Investigating the Use of Mixture Models to Determine KIR Copy Number from DNA Sequence Data

Devita Rahmadani Halim

Supervisors: Prof. Stephen Leslie, Dr. David Squire

Abstract

The killer immunoglobulin-like receptor (KIR) region of the human genome is notable for its high diversity in the numbers of gene copies and alleles it has. The current state of the art method for determining the copy number of KIR genes, and allele of each copy, is Pushing Immunogenetics to the Next Generation (PING). The PING pipeline uses ratios of the number of reads matched to each gene to a reference gene, *KIR3DL3* to determine copy number. This requires setting the boundaries (thresholds) between reported ratios to define copy numbers, a step that is currently performed by eye, manually. This project investigates the use of mixture models to eliminate this manual thresholding step. The expectation-maximization (EM) algorithm was used to fit mixture model on the read ratio data and estimate the parameters of copy number groups. These parameters were then used to determine thresholds between copy number groups. The algorithm developed in this project was able to successfully automate the thresholding step in PING pipeline and available for use.

1 Introduction

The killer immunoglobulin-like receptor (KIR) region located on chromosome 19q13.4 of the human genome encodes for a family of receptors that are expressed on the surface of natural killer (NK) cells and a subset of T-cells [1]. The genomic regions containing KIR genes have been associated with various diseases, such as infections, autoimmunity, cancers, and reproductive abnormalities [2, 3, 4, 5]. The region is characterised by its high diversity in gene copy number and the alleles of those genes. To understand this region better, Pushing Immunogenetics to the Next Generation (PING) pipeline was designed to genotype KIR loci [6]. One major problem in PING pipeline is to conduct a manual thresholding for copy number determination. This manual step is time consuming, prone to human error, and irreproducible. Hence, the ability to automate this manual step of the PING pipline is desirable. In an attempt to eliminate this manual step, we investigated the use of Gaussian mixture model (GMM) by performing an unsupervised soft-clustering of copy number groups using the expectation-maximisation (EM) estimation of the read-depth ratio distribution in the data set. Then, we automatically set thresholds for each KIR genes and reported any outliers for each KIR genes that need to be manually inspected.

2 Background

2.1 Killer-cell immunoglobulin-like receptors (KIRs)

The killer immunoglobulin-like receptors (KIR) region of human chromosome 19q13.4 is a 150-250,000 bp region within the leukocyte receptor complex (LRC) region that contains protein coding genes for KIR family [7, 8]. These genes are mostly expressed on the surface of natural killer (NK) cells that recognize human leukocyte antigen (HLA). KIR genes play significant roles in immune response and have been associated with several human disease including HIV

infection, autoimmune disease, cancer, and reproduction abnormality [2, 3, 4, 5]. One characteristic of KIR is it is highly diverse between individuals in its allelic variation and copy number variation (Figure 1). Specifically, the high diversity of this region is the result of different combinations of copy number variations (CNVs), deletions, duplications, and single-nucleotide variants (SNVs) [9].

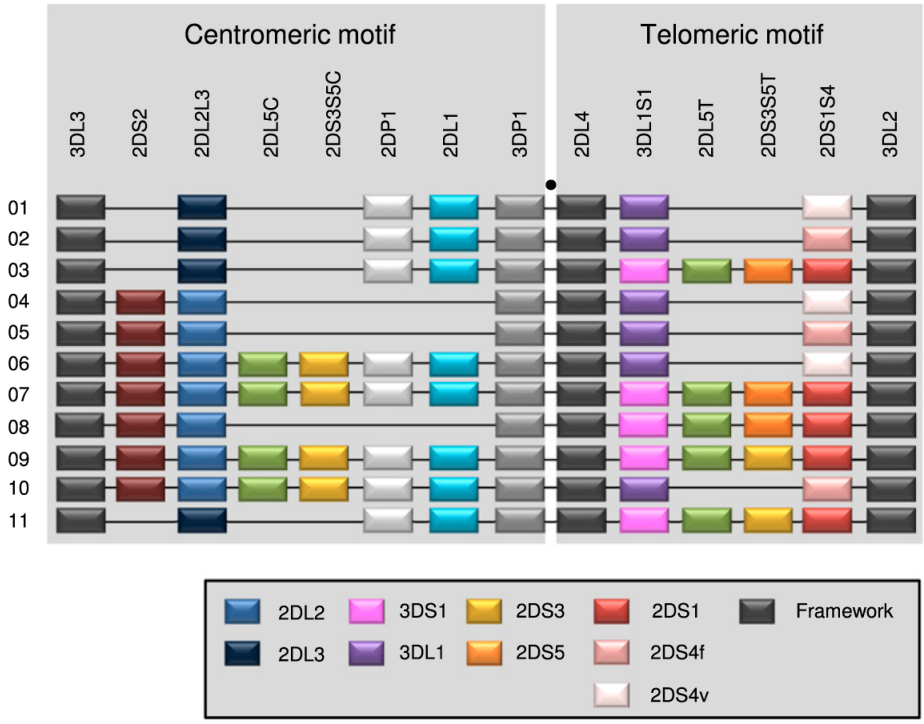


Figure 1: Common KIR haplotypes. Modified from [8].
Here is the eleven common haplotypes of KIR genes. KIR genes are highly diverse in allelic and copy number variation. For example, *2DL2* and *2DL3* are allelic variants of *2DL2L3, 3DL1* and *3DS1* are variants of *3DL1S1*, and *2DS3* and *2DS5* are variants of *2DS3S5*. For copy number variation, *2DL5*, for example, can be duplicated either sequentially (not shown in this figure), or maybe even at the other end of the genome. *KIR3DL3* is one of the framework genes, where every individual have two copies of *KIR3DL3*.

In this project, we focused on the copy number variation of KIR. Copy number variation (CNV) is a term used to describe a molecular event where a region of the genome are repeated with various number of repeats between individuals. Each individual can have various copy number of each KIR genes (0, 1, 2,... copies). Despite its role in immune response, the complex genomic structure of KIR region restrict our ability to have a deep understanding of this region. In this project, we investigated 14 KIR loci (*KIR2DS5, KIR2DL3, KIR2DS3, KIR2DS2, KIR2DL4, KIR3DL1, KIR3DS1, KIR2DL2, KIR3DL2, KIR2DS4, KIR2DL1, KIR2DS1, KIR2DL5*),and two pseudogenes (*KIR3DP1, KIR2DP1*).

2.2 Copy number variation and PING

CNVs is a common phenomenon in human genome and play a role in population diversification and development of diseases [10]. Hence, being able to determine copy number of KIR genes in KIR genotyping is essential. The current state of the art method for KIR sequence interpretation is PING pipeline. It is a high-throughput KIR sequence interpretation workflow that provides allele-level genotypes, copy number, and novel sequence analysis [6]. PING determined copy

number of KIR genes by computing ratio of reads to a reference gene, *KIR3DL3* (Figure 2).

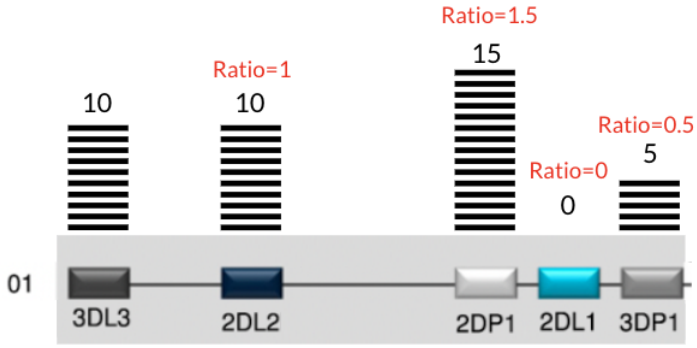


Figure 2: Generating read ratio data. Modified from [8].
This is a modified figure to illustrate how PING generated read ratio data. Since every KIR haplotype has one copy of *KIR3DL3* [11] (two copies in the genotype), copy number of other KIR genes can be determined from the mean number of virtual probe hits on each KIR genes divided by the mean number of *KIR3DL3* probe hits on the *KIR3DL3* gene [12]. In this example, this individual has two copies of *KIR2DL2*, three copies of *KIR2DP1*, zero copy of *KIR2DL1*, and one copy of *KIR3DP1*.

After the read ratio data for each gene were computed, plots for each KIR genes were produced by PING (for example, figure 3) . The read ratio data from PING pipeline for each KIR loci form clusters that correspond with each copy number group (Figure 3). Note that the read ratios are not exact multiples of 0.5 since the genes can have different lengths, and PING may target different numbers of sequences for each gene. However, we still expected the read ratios to be close to 0.5, noting that each KIR genes have similar length to each other. Then, the plots were used to manually set the thresholds between clusters of copy number groups. In figure 3, any read ratio with value below 0.2 will be identified as zero copy, between 0.2 and 0.9 will be identified as one copy, and between 0.9 and 1.7 as two copies.

