Linear Models. Inference and Interpretation

Nik Bear Brown

In this lesson we will build and interpreting simple models with linear regression. We then learn to interpret the diagnostics and quality of our models. We expand our modeling tool box with multivariate analysis. We finish by using logistic regression to predict probabilities and categories. .

Rationale: A model is a representation of reality albeit highly simplified. Building a statistical model involves constructing a mathematical description of some real-world phenomena that accounts for the uncertainty and randomness involved in that system. All statistical hypothesis tests and all statistical estimators are derived from statistical models. Statistical models are a fundamental part of the foundation of statistical inference, data analysis and machine learning.

## Additional packages needed

To run the code in M02\_Lesson\_02.Rmd you may need additional packages.

* If necessary install ggplot2 package.

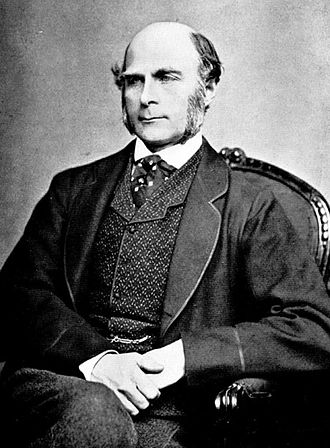
`install.packages("ggplot2");

require(ggplot2)

## Loading required package: ggplot2

## Data

We will be using [Francis Galton's](https://en.wikipedia.org/wiki/Francis_Galton) analysis of the heights of sons and fathers. Heights of sons of both tall and short fathers appeared to “revert” or “regress” to the mean of the group.

  
*Francis Galton 1850s*

Galton created the statistical concept of correlation and widely promoted regression toward the mean. Regression toward the mean become analysis be [Regression analysis Wikipedia](https://en.wikipedia.org/wiki/Regression_analysis))  
Feel free to tweet questions to [@NikBearBrown](<https://twitter.com/NikBearBrown>)

# Load our data  
  
data\_url <- 'http://nikbearbrown.com/YouTube/MachineLearning/M02/Galton\_heights\_sons\_and\_fathers.csv'  
galton <- read.csv(url(data\_url))

## The Modeling process

A. Formula (i.e. what amd which ) B. Fit (i.e. Estimate the unknown parameters for the model.) C. Analysis of fit (i.e. how good is the model)

We often create many models so we *store & explore*. That is, make models and save them as variables so we can compare the various iterations of the modeling process.

## A. Formula

|  |  |
| --- | --- |
| f(x) | x |
| response | ~ predictor |
| response | ~ explanatory |
| dependent | ~ independent |
| outcome | ~ predictor |
| forecast | ~ predictor |
| regressand | ~ regressor |
| explained | ~ explanatory |

head(galton)

## Family Father Mother Gender Height Kids  
## 1 1 78.5 67.0 M 73.2 4  
## 2 1 78.5 67.0 F 69.2 4  
## 3 1 78.5 67.0 F 69.0 4  
## 4 1 78.5 67.0 F 69.0 4  
## 5 2 75.5 66.5 M 73.5 4  
## 6 2 75.5 66.5 M 72.5 4

str(galton)

## 'data.frame': 898 obs. of 6 variables:  
## $ Family: int 1 1 1 1 2 2 2 2 3 3 ...  
## $ Father: num 78.5 78.5 78.5 78.5 75.5 75.5 75.5 75.5 75 75 ...  
## $ Mother: num 67 67 67 67 66.5 66.5 66.5 66.5 64 64 ...  
## $ Gender: Factor w/ 2 levels "F","M": 2 1 1 1 2 2 1 1 2 1 ...  
## $ Height: num 73.2 69.2 69 69 73.5 72.5 65.5 65.5 71 68 ...  
## $ Kids : int 4 4 4 4 4 4 4 4 2 2 ...

names(galton)

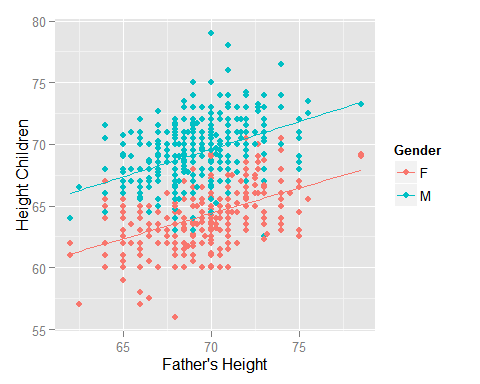
## [1] "Family" "Father" "Mother" "Gender" "Height" "Kids"

Galton's Regression to the Mean idea:

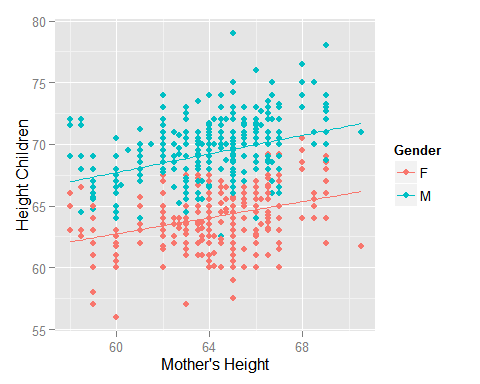
Tall fathers will have tall sons, but the height of the sons will be closer to the mean of the current adult male population. The same holds for short fathers and their short sons who, nevertheless, tend to be more average than their father.

In other words, an exceptionally tall person, say 7 feet, would expect to have tall children but not necessarily 7 feet ) as this is exceptional for normally distibuted data). Contrawise, an exceptionally short person, say 5 feet, would expect to have short children but not necessarily 5 feet ) as this is exceptional for normally distibuted data).

qplot(Father,Height,data=galton,color=Gender) + geom\_smooth(method=lm,se=FALSE) + ylab("Height Children") + xlab("Father's Height")



qplot(Mother,Height,data=galton,color=Gender) + geom\_smooth(method=lm,se=FALSE) + ylab("Height Children") + xlab("Mother's Height")



sons<-subset(galton, Gender=="M", select=c(Height,Father,Mother))

xxx

It seems that, in general, if the father is taller the child is taller. Also if the mother is taller the child is taller.

Eyeballing both slopes look roughly equal and about 0.4. So we'd expect from this graph a relation (i.e. formula or model) for the son's height.

and for the daughter's height.

Choosing predictors is typically done in 3 ways:

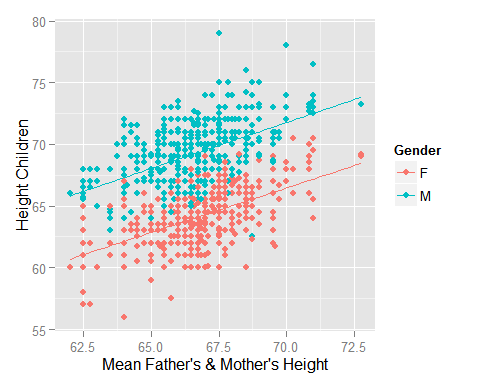
1. Theory / domain knowledge
2. EDA - exploritory data analysis.
3. Fitting various combinations of outcome and predictors and selecting the best fits (r.g. Stepwise regression)

Here Galton was testing an idea/theory. How dependent is a son's height on his father's height? Domain knowledge would also ask how dependent is a son's height on his mother's height? (I'm not sure is Dalton asked this question or not. We will ask this as well as whether tall mother's and father's tend to have even taller children (i.e. multivariate regression))

galton$MF<-((galton$Mother+galton$Father)/2.0)  
head(galton)

## Family Father Mother Gender Height Kids MF  
## 1 1 78.5 67.0 M 73.2 4 72.75  
## 2 1 78.5 67.0 F 69.2 4 72.75  
## 3 1 78.5 67.0 F 69.0 4 72.75  
## 4 1 78.5 67.0 F 69.0 4 72.75  
## 5 2 75.5 66.5 M 73.5 4 71.00  
## 6 2 75.5 66.5 M 72.5 4 71.00

qplot(MF,Height,data=galton,color=Gender) + geom\_smooth(method=lm,se=FALSE) + ylab("Height Children") + xlab("Mean Father's & Mother's Height")

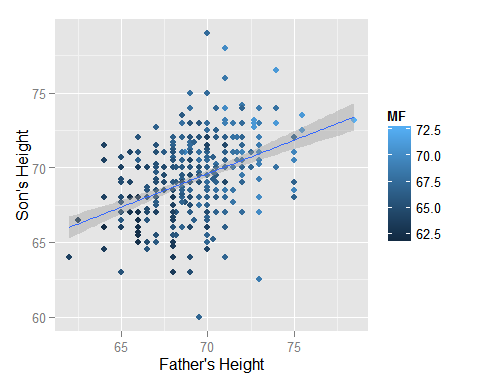


The slope looks a little steeper than with either the father's or mother's heights alone. Roughly the equations below:

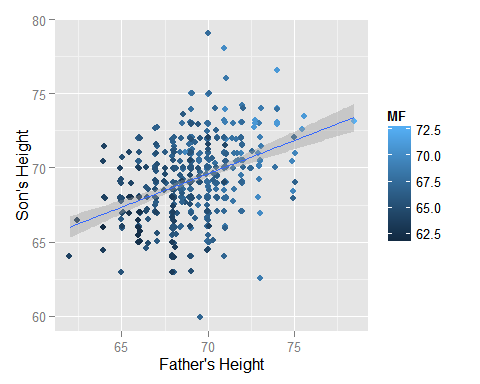
and

Let's work with just the son's height's (as Galton did)

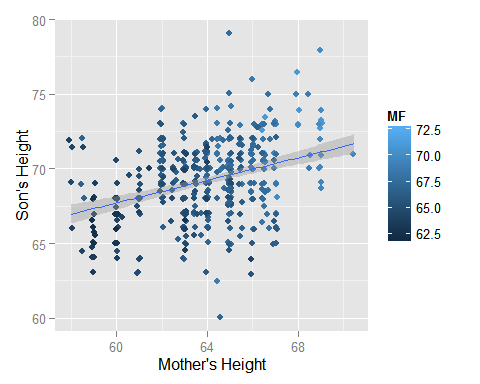
sons<-subset(galton, Gender=="M", select=c(Height,Father,Mother,MF,Kids))  
qplot(Father,Height,data=sons, color=MF) + geom\_smooth(method=lm) + ylab("Son's Height") + xlab("Father's Height")



qplot(Father,Height,data=sons, position = "jitter", color=MF) + geom\_smooth(method=lm) + ylab("Son's Height") + xlab("Father's Height")



qplot(Mother,Height,data=sons, position = "jitter", color=MF) + geom\_smooth(method=lm) + ylab("Son's Height") + xlab("Mother's Height")



head(sons)

## Height Father Mother MF Kids  
## 1 73.2 78.5 67.0 72.75 4  
## 5 73.5 75.5 66.5 71.00 4  
## 6 72.5 75.5 66.5 71.00 4  
## 9 71.0 75.0 64.0 69.50 2  
## 11 70.5 75.0 64.0 69.50 5  
## 12 68.5 75.0 64.0 69.50 5

## R's lm() function

*Description*

lm is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although aov may provide a more convenient interface for these).

*Usage*

lm(formula, data, subset, weights, na.action, method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE, singular.ok = TRUE, contrasts = NULL, offset, ...)

*Model Syntax in R*

* fx ~ x
* y ~ x

e.g. Height ~ Father or Height ~ Mother

and for multivariate models

* fx ~ x1 + x2

e.g. Height ~ Father + Mother

Note that many, many models in R use this syntax

* m1 <- lm(Income ~ Father + Mother, data=sons)
* m2 <- lm(Income ~ Father + Mother,data=sons)
* m3 <- aov(Income ~ Father,data=sons)

m\_sons\_fathers<-lm(Height ~ Father,data=sons)  
m\_sons\_fathers

##   
## Call:  
## lm(formula = Height ~ Father, data = sons)  
##   
## Coefficients:  
## (Intercept) Father   
## 38.2589 0.4477

Which translates in to the equations below:

## P-value, t-statistic and standard error

summary(m\_sons\_fathers)

##   
## Call:  
## lm(formula = Height ~ Father, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.3774 -1.4968 0.0181 1.6375 9.3987   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 38.25891 3.38663 11.30 <2e-16 \*\*\*  
## Father 0.44775 0.04894 9.15 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.424 on 463 degrees of freedom  
## Multiple R-squared: 0.1531, Adjusted R-squared: 0.1513   
## F-statistic: 83.72 on 1 and 463 DF, p-value: < 2.2e-16

The t-statistic is the coefficient divided by its standard error. For example, a of 38.2 divided by a standard error of 3.4 would give a t value of 11.2.

The standard error is an estimate of the standard deviation of the coefficient. It can be thought of as the spread between fitted and actual values.

If a coefficient is large compared to its standard error, then we can reject the hypothesis that = 0. Intuitively we can think of this if the slope is not small and there is a not much spread between fitted and actual values then we can be confident that the true slope is not 0.

A t-statistic (t value) of greater than 2 in magnitude, corresponds to p-values less than 0.05.

The p-value is a function of the observed sample results (a statistic) that is used for testing a statistical hypothesis. Before the test is performed, a threshold value is chosen, called the significance level of the test, traditionally 5% or 1% and denoted as .

