Assignment 3 STP 526, Fall 2009, S. Lohr

Due on Monday, 21 September 2007. Be sure to justify all your answers.

1. Read Chapter 3 (pages 73-86) of the text.

2. For the system of equations $\mathbf{Ax} = \mathbf{b}$ given below, find a generalized inverse of $\mathbf{A}$. Using $\mathbf{A}^-$, show that the system of equations is consistent, and use $\mathbf{A}^-$ to find the set of solutions. Use R for this problem.

$$\begin{aligned} 3x_1 - 5x_2 + 6x_3 + x_4 &= 7 \\ 4x_1 + 2x_3 - 3x_4 &= 5 \\ x_2 - 3x_3 + 7x_4 &= 0 \end{aligned}$$

3. Show that $\mathbf{A}^+\mathbf{A}$ is the projection matrix onto $\mathcal{C}(\mathbf{A}')$.

4. To protect confidentiality of data, researchers would like to release a set of transformed observations such that the first and second moments of the transformed data set match those of the original data set. Here is one method for doing this that has been proposed. Let $\mathbf{V}$ be an $n \times k$ matrix ($n \geq k + p$) such that $\mathbf{V}'\mathbf{1}_n = 0$ and $\mathbf{V}'\mathbf{V} = \mathbf{I}_k$. Let $\mathbf{P_V}$ be the projection matrix onto $\mathcal{C}(\mathbf{V})$, and let $\mathbf{T} = \mathbf{I} - m\mathbf{P_V}$ for some constant $m \neq 0$.

   (a) Is $\mathbf{P_V}$ nonnegative definite?

   (b) Find m so that $\mathbf{T}$ will be orthogonal.

   (c) Is $\mathbf{T}$ idempotent for the value of m you found?

   (d) Let $\mathbf{X}$ be an $n \times p$ matrix, and let $\mathbf{Y} = \mathbf{TX}$, using the value of m in part(b). Show that $\mathbf{X}'\mathbf{1}_n = \mathbf{Y}'\mathbf{1}_n$ and that $\mathbf{X}'\mathbf{X} = \mathbf{Y}'\mathbf{Y}$.

   (e) For $n = 10$, use R to randomly generate an n-vector $\mathbf{u}$ and an $n \times 3$ matrix X with independent $N(0, 1)$ entries (use rnorm). Now find $\mathbf{v} = (\mathbf{I} - \mathbf{P_1})\mathbf{u}$. Calculate $\mathbf{Y} = \mathbf{TX}$. Print out $\mathbf{X}$ and $\mathbf{Y}$, and show that the means (use the R command *apply*) and covariances (R command *cov*) of $\mathbf{X}$ and $\mathbf{Y}$ are the same. Be sure to include your R code for this problem.

