Solution for the assignment of the second class

Kovacs Marton

9/21/2021

Importing data

```
raw <- read_tsv("data/boldog.txt", locale = locale(encoding = "UTF-8"))
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double()
## )
## i Use `spec()` for the full column specifications.
```

R sometimes have a problem with special characters. I will fix that now.

```
# first I fix the special Hungarian characters
Encoding(names(raw)) <- 'latin1'
# I translate them to English letters for the sake of simplicity
iconv(names(raw), from = "latin1", to = "ASCII//TRANSLIT")
```

```
##  [1] "index"    "Neme"     "Eletkora" "Gyermeke" "Isk"      "Anyagi"
##  [7] "PElmeny%" "Testi-Fi" "AltLelki" "AltEgAll" "Fizero"   "Arcocska"
## [13] "Aggodalo" "Ideges"   "Feszult"  "Nyugtala" "Diener1"  "Diener2"
## [19] "Diener3"  "Diener4"  "Diener5"  "Diener6"  "Diener7"  "Diener8"
## [25] "Jollet"   "Savor"    "AVhat"    "Onreg"    "Rezil"    "M_Flow"
## [31] "GJerz"    "GJpszi"   "GJszoc"   "GJspir"   "GJerzpsz" "GJspszoc"
## [37] "PIK_MM"   "PIK_AV"   "PIK_Onr"  "PIK_Rez"  "PPozErz"  "PElmely"
## [43] "PPozKapc" "PErtCel"  "PTelj"    "PBoldog"  "PEgeszs"  "PNegErz"
## [49] "PMagany"  "PERMA"
```

Also, I like to use *snake_case* for my variable names, therefore I will transform the variable names.

```
raw <- janitor::clean_names(raw)
```

Data exploration

```
skimr::skim(raw) %>%
  kable()
```

skim_type	skim_variable	n_missing	complete_rate	numeric.mean	numeric.sd	numeric.p0	numeric.p2
numeric	index	0	1.000	972.344000	486.4327160	3.0	598.0000
numeric	neme	0	1.000	1.500000	0.5005008	1.0	1.0000
numeric	eletkora	0	1.000	52.522000	11.7257196	18.0	45.0000
numeric	gyermeke	0	1.000	1.750000	1.2371585	0.0	1.0000
numeric	isk	0	1.000	2.786000	0.8328212	1.0	2.0000
numeric	anyagi	0	1.000	3.100000	0.5644796	1.0	3.0000
numeric	p_elmeny_percent	0	1.000	58.460000	21.4183541	10.0	40.0000
numeric	testi_fi	3	0.994	4.164990	0.9572209	1.0	4.0000
numeric	alt_lelki	4	0.992	4.161290	1.0722362	1.0	4.0000
numeric	alt_eg_all	7	0.986	4.121704	1.1195761	1.0	3.0000
numeric	fizero	6	0.988	4.153846	1.1867317	1.0	3.0000
numeric	arcocska	0	1.000	2.570000	1.1435539	1.0	2.0000
numeric	aggodalo	0	1.000	2.772000	1.4325223	1.0	2.0000
numeric	ideges	0	1.000	2.490000	1.3746561	1.0	1.0000
numeric	feszult	0	1.000	2.610000	1.3992197	1.0	1.0000
numeric	nyugtala	0	1.000	2.406000	1.4578803	1.0	1.0000
numeric	diener1	0	1.000	5.514000	1.3286590	1.0	5.0000
numeric	diener2	0	1.000	5.228000	1.3219346	1.0	4.0000
numeric	diener3	0	1.000	5.574000	1.2310788	1.0	5.0000
numeric	diener4	0	1.000	5.418000	1.2611332	1.0	5.0000
numeric	diener5	0	1.000	5.756000	1.0727374	1.0	5.0000
numeric	diener6	0	1.000	5.768000	1.1245912	1.0	5.0000
numeric	diener7	0	1.000	5.620000	1.3738176	1.0	5.0000
numeric	diener8	0	1.000	5.532000	1.2148777	1.0	5.0000
numeric	jollet	0	1.000	4.488666	1.0610844	1.0	4.0000
numeric	savor	0	1.000	4.514667	1.0133860	1.0	4.0000
numeric	a_vhat	0	1.000	4.595600	0.9393964	1.0	4.0000
numeric	onreg	0	1.000	4.217332	1.2217368	1.0	3.3333
numeric	rezil	0	1.000	4.044667	1.0210453	1.0	3.3333
numeric	m_flow	0	1.000	4.773997	0.8909798	1.0	4.3333
numeric	g_jerz	0	1.000	4.152800	1.0313775	1.0	3.4000
numeric	g_jpszi	0	1.000	4.282000	0.9892843	1.0	3.7500
numeric	g_jszoc	0	1.000	3.949500	1.1789209	1.0	3.0000
numeric	g_jspir	0	1.000	4.245500	1.2003067	1.0	3.5000
numeric	g_jerzpsz	0	1.000	4.217400	0.9720234	1.1	3.6187
numeric	g_jspszoc	0	1.000	4.097500	1.1382101	1.0	3.3750
numeric	pik_mm	0	1.000	3.027334	0.6267487	1.0	2.6667
numeric	pik_av	0	1.000	3.175000	0.6232189	1.0	2.7500
numeric	pik_onr	0	1.000	3.064000	0.7912331	1.0	2.6667
numeric	pik_rez	0	1.000	3.150668	0.7424924	1.0	2.6667
numeric	p_poz_erz	0	1.000	21.742000	5.2751389	0.0	20.0000
numeric	p_elmely	0	1.000	22.360000	4.5585041	8.0	20.0000
numeric	p_poz_kapc	0	1.000	22.442000	5.5671017	0.0	19.0000
numeric	p_ert_cel	0	1.000	23.374000	4.6790931	0.0	21.0000
numeric	p_telj	0	1.000	22.716000	4.3420099	0.0	21.0000
numeric	p_boldog	0	1.000	7.404000	2.1217416	0.0	7.0000
numeric	p_egeszs	0	1.000	22.260000	5.2248443	0.0	19.0000
numeric	p_neg_erz	0	1.000	9.766000	5.7089779	0.0	5.0000
numeric	p_magany	0	1.000	3.162000	3.0565407	0.0	0.0000
numeric	perma	0	1.000	169.370000	31.2671131	22.0	155.0000

Transforming variables

```
processed <-
  raw %>%
  mutate(neme = case_when(neme == 1 ~ "ferfi",
                          neme == 2 ~ "no",
                          TRUE ~ NA_character_),
         isk = case_when(isk == 1 ~ "altalanos",
                         isk == 2 ~ "kozepiskola",
                         isk == 3 ~ "foiskola",
                         isk == 4 ~ "egyetem",
                         TRUE ~ NA_character_))
```

1. Which variables are intervalum scale variables?

Based on the name of the variables and on a paper focusing on the validation of the MET test (Vargha et al. (2020)), I decided that I would consider the following variables are on an interval scale:

```
# select variables
interval_vars <-
  processed %>%
  select(eletkora, jollet, 26:50) %>%
  names()

# print as a bullet list
cat(paste('-', interval_vars), sep = '\n')
```

- eletkora
- jollet
- savor
- a_vhat
- onreg
- rezil
- m_flow
- g_jerz
- g_jpszi
- g_jszoc
- g_jspir
- g_jerzpsz
- g_jspszoc
- pik_mm
- pik_av
- pik_onr
- pik_rez
- p_poz_erz
- p_elmely
- p_poz_kapc
- p_ert_cel
- p_telj
- p_boldog

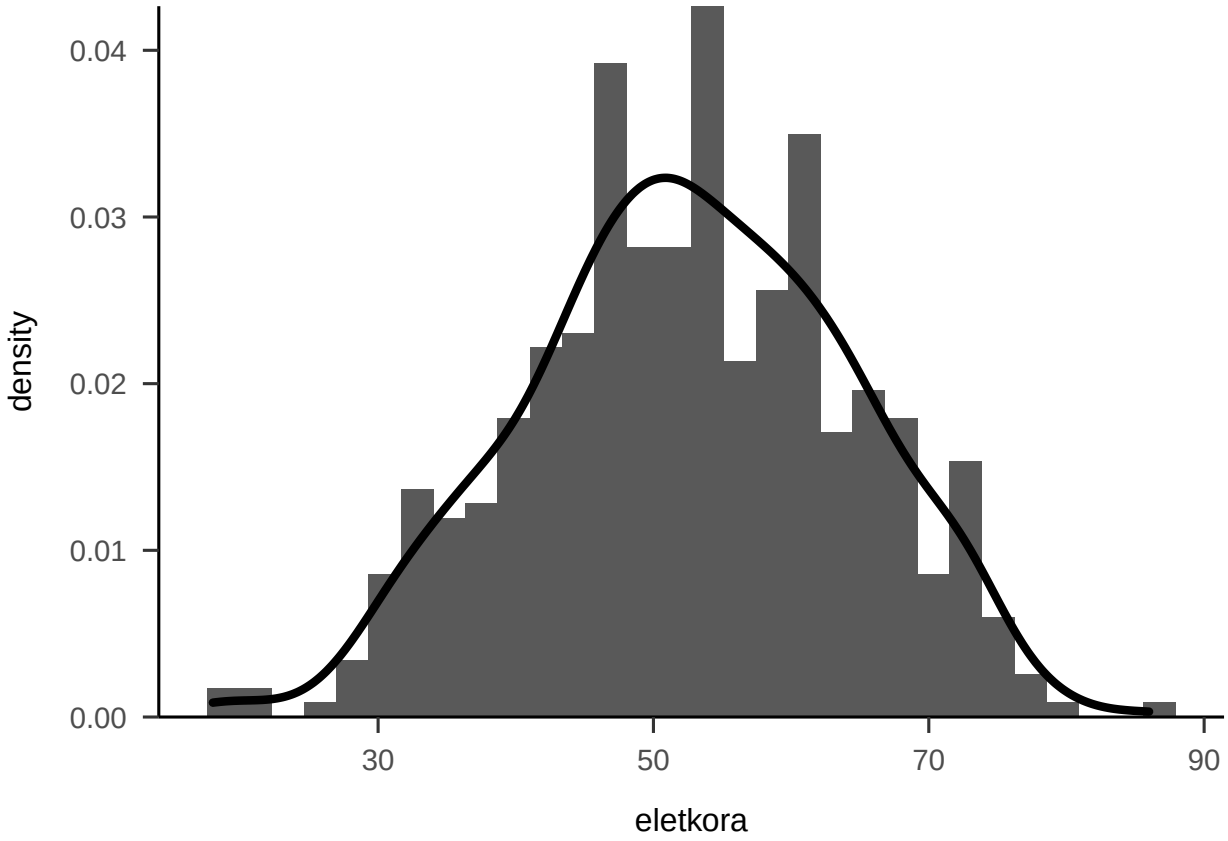
- p_egeszs
- p_neg_erz
- p_magany
- perma

2. & 3. Which of these variables are the least looking like a normal distribution, and which of these are normally distributed?

