Model Checking and Evaluation

Therri Usher

Friday, April 7, 2015

Discussion Time

In the past, after you have fit a model, what do you do then? For instance, do you draw conclusions from the model? Do you continue to try out other models?

Discuss in your groups before we discuss as a class.

Model Checking

We use exploratory data analysis and scientific reasoning to determine a model to fit and to get an idea of what the model will tell us.

However, it is still necessary to check that the model is a good way of explaining the data and if we can trust the estimates from the model.

In addition, we might have multiple models that may answer our research question. We have to find some way to compare the models in order to choose the best one.

Residuals

A residual is the difference between the observed value and the estimated value.

Studentized Residuals

Studentized residuals are a standardized form of residuals. A studentized residual is the residual divided by its standard deviation.

For linear regression, studentized residuals are equal to:

$$r_i = \frac{e_i}{s\sqrt{1 - h_{ii}}} = \frac{y_i - \hat{y}_i}{s\sqrt{1 - h_{ii}}}$$

where e_i is the residual for observation i, s is the sample standard deviation for y, and h_{ii} is the leverage for observation i.

Leverage and Influence

Leverage in statistics is the amount of *potential* influence on the fit of a model. It can also be thought of as the amount of input a data point has on its own predicted value. In regression, points that are far away from the average of an independent variable can have high leverage.

In contrast, influence is the *actual* effect a data point has on a model. Points that have a noticeable effect on regression coefficients are influential points.

Influential points are typically outliers and/or high leverage points.

Leverage and Influence

We can calculate the leverage of points for linear and GLM models. In linear regression, leverage values come from the "hat matrix", typically denoted as H. The leverage of observation i, h_{ii}, is the i^{th} diagonal of the hat matrix.

Leverage and Influence

For linear regression,

$$\hat{Y} = X(X'X)^{-1}X'Y$$
$$\hat{Y} = HY$$
$$H = X(X'X)^{-1}X'$$

PRESS Residuals

One way to measure influence of a point is to calculate its PRESS residual. A PRESS residual for observation i is the residual of the point generated from the model with all data points minus the residual of the point generated from the model with all data points except observation i.

$$e_{i,-i} = y_i - y_{i,-i} = \frac{e_i}{1 - h_{ii}}$$

If a point has large influence, we expect it to have a large PRESS residual.

Therefore, the PRESS statistic of a model is:

$$\sum_{i=1}^{n} e_{i,-i}^2$$

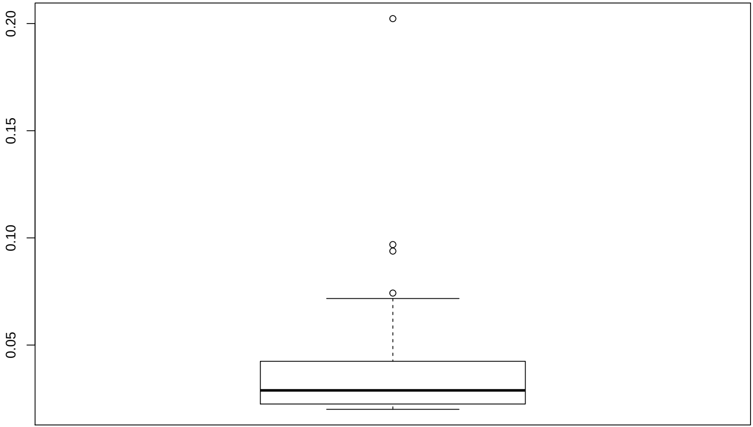
This can be thought of as a prediction error rate generated from a leave-one-out cross-validation.

Residuals and Leverage in R

```r
data(cars)
linear.fit <- lm(speed ~ dist, data=cars)
res <- linear.fit$residuals
student.res <- rstudent(linear.fit)
```

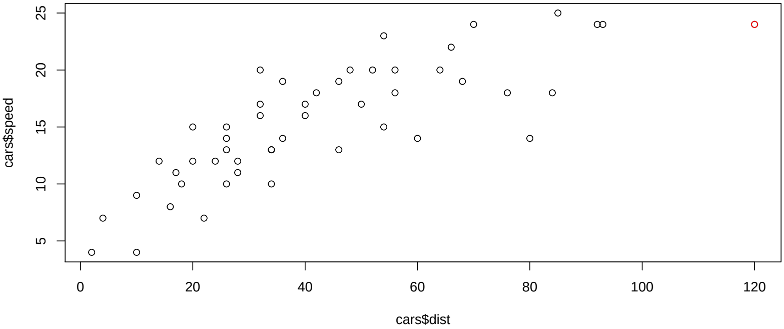
Residuals and Leverage in R

```
lev <- hatvalues(linear.fit)
boxplot(lev)
```



Residuals and Leverage in R

```
plot(cars$dist, cars$speed)
points(cars$dist[lev>0.1],cars$speed[lev>0.1],col="red")
```



Residuals and Leverage in R

```
press.res <- res / (1-lev)
(press.stat <- sum(press.res^2))
```

```
## [1] 526.3
```

Residual Plots

Residual plots are plots that have the fitted values of the outcome or a predictor as the independent variable and the residuals, studentized or not, as the dependent variable.

We can use residual plots to check if:

- The regression model is non-linear.
- The error terms have nonconstant variance.
- The error terms are not normally distributed.
- There are outliers.

Residual Plots

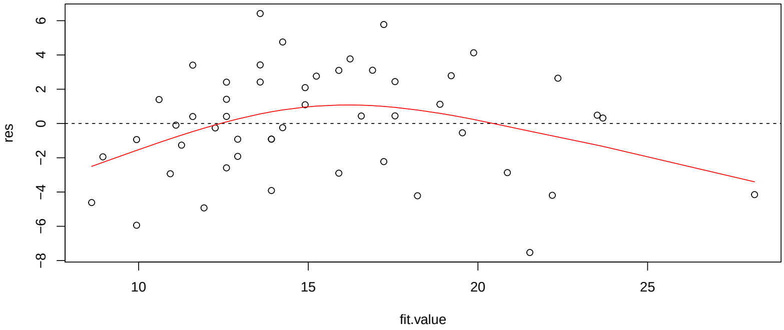
When we make a residual plot, we expect:

1. The residuals to be centered around zero. If not, there is evidence of non-linearity.
2. The residuals to have equal variance across the fitted values. Unequal variance across the plot indicates nonconstant variance.
3. When plotting studentized residuals, they should be within -3 and 3. Points outside the range are typically outliers.

