Oneway ANOVAs

Aishat Alli

2023-04-27

```
alphabets = read.csv("alphabets.csv")
```

```
alphabets$Subject = factor(alphabets$Subject) # convert to nominal factor
alphabets$Alphabet = factor(alphabets$Alphabet)
summary(alphabets)
```

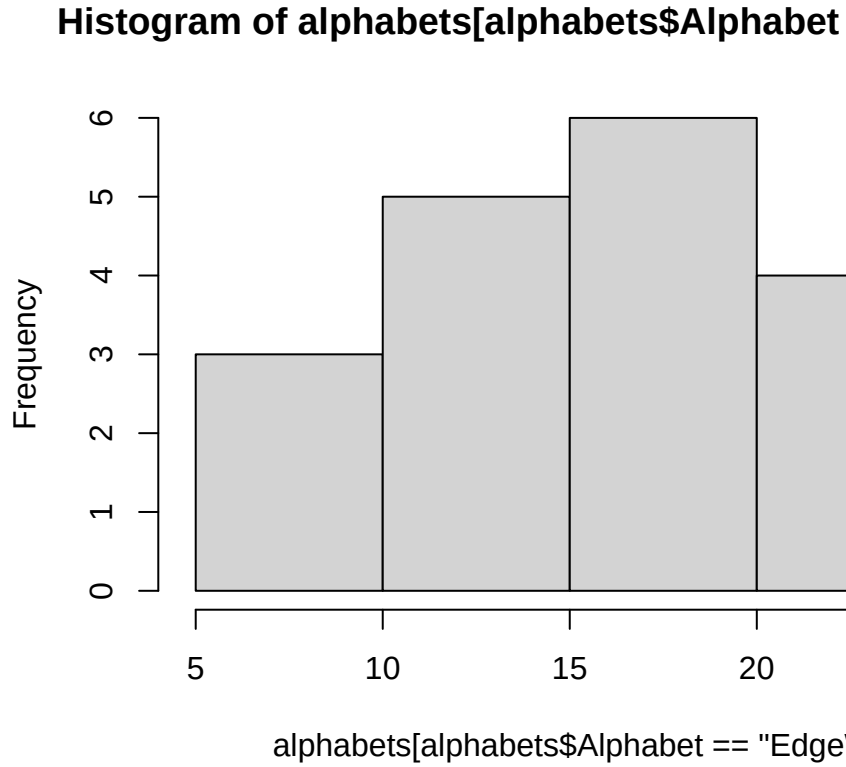
1. Download the file alphabets.csv from the course materials. This file describes a study in which people used a pen-based stroke alphabet to enter a set of text phrases. How many different stroke alphabets are being compared?

```
##     Subject         Alphabet       WPM
##  1      : 1   EdgeWrite :20   Min.   : 3.960
##  2      : 1   Graffiti  :20   1st Qu.: 9.738
##  3      : 1   Unistrokes:20   Median :13.795
##  4      : 1                   Mean   :14.517
##  5      : 1                   3rd Qu.:18.348
##  6      : 1                   Max.   :28.350
##  (Other):54
```

