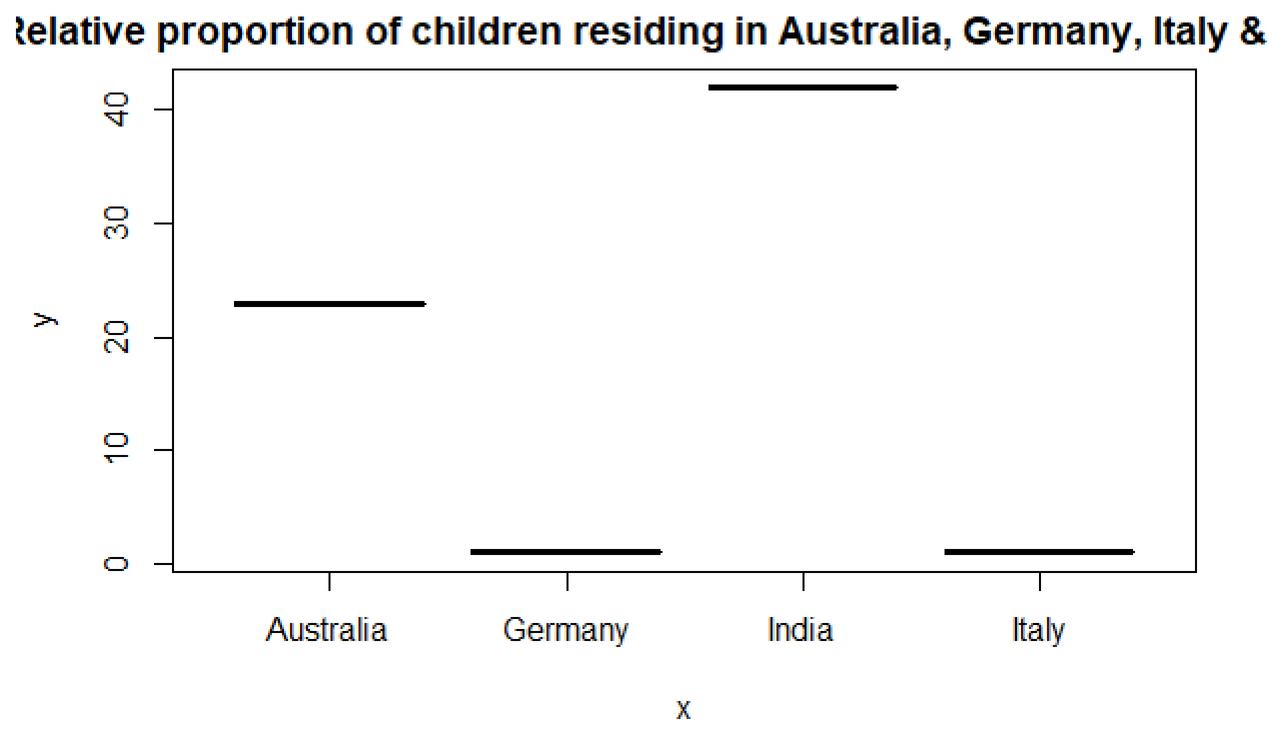
School of Computing Science and Digital Media
Coursework Assignment

Surname	Ugwumba
First name	Collins Nnamdi
Matriculation Number	1912840
Course	IT for the Oil and Gas Industry
Module Co-ordinator	Ines Arana
Module Number + Name	CMM020 Data Analysis & Visualisation
Coursework Title	CMM020 Coursework
Due Date	11 May, 2020

```
## Q1. Produce a plot with the relative proportion of children residing in
Australia, Germany, Italy and India.
```{r}
childplot <- read.csv("C:/Collins/childplot.csv")
plot(childplot$Country, childplot$Children, main="Relative proportion of
children residing in Australia, Germany, Italy & India")
```
```



```
#Q2
## Q2. univariate statistics on at least the first 4 attributes to describe the data.
```{r}
child <- read.csv("C:/Collins/child.csv")
child
```
```

