Actividad2.0

Cesar Vazquez

2022-11-03

Lectura del dataframe del menu de mc-donalds

```
M = read.csv("mc-donalds-menu.csv")
names(M)
```

```
##  [1] "Category"                      "Item"
##  [3] "Serving.Size"                  "Calories"
##  [5] "Calories.from.Fat"             "Total.Fat"
##  [7] "Total.Fat....Daily.Value."     "Saturated.Fat"
##  [9] "Saturated.Fat....Daily.Value." "Trans.Fat"
## [11] "Cholesterol"                   "Cholesterol....Daily.Value."
## [13] "Sodium"                        "Sodium....Daily.Value."
## [15] "Carbohydrates"                 "Carbohydrates....Daily.Value."
## [17] "Dietary.Fiber"                 "Dietary.Fiber....Daily.Value."
## [19] "Sugars"                        "Protein"
## [21] "Vitamin.A....Daily.Value."     "Vitamin.C....Daily.Value."
## [23] "Calcium....Daily.Value."       "Iron....Daily.Value."
```

0.- Elije una categoría como variable dependiente y acota tu base de datos.

1.- Determine dos variables como independientes y una como dependiente y realice los siguientes puntos:

```
M = M[M[, "Category"] == "Breakfast",]
M1 = M[,c(4,11,13)]
x = data.frame(M1)
x
```

```
##    Calories Cholesterol Sodium
## 1       300         260    750
## 2       250          25    770
## 3       370          45    780
## 4       450         285    860
## 5       400          50    880
## 6       430         300    960
## 7       460         250   1300
## 8       520         250   1410
## 9       410          35   1300
## 10      470          35   1420
## 11      430          30   1080
## 12      480          30   1190
## 13      510         250   1170
## 14      570         250   1280
## 15      460          35   1180
```

```
## 16      520           35   1290
## 17      410           30   1180
## 18      470           30   1290
## 19      540          280   1470
## 20      460          250   1250
## 21      400           35   1250
## 22      420           35   1030
## 23      550          265   1320
## 24      500           50   1320
## 25      620          275   1480
## 26      570           60   1480
## 27      670          295   1510
## 28      740          555   1560
## 29      800          555   1680
## 30      640           35   1590
## 31      690           35   1700
## 32     1090          575   2150
## 33     1150          575   2260
## 34      990           55   2170
## 35     1050           55   2290
## 36      350           20    590
## 37      520           50    930
## 38      300          115    790
## 39      150            0    310
## 40      460           15    370
## 41      290            5    160
## 42      260            5    115
```

a. La matriz de varianzas y covarianzas

```
covar = cov(x)
covar
```

```
##              Calories Cholesterol    Sodium
## Calories     49139.84    21398.78 100753.66
## Cholesterol  21398.78    29406.27  40893.82
## Sodium      100753.66    40893.82 259614.07
```

b. La matriz de correlación

```
corr = cor(x)
corr
```

```
##              Calories Cholesterol    Sodium
## Calories    1.0000000   0.5629271 0.8920308
## Cholesterol 0.5629271   1.0000000 0.4680298
## Sodium      0.8920308   0.4680298 1.0000000
```

c. Compara las matrices de correlación de pearson, de spearman y de kendall y menciona tus observaciones.

```
corrSpearman = cor(x,method = "spearman")
corrSpearman
```

```
##              Calories Cholesterol    Sodium
## Calories    1.0000000   0.5932814 0.8973882
## Cholesterol 0.5932814   1.0000000 0.5404475
## Sodium      0.8973882   0.5404475 1.0000000
```

```
corrKendall = cor(x,method = "kendall")
corrKendall
```

```
##              Calories Cholesterol    Sodium
## Calories    1.0000000   0.4517976 0.7525091
## Cholesterol 0.4517976   1.0000000 0.4253806
## Sodium      0.7525091   0.4253806 1.0000000
```

Vemos que las matrices de correlación y covarianza, tienen valores diferentes pero su signo son iguales, esto demuestra que en los tres métodos existe una correlación del mismo tipo (positiva).

