Clustering using GMM

Characterizing heterogeneous cellular responses to perturbations

Slack et. al

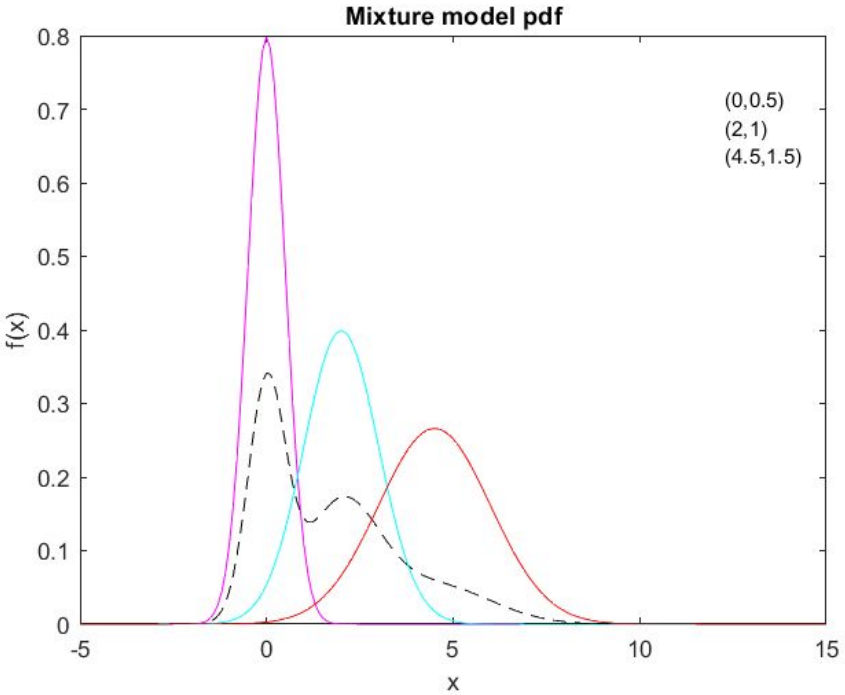
Project Aim

Understand the theory of GMM.

Perform clustering on a reduced dataset using GMM

Look at the application of the method, in a research paper.

Gmm - Motivation



Gaussian Mixture Model is a probabilistic model to represent normally distinct subpopulations within overall population.

We assume that the multimodal data has been generated as the sum of multiple unimodal Gaussians.

Gmm maintains the benefits of Gaussian models.

Multivariate GMM , parameters

A multivariate GMM is parametrized by two types of values:

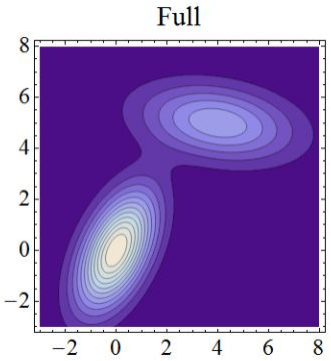
- Mixture component weights : $\phi_i,\ \Sigma\phi_i = 1$
- Model parameters : $\vec{\mu_k}$,Σ_k
- $\vec{\mu_k}$= $[E(X_1) \quad E(X_2)..... \quad E(Xn)]$
- $\Sigma_k|_{i,j}$ = Cov(X_i,X_j)
- $P(\vec{x}) = \sum_{i=1}^{k} \Phi_i N(\vec{x}|\vec{\mu_i}\ ,\ \Sigma_i)$

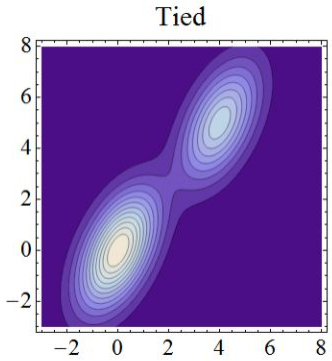
Types of Covariance Matrices

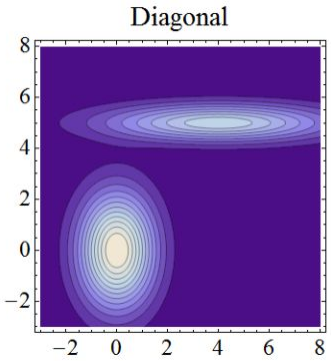
2-D Covariance Matrix:

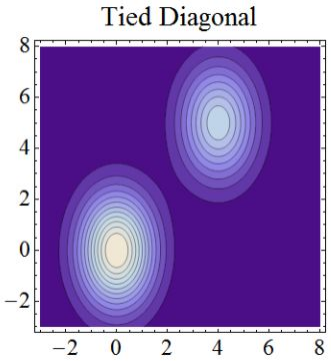
$$\begin{matrix} \sigma x_1^2 & cov(x_1, x_2) \\ cov(x_1, x_2) & \sigma x_2^2 \end{matrix}$$

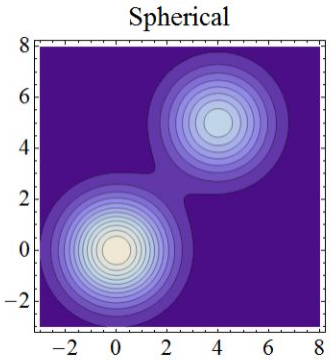
1) Full : non-diagonal terms are non-zero.
2) Tied(Shared) : All components share same covariance matrix.
3) Diagonal : Each component has its own diagonal covariance matrix. Correlation=0.
4) Spherical: Diagonal covariance matrix with equal values of variance.