score <dbl>	score2 <dbl>	age <int>	cost <dbl>	gender <fctr>	ethnicity <fctr>	jaundice <fctr>
4.6	4.4	5	1170.0	m	Others	no
4.4	4.4	5	1090.0	m	'Middle Eastern '	no
4.8	4.3	5	1130.0	m	NA	no
3.6	3.6	4	980.0	f	NA	yes
9.7	9.5	4	2475.0	m	Others	yes
4.9	4.5	3	1315.0	m	NA	no
7.3	7.1	4	1815.0	m	White-European	no
7.5	7.1	4	1955.0	f	'Middle Eastern '	no
6.7	6.4	2	1665.0	f	'Middle Eastern '	no
5.9	6.2	2	1445.0	f	NA	no

1-10 of 292 rows | 1-7 of 11 columns Previous 1 2 3 4 5 6 ... 30 Next

```
## To calculate the mean (average) for score, score2, age and cost:
```{r}
score <- child$score
mean(score)
score2 <- child$score2
mean(score2)
age <- child$age
mean(age)
cost <- child$cost
mean(cost)
```
```

```
[1] 6.394178
[1] 6.405479
[1] 4.19863
[1] 1951.241
```

```
## To obtain the median for score, score2, age and cost:
```{r}
median(score)
median(score2)
median(age)
median(cost)
```
```

```
[1] 6.5
[1] 6.5
[1] 4
[1] 1920
```

To obtain their sample standard deviation
```{r}
sd(score)
sd(score2)
sd(age)
sd(cost)
```

```
[1] 2.393117
[1] 2.401096
[1] 1.94643
[1] 778.2004
```

To obtain their population standard deviation
```{r}
sd(score)*sqrt((length(score)-1)/length(score))
sd(score2)*sqrt((length(score2)-1)/length(score2))
sd(age)*sqrt((length(age)-1)/length(age))
sd(cost)*sqrt((length(cost)-1)/length(cost))
```

```
[1] 2.389016
[1] 2.396981
[1] 1.943094
[1] 776.8667
```

To check their variance
```{r}
var(score)
var(score2)
var(age)
var(cost)
```

```
[1] 5.727011
[1] 5.765262
[1] 3.788589
[1] 605595.8
```

For their minimum and maximum values
```{r}
min(score)
max(score)
min(score2)
max(score2)
min(age)
max(age)
min(cost)
max(cost)
```

```
[1] 0
[1] 15
[1] 0
[1] 14
[1] 1
[1] 9
[1] -30
[1] 3840
```

And to obtain their range in one go
```{r}
range(score)
range(score2)
range(age)
range(cost)
```

```
[1]  0 15
[1]  0 14
[1] 1 9
[1]  -30 3840
```

To check their inter quartile range (3rd quantile minus first quantile)
```{r}
IQR(score)
IQR(score2)
IQR(age)
IQR(cost)
```

```
[1] 3.7
[1] 3.7
[1] 2
[1] 1205
```

