MVA - Probabilistic Graphical Models
DM1

Ariane ALIX, Sacha BOZOU

November 22nd, 2019

---

1 Learning in discrete graphical models

Consider the following model: z and x are discrete variables taking respectively M and K different values with $p(z = m) = \pi_m$ and $p(x = k|z = m) = \theta_{mk}$.

Let $\{(z_1, x_1)..., (z_n, x_n)\}$ be a sample of n observations. Since they are i.i.d, we have the likelihood function :

$$\begin{aligned}
L(\pi, \theta) &= \prod_{i=1}^{n} p(z_i, x_i|\pi; \theta) \\
&= \prod_{i=1}^{n} p(x_i|z_i; \pi; \theta)p(z_i|\pi) \text{ using Bayes' rule} \\
&= \prod_{i=1}^{n} \theta_{z_i x_i} \pi_{z_i} \\
&= \prod_{i=1}^{n} (\prod_{m=1}^{M} \prod_{k=1}^{K} \theta_{mk}^{\mathbb{1}_{\{z_i=m\}} \mathbb{1}_{\{x_i=k\}}})(\prod_{m=1}^{M} \pi_m^{\mathbb{1}_{\{z_i=m\}}})
\end{aligned}$$

We intoduce the variables $z_{im} = \mathbb{1}_{\{z_i=m\}}$ and $x_{ik} = \mathbb{1}_{\{x_i=k\}}$ to simplify the notations.
When passing to the log, we then get the following log-likelihood function:

$$\begin{aligned}
l(\pi, \theta) &= \sum_{i=1}^{n} (\sum_{m=1}^{M} \sum_{k=1}^{K} \log(\theta_{mk}^{z_{im} x_{ik}}) + \sum_{m=1}^{M} \log(\pi_m^{z_{im}})) \\
&= \sum_{m=1}^{M} \sum_{k=1}^{K} (\sum_{i=1}^{n} z_{im} x_{ik}) \log(\theta_{mk}) + \sum_{m=1}^{M} (\sum_{i=1}^{n} z_{im}) \log(\pi_m)
\end{aligned}$$

Our goal is to maximize this function $l(\pi, \theta)$ while respecting the constraints on the probabilities :

- $\sum_{m=1}^{M} \pi_m = 1$
- $\forall m \in \{1, ..., M\} \quad \sum_{k=1}^{K} \theta_{mk} = 1$

The two terms of the sum in $l(\pi, \theta)$ are independant, therefore we can maximize them separately.

MLE for π

We consider the problem:

$$\min_{\pi} - \sum_{m=1}^{M} (\sum_{i=1}^{n} z_{im}) \log(\pi_m)$$
$$\text{s.t} \quad \sum_{m=1}^{M} \pi_m = 1$$

The Langrangian of the problem is :

$$\mathcal{L}(\pi, \lambda) = - \sum_{m=1}^{M} (\sum_{i=1}^{n} z_{im}) \log(\pi_m) + \lambda(\sum_{m=1}^{M} \pi_m - 1)$$

And the dual function is :

$$g(\lambda) = \min_{\pi} \mathcal{L}(\pi, \lambda)$$

Since $\mathcal{L}(\pi, \lambda)$ is convex in π, we can find its minimum with respect to π by looking at the gradients with respect to the components of π :

$$\frac{\partial \mathcal{L}}{\partial \pi_m} = - \frac{\sum_{i=1}^{n} z_{im}}{\pi_m} + \lambda$$

Which is equal to 0 for $\pi_m = \frac{\sum_{i=1}^{n} z_{im}}{\lambda}$. To find λ, we look at the constraint that gives us :

$$\sum_{m=1}^{M} \frac{\sum_{i=1}^{n} z_{im}}{\lambda} = 1$$

Hence $\lambda = n$ and the solution is $\pi_m = \frac{\sum_{i=1}^{n} z_{im}}{n}$ with $\sum_{i=1}^{n} z_{im}$ the number of observations of z that are equal to m.

