practice9

Qiang Liu

2022-10-18

```
> data("iris")
> head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
> data("ToothGrowth")
> head(ToothGrowth)
```

```
##    len supp dose
## 1  4.2   VC  0.5
## 2 11.5   VC  0.5
## 3  7.3   VC  0.5
## 4  5.8   VC  0.5
## 5  6.4   VC  0.5
## 6 10.0   VC  0.5
```

```
> data("PlantGrowth")
> head(PlantGrowth)
```

```
##   weight group
## 1   4.17  ctrl
## 2   5.58  ctrl
## 3   5.18  ctrl
## 4   6.11  ctrl
## 5   4.50  ctrl
## 6   4.61  ctrl
```

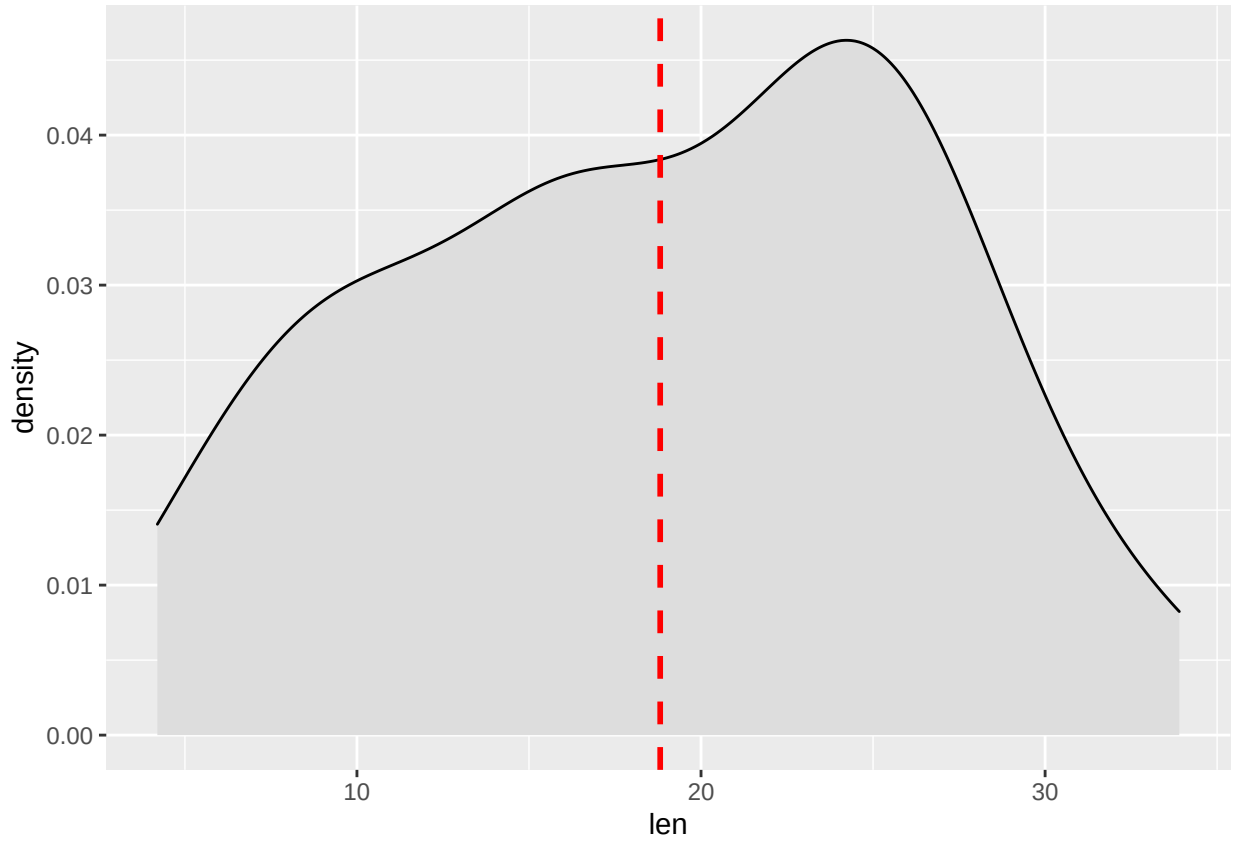
```
> # Store the data in the variable test_data
> test_data <- ToothGrowth
> set.seed(123456)
> dplyr::sample_n(test_data, 10)
```

```
##     len supp dose
## 1  23.0   OJ  2.0
## 2  23.3   OJ  1.0
## 3  29.4   OJ  2.0
## 4  14.5   OJ  1.0
## 5  11.2   VC  0.5
## 6  20.0   OJ  1.0
## 7  24.5   OJ  2.0
## 8  10.0   OJ  0.5
## 9   9.4   OJ  0.5
## 10  7.0   VC  0.5
```

```
> library(ggplot2)
```