Finally, plotting a predictor versus a residual can give us further information about non-linearity. We can also use a QQ-normal plot to assess normality of the residuals.

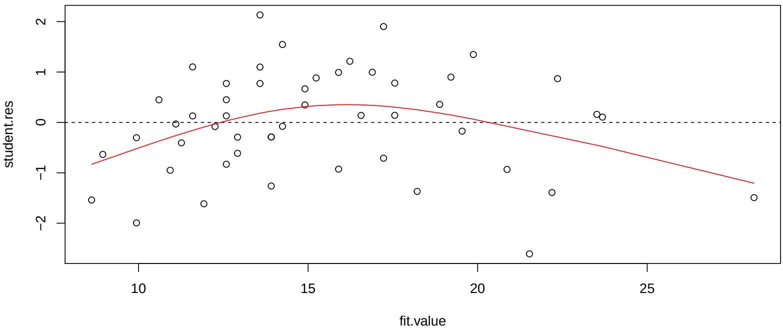
Residual Plots

```
fit.value <- fitted(linear.fit)
smoother1 <- smooth.spline(fit.value, res)
plot(fit.value, res)
abline(h=0, lty=2)
lines(smoother1, col="red")
```



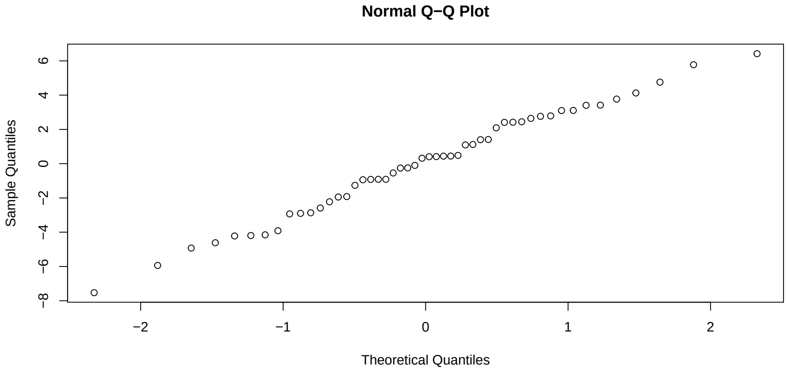
Residual Plots

```
smoother2 <- smooth.spline(fit.value, student.res)
plot(fit.value, student.res)
abline(h=0, lty=2)
lines(smoother2, col="red")
```



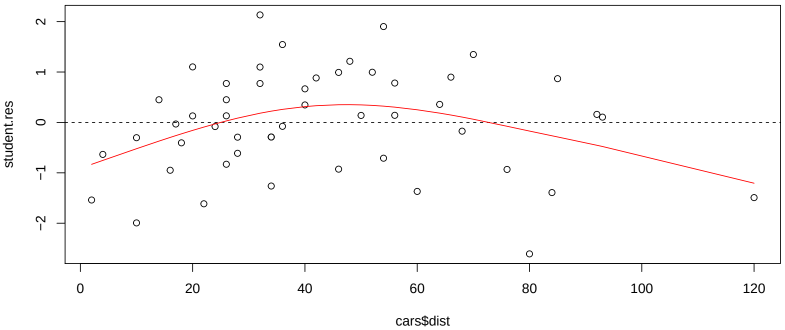
Residual Plots

```
qqnorm(res)
```



Residual Plots

```
smoother3 <- smooth.spline(cars$dist, student.res)
plot(cars$dist, student.res)
abline(h=0, lty=2)
lines(smoother3, col="red")
```



Partial Regression Plots

Partial regression plots are also called added variable plots or adjusted variable plots. They allow us to view the relationship of a predictor and the outcome, given all other predictors.

Say that we want to make the partial regression plot for Y and X_1:

1. Regress Y on all the predictors except X_1. Save the residuals.
2. Regress X_1 on all the other predictors. Save the residuals.
3. Plot the residuals from step 2 on the x-axis and the residuals from step 1 on the y-axis.

Partial Regression Plots

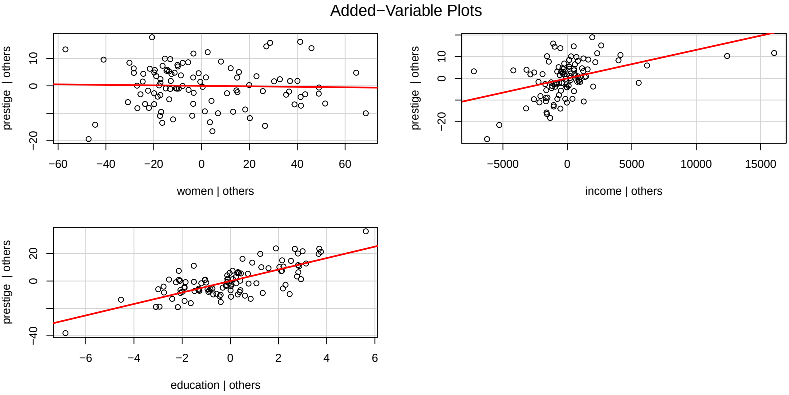
```r
require(car)
```

```
## Loading required package: car
```

```r
data(Prestige)
```

Partial Regression Plots

```
linear.fit2 <- lm(prestige ~ women + income + education,
                  data=Prestige)
avPlots(linear.fit2)
```