MLE for θ

We consider the problem:

$$\min_{\theta} - \sum_{m=1}^{M} \sum_{k=1}^{K} (\sum_{i=1}^{n} z_{im} x_{ik}) \log(\theta_{mk})$$
$$\text{s.t} \quad \forall m \in \{1, ..., M\} \quad \sum_{k=1}^{K} \theta_{mk} = 1$$

The Langrangian of the problem is :

$$\mathcal{L}(\theta, \lambda) = - \sum_{m=1}^{M} \sum_{k=1}^{K} (\sum_{i=1}^{n} z_{im} x_{ik}) \log(\theta_{mk}) + \sum_{m=1}^{M} \lambda_m (\sum_{m=1}^{M} \theta_{mk} - 1)$$

And the dual function is :

$$g(\lambda) = \min_{\theta} \mathcal{L}(\theta, \lambda)$$

Since $\mathcal{L}(\pi, \lambda)$ is convex in θ, we can find its minimum with respect to θ by looking at the gradients with respect to the components of θ :

$$\frac{\partial \mathcal{L}}{\partial \theta_{mk}} = -\frac{\sum_{i=1}^{n} z_{im} x_{ik}}{\theta_{mk}} + \lambda_m.$$

Which is equal to 0 for $\theta_{mk} = \frac{\sum_{i=1}^{n} z_{im} x_{ik}}{\lambda_m}$.

To find the λ_m, we look at the constraints that give us :

$$\sum_{k=1}^{K} \frac{\sum_{i=1}^{n} z_{im} x_{ik}}{\lambda_m} = 1$$

Hence $\lambda_m = \sum_{i=1}^{n} z_{im}$ and the solution is $\theta_{mk} = \frac{\sum_{i=1}^{n} z_{im} x_{ik}}{\sum_{i=1}^{n} z_{im}}$ with $\sum_{i=1}^{n} z_{im}$ the number of observations of z that are equal to m and $\sum_{i=1}^{n} z_{im} x_{ik}$ the number of observations where z is equal to m and x is equal to k simulatenously.

2 Linear classification

2.1 Generative model (LDA)

a. Let $\{(x_1, y_1)..., (x_n, y_n)\}$ be a sample of n observations with the x_i in $\mathbb{R}^2$ and the y_i in $\{0, 1\}$. Since they are i.i.d, we have the likelihood function :

$$\begin{aligned}
L(\pi, \mu_0, \mu_1, \Sigma) &= \prod_{i=1}^{n} p(x_i, y_i | \pi, \mu_0, \mu_1, \Sigma) \\
&= \prod_{i=1}^{n} p(x_i | y_i; \pi, \mu_0, \mu_1, \Sigma) p(y_i | \pi) \text{ using Bayes' rule} \\
&= \prod_{i=1}^{n} \pi^{y_i} (1-\pi)^{1-y_i} f_{\mu_{y_i}}(x_i)
\end{aligned}$$

Where $f_{\mu_{y_i}}(x_i) = \frac{1}{2\Pi\sqrt{\det \Sigma}} \exp(-\frac{1}{2}(x - \mu_{y_i})^T \Sigma^{-1} (x - \mu_{y_i}))$. To simplify we can write :

$$L(\pi, \mu_0, \mu_1, \Sigma) = \prod_{i=1}^{n} \pi^{y_i} (1-\pi)^{1-y_i} f_{\mu_0}(x_i)^{1-y_i} f_{\mu_1}(x_i)^{y_i}$$

And we get the log-likelihood :

$$\begin{aligned}
l(\pi, \mu_0, \mu_1, \Sigma) = \sum_{i=1}^{n} &(y_i \log \pi + (1 - y_i) \log(1 - \pi) \\
&+ (1 - y_i)(f_{\mu_0}(x_i)) \\
&+ y_i (f_{\mu_1}(x_i))
\end{aligned}$$

$$\begin{aligned} l(\pi, \mu_0, \mu_1, \Sigma) = &\sum_{i=1}^{n} (y_i \log \pi + (1 - y_i) \log(1 - \pi) \\ &+ (1 - y_i)(-\log(2\Pi) - \frac{1}{2} \log(\det \Sigma) - \frac{1}{2}(x_i - \mu_0)^T \Sigma^{-1}(x_i - \mu_0)) \\ &+ y_i(-\log(2\Pi) - \frac{1}{2} \log(\det \Sigma) - \frac{1}{2}(x_i - \mu_1)^T \Sigma^{-1}(x_i - \mu_1)) \end{aligned}$$

