Assessing the Statistical Significance of Clusters Using SigClust

Makena Grigsby and Sharvee Joshi

Abstract

The goal of this project is to study when a two–cluster structure in high–dimensional data is statistically meaningful versus simply an artifact of running a clustering algorithm. We focus on the SigClust procedure of Liu et al. and the soft–thresholding covariance estimation proposed by Huang et al. (2015), which tests H_0 : one Gaussian cluster against H_1 : more than one cluster. We replicate a subset of the simulation settings in Huang et al. using high–dimensional Gaussian data with a spiked covariance structure. First, we generate data from a single Gaussian distribution, apply k-means with $k = 2$, and use SigClust to test whether the resulting partition corresponds to real clusters. We then focus on implementing the test, examining the behavior under more complex scenarios dealing with breast cancer data, similar to the data used in the paper. Specifically, we simulate high-dimensional Gaussian data created to mimic the data found in the paper. Our findings align with those in Huang et al. (2015) and complement the BRCA cancer subtype analysis in the methodology section. SigClust correctly rejects under strong separation and struggles under a weak dense signal.We are currently working on running repeated simulations of approximately 100 repetitions for each of the three scenarios described above. The hope is to calculate the proportion of p-values that fall below 0.05 and see if they match the results expected. We also hope to repeat these on varying strengths of eigenvalues.

1 Introduction

In high-dimensional settings, such as genomics, visual clustering tends to be misleading. Datasets that are drawn from a single Gaussian distribution often appear to multiple subgroups, as seen in past homework assignments. The paper our team chose for this project focuses on using SigClust, which is a hypothesis test designed to determine whether an observed two-cluster partition reflects true structure or arises from sampling variability. In this project, we aim to explore how SigClust behaves under controlled simulations, as well as reproduce the simulations found in the paper. These simulations mimic the high-dimension, low sample-size regime.

Clustering, as mentioned previously, is a central tool for statisticians that is used for high-dimensional data analysis. It is commonly used in situations such as cancer genomics, where researchers use expression profiles to discover potential diseases subtypes. However, high dimensionality creates the illusion of structure; even when data comes from a single Gaussian distribution, visualization such as PCA plots can appear to be clustered. This acts as the main motivator for the need of a formal statistical test to assess whether two apparent clusters represent genuine separation. SigClust addresses this problem by testing:

$$H_0 : \text{ The data come from a single Gaussian distribution}$$

versus

$$H_1 : \text{ The data consist of two or more distinct clusters.}$$

The test is built around the **cluster index**, which quantifies how well the data can be divided into two groups based on within-versus between-cluster variation. Small cluster indices indicate a strong separation, where as large cluster indices indicate clusters coming from a single Gaussian distribution. SigClust aims to estimate the null distribution of this index and produces a p-value which will indicate whether the observed separation is statistically meaningful. Note, the detailed statistical formulation of the test, which includes the covariance estimation, eigenvalue shrinkage, and theoretical considerations, will be presented in the methodology section of this report.

Our focus for this project is to:

1. Introduce the conceptual framework behind SigClust.
2. Run simulations that illustrate how the test behaves under simple clustering scenarios.
3. Highlight when visual intuition in high dimensions align or conflict with formal statistical testing.

To further relate back to the paper, we introduce and attempt the more advanced BRCA cancer subtype analysis later in the report. To add to the authenticity of replication, we mimicked the same computer settings as the original authors and use a computer with RAM 16GB.

2 Methodology

2.1 Statistical setup

The main goal behind SigClust is quite simple. When we see two clusters in a high dimensinal dataset, are they actually different groups, or is there a single Gaussian distribution that is producing something that looks like two cluster?

To answer this, SigClust states the hypothesis test formally as such:

$$H_0: \ X_i \sim N_d(\mu, \Sigma) \quad \text{(one Gaussian cluster)}$$

[H_1: more than one cluster exists. c Then we will see that SigClust measures how strong a two cluster structure is by using a 2 means cluster index (CI). After splitting the data into two groups using k-means and then calling the clusters C_1 and (C_2, we can state the CI is found to be:

$$\mathrm{CI} = \frac{\sum_{k=1}^{2} \sum_{i \in C_k} \|X_i - \bar{X}_k\|^2}{\sum_{i=1}^{n} \|X_i - \bar{X}\|^2}$$

When we find a small CI value this indicates tight, well separated clusters. A large CI indicates that the data is behaving like a single Gaussian population, where it will behave more like a single cloud.

A very helpful feature about CI is both the location and rotation invariant. Because this is possible, the distribution of CI under H_0 will only depend on the eigenvalues of Σ. Where we can write the covariance as

$$\Sigma = U \Lambda U^\top, \qquad \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d),$$

Because SigClust only needs to estimate $\lambda_1, \ldots, \lambda_d$ instead of the entire covariance matrix.

