Linear regression model

Contents:

1. Introduction

Linear regression is probably the most widely used, and useful, statistical technique for solving environmental problems. Linear regression models are extremely powerful, and have the power to empirically tease out very complicated relationships between variables. Generally speaking, the technique is useful, among other applications, in helping explain observations of a dependent variable, usually denoted y, with observed values of one or more independent variables, usually denoted x_1, x_2, x_3 … A key feature of all regression models is the error term, which is included to capture sources of error that are not captured by other variables.

2. Mathematical basis

Suppose we have observations on *n* subjects consisting of a dependent or response variable Y and an explanatory variable *X* . The observations are usually recorded as in Table 2.1. We wish to measure both the ***direction*** and the ***strength*** of the relationship between *Y* and *X* . Two related measures, known as the ***covariance*** and the ***correlation coefficient,*** are developed below:

Table 2.1. Notation for the Data Used in Simple Regression and Correlation

Observation Number	**Response *Y***	**Predictor *X***
1	y_1	x_1
2	y_2	x_2
…		
n	y_n	x_n

On the scatter plot of Y versus X , let us draw a vertical line at $\bar{x}$ and a horizontal line at $\bar{y}$ as shown in Figure 2.1, where:

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} \quad \text{and} \quad \bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

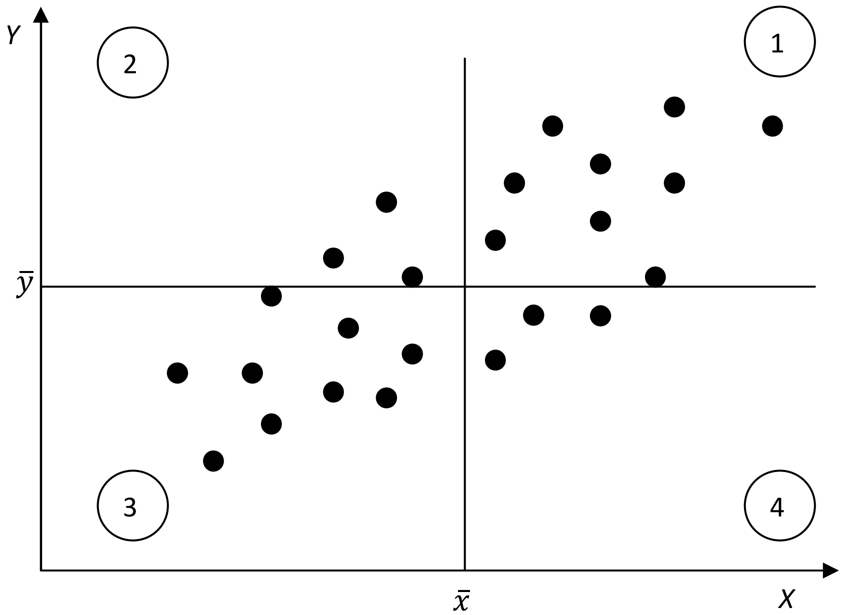
are the sample mean of Y and X, respectively. The two lines divide the graph into four quadrants. For each point i in the graph, compute the following quantities:

- $y_i - \bar{y}$, the deviation of each observation y_i from the mean of the response variable,

- $x_i - \bar{x}$ the deviation of each observation x_i from the mean of the predictor variable, and

- the product of the above two quantities, $(y_i - \bar{y})(x_i - \bar{x})$.

It is clear from the graph that the quantity $(y_i - \bar{y})$ is positive for every point in the first and second quadrants, and is negative for every point in the third and fourth quadrants.



Similarly, the quantity $(x_i - \bar{x})$ is positive for every point in the first and fourth quadrants, and is negative for every point in the second and third quadrants. These facts are summarized in Table 2.2.

Table 2.2. Algebraic Signs of the Quantities $(y_i - \bar{y})$ and $(x_i - \bar{x})$

Quadrant	$(\boldsymbol{y_i - \bar{y}})$	$(\boldsymbol{x_i - \bar{x}})$	$(\boldsymbol{y_i - \bar{y}})\ (\boldsymbol{x_i - \bar{x}})$
1	+	+	+
2	+	-	-
3	-	-	+
4	-	+	-

If the linear relationship between Y and X is positive (as X increases Y also increases), then there are more points in the first and third quadrants than in the second and fourth quadrants. In this case, the sum of the last column in Table 2.2 is likely to be positive because there are more positive than negative quantities. Conversely, if the relationship between Y and X is negative (as *X* increases *Y* decreases), then there are more points in the second and fourth quadrants than in the first and third quadrants. Hence the sum of the last column in Table 2.2 is likely to be negative. Therefore, the sign of the quantity:

$$Cov(Y, X) = \frac{\sum_{i=1}^{n}(y_i - \bar{y})\ (x_i - \bar{x})}{n - 1}$$

which is known as the *covariance* between *Y* and *X*, indicates the direction of the linear relationship between *Y* and *X*. If Cov(*Y*, *X)* > 0, then there is a positive relationship between *Y* and *X*, but if Cov(*Y*, *X*) < 0, then the relationship is negative. Unfortunately, Cov(*Y*, *X*) does not tell us much about the strength of such a relationship because it is affected by changes in the units of measurement. For example, we would get two different values for the Cov(*Y*, *X*) if we report *Y* and/or *X* in terms of thousands of dollars instead of dollars. To avoid this disadvantage of the covariance, we *standardize* the data before computing the covariance. To standardize the Y data, we first subtract the mean from each observation then divide by the standard deviation, that is, we compute:

$$z_i = \frac{y_i - \bar{y}}{S_y} \qquad (2.3)$$

where

$$S_y = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{n-1}} \qquad (2.4)$$

is the sample *standard deviation* of *Y*. It can be shown that the standardized variable *Z* in (2.3) has mean zero and standard deviation one. We standardize *X* in a similar way by subtracting the mean $\bar{x}$ from each observation x_i then divide by the standard deviation s_x. The covariance between the standardized *X* and *Y* data is known as the ***correlation coefficient*** between *Y* and *X* and is given by

$$Cor(Y, X) = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{y_i - \bar{y}}{s_y}\right)\left(\frac{x_i - \bar{x}}{s_x}\right) \qquad (2.5)$$

Equivalent formulas for the correlation coefficient are

$$Cor(Y, X) = \frac{Cov(Y,X)}{s_y s_x} \qquad (2.6)$$

$$= \frac{\sum(y_i - \bar{y})(x_i - \bar{x})}{\sqrt{\sum(y_i - \bar{y})^2 \sum(x_i - \bar{x})^2}} \qquad (2.7)$$

Thus, Cor(Y, X) can be interpreted either as the covariance between the standardized variables or the ratio of the covariance to the standard deviations of the two variables. From *(2.5),* it can be seen that the correlation coefficient is symmetric, that is, $Cor(Y, X) = Cor(X, Y)$.