(inclination) (same shape) (along axes)

Hard and Soft Clustering

Hard Clustering : Each data point is assigned to exactly one cluster. Component with maximum probability is chosen.

Soft Clustering : Data points belong to multiple clusters with different probabilities.

Soft Clustering is useful when assessing similarity to different groups or relationship to different groups of an item.

EM algorithm - parameter estimation

- MLE by differentiating log likelihood is analytically very difficult.
- EM is a numerical technique for MLE.
- MLE strictly increases with each iteration.

```
%Step 1 - initialize means with KMeans
[labels,mu] = kmeans(x,K);

% Compute weights and covariance matrices using MLE
for k = 1:K
    weight(k) = sum(labels == k)/N;
    Sigma{k} = cov(x(labels==k,:));
end

convergence = 0;
t = 1;

figure;
while t==1 || ~convergence %convergence criteria

    % Step 2: Expectation - Compute posteriors of each cluster
    for n = 1:N
        for k = 1:K
           posterior(n,k) = weight(k) * mvnpdf(x(n,:),mu(k,:),Sigma{k});
        end
        posterior(n,:) = posterior(n,:)./sum(posterior(n,:));
    end

    visualize(x,mu,Sigma,posterior);

    % Step 3: Maximization; i.e. update model parameters (means, covaria
    % matrices and weights
    for k = 1:K
        % Update means
```

- E(Expectation step): Calculate the expectation of Component assignment for each x_i , given model parameters(Randomly or otherwise).

```
% Step 3: Maximization; i.e. update model parameters (means, covaria
% matrices and weights
for k = 1:K
    % Update means
    sp(k) = sum(posterior(:,k));
    mu(k,:) = sum(bsxfun(@times,posterior(:,k),x))/sp(k);
     % Update covariance matrices
    Sigma{k} = zeros(nfeatures,nfeatures);
    for n = 1:N
        Sigma{k} = Sigma{k} + posterior(n,k)*(x(n,:)-mu(k,:))'*(x(n,
    end
    Sigma{k} = Sigma{k}./sp(k);
end
%Update the weights
weight = sp./sum(sp);
      % Step 4: Evaluation - compute log likelihood
llh(t) = 0;
for i = 1:n
    innerterm = 0;
    for k = 1:K
        innerterm = innerterm + (weight(k) * mvnpdf(x(i,:),mu(k,:),S
    end
    llh(t) = llh(t) + log(innerterm);
end
llh(t) = llh(t) / n;

if t > 1
    convergence = (llh(t) - llh(t-1)) < 0.000001;
end
```

- M(Maximisation step): E calculated is maximised with respect to model parameters. Parameter values are updated.

Data Visualisation

Ca response of HeLa cells for 339 time steps

In [4]:
```
y=pd.read_excel('dataclust.xlsx')
y
```

Out[4]:

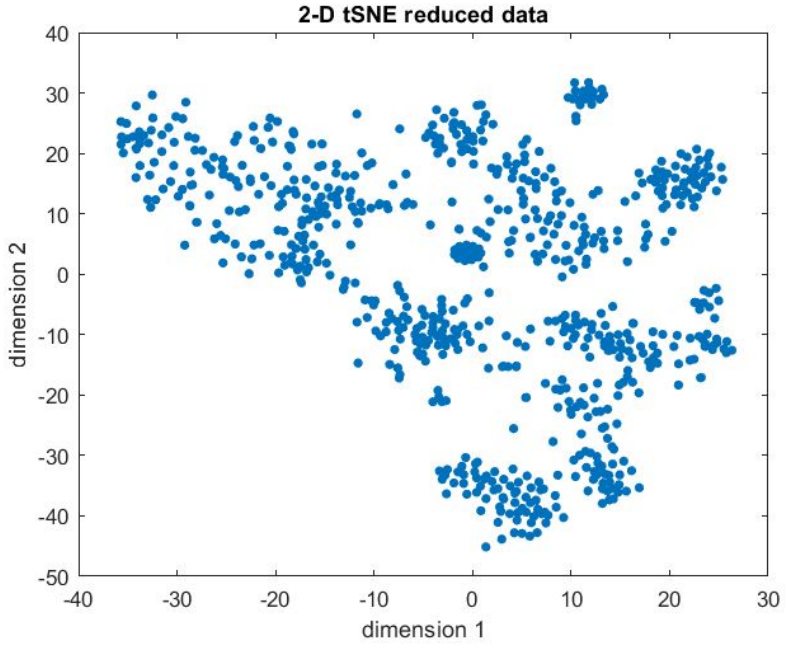
	1	2	3	4	5	6	7	8	9	10	...	747	748	749	750	751
0	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	...	1.000000	1.000000	1.000000	1.000000	1.000000
1	1.057755	1.040279	1.038932	1.036171	1.035690	1.019917	1.039168	1.021691	1.055702	1.058192	...	1.000000	1.000000	1.000000	1.000000	1.000000
2	1.113278	1.079068	1.076290	1.071082	1.069836	1.039236	1.076971	1.042517	1.109743	1.114332	...	1.009395	1.000000	1.006176	1.004041	1.009950
3	1.074079	1.054648	1.046887	1.052527	1.038519	1.033205	1.056914	1.026602	1.093340	1.083391	...	1.037915	1.000000	1.024926	1.016309	1.040154
4	1.037806	1.032323	1.019676	1.035624	1.009811	1.027713	1.039076	1.012361	1.078914	1.055200	...	1.028920	1.007791	1.019012	1.012440	1.030628
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
334	1.055944	1.051892	1.025236	1.053394	1.020252	1.123837	1.009974	1.018714	1.013406	1.152934	...	1.021002	1.019458	1.022853	1.019638	1.109049
335	1.050897	1.059033	1.021677	1.052365	1.014423	1.094077	1.012147	1.016420	1.008937	1.122026	...	1.038614	1.033554	1.032694	1.033357	1.099161
336	1.042054	1.060896	1.016642	1.048260	1.009942	1.068946	1.011330	1.019762	1.005958	1.097918	...	1.063969	1.052718	1.049757	1.053899	1.095883
337	1.031905	1.055671	1.011649	1.040843	1.006628	1.048479	1.009058	1.021850	1.003972	1.076345	...	1.074084	1.055122	1.058783	1.062549	1.090942
338	1.022370	1.041398	1.007923	1.029693	1.004419	1.033034	1.006466	1.017033	1.002648	1.054042	...	1.066869	1.046661	1.054056	1.056638	1.076258

