Analyse prédictive et factorielle sous SCILAB

Pierrick Michel
Wahiba Azzoug
EL Mehdi EL ALAMI KAABOUCH
Kafil ELkhadir

Master 2 Statistiques et Informatique pour la science des données 2017-2018

Plan

1 Introduction

2 Analyse statistique avec Scilab

3 Régression linéaire simple et multiple

4 Analyse Discriminante

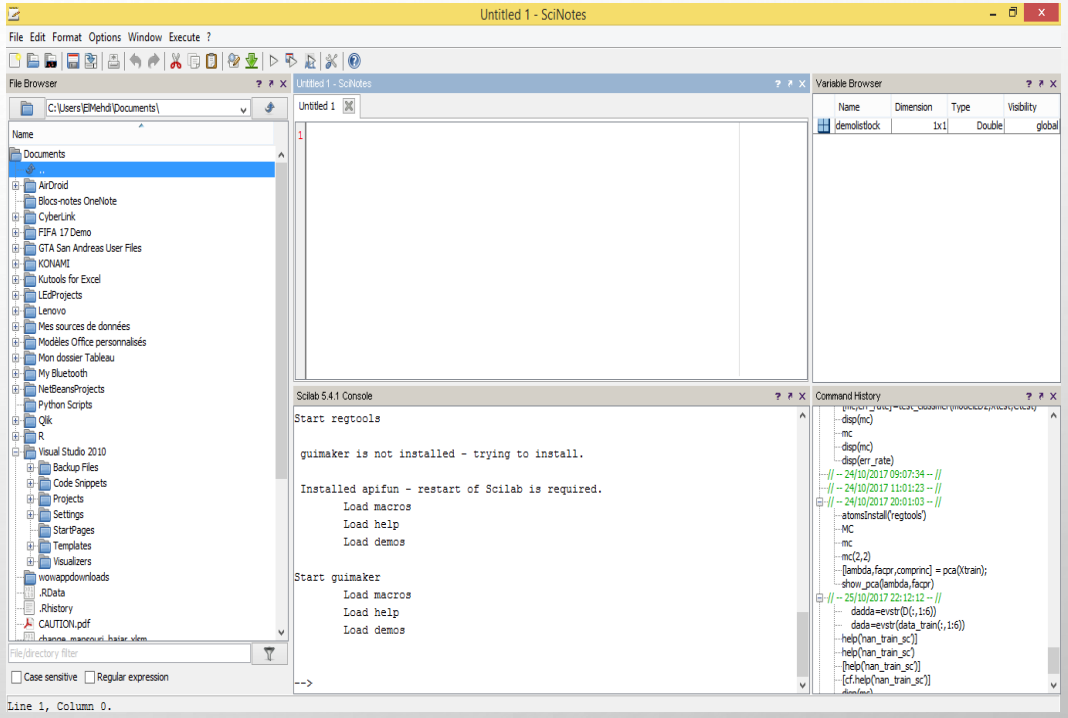
5 **Analyse en composantes principales (Exemple)**

1. Introduction

Dans le cadre des ateliers techniques de formation aux outils, nous avons voulu réaliser des tests statistiques à l'aide du logiciel Scilab.

Ce logiciel ressemble beaucoup à Rstudio en termes d'interface et de capacités à réaliser des analyses statistiques

Utilisation de la version 5.4.1 plutôt que 6 car packages non mis à jour.



1. Introduction

Téléchargement en ligne du package utile pour nos analyses, installation du package:

```
atomsInstall("C:\Users\Desktop\M2 SISE\Scilab\nan_1.3.4-1.bin.x64.windows.zip")
```

Les fonctions du package

```
Data Correlation and Covariance
  "nan_covm"
  "nan_ecovm"
  "nan_decovm"
  "nan_xcovf"
  "nan_conv2nan"
  "nan_cor"
  "nan_cov"
  "nan_corrcoef"
  "nan_corrcov"
  "nan_rankcorr"
  "nan_partcorrcoef"
  "nan_tiedrank"
```

```
hypothesis Tests
  "nan_ttest"
  "nan_ttest2"
```

```
Classification
"nan_train_sc"
"nan_test_sc"
"nan_classify"
"nan_xval"
"nan_kappa"
"nan_train_lda_sparse"
"nan_fss"
"nan_cat2bin"
"nan_row_col_deletion"
"nan_mahal"
"nan_rocplot"
"nan_confusionmat"
"nan_partest"

cluster Analysis
"nan_kmeans"
  "nan_pdist"
  "nan_linkage"
  "nan_squareform"
 "nan_cluster"
 nan_crosstab"
```

Il existe peu de package de statistique sous Scilab,
-Nan
-Libvsm
-Regtools

Les packages ne sont pas remis à jour pour la derniere versions de Scilab.

2. Analyse statistiques avec Scilab

Les analyses que nous avons choisi de traiter sont :

- Régression linéaire simple
- Régression linéaire multiple
- Analyse discriminante

Il existe des packages sous Scilab pour réaliser ces analyses.

