A Tutorial of the EM-algorithm and its Application to Outlier Detection

Jaehyeong Ahn

Konkuk University

jayahn0104@gmail.com

September 9, 2020

Table of Contents

EM-algorithm: An Overview

EM-algorithm: An Overview

Introduction

- The EM-algorithm (Expectation-Maximization algorithm) is an iterative procedure for computing the maximum likelihood estimator (MLE) when only a subset of the data is available (When the model depends on the unobserved latent variable)
- The first proper theoretical study of the algorithm was done by Dempster, Laird, and Rubin (1977) [1]
- The EM-algorithm is widely used in various research areas when unobserved latent variables are included in the model

EM-algorithm: An Overview

Data

- $Y = (y_1, \cdots, y_N)^T$: Observed data

Model

- Assume that Y is dependent on some unobserved latent variable Z where $Z = (z_1, \cdots, z_N)^T$
  - When Z is assumed as discrete random variable
    - $r_{ik} = P(z_i = k|Y, \theta), \quad k = 1, \cdots, K$
    - $z_i^* = \underset{k}{\operatorname{argmax}}\, r_{ik}$
  - When Z is assumed as continuous random variable
    - $r_i = f_{Z|Y,\Theta}(z_i|Y, \theta)$

Log-likelihood

- $\ell_{obs}(\theta; Y) = \log f_{Y|\Theta}(Y|\theta) = log \int f_{Y,Z|\Theta}(Y, Z|\theta)\, dz$

EM-algorithm: An Overview

Goal

- By maximizing $\ell_{obs}(\theta; Y)$ w.r.t. θ
  - Find $\hat{\theta}$ which satisfies $\partial_{\theta_j} \ell_{obs}(\theta; Y)|_{\theta=\hat{\theta}}$ =0, *for* $j = 1, \cdots, J$
  - Compute the estimated value $\hat{r}_{ik} = f_{Z|Y,}\Theta(z_i|Y, \hat{\theta})$

Problem

- The latent variable Z is not observable
  - It is difficult to compute the integral in $\ell_{obs}(\theta; Y)$
  - Thus the parameters can not be estimated separately

Solution

- Assume the latent variable Z is observed
  - Define the complete Data
  - Maximize the complete log likelihood

Data

- $Y = (y_1, \cdots, y_N)^T$: Observed data
- $Z = (z_1, \cdots, z_N)^T$: Unobserved (latent) variable
    - It is assumed as observed
- $X = (Y, Z)$: Complete Data

Model

- $r_i = f_{Z|Y,\Theta}(z_i|Y, \theta)$

Complete log-likelihood

- $\ell_C(\theta; X) = \log f_{X|\Theta}(X|\theta) = \log f_{X|\Theta}(Y, Z|\theta)$

Log-likelihood for observed data

- $\ell_{obs}(\theta; Y) = \log f_{Y|\Theta}(Y|\theta) = \log \int f_{X|\Theta}(Y, Z|\theta)\, dz$

EM-algorithm: An Overview

Estimation Idea

- Maximize $\ell_C(\theta; X)$ instead of maximizing $\ell_{obs}(\theta; Y)$
  - Since the parameters in $\ell_C(\theta; X)$ can be decoupled
- E-step
  - Compute the Expected Complete Log Likelihood ($ECLL$)
    - Taking conditional expectation on $\ell_C(\theta; X)$ given Y and current value of parameters $\theta^{(s)}$
    - This step estimates the realizations of z (since the value of z is not identified)
- M-step
  - Maximize the computed $ECLL$ w.r.t. θ
  - Update the estimates of Θ by $\theta^{(s+1)}$ which maximize the current $ECLL$
- Iterate this procedure until it converges

EM-algorithm: An Overview

Estimation

E-step

- Taking conditional expectation on $\ell_C(\theta; X)$ given Y and $\theta = \theta^{(s)}$
  - For $\theta^{(0)}$, set initial guess
- Compute $Q(\theta|\theta^{(s)}) = E_{\theta^{(s)}}\left[\ell_C(\theta; X)|Y\right]$

M-step

- Maximize $Q(\theta|\theta^{(s)})$ w.r.t. θ
- Put $\theta^{(s+1)} = \underset{\theta}{\operatorname{argmax}}\, Q(\theta|\theta^{(s)})$

- Iterate until it satisfies following inequality
  - $||\theta^{(s+1)} - \theta^{(s)}|| < \epsilon$, where ϵ denotes the sufficiently small value

Immediate Question

- Does maximizing the sequence $Q(\theta|\theta^{(s)})$ leads to maximizing $\ell_{obs}(\theta;Y)$?

