Stat 511 Final Exam

May 4, 2009
Prof. Vardeman

(**This exam will scored on a 160 point basis.**)

I have neither given nor received unauthorized assistance on this exam.

____
Name

____
Name Printed

1. A marketing study used as an example in Neter et al. concerned counts of customers visiting a particular lumber store during a two-week period from each of $n = 110$ different census tracts (these are metropolitan areas with populations of about 4000 residents each). Various demographic characteristics of the tracts were also obtained. Available for each tract are

y = the number of customers visiting the store from the tract

x_1 = the number of housing units in the tract

x_2 = the average personal income in the tract (dollars/year)

x_3 = the average housing unit age in the tract (years)

x_4 = the distance from the tract to the nearest competing store (miles)

x_5 = the distance from the tract to the store (miles)

Here we will model customer counts as independent Poisson variables with $\mathrm{E}y_i = \lambda_i$ and

$$\ln \lambda_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + \beta_4 x_{4i} + \beta_5 x_{5i}$$

There is an `R` output for these data attached to this exam. Use it to help you answer the following.

8 pts

a) In the presence of all other predictors, **which** of the x's appears to be **least important** to the description of y? **Explain**.

8 pts

b) **What** are approximate 95% confidence limits for the log-mean number of visits by tract #1 customers in a two week period? **What** are corresponding approximate 95% limits for the mean number of such visits?

10 pts

c) **Find** an appropriate prediction standard error for the number of visits in a future two week period by tract #1 customers. (Hints: $\widehat{\ln\lambda} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5$. How is a standard error for $\hat{\lambda} = \exp\left(\widehat{\ln\lambda}\right)$ related to a standard error for $\widehat{\ln\lambda}$? How is $\text{Var}Y_{new}$ related to λ?)

8 pts

d) Census tract #41 has one of the largest values of y in the data set (and corresponding large value of $\hat{\lambda}$) and thus is an important source of customers for the store. A competitor is about to open a new store only .1 mile away from this tract. **By what fraction** does the fitted model suggest that the mean number of visits from tract #41 customers will decrease? **Provide** 95% confidence limits for this fraction.

2. Consider the making of fitted values $\hat{y}_i$ from n pairs (x_i, y_i) under a model that says

$$y_i = \mu(x_i) + \varepsilon_i$$

for some unknown mean function $\mu(x)$ where the ε_i are iid with mean 0 and variance σ^2. A standard measure of the flexibility of the fitting method employed is

$$flex = \frac{1}{\sigma^2}\sum_{i=1}^{n}\text{Cov}(\hat{y}_i, y_i)$$

For "linear" fitting methods (ones for which $\hat{\mathbf{Y}} = \mathbf{MY}$ for some fixed $n \times n$ matrix $\mathbf{M}$) this is fairly easily computed. (Consider the covariance matrix of the $2n \times 1$ vector $(\hat{\mathbf{Y}}', \mathbf{Y}')'$ computed beginning from $\text{Var}\mathbf{Y} = \sigma^2\mathbf{I}$.)

10 pts

a) **What** is numerical value of *flex* for simple linear regression fitting? (Note that here, $\mathbf{M} = \mathbf{P_X}$, the projection matrix onto the column space of the simple linear regression $\mathbf{X}$ matrix.) **Explain**.

10 pts

b) Many popular smoothing methods are linear methods. In particular, kernel smoothing is a linear method. For a small problem where $n = 4$, $x_1 = 1, x_2 = 2, x_3 = 3,$ and $x_4 = 4$, the kernel $K(u) = \exp(-u^2)$ used with bandwidth $b = 2$, produces the matrix $\mathbf{M}$ given below. **What** is *flex* in this context? **Explain**.

$$\mathbf{M} = \begin{pmatrix} .4440 & .3458 & .1634 & .0468 \\ .2662 & .3418 & .2662 & .1258 \\ .1258 & .2662 & .3418 & .2662 \\ .0468 & .1634 & .3458 & .4440 \end{pmatrix}$$

3. Several nominally identical bolts are used to hold face-plates on a model of transmission manufactured by an industrial concern. Some testing was done to determine the torque required to loosen bolts number 3 and 4 on $n = 34$ transmissions. Since the bolts are tightened simultaneously by two heads of a pneumatic wrench fed from a single compressed air line, it is natural to expect the torques on a single face-plate to be correlated. Printout #2 concerns several aspects of the analysis of 34 pairs of measured torques (in ft-lbs).