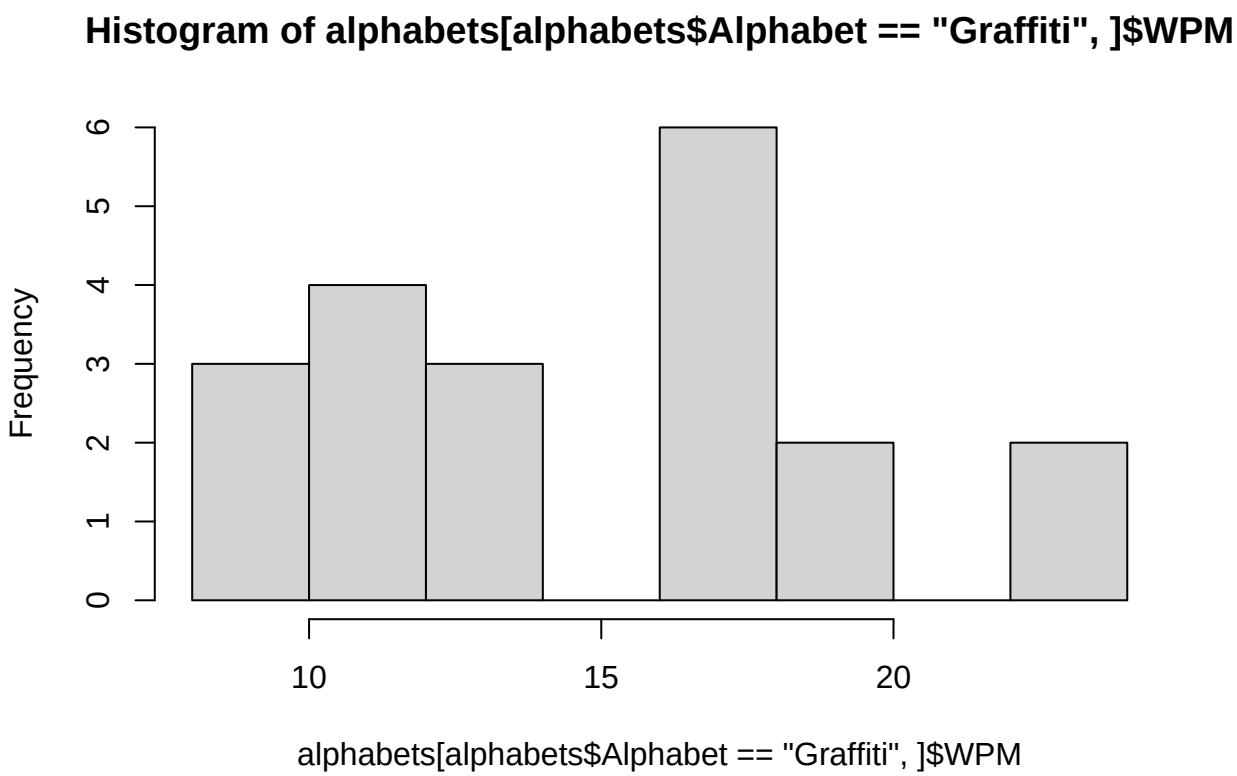
Ans: Three different stroke alphabets are being compared

```
hist(alphabets[alphabets$Alphabet == "EdgeWrite",]$WPM)
```

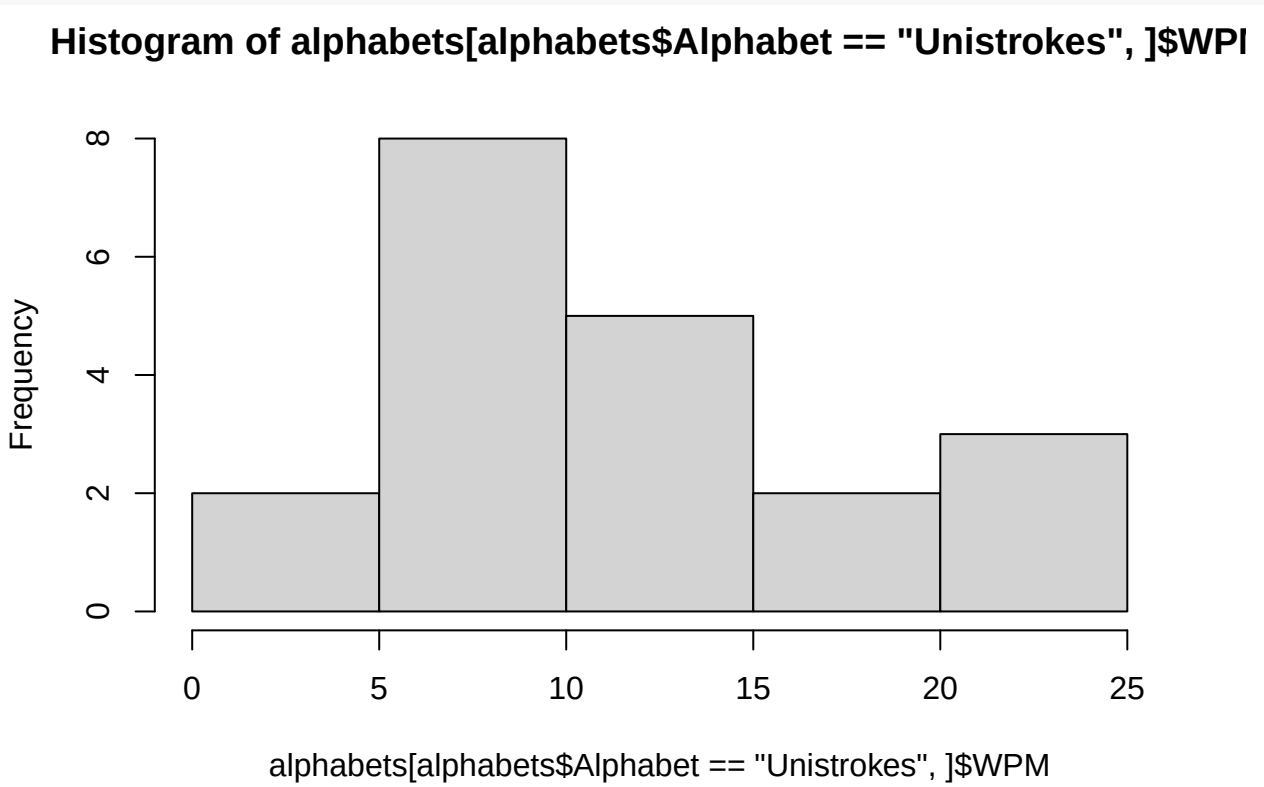
2. To the nearest hundredth (two digits), what was the average text entry speed in words per minute (WPM) of the EdgeWrite alphabet ?