339 rows × 756 columns

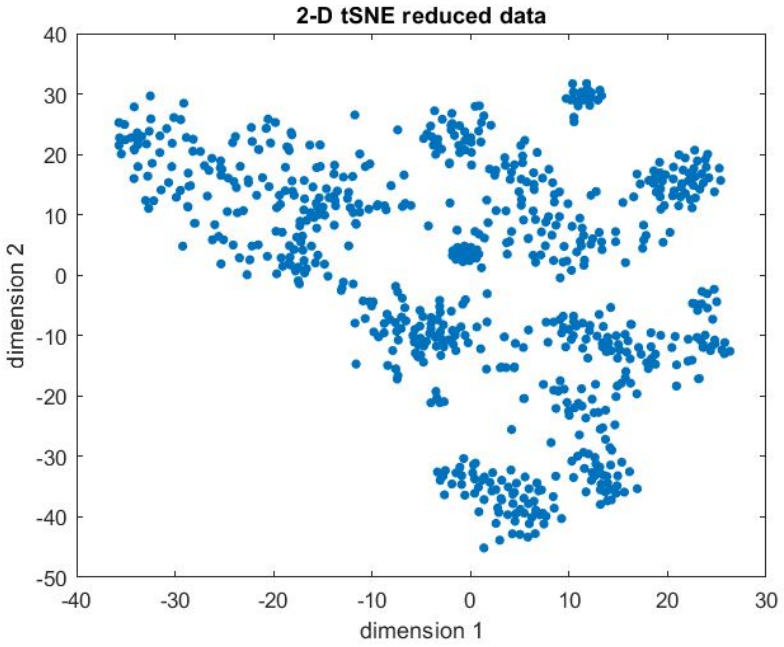
Data -tSNE reduction

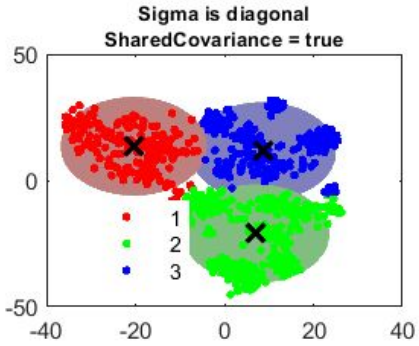
The Data is reduced by tSNE to

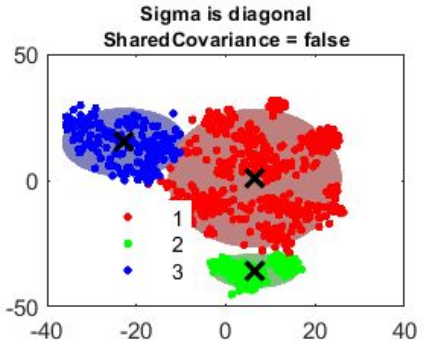
756 rows and two dimensions.