Unlike $Cov(Y, X)$, $Cor(Y, X)$ is scale invariant, that is, it does not change if we change the units of measurements. Furthermore, $Cor(Y, X)$ satisfies

$$-1 \le Cor(Y, X) \le 1 \qquad (2.8)$$

These properties make the $Cor(Y, X)$ a useful quantity for measuring both the direction and the strength of the relationship between *Y* and *X* . The magnitude of $Cor(Y, X)$ measures the strength of the linear relationship between *Y* and *X .* The closer $Cor(Y, X)$ is to 1 or -1, the stronger is the relationship between *Y* and *X .* The sign of $Cor(Y, X)$ indicates the direction of the relationship between *Y* and *X* . That is, $Cor(Y, X) > 0$ implies that *Y* and *X* are positively related. Conversely, $Cor(Y, X) < 0$, implies that *Y* and *X* are negatively related.

Note, however, that $Cor(Y, X) = 0$ does not necessarily mean that *Y* and *X* are not related. It only implies that they are not linearly related because the correlation coefficient measures only ***linear*** relationships. In other words, the $Cor(Y, X\ )$ can still be zero when *Y* and *X* are nonlinearly related. Furthermore, like many other summary statistics, the $Cor(Y, X)$ can be substantially influenced by one or few outliers in the data.

The relationship between a response variable *Y* and a predictor variable *X* is postulated as a linear model:

$$Y = \beta_0 + \beta_1 X + \varepsilon \qquad (2.9)$$

where β_0 and β_1*,* are constants called the ***model regression coefficients*** or ***parameters,*** and ε is a random disturbance or error. It is assumed that in the range of the observations studied, the linear equation (2.9) provides an acceptable approximation to the true relation between *Y* and *X*. In other words, *Y* is approximately a linear function of *X ,* and *E* measures the discrepancy in that approximation.

In particular *E* contains no systematic information for determining *Y* that is not already captured in *X*. The coefficient β_1*,* called the ***slope,*** may be interpreted as the change in *Y* for unit change in *X*. The coefficient β_0*,* called the ***constant*** coefficient or ***intercept,*** is the predicted value of *Y* when *X* = 0.

According to (2.9), each observation in Table 2.1 can be written as

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \ i = 1, 2, 3, \dots n, \qquad (2.10)$$

where y_i represents the i th value of the response variable *Y* , x_i represents the i th value of the predictor variable ***X,*** and ε represents the error in the approximation of y_i.

Regression analysis differs in an important way from correlation analysis. The correlation coefficient is symmetric in the sense that $Cor(Y, X)$ is the same as $or(X, Y\ )$. The variables *X* and *Y* are of equal importance. In regression analysis the response variable *Y* is of primary importance. The importance of the predictor *X* lies on its ability to account for the variability of the response variable *Y* and not in itself per se. Hence *Y* is of primary importance.

Based on the available data, we wish to estimate the parameters β_0 and β_1. This is equivalent to finding the straight line that gives the *best fit* (representation) of the points in the scatter plot of the response versus the predictor variable (see Figure 2.4). We estimate the parameters using the popular *least squares method,* which gives the line that minimizes the sum of squares of the *vertical distances* from each point to the line. The vertical distances represent the errors in the response variable. These errors can be obtained by rewriting (2.10) as:

$$\varepsilon_i = y_i - \beta_0 - \beta_1 x_i,\ i = 1, 2, 3, \dots n, \qquad (2.12)$$

The sum of squares of these distances can then be written as:

$$S(\beta_0, \beta_1) = \sum_{i=1}^{n} \varepsilon_i{}^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2. \qquad (2.13)$$

The values of $\widehat{\beta_0}$ and $\widehat{\beta_1}$ that minimize $S(\beta_0, \beta_1)$ are given by:

$$\widehat{\beta_1} = \frac{\sum (y_i - \bar{y})\,(x_i - \bar{x})}{\sum (x_i - \bar{x})^2} \qquad (2.14)$$

and

$$\widehat{\beta_0} = \bar{y} - \widehat{\beta_1}\bar{x} \qquad (2.15)$$

The estimates $\widehat{\beta_0}$ and $\widehat{\beta_1}$ are called the least squares estimates of β_0 and β_1 because they are the solution to the *least squares method*, the intercept and the slope of the line that has the smallest possible sum of squares of the vertical distances from each point to the line. For this reason, the line is called the *least squares regression line*. The least squares regression line is given by:

$$\hat{Y} = \widehat{\beta_0} + \widehat{\beta_1}X. \tag{2.16}$$

For each observation in our data we can compute:

$$\widehat{y_i} = \widehat{\beta_0} + \widehat{\beta_1}x_i \tag{2.17}$$

These are called the *fitted* values. Thus, the i th fitted value, $\widehat{y_i}$ *,* is the point on the least squares regression line (2.16) corresponding to x_i. The vertical distance corresponding to the i th observation is:

$$e_i = y_i - y_i\,, \; i = 1, 2, 3, \dots, n \tag{2.18}$$

These vertical distances are called the *ordinary least squares residuals*. One properties of the residuals in (2.18) is that their sum is zero. This means that the sum of the distances above the line is equal to the sum of the distances below the line.

