SY09 - TP3 Discrimination, theorie bayesienne de la decision

Matthieu PETIT - Pierre-Louis LACORTE

6 juin 2017

Objectif

Le but de ce TP est de réaliser une application mettant en œuvre la méthode d'apprentissage supervisé ainsi qu'une utilisation de la théorie bayésienne de la décision. La méthode d'apprentissage supervisé permet de réaliser des prédictions à partir d'observation. Pour ceci nous allons tenter de déterminer la classe à laquelle appartiennent certains individus à partir d'observations sur certains d'entre eux. Cette discrémination va être réalisée au moyen de deux classifieurs : le classifieur euclidien et par la méthode des K plus proches voisins.

Table des matières

1 Classifieur euclidien, K plus proches voisins **1**
1.1 Programmation 1
1.2 Evaluation des performances 2

2 Regle de Bayes **7**
2.1 Distributions marginales 7
2.2 Courbes d'iso-densité 8
2.3 Expression de la règle de Bayes et frontière de décision 9
2.4 Erreur de Bayes 11

1 Classifieur euclidien, K plus proches voisins

1.1 Programmation

Nous avons, dans le cadre de ce TP, réalisé quatre fonctions. Les premières utilisent la méthode ddes classifieurs euclidiens, une faisant l'apprentissage des paramètres et l'autre le classement des individus. Les autres fonctions implémentent la méthode des K plus proches voisins, l'une faisant le classement des individus et la seconde détermine le nombre optimal de voisins.

1.1.1 Test des fonctions

Afin de tester les fonctions précédement réalisées et de visualiser les frontières de décision ainsi définie, nous utilisons les fonctions **front.ceuc** et **front.kppv** sur le jeu de donnée Synth1-40.

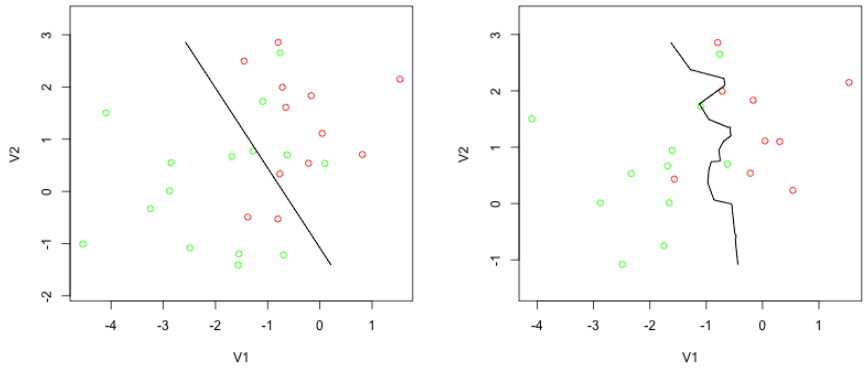

FIGURE 1 – Représentation des tests des fonctions et des classifieurs euclidien et des K plus proches voisins sur le jeu de données Synth1-40

1.2 Evaluation des performances

Maintenant que nous avons pu réaliser des tests et remarquer que nous obtenons des résultats qui semble dans l'ensemble correct. Il est intéressant d'étudier la performances de ces 2 deux classifieurs. Nous essaierons d'estimer le taux d'erreur ε. Sommairement, nous allons pour cela effectuer plusieurs séparation (20) d'un ensemble d'individus et faire une moyenne du taux d'erreur ainsi constaté.

1.2.1 Jeux de données Synth1-40, Synth1-100, Synth1-500 et Synth1-1000

Il nous est dans un premier temps demandé de déterminer pour chaque jeu de données quels sont les paramètres μ_k, Σ_k ainsi que les proportions π_k de chaque classes. Nous obtenons donc pour chaque jeu les résultats suivants :

