Contents

Chapter 0
References

- Böhning D. (1999) *Computer-assisted Analysis of Mixtures and Applications. Meta-analysis, Disease Mapping and Others*. London: Chapman & Hall.

- Cressie, N.A.C. (1991) *Statistics for Spatial Data.* New York: John Wiley.

- Davidian, M. and Giltinan, D.M. (1995) *Nonlinear Models for Repeated Measurement Data.* London: Chapman & Hall.

- Diggle, P.J. (1983) *Statistical Analysis of Spatial Point Patterns*. Mathematics in Biology. London: Academic Press.

- Fahrmeir, L. and Tutz, G. (2001) *Multivariate Statistical Modelling Based on Generalized Linear Models*. Heidelberg: Springer-Verlag.

- Fitzmaurice, G.M., Davidian, M., Verbeke, G., and Molenberghs, G.(2009). *Longitudinal Data Analysis. Handbook.* Hoboken, NJ: John Wiley & Sons.

- Lindsay B.G. (1995) *Mixture Models: Theory, Geometry and Applications*. NSF-CBMS regional conference series in probability and statistics, vol.5, Hayward: Institute of Mathematical Statistics.

- McLachlan G.J. and Basford K.E. (1988) *Mixture Models. Inference and Applications to Clustering*. New York: Marcel Dekker.

- McLachlan, G.J. and Peel, D. (2000) *Finite Mixture Models*. New York: John Wiley.

- Molenberghs, G. and Verbeke, G. (2005). *Models for Discrete Longitudinal Data.* New York: Springer-Verlag.

- Ripley, B.D. (1981) *Spatial Statistics*. New York: John Wiley.

- Titterington D.M., Smith A.F.M., & Makov U.E. (1985) *Statistical Analysis of Finite Mixture Distributions*. New York: John Wiley.

- Verbeke, G. and Molenberghs, G. (2000) *Linear mixed models for longitudinal data*. New York: Springer-Verlag.

Part I

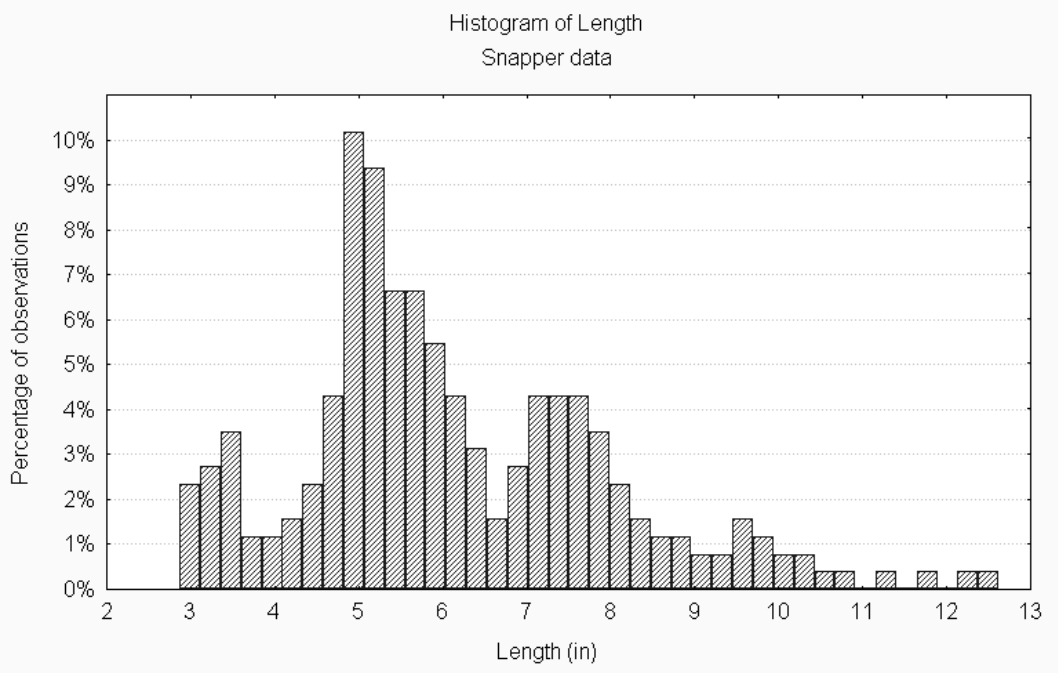
Inference for Mixed Populations

Chapter I.1
Introduction to Finite Mixtures

- Snapper data
- Unobserved heterogeneity
- The cocktail example
- Mixture of normals

I.1.1 Snapper Data

- Data set snapper.dat
- Length measurements (inches) of 256 snappers, with histogram:



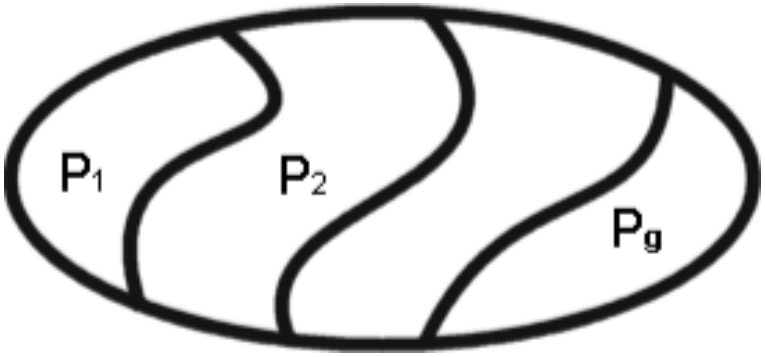
- Histogram shows multi-modality which cannot easily be described by standard distributions.

- Biological interpretation:
  - ▷ Underlying categories correspond to age classes
  - ▷ Within each age class a rather 'homogeneous' distribution seems plausible
  - ▷ The relative heights of the modes give an indication of the proportion of the population in that particular age class.

I.1.2 Unobserved Heterogeneity

- The multi-modality observed in the histogram suggests the presence of some underlying (latent) group structure
- In many cases, as in the snapper data, the group structure is not known or has not been recorded
- Let us assume that the population $\mathcal{P}$ of interest is composed of g sub-populations $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_g$:

$$\mathcal{P} = \{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_g\}$$



- Each population $\mathcal{P}_j$ represents a proportion π_j of the total population, $\Sigma_{j=1}^{g} \pi_j = 1$
- Let X indicate from which population an observation has been sampled:

$$X = j \quad \Longleftrightarrow \quad \text{Observation belongs to } \mathcal{P}_j$$

- The distribution of X is discrete with support $\{1, 2, \ldots, g\}$ and corresponding probabilities $\{\pi_1, \pi_2, \ldots, \pi_g\}$:

$$X \sim \begin{pmatrix} 1 & 2 & \ldots & g \\ \pi_1 & \pi_2 & \ldots & \pi_g \end{pmatrix}$$

- X is latent, as it is not observed

- Let the density of the outcome Y in sub-population $\mathcal{P}_j$ be $f_j(y)$
- The density of Y in the entire population $\mathcal{P}$ then equals:

$$f(y) = \sum_j f(y|X=j)P(X=j) = \sum_j \pi_j f_j(y)$$

- The distribution of Y is called a (finite) mixture with g components
- The densities $f_1(y), \ldots, f_g(y)$ often depend on (vectors of) (un-)known parameters $\theta_1, \ldots, \theta_g$.
- The densities $f_1(y), \ldots, f_g(y)$ can be continuous, discrete, or a mixture of both types.

I.1.3 The Cocktail Example

- A mixture can be compared to a cocktail which is a stirred mixture of a number of ingredients, each representing a percentage of the cocktail:

$$\begin{aligned} \text{Cocktail} &\longleftrightarrow \mathcal{P} &&\longleftrightarrow Y \sim f(y) \\ \text{Ingredient} &\longleftrightarrow \mathcal{P}_j &&\longleftrightarrow Y|X = j \sim f_j(y) \\ \text{Relative proportion} &\longleftrightarrow \pi_j &&\longleftrightarrow P(X = j) \end{aligned}$$

- Research questions:
  - ▷ How many ingredients ?
  - ▷ Which ingredients ?
  - ▷ Relative proportions ?

I.1.4 Snapper Data Revisited

- The four modes suggest a 4-component mixture
- A 4-component mixture of normals with equal variance has been fitted:

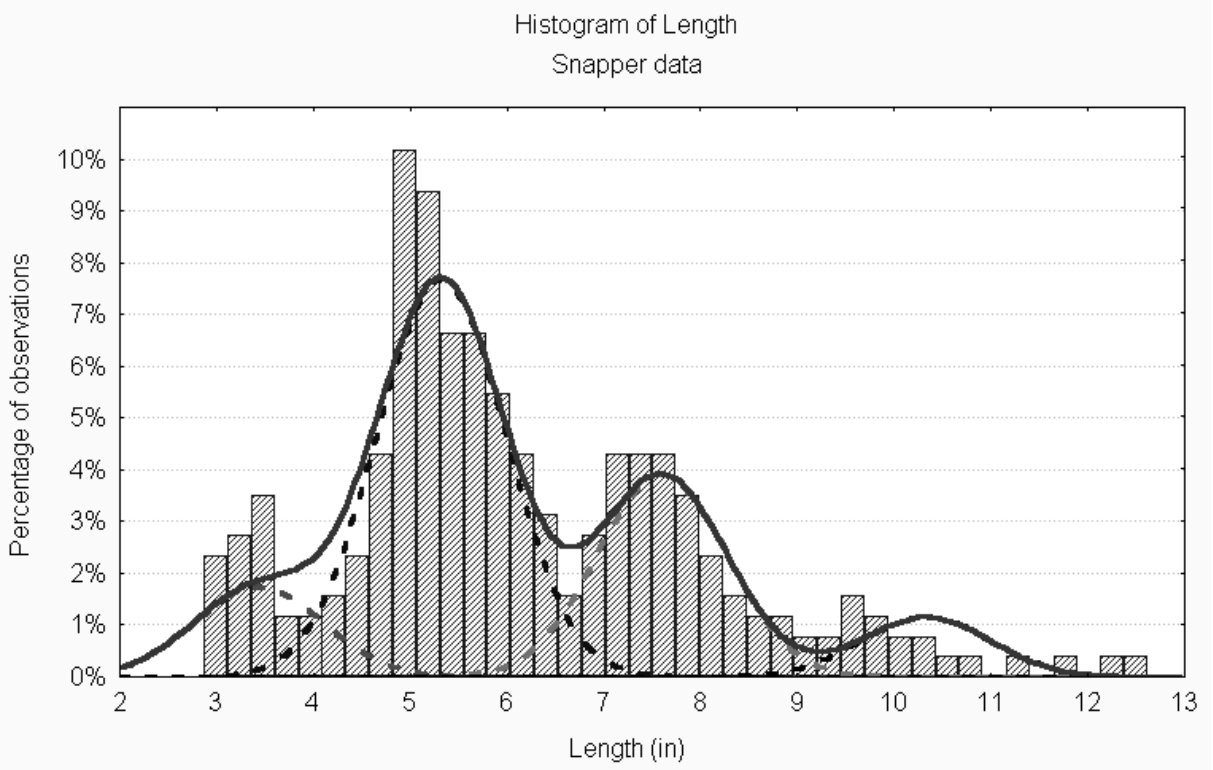
$$Y|X = j \sim N(\mu_j, \sigma^2) \qquad X \sim \begin{pmatrix} 1 & 2 & 3 & 4 \\ \pi_1 & \pi_2 & \pi_3 & \pi_4 \end{pmatrix}$$

- Equivalently, this can be written as:

$$Y|\mu \sim N(\mu, \sigma^2) \qquad \mu \sim \begin{pmatrix} \mu_1 & \mu_2 & \mu_3 & \mu_4 \\ \pi_1 & \pi_2 & \pi_3 & \pi_4 \end{pmatrix}$$

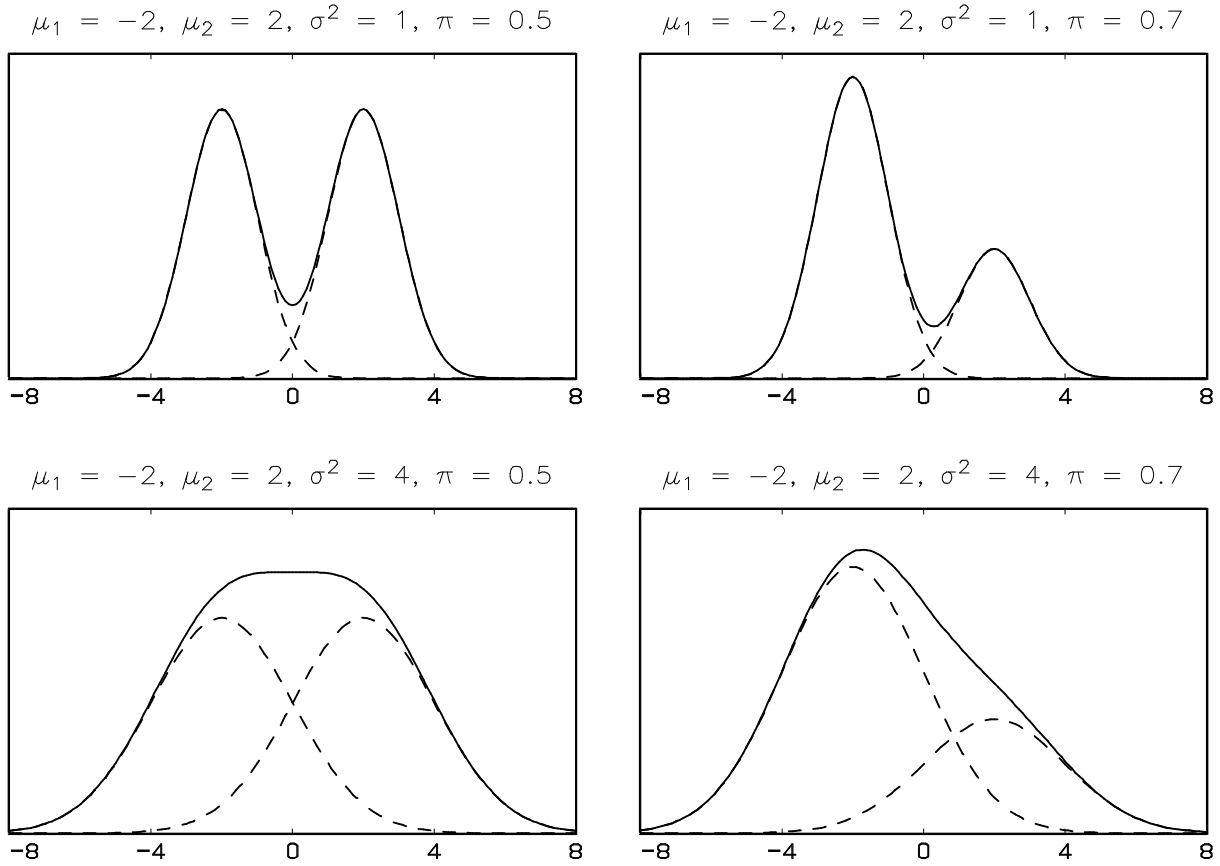
- Fitted model:

$$Y|\mu \sim N(\mu, 0.67^2) \qquad \mu \sim \begin{pmatrix} 3.43 & 5.32 & 7.60 & 10.33 \\ 0.12 & 0.53 & 0.27 & 0.08 \end{pmatrix}$$



I.1.5 Mixture of Two Normals With Equal Variance

- Graphical representation of the mixture: $\pi\ N(\mu_1, \sigma^2)\ +\ (1-\pi)\ N(\mu_2, \sigma^2)$



- Very flexible class of models:
  - ▷ Symmetric as well as skewed
  - ▷ Unimodal as well as multimodal
- If $|\mu_1 - \mu_2|/\sigma \leq 2$ then the mixture is unimodal for all π
- If $|\mu_1 - \mu_2|/\sigma > 2$ then the modality of the mixture depends on π
- In general, the modes of the mixture are closer to each other than the modes of the components.
- Hence the number of components may not be graphically visible as was the case with the snapper data
- Also, the 'appropriate' g very much depends on the component densities $f_j(y)$

Chapter I.2
Mean and Variance of Finite Mixtures

- ▷ General principle
- ▷ Examples

I.2.1 General Principle

- Moments can easily be obtained using the latent variable representation:

$$\mathsf{E}(Y) = \mathsf{E}[\mathsf{E}(Y|X)]$$

$$\mathsf{Var}(Y) = \mathsf{Var}[\mathsf{E}(Y|X)] + \mathsf{E}[\mathsf{Var}(Y|X)]$$

- Conditional moments $\mathsf{E}(Y|X)$ and $\mathsf{Var}(Y|X)$ directly follow from the component densities f_j

I.2.2 Normals With Common Variance

$$Y|\mu \sim N(\mu, \sigma^2) \qquad \mu \sim \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

$$\mathsf{E}(Y) = \mathsf{E}(\mu) = \sum_j \pi_j \mu_j$$

$$\mathsf{Var}(Y) = \mathsf{Var}(\mu) + \mathsf{E}(\sigma^2) = \mathsf{Var}(\mu) + \sigma^2$$

$$= \sum_j \pi_j \mu_j^2 - \left(\sum_j \pi_j \mu_j\right)^2 + \sigma^2$$

I.2.3 Normals With Common Mean

$$Y|\sigma^2 \sim N(\mu, \sigma^2) \qquad \sigma^2 \sim \begin{pmatrix} \sigma_1^2 & \sigma_2^2 & \cdots & \sigma_g^2 \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

$$\mathsf{E}(Y) = \mathsf{E}(\mu) = \mu$$

$$\mathsf{Var}(Y) = \mathsf{Var}(\mu) + \mathsf{E}(\sigma^2) = \mathsf{E}(\sigma^2)$$

$$= \sum_j \pi_j \sigma_j^2$$

I.2.4 Normals With General Mean and Variance

$$Y|(\mu,\sigma^2) \sim N(\mu,\sigma^2) \qquad (\mu,\sigma^2) \sim \begin{pmatrix} (\mu_1,\sigma_1^2) & (\mu_2,\sigma_2^2) & \cdots & (\mu_g,\sigma_g^2) \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

$$\mathsf{E}(Y) = \mathsf{E}(\mu) = \sum_j \pi_j \mu_j$$

$$\mathsf{Var}(Y) = \mathsf{Var}(\mu) + \mathsf{E}(\sigma^2) = \mathsf{Var}(\mu) + \sum_j \pi_j \sigma_j^2$$

$$= \sum_j \pi_j \mu_j^2 - \left(\sum_j \pi_j \mu_j\right)^2 + \sum_j \pi_j \sigma_j^2$$

I.2.5 Binomials

$$Y|p \sim \mathsf{Bin}(n,p) \qquad p \sim \begin{pmatrix} p_1 & p_2 & \cdots & p_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

$$\mathsf{E}(Y) = \mathsf{E}(np) = n\sum_j \pi_j p_j$$

$$\begin{aligned} \mathsf{Var}(Y) &= \mathsf{Var}(np) + \mathsf{E}[np(1-p)] = n^2\mathsf{Var}(p) + n\mathsf{E}(p) - n\mathsf{E}(p^2) \\ &= n(n-1)\mathsf{E}(p^2) - n^2[\mathsf{E}(p)]^2 + n\mathsf{E}(p) \\ &= n(n-1)\sum_j \pi_j p_j^2 - n^2\left(\sum_j \pi_j p_j\right)^2 + n\sum_j \pi_j p_j \end{aligned}$$

I.2.6 Poissons

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

$$\text{E}(Y) = \text{E}(\lambda) = \sum_j \pi_j \lambda_j$$

$$\begin{aligned} \text{Var}(Y) &= \text{Var}(\lambda) + \text{E}(\lambda) \\ &= \sum_j \pi_j \lambda_j^2 - \left(\sum_j \pi_j \lambda_j\right)^2 + \sum_j \pi_j \lambda_j \end{aligned}$$

Chapter I.3
Inference For Finite Mixtures With Fixed Support Size

- ▷ Introduction
- ▷ EM algorithm
- ▷ Example: Mixture of normals with general mean and variance
- ▷ Properties and remarks

I.3.1 Introduction

- In this chapter, we will study how the parameters in a finite mixture distribution can be estimated using maximum likelihood estimation (MLE)
- We will consider the number g of components to be fixed (known)
- Let $Y_1, \ldots, Y_N$ be distributed as:

$$\begin{aligned} Y_i \sim\ & \pi_1 f_{i1}(y_i) + \pi_2 f_{i2}(y_i) + \ldots + \pi_g f_{ig}(y_i) \\ =\ & \sum_{j=1}^{g} \pi_j f_{ij}(y_i) \end{aligned}$$

- $f_{i1}(y_i), \ldots, f_{ig}(y_i)$ are the density functions of Y_i in the g components of the mixture

- Often, we have that

$$f_{ij}(y_i) \;=\; f_j(y_i), \quad \text{for all } j$$

  assuming that all Y_i follow the same distribution

- The index i now allows the Y_i to have different distributions, e.g., to include covariates (see later)

- As before, we allow the densities $f_{ij}(y_i)$ to depend on unknown parameters, which are combined in the vector $\boldsymbol{\theta}$.

- This will often be explicitly denoted as $f_{ij}(y_i|\boldsymbol{\theta})$

- Further, let $\boldsymbol{\pi}$ be the vector of component probabilities: $\boldsymbol{\pi}' = (\pi_1, \ldots, \pi_g)$

- The vector $\boldsymbol{\psi}$ is the vector containing all unknown parameters in the model: $\boldsymbol{\psi}' = (\boldsymbol{\pi}', \boldsymbol{\theta}')$

- The likelihood function equals:

$$L(\boldsymbol{\psi}|\boldsymbol{y}) = \prod_{i=1}^{N} \left\{ \sum_{j=1}^{g} \pi_j \; f_{ij}(y_i|\boldsymbol{\theta}) \right\}$$

where $\boldsymbol{y}' = (y_1, \ldots, y_N)$ is the vector containing all observed response values.

- The corresponding log-likelihood equals:

$$\ell(\boldsymbol{\psi}|\boldsymbol{y}) = \sum_{i=1}^{N} \ln \left\{ \sum_{j=1}^{g} \pi_j \; f_{ij}(y_i|\boldsymbol{\theta}) \right\}$$

- Maximizing $\ell(\boldsymbol{\psi}|\boldsymbol{y})$ with respect to $\boldsymbol{\psi}$ in general requires numerical iterative procedures

- Also, the analytic expression of $\ell(\boldsymbol{\psi}|\boldsymbol{y})$ suggests that numerical maximization will be far from straightforward

- For example, classical Newton-Raphson procedures would require calculation of first- and second-order derivatives of $\ell(\boldsymbol{\psi}|\boldsymbol{y})$.

- An alternative procedure, especially convenient for mixture models, is the EM algorithm, the **E**xpectation–**M**aximization algorithm

- EM is designed for MLE in situations with missing data

- Here, the underlying latent variable X, i.e., the component membership, will be considered missing.

I.3.2 EM Algorithm

3.2.1 Observed and Complete Data Likelihoods

- We define indicators Z_{ij}, $i = 1, \ldots, N$, $j = 1, \ldots, g$:

$$Z_{ij} = \begin{cases} 1 & \text{if observation } i \text{ belongs to component } j \\ \\ 0 & \text{otherwise} \end{cases}$$

- We then have that

$$P(Z_{ij} = 1) \;=\; \pi_j$$

- The joint density of Y_i and all associated Z_{ij} equals

$$f_i(y_i, Z_{i1} = z_{i1}, \ldots, Z_{ig} = z_{ig})$$

$$= f_i(y_i \mid Z_{i1} = z_{i1}, \ldots, Z_{ig} = z_{ig}) \times P(Z_{i1} = z_{i1}, \ldots, Z_{ig} = z_{ig})$$

$$= \left\{ \prod_{j=1}^{g} [f_{ij}(y_i|\boldsymbol{\theta})]^{z_{ij}} \right\} \times \left\{ \prod_{j=1}^{g} \pi_j^{z_{ij}} \right\} = \prod_{j=1}^{g} [\pi_j f_{ij}(y_i|\boldsymbol{\theta})]^{z_{ij}}$$

- The joint likelihood function for the **observed** measurements $\boldsymbol{y}$ and for the vector $\boldsymbol{z}$ of all **unobserved** z_{ij} therefore equals:

$$L(\boldsymbol{\psi}|\boldsymbol{y}, \boldsymbol{z}) = \prod_{i=1}^{N} \prod_{j=1}^{g} [\pi_j f_{ij}(y_i|\boldsymbol{\theta})]^{zij}$$

- The corresponding log-likelihood function equals:

$$\ell(\boldsymbol{\psi}|\boldsymbol{y},\boldsymbol{z}) = \sum_{i=1}^{N}\sum_{j=1}^{g} z_{ij}\left\{\ln \pi_j \; + \; \ln f_{ij}(y_i|\boldsymbol{\theta})\right\}$$

- Terminology:

$$\begin{aligned} L(\boldsymbol{\psi}|\boldsymbol{y},\boldsymbol{z}) &: \text{Complete data likelihood} \\ \ell(\boldsymbol{\psi}|\boldsymbol{y},\boldsymbol{z}) &: \text{Complete data log-likelihood} \\ L(\boldsymbol{\psi}|\boldsymbol{y}) &: \text{Observed data likelihood} \\ \ell(\boldsymbol{\psi}|\boldsymbol{y}) &: \text{Observed data log-likelihood} \end{aligned}$$

- Note that maximizing $\ell(\boldsymbol{\psi}|\boldsymbol{y},\boldsymbol{z})$ is much easier than maximizing the log-likelihood $\ell(\boldsymbol{\psi}|\boldsymbol{y})$ of the observed data only.

