



RÉPUBLIQUE  
FRANÇAISE

*Liberté  
Égalité  
Fraternité*

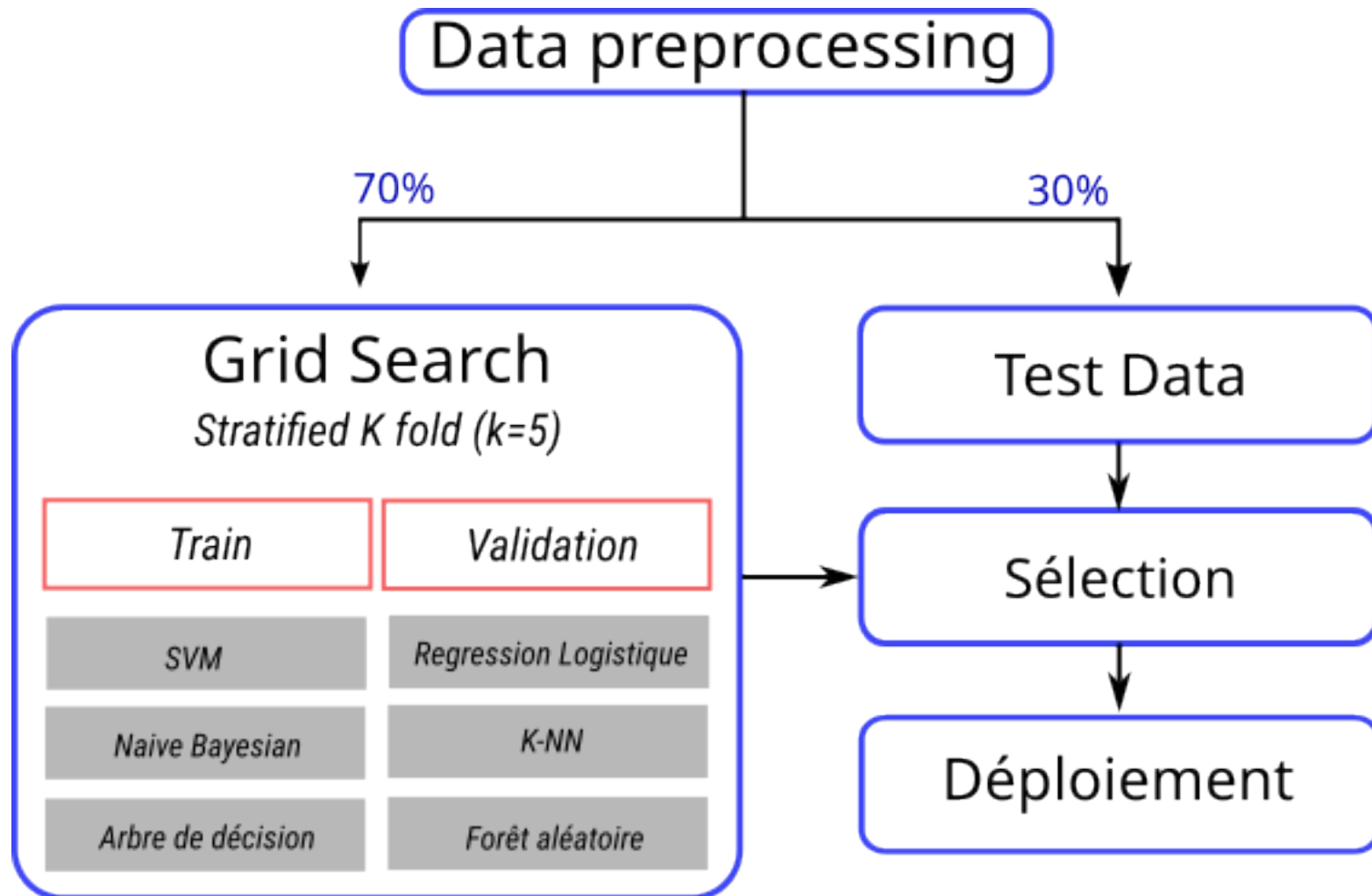


CENTRE  
VAL DE LOIRE )))


# Classifieur de SPAM SMS

Jonathan Caillaux

# Workflow



# Origine des données



## SMS Spam Collection

Donated on 6/21/2012

<https://archive.ics.uci.edu/dataset/228/sms+spam+collection>

The SMS Spam Collection is a public set of SMS labeled messages that have been collected for mobile phone spam research.

Dataset Characteristics	Subject Area	Associated Tasks
Multivariate, Text, Domain-Theory	Computer Science	Classification, Clustering

Feature Type	# Instances	# Features
Real	5574	-

### Dataset Information

#### Additional Information

This corpus has been collected from free or free for research sources at the Internet:

- > A collection of 425 SMS spam messages was manually extracted from the Grumbletext Web site. This is a UK forum in which cell phone users make public claims about SMS spam messages, most of them without reporting the very spam message received. The identification of the text of spam messages in the claims is a very hard and time-consuming task, and it involved carefully scanning hundreds of web pages. The Grumbletext Web site is: <http://www.grumbletext.co.uk/>.
- > A subset of 3,375 SMS randomly chosen ham messages of the NUS SMS Corpus (NSC), which is a dataset of about 10,000 legitimate messages collected for research at the Department of Computer Science at the National University of Singapore. The messages largely originate from Singaporeans and mostly from students attending the University. These messages were collected from volunteers who were made aware that their contributions were going to be made publicly available. The NUS SMS Corpus is available at: <http://www.comp.nus.edu.sg/~rpnlpir/downloads/corpora/smsCorpus/>.
- > A list of 450 SMS ham messages collected from Caroline Tagg's PhD Thesis available at <http://etheses.bham.ac.uk/253/1/Tagg09PhD.pdf>.

