# COGS 109: Assignment #5

Due on Sunday, November 15, 2015

*Tu, Zhuowen 2pm*

**Kyle Lee**
A01614951

# Problem 1

**Problem: (A.1)** One night your burglar alarm goes off. You are told that the probability of your burglar alarm going off when a burglar has entered your house is 95 % . You also know that the probability of it going off when there is no burglar is 1 %. Finally you know that the probability of burglary in your neighborhood is 2 (i.e. 2 in 100).
Let A be the event that a burglary happens.
Let B be the event that a burglar alarm goes off.
Given, $P(B|A) = .95$, $P(B|A^c) = .01$, $P(A) = .02$

(a) What is the prior for having a burglary?
**Solution:** $P(A) = .02$

(b) What is the likelihood for your burglar alarm going off if a burglary happens?
**Solution:** $P(B|A) = .95$

(c) What is the probability of your burglar alarm going off?
**Solution:** $P(B) = P(B|A)P(A) + P(B|A^c)P(A^c) = .95 * .02 + .01 * .98 = .02842$

(d) What is the probability that a burglary actually happened when the burglar alarm went off?
**Solution:** $P(A|B) = \frac{P(B|A)P(A)}{P(B)} = \frac{.95*.02}{.02842} \approx .669$

# Problem 2

**Problem: (A.2)** Now we still study the case of burglar alarm, but we introduce an additional possibility, having earthquake. To simplify the problem here, we consider burglary and earthquake as two disjoint events that dont happen at the same time. The probability of your burglar alarm going off when a burglar has entered your house is still 95 %; the probability of it going off when an earthquake occurs is 99 %; the probability of it going off under normal situations (neither burglary nor earthquake) is 0.5 %. Finally you know that the probability of burglary in your neighborhood is 2 %; the probability of earthquake is 1%.

Let $A_1$ be the event that a burglary indeed happens.
Let $A_2$ be the event that an earthquake occurred.
Let B be the event that a burglar alarm goes off.
Given, $P(B|A_1) = .95$, $P(B|A_2) = .99$, $P(B|(A_1 \cup A_2)^c) = .005$, $P(A_1) = .02$, $P(A_2) = .01$

(a) What is the prior for normal situations (neither burglary nor earthquake)?
**Solution:** $P((A_1 \cup A_2)^c) = 1 - P(A_1 \cup A_2) = 1 - P(A_1) - P(A_2) = 1 - .02 - .01 = .97$

(b) What is the probability of your burglar alarm going off?
**Solution:** $P(B) = P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + P(B|(A_1 \cup A_2)^c)P((A_1 \cup A_2)^c) = .95 * .02 + .99 * .01 + .97 * .005 = .03375$

(c) What is the probability that a burglary actually happened when the burglar alarm went off?
**Solution:** $P(A_1|B) = \frac{P(A_1 \cap B)}{P(B)} = \frac{P(B|A_1)P(A_1)}{P(B)} = \frac{.95*.02}{.03375} \approx .563$

(d) What is the probability that an earthquake actually occurred when the burglar alarm went off?
**Solution:** $P(A_2|B) = \frac{P(A_2 \cap B)}{P(B)} = \frac{P(B|A_2)P(A_2)}{P(B)} = \frac{.99*.01}{.03375} \approx .2933$

(e) Which event mostly likely happened, burglary, earthquake, or normal situations, if your burglar alarm went off, and why?

Problem 2 continued on next page. . . 