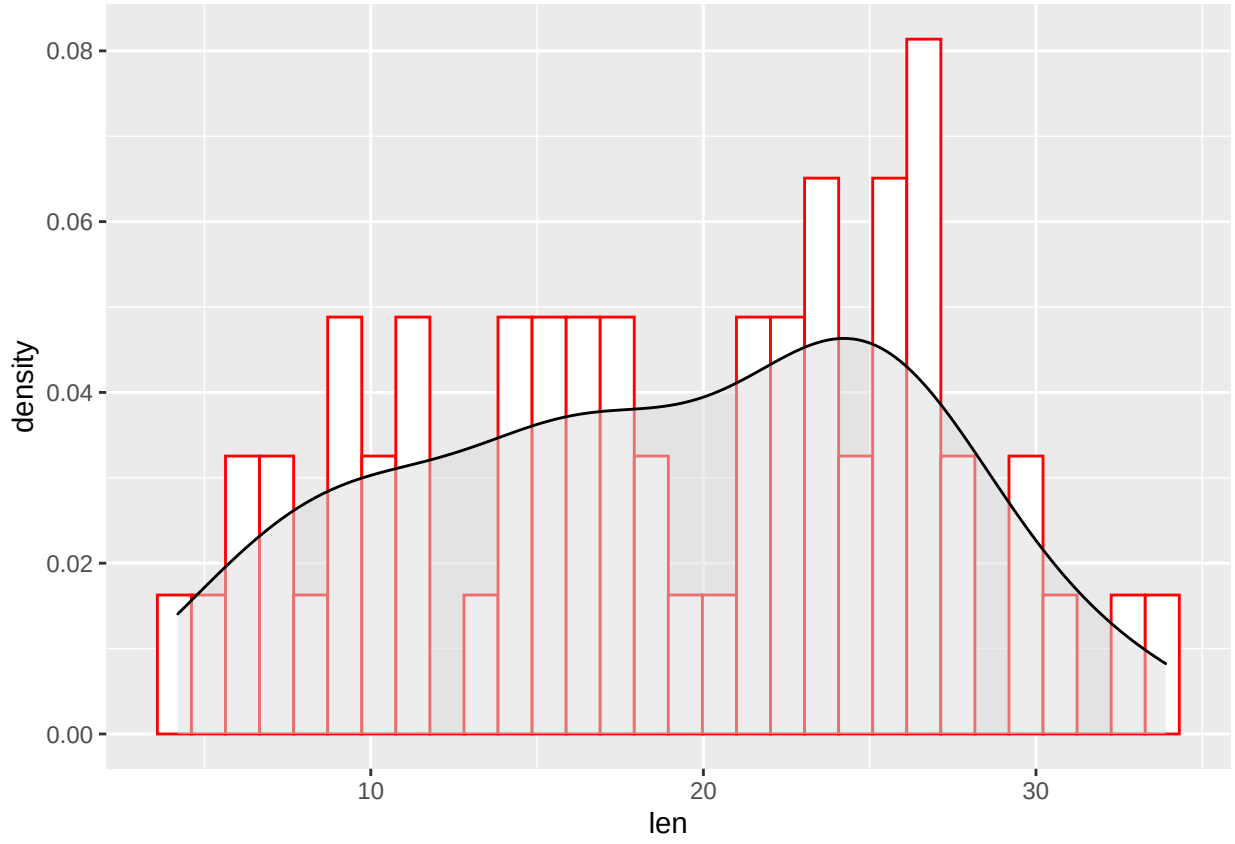
```
## Warning:  'ggplot2' R 4.2.1
```

```
> # Basic density plot
> p <- ggplot(test_data, aes(x=len)) + geom_density(color='black', fill='#dddddd')+
+   geom_vline(aes(xintercept=mean(len)), color="red", linetype="dashed", size=1)
> p
```

