Using Score Distributions to Compare Statistical Significance Tests for Information Retrieval Evaluation

Javier Parapar* David E. Losada† Manuel A. Presedo-Quindimil* Alvaro Barreiro*

*IRLab, Dept of Computer Science
†Universidade da Coruña
Campus de Elviña
15071, A Coruña (Spain)

†Centro Singular de Investigación
en Tecnoloxías da Información (CiTIUS)
Universidade de Santiago de Compostela
Campus Vida
15782, Santiago de Compostela (Spain)

javierparapar@udc.es david.losada@usc.es mpresedo@udc.es barreiro@udc.es

(Submitted to JASIST Oct 11, 2017; 1st revision May 22, 2018; 2nd revision Sep 5, 2018; accepted Jan 11, 2019)

Abstract

Statistical significance tests can provide evidence that the observed difference in performance between two methods is not due to chance. In Information Retrieval, some studies have examined the validity and suitability of such tests for comparing search systems. We argue here that current methods for assessing the reliability of statistical tests suffer from some methodological weaknesses, and we propose a novel way to study significance tests for retrieval evaluation. Using Score Distributions, we model the output of multiple search systems, produce simulated search results from such models, and compare them using various significance tests. A key strength of this approach is that we assess statistical tests under perfect knowledge about the truth or falseness of the null hypothesis. This new method for studying the power of significance tests in Information Retrieval evaluation is formal and innovative. Following this type of analysis, we found that both the sign test and Wilcoxon signed test have more power than the permutation test and the t-test. The sign test and Wilcoxon signed test also have a good behavior in terms of type I errors. The bootstrap test shows few type I errors, but it has less power than the other methods tested.

1 Introduction

Use of significance tests is a de facto standard of evaluation in Information Retrieval (IR). In the early days of IR experimentation, researchers tended to prefer the Wilcoxon signed-rank test and the sign test, which are simple and make few assumptions about the data. Other parametric alternatives, such as Student's t-test, require data drawn from specific distributions (e.g. Gaussian), but the output of retrieval experiments violates this assumption.

A number of studies have analysed the reliability of significance tests (Zobel, 1998; Voorhees & Buckley, 2002; Urbano, Marrero & Martín, 2013; Sanderson & Zobel, 2005; Cormack & Lynam, 2007; Sakai, 2016; Smucker, Allan & Carterette, 2007). Many of them suggest that, despite its assumptions, the t-test performs well and it should be preferred over the Wilcoxon signed-rank test and the sign test. Smucker and colleagues argued that the use of the Wilcoxon signed-rank test and the sign test should cease (Smucker et al., 2007), and have influenced a change towards the t-test and the permutation test. The way in which experimenters employ statistical testing is crucial and affects the dissemination of research results, as it has been demonstrated in other fields (Miettunen & Nieminen, 2003). A comprehensive and solid analysis of the relative merits of each significance test is therefore essential for IR experimentation.

A significance test stands on a null hypothesis ($H0$) and an alternative hypothesis ($H1$). In IR, many experiments run paired two-sided tests. In such a case, the null hypothesis states that the two retrieval outputs under

examination are drawn from the same population (any difference between them is due to chance), and the alternative hypothesis states that they are drawn from different populations. The test estimates the probability p (p-value) of observing a difference at least as large as that produced by the experiment given that $H0$ is true.

The studies on the reliability of significance tests for IR evaluation have followed two main methodologies. *Query splitting* methods split the topics of a test collection into two groups, run each significance test in each group, and check the coherence of the results. Smucker and colleagues followed an alternative approach (Smucker et al., 2007), where the p-values estimated by a permutation (or randomisation) test are the main reference and other significance tests are evaluated *in comparison to the permutation test*. We claim here that both methodologies have severe limitations and propose an innovative way for assessing significance tests.

Query splitting methods are limited to assess the consistency of a significance test with itself. But the results of the test can be consistently wrong over the splits (rejecting a true $H0$ –type I error– or failing to reject a false $H0$ –type II error–). With no knowledge of the truth or falseness of $H0$, we should not just equate consistency with reliability. Comparing significance tests based on the permutation test, as done in (Smucker et al., 2007), is not exempt from problems either. The permutation test can compute a good approximation to the exact p-values, but we should not produce miss or false alarm rates from such p-values. Given a certain significance level (α) and the p-values estimated by permutation, the miss and false alarm rates of each significance test are measured based on the agreement between the test's decisions and the permutation test's decisions. Such an approach implicitly assumes that those cases where the p-value produced by permutation is above α are cases where $H0$ is true and, conversely, those cases below α are cases where $H0$ is false. This rule compromises the quality of the analysis because the permutation test is not error-free. For example, with $\alpha = .05$, the permutation test would be making an average of 5% type I errors (no difference between the systems, but the permutation test says otherwise). As such, in 5% of cases, giving blind faith to the permutation test unfairly penalises any significance test that makes the correct decision (any significance test that is skeptical about the difference is actually right!). Likewise, the permutation test makes some type II errors and accepting such permutation test's decisions is unfair to those tests that detect the difference. In summary, the main flaw of existing methods to assess the reliability of significance tests is that they make strong assumptions about the truth value of the null hypothesis. Although the previous studies substantially contributed to analysing the use of significance tests in IR, we believe that a more robust methodology, based on actual knowledge about $H0$, can be designed. This is precisely the primary aim of our paper.

We propose a method for assessing the reliability of significance tests that works with simulated results of retrieval systems. We take many executions (runs) from TREC systems, and we model IR systems using Score Distributions (SDs) (Manmatha, Rath & Feng, 2001; Arampatzis & Robertson, 2011) learnt from those real runs. With the modeled systems, we can produce multiple retrieval results by sampling from the distributions. The core idea is that we can compare significance tests under complete knowledge about the null hypothesis. For example, we can experiment with the same system producing two ranked lists (null hypothesis true) or we can experiment with two different systems producing each a ranked list (null hypothesis false). Systems with distinctive retrieval characteristics can be obtained by manipulating the parameters of the inferred statistical distributions. Furthermore, this simulation method can be repeated as many times as needed, serving to reinforce the validity of the study.

We examined the tests with this innovative methodology and found some results that agree with those presented by Smucker and colleagues (Smucker et al., 2007): the permutation and the t-test tend to agree with each other and, to a lesser degree, with bootstrap; while the Wilcoxon test and the sign test disagree with both the permutation and the t-test. However, we found substantive evidence that suggests that these differences occur because the Wilcoxon and the sign test have higher power than the permutation test. These results are in agreement with the findings reported by authoritative statistical studies, which showed the relative lack of power of the permutation test (see Section 5.11 (Conover, 1999)).

The main contributions of this paper are:

1. a new method for assessing the reliability of the statistical tests based on the analysis of their power using data derived from simulated IR systems.

2. a complete empirical study that categorically concludes that, in typical IR experimentation, the Wilcoxon and sign tests are more powerful than the permutation, bootstrap or the t-test.

2 Related Work

Non-parametric tests of significance, such as the Wilcoxon test and the sign test, have been widely used in IR experiments. Both tests assume data drawn from continuous distributions, while retrieval experiments produce discrete data. Despite this fact, Van Rijsbergen suggested conservative use of such tests (Van Rijsbergen, 1979). The t-test has been also used regularly in IR. Hull (Hull, 1993) claimed that it often performs well even when the normality assumption is violated. Other popular tests in IR are the bootstrap test (Savoy, 1997; Cormack & Lynam, 2006) and the permutation test (Smucker et al., 2007).

Several papers studied the reliability of the significance tests. Zobel (Zobel, 1998) made multiple pairwise-comparisons of systems with two disjoint query sets. A type I error was recorded when a test observed a significant difference between two systems on the first query set and the ordering of the systems was different on the second query set. He concluded that the Wilcoxon test is more reliable and has higher power than both the t-test and ANOVA. Sanderson and Zobel (Sanderson & Zobel, 2005), who expanded a previous study by (Voorhees & Buckley, 2002), found that the t-test shows lower error rates when compared to the sign and the Wilcoxon tests. Cormack and Lynam (Cormack & Lynam, 2007) also performed a query-splitting study and observed that the t-test, Wilcoxon and the sign test are highly accurate and have high power, with the t-test proving superior overall. Urbano et al. (Urbano et al., 2013) also presented a comparison of significance tests based on query-splitting. Regarding *safety* (smallest error rates across significance levels), the t-test was the best, followed by the Wilcoxon test (for low levels of significance) and the permutation test (for the usual levels of significance). The bootstrap test consistently produced smaller p-values and, thus, it was the most powerful across significance levels. The authors also studied the agreement of the tests with themselves, and the Wilcoxon test turned out to be the most stable of all for small p-values, while the t-test the most stable overall. The permutation test was also evaluated in this study, but it was not optimal under any criterion considered. All these studies provide valuable evidence for IR experimentation. However, the query-splitting methodology makes an arbitrary division of topics and lacks real knowledge about the truth or falseness of the null hypothesis. The fact that a significance test gives consistent results over two splits of topics should not be considered a measure of quality. As a matter of fact, the test might be consistently wrong over the splits (consistently rejecting a true null hypothesis or consistently accepting a false null hypothesis).

Smucker and colleagues (Smucker et al., 2007) found that the bootstrap test, the t-test and the permutation test produce comparable p-values, and showed that both the Wilcoxon and the sign test are discordant with the permutation test. Following their experiments, they suggested discontinuing the use of these two tests. Our study also reveals a discordance between these two tests and the rest of the tests, but we argue that the output of the permutation test must not be taken as the main reference. Smucker et al.'s study evaluated significance tests in terms of how well each test matched the decisions of the permutation test. For example, a given test was assigned a false alarm when the test labeled a difference as significant while the permutation test considered it as non-significant (see e.g. Table 2, (Smucker et al., 2007)). We claim that assigning this kind of *ground truth role* to the permutation test limits the conclusions that can be drawn from the analysis. We show here that the Wilcoxon test and the sign test have higher power than the other tests and make a low number of type I errors. Therefore, significant differences found with these two tests must not be ignored on the basis that other tests fail to identify these differences.

