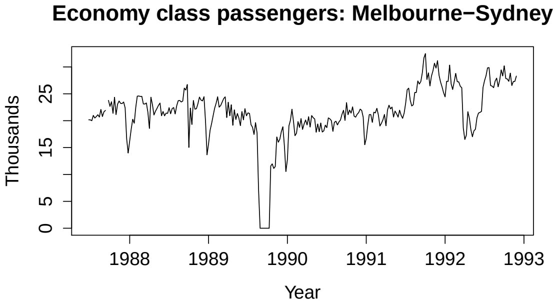
Basic Analysis Tools

Xiaodong Lin

Time plots

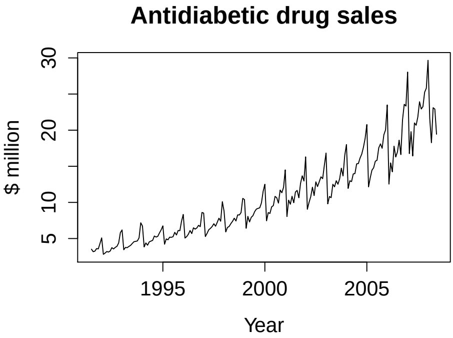
- Time plots: the observations are plotted against the time of observation, with consecutive observations joined by straight lines



- Any interesting features?

Time plots

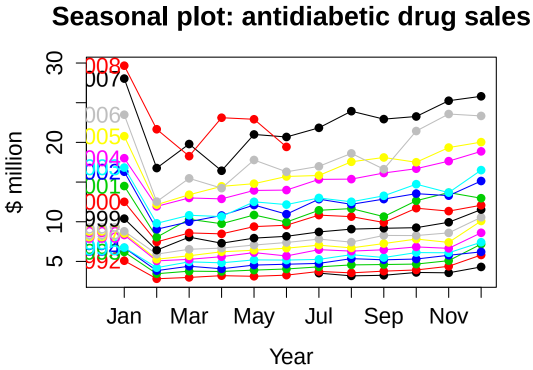
- Time plot with seasonality and trend.



- Trend, seasonal pattern and cycle?

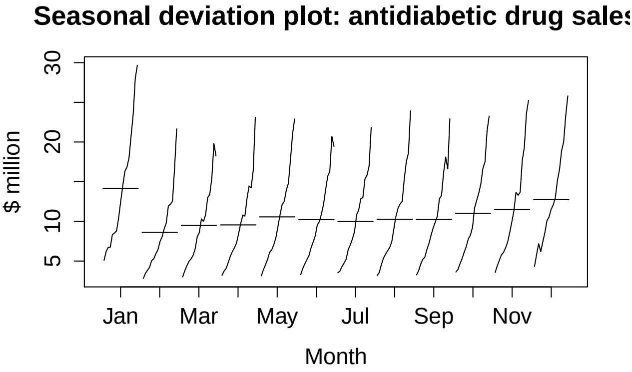
Seasonal plot.

- These are exactly the same data shown earlier, but now the data from each season are overlapped.



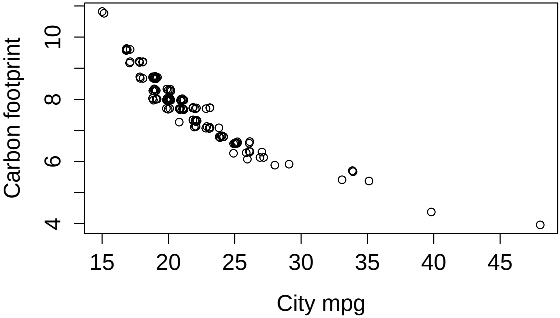
Seasonal subseries plots

- Horizontal line indicates the mean for the mean.



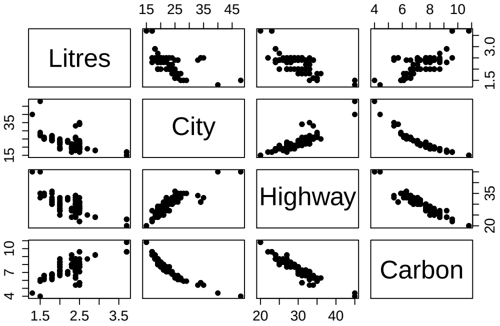
Scatter plots

- Exploits the relationship between variables.