Données	Paramètre	Classe 1	Classe 2
Synth1-40	π_k	0.45	0.55
	μ_k	$\begin{pmatrix}-0.3164892 & 1.0919580\end{pmatrix}$	$\begin{pmatrix}-1.8833879 & 0.1051202\end{pmatrix}$
	Σ_k	$\begin{pmatrix}0.6816320 & 0.1193613\\ 0.1193613 & 1.0128977\end{pmatrix}$	$\begin{pmatrix}1.375210 & 0.322732\\ 0.322732 & 1.439329\end{pmatrix}$
Synth1-100	π_k	0.54	0.46
	μ_k	$\begin{pmatrix}0.02572935 & 0.8153381\end{pmatrix}$	$\begin{pmatrix}-1.96542202 & -0.1265093\end{pmatrix}$
	Σ_k	$\begin{pmatrix}0.8816362 & -0.1312604\\ -0.1312604 & 1.1088357\end{pmatrix}$	$\begin{pmatrix}0.75697638 & -0.03763312\\ -0.03763312 & 0.76106976\end{pmatrix}$
Synth1-500	π_k	0.528	0.472
	μ_k	$\begin{pmatrix}0.1316485 & 0.87971948\end{pmatrix}$	$\begin{pmatrix}-1.8834513 & -0.08475189\end{pmatrix}$
	Σ_k	$\begin{pmatrix}1.05014864 & 0.05222139\\ 0.05222139 & 0.98388229\end{pmatrix}$	$\begin{pmatrix}0.9668693 & -0.1108938\\ -0.1108938 & 0.9794010\end{pmatrix}$
Sybnth1-1000	π_k	0.496	0.504
	μ_k	$\begin{pmatrix}-0.01279584 & 0.91568302\end{pmatrix}$	$\begin{pmatrix}-1.96120940 & 0.01834376\end{pmatrix}$
	Σ_k	$\begin{pmatrix}0.96857306 & -0.06521397\\ -0.06521397 & 1.07942842\end{pmatrix}$	$\begin{pmatrix}0.99041976 & 0.02094962\\ 0.02094962 & 0.94128568\end{pmatrix}$

TABLE 1 – Estimation des paramètres pour les jeux de données Synth1-40, Synth1-100, Synth1-500 et Synth1-1000

Il semble apparaitre dans les résultats précédents que les proportions de chacune des classes sont proches de 50%, μ_1 prend globalement la forme $\begin{pmatrix}0 & 1\end{pmatrix}$ tandis que μ_2 ressemble plutôt à $\begin{pmatrix}-2 & 0\end{pmatrix}$. Finalement Σ_1 et Σ_2 s'approchent tous deux de $\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$. Une fois l'estimation des paramètres des distributions conditionnellles et des proportions des classes réalisées nous pouvons essayer de traiter des taux d'erreur. Nous allons traiter $N = 20$ expériences et déterminer l'estimation ponctuelle du taux d'erreur ε ainsi que l'intervalle de confiance de cette erreur pour un seuil de 95% sur les estimations de l'ensemble d'apprentissage et sur l'ensemble de test. La première classification que nous effectuons est à l'aide du classificateur euclidien. Cette méthode peut être justifiée du fait de la présence de simplement deux classes qui présentent des répartitions équivalentes en volume.

Classifieur	Synth1-40	Synth1-100	Synth1-500	Synth1-1000
ε_{app}	21.48148	8.358209	13.16817	14.04048
IC_{app}	$[19.12214; 23.84082]$	$[7.359995; 9.356423]$	$[12.58195; 13.75438]$	$[13.59830; 14.48266]$
ε_{test}	17.30769	9.090909	13.20359	14.23423
IC_{test}	$[14.02992; 20.58547]$	$[7.193732; 10.988086]$	$[12.27423; 14.13296]$	$[13.37098; 15.09749]$

TABLE 2 – Estimation ponctuelle des taux d'erreur ε et intervalles de confiance sur cette erreur par la méthode du classifieur euclidien sur les jeux de données Synth1-40, Synth1-100, Synth1-500 et Synth1-1000

Tout d'abord la première chose que nous pouvons constater et que plus

l'échantillon traité est volumineux plus les résultats sont semblables pour l'ensemble d'apprentissage et l'ensemble de test. Les taux d'erreurs ponctuels restent cependant élevé puisque qu'ils ne descendent pas au dessous de 8% et peuvent même atteindre 21% Table 2. Les IC sont quand à eux de faible taille ce qui témoigne plutôt d'un taux d'erreurs "précis".

