Collinearity in linear models

Elizabeth Colantuoni

2/21/2021

Linear dependencies in X

There was a question about what happens when it is not possible to compute $(X^{'}X)^{-1}$. In regression analysis, this issue most often arises when there are linear dependencies in X such that you can find at least one column of X that is equal to a linear combination of the other columns of X.

Take as an example, the design matrix generated by the following model:

$$arm_i = \beta_0 + \beta_1 female_i + \beta_2 male_i + \epsilon_i$$

In this case the design matrix is of dimension $n \times 3$ and there is a linear dependency in X where $male = intercept - female$. $X^{'}X$ is of dimension 3×3, but $X^{'}X$ has rank 2 not rank 3, which is what would be required to find the inverse.

So what will happen when you try to fit this model?

```
load("NepalAnthro.rdata")
d = nepal.anthro[nepal.anthro$num==1,]
d$female=d$sex-1
d$male = d$sex
summary(lm(arm~female+male,data=d,na.action="na.omit")  )
```

```
##
## Call:
## lm(formula = arm ~ female + male, data = d, na.action = "na.omit")
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8354 -0.7354  0.0465  0.7646  2.7646
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.73535    0.11967 114.775   <2e-16 ***
## female      -0.08187    0.17552  -0.466    0.641
## male             NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.191 on 183 degrees of freedom
##   (15 observations deleted due to missingness)
## Multiple R-squared:  0.001187,   Adjusted R-squared:  -0.004271
## F-statistic: 0.2175 on 1 and 183 DF,  p-value: 0.6415
```

Notice that in the output from the linear model fit that the covariate "male" has been dropped from the model and in the heading for the coefficients table you see "(1 not defined because of singularities)".

Near linear dependencies in X

Things become more complicated when there are near singularities or linear dependencies in the design matrix. In the example below, I generate two highly correlated covariates and show how the regression results may be unusual when there are near perfect dependencies.

NOTE: I am using the "mvrnorm" command from the MASS package to generate multivariate normal covariate for the regression model. To use this command you need to specify

- the number of observations you want to generate
- the mean of the multivariate normal distribution; this will be a vector
- the variance of the multivariate normal distribution; this will be a matrix with variances on the diagonal elements and covariances on the off-diagonal.

NOTE: In the simulation below, I generate X_1, X_2, and X_3 to each have mean 0 and variance 1. Therefore, the off-diagonal elements of the variance matrix are the correlations. I am setting $Corr(X_1, X_2) = 0.99, Corr(X_1, X_3) = 0.2, Corr(X_2, X_3) = 0.1$.

```
set.seed(5231)
x = mvrnorm(100,Sigma=matrix(c(1,0.98,0.2,0.98,1,0.1,0.2,0.1,1),nrow=3),mu=c(0,0,0))
cor(x)
```

```
##           [,1]      [,2]      [,3]
## [1,] 1.0000000 0.9811575 0.1831770
## [2,] 0.9811575 1.0000000 0.1041959
## [3,] 0.1831770 0.1041959 1.0000000
```

```
y = 1 + 0.3 * x[,1] + 0.5 * x[,2] + 0.5 * x[,3] + rnorm(100)
summary(lm(y~x[,1]+x[,2]+x[,3]))$coefficients
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 0.9920786  0.1018609 9.7395461 5.394517e-16
## x[, 1]      0.6378611  0.6703814 0.9514897 3.437453e-01
## x[, 2]      0.1764307  0.6415110 0.2750236 7.838891e-01
## x[, 3]      0.4636549  0.1266231 3.6616933 4.099798e-04
```