Scatter plot matrix

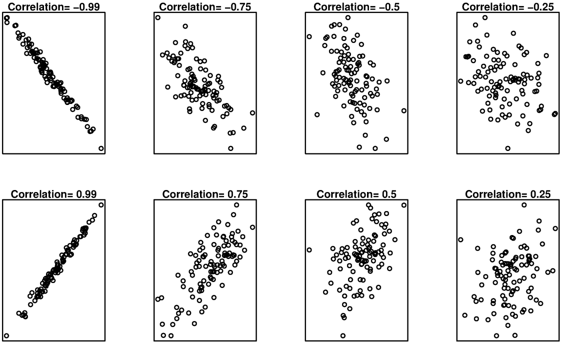
- When there are several potential predictor variables, it is useful to plot each variable against each other variable.



R codes

```
plot(melsyd[,"Economy.Class"],
            main="Economy class passengers: Melbourne-Sydney",
            xlab="Year",ylab="Thousands")

plot(a10, ylab="$ million", xlab="Year",
            main="Antidiabetic drug sales")

seasonplot(a10,ylab="$ million", xlab="Year",
main="Seasonal plot: antidiabetic drug sales",
year.labels.left=TRUE, col=1:20, pch=19)

monthplot(a10,ylab="$ million",xlab="Month",xaxt="n",
main="Seasonal deviation plot: antidiabetic drug sales")
axis(1,at=1:12,labels=month.abb,cex=0.8)

plot(jitter(fuel[,5]), jitter(fuel[,8]),
```

- Univariate: mean, median, Quantile, standard deviation ...
- Bivariate: correlation coefficient, covariance ...



- Correlation demonstrates linear relationships between two variables.
- It does not capture nonlinear relationships, thus exploratory graphes are very important.

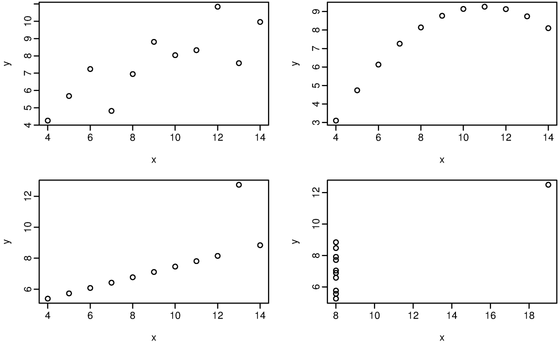
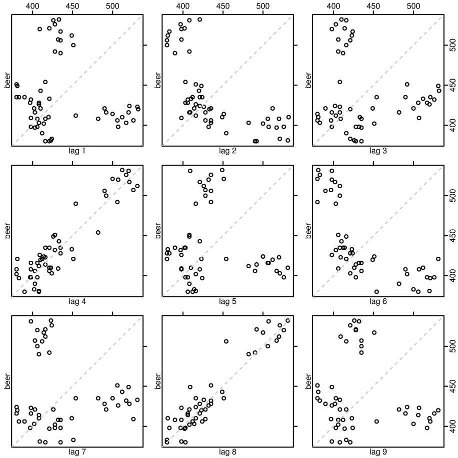


Figure: All the four data sets have the same correlations.

Autocorrelation

- Covariance and correlation measure extent of linear relationship between two variables.
- Autocorrelation demonstrates linear relationships between lagged variables.
- We measure the relationship between: y_t and y_{t-1}, y_t and y_{t-2}, y_t and y_{t-3} etc.

Beer production example. Lagged plot

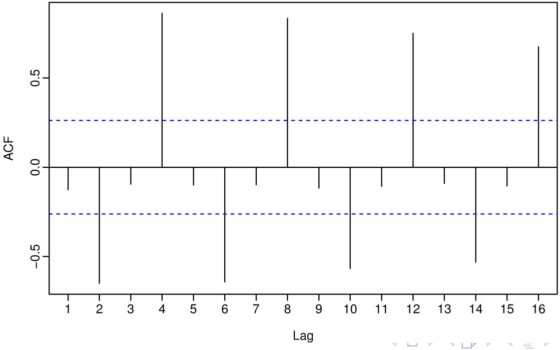


Autocorrelation function

- The bar in the ACF indicates the autocorrelation, with values between -1 and 1.
- The first bar indicates how successive values of y relate to each other.
- The second bar indicates how y values two periods apart relate to each other.
- the kth bar is almost the same as the sample correlation between y_t and y_{t-k}.

