CANCER RESEARCH UK | Cambridge Institute

Together we are beating cancer

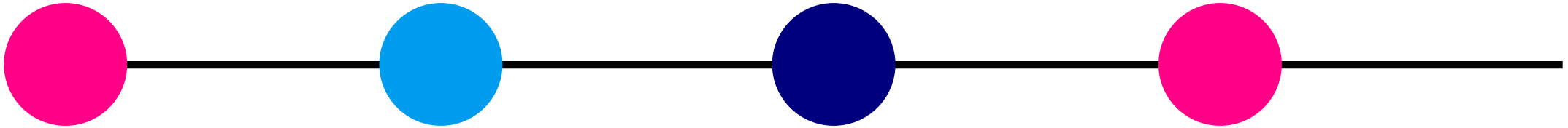
Luca Porcu & Chandra Chilamakuri (Bioinformatics core)

21st February 2025

Linear regression models

Fixed-effects models

Process flow



Stage 1

Effects and regression models

Stage 2

Analysis of variance (ANOVA)

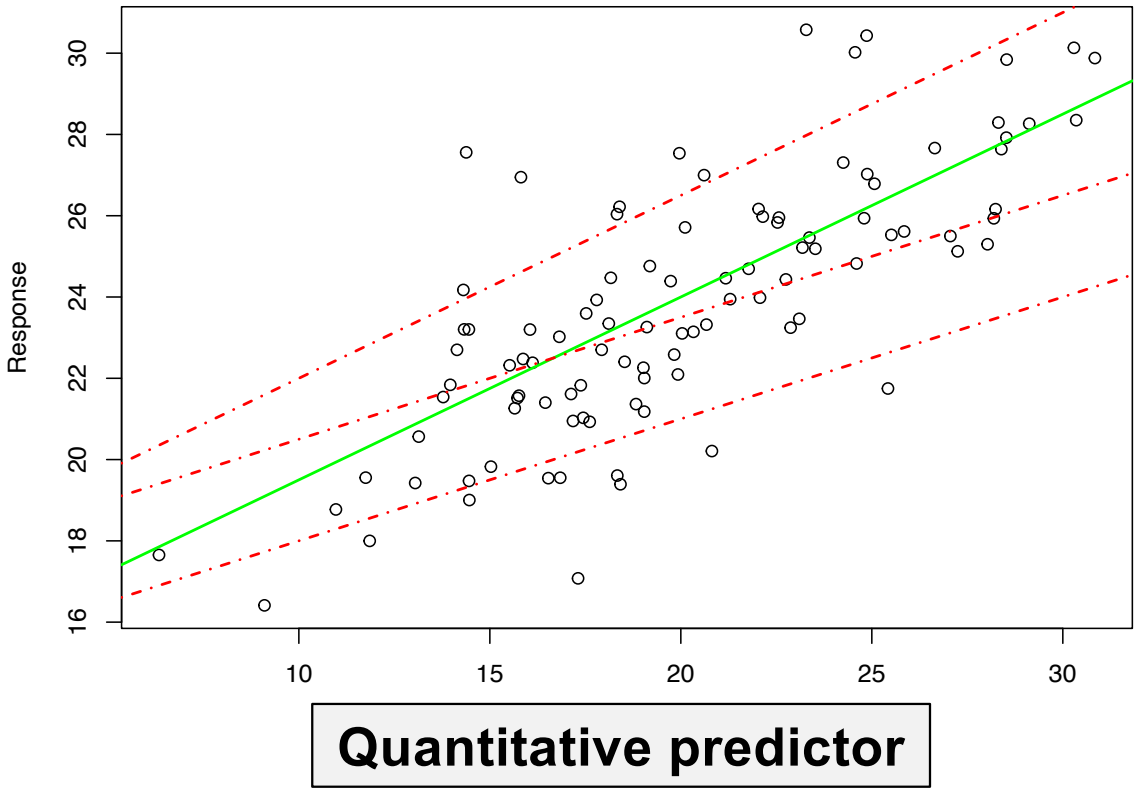
Stage 3

Simple regression model

Stage 4

Multivariable regression model

CANCER RESEARCH UK | Cambridge Institute



Simple regression model

Definition and classification

11.00 - 11.20 am

Together we are beating cancer

Simple regression model

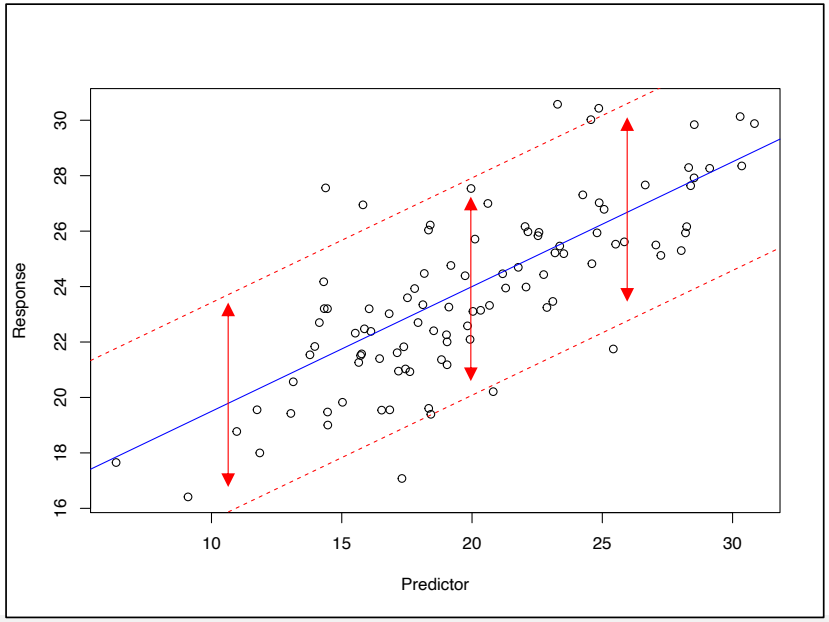
1. The unit k (e.g. mouse), k = 1,…,n

2. β_0 : intercept of the model; β_1: slope of the model (i.e. effect of predictor X)

