
Bayesian Mixture Models

A short tutorial using AutoStat

Clair Alston-Knox

Powered by



Wed Jun 03 2020

Mixtures for density estimation and clustering

Finite Normal mixture models are frequently used for statistical problems requiring either **density estimation**, or clustering of the data into subgroups using **soft classification**, which yields a probability of being classified into a particular sub-group.

A schematic example of the mixture model is given in Figure 1. The top graph represents an overall data density. The bottom graph illustrates how such a density could be represented by the addition of a number of Normal distributions.

In this example, the data density can be represented by 3 Normal distributions as follows:

$$f(\mathbf{y}) = 0.3N(-1.5, 0.7^2) + 0.6N(0.0, 0.6^2) + 0.1N(2.0, 0.5^2)$$

The sampling weights in this equation ($\{0.3, 0.6, 0.1\}$) represent the proportion of the sample belonging to each of the Normal sub-populations, and ensure that the area under the mixture probability density function is equal to 1.

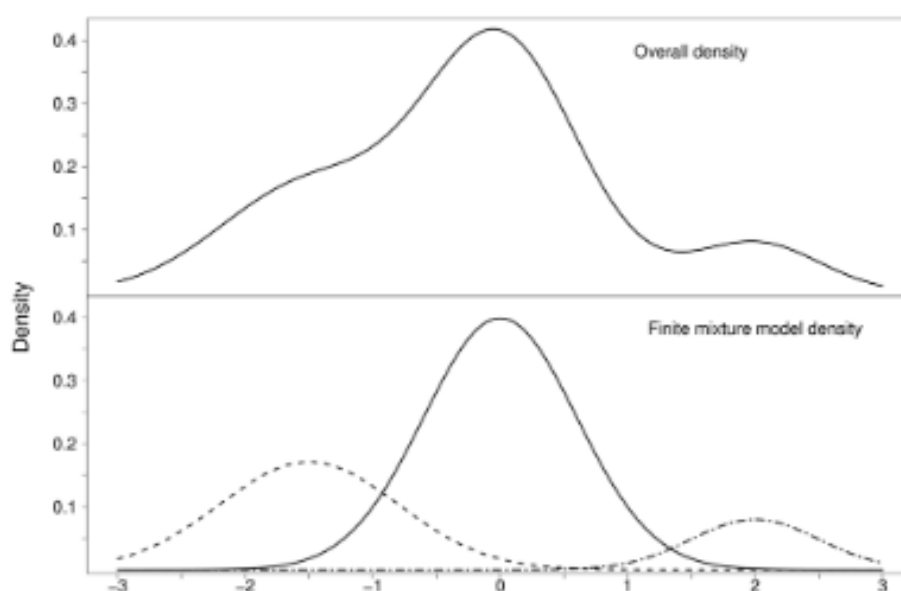


Figure 1: Schematic illustration of a 3 component mixture model

Figure 1 illustrates an example of a mixture comprised of 3 univariate Normal populations. The distributions have a mean of $(-1.5, 0.0, 2.0)$ and standard deviations $(0.7, 0.6, 0.5)$ and sample weights of $(0.3, 0.6, 0.1)$. The top graph represents the density of the 3 populations combined. The bottom graph shows the individual Normal components which are summed to represent this overall density.

The Finite Normal mixture model is extremely flexible, with the analyst being able to fit slightly skewed through to quite awkwardly shaped densities by the addition of many Normal distributions.

In practice, these Normal densities may either represent subgroups within the data which are unobserved, and may be of research interest, or this model may simply be used as a method of constructing a non-symmetric density.

Definition

In this data analysis, we assume that the true probability density function of the data, $g(y)$, may be approximated as a mixture of Normal distributions with a density $\hat{g}(y)$. Typically, in text books, this model is usually written in a standard form that looks like:

$$g(\mathbf{y}) \approx \hat{g}(\mathbf{y}) = \sum_{j=1}^K \pi_k N_d(y_i | \mu_k, \Sigma_k)$$

We note that in this model, K is the number of d-dimensional Normal distributions that are summed (denoted here by the summation symbol $\sum_{j=1}^K$) to obtain the overall density. We can see that each of the Normal distributions have their own unique mean (μ_k) and variance-covariance matrix (Σ_k^2). Also, as this is a probability density estimate, if we integrate the area under the “curve” between $\pm\infty$ we should obtain a value of 1. To meet this criteria, we need to weight each Normal component (remember, for any Normal this integral will return a value of 1), and these weights are given by the parameter π_k . The weights (π_k) also ensure that the proportion of each sub-population in the sample is adequately reflected in the overall density.

If we knew component membership for each measurement (y_i), estimation of the mixture parameters would be straight forward, as we would merely need to calculate (μ_k, σ_k^2) for each subsample. However, if we do not know this information, we need to reformulate the model in a way that incorporates this unknown quantity.

Using a *latent variable* approach, we can pose the mixture model using the following representation:

$$f(y_i | \mu, \Sigma) = \sum_{k=1}^K \pi_k N_d(y_i | \mu_k, \Sigma_k)^{z_i},$$

In this formulation, the vector $z = \{z_1, z_2, \dots, z_N\}$ is a set of unobserved indicators denoting component membership for the observed variables $y_i, i = 1, 2, \dots, N$. These unobserved observations, z_i , are then estimated as another parameter in the model.

Even though the component membership (z_i) is unobserved, we are able to solve the model in the absence of this information using a missing data strategy (latent model). This estimation procedure will be discussed in the **Bayesian Estimation** later in this tutorial.

Example: Old Faithful eruption times

The Old Faithful Geyser (pictured below), located in Yellow Stone National Park, is a classic example of the potential uses of the finite mixture model.

Since it's initial “discovery” by the Washburn Expedition of 1870, observers have recorded both the length of time it's eruptions last, and the waiting time between eruptions. For more information on Old Faithful Geyser: <https://yellowstone.net/geysers/old-faithful/>



Figure 2: Old Faithful Geyser erupting at Yellowstone National Park, September 2013.

The data being used in this tutorial can be viewed in the **Data Manager**.

The relationship between the waiting time and the length of the eruption has been observed that allows reasonably accurate prediction of the anticipated length of the next eruption based on the wait time. In a separate tutorial, a timeseries analysis leads to predictions of the next eruption time. These types of models allow visitors to the park to schedule their viewing of Old Faithful. See <https://www.nps.gov/yell/learn/kidsyouth/predict-old-faithful.html> or <http://web.pdx.edu/~jfreder/M212/oldfaithful.pdf> for further information.

The aim of the analysis in this tutorial will be 2 fold. **Firstly**, we will determine if there are clusters of eruption lengths using a univariate finite Normal mixture model. **We will then extend the analysis**, using a multivariate Normal mixture model to see if these clusters bear a relationship to the waiting time between eruptions.

Exploratory analysis

Univariate densities A sensible first step in the exploration of potential mixture model data is to view the density of the observations. One standard method of viewing a density is by using a *histogram*. In AutoStat, we use the **Visualisation** menu, which produces the following histogram of duration time of eruptions. Start by going to the __Visualisation Module__ and choose the *charttype* **Distributions**

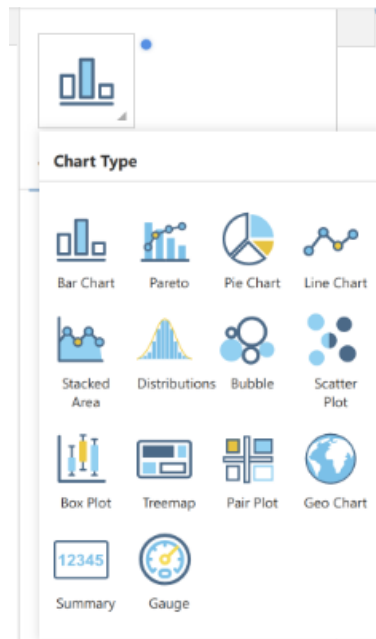


Figure 3: Visualisation module: Choosing a chart type

then select the dataset *OldFaithful.csv* and drag the *eruptions* variable into the *Values* box.

