Reduced Rank Regression

In a centered reduced rank regression model, multivariate vectors $Y \in R^N$ and $X \in R^M$ satisfy $\mathrm{E}(X) = 0$, $\mathrm{E}(Y) = 0$, $\mathrm{E}(Y|X) = BX$. The $N \times M$ matrix B of regression coefficients may not be of full rank. Possible values of the rank of B range from 0 (X and Y are independent) to $\min(M, N)$ (the full rank model). If the rank is assumed to be i then model singularities occur when the true data-generating model has rank $j < i$.

Simulating reduced rank regression data

This vignette uses simulated data, so we create a convenience function `simRR()` to simulate `n` observations from a reduced rank regression model of rank `r`. The parameters `N` and `M` determine the length of Y and X, respectively; `n` is the number of observations; and `r` is the rank of the coefficient matrix B. The output is a list of matrices `X` and `Y` such that each *column* represents an individual observation (i.e. `X` is transposed with respect to the usual design matrix)

```
simRR <- function(N, M, n, r) {

    sing.vals = (r:1) / r + 1 / r
    B = rortho(N)[,1:r] %*% diag(sing.vals) %*% rortho(M)[1:r,]
    X = matrix(rnorm(M * n), M, n)
    Y = B %*% X + matrix(rnorm(N * n), N, n)

    list(X=X, Y=Y)
}
```

The `simRR()` function uses a convenience function `rortho()` to generate a random orthonormal matrix.

```
rortho = function(n) {
  return(qr.Q(qr(array(runif(n), dim = c(n, n)))))
}
```

Following Drton and Plummer (2017, Section 5.1) we set the dimensions M, N for the simulated data and the true rank r as follows:

```
M = 15
N = 10
r = 5
```

Singular BIC for rank selection

The learning coefficients and multiplicities for reduced rank regression were derived by Aoyagi and Watanabe (2005). See Drton and Plummer (2017, Section 2.2) for a discussion.

The `ReducedRankRegressions()` function sets up a reduced rank regression model of given dimensions (`numResponses`, `numCovariates`), and with the rank of the coefficient matrix B up to a maximum value (`maxRank`). The return value is an object of class "ReducedRankRegressions" containing data on the learning coefficients and multiplicities.

```
set.seed(1234)
library(sBIC)
rreg = ReducedRankRegressions(numResponses=N, numCovariates=M, maxRank=min(M, N))
```

The output from the `ReducedRankRegressions()` function is combined with the data in the `sBIC()` function to produce the singular BIC as well as the Schwarz BIC. Here we use a simulated data set `XY` with 100 observations.

```
XY = simRR(N, M, 100, r)
results = sBIC(XY, rreg)
results
```

```
## $logLike
##  [1] -713.7313 -601.7517 -536.8049 -495.0693 -461.8093 -443.4735 -431.8450
##  [8] -427.5681 -424.5127 -422.0075 -421.6355
##
## $sBIC
##  [1] -768.9933 -707.6706 -684.4012 -672.6901 -664.9195 -664.2727 -668.8357
##  [8] -680.6701 -692.1950      -Inf      -Inf
##
## $BIC
##  [1] -768.9933 -707.6706 -688.7755 -688.4865 -692.0678 -705.9682 -721.9708
##  [8] -740.7196 -756.0850 -767.3953 -776.2336
##
## $modelPoset
## [1] "ReducedRankRegressions: 0x7fde166dbaa0"
```

The rank selected by choosing the maximum BIC or sBIC is given below. Note that since the smallest model has rank 0, we must subtract 1 from the index to get the rank of the optimal model.

```
which.max(results$BIC) - 1
```

```
## [1] 3
```

```
which.max(results$sBIC) -1
```

```
## [1] 5
```

Simulation study of rank selection

The asymptotic properties of BIC and sBIC can be further studied by simulating multiple data sets from the reduced rank regression model and comparing the results of rank selection (i.e. choosing the optimal model according to BIC or sBIC).

Here we set up the simulation parameters. We consider sample sizes ranging from 50 up to 1000 and simulate `sims=200` data sets for each sample size.

```
set.seed(1234)
n = c(50, 100, 200, 300, 500, 1000)
sims = 200
```

The code below carries out the rank selection and produces a list of frequency tables `tt` for the optimal model chosen by BIC and sBIC.

```
tt <- vector("list", length(n))
for (i in seq_along(n)) {

  rank.bic = rank.sbic = factor(rep(0, sims), levels = 0:min(M, N))

  for (j in 1:sims) {
    rreg = ReducedRankRegressions(N, M, min(M, N))
    XY = simRR(N, M, n[i], r)
    results = sBIC(XY, rreg)

    rank.bic[j] = which.max(results$BIC) - 1
    rank.sbic[j] = which.max(results$sBIC) - 1
  }

  tt[[i]] = rbind("BIC"=table(rank.bic),"sBIC"=table(rank.sbic))
}
```

The code below visualizes the frequency tables in `tt` as a series of bar plots. The output is shown in Figure 1.