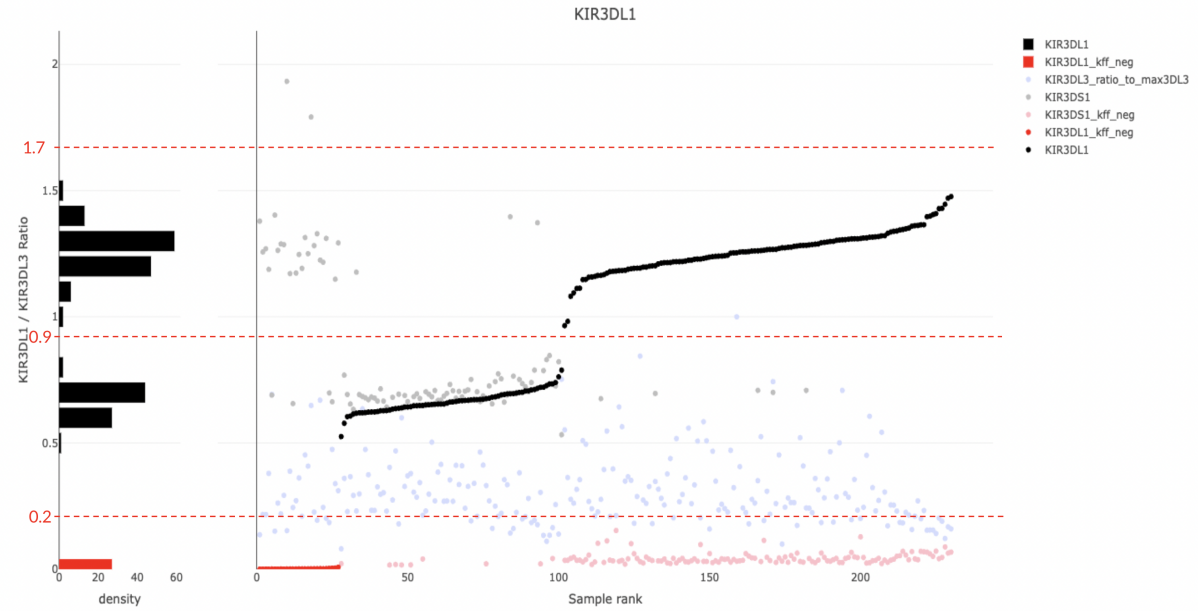


Figure 3: Copy number plot of *KIR3DL1* with manually set thresholds.
This is one example of the plots that were automatically generated by PING. This is the result for 230 samples. The x-axis represents the samples that had been sorted by ratio. The y-axis represents the read ratio of *KIR3DL1* to *KIR3DL3*. The histogram in the left side represents the frequency of the read ratios in the data set. Note that we want to focus on the opaque red and black dots while ignoring the light coloured dots since they do not relate to the problem in this project. The thresholds were set at 0.2, 0.9, and 1.7. In this case, *KIR3DL1* was present in zero, one, or two copies per individual in the data set.

This manual thresholding step, like any other manual data entry, is irreproducible, prone to user error, and time-consuming. Moreover, this cause discontinuous workflow where computers became idle while waiting for user input. Therefore, an effort to eliminate this manual step can provide a continuous workflow of PING pipeline. To do this, a distribution model, called mixture model, can be fitted to the read ratio data (Figure 3).

2.3 Mixture Model

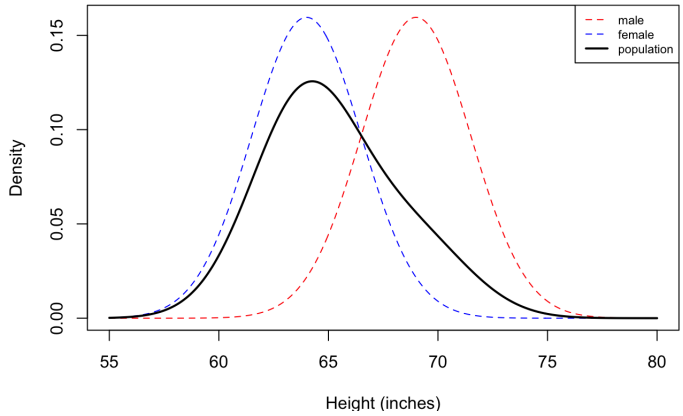


Figure 4: Example of mixture model on human height data. Adapted from [13].
The x-axis represents the height in inches. The y-axis represent the density of the distribution. The black curve is the distribution fitted without considering different gender. The blue and red dotted curves are the distribution fitted when the gender of the individuals were considered.

A mixture model is a collections of k probability distributions or densities where k is the number of component distributions. For example, figure 4 shows how a mixture model was fitted to human height data. The parameters of mixture model we want to focus on are mean, variance, and weights. The weight (ω) of a mixture model is the probability associated with each components of the mixture model. In our example, the weights of the blue distribution would be the probability of a sample belonging to a female. In this project, we focused on Gaussian mixture model (GMM), which is a mixture model consisting of normal distributions with unknown parameters (mean, variance, and weight, denoted as μ, σ^2, ω respectively).

3 Method

3.1 Utilizing Mixture Model to Determine Gene Copy Number

Since the read ratio data from PING pipeline for each KIR loci can be represented as clusters of different copy number groups, the idea of clustering using mixture model is reasonable. Firstly, since the number of reads per individual in the data set is large (10^5), we can apply the Central Limit Theorem (CLT), where the distribution of the sample means of the reads will be approximately normally distributed. Moreover, most of the copy number group clusters are normally distributed from inspection. Hence, in this case assuming each copy number groups to be normally distributed is appropriate. Moreover, the read ratio data is comprised of multiple clusters of copy number groups. Thus, representing the read ratio data with GMM is appropriate.

3.2 Workflow

To automatically compute the thresholds of read ratio data for each KIR genes, we developed an algorithm using Python programming language with the workflow shown in figure 5. Firstly, we determined the number of initial components (representing number of copy number groups) by utilizing an existing python package (Section 3.3.1). Then, we performed an unsupervised soft-clustering using expectation-maximisation (EM) algorithm (Section 3.4). If the number of components did not satisfy certain criteria, we adjusted the number of components (Section 3.3.2). Finally, we determined the thresholds using equal likelihood method (Section 3.7.2) and output the plots for each gene (6).

