|  |
| --- |
|  |
| Assignment |
| Daniel Elstob |
|  |
| **Liverpool Hope University** |
| **14011862@hope.ac.uk** |

|  |
| --- |
|  |

Contents

[Tables 3](#_Toc440654301)

[Figures 5](#_Toc440654302)

[Tutorial 1 6](#_Toc440654303)

[1. Create a square matric by directly inserting values 6](#_Toc440654304)

[2. Create a matrix using functions 6](#_Toc440654305)

[3. Subtracting Matrices 6](#_Toc440654306)

[4. Multiplying Matrices 7](#_Toc440654307)

[5. Raising a matrix by a power 7](#_Toc440654308)

[6. Matrix Expression 7](#_Toc440654309)

[7. Plotting graphs 7](#_Toc440654310)

[8. Experimenting with different plots 8](#_Toc440654311)

[9. 3 Dimensional Plots 9](#_Toc440654312)

[10. Simple XY Plot 10](#_Toc440654313)

[11. Exploring Plot Tools 11](#_Toc440654314)

[12. Importing Data 13](#_Toc440654315)

[Tutorial 2 15](#_Toc440654316)

[1. Using the given dataset 15](#_Toc440654317)

[1a) Plot a graph Time against Temperature and Title 15](#_Toc440654318)

[1b) Find the average temperature 15](#_Toc440654319)

[2. Consider the equation. For different value of x, solve. 16](#_Toc440654320)

[3. Consider the equation. 16](#_Toc440654321)

[3a) 16](#_Toc440654322)

[3b) Create a script of the equation 17](#_Toc440654323)

[Tutorial 3 18](#_Toc440654324)

[1. Temperature and Pressure 18](#_Toc440654325)

[1a) 18](#_Toc440654326)

[1b) 20](#_Toc440654327)

[2. National oceanic and atmospheric administration weather data. 21](#_Toc440654328)

[2a) Total precipitation in each month 21](#_Toc440654329)

[2b) Total precipitation for the year 21](#_Toc440654330)

[2c) The month and day that recorded the maximum precipitation during the year 21](#_Toc440654331)

[2d) The average yearly precipitation 22](#_Toc440654332)

[Tutorial 4 23](#_Toc440654333)

[1. Compute A\*B 23](#_Toc440654334)

[2. Compute the dot product 23](#_Toc440654335)

[3. Polynomial Algebra 23](#_Toc440654336)

[3a) Find f (x) + g(x) 23](#_Toc440654337)

[3b) Find f (x) \* g(x) 23](#_Toc440654338)

[3c) Find f (x) / g(x) 23](#_Toc440654339)

[4. Plotting Polynomials 24](#_Toc440654340)

[4a) Plot 24](#_Toc440654341)

[4b) Plot 24](#_Toc440654342)

[5. Obtaining the roots 25](#_Toc440654343)

[6. Confirm the equation 25](#_Toc440654344)

[7. Confirm the equation 25](#_Toc440654345)

[8. Using the following matrix 26](#_Toc440654346)

[8a) Create a vector V consisting of the elements in the second column of A 26](#_Toc440654347)

[8b) Create a vector W consisting of the elements in the second row of A 26](#_Toc440654348)

[9. Using the following matrix 26](#_Toc440654349)

[9a) Find the maximum and minimum values in each column 26](#_Toc440654350)

[9b) Find the maximum and minimum values in each row 26](#_Toc440654351)

[10. Plot a linear function from 0 to 100 on both X and Y axis 27](#_Toc440654352)

[11. Experimenting with plot features. 27](#_Toc440654353)

[12. Create a plot of complex numbers. 28](#_Toc440654354)

[14. Plot polynomials stored in coefficient vector format. 28](#_Toc440654355)

[15. Create your own subplots and overlay plots with legend(s) 29](#_Toc440654356)

[16. Create a cell array and place various different classes of data into it 31](#_Toc440654357)

[17. Briefly describe the concept of code vectorisation and vectorise the following strand of code 31](#_Toc440654358)

[I. Exploratory data analysis 32](#_Toc440654359)

[1. Examine the dataset PIMA 32](#_Toc440654360)

[1.1 Introduction 32](#_Toc440654361)

[1.2 Attribute Ranges 32](#_Toc440654362)

[1.3 Correlations 32](#_Toc440654363)

[1.4 Importance to Prediction 34](#_Toc440654364)

[1.5 Histogram Analysis 34](#_Toc440654365)

[1.6 Scatter Plot Analysis 35](#_Toc440654366)

[II. Data pre-processing 37](#_Toc440654367)

[2. Pre-processing Tasks 37](#_Toc440654368)

[2.1 Introduction 37](#_Toc440654369)

[2.2 Normalization 37](#_Toc440654370)

[2.3 Discretization 38](#_Toc440654371)

[III. Data set splitting 39](#_Toc440654372)

[3 Data Splitting Techniques 39](#_Toc440654373)

[3.1 Introduction 39](#_Toc440654374)

[3.2 Class Based Data Subsets 39](#_Toc440654375)

[3.3 Function Based Data Subsets 39](#_Toc440654376)

[IV. Prediction of the Target Variable using an Artificial Neural Network 41](#_Toc440654377)

[Datasets 41](#_Toc440654378)

[Performance 41](#_Toc440654379)

[Error Histogram 41](#_Toc440654380)

[Regression 42](#_Toc440654381)

[References 43](#_Toc440654382)

[Appendix 44](#_Toc440654383)

[Histograms 44](#_Toc440654384)

[C1 C2 44](#_Toc440654385)

[C4 C5 44](#_Toc440654386)

[C7 C8 44](#_Toc440654387)

[C9 45](#_Toc440654388)

[Scatter Plots 45](#_Toc440654389)

[C1 C2 45](#_Toc440654390)

[C3 C4 45](#_Toc440654391)

[C5 C6 46](#_Toc440654392)

[C7 46](#_Toc440654393)

# Tables

[Table 1 Create a square Matrix 6](#_Toc440654232)

[Table 2 Create a matrix using functions 6](#_Toc440654233)

[Table 3 Subtracting Matrices 6](#_Toc440654234)

[Table 4 Multiplying Matrices 7](#_Toc440654235)

[Table 5 Raising a matrix by a power 7](#_Toc440654236)

[Table 6 Matrix Expression 7](#_Toc440654237)

[Table 7 Plotting a Graph 7](#_Toc440654238)

[Table 8 Experiment Plotting 1 8](#_Toc440654239)

[Table 9 Experiment Plotting 2 8](#_Toc440654240)

[Table 10 Linear Line Plot 9](#_Toc440654241)

[Table 11 3D Plots 1 9](#_Toc440654242)

[Table 12 3D Plots 2 10](#_Toc440654243)

[Table 13 Simple Plot 10](#_Toc440654244)

[Table 14 Plot Tools 1 11](#_Toc440654245)

[Table 15 Plot Tools 2 11](#_Toc440654246)

[Table 16 Time vs. Temperature 15](#_Toc440654247)

[Table 17 Average of Time vs. Temperature 15](#_Toc440654248)

[Table 18 Solving X 16](#_Toc440654249)

[Table 19 Time vs. Distance 16](#_Toc440654250)

[Table 20 E = 17](#_Toc440654251)

[Table 21 Temperature and Pressure 1 18](#_Toc440654252)

[Table 22 P = 1.0e+04 \* Columns 1 to 181 19](#_Toc440654253)

[Table 23 Temperature and Pressure 2 20](#_Toc440654254)

[Table 24 Temperature and Pressure 3 20](#_Toc440654255)

[Table 25 Alter Dataset 21](#_Toc440654256)

[Table 26 Total Monthly Precipitation 21](#_Toc440654257)

[Table 27 Total Yearly Precipitation 21](#_Toc440654258)

[Table 28 Day and Month of Max Precipitation 21](#_Toc440654259)

[Table 29 Total Average Yearly Precipitation 22](#_Toc440654260)

[Table 30 Compute A\*B 23](#_Toc440654261)

[Table 31 Compute Dot Product 23](#_Toc440654262)

[Table 32 Polynomial Algebra 1 23](#_Toc440654263)

[Table 33 Polynomial Algebra 2 23](#_Toc440654264)

[Table 34 Polynomial Algebra 3 23](#_Toc440654265)

[Table 35 Plotting Polynomials 1 24](#_Toc440654266)

[Table 36 Plotting Polynomials 2 24](#_Toc440654267)

[Table 37 Obtaining Root Values 25](#_Toc440654268)

[Table 38 Polynomial Equations 25](#_Toc440654269)

[Table 39 Polynomial Equations 25](#_Toc440654270)

[Table 40 Creating Vectors 1 26](#_Toc440654271)

[Table 41 Creating Vectors 2 26](#_Toc440654272)

[Table 42 Finding ranges in Matrices 1 26](#_Toc440654273)

[Table 43 Finding ranges in Matrices 2 26](#_Toc440654274)

[Table 44 Plotting Linear Function 27](#_Toc440654275)

[Table 45 Plotting Experimentation 1 27](#_Toc440654276)

[Table 46 Plotting Experimentation 2 28](#_Toc440654277)

[Table 47 Complex Plots 28](#_Toc440654278)

[Table 48 Polynomial Plots 28](#_Toc440654279)

[Table 49 Exploring Subplots 1 29](#_Toc440654280)

[Table 50 Exploring Subplots 2 29](#_Toc440654281)

[Table 51 Exploring Subplots 3 30](#_Toc440654282)

[Table 52 Cell Arrays 31](#_Toc440654283)

[Table 53 Non-Vectorised Code 31](#_Toc440654284)

[Table 54 Vectorised Code 31](#_Toc440654285)

[Table 55 Range Psuedocode 32](#_Toc440654286)

[Table 56 Range Results 32](#_Toc440654287)

[Table 57 Correlations of Attributes 33](#_Toc440654288)

[Table 58 Sample Data Psuedocode 33](#_Toc440654289)

[Table 59 Sample Data Results 33](#_Toc440654290)

[Table 60 Attribute Pair Correlation 34](#_Toc440654291)

[Table 61 Number of Values in Bin of c3 35](#_Toc440654292)

[Table 62 Normalisation Program 37](#_Toc440654293)

[Table 63 Original and Normalised Values 37](#_Toc440654294)

[Table 64 Discretize Program 38](#_Toc440654295)

[Table 65 Original, Discretized Values and Bin Medians 38](#_Toc440654296)

[Table 66 All rows with Class Label 0 39](#_Toc440654297)

[Table 67 Average of Class Label 0 and 1 39](#_Toc440654298)

[Table 68 Std of Class Label 0 and 1 39](#_Toc440654299)

[Table 69 Loop Run and Save for divideset1 40](#_Toc440654300)

# Figures

[Figure 1 Plotting a Graph 8](#_Toc440654198)

[Figure 2 Experimenting with Graphs 1 8](#_Toc440654199)

[Figure 3 Experimenting with Graphs 2 9](#_Toc440654200)

[Figure 4 Linear Graph 9](#_Toc440654201)

[Figure 5 3D Plot 1 10](#_Toc440654202)

[Figure 6 3D Plot 2 10](#_Toc440654203)

[Figure 7 Simple Plot 11](#_Toc440654204)

[Figure 8 Exploring Plot Tools 1 11](#_Toc440654205)

[Figure 9 Exploring Plot Tools 2 12](#_Toc440654206)

[Figure 10 Exploring Plot Tools 3 12](#_Toc440654207)

[Figure 11 Exploring Plot Tools 4 12](#_Toc440654208)

[Figure 12 Exploring Plot Tools 5 13](#_Toc440654209)

[Figure 13 Exploring Plot Tools 6 13](#_Toc440654210)

[Figure 14 Exploring Plot Tools 7 13](#_Toc440654211)

[Figure 15 Importing Data 14](#_Toc440654212)

[Figure 16 Time vs. Temperature 15](#_Toc440654213)

[Figure 17 Equation Script 17](#_Toc440654214)

[Figure 18 Plotting Polynomials 1 24](#_Toc440654215)

[Figure 19 Plotting Polynomials 2 24](#_Toc440654216)

[Figure 20 Plotting a Linear Function 27](#_Toc440654217)

[Figure 21 Experimenting with Plot Features 1 27](#_Toc440654218)

[Figure 22 Experimenting with Plot Features 2 28](#_Toc440654219)

[Figure 23 Complex Numbers Plot 28](#_Toc440654220)

[Figure 24 Coefficient Vector Polynomial 29](#_Toc440654221)

[Figure 25 Experimenting with Subplots 1 29](#_Toc440654222)

[Figure 26 Experimenting with Subplots 2 30](#_Toc440654223)

[Figure 27 Experimenting with Subplots 3 30](#_Toc440654224)

[Figure 28 histogram\_analysis2 34](#_Toc440654225)

[Figure 29 Histogram of Attribute c3 Figure 30 Histogram of Attribute c6 35](#_Toc440654226)

[Figure 31 Attributes 1 vs. 2:8 36](#_Toc440654227)

[Figure 32 Scatter Plot of Attributes 2 and 5 Figure 33 Scatter Plot of Attributes 4 and 6 36](#_Toc440654228)

[Figure 34 Performance Plot 41](#_Toc440654229)

[Figure 35 Error Histogram 42](#_Toc440654230)

[Figure 36 Regression Plot 42](#_Toc440654231)

# Tutorial 1

## 1. Create a square matric by directly inserting values

A square matrix can be created by directly inserting values, splitting them into rows and column. A square matrix means there are equal rows and columns forming a square shape.

Table 1 Create a square Matrix

|  |
| --- |
| A1 = [129 14; 67 98]  A1 =  129 14  67 98 |

## 2. Create a matrix using functions

Matrices can be made using various functions such as directly with or without rows or by using functions such as zeros() in which you specify the rows and columns.

Table 2 Create a matrix using functions

|  |
| --- |
| a = [1 2 3 4] a = 1 2 3 4  b = [1 2 3; 4 5 6; 7 8 10] b =  1 2 3  4 5 6  7 8 10  z = zeros(5,1) z =  0  0  0  0  0 |

## 3. Subtracting Matrices

Matlab is able to compute mathematical equations very easily for example subtracting one matrix ‘A’ from another ‘B’ to form a third matrix ‘C’.

