NAME- **CHAYANIKA MUKHERJEE**

YEAR - **3RD**

COLLEGE- **TECHNO MAIN SALT LAKE**

1. ***PROJECT ON “ORANGE” DATASET USING R***

**LINEAR REGRESSION :-** It is the process of finding a line that best fits the data points available on the plot, so that we can use it to predict output values for inputs that are not present in the dataset we have, with the belief that those outputs would fall on the line.

* **Write a R program to load ORANGE dataset. Find the class. Print the first 6 instances using head() function.**

>data("Orange")

> head(Orange)

Tree age circumference

1 1 118 30

2 1 484 58

3 1 664 87

4 1 1004 115

5 1 1231 120

6 1 1372 142

> print(Orange)

Tree age circumference

1 1 118 30

2 1 484 58

3 1 664 87

4 1 1004 115

5 1 1231 120

6 1 1372 142

7 1 1582 145

8 2 118 33

9 2 484 69

10 2 664 111

11 2 1004 156

12 2 1231 172

13 2 1372 203

14 2 1582 203

15 3 118 30

16 3 484 51

17 3 664 75

18 3 1004 108

19 3 1231 115

20 3 1372 139

21 3 1582 140

22 4 118 32

23 4 484 62

24 4 664 112

25 4 1004 167

26 4 1231 179

27 4 1372 209

28 4 1582 214

29 5 118 30

30 5 484 49

31 5 664 81

32 5 1004 125

33 5 1231 142

34 5 1372 174

35 5 1582 177

>cor(Orange$circumference,Orange$age)

[1] 0.9135189

> cor(Orange$age,Orange$circumference)

[1] 0.9135189

> model <- lm(age ~ circumference, data= Orange)

> summary(model)

Call:

lm(formula = age ~ circumference, data = Orange)

Residuals:

Min 1Q Median 3Q Max

-317.88 -140.90 -17.20 96.54 471.16

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 16.6036 78.1406 0.212 0.833

circumference 7.8160 0.6059 12.900 1.93e-14 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 203.1 on 33 degrees of freedom

Multiple R-squared: 0.8345, Adjusted R-squared: 0.8295

F-statistic: 166.4 on 1 and 33 DF, p-value: 1.931e-14

***Prediction of age based on circumference***

|  |
| --- |
| > predict(model,data.frame("circumference"=177))  1  1400.035  > predict(model,data.frame("circumference"=199))  1  1571.987  > predict(model,data.frame("circumference"=82))  1  657.5155  > predict(model,data.frame("circumference"=100000000))  1  781599861  >model <- lm(circumference ~ age, data= Orange)  > summary(model)  Call:  lm(formula = circumference ~ age, data = Orange)  Residuals:  Min 1Q Median 3Q Max  -46.310 -14.946 -0.076 19.697 45.111  Coefficients:  Estimate Std. Error t value Pr(>|t|)  (Intercept) 17.399650 8.622660 2.018 0.0518 .  age 0.106770 0.008277 12.900 1.93e-14 \*\*\*  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  Residual standard error: 23.74 on 33 degrees of freedom  Multiple R-squared: 0.8345, Adjusted R-squared: 0.8295  F-statistic: 166.4 on 1 and 33 DF, p-value: 1.931e-14 |
|  |
| |  | | --- | |  | |

***Prediction of circumference based on age***

> predict(model,data.frame("age"=20))

1

19.53506

> predict(model,data.frame("age"=46))

1

22.31109

> predict(model,data.frame("age"=111000))

1

11868.91

> predict(model,data.frame("age"=13))

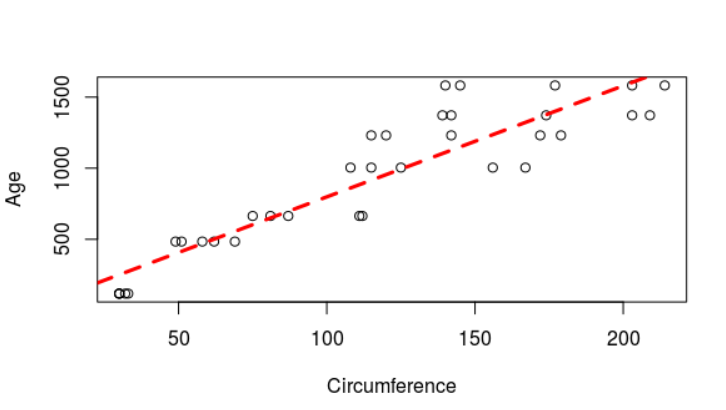
1

18.78766

>model <- lm(age ~ circumference,data= Orange)

> plot(Orange$circumference, Orange$age,xlab ='Circumference',ylab = 'Age')

> abline(model,col="red",lty=2,lwd=3)



**CONCLUSION** :- The above code uses Linear Regression model to predict the Age w.r.t circumference and the “best fit” line is represented with the red dotted line.

1. **PROJECT ON CLUSTERING ON ‘IRIS’ DATASET**

**USING R**

**CLUSTERING :-** It is the process of grouping of objects that are similar to each other forming a cluster, and dissimilar data points forming another cluster. It is an unsupervised learning and it basically works on unlabelled datasets.