- However, the obtained estimates would depend on the unobserved indicators z_{ij}.
- Compromise: Maximize the expected value of $\ell(\boldsymbol{\psi}|\boldsymbol{y}, \boldsymbol{Z})$, i.e., maximize

$$\mathsf{E}\left[\ell(\boldsymbol{\psi}|\boldsymbol{y}, \boldsymbol{Z}) \mid \boldsymbol{y}\right]$$

- An intuitive explanation is that the 'missing' observations z_{ij} are replaced by their expected values.

3.2.2 EM Algorithm

- The EM algorithm acts iteratively, in the sense that, starting from a 'first guess estimate' (starting value) $\boldsymbol{\psi}^{(1)}$ for $\boldsymbol{\psi}$, a series of estimates $\boldsymbol{\psi}^{(t)}$ is constructed, which converges to the MLE $\widehat{\boldsymbol{\psi}}$ of $\boldsymbol{\psi}$:

$$\boldsymbol{\psi}^{(1)} \rightarrow \boldsymbol{\psi}^{(2)} \rightarrow \ldots \rightarrow \boldsymbol{\psi}^{(t)} \rightarrow \boldsymbol{\psi}^{(t+1)} \rightarrow \ldots \rightarrow \boldsymbol{\psi}^{(\infty)} = \widehat{\boldsymbol{\psi}}$$

- Given $\boldsymbol{\psi}^{(t)}$, the updated estimate $\boldsymbol{\psi}^{(t+1)}$ is obtained through one E step and one M step.

- **E step:** Calculation of

$$\boxed{Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(t)}) = \mathsf{E}\left[\ell(\boldsymbol{\psi}|\boldsymbol{y}, \boldsymbol{Z}) \mid \boldsymbol{y}, \boldsymbol{\psi}^{(t)}\right]}$$

- **M step:** Maximize $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(t)})$ with respect to $\boldsymbol{\psi}$ to obtain the updated estimate $\boldsymbol{\psi}^{(t+1)}$.

- The procedure keeps iterating between the E step and the M step until convergence is attained, i.e., until

$$|\ell(\boldsymbol{\psi}^{(t+1)}|y) \; - \; \ell(\boldsymbol{\psi}^{(t)}|y)| \; < \; \varepsilon,$$

  for some small, pre-specified, $\varepsilon > 0$.

3.2.3 The E Step

- $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(t)})$ is obtained from:

$$\begin{aligned} Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(t)}) &= \mathsf{E}\left[\ell(\boldsymbol{\psi}|\boldsymbol{y},\boldsymbol{Z}) \mid \boldsymbol{y},\boldsymbol{\psi}^{(t)}\right] \\ &= \mathsf{E}\left[\left\{\sum_{i=1}^{N}\sum_{j=1}^{g} Z_{ij}\left[\ln \pi_j + \ln f_{ij}(y_i|\boldsymbol{\theta})\right]\right\} \;\middle|\; \boldsymbol{y},\boldsymbol{\psi}^{(t)}\right] \\ &= \sum_{i=1}^{N}\sum_{j=1}^{g} \mathsf{E}\left[Z_{ij} \mid \boldsymbol{y},\boldsymbol{\psi}^{(t)}\right]\left[\ln \pi_j + \ln f_{ij}(y_i|\boldsymbol{\theta})\right] \end{aligned}$$

- Hence, the E step only requires calculation of

$$\mathsf{E}\left[Z_{ij} \mid y_i,\boldsymbol{\psi}^{(t)}\right] = P\left(Z_{ij}=1 \mid y_i,\boldsymbol{\psi}^{(t)}\right) = \left.\frac{f_i(y_i \mid Z_{ij}=1)\; P(Z_{ij}=1)}{f_i(y_i|\boldsymbol{\theta})}\right|_{\boldsymbol{\psi}^{(t)}}$$

$$= \left.\frac{\pi_j\; f_{ij}(y_i|\boldsymbol{\theta})}{\sum_j \pi_j f_{ij}(y_i|\boldsymbol{\theta})}\right|_{\boldsymbol{\psi}^{(t)}} = \pi_{ij}(\boldsymbol{\psi}^{(t)})$$

- $\pi_{ij}(\boldsymbol{\psi}^{(t)})$ is the **posterior** probability for observation i to belong to the jth component of the mixture

- From now on, π_j will be called the **prior** probability for observation i to belong to the jth component of the mixture

- The E step reduces to calculating all posterior probabilities $\pi_{ij}(\boldsymbol{\psi}^{(t)})$, $i = 1, \ldots, N$, $j = 1, \ldots, g$.

3.2.4 The M Step

- The updated estimate $\boldsymbol{\psi}^{(t+1)}$ is obtained from maximizing

$$Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(t)}) = \sum_{i=1}^{N}\sum_{j=1}^{g}\pi_{ij}(\boldsymbol{\psi}^{(t)})\left[\ln\pi_j + \ln f_{ij}(y_i|\boldsymbol{\theta})\right]$$

  with respect to $\boldsymbol{\psi}' = (\boldsymbol{\pi}', \boldsymbol{\theta}')$.

- We first maximize with respect to $\boldsymbol{\pi}$:
  - ▷ This requires maximization of

$$\sum_{i=1}^{N}\sum_{j=1}^{g}\pi_{ij}(\boldsymbol{\psi}^{(t)})\ln\pi_j = \sum_{i=1}^{N}\sum_{j=1}^{g-1}\pi_{ij}(\boldsymbol{\psi}^{(t)})\ln\pi_j + \sum_{i=1}^{N}\pi_{ig}(\boldsymbol{\psi}^{(t)})\ln\left[1-\sum_{j=1}^{g-1}\pi_j\right]$$

    with respect to $\pi_1, \ldots, \pi_{g-1}$

▷ We set all first-order derivatives equal to zero:

$$\frac{\partial}{\partial \pi_j} = 0 \quad \Leftrightarrow \quad \sum_{i=1}^{N} \frac{\pi_{ij}(\boldsymbol{\psi}^{(t)})}{\pi_j^{(t+1)}} = \sum_{i=1}^{N} \frac{\pi_{ig}(\boldsymbol{\psi}^{(t)})}{\pi_g^{(t+1)}} \quad \Leftrightarrow \quad \frac{\pi_j^{(t+1)}}{\pi_g^{(t+1)}} = \frac{\sum_{i=1}^{N} \pi_{ij}(\boldsymbol{\psi}^{(t)})}{\sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\psi}^{(t)})}$$

▷ This implies that

$$1 = \sum_{j=1}^{g} \pi_j^{(t+1)} = \sum_{j=1}^{g} \frac{\pi_g^{(t+1)} \sum_{i=1}^{N} \pi_{ij}(\boldsymbol{\psi}^{(t)})}{\sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\psi}^{(t)})}$$

$$= \frac{\pi_g^{(t+1)} \sum_{i=1}^{N} \overbrace{\sum_{j=1}^{g} \pi_{ij}(\boldsymbol{\psi}^{(t)})}^{1}}{\sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\psi}^{(t)})} = \frac{N\ \pi_g^{(t+1)}}{\sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\psi}^{(t)})}$$

▷ Hence, $\pi_g^{(t+1)}$ is given by

$$\pi_g^{(t+1)} = \frac{\sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\psi}^{(t)})}{N}$$

▷ It now also follows that all $\pi_j^{(t+1)}$ are given by

$$\boxed{\pi_j^{(t+1)} = \frac{\sum_{i=1}^{N} \pi_{ij}(\boldsymbol{\psi}^{(t)})}{N}}$$

▷ The updated mixture component probabilities are the average posterior probabilities.

- Maximization with respect to $\boldsymbol{\theta}$ requires maximization of

$$\sum_{i=1}^{N} \sum_{j=1}^{g} \pi_{ij}(\boldsymbol{\psi}^{(t)}) \ln f_{ij}(y_i|\boldsymbol{\theta})$$

- In simple examples, this can be done analytically
- In general, however, this cannot be done analytically, and a classical maximization procedure, such as Newton-Raphson, is used.
- In such cases, the EM algorithm is **double iterative**, which can have serious consequences on the computation times.

I.3.3 Example: Normals With General Mean and Variance

$$Y_i \sim \sum_{j=1}^{g} \pi_j N(\mu_j, \sigma_j^2)$$

$$\boldsymbol{\theta} = (\mu_1, \ldots, \mu_g, \sigma_1^2, \ldots, \sigma_g^2)$$

- The log-likelihood corresponding to the above model is

$$\ell(\boldsymbol{\psi}|\boldsymbol{y}) = \sum_{i=1}^{N} \ln \left\{ \sum_{j=1}^{g} \pi_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{1}{2\sigma_j^2} (y_i - \mu_j)^2 \right] \right\}$$

- This can be re-written as

$$\ell(\boldsymbol{\psi}|\boldsymbol{y}) = \sum_{i=2}^{N} \ln \left\{ \sum_{j=2}^{g} \pi_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{1}{2\sigma_j^2} (y_i - \mu_j)^2 \right] + \pi_1 \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left[-\frac{1}{2\sigma_1^2} (y_i - \mu_1)^2 \right] \right\}$$

$$+ \ln \left\{ \sum_{j=2}^{g} \pi_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{1}{2\sigma_j^2} (y_1 - \mu_j)^2 \right] + \pi_1 \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left[-\frac{1}{2\sigma_1^2} (y_1 - \mu_1)^2 \right] \right\}$$

- Taking μ_1 equal to y_1, this becomes

$$\ell(\boldsymbol{\psi}|\boldsymbol{y}) = \sum_{i=2}^{N} \ln \left\{ \sum_{j=2}^{g} \pi_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{1}{2\sigma_j^2} (y_i - \mu_j)^2 \right] + \pi_1 \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left[-\frac{1}{2\sigma_1^2} (y_i - y_1)^2 \right] \right\}$$

$$+ \ln \left\{ \sum_{j=2}^{g} \pi_j \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left[-\frac{1}{2\sigma_j^2} (y_1 - \mu_j)^2 \right] + \pi_1 \frac{1}{\sqrt{2\pi\sigma_1^2}} \right\}$$

- The above expression converges to $+\infty$ when σ_1^2 approaches zero
- Hence, this mixture model leads to infinite likelihoods.
- This can only be solved by keeping the component variances away from zero
- One way to do so is by assuming all the variances to be equal, i.e., $\sigma_j^2 = \sigma^2$
- Indeed, $\sigma^2 = 0$ would then result in a discrete marginal mixture distribution with g components, which is not possible as soon as the number of distinct data points is larger than g.

- Sometimes a well fitting mixture of normals with general mean and variance can be obtained after convergence to a **local maximum** of the likelihood.

- However, since the solution is then **NOT** MLE, inference does not follow from standard likelihood theory.

- We will therefore only consider mixtures of normal distributions with common variance.

I.3.4 Properties and Remarks

3.4.1 Identifiability

- Consider the following mixture of 3 Poissons:

$$Y \sim \pi_1 \text{Poisson}(\lambda_1) + \pi_2 \text{Poisson}(\lambda_2) + \pi_3 \text{Poisson}(\lambda_3)$$

- The parameter vector $\boldsymbol{\psi}$ then equals: $\boldsymbol{\psi}' = (\pi_1, \pi_2, \pi_3, \lambda_1, \lambda_2, \lambda_3)$

- Note that likelihood value for

$$\boldsymbol{\psi}' = (0.1, 0.7, 0.2, 1, 5, 7)$$

  is exactly the same as the likelihood value for

$$\boldsymbol{\psi}' = (0.1, 0.2, 0.7, 1, 7, 5)$$

- In fact, any permutation of the elements in

$$\{(\lambda_1, \pi_1), (\lambda_2, \pi_2), (\lambda_3, \pi_3)\}$$

  leads to the same likelihood value.

- In general, for g components, there are $g!$ possible permutations of the mixture components, all yielding the same likelihood value, i.e., the likelihood has at least $g!$ local maxima with the same likelihood value.

- This shows that, for finite mixtures of distributions of the same parametric family (e.g., mixture of Normals, Binomials, Poissons, ...), the vector $\boldsymbol{\psi}$ is not uniquely identified.

- One way to make $\boldsymbol{\psi}$ identifiable is by ordering the mixture components according to the corresponding component probabilities, e.g.,

$$\pi_1 \geq \pi_2 \geq \ldots \geq \pi_g$$

3.4.2 Monotonicity Property of EM Algorithm

- It can be shown that an EM step cannot decrease the likelihood value $\ell(\boldsymbol{\psi}|y)$, i.e.,

$$\ell(\boldsymbol{\psi}^{(t+1)}|y) \;\geq\; \ell(\boldsymbol{\psi}^{(t)}|y), \qquad \text{for all } t$$

- This is called the monotonicity property of the EM algorithm
- It guarantees convergence of the iterative procedure
- Note that this does not guarantee convergence to a **global** maximum

3.4.3 Existence of Local Maxima

- Apart from the local maxima resulting from the non-identifiability problem, there may be local maxima yielding different likelihood values

- Example from Böhning (p.66). Mixture of two normals with common variance:

Setting	p_1	λ_1	λ_2	l	σ^2
initial values	0.9	0	−7		1.
at convergence	0.9907	−1.6194	−5.8535	−680.6718	0.7305
initial values	0.5	0	6		1.
at convergence	0.9962	−1.6472	6.8700	−695.1904	0.8270
initial values	0.5	−0.5	0.5		1.
at convergence	0.8355	−1.6749	−1.5034	−687.5996	0.8232

- Obviously, the second and third set of estimates correspond to local maxima, as the first set of estimates yields a higher log-likelihood value

- This suggests that multiple sets of starting values should be used in practice

3.4.4 Convergence to a Ridge

- Consider fitting the mixture

$$Y \sim \pi N(\mu_1, \sigma^2) + (1-\pi) N(\mu_2, \sigma^2)$$

of 2 normals with common variance, while the true distribution of Y is a single normal, i.e.,

$$Y \sim N(\mu, \sigma^2)$$

- We then have that the likelihood is maximized on a ridge of parameter values:

$$\mu_1 = \mu_2 \qquad \text{or} \qquad \pi = 0 \qquad \text{or} \qquad \pi = 1$$

- The EM algorithm is capable of converging to some particular point on that ridge.

- This is not the case for many other, more classical, maximization algorithms.

- This is why the EM algorithm is especially convenient for mixture models

3.4.5 Convergence Rate

- Although the monotonicity property guarantees convergence, this convergence can be painfully slow

- Example from Böhning (p.63). Mixture of three **known** distributions:

Iteration	p_1	p_2	p_3
1	1/3	1/3	1/3
10	0.1804	0.3272	0.4966
100	0.2043	0.0994	0.6968
1000	0.2103	0.0426	0.7471
10000	0.2102	0.0424	0.7473
∞	0.2102	0.0424	0.7473

- With badly selected starting values, such slow convergence can lead to long computation times, especially when the M step in the algorithm requires iterative maximization (i.e., when the EM is double iterative).

Chapter I.4
Fitting Finite Mixtures in R and SAS

- ▷ R-package CAMAN
- ▷ SAS procedure FMM
- ▷ Comparison of R with SAS
- ▷ Example: Child data
- ▷ Example: SIDS data

I.4.1 R-package CAMAN

- Developed by Peter Schlattmann, Johannes Hoehne, and Maryna Verba, based on C.A.MAN (D. Böhning & P. Schlattmann)
- Contains several functions for mixture analyses
- Function 'mixalg.EM' is based on EM algorithm for fitting finite mixtures with fixed number of components
- Let's fit a 4 component normal mixture to the snapper data
- Loading the data:

```
> load("c:/analysis/mixtureR/snapper.rdata")
```

- Data structure:

```
> snapper

   length frequency
1   2.875         6
2   3.125         7
3   3.375         9
.....................
40 12.625         1
```

- Fitting normal mixture with 4 components:

```
> em<-mixalg.EM(obs="length", weights="frequency", family="gaussian",
                data=snapper, p=c(0.10,0.50,0.30,0.10), t=c(3,5,8,10))
```

- 'obs=' specifies the outcome

- 'weights=' specifies a replication factor (weight)

- 'family=' specifies the distribution in each component
- 'data=' specifies the data set
- 'p=' specifies starting values for the component probabilities, and indirectly the number of components
- The procedure automatically rescales the starting values in 'p='
- Hence the following specifications are equivalent:

```
p=c(0.10,0.50,0.30,0.10)
p=c(10,50,30,10)
```

- 't=' specifies starting values for the component locations, and indirectly the number of components

- Generated output:

```
> em

Computer Assisted Mixture Analysis:

Data consists of 256 observations (rows).
The Mixture Analysis identified 4 components of a gaussian distribution:

DETAILS:
           p      mean
1 0.11755367  3.432325
2 0.53355806  5.319268
3 0.27207539  7.601072
4 0.07681288 10.334596
component variance: 0.447414325225651

Log-Likelihood: -505.7188     BIC: 1050.254
```

- Fitted model:

$$Y|\mu \sim N(\mu, 0.67^2) \qquad \mu \sim \begin{pmatrix} 3.43 & 5.32 & 7.60 & 10.33 \\ 0.12 & 0.53 & 0.27 & 0.08 \end{pmatrix}$$

I.4.2 SAS Procedure FMM

- The same 4-component normal mixture can be fitted to the snapper data, using the following syntax:

```
proc fmm data=snapper;
model length = / dist=gaussian equate=scale k=4
           parms(1 0.5,5 0.5, 9 0.5, 13 0.5);
probmodel  / parms(0,1.6,1.1);
freq frequency;
run;
```

- 'dist=' specifies the distribution in each component
- 'k=' specifies the number of components in the mixture
- 'parms(...)' specifies starting values for all parameters in the component densities

- Here, this implies specification of the mean and variance in each of the normal components of the mixture

- 'equate=' specifies parameter constraints across the components. In our model, we restricted all component variances to be equal

- The location parameters of mixture components are specified using default link functions which can be changed using a 'link=' option:

Distribution	Default link	Parameterisation
Normal $N(\mu, \sigma^2)$	identity	μ
Bernoulli $B(p)$	logit	$\ln[p/(1-p)]$
Binomial $B(n, p)$	logit	$\ln[p/(1-p)]$
Exponential $Exp(\lambda)$	log	$\ln(\lambda)$
Poisson $P(\lambda)$	log	$\ln(\lambda)$
Lognormal $LN(\mu, \sigma^2)$	identity	μ

- The PROBMODEL statement is used to specify starting values for the component probabilities using the logit link function, which can be changed using a 'link=' option

- For our 4-component model, this implies specification of the following starting values:

$$\begin{pmatrix} \pi_1 = 0.1 \\ \pi_2 = 0.5 \\ \pi_3 = 0.3 \\ \pi_4 = 0.1 \end{pmatrix} \longrightarrow \begin{pmatrix} \ln[\pi_1/\pi_4] = 0 \\ \ln[\pi_2/\pi_4] = 1.6 \\ \ln[\pi_3/\pi_4] = 1.1 \\ \ln[\pi_4/\pi_4] = 0 \end{pmatrix}$$

- Table with fit statistics (selection):

-2 Log Likelihood	1011.4
Effective Parameters	8
Effective Components	4

- Estimates for mixture components:

Parameter Estimates for 'Normal' Model

Component	Parameter	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	3.4323	0.1665	20.62	<.0001
2	Intercept	5.3193	0.07456	71.34	<.0001
3	Intercept	7.6011	0.1119	67.94	<.0001
4	Intercept	10.3346	0.1887	54.76	<.0001
1	Variance	0.4474	0.06084		
2	Variance	0.4474	0.06084		
3	Variance	0.4474	0.06084		
4	Variance	0.4474	0.06084		

- Due to the 'equate=scale' option, all component variances are equal

- Estimates for component probabilities:

Parameter Estimates for Mixing Probabilities

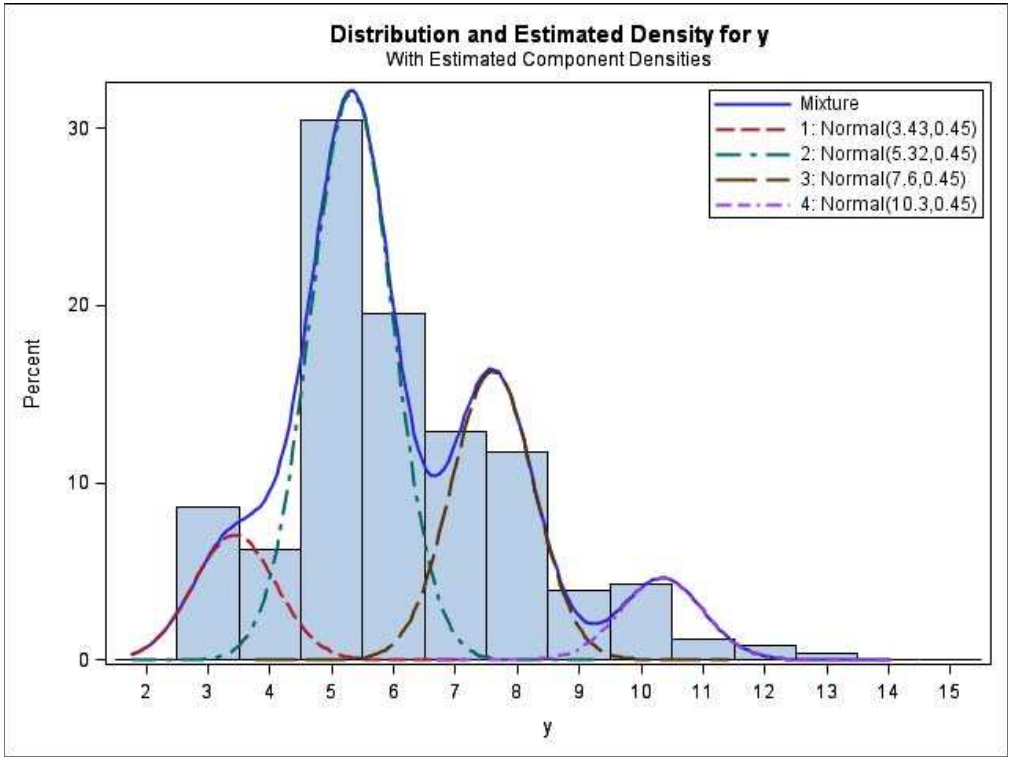
		Linked Scale				
Component	Parameter	Estimate	Standard Error	z Value	Pr > |z|	Probability
1	Probability	0.4255	0.3286	1.30	0.1953	0.1176
2	Probability	1.9382	0.2640	7.34	<.0001	0.5336
3	Probability	1.2647	0.2820	4.48	<.0001	0.2721

- Hence the fitted model is given by:

$$Y|\mu \sim N(\mu, 0.67^2) \qquad \mu \sim \begin{pmatrix} 3.43 & 5.32 & 7.60 & 10.33 \\ 0.12 & 0.53 & 0.27 & 0.08 \end{pmatrix}$$

- SAS easily allows plotting the fitted mixture density with individual component densities:

```
ods graphics on;
proc fmm data=snapper plots=density(bins=15);
model length = / dist=gaussian equate=scale k=4
           parms(3 0.5,5 0.5, 8 0.5, 10 0.5) ;
probmodel  / parms(0,1.6,1.1);
freq frequency;
run;
ods graphics off;
```



- Omission of the option 'equate=scale' requires fitting of a normal mixture with component-specific means as well as variances:

```
ods graphics on;
proc fmm data=snapper plots=density(bins=15);
model length = / dist=gaussian k=4 parms(3 0.5,5 0.5, 8 0.5, 10 0.5) ;
probmodel  / parms(0,1.6,1.1);
freq frequency;
run;
ods graphics off;
```

- Table with fit statistics (selection):

Fit Statistics

-2 Log Likelihood	977.2
Effective Parameters	11
Effective Components	4

- As expected the likelihood is larger ($\ell\ell = -488.6$ instead of $\ell\ell = -505.7$ before)

- Note that, since the likelihood is unbounded for this model, the estimation procedure converged to a local maximum

- Hence the reported estimates are not MLE's and the likelihood value cannot be used for formal model comparison based on LR's

- Estimates for mixture components:

Parameter Estimates for 'Normal' Model

Component	Parameter	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	3.2130	0.06178	52.00	<.0001
2	Intercept	5.2596	0.07330	71.76	<.0001
3	Intercept	7.4428	0.1110	67.05	<.0001
4	Intercept	8.6186	1.2434	6.93	<.0001
1	Variance	0.06314	0.02432		
2	Variance	0.3937	0.08644		
3	Variance	0.2060	0.1270		
4	Variance	3.0934	1.7754		

- Estimates for component probabilities:

Parameter Estimates for Mixing Probabilities

		Linked Scale				
Component	Parameter	Estimate	Standard Error	z Value	Pr > |z|	Probability
1	Probability	-0.7376	0.6574	-1.12	0.2619	0.0948
2	Probability	0.9956	0.7092	1.40	0.1604	0.5365
3	Probability	-0.1513	1.0026	-0.15	0.8801	0.1704