3. ε_k: the *random* part of the model (i.e. error term of the model). It is a blanket characterization of the uniqueness of the k_{th} unit

Equation of the statistical model:

$$y_k = \beta_0 + \beta_1 x_k + \varepsilon_k$$



n =100

Systematic component

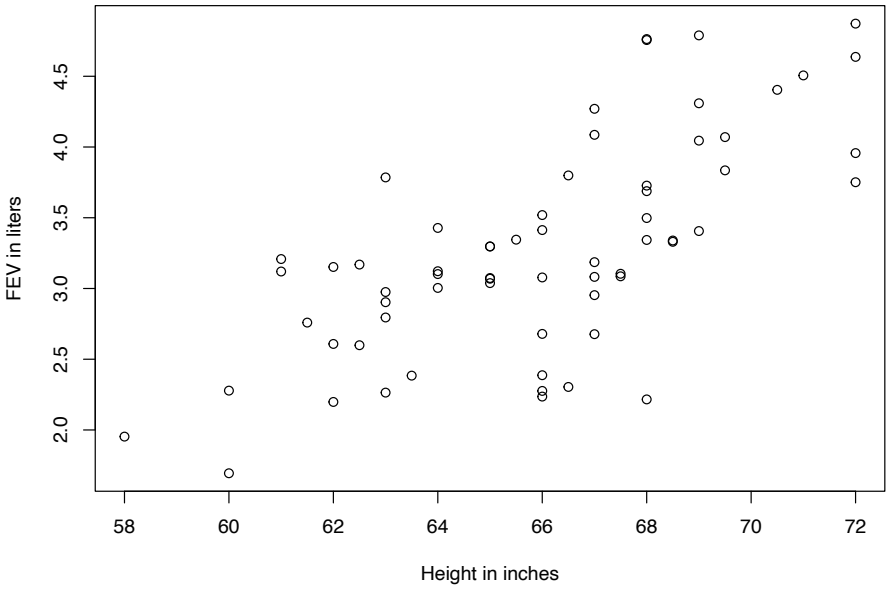
+/- 1.96 · SD

Assumptions of simple regression models are the following:

- The true relationship between μ_Y and the *quantitative* predictor X is linear
- ε_k is assumed to be independent of one another and normally distributed with mean = 0 and common standard deviation = σ

Estimation of parameters β_0, β_1 and σ


```
> library(GLMsData) # Load the GLMsData package
> data(lungcap) # Make the dataset lungcup available
> head(lungcap) # Show the first observations
> dSet = subset(lungcap, Smoke==1) # Select smokers
> plot(dSet$Ht, dSet$FEV, xlab = "Height in inches", ylab = "FEV") # Scatter plot
```



