- Pre-calculation

$(y_i, p_i),\ i = 1, ..m$ is the data with $z = 1$; $m = \sum_{z_i=1} z_i$ is the number of $z = 1$;

$(y_j, p_j)\ j = m+1, ..n$ is the data with $z = 0$; $n - m$ is the number of $z = 0$.

Mixing proportions $p_i = Pr(z = 1)$; $u = \frac{y-\mu_2}{\sigma}$; $1 - p_j = Pr(z = 0)$; $v = \frac{y-\mu_1}{\sigma}$. $\vec{\theta} = (\mu_1, \mu_2, \sigma)$

$$\phi(u;\theta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y-\mu_2}{\sigma})^2}; \quad \phi(v;\theta) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y-\mu_1}{\sigma})^2}$$

$$\frac{\partial\phi(u)}{\partial(\mu_1,\mu_2,\sigma)} = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y-\mu_2}{\sigma})^2} (\frac{y-\mu_2}{\sigma}) \begin{bmatrix} 0 \\ 1/\sigma \\ u/\sigma \end{bmatrix} = \phi(u)\frac{u}{\sigma} \begin{bmatrix} 0 \\ 1 \\ u \end{bmatrix}$$

$$\frac{\partial\phi(v)}{\partial(\mu_1,\mu_2,\sigma)} = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y-\mu_1}{\sigma})^2} (\frac{y-\mu_1}{\sigma}) \begin{bmatrix} 1/\sigma \\ 0 \\ v/\sigma \end{bmatrix} = \phi(v)\frac{v}{\sigma} \begin{bmatrix} 1 \\ 0 \\ v \end{bmatrix}$$

i) EM Algorithm:

E-Step

$$Pr(\vec{\theta}|\vec{y},\vec{z}) = \prod_{i=1}^{n} \left( \frac{z_i}{\sigma}\phi(\frac{y_i-\mu_2}{\sigma})^{z_i} p_i^{z_i} + \frac{1-z_i}{\sigma}\phi(\frac{y_j-\mu_1}{\sigma})^{1-z_i}(1-p_i)^{1-z_i} \right) = \prod_{i=1}^{m} \frac{1}{\sigma}\phi(u_i)^1 p_i^1 \cdot \prod_{j=m+1}^{n} \frac{1}{\sigma}\phi(v_j)^1(1-p_j)^1$$

$$\log(Pr(\vec{\theta}|\vec{y},\vec{z}^\star)) = \sum_{i=1}^{m} \left[\log Pr(\vec{\theta}|y_i, z_i^\star = 1)\right] + \sum_{j=m+1}^{n} \left[\log Pr(\vec{\theta}|y_j, z_j^\star = 0)\right]$$

$$= C - n\log(\sigma) + \sum_{i=1}^{m} \log\phi(u_i) + \sum_{j=m+1}^{n} \log\phi(v_j) + \sum_{i=1}^{m} \log p_i + \sum_{j=m+1}^{n} \log(1-p_j)$$

$$Q(\vec{\theta},\vec{\theta}^\star) = \frac{1}{S}\sum_{s=1}^{S} \log(Pr(\vec{\theta}|\vec{y},\vec{z}^{(s)})) = C - n\log(\sigma) + \sum_{i=1}^{\bar{m}} \log\phi(u_i) + \sum_{j=\bar{m}+1}^{n} \log\phi(v_j) + \sum_{i=1}^{\bar{m}} \log p_i + \sum_{j=\bar{m}+1}^{n} \log(1-p_j)$$

where $\bar{m} = \frac{1}{S}\sum_{s=1}^{S}\sum_{z_i=1} z_i^{(s)}$

M-Step

$$\frac{\partial Q(\vec{\theta},\vec{\theta}^\star)}{\partial\theta} = \begin{bmatrix} 0 \\ 0 \\ \frac{-n}{\sigma} \end{bmatrix} + \sum_{i=1}^{m} \frac{\phi(u_i)}{\phi(u_i)} \begin{bmatrix} 0 \\ u_i/\sigma \\ u_i^2/\sigma \end{bmatrix} + \sum_{j=m+1}^{n} \frac{\phi(v_j)}{\phi(v_j)} \begin{bmatrix} v_j/\sigma \\ 0 \\ v_j^2/\sigma \end{bmatrix} = \frac{1}{\sigma} \begin{bmatrix} \sum_{j=m+1}^{n} v_j \\ \sum_{i=1}^{m} u_i \\ \sum_{i=1}^{m} u_i^2 + \sum_{j=m+1}^{n} v_j^2 - n \end{bmatrix} \stackrel{set}{=} 0$$

