Graphical models - HWK1

Abdessamad Ed-dahmouni

October 2018

1 Exercise 1: Learning in discrete graphical models

We define:

$$n_m \coloneqq \sum_{i=1}^{n} \mathbb{1}_{\{z_i=m\}} \quad \text{and } n_{m,k} \coloneqq \sum_{i=1}^{n} \mathbb{1}_{\{z_i=m\}} \mathbb{1}_{\{x_i=k\}}$$

We have:

$$\hat{\pi}_m = \frac{n_m}{n}$$
$$\hat{\theta}_{mk} = \frac{n_{mk}}{n_m}$$

2 Exercise 2.1(a): Generative model (LDA)

We define $y_i^j \coloneqq 1$ if $y_i = j$ and 0 otherwise for $j \in \{0, 1\}$. We have:

$$\hat{\pi} = \frac{\sum_{i=1}^{n} y_i}{n}$$
$$\hat{\mu}_j = \frac{1}{n_j} \sum_{i=1}^{n} y_i^j x_i$$
$$\hat{\Sigma} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{\mu}_{y_i})(x_i - \hat{\mu}_{y_i})^T$$

3 Exercise 2.5.(a): QDA model

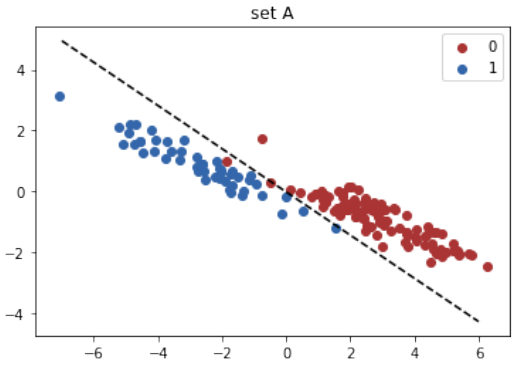
Similar to the calculations of the LDA model, we have:

$$\log(\ell(\pi, \mu, \Sigma)) = \sum_{j \in \{0,1\}} n_j \log(\pi_j) + \sum_{j \in \{0,1\}} \sum_{i=1}^{n} y_i^j \log(\mathcal{N}(x_i | \mu_j, \Sigma_j))$$

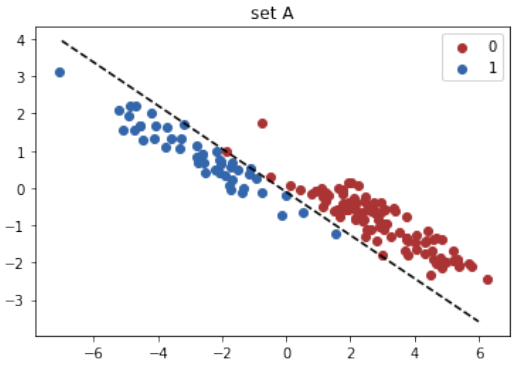
And the estimators are:

$$\hat{\pi} = \frac{\sum_{i=1}^{n} y_i}{n}$$
$$\hat{\mu}_j = \frac{1}{n_j} \sum_{i=1}^{n} y_i^j x_i$$
$$\hat{\Sigma}_j = \frac{1}{n_j} \sum_{i=1}^{n} y_i^j (x_i - \hat{\mu}_j)(x_i - \hat{\mu}_j)^T$$

