lab1 summary

2025-10-07

This is a summary of lab 1.

1 install packages

```
options(repos = c(CRAN = "https://cloud.r-project.org/"))
install.packages(c("corrplot"))
library("corrplot")
```

```
## corrplot 0.95 loaded
```

“install.packages” allows us to install the packages we need. If we write “install.packages(c(package a,package b…))”, then we can install multiple packages at a time.

We can use “installed.packages()” to check the packages that are already installed.

```
require("corrplot")
```

To check if a package is successfully installed, use “require()”.

After a package is installed, we need “library()” to load the package.

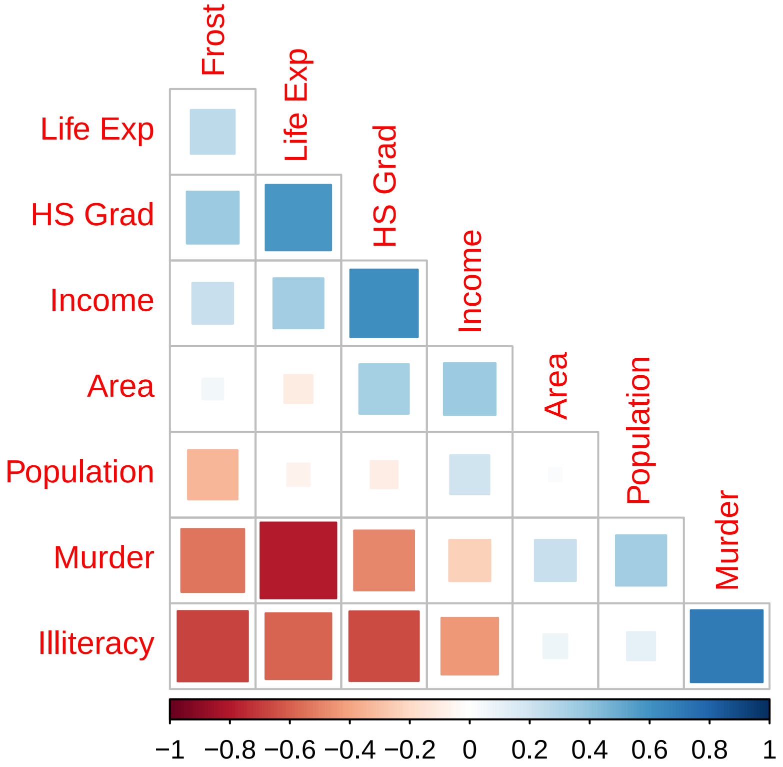
2 corrplot

In lab 1, we use the package corrplot. “An Introduction to corrplot Package” is an excellent article. The following is an excerpt.

R package corrplot provides a visual exploratory tool on correlation matrix. It has about 50 parameters, but the commonly used ones are just a few. The mostly used parameters include method, type, order, diag, and etc.

More detailed introduction can be found in the article.

```
R=cor(state.x77)
corrplot(R,method="square",diag=F,type="lower",order="AOE")
```



# 3 correlation test

```
cor.test(state.x77[,"Murder"],state.x77[,"Frost"],alternative="greater",method="pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  state.x77[, "Murder"] and state.x77[, "Frost"]
## t = -4.4321, df = 48, p-value = 1
## alternative hypothesis: true correlation is greater than 0
## 95 percent confidence interval:
##  -0.6871348  1.0000000
## sample estimates:
##        cor
## -0.5388834
```

```
#cor.test(~Murder+Frost,data=state.x77)
```

cor.test() provides t-test to check if the correlation coefficient is 0, using statistics $\sqrt{n-2}\frac{r}{\sqrt{1-r^2}}$(Pearson). "method" also provides Kendall or Spearman. "alternative" stands for the alternative hypothesis, where the default is "two.sided", standing for $\rho \neq 0$, and "greater" stands for $\rho > 0$, "less" stands for $\rho < 0$.

An equivalent expression is "cor.test(~A+B,data=C)", which means the test is conducted on A and B from data C.

4 generate data

In R, there are four functions that operate on any distribution.

1.density function:d+distribution(x) gives the density of the distribution at point x.

```
dnorm(0)
```

```
## [1] 0.3989423
```

2.distribution function:p+distribution(x) gives the CDF of the distribution at x.

```
pnorm(0)
```

```
## [1] 0.5
```

3.quantile function:q+distribution(x) gives the quantile,i.e. $qX(p) = x \Leftrightarrow P(X \leq x) = p$.

```
qnorm(0.95)
```

```
## [1] 1.644854
```

4.sample function:r+distribution generates random samples following the distribution.

```
rnorm(10)
```

```
##  [1]  2.64028295  0.60812867  0.20909411 -0.29440210  1.46789721  1.51712406
##  [7]  1.48255294  1.06975538  0.13435995 -0.08701695
```

Besides normal distribution, we can also generate other common distributions:

```
runif(10,-1,1)
```

```
##  [1]  0.7819024 -0.8749350 -0.2944525  0.8555369 -0.7084971 -0.8948389
##  [7]  0.3856123 -0.6216308  0.2372748 -0.5980277
```

```
rbinom(n = 5, size = 10, prob = 0.5)#conduct 5 experiments, each time toss the coin 10 times with equal
```

```
## [1] 4 5 2 3 6
```

```
rchisq(n = 10, df = 5)
```

```
##  [1] 1.195760 1.872737 7.336251 5.257300 4.806635 5.435835 6.282295 2.554950
##  [9] 2.253017 4.599618
```