Graphical models - HWK 1

Philbert Alexandre, Romero Raphaël

October 17, 2017

1 Learning in discrete graphical models

We must maximize the likelihood of the observations $(x_1, z_1)...(x_n, z_n)$ with respect to the parameters $\pi = (\pi_1, ..., \pi_n)$ and $\theta = (\theta_{m,k})_{1 \leq m \leq M, 1 \leq\ k \leq K}$

To do this we prefer to minimize the negative log-likelihood :

$$l(\pi, \theta) = -log(\prod_{i=1}^{n} \pi_{z_i} \theta_{z_i, x_i})$$

$$= -\sum_{i=1}^{n} (\sum_{m=1}^{M} log(\pi_m) 1_{\{z_i = m\}} + \sum_{m=1}^{M} log(\theta_{m,k} 1_{\{(z_i, x_i) = (m,k)\}})$$

under the constraints

$$\sum_{m=1}^{M} \pi_m = 1 \ \ and \ \ \forall m = 1, ..., M, \ \sum_{k=1}^{M} \theta_{m,k} = 1$$

In order to do this we minimize the lagragian associated with this problem

$$\mathcal{L}(\pi, \theta, \nu_0, \nu_1, ..., \nu_M) = -\sum_{i=1}^{n} (\sum_{m=1}^{M} log(\pi_m) 1_{\{z_i = m\}} + \sum_{m=1}^{M} log(\theta_{m,k} 1_{\{(z_i, x_i) = (m,k)\}}$$

$$+\nu_0 \sum_{m=1}^{M} (\pi_m - 1) + \sum_{m=1}^{M} \nu_m \sum_{k=1}^{K} (\theta_{m,k} - 1)$$

Since the Lagrangian is convex , we obtain the solution by derivating it with respect to the model parameters:

$$\frac{\partial \mathcal{L}}{\partial \pi} = 0 \leftrightarrow \forall m = 1, ..., M - \frac{n_m}{n} + \nu_0 = 0$$

where $n_m = \#\{i = 1, ..., n, z_i = m\}$

Since $\sum_{m=1}^{M} \pi_m = 1$ we have that

$$\nu_0 = \sum_{m=1}^{M} n_m = \sum_{m=1}^{M} \sum_{k=1}^{K} 1_{\{z_i = m\}} = n$$

So finally

$$\forall m = 1, ..., M, \pi_m = \frac{n_m}{n}$$

On the other hand,

$$\frac{\partial \mathcal{L}}{\partial \theta} = 0 \Leftrightarrow \forall m = 1, ..., M, k = 1, ..., K, -\frac{n_{m,k}}{\theta m, k} + \nu_m = 0$$

where $n_{m,k} = \#\{i = 1, ..., n, (z_i, x_i) = (m, k)\}$

From the condition $\sum_{k=1}^{K} \theta_{m,k} = 1$ we have that

$$\nu_m = \sum_{k=1}^{K} n_{m,k} = \sum_{i=1}^{n} \sum_{k=1}^{K} 1_{(z_i,x_i)=(m,k)} = n_m$$

So finally $\forall m = 1, ..., M, k = 1, ..., K$

$$\theta_{m,k} = \frac{n_{m,k}}{n_m}$$

2 Linear classication

1. **Generative model (LDA)**
(a)
Model : $y \sim Bernouilli(\pi)$ *and* $x|y = 1 \sim Normal(\mu_i, \Sigma)$
let's note $\Theta = (\pi, \mu_1, \mu_2, \Sigma)$ given the model we have that

$$p_\Theta(x, y) = p_\Theta(y)p_\Theta(x|y) = \pi^y(1-\pi)^{1-y}(\frac{1}{\sqrt{(2\pi)^d det\Sigma}} exp(-\frac{1}{2}(x-\mu_y)^t\Sigma^{-1}(x-\mu_y)$$

Hence the negative log-likelihood of the given observation $(x_1, y_1)...(x_n, y_n)$ is

$$l_\theta = \sum_{i=1}^{n} y_i \log(\pi) + \sum_{i=1}^{n}(1-y_i)\log(1-\pi)$$

$$+\frac{nd}{2}\log(2\pi) + \frac{n}{2}\log(det\Sigma) + \frac{1}{2}\sum_{i=1}^{n}(x-\mu_{y_i})^t\Sigma^{-1}(x-\mu_{y_i})$$

We first notice that the terms in π are independent from the other variables so we can first minimize l_θ w.r.t π which is exactly the MLE of Bernoulli model hence we have

$$\hat{\pi} = \frac{\sum_{i=1}^{n} y_i}{n}$$

Minimizing l_Θ w.r.t to (μ_0, μ_1, Σ) is similar to the MLE for the multivariate Gaussian model. For a fixed Σ the problem is convex in (μ_0, μ_1)
By fixing $\nabla_{\mu_i} l_\Theta = 0$ we obtain that

$$\hat{\mu_1} = \frac{\sum_{i=1}^{n} x_i 1_{y_i=1}}{N1} \quad where \quad N1 = \sum_{m=1}^{n} 1_{y_i=1}$$

$$\hat{\mu_0} = \frac{\sum_{i=1}^{n} x_i 1_{y_i=0}}{N0} \quad where \quad N0 = \sum_{m=1}^{n} 1_{y_i=0}$$