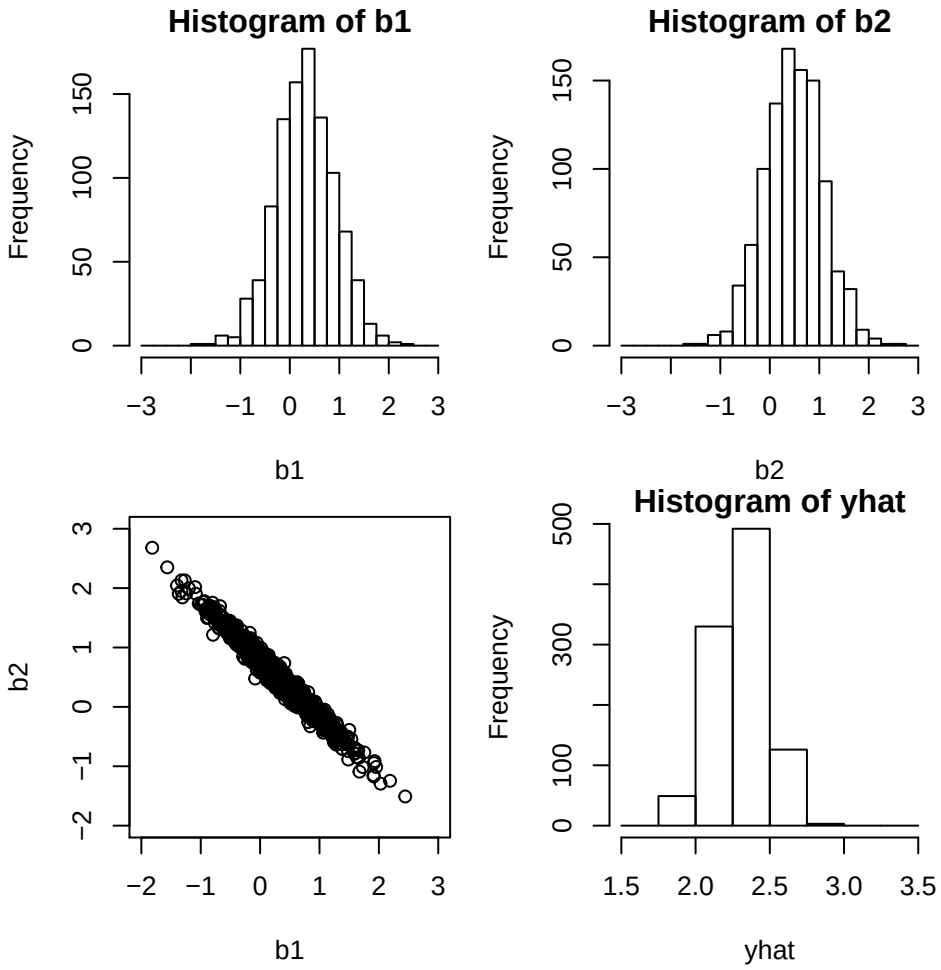
From the model output, we notice that $\hat{\beta}_0$ and $\hat{\beta}_3$ are roughly what we expect (1 and 0.5, respectively). However, $\hat{\beta}_1$ and $\hat{\beta}_2$ are different from what we expect and both have large standard error estimates. This could be attributable to sampling variation but in this case, there is more going on.

The code below generates $K = 1000$ samples and model fits from this scenario where $X1$ and $X2$ are highly correlated. From each of the K linear regression models, we will save:

1. $\hat{\beta}_1$ and $\hat{\beta}_2$
2. The predicted value of Y when $X_1 = X_2 = X_3 = 1$