Table 3 Subtracting Matrices

|  |
| --- |
| A = [3 3 3] A = 3 3 3  B = [2 2 2] B = 2 2 2  C = A – B C = 1 1 1 |

## 4. Multiplying Matrices

Another is multiplication of a matrix by a scalar number where the elements are calculated individually.

Table 4 Multiplying Matrices

|  |
| --- |
| D = [3 3 3]  D = 3 3 3  E = D \* 2  E = 6 6 6 |

## 5. Raising a matrix by a power

Matrices can also be multiplied by a power; again the elements are calculated individually.

Table 5 Raising a matrix by a power

|  |
| --- |
| F = [2, 2; 2, 2]  F =  2 2  2 2  G = F^3  G =  32 32  32 32 |

## 6. Matrix Expression

If we wanted to compute the previously learnt skills at the same time, for example C = (5AB2 – 3BT)2 T represents ‘transpose’ meaning the matrix changes from one dimension to another e.g. a row becomes a column.

Table 6 Matrix Expression

|  |
| --- |
| H = [1 2 0; 2 1 2; 0 2 1];  I = [3 0 3; 1 5 1; 1 1 3];  J = ((5\*H\*I^2) - (3\*I'))^2  J =  106653 158061 158943  128002 222059 187837  105407 147679 157922 |

## 7. Plotting graphs

Matlab can easily plot graphs using the plot() function.

Table 7 Plotting a Graph

|  |
| --- |
| X = -10:0.1:10;  plot(sin(X)); |

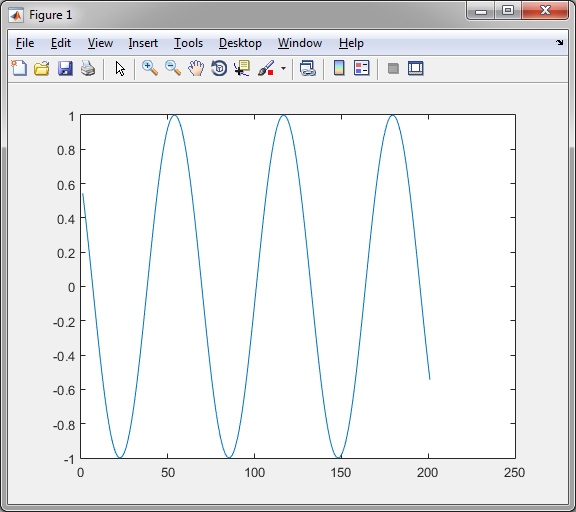


Figure 1 Plotting a Graph

## 8. Experimenting with different plots

Depending in the data that is put into the plot() function multiple lines can be within one graph.

Table 8 Experiment Plotting 1

|  |
| --- |
| x = -10:0.1:10;  plot( x, cos(x), x, 1 - x.^2./2, x, 1 - x.^2./2 + x.^4./24 ) |

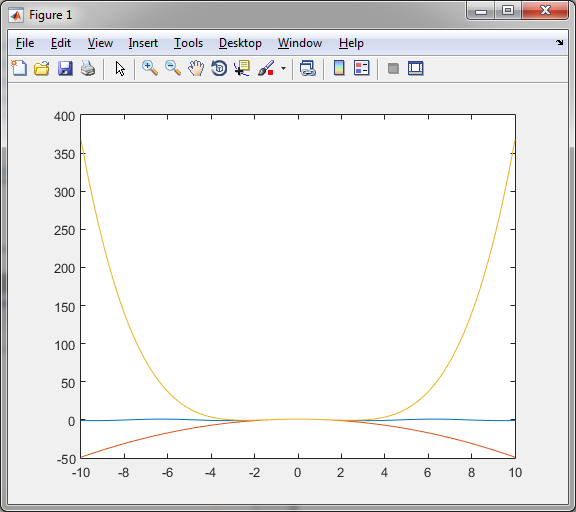


Figure 2 Experimenting with Graphs 1

A similar command is polar() function which creates a graph on a 360 degree plane rather than an XY graph. ‘Hold on’ allows multiple functions to be run at the same time.

Table 9 Experiment Plotting 2

|  |
| --- |
| x=[0:0.01:2.5];  polar(x,cos(x)), hold on, polar(x,sin(x)) |

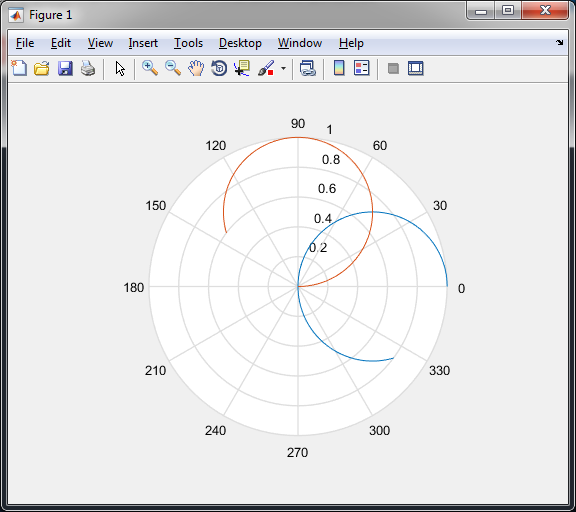


Figure 3 Experimenting with Graphs 2

We can also plot a linear line where as one value increase so does the other.

Table 10 Linear Line Plot

|  |
| --- |
| X1 = [1 2 3 4 5 6 7 8 9 10];  Y1 = [1 2 3 4 5 6 7 8 9 10];  plot(X1,Y1) |

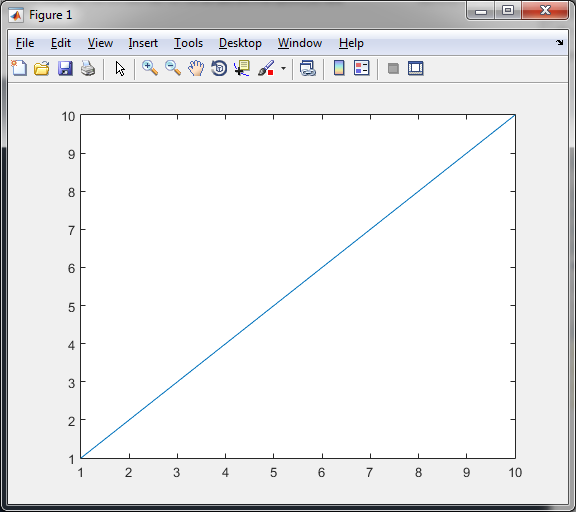


Figure 4 Linear Graph

## 9. 3 Dimensional Plots

When creating a 3D plot a few function can be used such as scatter3() which results in Figure 5 or Plot3() which results in Figure 6.

Table 11 3D Plots 1

|  |
| --- |
| [A,B,C] = sphere(20);  a = [0.75\*A(:); 0.5\*A(:); 0.3\*A(:)];  b = [3.2\*B(:); 5\*B(:); 0.43\*B(:)];  z = [0.25\*C(:); 0.15\*C(:); 0.5\*C(:)]; scatter3(a,b,c) |

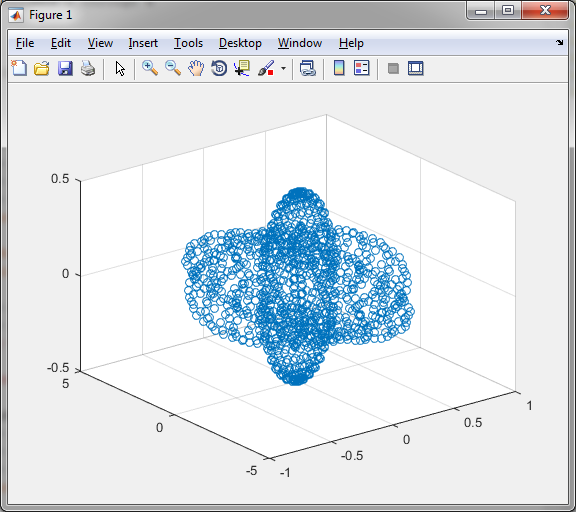


Figure 5 3D Plot 1

Table 12 3D Plots 2

|  |
| --- |
| plot3(a,b,c) |

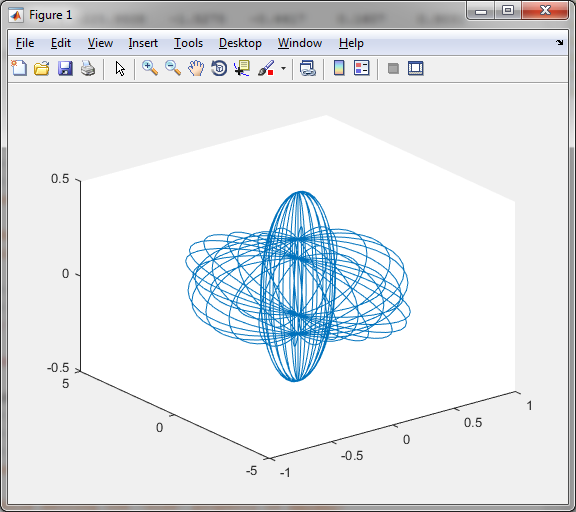


Figure 6 3D Plot 2

## 10. Simple XY Plot

Depending on the data given a simple plot can appear different Figure7 shows a simple plot one dimensional plot with data ranging from 1 to 100 in intervals of 10.

Table 13 Simple Plot

|  |
| --- |
| X = (1:10:100);  Y = (1:10:100);  plot(X,Y) |

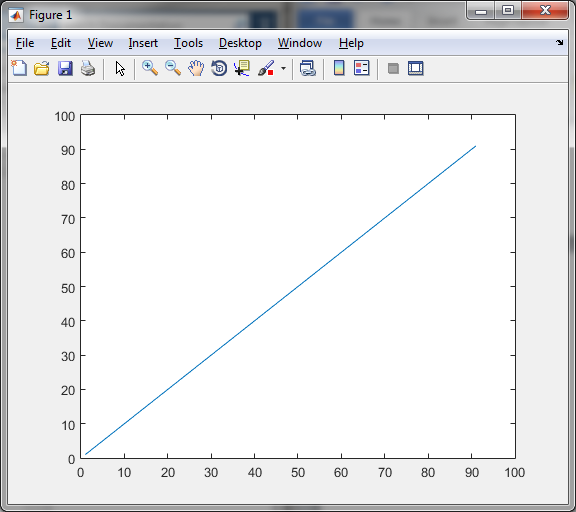


Figure 7 Simple Plot

## 11. Exploring Plot Tools

Plot Tools allows for an interface that lets the user dynamically create and alter graphs without the need to use code. For example if the user creates a plot (Figure 8) and runs ‘plottools’ Figure 9 will appear allow for user changes such as adding a subplot (Figure 10), add in a scatter graph for that subplot (Figure 11), add additional graphs into an existing subplot (Figure 12 and Figure 13) and finally changing the colours, text or adding a legend (Figure 14)