3. Example: Student statistics data

Let's consider a table of ten students. In order to study the relationships between the learning time, each chapter grades and the final grade, a table of records is taken.

Table 2.3.

Row	**Time (hours)**	**Chapter 1 mark**	**Chapter 2 mark**	**Final grade**
1	64	7	8	8
2	59	7	7	7
3	66	8	7	8
4	60	7	8	8
5	80	9	10	10
6	85	10	9	10

7	70	10	9	9
8	72	10	10	10
9	45	5	6	6
10	57	5	6	7
11	56	5	7	7
12	60	9	7	8
13	61	5	8	7
14	67	6	8	8
15	64	7	8	8
16	65	8	8	8
17	78	10	10	10
18	79	10	9	10
19	78	8	9	9
20	49	8	5	6
21	53	9	7	7
22	71	8	10	9
23	91	8	10	10

We focus on the data representing the time spent by each student to study (the response variable) and the final grade (the predictor variable).

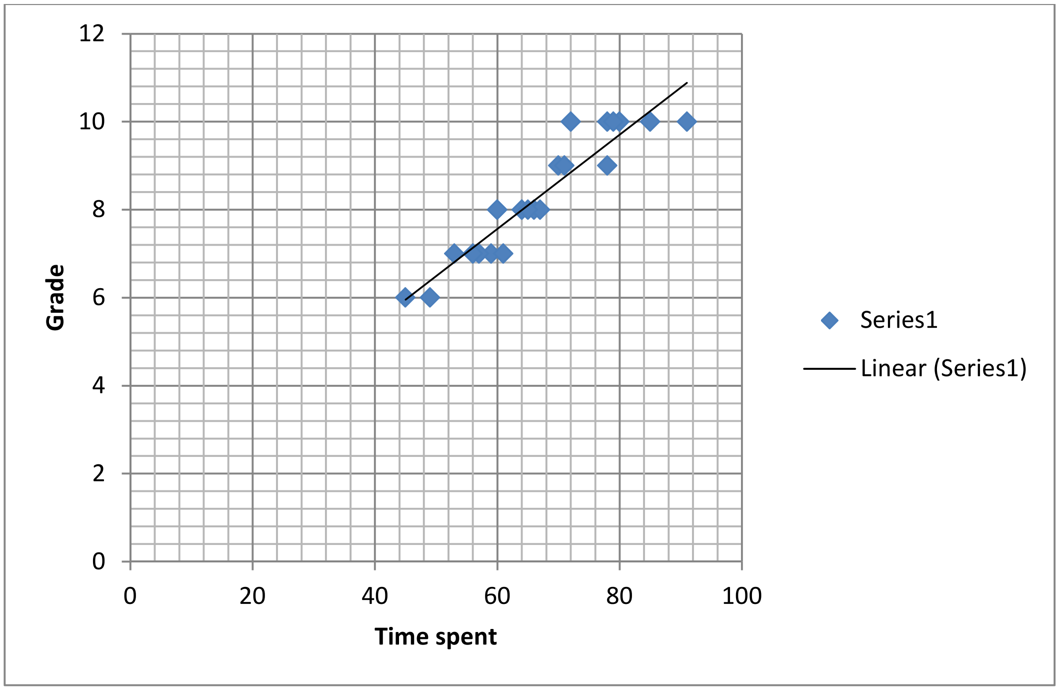


Fig 2.4 Scatter plot of Time spent versus Final grade.

The quantities needed to compute $\bar{y}$, $\bar{x}$, $Cov(Y, X\ )$, and $Cor(Y, X)$:

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} = \frac{190}{23} = 8.26$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{1530}{23} = 66.52$$

$$Cov(Y, X) = \frac{\sum_{i=1}^{n}(y_i - \bar{y})\ (x_i - \bar{x})}{n-1} = \frac{315.86}{22} = 14.35$$

$$Cor(Y, X) = \frac{\sum(y_i - \bar{y})(x_i - \bar{x})}{\sqrt{\sum(y_i - \bar{y})^2 \sum(x_i - \bar{x})^2}} = \frac{315.86}{\sqrt{38.434803 * 2945.739}} = 0.938$$

Before drawing conclusions from this value of $or(Y, X\ )$, we should examine the corresponding scatter plot of Y versus X . This plot is given in Figure 2.4. The high value of $Cor(Y, X)$ = 0.938 is consistent with the strong linear relationship between *Y* and *X* exhibited in Figure 2.4. We therefore conclude that there is a strong positive relationship between time spent and the final grade. Although $Cor(Y, X)$ is a useful quantity for measuring the direction and the strength of linear relationships, it cannot be used for prediction purposes, that is, we cannot use $Cor(Y, X)$ to predict the value of one variable given the value of the other.

Now, we must compute the regression parameters:

Using the data from Table 2.3, we have:

$$\widehat{\beta_1} = \frac{\sum(y_i - \bar{y})\ (x_i - \bar{x})}{\sum(x_i - \bar{x})^2} = \frac{315.86}{2945.739} = 0.107$$

and

$$\widehat{\beta_0}\ = \bar{y} - \ \widehat{\beta_1}\bar{x} = 8.26 - 0.107 * 66.52 = 1.129.$$

Then the equation of the least squares regression line is:

$$Final\ Grade = 0.107 * Time + 1.129 \qquad (2.19)$$

Table 2.5 The Fitted Values, $\widehat{y_i}$, and the Ordinary Least Squares Residuals, e_i, for the Student Data Set

i	x_i	y_i	$\widehat{y_i}$	e_i
1	64	8	7.97	0.03
2	59	7	7.44	-0.44
3	66	8	8.19	-0.19
4	60	8	7.54	0.46
5	80	10	9.68	0.32
6	85	10	10.22	-0.22
7	70	9	8.61	0.39
8	72	10	8.83	1.17
9	45	6	5.94	0.06
10	57	7	7.22	-0.22
11	56	7	7.12	-0.12
12	60	8	7.54	0.46
13	61	7	7.65	-0.65
14	67	8	8.29	-0.29
15	64	8	7.97	0.03
16	65	8	8.08	-0.08
17	78	10	9.47	0.53
18	79	10	9.58	0.42
19	78	9	9.47	-0.47
20	49	6	6.37	-0.37
21	53	7	6.80	0.20
22	71	9	8.72	0.28
23	91	10	10.86	-0.86