Our goal is to maximize this function, so we look at the gradients with respect to the parameters to find for which they are equal to 0:

For π.

$$\begin{aligned} &\frac{\partial l}{\partial \pi}(\pi, \mu_0, \mu_1, \Sigma) = 0 \\ &\Leftrightarrow \sum_{i=1}^{n} \frac{y_i}{\pi} - \frac{1 - y_i}{1 - \pi} = 0 \\ &\Leftrightarrow \frac{1}{\pi} \sum_{i=1}^{n} y_i = \frac{1}{1 - \pi} \sum_{i=1}^{n} 1 - y_i \\ &\Leftrightarrow (\frac{1}{\pi} + \frac{1}{1 - \pi}) \sum_{i=1}^{n} y_i = \frac{n}{1 - \pi} \\ &\Leftrightarrow (\frac{1 - \pi}{\pi} + 1) \sum_{i=1}^{n} y_i = n \\ &\Leftrightarrow \boxed{\hat{\pi} = \frac{\sum_{i=1}^{n} y_i}{n}} \end{aligned}$$

For μ_0.

$$\begin{aligned} &\frac{\partial l}{\partial \mu_0}(\pi, \mu_0, \mu_1, \Sigma) = 0 \\ &\Leftrightarrow -(\sum_{i=1}^{n} (1 - y_i) \Sigma^{-1}(x_i - \mu_0)) = 0 \\ &\Leftrightarrow \sum_{i=1}^{n} x_i(1 - y_i) - \mu_0(1 - y_i) = 0 \\ &\Leftrightarrow \boxed{\hat{\mu_0} = \frac{\sum_{i=1}^{n} x_i(1 - y_i)}{\sum_{i=1}^{n}(1 - y_i)}} \end{aligned}$$

For μ_1.

$$\begin{aligned}
&\frac{\partial l}{\partial \mu_1}(\pi, \mu_0, \mu_1, \Sigma) = 0 \\
&\Leftrightarrow -(\sum_{i=1}^{n}(y_i)\Sigma^{-1}(x_i - \mu_1)) = 0 \\
&\Leftrightarrow \sum_{i=1}^{n} x_i y_i - \mu_1 y_i = 0 \\
&\Leftrightarrow \boxed{\hat{\mu_1} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} y_i}}
\end{aligned}$$

For Σ.

$$\begin{aligned}
&\frac{\partial l}{\partial \Sigma^{-1}}(\pi, \mu_0, \mu_1, \Sigma) = 0 \\
&\Leftrightarrow \frac{\partial}{\partial \Sigma^{-1}}(\sum_{i=1}^{n} \frac{(1-y_i)}{2}(\log(\det \Sigma^{-1}) - Tr((x_i - \mu_0)^T \Sigma^{-1}(x_i - \mu_0))) \\
&\quad + \frac{y_i}{2}(\log \det \Sigma^{-1} - Tr((x_i - \mu_1)^T \Sigma^{-1}(x_i - \mu_1)))) = 0 \\
&\Leftrightarrow \sum_{i=1}^{n} \frac{1-y_i}{2}(\Sigma - (x_i - \mu_0)(x_i - \mu_0)^T) + \frac{y_i}{2}(\Sigma - (x_i - \mu_1)(x_i - \mu_1)^T) = 0 \\
&\Leftrightarrow \boxed{\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(1-y_i)(x_i - \mu_0)(x_i - \mu_0)^T + y_i(x_i - \mu_1)(x_i - \mu_1)^T}
\end{aligned}$$

b. We aim to determine the form of $p(y = 1|x)$. By applying Bayes' rule, we have :

$$\begin{aligned}
\mathbb{P}(Y = 1|X = x) &= \frac{\mathbb{P}(Y = 1, X = x)}{\mathbb{P}(X = x)} \\
&= \frac{\mathbb{P}(X = x|Y = 1)\mathbb{P}(Y = 1)}{\mathbb{P}(X = x)} \\
&= \frac{\mathbb{P}(X = x|Y = 1)\mathbb{P}(Y = 1)}{\mathbb{P}(X = x|Y = 1)\mathbb{P}(Y = 1) + \mathbb{P}(X = x|Y = 0)\mathbb{P}(Y = 0)} \\
&= \frac{f_{\mu_1}(x)\pi}{f_{\mu_1}(x)\pi + f_{\mu_0}(x)(1-\pi)} \\
&= \frac{1}{1 + \frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi}}
\end{aligned}$$

