CS 5135/6035 Learning Probabilistic Models

Exercise Questions for Lecture 11: EM MV Gaussian, EM Correctness

Gowtham Atluri

2/20/2020

Questions

1. **Scenario:** PetalLength and PetalWidth of 150 flowers that belong to one of the three different species (setosa, versicolor, and virginica) are available in the iris dataset (as part of the Julia's RDatasets package).

   Assumptions: Assume PetalLength and PetalWidth (denoted as a vector $\boldsymbol{x}$) are the only avaialble variables. Treat Species (denoted as z) as a latent variable. Assume that this data is a mixture of three bivariate Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)$, $\mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b)$, and $\mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$. For simplicity, species setosa is denoted as state a, species versicolor as b, virginica as c.

   Goal: Your goal is to estimate the parameters $\boldsymbol{\mu}_a$,$\boldsymbol{\Sigma}_a$,$\boldsymbol{\mu}_b$,$\boldsymbol{\Sigma}_b$,$\boldsymbol{\mu}_c$,$\boldsymbol{\Sigma}_c$,π_a,π_b, and π_c. The following questions will guide you in writing the EM algorithm (with E and M Steps), implementing it and demonstrating that it has estimated the correct parameters. **[12*2 = 24 points]**

   a. Write the expression for the marginal probability $p(z)$.
   b. Write the expression for the conditional probability $p(\boldsymbol{x}|z)$.
   c. Write the expression for the probability density for one data point $p(\boldsymbol{x})$.
   d. Write the expression for the likelihood.
   e. Write the expression for the log-likelihood.
   f. Write the expressions for posterior probabilities $p(z = a|\boldsymbol{x})$, $p(z = b|\boldsymbol{x})$, and $p(z = c|\boldsymbol{x})$.
   g. Write the update equations for estimating the three component means.
   h. Write the update equations for estimating the three component covariance matrices.
   i. Write the update equations for estimating the probabilities p_a, p_b, and p_c.
   j. Write the EM algorithm.
   k. Implement it in Julia.
   l. Plot the final estimated components on top of a scatter plot and comment on the goodness of the fit based on visual inspection.

2. Prove the following: **[1 point]**

   For any choice of $q(\boldsymbol{Z})$, the following decomposition holds

$$\log p(\boldsymbol{X}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q||p)$$

   where

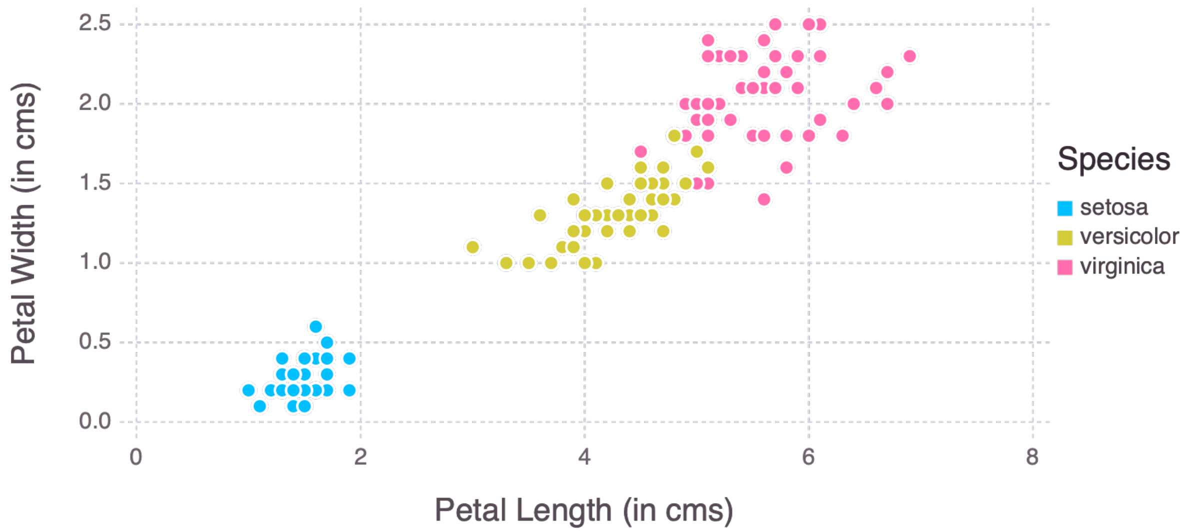
$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\boldsymbol{z}} q(\boldsymbol{Z}) \log \left\{ \frac{p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})}{q(\boldsymbol{Z})} \right\}$$

$$KL(q||p) = -\sum_{\boldsymbol{z}} q(\boldsymbol{Z}) \log \left\{ \frac{p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta})}{q(\boldsymbol{Z})} \right\}$$

