Linear Regression and Hypothesis Tests in R

Shirley Liao

6/23/2017

Linear Regression

Regression is a powerful tool for data analysis, specifically when examining associations between two variables. Unlike hypothesis tests, which may only tell you whether a significant association or difference exists, regressions give statisticians the ability to visualize and information about the strength and direction of relationships. The most well-known type of regression is a linear regression, which may be used when at least one of the variables is continuous.

Exercise 1

Download the dataset "survey" (alternately, you could look at "road" or "ships") after installing (if you haven't already) and attaching the MASS package. You may examine this data using the ?survey command. As we go through the examples today, try performing the same data analysis on your data. After the break today I will ask (or randomly choose) some of you to do a quick presentation of your work, so consider making a few graphs and jotting down the conclusion of your analysis.

Step 1: Determine your question of interest

The first step of a linear regression is the same as the first step of any statistical analysis - formalizing your question of interest. This should be done before you begin looking at the data. Ask yourself:

- Is my data collected to answer my question? Or have I been given a dataset and am trying to find an interesting association within it? Is my data a good fit for my question?
- Am I doing an exploratory (hypothesis-generating) or confirmatory (rigorous, policy-influencing) data analysis?
- What variable is my outcome (dependent)? What variable(s) are my explanatory/predictor/independent variables?
- Do I have a specific relationship between two variables that I am exploring? Or am I equally interested in any possible predictor which has an association with the outcome variable?
- Am I more interested in prediction or association?
- What variables am I controlling for as confounders?

The answers to some of these questions may change your method for analysis. For example, if you are more interested in a predictive model, you may add in any number of covariates which boost your predictive power, whereas if you were examining associations you may seek to be more parsimonious. This lecture is geared towards association studies, because they are more common and require more careful thought in model building and inference.

Step 2: Check data quality and missingness

While it's tempting to jump right into fitting a line and being done with it, the first step must be to perform an exploratory data analysis. This will help you detect problems with your data, possible outliers and data missingness. This last point we will not be touching upon today because the dataset we are working with has already been "cleaned".

```
#install.packages("MASS")
library(MASS)
data(birthwt)
```

This dataset is a study on risk factors associated with low infant birth weight. Look up the covariates and more details about the study using ?birthwt. Our outcome of interest is "bwt", or birth weight in grams. Our primary predictor of interest is mother's age ("age"). We consider all other covariates possible confounders (interested in controlling for them but not necessarily making inference on them), but will not be using the covariate "low", as it is a dichotomization of our outcome variable.

Let's take a look at the summaries of covariates of the birthweight data:

```
summary(birthwt)
```

```
##       low              age             lwt             race
##  Min.   :0.0000   Min.   :14.00   Min.   : 80.0   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:19.00   1st Qu.:110.0   1st Qu.:1.000
##  Median :0.0000   Median :23.00   Median :121.0   Median :1.000
##  Mean   :0.3122   Mean   :23.24   Mean   :129.8   Mean   :1.847
##  3rd Qu.:1.0000   3rd Qu.:26.00   3rd Qu.:140.0   3rd Qu.:3.000
##  Max.   :1.0000   Max.   :45.00   Max.   :250.0   Max.   :3.000
##      smoke            ptl               ht                ui
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.0000
##  Median :0.0000   Median :0.0000   Median :0.00000   Median :0.0000
##  Mean   :0.3915   Mean   :0.1958   Mean   :0.06349   Mean   :0.1481
##  3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :3.0000   Max.   :1.00000   Max.   :1.0000
##       ftv              bwt
##  Min.   :0.0000   Min.   : 709
##  1st Qu.:0.0000   1st Qu.:2414
##  Median :0.0000   Median :2977
##  Mean   :0.7937   Mean   :2945
##  3rd Qu.:1.0000   3rd Qu.:3487
##  Max.   :6.0000   Max.   :4990
```

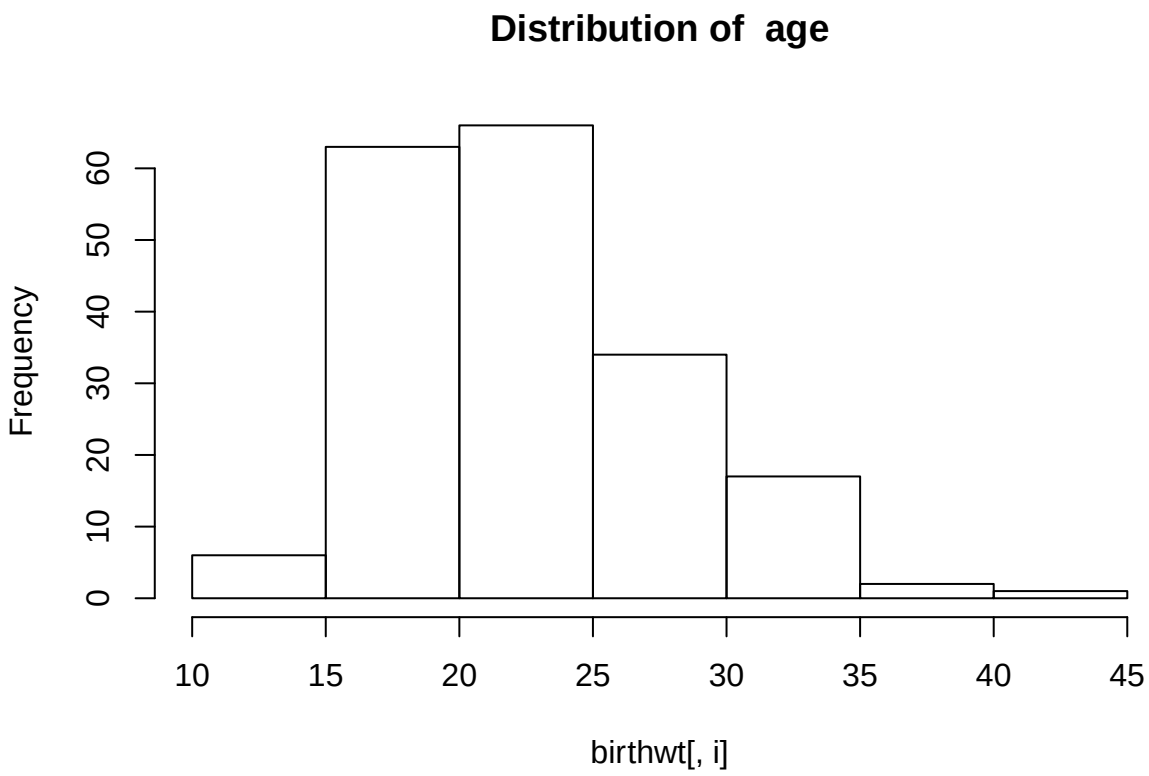
Immediately, we see that while quite a few of the covariates should be categorical, they are in fact entered as integers. Categorical covariates are treated very differently as predictors from continuous variables. While continuous predictors are required to have a linear relationship with the outcome, categorical predictors may fit separate means per category.