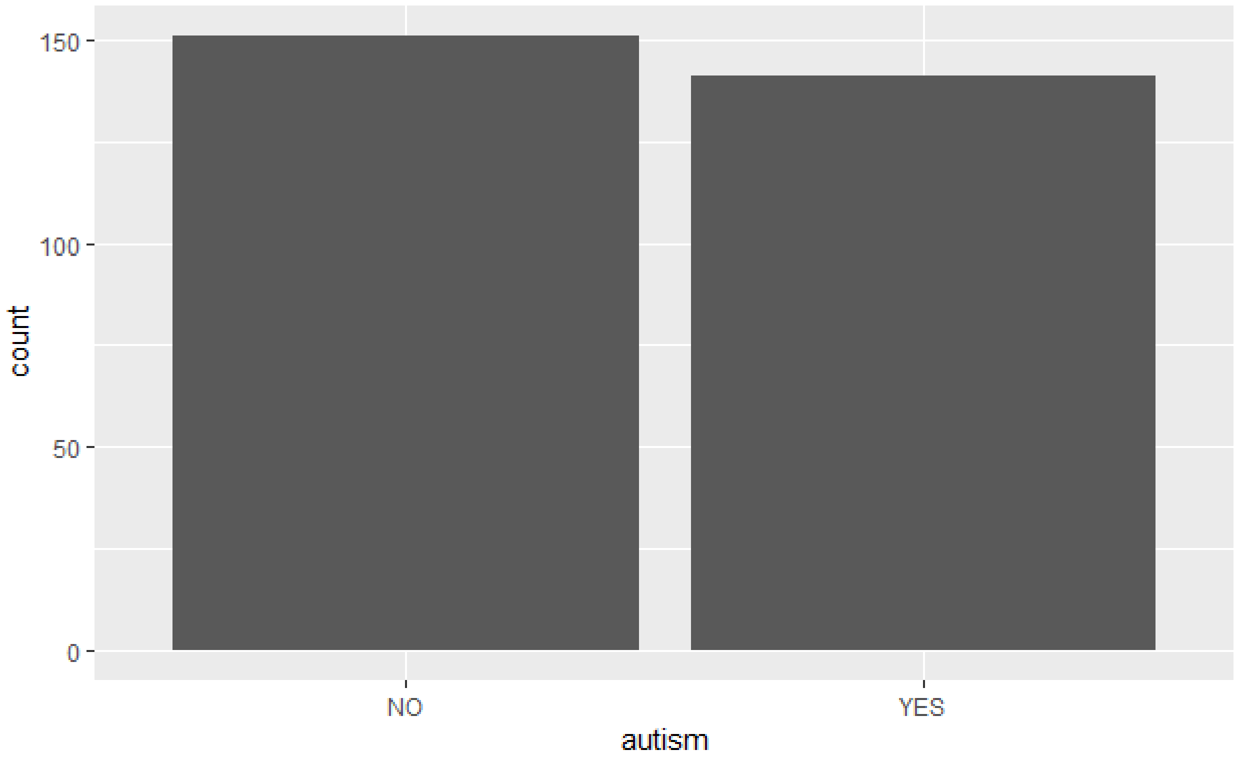
To obtain all their quantiles
```{r}
quantile(score)
quantile(score2)
quantile(age)
quantile(cost)
```

```
  0%  25%  50%  75% 100% 
 0.0  4.6  6.5  8.3 15.0 
  0%  25%  50%  75% 100% 
 0.0  4.6  6.5  8.3 14.0 
  0%  25%  50%  75% 100% 
   1    3    4    5    9 
  0%  25%  50%  75% 100% 
 -30 1360 1920 2565 3840 
```

To get their fivenum values
```{r}
fivenum(score)
fivenum(score2)
fivenum(age)
fivenum(cost)
```

```
[1]  0.0  4.6  6.5  8.3 15.0
[1]  0.0  4.6  6.5  8.3 14.0
[1] 1 3 4 5 9
[1]  -30 1360 1920 2570 3840
```