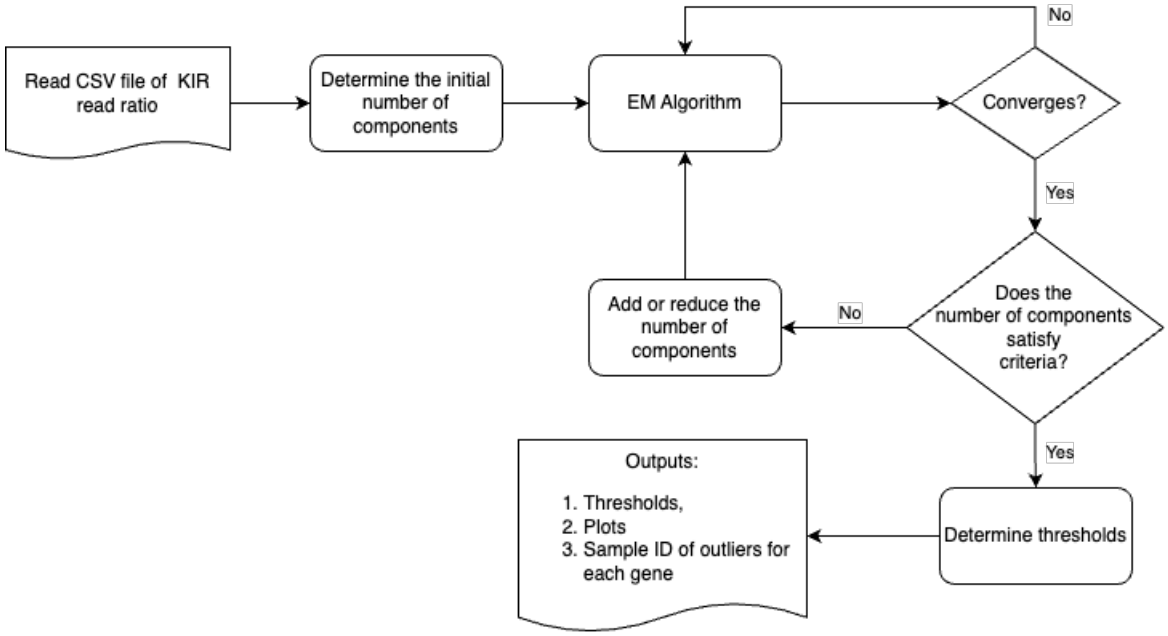


Figure 5: Workflow of the algorithm

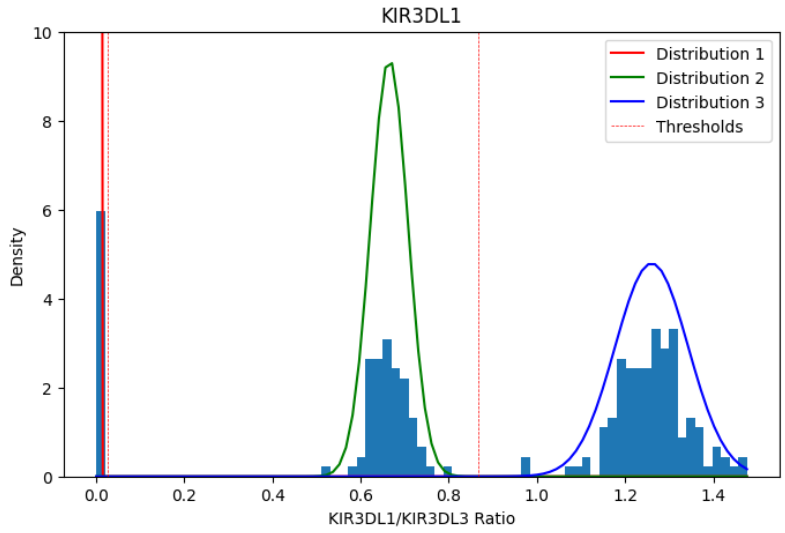


Figure 6: Final plot achieved using the algorithm for KIR3DL1. This is one of the final plot produced by the algorithm with the workflow specified above. The x-axis represents the read ratio of KIR3DL1 to KIR3DL3. The y-axis represents the density of the distributions. The coloured curves are the estimated mixture model generated. The red dotted lines are the thresholds set using the equal likelihood method.

3.3 Determining the number of components

3.3.1 Initial number of components

The initial number of components was obtained using a python package (sklearn.mixture. GaussianMixture)[14]. Firstly, since most KIR genes were found to have less than 3 copies and did not exceed 6 copies [15], the data was fitted with a range of number of components from one to six ($K \in [1, 6]$). Then, the Bayesian Information Criterion (BIC) scores for each model were computed. Finally, the model with the lowest BIC score was chosen to be the initial

number of components.

3.3.2 Adjusting the number of components

For some KIR genes, using the initial number of components determined earlier is insufficient due to over-fitting or under-fitting distribution (Figure 13a,13b). Hence, we defined two criteria to adjust the initial number of component based on the prior knowledge of how the data was generated. Firstly, to avoid under-fitting distribution, if 3% of the data points have low likelihood ($> 2.968\ \sigma$ away from the mean (99.7% confidence interval)) of being generated by any Gaussian, number of components was added by one. The minimum likelihood for each Gaussian was calculated using the equation below, which is the formula for normal distribution density, and substituting the x-value with the point of $2.968\ \sigma$ away from the mean:

$$p(x|\mu_k, \sigma_k^2) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{(x-\mu_k)^2}{2\sigma_k^2} \right]; x = (2.967738\sigma_k) + \mu_k$$

Secondly, to avoid over-fitting distribution, if the distance of the mean of two neighbouring Gaussians are less than 2.24σ away from the mean of both Gaussian (97.5% CI), the number of components was reduced by one. This second condition ensure the distributions to be well-separated. Since the data is in form of ratio read, we would expect the clusters of copy number groups to be well separated. Hence, the minimum distance criterion is appropriate.

3.4 The EM Algorithm

Given the set of data $X = \{x_1, x_2, ..., x_N\}$ that corresponds to ratio of reads of specific KIR gene to *KIR3DL3*, EM algorithm was applied to provide estimate of parameters that maximise the likelihood function. In the expectation step (E-step), the expectation of likelihood of each observation x_i given the parameters is computed using equation(1). Then, using Bayes Theorem, we computed posterior probability of the k^{th} Gaussian to explain the data set using equation(2). That is the likelihood of given x_i being generated by the k^{th} Gaussian.

$$p(x|k) = p(x|\mu_k, \sigma_k^2) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left[-\frac{(x-\mu_k)^2}{2\sigma_k^2} \right] \tag{1}$$

$$p(k|x) = \frac{\omega_k p(x|k)}{\sum_{k=1}^{K} \omega_k p(x|k)} \tag{2}$$

Then, the maximization step (M-Step) consists of maximizing over the parameters the expectation computed in the E-step. In this step, the parameters were re-estimated using equations(3, 4, 5).

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^{N} p(k|x_i) x_i}{\sum_{i=1}^{N} p(k|x_i)} \tag{3}$$

$$\sigma_k^{2(t+1)} = \frac{\sum_{i=1}^{N} p(k|x_i) \left(x_i - \mu_k^{(t+1)} \right)^2}{\sum_{i=1}^{N} p(k|x_i)} \tag{4}$$

$$\omega_k^{(t+1)} = \frac{\sum_{i=1}^{N} p(k|x_i)}{N} \tag{5}$$

The E and M steps were repeated until the stopping criteria were met.

3.5 Stopping Criteria

The EM algorithm were run until the parameters converge, that is the difference between successive iterations met certain tolerance. The stopping criteria used were the relative error criteria version of the recommendation by [16]. From inspection, the recommendation stopping criteria by [16] is suitable for our data since the final parameters computed had fitted the data according to our expectation. The stopping criteria are,

$$\frac{|\mu_k^t - \mu_k^{(t+1)}|}{|\mu_k^t|} < 10^{-6}; \forall k, k = 1, ..., K \tag{6}$$

$$\frac{|(\sigma_k^2)^t - (\sigma_k^2)^{(t+1)}|}{|(\sigma_k^2)^t|} < 10^{-4}; \forall k, k = 1, ..., K \tag{7}$$

$$\frac{|\omega_k^t - \omega_k^{(t+1)}|}{|\omega_k^t|} < 10^{-8}; \forall k, k = 1, ..., K \tag{8}$$

Once the stopping condition for all parameters are met, EM algorithm was terminated.

3.6 Parameters

One disadvantage of EM algorithm is that it only guarantees convergence to local optima. Hence, the choice of initial parameters is crucial since inappropriate choice of initial parameters can result in inaccurate estimation of the mixture model parameters. Since we expected the clusters of copy number groups to be well separated due to the reason previously mentioned (Section 3.3.2), the initial means were set by taking evenly spaced values over the the data set interval $[a, b]$ (Equation 9), where a is the least element of the data set and b is greatest element of the data set($\mu_1 = a = \inf X; \mu_2 = b = \sup X$). This choice of initial means ensure the distributions to have adequate distances between each other. Moreover, the initial variances need to be fairly small to avoid one distribution taking responsibilities of more than one copy number groups (under-fitting). Thus, the initial variances were set to be sufficiently small (Equation 10). Furthermore, we assume a general case where each copy number groups have equal weights. Thus, the initial weights were set to be equal for each component (Equation 11).

$$\mu_k^0 = \{a, \Big(\mu_i + \frac{\mu_i - \mu_{i+2}}{K-1}\Big), ..., b\}; \forall i, i = 1, ..., (b-2) \tag{9}$$

$$\sigma_{k_0}^2 = 0.005; \forall k, k = 1, ..., K \tag{10}$$

$$\omega_{k_0} = \Big[\frac{1}{K}\Big]; \forall k = 1, ..., K \tag{11}$$

Since the read ratio data was generated from the number of virtual probe hits on each KIR genes divided by number of hits on *KIR3DL3* gene [12], we expect a relatively small variances for each Gaussian. That is, no clusters should exceed the variance of some value. To get this maximum variance value, we tried several different values and land on variance of 0.01. To clarify this, we firstly set a constraint that the maximum value for the variance is 0.01. Then, it was found that the variances were ranging from 7.55967×10^-8 to 0.0096. Since no variance was found to be 0.01, this implies that 0.01 is the appropriate value for maximum variance constraint. This clarify that the maximum variance constraint of 0.01 will avoid under-fitting while also not being overly restrictive. Any further improvement would involve improving the maximum variance constraint by defining a more principled way instead of using fixed value.

3.7 Determining Thresholds

To differentiate copy number groups, thresholds between the copy number groups were computed. Two different methods in determining these thresholds were considered, which are intersection method and equal likelihood method. These two methods were compared to each other (Section 4.3) and we eventually decided to use the equal likelihood method.