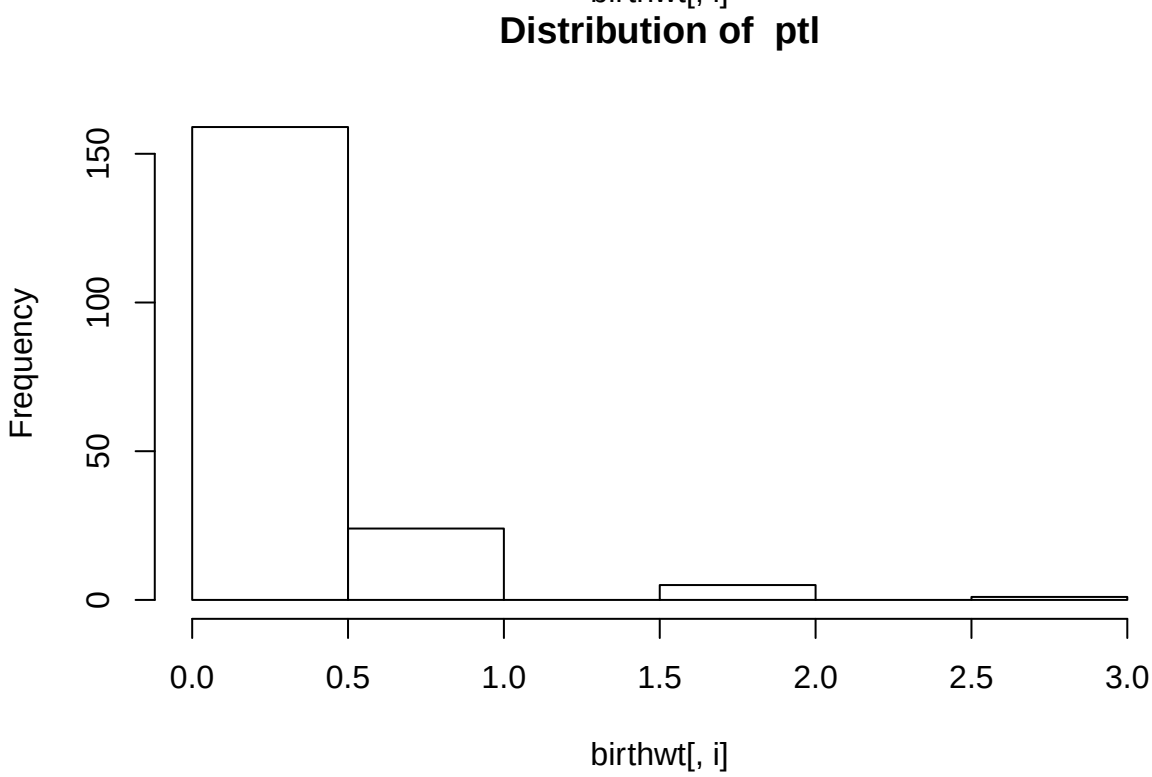
What covariates should be categorical? Manipulate the dataset so they are transformed into categorical variables. Do you notice anything else from the data summary?

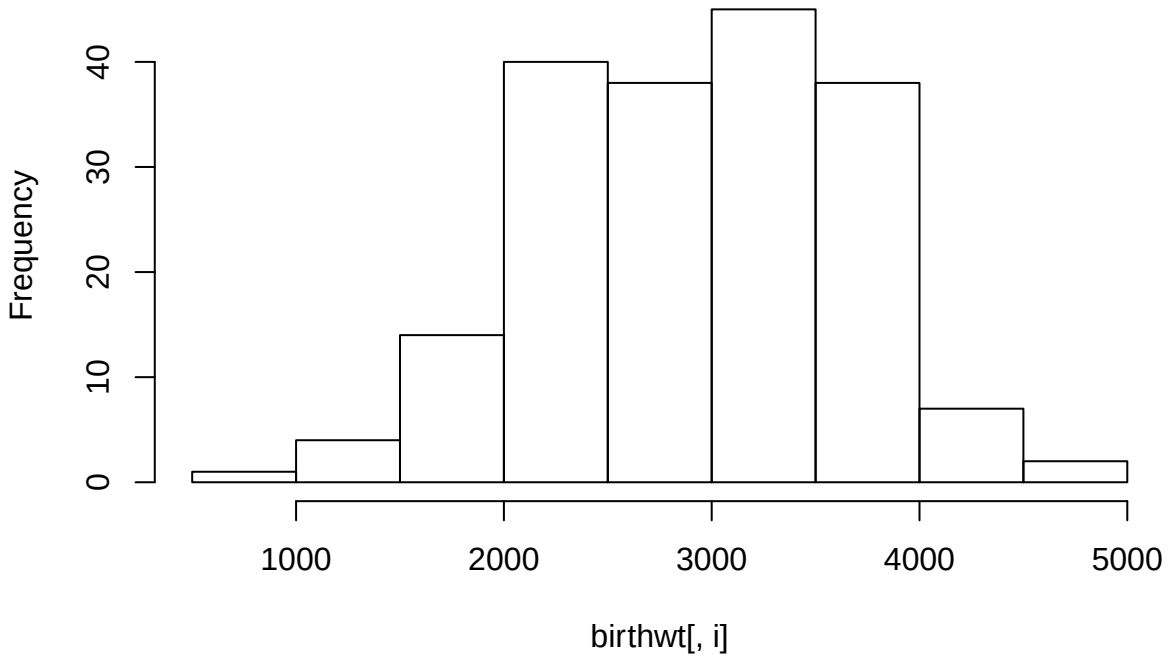
We may also wish to examine the distributions of predictors and outcomes by plotting histograms:

```
#par(c(3,2))

for(i in 1:length(birthwt)){
  if(typeof(birthwt[,i])!="character"){
    hist(birthwt[,i], main = paste("Distribution of ",names(birthwt)[i]))
  }
}
```


Distribution of age
Frequency
0
10
20
30
40
50
60
10
15
20
25
30
35
40
45
birthwt[, i]


Distribution of ptl
Frequency
0
50
100
150
0.0
0.5
1.0
1.5
2.0
2.5
3.0
birthwt[, i]



Normal or approximately normal distributions for covariates and outcome variables are good. If there is evidence of severe skew (particularly in the outcome variable) then transformations may be considered. Log transformations are the most common, but square root or probit transformations are used as well. Keep in mind, however, that transforming data changes your association of interest and may be more trouble than its worth.

Another important thing to check is correlations between predictors. Predictors which have high collinearity have strong associations with each other and generally shouldn't be in the same model because they "explain" the same relationship.

For example, if you had both weight and BMI as possible predictors of heart disease, these two variables are likely to have a strong association with each other. It makes little logical sense to put them both in the same model because one adds little information on top of the other. In this case, we would choose the variable that most accurately reflects what we are trying to control for and simply discard the other.

We may visually inspect associations between predictors using a correlation matrix. This only takes numerical arguments, so we must make a new matrix of only continuous variables:

```
contwt = data.frame(birthwt$age,birthwt$lwt,birthwt$ptl,birthwt$ftv)
cor(contwt)
```

```
##             birthwt.age birthwt.lwt birthwt.ptl birthwt.ftv
## birthwt.age  1.00000000   0.1800732  0.07160639  0.21539394
## birthwt.lwt  0.18007315   1.0000000 -0.14002900  0.14052746
## birthwt.ptl  0.07160639  -0.1400290  1.00000000 -0.04442966
## birthwt.ftv  0.21539394   0.1405275 -0.04442966  1.00000000
```

Two variables are considered collinear if their correlation is larger than 0.7. Typically, collinearity is not worried about with regards to categorical variables.