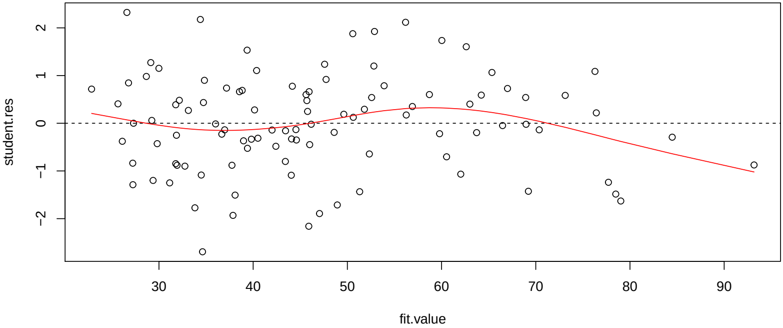


Residual Plots

```r
fit.value <- fitted(linear.fit2)
res <- residuals(linear.fit2)
student.res <- rstudent(linear.fit2)
```

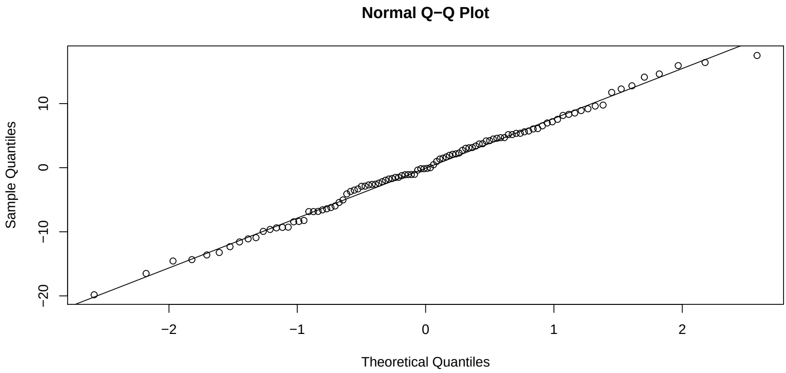
Residual Plots

```
smoother4 <- smooth.spline(fit.value, student.res)
plot(fit.value, student.res)
abline(h=0, lty=2)
lines(smoother4, col="red")
```



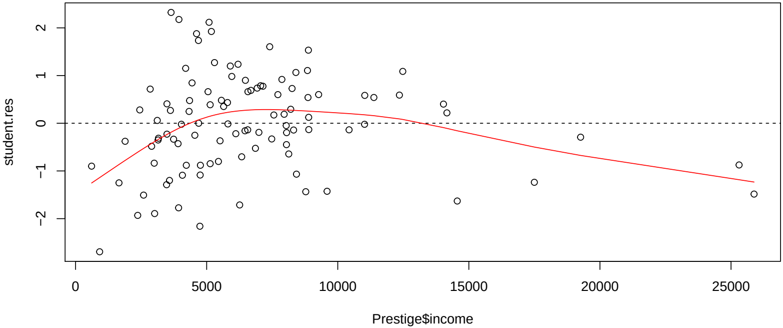
Residual Plots

```
qqnorm(res)
qqline(res)
```



Residual Plots

```
smoother5 <- smooth.spline(Prestige$income, student.res)
plot(Prestige$income, student.res)
abline(h=0, lty=2)
lines(smoother5, col="red")
```



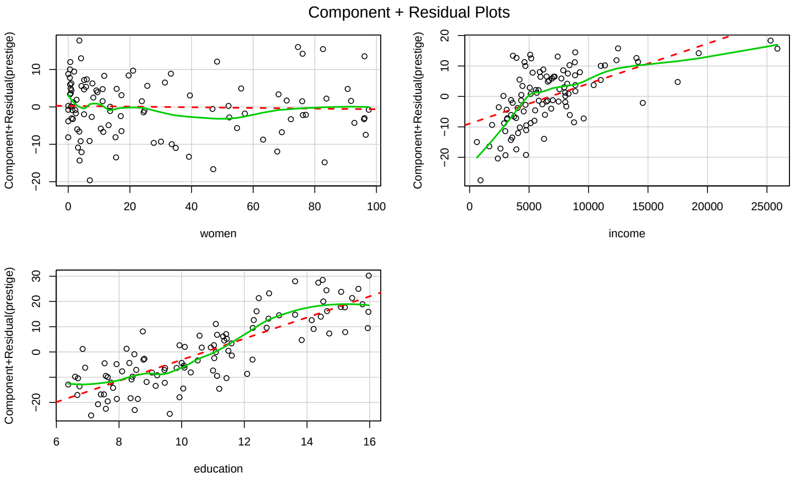
Partial Residual Plots

Partial residual plots, or component-plus-residual plots, also assess the relationship between a predictor and an outcome. However, partial residual plots can tell us if we need to transform the predictor or add a higher power of a predictor.

Partial residuals are defined as $e_i^{(j)} = e_i + \beta_j X_{ij}$. The plot is X_{ij} versus $e_i^{(j)}$.

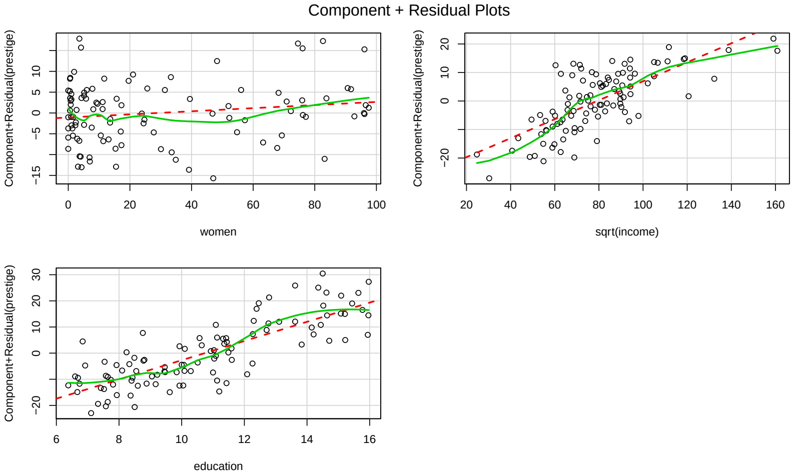
Partial Residual Plots

```
crPlots(linear.fit2)
```



