S21CSE471: Statistical Methods in AI **8-March Spring 2021**

Gaussian Mixture Models

Prepared by: Abhiram - 2020201030, Amanpreet Kaur - 2020202005, Adrija Chakraborty- 2020201063

In this note, we will examine a popular alternative to k-means clustering – Gaussian mixture modeling with Expectation-Maximization. First we will see the drawbacks that led to the concept of GMM, then we will discuss the working of Gaussian Mixture Models.

1 Anomaly Detection

Anomaly Detection (or outlier detection) is the identification of rare items, events or observations which raise suspicions by differing significantly from the majority of the data. Typically, anomalous data can be connected to some kind of problem or rare event such as e.g. bank fraud, medical problems, structural defects, malfunctioning equipment etc. This connection makes it very interesting to be able to pick out which data points can be considered anomalies, as identifying these events are typically very interesting from a business perspective.
How do we identify whether data points are normal or anomalous? In some simple cases, as in the example figure below, data visualization can give us important information.

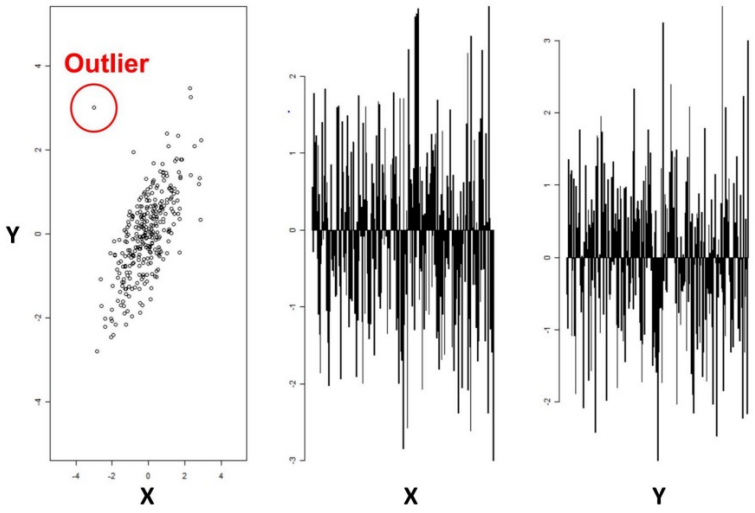

Figure 1: Anomaly Detection for two variables

In this case of two-dimensional data (X and Y), it becomes quite easy to visually identify anomalies through data points located outside the typical distribution. However, looking at the figures to the right, it is not possible to identify the outlier directly from investigating one variable at the time: It is the combination of the X and Y variable that allows us to easily identify the anomaly. This complicates the matter substantially when we scale up from two variables to 10-100s of variables, which is often the case in practical applications of anomaly detection.
As we have noted above, for identifying anomalies when dealing with one or two variables, data visualization can often be a good starting point. However, when scaling this up to high-dimensional data (which is often the case in practical applications), this approach becomes increasingly difficult. This is fortunately where multivariate statistics comes to help.
When dealing with a collection of data points, they will typically have a certain distribution (e.g. a Gaussian distribution). To detect anomalies in a more quantitative way, we first calculate the probability distribution $p(X)$ from the data points. Then when a new example, x, comes in, we compare $p(x)$ with a threshold r. If $p(x) < r$, it is considered as an anomaly. This is because normal examples tend to

have a large $p(x)$ while anomalous examples tend to have a small $p(x)$.

Let's say we have unlabelled training set of m examples: $\{x_{(1)}, x_{(2)}, .. , x_{(m)}\}$, where each x is a vector of n features.
Now we are going to model the $p(X)$ as

$$p(X) = p(x_1) * p(x_2) * .. * p(x_n)$$

We assume that all these features are distributed according to some Gaussian distribution with x_i having mean μ_i and variance σ_i^2. So,

$$p(X) = p(x_1, \mu_1, \sigma_1^2) * p(x_2, \mu_2, \sigma_2^2) * .. * p(x_n, \mu_n, \sigma_n^2)$$
$$p(X) = \prod_{i=1}^{n} p(x_i, \mu_i, \sigma_i^2)$$

So for any new data point x_p, we calculate $p(x_p)$ and check if $p(x_p)$ >= threshold, if yes then x_p is not an anomaly, otherwise it is an anomaly.

2 Introduction

The major problem with K- means clustering is the hard assignment of clusters. Each point is associated with only one cluster, and there is no uncertainty measure or probability that tells us how much a data point is associated with a specific cluster. Here comes the idea of using a soft clustering method, where data points can belong to multiple clusters at the same time but with different degrees of belief. Hence, instead of treating the data as a bunch of points, we assume that they are all generated by sampling a continuous function. This function is called a generative model. Data distribution generally follows normal distribution. So, by combining several Gaussian models, we can approximate any continuous density distribution.
Let us see how to follow this idea mathematically.