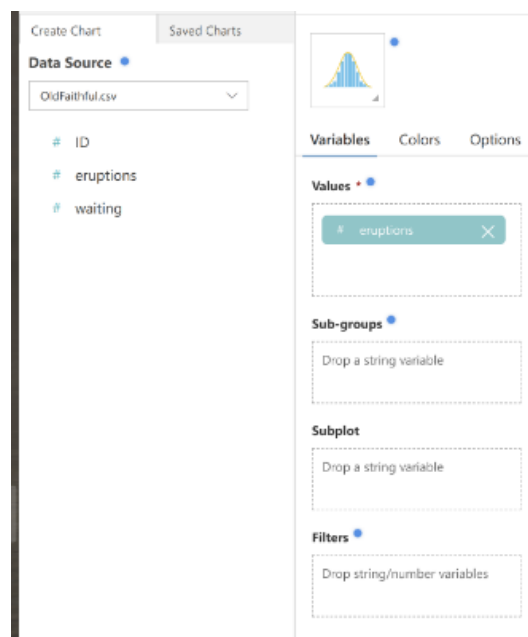


Figure 4: Visualisation module: Density plot

Using the histogram function, the user needs to decide how to construct their bins. The default bin width for histograms in AutoStat is *Scott's rule*. However, the user can alter this using either the rules in the dropdown box, or they can set the number of bins. For example, below is the histogram constructed using the *square root choice* rule.

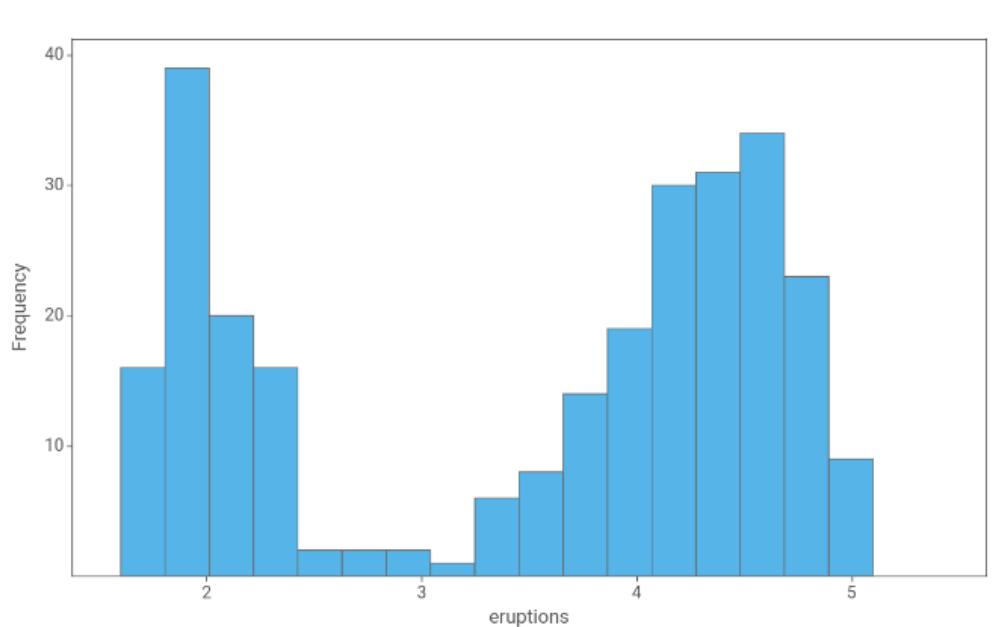


Figure 5: Histogram constructed using square root choice algorithm for binning

The histogram of eruption duration times (also below, using *Scott's rule*) indicates that there may be a mixture of at least 2 subpopulations. The first “obvious” group are centered at approximately 2 minutes in duration, and a second group has a much longer eruption time with an average of around 4.5 minutes.

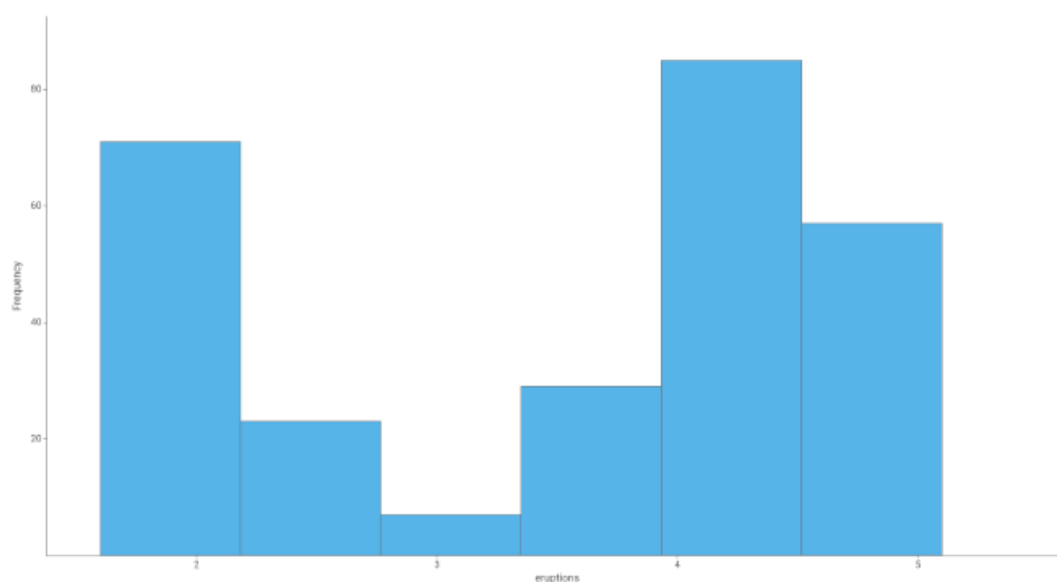
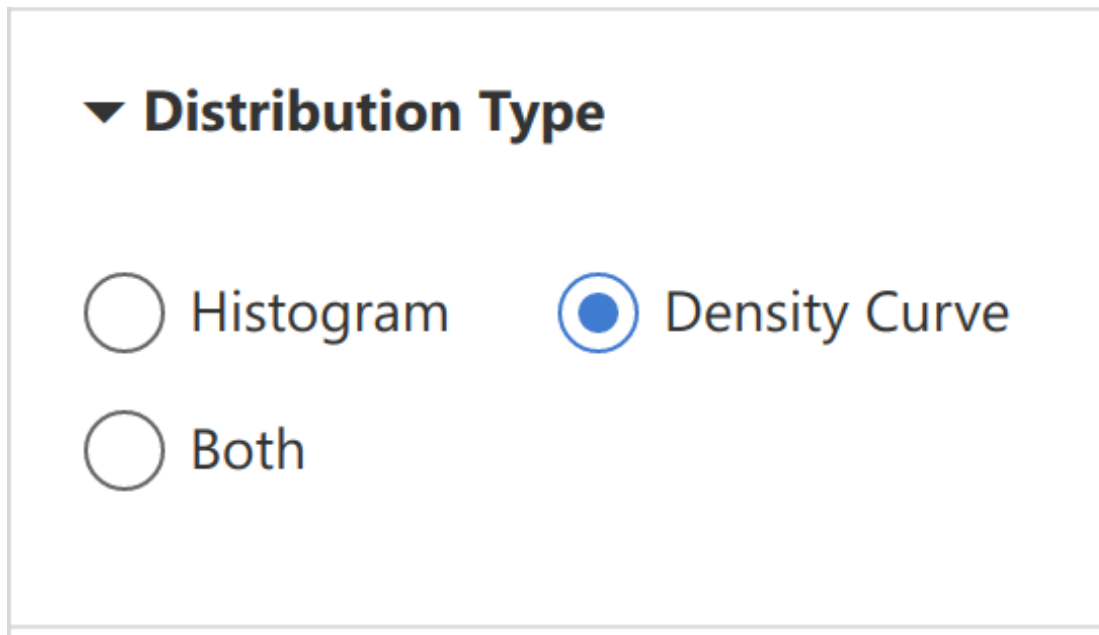


