Ecol 145 Final Exam
Dahl Winters
5/8/06

Problem 1

Three shells were collected from one stream (Boone) and four shells from a second stream (Buffalo). Multiple sections of each shell were then analyzed and a carbon isotope composition obtained for each shell section. The goal is to determine if there is a systematic difference in the carbon isotope composition of shells from the two different streams.

1. Fit two different models such that when these two models are compared in an appropriate fashion, the question posed above can be answered statistically. Carry out a statistical significance test using the two models to answer the researcher's question.

The models would have to be 2-level models with and without stream as the predictor.

- Level 1: repeated measures of carbon isotope composition on individual shells
- Level 2: the shells

If the two streams have different intercepts, then there must be a difference in the carbon isotope composition of the shells coming from the two different streams.

```
shells<-
read.table('http://www.unc.edu/courses/2006spring/ecol/145/001/data/final/pro
blem1.csv',header=TRUE,sep=',')
names(shells)
[1] "CARBONISOTOPE" "STREAM"        "SHELL"

library(nlme)
model1a<-lme(CARBONISOTOPE~1, random=~1|SHELL, data=shells, method='ML')
model1b<-lme(CARBONISOTOPE~STREAM, random=~1|SHELL, data=shells, method='ML')
anova(model1a,model1b)
        Model df      AIC      BIC    logLik   Test  L.Ratio p-value
model1a     1  3 1085.363 1097.104 -539.6815
model1b     2  4 1083.148 1098.802 -537.5742 1 vs 2 4.214755  0.0401
```

The p-value given by the likelihood ratio test above is just below the significance cutoff of 0.05, but it is still significant. This analysis suggests there is a difference in the carbon isotope composition of shells from the two different streams.

2. Carry out a second statistical test that answers this same question but only requires you to fit a single model. Explain why this test and the test in question 1 are yielding different conclusions.

```
summary(model1b)
Fixed effects: CARBONISOTOPE ~ STREAM
                  Value Std.Error  DF    t-value p-value
(Intercept)   -9.731287 0.5704290 363 -17.059593  0.0000
STREAMbuffalo -1.810200 0.7541505   5  -2.400317  0.0616
```

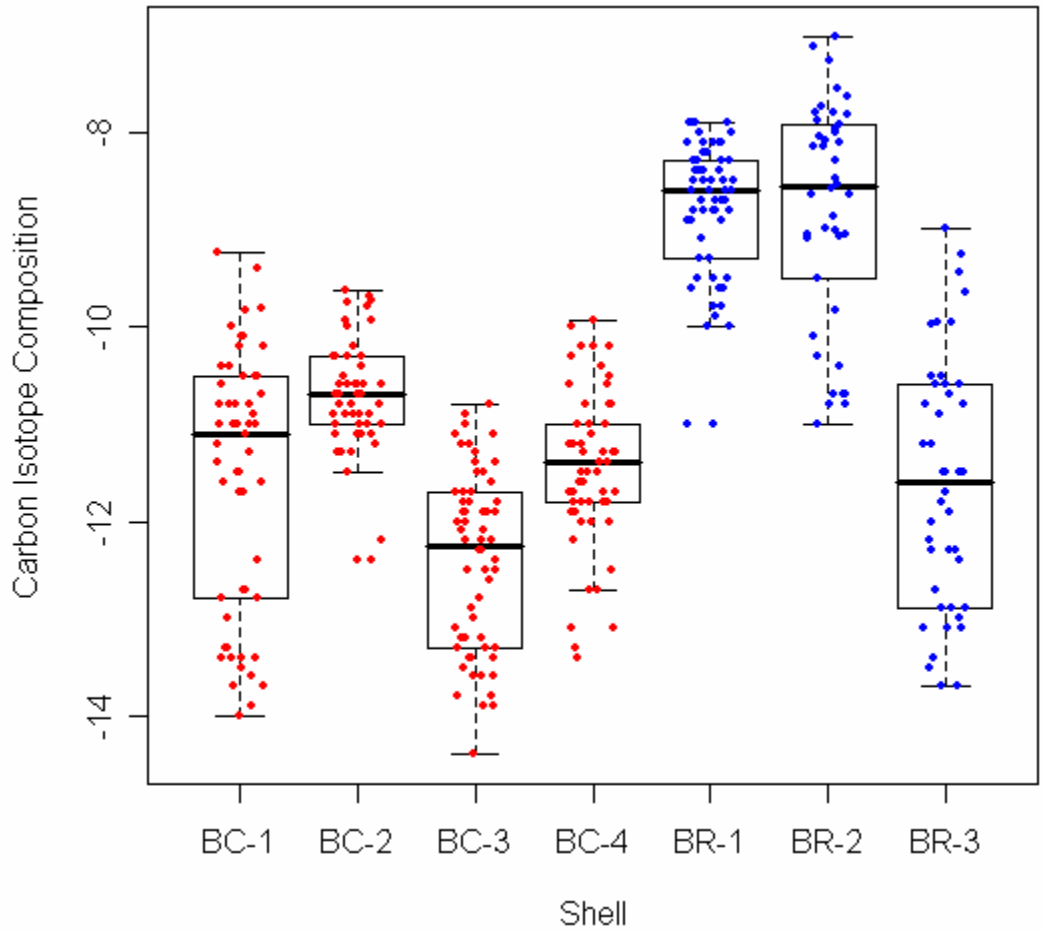
The summary output for the model with stream as a predictor (model2) gives the p-value from a Wald test for the predictor, which is nonsignificant if we consider 0.05 to be our cutoff. This would suggest there is no difference in carbon isotope composition of shells between streams.

The Wald test and the likelihood ratio test are likely yielding different conclusions because of the small sample size of shells present. There may be 370 different observations of C isotope composition, but those 370 observations are coming from just 7 shells. It is those 7 shells that we're using to try to establish whether there's a difference in C isotope composition between the two streams. The LR test is better at handling small sample sizes than the Wald test, so its conclusion should probably carry more weight.

3. Produce a single graph that best summarizes this experiment. Choose a graph that displays as much information as possible without becoming overly busy and confusing.

```
boxplot(shells$CARBONISOTOPE~shells$SHELL, ylab='Carbon Isotope
Composition',xlab='Shell',outline=FALSE)
points(jitter(as.numeric(shells$SHELL[shells$STREAM=='buffalo'])),
shells$CARBONISOTOPE[shells$STREAM=='buffalo'], pch=16, col=2, cex=.7)
points(jitter(as.numeric(shells$SHELL[shells$STREAM=='boone'])),
shells$CARBONISOTOPE[shells$STREAM=='boone'], pch=16, col=4, cex=.7)
```

The two streams are color-coded red for Buffalo and blue for Boone.



