Teste 2

Schmidt, Sophie C.

05 Juni, 2019

knitr::opts\_chunk$set(  
 collapse = TRUE,  
 warning = FALSE,  
 message = FALSE,  
 echo = TRUE,  
 comment = "#>",  
 fig.path = "../figures/",  
 fig.width=6,   
 fig.height=6  
)

Pakete für heute:

library(dplyr)  
library(tidyr)  
library(ggplot2)  
  
library(archdata)  
data("BACups")  
data("Fibulae")  
data("Acheulean")  
data("Bornholm")  
data("Snodgrass")  
  
if (!require("nortest")) install.packages("nortest")  
library(nortest)  
  
if (!require("ggpubr")) install.packages("ggpubr")  
library(ggpubr)  
  
if (!require("dgof")) install.packages("dgof")  
library(dgof)

## Test auf Normalverteilung mit Chi-Quadrat

Im Hintergrund ist die Berechnung hochkomplex, aber in R relativ simpel, wenn man das richtige Paket gefunden hat, in dem es die Funktion gibt, die man braucht…

Wir benutzen den Acheulean-Datensatz, da dort die Anzahlen von zB HA = Handaxes (Faustkeile) pro Fundplatz (rownames, die Namen der Zeilen) praktischerweise schon ausgezählt sind.

Ich muss nur

1. einen Vektor erstellen, der die ausgezählten Häufigkeiten meiner nominalskalierten Daten darstellt
2. die Anzahl der Klassen wissen. Die kann ich mir aber auch einfach von R noch einmal ausrechnen lassen: Die Funktion length zählt, wie lang ein Vektor ist (wie viele Werte drin sind)

#pearson.test stammt aus dem nortest-package  
pearson.test(Acheulean$HA, n.classes = length(Acheulean$HA), adjust = TRUE)   
#>   
#> Pearson chi-square normality test  
#>   
#> data: Acheulean$HA  
#> P = 2, p-value = 0.7358  
pearson.test(Acheulean$HA, n.classes = length(Acheulean$HA), adjust = FALSE)  
#>   
#> Pearson chi-square normality test  
#>   
#> data: Acheulean$HA  
#> P = 2, p-value = 0.9197

Was bedeutet “adjust”? Warum habe ich das einmal mit TRUE und einmal mit FALSE ausgeführt?

Ein Blick in die Hilfe hilft. Sucht einmal nach pearson.test und schaut was da steht.

Achtet insbesondere auf die Warnungen unter “Note”!

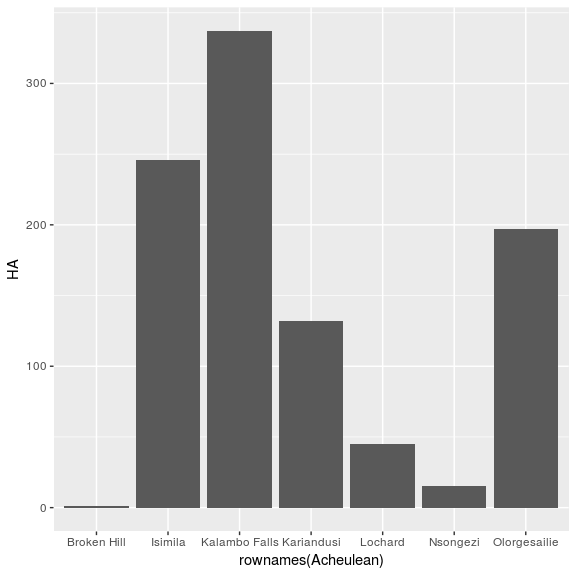
Das Ergebnis müssen wir jetzt aber auch noch interpretieren.

Die Regel war: Wenn p > 0.05 sind die Daten normalverteilt.

Das heißt?

Wer hätte das gedacht? Schauen wir uns die Daten doch noch einmal optisch an:

ggplot(data = Acheulean)+  
 geom\_col(aes(y = HA, x = rownames(Acheulean))) # geom\_col nehme ich, weil die Daten schon ausgezählt sind

 Man kann sich vorstellen: Wenn man die Reihenfolge auf der x-Achse verändert, sind die Daten tatsächlich normalverteilt!

Wie, Reihenfolge verändern?

Nicht vergessen: Das sind nominale Daten, die Reihenfolge ist egal!

Das ändert sich bei den…

## Ordinalen Daten

## K-S-Test

Der K-S-Test benötigt mindestens ordinal skalierte Daten, man kann ihn als Anpassungstest verwenden (und damit den Datensatz auf normalverteilung testen ) und als Zweichstichproben-Test nutzen.

Zuerst als Anpassungstest. Nehmen wir eine ordinale Variable, sagen wir die Perioden in dem Bornholmer Datensatz und ob es vllt eine Normalverteilung in der Häufigkeit der Belegung gibt.

