}$ of the sample space; these may be permutations, but also other transformations such as rotations (Langsrud, 2005; Solari *et al.*, 2014) and sign flipping (Hemerik *et al.*, 2020). We assume that $\mathcal{P}$ is an algebraic group with respect to the operation of composition of functions. The group structure is important as, without it, the resulting test may be highly conservative or anti-conservative (Hoeffding, 1952; Southworth *et al.*, 2009).

Denote with $T_i = T_i(X)$ and $T_i^\pi = T_i(\pi X)$, with $\pi \in \mathcal{P}$, the statistics computed on the observed and transformed data, respectively. The main assumption of permutation testing is that the joint distribution of the statistics T_i^π, with $i \in N$ and $\pi \in \mathcal{P}$, is invariant under all transformations in $\mathcal{P}$ of X. This assumption is common to most permutation-based multiple-testing methods, such as the maxT-method (Westfall and Young, 1993; Meinshausen, 2006; Goeman and Solari, 2010; Hemerik *et al.*, 2019). For some choices of the group $\mathcal{P}$, the assumption holds only asymptotically (Winkler *et al.*, 2014; Solari *et al.*, 2014; Hemerik *et al.*, 2020). Detailed illustration and examples can be found in Pesarin (2001), Huang *et al.* (2006) and Hemerik and Goeman (2018a). Here, we consider a slightly stronger assumption that is easier to check.

Assumption 1.1. *Given the partition $X = (X_1, \ldots, X_m)$, the statistic $T_S = T_S(X_S)$ is a function of $X_S = (X_i : i \in S)$ only. Moreover, $X_N \overset{d}{=} \pi X_N$ for each $\pi \in \mathcal{P}$, where $\overset{d}{=}$ denotes equality in distribution.*

Note that Assumption 1.1 holds in the particular case when H_S true implies that $X_S \overset{\mathrm{d}}{=} \pi X_S$ for each π.

If the cardinality of $\mathcal{P}$ is large, a valid α-level test may use B randomly chosen elements (Hemerik and Goeman, 2018b). The value of B does not need to grow with m or s; to have non-zero power we must only have $B \geq 1/\alpha$, though larger values of B give more power. For $\alpha = 0.05$, $B \geq 200$ is generally sufficient (see Section 1.9.2). Consider a vector $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_B)$, where $\pi_1 = \mathrm{id}$ is the identity in $\mathcal{P}$, and $\pi_2, \ldots, \pi_B$ are random elements drawn with replacement from a uniform distribution on $\mathcal{P}$. Then set $\omega_0 = \lceil (1-\alpha)B \rceil$, where $\lceil \cdot \rceil$ represents the ceiling function. A critical value for the test statistic T_S is given by the quantile $T_S^{(\omega_0)}$, where $T_S^{(1)} \leq \ldots \leq T_S^{(B)}$ are the sorted statistics T_S^{π}, with $\pi \in \boldsymbol{\pi}$. Then a permutation test may be defined as following.

Lemma 1.2. *Under Assumption 1.1, the test that rejects H_S when $T_S > T_S^{(\omega_0)}$ is an α-level test.*

In permutation testing, both the test statistic and the critical value are random variables. For our method it will be convenient to use an equivalent characterization of the test with a non-random critical value. Therefore, we define the centered statistic $C_S^{\pi} = T_S - T_S^{\pi}$ for each π, so that the observed value $C_S = C_S^{\mathrm{id}}$ is always zero, and all variability in the critical value is incorporated into the test statistic. We give a permutation test based on these new statistics, by using $\omega = \lceil \alpha B \rceil$ to obtain the quantile.

Theorem 1.3. *Under Assumption 1.1, the test that rejects H_S when $C_S^{(\omega)} > 0$ is an α-level test.*

For illustration, we introduce a recurring toy example with $m = 5$ univariate hypotheses and $B = 6$ transformations (Table 1.1). Let $M = \{1, 2, 3, 4, 5\}$, and suppose we are interested in testing subset $S = \{1, 2\}$ at significance level $\alpha = 0.4$. The vectors of the corresponding statistics T_S^{π} and C_S^{π} are obtained by summing columns 1 and 2 by row. Since $\omega = 3$ and $C_S^{(\omega)} = 2$, the permutation test rejects H_S.

1.4 True discovery guarantee

Based on the notation introduced above, consider the number of true discoveries $\delta(S) = |S \setminus N|$ made when rejecting H_S. We are interested in deriving simultaneous $(1-\alpha)$-confidence sets for this number, so that the simultaneity makes their coverage robust

TABLE 1.1: Toy example: original and centered test statistics.

	original T_i^π					centered C_i^π				
	H_1	H_2	H_3	H_4	H_5	H_1	H_2	H_3	H_4	H_5
id	6	5	4	1	1	0	0	0	0	0
π_2	1	2	1	0	4	5	3	3	1	-3
π_3	8	3	0	2	1	-2	2	4	-1	0
π_4	8	1	0	1	0	-2	4	4	0	1
π_5	0	6	1	1	2	6	-1	3	0	-1
π_6	7	0	1	2	1	-1	5	3	-1	0

against post-hoc selection. This way, the rejected hypothesis can be selected after reviewing all confidence sets, while still keeping correct $(1-\alpha)$-coverage of the corresponding confidence set (Goeman and Solari, 2011).

Let $d : 2^M \to \mathbb{R}$ be a random function, where 2^M is the power set of M. With the same notation of Goeman *et al.* (2021), we say that d has true discovery guarantee if $d(S)$ are simultaneous lower $(1-\alpha)$-confidence bounds for $\delta(S)$, i.e.,

$$P\left(\delta(S) \geq d(S) \text{ for each } S \subseteq M\right) \geq 1-\alpha.$$

An equivalent condition is that $\{d(S), \ldots, s\}$ is a $(1-\alpha)$-confidence set for $\delta(S)$, simultaneously for all $S \subseteq M$. Notice that the resulting confidence sets are one-sided, since hypothesis testing is focused on rejecting, not accepting. From $d(S)$ simultaneous $(1-\alpha)$-confidence sets can be immediately derived for other quantities of interest such as the TDP and the number or proportion of false discoveries (Goeman and Solari, 2011).

A general way to construct procedures with true discovery guarantee is provided by closed testing, based on the principle of testing different subsets by means of a valid α-level local test, which in this case is the permutation test. Throughout this chapter, we will loosely say that a set S is rejected when the corresponding hypothesis H_S is. Hence denote the collection of sets rejected by the permutation test by

$$\mathcal{R} = \left\{ S \subseteq M \,:\, C_S^{(\omega)} > 0 \right\}.$$

Genovese and Wasserman (2006) and Goeman and Solari (2011) defined a procedure d with true discovery guarantee as

$$d(S) = s - q(S),$$

where

$$q(S) = \max\{|V \cap S| \,:\, V \subseteq M,\, V \notin \mathcal{R}\}\,. \tag{1.1}$$

The main challenge of this method is its exponential complexity in the number of hypotheses. Indeed, the number of tests that must be evaluated to determine $d(S)$ may be up to order 2^m. In the toy example, where $m = 5$, this number is 32; it is immediate that it quickly grows to an infeasible size as m increases.

1.5 Shortcut

Fix the set of interest S, so that any dependence on it may be omitted in the notation. We propose a shortcut that quickly evaluates whether $q < z$, for any value z. This will allow to approximate q, and eventually define a procedure with true discovery guarantee.

First, we will re-write q as the unique change-point of an increasing function:

$$\phi : \{0, \ldots, s+1\} \longrightarrow \{0, 1\}, \qquad \phi(z) = 1 \quad \text{if and only if} \quad q < z \tag{1.2}$$

$$q = \max\{z \in \{0, \ldots, s+1\} \,:\, \phi(z) = 0\}\,. \tag{1.3}$$

Then we will approximate q from above with the change point $q^{(0)}$ of a second increasing function:

$$\underline{\phi} : \{0, \ldots, s+1\} \longrightarrow \{0, 1\}, \qquad \underline{\phi}(z) \leq \phi(z) \tag{1.4}$$

$$q^{(0)} = \max\left\{z \in \{0, \ldots, s+1\} \,:\, \underline{\phi}(z) = 0\right\}. \tag{1.5}$$

We start by giving an equivalent characterization of the quantity of interest q. For any $z \in \{0, \ldots, s+1\}$, we define the collection

$$\mathcal{V}_z = \{V \subseteq M \,:\, |V \cap S| \geq z\}$$

of sets that have at least size z overlap with S, and investigate whether all its elements are rejected. We define ϕ as in (1.2) so that it represents such rejection, taking

$$\phi(z) = \mathbf{1}\{\mathcal{V}_z \subseteq \mathcal{R}\} \quad (z \in \{0, \ldots, m\}), \tag{1.6}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function.

Lemma 1.4. *$\phi(0) = 0$ and $\phi(s+1) = 1$. Moreover, $\phi(z) = 0$ if and only if $z \in \{0, \dots, q\}$.*

Now we fix a value $z \in \{1, \dots, s\}$ and derive the shortcut to make statements on $\phi(z)$ without testing all the sets contained in $\mathcal{V}_z$. We do this by partitioning $\mathcal{V}_z$ by the size of its elements, obtaining

$$\mathcal{V}_z = \bigcap_{v=z}^{m} \mathcal{V}_z(v), \qquad \mathcal{V}_z(v) = \{V \in \mathcal{V}_z \,:\, |V| = v\}. \tag{1.7}$$

Each $\mathcal{V}_z(v)$ is the sub-collection of all sets of size v that have at least size z overlap with S. We can analyze these sub-collections separately and combine the results, noting that $\phi(z) = 1$ if and only if $\mathcal{V}_z(v) \subseteq \mathcal{R}$ for all $v \in \{z, \dots, m\}$.

By definition, $\mathcal{V}_z(v) \subseteq \mathcal{R}$ when all sets in the sub-collection have positive quantiles, i.e., $C_V^{(\omega)} > 0$ for each $V \in \mathcal{V}_z(v)$. The main idea of the shortcut is to obtain information on each sub-collection $\mathcal{V}_z(v)$ by bounding the corresponding quantiles from below. In particular, we will construct a bound

$$\ell_z \,:\, \{z, \dots, m\} \longrightarrow \mathbb{R}, \qquad \ell_z(v) \leq C_V^{(\omega)} \quad \text{for each } V \in \mathcal{V}_z(v). \tag{1.8}$$

This way, if $\ell_z(v) > 0$, we know that all sets in $\mathcal{V}_z(v)$ have positive quantiles. If ℓ_z is positive in its entire domain, then $\mathcal{V}_z(v) \subseteq \mathcal{R}$ for each v, and so $\phi(z) = 1$. Figure 1.1 displays the bound, which we will define in the following paragraphs, in the toy example for $z = 1$ and $z = 2$. Note that indeed all quantiles lie on it or above; the bound can be loose, as seen with $\ell_1(3)$. Since ℓ_2 lies entirely in the positive half-space, we know that $\phi(2) = 1$. In contrast, we cannot make a statement on $\phi(1)$ based on ℓ_1.

Fix a size $v \in \{z, \dots, m\}$. To define an $\ell_z(v)$ that does not exceed the minimum quantile over all sets in $\mathcal{V}_z(v)$, as required in (1.8), we approximate the minimum quantile from below with the quantile of the minimum. We do this by taking the smallest centered statistics for each transformation π, with some constraints from the structure of $\mathcal{V}_z(v)$.

In the toy example, choose $z = 1$, and let V be any set in the sub-collection $\mathcal{V}_1(v)$ of interest. Note that V must contain v indices, at least $z = 1$ of which is in S. Consider the centered statistics $C_i^{\pi_2}$ for transformation π_2 (second row in Table 1.1, right). First, we select the lowest value in S, then we sort the remaining values in ascending order, as in the second row of Table 1.2. If $b_v^{\pi_2}$ is the sum of the first v elements of the row, we know that $b_v^{\pi_2} \leq C_V^{\pi_2}$. After constructing the other rows of Table 1.2 according to the same principle, we define $\ell_1(v) = b_v^{(\omega)}$; since $b_v^{\pi} \leq C_V^{\pi}$ for each π, we obtain $\ell_1(v) \leq C_V^{(\omega)}$.

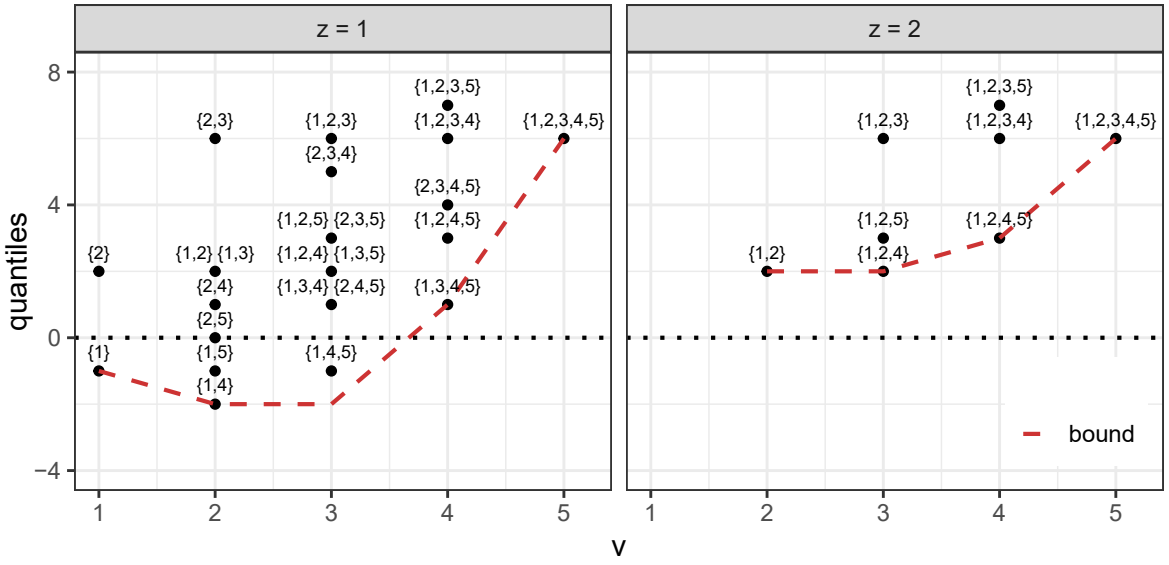

FIGURE 1.1: Toy example with $S = \{1, 2\}$: shortcut to evaluate $\phi(z)$ in $z = 1$ and $z = 2$. Points denote the quantiles for the sets in $\mathcal{V}_z$. The dashed line represents the bound ℓ_z.

TABLE 1.2: Toy example with $S = \{1, 2\}$: matrix of the sorted centered statistics to compute the bound ℓ_1. The value $\ell_1(v)$ is obtained by summing the first v columns by row, and then taking the quantile.

	selected in S	remaining			
	$i_1(\pi)$	$j_1(\pi)$	$j_2(\pi)$	$j_3(\pi)$	$j_4(\pi)$
id	0 (H_1)	0 (H_2)	0 (H_3)	0 (H_4)	0 (H_5)
π_2	3 (H_2)	-3 (H_5)	1 (H_4)	3 (H_3)	5 (H_1)
π_3	-2 (H_1)	-1 (H_4)	0 (H_5)	2 (H_2)	4 (H_3)
π_4	-2 (H_1)	0 (H_4)	1 (H_5)	4 (H_2)	4 (H_3)
π_5	-1 (H_2)	-1 (H_5)	0 (H_4)	3 (H_3)	6 (H_1)
π_6	-1 (H_1)	-1 (H_4)	0 (H_5)	3 (H_3)	5 (H_2)

In general, for each $\pi \in \boldsymbol{\pi}$, we select the z smallest centered statistics in S, and then the $v - z$ remaining smallest statistics. We define two permutations of the indices:

$$S = \{i_1(\pi), \ldots, i_s(\pi)\} \quad : \quad C^{\pi}_{i_1(\pi)} \leq \ldots \leq C^{\pi}_{i_s(\pi)} \tag{1.9}$$

$$M \setminus \{i_1(\pi), \ldots, i_z(\pi)\} = \{j_1(\pi), \ldots, j_{m-z}(\pi)\} \quad : \quad C^{\pi}_{j_1(\pi)} \leq \ldots \leq C^{\pi}_{j_{m-z}(\pi)}. \tag{1.10}$$

The set $\{i_1(\pi), \ldots, i_z(\pi)\}$ is a subset of S, containing the indices of the z smallest values in S (for transformation π). For instance, in the toy example we have $S = \{2, 1\}$, and

$M \setminus \{2\} = \{5, 4, 3, 1\}$. Then the value of the bound is defined as

$$\ell_z(v) = b_v^{(\omega)} \quad \text{where} \quad b_v^\pi = \sum_{h=1}^{z} C_{i_h}^\pi + \sum_{h=1}^{v-z} C_{j_h}^\pi \quad (\pi \in \boldsymbol{\pi}). \tag{1.11}$$

Lemma 1.5. *$\ell_z(v) \leq C_V^{(\omega)}$ for all $V \in \mathcal{V}_z(v)$. Hence $\min_v \ell_z(v) > 0$ implies $\phi(z) = 1$.*

Now we use the bound to define a function $\underline{\phi}$ as in (1.4). In the extremes, where the value of ϕ is known, we set $\underline{\phi}(0) = \phi(0) = 0$ and $\underline{\phi}(s+1) = \phi(s+1) = 1$ (see Lemma 1.4). Elsewhere, we set

$$\underline{\phi}(z) = \mathbf{1}\left\{\min_v \ell_z(v) > 0\right\} \quad (z \in \{1, \ldots, s\}). \tag{1.12}$$

This function may not be monotonic, but we are only interested in its smallest change point; indeed, if $\underline{\phi}(z) = 1$ for a value z, we know that $q < z$. We make it increasing and obtain a single change point in $q^{(0)}$, as defined in (1.5), by imposing

$$\underline{\phi}(z) = 1 \quad \text{if} \quad \underline{\phi}(z^*) = 1 \text{ for some } z^* \in \{0, \ldots, z\} \quad (z \in \{1, \ldots, s\}). \tag{1.13}$$

Proposition 1.6. *As $\underline{\phi}(z) \leq \phi(z)$ for each $z \in \{0, \ldots, s+1\}$, $q^{(0)} \geq q$.*

For instance, in the toy example of Figure 1.1, $\underline{\phi}(1) = 0$ and $\underline{\phi}(2) = 1$, and so $q^{(0)} = 1$. Finally, from this result we can approximate d from below with

$$d^{(0)} = s - q^{(0)}.$$

Theorem 1.7. *$d^{(0)} \leq d$.*

In conclusion, Proposition 1.6 represents the basis of the shortcut. For any value z, it allows to make statements on the value of $\phi(z)$ by constructing $\underline{\phi}(z) \leq \phi(z)$; it requires to evaluate a number of tests which is linear in the total number m of hypotheses, in contrast to the exponential number required by closed testing. Subsequently, Theorem 1.7 employs the shortcut to provide a lower $(1-\alpha)$-confidence bound $d^{(0)}$ for the number of true discoveries δ. The Theorem holds for all $S \subseteq M$, hence the procedure $d^{(0)}$ has true discovery guarantee; this means that $d^{(0)}(S)$ are lower $(1-\alpha)$-confidence bounds for $\delta(S)$, simultaneously for all $S \subseteq M$.

In Section 1.11 we propose an algorithm for the shortcut, then we embed it into a binary search to approximate q with reduced complexity. We prove that in the worst case the computational complexity is of order $m \log^2(m)$ in the number m of hypotheses, and linearithmic in the number of permutations. Moreover, we show how the method

can be combined with an algorithm of Tian *et al.* (2021) to find the largest set with given TDP among a collection of incremental sets.

1.6 Equivalence to closed testing

The shortcut of Proposition 1.6 defines $\underline{\phi}(z) \leq \phi(z)$ for any z. For those values of z for which $\underline{\phi}(z) = 1$, we know that also $\phi(z) = 1$. Where $\underline{\phi}(z) = 0$, however, there are two distinct cases. If $\phi(z) = 0$, the shortcut is equivalent to closed testing; otherwise, if $\phi(z) = 1$, it is conservative, as it does not reject all sets in $\mathcal{V}_z$ while closed testing does. In the toy example with $z = 1$ we are in the first case (Figure 1.1, left), but we cannot see that from the bound only. Now we propose a sufficient condition to state that $\underline{\phi}(z) = \phi(z)$. This will play an important role in the iterative shortcut of Section 1.7.

We will define an increasing function

$$\overline{\phi} : \{0, \dots, s+1\} \longrightarrow \{0, 1\}, \qquad \underline{\phi}(z) \leq \phi(z) \leq \overline{\phi}(z). \tag{1.14}$$

This way, if $\underline{\phi}(z) = \overline{\phi}(z)$ for a value z, we know that $\underline{\phi}(z) = \phi(z)$. Note that this holds in particular when either $\underline{\phi}(z) = 1$ or $\overline{\phi}(z) = 0$.

Fix $z \in \{1, \dots, s\}$. Based on partition (1.7) of $\mathcal{V}_z$, the main idea is to construct a greedy path of sets $V_z \subset \cdots \subset V_m$, with $V_v \in \mathcal{V}_z(v)$ for each v, and check whether their quantiles are all strictly positive. If we find a non-positive quantile, then we have established that $\mathcal{V}_z \not\subseteq \mathcal{R}$, and so $\underline{\phi}(z) = \phi(z) = 0$; the shortcut is equivalent to closed testing for this value of z. We will define the path

$$u_z : \{z, \dots, m\} \longrightarrow \mathbb{R}, \qquad u_z(v) = C^{(\omega)}_{V_v} \quad \text{with} \quad V_v \in \mathcal{V}_z(v) \tag{1.15}$$

that connects these quantiles. This way, if $u_z(v) \leq 0$, we know that $\mathcal{V}_z(v)$ contains a non-rejected set, and so we conclude that $\phi(z) = 0$. Figure 1.2 displays the bound ℓ_z and the path u_z, which we will define in the next paragraphs, for the toy example with $z = 1$ and $z = 2$. The path connects some of the quantiles, one for each size v, and so is never smaller than the bound. From ℓ_2 we already had $\phi(2) = 1$; as u_1 is entirely positive, results on $\phi(1)$ are still unsure.

Fix a size $v \in \{z, \dots, m\}$. We define $u_z(v)$ as the quantile of a set $V_v \in \mathcal{V}_z(v)$, as required in (1.15), choosing V_v such that it is unlikely to be rejected. We take V_v as the set containing the smallest observed non-centered statistics, with the constraint that V_v must be an element of $\mathcal{V}_z(v)$. This is a heuristic choice: the observed value of a test

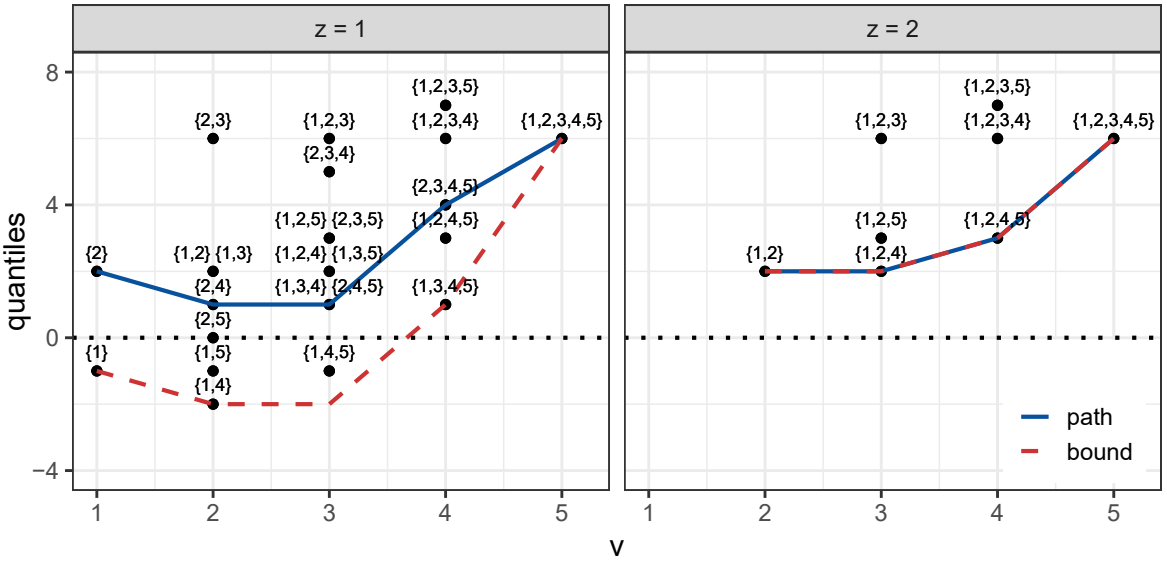

FIGURE 1.2: Toy example with $S = \{1, 2\}$: shortcut to evaluate $\phi(z)$ in $z = 1$ and $z = 2$. Points denote the quantiles for the sets in $\mathcal{V}_z$. The solid and dashed lines represent the path u_z and the bound ℓ_z, respectively.

TABLE 1.3: Toy example with $S = \{1, 2\}$: matrix of the sorted centered statistics to compute the path u_1. The value $u_1(v)$ is obtained by summing the first v columns by row, and then taking the quantile.

	selected in S	remaining			
	i_1 (H_2)	j_1 (H_4)	j_2 (H_5)	j_3 (H_3)	j_4 (H_1)
id	0	0	0	0	0
π_2	3	1	-3	3	5
π_3	2	-1	0	4	-2
π_4	4	0	1	4	-2
π_5	-1	0	-1	3	6
π_6	5	-1	0	3	-1

statistic T_i by itself does not provide full information on the rejection of H_i; still, if T_i is small, generally H_i is less likely to be rejected.

In the toy example, choose $z = 1$. The set $V_v \in \mathcal{V}_1(v)$ must contain v indices, at least $z = 1$ of which is in S. Consider the observed statistics T_i (first row in Table 1.1, left). First, we select the column of the smallest value in S, then sort the remaining columns so that their values are in ascending order. Table 1.3 presents the centered statistics C_i^π according to this new order. We define V_v as the set of the indices of the first v columns, obtaining $V_1 = \{2\}$, $V_2 = \{2, 4\}$, $V_3 = \{2, 4, 5\}$, $V_4 = \{2, 4, 5, 3\}$ and $V_5 = M$.

In general, we select the z smallest observed non-centered statistics in S, and then the $v - z$ remaining smallest statistics. We define two permutations of the indices:

$$S = \{i_1, \ldots, i_s\} \quad : \quad T_{i_1} \leq \ldots \leq T_{i_s} \tag{1.16}$$

$$M \setminus \{i_1, \ldots, i_z\} = \{j_1, \ldots, j_{m-z}\} \quad : \quad T_{j_1} \leq \ldots \leq T_{j_{m-z}}. \tag{1.17}$$

The set $\{i_1, \ldots, i_z\}$ is a subset of S, containing the indices of the z smallest values in S. For instance, in the toy example we have $S = \{2, 1\}$, and $M \setminus \{2\} = \{4, 5, 3, 1\}$. The value of the path is then defined as

$$u_z(v) = C^{(\omega)}_{V_v} \quad \text{where} \quad V_v = \{i_1, \ldots, i_z\} \cup \{j_1, \ldots, j_{v-z}\}. \tag{1.18}$$

It is immediate that $V_v \in \mathcal{V}_z(v)$, and $u_z(v) \geq \ell_z(v)$.

Lemma 1.8. $\min_v u_z(v) \leq 0$ *implies* $\phi(z) = 0$.

