MVA – Probabilistic Graphical Models
Homework 1

Wilson Jallet*

December 8, 2019

1 Learning in discrete graphical models

We suppose that $z \sim \mathcal{M}(\pi, 1)$ and, for every $m \in [\![1..M]\!]$, $x|z = m \sim \mathcal{M}(\theta_m, 1)$ where $\pi \in \Delta_{M-1} = \{p \in \mathbb{R}_+^M : \sum_{m=1}^M p_m = 1\}$ and for every $m \in [\![1..M]\!]$, $\theta_m \in \Delta_{K-1}$.

Given data $\mathcal{X} = ((x_n, z_n))_{1 \leq n \leq N}$, its likelihood under parameters (π, θ) is

$$\ell(\pi, \theta; \mathcal{X}) = \prod_{n=1}^{N} p(x_n|z_n)p(z_n) = \prod_{n=1}^{N} \theta_{z_n, x_n} \pi_{z_n} \tag{1}$$

And log-likelihood

$$L(\pi, \theta; \mathcal{X}) = \sum_{n=1}^{N} \log \theta_{z_n, x_n} + \log \pi_{z_n} \tag{2}$$

Computing the maximum likelihood estimate (MLE) is equivalent to the problem

$$\begin{aligned} &\min_{\pi, \theta} \quad -L(\pi, \theta; \mathcal{X}) \\ &\text{s.t. } \sum_{m=1}^{M} \pi_m = 1 \text{ and } \sum_{k=1}^{K} \theta_{mk} = 1 \text{ for } m \in [\![1..M]\!] \end{aligned} \tag{3}$$

This is a convex optimization problem.

We introduce the Lagrangian

$$\mathcal{L}(\pi, \theta, \nu, \xi) = -L(\pi, \theta; \mathcal{X}) + \nu(\pi \mathbb{1} - 1) + \sum_{m=1}^{M} \xi_m(\theta_m \mathbb{1} - 1), \quad \nu \in \mathbb{R},\ \xi \in \mathbb{R}^M$$

The partial derivatives are given by

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \pi_m} &= -\frac{\sum_{n=1}^{N} \mathbb{1}_{\{z_n = m\}}}{\pi_m} + \nu \\ \frac{\partial \mathcal{L}}{\partial \theta_{mk}} &= -\frac{\sum_{n=1}^{N} \mathbb{1}_{\{z_n = m, x_n = k\}}}{\theta_{mk}} + \xi_m \end{aligned}$$

with the convention that the first terms are 0 if the set $\mathcal{A}_m = \{n \in [\![1..n]\!] : z_n = m\}$ (resp. $\mathcal{B}_{m,k} = \{n \in [\![1..n]\!] : z_n = m,\ x_n = k\}$) is empty: then π_m (resp. θ_{mk}) does not appear in the log-likelihood.

*wilson.jallet@polytechnique.org

The Euler optimality conditions lead to

$$\nu\pi_m^* = \sum_{n=1}^N \mathbb{1}_{\{z_n=m\}} = |\mathcal{A}_m| \tag{4a}$$

$$\xi_m\theta_{mk}^* = \sum_{n=1}^N \mathbb{1}_{\{z_n=m, x_n=k\}} = |\mathcal{B}_{m,k}| \tag{4b}$$

which reduces to $0 = 0$ for indices $i \in \mathcal{A}_m$ or $n \in \mathcal{B}_{m,k}$. Primal feasibility then implies $\nu = N$ and $\xi_m = \sum_{n=1}^N \mathbb{1}_{\{z_n=m\}} = |\mathcal{A}_m|$.

Conclusion. Then, the MLE for the model is given by

$$\begin{aligned} \pi_m^* &= \frac{|\mathcal{A}_m|}{n} \\ \theta_{m,k}^* &= \frac{|\mathcal{B}_{m,k}|}{|\mathcal{A}_m|} \end{aligned} \tag{5}$$

which is the intuitive solution: the empirical probabilities of each class.