Les données proviennent de 3 sources différentes : un site internet, un manuscrit PDF et le corpus NSC

# Preprocessing

1. Suppression des doublons
2. Suppression des caractères non-imprimables
3. Suppression de caractères non ASCII
4. Suppression des entités HTML
5. Normalization remplacement des urls et emails par les mots URL et EMAIL
6. Remplacement des émoticônes par le mot EMOTICON
7. Tokenization
8. Suppression des Stop Words
9. Lemmatization

```
[ 'ú', 'ü', 'Ü', 'è', 'é', 'É', 'ì', ' 鋏 ',  
 ' 隼 ', '\x92', "'", '-', '...', '\x94', '"', \  
x91', "'", '\x93', '\x96', '»', '—', '†', 'i' ]
```

# Preprocessing

1. Suppression des doublons
2. Suppression des caractères non-imprimables
3. Suppression de caractères non ASCII
4. Suppression des entités HTML
5. Normalization : remplacement des urls et emails par les mots  
URL et EMAIL
6. Remplacement des émoticônes par le mot EMOTICON
7. Tokenization
8. Lemmatization
9. Suppression des Stop Words

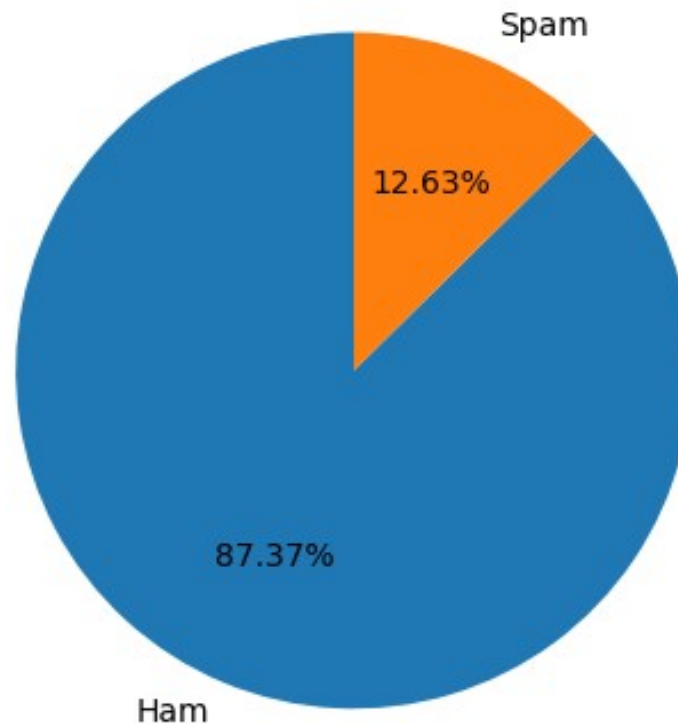
```
['&', '<', '>']
```

```
['<#>', '<EMAIL>', '<URL>',  
'<TIME>', '<DECIMAL>']
```

