HW3- Univariate Models

Sarah Wiegreffe

February 3, 2016

1. Carry out an exploratory analysis using the tree dataset. Develop and compare models for species cover for a habitat generalist Acer rubrum (Red maple) and a habitat specialist Abies fraseri (Frasier fir). Because this dataset includes both continuous and discrete explanatory variables use the function `Anova` in the packages `car`.

This will estimate partial effect sizes, variance explained, and p-values for each explanatory variable included in the model.

Compare the p-values you observe using the function `Anova` to those generated using `summary`.

For each species address the following additional questions:

```
* how well does the exploratory model appear to explain cover?
* which explanatory variables are the most important?
* do model diagnostics indicate any problems with violations of
  OLS assumptions?
* are you able to explain variance in one species better than another?
```

```r
#install.packages("car")
library(car)
```

```
## Warning: package 'car' was built under R version 3.1.3
```

```r
trees = read.csv('./quant-methods-course-page/quant_methods/data/treedata_subset.csv')
#colnames(trees)

#Subset dataset, removing useless columns plotID (unique for each), species, and spcode
acer = trees[trees$species == 'Acer rubrum', -c(1,2,3)]
abies = trees[trees$species == 'Abies fraseri', -c(1,2,3)]

#We have the following possibly significant independent variables:
#elev, tci, streamdist, disturb, and beers. We start with them all.
mod_acer = lm(cover ~ elev + beers + tci + streamdist + disturb, data = acer)
mod_abies = lm(cover ~ elev + beers + tci + streamdist + disturb, data = abies)

#Now analyzing the model:
Anova(mod_acer, type=3)
```

```
## Anova Table (Type III tests)
##
## Response: cover
##             Sum Sq  Df  F value    Pr(>F)
## (Intercept) 765.43   1 193.5096 < 2.2e-16 ***
## elev         40.44   1  10.2233  0.001448 **
## beers        35.61   1   9.0034  0.002789 **
## tci          12.58   1   3.1805  0.074947 .
## streamdist    29.09   1   7.3531  0.006856 **
## disturb        9.45   3   0.7962  0.496166
## Residuals   2828.21 715
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod_acer)
```

```
##
## Call:
## lm(formula = cover ~ elev + beers + tci + streamdist + disturb,
##     data = acer)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7073 -1.2446  0.3409  1.3575  5.2732
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    6.3502303  0.4564973  13.911  < 2e-16 ***
## elev          -0.0010108  0.0003161  -3.197  0.00145 **
## beers         -0.3269597  0.1089662  -3.001  0.00279 **
## tci           -0.0627613  0.0351922  -1.783  0.07495 .
## streamdist     0.0012895  0.0004756   2.712  0.00686 **
## disturbLT-SEL  0.0829610  0.2166747   0.383  0.70192
## disturbSETTLE -0.1044556  0.2804213  -0.372  0.70963
## disturbVIRGIN  0.3088364  0.2518161   1.226  0.22044
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.989 on 715 degrees of freedom
## Multiple R-squared:  0.04493,    Adjusted R-squared:  0.03558
## F-statistic: 4.805 on 7 and 715 DF,  p-value: 2.669e-05
```

```
#Same procedure for mod_abies
Anova(mod_abies, type=3)
```

```
## Anova Table (Type III tests)
##
## Response: cover
##             Sum Sq Df F value    Pr(>F)
## (Intercept) 59.401  1 23.1710 2.652e-05 ***
## elev        61.618  1 24.0358 2.022e-05 ***
## beers        0.014  1  0.0056    0.9406
## tci          5.667  1  2.2105    0.1458
## streamdist   1.636  1  0.6382    0.4296
## disturb     10.089  3  1.3118    0.2855
## Residuals   92.289 36
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(mod_abies)
```

```
##
## Call:
## lm(formula = cover ~ elev + beers + tci + streamdist + disturb,
##     data = abies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4630 -0.6472  0.0788  1.0872  3.8017
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -20.561173   4.271449  -4.814 2.65e-05 ***
## elev             0.012370   0.002523   4.903 2.02e-05 ***
## beers            0.037551   0.500269   0.075   0.9406
## tci              0.287641   0.193467   1.487   0.1458
## streamdist      -0.001266   0.001585  -0.799   0.4296
## disturbLT-SEL    2.188367   2.097905   1.043   0.3038
## disturbSETTLE    1.527604   2.341471   0.652   0.5183
## disturbVIRGIN    3.025596   1.735921   1.743   0.0899 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.601 on 36 degrees of freedom
## Multiple R-squared:  0.5824, Adjusted R-squared:  0.5011
## F-statistic: 7.171 on 7 and 36 DF,  p-value: 2.215e-05
```