4. Finally, as hard as this might be for you to do, I would like you to carry out an incorrect statistical analysis that addresses the researcher's question. The analysis I'm looking for is not just incorrect but egregiously incorrect. Answer the following two questions about your incorrect analysis.
    - What's wrong with it?
    - Why are its conclusions so different from the correct analyses you did in questions 1 and 2?

The incorrect analysis would be if the nesting structure weren't taken into account. It would give a case of pseudoreplication, where treatment effects are being tested for with replicates that are not statistically independent because structure exists in the data (Hurlbert SH 1984, Ecological Monographs). Such an analysis is incorrect because it assumes all measurements of carbon isotope composition on the 7 shells are independent of each other, when in fact they were taken from different shells. Neglecting the nesting structure means neglecting that there might be something about each shell that causes the carbon isotope measurements on it to differ from those measured on other shells.

```
model1c<-lm(CARBONISOTOPE~STREAM, data=shells)
summary(model1c)
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    -9.6302     0.1126  -85.53   <2e-16 ***
STREAMbuffalo  -1.9562     0.1454  -13.46   <2e-16 ***
```

The conclusion of this incorrect analysis is that stream is a highly significant predictor of carbon isotope composition. Instead of getting p-values hovering around 0.05, the p-value for stream is exceedingly small. Therefore, not taking the nesting structure into account in this incorrect analysis would lead researchers to conclude that stream was a highly significant predictor, when in fact it may not be.

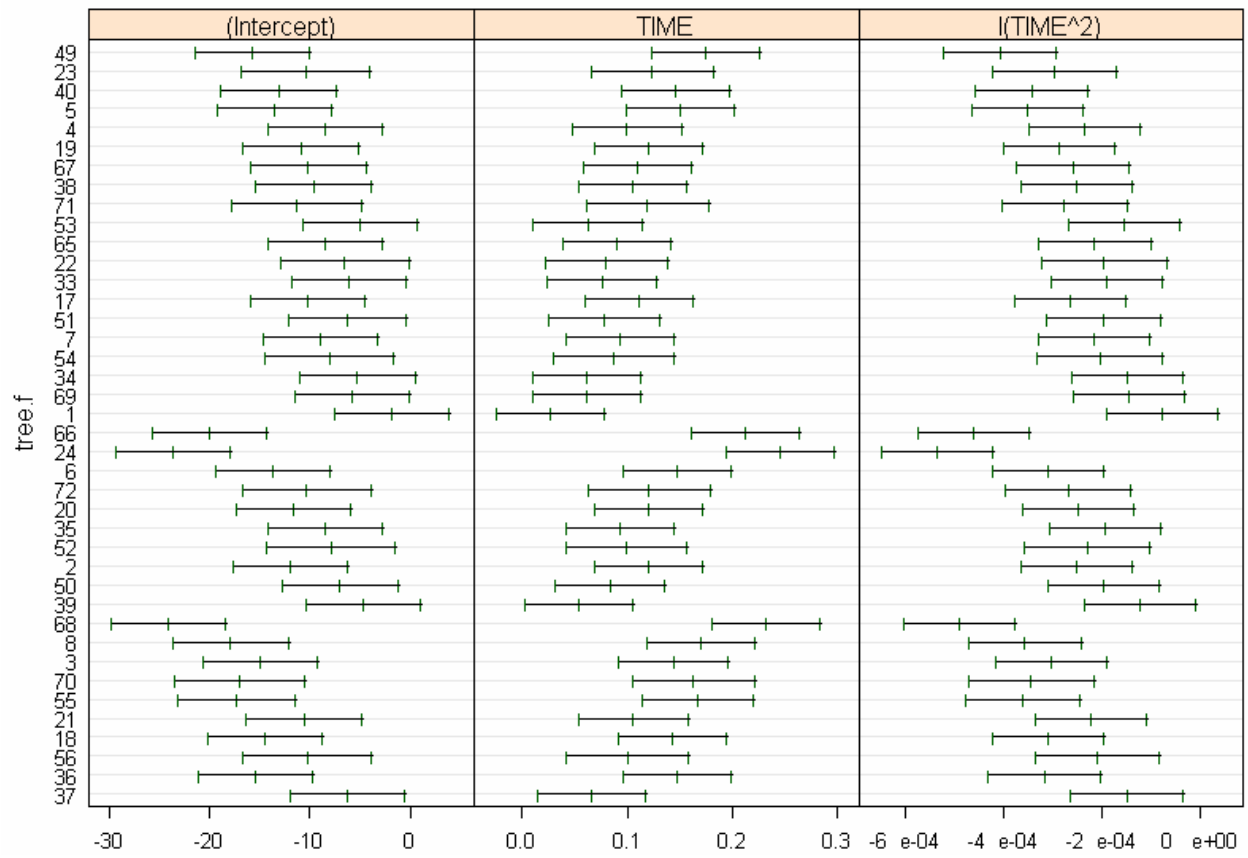
Problem 2

Find the most parsimonious model that accounts for all the relevant differences in water usage among tree species (SPECIES) and tree age classes (AGE).

```
waterdata<-
read.table('http://www.unc.edu/courses/2006spring/ecol/145/001/data/lab12/WaterUsageData.csv', header=TRUE, sep=',')

names(waterdata)
[1] "YEAR"    "SPECIES" "AGE"     "TREE"    "TIME"    "WU"      "TREECNT"
[8] "ID"

waterdata$tree.f<-factor(waterdata$TREE)
waterdata$species.f<-factor(waterdata$SPECIES)
waterdata$age.f<-factor(waterdata$AGE)

waterdata[1:3,]
  YEAR SPECIES AGE TREE TIME   WU TREECNT    ID tree.f species.f age.f
1    1       1   1    3  161 0.88       1 11101      3         1     1
2    1       1   1    8  161 1.00       2 11102      8         1     1
3    1       1   1   18  161 1.30       3 11103     18         1     1
```

Producing a plot to quantify parameter variability among level 2 units

```
library(lattice)
trellis.par.set(col.whitebg())
water.grp<-groupedData(WU~TIME|tree.f,data=waterdata,outer=~species.f*age.f)
out1<-lmList(WU~TIME+I(TIME^2),data=water.grp)
plot(intervals(out1))
```