Let's look at $\frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi}$.

$$
\begin{aligned}
\frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi} &= \frac{1-\pi}{\pi}\frac{\frac{1}{2\Pi\sqrt{\det\Sigma}}\exp(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0))}{\frac{1}{2\Pi\sqrt{\det\Sigma}}\exp(-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1))} \\
&= \frac{1-\pi}{\pi}\exp(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)+\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)) \\
&= \frac{1-\pi}{\pi}\exp(-\frac{1}{2}(x^T\Sigma^{-1}x+\mu_0^T\Sigma^{-1}\mu_0)+\mu_0^T\Sigma^{-1}x+\frac{1}{2}(x^T\Sigma^{-1}x+\mu_1^T\Sigma^{-1}\mu_1)-\mu_1^T\Sigma^{-1}x) \\
&= \frac{1-\pi}{\pi}\exp((\mu_0-\mu_1)^T\Sigma^{-1}x+\frac{1}{2}(\mu_1^T\Sigma^{-1}\mu_1-\mu_0^T\Sigma^{-1}\mu_0)) \\
&= \exp((\mu_0-\mu_1)^T\Sigma^{-1}x+\frac{1}{2}(\mu_1^T\Sigma^{-1}\mu_1-\mu_0^T\Sigma^{-1}\mu_0)+\log(\frac{1-\pi}{\pi})) \\
&= \exp(-(\omega^Tx+b))
\end{aligned}
$$

Where $\omega = \Sigma^{-1}(\mu_1-\mu_0)$
And $b = -\frac{1}{2}(\mu_1^T\Sigma^{-1}\mu_1-\mu_0^T\Sigma^{-1}\mu_0)-\log(\frac{1-\pi}{\pi})$
Therefore we have :

$$
\begin{aligned}
\mathbb{P}(Y=1|X=x) &= \frac{1}{1+\frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi}} \\
&= \frac{1}{1+\exp(-(\omega^Tx+b))} \\
&= \sigma(\omega^Tx+b)
\end{aligned}
$$

Which is similar to the form of the logistic regression.

c. The MLE has been implemented (cf. the Jupyter Notebook file *MVA DM1 Ariane ALIX Sacha BOZOU.ipynb*), applied to the datasets, and used to to plot a decision boundary corresponding to $p(y=1|x)=0.5$:

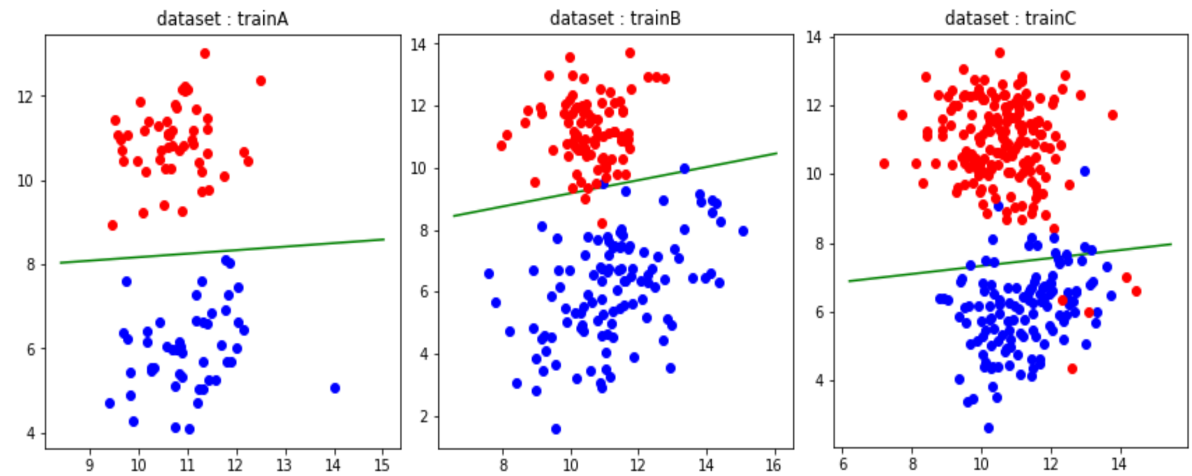

Figure 1 – Point cloud and decision boundary for the LDA

2.2 Logistic regression

In both logistic and linear regressions, we will use offest reparametrization.

