Bioinformatics Written Qualifying Exam 2022: Probabilistic RNA-seq outlier detection

Sandy Kim, 404830610

Abstract

High-throughput sequencing technologies have paved the way for a revolutionary shift in data acquisition which has changed the way we understand and continue to study functional genomics. While it has allowed scientists to study the transcriptome at higher power and detail than ever before, the complex nature of the assay of the data generated can lead to sample deviations in gene expression captured across samples due to both technical and biological variability. The high-dimensionality of the data produced by RNA-seq experiments have made it increasingly difficult to detect these outliers, which can severely skew downstream data analysis [1]. Many scientists often rely on qualitative heuristics to detect outliers such as inspecting principal component analysis biplots and clusterings of the data and visually identifying samples that deviate from the norm [2]. While there have been efforts to develop quantitative methods to automate this process such as utilizing robust principal component analysis methods and iterative procedures, there are few that rely on probabilistic methods [3, 4]. Here, I introduce an approach to detect outliers that leverages Bayesian statistical modeling and inference.

1 Introduction

High-throughput RNA sequencing, also known as RNA-seq, has made its way since its discovery to become a ubiquitous tool to study genomics. This powerful method allows scientists to interrogate the transcriptome at any given moment [5]. In particular, allowing them to see transcript presence and abundance in any biological sample usually to identify differentially expressed genes. However, the experimental protocol of RNA-seq is elaborate, with many steps along the way, such as fragmentation and reverse transcription. Each step allows for the opportunity for technical variation in the resulting data, whether it is, say, from inconsistent reagents or errors in reverse transcription. This unwanted variation is especially harmful when it is at a large enough magnitude to lead to inaccuracies in downstream analysis. These extreme deviations from the data are known as outliers.

In statistics, an outlier is a data point that differs significantly from the rest of the observed data. While it is simple to say that this point falls outside of the distribution of the observations, it is generally quite difficult to determine the difference that allows a data point to be considered an outlier. Given the high-dimensionality and complexity of RNA-seq data, this distinction only becomes more difficult.

Though with this, the ubiquity of high-throughput sequencing, and the evident fact that outliers are detrimental to data quality and downstream analysis, there is a lack of research in

outlier detection in specifically RNA-seq [1]. Only until about five years ago, methods explored detecting outliers [6, 7]. But still to this moment, the gold-standard way to identify outliers in the data is to visually inspect dimensionality-reduced RNA-seq data, for instance, a principal component bi-plot followed by clustering [2]. This method lacks statistical justification and along with the visual component, it is hard to curb biases.

Yet, there are very few methods that rely on probabilistic models [3, 4]. On top of that, very few methods address count-based outliers; in other words, outliers that exist prior to the normalization step. This is important as many normalization techniques such as estimating size factors is actually looking to minimize the between sample variability [8]. This variability includes important information that can distinguish outliers versus the rest of the observed data.

In my written qualifying exam, I explore the space of Bayesian statistics and its potential application to detecting outlier samples in generated as a result of RNA sequencing experiments. For this method, I focus on quality control prior to any normalization, thus models will reflect raw RNA-sequencing read count data and outliers will be within the sample space rather than gene space. I formulate two statistical models: one generative, and one inference, to investigate the ability to leverage probabilistic models to detect such outliers. The generative model used particularly to serve as a ground truth and the inference model is the vehicle used to predict which samples are outliers. I investigate the performance in its classification across various proportion of outliers in a constant gene and sample size. I also compare metrics calculated for my model against similar metrics for different methods that have the same overarching goal.

2 Methodology

2.1 Generative model

In order to generate gene read counts for a given sample, I used experimental data to learn parameters from. In particular, the RNA-seq data set used was retrieved from the Genotype-Tissue Expression (GTEx) Project open-access data. The data set includes RNA-seq read counts from the GTEx Analysis V7 of 56202 genes across 17382 samples, taken from various human tissue [9].