- This question will be answered in following slides (Proof for EM-algorithm) with 3 parts:

Non-decreasing / Convergence / Local maximum

Proof for EM-algorithm

Non-decreasing (Ascent Property)

Proposition 1.

The Sequence $\ell_{obs}(\theta^{(s)}; Y)$ in the EM-algorithm is non-decreasing

- Proof
  - We write $X = (Y, Z)$ for the complete data
  - Then

$$f_{Z|Y,\Theta}(Z|Y,\theta) = \frac{f_{X|\Theta}(Y,Z|\theta)}{f_{Y|\Theta}(Y|\theta)}$$

  - Hence,

$$\ell_{obs}(\theta; Y) = \log f_{Y|\Theta}(Y|\theta) = \log f_{X|\Theta}((Y,Z)|\theta) - \log f_{Z|Y,\Theta}(Z|Y,\theta)$$

Proof for EM-algorithm: Non-decreasing

- Taking conditional expectation given Y and $\varTheta = \theta^{(s)}$ on both sides yields

$$\begin{aligned}
\ell_{obs}(\theta; Y) &= E_{\theta^{(s)}}[\ell_{obs}(\theta; Y)|Y] \\
&= E_{\theta^{(s)}}[\log f_{X|\varTheta}((Y, Z)|\theta)|Y] - E_{\theta^{(s)}}[\log f_{Z|Y,\varTheta}(Z|Y, \theta)|Y] \\
&= Q(\theta|\theta^{(s)}) - H(\theta|\theta^{(s)})
\end{aligned}$$

- Where

$$Q(\theta|\theta^{(s)}) = E_{\theta^{(s)}}[\log f_{X|\varTheta}((Y, Z)|\theta)|Y]$$

$$H(\theta|\theta^{(s)}) = E_{\theta^{(s)}}[\log f_{Z|Y,\varTheta}(Z|Y, \theta)|Y]$$

- Then we have

$$\begin{aligned}
\ell_{obs}(\theta^{(s+1)}; Y) - \ell_{obs}(\theta^{(s)}; Y) = Q(\theta^{(s+1)}|\theta^{(s)}) - Q(\theta^{(s)}|\theta^{(s)}) \\
- \left[H(\theta^{(s+1)}|\theta^{(s)}) - H(\theta^{(s)}|\theta^{(s)})\right]
\end{aligned}$$

Proof for EM-algorithm: Non-decreasing

- Recall that

$$\ell_{obs}(\theta^{(s+1)}; Y) - \ell_{obs}(\theta^{(s)}; Y) = \underbrace{Q(\theta^{(s+1)}|\theta^{(s)}) - Q(\theta^{(s)}|\theta^{(s)})}_{(I)} - \underbrace{\left[H(\theta^{(s+1)}|\theta^{(s)}) - H(\theta^{(s)}|\theta^{(s)})\right]}_{(II)}$$

- (I) is non-negative
  - $\theta^{(s+1)} = \underset{\theta}{\operatorname{argmax}}\ Q(\theta|\theta^{(s)})$
  - Hence $Q(\theta^{(s+1)}|\theta^{(s)}) \geq Q(\theta^{(s)}|\theta^{(s)})$
  - Thus (I) ≥ 0

Proof for EM-algorithm: Non-decreasing

- (II) is non-positive
  - Using Jensen's inequality for concave functions (log is concave)

Theroem 1. Jensen's inequality

Let f be a concave function, and let X be a random variable. Then

$$E[f(X)] \leq f(EX)$$

$$
\begin{aligned}
H(\theta^{(s+1)}|\theta^{(s)}) - H(\theta^{(s)}|\theta^{(s)}) &= E_{\theta^{(s)}}\left[\log\left(\frac{f_{Z|Y,\Theta}(Z|Y,\theta^{(s+1)})}{f_{Z|Y,\Theta}(Z|Y,\theta^{(s)})}\right)|Y\right] \\
&\leq \log E_{\theta^{(s)}}\left[\left(\frac{f_{Z|Y,\Theta}(Z|Y,\theta^{(s+1)})}{f_{Z|Y,\Theta}(Z|Y,\theta^{(s)})}\right)|Y\right] \\
&= \log\int\frac{f_{Z|Y,\Theta}(z|Y,\theta^{(s+1)})}{f_{Z|Y,\Theta}(z|Y,\theta^{(s)})}f_{Z|Y,\Theta}(z|Y,\theta^{(s)})\,dz \\
&= \log 1 = 0
\end{aligned}
$$