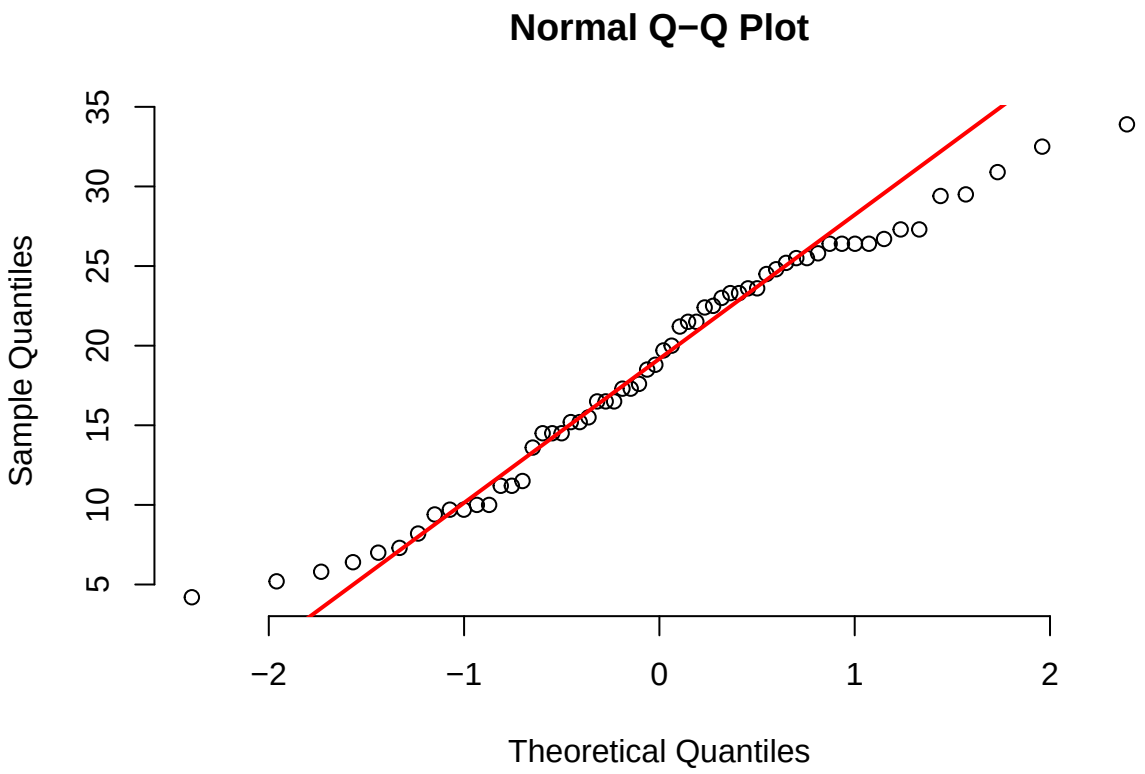



```
> ggplot(test_data, aes(x=len)) +
+ geom_histogram(aes(y=..density..), colour="red", fill="white")+
+ geom_density(alpha=.5, fill="#dddddd")
```

