[An Improved Algorithm For Eliminating Confounding Between Gene Expression and Guide RNA]

by

[Hongyu Du]

Department of [Biostatistics and Bioinformatics]
Duke University

Date: ________________________
Approved:

____
[Andrew Scott Allen], Supervisor

____
[William Majoros]

____
[Alejandro Ochoa]

Thesis submitted in partial fulfillment of the requirements for the degree of
Master in the Department of [Biostatistics and Bioinformatics]
in the Graduate School of Duke University
2022

Abstract

[An Improved Algorithm For Eliminating Confounding Between Gene Expression and Guide RNA]

by

[Hongyu Du]

Department of [Biostatistics and Bioinformatics]
Duke University

Date: ________________________
Approved:

____
[Andrew Scott Allen], Supervisor

____
[William Majoros]

____
[Alejandro Ochoa]

An abstract of a thesis submitted in partial fulfillment of the requirements for
the degree of Master in the Department of [Biostatistics and Bioinformatics]
in the Graduate School of Duke University
2022

Abstract

We evaluated two methods being used to test the relationship between the presence of CRISPR guide RNA and gene expression in single-cell CRISPR screens while accounting for confounding. The first method, SCEPTRE[1], uses a conditional randomization test to simulate the null distribution between guide presence and gene expression while accounting for confounding. However, during the conditional randomization used by SCEPTRE[3], the number of cells perturbed by a guide is potentially different in each resampled dataset, leading to increased variability and low power.We propose a new algorithm based on BiasedUrn[2]. We replace the conditional randomization in SCEPTRE with BiasedUrn resampling which maintains the number of perturbed cells in each dataset. We show that the new algorithm has the correct type-I error and higher power than SCEPTRE.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

Symbols

General symbols used in this thesis

Y/Y_i	gene expression
X/X_i	guide RNA presence
τ_0	guide RNA probability/intercept in logistic regression
τ	coefficient of confouding in logistic regression
$\hat{\tau_0}$	estimated guide RNA probability/intercept in logistic regression
$\hat{\tau}$	estimated coefficient of confouding in logistic regression
Z_i	confounding in SCEPTRE
β	guide RNA effect/coefficient of X in SCEPTRE
β_0	mean expression/intercept in SCEPTRE
γ	coefficient of confoudning in regression
α	dispersion parameter in SCEPTRE or intercept in BiasedUrn
π_i	probability in logistic regression
$\hat{\pi}_i$	estimated probability in logistic regression
B	number of conditional randomization test in SCEPTRE
X^b	new guide RNA presence
$\tilde{X}$	new guide RNA presence
$\hat{\beta}$	z-score

$z(X^{-b}, Y, Z)$	new z-score
N	the length of gRNA
N_1	the number of 1 in gRNA
s	the set of all possible X^b
$\hat{F_{null}}$	skew-t distribution
M	number of simulations
X_1/X_2	examples of resampled sequence of gRNA presence
$f(X^b; \mu, N_1)$	multivariate hypergeometric distribution
Y_p	sorted p-values
X_p	expectations for order statistics of uniform distribution
$T(X, Y.Z)$	test statistic

Abbreviations

SCEPTRE	Analysis of single-cell perturbation screens via conditional resampling
BiasedUrn	Biased Urn Bootstrap Method
NegBin	Negative Binomial Regression
Big-P	One group of p-values
gRNA	Guide RNA

Acknowledgements

First and foremost, I'd like to thank Duke University's department of biostatistics and bioinformatics for providing me with a learning platform and allowing me to access such a comprehensive scientific research project.

I am especially grateful to Andrew Scott Allen, my master's degree advisor, for his assistance during my graduate studies. I learned how to think independently about a complete scientific research project, how to avoid detours, and how to solve various theoretical, statistical, and computer science difficulties during my one and a half year acquaintance with him.

I'm exceptionally thankful to two different educators on my board of trustees, William Majoros and Alejandro Ochoa. They gave me extraordinary assistance and support for my intermittent logical examination task. Together, we finished the task proficiently and thoroughly.

At last, I might want to thank the wide range of various educators and schoolmates who helped me during my review at Duke University.

1

Introduction

Single-cell CRISPR screens are becoming increasingly popular and promising biotechnological methods in gene research. It could generate a large and comprehensive set of cellular phenotypes to assess the specific CRISPR-driven gene edits, knockdowns, and altered gene expression profiles. However, analyzing these screens presents numerous significant statistical challenges. Because of these difficulties and challenges, existing analysis methods have calibration issues. The most important is that there will be an incorrect relationship between gene expression and guide RNA or presence when a technical factor or confounding factor exists. SCEPTRE: conditional resampling analysis of single-cell perturbation screens is one solution to the problem. SCEPTRE associates perturbations and expression through resampling while also avoiding calibration issues caused by technical confounders and expression model misspecification. SCEPTRE outperforms CRISPR. However, the SCEPTRE resampling method was unable to ensure that the number of 0 values equals the number of 1 values in the guide RNA presence sequence in each resampling. This means that SCEPTRE has high variability, resulting in low statistical power. In SCEPTRE, we solve this problem by replacing resampling with BiasedUrn. The BiasedUrn bootstrap method is based on the Fisher non-central hypergeometric distribution. BiasedUrn ensures that the number of 0 values in the guide RNA presence sequence equals the number of 1 values in each resampling, reducing variability and increasing power.

This thesis is organized so that chapters 2 and 3 introduce the mathematical theory of SCEPTRE and the new algorithm. Chapter 4 discusses type I error simulations, power calculations, uniform distribution of p-values, and skew-t approximated p-values. This chapter also covers the theory behind these simulations. Chapters 5 and 6 provide a summary of the entire thesis as well as references. The deviation of how to calculate the sample sizes of estimated powers and estimated p-values is included in Chapter 7.

2

The SCEPTRE Theory

The fundamental SCEPTRE theory will be explained in this chapter. Negative binomial regression will be used to represent the relationship between gene expression, the presence of guide RNA, and confounding. The z-score, which is a very important parameter, is the estimated coefficient of guide RNA presence $\hat{\beta}$. The most important aspect of SCEPTRE is the resampling, which means we resampled a large number of new gRNA presence sequences. The entire resampling procedure is based on the logistic regression model between gRNA and confounding. As a result, the depiction of the relationship between gRNA and confounding is central to SCEPTRE. Finally, the p-value is calculated by comparing the new z-scores from each new sequence of gRNA presence to the original z-score $\hat{\beta}$.

