Introduction to Computational BioStatistics with R: generalized linear models

Erik Spence

28 October 2025

Today's slides

Today's slides can be found here. Go to the "Introduction to Computational BioStatistics with R" page, under Lectures, "Generalized Linear Models".

`https://scinet.courses/1391`

Today's class

Today we will continue our adventures in data analysis.

- Verification of models.
- Confounding variables.
- Generalized linear models.

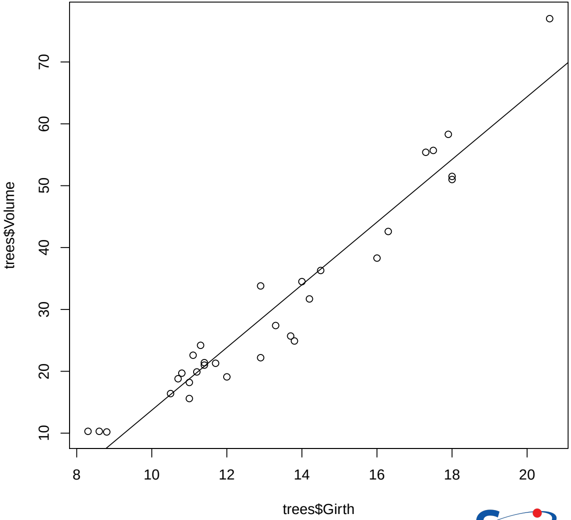
As always, ask questions.

Our linear model

At the end of last class we had created a linear model to describe the relationship between Girth and Volume.

```
>
> model <- lm(Volume ~ Girth,
+        data = trees)
>
> plot(trees$Girth, trees$Volume)
> abline(model)
>
```

How do we assess the quality of our model?



Our linear model, continued

As noted last class, the summary gives important information:

- Information about the null hypothesis $\beta_1 = ... = \beta_n = 0$.
- Information about the individual null hypotheses: $\beta_1 = 0$, $\beta_2 = 0$, etc.

Remember that the significance code only tells you the likelihood that $\beta_i = 0$.

```
> summary(model)
Call:
lm(formula = Volume ~ Girth, data = trees)
Residuals:
    Min      1Q  Median      3Q     Max
 -8.065  -3.107   0.152   3.495   9.587
Coefficients:
              Estimate  Std. Error  t value  Pr(>|t|)
 (Intercept)  -36.9435      3.3651   -10.98  7.62e-12  ***
 Girth          5.0659      0.2474    20.48   < 2e-16  ***
---
Signif.  codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'  0.1 ' ' 1

Residual standard error:  4.252 on 29 degrees of freedom
Multiple R-squared:  0.9353, Adjusted R-squared:  0.9331
F-statistic:  419.4 on 1 and 29 DF, p-value:  < 2.2e-16
>
```

That's great, but we're not done yet

It's always a good idea to do some further analysis of your model before declaring success. There are a few things in particular that should always be done.

- plot the residuals of the model, in various ways,
- examine the statistics of the residuals,
- examine the statistics of the model.

What are residuals? Residuals are the distance between the actual value, and the value predicted by the model, for each data point:

$$R_i = f(x_i) - y_i$$

where f is the model, evaluated at data point x_i, and y_i is the actual value of the dependent variable.

Step 1: plot the residuals

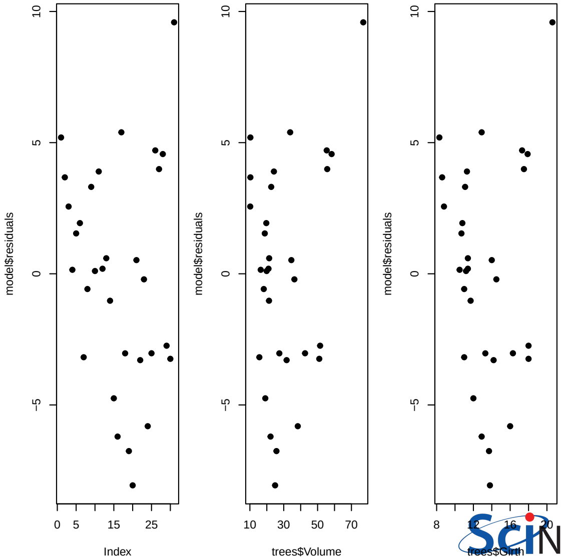
Always plot your residuals. Always.

```
> par(mfrow = c(1, 3))
>
> plot(model$residuals)
> plot(trees$Volume, model$residuals)
> plot(trees$Girth, model$residuals)
>
```