Nous pouvons venir appuyer l'estimation de nos taux d'erreurs avec la représentation graphique des jeux de données. Nous constatons en effet que le jeu de données Synth-100 est celui qui présente les individus les plus distincts (graphique en haut à droite Figure 2). Cette distance est constatée en remarquant qu'il y a le moins de "mélange" entre les points orange et les points bleus. Le taux d'erreur est donc plus faible pour ce jeu puisque les individus sont plus faciles à classer.

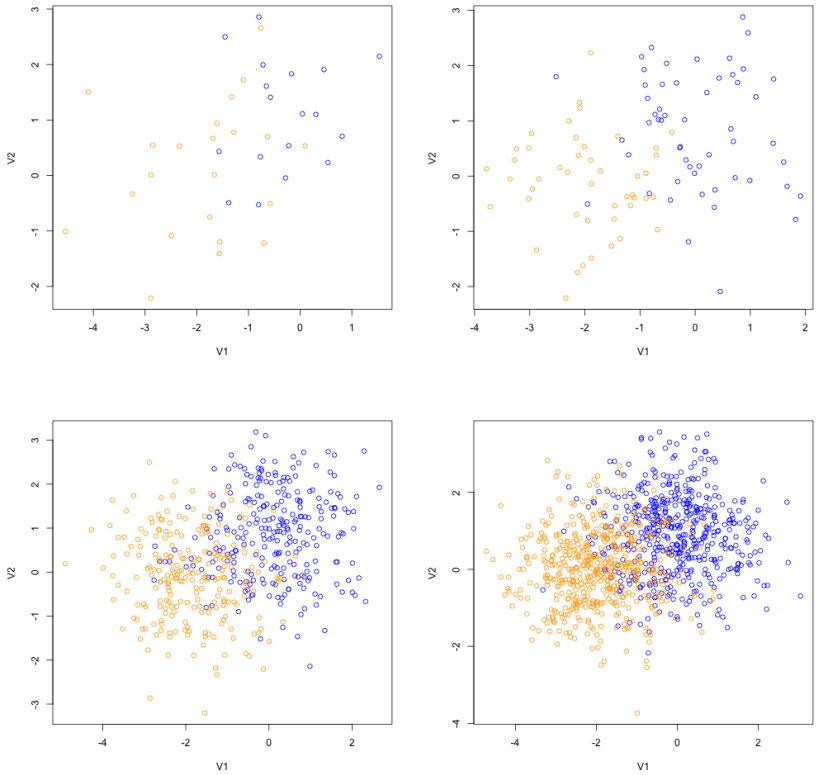

FIGURE 2 – Représentation graphique des jeux de données Synth1-n avec coloration des individus en fonction de leur classe

La seconde classification que nous traitons est celle des K plus proches voisins. Il est intéressant de reveler qu'importe le jeu de données de Synth1 sélectionné, la fonction *kppv.tune* retourne à chaque fois comme nombre de voisin optimal : 1. En effet, en utilisant l'ensemble d'apprentissage comme ensemble de

validation, l'algorithme va choisir pour chaque élément le voisin le plus proche, c'est-à-dire lui même. Nous reproduisons donc 20 séparations aléatoires sur les jeux de données et, pour chaque séparation, nous définissons le nombre optimal de voisins. À partir de la nous pouvons tester les performances du classifieur des K plus proches voisins. Finalement nous obtenons par cette méthode les taux d'erreur ε et les IC suivants :

KPPV	Synth1-40	Synth1-100	Synth1-500	Synth1-1000
ε_{app}	5.75	2.400000	1.72	1.20
IC_{app}	$[1.052952; 10.44705]$	$[0.3947401; 4.40526]$	$[0; 3.686341]$	$[8.925384; 11.794616]$
ε_{test}	18.50	7.708333	10.36	10.76
IC_{test}	$[13.180042; 23.81996]$	$[6.2552270; 9.16144]$	$[0; 2.928888]$	$[9.725855; 11.794145]$

TABLE 3 – Estimation ponctuelle des taux d'erreur ε et intervalles de confiance sur cette erreur par la méthode des K plus proches voisins sur les jeux de données Synth1-40, Synth1-100, Synth1-500 et Synth1-1000

La première constatation que nous pouvons faire est que, contrairement au classifieur euclidien, les taux d'erreurs ponctuels sur les ensembles d'apprentissage sont très inférieurs à ceux des ensembles de test. Finalement nous constatons que les résultats des taux d'erreurs sur les ensembles d'apprentissage et de test sont inférieurs par la méthodes des KPPV Table 3 que par la méthode du classifieur euclidien Table 2.

