Université de Paris

Text mining : exploration statistique de corpus de courriels ou de tweets

Consigne : Travaux à réaliser par groupes de 2 à 3.
Vous avez le choix entre les deux sujets
Date de rendu 31 janvier à 00 :00

Les données spam sont décrites dans le document
https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-scenar-explo-spam.pdf
Dans ce même document vous disposez une étude complète mettant en œuvre plusieurs méthodes que vous avez vues en cours. En suivant les différentes étapes organisées principalement en deux approches, vous serez en mesure de maitriser plusieurs méthodes utiles dans le cadre du text-mining. Ces étapes permettront d'extraire de la connaissance de l'ensemble des données dans un contexte d'apprentissage non supervisé et supervisé.

A. Travail à faire en respectant le document, toutes les étapes vous permettront de revoir plusieurs méthodes.

1. *D'abord visualiser les données et bien identifier la nature des variables.*

```
spam=read.table("https://www.math.univ-toulouse.fr/~besse/Wikistat/data/spam.dat", header=TRUE)
dim(spam)
summary(spam)
```

2. *Une standardisation vous est proposée, justifier ce choix.*
3. *A l'aide de la librairie {Factoshiny}, appliquer l'ACP et extraire les informations les plus utiles.*
4. *Exécuter le code de chaque étape et évaluer son intérêt dans l'étude.*

Sujet 1. La base *spam* brute à traiter est décrite dans
https://github.com/mohitgupta-omg/Kaggle-SMS-Spam-Collection-Dataset-/blob/master/spam.csv
On décide de construire simplement une matrice de co-occurrences. Le Travail dans A) a nécessité une standardisation de vos données, ici on s'appuiera sur cette nouvelle table et on vise à comparer deux approches celle que vous avez testée en A) et une autre qui ne nécessitera pas de transformation de la matrice. Vous pouvez vous contenter de prendre en compte uniquement des mots si vous le jugez suffisant eu regard votre expérience dans A).

1. A l'aide de la librairie {Factoshiny}, appliquer une méthode d'analyse factorielle appropriée et visualiser les deux catégories de courriels, que peut-on dire ? Des graphiques allégés suivant des indices de contribution ou de qualité de représentations seront appréciés.
2. Appliquer un algorithme de classification approprié avec un nombre de classes égal à 2, construire la matrice de confusion sur la base de la variable spam. Calculer le taux d'accuracy, NMI et ARI.
3. Que donne un partionnement avec ce même algorithme mais en 3 classes ? Visualiser ces classes
4. Appliquer un co-clustering approprié ; vous pouvez utiliser les packages
    a) {Blockmodels} https://arxiv.org/pdf/1602.07587.pdf
    b) {Blockcluster} https://hal.inria.fr/hal-01093554/document
    c) {Coclust} https://www.jstatsoft.org/article/view/v088i07

Dans {Coclust} vous pouvez utiliser les différents algorithmes proposés qui sont tous adaptés à ce type de données. Par contre avec {Blockmodels} et {Blockcluster} seul le modèle de Poisson est approprié ; d'ailleurs rien ne vous empêche de tester le modèle Gaussien pour comparer.

5. Visualiser ces classes à l'aide de méthodes de réduction de dimension.
6. Reprendre l'étape 5 en appliquant une pondération de type tf-idf.
7. *Facultatif : Pour aller plus loin, on peut utiliser des modèles de von-Mises Fischer.*

Sujet 2. Voici un challenge intéressant sur des tweets. Dans A) vous avez pu étudier avec diverses méthodes des données de type emails. Dans ce sujet on peut faire la même étude sur les données tweets qui sont des textes courts https://www.kaggle.com/gpreda/pfizer-vaccine-tweets

1. D'abord il faut se familiariser avec le package R pour construire votre matrice tweets x termes http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/fr_Tanagra_tweets_analysis.pdf
2. A l'aide de la librairie {Factoshiny}, appliquer une méthode d'analyse factorielle appropriée et visualiser les tweets et les mots? Des graphiques allégés suivant des indices contribution ou de qualité de représentations seront appréciés.
3. Appliquer un algorithme de classification approprié avec un nombre de classes estimé.
4. Visualiser ces classes à l'aide de méthodes de réduction de dimension.
5. Appliquer un co-clustering approprié ; vous pouvez utiliser les packages
    a) {Blockmodels} https://arxiv.org/pdf/1602.07587.pdf
    b) {Blockcluster} https://hal.inria.fr/hal-01093554/document
    c) {Coclust} https://www.jstatsoft.org/article/view/v088i07

    Dans {Coclust} vous pouvez utiliser les différents algorithmes proposés qui sont tous adaptés à ce type de données. Par contre avec {Blockmodels} et {Blockcluster} seul le modèle de Poisson est approprié ; d'ailleurs rien ne vous empêche de tester le modèle Gaussien pour comparer.
6. Reprendre l'étape 5 en appliquant une pondération de type tf-idf.
7. *Facultatif : Pour aller plus loin, on peut utiliser des modèles de von-Mises Fischer.*