Comparing various classifiers on the Characters Trajectories dataset.

By Shuning Zhao
2017/11/20.

Abstract
This article applies a range of softmax classifier models such as hidden Markov model, logistic regression and Gaussian mixture model to the characters trajectories dataset comparing their error rate (ER) on the validation set.

Introduction
The characters trajectories dataset is a dataset consisting labeled samples of pen pin trajectories recorded while writing individual characters. The x variables of this dataset contains 3-dimensional representations of trajectories horizontal and vertical velocities and the pen tip force respectively. This data set was created for a PhD study regarding primitive extraction with sequential models.

We chose Gaussian Hidden Markov Model (GHMM) as hidden Markov models are often used for time structured sequential data with GHMM being the most common one. We chose Gaussian mixture models(GMM) because different letters has different strokes hence the vastly different pen trajectories when writing them. Logistic regression was chosen simply because it is a well known softmax classifier model.

Model
The Hidden Markov Model consists of a discrete time, discrete state Markov chain where all states $z_t \in \{1, ..., K\}$ are hidden. The observation model $p(x_t|z_t)$ has a joint distribution of the form

$$p(z_{1:T}, x_{1:T}) = p(z_{1:T})p(x_{1:T}, z_{1:T}) = \Big[p(z_1)\prod_{t=2}^{T} p(z_t|z_{t-1})\Big]\Big[\prod_{t=1}^{T} p(x_t|z_t)\Big]$$

(Blunsom, 2004).

Mixture model is a latent variable model with $z_i \in \{1, ..., K\}$ representing the latent states. Where the Gaussian mixture model(GMM) is the most widely used mixture model. In GMM each base distribution in the mixture is a multivariate Gaussian with mean μ_k and covariance matrix Σ_k. Hence GMMs has the form

$$p(x_i|\theta) = \sum_{k=1}^{K} \mathcal{N}(x_i|\mu_k, \Sigma_k)$$

(Murphy, 2012).

Logistic regression is a regression model where the output is categorical. The link function for the logistic regression is the logit function. For a regression with output y and m input variables $x_1, ..., x_m$.

$$g(F(x)) = \ln\Big(\frac{F(x)}{1 - F(x)}\Big) = \beta_0 + \beta_1 x_1 + ... + \beta_m x_m$$

where $F(X)$ is the probability that the dependent variable equals a case, given some linear combination of the variables (Hosmer Jr et al., 2013).

Inference
Both the Gaussian Mixture model and Gaussian Hidden Markov Model in this article were trained using the Expectation Maximization(EM) algorithm. This iterative algorithm exploits the fact that

if the data were fully observed, then the MAP estimate would be easy to compute. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step (Moon, 1996).

In the case of GMM we have the expected complete-data log likelihood

$$Q(\theta, \theta^{old}) = \mathbb{E}_{p(Z|X,\theta^{old})}\big[\sum_i \log p(x_i, z_i|\theta)\big]$$

where we want to maximize $\mathcal{L}_{lower}(q, \theta) = Q(\theta, \theta^{old}) + \mathbb{H}(q)$ with respect to $q(Z|X)$ subject to a whole range of constraints (Murphy, 2012).

In the case of HMM the EM algorithm is also known is Baum-Welch algorithm (Baum et al., 1970) when applied to HMMs where we have the complete data log likelihood given by

$$\begin{aligned} Q(\theta, \theta^{old}) = &\sum_{k=1}^{K} \mathbb{E}[N_k^1] \log \pi_k + \sum_{j=1}^{K}\sum_{k=1}^{K} \mathbb{E}[N_{jk}] \log A_{jk} \\ &+ \sum_{i=1}^{N}\sum_{t=1}^{T_i}\sum_{k=1}^{K} p(z_t = k|x_i, \theta^{old}) \log p(x_{i,t}|\phi_k). \end{aligned}$$

where the expected terms are given by $\mathbb{E}[N_k^1] = \sum_{i=1}^{N} p(z_{i1} = k|x_i, \theta^{old})$
$\mathbb{E}[N_{jk}] = \sum_{i=1}^{N}\sum_{t=2}^{T_i} p(z_{i,t-1} = j, z_{i,t} = k|x_i, \theta^{old})$ and
$\mathbb{E}[N_j] = \sum_{i=1}^{N}\sum_{t=1}^{T_i} p(z_{i,t} = j|x_i, \theta^{old})$.

