Homework 6

Biology 697

10/9/2012

For this weeks' homework, we'll be working with a data set looking at mice in cold seasonally variable habitats. The data contains four columns: Julian day, temperature (celsius), activity (number of foraging bouts in 6 hours), and food (g food found over 6 hours). We'll investigate this data set and the patterns of correlation in the data in this week's homework.

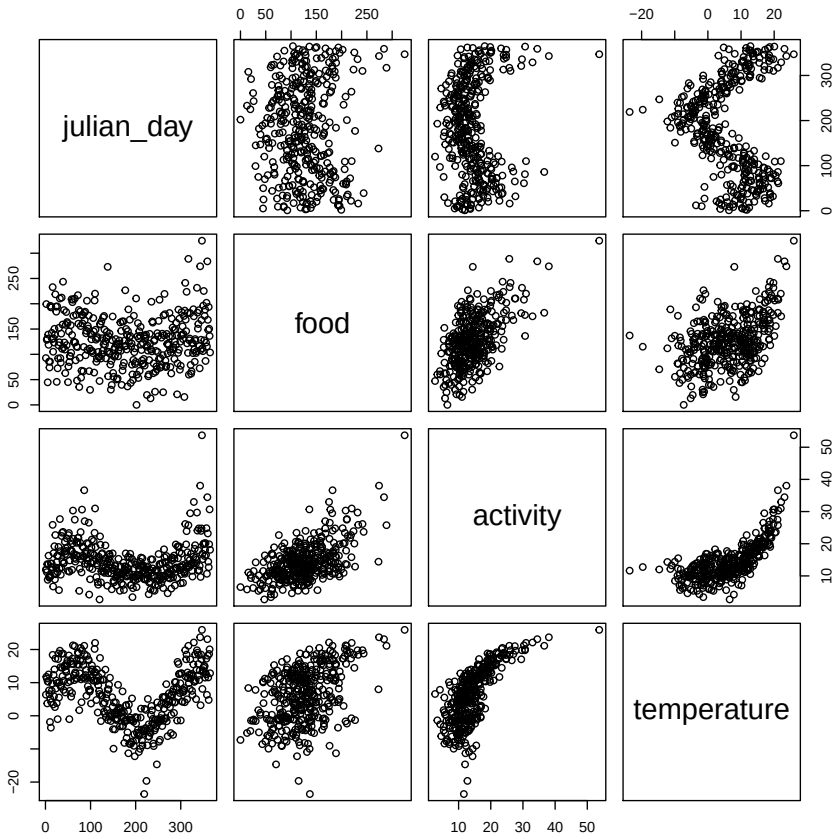
1 Correlation

We talked about several correlation metrics this week. Pearson's Correlation is when data are drawn from two normal distributions and share a linear relationship. Spearman's correlation is a non-parametric technique when normality is violated that uses ranks. The Distance Correlation is defined by comparing the pattern of pairwise distances between all values of X and the pattern of pairwise distances between all values of Y. `http://www.imstat.org/aoas/AOAS34INTRO.pdf` provides a lovely intro to it.

1.1

Which `pairs` of variables can we use Pearson's correlation? Which will require non-parametric tests, and if so, which ones?

```
activityDF <- read.csv("./seasonal_mouse_activity.csv")
pairs(activityDF)
```




```
# Activity v. Food is Spearman
# Temp v. Food or Activity is Spearman
# Julian Day  v. Anything is DCOR
```

1.2 Spearman

Spearman's correlation works by calculating correlation based on rank rather than observed value. Write a function to calculate and test Spearman's correlation, and run it on the relationship between temperature and foraging activity. You can compare it to results from `cor.test(method="spearman")`

```
spearman<-function(x,y){
  sp <- cor(rank(x), rank(y))
  se <- sqrt((1-sp^2)/length(x[-(1:2)]))
```

```
  t<-sp/se
  p<- pt(abs(t), df=length(x[-(1:2)]), lower.tail=F)*2
  return(list(spearman=sp, se=se, t=t, p=p))
}

with(activityDF, spearman(activity, temperature))

# $spearman
# [1] 0.7328
#
# $se
# [1] 0.03572
#
# $t
# [1] 20.52
#
# $p
# [1] 1.166e-62


with(activityDF, cor.test(activity, temperature, method="spearman"))

#
#  Spearman's rank correlation rho
#
# data:  activity and temperature
# S = 2165756, p-value < 2.2e-16
# alternative hypothesis: true rho is not equal to 0
# sample estimates:
#    rho
# 0.7328
```