a. For the logistic regression, we assume that :

$$ln(\frac{\mathbb{P}(Y=1|X=x)}{\mathbb{P}(Y=0|X=x)}) = \omega^T x$$

Equivalently :

$$\mathbb{P}(Y=1|X=x) = \sigma(\omega^T x)$$

with σ the sigmoid function

We iterate on the training data and we follow :

$$\omega^{new} = \omega^{old} + (X^T D^{-1}_{\eta^{old}} X X^T (Y - \eta^{old})$$

where : $\eta_i = \sigma(\omega^T x_i)$
and : $D_\eta = Diag(\eta_i(1-\eta_i))$

The decision boundary is defined by $\omega^T x = 0$

We give here after the numerical values for the parameters w and b learnt by the model on the different datasets.

- for trainA : w : 14.97 , -59.05 ; b : 339.41
- for train B : w : 1.84 , -3.71 ; b : 13.43
- for train C : w : -0.28 , -1.91 ; b : 18.81

b. The model has been implemented (cf. the Jupyter Notebook), applied to the datasets, and used to plot a decision boundary corresponding to $p(y=1|x) = 0.5$:

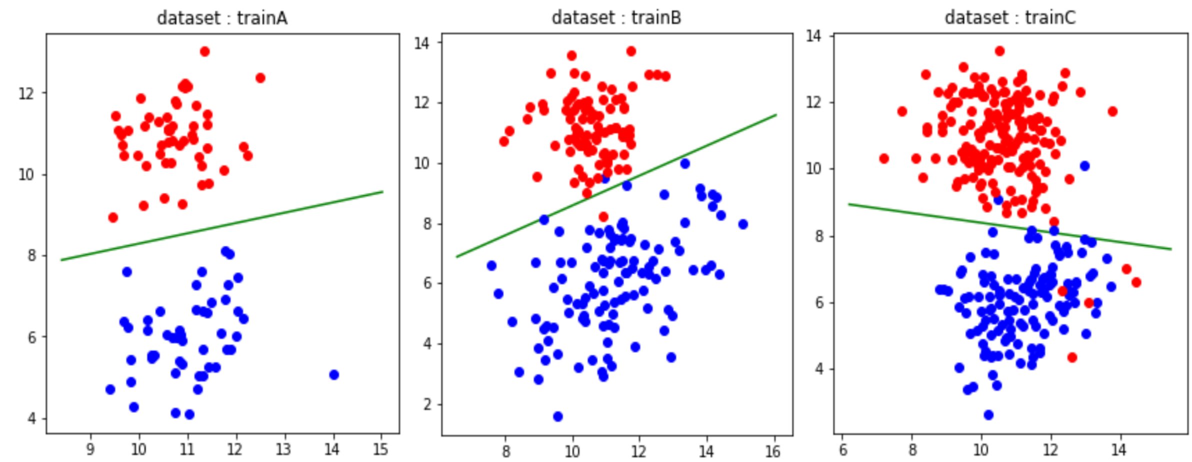

Figure 2 – Point cloud and decision boundary for the Logistic regression

2.3 Linear Regression

a. The probabilistic linear regression can be written with a noise ϵ as :

$$Y = w^T X + \epsilon \quad \text{with} \quad \epsilon \sim \mathcal{N}(0, \sigma^2)$$

And we have the probability law:

$$p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(y - w^T x)^2}{2\sigma^2})$$

The normal equation aims to minimize the cost defined by $\frac{1}{2\sigma^2}\|Xw - y\|_2^2 = (Xw - y)^T(Xw - y)$ with regard to w. Looking at its derivative in w, we have :

$$\begin{aligned}
\frac{\partial \text{cost}}{\partial w}(w) = 0 &\Leftrightarrow 2X^T Xw - 2X^T y = 0 \\
&\Leftrightarrow X^T Xw = X^T y \\
&\Leftrightarrow (X^T X)^{-1} X^T Xw = (X^T X)^{-1} X^T y \\
&\Leftrightarrow w = (X^T X)^{-1} X^T y
\end{aligned}$$

Therefore $\hat{w}_{MLE} = (X^T X)^{-1} X^T y$.