Sakai (Sakai, 2016) compared two versions of two-sample t-tests: Student's t-test and Welch's t-test. He also employed a query-splitting methodology but his approach has some restricted knowledge about the null hypothesis. More specifically, he modeled the case of $H0$ being true as follows. Given a query set of size n and m runs that processed these queries, the queries are randomly partitioned into two sets. For each of the runs and a given evaluation metric, he conducted a two-sided, two-sample test to determine whether or not the difference between the two means for the same run are statistically significant. The ground truth was that they are not, since the

scores actually come from the same system. This strategy, based on unpaired data, cannot be applied to evaluate significance tests under current TREC-like exercises (where all systems are evaluated with the same queries). However, as argued by Sakai, possible applications of two-sample tests in IR include comparing set of clicks from two search engines or comparing the difficulty of two test collections using the same search system.

Our paper is also related to some studies that also employed simulation in order to analyze other aspects of IR evaluation. For example, Urbano (Urbano, 2016) performed a study on test collection reliability that compared a number of measures and estimators of test collection accuracy. His method was based on stochastic simulation of evaluation results and, through large-scale simulation from TREC data, the bias of several estimators of test collection accuracy was analyzed.

Score distributions (Arampatzis, Robertson & Kamps, 2009) have been studied and modeled since the early days of IR (Swets, 1963). Different combinations of statistical distributions have been proposed for modeling the score distributions of relevant and non-relevant documents (two Gaussians (Swets, 1963), two negative exponentials (Swets, 1969), two Poissons (Bookstein, 1977), two Gammas (Baumgarten, 1999), a Gaussian and a negative exponential (Arampatzis, Beney, Koster & van der Weide, 2000; Manmatha et al., 2001; Arampatzis, Kamps & Robertson, 2009), or a Gaussian and a Gamma (Kanoulas, Pavlu & Dai, 2009; Dai, Kanoulas, Pavlu & Aslam, 2011)). Some studies (Kanoulas, Dai, Pavlu & Aslam, 2010) have also analysed SDs based on the scoring formulas of the retrieval models. Manmatha et al. (Manmatha et al., 2001) proposed the use of SDs to combine the outputs of multiple search engines. Arampatzis et al. (Arampatzis et al., 2000; Arampatzis, Kamps & Robertson, 2009) utilised SD models for threshold optimisation in a legal search task. Cummins employed SDs for query performance prediction (Cummins, 2014). Arampatzis et al. experimented with SDs in image retrieval (Arampatzis, Zagoris & Chatzichristofis, 2013), Parapar et al. employed SDs in pseudo-relevance feedback (Parapar, Presedo-Quindimil & Barreiro, 2014), and Losada at al. proposed a rank fusion approach based on SDs for prioritising assessments in IR evaluation (Losada, Parapar & Barreiro, 2018).

3 Analysing Significance Tests with Score Distribution Models

Score distributions model the way in which search systems generate retrieval scores. We can, therefore, simulate multiple search systems and evaluate their output using Average Precision. Different conditions –null hypothesis true/false– can be generated, and significance tests can be evaluated accordingly. Our method proceeds as follows:

1. For every (TREC run, query) pair, we learn a SD model (a mixture of statistical distributions) from the list of scores supplied by the run.

2. For each run we take its 50 (as many as different queries) mixtures and we experiment with the case of a true null hypothesis (same model producing two outputs):

   (a) we randomly extract 1000 samples from the SD model and obtain a *synthetic* list of scores and their relevance values (the method extracts samples from either the distribution of relevant documents or the distribution of non-relevant documents and, thus, each extracted score is assigned a relevance label, 1 if the score came from the relevant document distribution and 0 if the score came from the non-relevant document distribution). The resulting list of scores is sorted in descending order.

   (b) we repeat step (a) and obtain a second *synthetic* list.

   (c) we compute the average precision of both lists.

   Given the two sequences of 50 APs obtained, a significance test is run for assessing the significance of the difference found.

3. Step 2 is repeated 1000 times and we record the average number of times that the significance test falsely rejects $H0$.

4. Now, we experiment with the case of a true alternative hypothesis (the outputs come from different models). For each run we take its 50 (as many as different queries) mixtures and

   (a) we randomly extract 1000 samples from the SD model and obtain a *synthetic* list of scores and their relevance values. The resulting list of scores is sorted in descending order.

   (b) we alter the SD model's parameters, sample from the modified model and obtain a second *synthetic* list.

   (c) we compute the average precision of both lists.

   Given the two sequences of 50 APs obtained, a significance test is run for assessing the significance of the difference found.

5. Step 4 is repeated 1000 times and we record the average number of times that the significance tests correctly rejects $H0$.

6. Steps 4-5 are repeated several times by gradually separating the modified model from the original model (parameter manipulation of the mixture).

With this procedure, we can straightforwardly estimate the power of a significance test, $p(Reject\ H0|H0\ is\ false)$, the probability of a type I error, $p(Reject\ H0|H0\ is\ true)$, and so forth. The pseudo-code that implements this procedure is available in the Appendix.

3.1 Modeling Information Retrieval Systems

With real search systems, a comparative analysis of significance tests is tricky. We have no knowledge of the underlying retrieval models that generate the search results and, thus, we know nothing about the truth of the null hypothesis. Furthermore, we often observe a small number of executions from each search system. Such limited sample imposes limitations on the statistical analysis. By modeling search systems with statistical models we can produce as many samples as required and, additionally, we have certainty about the null hypothesis.

Score distribution models assume that the distribution generating the scores of relevant documents is different from the one producing the scores of non-relevant documents. Various combinations of distributions have been employed to model each group of documents, and the parameters of the mixture distribution can be learned from the observed documents' scores. We employ here a two log-normal distribution (Cummins, 2011), which adheres to the recall-fallout convexity hypothesis (Robertson, 2007), and shows higher goodness of fit when compared with other alternatives (Cummins, 2011; Cummins & O'Riordan, 2012).

Each retrieval system has a set of mixtures (one mixture per query). For every query $q^{(i)}$, the mixture has two log-normal distributions: $L_0^{(i)}$ for the non-relevant documents and $L_1^{(i)}$ for the relevant documents. If $P(s|1)^{(i)}$ is the probability density function (pdf) for the scores (s) of relevant documents and $P(s|0)^{(i)}$ is the pdf for the scores of non-relevant documents then the mixture is:

$$P(s)^{(i)} = \lambda^{(i)} P(s|1)^{(i)} + (1 - \lambda^{(i)}) P(s|0)^{(i)} \tag{1}$$

where $\lambda^{(i)}$, the mixture parameter, represents the proportion of relevant documents returned by the system for query $q^{(i)}$.

Some TREC runs produce negative scores and we followed the standard procudure of shifting all scores of these runs by some constant factor. We discarded those runs that do not supply retrieval scores for the returned documents. The use of alternative SD models that do not rely on the existence of scores (Robertson, Kanoulas & Yilmaz, 2013) is left to future work.

For each query $q^{(i)}$, TREC supplies a set of relevance assessments. Given the TREC run and the relevance assessments, we proceeded to learn the run's SD model as follows. The mixing weight is set to the proportion of relevant documents returned by the run. The scores of the relevant documents returned by the run are used to learn

the parameters of $L_1^{(i)}$. This learning stage was done following a maximum likelihood approach[1] (Dempster, Laird & Rubin, 1977), which provides better goodness of fit than the method of moments (Cummins, 2011). Following usual practice in pooling-based IR, the set of non-relevant documents retrieved by the run is composed of the judged non-relevant documents plus the unjudged documents. The procedure to learn $L_0^{(i)}$ from this set was the same than that used to learn $L_1^{(i)}$ from the set of relevant documents.

The ultimate objective of our study is to analyse significance tests with a series of per-query results. The procedure sketched above leads to SD models that can generate multiple system outputs in the form of lists of retrieval scores. Classic IR measures, such as Mean Average Precision (MAP), rely on relevance judgments at document level. By sampling separately from the two component distributions (e.g., see step 2(a) in the procedure sketched above), we ensure that each sampled score is assigned a relevance value. In this way, the resulting ranking of scores can be evaluated with any performance measure, such as MAP.

3.2 Power Curves

Given a certain level of significance, a common way to analyse significance tests consists of studying the form of the power curve. A power curve plots the probability of rejecting $H0$ against increasing differences between the systems being compared. Figure 1 shows an example of a power curve. The y-axis represents the probability of rejecting the null hypothesis ($P(Reject\ H0)$), while the x-axis represents the difference between the systems that are compared. The 0.0 point of the x-axis (leftmost point) corresponds to the case where $H0$ is true ($x = 0.0$, no difference between the systems). The rest of the points of the x-axis ($x > 0.0$) correspond to cases where $H0$ is false (and the larger x is, the more different the systems are). The height of the leftmost point represents the probability of a type I error (incorrect rejection of the null hypothesis). Typically, the power curve starts from a height equal to the significance level, rises smoothly and monotonically and, eventually, reaches the maximum probability of 1. The closer the curve is to a right angle, the more power the test has.

Such an analysis must only be done under certainty about the hypothesis being tested. Previous studies on significance tests for IR lack such certainty. By design, our methodology produces outputs where we know whether or not the test must reject $H0$. To simulate the case when $H0$ is true we just have to obtain two random samples from the same SD model (same system producing two outputs), and compare the two series of APs (one pair of samples per query). To simulate the case when $H0$ is false, we proceed by taking a SD model and altering its parameters. In this way, we can compare the original SD model against a modified SD model. Given the outputs produced by these two models, we are certain that the series of APs come from different statistical distributions. Note that the modifications are done on a per-query basis because each system has a SD model learnt for each query. There are multiple possibilities to modify the original SD model. For example, by changing $\lambda^{(i)}$ we can simulate the increase or decrease in the number of relevant documents returned. We decided to leave the same $\lambda^{(i)}$ for both models because changing $\lambda^{(i)}$ models a change in the proportion of relevant documents returned. In TREC, the number of relevant documents is fixed –for each query, we only have assessments for the pooled documents– and, as a consequence, it does not make sense to model a system that returns more relevant documents than those available in the pool. Another possibility consists of changing the mean of L_0 (or L_1), which has the effect of altering the position in the ranking of the non-relevant (or relevant) documents. We opted to gradually increase the mean (location) of L_1 (μ_1). This choice simulates relevant documents moving up in the ranking and, thus, improves the effectiveness of the original model unequivocally and monotonically. Our report of significance tests will, therefore, include power curve plots where the x-axis represents the percentage of increase in μ_1. In Section 4.3, we provide empirical evidence showing that increasing μ_1 leads to improvements in AP.

[1] We performed direct optimization of the log-likelihood with the Nelder-Mead method (Nelder & Mead, 1965) using the `fitdistrplus` R package: http://cran.r-project.org/web/packages/fitdistrplus/fitdistrplus.pdf.

