Galapagos Islands Data

The Galapagos Islands off the coast of Ecuador provide an excellent laboratory for studying the factors that influence the development and survival of different life species. Johnson and Raven (1973; Science p893-5) have presented the data below giving the number of species and related variables for 30 different islands. Counts are given both for the total number of species and the number of species that occur only on that one island (the endemics). The variables in the data set are: island name, number of species, endemics, area in km**2, elevation in meters, distance from nearest island, distance from Santa Cruz (which is near the center of the Galapagos), and area of adjacent island in km**2. In the output below data from selected islands are given.

We will fit a Poisson model, with the number of species as response. Our analysis is fairly similar to Faraway's analysis in his book. The point of the analysis is to mostly illustrate techniques. We will see that there are some real flaws with the model. We use Faraway's version of the data, which estimated a few missing values.

```
> ga = read.table("D:/My Documents/GLMcourse/glmSECTION/gala.txt",header=T)

> ga
             Species  Endemics   Area   Elevation  Nearest  Scruz  Adjacent
Baltra           58        23     25.09     346       0.6     0.6     1.84
Bartolome        31        21      1.24     109       0.6    26.3   572.33
Caldwell          3         3      0.21     114       2.8    58.7     0.78
Espanola         97        26     58.27     198       1.1    88.3     0.57
Isabela         347        89   4669.32    1707       0.7    28.1   634.49
Pinta           104        37     59.56     777      29.1   119.6   129.49
SantaCruz       444        95    903.82     864       0.6     0.0     0.52
SantaFe          62        28     24.08     259      16.5    16.5     0.52
Wolf             21        12      2.85     253      34.1   254.7     2.33
```

I omitted the second column (endemics) since it was not to be used in the analysis.

```
> ga <- ga[,-2]
```

I first fit a Poisson regression with a log link (default) using all predictors (the period implies everything in the data set except the response)

```
> modp <- glm(Species ~ ., family=poisson, ga)
> summary(modp)

Call:
glm(formula = Species ~ ., family = poisson, data = ga)
```

```
Deviance Residuals:
    Min       1Q    Median      3Q      Max
-8.2752  -4.4966  -0.9443   1.9168  10.1849

Coefficients:
             Estimate   Std. Error  z value   Pr(>|z|)
(Intercept)  3.155e+00  5.175e-02   60.963    < 2e-16
Area        -5.799e-04  2.627e-05 -22.074    < 2e-16
Elevation    3.541e-03  8.741e-05  40.507    < 2e-16
Nearest      8.826e-03  1.821e-03    4.846   1.26e-06
Scruz       -5.709e-03  6.256e-04   -9.126    < 2e-16
Adjacent    -6.630e-04  2.933e-05 -22.608    < 2e-16

(Dispersion parameter for poisson family taken to be 1)

    Null deviance:  3510.73  on 29  degrees of freedom
Residual deviance:  716.85  on 24  degrees of freedom
AIC: 889.68

Number of Fisher Scoring iterations: 5
```

Each effect is highly significant. I can get residuals from the model using the residuals command.

```
> rp     <- residuals(modp,type="pearson")
> rd     <- residuals(modp,type="deviance")
> rraw <-residuals(modp,type="response")            These are raw resids

> cbind(rraw,rp,rd)

                        rraw               rp             rd
Baltra            -20.7243375   -2.3357489   -2.4514468
Bartolome          10.5970710    2.3460625    2.1773837
Caldwell          -22.7224758   -4.4802192   -5.7054741
Champion            3.5946726    0.7769596    0.7566071
Coamano           -15.0144771   -3.6399960   -4.6330643
Daphne.Major      -18.5854494   -3.0726909   -3.4112843
Daphne.Minor       -8.0806055   -1.4266670   -1.4938927
Darwin             -0.9185636   -0.2779883   -0.2820294
Eden              -21.8300927   -3.9969460   -4.7542578
Enderby           -24.7763009   -4.7880776   -6.2590033
Espanola           69.1386497   13.0984472   10.1848901
Fernandina          5.2558464    0.5610913    0.5556252
Gardner1           42.0813319   10.5471741    8.1129276
Gardner2          -33.2216722   -5.3736162   -6.7899703
Genovesa           14.9744202    2.9933530    2.7512972
```

```
Isabela        -23.8210549    -1.2370259    -1.2506377
Marchena        -4.7584774    -0.6372540    -0.6466564
Onslow         -18.2937413    -4.0608923    -5.2267410
Pinta         -108.6035940    -7.4483298    -8.2752152
Pinzon         -13.4269526    -1.2184842    -1.2420429
Las.Plazas     -20.1959932    -3.5592976    -4.0872409
Rabida          16.9401773     2.3256002     2.2158922
SanCristobal    61.1786806     4.1357596     3.9625414
SanSalvado    -133.9783421    -6.9560120    -7.4541992
SantaCruz      146.7220046     8.5096926     7.9235897
SantaFe          1.1185583     0.1433561     0.1429205
SantaMaria     126.8346797    10.0851505     9.0539609
Seymour          7.1056825     1.1698389     1.1350150
Tortuga        -19.5216412    -3.2754413    -3.6771692
Wolf             2.9319966     0.6897765     0.6722795
```

Note sum of squared deviance residuals is the Deviance!

```
> sum(rd^2)

716.8458
```

