# Assignment 10)

1).

a) $C \in \{S, V_s, V_g\}$.

$$P(z_i) = \prod_c \pi_c^{I(z_i = c)}$$

b) $p(x_i|z_i) = \prod_c \mathcal{N}(x_i; \mu_c, 0.54)^{I(z_i=c)}$

c)

$p(x_i) = \sum_{z_i} p(x_i, z_i)$

$= \sum_{z_i} p(x_i|z_i)\, p(z_i)$

$= \sum_{z_i} \left( \mathcal{N}(x_i; \mu_c, 0.54)^{I(z_i=c)} * \pi_c^{I(z_i=c)} \right)$

$= \sum_{z_i} \left( \pi_c * \mathcal{N}(x_i; \mu_c, 0.54) \right)^{I(z_i=c)}$

'z' can take '3' categories.

• therefore opening up.

$= \pi_s * \mathcal{N}(x_i; \mu_s, 0.54) +$

$\pi_{vs} * \mathcal{N}(x_i; \mu_{vs}, 0.54)$

$+ \pi_{vg} * \mathcal{N}(x_i; \mu_{vg}, 0.54)$.

$L = \prod_{i=1}^{n} \Big( \pi_s * \mathcal{N}(x_i; \mu_s, 0.54) +$

$\pi_{vs} * \mathcal{N}(x_i; \mu_{vs}, 0.54) +$

$\pi_{vg} * \mathcal{N}(x_i; \mu_{vg}, 0.54) \Big)$

e).

$l = \sum_{i=1}^{n} \log \Big( \pi_s * \mathcal{N}(x_i; \mu_s, 0.54)$

$+ \pi_{vs} * \mathcal{N}(x_i; \mu_{vs}, 0.54)$

$+ \pi_{vg} * \mathcal{N}(x_i; \mu_{vg}, 0.54) \Big)$

(f)

w.r.t.

$p(z_i|x_i) = \dfrac{p(x_i|z_i)\, p(z_i)}{p(x_i)}$

$= \dfrac{\pi_c \;\; \pi_c * \mathcal{N}(x_i; \mu_c, 0.54)}{\pi_s * \mathcal{N}(x_i; \mu_s, 0.54) + \pi_{vs} * \mathcal{N}(x_i; \mu_{vs}, 0.54) + \pi_{vg} * \mathcal{N}(x_i; \mu_{vg}, 0.54)}$

from above

$P(z_i = \text{setosa} \mid x_i)$

$$= \frac{\pi_s \mathcal{N}(x_i; \mu_s, 0.54)}{\left(\pi_s \mathcal{N}(x_i; \mu_s, 0.54) + \pi_{vs} \mathcal{N}(x_i; \mu_{vs}, 0.54) + \pi_{vg} \mathcal{N}(x_i; \mu_{vg}, 0.54)\right)}$$

$\|^{ly}$

$P(z_i = \text{Versicolor} \mid x_i)$

$$= \frac{\pi_{vs} \mathcal{N}(x_i; \mu_{vs}, 0.54)}{\left(\pi_s \mathcal{N}(x_i; \mu_s, 0.54) + \pi_{vs} \mathcal{N}(x_i; \mu_{vs}, 0.54) + \pi_{vg} (x_i; \mu_{vg}, 0.54)\right)}$$

$P(z_i = \text{Virginica} \mid x_i)$

$$= \frac{\pi_{vg} \mathcal{N}(x_i; \mu_{vg}, 0.54)}{\left(\pi_s \mathcal{N}(x_i; \mu_s, 0.54) + \pi_{vs} \mathcal{N}(x_i; \mu_{vs}, 0.54) + \pi_{vg} (x_i; \mu_{vg}, 0.54)\right)}$$

9) first generally ~~doing~~ writing a differentiation equation for a univariate gaussian ~~equation against~~

$$\frac{\partial \mathcal{N}(x; \mu, \sigma^2)}{\partial \mu} = \frac{\partial}{\partial \mu}\left(\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right)$$

$$= \underbrace{\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}}_{\mathcal{N}(x; \mu, \sigma^2)} \cdot \frac{2(x-\mu)}{2\sigma^2}$$

$$= \mathcal{N}(x; \mu, \sigma^2) \frac{(x-\mu)}{\sigma^2} \quad \to (1)$$

Using the above equation we can write the update equation as follows

$$l = \sum_{i=1}^{n} \log\left[\pi_s \mathcal{N}(x_i; \mu_s, 0.54) + \pi_{vs} \mathcal{N}(x_i; \mu_{vs}, 0.54) + \pi_{vg} \mathcal{N}(x_i; \mu_{vg}, 0.54)\right]$$

$$\frac{\partial l}{\partial \mu_s} = \sum_{i=1}^{n} \left\{ \frac{\pi_s \mathcal{N}(x_i ; \mu_s, 0.54)}{\left(\pi_s \mathcal{N}(x_i ; \mu_s, 0.54) + \pi_{v_s} \mathcal{N}(x_i ; \mu_{v_s}, 0.54) + \pi_{v_g} \mathcal{N}(x_i ; \mu_{v_g}, 0.54)\right)} \right\} \left( \frac{x_i - \mu_s}{0.54} \right)$$