Table 14 Plot Tools 1

|  |
| --- |
| a = linspace(5,50,500);  b = tan(a);  plot(a,b) |

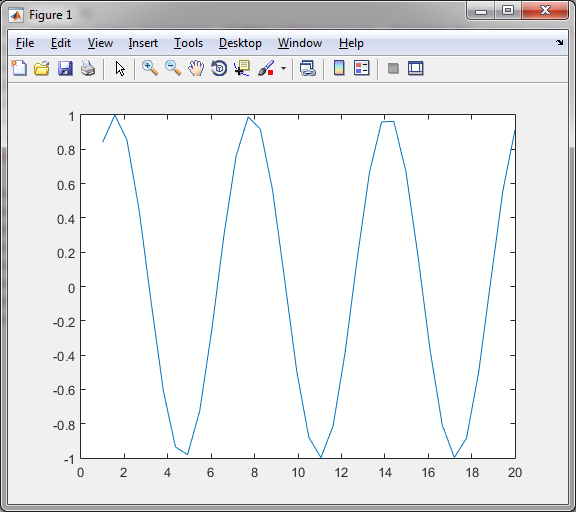


Figure 8 Exploring Plot Tools 1

Table 15 Plot Tools 2

|  |
| --- |
| plottools |

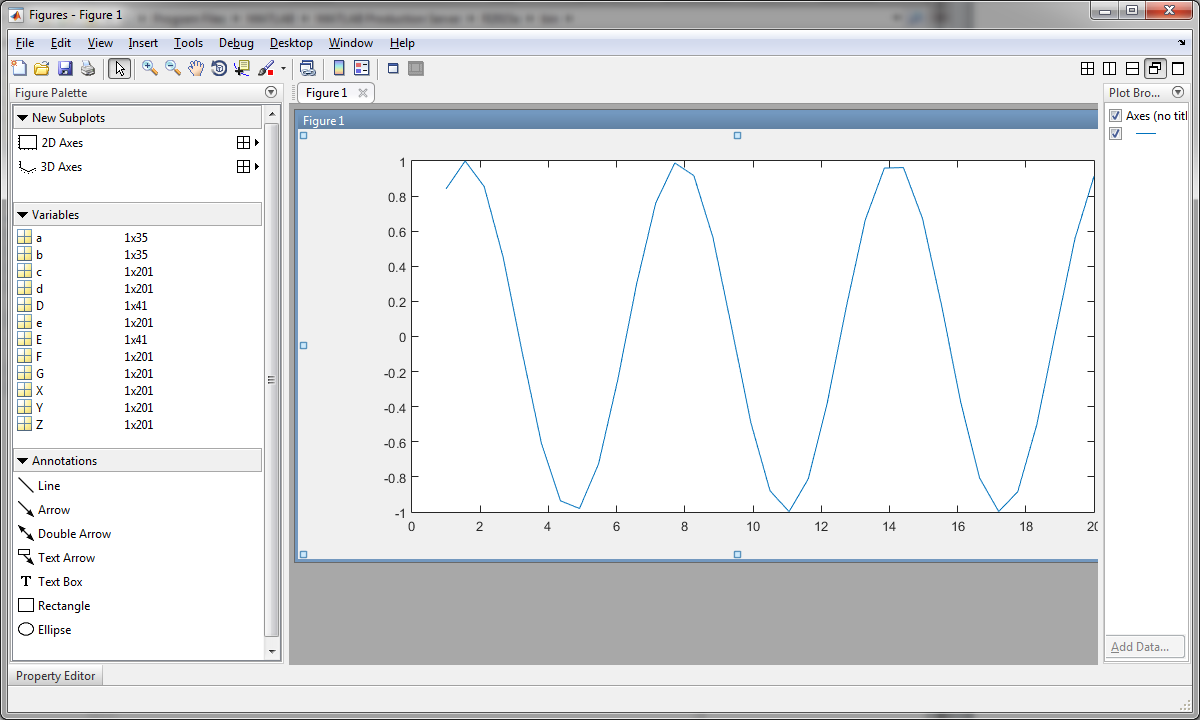


Figure 9 Exploring Plot Tools 2

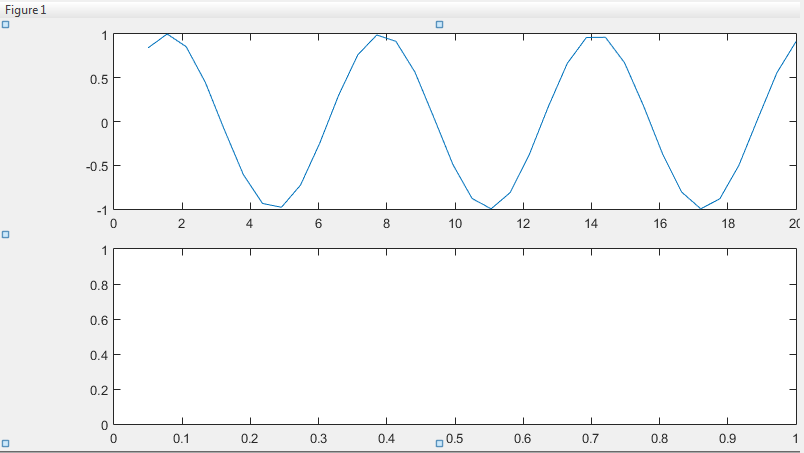


Figure 10 Exploring Plot Tools 3

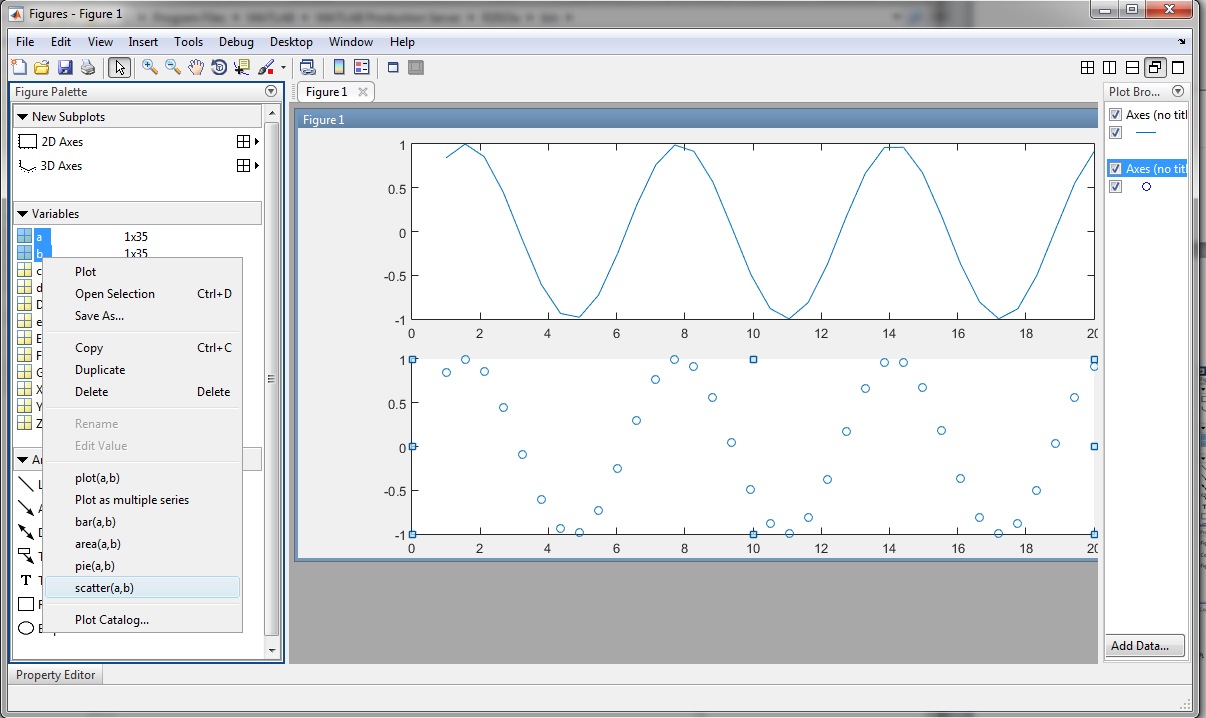


Figure 11 Exploring Plot Tools 4

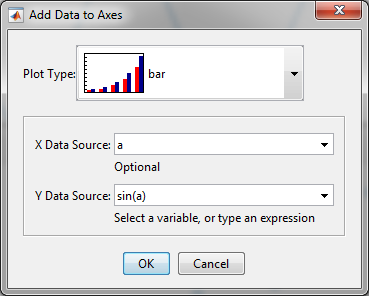


Figure 12 Exploring Plot Tools 5

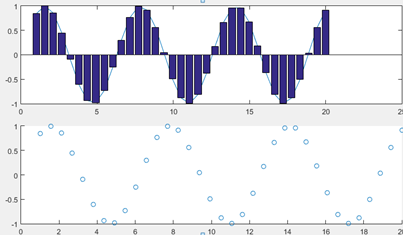


Figure 13 Exploring Plot Tools 6

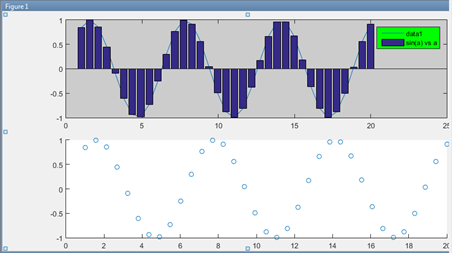


Figure 14 Exploring Plot Tools 7

## 12. Importing Data

Create a 10 rows and 5 columns table in Excel and import that data into Matlab using the ‘uiimport’ command, and either import the data via clipboard or as shown in Figure 15 via file. Once imported it allows for the new data to have names assigned.

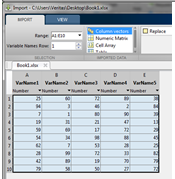
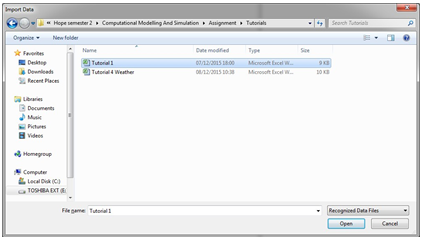


Figure 15 Importing Data

# Tutorial 2

## 1. Using the given dataset

### 1a) Plot a graph Time against Temperature and Title

Using the data from the given table of ‘Time’ and ‘Temperature’ plot a simple graph (Figure 16).

Table 16 Time vs. Temperature

|  |
| --- |
| plot(X,Y),xlabel('Time, minutes'),ylabel('Temperature, degrees F'),title('Temperature measurements'); |

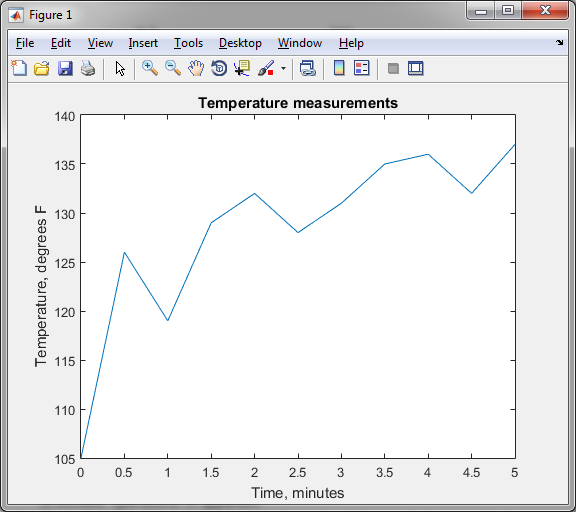


Figure 16 Time vs. Temperature

### 1b) Find the average temperature

The average temperature is shown in Table 17.

Table 17 Average of Time vs. Temperature

|  |
| --- |
| Z = mean(Y)  Z = 128.1818 |

## 2. Consider the equation. For different value of x, solve.

Table 18 shows the equation in matlab with different values of ‘x’ resulting in different values of ‘f’.

Table 18 Solving X

|  |
| --- |
| x=1.221;  f = (((x^3)-(2\*(x^2))+x-6.3)/((x^2)+(0.05005\*x)-3.14))  f = 3.9296  x = 4.674  f = (((x^3)-(2\*(x^2))+x-6.3)/((x^2)+(0.05005\*x)-3.14))  f = 2.9984  x = -3.4321  f = (((x^3)-(2\*(x^2))+x-6.3)/((x^2)+(0.05005\*x)-3.14))  f = -8.7060 |

## 3. Consider the equation.

3a) Assuming g = 9.8m/s2, create a table of time vs. distance

Table 19 shows the equation created in matlab and table 20 shows a transposed version of the data time vs. distance.

Table 19 Time vs. Distance

|  |
| --- |
| t = 0:10  t = 0 1 2 3 4 5 6 7 8 9 10  g = 9.8  g = 9.8000  d=0.5\*g\*t.^2  d = Columns 1 through 7  0 4.9000 19.6000 44.1000 78.4000 122.5000 176.4000  Columns 8 through 11  240.1000 313.6000 396.9000 490.0000  E = [t',d'] |

Table 20 E =

|  |  |
| --- | --- |
| Column 1 | Column 2 |
| 0.0000 | 0.0000 |
| 1.0000 | 4.9000 |
| 2.0000 | 19.6000 |
| 3.0000 | 44.1000 |
| 4.0000 | 78.4000 |
| 5.0000 | 122.5000 |
| 6.0000 | 176.4000 |
| 7.0000 | 240.1000 |
| 8.0000 | 313.6000 |
| 9.0000 | 396.9000 |
| 10.0000 | 490.0000 |

### 3b) Create a script of the equation

A script can be easily made as shown in Figure 17 that when run outputs the same results.

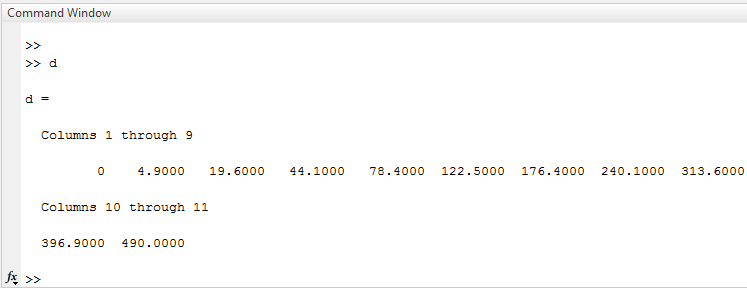
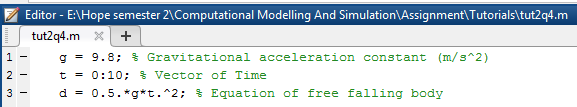


Figure 17 Equation Script

# Tutorial 3

## 1. Temperature and Pressure

1a) Taking into account that the temperature on the surface of the earth is never lower than -60oF or higher than 120oF, use the equation to find the saturation vapour pressure for temperatures in this range.

Table 21 Temperature and Pressure 1

|  |
| --- |
| T = -60:120;  Rair = 461;  H = (2.453\*10^6);  P = ((H/Rair)\*((1/273)-(1./T)))\*6.11 |

By using the code in Table 21 all the possible results have been presented in Table 22 and colour coded for clear and easy viewing.