Partial Residual Plots

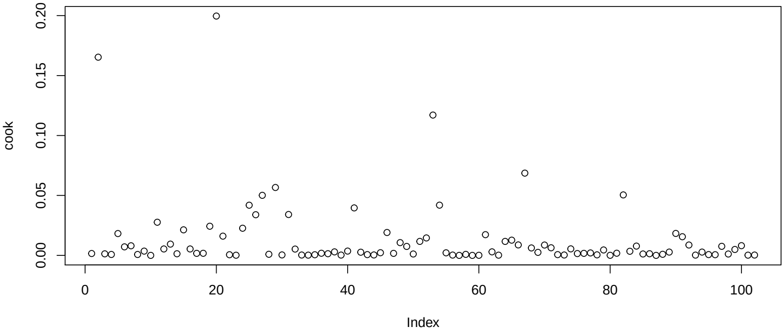
```
linear.fit3 <- lm(prestige ~ women + sqrt(income)
                      + education, data=Prestige)
crPlots(linear.fit3)
```



Cook's Distance

Cook's distance measures the influence of observation *i* on all of the fitted values. It is dependent on both the residual and the leverage of a data point.

```r
cook <- cooks.distance(linear.fit3)
plot(cook)
```



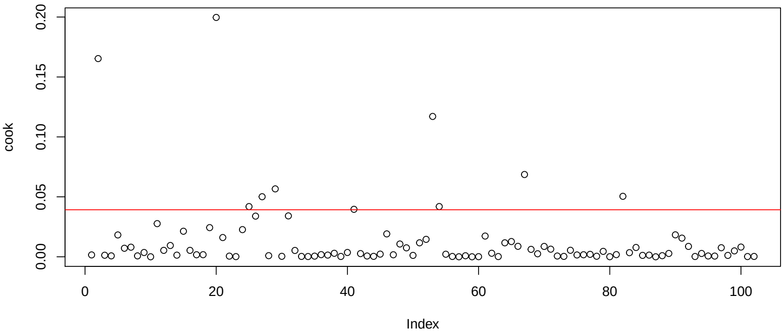
Cook's Distance

There are various ways for assessing highly influential observations. One suggestion is $D_i > 1$, where D_i is the Cook's distance for observation i. Others have suggested $D_i > \frac{4}{n}$, where n is the sample size.

One rather conservative approach is comparing the Cook's distance to a $F_{p,n-p,1-\alpha}$ distribtion where p is the number of predictors in the model.

Cook's Distance

```
n <- dim(Prestige)[1]
plot(cook)
abline(h=1, col="blue")
abline(h=4/n, col="red")
```



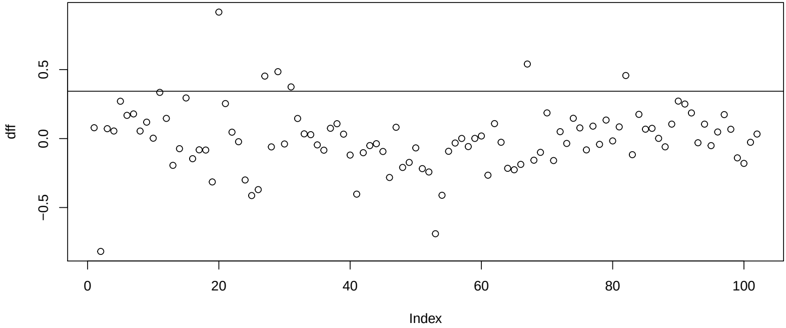
DFFITS

DFFITS is a measure of influence of an observation on its own fitted value.

A large value indicates a large influence on the fitted value. A rule of thumb is $|DFFITS_i| > 2\sqrt{p/n}$, where $DFFITS_i$ is the DFFITS of observation i.

DFFITS

```
dff <- dffits(linear.fit3)
plot(dff)
abline(h=(2*sqrt(3/n)))
```



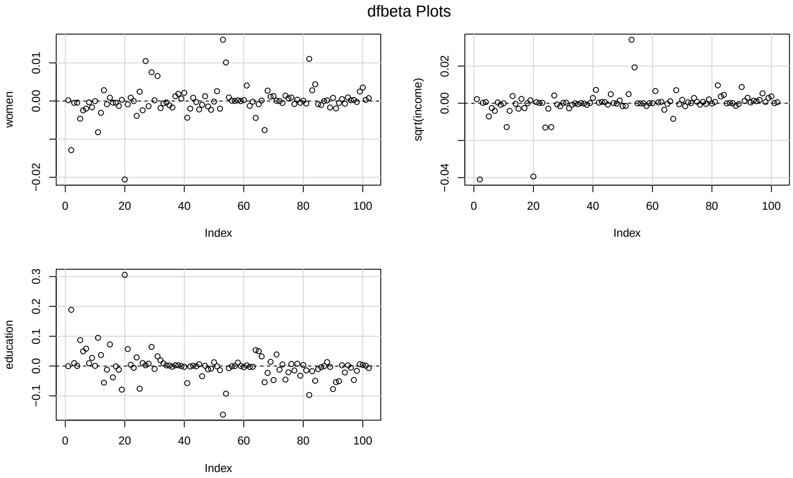
DFBETAS

DFBETAS measures an observation's influence on a regression coefficient.

A large value indicates a large impact on the regression coefficient. The rule of thumb is $|DFBETAS_{k,i}| > \frac{2}{\sqrt{n}}$ where $DFBETAS_{k,i}$ is the influence for observation i on regression coefficient k.

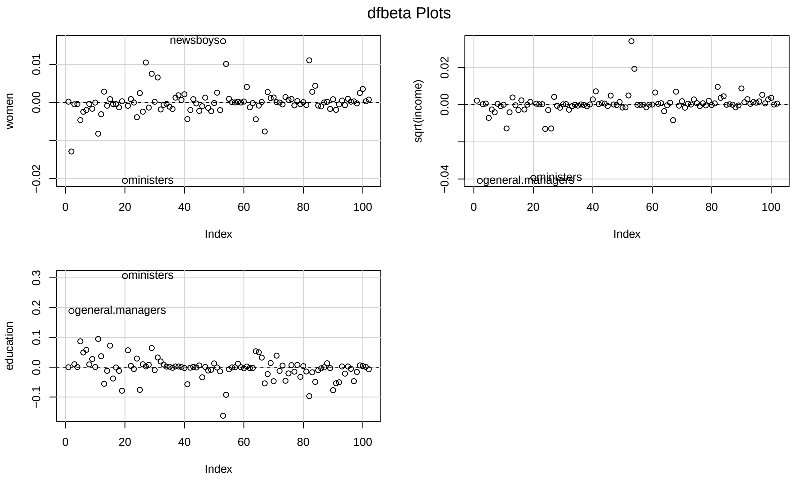
DFBETAS

```
dfb <- dfbetas(linear.fit3)
dfbetaPlots(linear.fit3)
```



DFBETAS

```
dfbetaPlots(linear.fit3, id.n=2)
```



Helpful Tip

Some of these plots can be created just by plotting the fit object.

```
plot(linear.fit3)
```

Generalized Linear Regression

Good news: We can still use many of the same diagnostic tools with generalized linear models!

