EM-Algorithm

January 9, 2020

1 Assignment 1

1.1 M-step, maximization w.r.t. u_i

$$L_m(u, \theta, \alpha) = \frac{1}{m} \sum_{\ell=1}^{m} \sum_{i \in D} \sum_{s_i \in \{0,1\}} \left[\alpha_\ell(s_i) \log p_{u_i, \theta^\ell}(x_i^\ell, s_i) - \alpha_\ell(s_i) \log \alpha_\ell(s_i) \right] \tag{1}$$

There is sum over i and we know that we are facing pixelwise independent shape model, so we can look for this equation for each pixel independently.

$$L_m(u, \theta, \alpha) = \frac{1}{m} \sum_{\ell=1}^{m} \sum_{i \in D} \sum_{s_i \in \{0,1\}} \left[\alpha_\ell(s_i) \log p_{u_i, \theta^\ell}(x_i^\ell, s_i) - \underbrace{\cancel{\alpha_\ell(s_i) \log \alpha_\ell(s_i)}}_{\text{it doesn't depend on } u_i} \right] \tag{2}$$

$$L_m(u, \theta, \alpha) = \frac{1}{m} \sum_{\ell=1}^{m} \sum_{i \in D} \sum_{s_i \in \{0,1\}} \alpha_\ell(s_i) \log p_{u_i, \theta^\ell}(x_i^\ell, s_i) \tag{3}$$

Using Bayes' theorem we can rewrite p_{u_i, θ^ℓ} to the form

$$\begin{aligned} p_{u_i, \theta^\ell}(x_i^\ell, s_i) &= p_{\theta^\ell}(x_i^\ell | s_i) p_{u_i}(s_i) \\ &= p_{\theta^\ell}(x_i^\ell | s_i) \cdot \frac{e^{u_i s_i}}{1 + e^{u_i}} \end{aligned} \tag{4}$$

Which leads us to

$$\begin{aligned} \log p_{u_i, \theta^\ell}(x_i^\ell, s_i) &= \log \left[p_{\theta^\ell}(x_i^\ell | s_i) \cdot \frac{e^{u_i s_i}}{1 + e^{u_i}} \right] \\ &= \log p_{\theta^\ell}(x_i^\ell | s_i) + u_i s_i - \log(1 + e^{u_i}) \end{aligned} \tag{5}$$

We want maximize w.r.t u_i, so we care only about parts which depends on each. (We just care about all examples of this pixel in training set)
There is sum over s_i so rewrite it.

$$\frac{1}{m}\sum_{\ell=1}^{m}[\alpha_\ell(s_i=0)\log p_{u_i,\theta^\ell}(x_i^\ell, s_i=0) + \alpha_\ell(s_i=1)\log p_{u_i,\theta^\ell}(x_i^\ell, s_i=1)] \tag{6}$$

We can now put together (6) and use it with results from (5)

$$\frac{1}{m}\sum_{\ell=1}^{m}[\alpha_\ell(s_i=0)(\underbrace{\cancel{\log p_{\theta^\ell}(x_i^\ell|s_i=0)}}_{\text{it doesn't depend on } u_i} + \cancel{u_i\cdot 0} - \log(1+e^{u_i})) \tag{7}$$

$$+\alpha_\ell(s_i=1)(\underbrace{\cancel{\log p_{\theta^\ell}(x_i^\ell|s_i=1)}}_{\text{it doesn't depend on } u_i} + u_i\cdot 1 - \log(1+e^{u_i}))]$$

$$= \frac{1}{m}\sum_{\ell=1}^{m}[\alpha_\ell(s_i=1)u_i - \underbrace{\cancel{(\alpha_\ell(s_i=1)+\alpha_\ell(s_i=0))}}_{\text{equals } 1}\log(1+e^{u_i})] \tag{8}$$

Cleaning up

$$\frac{1}{m}\sum_{\ell=1}^{m}[\alpha_\ell(s_i=1)u_i - \log(1+e^{u_i}) \tag{9}$$

$$= \frac{1}{m}\sum_{\ell=1}^{m}[\alpha_\ell(s_i=1)u_i] - \frac{1}{m}\sum_{\ell=1}^{m}[\log(1+e^{u_i})]$$

$$= \frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)u_i - \log(1+e^{u_i})\underbrace{\cancel{\frac{1}{m}\sum_{\ell=1}^{m}1}}_{\text{equals } 1}$$

So we ended up with maximization task

$$\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)u_i - \log(1+e^{u_i}) \to \max_{u_i} \tag{10}$$