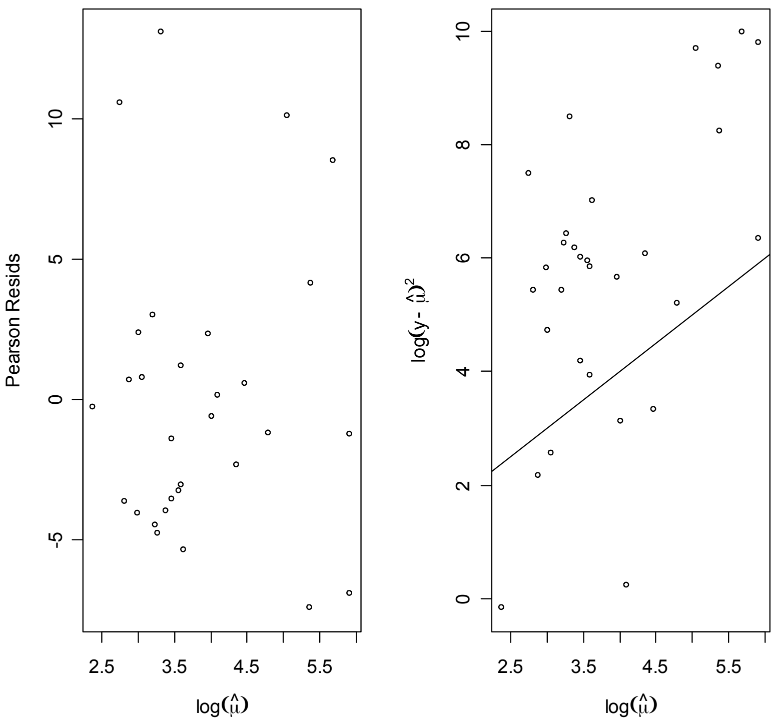
With count data, we might consider the presence of overdispersion. I saved the fitted values, then plotted the fitted values on a log scale (which is the linear predictor) against the Pearson residuals, and plotted the fitted values against the squared raw residuals, both on a log scale. Note that the form of the plot labels provides a nice touch.

If the mean structure is specified correctly, the first plot should show no dependence of the size or sign of the residual on the value of the linear predictor. If the response is Poisson, then the squared raw residual should be on average equal to the mean (i.e. expected squared raw residual estimates the variance, which equals the mean for a Poisson count). Why the log scale? Some residuals and fitted values are large, so this helps visualize the data.

```
> fv <- fitted(modp)
> par(mfrow=c(1,2))                    What does this do?
> plot( log(fv), rp, xlab=expression(log(hat(mu))), ylab="Pearson Resids")
> plot(log(fv), log(rraw^2), xlab=expression(log(hat(mu))), ylab=expression(log (y-
hat(mu))^2 ))

> abline(0,1)
```

Looking at the plots it would seem the primary problem lies with the potential for overdispersion. This is also clear from an estimate of the overdispersion parameter “phi” from the Pearson statistic:

```
> phi = sum(rp^2)/modp$df.res
> phi
 31.74914
```



I will refit the model, but account for overdispersion. Since the effect of including an overdispersion parameter is to divide variances by phi, all standard deviations are reduced by a factor of approximately the square root of 31.74, the estimate of phi given with the output – so R must be using the Pearson statistic for standardization.

```
> modpn <- glm(Species ~ ., family=quasipoisson, ga)
> summary(modpn)

Call:
glm(formula = Species ~ ., family = quasipoisson, data = ga)

Deviance Residuals:
    Min       1Q     Median      3Q      Max
-8.2752  -4.4966  -0.9443   1.9168  10.1849
```

```
Coefficients:
              Estimate   Std. Error  t value  Pr(>|t|)
(Intercept)  3.1548079  0.2915894  10.819    1.03e-10
Area        -0.0005799  0.0001480  -3.918    0.000649
Elevation    0.0035406  0.0004925   7.189    1.98e-07
Nearest      0.0088256  0.0102621   0.860    0.398291
Scruz       -0.0057094  0.0035251  -1.620    0.118379
Adjacent    -0.0006630  0.0001652  -4.012    0.000511

 (Dispersion parameter for quasipoisson family taken to be 31.74906)

    Null deviance:  3510.73  on 29  degrees of freedom
Residual deviance:   716.85  on 24  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

Note that R labels the Wald tests on the coefficients as t-statistics. After accounting for the overdispersion, neither the Nearest effect nor the Scruz effect is significant. The deviances remain unchanged.

As an alternative to the Wald tests, we can request approximate F-tests for each effect. These correspond to standardizing the drop in deviance from omitting each effect from the full model. Faraway notes that the F approximation is viewed to be more accurate than the normal or t approximation used with the Wald test. To get the approximate F-tests, use the drop1 command:

```
> drop1(modpn,test="F")

Single term deletions

Model:
Species ~ Area + Elevation + Nearest + Scruz + Adjacent

           Df   Deviance   F value    Pr(F)
<none>            716.85
Area        1    1204.35   16.3217   0.0004762
Elevation   1    2389.57   56.0028   1.007e-07
Nearest     1     739.41    0.7555   0.3933572
Scruz       1     813.62    3.2400   0.0844448 .
Adjacent    1    1341.45   20.9119   0.0001230
```

The deviance column specifies the model deviance when the specified effect is omitted from the full model, which has a deviance of 716.85. The p-values are not very different from that obtained from the Wald test.

A reasonable next step in the analysis might be to omit the least significant effect in the model, Nearest. After removing Nearest, Scruz is insignificant (p = .14). I then removed Scruz, leaving a model with three highly significant effects: Area, Elevation, and Adjacent. For the sake of brevity, some output is omitted.