1.3 Comparing Parametric v. Nonparametric Techniques

How do results differ between the three correlation techniques for the relationship between Julian day and temperature? What about activity and food? Use `dcor` from the energy package and `dcov.test` to evaluate distance based correlation

```
library(energy)

# Loading required package:  boot
#
# Attaching package:  'boot'
```

```
# The following object(s) are masked from 'package:lattice':
#
# melanoma
# The following object(s) are masked from 'package:car':
#
# logit


with(activityDF, {
  print(cor.test(julian_day, temperature))
  print(cor.test(julian_day, temperature, method="spearman"))
  print(dcov.test(julian_day, temperature))
  print(dcor(julian_day, temperature))

})

#
#  Pearson's product-moment correlation
#
# data:  julian_day and temperature
# t = -2.086, df = 363, p-value = 0.03765
# alternative hypothesis: true correlation is not equal to 0
# 95 percent confidence interval:
#  -0.209163 -0.006268
# sample estimates:
#     cor
# -0.1088
#
#
#  Spearman's rank correlation rho
#
# data:  julian_day and temperature
# S = 9027926, p-value = 0.02956
# alternative hypothesis: true rho is not equal to 0
# sample estimates:
#     rho
# -0.1139
#
#
#  dCov test of independence
#
# data:  index 1, replicates 199
# nV^2 = 25177, p-value = 0.755
# sample estimates:
#  dCov
# 8.305
```

```
#
# [1] 0.3973

## Pearson and Spearman's correlations have small
## negative values while dcor has a larger positive value.
## Intriguingly, all are different from 0.


with(activityDF, {
  print(cor.test(food, activity))
  print(cor.test(food, activity, method="spearman"))
  print(dcov.test(food, activity))
  print(dcor(food, activity))
})

#
#  Pearson's product-moment correlation
#
# data:  food and activity
# t = 13.62, df = 363, p-value < 2.2e-16
# alternative hypothesis: true correlation is not equal to 0
# 95 percent confidence interval:
#  0.5093 0.6456
# sample estimates:
#    cor
# 0.5815
#
#
#  Spearman's rank correlation rho
#
# data:  food and activity
# S = 4150064, p-value < 2.2e-16
# alternative hypothesis: true rho is not equal to 0
# sample estimates:
#    rho
# 0.4879
#
#
#  dCov test of independence
#
# data:  index 1, replicates 199
# nV^2 = 9062, p-value = 0.005
# sample estimates:
#  dCov
# 4.983
#
```

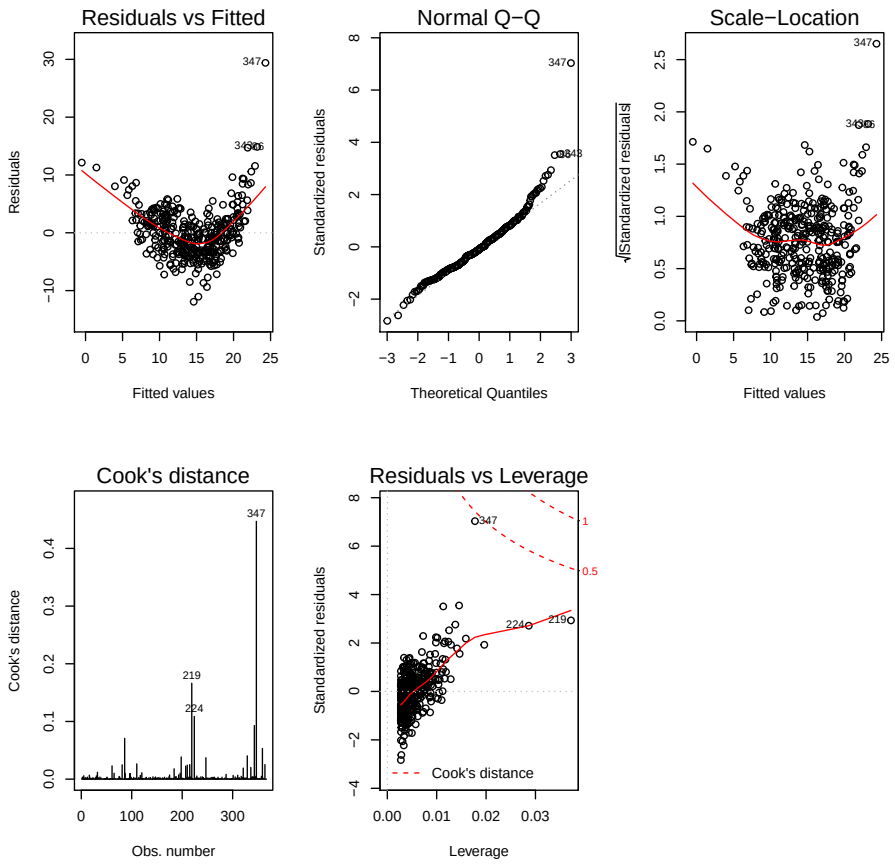
```
# [1] 0.4915


# While all are positive and different from 0
# Pearson's correlation is 0.58 while the other two are ~0.48.
```