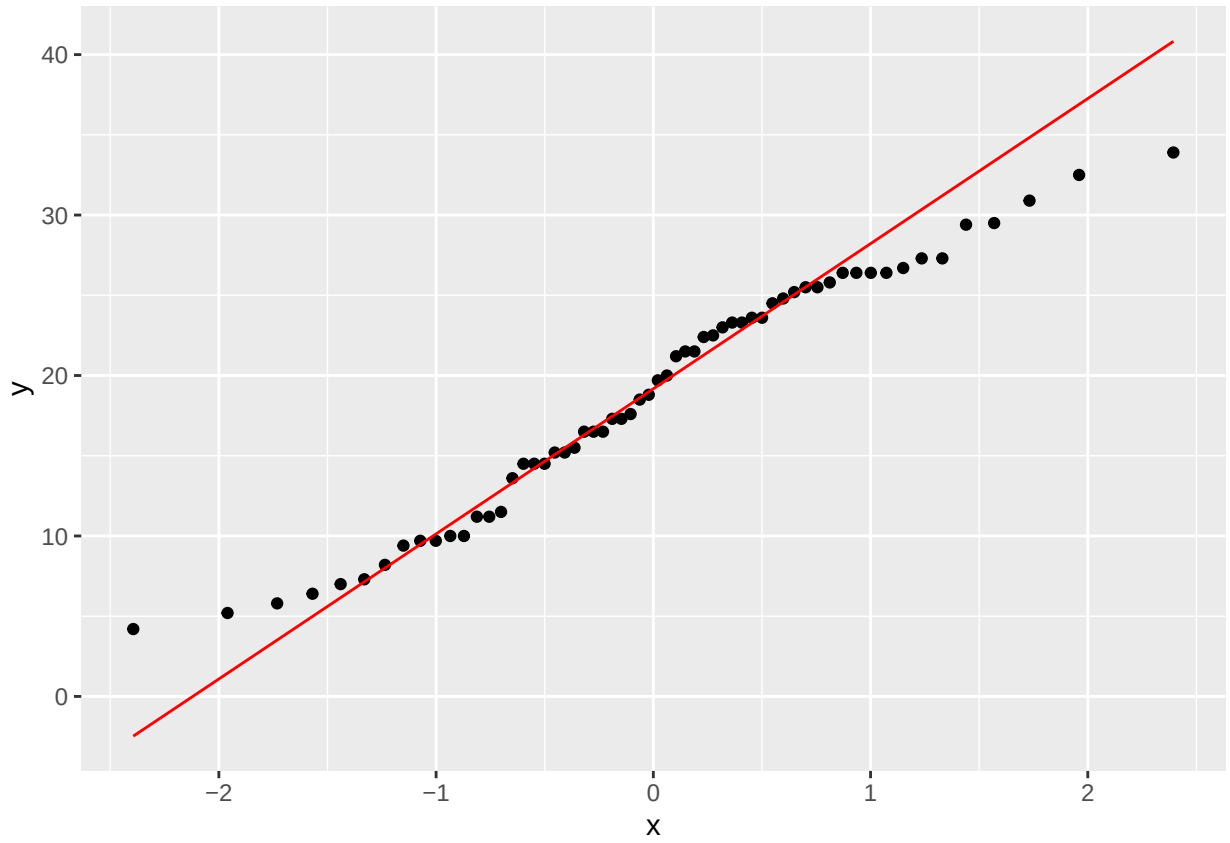
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
> ## Base R function
> qqnorm(test_data$len, frame = FALSE)
> qqline(test_data$len, col = "red", lwd = 2)
```



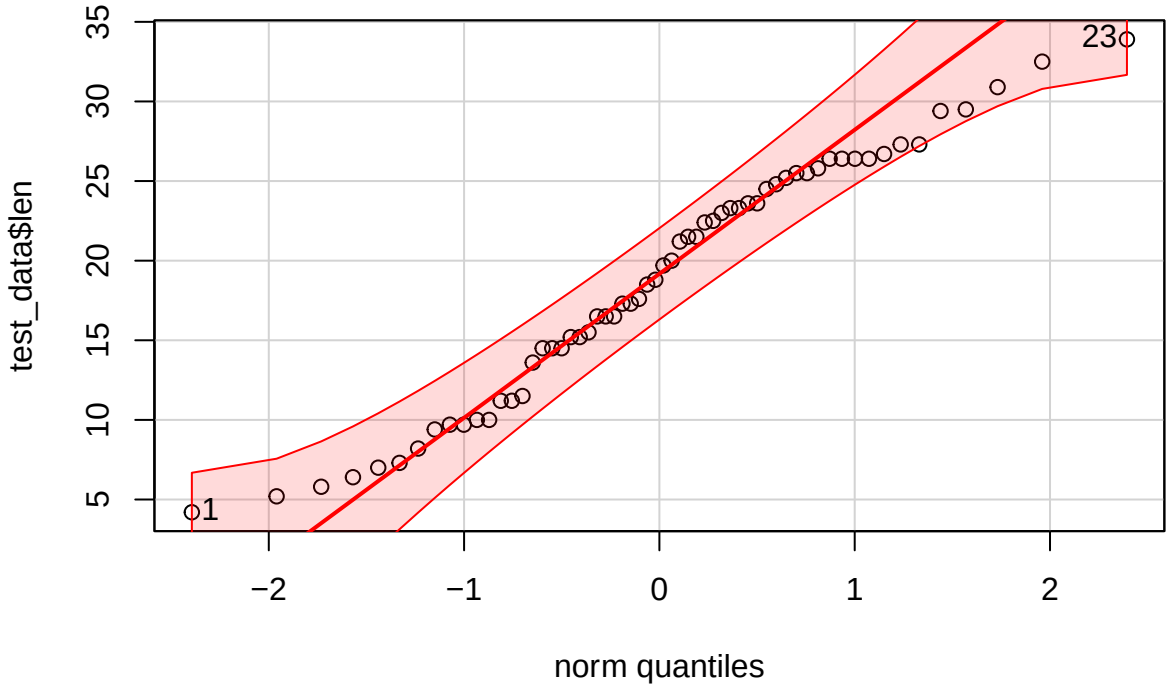