8 pts

a) **What** are approximate 90% confidence limits for the mean difference of bolt 4 and bolt 3 torques? **Explain**.

8 pts

b) **What** is a bootstrap standard error for the ratio of sample standard deviations $s_{\text{bolt3}} / s_{\text{bolt4}}$? **Explain**.

8 pts

c) **Why** would you expect the bootstrap to fail in the estimation of the upper .01 point for the bolt 4 torques in this situation?

4. The book *Statistical Analysis of Designed Experiments* by Tamhane considers an experiment run to study the corrosion resistance of 4 types of coating for steel bars. Steel bars were coated, baked, and tested for corrosion resistance as follows. An oven was set to one of 3 different temperatures (360°F, 370°F, or 380°F), 4 bars (one coated with each different coating) were loaded into the oven, all were baked for a fixed time, the bars were then removed and cooled, and corrosion testing was done (no units of measurement are stated for the response variable). After running the oven at each temperature once, the whole protocol was repeated. Let

y_{ijk} = a measured corrosion resistance of coating i under baking temperature j seen in the kth time the furnace is heated

and consider the model

$$y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + \gamma_k + \varepsilon_{ijk} \qquad (*)$$

for fixed effects $\mu, \alpha_i, \beta_j, \alpha\beta_{ij}$ (Factor A being Coating Type and Factor B being Temperature), and random effects γ_k and ε_{ijk} that are independent mean 0 normal variables, the γ_k with variance σ_γ^2 and the ε_{ijk} with variance σ^2. The γ_k are "firing" effects potentially peculiar to each different time $k = 1, 2, \ldots, 6$ that the oven is heated.

If one defines $l(1) = l(2) = l(3) = 1$ and $l(4) = l(5) = l(6) = 2$ the variable $l(k)$ specifies the replication (first or second) of which firing k is a part. In the event that there was a substantial time period between replications 1 and 2 of the experiment, it might make sense to entertain a generalization of model (*)

$$y_{ijk} = \mu + \alpha_i + \beta_j + \alpha\beta_{ij} + \delta_{l(k)} + \gamma_k + \varepsilon_{ijk} \qquad (**)$$

for δ_1 and δ_2 iid $\mathrm{N}(0, \sigma_\delta^2)$ independent of the γ_k and ε_{ijk}.

There is an `R` output based on Tamhane's data attached to this exam. Use it as needed to help you answer the following questions.

10 pts

a) **Exactly what** about the design of this study guarantees that the variance component σ_δ^2 of model (**) will be poorly determined? (Use 25 words or less, and do NOT to refer to any number from the data analysis on the output to answer this question.)

12 pts

b) The fundamental difference between models (*) and (**) is in their covariance structures. In terms of the variance components $\sigma_\gamma^2, \sigma_\delta^2,$ and σ^2 **fill in the table** below comparing the models.

	Model (*)	**Model (**)**
Var y_{ijk}		
covariance between two y's from different replications		
covariance between two y's from the same replication but different firings		
covariance between two y's from the same firing		

12 pts

c) Based on the R output, **argue carefully** that REML can't really distinguish between models (*) and (**) here. (Your answer to b) might be useful.)

Henceforth use model (*).

12 pts

d) The two from observations from coating i under baking temperature j could possibly be used to compute a (sample size 2) sample variance s_{ij}^2. **Identify** a constant c and degrees of freedom ν so that $cs_{ij}^2 \sim \chi_\nu^2$. **Why** can one not conclude that $\sum_{i,j} cs_{ij}^2 \sim \chi_{12\nu}^2$?

$c =$ ______________ $\nu =$ ______________

Why?:

12 pts

e) **Is there** definitive statistical evidence which of the two standard deviations σ_γ and σ is largest? **Explain**.