Figure 6: Histogram using Scott's rule to constrict bins

In this example, we note that this *grouping* is more obvious using the *square root choice* option for histogram bins. Sometimes, a more informative view may be obtained by using a kernel density

estimate. In AutoStat we can switch between histogram and kernel density representation using the radio button



▼ Distribution Type

☐ Histogram ☒ Density Curve

☐ Both

Figure 7: Changing from a histogram to a kernel density plot

which will produce the density plot below.

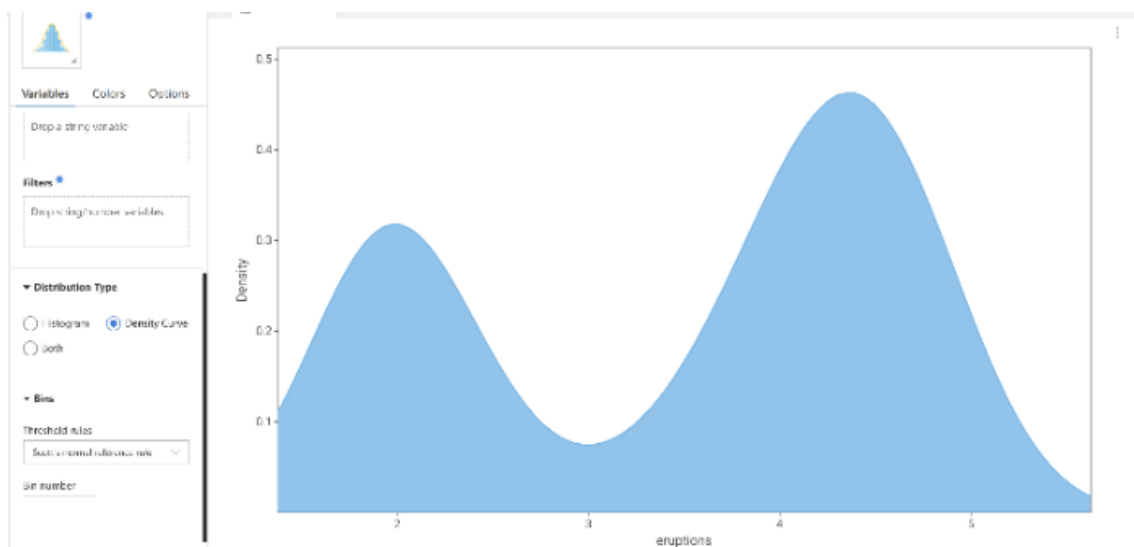


Figure 8: Density of eruption duration times

This gives a smoother version of the density, and again we note the presence of 2 peaks (at around 2 minutes and 4.5 minutes).

All of the graphics of the univariate density have indicated at least 2 peaks, therefore, the use of a mixture model with at least 2 components is a suitable analytical strategy.

Multivariate densities

When considering a multivariate mixture model, it is helpful to view both the univariate density plots and graphs that highlight potential relationships between the variables. The **pairplot** is a useful method of achieving such a visualisation. A pairplot is accessed in AutoStat using the * Visualisation * menu.

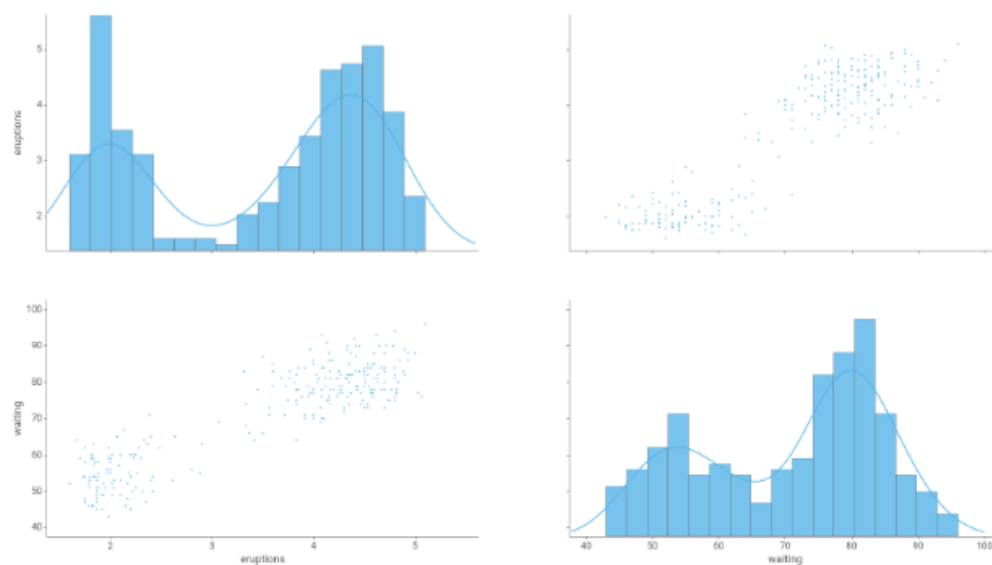


Figure 9: Pairplot of eruption duration and waiting times

The histograms given on the diagonal of pair plot indicate that both length of eruption times and the waiting time between eruptions may be suitable to be modelled by a finite mixture model, as both appear to be at least bimodal, and hence may be modelled using 2 or more Normal distributions.

The scatter plots in the pair plot indicate that there is a relationship between length of eruption times and the waiting time between eruptions, with shorter waiting times ($\approx 50 - 60$ minutes) coinciding with shorter eruption periods (≈ 2 minutes). The second cluster appear to show a relationship of longer waiting times (≈ 80 minutes) and longer eruption times (≈ 4.5 minutes).

Given the reasonably clear relationship between the variables in this plot, with preliminary indications of the appropriateness of a Normal mixture model, we can now proceed to performing the 2 stage analysis.

How many clusters

A line plot of the mean eruption time vs wait time is shown below. We can see that we have one group of eruption times around 2 minutes for wait times of less than 1 hour, and another

group of eruptions which last around 4.5 minutes when wait time is greater than 75 minutes, and that we have some measures between these clusters (around 65-75 minutes wait time) that may indicate the presence of a third cluster. We can see this by the differing wait times that look like a transition between the “short wait / short eruption” and “long wait / long eruption” states.