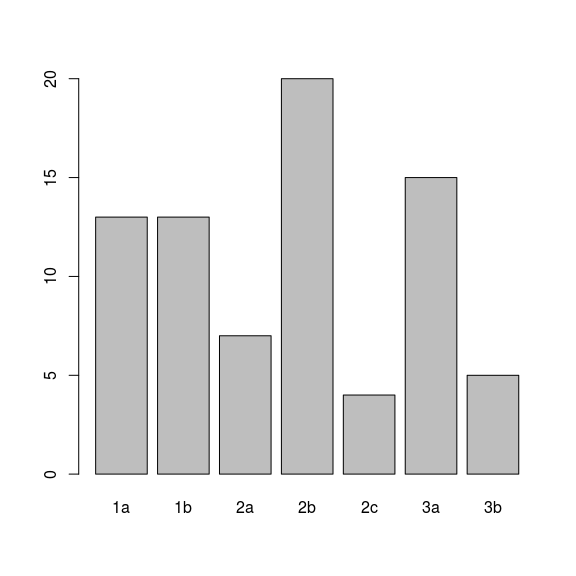
Bornholm$Period <- ordered(Bornholm$Period) # zuerst stelle ich sicher, dass R weiß, dass es sich um ordinale Daten handelt  
per <- table(Bornholm$Period) # dann zähl ich die Häufigkeiten  
  
ks.test(per, "pnorm" ) # und vergleiche die Häufigkeiten mit der Normalverteilung "pnorm"  
#>   
#> One-sample Kolmogorov-Smirnov test  
#>   
#> data: per  
#> D = 0.99997, p-value = 1.665e-06  
#> alternative hypothesis: two-sided

Der p-Wert ist sehr sehr klein, das heißt?

Daran denken: Die Nullhypothese ist, dass die beiden Verteilungen gleich sind. Wir testen in die “andere Richtung”, wenn p > 0.05 sind die Daten normalverteilt.

Schauen wir uns doch die Daten noch einmal kurz an, um zu überprüfen, ob wir das richtig verstanden haben:

barplot(per) # das ist der base-R-Befehl um ein Säulendiagramm zu erstellen. Nicht schön, aber schnell und wenig zu tippen.



Alles klar?

Dann weiter mit

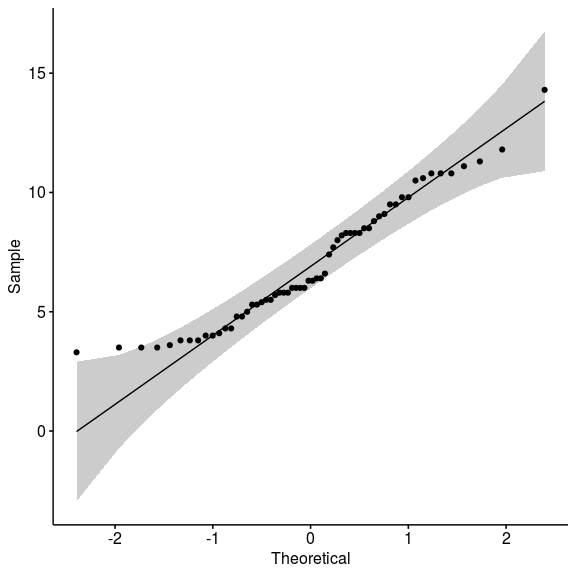
## metrischen Daten!

## Test auf Normalverteilung: optisch und mit Shapiro-Wilk

Optisch die normalverteilung abzuschätzen ist sinnvoll, dann kann man sich viele Tests sparen.

Deswegen schauen wir doch einmal, ob sich die Höhe der Gefäße im Datensatz BACups normalverteilt. Fangen wir mit einem Q-QPlot an!

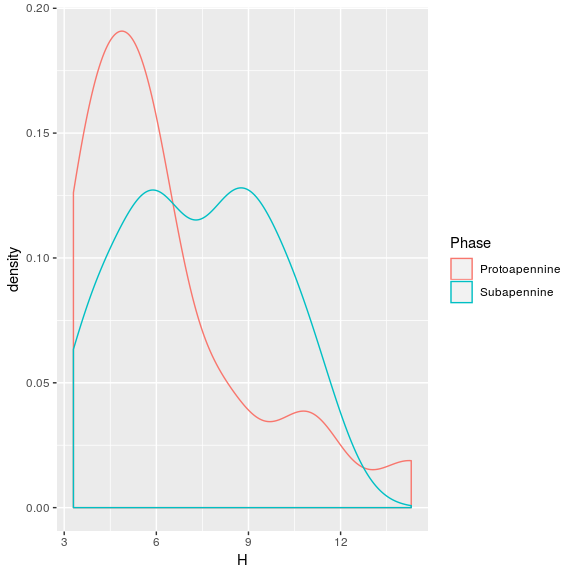
ggqqplot(BACups$H)

 Man sieht: auf der x-Achse wird die theoretische Verteilung abgetragen und auf der y-Achse die unserer Stichprobe (sample). Die Linie zeigt den Idealverlauf an.