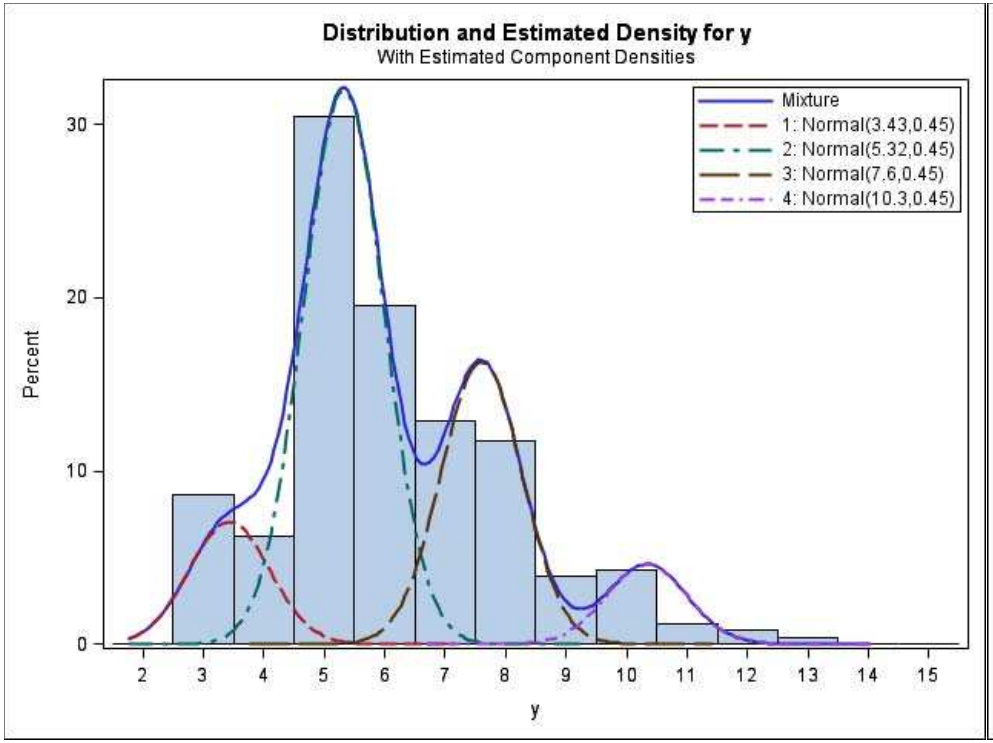
- Fitted model:

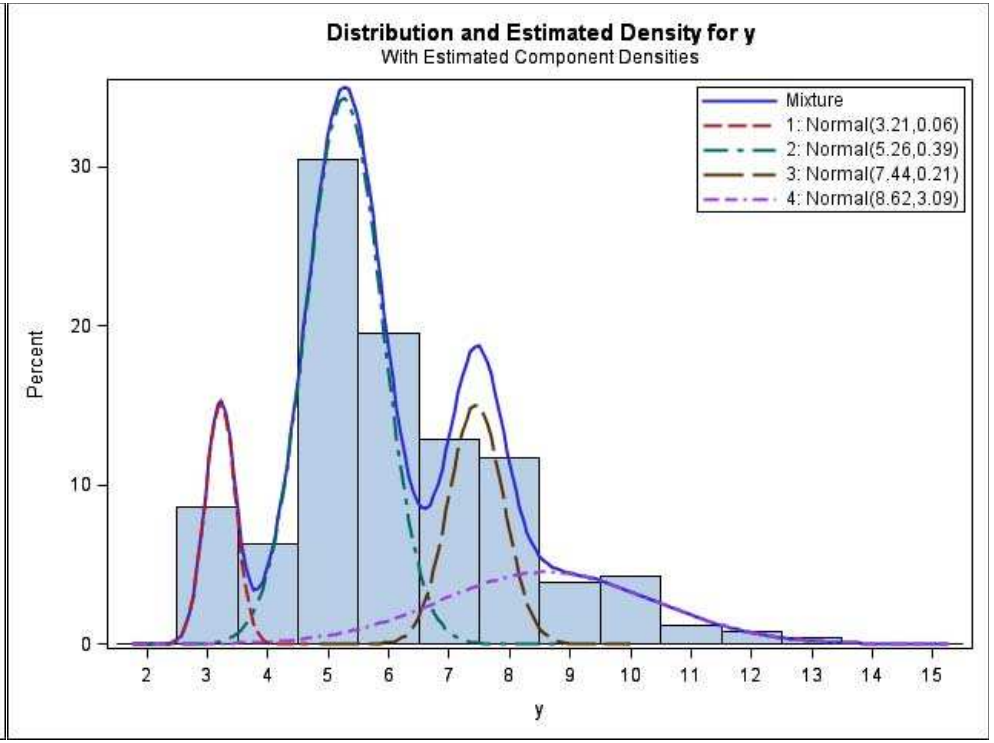
$$Y \sim 0.09N(3.21, 0.25^2) + 0.54N(5.26, 0.63^2) + 0.17N(7.44, 0.45^2) + 0.20N(8.62, 1.76^2)$$

while the previous model was equal to

$$Y \sim 0.12N(3.43, 0.67^2) + 0.53N(5.32, 0.67^2) + 0.27N(7.60, 0.67^2) + 0.08N(10.33, 0.67^2)$$

- Comparison of both fitted densities:





- The model with component-specific variances results in a less smooth density but seems to better capture the long right tail

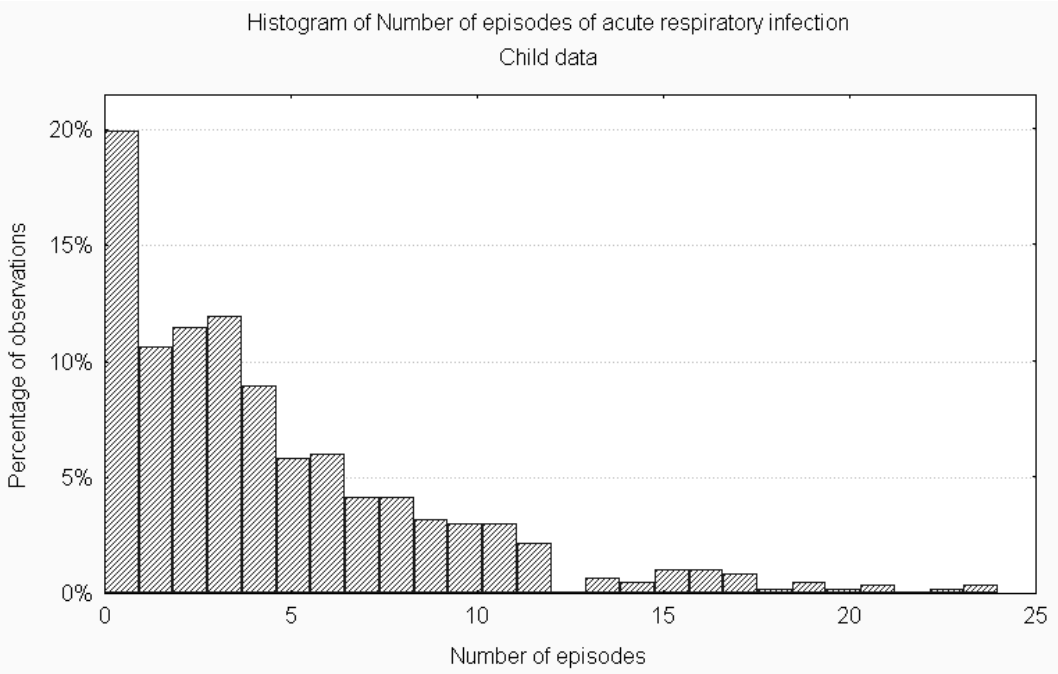
I.4.3 Comparison of R with SAS

- Advantages of SAS procedure FMM:
  - ▷ Many more distributions available for the mixture components
  - ▷ Component densities not restricted to one parametric family (see later)
  - ▷ More flexibility to add restrictions to parameters or to set parameters equal to pre-specified values (see later)
  - ▷ More flexibility to include covariates in components and/or component probabilities (see later)

- Advantages of R package CAMAN:
  - ▷ Based on EM which is more stable than optimization in SAS
  - ▷ Less sensitive to starting values
  - ▷ Starting values on original scale (no link functions)
  - ▷ Estimation of g is possible (see later)

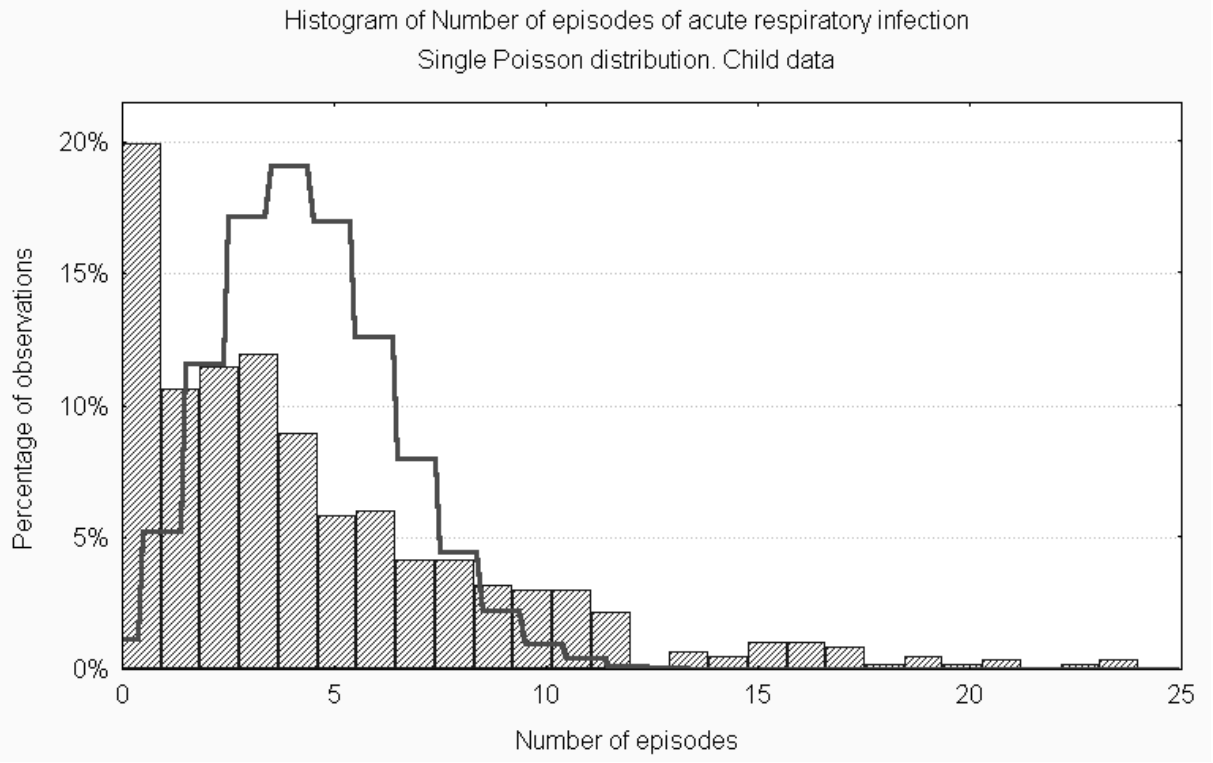
I.4.4 Example: Child Data

- Data set child.dat, on 602 pre-school children in north-eastern Thailand
- The response of interest is the number of episodes of acute respiratory infection (fever, cough, running nose,...), recorded within a 3-year period, with histogram:



- The Poisson distribution is often used in practice for describing count data

- Since the average number of episodes equals 4.45, we first try to approximate the above histogram by the Poisson(4.45):



- Obviously, a single Poisson distribution cannot account for the large percentage of children with no or almost no episodes

- This can also be observed from comparing the sample mean with the sample variance:

$$\overline{y} = 4.45 \quad \ll \quad 20.45 = s_y^2$$

- Hence, there is more variability in the data than what can be explained from a single Poisson distribution

- This phenomenon is called **overdispersion**

- One way to take into account the overdispersion is modelling underlying heterogeneity using a finite mixture

- A 4-component Poisson mixture will be used.

- R code:

```
em<-mixalg.EM(obs="counts", weights="frequency", family="poisson",
              data=child,t=c(0.5,3,10,15), p=c(0.25,0.25,0.25,0.25))
```

- SAS code:

```
proc fmm data=child;
model counts = / dist=poisson k=4 parms(-0.7, 1.1, 2.3, 2.7);
probmodel  / parms(0,0,0);
freq frequency;
run;
```

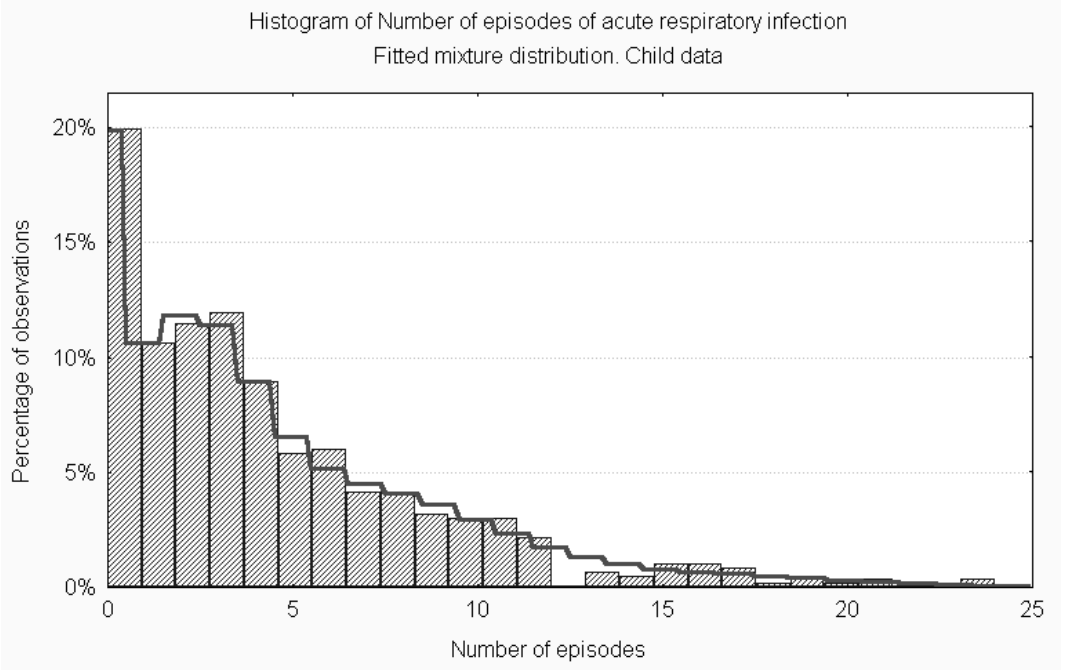
- Note that the parameters in the Poisson densities are now specified on a logarithmic scale:

$$(0.5, 3, 10, 15) \quad \rightarrow \quad (\ln(0.5), \ln(3), \ln(10), \ln(15))$$

- Fitted model ($ll = -1553.81$):

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} 0.143 & 2.817 & 8.164 & 16.156 \\ 0.197 & 0.480 & 0.270 & 0.053 \end{pmatrix}$$

- Graphical representation:



- As derived earlier, the mean and variance of the obtained mixture can be calculated as

$$\mathrm{E}(Y) = \sum_j \pi_j \lambda_j = 4.45$$

$$\mathrm{Var}(Y) = \sum_j \pi_j \lambda_j^2 - \left(\sum_j \pi_j \lambda_j\right)^2 + \sum_j \pi_j \lambda_j = 20.44$$

  which are very close to the observed average ($\overline{y} = 4.45$) and observed variance ($s_y^2 = 20.45$), illustrating that the mixture has taken account of the overdispersion in the data.

- Biological interpretation: Latent variable represents the health status:

Component	λ_j	π_j	Interpretation
1	0.143	0.197	almost always healthy
2	2.817	0.480	normal
3	8.164	0.270	above normal
4	16.156	0.053	high risk for infection

- Note that the first component is a Poisson distribution with mean $\lambda = 0.143$, which assigns probability 0.867 to the value $Y = 0$.

- One may wonder how much worse the model would be if the first component would be fixed at $\lambda = 0$, representing subjects who never experience acute respiratory infections

- In SAS, this can easily be achieved by mixing a degenerate distribution at 0 with a finite mixture of 3 Poisson distributions.

- SAS code:

```
proc fmm data=child;
model counts = / dist=constant(0) k=1;
model   + / dist=poisson k=3 parms(1.1, 2.3, 2.7);
probmodel  / parms(0,0,0);
freq frequency;
run;
```

- Fitted model ($ll = -1554.4$):

$$\begin{aligned} Y \sim\ & 0.161\ \mathbf{1_0} + 0.496\ \text{Poisson}(2.572) \\ & +0.286\ \text{Poisson}(7.905) + 0.057\ \text{Poisson}(15.960) \end{aligned}$$

  rather than the previous model

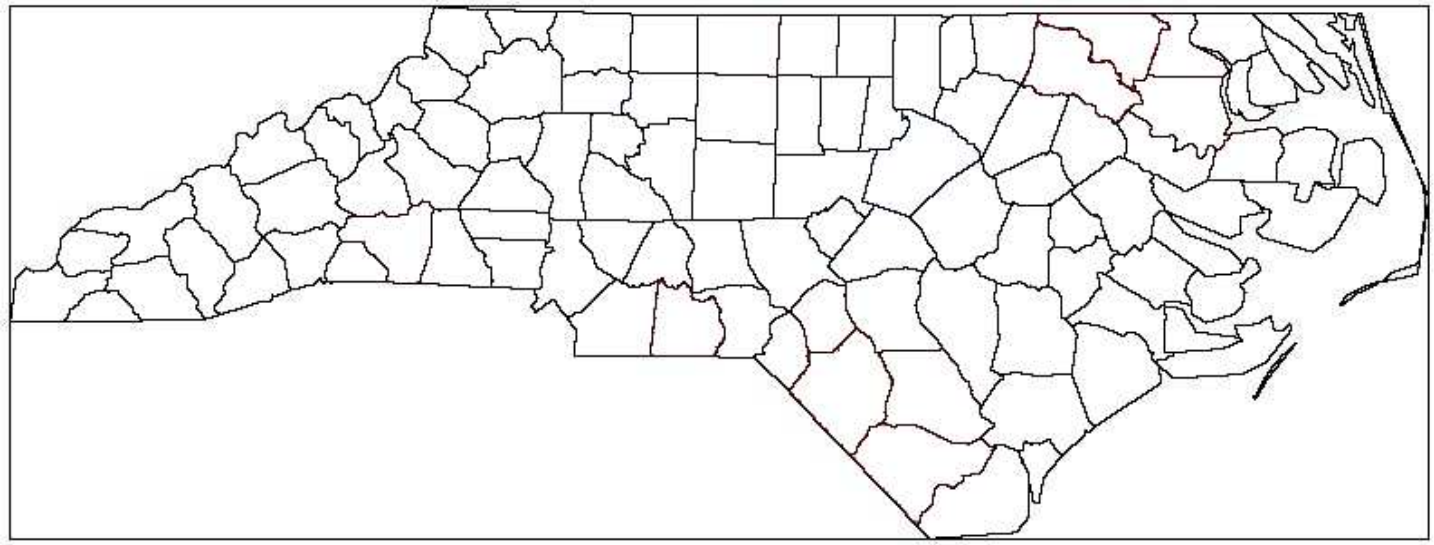
$$\begin{aligned} Y \sim\ & 0.197\ \text{Poisson}(0.143) + 0.480\ \text{Poisson}(2.817) \\ & +0.270\ \text{Poisson}(8.164) + 0.053\ \text{Poisson}(16.156) \end{aligned}$$

- The model is only slightly worse in terms of likelihood:
  $ll = -1554.4$ versus $ll = -1553.8$

- Note that a classical LR test does not apply due to a boundary null-hypothesis
  $H_0 : \lambda_1 = 0$

I.4.5 Example: SIDS Data

- SIDS: **S**udden **I**nfant **D**eath **S**yndrome
- Numbers of reported SIDS cases in 100 North Carolina counties, during the period 1974–1978:



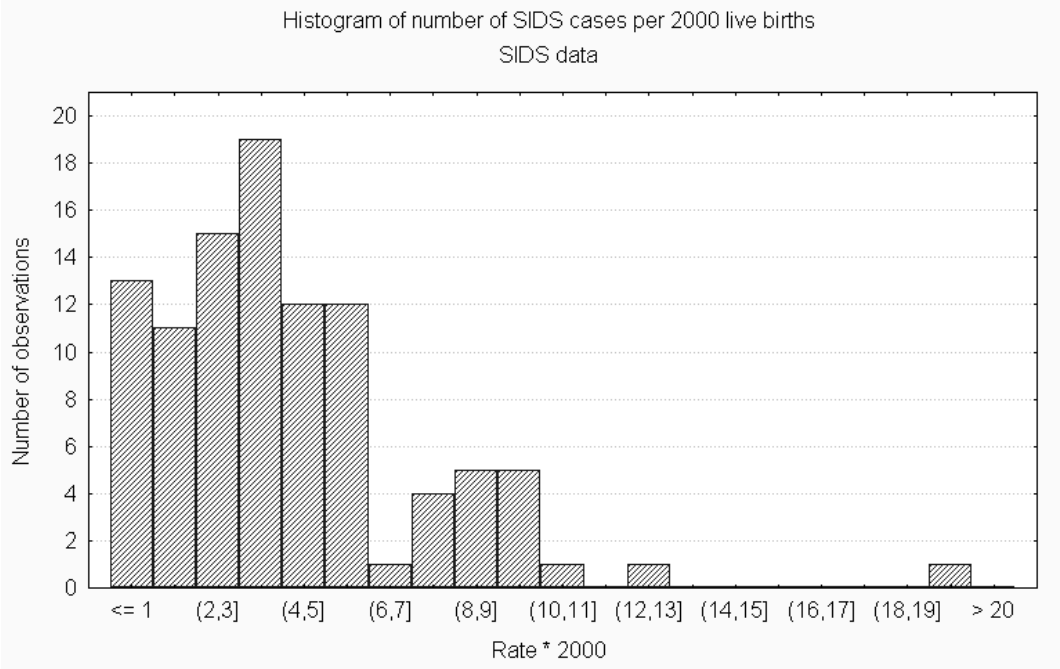
- As the counties are not of the same size, we need to correct for the number of live-births in each county

- Data set sids.dat

- Data structure:

County	Y_i	n_i	$R_i = Y_i/n_i$
1	13	4672	0.00278
2	0	487	0.00000
3	15	1570	0.00955
⋮	⋮	⋮	⋮
100	16	14484	0.00110

- Terminology:
  - ▷ Observed counts Y_i, $i = 1, \ldots, 100$
  - ▷ Number n_i of 'exposed' children
  - ▷ Rates: $R_i = \frac{Y_i}{n_i}$
- Histogram of the 100 rates:



- The histogram suggests the presence of heterogeneity among the counties
- This suggests that the counties are clustered with respect to their SIDS risk
- A 3-component mixture of Binomial distributions will be used
- R code:

```
em<-mixalg.EM(obs="frequency", pop.at.risk="nrisk", family="binomial",
              data=sids, t=c(0.001,0.002,0.005), p=c(1,1,1))
```

- SAS code:

```
proc fmm data=sids ;
model frequency/nrisk = / dist=binomial k=3 parms(-6.9,-6.2,-5.3);
probmodel  / parms(0,0);
run;
```

- Note that the parameters in the Binomial densities are now specified on a logit scale:

$$(0.001, 0.002, 0.005) \quad \rightarrow \quad (\ln(0.001/0.999), \ln(0.002/0.998), \ln(0.005/0.995))$$

- Fitted model ($ll = -233.70$):

$$Y_i|p \sim \text{Binomial}(n_i, p) \qquad p \sim \begin{pmatrix} 0.0013 & 0.0021 & 0.0042 \\ 0.33 & 0.53 & 0.14 \end{pmatrix}$$

- Note that the mixture distribution cannot be super-imposed on the histogram of the rates (as in previous examples) since the mixture is the distribution of the counts Y_i (all having different distributions) rather than the distribution of the rates R_i

- Later, it will be shown how the above mixture can be used to create so-called **disease maps** where counties are grouped based on their associated SIDS risk.