$P(z = \text{setosa} \mid x_i)$.

$$\frac{\partial l}{\partial \mu_s} = \sum_{i=1}^{n} P(z=s|x_i) * \frac{x_i - \mu_s}{0.54} = 0$$

Solving for $\mu_s$ we get

$$\mu_s = \frac{\sum_{i=1}^{n} P(s|x_i) \cdot x_i}{\sum_{i=1}^{n} P(s|x_i)}$$

$|||^{rly}$

$$\mu_{vs} = \frac{\sum_{i=1}^{n} P(V_s|x_i) x_i}{\sum_{i=1}^{n} P(V_s|x_i)}$$

$$\mu_{vg} = \frac{\sum_{i=1}^{n} P(V_g|x_i) x_i}{\sum_{i=1}^{n} P(V_g|x_i)}$$

h).

Step 1 :- pick initial values for $\mu_s$, $\mu_{vs}$, $\mu_{vg}$

Step 2 :- Max iter = 1000

Step 3 : for i=1 : max iter

Step 4 :-

Compute

$$P(z=s|x_i) = \frac{\pi_s * N(x_i; \mu_s, 0.54)}{\left(\pi_s * N(x_i; \mu_s, 0.54) + \pi_{vs} * N(x_i; \mu_{vs}, 0.54) + \pi_{vg} * N(x_i; \mu_{vg}, 0.54)\right)}$$

$$P(z=v_s|x_i) = \frac{\pi_{vs} * N(x_i; \mu_{vs}, 0.54)}{\left(\pi_s * N(x_i; \mu_s, 0.54) + \pi_{vs} * N(x_i; \mu_{vs}, 0.54) + \pi_{vg} * N(x_i; \mu_{vg}, 0.54)\right)}$$

$$P(z=v_g|x_i) = \frac{\pi_{vg} * N(x_i; \mu_{vg}, 0.54)}{\left(\pi_s * N(x_i; \mu_s, 0.54) + \pi_{vs} * N(x_i; \mu_{vs}, 0.54) + \pi_{vg} * N(x_i; \mu_{vg}, 0.54)\right)}$$

**Step 5** :- optimize $\mu_s$, $\mu_{vs}$ & $\mu_{vg}$.

$$\mu_s = \frac{\sum_{i=1}^{n} p(z=s|x_i)\, x_i}{\sum_{i=1}^{n} p(z=s|x_i)}$$

$$\mu_{vs} = \frac{\sum_{i=1}^{n} p(z=v_s|x_i)\, x_i}{\sum_{i=1}^{n} p(z=v_s|x_i)}$$

$$\mu_{vg} = \frac{\sum_{i=1}^{n} P(z=v_g|x_i)\, x_i}{\sum_{i=1}^{n} P(z=v_g|x_i)}$$

**Step 6**

If $|\mu_s^i - \mu_s^{i-1}| < \varepsilon$.

& $|\mu_{vs}^i - \mu_{vs}^{i-1}| < \varepsilon$.

& $|\mu_{vg}^i - \mu_{vg}^{i-1}| < \varepsilon$

terminate ; end.

**Step 7** end for.

I)

In [1]:
```
function E_step(x,mu_S,mu_Vs, mu_Vg,sigma,p_S, p_Vs, p_Vg)
    post_S = p_S*pdf.(Normal(mu_S,sigma),x)
    post_Vs = p_Vs*pdf.(Normal(mu_Vs,sigma),x)
    post_Vg = p_Vg*pdf.(Normal(mu_Vg,sigma),x)
    denom = post_S + post_Vs + post_Vg
    return post_S./denom, post_Vs./denom, post_Vg./denom;
end
```

Out[1]: `E_step (generic function with 1 method)`

In [2]:
```
function M_step(x,post_S, post_Vs, post_Vg)
    mu_S = (post_S'*x)./sum(post_S);
    mu_Vs = (post_Vs'*x)./sum(post_Vs);
    mu_Vg = (post_Vg'*x)./sum(post_Vg);
    return mu_S, mu_Vs, mu_Vg;
end
```