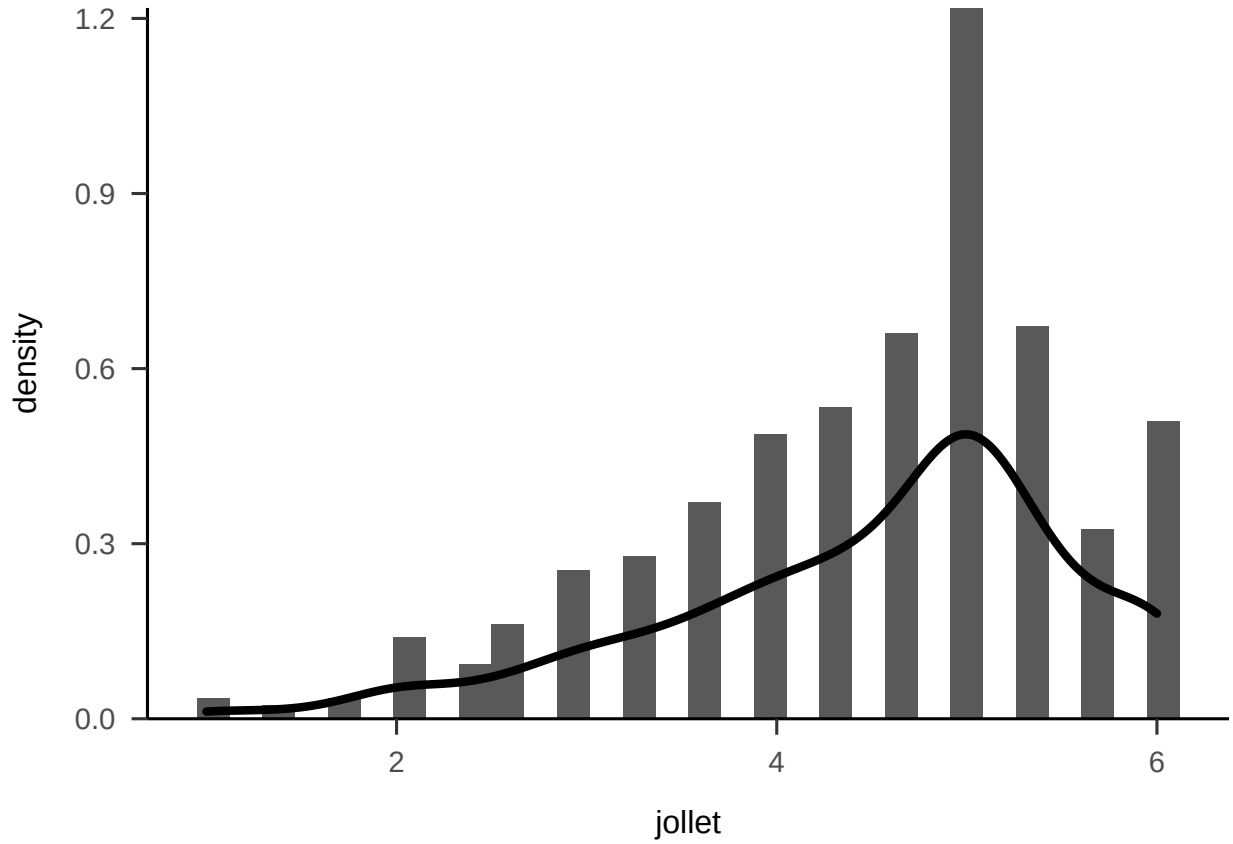
Plotting each variable.

```
map(interval_vars, ~ apa_hist(processed, .x))
```