4. WEKA

Regression is the easiest technique to use, but is also probably the least powerful (funny how that always goes hand in hand). This model can be as easy as one input variable and one output variable (called a Scatter diagram in Excel, or an XYDiagram in OpenOffice.org). Of course, it can get more complex than that, including dozens of input variables. In effect, regression models all fit the same general pattern. There are a number of independent variables, which, when taken together, produce a result — a dependent variable. The regression model is then used to predict the result of an unknown dependent variable, given the values of the independent variables.

Let's take an example with student grades-based regression model and create some data to examine.

Example 1: Let's try the example from chapter 3. As it was specified, we focus on the data representing the time spent by each student to study (the response variable) and the final grade (the predictor variable). We want to predict the final grade for a student who's study time is 55 hours.

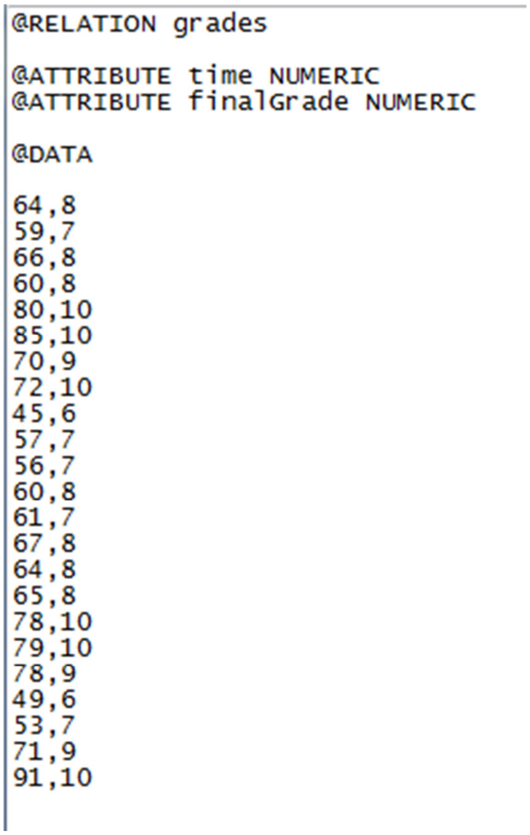
i	x_i	y_i
1	64	8
2	59	7
3	66	8
4	60	8
5	80	10
6	85	10
7	70	9
8	72	10
9	45	6
10	57	7
11	56	7
12	60	8
13	61	7
14	67	8
15	64	8
16	65	8
17	78	10
18	79	10
19	78	9

20	49	6
21	53	7
22	71	9
23	91	10
24	55	???

Building the data set for WEKA

To load data into WEKA, we have to put it into a format that will be understood. WEKA's preferred method for loading data is in the Attribute-Relation File Format (ARFF), where you can define the type of data being loaded, then supply the data itself. In the file, you define each column and what each column contains. In the case of the regression model, you are limited to a NUMERIC or a DATE column. Finally, you supply each row of data in a comma-delimited format. The ARFF file we'll be using with WEKA appears below. Notice in the rows of data that we've left out the new student study time. Since we are creating the model, we cannot input the new student into it since the final grade is unknown.

WEKA Format:

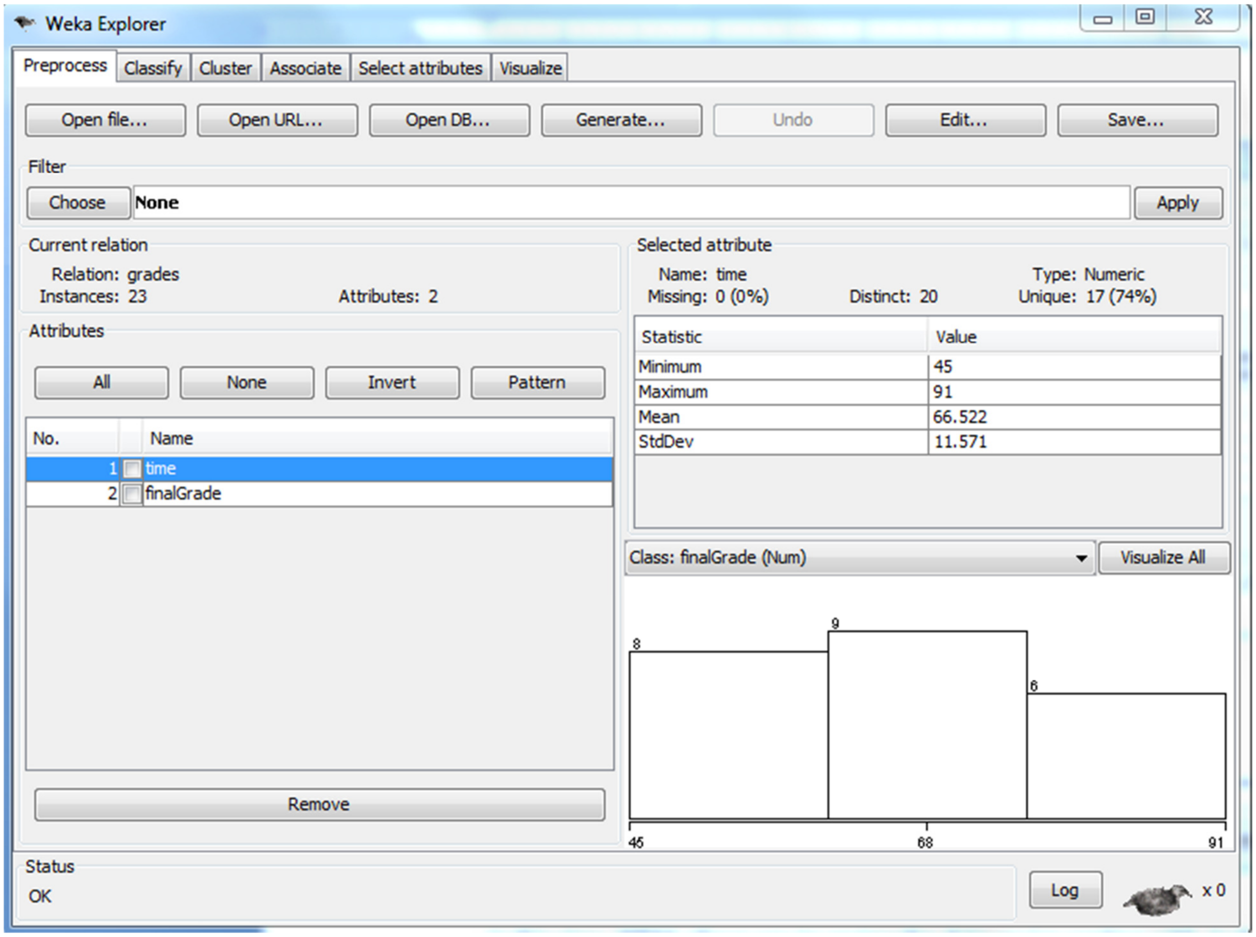


```
@RELATION grades

@ATTRIBUTE time NUMERIC
@ATTRIBUTE finalGrade NUMERIC

@DATA

64,8
59,7
66,8
60,8
80,10
85,10
70,9
72,10
45,6
57,7
56,7
60,8
61,7
67,8
64,8
65,8
78,10
79,10
78,9
49,6
53,7
71,9
91,10
```