Table 22 P = 1.0e+04 \* Columns 1 to 181

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Column 1 | Column 2 | Column 3 | Column 4 | Column 5 | Column 6 | Column 7 | Column 8 | Column 9 |
| 0.6609 | 0.6701 | 0.6796 | 0.6895 | 0.6997 | 0.7102 | 0.7212 | 0.7325 | 0.7443 |
| Column 10 | Column 11 | Column 12 | Column 13 | Column 14 | Column 15 | Column 16 | Column 17 | Column 18 |
| 0.7566 | 0.7693 | 0.7826 | 0.7964 | 0.8108 | 0.8259 | 0.8416 | 0.8580 | 0.8752 |
| Column 19 | Column 20 | Column 21 | Column 22 | Column 23 | Column 24 | Column 25 | Column 26 | Column 27 |
| 0.8932 | 0.9121 | 0.9319 | 0.9527 | 0.9747 | 0.9978 | 1.0222 | 1.0480 | 1.0753 |
| Column 28 | Column 29 | Column 30 | Column 31 | Column 32 | Column 33 | Column 34 | Column 35 | Column 36 |
| 1.1043 | 1.1351 | 1.1679 | 1.2028 | 1.2402 | 1.2802 | 1.3232 | 1.3695 | 1.4196 |
| Column 37 | Column 38 | Column 39 | Column 40 | Column 41 | Column 42 | Column 43 | Column 44 | Column 45 |
| 1.4737 | 1.5326 | 1.5969 | 1.6673 | 1.7447 | 1.8302 | 1.9253 | 2.0315 | 2.1511 |
| Column 46 | Column 47 | Column 48 | Column 49 | Column 50 | Column 51 | Column 52 | Column 53 | Column 54 |
| 2.2865 | 2.4413 | 2.6200 | 2.8284 | 3.0747 | 3.3702 | 3.7315 | 4.1830 | 4.7636 |
| Column 55 | Column 56 | Column 57 | Column 58 | Column 59 | Column 60 | Column 61 | Column 62 | Column 63 |
| 5.5377 | 6.6214 | 8.2470 | 10.9563 | 16.3749 | 32.6307 | #NAME? | -32.3925 | -16.1367 |
| Column 64 | Column 65 | Column 66 | Column 67 | Column 68 | Column 69 | Column 70 | Column 71 | Column 72 |
| -10.7181 | -8.0088 | -6.3832 | -5.2995 | -4.5254 | -3.9449 | -3.4933 | -3.1321 | -2.8365 |
| Column 73 | Column 74 | Column 75 | Column 76 | Column 77 | Column 78 | Column 79 | Column 80 | Column 81 |
| -2.5902 | -2.3818 | -2.2032 | -2.0483 | -1.9129 | -1.7934 | -1.6871 | -1.5920 | -1.5065 |
| Column 82 | Column 83 | Column 84 | Column 85 | Column 86 | Column 87 | Column 88 | Column 89 | Column 90 |
| -1.4291 | -1.3587 | -1.2945 | -1.2356 | -1.1814 | -1.1314 | -1.0850 | -1.0420 | -1.0020 |
| Column 91 | Column 92 | Column 93 | Column 94 | Column 95 | Column 96 | Column 97 | Column 98 | Column 99 |
| -0.9646 | -0.9297 | -0.8969 | -0.8661 | -0.8371 | -0.8098 | -0.7840 | -0.7596 | -0.7365 |
| Column 100 | Column 101 | Column 102 | Column 103 | Column 104 | Column 105 | Column 106 | Column 107 | Column 108 |
| -0.7145 | -0.6937 | -0.6739 | -0.6550 | -0.6370 | -0.6198 | -0.6034 | -0.5877 | -0.5726 |
| Column 109 | Column 110 | Column 111 | Column 112 | Column 113 | Column 114 | Column 115 | Column 116 | Column 117 |
| -0.5582 | -0.5444 | -0.5311 | -0.5184 | -0.5061 | -0.4943 | -0.4830 | -0.4720 | -0.4615 |
| Column 118 | Column 119 | Column 120 | Column 121 | Column 122 | Column 123 | Column 124 | Column 125 | Column 126 |
| -0.4513 | -0.4415 | -0.4320 | -0.4228 | -0.4139 | -0.4053 | -0.3970 | -0.3889 | -0.3811 |
| Column 127 | Column 128 | Column 129 | Column 130 | Column 131 | Column 132 | Column 133 | Column 134 | Column 135 |
| -0.3735 | -0.3662 | -0.3590 | -0.3521 | -0.3454 | -0.3388 | -0.3325 | -0.3263 | -0.3203 |
| Column 136 | Column 137 | Column 138 | Column 139 | Column 140 | Column 141 | Column 142 | Column 143 | Column 144 |
| -0.3144 | -0.3087 | -0.3031 | -0.2977 | -0.2924 | -0.2873 | -0.2823 | -0.2774 | -0.2726 |
| Column 145 | Column 146 | Column 147 | Column 148 | Column 149 | Column 150 | Column 151 | Column 152 | Column 153 |
| -0.2680 | -0.2634 | -0.2590 | -0.2546 | -0.2504 | -0.2462 | -0.2421 | -0.2382 | -0.2343 |
| Column 154 | Column 155 | Column 156 | Column 157 | Column 158 | Column 159 | Column 160 | Column 161 | Column 162 |
| -0.2305 | -0.2268 | -0.2231 | -0.2196 | -0.2161 | -0.2127 | -0.2093 | -0.2060 | -0.2028 |
| Column 163 | Column 164 | Column 165 | Column 166 | Column 167 | Column 168 | Column 169 | Column 170 | Column 171 |
| -0.1997 | -0.1966 | -0.1935 | -0.1905 | -0.1876 | -0.1848 | -0.1819 | -0.1792 | -0.1765 |
| Column 172 | Column 173 | Column 174 | Column 175 | Column 176 | Column 177 | Column 178 | Column 179 | Column 180 |
| -0.1738 | -0.1712 | -0.1686 | -0.1661 | -0.1636 | -0.1612 | -0.1588 | -0.1564 | -0.1541 |
| Column 181 |  | | | | | | | |
| -0.1518 |

1b) Present these results as a table of temperature in oF and saturation vapour pressure .

Table 23 Temperature and Pressure 2

|  |
| --- |
| F = T’;  Pr = P’;  R = table(F,Pr);  R = Table 18 |

Table 24 Temperature and Pressure 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| oF  oF   oF  oF   oF | | | | |
| -60 660.95  -59 670.13 | -23 1532.6  -22 1596.9 | 13 -2381.8  14 -2203.2 | 49 -544.41  50 -531.14 | 85 -263.4  86 -258.95 |
| -58 679.63  -57 689.47  -56 699.65  -55 710.21  -54 721.16  -53 732.52  -52 744.31  -51 756.57  -50 769.32  -49 782.59  -48 796.41  -47 810.83  -46 825.86  -45 841.57  -44 857.99  -43 875.17  -42 893.17  -41 912.05  -40 931.88  -39 952.72  -38 974.66  -37 997.78  -36 1022.2  -35 1048  -34 1075.3  -33 1104.3  -32 1135.1  -31 1167.9  -30 1202.8  -29 1240.2  -28 1280.2  -27 1323.2  -26 1369.5  -25 1419.6  -24 1473.7 | -21 1667.3  -20 1744.7  -19 1830.2  -18 1925.3  -17 2031.5  -16 2151.1  -15 2286.5  -14 2441.3  -13 2620  -12 2828.4  -11 3074.7  -10 3370.2  -9 3731.5  -8 4183  -7 4763.6  -6 5537.7  -5 6621.4  -4 8247  -3 10956  -2 16375  -1 32631  0 -Inf  1 -32392  2 -16137  3 -10718  4 -8008.8  5 -6383.2  6 -5299.5  7 -4525.4  8 -3944.9  9 -3493.3  10 -3132.1  11 -2836.5  12 -2590.2 | 15 -2048.3  16 -1912.9  17 -1793.4  18 -1687.1  19 -1592  20 -1506.5  21 -1429.1  22 -1358.7  23 -1294.5  24 -1235.6  25 -1181.4  26 -1131.4  27 -1085  28 -1042  29 -1002  30 -964.63  31 -929.67  32 -896.9  33 -866.11  34 -837.13  35 -809.81  36 -784.01  37 -759.6  38 -736.48  39 -714.54  40 -693.7  41 -673.87  42 -654.99  43 -636.99  44 -619.81  45 -603.39  46 -587.68  47 -572.65  48 -558.23 | 51 -518.39  52 -506.13  53 -494.34  54 -482.98  55 -472.03  56 -461.47  57 -451.29  58 -441.45  59 -431.95  60 -422.77  61 -413.89  62 -405.29  63 -396.97  64 -388.9  65 -381.09  66 -373.51  67 -366.16  68 -359.02  69 -352.09  70 -345.36  71 -338.82  72 -332.46  73 -326.27  74 -320.26  75 -314.4  76 -308.69  77 -303.14  78 -297.72  79 -292.45  80 -287.3  81 -282.29  82 -277.39  83 -272.62  84 -267.95 | 87 -254.61  88 -250.36  89 -246.21  90 -242.15  91 -238.18  92 -234.3  93 -230.5  94 -226.78  95 -223.14  96 -219.57  97 -216.08  98 -212.66  99 -209.31  100 -206.03  101 -202.81  102 -199.65  103 -196.56  104 -193.52  105 -190.54  106 -187.62  107 -184.76  108 -181.94  109 -179.18  110 -176.47  111 -173.81  112 -171.19  113 -168.62  114 -166.1  115 -163.62  116 -161.18  117 -158.79  118 -156.43  119 -154.12  120 -151.84 |

## 2. National oceanic and atmospheric administration weather data.

As the data (weather\_data.xls) had a few values of -99999 possible due to being incorrect or missing data, the dataset was altered so that any value exceeding -100 would become 0 (Table 25). With this alteration the following question were completed.

Table 25 Alter Dataset

|  |
| --- |
| A(A<-100)=0; |

### 2a) Total precipitation in each month

When the dataset is placed into a matrix the months are represented by the rows of the matrix and the days by the columns therefore the total monthly precipitation can be acquired via the code in Table 26.

Table 26 Total Monthly Precipitation

|  |
| --- |
| S=sum(A,2)  S =  978  645  193  579  180  365  74  207  622  3  596  54 |

### 2b) Total precipitation for the year

Total yearly precipitation can be acquired by Table 27

Table 27 Total Yearly Precipitation

|  |
| --- |
| B=sum(A);  SB=sum(B,2)  SY = 4496 |

### 2c) The month and day that recorded the maximum precipitation during the year

The month and day can be found by finding the max of the data and outputting the index it is at then using those values to find the row and column for the month and day.

Table 28 Day and Month of Max Precipitation

|  |
| --- |
| C = A(:);  [M,I]=max(C(:))  M = 272  I = 25  [I\_row, I\_col] = ind2sub(size(C),I)  I\_row = 25  I\_col = 1 |

### 2d) The average yearly precipitation

The average has been calculated using the average of each month for the year into a column and then finding the average of total monthly precipitation (Table 29).

Table 29 Total Average Yearly Precipitation

|  |
| --- |
| Av=mean(A,2)  Av =  31.5484  20.8065  6.2258  18.6774  5.8065  11.7742  2.3871  6.6774  20.0645  0.0968  19.2258  1.7419  Av2=mean(Av)  Av2 = 12.0860 |

# Tutorial 4

## 1. Compute A\*B

Table 30 Compute A\*B

|  |
| --- |
| A = [6,-2; 10,3; 4,7];  B = [9,8; -5,12 ];  C = A \* B  C = 64 24  75 116  1 116 |

2. Compute the dot product

Show using the dot() function that u = 6i- 8j+3k and w = 5i+3j-4k Answer = -6

Table 31 Compute Dot Product

|  |
| --- |
| u=[6;-8;3];  w=[5;3;-4];  V=dot(u,w)  V =  -6 |

## 3. Polynomial Algebra

Consider the following equations *f* (*x*)  9*x*3  5*x*2  3*x*  7 and *g*(*x*)  6*x*2  *x*  2

3a) Find f (x) + g(x)

Table 32 Polynomial Algebra 1

|  |
| --- |
| f = [9 -5 3 7];  g = [0 6 -1 2];  h = f+g  h = 9 1 2 9 |

3b) Find f (x) \* g(x)

Using conv() outputs the value of multiplication.

Table 33 Polynomial Algebra 2

|  |
| --- |
| i = conv(f,g)  i = 0 54 -39 41 29 -1 14 |

3c) Find f (x) / g(x)

Using deconv() outputs the division, to acquire the quotation and remainder out into two matrices.

Table 34 Polynomial Algebra 3

|  |
| --- |
| j = deconv(f,g(2:end))  j = 1.5000 -0.5833  [quot rem] = deconv(f,g(2:end))  quot = 1.5000 -0.5833  rem = 0 0 -0.5833 8.1667 |

## 4. Plotting Polynomials

4a) Plot *f* (*x*)  9*x*3  5*x*2  3*x*  7 at *x* = 0,2,4,…10

Table 35 Plotting Polynomials 1

|  |
| --- |
| x = 0:2:10  x = 0 2 4 6 8 10  f = ((9\*x.^3)-(5\*x.^2)+(3\*x)+7)  f = 7 65 515 1789 4319 8537  plot(f) |

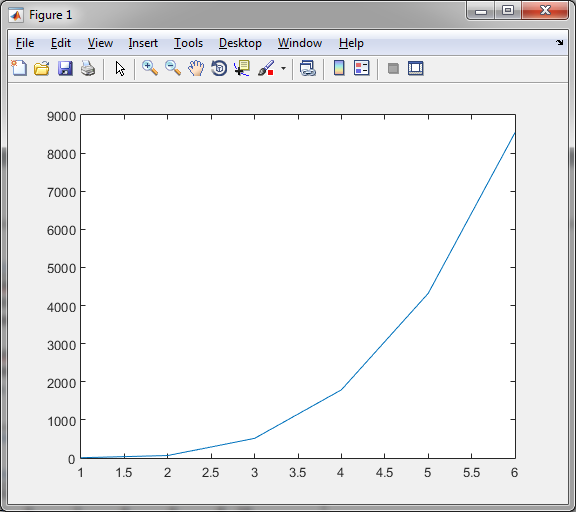


Figure 18 Plotting Polynomials 1

4b) Plot f (x)  9x3  5x2  3x  7 for  2  x  5 with the interval increasing by 0.01x = -2:0.01:5;

Table 36 Plotting Polynomials 2

|  |
| --- |
| x = -2:0.01:5;  f = ((9\*x.^3)-(5\*x.^2)+(3\*x)+7);  plot(f) |

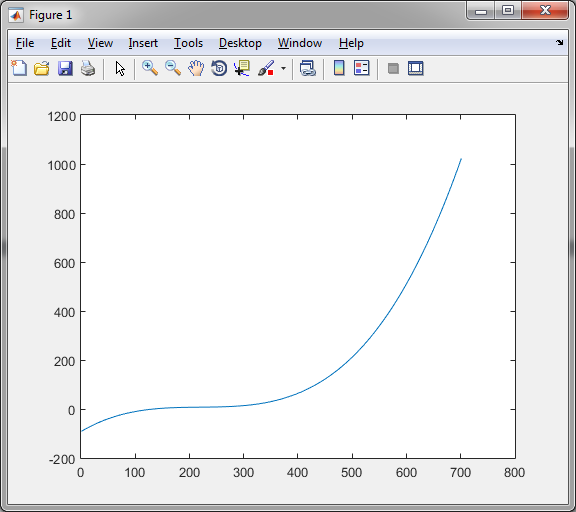


Figure 19 Plotting Polynomials 2

5. Obtaining the roots of *x*3 13*x*2  52  6  0

Table 37 Obtaining Root Values

|  |
| --- |
| w = [1 13 52 6]  w = 1 13 52 6  z = poly(w)  z = 1 -72 1137 -5122 4056 |

6. Confirm the equation

(20*x*3  7*x*2  5*x* 10)(4*x*2 12*x*  3)  80*x*5  212*x*4 124*x*3 121*x*2 105*x*  30

Table 38 Polynomial Equations

|  |
| --- |
| a = [20 -7 5 10]  a = 20 -7 5 10  b = [4 12 -3]  b = 4 12 -3  c = conv(a,b)  c = 80 212 -124 121 105 -30 |

7. Confirm the equation

with a remainder of 59x-41

Table 39 Polynomial Equations

|  |
| --- |
| d = [12 5 -2 3]  d = 12 5 -2 3  e = [3 -7 4]  e = 3 -7 4  [q r] = deconv(d,e)  q = 4 11  r = 0 0 59 -41 |

