Aims of the next four lectures

- To give you an overview of the **modelling cycle**.
- To have a detailed discussion of **diagnostic procedures**.
Residual analysis

- Another approach is to fit the model and examine the residuals.
- If the model is appropriate the residuals have no pattern
- A pattern in the residuals usually indicates that the model is not appropriate
- If this is the case we have two options
  1. Select another form of model e.g. non-linear regression – see other courses;
  2. Transform the data so that the regression model fits the transformed data – see next slide.

What constitutes a bad fit?

- Non-planar data: Seriously affects meaning and accuracy of estimated coefficients
- Outliers in the data: Seriously affects meaning and accuracy of estimated coefficients
- Non-constant scatter: Affects standard error of estimate
- Errors not independent: Affects standard error of estimate
- Errors not normally distributed: Affects standard error of estimate

Detecting non-planar data

- We can diagnose non-planar data (non-linearity) by fitting the model, and
  - plotting residuals versus fitted values;
  - residuals against explanatory variables;
  - fitting additive models
- In each case, a curved plot indicates non-planar data.

Plotting residuals vs. fitted values

```r
> data(cherry.df)
> cherry.lm <- lm(volume~diameter+height,data=cherry.df)
> plot(cherry.lm,which=1)
```

which=1: selects the plot of residuals vs. fitted values
Additive models

- These are models of the form
  \[ Y = g_1(x_1) + g_2(x_2) + \cdots + g_k(x_k) + \varepsilon \]
  where \( g_1, \ldots, g_k \) are transformations.

- Fitted using the `gam` function in R.
- The transformations are estimated by the software.
- Use the function to suggest good transformations.

Example: Cherry trees

```r
> library(mgcv)
> cherry.gam <- gam(volume ~ s(diameter) + s(height),
                      data=cherry.df)
> plot(cherry.gam, residuals=T, pages=1)
```

Fitting polynomials

- To fit a model \( y = \beta_0 + \beta_1 x + \beta_2 x^2 \), use
  \( y \sim \text{poly}(x, 2) \)

- To fit a model \( y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 \), use
  \( y \sim \text{poly}(x, 3) \)

  etc.

Orthogonal polynomials

- The model fitted by \( y \sim \text{poly}(x, 2) \) is of the form
  \[ Y = \beta_0 + \beta_1 p_1(x) + \beta_2 p_2(x) \]
  where
  - \( p_1 \): polynomial of degree 1, i.e. of the form \( a_1 + a_2 x \)
  - \( p_2 \): polynomial of degree 2, i.e. of the form \( b_1 + b_2 x + b_3 x^2 \).
- \( p_1, p_2 \) chosen to be uncorrelated (best possible estimation)
Adding a quadratic term: Cherry trees

Call:
  lm(formula = volume ~ poly(diameter, 2) + height, 
     data = cherry.df)
---
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.56553  6.72218 0.233 0.817603
poly(diameter, 2)1 80.25223  3.07346 26.111 < 2e-16 ***
poly(diameter, 2)2 15.39923  2.63157  5.852  3.13e-06 ***
height        0.37639   0.08823  4.266  0.000218 ***
---
Residual standard error: 2.625 on 27 degrees of freedom
Multiple R-squared:  0.9771, Adjusted R-squared:  0.9745
F-statistic: 383.2 on 3 and 27 DF,  p-value: < 2.2e-16

Quadratic equation

Call:
  lm(formula = volume ~ diameter + I(diameter^2) + height, 
     data = cherry.df)
---
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -9.92041  10.07911  -0.984 0.333729
  diameter   -2.88508   1.30985  -2.203 0.036343 *
I(diameter^2)  0.26862   0.04590   5.852  3.13e-06 ***
height        0.37639   0.08823   4.266  0.000218 ***
---
Residual standard error: 2.625 on 27 degrees of freedom
Multiple R-squared:  0.9771, Adjusted R-squared:  0.9745
F-statistic: 383.2 on 3 and 27 DF,  p-value: < 2.2e-16

Quadratic equation

volume = -9.92 - 2.89 × diameter + 0.27 × diameter² + 0.38 × height

Splines

- An alternative to polynomials are splines – these are piecewise cubics, which join smoothly at “knots”.
- Give a more flexible fit to the data.
- Values at one point not affected by values at distant points, unlike polynomials

