

# *Conception et implémentation*

## **Introduction**

Dans ce dernier chapitre, nous présentons les différentes étapes réalisées durant l'implémentation de notre application, nous commençons à décrire la création de la base de données, ensuite nous définissons les fonctionnalités de notre application et enfin nous discutons les résultats obtenus.

Nous avons utilisé deux modèles de classification pour diagnostiquer (ou séparer) les fœtus normaux (sains), des fœtus douteux (ou suspects) et ceux qui présentent une pathologie. Ces deux classifieurs supervisés sont : les séparateurs à vastes marges (SVM) et les réseaux de neurones multicouches (PMC).

## **III.1 Base fœtale**

### **III.1.1 Description de la base de données**

Dans ces dernières années, plusieurs chercheurs en gynécologie s'intéressent à l'interprétation informatisée d'un tracé cardiotocographique (CTG) pour la détection de l'état d'un fœtus. En effet, différents systèmes ont été mis en œuvre (exp : STAN, système Oxford 8000/8002, Système 2CTG et le système SisPorto...) et pour comparer le développement de ces systèmes, quelques repères ont été disponibles comme la création d'une base commune des signaux cardiotocographique.

Pour notre application nous avons utilisé une base de données cardiotocographique (CTG.xls) téléchargée à partir du site web [[archive.ics.uci.edu/ml/machine-learning-databases/00193 /CTG.xls](http://archive.ics.uci.edu/ml/machine-learning-databases/00193/CTG.xls)]. Cette base contient 2126 mesures ou tracés CTG traités automatiquement par un système d'analyse informatisée du rythme cardiaque fœtal qui est le système **SisPorto version 2.0** (voir figure III.1).

### III.1.2. Description du système SisPorto

Le système SisPorto est un programme d'analyse automatisée des cardiocrograms développé en 2000 à l'université de Porto. Il est le résultat d'un projet de collaboration entre la faculté de l'université de médecine et l'institut de génie biomédical [36][37].

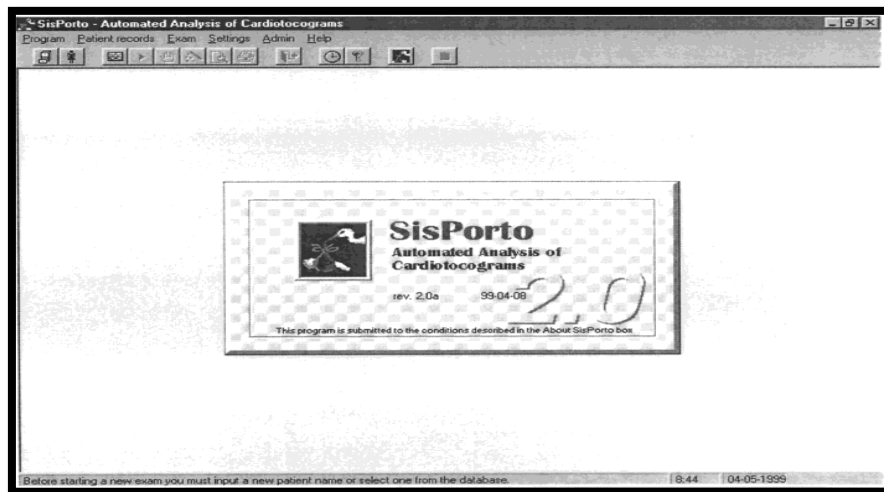


Figure III.1 menu principale du système SisPorto

### III.1.3. Configuration matérielle et logicielle

Le système SisPorto 2.0 fonctionne sur tout ordinateur personnel en utilisant le système d'exploitation Windows (Windows 3.1, Windows 95, Windows NT ou Windows 98). La configuration matérielle minimale requise est : un processus Pentium 166 MHz, 16 MB de RAM.

### III.1.4. Acquisition du signal et stockage

Les signaux numériques sont transmis par le moniteur fœtal au port série de l'ordinateur via un câble d'ordinateur standard, en utilisant un protocole RS232 ou RS485.

SisPorto 2.0 est capable d'acquérir les signaux cardiocrographique à partir des moniteurs fœtaux à condition qu'ils ont une sortie RS232 ou RS485, comme exemple on peut citer : le moniteur fœtal Sonicaid, Hewlett-Packard, Sonicaid Meridian 800, HPM1350B...).