3.7.1 Intersection method

This method put the threshold where the probability density function (PDF) of two Gaussian distributions intersect. That is by choosing the corresponding x-value of the minimum area under the "two-tails". Firstly, two set of Z-scores were computed for the two adjacent Gaussians. For the right Gaussian, the area left of all the Z-scores were computed by calculating the cumulative distribution function (CDF)(12). For the left Gaussian, the area right of all the Z-scores were computed using equation(13). Then, the threshold is the x-value corresponding to the minimum area under these two Gaussians, as shown in equation(14).

$$P(Z_1 \leq x) = \Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-\frac{u^2}{2}) du \tag{12}$$

$$P(Z_1 > x) = 1 - P(Z_1 \leq x) \tag{13}$$

$$\min P(Z_1 > x) + P(Z_2 \leq x) \tag{14}$$

3.7.2 Equal likelihood method

The thresholds can also be set at the point where the likelihood of being generated by one Gaussian is equal to the likelihood of being generated by the adjacent Gaussian. This can be done by solving equation(15) for x. Expanding and rearranging this equation gave a quadratic equation(16), which gave two possible solutions for each thresholds. The solution chosen was the one with the value within the range of the two means (μ_1, μ_2) of the Gaussians.

$$\omega_1 f(x|\mu_1, \sigma_1^2) = \omega_2 f(x|\mu_2, \sigma_2^2) \tag{15}$$

$$(\sigma_2^2 - \sigma_1^2)x^2 + 2(\sigma_1^2\mu_2 - \sigma_2^2\mu_1)x + \sigma_2^2\mu_1^2 - \sigma_1^2\mu_2^2 - 2\sigma_1^2\sigma_2^2 \log\left[\frac{\omega_1|\sigma_2|}{\omega_2|\sigma_1|}\right] = 0 \tag{16}$$

3.8 Unavailability of data for some copy number groups

Some KIR genes in the data set did not have samples corresponding to certain copy number groups (Figure 7).

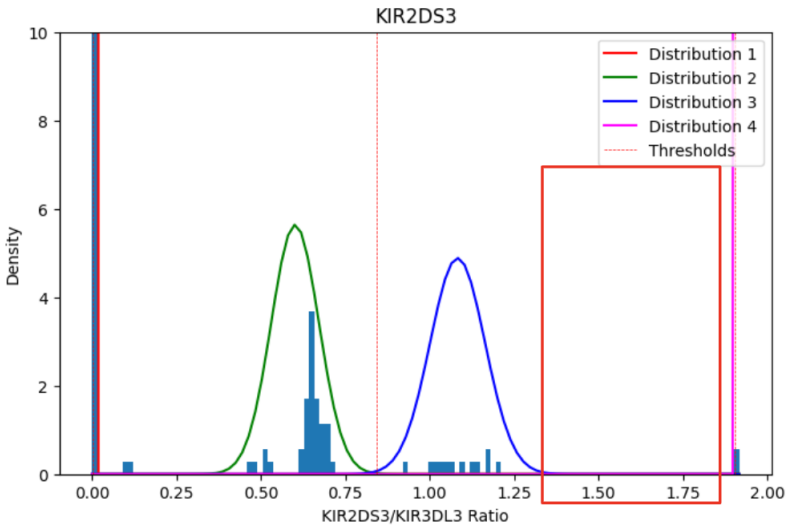


Figure 7: Estimated distribution of *KIR2DS3*.
The x-axis represents the *KIR2DS3*/*KIR3DL3* read ratio data. The y-axis represent the density. The red, green, blue, magenta curves represent the distribution of the zero, one, two, four copy number groups, respectively. The red box show the position of possible missing distribution corresponding to three copies. The red dotted lines are the thresholds with values $0.0028, 0.8422,$ and 1.9110.

.

For these genes, we defined conditions to add extra thresholds since not doing so will result in inaccuracy in the copy number we assigned. Two cases were identified in this particular data set. First, genes that had no data corresponding to zero copy. Second, genes that had no data for a copy number that is in between two copy number groups that had available data.

3.8.1 No data for zero copy

For the first case, we defined a condition where if the smallest mean in the mixture model is bigger than 0.2, we added a threshold at 0.07277, which was the average of 0-1 copy number threshold from other KIR genes. Moreover, if there was a distance between adjacent Gaussians that are more than 1.5-fold of the smallest distance between other Gaussians pair, this implies that the data does not have zero and one copy number. Thus, a threshold was added 3.89 σ away from the mean of the two copy number group.

3.8.2 No data for a middle copy number group

For the second case, we defined a condition if the distance between two adjacent Gaussians is more than 1.5-fold of the smallest distance between other Gaussians pair, two extra thresholds were added. The first threshold was set 3.89 σ away from the mean of the left Gaussian (99.99% CI). Another threshold was set 3.89 σ away from the right Gaussian. This value was chosen to ensure that the thresholds were not set at points within the existing Gaussians.

4 Results

4.1 Running the algorithm on simulated data

First, we tested the algorithm developed on simulated data of several different number of components ($K = 2, 3, 4$). The simulated data were set to have well-separated mean, different

variances, equal weights for two and three components and different weights for the four components. When the algorithm was tested on simulated data with overlapping distribution, the result was inaccurate with large percentage error, especially for variance and weight (Figure 8,table 1). However, this would not be a problem for the data set we were working with since all the KIR genes read ratio data have well-separated clusters.

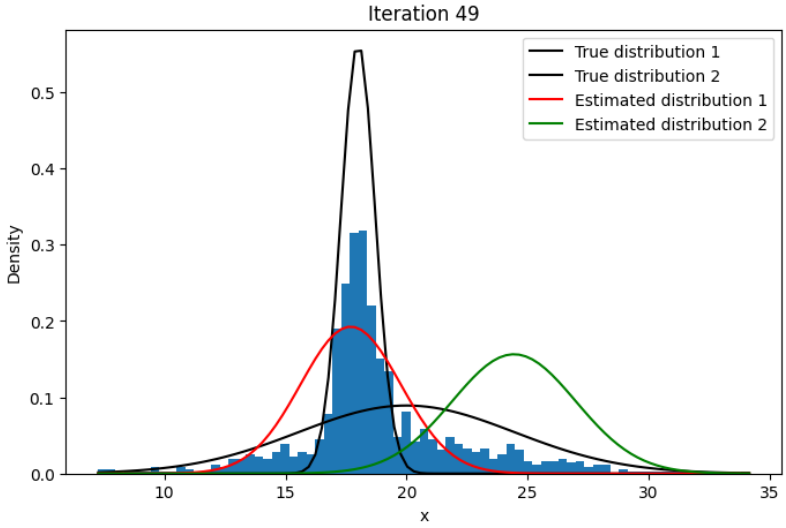
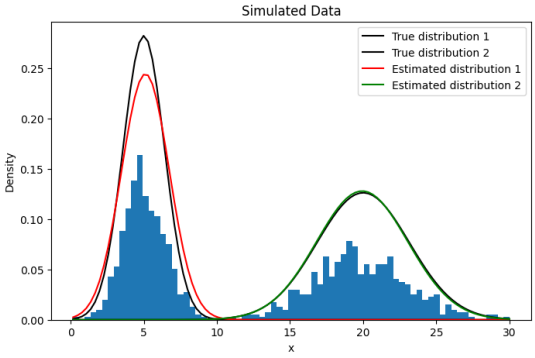


Figure 8: Example of estimating overlapping distribution with EM.
This is the example where we try using EM to estimate overlapping distribution.The x-axis represents the simulated data points value. The y-axis represents the density of the distribution. The black curves represent the true distribution. The estimated distributions are represented by the red and green coloured curves. The distribution curves were obtained using the formula for Gaussian density (Equation 1).

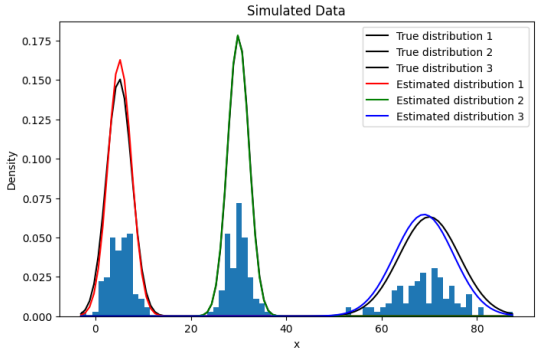
Table 1: Comparison of the estimated parameters to true parameters of simulated data with two components that are overlapping.

Distribution	Mean			Variance			Weight		
	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error
Distribution 1	20	25.3272	26.636%	20	7.2480	63.76%	0.5	0.1161	76.78%
Distribution 2	18	18.0313	0.1739%	0.5	5.6207	1024.14%	0.5	0.8839	76.78%

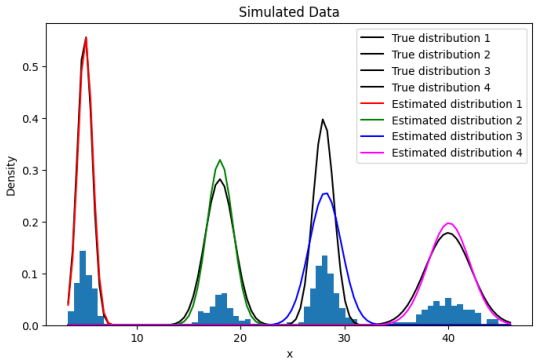
The figures representing the algorithm performance on simulated data were generated. Tables containing the true parameters values, estimated parameters value, and percentage errors were also generated.



(a) Two components



(b) Three components



(c) Four components

Figure 9: Results of running the algorithm on several simulated data with different number of components.
The x-axis represents the simulated data points value. The y-axis represents the density of the distribution. The black curves represent the true distribution. The estimated distributions are represented by the red, green, blue, magenta coloured curves. The distribution curves were obtained using the formula for Gaussian density (Equation 1).