$$\begin{bmatrix} \hat{\mu}_1 = & \frac{1}{n-m}\sum_{j=m+1}^{n} y_j \\ \hat{\mu}_2 = & \frac{1}{m}\sum_{i=1}^{m} y_j \\ \hat{\sigma}^2 = & \frac{1}{n}\left(\sum_{i=1}^{m}(y_i-\mu_2)^2 + \sum_{j=m+1}^{n}(y_j-\mu_1)^2\right) \end{bmatrix}$$

$$r = Pr(z = 1|\vec{y},\vec{\theta}^\star) = \frac{\phi(u^\star)p}{\phi(u^\star)p + \phi(v^\star)(1-p)}$$

```
EM_Mix_Normal<- function(y,S,crit,itera)
{
## "y" (w,p) is the data; "n" is the data size
## "theta" is the parameter vector: mu1,mu2, sigma.
## "thetastar" is the current parameter estimate.
## "itera" is the upper limit of iterations.
##########################################################
s <- 0 ; set.seed(121) # iteration counter
n <- nrow(y)
z <- ifelse (p>=median(p),1,0)
V<- w[which(z==0)]
U<- w[which(z==1)]
mu <- c(mean(V),mean(U))
sigma <- sd(w)
thetastar <- c(mu,sigma)# initial parameter values
repeat {
v <- (w-mu[1])/sigma
u <- (w-mu[2])/sigma
r <- dnorm(u)*p/(dnorm(v)*(1-p)+dnorm(u)*p)
Z <- replicate(S,rbinom(20, 1, r))
W <- replicate(S,w)
U <- W*Z
V <- W*(1-Z)
V <- V[which(V!=0)]
U <- U[which(U!=0)]
mu <- c(mean(V),mean(U))
sigma <- sqrt(sum(c(V-mu[1],U-mu[2])^2)/n/S)
theta <- c(mu,sigma)
s <- s +1
if( (abs(thetastar[1]-theta[1])< crit)| (s > itera)) #  # (sum(thetastar-theta)^2< crit)|
break
thetastar <- theta
}
return(list(s,theta,cbind(y,z,r)))
}
Theta<- EM_Mix_Normal(y,S,crit=1e-4,itera=100)
```

In previous approach, if responsibilities $r \geq p_i$, this observation is more likely $z = 1$ given $y$ and $\theta$ values. Else, let $z_j = 0$.

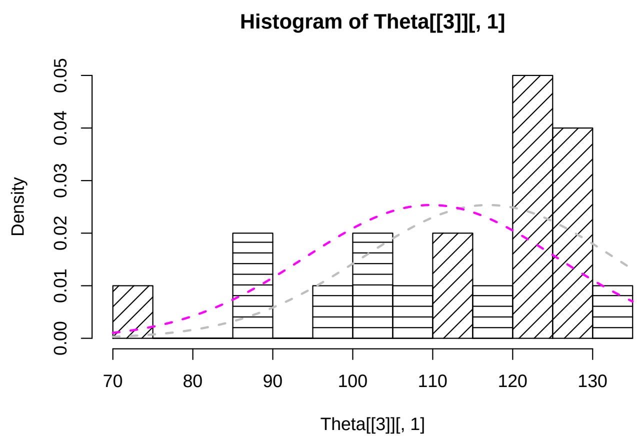
Then update the $\theta$ with preivous $\vec{z}$.Repeat the iteration until converge.