12 pts

f) **Is there** definitive statistical evidence of a difference in Temperature 1 and Temperature 2 main effects? **Explain**.

12 pts

g) If the object is to maximize mean corrosion resistance, **what combination of level**s of Coating and Temperature is indicated to be best by the study? Explain.

Printout for Problem 1

```
> Miller
     Y   X1     X2 X3   X4   X5
1    9  606  41393  3 3.04 6.32
2    6  641  23635 18 1.95 8.89
3   28  505  55475 27 6.54 2.05
4   11  866  64646 31 1.67 5.81
5    4  599  31972  7 0.72 8.11
6    4  520  41755 23 2.24 6.81
7    0  354  46014 26 0.77 9.27
8   14  483  34626  1 3.51 7.92
9   16 1034  85207 13 4.23 4.40
10  13  456  33021 32 3.07 6.03
.
.
.
40  16  234  33246 26 3.95 4.61
41  29 1004  45927 24 4.90 2.69
42   6  643  58315  8 0.78 6.26
.
.
.
107 10  752  71814  1 3.13 5.47
108  6  817  54429 47 1.90 9.90
109  4  268  34022 54 1.20 9.51
110  6  519  52850 43 2.92 8.62

> miller.glm <- glm(Y ~ X1 + X2 + X3 + X4 + X5, family=poisson())

> summary(miller.glm)

Call:
glm(formula = Y ~ X1 + X2 + X3 + X4 + X5, family = poisson())

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-2.932e+00  -5.887e-01  -9.434e-05   5.927e-01   2.234e+00

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.942e+00  2.072e-01  14.198  < 2e-16 ***
X1           6.058e-04  1.421e-04   4.262 2.02e-05 ***
X2          -1.169e-05  2.112e-06  -5.534 3.13e-08 ***
X3          -3.726e-03  1.782e-03  -2.091   0.0365 *
X4           1.684e-01  2.577e-02   6.534 6.39e-11 ***
X5          -1.288e-01  1.620e-02  -7.948 1.89e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 422.22  on 109  degrees of freedom
Residual deviance: 114.99  on 104  degrees of freedom
AIC: 571.02

Number of Fisher Scoring iterations: 4

> lambda <- predict(miller.glm, se.fit = TRUE)


> lambda$fit
       1        2        3        4        5        6        7        8        9
2.512666 2.171006 3.336689 2.129078 1.982460 2.184004 1.458190 2.397791 2.670277
.
.
.
      37       38       39       40       41       42       43       44       45
1.402975 2.162261 1.927029 2.670250 3.403162 1.945876 3.550344 2.967052 1.418526
```

```
.
.
.
     100      101      102      103      104      105      106      107      108 
2.489542 2.283455 2.420277 2.348353 2.479855 2.020812 2.492500 2.377671 1.671211 
     109      110 
1.483391 1.860632 
```

```
> exp(lambda$fit)
        1         2         3         4         5         6         7         8 
12.337775  8.767095 28.125856  8.407109  7.260584  8.881802  4.298174 10.998852 
.
.
.
       33        34        35        36        37        38        39        40 
 3.932143  7.506397  9.177026  6.673720  4.067281  8.690762  6.869074 14.443576 
       41        42        43        44        45        46        47        48 
30.058982  6.999761 34.825300 19.434551  4.131026  8.615824  8.502551 19.075131 
.
.
.
      105       106       107       108       109       110 
 7.544452 12.091472 10.779767  5.318607  4.407869  6.427799 
```

```
> lambda$se.fit
         1          2          3          4          5          6          7 
0.05467314 0.06170630 0.07170864 0.06815293 0.07410901 0.04263091 0.07558357 
.
.
.
        36         37         38         39         40         41         42 
0.06998541 0.10991558 0.07588834 0.06034853 0.06590329 0.07482010 0.08653370 
.
.
.
       106        107        108        109        110 
0.04978809 0.06757000 0.07488225 0.08863552 0.06517974 
```

```
> confint(miller.glm)
Waiting for profiling to be done...
                    2.5 %        97.5 %
(Intercept)  2.536768e+00  3.349269e+00
X1           3.273675e-04  8.845217e-04
X2          -1.585282e-05 -7.574589e-06
X3          -7.222279e-03 -2.361904e-04
X4           1.176987e-01  2.187234e-01
X5          -1.608136e-01 -9.728963e-02
```

```
> vcov(miller.glm)
              (Intercept)            X1            X2            X3            X4
(Intercept)  4.295161e-02 -3.700599e-06 -1.396580e-07 -9.186108e-05 -3.473158e-03
X1          -3.700599e-06  2.019844e-08 -1.739201e-10 -4.725457e-08 -3.749918e-08
X2          -1.396580e-07 -1.739201e-10  4.459139e-12  5.395604e-10 -8.008442e-09
X3          -9.186108e-05 -4.725457e-08  5.395604e-10  3.175228e-06 -6.657146e-07
X4          -3.473158e-03 -3.749918e-08 -8.008442e-09 -6.657146e-07  6.640413e-04
X5          -2.930245e-03  3.096327e-09  8.002867e-09  2.108939e-06  2.529766e-04
                       X5
(Intercept) -2.930245e-03
X1           3.096327e-09
X2           8.002867e-09
X3           2.108939e-06
X4           2.529766e-04
X5           2.624963e-04
```

