Correlation. ANOVA

Linguistic data: Quantitative analysis and vizualization

Part1:

Part 1: Pearson’s coefficient vs Spearman’s coefficient

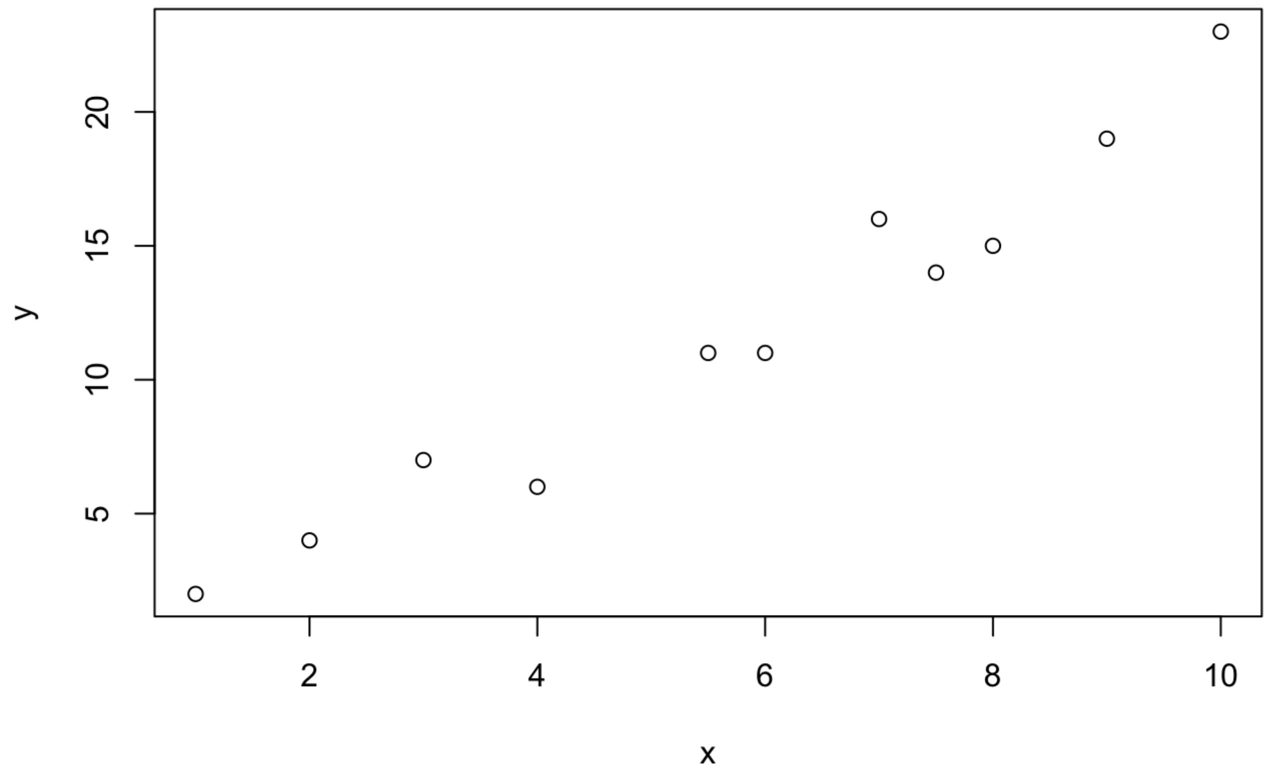
As we discussed, there are two widely used correlation coefficients, a Pearson’s one and a Spearman’s one. Since the latter is a measure of the rank correlation, it is usually used for variables in an ordinal scale. However, it can be helpful for quantitative variables as well because it is robust (not sensitive to outliers).

Consider two variables: `x` and `y`.

```
x <- c(1, 2, 6, 8, 9, 7, 7.5, 10, 3, 4, 5.5)
y <- c(2, 4, 11, 15, 19, 16, 14, 23, 7, 6, 11)
```

Let’s plot a simple scatterplot first:

```
plot(x, y)
```



As we can see, although there are only few points, variables `x` and `y` seem to be positively associated (as `x` increases, `y` increases). We can even say that this association is pretty strong. Let’s calculate two correlation coefficients and test their statistical significance.

```r
# Pearson's coefficient
cor.test(x, y)
```

```
## 
##  Pearson's product-moment correlation
## 
## data:  x and y
## t = 13.862, df = 9, p-value = 2.234e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9124984 0.9942928
## sample estimates:
##       cor 
## 0.9773737
```

What can we see in the output? The correlation coefficient itself is `cor` and here it is 0.977. So, we can conclude that the association between `x` and `y` is positive and very strong (the coefficient is approximately 1). Is it statistically significant at the 5% level of significance? Let us see.

$H_0 : corr(x, y) = 0$ (no linear association between x and y)

This null hypothesis should be rejected at the 5% significance level since p-value < 0.05. So, variables `x` and `y` are associated.

