Linear models

Simon Wood
Mathematical Sciences, University of Bath, U.K.

Linear models

- We have data on a *response variable*, y, the variability in which is believed to be partly predicted by data on some *predictor variables*, $x_1, x_2 \ldots$.
- We model this using a *linear model*

$$y_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \ldots + x_{im}\beta_m + \epsilon_i$$

  - The *parameters*, β_j, must be estimated from data
  - The *random variables*, ϵ_i, account for the variability in the response not explained by the predictors
  - Assumptions: the ϵ_i's have zero mean ($\mathbb{E}(\epsilon_i) = 0$) and constant variance σ^2. They are also independent: knowing the value of ϵ_i tells you nothing new about that value of $\epsilon_{j \neq i}$.

Linear model features

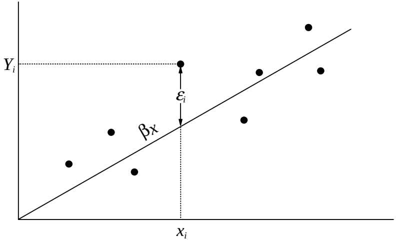
- A key difference in kind between β_j's and ϵ_i's is this: if a replicate data set were generated the β_j's would be the same, but the ϵ_i's would all be different.
- For some purposes (H_0 testing etc.) we assume that the ϵ_i's are Normally distributed.
- Why *linear* model?
  - Because the response is a (weighted) *linear* combination of the parameters and the random error.
  - The model can depend non-linearly on the predictors.

LM example 1

- Fitting a straight line through the origin. (e.g. simple model relating birth rate, y, and population size, x).
  - Model might be:

$$y_i = x_i\beta + \epsilon_i \qquad \epsilon_i \sim N(0, \sigma^2)$$

  - i.e.



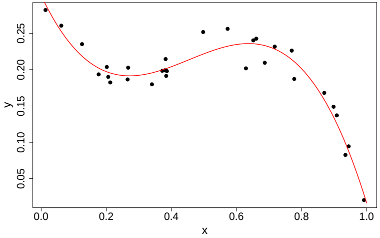
LM examples 2

- Fitting a 'plane' to x, z, y data

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \epsilon_i \quad \epsilon_i \sim N(0, \sigma^2)$$

- Fitting a polynomial to x, y data. e.g. the cubic

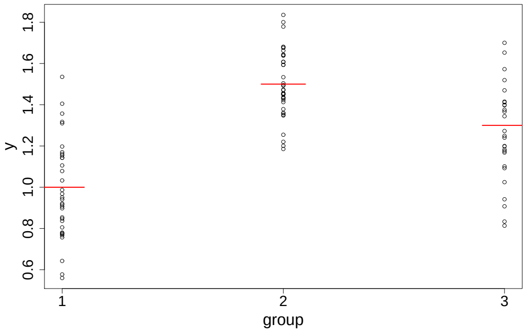
$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \epsilon_i \quad \epsilon_i \sim N(0, \sigma^2)$$



LM example 3

Suppose you have grouped data. A simple model might be something like

$$y_i = \beta_j + \epsilon_i \ \text{ if } y_i \text{ is from group } j \tag{1}$$



LM example 3 continued

- Why is this a linear model? Define *dummy variables*:

$$x_{ij} = \begin{cases} 1 & \text{if } y_i \text{ in group } j \\ 0 & \text{otherwise} \end{cases}$$

  then, $y_i = \beta_j + \epsilon_i$ if y_i is from group j, becomes ...

$$y_i = x_{i1}\beta_1 + x_{i2}\beta_2 + x_{i3}\beta_3 + \epsilon_i$$

- Variables that group data are known as *factors*. The group labels are known as *levels*. Statistical software treats such variables specially and generates corresponding dummy variables automatically.

Matrix vector form 1

- Linear model theory, and the understanding of mixed modelling extensions of linear models, requires that the linear model be written in matrix vector notation.
- To see how this works consider writing out the model,

$$y_i = \beta_1 + x_i\beta_2 + \epsilon_i, \qquad \text{for all } i \quad \ldots$$

$$\begin{array}{rcl} y_1 & = & \beta_1 + x_1\beta_2 + \epsilon_1 \\ y_2 & = & \beta_1 + x_2\beta_2 + \epsilon_2 \\ . & & . \\ . & & . \\ y_n & = & \beta_1 + x_n\beta_2 + \epsilon_n \end{array}$$

Matrix vector form 2

- In matrix vector form this system of equations is

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \cdot & \cdot \\ \cdot & \cdot \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \cdot \\ \cdot \\ \epsilon_n \end{bmatrix}.$$

- Generally this is written:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

where $\mathbf{X}$ is known as the *model matrix*, and $\mathbf{X}\beta$ $(= \boldsymbol{\eta})$ is the *linear predictor*.

