**Antworten**

1.

(a)

head(tv)

with(tv, table(V))

# zuerst eine Abbildung

boxplot(d ~ V, data = tv)

# Normalverteilt?

with(tv, tapply(d, V, shapiro.test))

# Beide: ja.

# Gleiche Varianzen?

var.test(d ~ V, data = tv)

# Nein (p < 0.05)

# daher

t.test(d ~ V, data = tv)

# Der Vokal hat einen signifikanten Einfluss auf die Dauer (t[12.5] = 4.3, p < 0.001)

(b)

fric = read.table(file.path(pfad, "fric.txt"))

head(fric)

with(fric, table(Fric))

# die anderen Frikative entfernen

temp = fric[,2] %in% c("x", "c")

fric = fric[temp,]

# Die anderen Frikative-Klassen haben einen Wert von Null

table(fric[,2])

# Um diese komplett zu entfernen:

fric[,2] = factor(fric[,2])

# zuerst eine Abbildung

boxplot(d ~ Fric, data = fric)

# Normalverteilt?

with(fric, tapply(d, Fric, shapiro.test))

# Die Kategorie "x" (also die zweite Kategorie) ist

# sicherlich wegen des Ausreißers nicht normalverteilt.

# Wenn es berechtigt ist, entferne wir diesen

temp = fric[,1] > 200 & fric[,2] =="x"

sum(temp)

fric = fric[!temp,]

# Neue Abbildung

boxplot(d ~ Fric, data = fric)

# Normalverteilt?

with(fric, tapply(d, Fric, shapiro.test))

# Beide: ja.

# Gleiche Varianzen?

var.test(d ~ Fric, data = fric)

# Ja

# daher

t.test(d ~ Fric, data = fric, var.equal=T)

# Die Kategorie Frikativ hat keinen signifikanten Einfluss auf die Dauer.

3(c)

dip = read.table(file.path(pfad, "dip.txt"))

head(dip)

table(dip[,2])

# wir wollen /aU/ und /OY/ von Sprecher 67

temp = dip[,2] %in% c("aU", "OY") & dip[,3] == "67"

sum(temp)

dip = dip[temp,]

dip[,2] = factor(dip[,2])

# zuerst eine Abbildung

boxplot(d ~ V, data = dip)

# Normalverteilt?

with(dip, tapply(d, V, shapiro.test))

# Ja

# Gleiche Varianzen?

var.test(d ~ V, data = dip)

# Ja

# daher

t.test(d ~ V, data = dip, var.equal=T)

# Die Kategorie Diphthong hat keinen signifikanten Einfluss auf die Dauer.

4. Die Vokaldauer in Funktionswörtern mÜssten kleiner sein.

vow = read.table(file.path(pfad, "vow.txt"))

temp = vow[,2] %in% c("bin", "ist", "nicht", "Kinder", "richten", "findet", "finden", "binden")

vow = vow[temp,]

vec = as.character(vow[,2])

temp = vec %in% c("bin", "ist", "nicht")

vec[temp] = "F"

vec[!temp] = "I"

vow[,2] = factor(vec)

# zuerst eine Abbildung

boxplot(d ~ Word, data = vow)

# Kein Test notwendig, da die Dauer von F größer ist als von I.

5.

F2 nach /b/ müsste niedriger sein als nach /d/

form = read.table(file.path(pfad, "form.txt"))

head(form)

table(form[,3])

table(form[,2])

temp = form[,3] == "I" & form[,2] %in% c("b", "d")

form = form[temp,]

form[,2] = factor(form[,2])

# zuerst eine Abbildung

boxplot(F2 ~ Kons, data = form)

# Normalverteilt?

with(form, tapply(F2, Kons, shapiro.test))

# Ja

# Gleiche Varianzen?

var.test(F2 ~ Kons, data = form)

# Ja

# daher

t.test(F2 ~ Kons, data = form, var.equal=T)

# Die Artikulationsstelle hatte keinen signifikanten Einfluss auf F2 (t[24] = 2.7, p < 0.05).

6.

p = read.table(file.path(pfad, "p.txt"))

# Abbildung

boxplot(p[,1] - p[,2])

# Normalverteilt? Hier die Unterschiede prüfen

shapiro.test(p[,1] - p[,2])

t.test(p[,1], p[,2], paired=T)

7.

# Es wird geprüft, ob das erste Element in

with(tv, levels(V))

# größer ist als das zweite

# (daher ob die Dauer von /a/ größer ist als diejenige von /I/

# da "a" an erster Stelle ist in

# with(tv, levels(V))

#

t.test(d ~ V, data = tv, alt="g")

# Die Dauer von /a/ ist signifikant größer

# (t[12.5] = 4.3, p < 0.01, one-tailed test) als diejenige von /I/.