The path is used to define a function $\overline{\phi}$ as in (1.14). Similarly to the definition of $\underline{\phi}$ in the previous Section, first we set $\overline{\phi}(0) = \phi(0) = 0$, $\overline{\phi}(s+1) = \phi(s+1) = 1$, and

$$\overline{\phi}(z) = \mathbf{1}\left\{\min_v u_z(v) > 0\right\} \quad (z \in \{1, \ldots, s\}). \tag{1.19}$$

Then we make the function increasing by taking only its largest change point, imposing

$$\overline{\phi}(z) = 0 \quad \text{if} \quad \overline{\phi}(z^*) = 0 \text{ for some } z^* \in \{z, \ldots, s+1\} \quad (z \in \{1, \ldots, s\}). \tag{1.20}$$

Proposition 1.9. $\underline{\phi}(z) \leq \phi(z) \leq \overline{\phi}(z)$ *for each* $z \in \{0, \ldots, s+1\}$. *Hence* $\underline{\phi}(z) = \overline{\phi}(z)$ *implies* $\underline{\phi}(z) = \phi(z)$, *i.e., equivalence between the shortcut and closed testing.*

For instance, in the toy example of Figure 1.2 we obtain $\underline{\phi}(1) = 0 < \overline{\phi}(1) = 1$ and $\underline{\phi}(2) = \overline{\phi}(2) = 1$. Hence the shortcut is equivalent to closed testing for $z = 2$, as we already observed, but we cannot establish equivalence for $z = 1$.

To summarize, the shortcut of Proposition 1.6 compares q with any value z, then Proposition 1.9 studies whether this shortcut is equivalent to closed testing or conservative. The following section shows how to improve the shortcut in case it is conservative.

1.7 Iterative shortcut

The shortcut we have described in Section 1.5 approximates closed testing and efficiently computes $q^{(0)} \geq q$; however, as seen in Section 1.6, it may be conservative. In this

section we improve this single-step shortcut by embedding it into a branch and bound algorithm. We obtain an iterative shortcut which defines closer approximations of q, and thus smaller confidence sets for δ, as the number of steps increases. Eventually, after a finite number of steps, it reaches the same results as full closed testing.

At each step $n \in \mathbb{N}$, we will define two increasing functions

$$\underline{\phi}^{(n)}, \overline{\phi}^{(n)} : \{0, \dots, s+1\} \longrightarrow \{0, 1\}, \qquad \underline{\phi}^{(n)}(z) \leq \phi(z) \leq \overline{\phi}^{(n)}(z). \tag{1.21}$$

We will approximate q from above with the change point of the first function,

$$q^{(n)} = \max \left\{ z \in \{0, \dots, s+1\} \, : \, \underline{\phi}^{(n)}(z) = 0 \right\}. \tag{1.22}$$

Then we will use the second to assess possible equivalence to closed testing. If $\underline{\phi}^{(n)}(z) = \overline{\phi}^{(n)}(z)$ for a value z, then $\underline{\phi}^{(n)}(z) = \phi(z)$ and so results cannot be further improved. Moreover, these functions will be defined so that $q^{(n)}$ becomes a better approximation of q as n increases, and finally converges to it after at most m steps:

$$q^{(n)} \geq q^{(n+1)} \geq q^{(m)} = q \quad (n \in \mathbb{N}). \tag{1.23}$$

In the next sections we introduce the structure of the branch and bound algorithm, then use it to construct the functions $\underline{\phi}^{(n)}$ and $\overline{\phi}^{(n)}$ with the desired properties.

1.7.1 Branch and bound

The branch and bound algorithm (Land and Doig, 1960; Mitten, 1970) is used when exploring a space of elements in search of a solution, and is based on the following principle. The space is partitioned into two subspaces, and each subspace is systematically evaluated; the procedure can be iterated until the best solution is found. Hence the algorithm consists of a branching rule, which defines how to generate subspaces, and a bounding rule, which gives bounds on the solution. This way, one can discard entire subspaces that, according to the bounding rule, cannot contain the solution.

Here, we want to evaluate $\phi(z)$ for any value z, i.e., determine whether the space $\mathcal{V}_z$ contains a non-rejected set (see definition (1.6)). The bounding rule that allows to make statements on the existence of such a set is the single-step shortcut of Propositions 1.6 and 1.9. If the shortcut is equivalent to closed testing, meaning that we are able to determine $\phi(z)$, the procedure stops; otherwise, we partition $\mathcal{V}_z$, and apply the shortcut within each resulting subspace. This procedure may be iterated as needed.

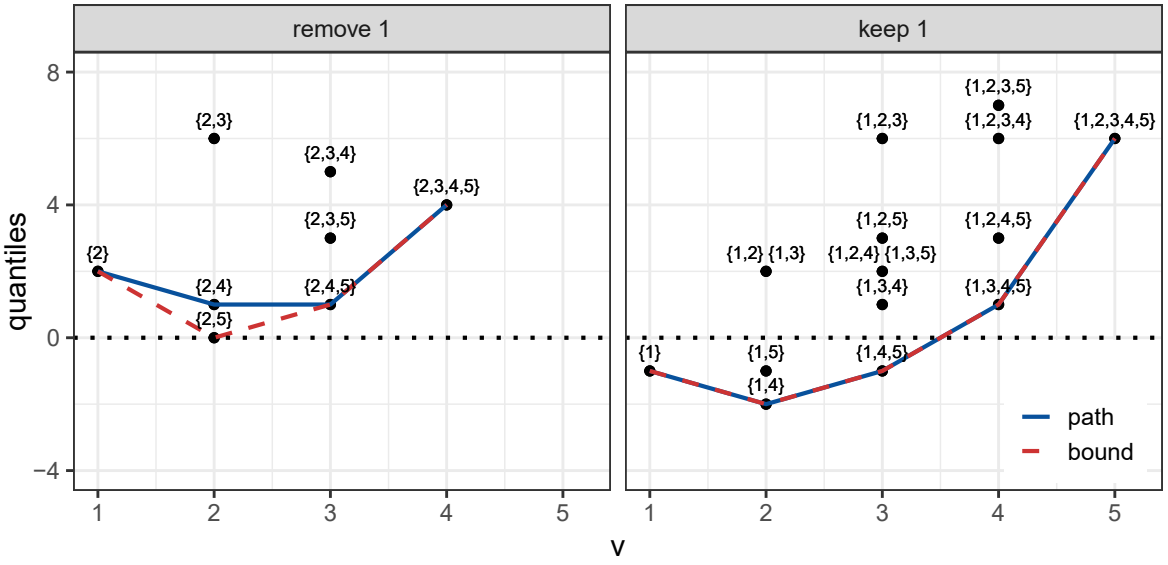

FIGURE 1.3: Toy example with $S = \{1, 2\}$: iterative shortcut at step $n = 1$ to evaluate $\phi(z)$ in $z = 1$. Points denote the quantiles for the sets in $\mathcal{V}_1^-$ and $\mathcal{V}_1^+$. The solid and dashed lines represent the path and the bound, respectively.

For instance, in the toy example, the single-step shortcut gives $\phi(2) = 1$ but cannot determine $\phi(1)$ (Figure 1.2). At step $n = 1$, we partition $\mathcal{V}_1$ into two subspaces $\mathcal{V}_1^-$ and $\mathcal{V}_1^+$, according to the inclusion of index $j^* = 1$: $\mathcal{V}_1^-$ contains all sets that do not include j^*, and $\mathcal{V}_1^+$ contains the others. We choose $j^* \in M$ as the index of the hypothesis that we believe we have most evidence against, i.e., having the greatest observed statistic T_i (first row in Table 1.1, left). Subsequently, we use the shortcut to examine each subspace. Figure 1.3 shows the bound ℓ_1 and the path u_1 in the two subspaces; the path indicates that $\mathcal{V}_1^+$ contains a non-rejected set, therefore we conclude that $\phi(1) = 0$.

In general, the branching rule is chosen to find an eventual non-rejected set with the smallest number of steps. Fix $z \in \{1, \dots, s\}$, as by Lemma 1.4 there is no need to partition $\mathcal{V}_0$ or $\mathcal{V}_{s+1}$. The space $\mathcal{V}_z$ of interest is partitioned into

$$\mathcal{V}_z^- = \{V \in \mathcal{V}_z : j^* \notin V\}, \qquad \mathcal{V}_z^+ = \{V \in \mathcal{V}_z : j^* \in V\}$$

where j^* is the index of the greatest observed non-centered statistic, with the constraint that the procedure cannot generate empty subspaces. Recall that any set $V \in \mathcal{V}_z$ has at least size z overlap with S. Hence, with the notation of (1.16) and (1.17), we fix the indices $\{i_1, \dots, i_z\}$ of the z smallest observed statistics in S, then we take $j^* = j_{m-z}$ as the index of the greatest remaining observed statistic. The same principle may be applied to partition any subspace.

At any step $n \in \mathbb{N}$, the procedure partitions $\mathcal{V}_z$ into $K_{n,z}$ subspaces $\mathcal{V}_z^1, \dots, \mathcal{V}_z^{K_{n,z}}$ without any successors, where $K_{n,z} \in \{1, \dots, 2^n\}$. Suppose to apply the single-step shortcut within a subspace $\mathcal{V}_z^k$. If the result is $\phi(z) = 0$, then $\mathcal{V}_z^k$ contains a non-rejected set, and we stop with $\phi(z) = 0$. In contrast, if the shortcut determines that $\phi(z) = 1$, all sets in $\mathcal{V}_z^k$ are rejected, and we may explore other subspaces. Finally, if the shortcut produces an unsure outcome, i.e., $\phi(z)$ is still unknown, $\mathcal{V}_z^k$ can be partitioned again.

1.7.2 Structure of the iterative shortcut

Fix a step $n \in \mathbb{N}$. For every z, the branching rule partitions $\mathcal{V}_z$ into $K_{n,z}$ subspaces $\mathcal{V}_z^1, \dots, \mathcal{V}_z^{K_{n,z}}$, and the bounding rule applies the shortcut within them. We use this structure to define the functions $\underline{\phi}^{(n)}$ and $\overline{\phi}^{(n)}$ introduced in (1.21). We consider the point-wise minimums of $\underline{\phi}$ and $\overline{\phi}$ within the different subspaces, and so we take

$$\underline{\phi}^{(n)}(z) = \min_k \left\{ \underline{\phi}(z) \text{ in } \mathcal{V}_z^k \right\}, \qquad \overline{\phi}^{(n)}(z) = \min_k \left\{ \overline{\phi}(z) \text{ in } \mathcal{V}_z^k \right\}.$$

Since $\underline{\phi}$ and $\overline{\phi}$ are increasing functions, also $\underline{\phi}^{(n)}$ and $\overline{\phi}^{(n)}$ are increasing. The following proposition shows that property (1.21) holds, so that we can approximate q from above with $q^{(n)}$, and we can assess possible equivalence to closed testing for any z. Moreover, the proposition gives property (1.23) by showing that $\underline{\phi}^{(n)}$ and $\overline{\phi}^{(n)}$ become closer to ϕ as n increases, and finally converge to it after at most m steps.

Proposition 1.10. *For any $n \in \mathbb{N}$ and any $z \in \{0, \dots, s+1\}$,*

$$\underline{\phi}^{(n)}(z) \le \underline{\phi}^{(n+1)}(z) \le \underline{\phi}^{(m)}(z) = \phi(z) = \overline{\phi}^{(m)}(z) \le \overline{\phi}^{(n+1)}(z) \le \overline{\phi}^{(n)}(z).$$

Hence $\underline{\phi}^{(n)}(z) = \overline{\phi}^{(n)}(z)$ implies $\underline{\phi}^{(n)}(z) = \phi(z)$, i.e., equivalence between the iterative shortcut and closed testing. Moreover, $q^{(n)} \ge q^{(n+1)} \ge q^{(m)} = q$.

In the toy example, consider step $n = 1$ of the iterative shortcut. For $z = 2$, from results of the single-step shortcut we have $\underline{\phi}^{(1)}(2) = \overline{\phi}^{(1)}(2) = \phi(2) = 1$ without partitioning $\mathcal{V}_2$. For $z = 1$, from Figure 1.3 we have $\underline{\phi}^{(1)}(1) = \overline{\phi}^{(1)}(1) = \phi(1) = 0$. After one step we obtain the same results as full closed testing, with $q^{(1)} = q = 1$. Then, similarly to Theorem 1.7, at each step n we may approximate d from below with

$$d^{(n)} = s - q^{(n)}.$$

Theorem 1.11. *$d^{(n)} \le d^{(n+1)} \le d^{(m)} = d$ for each $n \in \mathbb{N}$.*

Proposition 1.10 is the basis of the iterative shortcut. At any step n and for any z, it allows to make statements on the value of $\phi(z)$ by applying the single-step shortcut within at most 2^n subspaces. Then Theorem 1.11 gives lower $(1-\alpha)$-confidence bounds for the number of true discoveries δ. Even if the iterative shortcut is stopped early, before reaching convergence, $d^{(n)}$ is always a valid lower confidence bound; we have increasingly better approximations of d as n increases, and obtain full closed testing results after at most m steps. As the Theorem may be applied to any $S \subseteq M$, the procedure $d^{(n)}$ has true discovery guarantee, meaning that $d^{(n)}(S)$ are simultaneous lower $(1-\alpha)$-confidence bounds for $\delta(S)$.

In Section 1.11 we provide an algorithm for the iterative shortcut, where the complexity of each iteration is linearithmic both in the number of hypotheses and the number of permutations. It converges to full closed testing results after a number of iterations that is exponential in the number of hypotheses in the worst case.

1.8 Truncation

As shown in Section 1.11, in the worst case the single-step shortcut requires a number of operations of order $m \log^2(m)$; the iterative shortcut converges to closed testing after a number of iterations exponential in m, where the complexity of each iteration is linearithmic in m. In this section, we argue that this complexity is much reduced if the method is applied to truncated statistics, as it allows to shrink the effective total number of hypotheses from m to $m' \in \{s, \dots, m\}$. In practice, with large B, m' is obtained by taking all statistics in S, and only the non-truncated observed statistics in $M \setminus S$.

Truncation-based statistics were advocated in the truncation product method of Zaykin *et al.* (2002), in the context of p-value combinations. The main idea was to emphasize smaller p-values, which are more likely to correspond to false hypotheses, by taking into account only p-values smaller than a certain threshold, and setting to 1 the others; a natural, common choice for the threshold is the significance level α. A similar procedure, the rank truncation product (Dudbridge and Koeleman, 2003; Kuo and Zaykin, 2011), takes into account only the k smallest p-values, for a given k. Eventually, weights can be incorporated into both analyses. Such procedures provide an increased power in many scenarios, and in particular for signal detection, when there is a predominance of near-null effects. They have been widely applied in literature (Yu *et al.*, 2009; Li and Tseng, 2011; Biernacka *et al.*, 2012; Dai *et al.*, 2014); refer to Zaykin *et al.* (2007) and Finos (2003) for a review of the methods and their applications.

TABLE 1.4: Toy example with $S = \{1, 2\}$: test statistics after truncation of elements smaller than $t^* = 2$, and after dimensionality reduction.

	truncated $f(T_i^\pi)$					dim. reduction		
	H_1	H_2	H_3	H_4	H_5	H_1	H_2	$H_{4,5}$
id	6	5	4	0	0	6	5	0
π_2	0	2	0	0	4	0	2	4
π_3	8	3	0	2	0	8	3	2
π_4	8	0	0	0	0	8	0	0
π_5	0	6	0	0	2	0	6	2
π_6	7	0	0	2	0	7	0	2

With our notation, we can define a truncation-based statistic for H_S as following. For each hypothesis H_i, we set to a common ground value t^0 all statistics T_i^π smaller than a threshold t_i^*. The threshold t_i^* may depend on i, or be a prefixed value, or be the k-th greatest statistic T_i^π ($i \in M$, $\pi \in \boldsymbol{\pi}$) for a given k. The ground value must be $t^0 \leq \min_i t_i^*$; it may be chosen, for instance, as the minimum possible value of the test statistics, or set equal to the smallest threshold. Then

$$T_S = \sum_{i \in S} f_i(T_i), \qquad f_i(T_i) = t^0 \cdot \mathbf{1}\{T_i < t_i^*\} + T_i \cdot \mathbf{1}\{T_i \geq t_i^*\}.$$

For simplicity of notation, let $t_i^* = t^*$ be independent of i, and $t^0 = 0$, so that

$$T_S = \sum_{i \in S} f(T_i), \qquad f(T_i) = T_i \cdot \mathbf{1}\{T_i \geq t^*\}.$$

Table 1.4 shows the statistics $f(T_i)$ in the toy example after truncation with $t^* = 2$. Here, t^* is set as the k-th greatest statistic, where $k = \lceil Bm\alpha \rceil$ is chosen so that the proportion of non-null contributions $f(T_i^\pi)$ is approximately α. Observe that H_3 is such that the observed truncated statistic is the greatest over all permutations, i.e., $f(T_3) = \max_\pi f(T_3^\pi)$; as a consequence, adding $\{3\}$ to any set V can only increase the number of rejections. On the contrary, H_4 and H_5 are such that the observed statistics are the smallest over all permutations, and so adding $\{4\}$ or $\{5\}$ to any set can only decrease rejections. Truncation makes those two particular cases more common as well as easier to check, through the following conditions:

$$f(T_i^\pi) = 0 \quad \text{for all } \pi \in (\boldsymbol{\pi} \setminus \{\text{id}\}) \tag{1.24}$$

$$f(T_i) = 0 \tag{1.25}$$

Proposition 1.12. *Let $V \subseteq M$ and $i \in M$. If i satisfies condition (1.24), then $V \in \mathcal{R}$ implies $(V \cup \{i\}) \in \mathcal{R}$. If i satisfies condition (1.25), then $(V \cup \{i\}) \in \mathcal{R}$ implies $V \in \mathcal{R}$.*

The shortcut examines the collection $\mathcal{V}_z$ of sets that have at least size z overlap with S, searching for a set $V \notin \mathcal{R}$. In this case, the focus is on the number of indices in S, hence we may reduce the dimensionality of the problem by applying Proposition 1.12 to the remaining indices. If an index $i \in M \setminus S$ satisfies condition (1.24), then it is not useful for finding a non-rejected set, and so can be removed from M. If two indices $i, j \in M \setminus S$ satisfy condition (1.25), they may be collapsed into a new index h, so that $H_h = H_{\{i,j\}}$ can only decrease the number of rejections. This allows to reduce the total number of hypotheses from m for computational purposes to a substantially lower $m' \in \{s, \dots, m\}$. In the toy example column 3 is removed, while columns 4 and 5 are collapsed into a single column, reducing the number of hypotheses from $m = 5$ to $m' = 3$.

1.9 Applications

In this section, we use the iterative shortcut of Section 1.7 to analyze simulated and real data. For both analyses we use the `sumSome` package (Vesely, 2021b) developed in `R` (R Core Team, 2017), with underlying code in `C++`.

1.9.1 Simulations

In this section, we use the shortcut to compare the performance of different p-value combinations through simulations. When using p-value combinations, the unknown joint distribution of the data is often managed through worst-case distributions, defined either generally or under restrictive assumptions (Vovk and Wang, 2020; Tian *et al.*, 2021). However, this approach makes comparisons difficult, since different tests have different worst cases. In contrast, our method adapts to the unknown distribution through permutations, and thus allows to compare the tests on equal footing. Determining which test has the highest power in different settings is a major issue, for which a full treatment is out of the scope of the manuscript; we present a first exploration.

We simulate n independent observations from a multivariate normal distribution with m variables. We assume the model $\mathbf{Y} = \boldsymbol{\mu} + \boldsymbol{\varepsilon}$, with $\mathbf{Y}, \boldsymbol{\mu}, \boldsymbol{\varepsilon} \in \mathbb{R}^m$. The signal $\boldsymbol{\mu}$ is non-null for a proportion a of variables, and its value is computed so that the two-sided one-sample t-test with significance level α has a given power β. The noise is $\boldsymbol{\varepsilon} \sim \mathcal{N}_m(\mathbf{0}, \Sigma_{\rho^2})$, where ρ is the level of equi-correlation between pairs of variables.

From the resulting data, we obtain p-values by applying a two-sided one-sample t-test for each variable $i \in \{1, \dots, m\}$, with null hypothesis $H_i : \mu_i \neq 0$. P-values are computed for B random permutations. Moreover, we employ truncation by setting to a common ground value t^0 any p-value greater than a threshold t^*.

We analyze the subset S of false hypotheses (active variables), and the complementary subset $M \setminus S$ of true hypotheses (inactive variables), by means of different p-value combinations: Pearson (1933), Liptak (1958), Cauchy (Liu and Xie, 2020), and generalized means with parameter $r \in \{-2, -1, -0.5, 0, 1, 2\}$ (Vovk and Wang, 2020). The latter will be denoted by VW(r). Notice that VW(0) corresponds to Fisher (1925), and VW(1) to Edgington (1972). As a comparison, we also apply the maxT-method of Westfall and Young (1993), corresponding to the limit of VW(r) when r tends to $-\infty$; we apply the usual algorithm for the maxT.

We fix $n = 50$, $m = 1000$, $\alpha = 0.05$, $B = 200$ and $t^0 = 0.5$, then we consider $a \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 0.9\}$, $\beta \in \{0.5, 0.8, 0.95\}$, $\rho \in \{0, 0.3, 0.6, 0.9\}$, and $t^* \in \{0.005, 0.01, 0.05, 0.1, 1\}$, where $t^* = 1$ leads to no truncation. For each setting, we simulate data 1000 times, and compute the TDP lower confidence bound for the set S as the mean of $d(S)/s$ over the simulations. Furthermore, we compute the FWER as the proportion of simulations where $d(M \setminus S) > 0$, meaning that the method finds at least one discovery among the true hypotheses. The algorithm is run until convergence.

Figure 1.4 shows the average TDP lower confidence bounds obtained in different scenarios. Results are shown only for $\beta = 0.95$ and $t^* \in \{0.005, 0.05, 1\}$. Moreover, certain groups of tests have similar performances: (1) VW(1), VW(2) and Pearson; (2) VW(-1) and Cauchy. For clarity, among these tests, only VW(1) and VW(-1) are displayed in the plots. Results indicate that the intensity of the signal, determined by the parameter β, does not significantly affect the behavior of the tests; nevertheless, differences between tests are amplified when the signal is high. Furthermore, results suggest that truncation is generally advisable, unless the signal is very dense, i.e., a is high. Indeed, in most cases tests tend to be more powerful when t^* is low, and thus more statistics are truncated; the improvement is stronger for sparse signal, and when considering VW(0), VW(1) and Liptak.

When the signal is sparse, VW(r) with $r < 0$ performs best; the most powerful test is VW(-1) for low correlation, and VW(-2) for high correlation. The remaining tests perform well when the signal is dense; among those, in the considered scenarios VW(0) is the most powerful, but the powers of these tests become more and more similar as the signal becomes denser. These results confirm that when the individual contributions, i.e., the transformed p-values, have heavy-tailed distributions, the test is more directed

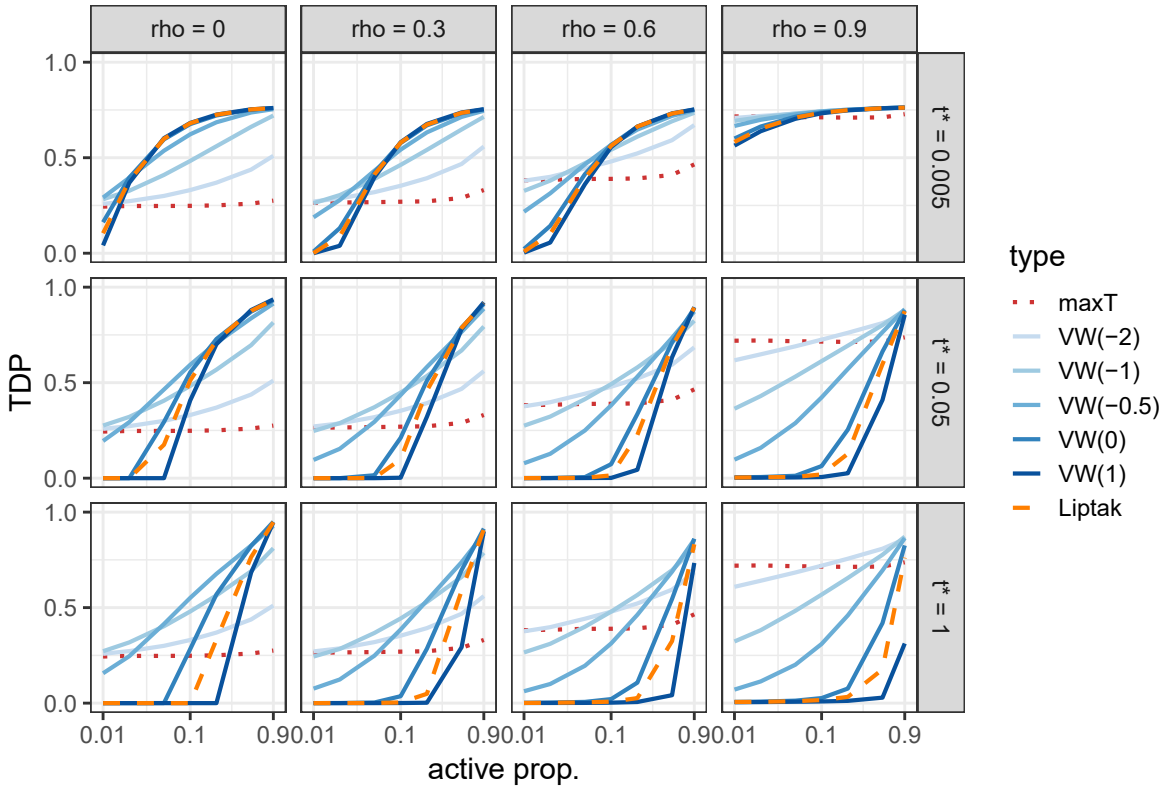

FIGURE 1.4: Simulated data: TDP lower confidence bounds for the set S of active variables, by active proportion a (log scale) and for different p-value combinations. Variables have equi-correlation ρ. P-values smaller than t^* are truncated.

towards sparse alternatives; on the contrary, when distributions are not heavy-tailed, the test is more directed towards dense alternatives (Vovk and Wang, 2020).

The computation time is less than a minute in most scenarios, with a maximum of around 5 minutes. Finally, simulations confirm that the method controls the FWER. Plots for both the computation time and the FWER are provided in Section 1.12.1.

Finally, we focus on the harmonic mean VW(-1) to give an example of a comparison between the shortcut and closed testing based on worst-case distributions. For the latter method we rely on the procedure of Tian *et al.* (2021). Figure 1.5 shows the ratio of the TDP determined by the shortcut and the TDP given by the second method. As expected, worst-case distributions tend to be very conservative, and so the corresponding method results to be less powerful than our permutation approach in all scenarios. The difference in power is considerably stronger in settings with low signal and low correlation, where the ratio reaches a maximum of around 7.8. In the worst case, where $\rho = 0.9$, $\beta = 0.95$ and $a = 0.9$, the shortcut's TDP is 1.3 times the TDP computed from worst-case distributions.