Identifiability

- Consider the 'balanced one-way ANOVA model':

$$y_{ij} = \alpha + \beta_i + \epsilon_{ij}$$

  where $i = 1 \ldots 3$ and $j = 1 \ldots 2$.
- In matrix-vector form...

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{21} \\ y_{22} \\ y_{31} \\ y_{32} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} + \begin{bmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{21} \\ \epsilon_{22} \\ \epsilon_{31} \\ \epsilon_{32} \end{bmatrix}$$

- Problem! $\boldsymbol{\beta}^{\mathrm{T}} = (\alpha + k, \beta_1 - k, \beta_2 - k, \beta_3 - k)$ gives the same $\mathbf{X}\boldsymbol{\beta}$, for any k. $\mathbf{X}$ is rank deficient: there is an infinite set of best fit parameter!

Identifiability constraints

- As we have seen, models involving factors can suffer from *identifiability* problems.
- A sure sign of this is that the model matrix, $\mathbf{X}$, is column rank deficient: some of its columns can be made up of linear combinations of the others.
- To deal with this problem, apply just enough linear constraints on the parameters that the problem goes away.
- The simplest constraint is to set just enough parameters to zero that the model becomes identifiable.

Identifiability constraints

- For the 1-way ANOVA model we might set $\beta_1 = 0$, so:

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_2 \\ \beta_3 \end{bmatrix}$$

- The reduced $\mathbf{X}\beta$ can match *any* value of the unreduced version, given the right choice of parameter values.
- Note also that the right hand $\mathbf{X}$ has full column rank.
- Imposition of constraints is automatic in modelling software, but interpretation requires awareness of it, and that there are many alternative constraints possible.

LM theory

- So, for any linear model, we have $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \mathbf{I}\sigma^2)$, and $\mathbf{X}$ is full rank $n \times p$.
- This implies a log likelihood[1]

$$l(\boldsymbol{\beta}, \sigma^2) = -\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2$$

- Hence the maximum likelihood estimates of $\boldsymbol{\beta}$ are

$$\hat{\boldsymbol{\beta}} = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2$$

  i.e. the *least squares estimates* of $\boldsymbol{\beta}$.
- Formally $\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$ (never used for computation!).
- $\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\|^2$ is known as the *residual sum of squares*.

[1] $\|\mathbf{v}\|^2 = \mathbf{v}^{\mathrm{T}}\mathbf{v}$ i.e. the squared Euclidian length of $\mathbf{v}$

LM inference

- Standard likelihood results give $\hat{\beta} \sim N(\beta, (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\sigma^2)$, but this result is exact in this case, not just approximate.
- Similarly the GLRT result is exact. Let $\mathbf{X}_0$ be the $n \times p_0$ null model matrix (nested in $\mathbf{X}$), then if the null model is correct

$$\frac{\|\mathbf{y} - \mathbf{X}_0\hat{\beta}_0\|^2 - \|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2}{\sigma^2} \sim \chi^2_{p-p_0}$$

- ...but unfortunately these general MLE results are only exact if σ^2 is known, which is unusual.
- $\hat{\sigma}^2 = \|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2/(n-p)$ is unbiased (but is not the MLE).
- It turns out that exact results can be obtained even when $\hat{\sigma}^2$ is used in place of σ^2.

LM inference 2

- Suppose that that $\hat{\sigma}^2_{\hat{\beta}_i}$ is the estimated variance of $\hat{\beta}_i$ as read from the i^{th} leading diagonal element of $(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\hat{\sigma}^2$.
- An exact result can be used for inference about β_i

$$\frac{\hat{\beta}_i - \beta_i}{\hat{\sigma}_{\hat{\beta}_i}} \sim t_{n-p}$$

- Similarly, for model comparison, under the null model

$$\frac{(\|\mathbf{y} - \mathbf{X}_0\hat{\beta}_0\|^2 - \|\mathbf{y} - \mathbf{X}\hat{\beta}\|^2)/(p - p_0)}{\hat{\sigma}^2} \sim F_{p-p_0,n-p}$$

is an exact result to use for hypothesis testing.