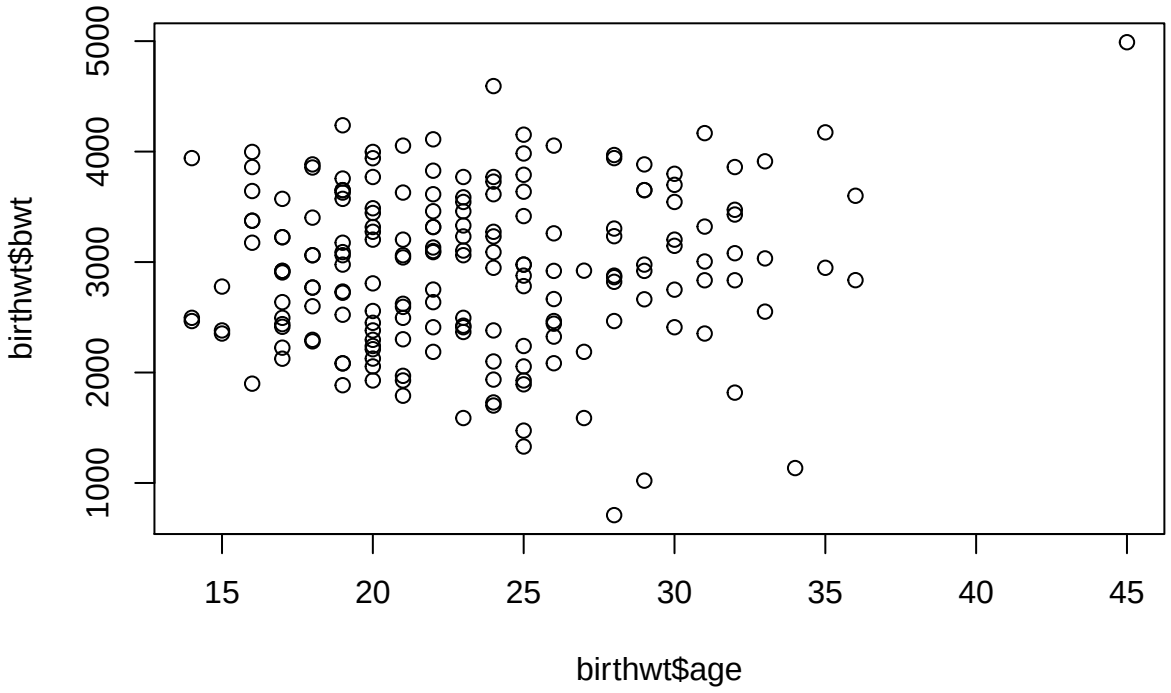
Step 3: Check (some) assumptions

In order to perform linear regression we must make four assumptions:

1. **L**inearity: the outcome has a linear relationship to the predictor(s)
2. **I**ndependence: each observed outcome must be independent of each other observed outcome. This may be violated in cases of repeated measures, spacial variables etc.
3. **N**ormality: outcomes are assumed to have a normal distribution around the linear fit.
4. **E**qual variances: outcomes are assumed to have constant variance around the linear fit.

We may visually check for evidence of non-linearity before we fit the regression:

```
plot(birthwt$age,birthwt$bwt)
```



Does this relationship look approximately linear? Do you see any possible outliers?

Commonly, we do not verify independence except in cases of time series, clustering, spacial data etc.

Step 4: Plot simple linear regression, check more assumptions

Finally, we will perform our first linear regression. We will begin with a simple linear regression of outcome versus our predictor of interest. This will allow us to isolate and consider our relationship of interest:

```
fitsimp = lm(bwt~age,data=birthwt)
summary(fitsimp)
```

```
##
```

```
## Call:
## lm(formula = bwt ~ age, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2294.78  -517.63    10.51   530.80  1774.92
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2655.74     238.86   11.12   <2e-16 ***
## age            12.43      10.02    1.24    0.216
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 728.2 on 187 degrees of freedom
## Multiple R-squared:  0.008157,   Adjusted R-squared:  0.002853
## F-statistic: 1.538 on 1 and 187 DF,  p-value: 0.2165
```

The summary function gives a lot of information about our linear regression. The first thing we should look at is the summary of our coefficient associated with age. In this case, it is 12.43. Thus, for every year the mother is older, our model predicts that her baby will be on average 12.43 grams heavier. This is a **positive** relationship, which means that as x increases, y increases as well. A **negative**, or inverse relationship would result in y decreasing as x increased.

What relationship would x and y have if the coeffcient for x was zero or near zero?

Secondly, we may look at the p-value associated with the age coefficient. If this p-value is significant, this means that the relationship between the predictor and outcome is significantly not "flat". A flat relationship means that values of y are the same at all values of x, thus x does not have a measureable relationship with y. The p-value associated with the age predictor is not significant, indicating that age does not have a particularly strong relationship with the outcome, if any at all.

Towards the bottom, we may read the R^2 and adjusted R^2 of the model, which tells us how much variability the model explains. The adjusted R^2 is the more accepted metric, especially with multiple linear regression, as it puts a penalty on having too many predictors. No matter which one we read, however, age does not seem to be a variable that explains infant birth weight well.

We can create a confidence interval for the coefficient if we wish:

```
confint(fitsimp)
```

```
##                   2.5 %     97.5 %
## (Intercept) 2184.543672 3126.94527
## age           -7.342539   32.20196
```

Keep in mind, however, that this is an unadjusted fit, which does not control for possible confounders. It is entirely possible that our relationship of interest will change once other predictors are added into the model.

Just in case you need it, here are some ways to grab coeffcient information from the model:

```
summary(fitsimp)$coef
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 2655.74447  238.85709 11.118550 2.267545e-22
## age           12.42971   10.02278  1.240146 2.164752e-01
```

```
summary(fitsimp)$coef[1,4]
```

```
## [1] 2.267545e-22
```

You may save other metrics from the output of this linear fit. We may obtain such a list:

```
names(fitsimp)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

```
names(summary(fitsimp))
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

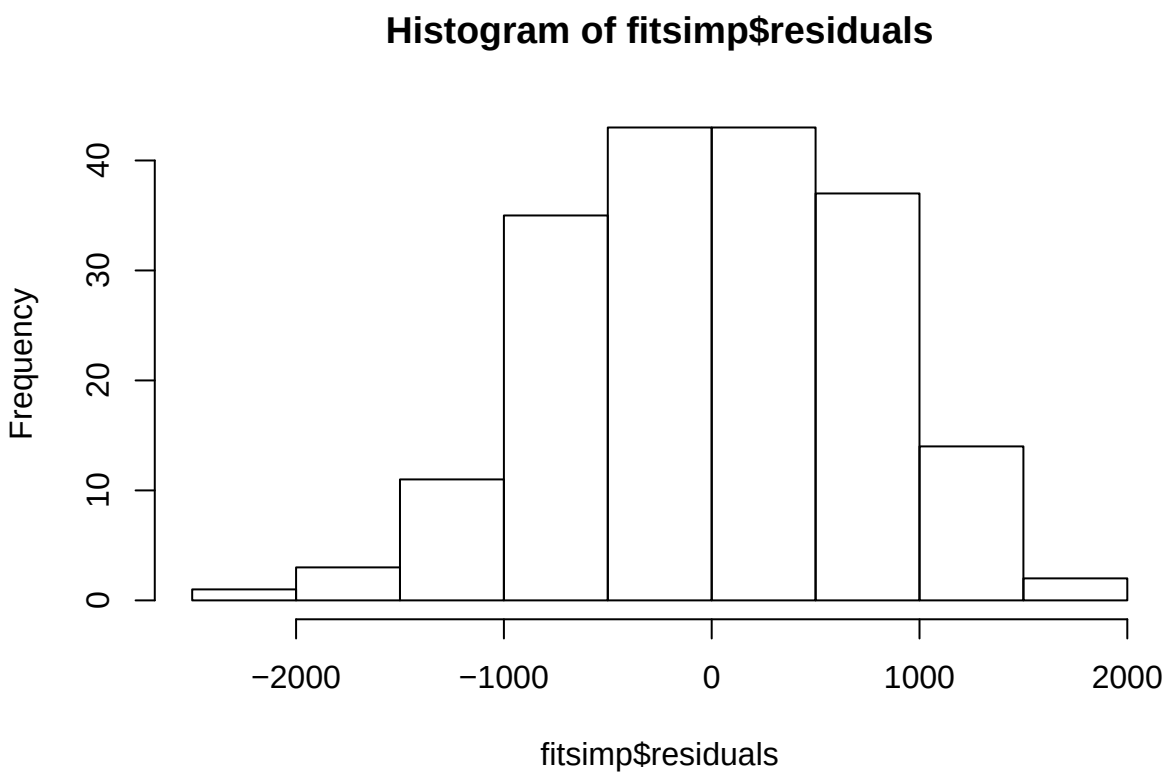
We may use this linear fit to predict outcomes for any age we choose to input (note that all inputs must be in the form of data frames, even single points):

```
predict(fitsimp,data.frame(age=23))
```

```
##        1
## 2941.628
```