1.1.1 Show it is concave and has a unique global maximum

Compute derivative w.r.t u_i

$$\frac{\partial}{\partial u_i}\left(\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)u_i - \log(1+e^{u_i})\right) = \frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1) - \frac{1}{1+e^{u_i}}e^{u_i} = 0 \tag{11}$$

No we divide right side by e^{-u_i}

$$\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)=\frac{e^{u_i}}{1+e^{u_i}}=\frac{1}{e^{-u_i}+1} \tag{12}$$

$$e^{-u_i}+1=\frac{1}{\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}$$

$$e^{-u_i}=\frac{1}{\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}-1$$

$$u_i=-\log\left(\frac{1-\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}{\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}\right)$$

$$u_i=\log\left(\frac{\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}{1-\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)}\right)$$

Now compute second derivative w.r.t u_i

$$\frac{\partial^2}{\partial u_i^2}\left(\frac{1}{m}\sum_{\ell=1}^{m}\alpha_\ell(s_i=1)-\frac{1}{1+e^{u_i}}e^{u_i}\right)=\frac{\partial^2}{\partial u_i^2}\left(-\frac{e^{u_i}}{1+e^{u_i}}\right)=\frac{\partial^2}{\partial u_i^2}\left(-\frac{1}{e^{-u_i}+1}\right) \tag{13}$$

$$\frac{\partial^2}{\partial u_i^2}\left(-\frac{1}{e^{-u_i}+1}\right)=\underbrace{-\frac{e^{-u_i}}{\underbrace{(e^{-u_i}+1)^2}_{\text{grater than 0}}}}_{\text{smaller than 0}}<0 \tag{14}$$

First derivative equals zero only for one particular situation (equation (12)) and second is smaller than zero (equation (14)) so function is concave and has a unique global maximum.

1.2 M-step, maximization w.r.t. θ_ℓ

We can begin with equation (2) with the results from (5)

$$L_m(u,\theta,\alpha) = \frac{1}{m}\sum_{\ell=1}^{m}\sum_{i\in D}\sum_{s_i\in\{0,1\}} \alpha_\ell(s_i)[\log p_{\theta^\ell}(x_i^\ell|s_i) + \underbrace{\cancel{u_i s_i - \log(1+e^{u_i})}}_{\text{don't depend on }\theta}] \tag{15}$$

$$= \frac{1}{m}\sum_{\ell=1}^{m}\sum_{i\in D}\sum_{s_i\in\{0,1\}} \alpha_\ell(s_i)\log p_{\theta^\ell}(x_i^\ell|s_i) \tag{16}$$

Now we want to maximize w.r.t. θ_ℓ and we know that there is one θ for each training set, so we are looking only at one image at each time. Then split sum over s in to parts.

$$\sum_{i\in D}\sum_{s_i\in\{0,1\}} \alpha_\ell(s_i)\log p_{\theta^\ell}(x_i^\ell|s_i)$$
$$= \sum_{i\in D}[\alpha_\ell(s_i=1)\log p_{\theta^\ell}(x_i^\ell|s_i=1) + \alpha_\ell(s_i=0)\log p_{\theta^\ell}(x_i^\ell|s_i=0)] \tag{17}$$

Then knowing that foreground and background pixels have different parameter θ_0 and θ_1:

$$\sum_{i\in D}[\alpha_\ell(s_i=1)\log p_{\theta_1^\ell}(x_i^\ell|s_i=1) + \alpha_\ell(s_i=0)\log p_{\theta_0^\ell}(x_i^\ell|s_i=0)] \tag{18}$$

And this splits into two independent maximization tasks for each θ

$$\sum_{i\in D}\alpha_\ell(s_i=1)\log p_{\theta_1^\ell}(x_i^\ell|s_i=1) \to \max_{\theta_1^\ell} \tag{19}$$
$$\sum_{i\in D}\alpha_\ell(s_i=0)\log p_{\theta_0^\ell}(x_i^\ell|s_i=0) \to \max_{\theta_0^\ell} \tag{20}$$