Finding which trees are of which species and which age, to check if the plot has been grouped correctly from bottom to top according to species and age

```
names(table(waterdata$TREE[waterdata$SPECIES==1 & waterdata$AGE==1]))
[1] "3"  "8"  "18" "21" "36" "37" "55" "56" "68" "70" #37-68: SPECIES=1, AGE=1
names(table(waterdata$TREE[waterdata$SPECIES==1 & waterdata$AGE==2]))
[1] "2"  "6"  "20" "24" "35" "39" "50" "52" "66" "72" #39-66: SPECIES=1, AGE=2
names(table(waterdata$TREE[waterdata$SPECIES==2 & waterdata$AGE==1]))
[1] "1"  "7"  "17" "22" "33" "34" "51" "54" "65" "69" #1-65: SPECIES=2, AGE=1
names(table(waterdata$TREE[waterdata$SPECIES==2 & waterdata$AGE==2]))
 [1] "4"  "5"  "19" "23" "38" "40" "49" "53" "67" "71" #53-49: SPECIES=2, AGE=2
```

All three parameters are varying a lot, so the plot suggests potentially all of these could be added in as a random effect. Furthermore, it looks like there is an effect of species and age on all three parameters, so such a model would be a good model to start with.

```
model2a<-lme(WU~(TIME+I(TIME^2))*species.f*age.f,
random=~TIME+I(TIME^2)|tree.f , data=waterdata, method='ML')

(This is equivalent to the following)
lme(WU ~ TIME + I(TIME^2) + species.f + age.f + species.f:age.f +
TIME:species.f + TIME:age.f + TIME:species.f:age.f +
I(TIME^2):species.f + I(TIME^2):age.f + I(TIME^2):species.f:age.f,
random=~TIME+I(TIME^2)|tree.f , data=waterdata, method='ML')


summary(model2a)
Fixed effects: WU ~ (TIME + I(TIME^2)) * species.f * age.f
```

```
                                Value Std.Error  DF    t-value p-value
(Intercept)                -14.679421 0.9707256 944 -15.122111  0.0000
TIME                         0.145740 0.0087520 944  16.652240  0.0000
I(TIME^2)                   -0.000304 0.0000193 944 -15.758796  0.0000
species.f2                   8.144862 1.3725799  36   5.933980  0.0000
age.f2                       2.775041 1.3711211  36   2.023921  0.0504
TIME:species.f2             -0.068191 0.0123765 944  -5.509761  0.0000
I(TIME^2):species.f2         0.000122 0.0000273 944   4.478353  0.0000
TIME:age.f2                 -0.013334 0.0123609 944  -1.078745  0.2810
I(TIME^2):age.f2             0.000022 0.0000273 944   0.820137  0.4123
species.f2:age.f2           -6.906683 1.9388840  36  -3.562195  0.0011
TIME:species.f2:age.f2       0.058497 0.0174805 944   3.346426  0.0009
I(TIME^2):species.f2:age.f2 -0.000124 0.0000385 944  -3.219886  0.0013
```

The Wald tests suggest all parameters are significant except the two-parameter interaction terms of TIME:age and TIME^2:age, and age itself is just above 0.05. However, because the three-parameter interaction terms are significant, the law of marginality says I can't get rid of these non-significant terms. I will try to remove species:age from the intercept, time, and time^2 separately and see if I get a reduction in the AIC. Then I can try removing the two-parameter interaction terms.

```
#removing the interaction term from the intercept only.
model2b<-update(model2a, .~.-species.f:age.f)
#removing the interaction term from time only.
model2c<-update(model2a, .~.-species.f:age.f:TIME)
#removing the interaction term from time^2 only.
model2d<-update(model2a, .~.-species.f:age.f:I(TIME^2))
#removing the interaction term from all 3 parameters.
model2e<-lme(WU~(TIME+I(TIME^2))*species.f+age.f,
random=~TIME+I(TIME^2)|tree.f , data=waterdata, method='ML')
#Finding AICs
sapply(list(model2a,model2b,model2c,model2d,model2e),AIC)
[1] 1328.665 1339.321 1337.719 1336.919 1340.807
```

The AIC results suggest not to remove any of the interaction terms, so the best model (model2a) can't be simplified by removing any of the species or age terms (which are nested in species:age) because of the law of marginality. So next I will have to try removing random effects to improve the model. Based on the results from Assignment 11 (when species and age weren't added into the model), the quadratic random effect was unnecessary, so I will try removing it first here. If I get a lower AIC, I will then try to remove TIME as a random effect.

```
#model2a (the best model so far), written out in its entirety:
model2a<-lme(WU~(TIME+I(TIME^2))*species.f*age.f,
random=~TIME+I(TIME^2)|tree.f , data=waterdata, method='ML')
#removing time^2:
model2f<-lme(WU~(TIME+I(TIME^2))*species.f*age.f,
random=~TIME|tree.f , data=waterdata, method='ML')
#AIC results
sapply(list(model2a,model2f),AIC)
[1] 1328.665 1325.678
```

I do get a better model by removing TIME^2 from the random effects. Now I'll try removing TIME.

```
#removing time:
model2g<-lme(WU~(TIME+I(TIME^2))*species.f*age.f,
random=~1|tree.f , data=waterdata, method='ML')
```

#AIC results
```
sapply(list(model2f,model2g,model2h),AIC)
[1] 1328.665 1325.678 1339.047
```

Removing time makes the AIC increase, so it looks like the best model needs to have random effects for both the intercept and TIME, which is model2f.

Finally I need to account for a correlation structure because there is a very high negative correlation between TIME and TIME^2.

Accounting for correlation structure – model2f with no correlation, model2fc with corCAR1
```
model2f<-lme(WU~(TIME+I(TIME^2))*species.f*age.f, random=~TIME|tree.f ,
data=waterdata, method='ML')

model2fc<-lme(WU~(TIME+I(TIME^2))*species.f*age.f, random=~TIME|tree.f ,
data=waterdata, method='ML', correlation=corCAR1(form=~TIME|tree.f))

sapply(list(model2f,model2fc),AIC)
[1] 1325.678 1296.588
```

Incorporating a corCAR1 correlation structure gives a 29-unit reduction in AIC, which is substantial.

From this analysis, the best model describing water use (model2fc) has TIME, TIME^2, species, and age as predictors, and has a corCAR1 correlation structure. The model seems to make biological sense—species*age should be included because water use by trees generally depends on both species and age. Larger trees need more water, and older trees also need more water (because older trees tend to be larger, which motivates the need for the interaction term species:age).

