TP4 : Classification binaire avec SVM, Mélange de modèles, et Modèles Probabilistes Mixtes

Exercice 1 :

Question 2

La stratification est importante ici car sans cela il y aura un déséquilibre entre les 2 classes ce qui fausserait l'entrainement.

Question 3 :

Dans notre rapport de classification on observe que le recall d'une détection de spam (1) n'est pas très bonne en comparaison au ham (0)

Rapport de classification:

	Metric precision	recall	f1-score	support
0 (ham)	0.98	0.99	0.99	965
1 (spam)	0.96	0.87	0.92	150
Accuracy			0.98	1115
Macro avg	0.97	0.93	0.95	1115
weighted avg	0.98	0.98	0.98	1115

Question 4 :

Dans notre cas on observe que l'erreur la plus fréquente est lors de la détection de faux positif (1) (FP). C'est-à-dire la détection de ham lorsque celle-ci est vrai négative (0)

		Réalité	
		Négative : 0	Positive : 1
Prédiction	Négative : 0	960	5
	Positive : 1	19	131

Question 5 : La courbe ROC en *Figure1* montre le compromis entre le taux de vrais positifs TPR et le taux de faux positifs FPR. Par exemple un score AUC proche de 1 est un bon modèle, un score proche de 0.5 est un modele aléatoire. On observe dans notre cas que le modele est correct. Avec un score se rapprochant de 1.

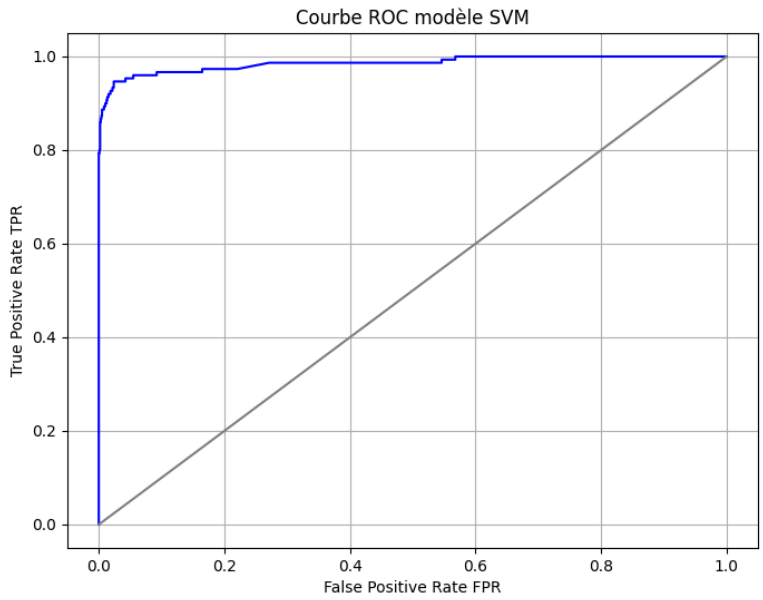


Figure 1 : Courbe ROC modèle SVM

Exercice 2 :

Question 1 :

Comparaison Matrice de confusion Annexes Figure 2 :

- **Logistic Regression :** On peut voir qu'il n'y a pas de FN ce qui est très bon pour la détection de spam (1). Il y a cependant quelques erreurs dans la détection de ham (0) 22.
- **Naive Bayes :** On observe quelques erreurs pour la prédiction de ham et spam cependant les erreurs sont environ équivalent en nombre (12(FP) -8(FN)) ce qui est pas mal et offre un bon compromis.
- **SVM :** Est performant mais légèrement moins robuste pour la classe spam 19 (FP).

Comparaison Rapport de classification Annexes Figure 2 :

- **Logistic Regression :** A un f1-score équilibré. Possède le meilleur taux de prédiction pour les spam (100%) avec ce dataset, cependant après recall les vrai positive ne sont bon qu'a 85%, on obtient cependant un f1_score honnorable.
- **Naive Bayes :** A un excellent F1-score, avec un score precision et recall équilibré.
- **SVM :** A de très bon résultat dans l'ensemble, mais plus de difficulté dans la détection de spam avec un score recall spam pas très bon.