Plot your residuals against everything:

- index,
- against the dependent variables,
- against the independent variables.

You should see a snowstorm. There should be no structure.



Step 1: plot the residuals, continued

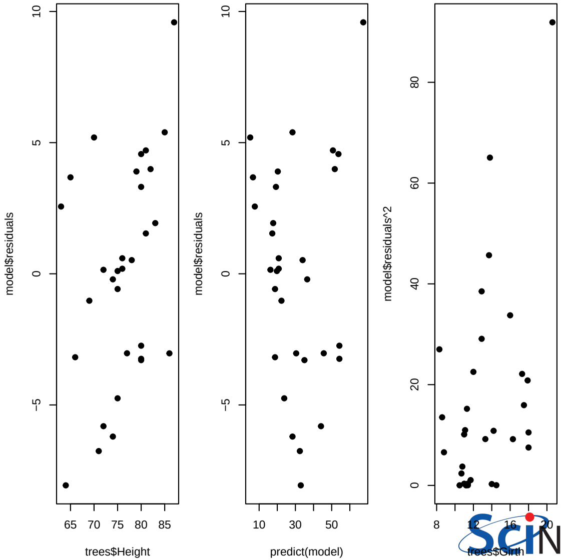
Plot your residuals some more.

```
> plot(trees$Height, model$residuals)
> plot(predict(model), model$residuals)
> plot(model$residuals**2)
>
```

Plot your residuals against everything:

- against independent variables that are not part of the model (if possible),
- against the predicted values,
- plot the square of the residuals.

The residuals squared should be steady in value if the noise is homoskedastic.

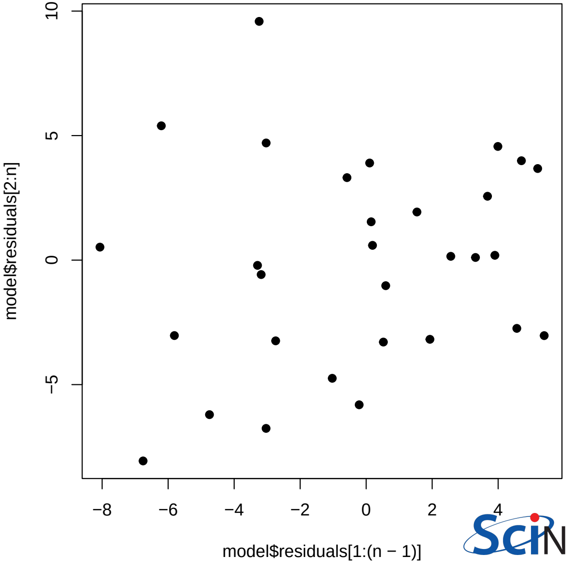


Step 1: plot the residuals, continued more

If your data are related to each other, meaning your x values were measured regularly in space or in time, then it's a good idea to plot the residuals against each other.

```
> par(mfrow = c(1, 1))
>
> n <- nrow(trees)
> plot(model$residuals[1:(n-1)],
+       model$residuals[2:n])
>
```

Again, the residuals should be uncorrelated, with no blobs or bands.