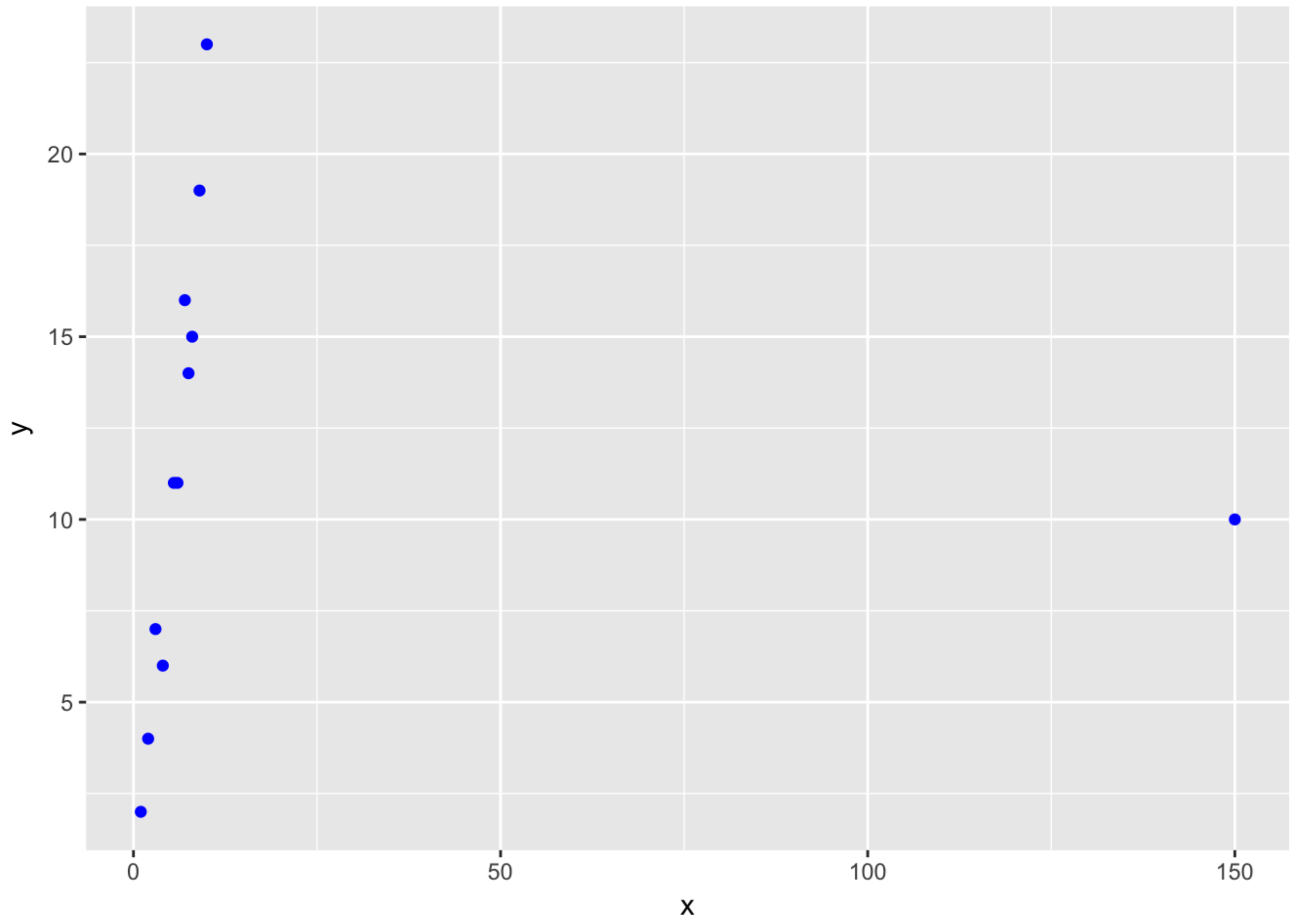
```r
# Spearman's coefficient
cor.test(x, y, method = 'spearman')
```

```
## Warning in cor.test.default(x, y, method = "spearman"): Cannot compute
## exact p-value with ties
```

```
## 
##  Spearman's rank correlation rho
## 
## data:  x and y
## S = 8.5188, p-value = 2.449e-06
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho 
## 0.9612781
```

Here we also get a very high positive coefficient (0.96). Now let us add an outlier, a non-typical observation to our data, a point (150, 10).

```r
x <- c(1, 2, 6, 8, 9, 7, 7.5, 10, 3, 4, 5.5, 150)
y <- c(2, 4, 11, 15, 19, 16, 14, 23, 7, 6, 11, 10)
# plot(x, y)
suppressMessages(library(ggplot2))
ggplot() +
    geom_point(aes(x = x, y = y), color = 'blue')
```



It seems that this point can spoil everything! We can calculate correlation coefficient for updated variables:

```
cor.test(x, y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  x and y
## t = -0.033164, df = 10, p-value = 0.9742
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.5808924  0.5668262
## sample estimates:
##         cor
## -0.01048683
```

A Pearson's correlation coefficient has broken down! Now it is negative, very small by absolute value and, what is more, insignificant! This coefficient is very sensitive to outliers, so here it "reacts" on a non-typical point in a very dramatic way. Now let's look at a Spearman's coefficient:

```
cor.test(x, y, method = 'spearman')
```

```
## Warning in cor.test.default(x, y, method = "spearman"): Cannot compute
## exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  x and y
## S = 64.613, p-value = 0.003127
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.7740817
```

Magic! This coefficient has not undergone serious changes, it is still positive and high. Besides, it is significant at the 5% significance level. So, with the help of this illustration we made sure that a Spearman's correlation coefficient is more robust than Pearson's one.

```
cor.test(x, y, method = 'kendall')
```

```
## Warning in cor.test.default(x, y, method = "kendall"): Cannot compute exact
## p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  x and y
## z = 3.093, p-value = 0.001981
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## 0.6870429
```

Part 2: try yourselves

Here you are suggested to work with a dataset on Chekhov's stories (`chekhov.csv` , link (https://raw.githubusercontent.com/LingData2019/LingData2020/master/data/chekhov.csv)).