This will allow us to understand the behavior of the estimated regression coefficients but also the predicted values generated from the fit of the regression model.

```
# Replicate the analysis and look at the distribution of the
# coefficients for x[,1] and x[,2]
K = 1000
set.seed(22121)
b1 = b2 = yhat = NULL
for(i in 1:K){
  x = mvrnorm(100,Sigma=matrix(c(1,0.98,0.2,0.98,1,0.1,0.2,0.1,1),nrow=3),mu=c(0,0,0))
  y = 1 + 0.3 * x[,1] + 0.5 * x[,2] + 0.5 * x[,3] + rnorm(100)
  coef = lm(y~x[,1]+x[,2]+x[,3])$coeff
  b1 = c(b1,coef[2])
  b2 = c(b2,coef[3])
  yhat = c(yhat,sum(coef))
  }
par(mfrow=c(2,2),mar=c(4,4,1,1))
hist(b1,breaks=seq(-3,3,0.25))
hist(b2,breaks=seq(-3,3,0.25))
plot(b1,b2,xlim=c(-2,3),ylim=c(-2,3))
hist(yhat,breaks=seq(1.5,3.5,0.25))
```




```
summary(b1);sqrt(var(b1))
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -1.81747 -0.06877   0.31926   0.31880   0.72347   2.44363
```

```
## [1] 0.5987389
```

```
summary(b2);sqrt(var(b2))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.5088  0.0742  0.4849  0.4822  0.8487  2.6797
```

```
## [1] 0.5891181
```

```
cor(b1,b2)
```

```
## [1] -0.9859484
```

```
summary(yhat);sqrt(var(yhat))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.768   2.183   2.300   2.302   2.426   2.923
```

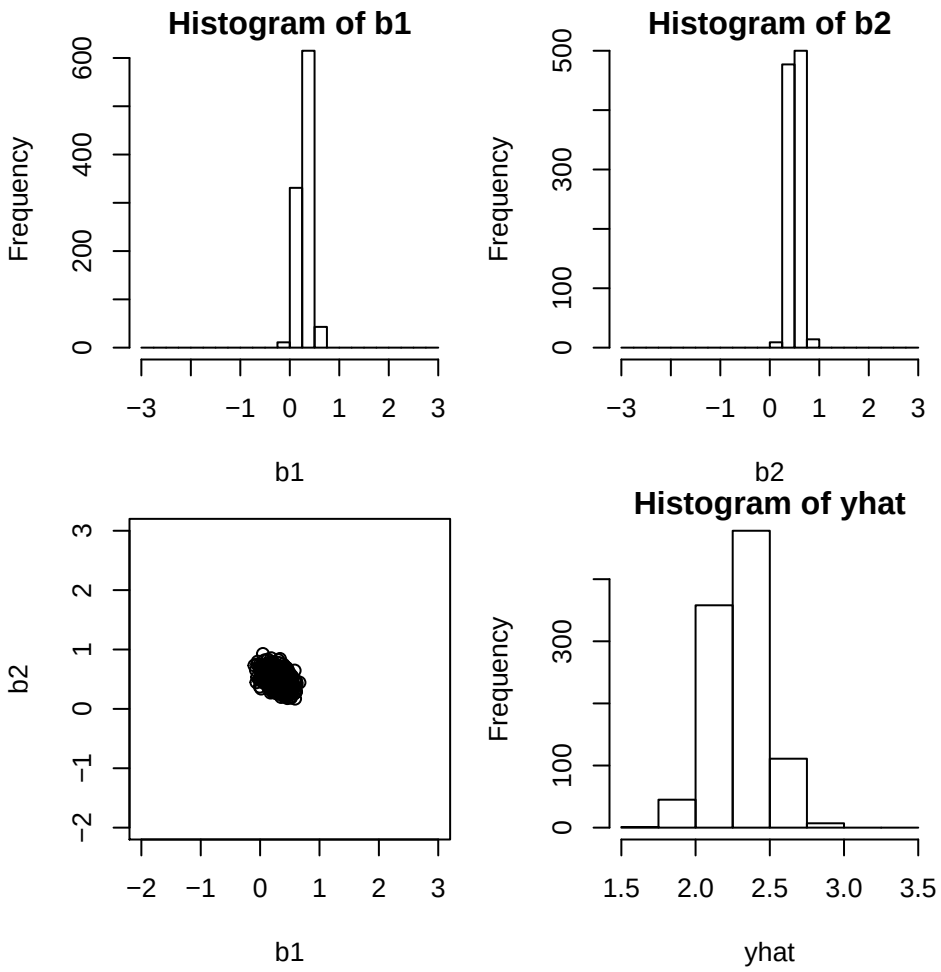
```
## [1] 0.1774432
```

We notice a few key features of the output:

- On average, $\hat{\beta}_1$ and $\hat{\beta}_2$ are unbiased with means 0.32 and 0.48.
- There is high variation in both $\hat{\beta}_1$ and $\hat{\beta}_2$; the standard deviation is roughly 0.6 for both estimates
- $\hat{\beta}_1$ and $\hat{\beta}_2$ are strongly negatively correlated. In a given sample, when $\hat{\beta}_1$ is estimated to be greater than the value of β_1 then we see that the corresponding estimate of β_2 is less than what we expect.
- The predicted value when $X_1 = X_2 = X_3 = 1$ is on average 2.3 with a standard deviation of 0.18.

Now compare the same results when we reduce the correlation between the first and second covariate in the design matrix; $Corr(X_1, X_2) = 0.4$.

```
# Replicate the analysis and look at the distribution of the
# coefficients for x[,1] and x[,2]
K = 1000
set.seed(7621)
b1 = b2 = yhat = NULL
for(i in 1:K){
  x = mvrnorm(100,Sigma=matrix(c(1,0.4,0.2,0.4,1,0.1,0.2,0.1,1),nrow=3),mu=c(0,0,0))
  y = 1 + 0.3 * x[,1] + 0.5 * x[,2] + 0.5 * x[,3] + rnorm(100)
  coef = lm(y~x[,1]+x[,2]+x[,3])$coeff
  b1 = c(b1,coef[2])
  b2 = c(b2,coef[3])
  yhat = c(yhat,sum(coef))
  }
par(mfrow=c(2,2),mar=c(4,4,1,1))
hist(b1,breaks=seq(-3,3,0.25))
hist(b2,breaks=seq(-3,3,0.25))
plot(b1,b2,xlim=c(-2,3),ylim=c(-2,3))
hist(yhat,breaks=seq(1.5,3.5,0.25))
```




```
summary(b1);sqrt(var(b1))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.0958  0.2160  0.2980  0.2942  0.3688  0.6625
```

```
## [1] 0.1179907
```

```
summary(b2);sqrt(var(b2))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1721  0.4312  0.5031  0.5064  0.5779  0.9249
```

```
## [1] 0.1102162
```

```
cor(b1,b2)
```

```
## [1] -0.3750474
```

```
summary(yhat);sqrt(var(yhat))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.672   2.174   2.292   2.292   2.403   2.827
```

```
## [1] 0.1781559
```

When comparing the simulation when $Corr(X_1, X_2) = 0.99$ and $Corr(X_1, X_2) = 0.4$, we note the following:

- In both simulations, $\hat{\beta}_1$ and $\hat{\beta}_2$ have mean roughly equal to the true values of 0.3 and 0.5, respectively. *Unbiased estimates of coefficients*
- When $Corr(X_1, X_2) = 0.4$, the variance of $\hat{\beta}_1$ and $\hat{\beta}_2$ are much smaller than the variances when $Corr(X_1, X_2) = 0.99$. *Significantly higher standard errors for the estimated coefficients when the covariates are highly correlated*
- When $Corr(X_1, X_2) = 0.4$, $\hat{\beta}_1$ and $\hat{\beta}_2$ are negatively correlated but less correlated compared to when $Corr(X_1, X_2) = 0.99$ *The regression coefficieints are related to one another but not as strongly; so that we will not tend to see extreme swings or differences in $\hat{\beta}_1$ and $\hat{\beta}_2$.
- The distribution of the predicted value when $X_1 = X_2 = X_3 = 1$ is roughly the same regardless of $Corr(X_1, X_2)$.

A couple of final take home messages about collinearity in columns of X:

1. The coefficients for columns of X that are highly correlated can cause unusual estimates for regression coefficients in a given sample.
2. Prediction is robust to highly correlated columns of X.