Table 2: Comparison of the estimated parameters to true parameters of simulated data with two components

Distribution	Mean			Variance			Weight		
	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error
Distribution 1	20	19.8229	0.8855%	10	10.4667	4.667%	0.5	0.5003	0.06%
Distribution 2	5	5.0205	0.41%	2	2.0530	2.65%	0.5	0.4997	0.06%

Table 3: Comparison of the estimated parameters to true parameters of simulated data with three components

Distribution	Mean			Variance			Weight		
	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error
Distribution 1	5	5.1096	2.192%	7	6.0011	14.27%	0.3333	0.3333	0%
Distribution 2	30	29.9619	0.127%	5	4.9926	0.148%	0.3333	0.3333	0%
Distribution 3	70	68.8952	1.5783%	40	38.1189	4.7028%	0.3333	0.3333	0%

Table 4: Comparison of the estimated parameters to true parameters of simulated data with three components

Distribution	Mean			Variance			Weight		
	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error	True value	Estimated value	Percentage error
Distribution 1	5	5.0526	1.052%	0.5	0.5124	2.48%	0.25	0.25	0%
Distribution 2	18	18.0799	0.4439%	2	1.7134	14.33%	0.16667	0.16667	0%
Distribution 3	28	28.1126	0.4021%	1	1.1452	14.52%	0.33333	0.3333	0%
Distribution 4	40	39.9693	0.0768%	5	4.5422	9.156%	0.25	0.25	0%

It was found that the algorithm estimated the parameters with percentage error of $0.89595\%, 8.3655\%, 0.015\%$ on average, for the mean, variance, and weight respectively. This error percentages were aligned with the tolerance set for the stopping criteria (Section 3.5).

4.2 Running the algorithm on read ratio data of KIR genes

We ran the algorithm on all KIR genes data from PING. For most of the KIR genes, the algorithm estimated the distribution without any sign of under-fitting of over-fitting. Here the figures and tables for *KIR3DL1* are shown to represent the genes with well-behaved data (plots for other KIR genes are available on appendix A.1).

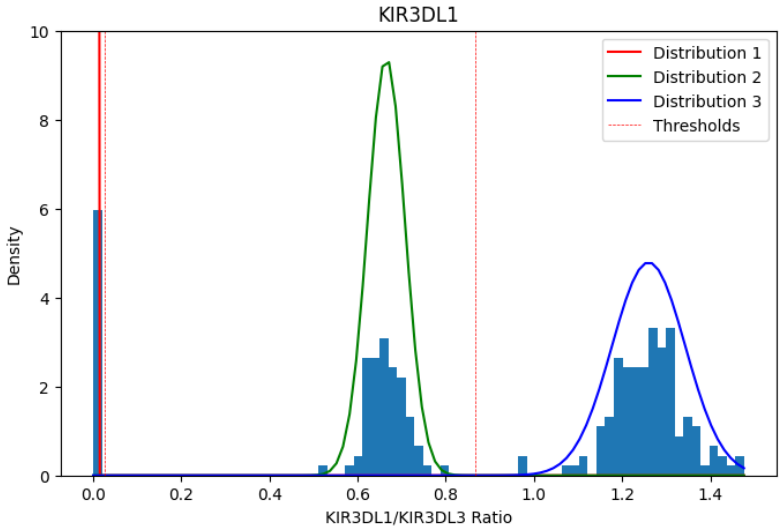


Figure 10: Estimated distribution of *KIR3DL1*.
The x-axis represents the *KIR3DL1/KIR3DL3* read ratio data. The y-axis represents the density of the distribution. The red curve (labelled as distribution 1) represents the distribution of the zero copy number. The green curve (distribution 2) represents the distribution of the one copy number group. The blue curve (distribution 3) represent the distribution of the two copy number. The red dotted lines are the thresholds with values 0.0265 and 0.8667.

Table 5: Estimated parameters of *KIR3DL1*.

Copy number group	Mean	Variance	Weight
0 copy	0.0016	$2.6894 \times 10^{-}6$	0.1174
1 copy	0.6649	0.0018	0.3217
2 copies	1.2601	0.0069	0.5609

Some of the KIR genes are more complicated. For example, KIR3DP1, KIR2DL4, KIR2DS3,

and KIR3DL2 did not have any samples that corresponds to several copy numbers. Using the method described at section 3.8, we computed extra thresholds for these genes to account for the missing distribution. To demonstrate this, the figure and table for KIR2DS3 are shown below.

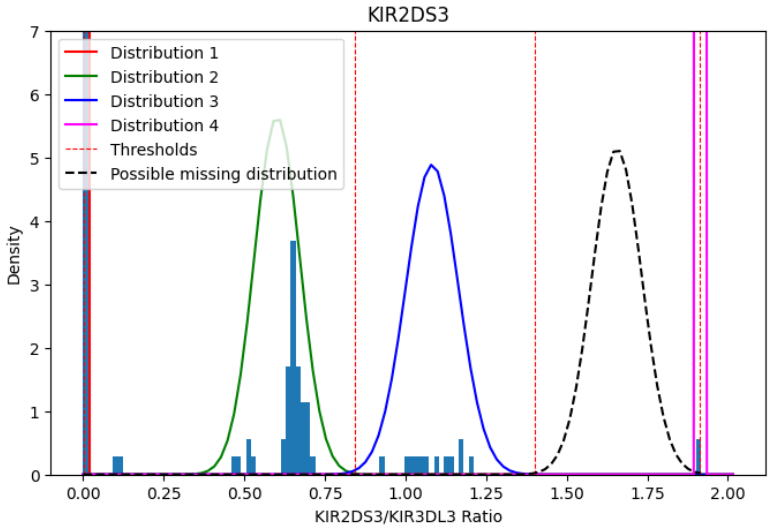


Figure 11: Estimated distribution of *KIR2DS3*.
The x-axis represents the *KIR2DS3*/*KIR3DL3* read ratio data. The y-axis represent the density. The red, green, blue, magenta curves represent the distribution of the zero, one, two, four copy number groups, respectively. The black dotted curve represents the possible missing distribution corresponding to three copies. The red dotted lines are the thresholds with values $0.0028, 0.8422, 1.4002,$ and 1.9113.

Table 6: Estimated parameters of *KIR2DS3*.

Copy number group	Mean	Variance	Weight
0 copy	0.0002	$7.5597 \times 10^{-}8$	0.7478
1 copy	0.6016	0.0050	0.1913
2 copies	1.0827	0.0067	0.0522
3 copies (no data)	1.6558	0.0060	0
4 copies	1.9154	$1.0655 \times 10^{-}6$	0.0087

4.3 Thresholds obtained for KIR genes

Thresholds for all the KIR genes were computed using two different methods mentioned at section 3.7. Thresholds resulting from the two methods are shown below together with the absolute difference between the two methods for comparison.

Table 7: Thresholds for all KIR genes using the two methods for 0-1 and 1-2 thresholds

KIR genes	0-1			1-2		
	Intersection method	Equal likelihood method	Absolute difference	Intersection method	Equal likelihood method	Absolute difference
KIR3DP1	0.0728	0.0728	0.0000	0.3438	0.3394	0.0044
KIR2DS5	0.0699	0.0952	0.0253	0.6708	0.6804	0.0096
KIR2DL3	0.0556	0.1062	0.0505	0.8904	0.8943	0.0039
KIR2DP1	0.0568	0.1091	0.0523	0.9226	0.9180	0.0046
KIR2DS3	0.0194	0.0028	0.0166	0.8324	0.8422	0.0098
KIR2DS2	0.0249	0.0252	0.0004	0.8698	0.8775	0.0077
KIR2DL4	0.0728	0.0728	0.0000	0.6098	0.5996	0.0102
KIR3DL1	0.0298	0.0265	0.0033	0.8650	0.8667	0.0017
KIR3DS1	0.2298	0.2564	0.0266	0.9265	0.9345	0.0080
KIR2DL2	0.0099	0.0167	0.0068	0.6409	0.6455	0.0047
KIR3DL2	0.0728	0.0728	0.0000	0.8149	0.8149	0.0000
KIR2DS4	0.0163	0.0088	0.0076	0.8331	0.8266	0.0065
KIR2DL1	0.1083	0.0964	0.0119	0.7824	0.7693	0.0130
KIR2DS1	0.0500	0.0533	0.0034	0.3442	0.3474	0.0032
KIR2DL5	0.0744	0.0767	0.0023	0.6200	0.6288	0.0088
Average difference	0.0138			0.0064		