## 8. Using the following matrix



### 8a) Create a vector V consisting of the elements in the second column of A

Table 40 Creating Vectors 1

|  |
| --- |
| a = [3 7 -4 12; -5 9 10 2; 6 13 8 11; 15 5 4 1];  V = a(:,2)  V =  7  9  13  5 |

### 8b) Create a vector W consisting of the elements in the second row of A

Table 41 Creating Vectors 2

|  |
| --- |
| W = a(2,:)  W = -5 9 10 2 |

## 9. Using the following matrix



### 9a) Find the maximum and minimum values in each column

Table 42 Finding ranges in Matrices 1

|  |
| --- |
| A = [3 7 -4 12; -5 9 10 2; 6 13 8 11; 15 5 4 1];  minA = min(A)  minA = -5 5 -4 1  maxA = max(A)  maxA = 15 13 10 12 |

### 9b) Find the maximum and minimum values in each row

Table 43 Finding ranges in Matrices 2

|  |
| --- |
| B = [3 7 -4 12; -5 9 10 2; 6 13 8 11; 15 5 4 1];  minB = min(B,[],2)  minB =  -4  -5  6  1 |

## 10. Plot a linear function from 0 to 100 on both X and Y axis

Table 44 Plotting Linear Function

|  |
| --- |
| x = [0:10:100];  y = [0:10:100];  plot(x,y) |

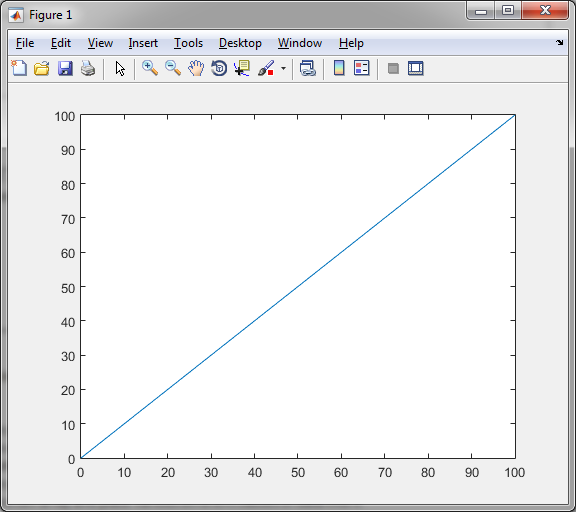


Figure 20 Plotting a Linear Function

## 11. Experimenting with plot features.

Table 45 Plotting Experimentation 1

|  |
| --- |
| F = [0:0.5:100];  G = 0.6 \* sqrt(2.2\*F);  plot(F,G) |

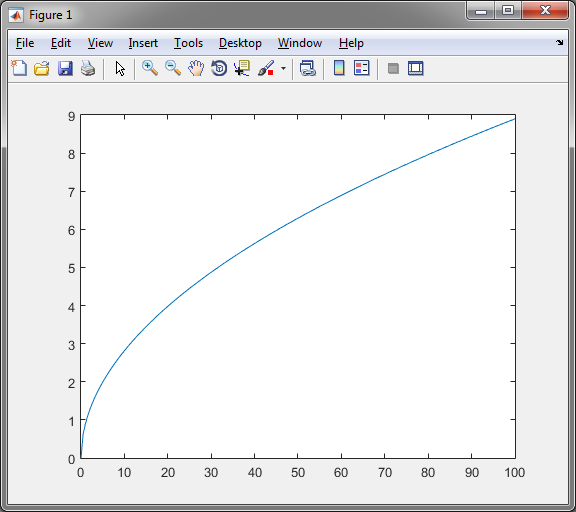


Figure 21 Experimenting with Plot Features 1

Table 46 Plotting Experimentation 2

|  |
| --- |
| xlabel('Time, minutes'),ylabel('Temperature, degrees F'),title('Shooting Star'),grid on |

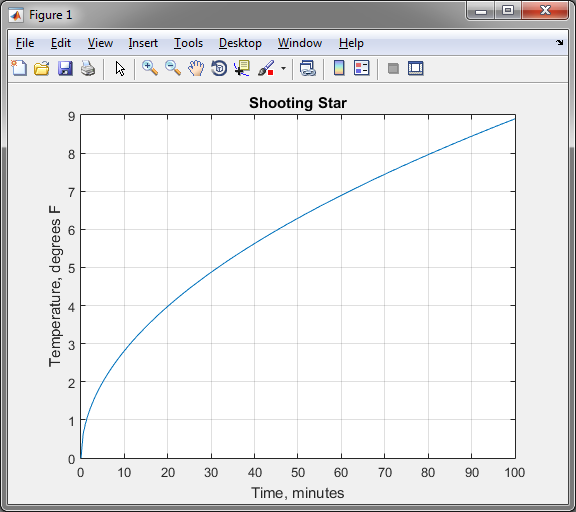


Figure 22 Experimenting with Plot Features 2

## 

## 12. Create a plot of complex numbers.

Table 47 Complex Plots

|  |
| --- |
| a = 0.14+0.7i;  b = [0:0.3:70];  plot(a.^b),xlabel('Real'),ylabel('Imaginary') |

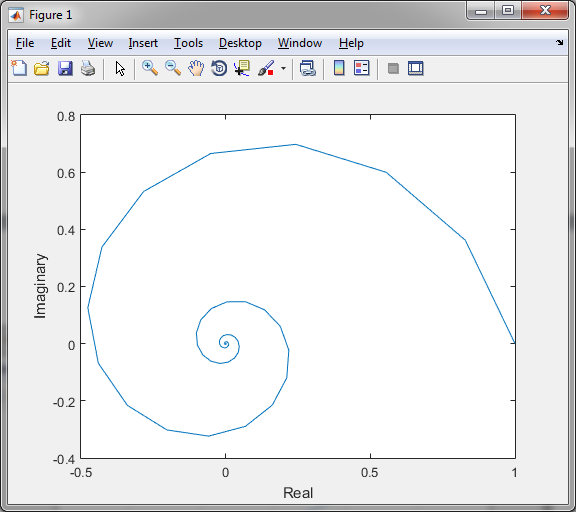


Figure 23 Complex Numbers Plot

## 14. Plot polynomials stored in coefficient vector format.

Table 48 Polynomial Plots

|  |
| --- |
| c = [-18:0.1:18];  d = [12,5,45,-12,-54,2];  plot(c,polyval(d,c)),xlabel('c'),ylabel('d') |

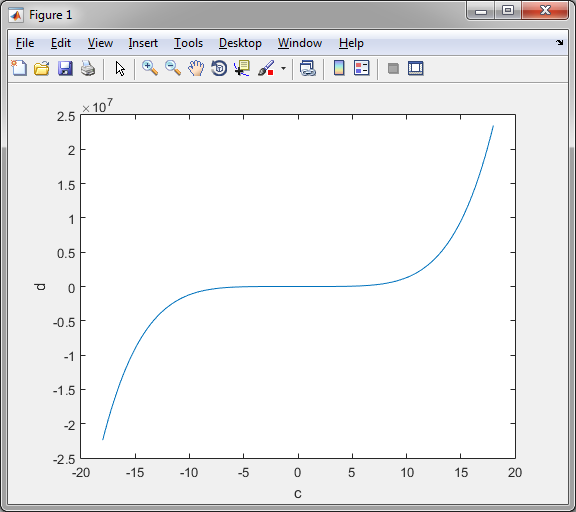


Figure 24 Coefficient Vector Polynomial

## 15. Create your own subplots and overlay plots with legend(s)

Table 49 Exploring Subplots 1

|  |
| --- |
| D = [0:0.25:10];  E = exp(-2.3\*D).\*sin(7\*D+3); subplot(1,2,1); |

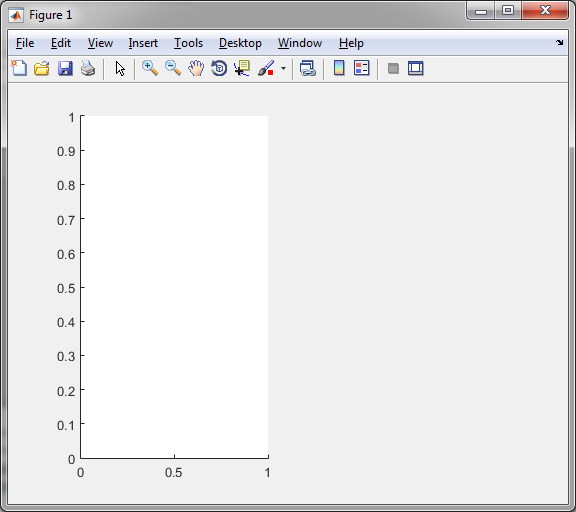


Figure 25 Experimenting with Subplots 1

Table 50 Exploring Subplots 2

|  |
| --- |
| plot(D,E), xlabel('D'), ylabel('E'), axis([0 5 -1 1]); |

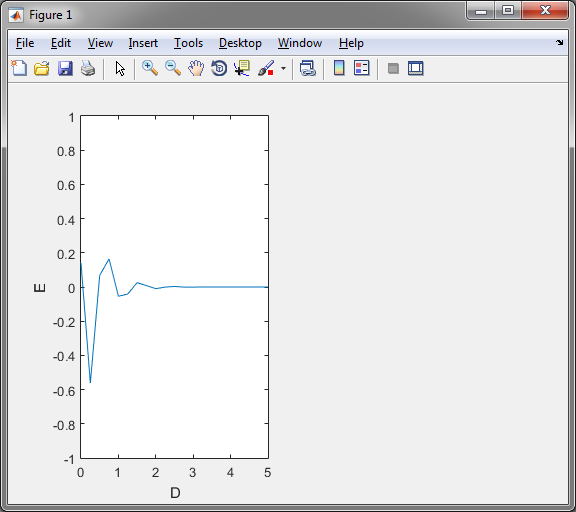


Figure 26 Experimenting with Subplots 2

Table 51 Exploring Subplots 3

|  |
| --- |
| d = [0:0.01:2];  e = sinh(d); c = tanh(e);  subplot(1,2,2);  plot(d,e,d,e,'--'), xlabel('d'), ylabel('Hyperbolic Sine and Tangent'), legend('sinh(d)','tanh(d)') |

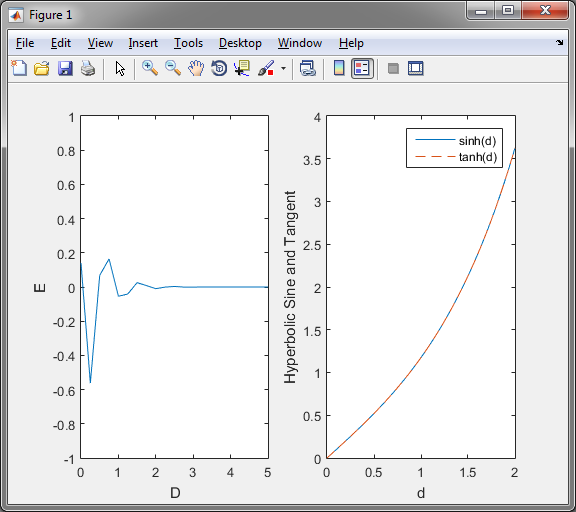


Figure 27 Experimenting with Subplots 3

## 16. Create a cell array and place various different classes of data into it

Table 52 Cell Arrays

|  |
| --- |
| Data = cell(5,1)  Data =  []  []  []  []  []  Data{4,1} = 'Hello'  Data =  []  []  []  'Hello'  []    Data{2,1} = 14  Data =  []  [14]  []  'Hello'  []  Data{5,1} = [3 2; 4 5]  Data =  []  [ 14]  []  'Hello'  [2x2 double] |

## 17. Briefly describe the concept of code vectorisation and vectorise the following strand of code

Table 53 Non-Vectorised Code

|  |
| --- |
| NonVect = zeros(1, 1000);  for i = 1 : 1000  NonVect(i) = 2 \* i  end; |

Vectorization is the simplification of code for example in Table 53 is a non-vectorised loop and the process of rewriting the loop so that instead of processing a single element of an array N times is vectorisation. Table 54 shows a vectorised version which does the exact same process but in less code and variables.

Table 54 Vectorised Code

|  |
| --- |
| i= 1:1000;  vect=(i\*2) |

# I. Exploratory data analysis

## 1. Examine the dataset PIMA

### 1.1 Introduction

This paper has conducted observation on an altered version of PIMA.txt dataset. On examination the dataset had many values of 0 in attributes 2 through 8 it is assumed they were incorrect or missing due to the certain medical attributes that could not possible be 0. To overcome this, the rows that have the missing data have been removed resulting in a reduction of rows from the original 768 to the new 392 losing 376 rows of data. Further examination will be done on the remaining.

### 1.2 Attribute Ranges

Various calculations can be made on the dataset to find out about the range of values each attribute has. The four key aspects that we will look at are the maximum, minimum, average and variance using code in Table 55.

Table 55 Range Psuedocode

|  |
| --- |
| MinumumValues := minimum(dataset)  MaximumValues := maximimum(dataset)  AverageValues := average(dataset)  VarianceValues := variance(dataset) |

After observing Table 56 for the maximum and minimum, attribute 2 representing the ‘plasma glucose concentration a 2 hours in an oral glucose tolerance test’ and attribute 5 representing ‘2-Hour serum insulin’ are both much higher than the minimum and could be an early indicator for their importance in diabetes diagnosis. Attribute 3 could also be considered high, representing ‘diastolic blood pressure’ and could also be an indicator for diabetes.

Table 56 Range Results

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Attribute** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **Maximum** | 17.0000 | 198.0000 | 110.0000 | 63.0000 | 846.0000 | 67.1000 | 2.4200 | 81.0000 |
| **Minimum** | 0.0000 | 56.0000 | 24.0000 | 7.0000 | 14.0000 | 18.2000 | 0.0850 | 21.0000 |
| **Average** | 3.3010 | 122.6276 | 70.6633 | 29.1454 | 156.0561 | 33.0862 | 0.5230 | 30.8648 |
| **Variance** | 0.0010 | 0.0952 | 0.0156 | 0.0111 | 1.4123 | 0.0049 | 0.0000 | 0.0104 |

After observing Table 56 for the average and variance similar to previous, attributes 2, 3, and 5 all have a high average which could mean many of the subjects in the dataset have testing positive for diabetes as it has been shown [2] that PIMA Indian women in particular are more likely to get diabetes than others. Variance is a value normalized by the number of observations, with a variance of 1.4123 attribute 5 could indicates that the data is very spaced apart from the average and from each other.

