PES University, Bangalore

Established under Karnataka Act No. 16 of 2013

UE20CS312 - Data Analytics - Worksheet 1b - Correlation Analysis

Ishita Bharadwaj, Dept. of CSE - PES1UG20CS648

Collaborated with Hita - PES1UG20CS645

Correlation

Correlation is a measure of the strength and direction of relationship that exists between two random variables and is measured using correlation coefficient. Correlation can assist data scientists to choose the variables for model building that is used for solving an analytics problem.

Solution 1

Find the total number of accidents in each state for the year 2016 and display your results. Make sure to display all rows while printing the dataframe. Print only the necessary columns. (Hint: use the grep command to help filter out column names).

```
library(ggpubr)
library(dplyr)
df <- read.csv('road_accidents_india_2016.csv', row.names=1)
```

```
acc_cols<-grep("Total.Accidents$",colnames(df),ignore.case = TRUE, value = TRUE)
print(acc_cols)
```

```
## [1] "Fine.Clear...Total.Accidents"  "Mist..Foggy...Total.Accidents"
## [3] "Cloudy...Total.Accidents"      "Rainy...Total.Accidents"
## [5] "Snowfall...Total.Accidents"    "Hail.Sleet...Total.Accidents"
## [7] "Dust.Storm...Total.Accidents"  "Others...Total.Accidents"
```

```
totalAccidents<-data.frame(state.ut=df$State..UT, total_acc=rowSums(df[,c(acc_cols)], na.rm = TRUE))
print(totalAccidents)
```

```
##             state.ut total_acc
## 0     Andhra Pradesh     24888
## 1  Arunachal Pradesh       249
## 2              Assam      7435
## 3              Bihar      8222
## 4       Chhattisgarh     13580
## 5                Goa      4304
## 6            Gujarat     21859
## 7            Haryana     11234
## 8   Himachal Pradesh      3168
## 9    Jammu & Kashmir      5501
## 10         Jharkhand      4932
## 11         Karnataka     44403
## 12            Kerala     39420
## 13    Madhya Pradesh     53972
## 14       Maharashtra     39878
## 15           Manipur       538
## 16         Meghalaya       620
## 17           Mizoram        83
## 18          Nagaland        75
## 19            Orissa     10532
## 20            Punjab      6952
## 21         Rajasthan     23066
## 22            Sikkim       210
## 23        Tamil Nadu     71431
## 24         Telangana     22811
## 25           Tripura       557
## 26       Uttarakhand      1591
## 27     Uttar Pradesh     35612
## 28       West Bengal     13580
## 29     A & N Islands       238
## 30        Chandigarh       428
## 31      D & N Haveli        70
## 32       Daman & Diu        71
## 33             Delhi      7375
## 34       Lakshadweep         1
## 35        Puducherry      1766
```

Solution 2

Find the (fatality rate $= \frac{\text{total number of deaths}}{\text{total number of accidents}}$) in each state. Find out if there is a significant linear correlation at a significance of $\alpha = 0.05$ between the *fatality rate* of a state and the *mist/foggy rate* (fraction of total accidents that happen in mist/foggy conditions).

Correlation between two continuous RVs: Pearson's correlation coefficient. Pearson's correlation coefficient between two RVs x and y is given by:

$$\rho = \frac{\text{Covariance}(x, y)}{\sigma_x \sigma_y}$$

where σ is the standard deviation of a variable.

Plot the fatality rate against the mist/foggy rate. (Hint: use the `ggscatter` library to plot a scatterplot with the confidence interval of the correlation coefficient).

Plot the fatality rate and mist/foggy rate (see this and this for R plot customization).