Out[2]: `M_step (generic function with 1 method)`

In [3]:
```
function EM(x,mu_S,mu_Vs, mu_Vg,sigma,p_S,p_Vs,p_Vg)
    maxIter = 1000;
    for i=1:maxIter
        post_S, post_Vs, post_Vg = E_step(x,mu_S,mu_Vs, mu_Vg,sigma,p_S,p_Vs,p
_Vg);
        mu_S_new,mu_Vs_new, mu_Vg_new = M_step(x,post_S, post_Vs, post_Vg);
        if(abs(mu_S-mu_S_new)<0.001 && abs(mu_Vs-mu_Vs_new)<0.001 && abs(mu_Vg
-mu_Vg_new)<0.001)
            break;
        end;
        mu_S = mu_S_new;
        mu_Vs = mu_Vs_new;
        mu_Vg = mu_Vg_new;
    end
    return mu_S,mu_Vs, mu_Vg;
end
```

Out[3]: `EM (generic function with 1 method)`

In [4]:
```
using RDatasets;
using Distributions;
data = dataset("datasets", "iris");
x = data[:PetalLength];
mu_S=2;
mu_Vs=3;
mu_Vg = 4;
p_S = 0.34;
p_Vs = 0.33;
p_Vg = 0.33;
sigma=0.54;
EM(x,mu_S,mu_Vs, mu_Vg,sigma,p_S,p_Vs,p_Vg);
```

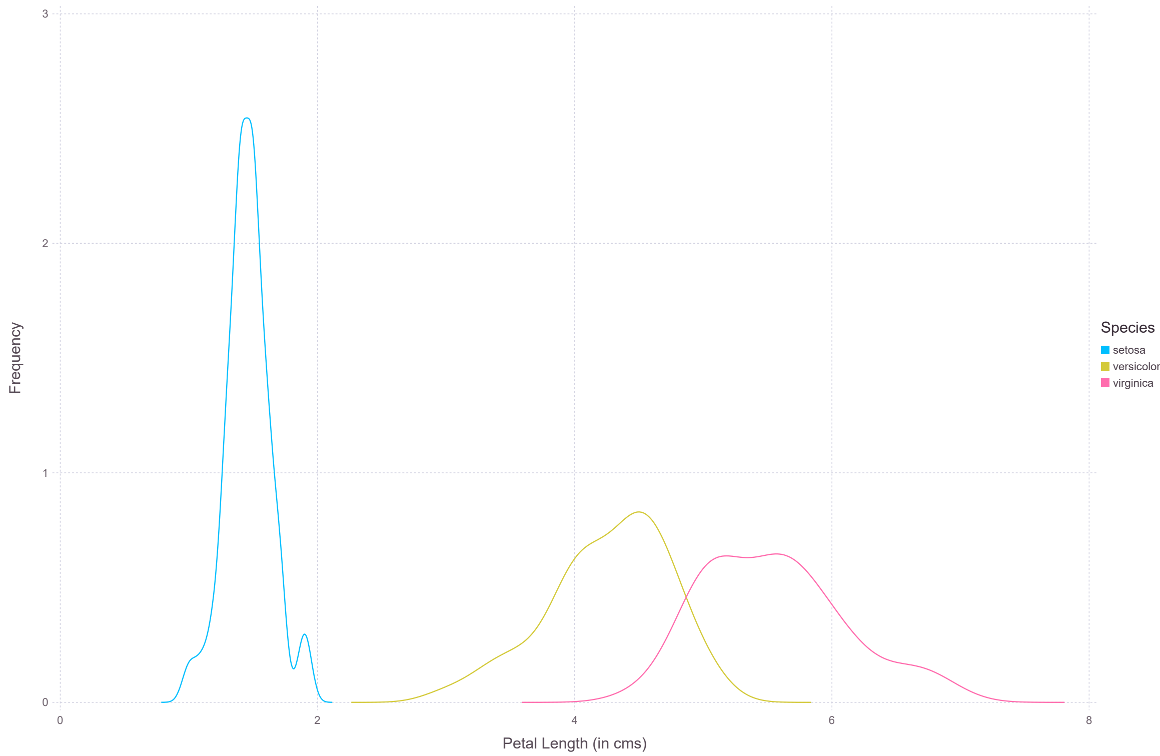
```
WARNING: Method definition unix2zdt(Real) in module TimeZones at /users/PES08
01/nifaullah/.julia/v0.6/TimeZones/src/conversions.jl:122 overwritten in modu
le RData at /users/PES0801/nifaullah/.julia/v0.6/RData/src/convert.jl:201.
```

In [15]:
```
print(mean(data[:PetalLength][1:50]))
print("\n")
print(mean(data[:PetalLength][51:100]))
print("\n")
print(mean(data[:PetalLength][101:150]))
print("\n")
print(var(data[:PetalLength][1:50]))
print("\n")
print(var(data[:PetalLength][51:100]))
print("\n")
print(var(data[:PetalLength][101:150]))
```

```
1.462
4.260000000000001
5.5520000000000005
0.030159183673469387
0.22081632653061226
0.3045877551020408
```