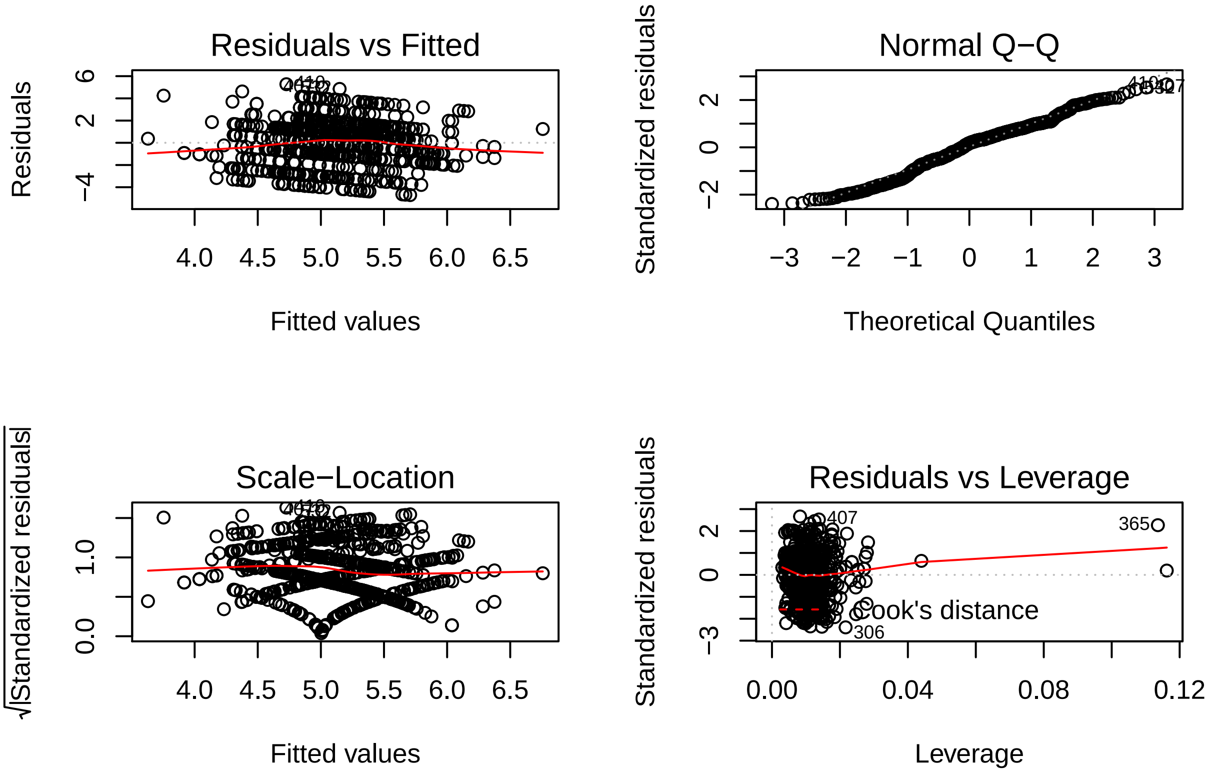
```r
#Compare two models
AIC(mod_abies)
```

```
## [1] 175.4592
```

```r
AIC(mod_acer)
```

```
## [1] 3055.95
```