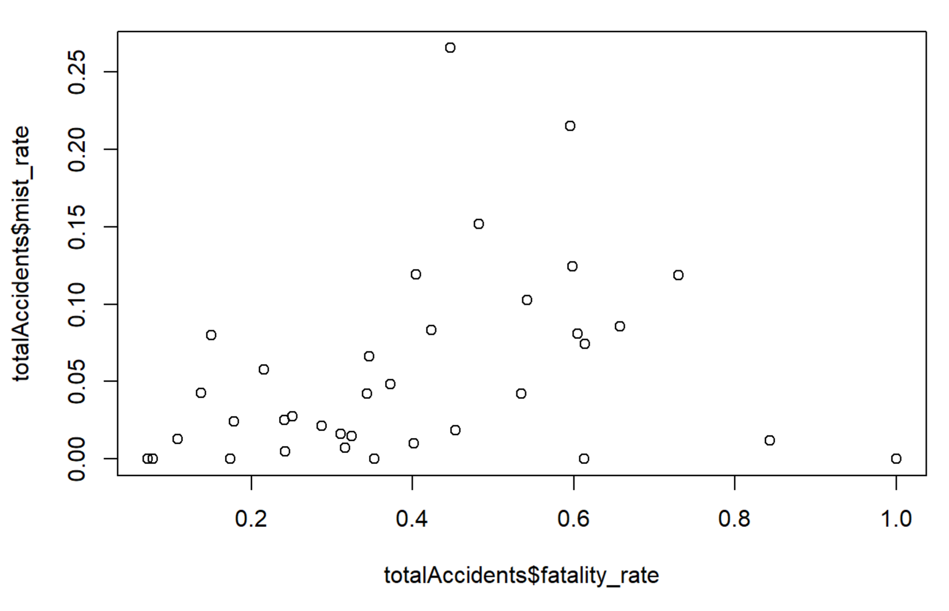
```
death_cols<-grep("Persons.Killed$",colnames(df),ignore.case = TRUE, value = TRUE)
print(death_cols)
```

```
## [1] "Fine.Clear...Persons.Killed"  "Mist..Foggy...Persons.Killed"
## [3] "Cloudy...Persons.Killed"      "Rainy...Persons.Killed"
## [5] "Snowfall...Persons.Killed"    "Hail.Sleet...Persons.Killed"
## [7] "Dust.Storm...Persons.Killed"  "Others...Persons.Killed"
```

```
totalAccidents$total_deaths<-NA
totalAccidents$fatality_rate<-NA
totalAccidents$mist_rate<-NA
totalAccidents$total_deaths<-c(rowSums(df[, c(death_cols)], na.rm=TRUE))
totalAccidents$fatality_rate<-c(totalAccidents$total_deaths/totalAccidents$total_acc)
totalAccidents$mist_rate<-df$Mist..Foggy...Total.Accidents/totalAccidents$total_acc
print(head(totalAccidents))
```

```
##            state.ut total_acc total_deaths fatality_rate  mist_rate
## 0    Andhra Pradesh     24888         8541    0.34317743 0.04222919
## 1 Arunachal Pradesh       249          149    0.59839357 0.12449799
## 2             Assam      7435         2572    0.34593141 0.06603900
## 3             Bihar      8222         4901    0.59608368 0.21515446
## 4      Chhattisgarh     13580         3908    0.28777614 0.02120766
## 5               Goa      4304          336    0.07806691 0.00000000
```

```
plot(x=totalAccidents$fatality_rate, y=totalAccidents$mist_rate)
```




```
r<-cor(totalAccidents$fatality_rate, totalAccidents$mist_rate,method='pearson')
print(r)
```

```
## [1] 0.2935159
```

```
corr_test<-cor.test(totalAccidents$fatality_rate, totalAccidents$mist_rate,method='pearson')
print(corr_test)
```

```
##
##  Pearson's product-moment correlation
##
## data:  totalAccidents$fatality_rate and totalAccidents$mist_rate
## t = 1.7903, df = 34, p-value = 0.08231
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.03875722  0.56734253
## sample estimates:
##       cor
## 0.2935159
```

Correlation coefficient is 0.29 which is not so high. This implies that the strength of linear relationship between fatality_rate and mist_rate is weak. This association is a weak association and not significant as p(0.08) is greater than alpha(0.05).

Solution 3

Rank the states based on total accidents and total fatalities (give a rank of 1 to the state that has the highest value of a property). You are free to use any tie-breaking method for assigning ranks.

Find the Spearman-Rank correlation coefficient between the two rank columns and determine if there is any statistical significance at a significance level of $\alpha = 0.05$. Also test the hypothesis that the correlation coefficient is at least 0.2.