5. Do problems 3.5, 3.6 and 3.10 in the book.

Suggested additional problems (DO NOT HAND IN): 3.1, 3.2, 3.3, 3.4, 3.16

```
R will calculate the singular value decomposition and generalized
inverses. To use function ginv, you need to load package MASS into
R (use ``Load Package'' from the Packages menu).

> bmat
     [,1] [,2] [,3]
```

```
[1,]    1   -1    4
[2,]    1    2    2
> svdb <- svd(bmat) #Singular value decomposition
> svdb  # Note v is 3 x 2.
$d
[1] 4.671366 2.275597

$u
           [,1]       [,2]
[1,] -0.8777123 -0.4791881
[2,] -0.4791881  0.8777123

$v
            [,1]       [,2]
[1,] -0.29047186  0.1751295
[2,] -0.01726773  0.9819896
[3,] -0.95672772 -0.0708947

> t(svdb$v) %*% svdb$v
              [,1]          [,2]
[1,]  1.000000e+00 -4.855193e-17
[2,] -4.855193e-17  1.000000e+00
> (svdb$v) %*% diag(1/svdb$d) %*% t(svdb$u)
            [,1]       [,2]
[1,]  0.01769912 0.09734513
[2,] -0.20353982 0.38053097
[3,]  0.19469027 0.07079646

# The above is intended so you see how a g-inverse is calculated.
# In practice, all you need is the ginv command.

> gbmat <- ginv(bmat)
> gbmat
            [,1]       [,2]
[1,]  0.01769912 0.09734513
[2,] -0.20353982 0.38053097
[3,]  0.19469027 0.07079646
> bmat %*% gbmat %*% bmat  # Check that this is a g-inverse
     [,1] [,2] [,3]
[1,]    1   -1    4
[2,]    1    2    2
> gbmat %*% bmat %*% gbmat  # It's also reflexive!
```

```
            [,1]       [,2]
[1,]  0.01769912 0.09734513
[2,] -0.20353982 0.38053097
[3,]  0.19469027 0.07079646

> svd(t(bmat)) # What is SVD(B')?
$d
[1] 4.671366 2.275597

$u
            [,1]       [,2]
[1,] -0.29047186 -0.1751295
[2,] -0.01726773 -0.9819896
[3,] -0.95672772  0.0708947

$v
           [,1]       [,2]
[1,] -0.8777123  0.4791881
[2,] -0.4791881 -0.8777123


# Note that ginv is more careful about numerical errors
# than using svd directly: matrix from Example 3.1.1, p. 74

> amat <- cbind(c(4,1,3),c(1,1,1),c(2,5,3),c(0,15,5))
> amat
     [,1] [,2] [,3] [,4]
[1,]    4    1    2    0
[2,]    1    1    5   15
[3,]    3    1    3    5
> svda <- svd(amat)
> svda  # Note third singular value is essentially 0
$d
[1] 1.702358e+01 5.215157e+00 1.490920e-15

$u
            [,1]       [,2]       [,3]
[1,] -0.07739392  0.8416032  0.5345225
[2,] -0.92921843 -0.2551951  0.2672612
[3,] -0.36133542  0.4760038 -0.8017837

$v
```

```
            [,1]       [,2]        [,3]
[1,] -0.13644609  0.8703915 -0.22992621
[2,] -0.08035608  0.2037162  0.97200266
[3,] -0.34569037  0.3519054 -0.02614890
[4,] -0.92489109 -0.2776345 -0.04075546

> ginv(amat)
            [,1]         [,2]         [,3]
[1,]  0.14108094 -0.035143365  0.082339508
[2,]  0.03324029 -0.005582340  0.020299416
[3,]  0.05836082  0.001649328  0.039456991
[4,] -0.04059883  0.064070033 -0.005709211
> ginv
function (X, tol = sqrt(.Machine$double.eps))
{
    if (length(dim(X)) > 2 || !(is.numeric(X) || is.complex(X)))
        stop("X must be a numeric or complex matrix")
    if (!is.matrix(X))
        X <- as.matrix(X)
    Xsvd <- svd(X)
    if (is.complex(X))
        Xsvd$u <- Conj(Xsvd$u)
    Positive <- Xsvd$d > max(tol * Xsvd$d[1], 0)
    if (all(Positive))
        Xsvd$v %*% (1/Xsvd$d * t(Xsvd$u))
    else if (!any(Positive))
        array(0, dim(X)[2:1])
    else Xsvd$v[, Positive, drop = FALSE] %*% ((1/Xsvd$d[Positive]) *
        t(Xsvd$u[, Positive, drop = FALSE]))
}
# For amat, Positive = c(T,T,F)
> Xsvd$v[, Positive, drop = FALSE]
            [,1]       [,2]
[1,] -0.13644609  0.8703915
[2,] -0.08035608  0.2037162
[3,] -0.34569037  0.3519054
[4,] -0.92489109 -0.2776345
> t(Xsvd$u[, Positive, drop = FALSE])
            [,1]       [,2]       [,3]
[1,] -0.07739392 -0.9292184 -0.3613354
[2,]  0.84160317 -0.2551951  0.4760038
```