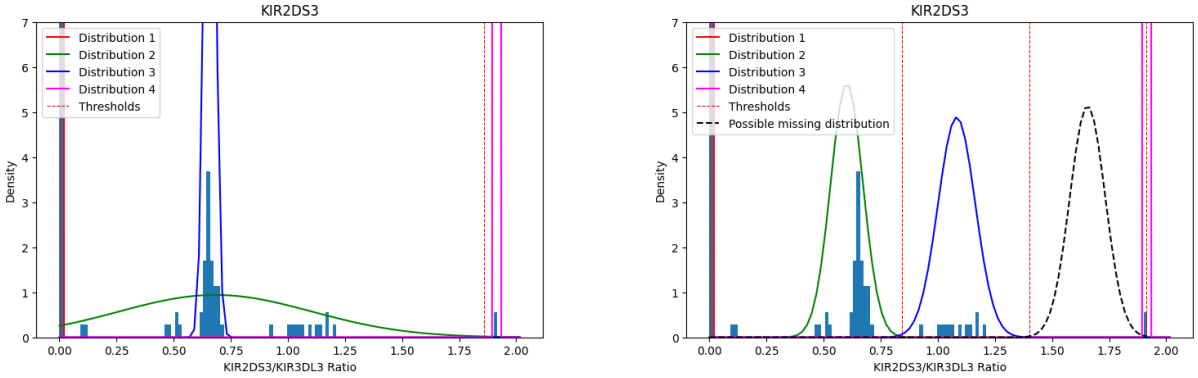
Table 8: Thresholds for all KIR genes using the two methods for 2-3 and 3-4 thresholds

KIR genes	2-3			3-4		
	Intersection method	Equal likelihood method	Absolute difference	Intersection method	Equal likelihood method	Absolute difference
KIR3DP1	0.6139	0.6181	0.0042	-	-	-
KIR2DS5	1.0901	1.1068	0.0167	-	-	-
KIR2DL3	-	-	-	-	-	-
KIR2DP1	-	-	-	-	-	-
KIR2DS3	1.4002	1.4002	0.0000	1.9113	1.9113	0.0000
KIR2DS2	-	-	-	-	-	-
KIR2DL4	1.0500	1.0637	0.0137	-	-	-
KIR3DL1	-	-	-	-	-	-
KIR3DS1	1.5651	1.5903	0.0253	-	-	-
KIR2DL2	-	-	-	-	-	-
KIR3DL2	1.0146	0.9990	0.0156	-	-	-
KIR2DS4	-	-	-	-	-	-
KIR2DL1	-	-	-	-	-	-
KIR2DS1	-	-	-	-	-	-
KIR2DL5	0.9072	0.9080	0.0008	-	-	-
Average difference	0.0109			0.0000		

From table 7 and 8, the overall trend show little to no difference of the thresholds obtained using the two different methods. The average difference was obtained to be 0.0078. The equal likelihood method was preferred since it is taking into account the weights of the Gaussians while intersection method did not.

4.4 Comparison between Unconstrained and Constrained Parameters

To demonstrate the importance of setting maximum variance constraint, two figures for KIR2DS3 are shown below.



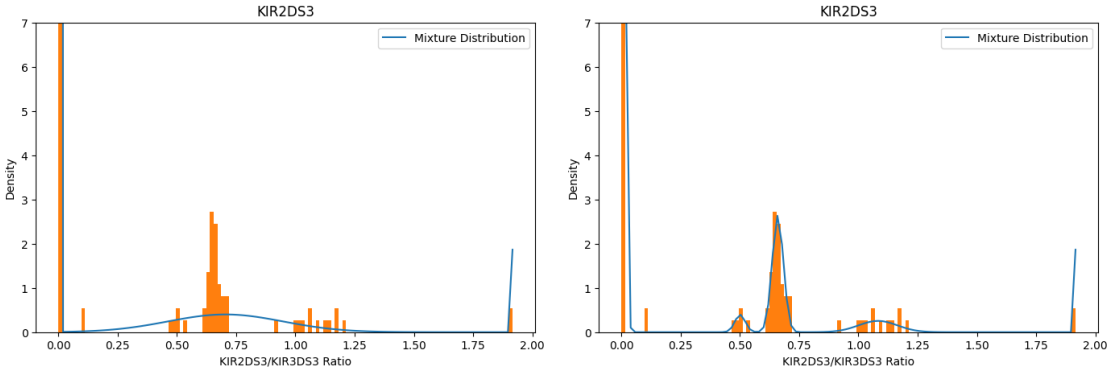
(a) Unconstrained **(b)** Constrained

Figure 12: Comparison of KIR2DS3 estimated distribution in the presence and absence of variance constraint.
The x-axis represents the *KIR2DS3/KIR3DL3* read ratio data. The y-axis represent the density. The red, green, blue, magenta curves represent the distribution of the zero, one, two, four copy number groups, respectively. The black dotted curve represents the possible missing distribution corresponding to three copies.

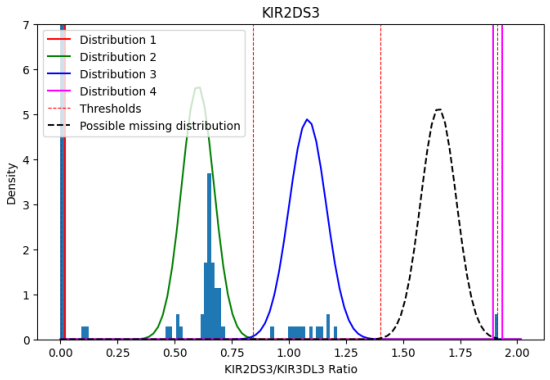
In the absence of variance constraint (Figure 12a), the green distribution take responsibility of more than one cluster. This shows that the presence of variance constraint is essential to avoid under-fitting.

4.5 Comparison with an existing python package

We compared the use of an existing Python package called sklearn.mixture.GaussianMixture [14] to our algorithm to fit all the KIR genes. It was found that sklearn tends to overfit or underfit (depending on whether AIC or BIC were used) the data in a similar way our algorithm did without variance constraint and adjustment of the number of component. Here, figures for *KIR2DS3* gene are shown to demonstrate the differences.



(a) Model with lowest BIC **(b)** Model with lowest AIC



(c) Our algorithm

Figure 13: Comparison of the algorithm to an existing python package on KIR2DS3. (a) Using BIC, sklearn underfit the data where one distributions was responsible for copy number one and two. (b) Using AIC, the data was overfit where two distribution is responsible for copy number one. (c) Using our algorithm, the data was fitted according to our expectation.

5 Discussion

5.1 The use of prior knowledge

For the KIR read ratio data, using prior knowledge of how the data was generated was shown to be essential. We were able to set the initial parameters to fit the data better and faster (reduced number of iterations needed). Moreover, we are also able to constraint the variance and adjusting the number of components to successfully avoid over-fitting and under-fitting. This is also the reason why an existing package for general cases of GMM sometimes fails (Figure 13). However, the downside is that the algorithm is highly specific since it was developed around this specific data set.

5.2 Uncertainty on certain regions

Since the variance of the zero copy number were often found to be relatively small, the thresholds for 0-1 copy number were more tight to the left than if they were manually set. This raise a question on how we should determined the copy number of samples that falls on the region where the likelihood of being generated by any copy number group distribution is low. Since we cannot reliably account for these samples automatically, an output containing sample ID with low likelihood of belonging to any copy number group was generated for manual inspection (Appendix A.3). After we evaluated the outliers, we can found several instances where the same

sample was reported as outliers for multiple genes suggesting there was possible errors on how the data for those samples were generated.

5.3 Future Improvements

There are several improvements that can be made on the algorithm. Firstly, it would be better if we can perform a model selection criterion after fitting models with different number of components. As for now, BIC was only used to determine the initial number of components and the adjustment was done without the use of any model selection criterion. Moreover, most of the conditions in this algorithm were set to fit one specific data set we were working with. The next step would be to try running the algorithm on different data sets and observe the results. Furthermore, the low variance found on most genes for certain copy number indicates the needs to set a minimum variance constraint. Any future work on this would try to implement this to avoid setting thresholds that are too restrictive. It would also be great if we can modify the maximum variance constraint instead of using a fixed value. Finally, we can also try different algorithm (e.g. Bayesian Gaussian Mixture Model) to observe any difference in the results compared to using EM.

6 Conclusion

To conclude, we developed an algorithm that successfully automate the manual thresholding step in PING. Firstly, we were able to apply the EM algorithm on KIR read ratio data. Then, using prior knowledge on how the data was generated, we were able to modify the generic EM algorithm by introducing maximum variance constraint and defining criteria to adjust the number of components for each KIR genes. Moreover, we were able to account for possible missing copy number group and reported the samples with low likelihood of being in any copy number group for manual inspection. Finally, we successfully computed thresholds for all KIR genes in the data set automatically.

Acknowledgements

I would like to thank both Stephen Leslie and David Squire for being the best supervisors, guiding me throughout this project, and caring about my well being. I would also like to thank Minguel Gobardja, and my brother Devano Halim for all the support and constructive comments. Also, thanks to Leoni Angela and Roseline for their support and help while I was learning to program for the first time for this project. Thanks to Alex Andrianopoulos for giving me the chance to take this subject and for coordinating the subject for the genetic stream. This project would not be possible without all of you.