- Hence $H(\theta^{(s+1)}|\theta^{(s)}) \leq H(\theta^{(s)}|\theta^{(s)})$

Theroem 1. Jensen's inequality

Let f be a concave function, and let X be a random variable. Then

$$E[f(X)] \leq f(EX)$$

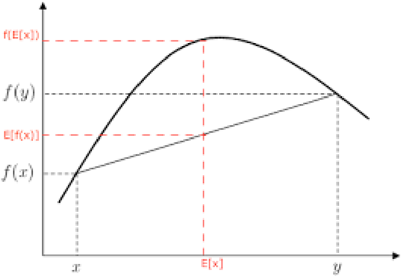


Figure: Jensen's inequality for concave function[3]

Proof for EM-algorithm: Non-decreasing

- Recall that

$$\ell_{obs}(\theta^{(s+1)};Y) - \ell_{obs}(\theta^{(s)};Y) = \underbrace{Q(\theta^{(s+1)}|\theta^{(s)}) - Q(\theta^{(s)}|\theta^{(s)})}_{(I)} - \underbrace{\left[H(\theta^{(s+1)}|\theta^{(s)}) - H(\theta^{(s)}|\theta^{(s)})\right]}_{(II)}$$

- We've proven that (I) ≥ 0 and (II) ≤ 0
- This shows $\ell_{obs}(\theta^{(s+1)};Y) - \ell_{obs}(\theta^{(s)};Y) \geq 0$
- Thus the sequence $\ell_{obs}(\theta^{(s)};Y)$ in the EM-algorithm is **non-decreasing**

Proof for EM-algorithm: Convergence

Convergence

- We will show that the sequence $\theta^{(s)}$ converges to some θ^* with $\ell(\theta^*; y) = \ell^*$, the limit of $\ell(\theta^{(s)})$

Assumption

- Ω is a subset of $\mathbb{R}^k$
- $\Omega_{\theta_o} = \{\theta \in \Omega : \ell(\theta; y) \geq \ell(\theta_o; y)\}$ is compact for any $\ell(\theta_o; y) > -\infty$
- $\ell(\theta_o; x)$ is continuous and differentiable in the interior of Ω

Proof for EM-algorithm: Convergence

Theorem 2

Suppose that $Q(\theta|\phi)$ is continuous in both θ and ϕ. Then all $limit\ points$ of any instance $\{\theta^{(s)}\}$ of the EM algorithm are $stationary\ points$, i.e. $\theta^* = \underset{\theta}{\operatorname{argmax}}\, Q(\theta|\theta^*)$, and $\ell(\theta^{(s)}; y)$ converges monotonically to some value $\ell^* = \ell(\theta^*; y)$ for some $stationary\ point\ \theta^*$

Theorem 3

Assume the hypothesis of Theorem 2. Suppose in addition that $\partial_\theta Q(\theta|\phi)$ is continuous in θ and ϕ. Then $\theta^{(s)}$ converges to a stationary point θ^* with $\ell(\theta^*; y) = \ell^*$, the limit of $\ell(\theta^{(s)})$, if either

- $\{\theta : \ell(\theta; y) = \ell^*\} = \{\theta^*\}$ or
- $|\theta^{(s+1)} - \theta^{(s)}| \to 0$ and $\{\theta : \ell(\theta; y) = \ell^*\}$ is discrete

Proof for EM-algorithm: Local Maximum

Local Maximum

- Recall that

$$\ell_C(\theta; X) = \log f_{X|\Theta}(X|\theta) = \log f_{Z|Y,\Theta}(Z|Y,\theta) + \log f_{Y|\Theta}(Y|\theta)$$

- Then

$$Q(\theta|\theta^{(s)}) = \int \log f_{Z|Y,\Theta}(z|Y,\theta) f_{Z|Y,\Theta}(z|Y,\theta^{(s)})\, dz + \ell_{obs}(\theta; Y)$$