- Because a Binomial(n, p) distribution with large n and small p can be well approximated by a Poisson(np) distribution, disease rates are often modelled using Poisson models

- In R, the model can be specified as:

```
em<-mixalg.EM(obs="frequency", pop.at.risk="nrisk", family="poisson",
              data=sids,  t=c(0.001, 0.002, 0.005), p=c(1, 1, 1))
```

- Since SAS models the mean of a Poisson distribution on a log scale, an offset needs to be used:

$$E(Y) \;=\; np \;=\; \exp[\ln(n) + \ln(p)]$$

- $\ln(n)$ is the offset that needs to be added to the linear predictor that models $\ln(p)$

- Furthermore, starting values for p in the various mixture components need to be specified on a logarithmic scale, rather than the logit scale as before:

$$(0.001, 0.002, 0.005) \quad \longrightarrow \quad (\ln(0.001), \ln(0.002), \ln(0.005))$$

- However, due to the small values of p, $\ln[p/(1-p)] \approx \ln(p)$

- SAS code:

```
data sids;
set sids;
offset=log(nrisk);
run;
proc fmm data=sids ;
model frequency = / dist=poisson offset=offset k=3 parms(-6.9,-6.2,-5.3);
probmodel  / parms(0,0);
run;
```

- Fitted model ($ll = -234.41$):

$$Y_i|p \sim \text{Poisson}(n_i p) \qquad p \sim \begin{pmatrix} 0.0013 & 0.0021 & 0.0042 \\ 0.33 & 0.53 & 0.14 \end{pmatrix}$$

which yields the same model for the risk parameter p as under the original Binomial model ($ll = -233.70$):

$$Y_i|p \sim \text{Binomial}(n_i, p) \qquad p \sim \begin{pmatrix} 0.0013 & 0.0021 & 0.0042 \\ 0.33 & 0.53 & 0.14 \end{pmatrix}$$

Chapter I.5
Inference for Number of Support Points

- ▷ Introduction
- ▷ Mixture of 2 known distributions
- ▷ Some results
- ▷ Conclusions

I.5.1 Introduction

- All examples discussed so far acted conditional on the number g of mixture components
- In this chapter, we will take a closer look at the selection for g
- An obvious approach is to fit models with increasing g
- As example, we continue the analysis of the snapper data (data set snapper.dat), and we fit mixtures of normals with common variance, for a variety of values g:

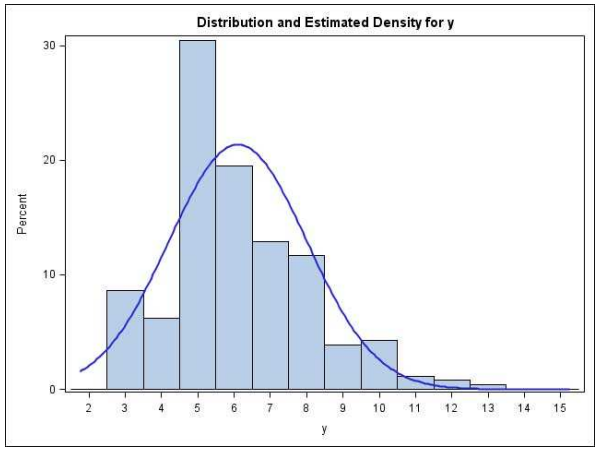
$$Y_i \sim \sum_{j=1}^{g} \pi_j N(\mu_j, \sigma^2)$$

- Summary of results:

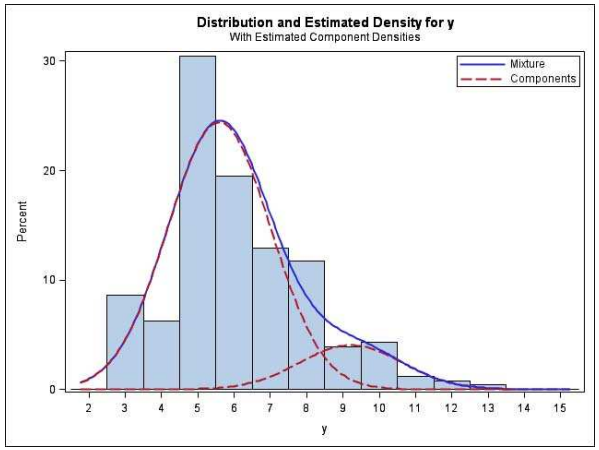
g	$\begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$	σ^2	ℓ
1	$\begin{pmatrix} 6.10 \\ 1 \end{pmatrix}$	3.60	-527.2
2	$\begin{pmatrix} 5.59 & 9.22 \\ 0.86 & 0.14 \end{pmatrix}$	2.00	-515.65
3	$\begin{pmatrix} 5.05 & 7.60 & 10.49 \\ 0.65 & 0.29 & 0.06 \end{pmatrix}$	1.11	-512.00
4	$\begin{pmatrix} 3.43 & 5.32 & 7.60 & 10.33 \\ 0.12 & 0.53 & 0.27 & 0.08 \end{pmatrix}$	0.45	-505.72
5	$\begin{pmatrix} 3.40 & 5.31 & 7.50 & 9.68 & 11.99 \\ 0.12 & 0.52 & 0.26 & 0.08 & 0.02 \end{pmatrix}$	0.32	-493.50
6	$\begin{pmatrix} 3.39 & 5.30 & 8.37 & 7.34 & 9.85 & 12.04 \\ 0.12 & 0.51 & 0.05 & 0.23 & 0.06 & 0.03 \end{pmatrix}$	0.29	-492.65

- Graphical representation:

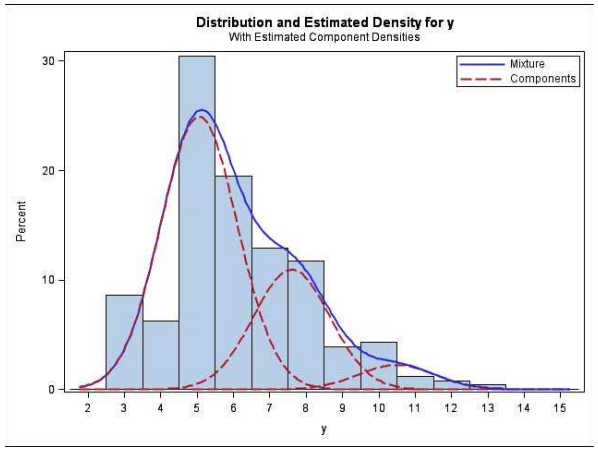
$g = 1$



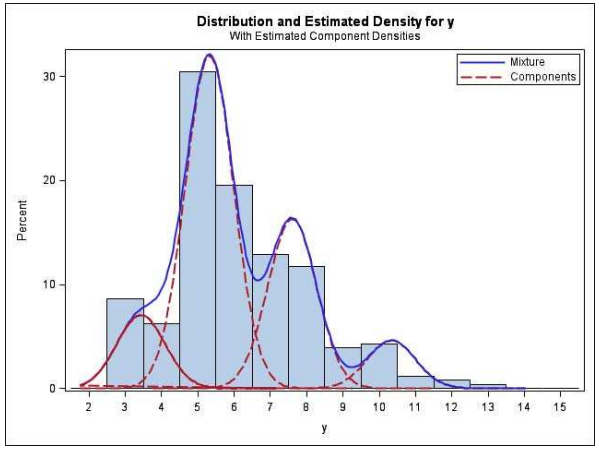
$g = 2$



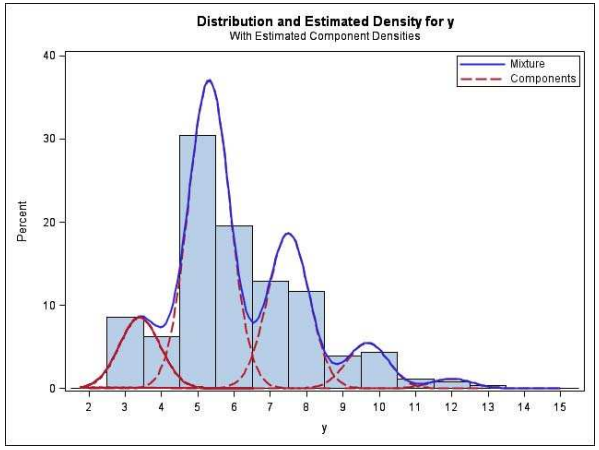
$g = 3$



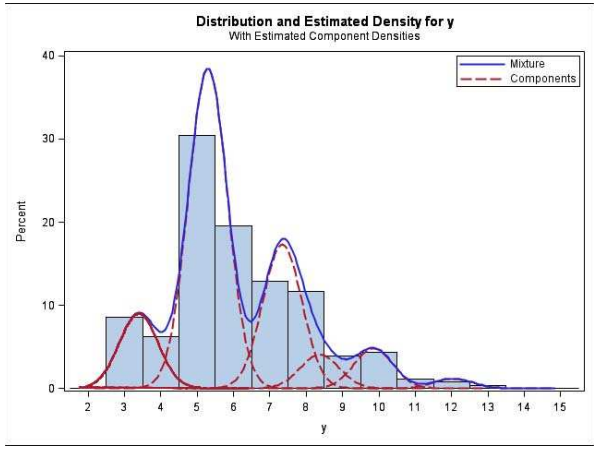
$g = 4$



$g = 5$



$g = 6$



- The one-component mixture equals the normal distribution with the sample mean and the sample variance as mean and variance:

$$Y_i \sim N(\overline{y}, s_y^2)$$

- The residual variance σ^2 decreases as more components are added to the mixture.

- This is to be expected from the previously derived result $\text{Var}(Y) = \text{Var}(X) + \sigma^2$ and from the fact that adding support points for X increases the variability of X.

- Adding components to the mixture increases the maximized log-likelihood value

- Selecting g requires some measure to compare models with different g

- One possible measure is the difference in maximized log-likelihood

$$\Longrightarrow \quad \text{LR test}$$

- However, this is not a standard testing procedure, as will be shown in the next section.

I.5.2 Mixture of 2 Known Distributions

- Suppose that for a continuous response Y, it is of interest to test whether the density of Y equals f_1, versus the alternative that the density is a mixture of f_1 and a 'contaminating density' f_2:

$$\begin{cases} H_0 : f(y) = f_1(y) \\ \\ H_A : f(y) = \pi f_1(y) + (1-\pi) f_2(y) \end{cases}$$

- Note that this is equivalent with

$$\begin{cases} H_0 : g = 1 \\ \\ H_A : g = 2 \end{cases}$$

- Let ℓ_1 and ℓ_2 denote the maximized log-likelihood values under the one-component model and the two-component model, respectively

- A classical likelihood ratio (LR) test would be based on the test statistic

$$\xi \;=\; 2(\ell_2 - \ell_1)$$

  and the p-value would be calculated assuming that

$$\xi \;\xrightarrow{H_0}\; \chi^2_1, \qquad \text{when } N \to +\infty$$

- However, we have the following result:

$$P[\xi \;=\; 0] \;\xrightarrow{H_0}\; 0.5$$

- Proof:
  - ▷ The log-likelihood function under the mixture equals

$$\ell_2(Y) \;=\; \sum_i \ln[\pi f_1(Y_i) + (1-\pi) f_2(Y_i)]$$

  - ▷ The first-order derivative evaluated at $\pi = 1$ equals

$$\left.\frac{\partial \ell_2(Y)}{\partial \pi}\right|_{\pi=1} \;=\; \sum_i \left.\frac{[f_1(Y_i) - f_2(Y_i)]}{[\pi f_1(Y_i) + (1-\pi) f_2(Y_i)]}\right|_{\pi=1} \;=\; N \;-\; \sum_i f_2(Y_i)/f_1(Y_i)$$

  - ▷ Note that

$$E\left[\frac{f_2(Y_i)}{f_1(Y_i)} \;\middle|\; H_0\right] \;=\; \int [f_2(y)/f_1(y)] f_1(y) dy \;=\; 1$$

▷ It now immediately follows from the Central Limit Theorem that

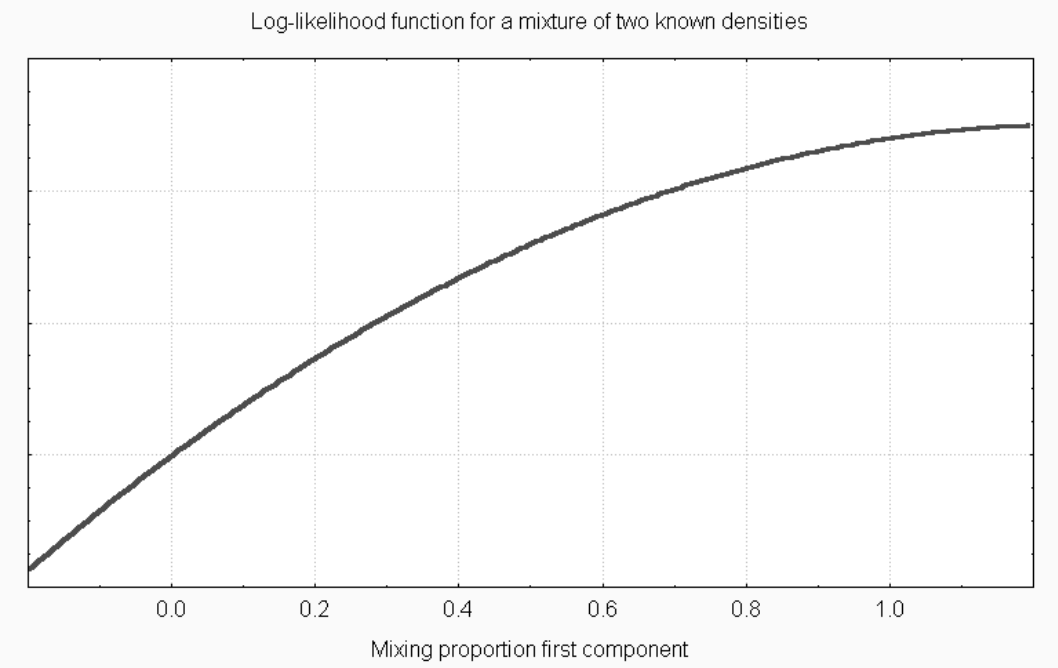
$$P\left[\left.\frac{\partial \ell_2(Y)}{\partial \pi}\right|_{\pi=1} > 0 \;\middle|\; H_0\right] \xrightarrow{H_0} 0.5$$

▷ The second-order derivative of ℓ_2 equals:

$$\frac{\partial^2 \ell_2(Y)}{\partial \pi^2} = -\sum_i \frac{[f_1(Y_i) - f_2(Y_i)]^2}{[\pi f_1(Y_i) + (1-\pi) f_2(Y_i)]^2} < 0$$

▷ Hence, we have that ℓ_2 is concave, and that the first-order derivative at $\pi = 1$ is strictly positive with 50% chance

▷ Graphically:



▷ This implies that

$$P(\widehat{\pi} = 1) \xrightarrow{H_0} 0.5$$

from which the result immediately follows ■

- The above property states that in half of the cases, the LR test statistic ξ will be exactly zero, for sufficiently large samples.
- Obviously the classical χ^2-approximation is not valid
- The reason is that H_0 is on the boundary of the parameter space under H_A
- This can be seen from rewriting the hypothesis as

$$\begin{cases} H_0 : \pi = 1 \\ H_A : \pi < 1 \end{cases}$$

- In general, the classical LR test cannot be used for testing for the number of components in a finite mixture
- The asymptotic distribution is to be derived for each testing problem separately

I.5.3 Some Results

- Testing $H_0 : \pi = 1$ in a mixture of two known densities:

$$\xi \xrightarrow{H_0} 0.5\chi_0^2 + 0.5\chi_1^2$$

  where χ_0^2 equals the discrete distribution with all probability mass at 0

- Testing $H_0 : \pi = 1$ in a mixture of two normals with common unknown variance:

$$\xi \xrightarrow{H_0} \chi_2^2$$

- Testing $H_0 : \pi = 1$ in a mixture of two Binomials Bin$(2, p_1)$ and Bin$(2, p_2)$:

$$\xi \xrightarrow{H_0} 0.5\chi_0^2 + 0.5\chi_1^2$$

- Testing $H_0 : \pi = 1$ in a mixture of two Poissons Poisson(λ_1) and Poisson(λ_2) with small λ_j ($\lambda_j \in [0, 0.1]$):

$$\xi \xrightarrow{H_0} 0.5\chi^2_0 + 0.5\chi^2_1$$

- In general, simulation methods are used to study the asymptotic behavior of ξ

- If there is any convergence at all, it can be painfully slow

- Thode, Finch, & Mendell (Biometrics, 1988, 44:1195–1201):

  "...one would need what is usually an infeasible large sample size ($N > 1000$) for the use of large-sample approximations to be justified."

- In practice, the null-distribution of ξ can be derived via bootstrap methods, which will not be discussed here any further.

I.5.4 Conclusions

- Inference on the number g of components in a finite mixture is far from straightforward

- One way to avoid the selection of g is to treat g as a parameter in the likelihood, and to estimate g from the available data

$$\Longrightarrow$$ **N**on-**P**arametric **M**aximum **L**ikelihood **E**stimation (NPMLE)

- Note that this involves more than classical ML estimation theory as the number of parameters in the likelihood is not fixed: The number of support points as well as the number of associated component probabilities increases with g.

Chapter I.6
Non-parametric Maximum Likelihood

- ▷ Introduction
- ▷ Definition of NPMLE
- ▷ Characterization of NPMLE
- ▷ Examples

I.6.1 Introduction

- All examples discussed so far acted conditional on the number g of mixture components, i.e., on the number g of support points for the latent variable X
- In this chapter, it will be investigated how g can be estimated from the data
- As an example, we analyze the SIDS data set using mixtures of g Poisson distributions, for varying g:

$$Y_i \sim \sum_{j=1}^{g} \pi_j \text{Poisson}(p_j n_i)$$

or equivalently

$$Y_i|p \sim \text{Poisson}(pn_i) \qquad p \sim \begin{pmatrix} p_1 & p_2 & \cdots & p_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

- Summary of results:

g	$\begin{pmatrix} p_1 & p_2 & \cdots & p_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$	ℓ
1	$\begin{pmatrix} 0.0020 \\ 1 \end{pmatrix}$	-255.58
2	$\begin{pmatrix} 0.0016 & 0.0035 \\ 0.75 & 0.25 \end{pmatrix}$	-237.28
3	$\begin{pmatrix} 0.0012 & 0.0021 & 0.0042 \\ 0.33 & 0.53 & 0.14 \end{pmatrix}$	-234.41
4	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.15 & 0.01 \end{pmatrix}$	-233.40
5	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.11 & 0.04 & 0.01 \end{pmatrix}$	-233.40
6	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0037 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.09 & 0.05 & 0.01 & 0.01 \end{pmatrix}$	-233.40

- Once the latent variable X has 4 support points, the log-likelihood value cannot be increased anymore by including more support points.
- It can be shown that for any $g > 4$, the maximized log-likelihood equals -233.40.
- This suggests using a 4-component mixture to describe the data.
- The resulting estimate for the distribution of the latent variable X is called a NPMLE: It maximizes the log-likelihood value over the class of **all** distributions for X.

I.6.2 Definition and Properties of NPMLE

- Let $f_i(y_i|x)$ denote the density function (continuous or discrete) of Y_i given the latent variable X
- Let G be the distribution function of X (continuous or discrete)
- G is called the **mixing distribution**.
- The marginal density of Y_i equals

$$f_i(y|G) = \int f_i(y_i|x)dG(x)$$

  where the integral becomes a sum in case X is discrete.

- The log-likelihood is then obtained as

$$\ell(G) = \sum_{i=1}^{N} \ln[f_i(y_i|G)]$$

- In many applications (with discrete responses) data incorporate replications such that $\ell(G)$ is of the form

$$\ell(G) \;=\; \sum_{i=1}^{m} \omega_i \ln[f_i(y_i|G)]$$

  where there are only m different values, each occurring $\omega_1, \ldots, \omega_m$ times.

- Note that possible dependence on unknown parameters is suppressed in the above notation

- A NPMLE for G is any distribution function $\widehat{G}$ for which $\ell(G)$ is maximized over the class of **all** distributions, i.e.,

$$\boxed{\ell(\widehat{G}) \;=\; \max_{G \in \Gamma} \; \ell(G)}$$

- Note that the log-likelihood is maximized over the class Γ of all distributions, i.e., discrete as well as continuous distributions.

- Property:

The log-likelihood $\ell(G)$ is concave in Γ

- This implies that $\ell(G)$ has a unique mode.

- Hence, we have that

A NPMLE $\widehat{G}$ exists

- In many cases, $\widehat{G}$ will be unique. However, it is not in general.

- Further, it can be shown that

$$\boxed{\widehat{G} \text{ is discrete with at most } m \text{ support points}}$$

- The discreteness of $\widehat{G}$ allows to maximize $\ell(G)$ over the class of all discrete distributions only.

- Let Ω be the class of all discrete distributions. We then have that

$$\boxed{\ell(\widehat{G}) = \max_{G \in \Gamma} \ell(G) = \max_{G \in \Omega} \ell(G)}$$

- The upper bound for the number of support points is seldom sharp, i.e., there will often be (many) less support points than indicated by the bound.

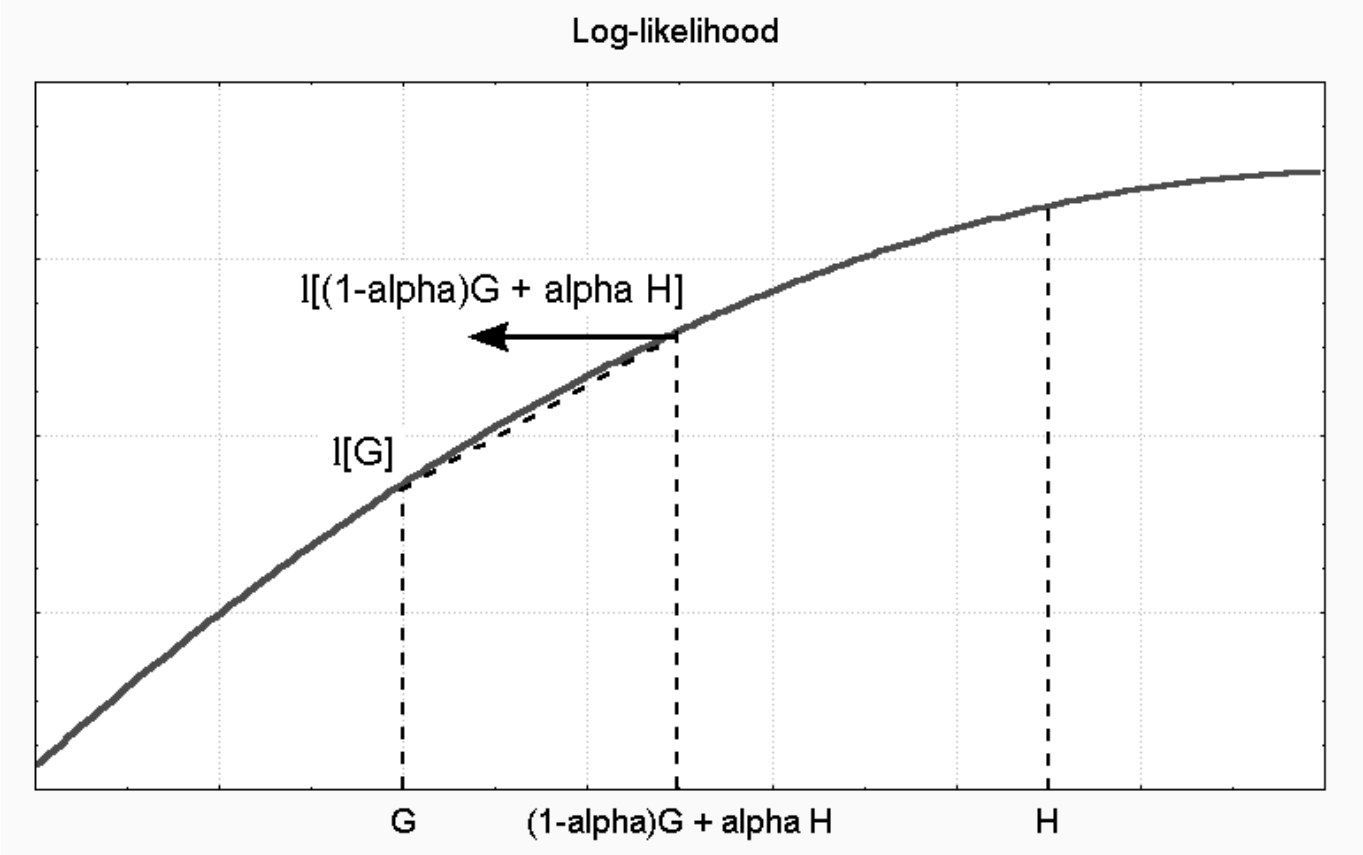
I.6.3 Characterization of a NPMLE

6.3.1 Directional Derivative and Gradient Function

- For G and H in Ω, the directional derivative of $\ell(\cdot)$ at G into the direction H is defined as

$$\Phi(G, H) = \lim_{\alpha \overset{>}{\to} 0} \frac{\ell[(1-\alpha)G + \alpha H] - \ell(G)}{\alpha}$$

- Graphical interpretation:



- Note that

$$\begin{aligned}
\Phi(G,H) &= \lim_{\alpha \stackrel{>}{\to} 0} \frac{\ell[(1-\alpha)G + \alpha H] - \ell(G)}{\alpha} \\
&= \left. \frac{\partial\ \ell[(1-\alpha)G + \alpha H]}{\partial \alpha} \right|_{\alpha=0} \\
&= \left. \frac{\partial\ \sum_i \ln[(1-\alpha) f_i(y_i|G) + \alpha f_i(y_i|H)]}{\partial \alpha} \right|_{\alpha=0} \\
&= \sum_i \frac{f_i(y_i|H)\ -\ f_i(y_i|G)}{f_i(y_i|G)} \\
&= \sum_i \frac{f_i(y_i|H)}{f_i(y_i|G)}\ -\ N
\end{aligned}$$

- For every discrete distribution H in Ω,

$$H = \begin{pmatrix} x_1 & x_2 & \cdots & x_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix},$$

we have that

$$\begin{aligned} \Phi(G,H) &= \sum_i \frac{f_i(y_i|H)}{f_i(y_i|G)} - N = \sum_i \frac{\sum_j \pi_j f_i(y_i|x_j)}{f_i(y_i|G)} - N \\ &= \sum_j \pi_j \left[\sum_i \frac{f_i(y_i|x_j)}{f_i(y_i|G)}\right] - N = N \left[\sum_j \pi_j d(G,x_j) - 1\right] \end{aligned}$$

with

$$d(G,x) = \frac{1}{N} \sum_i \frac{f_i(y_i|x)}{f_i(y_i|G)}$$

- $d(G, x)$ is called the gradient function of G, evaluated at x
- A NPMLE can now be defined as any $\widehat{G}$ in Ω such that

$$\Phi(\widehat{G}, H) \leq 0, \qquad \text{for all } H \text{ in } \Omega$$

- Hence, the NPMLE will be characterized with the gradient function $d(G, x)$
- Three theorems are useful to check if a candidate estimate $\widehat{G}$ is really NPML.