Question 2 :

Analyse Matrice de confusion Annexes Figure 2 :

- **Voting classifier Hard :** On observe que seulement 1 Fn à été incorrectement classé comme ham et 20 FP incorrectement classés comme spam.

Analyse Rapport de classification Annexes Figure 2 :

- **Voting classifier Hard :** Les métriques intéressantes ici sont celle des spam. Ou le score de précision est de rappel montre la faiblesse du modèle. A sa capaciter à détecter des spam de manière correct.

Question 3 :

Dans notre cas le vote soft serait le plus performant. Comme on peut le voir dans la *Figure 2 en Annexe*. Un écart dans le score de recall légèrement plus basse que celle du voting soft.

Question 4 :

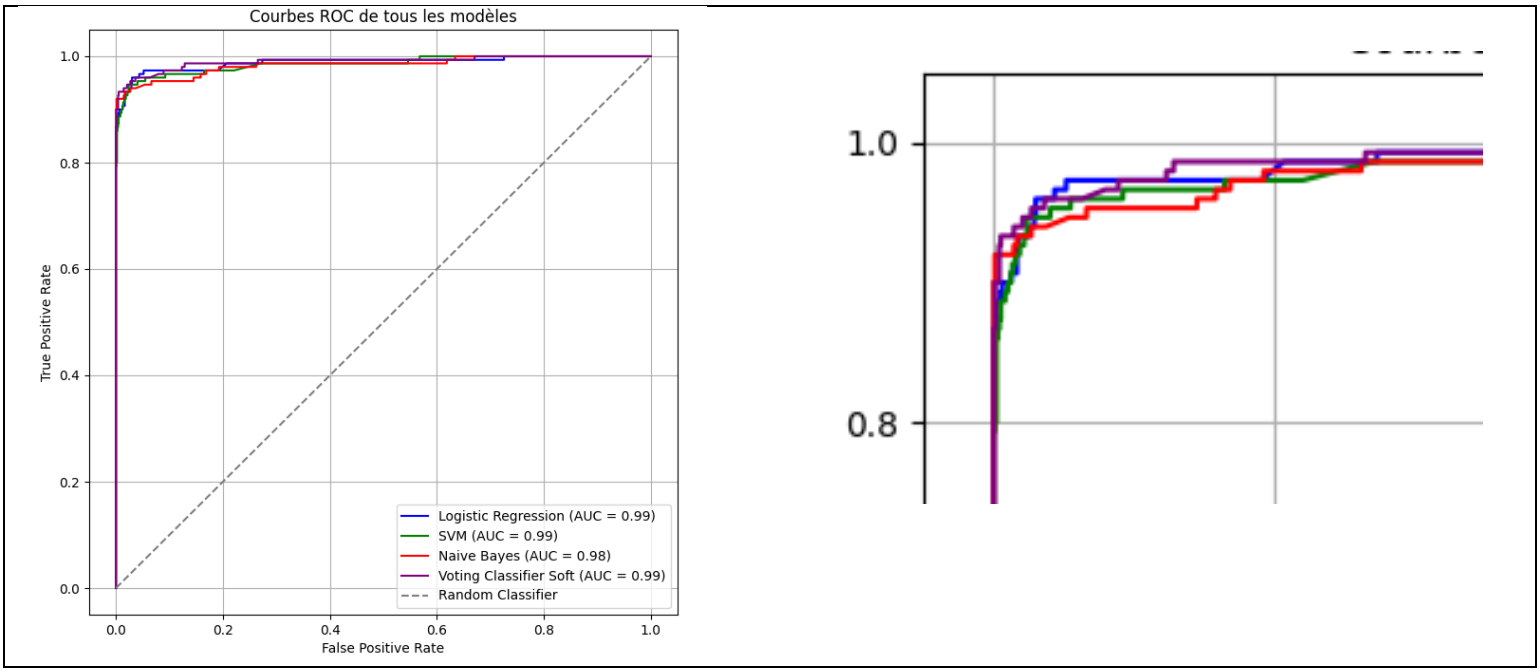


Figure 2 : Graphique courbe ROC de différents modeles

On observe sur les courbes ROC que les modèles (Logistic regression, SVM, Voting classifier Soft ont une excellente performance pour détecter les faux positif FP et les vrai positif VP. De plus on voit également que l'AUC de chacun est de 0.99 ce qui est très bon.