Variables

- `n_words` : number of words in a
- `n_unique` : number of unique words in a

1. How do you feel: is there a linear relationship between the number of words and the number of unique words?

2. Plot a scatterplot for these variables and check whether your intuition was true. Interpret the scatterplot obtained.

3. Check using a proper statistical test, whether `n_words` and `n_unique` are associated: formulate a null hypothesis, test it and make conclusions.

Part 3: Universal linguistic hierarchies: a case of Modern Greek (Standard and Cypriot dialects)

Based on: Leivada, Evelina; Westergaard, Marit, 2019, Universal linguistic hierarchies are not innately wired. PeerJ, v.7. link (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6679903/#fn-1). Source of data: TROLLing repository: Leivada, Evelina; Westergaard, Marit, 2019, "Replication Data for: Universal linguistic hierarchies are not innately wired”, https://doi.org/10.18710/NTLLUF (https://doi.org/10.18710/NTLLUF), DataverseNO, V1

Constructions with two adjectives

In English, the order of two adjectives in phrases like:

```
 a big black bag # ok
*a black big bag # unacceptable, ungrammatically ill-formed, or semantically anomalous
```

is powered by the semantic class of adjective (e.g. the `color` adjective closer to the noun than the `size` adjective).

A syntactic hierarchy of closeness to the noun in Chomsky’s Universal Grammar suggests the following order and is claimed to be innate and universal (= valid for all languages).

```
Subjective Comment > Evidential > Size > Length
> Height > Speed > Depth > Width > Temperature > Wetness > Age
> Shape > Color > Nationality/Origin > Material
# (adapted from Scott, 2002: 114)
```

The goal of Leivada&Westergaard research is identify what happens when people process orderings that either comply with the hierrarchy or violate it.

Method

Experiment 1 140 neurotypical, adult speakers completed a timed forced choice task that featured stimuli showing a combination of two adjectives and a concrete noun (e.g., ‘I bought a square black table’). Two types of responses were collected: (i) acceptability judgments on a 3-point Likert scale that featured the options
3: ‘correct’,
2: ‘neither correct nor wrong’,
1: ‘wrong’
and (ii) reaction times (RT). The task featured three conditions: 1. size adjective > nationality adjective, 2. color adjective > shape adjective, 3. subjective comment adjective > material adjective. Each condition had two orders. In the congruent order, the adjective pair was ordered in agreement with what is traditionally accepted as dictated by the universal hierarchy. In the incongruent order, the ordering was reversed, thus the hierarchy was violated.

In the second experiment, 30 bidialectals (native speakers of Standard and Cypriot Greek) were tested in both language varieties, 36 observations per participant, 18 for each variety.

Data

```
suppressMessages(library(readr))
mono <- read_delim("https://dataverse.no/api/access/datafile/:persistentId?persistentId=doi:10.18710/NTLLUF/XAMMNB", delim = ";")
bidialect <- read_delim("https://dataverse.no/api/access/datafile/:persistentId?persistentId=doi:10.18710/NTLLUF/PQAKFW", delim=";")
```

see also reading key for the data (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6679903/bin/peerj-07-7438-s001.txt)

Data overview

1.1

Create the following table using filter and by-group summary in tidyverse.

Demographic information for monolinguals (experiment 1).

```
   Participants
Gender     Male       66
           Female       74
Education    Secondary  18
           Tertiary  122
Handedness Right       126
           Left       14
```

1.2

Create the same summary for experiment 2.

Demographic information for bidialectals (experiment 2).

```
Participants
Gender     Male       11
           Female       19
Education    Secondary   6
           Tertiary   24
Handedness Right        28
           Left        2
```

1.3

Create a plot that shows RT distribution in experiment 1 (all participants and conditions taken together).

Do reaction times follow a normal distribution? Or do they skewed and have long left or right tails?

1.4

Normalise data applying the standard logarithm (RTlog = log10(RT))

1.5

Create a plot similar to that in 1.3 that shows RTlog distribution.

1.6

Filter out outliers: * automatic responces below 600 ms (i.e., when a button is pressed too fast, without allowing enough time for actual consideration of the presented stimuli) * usong ±3SD filter (SD - standard deviation)

Homework assignment

Reproduce Figure 1 - Figure 7 from the paper using ggplot2. Run ANOVA.

R code cookbook

Correlation matrix for multiple variables (dataframe), cource (https://www.r-bloggers.com/spearman-correlation-heat-map-with-correlation-coefficients-and-significance-levels-in-r/)

