Solution Proposal: Exam WS 22/23

Name, Vorname:

Matrikelnummer:

1.

a) Import der Daten:

```
> exam.data <- read.csv("exam_data.csv") %>% as_tibble()
> exam.data
# A tibble: 1,137 x 4
       X exam                         attempt score
   <int> <chr>                          <int> <int>
 1     1 516647/Mathematics                 1    75
 2     2 193841/Data Bases                  2    49
 3     3 326761/Computer Networks           2    75
 4     4 230275/Data Bases                  1    73
 5     5 911920/OOP                         1    55
 6     6 840873/Formal Languages            1    33
 7     7 473291/Mathematics                 2    56
 8     8 412501/Mathematics                 2   102
 9     9 320446/Formal Languages            2    26
10    10 332749/Formal Languages            1    67
# ... with 1,127 more rows
# i Use `print(n = ...)` to see more rows
```

b) Determine the type and the scale of all variables:

mat.nr: qualitative, nominal

gender: qualitative, nominal

semester: quantitative, ratio

course: qualitative, nominal

exam: qualitative, nominal

attempt: quantitative, ratio

score: quantitative, ordinal

c) Add a variable grade that indicates the grade of the exam.

```
> exam.data %>%
+   mutate(
+     grade = case_when(
+       score < 50 ~ 5,
+       score >= 50 & score < 65 ~ 4,
+       score >= 65 & score < 80 ~ 3,
+       score >= 80 & score < 90 ~ 2,
+       score >= 90 ~ 1
+     )
+   ) -> exam.data
> exam.data
# A tibble: 1,137 x 5
       X exam                         attempt score grade
   <int> <chr>                          <int> <int> <dbl>
 1     1 516647/Mathematics                 1    75     3
 2     2 193841/Data Bases                  2    49     5
 3     3 326761/Computer Networks           2    75     3
 4     4 230275/Data Bases                  1    73     3
 5     5 911920/OOP                         1    55     4
 6     6 840873/Formal Languages            1    33     5
```

```
 7      7 473291/Mathematics            2    56     4
 8      8 412501/Mathematics            2   102     1
 9      9 320446/Formal Languages       2    26     5
10     10 332749/Formal Languages       1    67     3
# ... with 1,127 more rows
# i Use `print(n = ...)` to see more rows
```

d) Split the variable exam into 2 columns

```
> exam.data
# A tibble: 1,137 x 6
       X mat.nr exam              attempt score grade
   <int> <chr>  <chr>               <int> <int> <dbl>
 1     1 516647 Mathematics             1    75     3
 2     2 193841 Data Bases              2    49     5
 3     3 326761 Computer Networks       2    75     3
 4     4 230275 Data Bases              1    73     3
 5     5 911920 OOP                     1    55     4
 6     6 840873 Formal Languages        1    33     5
 7     7 473291 Mathematics             2    56     4
 8     8 412501 Mathematics             2   102     1
 9     9 320446 Formal Languages        2    26     5
10    10 332749 Formal Languages        1    67     3
# ... with 1,127 more rows
# i Use `print(n = ...)` to see more rows
```

e) Determine the total number of tests in each exam and the number of students participating

```
Number of Tests:
> exam.data %>%
+   select(mat.nr) %>%
+   unique() %>%
+   summarise(anz.stud = n())
# A tibble: 1 x 1
  anz.stud
     <int>
1      243

Number of students in each exam
> exam.data %>%
+   group_by(exam) %>%
+   summarise(n = n())
# A tibble: 6 x 2
  exam                     n
  <chr>                <int>
1 Computer Networks      186
2 Data Bases             190
3 Formal Languages       190
4 Mathematics            185
5 OOP                    196
6 Software Engineering   190
```

f) For each subject, determine the absolute frequencies of the grades and store the result in a tibble with the variables grade, Computer Networks`, `Data Bases`, `Formal Languages`, Mathematics, OOP and `Software Engineering`.

```
> exam.data %>%
+   count(exam,grade) %>%         # 2 Punkte
+   spread(key = exam, value = n) # 3 Punkte
# A tibble: 5 x 7
  grade `Computer Networks` `Data Bases` `Formal Languages` Mathematics    O
OP `Software Engineering`
  <dbl>               <int>        <int>              <int>       <int> <in
t>                  <int>
1     1                   6           10                  4           4
4                      1
```