2 Linear classification

2.1 Generative model (LDA)

Maximum likelihood estimator. Denoting $p = (1-\pi, \pi)$, the log-likelihood of the data under the parameters (p, μ, Σ) is

$$L(p,\mu,\Sigma) = -\sum_{n=1}^N \frac{1}{2}(x_n - \mu_{y_n})^T\Sigma^{-1}(x_n - \mu_{y_n}) - \frac{n}{2}\log|\Sigma| + \sum_{n=1}^N \log p_{y_n} \tag{6}$$

We introduce the precision matrix $W \coloneqq \Sigma^{-1}$: the MLE problem is equivalent to the convex optimization problem

$$\begin{aligned} \min_{p,\mu,W} \ & \sum_{n=1}^N \left(\frac{1}{2}(x_n - \mu_{y_n})^T W (x_n - \mu_{y_n}) - \log p_{y_n}\right) - \frac{n}{2}\log|W| \\ \text{s.t. } & p_0 + p_1 = 1 \\ & W \succ 0 \end{aligned} \tag{7}$$

We again introduce the Lagrangian

$$\mathcal{L}(p,\mu,W,\nu) = -L(p,\mu,W) + \nu(p_0 + p_1 - 1)$$

and denote the classes $\mathcal{C}_i = \{n : y_n = i\}$. The partial derivatives are

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial p_i} &= -\frac{|\mathcal{C}_i|}{p_i} + \nu \\ \nabla_{\mu_i}\mathcal{L} &= \sum_{n\in\mathcal{C}_i} W(\mu_i - x_n) \end{aligned} \tag{8}$$

The Euler optimality conditions for π and primal feasibility lead to, as before,

$$p_i^* = \frac{|\mathcal{C}_i|}{N} \quad \text{for } i = 0, 1 \tag{9}$$

(a) Point cloud of dataset `trainA` in $\mathbb{R}^2$, and decision boundary (**??**). It is apparent this dataset is linearly separable.

(b) Mixture of Gaussians underlying the LDA for dataset `trainA`.

(c) Point cloud of dataset `trainB` along with the LDA decision boundary. The classes are more interlaced than dataset A, but less than C.

(d) Point cloud of dataset `trainC` along with the LDA decision boundary. The classes are much more interlaced than datasets A and B.

Figure 1: Linear discriminant analysis.

so the Bernoulli law parameter is

$$\pi^* = p_1^* = |\mathcal{C}_1|/n \tag{10}$$

The Gaussian means are given by the class barycenters:

$$\mu_i^* = \frac{1}{|\mathcal{C}_i|}\sum_{n\in\mathcal{C}_i} x_n \quad \text{for } i = 0, 1 \tag{11}$$

Recalling that $\nabla_M \log|M| = M^{-1}$, we have the Euler condition for W

$$\nabla_W \mathcal{L} = \frac{1}{2}\sum_n (x_n - \mu_{y_n})(x_n - \mu_{y_n})^T - \frac{n}{2}W^{-1} = 0$$

so at the optimum the precision matrix is the empirical covariance

$$\Sigma^* = (W^*)^{-1} = \frac{1}{n}\sum_{n=1}^{N}(x_n - \mu_{y_n}^*)(x_n - \mu_{y_n}^*)^T \tag{12}$$

Conditional distribution. The posterior distribution of the class label y given x is

$$p(y=1|x) = \frac{\pi e^{-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)}}{\pi e^{-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)} + (1-\pi)e^{-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)}} \tag{13}$$

In the logistic regression model, the posterior distribution is given by

$$p(y=1|x) = \frac{1}{1+e^{-f(x)}}$$

Decision boundary. Setting $p(y=1|x) = 1/2$, we have that x satisfies

$$\log\left(\frac{\pi}{1-\pi}\right) = (\mu_1 - \mu_0)^T\Sigma^{-1}\left(x - \frac{\mu_0+\mu_1}{2}\right) \tag{14}$$

This is the equation of a hyperplane with normal vector $a = \Sigma^{-1}(\mu_1 - \mu_0)$. Finding a support vector w to a 2D line of normal vector a can be done by start with e and define $w := e - \langle \frac{a}{\|a\|}, e\rangle \frac{a}{\|a\|}$. **??** shows the contour plot of the posterior probability (**??**), along with the decision boundary.

Figure 2: Logistic regression on datasets A and C. The decision boundary is apparent, and the transition from one class to another is sharper than in LDA or linear regression, but we see interlaced classes lead to a “fuzzy” boundary.