2 Assignment 2

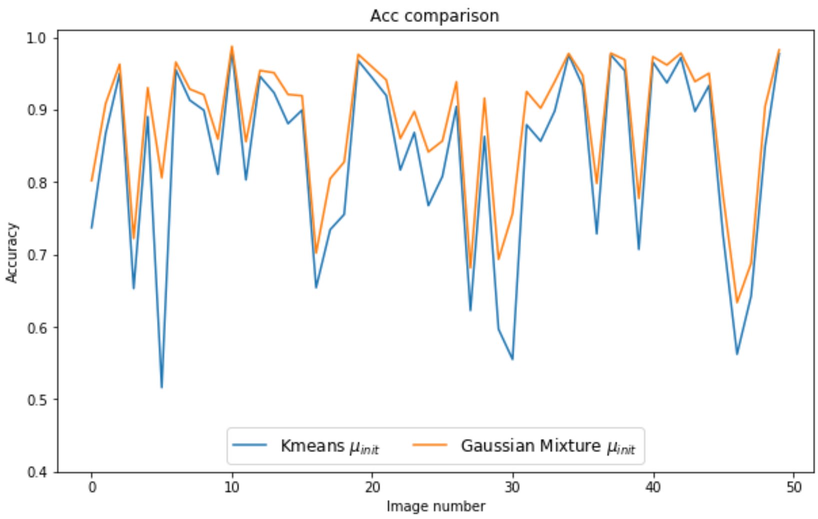

Figure 1: Accuracy comparison for Kmeans and Gaussian Mixture with initialized means

For each image Gaussian Mixture obtained better results (50:0). When there were no initialization then Kmeans ended up once with better result (49:1). It is expected result due to fact that Kmeans looks for hard boundaries between classes when mixture of Gaussians allows to exists areas where both classes could coexists. (Kmeans calculate euclidean distance when GM weighted distance).

Figure 2: Areas to initialize background and foreground means

To initialize means I choose circle in the middle of image and then compute means for each color, from "hand" image, in this area. I also used the similar approach for background pixels, using this tie only pixels from frame near borders.

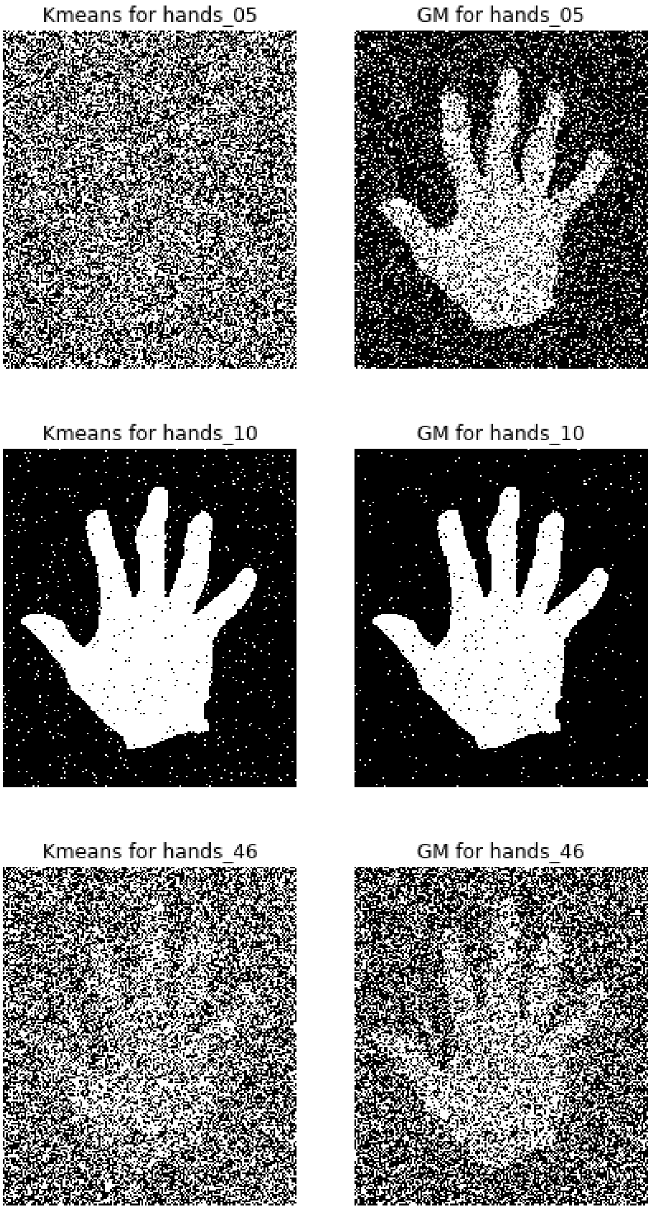

Figure 3: Few examples of segmentation for Kmeans and Gausssian Mixture

3 Assignment 3

3.1 E-step

To compute alphas I use $p_{u_i}(s_i = 1)$ initialized using "model_init.png" and $p_{\theta^\ell_{0/1}}(x^\ell_i|s_i = 0, 1)$ with initialized parameters as mentioned in Assignment 2.