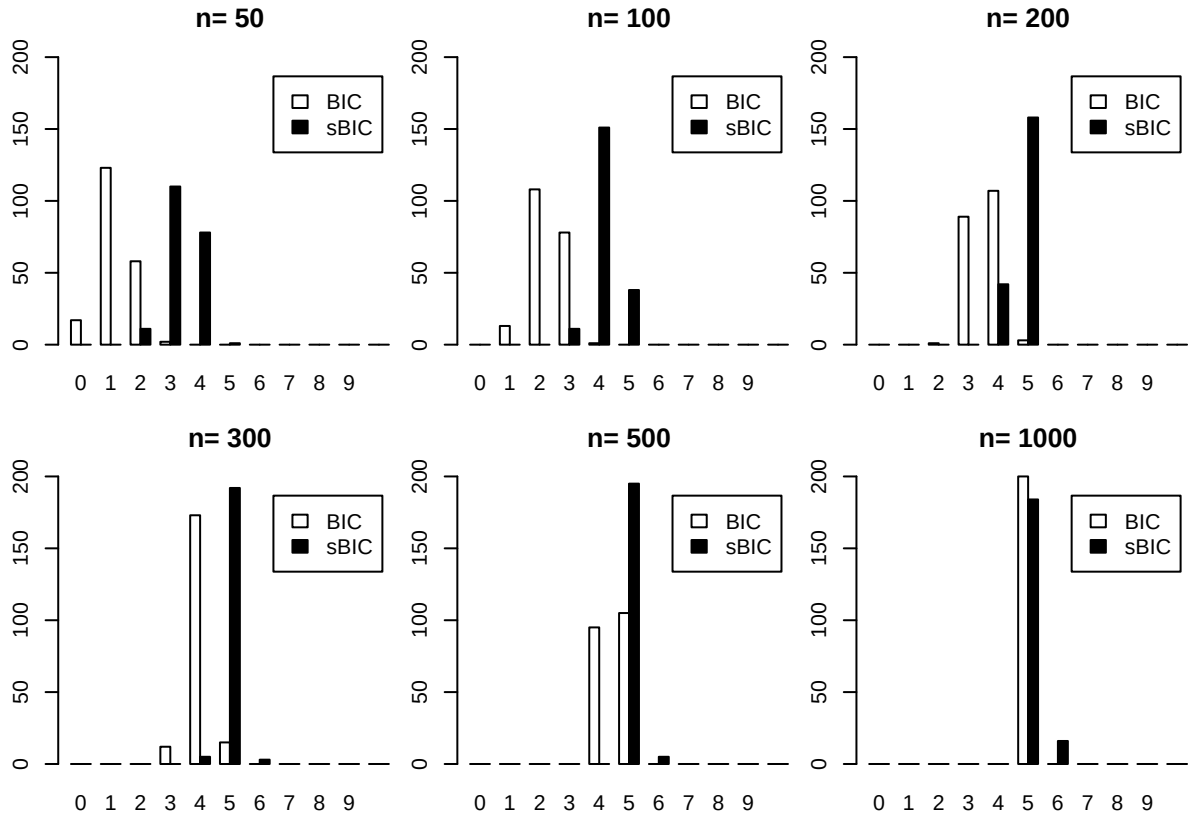

Figure 1: Frequencies of rank estimates in reduced rank regression for a model with true rank 5

Figure 1 partly reproduces Figure 2 of Drton and Plummer (2017). The latter also includes a comparison with the Widely Applicable Bayesian Information Criterion (WBIC) of Watanabe (2013).

Bibliography

- Aoyagi, M. and Watanabe, S. (2005) Stochastic complexities of reduced rank regression in Bayesian estimation. Neurl Netwrks; 18: 924-933.
- Drton M. and Plummer M. (2017), A Bayesian information criterion for singular models. J. R. Statist. Soc. B; 79: 1-38.
- Watanabe, S. (2013) A widely applicable Bayesian information criterion. J. Mach. Learn. Res.; 14: 867-897