Regression and Data Fitting – Part 1

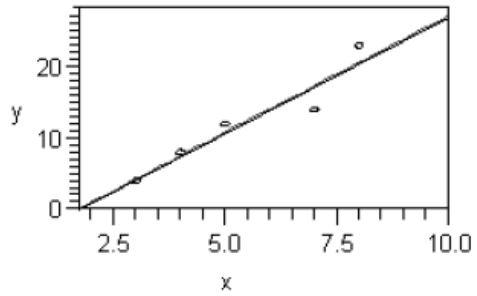
Method of Least Squares
Linear Regression - Correlation

A lecture
by
Gilberto E. Urroz

Reference: *RegressionAndDataFitting.mw*
(Maple worksheet)

Regression

- Relationship between 2 or more variables when uncertainty exists
- E.g., scatterplot suggests linear relationship



Method of Least Squares - 1

- Method of least squares
- Given data set: $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ calculate coefficients $a_1, a_2, \ldots, a_m$ in the fitting equation: $yf = f(a_1, a_2, \ldots, a_m, x)$ so that the *sum of squared errors*

$$SSE = \sum_{i=1}^{n} \left(y_i - yf_i\right)^2 = \sum_{i=1}^{n} \left(y_i - f\left(a_1, a_2, \ldots, a_m, x_i\right)\right)^2$$

is minimized.

Method of Least Squares - 2

- Set up the equations:

$$\frac{\partial}{\partial a_1} SSE = 0,$$

$$\frac{\partial}{\partial a_2} SSE = 0,$$

$$\ldots$$

$$\frac{\partial}{\partial a_m} SSE = 0,$$

- Solve for the coefficients $a_1, a_2, \ldots, a_m$

Simple linear fitting - 1

- Consider the function $yf = a_1 + a_2 x$
- The *SSE* is given by

$$SSE := \sum_{i=1}^{n} \left(y_i - a_1 - a_2 \cdot x_i\right)^2$$

$$= n\, a_1^2 + \sum_{i=1}^{n} \left(y_i^2 - 2\, y_i\, a_1 - 2\, y_i\, a_2\, x_i + 2\, a_1\, a_2\, x_i + a_2^2 x_i^2\right)$$

- Form the equations

$$Eq1 := \frac{\partial}{\partial a_1} SSE = 0 \implies 2\, n\, a_1 + \sum_{i=1}^{n} \left(-2\, y_i + 2\, a_2\, x_i\right) = 0$$

$$Eq2 := \frac{\partial}{\partial a_2} SSE = 0 \implies \sum_{i=1}^{n} \left(-2\, y_i\, x_i + 2\, a_1\, x_i + 2\, a_2\, x_i^2\right) = 0$$

Simple linear fitting - 2

- Use the definitions:

$$S_x = \sum_{i=1}^{n} x_i,\ S_{xx} = \sum_{i=1}^{n} x_i^2,\ S_y = \sum_{i=1}^{n} y_i,\ S_{xy} = \sum_{i=1}^{n} x_i \cdot y_i$$

- The equations become

$$Eq1 := n \cdot a_1 - S_y - a_2 \cdot S_x = 0 \implies n\, a_1 - S_y - a_2 S_x = 0$$

$$Eq2 := -S_{xy} + a_1 \cdot S_x + a_2 \cdot S_{xx} = 0 \implies -S_{xy} + a_1 S_x + a_2 S_{xx} = 0$$

- The solutions are:

$$solve\left(\{Eq1, Eq2\}, \{a_1, a_2\}\right) \implies \left\{a_1 = \frac{S_y S_{xx} + S_x S_{xy}}{n S_{xx} + S_x^2},\ a_2 = \frac{n S_{xy} - S_x S_y}{n S_{xx} + S_x^2}\right\}$$

Package: *CurveFitting* Function: *LeastSquares*

- Linear relationship – data given as a matrix

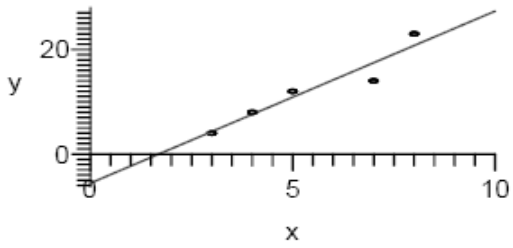
$$restart : with(CurveFitting) : with(plots) : with(Statistics) :$$
$$A := [[3,4],[4,8],[5,12],[7,14],[8,23]]$$
$$[[3,4],[4,8],[5,12],[7,14],[8,23]]$$