2.- Desarrolle métodos de mínimos cuadrados y desglose cada elemento visto en clase.

```
fit <- lm(x$Cholesterol ~ x$Calories + x$Sodium)
summary(fit)
```

```
##
## Call:
## lm(formula = x$Cholesterol ~ x$Calories + x$Sodium)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -325.42  -82.68  -47.26  110.35  304.27
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -69.11274   59.56181  -1.160   0.2530
## x$Calories    0.55072    0.22557   2.441   0.0193 *
## x$Sodium     -0.05621    0.09814  -0.573   0.5701
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 144.7 on 39 degrees of freedom
## Multiple R-squared:  0.3226, Adjusted R-squared:  0.2878
## F-statistic: 9.286 on 2 and 39 DF,  p-value: 0.0005031
```

```
# y gorrito
df = x
y =
fit$coefficients[1]+fit$coefficients[2]*x$Cholesterol+fit$coefficients[3]
*x$Sodium
df["y"] = y
```

```
yprom = mean(y)
yprom
```

```
## [1] -53.00655
```

```
y1 = y - yprom
df["yi - ymedia"] = y1
```

```
scR = (y1)^2
df["SCR"] = scR
```

```
y2 = df$Calories - y
df["Yi - ^Yi"] = y2
```

```
sce = (y2)^2
df["sce"] = sce
```

```
# Yi - Ymedia
y3 = df$Calories - yprom
df["Yi - Ymedia"] = y3
```

```
# sct
sct = (y3)^2
df["sct"] = sct
```

```
df
```

```
##    Calories Cholesterol Sodium            y yi - ymedia          SCR
Yi - ^Yi
## 1       300         260    750   31.9160537   84.922603  7211.848572
268.0839
## 2       250          25    770  -98.6269249  -45.620375  2081.218631
348.6269
## 3       370          45    780  -88.1746674  -35.168118  1236.796501
458.1747
## 4       450         285    860   39.5008453   92.507395  8557.618132
410.4992
## 5       400          50    880  -91.0421333  -38.035584  1446.705619
491.0421
## 6       430         300    960   42.1405609   95.147111  9052.972673
387.8594
## 7       460         250   1300   -4.5069398   48.499610  2352.212168
464.5069
## 8       520         250   1410  -10.6901021   42.316448  1790.681738
530.6901
## 9       410          35   1300 -122.9113439  -69.904794  4886.680240
532.9113
## 10      470          35   1420 -129.6566119  -76.650062  5875.232028
599.6566
## 11      430          30   1080 -113.2986099  -60.292060  3635.132518
543.2986
## 12      480          30   1190 -119.4817723  -66.475223  4418.955211
599.4818
## 13      510         250   1170    2.8004339   55.806984  3114.419429
507.1996
```

```
## 14      570         250   1280   -3.3827284   49.623821  2462.523642
573.3827
## 15      460          35   1180 -116.1660758  -63.159526  3989.125733
576.1661
## 16      520          35   1290 -122.3492382  -69.342688  4808.408440
642.3492
## 17      410          30   1180 -118.9196666  -65.913117  4344.538974
528.9197
## 18      470          30   1290 -125.1028290  -72.096279  5197.873479
595.1028
## 19      540         280   1470    2.4588086   55.465358  3076.405976
537.5412
## 20      460         250   1250   -1.6964114   51.310138  2632.730295
461.6964
## 21      400          35   1250 -120.1008155  -67.094266  4501.640498
520.1008
## 22      420          35   1030 -107.7344908  -54.727941  2995.147528
527.7345
## 23      550         265   1320    2.6296213   55.636171  3095.383525
547.3704
## 24      500          50   1320 -115.7747828  -62.768233  3939.851083
615.7748
## 25      620         275   1480   -0.8568879   52.149662  2719.587234
620.8569