Problem 3

Dengue fever is endemic in many parts of the world and is spreading. The data here consist of the daily admission records of hospitals and clinics throughout Sri Lanka over a many year period. There are only two variables.

The goal of the analysis is to determine if the appearance of diseased individuals at clinics is in any way related to the lagged temperature variable.

1. Build the best model you can that relates the disease presence-absence variable (DISEASE) to the temperature variable (temp).

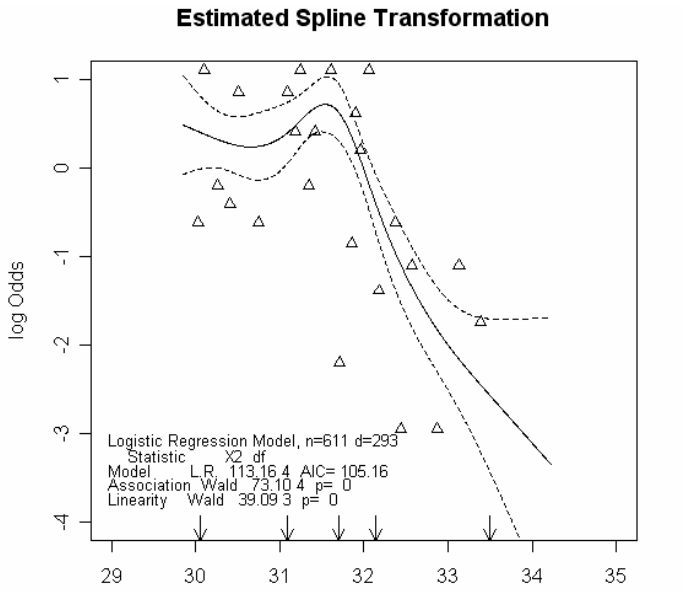
```
data<-
read.table('http://www.unc.edu/courses/2006spring/ecol/145/001/data/final/pro
blem3.csv',header=TRUE,sep=',')
names(data)
[1] "DISEASE" "temp"
data<-data[!is.na(data$temp),] #gets rid of the NAs in the dataset
```

The binary response variable Y for presences and absences of disease
```
Y<-ifelse(data$DISEASE==0, 1, 0)
Y
  [1] 0 0 1 1 1 1 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 0 1 1
 [35] 1 1 1 0 1 0 0 0 0 0 0 0 1 1 1 1 1 1 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0
data$DISEASE #to see if 0s have converted to 1s and vice versa
```

```
 [1] 1 1 0 0 0 0 0 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 1 0 0 1 0 0
[35] 0 0 0 1 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 1 1 0 0 1 1
```

Deciding on a linear vs. quadratic logistic regression model for the binomial data

```
library(Design)
rcspline.plot(y=Y,x=data$temp,nk=5,m=20)
```



The Wald test for linearity has a p-value below 0.05, which rejects linearity. There is a hump in the plot that rises higher than either end of the graph, which would suggest a quadratic model. The AIC results also suggest a quadratic model would be best:

#No predictor, for comparison

```
model3a<-glm(Y~1,data=data,family=binomial,na.action=na.omit)
```

#temp as a linear term

```
model3b<-glm(Y~temp,data=data,family=binomial,na.action=na.omit)
```

#temp as linear and quadratic terms

```
model3c<-glm(Y~temp+I(temp^2),data=data,family=binomial,na.action=na.omit)
```

```
sapply(list(model3a,model3b,model3c),AIC)
[1] 848.0027 780.6152 750.1037
```

The quadratic model (model3c) has the lowest AIC and so is the best model.

Wald significance test from model3c summary output:

The p-values for all parameters are all far below 0.05. These results for the temperature variable (both terms) indicate that temperature has a significant effect on the presence/absence of disease.

```
summary(model3c)
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -468.9778    96.9742  -4.836 1.32e-06 ***
temp          30.4773     6.1826   4.929 8.24e-07 ***
I(temp^2)     -0.4945     0.0985  -5.020 5.16e-07 ***
```

2. Carry out whatever tests you deem appropriate to demonstrate the quality of your best model.

I carried out two goodness of fit tests, but they gave conflicting results.

Goodness of fit test #1: Hosmer-Lemeshow test

```
p.groups<-cut(fitted(model3c),quantile(fitted(model3c),seq(0,1,.1)),
include.lowest=TRUE)
table(p.groups)
p.groups
[0.00351,0.113]   (0.113,0.339]   (0.339,0.469]   (0.469,0.525]
             62              62              62              70
  (0.525,0.549]   (0.549,0.585]    (0.585,0.61]    (0.61,0.632]
             59              54              60              69
  (0.632,0.642]   (0.642,0.652]
             52              61
```

Obtain observed counts of presences/absences in each category

```
Oi<-table(Y,p.groups)
Oi
   p.groups
Y   [0.00351,0.113] (0.113,0.339] (0.339,0.469] (0.469,0.525]
  0              56             54             41             30
  1               6              8             21             40
   p.groups
Y   (0.525,0.549] (0.549,0.585] (0.585,0.61] (0.61,0.632] (0.632,0.642]
  0            15            29           22           31            14
  1            44            25           38           38            38
   p.groups
Y   (0.642,0.652]
  0            26
  1            35
```

The total counts in each category

```
ni<-apply(Oi,2,sum)
ni
[0.00351,0.113]   (0.113,0.339]   (0.339,0.469]   (0.469,0.525]
             62              62              62              70
  (0.525,0.549]   (0.549,0.585]    (0.585,0.61]    (0.61,0.632]
             59              54              60              69
  (0.632,0.642]   (0.642,0.652]
             52              61
```

Obtain expected failures and successes

```
Ei<-rbind(ni-tapply(fitted(model3c),p.groups,sum),tapply(fitted(model3c),
p.groups,sum))
Ei
     [0.00351,0.113] (0.113,0.339] (0.339,0.469] (0.469,0.525]
[1,]       58.990372      46.80970      36.92899      34.59923
[2,]        3.009628      15.19030      25.07101      35.40077
     (0.525,0.549] (0.549,0.585] (0.585,0.61] (0.61,0.632] (0.632,0.642]
[1,]      27.09650      23.09256     24.22101     25.95281      18.84356
[2,]      31.90350      30.90744     35.77899     43.04719      33.15644
     (0.642,0.652]
[1,]      21.46526
[2,]      39.53474
```

Carry out H-L test

```
sum((Oi-Ei)^2/Ei)
[1] 27.92211
1-pchisq(sum((Oi-Ei)^2/Ei),df=8)
[1] 0.0004892879
```

The Hosmer-Lemeshow test gives a p-value (0.00049) that is below 0.05, which suggests the model has a significant lack of fit.