### 1.3 Correlations

Correlation is a measurement that indicates whether there is a relationship between two or more variables. To make calculations easier the attributes and target class of the PIMA dataset have been split up column wise to make correlation examination easier for example attribute 1 becomes column 1 (c1) and the full correlation is shown in Table 57.

Table 57 Correlations of Attributes

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **c1** | **c2** | **c3** | **c4** | **c5** | **c6** | **c7** | **c8** | **c9** |
| **c1** | 1.0000 | 0.1983 | 0.2134 | 0.0932 | 0.0790 | -0.0253 | 0.0076 | 0.6796 | 0.2566 |
| **c2** | 0.1983 | 1.0000 | 0.2100 | 0.1989 | 0.5812 | 0.2095 | 0.1402 | 0.3436 | 0.5157 |
| **c3** | 0.2134 | 0.2100 | 1.0000 | 0.2326 | 0.0985 | 0.3044 | -0.0160 | 0.3000 | 0.1927 |
| **c4** | 0.0932 | 0.1989 | 0.2326 | 1.0000 | 0.1822 | 0.6644 | 0.1605 | 0.1678 | 0.2559 |
| **c5** | 0.0790 | 0.5812 | 0.0985 | 0.1822 | 1.0000 | 0.2264 | 0.1359 | 0.2171 | 0.3014 |
| **c6** | -0.0253 | 0.2095 | 0.3044 | 0.6644 | 0.2264 | 1.0000 | 0.1588 | 0.0698 | 0.2701 |
| **c7** | 0.0076 | 0.1402 | -0.0160 | 0.1605 | 0.1359 | 0.1588 | 1.0000 | 0.0850 | 0.2093 |
| **c8** | 0.6796 | 0.3436 | 0.3000 | 0.1678 | 0.2171 | 0.0698 | 0.0850 | 1.0000 | 0.3508 |
| **c9** | 0.2566 | 0.5157 | 0.1927 | 0.2559 | 0.3014 | 0.2701 | 0.2093 | 0.3508 | 1.0000 |

#### 1.3.1 Attributes and Target class

When examining the c9 in Table 57 of all the attributes and the target class the cell with the Largest Mutual Correlation (LMC) of 0.5157 is attribute 2 and the target 9. Attribute 2, has the highest correlation to the target class, this further indicates that attribute 2 could be important when trying to predict whether a person has or is likely to have diabetes. To further test if it is important, research into the significance of plasma glucose concentration for type 2 diabetes was done. The study [1] states that subjects with 2 hours plasma glucose over 140 can have an increased risk therefore to test this, using code in Table 58, the rows with over 140 in column 2 have, for the majority, been tested positive for diabetes as shown in Table 59, however not all have in the sample 10 records, some subjects have as high as 180 (8th row) but do not test positive for diabetes.

Table 58 Sample Data Psuedocode

|  |
| --- |
| SampleData := dataset(dataset(allRows, secondColumn)where value is greater than 140, allResults) |

Table 59 Sample Data Results

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| 2.0000 | 197.0000 | 70.0000 | 45.0000 | 543.0000 | 30.5000 | 0.1580 | 53.0000 | 1.0000 |
| 1.0000 | 189.0000 | 60.0000 | 23.0000 | 846.0000 | 30.1000 | 0.3980 | 59.0000 | 1.0000 |
| 5.0000 | 166.0000 | 72.0000 | 19.0000 | 175.0000 | 25.8000 | 0.5870 | 51.0000 | 1.0000 |
| 11.0000 | 143.0000 | 94.0000 | 33.0000 | 146.0000 | 36.6000 | 0.2540 | 51.0000 | 1.0000 |
| 13.0000 | 145.0000 | 82.0000 | 19.0000 | 110.0000 | 22.2000 | 0.2450 | 57.0000 | 0.0000 |
| 3.0000 | 158.0000 | 76.0000 | 36.0000 | 245.0000 | 31.6000 | 0.8510 | 28.0000 | 1.0000 |
| 3.0000 | 180.0000 | 64.0000 | 25.0000 | 70.0000 | 34.0000 | 0.2710 | 26.0000 | 0.0000 |
| 9.0000 | 171.0000 | 110.0000 | 24.0000 | 240.0000 | 45.4000 | 0.7210 | 54.0000 | 1.0000 |
| 8.0000 | 176.0000 | 90.0000 | 34.0000 | 300.0000 | 33.7000 | 0.4670 | 58.0000 | 1.0000 |
| 7. 0000 | 150.0000 | 66.0000 | 42.0000 | 342.0000 | 34.7000 | 0.7180 | 42.0000 | 0.0000 |

#### 1.3.2 Attributes and Attributes

Similar to 1.3.1 to test if there are any relationships between the attributes the same Table 57 was used but with each attribute to another. Table 60 shows the LMC of each column followed by the other attribute for example column 3 and column 6 becomes c3, c6. On examination it can be seen that c6 pairs with two other attributes and c4, c6 is the second highest LMC. The first LMC is c1 and c8 which have been chosen to show the LMC, with a LMC of 0.6796 attribute 1 representing ‘Number of times pregnant’ and attribute 8 representing ‘Age’ are the highest correlation which is to be expected as the older a woman is the more likely they are to have had children and had numerous of them.

Table 60 Attribute Pair Correlation

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Attributes** | c1,c8 | c2,c5 | c3,c6 | c4,c6 | c5,c2 | c6,c4 | c7,c4 | c8,c1 |
| **LMC** | 0.6796 | 0.5812 | 0.3044 | 0.6644 | 0.5812 | 0.6644 | 0.1605 | 0.6796 |

### 1.4 Importance to Prediction

The correlation coefficients in Table 57 indicate how much predictive power the individual attributes group together with respect to each other or the target class in a linear sense. Therefore a regression line for the target classes could possibly be calculated with the attribute that has the LMC. However basing prediction around a single attribute if not advisable as any new dataset used could behave different than the older data or the attribute that has the LMC may not be useful in prediction.

In the previous sections it was determined that with a LMC of 0.6796 attributes 1 and 8 are the highest correlation however this may not be useful for prediction of a target class for diabetes however [4] discusses how within the first 5 years after having a child the progression to diabetes was show to steeply increase. Therefor the next LMC of 0.6644 was also examined, this time attributes 4 representing ‘triceps skin fold thickness’ and 6 representing ‘body mass index’ which again are to be expected to have a close relationship as the more body fat a person has the thicker the triceps become. This correlation may not directly mean diabetes, however these are both attributes that can be a common feature of a person with obesity that could lead to diabetes.

It should also be mentioned that none of these attributes were the LMC to the target class. All had a correlation under 0.2800 except c8 which had 0.3508 compared with c2 which achieved 0.5157. This could suggest that relationships between attributes is not too important as having two correlated attributes may be helpful however correlation does not tell causation of the underlying relationship. For prediction knowing the reason may not matter as long as the pattern continues. [3] suggests that there are better attributes to help predict diabetes than attribute 2 exclusively and adding it to the prediction model, although slightly improving the model, holds greater inconvenience and costs.

### 1.5 Histogram Analysis

Histograms graphically show the location and scale of the data, by creating subplots using the function histogram\_analysis2 we can see all of the columns at once (Figure 28) before selecting on which attribute we want specifically.

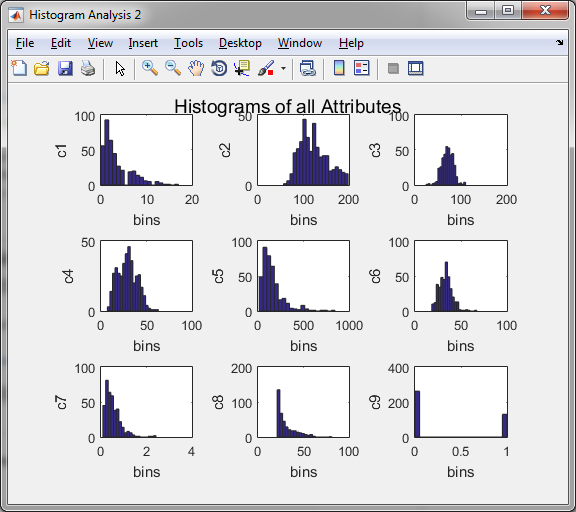


Figure 28 histogram\_analysis2

We can then use the function histogram\_analysis to create a histogram of just the selected column and give a title, and in order to gain more than just visual information we can output sizeOfBins which representing the how many values are within the bins (Table 61).

Table 61 Number of Values in Bin of c3

|  |
| --- |
| sizeOfBins := histogramAnalysisFunction(Column3, 'Title of Figure') |

Of the nine histograms show in Figure 28 the there are none that perfectly fit the definition of normal distribution with most of them resembling a right-skewed distribution; c1, c5, c7. Below are two of the histograms figures that are close to a normal distribution, these are histograms of the 3rd (Figure 29) and 6th (Figure 30) attributes. Both of these figures resemble the shape of a normal distribution and using the code of Table 61 it becomes clearer how many the values the bins hold. Of the two figures Figure 29 has a better spread of values on either side compared to Figure 30 which spikes in the centre and a more uneven distribution of bin on either side. Therefore the histogram that resembles the most normal distribution is Figure 29, the 3rd attribute representing ‘Diastolic blood pressure (mm Hg)’.

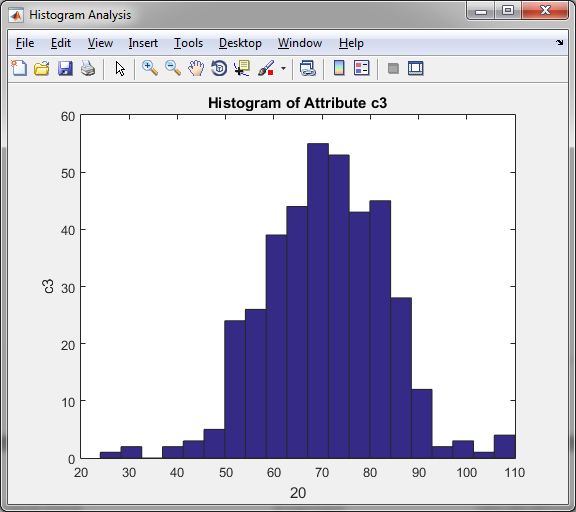
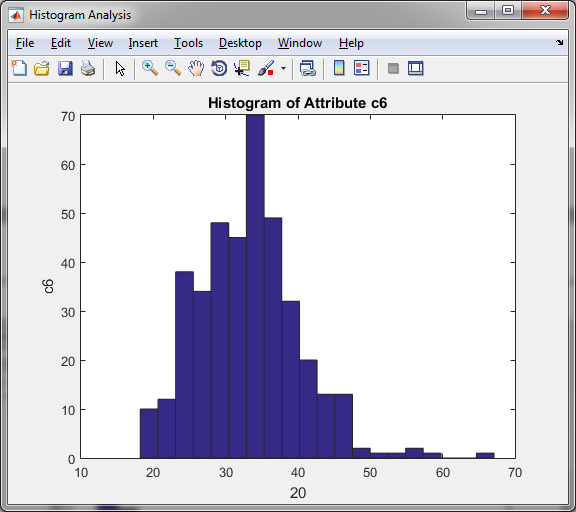
 

Figure 29 Histogram of Attribute c3 Figure 30 Histogram of Attribute c6

### 1.6 Scatter Plot Analysis

A scatterplot is a useful visual representation of the relationship of two variables and can aid to the assumption of correlation. Similar to the functions above, to quickly examine all the possible attribute pairs a function scatter\_plot2 has been created that, when given one of the column attributes will create the specific subplots of scatter plots for all pairwise attributes, for example running the function and giving the input of c1 will call Figure 31 of attribute 1 vs 2:8. However this function won’t show the plots lower than the user input for example if c5 is chosen then the subplots will be 5 vs 6:8. This is due to the code being ‘if’, ‘else’ statements and on the assumption that all scatter plots will be called in order C1 to C8 to quickly examine. Further figures for this function are in the appendix to show how these scatter plots are created based on the chosen column.

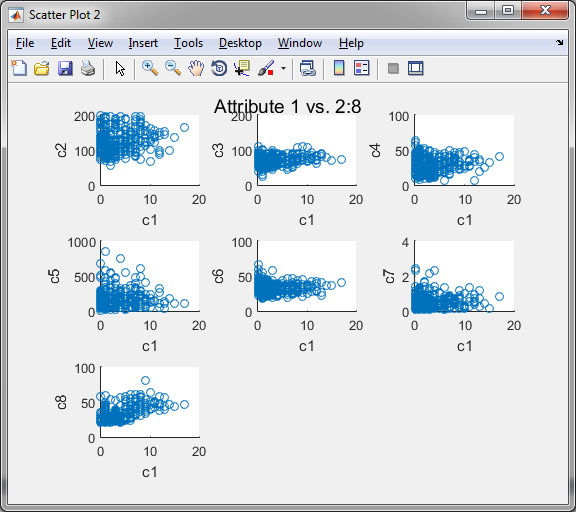


Figure 31 Attributes 1 vs. 2:8

To allow for specific attribute pair a function called scatter\_plot was created. On examination there are two plots that could suggest linear dependency, Figure 32 and Figure 33.

* Figure 32 of attribute 2, representing Plasma glucose concentration a 2 hours in an oral glucose tolerance test and attribute 5, 2-Hour serum insulin (mu U/ml).
* Figure 33 of attribute 4, representing Triceps skin fold thickness (mm) and attribute 6, Body mass index (weight in kg/(height in m)^2).

Both of these show a positive linear dependency as we would expect from previous sections, for example as body mass increased so should triceps. The markers are more spread on the latter half of Figure 32 could be something that happens to a rare few subjects or outliers and the same for the few markers at the edges of the Figure 33. Of the two figures the one with the most linearity is Figure 33 as there is a clear positive correlation between skin thickness and body mass index whereas in Figure 32 the markers show a strong linear dependency at the start and then become more spread out.