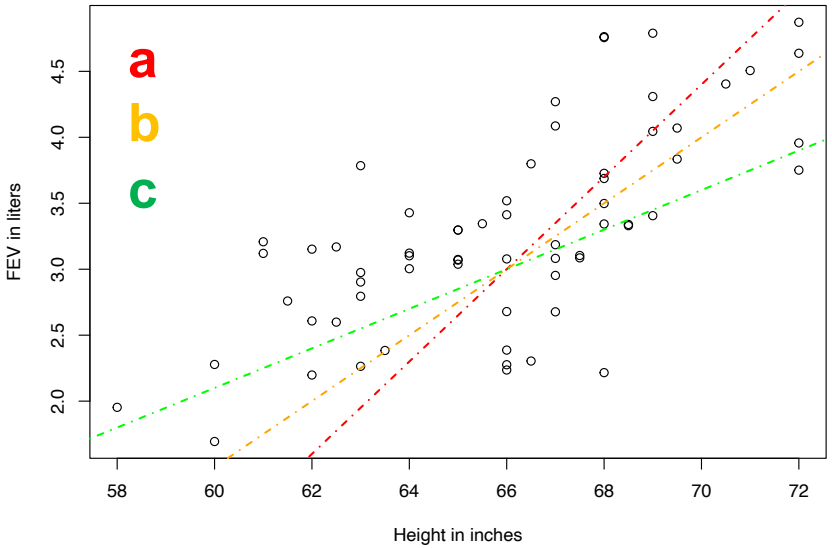
A simple regression model is reasonable

Estimation of parameters β_0, β_1 and σ


```
> beta1a = 0.35; beta1b = 0.25; beta1c = 0.15; # Plausible slopes
> beta0a = -20.1; beta0b = -13.5; beta0c = -6.9; # Plausible intercepts (x=66; μY ≃ 3.0)
> RSSi = sum((dSet$FEV - (beta0i + beta1i * dSet$Ht))^2) # i = a, b, c
> print(c(RSSa, RSSb, RSSc) # Show residual sum-of-squares (RSS)
```

Output

```
[1] 47.55759        29.36609        24.22459
```



- β_0 and β_1 estimators are those values that minimize RSS
- σ^2 estimator is given by $\frac{RSS}{N-2}$

Hypothesis testing and estimation in R


```
> fittedModel = lm(FEV ~ Ht, data = dSet) # Model fitting
> summary(fittedModel); # Output
```

Output

Coefficients:

	Estimate	Std. Error	t value	**Pr(>|t|)**
(Intercept)	-7.40165	1.41588	-5.228	2.07e-06 ***
Ht	0.16191	0.02144	7.551	2.18e-10 ***

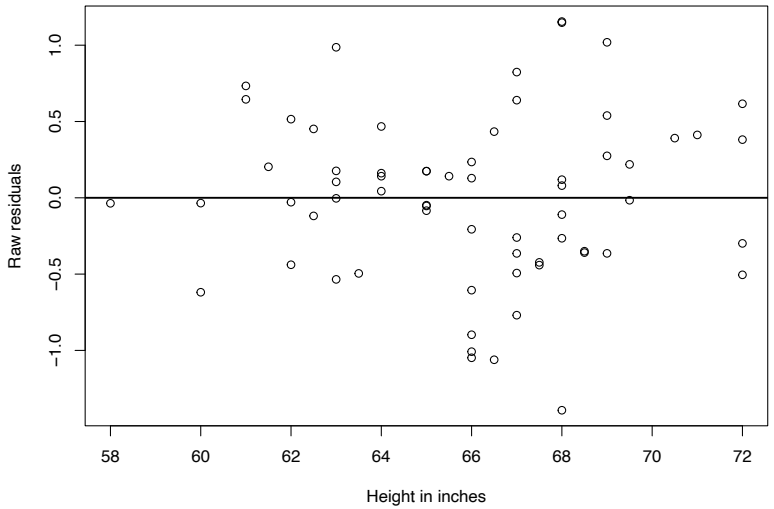
```
> confint(fittedModel) # 95% confidence intervals of β0 and β1
```