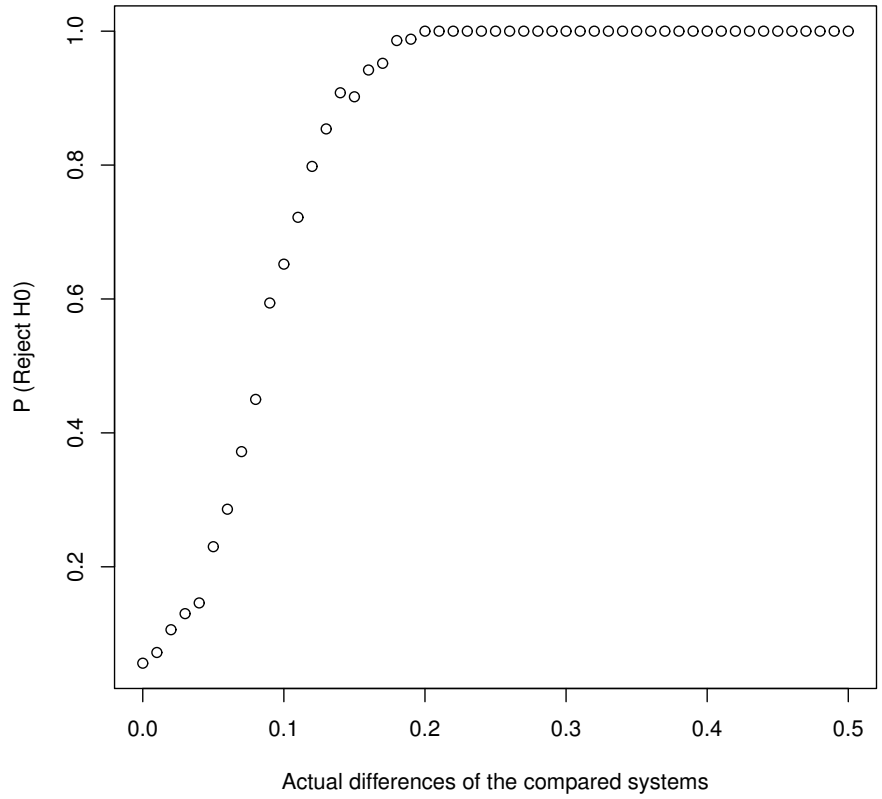


Figure 1: Example of Power Curve for the Wilcoxon Test. The X axis represents increasing differences between the systems being compared. The null hypothesis, H0, only holds in the leftmost point, the other points correspond with H0 false.

Table 1: Summary of TREC system runs used in the experiments.

Edition	Query set	Relevant Documents	Systems	Used systems
TREC 3	151-200	9805	40	32
TREC 5	251-300	5524	61	38
TREC 6	301-350	4611	74	48
TREC 7	351-400	4674	103	62
TREC 8	401-450	4728	130	88

4 Experiments

We performed a thorough analysis of several significance tests using the ad-hoc retrieval runs submitted to TREC 3, 5, 6, 7 and 8 (Voorhees & Harman, 2005). The number of systems that we used in our experimentation (see Table 1) is lower than the official number of participants because we removed the systems that did not return retrieval scores or retrieved very few documents per query. We worked with the top 1000 documents ranked by the systems and we set the number of random samples generated from the SD models to 1000.

4.1 Significance Tests

We compared the following tests: t-test, Wilcoxon signed rank test, sign test, permutation test and bootstrap. The comparison was done for the two-sided case using paired data.

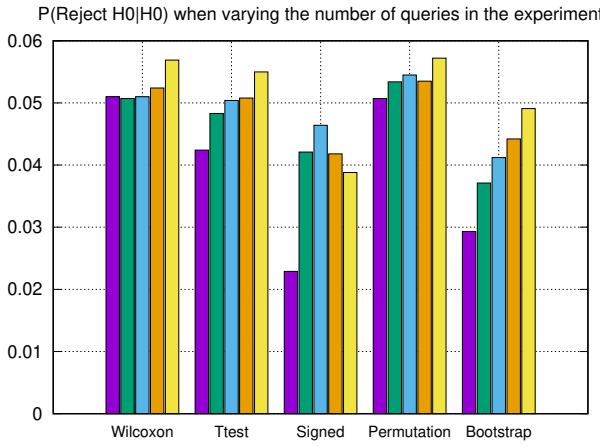
The t-test assumes that the obtained differences follow a normal distribution. The null hypothesis is that the mean of the distribution of differences is zero. Wilcoxon assumes that the differences can be ranked, but ignores the magnitude of the differences. The rank values are assigned the sign of the measured difference, and the null hypothesis is that the sum of the positive ranks is the same as the sum of negative ranks. The sign test relies on even less stringent assumptions: under the null hypothesis, we would expect the same number of positive and negative differences. The permutation test is free of mathematical assumptions. The null hypothesis is that the two systems are identical and any permutation of the matched pair observations will produce an output equally probable. Given a statistic for the test and computing all possible permutations of the observed values it is possible to compute the exact p-value. The null hypothesis of the bootstrap test is that the observed values are random samples from the same distribution.

For the t-test, Wilcoxon, sign test, and permutation test we used the implementation of the JSC (Java Statistical Classes) by Andrew Bertie from The Open University. For bootstrap, we implemented the test as described in (Efron & Tibshirani, 1993) (*the one-sample problem*). Following (Smucker et al., 2007), we took the difference of means as the statistic for both permutation and bootstrap, and we extracted 100.000 samples of random permutations. We also considered the popular t-test statistic for permutation and bootstrap but these two tests performed better when the difference of means was the statistic.

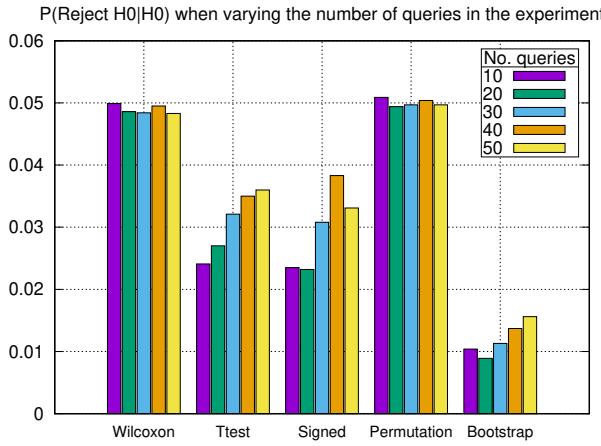
4.2 Experiments: Type I Error and Power Curves

To estimate the probability of a type I error, $p(Reject\ H0|H0\ is\ true)$, we produced two samples from the same SD model and ran the significance test. This procedure was repeated 1000 times for each system and we averaged –over all TREC systems– the number of times that $H0$ was rejected (the significance level was set to 0.05). Figure 2 shows the results of this experiment with varying number of queries.

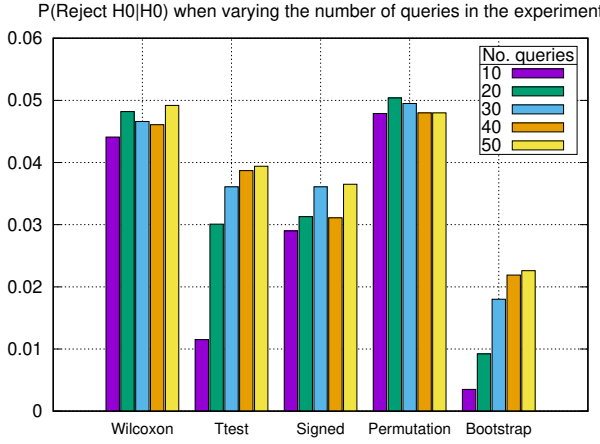
Wilcoxon and the permutation test tend to achieve the expected probability of rejecting $H0$ when $H0$ is true (the significance level). The other three tests are more conservative and show a probability of rejecting $H0$ lower than the significance level. Such conservative behavior becomes apparent with small query sets. Bootstrap is extremely conservative (particularly in TREC 5 and 6, where the retrieval performance of the systems is the lowest). By design, tests are expected to have 5% of type I errors and, thus, the t-test, the sign test and bootstrap do



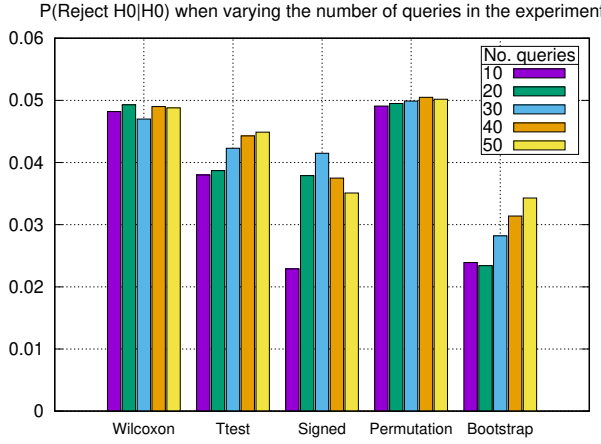
(a) TREC 3



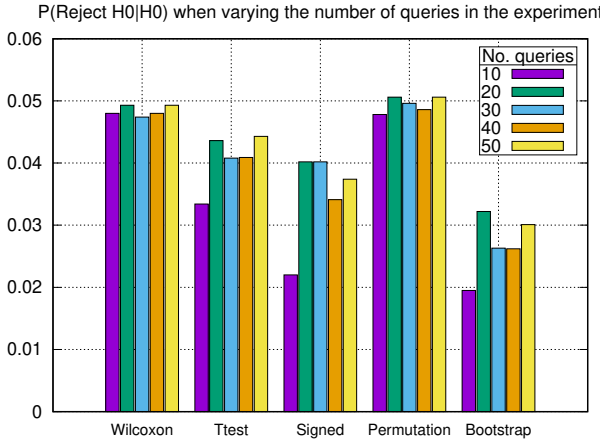
(b) TREC 5



(c) TREC 6



(d) TREC 7



(e) TREC 8

Figure 2: Average $p(Reject\ H0|H0\ is\ true)$ in different TREC collections ($\alpha = 0.05$)

not behave as expected. From a practical perspective, these three tests are making fewer errors, which might seem convenient. However, from a statistical standpoint, this result suggests that the p-values obtained by these tests are worse estimations of the probability of finding the observed difference when the null hypothesis is true. We also experimented with other significance levels –from 0.01 to 0.25, steps of 0.01– and found consistent results: Wilcoxon and the permutation test yielded results that matched with the significance level, while the other three tests made fewer errors than expected.