If the p-value is equal to or smaller than the significance level (), it suggests that the observed data are inconsistent with the assumption that the null hypothesis is true and thus that hypothesis must be rejected (but this does not automatically mean the alternative hypothesis can be accepted as true). When the p-value is calculated correctly, such a test is guaranteed to control the Type I error rate to be no greater than .

from [P-value](https://en.wikipedia.org/wiki/P-value)

## Mothers,Mean(Father,Mother) and Kids as predictors

Of course a son's height could be effected by the mothers height. We can also model questions like do families have taller kids as they get better at it (i.e. have more kids).

m\_sons\_mothers<-lm(Height ~ Mother,data=sons)  
summary(m\_sons\_mothers)

##   
## Call:  
## lm(formula = Height ~ Mother, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.4152 -1.6686 0.2081 1.7081 9.3964   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 45.11877 3.16281 14.265 < 2e-16 \*\*\*  
## Mother 0.37669 0.04938 7.628 1.37e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.483 on 463 degrees of freedom  
## Multiple R-squared: 0.1116, Adjusted R-squared: 0.1097   
## F-statistic: 58.19 on 1 and 463 DF, p-value: 1.367e-13

m\_sons\_mfs<-lm(Height ~ MF,data=sons)  
summary(m\_sons\_mfs)

##   
## Call:  
## lm(formula = Height ~ MF, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.5364 -1.5364 0.1963 1.5217 9.0914   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.67152 4.12744 4.766 2.52e-06 \*\*\*  
## MF 0.74425 0.06197 12.011 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.3 on 463 degrees of freedom  
## Multiple R-squared: 0.2376, Adjusted R-squared: 0.2359   
## F-statistic: 144.3 on 1 and 463 DF, p-value: < 2.2e-16

m\_sons\_kids<-lm(Height ~ Kids,data=sons)  
summary(m\_sons\_kids)

##   
## Call:  
## lm(formula = Height ~ Kids, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.1138 -1.7779 0.1982 1.6624 9.8862   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 69.8973 0.3003 232.732 <2e-16 \*\*\*  
## Kids -0.1119 0.0460 -2.433 0.0153 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.618 on 463 degrees of freedom  
## Multiple R-squared: 0.01263, Adjusted R-squared: 0.0105   
## F-statistic: 5.922 on 1 and 463 DF, p-value: 0.01533

So our model equations are below:

or rounded

Note that we can estimate $$ variance in the data.

sd(sons$Height)

## [1] 2.631594

So if we wanted to predict a sons heights based on the fathers, $ son\_{height} = 38 + 0.45 x\_{father} 2.6 $ and the fathers height is 77 inches the we'd expect 67% of his chidlern to be 72.65 2.6 inches (i.e. 1 SD).

## Does both the Mothers and fathers heights matter?

The mean father's & mother's height seems to have more of an effect than either the father's or mother's height. The slope of 0.74 is nearly the slope of the father's 0.45 and mother's 0.38 combined.

summary(m\_sons\_fathers)

##   
## Call:  
## lm(formula = Height ~ Father, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.3774 -1.4968 0.0181 1.6375 9.3987   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 38.25891 3.38663 11.30 <2e-16 \*\*\*  
## Father 0.44775 0.04894 9.15 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.424 on 463 degrees of freedom  
## Multiple R-squared: 0.1531, Adjusted R-squared: 0.1513   
## F-statistic: 83.72 on 1 and 463 DF, p-value: < 2.2e-16

summary(m\_sons\_mothers)

##   
## Call:  
## lm(formula = Height ~ Mother, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.4152 -1.6686 0.2081 1.7081 9.3964   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 45.11877 3.16281 14.265 < 2e-16 \*\*\*  
## Mother 0.37669 0.04938 7.628 1.37e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.483 on 463 degrees of freedom  
## Multiple R-squared: 0.1116, Adjusted R-squared: 0.1097   
## F-statistic: 58.19 on 1 and 463 DF, p-value: 1.367e-13

summary(m\_sons\_mfs)

##   
## Call:  
## lm(formula = Height ~ MF, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.5364 -1.5364 0.1963 1.5217 9.0914   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.67152 4.12744 4.766 2.52e-06 \*\*\*  
## MF 0.74425 0.06197 12.011 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.3 on 463 degrees of freedom  
## Multiple R-squared: 0.2376, Adjusted R-squared: 0.2359   
## F-statistic: 144.3 on 1 and 463 DF, p-value: < 2.2e-16

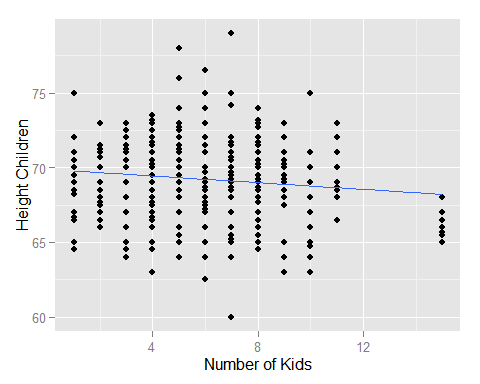
## Do kids get taller as a couple has more?

The p-value and t-statistic and common sense tell us that we don't get "better" at making taller kids. In fact, very large families tend to be smaller (maybe due to malnutrition in very large families?)

summary(m\_sons\_kids)

##   
## Call:  
## lm(formula = Height ~ Kids, data = sons)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.1138 -1.7779 0.1982 1.6624 9.8862   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 69.8973 0.3003 232.732 <2e-16 \*\*\*  
## Kids -0.1119 0.0460 -2.433 0.0153 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.618 on 463 degrees of freedom  
## Multiple R-squared: 0.01263, Adjusted R-squared: 0.0105   
## F-statistic: 5.922 on 1 and 463 DF, p-value: 0.01533

qplot(Kids,Height,data=sons) + geom\_smooth(method=lm,se=FALSE) + ylab("Height Children") + xlab("Number of Kids")



## Analysis of variance (ANOVA)

Analysis of variance (ANOVA) is a collection of statistical models used to analyze the differences among group means and their associated procedures (such as "variation" among and between groups), developed by statistician and evolutionary biologist Ronald Fisher.

* from [Analysis of variance](https://en.wikipedia.org/wiki/Analysis_of_variance)

R's anova() function returns a p-value and a F-statistic (which is like a t-statistic)

*F-statistic*

An [F-test](https://en.wikipedia.org/wiki/F-test) is any statistical test in which the test statistic has an F-distribution under the null hypothesis. Like a t-statistic, or a p-value it provides an estimate of whether one should accept or reject the null hypothesis. The F-test is sensitive to non-normality (as is a t-statistic) but is appropriate under the assumptions of normality and [homoscedasticity](https://en.wikipedia.org/wiki/Homoscedasticity).

anova(m\_sons\_fathers)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 83.719 < 2.2e-16 \*\*\*  
## Residuals 463 2721.28 5.88   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(m\_sons\_mothers)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Mother 1 358.75 358.75 58.187 1.367e-13 \*\*\*  
## Residuals 463 2854.59 6.17   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(m\_sons\_mfs)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## MF 1 763.35 763.35 144.26 < 2.2e-16 \*\*\*  
## Residuals 463 2449.98 5.29   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(m\_sons\_kids)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Kids 1 40.6 40.580 5.9219 0.01533 \*  
## Residuals 463 3172.8 6.853   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

A common use of ANOVA in simple regression to compare two or more models. Does the p-value or F-statistic improve or not?

anova(m\_sons\_fathers,m\_sons\_mfs)

## Analysis of Variance Table  
##   
## Model 1: Height ~ Father  
## Model 2: Height ~ MF  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 463 2721.3   
## 2 463 2450.0 0 271.3

anova(m\_sons\_mothers,m\_sons\_mfs)

## Analysis of Variance Table  
##   
## Model 1: Height ~ Mother  
## Model 2: Height ~ MF  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 463 2854.6   
## 2 463 2450.0 0 404.61

anova(m\_sons\_fathers,m\_sons\_mothers,m\_sons\_mfs)

## Analysis of Variance Table  
##   
## Model 1: Height ~ Father  
## Model 2: Height ~ Mother  
## Model 3: Height ~ MF  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 463 2721.3   
## 2 463 2854.6 0 -133.31   
## 3 463 2450.0 0 404.61

anova(m\_sons\_mfs,m\_sons\_kids)

## Analysis of Variance Table  
##   
## Model 1: Height ~ MF  
## Model 2: Height ~ Kids  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 463 2450.0   
## 2 463 3172.8 0 -722.77

anova(m\_sons\_fathers,m\_sons\_mothers,m\_sons\_mfs,m\_sons\_kids)

## Analysis of Variance Table  
##   
## Model 1: Height ~ Father  
## Model 2: Height ~ Mother  
## Model 3: Height ~ MF  
## Model 4: Height ~ Kids  
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 463 2721.3   
## 2 463 2854.6 0 -133.31   
## 3 463 2450.0 0 404.61   
## 4 463 3172.8 0 -722.77

Alternatively one can just check if the the residual sum of squares (RSS) improves (goes down) between two models.

## Residual Plots

The error term has the following assumptions:

* have mean zero; otherwise the forecasts will be systematically biased.
* statistical independence of the errors (in particular, no correlation between consecutive errors in the case of time series data).
* homoscedasticity (constant variance) of the errors.
* normality of the error distribution.

The typical plots are:

* *Residuls vs Fitted*

the residuals and the fitted values should be uncorrelated in a [homoscedastic](https://en.wikipedia.org/wiki/Homoscedasticity) linear model with normally distributed errors. There should not be a dependency between the residuals and the fitted values,

* *Residuls vs Normal*

This is a [Q–Q plot](https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot) to check if the residuls are normal (i.e. normality of the error distribution.)

* *Standardized Residuls vs Fitted Values*

[standardized residuals](https://en.wikipedia.org/wiki/Studentized_residual) means every residual plot you look at with any model is on the same standardized y-axis. This makes it easier to compare many residul plots. This process is also called *studentizing* (after [William Sealey Gosset](https://en.wikipedia.org/wiki/William_Sealy_Gosset), who wrote under the pseudonym Student).

The key reason for studentizing is that, in regression analysis of a multivariate distribution, the variances of the residuals at different input variable values may differ, even if the variances of the errors at these different input variable values are equal.

* *Residuls vs Leverage*

We use leverage to check for outliers. To understand [leverage](https://en.wikipedia.org/wiki/Leverage_%28statistics%29), recognize that simple linear regression fits a line that will pass through the center of your data. High-leverage points are those observations, if any, made at extreme or outlying values of the independent variables such that the lack of neighboring observations means that the fitted regression model will pass close to that particular observation.

To think of the leverage of a point consider how the slope might change if the model were fit without the data point in question. A common way to estimate of the influence of a data point is [Cook's distance or Cook's D](https://en.wikipedia.org/wiki/Cook%27s_distance)

where is the fitted value based on the fit with the point removed. ${S^2} $is the [Mean squared error](https://en.wikipedia.org/wiki/Mean_squared_error)

An alternate form of Cook's distance:

To be influential a point must:

Have high leverage and Have a high standardized residual

Analyists often look for and remove high leverage points and re-fit a model.

head(lm.influence(m\_sons\_mothers))

