MAST30025: Linear Statistical Models

Assignment 3 Solutions

Total marks: 45

1. Consider the matrix $A(A^TA)^cA^T$.

   (a) Show that this matrix is unique (invariant to the choice of conditional inverse).
   Solution [3 marks]:

$$\begin{aligned} A(A^TA)^c_1A^T &= A(A^TA)^c_2A^TA(A^TA)^c_1A^T \\ &= A(A^TA)^c_2\left[A(A^TA)^c_1A^TA\right]^T \\ &= A(A^TA)^c_2A^T. \end{aligned}$$

   (b) Show that the rank of this matrix is $r(A)$.
   Solution [3 marks]: We have $r(A(A^TA)^cA^T) \leq r(A)$, but also

$$r(A) = r(A(A^TA)^cA^TA) \leq r(A(A^TA)^cA^T).$$

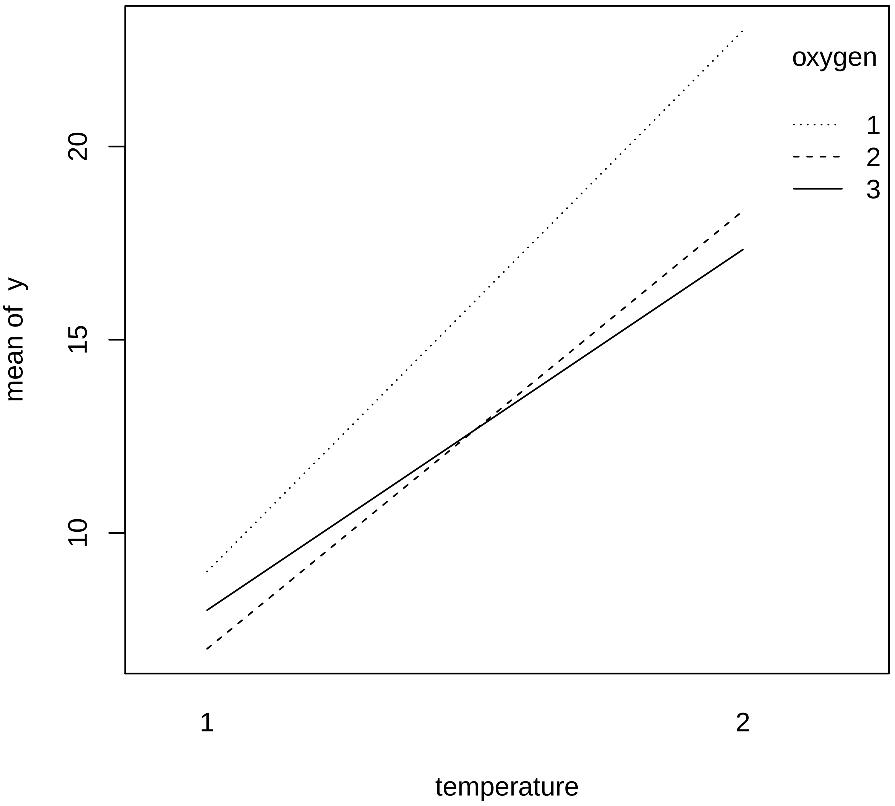
2. We study the amount of rotting of a potato exposed to a variety of levels of oxygen, and a variety of temperatures. A small experiment is conducted and the following data obtained:

	Oxygen level		
Temperature	1	2	3
10	13	10	15
	11	4	2
	3	7	7
16	26	15	20
	19	22	24
	24	18	8

For this question, you may not use the `lm` or `ginv` functions in R.

   (a) Plot an interaction plot of the data. Does there appear to be interaction?
   Solution [3 marks]:

```
> n <- 18
> r <- 4
> y <- c(13,10,15,11,4,2,3,7,7,26,15,20,19,22,24,24,18,8)
> oxygen <- rep(1:3,6)
> temperature <- rep(1:2,each=9)
> interaction.plot(temperature, oxygen, y)
```



There does not appear to be much interaction.

(b) Fit an additive model, outputting your design matrix. Estimate the common variance.

