Bayesian linear regression for unequal weighting

Stefan

September 23, 2016

Multiple linear regression

linear model: $X\beta = y$

first column of X is ones, the other columns of X are forecasts produced by different models, y is the observation

OLS: minimise $||X\beta - y||^2$

OLS estimator: $\hat{\beta} = (X'X)^{-1}X'y$

if all X are independent, $(X'X)^{-1}$ is diagonal, and the elements of $\hat{\beta}$ are equal to the **correlations** between the columns of X and y.

but in general the forecasts are correlated with one another, **multi-collinearity**

as a result, a predictor can be positively correlated with y, but receive a negative weight

also, the estimated parameters have **high variance**

related is this interesting point from wikipedia entry on **Tikhonov regularization**: $X\beta$ is a like low-pass filter, filtering out the variances within the elements of X, producing a weighted average; therefore, the solution to the inverse problem acts as a high-pass filter, amplifying noise

Example:

```
load("../../data/enso-nao.Rdata")
X = cbind(1, enso[, -1])
y = enso[, 1, drop = FALSE]
# nao[, -1] = nao[, -1] / 1000 X = cbind(1, nao[,-1]) y = nao[, 1,
# drop=FALSE]
```

```
# X = cbind( 1,
# rowMeans(read.table('../../data/CMC1-CanCM3-NAO-hind-ens-members-ic-Nov-val-DJF-1982-20
# rowMeans(read.table('../../data/CMC2-CanCM4-NAO-hind-ens-members-ic-Nov-val-DJF-1982-20
# rowMeans(read.table('../../data/ECMWFS4-NAO-hind-ens-members-ic-Nov-val-DJF-1982-2010.t
# #rowMeans(read.table('../../data/CFS2-NAO-hind-ens-members-ic-Nov-val-DJF-1982-2010.txt
# rowMeans(read.table('../../data/MFS3-NAO-hind-ens-members-ic-Nov-val-DJF-1982-2010.txt
# ) y =
```

```r
# as.matrix(read.table('../../data/ERAINT-NAO-DJF-1982-2010.txt'))/1000
# i.nna = !is.na(rowSums(cbind(X,y))) X = X[i.nna, ] y = y[i.nna, ,
# drop=FALSE]
```

```r
n = length(y)
```

Simple linear regression on the multi-model ensemble mean vs multiple linear regression

Simple linear regression on the multi-model ensemble mean

$y = (XM)\beta$ where M is the $((m+1) \times 2)$ matrix that transforms the row vector $(1, x_1, ..., x_m)$ into the row vector $(1, \sum x_i/m)$, i.e. calculates the multimodel ensemble mean

```r
m = ncol(X) - 1
M = rbind(c(1, 0), cbind(rep(0, m), rep(1/m, m)))
print(M)
```

```
##      [,1]   [,2]
## [1,]    1 0.0000
## [2,]    0 0.1667
## [3,]    0 0.1667
## [4,]    0 0.1667
## [5,]    0 0.1667
## [6,]    0 0.1667
## [7,]    0 0.1667
```

We calculate the simple OLS estimators for regression on the MMM:

```r
XM = X %*% M
beta_mmm = solve(crossprod(XM)) %*% crossprod(XM, y)
rss_mmm = sum((y - XM %*% beta_mmm)^2)
sigma2_mmm = rss_mmm/n
print(c(beta_mmm, sigma2_mmm))
```

```
## [1] 2.3245 0.9529 0.2416
```

Multiple linear regression on the multi-model ensemble

```
beta_ols = solve(crossprod(X)) %*% crossprod(X, y)
rss_ols = sum((y - X %*% beta_ols)^2)
sigma2_ols = rss_ols/n
print(c(beta_ols, sigma2_ols))
```

```
## [1] -4.0686 -0.1044  0.3341  0.1674  0.6636 -0.1630  0.2754  0.1911
```

Bayesian linear regression

prior distribution on parameters:

$(\sigma^2|a,b) \sim IG(a,b)$

$(\beta|\sigma^2, M) \sim N(\beta_0, \sigma^2 M^{-1})$

the marginal prior of beta is

$(\beta|a,b,\beta_0,M) \sim T(2a, \beta_0, b/aM^{-1})$

posterior predictive distribution:

$y^*|y,X,x^* \sim T(n+2a, \hat{\mu}, \hat{\Sigma})$

where

$\hat{\mu} = x^*(M+X'X)^{-1}(X'y+M\beta_0)$

and

$\hat{\Sigma} = (n+2a)^{-1}(2b+s^2+(\beta_0-\hat{\beta})'(M^{-1}+(X'X)^{-1})^{-1}(\beta_0-\hat{\beta}))(1+x^*(M+X'X)^{-1}x^{*'})$

- we use the idea of shrinking beta towards the equal-weighting solution, and demanding that no elements should be negative
- informative, data-driven prior
- set prior mean of beta equal to the equal weights solution, and prior variance such that no element of beta will become zero

$mode(\sigma^2) = b/(a+1)$ and $sd(\sigma^2) = b/(a-1)/\sqrt{a-2}$

- we set a and b of the IG distribution such that
- the prior standard deviation is equal to the prior mode
- the prior mode of σ^2 corresponds to the maximum-likelihood estimator of σ^2 in the equal-weighting linear regression
- so we get $\mathbf{a = 4.5}$ and $\mathbf{b = 5.5\hat{\sigma}^2_{ols}}$

```
a = 4.5
b = 5.5 * sigma2_ols
```

- we set the prior mean of β equal to the equal-weighting linear regression solution

- $\beta_0 = \hat{\beta}_{ols}$

```
beta0 = c(beta_mmm[1], rep(beta_mmm[2]/m, m))
```

- we set the prior variance such that a priori the elements of β are unlikely to be smaller than zero
- that is, we want twice the prior standard deviation to be equal to the prior means of *beta*
- under the T distribution, the prior marginal variance is $b/(a-1)M^{-1}$
- we have previously chosen $b/(a-1) = 5.5/3.5\hat{\sigma}^2_{ols}$
- so setting $M^{-1} = diag(3.5/5.5/\hat{\sigma}^2_{ols})$ sets the variance equal to one
- so setting $M^{-1} = diag(7/11/\hat{\sigma}^2_{ols})$ ……..