1

Linear Models & Smooth Regression

- Linear models
- Diagnostics
- Robust regression
- Bootstrapping linear models
- Scatterplot smoothing
- Spline regression
- Non-linear regression

2

Linear Models

- *Linear* in parameters
  - not necessarily fitting a straight line
- General **R** statement is `lm(formula)`
  - **`lm`** for **l**inear **m**odel
- `formula`:

Dependent variable ~ linear function of independent variables

- `~` means here "is related to"

3

- e.g. `lm(time~dist)` fits model

$time_i = \alpha + \beta\ dist_i + error_i$

and produces estimates of α and β

- e.g. `lm(time~dist + climb)` fits model

$time_i = \alpha + \beta_1\ dist_i + \beta_2\ climb_i + error_i$

- `lm()` produces an *object* which can by examined in usual way with `summary()` and other special commands

4

e.g. `hillslm<-lm(time~dist)`

- produces object `hillslm`
- `hillslm$coefficients` gives vector of coefficients
- `hillslm$fitted` gives fitted values $\hat{\alpha} + \hat{\beta}\,dist_i$
- `hillslm$residuals` gives vector of residuals
  - i.e. estimates of the errors in model $time_i = \alpha + \beta\ dist_i + errror_i$
  - residual = $time_i - fitted_i$

5

Regression Diagnostics

- analysis of the residuals can indicate whether model is satisfactory:
- if model is appropriate for the data then
  - Residuals should look like a random sample from a Normal distribution
  - Residuals should be independent of fitted values
- easy to check these graphically
  - `plot.lm(hillslm)` will give basic checks

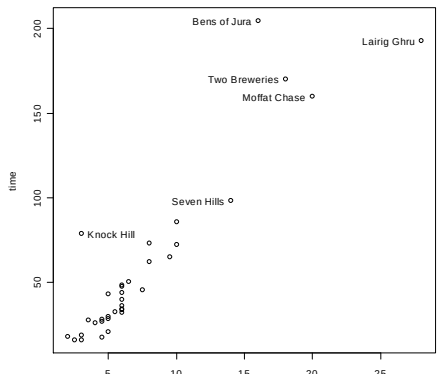
6

Example: Scottish Hill Races

- in data set hills in the MASS library

```
> library(MASS)
> data(hills)
> attach(hills)
> plot(dist,time)
> identify(dist,time,row.names(hills))
 [1]  7 11 17 18 33 35
```

- Note use of `identify()` which allows interactive identification of points
  - good to identify obvious outliers



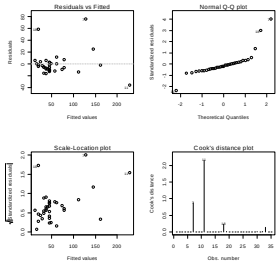
- Now fit model and look at results

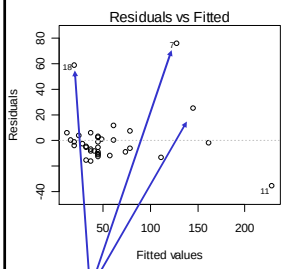
```
> hillslm<- lm(time~dist)
> summary(hillslm)
Call:
lm(formula = time ~ dist)
Residuals:
    Min      1Q  Median      3Q     Max
-35.745  -9.037  -4.201   2.849  76.170
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.8407     5.7562  -0.841    0.406
dist          8.3305     0.6196  13.446    6e-15 ***
```

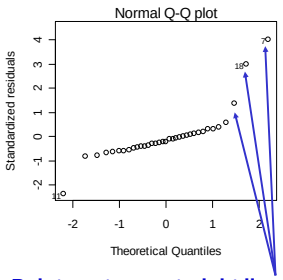
  - Note five summary statistics of residuals and estimates of coefficients with st. errs.
    - Also a lot more detail — more than usually needed on any single occasion

- Look at basic diagnostics with

```
> plot.lm(hillslm)
```







Outliers and so plot does not look random

Points not on a straight line so deviation from Normality

(Other two plots of less interest here)

- Next Steps:
  - Remove outliers from data
    - `> lm(time~dist,data=hills[-7,])`
    - Removes 7th observation
    - `> lm(time~dist,data=hills[-c(7,18),])`
    - Removes both 7th and 18th
  - Use a ***robust method*** for estimating linear relationship which is not so greatly affect by outliers

- Robust Regression
  - Available routines:
    - `rlm()` in MASS library and `lqs()` in `lqs` library
  - Example on hills data:

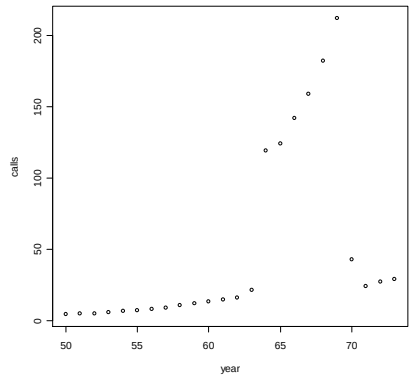
```
hillslm1<-lm(time~dist,
             data=hills[-c(7,18),])
```