In the case of logistic regression, since the standard logistic regression can only have binary outputs $y \in \{0, 1\}$ we need to generalize this to a softmax regression so it could handle multiple classes $y \in \{1, ..., K\}$. For the vectors $\beta = \{\beta_0, \beta_1 ... \beta_m\}$ and $x \in \mathbb{R}^m$ and the training set $\{(x^{(1)}, y^{(1)}), ..., (x^{(n)}, y^{(n)})\}$ re-arrange $g(F(x))$ and we get $F(X) = \frac{1}{1+exp(-\beta^T x)}$ the model parameters β were trained to minimize the loss function

$$J(\beta) = -\Big[\sum_{i=1}^{m} y^{(i)} \log F(x^{(i)}) + (1 - y^{(i)}) \log(1 - F(x^{(i)}))\Big]$$

Given an input x we want our hypothesis to estimate the probability that $P(y = k|x)$ for each k in $1, ..., K$ our hypothesis F(X) now takes the form:

$$F(x) = \begin{bmatrix} P(y = 1|x; \beta) \\ P(y = 2|x; \beta) \\ \vdots \\ P(y = K|x; \beta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{K} \exp(\beta^{(j)T} x)} \begin{bmatrix} \exp(\beta^{(1)T} x) \\ \exp(\beta^{(2)T} x) \\ \vdots \\ \exp(\beta^{(K)T} x) \end{bmatrix}$$

here $\beta^{(1)}, \beta^{(2)}, ..., \beta^{(K)} \in \mathbb{R}^m$ are the parameters of the model. Hence the loss function becomes

$$J(\beta) = -\Big[\sum_{i=1}^{m}\sum_{k=1}^{K} \mathbb{I}\{y^{(i)} = k\} \log \frac{\exp(\beta^{(k)T} x^{(i)})}{\sum_{j=1}^{K} \exp(\beta^{(j)T} x^{(i)})}\Big]$$

(Menard, 2002).

Parameter Estimation

In the case of GMM for E-Step we want to minimize the lagrangian $L = q(Z|X)\log q(Z|X)$. From page 10 of the week 4 lecture notes, we update the equation below:

$$r_{ik} = \frac{\pi_k p(x_i|\theta^{old})}{\sum_{k'} \pi_{k'} p(x_i|\theta^{old})}$$

For M-Step we maximize $\mathcal{L}_{lower}(q, \theta)$ w.r.t θ. Hence we have

$$\hat{\pi}_k = \frac{1}{N}\sum_{i=1}^{N} r_k^{(i)}$$

where $r_k = \sum_i r_{ik}$ is the weighted number of points assigned to cluster k.
To derive the M step for the μ_k and Σ_k terms we have the following update equations

$$\hat{\mu}_k = \frac{\sum_i r_{ik} x_i}{r_k}$$
$$\hat{\Sigma}_k = \frac{\sum_i r_{ik}(x_i - \mu_k)(x_i - \mu_k)^T}{r_k} = \frac{\sum_i r_{ik} x_i x_i^T}{r_k} - \mu_k \mu_k^T$$

Please see appendix A for detailed proof.

Similarly in the case of GHMM for the E-step the expected sufficient statistics $\mathbb{E}[N_k^1], \mathbb{E}[N_{jk}]$ and $\mathbb{E}[N_j]$ can be computed by running the forwards-backwards algorithm on each sequence. This algorithm computes smoothed node and edge marginals such as $\gamma_{i,t}(j) = p(z_t = j|x_{i,1:T_i}, \theta)$.