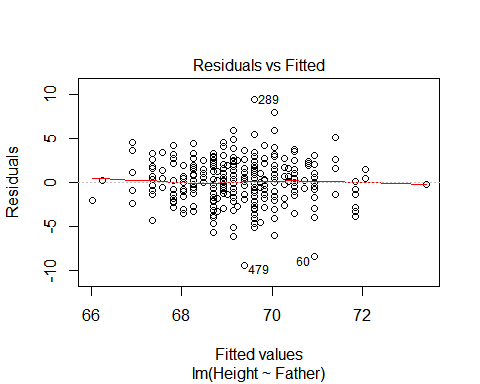
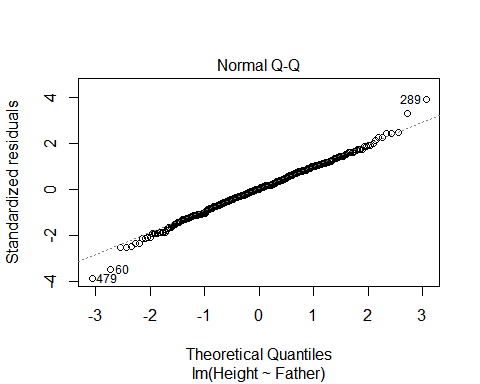
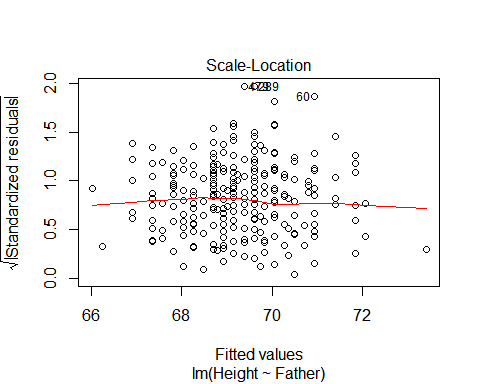
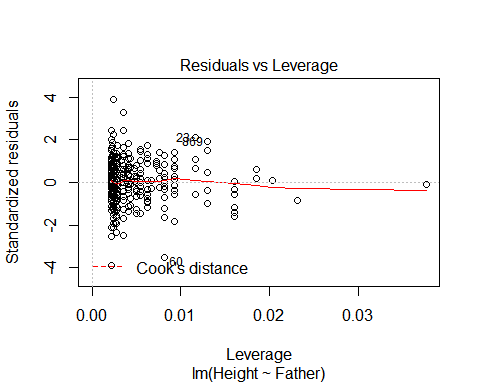
## $hat  
## 1 5 6 9 11 12   
## 0.005698049 0.004612389 0.004612389 0.002150548 0.002150548 0.002150548   
## 16 17 18 23 24 25   
## 0.014137718 0.014137718 0.014137718 0.008462661 0.008462661 0.008462661   
## 26 34 35 43 44 45   
## 0.008462661 0.003740825 0.003740825 0.005698049 0.005698049 0.004612389   
## 46 48 49 50 51 52   
## 0.004612389 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993   
## 57 58 59 60 67 68   
## 0.002247389 0.002247389 0.002247389 0.002247389 0.012018328 0.012018328   
## 69 75 78 79 81 82   
## 0.012018328 0.008462661 0.005698049 0.005698049 0.002541993 0.002541993   
## 83 91 92 93 96 97   
## 0.002541993 0.002550159 0.002550159 0.002550159 0.002550159 0.002550159   
## 99 100 109 110 111 115   
## 0.002550159 0.002550159 0.003740825 0.003740825 0.003740825 0.003740825   
## 116 117 120 121 122 123   
## 0.003740825 0.003740825 0.003740825 0.003740825 0.003740825 0.003740825   
## 126 127 128 129 131 132   
## 0.012018328 0.012018328 0.012018328 0.012018328 0.005698049 0.005698049   
## 133 135 139 140 141 145   
## 0.005698049 0.003724493 0.003724493 0.003724493 0.003724493 0.003724493   
## 147 148 149 150 152 153   
## 0.003724493 0.003724493 0.003724493 0.003724493 0.003034361 0.003034361   
## 154 155 156 159 161 162   
## 0.003034361 0.003034361 0.003034361 0.003034361 0.002541993 0.002541993   
## 163 164 174 175 176 178   
## 0.002541993 0.002541993 0.002247389 0.002247389 0.002247389 0.002150548   
## 179 180 181 182 183 190   
## 0.002150548 0.002150548 0.002247389 0.002247389 0.002247389 0.002251472   
## 191 195 196 197 198 204   
## 0.002251472 0.002550159 0.002550159 0.002550159 0.002550159 0.002550159   
## 205 206 208 209 213 214   
## 0.002550159 0.002550159 0.003740825 0.003740825 0.003740825 0.003740825   
## 215 218 223 224 225 226   
## 0.003740825 0.003046610 0.003740825 0.003740825 0.003740825 0.003740825   
## 227 231 232 233 234 235   
## 0.003740825 0.016414044 0.016414044 0.012018328 0.012018328 0.012018328   
## 237 238 243 244 245 250   
## 0.012018328 0.012018328 0.008462661 0.005698049 0.005698049 0.004612389   
## 251 252 253 261 262 263   
## 0.004612389 0.004612389 0.004612389 0.002541993 0.002541993 0.002541993   
## 264 270 271 272 273 274   
## 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993   
## 278 279 283 284 289 290   
## 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993   
## 291 296 297 299 300 301   
## 0.002541993 0.002541993 0.002541993 0.002541993 0.002541993 0.002341499   
## 302 303 308 309 310 311   
## 0.002341499 0.002341499 0.002150548 0.002150548 0.002150548 0.002150548   
## 312 313 315 316 317 319   
## 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548 0.002165553   
## 320 324 325 326 327 328   
## 0.002165553 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548   
## 329 333 337 338 339 340   
## 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548   
## 341 342 346 347 348 354   
## 0.002150548 0.002150548 0.002187371 0.002187371 0.002187371 0.002550159   
## 355 356 359 360 363 364   
## 0.002550159 0.002550159 0.002251472 0.002251472 0.002550159 0.002550159   
## 367 368 371 372 373 374   
## 0.002550159 0.002550159 0.003740825 0.003740825 0.003740825 0.003740825   
## 375 379 380 381 382 386   
## 0.003740825 0.002824298 0.002824298 0.002824298 0.002824298 0.003740825   
## 387 389 390 391 392 397   
## 0.003740825 0.005722547 0.005722547 0.008495324 0.008495324 0.014137718   
## 398 400 401 405 406 407   
## 0.014137718 0.016414044 0.016414044 0.010141612 0.010141612 0.010141612   
## 416 417 418 419 424 425   
## 0.003724493 0.003724493 0.003724493 0.003724493 0.003724493 0.003724493   
## 426 427 428 429 430 431   
## 0.003724493 0.005022922 0.005022922 0.005022922 0.005022922 0.005022922   
## 432 433 434 435 439 440   
## 0.005022922 0.003724493 0.003724493 0.003724493 0.004612389 0.004612389   
## 441 442 444 445 448 454   
## 0.004612389 0.004612389 0.004612389 0.004612389 0.004612389 0.003724493   
## 455 456 457 461 462 463   
## 0.003724493 0.003724493 0.003724493 0.003724493 0.003724493 0.003724493   
## 464 470 471 472 477 478   
## 0.003724493 0.002541993 0.002541993 0.002541993 0.002247389 0.002247389   
## 479 484 485 488 491 492   
## 0.002247389 0.002150548 0.002150548 0.002550159 0.002550159 0.002550159   
## 493 494 495 498 499 500   
## 0.002550159 0.002550159 0.002550159 0.002251472 0.002251472 0.002251472   
## 505 509 510 511 512 513   
## 0.002251472 0.003740825 0.003740825 0.003740825 0.003740825 0.003740825   
## 514 515 517 518 519 520   
## 0.003740825 0.003740825 0.003046610 0.003046610 0.003046610 0.003046610   
## 525 526 529 530 531 534   
## 0.003740825 0.003740825 0.005722547 0.005722547 0.005722547 0.005722547   
## 535 536 537 538 539 543   
## 0.005722547 0.005722547 0.005722547 0.005722547 0.005722547 0.008495324   
## 546 547 550 551 553 554   
## 0.008495324 0.008495324 0.007010053 0.018835057 0.005698049 0.005698049   
## 556 557 558 559 560 561   
## 0.004612389 0.004612389 0.004612389 0.004612389 0.004612389 0.004612389   
## 562 563 564 566 567 568   
## 0.004612389 0.004612389 0.004612389 0.002541993 0.002541993 0.003034361   
## 570 571 572 573 577 578   
## 0.003034361 0.003034361 0.003034361 0.003034361 0.002541993 0.002541993   
## 581 582 583 584 589 590   
## 0.002541993 0.002541993 0.002541993 0.002541993 0.002150548 0.002150548   
## 591 592 599 600 603 604   
## 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548   
## 605 606 609 610 611 619   
## 0.002150548 0.002150548 0.002150548 0.002150548 0.002150548 0.002550159   
## 620 621 622 623 627 628   
## 0.002550159 0.002550159 0.002550159 0.002550159 0.002251472 0.002251472   
## 629 631 632 633 636 637   
## 0.002251472 0.002550159 0.002550159 0.002550159 0.002550159 0.002550159   
## 638 639 640 644 645 646   
## 0.002550159 0.002550159 0.002550159 0.002550159 0.002550159 0.002550159   
## 650 651 652 653 654 655   
## 0.002251472 0.002550159 0.002251472 0.002251472 0.002251472 0.002251472   
## 657 658 660 661 662 663   
## 0.003046610 0.003740825 0.003046610 0.005722547 0.005722547 0.005722547   
## 666 667 674 675 676 678   
## 0.007877484 0.008495324 0.008495324 0.008495324 0.008495324 0.012059156   
## 679 680 681 682 683 684   
## 0.012059156 0.012059156 0.012059156 0.012059156 0.012059156 0.012059156   
## 689 690 691 694 695 696   
## 0.004055920 0.004055920 0.004055920 0.004612389 0.002541993 0.002541993   
## 701 702 706 707 710 713   
## 0.003034361 0.003034361 0.003034361 0.003034361 0.002541993 0.002541993   
## 714 724 725 726 727 728   
## 0.002541993 0.002150548 0.002150548 0.002150548 0.002150548 0.002251472   
## 729 730 731 732 733 734   
## 0.002251472 0.002251472 0.002251472 0.002251472 0.002251472 0.002251472   
## 736 737 739 740 741 742   
## 0.002550159 0.002550159 0.003740825 0.003740825 0.003740825 0.003740825   
## 744 745 746 747 748 749   
## 0.005722547 0.005698049 0.005698049 0.005698049 0.005698049 0.005698049   
## 753 754 755 762 763 767   
## 0.005698049 0.005698049 0.005698049 0.003724493 0.003724493 0.003724493   
## 768 773 774 775 776 781   
## 0.003724493 0.002541993 0.002541993 0.002541993 0.002541993 0.003034361   
## 782 783 786 789 790 791   
## 0.003034361 0.003034361 0.002550159 0.002550159 0.002550159 0.002550159   
## 795 796 802 803 804 805   
## 0.003046610 0.003046610 0.004632804 0.008495324 0.008495324 0.008495324   
## 807 808 809 810 811 812   
## 0.008495324 0.012059156 0.012059156 0.012059156 0.012059156 0.012059156   
## 813 814 823 824 825 828   
## 0.012059156 0.012059156 0.005698049 0.005698049 0.005698049 0.003724493   
## 832 833 834 841 843 844   
## 0.002541993 0.002541993 0.002541993 0.003034361 0.002541993 0.002541993   
## 845 846 849 850 855 857   
## 0.002541993 0.002541993 0.002150548 0.002150548 0.002550159 0.002550159   
## 858 860 861 862 864 865   
## 0.002550159 0.002550159 0.002550159 0.002550159 0.008495324 0.008495324   
## 866 867 869 870 876 877   
## 0.008495324 0.008495324 0.002150548 0.002150548 0.002150548 0.002150548   
## 883 884 886 889 891 892   
## 0.002550159 0.008495324 0.003724493 0.002550159 0.002541993 0.002541993   
## 893 894 895   
## 0.002541993 0.002541993 0.002541993   
##   
## $coefficients  
## (Intercept) Mother  
## 1 -0.2106377787 3.387023e-03  
## 5 -0.2041852378 3.302589e-03  
## 6 -0.1428941960 2.311239e-03  
## 9 0.0040535746 -3.627522e-06  
## 11 0.0029105164 -2.604605e-06  
## 12 -0.0016617162 1.487061e-06  
## 16 0.6954763641 -1.070082e-02  
## 17 0.2648334689 -4.074812e-03  
## 18 0.1212858371 -1.866143e-03  
## 23 -0.5756387342 9.189030e-03  
## 24 -0.3260715574 5.205142e-03  
## 25 -0.2262446867 3.611587e-03  
## 26 -0.2262446867 3.611587e-03  
## 34 0.2935210667 -4.399512e-03  
## 35 0.0810750041 -1.215212e-03  
## 43 0.1746219625 -2.807894e-03  
## 44 0.2487103743 -3.999224e-03  
## 45 -0.0509576333 8.242130e-04  
## 46 -0.0203121124 3.285378e-04  
## 48 -0.0553417087 9.453686e-04  
## 49 -0.0207017279 3.536350e-04  
## 50 -0.0137737317 2.352883e-04  
## 51 -0.0137737317 2.352883e-04  
## 52 0.0093195889 -1.592009e-04  
## 57 -0.0476821126 8.993660e-04  
## 58 -0.0372820308 7.032027e-04  
## 59 -0.0216819081 4.089578e-04  
## 60 0.0719188283 -1.356512e-03  
## 67 -0.2629026484 4.178589e-03  
## 68 -0.2377406660 3.778663e-03  
## 69 -0.1999976924 3.178774e-03  
## 75 -0.2262446867 3.611587e-03  
## 78 -0.1958200964 3.148757e-03  
## 79 -0.0476432728 7.660964e-04  
## 81 -0.1061470140 1.813245e-03  
## 82 -0.0207017279 3.536350e-04  
## 83 0.0023915927 -4.085412e-05  
## 91 0.0456464089 -6.575930e-04  
## 92 0.0456464089 -6.575930e-04  
## 93 0.0041450478 -5.971454e-05  
## 96 0.0041450478 -5.971454e-05  
## 97 -0.0511901003 7.374567e-04  
## 99 0.1148153440 -1.654057e-03  
## 100 -0.0511901003 7.374567e-04  
## 109 0.1341865198 -2.011287e-03  
## 110 0.0810750041 -1.215212e-03  
## 111 0.0810750041 -1.215212e-03  
## 115 0.2935210667 -4.399512e-03  
## 116 0.1872980354 -2.807362e-03  
## 117 0.0279634885 -4.191376e-04  
## 120 0.1872980354 -2.807362e-03  
## 121 0.1607422776 -2.409325e-03  
## 122 0.1607422776 -2.409325e-03  
## 123 0.0810750041 -1.215212e-03  
## 126 -0.8667902257 1.377681e-02  
## 127 -0.3635505780 5.778292e-03  
## 128 -0.2377406660 3.778663e-03  
## 129 -0.1119307541 1.779033e-03  
## 131 -0.2106377787 3.387023e-03  
## 132 -0.1958200964 3.148757e-03  
## 133 0.1005335508 -1.616564e-03  
## 135 -0.0009584331 1.564034e-05  
## 139 -0.0009584331 1.564034e-05  
## 140 0.0475727559 -7.763234e-04  
## 141 0.0718383504 -1.172305e-03  
## 145 -0.0494896221 8.076041e-04  
## 147 -0.2921455673 4.767423e-03  
## 148 -0.0980208112 1.599568e-03  
## 149 -0.0494896221 8.076041e-04  
## 150 0.1931663230 -3.152215e-03  
## 152 -0.0253505634 4.199346e-04  
## 153 -0.0790523032 1.309509e-03  
## 154 -0.0790523032 1.309509e-03  
## 155 -0.0432511434 7.164594e-04  
## 156 0.0283511764 -4.696400e-04  
## 159 -0.1148534631 1.902559e-03  
## 161 0.0254849133 -4.353432e-04  
## 162 0.0439595697 -7.509345e-04  
## 163 0.0370315736 -6.325878e-04  
## 164 0.0370315736 -6.325878e-04  
## 174 -0.0268819490 5.070394e-04  
## 175 -0.0164818671 3.108761e-04  
## 176 -0.0112818262 2.127945e-04  
## 178 -0.0028047744 2.509978e-06  
## 179 -0.0028047744 2.509978e-06  
## 180 -0.0028047744 2.509978e-06  
## 181 -0.0268819490 5.070394e-04  
## 182 -0.0164818671 3.108761e-04  
## 183 -0.0060817853 1.147129e-04  
## 190 0.0293687594 -3.927968e-04  
## 191 -0.0305225186 4.082279e-04  
## 195 0.0594801959 -8.568858e-04  
## 196 0.0318126219 -4.583002e-04  
## 197 0.0318126219 -4.583002e-04  
## 198 -0.1341928224 1.933214e-03  
## 204 0.0594801959 -8.568858e-04  
## 205 0.0594801959 -8.568858e-04  
## 206 0.0318126219 -4.583002e-04  
## 208 0.1341865198 -2.011287e-03  
## 209 0.0810750041 -1.215212e-03  
## 213 0.1872980354 -2.807362e-03  
## 214 0.1076307620 -1.613250e-03  
## 215 0.1076307620 -1.613250e-03  
## 218 0.0540321726 -7.990858e-04  
## 223 0.1076307620 -1.613250e-03  
## 224 0.0810750041 -1.215212e-03  
## 225 0.0279634885 -4.191376e-04  
## 226 0.0279634885 -4.191376e-04  
## 227 -0.1313710585 1.969087e-03  
## 231 0.7105841690 -1.094713e-02  
## 232 0.3187116956 -4.910012e-03  
## 233 0.0138791579 -2.205961e-04  
## 234 0.1396890698 -2.220226e-03  
## 235 0.2654989818 -4.219855e-03  
## 237 0.1396890698 -2.220226e-03  
## 238 0.3032419553 -4.819744e-03  
## 243 -0.4258984281 6.798697e-03  
## 244 0.0264451390 -4.252337e-04  
## 245 0.1005335508 -1.616564e-03  
## 250 -0.1735397169 2.806914e-03  
## 251 -0.1122486751 1.815563e-03  
## 252 -0.1122486751 1.815563e-03  
## 253 0.2248520548 -3.636864e-03  
## 261 -0.0553417087 9.453686e-04  
## 262 -0.0137737317 2.352883e-04  
## 263 0.0139382530 -2.380987e-04  
## 264 0.0254849133 -4.353432e-04  
## 270 -0.0784350293 1.339858e-03  
## 271 -0.0553417087 9.453686e-04  
## 272 -0.0207017279 3.536350e-04  
## 273 0.1063115353 -1.816055e-03  
## 274 0.1063115353 -1.816055e-03  
## 278 0.0601248941 -1.027077e-03  
## 279 0.1063115353 -1.816055e-03  
## 283 -0.0091550676 1.563904e-04  
## 284 -0.0091550676 1.563904e-04  
## 289 -0.2169949527 3.706792e-03  
## 290 -0.1246216704 2.128836e-03  
## 291 -0.0322483881 5.508795e-04  
## 296 -0.0784350293 1.339858e-03  
## 297 -0.0668883690 1.142613e-03  
## 299 0.0139382530 -2.380987e-04  
## 300 0.0139382530 -2.380987e-04  
## 301 -0.0388364401 6.912843e-04  
## 302 -0.0078842209 1.403382e-04  
## 303 0.0122347215 -2.177767e-04  
## 308 0.0033677397 -3.013772e-06  
## 309 0.0017674582 -1.581689e-06  
## 310 -0.0028047744 2.509978e-06  
## 311 -0.0050908907 4.555811e-06  
## 312 -0.0073770070 6.601644e-06  
## 313 -0.0096631234 8.647477e-06  
## 315 0.0017674582 -1.581689e-06  
## 316 -0.0028047744 2.509978e-06  
## 317 -0.0057767256 5.169561e-06  
## 319 -0.0075214393 2.083539e-04  
## 320 -0.0019454370 5.389121e-05  
## 324 0.0109119235 -9.765022e-06  
## 325 0.0006244001 -5.587723e-07  
## 326 -0.0005186581 4.641443e-07  
## 327 -0.0028047744 2.509978e-06  
## 328 -0.0028047744 2.509978e-06  
## 329 -0.0028047744 2.509978e-06  
## 333 -0.0028047744 2.509978e-06  
## 337 0.0040535746 -3.627522e-06  
## 338 0.0017674582 -1.581689e-06  
## 339 0.0017674582 -1.581689e-06  
## 340 0.0017674582 -1.581689e-06  
## 341 0.0006244001 -5.587723e-07  
## 342 -0.0016617162 1.487061e-06  
## 346 0.0087705926 -1.071906e-04  
## 347 -0.0209221920 2.557025e-04  
## 348 -0.0357685842 4.371490e-04  
## 354 0.1009815570 -1.454764e-03  
## 355 0.0041450478 -5.971454e-05  
## 356 -0.0511901003 7.374567e-04  
## 359 0.0293687594 -3.927968e-04  
## 360 -0.0230361089 3.080998e-04  
## 363 -0.0235225262 3.388711e-04  
## 364 -0.0511901003 7.374567e-04  
## 367 0.0318126219 -4.583002e-04  
## 368 -0.0650238873 9.367496e-04  
## 371 0.1872980354 -2.807362e-03  
## 372 0.0810750041 -1.215212e-03  
## 373 0.0545192463 -8.171750e-04  
## 374 0.0545192463 -8.171750e-04  
## 375 -0.0251480272 3.769373e-04  
## 379 0.0692708361 -1.016133e-03  
## 380 0.0339794001 -4.984434e-04  
## 381 0.0163336820 -2.395986e-04  
## 382 -0.1248320624 1.831160e-03  
## 386 0.1872980354 -2.807362e-03  
## 387 0.1872980354 -2.807362e-03  
## 389 0.2441558997 -3.709763e-03  
## 390 -0.0862954628 1.311194e-03  
## 391 -0.0752029673 1.150549e-03  
## 392 -0.3362820931 5.144863e-03  
## 397 0.6237025483 -9.596486e-03  
## 398 -0.3811308740 5.864201e-03  
## 400 0.7889586636 -1.215455e-02  
## 401 -0.1515352724 2.334524e-03  
## 405 -0.4599352275 7.324332e-03  
## 406 -0.0088000758 1.401386e-04  
## 407 0.1039837121 -1.655910e-03  
## 416 -0.1465520002 2.391532e-03  
## 417 -0.0980208112 1.599568e-03  
## 418 -0.0834614545 1.361979e-03  
## 419 -0.0737552167 1.203586e-03  
## 424 -0.0591958600 9.659968e-04  
## 425 -0.0494896221 8.076041e-04  
## 426 -0.0009584331 1.564034e-05  
## 427 -0.3158268513 5.095005e-03  
## 428 -0.2494217968 4.023740e-03  
## 429 -0.1166116878 1.881211e-03  
## 430 0.1158060030 -1.868214e-03  
## 431 0.2154135848 -3.475110e-03  
## 432 0.2818186393 -4.546375e-03  
## 433 -0.0009584331 1.564034e-05  
## 434 0.0718383504 -1.172305e-03  
## 435 0.0961039449 -1.568287e-03  
## 439 -0.1735397169 2.806914e-03  
## 440 -0.0509576333 8.242130e-04  
## 441 -0.0203121124 3.285378e-04  
## 442 -0.0203121124 3.285378e-04  
## 444 -0.0203121124 3.285378e-04  
## 445 0.1635610130 -2.645514e-03  
## 448 -0.0509576333 8.242130e-04  
## 454 -0.0494896221 8.076041e-04  
## 455 -0.0494896221 8.076041e-04  
## 456 -0.0252240276 4.116222e-04  
## 457 -0.0252240276 4.116222e-04  
## 461 -0.1465520002 2.391532e-03  
## 462 -0.0980208112 1.599568e-03  
## 463 0.0475727559 -7.763234e-04  
## 464 0.0475727559 -7.763234e-04  
## 470 -0.0091550676 1.563904e-04  
## 471 0.0254849133 -4.353432e-04  
## 472 0.0601248941 -1.027077e-03  
## 477 -0.0029617608 5.586386e-05  
## 478 0.0147183783 -2.776137e-04  
## 479 0.0979190329 -1.846920e-03  
## 484 0.0056538560 -5.059605e-06  
## 485 -0.0062339489 5.578727e-06  
## 488 0.0041450478 -5.971454e-05  
## 491 0.0871477700 -1.255471e-03  
## 492 0.1148153440 -1.654057e-03  
## 493 0.0318126219 -4.583002e-04  
## 494 0.0318126219 -4.583002e-04  
## 495 -0.1341928224 1.933214e-03  
## 498 0.0218823497 -2.926687e-04  
## 499 -0.0305225186 4.082279e-04  
## 500 -0.0454953381 6.084841e-04  
## 505 0.0218823497 -2.926687e-04  
## 509 0.2404095511 -3.603437e-03  
## 510 0.1872980354 -2.807362e-03  
## 511 0.0279634885 -4.191376e-04  
## 512 0.2404095511 -3.603437e-03  
## 513 0.1341865198 -2.011287e-03  
## 514 0.1341865198 -2.011287e-03  
## 515 0.0279634885 -4.191376e-04  
## 517 0.0944101145 -1.396238e-03  
## 518 0.0540321726 -7.990858e-04  
## 519 0.0540321726 -7.990858e-04  
## 520 0.0136542307 -2.019334e-04  
## 525 0.1872980354 -2.807362e-03  
## 526 -0.0251480272 3.769373e-04  
## 529 0.1497412247 -2.275204e-03  
## 530 0.1104017768 -1.677471e-03  
## 531 0.0710623288 -1.079738e-03  
## 534 -0.0076165670 1.157279e-04  
## 535 -0.0076165670 1.157279e-04  
## 536 -0.0469560149 7.134608e-04  
## 537 -0.3223321503 4.897591e-03  
## 538 -0.4010110462 6.093057e-03  
## 539 -0.4010110462 6.093057e-03  
## 543 0.2903078087 -4.441491e-03  
## 546 0.1336603332 -2.044902e-03  
## 547 -0.1796346176 2.748274e-03  
## 550 0.1456706498 -2.222068e-03  
## 551 0.1116942454 -1.768209e-03  
## 553 -0.1958200964 3.148757e-03  
## 554 -0.0476432728 7.660964e-04  
## 556 0.0103334085 -1.671375e-04  
## 557 0.0716244503 -1.158488e-03  
## 558 0.0716244503 -1.158488e-03  
## 559 0.0900117629 -1.455893e-03  
## 560 0.1022699712 -1.654163e-03  
## 561 0.1022699712 -1.654163e-03  
## 562 0.1329154921 -2.149838e-03  
## 563 0.1329154921 -2.149838e-03  
## 564 0.1329154921 -2.149838e-03  
## 566 0.0485782338 -8.298323e-04  
## 567 0.0832182147 -1.421566e-03  
## 568 0.1357546560 -2.248789e-03  
## 570 -0.0683119553 1.131594e-03  
## 571 -0.0611517233 1.012984e-03  
## 572 -0.0325107954 5.385445e-04  
## 573 0.1536552360 -2.545314e-03  
## 577 -0.0553417087 9.453686e-04  
## 578 -0.0553417087 9.453686e-04  
## 581 0.0093195889 -1.592009e-04  
## 582 0.0370315736 -6.325878e-04  
## 583 0.0832182147 -1.421566e-03  
## 584 0.0832182147 -1.421566e-03  
## 589 0.0040535746 -3.627522e-06  
## 590 -0.0028047744 2.509978e-06  
## 591 -0.0028047744 2.509978e-06  
## 592 -0.0050908907 4.555811e-06  
## 599 -0.0073770070 6.601644e-06  
## 600 -0.0142353560 1.273914e-05  
## 603 0.0045107978 -4.036689e-06  
## 604 0.0045107978 -4.036689e-06  
## 605 -0.0005186581 4.641443e-07  
## 606 -0.0016617162 1.487061e-06  
## 609 -0.0005186581 4.641443e-07  
## 610 -0.0050908907 4.555811e-06  
## 611 -0.0073770070 6.601644e-06  
## 619 0.0456464089 -6.575930e-04  
## 620 0.0318126219 -4.583002e-04  
## 621 -0.0235225262 3.388711e-04  
## 622 -0.0788576743 1.136042e-03  
## 623 -0.0788576743 1.136042e-03  
## 627 0.0668008082 -8.934372e-04  
## 628 0.0143959399 -1.925406e-04  
## 629 0.0069095301 -9.241253e-05  
## 631 -0.0511901003 7.374567e-04  
## 632 0.0318126219 -4.583002e-04  
## 633 -0.0235225262 3.388711e-04  
## 636 0.0594801959 -8.568858e-04  
## 637 -0.0235225262 3.388711e-04  
## 638 -0.0788576743 1.136042e-03  
## 639 -0.0926914613 1.335335e-03  
## 640 -0.1065252484 1.534628e-03  
## 644 -0.0511901003 7.374567e-04  
## 645 -0.0511901003 7.374567e-04  
## 646 -0.0788576743 1.136042e-03  
## 650 -0.0125551352 1.679205e-04  
## 651 0.0318126219 -4.583002e-04  
## 652 0.0143959399 -1.925406e-04  
## 653 -0.0005768796 7.715561e-06  
## 654 -0.0305225186 4.082279e-04  
## 655 -0.0529817479 7.086122e-04  
## 657 -0.0065347403 9.664275e-05  
## 658 -0.0410814819 6.157598e-04  
## 660 -0.0872906241 1.290947e-03  
## 661 0.0317228809 -4.820050e-04  
## 662 -0.0076165670 1.157279e-04  
## 663 -0.3223321503 4.897591e-03  
## 666 -0.1087383690 1.661801e-03  
## 667 -0.3884979182 5.943726e-03  
## 674 -0.0229871422 3.516860e-04  
## 675 -0.0752029673 1.150549e-03  
## 676 -0.1274187925 1.949412e-03  
## 678 0.2160720509 -3.319514e-03  
## 679 0.0856388495 -1.315669e-03  
## 680 -0.3056607547 4.695865e-03  
## 681 -0.3447907152 5.297019e-03  
## 682 -0.4360939561 6.699710e-03  
## 683 -0.4360939561 6.699710e-03  
## 684 -0.5665271575 8.703555e-03  
## 689 -0.1418227283 2.305014e-03  
## 690 -0.1418227283 2.305014e-03  
## 691 -0.0774653819 1.259028e-03  
## 694 -0.0509576333 8.242130e-04  
## 695 -0.0022270714 3.804370e-05  
## 696 0.0485782338 -8.298323e-04  
## 701 -0.0074499835 1.234097e-04  
## 702 0.0283511764 -4.696400e-04  
## 706 -0.0074499835 1.234097e-04  
## 707 0.0748926842 -1.240605e-03  
## 710 0.1063115353 -1.816055e-03  
## 713 -0.0322483881 5.508795e-04  
## 714 0.0139382530 -2.380987e-04  
## 724 0.0051966327 -4.650439e-06  
## 725 0.0017674582 -1.581689e-06  
## 726 -0.0050908907 4.555811e-06  
## 727 -0.0050908907 4.555811e-06  
## 728 0.0293687594 -3.927968e-04  
## 729 0.0173905038 -2.325919e-04  
## 730 0.0024176843 -3.233567e-05  
## 731 -0.0080632894 1.078437e-04  
## 732 -0.0155496991 2.079717e-04  
## 733 -0.0305225186 4.082279e-04  
## 734 -0.0529817479 7.086122e-04  
## 736 0.0041450478 -5.971454e-05  
## 737 -0.0235225262 3.388711e-04  
## 739 0.0810750041 -1.215212e-03  
## 740 0.0545192463 -8.171750e-04  
## 741 0.0279634885 -4.191376e-04  
## 742 0.0014077307 -2.110011e-05  
## 744 -0.0862954628 1.311194e-03  
## 745 -0.0105990669 1.704313e-04  
## 746 -0.0105990669 1.704313e-04  
## 747 0.2487103743 -3.999224e-03  
## 748 0.3227987861 -5.190554e-03  
## 749 0.3227987861 -5.190554e-03  
## 753 -0.1217316846 1.957426e-03  
## 754 0.3968871979 -6.381884e-03  
## 755 0.3968871979 -6.381884e-03  
## 762 0.1931663230 -3.152215e-03  
## 763 0.2416975120 -3.944178e-03  
## 767 -0.0980208112 1.599568e-03  
## 768 0.0961039449 -1.568287e-03  
## 773 0.0208662492 -3.564454e-04  
## 774 0.0254849133 -4.353432e-04  
## 775 0.0716715544 -1.224321e-03  
## 776 0.1178581956 -2.013300e-03  
## 781 -0.0790523032 1.309509e-03  
## 782 -0.0432511434 7.164594e-04  
## 783 0.0999534961 -1.655739e-03  
## 786 0.0318126219 -4.583002e-04  
## 789 -0.0456565855 6.577396e-04  
## 790 -0.0511901003 7.374567e-04  
## 791 -0.1065252484 1.534628e-03  
## 795 0.0540321726 -7.990858e-04  
## 796 -0.0267237112 3.952189e-04  
## 802 0.1129673275 -1.707086e-03  
## 803 0.0292286829 -4.471768e-04  
## 804 -0.0752029673 1.150549e-03  
## 805 -0.2840662679 4.346000e-03  
## 807 -0.2840662679 4.346000e-03  
## 808 0.0856388495 -1.315669e-03  
## 809 -0.0447943519 6.881755e-04  
## 810 -0.1100109526 1.690098e-03  
## 811 -0.1752275533 2.692020e-03  
## 812 -0.2143575138 3.293174e-03  
## 813 -0.2404441540 3.693943e-03  
## 814 -0.3056607547 4.695865e-03  
## 823 0.2857545802 -4.594889e-03  
## 824 0.3227987861 -5.190554e-03  
## 825 0.3227987861 -5.190554e-03  
## 828 0.3387598901 -5.528106e-03  
## 832 0.0139382530 -2.380987e-04  
## 833 0.0370315736 -6.325878e-04  
## 834 0.0370315736 -6.325878e-04  
## 841 -0.0325107954 5.385445e-04  
## 843 0.0093195889 -1.592009e-04  
## 844 0.0139382530 -2.380987e-04  
## 845 0.0370315736 -6.325878e-04  
## 846 0.0439595697 -7.509345e-04  
## 849 -0.0050908907 4.555811e-06  
## 850 -0.0050908907 4.555811e-06  
## 855 0.0318126219 -4.583002e-04  
## 857 -0.0788576743 1.136042e-03  
## 858 -0.0788576743 1.136042e-03  
## 860 0.0594801959 -8.568858e-04  
## 861 0.0594801959 -8.568858e-04  
## 862 0.0041450478 -5.971454e-05  
## 864 0.0292286829 -4.471768e-04  
## 865 0.0292286829 -4.471768e-04  
## 866 -0.0752029673 1.150549e-03  
## 867 -0.0752029673 1.150549e-03  
## 869 0.0051966327 -4.650439e-06  
## 870 -0.0028047744 2.509978e-06  
## 876 0.0029105164 -2.604605e-06  
## 877 -0.0028047744 2.509978e-06  
## 883 -0.1203590354 1.733921e-03  
## 884 -0.1796346176 2.748274e-03  
## 886 0.2902287010 -4.736142e-03  
## 889 -0.0650238873 9.367496e-04  
## 891 -0.0553417087 9.453686e-04  
## 892 -0.0207017279 3.536350e-04  
## 893 0.0208662492 -3.564454e-04  
## 894 0.0254849133 -4.353432e-04  
## 895 0.0439595697 -7.509345e-04  
##   
## $sigma  
## 1 5 6 9 11 12 16 17   
## 2.482169 2.480852 2.483332 2.484339 2.485003 2.485480 2.475322 2.484207   
## 18 23 24 25 26 34 35 43   
## 2.485395 2.471067 2.481021 2.483454 2.483454 2.472327 2.484692 2.483277   
## 44 45 46 48 49 50 51 52   
## 2.480771 2.485408 2.485663 2.483203 2.485360 2.485555 2.485555 2.485640   
## 57 58 59 60 67 68 69 75   
## 2.476521 2.480097 2.483813 2.464755 2.483786 2.484137 2.484597 2.483454   
## 78 79 81 82 83 91 92 93   
## 2.482650 2.485530 2.476472 2.485360 2.485706 2.484522 2.484522 2.485701   
## 96 97 99 100 109 110 111 115   
## 2.485701 2.484216 2.478182 2.484216 2.482920 2.484692 2.484692 2.472327   
## 116 117 120 121 122 123 126 127   
## 2.480270 2.485590 2.480270 2.481704 2.481704 2.484692 2.464704 2.482028   
## 128 129 131 132 133 135 139 140   
## 2.484137 2.485362 2.482169 2.482650 2.484904 2.485711 2.485711 2.485291   
## 141 145 147 148 149 150 152 153   
## 2.484753 2.485256 2.469824 2.483927 2.485256 2.478778 2.485492 2.483581   
## 154 155 156 159 161 162 163 164   
## 2.483581 2.485073 2.485437 2.481212 2.485179 2.484129 2.484588 2.484588   
## 174 175 176 178 179 180 181 182   
## 2.482794 2.484615 2.485197 2.485054 2.485054 2.485054 2.482794 2.484615   
## 183 190 191 195 196 197 198 204   
## 2.485562 2.484031 2.483897 2.483693 2.485134 2.485134 2.475421 2.483693   
## 205 206 208 209 213 214 215 218   
## 2.483693 2.485134 2.482920 2.484692 2.480270 2.483915 2.483915 2.484929   
## 223 224 225 226 227 231 232 233   
## 2.483915 2.484692 2.485590 2.485590 2.483036 2.476597 2.483880 2.485705   
## 234 235 237 238 243 244 245 250   
## 2.485167 2.483747 2.485167 2.483149 2.477705 2.485655 2.484904 2.482202   
## 251 252 253 261 262 263 264 270   
## 2.484243 2.484243 2.479817 2.483203 2.485555 2.485552 2.485179 2.480670   
## 271 272 273 274 278 279 283 284   
## 2.483203 2.485360 2.476443 2.476443 2.482750 2.476443 2.485642 2.485642   
## 289 290 291 296 297 299 300 301   
## 2.446867 2.472967 2.484859 2.480670 2.482046 2.485552 2.485552 2.482961   
## 302 303 308 309 310 311 312 313   
## 2.485597 2.485438 2.484764 2.485450 2.485054 2.483546 2.481163 2.477903   
## 315 316 317 319 320 324 325 326   
## 2.485450 2.485054 2.482923 2.482533 2.485498 2.475750 2.485678 2.485688   
## 327 328 329 333 337 338 339 340   
## 2.485054 2.485054 2.485054 2.485054 2.484339 2.485450 2.485450 2.485450   
## 341 342 346 347 348 354 355 356   
## 2.485678 2.485480 2.485368 2.483760 2.480006 2.479889 2.485701 2.484216   
## 359 360 363 364 367 368 371 372   
## 2.484031 2.484678 2.485395 2.484216 2.485134 2.483299 2.480270 2.484692   
## 373 374 375 379 380 381 382 386   
## 2.485250 2.485250 2.485613 2.484028 2.485306 2.485617 2.480242 2.480270   
## 387 389 390 391 392 397 398 400   
## 2.480270 2.481490 2.485184 2.485483 2.481153 2.477359 2.482596 2.474471   
## 401 405 406 407 416 417 418 419   
## 2.485297 2.478385 2.485708 2.485337 2.481722 2.483927 2.484418 2.484701   
## 424 425 426 427 428 429 430 431   
## 2.485060 2.485256 2.485711 2.475793 2.479530 2.484361 2.484380 2.481102   
## 432 433 434 435 439 440 441 442   
## 2.477817 2.485711 2.484753 2.483996 2.482202 2.485408 2.485663 2.485663   
## 444 445 448 454 455 456 457 461   
## 2.485663 2.482594 2.485408 2.485256 2.485256 2.485593 2.485593 2.481722   
## 462 463 464 470 471 472 477 478   
## 2.483927 2.485291 2.485291 2.485642 2.485179 2.482750 2.485675 2.484837   
## 479 484 485 488 491 492 493 494   
## 2.446722 2.483041 2.482464 2.485701 2.481376 2.478182 2.485134 2.485134   
## 495 498 499 500 505 509 510 511   
## 2.475421 2.484779 2.483897 2.481679 2.484779 2.476740 2.480270 2.485590   
## 512 513 514 515 517 518 519 520   
## 2.476740 2.482920 2.482920 2.485590 2.483322 2.484929 2.484929 2.485661   
## 525 526 529 530 531 534 535 536   
## 2.480270 2.485613 2.484124 2.484848 2.485353 2.485707 2.485707 2.485555   
## 537 538 539 543 546 547 550 551   
## 2.478350 2.474309 2.474309 2.482315 2.484991 2.484411 2.484600 2.485508   
## 553 554 556 557 558 559 560 561   
## 2.482650 2.485530 2.485698 2.485113 2.485113 2.484767 2.484493 2.484493   
## 562 563 564 566 567 568 570 571   
## 2.483653 2.483653 2.483653 2.483778 2.480036 2.479423 2.484120 2.484436   
## 572 573 577 578 581 582 583 584   
## 2.485351 2.477653 2.483203 2.483203 2.485640 2.484588 2.480036 2.480036   
## 589 590 591 592 599 600 603 604   
## 2.484339 2.485054 2.485054 2.483546 2.481163 2.468735 2.484011 2.484011   
## 605 606 609 610 611 619 620 621   
## 2.485688 2.485480 2.485688 2.483546 2.481163 2.484522 2.485134 2.485395   
## 622 623 627 628 629 631 632 633   
## 2.482162 2.482162 2.477010 2.485307 2.485618 2.484216 2.485134 2.485395   
## 636 637 638 639 640 644 645 646   
## 2.483693 2.485395 2.482162 2.480807 2.479232 2.484216 2.484216 2.482162   
## 650 651 652 653 654 655 657 658   
## 2.485404 2.485134 2.485307 2.485710 2.483897 2.480241 2.485699 2.485449   
## 660 661 662 663 666 667 674 675   
## 2.483669 2.485640 2.485707 2.478350 2.485184 2.479626 2.485689 2.485483   
## 676 678 679 680 681 682 683 684   
## 2.485057 2.484501 2.485521 2.483289 2.482629 2.480779 2.480779 2.477383   
## 689 690 691 694 695 696 701 702   
## 2.482652 2.482652 2.484799 2.485408 2.485707 2.483778 2.485692 2.485437   
## 706 707 710 713 714 724 725 726   
## 2.485692 2.483799 2.476443 2.484859 2.485552 2.483455 2.485450 2.483546   
## 727 728 729 730 731 732 733 734   
## 2.483546 2.484031 2.485122 2.485699 2.485584 2.485240 2.483897 2.480241   
## 736 737 739 740 741 742 744 745   
## 2.485701 2.485395 2.484692 2.485250 2.485590 2.485710 2.485184 2.485702   
## 746 747 748 749 753 754 755 762   
## 2.485702 2.480771 2.477384 2.477384 2.484528 2.473113 2.473113 2.478778   
## 763 767 768 773 774 775 776 781   
## 2.474848 2.483927 2.483996 2.485354 2.485179 2.481503 2.474315 2.483581   
## 782 783 786 789 790 791 795 796   
## 2.485073 2.482304 2.485134 2.484522 2.484216 2.479232 2.484929 2.485519   
## 802 803 804 805 807 808 809 810   
## 2.484424 2.485676 2.485483 2.482460 2.482460 2.485521 2.485659 2.485397   
## 811 812 813 814 823 824 825 828   
## 2.484915 2.484520 2.484213 2.483289 2.479188 2.477384 2.477384 2.464326   
## 832 833 834 841 843 844 845 846   
## 2.485552 2.484588 2.484588 2.485351 2.485640 2.485552 2.484588 2.484129   
## 849 850 855 857 858 860 861 862   
## 2.483546 2.483546 2.485134 2.482162 2.482162 2.483693 2.483693 2.485701   
## 864 865 866 867 869 870 876 877   
## 2.485676 2.485676 2.485483 2.485483 2.483455 2.485054 2.485003 2.485054   
## 883 884 886 889 891 892 893 894   
## 2.477437 2.484411 2.470032 2.483299 2.483203 2.485360 2.485354 2.485179   
## 895   
## 2.484129   
##   
## $wt.res  
## 1 5 6 9 11 12   
## 2.84305971 3.33140426 2.33140426 1.77312700 1.27312700 -0.72687300   
## 16 17 18 23 24 25   
## 4.84491702 1.84491702 0.84491702 5.76637062 3.26637062 2.26637062   
## 26 34 35 43 44 45   
## 2.26637062 5.52650519 1.52650519 -2.35694029 -3.35694029 0.83140426   
## 46 48 49 50 51 52   
## 0.33140426 2.39643790 0.89643790 0.59643790 0.59643790 -0.40356210   
## 57 58 59 60 67 68   
## 4.58478245 3.58478245 2.08478245 -6.91521755 2.08968152 1.88968152   
## 69 75 78 79 81 82   
## 1.58968152 2.26637062 2.64305971 0.64305971 4.59643790 0.89643790   
## 83 91 92 93 96 97   
## -0.10356210 1.64981609 1.64981609 0.14981609 0.14981609 -1.85018391   
## 99 100 109 110 111 115   
## 4.14981609 -1.85018391 2.52650519 1.52650519 1.52650519 5.52650519   
## 116 117 120 121 122 123   
## 3.52650519 0.52650519 3.52650519 3.02650519 3.02650519 1.52650519   
## 126 127 128 129 131 132   
## 6.88968152 2.88968152 1.88968152 0.88968152 2.84305971 2.64305971   
## 133 135 139 140 141 145   
## -1.35694029 0.01974881 0.01974881 -0.98025119 -1.48025119 1.01974881   
## 147 148 149 150 152 153   
## 6.01974881 2.01974881 1.01974881 -3.98025119 0.70809336 2.20809336   
## 154 155 156 159 161 162   
## 2.20809336 1.20809336 -0.79190664 3.20809336 -1.10356210 -1.90356210   
## 163 164 174 175 176 178   
## -1.60356210 -1.60356210 2.58478245 1.58478245 1.08478245 -1.22687300   
## 179 180 181 182 183 190   
## -1.22687300 -1.22687300 2.58478245 1.58478245 0.58478245 1.96147154   
## 191 195 196 197 198 204   
## -2.03852846 2.14981609 1.14981609 1.14981609 -4.85018391 2.14981609   
## 205 206 208 209 213 214   
## 2.14981609 1.14981609 2.52650519 1.52650519 3.52650519 2.02650519   
## 215 218 223 224 225 226   
## 2.02650519 1.33816064 2.02650519 1.52650519 0.52650519 0.52650519   
## 227 231 232 233 234 235   
## -2.47349481 4.53326157 2.03326157 -0.11031848 -1.11031848 -2.11031848   
## 237 238 243 244 245 250   
## -1.11031848 -2.41031848 4.26637062 -0.35694029 -1.35694029 2.83140426   
## 251 252 253 261 262 263   
## 1.83140426 1.83140426 -3.66859574 2.39643790 0.59643790 -0.60356210   
## 264 270 271 272 273 274   
## -1.10356210 3.39643790 2.39643790 0.89643790 -4.60356210 -4.60356210   
## 278 279 283 284 289 290   
## -2.60356210 -4.60356210 0.39643790 0.39643790 9.39643790 5.39643790   
## 291 296 297 299 300 301   
## 1.39643790 3.39643790 2.89643790 -0.60356210 -0.60356210 2.50944463   
## 302 303 308 309 310 311   
## 0.50944463 -0.79055537 1.47312700 0.77312700 -1.22687300 -2.22687300   
## 312 313 315 316 317 319   
## -3.22687300 -4.22687300 0.77312700 -1.22687300 -2.52687300 2.69778918   
## 320 324 325 326 327 328   
## 0.69778918 4.77312700 0.27312700 -0.22687300 -1.22687300 -1.22687300   
## 329 333 337 338 339 340   
## -1.22687300 -1.22687300 1.77312700 0.77312700 0.77312700 0.77312700   
## 341 342 346 347 348 354   
## 0.27312700 -0.72687300 0.88613373 -2.11386627 -3.61386627 3.64981609   
## 355 356 359 360 363 364   
## 0.14981609 -1.85018391 1.96147154 -1.53852846 -0.85018391 -1.85018391   
## 367 368 371 372 373 374   
## 1.14981609 -2.35018391 3.52650519 1.52650519 1.02650519 1.02650519   
## 375 379 380 381 382 386   
## -0.47349481 1.96282282 0.96282282 0.46282282 -3.53717718 3.52650519   
## 387 389 390 391 392 397   
## 3.52650519 3.10319428 -1.09680572 -0.72011662 -3.22011662 4.34491702   
## 398 400 401 405 406 407   
## -2.65508298 5.03326157 -0.96673843 4.07802607 0.07802607 -0.92197393   
## 416 417 418 419 424 425   
## 3.01974881 2.01974881 1.71974881 1.51974881 1.21974881 1.01974881   
## 426 427 428 429 430 431   
## 0.01974881 4.75606644 3.75606644 1.75606644 -1.74393356 -3.24393356   
## 432 433 434 435 439 440   
## -4.24393356 0.01974881 -1.48025119 -1.98025119 2.83140426 0.83140426   
## 441 442 444 445 448 454   
## 0.33140426 0.33140426 0.33140426 -2.66859574 0.83140426 1.01974881   
## 455 456 457 461 462 463   
## 1.01974881 0.51974881 0.51974881 3.01974881 2.01974881 -0.98025119   
## 464 470 471 472 477 478   
## -0.98025119 0.39643790 -1.10356210 -2.60356210 0.28478245 -1.41521755   
## 479 484 485 488 491 492   
## -9.41521755 2.47312700 -2.72687300 0.14981609 3.14981609 4.14981609   
## 493 494 495 498 499 500   
## 1.14981609 1.14981609 -4.85018391 1.46147154 -2.03852846 -3.03852846   
## 505 509 510 511 512 513   
## 1.46147154 4.52650519 3.52650519 0.52650519 4.52650519 2.52650519   
## 514 515 517 518 519 520   
## 2.52650519 0.52650519 2.33816064 1.33816064 1.33816064 0.33816064   
## 525 526 529 530 531 534   
## 3.52650519 -0.47349481 1.90319428 1.40319428 0.90319428 -0.09680572   
## 535 536 537 538 539 543   
## -0.09680572 -0.59680572 -4.09680572 -5.09680572 -5.09680572 2.77988338   
## 546 547 550 551 553 554   
## 1.27988338 -1.72011662 1.59153883 -0.67535212 2.64305971 0.64305971   
## 556 557 558 559 560 561   
## -0.16859574 -1.16859574 -1.16859574 -1.46859574 -1.66859574 -1.66859574   
## 562 563 564 566 567 568   
## -2.16859574 -2.16859574 -2.16859574 -2.10356210 -3.60356210 -3.79190664   
## 570 571 572 573 577 578   
## 1.90809336 1.70809336 0.90809336 -4.29190664 2.39643790 2.39643790   
## 581 582 583 584 589 590   
## -0.40356210 -1.60356210 -3.60356210 -3.60356210 1.77312700 -1.22687300   
## 591 592 599 600 603 604   
## -1.22687300 -2.22687300 -3.22687300 -6.22687300 1.97312700 1.97312700   
## 605 606 609 610 611 619   
## -0.22687300 -0.72687300 -0.22687300 -2.22687300 -3.22687300 1.64981609   
## 620 621 622 623 627 628   
## 1.14981609 -0.85018391 -2.85018391 -2.85018391 4.46147154 0.96147154   
## 629 631 632 633 636 637   
## 0.46147154 -1.85018391 1.14981609 -0.85018391 2.14981609 -0.85018391   
## 638 639 640 644 645 646   
## -2.85018391 -3.35018391 -3.85018391 -1.85018391 -1.85018391 -2.85018391   
## 650 651 652 653 654 655   
## -0.83852846 1.14981609 0.96147154 -0.03852846 -2.03852846 -3.53852846   
## 657 658 660 661 662 663   
## -0.16183936 -0.77349481 -2.16183936 0.40319428 -0.09680572 -4.09680572   
## 666 667 674 675 676 678   
## -1.09545444 -3.72011662 -0.22011662 -0.72011662 -1.22011662 1.65657247   
## 679 680 681 682 683 684   
## 0.65657247 -2.34342753 -2.64342753 -3.34342753 -3.34342753 -4.34342753   
## 689 690 691 694 695 696   
## 2.64441099 2.64441099 1.44441099 0.83140426 0.09643790 -2.10356210   
## 701 702 706 707 710 713   
## 0.20809336 -0.79190664 0.20809336 -2.09190664 -4.60356210 1.39643790   
## 714 724 725 726 727 728   
## -0.60356210 2.27312700 0.77312700 -2.22687300 -2.22687300 1.96147154   
## 729 730 731 732 733 734   
## 1.16147154 0.16147154 -0.53852846 -1.03852846 -2.03852846 -3.53852846   
## 736 737 739 740 741 742   
## 0.14981609 -0.85018391 1.52650519 1.02650519 0.52650519 0.02650519   
## 744 745 746 747 748 749   
## -1.09680572 0.14305971 0.14305971 -3.35694029 -4.35694029 -4.35694029   
## 753 754 755 762 763 767   
## 1.64305971 -5.35694029 -5.35694029 -3.98025119 -4.98025119 2.01974881   
## 768 773 774 775 776 781   
## -1.98025119 -0.90356210 -1.10356210 -3.10356210 -5.10356210 2.20809336   
## 782 783 786 789 790 791   
## 1.20809336 -2.79190664 1.14981609 -1.65018391 -1.85018391 -3.85018391   
## 795 796 802 803 804 805   
## 1.33816064 -0.66183936 1.71484973 0.27988338 -0.72011662 -2.72011662   
## 807 808 809 810 811 812   
## -2.72011662 0.65657247 -0.34342753 -0.84342753 -1.34342753 -1.64342753   
## 813 814 823 824 825 828   
## -1.84342753 -2.34342753 -3.85694029 -4.35694029 -4.35694029 -6.98025119   
## 832 833 834 841 843 844   
## -0.60356210 -1.60356210 -1.60356210 0.90809336 -0.40356210 -0.60356210   
## 845 846 849 850 855 857   
## -1.60356210 -1.90356210 -2.22687300 -2.22687300 1.14981609 -2.85018391   
## 858 860 861 862 864 865   
## -2.85018391 2.14981609 2.14981609 0.14981609 0.27988338 0.27988338   
## 866 867 869 870 876 877   
## -0.72011662 -0.72011662 2.27312700 -1.22687300 1.27312700 -1.22687300   
## 883 884 886 889 891 892   
## -4.35018391 -1.72011662 -5.98025119 -2.35018391 2.39643790 0.89643790   
## 893 894 895   
## -0.90356210 -1.10356210 -1.90356210