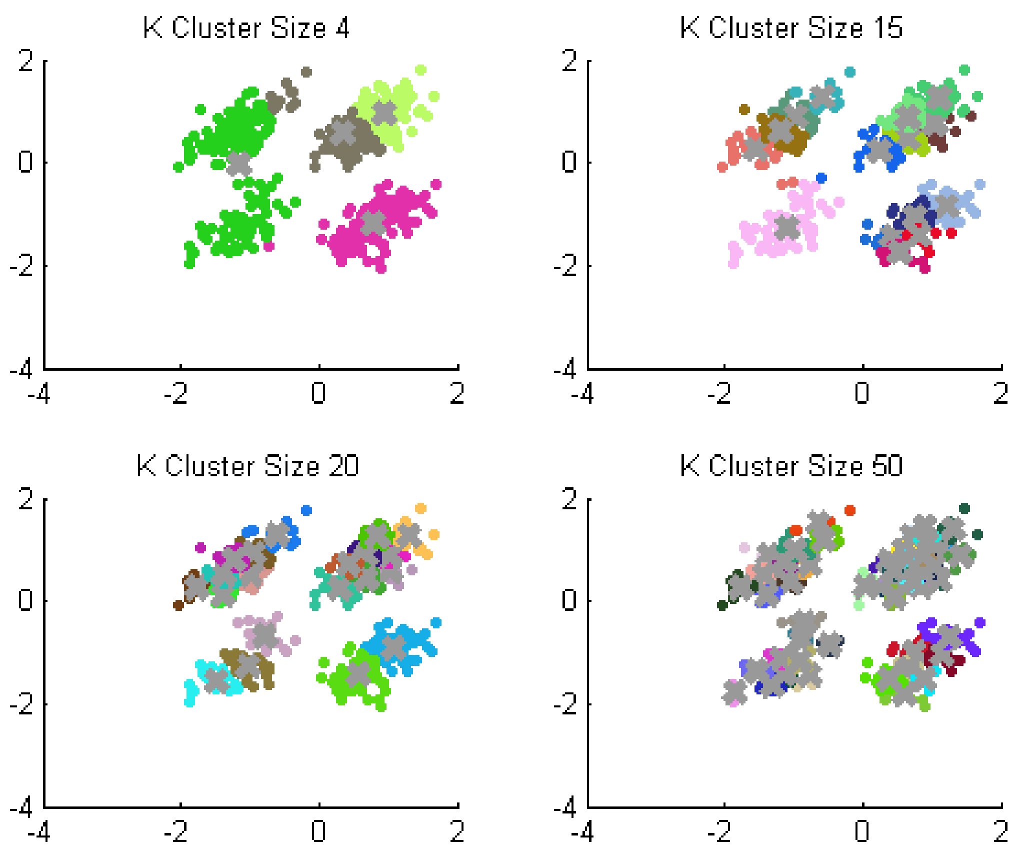
**Solution:** $P((A_1 \cup A_2)^c|B) = \frac{P(B|(A_1\cup A_2)^c)P((A_1\cup A_2)^c)}{P(B)} = \frac{P(B|(A_1\cup A_2)^c)(1-P(A_1)-P(A_2))}{P(B)} = \frac{.005*(1-.02-.01)}{.03375} \approx .144$

A burglary would more like occur if our burglar alarm went off since $P(A_1|B) > P(A_2|B)$ and $P(A_1|B) > P((A_1 \cup A_2)^c|B)$.

## Problem 3

**Problem: (B.1)** Use the same K-means algorithm in the homework 4 to perform clustering on kmeandata, but this time, only plot the final results. The convergence criteria is when the total distance change between centers at two different iterations are less than 0.001 (or you can simply run the algorithm for a large number of iterations e.g. 200). Create a figure that contains two by two subplots. In each subplot, plot the final result of k-means clustering with different k. Subplot 1 uses $K = 4$, subplot 2 uses $K = 15$, subplot 3 uses $K = 20$, and then subplot 4 uses $K = 50$.