Bad news: The calculations of the diagnostic tools differ from linear regression.

Residuals differ some in the GLM framework. There are two different kinds to consider:

1. Pearson - raw residuals minus estimated mean divided by their standard error
2. Deviance - can think of as each point's contribution to the likelihood

Both kinds of residuals can be standardized to create studentized Pearson and studentized deviance residuals. A studentized residual with an absolute value of 2 or more warrants further inquiry.

Pearson and Deviance Residuals

```r
dat <- read.csv("http://www.ats.ucla.edu/stat/data/binary.c
logit.fit <- glm(admit ~ gre + gpa + rank, data=dat,
                 family="binomial")
fit.values <- logit.fit$fitted.values
pear.res <- residuals(logit.fit, type="pearson")
dev.res <- residuals(logit.fit, type="deviance")
stu.dev.res <- rstudent(logit.fit, type="deviance")
stu.pear.res <- pear.res/sqrt(1-hatvalues(logit.fit))
```

GLM - Diagnostic Plots

Fitted values versus studentized residuals

- A trend that is not near or around zero can indicate model misspecification, such as a missing predictor or improper link function.
- Non-constant variability in the residuals can indicate a problem with the mean-variance relationship and therefore the assumed outcome distribution.

Predictors versus studentized residuals

- Plot them for all predictors that are in AND not in the model. Trends in the mean of the residuals versus predictors can indicate missing predictors.

GLM - Diagnostic Plots

Distribution of studentized residuals

- Pearson residuals are supposed to be asymptotically normal. However, this often breaks down. Do not worry if this happens for your model.

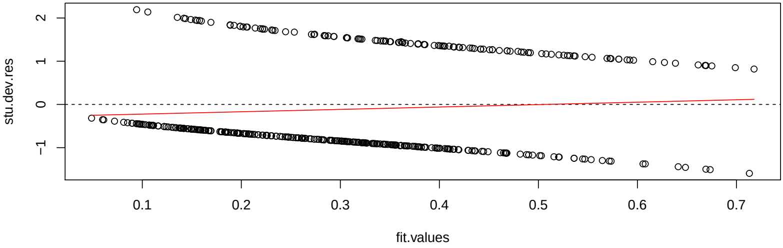
Leverage plots

- They have the same interpretation as linear regression.

GLM - Residual Plots

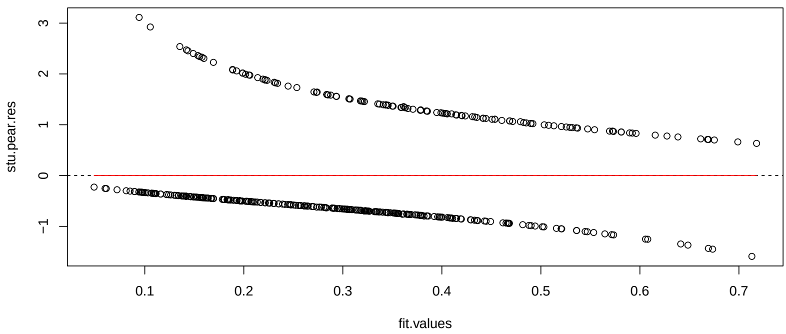
Notice that the residuals are patterned. That is not a cause for alarm, particularly for dichotomous data, which can only take on one of two values.

```r
smoother6 <- smooth.spline(fit.values, stu.dev.res)
plot(fit.values, stu.dev.res)
abline(h=0, lty=2)
lines(smoother6, col="red")
```



GLM - Residual Plots

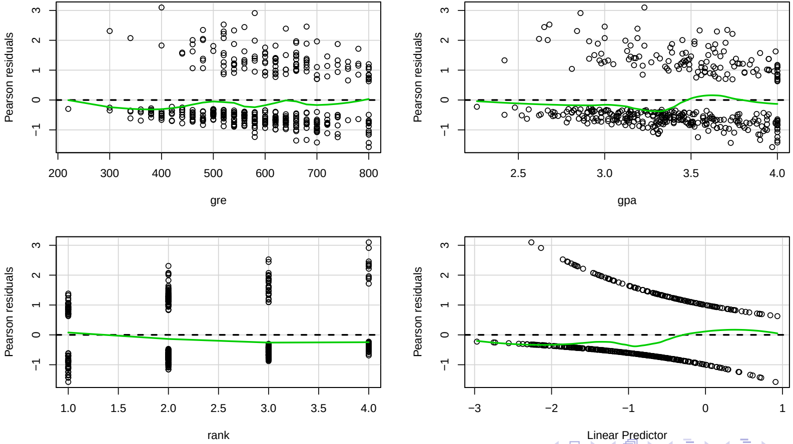
```
smoother7 <- smooth.spline(fit.values, stu.pear.res)
plot(fit.values, stu.pear.res)
abline(h=0, lty=2)
lines(smoother7, col="red")
```



GLM - Residual Plots

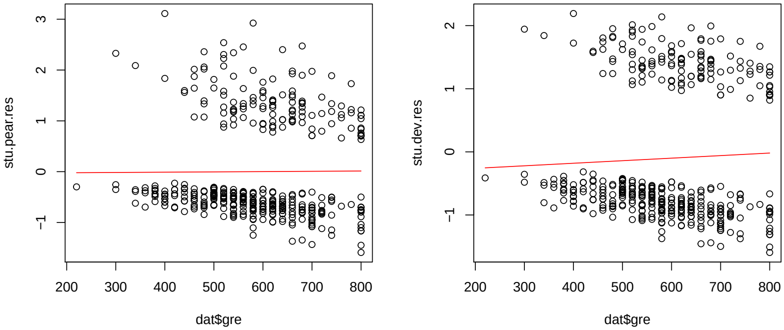
We can observe relationships between Pearson residuals and predictors.

```
residualPlots(logit.fit)
```



