NOTICE FOR R-CODE KF.R

CHRISTOPHE GIRAUD

CHRISTOPHE.GIRAUD@POLYTECHNIQUE.EDU

1. Introduction

The KF estimator is described in "Low Rank Multivariate Regression" `arXiv:1009.5165v1`. It handles the multivariate regression setting

$$Y = XA + \sigma E$$

where Y is a $m \times n$ matrix of response variables, X is a $m \times p$ matrix of predictors, A is $p \times n$ matrix of regression coefficients and E is a $m \times n$ noise matrix with i.i.d. $\mathcal{N}(0,1)$ entries. The matrix Y and X are assumed to be known, while A, σ and E are unknown. The KF procedure is adapted to the case where A is expected to have a small rank r, the rank r being unknown. The main feature of KF is to select this rank in such a way that the final estimator is nearly as good as if we knew r.

2. How can I use this code ?

To use the KF estimator first load the file `KF.R` :

```
> source("KF.R")
```

A demo is available in the file `demo.R`

Example:

```
> data <- generateData(100,25,25,0.5,0.5,10)  # generate synthetic data
> names(data)
> result <- KF(data$Y,data$X)  # estimate A
> names(result)
```

Below we provide details on the three main functions `KF`, `pen` and `generateData`

3. Main Function KF

Description. Main function. Compute the KF estimator from data Y and X.

CALL : `KF(Y, X, rmax=min(dim(X)[1],dim(X)[2],dim(Y)[2]), K=2, CV=FALSE, V=10, rkX=0, tol=10**(-6))`

INPUT :
`Y` : $m \times n$ matrix of response variables.
`X` : $m \times p$ matrix of predictors.
`rmax` : maximum value of the rank. Default is `min(dim(X)[1],dim(X)[2],dim(Y)[2])`.
`K` : tuning parameter of the procedure. Default is `K=2`/
`CV` : if TRUE `V`-fold Cross-Validation is used to select `K`. Default is FALSE.
`V` : number of split in the `V`-fold CV. No effect if `CV=FALSE`.
`rkX` : rank of `X`. By default this rank is evaluated during the call to `KF`.
`tol` : tolerance on singular values for computing the rank of `X`.

OUTPUT : list of elements :
`A` : `length(K) x p x n` array. `A[k,,]` = estimated matrix $\widehat{A}$ for $K =$ `K[k]`.
`XA` : `length(K) x m x n` array. `XA[k,,]` = $X\widehat{A}$ for $K =$ `K[k]`.
`rank` : `length(K)` vector. `rank[k]` = rank of $\widehat{A}$ for $K =$ `K[k]`.

Examples.

```
> source("KF.R")   # load KF.R
> data <- generateData(100,25,25,0.5,0.5,10)   # generate data
> result <- KF(data$Y,data$X)   # basic call to KF
> A <- result$A        # estimator of A
> XA <- result$XA      # equals X%*%A
> rank <- result$rank  # rank of A

# ADDITIONNAL EXAMPLES
# example with different values of K and prescribed rmax
> result <-  KF(data$Y,data$X, K=c(1.5,2,2.5),rmax=15)

# example with V-Fold CV
> result <- KF(data$Y,data$X,CV=TRUE,V=20)
```

4. Auxiliary Function `pen`

Description. Compute the penalty

$$\mathrm{pen}(r) = \frac{K\mathcal{S}_{q\times n}(r)^2}{nm - 1 - K\mathcal{S}_{q\times n}(r)^2}$$

(provided for advanced use)

CALL : `pen(n, m, q, rmax, K, Nsim=200)`

INPUT :
`n,m,q` : integer (parameter of the penalty)
`rmax` : maximum rank
`Nsim` : sample size for Monte Carlo evaluation (when $nq < 1000$)
`K` : vector (tuning parameter)

OUTPUT :
return a `length(K)` x `rmax` array

5. Auxiliary Function `generateData`

Description. Generate synthetic data as in the numerical examples of `arXiv:1009.5165v1`.

CALL : `generateData(m, n, p, rho, b, r)`

INPUT :
`m,n,p,rho,b,r` : as in the description of the numerical experiments in Section 6 of `arXiv:1009.5165v1`

OUTPUT : list with elements :
`Y, A, X` : data generated according to the scheme explained in Section 6 of `arXiv:1009.5165v1`

Christophe Giraud, CMAP, UMR CNRS 7641, Ecole Polytechnique, France.