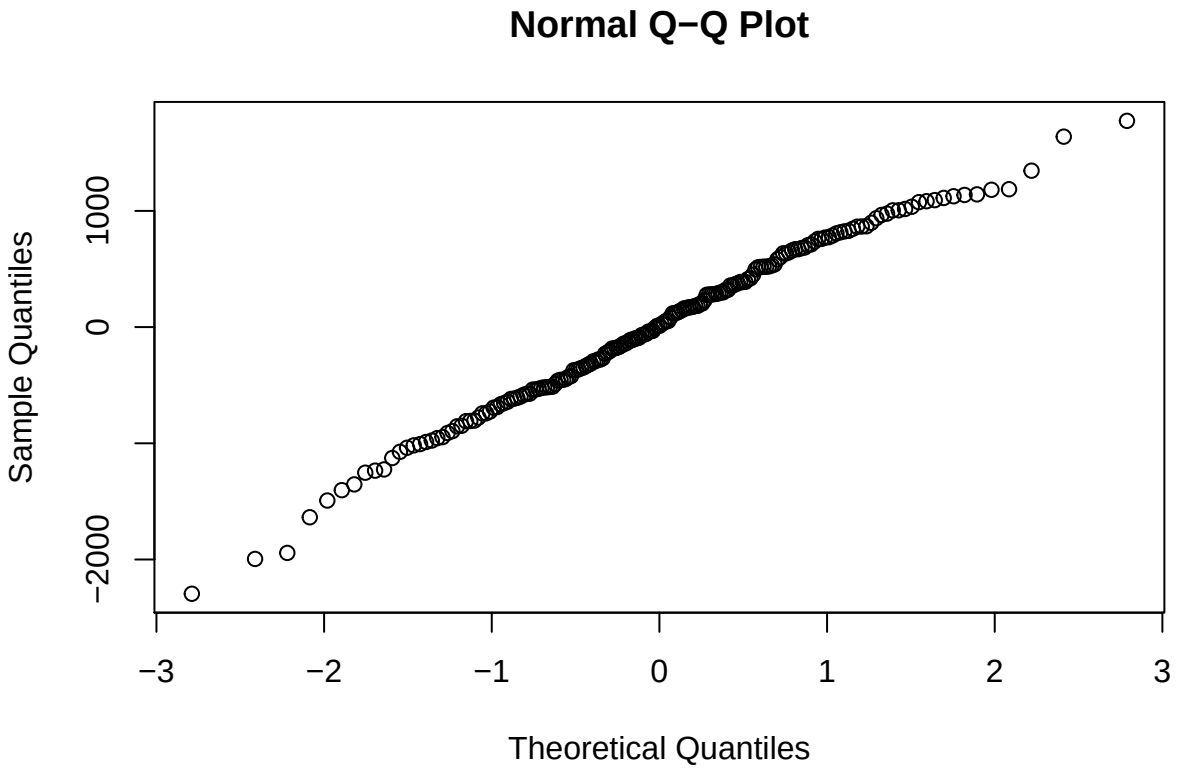
Now, we may continue to check assumptions of our linear model. By plotting a histogram of the residuals (distances between observed points and the predicted line), we may access normality:

```
hist(fitsimp$residuals)
```



This does look pretty normal, but we may want to plot a qq plot to make sure:

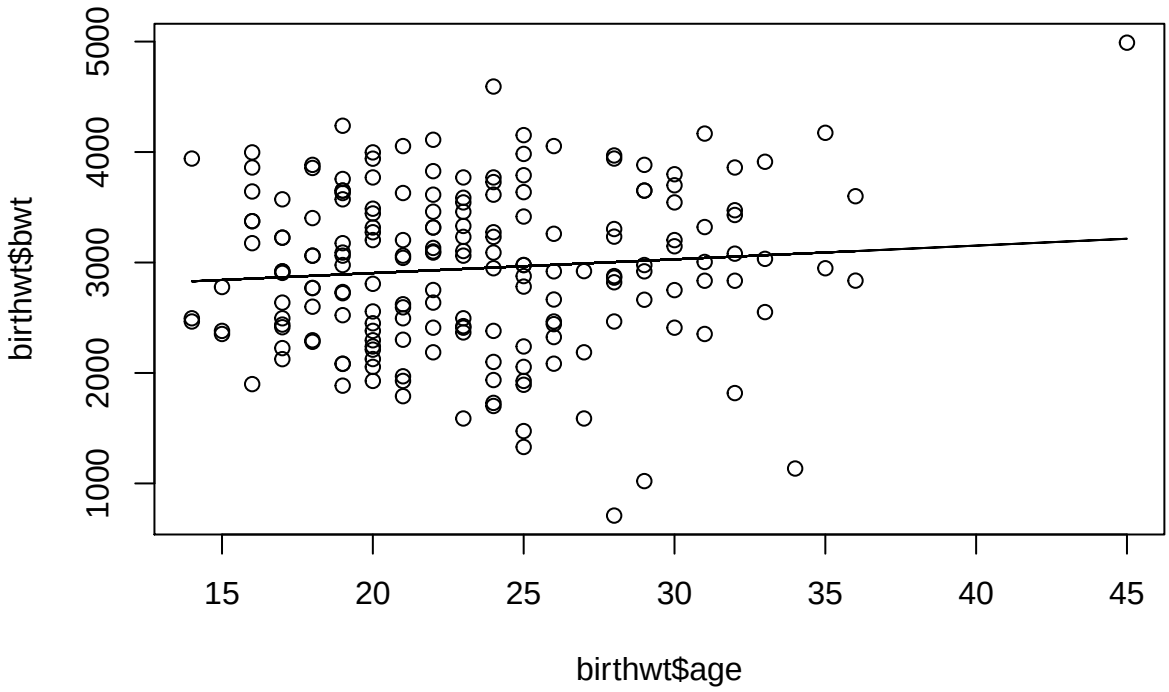
```
qqnorm(fitsimp$residuals)
```



If the residuals form a straight line (which they look to be doing here) when plotted against theoretical quantiles of a normal distribution, this is good evidence that residuals are distributed normally.

If we would like, we can visualize the simple linear regression by plotting the fitted values over our scatterplot:

```
plot(birthwt$age,birthwt$bwt)
lines(birthwt$age,predict(fitsimp))
```



What we should examine here is whether the residuals all look evenly distributed around the line or whether they exhibit signs of "fanning". The former supports our assumption of equal variance, while the latter indicates that this assumption was wrong.