Goodness of fit test #2: Harrell's H-L test alternative

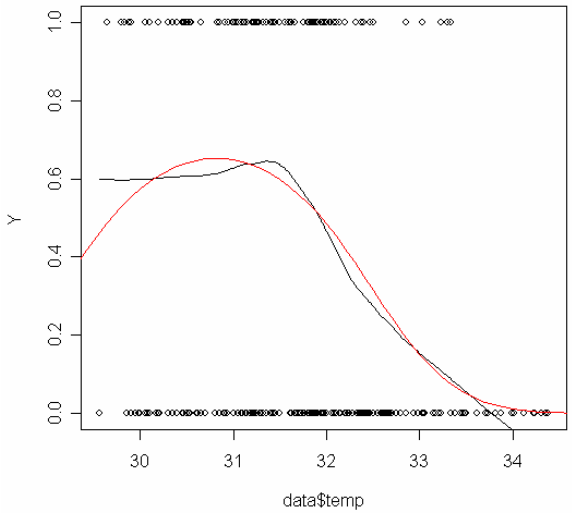
This needs to be centered because the linear and quadratic terms are so highly correlated.

```
library(Design)
temp<-data$temp-mean(data$temp)
temp2<-data$temp^2-mean(data$temp^2)
out.h<-lrm(Y~temp+temp2, data=data, x=TRUE, y=TRUE)
residuals.lrm(out.h,type='gof')
Sum of squared errors     Expected value|H0                    SD
          130.1040898           130.6195580             0.5779676
                    Z                     P
           -0.8918634             0.3724661
```

This test gives a p-value above 0.05, which suggests the opposite of the H-L test. However, the H-L test isn't supposed to be as good as this alternative test. Both Harrell's test and graphing the model along with a lowess curve to the actual data show that the model really does have a good fit.

Lowess curve

```
plot(data$temp,Y)
lines(lowess(Y~data$temp))

coef(model3c)
 (Intercept)         temp    I(temp^2)
-468.9777870   30.4773268   -0.4944927

lines(seq(0,35,.1), 1/(1+exp(-
coef(model3c)[1]-
coef(model3c)[2]*seq(0,35,.1)-
coef(model3c)[3]*seq(0,35,.1)^2 )), col=2)
```



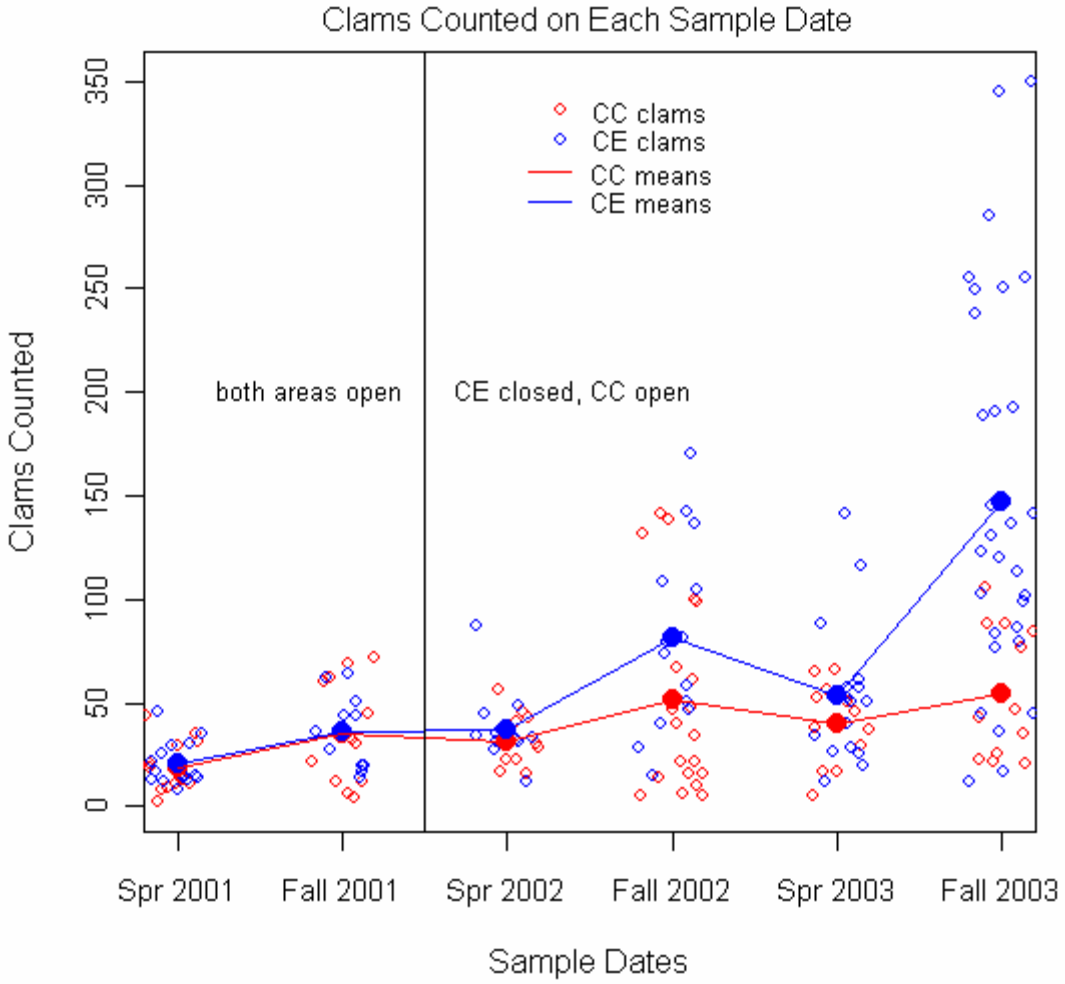
Problem 4

The goal is to develop a suitable model of the number of clams harvested during the course of this study such that the model adequately describes the data and at the same time addresses the question of whether the NCDMF rotation plan is a viable means of protecting the clam population.