Counts were filtered by the protein-coding genes in humans, using a list of Ensembl identification codes, which was acquired through Ensembl BioMaRt, which brought the number of genes down to 18251. The final data set was of dimension 18251×17382.

The generative model is formulated as follows. Let g be a gene, where $g \in [1, G]$. For each gene g in the GTEx dataset, I take the mean of the read counts across all samples. These values are then held in a g-dimensional vector μ.

Let n be the number of samples, where $n \in [1, N]$. Let $p_{outlier}$ be the proportion of outlier in the sample space. For the first $Np_{outlier}$ samples, I draw from a categorial distribution, an $Np_{outlier}$-dimensional vector that holds the outlier effect sizes $I \sim Cat(K, p)$, where $K = \langle -1, 1 \rangle$, indicating low and high, respectively, and $p = \langle 0.5, 0.5 \rangle$. To transform the outlier effect sizes from log space, I raise I as e^I. Let Y, a $G \times N$-dimensional matrix of the read counts. Read counts are sampled as $Y_{n,g} \sim NB(\mu_g * e^I, \phi)$, where μ is the mean read counts from the GTEx dataset as stated above, and ϕ is a dispersion parameter set to be 100. A graphical representation of the generative model is shown in Figure 1.

2.2 Inference model

In order to perform inference, I propose a relatively simple hierarchical model.

Let ψ be an n-dimensional vector outlier indicator, sampled as $\psi_n \sim Bern(\pi)$, where $\pi \sim Beta(1, 10)$. Let β be the log outlier effect size, which is sampled as $\beta_n = \psi_n \times Norm(\rho_n, \tau)$ where $\rho_n \sim Norm(0, \sigma)$ and $\sigma \sim Gamma(1, 0.1)$, and $\tau \sim Gamma(1, 1)$. To transform the outlier effect sizes from log space, I raise β as 2^β. To generate final read counts, I sample a few hyperparameters. Let ϕ be a G-dimensional vector holding dispersion values on the gene-level, sampled as $\phi_g \sim Gamma(\Phi, 1)$, where $\Phi \sim Gamma(1, 0.1)$. Let μ be a constant G-dimensional vector taken to be the mean counts of each gene across samples of the data to be fit. To generate the resulting read counts, let Y be a $G \times N$-dimensional matrix of the read counts. Read counts are sampled as $Y_{g,n} \sim NB(\mu * 2^{\beta_n}, \phi_g)$. The model was implemented in R, using BUGS code, and the model was fit by Markov chain Monte Carlo (MCMC) using the nimble package. Initial values for parameters were not provided and instead sampled by the sampler. A graphical representation of the inference model is shown in Figure 2.

3 Results

3.1 Data preprocessing

I simulated RNA-seq read counts across 100 samples. Since MCMC is computationally intensive and this expense grows exponentially with higher dimensions, I only looked at the first 100 genes of the original 18251. I made simulations with varying proportions of outliers from 0.1 to 1, with increments of 0.1, to see it can affect the model performance. A principal component biplot of the samples from the simulated data is shown in Figure 3. From the PCA biplots, it is evident that there is a clear separation between the low (leftward of PC 1) and high (rightward of PC1) outliers and the rest of the observed data, across all proportion of outliers. To demonstrate that simulated data is relatively realistic, a PCA biplot of the first 100 genes and 100 samples from the GTEx dataset is shown in Figure 4.

The inference model was then fit using the simulated data, with μ being the mean of the counts for the first 100 genes across all samples. In the interest of time, I ran two MCMC chains of 2000 iterations and 1000 iteration burn-in each, for all proportion of outliers of interest.

3.2 Inference model can capture low outliers

Looking to see whether the model correctly predicts outliers in the sample, I watch particularly the transformed outlier effect size parameter 2^β, shown in Figure 5. Interestingly enough, the inference model, for the most part, is able to capture the low outliers (samples that have a significant reduced read count across genes versus the rest of the observed data), but not the high outliers. Though, the magnitude of these effects inferred are of much smaller magnitude than those that are simulated. Additionally, the inference model predicts that the effect size of the low outliers are more reflective of the high outliers (such samples are fitted to have a higher read count across genes versus the rest of the observed data). We also observe that no false positives (all non-outliers are not mistakenly detected to be outliers), and all have a mean transformed effect size of 1 ($\beta = 0$).