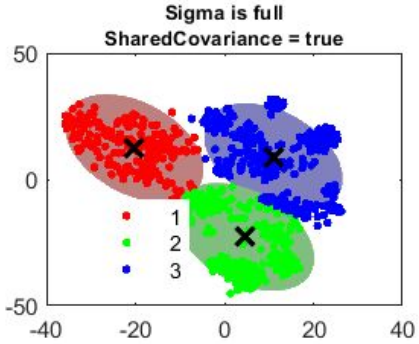


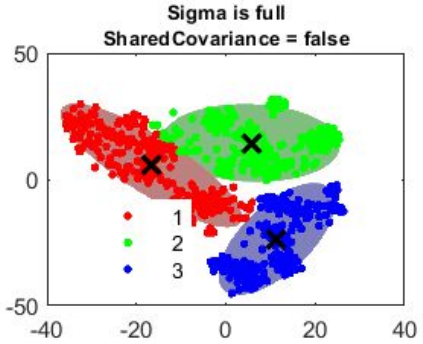
Fitting GMM with different covariances



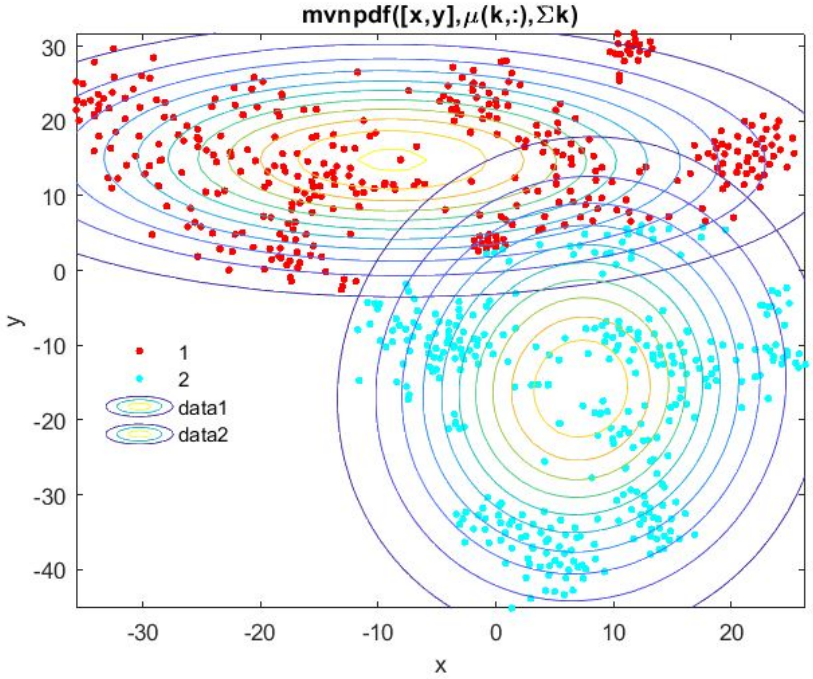




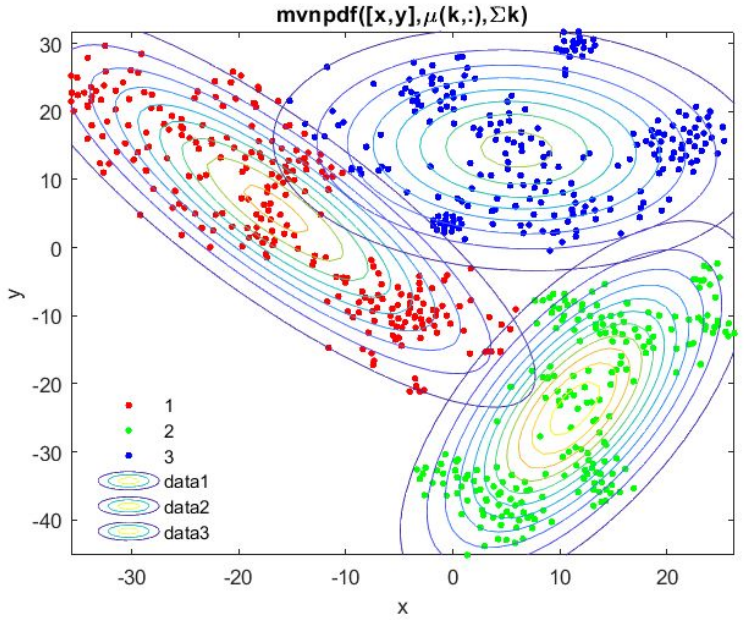




How many clusters is optimal ?