The t statistic is given by

$$t = \frac{r_s - \rho_s}{\sqrt{\frac{1 - r_s^2}{n - 2}}}$$

Where r_s is the calculated Spearman-Rank correlation coefficient and ρ_s is the value of the population correlation coefficient being tested against.

```
totalAccidents$acc_ranks<-NA
totalAccidents$acc_ranks<-rank(desc(totalAccidents$total_acc),ties.method = 'random')
totalAccidents$death_ranks<-rank(desc(totalAccidents$total_deaths),ties.method = 'random')
print(data.frame(totalAccidents))
```

##	state.ut	total_acc	total_deaths	fatality_rate	mist_rate	acc_ranks
## 0	Andhra Pradesh	24888	8541	0.34317743	0.042229187	[illegible]
## 1	Arunachal Pradesh	249	149	0.59839357	0.124497992	[illegible]
## 2	Assam	7435	2572	0.34593141	0.066038999	[illegible]
## 3	Bihar	8222	4901	0.59608368	0.215154464	[illegible]
## 4	Chhattisgarh	13580	3908	0.28777614	0.021207658	[illegible]
## 5	Goa	4304	336	0.07806691	0.000000000	[illegible]
## 6	Gujarat	21859	[illegible]	[illegible]	[illegible]	[illegible]
## 7	Haryana	11234	[illegible]	[illegible]	[illegible]	[illegible]
## 8	Himachal Pradesh	3168	[illegible]	[illegible]	[illegible]	[illegible]
## 9	Jammu & Kashmir	5501	[illegible]	[illegible]	[illegible]	[illegible]
## 10	Jharkhand	4932	[illegible]	[illegible]	[illegible]	[illegible]
## 11	Karnataka	44403	[illegible]	[illegible]	[illegible]	[illegible]
## 12	Kerala	39420	[illegible]	[illegible]	[illegible]	[illegible]
## 13	Madhya Pradesh	53972	[illegible]	[illegible]	[illegible]	[illegible]
## 14	Maharashtra	39878	[illegible]	[illegible]	[illegible]	[illegible]
## 15	Manipur	538	[illegible]	[illegible]	[illegible]	[illegible]
## 16	Meghalaya	620	[illegible]	[illegible]	[illegible]	[illegible]
## 17	Mizoram	83	[illegible]	[illegible]	[illegible]	[illegible]
## 18	Nagaland	75	[illegible]	[illegible]	[illegible]	[illegible]
## 19	Orissa	10532	[illegible]	[illegible]	[illegible]	[illegible]
## 20	Punjab	6952	[illegible]	[illegible]	[illegible]	[illegible]
## 21	Rajasthan	23066	[illegible]	[illegible]	[illegible]	[illegible]
## 22	Sikkim	210	[illegible]	[illegible]	[illegible]	[illegible]
## 23	Tamil Nadu	71431	[illegible]	[illegible]	[illegible]	[illegible]
## 24	Telangana	22811	[illegible]	[illegible]	[illegible]	[illegible]
## 25	Tripura	557	[illegible]	[illegible]	[illegible]	[illegible]
## 26	Uttarakhand	1591	[illegible]	[illegible]	[illegible]	[illegible]
## 27	Uttar Pradesh	35612	[illegible]	[illegible]	[illegible]	[illegible]
## 28	West Bengal	13580	[illegible]	[illegible]	[illegible]	[illegible]
## 29	A & N Islands	238	[illegible]	[illegible]	[illegible]	[illegible]
## 30	Chandigarh	428	[illegible]	[illegible]	[illegible]	[illegible]
## 31	D & N Haveli	70	[illegible]	[illegible]	[illegible]	[illegible]
## 32	Daman & Diu	71	[illegible]	[illegible]	[illegible]	[illegible]
## 33	Delhi	7375	[illegible]	[illegible]	[illegible]	[illegible]
## 34	Lakshadweep	1	[illegible]	[illegible]	[illegible]	[illegible]
## 35	Puducherry	1766	[illegible]	[illegible]	[illegible]	[illegible]