# Preprocessing

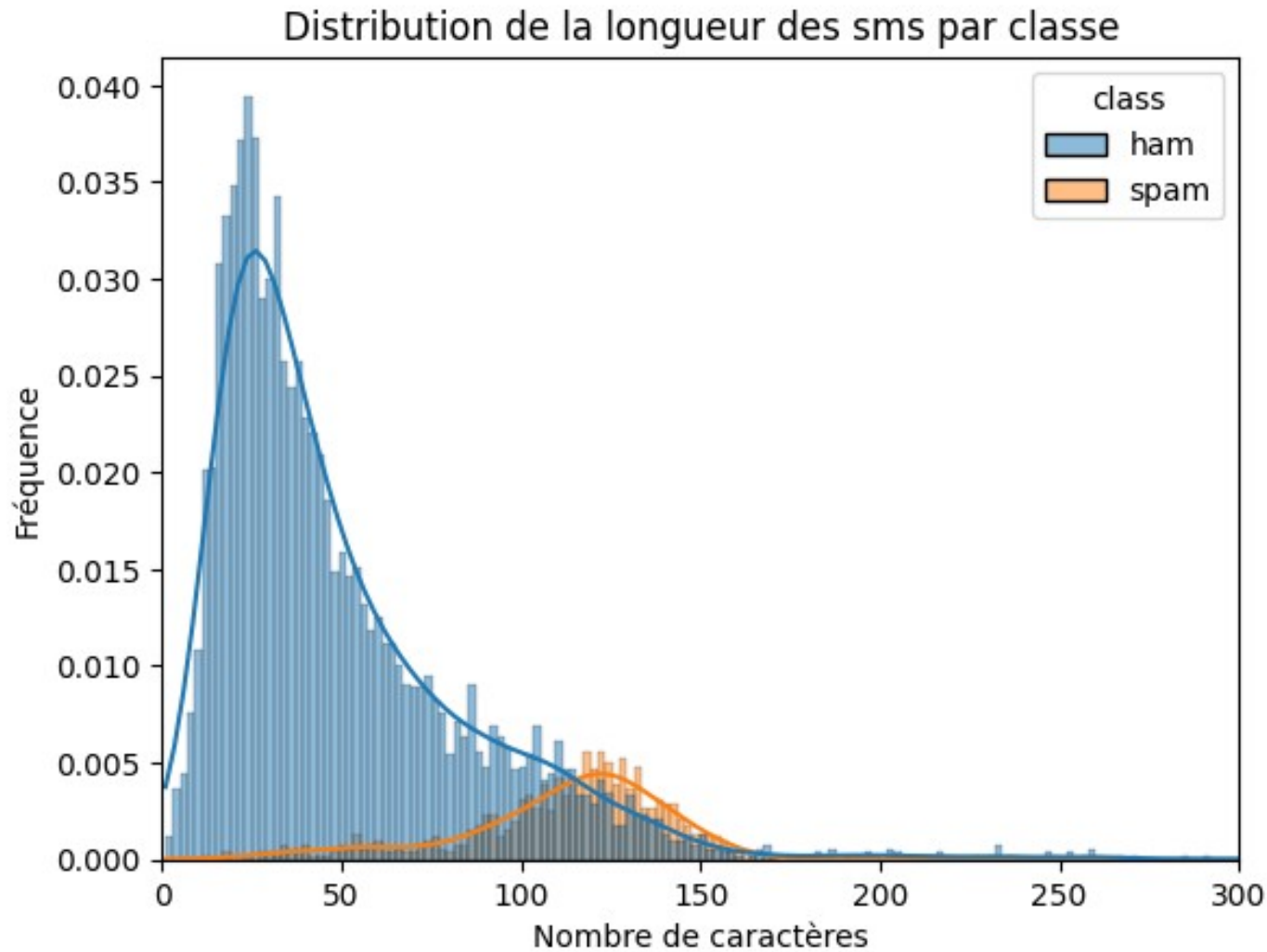
A la fin on obtient 5171 sms

Répartition des Classes  
après suppression des doublons



Le dataset est très déséquilibré

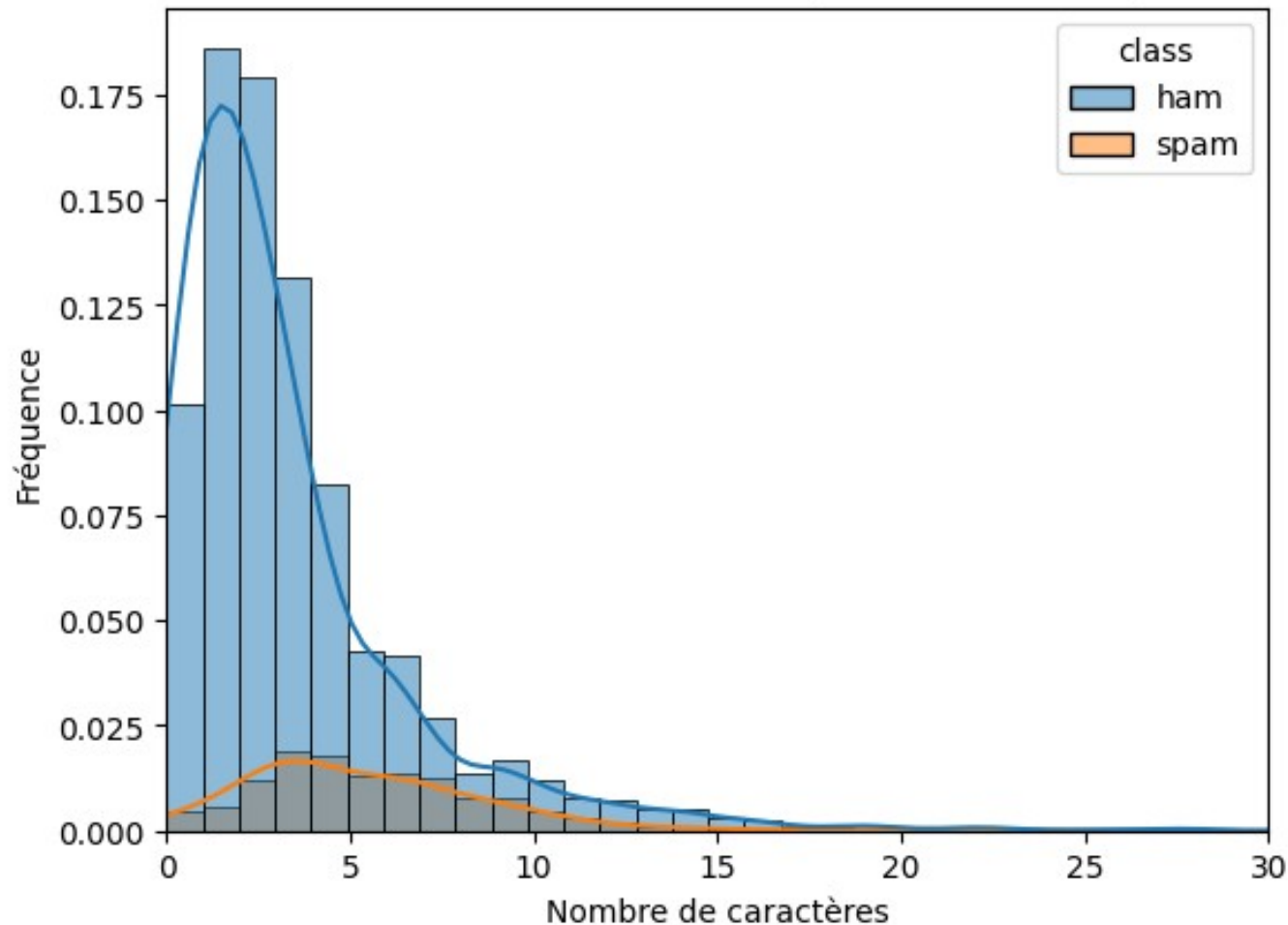
# Preprocessing



En moyenne les spams sont plus longs.

# Preprocessing

Distribution du nombre de caractères spéciaux des sms par classe



Il semble que les spams contiennent en moyenne plus de caractères spéciaux. Probablement du aux symboles \$£@ %



# GridSearch

**Modèles retenus :**

SVM (C)	Régression Logistique Avec Pénalité L2
Naive Bayes (variance, alpha)	K-NN (k, distance)
Arbre de Décision (critère, profondeur max)	Forets Aléatoires (critère, profondeur max)