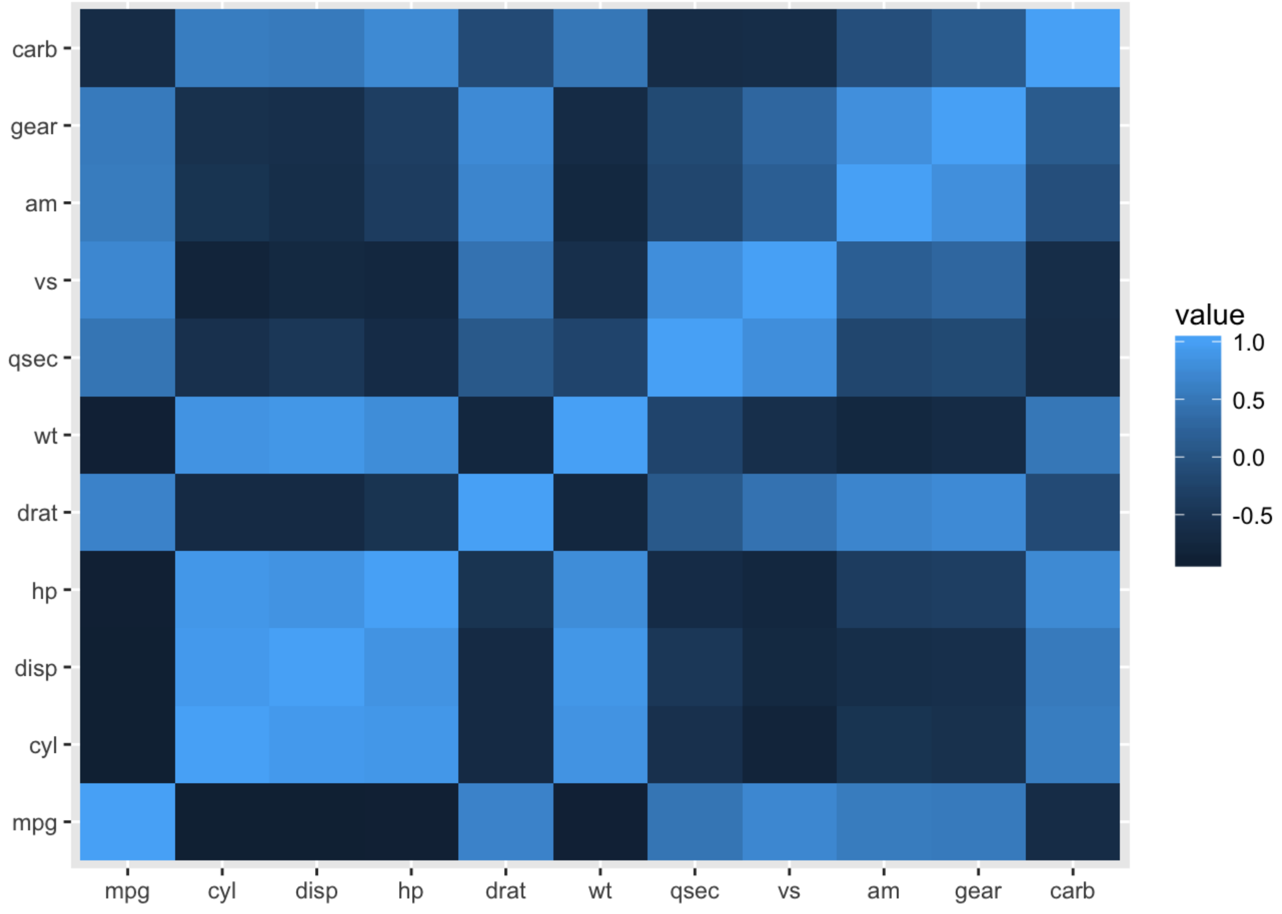
```
suppressMessages(library(Hmisc))
suppressMessages(library(reshape2))
cormatrix <- rcorr(as.matrix(mtcars), type='spearman')
cormatrix
```

```
##        mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
## mpg   1.00 -0.91 -0.91 -0.89  0.65 -0.89  0.47  0.71  0.56  0.54 -0.66
## cyl  -0.91  1.00  0.93  0.90 -0.68  0.86 -0.57 -0.81 -0.52 -0.56  0.58
## disp -0.91  0.93  1.00  0.85 -0.68  0.90 -0.46 -0.72 -0.62 -0.59  0.54
## hp   -0.89  0.90  0.85  1.00 -0.52  0.77 -0.67 -0.75 -0.36 -0.33  0.73
## drat  0.65 -0.68 -0.68 -0.52  1.00 -0.75  0.09  0.45  0.69  0.74 -0.13
## wt   -0.89  0.86  0.90  0.77 -0.75  1.00 -0.23 -0.59 -0.74 -0.68  0.50
## qsec  0.47 -0.57 -0.46 -0.67  0.09 -0.23  1.00  0.79 -0.20 -0.15 -0.66
## vs    0.71 -0.81 -0.72 -0.75  0.45 -0.59  0.79  1.00  0.17  0.28 -0.63
## am    0.56 -0.52 -0.62 -0.36  0.69 -0.74 -0.20  0.17  1.00  0.81 -0.06
## gear  0.54 -0.56 -0.59 -0.33  0.74 -0.68 -0.15  0.28  0.81  1.00  0.11
## carb -0.66  0.58  0.54  0.73 -0.13  0.50 -0.66 -0.63 -0.06  0.11  1.00
##
## n= 32
##
##
## P
##      mpg    cyl    disp   hp     drat   wt     qsec   vs     am     gear
## mpg         0.0000 0.0000 0.0000 0.0000 0.0000 0.0071 0.0000 0.0008 0.0013
## cyl  0.0000        0.0000 0.0000 0.0000 0.0000 0.0006 0.0000 0.0022 0.0008
## disp 0.0000 0.0000        0.0000 0.0000 0.0000 0.0081 0.0000 0.0001 0.0003
## hp   0.0000 0.0000 0.0000        0.0023 0.0000 0.0000 0.0000 0.0416 0.0639
## drat 0.0000 0.0000 0.0000 0.0023        0.0000 0.6170 0.0102 0.0000 0.0000
## wt   0.0000 0.0000 0.0000 0.0000 0.0000        0.2148 0.0004 0.0000 0.0000
## qsec 0.0071 0.0006 0.0081 0.0000 0.6170 0.2148        0.0000 0.2644 0.4182
## vs   0.0000 0.0000 0.0000 0.0000 0.0102 0.0004 0.0000        0.3570 0.1170
## am   0.0008 0.0022 0.0001 0.0416 0.0000 0.0000 0.2644 0.3570        0.0000
## gear 0.0013 0.0008 0.0003 0.0639 0.0000 0.0000 0.4182 0.1170 0.0000
## carb 0.0000 0.0005 0.0014 0.0000 0.4947 0.0036 0.0000 0.0000 0.7264 0.5312
##      carb
## mpg  0.0000
## cyl  0.0005
## disp 0.0014
## hp   0.0000
## drat 0.4947
## wt   0.0036
## qsec 0.0000
## vs   0.0000
## am   0.7264
## gear 0.5312
## carb
```

```
str(cormatrix)
```

