Design of experiment exercise

Augusta Mukam

25/01/2022

Describing the experiment

I generated a dataset of 25 entries by modifying the 11 xi values and getting the output value corresponding to them. The goal is to get the function F like F(X)=Y, X is a vector of 11 elements. We store it in a csv file

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
df =read.csv(file="user_e7268.csv", colClasses=c("NULL", NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA
df
```

```
##       x1   x2   x3     x4   x5     x6      x7      x8    x9    x10    x11
## 1  0.101 0.90 0.25 0.1600 0.39 0.3500 0.50000 0.70000 0.800 0.9000 1.0000
## 2  0.100 0.10 0.90 0.2500 0.16 0.3900 0.35000 0.50000 0.700 0.8000 0.9000
## 3  0.100 0.10 0.90 0.2500 0.60 0.9000 0.35000 0.50000 0.500 0.8000 0.9800
## 4  0.100 0.10 0.00 0.2500 0.60 0.7900 0.34500 0.89500 0.500 0.2000 0.8200
## 5  0.000 0.01 0.00 0.0525 0.06 0.0769 0.01345 0.04895 0.095 0.0182 0.0852
## 6  1.000 1.00 1.00 1.0000 1.00 1.0000 1.00000 1.00000 1.000 1.0000 1.0000
## 7  0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 8  0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 1.0000
## 9  0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 1.0000 0.0000
## 10 0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 1.000 0.0000 0.0000
## 11 0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 1.00000 0.000 0.0000 0.0000
## 12 0.000 0.00 0.00 0.0000 0.00 0.0000 1.00000 0.00000 0.000 0.0000 0.0000
## 13 0.000 0.00 0.00 0.0000 0.00 1.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 14 0.000 0.00 0.00 0.0000 1.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 15 0.000 0.00 0.00 1.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 16 0.000 0.00 1.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 17 0.000 1.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 18 1.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 19 1.000 1.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 20 0.000 0.00 1.00 1.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 21 0.000 0.00 0.00 0.0000 1.00 1.0000 0.00000 0.00000 0.000 0.0000 0.0000
## 22 0.000 0.00 0.00 0.0000 0.00 0.0000 1.00000 1.00000 0.000 0.0000 0.0000
```

```
## 23 0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 1.000 1.0000 0.0000
## 24 0.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 1.0000 1.0000
## 25 1.000 0.00 0.00 0.0000 0.00 0.0000 0.00000 0.00000 0.000 0.0000 1.0000
##             y
## 1  -0.4281779
## 2  -0.1386140
## 3   0.2632307
## 4   0.2611612
## 5   0.8832912
## 6  -0.8763420
## 7   1.0152508
## 8   1.0145791
## 9   1.0131179
## 10 -0.9812617
## 11  1.0135006
## 12  1.0130075
## 13  1.0156932
## 14  1.0118228
## 15  2.0126428
## 16  1.0140150
## 17  1.0161708
## 18  0.4207322
## 19  0.4248749
## 20  2.0143936
## 21  1.0154428
## 22  1.0107968
## 23 -0.9832606
## 24  1.0147940
## 25  0.4172822
```

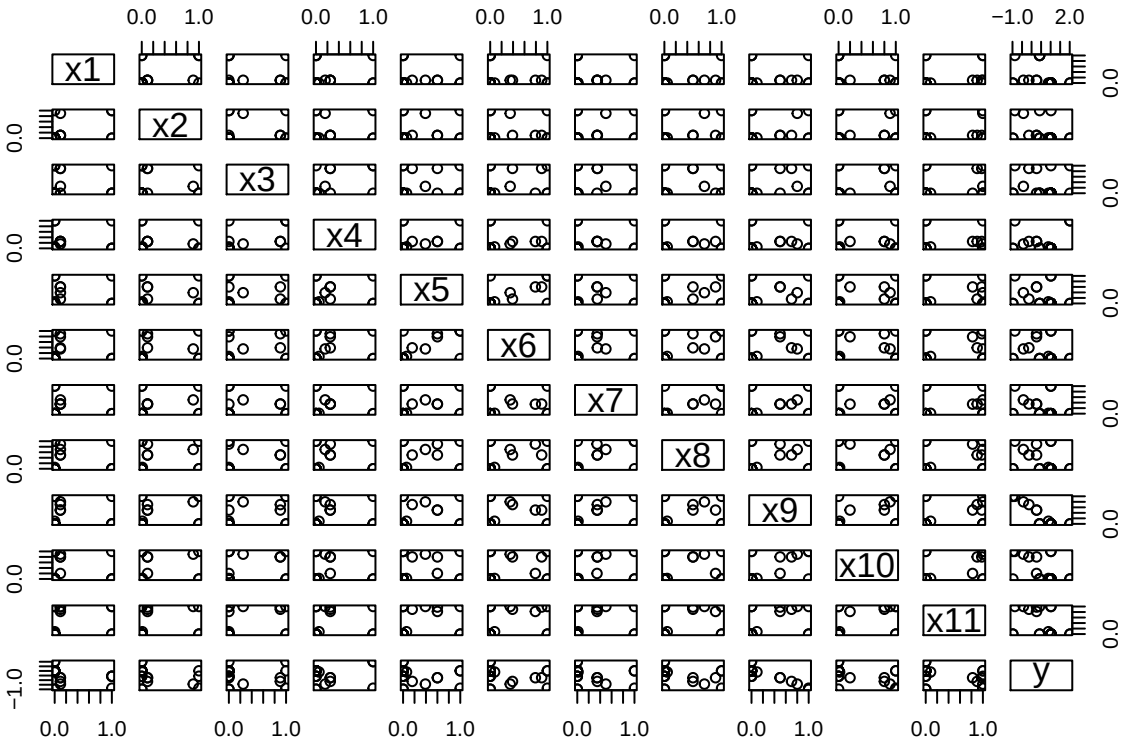
First, we figure out that the intercept F(0) is 1.0152508.

We can use a linear regression for a sanity check and we see that we obtain exactly the same result. This linear regression will also allow us to predict.

```
form <- as.formula(paste("y~", paste0("x", 1:11, collapse="+")))
reg=lm(form,data=df)
reg
```

```
##
## Call:
## lm(formula = form, data = df)
##
## Coefficients:
## (Intercept)           x1           x2           x3           x4           x5
##    1.044933    -0.663815    -0.011483     0.004691     0.933063    -0.039211
##          x6           x7           x8           x9          x10          x11
##    0.003296    -0.050583     0.010876    -1.998210    -0.026140     0.035827
```

```
plot(df)
```


x1
x2
x3
x4
x5
x6
x7
x8
x9
x10
x11
y
0.0 1.0
−1.0 2.0
0.0
−1.0