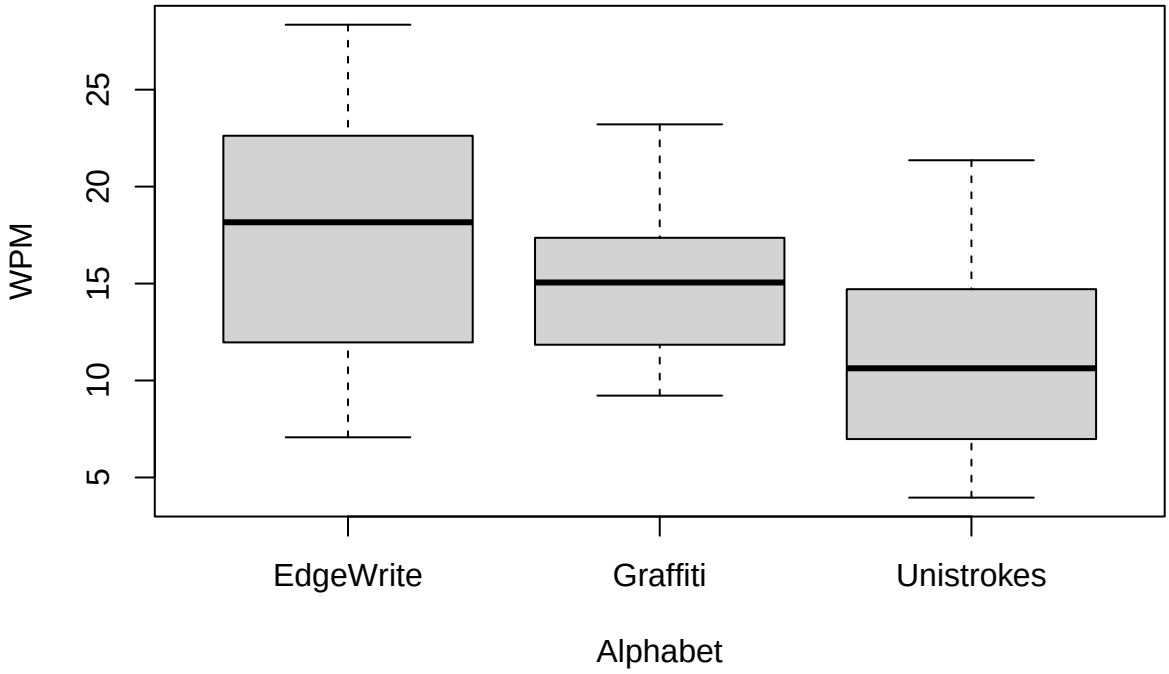

```
hist(alphabets[alphabets$Alphabet == "Graffiti",]$WPM)
```



```
hist(alphabets[alphabets$Alphabet == "Unistrokes",]$WPM)
```




```
plot(WPM ~ Alphabet, data=alphabets) # boxplot
```




```
install.packages("plyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(plyr)
ddply(alphabets, ~ Alphabet, summarise, WPM.mean=mean(WPM), WPM.sd=sd(WPM))
```

```
##     Alphabet WPM.mean   WPM.sd
## 1  EdgeWrite  17.1400 6.393038
## 2   Graffiti  15.0395 4.096276
## 3 Unistrokes  11.3715 5.323856
```

Ans: Average text entry speed in words per minute (WPM) of the EdgeWrite alphabet is 17.14

```
shapiro.test(alphabets[alphabets$Alphabet == "EdgeWrite",]$WPM)
```