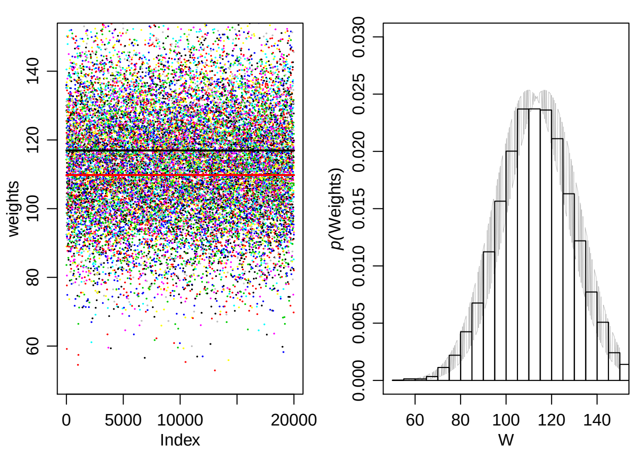
In current approach, we draw many times of $\vec{z}$ from Bernoulli($\vec{r}$) (or, equivalently, Binomial $(1, \vec{r})$) and get the mean values. The convergent result is $\theta = (\mu_1, \mu_2, \sigma) =$ 116.95300667, 109.761061, 15.7341892 respectively.

| | x |
|---|---|
| Iterations | 36 |

| | x |
|---|---|
| mu1 | 116.953007 |
| mu2 | 109.761061 |
| sigma | 15.734189 |

| w | p | z | r |
|---|---|---|---|
| 101.76 | 0.79 | 1 | 0.84048551 |
| 124.76 | 0.59 | 1 | 0.50816558 |
| 85.96 | 0.70 | 1 | 0.83797740 |
| 72.29 | 0.40 | 0 | 0.68732592 |
| 112.69 | 0.32 | 0 | 0.32422951 |
| 98.99 | 0.52 | 0 | 0.62185733 |
| 127.63 | 0.26 | 0 | 0.18836779 |
| 125.54 | 0.44 | 0 | 0.35546013 |
| 126.32 | 0.62 | 1 | 0.52820137 |
| 121.04 | 0.17 | 0 | 0.14077715 |
| 123.06 | 0.91 | 1 | 0.88408796 |
| 127.73 | 0.69 | 1 | 0.59448356 |
| 117.89 | 0.24 | 0 | 0.21680466 |
| 109.12 | 0.51 | 0 | 0.54068350 |
| 123.65 | 0.64 | 1 | 0.56864204 |
| 123.75 | 0.44 | 0 | 0.36746240 |
| 134.77 | 0.73 | 1 | 0.59206331 |
| 88.54 | 0.87 | 1 | 0.93226154 |
| 103.99 | 0.23 | 0 | 0.28167262 |
| 110.73 | 0.95 | 1 | 0.95350291 |




ii) Louis'Method:

By simulation method, We draw $S$ times of $\vec{z} \sim$ Binomial $(1, \vec{r})$.

Then we can approximate the complete information and missing information.

- Complete Information:

$$\frac{\partial}{\partial\theta}\log(p(\vec{\theta}|y,z)) = \frac{1}{\sigma}\begin{bmatrix} \sum_{j=m+1}^{n} v_j \\ \sum_{i=1}^{m} u_i \\ \sum_{i=1}^{m} u_i^2 + \sum_{j=m+1}^{n} v_j^2 - n \end{bmatrix}$$

$$\frac{\partial^2}{\partial\theta^2}\log(p(\vec{\theta}|y,z)) = \frac{-1}{\sigma^2}\begin{bmatrix} n-m & 0 & 2\sum_{j=m+1}^{n} v_j \\ 0 & m & 2\sum_{i=1}^{m} u_i \\ 2\sum_{j=m+1}^{n} v_j & 2\sum_{i=1}^{m} u_i & 3(\sum_{i=1}^{m} u_i^2 + \sum_{j=m+1}^{n} v_j^2) - n \end{bmatrix}$$