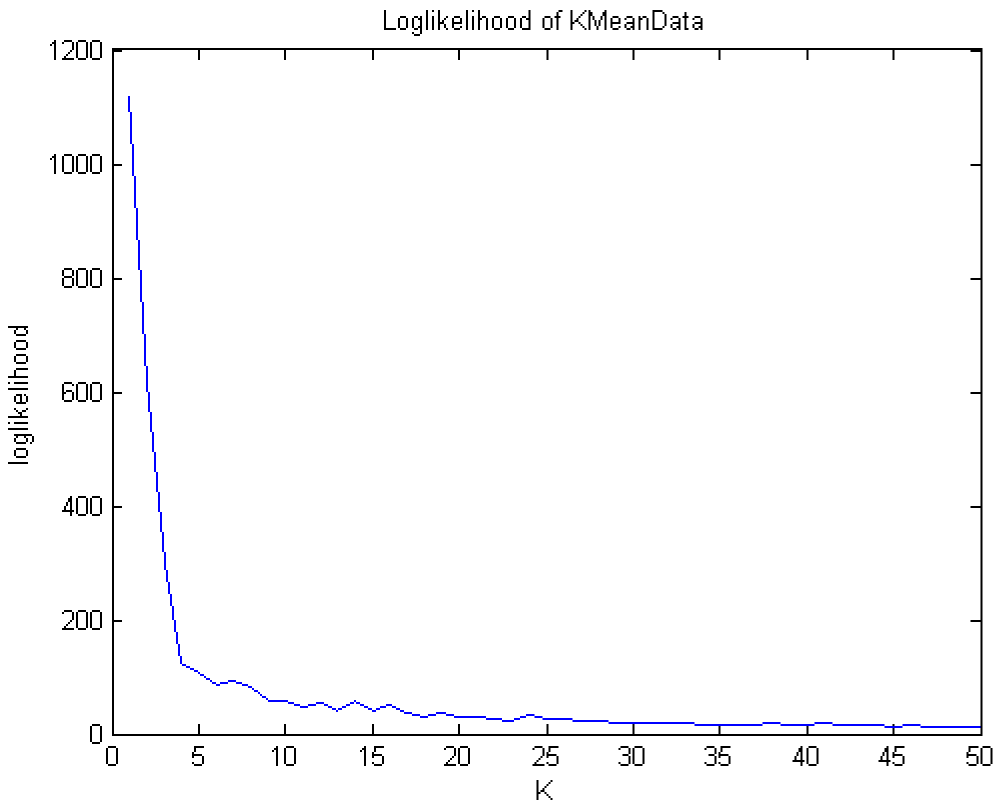
**Solution:**




## Problem 4

**Problem: (B.2)** Repeat the procedure as described in matlab question 1, but this time, try K as 1, 2, 3, ..., 50. (It means that you have to repeat the same procedure for 50 times. In each repetition, use different K instead. It is impossible to do this by copy paste codes repeatedly. Please use a forloop in this part). As defined in the lecture 12 slides 36-37, you can compute the loglikelihood of all your data points based on a mixture model. For the simplicity of this homework assignment, you can simply use the distance of each sample to its closest cluster center as $\log(p(s|\theta))$. In another word, $\log(p(s|\theta))$ is simply the sum of every sample point to its closest cluster center

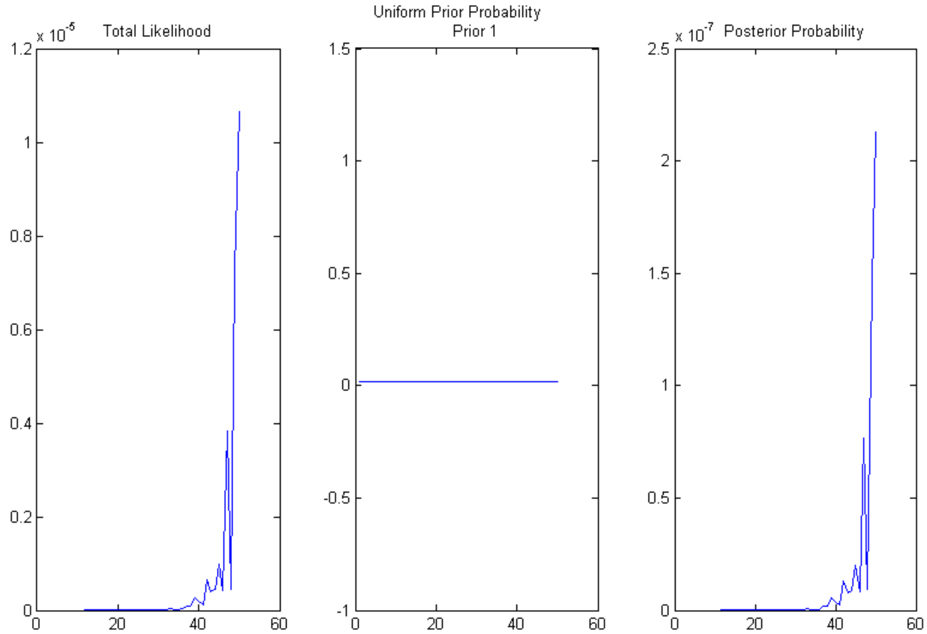
(a) **Problem:** Plot the total $\log(p(s|\theta))$ by summing up all the sample points as a function of K.




(b) **Problem:** Which K explains our data best? Do you think it is a reasonable answer? If not, what happened to our algorithm?