Loading the data into WEKA

Now that the data file has been created, it's time to create our regression model. Start WEKA, then choose the **Explorer.** You'll be taken to the Explorer screen, with the **Preprocess** tab selected. Select the **Open File** button and select the ARFF file you created in the section above. After selecting the file, your WEKA Explorer should look similar to the screenshot in Figure 4.1.

Figure 4.1:



Filter: None

Current relation

Relation: grades
Instances: 23 Attributes: 2

Attributes

No.		Name
1	□	time
2	□	finalGrade

Selected attribute

Name: time Type: Numeric
Missing: 0 (0%) Distinct: 20 Unique: 17 (74%)

Statistic	Value
Minimum	45
Maximum	91
Mean	66.522
StdDev	11.571

Class: finalGrade (Num)

Status
OK

In this view, WEKA allows you to review the data you're working with. In the left section of the Explorer window, it outlines all of the columns in your data (Attributes) and the number of rows of data supplied (Instances). By selecting each column, the right section of the Explorer window will also give you information about the data in that column of your data set. For example, by selecting the **time** column in the left section (which should be selected by default), the right-section should change to show you additional statistical information about the column. It shows the maximum value in the data set for this column is 91 hours, and the minimum is grade 45 hours. The average time is 66,52 hours, with a standard deviation of 11,57. (Standard deviation is a statistical measure of variance.) Finally, there's a visual way of examining the data, which you can see by clicking the **Visualize All** button. Due to our limited number of rows in this data set, the visualization is not as powerful as it would be if there were more data points (in the hundreds, for example).

Enough looking at the data. Let's create a model and get a grade for the new student.