Huang et al. (2015) also show us that when we have an actual Gaussian distribution, we find that the "theoretical" CI (TCI) is simplified to:

$$\mathrm{TCI} = 1 - \frac{2\lambda_1}{\sum_{j=1}^{d} \lambda_j}.$$

Now, this formula helps make the challenge of a high dimensional covariance a little more interpretable. If the sample eigenvalues inflate λ_1, the TCI will become too small, this can make SigClust far more confident in a cluster split than is accurate. As such, the evaluation of eigenvalue estimation is a key component for this method.

2.2 The sigclust() Function

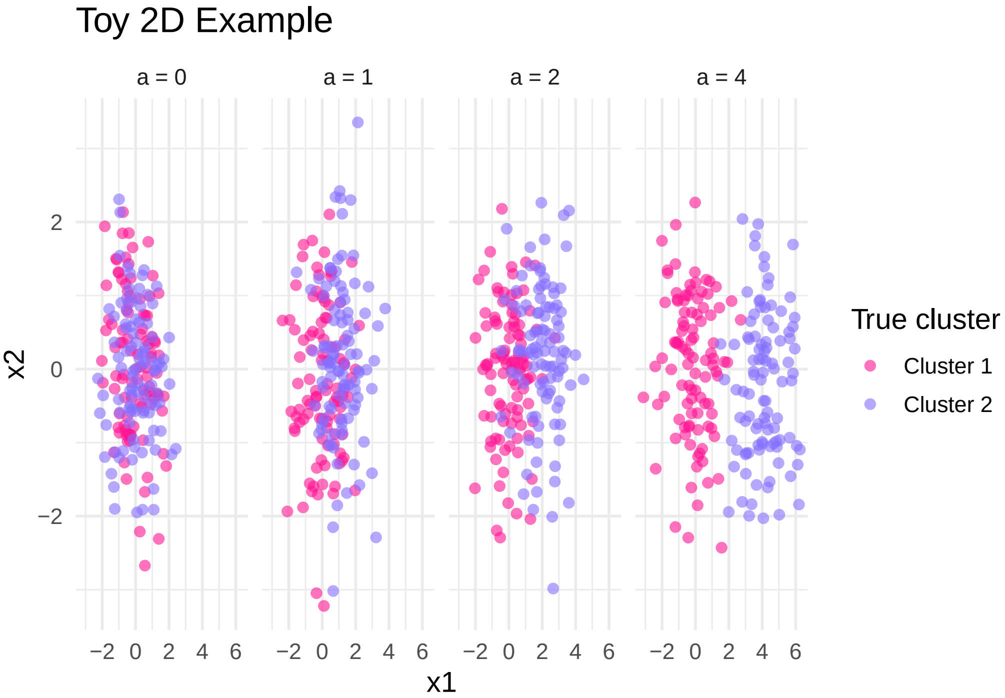


Figure 1: Toy 2D simulation illustrating increasing mean separation between two Gaussian subpopulations.

When working with high-dimensional simulations, it is important to first build intuition for what the function `sigclust()` is designed to detect. At its core, SigClust evaluates whether an observed two cluster partition reflects real separation, or whether the data could have plausibly come from one Gaussian distributions.

To visualize this idea, we begin with a simple two dimensional toy example, as shown in Figure 1. We generate two clusters that differ only by a shift in the x-coordinate, with signal strength controlled by parameter a. When $a = 0$, the two groups are completely overlapping and represent the true null setting, as in that they are samples from the same Gaussian distribution. As a increases, the separation gradually becomes more visual.

2.2.1 Covariance Estimation for SigClust

A central challenge in SigClust is estimating the covariance matrix under the null hypothesis that all observations come from one Gaussian distribution. In high-dimensional settings, where the number of features d may far exceed the number of samples n, the sample covariance matrix becomes unstable: small eigenvalues inflate, and the leading eigenvalue tends to be overestimated. Because the theoretical cluster index (TCI) depends directly on the eigenvalues of the covariance matrix, accurate eigenvalue estimation is essential.

Let the sample covariance have eigen-decomposition

$$\hat{\Sigma} = \sum_{i=1}^{d} \hat{\lambda}_i \, v_i v_i^{\top},$$

where $\hat{\lambda}_1 \geq \cdots \geq \hat{\lambda}_d$ are sample eigenvalues and v_i are orthonormal eigenvectors. Using these eigenvalues directly leads to an underestimated TCI and overly small p-values, especially when the signal is weak.