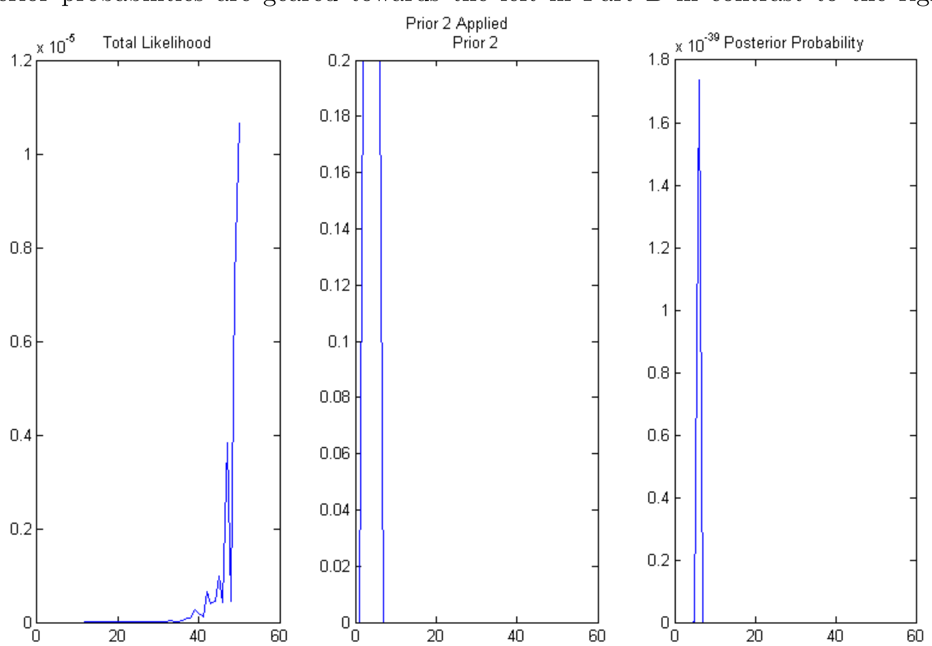
It seems as though $K = 50$ explains our data best which is a perfectly reasonable answer. I would go even further that if we have more clusters, then we will minimize the loglikelihood which reflects some principles of the Law of Large Numbers. For one instance, we get a better fit for our data model if we have more clusters and if we perform a high amount of iterations to reduce our tolerance to nearly 0. But at some point, having more clusters than data points would be self-defeating.. The more clusters we have, the more we are able to accurately classify the mean data. We observe that the loglikelihood as a function of $K$ is not strictly monotonic, but we can create subsequences of $K$ so that we have a strictly monotonic subsequence.

# Problem 5

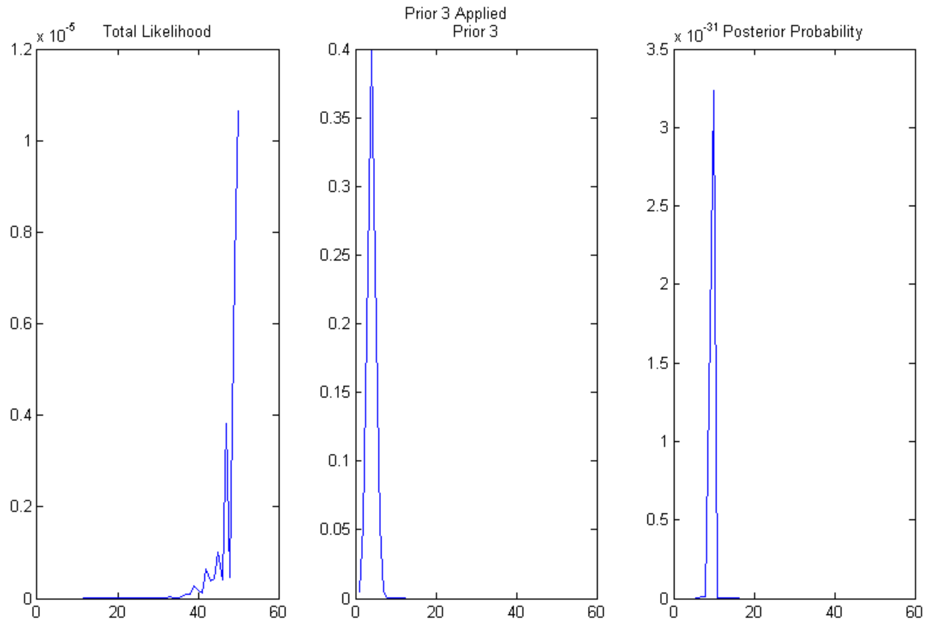



(a)

(b) We observe that Part B is different from Part A in the degree that the posterior probabilities are very different. In Part B, we see that we assume a uniform distribution in the interval from $[2, 6]$. The posterior probabilities are geared towards the left in Part B in contrast to the right in Part A.