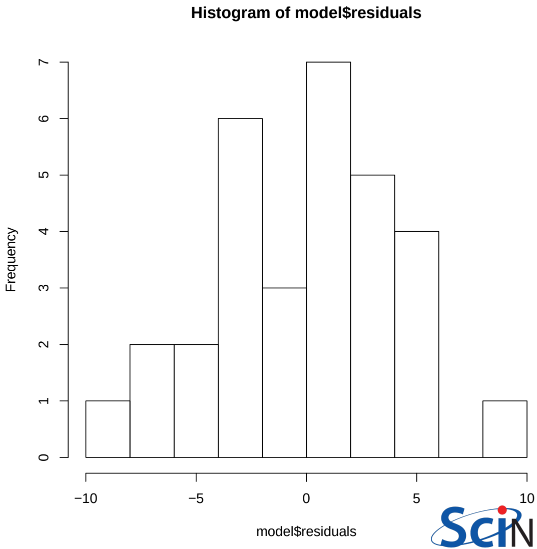


Step 2: plot the residuals via histogram

Always plot a histogram of your residuals. Things to look for:

- The mean should be zero. If your residuals are not centred on zero your model is missing something.
- The distribution should be symmetric. If it's not, it's biased (there 'structure' in the data which has not been captured by the model).
- Distribution should be a Gaussian (an assumption made as part of the fit).

```
> par(mfrow = c(1, 1))
> hist(model$residuals, breaks = 11)
```

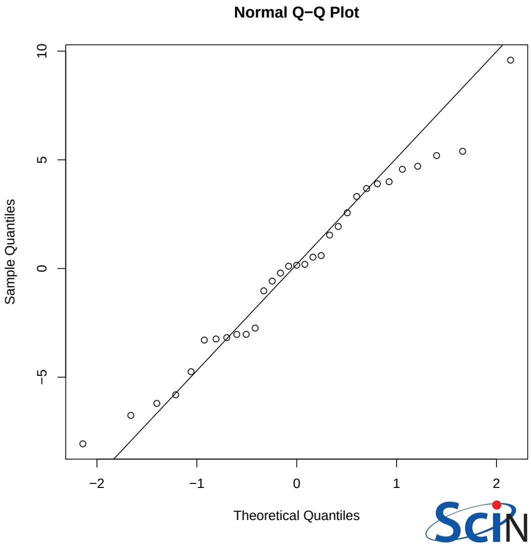

Histogram of model$residuals

Step 3: plot the residuals via Q-Q plot

Plot your residuals on a Q-Q plot.

- A Q-Q plot graphically demonstrates how normally-distributed the residuals are.
- Ideally the residuals should be normally distributed.
- A perfect Gaussian will lie exactly on the line.

```
>
> qqnorm(model$residuals)
> qqline(model$residuals)
>
```



Using R^2

R^2 is a highly mis-used statistical result of building a linear model.

$$R^2 = 1 - \frac{\sum_i \left(y_i - f(x_i)\right)^2}{\sum_i \left(y_i - \bar{y}\right)^2}$$

Some people say that $R^2 =$ (explained variation) / (total variation). This is dodgy at best, as we'll see on the next slide.

```
> summary(model)
Call:
lm(formula = Volume ~ Girth, data = trees)
Residuals:
    Min       1Q   Median      3Q     Max
 -8.065   -3.107    0.152   3.495   9.587
Coefficients:
               Estimate   Std. Error   t value   Pr(>|t|)
 (Intercept)   -36.9435       3.3651    -10.98   7.62e-12   ***
 Girth           5.0659       0.2474     20.48    < 2e-16   ***
---
Signif.  codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'  0.1 ' ' 1

Residual standard error:  4.252 on 29 degrees of freedom
Multiple R-squared:  0.9353, Adjusted R-squared:  0.9331
F-statistic:  419.4 on 1 and 29 DF, p-value:  < 2.2e-16
> 
```

R^2

Recall that R^2 is one minus the ratio of the square of the residuals over the variance in y. Suppose we know the true coefficients, β_0 and β_1, for some model, what would the R^2 be?