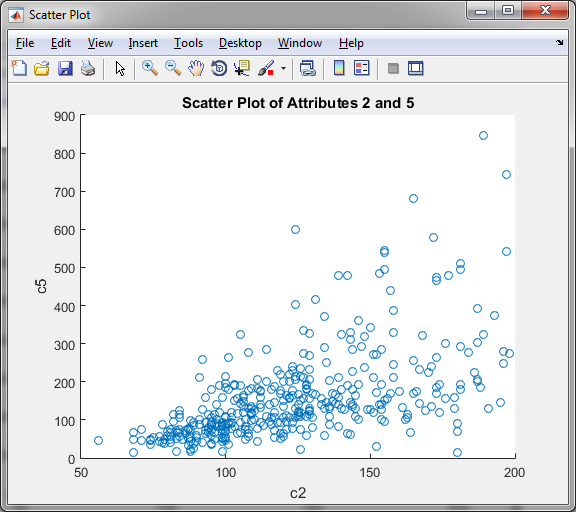
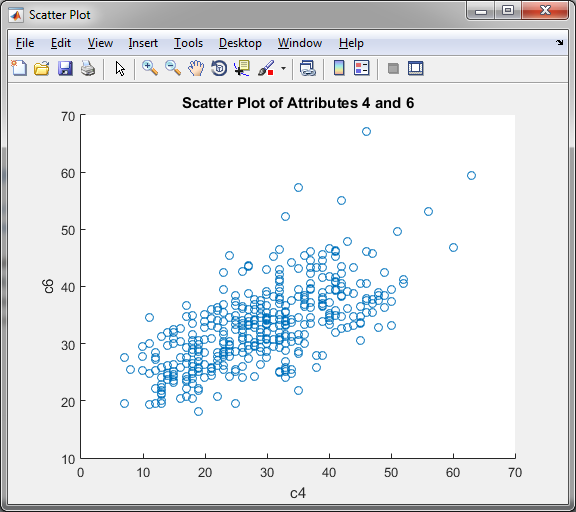
 

Figure 32 Scatter Plot of Attributes 2 and 5 Figure 33 Scatter Plot of Attributes 4 and 6

# II. Data pre-processing

## 2. Pre-processing Tasks

### 2.1 Introduction

Data pre-processing is the act of transforming any raw data to prepare it for another processing procedure and is a common practice of data mining. Two of these methods will be used on the PIMA dataset; normalization and discretisation.

### 2.2 Normalization

The process of organising data such that it is more efficient and can aid in feature extraction that will be useful for neural networks. Due to the variety of measurement units for example weight could be in kilograms or pounds may lead to vary different results. Through normalisation the data is restricted to a value between -1 and 1.

The method of normalization that will be used in this project will be z-score normalization in which attribute *x* values are normalized according to the average and standard deviation of *x*. The value *x* is then normalized using the equation (1)

(1)

Here is the average and the standard deviation of X. An example of how to achieve these variables is shown in Table 62 with data representing the user input such as one of the column attributes.

Table 62 Normalisation Program

|  |
| --- |
| **Program** [normalizedVector] := normalized(data)  Average of data := average(data)  Standard Deviation of data := maximum(data)  normalizedVector := size(data)  **if** Standard Deviation of data **is greater than** 0  normalizedVector := (data - Average of data) / Standard Deviation of data  **END** |

To show the program works correctly normalization has been applied to attribute 3 of the PIMA dataset. Table 63 shows the first 5 values of the original data of attribute 3 and the new normalized values.

Table 63 Original and Normalised Values

|  |  |
| --- | --- |
| Original Values | Normalized Values |
| 66 | -0.0424 |
| 40 | -0.2788 |
| 50 | -0.1878 |
| 70 | -0.0060 |
| 60 | -0.0969 |

### 2.3 Discretization

Discretisation is the dividing of data into finite elements to prepare for analysis. A numeric attribute can be categorized by dividing the chosen dataset into sub-ranges called bins. For example attribute ‘Age’ could be divided into groups of 20-35, 36-51, 60+ and each value be categorised into one of these groups.

The method of discretization that will be used in this project will be Equal-width binning which divides the range of values into N sub-ranges of the same size using the equation (2)

(2)

An example of how to achieve these variables is shown in Table 64 with data representing the user input such as one of the column attributes.

Table 64 Discretize Program

|  |
| --- |
| **Program** [binNumber, binCentre] := discretize\_attribute (data)  numberOfBins := 10;  binEdges := linspace(minimum(data), maximum(data), numberOfBins+1);  binLower := binEdges(1 to end -1);  binUpper := binEdges(2 to end);  binCentre := (binUpper+binLower) each element divide by 2;  [~, binNumber] := histcounts(data, numberOfBins);  **END** |

To show the program works correctly normalization has been applied to attribute 3 of the PIMA dataset. Table 65 shows the first 5 values of the original data of attribute 3, the new discretized values representing the bin label of whichever bin they were put into and the bin median of the bin edges.

Table 65 Original, Discretized Values and Bin Medians

|  |  |  |
| --- | --- | --- |
| Original Values | Discretized Values | Bin Medians |
| 66 | 5 | 28.3000 |
| 40 | 2 | 36.9000 |
| 50 | 4 | 45.5000 |
| 70 | 6 | 54.1000 |
| 60 | 5 | 62.7000 |

# **III. Data set splitting**

## 3 Data Splitting Techniques

### 3.1 Introduction

Data splitting techniques are used to allow for the validation of data mining models datasets can be divided into training and testing subsets. There are two processes for the creation of new data subsets; a split and a subset. A subset is the selection of cases that match the criteria leaving a single new dataset whereas a split acts as a partition of a dataset based on certain criteria splitting the dataset into two or more datasets.

### 3.2 Class Based Data Subsets

An example of a subset is class based data subset where using the code in Table 66 the data that has a class label of 0 has been placed into a new dataset called ‘ClassLabelZero’ and a separate dataset using similar code for ‘ClassLabelOne’. Each attribute then had their average and standard deviation computed shown in Table 67 and Table 68 respectively, which when compared show that certain attributes are higher such as the expected attributes 2 and 5 as seen in previous questions and could be used to predict diabetes.

Table 66 All rows with Class Label 0

|  |
| --- |
| ClassLabelZero := dataset(dataset(allRows, ninthColumn) where equal to Zero, allResults)  Average of ClassLabelZero := average(ClassLabelZero)  StandardDeviation of ClassLabelZero := standardDeviation(ClassLabelZero) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table 67 Average of Class Label 0 and 1   |  |  |  | | --- | --- | --- | |  | **Ave of 0** | **Ave of 1** | | **c1** | 2.7214 | 4.4692 | | **c2** | 111.4313 | 145.1923 | | **c3** | 68.9695 | 74.0769 | | **c4** | 27.2519 | 32.9615 | | **c5** | 130.8550 | 206.8462 | | **c6** | 31.7508 | 35.7777 | | **c7** | 0.4722 | 0.6256 | | **c8** | 28.3473 | 35.9385 | | **c9** | 0.0000 | 1.0000 | | Table 68 Std of Class Label 0 and 1   |  |  |  | | --- | --- | --- | |  | **Std 0** | **Std 1** | | **c1** | 2.6178 | 3.9162 | | **c2** | 24.6421 | 29.8394 | | **c3** | 11.8928 | 13.0215 | | **c4** | 10.4341 | 9.6428 | | **c5** | 102.6262 | 132.6999 | | **c6** | 6.7950 | 6.7347 | | **c7** | 0.2992 | 0.4059 | | **c8** | 8.9890 | 10.6347 | | **c9** | 0.0000 | 0.0000 | |

### 3.3 Function Based Data Subsets

3.3.1 Random Size Subsets

One example of a splitting is random size subsets where using the code in Table 69 a function called ‘divideset1’ has been created that calls in the inputs for a dataset and uses the probability p\_train, it then divides the data into two non-overlapping data subsets and outputs the training data (training1\_s) and testing data (testing1\_s). To test this function the PIMA dataset was run 20 times, saving all the results to ‘divideset1Results.txt’. The average length of the training\_s is 243 and testing1\_s is 149 due to new dataset’s sizes being influenced by p\_train. divideset1Results.txt'

Table 69 Loop Run and Save for divideset1

|  |
| --- |
| turnOnDiary(‘Location to save to and name file will be saved as’) FOR i := 1 to 20 do  [TrainingSubset, TestingSubset] := DivideSetFunction(dataset, probabilityValue) END LOOP turnOffDiary('turn off') |

#### 3.3.2 Fixed Size Subsets

Another example is fixed size subsets which have been implemented using the function called ‘divideset2’, and similar to ‘divideset1’ will calls in the inputs for a dataset and a probability, divides the data into two non-overlapping data subsets and outputs training\_s and testing\_s, but unlike the first, ‘divideset2’ creates new datasets that are fixed based on p\_train. This function was tested on the PIMA dataset and also run 20 times, saving all the results to ‘divideset2Results.text’. The function always creates the same size dataset of training\_s being length 259 and testing\_s 133.

# **IV. Prediction of the Target Variable using an Artificial Neural Network**

## Datasets

When creating an Artificial Neural Network (ANN) choosing the correct datasets that will be used for input and target value is important. One consideration is that all data that will be used for the input and target should be between -1 and 1 meaning any values greater than 1 or smaller than -1 need to be normalised.

Therefore all of the columns attributes i.e. c1 were individually normalize using the function created for 2a and create a new vector named c1n. All the new normalized attributes were put together and transposed into inputs ‘atrnt’ and targets ‘tarnt’ to be used for training the ANN. After creating an ANN using the previously mention inputs and targets, the hidden layer is given 10 neurons and is trained.

## Performance

The performance plot shows how the mean squared error drops rapidly as it learning. The blue line indicated the decreasing error on the training data, the green line indicates the validation error and training stops once the validation error stops decreasing. The red line shows the testing errors showing how well it will handle new data. Figure 34 show the performance plot of this ANN with the best validation performance at the 3rd epoch.

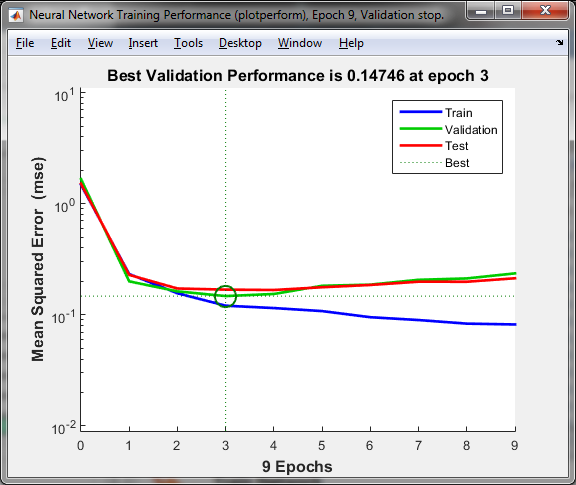


Figure 34 Performance Plot

## Error Histogram

The error histogram shows the distribution of errors, most errors occur close to 0 indicated by the yellow line. If errors are not it may be required for more neurons to be used for a better result. In Figure 35 the errors are show that although the error are around the zero line they sit mostly to the left of it. This could indicate that more neuron are needed to have a more even distribution around the zero line.

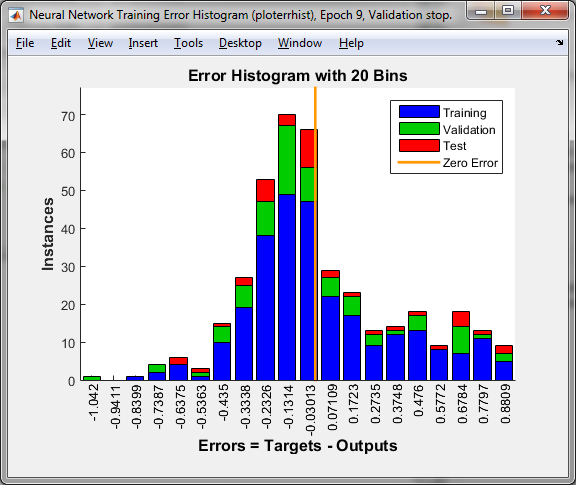


Figure 35 Error Histogram

## Regression

The regression plot shows network outputs versus targets, ideally each point will fall close to the dotted line and any that fall out of group are considered outliers which could warrant investigation as they could offer additional insight into the problems. The regression line indicated by the 45 degrees coloured line indicated how well the outputs are centred around the targets and should fall near the dotted line.

The overall distance between the dotted line and the coloured line is summarized by the R-value which should be near 1. Figure 36 shows the regression plot for the created ANN and that there are two different target areas that match the class labels but do not gather in together instead are a vertical line.

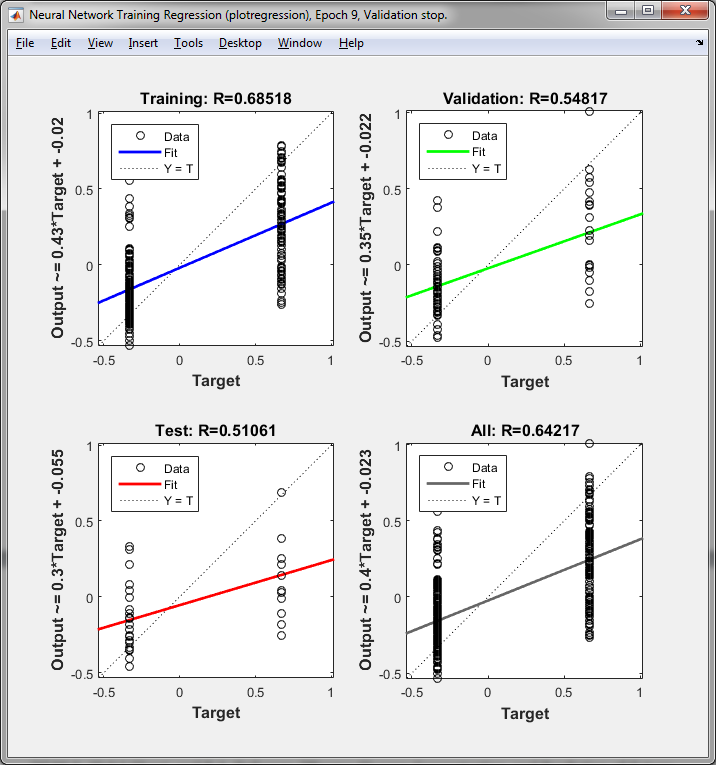


Figure 36 Regression Plot

# References

[1] M.A. Abdul-Ghani and R.A. Defronzo “Plasma Glucose Concentration and Prediction of Future Risk of Type 2 Diabetes”, Diabetes Care vol. 32, 2009. doi: 10.2337/dc09-S309

[2] R.S. Lindsay, T. Funahashi, R. L. Hanson, Y. Matsuzawa, S. Tanaka, P. A. Tataranni, P.A., Knowler, W.C. and J. Krakoff, “Adiponectin and development of type 2 diabetes in the Pima Indian population”, The Lancet,360(9326), pp.57-58, 2002.