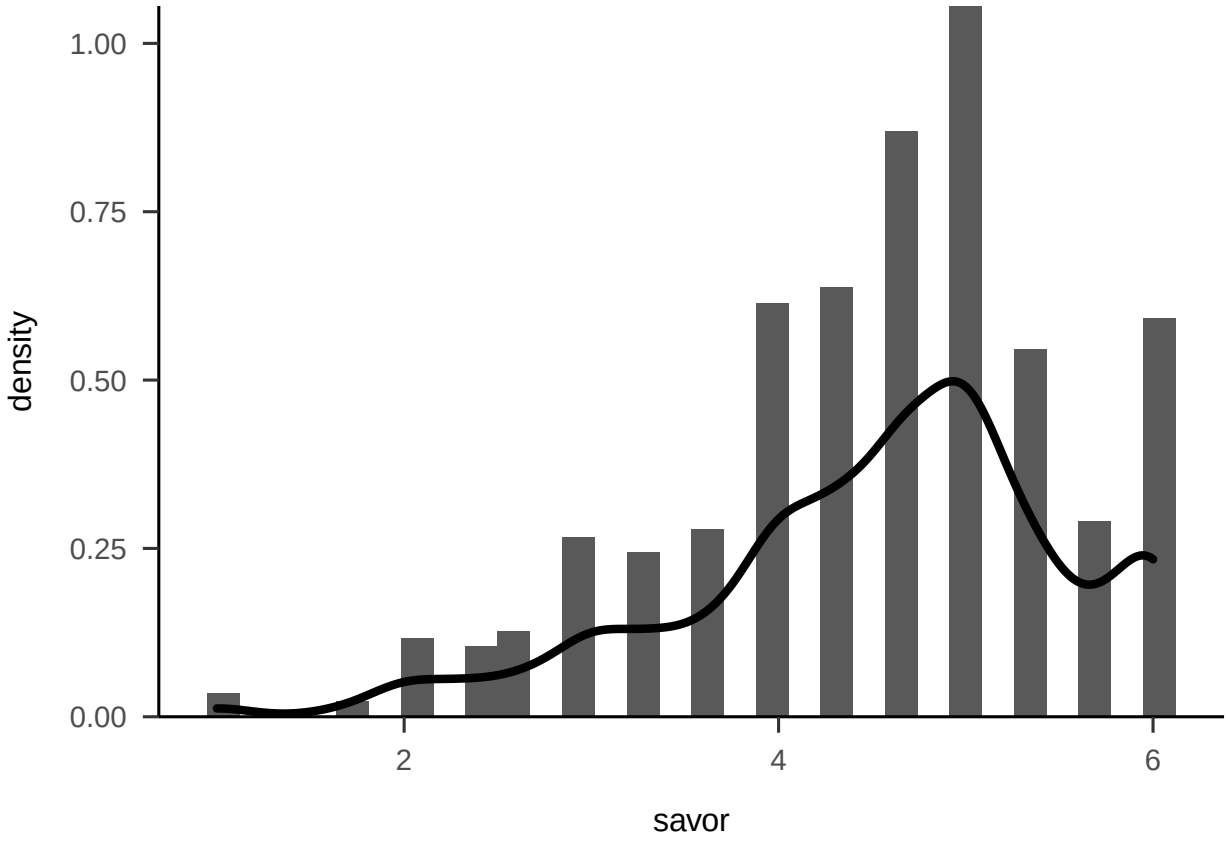
```
## [[1]]
```



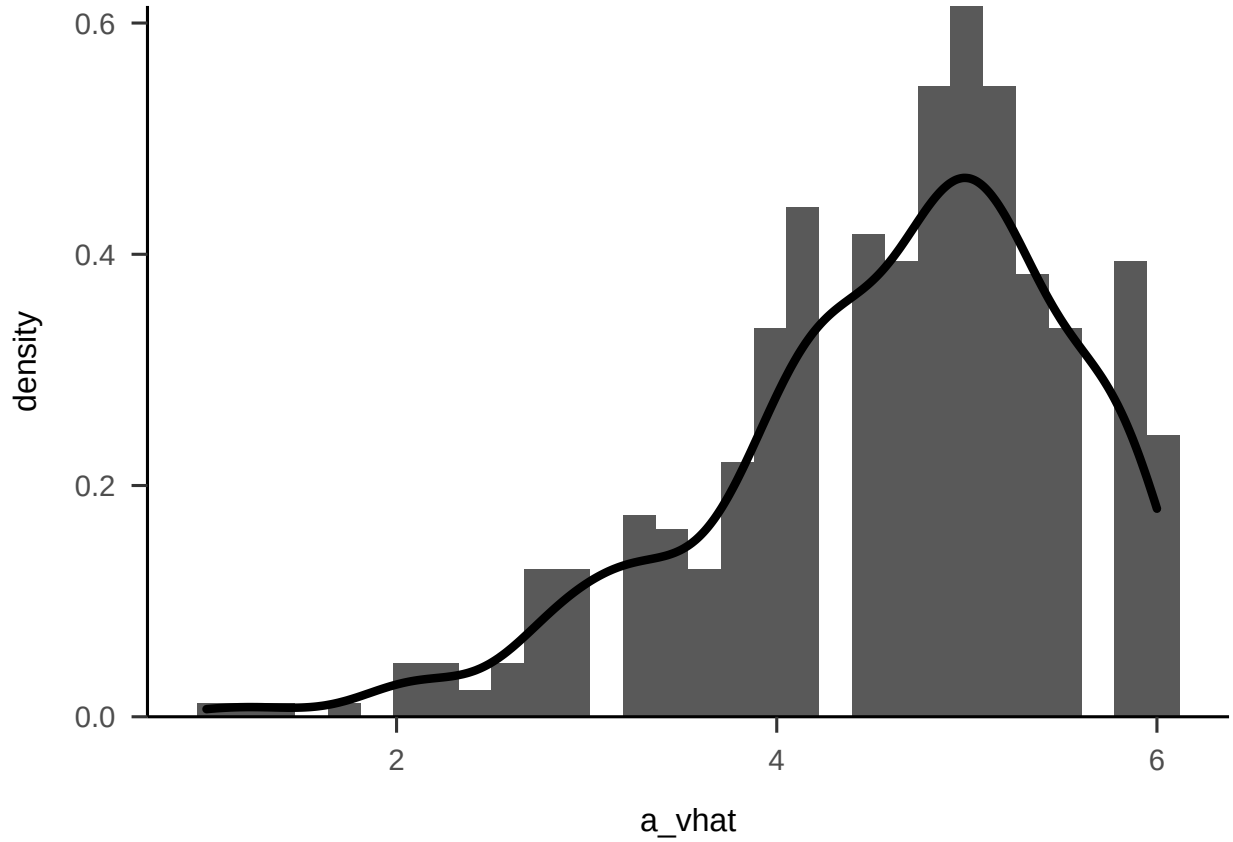

```
##
## [[2]]
```



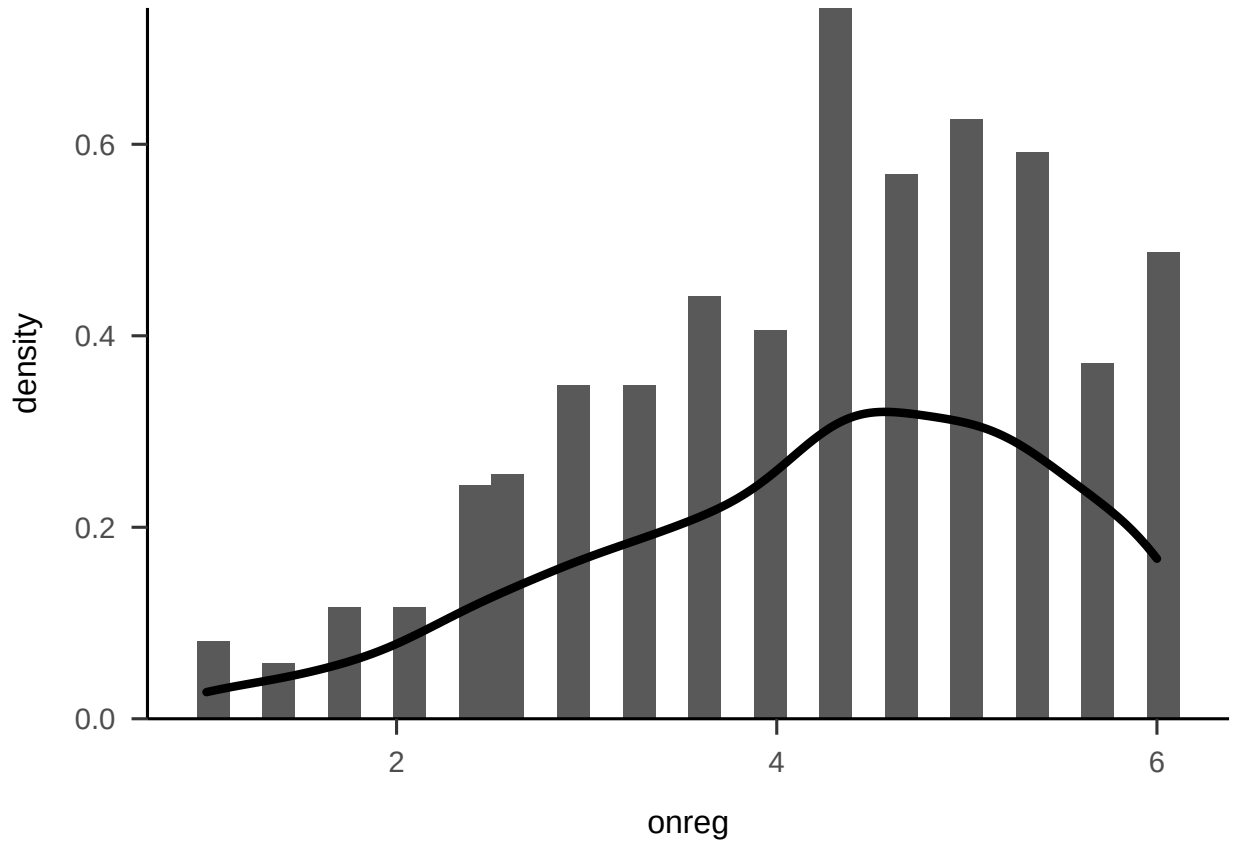

```
##
## [[3]]
```



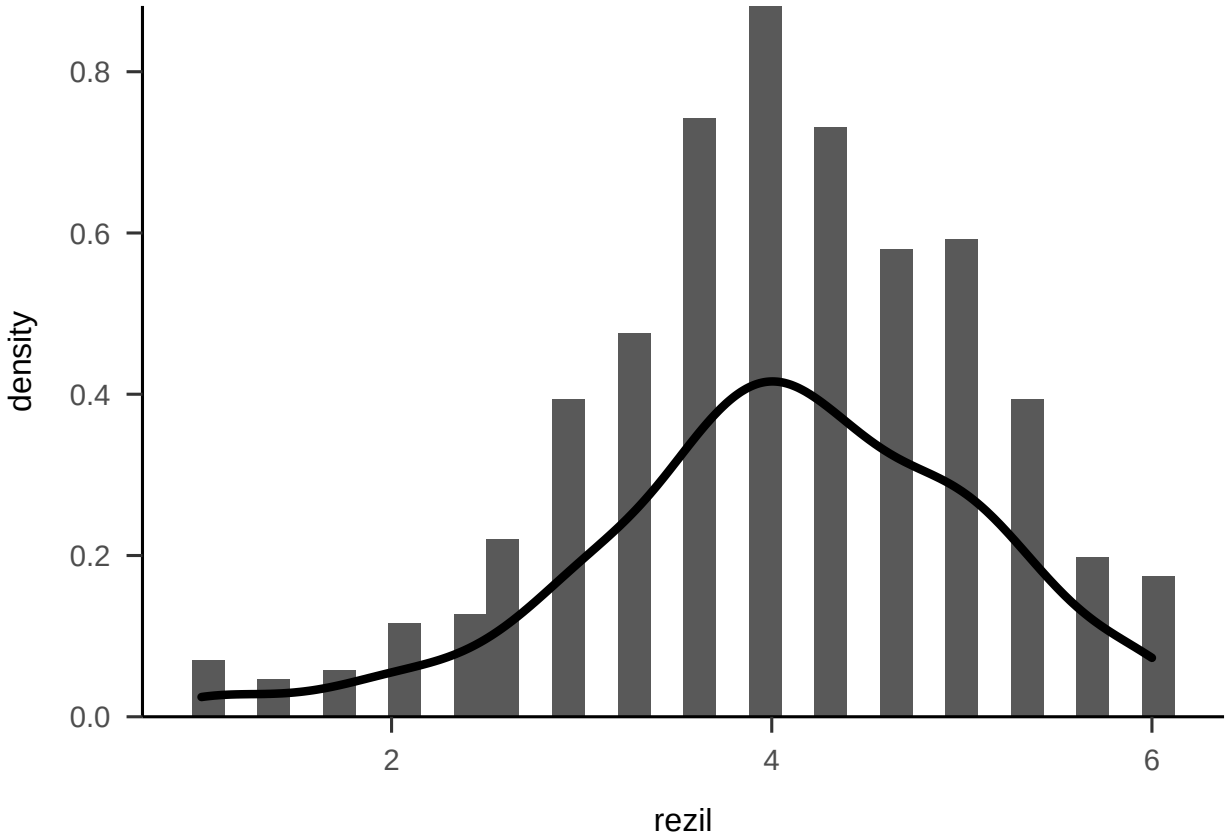

```
##
## [[4]]
```