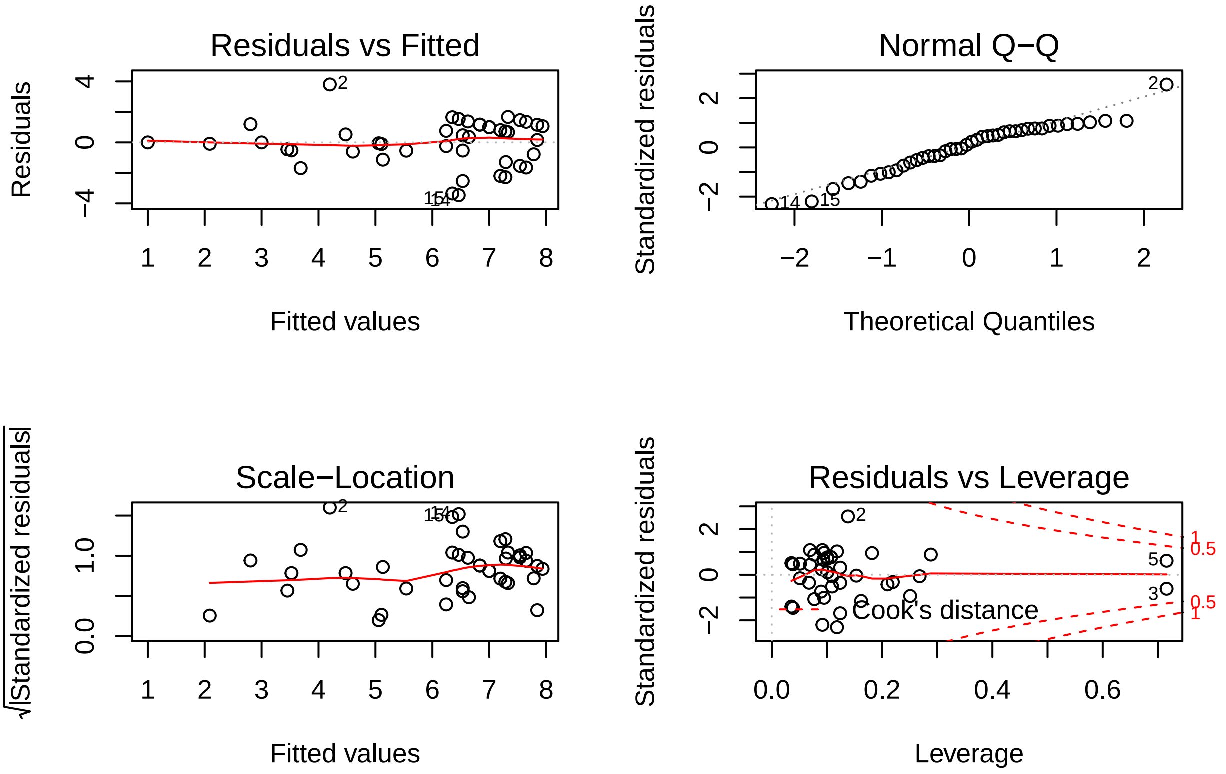
```r
#Model Diagnostics via plotting
par(mfrow=c(2,2))
plot(mod_acer)
```



```r
par(mfrow=c(2,2))
plot(mod_abies)
```

```
## Warning: not plotting observations with leverage one:
##   1, 4
```

```
## Warning: not plotting observations with leverage one:
##   1, 4
```



The main difference between the Anova() and summary() model summaries is that summary() considers each level in a discrete variable (in this case, disturb) as its own independent variable, whereas Anova() provides a p-value for the variable as a whole, which is much more useful for analysis. The summary function also provides less precision in p-values.

For both models, while there don't appear to be any significant outliers in the data, the residuals do not appear to be evenly dispersed about 0, but rather show a sloping trend which is disconcerting. This means that the OLS assuption of homoskedastic residuals appears to be violated for both models. Apart from this, the abies model appears to explain cover better than the acer model because it has smaller sum of squares value for the residuals. The most important explanatory variables for the abies model are elev and (somewhat) tci (can be seen after further updating the model). For acer, they are elev, beers, streamdist, and (somewhat) tci. It is much easier to explain variance in abies than acer model because the sum squared of the residuals is so much lower and it also has a much lower AIC.