Let us now analyse the tests when $H0$ is false. Each experiment compared one SD model against a transformed model, and we obtained different transformed models by increasing $\mu_1^{(i)}$ up to 30% (in steps of 0.5%). The resulting power curves, which plot the probability of rejection of the null hypothesis (averaged over all systems and samples), are shown in Figures 3, 4, 5, 6 and 7.

With less than 30 queries, all tests perform poorly (many type II errors). This result is in line with previous studies (Webber, Moffat & Zobel, 2008) and provides further empirical evidence on the need for large sets of queries when we want to derive statistical conclusions about the superiority of one system over the other. Many query splitting studies analysed significance tests using less than 30 queries in each split and, therefore, their results must be taken with caution. Our discussion below therefore focuses on the graphs with more than 30 queries (Figures 3, 4, 5, 6 and 7).

The form of each power curve is related to the difficulty of the track. For example, on average, TREC-3 has many relevant documents and, thus, the differences in performance between systems are higher and easier to detect by the tests. As a consequence, TREC-3 plots tend to show a right angle.

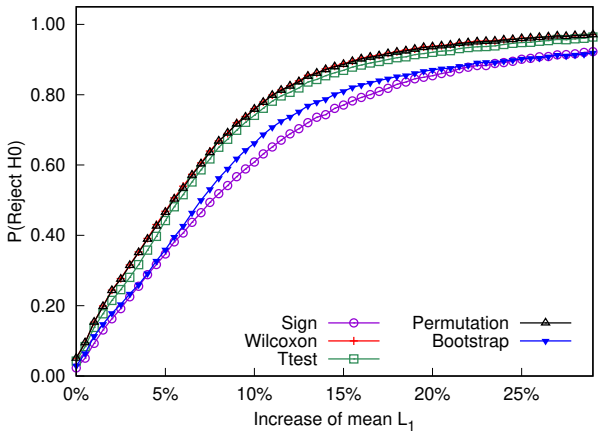
The power curves are quite revealing on the relative merits of the significance tests. The sign test and Wilcoxon perform the best at rejecting $H0$ when it is false. The permutation test, the t-test and bootstrap are inferior to both the sign test and Wilcoxon. Our results suggest that the permutation test and the t-test have a similar behavior (and bootstrap is slightly inferior to them). This outcome is in agreement with the study presented by Smucker et al. (Smucker et al., 2007). But Smucker et al. did not analyse the power of the tests and, furthermore, they evaluated Wilcoxon and the sign test in terms of how well their results match with the results yielded by the permutation test. As argued above, we believe that significance tests should be compared with no a priori assumptions about which test is the best.

Our results also agree with accepted statistical principles. It is known that the power of a statistical test is mainly affected by: i) the effect size (the difference between the null and alternative values), ii) the sample size, iii) the variability in the samples, and iv) the significance level of the test. The power of a test depends on these four factors, and it is not uncommon that simpler tests perform better than complex ones. Furthermore, the relative merits of the tests change under different conditions. For example, our TREC8 experiments show that when the sample size is small (10 queries) and the effect size is high (more than 15%) the sign test has less power. Our results also agree with the findings reported by Conover about the relative loss of power of the permutation test (see Section 5.11, (Conover, 1999)). As a matter of fact, simulation studies conducted by Conover and Iman suggested a preference on the tests (t-test $<<$ permutation $<<$ Wilcoxon) that matches with ours. Our study also agrees with Kempthorne and Doerfler (Kempthorne & Doerfler, 1969), who concluded that the permutation test behaves very well under $H0$. The permutation test matches very well with the significance value (see Fig. 2). However, only Wilcoxon behaves well under both $H0$ (Fig. 2) and $H1$ (power curves).

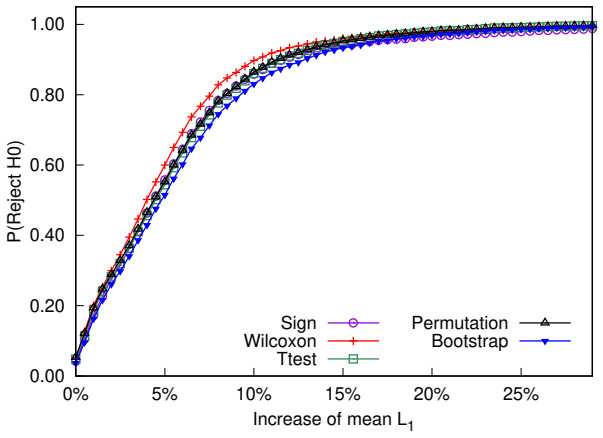
4.3 Discussion

Studying significance tests based on the APs obtained from the simulated runs is a reliable way to understand how the tests detect differences in search performance. An essential component of our methodology is the way in which we produce SD models with increased performance. In section 3, we claimed that increasing $\mu_1^{(i)}$ leads to better performance. Figure 8 plots the AP –averaged over all systems and running 50 simulations per model– with varying increases of $\mu_1^{(i)}$. The curves, which are monotonic increasing, demonstrate that manipulating the mean of L_1 leads to models with increasingly better performance.

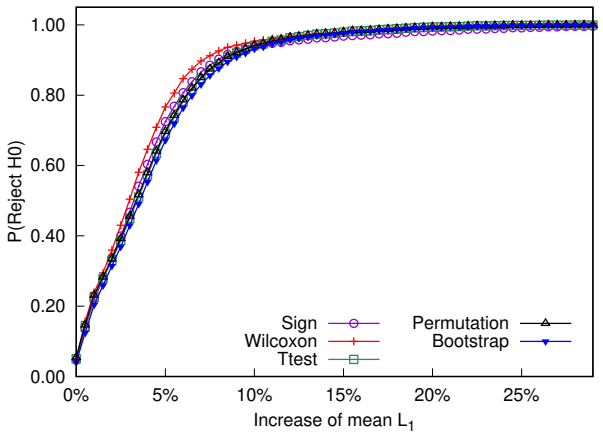
Observe that increasing $\mu_1^{(i)}$ does not improve the performance of the modeled system for every query. The



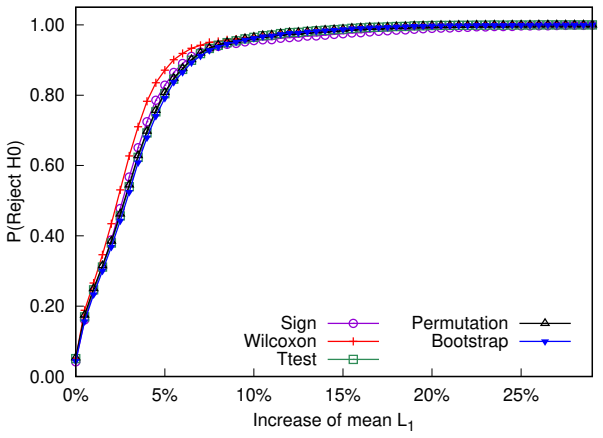
(a) 10 queries



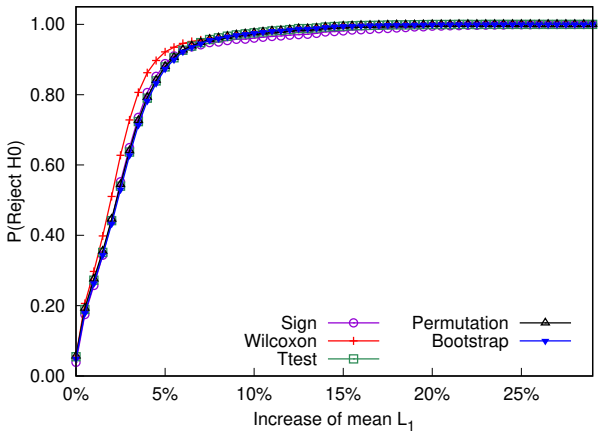
(b) 20 queries



(c) 30 queries

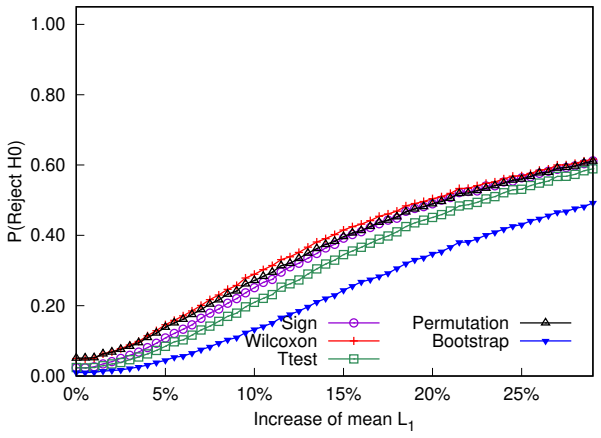


(d) 40 queries

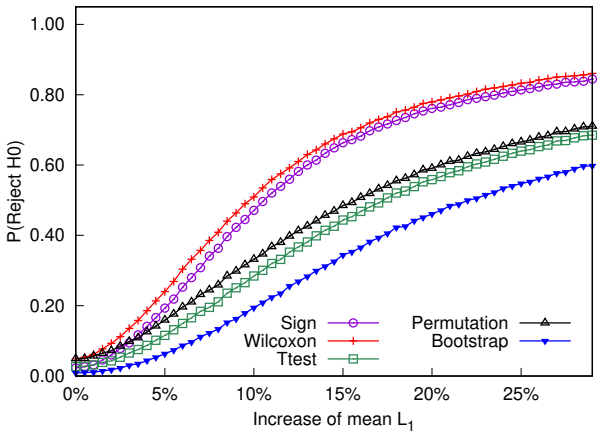


(e) 50 queries

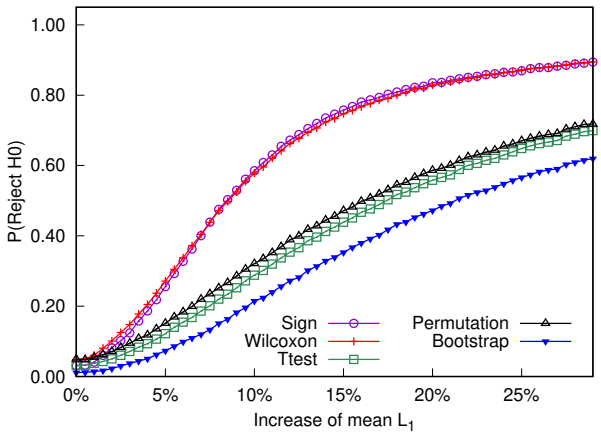
Figure 3: Average $P(Reject\ H_0)$ ($\alpha = 0.05$) in TREC 3.