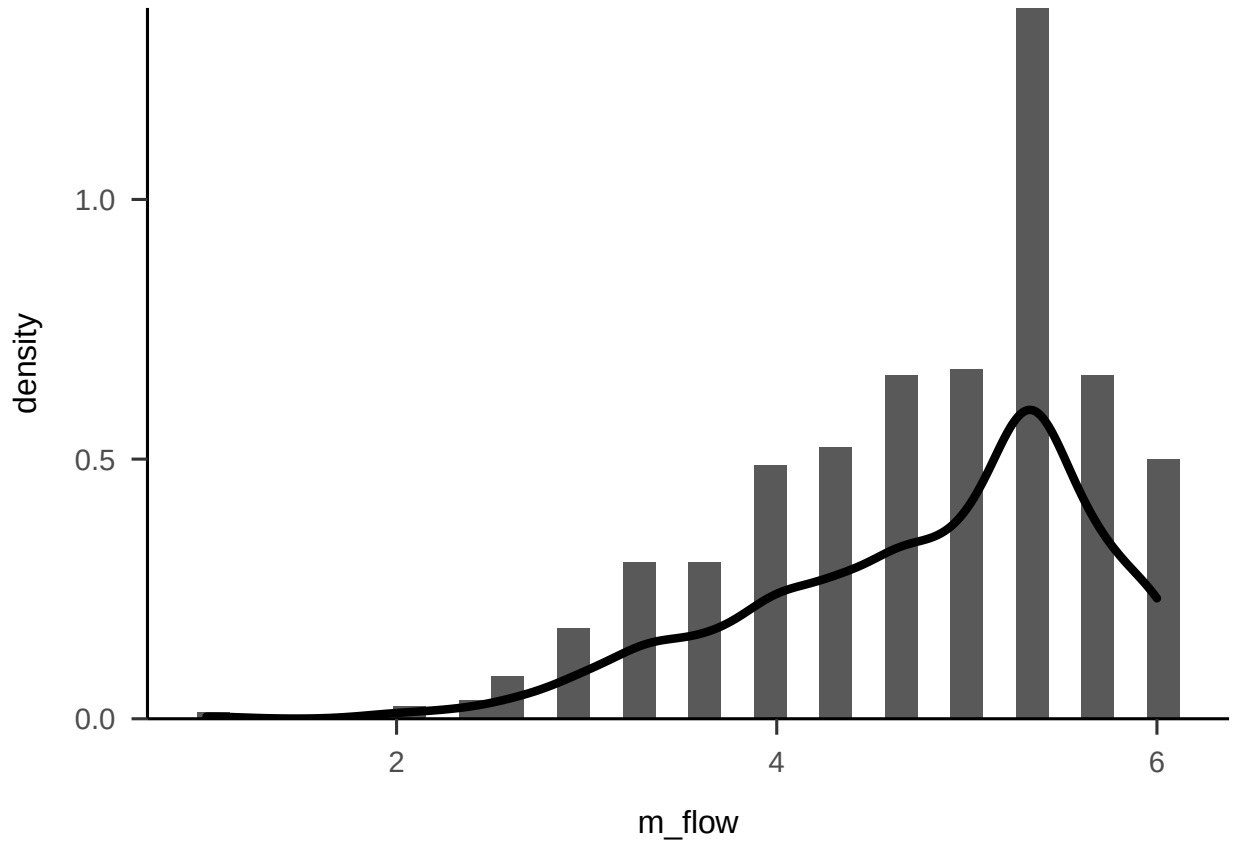

```
##
## [[5]]
```



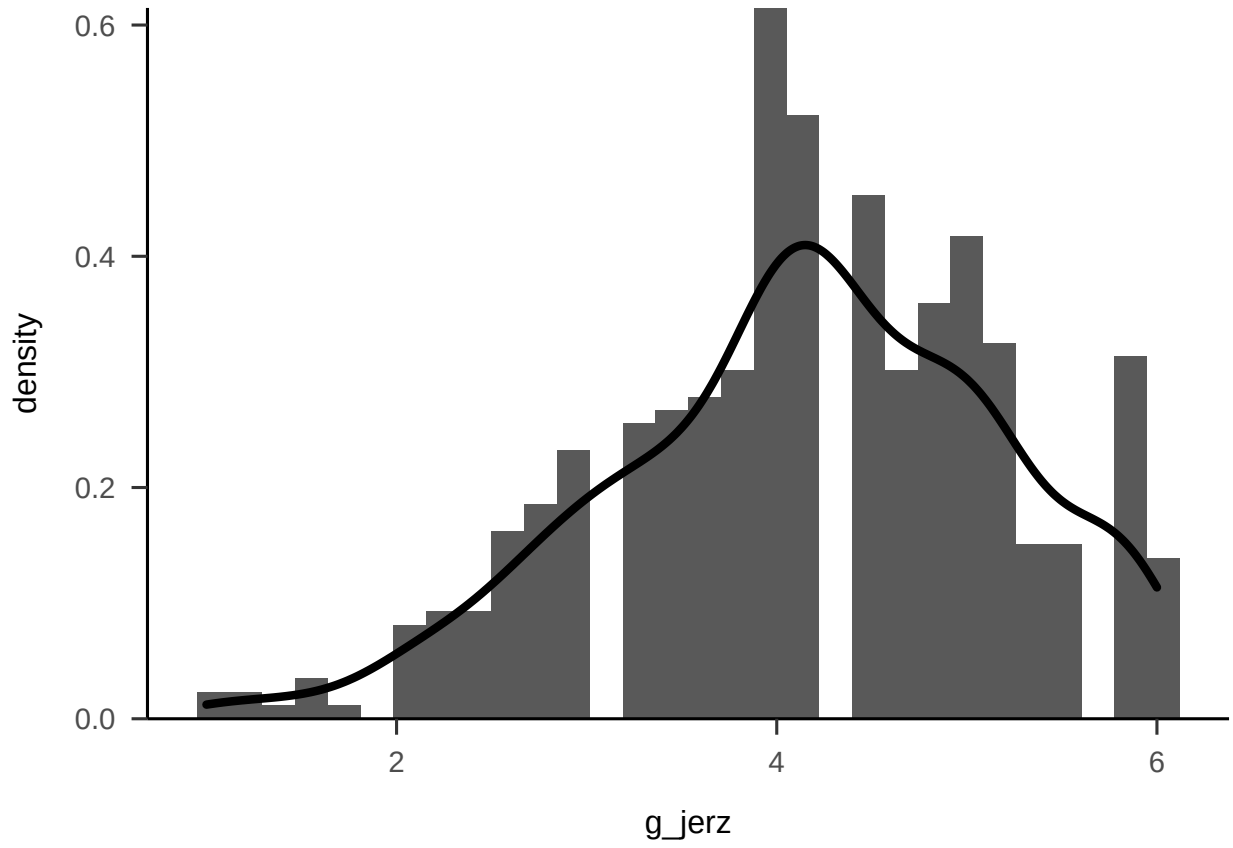

```
##
## [[6]]
```



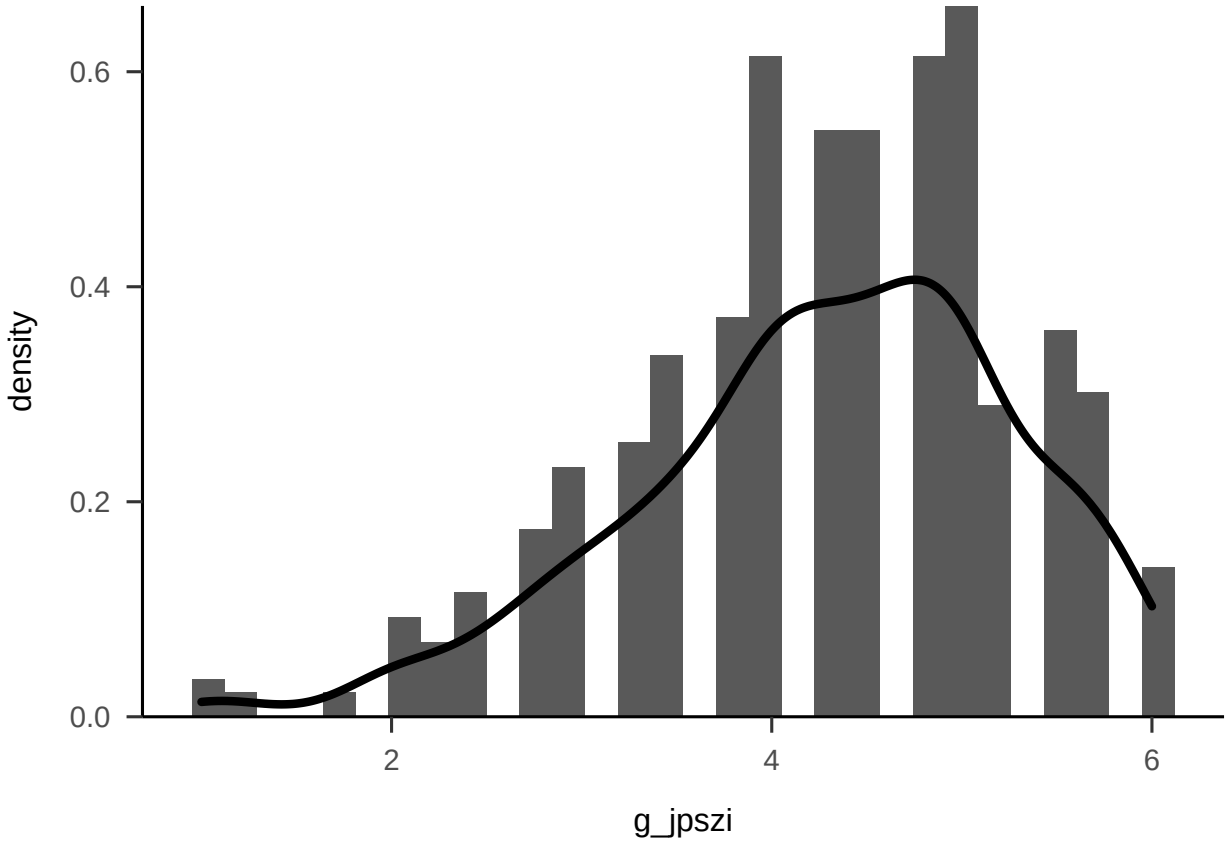

```
##
## [[7]]
```



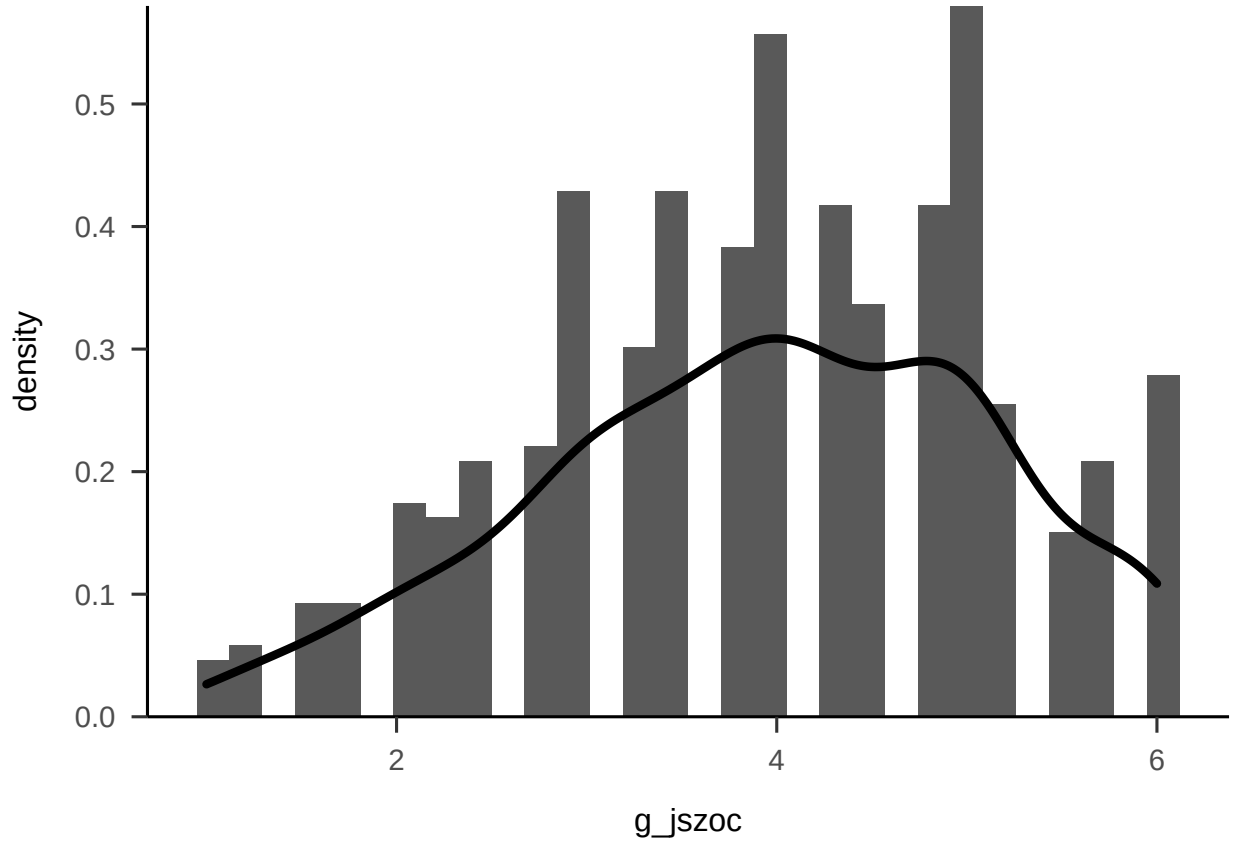

```