```
> modnew1 <- update(modpn,  ~ .  - Nearest)
> summary(modnew1)

Call:
glm(formula = Species ~ Area + Elevation + Scruz + Adjacent, family = quasipoisson,
data = ga)

Coefficients:
              Estimate    Std. Error  t value     Pr(>|t|)
(Intercept)  3.1599640    0.2805140  11.265     2.75e-11
Area        -0.0005978    0.0001396  -4.283      0.000239
Elevation    0.0035769    0.0004675   7.651      5.25e-08
Scruz       -0.0038565    0.0025216  -1.529      0.138723
Adjacent    -0.0007030    0.0001521  -4.621      9.96e-05

 (Dispersion parameter for quasipoisson family taken to be 29.53500)

> modnew2 <- update(modnew1,  ~ .  - Scruz)
> summary(modnew2)

Call:
glm(formula = Species ~ Area + Elevation + Adjacent, family = quasipoisson,
   data = ga)

Coefficients:
              Estimate    Std. Error  t value  Pr(>|t|)
(Intercept)  2.9613109  0.2617588  11.313    1.53e-11
Area        -0.0005704  0.0001381  -4.129    0.000334
Elevation    0.0035891  0.0004721   7.602    4.54e-08
Adjacent    -0.0007508  0.0001524  -4.928   4.07e-05

 (Dispersion parameter for quasipoisson family taken to be 30.08155)

   Null deviance:   3510.73  on 29  degrees of freedom
Residual deviance:  818.74  on 26  degrees of freedom
AIC: NA
```

I then obtained some diagnostic information for this model, using built-in R functions. Could not figure out how to get standardized deviance residuals, but these are easy to compute, by saving the deviance residuals and dividing them by the square-root of the estimated dispersion times one minus the leverage.

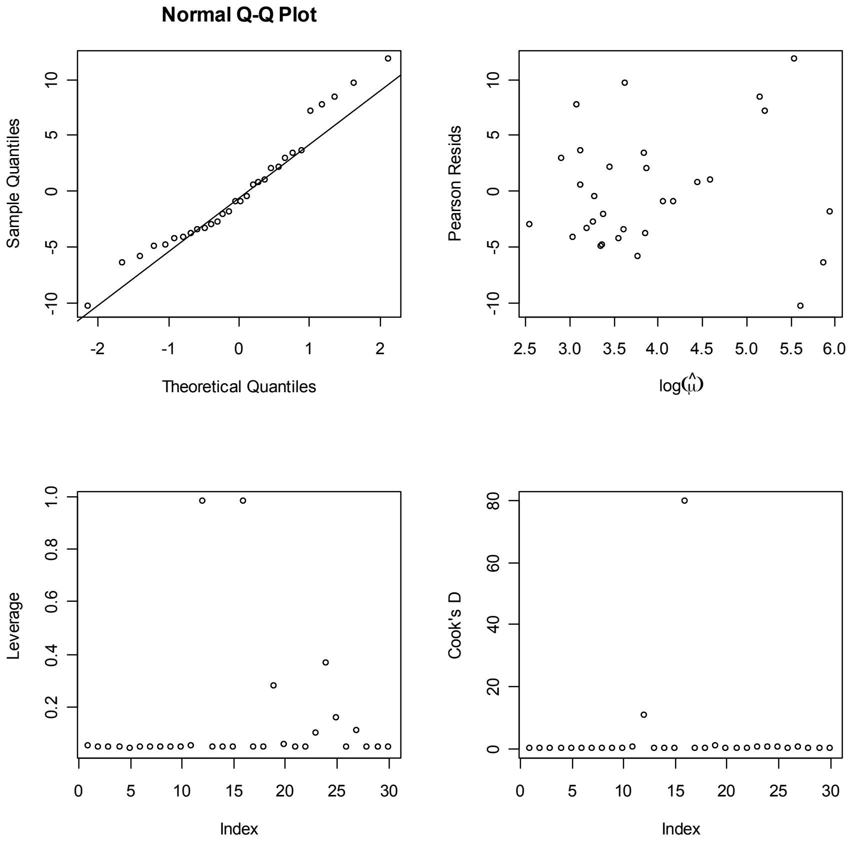
```
> rp = residuals(modnew2,type="pearson")           Pearson residuals
> rpstd = rstudent(modnew2)                         Standardized Pearson residuals
>  lev   = influence(modnew2)$hat                    Case Leverages
> cookd = cooks.distance(modnew2)                   Cook's Distances
> fv <- fitted(modnew2)                             Fitted Values