The Influence Matrix

- Let $\mu_i = E(y_i)$. Clearly $\hat{\boldsymbol{\mu}} = \mathbf{X}\hat{\boldsymbol{\beta}}$, and hence $\hat{\boldsymbol{\mu}} = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$.
- $\mathbf{A} = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}$ is the *influence matrix* or *hat matrix*.
- The leading diagonal elements of $\mathbf{A}$ are a measure of how influential individual data points are in the model fit.
- $\mathbf{A}$ also has some interesting properties
  1. $\mathbf{AA} = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}} = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}} = \mathbf{A}$.
  2. $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}) = \mathrm{tr}((\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{X}) = \mathrm{tr}(\mathbf{I}_p) = p$.
  3. Clearly $\partial\hat{\mu}_i/\partial y_i = A_{ii}$.

LM checking

- The *residuals* are $\hat{\epsilon}_i = y_i - \hat{\mu}_i$.
- If the model fits they should be approximately i.i.d $N(0, \sigma^2)$.
- The exact distribution can be obtained from the fact that $\hat{\boldsymbol{\epsilon}} = (\mathbf{I} - \mathbf{A})\mathbf{y}\ldots$

  $$\hat{\boldsymbol{\epsilon}} \sim N(\mathbf{0}, (\mathbf{I} - \mathbf{A})\sigma^2)$$

  This can be used to standardize the residuals to have exactly constant variance, if the ϵ_i have constant variance.
- Residuals are plotted to check that they
  1. have constant variance, rather than variance varying with μ_i or some predictor.
  2. are independent, rather than varying with μ_i or some predictor, or being serially correlated w.r.t to some predictor.
  3. are approximately normally distributed.

Stable $\hat{\beta}$ computation

- Can QR decompose $\mathbf{X}$

$$\mathbf{X} = \mathbf{Q}\begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1\mathbf{R}$$

- $\mathbf{Q}$ is $\perp$. $\mathbf{Q}_1$ is its first p columns. $\mathbf{R}$ is $p \times p$ upper triangular.
- Hence for any vector, $\mathbf{v}$, $\|\mathbf{Q}\mathbf{v}\|^2 = \|\mathbf{v}\|^2$, so

$$\begin{aligned} \|\mathbf{y} - \mathbf{X}\beta\|^2 &= \|\mathbf{Q}^{\mathrm{T}}\mathbf{y} - \mathbf{Q}^{\mathrm{T}}\mathbf{X}\beta\|^2 = \left\|\mathbf{Q}^{\mathrm{T}}\mathbf{y} - \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}\beta\right\|^2 \\ &= \|\mathbf{Q}_1^{\mathrm{T}}\mathbf{y} - \mathbf{R}\beta\|^2 + \|\mathbf{Q}_2^{\mathrm{T}}\mathbf{y}\|^2 \end{aligned}$$

- Since $\|\mathbf{Q}_2^{\mathrm{T}}\mathbf{y}\|^2$ does not depend on β then

$$\hat{\beta} = \mathbf{R}^{-1}\mathbf{Q}_1^{\mathrm{T}}\mathbf{y}$$

Linear models in R

- R has extensive facilities for linear modelling.
- The main linear model fitting function is `lm`.
- The basic approach is:
  1. The model structure is specified using a *model formula*, supplied to `lm`.
  2. `lm` fits the model, dealing with identifiability constraints, model matrix construction and fitting internally, and returns a *fitted model object*.
  3. The fitted model object is interrogated using *methods functions* to e.g. extract model summaries, perform F-ratio testing, produce residual plots, extract estimates etc.
- This basic approach is the same for linear models, generalized linear models, generalized linear mixed models, generalized additive models, etc.