2.1 Revisiting Gaussian distribution

Before we go into Gaussian Mixture Models, let us quickly recap what a Gaussian distribution looks like. The mathematical form of the Gaussian distribution in 1-dimension (uni-variate Gaussian) can be written as:

$$\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \tag{1}$$

where,

μ : mean of the Gaussian that defines its centre.

σ : covariance that defines the width of the Gaussian.

x : random observation where this distribution is placed

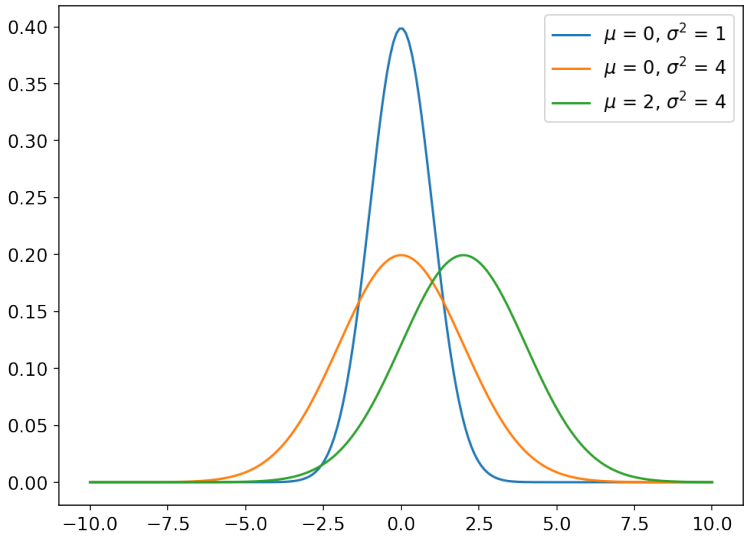

Figure 2: Gaussian Distribution with different values of the mean and variance

2.2 GMM parameters

It turns out the univariate (one-dimensional) Gaussian can be extended to a multivariate (multi-dimensional) case. The form of a d- dimensional Gaussian is as follows:

$$N(x|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}exp\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\} \tag{2}$$

where,

μ : mean vector of different Gaussians that defines their centres.

Σ : d -by- d covariance matrix.

π : mixing probability that defines how big or small the Gaussian will be.

A GMM has an equivalent representation as a generative model for our data:

$$z_i \sim^{\text{iid}} Mult(\pi, 1) \tag{3}$$
$$x_i|z_i \sim N(\mu_{z_i}, \Sigma_{z_i}) \tag{4}$$

where z_i represents the latent component indicator or latent class / cluster for data point x_i.

3 Clustering with GMM

Let,

$$\phi\left(x;\mu_j,\Sigma_j\right) = \frac{1}{\left|2\pi\Sigma_j\right|^{1/2}}\exp\left(-\frac{1}{2}\left(x-\mu_j\right)^T\Sigma_j^{-1}\left(x-\mu_j\right)\right)$$

Under the GMM, our clustering task amounts to inferring the latent component z_i responsible for each x_i. For the moment, let us ignore the fact that we do not know the parameters of the GMM and imagine how we would carry out the clustering task given (,1:k,1:k).Since a GMM with known parameters defines

a joint distribution over (xi,zi), it is natural to consider the conditional distribution of each z_i given x_i:

$$p\left(z_i = j \mid x_i\right) = \frac{p\left(z_i = j\right) p\left(x_i \mid z_i = j\right)}{p\left(x_i\right)}$$
$$= \frac{\pi_j \phi\left(x_i; \mu_j, \Sigma_j\right)}{\sum_{l=1}^{k} \pi_l \phi\left(x_i; \mu_l, \Sigma_t\right)}$$

These conditionals reflect our updated beliefs concerning z_i after x_i is observed: before we observe x_i , we have the prior belief that it belongs to cluster j with probability π_j; after observing x_i , we can update this belief in accordance with the likelihood of x_i under each Gaussian component The conditional distribution provides us with what is called a soft clustering since it assigns some probability tox_i belonging to each cluster.

4 Expectation Maximization Algorithm

Expectation maximization is an iterative algorithm for using maximum likelihood to estimate the parameters of a statistical model with unobserved (hidden)variables. It has two main steps. First is the E-step.We compute some probability distribution of the model so we can use it for expectations.Second comes the M-step, which stands for maximization. In this step, we maximize the lower bound of the log-likelihood function by generating a new set of parameters with respect to the expectations.

Step 1 :- Find the Conditions for which maximum likelihood can be reached. So for that recall the older maximum likelihood function which is in form of sum of logs and finding optimum parameters from that function is hard. So try to reduce derivative of that function in terms of our new introduced variable

$$\ln p(\mathbf{X} \mid \pi, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \right\}$$

First Taking Derivative w.r.t to µk.

$$0 = -\sum_{n=1}^{N} \frac{\pi_k \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\sum_{\boldsymbol{j}} \boldsymbol{\pi_j} \boldsymbol{\mathcal{N}}\left(\mathbf{x_n} \mid \boldsymbol{\mu_j}, \boldsymbol{\Sigma_j}\right)} \boldsymbol{\Sigma}_k \left(\mathbf{x}_n - \boldsymbol{\mu}_k\right)$$

As we defined (znk) before, it can be used to reduce the term as shown And after simplifying it, µk can be estimated.

