Expectation-Maximization

Tom Shen

February 7, 2013

1 Current Implementation

Taken from p. 193 of Tom Mitchell's book

1.1 Expectation

$$E[z_{ij}] = \frac{p(x = x_i|\mu = \mu_j)}{\sum_{n=1}^{k} p(x = x_i|\mu = \mu_n)}$$

where k is the number of distributions.

1.2 Maximization

$$\mu_j = \frac{\sum_{i=1}^{m} E[z_{ij}]x_i}{\sum_{i=1}^{m} E[z_{ij}]}$$

where m is the number of data points x_i.

1.3 Progress so far

I have code in both Python and Java to generate points for multidimensional k gaussians, with means chosen randomly between given upper and lower bounds, and a fixed standard devation.

I also have a working implementation of exp-max in Python. Given a list of points, the number of distributions to look for, and a fixed standard deviation, it can usually locate the means within less than one standard devation for up to five distributions.

1.4 Some data

Unless otherwise stated, $k = 2$, $dim = 1$, $n = 100$, $\sigma = 3$, $-100 \leq \mu_{actual} \leq 100$ and $\epsilon = 0.01$.

Randomly Chosen Inital Means vs Fixed

	Trials	Actual Means	Model Means	Absolute diff	Error
fixed	10	(-36.8090705028, 40.6633826996)	(-36.8946854572, 41.1760521178)	(0.0856149544348, 0.51266941823)	(0.0023, 0.0126)
random	10	(-36.8090705028, 40.6633826996)	(-36.894685457234772, 41.176052117830231)	(0.0856149544347744, 0.5126694182302316)	(0.0023, 0.0126)

Choosing the initial mean to be the min and max, as opposed to randomly choosing them in the upper half and lower half of the range of data values does not seem to make a difference.

With $k = 2$, 10 random trials of randomly selecting initial means yield error of (0.0172689842972, 0.0127461938775), which seems fairly good.

With $k = 3$, however, we get the following error values:

```
(0.028750598793282213, 0.0072093421635908705, 0.00919114184419186)
(0.041610845591266356, 0.0008388275724106088, 0.012176240155513182)
(0.011526423518458699, 2.7579486693719364, 1.6055964277443517)
(0.5275713546022011, 0.94884037058708, 0.13040451659059452)
(0.003565330217785441, 0.008564614511567386, 0.0003489881154241546)
(0.034788935276553676, 1.5905820435426408, 0.1397449102975081)
(0.0016378658584708843, 0.006751174806963033, 0.009998792198633286)
(0.008309733238875022, 0.016852273163172424, 0.003988309913453066)
(0.003166066783719598, 0.004945421417376647, 0.0012873338152945113)
(0.02323640024433738, 0.007196434283362816, 0.00020374166800173914)
Average: [0.06841635541249505, 0.53497291714201, 0.1912940402342966]
Runtime: 24.476 seconds
```

While most were fairly small, the average error is large due to the influence of a few errors that are very large – probably due to poor choice of initial means. More trials for each set of data probably would have fixed this.

Choosing the initial means as the min of the data, the max of the data, and their average yielded much better results:

```
(4.273820952074582e-05, 0.047495838538832484, 0.0035334024827284463)
(0.005066045623486773, 0.0036786384737673206, 0.003920287784773603)
(0.0021423505244903837, 0.005711630606053978, 0.0012481065960590256)
(0.004477880115874205, 0.007893610895341538, 0.008326757402101271)
(0.004983047511272297, 0.007015704447088302, 0.005672681435631253)
(0.0030292760704276065, 2.293145224035619, 0.0648166471623687)
(0.002195788842099555, 0.005869201044331433, 0.0021932517633572544)
(0.002823877813997543, 0.0032915003996282313, 0.009455250972531064)
(0.006674093624508037, 0.2274426182131443, 0.009502693668459182)
(0.006924882353794085, 0.004669473625713794, 0.00672439019813716)
Average:[0.0038359980689471225, 0.26062134402795206, 0.011539346946614697]
Runtime: 23.254 seconds.
```

But again, the choice of the center initial mean resulted in large error for one of the trials.

1.5 Issues

1. Time complexity - for $k \geq 5$ and 100 points in each distribution, the code takes nearly half a minute to run. The code could possible run in $O(n^3)$ or even $O(2^n)$ time.

2. Underflow (NaN) - if the initial model means are randomly chosen too far from the actualy means, or if the model standard deviation is below 1, the probability that a data point is in either distribution ends up being 0, which results in division by 0, making the results meaningless.