Solution [3 marks]:

```
> X <- matrix(0,n,6)
> X[,1] <- 1
> X[cbind(1:n,oxygen+1)] <- 1
> X[cbind(1:n,temperature+4)] <- 1
> X
```

```
      [,1] [,2] [,3] [,4] [,5] [,6]
 [1,]    1    1    0    0    1    0
 [2,]    1    0    1    0    1    0
 [3,]    1    0    0    1    1    0
 [4,]    1    1    0    0    1    0
 [5,]    1    0    1    0    1    0
 [6,]    1    0    0    1    1    0
 [7,]    1    1    0    0    1    0
 [8,]    1    0    1    0    1    0
 [9,]    1    0    0    1    1    0
[10,]    1    1    0    0    0    1
[11,]    1    0    1    0    0    1
[12,]    1    0    0    1    0    1
[13,]    1    1    0    0    0    1
[14,]    1    0    1    0    0    1
[15,]    1    0    0    1    0    1
[16,]    1    1    0    0    0    1
```

```
[17,]    1    0    1    0    0    1
[18,]    1    0    0    1    0    1
> XtXc <- matrix(0,6,6)
> XtXc[2:5,2:5] <- solve((t(X)%*%X)[2:5,2:5])
> b <- XtXc%*%t(X)%*%y
> (s2 <- sum((y-X%*%b)^2)/(n-r))
[1] 26.12698
```

(c) Calculate a 95% confidence interval for the difference between the temperature effects.

Solution [2 marks]:

```
> tt <- c(0,0,0,0,1,-1)
> tt%*%b + c(-1,1)*qt(0.975,n-r)*sqrt(s2*t(tt)%*%XtXc%*%tt)
[1] -16.723555  -6.387556
```

(d) Test the hypothesis that oxygen level has no effect on rotting.

Solution [2 marks]:

```
> C <- matrix(c(0,1,-1,0,0,0,0,0,1,-1,0,0),2,6,byrow=T)
> Fstat <- (t(C%*%b) %*% solve(C %*% XtXc %*% t(C)) %*% C%*%b)/2/s2
> pf(Fstat, 2, n-r, lower=F)
          [,1]
[1,] 0.4481124
```

We cannot reject this hypothesis.

(e) Suppose we are interested in the effect of oxygen level only, but know that temperature affects the results, so we include it in our model. What type of design would this study be?

Solution [1 mark]: It would be a complete block design.

3. Consider the two-factor model with interaction

$$y_{ij} = \mu + \tau_i + \beta_j + \xi_{ij}.$$

Suppose that there are a and b levels of the factors respectively. Now consider the set of equations

$$\xi_{ij} - \xi_{1j} - \xi_{i1} + \xi_{11} = 0, \quad i = 2, \dots, a, \ j = 2, \dots, b.$$

(a) Show that the equations are not redundant.

Solution [2 marks]: Each equation contains a term ξ_{ij} which does not appear in any other equation, so they are not redundant.

(b) Show that these equations are equivalent to the hypothesis of no interaction.

Solution [3 marks]: Clearly the equations are a subset of the equations which test for interaction. We need to show that all the other equations can be expressed in terms of them. Choose $i \neq i', j \neq j'$, then

$$\begin{aligned} \xi_{ij} - \xi_{i'j} - \xi_{ij'} + \xi_{i'j'} &= (\xi_{ij} - \xi_{1j} - \xi_{i1} + \xi_{11}) - (\xi_{i'j} - \xi_{1j} - \xi_{i'1} + \xi_{11}) \\ &\quad - (\xi_{ij'} - \xi_{1j'} - \xi_{i1} + \xi_{11}) + (\xi_{i'j'} - \xi_{1j'} - \xi_{i'1} + \xi_{11}). \end{aligned}$$

Thus all other equations can be written in terms of these equations.

(c) Thereby calculate the rank of the hypothesis of no interaction.

Solution [1 mark]: There are $(a-1)(b-1)$ non-redundant equations, so that is the rank of the hypothesis.