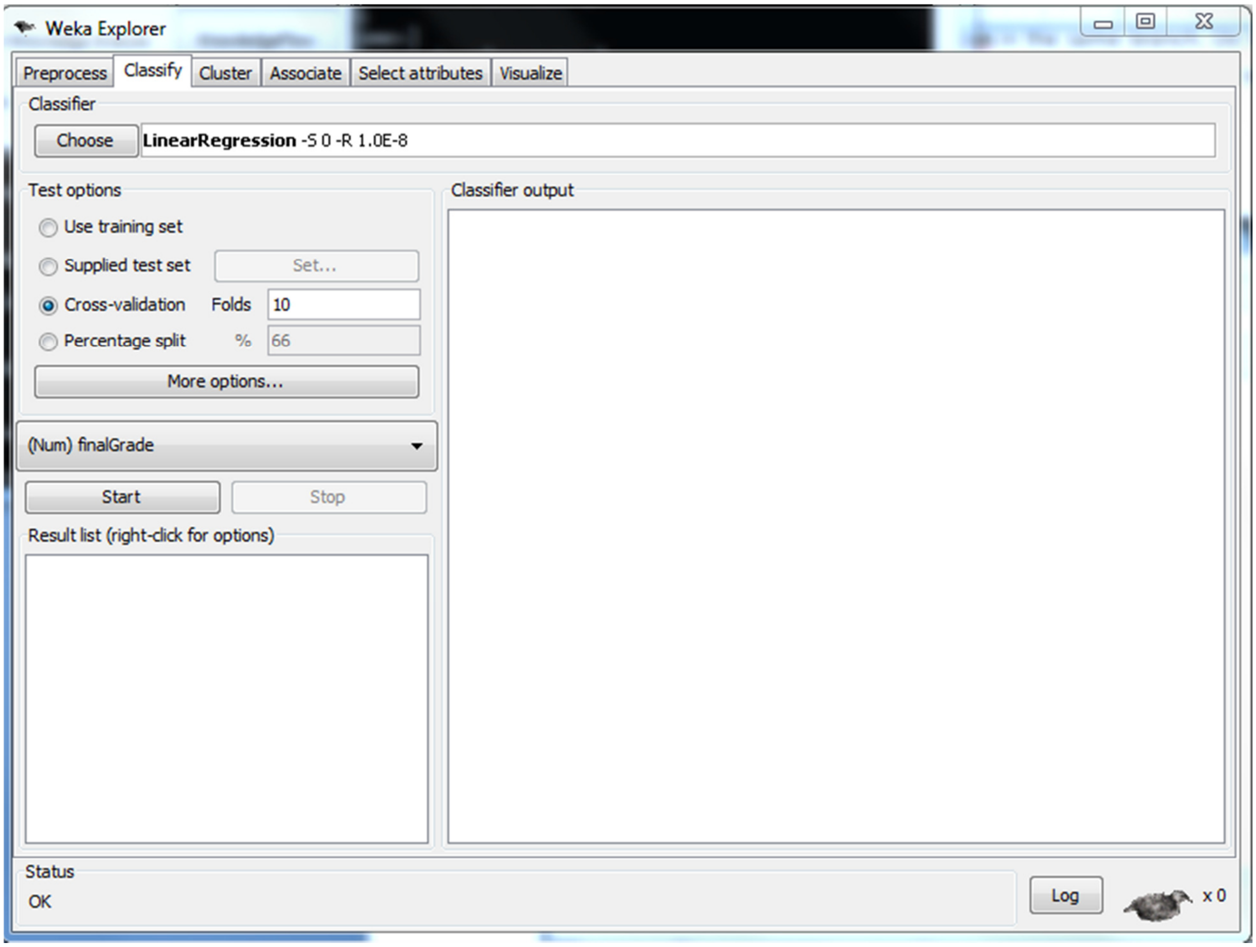
Creating the regression model with WEKA

To create the model, click on the **Classify** tab. The first step is to select the model we want to build, so WEKA knows how to work with the data, and how to create the appropriate model:

1. Click the **Choose** button, then expand the **functions** branch.
2. Select the **LinearRegression** leaf.

This tells WEKA that we want to build a regression model. As you can see from the other choices, though, there are lots of possible models to build. This should give you a good indication of how we are only touching the surface of this subject. Also of note: There is another choice called **SimpleLinearRegression** in the same branch. When you've selected the right model, your WEKA Explorer should look like Figure 4.2.

Figure 4.2:



Classifier

LinearRegression -S 0 -R 1.0E-8

Test options

○ Use training set

○ Supplied test set

● Cross-validation Folds 10

○ Percentage split % 66

(Num) finalGrade

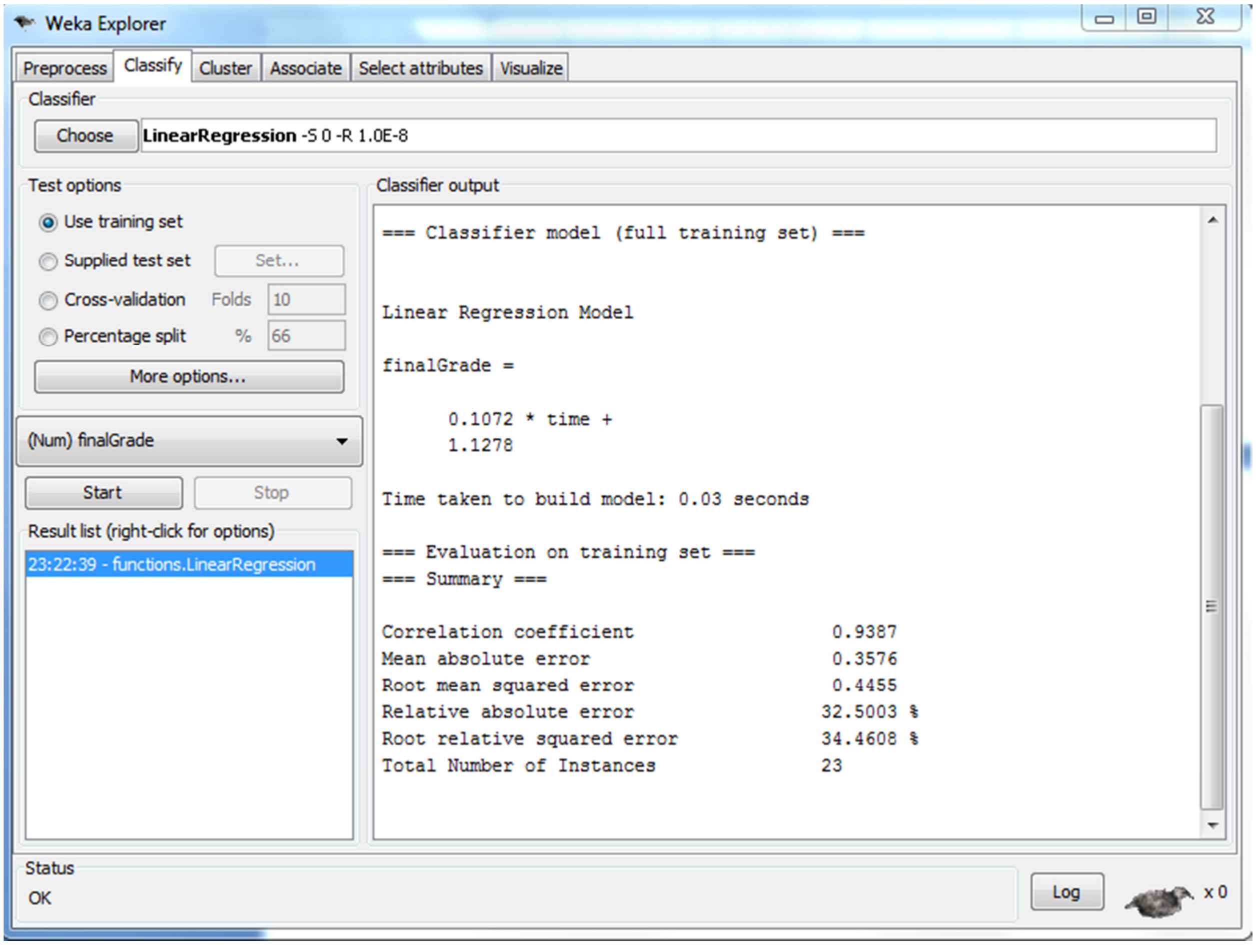
Status

OK

Now that the desired model has been chosen, we have to tell WEKA where the data is that it should use to build the model. Though it may be obvious to us that we want to use the data we supplied in the ARFF file, there are actually different options, some more advanced than what we'll be using. The other three choices are **Supplied test set,** where you can supply a different set of data to build the model; **Cross-validation,** which lets WEKA build a model based on subsets of the supplied data and then average them out to create a final model; and **Percentage split,** where WEKA takes a percentile subset of the supplied data to build a final model. These other choices are useful with different models. With regression, we can simply choose **Use training set.** This tells WEKA that to build our desired model, we can simply use the data set we supplied in our ARFF file.