GLM - Residual Plots

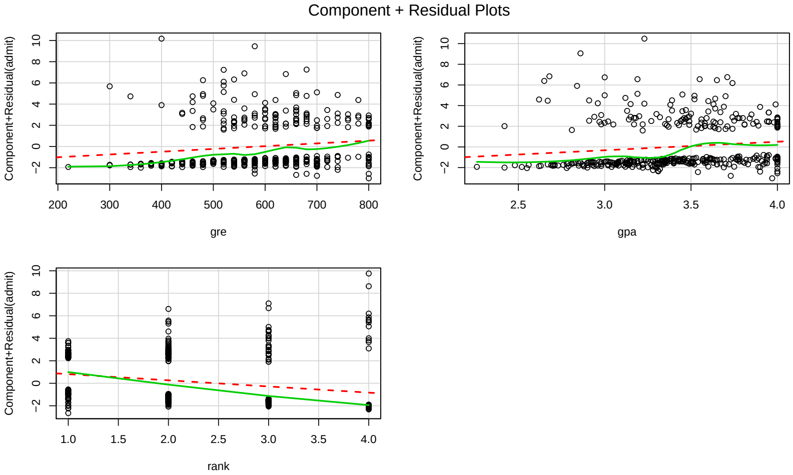
```
par(mfrow=c(1,2))
plot(dat$gre, stu.pear.res)
lines(smooth.spline(dat$gre, stu.pear.res), col="red")
plot(dat$gre, stu.dev.res)
lines(smooth.spline(dat$gre, stu.dev.res), col="red")
```



GLM - Partial Residual Plots

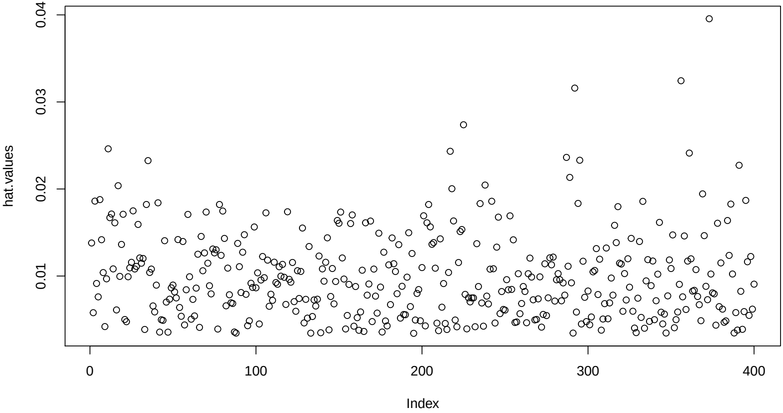
Partial residual plots have the same interpretation.

```
crPlots(logit.fit)
```



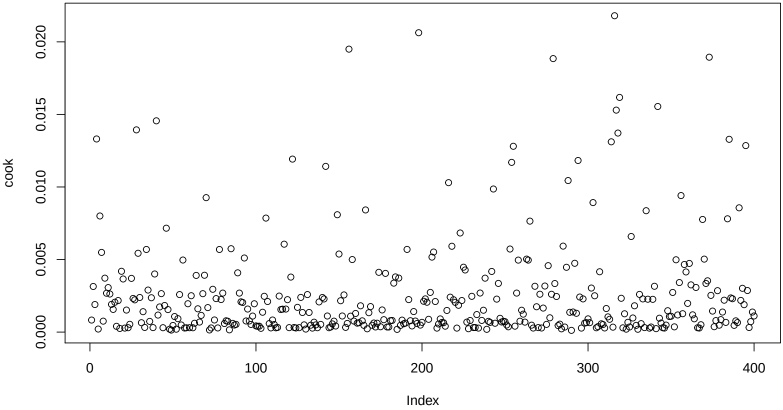
GLM - Leverage

```r
hat.values <- hatvalues(logit.fit)
plot(hat.values)
```



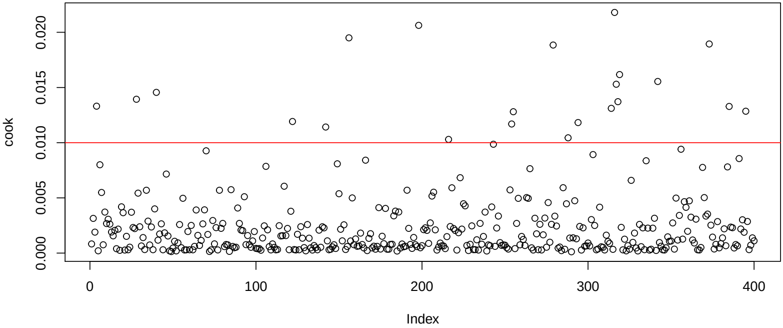
GLM - Cook's Distance

```r
cook <- cooks.distance(logit.fit)
plot(cook)
```



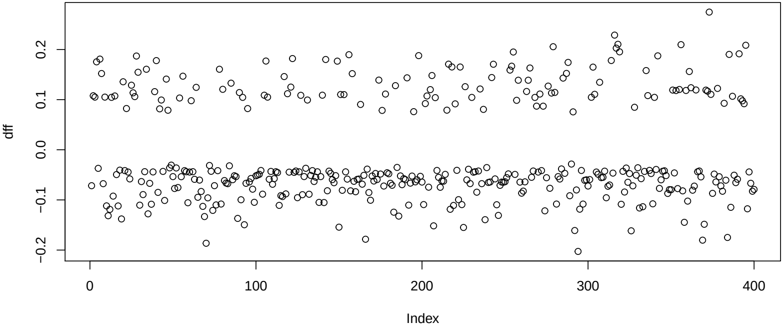
GLM - Cook's Distance

```
n <- dim(dat)[1]
plot(cook)
abline(h=1, col="blue")
abline(h=4/n, col="red")
```