    - Gives intercept and slope as –5.81 and 7.91

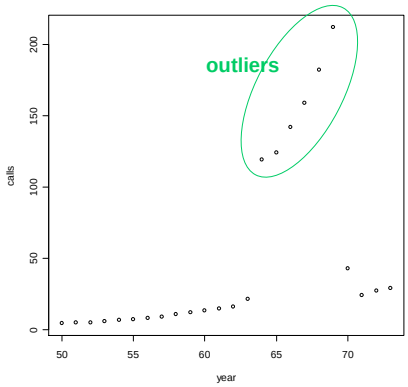
```
hillsrlm<-rlm(time~dist)
```

    - Gives intercept and slope as –6.36 and 8.051 (and with smaller standard errors of estimates)

13

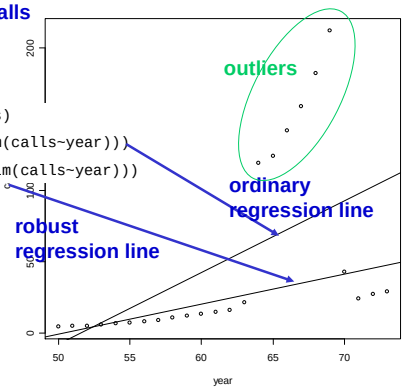


14



Data: phone calls in Belgium

```
> plot(year,calls)
> lines(abline(lm(calls~year)))
> lines(abline(rlm(calls~year)))
```



16

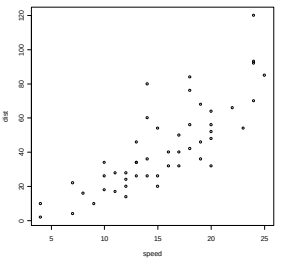
- **Bootstrapping linear models**
  - May not want to assume a Normal error structure and if so then we may want some bootstrap estimate of (say) a confidence interval for the slope
  - Not appropriate to sample points $\{(x_i,y_i);i=1,\ldots n\}$
    - Since usually the x-values are *fixed*
  - Instead have to bootstrap residuals

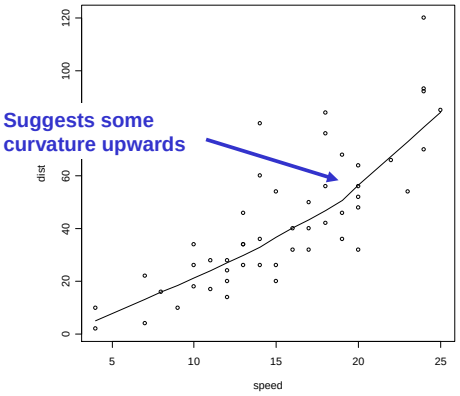
17

- Steps:-
  - Fit regression (with `lm()` or `rlm()` or …)
  - Extract residuals with `obj$residuals`
  - Extract coefficients with `obj$coefficients`
  - Take bootstrap samples of residuals and construct new bootstrap data sets with
    - actual x-values +
    - estimated coefficients +
    - bootstrapped residuals
  - Calculate quantity of interest

18

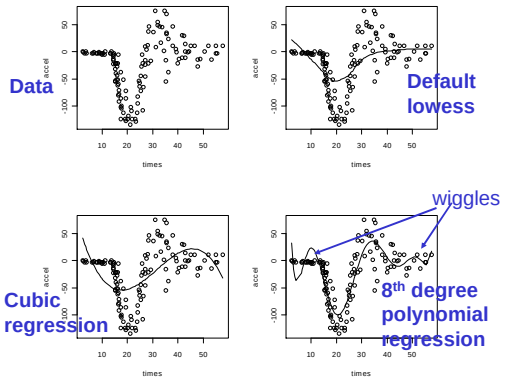
- **Scatterplot smoothing**
  - tool for informal investigation of structure in scatterplot
  - can see increase in distance with speed but is this constant or does it tail upwards?
    - use lowess

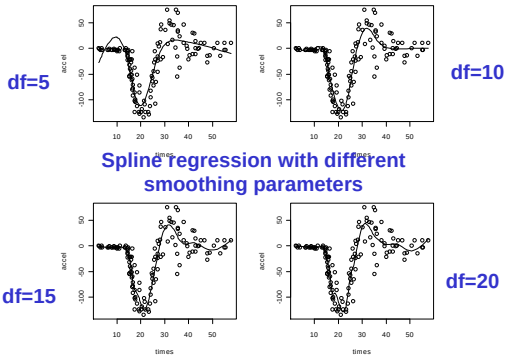




Spline regression

- Polynomial regression often not satisfactory since behaviour of a polynomial depends upon values over its entire range
- Instead, ***spline*** regression uses 'local piecewise polynomials' which adapt to local values better and do not influence distant ones





Non-linear regression

- May be external reasons for specifying a particular *non-linear* model
- e.g of form

$$y = \alpha + \beta x^{-2/\theta}$$

- Can be estimated using routine `nls()` in library `nls` (non-linear `least squares`)
  - Need to specify *starting values*

Summary

- Seen how regression diagnostics leads to dropping outliers or use of ***robust regression*** methods
- Ideas of bootstrapping linear models
- ***Scatterplot smoothing*** useful informal tool
  - Use of `lowess()`
- Smooth regression with **splines**
- Availability of non-linear regression