##
## [[8]]
```



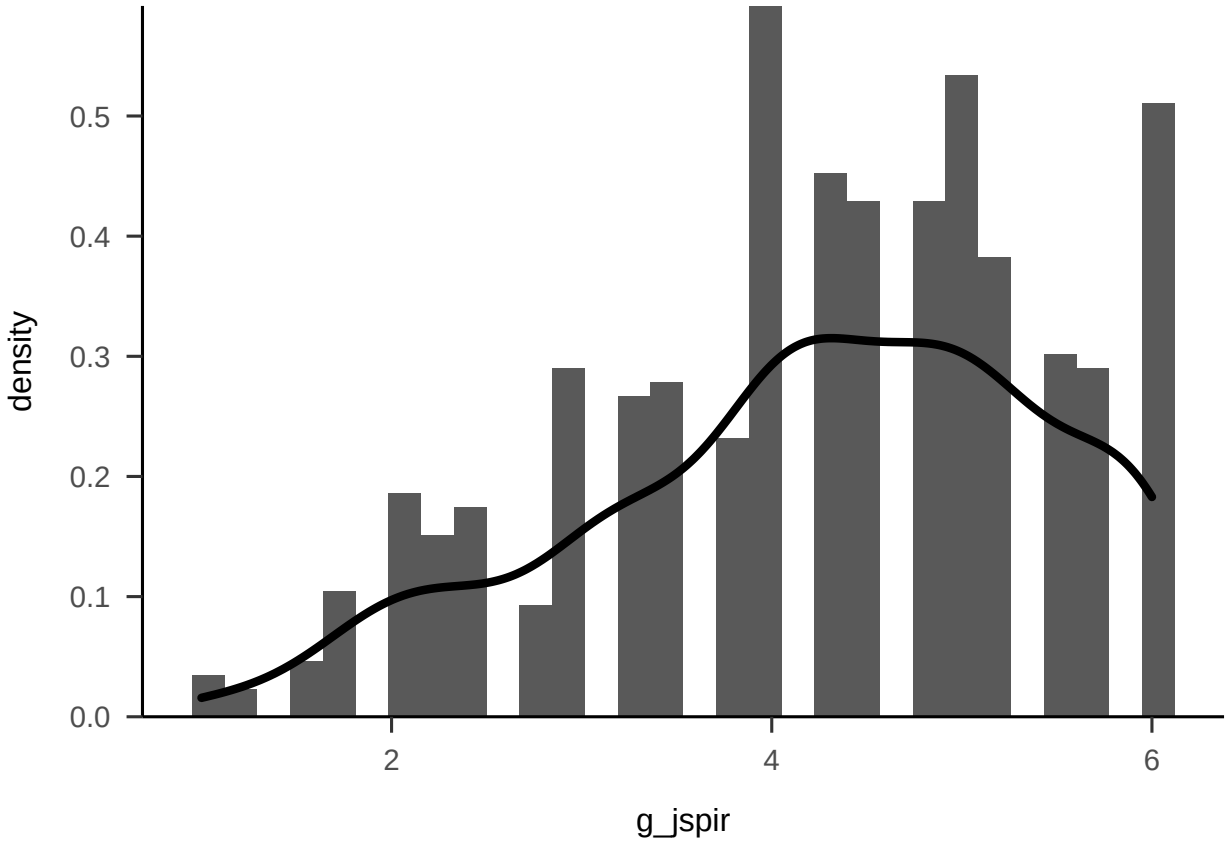

```
##
## [[9]]
```



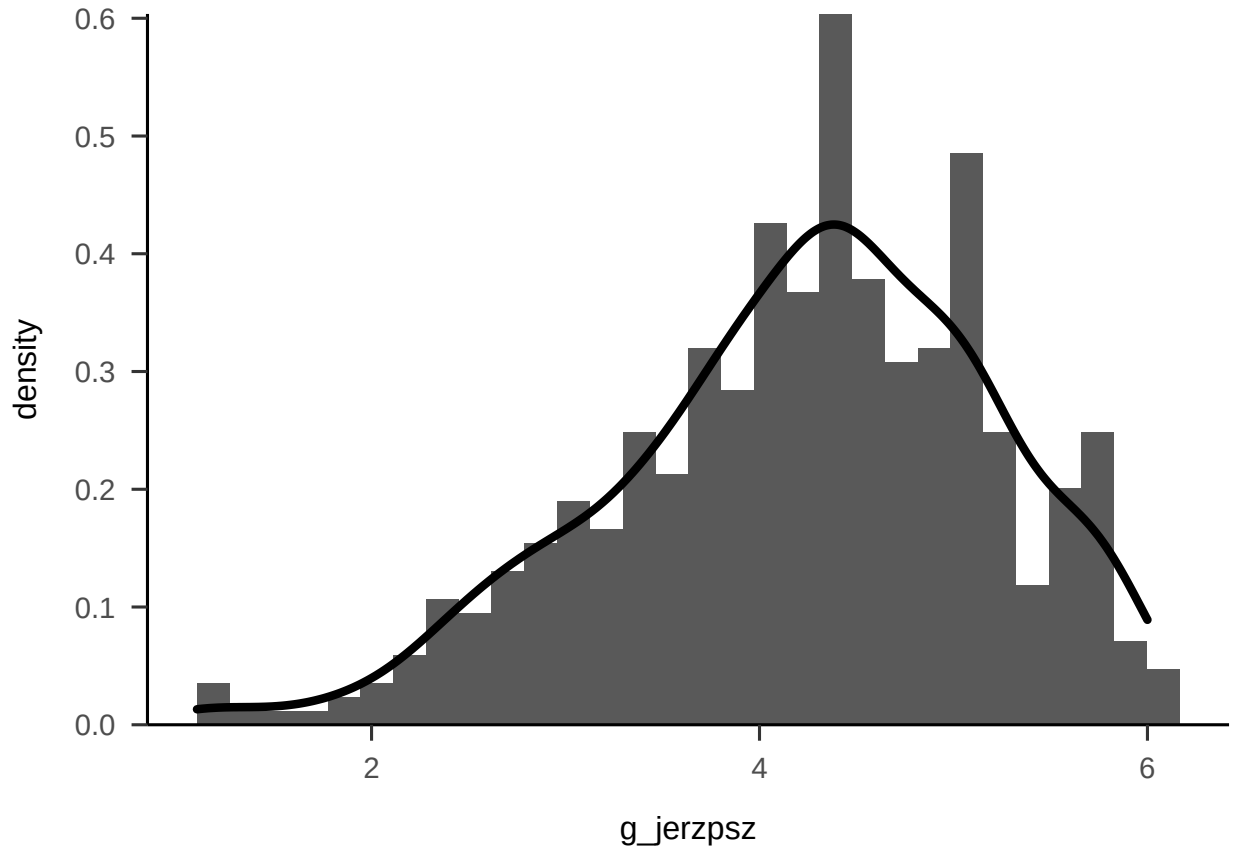

```
##
## [[10]]
```



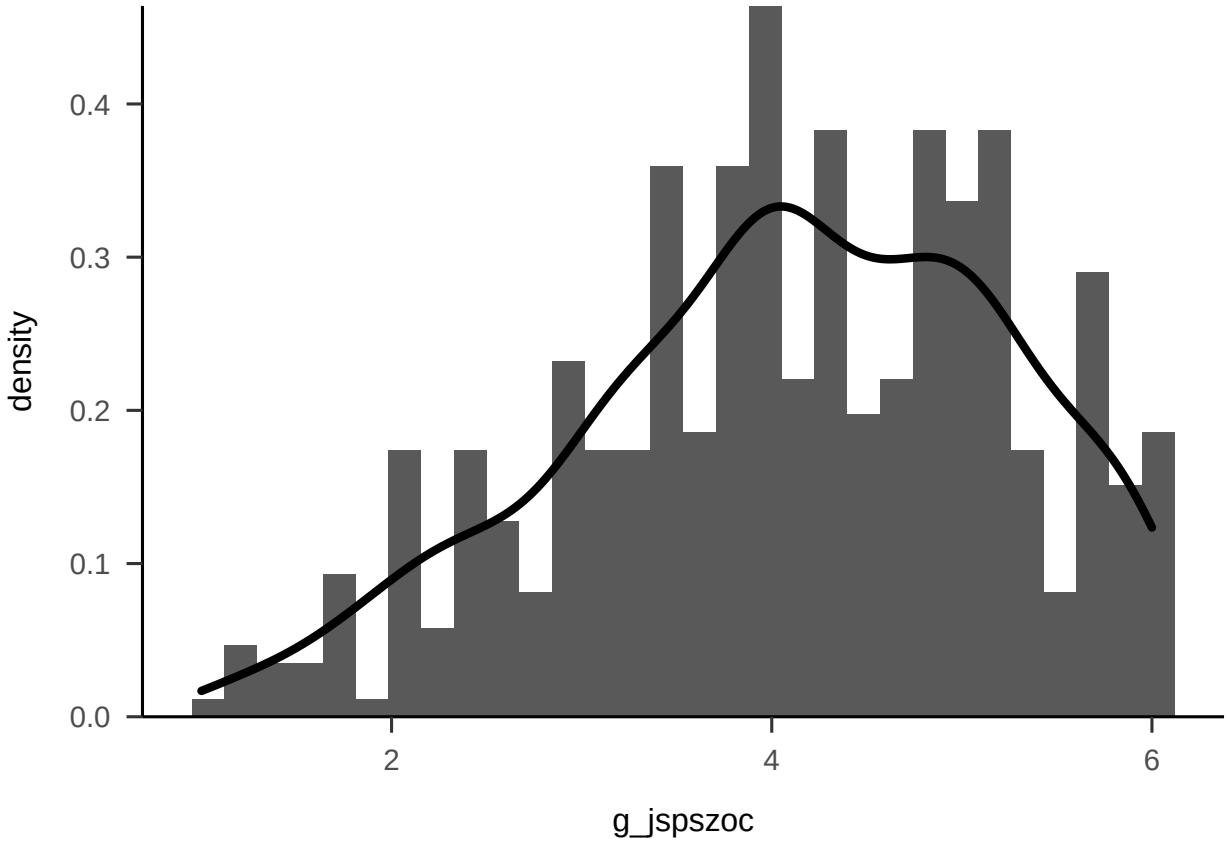

```
##
## [[11]]
```



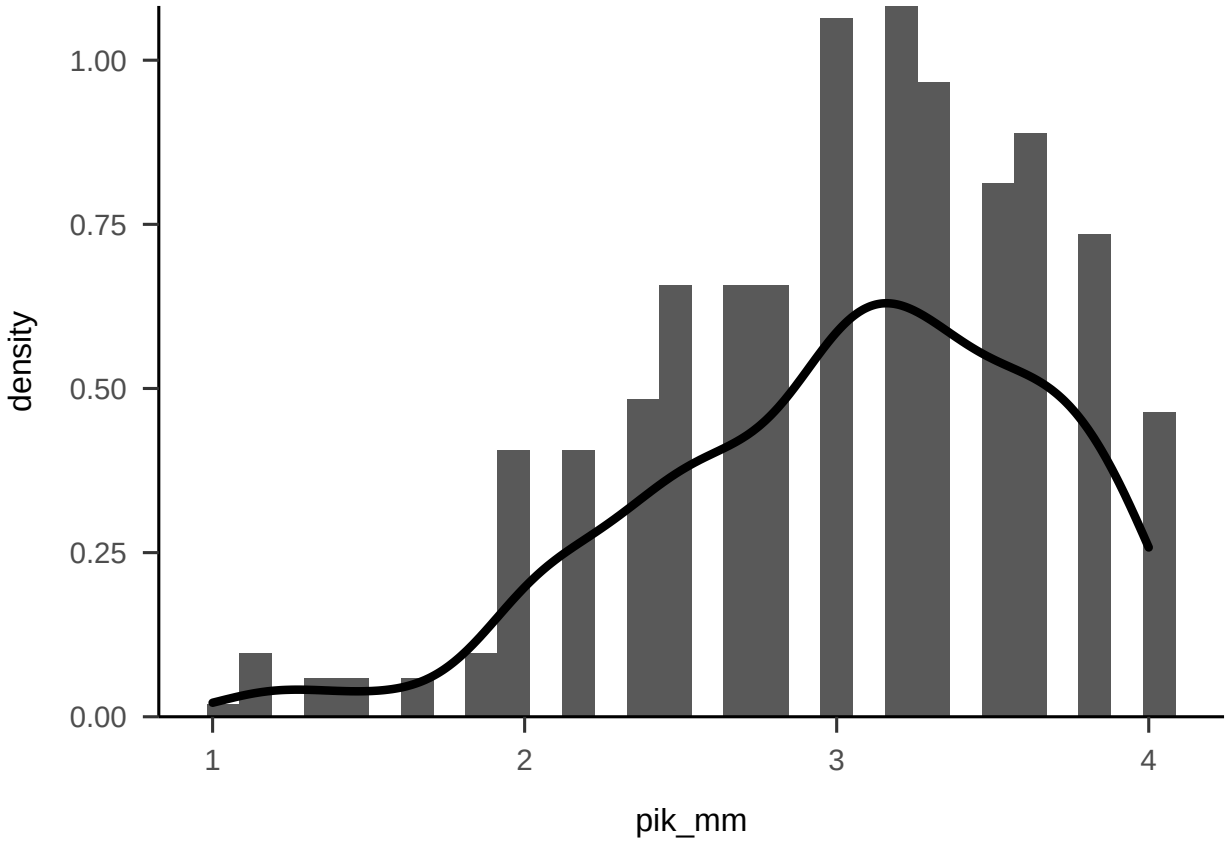