* **Write a R program to load an ‘IRIS’ dataset and plot a cluster graph.**

> data('iris')

> head(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

1 5.1 3.5 1.4 0.2 setosa

2 4.9 3.0 1.4 0.2 setosa

3 4.7 3.2 1.3 0.2 setosa

4 4.6 3.1 1.5 0.2 setosa

5 5.0 3.6 1.4 0.2 setosa

6 5.4 3.9 1.7 0.4 setosa

> print(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width Species

1 5.1 3.5 1.4 0.2 setosa

2 4.9 3.0 1.4 0.2 setosa

3 4.7 3.2 1.3 0.2 setosa

4 4.6 3.1 1.5 0.2 setosa

5 5.0 3.6 1.4 0.2 setosa

6 5.4 3.9 1.7 0.4 setosa

7 4.6 3.4 1.4 0.3 setosa

8 5.0 3.4 1.5 0.2 setosa

9 4.4 2.9 1.4 0.2 setosa

10 4.9 3.1 1.5 0.1 setosa

11 5.4 3.7 1.5 0.2 setosa

12 4.8 3.4 1.6 0.2 setosa

13 4.8 3.0 1.4 0.1 setosa

14 4.3 3.0 1.1 0.1 setosa

15 5.8 4.0 1.2 0.2 setosa

16 5.7 4.4 1.5 0.4 setosa

17 5.4 3.9 1.3 0.4 setosa

18 5.1 3.5 1.4 0.3 setosa

19 5.7 3.8 1.7 0.3 setosa

20 5.1 3.8 1.5 0.3 setosa

21 5.4 3.4 1.7 0.2 setosa

22 5.1 3.7 1.5 0.4 setosa

23 4.6 3.6 1.0 0.2 setosa

24 5.1 3.3 1.7 0.5 setosa

25 4.8 3.4 1.9 0.2 setosa

26 5.0 3.0 1.6 0.2 setosa

27 5.0 3.4 1.6 0.4 setosa

28 5.2 3.5 1.5 0.2 setosa

29 5.2 3.4 1.4 0.2 setosa

30 4.7 3.2 1.6 0.2 setosa

31 4.8 3.1 1.6 0.2 setosa

32 5.4 3.4 1.5 0.4 setosa

33 5.2 4.1 1.5 0.1 setosa

34 5.5 4.2 1.4 0.2 setosa

35 4.9 3.1 1.5 0.2 setosa

36 5.0 3.2 1.2 0.2 setosa

37 5.5 3.5 1.3 0.2 setosa

38 4.9 3.6 1.4 0.1 setosa

39 4.4 3.0 1.3 0.2 setosa

40 5.1 3.4 1.5 0.2 setosa

41 5.0 3.5 1.3 0.3 setosa

42 4.5 2.3 1.3 0.3 setosa

43 4.4 3.2 1.3 0.2 setosa

44 5.0 3.5 1.6 0.6 setosa

45 5.1 3.8 1.9 0.4 setosa

46 4.8 3.0 1.4 0.3 setosa

47 5.1 3.8 1.6 0.2 setosa

48 4.6 3.2 1.4 0.2 setosa

49 5.3 3.7 1.5 0.2 setosa

50 5.0 3.3 1.4 0.2 setosa

51 7.0 3.2 4.7 1.4 versicolor

52 6.4 3.2 4.5 1.5 versicolor

53 6.9 3.1 4.9 1.5 versicolor

54 5.5 2.3 4.0 1.3 versicolor

55 6.5 2.8 4.6 1.5 versicolor

56 5.7 2.8 4.5 1.3 versicolor

57 6.3 3.3 4.7 1.6 versicolor

58 4.9 2.4 3.3 1.0 versicolor

59 6.6 2.9 4.6 1.3 versicolor

60 5.2 2.7 3.9 1.4 versicolor

61 5.0 2.0 3.5 1.0 versicolor

62 5.9 3.0 4.2 1.5 versicolor

63 6.0 2.2 4.0 1.0 versicolor

64 6.1 2.9 4.7 1.4 versicolor

65 5.6 2.9 3.6 1.3 versicolor

66 6.7 3.1 4.4 1.4 versicolor

67 5.6 3.0 4.5 1.5 versicolor

68 5.8 2.7 4.1 1.0 versicolor

69 6.2 2.2 4.5 1.5 versicolor

70 5.6 2.5 3.9 1.1 versicolor

71 5.9 3.2 4.8 1.8 versicolor

72 6.1 2.8 4.0 1.3 versicolor

73 6.3 2.5 4.9 1.5 versicolor

74 6.1 2.8 4.7 1.2 versicolor

75 6.4 2.9 4.3 1.3 versicolor

76 6.6 3.0 4.4 1.4 versicolor

77 6.8 2.8 4.8 1.4 versicolor