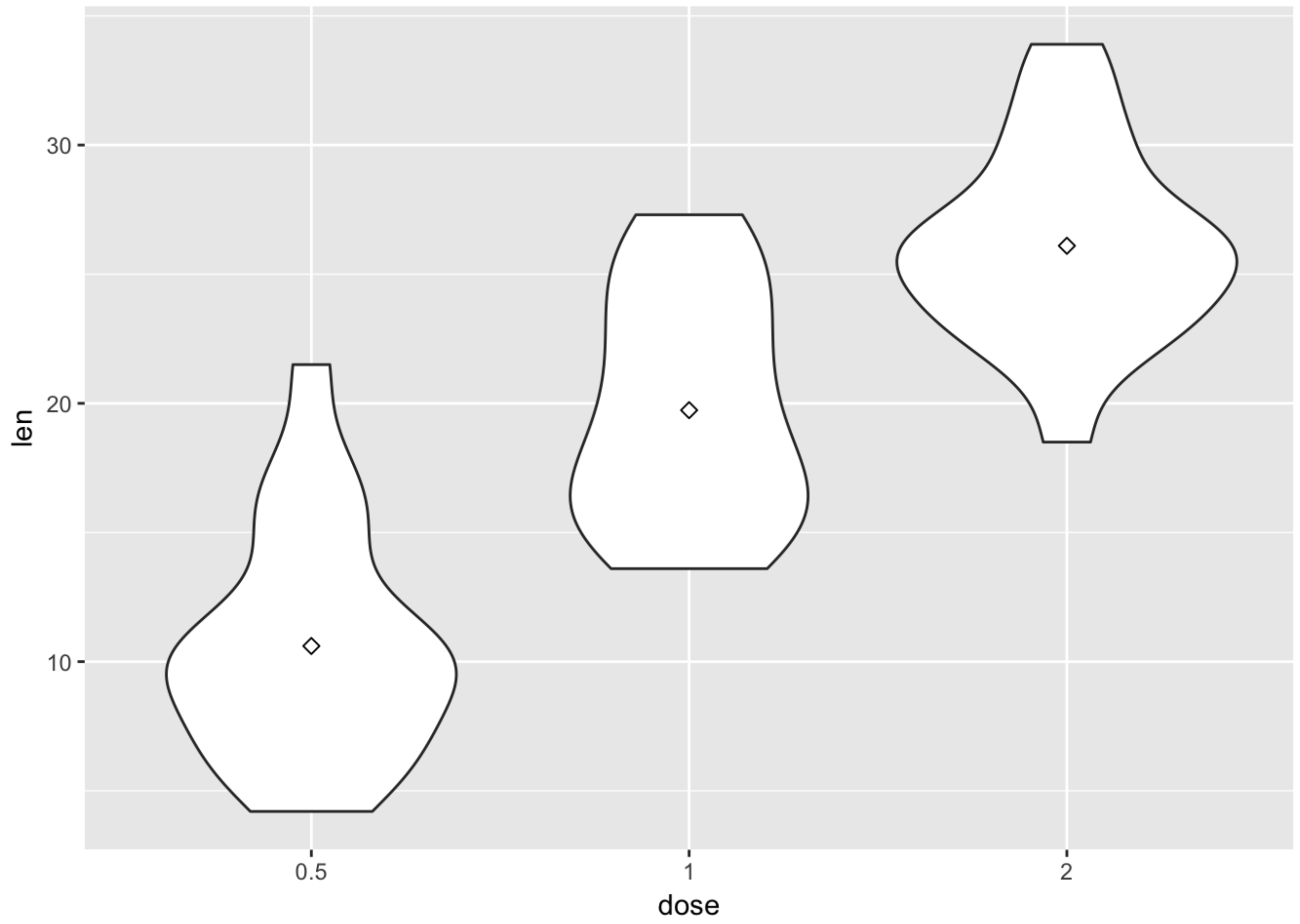
```
## List of 3
##  $ r: num [1:11, 1:11] 1 -0.911 -0.909 -0.895 0.651 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##  $ n: int [1:11, 1:11] 32 32 32 32 32 32 32 32 32 32 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##  $ P: num [1:11, 1:11] NA 4.69e-13 6.37e-13 5.09e-12 5.38e-05 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##   .. ..$ : chr [1:11] "mpg" "cyl" "disp" "hp" ...
##  - attr(*, "class")= chr "rcorr"
```

```
cordata <- melt(cormatrix$r)
ggplot(cordata, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() + xlab("") + ylab("")
```