Model matrices in R

- In R a model matrix, **X**, is usually set up automatically, using a *model formula*. Usually this is done 'behind the scenes' when a modelling function is used, but for now we'll look at the process explicitly.
- As an example consider data frame `hubble` in the library `gamair`. This contains Velocities, `y`, and Distances, `x` of 24 galaxies (relative to us).
- We might try modelling these data with a straight line $\mathrm{y}_i = \beta_0 + \beta_1 \mathrm{x}_i + \epsilon_i$. The model formula $\mathrm{y} \sim \mathrm{x}$ would set this up. The variable to the left of $\sim$ specifies the *response variable*, whereas everything to the right of $\sim$ specifies the *linear predictor/model matrix*.
- Let's try it...

model.matrix

- ```
  library(gamair);data(hubble)
  model.matrix(y~x,data=hubble)
      (Intercept)     x
  1             1  2.00
  2             1  9.16
  3             1 16.14
  .             .    .
  .             .    .
  ```
- `model.matrix` actually ignores the response in the formula. Note that the `data` argument tells it where to find the variables referred to in the formula.
- By default a constant is included in the linear predictor, unless a `-1` is added to the formula. suppose that we want a quadratic model and no constant term...

  ```
  model.matrix(y~x+I(x^2)-1,data=hubble)
  ```
  ```

More `model.matrix`

- `PlantGrowth` contains data on plant weight under 2 growth treatments and a control. A possible model…

$$w_i = \alpha + \beta_j \text{ if plant } i \text{ is from group } j$$

- `model.matrix(weight~group,data=PlantGrowth)`

```
   (Intercept) grouptrt1 grouptrt2
1            1         0         0
2            1         0         0
.            .         .         .
10           1         0         0
11           1         1         0
12           1         1         0
.            .         .         .
```

- `model.matrix` treated `group` as a factor variable and has automatically imposed identifiability constraints.

Factor variables in R

- How did `model.matrix` 'know' how to treat `group`?
- Because the variable `group` has been assigned a *class* `factor`. This means that each unique value of `group` is treated as the label identifying a group (i.e. as the level of a factor).
- Type `PlantGrowth$group` and notice how the levels of `group` are printed last.
- To declare a variable to be a factor one uses something like:

```
x <- c(1,1,1,"a","a",1,"c","c","a")
x <- factor(x)
```

Model formulae in general

Consider `y ~ a*b + x:z + I(v^2) -1`

- `+` means **and**. i.e. `c+d` means that the linear predictor depends on `c` and `d`.
- `x:z` mean the **interaction** of `x` and `z`.
- `a*b` is short for `a + b + a:b`.
- `I(v^2)` means that the linear predictor depends on v^2. The identity function `I()` simply returns its evaluated argument, thereby returning the usual meaning to arithmetic operations within the formula.
- `-1` means that the linear predictor has **no constant**.

`lm` in R

- Within R, linear models are fitted using `lm()`.
  - The model to fit is specified using a 'model formula'.
  - The data to fit are best supplied in a 'data frame'.
  - The function returns a 'fitted model object'.
- For example, the model

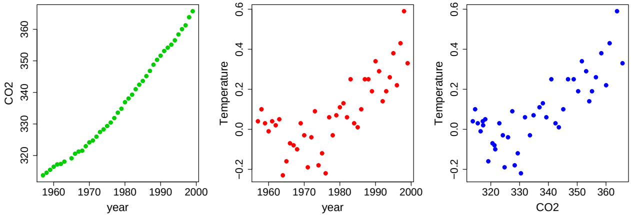
$$y_i = \beta_0 + x_i\beta_1 + z_i\beta_2 + \epsilon_i$$

  would be estimated with a command like

```
mod.1 <- lm(y ~ x + z , dat)
```

  - `y ~ x + z` is the model formula.
  - `dat` is a 'data frame' containing the variables referred to in the formula.
  - The object returned by `lm` has been assigned to an object, `mod.1`.

Example CO_2 and Global temperature



- CO_2 is p.p.m. measured at Siple station Antarctica.
- Temperatures are mean global anomalies (from 1961-1990 mean).
- Try $\texttt{temp}_i = \beta_0 + \beta_1 \texttt{CO2}_i + \epsilon_i$.

CO_2 continued

- If data are in data frame `gw` then fit as follows.