(d) Show that the hypothesis is testable, provided there exists at least one sample from each combination of factor levels.

Solution [3 marks]: We can write

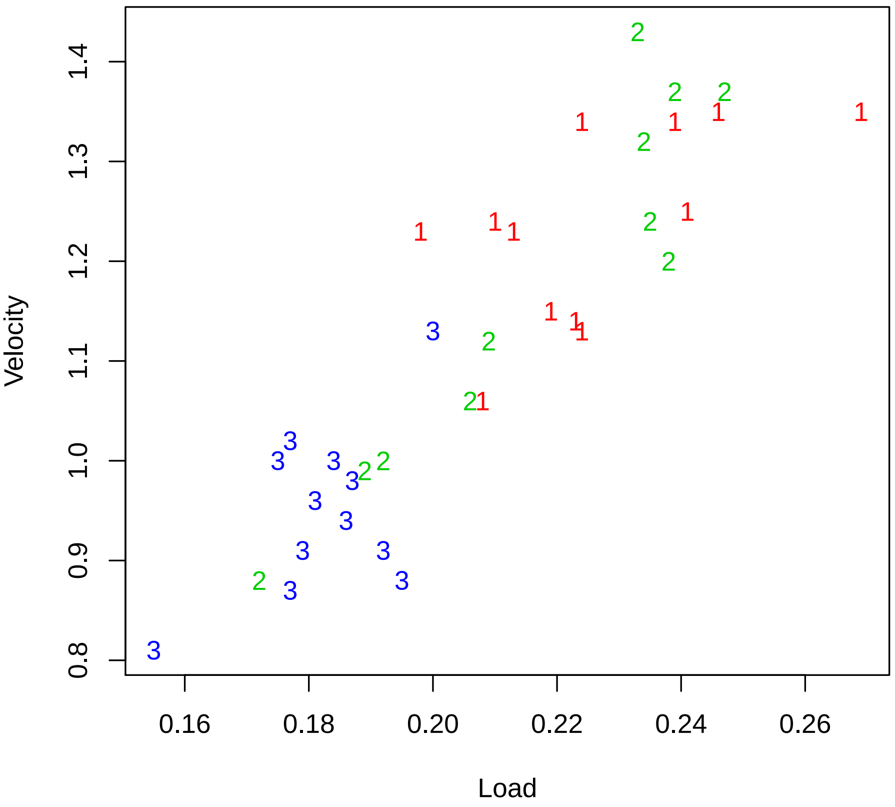
$$\begin{aligned} \xi_{ij} - \xi_{1j} - \xi_{i1} + \xi_{11} &= (\mu + \tau_i + \beta_j + \xi_{ij}) - (\mu + \tau_1 + \beta_j + \xi_{1j}) \\ &\quad - (\mu + \tau_i + \beta_1 + \xi_{i1}) + (\mu + \tau_1 + \beta_1 + \xi_{11}). \end{aligned}$$

This is a linear combination of elements of $X\boldsymbol{\beta}$, since there exists at least one sample from each combination of factor levels. Therefore hypotheses involving it are testable.

4. Maple trees have winged seeds called samara. An experiment is conducted to investigate the effect of shape on the speed of descent. Samara were collected from three trees, and their "disk loading" (a quantity based on size and weight, which was used to quantify shape) and descent velocity are calculated. The data is given in the file `heli.csv`, available on the LMS.

   (a) Plot the data, using different colours and/or symbols for each tree. What do you observe?
   Solution [3 marks]:

```
> heli <- read.csv('../data/heli.csv')
> plot(Velocity ~ Load, pch=as.character(Tree), col=Tree+1, data=heli)
```



   There appears to be a clear linear relationship between velocity and disk loading. It's not clear if the tree has an effect, either directly or through interaction.