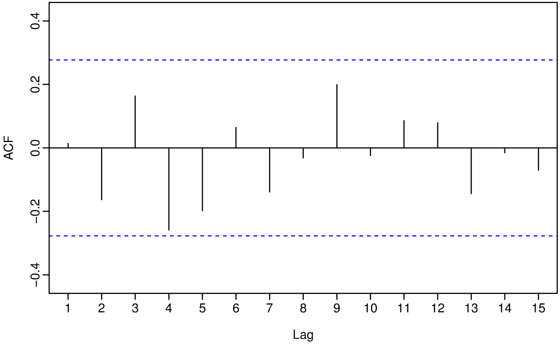
Autocorrelation function

- Autocorrelation is defined over different lags, we can plot out the ACF.
- The plot shows that significant autocorrelation exists in this data.



Autocorrelation function

- There is no significant autocorrelation in white noise data.



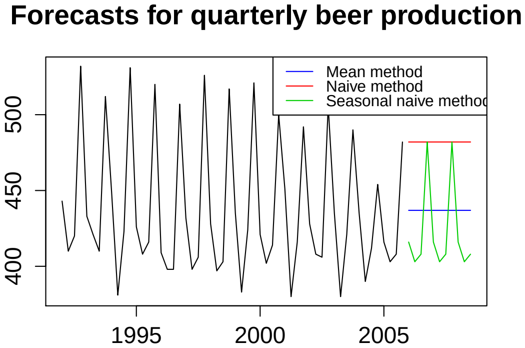
R codes

```
fuel2 <- fuel[fuel$Litres<2,]
summary(fuel2[,"Carbon"])
sd(fuel2[,"Carbon"])

beer2 <- window(ausbeer, start=1992, end=2006-.1)
lag.plot(beer2, lags=9, do.lines=FALSE)

Acf(beer2)

set.seed(30)
x <- ts(rnorm(50))
plot(x, main="White noise")

Acf(x)
```

- Average method: Forecasts of all future values equal the mean of the historical data.
$$\hat{y}_{T+h} = \bar{y} = \sum_{i=1}^{T} y_i / T.$$
meanf(x, h=20)
- Naive method: Forecast= the value of the last observation.
naive(x, h=20) or rwf(x, h=20)
- Seasonal naive method: Forecast = value of the observation at the same period last season.
snaive(x, h=20)
- Drift method: adding amount of changes over time.
$$\hat{y}_{T+h} = y_T + h(\frac{y_t - y_1}{T-1}).$$
rwf(x, drift=TRUE, h=20)

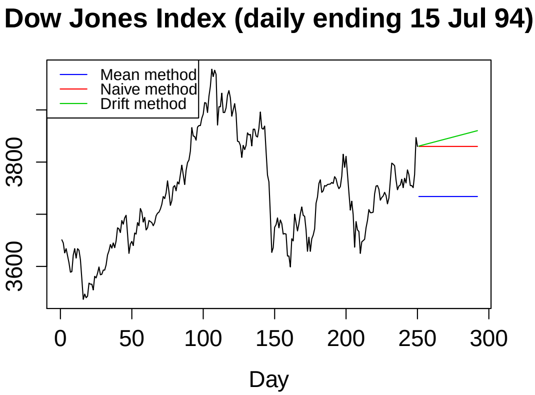
Forecast plots

- Quarterly beer forecast



Forecast plots

- Forecast DJ index



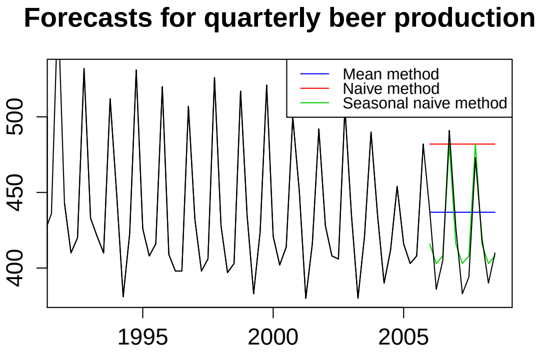
Measures of forecasting accuracy

Let y_t be the observation at time t and $\hat{y}_t$ be the forecast at time t. Denote the forecast error as $\epsilon_t = y_t - \hat{y}_t$.