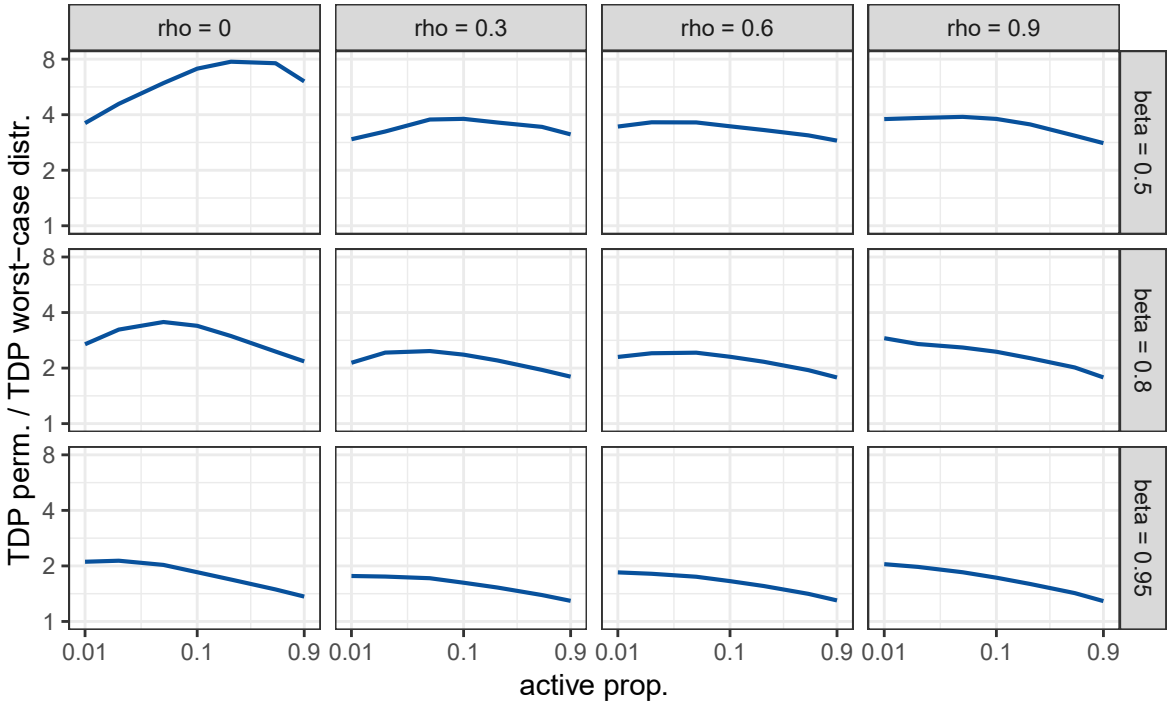

FIGURE 1.5: Simulated data, VW(-1): ratio (log scale) between the TDP lower confidence bounds for the set S of active variables given by the permutation-based shortcut and by closed testing based on worst-case distributions. Results are plotted by active proportion a (log scale). Variables have equi-correlation ρ, and the signal increases with the parameter β.

1.9.2 fMRI data

We apply the shortcut to fMRI brain imaging data, demonstrating feasibility of the method on large datasets, adaptation to the correlation structure and post-hoc flexibility. In fMRI imaging, Blood Oxygen Level Dependent (BOLD) response is measured, i.e., changes in blood flow in the brain induced by a sequence of stimuli, at the level of small volume units called voxels. Brain activation is then inferred as correlation between the stimuli and the BOLD response. Researchers are interested in studying this activation within different clusters, brain regions of connected voxels.

Typically, voxels are highly correlated. This is usually taken into account by means of cluster extent thresholding (Nichols, 2012; Woo *et al.*, 2014; Rosenblatt *et al.*, 2018). However, when the method finds activation in a given cluster, it only indicates that the cluster contains at least one active voxel, but does not provide any information on the proportion of active voxels (TDP) nor their spatial location. This leads to the spatial specificity paradox, the counter-intuitive property that activation in a large cluster is a weaker finding than in a small cluster (Woo *et al.*, 2014). Moreover, follow-up

inference inside a cluster leads to inflated Type I error rates (Kriegeskorte *et al.*, 2009). In contrast, our approach not only adapts to the high correlation, but also provides confidence sets for the TDP, and allows for post-hoc selection and follow-up inference inside clusters. In the following sections, we study two different datasets.

1.9.2.1 Auditory data

We analyze data collected by Pernet *et al.* (2015), available at `https://openneuro.org/datasets/ds000158/versions/1.0.0`, which compares subjects examined while listening to vocal and non-vocal sounds. Data consists of brain images for 140 subjects, each composed of 168,211 voxels. As for any standard fMRI analysis (Lindquist, 2008), as first-level analysis for each subject we estimate the contrast map that describes the difference in activation during vocal and non-vocal stimuli, with the same procedure of Andreella *et al.* (2020). Then these contrast maps are used to run the second-level analysis; for each voxel we compute a test statistic by means of a two-sided one-sample t-test, with the null hypothesis that the voxel's mean contrast between subjects is zero. Finally, we define the global test statistic for a cluster as the sum of its voxels' t-statistics.

We examine supra-threshold clusters with threshold 3.2, and then we make follow-up inference inside those by studying clusters with threshold 4. The significance level is taken as $\alpha = 0.05$. We construct statistics for the permutation test by using B elements from the group of sign-flipping transformations, which satisfies Assumption 1.1 (Winkler *et al.*, 2014). Moreover, we employ truncation as in Section 1.8 by setting to $t^0 = 0$ any statistic smaller than $t^* = 3.2$; this way, we take into account only statistics at least as extreme as the cluster-defining threshold. We use two settings. First, we apply a 'quick' analysis, fast and feasible on a standard machine, by using $B = 200$ transformations and stopping after 50 iterations of the single-step shortcut. Subsequently, we consider a 'long' analysis, run on the platform CAPRI (University of Padova, 2017), that employs $B = 1000$ transformations and stops after 1000 iterations. Computation time for the 'quick' setting is less than 10 minutes on a standard PC, while the 'long' setting requires around 10 hours for clusters with threshold 3.2, and 27 hours for follow-up inference on clusters with threshold 4.

Results are shown in Table 1.5, which contains the lower confidence bound for the TDP of each cluster, as well as the size, the FWER-corrected p-value based on random field theory (RFT), and the coordinates of the maximum t-statistic; as RFT does not allow for follow-up inference, p-values are computed only for clusters with threshold 3.2. Moroever, Figure 1.6 contains the map of the TDP lower confidence bounds obtained from the 'quick' setting. Our method finds activation in concordance with

TABLE 1.5: Auditory data: analysis of supra-threshold clusters with thresholds 3.2 and 4. Clusters with no discoveries are not shown.

cluster S	threshold thr	size s	TDP $d(S)/s$ quick	TDP $d(S)/s$ long	RFT p-value p_{FWER}	coordinates x	y	z
FP/CG/SFG/TOF/LO/LG OFG/ITG/SG/AG/NA	3.2	40094	98.21%	98.07%	< 0.0001	-30	-34	-16
Left LO/TOF	4	8983	94.79%	93.86%	-	-30	-34	-16
Right LO/LG/ITG	4	7653	93.85%	92.73%	-	28	-30	-18
Left SFG/FP	4	1523	69.67%	65.92%	-	-28	34	42
CG	4	1341	65.62%	61.45%	-	6	40	-2
Right FP	4	1327	66.01%	61.34%	-	30	56	28
Left SG/AG	4	859	47.85%	41.09%	-	-50	-56	36
Right STG/PT/MTG HG/PrG/T	3.2	12540	95.41%	94.76%	< 0.0001	60	-10	0
STG/PT/MTG/HG	4	9533	95.17%	94.59%	-	60	-10	0
PrG	4	485	25.15%	17.94%	-	52	0	48
Left STG/PT/MTG/ HG/IFG/T	3.2	10833	94.66%	93.93%	< 0.0001	-60	-12	2
HG/PT/MTG/STG	4	7894	94.20%	93.46%	-	-60	-12	2
IFG	4	667	38.98%	32.53%	-	-40	14	26

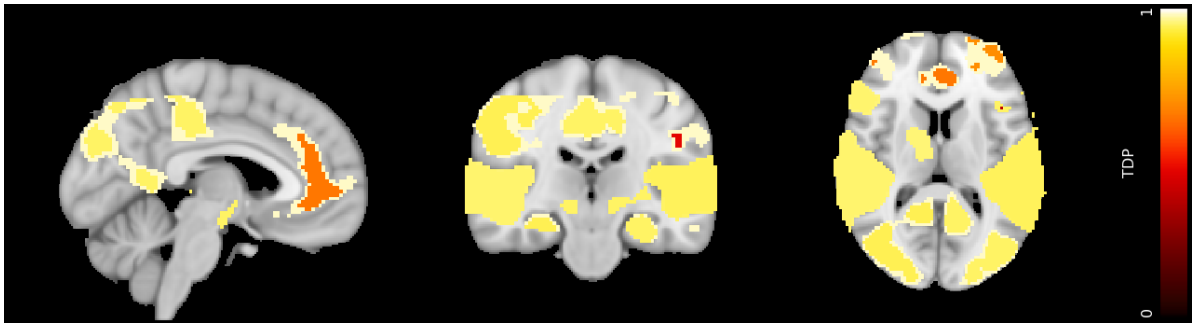

FIGURE 1.6: Auditory data: map of the TDP lower confidence bounds for supra-threshold clusters with thresholds 3.2 and 4.

previous studies. An extensive comparison with other methods is beyond the scope of this manuscript, however our results can be immediately compared to those in Andreella *et al.* (2020), since the same data was used. Notice that this dataset is characterized by a strong signal, for example leading to a cluster with around 40,000 voxels (24% of the brain) among which more than 98% are active. A dataset with more subtle signal is studied in the following section.

Results indicate that the setting of the 'long' analysis did not provide larger TDP values than the 'quick'. While in this particular case 'quick' calculation settings tend to give slightly better results, the difference is dominated by the variability due to the random permutations. In Section 1.12.2 we further investigate the role of the number of iterations and permutations, confirming that the 'quick' setting provides suitable power.

1.9.2.2 Arrow data

We analyze data collected by Kelly *et al.* (2008) and available at `https://openneuro.org/datasets/ds000102/versions/00001`, where subjects were examined while performing a slow event-related Eriksen Flanker task. Data consists of brain images for 26 subjects, each composed of 252,833 voxels.

The analysis is carried out as in the 'quick' setting of Section 1.9.2.1, with the same definition of the test statistics, truncation and supra-threshold clusters. Computation time on a standard PC is around 7 minutes for clusters with threshold 3.2, and 8 minutes for follow-up inference on clusters with threshold 4.

When considering the entire brain, the analysis determines that 8.22% of all voxels are active. Regarding cluster analysis, it finds activation only in two clusters with threshold 3.2: (1) PoG/PrG/SPL/LOC, with 9,752 voxels and 35.77% discoveries; (2) LOC/SPL/OFG/OFG with 1,493 voxels and 19.26% discoveries.

1.10 Discussion

We have proposed a new perspective on the age-old subject of global testing, arguing that all global tests automatically come with an inbuilt selective inference method, allowing many additional inferences to be made without paying a price in terms of the global test's α-level. Our proposed approach provides not just p-values but gives a confidence bound for the TDP, which is considerably more informative; indeed, reporting a p-value only infers the presence of some discoveries, while the TDP allows to quantify the proportion of these discoveries. Moreover, such TDP confidence bounds come not just for the full testing problem, but also simultaneously for all subsets of hypotheses; this way, subsets of interest may be chosen post hoc, without compromising the validity of the method. Examples of methods that make this type of inference on the TDP are Rosenblatt *et al.* (2018) and Ebrahimpoor *et al.* (2020), which allow to analyze brain imaging and genomics data, respectively.

To construct simultaneous confidence bounds for the TDP of all subsets, we have provided a general closed testing procedure for sum tests, a broad class of global tests that includes many p-value combinations and other popular multiple testing methods. The procedure uses permutation testing to adapt to the unknown joint distribution of the data, avoiding strong assumptions or potential loss of power due to worst-case distributions.

We have presented an iterative shortcut for this procedure, where the complexity of each iteration is linearithmic both in the number of hypotheses and in the number

of permutations. It converges to full closed testing results after a finite, but possibly exponential, number of iterations; furthermore, it may be stopped at any time while still providing control of the TDP. As shown in simulations, when studying 1000 hypotheses, in most cases the procedure converges to closed testing in seconds. Moreover, the method is feasible in high-dimensional settings, as shown in two applications on fMRI data, where the analysis required less than 15 minutes on a standard PC. An implementation is available in the `sumSome` package (Vesely, 2021b) in `R`, with underlying code in `C++`.

Our method is extremely flexible, allowing any sum test of choice; different choices of the sum test have very different power properties, as we have illustrated. More research is needed on the performance of different sum tests in different scenarios. Notice that the test statistic, including the eventual truncation, needs to be chosen a priori, before performing the analysis. Moreover, permutations are known to have a better performance than worst-case distributions under general dependence structure, but we have not performed a systematic investigation to quantify the improvement given by permutations in the case of sum tests. Finally, these sum tests procedures may be compared with other permutation-based procedures that rely on bounding functions (Andreella *et al.*, 2020; Blanchard *et al.*, 2020). However, these comparisons are beyond the scope of this manuscript.

1.11 Appendix: Algorithmic implementation

In this section we provide an outline and pseudocode for the full method. In Section 1.11.1 we give the algorithms that evaluate $\phi(z)$ for a value z through the single-step and the iterative shortcut; then in Section 1.11.2 we approximate q by embedding these algorithms into a binary search method. Moreover, in Section 1.11.3 we show how the method can be employed to find the largest subset with a given TDP. Finally, in Section 1.11.4 we give an improved version of the algorithms of Section 1.11.1.

1.11.1 Algorithms for the shortcut

Algorithm 1 implements the single-step shortcut of Propositions 1.6 and 1.9, evaluating $\phi(z)$ for any given z by constructing the bound ℓ_z and the path u_z. Recall that $\ell_z(v) \leq u_z(v)$ for each size v. By Lemmas 1.5 and 1.8, we have $\phi(z) = 1$ if ℓ_z is entirely positive, and $\phi(z) = 0$ if u_z is not; in the intermediate case where u_z lays in the positive half-space but ℓ_z does not, the value of $\phi(z)$ remains unsure.

Algorithm 1: Single-step shortcut to evaluate $\phi(z)$ for any z.

Data: $z \in \{0, \dots, s+1\}$
Result: $\phi(z)$ (0, 1 or unsure)
if $z = 0$ **then return** 0;
if $z = s + 1$ **then return** 1;
Unsures = FALSE;
for $v = z, \dots, m$ **do**
  compute $\ell_z(v)$ as in (1.11);
  if $\ell_z(v) \leq 0$ **then**
    Unsures = TRUE;
    compute $u_z(v)$ as in (1.18);
    if $u_z(v) \leq 0$ **then return** 0;
  end
end
if *Unsures* **then return** unsure;
return 1;

From the result, we may obtain $\underline{\phi}(z)$ and $\overline{\phi}(z)$ as in (1.12) and (1.19), respectively. We do this by taking $\underline{\phi}(z) = 1$ if and only if the algorithm returns $\phi(z) = 1$, and $\overline{\phi}(z) = 0$ if and only if it returns $\phi(z) = 0$.

The procedure requires to evaluate at most $2(m - z + 1)$ tests, but this number may be smaller. As shown in the following lemma, the worst-case complexity is linearithmic both in the number m of hypotheses and in the number B of permutations. Recall that the choice of B does not depend on m or s.

Lemma 1.13. *In the worst case, the computational complexity of Algorithm 1 is of order* $mB\log(mB)$.

Subsequently, Algorithm 2 implements the iterative shortcut of Proposition 1.10, embedding Algorithm 1 into a branch and bound method. First, we apply the single-step shortcut on $\mathcal{V}_z$; if it returns an unsure outcome, $\mathcal{V}_z$ is partitioned into $\mathcal{V}_z^-$ and $\mathcal{V}_z^+$ as in Section 1.7.1. Since $\mathcal{V}_z^-$ does not include the index j^* of the hypothesis that we believe we have most evidence against, it appears more likely to contain a non-rejected set. With this reasoning, the subspaces are analyzed by means of a depth-first search, meaning that the algorithm starts by exploring $\mathcal{V}_z^-$, and explores as far as needed along the branch where indices are removed. The user can set a maximum number $h_{\max}$ of iterations, where each iteration represents the analysis of a subspace by means of the single-step shortcut. The procedure stops when it converges to closed testing results or when the number of iterations reaches $h_{\max}$.

Algorithm 2: Iterative shortcut to evaluate $\phi(z)$ for any z.

Data: $z \in \{0, \dots, s+1\}$; $h_{\max} \in \mathbb{N}$ (maximum number of iterations)
Result: $\phi(z)$ (0, 1 or unsure)
$X = \mathcal{V}_z$;
shortcut on X from Algorithm 1;
if *shortcut returns 0* **then return** 0;
if *shortcut returns 1* **then return** 1;
$h = 0$;
Stack = empty list;
while $h < h_{max}$ **do**
  while *shortcut returns unsure* **and** $h < h_{max}$ **do**
    $++h$;
    partition X into $\mathcal{V}_z^-$ and $\mathcal{V}_z^+$ as in Section 1.7.1;
    add $\mathcal{V}_z^+$ to Stack;
    $X = \mathcal{V}_z^-$;
    shortcut on X from Algorithm 1;
    if *shortcut returns 0* **then return** 0;
  end
  while *Stack is not empty* **and** *shortcut returns 1* **and** $h < h_{max}$ **do**
    $++h$;
    X = last element added in Stack;
    remove last element from Stack;
    shortcut on X from Algorithm 1;
    if *shortcut returns 0* **then return** 0;
  end
end
if *Stack is empty* **and** *shortcut returns 1* **then return** 1;
return unsure

Similarly to the single-step shortcut, we obtain $\underline{\phi}^{(n)}(z)$ and $\overline{\phi}^{(n)}(z)$ by taking $\underline{\phi}^{(n)}(z) = 1$ if and only if the algorithm returns $\phi(z) = 1$, and $\overline{\phi}^{(n)}(z) = 0$ if and only if it returns $\phi(z) = 0$.

Each iteration of the algorithm applies the single-step shortcut of Algorithm 1 in a subspace, with complexity that is linearithmic in the number of hypotheses and the number of permutations (Lemma 1.13). The following lemma shows that the total complexity may be exponential in the number of hypotheses. In many cases this number is lower; moreover, by Theorem 1.11 we obtain a valid lower $(1 - \alpha)$-confidence bound for δ even if the algorithm is stopped early.

Lemma 1.14. *In the worst case, Algorithm 2 converges after a number of iterations of order 2^m, where each iteration has complexity of order $mB \log(mB)$.*

1.11.2 Binary search method

Algorithms 1 and 2 evaluate $\phi(z)$ for a single fixed z, but we are interested in the change point q of the function $\phi : \{0, \dots, s+1\} \to \{0, 1\}$, given in (1.3). In this section, we show how to approximate q without studying all values $z \in \{0, \dots, s+1\}$.

By Lemma 1.4, ϕ has a single change point in q, and takes opposite values in the extremes of its domain. Therefore q may be found by embedding the shortcut within a binary search algorithm (Knuth, 1998). The procedure consists of iteratively bisecting the domain of ϕ and selecting the subset that must contain the change point, based on the values that the function takes in the extremes and the bisection point. In the worst case, q is determined after a number of steps of order $\log_2(s)$. The following lemma shows that combining the single-step shortcut with a binary search has complexity at most of order $m \log^2(m)$ in the number m of hypotheses, and linearithmic in the number of permutations. The worst-case complexity of using the iterative shortcut of Algorithm 2 remains exponential, as in Lemma 1.14.

Lemma 1.15. *In the worst case, embedding Algorithm 1 into a binary search requires a number of operations of order $mB \log(m) \log(mB)$.*

Even if stopped early, this procedure provides an approximation from above of q, and thus a valid lower $(1 - \alpha)$-confidence bound for the number of true discoveries δ. For instance, if we stop after finding that $q \in \{z_1, \dots, z_2\}$, we know that $q \leq z_2$, and so $d \geq s - z_2$. As a result, $s - z_2$ is a lower $(1 - \alpha)$-confidence bound for δ.

1.11.3 Largest subset with given TDP

In this section we show how the shortcut can be used to study incremental sets in any desired ordering, and quickly determine the largest set having TDP at least equal to a given value γ. This is achieved by combining the binary search of Section 1.11.2 with an algorithm of Tian *et al.* (2021).

In the toy example, suppose we want to study the incremental sets $S_1 = \{1\}$, $S_2 = \{1, 2\}$, $S_3 = \{1, 2, 3\}$, $S_4 = \{1, 2, 3, 4\}$ and $S_5 = M$, with the aim of finding the largest one having TDP lower confidence bound at least equal to 0.5. This means finding the greatest size s such that $d(S_s)/s \geq 0.5$; the values $d(S_s)$ can be computed by means of the binary search.

In general, let $S_1 \subset \ldots \subset S_m = M$ be a collection of incremental sets, with size $|S_s| = s$ for each s, and fix $\gamma \in [0, 1]$. Then Algorithm 3 finds the greatest size s with $d(S_s)/s \geq \gamma$. From the result, we know that the TDP of the selected set is at least γ with confidence $1 - \alpha$.

Algorithm 3: Procedure to study a collection of incremental sets, and determine the size of the largest set with TDP lower confidence bound at least equal to γ.

Data: $S_1 \subseteq \ldots \subseteq S_m$ (incremental sets with $|S_s| = s$ for each s); $\gamma \in [0, 1]$
Result: $\max\{s \in \{1, \ldots, m\} : d(S_s)/s \geq \gamma\}$ if the maximum exists, 0 otherwise
if $\gamma = 0$ **then return** m;
s = m;
while $s > 0$ **do**
  binary search on S_s to compute $d(S_s)$;
  if $d(S_s)/s \geq \gamma$ **then return** s;
  $s = \lfloor d(S_s)/\gamma \rfloor$;
end
return 0

1.11.4 Algorithms for the shortcut with reduced complexity

In this section, we provide improved versions of Algorithms 1 and 2, which require fewer computations. Algorithms 4 and 5 evaluate $\phi(z)$ for any z, using the single-step shortcut of Propositions 1.6 and 1.9 and the iterative shortcut of Proposition 1.10, respectively. The number of computations is reduced by exploiting some properties of the bound and the path.

The structure of Algorithm 4 is constructed so that it will be useful for Algorithm 5. The user can define for which sizes $v \in \{v_1, \ldots, v_2\} \subseteq \{z, \ldots, m\}$ they need to check whether $\mathcal{V}_z(v) \subseteq \mathcal{R}$ (see partition (1.7)), supposing this is already known to be true

Algorithm 4: Single-step shortcut to evaluate $\phi(z)$ for any z (algorithm with reduced complexity).

Data: $z \in \{0, \dots, s+1\}$; $v_1, v_2 \in \{z, \dots, m\}$ (smallest and greatest sizes to check); get_path (TRUE to compute the path)
Result: $\phi(z)$ (0, 1 or unsure); if $\phi(z)$ is unsure, v_1 and v_2 are updated
if $z = 0$ **then return** 0;
if $z = s + 1$ **then return** 1;
compute c_1 and c_2 as in (1.26) and (1.27);
Unsures = empty list;
for $v = c_1, c_1 - 1 \dots, v_1$ **do**
  compute $\ell_z(v)$ as in (1.11);
  if $\ell_z(v) > 0$ **then break**;
  add v to Unsures;
  if *get_path* **then**
    compute $u_z(v)$ as in (1.18);
    if $u_z(v) \leq 0$ **then return** 0;
  end
end
for $v = v_1 + 1, v_1 + 2, \dots, v_2$ **do**
  compute $\ell_z(v)$ as in (1.11);
  if $\ell_z(v) > 0$ **and** $v \geq c_2$ **then break**;
  else if $\ell_z(v) \leq 0$ **then**
    add v to Unsures;
    if *get_path* **then**
      compute $u_z(v)$ as in (1.18);
      if $u_z(v) \leq 0$ **then return** 0;
    end
end
if *Unsures is empty* **then return** 1;
update $v_1 = \min$ Unsures and $v_2 = \max$ Unsures;
return unsure;

for the remaining sizes. The algorithm not only evaluates $\phi(z)$, but also updates the values of v_1 and v_2, keeping track of the new set, possibly empty, of sizes that need to be further examined. Moreover, the user can state whether they already know that the path u_z is entirely positive; in this case, it is not computed.

We reduce the number of computations needed by Algorithm 4 as follows. In Lemma 1.16 we will show that there exist $c_1, c_2 \in \{z, \dots, m\}$ such that the bound $\ell_z(v)$ is decreasing for $v \leq c_1$, and increasing for $v \geq c_2$. Since the single-step shortcut only uses the sign of $\min_v \ell_z(v)$ (Lemma 1.5), it is not always necessary to compute $\ell_z(v)$ for all sizes v. For instance, in the toy example with $z = 1$ (Figure 1.1, left), we have $c_1 = c_2 = 2$; since $\ell_1(4) > 0$ and $4 \geq c_2$, we know that $\ell_1(5) > 0$ without computing it.

We find c_1 and c_2 as following. Consider the toy example with $z = 1$, and recall

Algorithm 5: Iterative shortcut to evaluate $\phi(z)$ for any z (algorithm with reduced complexity).

Data: $z \in \{0, \ldots, s+1\}$; $h_{\max}$ (maximum number of iterations)
Result: $\phi(z)$ (0, 1 or unsure)
$X = \mathcal{V}_z$;
$v_1 = z$; $v_2 = m$;
get_path = TRUE;
shortcut on X from Algorithm 4;
if *shortcut returns 0* **then return** 0;
if *shortcut returns 1* **then return** 1;
$h = 0$;
Stack = empty list;
while $h < h_{max}$ **do**
  get_path = FALSE;
  while *shortcut returns unsure* **and** $h < h_{max}$ **do**
    $++h$;
    partition X into $\mathcal{V}_z^-$ and $\mathcal{V}_z^+$ as in Section 1.7.1;
    add ($\mathcal{V}_z^+$, v_1, v_2) to Stack;
    $X = \mathcal{V}_z^-$;
    shortcut on X from Algorithm 4;
    if *shortcut returns 0* **then return** 0;
  end
  get_path = TRUE;
  while *Stack is not empty* **and** *shortcut returns 1* **and** $h < h_{max}$ **do**
    $++h$;
    (X, v_1, v_2) = last element added in Stack;
    remove last element from Stack;
    shortcut on X from Algorithm 4;
    if *shortcut returns 0* **then return** 0;
  end
end
if *Stack is empty* **and** *shortcut returns 1* **then return** 1;
return unsure

definition (1.11). The value $\ell_1(v)$ is computed by summing by row the first v columns of Table 1.2, and then taking the quantile. Note that the elements of column 3 are all non-negative; as a consequence, $b_2^\pi \leq b_3^\pi$ for each π, and so $\ell_1(2) \leq \ell_1(3)$. Moreover, since the statistics are sorted so that $C^\pi_{j_1(\pi)} \leq \ldots \leq C^\pi_{j_4(\pi)}$ for each π, all elements of columns 4 and 5 must be non-negative, and so $\ell_1(2) \leq \ldots \leq \ell_1(5)$. A similar argument may be used to show that, since all elements of column 2 are non-positive, $\ell_1(1) \geq \ell_1(2)$.

In general, fix $z \in \{1, \ldots, s\}$, and consider the statistics $C^\pi_{j_h(\pi)}$, with $h \in \{1, \ldots, m - z\}$, that remain after selecting the z smallest statistics in S for each π. We identify the

last column where these statistics are all non-positive, taking

$$c_1 = z + \max\left\{h \in \{1, \dots, m-z\} \,:\, C^{\pi}_{j_h(\pi)} \leq 0 \text{ for all } \pi \in \boldsymbol{\pi}\right\} \tag{1.26}$$

if the maximum exists, and $c_1 = z$ otherwise. Similarly, we identify the last column before these statistics become all non-negative, taking

$$c_2 = z + \max\left\{h \in \{1, \dots, m-z\} \,:\, C^{\pi}_{j_h(\pi)} < 0 \text{ for some } \pi \in \boldsymbol{\pi}\right\} \tag{1.27}$$

if the maximum exists, and $c_2 = z$ otherwise.

Lemma 1.16. *Define c_1 as in (1.26), and c_2 as in (1.27). Then $\ell_z(z) \geq \ell_z(z+1) \geq \ldots \geq \ell_z(c_1)$, and $\ell_z(c_2) \leq \ell_z(c_2+1) \ldots \leq \ell_z(m)$.*

Subsequently, Algorithm 5 embeds Algorithm 4 within a branch and bound method. With respect to Algorithm 2, the number of computations is reduced as following. First, Lemma 1.17 shows that it is not necessary to compute the path u_z when we apply the single-step shortcut within $\mathcal{V}_z^-$; where it is defined, it coincides with the path in $\mathcal{V}_z$ (e.g., compare Figures 1.2 and 1.3). The same argument applies any time an index is removed.

Lemma 1.17. *$u_z(v)$ is the same in $\mathcal{V}_z$ and $\mathcal{V}_z^-$ for each $v \in \{z, \dots, m-1\}$.*

Moreover, when studying subspaces it is only necessary to further examine those sizes v for which we are unsure whether $\mathcal{V}_z(v) \subseteq \mathcal{R}$. For instance, in the toy example with $z = 1$, the single-step shortcut gives $\ell_1(4)$, $\ell_1(5) > 0$ (Figure 1.2, left), and so we only need to further examine $v \in \{1, 2, 3\}$ when applying the single-step shortcut within $\mathcal{V}_1^-$ and $\mathcal{V}_1^+$.