78 6.7 3.0 5.0 1.7 versicolor

79 6.0 2.9 4.5 1.5 versicolor

80 5.7 2.6 3.5 1.0 versicolor

81 5.5 2.4 3.8 1.1 versicolor

82 5.5 2.4 3.7 1.0 versicolor

83 5.8 2.7 3.9 1.2 versicolor

84 6.0 2.7 5.1 1.6 versicolor

85 5.4 3.0 4.5 1.5 versicolor

86 6.0 3.4 4.5 1.6 versicolor

87 6.7 3.1 4.7 1.5 versicolor

88 6.3 2.3 4.4 1.3 versicolor

89 5.6 3.0 4.1 1.3 versicolor

90 5.5 2.5 4.0 1.3 versicolor

91 5.5 2.6 4.4 1.2 versicolor

92 6.1 3.0 4.6 1.4 versicolor

93 5.8 2.6 4.0 1.2 versicolor

94 5.0 2.3 3.3 1.0 versicolor

95 5.6 2.7 4.2 1.3 versicolor

96 5.7 3.0 4.2 1.2 versicolor

97 5.7 2.9 4.2 1.3 versicolor

98 6.2 2.9 4.3 1.3 versicolor

99 5.1 2.5 3.0 1.1 versicolor

100 5.7 2.8 4.1 1.3 versicolor

101 6.3 3.3 6.0 2.5 virginica

102 5.8 2.7 5.1 1.9 virginica

103 7.1 3.0 5.9 2.1 virginica

104 6.3 2.9 5.6 1.8 virginica

105 6.5 3.0 5.8 2.2 virginica

106 7.6 3.0 6.6 2.1 virginica

107 4.9 2.5 4.5 1.7 virginica

108 7.3 2.9 6.3 1.8 virginica

109 6.7 2.5 5.8 1.8 virginica

110 7.2 3.6 6.1 2.5 virginica

111 6.5 3.2 5.1 2.0 virginica

112 6.4 2.7 5.3 1.9 virginica

113 6.8 3.0 5.5 2.1 virginica

114 5.7 2.5 5.0 2.0 virginica

115 5.8 2.8 5.1 2.4 virginica

116 6.4 3.2 5.3 2.3 virginica

117 6.5 3.0 5.5 1.8 virginica

118 7.7 3.8 6.7 2.2 virginica

119 7.7 2.6 6.9 2.3 virginica

120 6.0 2.2 5.0 1.5 virginica

121 6.9 3.2 5.7 2.3 virginica

122 5.6 2.8 4.9 2.0 virginica

123 7.7 2.8 6.7 2.0 virginica

124 6.3 2.7 4.9 1.8 virginica

125 6.7 3.3 5.7 2.1 virginica

126 7.2 3.2 6.0 1.8 virginica

127 6.2 2.8 4.8 1.8 virginica

128 6.1 3.0 4.9 1.8 virginica

129 6.4 2.8 5.6 2.1 virginica

130 7.2 3.0 5.8 1.6 virginica

131 7.4 2.8 6.1 1.9 virginica

132 7.9 3.8 6.4 2.0 virginica

133 6.4 2.8 5.6 2.2 virginica

134 6.3 2.8 5.1 1.5 virginica

135 6.1 2.6 5.6 1.4 virginica

136 7.7 3.0 6.1 2.3 virginica

137 6.3 3.4 5.6 2.4 virginica

138 6.4 3.1 5.5 1.8 virginica

139 6.0 3.0 4.8 1.8 virginica

140 6.9 3.1 5.4 2.1 virginica

141 6.7 3.1 5.6 2.4 virginica

142 6.9 3.1 5.1 2.3 virginica

143 5.8 2.7 5.1 1.9 virginica

144 6.8 3.2 5.9 2.3 virginica

145 6.7 3.3 5.7 2.5 virginica

146 6.7 3.0 5.2 2.3 virginica

147 6.3 2.5 5.0 1.9 virginica

148 6.5 3.0 5.2 2.0 virginica

149 6.2 3.4 5.4 2.3 virginica

150 5.9 3.0 5.1 1.8 virginica

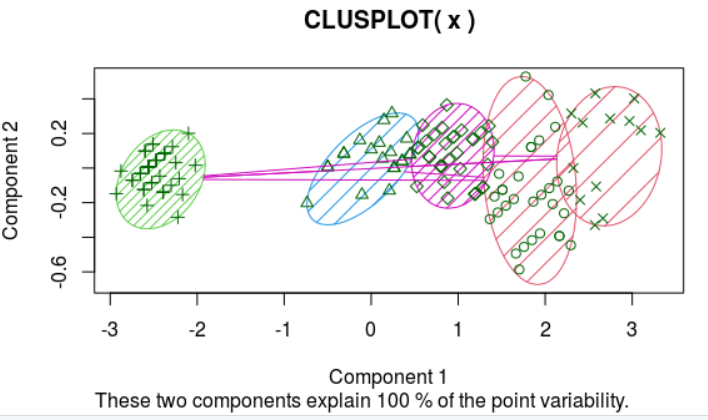
> library(cluster)

> x=iris[,3:4]

> model=kmeans(x,5)

> clusplot(x,model$cluster)

> clusplot(x,model$cluster,color = T,shade = T)

****

**CONCLUSION :-** The above code helps to create 4 clusters on “IRIS” dataset with the help of K-Means Clustering Model.