1.2.2 Jeu de données Synth2-1000

Nous considérons maintenant le jeu de données Synth2-1000 qui présente des distributions différentes des précédents jeux de données Synth-n étudiés. Noius estimimons dans un premier temps les paramètres μ_k, Σ_k ainsi que les proportions des classes π_k :

Paramètre	Classe 1	Classe 2
π_k	0.523	0.496
μ_k	$[3.018388; -0.006382269]$	$[-2.142281; -0.026524483]$
Σ_k	$\begin{bmatrix} 0.9904026 & 0.1131386 \\ 0.1131386 & 1.0928705 \end{bmatrix}$	$\begin{bmatrix} 4.4347816 & -0.1543981 \\ -0.1543981 & 1.0308554 \end{bmatrix}$

TABLE 4 – Estimation des paramètres pour le jeu de donnée Synth2-1000

Nous déterminons à nouveau les estimations ponctuelles ainsi que les intervalles de confiance sur l'ensemble d'apprentissage et de test par la méthode des classifieurs euclidiens ainsi que par la méthode des K plus proches voisins :

Synth2-1000	Classifieur euc.	KPPV
ε_{app}	6.274363	0.96
IC_{app}	$[6.072952; 6.475774]$	$[0.03357783; 1.886422]$
ε_{test}	6.486486	4.20
IC_{test}	$[6.106618; 6.866355]$	$[3.55007962; 4.849920]$

TABLE 5 – Estimation ponctuelle des taux d'erreur ε et intervalles de confiance sur cette erreur par la méthode du classifieur euclidien et des K plus proches voisins sur le jeu de données Synth2-1000

Les estimations ponctuelles et les intervalles de confiances sont tous deux petits, plus petits que pour les jeux Synth1-n, ceci peut être expliqué par le fait que les classes présentent des centres de gravité μ_k plus éloignés. Les classes sont plus éloignées l'une de l'autre. La classification en est donc facilitée.

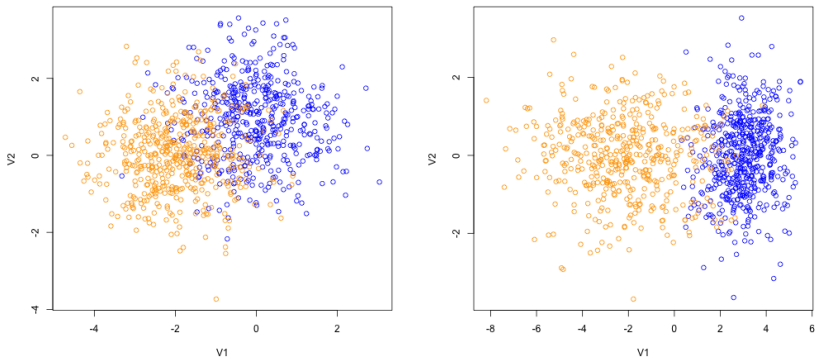

FIGURE 3 – Représentation graphique des jeux de données de Synth1-1000 (gauche) et Synth2-1000 (droite).

A nouveau, nous pouvons constater graphiquement les meilleurs résultats obtenus avec l'échantillon Synth2 que'avec Synth1 du fait d'une meilleur séparation des individus Figure 3. Cette séparation induit à nouveau une facilité de classification, et donc un taux d'erreur plus faible.

1.2.3 Jeux de données réelles

La classification dans le cadre de jeu de données test permet d'évaluer la précision d'un classifieur mais nous pouvons aussi utiliser ces classifieurs sur de vrais jeux de données. Nous allons ici tester l'efficacité des classifieurs sur les jeux de données Pima et Breastcancer.