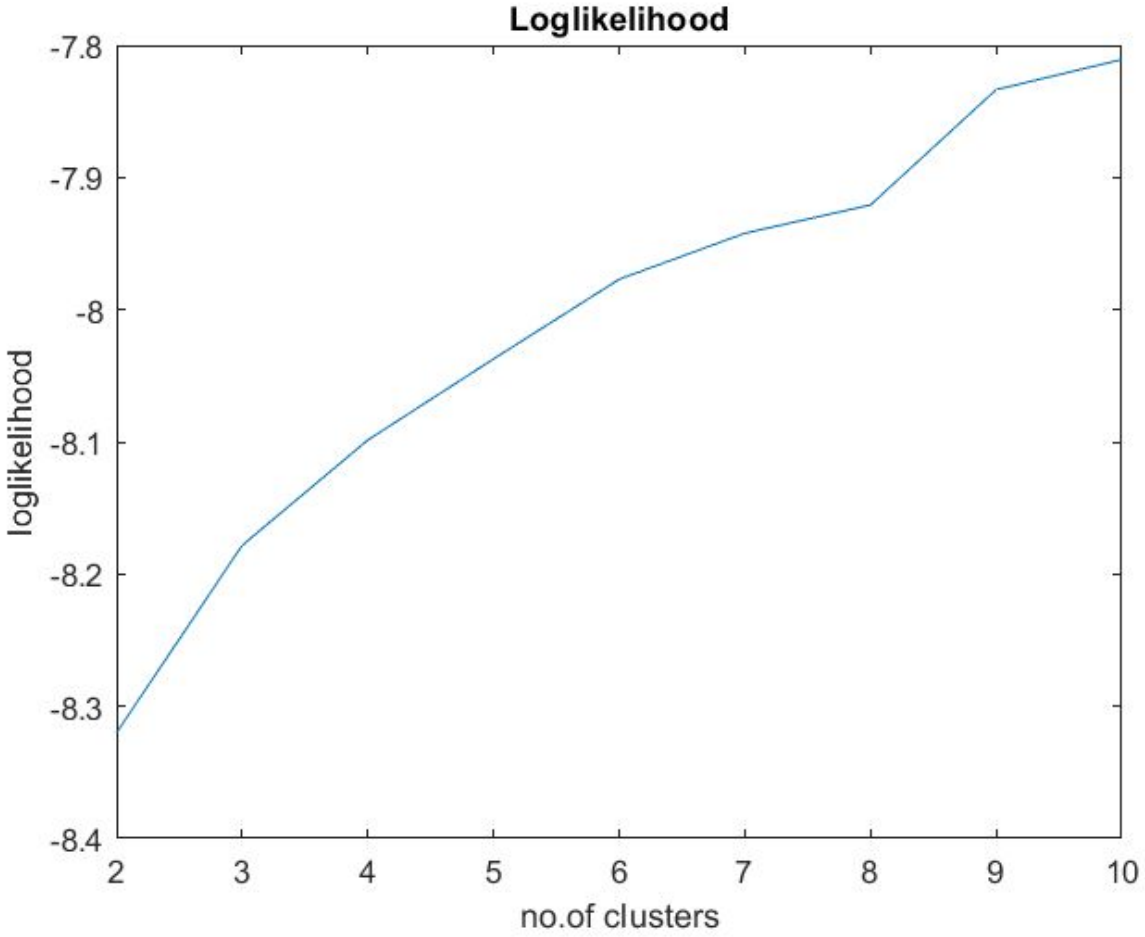


k=2
llh=-8.3201



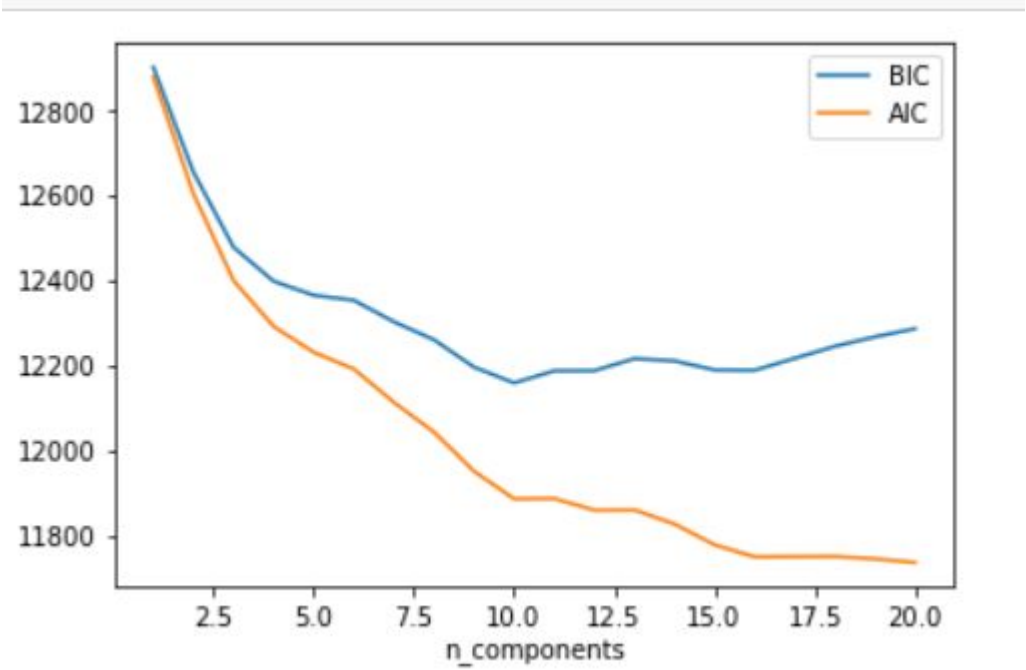
k=3
llh=-8.1783

No. of Clusters	Log Likelihood
2	-8.3201
3	-8.1783
4	-8.0986
5	-8.0373
6	-7.9770
7	-7.9422
8	-7.9208
9	-7.8336
10	-7.8108



Log Likelihood monotonically increases.

Bayesian Information Criterion - BIC



(Sample)

BIC is a method for scoring and selecting a model.