What if my relationship is non-linear?

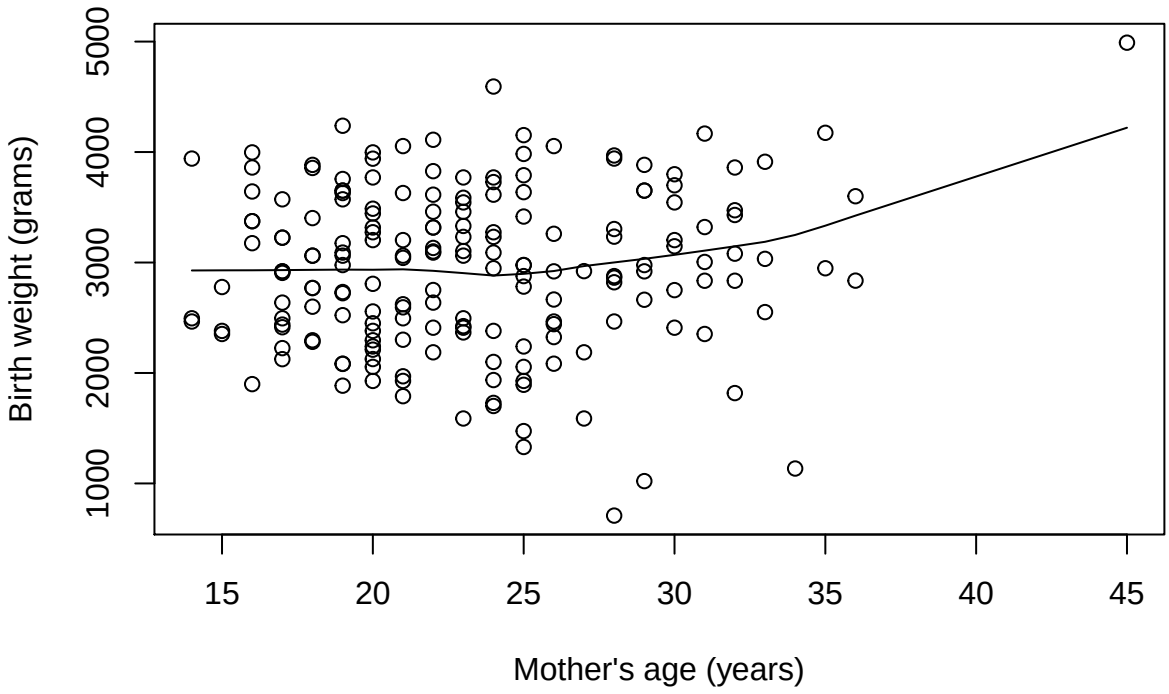
There are many options for complex modelling techniques outside of linear regression. We will explore just a few here:

1. Smoothing

"Smoothing" describes a broad umbrella of techniques which fit a **non-parametric** model to covariate-outcome data. ** Parametric** models are models with consistent "shape". They require assumptions and in return allow you to summarize a data relationship in terms of parameters (for example, β_j in a linear regression summarizes the relationship between covariate x_j and the outcome). ** Non-parametric** models require fewer (if any) assumptions and are much more flexible, but may be hard to summarize.

The ** LOWESS** is a locally weighted running line smoother of Y over a neighborhood of x (which "moves" down the x axis). The algorithm gives more weight to points in the "middle" of the neighborhood.

```
plot(birthwt$age,birthwt$bwt,ylab="Birth weight (grams)",xlab="Mother's age (years)")
lines(lowess(birthwt$age,birthwt$bwt))
```



They are useful for both **visualization** and **prediction**, but does not quantify an association like a regression does. Typically, a LOWESS is performed on at most two predictors.

2. Categorized Covariate

If you suspect that the outcome is not linearly related to a continuous covariate, you may try to turn the covariate into a categorical variable, aka putting groups of similar values into "bins". Be aware that you are changing the interpretation of your regression! A covariate is usually categorized by using either scientific justification (low/medium/high ranges for BMI) or equally spaced/sized intervals.

Perhaps we wish to separate mother's ages into three brackets: under 25, 25-34 and 35 and over.

```
birthwt$catage = rep("25 to 34",dim(birthwt)[1])
birthwt$catage[birthwt$age<25] = "under 25"
birthwt$catage[birthwt$age>34] = "over 34"

fitcat = lm(bwt ~ catage,data=birthwt)

summary(fitcat)
```

```
##
## Call:
## lm(formula = bwt ~ catage, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2177.48  -533.48    35.52   543.30  1649.30
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2886.48      90.19  32.005   <2e-16 ***
## catageover 34    823.12     335.04   2.457   0.0149 *
## catageunder 25    57.22     111.68   0.512   0.6090
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 721.5 on 186 degrees of freedom
## Multiple R-squared:  0.03143,    Adjusted R-squared:  0.02102
## F-statistic: 3.018 on 2 and 186 DF,  p-value: 0.05129
```

3. Polynomial/Nonlinear Terms Adding polynomials or nonlinear terms to a regression model is a ** parametric** strategy for flexible modelling. Common nonlinear terms include:

- Higher order polynomials: quadratic (x^2), cubic (x^3), … etc.
- Fractional polynomials: $x^2, x^{\frac{3}{2}}, x^1, log(x)$...
- Box-Tidwell transform: $xlog(x)$
- Piecewise polynomials: regression splines!

Let's add a quadratic term to our model fit:

```
birthwt$agesq = birthwt$age^2
fitquad = lm(bwt ~ age + agesq, data=birthwt)
summary(fitquad)
```

```
##
## Call:
## lm(formula = bwt ~ age + agesq, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2219.14  -541.47    21.68   592.53  1737.16
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4605.173    817.736   5.632 6.51e-08 ***
## age         -150.858     66.331  -2.274   0.0241 *
## agesq          3.249      1.305   2.490   0.0137 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 718.3 on 186 degrees of freedom
## Multiple R-squared:  0.04014,    Adjusted R-squared:  0.02982
## F-statistic: 3.889 on 2 and 186 DF,  p-value: 0.02215
```

Do you think that the quadratic fit is a good addition to the data?

The major downside to polynomial terms is difficulty in interpretation. Try giving an interpretation of the relationship between age and birth weight, now that the quadratic term is in there.

NOTE: if an upper level term is kept in the model, the corresponding lower level term must also be included, even if it is not significant. This is important for the stability of the model.

In summary, the more flexibility you give a model, the better it will "fit" the data. You'll find that the R^2 increases and residuals will decrease. However, the trade-off will be that relationships will be harder to interpret. Also, prediction power of the model may also suffer if it is "overfit" to your sample of the data. We will talk more about this in the next setction.

Step 5: Variable selection

Variable selection is a large field in statistics, and could probably take up the entire lecture and then some. How involved you wish to become with variable selection depends on your question of interest: 1. Exploratory studies often consider any association that hints at being significant, commonly lowering their p-value for significance to 0.20. 2. Confirmatory studies must be very judicious regarding how many models they fit, due to **multiplicity** issues which increase error. 3. If you are interested in the relationship between one predictor and one outcome and you consider all the other predictors confounders you do not want to make inference on, sometimes it's defensible to simply throw all possible confounders into the model to make sure that they are all controlled for. The downside to this is that you may end up with a very complex model with inflated standard errors (after a point, adding useless predictors will increase your error bounds instead of adding precision). 4. Predictive models only care about metrics such as R^2, AIC, cross-validated prediction error etc. and there are automated methods to maximize these quantities, probably at the cost of throwing in a predictor or two in error.