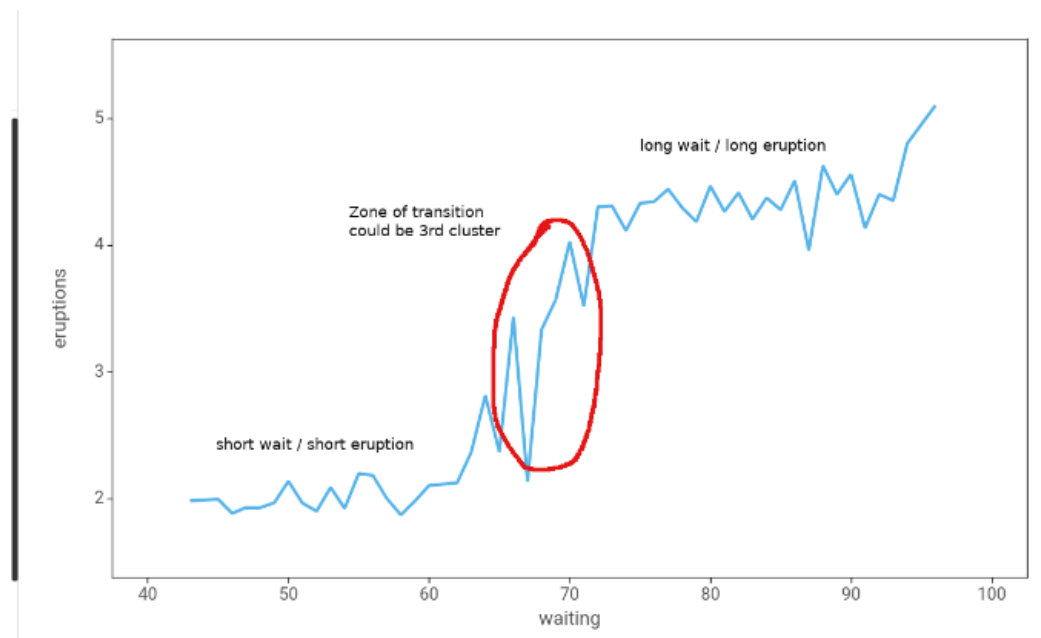


Figure 10: Line chart showing mean eruption duration against the time elapsed between eruptions

However, the scatter plot below (also seen within the pairplot) indicate that if this 3rd grouping is present, we have very few observations, most likely less than 7, which would reasonably be allocated to component “3”. We will illustrate these values later in the tutorial in regards to their allocation probabilities within the 2 cluster model.

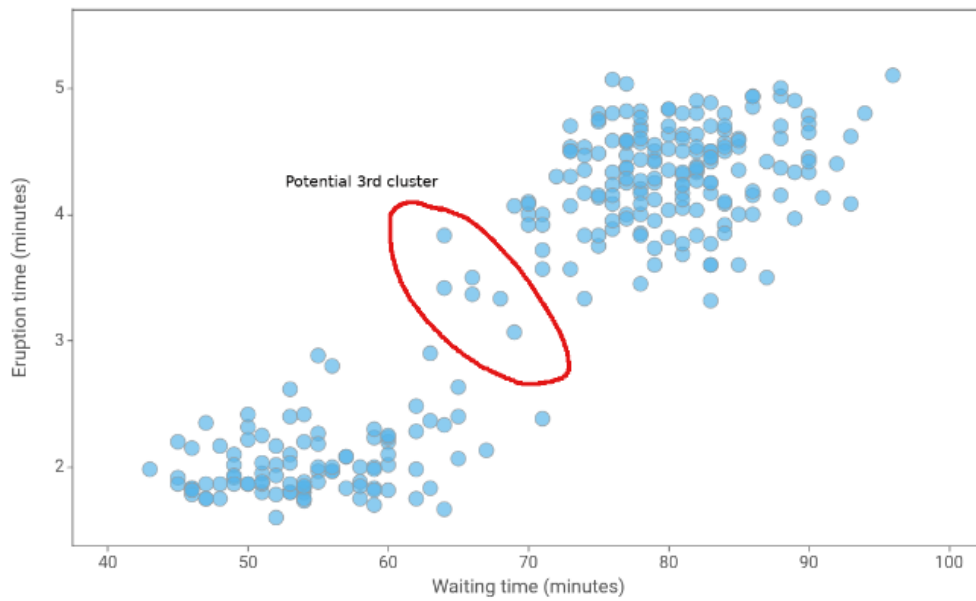


Figure 11: Scatter chart of eruption duration against waiting time

Univariate Mixture: Bayesian Estimation

The Bayesian estimation procedure in AutoStat relies on the previously mentioned latent variable method, where the density of the data is represented using:

$$f(y_i | \mu, \Sigma) = \sum_{k=1}^K \pi_k N_d(y_i | \mu_k, \Sigma_k)^{z_i},$$

Using this formulation, we implement estimation in 2 stages **firstly**, using the current estimates of the parameters (μ, σ^2, π) , **we assign each data observation to a single component** $1, \dots, K$. This latent variable is represented by z_i . The allocation is done using a single simulation from a multinomial distribution, where the probability of membership to component j for observation y_i is:

$$\lambda_j(y_i | \mu_j, \sigma_j^2, \pi_j) = \frac{\pi_j \left(\sqrt{2\pi\sigma_j^2} \right)^{-1} \exp \left[-\frac{1}{2} \left(\frac{y_i - \mu_j}{\sigma_j} \right)^2 \right]}{\sum_{t=1}^K \pi_t \left(\sqrt{2\pi\sigma_t^2} \right)^{-1} \exp \left[-\frac{1}{2} \left(\frac{y_i - \mu_t}{\sigma_t} \right)^2 \right]}$$

The model parameters we are estimating are highlighted in blue. These estimates are formed by simulating values from the posterior distribution for each parameter. At each iteration (S) of the Gibbs sampler, we use the current simulated draw (S-1) of the model parameters $(\mu^{S-1}, \sigma^{2S-1}, \pi^{S-1})$ to simulate the updated allocations z_i . This process requires that these probabilities are recalculated before each new allocation as we iterate through the algorithm.

Now that the measurements have each been allocated to a component, we use this allocation to **update the parameters** (μ, σ^2, π) as stage 2 in the MCMC algorithm.

In order to achieve this, we use the allocated component membership (z_i) to calculate the current component mean (\bar{y}_j), variance (\hat{s}_j^2) and the number of observations allocated to each component (m_j). These calculated values will change at each iteration, as z_i are reallocated.

($\bar{y}_j, \hat{s}_j^2, m_j$) are used in stage 2 of the procedure to sample from the posterior distributions for the unknown parameters (μ, σ^2, π).