Ces signaux (ou tracés) acquis sont ensuite stockés dans des fichiers tels que reçus du moniteur fœtal sans aucune réduction de signal, ce qui caractérise le système SisPorto

par rapport aux autres. Le système SisPorto 2.0 permet aussi l'impression et la visualisation des signaux traités ou non traités et donne la possibilité d'effectuer un enregistrement simultané des tracés CTG de jumeaux.[36]

### III.1.5. Traitement des tracés CTG

Le traitement des tracés cardiotocographique (en temps réel) commence automatiquement, 10 minute après l'acquisition et l'affichage sera modifié chaque 5 minute, par contre les tracés déjà stockés dans la base de données sont traités une seule fois. L'analyse du tracé est basée sur l'interprétation proposées par la Fédération Internationale de Gynécologie et d'Obstétrique (FIGO) [38]. La figure III.2 nous montre un tracé CTG analysé par le système. Les valeurs affichées correspondent à une durée de 10 minutes.

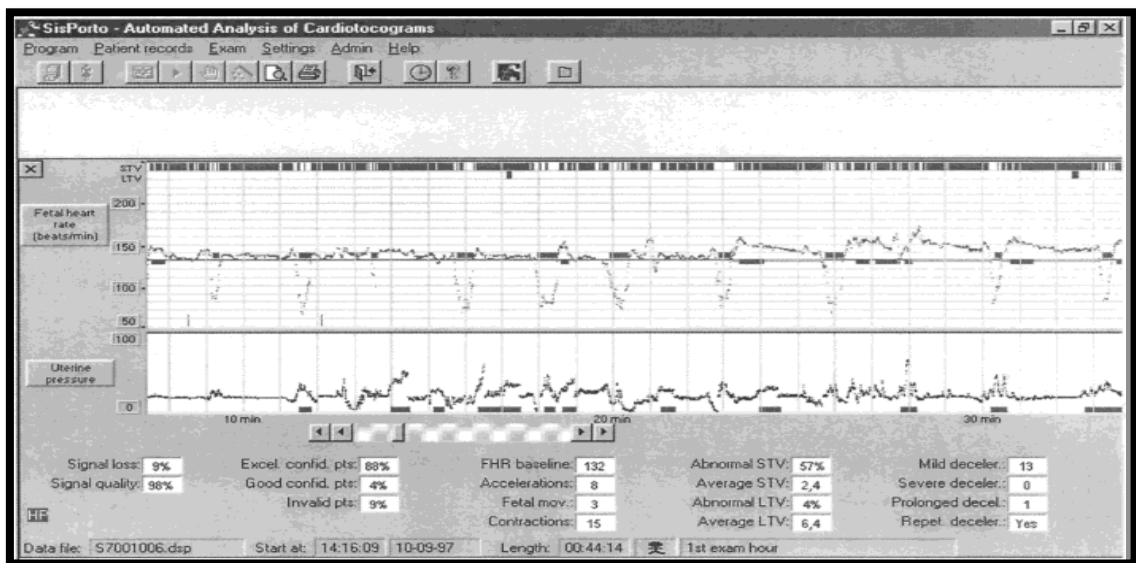


Figure III.2 analyse d'un signal cardiotocographique par le système SisPorto 2.0 [36]

### III.1.6. Définition des attributs

Le système SisPorto 2.0 utilise 21 attributs (paramètres) calculés au cours de l'analyse du tracé, seulement 11 attributs caractérisent bien le rythme cardiaque foetal et qui sont utilisés dans notre application (voir tableau III.1). Alors que les 10 autres attributs (signal loss, Width, Min, Max,...) décrivent la qualité du signal traité et n'ont pas une relation directe avec le RCF.

	<b>Attributs</b>	<b>Définition</b>
1	LB	Fréquence cardiaque de base (battements par minute)
2	AC	Nombre d'accélération
3	FM	Nombre de mouvements fœtaux
4	UC	Nombre de contractions utérines
5	ASTV	Durée moyenne d'une variabilité à court terme anormale
6	mSTV	Valeur moyenne de la variabilité à court terme
7	ALTV	Durée moyenne d'une variabilité à long terme anormale
8	mLTV	Valeur moyenne de la variabilité à long terme
9	DL	Nombre de décélérations légères (précoces)
10	DS	Nombre de décélérations sévères (ou variables compliqués)
11	DP	Nombre de décélérations prolongés

Table III.1 : attributs utilisés

## **III.2. Construction des classifieurs**

L'objectif de cette étude est de construire un classifieur supervisé, capable de séparer les différents états du fœtus et de prédire la classe adéquate.