Moreover, $\hat{\sigma}^2_{MLE}$ should minimize the log-likelihood defined by :

$$\frac{1}{2} n \log(2\pi\sigma^2) + \frac{1}{2} \sum_{i=1}^{n} \frac{(y_i - w^T x_i)^2}{\sigma^2}$$

Hence, $\hat{\sigma}^2_{MLE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{w}^T_{MLE} x_i)^2$

For the decision boundary, we have :

$$\begin{aligned}
p(y = 1|x) = 0.5 &\Leftrightarrow \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(1 - w^T x)^2}{2\sigma^2}) = 0.5 \\
&\Leftrightarrow -\frac{1}{2} \log(2\pi\sigma^2) - \frac{(1 - w^T x)^2}{2\sigma^2} = \log(\frac{1}{2}) \\
&\Leftrightarrow \frac{(1 - w^T x)^2}{2\sigma^2} = \frac{1}{2} \log(\frac{2}{\pi\sigma^2}) \\
&\Leftrightarrow (1 - w^T x) = \sqrt{\sigma^2 \log(\frac{2}{\pi\sigma^2})} \\
&\Leftrightarrow w^T x - 1 + \sqrt{\sigma^2 \log(\frac{2}{\pi\sigma^2})} = 0
\end{aligned}$$

We give here after the numerical values for the parameters w, b and σ^2 (the variance of the noise) learnt by the model on the different datasets.

- for trainA : w : 0.06 , -0.18 ; b : 1.38 ; σ^2=0.03
- for train B : w : 0.08 , -0.15 ; b : 0.88 ; σ^2=0.05
- for train C : w : 0.02 , -0.16 ; b : 1.64 ; σ^2=0.06

b. The model has been implemented (cf. the Jupyter Notebook), applied to the datasets, and used to plot a decision boundary corresponding to $p(y = 1|x) = 0.5$:

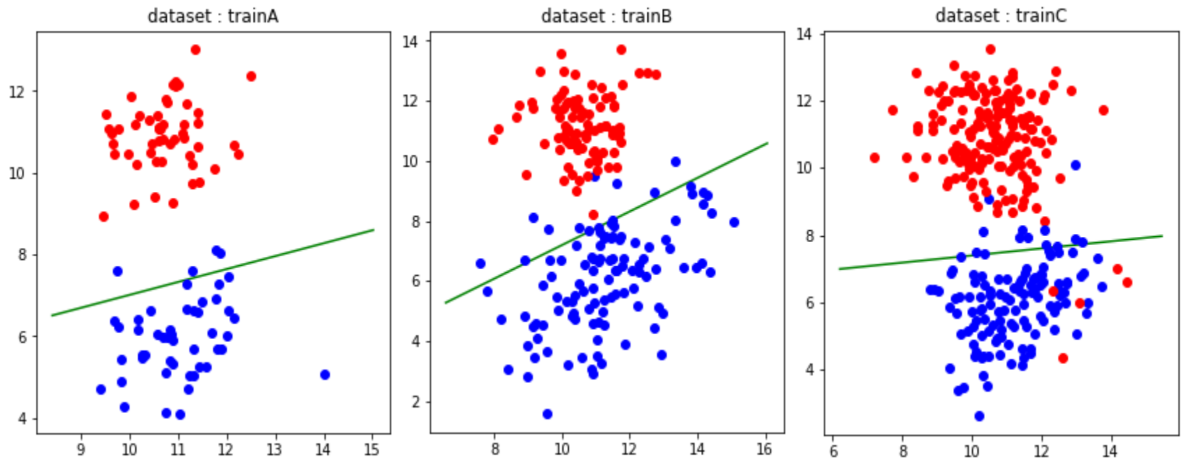

Figure 3 – Point cloud and decision boundary for the Linear regression

2.4 Application

a.