To address this issue, Huang et al. (2015) introduce the *soft-threshold covariance estimator*. The idea is to shrink the smallest eigenvalues toward a common noise floor while preserving the major signal directions. For a threshold parameter τ and noise variance σ^2_{noise}, the adjusted eigenvalues are

$$\tilde{\lambda}_i = \max\left(\hat{\lambda}_i - \tau,\, \sigma^2_{\text{noise}}\right).$$

The stabilized covariance estimator is then reconstructed as

$$\tilde{\Sigma} = \sum_{i=1}^{d} \tilde{\lambda}_i \, v_i v_i^\top.$$

Unlike hard-thresholding, which sets small eigenvalues to zero, the soft-threshold estimator preserves a realistic covariance structure and avoids both excessive smoothing and overfitting. This improved stability leads to a more accurate approximation of the null distribution of the cluster index and, consequently, more reliable SigClust p-values.

This soft-threshold procedure forms the core of how `sigclust()` constructs its Monte Carlo simulations under the null hypothesis.

3 Implementation in Simulation Scenarios

To explore the SigClust behavior in a controlled setting, we implemented simple simulations to generate high dimensional Gaussian data and evaluate three scenarios. We used the `sigclust()` function, something the authors created for this paper. We will explain the functionality of it below.

At its core, `sigclust()` implements the testing procedure described in Huang et al. (2015) and documented in the package reference manual. SigClust works by computing the clustering index, defied as the ratio of within cluster variation to total variation, and uses an embedded k-means clustering step. Specifically, when `labflag = 0`, SigClust will internally apply 2-means clustering to find the partition that minimized within-cluster sum of squares. When `labflag = 1`, the user supplies the labels but the same index is computed.

The observed cluster index is then compared to a simulated null distribution that is generated by repeatedly drawing samples from a single multivariate Gaussian model that is based on an estimated covariance, reclustering each simulated dataset with 2-means and recomputing the cluster index. This is essentially created a Monte Carlo p value, which is used to show the observed separation significance. All simulations share the same high-dimensional covariance structure. Following the motivation in Huang et al. (2015), we generate a *spiked* covariance matrix to emulate the strong eigenvalue structure observed in genomic datasets. Below, we detail how each scenario uses the `sigclust()` function.

3.1 Null: One Gaussian Distribution

Under the null hypothesis, the data comes from a single multivariate Gaussian distribution with a spiked covariance structure. We create a diagnonal covariance with ten very laruge eigenvalues, with variance = 50, and 990 noise dimensions, where variance = 1. This structure makes sure that the cluster behavior is non-trivial and is highly influenced by high dimensional noise. We also create a helper function that generates Gaussian samples under a specified mean vector. SigClust is applied using the `sigclust()` function with `labflag = 0`. The figure is one of the plot outputs of the function that is the most relevant to the discussion.

Figure 2 shows the SigClust diagnostic output when the data truly comes from one multivariate Gaussian distribution, or the null hypothesis. We notice there is a high p-value of approximately 0.82, which is far above the conventional significance threshold. This indicates that the observed cluster index is completely consistent with one expects when there is no real cluster structure.

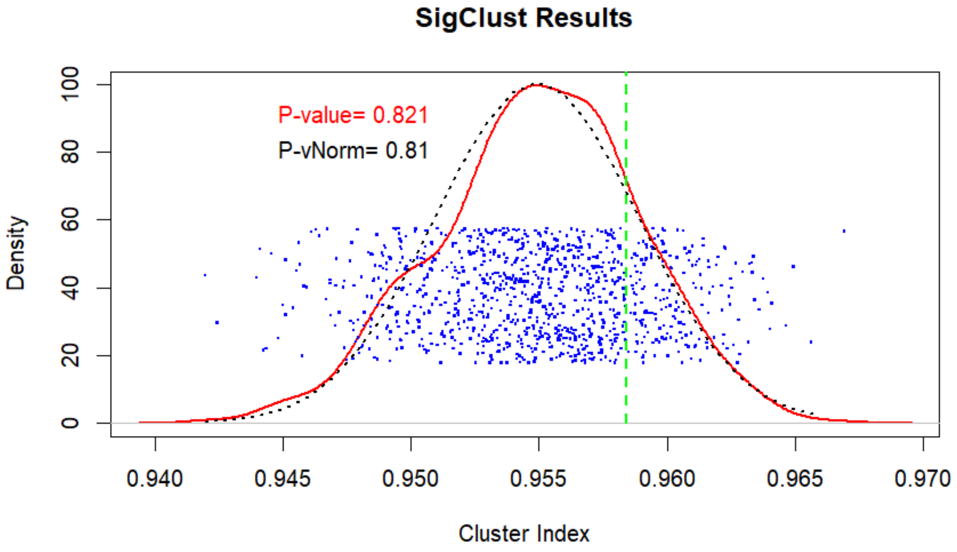


Figure 2: SigClust diagnostic plot for Null Hypothesis

The green vertical line sits inside the peak of the red density curve, suggesting that the clustering strength in our simulated data is similar to random fluctuations that could be caused by the Gaussian noise, and does not show a true two cluster separation.