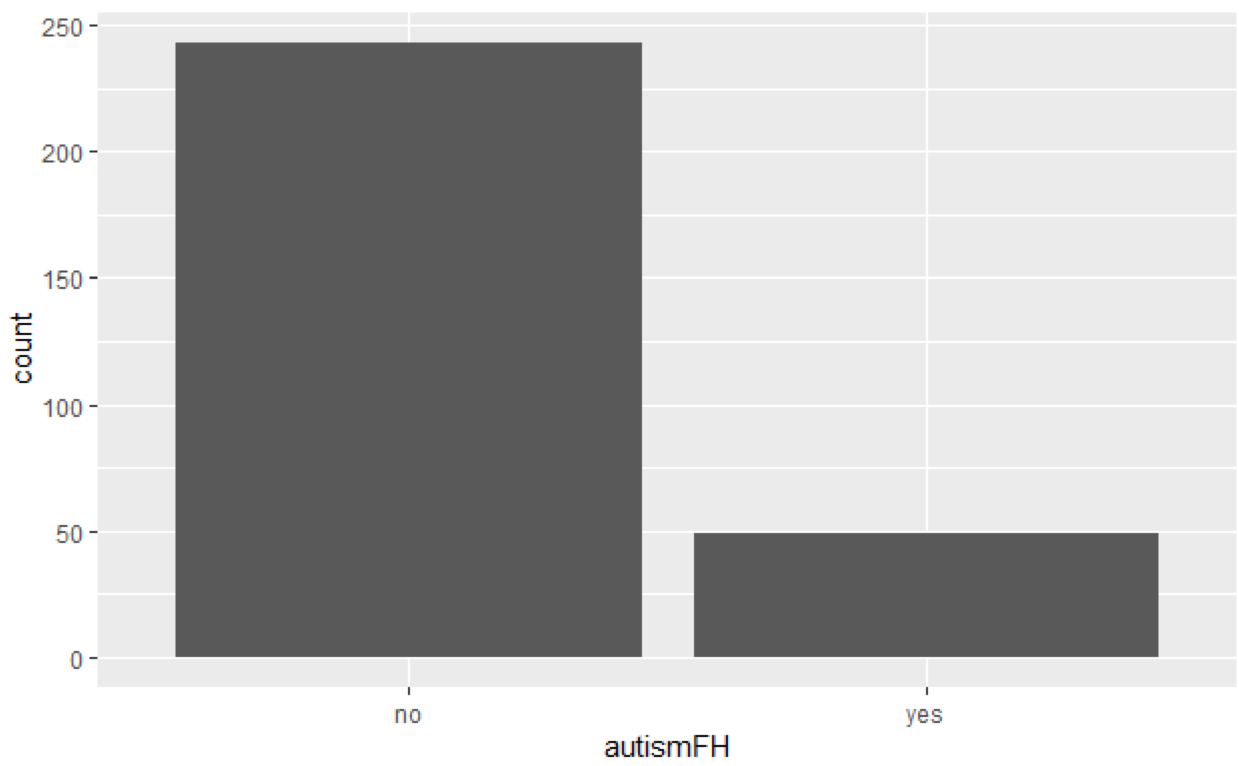
#Q3
Stacking
```{r}
p <- ggplot(child, aes(autism, fill = score))
p <- p + geom_bar(position = "stack")
p
```



3a values got from the plot above
```{r}
noAutism <- 150
autism <- 292-150
totalValue <- 292
meanAutism <- autism/totalValue
meanAutism
meannoAutism <- noAutism/totalValue
meannoAutism
significantValue <- meannoAutism - meanAutism
significantValue
```

```
[1] 0.4863014
[1] 0.5136986
[1] 0.02739726
```

Stacking for b
```{r}
p <- ggplot(child, aes(autismFH, fill = score))
p <- p + geom_bar(position = "stack")
p
```