$$\alpha_\ell(s_i) = p_{u_i,\theta^\ell_1}(s_i|x^\ell_i) = \frac{p_{u_i,\theta^\ell_1}(x^\ell_i, s_i)}{p_{\theta^\ell_1}(x^\ell_i)} = \frac{p_{\theta_1}(x^\ell_i|s_i)p_{u_i}(s_i)}{\sum_{s_i} p_{u_i,\theta^\ell_1}(x^\ell_i, s_i)} = \frac{p_{\theta_1}(x^\ell_i|s_i)p_{u_i}(s_i)}{\sum_{s_i} p_{\theta_1}(x^\ell_i|s_i)p_{u_i}(s_i)} \tag{21}$$

3.2 M-step

We compute u_i using equation 12 and for task to maximize θ we found similarities to maximum likelihood estimation (equations 19, 20)(we maximize parameters for background independently from foreground).

$$\hat{\mu}^\ell_{s_i}, \hat{\Sigma}^\ell_{s_i} = \arg\max_{\mu,\Sigma} \underbrace{\sum_{i\in D} \alpha_\ell(s_i) \log \underbrace{p_{s_i}(x^\ell_i|\mu, \Sigma)}_{s_i \text{ is set}}}_{\text{our loglikelihood}} \tag{22}$$

Knowing that $p_{\theta_1}(x^\ell_i|s_i = 0, 1)$ is given by multivariate normal distribution:

$$p(x|, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} e^{(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))} \tag{23}$$

After computing derivatives with respect to μ, θ we ended up with estimates:

$$\hat{\mu}^\ell_{s_i} = \frac{\sum_{i=1}^m \alpha(s_i)x^\ell_i}{\sum_{i=1}^m \alpha(s_i)} \tag{24}$$

$$\hat{\Sigma}^\ell_{s_i} = \frac{\sum_{i=1}^m \alpha(s_i)(x^\ell_i - \hat{\mu}^\ell_{s_i})^T(x^\ell_i - \hat{\mu}^\ell_{s_i})}{\sum_{i=1}^m \alpha(s_i)} \tag{25}$$

Which are called weighted average and covariance. We compute them for each image and for background and foreground independently.

3.3 Results

As a stopping criteria i choose firstly average accuracy higher than 0.99 but changed later to check relative change of loglikelihood $\Delta\ell = \left|\frac{\ell^t - \ell^{t+1}}{\ell^t}\right|$ and when it is smaller than 10^{-5} then end the algorithm (inspired by: *Convergence Time of the EM Algorithm Depending on the Inital Parameter Values* `https://bit.ly/304aSlh`).

Epoch	0	1	2	3	4	5	6	7	8	9
Accuracy	0.9384	0.977	0.9838	0.9866	0.9881	0.9889	0.9894	0.9897	0.9899	0.99

Table 1: Iterations and average accuracy over all images for EM algorithm

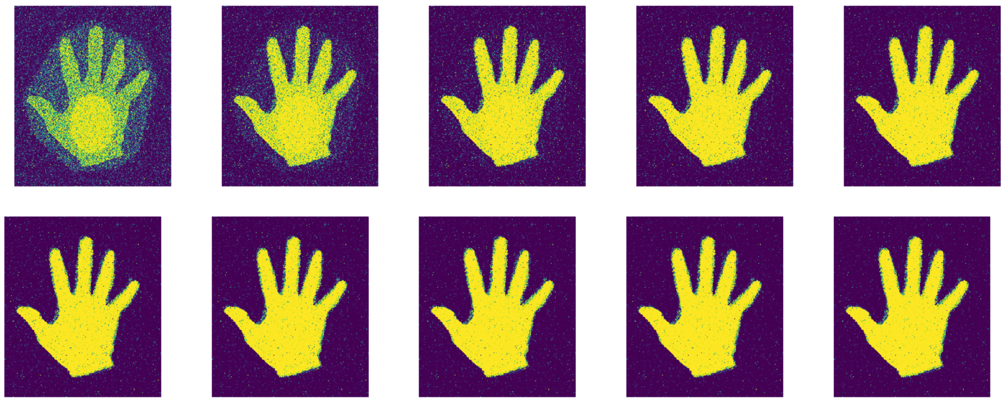

Figure 4: How $\alpha_\ell(s_i = 1)$ changed for "hand_00.png" during iterations (easier to see the borders between foreground and background in yellow-blue scale)