$$\begin{aligned} R^2 &= 1 - \frac{SS_{\text{model}}}{SS_{\text{total}}} \\ &= 1 - \frac{\sum_i \left(y_i - f(x_i)\right)^2}{\sum_i \left(y_i - \bar{y}\right)^2} \\ &\vdots \\ &= \frac{\beta_1^2 \mathrm{Var}[\mathrm{x}]}{\beta_1^2 \mathrm{Var}[\mathrm{x}] + \sigma^2} \end{aligned}$$

where $f(x_i)$ is the value of the model for input x_i, $\bar{y}$ is the mean value of y, β_1 is the model coefficient for x_1, σ^2 is the variance of the noise in the data, SS stands for "Sum of Squares", and the sum is over all data points.

Problems with R^2

$$\boldsymbol{R^2} = \frac{\boldsymbol{\beta_1^2}\mathbf{Var}[\mathbf{x}]}{\boldsymbol{\beta_1^2}\mathbf{Var}[\mathbf{x}] + \boldsymbol{\sigma^2}}$$

$\boldsymbol{R^2}$ is not a good measure of fit, nor does it indicate that a model has predictive value. Why?

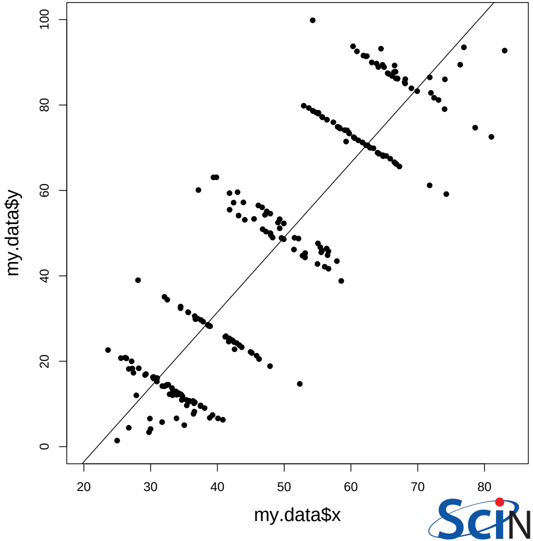
- $\boldsymbol{R^2}$ can be small when the model is correct, if $\boldsymbol{\beta_1^2}\mathbf{Var}[\mathbf{x}]$ is small, or $\boldsymbol{\sigma^2}$ is large.
- $\boldsymbol{R^2}$ can be large when the model is wrong, if $\boldsymbol{\beta_1^2}\mathbf{Var}[\mathbf{x}]$ is large.
- $\boldsymbol{R^2}$ cannot be compared across different data sets.
- $\boldsymbol{R^2}$ is sometimes called the "Coefficient of Determination", which is dumb, dumb, dumb, as it does not explain or 'determine' anything.

It's not clear at all of what use the $\boldsymbol{R^2}$ statistic is.

Simpson’s paradox

Simpson’s paradox (also called the Yule-Simpson effect) is a phenomenon in statistics in which a trend appears in several different groups of data, but disappears or reverses when these groups are combined.

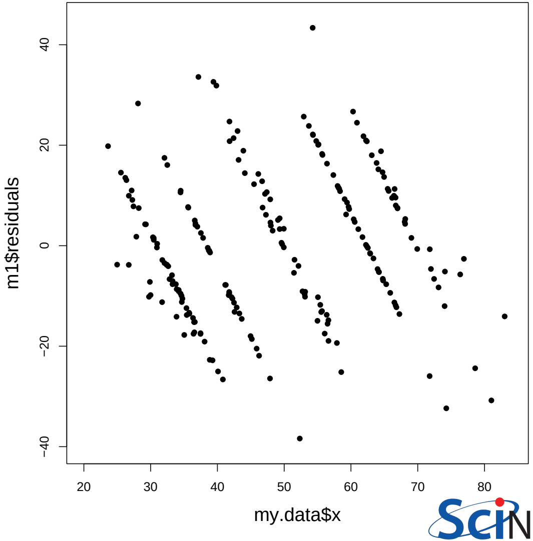
```
> library(datasauRus)
>
> sp <- simpsons_paradox
> my.data <- sp[sp$dataset == ’simpson_2’,]
>
> plot(my.data$x, my.data$y)
> m1 <- lm(y ~ x, data = my.data)
> abline(m1)
```



Simpson's paradox, continued

Plotting the residuals in this case will demonstrate plenty of structure, indicating that there's something wrong with the model, or that, as in this case, the model is incomplete.