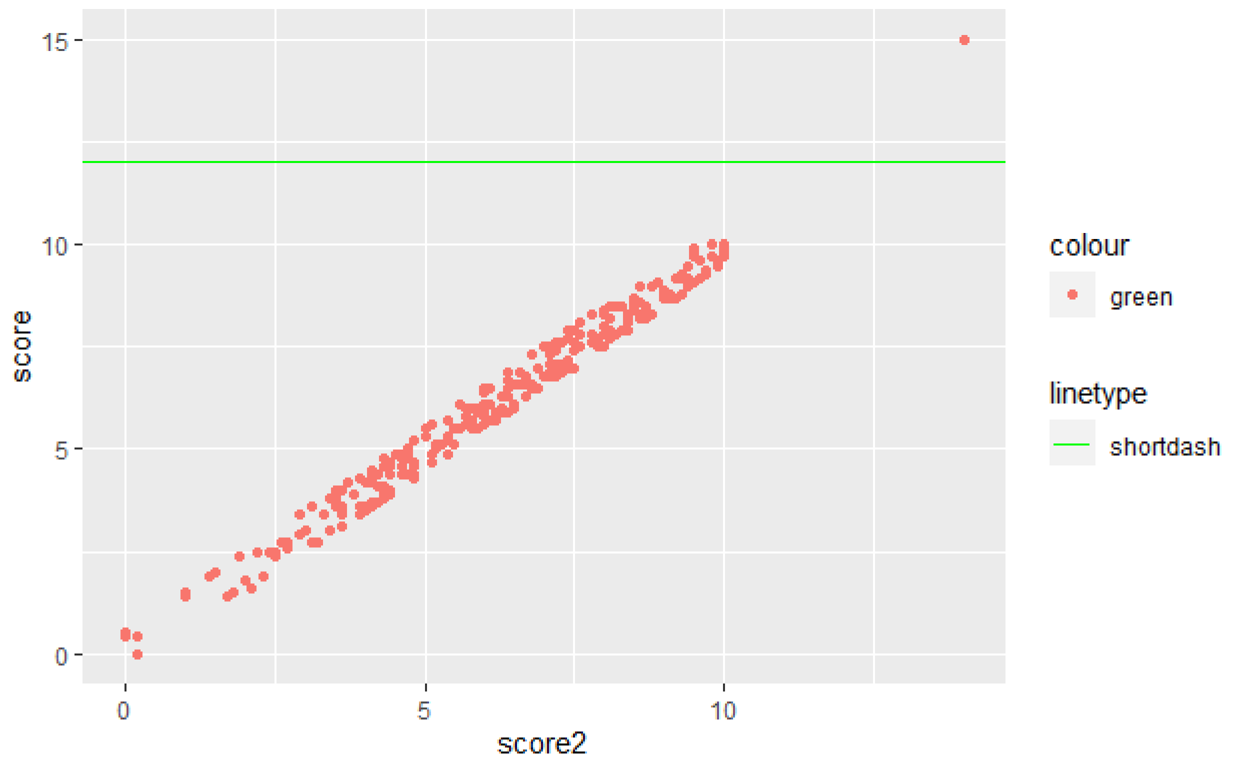
3b values got from the plot
```{r}
autFH <- 50
noAutFH <- 292 - 50
totalValueb <- 292
meanAutFH <- autFH/totalValueb
meanAutFH
meannoAutFH <- noAutFH/totalValueb
meannoAutFH
difference <- meannoAutism - meanAutism
difference
```

```
[1] 0.1712329
[1] 0.8287671
[1] 0.02739726
```

3c plot

```{r}
p <- ggplot(child, aes(score2, score))
p <- p + geom_point(aes(colour = "green"))
p <- p + geom_hline(aes (yintercept = 7, linetype = "shortdash"), colour =
"green")
p
```