1. Begin by plotting the data over time. Your plot should exhibit the following features.

```
clams<-
read.table('http://www.unc.edu/courses/2006spring/ecol/145/001/data/final/pro
blem4.csv',header=TRUE,sep=',')

names(clams)
[1] "TREATMENT"    "YEAR"         "SEASON"       "LEGAL.CLAMS"      "DATE"

#  Making sure the sample periods will plot in the correct order
date.f<-factor(clams$DATE, levels =
c('SPR2001','FALL2001','SPR2002','FALL2002','SPR2003','FALL2003'))
```

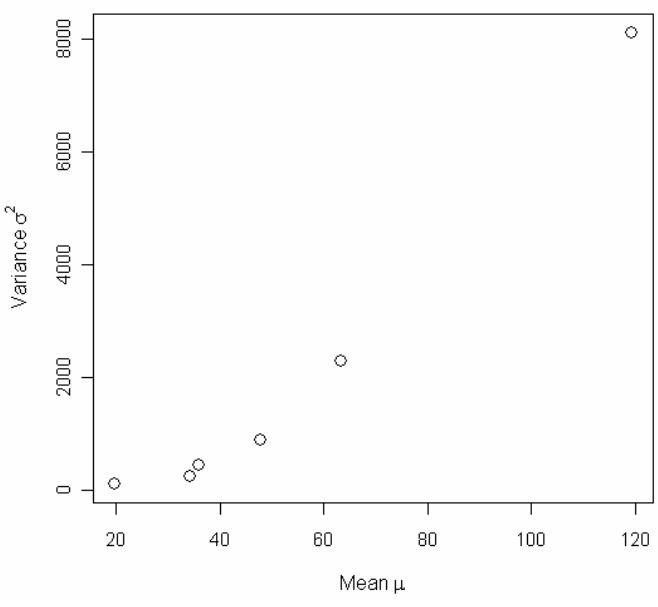
```
clams$DATES<-as.numeric(date.f)-1
clams$DATES
  [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
 [35] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [69] 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[103] 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
[137] 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
[171] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5

# mean of each treatment area at each of the 6 dates
CCmeans<-tapply(clams$LEGAL.CLAMS[clams$TREATMENT=='CC'],
clams$DATES[clams$TREATMENT=='CC'],mean)
CEmeans<-
tapply(clams$LEGAL.CLAMS[clams$TREATMENT=='CE'],clams$DATES[clams$TREATMENT==
'CE'],mean)
CCmeans
       0        1        2        3        4        5
18.85714 35.50000 31.16667 51.15000 39.91667 54.69231
CEmeans
        0         1         2         3         4         5
 20.29412  36.33333  37.16667  81.00000  53.68750 146.58065

# plotting the points
plot(clams$DATES,clams$LEGAL.CLAMS,type='n',axes=FALSE,xlab='Sample
Dates',ylab='Clams Counted')
axis(1,cex.axis=.9,labels=c('Spr 2001','Fall 2001','Spr 2002','Fall
2002','Spr 2003','Fall 2003'))
axis(2,cex.axis=.9)
box()
# highlighting the treatment areas in two different colors (CC=red, CE=blue)
points(jitter(clams$DATES[clams$TREATMENT=='CC']),
clams$LEGAL.CLAMS[clams$TREATMENT=='CC'], col=2)
points(jitter(clams$DATES[clams$TREATMENT=='CE']),
clams$LEGAL.CLAMS[clams$TREATMENT=='CE'], col=4)
# plotting the means
points(0:5,CCmeans,col=2,pch=16,cex=1.5)
points(0:5,CEmeans,col=4,pch=16,cex=1.5)
lines(seq(0,5,1),CCmeans,col=2)
lines(seq(0,5,1),CEmeans,col=4)
# adding in a line dividing dates when both treatment areas were open (SPR2001 and FALL2001) and
when one was open and the other closed
abline(v=1.5)
text(x=0.8,y=200,labels='both areas open',cex=0.8)
text(x=2.4,y=200,labels='CE closed, CC open',cex=0.8)
# adding the title and legend
mtext("Clams Counted on Each Sample Date", side=3, line=.5)
legend(2.2,350, c('CC clams', 'CE clams'), col=c(2,4), pch=c(1,1),
cex=c(.8,.8), bty='n')
legend(2,320, c('CC means', 'CE means'), col=c(2,4), lty=c(1,1),
cex=c(.8,.8), bty='n')
```



2. Choose an appropriate probability generating mechanism for these data. I don't expect you to do an exhaustive search here but you do need to do some kind of preliminary analysis to justify your choice of probability distribution.

Do mean-variance plot to get ideas for the probability generating mechanism.

```
#  getting means and vars for each category
tapply(clams$LEGAL.CLAMS,clams$DATES,
mean)->means
tapply(clams$LEGAL.CLAMS,clams$DATES,
var)->vars

#  plotting vars vs. means
plot(means, vars, axes=FALSE,
xlab=expression(paste("Mean ",mu)),
ylab=expression(paste("Variance
",sigma^2)), cex=1.5)
axis(1,cex.axis=.9)
axis(2,cex.axis=.9)
box()
```



The mean-variance plot of the count data shows heteroscedasticity, which means the probability-generating mechanism should be either a Poisson or negative binomial. I will choose a negative binomial distribution because the independence and homogeneity assumptions of the Poisson are not met in this case.

3. Using some of these observations begin by fitting a model that includes the following:

- an intercept,
- a treatment effect,
- a seasonality effect,
- a linear effect due to date (treating the sample dates as equally spaced and coded, e.g., 0, 1, 2, 3, 4, 5),
- a date by treatment interaction, and
- a season by treatment interaction.

Next use the observations listed above to try to simplify this model as much as possible. Carry out your simplifications by fitting a sequence of models in which various terms are eliminated. Stop when either significance testing or an appropriate information theoretic criterion dictates that further simplification is unwarranted. Do not use an automatic model building program for this!

```
names(clams)
[1] "TREATMENT"   "YEAR"        "SEASON"      "LEGAL.CLAMS"
[5] "DATE"        "DATES"

library(MASS)

model4a<-
glm.nb(clams$LEGAL.CLAMS~TREATMENT+SEASON+DATES+DATES:TREATMENT+SEASON:TREATM
ENT,data=clams)
summary(model4a)
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)            3.41026    0.16773  20.332  < 2e-16 ***
TREATMENTCE            0.05995    0.23485   0.255  0.79852
SEASONSPR             -0.35276    0.15084  -2.339  0.01936 *
DATES                  0.14794    0.04513   3.278  0.00105 **
TREATMENTCE:DATES      0.14738    0.05923   2.488  0.01284 *
TREATMENTCE:SEASONSPR -0.17514    0.20767  -0.843  0.39904
AIC(model1)
[1] 1757.676
```