```
> p <- ggplot(test_data, aes(sample = len)) +
+ stat_qq() + stat_qq_line(color='red')
> print(p)
```




```
> library("car")
```

```
##      carData
```

```
> out <- qqPlot(test_data$len, col.lines = 'red')
```



```
> shapiro.test(test_data$len)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  test_data$len
## W = 0.96743, p-value = 0.1091
```

```
> # Generate some random data
> set.seed(123456)
> wdata <- data.frame(
+ name = paste0(rep("M_", 10), 1:10),
+ weight = round(rnorm(10, 20, 2), 1)
+ )
> # the first 6 rows of the data
> head(wdata)
```

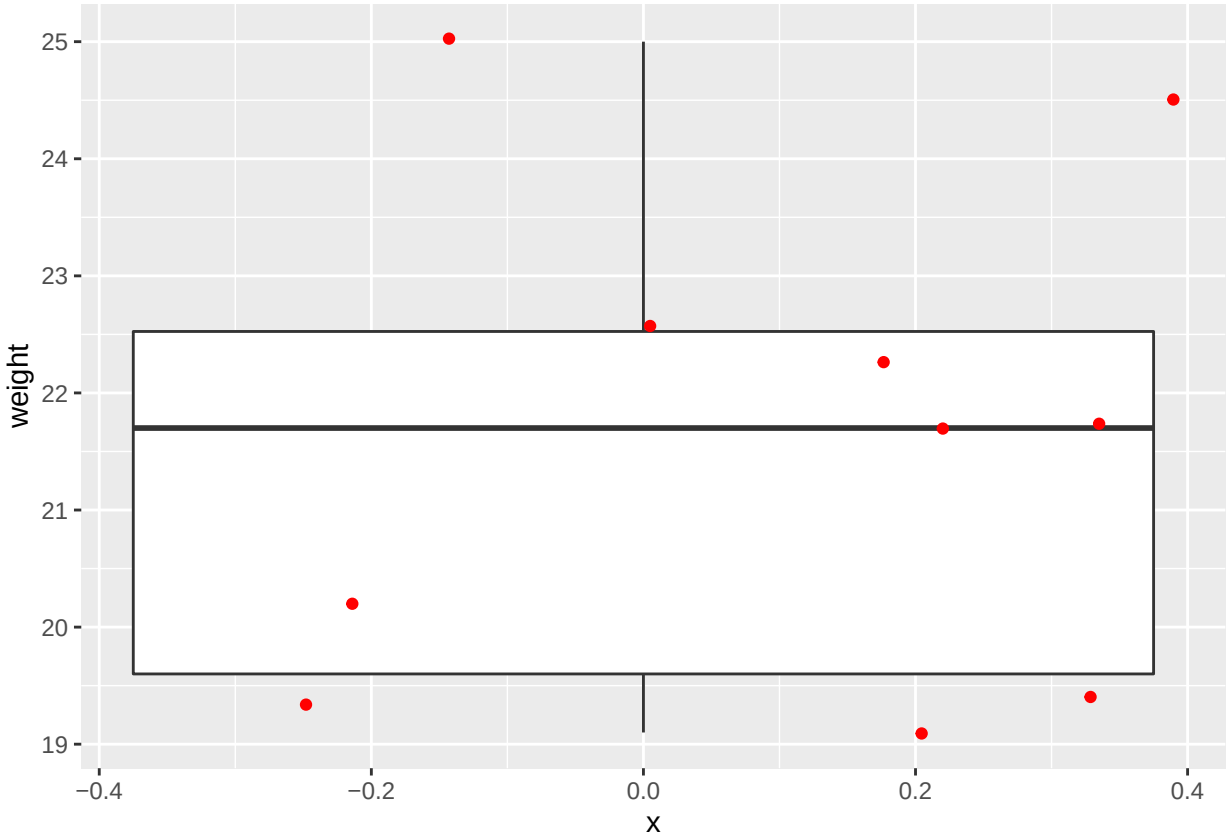
```
##   name weight
## 1  M_1   21.7
## 2  M_2   19.4
## 3  M_3   19.3
## 4  M_4   20.2
```