Both results are independent from Σ
The rest of the proof is exactly the same as in lecture on the MLE for multivariate gaussian model so we have :

$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu_{y_i})^t(x_i-\mu_{y_i})$$

(b)
let's express $p(y = 1|x)$. From bayes rules we have

$$p(y=1|x) = \frac{p(x|y=1)p(y=1)}{p(x|y=1)p(y=1)p(x|y=0)p(y=0)}$$

$$= \frac{1}{1 + \frac{p(x|y=0)p(y=0)}{(x|y=1)p(y=1)}}$$

$$= \frac{1}{1 + \frac{1-\pi}{(\pi)} exp(-((\Sigma^{-1}(\mu_1 - \mu_0))^t x + \frac{1}{2}(\mu_0^t \Sigma^{-1} \mu_0 - \mu_1^t \Sigma^{-1} \mu_1))}$$

hence

$$p(y = 1|x) = \sigma(\omega^t x + b)$$

with $\omega = \Sigma^{-1}(\mu_1 - \mu_0)$, $b = \frac{1}{2}(\mu_0^t \Sigma^{-1} \mu_0 - \mu_1^t \Sigma^{-1} \mu_1) - \log(\frac{1-\pi}{\pi})$ and σ the sigmoid function

(c)

the following figures show the training datasets A, B and C as a point cloud $\mathbb{R}^2$. The blue points have the label 0, the green points have the label 1. The points are represented with their real labels. The blue line represent the hyperplane $p(y|x) = 0.5$ computed with the estimated parameters ω an b

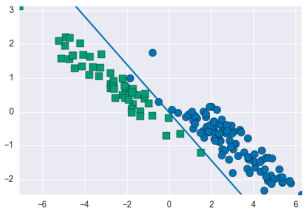

Figure 1: dataset A.train

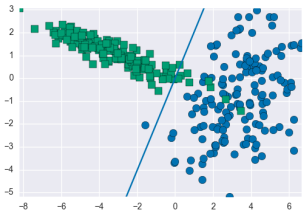

Figure 2: dataset B.train

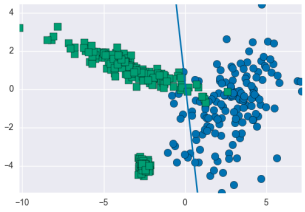

Figure 3: dataset C.train

2. **Logistic regression**
(a)

DataSet	ω	b
A	$\begin{bmatrix} -182.06790875 \\ -315.05580399 \end{bmatrix}$	-30.64398373
B	$\begin{bmatrix} -1.70518566 \\ 1.02378525 \end{bmatrix}$	1.3495913
C	$\begin{bmatrix} -2.20323226 \\ 0.70926552 \end{bmatrix}$	0.95918875

Table 1: Computed parameters for the logistic regression.

(b)

the following figures show the training datasets A, B and C as a point cloud $\mathbb{R}^2$. The blue points have the label 0, the green points have the label 1. The points are represented with their real labels. The blue line represent the hyperplane $p(y|x) = 0.5$ computed with the estimated parameters ω an b

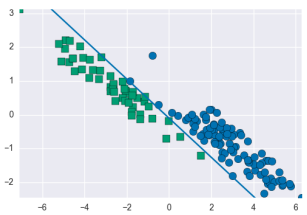

Figure 4: training dataset A

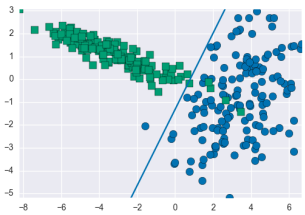

Figure 5: training dataset B

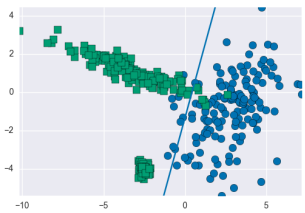

Figure 6: training dataset C

3. **Linear regression**
(a)

DataSet	ω	b
A	$\begin{bmatrix} -0.2640075 \\ -0.37259311 \end{bmatrix}$	0.492292037565
B	$\begin{bmatrix} -0.10424575 \\ 0.05179118 \end{bmatrix}$	0.500050427
C	$\begin{bmatrix} -0.12769333 \\ -0.01700142 \end{bmatrix}$	0.508399815826

Table 2: Computed parameters for the linear regression.