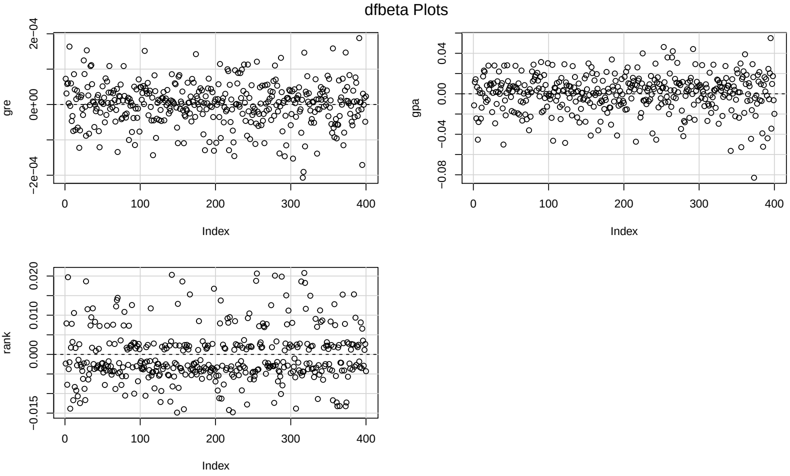
GLM - DFFITS

```r
dff <- dffits(logit.fit)
plot(dff)
abline(h=(2/sqrt((3+1)/n)))
```



GLM - DFBETAS

```
dfb <- dfbetas(logit.fit)
dfbetaPlots(logit.fit)
```



Take-Home Message

Diagnostic plots are well-defined for linear regression but not so much for GLM's. Use them but do not abandon scientific reasoning in guiding your decisions.

Active Learning Exercise

1. Load the Freedman data in the car package. Build a model with crime as the outcome and all other variables as predictors. Assess the model fit using plots and diagnostic tools. Is there any evidence that the model might be misspecified?

Active Learning Exercise

```
data(diamonds)
n.obs <- dim(diamonds)[1]
big <- matrix(nrow=n.obs)

big[diamonds$carat < 1] <- 0
big[diamonds$carat >= 1] <- 1
```

2. Use the code above to load and modify the diamonds data set in ggplot2 package. (Remember to load the package.) Fit a model for predicting whether a diamond is “big”. Are there any highly influential data points? If so, which ones?

Model Fit

Even if all the diagnostic tools of a model indicate that nothing is wrong with the model, that does not necessarily mean that the model fits the data well. There are additional methods for assessing goodness of fit, or the ability of the model to fit the set of observed data.

Mean Squared Error

In linear regression, the mean squared error is the sum of the squared residuals, or

$$\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

It is a measure of the errors in the prediction of the outcome and can give us an idea of how well the model fits the data. In linear regression, this is the quantity that the parameters seek to minimize.

NOTE: Only compare MSE's between models if they were fit using the same samples.

Mean Squared Error

```r
require(car)
data(Prestige)

n <- dim(Prestige)[1]
model1 <- lm(prestige ~ education, data=Prestige)
model2 <- lm(prestige ~ education + income + women,
             data=Prestige)
```

Mean Squared Error

```r
sum(residuals(model1)^2)
```

```
## [1] 8287
```

```r
sum(residuals(model1)^2)/(102-2)
```

```
## [1] 82.87
```

```r
summary(model1)$sigma^2
```

```
## [1] 82.87
```

Mean Squared Error

```
sum(residuals(model2)^2)
```

```
## [1] 6034
```

```
sum(residuals(model2)^2)/(102-4)
```

```
## [1] 61.57
```

```
summary(model2)$sigma^2
```

```
## [1] 61.57
```

R^2

R^2 is the estimated proportion of the variation in the data that is explained by the model. For linear regression, it is equal to:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

R^2

One of the problems with R^2 is that the more predictors you add, the higher R^2 gets, regardless of fit. Therefore, the adjusted R^2 accounts for sample size and number of parameters in the model.

$$R^2 = 1 - \frac{\frac{SS_{res}}{df_{res}}}{\frac{SS_{tot}}{df_{tot}}} = 1 - \frac{\frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{n-1}}{\frac{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}{n-p-1}}$$

R^2

```r
summary(model1)$r.squared
```

```
## [1] 0.7228
```

```r
summary(model1)$adj.r.squared
```

```
## [1] 0.72
```

```r
summary(model2)$r.squared
```

```
## [1] 0.7982
```

```r
summary(model2)$adj.r.squared
```

```
## [1] 0.792
```

Null and Saturated Models

A null model is a model that seeks to explain the data for all n data points with one parameter.

A saturated model is a model that assumes each data points has its own parameter, n parameters for n data points. As a result, saturated models fit the observed data perfectly.

For this discussion, the model of interest will be called the proposed model, which has $p+1$ parameters for n data points and $n > p+1$.

Maximum Likelihood

All generalized linear regression models utilize maximum likelihood estimates to fit the models. With maximum likelihood estimation, we assume a distribution of the data. The distribution of the data is the likelihood of the model parameters. We then find the parameter values that maximize the likelihood of observing the data, and use the parameter values as estimates of the true values.

Maximum likelihood estimates have nice properties, including consistency and asymptotic normality. Another nice property is that the same numbers that maximize the likelihood also maximize the log of the likelihood.

We can input the parameter estimates and obtain the observed maximum likelihood, as well as the log-likelihood, commonly denoted by ℓ.

AIC and BIC

AIC and BIC are measures of the relative quality of a statistical model given the data. They are calculated as:

- AIC: $-2\ell + 2p$
- BIC: $-2\ell + p \log n$

The model with the smallest value is considered to have the better fit, according to AIC and BIC. BIC penalizes the use of more parameters more strongly than AIC. However, AIC is more consistent in finding the best model given that the true model is not available. Like R^2, there are adjustments for AIC and BIC.