Sample code

1. To plot the densities of the data

```
    using RDatasets, Gadfly, Distributions;
    data = dataset("datasets","iris");
    myplot = plot(data,x=:PetalLength,y=:PetalWidth, color=:Species,Geom.point,
                  Guide.xlabel("Petal Length (in cms)"),
                  Guide.ylabel("Petal Width (in cms)"), major_label_font_size=18pt,
                  minor_label_font_size=14pt,
                  key_title_font_size = 18pt,
                  key_label_font_size = 14pt,
                  major_label_color=colorant"black",
                  minor_label_color=colorant"black",
                  Coord.Cartesian(xmin=0, xmax=8));
    draw(PNG("./figs/iris_densities.png", 6inch, 3inch,dpi=300), myplot);
```



2. Sample code for plotting histogram and EM results

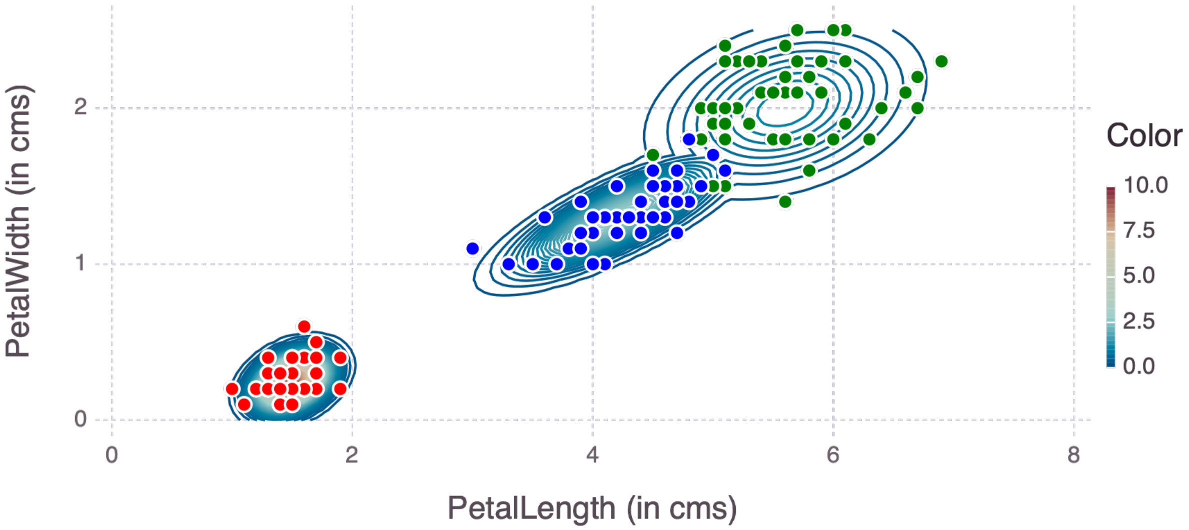
```
data = dataset("datasets","iris");
data_mat_a = data[find(data[:Species].=="setosa"),[:PetalLength,:PetalWidth]];
data_mat_b = data[find(data[:Species].=="versicolor"),[:PetalLength,:PetalWidth]];
data_mat_c = data[find(data[:Species].=="virginica"),[:PetalLength,:PetalWidth]];
nrows_a = size(data_mat_a,1);
nrows_b = size(data_mat_b,1);
nrows_c = size(data_mat_c,1);

#Estimate these using EM for MV Gaussian approach
mean_vec_a = vec([1.5 0.25]);
mean_vec_b = vec([4.2 1.3]);
mean_vec_c = vec([5.6 2.0]);
cov_mat_a = [0.031 0.0061; 0.0061 0.0109];
cov_mat_b = [0.22 0.0732; 0.0732 0.039];
cov_mat_c = [0.3008 0.0466; 0.0466 0.0746];

d_a = MvNormal(mean_vec_a,cov_mat_a);
d_b = MvNormal(mean_vec_b,cov_mat_b);
d_c = MvNormal(mean_vec_c,cov_mat_c);
```

```
a = collect(0:0.05:8);
b = collect(0:0.05:2.5);
pdf_mv = zeros(length(a),length(b));
for i=1:length(a)
    for j=1:length(b)
        pdf_mv[i,j] = maximum([pdf(d_a,[a[i],b[j]]),pdf(d_b,[a[i],b[j]]),pdf(d_c,[a[i],b[j]])]);
    end
end

myplot = plot(layer(x=data_mat_a[:,1],y=data_mat_a[:,2],
Geom.point,Theme(default_color=colorant"red")),layer(x=data_mat_b[:,1],y=data_mat_b[:,2],
Geom.point,Theme(default_color=colorant"blue")),layer(x=data_mat_c[:,1],y=data_mat_c[:,2],
Geom.point,Theme(default_color=colorant"green")),layer(z=pdf_mv,x=a,y=b, Geom.contour(levels=80)),
Coord.Cartesian(xmin=0, xmax=8,ymin=0,ymax=2.55),
major_label_font_size=18pt,
minor_label_font_size=14pt,
key_title_font_size = 18pt,
key_label_font_size = 14pt,
major_label_color=colorant"black",
minor_label_color=colorant"black",Guide.xlabel("PetalLength (in cms)"),Guide.ylabel("PetalWidth (in cms
draw(PNG("./figs/fitting_multivar_iris.png", 6inch, 3inch,dpi=300), myplot);
```