Output

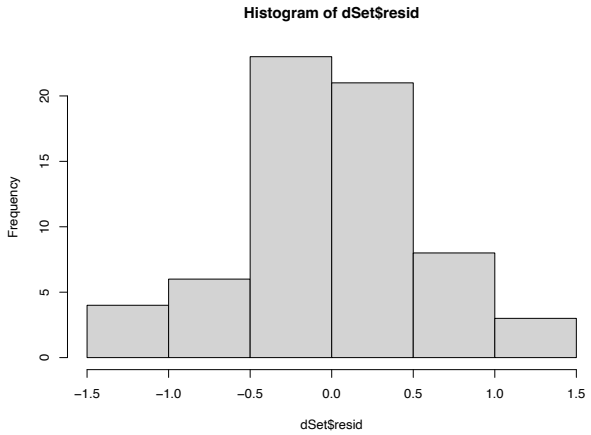
	2.5 %	**97.5 %**
(Intercept)	-10.2310609	-4.5722344
Ht	0.1190583	0.2047593

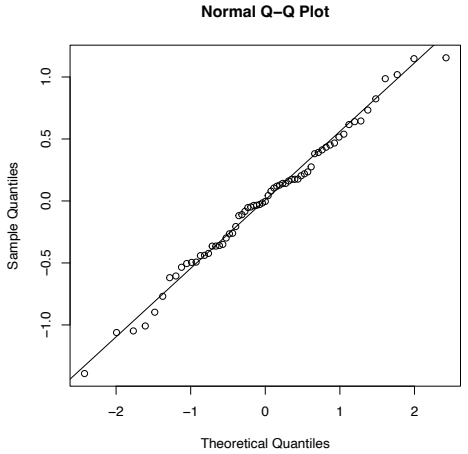
Diagnostics: residuals


```
> dSet$resid = resid(fittedModel) # Raw residuals (difference between the observed data of the dependent variable Y and the fitted value)
> plot(dSet$Ht, dSet$resid, xlab = "Height in inches", ylab = "Raw residuals") # Plotting residuals against predictor
> hist(dSet$resid) # Histogram
> qqnorm(dSet$resid); qqline(dSet$resid) # Q-Q plot
> shapiro.test(dSet$resid); # Shapiro-Wilk normality test
```



We can detect heteroscedasticity and asymmetrical distribution of residuals





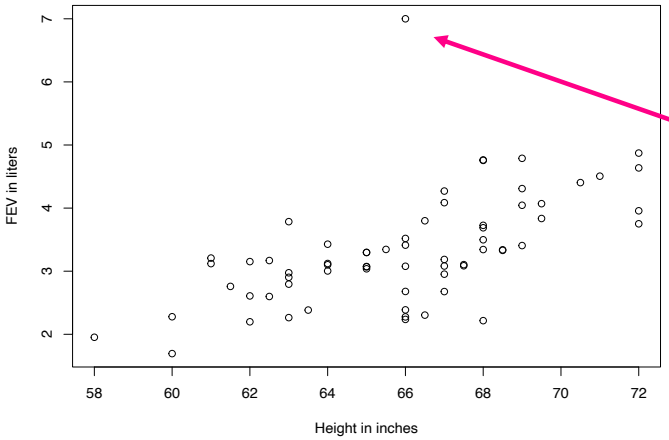
Shapiro-Wilk normality test

data: dSet$resid
W = 0.98933, p-value = 0.8509

We can check normality assumption of residuals

Outliers observations


```
> outlier = data.frame("Age"=14, "FEV"=7.000, "Ht"=66.0, "Gender"="M", "Smoke"=1) # Outlier
> dSetOutlier = rbind(dSet,outlier) # Adding the outlier to the data set
> plot(dSetOutlier$Ht, dSetOutlier$FEV, xlab = "Height in inches", ylab = "FEV in liters") # New scatter plot
```



Outlier

Question: is it an influential observation?

```
> fittedModel = lm(FEV ~ Ht, data = dSetOutlier) # Model fitting
> summary(fittedModel); # Output
```

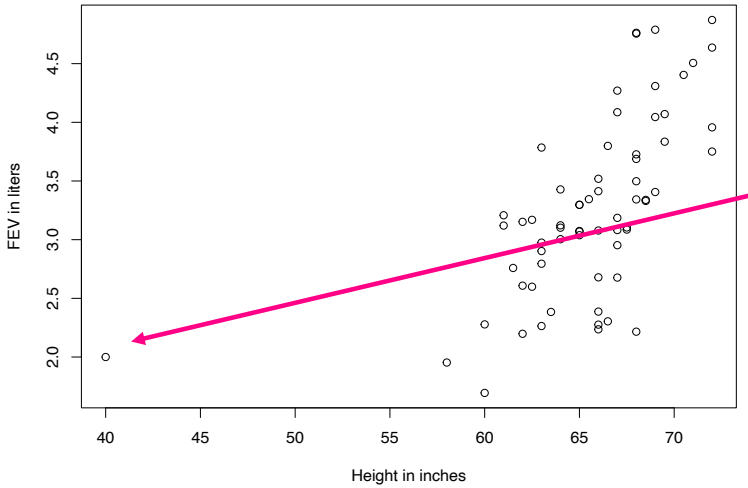
Output

Coefficients:

	Estimate	Std. Error	t value	**Pr(>|t|)**
(Intercept)	*(-7.40165)* -7.3624	*(1.41588)* 1.8421	*(-5.228)* -3.997	*(2.07e-06 ***)* 0.000169 ***
Ht	*(0.16191)* 0.1622	*(0.02144)* 0.0279	*(7.551)* 5.813	*(2.18e-10 ***)* 2.11e-07 ***

Influential observations


```
> InfObs = data.frame("Age"=14, "FEV"=2.000, "Ht"=40.0, "Gender"="M", "Smoke"=1) # Influential observation
> dSetInfObs = rbind(dSet,InfObs) # Adding the influential observation to the data set
> plot(dSetInfObs$Ht, dSetInfObs$FEV, xlab = "Height in inches", ylab = "FEV in liters") # New scatter plot
```



Outlier and far away x-value observation

Question: is it an influential observation?

```
> fittedModel = lm(FEV ~ Ht, data = dSetInfObs) # Model fitting
> summary(fittedModel); # Output
```

Output

Coefficients:

	Estimate	Std. Error	t value	**Pr(>|t|)**
(Intercept)	*(-7.40165)* -3.63162	*(1.41588)* 1.08780	*(-5.228)* -3.339	*(2.07e-06 ***)* 0.00141 ***
Ht	*(0.16191)* 0.10508	*(0.02144)* 0.01655	*(7.551)* 6.348	*(2.18e-10 ***)* 2.55e-08 ***

Observations with high leverage

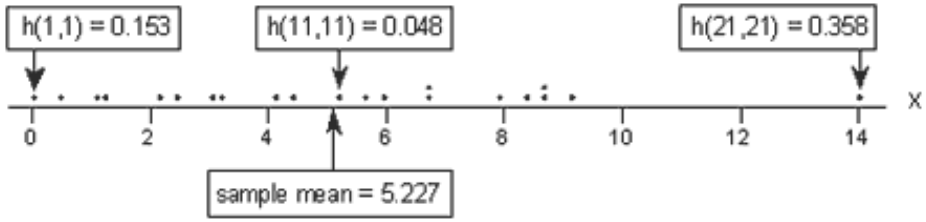
An observation is considered to have **high leverage** if it has a value (or values) for the predictor variables that are much more extreme compared to the rest of the observations in the dataset.

$$h_i = \frac{1}{n} + (x_i - x_m)^2 / SSx$$

```
> hats = hatvalues(modelFitted) # Computing leverage of each observation
```

h_i	leverage of observation i
x_i	predictor value of observation i
x_m	sample mean of the predictor
SS_x	sum of the squared difference between the predictor value of observation i and the sample mean of the predictor
n	number of observation

Large values of h_i (perhaps two or three times the mean value of the h_i) identify observations with unusual values of the predictor



Influential observations

Influential observations are **outliers** with **high leverage**.

A popular measure of influence for observation i is **Cook's distance**:

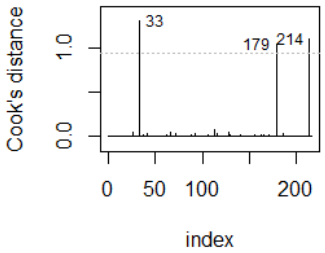
$D_i \div (\Delta_i, h_i)$

```
> cookDistance = cook.distance(modelFitted)
# Computing Cook's distance of each observation
```

h_i	leverage of observation i
Δ_i	Sum of the squared differences between predictions made with all observations in the analysis and predictions made leaving out the observation in question

A rule of thumb is that observations with D > 1 may be flagged as potentially influential.

```
> cook_plot(modelFitted, id_n = 3)
# Computing Cook's distance of each observation
```