- Invoke function *LeastSquares*

$$y := LeastSquares(A, x) \Rightarrow -\frac{479}{86} + \frac{283}{86}\,x$$
$$evalf(y) \Rightarrow -5.569767442 + 3.290697674\,x$$

Plot original and fitted data

$$p1 := listplot(A, style = point, symbol = circle) :$$
$$p2 := plot(y, x = 0..10) :$$
$$display([p1, p2], labels = ["x", "y"])$$

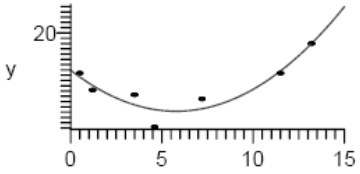


Linear relationship – data given as two lists

$$xx := [1,2,3,4,5] : yy := [0.23, 1.23, 5.34, 7.23, 10.2] :$$
$$y := LeastSquares(xx, yy, x)$$
$$-2.936000000 + 2.59399999999999852\,x$$
$$p1 := ScatterPlot(xx, yy, symbol = circle) :$$
$$p2 := plot(y, x = 0..6) :$$
$$display([p1, p2], labels = ["x", "y"])$$

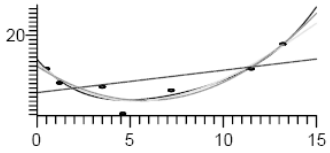
Quadratic relationship

$$xx := [0.5, 1.2, 3.5, 4.6, 7.2, 11.5, 13.2] : yy := [12.4, 9.2, 8.3, 2.2, 7.5, 12.4, 18.0] :$$
$$y := LeastSquares(xx, yy, x, curve = a + b \cdot x + c \cdot x^2)$$
$$12.94378502 - 2.68982596069865698\,x + 0.232720553690006604\,x^2$$
$$p1 := ScatterPlot(xx, yy, symbol = circle) :$$
$$p2 := plot(y, x = 0..15) :$$
$$display([p1, p2], labels = ["x", "y"])$$



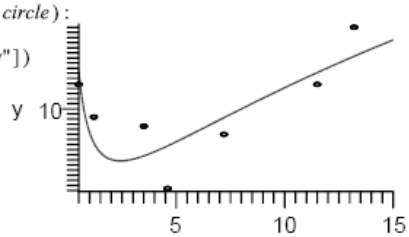
Various polynomial fittings

$$xx := [0.5, 1.2, 3.5, 4.6, 7.2, 11.5, 13.2] : yy := [12.4, 9.2, 8.3, 2.2, 7.5, 12.4, 18.0] :$$
$$y1 := LeastSquares(xx, yy, x, curve = a + b \cdot x)$$
$$6.976183111 + 0.507595161163212128\,x$$
$$y2 := LeastSquares(xx, yy, x, curve = a + b \cdot x + c \cdot x^2)$$
$$12.94378502 - 2.68982596069865698\,x + 0.232720553690006604\,x^2$$
$$y3 := LeastSquares(xx, yy, x, curve = a + b \cdot x + c \cdot x^2 + d \cdot x^3)$$
$$13.56814473 - 3.40093509997067356\,x + 0.379042552451421542\,x^2 - 0.0074454743510249154 0\,x^3$$
$$y4 := LeastSquares(xx, yy, x, curve = a + b \cdot x + c \cdot x^2 + d \cdot x^3 + e \cdot x^4)$$
$$14.46523955 - 4.87629121937350440\,x + 0.878002136366566986\,x^2 + 0.00207460920718055856\,x^4 - 0.0644434855567681958\,x^3$$
$$p1 := ScatterPlot(xx, yy, symbol = circle) :$$
$$p2 := plot([y1, y2, y3, y4], x = 0..15) :$$
$$display([p1, p2])$$



Quadratic logarithmic relationship

$$xx := [0.5, 1.2, 3.5, 4.6, 7.2, 11.5, 13.2] : yy := [12.4, 9.2, 8.3, 2.2, 7.5, 12.4, 18.0] :$$
$$y := LeastSquares(xx, yy, x, curve = a + b \cdot ln(x) + c \cdot ln(x)^2)$$
$$7.679924978 + 3.56372219939489466 \ln(x)^2 - 6.29853349240963122 \ln(x)$$
$$p1 := ScatterPlot(xx, yy, symbol = circle) :$$
$$p2 := plot(y, x = 0.5..15) :$$
$$display([p1, p2], labels = ["x", "y"])$$