Finally, the last step to creating our model is to choose the dependent variable (the column we are looking to predict). We know this should be the final grade, since that's what we're trying to determine for the new student. Right below the test options, there's a combo box that lets you choose the dependent variable. The column **finalGrade** should be selected by default. If it's not, please select it. Now we are ready to create our model. Click **Start**. Figure 4.3 shows what the output should look like.

Figure 4.3:



Classifier: LinearRegression -S 0 -R 1.0E-8

● Use training set
○ Supplied test set
○ Cross-validation Folds 10
○ Percentage split % 66

(Num) finalGrade

23:22:39 - functions.LinearRegression

Classifier output

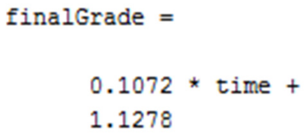
```
=== Classifier model (full training set) ===


Linear Regression Model

finalGrade =

      0.1072 * time +
      1.1278

Time taken to build model: 0.03 seconds

=== Evaluation on training set ===
=== Summary ===

Correlation coefficient                  0.9387
Mean absolute error                      0.3576
Root mean squared error                  0.4455
Relative absolute error                 32.5003 %
Root relative squared error             34.4608 %
Total Number of Instances               23
```

Status: OK

Interpreting the regression model

WEKA puts the regression model right there in the output, as shown in Listing 1.

Listing 1: Regression output



```
finalGrade =

      0.1072 * time +
      1.1278
```

It is the same result we obtained mathematically. Listing 2 shows the results, plugging in the value for the new student.

Listing 2: Final grade using regression model

$$Final\ Grade = 0.1072 * 55 + 1.1278 = 7.02$$

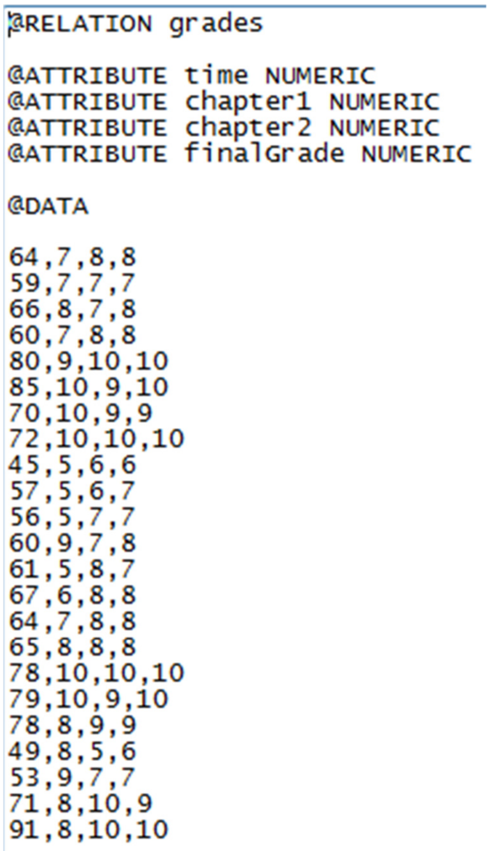
Example 2: Now let's consider another example, a little more complex. There are grades received for **chapter1**, **chapter2** evaluation, **study time** and a final grade called **finalGrade**. The last row represents grades received by a new student. We want to predict his final grade based on given data.

Row	Time (hours)	Chapter 1 mark	Chapter 2 mark	Final grade
1	64	7	8	8
2	59	7	7	7
3	66	8	7	8
4	60	7	8	8
5	80	9	10	10
6	85	10	9	10
7	70	10	9	9
8	72	10	10	10
9	45	5	6	6
10	57	5	6	7
11	56	5	7	7
12	60	9	7	8

13	61	5	8	7
14	67	6	8	8
15	64	7	8	8
16	65	8	8	8
17	78	10	10	10
18	79	10	9	10
19	78	8	9	9
20	49	8	5	6
21	53	9	7	7
22	71	8	10	9
23	91	8	10	10
24	69	7	8	???

We repeat the same steps from the example from above.

