Multinomial Example

```
// Multinomial Example RCPP CODE
NumericVector multi_e(NumericVector x, NumericVector p){
  double n1;
  double n2;
  double n3;
  n1 = 38;
  n2 = 34;
  n3 = 125*(p[2]/(p[3]+p[2]));
  NumericVector n = {n1,n2,n3};
  return n;
}

NumericVector multi_m(NumericVector n){
  double theta = (34+n[2])/(72+n[2]);
  NumericVector p = {1.0/2.0-theta/2, theta/4, theta/4, 1.0/2.0};
  return p;
}

// [[Rcpp::export]]
NumericVector multi_out(NumericVector x, NumericVector p, int itr){

  for(int i = 0; i<itr; i++){
    NumericVector n = multi_e(x, p);
    p = multi_m(n);
  }

  return(p);

}
```

```
#Multinomial R Code
x = c(38,34,125)
n = rep(0,4)
theta = 1/2
p = c(1/2-theta/2, theta/4, theta/4, 1/2)
itr = 50

multi_out(x,p, itr)
```

결과값은

0.1865893 0.1567054 0.1567054 0.5000000

PEPPERED MOTHS EXAMPLE

```
// RCPP
NumericVector pep_e1(NumericVector x, NumericVector p){
```

```
  double n_cc = (x[0]*pow(p[0],2))/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ci = (2*x[0]*p[0]*p[1])/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ct = (2*x[0]*p[0]*p[2])/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ii = (x[1]*pow(p[1],2))/(pow(p[1],2)+2*p[1]*p[2]);
  double n_it = (2*x[1]*p[1]*p[2])/(pow(p[1],2)+2*p[1]*p[2]);
  NumericVector n = {n_cc,n_ci,n_ct,n_ii,n_it,x[2]};
  return n;
}

NumericVector pep_m1(NumericVector x, NumericVector n){
  double p_c = (2*n[0]+n[1]+n[2])/(2*sum(x));
  double p_i = (2*n[3]+n[4]+n[1])/(2*sum(x));
  double p_t = (2*n[5]+n[2]+n[4])/(2*sum(x));
  NumericVector p = {p_c,p_i,p_t};
  return p;
}

// [[Rcpp::export]]
NumericVector pep_out1(NumericVector x, NumericVector p, int itr){

  for(int i = 0; i<itr; i++){
    NumericVector n = pep_e1(x, p);
    p = pep_m1(x, n);
  }

  return(p);

}
```

```
# R CODE
x = c(85, 196, 341) #observed data (carbonaria, insularia, typica)
n = rep(0,6) #expected number of each phenotype (CC, CI, CT, II, IT, TT)
p = rep(1/3,3) #probabilities of allele(carbonaria, insularia, typica)
itr = 50

pep_out1(x, p, itr)
```

결과값

0.07083691 0.18873652 0.74042657

Problem 4.1

(a)

Exercise 4.1

Complete Data : ~ Multinomial dist $(p_{CC}, p_{CI}, p_{CT}, p_{II}, p_{IT}, p_{TT})$

$y = (n_{CC}, n_{CI}, n_{CT}, n_{II}, n_{IT}, n_{TT})$

Observed data

$$x = (\underset{p_C}{n_{CC}+n_{CI}+n_{CT}},\ \underset{p_I}{n_{II}+n_{IT}},\ \underset{p_T}{n_{TT}},\ \underset{p_I \& p_T}{n_{II}+n_{IT}+n_{TT}}) = (n_C, n_I, n_T, n_U)$$

$(p_C^2, 2p_Cp_I, 2p_Cp_T, p_I^2, 2p_Ip_T, p_T^2)$

Complete-data log likelihood

$$l(p) = n_{CC}\log p_C^2 + n_{CI}\log 2p_Cp_I + n_{CT}\log 2p_Cp_T + n_{II}\log p_I^2 + n_{IT}\log 2p_Ip_T + n_{TT}\log p_T^2 + \text{Constant}$$

E-Step. $$E\{N_{CC} \mid n_C, n_I, n_T, p^{(t)}\} = n_{CC}^{(t)} = n_C \times \frac{p_C^{(t)2}}{\underbrace{p_C^{(t)2} + 2p_C^{(t)}p_I^{(t)} + 2p_C^{(t)}p_T^{(t)}}_{\text{C와 관련된 것들}}}$$

$$E\{N_{CI} \mid n_C, n_I, n_T, p^{(t)}\} = n_{CI}^{(t)} = n_C \times \frac{2p_C^{(t)}p_I^{(t)}}{p_C^{(t)2} + 2p_C^{(t)}p_I^{(t)} + 2p_C^{(t)}p_T^{(t)}}$$