- Differentiating w.r.t. θ_j and putting equal to 0 in order to maximize Q gives

$$0 = \partial_{\theta_j} Q(\theta|\theta^{(s)}) = \int \frac{\partial_{\theta_j} f_{Z|Y,\Theta}(z|Y,\theta)}{f_{Z|Y,\Theta}(z|Y,\theta)} f_{Z|Y,\Theta}(z|Y,\theta^{(s)})\, dz + \partial_{\theta_j} \ell_{obs}(\theta, Y)$$

Proof for EM-algorithm: Local Maximum

- Recall that

$$0 = \partial_{\theta_j} Q(\theta|\theta^{(s)}) = \int \frac{\partial_{\theta_j} f_{Z|Y,\Theta}(z|Y,\theta)}{f_{Z|Y,\Theta}(z|Y,\theta)} f_{Z|Y,\Theta}(z|Y,\theta^{(s)})\, dz + \partial_{\theta_j} \ell_{obs}(\theta, Y)$$

- If $\theta^{(s)} \to \theta^*$ then we have for θ^* that (with $j = 1, \cdots, J$)

$$\begin{aligned}
0 &= \partial_{\theta_j} Q(\theta^*|\theta^*) \\
&= \int \frac{\partial_{\theta_j} f_{Z|Y,\Theta}(z|Y,\theta^*)}{f_{Z|Y,\Theta}(z|Y,\theta^*)} f_{Z|Y,\Theta}(z|Y,\theta^*)\, dz + \partial_{\theta_j} \ell_{obs}(\theta^*; Y) \\
&= \partial_{\theta_j} \int f_{Z|Y,\Theta}(z|Y,\theta^*)\, dz + \partial_{\theta_j} \ell_{obs}(\theta^*; Y) \\
&= \partial_{\theta_j} \ell_{obs}(\theta^*; Y)
\end{aligned}$$

An Example: Gaussian Mixture Model (GMM)

Introduction

- Mixture models make use of latent variables to model different parameters for different groups (or **clusters**) of data points
- For a point y_i, let the cluster to which that point belongs be labeled z_i; where z_i is latent, or unobserved
- In this example, we will assume our observable features $\boldsymbol{y}_i$ to be distributed as a Gaussian, chosen based on the cluster that point $\boldsymbol{y}_i$ is associated with

An Example: Gaussian Mixture Model (GMM)

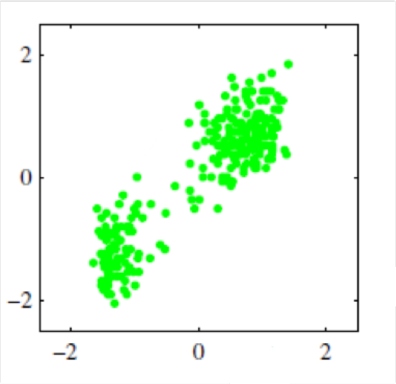


Figure: Gaussian Mixture Model Example[5]

An Example: Gaussian Mixture Model (GMM)

Data

- $Y = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)^T$: Observed data
  - $\forall i \ \boldsymbol{y}_i \in \mathbb{R}^p$
- $Z = (z_1, \cdots, z_N)^T$: Unobserved (latent) variable
  - Assume that Z is observed
  - $\forall i \ z_i \in \{1, 2, \cdots, K\}$
- $X = (Y, Z)$: Complete data

An Example: Gaussian Mixture Model (GMM)

Distribution Assumption

$$z_i \;\sim\; Mult(\boldsymbol{\pi}),\; \boldsymbol{\pi} \in \mathbb{R}^K$$

$$\boldsymbol{y}_i | z_i = k \;\sim\; \mathcal{N}_P(\mu_k, \Sigma_k)$$

Model

$$r_{ik} \overset{\text{def}}{=} p(z_i = k | \boldsymbol{y}_i, \theta) = \frac{p(\boldsymbol{y}_i | z_i = k, \theta) p(z_i = k | \theta)}{\sum_{k=1}^{K} \left(p(\boldsymbol{y}_i | z_i = k, \theta) p(z_i = k | \theta) \right)}$$

$$z_i^* = \underset{k}{\operatorname{argmax}}\; r_{ik}$$

- θ denotes the general parameter; $\theta = \{\pi, \mu, \Sigma\}$

An Example: Gaussian Mixture Model (GMM)

Notation Simplification

- Write $z_i = k$ as

$$\boldsymbol{z}_i = (z_{i1}, \cdots, z_{ik}, \cdots, z_{iK})^T = (0, \cdots, 1, \cdots, 0)^T$$