- Mean absolute error: $MAE = mean(|\epsilon_t|)$.
- Root mean square error: $RMSE = \sqrt{mean(\epsilon_t^2)}$.
- Mean absolute percentage error: $MAPE = mean(|p_t|)$ where $p_t = 100\epsilon_t / y_t$.
- MAE, MSE, RMSE are all scale dependent and MAPE is scale independent.

Forecast plots

- Quarterly beer forecast with overlay.



Prediction comparison

```
accuracy(beerfit1, beer3)
                        ME     RMSE      MAE
Training set  8.121418e-15 44.17630 35.91135
Test set     -1.718344e+01 38.01454 33.77760
                    MPE     MAPE     MASE
Training set -0.9510944 7.995509 2.444228
Test set     -4.7345524 8.169955 2.298999
                    ACF1 Theil's U
Training set -0.12566970        NA
Test set     -0.08286364 0.7901651
accuracy(beerfit1)
                       ME    RMSE      MAE
Training set 8.121418e-15 44.1763 35.91135
                    MPE     MAPE     MASE       ACF1
Training set -0.9510944 7.995509 2.444228 -0.1256697
```

Prediction comparison

- In sample accuracy: testing and training use the same data. A perfect fit can always be achieved. Such a model usually lead to overfitting

- Problems can be overcome by measuring true out-of-sample forecast accuracy. That is, total data divided into training set and test set. Training set used to estimate parameters.

  The test set is not be used for any aspect of model development or calculation of forecasts.

  Forecast accuracy is based only on the test set.

- Rolling forecast.

In and out sample testing error

```
accuracy(beerfit1, beer3)
     RMSE          MAE          MPE         MAPE
 38.01454162  33.77759740  -4.73455240   8.16995482
        MASE         ACF1     Theil's U
  0.60930399  -0.08286364   0.79016506
> accuracy(beerfit2, beer3)
     RMSE          MAE          MPE         MAPE
-  70.90646848  63.90909091 -15.54318218  15.87645380
        MASE         ACF1     Theil's U
  1.15283700  -0.08286364   1.42852395
> accuracy(beerfit3, beer3)
      RMSE         MAE         MPE        MAPE        MASE
12.9684933 11.2727273 -0.7530978  2.7298475  0.2033454
      ACF1   Theil's U
-0.1786912  0.2257300
```

R code

```
beer2 <- window(ausbeer,start=1992,end=2006-.1)
beerfit1 <- meanf(beer2, h=11)
beerfit2 <- naive(beer2, h=11)
beerfit3 <- snaive(beer2, h=11)

plot(beerfit1, plot.conf=FALSE,
     main="Forecasts for quarterly beer production")
lines(beerfit2$mean,col=2)
lines(beerfit3$mean,col=3)
lines(ausbeer)
legend("topright",lty=1,col=c(4,2,3),cex=0.7,
legend=c("Mean method","Naive method","Seasonal naive method"))
```

R code

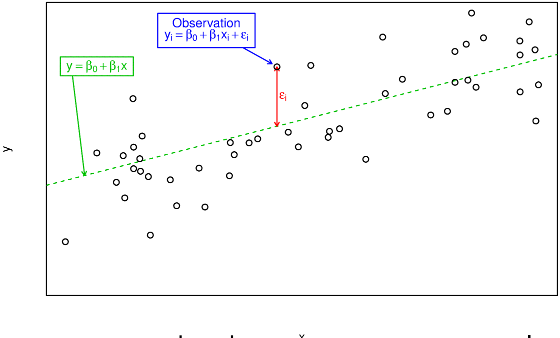
```
dj2 <- window(dj,end=250)
plot(dj2,main="Dow Jones Index (daily ending 15 Jul 94)",
     ylab="",xlab="Day",xlim=c(2,290))
lines(meanf(dj2,h=42)$mean,col=4)
lines(rwf(dj2,h=42)$mean,col=2)
lines(rwf(dj2,drift=TRUE,h=42)$mean,col=3)
legend("topleft",lty=1,col=c(4,2,3),cex=0.7,
       legend=c("Mean method","Naive method","Drift method"))

beer3 <- window(ausbeer, start=2006)
accuracy(beerfit1, beer3)
accuracy(beerfit2, beer3)
accuracy(beerfit3, beer3)
```

Simple regression model

- Consider the following simple regression model

$$y = \beta_0 + \beta_1 x + \epsilon$$

where x is the predictor variable and y is the response variable.