```
##
## [[12]]
```



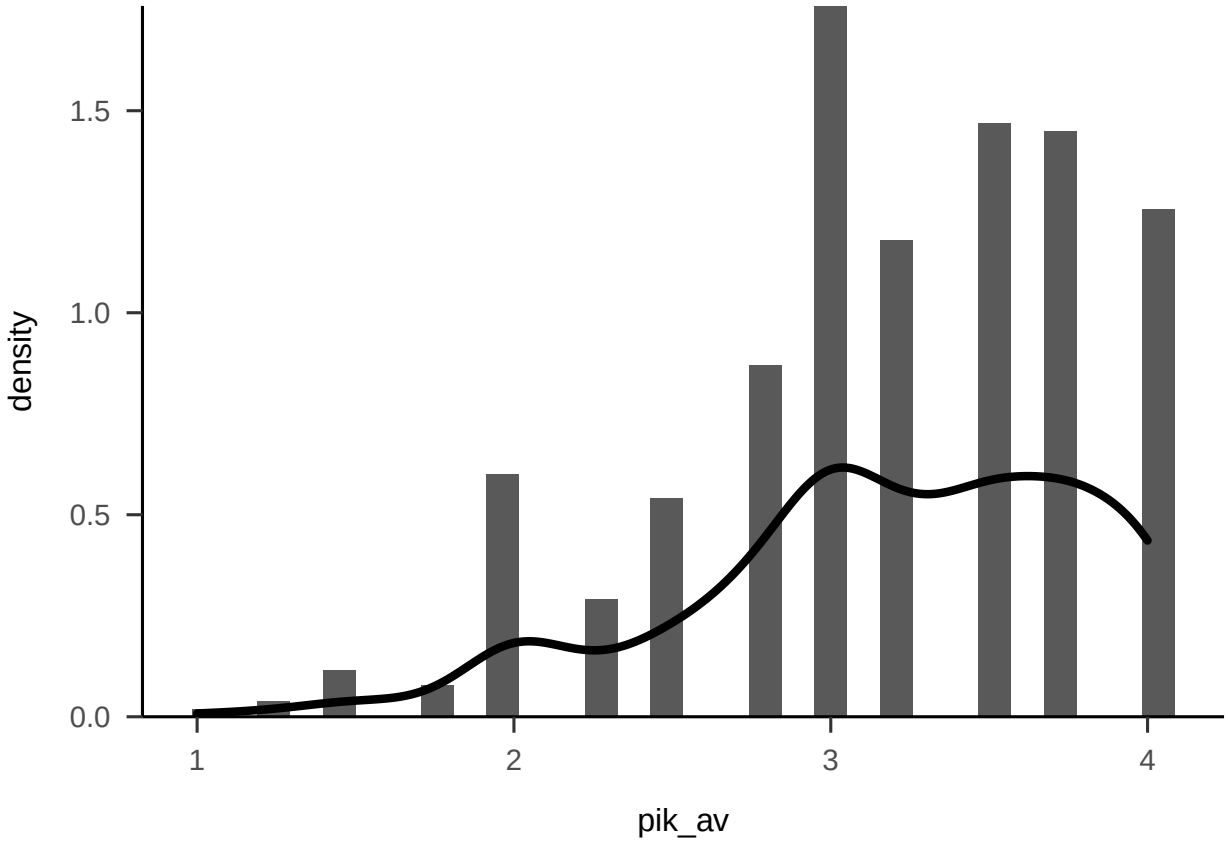

```
##
## [[13]]
```



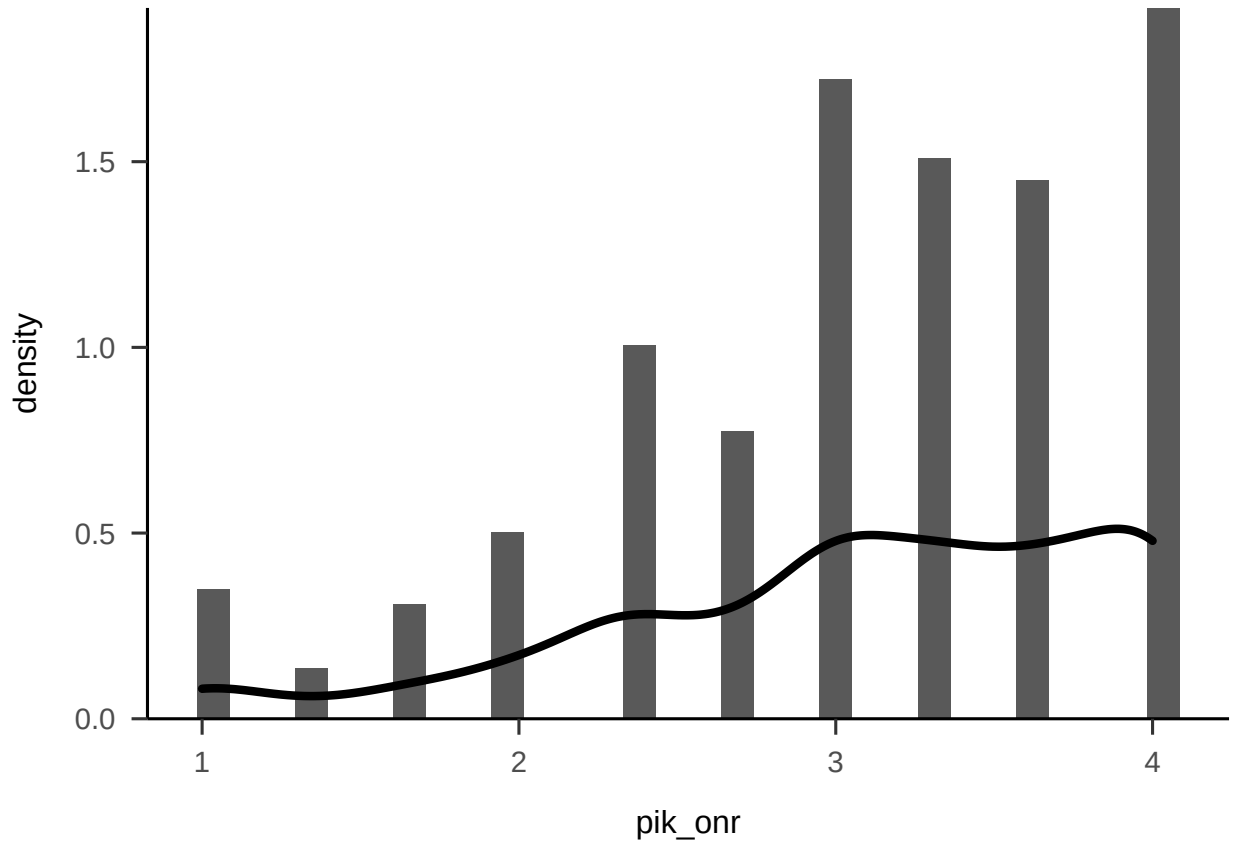

```
##
## [[14]]
```



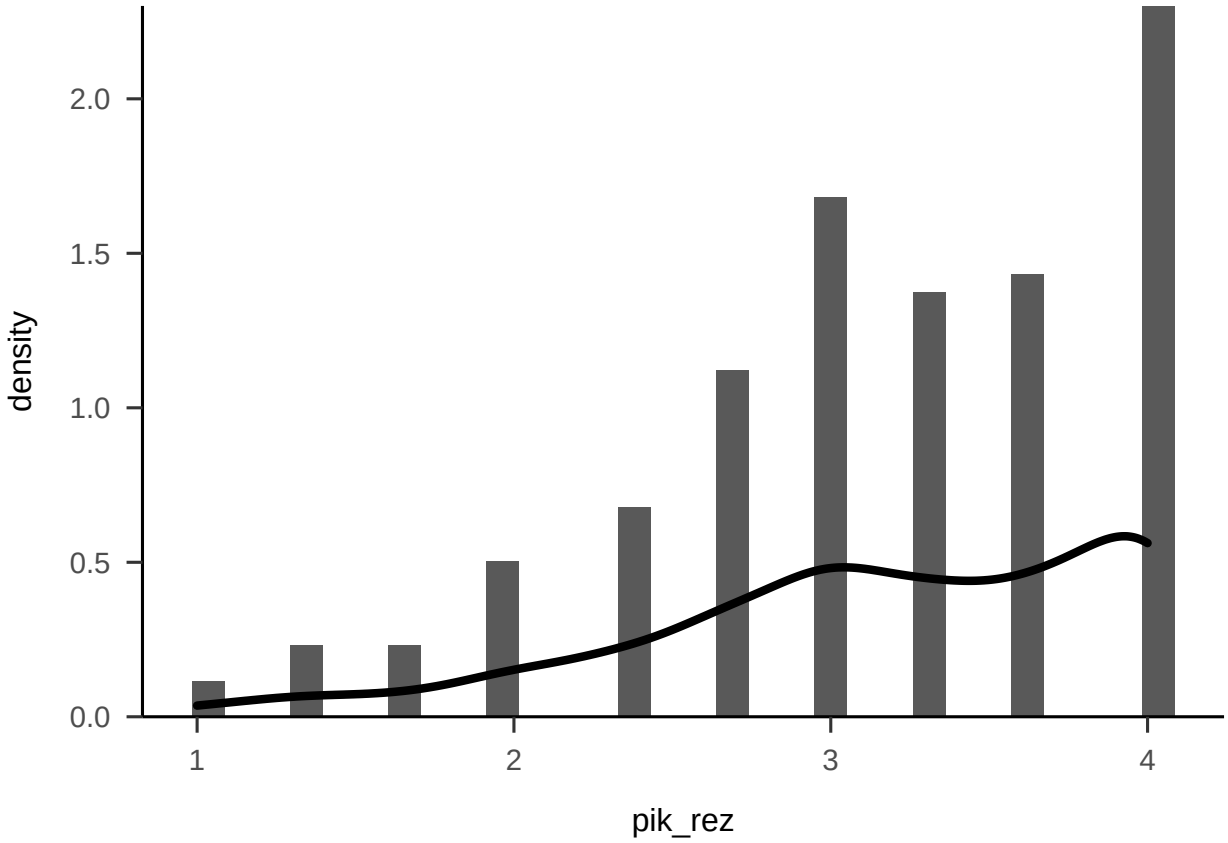

```
##
## [[15]]
```



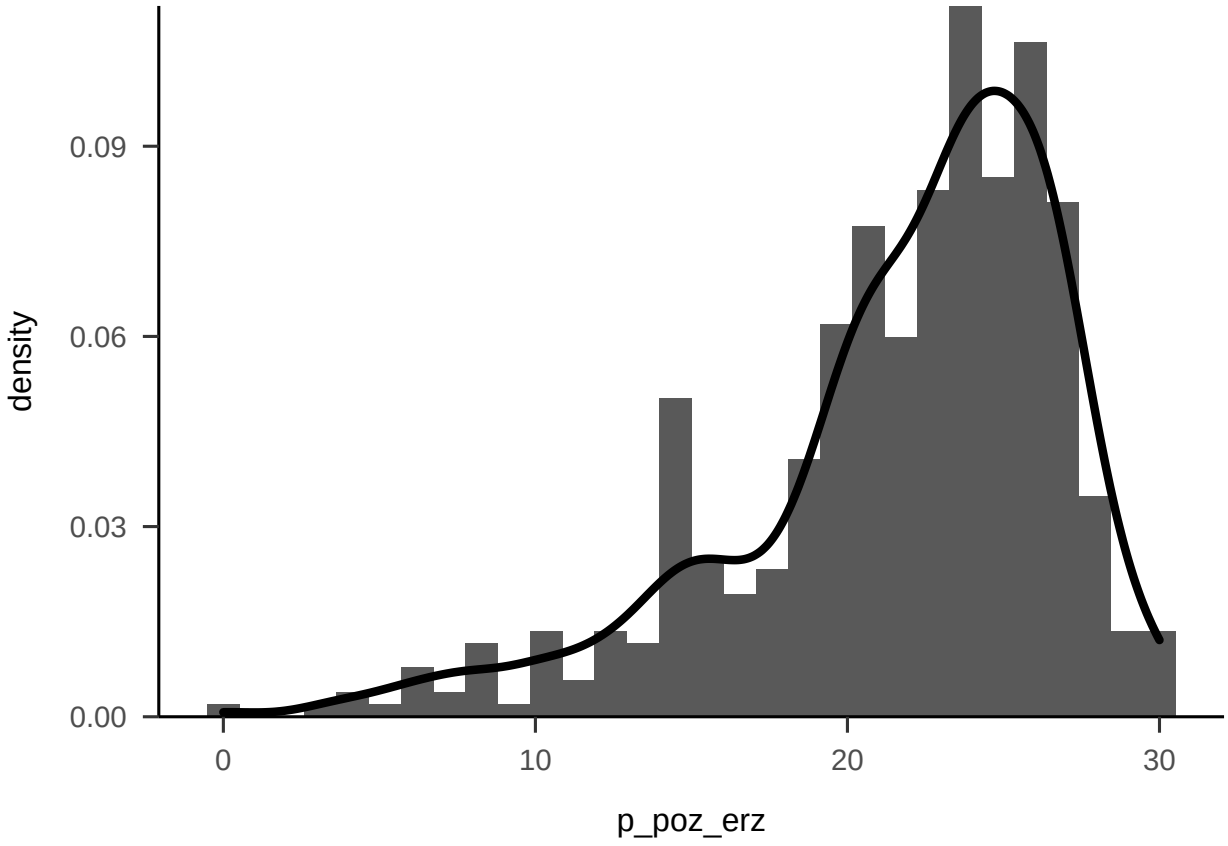