**Métriques :** accuracy, precision, recall, **f1-score**

**Cross-Validation :** StratifiedKFold (k = 5)

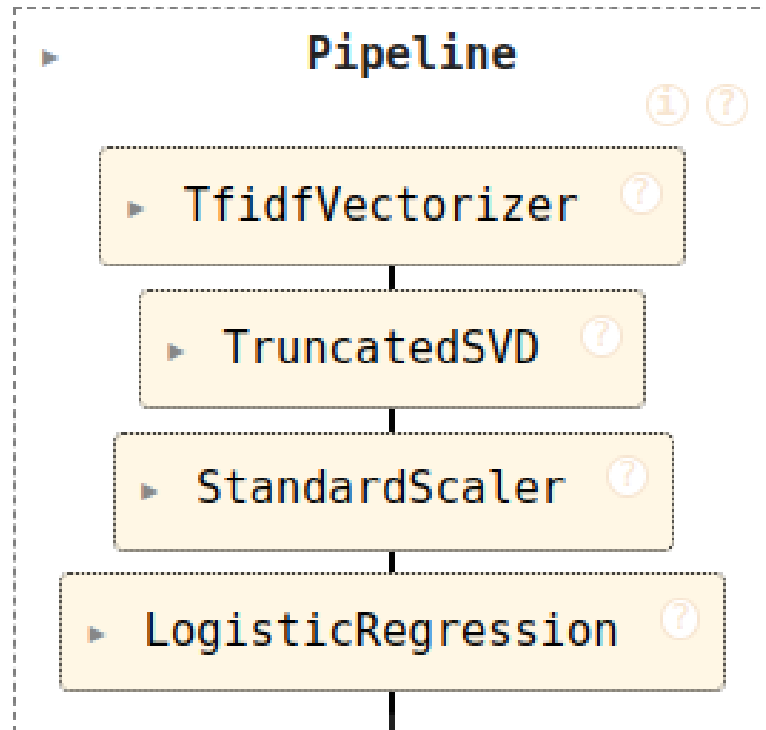
# GridSearch

Cross-Validation : train set est séparé en k folds. A chaque iteration un fold différent est sélectionné pour la validation

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
First Iteration	Test				
Second Iteration		Test			
Third Iteration			Test		
Fourth Iteration				Test	
Fifth Iteration					Test

**Cross-Validation** : StratifiedKFold ( $k = 5$ )

# Pipelines



- TF-IDF Vectorizer
- SVD Tronquée (n=675)
- Un Scaler (Standard/MinMax)
- Un Classifieur

L'utilisation d'une SVD tronquée permet de réduire dimensionnellement les caractéristiques sans avoir à jouer sur le paramètre `max_feature` du Tf-IDF vectorizer