- **Theorem 1:**

> $\widehat{G}$ is NPMLE if and only if for all x, $d(\widehat{G}, x) \leq 1$

- **Theorem 2:**

> If $\widehat{G}$ is a NPMLE, we have that $d(\widehat{G}, x) = 1$ for all support points x of $\widehat{G}$
>
> $d(\widehat{G}, x)$ is identically one if and only if $\widehat{G}$ is not unique

- **Theorem 3:**

> If all $f_i(y_i|x)$, as functions of x, have unique modes in some interval $[a, b]$, then $\widehat{G}$ can only have support points in the interval $[a, b]$.

- Theorem 1 provides a tool to check whether $\widehat{G}$ is a NPMLE
- Theorem 2 provides a tool to check uniqueness
- Theorem 3 allows to restrict attention to the interval $[a, b]$

I.6.4 Example: The SIDS Data

- Under the Poisson(pn_i) model, we have that

$$f_i(y_i|p) \;=\; \exp(-pn_i)\,\frac{(pn_i)^{y_i}}{y_i!}$$

  which, as a function of p, is uniquely maximized for $p = y_i/n_i$.

- Theorem 3 then implies that a NPMLE $\widehat{G}$ will have support points between the smallest and the largest observed rate, i.e., in the interval $[0, 0.0096]$

- Gradually increasing the number of components in a mixture suggested that the following 4-component mixture is NPML:

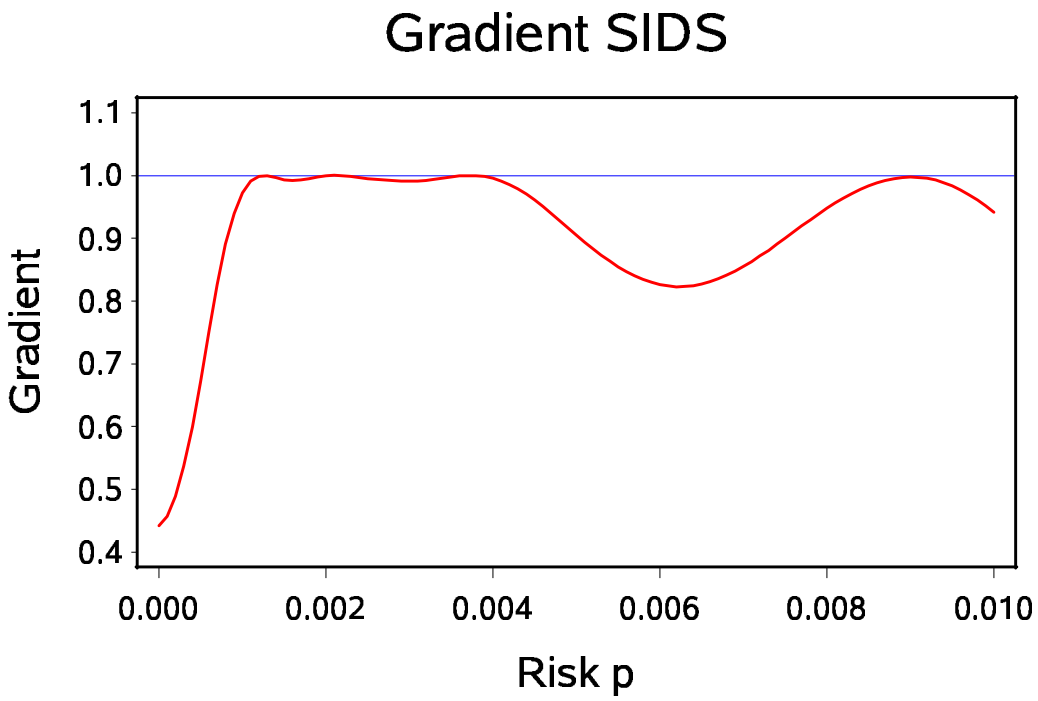
$$Y_i|p \;\sim\; \text{Poisson}(pn_i) \qquad p \;\sim\; G \;=\; \begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0090 \\ 0.33 & 0.51 & 0.15 & 0.01 \end{pmatrix}$$

- The gradient function $d(G, p)$ for the above mixing distribution G can be obtained using the SAS code:

```
data test; set sids;

data test; set test;
do x=0 to 0.01 by 0.0001; output; end;

data test; set test;
gradient=pdf('POISSON',y,x*n)/(0.3263*pdf('POISSON',y,0.001254*n)
          + 0.5124*pdf('POISSON',y,0.002081*n)
          + 0.1505*pdf('POISSON',y,0.003747*n)
          + 0.0108*pdf('POISSON',y,0.009013*n));

proc sort data=test; by x;

proc means data=test; var gradient; by x;
output out=out;

data out; set out; if _STAT_='MEAN';
```

```
title h=2.5 'Gradient SIDS';
proc gplot data=out;
plot gradient*x / nolegend haxis=axis1 vaxis=axis2 cvref=blue vref=1;
symbol c=red i=join w=5 l=1 mode=include;
axis1 label=(h=2 'Risk p')  value=(h=1.5) order=(0 to 0.01 by 0.002)
      minor=none  w=5 ;
axis2 label=(h=2 angle=90 'Gradient') value=(h=1.5) order=(0.4 to 1.1 by 0.1)
      minor=none w=5 ;
run;quit;
```

- Result:



- G is NPMLE because $d(G, p) \leq 1$ in the interval $[0, 0.0096]$
- Note also that $d(G, p) = 1$ for p in $\{0.0013, 0.0021, 0.0037, 0.0090\}$
- The obtained estimate is unique as the gradient function is not identically one.

I.6.5 Example: Accident Data

- We now consider the number of accident claims during one year, out of 9461 insurance policies issued by La Royal Belge Insurance Company:

Count y_i:	0	1	2	3	4	5	6	7
Frequency ω_i:	7840	1317	239	42	14	4	4	1

- These data have been analyzed frequently in the statistical literature:
  - ▷ Thyrion (Astin Bulletin, 1960, 1:142–162)
  - ▷ Simar (Annals of Statistics, 1976, 4:1200–1209)
  - ▷ Carlin & Louis (Chapman & Hall, 1996)
  - ▷ Böhning (Chapman & Hall, 1999)

- Under the Poisson(λ) model, we have that

$$f_i(y_i|\lambda) \;=\; \frac{e^{-\lambda}\,\lambda^{y_i}}{y_i!}$$

  which, as a function of λ, is uniquely maximized for $\lambda = y_i$.

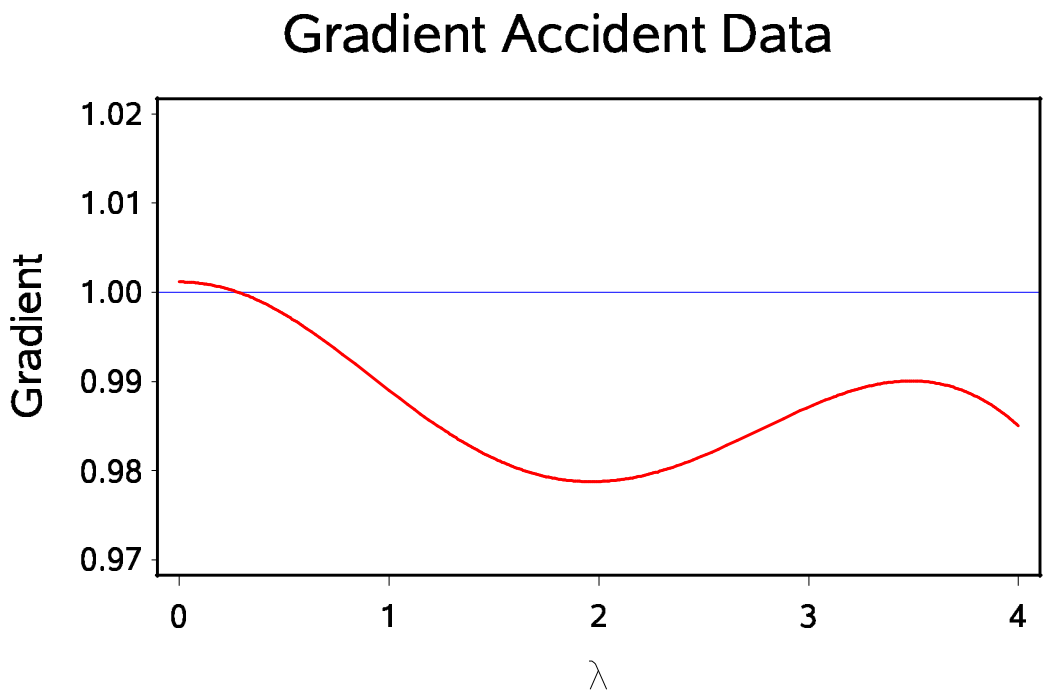
- Theorem 3 then implies that a NPMLE $\widehat{G}$ will have support points between the smallest and the largest observation, i.e., in the interval $[0, 7]$

- Simar and Carlin & Louis report that a NPMLE is given by

$$Y|\lambda \;\sim\; \text{Poisson}(\lambda) \qquad \lambda \;\sim\; G \;=\; \begin{pmatrix} 0.089 & 0.580 & 3.176 & 3.669 \\ 0.7600 & 0.2362 & 0.0037 & 0.0002 \end{pmatrix}$$

- The reported maximized log-likelihood value equals $\ell = -5341.5310$.

- The gradient function $d(G, p)$ for the above mixing distribution G equals:

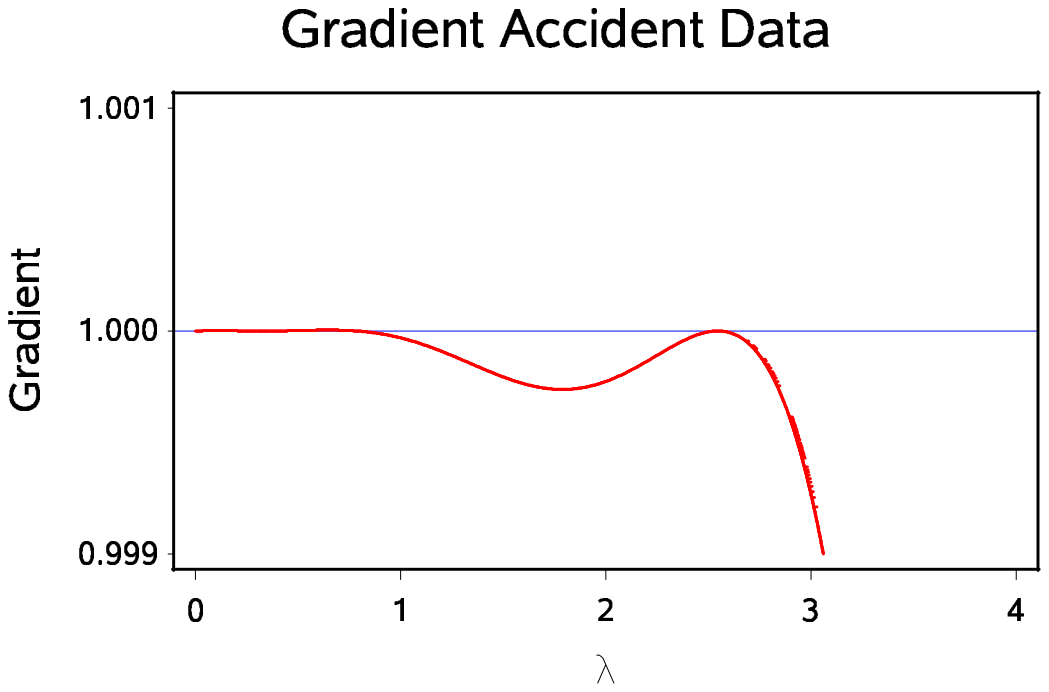


- Hence, G is **not** NPMLE because $d(G, p) > 1$ in the neighborhood of 0, and does not reach 1 at all support points

- Böhning reports that a NPMLE is given by

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim G = \begin{pmatrix} 0.000 & 0.336 & 2.545 \\ 0.4184 & 0.5730 & 0.0087 \end{pmatrix}$$

- The corresponding maximized log-likelihood value now equals $\ell = -5340.7040$, which is indeed larger than the value reported by Simar and Carlin & Louis, and the gradient function equals:



Chapter I.7
Numerical Algorithms

- ▷ CAMAN approach to NPMLE
- ▷ Vertex exchange method (VEM)
- ▷ Stopping rule

I.7.1 CAMAN Approach to NPMLE

7.1.1 Introduction

- The theorems discussed earlier imply that, under mild regularity conditions, a NPMLE $\widehat{G}$ is discrete, with support 'within the range of the data'.
- Moreover, the gradient $d(\widehat{G}, x)$ of $\widehat{G}$ in any direction x is never larger than 1, and equals 1 for all support points of $\widehat{G}$.
- In general, although $\widehat{G}$ will be discrete, the support could be large, especially for continuous data, or when analyzing rates.
- One approach could be to start the EM algorithm discussed earlier, for a mixture model with a 'sufficiently large' number of components g.

- The idea is then that, if g is larger than the support size of $\widehat{G}$, some of the estimated support points will coincide, or some support points will get weight (probability) zero.

- Coinciding support points have already been obtained earlier in the analysis of the SIDS data:

g	$\begin{pmatrix} p_1 & p_2 & \cdots & p_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$	ℓ
4	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.15 & 0.01 \end{pmatrix}$	-233.40
5	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.11 & 0.04 & 0.01 \end{pmatrix}$	-233.40
6	$\begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0037 & 0.0037 & 0.0090 \\ 0.32 & 0.52 & 0.09 & 0.05 & 0.01 & 0.01 \end{pmatrix}$	-233.40

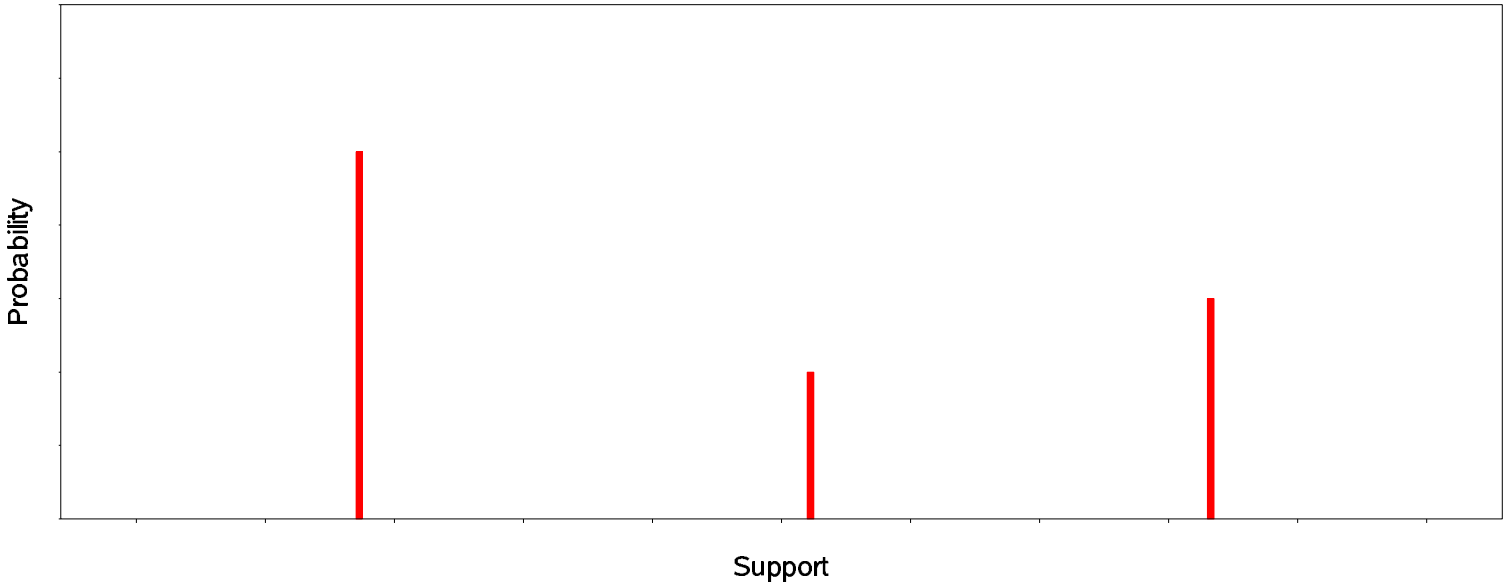
- Indeed, as was seen later, the NPMLE $\widehat{G}$ was unique, and contained only 4 support points.

- However, if the EM algorithm is started from a mixing distribution with many support points (e.g., $g = 25$), g will often be much too large, leading to extremely slow convergence of the algorithm.

- In CAMAN, this is solved by splitting up the estimation procedure in two different phases

- The procedure will be illustrated in a specific example where the NPMLE consists of 3 support points

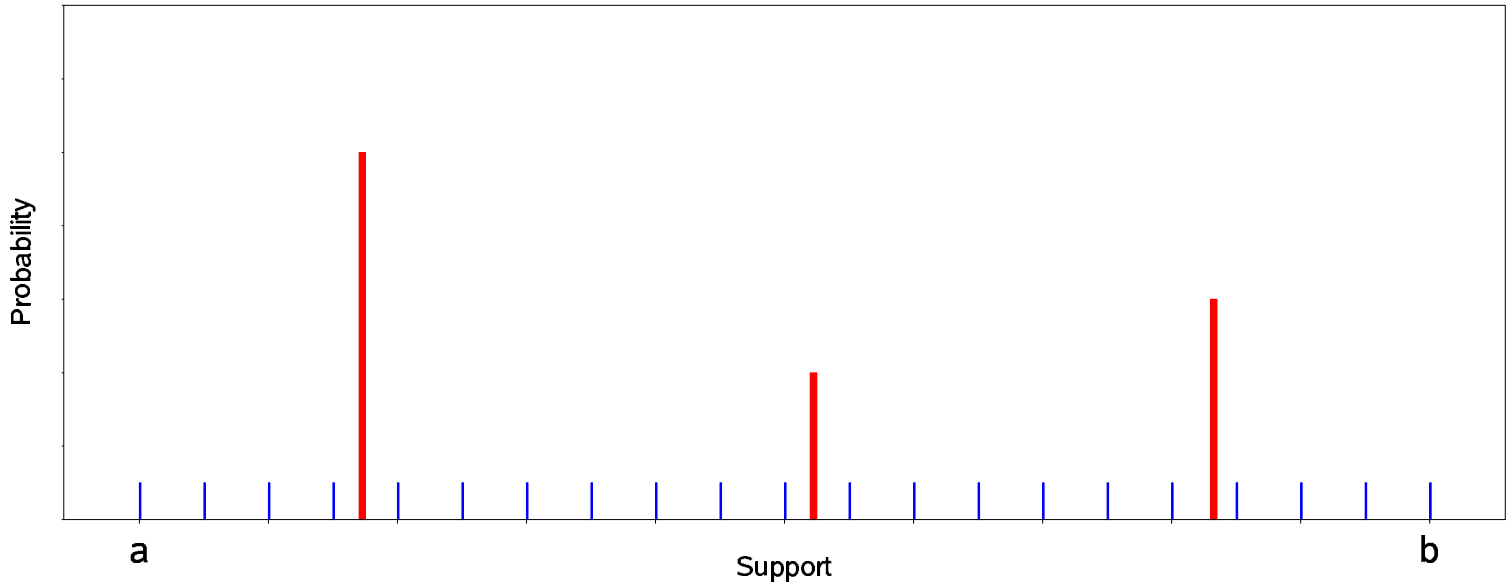
- Graphical representation of final solution:



7.1.2 Phase 1 of CAMAN

- Let $[a, b]$ be the interval that will contain all support points (Theorem 3)

- A large grid $\Lambda = \{a = x_1, \ldots, x_L = b\}$ is specified as 'first guess' for the support points of $\widehat{G}$:

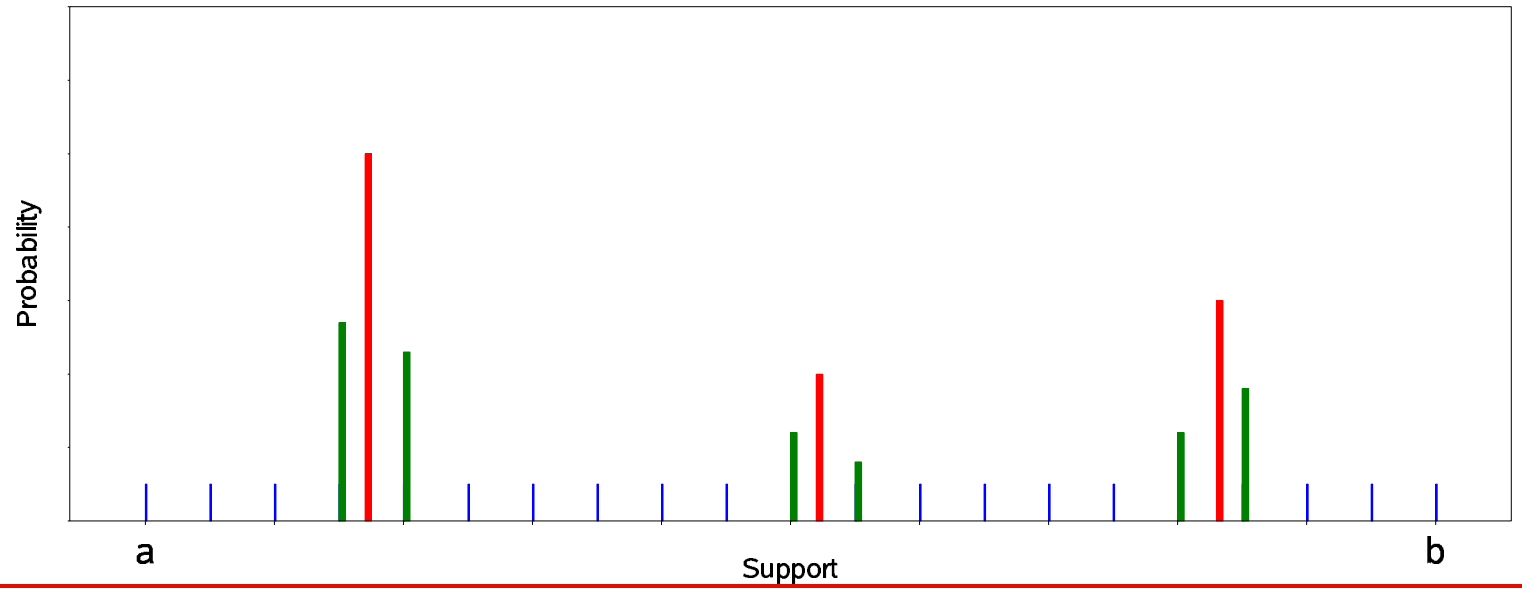


- The log-likelihood $\ell(G)$ is maximized over this grid, i.e., over all probability distributions with support Λ

- This only requires estimation of the corresponding weights $\{\pi_1, \ldots, \pi_L\}$ and possibly also parameters in the models $f_i(y_i|x)$ (e.g., the variance σ^2 in the normal model).

- This can be done using any of the numerical methods which will be discussed in the next sections.

- If L was chosen sufficiently large, Phase 1 results in many grid points with zero weight, while grid points in the region of the final solution receive positive weight:



- From now on, it will be assumed that the number g of support points in $\widehat{G}$ is at most the number of grid points with positive weight.

- This assumption is only justified if the number L of grid points was chosen sufficiently large, i.e., if the grid can be assumed to be a good approximation to the support of $\widehat{G}$.