- BIC = -2 * LL + log(N) * k

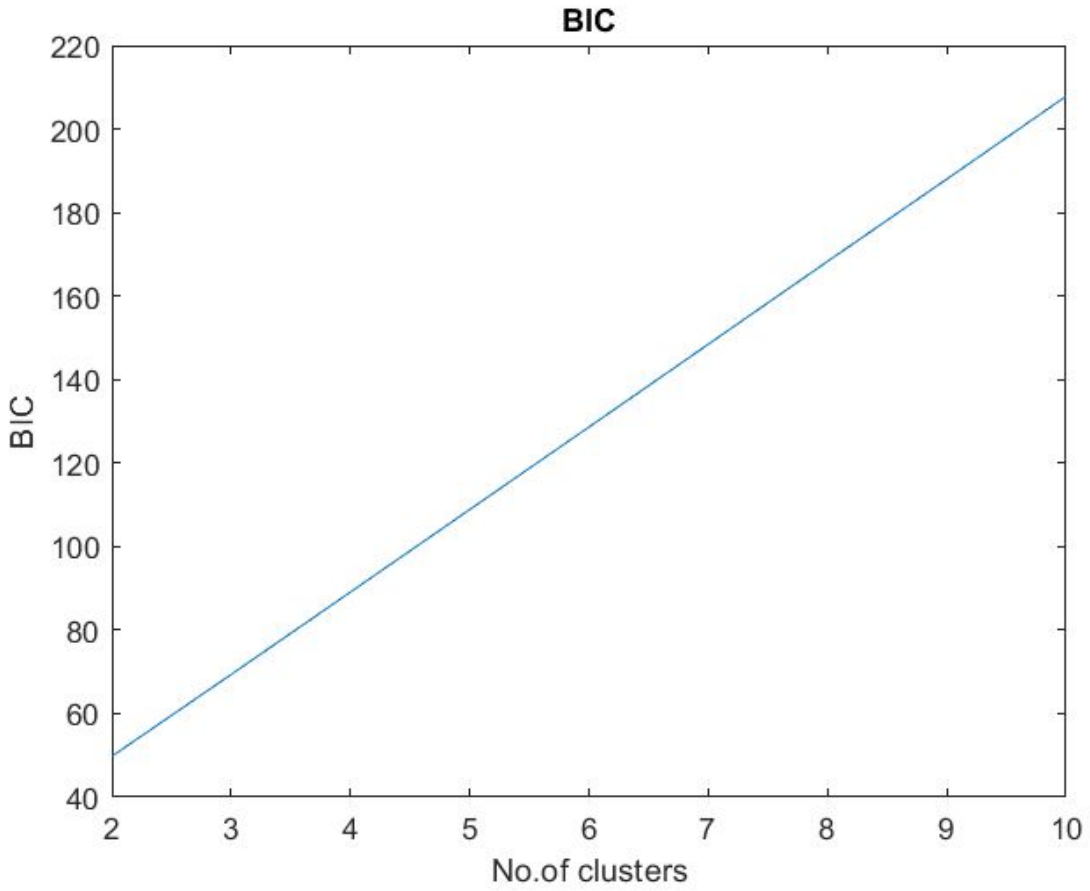
Where *log()* has the base-e called the natural logarithm, *LL* is the log-likelihood of the model, *N* is the number of examples in the training dataset, and *k* is the number of parameters in the model.

k(no. Of parameters - GMM)

- For a GMM with n components, no. of parameters = 3*n -1

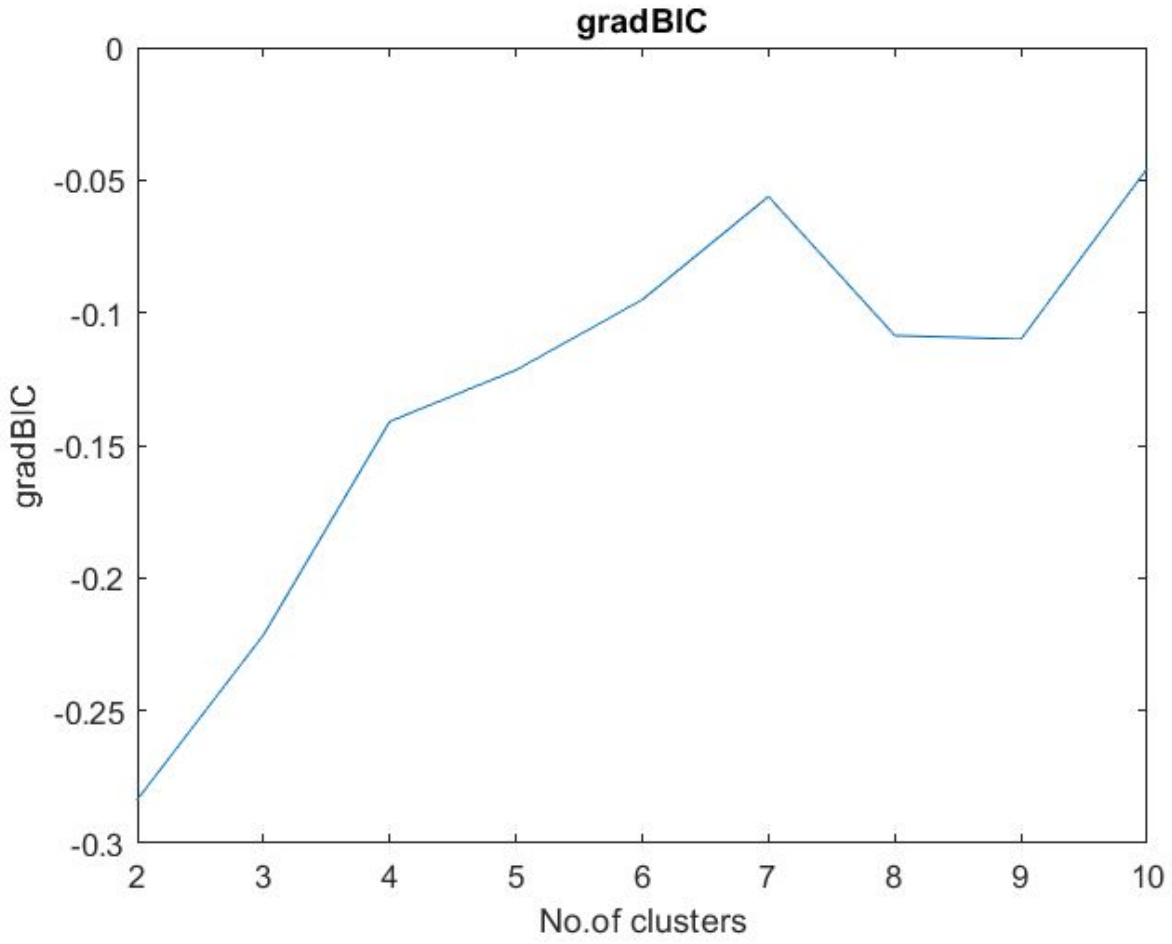
- This is because the parameters are: mean, covariance matrix and component weight for each mixture model.