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) \mathbf{x}_n$$

where

$$N_k = \sum_{n=1}^{N} \gamma\left(z_{nk}\right)$$

Taking Derivative w.r.t to $\sum_k$ we will get

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_k\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_k\right)^{\mathrm{T}}$$

Taking Derivative w.r.t to π_k
Here we must take account of the constraint $\sum \pi_k = 1$ Can be achieved using a Lagrange multiplier

$$\ln p(\mathbf{X} \mid \pi, \boldsymbol{\mu}, \boldsymbol{\Sigma}) + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right)$$

$$0 = \sum_{n=1}^{N} \frac{\mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \pi_j \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)} + \lambda$$

Which Gives

$$\pi_k = \frac{N_k}{N}$$

EM Algorithm For GMM

Given a Gaussian mixture model,maximize the likelihood function w.r.t parameters

- Step 1 : Initialize μ_k , σk and πk and find the initial value of log likelihood.
- Step 2 : E step : Evaluate the (Znk) using the current parameter values.

$$\gamma\left(z_{nk}\right) = \frac{\pi_k \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)}$$

- step 3 : M step : Re-estimate the parameters from $\gamma(Z_{nk})$

$$\begin{aligned}
\boldsymbol{\mu}_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right) \mathbf{x}_n \\
\boldsymbol{\Sigma}_k^{\text{new}} &= \frac{1}{N_k} \sum_{n=1}^{N} \gamma\left(z_{nk}\right)\left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right)\left(\mathbf{x}_n - \boldsymbol{\mu}_k^{\text{new}}\right)^{\mathrm{T}} \\
\pi_k^{\text{new}} &= \frac{N_k}{N}
\end{aligned}$$

- Step 4 : Evaluate the log likelihood

$$\ln p(\mathbf{X} \mid \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right) \right\}$$

- Step 5 : Check for convergence of either the parameters or the log likelihood.the convergence criterion is not satisfied return to step 2.

- Results of EM
  - EM Gives soft Assignments
  - All points contribute to estimate all components
  - Each point has unit weight to contribute, but splits it across the K components
  - Weight contributed by point to component is proportional to the likelihood that point was generated by that component.

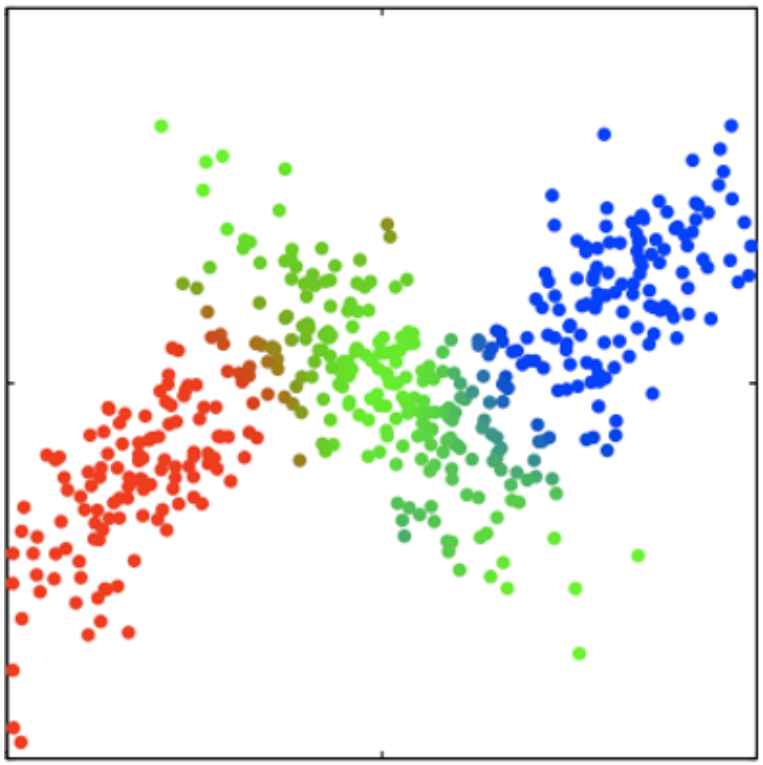

Figure 3: EM Clustering

References

[1] Andrew NG's machine learning course. Lectures on Unsupervised Learning, k-means clustering, Mixture of Gaussians and The EM Algorithm http://cs229.stanford.edu/materials.html

[2] Arthur Dempster, Nan Laird, and Donald Rubin (1977) Maximum Likelihood from Incomplete Data via the EM https://www.jstor.org/stable/2984875.

[3] Ramesh Sridharan's Gaussian mixture models and the EM algorithm https://people.csail.mit.edu/rameshvs/content/gmm-em.pdf

[4] https://towardsdatascience.com/