We have (cf computation on Jupyter notebook)

- MLE on Generative Model LDA
  Error of classification for dataset trainA : 0.00%
  Error of classification for dataset testA : 1.00%

  Error of classification for dataset trainB : 2.00%
  Error of classification for dataset testB : 4.00%

  Error of classification for dataset trainC : 6.33%
  Error of classification for dataset testC : 7.33%

- Logistic Regression

Error of classification for dataset trainA : 0.00%
Error of classification for dataset testA : 1.00%

Error of classification for dataset trainB : 1.00%
Error of classification for dataset testB : 3.50%

Error of classification for dataset trainC : 3.00%
Error of classification for dataset testC : 4.67%

- Linear Regression

2.5 Generative model (LDA)

The Maximum Likelihood Estimators for the parameters are the same as in the LDA, except for Σ.

We now have this log-likelihood:

$$\begin{aligned} l(\pi, \mu_0, \mu_1, \Sigma_0, \Sigma_1) = \sum_{i=1}^{n} (y_i \log \pi + (1 - y_i) \log(1 - \pi) \\ + (1 - y_i)(f_{\mu_0}(x_i)) \\ + y_i(f_{\mu_1}(x_i)) \end{aligned}$$

Where $f_{\mu_{y_i}}(x_i) = \frac{1}{2\Pi\sqrt{\det \Sigma_{y_i}}} \exp(-\frac{1}{2}(x - \mu_{y_i})^T \Sigma_{y_i}^{-1}(x - \mu_{y_i}))$.

And we already have the MLEs :

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

$$\boxed{\hat{\mu_0} = \frac{\sum_{i=1}^{n} x_i(1 - y_i)}{\sum_{i=1}^{n}(1 - y_i)}}$$

$$\boxed{\hat{\mu_1} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} y_i}}$$

We now have to find $\hat{\Sigma_0}$ and $\hat{\Sigma_1}$.

$$
\begin{aligned}
&\frac{\partial l}{\partial \Sigma_0^{-1}}(\pi, \mu_0, \mu_1, \Sigma_0, \Sigma_1) = 0 \\
&\Leftrightarrow \frac{\partial}{\partial \Sigma_0^{-1}}(\sum_{i=1}^{n} \frac{(1-y_i)}{2}(\log(\det \Sigma_0^{-1}) - Tr((x_i - \mu_0)^T \Sigma_0^{-1}(x_i - \mu_0))) = 0 \\
&\Leftrightarrow \sum_{i=1}^{n} \frac{1-y_i}{2}(\Sigma_0 - (x_i - \mu_0)(x_i - \mu_0)^T) = 0 \\
&\Leftrightarrow \sum_{i=1}^{n}(1-y_i)\Sigma_0 = \sum_{i=1}^{n}(1-y_i)(x_i - \mu_0)(x_i - \mu_0)^T) \\
&\Leftrightarrow \boxed{\hat{\Sigma_0} = \frac{1}{\sum_{i=1}^{n}(1-y_i)} \sum_{i=1}^{n}(1-y_i)(x_i - \mu_0)(x_i - \mu_0)^T}
\end{aligned}
$$

And with the same reasoning we get :

$$
\boxed{\hat{\Sigma_1} = \frac{1}{\sum_{i=1}^{n} y_i} \sum_{i=1}^{n} y_i(x_i - \mu_1)(x_i - \mu_1)^T}
$$

Follwing the same logic as for the boundary of the LDA :

$$
\mathbb{P}(Y = 1|X = x) = \frac{1}{1 + \frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi}}
$$