### **III.2.1. Environnement de développement**

Notre choix du langage java a été guidé par les avantages qu'offre la programmation orientée objet. Le langage Java examine le programme au fil de l'exécution et libère la mémoire automatiquement. Cette fonctionnalité diminue les reprises de panne du programme et ne laisse pas la possibilité d'une mauvaise utilisation de la mémoire.

Notre application est réalisée sous Netbeans version 6.5 sous windows. Le choix de la programmation sous windows a été pris à cause de l'interface graphique qu'offre cet environnement.

Le choix de Netbeans est fondamental puisque c'est un logiciel standard de développement utilisé pour la création des interfaces très puissantes. Par contre les phases de calcul sont réalisées sous Matlab. Notons que Matlab et Netbeans sont reliés par la bibliothèque JMatlink.

### **III.2.2. Présentation de l'application**

La fenêtre principale de notre application est composée des quatre menus suivants : menu Fichier, menu Apprentissage, menu Classification et menu Aide.



Figure III.3 fenêtre principale

1. **Menu Fichier** : permet le chargement de la base d'apprentissage



Figure III.4 menu fichier

- Une fenêtre s'ouvre permettant de charger la base d'apprentissage, ensuite de l'afficher dans une table.

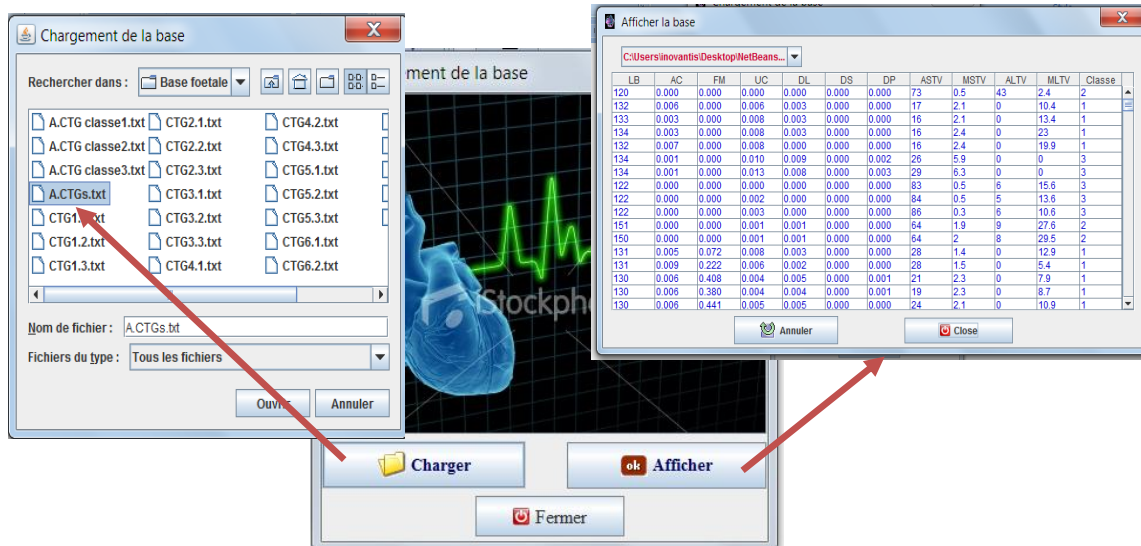


Figure III.5 chargement et affichage de la base d'apprentissage

- 2. **Menu apprentissage :** permet le choix du classifieur pour la phase d'apprentissage.

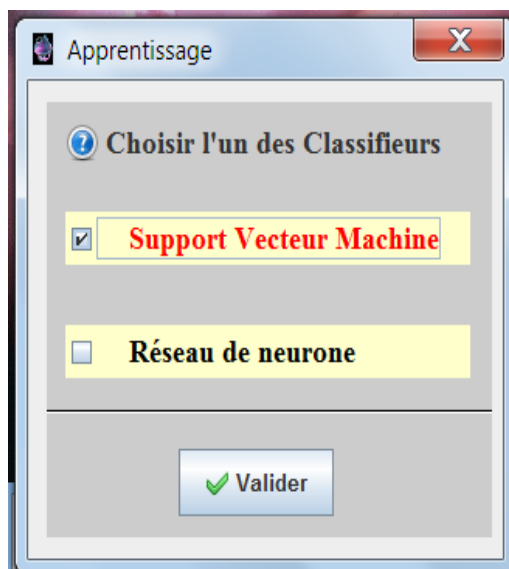


Figure III.6 menu apprentissage

- Après, on choisit l'un des deux classifieurs. Cette fenêtre permet de calculer la capacité d'apprentissage du classifieur et permet de lancer un nouvel apprentissage.