head(cooks.distance(m\_sons\_mothers))

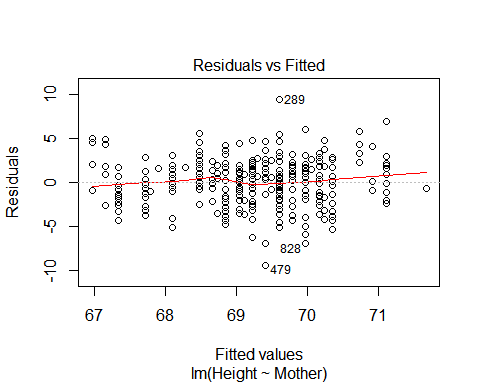
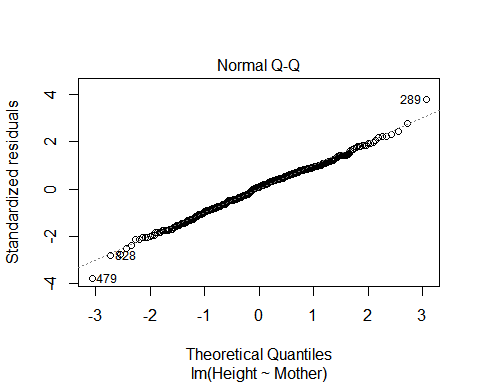
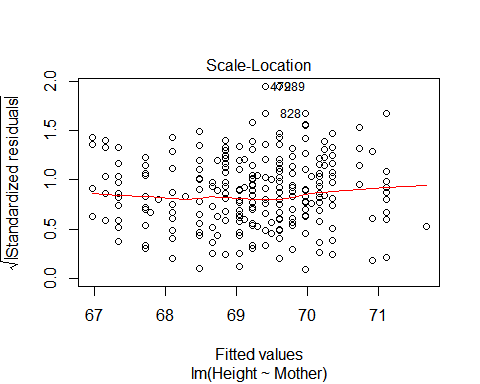
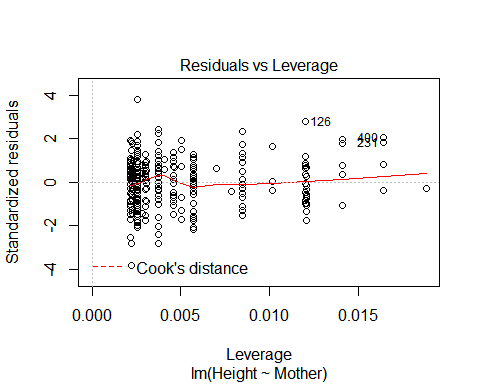
## 1 5 6 9 11   
## 0.0037780650 0.0041899035 0.0020520339 0.0005506893 0.0002839033   
## 12   
## 0.0000925431