This is the fundamental process and concept of SCEPTRE. The mathematical steps for developing SCEPTRE's statistical model are detailed below.

In statistical genetics analysis, each cell contains a unique set of genes, and all genes within a single cell have a negative binomial distribution. Y_i is the gene expression in each cell ($i = 1, 2, \cdots, n$). It can be represented by,

$$Y_i \sim NegBin(\mu_i, \alpha)$$

where α is the dispersion parameter.

Let $X_i \in \{0, 1\}$ indicate whether the guide RNA presence was in the cell and Z_i be a list of cell-level technical factors (confounding). Letting $(X, Y, X) = (X_i, Y_i, Z_i)_{i=1}^{n}$, consider any test statistic $T(X, Y, Z)$ measuring the effect of the gRNA presence on the expression of the gene.

Since we have the background of gene expression Y_i, gRNA sequence X_i, and confounding Z_i, let us talk about how to use negative binomial regression and resampling to do the SCEPTRE.

In the first step, we fit the technical factor effects $\left(\hat{\beta}_0, \hat{\gamma}\right)$ on gene expression from the negative binomial regression without the gRNA term,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right); \log\left(\mu_i\right) = \beta_0 + Z_i^T \gamma$$

In the second step, we fit $\hat{\beta}$ from a negative binomial regression with the estimated contribution of Z_i from step 1 as offsets,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right); \log\left(\mu_i\right) = X_i\beta + \hat{\beta}_0 + Z_i^T \hat{\gamma}$$

The z-score is $\hat{\beta}$, which will be used to calculate the p-values later on.

In the third step, we build the relationship between gRNA presence and confounding through a logistic regression model. Letting $\pi_i = P[X_i = 1 | Z_i]$. Then, assume that,

$$X_i \sim Ber\left(\pi_i\right); \log\left(\frac{\pi_i}{1 - \pi_i}\right) = \tau_0 + Z_i^T \hat{\tau}$$

We fit $(\hat{\tau_0}, \hat{\tau})$ via a logistic regression of X_i and Z_i. Then, extract the fitted probabilities,

$$\hat{\pi_i} = \left(1 + exp\left(-\left(\tau_0 + Z_i^T \tau\right)\right)\right)^{-1}$$

The forth step is the core of SCEPTRE. For $b = 1, \cdots\cdots, B$, resample the gRNA presence assignments based on the probabilities $\hat{\pi}_i$ to obtain X^b. The vector X^b is a new sequence of gRNA presence. We have a B number total of new gRNA presence. Then, extract a new z-score $z\left(X^b, Y, Z\right)$ from the reduced negative binomial regression like in step 1 and step 2 each time.

In the fifth step, we fit s skew-t distribution $\hat{F_{null}}$ to reduce the resampled z-scores set $\left\{z\left(X^b, Y, Z\right)\right\}_{b=1}^{B}$.

The final step returns the p-value,

$$P_{SCEPTRE} = P\left[\hat{F_{null}} \leqslant z\left(X, Y, Z\right)\right]$$

Generally speaking, let $z(X^{(b)}, Y, Z) = T(\tilde{X}, T, Z)$ and $\hat{\beta} = T(X, T, Z)$. Then, the p-value is given by,

$$p = P[T(\tilde{X}, T, Z) \geqslant T(X, T, Z) | X, Y, Z]$$

In summary, it means repeatedly sampling $\tilde{X}$ from the distribution $\tilde{X}_i \sim Ber(\pi_i)$, recomputing the test statistic with X replaced by $\tilde{X}$, and defining the p-value as the probability the resampled test statistic exceeds the original.

3

The New Algorithm Theory

The new algorithm's main point is to replace SCEPTRE's conditional randomization with the BiasedUrn permutation. Each guide RNA presence will have an equal number of 0 and 1 in the resampled data set. Before we get into the specific algorithm, let's take a look at the BiasedUrn.

The BiasedUrn method was inspired by the paper "A Permutation Procedure to Correct for Confounders in Case-Control Studies, Including Tests of Rare Variation." When there is confounding, a permutation procedure is used to investigate the relationship between a case-control study and rare variations. The vector of the case-control study is similar to the presence of guide RNA because both contain only 0 or 1. The vector of rare variation is similar to the vector of gene expression in that both contain a large number of 0. Both of them are also perplexing. They both have very similar data structures. They do, however, both employ the concept of resampling. SCEPTRE generates a large number of guide RNA presences, while BiasedUrn generates a large number of case-control groups.

Now, we will introduce and explain more in detail how the new algorithm works mathematically.

First, let's go over the negative binomial regression section of SCEPTRE.

Each cell contains a unique set of genes, and all genes in a single cell have a negative binomial distribution. Y_i is the gene expression in each cell $i = 1, \cdots, n$. Then,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right)$$

Let $X_i \in \{0,1\}$ indicates whether the gRNA was present in the cell and $Z_i \in R^d$ be a list of cell-level technical factors (confounding). In the first step, we fit $\left(\hat{\beta}_0, \hat{\gamma}\right)$ from the negative binomial regression, except without the gRNA term,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right); \log\left(\mu_i\right) = \beta_0 + Z_i^T \gamma$$

In the second step, we fit $\hat{\beta}$ from a negative binomial regression with the estimated contribution of Z from step 1 as offsets,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right); \log\left(\mu_i\right) = X_i\beta + \beta_0 + Z_i^T \gamma$$

The z-score is also $\hat{\beta}$ this time.

Let us now look at the difference between the new algorithm and SCEPTRE.

In SCEPTRE, the new gRNA sequence X^b follow the distribution $Ber(\pi_i)$. Even if the parameter π_i remains constant across resampling, the number of 1 and 0 in new sequences of guide RNA presence will imbalanced each time. For example,

$$X_1 = (0, 1, 0, 1, 1)$$

$$X_2 = (0, 1, 0, 0, 1)$$

X_1 and X_2 could both follow the distribution $Ber(0.6)$, but the number of 1 and 0 are different between X_1 and X_2, which means that SCEPTRE has a large variability.