En revanche pour le Naives Bayes on observe de moins bonne performance est un AUC de 0.98.

Question 5 :

Théoriquement si nous utilisons plusieurs modèles pour une même tache comme ici la détection de spam et ham. Chaque modèle possède sa force et donc un mélange de ces modèles ne peut être que meilleur. Cependant par la suite nous allons voir les limites de l'utilisation d'un mélange de modèles.

Exercice 3 :

Question 1 :

Le GMM fonctionne de manière avec plusieurs clusters ou chaque groupe suit une distribution gaussienne. Par exemple un GMM attribue à chaque point 50% d'être un spam et 50% d'être un ham. Elle permet également de travailler sur des données complexes. Comme des données

avec du bruit ou dans notre cas ici la classification ham/spam. Car la construction d'un spam ou d'un ham est assez similaire.

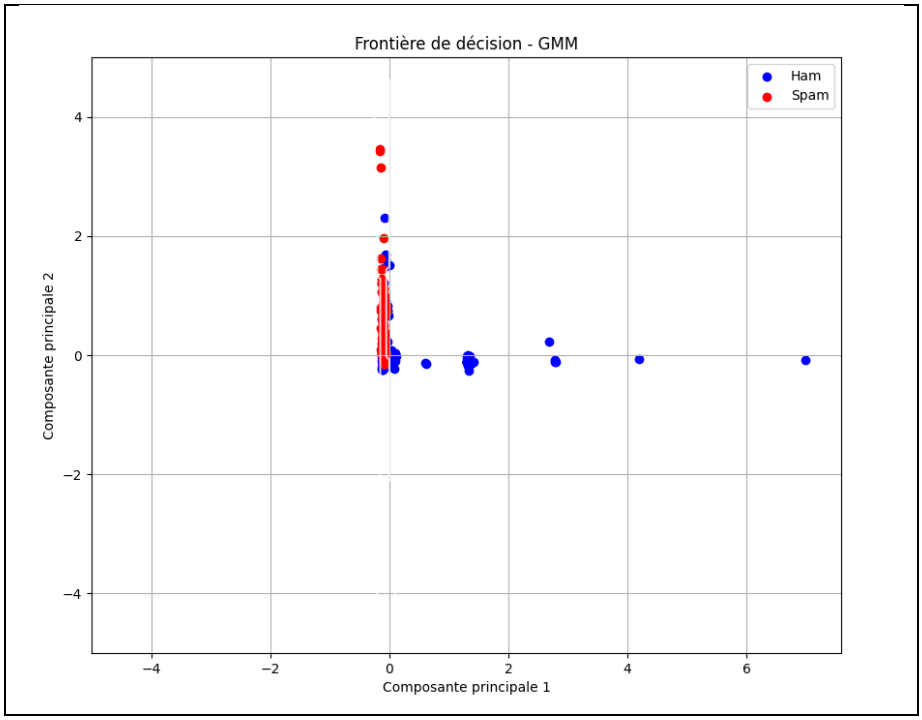
Question 3 :

Voir Annexe *Figure 2*

Question 4 :

En comparaison des autres modèles on peut observer une grande difficulté à détecter les vrai positif avec le score de recall le plus bas parmi tous les modèles (76%).

Question 5 :