AIC and BIC

```
AIC(model1)
```

```
## [1] 744
```

```
AIC(model2)
```

```
## [1] 715.6
```

```
BIC(model1)
```

```
## [1] 751.9
```

```
BIC(model2)
```

```
## [1] 728.8
```

Likelihood Ratio Test

Consider the following models:

$$\begin{aligned} \text{Model 1: } & Y = \beta_0 + \beta_1 X_1 + \epsilon \\ \text{Model 2: } & Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon \end{aligned}$$

Notice that Model 1 is a constrained model because it assumes $\beta_2 = 0$. In contrast, Model 2 is unconstrained. Because Model 1 can be created from Model 2 by placing constraints on the parameters. Therefore, Model 1 is *nested* within Model 2.

Likelihood Ratio Test

Because the two models are nested, we can use a likelihood ratio test to compare the model fits:

Ho: The constrained model (Model 1) fits the data as well as the unconstrained model (Model 2).

Ha: The constrained model (Model 1) fits the data significantly less well than the unconstrained model (Model 2).

The test statistic is:

$$-2(\ell(\text{Model 1}) - \ell(\text{Model 2}))$$

or -2 times the difference in the log likelihoods. We then compare the test statistic to a chi-square distribution with degrees of freedom $df_2 - df_1$, where df_2 is the degrees of freedom for Model 2, $n - p_2$, and p_2 is the number of parameters in the model. It is likewise for df_1 and Model 1.

Likelihood Ratio Test

```r
require(lmtest)
```

```
## Loading required package: lmtest
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Likelihood Ratio Test

```
logLik(model1)
```

```
## 'log Lik.' -369 (df=3)
```

```
lrtest(model2, model1)
```

```
## Likelihood ratio test
##
## Model 1: prestige ~ education + income + women
## Model 2: prestige ~ education
##   #Df LogLik Df Chisq Pr(>Chisq)
## 1   5   -353
## 2   3   -369 -2  32.4    9.4e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Deviance

Deviance is another quality of fit statistic for comparing models. For nested models, it is proportional to a likelihood ratio test. In R, the output gives a null deviance and a residual deviance.

The null deviance compares the saturated and null models:

$$2(\ell(\text{Saturated Model}) - \ell(\text{Null Model})) \text{ on } df = df_{Null} - df_{Sat}$$

The residual deviance compares the saturated and proposed models:

$$2(\ell(\text{Saturated Model}) - \ell(\text{Proposed Model})) \text{ on } df = df_{Pro} - df_{Sat}$$

NOTE: We want small differences in deviance.

Deviance

```
same.model1 <- glm(prestige ~ education, data=Prestige,
                   family="gaussian")
same.model2 <- glm(prestige ~ education+income+women,
                   data=Prestige, family="gaussian")

resid.dev <- same.model2$deviance
resid.df <- same.model2$df.residual
null.dev <- same.model2$null.deviance
null.df <- same.model2$df.null
```

Deviance

```r
(resid.pvalue <- 1 - pchisq(resid.dev, resid.df))
```

```
## [1] 0
```

```r
(null.pvalue <- 1 - pchisq(null.dev, null.df))
```

```
## [1] 0
```

Deviance

We can also compare the deviance of two models to see the overall effect of the extra predictors.

```r
anova(same.model1, same.model2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: prestige ~ education
## Model 2: prestige ~ education + income + women
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       100       8287
## 2        98       6034  2     2253  1.1e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Goodness of Fit

Like in linear regression, we can look at the sums of squared residuals to get an idea about goodness of fit.

```r
dat <- read.csv("http://www.ats.ucla.edu/stat/data/binary.c
logit.fit1 <- glm(admit ~ gre + gpa, data=dat,
                   family="binomial")
logit.fit2 <- glm(admit ~ gre + gpa + rank, data=dat,
                  family="binomial")
```

Goodness of Fit

The sum of squared deviance residuals is equal to the residual deviance of a model. For GLMs, we can compare the sum of squared Pearson residuals to a chi-square distribution with $n - p$ degrees of freedom. A significant p-value indicates a lack of fit.

```
sum(residuals(logit.fit1)^2)
```

```
## [1] 480.3
```

```
deviance(logit.fit1)
```

```
## [1] 480.3
```

```
pearson.stat <-sum(residuals(logit.fit1, type="pearson")^2)
n <- dim(dat)[1]
1 - pchisq(pearson.stat, n-2)
```

```
## [1] 0.4891
```

Hosmer-Lemeshow Test

The Hosmer-Lemeshow test is a goodness-of-fit test for logistic regression, particularly for risk prediction. It groups similar predicted properties together and calculates a chi-square statistic on the groups. It is more useful when there is more than one predictor and/or continuous predictors.

It is a conservative test, meaning that the probability of rejecting the null hypothesis is smaller even when the alternative is true. It is highly dependent on the grouping and does not work well with one or two categorical predictors.

Sensitivity Analysis

Sensitivity analysis involves assessing the changes in the results and conclusions of a model based on changes in values and assumptions of the model.

The idea is that if a model is sound, changes in values and assumptions of the statistical model should not strongly change its results and conclusions.

Examples include:

- Changing the imputation method for missing data.
- Assuming a different correlation structure for clustered data.
- Identifying sensitive or important variables.
- Changing the error rate of a predictive model.

Sensitivity Analysis

There is a ton of literature on sensitivity analysis. It is a broad field with a lot of different applications.

My Advice: If you are making an assumption you are unsure of, fit one model using the assumption and a second model without using the assumption and compare the model fits. If the model fits are very different (not necessarily statistically significant), then consider using the second model without the assumption.

Take-Home Message

There is no one way to definitively choose one “better” model over another. Take the findings from each method as evidence, not as absolute truth. Once again, do not abandon scientific reasoning when using these tools. Combine your reasoning and your statistical results to make a supported, logical, and well-documented decision.

Active Learning Exercise

Load the Freedman data in the car package. Then use the following code:

```
newFreedman <- Freedman[!is.na(Freedman$density),]
```

You will use the newFreedman data for the active learning exercise. Compare three linear models:

1. crime $\sim$ nonwhite
2. crime $\sim$ nonwhite + population
3. crime $\sim$ nonwhite + population + density

Use the model fit statistics and tests to determine which model is the better model, in terms of model fit and parsimony.