$$E[\frac{\partial^2}{\partial\theta^2}\log(p(\vec{\theta}|y,z))] = \frac{1}{S}\sum_{1}^{S}\frac{-1}{\sigma^2}\begin{bmatrix} n-m & 0 & 2\sum_{j=m+1}^{n} v_j \\ 0 & m & 2\sum_{i=1}^{m} u_i \\ 2\sum_{j=m+1}^{n} v_j & 2\sum_{i=1}^{m} u_i & 3(\sum_{i=1}^{m} u_i^2 + \sum_{j=m+1}^{n} v_j^2) - n \end{bmatrix}$$

```r
set.seed(121)
# r <- dnorm(u)*p/(dnorm(v)*(1-p)+dnorm(u)*p)
Z <- replicate(S,rbinom(20, 1, r))
W <- replicate(S,w)
V <- W*(1-Z)
U <- W*Z
V <- V[which(V!=0)]
U <- U[which(U!=0)]
v <- (V-mu[1])/sigma
u <- (U-mu[2])/sigma
(Compy <- matrix(c(sum(Z==0),   0      , 2*sum(v),
                     0,    sum(Z)     , 2*sum(u),
               2*sum(v), 2*sum(u) , 3*(sum(u^2)+sum(v^2))-n*S
),3,3,)/S/sigma^2)  # Complete Information
##                 [,1]            [,2]            [,3]
## [1,] 0.0364884882284 0.0000000000000 0.0000048510812
## [2,] 0.0000000000000 0.0442984705212 0.0000064707178
## [3,] 0.0000048510812 0.0000064707178 0.1615750279796
```

- Missing Information:

$$Var\left[\frac{\partial}{\partial\theta}\log(p(\vec{\theta}|y,z))\right] = \frac{1}{S}\sum_{1}^{S}\left(\frac{\partial}{\partial\theta}\log(p(\vec{\theta}|y,z))\bigg|_{\hat{\theta}}\right)^2 = \frac{1}{S}\sum_{1}^{S}\frac{1}{\sigma^2}\begin{bmatrix}\sum_{j=m+1}^{n} v_j \\ \sum_{i=1}^{m} u_i \\ \sum_{i=1}^{m} u_i^2 + \sum_{j=m+1}^{n} v_j^2 - n\end{bmatrix}^2$$

```r
set.seed(121)  # MC simulation
M1<-M2<-M3<-matrix(NA,S,1)
M <- matrix(NA,S,3)
for (s in 1:S){
z_sim<-rbinom(20,1,r)
u_sim <- (w[which(z_sim==1)]-mu[2])/sigma
v_sim <- (w[which(z_sim==0)]-mu[1])/sigma
M1[s,] <- sum(v_sim)
M2[s,] <- sum(u_sim)
M3[s,] <- sum(u_sim^2)+sum(v_sim^2)
M[s,] <- c(M1[s],M2[s],M3[s]-n)
}
(Miss <- var(M)/sigma^2) # Missing Information
##              [,1]         [,2]         [,3]
## [1,]  0.015948173 -0.014971965 -0.014133296
## [2,] -0.014971965  0.017139205  0.014677705
## [3,] -0.014133296  0.014677705  0.013169230
```

```r
(I <- Compy-Miss) # Apply the Louis' method
##             [,1]        [,2]         [,3]
## [1,] 0.020540316  0.014971965  0.014138147
## [2,] 0.014971965  0.027159266 -0.014671234
## [3,] 0.014138147 -0.014671234  0.148405798
interval <- pnorm(0.975)*sqrt(diag(solve(I))) # Calculate Confidence Interval
CI <- matrix(c(theta-interval,theta+interval),3,2,dimnames =list(c("mu1","mu2","sigma"),c("CI-L","CI-U")
```

Table 1: Louis' Method: Var-Cov Matrix and Confidence intervals

| | mu1 | mu2 | Sigma | CI-L | CI-U |
|---|---|---|---|---|---|
| **mu1** | 114 | -72.6 | -18.04 | 108 | 125.9 |
| **mu2** | -72.6 | 85.12 | 15.33 | 102.1 | 117.5 |
| **Sigma** | -18.04 | 15.33 | 9.973 | 13.1 | 18.37 |

We get the point-wise 95% confidence intervals for the parameters by Louis' method.

iii) Compute the observed information matrix directly and obtain its inverse. Find out 95% confidence intervals for the parameters.

$$r_i = \frac{p_i\phi(u_i)}{p_i\phi(u_i) + (1-p_i)\phi(v_i)}$$

$$\frac{\partial}{\partial\theta} r_i = \frac{[p_i\phi(u_i) + (1-p_i)\phi(v_i)] p_i\phi(u_i)\frac{u_i}{\sigma}\begin{bmatrix}0\\1\\u_i\end{bmatrix} - p_i\phi(u_i)\left(p_i\phi(u_i)\frac{u_i}{\sigma}\begin{bmatrix}0\\1\\u_i\end{bmatrix} + (1-p_i)\phi(v_i)\frac{v_i}{\sigma}\begin{bmatrix}1\\0\\v_i\end{bmatrix}\right)}{[p_i\phi(u_i) + (1-p_i)\phi(v_i)]^2} = r_i[1-r_i]\frac{1}{\sigma}\begin{bmatrix}-v_i\\u_i\\u_i^2 - v_i^2\end{bmatrix}$$