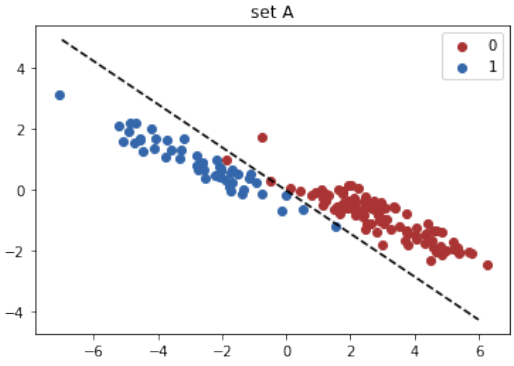
4 Set A



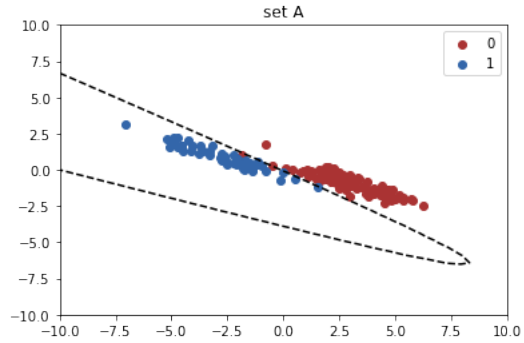
(a) LDA



(b) Logistic regression



(c) Linear regression



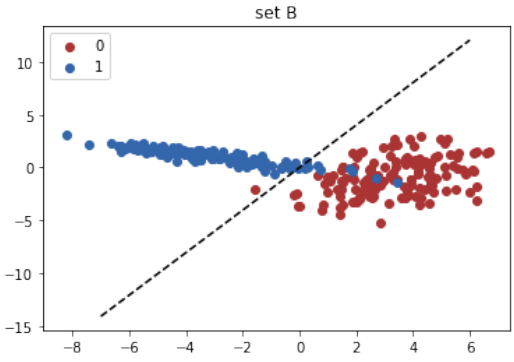
(d) QDA

Figure 1: Plot of the training data and decision boundaries.

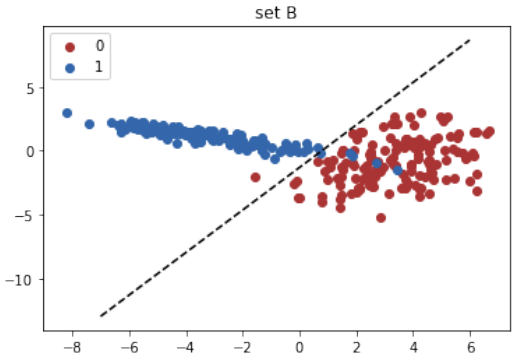
Method	Training error	Test error
LDA	0.0133	0.0200
Logistic regression	0.0000	0.0347
Linear regression	0.0133	0.0207
QDA	0.0067	0.0187

Comments: This data set has a sample size of 450 for training and 4500 for testing. The training set is considerably smaller than the test set, which helps us see if the models generalize well. Logistic regression tends to perform well on the training set, but has the highest test error. It's not really an overfitting situation since it has the same number of parameters as linear regression and less parameters than LDA and QDA, which can mean that modeling the conditional distribution of $y|x$ using logistic regression is not the right thing to do here.
QDA gives the best results both on the training and test errors. However, compared to LDA on the test set, the difference is very small, which means that we can get away with assuming $\Sigma_1 = \Sigma_2$.