3. EM code

```
## Coding EM

function E_step(x,mu_M,mu_F,sigma_M,sigma_F,pi_M)
    numerator = zeros(size(x,1));
    denominator = zeros(size(x,1));
    post_x = zeros(size(x,1));
    for i=1:size(x,1)
        numerator[i] = pi_M.*pdf(MvNormal(mu_M,sigma_M),x[i,:]);
        denominator[i] = numerator[i] + (1-pi_M)* pdf(MvNormal(mu_F,sigma_F),x[i,:]);
        post_x[i] = numerator[i] ./denominator[i];
    end

    return post_x;
end

function M_step(x,post_x)
    mu_M = sum(post_x.*x,1)./sum(post_x);
    mu_M = Vector(mu_M[:]);
    mu_F = sum((1.-post_x).*x,1)./sum((1.-post_x));
    mu_F = Vector(mu_F[:]);
    sigma_M = round.((post_x.*(x.-mu_M'))'*(x.-mu_M')
              /sum(post_x),5);
    sigma_F = round.(((1.-post_x).*(x.-mu_F'))'*(x.-mu_F')
              /sum(1.-post_x),5);
    pi_M = sum(post_x)/size(x,1);
    return mu_M, mu_F, sigma_M, sigma_F, pi_M;
end

function EM(x,mu_M,mu_F,sigma_M, sigma_F,pi_M)
    maxIter = 1000;
    for i=1:maxIter
        print(i,"\n");
        post_x = E_step(x,mu_M,mu_F,sigma_M,sigma_F,pi_M); print(post_x,"\n");
        mu_M_new, mu_F_new,sigma_M_new, sigma_F_new, pi_M_new = M_step(x,post_x);
            print(mu_M_new," ",mu_F_new,"\n");
            print(sigma_M_new," ",sigma_F_new,"\n");
        if(sum(abs.(mu_M-mu_M_new))<0.001 && sum(abs.(mu_F-mu_F_new))<0.001
                && sum(abs.(sigma_M-sigma_M_new))<0.001 && sum(abs.(sigma_F-sigma_F_new))<0.001)
            break;
        end;
        mu_M = mu_M_new; mu_F = mu_F_new;
        sigma_M = sigma_M_new; sigma_F = sigma_F_new;
        pi_M = pi_M_new;
    end
    return mu_M, mu_F, sigma_M, sigma_F, pi_M;
end

data = dataset("car","Davis");
data = data[[1:11; 13:end],:]; #droppping an outlier
x = convert(Array,data[:,[:Height,:Weight]]);
mu_M=[180, 78];
mu_F=[160, 50];
sigma_M = [10.0 0.0; 0.0 10.0];
sigma_F = [10.0 0.0; 0.0 10.0];
pi_M = 0.5;

mu_M, mu_F, sigma_M, sigma_F, pi_M = EM(x,mu_M,mu_F,sigma_M,sigma_F,pi_M);
```