# Résultats GridSearch

## Modèles retenus :

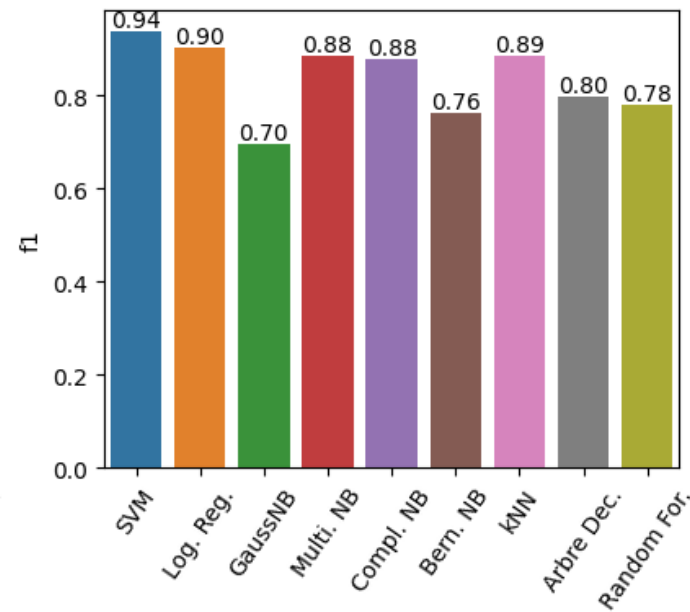
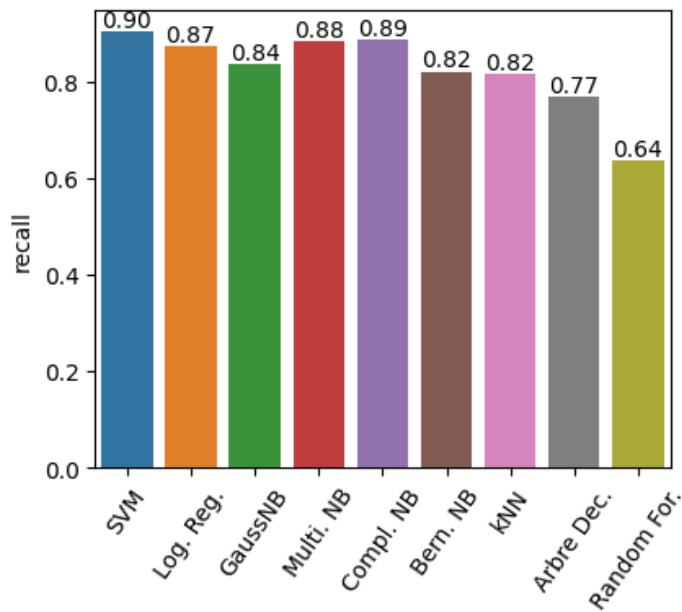
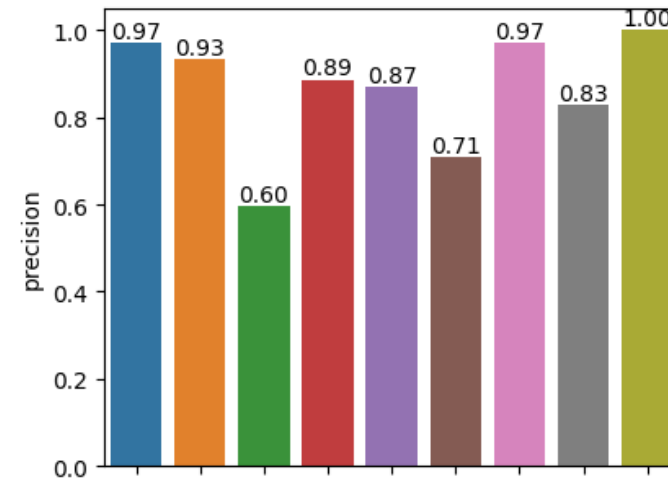
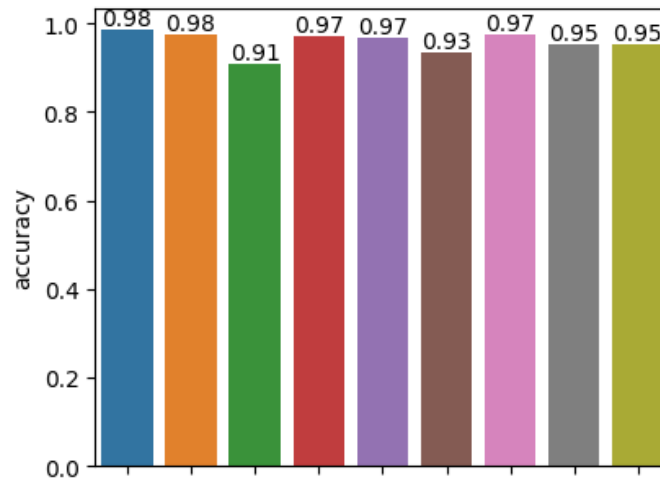
SVM (C= 2.625)	Régression Logistique Avec Pénalité L2 (C = 0.15)
Naive Bayes (variance, alpha) (s=1e-9   a=0.5   a=1.5   a=1.0)	K-NN (k=9, distance=cosine)
Arbre de Décision (critère=gini, profondeur max=5)	Forets Aléatoires (critère='entropy', profondeur max=13)

Paramètres sélectionnés par rapport au f1-score.