References

[1] L. Moretta and A. Moretta, "Killer immunoglobulin-like receptors," *Current opinion in immunology*, vol. 16, no. 5, pp. 626–633, 2004.

[2] M. Carrington and P. Norman, "The kir gene cluster," 2003.

[3] F. Colucci, "The role of kir and hla interactions in pregnancy complications," *Immunogenetics*, vol. 69, no. 8, pp. 557–565, 2017.

[4] S. Chaisri, N. Pabalan, S. Tabunhan, P. Tharabenjasin, N. Sankuntaw, and C. Leelayuwat, "Effects of the killer immunoglobulin–like receptor (kir) polymorphisms on hiv acquisition: a meta-analysis," *PloS one*, vol. 14, no. 12, p. e0225151, 2019.

[5] M. Carrington and M. Martin, "The impact of variation at the kir gene cluster on human disease," *Immunobiology of Natural Killer Cell Receptors*, pp. 225–257, 2006.

[6] W. M. Marin, R. Dandekar, D. G. Augusto, T. Yusufali, B. Heyn, J. Hofmann, V. Lange, J. Sauter, P. J. Norman, and J. A. Hollenbach, "High-throughput interpretation of killer-cell immunoglobulin-like receptor short-read sequencing data with ping," *PLoS computational biology*, vol. 17, no. 8, p. e1008904, 2021.

[7] D. Roe, C. Vierra-Green, C.-W. Pyo, K. Eng, R. Hall, R. Kuang, S. Spellman, S. Ranade, D. Geraghty, and M. Maiers, "Revealing complete complex kir haplotypes phased by long-read sequencing technology," *Genes & Immunity*, vol. 18, no. 3, pp. 127–134, 2017.

[8] W. Jiang, C. Johnson, J. Jayaraman, N. Simecek, J. Noble, M. F. Moffatt, W. O. Cookson, J. Trowsdale, and J. A. Traherne, "Copy number variation leads to considerable diversity for b but not a haplotypes of the human kir genes encoding nk cell receptors," *Genome research*, vol. 22, no. 10, pp. 1845–1854, 2012.

[9] S. Sakaue, K. Hosomichi, J. Hirata, H. Nakaoka, K. Yamazaki, M. Yawata, N. Yawata, T. Naito, J. Umeno, T. Kawaguchi, *et al.*, "Decoding the diversity of killer immunoglobulin-like receptors by deep sequencing and a high-resolution imputation method," *Cell Genomics*, vol. 2, no. 3, p. 100101, 2022.

[10] O. Pös, J. Radvanszky, G. Buglyó, Z. Pös, D. Rusnakova, B. Nagy, and T. Szemes, "Dna copy number variation: Main characteristics, evolutionary significance, and pathological aspects," *Biomedical Journal*, vol. 44, no. 5, pp. 548–559, 2021.

[11] C.-W. Pyo, L. A. Guethlein, Q. Vu, R. Wang, L. Abi-Rached, P. J. Norman, S. G. Marsh, J. S. Miller, P. Parham, and D. E. Geraghty, "Different patterns of evolution in the centromeric and telomeric regions of group a and b haplotypes of the human killer cell ig-like receptor locus," *PloS one*, vol. 5, no. 12, p. e15115, 2010.

[12] P. J. Norman, J. A. Hollenbach, N. Nemat-Gorgani, W. M. Marin, S. J. Norberg, E. Ashouri, J. Jayaraman, E. E. Wroblewski, J. Trowsdale, R. Rajalingam, *et al.*, "Defining kir and hla class i genotypes at highest resolution via high-throughput sequencing," *The American Journal of Human Genetics*, vol. 99, no. 2, pp. 375–391, 2016.

[13] M. Bonakdarpour, "Introduction to mixture models." `https://github.com/stephens999/fiveMinuteStats.git`, 2016.

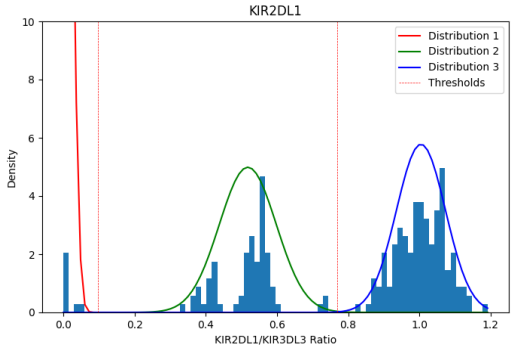
[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[15] R. M. Pyke, R. Genolet, A. Harari, G. Coukos, D. Gfeller, and H. Carter, "Computational kir copy number discovery reveals interaction between inhibitory receptor burden and survival," in *BIOCOMPUTING 2019: Proceedings of the Pacific Symposium*, pp. 148–159, World Scientific, 2018.

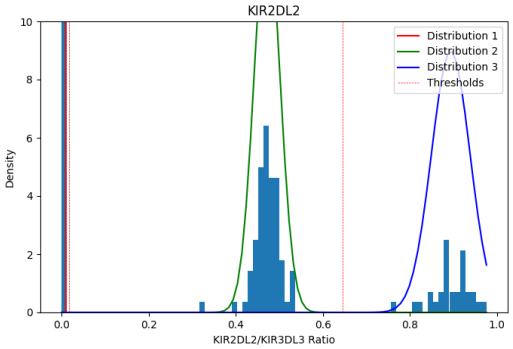
[16] R. Abbi, E. El-Darzi, C. Vasilakis, and P. Millard, "Analysis of stopping criteria for the em algorithm in the context of patient grouping according to length of stay," vol. 1, pp. 3–9, 2008.