1.12 Appendix: Applications

In this section, we provide additional information on the applications of Section 1.9. First, we give results on the computation time and the FWER for the analysis of simulated data. Subsequently, we investigate the impact of the number of iterations and permutations on power in the application to fMRI data.

1.12.1 Simulations

We provide additional information on the simulations of Section 1.9.1; we give results on the computation time and the FWER in different settings and for different p-value

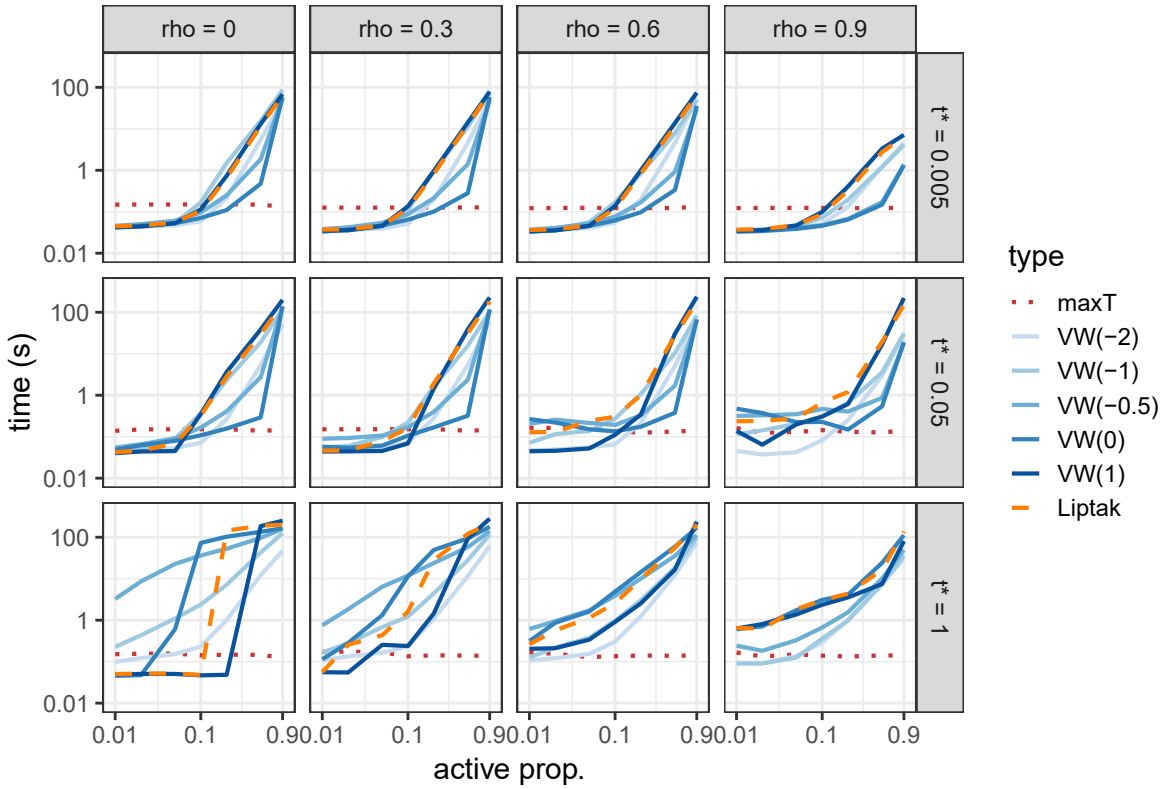

FIGURE 1.7: Simulated data: computation time (log scale) for the analysis of the set S of active variables, by active proportion a (log scale) and for different p-value combinations. Variables have equi-correlation ρ. P-values smaller than t^* are truncated.

combinations. These are shown for the scenarios with $\beta = 0.95$ and $t^* \in \{0.005, 0.05, 1\}$, but other values of β and t^* lead to analogous results.

Figure 1.7 displays the computation time needed to analyze the set S of false hypotheses. Time increases with the density of the signal and when less p-values are truncated, i.e., when a and t^* are high. In most scenarios, the method converges in less than a minute; the maximum time is around 4 minutes and 40 seconds, obtained when applying VW(1) in a scenario with dense signal ($a = 0.9$), medium-low correlation ($\rho = 0.3$) and no truncation ($t^* = 1$).

Figure 1.8 shows the FWER, computed as the proportion of times when the method finds at least one discovery among the set $M \setminus S$ of true hypotheses. Results confirm that the procedure controls the FWER; indeed, the FWER never exceeds the significance level α by more than two standard deviations, i.e., it is never higher than 0.063.

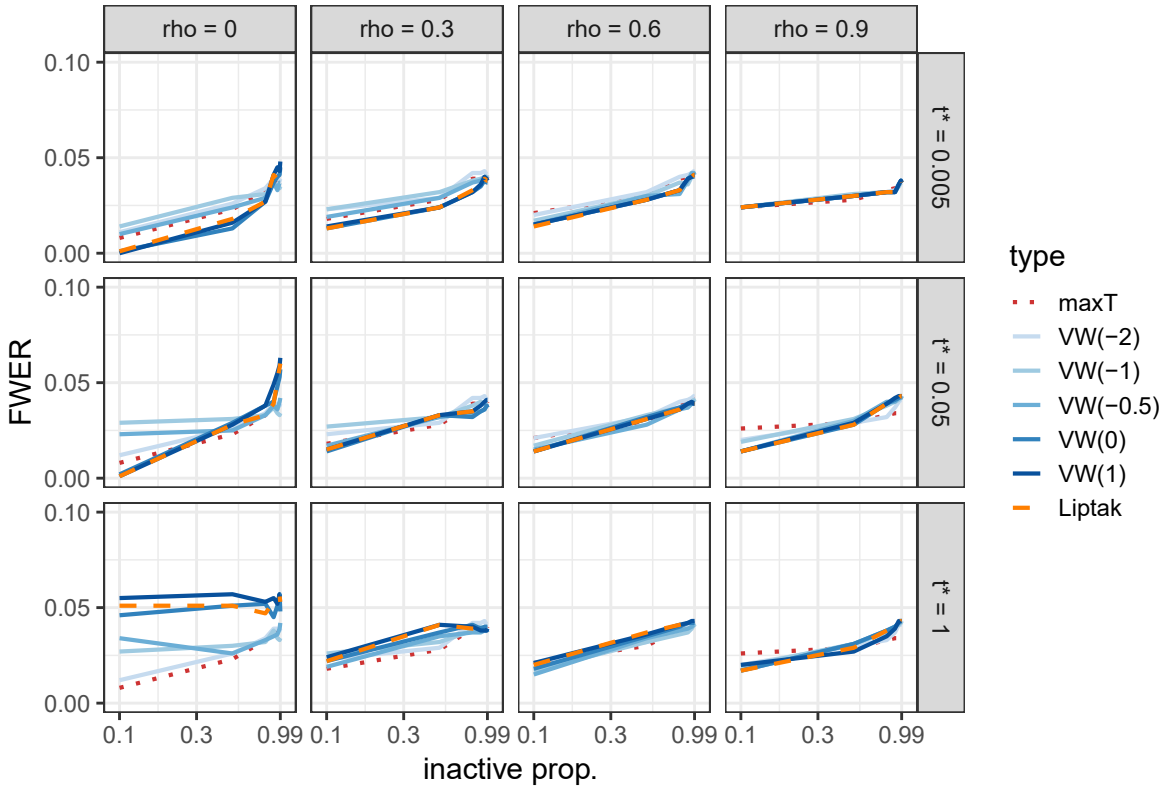

FIGURE 1.8: Simulated data: FWER computed on the set $M \setminus S$ of inactive variables, by inactive proportion $1 - a$ (log scale) and for different p-value combinations. Variables have equi-correlation ρ. P-values smaller than t^* are truncated.

1.12.2 fMRI data

In this section, we investigate the role of the number of iterations of the single-step shortcut and the number of permutations. We do this by examining two clusters from the Auditory data of Section 1.9.2.1: (1) the biggest cluster with threshold 3.2 (FP/CG/SFG/TOF/LO/LG); (2) its smallest sub-cluster with non-null activation (Left SG/AG).

First, we analyze these two clusters, stopping the algorithm at different times; the number of permutations is fixed at $B = 200$. Figure 1.9 shows the number of rejected, non-rejected and unsure hypotheses by the number of iterations. As expected, the number of unsure hypotheses quickly decreases, and becomes less than 0.5% of the total after only 20 iterations. A high number of iterations is required only for the very last unsure hypotheses.

Subsequently, to investigate the impact of the number B of permutations on power, we study the clusters with different values B. Figure 1.10 shows the mean TDP lower confidence bound by B, obtained by performing each analysis 1000 times and using

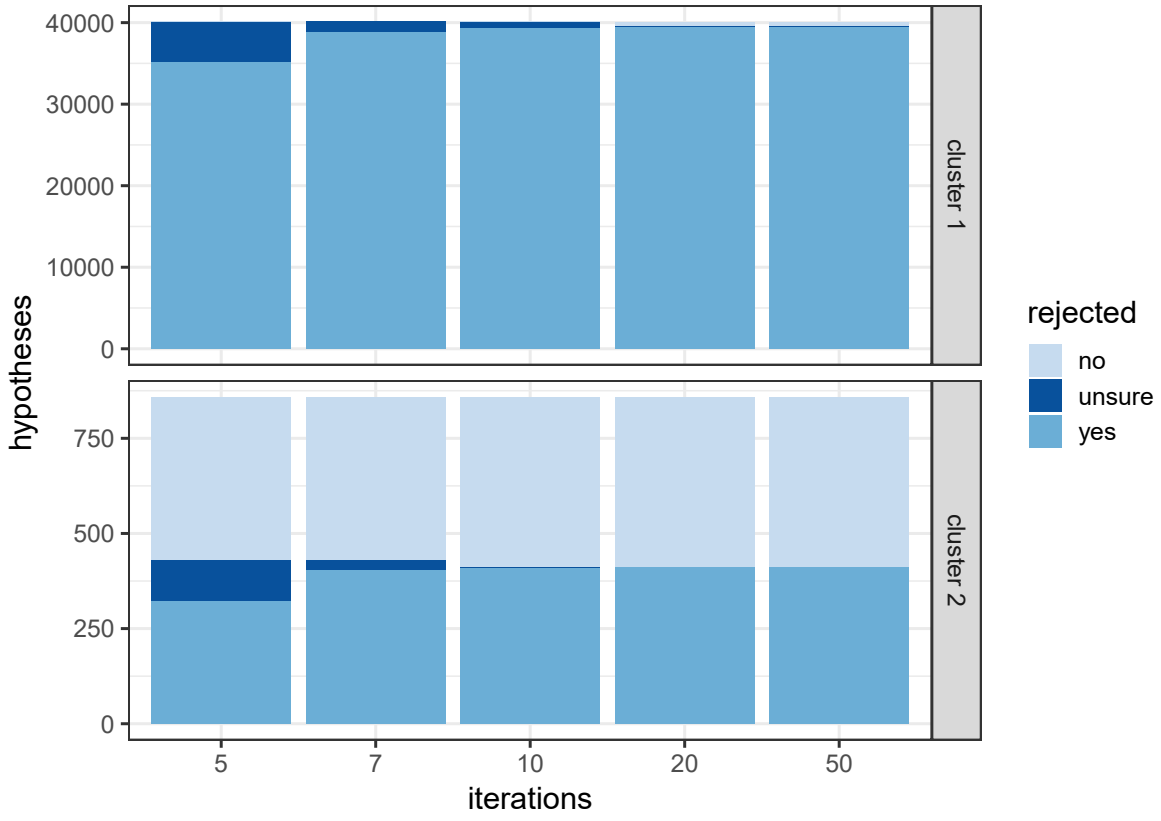

FIGURE 1.9: Auditory data: rejected, non-rejected and unsure hypotheses by number of iterations, for clusters (1) FP/CG/SFG/TOF/LO/LG; (2) Left SG/AG.

at most 50 iterations. The power is increasing for $B \leq 300$, and then becomes approximately constant; from $B = 200$ to 300, the gain is very small. Notice that the power peaks when B is a multiple of $1/\alpha$, since the permutation test is exact only for these values of B (Hemerik and Goeman, 2018a). Aside from a lower mean power, another drawback of using few permutations is that results may be variable, due to the randomness of the permutations.

1.13 Appendix: Proofs

Lemma 1.2. *Under Assumption 1.1, the test that rejects H_S when $T_S > T_S^{(\omega_0)}$ is an α-level test.*

Proof. From Assumption 1.1, $X_N \overset{\mathrm{d}}{=} \pi X_N$ for each $\pi \in \mathcal{P}$. If H_S is true, i.e., if $S \subseteq N$, then the joint distribution of the test statistics $T_S^\pi = T_S(\pi X_S)$, with $\pi \in \mathcal{P}$, is invariant under all transformations in $\mathcal{P}$. Then proof of Lemma 1.2 is in Hemerik and Goeman (2018b) (see Theorem 1). □

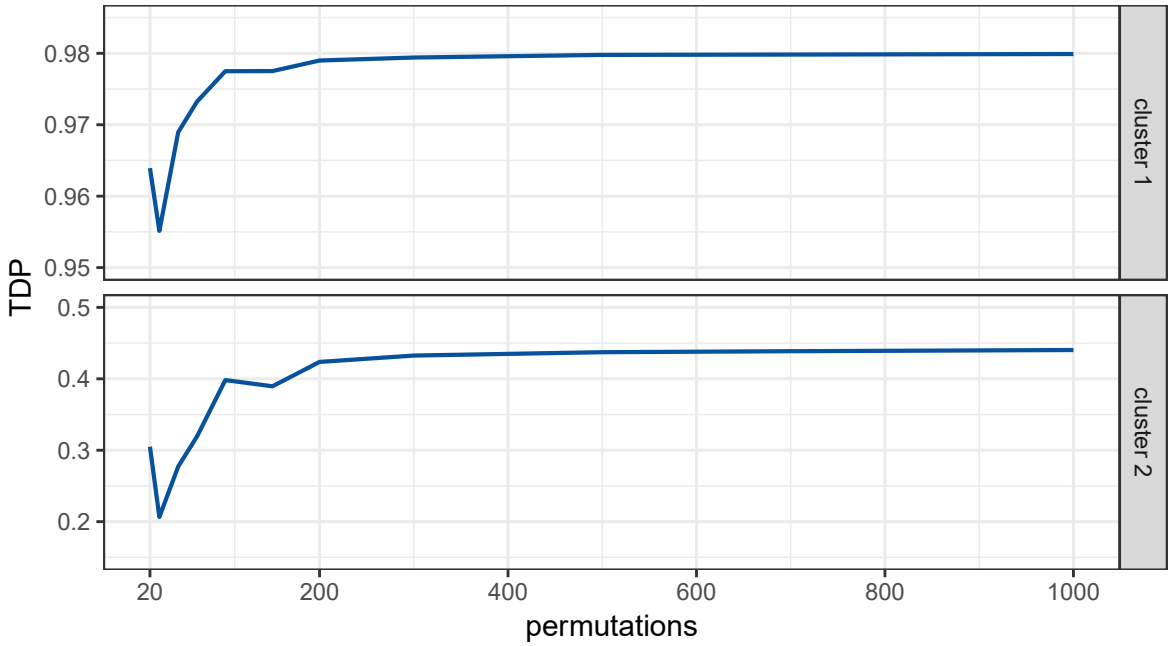

FIGURE 1.10: Auditory data: TDP lower confidence bounds by number of permutations, for clusters (1) FP/CG/SFG/TOF/LO/LG; (2) Left SG/AG.

Theorem 1.3. *Under Assumption 1.1, the test that rejects H_S when $C_S^{(\omega)} > 0$ is an α-level test.*

Proof. The k-th smallest centered statistic is $C_S^{(k)} = T_S - T_S^{(B-k+1)}$, for any $k \in \{1, \dots, B\}$. In particular, since $\omega = B - \omega_0 + 1$, we obtain $C_S^{(\omega)} = T_S - T_S^{(\omega_0)}$. Then

$$T_S > T_S^{(\omega_0)} \quad \text{if and only if} \quad T_S - T_S^{(\omega_0)} > 0 \quad \text{if and only if} \quad C_S^{(\omega)} > 0.$$

By Lemma 1.2, under Assumption 1.1 the test that rejects H_S when $C_S^{(\omega)} > 0$ is an α-level test. □

Lemma 1.4. *$\phi(0) = 0$ and $\phi(s+1) = 1$. Moreover, $\phi(z) = 0$ if and only if $z \in \{0, \dots, q\}$.*

Proof. When $z = 0$, we have $\mathcal{V}_0 = 2^M$ and $\emptyset \in (\mathcal{V}_0 \setminus \mathcal{R})$, so $\phi(0) = 0$. When $z = s + 1$, we have $\mathcal{V}_{s+1} = \emptyset \subseteq \mathcal{R}$, therefore $\phi(s+1) = 1$.

Fix a generic $z \in \{0, \dots, s+1\}$. By definition (1.6), $\phi(z) = 0$ if and only if there exists a set $V \subseteq M$ such that $|V \cap S| \geq z$ and $V \notin \mathcal{R}$. By definition (1.1) of q, this is true if and only if $z \leq q$. □

Lemma 1.5. *$\ell_z(v) \leq C_V^{(\omega)}$ for all $V \in \mathcal{V}_z(v)$. Hence $\min_v \ell_z(v) > 0$ implies $\phi(z) = 1$.*

Proof. Fix a set $V \in \mathcal{V}_z(v)$, so that $|V| = v$ and $|V \cap S| \geq z$, and a transformation π. The corresponding centered statistic may be written as

$$C_V^\pi = \sum_{h=1}^{z} C_{\hat{i}_h}^\pi + \sum_{h=1}^{v-z} C_{\hat{j}_h}^\pi$$

where, similarly to definition (1.11) of b_v^π, we have

$$\begin{aligned} V \cap S = \{\hat{i}_1(\pi), \ldots, \hat{i}_{|V\cap S|}(\pi)\} \quad &: \quad C_{\hat{i}_1(\pi)}^\pi \leq \ldots \leq C_{\hat{i}_{|V\cap S|}(\pi)}^\pi \\ V \setminus \{\hat{i}_1(\pi), \ldots, \hat{i}_z(\pi)\} = \{\hat{j}_1(\pi), \ldots, \hat{j}_{v-z}(\pi)\} \quad &: \quad C_{\hat{j}_1(\pi)}^\pi \leq \ldots \leq C_{\hat{j}_{v-z}(\pi)}^\pi. \end{aligned}$$

We compare the definitions of b_v^π and C_V^π, starting with the elements in S. Since $(V \cap S) \subseteq S$, the statistics in $V \cap S$ cannot be smaller than the first smallest statistics in S, i.e.,

$$C_{\hat{i}_h(\pi)}^\pi \geq C_{i_h(\pi)}^\pi \quad (h \in \{1, \ldots, |V \cap S|\}). \tag{1.28}$$

A similar comparison can be made for the elements outside S. Write

$$\begin{aligned} \{\hat{j}_1(\pi), \ldots, \hat{j}_{v-z}(\pi)\} &= \{\hat{i}_{z+1}(\pi), \ldots, \hat{i}_{|V\cap S|}(\pi)\} \cup (V \setminus S) \\ \{j_1(\pi), \ldots, j_{m-z}(\pi)\} &= \{i_{z+1}(\pi), \ldots, i_s(\pi)\} \cup (M \setminus S). \end{aligned}$$

As $(V \setminus S) \subseteq (M \setminus S)$, the statistics in $V \setminus S$ cannot be smaller than the first smallest statistics in $M \setminus S$. By combining this result with (1.28), we obtain

$$C_{\hat{j}_h(\pi)}^\pi \geq C_{j_h(\pi)}^\pi \quad (h \in \{1, \ldots, v - z\}).$$

As a consequence, $C_V^\pi \geq b_v^\pi$. Since the inequality holds for any transformation $\pi \in \boldsymbol{\pi}$, it holds also for the quantile, so that $C_V^{(\omega)} \geq b_v^{(\omega)} = \ell_z(v)$.

If $\ell_z(v) > 0$, then $C_V^{(\omega)} > 0$ for all $V \in \mathcal{V}_z(v)$, and so $\mathcal{V}_z(v) \subseteq \mathcal{R}$. Finally, if $\min_v \ell_z(v) > 0$, then $\mathcal{V}_z(v) \subseteq \mathcal{R}$ for each $v \in \{z, \ldots, m\}$, and so $\mathcal{V}_z \subseteq R$; by definition, $\phi(z) = 1$. □

Proposition 1.6. *As $\underline{\phi}(z) \leq \phi(z)$ for each $z \in \{0, \ldots, s+1\}$, $q^{(0)} \geq q$.*

Proof. Fix a value $z \in \{0, \ldots, s+1\}$, and suppose that $\underline{\phi}(z) = 1$. By definitions (1.12) and (1.13), this means that there exists $z^* \in \{0, \ldots, z\}$ with $\min_v \ell_{z^*} > 0$. By Lemma 1.5, this implies that $\phi(z^*) = 1$ and, since ϕ is increasing, $\phi(z) = 1$. As $\underline{\phi}(z) = 1$ implies

$\phi(z) = 1$, we have $\underline{\phi}(z) \leq \phi(z)$. When comparing the change points of $\underline{\phi}$ and ϕ, we obtain $q^{(0)} \geq q$. □

Theorem 1.7. $d^{(0)} \leq d$.

Proof. From Proposition 1.6 we have $q^{(0)} \geq q$. Since $d^{(0)} = s - q^{(0)}$ and $d = s - q$, we obtain $d^{(0)} \leq d$. □

Lemma 1.8. $\min_v u_z(v) \leq 0$ *implies* $\phi(z) = 0$.

Proof. Suppose that $\min_v u_z(v) \leq 0$. This means that there exists $v \in \{z, \ldots, m\}$ with $u_z(v) \leq 0$. Since $u_z(v) = C_{V_v}^{(\omega)}$ with $V_v \in \mathcal{V}_z(v)$, we have that $\mathcal{V}_z(v) \not\subseteq \mathcal{R}$. Hence $\mathcal{V}_z \not\subseteq \mathcal{R}$, and so $\phi(z) = 0$. □

Proposition 1.9. $\underline{\phi}(z) \leq \phi(z) \leq \overline{\phi}(z)$ *for each* $z \in \{0, \ldots, s+1\}$. *Hence* $\underline{\phi}(z) = \overline{\phi}(z)$ *implies* $\underline{\phi}(z) = \phi(z)$, *i.e., equivalence between the shortcut and closed testing.*

Proof. Fix a value $z \in \{0, \ldots, s+1\}$, and suppose that $\overline{\phi}(z) = 0$. By definitions (1.19) and (1.20), this means that there exists $z^* \in \{z, \ldots, s+1\}$ with $\min_v u_{z^*} \leq 0$. By Lemma 1.5, this implies that $\phi(z^*) = 0$ and, since ϕ is increasing, $\phi(z) = 0$. From the result of Proposition 1.6, and since $\overline{\phi}(z) = 0$ implies $\phi(z) = 0$, we have $\underline{\phi}(z) \leq \phi(z) \leq \overline{\phi}(z)$. As a consequence, if $\underline{\phi}(z) = \overline{\phi}(z)$, then $\underline{\phi}(z) = \phi(z) = \overline{\phi}(z)$. □

Proposition 1.10. *For any* $n \in \mathbb{N}$ *and any* $z \in \{0, \ldots, s+1\}$,

$$\underline{\phi}^{(n)}(z) \leq \underline{\phi}^{(n+1)}(z) \leq \underline{\phi}^{(m)}(z) = \phi(z) = \overline{\phi}^{(m)}(z) \leq \overline{\phi}^{(n+1)}(z) \leq \overline{\phi}^{(n)}(z).$$

Hence $\underline{\phi}^{(n)}(z) = \overline{\phi}^{(n)}(z)$ *implies* $\underline{\phi}^{(n)}(z) = \phi(z)$, *i.e., equivalence between the iterative shortcut and closed testing. Moreover,* $q^{(n)} \geq q^{(n+1)} \geq q^{(m)} = q$.

Proof. Fix $n \in \mathbb{N}$ and $z \in \{0, \ldots, s+1\}$. First, we prove that $\underline{\phi}^{(n)}(z) \leq \phi(z) \leq \overline{\phi}^{(n)}(z)$; as a consequence, $\underline{\phi}^{(n)}(z) = \overline{\phi}^{(n)}(z)$ implies $\underline{\phi}^{(n)}(z) = \phi(z)$. When we apply the shortcut within a subspace $\mathcal{V}_z^k$ of $\mathcal{V}_z$, by Proposition 1.9 we obtain $\underline{\phi}(z) \leq \phi(z) \leq \overline{\phi}(z)$. Since this property holds for any subspace, it holds also when we take the minimum of $\underline{\phi}(z)$ and $\overline{\phi}(z)$ over all subspaces, and so $\underline{\phi}^{(n)}(z) \leq \phi(z) \leq \overline{\phi}^{(n)}(z)$.

Subsequently, we prove that

$$\underline{\phi}^{(n)}(z) \leq \underline{\phi}^{(n+1)}(z) \leq \phi(z) \leq \overline{\phi}^{(n+1)}(z) \leq \overline{\phi}(z),$$

and so $q^{(n)} \geq q^{(n+1)} \geq q$. The single-step shortcut examines $\mathcal{V}_z$. If it determines that $\underline{\phi}(z) = \phi(z) = \overline{\phi}(z)$, the procedure does not partition $\mathcal{V}_z$, and trivially we have $\underline{\phi}^{(n)}(z) = \phi(z) = \overline{\phi}^{(n)}(z)$ for any $n \in \mathbb{N}$. Otherwise, if $\underline{\phi}(z) = 0 < \overline{\phi}(z) = 1$, at step $n = 1$ we partition $\mathcal{V}_z$. By Proposition 1.10, and since $\underline{\phi}^{(1)}$ and $\overline{\phi}^{(1)}$ take values in $\{0, 1\}$, we must have $\underline{\phi}(z) \leq \underline{\phi}^{(1)}(z) \leq \phi(z) \leq \overline{\phi}^{(1)}(z) \leq \overline{\phi}(z)$. The same argument may be applied for any subsequent step $n \in \mathbb{N}$.

Finally we prove that $\underline{\phi}^{(m)}(z) = \phi(z) = \overline{\phi}^{(m)}(z)$, and thus $q^{(m)} = q$. When a subspace contains a single set V, both the bound (1.11) and the path (1.18) coincide with its quantile $C_V^{(\omega)}$, and so the shortcut in the subspace must be equivalent to closed testing. Assume the worst case, where $s = m$ and $z = 1$, and where the shortcut is not equivalent to closed testing in any subspace containing more than one set. The space of interest is $\mathcal{V}_1 = 2^M \setminus \{\emptyset\}$, with size $|\mathcal{V}_1| = 2^m - 1$. After m steps, the procedure generates $2^m - 1$ subspaces, each of them containing exactly one set, and so the shortcut is equivalent to closed testing within each one. As a consequence, $\underline{\phi}^{(m)}(z) = \phi(z) = \overline{\phi}^{(m)}(z)$. □

Theorem 1.11. *$d^{(n)} \leq d^{(n+1)} \leq d^{(m)} = d$ for each $n \in \mathbb{N}$.*

Proof. From Proposition 1.10, for each $n \in \mathbb{N}$ we have $q^{(n)} \geq q^{(n+1)} \geq q^{(m)} = q$, and so $d^{(n)} \leq d^{(n+1)} \leq d^{(m)} = d$. □

Proposition 1.12. *Let $V \subseteq M$ and $i \in M$. If i satisfies condition (1.24), then $V \in \mathcal{R}$ implies $(V \cup \{i\}) \in \mathcal{R}$. If i satisfies condition (1.25), then $(V \cup \{i\}) \in \mathcal{R}$ implies $V \in \mathcal{R}$.*

Proof. Recall that $V \subseteq \mathcal{R}$ if and only if $T_V > T_V^{(\omega_0)}$ (equivalence between Lemma 1.2 and Theorem 1.3). Fix an index $i \in M$, and define $Q = S \cup \{i\}$. If $i \in S$, we obtain the trivial case where $Q = S$, hence suppose that $i \in M \setminus S$. In this case, $T_Q^\pi = T_S^\pi + f(T_i^\pi)$, with $f(T_i^\pi) \geq 0$, for each π.