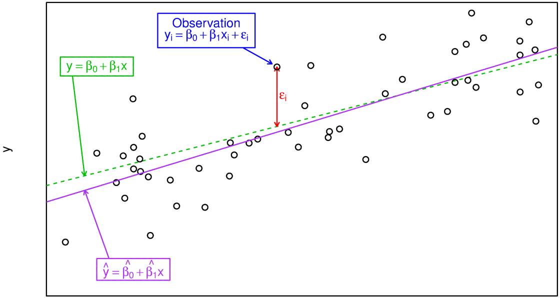
- The errors are assumed to have mean zero, uncorrelated and uncorrelated to the predictor variable.

Least square estimate

- How do we define the regression line? What is "best"? Minimize the sum of the squared errors

$$\sum_{i=1}^{N}(y_i - \beta_0 - \beta_1 x_i)^2$$

.



Estimates and residuals

- The true line
$$y = \beta_0 + \beta_1 x.$$
- The fitted line
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$
Thus for each individual x_i, we obtain the estimate $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$ for $i = 1, \cdots N$.
- Residual is defined as $e_i = y_i - \hat{y}_i = y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i$. It is used to estimate the unknown ϵ.
- The residuals are centered around 0, and the correlation with the observations is 0
$$\sum_{i=1}^{N} e_i = 0 \text{ and } \sum_{i=1}^{N} x_i e_i = 0.$$

Correlation coefficients and regression

- Recall the correlation coefficient

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2}\sqrt{\sum(y_i - \bar{y})^2}}.$$

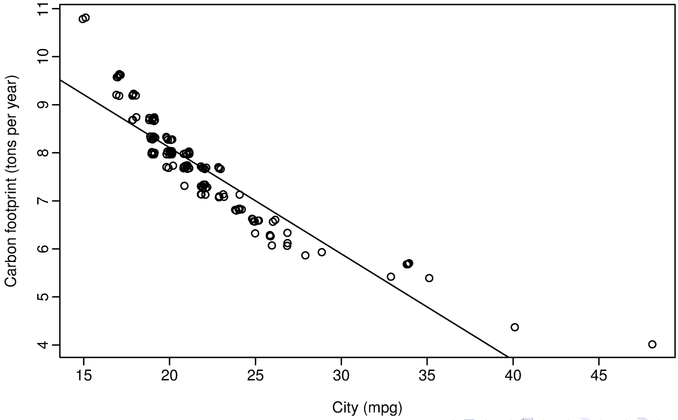
- The slope coefficient $\hat{\beta}_1$ can be written as

$$\hat{\beta}_1 = r\frac{s_y}{s_x},$$

  where s_x and s_y are the standard deviation of the x and y observations respectively.
- Connection and differences between regression and correlation.

Carbon footprint

- This is a regression between city mpg and the carbon footprint of 134 different car models.



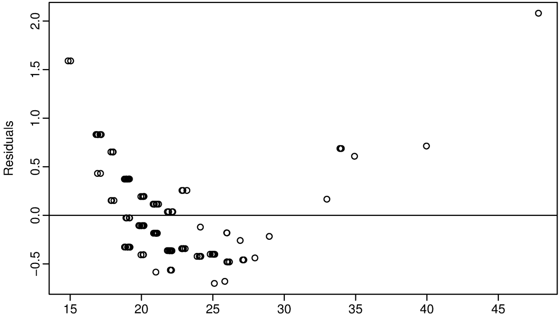
R code and outputs

```
plot(jitter(Carbon) ~ jitter(City),xlab="City (mpg)",
  ylab="Carbon footprint (tons per year)",data=fuel)
fit <- lm(Carbon ~ City, data=fuel)
abline(fit)

> fit
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.525647   0.199232   62.87   <2e-16 ***
City        -0.220970   0.008878  -24.89   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.4703 on 132 degrees of freedom
Multiple R-squared: 0.8244, Adjusted R-squared: 0.823
F-statistic: 619.5 on 1 and 132 DF,  p-value: < 2.2e-16
```