The significant parameters in the above model were the intercept, season, dates, and the treatment:date interaction term, all with a p-value below 0.05. The non-significant parameters were treatment and the treatment:season interaction term. I will try to remove the non-significant parameters starting from the interaction term to see if the model improves (as judging from a comparison of the AICs).

```
#  Removing season:treatment
model4b<-
glm.nb(clams$LEGAL.CLAMS~TREATMENT+SEASON+DATES+DATES:TREATMENT,data=clams)
sapply(list(model4a,model4b),AIC)
[1] 1757.676 1756.370
```

Removing season:treatment lowers the AIC, but it still has a nonsignificant interaction term, so next I will remove treatment from the model.

```
summary(model4b)
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        3.47086    0.14896  23.300  < 2e-16 ***
TREATMENTCE       -0.07292    0.17692  -0.412  0.68023
SEASONSPR         -0.44016    0.10388  -4.237 2.26e-05 ***
DATES              0.13987    0.04383   3.191  0.00142 **
TREATMENTCE:DATES  0.16733    0.05471   3.059  0.00222 **

model4c<-glm.nb(clams$LEGAL.CLAMS~SEASON+DATES+DATES:TREATMENT,data=clams)
sapply(list(model4a,model4b,model4c),AIC)
[1] 1757.676 1756.370 1754.536
```

Removing the treatment term gives an even better model. So now the best model is model4c, which has season, dates and a dates:treatment interaction term. Also, all its terms are now significant, so I will call this the best model.

#The final best model

```
model4c<-glm.nb(clams$LEGAL.CLAMS~SEASON+DATES+DATES:TREATMENT,data=clams)
summary(model4c)
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        3.43302    0.11766  29.178  < 2e-16 ***
SEASONSPR         -0.44303    0.10380  -4.268 1.97e-05 ***
DATES              0.15084    0.03558   4.240 2.24e-05 ***
DATES:TREATMENTCE  0.14807    0.02956   5.009 5.48e-07 ***
```

4. Carry out an appropriate goodness of fit test for your final model. Does the model fit?

Finding expected values

```
exp.prob<-function(x) mean(dnbinom(x, mu=fitted(model4c),
size=model4c$theta))
p.actual<-sapply(0:1000,exp.prob)  #the expected probs.
sum(p.actual)
[1] 0.9999999
```

function by Jack Weiss

```
#first argument is desired minimum probability
#second argument is a vector of expected probabilities
get.breaks<-function(x,probs)
{
cum.probs<-cumsum(probs)
cur.val<-0
 index.list<-(-1)
repeat {
  cur.index<-(1:length(cum.probs))[cum.probs>=(x+cur.val)][1]
  #back off 1 because actual x-values start at 0, not 1
  index.list<-c(index.list,cur.index-1)
  cur.val<-cum.probs[cur.index]
  if (1-cur.val < x) {
    index.list<-index.list[1:(length(index.list)-1)]
    break
    }
  }
 index.list
}
```

```
get.breaks(7/dim(clams)[1],p.actual)
[1]  -1   6  10  13  16  19  22  25  28  31  35  39  43  48  53  59  66
[18]  74  84  96 112 134 170
breaks<-c(get.breaks(7/dim(clams)[1],p.actual),1000)
breaks
[1]   -1    6   10   13   16   19   22   25   28   31   35   39   43
[14]   48   53   59   66   74   84   96  112  134  170 1000
groups<-cut(0:1000,breaks) #0:999 gives 1000 numbers to be cut by breaks
groups
   [1] (-1,6]      (-1,6]      (-1,6]      (-1,6]      (-1,6]
   [6] (-1,6]      (-1,6]      (6,10]      (6,10]      (6,10]
  [11] (6,10]      (10,13]     (10,13]     (10,13]     (13,16]
  [16] (13,16]     (13,16]     (16,19]     (16,19]     (16,19]    ...
sums<-tapply(p.actual[1:1000],groups,sum)
sums
     (-1,6]      (6,10]     (10,13]     (13,16]     (16,19]     (19,22]
 0.04620752  0.05102225  0.04307270  0.04445243  0.04426541  0.04310324
    (22,25]     (25,28]     (28,32]     (32,36]     (36,40]     (40,44]
 0.04137414  0.03934959  0.04912478  0.04531683  0.04167738  0.03828249
    (44,49]     (49,54]     (54,60]     (60,67]     (67,75]     (75,84]
 0.04348713  0.03911405  0.04182794  0.04266147  0.04185372  0.03972932
    (84,95]    (95,109]   (109,127]   (127,154] (154,1e+03]
 0.03994061  0.04007553  0.03836650  0.03869264  0.06700231
sum(sums)
[1] 1

Ei<-sums*dim(clams)[1]  #the expected counts.
Ei
     (-1,6]      (6,10]     (10,13]     (13,16]     (16,19]     (19,22]
   8.548392    9.439117    7.968449    8.223700    8.189101    7.974100
    (22,25]     (25,28]     (28,32]     (32,36]     (36,40]     (40,44]
   7.654215    7.279674    9.088083    8.383613    7.710316    7.082261
    (44,49]     (49,54]     (54,60]     (60,67]     (67,75]     (75,84]
   8.045120    7.236100    7.738168    7.892371    7.742939    7.349925
    (84,95]    (95,109]   (109,127]   (127,154] (154,1e+03]
   7.389012    7.413972    7.097802    7.158139   12.395428
sum(Ei)
[1] 185
```

Finding observed values

```
obs<-table(clams$LEGAL.CLAMS)
zeros<-rep(0,1001)
data1<-data.frame(as.numeric(names(obs)),as.vector(obs))
data2<-data.frame(0:1000,zeros)
colnames(data1)<-c("key","obs")
colnames(data2)[1]<-"key"
bothfiles<-merge(data1,data2,all=TRUE)
obs.filled<-ifelse(is.na(bothfiles$obs),0,bothfiles$obs) #zeros replace NAs
Oi<-tapply(obs.filled[1:1001],groups,sum)
Oi
     (-1,6]      (6,10]     (10,13]     (13,16]     (16,19]     (19,22]
          7           4          11          12          10          10
    (22,25]     (25,28]     (28,31]     (31,35]     (35,39]     (39,43]
          6           6          10          11           5           6
    (43,48]     (48,53]     (53,59]     (59,66]     (66,74]     (74,84]
         14          10           5           8           4           7
    (84,96]    (96,112]   (112,135]   (135,172] (172,1e+03]
```

```
         5          8          6          9         11
sum(Oi)
[1] 185
```

model4c<-glm.nb(clams$LEGAL.CLAMS~SEASON+DATES+DATES:TREATMENT,data=clams)
The model has 4 parameters: season, dates, dates:treatment, and the dispersion term.