Printout for Problem 3

```
> pairs
   bolt3 bolt4
1     16    16
2     15    16
3     15    17
4     15    16
5     20    20
6     19    16
7     19    20
8     17    19
9     15    15
10    11    15
11    17    19
12    18    17
13    18    14
14    15    15
15    18    17
16    15    17
17    18    20
18    15    14
19    17    17
20    14    16
21    17    18
22    19    16
23    19    18
24    19    20
25    15    15
26    12    15
27    18    20
28    13    18
29    14    18
30    18    18
31    18    14
32    15    13
33    16    17
34    16    16

> mean(pairs)
   bolt3    bolt4
16.35294 16.82353

> sd(pairs)
   bolt3    bolt4
2.172589 1.961301

> B<-10000


> results<-bootstrap(bolt3,B,mean)

> mean(results$thetastar)
[1] 16.35864

> sd(results$thetastar)
[1] 0.3640961

> quantile(results$thetastar,seq(0,1,.05))
      0%       5%      10%      15%      20%      25%      30%      35%
14.94118 15.73529 15.88235 16.00000 16.05882 16.11765 16.17647 16.23529
     40%      45%      50%      55%      60%      65%      70%      75%
16.26471 16.32353 16.35294 16.41176 16.45294 16.50000 16.55882 16.61765
     80%      85%      90%      95%     100%
16.67647 16.73529 16.82353 16.97059 17.70588
```

```
> results<-bootstrap(bolt4,B,mean)

> mean(results$thetastar)
[1] 16.82381

> sd(results$thetastar)
[1] 0.3376682

> quantile(results$thetastar,seq(0,1,.05))
      0%       5%      10%      15%      20%      25%      30%      35% 
15.58824 16.26471 16.38235 16.47059 16.52941 16.58824 16.64706 16.70588 
     40%      45%      50%      55%      60%      65%      70%      75% 
16.73529 16.79412 16.82353 16.88235 16.91176 16.94118 17.00000 17.05882 
     80%      85%      90%      95%     100% 
17.11765 17.17647 17.26471 17.38235 18.11765 


> results<-bootstrap(bolt4-bolt3,B,mean)

> mean(results$thetastar)
[1] 0.4735147

> sd(results$thetastar)
[1] 0.3608869

> quantile(results$thetastar,seq(0,1,.05))
        0%         5%        10%        15%        20%        25%        30% 
-0.8235294 -0.1176471  0.0000000  0.0882353  0.1764706  0.2352941  0.2941176 
       35%        40%        45%        50%        55%        60%        65% 
 0.3235294  0.3823529  0.4411765  0.4705882  0.5294118  0.5588235  0.6176471 
       70%        75%        80%        85%        90%        95%       100% 
 0.6470588  0.7058824  0.7647059  0.8529412  0.9411765  1.0588235  1.9117647 


> results<-bootstrap(bolt3,B,sd)

> mean(results$thetastar)
[1] 2.132172

> sd(results$thetastar)
[1] 0.2359272

> quantile(results$thetastar,seq(0,1,.05))
      0%       5%      10%      15%      20%      25%      30%      35% 
1.268008 1.747165 1.825864 1.882701 1.925761 1.968559 2.004674 2.037964 
     40%      45%      50%      55%      60%      65%      70%      75% 
2.071150 2.102159 2.130753 2.162103 2.193777 2.225480 2.257094 2.292552 
     80%      85%      90%      95%     100% 
2.332900 2.377104 2.437892 2.522802 3.045115 


> results<-bootstrap(bolt4,B,sd)

> mean(results$thetastar)
[1] 1.924021

> sd(results$thetastar)
[1] 0.185172

> quantile(results$thetastar,seq(0,1,.05))
      0%       5%      10%      15%      20%      25%      30%      35% 
1.177629 1.612120 1.686689 1.728703 1.770475 1.800178 1.829643 1.857681 
     40%      45%      50%      55%      60%      65%      70%      75% 
1.879859 1.906925 1.930153 1.954245 1.974210 2.000000 2.022379 2.050390 
     80%      85%      90%      95%     100% 
2.079738 2.114377 2.157150 2.220740 2.625971 
```

```
> theta<-function(x,xdata)
+ {
+ sd(xdata[x,1])/sd(xdata[x,2])
+ }

> results<-bootstrap(1:34,B,theta,pairs)

> mean(results$thetastar)
[1] 1.115304

> sd(results$thetastar)
[1] 0.1596734

> quantile(results$thetastar,seq(0,1,.05))
       0%        5%       10%       15%       20%       25%       30% 
0.6256857 0.8664018 0.9168158 0.9521241 0.9790108 1.0054748 1.0264038 
      35%       40%       45%       50%       55%       60%       65% 
1.0478070 1.0667396 1.0861363 1.1061987 1.1258651 1.1473283 1.1684573 
      70%       75%       80%       85%       90%       95%      100% 
1.1917080 1.2168482 1.2439695 1.2794890 1.3243544 1.3880104 2.0475895 
```

