**d=read.csv("emp.csv")**

**> d**

**> a=is.na(d $age)**

**> a**

> m=mean(d $age,na.rm=TRUE)

> m

> d $age[a]=m

> me=median(d $salary,na.rm=TRUE)

> me

> s=is.na(d $salary)

> s

> s=is.na(d $salary)

> s

> normalize<-function(x) + {return(x-min(x)/(max(x)-min(x)))}

> df=data.frame(s=d $age,a=d $salary,stringsAsFactors = FALSE)

> df

> normalize<-function(x) + {return(x-min(x)/(max(x)-min(x)))}

> df=data.frame(s=d $age,a=d $salary,stringsAsFactors = FALSE)

> df

> d $salary[s]=m

> d

> normalize<-function(x) + {return(x-min(x)/(max(x)-min(x)))}

> df=data.frame(s=d $age,a=d $salary,stringsAsFactors = FALSE)

> df

> dfnor<-as.data.frame(lapply(df[1:2],normalize))

> dfnor

> dfnormz=as.data.frame(scale(df[1:2]))

> dfnormz

DATA INTEGRATION:

Cars

> corelation=cor.test(x=cars $speed,y=cars $dist,method="pearson")

> correlation

> var<-length(colnames(cars))

> var

> var<-length(colnames(cars))

> var

> var<-length(colnames(cars))

> var

> sxy<-sum(cars $speed\*cars $dist)-sum(cars $speed)\*sum(cars $dist)/n

> sxy

[1] 5387.4

> sxx<-sum(cars $speed\*cars ^2)-sum(cars $speed)^2/n

> sxx [1] 2674440 > sxx<-sum(cars $speed^2)-sum(cars $speed)^2/n

> sxx

[1] 1370

> syy<-sum(cars $dist^2)-sum(cars $dist)^2/n > syy

[1] 32538.98

> r=sxy/sqrt(sxx\*syy)

> r [1] 0.8068949

> t=(r\*sqrt(df)/sqrt(1-r^2))

> t

[1] 9.46399

**Chi-square Test:**

The chi-square test of independence is used to analyze the frequency table (i.e. contengency table) formed by two categorical variables. The chi-square test evaluates whether there is a significant association between the categories of the two variables.

>library(MASS)

>MASS

>tb1<-table(survey$Smoke,survey$Exer)

>tb1

>chisq.test(tb1)

>summary(tb1)

>ctb1<-cbind(tb1[,"Freq"],tb1[,"None"]+tb1[,"Some"])

>ctb1

>chisq.test(tb1)

Week-1

->Demomstration of preprocessing on datasets student.arff and labor.arff.

Steps of execution:

Step1: Loading the data. We can load the dataset into weka by clicking on open button in preprocessing interface and selecting the appropriate file.

Step2: Once the data is loaded, weka will recognize the attributes and during the scan of the data weka will compute some basic strategies on each attribute. The left panel in the above figure shows the list of recognized attributes while the top panel indicates the names of the base relation or table and the current working relation (which are same initially).

Step3: Clicking on an attribute in the left panel will show the basic statistics on the attributes for the categorical attributes the frequency of each attribute value is shown, while for continuous attributes we can obtain min, max, mean, standard deviation and deviation etc.,

Step4: The visualization in the right button panel in the form of cross-tabulation across two attributes. Note: we can select another attribute using the dropdown list

Step5: Selecting or filtering attributes Removing an attribute- When we need to remove an attribute, we can do this by using the attribute filters in weka. In the filter model panel, click on choose button, This will show a popup window with a list of available filters. Scroll down the list and select the “weka filters unsupervised Attribute remove” filters.

Step 6:

a) Next click the textbox immediately to the right of the choose button. In the resulting dialog box enter the index of the attribute to be filtered out.

b) Make sure that invert selection option is set to false. The click OK now in the filter box you will see “Remove-R-7”.

c) Click the apply button to apply filter to this data. This will remove the attribute and create new working relation.

d) Save the new working relation as an arff file by clicking save button on the top (button) panel(student.arff)

Dataset student .arff

@relation student

@attribute age {40}

@attribute income {low, medium, high}

@attribute student {yes, no}

@attribute credit-rating {fair, excellent}

@attribute buyspc {yes, no}

@data % 40, medium, no, fair, yes

>40, low, yes, fair, yes

>40, low, yes, excellent, no 30-40, low, yes, excellent, yes 40, medium, yes, fair, yes 40, medium, no, excellent, no %