7.1.3 Phase 2 of CAMAN

- All grid points with positive weight in Phase I, together with the corresponding weights, are used as starting values for the EM algorithm, discussed earlier

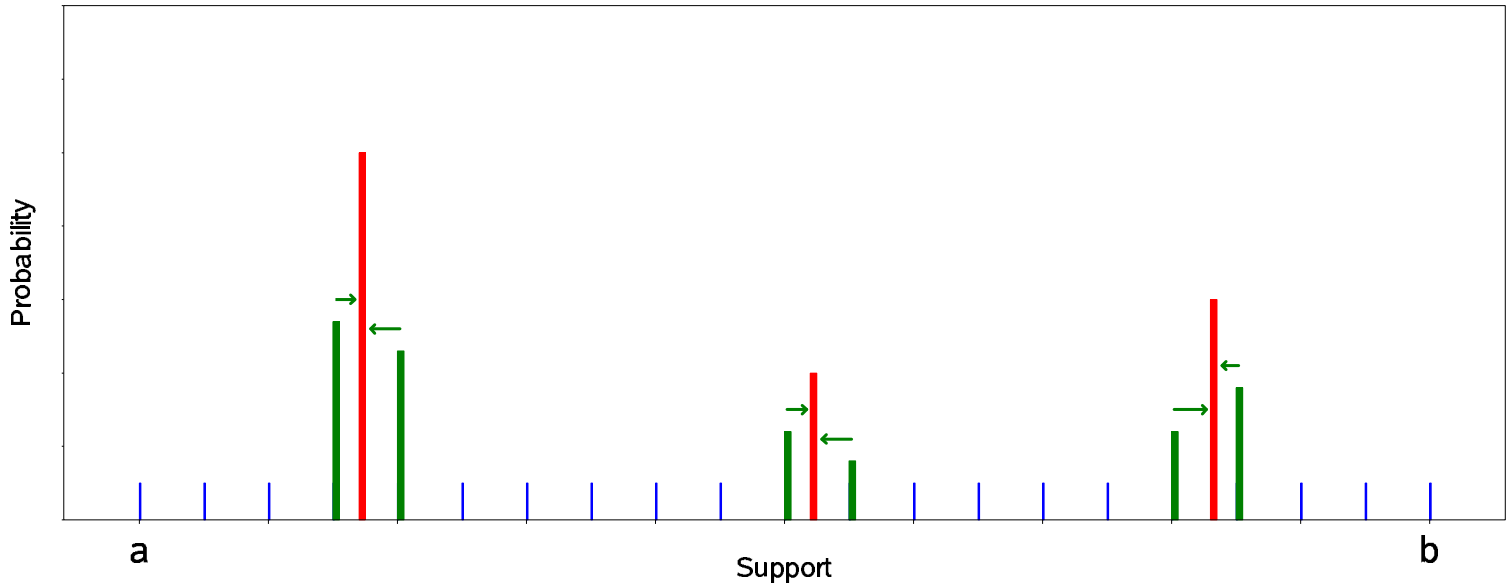
- So, in this phase, weights **as well as** support points are estimated, while in Phase 1 this was only the case for the weights.

- Note also that, in this phase, the number of support points is also kept fixed, but this number is now (much) smaller than in Phase I.

- In case the number of grid points with positive weight in Phase 1 was still larger than the number g of support points in $\widehat{G}$, than some estimated support points will coincide, or will receive zero weight.

- Coinciding points are combined. Points with zero weight are left out.

- In practice, the effect of Phase 2 is that the grid points with positive support obtained in Phase 1 converge to the final solution with combined weights:



- In order to assure that the so-obtained estimate $\widehat{G}$ is really NPMLE, it should be checked that the gradient function $d(\widehat{G}, x)$ is never larger than 1, and equals 1 for all support points of $\widehat{G}$.

I.7.2 Vertex Exchange Method (VEM)

- The first phase in the CAMAN approach to the calculation of a NPMLE $\widehat{G}$ is the fitting of a finite mixture model, with large but fixed support for the mixing distribution G.

- Several algorithms have been proposed but the vertex exchange method (VEM), implemented in CAMAN, is the most efficient one so far.

- All algorithms maximize the mixture log-likelihood $\ell(G)$ over the class of all discrete mixing distributions with support equal to (a subset of) $\Lambda = \{x_1, \ldots, x_L\}$.

- Let G be a current guess for the final solution. Improving G implies reducing the weight of some support points, while increasing the weight of others.

- General idea:

> G can be improved by moving weight from a 'bad' support point to a 'good' one

- VEM will replace G by

$$G - \alpha\pi^{-}G_{x^{-}} + \alpha\pi^{-}G_{x^{+}} = G + \alpha\pi^{-}(G_{x^{+}} - G_{x^{-}}),$$

  for some $\alpha \in [0, 1]$ and support points x^{-} and x^{+}, and with G_x representing the degenerate distribution with support x.

- All support points of G, except x^{-} and x^{+}, keep their original weights, while a proportion α of the weight π^{-} of the 'bad' support point x^{-} is moved to a 'good' support point x^{+}.

- α is called the step-length

- The 'optimal' choice for x^-, x^+ and α is the one which maximizes the gain in log-likelihood, i.e., which maximizes

$$\ell[G + \alpha\pi^-(G_{x^+} - G_{x^-})] \; - \ell[G]$$

- First-order Taylor approximation for small α yields

$$\begin{aligned}
&\ell[G + \alpha\pi^-(G_{x^+} - G_{x^-})] \; - \ell[G] \\
&\approx \; \alpha \; \left.\frac{\partial \ell[G + \alpha\pi^-(G_{x^+} - G_{x^-})]}{\partial \alpha}\right|_{\alpha=0} \\
&= \; \alpha \; \left.\frac{\partial \ell\{(1-\alpha)G + \alpha[G + \pi^-(G_{x^+} - G_{x^-})]\}}{\partial \alpha}\right|_{\alpha=0} \\
&= \; \alpha \; \Phi[G, G + \pi^-(G_{x^+} - G_{x^-})] \\
&= \; \alpha \; \left[\sum_i \frac{f_i(y_i \mid G + \pi^-(G_{x^+} - G_{x^-}))}{f_i(y_i|G)} \; - \; N\right]
\end{aligned}$$

$$= \alpha \left[\sum_i \frac{f_i(y_i|G) - \pi^- f_i(y_i|x^-) + \pi^- f_i(y_i|x^+)}{f_i(y_i|G)} - N \right]$$

$$= \alpha \left[N - \pi^- \sum_i \frac{f_i(y_i|x^-)}{f_i(y_i|G)} + \pi^- \sum_i \frac{f_i(y_i|x^+)}{f_i(y_i|G)} - N \right]$$

$$= \alpha \, N \, \pi^- \, [d(G, x^+) - d(G, x^-)]$$

- Obviously, the best choice for x^+ is the support point in Λ for which the gradient function is maximal

- Further, the best choice for x^- is the support point in Λ for which the gradient function is minimal

- Given G, an updated version of G is obtained from the following algorithm:

  ▷ Find $x^+ \in \Lambda$ which maximizes $d(G, x)$

  ▷ Find $x^- \in \Lambda$ which minimizes $d(G, x)$

  ▷ Find α which maximizes

  $$\ell[G + \alpha\pi^-(G_{x^+} - G_{x^-})] \; - \ell[G]$$

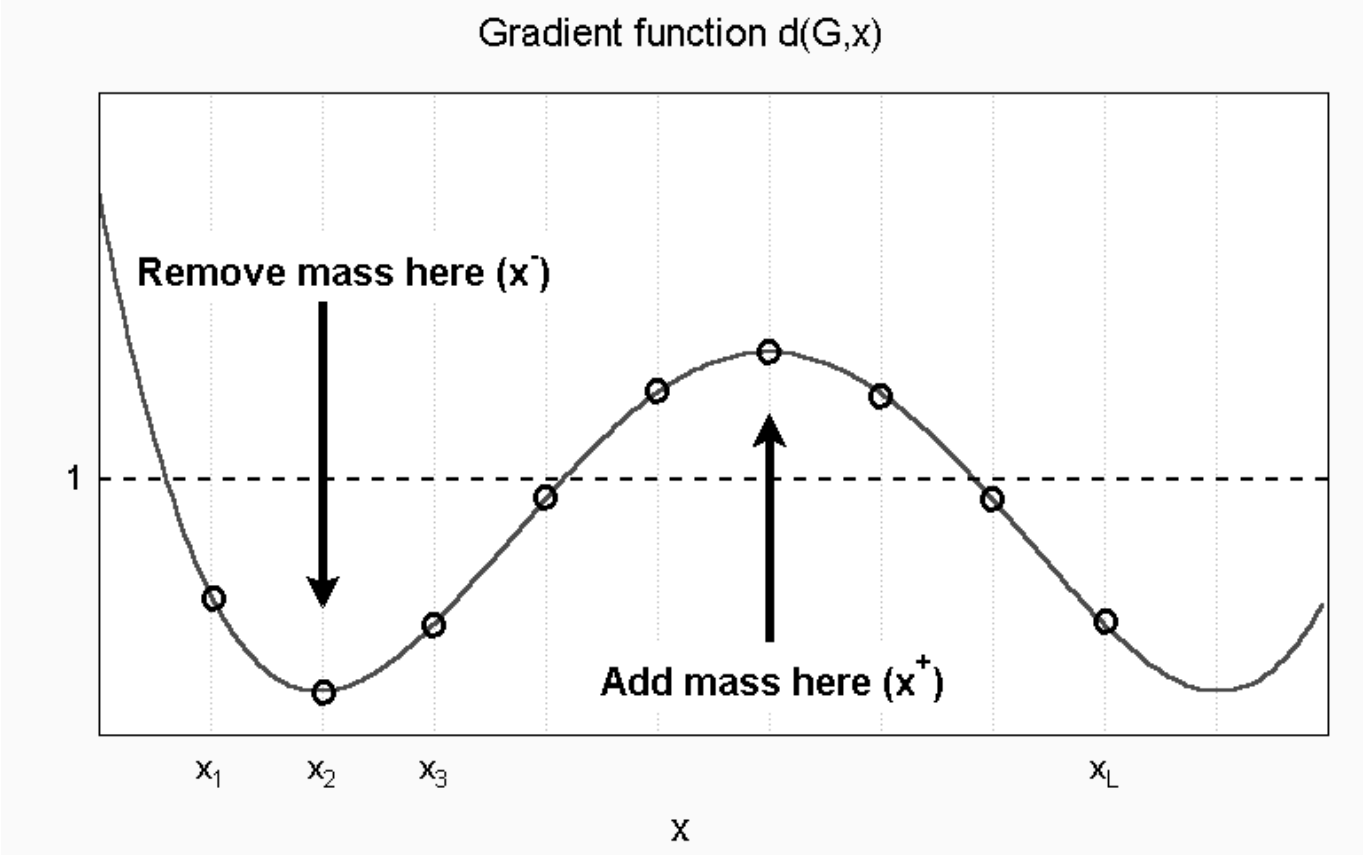
  over α

  ▷ Replace G by $G + \alpha\pi^-(G_{x^+} - G_{x^-})$

- This algorithm is repeated until convergence.

- Note that maximization with respect to α usually requires iterative optimization procedures

- Graphical representation of selection of x^- and x^+:



- Let $G^{(t)}$ be any sequence created by the above algorithm, and let $\widehat{G}$ be NPMLE, then one can show that

$$\ell[G^{(t)}] \longrightarrow \ell[\widehat{G}], \qquad \text{monotonously}$$

provided the grid $\Lambda = \{x_1, \ldots, x_L\}$ is sufficiently dense.

I.7.3 Stopping Rule

- The numerical algorithms discussed earlier all result in a sequence

$$\left\{G^{(1)}, G^{(2)}, \ldots, G^{(t)}, G^{(t+1)}, \ldots\right\}$$

  which converges to a NPMLE $\widehat{G}$.

- In practice, one needs a stopping rule to decide when the iterative process is terminated, i.e., which $G^{(t)}$ will be considered to be sufficiently close to $\widehat{G}$ in order to be acceptable as NPMLE.

- We know from the first theorem of the characterization of NPMLE's that $d(\widehat{G}, x) \leq 1, \forall x$.

- An obvious stopping rule is then to select a small value $\varepsilon > 0$, and to stop the iterative procedure at the smallest t for which

$$d\left(G^{(t)}, x\right) \leq 1 \ + \ \varepsilon, \qquad \text{for all } x$$

- For VEM, it is sufficient to stop the estimation process as soon as

$$d\left(G^{(t)}, x^{+}\right) \leq 1 \ + \ \varepsilon$$

  which needs to be calculated anyway.

- In CAMAN, the ε is specified as the accuracy

- In order to guarantee that the iterative procedure will eventually stop, a maximal number of iteration steps is also required.

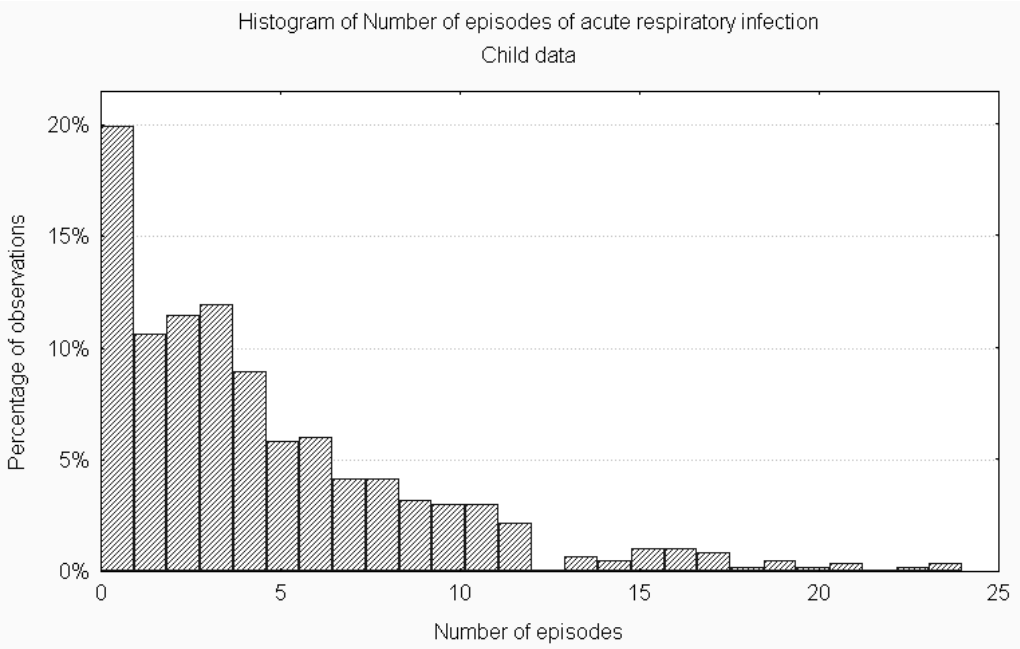
Chapter I.8
Examples in CAMAN

- Child data
- Snapper data

I.8.1 Child Data

- We will now illustrate the use of CAMAN for the estimation of the NPMLE for the Child data.

- Recall that the response of interest is the number of episodes of acute respiratory infection (fever, cough, running nose,...), recorded within a 3-year period, with histogram:



- The model equals

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

  with unknown number g of components in the mixture.

- In CAMAN, three related procedures are available:
  - ▷ mixalg.VEM: Phase 1
  - ▷ mixalg.EM: Phase 2 (discussed before)
  - ▷ mixalg: Phase 1 & Phase 2

- Phase 1 can be performed using the following syntax:

```
vem<-mixalg.VEM(obs="counts", weights="frequency", family="poisson",
                data=child, acc=10^(-8),numiter=5000, startk=50)
```

- The option 'acc=' specifies the accuracy ε used in the stopping rule
- The option 'numiter=' specifies the maximum number of iteration steps
- The option 'startk=' specifies the number of grid points in the initial grid $\Lambda = \{x_1, \ldots, x_L\}$
- Results:

```
Computer Assisted Mixture Analysis (VEM):
Data consists of 602 observations (rows).
The VEM-algorithm identified 8 grid points with positive support
              p          t
1 0.1151572058  0.0000000
2 0.0929999812  0.4897959
3 0.0627187858  2.4489796
4 0.4099903721  2.9387755
5 0.0608071940  7.8367347
6 0.2051281574  8.3265306
7 0.0522385437 16.1632653
8 0.0009597599 16.6530612
Log-Likelihood: -1553.883     BIC: 3741.392
```

- Out of the initial 50 grid points, only 8 received positive weight after the VEM run. All other grid points received weight 0.

- The maximized log-likelihood value after finalizing Phase 1 of the CAMAN procedure equals -1553.883

- Positive weight is often given to neighboring points:

Grid value	Weight
...	...
7.84	0.0608
8.33	0.2051
...	...
16.16	0.0522
16.65	0.0010
...	...

- Phase 2 can be performed using the 'mixalg.EM' procedure, with the results from the VEM run as input

- Alternatively, Phase 1 & Phase 2 can be performed jointly using:

```
npml<-mixalg(obs="counts", weights="frequency", family="poisson",
                data=child, acc=10^(-8),numiter=50000, startk=50)
```

- Results:

```
Computer Assisted Mixture Analysis:
Data consists of 602 observations (rows).
The Mixture Analysis identified 5 components of a poisson distribution:

              p     lambda
1 7.740583e-06  0.0000000
2 1.969225e-01  0.1433966
3 4.799752e-01  2.8172852
4 2.692583e-01  8.1641705
5 5.383626e-02 16.1558261

Log-Likelihood: -1553.81     BIC: 3165.223
```

- Two of the original 8 support points have converged to the values 2.8173, 8.1642, and 16.1558.

- The final solution has 5 distinct support points only.

- CAMAN allows to combine identical support points, i.e., support points which differ less than a limit, which can be pre-specified using an additional option 'limit=':
  - ▷ Support points with weights less than 'limit' are deleted
  - ▷ Support points less than 'limit' different are combined

- The default limit as well as some additional information about input parameters and results can be obtained with:

```
summary(npml)
```

- Results:

```
number of VEM-iterations done: 7445
alteration within the final VEM-iteration step: 9.9405e-09
number of EM-iterations done: 34306
alteration within the final EM-iteration step: 9.997101e-09

User-defined parameters:
   max number of iterations: 50000
   limit for combining components: 0.01
   threshold for converging: 1e-08
   number of grid points (startk): 50
```

- The maximized log-likelihood after Phase 2 equals -1553.81 which is only a minor improvement compared to the approximation obtained from the first phase (log-likelihood -1553.883).

- This minor further increase in log-likelihood required 34306 additional EM-steps, further illustrating the slow convergence of the EM-algorithm

- This suggests that the 8-point result from the first phase was already a (very) good approximation of the full NPMLE $\widehat{G}$.

- This also explains the results from Phase 1:

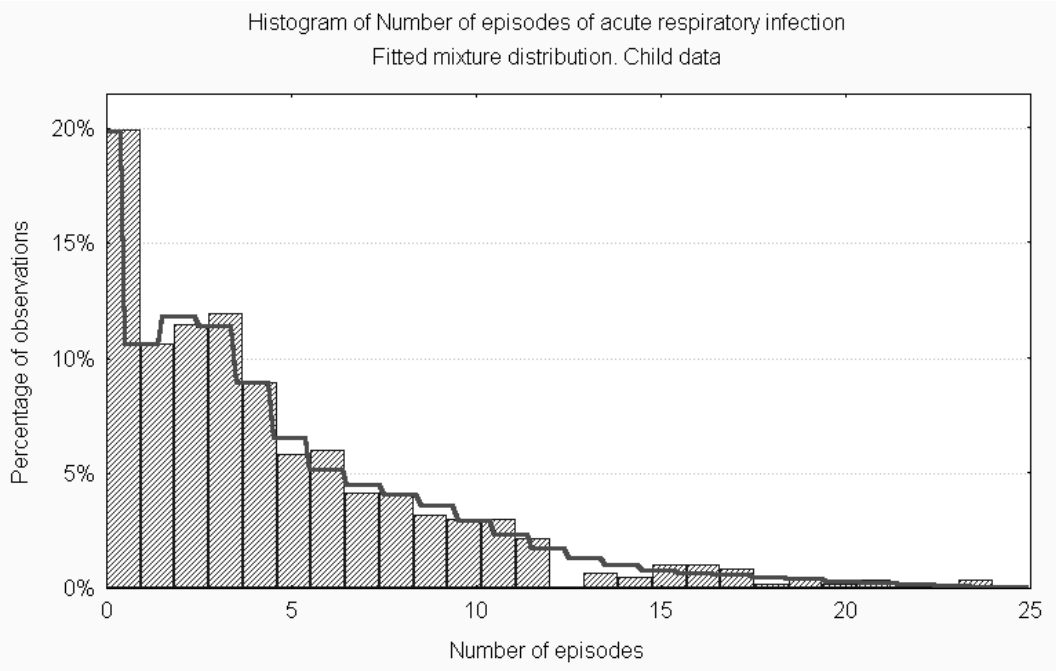
Grid value	Weight	
...	...	
7.84	0.0608	$\Big\}$ $= 0.2659 \approx 0.2693$ at 8.1642
8.33	0.2051	$\Big\}$
...	...	
16.16	0.0522	$\Big\}$ $= 0.0532 \approx 0.0538$ at 16.1558
16.65	0.0010	$\Big\}$
...	...	

- The neighboring points with positive weights suggest that the NPMLE has a support point somewhere between the two neighbors, with weight approximately equal to the sum of the weights of the neighbors.

- The fitted model can be summarized as:

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} 0 & 0.143 & 2.817 & 8.164 & 16.156 \\ 0.001 & 0.197 & 0.480 & 0.269 & 0.054 \end{pmatrix}$$

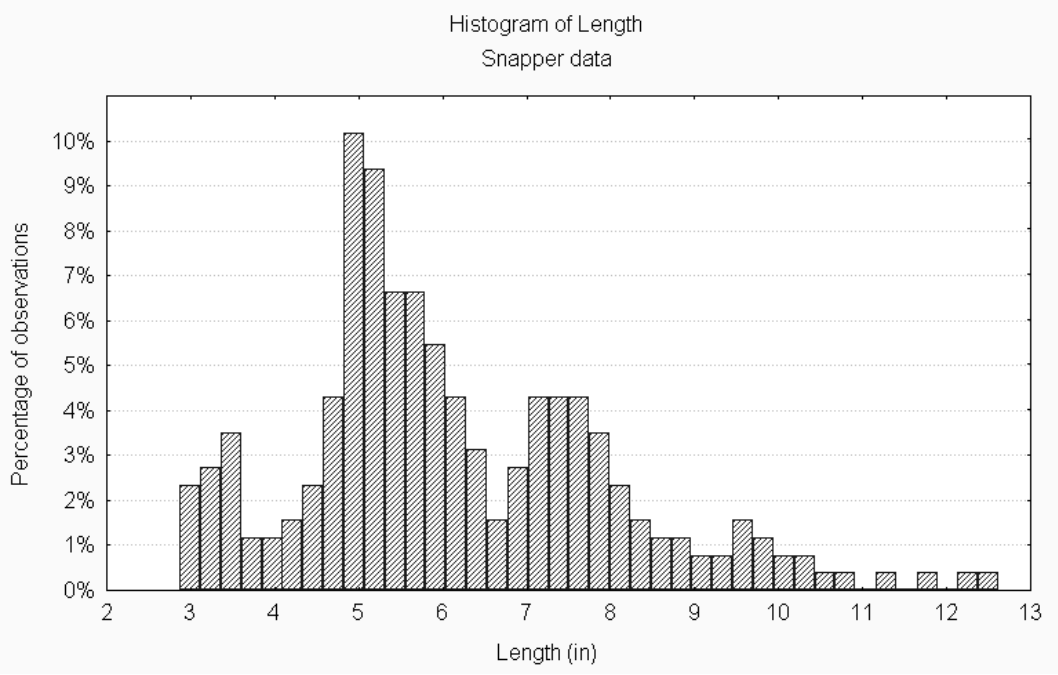
- Graphical representation:



I.8.2 Snapper Data

- We will now illustrate the use of CAMAN for the estimation of the NPMLE for the Snapper data.

- Recall that the response of interest is the length, with histogram:



- The model equals

$$Y|\mu \sim N(\mu, \sigma^2) \qquad \mu \sim \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_g \\ \pi_1 & \pi_2 & \cdots & \pi_g \end{pmatrix}$$

  with unknown number g of components in the mixture.

- A NPML estimate can be obtained in CAMAN using following syntax:

```
npml<-mixalg(obs="length", weights="frequency", family="gaussian",
             data=snapper, acc=10^(-8),startk=50)
```

- The result equals:

```
Computer Assisted Mixture Analysis:

Data consists of 256 observations (rows).
The Mixture Analysis identified 1 components of a gaussian distribution:
```

```
DETAILS:
  p     mean
1 1 6.103516
component variance: 0

Log-Likelihood: -563.7445     BIC: 1133.034
```

- Only 1 component is identified, and the component variance is set equal to 0, indicating a problem.

- Indeed the specified model is unidentified, which follows from

$$\text{Var}(Y) \;=\; \sigma^2 \;+\; \text{Var}(\mu)$$

- The only term identified from the data is Var(Y), estimated by the sample variance $s_y^2 = 3.60$

- How to split the total variance into within-component variability σ^2 and between-comonent variability Var(μ) is to be decided by the user.

- When NPMLE is the objective, the within-component variance σ^2 needs to be pre-specified using a 'var=' option.