In the new algorithm BiasedUrn, we use the multivariate hypergeometric distribution to do the resampling,

$$f(X^b; \mu, N_1) = \frac{g(X^b; \mu)}{\sum_s g(s; \mu)}$$

where $g(X^b;\mu) = \prod_{j=1}^{N} \mu_j^{X_j^b}$, N is the length of gRNA, N_1 is the number of 1 in gRNA, and s indicate the set of all possible X^b configurations consistent with N and N_1. Based on this, if $X_i = (0,0,1,1,1)$, then the possible X_1 and X_2 could be,

$$X_1 = (0,1,0,1,1)$$

$$X_2 = (0,1,1,0,1)$$

All possible X^b must have three 1 and two 0. Because of this, the new algorithm BiasedUrn has less variability than SCEPTRE.

In the next step, we also fit skew-t approximation $\hat{F_{null}}$ to reduce the sample of resampled z-scores $\left\{z\left(X^b, Y, Z\right)\right\}_{b=1}^{B}$.

Finally, we have the p-value,

$$P_{BiasedUrn} = P\left[\hat{F_{null}} \leqslant z\left(X, Y, Z\right)\right]$$

4

simulation Outcomes

This chapter will go over the simulation results, such as type-I error, power calculation, uniform distribution of p-values, and skew-t approximated p-values. We will also go over type-I error simulation and mathematical theory, as well as power calculation and p-value calculation.

4.1 Simulation Theory

In this section, we will primarily discuss how to generate gene expression data, and guide RNA presence based on confounding factors.

The simulation theory is distinct from the general theory. This is because in real simulation, we only have data of confounding Z_i at first. Therefore, we need to use $X_i|Z_i$ and $Y_i|X_i, Z_i$ to create gRNA presence X_i and gene expression Y_i. Since confounding Z_i will be given at first, we set the original value of τ_0 and τ by ourselves to calculate π_i from,

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \tau_0 + Z_i^T\tau$$

The π_i is

$$\pi_i = \left(1 + exp\left(-\left(\tau_0 + Z_i^T \tau\right)\right)\right)^{-1}$$

After that, we can get X_i from,

$$X_i \sim Ber\left(\pi_i\right)$$

Finally, we set the original value of β, intercept β_0, and γ in order to calculate Y_i,

$$Y_i \sim NegBin\left(\mu_i, \alpha\right); \log\left(\mu_i\right) = X_i\beta + \beta_0 + Z_i^T\gamma$$

To summarize, we should initially set the parameters β, β_0, γ, τ_0, and τ to their default values. The logistic model and negative binomial regression are then used to calculate X_i and Y_i based on these initial parameters and the confounding Z_i.

We have now successfully created Z_i, X_i, and Y_i. We could use the SCEPTRE or the new algorithm to calculate p-values based on this information.

4.2 Estimated Type I Error

A statistical model's Type I error and power calculation are critical. A good type I error or power indicates that the model is fit. In this section, we will look at the estimated type I error and power function. If the new algorithm is sufficiently good, it should have a 0.05 estimated type I error when the original z-score is 0.05 and be more powerful than SCEPTRE. The procedure for calculating estimated type I error and power is identical to that used for the Monte-Carlo simulation.

The statistical model can be written in general as,

$$Y \sim NB\left(\log\left(\alpha + \beta x + \gamma z\right), \theta\right)$$

The model is used to test for $\beta = 0$ to calculate the type I error. We could then simulate X, Y, and Z under $\beta = 0$ conditions. In the first step, we use SCEPTRE data to test if $\beta = 0$ and simulate the logistic model for $X|Z$. For each simulation of SCEPTRE data, we fit $NB\left(\log\left(\alpha + \beta x + \gamma z\right), \theta\right)$ under $\beta = 0$. The first step is repeated numerous times in the second step. Then, we compute one p-value by comparing repeated estimated β to the absolute value of the original z-score. In the third step, we fit a logistic model to $X|Z$, but this time we use a new algorithm to generate new X, which will be used to determine whether $\beta = 0$ in $NB\left(\log\left(\alpha + \beta x + \gamma z\right), \theta\right)$. Repeat this process as many times as necessary. The p-value from the new algorithm is then calculated by comparing repeated estimated β to the absolute value of the original z-score.

We obtained one p-value each from SCEPTRE and the new algorithm. We go through the entire process several times. The estimated type I error is the proportion of p-values from SCEPTRE and the new algorithm that are less than 0.05.

To summarize, in order to obtain the estimated type I error, we must set $\beta = 0$, which means that the guide RNA effect equals 0 in a simulated data process. The data is then used to calculate a z-score. Following the calculation of a set of new z-scores, we compare them to the original z-score under $H_0 : \beta = 0$. Then we get a single p-value. We repeat it several times to determine the number of p-values less than 0.05, which is the estimated type I error.

4.3 Estimated Type I Error Outcomes

The simulation is now used to demonstrate that the new algorithm has the correct estimated type I error. In the simulation we chose, the original data set is as follows: $\beta = 0$, $B = 500$, $\tau_0 = 0.1$, $\tau = 0.9$, $\gamma = 1$, and $\beta_0 = log5$. We employ a total of 5,000 simulations. Tables 4.1 through 4.3 show the results of the estimated type I error.

According to the results, the new algorithm BiasedUrn has a perfect estimated type I error of around 0.05.

4.4 Sample Size Recalculation

Three important numbers must be considered when simulating the new algorithm. The first is the sequence length of the presence of guide RNA. The second parameter is the number of resamples. Finally, the number of p-values used to calculate power is the third. The sequence length of the guide RNA presence is 500 in most cases, and the length of the guide RNA presence used in the paper 'SCEPTRE' is also 500. A longer sequence, such as 1000, could also be used. If we use the skew-t approximation, the number of resampling in each p-value calculation is 500. However, determining the number of p-values required to calculate power is difficult. Because the power is an estimate, it has its own confidence interval. The range of the confidence interval will be extremely large if the sample size is too small. The power function and power values may then be heavily skewed. The sample size deviations required to control the bias are listed in the appendix.