With :

$$
\begin{aligned}
\frac{f_{\mu_0}(x)(1-\pi)}{f_{\mu_1}(x)\pi} &= \frac{1-\pi}{\pi} \frac{\frac{1}{2\Pi\sqrt{\det \Sigma_0}} \exp(-\frac{1}{2}(x-\mu_0)^T \Sigma_0^{-1}(x-\mu_0))}{\frac{1}{2\Pi\sqrt{\det \Sigma_1}} \exp(-\frac{1}{2}(x-\mu_1)^T \Sigma_1^{-1}(x-\mu_1))} \\
&= \frac{(1-\pi)\sqrt{det\Sigma_1}}{\pi\sqrt{det\Sigma_0}} \exp(-\frac{1}{2}(x-\mu_0)^T \Sigma_0^{-1}(x-\mu_0) + \frac{1}{2}(x-\mu_1)^T \Sigma_1^{-1}(x-\mu_1)) \\
&= \exp(-(x^T K x + \omega^T x + b))
\end{aligned}
$$

Where : $\omega = \Sigma_1^{-1}\mu_1 - \Sigma_0^{-1}\mu_0$
And $b = -\frac{1}{2}(\mu_1^T \Sigma_1^{-1}\mu_1 - \mu_0^T \Sigma_0^{-1}\mu_0) - \log(\frac{(1-\pi)\sqrt{det\Sigma_1}}{\pi\sqrt{det\Sigma_0}})$
And $K = \frac{1}{2}\Sigma_0^{-1} - \frac{1}{2}\Sigma_1^{-1}$

The decision boundary is defined by :

$$
x^T K x + \omega^T x + b = 0
$$

a. We implemented the new model in the Jupyter Noteboook.

We give here after the numerical values for the parameters w, b and K learnt by the model on the different datasets.

- for trainA : w : -9.38 , -5.91 ; b : 79.95 ; K= $\begin{vmatrix} 0.51 & -0.07 \\ -0.07 & 0.33 \end{vmatrix}$
- for train B : w : -5.07 , -3.97 ; b : 46.86 ; K= $\begin{vmatrix} 0.25 & -0.01 \\ -0.01 & 0.18 \end{vmatrix}$
- for train C : w : -5.07 , -4.01 ; b : 46.38 ; K= $\begin{vmatrix} 0.19 & 0.05 \\ 0.05 & 0.13 \end{vmatrix}$

c. We observe:

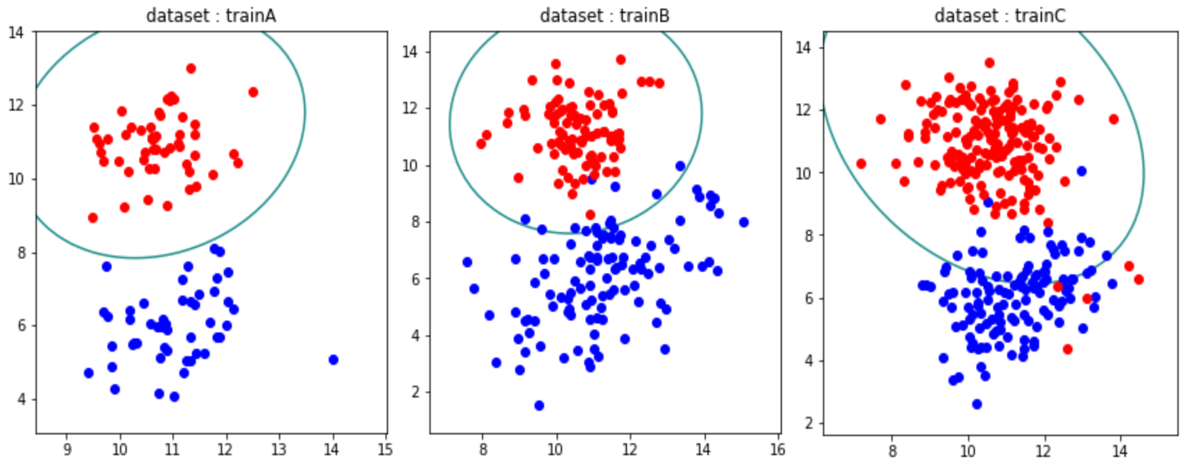

Figure 3 – Point cloud and decision boundary for the Linear regression

c. We get the following missclassification errors :

Error of classification for dataset trainA : 0.00%
Error of classification for dataset testA : 2.00%

Error of classification for dataset trainB : 5.50%
Error of classification for dataset testB : 7.00%

Error of classification for dataset trainC : 14.00%
Error of classification for dataset testC : 16.67%

d.