   (b) Test for the presence of interaction between disk loading and tree.
   Solution [2 marks]:

```
> heli$Tree <- factor(heli$Tree)
> amodel <- lm(Velocity ~ Load + Tree, data=heli)
> imodel <- lm(Velocity ~ Load * Tree, data=heli)
> anova(amodel, imodel)

Analysis of Variance Table

Model 1: Velocity ~ Load + Tree
Model 2: Velocity ~ Load * Tree
  Res.Df     RSS Df Sum of Sq     F  Pr(>F)
1     31 0.20344
2     29 0.16549  2  0.037949 3.325 0.05011 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interaction is almost significant, but not at a 5% level.

(c) Use backward elimination from the model with interaction to select variables for the data.

Solution [2 marks]:

```
> drop1(imodel, scope=~., test='F')
```

```
Single term deletions

Model:
Velocity ~ Load * Tree
          Df Sum of Sq     RSS     AIC F value  Pr(>F)
<none>                 0.16549 -175.40
Load       1  0.039792 0.20528 -169.85  6.9729 0.01319 *
Tree       2  0.037553 0.20305 -172.24  3.2903 0.05154 .
Load:Tree  2  0.037949 0.20344 -172.17  3.3250 0.05011 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> model2 <- lm(Velocity ~ Load + Tree, data=heli)
> drop1(model2, scope=~., test='F')
```

```
Single term deletions

Model:
Velocity ~ Load + Tree
       Df Sum of Sq     RSS     AIC F value    Pr(>F)
<none>              0.20344 -172.17
Load    1  0.315542 0.51898 -141.39 48.0817 8.884e-08 ***
Tree    2  0.011322 0.21476 -174.28  0.8626    0.4319
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> model3 <- lm(Velocity ~ Load, data=heli)
> drop1(model3, scope=~., test='F')
```

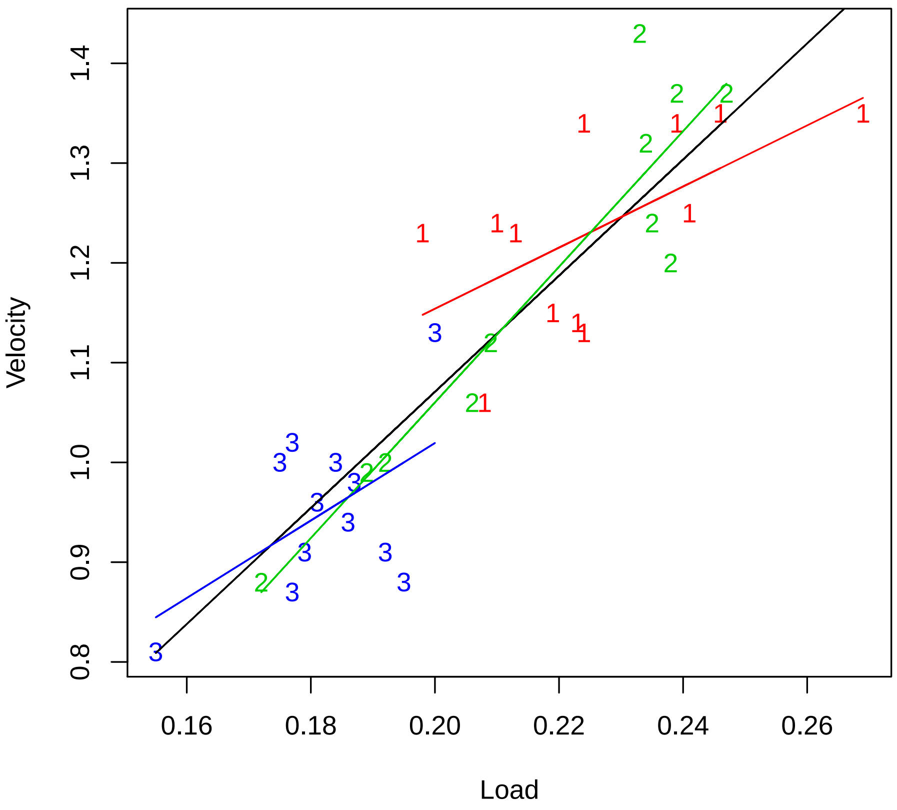
```
Single term deletions

Model:
Velocity ~ Load
       Df Sum of Sq     RSS     AIC F value    Pr(>F)
<none>              0.21476 -174.28
Load    1   0.84364 1.05840 -120.45  129.63 5.704e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The final model depends on `Load` only. Note that we should not drop the `Tree` variable before the interaction term is dropped.