Cherry splines

Call:
  lm(formula = volume ~ bs(diameter, knots = knot.points) + height, 
     data = cherry.df)
---
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -16.3679   7.4856  -2.187 0.03921 *
  bs(diameter, knots = knot.points)1  0.1941  7.9374   0.024 0.98070
  bs(diameter, knots = knot.points)2  5.5744  3.1704   1.758 0.09201 .
  bs(diameter, knots = knot.points)3 10.7976  3.9798   2.713 0.01240 *
  bs(diameter, knots = knot.points)4 31.4053  5.5545   5.654 9.35e-06 ***
  bs(diameter, knots = knot.points)5 42.2665  6.1297   6.895 4.97e-07 ***
  bs(diameter, knots = knot.points)6 58.6454  4.2781  13.708 1.49e-12 ***
height       0.3970   0.1050   3.780 0.00097 ***
---
Residual standard error: 2.8 on 23 degrees of freedom
Multiple R-squared:  0.9778, Adjusted R-squared:  0.9715
F-statistic: 144.4 on 7 and 23 DF,  p-value: < 2.2e-16

Example with 4 knots
Example: Tyre abrasion data

Data collected in an experiment to study the abrasion resistance of tyres.

Variables are:
- Hardness: Hardness of rubber
- Tensile: Tensile strength of rubber
- Abrasion Loss: Amount of rubber worn away in a standard test (response)

Call:
```r
lm(formula = abloss ~ hardness + tensile, data = rubber.df)
```

Coefficients:
```
            Estimate Std. Error t value Pr(>|t|)  
(Intercept) 885.1611    61.7516 14.334 3.84e-14 ***  
hardness    -6.5708     0.5832 -11.267 1.03e-11 ***  
tensile      -1.3743     0.1943  -7.073 1.32e-07 ***  
```

Residual standard error: 36.49 on 27 degrees of freedom.
Multiple R-squared:  0.8402, Adjusted R-squared:  0.8284
F-statistic: 71.07 on 2 and 27 DF,  p-value: 2.163e-11
Tyre abrasion data

- We will use this example to illustrate all the methods we have discussed so far to check if the data are planar, scattered about a flat regression plane i.e.
  - Pairs plot
  - Spinning plot
  - Coplot
  - Residual vs. fitted value plot
  - Fitting GAMs

Spinning – Hint of a “kink”

Coplot – Suggestion of non-linearity

Residuals vs. fitted values – weak suggestion of non-planarity

GAMs – Quite strong indication of non-planarity

hardness looks okay, but tensile needs transformation.

\[ \text{lm(abloss ~ hardness + tensile)} \]
Fitting a fourth degree polynomials

```r
> rubber.poly <- lm(abloss~hardness+tensile+I(tensile^2)+I(tensile^3)+I(tensile^4),data=rubber.df)
> summary(rubber.poly)
```

```
Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.862e+04 4.177e+03 -4.458 0.000165 ***
hardness -6.261e+00 4.124e-01 -15.182 8.35e-14 ***
tensile 4.414e+02 9.836e+01 4.487 0.000153 ***
I(tensile^2) -3.693e+00 8.546e-01 -4.321 0.000233 ***
I(tensile^3) 1.342e-02 3.246e-03 4.133 0.000377 ***
I(tensile^4) -1.794e-05 4.553e-06 -3.940 0.000613 ***
---
Residual standard error: 23.25 on 24 degrees of freedom
Multiple R-squared: 0.9423, Adjusted R-squared: 0.9303
F-statistic: 78.46 on 5 and 24 DF, p-value: 4.504e-14
```

Fitting splines

```r
> rubber.bs <- lm(abloss~hardness+bs(tensile,df=4),
  data=rubber.df)
> summary(rubber.bs)
```

```
Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept)  612.1556  43.0348 14.225 3.43e-13 ***
hardness   -6.1914   0.4139 -14.959 1.15e-13 ***
bs(tensile, df = 4)1 195.5549  40.6339  4.813 6.69e-05 ***
bs(tensile, df = 4)2 -148.3497  38.6717 -3.836 0.000796 ***
bs(tensile, df = 4)3  -24.2971  37.7010 -0.644 0.525385
bs(tensile, df = 4)4 -61.0593  25.4829  -2.396 0.024720 *
---
Residual standard error: 23.36 on 24 degrees of freedom
Multiple R-squared: 0.9418, Adjusted R-squared: 0.9297
F-statistic: 77.7 on 5 and 24 DF, p-value: 5.021e-14
```

Diagnostic steps

We test for using `plot`:
- **Planarity**: Residuals vs. fitted values, Residuals vs. covariates, added variable plots, GAM plots
- **Constant Variance**: ...
- **Outliers**: ...
- **Independence**: ...
- **Normality of Errors**: ...

---

[Image with a comic strip and a note:](http://xkcd.com/1252/)