- However, since $\Sigma\phi_i = 1$, the no. of parameters reduces by 1.



BIC v/s no. of Clusters

- The curve is unexpectedly monotonically increasing, and offers us no help in finding the optimal no. of clusters.
- Hence, we look if the gradient of BIC to help us.



We want to select a low number of clusters (at least less than 10).

The gradient of BIC , in many cases, becomes nearly constant as the number of clusters increases(no additional information recovered).

k=6,7,8 all possibilities for optimal cluster no.

KL divergence to compare distributions

Kullback-Leibler Divergence

The Kullback-Leibler Divergence score, quantifies how much one probability distribution differs from another probability distribution.

The KL divergence between two distributions Q and P is often stated using the following notation:

- KL(P || Q) = sum x in X P(x) * log(P(x) / Q(x))

Where the “||” operator indicates “*divergence*” or Ps divergence from Q

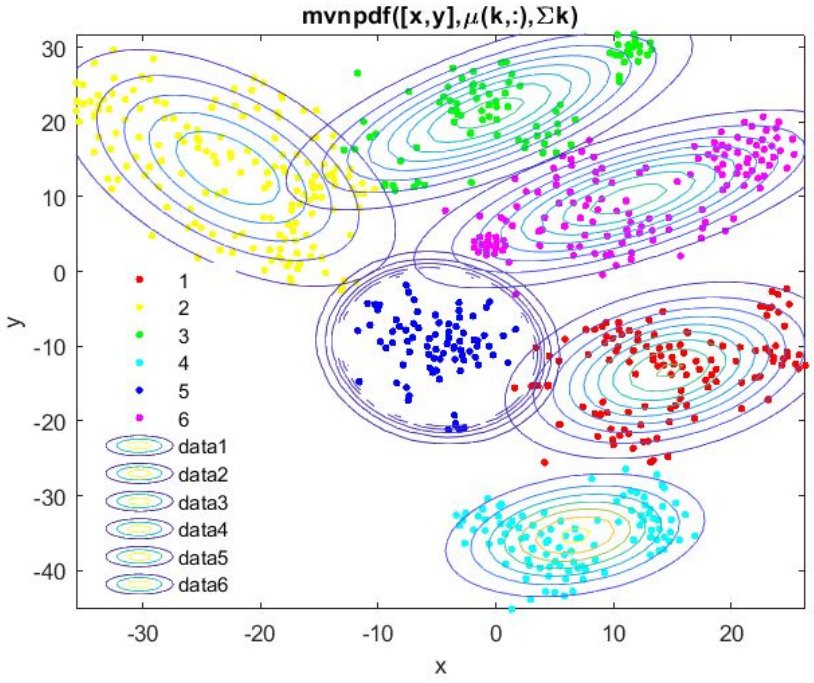
```
clear
close all
clc
[a,b,c,d,e,f,g,h]=GMMClustering(8)
%%
x=xlsread('tSNE_DR.xlsx');
b=zeros(size(x));
sigma=cat(3,g(1,1),g(1,2),g(1,3),g(1,4),g(1,5),g(1,6),g(1,7),g(1,8));
sigma=cell2mat(sigma);
for n=1:size(x)
    for k=1:8
        y(n,k)=h(k)*mvnpdf(x(n,:),f(k),sigma(:,:,k));
        b(n)=b(n)+y(n,k);
    end
end
```

```
%%
p=zeros(756,2);
for j=1:2
    for i=1:756
        p(i,j)= min(x(i,j))+(i-1)*((max(x(i,j))-min(x(i,j)))/756);
    end
end
 [u,xi]=ksdensity(x,p)
o=zeros(756,1);
r=zeros(756,1);
for i=1:756
    o(i)=log(u(i)/b(i));
    r(i)=u(i)*o(i);
end
kl=sum(r)
```

