Практическая реализация EM-алгоритма для смеси гауссовских распределений

Г.А. Ситкарев, <sitkarev@unixkomi.ru>

Сыктывкарский Государственный Университет

1. Условия задачи

Дано N точек $(\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N)$ размерности M. Известно, что все точки принадлежат K многомерным гауссовским распределениям вида $\omega_k\ \eta(\mathbf{x} \mid \boldsymbol{\mu}_k, \Sigma_k)$ с неизвестными параметрами $\boldsymbol{\mu}_k$, Σ_k, ω_k:

$\boldsymbol{\mu}_k$ вектор средних значений для k-го распределения;

Σ_k ковариационная матрица $M \times M$ для k-го распределения;

ω_k вес k-го распределения, $\sum\limits_k \omega_k = 1$.

Каждая точка $\mathbf{x}_n$ имеет некоторую вероятность присутствия в распределении k. Эту вероятность мы будем обозначать как P_{nk}. Все параметры P_{nk} для всех точек удобно свести в матрицу $P_{N\times K}$:

$$P_{N\times K} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & \cdots & P_{1K} \\ P_{21} & P_{22} & P_{23} & \cdots & P_{2K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{N1} & P_{N2} & P_{N3} & \cdots & P_{NK} \end{bmatrix}.$$

2. Правдоподобность и вероятности P_{nk}

Вероятностную плотность для точки с координатами $\mathbf{x}$ будем обозначать как $P(\mathbf{x})$. Это вероятность попасть в точку по этим координатам, если случайным образом выбирать её из всей выборки. Правдоподобность параметров распределения будем обозначать как $\mathscr{L}$. Наша задача максимизировать $\mathscr{L}$, подобрав параметры $\boldsymbol{\mu}_k$, Σ_k, ω_k. Так как мы полагаем точки с данными независимыми, их $\mathscr{L}$ будет произведением вероятностей присутствия точки из выборки в $\mathbf{x}_n$:

$$\mathscr{L} = \prod_{n=1}^{N} P(\mathbf{x}_n) = P(\mathbf{x}_1) \times P(\mathbf{x}_2) \times \cdots \times P(\mathbf{x}_N).$$

Для того, чтобы посчитать $\mathscr{L}$, нам нужно найти плотности распределения в точках $\mathbf{x}_n$ и перемножить их. Для каждой точки $\mathbf{x}_n$ плотность определяется как сумма плотностей из K распределений, пропорционально весу k-го распределения:

$$P(\mathbf{x}_n) = \sum_{k=1}^{K} \omega_k\ \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \Sigma_k).$$

Так как значения P_{nk}, то есть значения строки n матрицы $P_{N\times K}$, есть вероятности вхождения $\mathbf{x}_n$ в распределение k, их сумма должна равняться единице:

$$\sum_{k=1}^{k} P_{nk} = 1.$$

Значит $P(\mathbf{x}_n)$ это сумма

$$P(\mathbf{x}_n) = P_{n1} \cdot P(\mathbf{x}_n) + P_{n2} \cdot P(\mathbf{x}_n) + \cdots + P_{NK} \cdot P(\mathbf{x}_n),$$

но нам также известно, что

$$P(\mathbf{x}_n) = \omega_1\, \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_1, \Sigma_1) + \omega_2\, \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_2, \Sigma_2) + \cdots + \omega_K\, \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_K, \Sigma_K).$$

Последнее означает, что каждое значение P_{nk} можно вычислить по формуле:

$$P_{nk} = \frac{\omega_k\, \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \Sigma_k)}{P(\mathbf{x}_n)}.$$

При заполнении матрицы $P_{N\times K}$ значениями P_{nk}, вычисления рационально построить по следующей схеме:

Для каждой строки n матрицы $P_{N\times K}$:

а) Заполнить строку значениями $P_{nk} \cdot P(\mathbf{x}_n) = \omega_k\, \eta(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \Sigma_k)$.

б) Подсчитать сумму всех элементов в строке $\sum P_{nk}$.

в) Поделить все элементы строки на сумму, полученную на предыдущем шаге.

3. Формулы максимизации

Значения $\bar{\boldsymbol{\mu}}_k$, $\bar{\Sigma}_k$, $\bar{\omega}_k$ на шаге максимизации вычисляются по следующим формулам:

$$\bar{\omega}_k = \frac{1}{N} \sum_n P_{nk} \tag{1}$$

$$\bar{\boldsymbol{\mu}}_k = \frac{\sum_n P_{nk} \cdot \mathbf{x}_n}{\sum_n P_{nk}} \tag{2}$$

$$\bar{\Sigma}_k = \frac{\sum_n P_{nk} \cdot (\mathbf{x}_n - \bar{\boldsymbol{\mu}}_k)(\mathbf{x}_n - \bar{\boldsymbol{\mu}}_k)^T}{\sum_n P_{nk}} \tag{3}$$

4. Пошаговое выполнение алгоритма

Пользователь предоставляет данные $\mathbf{x}_n$, размерности N, M и K, значение ε и, возможно, вектор начальных значений для $\boldsymbol{\mu}_k$. Реализация алгоритма может состоять из следующих шагов:

1. На первом шаге:
   а) Установить веса ω_k, как $\omega_k = \frac{1}{k}$.
   б) Установить Σ_k в $I = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$.
   в) Выбрать $\boldsymbol{\mu}_k$, как K случайно взятых значения из набора $\mathbf{x}_n$, или же взять начальные значения, предоставленные пользователем.
2. На втором шаге:
   а) Выполнить шаг *E*:
      а.1) вычислить P_{nk} и $P(\mathbf{x}_n)$;
      а.2) вычислить сумму всех логарифмов $P(\mathbf{x}_n)$;
      а.3) сохранить $\sum_n \log P(\mathbf{x}_n)$ в переменную *loglike_prev*.
   б) Выполнить шаг *M*:
      б.1) пользуясь формулой (1) и весами $P_{N\times K}$, вычислить $\bar{\omega}_k$;
      б.2) пользуясь формулой (2) и весами $P_{N\times K}$, вычислить $\bar{\boldsymbol{\mu}}_k$;
      б.3) пользуясь формулой (3) и весами $P_{N\times K}$, вычислить $\bar{\Sigma}_k$.

3. На третьем шаге:
   а) Выполнить шаг *E*, как в 2а, но сохраняя результат 2а.3 в переменную *loglike*.
   б) Вычислить $abs(loglike - loglike\_prev)$, и сравнить с ε, заданным пользователем. Если $abs(loglike - loglike\_prev) < \varepsilon$, тогда перейти на шаг 4.
   в) Выполнить шаг *M*, как в 2б.
   г) Перейти на шаг 3.
4. Завершить выполнение алгоритма с успешным статусом, а значения $\bar{\boldsymbol{\mu}}_k$, $\bar{\Sigma}_k$, $\bar{\omega}_k$ вернуть пользователю.