For the M-Step we have

$$\hat{A}_{jk} = \frac{\mathbb{E}[N_{jk}]}{\sum_{k'} \mathbb{E}[N_{jk'}]}, \hat{\pi}_k = \frac{\mathbb{E}[N_k^1]}{N}$$
$$\hat{\mu}_k = \frac{\mathbb{E}[\bar{x}_k]}{\mathbb{E}[N_k]}, \hat{\Sigma}_k = \frac{\mathbb{E}[(\bar{xx}_k^T)] - \mathbb{E}[N_k]\hat{\mu}_k\hat{\mu}_k^T}{\mathbb{E}[N_k]}$$

where the expected sufficient statistics are given by $\mathbb{E}[\bar{x}_k] = \sum_{i=1}^{N}\sum_{t=1}^{T_i}\gamma_{i,t}(k)x_{i,t}$ and $\mathbb{E}[(\bar{xx}_k^T)] = \sum_{i=1}^{N}\sum_{t=1}^{T_i}\gamma_{i,t}(k)x_{i,t}x_{i,t}^T$ (Bicego et al., 2008).

In the case of logistic regression, we need to solve for the minimum of $J(\beta)$, taking derivatives we can see that

$$\nabla_{\beta^{(k)}} J(\beta) = -\sum_{i=1}^{m}\left[x^{(i)}\Big(\mathbb{I}\{y^{(i)} = k\} - P(y^{(i)} = k|x^{(i)};\beta)\Big)\right]$$

One can plug the above formula into a iterative optimization algorithm such is Netwon's method or line search to minimize $J(\beta)$ (Zhang and Hager, 2004).

Results
For modeling training the training set of the characters trajectories dataset was split into training and validation sets on a 70% to 30% ratio. Then run testing set through the model for the submission results.

For Data Processing, all observations were zero-padded so they are all the same length.

For Gaussian HMM the training set was divided into 20 subsets grouped by their output class. Then trained using Gaussian HMM model for each class. After all 20 models has been trained the data were ran through each of the 20 models where the log likelihood for the observation on each of the models are obtained. The model with the highest log likelihood indicates that this observation belongs to

this class. All numbers of hidden states from 3 to 30 were tried, using 2-fold cross validation due to the small size of the training set.

The code for GMM used was the code from the recognizing handwritten digits MNIST task with little changes. A variety of models were trained over covariance type 'full', 'tied' and 'spherical' and with the number of components ranging from 1 to 30. As well as trying with and without PCA with number of PCA components going from 10 to 50.

The logistic regression were fitted the training set into the model using 0, 2 and 10 fold cross validation, the results were all similar on the validation set. Hence no cross validated versions were submitted in the final codes.

Final results are as follow:

Model	Training ER	Training MNLP	Validation ER	Validation MNLP
Gaussian HMM	0.057	-927.80	0.130	-875.12
GMM	0.11	350.79	0.133	283.97
GMM with PCA	0.092	30.60	0.107	18.66
Logistic Regression	0.028	0.05	0.030	0.17

where the Error Rate (ER) and the mean negative log probability (MNLP) are calculated using the following formulas

$$ER = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = y_i)$$

$$MNLP = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} \mathbb{I}(y_i = j) \log p_{pred}(y_i|x_i).$$

A Appendix

The expected complete data log likelihood for GMM is given by

$$\begin{aligned}
Q(\theta, \theta^{old}) &= \mathbb{E}_{p(Z|X,\theta^{old})}\big[\sum_i \log p(x_i, z_i|\theta)\big] \\
&= \sum_i \mathbb{E}\Big[\log\Big[\prod_{k=1}^{K}(\pi_k p(x_i|\theta_k))^{\mathbb{I}(z_i=k)}\Big]\Big] \\
&= \sum_i \sum_k \mathbb{E}[\mathbb{I}(z_i = k)] \log[\pi_k p(x_i|\theta_k)] \\
&= \sum_i \sum_k p(z_i = k|x_i, \theta^{old}) \log[\pi_k p(x_i|\theta_k)] \\
&= \sum_i \sum_k r_{ik} \log \pi_k + \sum_i \sum_k r_{ik} \log p(x_i|\theta_k)
\end{aligned}$$

where $r_{ik} = p(z_i = k|x_i, \theta^{old})$ is the responsibility that cluster k takes for data point i. This is calculated during the E step of the EM algorithm.