  - Where $z_{ij} = I(j = k) \in \{0, 1\}$ for $j = 1, \cdots, K$

- Using this:

$$p(\boldsymbol{z}_i | \pi) = \prod_{k=1}^{K} \pi_k^{\boldsymbol{z}_i^k}$$

$$p(\boldsymbol{y}_i | \boldsymbol{z}_i, \theta) = \prod_{k=1}^{K} \mathcal{N}(\mu_k, \Sigma_k)^{\boldsymbol{z}_i^k} = \prod_{k=1}^{K} \phi(\boldsymbol{y}_i | \mu_{k,k})^{\boldsymbol{z}_i^k}$$

Log-likelihood for observed data

$$
\begin{aligned}
\ell_{obs}(\theta; Y) &= \log\, f_{Y|\Theta}(Y|\theta) \\
&= \sum_{i=1}^{N} \log\, p(\boldsymbol{y}_i|\theta) \\
&= \sum_{i=1}^{N} \log\, \left[\sum_{\boldsymbol{z}_i}^{K} p(\boldsymbol{y}_i, \boldsymbol{z}_i|\theta) \right] \\
&= \sum_{i=1}^{N} \log\, \left[\sum_{\boldsymbol{z}_i \in Z} \prod_{k=1}^{K} \pi_k^{\boldsymbol{z}_i^k} \mathcal{N}(\mu_k, \Sigma_k)^{\boldsymbol{z}_i^k} \right]
\end{aligned}
$$

- This does not decouple the likelihood because the log cannot be 'pushed' inside the summation

Complete log-likelihood

$$
\begin{aligned}
\ell_C(\theta; X) &= \log\ f_{X|\Theta}(Y, Z|\theta) \\
&= \log\ f_{Y|Z,\Theta}(Y|Z, \theta) + \log\ f_{Z|\Theta}(Z|\theta) \\
&= \sum_{i=1}^{N} \left[\log \prod_{k=1}^{K} \phi(\boldsymbol{y}_i|\mu_k, \Sigma_k)^{\boldsymbol{z}_i^k} + \log \prod_{k=1} \pi_k^{\boldsymbol{z}_i^k} \right] \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} \left[\boldsymbol{z}_i^k \log\ \phi(\boldsymbol{y}_i|\mu_k, \Sigma_k) + \boldsymbol{z}_i^k \log\ \pi_k \right]
\end{aligned}
$$

- Parameters are now decoupled since we can estimate π_k and μ_k , Σ_k separately

An Example: Gaussian Mixture Model (GMM)

Estimation

- E-step

$$Q(\theta|\theta^{(s)}) = E_{\theta^{(s)}}\left[\sum_{i=1}^{N}\sum_{k=1}^{K} \boldsymbol{z}_i^k \log \phi(\boldsymbol{y}_i|\mu_k, \Sigma_k) + \boldsymbol{z}_i^k \log \pi_k \middle| Y\right]$$

$$= \sum_{i=1}^{N}\sum_{k=1}^{K}\left[E_{\theta^{(s)}}\left[\boldsymbol{z}_i^k|Y\right] \log \phi(\boldsymbol{y}_i|\mu_k, \Sigma_k) + E_{\theta^{(s)}}\left[\boldsymbol{z}_i^k|Y\right] \log \pi_k\right]$$

An Example: Gaussian Mixture Model (GMM)

- E-step
  - Note that $\boldsymbol{z}_i^k = 1|Y \sim Berrnoulli\left(p(\boldsymbol{z}_i^k = 1|Y, \theta)\right)$
  - Hence

$$
\begin{aligned}
r_{ik}^{(s)} \overset{\text{def}}{=} E_{\theta^{(s)}}\left[\boldsymbol{z}_i^k|Y\right] &= p(\boldsymbol{z}_i^k = 1|Y, \theta^{(s)}) \\
&= \frac{p(\boldsymbol{z}_i^k = 1, \boldsymbol{y}_i|\theta^{(s)})}{\sum_{k=1}^K p(\boldsymbol{z}_i^k = 1, \boldsymbol{y}_i|\theta^{(s)})} \\
&= \frac{p(\boldsymbol{y}_i|\boldsymbol{z}_i^k = 1, \theta^{(s)})\, p(\boldsymbol{z}_i^k = 1|\theta^{(s)})}{\sum_{k=1}^K p(\boldsymbol{y}_i|\boldsymbol{z}_i^k = 1, \theta^{(s)})\, p(\boldsymbol{z}_i^k = 1|\theta^{(s)})} \\
&= \frac{\phi(\boldsymbol{y}_i|\mu_k^{(s)}, \Sigma_k^{(s)})\, \pi_k^{(s)}}{\sum_{k=1}^K \phi(\boldsymbol{y}_i|\mu_k^{(s)}, \Sigma_k^{(s)})\, \pi_k^{(s)}}
\end{aligned}
$$