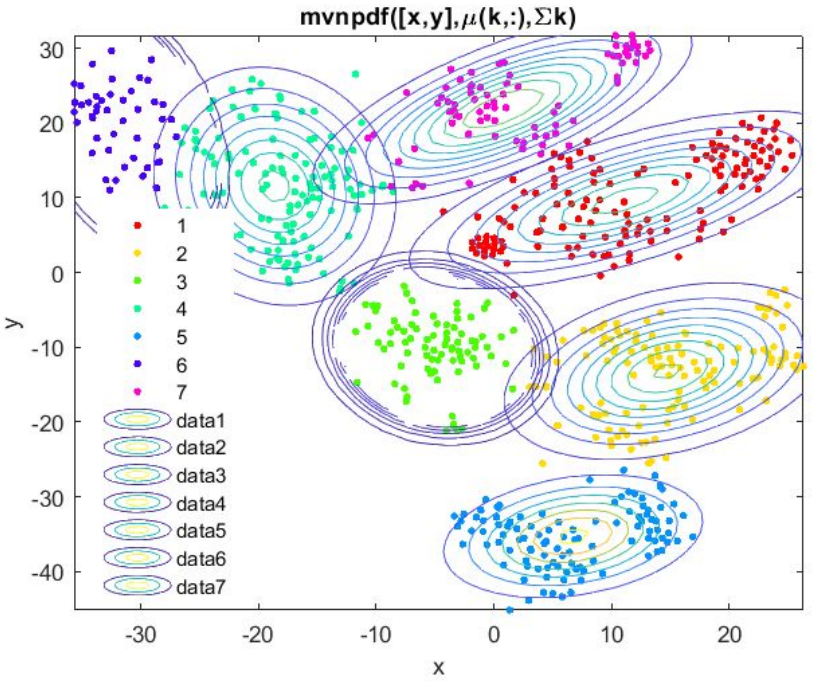
We thus find the KL divergence between given data and our distribution. Kernel function gives pdf of given data.

No. of Clusters	KL Divergence Values
6	0.6273
7	0.7760
8	0.9721

k=6 and k=7 have similar log likelihood values as well (-7.977 and -7.942) as low KL divergence.
Hence, they both seem to be optimal clusters .


mvnpdf([x,y],μ(k,:),Σk)
y
x
1
2
3
4
5
6
data1
data2
data3
data4
data5
data6

k=6


mvnpdf([x,y],μ(k,:),Σk)
y
x
1
2
3
4
5
6
7
data1
data2
data3
data4
data5
data6
data7

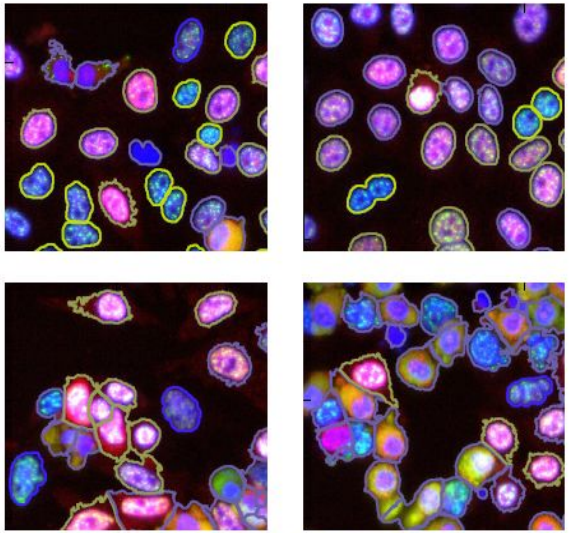
k=7

Paper Explanation

- Slack et. al characterise heterogeneous cellular responses to perturbations by analysing drug-treated HeLa cells.
- They investigate if the heterogeneity contains biologically important information.
- They are investigating the possibility that the cellular population can be described as a mixture of phenotypically distinct subpopulations.
- USL using GMM clustering is used to identify subpopulations and assign probabilities to cells belonging to subpopulations.

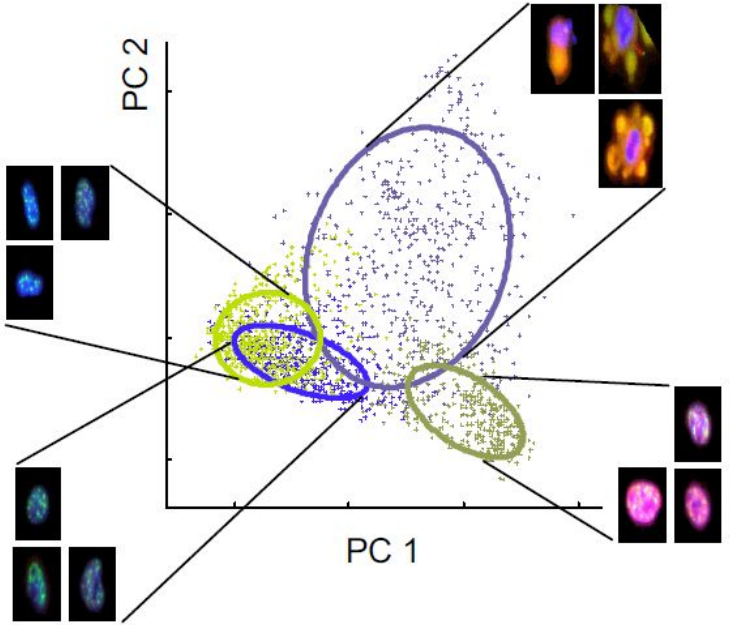
A

Image database



B

Feature Space



A: HeLa cells treated with drugs and labelled with fluoroscent markers (4 sub populations shown)

B: High dimensional data of phenotypic features of cells reduced to 2 dimensions using PCA and GMM clustering done. Ellipses represent deviation from mean of each subpopulation.

Project Guide:
Dr. Lopamudra Giri,
Assistant Professor,
Department of Chemical Engineering,
IIT Hyderabad

Submitted by:

Jacob Kunnathoor

IIT Hyderabad