```
## 5  M_5   24.5
## 6  M_6   21.7
```

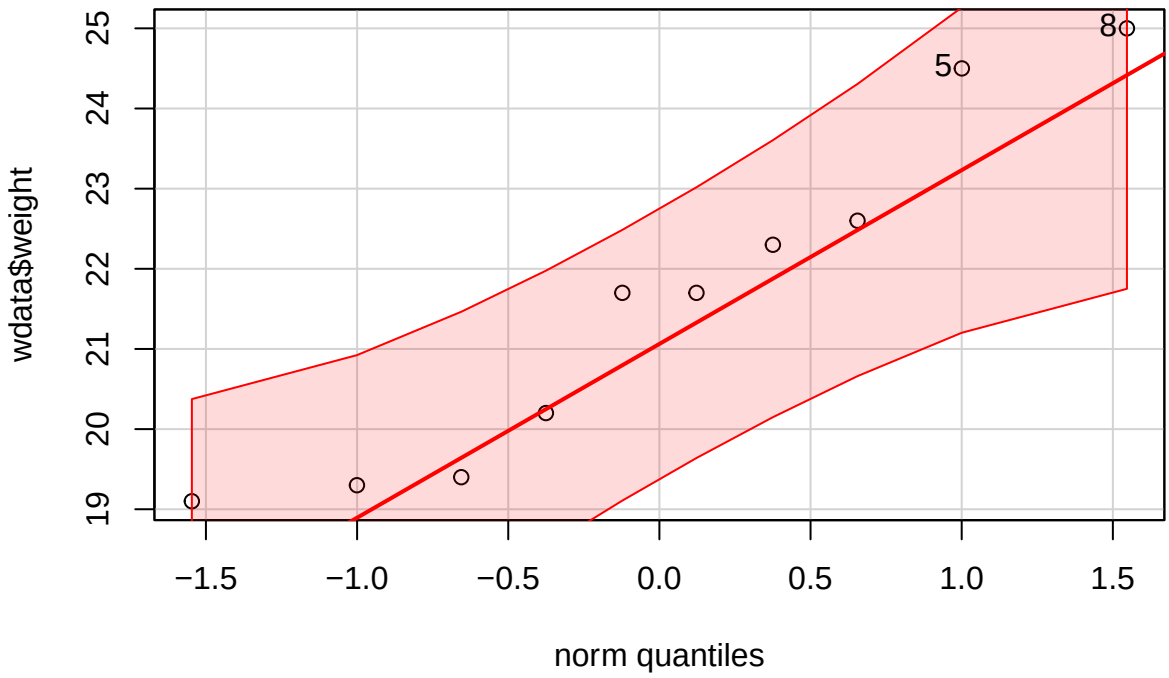
```
> # Statistical summaries of weight
> summary(wdata$weight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.10   19.60   21.70   21.58   22.52   25.00
```

```
> # Visualize your data using box plots
> ggplot(wdata, aes(y=weight)) + geom_boxplot() + geom_jitter(aes(x=0), color='red')
```




```
> # Visual inspection of the data normality using Q-Q plots
> out <- qqPlot(wdata$weight, col.lines = 'red')
```



```
> # One-sample t-test
> res <- t.test(wdata$weight, mu = 25)
> # Printing the results
> res
```