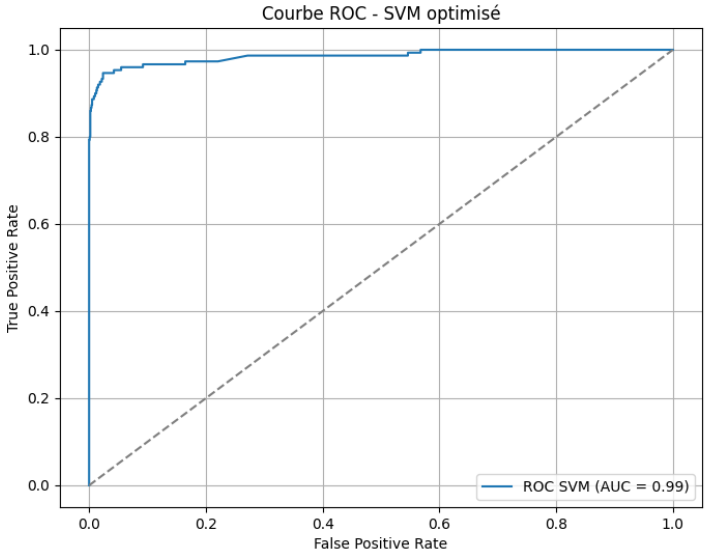
Exercice 4 :

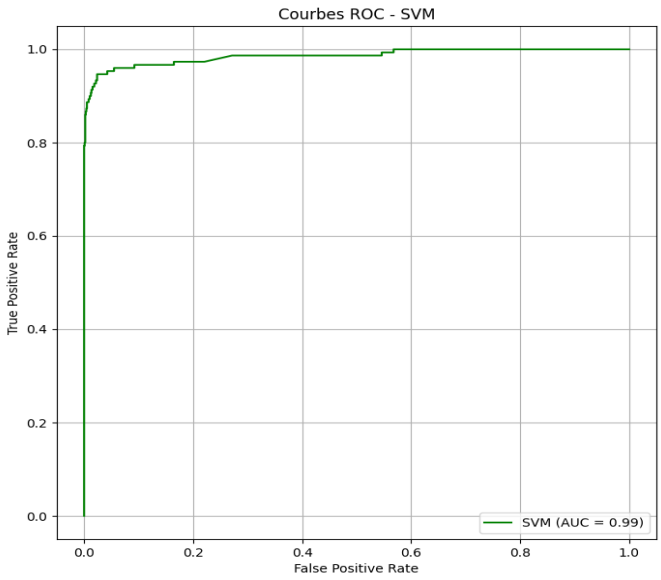
Question 1 :

On peut optimiser divers paramètres, le C, Kernel, Gamma et bien d'autre cependant dans ce TP nous allons seulement nous occuper de ces hyperparamètres.

- « **C** » contrôle à quel point la marge de séparation par rapport aux erreurs de classification.
- « **Kernel** » aide à trouver une marge de séparation même si les données ne sont pas linéairement séparables.
- « **Gamma** » détermine l'influence d'un seul point sur la séparation, par exemple un petit gamma dépendra des points éloignés à l'inverse d'un grand gamma qui est plus précis avec les points proches

Question 2 :





Après avoir effectué une recherche par GridSearchCV pour optimiser le modèles SVM et GMM j'obtiens les valeurs suivantes.

- Temps d'entraînement SVM : 2241.61 secondes
- Meilleurs paramètres SVM : {'C': 1, 'gamma': 'scale', 'kernel': 'linear'}
- Meilleurs paramètres GMM : {'covariance_type' : 'spherical', 'n_components' : 4}

Voting Classifier Accuracy : 0.9874439461883409

Question 5 : En général l'optimisation sur nos modèles permet une exécution plus rapide.

Exercice 5 :

Question 1 :

- **Voting Classifier (Soft) :** Combine les forces des autres modèles le scores est équilibrés entre précision, rappel et F1-score.
- **Logistic Regression :** Est très proche en termes d'AUC-ROC mais a un F1-score légèrement inférieur.
- **Naive Bayes** : A un excellent F1-score mais un AUC-ROC légèrement inférieur.
- **SVM :** Est performant mais légèrement moins robuste pour la classe spam 19 (FP).

Question 2 :

Analyse temps d'entrainement :

Naïve Bayes	0.13 secondes
SVM	22.84 secondes
Regression logistique	0.15 secondes
Classifier Hard	33.99 secondes
Classifier Soft	16.75 secondes

Observations :

- Naive Bayes et Logistic Regression sont très rapides et légers, adaptés pour des tâches où les ressources sont limitées.
- SVM est beaucoup plus exigeant en temps et en ressources.
- Voting Classifiers combinent plusieurs modèles, ce qui explique le cout plus élevé.