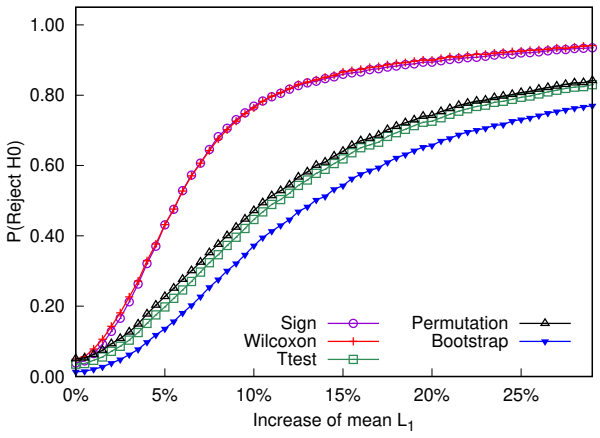
(a) 10 queries



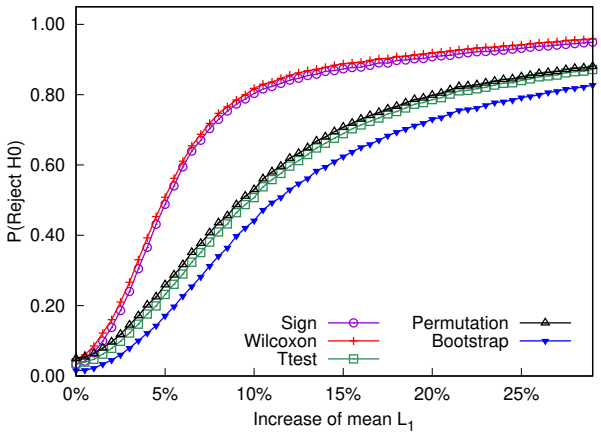
(b) 20 queries



(c) 30 queries

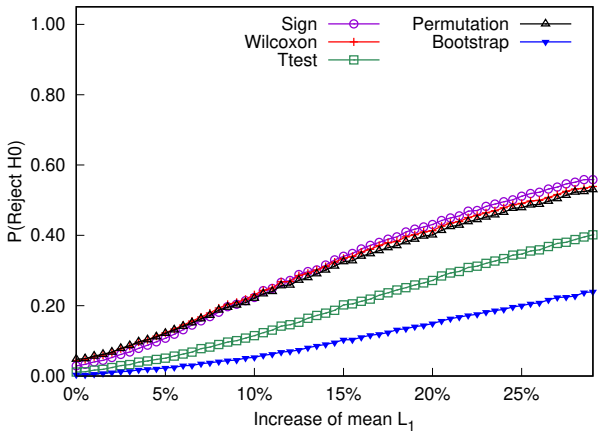


(d) 40 queries

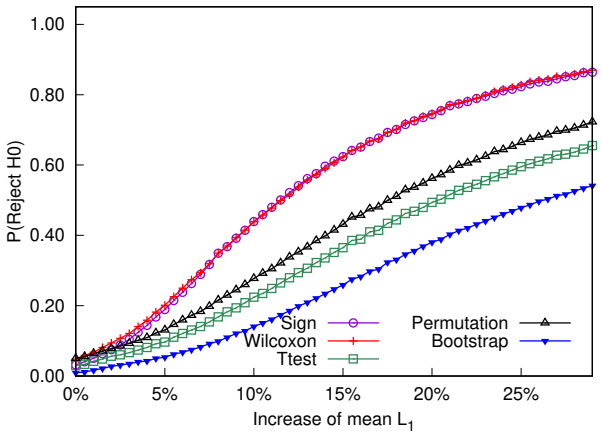


(e) 50 queries

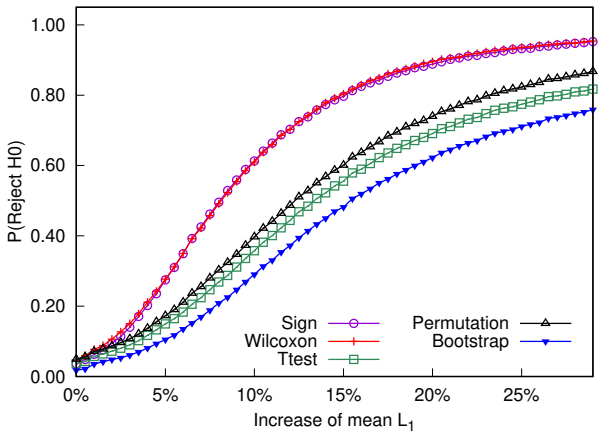
Figure 4: Average $P(Reject\ H_0)$ ($\alpha = 0.05$) in TREC 5.



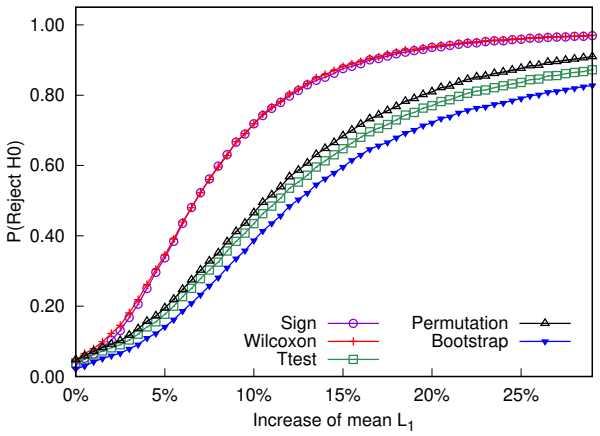
(a) 10 queries



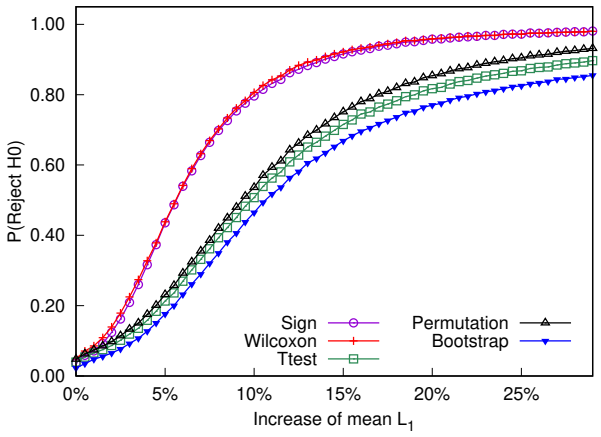
(b) 20 queries



(c) 30 queries

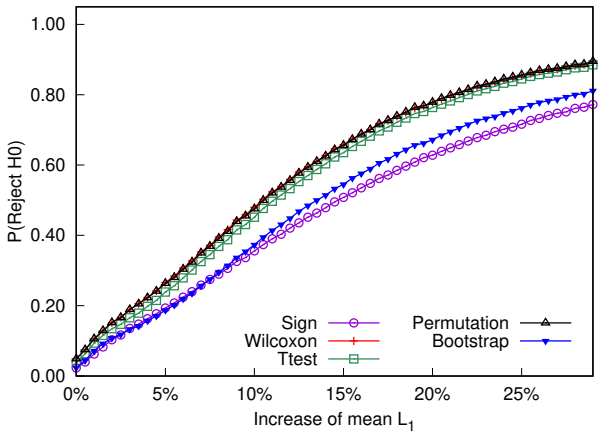


(d) 40 queries

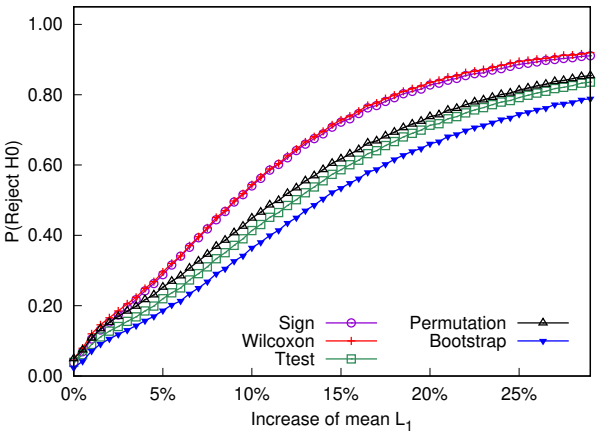


(e) 50 queries

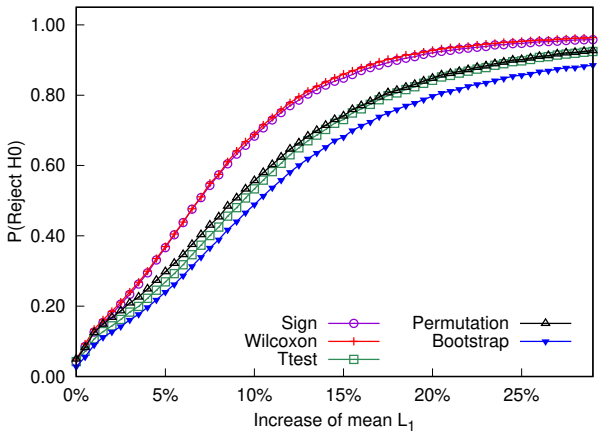
Figure 5: Average $P(Reject\ H_0)$ ($\alpha = 0.05$) in TREC 6.