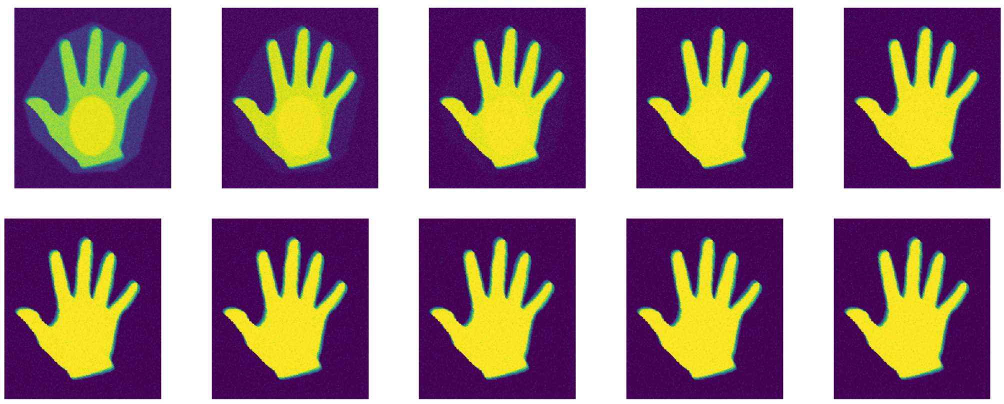

Figure 5: How $p_u(s_i = 1)$ changed during iterations

As we can see after the first iteration we obtain very good accuracy which only gets better and better as algorithm works.

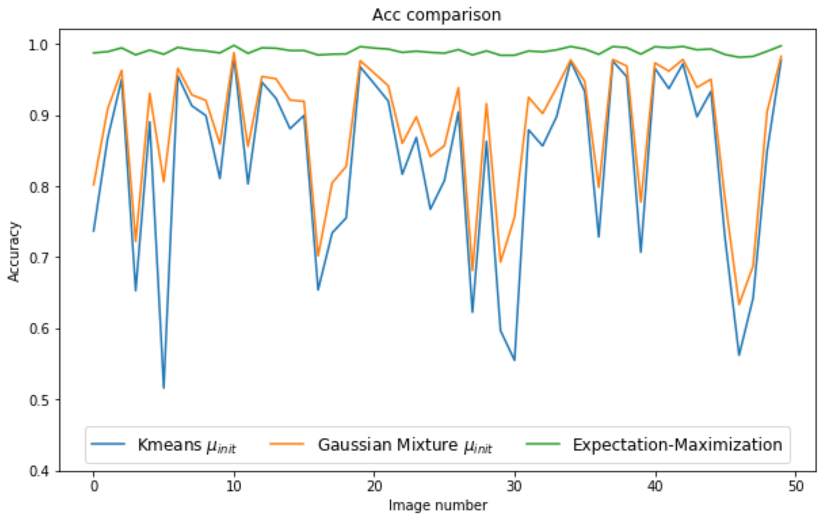

Figure 6: Accuracy comparison for Kmeans, Gaussian Mixture with initialized means and Expected Maximization

There is no comparison between Kmeans or even Gaussian Mixture to Expected Maximization, which achieves best result of accuracy for each image in just 10 epochs.

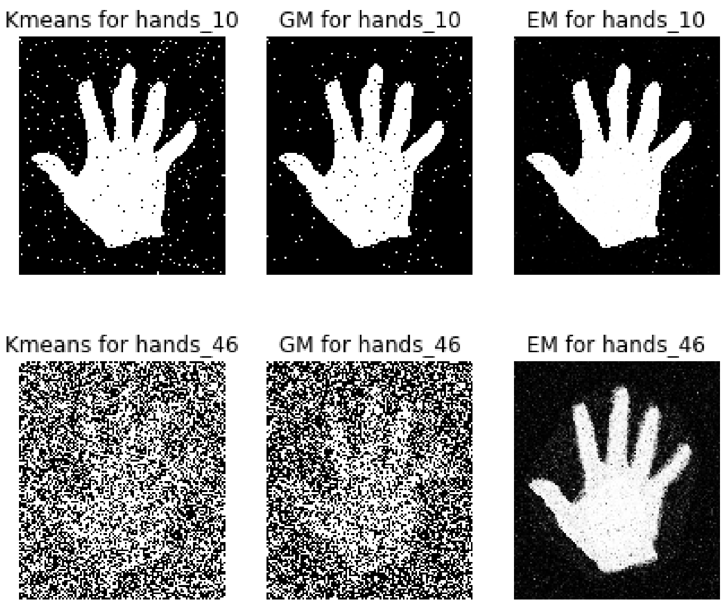

Figure 7: Few examples of segmentation for Kmeans, Gausssian Mixture and Expected Maximization (just plotted values of alpha)