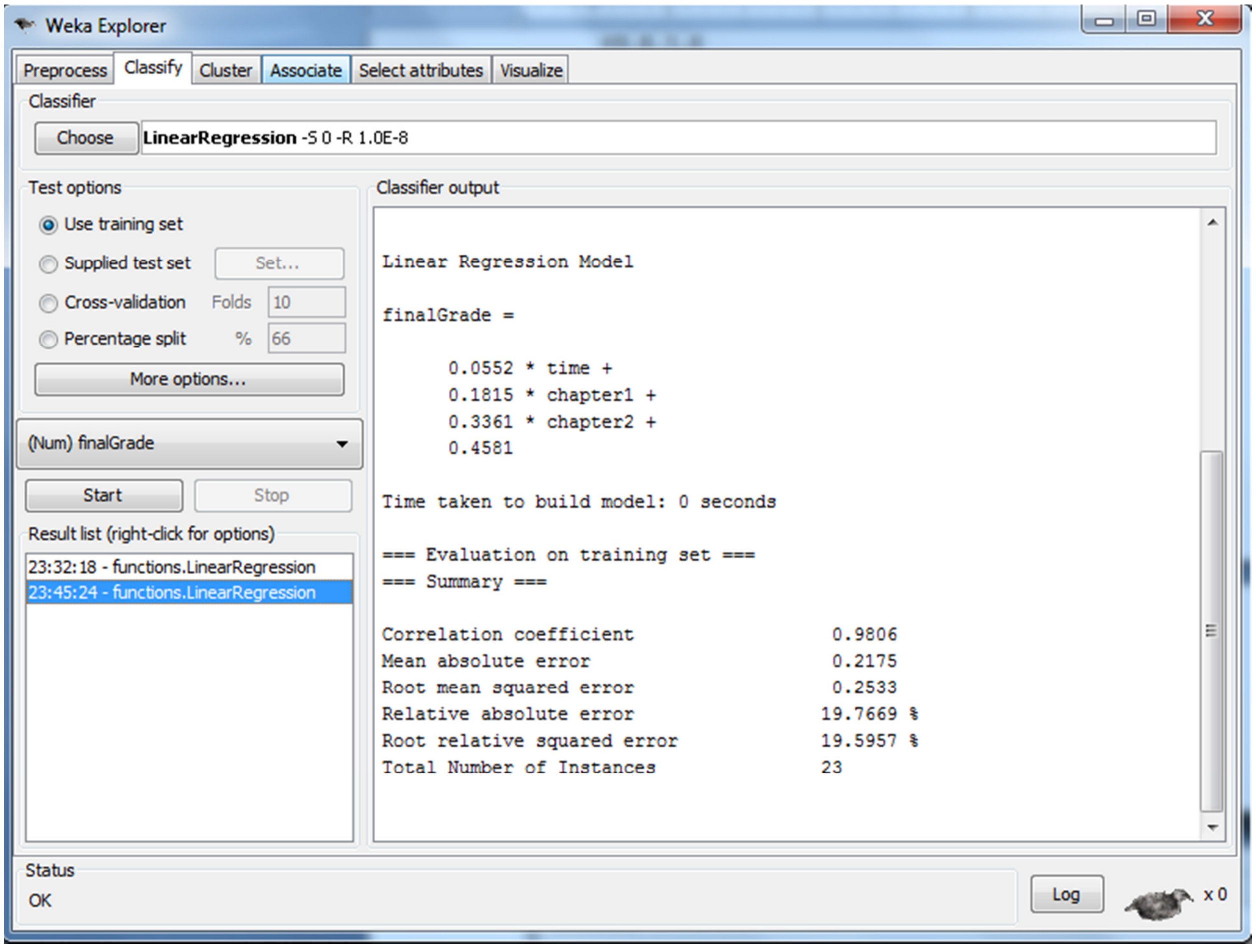
WEKA Format:



```
@RELATION grades

@ATTRIBUTE time NUMERIC
@ATTRIBUTE chapter1 NUMERIC
@ATTRIBUTE chapter2 NUMERIC
@ATTRIBUTE finalGrade NUMERIC

@DATA

64,7,8,8
59,7,7,7
66,8,7,8
60,7,8,8
80,9,10,10
85,10,9,10
70,10,9,9
72,10,10,10
45,5,6,6
57,5,6,7
56,5,7,7
60,9,7,8
61,5,8,7
67,6,8,8
64,7,8,8
65,8,8,8
78,10,10,10
79,10,9,10
78,8,9,9
49,8,5,6
53,9,7,7
71,8,10,9
91,8,10,10
```

After clicking **Start** we obtain the following model:

Figure 4.4:



LinearRegression -S 0 -R 1.0E-8

● Use training set
○ Supplied test set
○ Cross-validation Folds 10
○ Percentage split % 66

(Num) finalGrade

23:32:18 - functions.LinearRegression
23:45:24 - functions.LinearRegression

```
Linear Regression Model

finalGrade =

      0.0552 * time +
      0.1815 * chapter1 +
      0.3361 * chapter2 +
      0.4581

Time taken to build model: 0 seconds

=== Evaluation on training set ===
=== Summary ===

Correlation coefficient                  0.9806
Mean absolute error                      0.2175
Root mean squared error                  0.2533
Relative absolute error                 19.7669 %
Root relative squared error             19.5957 %
Total Number of Instances               23
```

Status
OK

Interpreting the regression model

Listing 3: Regression output

```
finalGrade =

      0.0552 * time +
      0.1815 * chapter1 +
      0.3361 * chapter2 +
      0.4581
```

Listing 4 shows the results, plugging in the values for the new student.

Listing 4: Final grade using regression model

$$Final\ Grade = 0.0552 * 69 + 0.1815 * 7 + 0.3361 * 8\ +\ 0.4581 = 8.22$$

5. Using linear regression from WEKA in Java

WEKA provides methods for accessing linear regression data. The following program written in Java highlights some of the basic functions for getting computed data:

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.util.Enumeration;
import java.util.Iterator;

import weka.classifiers.functions.LinearRegression;
import weka.core.Attribute;
import weka.core.Instances;

public class Main {

        public static void main(String[] args) throws Exception {
                BufferedReader reader = new BufferedReader(new FileReader(
                                "D:/Weka-3-6/My Files/example_v2.arff"));
                Instances data = new Instances(reader);
                reader.close();
                // setting class attribute
                data.setClassIndex(data.numAttributes() - 1);

                // Get attributes from data
                Enumeration<Attribute> enumAttr = data.enumerateAttributes();
                System.out.println("The list of attributes used: ");
                while (enumAttr.hasMoreElements()) {
                        System.out.println(enumAttr.nextElement().name());
                }

                // Create a linear regression object and assign the data set to
it
                LinearRegression linearRegression = new LinearRegression();
                linearRegression.buildClassifier(data);
```

```
            // Compute linear regression and provide the results
            System.out.println(linearRegression.toString());

            // Coefficients can be accessed separately
            double[] result = linearRegression.coefficients();
            System.out.println("Coefficients used in regression are: ");
            for (int i = 0; i < result.length; i++) {
                  System.out.println(result[i]);
            }
      }
}
```

6. References

[1] http://www.ibm.com/developerworks/opensource/library/os-weka1/index.html

[2] ***Regression Analysis by Example, Fourth Edition.*** By Samprit Chatterjee and Ali S. Hadi

Copyright @ 2006 John Wiley & **Sons,** Inc.