```
##
##  One Sample t-test
##
## data:  wdata$weight
## t = -5.1418, df = 9, p-value = 0.0006098
## alternative hypothesis: true mean is not equal to 25
## 95 percent confidence interval:
##  20.07537 23.08463
## sample estimates:
## mean of x
##     21.58
```

```
> t.test(wdata$weight, mu = 25, alternative = "less")
```

```
##
##  One Sample t-test
##
```

```
## data:  wdata$weight
## t = -5.1418, df = 9, p-value = 0.0003049
## alternative hypothesis: true mean is less than 25
## 95 percent confidence interval:
##      -Inf 22.79926
## sample estimates:
## mean of x
##     21.58
```

```
> t.test(wdata$weight, mu = 25, alternative = "greater")
```

```
##
##  One Sample t-test
##
## data:  wdata$weight
## t = -5.1418, df = 9, p-value = 0.9997
## alternative hypothesis: true mean is greater than 25
## 95 percent confidence interval:
##  20.36074      Inf
## sample estimates:
## mean of x
##     21.58
```

```
> # printing the p-value
> res$p.value
```

```
## [1] 0.0006097862
```

```
> # printing the mean
> res$estimate
```

```
## mean of x
##     21.58
```

```
> # printing the confidence interval
> res$conf.int
```

```
## [1] 20.07537 23.08463
## attr(,"conf.level")
## [1] 0.95
```

```
> # F-test
> res.ftest <- var.test(len ~ supp, data = test_data)
> res.ftest
```

```
##
##  F test to compare two variances
##
```

```
## data:  len by supp
## F = 0.6386, num df = 29, denom df = 29, p-value = 0.2331
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3039488 1.3416857
## sample estimates:
## ratio of variances
##          0.6385951
```

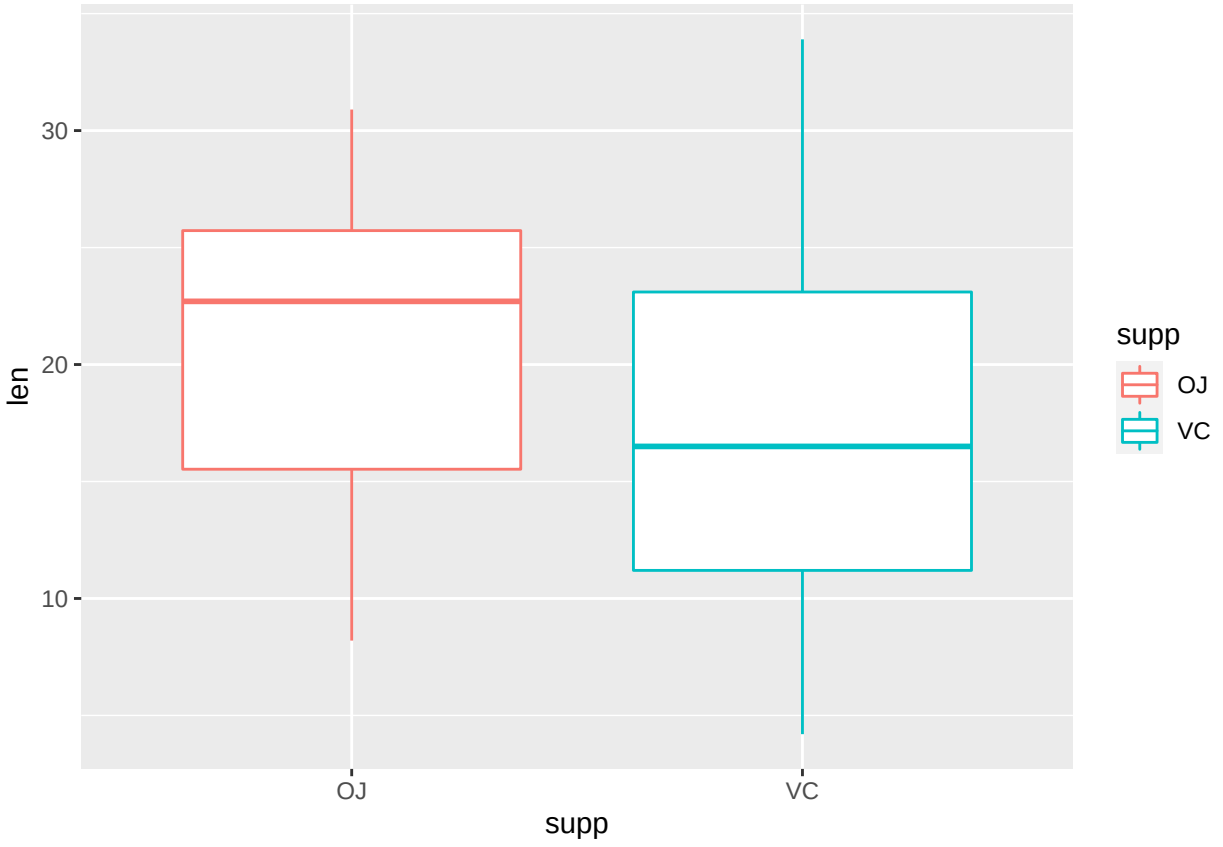
```
> # ratio of variances
> res.ftest$estimate
```