Table 4.1: Outcome 1 of the Estimated Type I Error

SCEPTRE	**BiasedUrn**
0.056	0.054

Table 4.2: Outcome 2 of the Estimated Type I Error

SCEPTRE	**BiasedUrn**
0.06	0.058

Table 4.3: Outcome 3 of the Estimated Type I Error

SCEPTRE	**BiasedUrn**
0.051	0.049

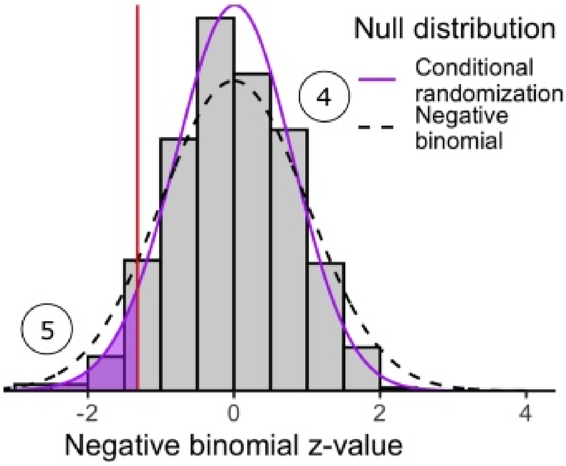

FIGURE 4.1: Skew-t Distribution

4.5 Skew-t Distribution Approximation

According to the appendix, the estimated number of z-values required is 182329 if we want to limit the difference in CI to 0.002 under the p-value of 0.05. However, because it is only one loop, it is impossible to calculate 182329 with a computer in most cases. Even using the DCC (Duke Computing Cluster) supercomputer will take a long time. As a result, we will introduce the skew-t distribution from SCEPTRE briefly. The skew-t distribution is depicted in Figure 4.1.

The conditional randomization test functions very similarly to the negative binomial distribution. They both have a bell shape. This means that the distribution of the resampled z-values may be normal. As a result, the p-value could be calculated by comparing the original z-value to the null distribution. We know that the number of samples required to simulate a p-value followed by a null distribution is small, roughly 182329. Actually, 500 in z-values is sufficient to simulate a set of data followed by a null distribution. As a result, in calculating p-values, we use 500 as the number of resampled z-values.

4.6 Skew-t Distribution Simulation Outcomes

In this section, we will use simulations to demonstrate that the skew-t distribution can be used to calculate p-values in the new algorithm. We run the simulation with $B = 500$, $\tau_0 = 0.1$, $\tau = 0.9$, $\gamma = 1$, and $\beta 0 = log 5$. The value of β will change, allowing us to have very small p-values. There will be 1,000,500 permutations of B. The first one million will be used to compute the true p-value. The final 500 will be used to compute the skew-t p-value. Table 4.4 displays the results.

4.7 Expectation Theory

This section focuses on the distribution of estimated p-values. In each simulation, we use a large number of resampled z-scores to calculate a single p-value. The distribution of these z-scores is normally distributed. If we want to confirm the correctness of these p-values, the distribution of p-values should follow a uniform distribution based on the null hypothesis $H_0 : \beta = 0$.

We use Q-Q plots to determine whether these p-values are suitable for further investigation. We need to compare the consistency of p-values and their expectations because we hope these p-values follow a uniform distribution. The expectations should be defined as uniform distribution expectations.

The estimated p-values will now be arranged in a new order in a vector Y. There-

Table 4.4: Skew-t Test Outcomes

gRNA-effect	**SCEPTRE**	**Skew-t**	**BiasedUrn**	**Skew-t**
0.35	$9.522903e-05$	$5.078626e-05$	$9.522903e-05$	0.0001947902
0.35	$9.522903e-05$	0.0002868456	$9.522903e-05$	0.0002061783
0.35	0.0004761451	0.0009929543	0.0002856871	0.0001456641
0.01	0.4920484	0.5124165	0.4913818	0.4967695
0.01	0.6239406	0.6081907	0.6407009	0.6328151
0.01	0.8284925	0.8135072	0.8093515	0.8129649

fore,

$$Y_p \;\sim\; sort(P - values)$$

Then we must compute the expectations for uniform distribution order statistics. It is,

$$X_p \;\sim\; c\left(\frac{1}{N+1}, ..., \frac{N}{N+1}\right)$$

where N is the number of p-values we consider.

If the p-value distribution is uniform, the points (X_p, Y_p) should follow the line $y = x$.

4.8 Q-Q Plot

The initial data set in simulation is: $M = 500$, $B = 5,000$, $\tau_0 = 0.1$, $\tau = 0.9$, $\beta = 0$, $\beta_0 = log(5)$. Figure 4.2 depicts the Q-Q plot.

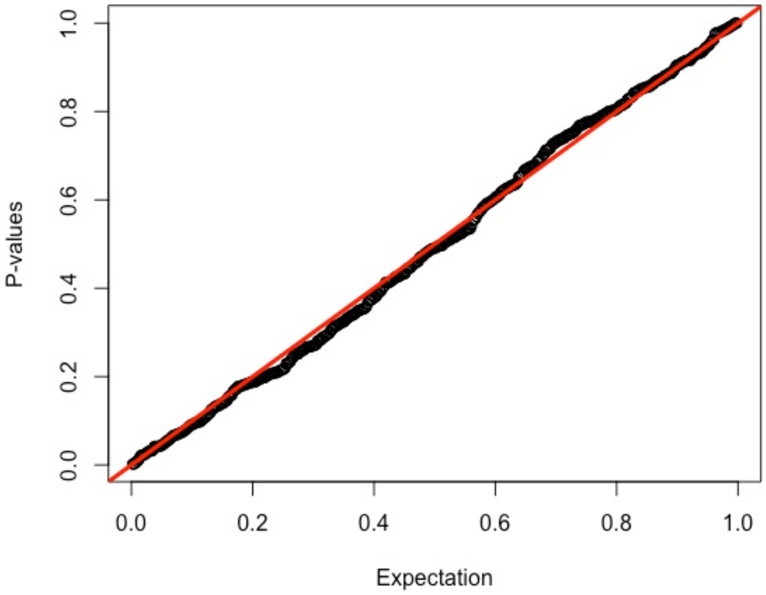

FIGURE 4.2: Q-Q Simulation Plot

Figure 4.3 depicts the distribution of P-values, also known as the 'Big-P.'