- Obviously, σ^2 should be set equal to some value less than $s_y^2 = 3.60$

- NPMLE for $\sigma^2 = 3$:

```
          p     mean
1 0.9304275 5.826143
2 0.0695725 9.812964

Log-Likelihood: -519.7317     BIC: 1056.099
```

- NPMLE for $\sigma^2 = 2$:

```
           p      mean
1 0.81525492  5.504159
2 0.15510549  8.370224
3 0.02963959 10.727394

Log-Likelihood: -514.9862     BIC: 1057.698
```

- NPMLE for $\sigma^2 = 1$:

```
           p      mean
1 0.07929415  3.975117
2 0.57967082  5.207934
3 0.26631879  7.544679
4 0.06047305  9.793590
5 0.01424319 11.787159

Log-Likelihood: -510.9503     BIC: 1071.807
```

- NPMLE for $\sigma^2 = 0.2$:

```
           p      mean
1 0.114842014  3.336916
2 0.381770078  5.069379
3 0.164837218  5.981665
4 0.209089544  7.444038
5 0.050440157  8.489854
6 0.052514104  9.747340
7 0.009905078 10.497398
8 0.004406573 11.187444
9 0.012195233 12.226948

Log-Likelihood: -488.2221     BIC: 1070.712
```

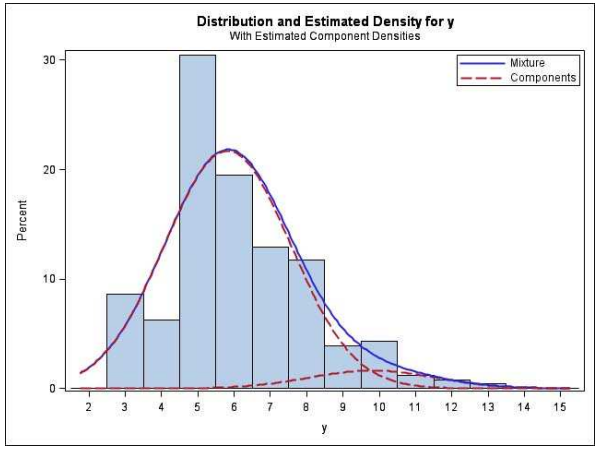
- Summary:

σ^2	ℓ	$\widehat{g}$
3	-519.73	2
2	-514.99	3
1	-510.95	5
0.2	-488.22	9

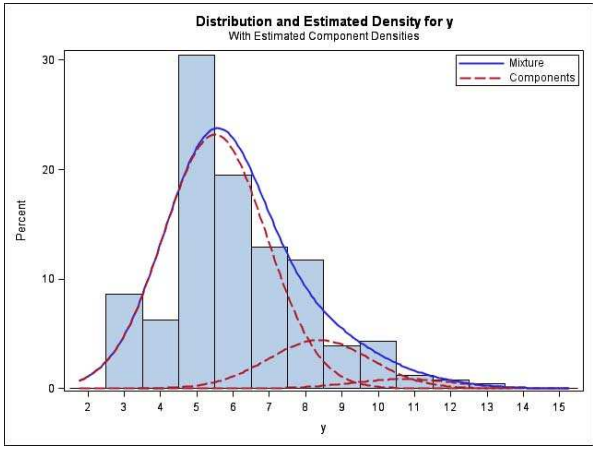
- The number of components ranges from 2 to 9
- The maximized log-likelihood values show considerable variation
- This suggests quite different fits of the models to the observed data

- This is also seen in the resulting mixture densities:

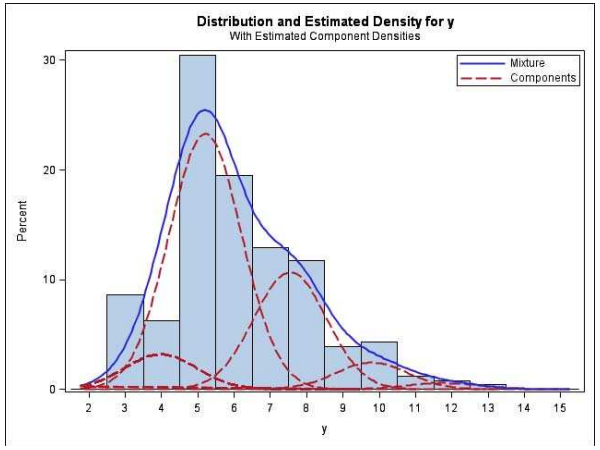
$\sigma^2 = 3$



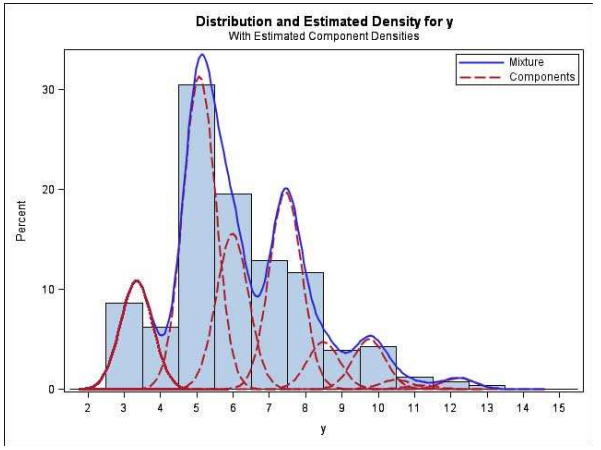
$\sigma^2 = 2$



$\sigma^2 = 1$



$\sigma^2 = 0.2$



- Specifying a large value for σ^2 results in a mixture with a small number of components with much variability, and therefore in a (very) smooth mixture density.

- Specifying a small value for σ^2 results in a mixture with a large number of components with little variability, and therefore in a (very) non-smooth mixture density.

- From this perspective, the above procedure can be viewed as a method for density estimation, with smoothness parameter σ^2.

- Note that the fact that g and σ^2 are not simultaneously identified from the data is due to absence of a variance-mean link in the Gaussian family.

- For other models (Poisson, Binomial, ...), this problem does not occur since the variance is immediately tied to the mean:

$$\begin{aligned} Y \sim \text{Poisson}(\lambda) &\Rightarrow \text{Var}(Y) = \lambda = \text{E}(Y) \\ Y \sim \text{Binomial}(n, p) &\Rightarrow \text{Var}(Y) = np(1-p) = \text{E}(Y)[n - \text{E}(Y)]/n \end{aligned}$$

  which shows that there is no additional parameter which can be used to tune the variability within the mixture components.

Chapter I.9
Classification

- Introduction
- Posterior probabilities
- Examples
- Cluster analysis versus discriminant analysis

I.9.1 Introduction

- We re-consider the Child data, where the response of interest is the number of episodes of acute respiratory infection (fever, cough, running nose,...), recorded within a 3-year period

- The NPMLE for the mixing distribution in a Poisson model was earlier found to be ($ll = -1553.81$):

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} 0 & 0.143 & 2.817 & 8.164 & 16.156 \\ 0.001 & 0.197 & 0.480 & 0.269 & 0.054 \end{pmatrix}$$

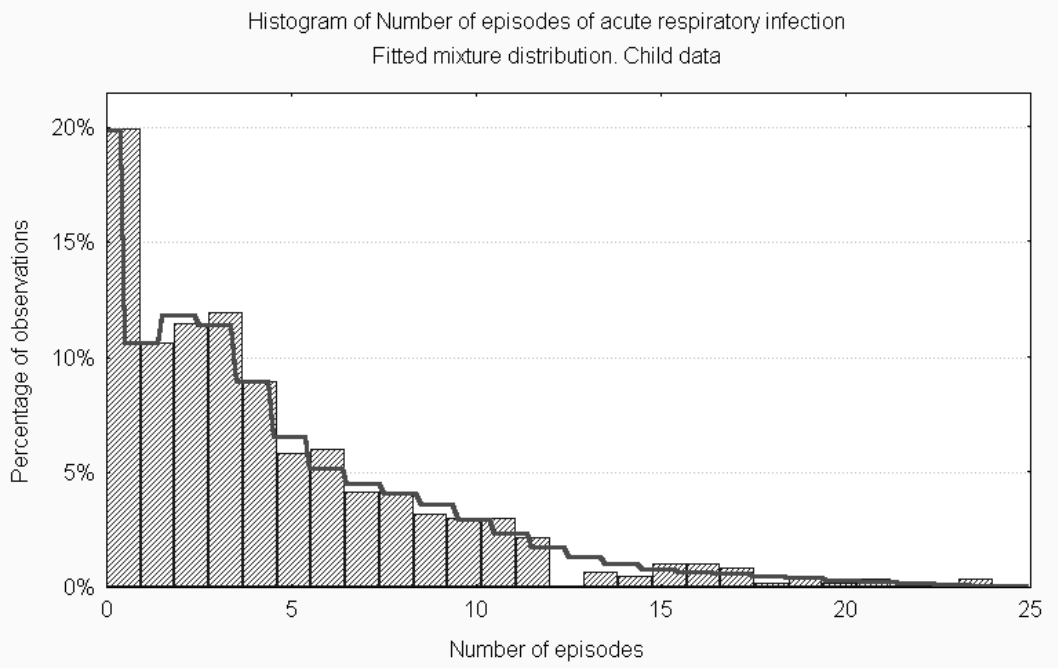
- The extremely small weight for the first component motivates a 4-component mixture.

- The fitted model, obtained before, then becomes ($ll = -1553.81$):

$$Y|\lambda \sim \text{Poisson}(\lambda) \qquad \lambda \sim \begin{pmatrix} 0.143 & 2.817 & 8.164 & 16.156 \\ 0.197 & 0.480 & 0.270 & 0.053 \end{pmatrix}$$

- Note the minor change in log-likelihood (< 0.01).

- Graphical representation:



- As discussed before, a biological interpretation can be given to the components of the mixture:

Component	λ_j	π_j	Interpretation
1	0.143	0.197	almost always healthy
2	2.817	0.480	normal
3	8.164	0.270	above normal
4	16.156	0.053	high risk for infection

- Once a mixture model has been fitted, one might be interested in classifying observations in the different mixture components, i.e., in deciding what component of the mixture a specific observation is most likely to belong to.

- In practice this is often done based on posterior probabilities.

I.9.2 Posterior Probabilities

- Consider the following finite mixture model for the response Y of interest:

$$Y_i \sim \pi_1 f_{i1}(y_i) + \pi_2 f_{i2}(y_i) + \ldots + \pi_g f_{ig}(y_i) = \sum_{j=1}^{g} \pi_j f_{ij}(y_i)$$

- $f_{i1}(y_i), \ldots, f_{ig}(y_i)$ are the density functions of Y_i (possibly depending on unknown parameters $\boldsymbol{\theta}$) in the g components of the mixture

- As in the discussion of the EM algorithm, we define indicators Z_{ij}, $i = 1, \ldots, N$, $j = 1, \ldots, g$:

$$Z_{ij} = \begin{cases} 1 & \text{if observation } i \text{ belongs to component } j \\ 0 & \text{otherwise} \end{cases}$$

- We then have that

$$P(Z_{ij} = 1) \;=\; \pi_j$$

- The component probabilities π_j are therefore often called **prior** probabilities, in the sense that they express how likely the ith subject is to belong to component j, without taking into account the observed response value y_i for that observation.

- The **posterior** probability for observation i to belong to the jth component then equals

$$\begin{aligned}
\pi_{ij} \;&=\; P(Z_{ij} = 1 \mid y_i) \\
&=\; \frac{f_i(y_i \mid Z_{ij} = 1)\; P(Z_{ij} = 1)}{f_i(y_i)} \\
&=\; \frac{\pi_j\; f_{ij}(y_i)}{\sum\limits_j \pi_j f_{ij}(y_i)}
\end{aligned}$$

- π_{ij} expresses how likely the ith subject is to belong to component j, taking into account the observed response value y_i for that observation.

- In practice, the posterior probabilities depend on the unknown parameters $\pi_1, \ldots, \pi_g$ and $\boldsymbol{\theta}$, but once the mixture model has been fitted, these parameters can be replaced by their estimates.

- A natural classification rule now immediately follows:

Classify observation i into component j
if and only if

$$\pi_{ij} = \max_k \{\pi_{ik}\},$$

i.e., classify into the component to which observation i is most likely to belong

I.9.3 Example: Child Data

- We will now classify the child data using the posterior probabilities corresponding to the 4-component Poisson model obtained earlier
- In CAMAN, the procedures 'mixalg.EM' and 'mixalg' automatically calculate posterior probabilities and perform classifications
- The results are saved in the attributes 'prob' and 'classification':

```
em<-mixalg.EM(obs="counts",weights="frequency",family="poisson",data=child,
              t=c(0.5,3,10,15), p=c(0.25,0.25,0.25,0.25), acc=10^(-20))
round(cbind(child,em@prob,em@classification),digits=4)
```

- Result:

counts	frequency	1	2	3	4	em@classification
0	120	0.8557	0.1439	0.0004	0.0000	1
1	64	0.2310	0.7631	0.0059	0.0000	2
2	69	0.0148	0.9635	0.0216	0.0000	2
3	72	0.0007	0.9382	0.0610	0.0000	2
4	54	0.0000	0.8413	0.1585	0.0002	2
5	35	0.0000	0.6463	0.3529	0.0007	2
6	36	0.0000	0.3863	0.6112	0.0025	3
7	25	0.0000	0.1779	0.8156	0.0066	3
8	25	0.0000	0.0690	0.9165	0.0146	3
9	19	0.0000	0.0246	0.9457	0.0298	3
10	18	0.0000	0.0084	0.9335	0.0581	3
11	18	0.0000	0.0027	0.8878	0.1094	3
12	13	0.0000	0.0009	0.8032	0.1959	3
13	4	0.0000	0.0002	0.6743	0.3254	3
14	3	0.0000	0.0001	0.5115	0.4885	3
15	6	0.0000	0.0000	0.3460	0.6540	4
16	6	0.0000	0.0000	0.2110	0.7890	4
17	5	0.0000	0.0000	0.1190	0.8810	4
18	1	0.0000	0.0000	0.0639	0.9361	4
19	3	0.0000	0.0000	0.0334	0.9666	4
20	1	0.0000	0.0000	0.0171	0.9829	4
21	2	0.0000	0.0000	0.0087	0.9913	4
23	1	0.0000	0.0000	0.0022	0.9978	4
24	2	0.0000	0.0000	0.0011	0.9989	4

- In SAS, posterior probabilities and classification results can be saved in an output data set:

```
proc fmm data=child;
model y= / dist=poisson k=4 parms(-0.7, 1.1, 2.3, 2.7);
probmodel  / parms(0,0,0);
output out=out posterior class;
freq w;
run;
```

- Result:

y	w	Post_1	Post_2	Post_3	Post_4	Class
0	120	0.8557	0.1439	0.0004	0.0000	1
1	64	0.2310	0.7631	0.0059	0.0000	2
.	..	.....	.....	.....	.....	2
5	35	0.0000	0.6463	0.3529	0.0007	2
6	36	0.0000	0.3863	0.6113	0.0025	3
.	..	.....	.....	.....	.....	3
14	3	0.0000	0.0001	0.5115	0.4885	3
15	6	0.0000	0.0000	0.3460	0.6539	4
.	..	.....	.....	.....	.....	4
24	2	0.0000	0.0000	0.0011	0.9989	4

- All zero counts are classified in the first component, i.e., in the component which can be interpreted as the group of children which are almost always healthy.

- Counts in the range $[1, 5]$ are classified in the second component, i.e., in the component which can be interpreted as the group of children with normal risk for acute respiratory infection.

- Counts in the range $[6, 14]$ are classified in the third component, i.e., in the component which can be interpreted as the group of children with increased risk for acute respiratory infection.

- Counts which are at least 15 are classified in the last component, i.e., in the component which can be interpreted as the group of children with high risk for acute respiratory infection.

- The proportion of children classified in each component equals:

Component	# Children	Proportion	$\widehat{\pi}_j$
1	120	0.20	0.197
2	294	0.49	0.480
3	161	0.27	0.270
4	27	0.04	0.053
	602	1	1

- Note that these proportions are very close to the estimated component probabilities $\widehat{\pi}_j$.

I.9.4 Example: SIDS Data

- We re-consider the SIDS data, with a Poisson model for the observed rates of SIDS in 100 counties in North-Carolina

- The NPMLE obtained before is given by:

$$Y_i|p \sim \text{Poisson}(p\, n_i) \qquad p \sim \begin{pmatrix} 0.0013 & 0.0021 & 0.0037 & 0.0090 \\ 0.33 & 0.51 & 0.15 & 0.01 \end{pmatrix}$$

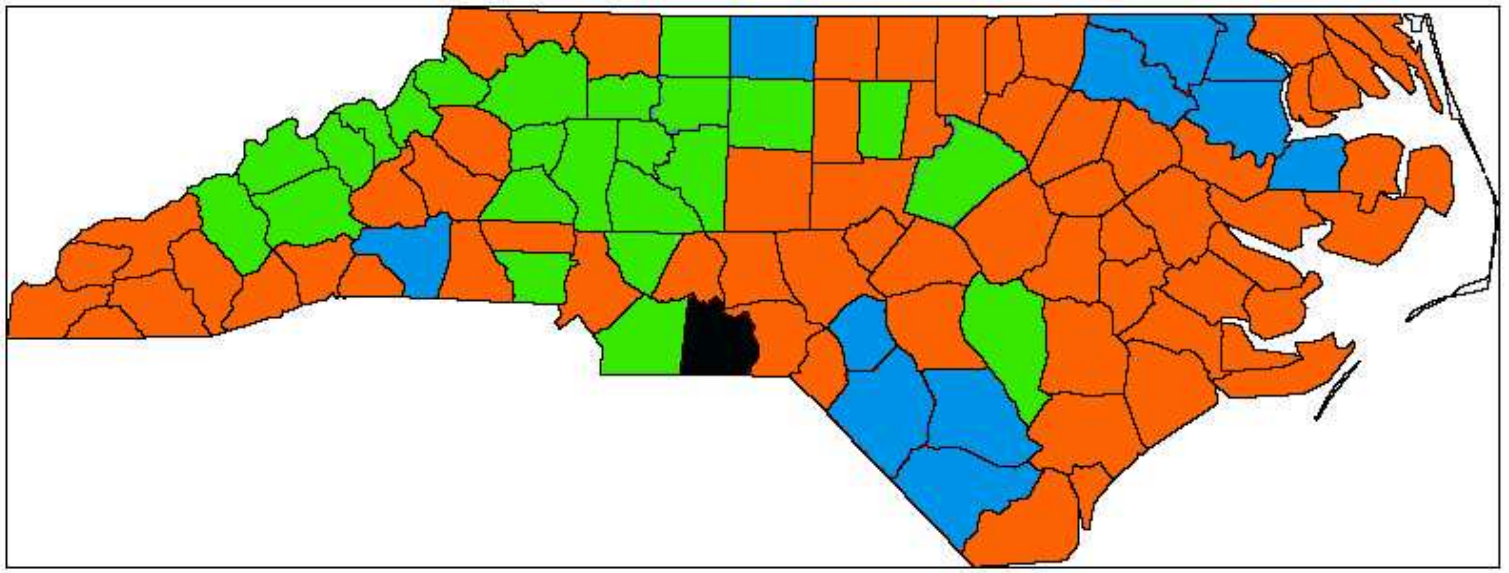
- The fitted mixture can now be used to classify the 100 counties in either one of the 4 mixture components

- We obtain the following classification results:

Component	# Counties	Proportion	$\widehat{\pi}_j$
1	24	0.24	0.32
2	64	0.64	0.52
3	11	0.11	0.15
4	1	0.01	0.01
	100	1	1

- In practice, such classifications are often used to create so-called disease maps, i.e., geographical maps in which the different regions are represented according to their risk for certain 'diseases'.

- For the SIDS example, this results in the following map:



- Classification legend for the disease map:

Component	# Counties	$\widehat{\pi}_j$	Color	Component	# Counties	$\widehat{\pi}_j$	Color
1	24	0.33	green	3	11	0.15	blue
2	64	0.51	orange	4	1	0.01	black

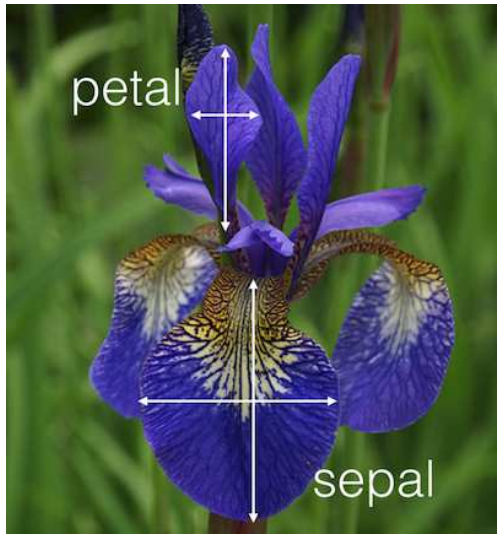
I.9.5 Example: Iris Data

- So far, classification was done based on a fitted mixture model, i.e., the model was used to classify observations in **detected clusters**.
- Mixture models can also be used to classify observations in **known groups**.
- As an example, we consider Fisher's Iris data set, and restrict attention to the Versicolor and Virginica species:

Versicolor

Virginica

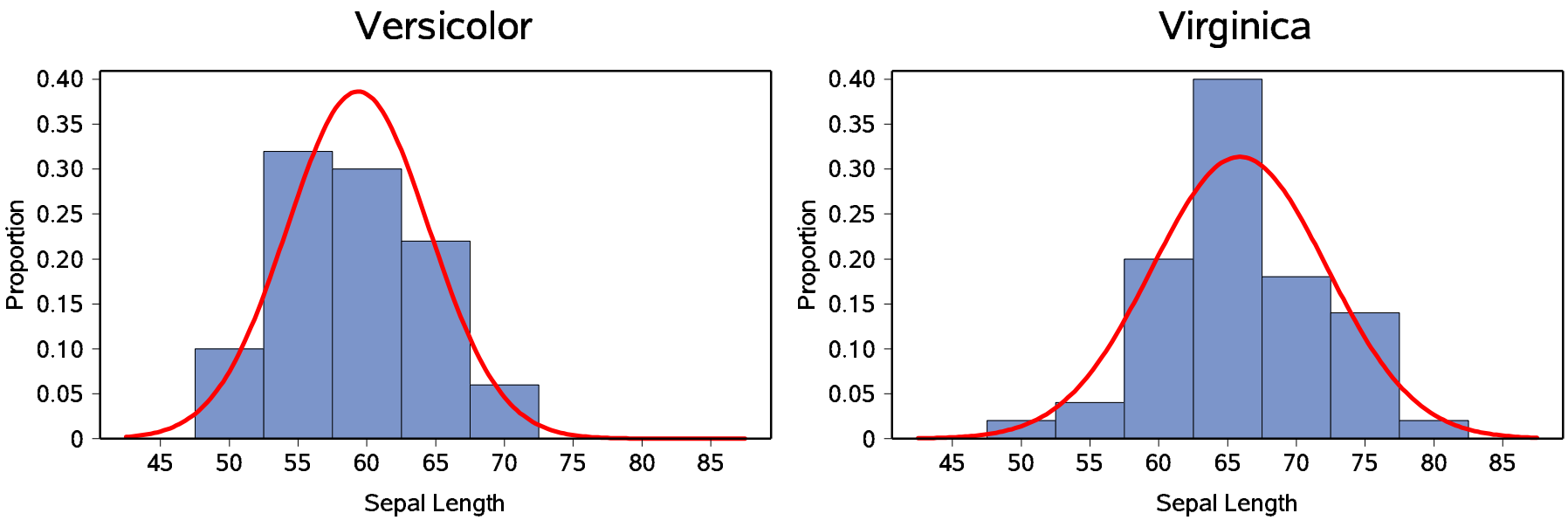
- We focus on the outcome Sepal Length ($= Y$):



- We have data for 50 flowers of both types:

Type	Number	Mean	Stand.Dev.	Variance
Versicolor	50	59.36	5.16	26.64
Virginica	50	65.88	6.36	40.43

- Histogram in both samples:



- The distribution for the sepal length of a randomly selected flower from one of the two species is:

$$Y \sim \pi\ N(59.36, 5.16^2) + (1-\pi)\ N(65.88, 6.36^2)$$

- π is the probability that the flower is of the Versicolor type, i.e. the proportion of Versicolor flowers in the general population of Versicolor and Virginica flowers.

- The mixture model can be used to classify new flowers, provided an 'estimate' for π is available.

- Note that classification is then based on a 2-component mixture which is not fitted as such to the available data.

- Since it was decided by design to select 50 flowers of each type, π cannot be estimated from the data set at hand.

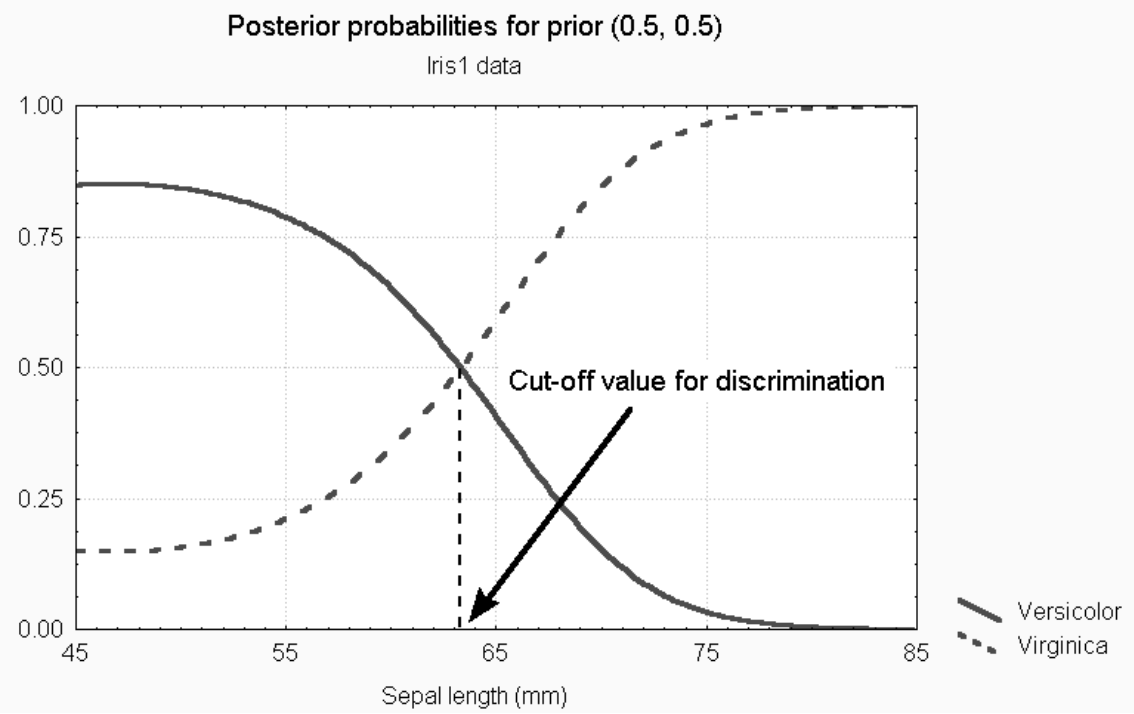
- If the sample would have been a random sample of 100 flowers, which happens to contain 50 flowers of each type, π could be set equal to 0.5

- As before, classification is based on posterior probabilities.