Posterior simulations

AutoStat implements the estimation of the mixture model parameters using the following conjugate priors.

$$\sigma_k \sim \text{InvGamma}(0.5 * \nu, 0.5 * s)$$

$$\mu_k | \Sigma_k \sim \text{N} \left(\xi, \frac{\sigma_k}{n} \right)$$

$$\pi \sim \text{Dirichlet}(\alpha, \alpha, \dots, \alpha)$$

Using the above priors, the simulation of a new value from the posterior distribution of each parameter is achieved using the following distributions:

$$\mu_j \sim \text{N} \left(\frac{m_j \bar{y}_j + n \xi}{m_j + n}, \frac{\sigma_j^2}{m_j + n} \right)$$

$$\sigma_j^2 \sim \text{InvGamma} \left(\frac{m_j + \nu + 1}{2}, \frac{1}{2} \left[s^2 + \hat{s}_j^2 + \frac{m_j}{n m_j} + n (\bar{y}_j - \xi)^2 \right] \right)$$

$$\pi | y, Z \sim \text{Dirichlet}(\alpha + m_1, \dots, \alpha + m_K)$$

The hyper-parameters of the *prior distributions* in the above posterior distributions are marked in red. These hyper-parameters are values that the user needs to define in order to form a *prior distribution*. In AutoStat, for the univariate mixture, the default prior hyperparameter specifications are based on the recommendations of Raftery (1996). The hyper-parameters are data-dependent as follows:

$$\xi = \bar{y} = 3.49$$

$$n = \frac{2.6}{y * \max - y * \min} = 0.74$$

$$\nu = 2 * 1.28$$

$$s^2 = 0.36s_y^2 = 2 * 0.47$$

$$\alpha = 1$$

To the right we have provided the values of the data-dependant default priors for this example in red.

The default priors appear in the the right of the AutoStat * Gaussian Mixture Model* menu. These can be altered by the user if desired.

The Model Builder

The Mixture Model inputs will look like this in the **Model Builder**

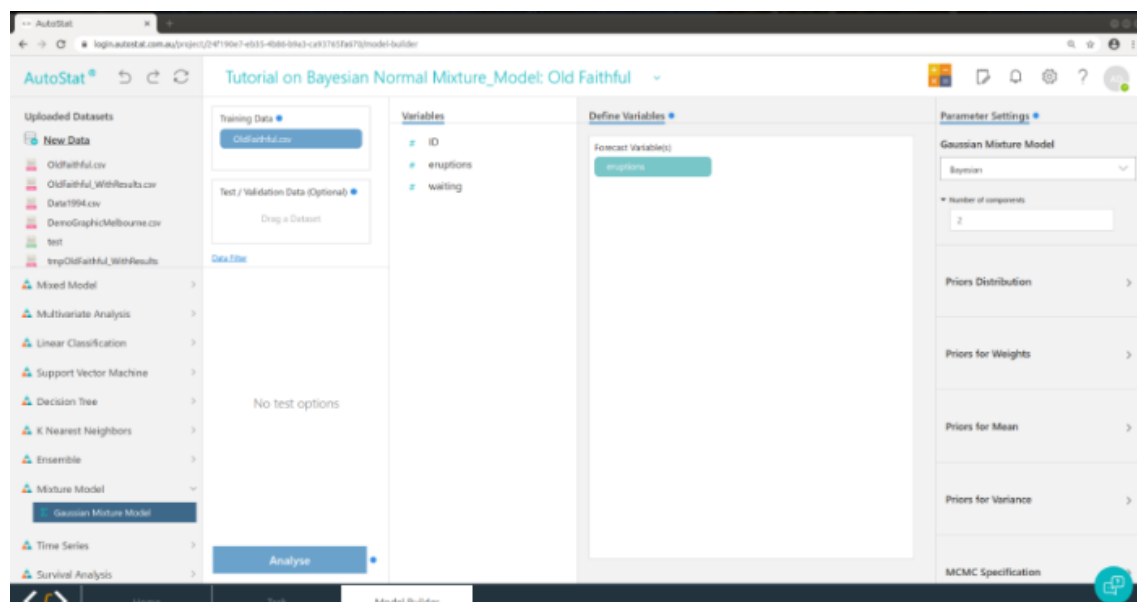
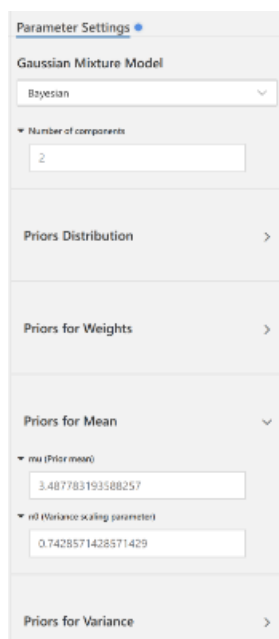


Figure 12: Model Builder: Setting up the mixture model

To access the prior parameters and MCMC specification, such as the number of iterations, random seed and burn-in period, use the drop down menu for each component of the model to the right of the model builder screen by selecting the appropriate “>” symbol.



The screenshot shows a 'Parameter Settings' dialog box for a 'Gaussian Mixture Model'. The 'Bayesian' option is selected in a dropdown menu. The 'Number of components' is set to 2. There are expandable sections for 'Priors Distribution', 'Priors for Weights', 'Priors for Mean', and 'Priors for Variance'. The 'Priors for Mean' section is expanded, showing two input fields: 'mu (Prior mean)' with the value 3.487783193588257 and 'sigma (Variance scaling parameter)' with the value 0.7428571428571429.

Figure 13: Prior distribution specification in Model Builder

MCMC specifications

Finally, the user needs to specify the following:

1. The **random seed**. It is important that the user keeps a record of this value as using a different seed will likely yield estimates that are slightly different (at some decimal place).
2. The number of iterations of the algorithm you would like to include in the posterior estimation. This is akin to the number of simulations from the posterior distribution you will draw.
3. The burnin length. This is the number of initial simulations you will discard. These initial estimates are not included in the posterior estimates as they are considered to be initial draws which have not yet resulted in stable estimates of the posterior.

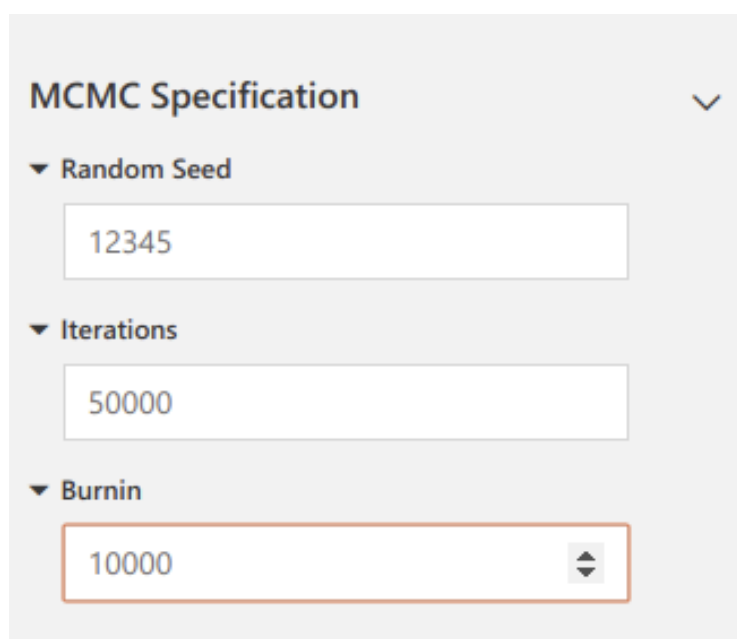


Figure 14: MCMC specification in Model Builder

Results

The results of the Bayesian mixture model can be found in the **Model Results** module.

From the results, we have identified 2 components, with the estimated mixture given as

$$f(\mathbf{y}) = 0.36 * N(2.05, 0.30) + 0.64 * N(4.28, 0.42)$$

with the standard errors and highest predictive posterior density given in the table below. The **IFactor** in the table indicates how efficient the MCMC sampling is. This value is a multiple of how many more iterations of the algorithm you would need to perform to gain an independent sample of the same size as your current model run. For example, if the IFactor was 2, you would need to run twice as many iterations and thin the sample to every 2nd iterate to gain the desired independent sample. As all our IFactors are around 1, we are satisfied with the performance of the algorithm in this case.