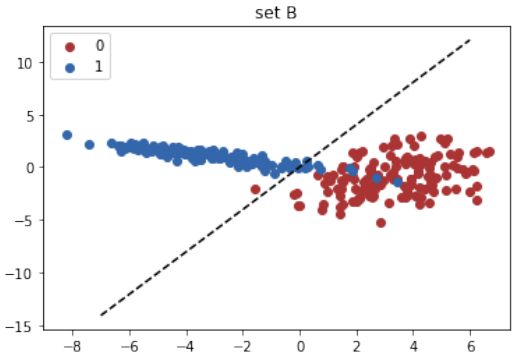
5 Set B



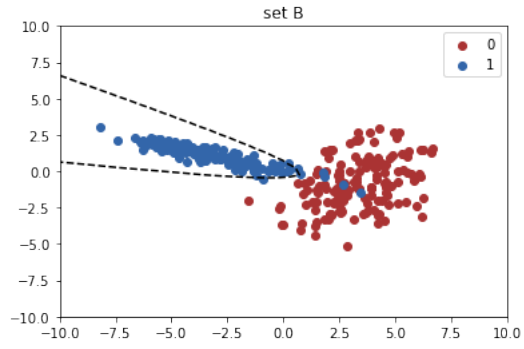
(a) LDA



(b) Logistic regression



(c) Linear regression



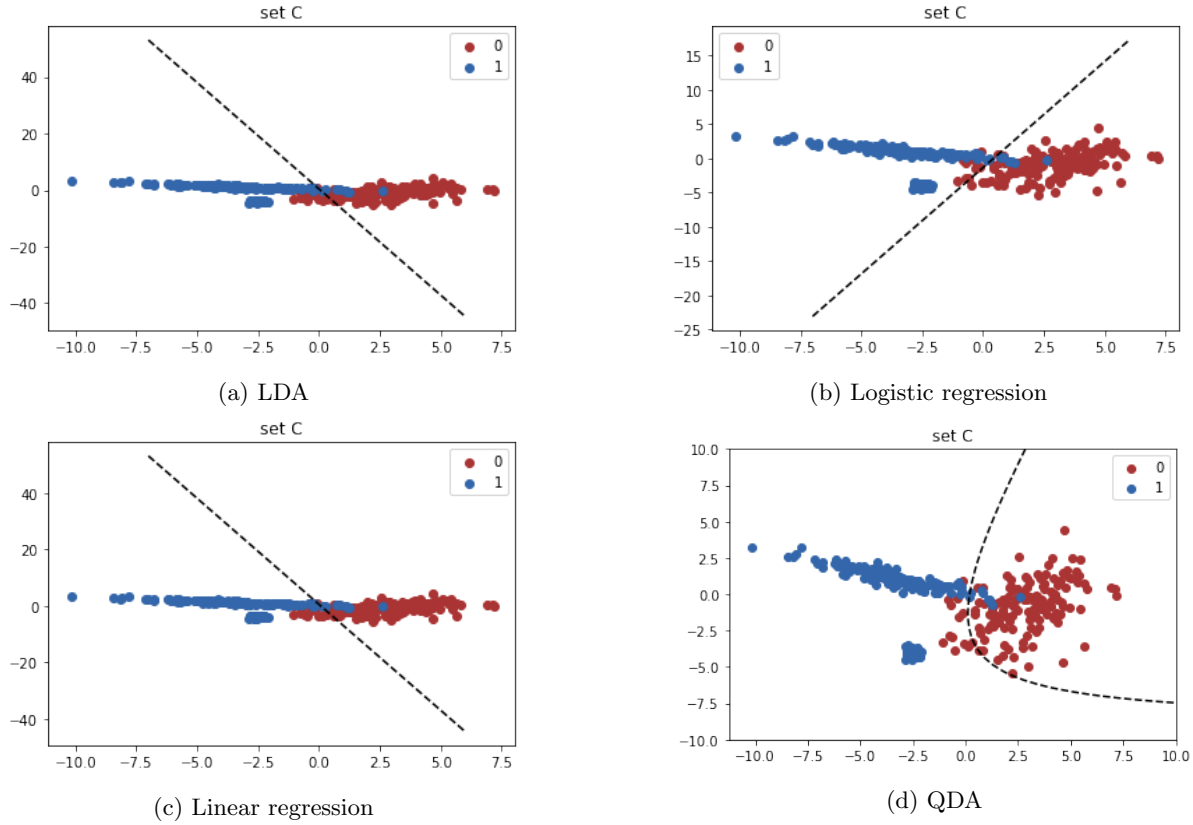
(d) QDA

Figure 2: Plot of the training data and decision boundaries.

Method	Training error	Test error
LDA	0.0300	0.0415
Logistic regression	0.0200	0.0430
Linear regression	0.0300	0.0415
QDA	0.0233	0.0230

Comments: This time we have 900 data points for training and 6000 to test. QDA is once again the best model in terms of training and test error (error rate didn't change from training to testing). This time the difference compared to LDA is quite noticeable, which means that it's better to give up the assumption $\Sigma_1 = \Sigma_2$.
We also notice that the performance of LDA and linear regression are very close, this holds for the 3rd data set as well.

6 Set C



(a) LDA

(b) Logistic regression

(c) Linear regression

(d) QDA

Figure 3: Plot of the training data and decision boundaries.

Method	Training error	Test error
LDA	0.0550	0.0423
Logistic regression	0.0400	0.0227
Linear regression	0.0550	0.0423
QDA	0.0525	0.0403

Comments: This set contains 1200 data points for training and 9000 to test. the performance of LDA, linear regression, and QDA are very similar. This time the data has two clusters of class $y = 1$, we can see on (d) how the small roundish cluster prevents QDA from fitting a quadratic boundary around the other cluster without including outliers as in set B.
Logistic regression yields the best result both in terms of training and test, it can be seen on the graph, as it fits the form of cluster 0 better compared to LDA and linear regression that cut through it.