A Supplementary data

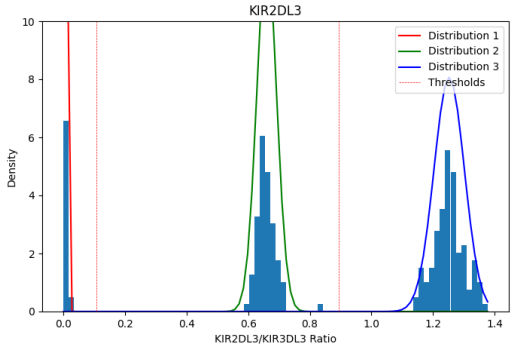
A.1 Plots produced for all KIR genes



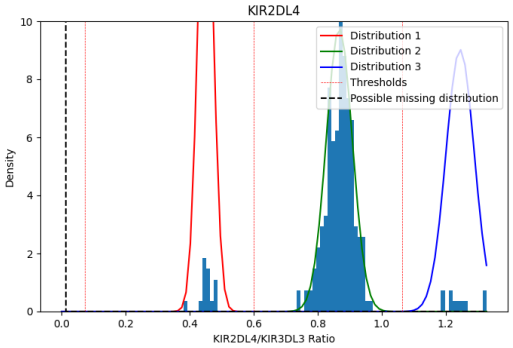
(a) KIR2DL1



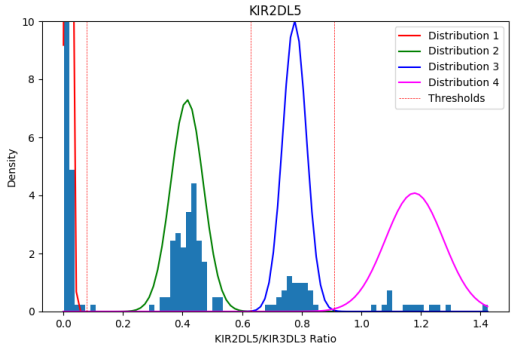
(b) KIR2DL2



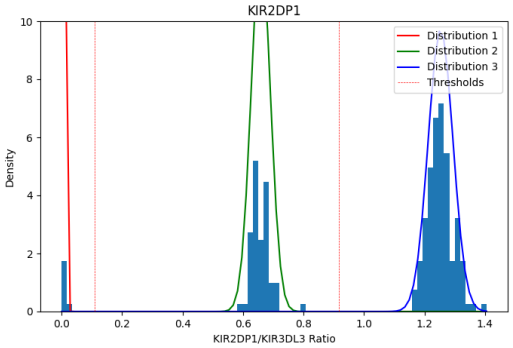
(c) KIR2DL3



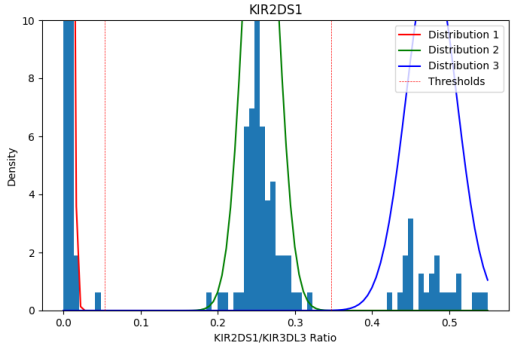
(d) KIR2DL4



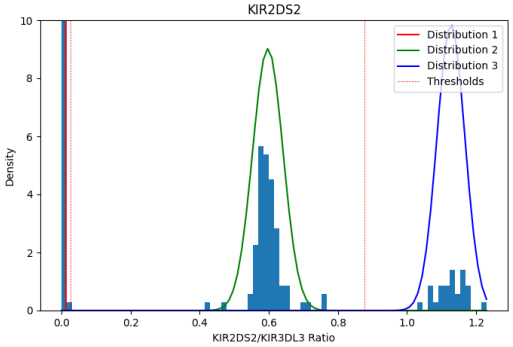
(e) KIR2DL5



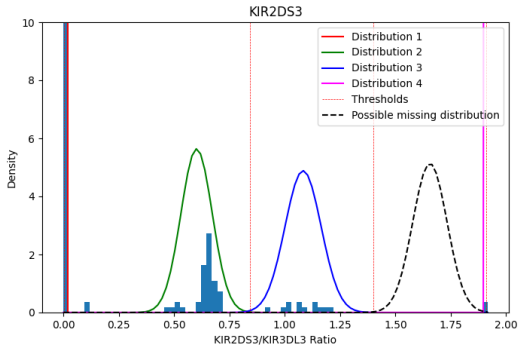
(f) KIR2DP1



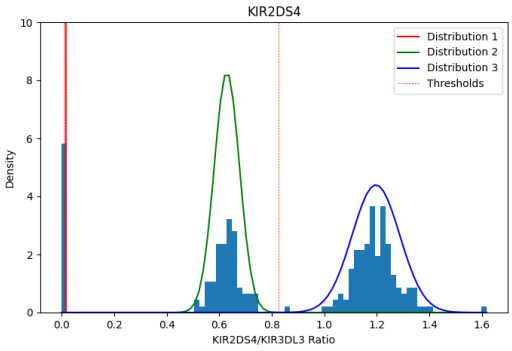
(g) KIR2DS1



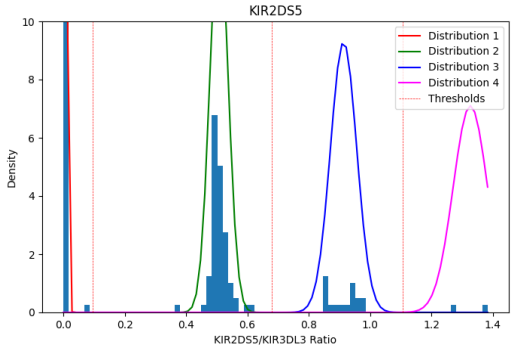
(h) KIR2DS2



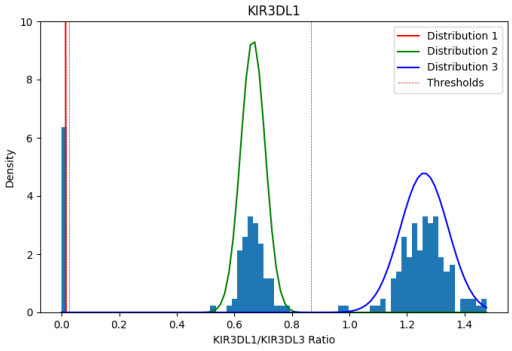
(i) KIR2DS3



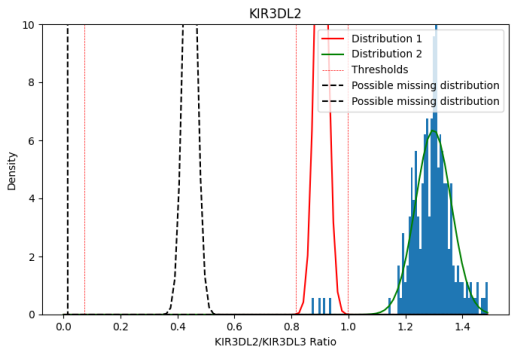
(j) KIR2DS4



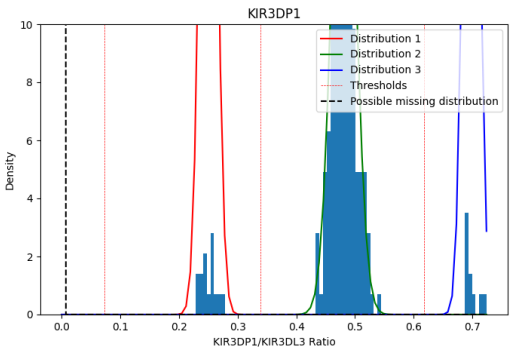
(k) KIR2DS5



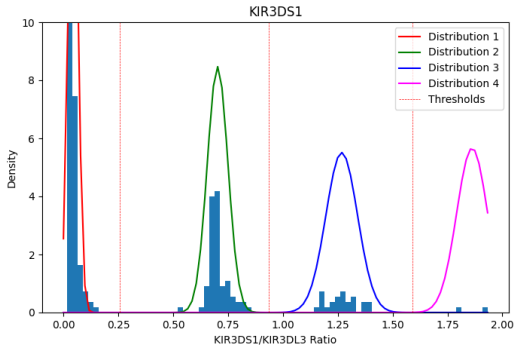
(l) KIR3DL1



(m) KIR3DL2



(n) KIR3DP1



(o) KIR3DS1

Figure 15: Plots for every KIR genes in the data set.
The x-axis represents the read ratio data of each KIR genes to KIR3DL3. The y-axis represents the density. The coloured curves represent the distribution of the zero, one, two, four copy number groups, respectively. The black dotted curve represents the possible missing distribution. The red dotted lines are the thresholds.

A.2 Output file of thresholds for all KIR genes

	0-1	1-2	2-3	3-4	4-5	5-6
KIR3DP1	0.07277145626097840	0.3394181192535230	0.618116935397961			
KIR2DS5	0.09518455245206580	0.6803895784776070	1.1068393847216800			
KIR2DL3	0.10619147769714200	0.8942923606439790				
KIR2DP1	0.10906686354171700	0.9179859366802240				
KIR2DS3	0.0027757371436523000	0.8421704246284240	1.40021730536778	1.911338805947600		
KIR2DS2	0.02524886610364190	0.8774895662603310				
KIR2DL4	0.07277145626097840	0.5995979092680050	1.0636978660198500			
KIR3DL1	0.02649180055239730	0.866671490648511				
KIR3DS1	0.2563678737343200	0.9344950467555710	1.590347514871910			
KIR2DL2	0.016688561088248800	0.6455347959555760				
KIR3DL2	0.07277145626097840	0.8149210304146100	0.998994200932026			
KIR2DS4	0.008760822422648270	0.8266234251177160				
KIR2DL1	0.09644867596897050	0.769325692657364				
KIR2DS1	0.05334386995725990	0.347434929242454				
KIR2DL5	0.07668837446967650	0.6288460626522760	0.9079788202284770			

A.3 Output file of outliers for all KIR genes

	Sample number with low likelihood
KIR3DP1	['69F_S64_L001_R']
KIR2DS5	['1037_S321_L001_R', '1097_S379_L001_R', '1121_S594_L002_R']
KIR2DL3	['1097_S379_L001_R', '1125_S598_L002_R', '23F_S19_L001_R']
KIR2DP1	['16F_S12_L001_R', '23F_S19_L001_R']
KIR2DS3	['1028_S312_L001_R', '1038_S322_L001_R', '1041_S325_L001_R', '1105_S579_L002_R', '1136_S608_L002_R', '1157_S627_L002_R']
KIR2DS2	['1038_S322_L001_R', '1059_S343_L001_R', '1079_S363_L001_R', '1136_S608_L002_R', '23F_S19_L001_R']
KIR2DL4	['27F_S23_L001_R', '66F_S61_L001_R']
KIR3DL1	['1111_S585_L002_R', '1125_S598_L002_R', '1157_S627_L002_R', '20F_S16_L001_R']
KIR3DS1	['1019_S303_L001_R', '1037_S321_L001_R', '1043_S327_L001_R', '1055_S339_L001_R', '1117_S590_L002_R', '1121_S594_L002_R']
KIR2DL2	['1129_S601_L002_R', '1136_S608_L002_R', '23F_S19_L001_R']
KIR3DL2	['1055_S339_L001_R', '1077_S361_L001_R']
KIR2DS4	['1044_S328_L001_R', '1138_S610_L002_R', '23F_S19_L001_R']
KIR2DL1	[]
KIR2DS1	['1037_S321_L001_R', '1121_S594_L002_R']
KIR2DL5	['1037_S321_L001_R', '1113_S586_L002_R']

B Programming

This algorithm was written in Python language and is available for use. It can be accessed via GitHub (`https://github.com/devitahalim/scie30001-project.git`). The algorithm has 109 lines of main code and 464 lines of the functions used in the main code. The data set used is also available in the repository under the folder "data". The run time for the data set used were approximately 1.8657 seconds on standard laptop.