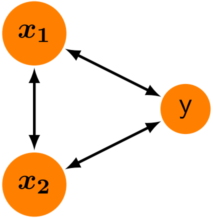
```
>
> plot(my.data$x, m1$residuals)
>
```



Confounding variables

Confounding variables are independent variables that influence both the dependent variable and at least one other independent variable.

- Failing to include, or deal with, these variables can result in an exaggeration or masking of the relationship between the independent and dependent variables.
- Omitting confounding variables forces the model to attribute their affects to other variables, resulting in a "confounding" of the actual relationship between variables.
- This results in "omitted variable bias".



On the right, two independent variables, x_1 and x_2, are correlated with both the dependent variable, y, and each other.

Preventing confounding-variable problems

If you have the opportunity, you can deal with potential problems with confounding variables at the design stage of your study. These are several ways to approach this:

- Matching: create two groups, where each member of one group has an identical counterpart with respect to confounding variables in the other group. A 'propensity score' is often used find matches.
- Restriction: create similar subject groups that are the same with respect to the confounding variable, thus the confounding variable has no effect on the dependent variable, within that group.
- Stratification: estimate the relationship between your dependent and independent variables within groups with different values of confounding variables, then average the estimates.
- Randomization: subjects are randomly, with respect to confounding variables, assigned to different groups. As such the average values of confounding variables should be the same in each group.

Confounding variables, continued

Failing to include your confounding variables in your model may cause problems.

- The missing confounding variable can cause a bias in the estimated coefficients in your model, especially if the two variables are negatively correlated, as the model attempts to compensate for the missing variable.
- The amount of bias depends on the strength of the correlation. The stronger the correlation the stronger the bias.
- Conversely, independent variables that have no correlation to any other independent variables will not be biased at all. Thus the coefficient for this feature will not change as different features are added/removed from the model.
- If you include different combinations of independent variables in your model, and the coefficients change significantly, you are observing "omitted variable bias".

One of the assumptions of our linear model is that the noise is uncorrelated with the data. Any correlations in the data will appear as some sort of structure in your residuals.

Multicollinearity in your data

Suppose you've included all confounding variables in your model in the hopes that the model can deal with them. This may introduce what is called "multicollinearity" into your model.

- This can be an issue in itself.
- Why? Well, the independent variables should be independent. If they're correlated to each other they're obviously not independent.
- If they're not independent, it becomes harder for the model to estimate the coefficient associated with the independent variables, but they don't vary independently of each other.
- As a result,
    - coefficients become unstable depending on which variables are included in the model,
    - the precision of the coefficients is reduced.

If the two variables are positively correlated you may be able to remove one of them from the model. If they're negatively correlated you probably cannot.

Multicollinearity in your data, continued

So how much is multicollinearity a problem?

If the correlations between variables isn't too strong, you can probably safely ignore it.

If it is strong you should probably remove a variable, or possibly centring and scaling the data may help.

Not surprisingly, there's a measure we can use to determine the multicollinearity of the independent variables in our model: "variance inflation factor" (VIF).

- VIF gives the strength of the correlation between independent variables.
- It starts at 1, and has no upper limit.
  - A value near 1 indicates weak correlation.
  - A value from 1 - 5 indicates moderate correlation, but is probably harmless.
  - A value greater than 5 indicates strong multicollinearity.

As you might expect, code to calculate VIF in R already exists.

Measuring multicollinearity

The 'car' library contains the 'vif' function, which calculates the multicollinearity of the independent variables in the model.