(b)
the following figures show the training datasets A, B and C as a point cloud $\mathbb{R}^2$. The blue points have the label 0, the green points have the label 1. The points are represented with their real labels. The blue line represent the hyperplane $p(y|x) = 0.5$ computed with the estimated parameters ω an b

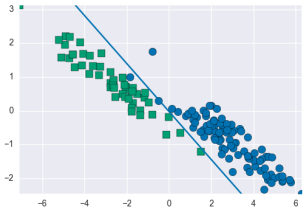

Figure 7: training dataset A

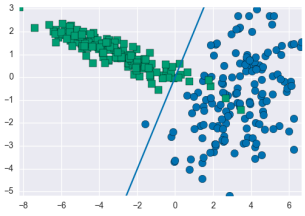

Figure 8: training dataset B

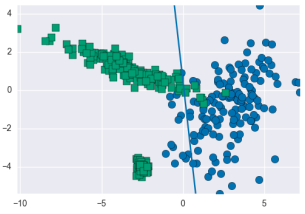

Figure 9: training dataset C

4. **(a)**

DataSet	LDA	Logistic regression	Linear regression
A.train	0.0133333333333	0.0	0.0133333333333
A.test	0.02	0.0346666666667	0.0206666666667
B.train	0.03	0.02	0.03
B.test	0.0415	0.043	0.0415
C.train	0.055	0.04	0.055
C.test	0.0423333333333	0.0226666666667	0.0423333333333

Table 3: misclassification error for each data set. The classifiers are learn form .train data sets

(b)
Our first observation is that LDA and Linear regression seem to be equivalent on data with binary labels. Our second observation is that Logistic regression always perform better (for these datasets) in the training datasets. However the LDA model gives better result on the test set A. This can be easily explained as the dataset A seem to fit the assumptions made for the LDA. Indeed, data labeled 1 and data labeled 0 seem to both be distributed according a Gaussian law with same covariance matrix. For the dataset B results for LDA and logistic regression are similar. Finally, we observe that the logistic regression performs better on the dataset A, which is far to fit the LDA assumptions.

5. **QDA model**
(a)

The MLE of parameters is similar to the LDA case except this time we have :

$$\hat{\Sigma}_j = \frac{1}{n_j}\sum_{i=1}^{n} 1_{yi=j}(x_i - \mu_{y_i})^t(x_i - \mu_{y_i}) \;\; with \;\; n_j = \sum_{i=1}^{n} 1_{y_i=j}$$

for j = 0,1
furthermore, we have

$$p(y = 1|x) = \sigma(x^t C x + \omega^t x + b)$$

with

$$C = \frac{1}{2}(\Sigma_0^{-1} - \Sigma_1^{-1})$$

$$\omega = \Sigma_1^{-1}\mu_1 - \Sigma_0^{-1}\mu_0$$

$$b = \frac{1}{2}(\mu_0^t\Sigma_0^{-1}\mu_0 - \mu_1^t\Sigma_1^{-1}\mu_1) - \log(\frac{1-\pi}{\pi}) - \frac{1}{2}\log(\frac{det\Sigma_1}{det\Sigma_0})$$

Data	C	ω	b
A	[[-0.7587202 -1.51361485], [-1.51361485 -2.86166338]]	[-7.36527314 -10.87335416]	0.577701351308
B	[[-0.47982628 -1.92382528] [-1.92382528 -5.52933507]]	[-2.28065009 1.45700199]	3.87732747977
C	[[0.00244301 -0.14592984] [-0.14592984 0.11805533]]	[-2.66524064 0.34888942]	0.110042748892

Table 4: Computed parameters for the QDA.

(b)
the following figures show the training datasets A, B and C as a point cloud $\mathbb{R}^2$. The blue points have the label 0, the green points have the label 1. The points are represented with their real labels. The black line represent the conic $p(y|x) = 0.5$ computed with the estimated parameters C, ω and b

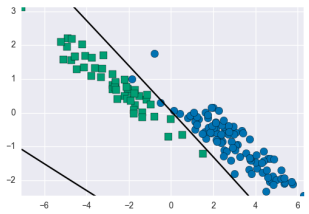

Figure 10: training dataset A

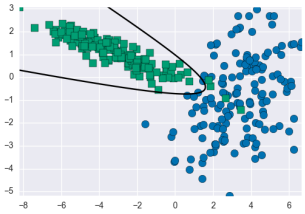

Figure 11: training dataset B

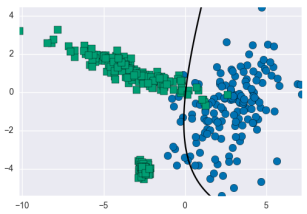

Figure 12: training dataset C

(c)

DataSet	train	test
A	0.0133333333333	0.0246666666667
B	0.0166666666667	0.0215
C	0.0525	0.0383333333333

Table 5: misclassification error for QDA for each data set. The QDA is learned form .train data sets

(d)
Our first observation is that QDA improves the prediction for the dataset B as the error ratio has been divided by 2 compare to the linear classifiers. Again, this could be explain by the fact that the data set B seems to fit the assumption that both labels follow a Gaussian

distribution but with different covariance matrix. For the data set A we observe similar result as for the LDA which is coherent as LDA is a special case of a QDA. Finally, QDA does not perform better than logistic regression for the dataset C. It is possible that the hypothesis required for the QDA model are too inappropriate, as the data labeled 1 is clearly not distributed according a Gaussian law.