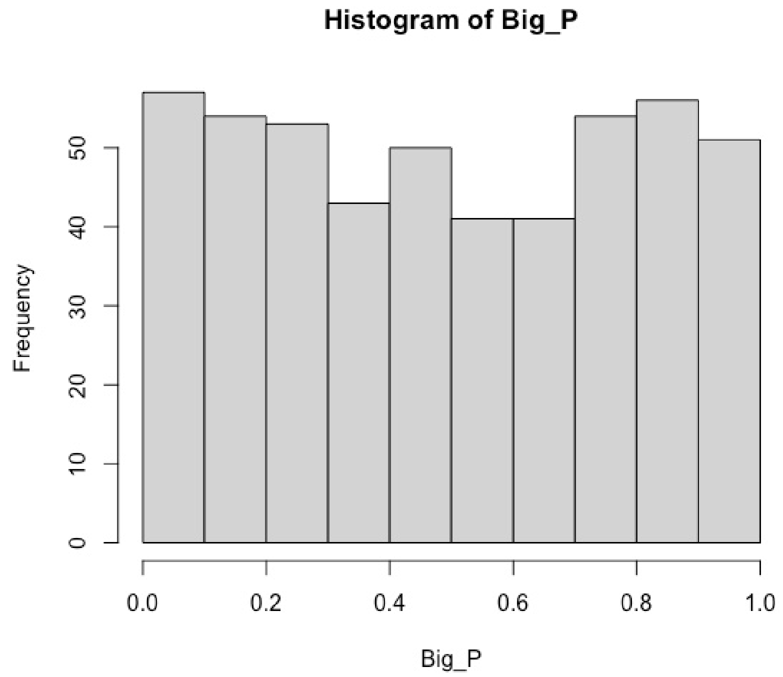

Figure 4.3: Distribution Plot

We can confirm that the generated p-values under the null hypothesis are adequate based on the distribution results and the Q-Q plot.

4.9 Estimated Power

To calculate the estimated power, we must set $\beta = t$, which means that the guide RNA effect equals t during the simulation process. Simultaneously, t is not equal to 0. The data can then be used to calculate a z-score. We compare the new z-scores to the original z-scores after calculating them under $H_1 : \beta = t$. Then we get a single p-value. This is repeated many times to calculate the number of p-values less than 0.05, which is the estimated power.

Figure 4.4 depicts the relationship between estimated type I error and estimated power.

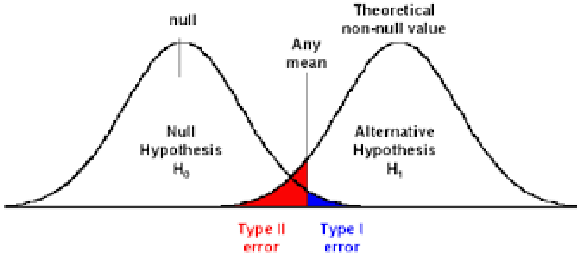

FIGURE 4.4: Type I Error, Type II Error, and Power

4.10 Estimated Power Outcomes

In this section, we will compare the results of powers and power functions between SCEPTRE and the new BiasedUrn algorithm.

The original data set used in outcome 1 is as follows:

$$\tau_0 = 0.04,\ \tau = 0.9,\ \gamma = 1,\ \beta_0 = log5,\ B = 5,000,\ M = 100$$

The power functions of SCEPTRE and BiasedUrn are depicted in Figure 4.5.

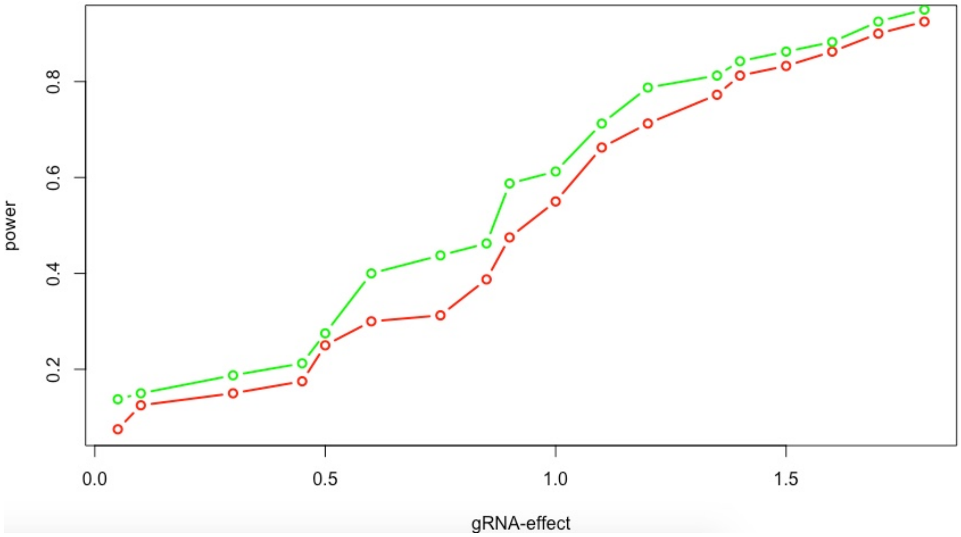

Figure 4.5: Power Function of Outcome 1

The original data set used in outcome 2 is as follows:

$$\tau_0 = 0.04,\ \tau = 0.9,\ \gamma = 0.3,\ \beta_0 = log5,\ B = 5,000,\ M = 5000$$

The power functions of SCEPTRE and BiasedUrn are depicted in Figure 4.6.