> cbind(ga$Species,fv,rp,rpstd,lev,cookd)
```

```
               Species    fv         rp       rpstd      lev        cookd
Baltra            58     65.85     -0.967   -0.1767     0.045   3.929696e-04
Bartolome         31     18.58      2.881    0.4729     0.041   3.146199e-03
Caldwell           3     29.07     -4.835   -1.1281     0.045   9.667971e-03
Champion          25     22.78      0.463    0.0816     0.044   8.759058e-05
Coamano            2     12.92     -3.038   -0.6778     0.040   3.364752e-03
Daphne.Major 18          29.57     -2.128   -0.4107     0.045   1.872732e-03
Daphne.Minor 24          26.97     -0.572   -0.1042     0.045   1.350634e-04
Darwin            10     35.19     -4.246   -0.9070     0.045   7.462747e-03
Eden               8     24.59     -3.346   -0.7003     0.044   4.572919e-03
Enderby            2     28.87     -5.001   -1.1960     0.045   1.034475e-02
Espanola          97     38.02      9.563    1.5046     0.045   3.811734e-02
Fernandina        93     86.12      0.741    0.9200     0.979   1.068162e+01
Gardner1          58     22.04      7.657    1.1766     0.044   2.350568e-02
Gardner2           5     43.61     -5.847   -1.3711     0.045   1.410074e-02
Genovesa          40     22.80      3.600    0.5876     0.043   5.151113e-03
Isabela          347    383.14     -1.846   -2.6919     0.981   7.970545e+01
Marchena          51     58.78     -1.014   -0.1855     0.044   4.156694e-04
Onslow             2     21.13     -4.162   -0.9694     0.044   7.011612e-03
Pinta            104    275.57    -10.335   -2.6976     0.278   4.746153e-01
Pinzon           108     98.97      0.907    0.1608     0.054   4.200156e-04
Las.Plazas        12     26.56     -2.826   -0.5681     0.044   3.269615e-03
Rabida            70     46.80      3.390    0.5697     0.044   4.653138e-03
SanCristobal     280    184.21      7.057    1.2490     0.095   4.808517e-02
SanSalvador      237    358.86     -6.433   -1.5441     0.366   3.138335e-01
SantaCruz        444    256.32     11.722    2.2437     0.156   2.513176e-01
SantaFe           62     48.26      1.976    0.3399     0.045   1.610927e-03
SantaMaria       285    174.30      8.384    1.4937     0.105   7.741402e-02
Seymour           44     32.10      2.099    0.3570     0.045   1.811534e-03
Tortuga           16     37.14     -3.468   -0.7037     0.045   4.949248e-03
Wolf              21     47.75     -3.871   -0.7853     0.045   6.168078e-03
```

Looking at these summaries we see two cases with extremely large Cook's distances and leverages (Fernandina and Isabella) and several cases with large standardized Pearson residuals (Isabella, Santa Cruz, Pinta). Leverages in linear regression measure distance in the covariate space from the center, but in GLMs the interpretation is a bit more complicated, because leverages depend on variances. Nonetheless, high leverage cases typically are somewhat extreme in the covariate space.

Diagnostic plots are useful in GLM analyses. I made a normal q-q plot of the standardized Pearson residuals (should appear as a straight line), a plot of the linear predictor against the Pearson residuals, and index plots of the case leverages and Cook's distance. The leverage and Cook's distance plots highlight serious problems.

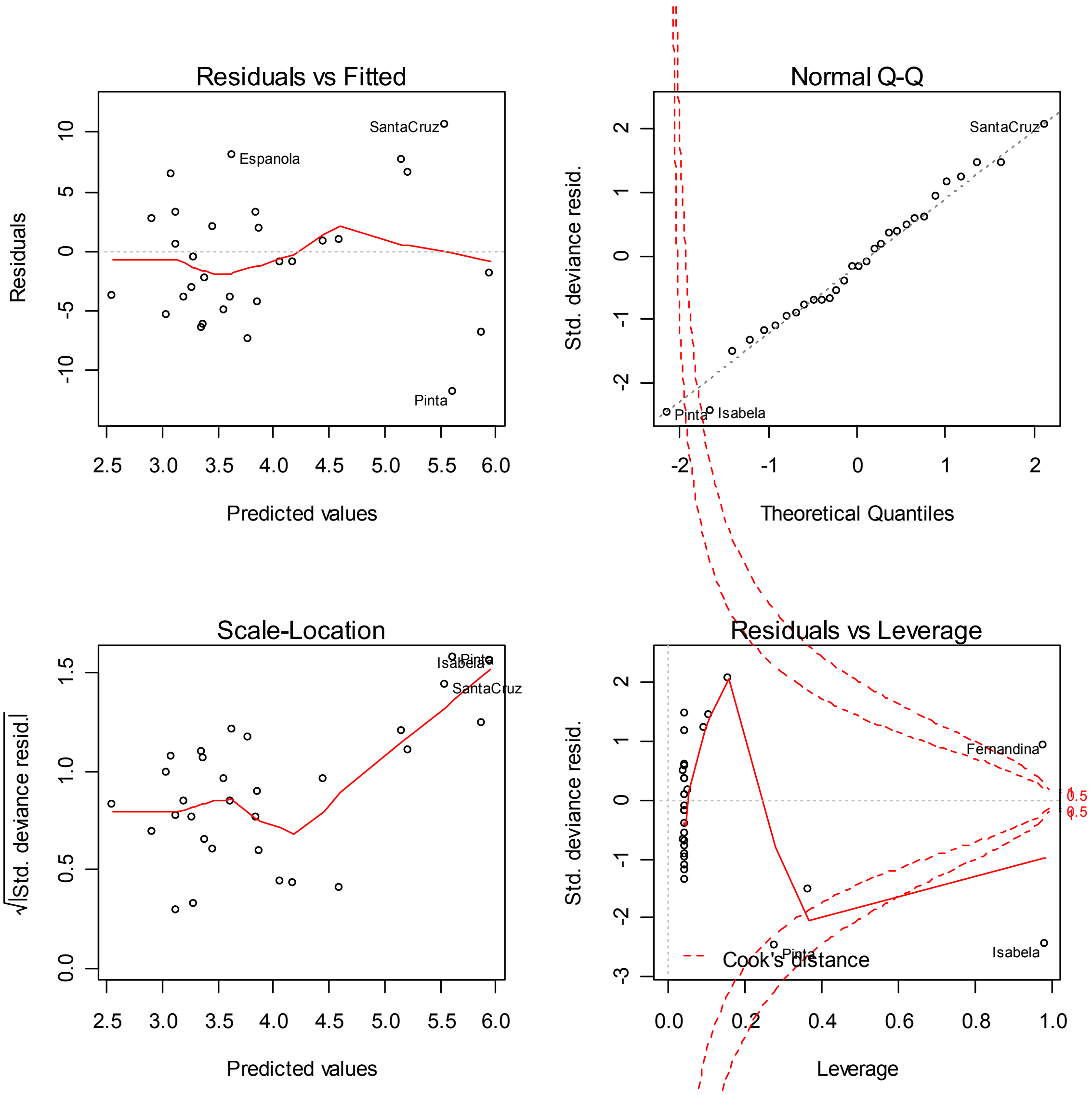
```
> par(mfrow=c(2,2))
> qqnorm(rp);  qqline(rp);
> plot(log(fv), rp, xlab=expression(log(hat(mu))), ylab="Pearson Resids");
> plot(lev, ylab="Leverage");
> plot(cookd, ylab="Cook's D")
```



It is perhaps easier to allow R to create diagnostic plots directly, using the plot command, applied to our model output object:

```
> par(mfrow=c(2,2))
> plot(modnew2)
```

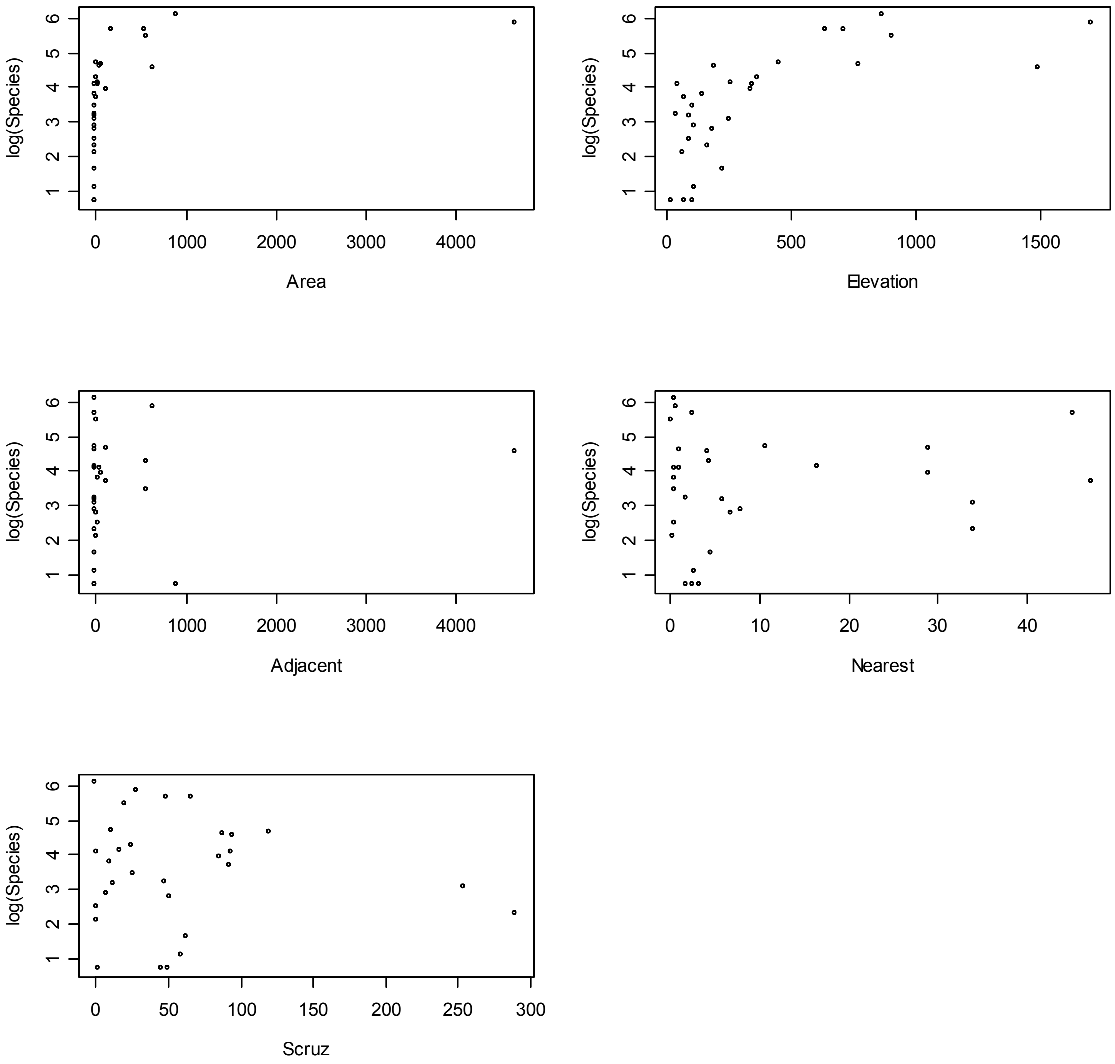
The plot of residuals against fitted values uses the raw deviance residuals plotted against the linear predictor. The normal q-q plot uses the standardized deviance residuals. The scale-location plot is assessing changes in variance as a function of mean. The residual versus leverage plot also includes contours for Cook's distance, which may make it challenging to interpret. A nice feature is that extreme cases are labeled.



Galapagos Islands Data: Revisited

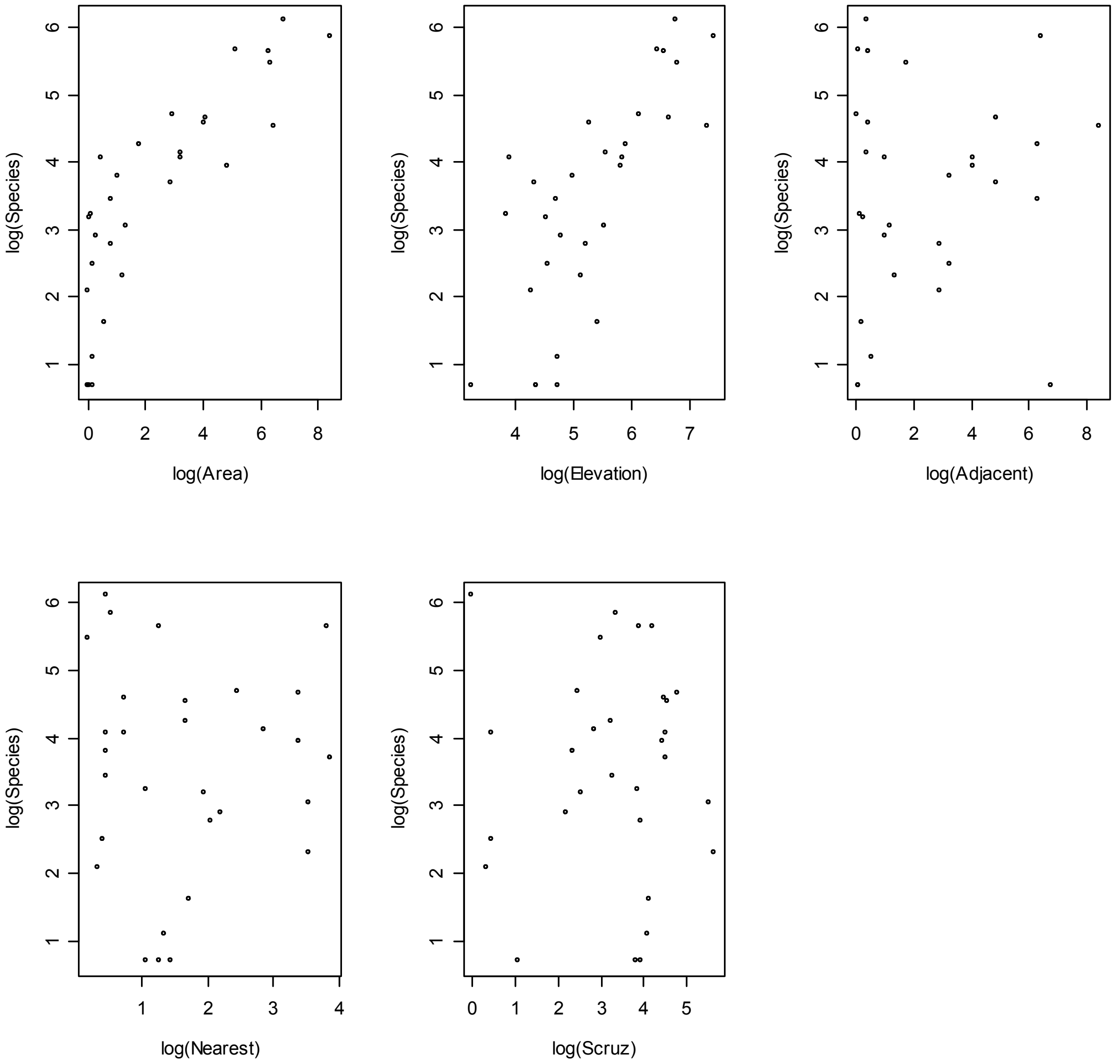
Our analysis of these data had serious deficiencies. Prior to building the model, we did not even consider whether the relationships suggested by the log link made sense. A reasonable first step should have been to plot log(Species), as an approximation to the log-transformed mean, against each predictor. This is done below, and the plots clearly show the need to transform one or more predictors (Area, Elevation, Adjacent).