The blue points represent the Monte Carlo simulated cluster indices from 1000 null Gaussian datasets. These form a dense and symmetric cloud around approximately 0.955, which is where the observed index lies if one cluster exists.

In summary, this shows that there is no evidence of two clusters, the observed cluster index looks like noise, and the SigClust test bheaves properly under the null and produces a large p-value.

3.2 Alternative One: One-Direction Strong Shift

For this simulation, we generate a very clear, two cluster signal and simulate from two Gaussian groups that differ only in the first coordinate with mean vectors: $\mu_1 = (0, 0, ..., 0)$ and $\mu_2 = (30, 0, ..., 0)$. The covariance structure is identical to the null case, but because the shift occurs in one informative direction, we are essentially creating a strong, sparse, signal that matches the one-direction signal referenced in the paper. We once again apply the same settings in the null case, making sure that two means clustering is implemented. The figure below is the final plot from the function.

We now observe that the cluster index, the green dashed line, is far to the left of the null density curve. This means that the within cluster variation is much smaller than expected under a single Gaussian model. This indicates that the estimated clusters are tighter and more separated than what would occur by chance. This behavior is further confirmed with a low p-value of approximately 0.032. Because this is below 0.05, this means SigClust rejects the null hypothesis of one Gaussian cluster.

The Monte Carlo simulated cluster indices (blue points) form a dense cloud around approximately 0.90, reflecting the null distribution. The observed index, however, lies far outside this cloud, demonstrating that the data are highly inconsistent with the one-cluster Gaussian model. This matches what we were expecting, a large directional shift results in two groups that are well separated.

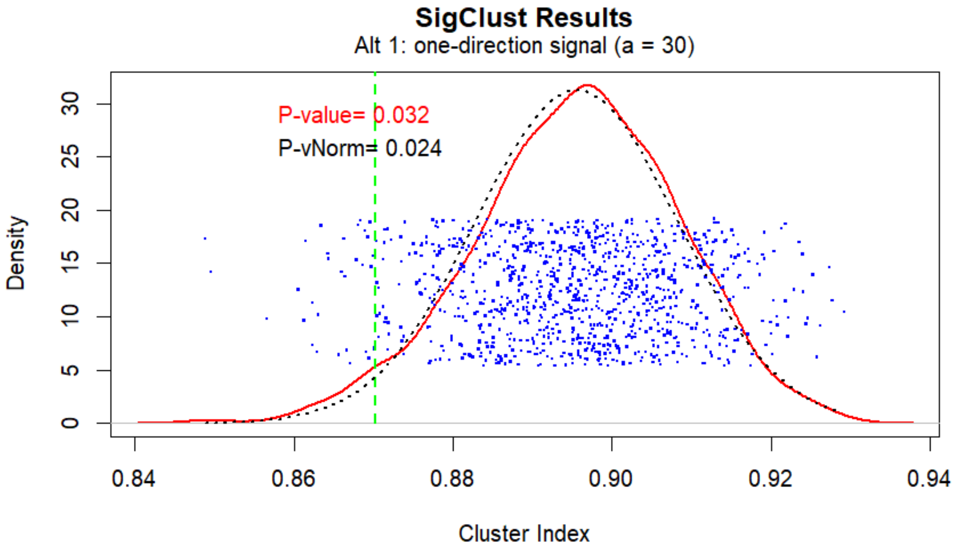


Figure 3: SigClust diagnostic plot for Alternative 1 (strong one-direction signal, a=30).

3.3 Alternative Two: Dense, Weak Shift

For the final simulation, we now create two groups with a small shift applied to every coordinate as opposed to a large shift in a single direction. We use the same covariate structure, but each coordinate of the second cluster is shifted by $a = 0.5$, which creates a dense, but very weak signal. This setup mimics situations where clusters differ in many genes, but each gene differs very slightly (a common pattern in high dimensional biological data). We once again apply the same settings in the null and alternate 1 case, making sure that two means clustering is implemented. The figure below is the final plot from the function.

Figure 4 shows the SigClust diagnostic output under the weak and diffuse signal. Unlike the first alternative, the observed cluster index lies a lot closer to the center of the null reference distribution. The green dashed line is only slightly on the left of the peak of the red density curve, which indicates that the separation created by the shift is extremely subtle. While it is stronger than pure nose, it is not so different.