2. You may have noticed that the variable cover is defined as positive integers between 1 and 10. and is therefore better treated as a discrete rather than continuous variable. Re-examine your solutions to the question above but from the perspective of a General Linear Model (GLM) with a Poisson error term (rather than a Gaussian one as in OLS). The Poisson distribution generates integers 0 to positive infinity so this may provide a good first approximation.

For assessing the degree of variation explained you can use a pseudo-R-squared statistic (note this is just one of many possible).

Compare the residual sums of squares between the traditional OLS and glm models using `anova` (Note: not `Anova`) as such.

Does it appear that changing the error distribution changed the results much? In what ways?

```r
acer_glm = glm(cover ~ ., data= acer, family='poisson')
acer_ols = glm(cover ~ ., data = acer, family='gaussian')
```

```
abies_glm = glm(cover ~ ., data= abies, family='poisson')
abies_ols = glm(cover ~ ., data = abies, family='gaussian')

pseudo_r2 = function(glm_mod) {
    1 -  glm_mod$deviance / glm_mod$null.deviance
}

pseudo_r2(acer_glm)
```

```
## [1] 0.03997917
```

```
pseudo_r2(abies_glm)
```

```
## [1] 0.60931
```

```
anova(acer_ols, acer_glm)
```

```
## Analysis of Deviance Table
##
## Model 1: cover ~ elev + tci + streamdist + disturb + beers
## Model 2: cover ~ elev + tci + streamdist + disturb + beers
##   Resid. Df Resid. Dev Df Deviance
## 1       715    2828.21
## 2       715     623.38  0   2204.8
```

```
anova(abies_ols, abies_glm)
```

```
## Analysis of Deviance Table
##
## Model 1: cover ~ elev + tci + streamdist + disturb + beers
## Model 2: cover ~ elev + tci + streamdist + disturb + beers
##   Resid. Df Resid. Dev Df Deviance
## 1        36     92.289
## 2        36     16.126  0   76.164
```

```
Anova(abies_glm)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: cover
##            LR Chisq Df Pr(>Chisq)
## elev        11.3450  1  0.0007565 ***
## tci          1.1830  1  0.2767545
## streamdist   0.3059  1  0.5802166
## disturb      3.3953  3  0.3346007
## beers        0.0155  1  0.9008297
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
Anova(abies_ols)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: cover
##            LR Chisq Df Pr(>Chisq)
## elev        24.0358  1  9.456e-07 ***
## tci          2.2105  1     0.1371
## streamdist   0.6382  1     0.4244
## disturb      3.9355  3     0.2685
## beers        0.0056  1     0.9402
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
Anova(acer_glm)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: cover
##            LR Chisq Df Pr(>Chisq)
## elev         7.7744  1   0.005299 **
## tci          2.5877  1   0.107699
## streamdist   5.4866  1   0.019163 *
## disturb      1.9033  3   0.592714
## beers        6.9611  1   0.008330 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
Anova(acer_ols)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: cover
##            LR Chisq Df Pr(>Chisq)
## elev        10.2233  1   0.001387 **
## tci          3.1805  1   0.074523 .
## streamdist   7.3531  1   0.006695 **
## disturb      2.3887  3   0.495733
## beers        9.0034  1   0.002695 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow=c(2,2))
plot(acer_glm)
par(mfrow=c(2,2))
plot(abies_glm)
```

By using a GLM model with a Poisson error term, the residual sum of squares for both the abies and acer models was reduced drastically, meaning that variance is much lower once cover has been considered as a discrete variable instead of continuous. By looking at the pseudo R^2 values from the glm's, we can see clearly that the abies exploratory model does a MUCH better job at explaining cover than the acer one. However, the plots still show some abnormal trends of residuals. Changing the error distribution did not appear to change the results (e.g. which variables are significant in the model) much at all (see Anova() output), but greatly improved the error and variance in the model.

3. Provide a plain English summary (i.e., no statistics) of what you have found and what conclusions we can take away from your analysis?