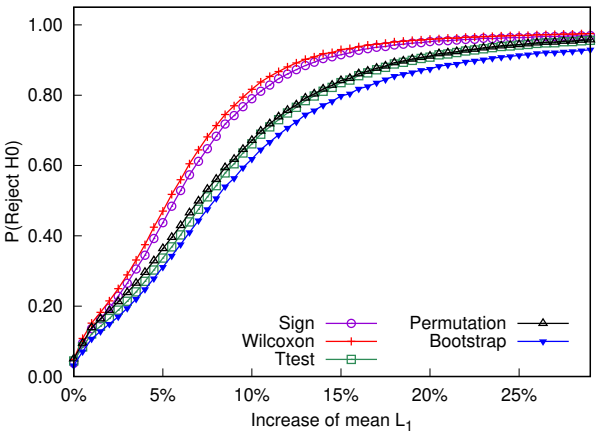
(a) 10 queries



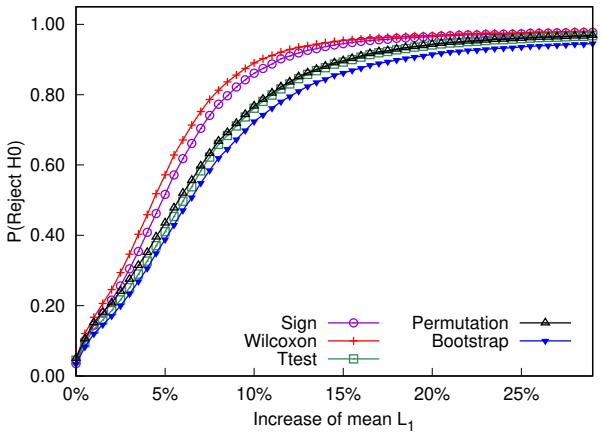
(b) 20 queries



(c) 30 queries

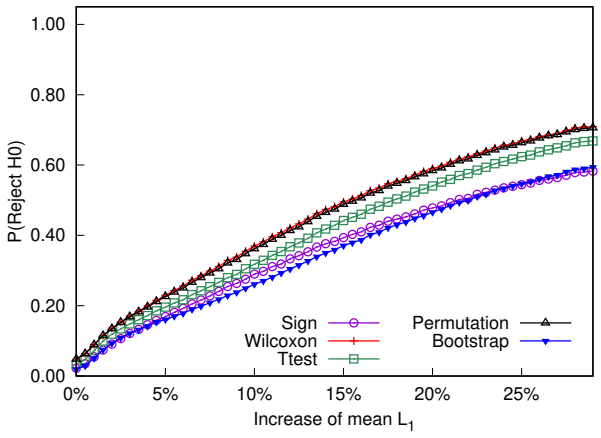


(d) 40 queries

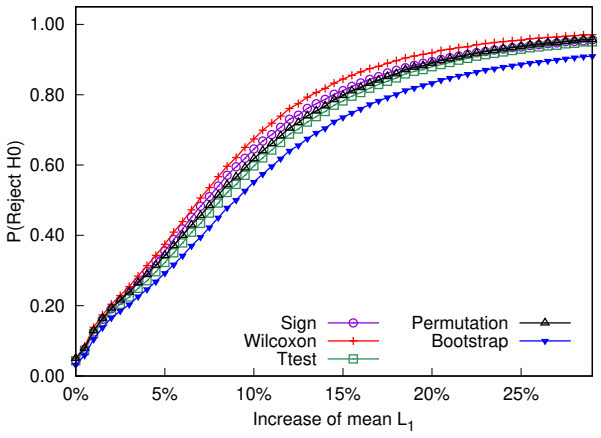


(e) 50 queries

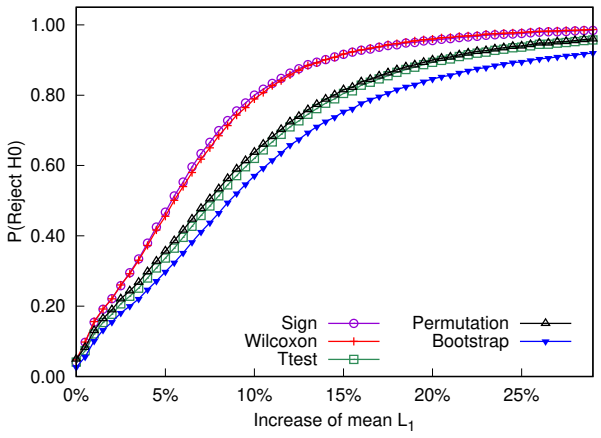
Figure 6: Average $P(Reject\ H_0)$ ($\alpha = 0.05$) in TREC 7.



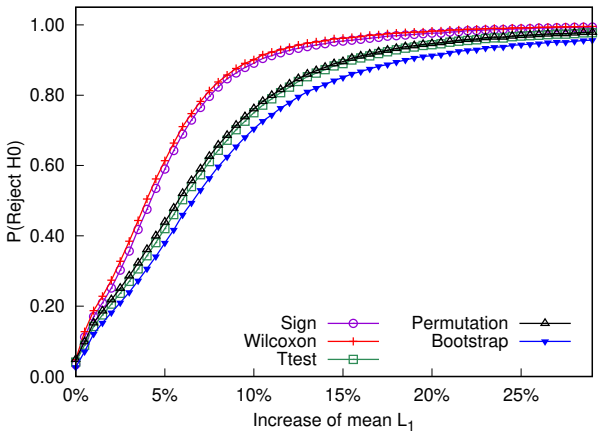
(a) 10 queries



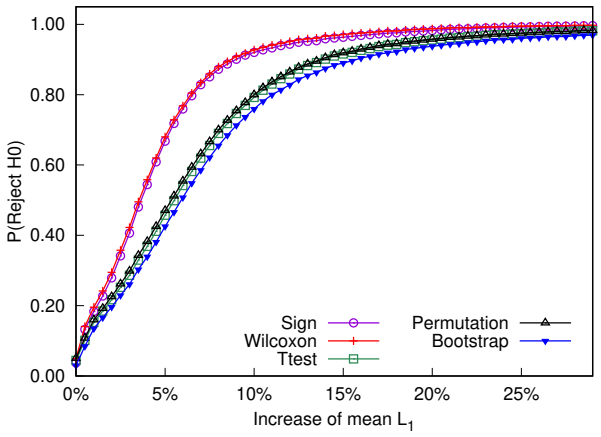
(b) 20 queries



(c) 30 queries



(d) 40 queries



(e) 50 queries

Figure 7: Average $P(Reject\ H_0)$ ($\alpha = 0.05$) in TREC 8.

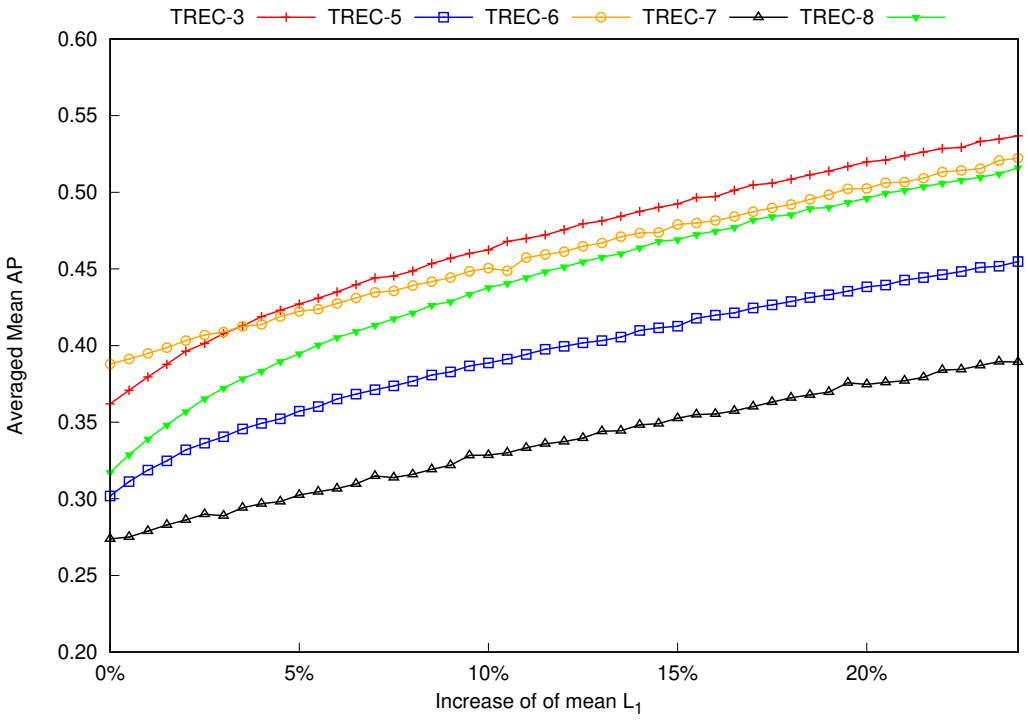


Figure 8: Average Mean Average Precision against increased means of the L_1 distributions.

process is stochastic in nature and the improved systems tend to perform better for many queries, but they also lead to decreased performance for a few queries. This is a natural consequence of the sampling process (sampling from a better model tends to produce better performance, but individual samples show some variance). To further illustrate this point, Figure 9 shows the effect of a 5% increment in $\mu_1^{(i)}$ (similar boxplots were obtained for the other percentages of improvements). The figure presents a summary of the distribution of ΔAP (% variation: AP improved model vs AP original model) for all system-query pairs in our simulated study[2]. In all collections, the median improvement in AP is positive, but there is a large variance, with many queries improved and some other queries damaged. This demonstrates that the simulation is realistic (the system that is intrinsically better underperforms for some individual queries) and reflects the typical situation of topic variation in IR experiments (an improvement in search technology leads to improved performance for the majority of topics, but it also leads to poorer performance for a minority of topics).

5 Conclusions

In this work, we have thoroughly analysed a number of significance tests that have been commonly employed in IR experiments. A crucial contribution of our paper is the proposal of an innovative methodology to compare significance tests. Our method models simulated search systems and evaluates the tests under complete certainty about the truth value of the null hypothesis. Following the lessons learnt in the area of Score Distributions for IR, we built models –learnt from TREC runs— that mimic the behavior of real search systems, we created different situations where the null hypothesis is true or false, and we evaluated the ability of significance tests to correctly accept or reject the null hypothesis.

The experiments performed revealed that Wilcoxon and the sign test are the most reliable tests for IR evaluation. Both of them have more power than the permutation test, bootstrap and the t-test. Furthermore, the behavior

[2]The boxplot represents 100 simulations for each query-system pair.