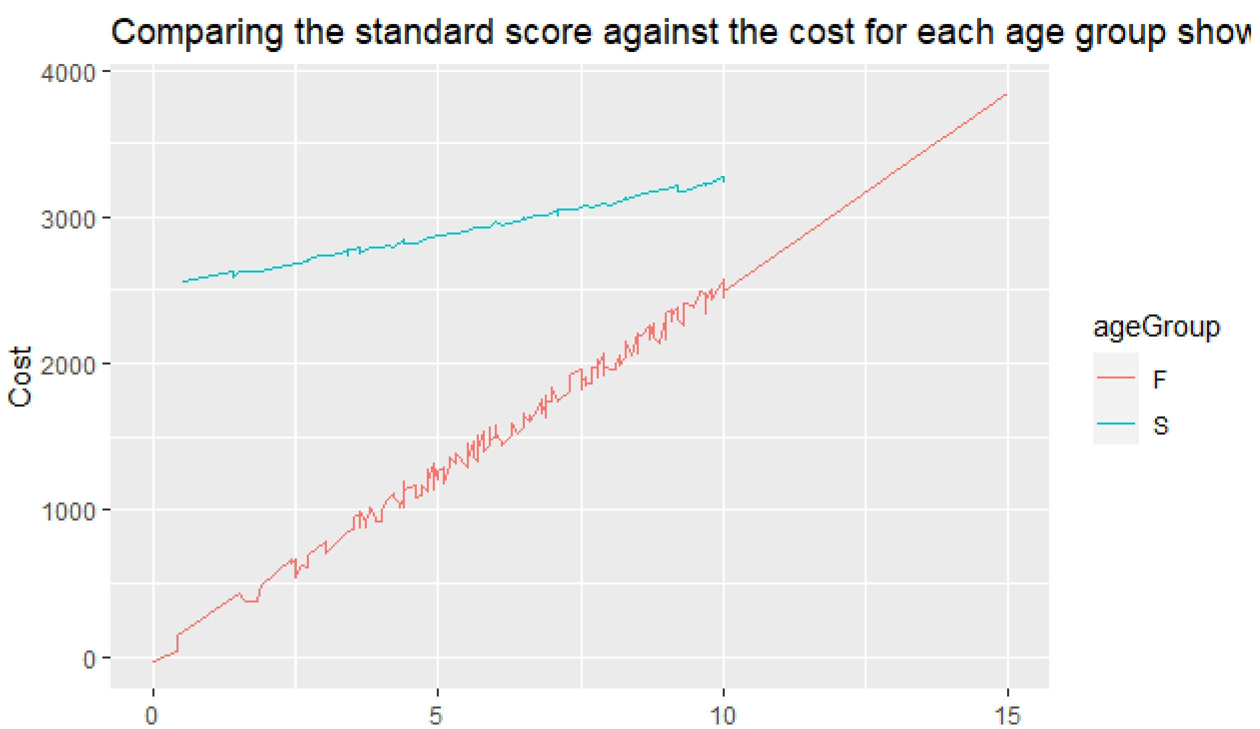
3d plot
```{r}
p <- ggplot(child, aes(score2, score))
p <- p + geom_point(aes(colour = "green"))
p <- p + geom_hline(aes (yintercept = 12, linetype = "shortdash"), colour =
"green")
p
```



Q4

```{r}
childfiveplot <- read.csv("C:/Collins/childfive.csv")

p <- ggplot(childfiveplot, aes(score, cost, autismFH, group=ageGroup))
p <- p + geom_line(aes(colour=factor(ageGroup)))
p <- p + labs(x="Standard Score", y="Cost" , title="Comparing the standard score against the cost for each age group showing whether there was a family history of autism", colour= "ageGroup")
p
```



```
# Q5
##generate sample data
```{r}
datasetnumber <- c(2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20)
Percentageofautism <-
c(51,47,48,45,44,49,54,50,51,53,50,47,48,50,51,50,49,50,52)
x <- datasetnumber*Percentageofautism/19
x
```

  [1] 5.368421 7.421053 10.105263 11.842105 13.894737 18.052632 22.736842
  [8] 23.684211 26.842105 30.684211 31.578947 32.157895 35.368421 39.473684
 [15] 42.947368 44.736842 46.421053 50.000000 54.736842

Q5
#Calculate the mean (mean)
```{r}
mean(x)
```

 [1] 28.84488

#sample standard deviation (sd)
```{r}
sd(x)
```

 [1] 15.11801

#minimum value (min)
```{r}
min(x)
```

 [1] 5.368421

#maximum value (max)
```{r}
max(x)
```

 [1] 54.73684

#degrees of freedom
```{r}
df <- length(x) -1
df
```

 [1] 18

#standard error
```{r}
SE <- sd(x)/sqrt(length(x))
SE
```

 [1] 3.468309
```

#Calculate the t-score

```{r}
t.score <- ((mean(x)) - 29)/SE
t.score
```

```
[1] -0.04472631
```

#The relationship between the minimum value and the mean

It is less than 2. So conducting a test is OK

```{r}
(mean(x)- min(x))/sd(x)
```

```
[1] 1.55288
```