Quadratic logarithmic relationship – requires simple logarithms

$xx := [0.5, 1.2, 3.5, 4.6, 7.2, 11.5, 13.2] : yy := [12.4, 9.2, 8.3, 2.2, 7.5, 12.4, 18.0] :$

$y := LeastSquares(xx, yy, x, curve = a \cdot \ln(c \cdot x) + b \cdot \ln(d \cdot x)^2)$

```
Error, (in CurveFitting:-LeastSquares) curve to fit
is not linear in the parameters
```

Note:

$$f(x) = a\ln(c) + a\ln(x) + b\,(\ln(d) + \ln(x))^2 =$$
$$a\ln(c) + a\ln(x) + b\ln(d)^2 + 2b\ln(d)\ln(x) + b\ln(x)^2 =$$
$$(a\ln(c) + b\ln(d)^2) + (a + 2b\ln(d))\ln(x) + b\ln(x)^2 =$$
$$A + B\ln(x) + C\ln(x)^2$$

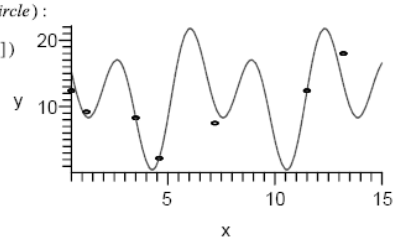
Sine-cosine combination

$xx := [0.5, 1.2, 3.5, 4.6, 7.2, 11.5, 13.2] : yy := [12.4, 9.2, 8.3, 2.2, 7.5, 12.4, 18.0] :$

$y := LeastSquares\left(xx, yy, x, curve = a_0 + \sum_{i=1}^{2}\left(a_i \cdot \cos(i \cdot x) + b_i \cdot \sin(i \cdot x)\right)\right)$

$$11.81759880 + 3.63356530012640944\cos(x) + 2.82100762899654534\sin(x)$$
$$+ 5.43446413468777400\cos(2x) - 4.89454688964146188\sin(2x)$$

$p1 := ScatterPlot(xx, yy, symbol = circle) :$
$p2 := plot(y, x = 0.5..15) :$
$display([p1, p2], labels = ["x", "y"])$



Linear regression – Example 1

Example 1 - Discharge and drainage area

Let y be the peak discharge (ft^3/s) from a basin whose drainage area (*acres*) is x. Determine the least-square fitting, $yf = a + bx$, for this data set (data are entered below).

Solution. Simply type in the data and use function *LeastSquares* from package *CurveFitting*, as shown next:

$restart : with(CurveFitting) : with(Statistics) : with(plots) :$
$xx := [6, 10, 13, 17, 23, 26, 28] : yy := [21, 24, 52, 41, 43, 71, 63] :$
A graph of the data shows that a linear relationship can be fit to the data:
$ScatterPlot(xx, yy, symbol = circle)$
The least-square fitting is given next:
$y := evalf(LeastSquares(xx, yy, x))$

$$11.54166667 + 1.904132791\,x$$

To compare the fitting with the original data set use:
$p1 := ScatterPlot(xx, yy, symbol = circle, labels = ["A(acres)", "Q(cfs)"]) :$
$p2 := plot(y, x = 0..30) :$
$display([p1, p2])$

Linear regression – Example 2

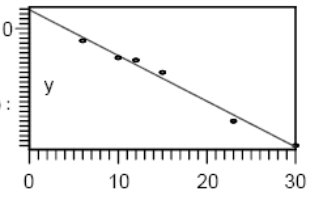
Example 2 - Rainfall and duration

Let x be the storm duration (*min*) and y be the rainfall intensity (*in/hr*) for a given meteorological station. Determine the least-square fitting, $yf = a + bx$, for this data set (data are entered below).