Residual analysis

- We expect the residuals to scatter around 0 and do not show systematic patterns.

```
res <- residuals(fit)
plot(jitter(res)~jitter(City), ylab="Residuals",
xlab="City", data=fuel)
abline(0,0)
```



Goodness of fit

- The concept of R^2: the proportion of variation in the forecast variable that is accounted for (or explained) by the regression model.
- A high R^2 does not always indicate a good model for estimation and forecasting.
- For instance, in the car example, $R^2 = 82\%$, which is quite high. But from the residual analysis, we know that the linear regression model is not a good fit for the data.
- For simple regression, the R^2 equals the square of the correlation coefficient between x and y.

Residual sum of square

- SS residual:

$$s_e^2 = \frac{1}{n-2}\sum_{i=1}^{n} e_i^2.$$

- The standard error is related to the size of the average error that the model produces.
- This quantity is scale dependent. It's also used for generating forecasting intervals.

Forecasting

- Forecasts from a simple regression model for a specific "new" x:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x.$$

- The prediction interval for this forecast is

$$\hat{y} \pm z_{\alpha/2} s_e \sqrt{1 + 1/n + \frac{(x - \bar{x})^2}{(n-1)s_x^2}}.$$

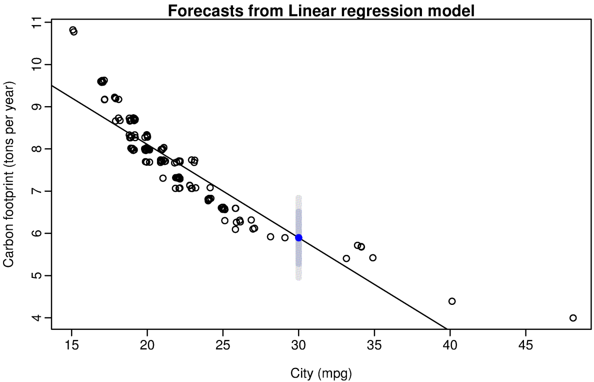
- The estimated regression line for the car example is

$$\hat{y} = 12.53 - 0.22x.$$

- For a new car model with city mpg=30, the forecasted carbon footprint is $\hat{y} = 5.9$ tons of CO2/year. We can also compute the corresponding forecasting intervals.

Forecasting

- Forecast with 80% and 95% forecast intervals for a car with 30 city mpg.



Inferences

- You may be interested in testing whether the pre?dictor variable x has had a significant effect on y.
- If x and y are unrelated, then the slope parameter $\beta_1 = 0$. We can construct a test to see if it is plausible given the observed data.

$$H_0 : \beta_1 = 0.$$

- It is also some times useful to provide an interval estimate for β_1 and β_0.

```
confint(fit,level=0.95)
                 2.5 %     97.5 %
(Intercept) 12.1315464 12.9197478
City        -0.2385315 -0.2034092
```

Nonlinear model

- One simple way to estimate a nonlinear model is to transform the variables.

- The simplest way is log-log transform

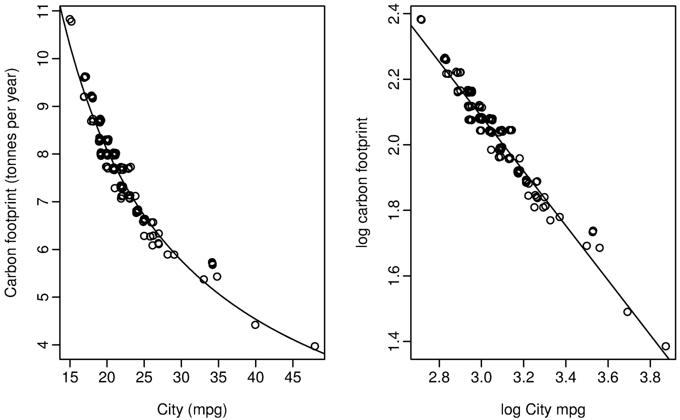
$$\log y_i = \beta_0 + \beta_1 \log x_i + \epsilon_i.$$

- Intepretation: average per?cent age change in y resulting from a 1% change in x.

- Other forms: log-linear and linear-log.

Car example

- Fitting of a log-log functional to the car data example.



Car example

- Residual of the log-log fit.