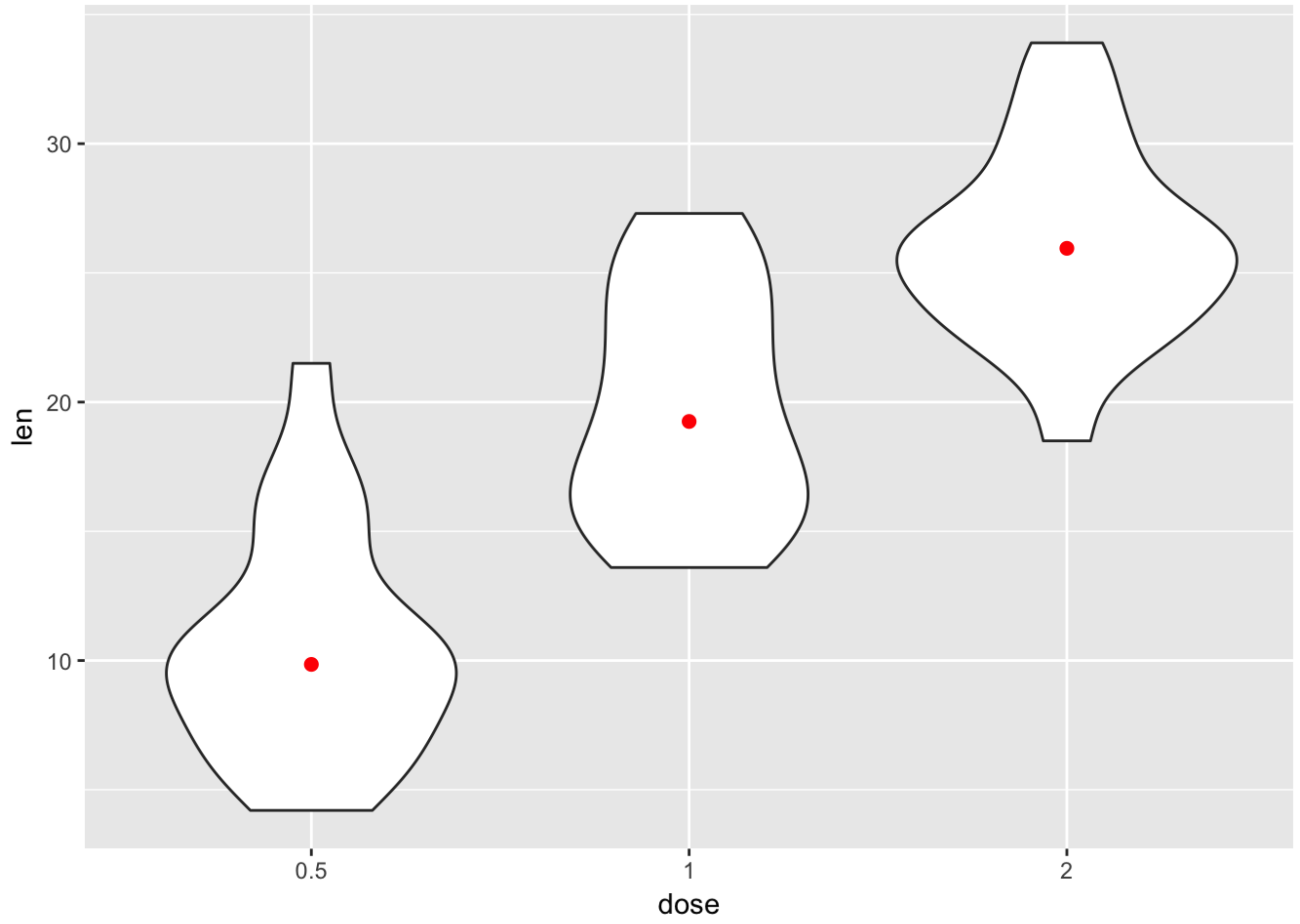


Violin plot – is similar to box plots, except that they also show the kernel probability density of the data at different values. Typically, violin plots will include a marker for the median of the data and a box indicating the interquartile range, as in standard box plots.