Component 1

Variable	Mean	SD	HPD 2.5%	HPD 97.5	IFactor
mean 1	2.0467	0.0322	1.9815	2.1090	1.1807
variance 1	0.3041	0.0244	0.2587	0.3541	1.7211
weights 1	0.3564	0.0289	0.2996	0.4116	1.1247

Component 2

Variable	Mean	SD	HPD 2.5%	HPD 97.5	IFactor
mean 2	4.2849	0.0327	4.2215	4.3496	1.0243
variance 2	0.4219	0.0246	0.3767	0.4722	1.5348
weights 2	0.6436	0.0289	0.5884	0.7004	1.1247

Figure 15: Summary tables from Model Output module

The IFactor estimate is further illustrated in the **auto-correlation** plots, which are shown as the middle graph in the **MCMC Charts** tab. From the graphs in this example, we see no evident auto-correlation in the MCMC posterior draw chain for any of the estimated coefficients.

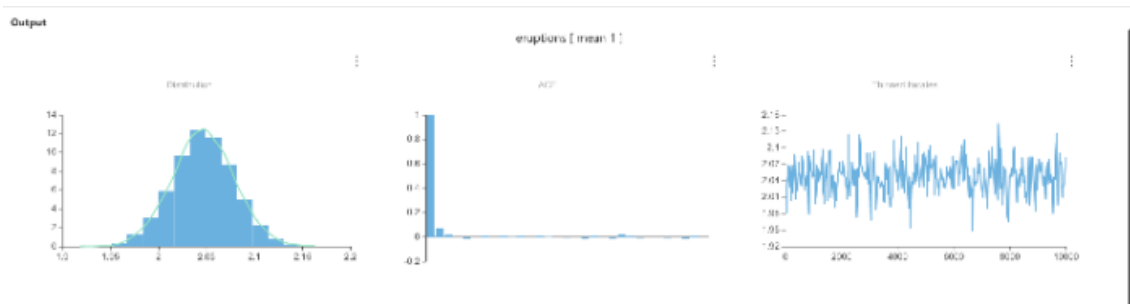


Figure 16: Graphical representation of MCMC iterations

The posterior distribution, illustrated as the first graph (histogram) is nicely symmetric for all coefficients, indicating that the solution has converged on a single estimate, with no evidence of bimodality, or skewing in the case of μ and π . Skewing in the case of the variance estimate is acceptable and expected, due to its Inverse-Gamma specification.

Finally, the trace plots of the iterates for each coefficient are also indicate that estimates have converged to a single posterior distribution, with random scatter around a central tendency.

Key Concept: *It is important to note that in the case of a mixture model, an algorithm that is mixing well should in fact provide trace plots that may be subject to label switching. This is not generally true of other models, where we would expect to see trace plots that have converged to a single estimate. Mixture models have no natural ordering of the components, and as such, at any iteration, these posterior simulations can be reordered. However, in AutoStat, we have implemented a label switching algorithm during the MCMC estimation to increase computational efficiency and prevent the need for post-processing.*

We can save the results of our univariate mixture model either into a new file or the existing file using the **save results** button in the **Model Output** module. We can rename the output variables if desired, by choosing the edit pencil associated with each output variable.

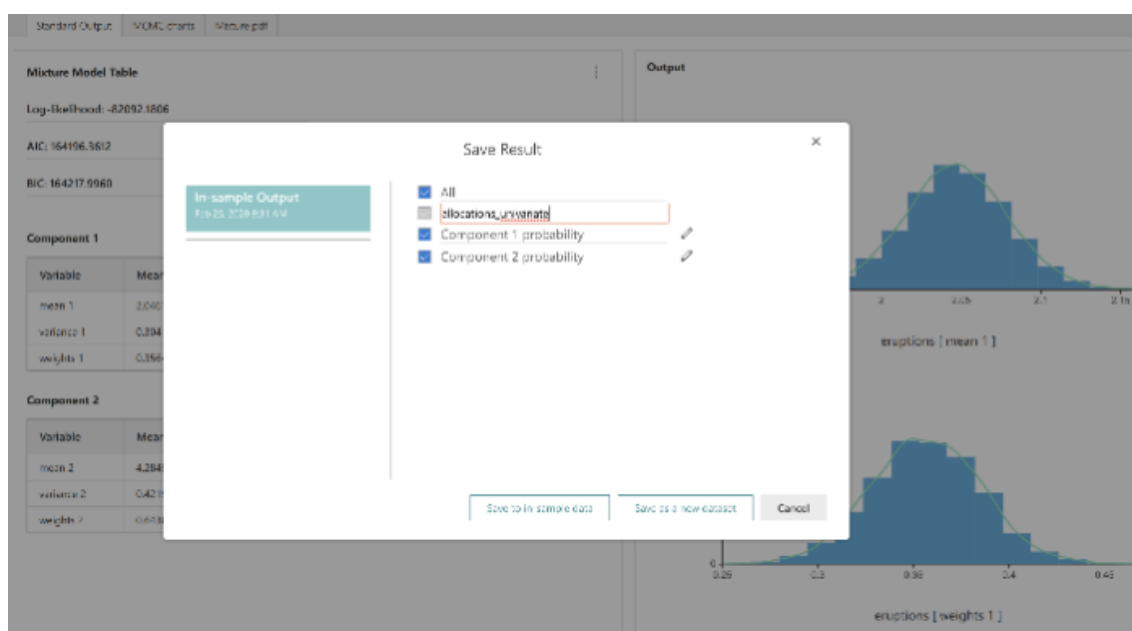


Figure 17: Saving results of MCMC estimation

We can then use the results in other AutoStat modules, to perform further analyses or visualisations. Below we have used the component allocation probabilities in a line chart to understand the region of eruption times that are less certain in their component membership. The blue line is the probability of the eruption time belonging to component 1, and the mustard line represents the probability of being allocated to component 2. We can observe this uncertainty in the region between 2.5 and 3.5 minutes, with around the 3 minute eruption time being equal weight as to whether it belongs to the first or second component. This is in keeping with our earlier observations during the exploratory data analysis.