For the M step of the EM algorithm for GMM we look at the parts of Q that depend on μ_k and Σ_k to calculate μ_k and Σ_k.

$$\begin{aligned}
l(\mu_k, \Sigma_k) &= \sum_i \sum_k r_{ik} \log p(x_i|\theta_k) \\
&= -\frac{1}{2}\sum_i r_{ik}[\log|\Sigma_k| + (x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k)]
\end{aligned}$$

Using the substitution $y_i = x_i - \mu$ and the chain rule we have

$$\begin{aligned}
\frac{\delta(x_i - \mu_k)^T \Sigma_k^{-1}(x_i - \mu_k)}{\delta \mu_k} &= \frac{\delta}{\delta y_i} y_i^T \Sigma_k^{-1} y_i \frac{\delta y_i}{\delta \mu_k} \\
&= -1(\Sigma_k^{-1} + \Sigma_k^{-T}) y_i
\end{aligned}$$

Hence

$$\begin{aligned}
\frac{\delta}{\delta \mu_k} l(\mu_k, \Sigma_k) &= -\frac{1}{2}\sum_i r_{ik}[-2\Sigma_k^{-1}(x_i - \mu_k)] \\
&= \Sigma_k^{-1} \sum_i r_{ik}(x_i - \mu_k) \\
&= 0 \\
\hat{\mu}_k &= \frac{\sum_i r_{ik} x_i}{r_k}
\end{aligned}$$

Using the trace trick we can rewrite the log-likelihood for Σ_k as follow

$$\begin{aligned}
l(\Sigma_k) &= \frac{r_{ik}}{2}\log|\Sigma_k^{-1}| - \frac{1}{2}\sum_i tr[(x_i - \mu_k)(x_i - \mu_k)^T \Sigma_k^{-1}] \\
&= \frac{r_{ik}}{2}\log|\Sigma_k^{-1}| - \frac{1}{2} tr[S_{\mu_k}\Sigma_k^{-1}]
\end{aligned}$$

where $S_{\mu_k} = \sum_i (x_i - \mu_k)(x_i - \mu_k)^T$ taking derivatives of this expression with respect to Σ_k^{-1} gives

$$\frac{\delta l(\Sigma_k^{-1})}{\delta \Sigma_k^{-1}} = \frac{r_k}{2}\Sigma_k^T - \frac{1}{2}S_{\mu_k}^T = 0$$

$$\Sigma_k^{-T} = \Sigma_k^{-1} = \Sigma_k = \frac{1}{r_k}S_{\mu_k}$$

Therefor $\hat{\Sigma}_k = \frac{\sum_i r_{ik}(x_i - \mu_k)(x_i - \mu_k)^T}{r_k} = \frac{\sum_i r_{ik} x_i x_i^T}{r_k} - \mu_k \mu_k^T$.

References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, 41(1):164–171.

Bicego, M., Gonzalez-Jimenez, D., Grosso, E., and Castro, J. A. (2008). Generalized gaussian distributions for sequential data classification. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE.

Blunsom, P. (2004). Hidden markov models. *Lecture notes, August*, 15:18–19.

Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied logistic regression*, volume 398. John Wiley & Sons.

Menard, S. (2002). *Applied logistic regression analysis*, volume 106. Sage.

Moon, T. K. (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60.

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

Zhang, H. and Hager, W. W. (2004). A nonmonotone line search technique and its application to unconstrained optimization. *SIAM journal on Optimization*, 14(4):1043–1056.