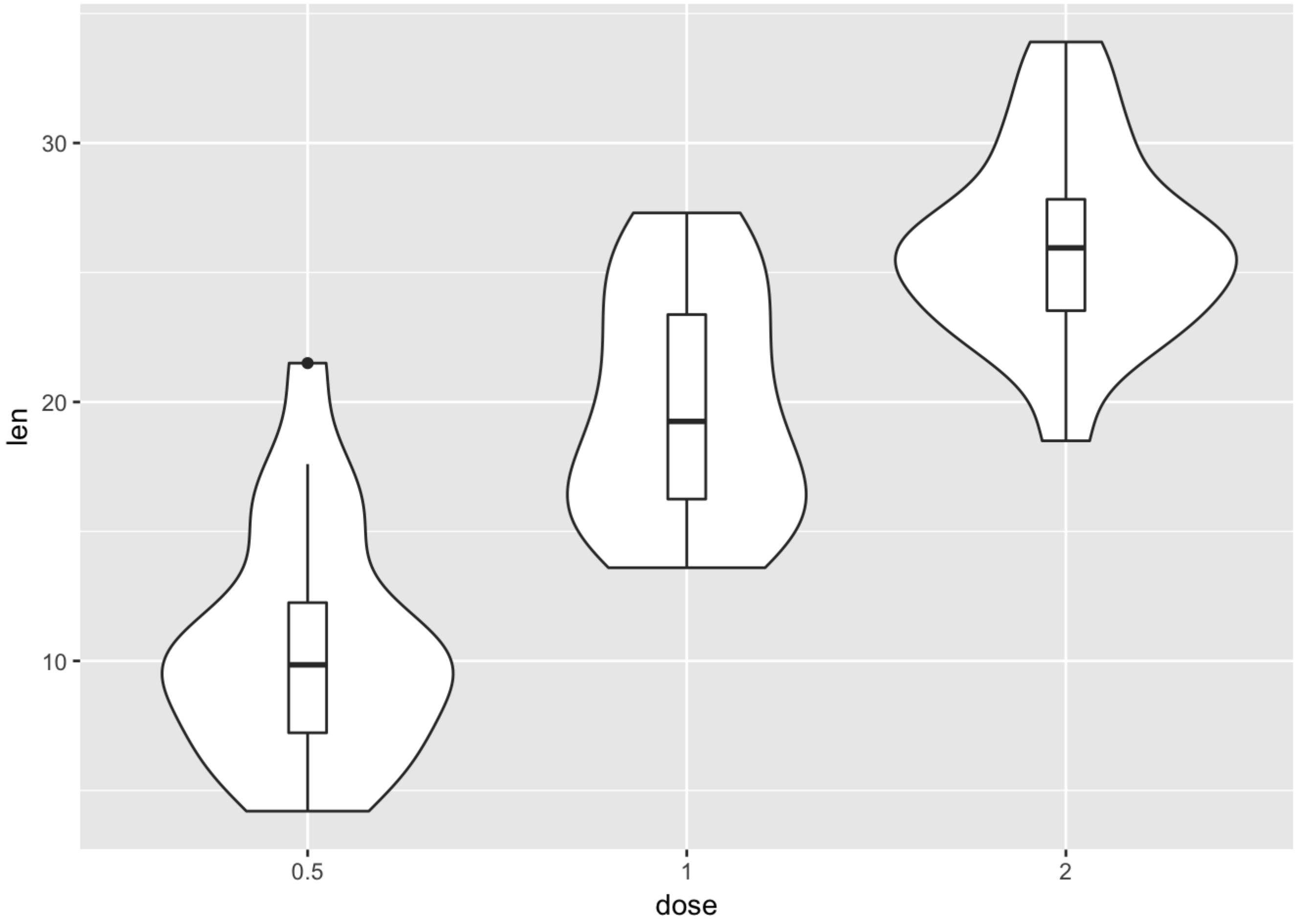
```
ToothGrowth$dose <- as.factor(ToothGrowth$dose)
p <- ggplot(ToothGrowth, aes(x=dose, y=len)) +
  geom_violin()
p + stat_summary(fun.y=mean, geom="point", shape=23, size=2)
```



```
p + stat_summary(fun.y=median, geom="point", size=2, color="red")
```