## 26      570          60   1480 -119.2612920  -66.254742  4389.690865
689.2613
## 27      670         295   1510    8.4711583   61.477708  3779.508586
661.5288
## 28      740         555   1560  148.8473512  201.853901 40744.997312
591.1526
## 29      800         555   1680  142.1020831  195.108633 38067.378619
657.8979
## 30      640          35   1590 -139.2124083  -86.205859  7431.450048
779.2124
## 31      690          35   1700 -145.3955707  -92.389021  8535.731186
835.3956
## 32     1090         575   2150  126.6974798  179.704030 32293.538233
963.3025
## 33     1150         575   2260  120.5143174  173.520867 30109.491343
1029.4857
## 34      990          55   2170 -160.8001740 -107.793624 11619.465428
1150.8002
## 35     1050          55   2290 -167.5454420 -114.538892 13119.157846
1217.5454
## 36      350          20    590  -91.2626136  -38.256064  1463.526426
441.2626
## 37      520          50    930  -93.8526617  -40.846112  1668.404860
613.8527
## 38      300         115    790  -50.1865020    2.820048     7.952669
350.1865
```

```
## 39      150           0    310  -86.5380181  -33.531468  1124.359367
236.5380
## 40      460          15    370  -81.6498797  -28.643330   820.440351
541.6499
## 41      290           5    160  -75.3528422  -22.346292   499.356787
365.3528
## 42      260           5    115  -72.8233667  -19.816817   392.706234
332.8234
##           sce Yi - Ymedia        sct
## 1    71869.00    353.0065  124613.62
## 2   121540.73    303.0065   91812.97
## 3   209924.03    423.0065  178934.54
## 4   168509.56    503.0065  253015.59
## 5   241122.38    453.0065  205214.93
## 6   150434.94    483.0065  233295.33
## 7   215766.70    513.0065  263175.72
## 8   281631.98    573.0065  328336.51
## 9   283994.50    463.0065  214375.07
## 10  359588.05    523.0065  273535.85
## 11  295173.38    483.0065  233295.33
## 12  359378.40    533.0065  284095.98
## 13  257251.40    563.0065  316976.38
## 14  328767.75    623.0065  388137.16
## 15  331967.35    513.0065  263175.72
## 16  412612.54    573.0065  328336.51
## 17  279756.01    463.0065  214375.07
## 18  354147.38    523.0065  273535.85
## 19  288950.53    593.0065  351656.77
## 20  213163.58    513.0065  263175.72
## 21  270504.86    453.0065  205214.93
## 22  278503.69    473.0065  223735.20
## 23  299614.33    603.0065  363616.90
## 24  379178.58    553.0065  305816.24
## 25  385463.28    673.0065  452937.82
## 26  475081.13    623.0065  388137.16
## 27  437620.41    723.0065  522738.47
## 28  349461.45    793.0065  628859.39
## 29  432829.67    853.0065  727620.17
## 30  607171.98    693.0065  480258.08
## 31  697885.76    743.0065  552058.73
## 32  927951.75   1143.0065 1306463.97
## 33 1059840.77   1203.0065 1447224.76
## 34 1324341.04   1043.0065 1087862.66
## 35 1482416.90   1103.0065 1216623.45
## 36  194712.69    403.0065  162414.28
## 37  376815.09    573.0065  328336.51
## 38  122630.59    353.0065  124613.62
## 39   55950.23    203.0065   41211.66
## 40  293384.59    513.0065  263175.72
```

```
## 41  133482.70    343.0065  117653.49
## 42  110771.39    313.0065   97973.10
```

```
library(plot3D)
scatter3D(x$Calories,x$Cholesterol,x$Sodium, pch = 19, cex = 1,colvar =
NULL, col="red",
          theta = 20, phi = 10, bty="b",
          xlab = "Calorias", ylab = "Colesterol", zlab = "Sodio",
main="Diagrama de dispersión Calories-Colesterol-Sodio ")
```