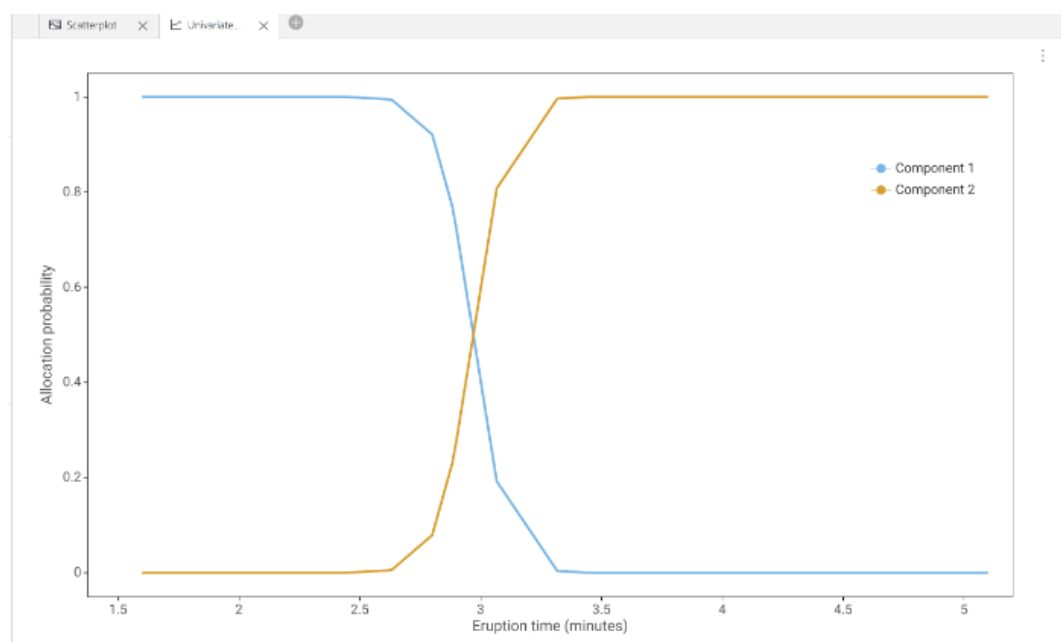


Figure 18: Line chart of mean allocation probability vs eruption time

Bivariate mixtures

The conjugate priors for the multivariate mixture model are the multivariate extensions of the univariate model, with the sampling weights prior remaining the same as its role is unchanged with the increasing dimensions.

$$\Sigma_k^{-1} \sim \text{Wishart}(c_0, C_0)$$

$$\mu_k \mid \Sigma_k \sim \text{MVN}\left(b_0, \frac{\Sigma_k}{N_0}\right)$$

$$\pi \sim \text{Dirichlet}(\xi_0, \xi_0, \dots, \xi_0)$$

In AutoStat, the default hyper-parameters for the multivariate mixtures is based on the recommendations of Robert (1996):

$$b_0 = \bar{\mathbf{Y}} = (3.49, 70.89)$$

$$N_0 = 1 = 1$$

$$c_0 = 3 = 3 * 2$$

$$C_0 = 0.75S_y^2 = \text{diag}(2.71, 0.02)$$

where \bar{Y} is the sample mean for each variable and S_y^2 is the sample variance–covariance matrix (with only the diagonal elements displayed on the GUI).

Results

Once again we have estimated a 2 component mixture model, with the parameters given below. We note that we have captured the “short wait / short eruption” (component 2) and “long wait / long eruption” (component 1) states we were expecting from our exploratory data analysis.

Component 1					
Variable	Mean	SD	HPD 2.5%	HPD 97.5	IFactor
mean [eruptions]	4.2889	0.0324	4.2248	4.3517	1.0454
mean [waiting]	79.9609	0.4865	79.0116	80.9276	1.0153
mean [Weights]	0.6411	0.0291	0.5847	0.6986	1.0146
$\begin{bmatrix} 0.1931 & 0.6360 \\ 0.0005 & 0.0661 \\ 0.6360 & 39.0514 \\ 0.0661 & 17.6403 \end{bmatrix}$					
Component 2					
Variable	Mean	SD	HPD 2.5%	HPD 97.5	IFactor
mean [eruptions]	2.0570	0.0380	1.9819	2.1315	1.1652
mean [waiting]	54.7132	0.6680	53.3811	56.0067	1.0458
mean [Weights]	0.3589	0.0291	0.3014	0.4153	1.0146
$\begin{bmatrix} 0.1409 & 0.2286 \\ 0.0005 & 0.1384 \\ 0.2286 & 42.3934 \\ 0.1384 & 37.3241 \end{bmatrix}$					

Figure 19: Bivariate mixture summary tables

We note that the component means for eruption times are very similar to the univariate model (equivalent to 2dp) and similarly, the sampling weights are also on par with this previous model. The component variance is now represented in matrix form.

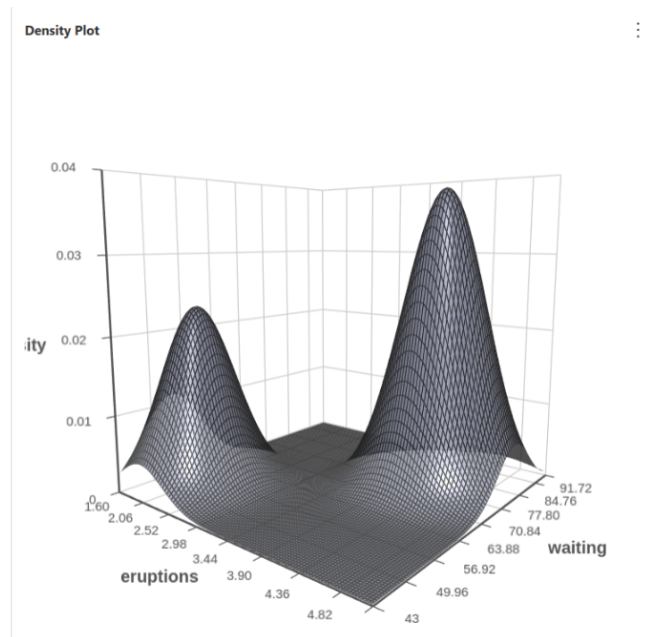


Figure 20: Posterior density estimate of bivariate model

The density for this example is represented in the results in both the marginal and joint form (given below). In AutoStat, this joint density can be rotated to see potentially hidden features. We observe 2 quite sepearted components in this visualisation.

The results can be saved and graphed using other visualisations. For example, a simple scatter plot of the allocations can provide information such as the allocation of the “potential 3rd cluster”, which we see have all been allocated to component 1 (long wait / long duration)

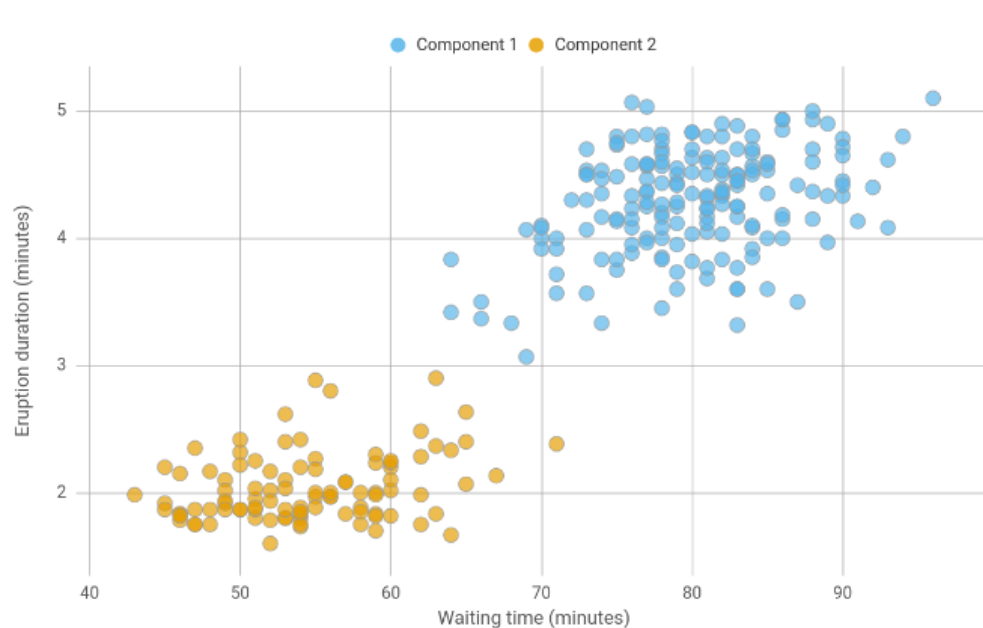


Figure 21: Allocation of data into components 1 and 2

Similarly, the pairplot can display this information with the addition of the density plot for each allocation subgroup.