Generating the residual plots can by done by passing the model to the plot function, plot(model)

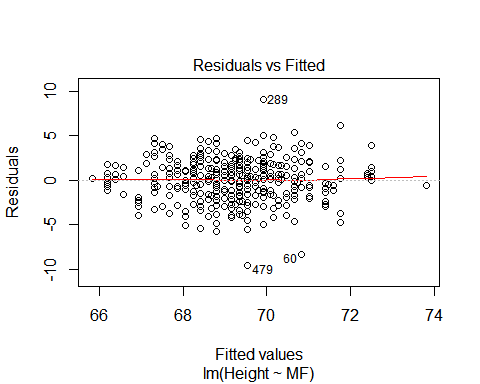
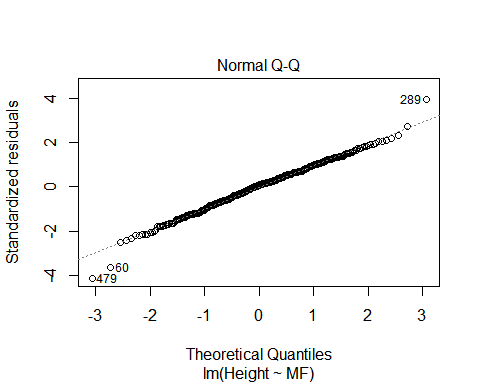
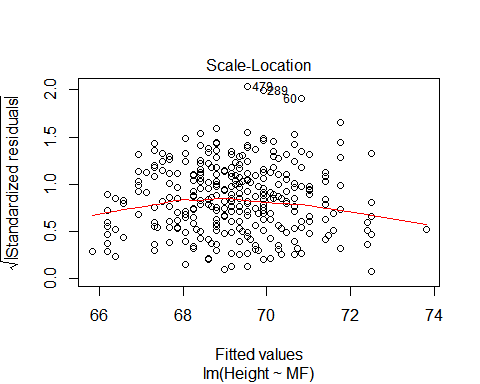
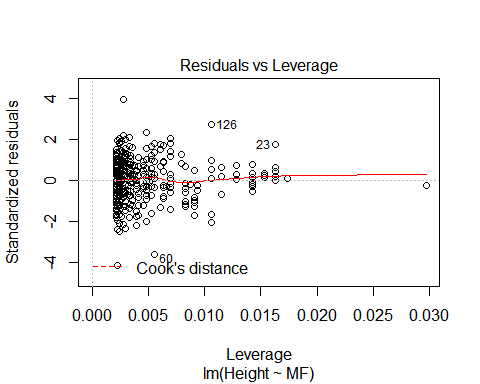
plot(m\_sons\_fathers)