Printout for Problem 4

```
> corrosion
   resist Coat Temp Firing
1      73    2    1      1
2      83    3    1      1
3      67    1    1      1
4      89    4    1      1
5      65    1    2      2
6      87    3    2      2
7      86    4    2      2
8      91    2    2      2
9     147    3    3      3
10    155    1    3      3
11    127    2    3      3
12    212    4    3      3
13     33    1    1      4
14     54    4    1      4
15      8    2    1      4
16     46    3    1      4
17    150    4    2      5
18    140    1    2      5
19    121    3    2      5
20    142    2    2      5
21    153    4    3      6
22     90    3    3      6
23    100    2    3      6
24    108    1    3      6

> options(contrasts=c("contr.sum","contr.sum"))

> tamhane1.lmer<-lmer(resist ~ 1 + Coat*Temp + (1|Firing), corrosion)

> summary(tamhane1.lmer)
Linear mixed model fit by REML 
Formula: resist ~ 1 + Coat * Temp + (1 | Firing) 
   Data: corrosion 
   AIC   BIC logLik deviance REMLdev
 156.3 172.8 -64.17    189.2   128.3


Random effects:
 Groups   Name        Variance Std.Dev.
 Firing   (Intercept) 1172.17  34.237  
 Residual              124.54  11.160  
Number of obs: 24, groups: Firing, 6
```

```
Fixed effects:
             Estimate Std. Error t value
(Intercept) 101.1250    14.1605   7.141
Coat1        -6.4583     3.9456  -1.637
Coat2       -10.9583     3.9456  -2.777
Coat3        -5.4583     3.9456  -1.383
Temp1       -44.5000    20.0259  -2.222
Temp2         9.1250    20.0259   0.456
Coat1:Temp1  -0.1667     5.5799  -0.030
Coat2:Temp1  -5.1667     5.5799  -0.926
Coat3:Temp1  13.3333     5.5799   2.390
Coat1:Temp2  -1.2917     5.5799  -0.231
Coat2:Temp2  17.2083     5.5799   3.084
Coat3:Temp2  -0.7917     5.5799  -0.142

Correlation of Fixed Effects:
            (Intr) Coat1  Coat2  Coat3  Temp1  Temp2  Ct1:T1 Ct2:T1 Ct3:T1 Ct1:T2 Ct2:T2
Coat1        0.000
Coat2        0.000 -0.333
Coat3        0.000 -0.333 -0.333
Temp1        0.000  0.000  0.000  0.000
Temp2        0.000  0.000  0.000  0.000 -0.500
Coat1:Temp1  0.000  0.000  0.000  0.000  0.000  0.000
Coat2:Temp1  0.000  0.000  0.000  0.000  0.000  0.000 -0.333
Coat3:Temp1  0.000  0.000  0.000  0.000  0.000  0.000 -0.333 -0.333
Coat1:Temp2  0.000  0.000  0.000  0.000  0.000  0.000 -0.500  0.167  0.167
Coat2:Temp2  0.000  0.000  0.000  0.000  0.000  0.000  0.167 -0.500  0.167 -0.333
Coat3:Temp2  0.000  0.000  0.000  0.000  0.000  0.000  0.167  0.167 -0.500 -0.333 -0.333

> anova(tamhane1.lmer)