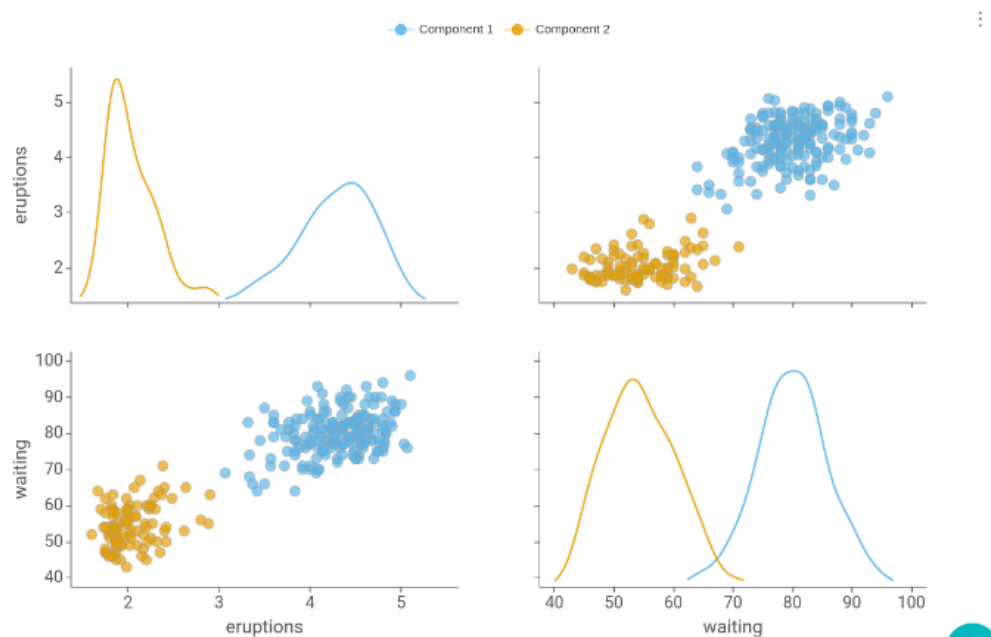


Figure 22: Pairplot of data with allocation subgroups

Information can also be displayed at the individual variable level. For example, using the allocation as a subgroup, the histogram of waiting time shows us a zone (around 65 - 75 minutes) where observations could be allocated to either component

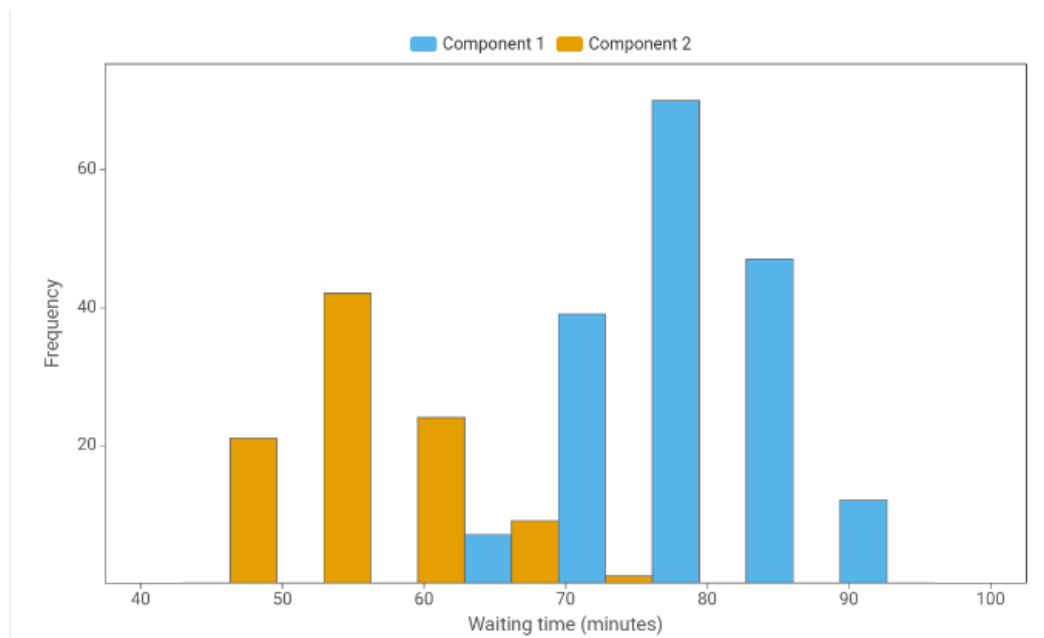


Figure 23: Histogram of waiting time with allocation as subgroups

Further exploration, as in the scatter plot below, reveals that while there is overlap between the

wait times in terms of allocations, as shown in the above histogram, and represented by the area between the green dashed lines below, the same is not true of the eruption duration times, with a value of around 3 minutes separating the component allocations (red line in scatterplot below).

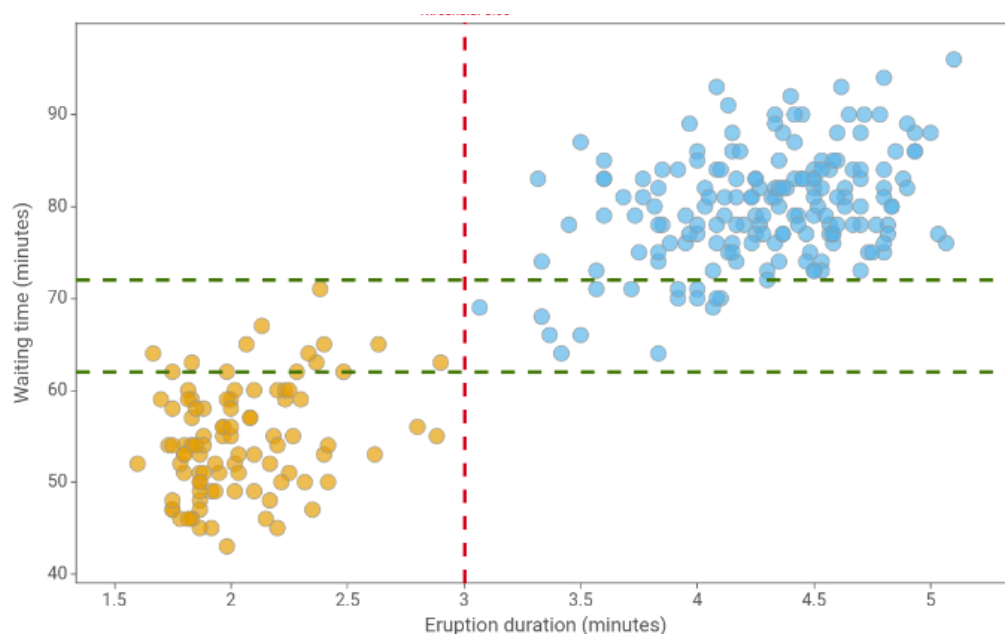


Figure 24: Diagrammatic representation of overlap between components

References

1. A. E. Raftery. Hypothesis testing and model selection. In W.R. Gilks, S. Richardson, and D. Spiegelhalter, editors, *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics, 1995.
2. C.P. Robert. Mixtures of distributions: Inference and estimation. In W.R. Gilks, S. Richardson, and D. Spiegelhalter, editors, *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC Interdisciplinary Statistics, 1995.