The 'ape' library contains many phylogenetic data sets, including data concerning species in the order 'carnivora'.

This model is looking at the relationship between weaning age (WA), gestation length (GL), litter size (LS), female weight (FW), female brain weight (FB) and birth weight (BW).

Examining the coefficients indicates that FB can be dropped from the model.

```
>
> library(ape)
> library(car)
> data(carnivora)
>
> model <- lm(WA ~ GL + LS + FW + FB + BW,
+               data = carnivora)
> vif(model)
      GL       LS       FW        FB       BW
2.267679 1.295047 8.339708 10.930205 3.182454
> >
> model2 <- lm(WA ~ GL + LS + FW + BW,
+               data = carnivora)
> vif(model2)
      GL       LS       FW       BW
1.752621 1.158614 2.197752 2.940201
>
```

Other regression models

There are other types of regression models available:

- Logistic (Logit) Regression: used to fit a categorical variable against a continuous independent variable
- Multinomial Logistic Regression: logistic regression where the dependent variable has more than two outcome categories. If the multiple categories are ordered this is called Ordinal Logistic Regression.
- Generalized Linear Models: multiple independent variables, different link functions and noise families.
- Mixed effect models: regressions that deal with a “random effect” which causes different categories to behave differently.
- Cox regression: analyses factors affecting survival.
- Weighted ordinary least squares.

We will examine Generalized Linear Models today, and Logistic Regression later.

Generalized linear models

The linear model built by lm has some built-in assumptions:

- Normally distributed noise,
- Uncorrelated noise,
- Constant variance of the noise.
- Data is not correlated with the noise.
- Independent variables are independent.

There are situations where these assumptions are dramatically violated. To deal with this, let us examine "Generalized Linear Models". These allow

- Non-normally distributed noise.
- Non-constant variance.

If you find that you have structure in your residuals, it's possible that you need to use a generalized linear model.

Generalized linear models, continued

When should you use a generalized linear model?

- You know that your data should come from a non-linear, non-polynomial distribution (exponential, Poisson, etc).
- You don't know what your distribution should be, and you've got structure in your residuals.

How do generalized linear models work? Let's start with a regular linear model. Assuming the vectors of data are (X, Y), the problem is to find the vector of coefficients β such that

- $E(Y) = X\beta$
- assuming that $Y \sim N(X\beta, \sigma^2)$,

where E is the expectation value, $N(\mu, \sigma^2)$ is the symbol for a normal distribution centred on μ with a standard deviation of σ.

Generalized linear models, continued

As an example, for a log-linked Gaussian GLM, we have

- $\log\left(\boldsymbol{E}(\boldsymbol{Y})\right) = \boldsymbol{X\beta}$,
- which means that $\boldsymbol{E}(\boldsymbol{Y}) = \boldsymbol{e^{X\beta}}$,
- $\boldsymbol{Y \sim N(e^{X\beta}, \sigma^2)}$.

where $\boldsymbol{E}$ is the expectation value, $\boldsymbol{N(\mu, \sigma^2)}$ is the symbol for a normal distribution centred on $\boldsymbol{\mu}$ with a standard deviation of $\boldsymbol{\sigma}$.

Generalized linear models consist of 3 parts:

- A "link" function. A function which transforms the data such that it becomes linear.
- A linear predictor ($\boldsymbol{X\beta}$).
- A probability distribution, which describes the type of noise to be expected in the dependent variable.

Generalized linear models, continued more

There are many possible link functions available. The most common ones are

- Identity: $\boldsymbol{E(Y) = X\beta}$,
- Log: $\boldsymbol{\log\left(E(Y)\right) = X\beta \quad \rightarrow \quad E(Y) = e^{X\beta}}$.
- Logit: $\boldsymbol{\log\left(\frac{E(Y)}{1-E(Y)}\right) = X\beta \quad \rightarrow \quad E(Y) = \frac{1}{1+e^{-X\beta}}}$
- Inverse: $\boldsymbol{1/E(Y) = X\beta \quad \rightarrow \quad E(Y) = 1/\left(X\beta\right)}$

The identity link function results in a standard linear regression. By performing a generalized linear model using this link function, with Gaussian noise, you will get the same result as using the lm function.