2 Regression

2.1 Diagnostics

Aside from eyeballing the relationships, show why we cannot evaluate the relationship between temperature and activity using a ordinary least squares linear regression.

```
badlm<-lm(activity ~ temperature, data=activityDF)

par(mfrow=c(2,3))
plot(badlm, which=1:5)
```

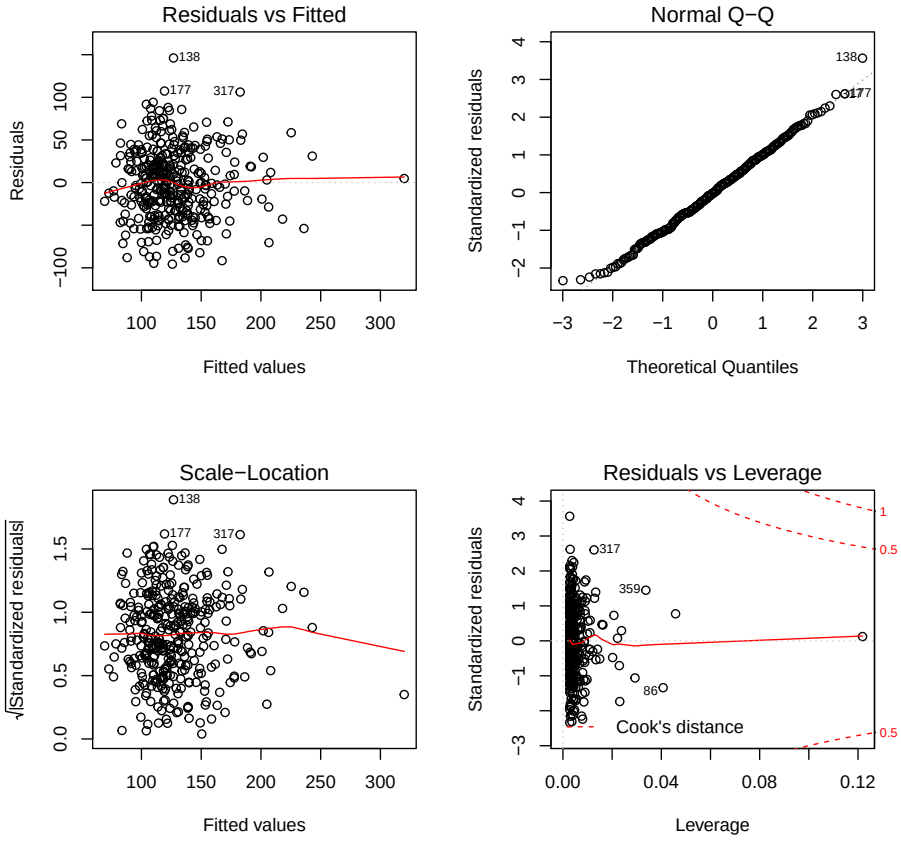



```
par(mfrow=c(1,1))

# BIG pattern in the fitted v. residual relationship.
# Also patterns in leverage v. std. residuals, and some big deviations in the QQ plot
```

2.2 Fit

Evaluate and discuss the relationship between foraging and food found. If it can indeed be evaluated.

```
goodLM <- lm(food ~ activity, data=activityDF)

#diagnostics
par(mfrow=c(2,2))
plot(goodLM)
```




```
par(mfrow=c(1,1))

#Diagnostics are all a-ok


#see the output
summary(goodLM)

#
# Call:
# lm(formula = food ~ activity, data = activityDF)
#
# Residuals:
#    Min     1Q Median     3Q    Max
# -95.73 -27.04  -0.66  27.77 146.15
#
```

```
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)   56.024      5.636    9.94   <2e-16 ***
# activity       4.914      0.361   13.62   <2e-16 ***
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# Residual standard error: 41.1 on 363 degrees of freedom
# Multiple R-squared: 0.338,Adjusted R-squared: 0.336
# F-statistic:  185 on 1 and 363 DF,  p-value: <2e-16


# Activity positively appears to influence food encountered.
# However, the relationship only explains 33% of the variation in the data.
# So, more is going on here...
```

3 The Effect of Range

One of the major issues with regression is that estimation and hypotheses testing can be influenced by the spread of your data. Take a look at the relationship between activity and food acquisituion.