$$\log(Pr(\vec{\theta}|\vec{y})) = \sum_{i=1}^{n}\log\left(\frac{1-p_i}{\sigma}\phi(v_i) + \frac{p_i}{\sigma}\phi(u_i)\right) = C - n\log(\sigma) + \sum_{i=1}^{n}\log\left((1-p_i)\phi(v_i) + p_i\phi(u_i)\right)$$

$$\frac{\partial}{\partial\theta}\log(p(\vec{\theta}|y)) = \frac{1}{\sigma}\sum_{i=1}^{n}\begin{bmatrix}\frac{(1-p_i)\phi(v_i)}{p_i\phi(u_i)+(1-p_i)\phi(v_i)}v_i\\ \frac{p_i\phi(u_i)}{p_i\phi(u_i)+(1-p_i)\phi(v_i)}u_i\\ \frac{p_i\phi(u_i)u_i^2+(1-p_j)\phi(v_i)v_i^2}{p_i\phi(u_i)+(1-p_i)\phi(v_i)} - 1\end{bmatrix} = \frac{1}{\sigma}\sum_{i=1}^{n}\begin{bmatrix}(1-r_i)v_i\\ r_iu_i\\ r_iu_i^2 + (1-r_i)v_i^2 - 1\end{bmatrix}$$

$$\frac{\partial^2\log(p(\vec{\theta}|y))}{\partial\theta^2} = \frac{1}{\sigma^2}\sum_{i=1}^{n}\begin{bmatrix}(1-r_i)(r_iv_i^2-1) & -r_i(1-r_i)u_iv_i & -(1-r_i)v_i[r_i(u_i^2-v_i^2)+2]\\ -r_i(1-r_i)u_iv_i & r_i((1-r_i)u_i^2-1) & r_iu_i[(1-r_i)(u_i^2-v_i^2)-2]\\ -(1-r_i)v_i[r_i(u_i^2-v_i^2)+2] & r_iu_i[(1-r_i)(u_i^2-v_i^2)-2] & r_i(1-r_i)(u_i^2-v_i^2)^2 - 3(r_iu_i^2+(1-r_i)v_i^2)+1\end{bmatrix}$$

```
(I_o <- matrix(c(sum((1-r)*(r*v^2-1)), sum(-r*(1-r)*u*v)  , sum(-(1-r)*v*(r*(u^2-v^2)+2)) ,
            sum(-r*(1-r)*u*v),      sum(r*((1-r)*u^2-1)) ,  sum(r*u*((1-r)*(u^2-v^2)-2)) ,
sum(-(1-r)*v*(r*(u^2-v^2)+2)),sum(r*u*((1-r)*(u^2-v^2)-2)),sum(r*(1-r)*(u^2-v^2)^2-3*(r*u^2+(1-r)*v^2)+
),nrow = 3, ncol = 3)/(sigma^2)) # Observed exact data
##             [,1]         [,2]         [,3]
## [1,] -0.020577509 -0.014939711 -0.014121235
## [2,] -0.014939711 -0.027180522  0.014661607
## [3,] -0.014121235  0.014661607 -0.148411150
```

```
interval <- pnorm(0.975)*sqrt(diag(solve(-I_o))) # Calculate Confidence Interval
CI_o <- matrix(c(theta-interval,theta+interval),3,2,dimnames =list(c("mu1","mu2","sigma"),c("CI-L","CI-
```

Table 2: Direct Method: Var-Cov Matrix and Confidence intervals which gives the 95% confidence intervals for $\mu_1, \mu_2$, and $\sigma$. Two methods give same result.

| | mu1 | mu2 | Sigma | CI-L | CI-U |
|---|---|---|---|---|---|
| **mu1** | 112.8 | -71.61 | -17.81 | 108.1 | 125.8 |
| **mu2** | -71.61 | 84.32 | 15.14 | 102.1 | 117.4 |
| **Sigma** | -17.81 | 15.14 | 9.929 | 13.1 | 18.37 |