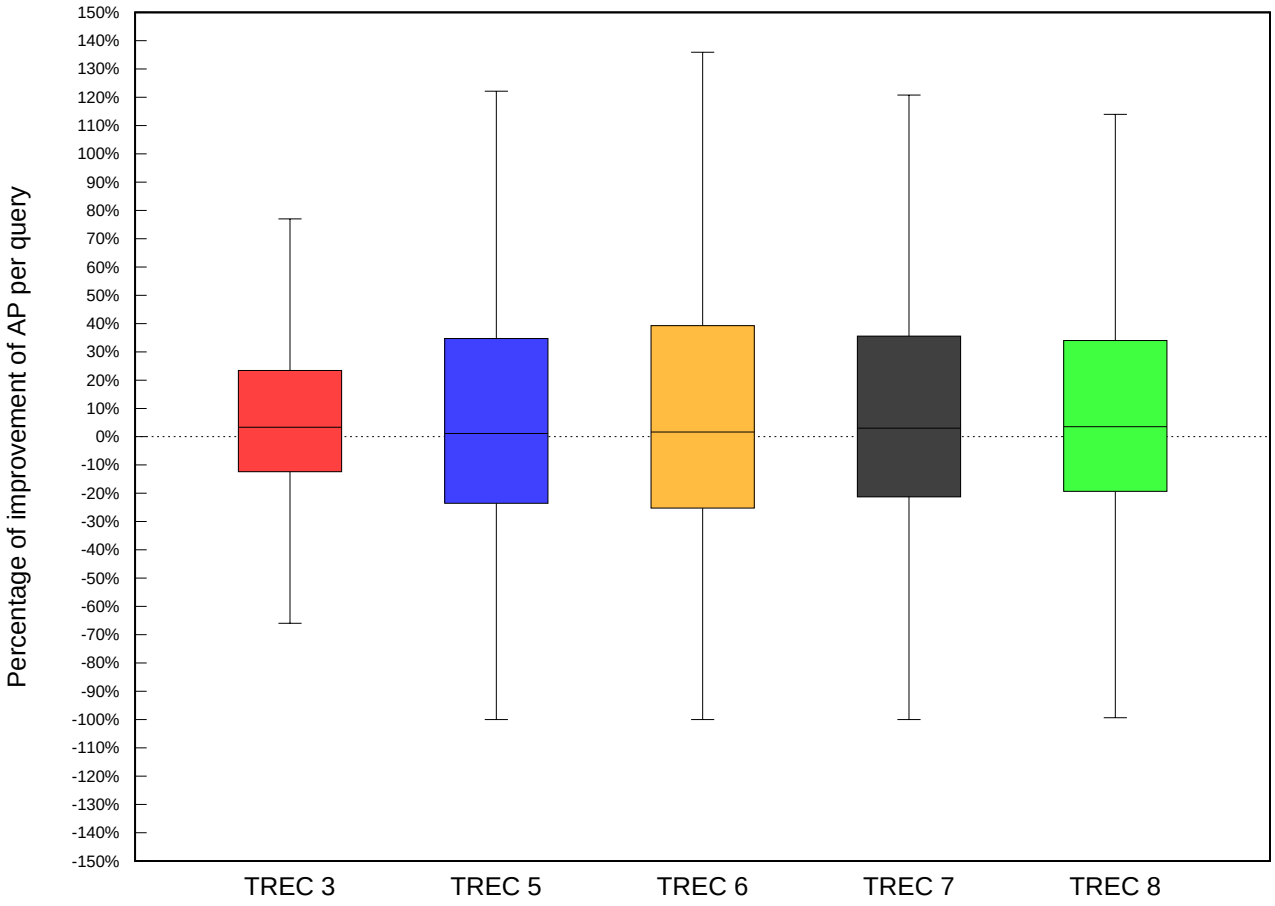


Figure 9: Effect of a 5% increase in the mean of L_1. For each collection, the boxplot summarises the distribution of improvements across queries.

of Wilcoxon and the sign test concerning type I errors is solid. Previous studies (Smucker et al., 2007) claimed that the use of Wilcoxon and sign test should be discontinued. Our study reveals otherwise. We confirmed that the t-test and bootstrap resemble more the permutation test, but we also provided substantive evidence on the weak power of the permutation test when compared with Wilcoxon or the sign test. Smucker et al (Smucker et al., 2007) showed that, in comparison with the permutation test, Wilcoxon and the sign test produce different p-values. Such outcome led the authors to argue against the two non-parametric tests (in a study where miss and false alarm rates were computed based on the agreement between each test and the permutation test). Our methodology is agnostic about which test is more reliable, adds another valuable tool to IR practitioners, and shows that the current support to the t-test or permutation should be revisited. Following our study, we would recommend IR practitioners to employ Wilcoxon or the sign test. Compared to the three other tests, Wilcoxon and the sign tests have more power and, thus, a higher ability to detect real improvements in search technologies. This ability is instrumental in advancing the field of IR.

Our results are also in agreement with well-known statistical facts. We showed that the larger the query set is, the more reliable the comparison is. Experiments with less than 30 queries are questionable and, therefore, it is misleading to compare significance tests based on query-splitting methods whose splits have less than 30 queries. On the other hand, Conover (Conover, 1999) in his authoritative book on non-parametric statistics, exposed a number of cases where parametric methods, such as the t-test, have low power compared to non-parametric methods based on ranks. Such behavior is particularly apparent with data that show specific deviations from the normal distribution. Our evaluation fits well with such statistical knowledge (as a matter of fact, the distribution of differences in ad-hoc retrieval is known not to follow a normal distribution). In his book, Conover also referred to simulation studies where the permutation test showed a relative lack of power when compared to tests based on ranks. Overall, our empirical findings are in line with Conover's findings. In a thorough series of experiments performed with multiple retrieval collections, we showed that the t-test and the permutation test –whose p-values are similar to those of the t-test– have less power than the sign test and Wilcoxon. This suggests that, under the typical conditions of IR evaluation, we should prefer rank-based methods over alternative significance tests. Finally, the weak results obtained with the bootstrap test were a bit surprising. Such poor behavior suggests the need for more effective bootstrap methods in IR experimentation.

Acknowledgements

This work has received financial support from the i) "Ministerio de Economía y Competitividad" of the Government of Spain and FEDER Funds under the research project TIN2015-64282-R, ii) Xunta de Galicia (project GPC 2016/035), and iii) Xunta de Galicia – "Consellería de Cultura, Educación e Ordenación Universitaria" and the European Regional Development Fund (ERDF) through the following 2016-2019 accreditations: ED431G/01 ("Centro singular de investigación de Galicia") and ED431G/08.

We also thank the anonymous reviewers for their really useful suggestions and comments.

References

Arampatzis, A., Beney, J., Koster, C. H. A. & van der Weide, T. P. (2000). Incrementality, Half-life, and Threshold Optimization for Adaptive Document Filtering. In *Proceedings of the 9th text retrieval conference, TREC 2000.* National Institute for Science and Technology (NIST).

Arampatzis, A., Kamps, J. & Robertson, S. (2009). Where to stop reading a ranked list? In *Proc. SIGIR '09* (pp. 524–531). New York, USA: ACM Press.

Arampatzis, A. & Robertson, S. (2011, February). Modeling score distributions in information retrieval. *Inf. Retr.*, *14*(1), 26–46.

Arampatzis, A., Robertson, S. & Kamps, J. (2009). Score Distributions in Information Retrieval. In *Proceedings ICTIR 2009* (pp. 139–151).

Arampatzis, A., Zagoris, K. & Chatzichristofis, S. A. (2013, January). Dynamic two-stage image retrieval from large multimedia databases. *Information Processing & Management*, *49*(1), 274-285.

Baumgarten, C. (1999, August). A probabilistic solution to the selection and fusion problem in distributed information retrieval. In *Proceedings SIGIR '99* (pp. 246–253). New York, USA: ACM Press.

Bookstein, A. (1977, January). When the most "pertinent" document should not be retrieved An analysis of the Swets model. *Information Processing & Management*, *13*(6), 377–383.

Conover, W. (1999). *Practical nonparametric statistics* (3rd ed.). New York: Wiley.

Cormack, G. V. & Lynam, T. R. (2006). Statistical precision of information retrieval evaluation. In *Proceedings SIGIR 2006* (pp. 533–540). New York, NY, USA: ACM.

Cormack, G. V. & Lynam, T. R. (2007, July). Validity and power of t-test for comparing MAP and GMAP. In *Proceedings SIGIR '07* (p. 753-754). New York, USA: ACM Press.

Cummins, R. (2011). Measuring the ability of score distributions to model relevance. In *Proceedings of the 7th asia conference on information retrieval technology* (pp. 25–36). Berlin, Heidelberg: Springer-Verlag.

Cummins, R. (2014). Document score distribution models for query performance inference and prediction. *ACM Trans. Inf. Syst.*, *32*(1), 1–28.

Cummins, R. & O'Riordan, C. (2012). On theoretically valid score distributions in information retrieval. In *Proceedings of ECIR 2012* (pp. 451–454). Berlin, Heidelberg: Springer-Verlag.

Dai, K., Kanoulas, E., Pavlu, V. & Aslam, J. A. (2011). Variational Bayes for modeling score distributions. *Information retrieval*, *14*(1), 47–67.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, *39*(1), 1–38.

Efron, B. & Tibshirani, R. (1993). *An Introduction to the Bootstrap* [Miscellaneous]. Macmillan Publishers Limited.

Hull, D. (1993). Using statistical testing in the evaluation of retrieval experiments. In *Proceedings SIGIR 1993* (pp. 329–338). New York, NY, USA: ACM.

Kanoulas, E., Dai, K., Pavlu, V. & Aslam, J. A. (2010). Score distribution models: Assumptions, intuition, and robustness to score manipulation. In *Proceedings of SIGIR 2010* (pp. 242–249). New York, NY, USA: ACM.

Kanoulas, E., Pavlu, V. & Dai, K. (2009). Modeling the score distributions of relevant and non-relevant documents. In *Proceedings ICTIR 2009* (pp. 152–163).

Kempthorne, O. & Doerfler, T. (1969, February). The behaviour of some significance tests under experimental randomization. *Biometrika*, *56*(2), 231–248.

Losada, D., Parapar, J. & Barreiro, A. (2018). A rank fusion approach based on score distributions for prioritizing relevance assessments in information retrieval evaluation. *Information Fusion*(39), 56–71.

Manmatha, R., Rath, T. & Feng, F. (2001). Modeling score distributions for combining the outputs of search engines. In *Proceedings SIGIR 2001* (pp. 267–275). New York, USA: ACM Press.

Miettunen, J. & Nieminen, P. (2003). The effect of statistical methods and study reporting characteristics on the number of citations: A study of four general psychiatric journals. *Scientometrics*, *57*(3), 377–388.

Nelder, J. A. & Mead, R. (1965, January). A Simplex Method for Function Minimization. *The Computer Journal*, *7*(4), 308–313.

Parapar, J., Presedo-Quindimil, M. A. & Barreiro, A. (2014). Score distributions for pseudo relevance feedback. *Information Sciences*, *273*, 171 - 181.

Robertson, S. (2007). On score distributions and relevance. In *Proceedings ECIR 2007* (pp. 40–51). Springer Berlin Heidelberg.