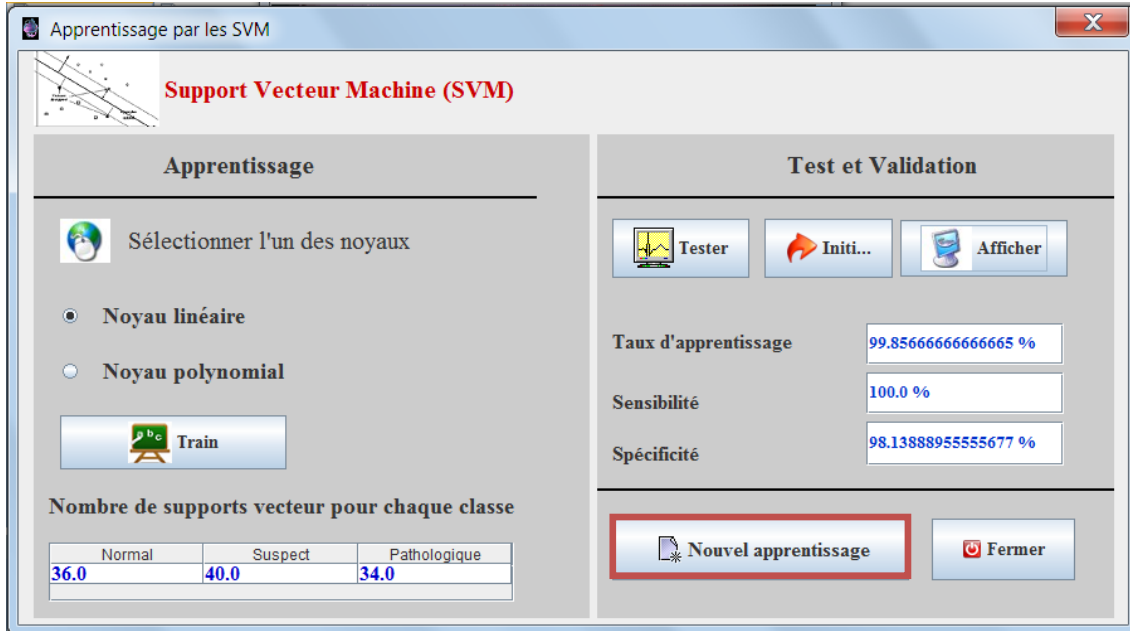


Figure III.7 apprentissage par SVM

- Lancer un nouvel apprentissage, la fenêtre ci-dessous s'affiche. Celle-ci permet de modifier les paramètres d'apprentissage et de tester de nouvelles données.

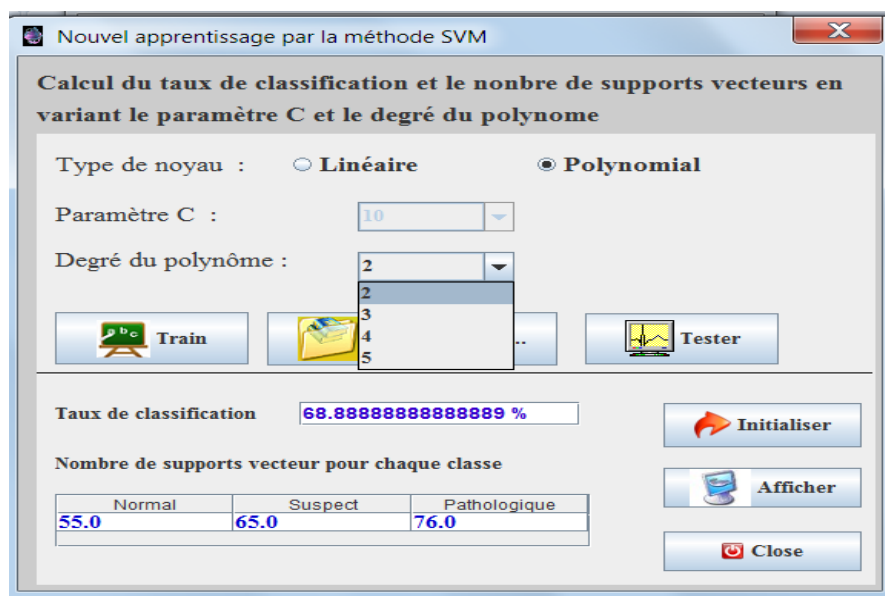


Figure III.8 nouvel apprentissage par SVM

- En choisissant le classifieur réseau de neurone, la fenêtre suivante s'affiche. Le bouton **Tester** permet de classifier les exemples d'apprentissage. Le bouton **nouvel apprentissage** nous donne la possibilité de modifier les paramètres d'apprentissage.