An Example: Gaussian Mixture Model (GMM)

- M-step
  - Recall that

$$Q(\theta|\theta^{(s)}) = \sum_{i=1}^{N}\sum_{k=1}^{K}\left[r_{ik}^{(s)} \log\, \phi(\boldsymbol{y}_i|\mu_k, \Sigma_k) + r_{ik}^{(s)} \log\, \pi_k\right]$$

  - Set $\theta^{(s+1)} = \underset{\theta}{\operatorname{argmax}}\; Q(\theta|\theta^{(s)})$
    - $\pi_k^{(s+1)} = \frac{\sum_{i=1}^{N} r_{ik}^{(s)}}{N}$
    - $\mu_k^{(s+1)} = \frac{\sum_{i=1}^{N} r_{ik}^{(s)}\boldsymbol{y}_i}{\sum_{i=1}^{N} r_{ik}^{(s)}}$
    - $\Sigma_k^{(s+1)} = \frac{\sum_{i=1}^{N} r_{ik}^{(s)}\left(\boldsymbol{y}_i - \mu_k^{(s+1)}\right)\left(\boldsymbol{y}_i - \mu_k^{(s+1)}\right)^T}{\sum_{i=1}^{N} r_{ik}^{(s)}}$

An Example: Gaussian Mixture Model (GMM)

- Iterate until it satisfies following inequality
  - $||\theta^{(s+1)} - \theta^{(s)}|| < \epsilon$
    - Let $\hat{\theta} = \theta^{(s)}$

What we get

- $\hat{\theta} = (\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\mu}}, \hat{\Sigma})$
- $\hat{r}_{ik} = p(z_i = k | Y, \hat{\theta}) = \frac{\phi(\boldsymbol{y}_i | \hat{\mu}_k, \hat{\Sigma}_k)\ \hat{\pi}_k}{\sum_{k=1}^{K} \phi(\boldsymbol{y}_i | \hat{\mu}_k, \hat{\Sigma}_k)\ \hat{\pi}_k}$
- $\hat{z}_i = \underset{k}{\operatorname{argmax}}\ \hat{r}_{ik}$

An Example: Gaussian Mixture Model (GMM)

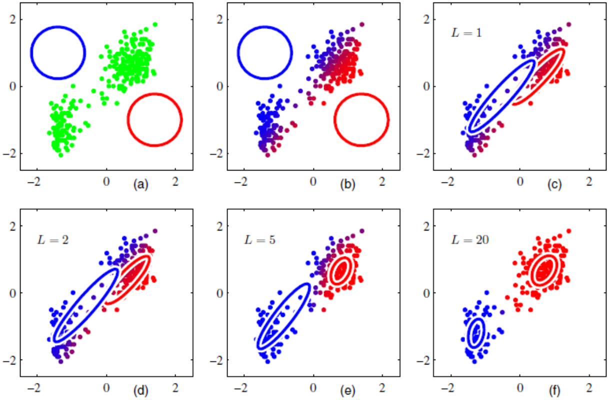


Figure: Gaussian Mixture Model Fitting Example[5]

Application to Outlier Detection

Application to Outlier Detection: Directly Used

Basic Idea

- For cases in which the data may have many different clusters with different orientations
- Assume a specific form of the generative model (e.g., a mixture of Gaussians)
- Fit the model to the data (usually for normal behavior)
  - Estimate the parameters with EM-algorithm
- Fit this model to the unseen (test) data and get the estimation of the fit (joint) probabilities
  - Data points that fit the distribution will have high fit (joint) probabilities
  - Whereas anomalies (outliers) will have very low fit (joint) probabilities