Pima	Classifieur euc.	KPPV
ε_{app}	24.76056	1.390977
IC_{app}	$[24.17113; 25.34999]$	$[0.00000; 3.407674]$
ε_{test}	24.46328	15.375940
IC_{test}	$[22.99208; 25.93448]$	$[13.72249; 17.480519]$

TABLE 6 – Estimation ponctuelle des taux d'erreur ε et intervalles de confiance sur cette erreur par la méthode du classifieur euclidien et des K plus proches voisins sur le jeu de données Pima

Breastcancer	Classifieur euc.	KPPV
ε_{app}	24.22535	2.443609
IC_{app}	$[23.66766; 24.78304]$	$[0.03357783; 1.886422]$
ε_{test}	23.87006	15.601504
IC_{test}	$[22.72591; 25.01420]$	$[13.72249; 17.480519]$

TABLE 7 – Estimation ponctuelle des taux d'erreur ε et intervalles de confiance sur cette erreur par la méthode du classifieur euclidien et des K plus proches voisins sur le jeu de données Breastcancer

On constate donc que pour les deux jeux de données le classifieur euclidien présente un taux d'erreur equivalent pour l'ensemble d'apprentissage et l'ensemble de test autour de 25%. Cette erreur est importante, d'autant plus qu'avec un intervalle de confiance de 95% on ne descends pas au dessous de 22% d'erreur. On souligne donc avec ces tests les limites du classifieur euclidien. Ce classifieur est plus rapide que celui des K plus proches voisins mais fonctionne uniquement dans le cadre de jeux de données "simples", c'est-à-dire des classes d'individus suffisament distantes les unes par rapport aux autres.

La méthode des KPPV présente des taux d'erreurs sur les ensemble d'apprentissage intéressant puisqu'autour de $1 \sim 2\%$. Les erreurs sur les ensembles de tests sont eux autour de 15%. La classification par la méthode des KPPV semble donc plus précise dans que par la méthode du classificateur euclidien sur des ensembles plus complexes, plus proches de la réalité.

2 Regle de Bayes

En réalité, les effectifs des jeux de données étudiées précédemment ont été obtenu par le biais d'une loi binomiale et les individus ont été distribué dans les classes à l'aide d'une loi normale bivariée.

2.1 Distributions marginales

Nous souhaitons calculer les distributions marginales des variables X^1 et X^2 dans chacune des classes. Dans un premier temps nous allons nous intéresser

aux distributions marginales des variables pour les jeux de données Synth1. Étudiant des lois normales de dimension 2 l'expression usuelle de la densité est donnée par :

$$f(x) = \frac{1}{2\pi\sqrt{det\Sigma}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)) \tag{1}$$

Et la distribution suivie par X peut-être donnée par :

$$X \sim N(\mu, \Sigma) \tag{2}$$

Ceci nous permet de trouver les distributions de chacune des variables dans les 2 classes. Finalement on a :

$$X_1^1 \sim N(0,1) \tag{3}$$
$$X_1^2 \sim N(1,1) \tag{4}$$
$$X_2^1 \sim N(-2,1) \tag{5}$$
$$X_2^2 \sim N(0,1) \tag{6}$$

Nous aurions aussi pu résoudre ce problème en calculant pour chacune des variables la fonction de densité à l'aide de l'équation 1, séparer les fonctions de densité de chacune des variables du fait de leur indépendance et identifer les paramètres en rapprochant les fonctions ainsi obtenues avec la fonction de densité d'une variable normale :

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{7}$$

De la même manière pour Synth2 on a :

$$X_1^1 \sim N(3,1) \tag{8}$$
$$X_1^2 \sim N(0,1) \tag{9}$$
$$X_2^1 \sim N(-2,\sqrt{5}) \tag{10}$$
$$X_2^2 \sim N(0,1) \tag{11}$$