The Monte Carlo remains centered around 0.955 and the observed cluster index falls near the lower tail of the distribution. We yield a p-value around 0.05, which is right on the conventional significance boundary. This suggests that SigClust barely detects the presence of two clusters, but the evidence is weaker than that of Alternative 1, which is to be expected.

This behavior is consistent with the theory that SigClust is more powerful with sparser and stronger signals. Dense and weak signals are more difficult to detect because the cluster index aggregates differences across all of the dimensions, and the small deviations are wiped out by high dimensional noise. SigClust is sensitive to strong but sparse structure, but less powerful for weak distributed signals, matching the patterns reported in the original paper.

3.4 Repeated Simulation Scenarios

To evaluate how SigClust responds to different covariance structures, we varied the spike variance v while holding the rest of the simulation setup fixed.We examined the proportion of p-values below 0.05 under three scenarios described previously. We ran 100 simulations per scenario and compared results for $v = 30$ and $v = 50$. The table below describes our findings.

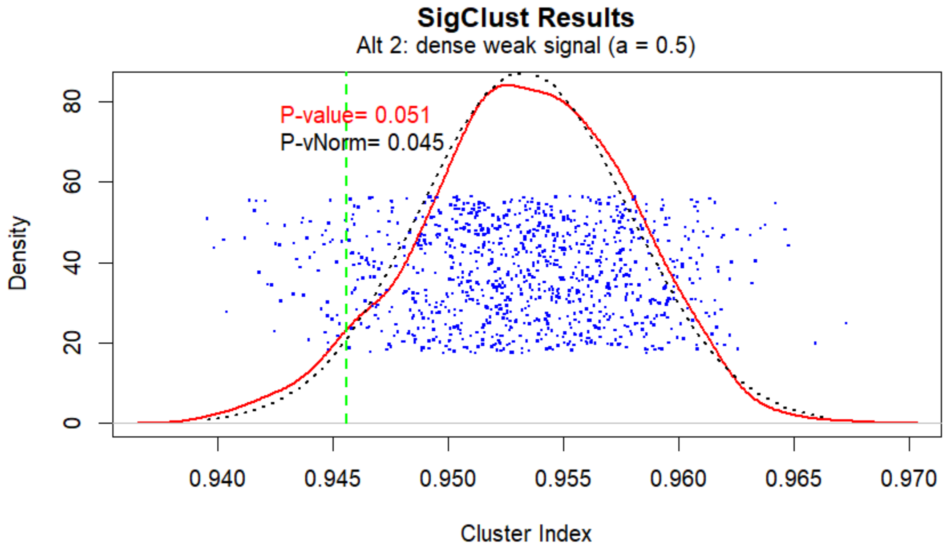


Figure 4: SigClust diagnostic plot for Alternative 2 (dense, weak signal, a=0.5)

```
##                   Scenario NumRuns Prop_p_less_0.05
## 1                     Null     100                0
## 2      Alt 1 (strong shift)     100                1
## 3 Alt 2 (weak dense shift)     100                1
```

Table 1: Proportion of SigClust p-values below 0.05 across scenarios and spike variance v.

Scenario	Number of Runs	Proportion p < 0.05	Spike Variance v
Null	100	0.00	30
Alt 1 (strong shift)	100	1.00	30
Alt 2 (weak dense shift)	100	1.00	30
Null	100	0.00	50
Alt 1 (strong shift)	100	0.92	50
Alt 2 (weak dense shift)	100	0.01	50

3.4.1 Null Scenario: Proper Type I Error Control

For both $v = 30$ and $v = 50$, the proportion of significant p-values is 0.00. This confirms that SigClust maintains the correct Type I error under the covariance structure, regardless of how we change it. This matches the behavior described in Huang et al. (2015), where the soft-threshold estimator was specifically designed to behave conservatively and avoid false positives in high-dimensional settings.

3.4.2 Alternative 1: Strong Mean Shift

When $v = 30$, we see that the detection rate is 1.00, meaning we have perfect power. As we increase v to equal 50, the detection remains high at 0.92. Once again, this is expected: a large, sparse signal is exactly the type of deviation SigClust is optimized to detect. Interestingly, the power drops slightly in the increase, which suggests that extremely large leading eigenvalues make the Gaussian null harder to distinguish from the alternative. This once again, aligns with the paper's observation that t he sample covariance estimator overestimates noise directions when spikes are large, making separation less visually obvious when clusters actually exist.