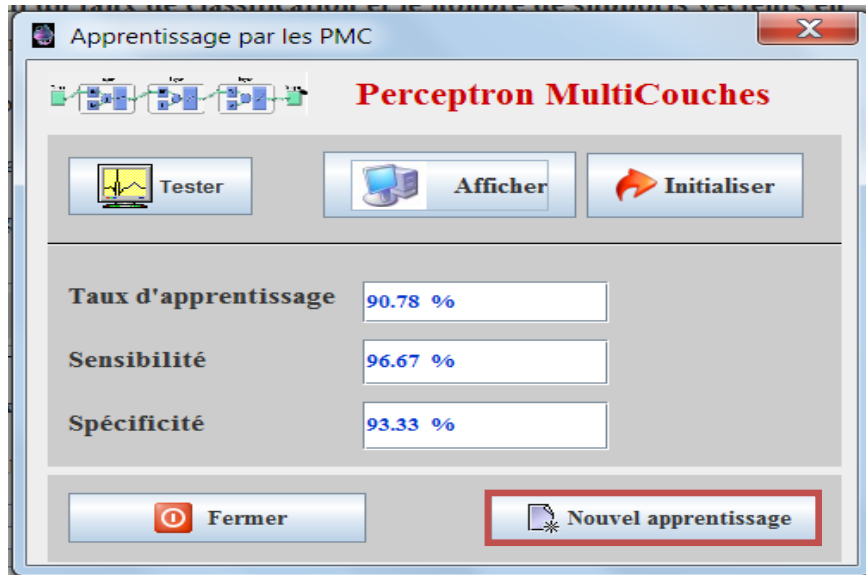


Figure III.9 apprentissage par PMC

- Après le choix des paramètres, lancer l'apprentissage par le bouton **Train**. Le modèle d'apprentissage est enregistré. Ensuite, charger la base de test et tester.



Figure III.10 nouvel apprentissage par PMC



**3. Menu classification :** permet de classifier de nouvelles données du tracé CTG. La fenêtre suivante montre la classification d'un tracé CTG par le classifieur SVM.

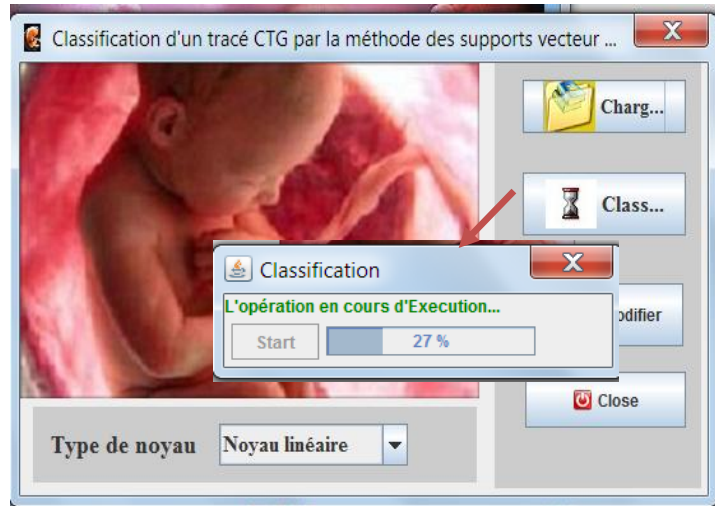


Figure III.11 classification d'un tracé CTG par SVM

➤ En fin, une boîte de dialogue apparaît pour identifier l'état du fœtus.