```
## ratio of variances
##          0.6385951
```

```
> # p-value of the test
> res.ftest$p.value
```

```
## [1] 0.2331433
```

```
> # Visualize your data using box plots by group
> ggplot(test_data, aes(x=supp, y=len, color=supp)) + geom_boxplot()
```



```
> # Shapiro-Wilk normality test for OJ
> with(test_data, shapiro.test(len[supp == "OJ"]))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  len[supp == "OJ"]
## W = 0.91784, p-value = 0.02359
```

```
> # Shapiro-Wilk normality test for OJ
> with(test_data, shapiro.test(len[supp == "OJ"]))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  len[supp == "OJ"]
## W = 0.91784, p-value = 0.02359
```

```
> res.ftest <- var.test(len ~ supp, data = test_data)
> res.ftest
```

```
##
##  F test to compare two variances
##
## data:  len by supp
## F = 0.6386, num df = 29, denom df = 29, p-value = 0.2331
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3039488 1.3416857
## sample estimates:
## ratio of variances
##          0.6385951
```

```
> res.ftest <- var.test(len ~ supp, data = test_data)
> res.ftest
```

```
##
##  F test to compare two variances
##
## data:  len by supp
## F = 0.6386, num df = 29, denom df = 29, p-value = 0.2331
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.3039488 1.3416857
## sample estimates:
## ratio of variances
##          0.6385951
```

```
> t.test(len ~ supp, data = test_data,
+ var.equal = TRUE, alternative = "less")
```

```
##
##  Two Sample t-test
##
## data:  len by supp
## t = 1.9153, df = 58, p-value = 0.9698
## alternative hypothesis: true difference in means between group OJ and group VC is less than 0
## 95 percent confidence interval:
##     -Inf 6.92918
## sample estimates:
## mean in group OJ mean in group VC
##         20.66333         16.96333
```

```
> t.test(len ~ supp, data = test_data,
+ var.equal = TRUE, alternative = "greater")
```

```
##
##  Two Sample t-test
##
## data:  len by supp
## t = 1.9153, df = 58, p-value = 0.0302
## alternative hypothesis: true difference in means between group OJ and group VC is greater than 0
## 95 percent confidence interval:
##  0.4708204       Inf
## sample estimates:
## mean in group OJ mean in group VC
##         20.66333         16.96333
```

```
> library(dplyr)
```

```
## Warning:   'dplyr' R 4.2.1
```

```
##
##    'dplyr'
```

```
## The following object is masked from 'package:car':
##
##     recode
```

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

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

```
> species <- levels(iris$Species)
> out <- data.frame()
> cmps <- combn(species, 2)
> for(i in 1:ncol(cmps)){
+   cmp <- cmps[,i]
+   test_data <- iris %>% filter(Species %in% cmp) %>% select(Species,Sepal.Length)
+   pvals <- c()
+   for(s in cmp){
+     x <- test_data[test_data$Species == s, 'Sepal.Length']
+     test <- shapiro.test(x)
+     ## print(test$p.value)
+     pvals <- c(pvals, test$p.value)
+   }
+   if(all(pvals > 0.05)){
+     test <- t.test(Sepal.Length ~ Species, data=test_data)
+   }
+ }
```