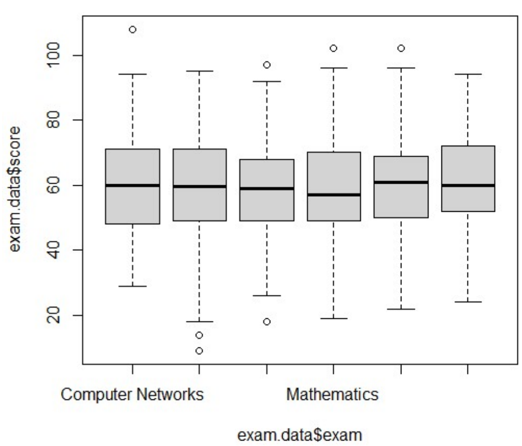
```
2       2             8           12              6           16
15                   19
3       3             58          46              57          43
62                   57
4       4             53          69              75          72
69                   76
5       5             61          53              48          50
46                   37
```

f) For each subject, determine the minimum, maximum, the three quartiles, the mean of the variable score, the number of participants and the dropout rates.

```
> exam.data %>%
+   mutate(
+     fail = if_else(grade <= 4,0,1)
+   ) %>%
+   group_by(exam) %>%
+   summarise(
+     Min = min(score),
+     Max = max(score),
+     Q1 = quantile(score, probs = 0.25),
+     Q2 = quantile(score, probs = 0.5),
+     Q3 = quantile(score, probs = 0.75),
+     Mean = mean(score),
+     no.participants = n(),
+     dropout.rate = sum(fail)/no.participants
+   )
# A tibble: 6 x 9
  exam                   Min   Max    Q1    Q2    Q3  Mean no.participants
dropout.rate
  <chr>                <int> <int> <dbl> <dbl> <dbl> <dbl>           <int>
<dbl>
1 Computer Networks       29   108  48    60    70.8  59.4             186
0.328
2 Data Bases               9    95  49    59.5  71    60.3             190
0.279
3 Formal Languages        18    97  49.2  59    68    58.9             190
0.253
4 Mathematics             19   102  49    57    70    58.6             185
0.270
5 OOP                     22   102  50    61    69    60.3             196
0.235
6 Software Engineering    24    94  52    60    71.8  61.2             190
0.195
```

h) Create side by side boxplots of the score for each subject and interpret the results.

```
> boxplot(exam.data$score ~ exam.data$exam)
```



no differences between the subjects

I) Determine the contingency table of the variables attempt and grade and evaluate the indifference table and chi-square value.

```
> chisq.test(exam.data$attempt, exam.data$grade)$observed %>% addmargins()
                 exam.data$grade
exam.data$attempt    1    2    3    4    5  Sum
              1     14   49  206  226  184  679
              2     14   21   84  142   84  345
              3      1    6   33   46   27  113
              Sum   29   76  323  414  295 1137
> chisq.test(exam.data$attempt, exam.data$grade)$expected %>% addmargins()
                 exam.data$grade
exam.data$attempt         1        2         3         4         5  Sum
              1   17.318382 45.38610 192.89094 247.23483 176.16974  679
              2    8.799472 23.06069  98.00792 125.62005  89.51187  345
              3    2.882146  7.55321  32.10114  41.14512  29.31838  113
              Sum 29.000000 76.00000 323.00000 414.00000 295.00000 1137
> chisq.test(exam.data$attempt, exam.data$grade)$statistic
X-squared
 14.05124
```

---

2) A biased coin (head with probability 1/3) is tossed. If the coins shows tail a fair die is rolled 5 times and if the coin shows head a biased die (6 with probability 0.4) is rolled 5 times. The number of sixes are counted.

a) Determine the density of the random X which counts the number of sixes.