2.2 Courbes d'iso-densité

Il nous est ensuite demandé de calculer les expressions des courbes d'iso-densité de chacune des classes. Ces courbes permettent de représenter la densité de probabilité d'appartenance d'un individu à une des classes. Les courbes d'iso-densité ont pour équation :

$$(x-\mu)^T\Sigma^{-1}(x-\mu) = k \tag{12}$$

où k est une constante.
On trouve donc pour les jeux de données Synth1-n des courbes d'iso-densité

d'équation :

$$X^1 : {x_1}^2 + (x_2 - 1)^2 = k_1 \tag{13}$$

$$X^2 : (x_1 + 2)^2 + {x_2}^2 = k_2 \tag{14}$$

Et pour Synth2 :

$$X^1 : (x_1 - 3)^2 + {x_2}^2 = k_1 \tag{15}$$

$$X^2 : \frac{1}{5}(x_1 - 3)^2 + {x_2}^2 = k_2 \tag{16}$$

Nous remarquons donc que les expressions d'iso-densité correspondent à des équations de cercles de centre μ. De plus, comme Σ est diagonale, les axes des cercles sont parallèles aux axes de coordonnées.

2.3 Expression de la règle de Bayes et frontière de décision

La règle de Bayes est une règle qui vise à minimiser la probabilité d'erreur d'une règle de décision. La règle δ^* minimisant l'erreur pour x fixé est définie par :

$$\delta^*(x) = \begin{Bmatrix} a_1 \; si \; \mathbb{P}(w_2|x) < \mathbb{P}(w_1|x) \\ a_2 \; sinon. \end{Bmatrix} \tag{17}$$

Qui peut s'exprimer en fonction du rapport de vraisemblance des deux classes par :

$$\delta^*(x) = \begin{Bmatrix} a_1 \; si \; \frac{f_1(x)}{f_2(x)} > \frac{\pi_2}{\pi_1} \\ a_2 \; sinon. \end{Bmatrix} \tag{18}$$

2.3.1 Jeux de données Synth1-n

Nous souhaitons montrer que pour les jeux de données Synth1-n, la règle de Bayes est une fonction linéaire de x_1 et x_2 et d'une constante k. Nous obtienons donc :

$$\frac{f_1(x)}{f_2(x)} = \frac{\frac{1}{2\pi\sqrt{det\Sigma}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))}{\frac{1}{2\pi\sqrt{det\Sigma}} exp(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu))} \tag{19}$$

$$= \frac{\frac{1}{2\pi} exp(-\frac{1}{2}{x_1}^2 + (x_2 - 1))}{\frac{1}{2\pi} exp(-\frac{1}{2}((x_1+2)^2 + {x_2}^2)} \tag{20}$$

$$= 2x_1 + x_2 + \frac{3}{2} \tag{21}$$

Avec $\pi_1 = 0.5$ et $\pi_2 = 0.5$, le rapport $\frac{\pi_2}{\pi_1} = 1$ Nous trouvons donc que la règle de Bayes est bien une fonction linéaire de x_1 et x_2 et d'une constante $k = \frac{3}{4}$. Pour trouver la frontière de décision nous pouvons nous intéresser à la valeur de x_1 et x_2 lorsque $f_1(x) = f_2(x)$:

$$\frac{1}{2\pi} exp(-\frac{1}{2}{x_1}^2 + (x_2 - 1)) = \frac{1}{2\pi} exp(-\frac{1}{2}((x_1+2)^2 + {x_2}^2) \tag{22}$$

Ce qui nous amène à une expression de la frontière de décision telle que :

$$x_2 = -2x_1 - \frac{3}{2} \tag{23}$$

Il est intéressant de souligner que dans notre cas, avec Synth1-n, comme $\Sigma_1 = \Sigma_2$, la règle de Bayes est une règle de décision linéaire.