**Cross-Validation** : StratifiedKFold (k = 5)

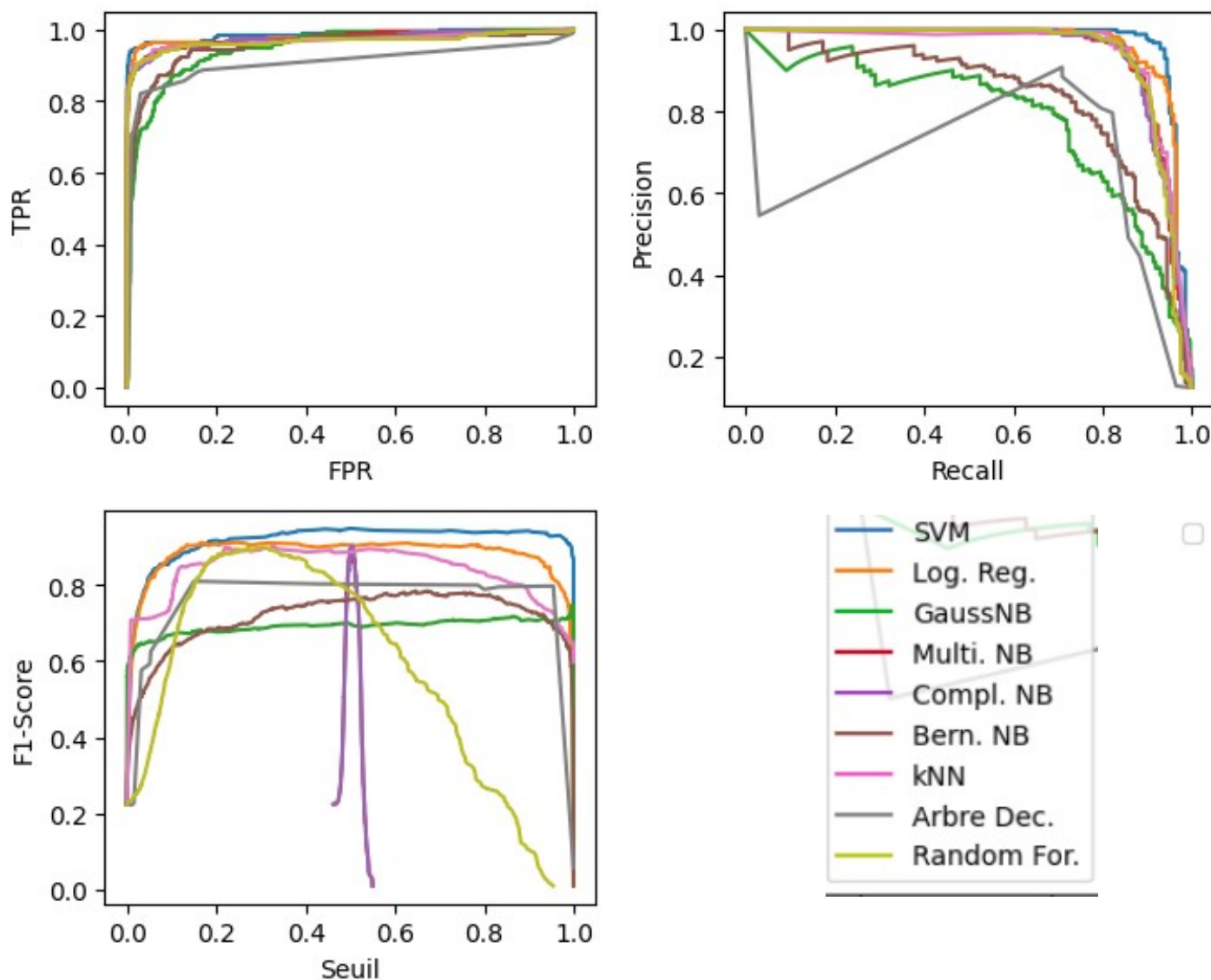
# Comparaison Métriques

Comparaison des Modèles



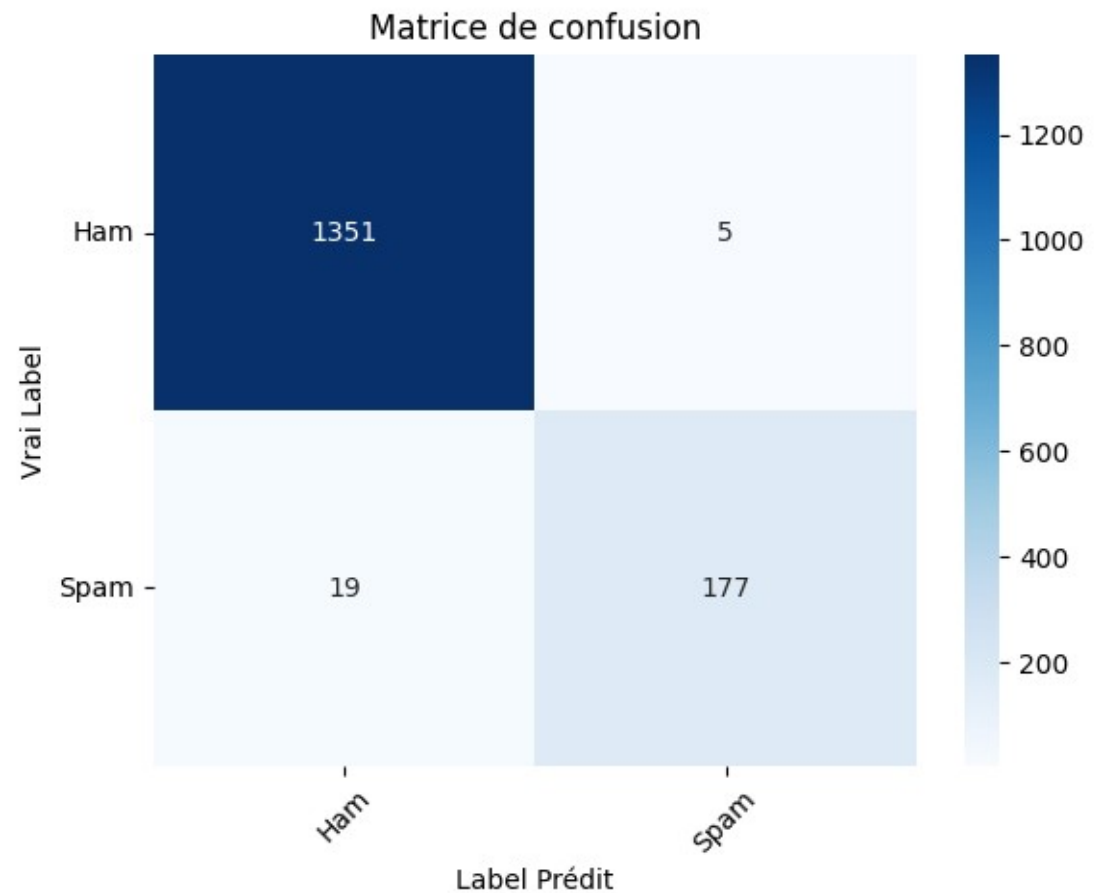
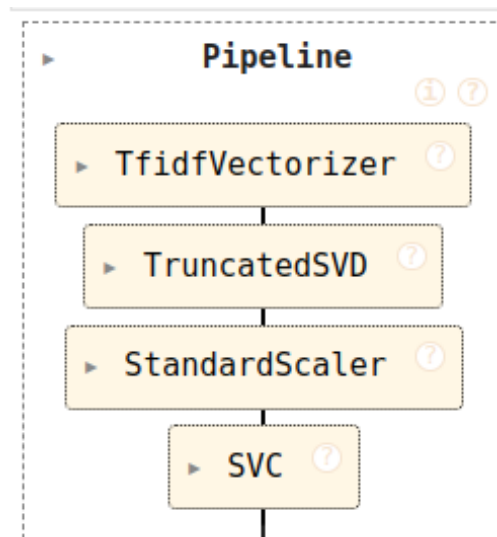
# Effet du Seuil

Effet du Seuil de Décision