##	death_ranks
## 0	[illegible]
## 1	[illegible]
## 2	[illegible]
## 3	[illegible]
## 4	[illegible]
## 5	[illegible]
## 6	[illegible]
## 7	[illegible]
## 8	[illegible]
## 9	[illegible]
## 10	[illegible]
## 11	4
## 12	15
## 13	6
## 14	3
## 15	30
## 16	27
## 17	31
## 18	33
## 19	14
## 20	11
## 21	5
## 22	29
## 23	2
## 24	9
## 25	25
## 26	21
## 27	1
## 28	10
## 29	35
## 30	26
## 31	32
## 32	34
## 33	19
## 34	36
## 35	24

```
rs<-cor(totalAccidents$acc_ranks, totalAccidents$death_ranks, method = 'spearman')
print(rs)
```

```
## [1] 0.957529
```

```
corr_test<-cor.test(totalAccidents$acc_ranks, totalAccidents$death_ranks, method = 'spearman')
print(corr_test)
```

```
##
##  Spearman's rank correlation rho
##
## data:  totalAccidents$acc_ranks and totalAccidents$death_ranks
## S = 330, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.957529
```

Yes, the correlation coefficient is alteast 0.2

```
degrees<-nrow(totalAccidents)-2
t_stat<-(rs-0.2)/sqrt((1-rs*rs)/(nrow(totalAccidents)-2))
2*pt(q=t_stat, df=degrees, lower.tail=FALSE)
```

```
## [1] 7.921248e-17
```

Solution 4

Convert the column `Hail.Sleet...Total.Accidents` to a binary column as follows. If a hail/sleet accident has occurred in a state, give that state a value of 1. Otherwise, give it a value of 0. Once converted, find out if there is a significant correlation between the `hail_accident_occcur` binary column created and the number of rainy total accidents for every state.

Calculate the point bi-serial correlation coefficient between the two columns. (Hint: it is equivalent to calculating the Pearson correlation between a continuous and a dichotomous variable. You could also use the `ltm` package's `biserial.cor` function).

```
df$hail_accident_occcur<-factor(ifelse(df$Hail.Sleet...Total.Accidents==0,0,1))
cat("Hail.Sleet...Total.Accidents:\n",df$Hail.Sleet...Total.Accidents)
```

```
## Hail.Sleet...Total.Accidents:
##  92 12 15 0 0 0 64 70 0 0 160 47 0 426 0 10 0 0 0 47 0 28 9 0 0 0 32 199 6 0 0 0 0 171 0 1
```

```
#hail_accident_occcur
print(df$hail_accident_occcur)
```

```
##  [1] 1 1 1 0 0 0 1 1 0 0 1 1 0 1 0 1 0 0 0 1 0 1 1 0 0 0 1 1 1 0 0 0 0 1 0 1
## Levels: 0 1
```

```
library(ltm)
bi<-biserial.cor(df$Rainy...Total.Accidents,df$hail_accident_occcur)
print(bi)
```

```
## [1] -0.1429725
```

Solution 5

Similar to in Problem 4, create a binary column to represent whether a dust storm accident has occurred in a state (1 = occurred, 0 = not occurred). Convert the two columns into a contingency table.

Calculate the phi coefficient of the two tables. (Hint: use the `psych` package).

```
df$dust_storm_occcur<-NA
df$dust_storm_occcur<-factor(ifelse(df$Dust.Storm...Total.Accidents==0,0,1))
cat("Dust.Storm...Total.Accidents:\n",df$Dust.Storm...Total.Accidents)
```

```
## Dust.Storm...Total.Accidents:
##  42 0 36 582 132 0 302 243 0 0 247 580 0 429 474 3 0 0 0 249 69 8 0 0 4 0 9 1521 196 0 0 0 0 122 0 3
```

```
#dust_storm_occcur:
print(df$dust_storm_occcur)
```

```
##  [1] 1 0 1 1 1 0 1 1 0 0 1 1 0 1 1 1 0 0 0 1 1 1 0 0 1 0 1 1 1 0 0 0 0 1 0 1
## Levels: 0 1
```

```
##Contingency Table For Hail accident occurance
table1 = table(df$hail_accident_occcur)
print.table(table1)
```

```
##
##  0  1
## 19 17
```

```
##Contingency Table for Dust Storm Occurance
table2=table(df$dust_storm_occcur)
print.table(table2)
```

```
##
##  0  1
## 16 20
```

```
##Contingency Table
library(psych)
conTable = table(df$hail_accident_occcur,df$dust_storm_occcur)
print(conTable)
```

```
##
##      0  1
##   0 14  5
##   1  2 15
```

```
##Phi Correlation Coefficient
phi(conTable)
```

```
## [1] 0.62
```