The abies model had fewer significant variables, and also relatedly much less variance (less sum squared of errors) than the acer model. We cannot assume OLS due to the heterskedasticity of the residuals for both models, so a generalized linear model with a Poisson error term was used and greatly reduced variance due to the fact that cover is a discrete variable. Some variables, such as disturb, were not at all significant in the models. Overall, the abies model performed much better than the acer one.

4. (optional) Examine the behavior of the function `step()` using the exploratory models developed above. This is a very simple and not very robust machine learning stepwise algorithm that uses AIC to select a best model. By default it does a backward selection routine.

```
new_mod_acer = step(mod_acer)
```

```
## Start:  AIC=1002.17
## cover ~ elev + beers + tci + streamdist + disturb
##
##              Df Sum of Sq    RSS     AIC
## - disturb     3     9.449 2837.7  998.58
## <none>                    2828.2 1002.17
## - tci         1    12.581 2840.8 1003.37
## - streamdist  1    29.085 2857.3 1007.56
## - beers       1    35.613 2863.8 1009.21
## - elev        1    40.439 2868.7 1010.43
##
## Step:  AIC=998.58
## cover ~ elev + beers + tci + streamdist
##
##              Df Sum of Sq    RSS     AIC
## <none>                    2837.7  998.58
## - tci         1    14.370 2852.0 1000.23
## - streamdist  1    31.491 2869.2 1004.56
## - beers       1    35.515 2873.2 1005.57
## - elev        1    45.778 2883.4 1008.15
```

```
AIC(mod_acer)
```

```
## [1] 3055.95
```

```
AIC(new_mod_acer)
```

```
## [1] 3052.362
```

```
anova(mod_acer, new_mod_acer)
```

```
## Analysis of Variance Table
##
## Model 1: cover ~ elev + beers + tci + streamdist + disturb
## Model 2: cover ~ elev + beers + tci + streamdist
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    715 2828.2
## 2    718 2837.7 -3   -9.4488 0.7962 0.4962
```

```
new_mod_abies = step(mod_abies)
```

```
## Start:  AIC=48.59
## cover ~ elev + beers + tci + streamdist + disturb
##
##              Df Sum of Sq     RSS    AIC
## - beers       1     0.014  92.304 46.599
## - disturb     3    10.089 102.379 47.157
## - streamdist  1     1.636  93.926 47.366
## <none>                     92.289 48.593
## - tci         1     5.667  97.956 49.215
## - elev        1    61.618 153.908 69.095
##
## Step:  AIC=46.6
## cover ~ elev + tci + streamdist + disturb
##
##              Df Sum of Sq     RSS    AIC
## - streamdist  1     1.665  93.969 45.386
## - disturb     3    10.679 102.983 45.417
## <none>                     92.304 46.599
## - tci         1     6.745  99.049 47.703
## - elev        1    64.662 156.966 67.961
##
## Step:  AIC=45.39
## cover ~ elev + tci + disturb
##
##           Df Sum of Sq     RSS    AIC
## - disturb  3    12.021 105.990 44.683
## <none>                  93.969 45.386
## - tci      1     6.807 100.776 46.463
## - elev     1    78.687 172.656 70.153
##
## Step:  AIC=44.68
## cover ~ elev + tci
##
##        Df Sum of Sq    RSS    AIC
## <none>               105.99 44.683
## - tci   1     9.239 115.23 46.360
## - elev  1   114.046 220.04 74.822
```

```
AIC(mod_abies)
```

```
## [1] 175.4592
```

```
AIC(new_mod_abies)
```

```
## [1] 171.5494
```

```
anova(mod_abies, new_mod_abies)
```

```
## Analysis of Variance Table
##
## Model 1: cover ~ elev + beers + tci + streamdist + disturb
## Model 2: cover ~ elev + tci
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1     36  92.289
## 2     41 105.990 -5     -13.7 1.0688 0.3937
```

The step() function reduces the AIC of the models and improves them overall by removing variables that are not significant, simplifying the parameters to find the minimum adequate model.