Question 3 :

Préférences pour Voting Classifier :

- Données complexes : Lorsque les données sont très variées, le Voting Classifier peut équilibrer les forces de plusieurs modèles.
- Robustesse : Si les performances d'un seul modèle sont insuffisantes comme dans des taches très légers et rapide, le Voting Classifier peut réduire les erreurs.
- Equilibre : Pour un bon compromis entre précision et rappel, le Voting Classifiers (Soft) sont souvent plus robustes.

Question 4 :

Modeles	Avantages	Limites
SVM	Performant pour des données linéaires	Lent pour les gros datasets
GMM	Flexible avec des données non gaussienne	Plus adapté pour des taches de clustering que pour la classification
Voting classifier	Combine plusieurs modeles benefice donc la force de chacun	Peut être très couteux à entrainer !

Question 5 :

Dans ce TP nous avons pu voir que des modèles légers sont plus adapter pour des prédiction rapide (naive Bayes) alors que des modèles comme le voting classifier sont plus couteux mais on de meilleure performance sur de grande base de données. Avec une bonne optimisation de chaque modèle et un bon mélange de modèle dans le Classifier voting on peut avoir de bon compromis entre rapidité et précision.

Annexes :

Matrix Logistic Regression :

Matrice de confusion :

		Réalité	
		Négative : 0	Positive : 1
Prédiction	Négative : 0	955	0
	Positive : 1	22	128

SVM

Matrice de confusion :

		Réalité	
		Négative : 0	Positive : 1
Prédiction	Négative : 0	960(VN)	5(FN)

	Positive : 1	19(FP)	131(VP)

Rapport de classification:

	Metric precision	recall	f1-score	support
0	0.98	1	0.90	965
1	1	0.85	0.90	150
Accuracy			0.98	1115
Macro avg	0.90	0.93	0.95	1115
weighted avg	0.98	0.98	0.98	1115

Rapport de classification

	Metric precision	recall	f1-score	support
0 (ham)	0.98	0.99	0.99	965
1 (spam)	0.96	0.87	0.92	150
Accuracy			0.98	1115
Macro avg	0.97	0.93	0.95	1115
weighted avg	0.98	0.98	0.98	1115

Naives Bayes :

Matrice de confusion :

		Réalité	
		Negative: 0	**Positive: 1**
Prédiction	**Negative: 0**	957	8
	Positive : 1	12	138

Rapport de classification :

	Metric precision	recall	f1-score	support
0	0.99	0.99	0.99	965
1	0.95	0.92	0.93	150
Accuracy			0.98	1115
Macro avg	0.97	0.96	0.96	1115
weighted avg	0.98	0.98	0.98	1115

Classifier voting Soft :

Matrice de confusion :

		Réalité	
		Negative: 0	**Positive : 1**
Prédiction	**Negative: 0**	994	1
	Positive : 1	18	132

Rapport de classification :

	Metric precision	recall	f1-score	support
0	0.98	1	0.99	965
1	0.99	0.88	0.93	150
Accuracy			0.98	1115
Macro avg	0.99	0.94	0.96	1115
weighted avg	0.98	0.98	0.98	1115

GMM :

Matrice de confusion :

		Réalité	
		Negative : 0	**Positive : 1**
Prédiction	**Negative : 0**	962	3
	Positive : 1	36	114

Rapport de classification :

	Metric precision	recall	f1-score	support
0	0.96	1	0.98	965
1	0.97	0.76	0.85	150
Accuracy			0.97	1115
Macro avg	0.97	0.88	0.92	1115
weighted avg	0.97	0.97	0.96	1115

Classifier voting Hard :

Matrice de confusion :

		Réalité	
		Negative: 0	**Positive : 1**
Prédiction	**Negative : 0**	964	1
	Positive : 1	20	130

Rapport de classification :

	Metric precision	recall	f1-score	support
0	0.98	1	0.99	965
1	0.99	0.87	0.93	150
Accuracy			0.98	1115
Macro avg	0.99	0.93	0.96	1115
weighted avg	0.98	0.98	0.98	1115

Figure 3 : Tableau des performances des différents modèles