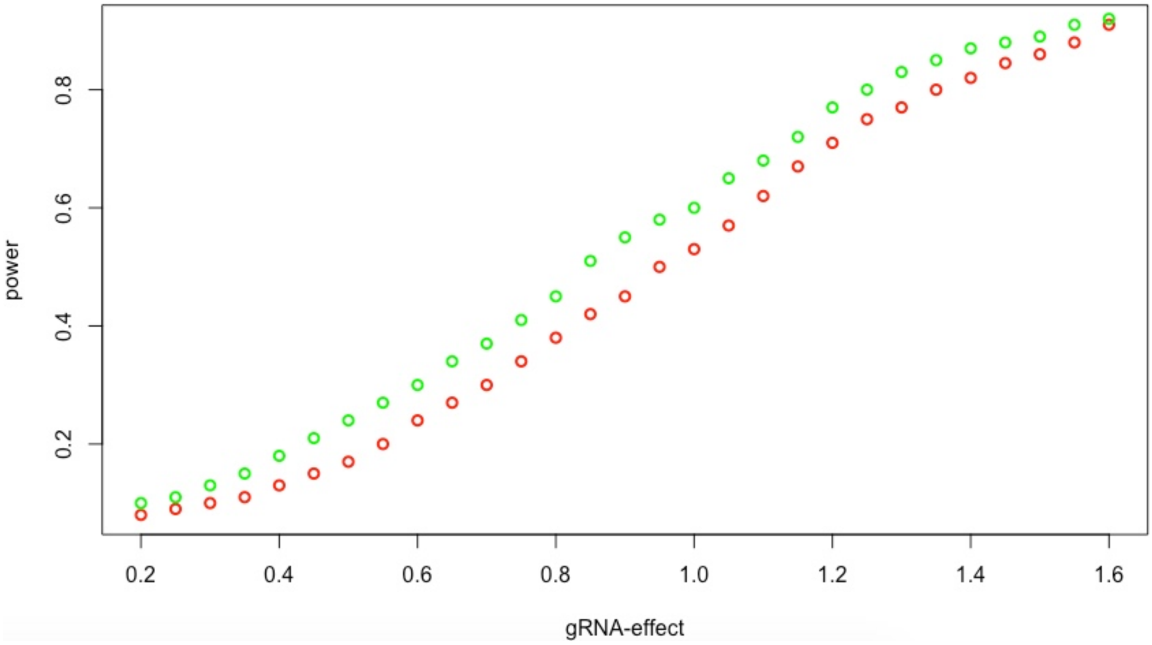

Figure 4.6: Power Function of Outcome 2

The original data set used in outcome 3 is as follows:

$$\tau_0 = 0.1,\ \tau_1 = 1,\ \tau_2 = -0.9,\ \gamma_1 = 1,\ \gamma_2 = -0.9,\ \beta_0 = log5,\ B = 5,000,\ M = 5000$$

The power functions of SCEPTRE and BiasedUrn are depicted in Figure 4.7.

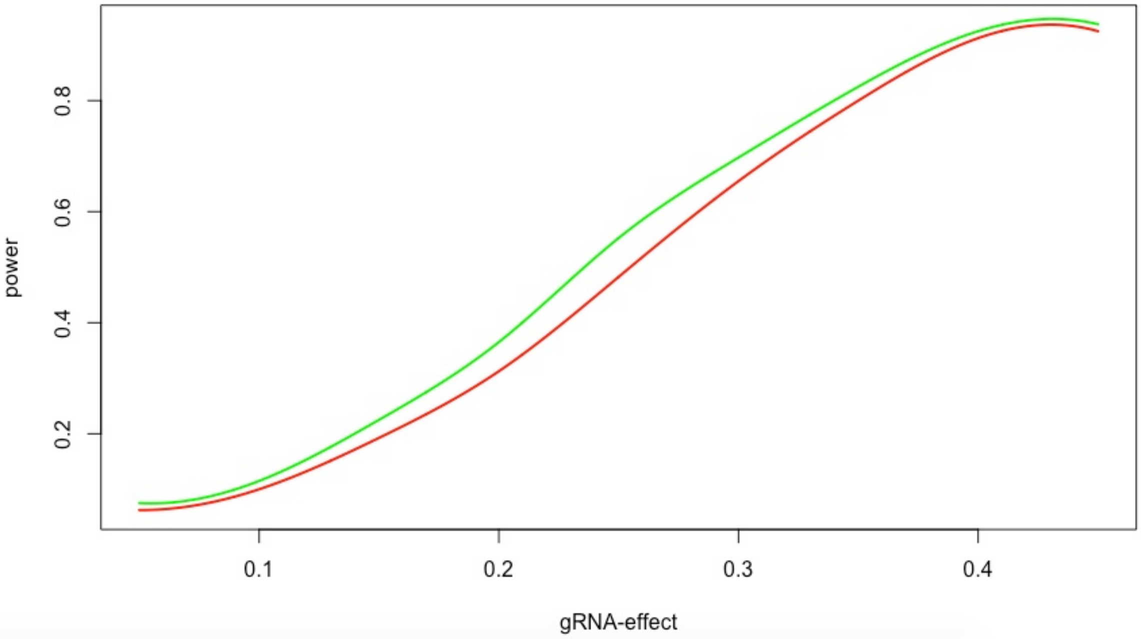

FIGURE 4.7: Power Function of Outcome 3

The original data set used in outcome 4 is as follows:

$$\tau_0 = 0.3,\ \tau_1 = -1,\ \tau_2 = 0.9,\ \gamma_1 = -1,\ \gamma_2 = 0.9,\ \beta_0 = log5,\ B = 5,000,\ M = 5000$$

The power functions of SCEPTRE and BiasedUrn are depicted in Figure 4.8.

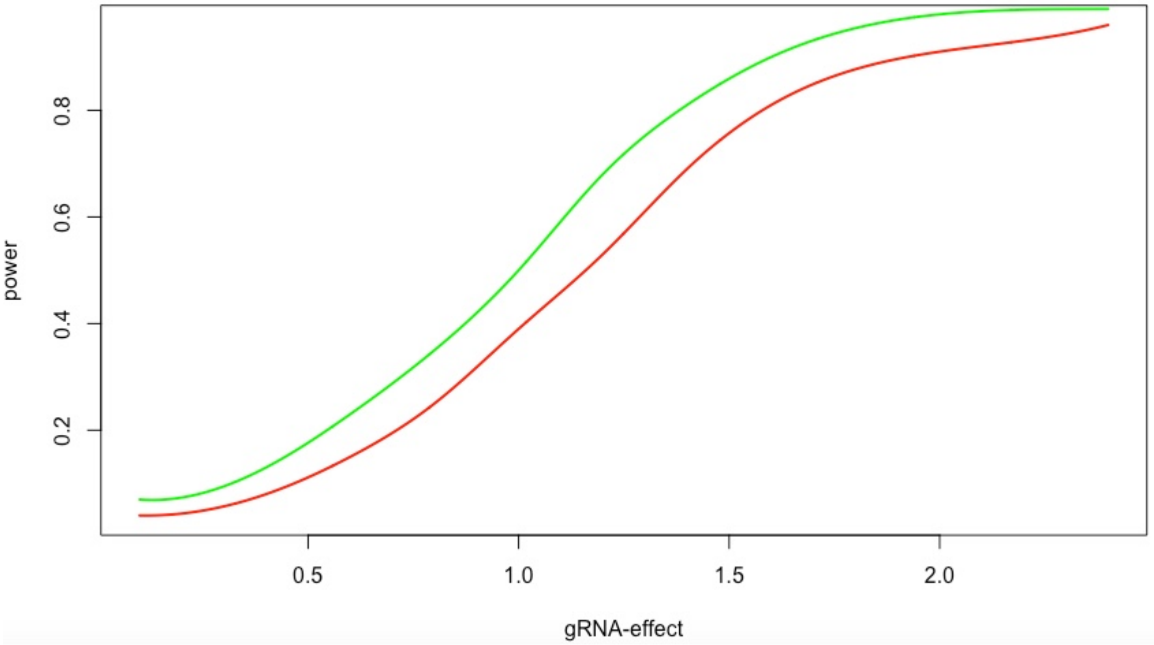

Figure 4.8: Power Function of Outcome 4

In simulation 1, the powers of SCEPTRE and BiasedUrn are shown in Table 4.5.