Analysis of Variance Table
          Df Sum Sq Mean Sq F value
Coat       3 4289.1  1429.7 11.4798
Temp       2  682.2   341.1  2.7388
Coat:Temp  6 3269.8   545.0  4.3757

> tamhane2.lmer<-lmer(resist ~ 1 + Coat*Temp + (1|Firing) + (1|Replic), corrosion)

> summary(tamhane2.lmer)
Linear mixed model fit by REML
Formula: resist ~ 1 + Coat * Temp + (1 | Firing) + (1 | Replic)
   Data: corrosion
   AIC BIC logLik deviance REMLdev
 158.3 176 -64.17    189.2   128.3
Random effects:
 Groups   Name        Variance   Std.Dev.
 Firing   (Intercept) 1.1722e+03 3.4237e+01
 Replic   (Intercept) 9.0930e-18 3.0155e-09
 Residual             1.2454e+02 1.1160e+01
Number of obs: 24, groups: Firing, 6; Replic, 2


Fixed effects:
             Estimate Std. Error t value
(Intercept) 101.1250    14.1616   7.141
Coat1        -6.4583     3.9456  -1.637
Coat2       -10.9583     3.9456  -2.777
Coat3        -5.4583     3.9456  -1.383
Temp1       -44.5000    20.0275  -2.222
Temp2         9.1250    20.0275   0.456
Coat1:Temp1  -0.1667     5.5799  -0.030
Coat2:Temp1  -5.1667     5.5799  -0.926
Coat3:Temp1  13.3333     5.5799   2.390
Coat1:Temp2  -1.2917     5.5799  -0.231
Coat2:Temp2  17.2083     5.5799   3.084
Coat3:Temp2  -0.7917     5.5799  -0.142
```

```
Correlation of Fixed Effects:
            (Intr) Coat1  Coat2  Coat3  Temp1  Temp2  Ct1:T1 Ct2:T1 Ct3:T1 Ct1:T2 Ct2:T2
Coat1        0.000
Coat2        0.000 -0.333
Coat3        0.000 -0.333 -0.333
Temp1        0.000  0.000  0.000  0.000
Temp2        0.000  0.000  0.000  0.000 -0.500
Coat1:Temp1  0.000  0.000  0.000  0.000  0.000  0.000
Coat2:Temp1  0.000  0.000  0.000  0.000  0.000  0.000 -0.333
Coat3:Temp1  0.000  0.000  0.000  0.000  0.000  0.000 -0.333 -0.333
Coat1:Temp2  0.000  0.000  0.000  0.000  0.000  0.000 -0.500  0.167  0.167
Coat2:Temp2  0.000  0.000  0.000  0.000  0.000  0.000  0.167 -0.500  0.167 -0.333
Coat3:Temp2  0.000  0.000  0.000  0.000  0.000  0.000  0.167  0.167 -0.500 -0.333 -0.333

> anova(tamhane2.lmer)
Analysis of Variance Table
          Df Sum Sq Mean Sq F value
Coat       3 4289.1  1429.7 11.4798
Temp       2  686.2   343.1  2.7548
Coat:Temp  6 3269.8   545.0  4.3757

> sims <- mcmcsamp(tamhane1.lmer, 50000)

> HPDinterval(sims)
$fixef
                 lower      upper
(Intercept)  82.547850 118.673994
Coat1       -25.154202  11.457372
Coat2       -29.417369   7.078033
Coat3       -23.732899  13.052202
Temp1       -70.296770 -19.465729
Temp2       -16.510976  34.181738
Coat1:Temp1 -26.070571  25.367604