- In our example, the ith flower would be classified into the Versicolor group if

$$\pi_{i1} \geq \pi_{i2} \quad \Leftrightarrow \quad \pi_{i1} = \frac{\pi \ f_{i1}(y_i)}{\pi f_{i1}(y_i) + (1-\pi) f_{i2}(y_i)} \geq 0.5$$

- The posterior probabilities for both groups, as functions of the sepal length, assuming equal prior probability for both groups (i.e., $\pi = 0.5$) are:



- Flowers with sepal length not larger than $63mm$ are classified into the 'Versicolor' group, otherwise they are classified into the 'Virginica' group.

- If we use this cut-off value to classify the flowers in our data set, we obtain the following classification table:

		Reality	
		Versicolor	Virginica
Classification	Versicolor	39	19
	Virginica	11	31
		50	50

- The above table can be used to estimate the error rate for the classification of future flowers:

$$P(\text{Flower wrongly classified}) = 0.5 \times P(\text{Flower wrongly classified} \mid \text{of Versicolor type})$$

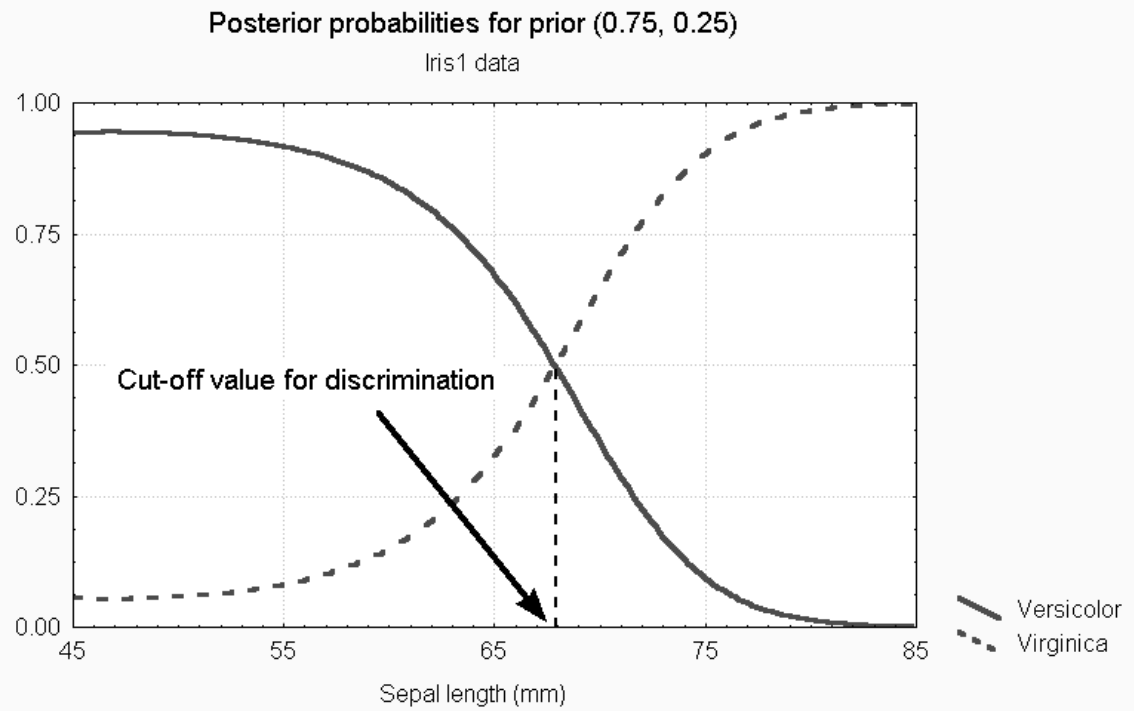
$$+ 0.5 \times P(\text{Flower wrongly classified} \mid \text{of Virginica type})$$

$$= 0.5 \times \frac{11}{50} + 0.5 \times \frac{19}{50} = 0.30$$

- Hence, it is to be expected that, using the derived cut-off value, 30% of all flowers would be wrongly classified.

- To illustrate that the cut-off value and hence also the error rate highly depends on the prior probabilities π and $1 - \pi$, we repeat the calculations assuming that there are three times as many Versicolor flowers as Virginica flowers, i.e., for $\pi = 0.75$.

- The posterior probabilities for both groups, as functions of the sepal length, now become:



- Flowers with sepal length smaller than $68mm$ are classified into the 'Versicolor' group, otherwise they are classified into the 'Virginica' group.

- As was to be expected, more flowers will be classified into the 'Versicolor' group
- If we use this cut-off value to classify the flowers in our data set, we obtain the following classification table:

Classification	Reality: Versicolor	Reality: Virginica
Versicolor	47	33
Virginica	3	17
	50	50

- In comparison to our first analysis, many more Virginica flowers are wrongly classified, while the Versicolor flowers are now much better classified.

- The above table can again be used to estimate the error rate for the classification of future flowers:

$$P(\text{Flower wrongly classified}) = 0.75 \times P(\text{Flower wrongly classified} \mid \text{of Versicolor type})$$
$$+ 0.25 \times P(\text{Flower wrongly classified} \mid \text{of Virginica type})$$
$$= 0.75 \times \frac{3}{50} + 0.25 \times \frac{33}{50} = 0.21$$

- Hence, only 21% of all flowers are now expected to be wrongly classified

- Note that the above estimates for the error rates are likely to be over-optimistic as they are obtained from testing the discriminant rule with the same observations as those used to construct the rule.

- More realistic estimates can be obtained using 'training' and 'test' data, or using cross-validation. This will not be discussed here any further.

I.9.6 Cluster Analysis versus Discriminant Analysis

- Using the Child data and the SIDS data, it has been illustrated how observations can be classified in clusters which were detected using NPMLE's.
- In those analyses, the first step was to check whether there is underlying heterogeneity. Afterwards, the observations were classified into the different mixture components.
- This is called **cluster analysis**
- There also exist other approaches to cluster analysis, which are not based on finite mixtures

- Often, as was the case for the Iris data, one is interested in finding 'optimal' rules for classifying observations (possibly future observations) in known groups.
- This is called **discriminant analysis**
- There also exist other approaches to discriminant analysis, which are not based on finite mixtures

Chapter I.10
Model Extensions

- Introduction
- Case study: MMSE data

I.10.1 Introduction

- Mixture models can be used to describe latent heterogeneity
- In all examples so far, interest was in describing the distribution of a single outcome Y
- Mixture models can be incorporated in statistical models as well, to account for heterogeneity not explained by covariates included in the model.
- This will be illustrated in a Binomial regression model, but equally well is applicable in other contexts
- Due to the flexibility of the SAS procedure FMM, all analyses will be performed in SAS

I.10.2 Case Study: MMSE Data

- We consider data from 58 elderly hip fracture patients, treated at the University Hospital Gasthuisberg in Leuven, between September 16, 1996, and February 28, 1997.
- Of interest is the Mini Mental State Examination (MMSE) score.
- The MMSE is the number of correctly answered questions, out of 30.
- High MMSE values indicate good cognitive functioning, while low MMSE values indicate bad cognitive functioning.
- Of interest is the relation between age and MMSE one day after hip surgery

- Descriptive statistics:

Outcome	Mean	Stand.Dev.	Minimum	Maximum
Age	78.71	8.20	65	95
MMSE	18.88	8.32	0	30

- A natural model is a Binomial logistic model (Model 1):

$$\text{MMSE}_i \sim \text{Binomial}(30, p_i), \qquad \ln\left[\frac{p_i}{1-p_i}\right] = \beta_0 + \beta_1 \text{Age}_i$$

- Results:

Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	6.8862	0.5435	5.8210	7.9514	160.54	<.0001
age	1	-0.0801	0.0068	-0.0933	-0.0668	140.29	<.0001
Scale	0	1.0000	0.0000	1.0000	1.0000		

- Allowing the scale parameter to deviate from one (Pearson χ^2 method) yields:

Parameter	DF	Estimate	Standard Error	Wald 95% Confidence Limits		Wald Chi-Square	Pr > ChiSq
Intercept	1	6.8862	1.5138	3.9192	9.8531	20.69	<.0001
age	1	-0.0801	0.0188	-0.1170	-0.0432	18.08	<.0001
Scale	0	2.7853	0.0000	2.7853	2.7853		

NOTE: The scale parameter was estimated by the square root of Pearson's Chi-Square/DOF.

- There is strong evidence for overdispersion.
- Clinicians hypothesize that part of the overdispersion can be explained from the fact that some patients are neuro-psychiatric at admission, while others are not.
- Correction for the neuro-psychiatric status is only possible if it was recorded
- Alternatively, mixture models can be used as an attempt to account for this heterogeneity in the population studied.

- A 2-component mixture with component-specific regression coefficients (Model 2):

$$\text{MMSE}_i \sim \pi\ \text{Binomial}(30, p_{1i}) + (1-\pi)\ \text{Binomial}(30, p_{2i})$$

$$\ln\left[\frac{p_{1i}}{1-p_{1i}}\right] = \beta_{10} + \beta_{11}\text{Age}_i, \qquad \ln\left[\frac{p_{2i}}{1-p_{2i}}\right] = \beta_{20} + \beta_{21}\text{Age}_i$$

- The model assumes that, at each age, the population consists of two sub-populations:
  - ▷ The proportion of each sub-population does not depend on age
  - ▷ The probability to correctly answer an MMSE item is age-specific
  - ▷ The relation between age and correctly answering an MMSE item is different for both sub-populations

- SAS code:

```
proc fmm data=test ;
model mmse/n = age /  dist=binomial k=2;
run;
```

- Relevant SAS output:

Parameter Estimates for 'Binomial' Model

Component	Effect	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	8.6137	0.7686	11.21	<.0001
1	age	-0.09505	0.009294	-10.23	<.0001
2	Intercept	11.4677	1.7088	6.71	<.0001
2	age	-0.1624	0.02319	-7.00	<.0001

Parameter Estimates for Mixing Probabilities

	Linked Scale				
Effect	Estimate	Standard Error	z Value	Pr > |z|	Probability
Intercept	1.2197	0.3294	3.70	0.0002	0.7720

- A simplified model is obtained by assuming the age effects to be the same for both sub-populations (Model 3):

$$\text{MMSE}_i \sim \pi\ \text{Binomial}(30, p_{1i}) + (1-\pi)\ \text{Binomial}(30, p_{2i})$$

$$\ln\left[\frac{p_{1i}}{1-p_{1i}}\right] = \beta_{10} + \beta_1 \text{Age}_i, \qquad \ln\left[\frac{p_{2i}}{1-p_{2i}}\right] = \beta_{20} + \beta_1 \text{Age}_i$$

- The model assumes that, at each age, the population consists of two sub-populations:
  - ▷ The proportion of each sub-population does not depend on age
  - ▷ The probability to correctly answer an MMSE item is age-specific
  - ▷ The relation between age and correctly answering an MMSE item is the same for both sub-populations

- SAS code:

```
proc fmm data=test ;
model mmse/n = age /  dist=binomial k=2;
restrict age 1, age -1;
run;
```

- The RESTRICT statement allows specification of linear equality or inequality constraints:
  - ▷ Fixing a parameter at a particular value
  - ▷ Equating parameters in different components in a mixture
  - ▷ Imposing order conditions on parameters
  - ▷ Specifying contrasts among parameters
- The above RESTRICT statement is equivalent to:

```
restrict age 1, age -1 = 0;
```

- Restrictions for effects in specific mixture components are separated by commas.
- Many options possible, see SAS help function
- Relevant SAS output:

Parameter Estimates for 'Binomial' Model

Component	Effect	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	9.1069	0.9879	9.22	<.0001
1	age	-0.1007	0.01206	-8.35	<.0001
2	Intercept	6.9031	0.9396	7.35	<.0001
2	age	-0.1007	0.01206	-8.35	<.0001

Parameter Estimates for Mixing Probabilities

	Linked Scale				
Effect	Estimate	Standard Error	z Value	Pr > |z|	Probability
Intercept	1.1527	0.3239	3.56	0.0004	0.7600

- Allowing the mixture weights π and $1-\pi$ in Model 2 to depend on age can be obtained as follows (Model 4):

$$\text{MMSE}_i \sim \pi_i \text{ Binomial}(30, p_{1i}) + (1-\pi_i) \text{ Binomial}(30, p_{2i})$$

$$\ln\left[\frac{p_{1i}}{1-p_{1i}}\right] = \beta_{10} + \beta_{11}\text{Age}_i, \qquad \ln\left[\frac{p_{2i}}{1-p_{2i}}\right] = \beta_{20} + \beta_{21}\text{Age}_i$$

$$\ln\left[\frac{\pi_i}{1-\pi_i}\right] = \alpha_0 + \alpha_1\text{Age}_i$$

- The model assumes that, at each age, the population consists of two sub-populations:
  - ▷ The proportion of each sub-population depends on age
  - ▷ The probability to correctly answer an MMSE item is age-specific
  - ▷ The relation between age and correctly answering an MMSE item is different for both sub-populations

- SAS code:

```
data test;
set test;
agec=age-80;
run;

proc fmm data=test ;
model mmse/n = agec /  dist=binomial k=2;
probmodel age  / parms(1.2 0);
run;
```

- Good starting values are needed for the parameters in the model for the component weights

- In order to be able to use the results from Model 2 as starting values, the age covariate was centered at 80 years in the model for π_i

- Relevant SAS output:

Parameter Estimates for 'Binomial' Model

Component	Effect	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	8.7175	0.7851	11.10	<.0001
1	age	-0.09629	0.009485	-10.15	<.0001
2	Intercept	11.7760	1.7591	6.69	<.0001
2	age	-0.1663	0.02362	-7.04	<.0001

Parameter Estimates for Mixing Probabilities

Effect	Estimate	Standard Error	z Value	Pr > |z|
Intercept	1.2644	0.3397	3.72	0.0002
agec	0.03735	0.04282	0.87	0.3831

- Summary of results:

Effect		Parameter	Model 1	Model 2	Model 3	Model 4
Component 1:	Intercept	β_{10}	6.886 (0.544)	8.614 (0.769)	9.107 (0.988)	8.718 (0.785)
	Age	β_{11}	-0.080 (0.007)	-0.095 (0.009)	-0.101 (0.012)	-0.096 (0.009)
Component 2:	Intercept	β_{20}		11.468 (1.709)	6.903 (0.940)	11.776 (1.759)
	Age	β_{21}		-0.162 (0.023)	-0.101 (0.012)	-0.166 (0.024)
Weight 1:	Probability	π		0.772	0.760	
	Intercept	α_0		1.220 (0.329)	1.153 (0.324)	1.264 (0.340)
	Age	α_1				0.037 (0.043)
Deviance:		$-2\ell\ell$	662.0	412.9	421.3	412.1

- Of all models fitted, Model 2 is best supported by the data

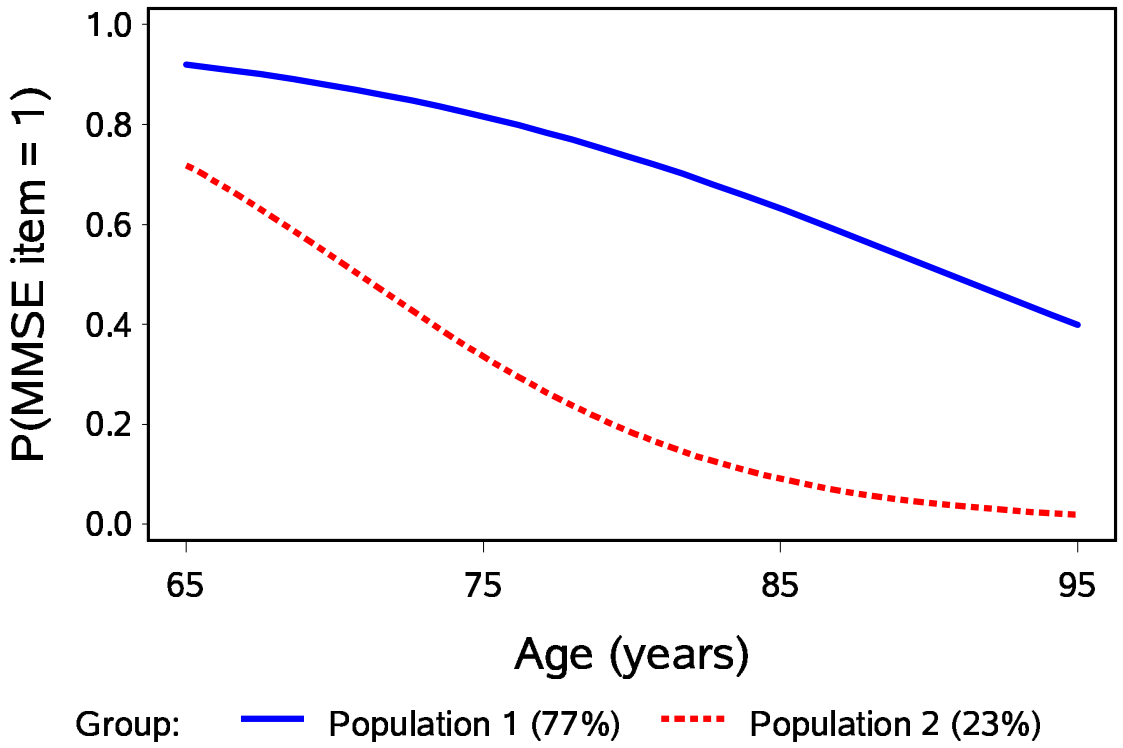
- Interpretation:
  - ▷ Two sub-populations representing 77% and 23% of the population, respectively
  - ▷ The proportion of each sub-population does not change with age
  - ▷ The probability of correctly answering an MMSE item decreases with age
  - ▷ This decrease is significantly steeper in the second sub-population than in the first

- Fitted probabilities to correctly answer an MMSE item:

$$\ln\left[\frac{p_{1i}}{1-p_{1i}}\right] = 8.614 - 0.095\text{Age}_i \quad \text{(Population 1)}$$

$$\ln\left[\frac{p_{2i}}{1-p_{2i}}\right] = 11.468 - 0.162\text{Age}_i \quad \text{(Population 2)}$$

- Graphical representation:



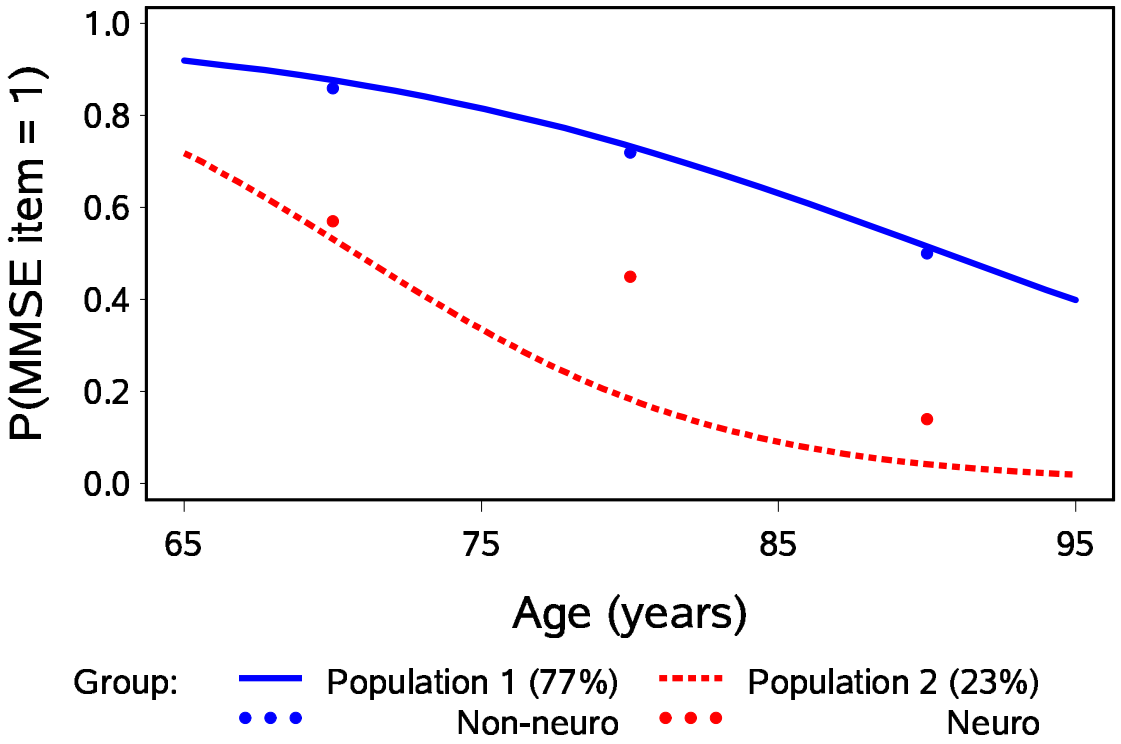
- At any age, subjects in population 1 are more likely to correctly answer MMSE items than subjects in population 2

- An informal check whether the sub-populations detected truly correspond to patients who are (not) neuro-psychiatric, observed proportions are calculated for both groups, and for specific age intervals:

	Not neuro-psychiatric		Neuro-psychiatric	
Age range	Average MMSE	$P(\text{MMSE item} = 1)$	Average MMSE	$P(\text{MMSE item} = 1)$
$[65, 75]$	25.88	25.88/30=0.86	17.00	17.00/30=0.57
$]65, 85]$	21.58	21.58/30=0.72	13.40	13.40/30=0.45
$]85, 95]$	14.91	14.91/30=0.50	4.33	4.33/30=0.14

- These observed proportions can now be graphically compared to the fitted probabilities to correctly answer an MMSE item

- Result:



- Success rates for patients who are not neuro-psychiatric are well described by the first mixture component

- Success rates for neuro-psychiatric patients are less well described by the second mixture component

- This is also observed when cross-classifying the neuro-status with the classification based on posterior probabilities:

Classification	Neuro-psychiatric ? Yes	Neuro-psychiatric ? No	
Component 1	9	36	45
Component 2	9	4	13
	18	40	

- Also, when the model is corrected for neuro-status, there still is evidence for the presence of two mixture components:

```
proc fmm data=test;
model mmse/n = age neuro neuro*age
               /dist=binomial k=1;
run;
```

```
proc fmm data=test;
model mmse/n = age neuro neuro*age
               /dist=binomial k=2;
run;
```

- Relevant SAS output for 1-component mixture:

Fit Statistics

-2 Log Likelihood	542.4

Parameter Estimates for 'Binomial' Model

Effect	Estimate	Standard Error	z Value	Pr > |z|
Intercept	7.4909	0.6669	11.23	<.0001
age	-0.08238	0.008214	-10.03	<.0001
NEURO	-1.5639	1.3462	-1.16	0.2453
age*NEURO	0.004425	0.01682	0.26	0.7925

- Relevant SAS output for 2-component mixture:

Fit Statistics

-2 Log Likelihood 373.3

Parameter Estimates for 'Binomial' Model

Component	Effect	Estimate	Standard Error	z Value	Pr > |z|
1	Intercept	6.1356	1.0411	5.89	<.0001
1	age	-0.05973	0.01303	-4.58	<.0001
1	NEURO	-0.3562	1.8429	-0.19	0.8467
1	age*NEURO	-0.00833	0.02303	-0.36	0.7176
2	Intercept	5.9603	1.7478	3.41	0.0006
2	age	-0.07600	0.01978	-3.84	0.0001
2	NEURO	4.5602	3.2644	1.40	0.1624
2	age*NEURO	-0.08705	0.04215	-2.07	0.0389

Parameter Estimates for Mixing Probabilities

	Linked Scale				
Effect	Estimate	Standard Error	z Value	Pr > |z|	Probability
Intercept	0.9591	0.4082	2.35	0.0188	0.7229

- Conclusion:

> The mixture components do not coincide with the neuro-psychiatric and non-neuro-psychiatric subpopulations

- Alternative conclusion:

> The neuro-psychiatric status does not entirely explain the presence of mixture components

Part II

Non-linear Models

Chapter II.1
Non-Linear Mixed Models

- From linear to non-linear models
- Orange Tree Example
- Song Bird Example