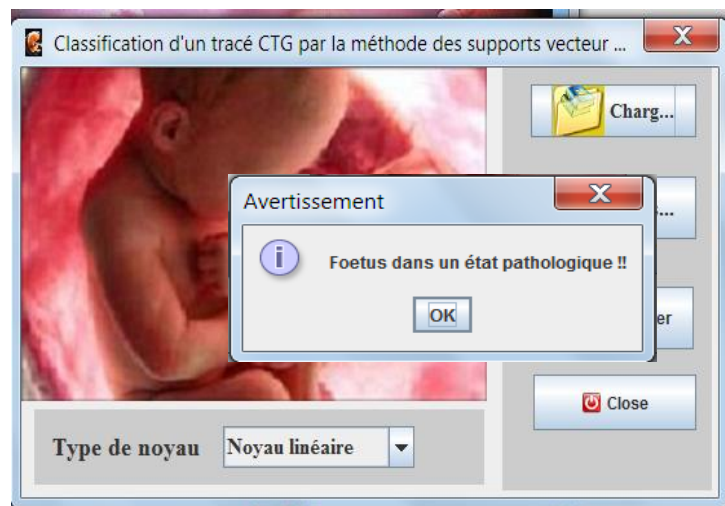


Figure III.12 résultat de classification

### III.2.3 Classification par les Supports Vecteurs Machines (SVM)

Nous décomposons la base foetal en deux parties, partie apprentissage qui contient 1163 exemples de la base et la partie test qui représente 50 exemples pour chaque classe (voir tableau III.2). L'approche utilisée est un-contre-un pour les deux noyaux suivants : linéaire et polynomial.

	Classe 1	Classe2	Classe3	totale
phase d'apprentissage	800	237	126	1163
Phase de test	50	50	50	150

Table III.2 nombre d'exemples pour chaque classe

Pour classifier un tracé CTG d'un fœtus, il faut d'abord faire apprendre le système, en lui donnant en entrée les exemples d'apprentissage, leurs étiquettes (ou classes) et les paramètres d'apprentissages (par exemple : la valeur du degré pour le noyau polynomial, gamma pour le noyau gaussien, le paramètre de pénalisation C). Les résultats de l'apprentissage sont les paramètres suivants (Alpha, SVs, le Biais b, nSV ) nécessaires à la construction du classifieur SVM.

Pendant la phase de test, les exemples (non pris au cours de l'apprentissage) sont soumis au classifieur SVM pour décider à quelle classe appartient ce tracé CTG.

#### La 1<sup>ère</sup> Partie

Dans cette partie nous allons tester la capacité d'apprentissage du classifieur linéaire et polynomial degré 4, pour cette raison nous allons utiliser la même base d'apprentissage pour la classification. (Voir tableau III.3, III.4, III.5)

	Classe 1	Classe 2	Classe 3
Classe 1	100	0	0
Classe 2	0	99.6	0.4
Classe 3	0	0	100

Table III.3 matrice de confusion (noyau linéaire)

	Classe 1	Classe 2	Classe 3
Classe 1	96.6667	0	3.3333
Classe 2	0	86.6667	13.3300
Classe 3	0	3.3333	96.6633

Table III.4 matrice de confusion (noyau polynomial degré4)

## 2<sup>ème</sup> partie

Dans cette partie nous classifions des exemples de test (50 pour chaque classe), en variant le paramètre de pénalisation C du classifieur SVM au cours de l'apprentissage. Dans le but de diminuer le taux d'erreur ensuite de comparer les résultats obtenus par les deux noyaux (linéaire et polynomial degré2,3,4,5) (voir tableau III.5).

Noyau	Taux de classification ( %)					
	C = 10	C = 100	C = 1000	C = 2000	C=10000	C=20000
Linéaire	78.89	81	83.22	84.34	85.45	85.45
Polynomial degré 2	68.89	68.89	68.89	68.89	68.89	68.89
Polynomial degré 3	73.33	73.33	73.33	73.33	73.33	73.33
Polynomial degré 4	74.44	74.44	74.44	74.44	74.44	74.44
Polynomial degré 5	71	71	71	71	71	71

Table III.5 : taux de classification pour les deux noyaux

Pour chaque apprentissage nous allons extraire le nombre de supports vecteurs pour chaque changement du paramètre C. le tableau III.6 illustre le nombre de supports vecteurs.

Noyau	C = 10	C = 100	C = 1000	C =2000	C=10000	C=20000
Linéaire	270	266	201	125	110	110
Polynomial degré 2	196	196	196	196	196	196
Polynomial degré 3	111	111	111	111	111	111
Polynomial degré 4	108	108	108	108	108	108
Polynomial degré 5	290	290	290	290	290	290

Table III.6 : nombre de supports vecteurs pour chaque noyau

Le tableau ci-dessous nous indique le taux de reconnaissance des trois classes pendant la phase de test pour les deux noyaux.