3.3 Performance shows low sensitivity and decreasing accuracy across proportion of outliers

To break down exactly what the model is calling correctly and incorrectly, we can plot the decisions on confusion matrices as seen in Figure 6. We can see that the model has a high propensity in calling false negatives, and as previously stated, most, if not all are upon the high outliers. As a result, the inference model has relatively low sensitivity (or power), averaging at about 50% across all proportions. However, the model's specificity actually continues to drop as the proportion of outliers increase. As such, the overall accuracy of the predictions also decreases down to 45% as the proportion of outliers increase, as shown in Figure 7.

3.4 Model performs worse than existing methods

Compared to other existing methods, my inference model performs much worse in detecting outliers. One existing method, PcaGrid, reported in its paper, that the method had "achieved 100% sensitivity and 100% specificity in all the tests using positive control outliers with varying degrees of divergence" [3]. Another method, iLOO (iterative Leave One Out), reported in its paper, that the method had an average accuracy of about 95% across sample sizes from 5 to 20 [4]. However, these metrics were scored on completely different data sets.

4 Discussion

4.1 Broader impact

The results of my written qualifying exam suggest that probablistic models can perhaps be successfully used to detect outliers in data. The observation that the model can pick up on only low outliers comes with a comfort, that often times, outlier samples in RNA-seq are due to overall low read counts or dropouts, and rarely are outliers have value higher than the norm. This is exhibited in Figure 4. Also, even though accuracy decreases as the proportion of outliers increase, it is still quite strong at the more realistic proportions such as 0.1.

4.2 Caveats to the study

There are many caveats and flaws to this study. First, given the computational cost of running inference procedures such as MCMC, a method like this would be virtually impossible to apply to large studies such as GTEx, where 10,000+ genes are investigated, with 10,000+ samples [9]. A method like this would better applied to smaller groups of genes of interest. An example of such setting would be high-throughput CRISPR screens, where a select and often relatively few number of genes are targeted and monitored, but RNA-seq is still used [10]. Additionally, to carry on with the notion of computational expense, there could have been more iterations in each of the MCMC chains that I ran if I had more time, to allow sampling to fully converge.

Next, the simulation created outlier effect sizes that were entirely distinct from the normal count data as shown in the PCA biplots and count data of outlier samples were distinguished across every single gene. Such separation seen in the biplots presented are uncommon, and in

reality, much of the difficulty in identifying outliers in such high dimensions are due to counts in only a few of many genes being substantially different from the rest. Both the generative and inference model also follow the same final distribution for count data, a negative binomial. It is also important to note that such distinctions can contain valuable information such as differentially expressed genes, which can mistakenly be filtered out in procedures like this.

Also, the inference model does not pick up the right magnitude of effects; in fact, the magnitude it picks up is immensely small, in the hundredths. This bleeds into an analysis that is presented with very little statistical rigor as the posterior probability of inclusion is very small overall, and a very high α would have been used. Rather, this study relied on the small differences from having a transformed outlier effect size being 1, as the non-outliers had a definite mean of $\beta = 1$. The direction of the effects detected were also opposite. Therefore, the results presented are not strong, and only lay the groundwork for future work for exploration of this field.

Lastly, in addressing the poor performance compared to existing methods, it is also important to note the fundamental differences of each method. PcaGrid relies on a grid search algorithm and iLOO relies on iteratively leaving samples out, as suggested by its name [3, 4]. Yet, it still remains objectively true that my inference model performs poorly. This can be expanded upon; if given more time, one can learn how to use and apply these methods on my own simulations to level the playing field.