$$E\{N_{CT} \mid \quad '' \quad\} = n_{CT}^{(t)} = n_C \times \frac{2p_C^{(t)}p_T^{(t)}}{p_C^{(t)2} + 2p_C^{(t)}p_I^{(t)} + 2p_C^{(t)}p_T^{(t)}}$$

$$E\{N_{II} \mid \quad '' \quad\} = n_{II}^{(t)} = n_I \times \frac{p_I^{(t)2}}{p_I^{(t)2} + 2p_I^{(t)}p_T^{(t)}} + n_U \times \frac{p_I^{(t)2}}{p_I^{(t)2} + 2p_I^{(t)}p_T^{(t)} + p_T^{(t)2}}$$

$$E\{N_{IT} \mid \quad '' \quad\} = n_{IT}^{(t)} = n_I \times \frac{2p_I^{(t)}p_T^{(t)}}{p_I^{(t)2} + 2p_I^{(t)}p_T^{(t)}} + n_U \times \frac{2p_I^{(t)}p_T^{(t)}}{p_I^{(t)2} + 2p_I^{(t)}p_T^{(t)} + p_T^{(t)2}}$$

$$E\{N_{TT} \mid \quad '' \quad\} = n_{TT}^{(t)} = n_T + n_U \times \frac{p_T^{(t)2}}{p_I^{(t)2} + 2p_I^{(t)}p_T^{(t)} + p_T^{(t)2}}$$

$$Q(p|p^{(t)}) = n_{cc}^{(t)} \log p_c^2 + n_{cI}^{(t)} \log 2p_c p_I + n_{cT}^{(t)} \log 2p_c p_T + n_{II}^{(t)} \log p_I^2 + n_{IT}^{(t)} \log 2p_I p_T + n_{TT} \log p_T^2 + C.$$

M-Step.

$$\frac{dQ(p|p^{(t)})}{dp_c} = \frac{2n_{cc}^{(t)} + n_{cI}^{(t)} + n_{cT}^{(t)}}{p_c} + \frac{2n_T + n_{cT}^{(t)} + n_{IT}^{(t)}}{1-p_c-p_I} \overset{set}{=} 0$$

$$\frac{dQ(p|p^{(t)})}{dp_I} = \frac{2n_{II}^{(t)} + n_{cI}^{(t)} + n_{IT}^{(t)}}{p_I} + \frac{2n_T + n_{cT}^{(t)} + n_{IT}^{(t)}}{1-p_c-p_I} = 0$$

$$p_c^{(t+1)} = \frac{2n_{cc}^{(t)} + n_{cI}^{(t)} + n_{cT}^{(t)}}{2n} \qquad p_T = \frac{2n_{TT}^{(t)} + n_{cT}^{(t)} + n_{IT}^{(t)}}{2n}$$

$$p_I^{(t+1)} = \frac{2n_{II}^{(t)} + n_{cI}^{(t)} + n_{IT}^{(t)}}{2n}$$

(b)

```
// Exercise 4.1
// RCPP
// E-step
NumericVector pep_e2(NumericVector x, NumericVector p){
  double n_cc = (x[0]*pow(p[0],2))/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ci = (2*x[0]*p[0]*p[1])/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ct = (2*x[0]*p[0]*p[2])/(pow(p[0],2)+2*p[0]*p[1]+2*p[0]*p[2]);
  double n_ii = (x[1]*pow(p[1],2))/(pow(p[1],2)+2*p[1]*p[2]) +
((x[3])*pow(p[1],2))/(pow(p[1],2)+2*p[1]*p[2]+pow(p[2],2));
  double n_it = (2*x[1]*p[1]*p[2])/(pow(p[1],2)+2*p[1]*p[2]) +
(2*x[3]*p[1]*p[2])/(pow(p[1],2)+2*p[1]*p[2]+pow(p[2],2));
```

```
  double n_tt = x[2] +
((x[3])*pow(p[2],2))/(pow(p[1],2)+2*p[1]*p[2]+pow(p[2],2));
  NumericVector n = {n_cc,n_ci,n_ct,n_ii,n_it,n_tt};
  return n;
}

// M-step
NumericVector pep_m2(NumericVector x, NumericVector n){
  double p_c = (2*n[0]+n[1]+n[2])/(2*sum(x));
  double p_i = (2*n[3]+n[4]+n[1])/(2*sum(x));
  double p_t = (2*n[5]+n[2]+n[4])/(2*sum(x));
  NumericVector p = {p_c,p_i,p_t};
  return p;
}

// [[Rcpp::export]]
NumericVector pep_out2(NumericVector x, NumericVector p, int itr){

  for(int i = 0; i<itr; i++){
    NumericVector n = pep_e2(x, p);
    p = pep_m2(x, n);
  }

  return(p);

}
```

```
#R
#observed data (carbonaria, insularia, typica, in+ty)
x = c(85, 196, 341, 578)
#expected number of each phenotype (CC, CI, CT, II, IT, TT)
n = rep(0,6)
p = rep(1/3,3) #probabilities of allele(carbonaria, insularia, typica)
itr = 30

pep_out2(x, p, itr)
```