In case of head X ~ B(n=5, p=0.4) and in case of a tail X~B(n=5, p=1/6). Weg et

P(X=i) = P(X=i | head) / 3 + P(X=i | tail) * 2/3

```
> random.exp <- tibble(
+   no = 0:5,
+   dens.head = dbinom(no, size = 5, prob = 0.4),
+   dens.tail = dbinom(no, size = 5, prob = 1/6),
+   dens = dens.head/3 + 2*dens.tail/3,
+ )
> random.exp
# A tibble: 6 x 4
     no dens.head dens.tail    dens
  <int>     <dbl>     <dbl>   <dbl>
1     0    0.0778  0.402    0.294
2     1    0.259   0.402    0.354
3     2    0.346   0.161    0.222
4     3    0.230   0.0322   0.0982
5     4    0.0768  0.00322  0.0277
6     5    0.0102  0.000129 0.00350
```

b) Evaluate the expected value and the variance of the random variable X.

```
> EX <- sum(random.exp$no * random.exp$dens)
> EX2 <- sum(random.exp$no^2 * random.exp$dens)
> VarX <- EX2 - EX^2
> EX; VarX
[1] 1.222222
[1] 1.165432
```

c) What is the probability that the coin had shown a head if 3 sixes has been in the 5 rolls?

From the Bayes Theoren weg et

P(head | X=3) = P(head and X=3) / P(X=3) = P(X=3 | head) * P(head) / P(X=3) =

dbinom(3,size=5,prob=0.4) * (1/3) / 0.0982 = 0.782

---

3) The weight of bags of grain can be assumed to be random variable with expected value 50 kg and standard deviation 2 kg. The price for one kilogram grain, which a farmer achieves, is 0.53 Euro per kg. 300 bags of grain fit into a truck. Let X be the price a farmer can obtain for a fully loaded truck.

a) Determine an approximate distribution of the random variable X.

Applying the central limit theorem we get X approximately

N(mu = 0.53*300*50, sigma^2 = 300*0.53^2*2^2)

b) Find the probability that X is bigger than 8000 Euro.

1-pnorm(8000, mean = mu, sd = sigma) = 0.003231175

c) What are the lower and upper bounds of the interval containing the middle 80% of X?

qnorm(c(0.1,0.9), mean = mu, sd = sigma) = c(7926.471, 7973.529)

d) What is the minimum number of bags a farmer must sell to earn at least 20000 euros with a probability of 95%?

Analytic solution:

P(X_1+…+X_n >= 20000) = 0.95, i.e. 20000 is approximately the 5% quantil of

N(n*0.53*50, n*0.53^2*2^2). From the relationship between quantile of an arbrtrary and standard normal distribution we get

20000 = n * 0.53 * 50 + 2*0.53* sqrt(n) * qnorm(0.05)

With u = sqrt(n) we have the equation

20000 = u^2 * 0.53 * 50 + 2*0.53* u * qnorm(0.05). The solutions are

```
c(-qnorm(0.05)/50 - sqrt(-(qnorm(0.05)/50)^2 + 20000/(50*0.53)),
+   -qnorm(0.05)/50 + sqrt(-(qnorm(0.05)/50)^2 + 20000/(50*0.53)))^2
[1] 752.9095 756.5245
Since u must be positive we get n = 27.50499^2 = 756.5245
```

```
> tibble(
+   n.bags = 300:900,
+   p.bigger.20000 = 1-pnorm(20000,
+                            mean = n.bags*0.53*50,
+                            sd = (0.53^2*2^2*n.bags)^0.5)
+ ) %>%
+   filter(p.bigger.20000 > 0.95) %>%
+   summarise(min.n.bags = min(n.bags))
# A tibble: 1 x 1
  min.n.bags
       <int>
1        757
```

4) In a representative sample 100 people were asked if they prefer candidate A or not if the election were held next Sunday.

a) Determine the relative frequency of voter prefering A.

```
survey <- c(1, 0, …, 1, 0, 1, 0, 0, 0)
> n <- length(survey)
> p.hat <- sum(survey)/n
> p.hat
[1] 0.32
```

b) Show that the relative frequency is an unbiased point estimators for the proportion of voters preferring A in the whole population.