# Conclusion

Le modèle correspondant à nos besoin est le classificateur SVM



# SpamSift

## Spamlit

SMS Text

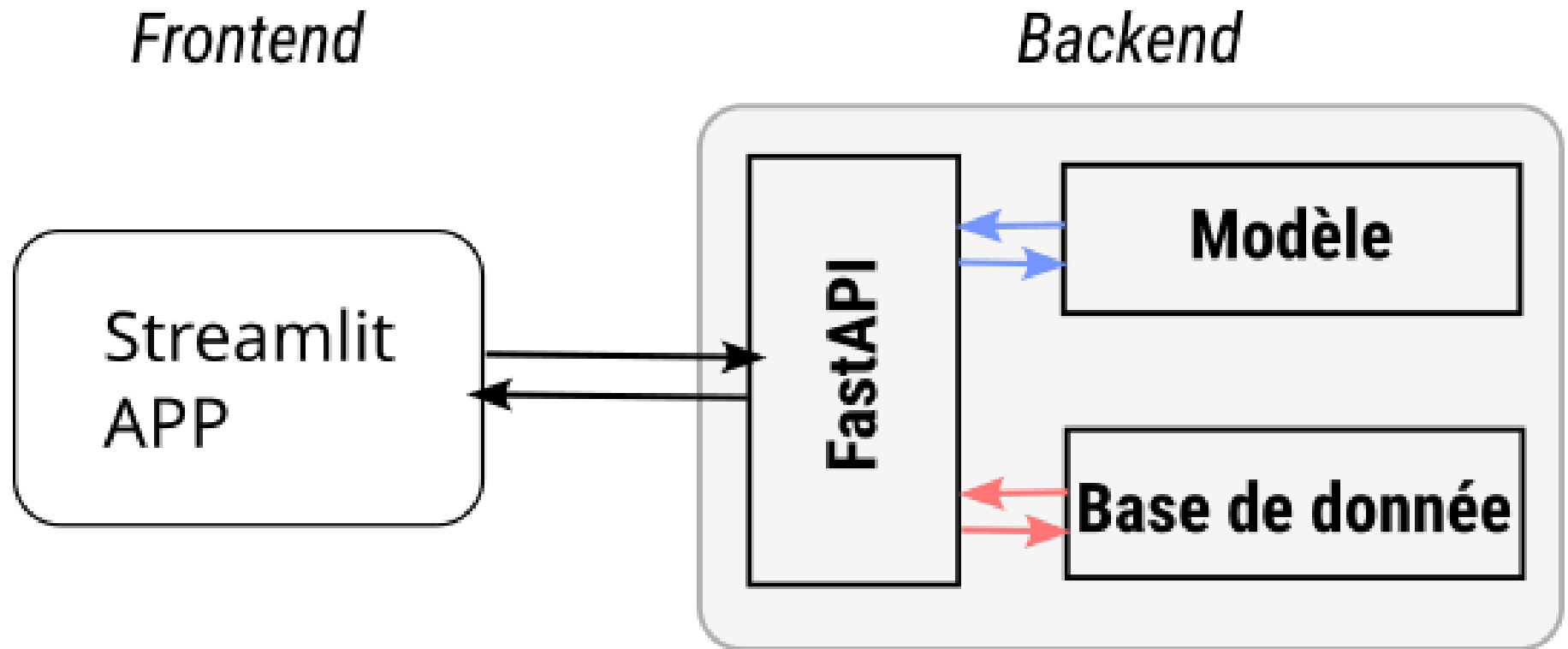
want new nokia 3510i colour phone deliveredtomorrow ? 300 free minute mobile + 100 free text + free camcorder reply call 08000930705