결과값

0.03606708 0.19579918 0.76813373

p_c = 0.03606708

p_i = 0.19579918

p_t = 0.76813373

Problem 4.2

(a)

Exercise 4.2

$N = 1500$

$\sum_{i=1}^{16} n_i = N$

Complete Data $Y = C(n_{z0}, n_{t0}, n_{t1}, n_{t2}, \cdots, n_{t16}, n_{p0}, n_{p1}, \cdots n_{p16})$

Observed Data $X = M(Y) = C(n_0, n_1, n_2, \cdots\cdots, n_{16})$

$\pi_i(\theta) = \alpha \cdot 1_{[i=0]} + \beta \mu^i \exp\{-\mu\} + (1-\alpha-\beta)\lambda^i \exp\{-\lambda\}$

$z_0(\theta) = \dfrac{\alpha}{\pi_0(\theta)}$

$t_i(\theta) = \dfrac{\beta \mu^i \exp\{-\mu\}}{\pi_i(\theta)}$

$p_i(\theta) = \dfrac{(1-\alpha-\beta)\lambda^i \exp\{-\lambda\}}{\pi_i(\theta)}$

$p = (z_0(\theta), t_0(\theta), t_1(\theta), \cdots\cdots, t_{16}(\theta), p_0(\theta), p_1(\theta), \cdots\cdots, p_{16}(\theta))$

Let $1-\alpha-\beta = r$

Complete log likelihood $= n_{z,0}\log(\alpha) + \sum_{i=0}^{16} n_{t,i} \log \beta \mu^i \exp\{-\mu\} + \sum_{i=0}^{16} n_{p,i} \log r \lambda^i \exp\{-\lambda\}$

$= n_{z,0}\log(\alpha) + \sum_{i=0}^{16} n_{t,i} \{\log(\beta) + i\log(\mu) - \mu\} + \sum_{i=0}^{16} n_{p,i} \{\log(r) + i\log(\lambda) - \lambda\}$

$E[n_{z,0} \mid n_0, n_1, \cdots, n_{16}, \theta^{(t)}] = n_0 \times \dfrac{\alpha^{(t)}}{\pi_0(\theta^{(t)})} = n_0 \times z_0(\theta^{(t)})$

$E[n_{t,i} \mid n_0, n_1, \cdots, n_{16}, \theta^{(t)}] = n_i \times \dfrac{\beta^{(t)}(\mu^{(t)})^i \exp(-\mu^{(t)})}{\pi_i(\theta^{(t)})} = n_i \times t_i(\theta^{(t)})$

$E[n_{p,i} \mid n_0, n_1, \cdots, n_{16}, \theta^{(t)}] = n_i \times \dfrac{r^{(t)}(\lambda^{(t)})^i \exp(-\lambda^{(t)})}{\pi_i(\theta^{(t)})} = n_i \times p_i(\theta^{(t)})$

$\therefore Q(\theta|\theta^{(t)}) = n_0 z_0(\theta^{(t)}) \log(\alpha) + \sum_{i=0}^{16} n_i t_i(\theta^{(t)}) \{\log(\beta) + i\log(\mu) - \mu\} + \sum_{i=0}^{16} n_i p_i(\theta^{(t)}) \{\log(r) + i\log(\lambda) - \lambda\}$

Lagrange term을 이용해보자. ($\alpha+\beta+r=1$ 이라는 조건 존재).

$Q_{lagr}(\theta|\theta^{(t)}) = Q(\theta|\theta^{(t)}) + \phi(1-\alpha-\beta-r)$