```
> gw.mod1<-lm(temp~co2,data=gw)
> gw.mod1

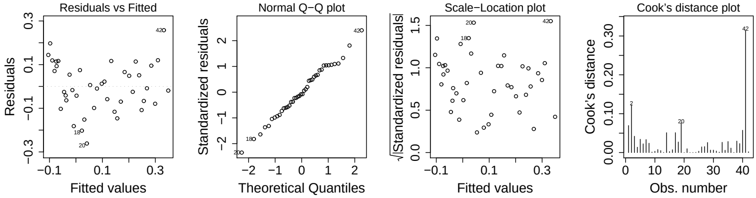
Call:
lm(formula = temp ~ co2, data = gw)

Coefficients:
(Intercept)          co2
   -2.83996      0.00872
```

- Suggests an increase of 0.0087 C for each extra p.p.m. CO_2, but we need to check model assumptions…

Model checking with `plot(gw.mod1)`

- Some default residual plots are produced by `plot(gw.mod1)`.



- There is a trend in the mean of the residuals, violating **independence**.
- The QQ plot is close to a straight line, so **normality** is OK.
- The residual magnitudes seem consistent with **constant variance**.
- The 42nd observation has a very high influence on the results.

Revising the CO_2 model

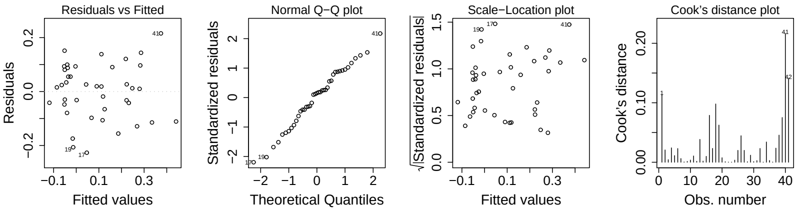
- Naively, we might add a CO_2^2 term to the model, but this is not very physical. A better model would recognize inter year correlation in mean temperature. e.g. assuming data are in time order,

$$\texttt{temp}_i = \beta_0 + \beta_1 \texttt{CO2}_i + \beta_2 \texttt{temp}_{i-1} + \epsilon_i.$$

- Note that we are **not** assuming that the the ϵ_i are measurement errors: rather they represent 'unexplained variability in the mean temperature'.

Fit the revised model

```
n <- nrow(gw)
gw.mod2<-lm(temp[2:n]~co2[2:n]+temp[1:(n-1)],data=gw)
plot(gw.mod2)
```



…this is much better. All assumptions look OK now.

Hypothesis testing

- Is there formal evidence that the revised model is better than the initial model?
- Can test this by using the `anova` method for `lm` models to perform an F-ratio test.

```
> gw.mod0<-lm(temp[2:n]~co2[2:n],data=gw) # must fit same data!
> anova(gw.mod0,gw.mod2)
Analysis of Variance Table

Model 1: temp[2:n] ~ co2[2:n]
Model 2: temp[2:n] ~ co2[2:n] + temp[1:(n - 1)]
  Res.Df     RSS Df Sum of Sq      F  Pr(>F)
1     39 0.48759
2     38 0.42501  1   0.06258 5.5957 0.02321 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Final CO_2 model

- So we reject the null hypothesis that the simple model is correct.
- Now examine the fitted full model

```
> summary(gw.mod2)
...
Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1.919990   0.568855  -3.375  0.00171 **
co2[2:n]           0.005896   0.001715   3.437  0.00144 **
temp[1:(n - 1)]    0.347253   0.146798   2.366  0.02321 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1058 on 38 degrees of freedom
Multiple R-Squared: 0.6694,     Adjusted R-squared: 0.652
F-statistic: 38.47 on 2 and 38 DF,  p-value: 7.37e-10
```

CO_2 follow up

- We would probably go on to obtain confidence intervals for parameters. e.g. for β_1 the 'CO_2 effect'

```
> b1 <- .005896; cb <- qt(.975,df=38)*.001715
> c(b1-cb,b1+cb)
[1] 0.002424164 0.009367836
```

- i.e. each extra p.p.m. CO_2 seems to be associated with a global mean temperature rise of between .0024 and .0094 Celsius.
- Note the importance of checking the model assumptions: failing to do this can lead to the use of inadequate models and lead to completely invalid conclusions.

Summary

- Linear models can all be written $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim N(\mathbf{0}, \mathbf{I}\sigma^2)$
- The parameters $\boldsymbol{\beta}$ are estimated by minimizing $\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2$ w.r.t. $\boldsymbol{\beta}$.
- The formal expression for the estimates is $\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$.
- $\hat{\sigma}^2 = \|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\|^2/(n - \dim(\boldsymbol{\beta}))$
- $\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\sigma^2)$.
- Model comparison/ hypothesis testing is done using F-ratio tests.
- Models must be checked by careful examination of the residuals $\hat{\boldsymbol{\epsilon}} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}$.