Robertson, S., Kanoulas, E. & Yilmaz, E. (2013). Modelling score distributions without actual scores. In *Proceedings ICTIR 2013* (pp. 20:85–20:92). New York, NY, USA: ACM.

Sakai, T. (2016). Two sample t-tests for IR evaluation: Student or welch? In *Proceedings SIGIR 2016* (pp. 1045–1048). New York, USA: ACM.

Sanderson, M. & Zobel, J. (2005). Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proceedings SIGIR 2005* (pp. 162–169). New York, NY, USA: ACM.

Savoy, J. (1997). Statistical inference in retrieval effectiveness evaluation. *Information Processing & Management*, *33*(4), 495 - 512.

Smucker, M. D., Allan, J. & Carterette, B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings CIKM 2007* (pp. 623–632). New York, NY, USA: ACM.

Swets, J. A. (1963). Information Retrieval Systems. *Science*, *141*(3577), 245–250.

Swets, J. A. (1969). Effectiveness of Information Retrieval Methods. *American Documentation*, *20*, 72–89.

Urbano, J. (2016, 1st Jun). Test collection reliability: a study of bias and robustness to statistical assumptions via stochastic simulation. *Information Retrieval Journal*, *19*(3), 313–350.

Urbano, J., Marrero, M. & Martín, D. (2013). A comparison of the optimality of statistical significance tests for information retrieval evaluation. In *Proceedings SIGIR 2013* (pp. 925–928). New York: ACM.

Van Rijsbergen, C. J. (1979). *Information retrieval* (2nd ed.). Newton, MA, USA: Butterworth-Heinemann.

Voorhees, E. M. & Buckley, C. (2002). The Effect of Topic Set Size on Retrieval Experiment Error. In *Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 316–323).

Voorhees, E. M. & Harman, D. K. (2005). *TREC: Experiment and evaluation in information retrieval*. The MIT Press.

Webber, W., Moffat, A. & Zobel, J. (2008). Statistical power in retrieval experimentation. In *Proceedings of the 17th ACM conference on information and knowledge management* (pp. 571–580). New York, NY, USA: ACM.

Zobel, J. (1998). How reliable are the results of large-scale information retrieval experiments? In *Proceedings SIGIR 1998* (pp. 307–314). New York, NY, USA: ACM.

Appendix

The following pseudo-code computes the probability of a type I error, $p(Reject\ H0|H0\ is\ true)$:

Algorithm 1: Pseudo-code for computing the probability of a type I error

Input: A set of TREC systems $\{\mathcal{S}_1,\mathcal{S}_2,\ldots,\mathcal{S}_m\}$, a set of queries $\{q_1,q_2,\ldots,q_n\}$, a set of set of relevance judgements $\{\mathcal{J}_1,\mathcal{J}_2\ldots,,\mathcal{J}_n\}$, for each system $\mathcal{S}_j$ and query q_i a set of scores $S_{j,i} = \{s^1_{j,i}, s^2_{j,i}, \ldots, s^{1000}_{j,i}\}$ and a significance level α

for $j \leftarrow 1$ ***to*** m **do**
  for $i \leftarrow 1$ ***to*** n **do**
    $mixture[j,i] \leftarrow learnLogNormalMixture(S_{j,i}, \mathcal{J}_i)$

for $j \leftarrow 1$ ***to*** m **do**
  $rejectionWilcoxon \leftarrow 0$;
  $rejectionSign \leftarrow 0$;
  $rejectionTtest \leftarrow 0$;
  $rejectionPermutation \leftarrow 0$;
  $rejectionBootstrap \leftarrow 0$;
  for $k \leftarrow 1$ ***to*** 1000 **do**
    for $i \leftarrow 1$ ***to*** n **do**
      $S^1_{j,i} \leftarrow sort(randomSample(mixture[j,i], 1000))$;
      $S^2_{j,i} \leftarrow sort(randomSample(mixture[j,i], 1000))$;
      $ap^1[j,i] \leftarrow ap(S^1_{j,i}, mixture[j,i])$;
      $ap^2[j,i] \leftarrow ap(S^2_{j,i}, mixture[j,i])$;
    $rejectionWilcoxon \leftarrow rejectionWilcoxon + testWilcoxon(ap^1[j], ap^2[j], \alpha)$;
    $rejectionSign \leftarrow rejectionSign + testSign(ap^1[j], ap^2[j], \alpha)$;
    $rejectionTtest \leftarrow rejectionTtest + testTtest(ap^1[j], ap^2[j], \alpha)$;
    $rejectionPermutation \leftarrow rejectionPermutation + testPermutation(ap^1[j], ap^2[j], \alpha)$;
    $rejectionBootstrap \leftarrow rejectionBootstrap + testBootstrap(ap^1[j], ap^2[j], \alpha)$;
  $pRejectH0GivenH0[wilcoxon] \leftarrow pRejectH0GivenH0[wilcoxon] + (rejectionWilcoxon/1000)$;
  $pRejectH0GivenH0[sign] \leftarrow pRejectH0GivenH0[sign] + (rejectionSign/1000)$;
  $pRejectH0GivenH0[ttest] \leftarrow pRejectH0GivenH0[ttest] + ()rejectionTtest/1000)$;
  $pRejectH0GivenH0[permutation] \leftarrow pRejectH0GivenH0[permutation] + (rejectionPermutation/1000)$;
  $pRejectH0GivenH0[bootstrap] \leftarrow pRejectH0GivenH0[bootstrap] + (rejectionBootstrap/1000)$;
$pRejectH0GivenH0[wilcoxon] \leftarrow pRejectH0GivenH0[wilcoxon]/m$;
$pRejectH0GivenH0[sign] \leftarrow pRejectH0GivenH0[sign]/m$;
$pRejectH0GivenH0[ttest] \leftarrow pRejectH0GivenH0[ttest]/m$;
$pRejectH0GivenH0[permutation] \leftarrow pRejectH0GivenH0[permutation]/m$;
$pRejectH0GivenH0[bootstrap] \leftarrow pRejectH0GivenH0[bootstrap]/m$;

where m is the number of systems, n is the number of queries, the $sort()$ method sorts in descending order, $randomSample(mixture[j,i], x)$ generates an array of size x with random samples from the mixture, and $testNameOfTheTest(ap^1[j], ap^2[j], \alpha)$ returns 1 when the test determines rejection of $H0$ given the provided $alpha$ and 0 otherwise.

The following pseudo-code produces the power figures:

Algorithm 2: Pseudo-code for computing the power plots of the statistical tests for a TREC edition

Input: A set of TREC systems $\{\mathcal{S}_1,\mathcal{S}_2, \ldots, \mathcal{S}_m\}$, a set of queries $\{q_1,q_2, \ldots, q_n\}$, a set of set of relevance judgements $\{\mathcal{J}_1,\mathcal{J}_2 \ldots, , \mathcal{J}_n\}$, for each system $\mathcal{S}_j$ and query q_i a set of scores $S_{j,i} = \{s^1_{j,i}, s^2_{j,i}, \ldots, s^{1000}_{j,i}\}$ and a significance level α

for $j \leftarrow 1$ ***to*** m **do**
  for $i \leftarrow 1$ ***to*** n **do**
    $mixture[j,i] \leftarrow learnLogNormalMixture(S_{j,i}, \mathcal{J}_i)$

for $j \leftarrow 1$ ***to*** m **do**
  $rejectionWilcoxon \leftarrow 0$;
  $rejectionSign \leftarrow 0$;
  $rejectionTtest \leftarrow 0$;
  $rejectionPermutation \leftarrow 0$;
  $rejectionBootstrap \leftarrow 0$;
  for $h \leftarrow 0.0$ ***to*** 0.30 **do**
    for $k \leftarrow 1$ ***to*** 1000 **do**
      for $i \leftarrow 1$ ***to*** n **do**
        $S^1_{j,i} \leftarrow sort(randomSample(mixture[j,i], 1000))$;
        $mixture'[j,i] \leftarrow mixture[j,i]$;
        $mixture'[j,i].\mu_1 \leftarrow mixture[j,i].\mu_1 \times (1+h)$;
        $S^2_{j,i} \leftarrow sort(randomSample(mixture'[j,i], 1000))$;
        $ap^1[j,i] \leftarrow ap(S^1_{j,i}, mixture[j,i])$;
        $ap^2[j,i] \leftarrow ap(S^2_{j,i}, mixture'[j,i])$;
      $rejectionWilcoxon \leftarrow rejectionWilcoxon + testWilcoxon(ap^1[j], ap^2[j], \alpha)$;
      $rejectionSign \leftarrow rejectionSign + testSign(ap^1[j], ap^2[j], \alpha)$;
      $rejectionTtest \leftarrow rejectionTtest + testTtest(ap^1[j], ap^2[j], \alpha)$;
      $rejectionPermutation \leftarrow rejectionPermutation + testPermutation(ap^1[j], ap^2[j], \alpha)$;
      $rejectionBootstrap \leftarrow rejectionBootstrap + testBootstrap(ap^1[j], ap^2[j], \alpha)$;
    $pRejectH0[wilcoxon, h] \leftarrow pRejectH0[wilcoxon, h] + (rejectionWilcoxon/1000)$;
    $pRejectH0[sign, h] \leftarrow pRejectH0[sign, h] + (rejectionSign/1000)$;
    $pRejectH0[ttest, h] \leftarrow pRejectH0[ttest, h] + ()rejectionTtest/1000)$;
    $pRejectH0[permutation, h] \leftarrow pRejectH0[permutation, h] + (rejectionPermutation/1000)$;
    $pRejectH0[bootstrap, h] \leftarrow pRejectH0[bootstrap, h] + (rejectionBootstrap/1000)$;

for $h \leftarrow 0.0$ ***to*** 0.15 **do**
  $pRejectH0[wilcoxon, h] \leftarrow pRejectH0[wilcoxon, h]/m$;
  $pRejectH0[sign, h] \leftarrow pRejectH0[sign, h]/m$;
  $pRejectH0[ttest, h] \leftarrow pRejectH0[ttest, h]/m$;
  $pRejectH0[permutation, h] \leftarrow pRejectH0[permutation, h]/m$;
  $pRejectH0[bootstrap, h] \leftarrow pRejectH0[bootstrap, h]/m$;

where with $mixture'[j,i].\mu_1 \leftarrow mixture[j,i].\mu_1 \times (1+h)$ we are altering the mean of the distribution of relevant documents in the mixture by a h percentage.