Let p be the probability that a randomly chosen voter supports A. Then X = # number of voters preferring A in a sample of size n follows a B(n, p)-distribution. The expected value of the relative frequency

p.hat = X / n is n*p_i / n = p_i, i.e. p.hat is an unbiased estimator of p.

c) Determine a normal approximation and the exact two-sided 95% confidence interval of the unknown proportions p.

```
> alpha <- 0.05
> X <- sum(survey)
> n <- length(survey)
> L.approx <- p.hat-qnorm(1-alpha/2,mean=0, sd=1)*sqrt(p.hat*(1-p.hat)/n)
> U.approx <- p.hat+qnorm(1-alpha/2,mean=0, sd=1)*sqrt(p.hat*(1-p.hat)/n)
> # approx. CI
> c(L.approx, U.approx)
[1] 0.2285724 0.4114276

Approximation auch mittels t-test möglich
> t.test(survey, alternative = "two.sided", conf.level = 1-alpha)

        One Sample t-test

data:  survey
t = 6.8256, df = 99, p-value = 7.095e-10
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 0.2269747 0.4130253
sample estimates:
mean of x
     0.32
```

d) Find an approximate value of the minimal sample size so that the length of the approximate confidence interval for p is less the 0.1.

Length of the confidence interval

L = 2*sqrt(p.hat*(1-p.hat)/n)*qnorm(1-alpha/2) <=

<= 2*sqrt(0.5*0.5/n)*qnorm(1-alpha/2) = sqrt(1/n)*qnorm(1-alpha/2) <= 0.1

```
ceiling((qnorm(1-alpha/2, mean = 0, sd = 1)/0.1)^2)
[1] 385
```

e) Determine the confidence level so that the width of the approximate confidence interval of p is equal to 0.1.

From L = 2*sqrt(p.hat*(1-p.hat)/n)*qnorm(1-alpha/2) = 0.1 ->

0.1 / (2*sqrt(p.hat*(1-p.hat)/n)) = qnorm(1-alpha/2) = 0.1 -> 1-alpha =

```
2*pnorm(0.1/(2*sqrt(p.hat*(1-p.hat)/n)))-1 # 0.7162198
[1] 0.7162198
```

---

5)

a) What are the null hypothesis and alternative for an appropriate statistical test?

H0: mu <= 5, H1: mu > 5

b) Perform an appropriate statistical test to verify the null hypothesis at the 5% level. What is the test decision and what is the p-value?

one sample t-test

```
> t.test(x=X, alternative = "greater", mu = 5, conf.level = 0.95)

        One Sample t-test

data:  X
t = 3.8881, df = 19, p-value = 0.0004947
alternative hypothesis: true mean is greater than 5
95 percent confidence interval:
 5.095785      Inf
sample estimates:
mean of x
   5.1725
```

Since p-value <= 0.05 H0 can be rejected.

c) In making the above conclusion, which type of error are you risking, type I or type II?

type I error

d) Perform an appropriate statistical test to verify the conjecture at the 5% level and determine the pvalue.

```
> Y <- c(5.11, 5.20, 4.91, 4.94, 5.22, 4.59, 5.45, 4.80, 5.13, 4.79,
+        4.69, 4.14, 4.64, 5.20, 5.16, 5.49, 4.94, 5.75, 5.36, 4.69)
```

Since each car is used in each sample, we can use a paired samples t-test to determine if the mean co nsumption has been changed by the fuel treatment.

```
> t.test(x=X, y=Y,alternative = "two.sided", paired = TRUE, var.equal = TRU
E,
+        conf.level = 0.95)

        Paired t-test

data:  X and Y
t = 2.2711, df = 19, p-value = 0.03496
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01274405 0.31225595
sample estimates:
mean of the differences
                 0.1625
```

Since the the p-value is lower than alpha the Null-Hypothesis that there is no difference can be rejected.