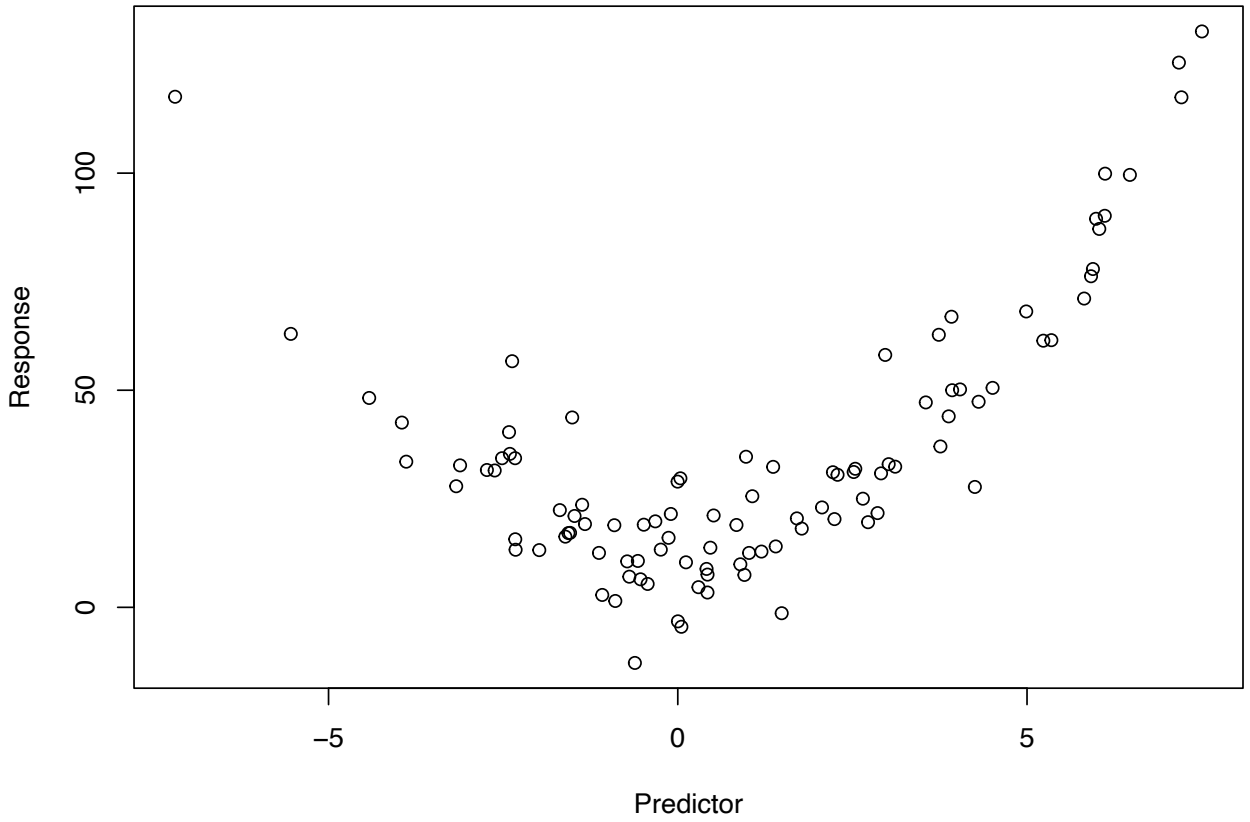
Source of problems and possible solutions

Solution	Linearity	Normality	Unequal variance	Outliers	Influential observations
Weighting			✓	✓	✓
Introducing other predictors (multivariable model)	✓	✓	✓	✓	✓
Data transformation	✓	✓	✓	✓	✓
Different random component (e.g. gamma, beta, Poisson) and g($\boldsymbol{\mu}$)		✓	✓	✓	✓

Introducing other predictors

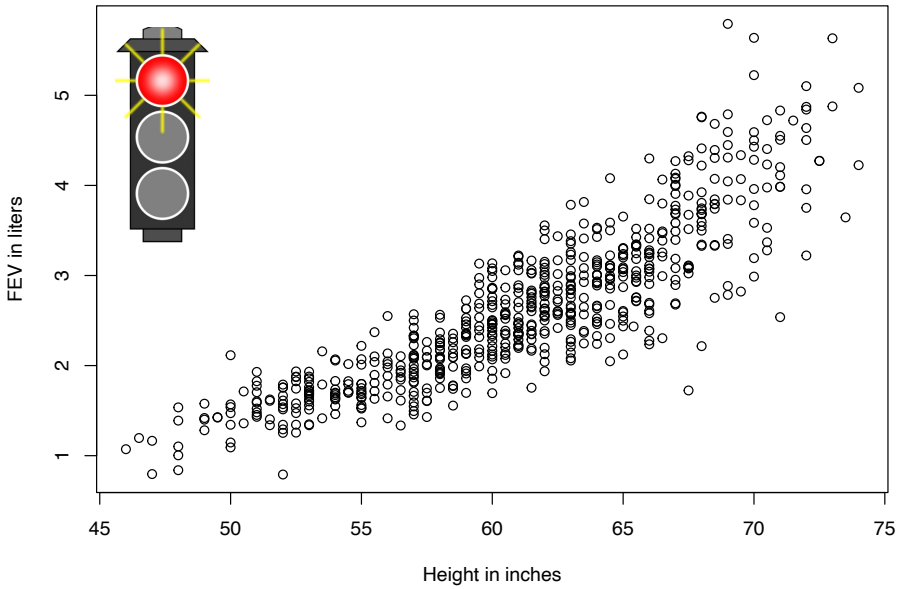
Equation of a plausible statistical model:

$$y_k = \beta_0 + \beta_1 x_k + \beta_2 x_k^2 + \varepsilon_k$$

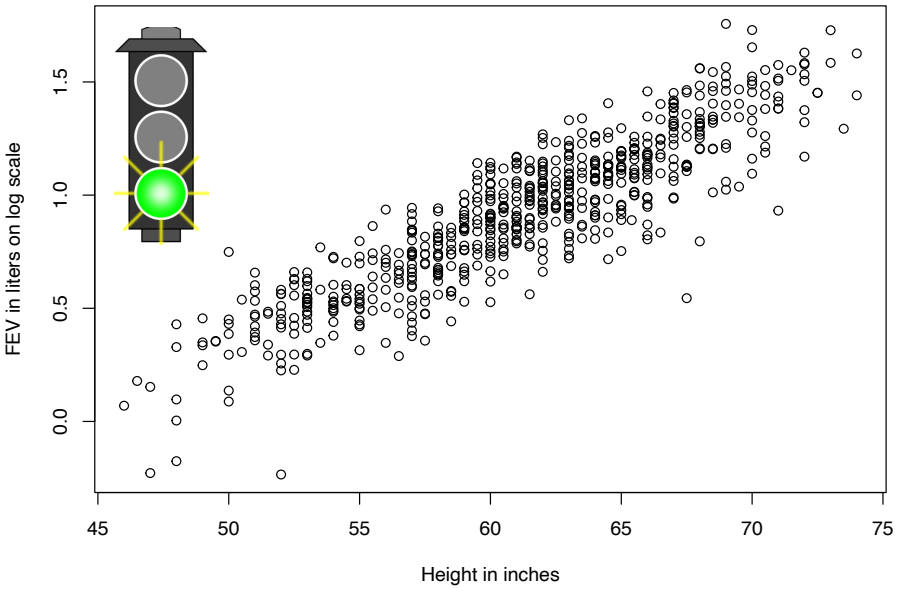


Data transformation


```
> plot(lungcap$Ht, lungcap$FEV, xlab = "Height in inches", ylab = "FEV") # Scatter plot
> lungcap$logFEV = log(lungcap$FEV) # log transformation of the response variable
> plot(dSet$Ht, dSet$logFEV, xlab = "Height in inches", ylab = "FEV on log scale") # Scatter plot
```