```
> par(mfrow=c(3,2))
  plot(ga$Area, log(ga$Species),xlab="Area",ylab="log(Species)")
  plot(ga$Elevation, log(ga$Species),xlab="Elevation",ylab="log(Species)")
  plot(ga$Adjacent, log(ga$Species),xlab="Adjacent",ylab="log(Species)")
  plot(ga$Nearest, log(ga$Species),xlab="Nearest",ylab="log(Species)")
  plot(ga$Scruz, log(ga$Species),xlab="Scruz",ylab="log(Species)")
```



A logarithmic transformation is sensible, so for consistency, I transformed each predictor to a log scale (after adding 1 since one or more variables had zeros), and replotted the data. I also created a data frame containing the original response, and the log transformed predictors.

The transformation does an adequate (but not perfect) job of linearizing some of the relationships, so I will pursue building a model using the log-transformed predictors.



Here is the code that generated the plots, and for creating the data frame.

```
> Species =  ga$Species
> Larea   =  log(1+ ga$Area)
> Lelev   =  log(1+ ga$Elevation)
> Ladj    =  log(1+ ga$Adjacent)
> Lnear   =  log(1+ga$Nearest)
> Lscruz  =  log(1+ga$Scruz)

 > plot(Larea, log(Species), xlab="log(Area)",       ylab="log(Species)")
 > plot(Lelev, log(Species), xlab="log(Elevation)",ylab="log(Species)")
 > plot(Ladj,   log(Species), xlab="log(Adjacent)",ylab="log(Species)")
 > plot(Lnear, log(Species), xlab="log(Nearest)",   ylab="log(Species)")
 > plot(Lscruz,log(Species), xlab="log(Scruz)",      ylab="log(Species)")

> gb = data.frame(Species,Larea,Lelev,Ladj,Lnear,Lscruz)
> gb                                                          First few rows shown

     Species        Larea          Lelev         Ladj            Lnear       Lscruz
1       58        3.26155210    5.849325   1.04380405    0.4700036   0.4700036
2       31        0.80647587    4.700480    6.35146147   0.4700036   3.3068867
3        3        0.19062036    4.744932    0.57661336   1.3350011   4.0893320
```

As with the earlier analysis, I will adjust for overdispersion. Fitting the model with each of the 5 predictors gives the following summary.

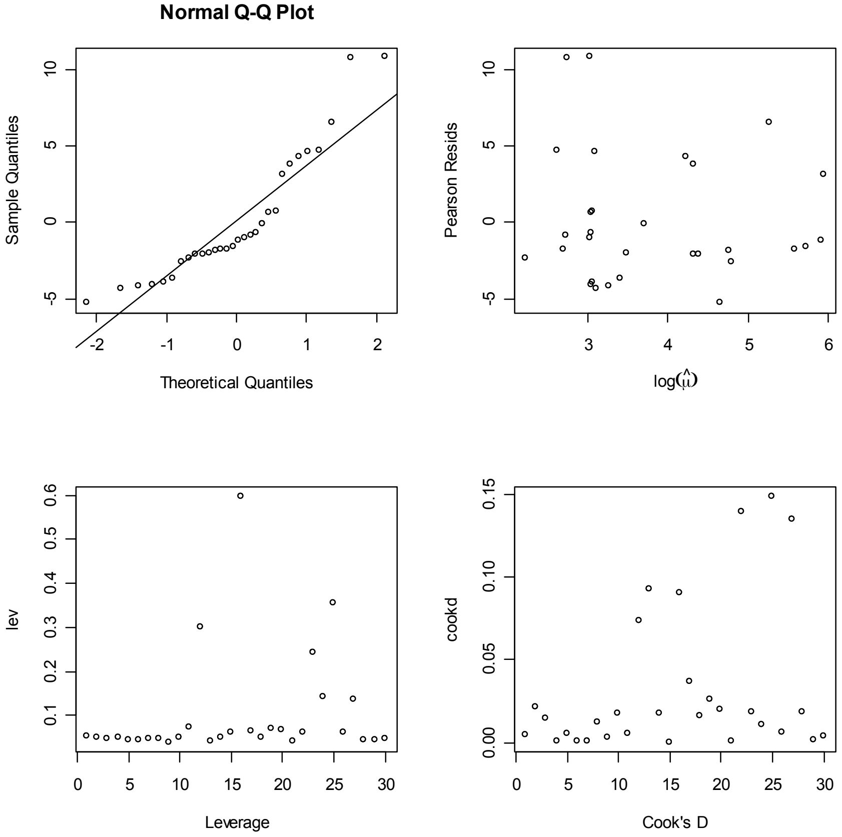
```
> modpn <- glm(Species ~ .,family=quasipoisson, gb)
> summary(modpn)