3.Conduct Shapiro-wilk normality test on the WPM response for each alphabet. Which of them violate the normality test?

```
##
##  Shapiro-Wilk normality test
##
## data:  alphabets[alphabets$Alphabet == "EdgeWrite", ]$WPM
## W = 0.95958, p-value = 0.5355
```

```
shapiro.test(alphabets[alphabets$Alphabet == "Graffiti",]$WPM)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  alphabets[alphabets$Alphabet == "Graffiti", ]$WPM
## W = 0.94311, p-value = 0.2743
```

```
shapiro.test(alphabets[alphabets$Alphabet == "Unistrokes",]$WPM)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  alphabets[alphabets$Alphabet == "Unistrokes", ]$WPM
## W = 0.94042, p-value = 0.2442
```

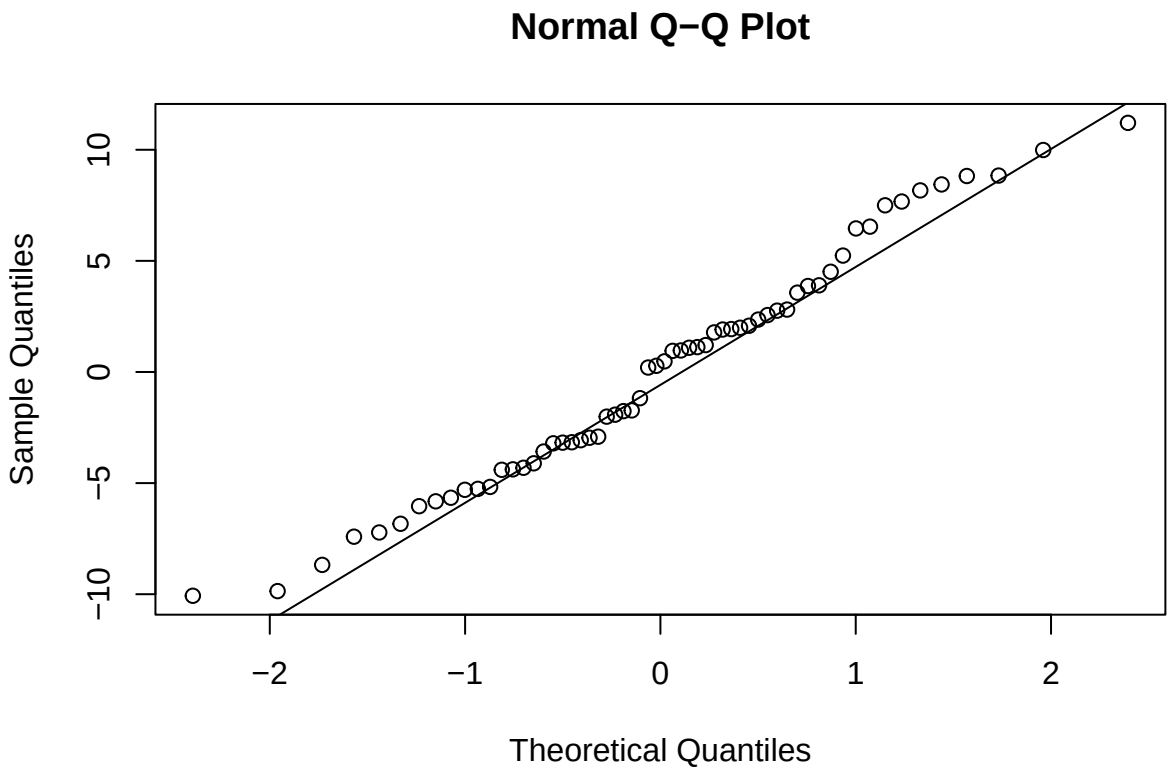
Ans: The p value for each of them is not significant, so we do not have a departure from normality.

```
m = aov(WPM ~ Alphabet, data=alphabets) # fit model
shapiro.test(residuals(m)) # test residuals
```

4. Conduct a Shapiro-Wilk normality test on the residuals of a WPM by Alphabetical model. To the nearest ten-thousandth (four digits), what is the p-value from such a test? Hint: fit a model with aov and then run shapiro.test on the model residuals.