4.3 Future directions

That being said, there is a lot of room for future studies to expand on the use of probabilistic modeling for outlier detection. One would be to improve the model such that correct magnitudes can be detected; in this model in particular, ψ was never really strongly associated to 1. If used in the right cases and the model had behaved well, downstream analysis can be presented even more strongly, since ψ can be used as a posterior inclusion probability. As a result, p-values can be reported, which would be valuable. More MCMC chains, iterations, and simulations can also be ran to increase sample size, allow for full convergence in sampling, and give proper error bars for effect sizes.

References

[1] Gabriela A. Merino, Cristóbal Fresno, Frederico Netto, Emmanuel Dias Netto, Laura Pratto, and Elmer A. Fernández. The impact of quality control in RNA-seq experiments. *Journal of Physics: Conference Series*, 705:012003, April 2016. Publisher: IOP Publishing.

[2] Ana Conesa, Pedro Madrigal, Sonia Tarazona, David Gomez-Cabrero, Alejandra Cervera, Andrew McPherson, Michał Wojciech Szcześniak, Daniel J. Gaffney, Laura L. Elo, Xuegong Zhang, and Ali Mortazavi. A survey of best practices for RNA-seq data analysis. *Genome Biology*, 17(1):13, January 2016.

[3] Xiaoying Chen, Bo Zhang, Ting Wang, Azad Bonni, and Guoyan Zhao. Robust principal component analysis for accurate outlier sample detection in RNA-Seq data. *BMC Bioinformatics*, 21(1):269, June 2020.

[4] Nysia I. George, John F. Bowyer, Nathaniel M. Crabtree, and Ching-Wei Chang. An Iterative Leave-One-Out Approach to Outlier Detection in RNA-Seq Data. *PLOS ONE*, 10(6):e0125224, June 2015. Publisher: Public Library of Science.

[5] Ali Mortazavi, Brian A. Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nature Methods*, 5(7):621–628, July 2008. Number: 7 Publisher: Nature Publishing Group.

[6] Jun Li and Robert Tibshirani. Finding consistent patterns: A nonparametric approach for identifying differential expression in RNA-Seq data. *Statistical Methods in Medical Research*, 22(5):519–536, October 2013. Publisher: SAGE Publications Ltd STM.

[7] Uri Shaham, Kelly P Stanton, Jun Zhao, Huamin Li, Khadir Raddassi, Ruth Montgomery, and Yuval Kluger. Removal of batch effects using distribution-matching residual networks. *Bioinformatics*, 33(16):2539–2546, August 2017.

[8] Mark D. Robinson and Alicia Oshlack. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology*, 11(3):R25, March 2010.

[9] THE GTEX CONSORTIUM. The GTEx Consortium atlas of genetic regulatory effects across human tissues. *Science*, 369(6509):1318–1330, September 2020. Publisher: American Association for the Advancement of Science.

[10] Ophir Shalem, Neville E. Sanjana, and Feng Zhang. High-throughput functional genomics using CRISPR–Cas9. *Nature Reviews Genetics*, 16(5):299–311, May 2015. Number: 5 Publisher: Nature Publishing Group.

Figures

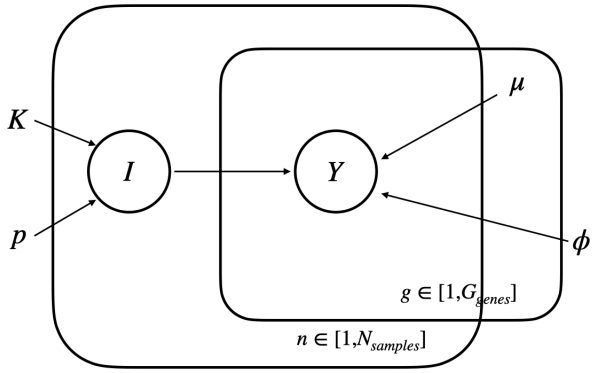

Figure 1: Graphical schematic of the generative model.