Call:
glm(formula = Species ~ ., family = quasipoisson, data = gb)

Coefficients:
              Estimate   Std. Error   t value    Pr(>|t|)
(Intercept)  2.25154    1.24722      1.805  0.08360 .
Larea          0.35230    0.09522      3.700  0.00112 **
Lelev          0.21415    0.26472      0.809  0.42648
Ladj         -0.12403    0.04088     -3.034  0.00573 **
Lnear        -0.03206    0.08947     -0.358  0.72320
Lscruz       -0.02868    0.07381    -0.389   0.70105

 (Dispersion parameter for quasipoisson family taken to be 20.27426)

    Null deviance:    3510.73  on 29  degrees of freedom
Residual deviance:   431.51  on 24  degrees of freedom
```

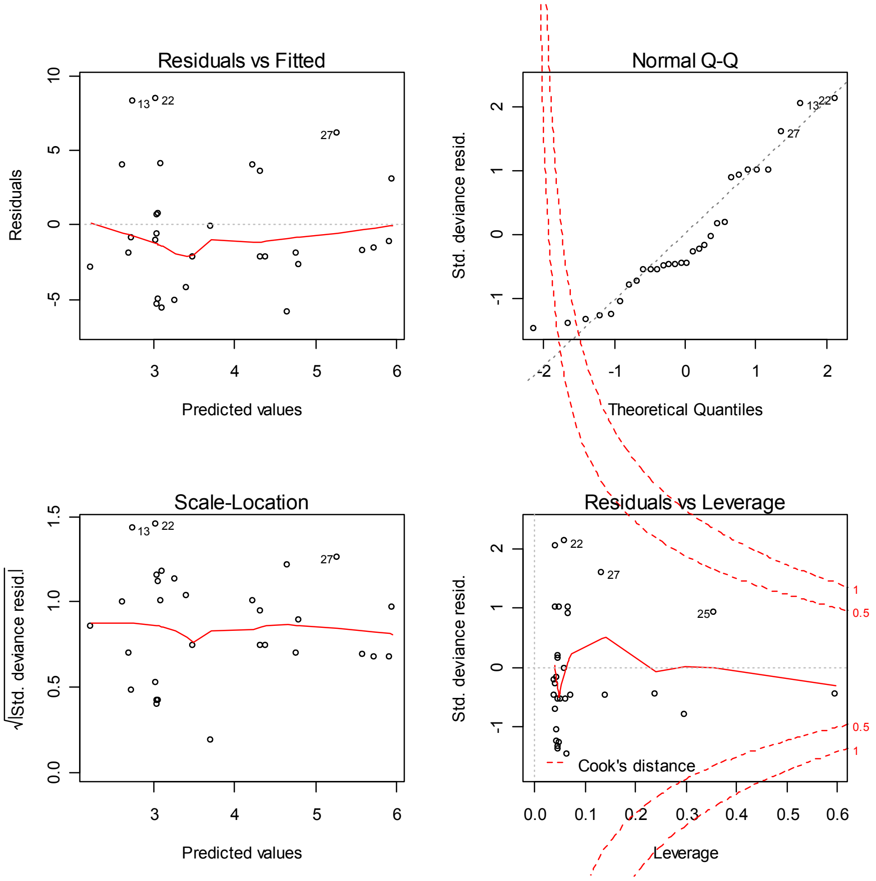
I performed a backward elimination, starting by removing the least significant effect in the full model, Lnear. I then refit the model, omitted the least significant effect (Lcruz), and continued this process until each remaining effect was significant. The resulting model included effects for Larea and Ladj, each of which is highly significant.

```
> summary(modpn3)
```

```
Call:
glm(formula = Species ~ Larea + Ladj, family = quasipoisson, data = gb)

Coefficients:
             Estimate  Std. Error  t value    Pr(>|t|)
(Intercept)  3.05199    0.20863    14.629   2.36e-14 ***
Larea        0.43282    0.03787    11.430   7.50e-12 ***
Ladj        -0.12333    0.03410    -3.616   0.00121 **

 (Dispersion parameter for quasipoisson family taken to be 19.01247)

    Null deviance:  3510.7  on 29  degrees of freedom
Residual deviance:   459.2  on 27  degrees of freedom
```

As with the earlier analysis, I made diagnostic plots based on summaries that I saved and by using the built-in R function plot. Isabella (case 16), which is the largest island, and which has a large adjacent island, has a large leverage, but no individual case appears to have a strong effect on the estimated regression coefficient vector (i.e. a large Cook's distance).





The residuals should be approximately normal when the model holds and (under a Poisson model) when all the means are large (which they are not). Although there may be some deficiencies with this model, it would appear to be a marked improvement upon our earlier attempt. I will stop at this point, and just summarize that the two primary features that would appear to impact the number of species are the island's size, and the size of the adjacent island. This is perhaps not too surprising.

Any comments?

Discussion: How does the area of the island and the area of the adjacent island impact the number of species on the island? Does this make sense?

You should note that I am giving you all the R code so you can reproduce things yourself. When handing in an analysis, there is not a need to include all the code, but rather limit your discussion to the statistical issues, and relevant summaries.

Quasi-likelihood with different variance functions

In GLMs with “family=quasi” we can try different combinations of variance and link functions. (These are restricted to just a few “canned” choices, but the user can in principle program any function.) Here we illustrate by first reproducing the last model with “family=quasi”:

```
> glm2 <- glm(Species ~ Larea+Ladj, family=quasi(variance = "mu",
link = "log"), galab)
> summary(glm2)
Coefficients:
           Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.05199    0.20863  14.629 2.36e-14 ***
Larea        0.43282    0.03787  11.430 7.50e-12 ***
Ladj        -0.12333    0.03410  -3.616  0.00121 **
---
(Dispersion parameter for quasi family taken to be 19.01248)
    Null deviance: 3510.7  on 29  degrees of freedom
Residual deviance:  459.2  on 27  degrees of freedom
```

The plot of squared residuals vs. fitted values on p. 2 (which suggested overdispersion), also suggests that we might try the variance to be something like a polynomial in the mean. Here we try V(mu)=mu^2:

```
> glm3 <- glm(Species ~ Larea+Ladj, family=quasi(variance =
"mu^2", link = "log"), galab)
> summary(glm3)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.712341   0.222159  12.209 1.67e-12 ***
Larea        0.450912   0.055039   8.193 8.48e-09 ***
Ladj        -0.005249   0.055561  -0.094    0.925
---
(Dispersion parameter for quasi family taken to be 0.5529679)
    Null deviance: 56.266  on 29  degrees of freedom
Residual deviance: 18.274  on 27  degrees of freedom
```

However, the result is not as good; there is now a funnel shape in the scale-location diagnostic plot...

Automatic model selection (without overdispersion)

In GLMs with overdispersion, only the mean and variance functions are specified, which is not enough to determine a distribution, and thus there is no likelihood (and therefore no AIC).

For GLMs without overdispersion, -2log(Likelihood)=Deviance and thus:

$$\textbf{AIC = Deviance + 2(number of parameters)}$$

We can automate model selection in these cases via the "step" function in R, which does stepwise model selection. Using "step" we obtain the full model with the 5 log transformed predictors (and no overdispersion).

```
> galab = data.frame(Larea,Lelev,Ladj,Lnear,Lscruz,Species)
> glm.full <- glm(Species ~ ., family=poisson, galab)
> summary(glm.full)
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.25154    0.27699   8.128 4.35e-16 ***
Larea        0.35230    0.02115  16.659  < 2e-16 ***
Lelev        0.21415    0.05879   3.643  0.00027 ***
Ladj        -0.12403    0.00908 -13.660  < 2e-16 ***
Lnear       -0.03206    0.01987  -1.614  0.10661
Lscruz      -0.02868    0.01639  -1.749  0.08022 .
---
(Dispersion parameter for poisson family taken to be 1)
    Null deviance: 3510.73  on 29  degrees of freedom
Residual deviance:  431.51  on 24  degrees of freedom
AIC: 604.34
```

Note that in defining "galab" we need to place the response as the last column of the dataframe, since the package "bestglm" on the next page expects it that way!

```
> step(glm.full, trace=T)
Start:  AIC=604.34
Species ~ Larea + Lelev + Ladj + Lnear + Lscruz

         Df Deviance    AIC
<none>        431.51 604.34
- Lnear   1   434.12 604.95
- Lscruz  1   434.57 605.40
- Lelev   1   445.13 615.96
- Ladj    1   626.41 797.24
- Larea   1   705.54 876.37
```

For traversing the entire model space, the version of the “regsubsets” function (package “leaps”), is implemented in package “bestglm” (function has same name), but again, it does not handle overdispersion. We illustrate with a variety of information criteria (IC), in particular cross-validation (CV), which is generally applicable when a likelihood-based criterion like AIC/BIC is not available:

```
### BIC
> bestglm(galab, family=poisson, IC="BIC")
Best Model:
               Estimate Std. Error    z value     Pr(>|z|)
(Intercept)  2.25147641 0.27637148   8.146558 3.744279e-16
Larea        0.35540034 0.02097196  16.946450 2.044295e-64
Lelev        0.20991934 0.05855078   3.585253 3.367521e-04
Ladj        -0.11958950 0.00864749 -13.829390 1.694523e-43
Lscruz      -0.04452264 0.01317517  -3.379285 7.267466e-04

### CV default
> bestglm(galab, family=poisson, IC="CV")
Best Model:
            Estimate  Std. Error  z value Pr(>|z|)
(Intercept) 2.957891 0.045116023 65.56187        0
Larea       0.385460 0.007692144 50.11087        0

### 10-fold CV of Hastie et al (2009).
> bestglm(galab, family=poisson, IC="CV",
CVArgs=list(Method="HTF", K=10, REP=1))
Best Model:
              Estimate  Std. Error   z value    Pr(>|z|)
(Intercept)  3.0519894 0.047847451  63.78583 0.00000e+00
Larea        0.4328238 0.008684844  49.83668 0.00000e+00
Ladj        -0.1233313 0.007821115 -15.76902 5.08364e-56

### 10-fold with DH Algorithm.
> bestglm(galab, IC="CV", CVArgs=list(Method="DH", K=10,
REP=100))
Best Model:
              Estimate Std. Error   t value     Pr(>|t|)
(Intercept)  45.315046  21.698940  2.088353 4.670363e-02
Larea        40.402631   4.023028 10.042840 1.939359e-10
Ladj         -9.506318   4.063597 -2.339385 2.727081e-02
Lnear       -21.494096   8.532889 -2.518971 1.825537e-02
```