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(m)
## W = 0.97762, p-value = 0.3363
```

```
qqnorm(residuals(m)); qqline(residuals(m)) # plot residuals
```



Ans: p-value = 0.3363

```
install.packages("car")
```

5. Conduct a Brown-Forsythe homoscedasticity test on WPM by Alphabet. To the nearest ten-thousandth(four digits), what is the p-value from such a test? Hint: Use the car library and its leveneTest function with center=median

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(car)
```

```
## Loading required package: carData
```

```
leveneTest(WPM ~ Alphabet, data=alphabets, center=median) # Brown-Forsythe test
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2  1.6219 0.2065
##       57
```

Ans: p-value = 0.2065, this shows no significant difference, so that means that we don't have a violation with WPM by Alphabets.

```
m = aov(WPM ~ Alphabet, data=alphabets) # fit model
anova(m) # report anova
```

Q6 Conduct a oneway ANOVA on WPM by Alphabet. To the nearest hundredth(two digits), what is the F statistic from such a test?

```
## Analysis of Variance Table
##
## Response: WPM
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## Alphabet   2  340.95 170.473  5.9472 0.00451 **
## Residuals 57 1633.88  28.665
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ans: F statistic = 5.9472 The p-value shows that there's difference among the aplhabets, it does not tell us exactly what the difference is, nor does it tell us where exactly the difference lies in terms of comparisons between each of these alphabets, so we have to look further.

```
install.packages("multcomp")
```

7. Perfomr simultaneous pairwise comparisons among levels of Alphabet using the Tukey approach. Adjust for multiple comparisons using Holm's sequential Bonferroni procedure. To the nearest ten-thousandth, what is the corrected p-value for the comparison of Unistrokes to Graffiti? Hint: Use the multicomp library and its mcp function called from within its glht function.

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(multcomp)
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: survival
```

```
## Loading required package: TH.data
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'TH.data'
```

```
## The following object is masked from 'package:MASS':
##
##     geyser
```

```
summary(glht(m, mcp(Alphabet="Tukey")), test=adjusted(type="holm")) # Tukey means compare all pairs
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: aov(formula = WPM ~ Alphabet, data = alphabets)
##
## Linear Hypotheses:
##                             Estimate Std. Error t value Pr(>|t|)
```

```
## Graffiti - EdgeWrite == 0     -2.100      1.693  -1.241  0.21982
## Unistrokes - EdgeWrite == 0   -5.769      1.693  -3.407  0.00363 **
## Unistrokes - Graffiti == 0    -3.668      1.693  -2.166  0.06894 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- holm method)
```

Ans: The corrected p-value for the comparison of Unistrokes to Graffiti is 0.0689

8. According to the results of the simultaneous pairwise comparisons, which of the levels of Alphabet are significantly different in terms of WPM?

Ans: Unistrokes versus EdgeWrite is significantly different, Graffiti versus EdgeWrite are not detectably different according to this test and Unistrokes versus Graffiti is a bit slightly significant.

```
install.packages("coin")
```

9. Conduct a Kruskal-Wallis test on WPM by Alphabet. To the nearest ten-thousandth, what is the p-value from such a test? Hint: Use the coin library and its krukal_test function with distribution="asymptotic"

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(coin)
kruskal_test(WPM ~ Alphabet, data=alphabets, distribution="asymptotic") # can't do exact with 3 levels
```

```
##
##  Asymptotic Kruskal-Wallis Test
##
## data:  WPM by
##   Alphabet (EdgeWrite, Graffiti, Unistrokes)
## chi-squared = 9.7019, df = 2, p-value = 0.007821
```

Ans: p-value = 0.0078

```
gf.ew = wilcox.test(alphabets[alphabets$Alphabet == "Graffiti",]$WPM, alphabets[alphabets$Alphabet == "
us.ew = wilcox.test(alphabets[alphabets$Alphabet == "Unistrokes",]$WPM, alphabets[alphabets$Alphabet ==
us.gf = wilcox.test(alphabets[alphabets$Alphabet == "Unistrokes",]$WPM, alphabets[alphabets$Alphabet ==
p.adjust(c(gf.ew$p.value, us.ew$p.value, us.gf$p.value), method="holm")
```

10. Conduct a nonparametric posthoc pairwise comparisons of WPM among all levels of Alphabet manually using three separate Mann-Whitney U tests. Adjust the p-value using Holm's sequential Bonferroni procedure. To the nearest ten-thousandth, what is the corrected p-value for Unistrokes vs. Graffiti?

```
## [1] 0.20358147 0.01810677 0.04146919
```

Ans: The corrected p-value for Unistrokes vs. Graffiti is 0.041