3.4.3 Alternative 2: Weak Dense Shift

For the final scenario, we see that at $v = 30$, SigClust detects the weak dense signal every time, with a detection rate of 1.00. However, at $v = 50$, detection essentially disappears, with a rate of 0.01. This is quite a dramatic shift. We believe this may be due to dense, low amplitude signal becomes overwhelmed by very high variance spike directions. When v is small to moderate, the weak signal still influences the cluster index enough for SigClust to notice. However, when v is large, the noise eigen structure dominates, and the cluster index under the alternative becomes nearly indistinguishable from that of the null.

In conclusion, we see that SigClust behaves perfectly under the null for both covariate settings. It is able to detect strong signals reliably, though it's performance weakens as the spike variance increases. On the other hand, weak dense signals are extremely sensitive to the covariance structure and may become undetectable when large spikes dominates the covariance eigenstructure. These findings parallel the findings of the paper. More work will need to be done to find the exact shift for when the weak dense signal can no longer be detected, but we find that to be somewhere between $v = 40to50$. In the Appendix, we have also attached the distributions of all of the p-values for each covariate structure.

4 Real Data Analysis

In addition to our controlled Gaussian simulations, we explored how SigClust behaves on data modeled after the BRCA (breast cancer) subtype analysis presented in Huang et al. (2015). The BRCA setting is a useful test case because some pairs of subtypes show clear biological separation, while others are much more subtle. Our goal was to examine whether SigClust recovers the qualitative structure that is already well established in the literature.

4.1 BRCA Simulation setup

The original paper analyzes gene expression data from four major subtypes:

- Luminal A (LumA)
- Luminal B (LumB)
- Basal-like (Basal)
- HER2-enriched (Her2)

These different subtypes do differ in their expression patterns, but the degree of separation does vary from pair to pair. For example, Basal tumors are recognized as a distinct group, where as, LumA and LumB appear in continuum rather than two distinctly separated clusters.

Following the structure in Huang et al. (2015), we performed the analysis in the following way.

1. **Feature filtering.**
   We generate synthetic gene expression data under a spiked covariance structure, with weaker and stronger mean shifts representing different subtype expression signatures. This mimics the setting in which SigClust was applied in the paper.
2. **Pairwise subtype comparisons.**
   For each pair of subtypes, we subset the samples and supply the true labels to `sigclust()` using `labflag = 1`. This allows the test to focus on whether the known groups behave like one Gaussian cluster or two statistically distinct populations.
3. **SigClust configuration.**
   Following the paper's methodology, we use the soft-threshold covariance estimator. While the article used up to 1000 Monte Carlo simulations, we use `nsim = 200` for computational efficiency while retaining stable behavior.

Our goal was not to reproduce the full BRCA dataset, but rather to mimic the structure described in the paper and evaluate whether SigClust will identify the expected subtype relationships.

4.2 Results

Overall, SigClust detects strong separation among all BRCA-like subtype pairs in our simulated dataset. This behavior is expected, since the subtype shifts were introduced along informative directions within a high-dimensional spiked covariance model. As a result, the observed cluster indices fall well below their corresponding null distributions for all comparisons.

4.2.1 Basal vs other subtypes

Basal tumors are typically the most distinct BRCA subtype, and our simulated data reflect this structure. All Basal comparisons (Basal–LumA, Basal–LumB, and Basal–Her2) yield extremely small p-values ($<$ 1e-4). SigClust easily detects the subtype shifts, with the observed cluster indices falling deep into the left tail of the null distribution.

4.2.2 LumA vs LumB

In real BRCA studies, LumA and LumB often form a continuum and show subtle expression differences. In our simulation, LumB receives only a small mean shift relative to LumA. Despite this weak signal, SigClust still returns a very small p-value (< 1e-4) due to the presence of a consistent shift in the informative dimensions. This demonstrates the sensitivity of SigClust when differences are embedded within a high-dimensional covariance structure.

4.2.3 Her2 vs LumB

Her2 and LumB typically show intermediate separation in real TCGA data. In our simulation, the Her2 subtype receives a moderate shift, while LumB receives a weak one, producing clear separation between the two groups. SigClust again returns a very small p-value (< 1e-4), indicating strong evidence against a single-cluster Gaussian model.

4.2.4 Her2 vs LumA

The Her2–LumA comparison is often borderline in real BRCA analyses. In our simulated setting, however, the Her2 group is shifted more strongly than LumA, producing a well-defined separation. SigClust detects this difference with a very small p-value (< 1e-4), consistent with the simulated effect rather than the subtle separation documented in the original paper.

4.2.5 Summary

Across all subtype pairs, SigClust identifies statistically significant separation:

- The method is highly sensitive to mean shifts in the informative directions.
- Even subtle differences (e.g., LumA vs LumB) are detectable under this high dimensional structure.
- The soft-threshold covariance estimator provides stable and interpretable results.
- While the simulation does not reproduce the exact biological subtleties of BRCA, it mirrors the high-dimensional setting explored in Huang et al. (2015).