#The relationship between the minimum value and the mean

It is less than 2. So conducting a test is OK

```{r}
(max(x) -mean(x))/sd(x)
```

```
[1] 1.712657
```

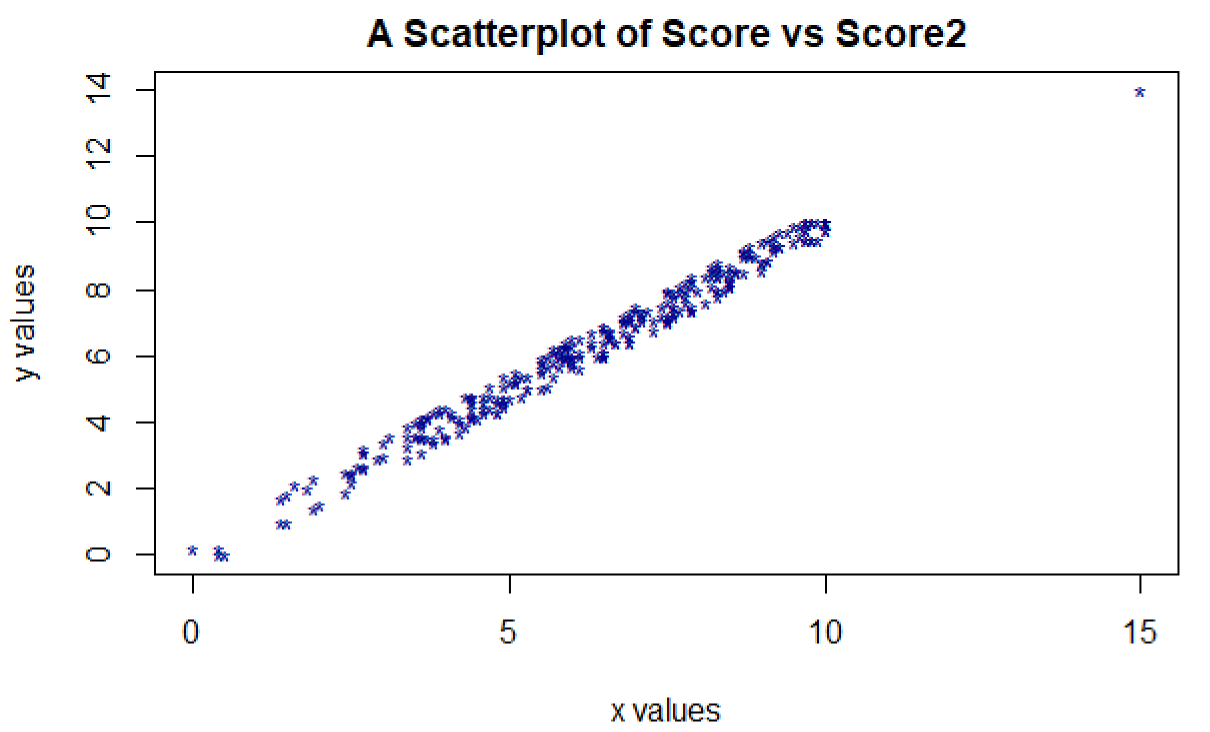
#Statistical inference

##Statistical inference

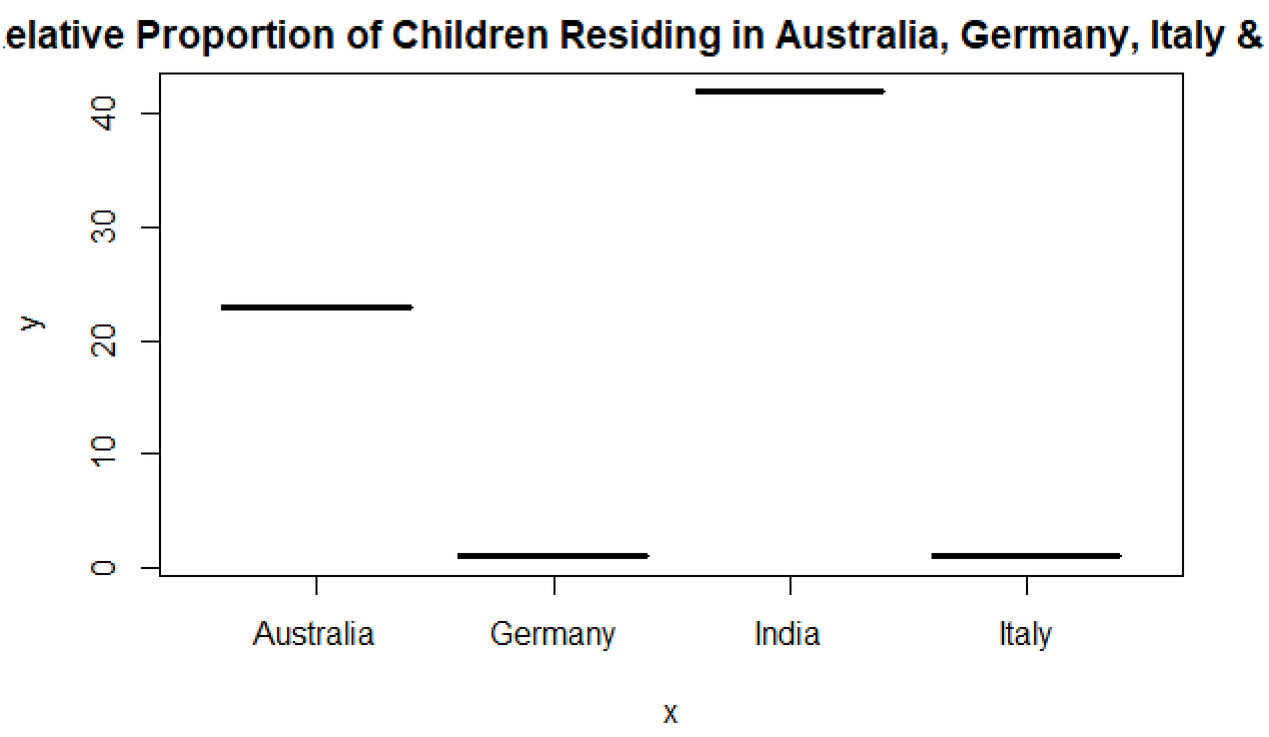
```{r}
t.test(x, conf.level=0.90, mu=29)
t.test(x, conf.level=0.95, mu=29)
t.test(x, conf.level=0.98, mu=29)
```

```
        One Sample t-test

data:  x
t = -0.044726, df = 18, p-value = 0.9648
alternative hypothesis: true mean is not equal to 29
90 percent confidence interval:
 22.83061 34.85914
sample estimates:
mean of x
 28.84488


        One Sample t-test

data:  x
t = -0.044726, df = 18, p-value = 0.9648
alternative hypothesis: true mean is not equal to 29
95 percent confidence interval:
 21.55823 36.13152
sample estimates:
mean of x
 28.84488


        One Sample t-test

data:  x
t = -0.044726, df = 18, p-value = 0.9648
alternative hypothesis: true mean is not equal to 29
98 percent confidence interval:
 19.99243 37.69732
sample estimates:
mean of x
 28.84488
```

Q6
Scatterplot - Bad Visualisation method
```{r}
plot(score, score2, main="A Scatterplot of Score vs Score2", xlab="x values",
ylab = "y values", col="darkblue", pch="*")
```



Boxplot 2 - Good Visualisation Method
```{r}
plot(childplot$Country, childplot$Children, main="Relative Proportion of
Children Residing in Australia, Germany, Italy & India")
```



```
## Barplot 3 - Good Visualisation Method
```{r}
p <- ggplot(child, aes(cost))
p <- p + geom_histogram(colour="darkgreen", fill="yellow", binwidth=1000)
p
```
```