(c) We observe that Part C is different from Part A in the degree that the posterior probabilities are very different. The prior probabilities does not have much of an effect since most of the weight of the likelihood function is around $K \approx 45$. In Part A, we assume a uniform distribution in the interval from $[2, 6]$ while prior 3 assumes a somewhat skewed shifted normal distribution.




```
%% Homework #4 %%
%% Problem 1 %%

% define # of clusters
K = [4,15,20,50];
% store in figure 1
figure(1);
for i=1:length(K)
    % define centers
    centers = kmeandata(randi(size(kmeandata,1),K(i),1),:);

    for t=1:200
        % calculate distance
        for j=1:K(i)
            dist(:,j) = sqrt((kmeandata(:,1) - centers(j,1)).^2 ...
                +(kmeandata(:,2) - centers(j,2)).^2);
        end

        % Assign each data point with the id of its nearest cluster
        [v g_ind] = min(dist,[],2);


        % recalibrate centers
        for p=1:K(i)
            centers(p,:) = mean(kmeandata(g_ind == p,:));
        end
    end

    % draw plots
    subplot(2,2,i);drawnow;

    % plot data and recalibrated centers
```

```
    for m = 1:K(i);
        colorspec = rand(1,3);
        scatter(kmeandata(g_ind == m,1),kmeandata(g_ind == ...
            m,2),20,colorspec,'filled');
        hold on;

        % draw recalibrated centers
        scatter(centers(m,1),centers(m,2),80,'Marker','x', ...
            'MarkerEdgeColor',[0.6 0.6 0.6],'LineWidth',4);
    end
    title(sprintf('K Cluster Size %d', K(i)))
    hold off;
end

clear
load('HW5_data.mat');
%% Problem 2 %%

% let K be a matrix going from 1-50
K = [1:1:50];
iter = 10;
% store loglikelihood
nlglvec = zeros(length(K),1);
% store in figure 2
figure(2);
for i=1:length(K)
    % define centers
    centers2 = kmeandata(randi(size(kmeandata,1),K(i),1),:);
    dist2=[];
    for t=1:200
        % calculate distance between kmeandata and centers
        for j=1:K(i)
            dist2(:,j) = sqrt((kmeandata(:,1) - centers2(j,1)).^2 ...
                +(kmeandata(:,2) - centers2(j,2)).^2);
        end

        % assign each data point with the id of its nearest cluster
        [v2 g_ind2] = min(dist2,[],2);

        % sum for loglikelihood
        nlglvec(i) = sum(v2.^2);

        % recalibrate centers
        for n=1:K(i)
            centers2(n,:) = mean(kmeandata(g_ind2==n,:));
        end
    end
end
% plot and label
plot(K,nlglvec)
title('Loglikelihood of KMeanData')
xlabel('K')
ylabel('loglikelihood')
```

```
%% Problem 3 %%
% create figure 3 plotting prior1, loglikelihood, and applied prior1
figure(3)
subplot(1,3,1)
plot(K,exp(-nlglvec))
title('Total Likelihood')
subplot(1,3,2)
plot(K,prior1)
title('Prior 1')
subplot(1,3,3)
plot(K,exp(-nlglvec).*prior1)
title('Posterior Probability')
annotation('textbox', [0 0.9 1 0.1], ...
    'String', 'Uniform Prior Probability', ...
    'EdgeColor', 'none', ...
    'HorizontalAlignment', 'center')

% create figure 4 plotting prior2, loglikelihood, and applied prior2
figure(4)
subplot(1,3,1)
plot(K,exp(-nlglvec))
title('Total Likelihood')
subplot(1,3,2)
plot(K,prior2)
title('Prior 2')
subplot(1,3,3)
plot(K,exp(-nlglvec).*prior2)
title('Posterior Probability')
annotation('textbox', [0 0.9 1 0.1], ...
    'String', 'Prior 2 Applied', ...
    'EdgeColor', 'none', ...
    'HorizontalAlignment', 'center')

% create figure 5 plotting prior3, loglikelihood, and applied prior3
figure(5)
subplot(1,3,1)
plot(K,exp(-nlglvec))
title('Total Likelihood')
subplot(1,3,2)
plot(K,prior3)
title('Prior 3')
subplot(1,3,3)
plot(K,exp(-nlglvec).*prior3)
title('Posterior Probability')
annotation('textbox', [0 0.9 1 0.1], ...
    'String', 'Prior 3 Applied', ...
    'EdgeColor', 'none', ...
    'HorizontalAlignment', 'center')
```