	Classe 1	Classe 2	Classe3
Taux de classification (noyau linéaire)	91.01	72.67	88.33
Taux de classification (noyau Polynomial)	81.67	65.43	76

Table III.7 : classification des trois classes par SVM (C=10000 et degré = 4)

### Discussion

Les résultats de classification par la méthode des supports vecteurs machine (SVM) sont présentés dans les tableaux III.5 et III.6. D'après le premier tableau, on constate que le paramètre C influe sur le taux de classification dans le cas d'un noyau linéaire, en faisant augmenter le paramètre C le taux de classification augmente et atteint une valeur maximale de l'ordre 85.45 % pour un C=10000.

Par contre, dans le cas d'un noyau polynomial on constate que le taux de classification varie en fonction du degré du polynôme, et il n'est pas influencé par le paramètre C. Pour un degré du polynôme égal à 4, on obtient une valeur maximale du taux de classification. Par contre, on remarque que pour un degré du polynôme supérieur à 4 le taux de classification diminue. D'après les résultats obtenus, nous concluons que le classifieur linéaire est meilleur que le classifieurs polynomial pour cette base.

Le tableau III.7 nous donne la variation du nombre des supports vecteurs en fonction du paramètre C et du type de noyau. On constate que le nombre de vecteurs supports diminue lorsque le C augmente et le degré du polynôme augmente.

Les résultats obtenus dans le tableau III.8 montrent que les deux classifieurs ont bien reconnu la classe 1 et la classe 3, par contre la classe 2 a été faiblement reconnue surtout pour le noyau polynomial.

### III.2.4 Classification neuronale

Vu la robustesse des réseaux de neurones et la variété de ces méthodes dans la classification, nous avons choisi l'un des plus répandu qui est le perceptron multicouche (PMC).

#### L'architecture du PMC

La couche d'entrée du perceptron multicouche (PMC) contient 11 neurones. Le PMC reçoit dans son entrée 1163 vecteurs, ces vecteurs sont extraits à partir de la base d'apprentissage.

La couche cachée du PMC contient 25 neurones. Le choix du nombre de neurones pour cette couche cachée est obtenu expérimentalement.

La couche de sortie contient trois neurones, chaque neurone représente une classe. La sortie de ces neurones peut prendre la valeur 1, 2 ou 3.

#### Phase d'apprentissage

Après l'apprentissage du réseau de neurone et au bout de 311 itérations, ce dernier converge et atteint le but. L'erreur minimale pour laquelle le PMC termine son apprentissage est fixée à  $10^{-3}$  et le nombre maximal d'époques prévu à l'entraînement est égal à 1000 époques (voir figure III.13).

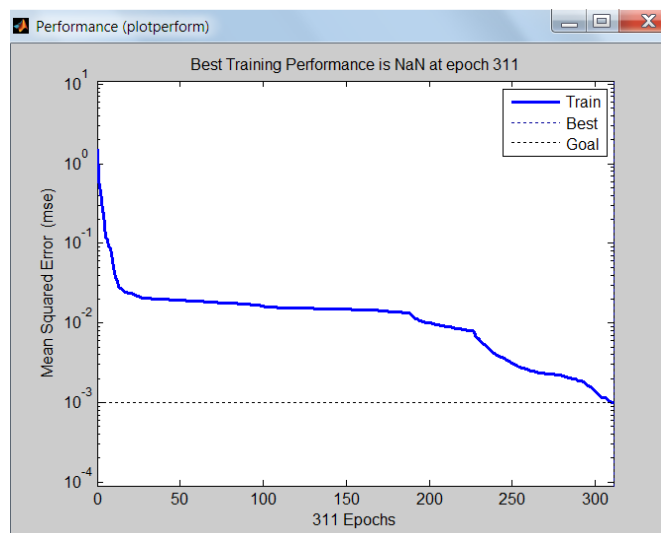


Figure III.13 schéma de convergence du PMC

**Phase de test**

Une fois l'apprentissage est terminé, une étape de test commence. Les tests concernent la vérification de la performance du réseau de neurone. En effet, cette validation est exprimée par le taux de reconnaissance (TC), la sensibilité(SE) et la spécificité (SP).