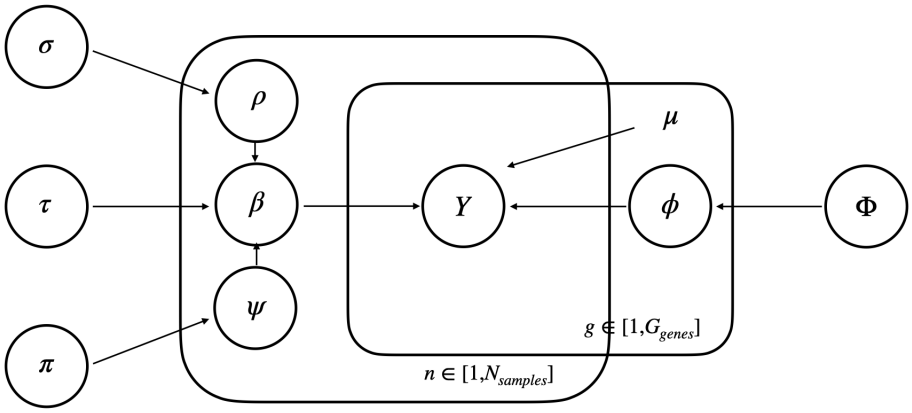

Figure 2: Graphical schematic of the inference model.

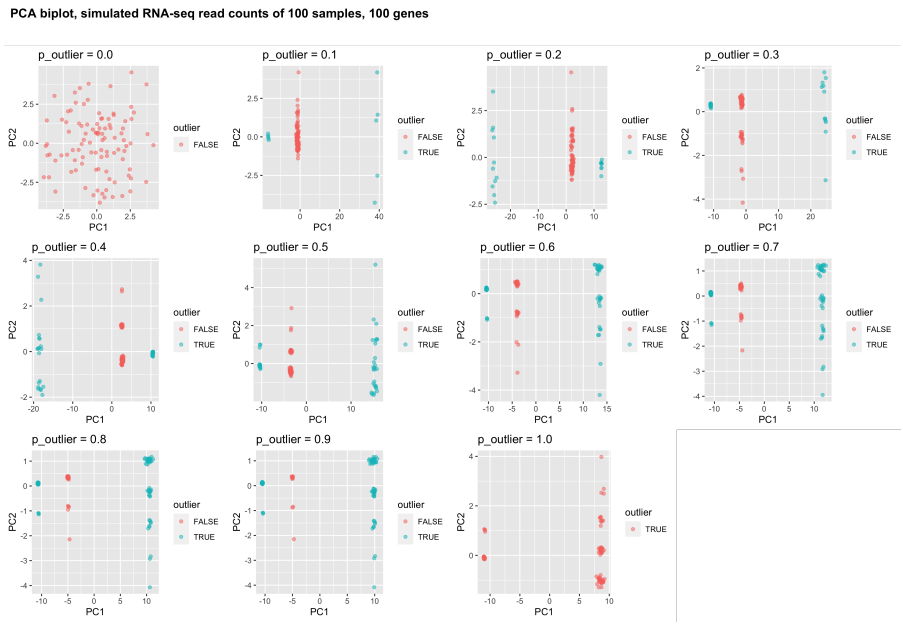

Figure 3: PCA bi-plot (PC 1 and PC 2) of the generated count data of 100 samples, across proportion of outliers from 0 to 1 in increments of 0.1 (left to right, top to bottom). Outliers are colored in blue and the rest of the data in red. Dimensionality of the genes were reduced, and as a result, each point indicates a sample.