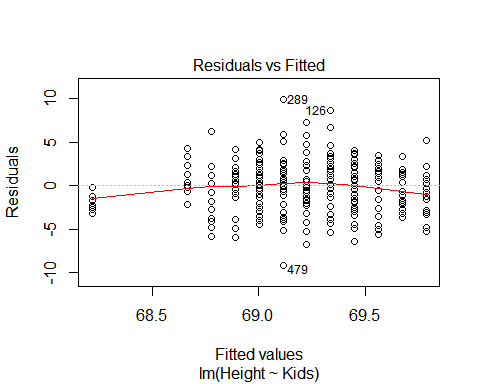
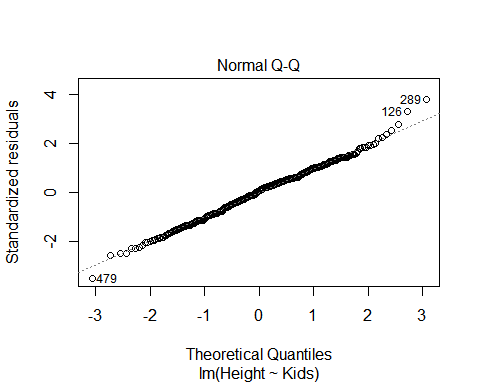
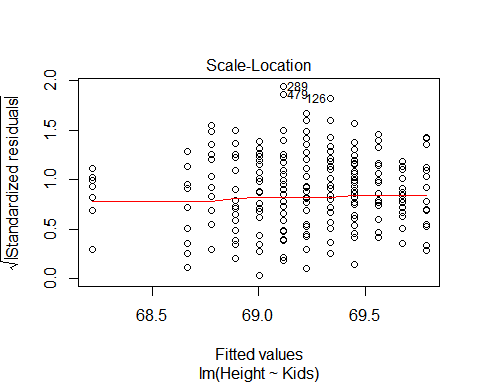
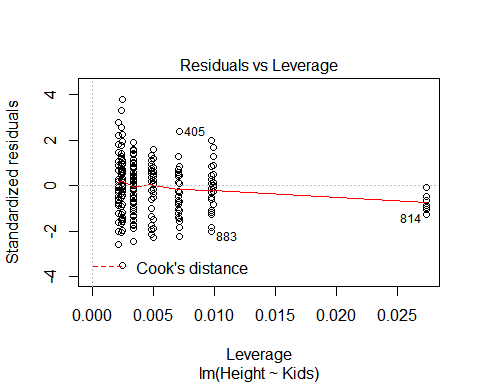
plot(m\_sons\_mothers)

plot(m\_sons\_mfs)

plot(m\_sons\_kids)

## ggplot for Residual Plots

Sometimes we want to make prettier, more detailed or alternate residual plots. It is easy to get data from out model object for plotting.

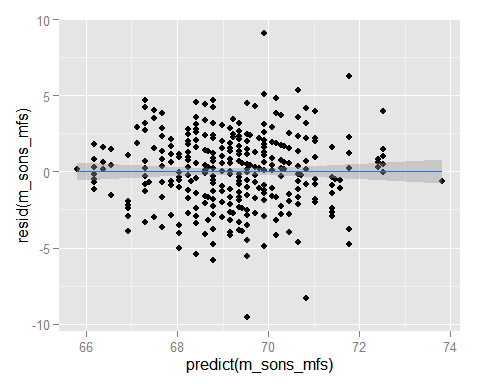
names(m\_sons\_mfs)

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

There are functions to extract data from model objects, which can be used to pass data to ggplot2. For example,

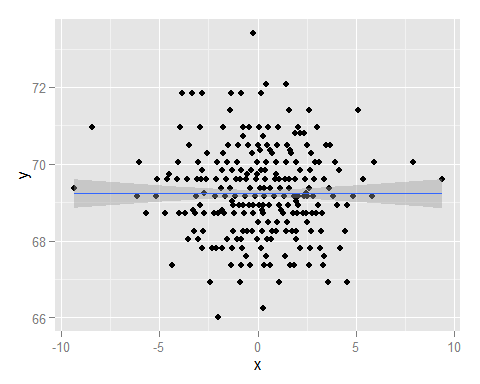
* predict(m\_sons\_mfs)
* resid(m\_sons\_mfs)
* coef(m\_sons\_mfs)
* lm.influence(m\_sons\_mothers)
* cooks.distance(m\_sons\_mothers)

qplot(predict(m\_sons\_mfs),resid(m\_sons\_mfs)) + geom\_smooth(method=lm)



The $ syntax can also be used (e.g. m\_sons\_mfs$residuals) to pass data to ggplot2.

qplot(x,y, data=data.frame(x=residuals(m\_sons\_fathers),y=predict(m\_sons\_fathers))) + geom\_smooth(method=lm)



## No y-intercept

You can run regression without the constraint of a y-intercept.

m\_sons\_fathers\_no\_y<-lm(Height ~ Father - 1,data=sons) # no y  
m\_sons\_fathers\_no\_y

##   
## Call:  
## lm(formula = Height ~ Father - 1, data = sons)  
##   
## Coefficients:  
## Father   
## 1

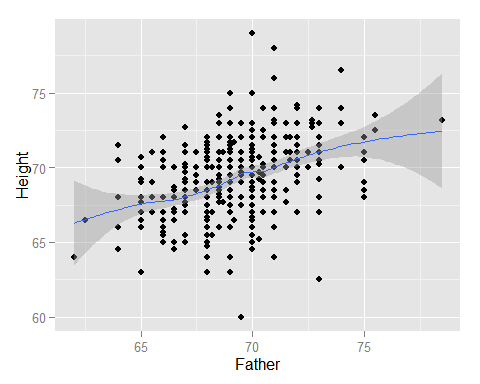
Or use an alternate syntax.

m\_sons\_fathers\_no\_y<-lm(Height ~ Father + 0,data=sons) # no y  
m\_sons\_fathers\_no\_y

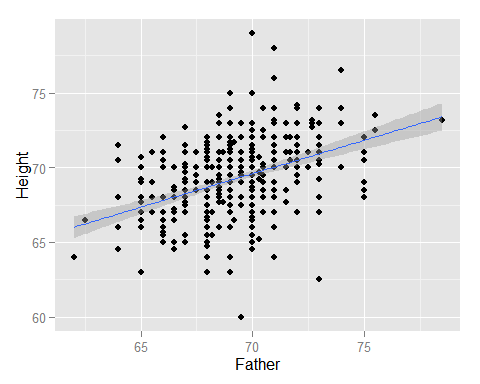
##   
## Call:  
## lm(formula = Height ~ Father + 0, data = sons)  
##   
## Coefficients:  
## Father   
## 1

We can also pass the model formula directly to ggplot2 to visualize the regression line.

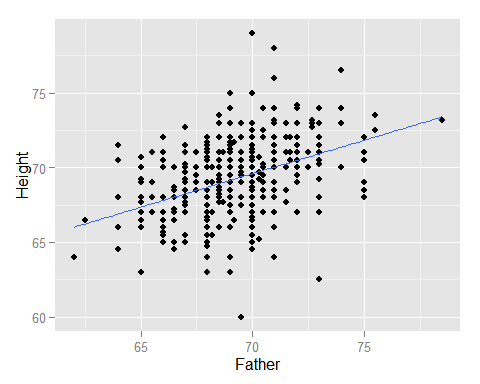
p<-qplot(Father,Height,data=sons)  
p+stat\_smooth()



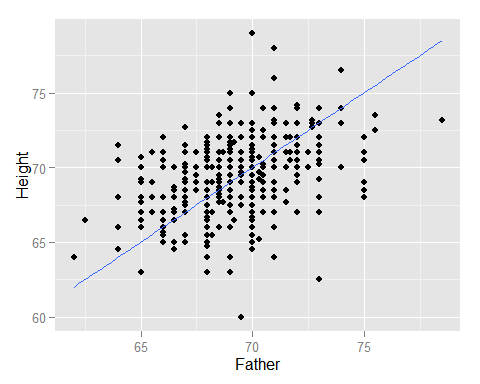
p + stat\_smooth(method = "lm")



p + stat\_smooth(method = "lm", formula = y ~ x, se=FALSE)



p + stat\_smooth(method = "lm", formula = y ~ x+0, se=FALSE)



## Multivariate Linear Regression

Multivariate linear regression (also known as *multiple linear regression*) is the extension of a single predictor variable to a set predictor variables, {}, that is

these n equations are stacked together and written in vector form as

Where , , and are vectors and is a matrix (sometimes called the design matrix).

m\_sons\_fathers\_mothers<-lm(Height ~ Father + Mother,data=sons)   
m\_sons\_fathers\_mothers

##   
## Call:  
## lm(formula = Height ~ Father + Mother, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother   
## 19.3999 0.4118 0.3336

anova(m\_sons\_fathers\_mothers)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 93.047 < 2.2e-16 \*\*\*  
## Mother 1 278.10 278.10 52.589 1.745e-12 \*\*\*  
## Residuals 462 2443.17 5.29   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Multiple testing bias

Note that when rejecting the null hypothesis at a p-value of 0.05 means the if we try 20 predcitors we would expect 1 to falsly reject the null hypothesis.

## Interaction Terms in R

An *interaction variable* is a variable constructed from an original set of variables to try to represent either all of the interaction present or some part of it. An *interaction variable* models the simultaneous influence of two predictors on a third response variable is not additive. We want to know whether the joint effect is higher than the sum of both effects.

For example, smoking and inhaling asbestos fibers both increase the risk of lung cancer, but exposure to asbestos and smoking multiplies the cancer risk. ere, the joint effect of inhaling asbestos and smoking is higher than the sum of both effects. from [*Relation between exposure to asbestos and smoking jointly and the risk of lung cancer*](http://oem.bmj.com/content/58/3/145)

Simple linear models assume that if a predictor variable affects the outcome variable, it does so in a way that is independent of all the other predictor variables.

We deal with non-independence of predictors by including interaction terms in our models. Interaction variables introduce an additional level of regression analysis by allowing researchers to explore the synergistic effects of combined predictors.

Interaction variables can be: \* interaction between two categorical variables, \* interaction between one continuous and one categorical variables, \* and the interaction between two continuous variables.

g\_interaction<-lm(Height ~ Father + Mother + Father:Mother, data=sons)  
g\_interaction

##   
## Call:  
## lm(formula = Height ~ Father + Mother + Father:Mother, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother Father:Mother   
## 57.857809 -0.143259 -0.266468 0.008658

anova(g\_interaction)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 92.8910 < 2.2e-16 \*\*\*  
## Mother 1 278.10 278.10 52.5008 1.822e-12 \*\*\*  
## Father:Mother 1 1.20 1.20 0.2262 0.6346   
## Residuals 461 2441.98 5.30   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The same model can be specified with a simplier syntax. 'Father\*Mother' is the same as 'Father + Mother + Father:Mother'.

g\_interaction<-lm(Height ~ Father\*Mother, data=sons)  
g\_interaction

##   
## Call:  
## lm(formula = Height ~ Father \* Mother, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother Father:Mother   
## 57.857809 -0.143259 -0.266468 0.008658

anova(g\_interaction)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 92.8910 < 2.2e-16 \*\*\*  
## Mother 1 278.10 278.10 52.5008 1.822e-12 \*\*\*  
## Father:Mother 1 1.20 1.20 0.2262 0.6346   
## Residuals 461 2441.98 5.30   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The same model can be specified with an exponential syntax as well. 'Father\*Mother' is the same as '(Father + Mother)^2'.

g\_interaction<-lm(Height ~ (Father + Mother)^2, data=sons)  
g\_interaction

##   
## Call:  
## lm(formula = Height ~ (Father + Mother)^2, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother Father:Mother   
## 57.857809 -0.143259 -0.266468 0.008658

anova(g\_interaction)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 92.8910 < 2.2e-16 \*\*\*  
## Mother 1 278.10 278.10 52.5008 1.822e-12 \*\*\*  
## Father:Mother 1 1.20 1.20 0.2262 0.6346   
## Residuals 461 2441.98 5.30   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

\_ higher (nth) order interactions\_

The exponential syntax also allows higher order interactions to be specified.

g\_interaction\_3<-lm(Height ~ (Father + Mother + Kids)^3, data=sons)  
g\_interaction\_3

##   
## Call:  
## lm(formula = Height ~ (Father + Mother + Kids)^3, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother   
## -337.00076 5.81499 5.90853   
## Kids Father:Mother Father:Kids   
## 75.14270 -0.08438 -1.13250   
## Mother:Kids Father:Mother:Kids   
## -1.17478 0.01768

anova(g\_interaction\_3)

## Analysis of Variance Table  
##   
## Response: Height  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Father 1 492.06 492.06 94.6754 < 2.2e-16 \*\*\*  
## Mother 1 278.10 278.10 53.5093 1.164e-12 \*\*\*  
## Kids 1 19.82 19.82 3.8135 0.05145 .   
## Father:Mother 1 1.71 1.71 0.3289 0.56658   
## Father:Kids 1 0.73 0.73 0.1413 0.70720   
## Mother:Kids 1 15.21 15.21 2.9274 0.08777 .   
## Father:Mother:Kids 1 30.53 30.53 5.8751 0.01574 \*   
## Residuals 457 2375.16 5.20   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Multi-colinearity

In statistics, multicollinearity (also collinearity) is a phenomenon in which two or more predictor variables in a multiple regression model are highly correlated, meaning that one can be linearly predicted from the others with a non-trivial degree of accuracy. In this situation the coefficient estimates of the multiple regression may change erratically in response to small changes in the model or the data. Multicollinearity does not reduce the predictive power or reliability of the model as a whole, at least within the sample data set; it only affects calculations regarding individual predictors. That is, a multiple regression model with correlated predictors can indicate how well the entire bundle of predictors predicts the outcome variable, but it may not give valid results about any individual predictor, or about which predictors are redundant with respect to others.

* from [multicollinearity - Wikipedia](https://en.wikipedia.org/wiki/Multicollinearity)

cor(sons$Height,sons$Father)

## [1] 0.3913174

cor(sons$Height,sons$Mother)

## [1] 0.3341309

cor(sons$Father,sons$Mother)

## [1] 0.1063346

cor(galton$Height,as.numeric(galton$Gender))

## [1] 0.7142578

## Stepwise regression

Stepwise regression means to iteratively select the best predictor (that improves the model the most), then the next best until we have no predictors that improves the model or use all of the predictors. This is also called forward stepwise selection.