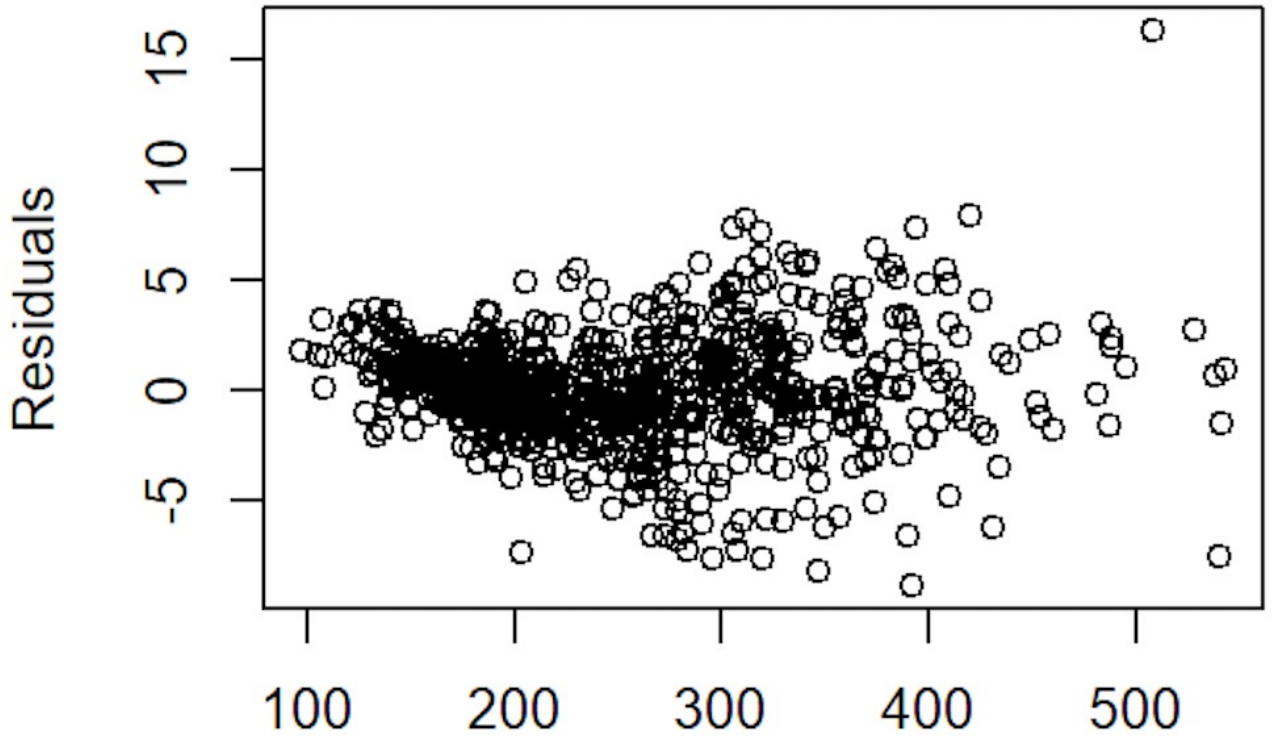
Response on natural scale



Response on $\log_e$ scale

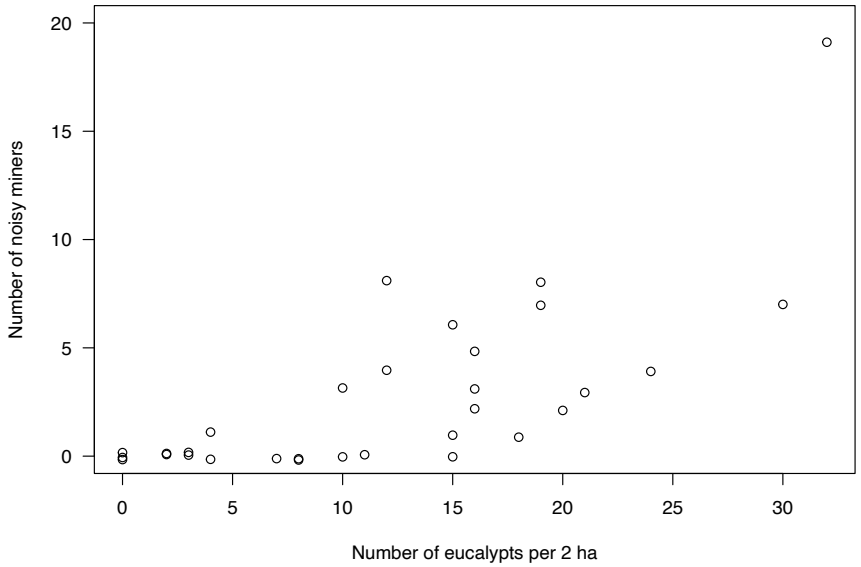
Weighted least square


```
> library(nlme) # Load the nlme package
> fittedModel = gls(Response ~ Predictor, weight = varPower( ~ Predictor) , data = dSet)
    # Modeling variance as function of the predictor
> summary(fittedModel) # Output
```



Different random component and g(μ)


```
> library(GLMsData) # Load the GLMsData package
> data(nminer) # Make the dataset nminer available
> plot(jitter(Minerab) ~ Eucs, data=nminer, las=1, ylim=c(0, 20), xlab="Number of eucalypts per 2 ha",
ylab="Number of noisy miners" ) # The number of noisy miners plotted against the number of eucalypt trees
```



Y = Poisson(μ)	Random component
log (μ) = $\beta_0 + \beta_1 x$	Systematic component

https://bioinformatics-core-shared-training.github.io/Fixed-and-Mixed-effects-models/