First, assume that $S \in \mathcal{R}$ and that property (1.24) holds. Since $f(T_i) \geq 0$, we have $T_Q \geq T_S$. For $\pi \neq \text{id}$, we have $f(T_i^\pi) = 0$, and so $T_Q^\pi = T_S^\pi$. Hence all test statistics for Q and S coincide, with the exception of the observed statistic. Since $S \in \mathcal{R}$, $T_S > T_S^{(\omega_0)}$, and so when ordering the statistics we obtain $T_Q^{(k)} = T_S^{(k)}$ for all $k \leq \omega_0$. Therefore $T_Q \geq T_S > T_S^{(\omega_0)} = T_Q^{(\omega_0)}$, and thus $Q \in \mathcal{R}$.

Subsequently, assume that $Q \in \mathcal{R}$ and that property (1.25) holds. Since $f(T_i) = 0$, we have $T_S = T_Q$. For $\pi \neq \text{id}$, since $f(T_i^\pi) \geq 0$, we have $T_Q^\pi \geq T_S^\pi$ and so $T_Q^{(\omega_0)} \geq T_S^{(\omega_0)}$. We obtain $T_S = T_Q > T_Q^{(\omega_0)} \geq T_S^{(\omega_0)}$, and thus $S \in \mathcal{R}$. □

Lemma 1.13. *In the worst case, the computational complexity of Algorithm 1 is of order $mB \log(mB)$.*

Proof. To compute the values of the bound $\ell_z(v)$ for $v \in \{z, \dots, m\}$ as in (1.11), the algorithm operates as following. First, it sorts the centered test statistics for each permutation as in (1.9) and (1.10), with a number of operations of order $B\{s\log(s) + (m-z)\log(m-z)\}$. Subsequently, it computes

$$\ell_z(z) = b_z^{(\omega)} \quad \text{where} \quad b_z^\pi = \sum_{h=1}^{z} C_{i_h}^\pi \quad (\pi \in \boldsymbol{\pi})$$
$$\ell_z(v) = b_v^{(\omega)} \quad \text{where} \quad b_v^\pi = b_{v-1}^\pi + C_{j_{v-z}}^\pi \quad (v \in \{z+1, \dots, m\},\ \pi \in \boldsymbol{\pi}).$$

This requires mB sums and $m-z+1$ sortings of B elements, and so it uses $mB + (m-z+1)B\log(B)$ operations.

In the worst case, when $s = m$ and $z = 1$, the total number of operations needed to compute ℓ_z is of order $mB\{\log(m) + \log(B)\} = mB\log(mB)$. The same argument applies to the path u_z. □

Lemma 1.14. *In the worst case, Algorithm 2 converges after a number of iterations of order 2^m, where each iteration has complexity of order $mB\log(mB)$.*

Proof. In the worst case, $s = m$, $z = 1$, and all subspaces containing more than one set need to be partitioned. After m steps, the procedure generates $2^m - 1$ subspaces, each of them containing exactly one set. In this case, the total number of iterations of the single-step shortcut is of order 2^m. The computational complexity of each iteration is given by Lemma 1.13. □

Lemma 1.15. *In the worst case, embedding Algorithm 1 into a binary search requires a number of operations of order $mB\log(m)\log(mB)$.*

Proof. By Lemma 1.13, the single-step shortcut of Algorithm 1 has complexity at most of order $mB\log(mB)$. In the worst case, when $s = m$, the binary search applies the shortcut at most $\log_2(m)$ times (Knuth, 1998). Hence the worst-case complexity is of order $mB\log(m)\log(mB)$. □

Lemma 1.16. *Define c_1 as in (1.26), and c_2 as in (1.27). Then $\ell_z(z) \geq \ell_z(z+1) \geq \dots \geq \ell_z(c_1)$, and $\ell_z(c_2) \leq \ell_z(c_2+1) \dots \leq \ell_z(m)$.*

Proof. For any $v \in \{z+1, \dots, m\}$, we have

$$\ell_z(v-1) = b_{v-1}^{(\omega)} \quad \text{where} \quad b_{v-1}^\pi = b_v^\pi - C_{j_{v-z}}^\pi \quad (\pi \in \boldsymbol{\pi}).$$

If $v \leq c_1$, then $C^{\pi}_{j_{v-z}(\pi)} \leq 0$ for all π, and so $\ell_z(v-1) \geq \ell_z(v)$; as a consequence, $\ell_z(z) \geq \ell_z(z+1) \ldots \geq \ell_z(c_1)$. Similarly, if $v > c_2$, then $C^{\pi}_{j_{v-z}(\pi)} \geq 0$ for all π, and so $\ell_z(v-1) \leq \ell_z(v)$; therefore $\ell_z(c_2) \leq \ell_z(c_2+1) \leq \ldots \leq \ell_z(m)$. □

Lemma 1.17. *$u_z(v)$ is the same in $\mathcal{V}_z$ and $\mathcal{V}_z^-$ for each $v \in \{z, \ldots, m-1\}$.*

Proof. By definition (1.18), in $\mathcal{V}_z$ the path is $u_z(v) = C^{(\omega)}_{V_v}$ with $V_v = \{i_1, \ldots, i_z\} \cup \{j_1, \ldots, j_{v-z}\}$ for any $v \in \{z, \ldots, m\}$. Recall that $\mathcal{V}_z^- = \{V \in \mathcal{V}_z : j^* \notin V\}$, where $j^* = j_{m-z}$ (see Section 1.7.1). Therefore in $\mathcal{V}_z^-$ the greatest set is $M \setminus \{j^*\}$, and so the path u_z is defined for sizes $v \in \{z, \ldots, m-1\}$. Moreover, $j^* \notin V_v$ for all $v \in \{z, \ldots, m-1\}$, hence in $\mathcal{V}_z^-$ the path is defined as in $\mathcal{V}_z$. □

Chapter 2

Resampling-based inference for high-dimensional regression

2.1 Introduction

In the framework of linear regression, interest usually lies in discovering relevant predictor variables and assessing statistical significance. However, many challenges arise in high-dimensional settings, where the number of variables is potentially much larger than the sample size. Different methods have been proposed in literature to obtain error control and significance (Wasserman and Roeder, 2009; Meinshausen *et al.*, 2009; Meinshausen and Bühlmann, 2010; Bühlmann, 2013; Zhang and Zhang, 2014; Lee *et al.*, 2016; Dezeure *et al.*, 2017); for a review, see Dezeure *et al.* (2015).

In this chapter we propose a multiple testing method for high-dimensional linear regression that provides test statistics for any subset of variables and ensures asymptotic error control. We use the permutation framework, which has proven to be often more powerful than the parametric approach, especially when testing multiple hypotheses (Westfall and Young, 1993; Pesarin, 2001; Hemerik and Goeman, 2018b; Hemerik *et al.*, 2019).

We will employ the sample-splitting framework of the Multisplit (Meinshausen *et al.*, 2009), a powerful method that exploits variable selection techniques to compute adjusted p-values for all coefficients. The procedure repeatedly splits the data into two random subsets, using the first to select variables and the second to obtain raw p-values via ordinary least squares (OLS) estimation; then the raw p-values are adjusted and aggregated over the splits. Moreover, we will rely on the test introduced by Hemerik *et al.* (2020) for parameters in generalized linear models (GLMs), based on sign-flipping score contributions. The test is asymptotically exact, and allows for estimation of other unknown

parameters. Moreover, it is robust even for some misspecifications of the model such as overdispersion and heteroscedasticity and, in some cases, in presence of ignored nuisance parameters. Under the correct model, the power has been shown to be comparable to the parametric counterpart.

First, we introduce an approach similar to Meinshausen *et al.* (2009) that constructs permutation test statistics for each individual variable by means of repeated splits of the data. Then we define an asymptotically exact test for any subset by aggregating the individual variables' statistics with a suitable function. Different combining functions are possible, including the maximum and weighted sums. As we can test any subset, the procedure can be embedded into the closed testing framework (Marcus *et al.*, 1976). In particular it can be used within the methods that give simultaneous confidence sets for the true discovery proportion (TDP) (Genovese and Wasserman, 2006; Goeman and Solari, 2011; Goeman *et al.*, 2019), as well as the shortcut proposed in Chapter 1. This way we are able to provide confidence statements on the TDP of all subsets, valid even under post-hoc selection. Subsequently we propose an approximation of the procedure that requires less memory usage and shorter computation time, and can be scaled up to higher dimensions. The approximate version asymptotically controls the FWER when testing any single variable's hypothesis, and in simulations shows a good control even when studying variable sets.

The structure of the chapter is as follows. In Section 2.2 we introduce the model and its assumptions, as well as the Multisplit method of Meinshausen *et al.* (2009) and the test of Hemerik *et al.* (2020). Then we define the method and the approximate version in Sections 2.3 and 2.4, respectively. Finally, in Section 2.5 we explore the behavior of the proposed methods on simulated and real data. Proofs and some additional results are postponed to the Appendices (Sections 2.8 and 2.9).

2.2 High-dimensional linear regression

In this section we introduce notation and assumptions, as well as the two building blocks of the proposed procedure: the Multisplit method (Meinshausen *et al.*, 2009) and the permutation test based on sign-flipping score contributions (Hemerik *et al.*, 2020; Finos *et al.*, 2021).

We consider a linear regression framework with n observations and m variables, potentially high-dimensional ($n < m$). The model is

$$Y = X\beta + \varepsilon, \qquad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 I)$$

where $\mathcal{N}_n$ denotes the multivariate normal distribution of size n, and $I \in \mathbb{R}^{n\times n}$ is the identity matrix. Here $Y \in \mathbb{R}^n$ is the response vector, $X \in \mathbb{R}^{n\times m}$ is a fixed design matrix, $\beta \in \mathbb{R}^m$ is the vector of coefficients and $\varepsilon \in \mathbb{R}^n$ is a random error vector. Even though some results will be asymptotic in n, for simplicity of notation we omit any dependence on it. Moreover, we assume that X has rank m, and $X^\top X/n$ converges to a finite positive semi-definite matrix as $n \to \infty$.

We are interested in exploring which variables in X are active, meaning that they have non-null coefficients and so an effect on the response Y. Let $M = \{1, \dots, m\}$ be the set of variable indices, and $N = \{j \in M \,:\, \beta_j = 0\}$ the unknown subset corresponding to inactive variables. For any $j \in M$, we may define the null hypothesis $H_j \,:\, \beta_j = 0$, that is true when $j \in N$, regardless of the value of other variables' coefficients. We want to study more variables taken together, i.e., test intersection hypotheses of the form

$$H_S = \bigcap_{j\in S} H_j \,:\, \beta_j = 0 \text{ for all } j \in S, \qquad S \subseteq M,\ S \neq \emptyset \tag{2.1}$$

with significance level $\alpha \in [0, 1)$. H_S is true if all variables in S are inactive, i.e., $S \subseteq N$.

To develop a multiple testing method to test any H_S, we rely on a selection procedure that estimates the set of active variables, returning a subset $A \subseteq M$. As in Meinshausen *et al.* (2009), we assume that this procedure has the following properties.

Assumption 2.1 (sparsity)**.** *The number of selected variables is at most half the sample size:* $|A| \leq n/2$.

Assumption 2.2 (screening property)**.** *Asymptotically, all active variables are selected:*

$$\lim_{n\to\infty} P(M \setminus N \subseteq A) = 1.$$

The ideal selection procedure, for which the screening property always holds, is an oracle method that selects all truly active variables, plus eventually some others. Even though such a procedure is not available in practice, we will use it in simulations to show the performance of the proposed method when Assumptions 2.1 and 2.2 are ensured. When studying real data, we suggest using the Lasso (Tibshirani, 1996) with a suitable calibration of the λ parameter, so that it selects enough variables for the screening property to be likely. If m_1 is an estimate of the expected number of active variables, we recommend choosing λ so that the Lasso selects $\min(2m_1, n/2)$ variables. If there is no information available to give an estimate m_1, we recommend selecting as many variables as possible, i.e., $n/2$.

2.2.1 Multisplit

The Multisplit method of Meinshausen *et al.* (2009) is a multiple testing procedure for high-dimensional linear regression that provides adjusted p-values p_j for each variable $j \in M$. Building on a proposal of Wasserman and Roeder (2009), the method repeatedly splits the data into two subsets for a number Q of times. The first subset is used to perform variable selection, while the second is used to compute raw p-values for the selected variables. P-values are obtained by adjusting the raw p-values and aggregating over the Q splits. Regarding the suggested number of splits, the Authors use $Q = 50$ in simulations.

For each split $q \in \{1, \dots, Q\}$, the n observations are randomly divided into two subsets $\mathcal{D}_0^q$ and $\mathcal{D}^q$ of equal size $n/2$. First, observations in $\mathcal{D}_0^q$ are employed to obtain an estimate $A^q \subseteq M$ of the set of active variables, using a variable selection procedure for which Assumptions 2.1 and 2.2 are assumed to hold. Then observations in $\mathcal{D}^q$ are used to compute raw p-values $\tilde{p}_j^q$ for each $j \in A^q$ via OLS estimation. Raw p-values for non-selected variables are set to 1.

Finally, these raw p-values are adjusted as

$$p_j^q = \min\{|A^q|\tilde{p}_j^q, 1\} \qquad (j \in M,\, q \in \{1, \dots, Q\}) \tag{2.2}$$

and aggregated over the splits as

$$p_j = \min\left\{1,\, (1 - \log(\gamma_{\min}) \inf_{\gamma \in (\gamma_{\min}, 1)} C_j(\gamma)\right\} \qquad (j \in M) \tag{2.3}$$

where

$$C_j(\gamma) = \min\left\{1, c_\gamma(p_j^1), \dots, c_\gamma(p_j^Q)\right\}.$$

Here c_γ is the empirical γ-quantile function, and $\gamma_{\min} \in (0, 1)$ is a lower bound for γ (typically 0.05).

The method identifies as active all variables j with $p_j \leq \alpha$. Meinshausen *et al.* (2009) show that this procedure asymptotically controls the family-wise error rate (FWER) at level α. Moreover, they prove that the p-values can be used to define a procedure with asymptotic control of the false discovery rate (FDR), extending the methodology of Benjamini and Hochberg (1995).

An algorithm for the Multisplit method is presented in Section 2.7.1.

2.2.2 Sign-flipping score contributions

In this section we introduce the permutation tests proposed by Hemerik *et al.* (2020) and Finos *et al.* (2021) restricting to our setting, i.e., linear regression. Throughout this section suppose that the framework is low-dimensional with $n > m$.

Fix any variable $j \in M$. To test the individual hypothesis H_j against a two sided alternative, Hemerik *et al.* (2020) provide a permutation test constructed from the absolute value of the effective score

$$T_j^1 = \frac{1}{\sqrt{n}} \left| X_j^\top R_{-j} Y \right|,$$

where $X_j \in \mathbb{R}^n$ and $X_{-j} \in \mathbb{R}^{n \times (m-1)}$ are obtained from the design matrix X by taking and removing the j-th column, respectively, and

$$R_{-j} = I - X_{-j}(X_{-j}^\top X_{-j})^{-1} X_{-j}^\top \in \mathbb{R}^{n \times n} \tag{2.4}$$

is the residual maker matrix defined from X_{-j}.

A critical value for the test statistic T_j^1 is constructed using B random transformations of the data. The value of B used to define the test does not need to grow with m. Larger values of B tend to give more power, but to have non-zero power we only need $B \geq 1/\alpha$. Hence let $F_1, \ldots, F_B \in \mathbb{R}^{n \times n}$ be diagonal sign-flipping matrices, where $F_1 = I$ is the identity, while the diagonal elements of the other matrices are independently and uniformly drawn from $\{-1, 1\}$. These matrices define

$$T_j^b = |t_{j,b}^\top Y| \qquad t_{j,b} = \frac{1}{\sqrt{n}} R_{-j} F_b R_{-j} X_j \in \mathbb{R}^n \qquad (b \in \{1, \ldots, B\}). \tag{2.5}$$

Then a critical value is $T_j^{(\omega)}$, where $T_j^{(1)} \leq \ldots \leq T_j^{(B)}$ are the sorted values, $\omega = \lceil (1-\alpha)B \rceil$, and $\lceil \cdot \rceil$ denotes the ceiling function. As shown in the following proposition, the resulting test is asymptotically exact, but may be anti-conservative (Hemerik *et al.*, 2020; Finos *et al.*, 2021).

Proposition 2.3. *The test that rejects H_j when $T_j^1 > T_j^{(\omega)}$ is asymptotically an α-level test for any $j \in M$. For finite n, it may be anti-conservative as*

$$var(T_j^1) \geq var(T_j^b) \qquad (b \in \{1, \ldots, B\}).$$

From this framework, Finos *et al.* (2021) construct a test that is exact for any sample size n. Observing that

$$\mathrm{sd}(T_j^b \mid F_b) = \sigma \|t_{j,b}\|$$

for any transformation b, the intuition is to divide each T_j^b by the known part of this standard deviation, $\|t_{j,b}\|$. The resulting statistic is based on the absolute value of the standardized scores

$$\tilde{T}_j^b = |\tilde{t}_{j,b}^\top Y|, \qquad \tilde{t}_{j,b} = \mathrm{unit}(t_{j,b}) \tag{2.6}$$

where

$$\mathrm{unit}(t_{j,b}) = \begin{cases} 0 & \text{if } \|t_{j,b}\| = 0 \\ \|t_{j,b}\|^{-1} t_{j,b} & \text{otherwise} \end{cases} \tag{2.7}$$

denotes the normalization of a vector.

Theorem 2.4. *The test that rejects H_j when $\tilde{T}_j^1 > \tilde{T}_j^{(\omega)}$ is an α-level test for any $j \in M$.*

In the next section we rely on the ideas underlying Theorem 2.4, as well as the Multisplit framework, to construct permutation test statistics for each variable in high-dimensional linear regression. Then we show how the resulting statistics may be employed to study sets of variables.

2.3 Resampling-based Multisplit

In this section we propose an asymptotically exact test for any intersection hypothesis H_S, as given in (2.1), valid even in high-dimensional settings. The method builds on the idea that we can efficiently construct a test statistic for H_S by combining statistics for the individual hypotheses H_j with $j \in S$ in a suitable way. If the set of interest has size $|S| = s$, we take as combining function any $g : \mathbb{R}^s \longrightarrow \mathbb{R}$ which is increasing in each argument, such as the maximum or (weighted) sums.

First, we prove this in the low-dimensional setting of Section 2.2.2. In this framework, Hemerik *et al.* (2020) provide a permutation test for any intersection hypothesis, but it requires to compute new test statistics for each set S. In the following lemma we prove that an asymptotically exact test for H_S can be defined using

$$T_S^b = g\left(T_{j_1}^b, \ldots, T_{j_s}^b\right) \qquad (S = \{j_1, \ldots, j_s\},\, b \in \{1, \ldots, B\}). \tag{2.8}$$

Lemma 2.5. *The test that rejects H_S when $T_S^1 > T_S^{(\omega)}$ is asymptotically an α-level test for any non-empty $S \subseteq M$.*

This way, to study all subsets it is sufficient to compute the m individual statistics $T_1, \ldots, T_m$. Notice that the lemma holds also for the standardized version in (2.6).

Now we provide an analogous method for the high-dimensional setting. Similarly to Meinshausen *et al.* (2009), we split the data into two subsets for a number Q of times. The first subset is used to select variables through a procedure that is assumed to fulfill Assumptions 2.1 and 2.2. Then the second subset is used to construct permutation test statistics from random sign-flips of score contributions, as in Hemerik *et al.* (2020) and Finos *et al.* (2021).

Fix B diagonal sign-flipping matrices $F_1, \ldots, F_B \in \mathbb{R}^{n \times n}$, where $F_1 = I$ is the identity, while the diagonal elements of the other matrices are independently and uniformly drawn from $\{-1, 1\}$. For each split $q \in \{1, \ldots, Q\}$, we randomly divide observations into two equally-sized subsets $\mathcal{D}_0^q$ and $\mathcal{D}^q$. We use observations in $\mathcal{D}_0^q$ to estimate the set of active variables with $A^q \subseteq M$. Subsequently, the intuition is to use observations in $\mathcal{D}^q$ and selected variables in A^q to compute test statistics as in (2.5), sum these test statistics over the splits and suitably standardize as in (2.6). However, particular attention is to be paid to the use of transformations; if the same observation is in $\mathcal{D}^{q_1}$ and $\mathcal{D}^{q_2}$, then it must undergo the same sign-flipping transformations between the two splits. The procedure may be efficiently written as follows.

For each split q, we restrict the design matrix X to observations in $\mathcal{D}^q$ and variables in A^q, obtaining

$$X^q = X_{\mathcal{D}^q, A^q}.$$

For each selected variable $j \in A^q$, we define the split's residual maker matrix

$$R^q_{-j} = 0 \in \mathbb{R}^{n \times n} \qquad \text{except} \qquad R^q_{-j;\mathcal{D}^q,\mathcal{D}^q} = I - X^q_{-j}(X^{q\top}_{-j} X^q_{-j})^{-1} X^{q\top}_{-j}. \tag{2.9}$$

All elements of R^q_{-j} are zero except those corresponding to observations in $\mathcal{D}^q$, which are computed from the residual maker matrix defined from X^q_{-j} (see (2.4)).

Now fix any variable $j \in M$ and any transformation b. To compute a permutation test statistic for H_j, we transform and aggregate the splits' residual maker matrices, obtaining

$$C_{j,b} = \sum_{q\,:\,j \in A^q} R^q_{-j} F_b R^q_{-j} \in \mathbb{R}^{n \times n}. \tag{2.10}$$

Then we take

$$U_j^b = |u_{j,b}^\top Y|, \qquad u_{j,b} = \text{unit}(C_{j,b} X_j) \tag{2.11}$$

where $\text{unit}(\cdot)$ is the normalization given in (2.7). Notice that $U_j^b = 0$ if j is never selected. Subsequently, these individual test statistics U_j^b can be combined to test any intersection hypothesis H_S analogously to Lemma 2.5, using

$$U_S^b = g\left(U_{j_1}^b, \ldots, U_{j_s}^b\right) \qquad (S = \{j_1, \ldots, j_s\},\, b \in \{1, \ldots, B\}).$$

Theorem 2.6. *The test that rejects H_j when $U_j^1 > U_j^{(\omega)}$ is an α-level test for any $j \in M$. Moreover, the test that rejects H_S when $U_S^1 > U_S^{(\omega)}$ is asymptotically an α-level test for any non-empty $S \subseteq M$.*

To summarize, we have proposed a method to construct permutation test statistics for all variables in high-dimensional linear regression, using Q random splits and B random transformations. These individual test statistics are sufficient to define an asymptotically exact permutation test for any intersection hypothesis H_S. Indeed, by Theorem 2.6 such a test can be obtained combining the statistics for the variables in S through any function g that is increasing in each argument. As the method provides a test for all H_S, it can be embedded into the closed testing framework (Marcus *et al.*, 1976). For instance, it can be used within procedures that give simultaneous confidence sets for the TDP as in Genovese and Wasserman (2006) and Goeman and Solari (2011) and, if the function g is a sum, in the shortcut of Chapter 1.

In Section 2.7.2 we provide an algorithm for the method. In the worst case, it requires a number of operations of order n^4QB, and memory usage of order n^2Q. The following section shows how to define a new, faster method.

2.4 Approximate method

Section 2.3 provides a procedure to test any intersection hypothesis H_S in high-dimensional linear regression. As the method requires intensive memory usage, in this section we propose an approximation that is less expensive. We prove that the resulting approximate method allows to study any variable individually, then in Section 2.5.1 we show through simulations that it maintains a good error control when studying sets of variables.

Consider any variable $j \in M$ and any random transformation b. The new procedure is built by approximating the sum $C_{j,b}$ as defined in (2.10) with

$$\bar{C}_j = \bar{R}_{-j} F_b \bar{R}_{-j} \in \mathbb{R}^{n \times n}, \qquad \bar{R}_{-j} = \sum_{q\,:\,j \in A^q} R^q_{-j}. \tag{2.12}$$

While computing $C_{j,b}$ requires to store the splits' residual maker matrices $R^1_{-j}, \dots, R^Q_{-j}$, this is no longer necessary for $\bar{C}_j$. Indeed, the matrices can be summed as soon as they are determined and only $\bar{R}_{-j}$ is saved. The resulting test statistic is

$$\bar{U}^b_j = |\bar{u}^\top_{j,b} Y|, \qquad \bar{u}_{j,b} = \mathrm{unit}(\bar{C}_j X_j). \tag{2.13}$$

The following theorem shows that the statistics $\bar{U}^1_j, \dots, \bar{U}^B_j$ define a valid test for any individual hypothesis H_j.

Theorem 2.7. *The test that rejects H_j when $\bar{U}^1_j > \bar{U}^{(\omega)}_j$ is an α-level test for any $j \in M$.*

In conclusion, Theorem 2.6 gives an asymptotically valid, but computationally intensive, multiple testing method. Theorem 2.7 provides a less expensive procedure based on an approximation, which defines a valid α-level test for any individual hypothesis. In the following section we use simulations to study the behavior of the approximate method when used to test intersection hypotheses, comparing it to the exact method and the Multisplit of Meinshausen *et al.* (2009), as well as investigating the role of the variable selection procedure. We show in particular that approximate method controls the type I error in most of the considered settings.

An algorithm for the approximate method is provided in Section 2.7.3. The computational complexity is lower than the exact method, but still polinomial in n and linear both in Q and in B; the memory usage is of order n^2. As memory operations (write and read) affect the running time of an algorithm, the approximate method will prove to be much faster than the exact.

2.5 Applications

In this section we explore the performance of the proposed method through simulations, comparing it to the Multisplit (Meinshausen *et al.*, 2009). Then we embed the approximate method into the shortcut of Chapter 1 to study real gene expression data. The proposed method and the Multisplit are implemented in the packages `splitFlip`

(Vesely, 2021a) and `hdi` (Meier *et al.*, 2021) developed in R (R Core Team, 2017), respectively.

2.5.1 Simulations

We use simulations to study the performance of the proposed exact and approximate methods of Sections 2.3 and 2.4. First we compare the two methods, then we further investigate the behavior of the approximate method, comparing it to the Multisplit (Meinshausen *et al.*, 2009) and using different variable selection procedures. We correct for multiplicity with the maxT-method (Westfall and Young, 1993), corresponding to the combining function $g = \max$.

We use the same simulation settings proposed in Meinshausen *et al.* (2009). The design matrix $X \in \mathbb{R}^{n \times m}$ is defined in two ways. First, we simulate a Toeplitz design matrix coming from a centered multivariate normal distribution with $\text{cov}(X_j, X_h) = \rho^{|j-h|}$ for $j, h \in M$. Then we take the real $71 \times 4{,}088$ design matrix of the `riboflavin` dataset from the `R` package `hdi` (Meier *et al.*, 2021), which contains gene expression levels of *Bacillus subtilis*. Subsequently, the response variable is computed as $Y = X\beta + \varepsilon$, with $\varepsilon \sim \mathcal{N}_n(0, \sigma^2 I)$. The coefficient vector β is such that m_1 elements are non-null, with values either all equal to 1 (uniform-strength setting) or equal to $1, 2, \ldots, m_1$ (increasing-strength setting). The error standard deviation σ is computed so that the signal-to-noise ratio is SNR.

We analyze the set M of all variables with significance level α, using B random sign-flipping transformations and Q random splits of the data. We consider two variable selection procedures that select $2m_1$ variables: an oracle method where all truly active variables are always selected, and the Lasso with suitable λ-calibration. As observed in Section 2.2, the oracle selection allows to emphasize the behavior of the proposed methods when assumptions are met, while the Lasso allows to appreciate the performance in practical applications, when oracle selection is not feasible.