Application to Outlier Detection: Directly Used

Simulation in R

https://rpubs.com/JayAhn/650433

Application to Outlier Detection: Indirectly Used

Introduction

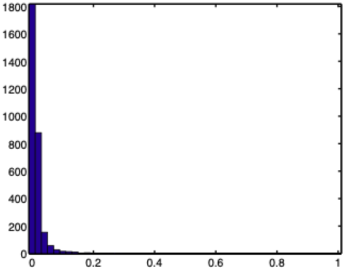
- Interestingly, EM-algorithms can also be used as a final step after many such outlier detection algorithms for converting the scores into probabilities [7]
- Converting the outlier scores into well-calibrated probability estimates is more favorable for several reasons
  1. The probability estimates allow us to select the appropriate threshold for declaring outliers using a Bayesian risk model
  2. The probability estimates obtained from individual models can be aggregated to build an ensemble outlier detection framework

Application to Outlier Detection: Indirectly Used

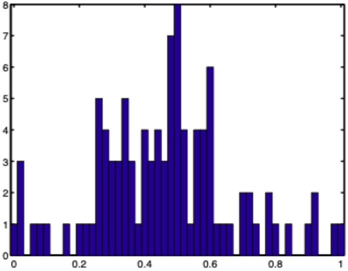
Motivation

- Since the outlier detection problem is mainly about unsupervised learning environment, it is hard to select the appropriate threshold for decalring outliers
- Every outlier detection model outputs different outlier score with different scale which leads to a difficult problem during constructing an outlier ensemble model

Outlier Score Distributions



(a) Outlier Score Distribution for Normal Examples



(b) Outlier Score Distributions for Outliers

Figure: Outlier Score Distributions [7]

Application to Outlier Detection: Indirectly Used

Basic Idea

- Treat an outlier score as an univariate random variable
- Assume the label of outlierness as an unobserved latent variable
- Estimate the posterior probabilities for the latent variable with EM-algorithm
  1. Model the posterior probability for outlier scores using a sigmoid function
  2. Model the score distribution as a mixture model (mixture of exponential and Gaussian) and calculate the posterior probabilities via the Bayes' rule

Bayesian Risk Model

- Bayesian risk model minimizes the overall risk associated with some cost function
- For example, in the case of a two-class problem
    - The Bayes decision rule for a given observation x is to decide ω_1 if:

$$(\lambda_{21} - \lambda_{11})P(\omega_1|x) > (\lambda_{12} - \lambda_{22})P(\omega_2|x)$$

    - Where ω_1, ω_2 are the two classes while λ_{ij} is the cost of misclassifying ω_j as ω_i
    - Since $P(\omega_2|x) = 1 - P(\omega_1|x)$, the preceding inequality suggests that the appropriate outlier threshold is automatically determined once the cost functions are known
    - In the case of a zero-one loss function, the threshold which minimizes the overall risk is 0.5, where

$$\lambda_{ij} = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{if } i \neq j \end{cases}$$

Summary

Summary

- This slide had an overview on EM-algorithm and its application to Outlier detection
- We've checked the basic procedure of the EM-algorithm for estimating the parameters of model which has the unobserved latent variable
- This slide also has shown that the log-likelihood for observed data is maximized by EM-algorithm through 3 parts: Non-decreasing / Convergence / Local Maximum
- Further more, we've seen that the EM-algorithm can be applied for the Outlier detection not only directly but also indirectly

References

- [1] Dempster, Arthur P., Nan M. Laird, and Donald B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm." Journal of the Royal Statistical Society: Series B (Methodological) 39, no. 1 (1977): 1-22.
- [2] https://www.math.kth.se/matstat/gru/Statistical%20inference/Lecture8.pdf
- [3] https://www.cs.cmu.edu/∼epxing/Class/10708-17/notes-17/10708-scribe-lecture8.pdf
- [4] http://www2.stat.duke.edu/∼sayan/Sta613/2018/lec/emnotes.pdf
- [5] Contributions to collaborative clustering and its potential applications on very high resolution satellite images

References

- [6] Kriegel, Hans-Peter, Peer Kroger, Erich Schubert, and Arthur Zimek. "Interpreting and unifying outlier scores." In Proceedings of the 2011 SIAM International Conference on Data Mining, pp. 13-24. Society for Industrial and Applied Mathematics, 2011.
- [7] Gao, Jing, and Pang-Ning Tan. "Converting output scores from outlier detection algorithms into probability estimates." In Sixth International Conference on Data Mining (ICDM'06), pp. 212-221. IEEE, 2006.

Q&A