```
##
## [[16]]
```



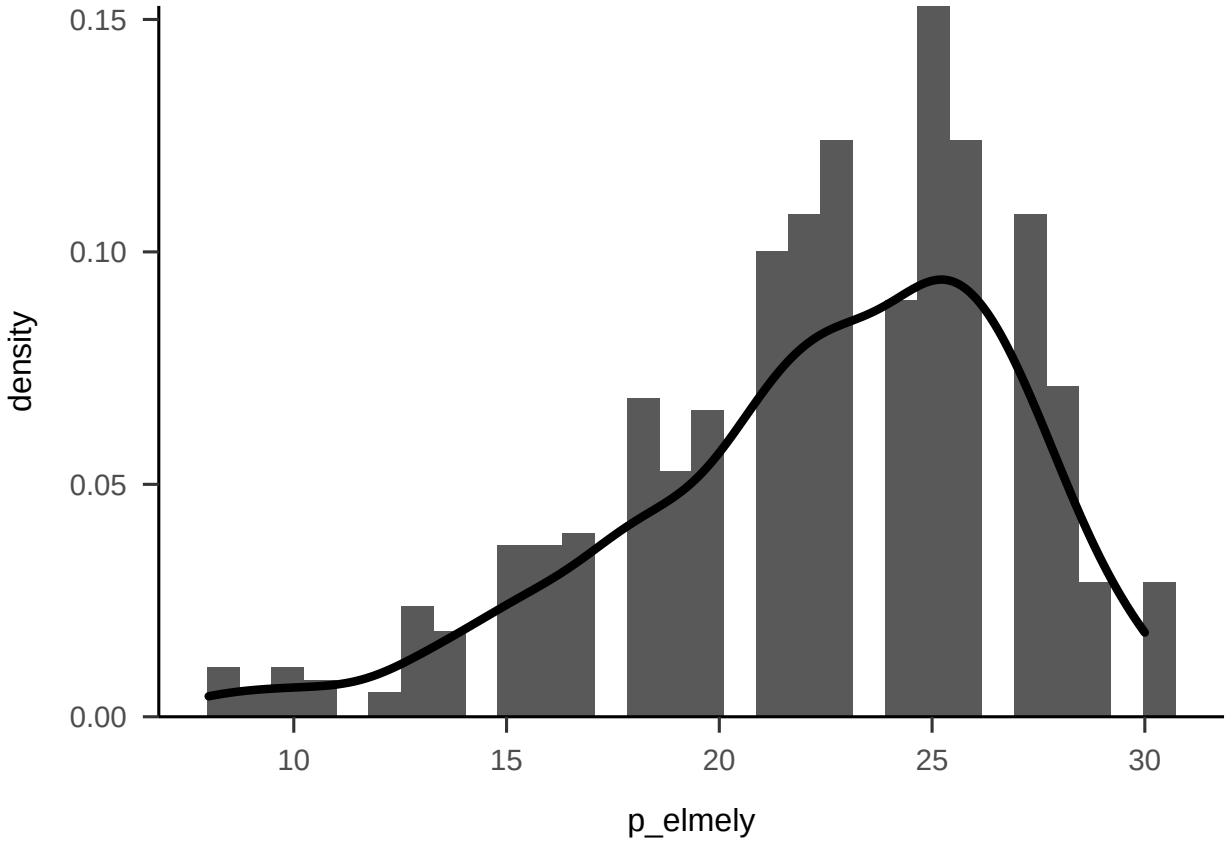

```
##
## [[17]]
```



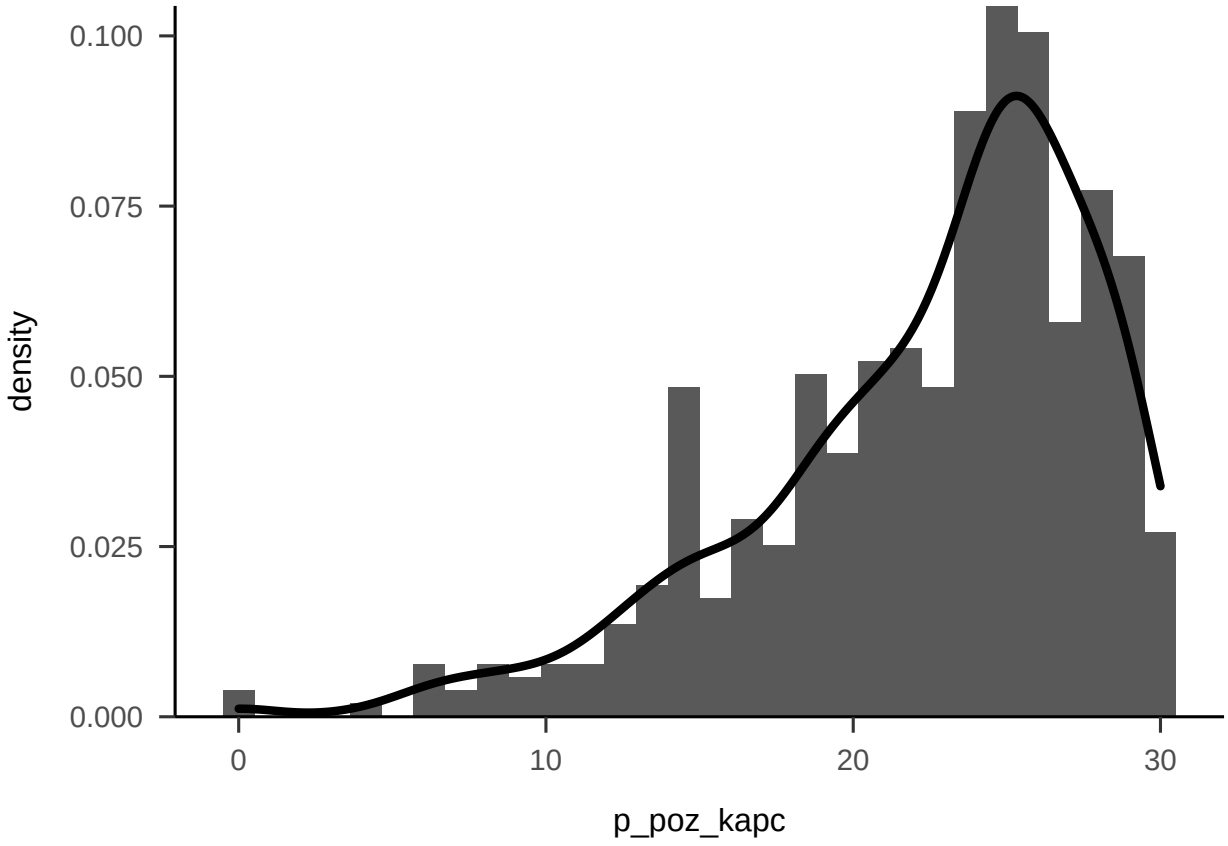

```
##
## [[18]]
```



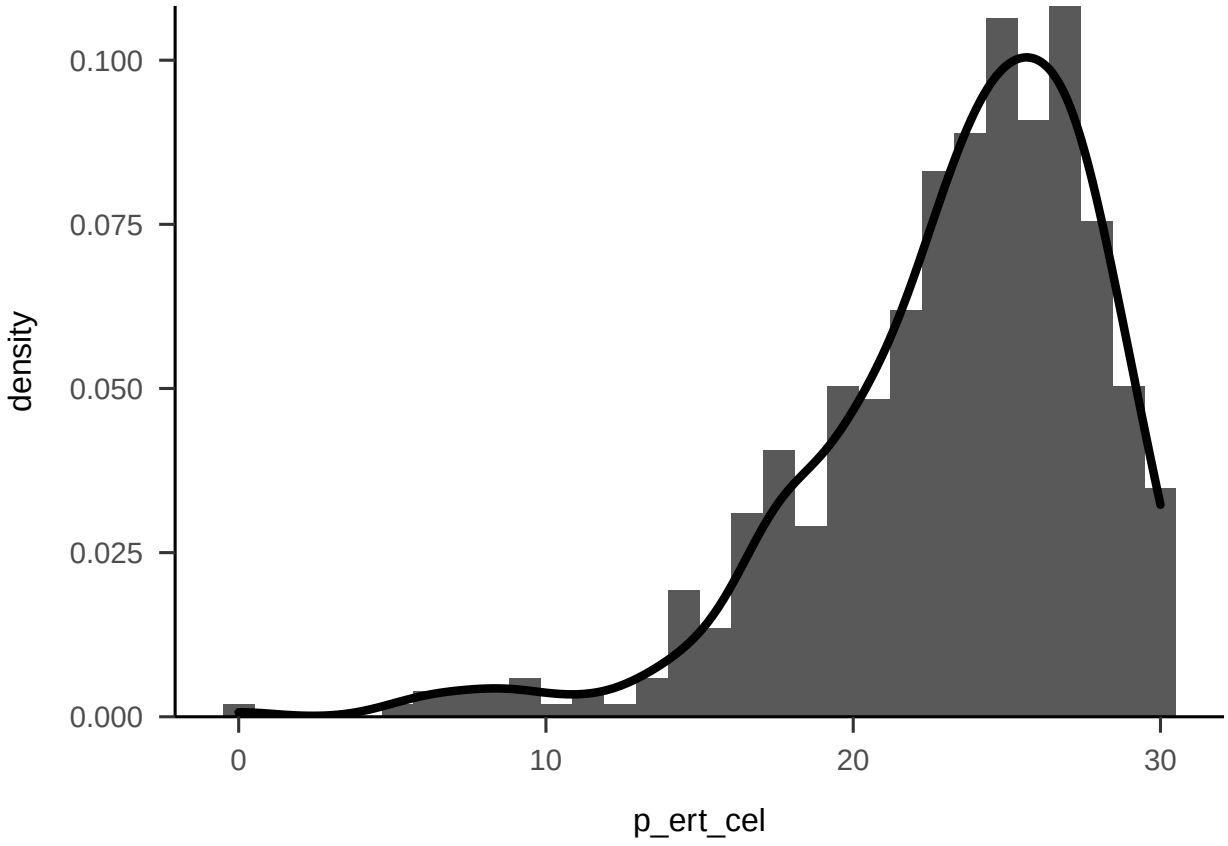

```
##
## [[19]]
```