As a basic scenario, we take $n = 100$, $m = 100$ and $\rho \in \{0, 0.2, 0.5, 0.7, 0.9\}$ for the simulated design matrix, we fix $m_1 = 5$, SNR $= 4$ and $Q = 50$, and we use the oracle selection procedure; then we vary some of the parameters in the different analyses. For each setting, we simulate data 1000 times. We compute the number of rejections as the mean over the simulations, and the FWER as the proportion of simulations where the method rejects the null hypothesis for at least one inactive variable. Results are shown only for the uniform-strength setting, as those for the increasing-strength setting display the same behavior.

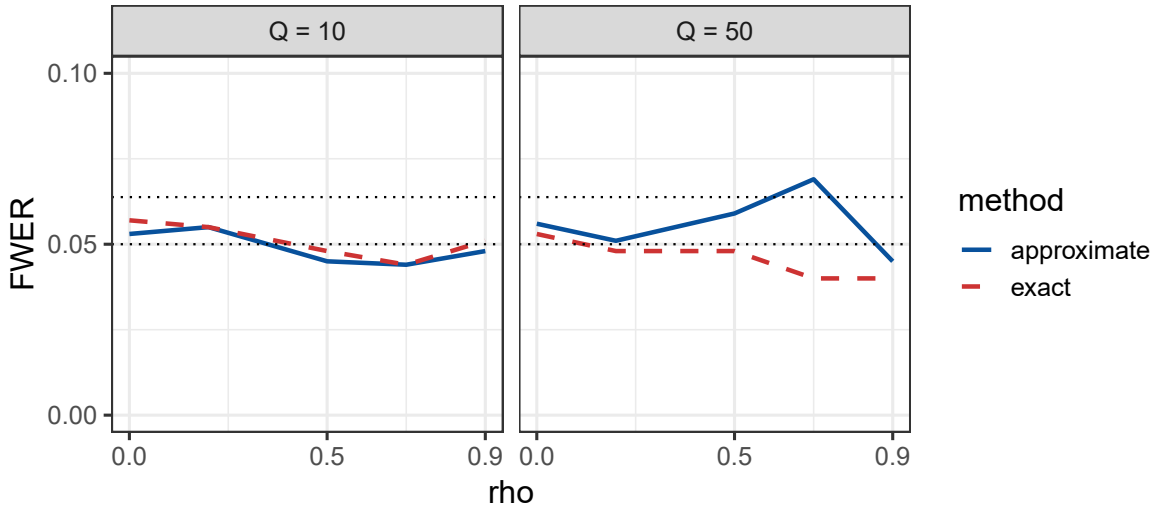

FIGURE 2.1: Simulated design matrix: FWER by covariance parameter ρ, for the approximate and exact methods using Q splits. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

TABLE 2.1: Real design matrix: results for the approximate and exact methods using Q splits.

	approximate		exact	
Q	10	50	10	50
FWER	0.046	0.048	0.045	0.048
rejections	3.1	3.4	3.9	3.9
time (s)	7.4	32.3	12.8	59.8

2.5.1.1 Approximate and exact

We compare the approximate method of Section 2.4 with the exact method of Section 2.3, expanding the basic scenario with $Q \in \{10, 50\}$.

The FWER and the total number of rejections obtained using the simulated design matrix are shown in Figures 2.1 and 2.2. The exact method always controls the FWER, as it never exceeds the significance level α by more than two standard deviations; the FWER given by the approximate method exceeds this threshold only in the setting with $\rho = 0.7$ and $Q = 50$. In terms of number of rejections, the approximate method is close to the exact, especially when the number of splits is high. The computation time is at most 32 seconds for the approximate, and 168 seconds for the exact.

Results for the real design matrix are in Table 2.1. In this case, both methods control the FWER; the approximate is slightly less powerful, but faster.

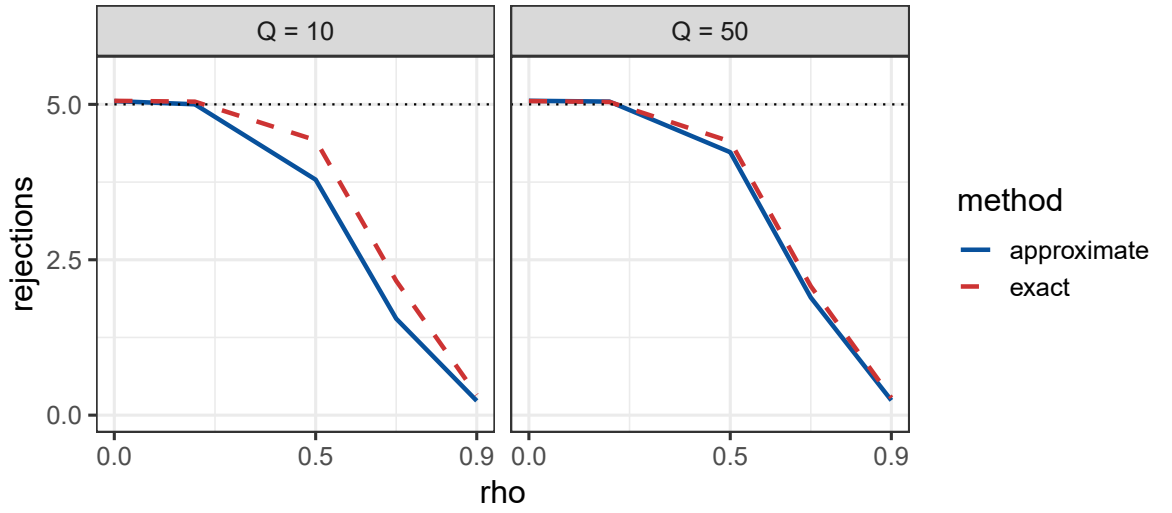

FIGURE 2.2: Simulated design matrix: number of rejections by covariance parameter ρ, for the approximate and exact methods using Q splits. The dotted line denotes the true number of active variables.

2.5.1.2 Approximate and Multisplit

Now we compare the approximate method with the Multisplit of Meinshausen *et al.* (2009). We expand the basic scenario taking $m \in \{100, 1000\}$ for the simulated data, as well as $m_1 \in \{5, 10\}$ and SNR $\in \{0.25, 1, 4, 16\}$. The settings with $\rho = 0.5$ correspond to those investigated in Meinshausen *et al.* (2009).

Figures 2.3 and 2.4 show results for the simulated design matrix with $m = 100$, while results for $m = 1000$, having an analogous behavior, are postponed to Section 2.8.1. The proposed method appears to control the FWER when the covariance parameter ρ is not too high, or the signal-to-noise ratio SNR is particularly low. Among the scenarios where the FWER is controlled, the method is always more powerful than the Multisplit, with greatest differences when ρ and SNR are low.

Figures 2.6 and 2.5 contain results for the real design matrix, for which the approximate method always controls the FWER and is more powerful than the Multisplit.

Computation times, shown in Section 2.5.1.2, are feasible, never exceeding 3 minutes.

2.5.1.3 Oracle and Lasso

Finally, we examine the approximate method for different sample sizes n, using both selection procedures illustrated in Section 2.2: oracle and Lasso (Tibshirani, 1996). The oracle is defined so that it always selects the m_1 truly active variables, plus m_1 others chosen at random. The Lasso selects the same number $2m_1$ of variables, with suitable λ-calibration.

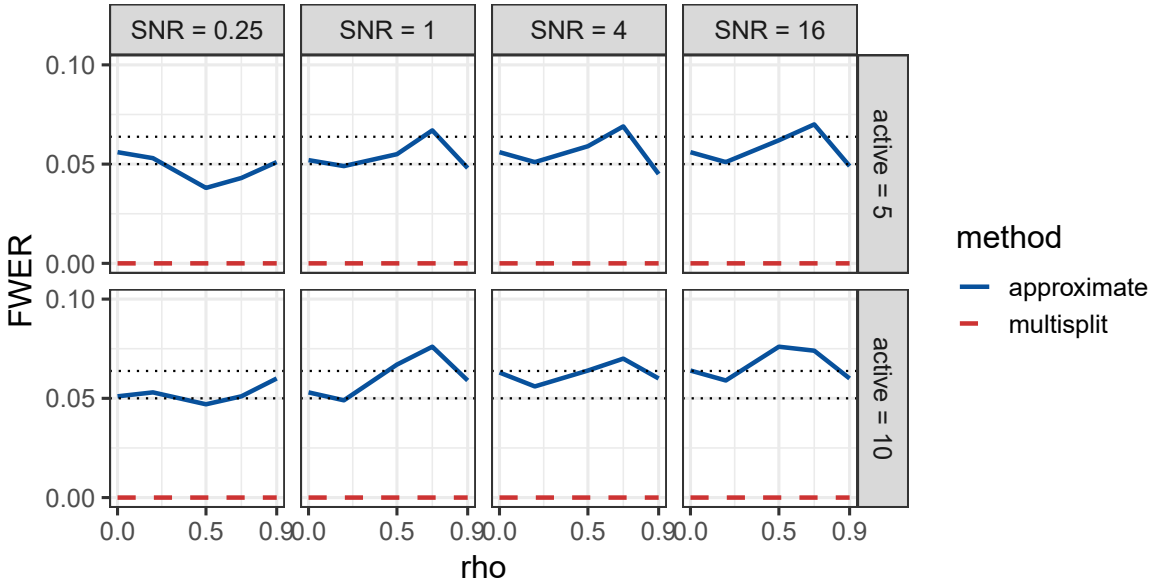

FIGURE 2.3: Simulated design matrix: FWER by covariance parameter ρ, for the approximate method and the Multisplit. *Active* and SNR denote the true number of active variables and the signal-to-noise ratio. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

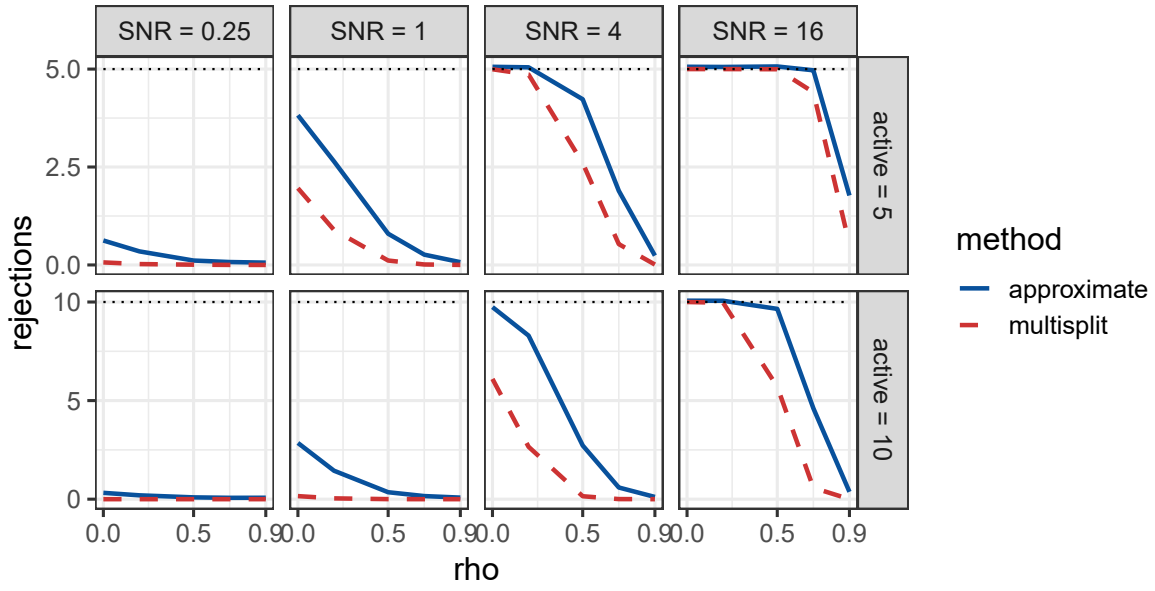

FIGURE 2.4: Simulated design matrix: number of rejections by covariance parameter ρ, for the approximate method and the Multisplit. *Active* and SNR denote the number of active variables and the signal-to-noise ratio. The dotted line corresponds to *active*.

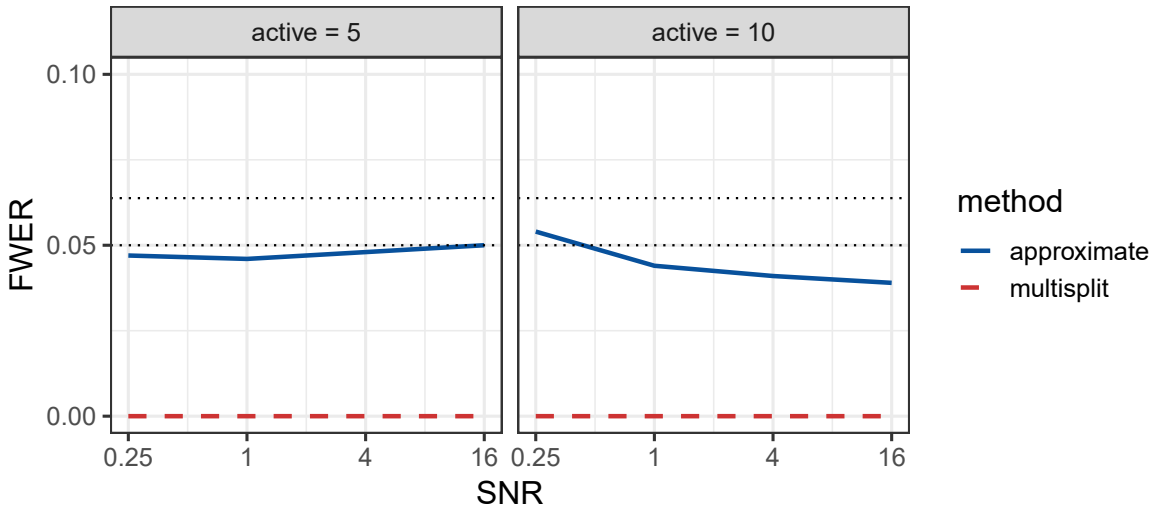

FIGURE 2.5: Real design matrix: FWER by signal-to-noise ratio SNR (log scale), for the approximate method and the Multisplit. *Active* denotes the number of active variables. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

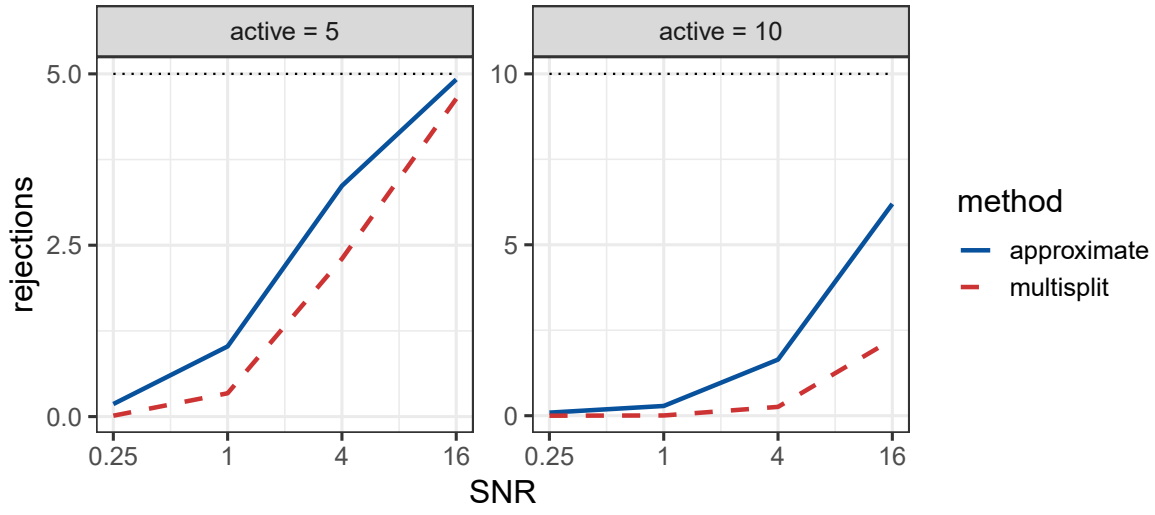

FIGURE 2.6: Real design matrix: number of rejections by signal-to-noise ratio SNR (log scale), for the approximate method and the Multisplit. *Active* denotes the number of active variables. The dotted line corresponds to *active*.

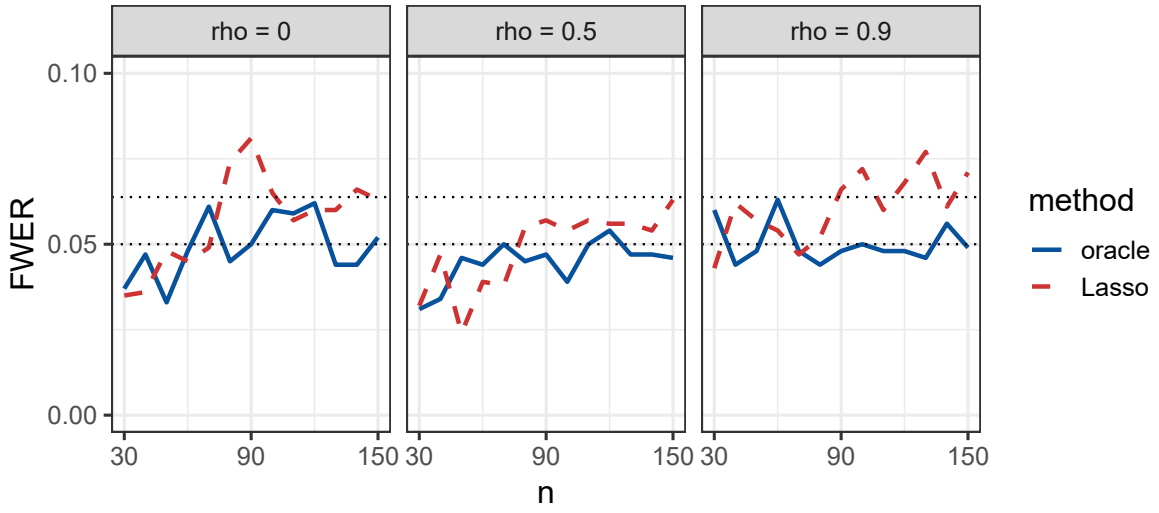

FIGURE 2.7: Simulated design matrix: FWER by sample size n, for the approximate method using oracle selection and Lasso. ρ denotes the covariance parameter. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

We study $n \in \{30, 40, \ldots, 150\}$ for the simulated design matrix. In the considered settings, the FWER is always controlled by the oracle but not by the Lasso, which fails in some scenarios both with low and high covariance parameter ρ (Figure 2.7). Moreover, in the cases where both methods control the FWER, as expected the oracle is always at least as powerful as the Lasso. This difference is more noticeable when n and ρ are low (Figure 2.8).

For the real data, where only 71 observations are available, we take $n \in \{30, 40, \ldots, 70\}$. The oracle still controls the FWER, while the Lasso looses control for all sample sizes. As expected, the power of the oracle increases with n (Figures 2.9 and 2.10).

These results underline that particular attention must be paid to the choice of the selection method. Indeed, the asymptotic error control of the proposed methods (Theorems 2.6 and 2.7) relies on the screening property given in Assumption 2.2, and is not ensured when the property is not fulfilled.

2.5.2 Riboflavin data

In this section we study the performance of the approximate method of Section 2.4 and the Multisplit (Meinshausen *et al.*, 2009) on real data. We analyze the `riboflavin` dataset from the `R` package `hdi` (Meier *et al.*, 2021), containing data on riboflavin production by *Bacillus subtilis*. Data consists of 71 observations of riboflavin production rate, as well as gene expression levels for 4,088 genes. We assume a linear model where the first is the response and the latter are the predictors. We are interested in making

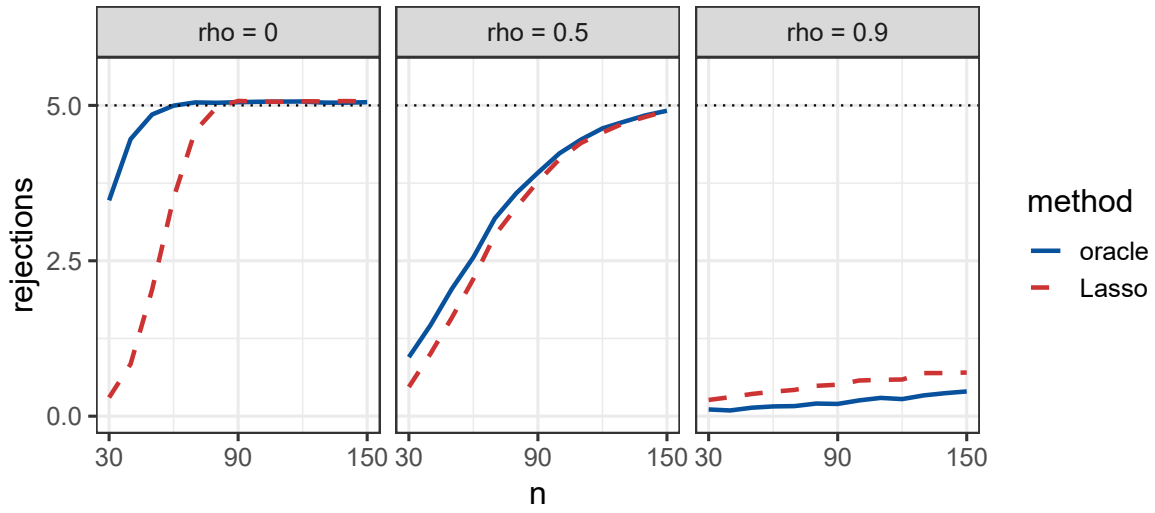

FIGURE 2.8: Simulated design matrix: number of rejections by sample size n, for the approximate method using oracle selection and Lasso. ρ denotes the covariance parameter. The dotted line corresponds to the true number of active variables.

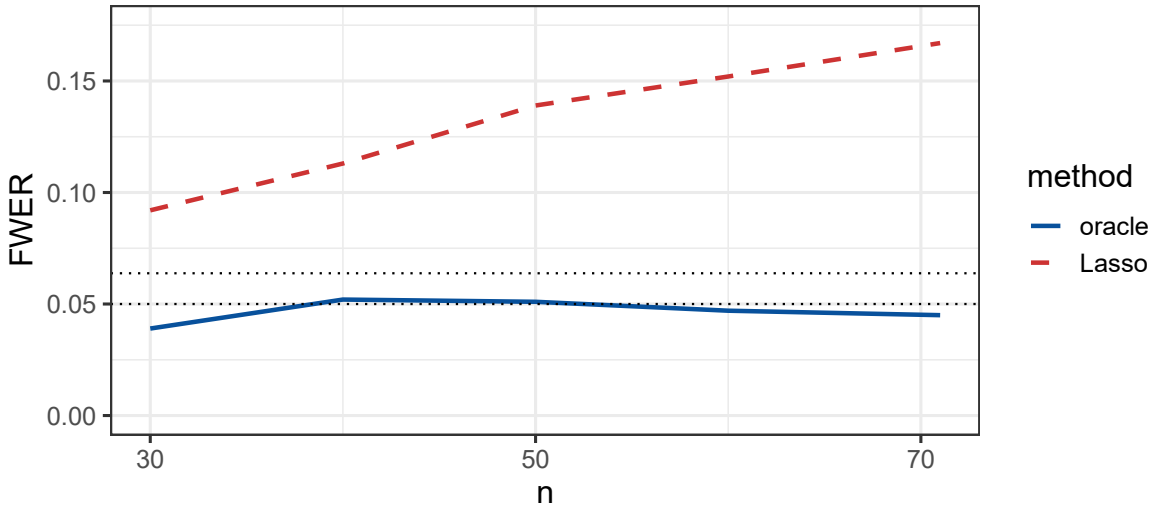

FIGURE 2.9: Real design matrix: FWER by sample size n, for the approximate method using oracle selection and Lasso. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

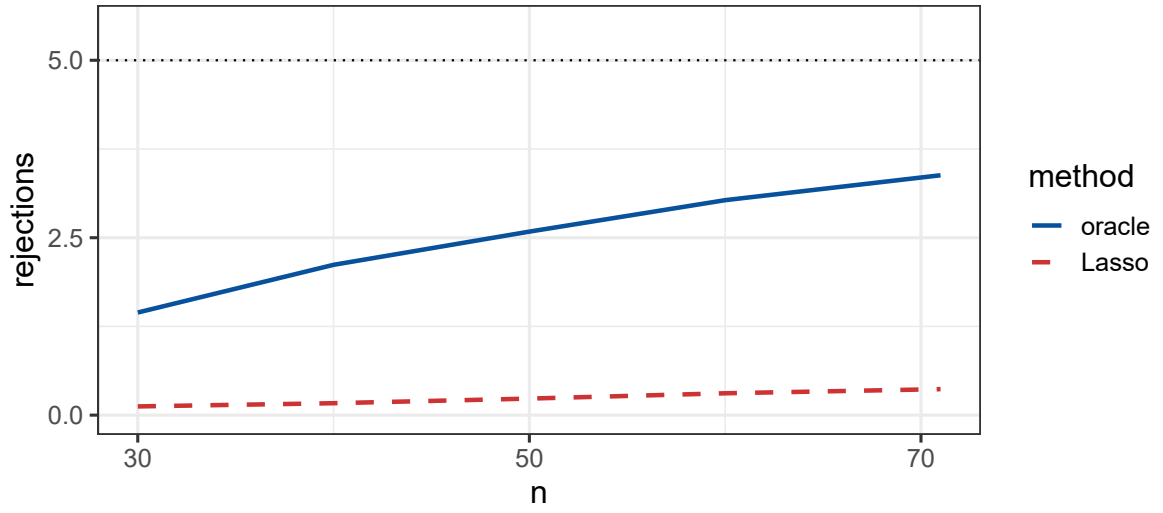

FIGURE 2.10: Real design matrix: number of rejections by sample size n, for the approximate method using oracle selection and Lasso. The dotted line corresponds to the true number of active variables.

inference on the influence of genes on the response, especially at the level of pathways, collections of genes associated with a specific biological process that interact with each other. We consider the 115 pathways contained in the KEGG database (Kanehisa and Goto, 2000).

We take $\alpha = 0.05$, $Q = 100$ and $B = 200$. Moreover, we suppose that few genes influence the response, estimating this number with $m_1 = 10$. We perform four different analyses, all based on Lasso selection (Tibshirani, 1996): (1) Multisplit with 10-fold cross-validation (dafult of the package `hdi`); (2) Multisplit with the calibration of the λ parameter suggested in Section 2.2; (3) approximate method combined with the maxT-method (Westfall and Young, 1993), as in the previous sections; (4) approximate method combined with the iterative shortcut of Section 1.7.

Analyses (1), (2) and (3) all give the same result, finding one single active gene: the negative regulatory protein YxlD. This gene is not contained in any of the considered pathways. The approximate method (3) requires around 16 seconds on a standard PC.

In analysis (4) we compute the test statistics as in (3), then we combine them through unweighted sums within the iterative shortcut. For the set of all genes, we obtain a lower $(1 - \alpha)$-confidence bound for the number of true discoveries of 4 (0.10% of all genes). When studying the 115 pathways, however, we always obtain a lower confidence bound of zero. This may be due to the fact that active genes do not appear in pathways, or to a signal too low to be detected. Computation time is around 43 seconds.

2.6 Discussion

We have considered the problem of testing multiple hypotheses in high-dimensional linear regression. Our proposed approach provides asymptotically valid resampling-based tests for any subset of hypotheses, which can be employed within closed testing procedures (Genovese and Wasserman, 2006; Goeman and Solari, 2011; Goeman *et al.*, 2019) to make confidence statements on the number of active predictor variables (TDP) within any set. These confidence statements are valid even when the subsets of interest are chosen post hoc, after seeing the data.