Fast alle Punkte liegen in der grauen Zone, die das Konfidenzintervall von 95% darstellt. Das sieht also nicht ganz schlecht aus (wenn auch nicht so ganz exakt).

Schauen wir deshalb doch mit der Visualisierung als Dichte-Grafik weiter. Und differenzieren wir besser mal die zwei Gruppen, von denen wir wissen, dass sie im Datensatz stecken:

ggplot(BACups) +  
 geom\_density(aes(x = H, col = Phase)) # kurz überlegen: was genau passiert hier?

 Wenn ich mir das anschaue, ist die Höhe der Protoappeninen Tassen ja nicht ganz der Normalverteilung unähnlich, die subappenine Phase aber gar nicht. Testen wir also die protoappeninen Gefäße jetzt mit dem richtigen Test: Shapiro, los geht’s!

BACups\_p <- BACups %>%  
 filter(Phase == "Protoapennine") # ich kreiiere einen neuen Datensatz, in dem nur die BACups vorkommen, die Protoappenin in der Spalte "Phase" stehen haben  
  
BACups\_s <- BACups %>%  
 filter(Phase == "Subapennine") # ich kreiiere einen neuen Datensatz, in dem nur die BACups vorkommen, die Subappenin in der Spalte "Phase" stehen haben  
  
 shapiro.test(BACups\_p$H)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: BACups\_p$H  
#> W = 0.83274, p-value = 0.002776

Hmmmh. Das Ergebnis ist etwas enttäuschend. p ist KLEINER als 0.05, deswegen sind die Daten wohl nicht normalverteilt (Ich nehme die Alternativhypothese mit großer Wahrscheinlichkeit an.)

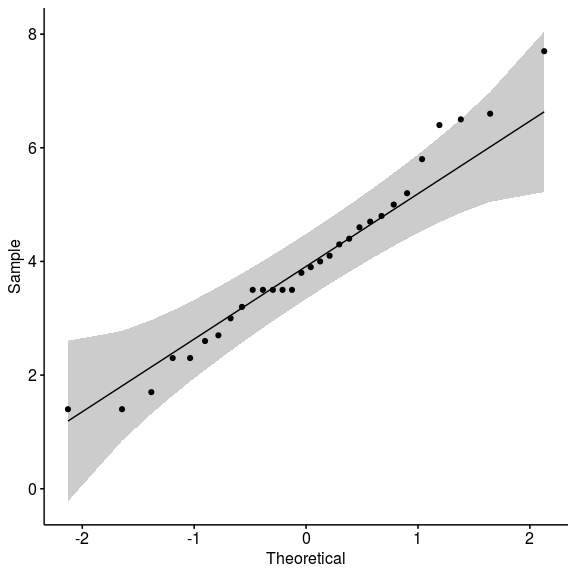
Die Sache mit den Gruppen ist total wichtig. Damit man nicht jedes mal einen neuen Datensatz anlegen muss, gibt es einen Trick: Man benutzt tapply. Tapply macht aus meinem Datensatz Gruppen und wendet für jede Gruppe den Test an, den ich auswähle:

# tapply(datensatz$testvariable, datensatz$gruppenvariable, shapiro.test)  
   
 # Probiert es aus!  
  
tapply(BACups$H, BACups$Phase, shapiro.test)  
#> $Protoapennine  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: X[[i]]  
#> W = 0.83274, p-value = 0.002776  
#>   
#>   
#> $Subapennine  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: X[[i]]  
#> W = 0.95576, p-value = 0.1198

Ok. Das Ergebnis bleibt aber doch das gleiche.

Dann schauen wir doch einmal in einen anderen Datensatz. Es gibt ja noch den zu den Fibeln. Testen wir doch einmal, ob die Dicke des Bogens sich vielleicht normalverteilt. Zuerst der Q-Q-Plot

ggqqplot(Fibulae$BT)



ggplot(Fibulae) +  
 geom\_density(aes(x = BT))

 Die density-Funktion zeigt mir ein ganz überzeugendes Bild. Was sagt shapiro diesmal?

shapiro.test(Fibulae$BT)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: Fibulae$BT  
#> W = 0.97135, p-value = 0.5769

Aha! Ein p-Wert von 0.58. Wie kann der interpretiert werden?

Super, jetzt wissen wir, wie wir am Anfang einmal überprüfen, ob unsere Daten normalverteilt sind. Wichtig wichtig für viele Testverfahren!

Jetzt vergleichen wir zwei Gruppen miteinander.