7 Detailed proofs:

7.1 Exercise 1: Learning in discrete graphical models

The likelihood of the set of observations is:

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

We define:

$$n_m \coloneqq \sum_{i=1}^{n} \mathbb{1}_{\{z_i = m\}} \quad \text{and } n_{m,k} \coloneqq \sum_{i=1}^{n} \mathbb{1}_{\{z_i = m\}} \mathbb{1}_{\{x_i = k\}}$$

The log likelihood is:

$$\log(\ell(\pi, \theta)) = \sum_{m=1}^{M} n_m \log(\pi_m) + \sum_{m=1}^{M} \sum_{k=1}^{K} n_{m,k} \log(\theta_{mk})$$

$-\log(\ell(\pi, \theta))$ is a convex function of π and θ and the constraints are linear.
The Lagrangian of the minimization problem is:

$$L(\pi, \theta, \lambda, \mu) = -\log(\ell(\pi, \theta)) + \lambda(\sum_{m=1}^{M} \pi_m - 1) + \sum_{m=1}^{M} \mu_m (\sum_{k=1}^{K} \theta_{mk} - 1)$$

We find the optimal points by calculating its gradient:

$$\frac{\partial L(\pi, \theta, \lambda, \mu)}{\partial \pi_m} = -\frac{n_m}{\pi_m} + \lambda \quad \Rightarrow n_m / \pi_m \text{ is constant}$$
$$\frac{\partial L(\pi, \theta, \lambda, \mu)}{\partial \theta_{mk}} = -\frac{n_{mk}}{\theta_{mk}} + \mu_m \Rightarrow n_{mk} / \theta_{mk} \text{ is constant for } m \in \{1, \dots M\}$$

This leads to:

$$\hat{\pi}_m = \frac{n_m}{n}$$
$$\hat{\theta}_{mk} = \frac{n_{mk}}{n_m}$$

7.2 Exercise 2.1(a): Generative model (LDA)

We define $\theta \coloneqq (\pi, \mu_0, \mu_1, \Sigma)$, $\pi_0 \coloneqq \pi$, $\pi_1 \coloneqq 1 - \pi$, and $y_i^j \coloneqq 1$ if $y_i = j$ and 0 otherwise.

$$\begin{aligned}
\log(\ell(\pi, \mu_0, \mu_1, \Sigma)) &= \sum_{i=1}^{n} \log p_\theta(x_i, y_i) \\
&= \sum_{i=1}^{n} \log(p_\theta(y_i) p_\theta(x_i | y_i)) \\
&= \sum_{i=1}^{n} \log(p_\theta(y_i)) + \log(p_\theta(x_i | y_i)) \\
&= \sum_{i=1}^{n} \sum_{j \in \{0,1\}} y_i^j \log(\pi_j) + \sum_{i=1}^{n} \sum_{j \in \{0,1\}} y_i^j \log(\mathcal{N}(x_i | \mu_j, \Sigma)) \\
&= \sum_{j \in \{0,1\}} n_j \log(\pi_j) + \sum_{i=1}^{n} \sum_{j \in \{0,1\}} y_i^j \log(\mathcal{N}(x_i | \mu_j, \Sigma))
\end{aligned}$$

Where $n_j \coloneqq \sum_{i=1}^{n} y_i^j$. Maximizing the log likelihood leads to separate maximization of:

$$\begin{aligned}
&\sum_{j\in\{0,1\}} n_j \log(\pi_j) && \text{for } \pi \;\Rightarrow \hat{\pi} = \frac{n_1}{n_0+n_1} = \frac{\sum_{i=1}^{n} y_i}{n} \\
&\sum_{i=1}^{n} y_i^0 \log(\mathcal{N}(x_i|\mu_0,\Sigma)) && \text{for } \mu_0 \Rightarrow \hat{\mu}_0 = \frac{1}{n_0}\sum_{i=1}^{n}(1-y_i)x_i \\
&\sum_{i=1}^{n} y_i^1 \log(\mathcal{N}(x_i|\mu_1,\Sigma)) && \text{for } \mu_1 \Rightarrow \hat{\mu}_1 = \frac{1}{n_1}\sum_{i=1}^{n} y_i x_i \\
&\sum_{j\in\{0,1\}} y_i^j \log(\mathcal{N}(x_i|\mu_j,\Sigma)) && \text{for } \Sigma \;\Rightarrow \hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{\mu}_{y_i})(x_i-\hat{\mu}_{y_i})^T
\end{aligned}$$

This separation of variables helps use the results from MLE of a Gaussian distribution directly (same applies for QDA).