A variant called *backward elimination*, involves starting with all candidate predictors, testing the deletion of each variable using a chosen model comparison criterion, deleting the variable (if any) that improves the model the most by being deleted, and repeating this process until no further improvement is possible.

Bidirectional elimination, a combination of the above, testing at each step for variables to be included or excluded.

beg<-lm(Height ~ Father, data=sons)  
end<-lm(Height ~ ., data=sons)  
empty<-lm(Height ~ 1, data=sons)  
bounds<-list(upper=end,lower=empty)  
stepwise\_reg<-step(beg,bounds, direction = "forward")

## Start: AIC=825.57  
## Height ~ Father  
##   
## Df Sum of Sq RSS AIC  
## + Mother 1 278.104 2443.2 777.44  
## + MF 1 278.104 2443.2 777.44  
## + Kids 1 19.044 2702.2 824.31  
## <none> 2721.3 825.57  
##   
## Step: AIC=777.44  
## Height ~ Father + Mother  
##   
## Df Sum of Sq RSS AIC  
## + Kids 1 19.82 2423.3 775.65  
## <none> 2443.2 777.44  
##   
## Step: AIC=775.65  
## Height ~ Father + Mother + Kids  
##   
## Df Sum of Sq RSS AIC  
## <none> 2423.3 775.65

stepwise\_reg

##   
## Call:  
## lm(formula = Height ~ Father + Mother + Kids, data = sons)  
##   
## Coefficients:  
## (Intercept) Father Mother Kids   
## 20.41215 0.40346 0.33402 -0.07856

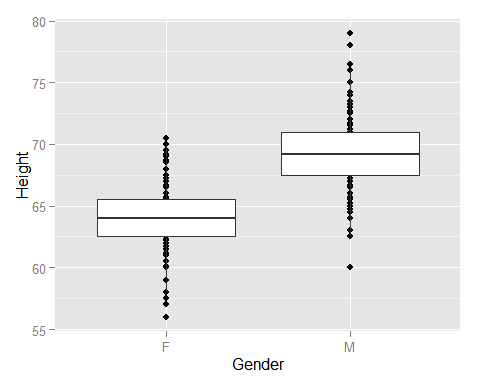
# Logistic regression

[Logistic regression](https://en.wikipedia.org/wiki/Logistic_regression), or logit regression, or is a regression model where the outcome variable is categorical. Often this is used when the variable is binary (e.g. yes/no, survived/dead, pass/fail, etc.)

Logistic regression measures the relationship between the categorical response variable and one or more predictor variables by estimating probabilities.

*Can height predict gender?*

qplot(Gender,Height,data=galton)+geom\_boxplot()



Note that logistic regression estimates probabilities.

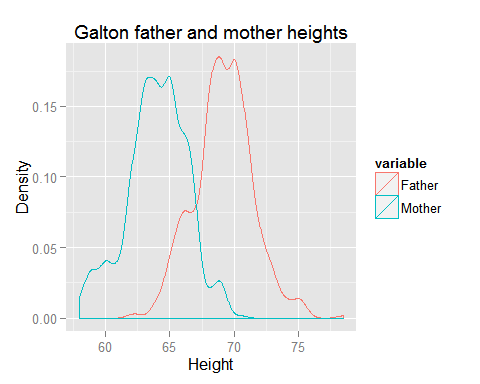
set.seed(333)  
require(reshape2)

## Loading required package: reshape2

norm\_plot<-sons[c("Father","Mother")]  
rnd <- melt(data=norm\_plot)

## No id variables; using all as measure variables

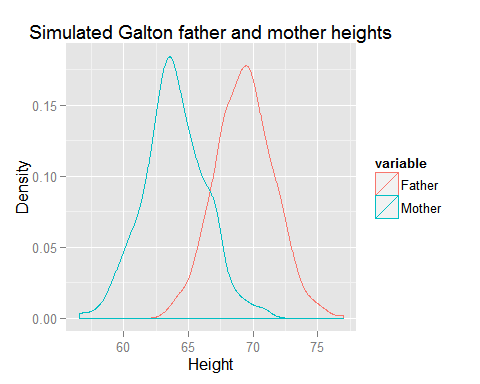
qplot(x=value, data=rnd, geom="density", group=variable, color=variable) + labs(title="Galton father and mother heights", y="Density", x="Height")



h\_normal <-data.frame(Father=rnorm(n=length(sons$Height),mean=mean(sons$Father),sd=sd(sons$Father)),Mother=rnorm(n=length(sons$Height),mean=mean(sons$Mother),sd=sd(sons$Mother)))  
rnd <- melt(data=h\_normal)

## No id variables; using all as measure variables

qplot(x=value, data=rnd, geom="density", group=variable, color=variable) + labs(title="Simulated Galton father and mother heights", y="Density", x="Height")



Can height predict gender?

m\_gender\_logistic<-lm(as.numeric(Gender) ~ Height, data=galton)  
m\_gender\_logistic

##   
## Call:  
## lm(formula = as.numeric(Gender) ~ Height, data = galton)  
##   
## Coefficients:  
## (Intercept) Height   
## -5.13606 0.09967

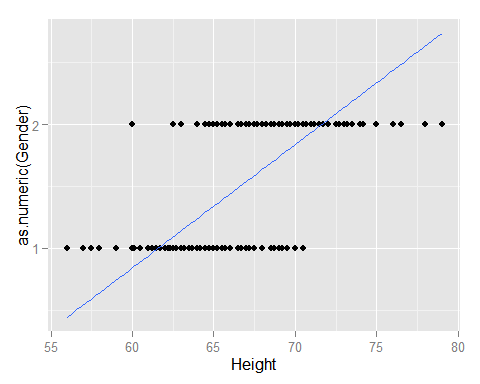
summary(m\_gender\_logistic)

##   
## Call:  
## lm(formula = as.numeric(Gender) ~ Height, data = galton)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.89051 -0.24267 -0.04001 0.25900 1.15600   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.136062 0.218130 -23.55 <2e-16 \*\*\*  
## Height 0.099668 0.003263 30.55 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3501 on 896 degrees of freedom  
## Multiple R-squared: 0.5102, Adjusted R-squared: 0.5096   
## F-statistic: 933.2 on 1 and 896 DF, p-value: < 2.2e-16

anova(m\_gender\_logistic)

## Analysis of Variance Table  
##   
## Response: as.numeric(Gender)  
## Df Sum Sq Mean Sq F value Pr(>F)   
## Height 1 114.39 114.386 933.18 < 2.2e-16 \*\*\*  
## Residuals 896 109.83 0.123   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

qplot(Height,as.numeric(Gender),data=galton) + stat\_smooth(method = "lm", formula = y ~ x, se=FALSE)



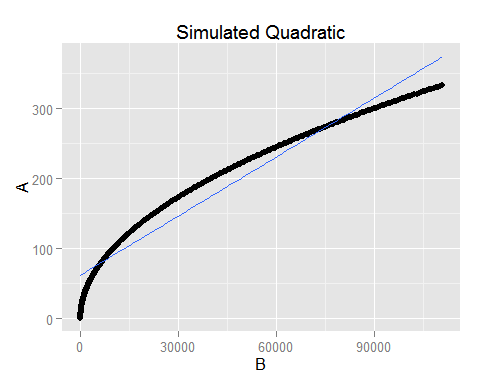
## Non-linear transformations of predictors

Even if the data is non-linear one might be able to use a tranform so it has a linear form and simple linear regression can be applied.

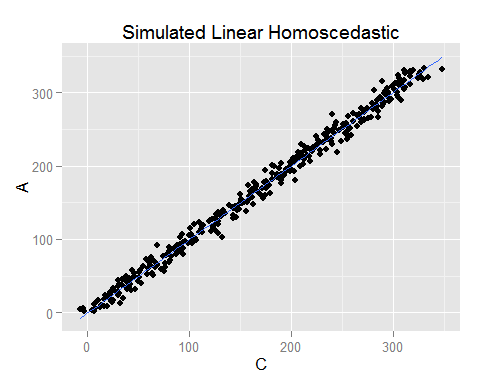
set.seed(3333)  
# create some random data  
trails<-333  
n<-1:trails  
n2<-n^2  
r\_data <-data.frame(A=n,  
 B=n2,  
 C=n+rnorm(n=trails,sd=9),  
 D=n2+rnorm(n=trails,sd=999))  
head(r\_data)

## A B C D  
## 1 1 1 6.659495 -1283.1624  
## 2 2 4 -3.220798 1089.1395  
## 3 3 9 4.853541 -935.6342  
## 4 4 16 -7.090667 -1163.5039  
## 5 5 25 7.484705 211.0381  
## 6 6 36 -4.279625 -485.2760

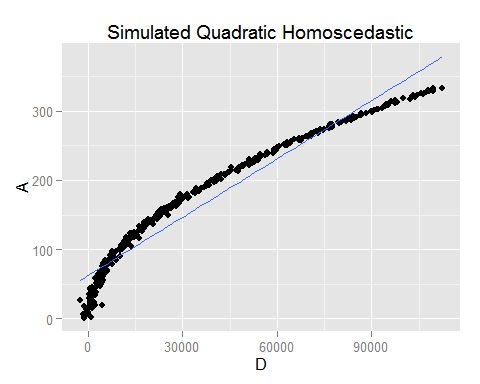
qplot(B,A,data=r\_data,main="Simulated Quadratic") + geom\_smooth(method=lm,se=FALSE)



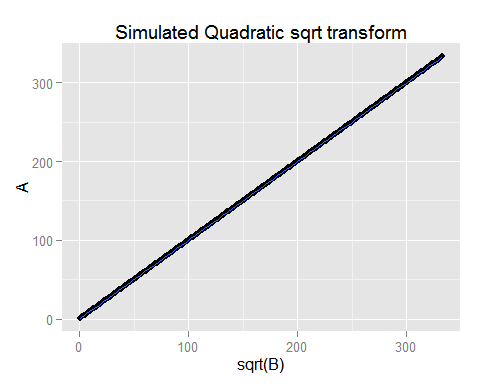
qplot(C,A,data=r\_data, main="Simulated Linear Homoscedastic") + geom\_smooth(method=lm,se=FALSE)



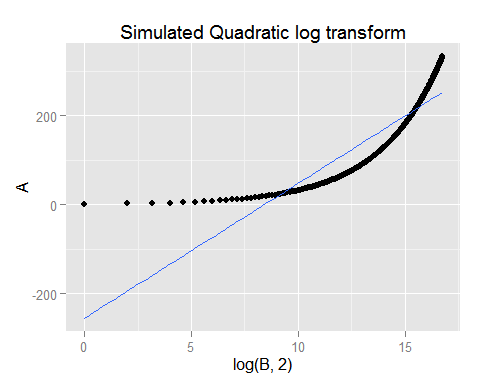
qplot(D,A,data=r\_data, main="Simulated Quadratic Homoscedastic") + geom\_smooth(method=lm,se=FALSE)



qplot(sqrt(B),A,data=r\_data, main="Simulated Quadratic sqrt transform") + geom\_smooth(method=lm,se=FALSE)



qplot(log(B,2),A,data=r\_data, main="Simulated Quadratic log transform") + geom\_smooth(method=lm,se=FALSE)



qplot(sqrt(D),A,data=r\_data, main="Simulated Quadratic Homoscedastic sqrt transform") + geom\_smooth(method=lm,se=FALSE)

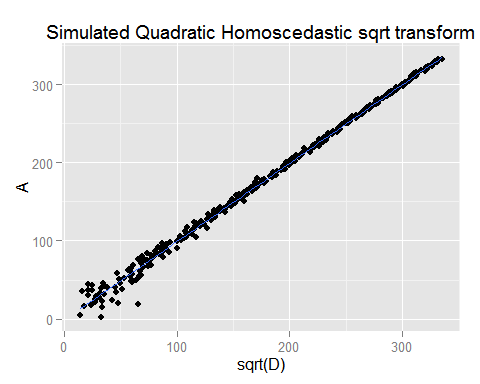
## Warning in sqrt(D): NaNs produced

## Warning in sqrt(D): NaNs produced

## Warning in sqrt(D): NaNs produced

## Warning in loop\_apply(n, do.ply): Removed 14 rows containing missing values  
## (stat\_smooth).

## Warning in loop\_apply(n, do.ply): Removed 14 rows containing missing values  
## (geom\_point).



qplot(log(D,2),A,data=r\_data, main="Simulated Quadratic Homoscedastic log transform") + geom\_smooth(method=lm,se=FALSE)

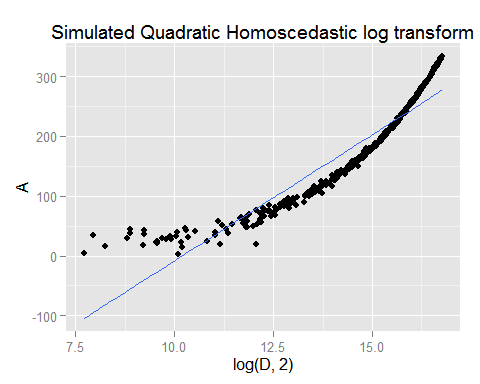
## Warning: NaNs produced

## Warning: NaNs produced

## Warning: NaNs produced

## Warning in loop\_apply(n, do.ply): Removed 14 rows containing missing values  
## (stat\_smooth).

## Warning in loop\_apply(n, do.ply): Removed 14 rows containing missing values  
## (geom\_point).



## Assingment

* Load the file M01\_quasi\_twitter.csv (it is online at '<http://nikbearbrown.com/YouTube/MachineLearning/M01/M01_quasi_twitter.csv>')
* Generate a linear model for the following:
  + A relation between followers\_count & gender
  + A relation between dob\_year & statuses\_count
  + A significant linear linear model of your choosing.
  + A multivariate relation between wage & height, race, age, education & experience
  + A significant logistic linear model of your choosing.  
    Answer the following questions:
  + Is the relationship significant?
  + Are any model assumptions violated?
  + Is there any multi-colinearity in multivariate models?
  + In in multivariate models are predictor variables independent of all the other predictor variables?
  + In in multivariate models rank the most significant predictor variables and exclude insignificant one from the model.
  + Does the model make sense? Write up your report as an .Rmd file.

## Resources

* [Simple Linear Regression | R Tutorial](http://ww2.coastal.edu/kingw/statistics/R-tutorials/simplelinear.html)