## Vergleich zweier Gruppen

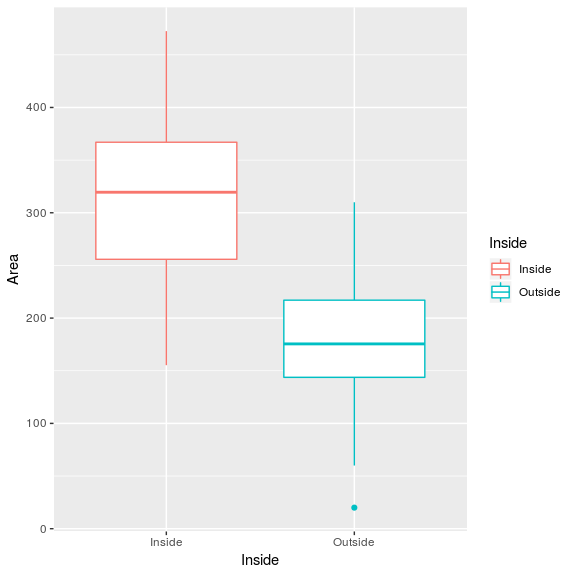
## U-Test

Nehmen wir doch einfach die Area-Angaben aus dem Snodgrass-Datensatz und schauen, ob die Gruppen “innerhalb der Mauer” und “außerhalb der Mauer” signifikant unterschiedlich sind:

snod\_in <- subset(Snodgrass, Inside == "Inside" )  
snod\_out <- subset(Snodgrass, Inside == "Outside" )  
  
wilcox.test(snod\_in$Area, snod\_out$Area)  
#>   
#> Wilcoxon rank sum test with continuity correction  
#>   
#> data: snod\_in$Area and snod\_out$Area  
#> W = 1879, p-value = 2.306e-12  
#> alternative hypothesis: true location shift is not equal to 0

Der p-Wert ist wieder super klein, das heißt, die beiden Gruppen sind signifikant unterschiedlich! Schauen wir uns das einmal an:

ggplot(data = Snodgrass)+  
 geom\_boxplot(aes(y = Area, x = Inside, col = Inside))

 Ich bin überzeugt. Das können wir wirklich stichhaltig interpretieren!

### K-S-Test mit zwei Verteilungen:

Kriegen wir das gleiche Ergebnis mit dem K-S-Test?

ks.test(snod\_in$Area, snod\_out$Area)  
#>   
#> Two-sample Kolmogorov-Smirnov test  
#>   
#> data: snod\_in$Area and snod\_out$Area  
#> D = 0.71351, p-value = 3.27e-10  
#> alternative hypothesis: two-sided

Ja! Goil!

## F-Test

Der F-Test vergleicht die Varianzen von metrischen Variablen. Schauen wir uns doch noch einmal die Höhe der BACups an:

var.test(x = BACups\_p$H,  
 y = BACups\_s$H,  
 alternative = "two.sided")  
#>   
#> F test to compare two variances  
#>   
#> data: BACups\_p$H and BACups\_s$H  
#> F = 1.4551, num df = 19, denom df = 39, p-value = 0.3158  
#> alternative hypothesis: true ratio of variances is not equal to 1  
#> 95 percent confidence interval:  
#> 0.6942474 3.4016331  
#> sample estimates:  
#> ratio of variances   
#> 1.455127

Die Alternativhypothese ist, dass das Verhältnis der beiden Varianzen zueinander nicht 1 entspricht – sie also unterschiedlich sind.

Allerdings lässt der p-Wert keine Annahme der Alternativhypothese zu…

## t-test

Wir testen jetzt, ob sich die Mittelwerte so weit unterscheiden, dass wir von zwei unterschiedlichen Gruppen ausgehen können:

t.test(x = BACups\_p$H,  
 y = BACups\_s$H,  
 alternative = "two.sided")  
#>   
#> Welch Two Sample t-test  
#>   
#> data: BACups\_p$H and BACups\_s$H  
#> t = -1.4401, df = 32.435, p-value = 0.1594  
#> alternative hypothesis: true difference in means is not equal to 0  
#> 95 percent confidence interval:  
#> -2.6309508 0.4509508  
#> sample estimates:  
#> mean of x mean of y   
#> 6.26 7.35

Es wurde ein Welch-Test durchgeführt, weil die Varianzen nicht gleich sind (auch wenn wir oben die Alternativhypothese nicht annehmen konnten, war das Ergebnis auch nicht eindeutig so, dass das Varianzverhältnis 1 entspricht).

Der p-Wert erlaubt es uns wieder nicht, die Alternativhypothese sicher anzunehmen.

Tcha. Das passiert manchmal.

Tatsächlich haben wir in der Archäologie selten normalverteilte Daten. Deswegen sind der Mann-Whitney-Test und der K-S-Test wichtiger für uns.