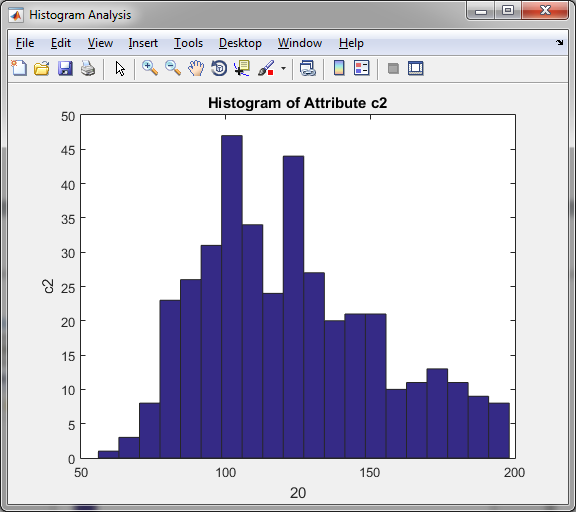
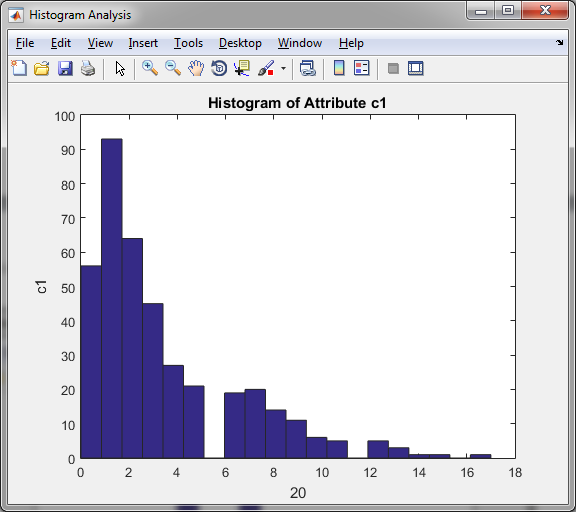
[3] M.P. Stern, K. Williams, and S.M. Haffner, “Identification of persons at high risk for type 2 diabetes mellitus: do we need the oral glucose tolerance test?”, *Annals of Internal Medicine*, *136*(8), pp.575-581, 2002.

[4] C. Kim, K.M. Newton, and R.H. Knopp, “Gestational Diabetes and the Incidence of Type 2 Diabetes A systematic review”, *Diabetes care*, *25*(10), pp.1862-1868, 2002.

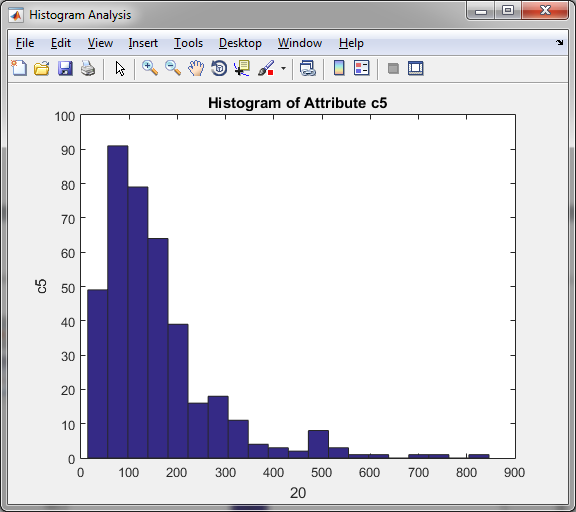
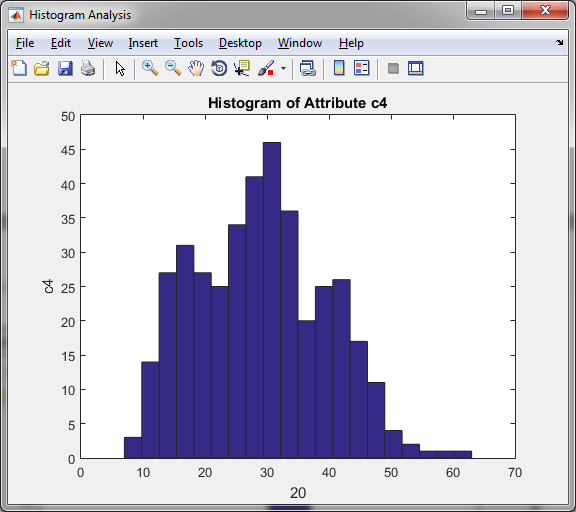
# Appendix

## Histograms

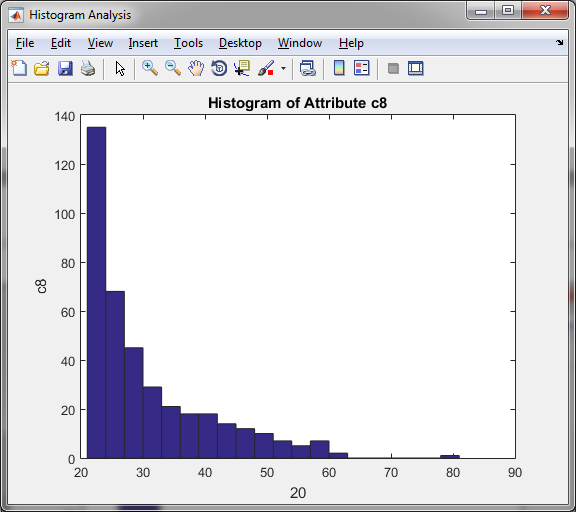
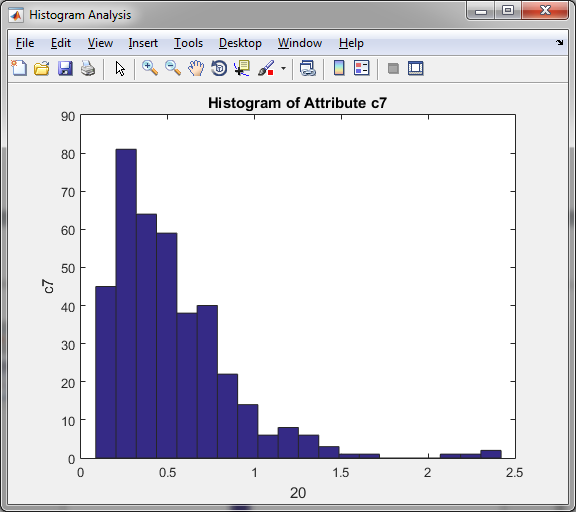
### C1 C2



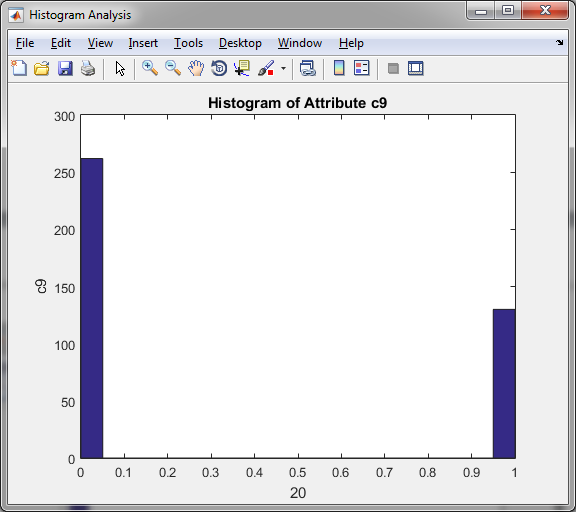
### C4 C5



### C7 C8

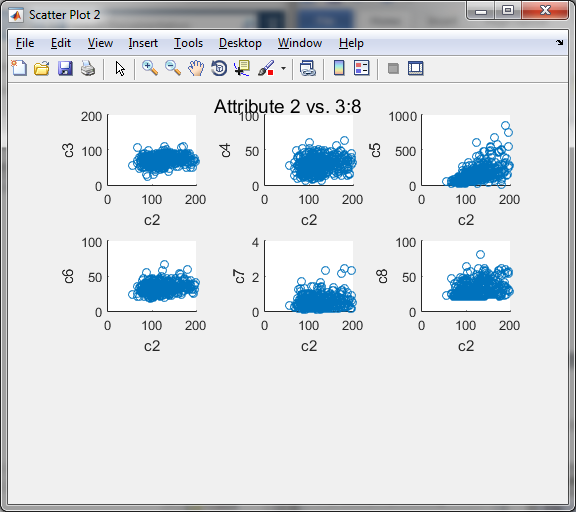
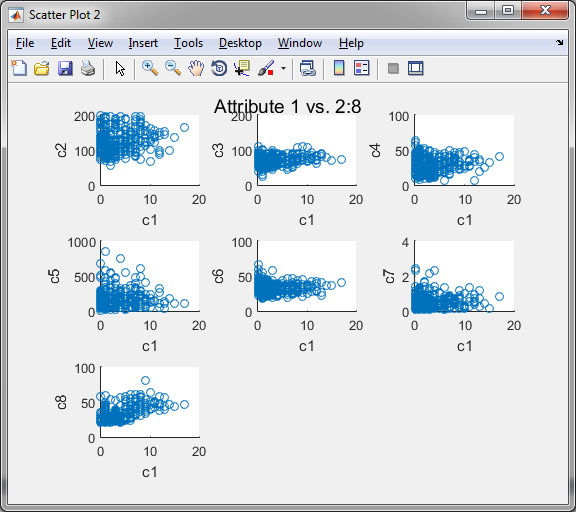


### C9

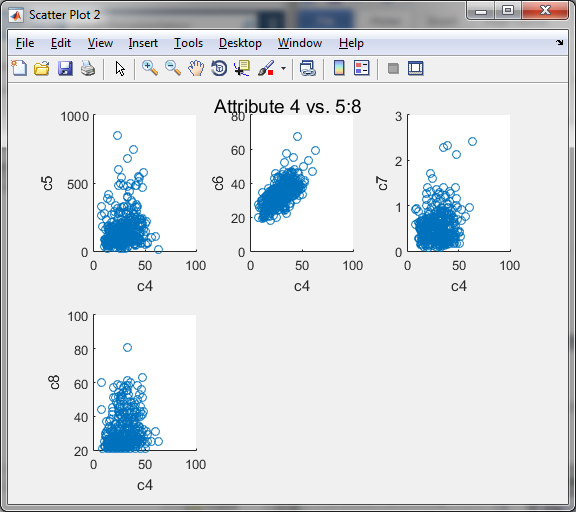
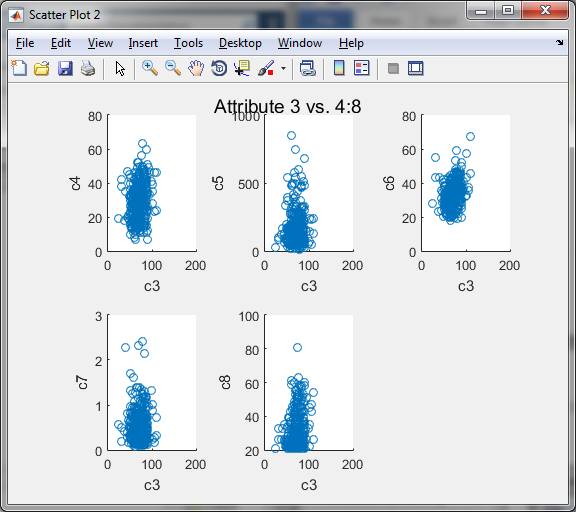


## Scatter Plots

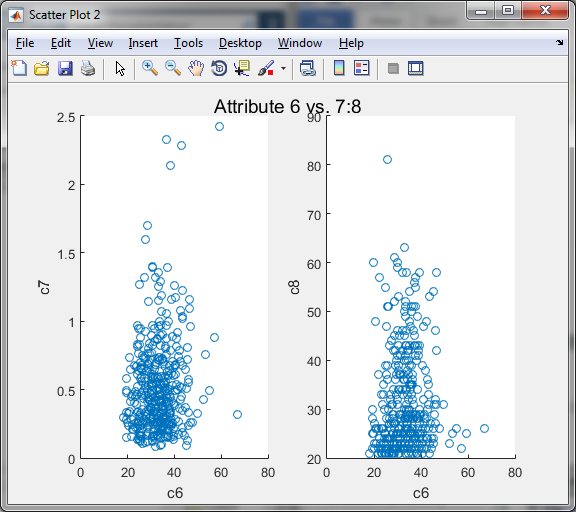
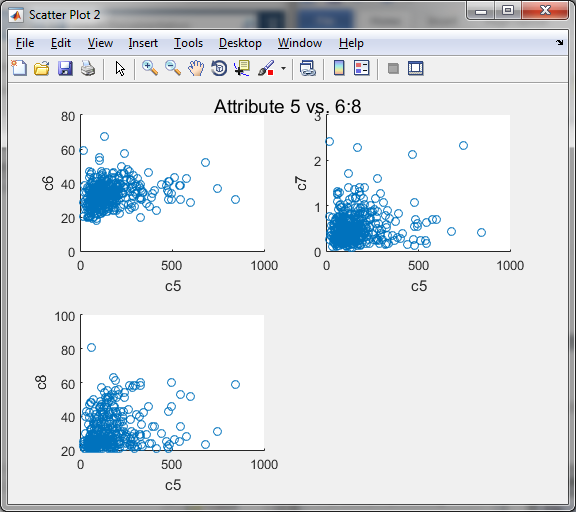
### C1 C2



### C3 C4



### C5 C6



### C7