To construct a test for a generic subset of hypotheses, we have provided a procedure that repeatedly splits the data into two random subsets, using the first to select variables and the second to build permutation test statistics for each variable. Then statistics for any subsets can be defined by aggregating individual statistics with different functions, including the maximum and weighted sums. The computational complexity is linear in the number of splits and permutations, and polynomial in the sample size in the worst case. As the method has intensive memory usage, requiring to store many matrices, we have proposed a second procedure based on an approximation. It still defines asymptotically valid tests for individual hypotheses, and in simulations it shows satisfactory error control for sets of hypotheses in many settings. An implementation of both methods is available in the `splitFlip` package (Vesely, 2021a) in `R`.

Our method is extremely flexible, allowing different selection procedures and several combining functions. Moreover, if the combining function is a sum, then the method can be embedded into the shortcut of Chapter 1. Particular attention is to be paid to the choice of the selection procedure, as it must fulfill the method's assumptions. We suggest using the Lasso (Tibshirani, 1996) with a suitable calibration of the λ parameter, so that enough variables are selected for the screening property to be likely. More research is needed on the properties of combining functions for the individual statistics; we expect that different functions will have different power properties, and will perform best in different scenarios. As the methods are asymptotic, their behavior should be further explored with finite sample size to analyze in which cases asymptotic properties still hold. For the approximate method, we have studied error control through simulations, but further scenarios should be considered in order to establish in which situations the error control is ensured. Moreover, the test of Hemerik *et al.* (2020) that our method builds on is robust against some model misspecifications; hence it would be of interest to assess if the method maintains such robustness. Finally, it is necessary to investigate the behavior of the methods when used within closed testing procedures, and in particular

the shortcut of Chapter 1, which has the advantage of needing short computation times.

2.7 Appendix: Algorithmic implementation

We provide an outline and pseudocode for the relevant procedures presented in this chapter: the Multisplit method of Meinshausen *et al.* (2009) (Section 2.7.1), the proposed exact method (Section 2.7.2), and the corresponding approximate method (Section 2.7.3).

2.7.1 Algorithm for the Multisplit method

Algorithm 6 implements the Multisplit method (Meinshausen *et al.*, 2009) introduced in Section 2.2.1, which provides adjusted p-values $p_1, \ldots, p_m$ for all variables in high-dimensional linear regression.

Algorithm 6: Multisplit method to compute p_j for each $j \in M$.

Data: $Y \in \mathbb{R}^n$; $X \in \mathbb{R}^{n \times m}$; Q
Result: $p_1, \ldots, p_m$
$W = q \times m$ null matrix;
for $q = 1, \ldots, Q$ **do**
  randomly split $\{1, \ldots, n\}$ into $\mathcal{D}_0^q$ and $\mathcal{D}^q$;
  use $Y_{\mathcal{D}_0^q}$ and $X_{\mathcal{D}_0^q, M}$ to select variables $A^q \subseteq M$;
  for $j = 1, \ldots, m$ **do**
    if $j \in A^q$ **then**
      $\tilde{p}_j^q$ = raw p-value computed via OLS estimation with $Y_{\mathcal{D}^q}$ and $X_{\mathcal{D}^q, A^q}$;
    else
      $\tilde{p}_j^q = 1$;
    end
    $W_{qj} = p_j^q$ computed as in (2.2);
  end
end
for $j = 1, \ldots, m$ **do**
  compute p_j as in (2.3) using the j-th column of W;
end
return $p_1, \ldots, p_m$;

2.7.2 Algorithm for the exact method

Algorithm 7 implements the exact method given in Section 2.3 that uses B random sign-flipping transformations to compute the test statistics $U_j^1, \ldots, U_j^B$ for all variables $j \in M$ in high-dimensional linear regression. Results are returned as a $B \times m$ matrix of test statistics, where columns correspond to variables and rows to transformations.

Algorithm 7: Method to compute U_j^b for $j \in M$ and $b \in \{1, \dots, B\}$.

Data: $Y \in \mathbb{R}^n$; $X \in \mathbb{R}^{n \times m}$; B; Q
Result: G ($B \times m$ matrix with $G_{bj} = U_j^b$)
$G = B \times m$ null matrix;
$F_1 = n \times n$ identity matrix;
$F_2, \dots, F_B = n \times n$ diagonal matrices with elements independently and uniformly drawn from $\{-1, 1\}$;
Queue = empty list;
for $q = 1, \dots, Q$ **do**
  randomly split $\{1, \dots, n\}$ into $\mathcal{D}_0^q$ and $\mathcal{D}^q$;
  use $Y_{\mathcal{D}_0^q}$ and $X_{\mathcal{D}_0^q, M}$ to select variables $A^q \subseteq M$;
  add $(\mathcal{D}^q, A^q)$ to Queue;
end
for $j = 1, \dots, m$ **do**
  Rs = empty list;
  for $q = 1, \dots, Q$ **do**
    $(\mathcal{D}^q, A^q) = q$-th elements of Queue;
    if $j \in A^q$ **then**
      compute R_{-j}^q as in (2.9);
      add R_{-j}^q to Rs;
    end
  end
  if *Rs is empty* **then** $G_{\{1,\dots,B\},j} = (0, \dots, 0)$;
  for $b = 1, \dots, B$ **do**
    compute $C_{j,b}$ as in (2.10) using elements of Rs;
    $G_{bj} = U_j^b$ computed as in (2.11);
  end
end
return G;

The following lemma shows the worst-case computational complexity and memory usage of the algorithm. The complexity is polinomial in the sample size n, and linear both in the number Q of splits and in the number B of transformations. The memory usage is quadratic in n and linear in Q.

Lemma 2.8. *In the worst case, Algorithm 7 (excluding the variable selection procedure) has computational complexity of order n^4QB, and memory usage of order n^2Q.*

2.7.3 Algorithm for the approximate method

Algorithm 8 implements the method of Section 2.4, and represents an approximation for the procedure of Algorithm 7. It relies on B random sign-flipping transformations to compute the test statistics $\bar{U}_j^1, \dots, \bar{U}_j^B$ for all variables $j \in M$ in high-dimensional linear

regression. Results are returned as a $B \times m$ matrix of test statistics, where columns correspond to variables and rows to transformations.

Algorithm 8: Method to compute $\bar{U}_j^b$ for $j \in M$ and $b \in \{1, \dots, B\}$.

Data: $Y \in \mathbb{R}^n$; $X \in \mathbb{R}^{n \times m}$; B; Q
Result: G ($B \times m$ matrix with $G_{bj} = \bar{U}_j^b$)
$G = B \times m$ null matrix;
$F_1 = n \times n$ identity matrix;
$F_2, \dots, F_B = n \times n$ diagonal matrices with elements independently and uniformly drawn from $\{-1, 1\}$;
Queue = empty list;
for $q = 1, \dots, Q$ **do**
  randomly split $\{1, \dots, n\}$ into $\mathcal{D}_0^q$ and $\mathcal{D}^q$;
  use $Y_{\mathcal{D}_0^q}$ and $X_{\mathcal{D}_0^q, M}$ to select variables $A^q \subseteq M$;
  add $(\mathcal{D}^q, A^q)$ to Queue;
end
for $j = 1, \dots, m$ **do**
  $\bar{R}_j = n \times n$ null matrix;
  for $q = 1, \dots, Q$ **do**
    $(\mathcal{D}^q, A^q) = q$-th elements of Queue;
    if $j \in A^q$ **then**
      compute R_{-j}^q as in (2.9);
      $\bar{R}_j = \bar{R}_j + R_{-j}^q$;
    end
  end
  if $\bar{R}_j$ *is null* **then** $G_{\{1,\dots,B\},j} = (0, \dots, 0)$;
  for $b = 1, \dots, B$ **do**
    compute $\bar{C}_j$ as in (2.12);
    $G_{bj} = \bar{U}_j^b$ computed as in (2.13);
  end
end
return G;

As shown in the following lemma, the computational complexity and the memory usage are lower than those of Algorithm 7.

Lemma 2.9. *In the worst case, Algorithm 8 (excluding the variable selection procedure) has computational complexity of order $n^4 Q + n^3 B$, and memory usage of order n^2.*

2.8 Appendix: Simulations

In this section, we give additional information on the simulations of Section 2.5.1. We provide results for the simulated matrix with $m = 1000$ and the computation time

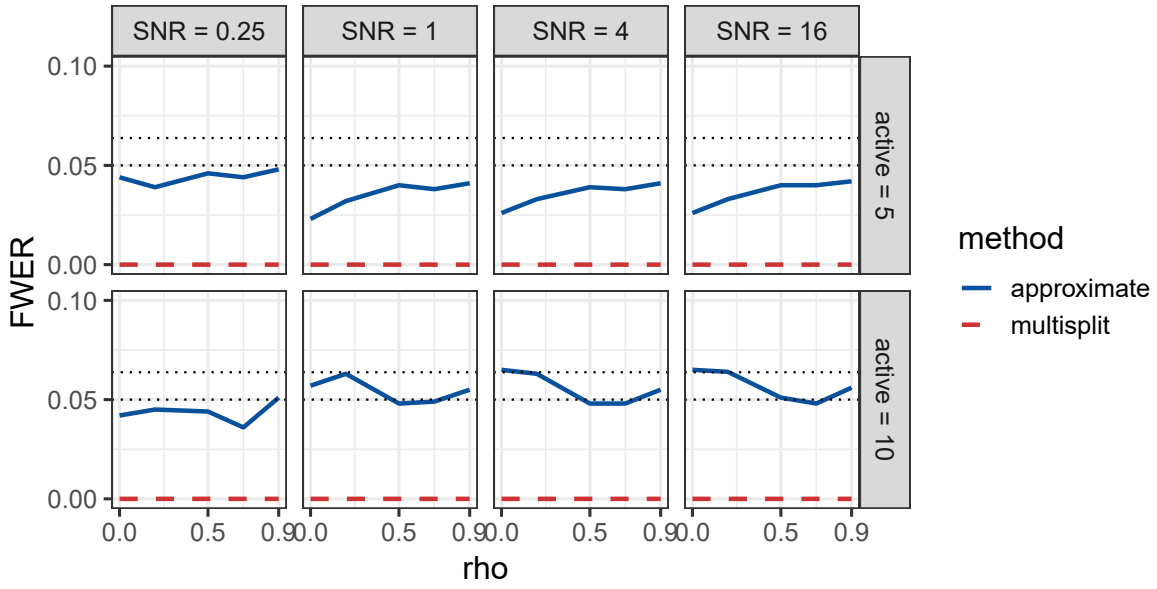

FIGURE 2.11: Simulated design matrix with $m = 1000$ variables: FWER by covariance parameter ρ, for the approximate method and the Multisplit. *Active* and SNR denote the true number of active variables and the signal-to-noise ratio. The dotted lines correspond to the significance level $\alpha = 0.05$ and an upper bound (α plus two standard deviations, approximately 0.063).

TABLE 2.2: Simulated and real design matrices with m variables: maximum computation time (s) for the approximate method and the Multisplit.

	simulated		real
m	100	1000	4088
approximate	34.2	137.7	58.8
multisplit	0.08	0.33	1.3

for the simulations of Section 2.5.1.2, as well as the computation time for Section 2.5.1.3.

2.8.1 Approximate and Multisplit

Figures 2.11 and 2.12 show the FWER and the number of rejections for the simulated design matrix with $m = 1000$. These display an analogous behavior to results for $m = 100$ (Figure 2.4). Table 2.2 contains the maximum computation time over the different settings for both the simulated and real design matrices.

2.8.2 Oracle and Lasso

Figures 2.13 and 2.14 present the computation time needed by the approximate method using oracle selection and Lasso, for different sample sizes n. As expected, time

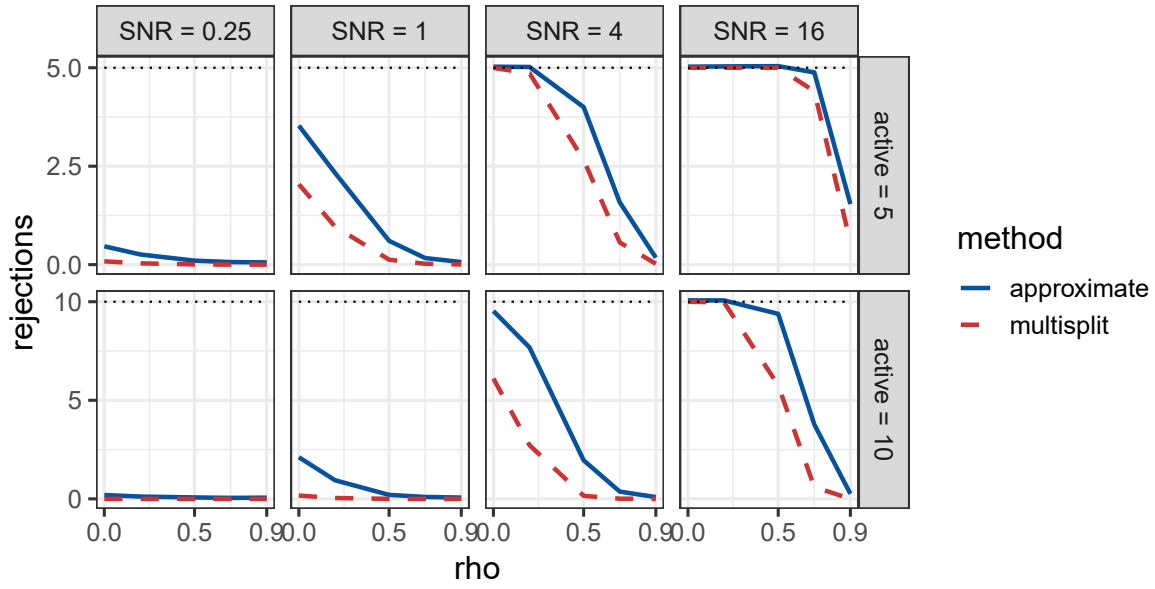

FIGURE 2.12: Simulated design matrix with $m = 1000$ variables: number of rejections by covariance parameter ρ, for the approximate method and the Multisplit. *Active* and SNR denote the true number of active variables and the signal-to-noise ratio. The dotted line corresponds to *active*.

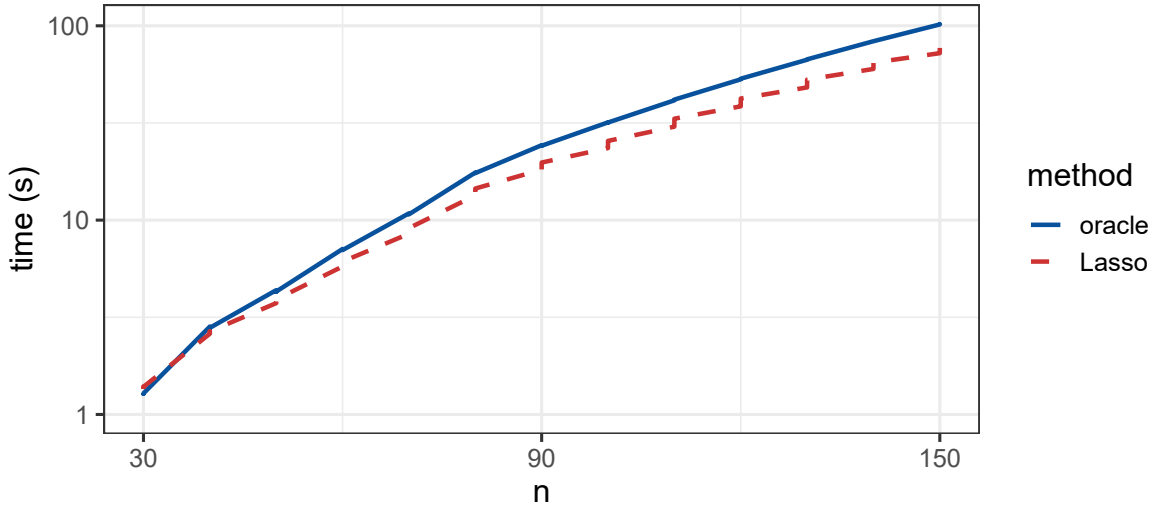

FIGURE 2.13: Simulated design matrix: maximum computation time (log scale) by sample size n, for the approximate method using oracle selection and Lasso.

increases with n for both selection procedures.

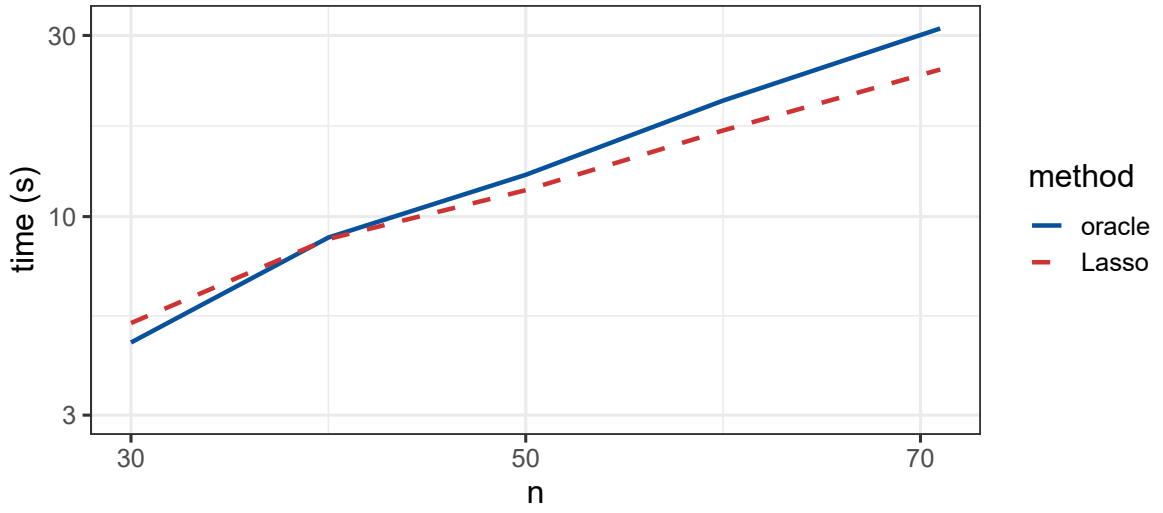

FIGURE 2.14: Real design matrix: maximum computation time (log scale) by sample size n, for the approximate method using oracle selection and Lasso.

2.9 Appendix: Proofs

2.9.1 Projection matrices

Here we recall some properties of projection matrices, which will be used within proofs.

1. If a matrix $R \in \mathbb{R}^{n \times n}$ is symmetric ($R^\top = R$) and idempotent $RR = R$, then it is a projection matrix. As a consequence, it is positive semi-definite, i.e., $z^\top Rz \geq 0$ for any $z \in \mathbb{R}^n$.

2. For any variable $j \in M$, the residual maker matrix R_{-j} defined in (2.4) is a projection matrix with $R_{-j}X_{-j} = 0$.

Proposition 2.3. *The test that rejects H_j when $T_j^1 > T_j^{(\omega)}$ is asymptotically an α-level test for any $j \in M$. For finite n, it may be anti-conservative as*

$$var(T_j^1) \geq var(T_j^b) \qquad (b \in \{1, \ldots, B\}).$$

Proof. Proof of the first part of the Proposition in a more general case is in Hemerik *et al.* (2020) (see Theorem 2). We briefly recall the main steps in our notation.

Assume that H_j is true, so that

$$Y = X_{-j}\beta_{-j} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 I).$$

Consider the B-dimensional vector of effective scores

$$\mathbf{V}_j = (V_j^1, \dots, V_j^B)^\top, \qquad V_j^b = t_{j,b}^\top Y,$$

such that $T_j^b = |V_j^b|$ for all b. First the Authors prove that for any transformation b

$$V_j^b = V_j^{*b} + o_{P(\beta)}, \qquad V_j^{*b} = \frac{1}{\sqrt{n}} X_j^\top R_{-j} F_b \varepsilon = \frac{1}{\sqrt{n}} t_{j,1}^\top F_b \varepsilon \tag{2.14}$$

and so $\mathbf{V}_j$ is asymptotically equivalent to $\mathbf{V}_j^* = (V_j^{*1}, \dots, V_j^{*B})^\top$ as $n \to \infty$. Since

$$\mathbf{V}_j^* \sim \mathcal{N}_B \left(0, \frac{\sigma^2 \|t_{j,1}\|^2}{n} I\right),$$

we have

$$\mathbf{V}_j, \mathbf{V}_j^* \xrightarrow[n\to\infty]{\text{d}} Z \sim \mathcal{N}_B \left(0, \xi^2 I\right), \qquad \xi^2 = \sigma^2 \lim_{n\to\infty} \frac{\|t_{j,1}\|^2}{n}.$$

As a consequence, the statistics $T_j^1, \dots, T_j^B$ converge to i.i.d. random variables. This implies that, under H_j,

$$\lim_{n\to\infty} P\left(T_j^1 > T_j^{(\omega)}\right) = \frac{\lfloor \alpha B \rfloor}{B} \leq \alpha$$

(Lemma 1 in Hemerik *et al.* (2020)).

Proof of the second part of the Proposition is in Finos *et al.* (2021). To show that $\text{var}(T_j) \geq \text{var}(T_j^b)$ for any b, it is sufficient to observe that

$$\text{var}(T_j^1) - \text{var}(T_j^b) = \frac{\sigma^2}{n} \text{E}\left[t_{j,1}^\top (I - F_b R_{-j} F_b) t_{j,1}\right] \geq 0$$

since $I - F_b R_{-j} F_b$ is a projection matrix, and so positive semi-definite (see Section 2.9.1). □

Theorem 2.4. *The test that rejects H_j when $\tilde{T}_j^1 > \tilde{T}_j^{(\omega)}$ is an α-level test for any $j \in M$.*

Proof. Proof of the Proposition for the more general case of generalized linear models is in Finos *et al.* (2021), and we recall it here for the case of the linear model.

Assume that H_j is true, so that

$$Y = X_{-j}\beta_{-j} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 I).$$

Consider the B-dimensional vector of standardized scores

$$\tilde{\mathbf{V}}_j = (\tilde{V}_j^1, \dots, \tilde{V}_j^B)^\top, \qquad \tilde{V}_j^b = \tilde{t}_{j,b}^\top Y,$$

such that $\tilde{T}_j^b = |\tilde{V}_j^b|$ for all b, as well as any couple of transformations $b, c \in \{1, \dots, B\}$. By the properties of the residual maker matrix R_{-j} (see Section 2.9.1),

$$\tilde{V}_j^b = \tilde{t}_{j,b}^\top Y = \tilde{t}_{j,b}^\top \varepsilon$$

and $\tilde{\mathbf{V}}_j$ follows a multivariate normal distribution with mean zero. Moreover, since the sign-flipping matrices $F_2, \dots, F_B$ are independent with mean zero,

$$\operatorname{cov}(\tilde{V}_j^b, \tilde{V}_j^c) = \sigma^2 \mathrm{E}\left[\tilde{t}_{j,b}^\top \tilde{t}_{j,c}\right] = \begin{cases} \sigma^2 & \text{if} \quad b = c \\ 0 & \text{otherwise} \end{cases}$$

and so $\tilde{\mathbf{V}}_j \sim \mathcal{N}_B(0, \sigma^2 I)$ is a vector of i.i.d. random variables. As a consequence, $\tilde{T}_j^1, \dots, \tilde{T}_j^B$ are i.i.d. random variables and, by the Monte Carlo testing principle (Lehmann and Romano, 2005),

$$P\left(\tilde{T}_j^1 > \tilde{T}_j^{(\omega)}\right) = \frac{\lfloor \alpha B \rfloor}{B} \leq \alpha.$$

□

Lemma 2.5. *The test that rejects H_S when $T_S^1 > T_S^{(\omega)}$ is asymptotically an α-level test for any non-empty $S \subseteq M$.*

Proof. Suppose that H_S is true. As each H_j with $j \in S$ is individually true, we can write

$$Y = X_{-j_1}\beta_{-j_1} + \varepsilon = \ldots = X_{-j_s}\beta_{-j_s} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}_n(0, \sigma^2 I).$$

Analogously to the proof of Proposition 2.3, define the sB-dimensional vector of effective scores

$$\mathbf{V}_S = (V_{j_1}^1, \dots, V_{j_1}^B, \dots, V_{j_s}^1, \dots, V_{j_s}^B)^\top, \qquad V_j^b = t_{j,b}^\top Y,$$

such that $T_j^b = |V_j^b|$ for all variables j and transformations b.

Consider any couple of variables $j, h \in S$ and any couple of transformations $b, c \in \{1, \dots, B\}$. Recall that R_{-j_h} is a projection matrix (see Section 2.9.1), and $F_2, \dots, F_B$ are independent with mean zero. From (2.14),

$$V_j^b = V_j^{*b} + o_{P(\beta)}, \qquad V_j^{*b} = t_{j,1}^\top F_b \varepsilon$$

and so $\mathbf{V}_S$ is asymptotically equivalent to

$$\mathbf{V}_S^* = (V_{j_1}^{*1}, \ldots, V_{j_1}^{*B}, \ldots, V_{j_s}^{*1}, \ldots, V_{j_s}^{*B})^\top$$

as $n \to \infty$. Moreover, $\mathbf{V}_S^*$ follows a multivariate normal distribution with mean zero and

$$\operatorname{cov}\left(V_j^{*b}, V_h^{*c}\right) = \sigma^2 \mathrm{E}\left[t_{j,1}^\top F_b F_c t_{h,1}\right] = \begin{cases} \sigma^2 t_{j,1}^\top t_{h,1} & \text{if} \quad b = c \\ 0 \quad \text{otherwise.} \end{cases}$$

Then

$$\mathbf{V}_S, \mathbf{V}_S^* \xrightarrow[n\to\infty]{\mathrm{d}} Z \sim \mathcal{N}_{sB}\left(0, \Xi \otimes I\right)$$

where $\otimes$ denotes the Kronecker product and

$$I \in \mathbb{R}^{B\times B}, \qquad \Xi = (\xi_{k\ell}) \in \mathbb{R}^{s\times s}, \qquad \xi_{k\ell} = \sigma^2 \lim_{n\to\infty} t_{j_k,1}^\top t_{j_\ell,1}.$$

Equivalently, we can say that

$$\begin{pmatrix} V_{j_1}^1 & \ldots & V_{j_s}^1 \\ \vdots & & \vdots \\ V_{j_1}^B & \ldots & V_{j_s}^B \end{pmatrix} \xrightarrow[n\to\infty]{\mathrm{d}} Z' \sim \mathcal{MN}_{s\times B}\left(0, I, \Xi\right)$$

where $\mathcal{MN}_{s\times B}$ denote the matrix normal distribution.

As the B vectors $(V_{j_1}^1, \ldots, V_{j_s}^1), \ldots, (V_{j_1}^B, \ldots, V_{j_s}^B)$ converge to i.i.d. random vectors, then also $(T_{j_1}^1, \ldots, T_{j_s}^1), \ldots, (T_{j_1}^B, \ldots, T_{j_s}^B)$ converge to i.i.d. random vectors. Therefore the combinations of their elements $T_S^1, \ldots, T_S^B$ defined in (2.8) converge to i.i.d. random variables. Moreover, for each variable j high values of T_j^1 correspond to evidence against H_j, and g is increasing in each argument. Therefore high values of T_S correspond to evidence against H_S. From Hemerik *et al.* (2020) (see Lemma 1), it follows that

$$\lim_{n\to\infty} P\left(T_S^1 > T_S^{(\omega)}\right) = \frac{\lfloor \alpha B \rfloor}{B} \leq \alpha.$$

□

Theorem 2.6. *The test that rejects H_j when $U_j^1 > U_j^{(\omega)}$ is an α-level test for any $j \in M$. Moreover, the test that rejects H_S when $U_S^1 > U_S^{(\omega)}$ is asymptotically an α-level test for any non-empty $S \subseteq M$.*

Proof. Assume that all active variables are selected by the variable selection procedure;

by Assumption 2.2, this is true at least asymptotically, so this assumption does not affect asymptotic results. For simplicity of notation suppose that $Q = 2$; the same reasoning applies to the more general case. Finally, denote by $b, c \in \{1, \dots, B\}$ any couple of transformations.