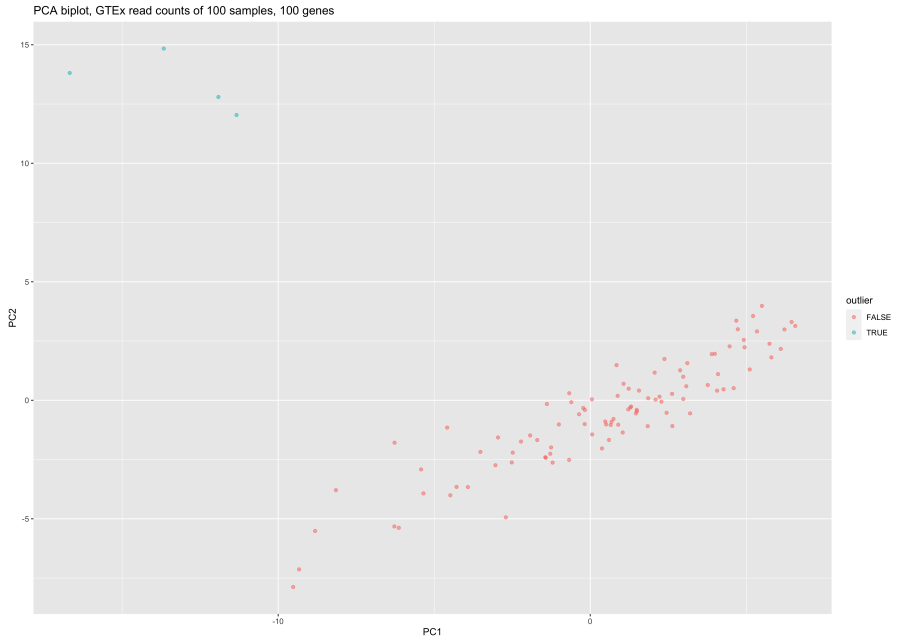

Figure 4: PCA bi-plot (PC 1 and PC 2) of the GTEx subsetted data, with outliers colored in blue and the rest of the data in red. Dimensionality of the genes were reduced, and as a result, each point indicates a sample.

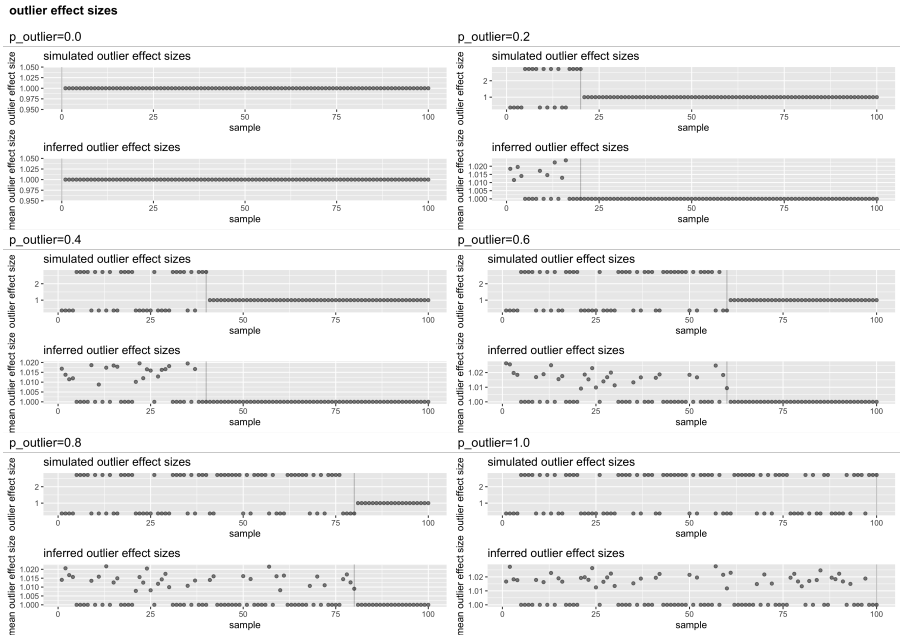

Figure 5: Simulated (top) and inferred (bottom) outlier effect sizes of outlier proportions from 0 to 1 in increments of 0.2 (left to right, top to bottom)