2.2 Logistic regression

Introduce the logistic function

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

which has property $\sigma'(x) = \sigma(x)(1-\sigma(x))$. Under the model, the probability that $y=1$ given x is

$$p(y=1|x) = \sigma(w^T x)$$

Denoting $\varepsilon_n = 2y_n - 1 \in \{-1, 1\}$ and $\bar{x}_n = (1, x_n)$, the log-likelihood is given by

$$L(w) = \sum_{n=1}^{N} \log \sigma(\varepsilon_n w^T \bar{x}_n) \tag{15}$$

In this formulation, the vector $w \in \mathbb{R}^3$ also holds the bias as w_0. To compute the MLE $w^* = \operatorname{argmax}_w L(w)$, we can use Newton's method: we only require the gradient and hessian matrix, which are given respectively by

$$\nabla_w L(w) = \sum_{n=1}^{N} \left(1 - \sigma(\varepsilon_n w^T \bar{x}_n)\right) \varepsilon_n \bar{x}_n \tag{16}$$

$$\nabla_w^2 L(w) = \sum_{n=1}^{N} (\sigma(\varepsilon_n w^T \bar{x}_n) - 1)\sigma(\varepsilon_n w^T \bar{x}_n)\bar{x}_n \bar{x}_n^T \tag{17}$$

We obtain the results in **??**, with weights

$$(b, w_1, w_2) = (174.22681818, 7.82121503, -30.17412171)$$

2.3 Linear regression

The linear regression model is as follows:

$$y = w^T x + b + \varepsilon \tag{18}$$

The weights $\bar{w} = (b, w)$ are given using the usual formula

$$X^T X \bar{w} = X^T Y \tag{19}$$

with $X = \begin{bmatrix} 1 & x_1^T \\ \vdots & \vdots \\ 1 & x_n^T \end{bmatrix}$ and $Y = (y_1, \dots, y_n)^T \in \mathbb{R}^n$.

On dataset A, we obtain the weights

$$(b, w_1, w_2) = (1.38345774, 0.05582438, -0.17636636)$$

(a) Linear regression for dataset A. (b) For dataset C.

Figure 3: Linear regression model (**??**) with decision boundary.

	A	B	C	mean
lda_train	0.000000	0.035	0.046667	0.027222
lda_test	0.036667	0.010	0.035000	0.027222
logistic_train	0.000000	0.010	0.030000	0.013333
logistic_test	0.036667	0.000	0.060000	0.032222
linear_train	0.000000	0.020	0.026667	0.015556
linear_test	0.036667	0.010	0.060000	0.035556
qda_train	0.000000	0.010	0.026667	0.012222
qda_test	0.040000	0.010	0.060000	0.036667

Figure 4: Misclassication errors for the different models on the train and test sets: LDA, logistic regression, linear regression, and QDA.

2.4 Application

We computed the different classification errors for all datasets (on the training and testing subsets) for the different models: they are summarized in **??**.

On average, the error on the training set is lower than that on the testing set.

LDA yields consistent results across training and testing on average: however, it overfits on dataset A which has a linearly separable training set, which is a problem all the other models have. It has higher training than testing error on datasets B and C which had training sets with interlaced classes, but it ends up being robust when testing on them (see **????**).

Logistic regression performs well overall too, and has lower error on the non-linearly separable datasets B and C, offering better overall training error and staying consistent with testing.

The linear model has inconsistent results across the datasets: it generalizes especially poorly in the case of dataset C.

2.5 QDA model

This time, we suppose a mixture model for x where the covariance matrices Σ_0, Σ_1 are not necessarily equal. The maximum likelihood estimates **????** work out the same[1], but the estimates of the precision matrices $W_0 = \Sigma_0^{-1}$ and $W_1 = \Sigma_1^{-1}$ and the structure of the decision boundary change. Writing out the Euler conditions for W_0, W_1 lead to the MLEs being the empirical covariances of each class:

$$\Sigma_i^* = (W_i^*)^{-1} = \frac{1}{|\mathcal{C}_i|} \sum_{n \in \mathcal{C}_i} (x_n - \mu_i^*)(x_n - \mu_i^*)^T \tag{20}$$

[1]The barycenters in **??** are independent of the covariance.

(a) QDA on dataset A. (b) QDA on dataset C.

Figure 5: Point cloud and decision boundary for the QDA model.

The QDA model's plots for the posterior probabilities and decision boundaries are given **??**. The model overfits on dataset A (as the other models do). It has good and consistent results on dataset B, but has worse results on dataset C (just like the other models except for LDA).