Le but étant d'apprendre à coder avec ce logiciel, seuls les packages «nan» et «libsvm» sont nécessaires.

3. Régression linéaire simple et multiple

Y est une variable quantitative qui est expliquée, par une ou plusieurs variables quantitatives Xj(1,,,,p)

Notre objectif est :

- estimer les paramètres du modèle par la méthode des moindres carrées

$$\hat{a} = (X'X)^{-1}X'Y$$

- évaluer globalement la pertinence du modèle

$$F_{calc} = \frac{\frac{R^2}{p}}{\frac{1-R^2}{n-p-1}},$$

Jeu de données :
Parmi l'un des polluants dans l'air nous retrouvons le NO2. Selon la concentration d'O3 dans l'air et des températures journalières nous allons prédire la concentration de ce gaz.

```
function [b ] = inverse(X)

//**********************Pré-traitement
//**********************Calculer l'inverse
b =inv(X'*X)

endfunction

//// DATA
//Variables
X1=strtod(data_ozone(1:61,2:2))
X2=strtod(data_ozone(1:61,3:3))
X3=strtod(data_ozone(1:61,4:4))
X4=strtod(data_ozone(1:61,5:5))
X5=strtod(data_ozone(1:61,6:6))

//Création du matrice
X=[ones(61,1) X1 X2 X3 X4 X5 ]//X6 X7]
```

4. L'analyse discriminante

➔ Est Une extension de la régression dans le cas où la variable à expliquer est qualitative.

L'analyse discriminante sous SCILAB :

(1) charger l'échantillon d'apprentissage
(2) le coder en implémentant une fonction

```
(1)  Data_train=csvRead("C:\Users\ElMehdi\Desktop\SCILAB\data.txt","\t",".","string",[],[],[2 1 150 9])
(2)  function [descripteurs, cible]=recodage(base)
     ..
     descripteurs: evstr(base(:,1:8)) //transformer les colonnes 1 à 8 en numérique
      y=base(:,9:9) //Récupération de la variable cible
     .....
```

(3) Construction du modèle prédictif avec la Toolbox « Nan » - une libraire de l'apprentissage automatique.

```
datatype: "classifier:statistical:ld2"
Labels: [1,2]
MD: hypermat
NN: hypermat
weights: [7x2 constant]
```

```
(3)  Il faut avoir installé et chargé la Toolbox « Nan » pour pouvoir poursuivre.
     atomsInstall('nom de la toolbox') // pour les installer
     atomsLoad('nom de la toolbox') // pour les charger
     modelLD2=nan_train_sc(XTrain,CTrain,'LD2') //apprentissage - méthode LD2
     disp(modelLD2)
```

Evaluation du modèle sur un échantillon test

(1) charger l'échantillon test
(2) le coder en respectant le schéma utilisé pour le fichier d'apprentissage

```
     //chargement des données - format chaîne
     Data_test=csvRead("C:\Users\ElMehdi\Desktop\SCILAB\data.txt","\t",".","string",[],[],[2 1 60 9])
     //le coder en utilisant la fonction recodage
     [XTest,CTest]=recodage(Data_test)
     disp(tabul(data_test)) // Pour afficher le nombre d'observation dans chaque classe.
```

(3) produire la prédiction en appliquant le modèle sur l'échantillon test,
(4) (4) créer une fonction pour la matrice de confusion en confrontant les valeurs prédites par le modèle avec celles observées dans l'échantillon test
(5) (5) déduire de la matrice de confusion le taux d'erreur.

```
     (3) et (4) et (5)
     function [MC, ERR_RATE]=test_classifier(classifier, descripteurs, cible) //Fonction pour l'évaluation d'un classifieur
     pred=nan_test_sc(classifier,descripteurs) //prédiction
     //matrice de confusion : cible observée vs. prédiction
     MC=nan_confusionmat(cible,pred.classlabel)
     //taux d'erreur
     ERR_RATE = 1.0 - sum(diag(MC))/sum(MC)
     endfunction
     //évaluation
     [mc,err_rate] = test_classifier(modelLD2,XTest,CTest)
```

4. L'analyse discriminante

➔ **L'objectif** :Prédire la classe d'un nouvel objet décrit par la valeur de ces attributs.

Exemple :

Donner une prévision d'avoir un cancer de prostate ou pas ,d'une personne de 64 ans ayant un niveau de sérum de 0.4 et qui a des résultats d'une radiographie positive et sa taille grande ainsi leur résultat d'une biopsie moins sérieux .