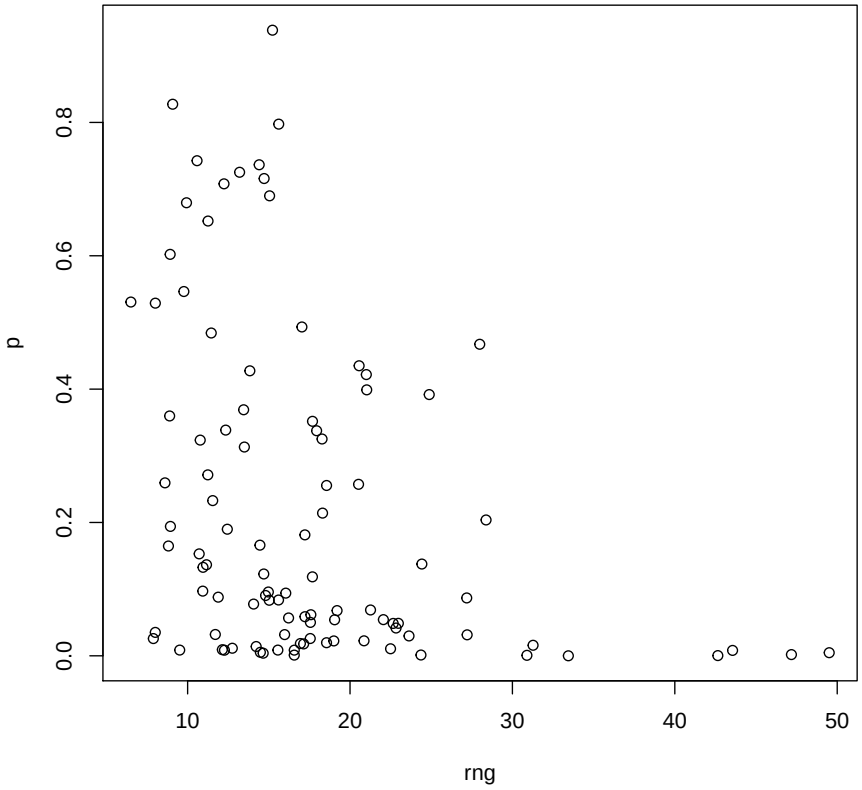
3.1

Fit and evaluate the relationship using only activity values between 3 and 9. How does it compare to the fit of the full model? If you answere that the relationship went away, you'd be right - even with 46 data points. Range can matter a great deal - particularly with small sample sizes or high variation. Explore this for yourself. Generate 100 simulations where you draw out a subset of 10 random rows from the data. Calculate the range of the observed values of activity in each simulation. Also get the p-value for the slope of activity's influence on food. (n.b. summary(aLM) produces a list. One item in that list is called coef, which is a matrix from which you can pluck a p-value). Is there some critical range past which the p-value seem to settle down.?

```
smallLM <- lm(food ~ activity, data=activityDF[which(activityDF$activity > 3
                                                     & activityDF$activity < 9),])
summary(smallLM)

#
# Call:
# lm(formula = food ~ activity, data = activityDF[which(activityDF$activity >
#     3 & activityDF$activity < 9), ])
```

```
#
# Residuals:
#    Min     1Q Median     3Q    Max
# -82.75 -21.44  -2.14  33.95  75.37
#
# Coefficients:
#             Estimate Std. Error t value Pr(>|t|)
# (Intercept)    42.66      27.96    1.53     0.13
# activity        6.15       3.88    1.58     0.12
#
# Residual standard error: 37.9 on 44 degrees of freedom
# Multiple R-squared: 0.054,Adjusted R-squared: 0.0325
# F-statistic: 2.51 on 1 and 44 DF,  p-value: 0.12

#no relationship!
```

```
set.seed(9000)
n.sims<-100
p<-rep(NA, n.sims)
rng <- rep(NA, n.sims)
sizeDraw <- 10
for (i in 1:100) {
  smallDF <- activityDF[sample(1:nrow(activityDF), size=sizeDraw),]
  rng[i] <- max(smallDF$activity) - min(smallDF$activity)
  alm <- lm(food ~ activity, data=smallDF)
  p[i] <- summary(alm)$coef[2,4]
}

plot(p ~ rng)
```




```
#looks like about 30
```

4 Extra Credit

Hubway, the Boston based bike rental company, is releasing all of their trip data. The data set is huge - about 60 MB. They're also providing lat and long information for all stations. They are hosting a data visualization challenge at `http://hubwaydatachallenge.org`. For your extra credit, find and visualize something interesting in the data. Note, ggplot and it's map geom might come in handy (or not). If you also want to play with breaking down and analyzing data using different groupings, you may want to look into the plyr library at `http://plyr.had.co.nz/` and available on CRAN. We'll be using plyr later in the course, but, it might be useful for exploring the data.

Extra points for each interesting or surprising thing you find. And, heck, if you get into this, enter the challenge!