First, assume that H_j is true. For simplicity, suppose that j is selected in both splits. For $q \in \{1, 2\}$, as H_j is true and all active variables are contained in A^q we can write

$$Y_{\mathcal{D}^q} = X_{-j;\mathcal{D}^q,A^q}\beta_{-j;A^q} + \varepsilon_{\mathcal{D}^q}, \qquad \varepsilon_{\mathcal{D}^q} \sim \mathcal{N}_{n/2}(0, \sigma^2 I). \tag{2.15}$$

considering only observations in $\mathcal{D}^q$ and variables in A^q. Since the matrix R^q_{-j} given in (2.4) has non-null elements only corresponding to observations in $\mathcal{D}^q$, we have $R^q_{-j}Y = R^q_{-j}\varepsilon$. Hence

$$C_{j,b}Y = (R^1_{-j}F_bR^1_{-j} + R^2_{-j}F_bR^2_{-j})Y = C_{j,b}\varepsilon$$
$$u^\top_{j,b}Y = \frac{X_j^\top C_{j,b}Y}{\|C_{j,b}X_j\|} = u^\top_{j,b}\varepsilon.$$

Proof for the first part of the theorem is the same as for Theorem 2.4, substituting $\tilde{T}^b_j$ with U^b_j, and $\tilde{t}_{j,b}$ with $u_{j,b}$.

Subsequently, assume that H_S is true, and consider any couple of variables $j, h \in S$. For simplicity, suppose that j and h are selected in both splits. For $q \in \{1, 2\}$, by the structure of R^q_{-j} the effective score for the model in (2.15) can be written as

$$V^{qb}_j = \frac{1}{\sqrt{n}}X^\top_{j;\mathcal{D}^q}R^q_{-j;\mathcal{D}^q,\mathcal{D}^q}F_{b;\mathcal{D}^q,\mathcal{D}^q}R^q_{-j;\mathcal{D}^q,\mathcal{D}^q}Y_{\mathcal{D}^q} = \frac{1}{\sqrt{n}}X_j^\top R^q_{-j}F_bR^q_{-j}Y.$$

From (2.14), V^{qb}_j is asymptotically equivalent to

$$V^{*qb}_j = \frac{1}{\sqrt{n}}X_j^\top R^q_{-j}F_b\varepsilon$$

and so the sB-dimensional vectors

$$\mathbf{V}_S = (V^1_{j_1}, \dots, V^B_{j_1}, \dots, V^1_{j_s}, \dots, V^B_{j_s})^\top, \qquad V^b_j = V^{1b}_j + V^{2b}_j$$
$$\mathbf{V}^*_S = (V^{*1}_{j_1}, \dots, V^{*B}_{j_1}, \dots, V^{*1}_{j_s}, \dots, V^{*B}_{j_s})^\top, \qquad V^{*b}_j = V^{*1b}_j + V^{*2b}_j$$

are asymptotically equivalent. As in the proof of Lemma 2.5, we observe that $\mathbf{V}_S^*$ follows a multivariate normal distribution with mean zero and

$$\begin{aligned}\operatorname{cov}\left(V_j^{*b}, V_h^{*c}\right) &= \frac{\sigma^2}{n}\mathrm{E}\left[X_j^\top (R_{-j}^1 + R_{-j}^2) F_b F_c (R_{-h}^1 + R_{-h}^2) X_h\right] \\ &= \begin{cases} \sigma^2 X_j^\top C_{j,b} C_{h,b} X_h & \text{if} \quad b = c \\ 0 \quad \text{otherwise.} \end{cases}\end{aligned}$$

Therefore

$$\mathbf{V}_S, \mathbf{V}_S^* \xrightarrow[n\to\infty]{\mathrm{d}} Z \sim \mathcal{N}_{sB}\left(0, \Xi \otimes I\right)$$

with

$$I \in \mathbb{R}^{B\times B}, \qquad \Xi = (\xi_{k\ell}) \in \mathbb{R}^{s\times s}, \qquad \xi_{k\ell} = \sigma^2 \lim_{n\to\infty} \frac{1}{n} X_{j_k}^\top C_{j_k,b} C_{j_\ell,b} X_{j_\ell}$$

and the B vectors $(V_{j_1}^1, \ldots, V_{j_s}^1), \ldots, (V_{j_1}^B, \ldots, V_{j_s}^B)$ converge to i.i.d. random vectors. Finally, for each $j \in S$ and each transformation b

$$U_j^b = \sigma^2 \|V_j^b\|^{-1} |V_j^b|$$

and so also the vectors $(U_{j_1}^1, \ldots, U_{j_s}^1), \ldots, (U_{j_1}^B, \ldots, U_{j_s}^B)$ converge to i.i.d. random vectors. As observed in the proof of Lemma 2.5, this means that the combinations of their elements $U_S^1, \ldots, U_S^B$ converge to i.i.d. random variables, and

$$\lim_{n\to\infty} P\left(U_S^1 > U_S^{(\omega)}\right) = \frac{\lfloor \alpha B \rfloor}{B} \leq \alpha.$$

□

Theorem 2.7 *The test that rejects H_S when $\bar{U}_S^1 > \bar{U}_S^{(\omega)}$ is asymptotically an α-level test.*

Proof. Proof is the same as for the first part of Theorem 2.6, observing that

$$\bar{R}_{-j} Y = \sum_{q\,:\, j\in A^q} R_{-j}^q Y = \bar{R}_{-j}\varepsilon$$

and so

$$\bar{u}_{j,b}^\top Y = \frac{X_j^\top \bar{R}_{-j} F_b \bar{R}_{-j} Y}{\|\bar{R}_{-j} F_b \bar{R}_{-j} X_j\|} = \bar{u}_{j,b}^\top \varepsilon.$$

□

Lemma 2.8 *In the worst case, Algorithm 7 (excluding the variable selection procedure) has computational complexity of order n^4QB, and memory usage of order n^2Q.*

Proof. Fix any $j \in M$, and denote the number of splits where j is selected with $s_j = |\{q : j \in A^q\}|$. Recall that, for square matrices of size n, the computational complexity of multiplication, transposition and inversion is of order n^3. Hence computing R^q_{-j} as in (2.9) for all splits that select j requires $n^3 s_j$ operations. Computing $C_{j,b}$ as in (2.10) and U^b_j as in (2.11) for all transformations requires $n^3 B s_j$ and $n^3 B$ operations, respectively.

Therefore the total complexity of the algorithm is order

$$n^3 B S_{\text{tot}}, \qquad S_{\text{tot}} = \sum_{j \in M} s_j.$$

In the worst case, where we select $n/2$ variables in each split, we have $S_{\text{tot}} = nQ/2$, and so the complexity is of order n^4QB.

Moreover, for each variable j the algorithm needs to store s_j square matrices of size n, with memory usage of order $n^2 s_j$. In the worst case, $s_j = Q$, and so the memory usage is of order n^2Q. □

Lemma 2.9 *In the worst case, Algorithm 8 (excluding the variable selection procedure) has computational complexity of order $n^4Q + n^3B$, and memory usage of order n^2.*

Proof. Analogously to the proof of Lemma 2.8, fix any $j \in M$, and denote the number of splits where j is selected with $s_j = |\{q : j \in A^q\}|$. Computing R^q_{-j} as in (2.9) and $\bar{R}_j$ as in (2.12) for all splits that select j requires $n^3 s_j$ and $n^2 s_j$ operations, respectively. Computing $\bar{C}_j$ as in (2.12) and $\bar{U}^b_j$ as in (2.13) for all transformations requires $n^3 B$ operations.

Therefore the total complexity of the algorithm is order

$$n^3(B + S_{\text{tot}}), \qquad S_{\text{tot}} = \sum_{j \in M} s_j.$$

In the worst case, where we select $n/2$ variables in each split, we have $S_{\text{tot}} = nQ/2$, and so the complexity is of order $n^4Q + n^3B$.

Moreover, for each variable j the algorithm needs to store only 2 square matrices of size n, R^q_{-j} and $\bar{R}_j$. Hence the memory usage is of order n^2. □

Conclusions

Discussion

In this manuscript we have considered the problem of testing multiple hypotheses in high-dimensional settings, arguing that more tools are needed to support an exploratory approach, where one may want to test many subsets of hypotheses and select the subsets of interest post hoc. We have focused on resampling-based methods, as these rely on minimal assumptions and, in general, offer an improvement in power over the parametric approach, especially in presence of multiple hypotheses. We have provided two general and flexible methods to perform multiple testing on high-dimensional data: a method to make confidence statements on the proportion of true discoveries (TDP), and a method to make inference on predictor variables in linear regression.

In Chapter 1 we have focused on sum tests, a popular and broad class of global tests with the characteristic of using sums to aggregate signal from multiple features. Using results from the closed testing framework, we have argued that all global tests automatically come with an inbuilt selective inference method, through which we can make many additional inferences without paying a price in terms of the global test's α-level. This allows in particular to construct confidence sets for the TDP simultaneously over all subsets of hypotheses, so that the confidence sets are valid even under post-hoc selection. To compute these TDP confidence sets, we have proposed a general closed testing procedure in terms of an iterative shortcut for the full closed testing method that relies on permutation testing. The shortcut converges to full closed testing results, but can be stopped at any number of iterations, still providing valid confidence sets for the TDP. It is exact and extremely flexible, as it applies to any sum test and adapts to any data correlation structure.

In Chapter 2 we have considered hypotheses testing on linear regression coefficients. We have provided an approach to construct asymptotically valid resampling-based tests for any subset of hypotheses, which can be used in closed testing procedures, as well as the shortcut of Chapter 1. The approach is presented in two ways: an exact method, and an approximate method that is less computationally intensive. In this framework,

to build test statistics for any set of hypotheses it is sufficient to define test statistics for individual hypotheses, relying on a variable selection procedure, and then combine these through a suitable function. The resulting method is extremely flexible, allowing different selection procedures and several combining functions.

Future directions of research

In Chapter 1 we have argued that different sum tests have different power properties, depending on the setting, and that permutations generally give an improvement over approaches based on worst-case distributions. More research is needed to study the behavior of different sum tests, as well as to quantify the improvement given by permutations. Moreover, the method may be compared with other permutation-based procedures that rely on bounding functions such as those proposed in Andreella *et al.* (2020) and Blanchard *et al.* (2020).

With regard to the approach given in Chapter 2, many aspects still need to be investigated. First, different combining functions may be explored, in order to assess their properties in different scenarios. Moreover, as the proposed exact and approximate methods are asymptotic, their behavior should be further explored with finite sample size to analyze in which cases asymptotic properties still hold. For the approximate method, error control has been shown in simulations, but further scenarios should be considered in order to establish in which situations it is ensured. Moreover, the test of Hemerik *et al.* (2020) that the methods build on is robust against some model misspecifications, and so it would be of interest to assess if the methods maintain such robustness. Finally, it is necessary to investigate the behavior of the methods when used within closed testing procedures, and in particular the shortcut of Chapter 1, which has the advantage of needing short computation times.

Bibliography

Andreella, A., Hemerik, J., Weeda, W., Finos, L. and Goeman, J. J. (2020) Permutation-based true discovery proportions for fMRI cluster analysis. Unpublished.

Benjamini, Y. and Hochberg, Y. (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **57**, 289–300.

Bühlmann, P. (2013) Statistical significance in high-dimensional linear models. *Bernoulli* **19**, 1212–1242.

Biernacka, J. M., Jenkins, G. D., Wang, L., Moyer, A. M. and Fridley, B. L. (2012) Use of the gamma method for self-contained gene-set analysis of snp data. *European Journal of Human Genetics* **20**, 565–571.

Blanchard, G., Neuvial, P. and Roquain, E. (2020) Post hoc confidence bounds on false positives using reference families. *Annals of Statistics* **48**(3), 1281–1303.

Dai, H., Leeder, J. S. and Cui, Y. (2014) A modified generalized fisher method for combining probabilities from dependent tests. *Frontiers in Genetics* **5**.

Dezeure, R., Bühlmann, P., Meier, L. and Meinshausen, N. (2015) High-dimensional inference: confidence intervals, p-values and R-software hdi. *Statistical Science* **30**(4), 533–558.

Dezeure, R., Bühlmann, P. and Zhang, C.-H. (2017) High-dimensional simultaneous inference with the bootstrap. *TEST* **26**, 685–719.

Donoho, D. and Jin, J. (2015) Higher criticism for large-scale inference, especially for rare and weak effects. *Statistical Science* **30**(1), 1–25.

Dudbridge, F. and Koeleman, B. P. C. (2003) Rank truncated product of p-values, with application to genomewide association scans. *Genetic Epidemiology* **25**(4), 360–366.

Ebrahimpoor, M., Spitali, P., Hettne, K., Tsonaka, R. and Goeman, J. (2020) Simultaneous enrichment analysis of all possible gene-sets: unifying self-contained and competitive methods. *Briefings in Bioinformatics* **21**(4), 1302–1312.

Edgington, E. S. (1972) An additive method for combining probability values from independent experiments. *The Journal of Psychology* **80**(2), 351–363.

Ernst, M. D. (2004) Permutation methods: a basis for exact inference. *Statistical Science* **19**(4), 676–685.

Finos, L. (2003) *Metodi Non Parametrici per l'Analisi Multi-Focus e per il Controllo della Molteplicità con Applicazioni in Ambito Biomedico.* Ph.D. thesis, Department of Statistical Sciences, University of Padova.

Finos, L., Hemerik, J. and Goeman, J. J. (2021) Unpublished.

Fisher, R. A. (1925) *Statistical Methods for Research Workers.* Edinburgh: Oliver and Boyd.

Fisher, R. A. (1936) "the coefficient of racial likeness" and the future of craniometry. *The Journal of the Royal Anthropological Institute of Great Britain and Ireland* **66**, 57–63.

Genovese, C. R. and Wasserman, L. (2006) Exceedance control of the false discovery proportion. *Journal of the American Statistical Association* **101**(476), 1408–1417.

Goeman, J. J., van de Geer, S. A. and van Houwelingen, H. C. (2006) Testing against a high dimensional alternative. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 477–493.

Goeman, J. J., Hemerik, J. and Solari, A. (2021) Only closed testing procedures are admissible for controlling false discovery proportions. *The Annals of Statistics* **49**(2), 1218–1238.

Goeman, J. J., Meijer, R. J., Krebs, T. J. P. and Solari, A. (2019) Simultaneous control of all false discovery proportions in large-scale multiple hypothesis testing. *Biometrika* **106**(4), 841–856.

Goeman, J. J. and Solari, A. (2010) The sequential rejection principle of familywise error control. *Annals of Statistics* **38**(6), 3782–3810.

Goeman, J. J. and Solari, A. (2011) Multiple testing for exploratory research. *Statistical Science* **26**(4), 584–597.

Hemerik, J. and Goeman, J. J. (2018a) Exact testing with random permutations. *TEST* **27**, 811–825.

Hemerik, J. and Goeman, J. J. (2018b) False discovery proportion estimation by permutations: confidence for significance analysis of microarrays. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **80**(1), 137–155.

Hemerik, J., Goeman, J. J. and Finos, L. (2020) Robust testing in generalized linear models by sign flipping score contributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **82**(3), 841–864.

Hemerik, J., Solari, A. and Goeman, J. J. (2019) Permutation-based simultaneous confidence bounds for the false discovery proportion. *Biometrika* **106**(3), 635–649.

Hoeffding, W. (1952) The large-sample power of tests based on permutations of observations. *The Annals of Mathematical Statistics* **23**(2), 169–192.

Huang, Y., Xu, H., Calian, V. and Hsu, J. C. (2006) To permute or not to permute. *Bioinformatics* **22**(18), 2244–2248.

Kanehisa, M. and Goto, S. (2000) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research* **28**(1), 27–30.

Kelly, A. M. C., Uddin, L. Q., Biswal, B. B., Castellanos, F. X. and Milham, M. P. (2008) Competition between functional brain networks mediates behavioral variability. *Neuroimage* **39**(1), 527–537.

Knuth, D. (1998) *The Art of Computer Programming*. Volume 3. Boston: Addison-Wesley.

Kriegeskorte, N., Simmons, W. K., Bellgowan, P. S. F. and Baker, C. I. (2009) Circular analysis in systems neuroscience: the dangers of double dipping. *Nature Neuroscience* **12**, 535–540.

Kuo, C.-L. and Zaykin, D. V. (2011) Novel rank-based approaches for discovery and replication in genomewide association studies. *Genetics* **189**(1), 329–340.

Lancaster, H. O. (1961) The combination of probabilities: an application of orthonormal functions. *Australian Journal of Statistics* **3**, 20–33.

Land, A. H. and Doig, A. G. (1960) An automatic method of solving discrete programming problems. *Econometrica* **28**(3), 497–520.

Langsrud, Ø. (2005) Rotation tests. *Statistics and Computing* **15**, 53–60.

Lee, J. D., Sun, D. L., Sun, Y. and Taylor, J. E. (2016) Exact post-selection inference, with application to the lasso. *Annals of Statistics* **44**(3), 907–927.

Lehmann, E. L. and Romano, J. P. (2005) *Testing Statistical Hypotheses*. New York: Springer.

Li, J. and Tseng, G. C. (2011) An adaptively weighted statistic for detecting differential gene expression when combining multiple transcriptomic studies. *Annals of Applied Statistics* **5**(2A), 994–1019.

Lindquist, M. A. (2008) The statistical analysis of fMRI data. *Statistical Science* **23**(4), 439–464.

Liptak, T. (1958) On the combination of independent tests. *Magyar Tudományos Akadémia Matematikai Kutató Intezetenek Kozlemenyei* **3**, 1971–1977.

Liu, Y. and Xie, J. (2020) Cauchy combination test: a powerful test with analytic p-value calculation under arbitrary dependency structures. *Journal of the American Statistical Association* **115**(529), 393–402.

Loughin, T. M. (2004) A systematic comparison of methods for combining p-values from independent tests. *Computational Statistics & Data Analysis* **47**(3), 467 485.

Marcus, R., Peritz, E. and Gabriel, K. R. (1976) On closed testing procedures with special reference to ordered analysis of variance. *Biometrika* **63**(3), 655–660.

Meier, L., Dezeure, R., Meinshausen, N., Maechler, M. and Bühlmann, P. (2021) *hdi: High-Dimensional Inference*. R package version 0.1-9.

Meijer, R. J. and Goeman, J. J. (2016) Multiple testing of gene sets from gene ontology: possibilities and pitfalls. *Briefings in Bioinformatics* **17**(5), 808–818.

Meinshausen, N. (2006) False discovery control for multiple tests of association under general dependence. *Scandinavian Journal of Statistics* **33**(2), 227–237.

Meinshausen, N. and Bühlmann, P. (2010) Stability selection. *Journal of the Royal Statistical Society: Series B (Methodological)* **104**(488), 417–473.

Meinshausen, N., Meier, L. and Bühlmann, P. (2009) p-values for high-dimensional regression. *Journal of the American Statistical Association* **104**(488), 1671–1681.

Mitten, L. G. (1970) Branch-and-bound methods: general formulation and properties. *Operations Research* **18**(1), 24–34.

Nichols, T. E. (2012) Multiple testing corrections, nonparametric methods, and random field theory. *NeuroImage* **62**(2), 811–815.

Pearson, K. (1933) On a method of determining whether a sample of size n supposed to have been drawn from a parent population having a known probability integral has probably been drawn at random. *Biometrika* **25**(3-4), 379–410.

Pernet, C. R., McAleer, P., Latinus, M., Gorgolewski, K. J., Charest, I., Bestelmeyer, P. E. G., Watson, R. H., Fleming, D., Crabbe, F., Valdes-Sosa, M. and Belin, P. (2015) The human voice areas: Spatial organization and inter-individual variability in temporal and extra-temporal cortices. *NeuroImage* **119**, 164–174.

Pesarin, F. (2001) *Multivariate Permutation Tests: with Applications in Biostatistics*. New York: Wiley.

Pesarin, F. and Salmaso, L. (2010) *Permutation Tests for Complex Data: Theory, Applications and Software*. New York: Wiley.

R Core Team (2017) *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing.

Rosenblatt, J. D., Finos, L., Weeda, W. D., Solari, A. and Goeman, J. J. (2018) All-resolutions inference for brain imaging. *NeuroImage* **181**, 786–796.

Simes, R. J. (1986) An improved Bonferroni procedure for multiple tests of significance. *Biometrika* **73**(3), 751–754.

Solari, A., Finos, L. and Goeman, J. J. (2014) Rotation-based multiple testing in the multivariate linear model. *Biometrics* **70**(4), 954–961.

Southworth, L. K., Kim, S. K. and Owencorresponding, A. B. (2009) Properties of balanced permutations. *Journal of Computational Biology* **16**(4), 625–638.

Tian, J., Chen, X., Katsevich, E., Goeman, J. J. and Ramdas, A. (2021) Large-scale simultaneous inference under dependence. Unpublished.

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* **58**(1), 267–288.

University of Padova (2017) *CAPRI: Calcolo ad Alte Prestazioni per la Ricerca e l'Innovazione*. Strategic Research Infrastructure Grant.

Vesely, A. (2021a) *splitFlip: Permutation-based multisplit*. R package version 1.1.0.

Vesely, A. (2021b) *sumSome: Permutation true discovery guarantee by sum-based tests*. R package version 1.1.0.

Vovk, V. and Wang, R. (2020) Combining p-values via averaging. *Biometrika* **asaa027**.

Wasserman, L. and Roeder, K. (2009) High dimensional variable selection. *Annals of Statistics* **37**(5A), 2178–2201.

Westfall, P. H. and Young, S. S. (1993) *Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment*. New York: Wiley.

Wilson, D. J. (2019) The harmonic mean p-value for combining dependent tests. *Proceedings of the National Academy of Sciences* **116**(4), 1195–1200.

Winkler, A. M., Ridgway, G. R., Webster, M. A., Smith, S. M. and Nichols, T. E. (2014) Permutation inference for the general linear model. *NeuroImage* **92**(6), 381–397.

Won, S., Morris, N., Lu, Q. and Elston, R. C. (2009) Choosing an optimal method to combine p-values. *Statistics in Medicine* **28**(11), 1537–1553.

Woo, C.-W., Krishnan, A. and Wager, T. D. (2014) Cluster-extent based thresholding in fmri analyses: pitfalls and recommendations. *Neuroimage* **33**, 412–419.

Wu, M. C., Lee, S., Cai, T., Li, Y., Boehnke, M. and Lin, X. (2011) Rare-variant association testing for sequencing data with the sequence kernel association test. *The American Journal of Human Genetics* **89**(1), 82–93.

Yu, K., Li, Q., Bergen, A. W., Pfeiffer, R. M., Rosenberg, P. S., Caporaso, N., Kraft, P. and Chatterjee, N. (2009) Pathway analysis by adaptive combination of p-values. *Genetical Epidemiology* **33**, 700–709.

Zaykin, D. V., Zhivotovsky, L. A., Czika, W., Shao, S. and Wolfinger, R. D. (2007) Combining p-values in large-scale genomics experiments. *Pharmaceutical Statististics* **6**, 217–226.

Zaykin, D. V., Zhivotovsky, L. A., Westfall, P. H. and Weir, B. S. (2002) Truncated product method for combining p-values. *Genetic Epidemiology* **22**, 170–185.

Zhang, C.-H. and Zhang, S. S. (2014) Confidence intervals for low dimensional parameters in high dimensional linear modelsù. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **76**, 217–242.

Anna Vesely

CURRICULUM VITAE

Contact Information

University of Padova
Department of Developmental Psychology and Socialisation
via Venezia, 8
35131 Padova, Italy

Tel. +39 049 827 6500
e-mail: anna.vesely@unipd.it

Current Position

Since January 2022
Research Fellow
University of Padova
Department of Developmental Psychology and Socialisation

Research interests

- High-dimensional data
- Multiple hypothesis testing and selective inference
- Permutation testing
- Applications to psychometrics, neuroscience and biology in general

Education

October 2018 – December 2021
PhD (*dottorato*) in Statistical Sciences
University of Padova, Department of Statistical Sciences
Thesis title: 'Resampling-based methods for multiple testing on high-dimensional data'
Supervisor: Prof. Livio Finos
Co-supervisor: Prof. Jelle J. Goeman

September 2016 – September 2018
Master (*laurea magistrale*) degree in Statistical Sciences
University of Bologna, Department of Statistical Sciences
Title of dissertation: 'Agent-based models of migration decision-making: a case study of Italy'
Supervisor: Prof. Roberto Impicciatore
Final mark: 110/110 *cum laude*

September 2011 – March 2016
Bachelor degree (*laurea triennale*) in Mathematics
University of Bologna, Department of Mathematics
Title of dissertation: 'Grafi aleatori: il modello di Erdős-Rényi' ('Random graphs: Erdős-Rényi model')
Supervisor: Prof. Massimo Ferri
Final mark: 103/110

Visiting period

February 2020 – July 2020
Leiden University Medical Center
Leiden, The Netherlands
Supervisor: Prof. Jelle J. Goeman

Further education

February 2021
Winter school MRInference
University of Padova

July 2019
Data Research Camp
University of Padova

Computer skills

- Programming languages: R, C++, Python, MATLAB (base)
- Statistical packages: R software, SAS (*Certification SAS Base Programming for SAS 9*), STATA, NetLogo
- Other softwares: LaTeX, GIT, Moodle

Language skills

Italian: native; English: fluent; French: moderate.

Publications

Articles in journals

Vesely A., Finos L., Goeman J. J., Andreella A. (2021). Valid double-dipping via permutation-based closed testing. *Book of Short Papers SIS 2021*, 776–781.

Zandonella Callegher C., Bertoldo G., Toffalini E., Vesely A., Andreella A., Pastore M., Altoé G. (2021). PRDA: An R package for Prospective and Retrospective Design Analysis. *Journal of Open Source Software*.

Mancini F., Gandolfi S., Marudi A., Vesely A., Righini G., Canalini A., Oddolini F., Cremonini C., Bertellini E., Bandiera G. (2019). The role of Early Head CT following the OHCA survival: results based on patients enrolled in RIAC. *Italian Journal of Emergency Medicine*.

Working papers

Vesely A., Finos L., Goeman J. J. (2021). Permutation-based true discovery guarantee by sum tests. *arXiv* 2102.11759.

Conference presentations

Vesely A., Finos L., Goeman J. J. (2021). Permutation-based true discovery guarantee by sum tests. *31st International Biometric Conference*, Riga, Latvia, July 2022 (programmed).

Vesely A., Goeman J. J., Finos L. (2022). Resampling-based inference for high-dimensional regression. (invited) *18th Conference of the Spanish Biometric Society*, Madrid, Spain, May 2022 (programmed).

Vesely A., Finos L., Goeman J. J. (2021). Permutation-based true discovery guarantee by sum tests. *13th Virtual Conference of the Italian Region of the International Biometric Society*, online, 29 September 2021.

Vesely A., Finos L., Goeman J. J., Andreella A. (2021). Valid double-dipping via permutation-based closed testing. *50th Scientific Meeting of the Italian Statistical Society*, Pisa, Italy, 22 June 2021.

Vesely A., Finos L., Goeman J. J. (2021). Permutation-based true discovery guarantee by sum tests. (invited) *International Seminar on Selective Inference*, online, 25 February 2021.

Vesely A., Impicciatore R. (2021). Out-migration decision-making in Italy: an agent-based model. *StaTalk 2019 @ UniBO*, Bologna, Italy, 29 March 2019.

Teaching experience

October 2021 - January 2022
Psychological testing (testing psicologico)
Bachelor degree in Developmental Psychology and Socialisation
Exercises and laboratory, 12 hours
University of Padova
Instructor: Prof. Gianmarco Altoé

August 2021
Introduction to LaTeX
Master degree in Statistical Sciences
Laboratory, 2.5 hours
University of Padova
Instructor: Prof. Francesco Lisi

References

Prof. Livio Finos
Department of Developmental
Psychology and Socialisation
Padova, Italy
e-mail: livio.finos@unipd.it

Prof. Jelle J. Goeman
Department of Biomedical Data Sciences,
Leiden University Medical Center
Leiden, The Netherlands
e-mail: j.j.goeman@lumc.nl