Résultats de l'analyse discriminante sous SCILAB : disp(modelLD2.weights) ➔Pour accéder aux champs de l'objets weights.

```
datatype: "classifier:statistical:ld2"
Labels: [1,2]
MD: hypermat
NN: hypermat
weights: [7x2 constant]

  0.5330020  - 0.5330020
- 0.0075632    0.0075632
- 0.0060413    0.0060413
  0.4370939  - 0.4370939
  0.4058681  - 0.4058681
  0.7191982  - 0.7191982
```

Y= a0 +a1*age+a2*nv_serum+a3*radio+a4*taille+a5*biop

5. Analyse en composantes principales

Exemple :
a=rand(100,10,'n'); [lambda,facpr,comprinc] = pca(a);
show_pca(lambda,facpr)

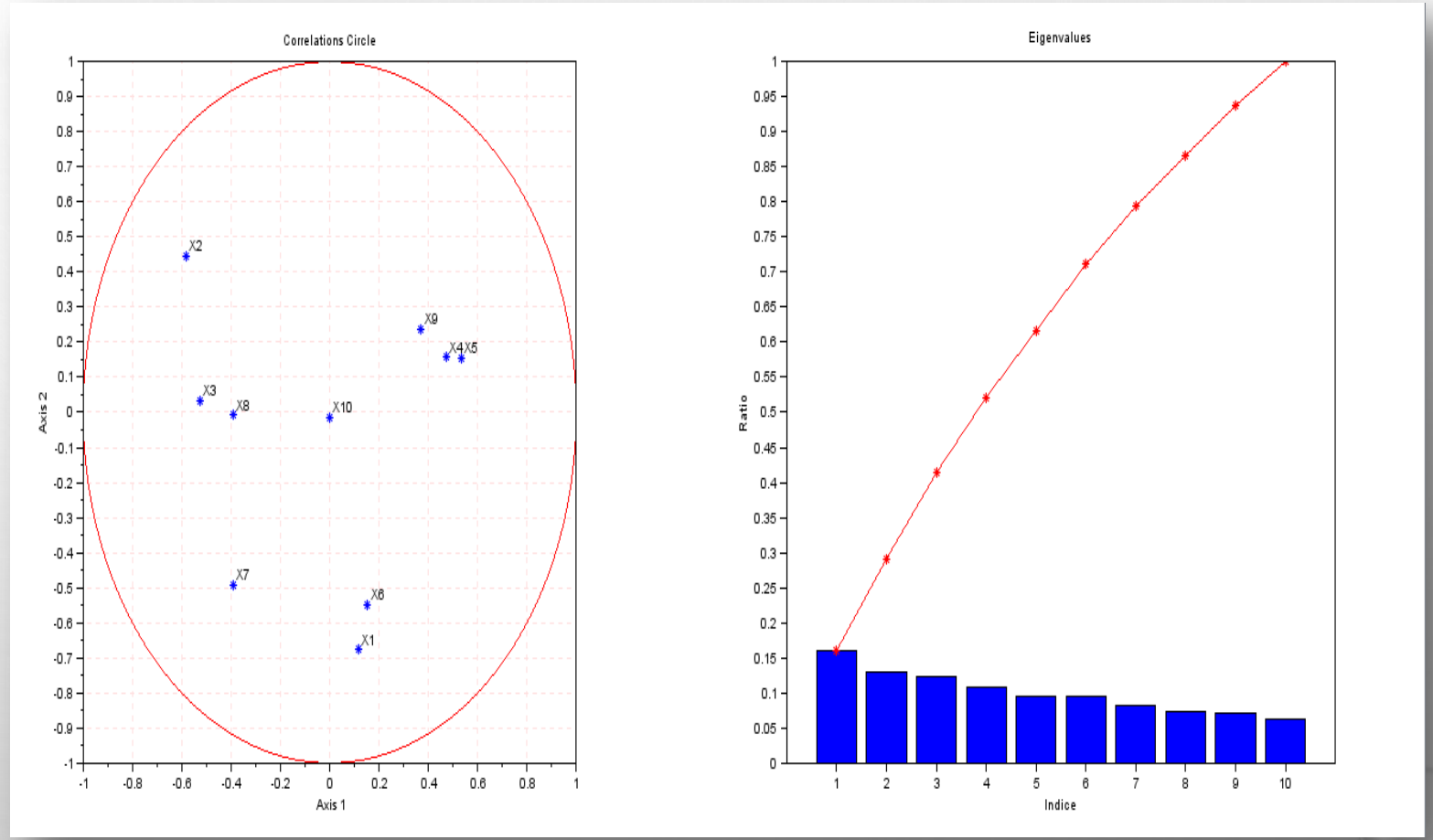
Lambda : est une matrice numérique p x 2. Dans la première colonne, nous trouvons les valeurs propres de V, où V est la matrice de corrélation p x p et dans la seconde colonne, les rapports de la valeur propre correspondante sur la somme des valeurs propres.

Facpr : sont les principaux facteurs: vecteurs propres de V. Chaque colonne est un vecteur propre du dual de R ^ p.

comprinc : sont les principaux composants. Chaque colonne (c_i = Xu_i) de cette matrice n x n est la projection M-orthogonale des individus sur l'axe principal. Chacune de ces colonnes est une combinaison linéaire des variables x1, ..., xp avec la variance maximale sous condition u'_i M ^ (- 1) u_i = 1

show_pca - Visualisation des résultats de l'analyse des composants principales

princomp - Analyse en composantes principales



MERCI POUR VOTRE ATTENTION !