These results demonstrate how SigClust behaves when subtype specific differences are embedded within a high-dimensional spiked covariance model.

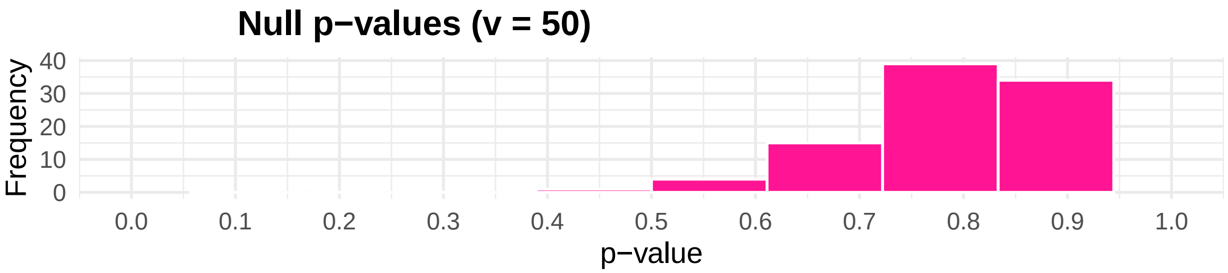
5 Conclusions

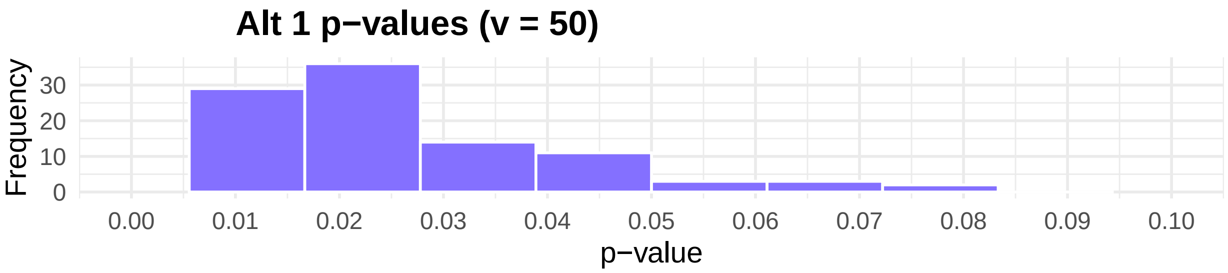
In this project, we explored how SigClust behaves under controlled, high dimensional simulation settings and recreated the paper's results. We observed substantial p-value variability across runs, with SigClust sometimes rejecting and sometimes failing to reject the null. This instability reflects a known phenomenon described in the original SigClust paper: power decreases dramatically for dense low-amplitude signals, especially in the presence of large eigenvalue spikes. Thus, our findings match the theoretical behavior reported by Huang et al. (2015).

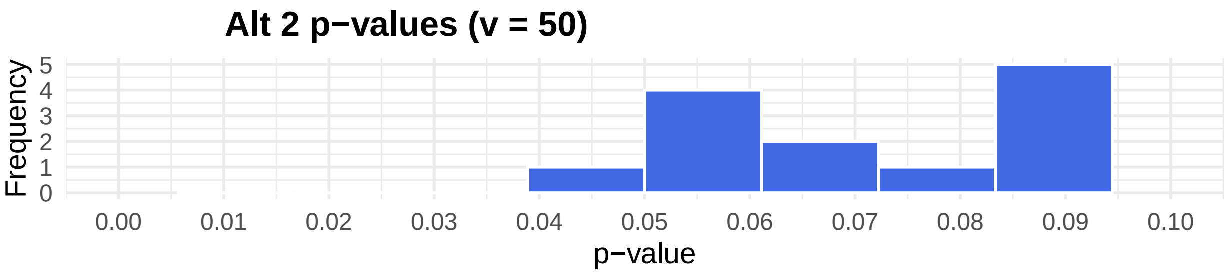
For an implementation perspective, SigClust was computationally intensive. From the Monte Carlo procedure and eigenvalue shrinkage to the repeated k-means clustering, simulations became expensive and tuning `nsim` and the number of variables was mandatory. After repeatedly running each scenario 100 times, we found that keeping `nsim=1000` made the repetition code run for approximately 60 minutes on a large ram computer. While this is not as computationally heavy in comparison to other projects, it can be improved upon, with perhaps parallel computing.