Les résultats obtenus sont présentés ci-dessous :

Neurones cachés	Résultats (%)					
	Erreur minimale	Nbre d'itér.	Durée	TC	SE	SP
8 neurones	0.001	800	6 :48	35.71	13.33	88.33
8 neurones	0.0002	900	0 :15	57.78	41.45	90
25 neurones	0.001	311	3 :15	73.52	86.23	90
25 neurones	0.0002	94	0 :10	71.11	26.67	88.33
30 neurones	0.001	98	1 :55	53.75	35.75	81.67
30 neurones	0.0002	1000	1 :58	51.11	58	96.67

Table III.8 résultats de la classification neuronale

D'après les résultats du tableau III.8, nous remarquons que lorsque le nombre de neurones cachés est égal à 8, le taux de classification est de 35.71 %. Lorsqu'on augmente le nombre à 25, nous obtenons un taux satisfaisant de 73.52 %. Ceci signifie que le classifieur n'a pas pu séparer correctement dans le premier cas les exemples dans son espace de séparation et qu'il a besoin d'un autre espace plus grand.

Le tableau III.9 représente le taux de classification et d'erreur pour chaque classe. La base de test contient 50 exemples pour chaque classe.

	Taux de classification ( %)		
	Classe 1	Classe 2	Classe 3
Taux de classification	78.89	60.369	66.956
Taux d'erreur	21.11	39.631	33.044

Table III.9 classification des trois classes par PMC

D'après les résultats du tableau ci-dessus, nous remarquons que le classifieur neuronal a bien reconnu la classe 1 par rapport aux deux autres.

Le tableau III.11 donne une comparaison entre les deux classifieurs SVM et PMC, pendant la phase d'apprentissage et de test.

	Noyau linéaire			Noyau polynomial			PMC		
	TC	SE	SP	TC	SE	SP	TC	SE	SP
Base app.	99.86	100	96.22	93.33	96.98	89.56	90.78	96.67	93.33
Base test	85.45	91.68	94.11	74.44	87.69	98.22	73.52	86.23	90

Table III.10 résultats des deux classifieurs pour les phases apprentissage et test

La figure III.14 illustre le taux de classification et le taux d'erreur globaux, la sensibilité et la spécificité globales de chaque méthode pour la phase de test.

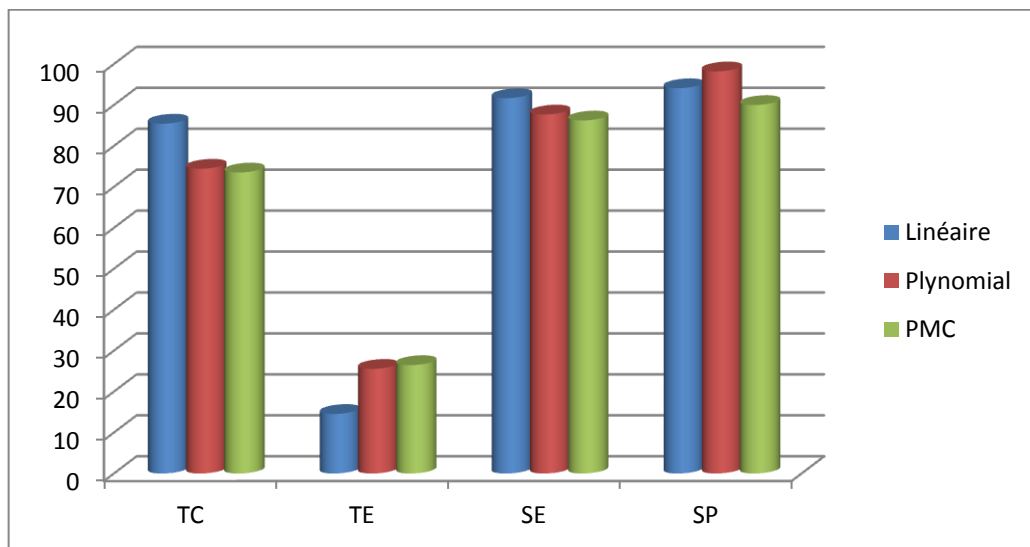


Figure III.14 comparaison des trois classifieurs

Ces résultats montrent que les deux classifieurs SVM (noyau linéaire et polynomial) ont un taux de classification supérieur au modèle PMC.

### **III.3 Conclusion**

Dans ce dernier chapitre, nous avons montré que la régularisation des paramètres du noyau SVM peut améliorer la performance du classifieur. Les expérimentations effectuées justifient la supériorité du noyau linéaire par rapport au noyau polynomial. Concernant le réseau de neurone la détermination de la structure d'un perceptron multicouches reste un problème crucial lors de la phase d'apprentissage.

La classification par la méthode des séparateurs à vaste marges (SVM) est meilleure que celle du réseau de neurone (PMC).