Table 4.5: Outcome 1 of Estimated Power

Guide RNA Effects	**Power of SCEPTRE**	**Power of BiasedUrn**
0.05	7.5%	13.75%
0.1	12.5%	15%
0.3	15%	18.75%
0.45	17.5%	21.25%
0.5	25%	27.5%
0.6	30%	40%
0.75	31.25%	43.75%
0.85	38.75%	46.25%
0.9	47.5%	58.75%
1	55%	61.25%
1.1	66.25%	71.25%
1.2	71.25%	78.75%
1.35	77.25%	81.25%
1.4	81.25%	84.25%
1.5	83.25%	86.25%
1.6	86.25%	88.25%
1.7	90%	92.5%
1.8	92.5%	95%

In simulation 2, the powers of SCEPTRE and BiasedUrn are shown in Table 4.6.

Table 4.6: Outcome 2 of Estimated Power

Guide RNA Effects	**Power of SCEPTRE**	**Power of BiasedUrn**
0.2	8%	10%
0.25	9%	11%
0.3	10%	13%
0.35	11%	15%
0.4	13%	18%
0.45	15%	21%
0.5	17%	24%
0.55	20%	27%
0.6	24%	30%
0.65	27%	34%
0.7	30%	37%
0.75	34%	41%
0.8	38%	45%
0.85	42%	51%
0.9	45%	55%
0.95	50%	58%
1	53%	60%
1.05	57%	65%
1.1	62%	68%
1.15	67%	72%
1.2	71%	77%
1.25	75%	80%
1.3	77%	83%
1.35	80%	85%
1.4	82%	87%
1.45	83%	88%

In simulation 3, the powers of SCEPTRE and BiasedUrn are shown in Table 4.7.

Table 4.7: Outcome 3 of Estimated Power

Guide RNA Effects	**Power of SCEPTRE**	**Power of BiasedUrn**
0.05	6.25%	7.5%
0.10	10%	11.5%
0.15	19.25%	22.5%
0.20	31.25%	36.5%
0.25	48.25%	55.25%
0.30	65.5%	69.75%
0.35	80%	82.5%
0.40	91.25%	92.5%
0.45	92.5%	93.75%

In simulation 4, the powers of SCEPTRE and BiasedUrn are shown in Table 4.8.

Table 4.8: Outcome 4 of Estimated Power

Guide RNA Effects	**Power of SCEPTRE**	**Power of BiasedUrn**
0.1	4%	7%
0.4	8%	13%
0.6	15%	23%
0.8	25%	35%
1.0	39%	50%
1.2	53%	68%
1.4	69%	81%
1.6	81%	90%
2.0	91%	98%
2.4	96%	99%

We can conclude from the figures and tables above that BiasedUrn has greater power than SCEPTRE.

5

Summary and Limitations

5.1 Limitations

My master thesis also has numerous limitations. More estimated power and skew-t approximated p-values, for example, are required based on different initial values. Much more research is needed to determine whether the skew-t approximation is appropriate when p-values are very small. Gene expression, on the other hand, is a very long sequence with many zeros that represent missing data. It is difficult to ensure the correctness of SCEPTRE and BiasedUrn if there are missing values in gene expression. More work on missing data analysis in genetic datasets could be done.

5.2 Summary

In my master's thesis, we first discussed the new biotechnological methods single-cell CRISPR screens, as well as their drawbacks and calibration issues. Then we discussed SCEPTRE, which is a method for associating perturbations and expressions while avoiding calibration through resampling. SCEPTRE, on the other hand, is insufficient. It is highly variable because the number of 0 values in the guide RNA presence sequence does not equal the number of 1 values in each resampling. Based on this, we consider BiasedUrn, another permutation method in which the number of 0 values in the guide RNA presence sequence equals the number of 1 values in each resampling. Finally, we ran numerous simulations to demonstrate that the new algorithm BiasedUrn has correct type-I error, uniform p-value distribution, and higher powers than SCEPTRE. We make a small contribution to single-arm cell research.

This project has provided me with a wealth of statistical knowledge and computer science skills. I've learned how to solve problems in a variety of challenging situations. I've learned how to conduct extensive research on my own. Confounding between gene expression and guide RNA is still a significant issue in statistical genetics research. More work and research in this area will be required in the future.

Neil Armstrong has said 'That's one small step for man, one giant leap for mankind'. From this paper, I want to indicate 'That's one small step for mankind, one giant leap for me'. Keep on going!

5.3 R Codde

Code is available at: `https://github.com/Hongyu23/BiasedUrn_SCEPTRE`

6

Reference

[1] Timothy Barry, Xuran Wang, John A. Morris, Kathryn Roeder, Eugene Katsevich. 'SCEPTRE improves calibration and sensitivity in single-cell CRISPR screen analysis'. Genome Biology. (2021) 22:344.

[2] Michael P. Epstein, Richard Duncan, Yunxuan Jiang, Karen N. Conneely, Andrew S. Allen, Glen A. Satten. 'A Permutation Procedure to Correct for Confounders in Case-Control Studies, Including Tests of Rare Variation'. American Society of Human Genetics. 2012.

[3] Emmanuel Candes, Yingying Fan, Lucas Janson, Jinchi Lv. 'Panning for Gold: Model-X Knockoffs for High-dimensional Controlled Variable Selection'.

[4] Abhishek Sarkar, Mattew Stephens. 'Separating measurement and expression models clarifies confusion in single-cell RNA sequencing analysis'. Nature Genetics.

[5] D. Y. Lin. 'An efficient Monte Carlo approach to assessing statistical significance in genomic studies'. Bioinformatics, Vol.21 no.6 2005, pages 781-787.

[6] George Casella, Roger L. Berger. 'Statistical Inference'. 2nd Edition.