$xx := [6, 10, 12, 15, 23, 30] : yy := [9.5, 8.8, 8.7, 8.2, 6.2, 5.2] :$
A graph of the data shows a decreasing linear relationship:
$ScatterPlot(xx, yy, symbol = circle, axes = boxed)$
The least-square fitting is given by:
$y := LeastSquares(xx, yy, x)$

$$10.76566164 - 0.187437185929648420\,x$$

To compare the fitting with the original data use:
$p1 := ScatterPlot(xx, yy, symbol = circle, axes = boxed) :$
$p2 := plot(y, x = 0..30, labels = ["x", "y"]) :$
$display([p1, p2])$



Covariance and correlation coefficient

- Given data sets $[x_1, x_2, \ldots, x_n], [y_1, y_2, \ldots, y_n]$
- Mean values and standard deviations:

  $(\bar{x}, \bar{y})$ (s_x, s_y)

- Covariance:

$$s_{xy} = \frac{1}{n} \cdot \sum_{i=1}^{n}\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)$$

- Correlation coefficient:

$$r_{xy} = \frac{n}{n-1} \cdot \frac{s_{xy}}{s_x \cdot s_y}$$

Correlation coefficient

- If x and y are strongly correlated r_{xy} should be close to ± 1
  - For y increasing with x, r_{xy} should be near +1
  - For y decreasing with x, r_{xy} should be near -1
- If there is little correlation between x and y, r_{xy} should be near zero.

Covariance and *Correlation* functions in *Maple* – Example 1

- Belong in the *Statistics* package

(Example 1 in *Simple linear regression examples*) we have:

$xx := [6, 10, 13, 17, 23, 26, 28] : yy := [21, 24, 52, 41, 43, 71, 63] :$

$sxy := Covariance(xx, yy)$

114.7142857

$rxy := Correlation(xx, yy)$

0.8561487692

This result shows a correlation coefficient of about *0.86*, a value that indicates a good correlation between x and y. Perfect correlation would be indicated by a value of $r_{xy} = +1$.

Covariance and *Correlation* functions in *Maple* – Example 2

For the of Example 2 in the section *Simple linear regression examples* we have:

$xx := [6, 10, 12, 15, 23, 30] : yy := [9.5, 8.8, 8.7, 8.2, 6.2, 5.2] :$

$sxy := Covariance(xx, yy)$

−12.43333333

$rxy := Correlation(xx, yy)$

−.9932562144

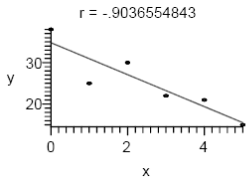
This result shows a correlation coefficient of -0.99. Very good indeed, close to the ideal value of -1 for a perfect correlation.

Examples of different correlation coefficients - 1

- Use a user-defined program *plotCorrelation*:

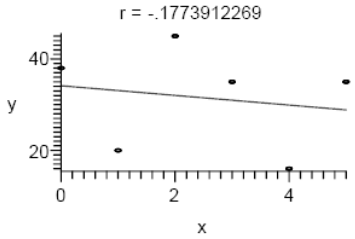
```
plotCorrelation := proc(xx, yy)
  local xy, p1, p2, sxy, rxy, x, y :
  p1 := Statistics[ScatterPlot](xx, yy, symbol = circle) :
  y := CurveFitting[LeastSquares](xx, yy, x) :
  p2 := plot(y, x = stats[describe, range](xx), labels = ["x", "y"]) :
  rxy := evalf(Statistics[Correlation](xx, yy)) :
  plots[display]([p1, p2], title = cat("r = ", convert(rxy, string))) :
end proc:

xx := [0, 1, 2, 3, 4, 5] : yy := [38, 25, 30, 22, 21, 15] : plotCorrelation(xx, yy)
```



Examples of different correlation coefficients - 2

```
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 32, 25, 18, 15] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 38, 29, 16, 15] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 40, 30, 14, 22] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 45, 32, 18, 25] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 45, 35, 16, 30] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 45, 35, 16, 35] : plotCorrelation(xx, yy)
```



Examples of different correlation coefficients - 3

```
xx := [0, 1, 2, 3, 4, 5] : yy := [38, 20, 45, 35, 16, 42.5] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [35, 16, 35, 45, 20, 38] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [30, 16, 35, 45, 20, 38] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [25, 18, 32, 45, 20, 38] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [22, 14, 30, 40, 20, 38] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [15, 16, 29, 38, 20, 38] : plotCorrelation(xx, yy)
xx := [0, 1, 2, 3, 4, 5] : yy := [15, 18, 25, 32, 20, 38] : plotCorrelation(xx, yy)
```