Generalized linear models, continued even more

Once a link function has been chosen, the type of error in the data must be chosen. The different error families have different default link functions.

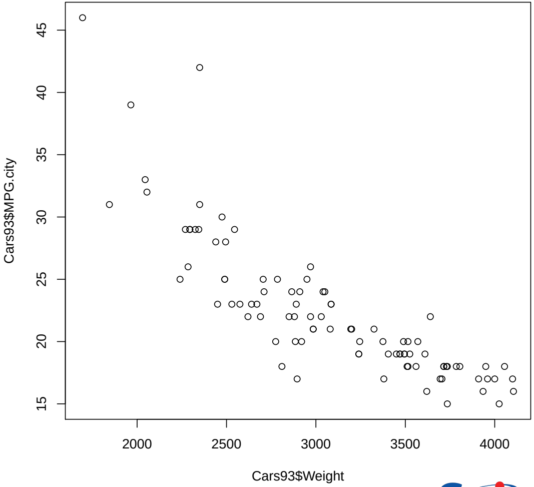
Error family	Default link	Link inverse	Use for:
gaussian	identity	1	Normally distributed error
poisson	log	exp	Counts
binomial	logit	$1/(1+e^{-x})$	Proportions or binary data
gamma	inverse	$1/x$	Continuous data with non-constant error

```
> glm(formula, family = binomial(link = log))
```

GLM example

Consider the Cars93 data set.
Plotting the MPG in the city, versus
Weight, suggests a non-linear relationship.

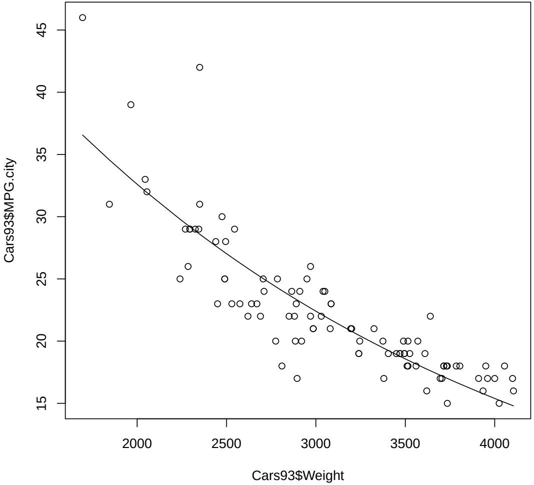
```
>
> library(MASS)
>
> plot(Cars93$Weight, Cars93$MPG.city)
>
```



GLM example, continued

Let's perform a GLM, using Gaussian noise and the log link function.

```
> sorted.weights <- sort(Cars93$Weight)
>
> glm1 <- glm(MPG.city ~ Weight,
+     data = Cars93,
+     family = gaussian(link = "log"))
>
> plot(Cars93$Weight, Cars93$MPG.city)
> lines(sorted.weights,
+     predict(glm1,
+     data.frame(Weight = sorted.weights),
+     type = "response"))
>
```



Summary

We've started looking at the quality of our linear models. Things to remember:

- Plot the residuals! There is important information in there!
  - make sure you get a snow storm!
  - make sure there is no structure, and no clumps, in your residuals.
  - make sure the spread in the data is constant, and not increasing or decreasing.
  - make sure the histogram of the residuals is Gaussian.
- Be aware of confounding variables. Watch out for multicollinearity!
- If the data are not polynomial, or the residuals are not normally distributed, you may need to use a Generalized Linear Model.
- You will likely need to play around with the different noise families and link functions to find one that best works with your data.
- Other types of regression include logistic regression, for fitting categories, and multinomial regression, for multiple dependent variables.