Consequences of "automated" model building: - **Multiplicity**: the more hypothesis tests you run (this includes the p-values associated with coeffcients), the higher your chance of experiencing a Type 1 error. - **Overfitting**: this is a result of your model being more "fit" to the data than the actual relationship between the predictor and outcome in nature. Your model will be able to "predict" results close to this particular data sample but may predict with wide errors when you input novel predictor values. If you have a large dataset, it may be worth setting some observations aside for the purposes of "out of sample" prediction. - **Complex interpretations**: as a quantitiative scientists, you will often work with non-quantitative scientists or write papers you hope will be read by non-quantitiative scientists. If your modelling is not easily understood and non-intuitive, you will get a lot of confused and angry questions (or they may just take your word for it).

My rule of thumb is **parsimony**. Build the simplest model you can which is nonetheless accurate and answers your desired question. This includes variable selection, but also flexible modelling techniques and other bells and data transformations. Interpretability is just as important as fit in the case where inference is the goal. Furthermore, if you wish to create a scientifically sound model, only make choices/add in covariates which have a logical or scientific reason to be included into the model.

Looking through all of the covariates, I've concluded that they are all worth controlling for

```
fitfull = lm(bwt ~ age + smoke + ftv + ui + ht + ptl + race + lwt,data= birthwt)

summary(fitfull)
```

```
##
## Call:
## lm(formula = bwt ~ age + smoke + ftv + ui + ht + ptl + race +
##     lwt, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1825.26  -435.21    55.91   473.46  1701.20
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2927.962    312.904   9.357  < 2e-16 ***
## age           -3.570      9.620  -0.371 0.711012
## smoke1      -352.045    106.476  -3.306 0.001142 **
## ftv          -14.058     46.468  -0.303 0.762598
## ui1         -516.081    138.885  -3.716 0.000271 ***
## ht1         -592.827    202.321  -2.930 0.003830 **
## ptl          -48.402    101.972  -0.475 0.635607
## race2       -488.428    149.985  -3.257 0.001349 **
```

```
## race3       -355.077    114.753  -3.094 0.002290 **
## lwt            4.354      1.736   2.509 0.013007 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 650.3 on 179 degrees of freedom
## Multiple R-squared:  0.2427, Adjusted R-squared:  0.2047
## F-statistic: 6.376 on 9 and 179 DF,  p-value: 7.891e-08
```

Look at the p-values of these covariates and make a conclusion.

How does adding these covariates to the model change your inference about the relationship between mother's age and infant birth weight?

Compare this model against the simple linear regression we ran earlier using R^2.

We can compare two **nested** models using the ANOVA function. Model A is nested within model B if it is possible to recreate model A entirely by removing or manipulating covariate information in model B.

```
anova(fitsimp,fitfull)
```

```
## Analysis of Variance Table
##
## Model 1: bwt ~ age
## Model 2: bwt ~ age + smoke + ftv + ui + ht + ptl + race + lwt
##   Res.Df      RSS Df Sum of Sq      F  Pr(>F)
## 1    187 99154173
## 2    179 75702317  8  23451856 6.9316 6.1e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A significant p-value for this test indicates that the added variables explain relevant variability in outcome over the simple model. So what's to stop us from fitting all kinds of models and performing multiple ANOVAs? Or fitting all models possible and taking the one with the highest R^2?

Step 6: Interactions

Interactions allow us to examine the effect of two predictors who affect each others' relationship with the outcome. For example, we may wish to examine the interaction of mother's age and smoking status on birth weight, since an older smoking mother has probably been smoking longer and may be smoking more than a younger smoking mother, and this may impact the birth weight.

```
fitinteract = lm(bwt ~ age*smoke,data= birthwt)
summary(fitinteract)
```

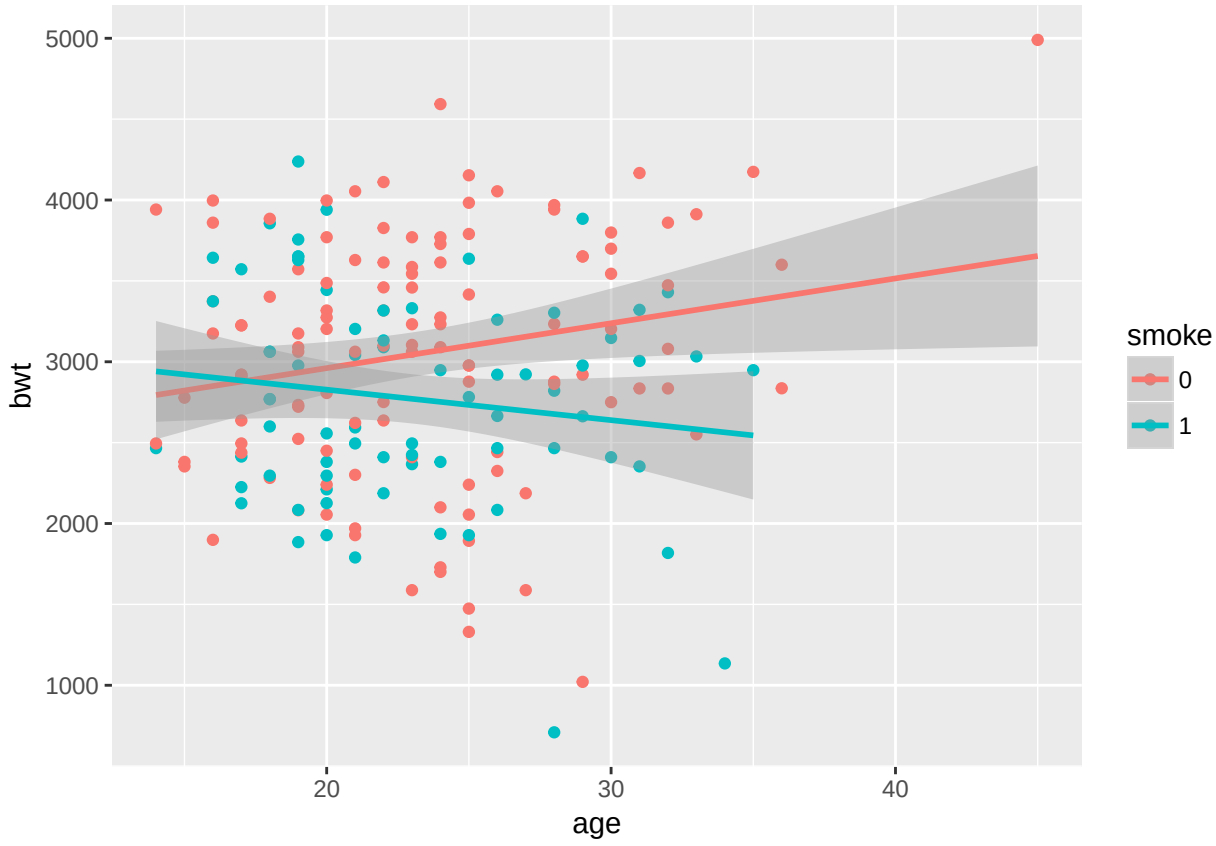
```
##
## Call:
```

```
## lm(formula = bwt ~ age * smoke, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2189.27  -458.46    51.46   527.26  1521.39
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2406.06     292.19   8.235 3.18e-14 ***
## age            27.73      12.15   2.283   0.0236 *
## smoke1        798.17     484.34   1.648   0.1011
## age:smoke1    -46.57      20.45  -2.278   0.0239 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 709.3 on 185 degrees of freedom
## Multiple R-squared:  0.06909,    Adjusted R-squared:  0.054
## F-statistic: 4.577 on 3 and 185 DF,  p-value: 0.004068
```

Let's explore what we just fit graphically:

```
library(ggplot2)
birthwt$smoke = as.character(birthwt$smoke)

ggplot(birthwt, aes(y=bwt,x=age,color=smoke)) +geom_point() + stat_smooth(method="lm")
```



Interpret the relationship between age, smoking and birthweight:

```r
fitinteractfull = lm(bwt ~ age*smoke + ftv + ui + ht + ptl + race + lwt,data= birthwt)
summary(fitinteractfull)
```

```
##
## Call:
## lm(formula = bwt ~ age * smoke + ftv + ui + ht + ptl + race +
##     lwt, data = birthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1851.51  -431.45    62.81   478.45  1467.80
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2549.226    353.944   7.202 1.61e-11 ***
## age           11.913     11.823   1.008  0.31503
## smoke1       649.120    465.638   1.394  0.16504
## ftv           -3.797     46.208  -0.082  0.93461
## ui1         -564.770    139.166  -4.058 7.39e-05 ***
## ht1         -598.714    200.185  -2.991  0.00318 **
## ptl          -31.485    101.177  -0.311  0.75602
## race2       -418.284    151.753  -2.756  0.00645 **
## race3       -316.413    114.875  -2.754  0.00649 **
## lwt            4.232      1.718   2.463  0.01471 *
## age:smoke1   -42.792     19.386  -2.207  0.02857 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 643.4 on 178 degrees of freedom
## Multiple R-squared:  0.2629, Adjusted R-squared:  0.2215
## F-statistic: 6.349 on 10 and 178 DF,  p-value: 2.565e-08
```

What can you conclude from this linear fit?

You could, of course, make interactions of all possible covariates with all other possible covariates, including three-way interactions etc. etc. But the standard trade-off remains: complexity versus fit. If you wish to obtain a model which makes scientific sense, only explore interactions which you can justify with previous research or logic. In most regression studies, interactions are only considered with the predictor of interest, since covariates we are merely "controlling for" have no need for such complex procedures.

Step 7: Outlier detection

Outliers can happen for a variety of reasons, from data entry errors to inklings of a few phenomenon. While it's tempting to immediately discard outliers which do not fit the general trend, if outlier detection and removal is not performed in a defensible manner you may be throwing out good information.

An **influential point** is an outlier whose presence affects the slope of the regression line. A point with

high **leverage** is an outlier on the x (predictor's) axis. A data point with high leverage isn't necessarily an influential point.

There are multiple metrics to measure leverage and influence, one of them is Cook's Distance (CD). The CD of a data point measures how much the slope of the regression line would change if that point was taken out. The threshold of Cook's Distance is $\frac{4}{N}$ where N is our sample size.

```
CD = cooks.distance(fitfull)
birthwt[which(CD>4/(dim(birthwt)[2])),]
```

```
##  [1] low    age    lwt    race   smoke  ptl    ht     ui     ftv    bwt
## [11] catage agesq
## <0 rows> (or 0-length row.names)
```

It seems that there are no influential points as determined by Cook's distance. If we were so interested in examining the outlier (45 yo mother with 4990 gram infant) pointed out before, a common solution would be to run a regression with and without that point present:

```
newbirthwt = subset(birthwt,age<45)

newfitfull = lm(bwt ~ age + smoke + ftv + ui + ht + ptl + race + lwt,data= newbirthwt)

summary(newfitfull)
```

```
##
## Call:
## lm(formula = bwt ~ age + smoke + ftv + ui + ht + ptl + race +
##     lwt, data = newbirthwt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1761.10  -454.81    46.43   459.78  1394.13
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3040.129    309.515   9.822  < 2e-16 ***
## age          -12.111      9.909  -1.222 0.223243
## smoke1      -335.793    104.613  -3.210 0.001576 **
## ftv           -7.247     45.649  -0.159 0.874036
## ui1         -514.842    136.249  -3.779 0.000215 ***
## ht1         -594.324    198.480  -2.994 0.003142 **
## ptl          -32.922    100.185  -0.329 0.742838
## race2       -494.545    147.153  -3.361 0.000951 ***
## race3       -338.940    112.719  -3.007 0.003021 **
## lwt            4.789      1.710   2.801 0.005656 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 638 on 178 degrees of freedom
## Multiple R-squared:  0.2435, Adjusted R-squared:  0.2052
## F-statistic: 6.365 on 9 and 178 DF,  p-value: 8.255e-08
```

Do the estimates for the coefficient of age change enough to justify taking the point out? How would you explain the new fit?

##Logistic Regression

Logistic regressions are utilized when the outcome data is binary. This type of model fit has different assumptions than the linear regression model:

Differences between logistic and linear regressions

1. **Data distributions**: We assume that binary data is distributed according to a binomial/bernoulli distribution, whereas we assume that linear regressions have normal outcomes. The latter, we have to check through plotting histograms, QQ plots etc. but there are no such tests for binary data.
2. **Shape**: The "shape" of a logistic regression is hard to visualize/plot because we are no longer fitting a line to the outcome, we are fitting a line to the logit of the probability of having a positive outcome. Linearity can be assessed by plotting the log odds of the outcome against the predictor of interest, but usually this step is not performed.
3. **Variance structure**: Binomial distributions have variances which depend on the probability of success. With logistic regression, we don't have to assume that the outcome has equal variance in residuals along the fitted line because we know it does not.

Similarities between linear and logistic regression

1. **Independence of data**
2. **No multicollinearity within predictors**
3. **Model is correctly specified**

Fitting a logistic regression

The data we are using comes from a population of women in Phoenix, Arizona of Pima Indian heritage. The purpose of this study is to find possible predictors for diabetes. Examine the data using ?Pima.te.

At this time, we will assume that all exploratory data analysis/examining correlations have been performed already and go onto the model fit:

```
#install.packages("MASS")
library(MASS)
data(Pima.te)
logfit = glm(type~npreg+glu+bp+skin+bmi+ped+age,data=Pima.tr,family=binomial)
summary(logfit)
```

```
##
## Call:
## glm(formula = type ~ npreg + glu + bp + skin + bmi + ped + age,
##     family = binomial, data = Pima.tr)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9830  -0.6773  -0.3681   0.6439   2.3154
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.773062   1.770386  -5.520 3.38e-08 ***
## npreg        0.103183   0.064694   1.595  0.11073
## glu          0.032117   0.006787   4.732 2.22e-06 ***
```

```
## bp          -0.004768   0.018541  -0.257  0.79707
## skin        -0.001917   0.022500  -0.085  0.93211
## bmi          0.083624   0.042827   1.953  0.05087 .
## ped          1.820410   0.665514   2.735  0.00623 **
## age          0.041184   0.022091   1.864  0.06228 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 256.41  on 199  degrees of freedom
## Residual deviance: 178.39  on 192  degrees of freedom
## AIC: 194.39
##
## Number of Fisher Scoring iterations: 5
```

Coefficients of a logistic regression are reported in log odds ratios, which is rather hard to interpret. We should transform them into odds ratios:

```
oddsR <- round(coef(summary(logfit)),4)
oddsR[, "Estimate"] <- round(exp(coef(logfit)),4)
oddsR
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.0001     1.7704 -5.5203   0.0000
## npreg         1.1087     0.0647  1.5949   0.1107
## glu           1.0326     0.0068  4.7319   0.0000
## bp            0.9952     0.0185 -0.2571   0.7971
## skin          0.9981     0.0225 -0.0852   0.9321
## bmi           1.0872     0.0428  1.9526   0.0509
## ped           6.1744     0.6655  2.7353   0.0062
## age           1.0420     0.0221  1.8643   0.0623
```

Let's interpret the association between number of pregnancies and diabetes. The odds ratio of this association is 1.11, which means that the odds of being diagnosed with diabetes when a woman has been pregnant once is 1.11 times (or 11% higher) than when a woman has never been pregnant. This relationship is the same between women who have had 2 children versus 1 child, 3 children versus 2 children etc, after controlling for glucose concentration, blood pressure, skin fold thickness, bmi a pedigree score and age.

Calibrating a logistic regression

Because of the different distribution underlying the logistic regression, we can no longer rely on the same metrics which we used with linear regression to determine whether the model has a good fit. For example, the R^2 doesn't make sense regarding logistic regression because there is not the same "variability" around a linear fit.

The most widely-used method of determining goodness-of-fit of logistic regressions is with a ROC (reciever operating characteristic) curve. Before we begin plotting ROC curves, we must first define some terms.

ROC curves attempt to measure **discrimination**, which is the model's ability to assign high probabilities to "cases" and low probabilities to "controls". Remember that the logistic regression model fit can be made to predict probabilities from our observed covariate values. Do this now and plot a histogram of these probabilities:

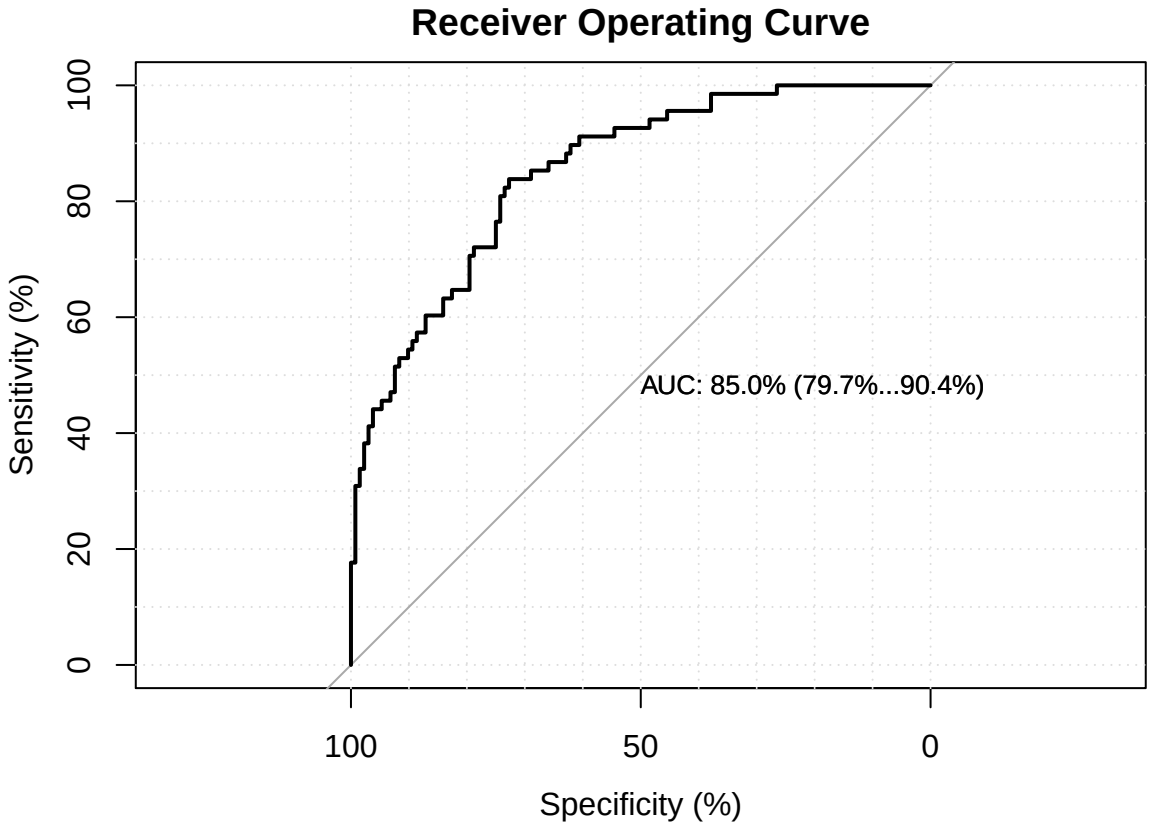
Does this look like a model with high discrimination?

Imagine you wanted to ‘’draw a line” somewhere along this histogram, where you will diagnose everyone above this line with diabetes and everyone below this line as not having diabetes. Your **sensitivity** refers to our’‘true positive rate” (probability of diagnosing diabetes when the person actually has diabetes) and **specificity** refers to our”true negative rate” (probability of diagnosing no diabetes when the patient does not actually have diabetes).

What is the relationship between discrimination and sensitivity/specificity?

The ROC curve is a plot of sensitivity versus 1 - specificity (what is this?) along a continuous series of diagnosis boundaries.

```
#install.packages("LogisticDx")
library(LogisticDx)
gof(logfit)
```




```
##       chiSq  df    pVal
## PrI  177.02 192 0.77369
## drI  178.39 192 0.75092
## PrG  177.02 192 0.77369
## drG  178.39 192 0.75092
## PrCT 177.02 192 0.77369
## drCT 178.39 192 0.75092
```

```
##                       val df     pVal
## HL chiSq          6.17539  8 0.627593
## mHL F             0.97346  9 0.463365
## OsRo Z                 NA NA 0.435219
## SstPgeq0.5 Z      0.83952 NA 0.401178
## SstPl0.5 Z        0.75988 NA 0.447329
## SstBoth chiSq     1.28220  2 0.526712
## SllPgeq0.5 chiSq 0.84665  1 0.357503
## SllPl0.5 chiSq   0.62304  1 0.429919
## SllBoth chiSq    5.08343  2 0.078731 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Visually, a "good" ROC curve will have high sensitivity and low 1 - specificity at any cut-off we choose. A model with perfect discrimination (1s for all cases, 0 for all controls) the ROC curve would be very close to the axis. With bad discrimination, the ROC curve would lie on the middle line, representing that at any choice for diagnosis boundary, there is a 50/50 chance of getting the correct diagnosis (same amount of information as a random coin flip). The ROC curve does not go below this boundary.

A one-number summary of the ROC curve can be found in the AUC, or C-statistic. This measures the area underneath the ROC curve as a proportion of total area in the graph. When c = 0.5 it indicates that the model assigns similar probabilities to observations who experience events and observations who do not (low discrimination). When c = 1, it indicates that the model always assigns higher probabilities to observations who experience events (high discrimination). Any c statistic higher than 0.7 is acceptable, but closer to 1 is better.

Logistic regression exercise

1. Upload the data "mes" (you must first have installed and attached "LogisticDX") using the data() function. Use ?mes to get some background on the data.
2. Your outcome variable will be "ME", which is a factor with three levels. Make a new variable called "ME2" which dichotomizes this information into "never" and "has gotten a mammography at some time". Attach this variable as a column onto mes. Change SYMPT and DETEC to numerical variables.
3. Run a logistic regression with all of the variables. What variables are significant? What does this mean?
4. Transform the coeffcients into odds ratios. Interpret one of the relationships.
5. Examine the discrimination of this model using a ROC curve and associated C-statistic. Is this a well-fit model?