Vérifier

Le message est un spam

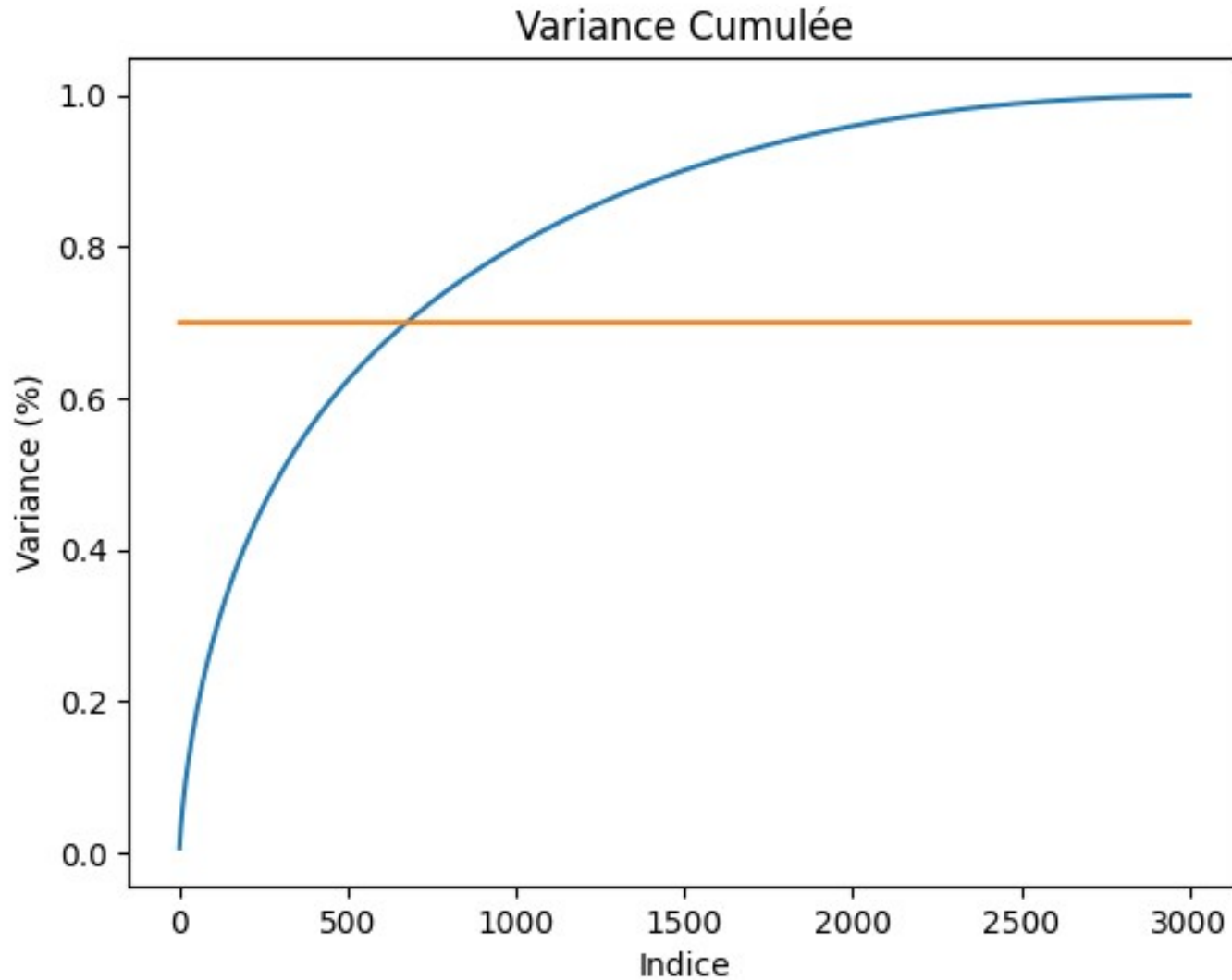


# SpamSift



Merci pour votre attention

# Variance SVD



Seuil de 70 % atteint pour  $n > 675$