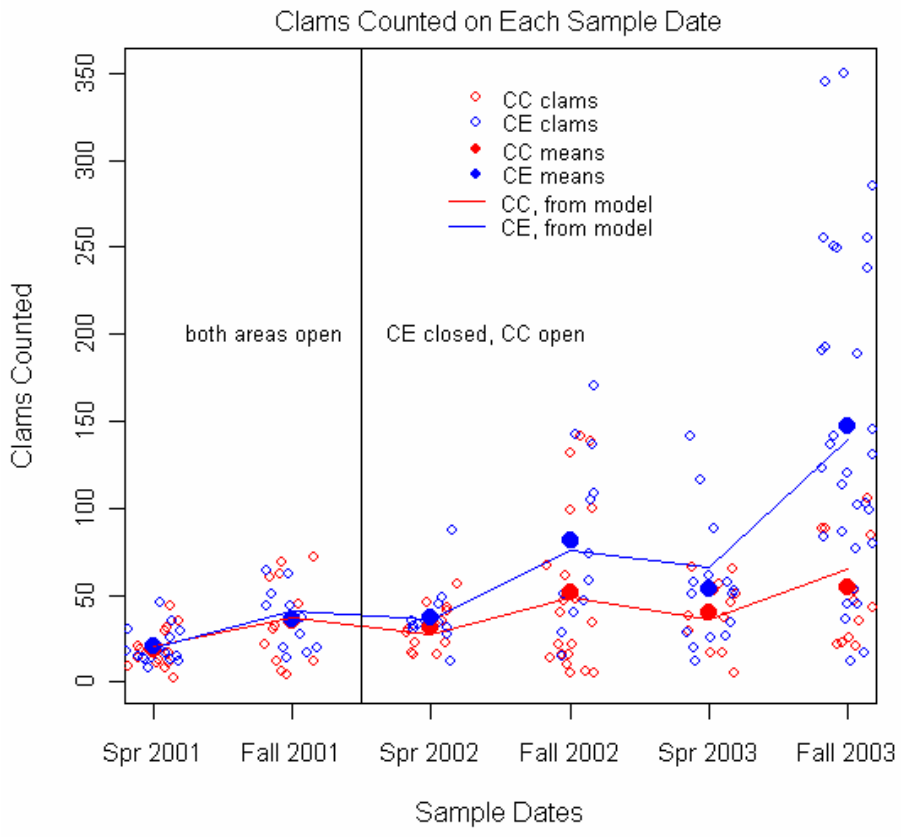
```
pearson<-sum((Oi-Ei)^2/Ei)
pearson
[1] 19.70515
df<-length(Oi)-1-4 #(n-1-p)
df
[1] 18
p.val<-1-pchisq(pearson,df)
p.val
[1] 0.3496618
qchisq(.95,df)
[1] 28.8693
```

The p-value is above 0.05, which gives no evidence that the model doesn't fit (at least with this particular grouping). Also, the observed value of the test statistic was 18.36, which is lower than the critical 5% value given by qchisq (28.87), which means we shouldn't reject this model.

5. Redo your plot from Question 1 except do not connect the treatment means by line segments. Instead superimpose your final model from Question 3 on the plot as two lines representing the predictions for the two treatments. Based on your plot how well does it appear that the model predicts the observed means?

```
# plotting the points
plot(clams$DATES,clams$LEGAL.CLAMS,type='n',axes=FALSE,xlab='Sample
Dates',ylab='Clams Counted')
axis(1,cex.axis=.9,labels=c('Spr 2001','Fall 2001','Spr 2002','Fall
2002','Spr 2003','Fall 2003'))
axis(2,cex.axis=.9)
box()
# highlighting the treatment areas in two different colors (CC=red, CE=blue)
points(jitter(clams$DATES[clams$TREATMENT=='CC']),
clams$LEGAL.CLAMS[clams$TREATMENT=='CC'], col=2)
points(jitter(clams$DATES[clams$TREATMENT=='CE']),
clams$LEGAL.CLAMS[clams$TREATMENT=='CE'], col=4)
# plotting the points for the means
points(0:5,CCmeans,col=2,pch=16,cex=1.5)
points(0:5,CEmeans,col=4,pch=16,cex=1.5)

# plotting two lines, one for each treatment

coef(model3)
      (Intercept)       TREATMENTCE         SEASONSPR             DATES
       3.47086208       -0.07291756       -0.44016001        0.13986742
TREATMENTCE:DATES
       0.16732992

trt.fxn<-function(x,y,z) {
exp(coef(model3)[1] +coef(model3)[2]*x +coef(model3)[3]*y +coef(model3)[4]*z
+coef(model3)[5]*x*z)   }
```

```
lines(seq(0,5,1),trt.fxn(0,c(1,0,1,0,1,0),seq(0,5,1)),col=2) #CC
lines(seq(0,5,1),trt.fxn(1,c(1,0,1,0,1,0),seq(0,5,1)),col=4) #CE

#  adding in a line dividing dates when both treatment areas were open (SPR2001 and FALL2001) and when one was open and the other closed
abline(v=1.5)
text(x=0.8,y=200,labels='both areas open',cex=0.8)
text(x=2.4,y=200,labels='CE closed, CC open',cex=0.8)
#  adding the title and legend
mtext("Clams Counted on Each Sample Date", side=3, line=.5)
legend(2.2,350, c('CC clams', 'CE clams'), col=c(2,4), pch=c(1,1),
cex=c(.8,.8), bty='n')
legend(2.2,320, c('CC means', 'CE means'), col=c(2,4), pch=c(16,16),
cex=c(.8,.8), bty='n')
legend(2,290, c('CC, from model', 'CE, from model'), col=c(2,4), lty=c(1,1),
cex=c(.8,.8), bty='n')
```



The model predicts the observed means very well. It does a better job for earlier time periods than for later ones, perhaps because of the greater variability in the data then.

6. What are your conclusions? Does the rotation plan appear to work?

Based on the above graph, the rotation plan does seem to work. After closing off the experimental area (CE, blue) before spring 2002, the CE area begins to have more clams than the control area (CC, red). This is the case for both the observed means (filled-in points) as well as the model predictions, which fall very close to the observed means.