Coat2:Temp1 -30.948958  20.575927
Coat3:Temp1 -11.972080  39.663302
Coat1:Temp2 -27.727084  23.709982
Coat2:Temp2  -8.811398  42.927610
Coat3:Temp2 -26.610202  24.517419
attr(,"Probability")
[1] 0.95

$ST
     lower    upper
[1,]     0 1.213842
attr(,"Probability")
[1] 0.95

$sigma
        lower    upper
[1,] 12.67674 39.12244
attr(,"Probability")
[1] 0.95

> fitted(tamhane1.lmer)
 [1]  65.85394  92.37750  83.05505  89.37893  73.70902  76.37162  93.92486  85.47721
 [9] 144.18821 138.40850 120.00251 191.41291  40.87336  47.19724  23.67225  50.19581
[17] 137.57972 117.36388 120.02648 129.13206 155.60572 108.38102  84.19532 102.60130

> ranef(tamhane1.lmer)
$Firing
  (Intercept)
1    21.23460
2   -79.41079
3    61.62743
4  -164.23078
5   112.53185
6   -95.81043
```

```
> vcov(tamhane1.lmer)
12 x 12 Matrix of class "dpoMatrix"
         [,1]      [,2]      [,3]      [,4]     [,5]     [,6]      [,7]      [,8]
 [1,] 97.9527   0.00000   0.00000   0.00000   0.0000   0.0000   0.00000   0.00000
 [2,]  0.0000  99.70052 -33.23351 -33.23351   0.0000   0.0000   0.00000   0.00000
 [3,]  0.0000 -33.23351  99.70052 -33.23351   0.0000   0.0000   0.00000   0.00000
 [4,]  0.0000 -33.23351 -33.23351  99.70052   0.0000   0.0000   0.00000   0.00000
 [5,]  0.0000   0.00000   0.00000   0.00000 195.9054 -97.9527   0.00000   0.00000
 [6,]  0.0000   0.00000   0.00000   0.00000 -97.9527 195.9054   0.00000   0.00000
 [7,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000 199.40103 -66.46701
 [8,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000 -66.46701 199.40103
 [9,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000 -66.46701 -66.46701
[10,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000 -99.70052  33.23351
[11,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000  33.23351 -99.70052
[12,]  0.0000   0.00000   0.00000   0.00000   0.0000   0.0000  33.23351  33.23351
           [,9]     [,10]     [,11]     [,12]
 [1,]   0.00000   0.00000   0.00000   0.00000
 [2,]   0.00000   0.00000   0.00000   0.00000
 [3,]   0.00000   0.00000   0.00000   0.00000
 [4,]   0.00000   0.00000   0.00000   0.00000
 [5,]   0.00000   0.00000   0.00000   0.00000
 [6,]   0.00000   0.00000   0.00000   0.00000
 [7,] -66.46701 -99.70052  33.23351  33.23351
 [8,] -66.46701  33.23351 -99.70052  33.23351
 [9,] 199.40103  33.23351  33.23351 -99.70052
[10,]  33.23351 199.40103 -66.46701 -66.46701
[11,]  33.23351 -66.46701 199.40103 -66.46701
[12,] -99.70052 -66.46701 -66.46701 199.40103
```