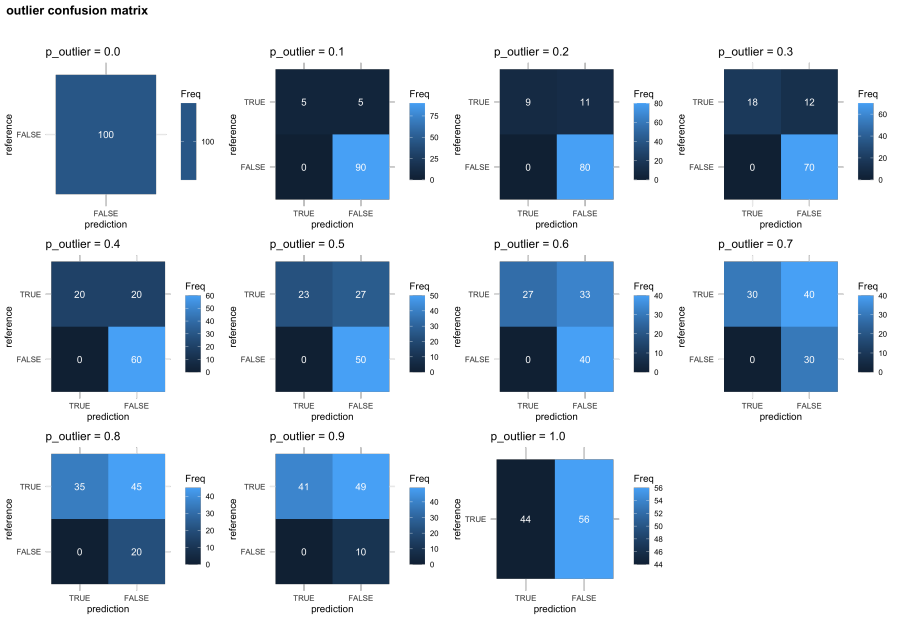

Figure 6: Confusion matrices of outlier detection model predictions, across proportion of outliers from 0 to 1, in increments of 0.1 (left to right, top to bottom).

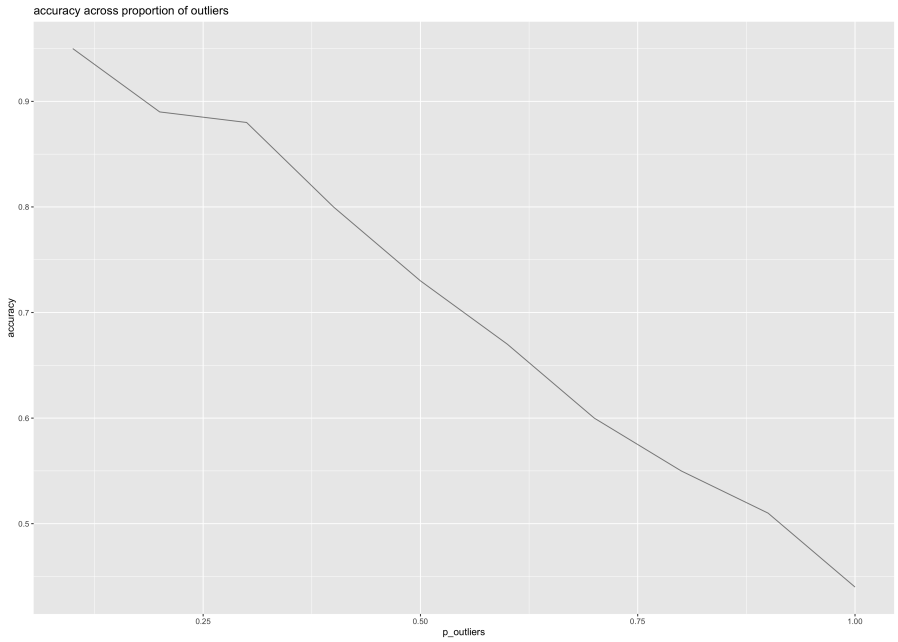

Figure 7: Observed overall accuracy of outlier detection by inference model on simulated data, across proportion of outliers from 0 to 1.