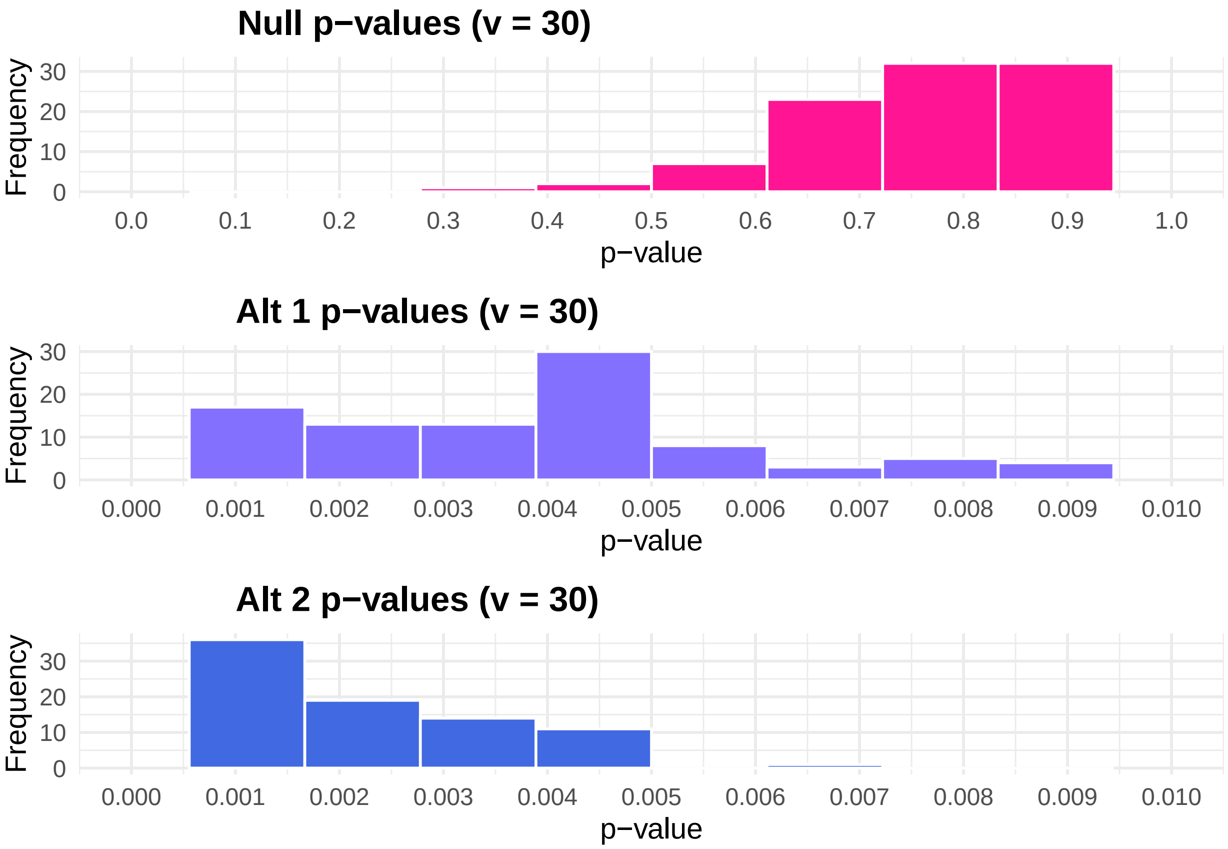
Overall, this project does help clarify how SigClust behaves in both idealized null settings and more challenging alternatives. Future work perhaps could explore large Monte Carlo sample sizes, if SigClust works with more than two clusters, alternative covariance structures, and working on refining how it works with weaker signals.

#Appendix









6 References

Huang, H., Liu, Y., Yuan, M., & Marron, J. S. (2015). *Statistical Significance of Clustering Using SigClust.* Journal of Computational and Graphical Statistics, 24(3), 675–692. https://doi.org/10.1080/10618600.2014.951547

Huang, H., Liu, Y., Yuan, M., & Marron, J. S. (2015). *SigClust: Statistical Significance of Clustering.* R package documentation. https://cran.r-project.org/web/packages/sigclust/sigclust.pdf

7 Team member contribution

Team Member	**Contributions**
Sharvee Joshi	Set up GitHub repository and project structure; implemented SigClust simulations (null, Alt 1, Alt 2); wrote R functions for covariance construction, data generation, and SigClust execution; created and interpreted diagnostic figures; authored the abstract and introduction; contributed to implementation narrative and debugging.
Makena Grigsby	Developed the methodology section, including theory behind SigClust and covariance estimation; summarized statistical framework; conducted BRCA data analysis; reproduced simulation components from the SigClust article; authored methodology and applied data-analysis sections.