$\dfrac{\partial Q_{lagr}(\theta|\theta^{(t)})}{\partial \alpha} = n_0 z_0(\theta^{(t)}) \times \dfrac{1}{\alpha} - \phi$

$$\therefore \alpha^{(t+1)} = \frac{1}{\phi} n_0 z_0(\theta^{(t)})$$

$$\frac{\partial Q_{lagr}(\theta|\theta^{(t)})}{\partial \beta} = \sum_{i=0}^{16} n_i t_i(\theta^{(t)}) \frac{1}{\beta} - \phi$$

$$\therefore \beta^{(t+1)} = \frac{1}{\phi} \sum_{i=0}^{16} n_i t_i(\theta^{(t)})$$

$$\frac{\partial Q_{lagr}(\theta|\theta^{(t)})}{\partial r} = \sum_{i=0}^{16} n_i p_i(\theta^{(t)}) \frac{1}{r} - \phi$$

$$\therefore r^{(t+1)} = \frac{1}{\phi} \sum_{i=0}^{16} n_i p_i(\theta^{(t)})$$

$$\alpha^{(t+1)} + \beta^{(t+1)} + r^{(t+1)} = 1 = \frac{1}{\phi} \left\{ n_0 z_0(\theta^{(t)}) + \sum_{i=0}^{16} n_i t_i(\theta^{(t)}) + \sum_{i=0}^{16} n_i p_i(\theta^{(t)}) \right\} = \frac{N}{\phi}$$

$$\phi = N$$

$$\therefore \alpha^{(t+1)} = \frac{1}{N} n_0 z_0(\theta^{(t)})$$

$$\beta^{(t+1)} = \frac{1}{N} \sum_{i=0}^{16} n_i t_i(\theta^{(t)}) = \sum_{i=0}^{16} \frac{n_i t_i(\theta^{(t)})}{N}$$

$$\frac{\partial Q_{lagr}(\theta|\theta^{(t)})}{\partial \mu} = \sum_{i=0}^{16} n_i t_i(\theta^{(t)}) \left(\frac{i}{\mu} - 1\right)$$

$$\therefore \mu^{(t+1)} = \frac{\sum_{i=0}^{16} i n_i t_i(\theta^{(t)})}{\sum_{i=0}^{16} n_i t_i(\theta^{(t)})}$$

$$\frac{\partial Q_{lagr}(\theta|\theta^{(t)})}{\partial \lambda} = \sum_{i=0}^{16} n_i p_i(\theta^{(t)}) \left(\frac{i}{\lambda} - 1\right)$$

$$\therefore \lambda^{(t+1)} = \frac{\sum_{i=0}^{16} i n_i p_i(\theta^{(t)})}{\sum_{i=0}^{16} n_i p_i(\theta^{(t)})}$$

(b)

```
// RCPP
// [[Rcpp::export]]
vec hiv_out(vec param, vec obs, int itr){
  vec new_param = zeros(4);
  double len = obs.size();
  double N = sum(obs);
  double z_0, t_sum, p_sum, z_sum;
  vec t(len);
  vec p(len);
  vec pi_f(len);
```

```
  for(int i = 0; i<itr; i++){
    //E-step
    //statistics update
    //0 group setting
    pi_f[0] = param[0] + param[1]*exp(-param[2]) + (1-param[0]-param[1])*exp(-
param[3]);
    z_0 = param[0] / pi_f[0];
    t[0] = (param[1]*exp(-param[2])) / pi_f[0];
    p[0] = ( (1-param[0]-param[1])*exp(-param[3]) ) / pi_f[0];

    //1~17 group update
    for(int j = 1; j < len; j++){
      pi_f[j] = param[1]*pow(param[2], j)*exp(-param[2]) + (1-param[0]-
param[1])*pow(param[3], j)*exp(-param[3]);
      t[j] = (param[1]*pow(param[2], j)*exp(-param[2])) / pi_f[j];
      p[j] = ((1-param[0]-param[1])*pow(param[3], j)*exp(-param[3])) / pi_f[j];
    }

    t_sum = 0;
    p_sum = 0;
    z_sum = 0;

    z_sum = obs[0] * z_0;

    for(int k=0; k < len; k++){
      t_sum = t_sum + obs[k]*t[k];
      p_sum = p_sum + obs[k]*p[k];
    }

    // M-step
    new_param = {0, 0, 0, 0};
    new_param[0] = z_sum / N ;
    for(int l=0; l<len; l++){
      new_param[1] = new_param[1] + (obs[l]*t[l]) / N ;
      new_param[2] = new_param[2] + (l*obs[l]*t[l]) / t_sum ;
      new_param[3] = new_param[3] + (l*obs[l]*p[l]) / p_sum ;

    }

    param = new_param;

  }

  return(param);

}
```

```
#R
#data
obs =
c(379.0,299.0,222.0,145.0,109.0,95.0,73.0,59.0,45.0,30.0,24.0,12.0,4.0,2.0,0.0,1
.0,1.0)
itr = 100
alpha = 0.5
beta = 0.2
mu = 1.0
lambda = 5.0
param = c(alpha, beta, mu, lambda)

#RCPP
hiv_out(param, obs, itr)
```

결과값

alpha: 0.1221626
beta: 0.5625419
mu: 1.4674525
lambda: 5.9388614