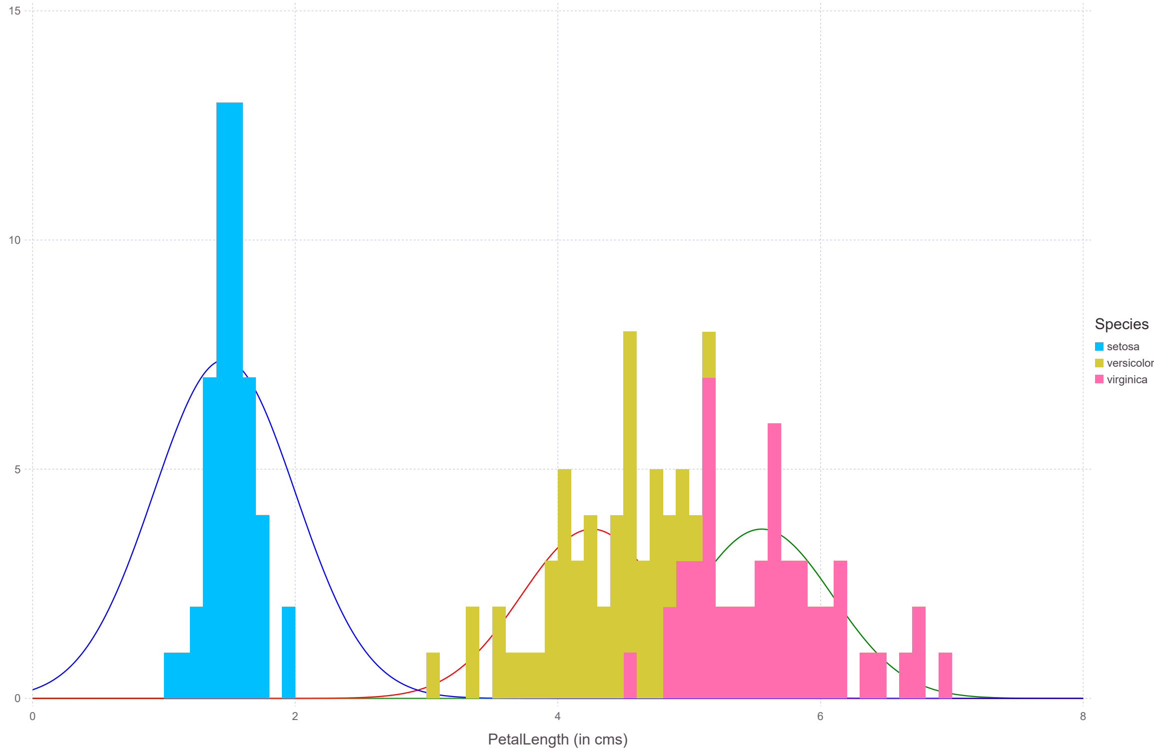
In [13]:
```
using Gadfly;
set_default_plot_size(12inch, 8inch)
Gadfly.plot(data,x=:PetalLength,color=:Species,Geom.density,
    Guide.xlabel("Petal Length (in cms)"),
    Guide.ylabel("Frequency"),
    Coord.Cartesian(xmin=0, xmax=8))
```

Out[13]:



In [12]:
```
myplot = plot(layer(data,x=:PetalLength,color=:Species,Geom.histogram,
    Theme(default_color=colorant"purple")),
    layer(x=0:0.02:8,y=pdf.(Normal(1.462,0.54),0:0.02:8)*10,Geom.line,
    Theme(default_color=colorant"blue")),
    layer(x=0:0.02:8,y=pdf.(Normal(4.26,0.54),0:0.02:8)*5,Geom.line,
    Theme(default_color=colorant"red")),
    layer(x=0:0.02:8,y=pdf.(Normal(5.552,0.54),0:0.02:8)*5,Geom.line,
    Theme(default_color=colorant"green")),
    Guide.xlabel("PetalLength (in cms)"),Guide.ylabel(""),Coord.Cartesian(xmin
=0, xmax=8))
```

Out[12]:




J) For Setosa mean estimate seems like a good fit but the overall Gaussian seems to have higher variance than the actual data, therefore this would be considered an approximate fit but not a good or a exact fit.
For Versicolor mean estimate doesn't seem to appropriately represent the mode of the data (which is the case for any Gaussian distribution), in that sense this looks like an a approximate fit than exact fit for the versicolor data.
For Virginica also mean estimate doesn't seem to appropriately represent the mode of the data hence again this is a somewhat approximate fit for data but not an exact fit