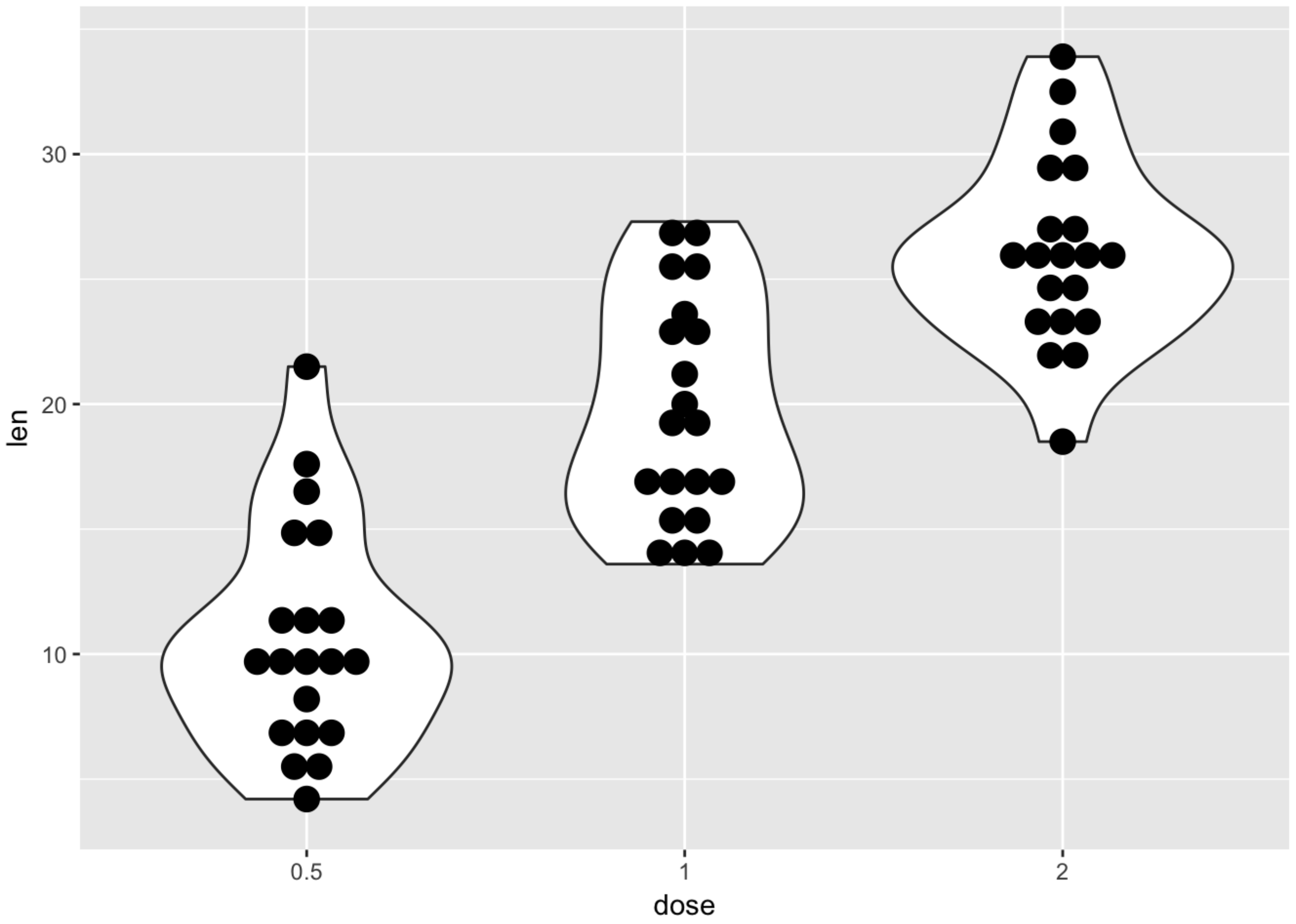


```
p + geom_boxplot(width=0.1)
```

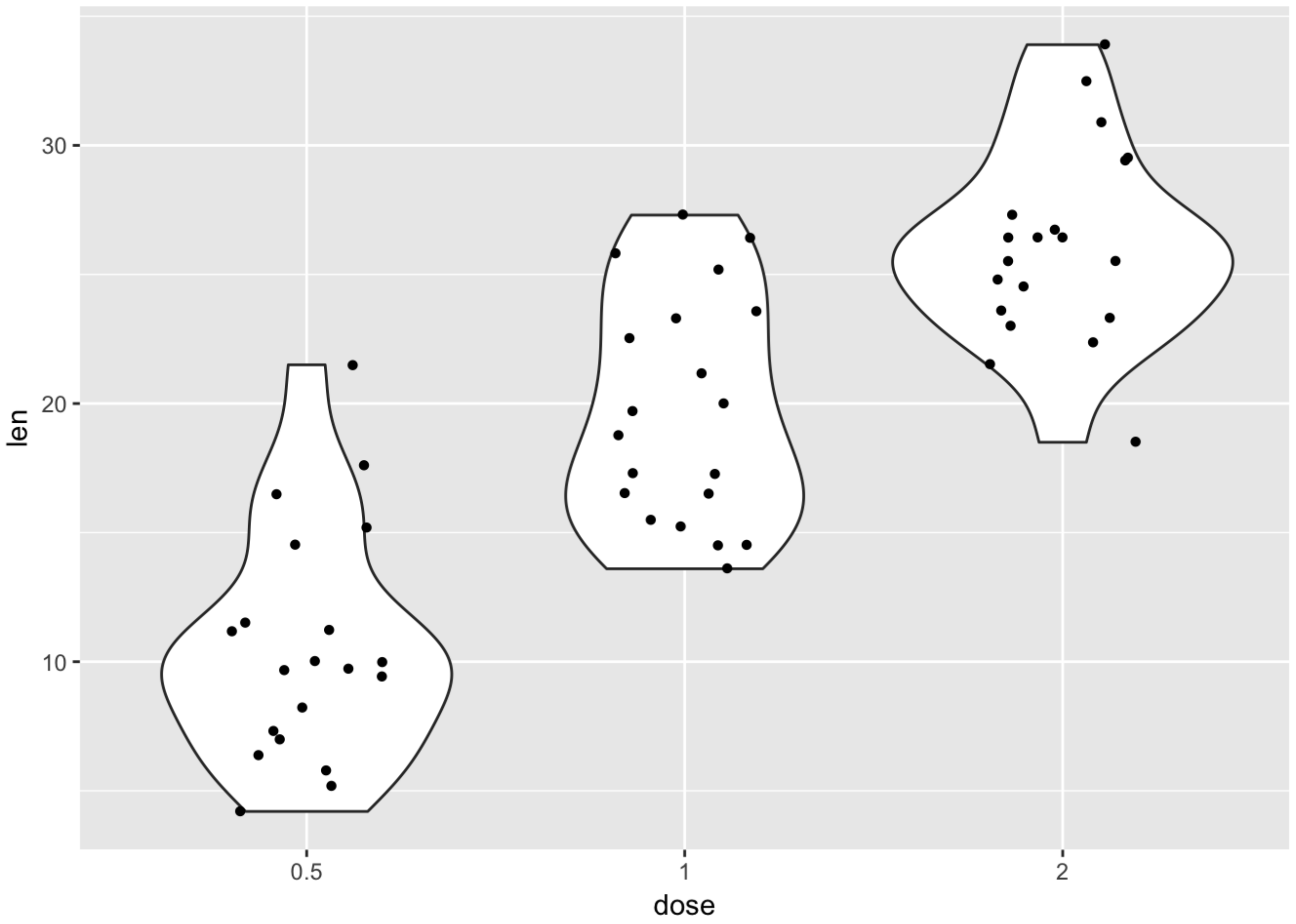


```
# violin plot with dot plot
p + geom_dotplot(binaxis='y', stackdir='center', dotsize=1)
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



```
# violin plot with jittered points
# 0.2 : degree of jitter in x direction
p + geom_jitter(shape=16, position=position_jitter(0.2))
```



30
20
10
len
0.5
1
2
dose