(d) Add lines corresponding to model from part (c), and the full model with interaction, to the plot from question (a).

Solution [2 marks]:

```
> heli <- read.csv('../data/heli.csv')
> plot(Velocity ~ Load, pch=as.character(Tree), col=Tree+1, data=heli)
> with(heli, lines(Load, fitted(model3)))
> for (i in 1:3) { with(heli, lines(Load[Tree==i], fitted(imodel)[Tree==i], col=i+1)) }
```



(e) In the full model with interaction, test the hypothesis that a samara from tree 2 with a disk loading of 0.2 has an average descent velocity of 1.

Solution [2 marks]:

```
> library(car)
> linearHypothesis(imodel, c(1,0.2,1,0,0.2,0), 1)
```

```
Linear hypothesis test

Hypothesis:
(Intercept)  + 0.2 Load  + Tree2  + 0.2 Load:Tree2 = 1

Model 1: restricted model
Model 2: Velocity ˜ Load * Tree

  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1     30 0.19127
2     29 0.16549  1  0.025781 4.5177 0.04219 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We reject the hypothesis at a 5% level.

5. An apple orchard has 32 trees set aside for an experiment which aims to examine the effect of mulching on tree growth. There are four mulching treatments: 1. Control (no mulch); 2. Wood chips; 3. Garden compost; 4. Clippings from a local council collection. The trees are in a 4 × 8 rectangle, labeled as shown in the diagram below. The experimenter has the resources to maintain 16 plots, each consisting of 2 adjacent trees. All trees in the same plot must have the same treatment.

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32

In this question, if you need randomisation, use R and reproduce your R commands and output.

(a) Construct an appropriate experimental design. Draw the allocated treatments and write down the matrices in the corresponding linear model.

Solution [4 marks]: We use a completely randomised design. The plots can be oriented horizontally or vertically — we choose horizontally. Here is one possible allocation:

```
> (alloc <- sample(16,16)) #plot allocation
```

```
 [1]  2 13  6 10  3 12  1  8  4 14  5  7 16  9 15 11
```

```
> (talloc <- as.vector(rbind(2*alloc-1,2*alloc)))
```

```
 [1]  3  4 25 26 11 12 19 20  5  6 23 24  1  2 15 16  7  8 27 28  9 10 13 14 31
[26] 32 17 18 29 30 21 22
```

```
> talloc2 <- c()
> for (i in 1:4) { talloc2[talloc[(8*i-7):(8*i)]] <- i }
> matrix(talloc2, 4,8, byrow=T) #tree allocation
```

```
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,]    2    2    1    1    2    2    3    3
[2,]    3    3    1    1    3    3    2    2
[3,]    4    4    1    1    4    4    2    2
[4,]    1    1    3    3    4    4    4    4
```

The matrices are $\mathbf{y} = (y_1, y_2, \ldots, y_{32})^T, \boldsymbol{\beta} = (\mu, \tau_1, \tau_2, \tau_3, \tau_4)^T$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{32})^T$, and:

```
> X <- matrix(0,32,5)
> X[,1] <- 1
> X[cbind(1:32,talloc2+1)] <- 1
> X
```

```
      [,1] [,2] [,3] [,4] [,5]
 [1,]    1    0    1    0    0
 [2,]    1    0    1    0    0
 [3,]    1    1    0    0    0
 [4,]    1    1    0    0    0
 [5,]    1    0    1    0    0
 [6,]    1    0    1    0    0
 [7,]    1    0    0    1    0
 [8,]    1    0    0    1    0
 [9,]    1    0    0    1    0
[10,]    1    0    0    1    0
[11,]    1    1    0    0    0
[12,]    1    1    0    0    0
[13,]    1    0    0    1    0
[14,]    1    0    0    1    0
[15,]    1    0    1    0    0
[16,]    1    0    1    0    0
[17,]    1    0    0    0    1
[18,]    1    0    0    0    1
[19,]    1    1    0    0    0
[20,]    1    1    0    0    0
[21,]    1    0    0    0    1
[22,]    1    0    0    0    1
[23,]    1    0    1    0    0
[24,]    1    0    1    0    0
[25,]    1    1    0    0    0
[26,]    1    1    0    0    0
[27,]    1    0    0    1    0
[28,]    1    0    0    1    0
[29,]    1    0    0    0    1
[30,]    1    0    0    0    1
[31,]    1    0    0    0    1
[32,]    1    0    0    0    1
```

(b) Now suppose the ground slopes down from the left to the right of the diagram. Repeat question (a).

Solution [4 marks]: Here we must block according to the height of the plots (position on the slope). The best way to do this is to have the plots oriented horizontally. Here is one possible allocation:

```
> alloc <- c()
> for (i in 1:4) { alloc <- c(alloc, sample(4,4)) }
> (malloc <- matrix(alloc,4,4)) #plot allocation
```

```
     [,1] [,2] [,3] [,4]
[1,]    2    3    3    2
[2,]    4    4    2    3
[3,]    3    2    1    1
[4,]    1    1    4    4
```

```
> talloc <- rep(as.vector(t(malloc)),each=2)
> matrix(talloc, 4,8, byrow=T) #tree allocation
```

```
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,]    2    2    3    3    3    3    2    2
[2,]    4    4    4    4    2    2    3    3
[3,]    3    3    2    2    1    1    1    1
[4,]    1    1    1    1    4    4    4    4
```

$\mathbf{y}$ and $\boldsymbol{\varepsilon}$ are the same as before, and we have $\boldsymbol{\beta} = (\mu, \beta_1, \beta_2, \beta_3, \beta_4, \tau_1, \tau_2, \tau_3, \tau_4)^T$, and:

```
> X <- matrix(0,32,9)
> X[,1] <- 1
> X[cbind(1:32,rep(1:4,4,each=2)+1)] <- 1
> X[cbind(1:32,talloc+5)] <- 1
> X
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
 [1,]    1    1    0    0    0    0    1    0    0
 [2,]    1    1    0    0    0    0    1    0    0
 [3,]    1    0    1    0    0    0    0    1    0
 [4,]    1    0    1    0    0    0    0    1    0
 [5,]    1    0    0    1    0    0    0    1    0
 [6,]    1    0    0    1    0    0    0    1    0
 [7,]    1    0    0    0    1    0    1    0    0
 [8,]    1    0    0    0    1    0    1    0    0
 [9,]    1    1    0    0    0    0    0    0    1
[10,]    1    1    0    0    0    0    0    0    1
[11,]    1    0    1    0    0    0    0    0    1
[12,]    1    0    1    0    0    0    0    0    1
[13,]    1    0    0    1    0    0    1    0    0
[14,]    1    0    0    1    0    0    1    0    0
[15,]    1    0    0    0    1    0    0    1    0
[16,]    1    0    0    0    1    0    0    1    0
[17,]    1    1    0    0    0    0    0    1    0
[18,]    1    1    0    0    0    0    0    1    0
[19,]    1    0    1    0    0    0    1    0    0
[20,]    1    0    1    0    0    0    1    0    0
[21,]    1    0    0    1    0    1    0    0    0
[22,]    1    0    0    1    0    1    0    0    0
[23,]    1    0    0    0    1    1    0    0    0
[24,]    1    0    0    0    1    1    0    0    0
[25,]    1    1    0    0    0    1    0    0    0
[26,]    1    1    0    0    0    1    0    0    0
[27,]    1    0    1    0    0    1    0    0    0
[28,]    1    0    1    0    0    1    0    0    0
[29,]    1    0    0    1    0    0    0    0    1
[30,]    1    0    0    1    0    0    0    0    1
[31,]    1    0    0    0    1    0    0    0    1
[32,]    1    0    0    0    1    0    0    0    1
```

An alternate formulation would be to have 8 block effects.