7

Appendix

7.1 Sample Size Calculation In Estimated Power

The estimated power could be calculated below.

$$\widehat{Power} = \sum_{i=1}^{Sample\ Size} \frac{I\left(P_i \leqslant 0.05\right)}{Sample\ Size} \sim Bernoulli\left(P_1\right)$$

The true power is,

$$Power = Probability\left(P_i \leqslant 0.05\right)$$

The variance of the estimated power could then be calculated. It is,

$$Var\left(\widehat{Power}\right) = Var\left(\sum_{i=1}^{Sample\ Size} \frac{I\left(P_i \leqslant 0.05\right)}{Sample\ Size}\right)$$

$$= \frac{1}{\left(Sample\ Size\right)^2} \sum_{i=1}^{Sample\ Size} Var\left(I\left(P_i \leqslant 0.05\right)\right)$$

$$> \frac{Sample\ Size}{\left(Sample\ Size\right)^2} Var\left(I\left(P_i \leqslant 0.05\right)\right)$$

Then,

$$Var\left(\widehat{Power}\right) = \frac{P_1\left(1 - P_1\right)}{Sample\ Size}$$

Therefore,

$$\widehat{Var}\left(\widehat{Power}\right) = \frac{\widehat{Power}\left(1 - \widehat{Power}\right)}{Sample\ Size}$$

According to central limit theory, we could have,

$$\widehat{Power} \longrightarrow N\left(P_1, Var\left(\widehat{Power}\right)\right)$$

Now, we could have the 95% CI,

$$\widehat{Power} \pm 1.96\sqrt{Var\left(\widehat{Power}\right)}$$

$$= \widehat{Power} \pm 1.96\sqrt{\frac{\widehat{Power}\left(1 - \widehat{Power}\right)}{Sample\ Size}}$$

The greater the sample size, the smaller the CI. We must take into account the median power 0.5 because $power \times (1 - power)$ will be largest if $power = 1 - power$. If we want to control the CI [0.49 0.51], then,

$$1.96\sqrt{\frac{\widehat{Power}\left(1 - \widehat{Power}\right)}{Sample\ Size}} = 0.01$$

As a result, the sample size will be 9604. If we want a precise estimated power and a better shape of the power function, the sample size should be 9604.

7.2 Sample Size Calculation In Estimated p-value

The following is the derivation of sample size calculation in estimated p-value.

$$P\widehat{-value} = \sum_{i=1}^{Resampled\ Z-values} \frac{I\left(z - value \leqslant z - score\right)}{Resampled\ Z - values} \sim Bernoulli\left(P_2\right)$$

The true p-value is,

$$P - value = Probability\left(z - value \leqslant z - score\right)$$

The variance of the p-value could then be calculated. It is,

$$Var\left(\widehat{P - value}\right) = Var\left(\sum_{i=1}^{Resampled\ Z-values} \frac{I\left(z - value \leqslant z - score\right)}{Resampled\ Z - values}\right)$$

$$= \frac{1}{\left(Resampled\ Z - values\right)^2} \sum_{i=1}^{Resampled\ Z-values} Var\left(I\left(z - value \leqslant z - score\right)\right)$$

$$> \frac{Resampled\ Z - values}{\left(Resampled\ Z - values\right)^2} Var\left(I\left(z - value \leqslant z - score\right)\right)$$

Then,

$$Var\left(\widehat{P - value}\right) = \frac{P_2\left(1 - P_2\right)}{Resampled\ Z - values}$$

Therefore,

$$\widehat{Var}\left(\widehat{P - value}\right) = \frac{\widehat{P - value}\left(1 - \widehat{P - value}\right)}{Resampled\ Z - values}$$

According to central limit theory, we could have,

$$\widehat{P - value} \longrightarrow N\left(P_2, Var\left(\widehat{P - value}\right)\right)$$

The 95% CI will be,

$$\widehat{P - value} \pm 1.96\sqrt{Var\left(\widehat{P - value}\right)}$$

$$= \widehat{P - value} \pm 1.96\sqrt{\frac{\widehat{P - value}\left(1 - \widehat{P - value}\right)}{Resampled\ Z - values}}$$

The CI will be smaller as the number of Resampled Z-values increases. Consider the p-value of 0.05 as an example.If we want to control the CI between [0.049 0.051], then,

$$1.96\sqrt{\frac{P-\widehat{value}\left(1-P-\widehat{value}\right)}{Resampled\ Z-values}} = 0.001$$

If we want a precisely estimated p-value, the number of resampled z-values will be 182329.

7.3 Skew-t Distribution

We mentioned in the previous chapter that the distribution of resampled z-values follows the skew-t distribution. This is why 500 is sufficient for the number of resamplings. Now we'll go over the history of the skew-t distribution.

Skew-t distribution has the pdf of:

$$f_{SGT}(x;\mu,\delta,\lambda,p=2,q)$$

$$= f_{ST}(x;\mu,\delta,\lambda,q) = \frac{\Gamma(\frac{1}{2}+q)}{v\delta(\pi q)^{1/2}\Gamma(q)(\frac{|x-\mu+m|^2}{q(v\delta)^2(\lambda sign(x-\mu+m)+1)^2}+1)^{\frac{1}{2}+q}}$$

where:

$$m = \frac{2v\delta\lambda q^{1/2}\Gamma(1-\frac{1}{2})}{\pi^{1/2}\Gamma(q)}$$

gives a mean of μ. Also:

$$v = \frac{1}{q^{1/2}\sqrt{(3\lambda^2+1)(\frac{1}{2q-2})-\frac{4\lambda^2}{\pi}(\frac{\Gamma(q-\frac{1}{2})}{\Gamma(q)})^2}}$$

In the above formulas, μ is the location parameter, $\delta > 0$ is the scale parameter, $-1 < \lambda < 1$ is the skewness parameter, and $p > 0$ and $q > 0$ are the kurtosis parameters. m and v are not parameters, but rather functions of the other parameters used to scale or shift the distribution to match the various parameterizations of this distribution. In general, the plot of a skewed t distribution resembles that of a normal distribution. However, the skewed t distribution has a lower curve than the normal distribution.

$$\hat{\mu_1} = (\hat{\pi_0} + \hat{\pi_3})\hat{\mu_1}^{(0)} + \hat{\pi_1}\hat{\mu_1}^{(1)} + \hat{\pi_2}(\hat{\beta}^{(0)}_{10\cdot 2} + \hat{\beta}^{(0)}_{12\cdot 2}\hat{\mu}^{(2)}_2)$$