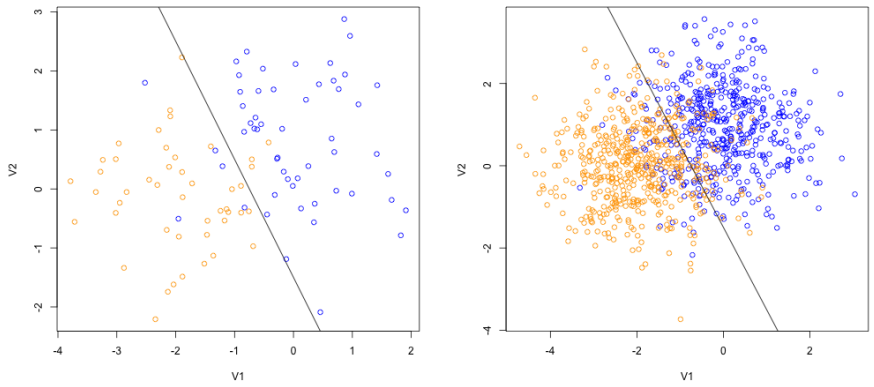

FIGURE 4 – Jeu de données Synth1-100 et Synth1-1000 avec la frontière de décision $x_2 = -2x_1 - \frac{3}{2}$

2.3.2 Jeux de données Synth2

De la même manière que précédement nous souhaitons trouver la règle de Bayes pour Synth-2. Nous pouvons exprimer cette règle en fonction de x_1 et x_2 comme suit :

$$\frac{f_1(x)}{f_2(x)} > 1 \tag{24}$$

$$\frac{\frac{1}{2\pi}exp(-\frac{1}{2}(x_1-3)^2 + {x_2}^2)}{\frac{1}{2\pi\sqrt{5}}exp(-\frac{1}{2}(\frac{1}{5}(x_1+2)^2 + {x_2}^2)} > 1 \tag{25}$$

$$-4{x_1}^2 + 34x_1 + 41 > -5ln(5) \tag{26}$$

$$\tag{27}$$

Pour trouver la frontière de décision nous nous intéressons à la même relation que précédement entre $f_1(x)$ et $f_2(x)$, cherchons x_1 et x_2 tel que $f_1(x) = f_2(x)$, c'est-à-dire :

$$\frac{1}{2\pi}exp(-\frac{1}{2}(x_1-3)^2 + {x_2}^2) = \frac{1}{2\pi\sqrt{5}}exp(-\frac{1}{2}(\frac{1}{5}(x_1+2)^2 + {x_2}^2) \tag{28}$$

nous obtenons donc une expression de courbe telle que :

$$-4{x_1}^2 + 34x_1 + 41 + 5ln(5) = 0 \tag{29}$$

Nous pouvons souligner que dans ce cas la règle de décision est une règle quadratique. Sa représentation graphique est donnée Figure 5, où la frontière de décision, sur l'échantillon des données, peut être associée à une droite.

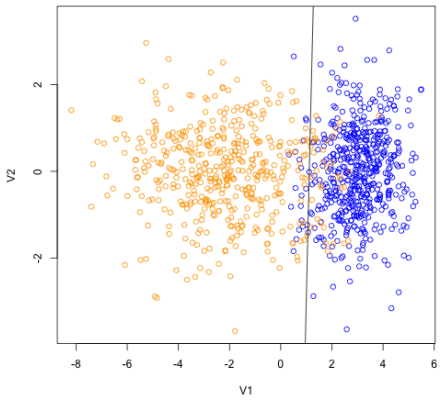

FIGURE 5 – Frontière de décision de la règle de Bayes sur le jeu de données Synth2-1000

2.4 Erreur de Bayes

Nous pouvons dans certains cas simples (le traitement de seulement deux classes peut en être un), déterminer exactement la probabilité d'erreur de Bayes ε^*. Cette dernière est donnée par :

$$\varepsilon^* = \phi(-\frac{\Delta}{2}) \tag{30}$$

où $\Delta^2 = (\mu_2 - \mu_1)^T \Sigma^{-1} (\mu_2 - \mu_1)$.
Nous trouvons pour Synth1-n et Synth2 la même valeur de Δ qui est : $\Delta = \sqrt{5}$.

Finalement nous obtenons le taux d'erreur en se rapportant la fontion de répartition de la loi normale univariée :

$$\varepsilon^* = \phi(-\frac{\sqrt{5}}{2}) = 1 - \phi(\frac{\sqrt{5}}{2}) = 0,13136 \tag{31}$$