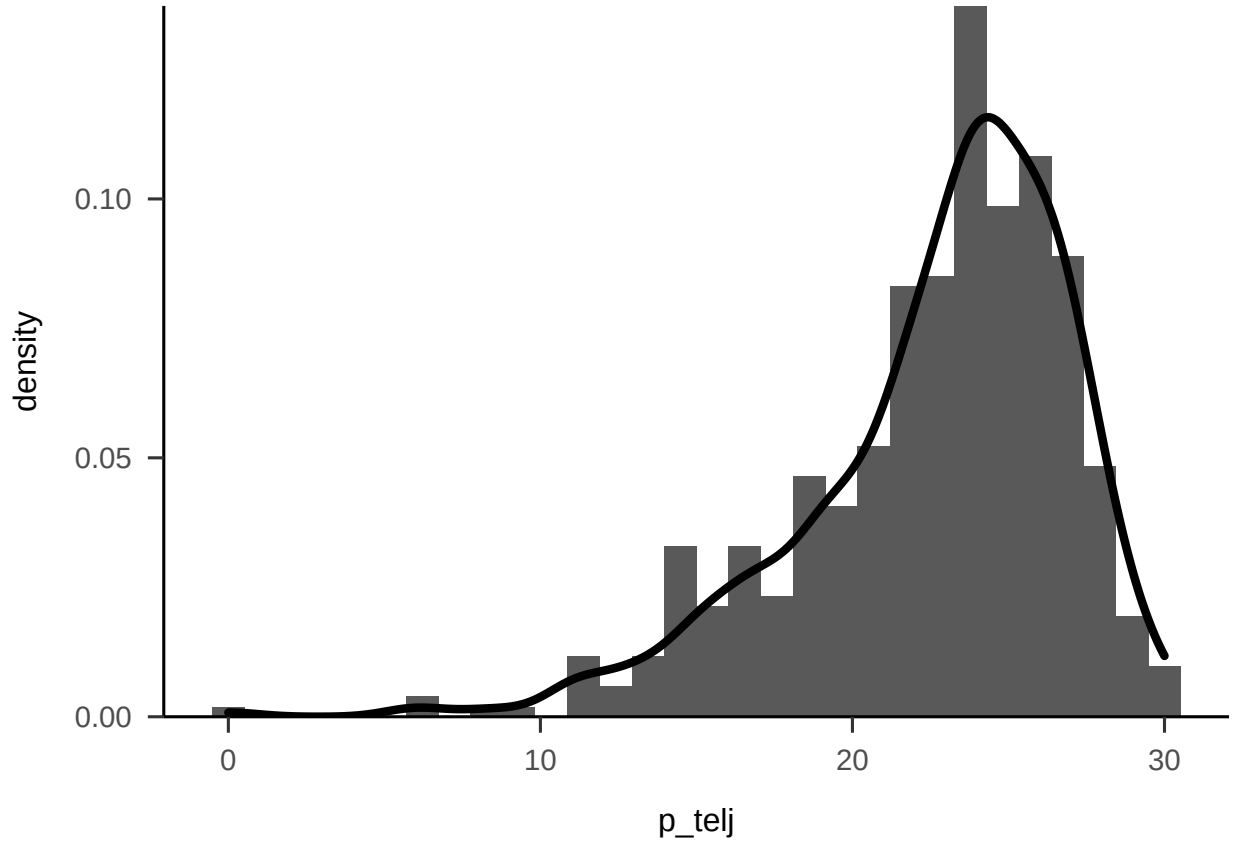

```
##
## [[20]]
```



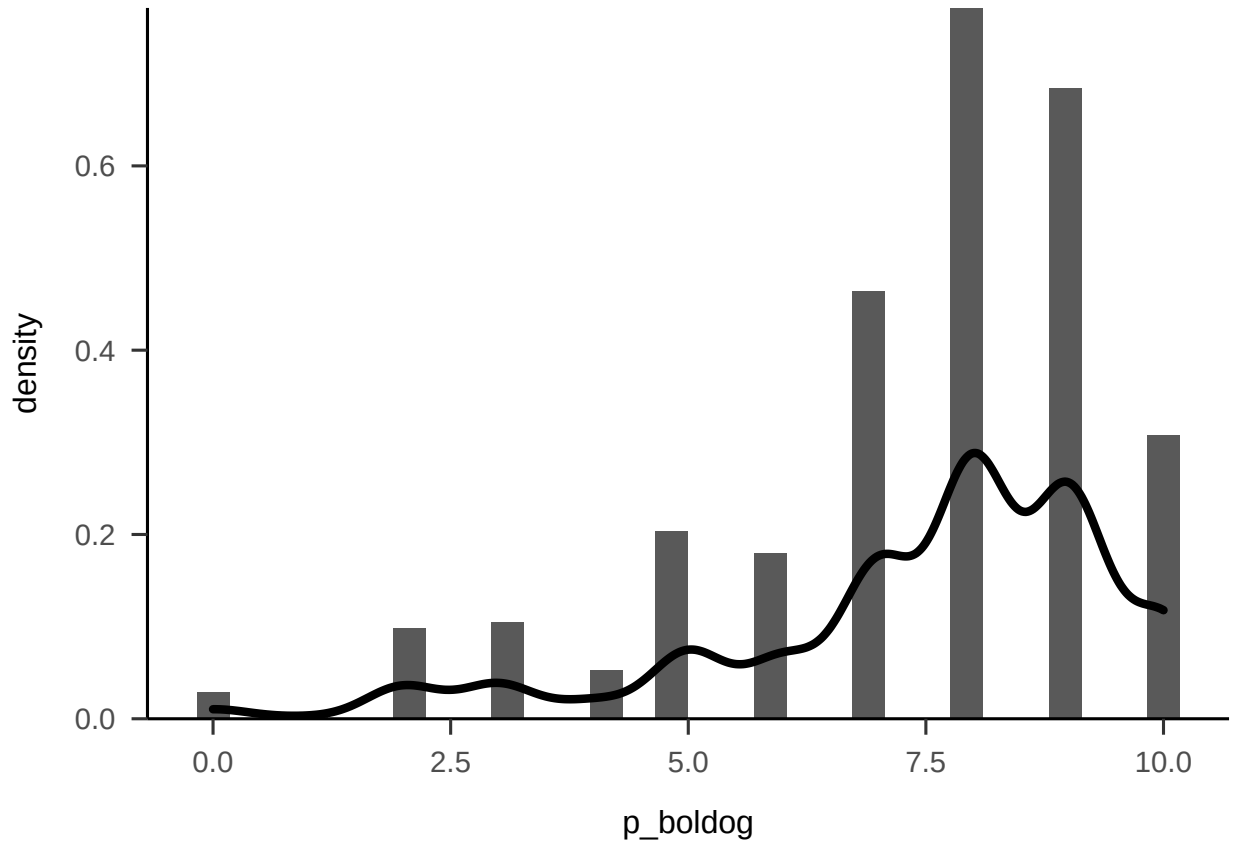

```
##
## [[21]]
```



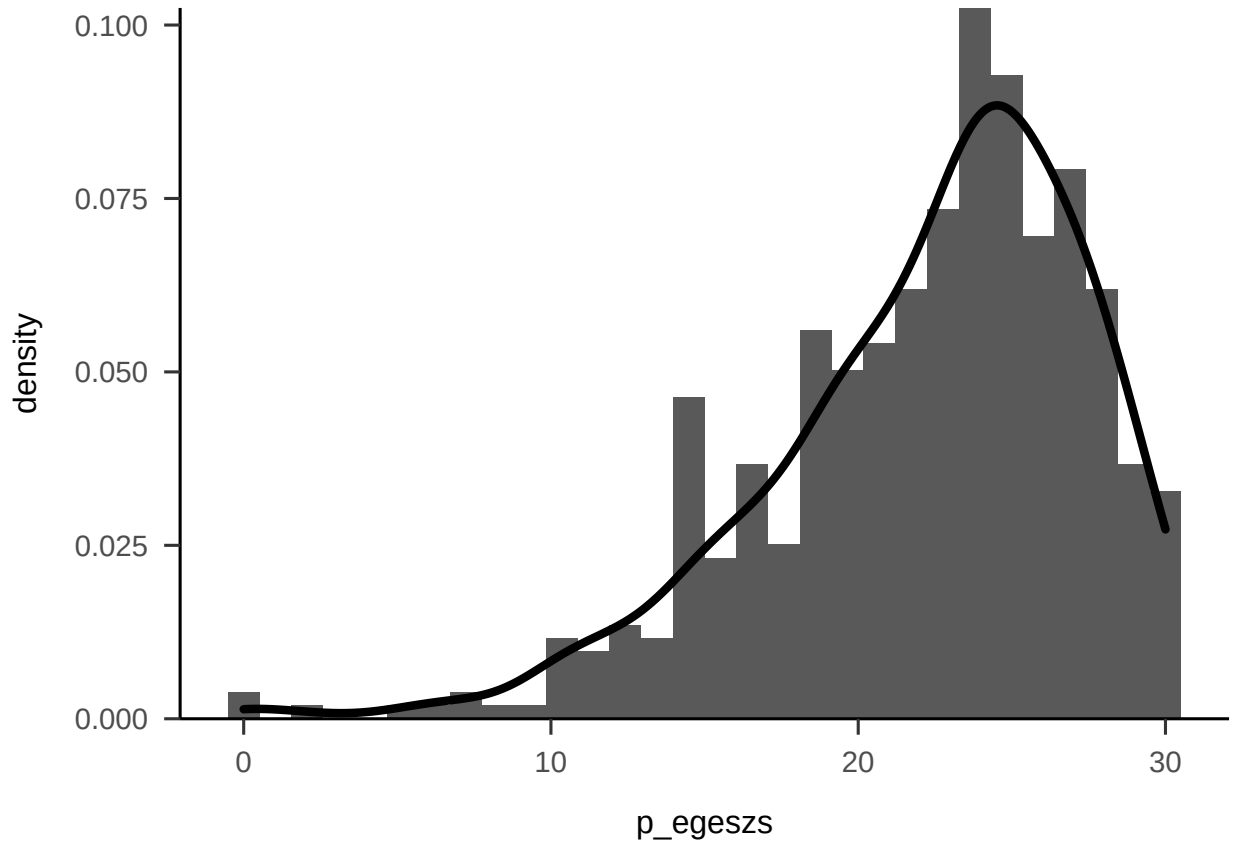

```
##
## [[22]]
```



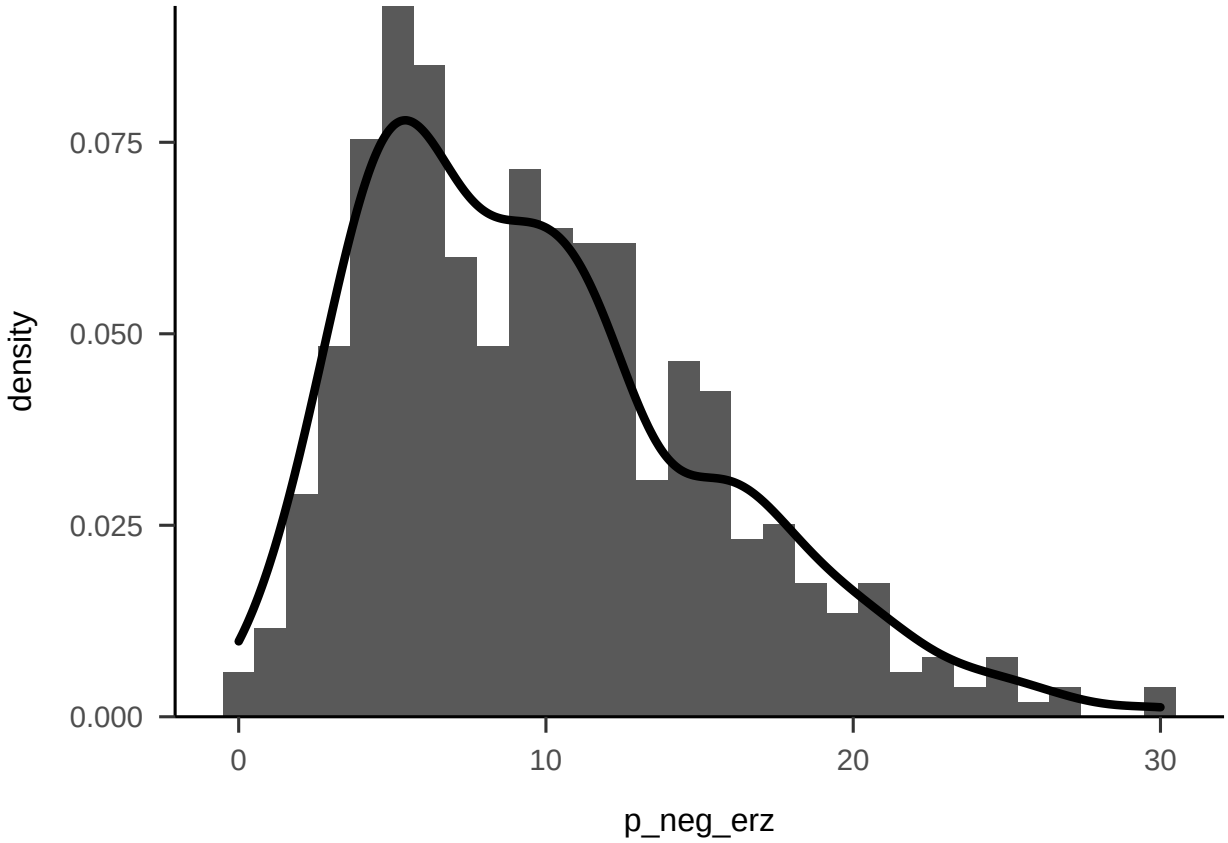

```
##
## [[23]]
```



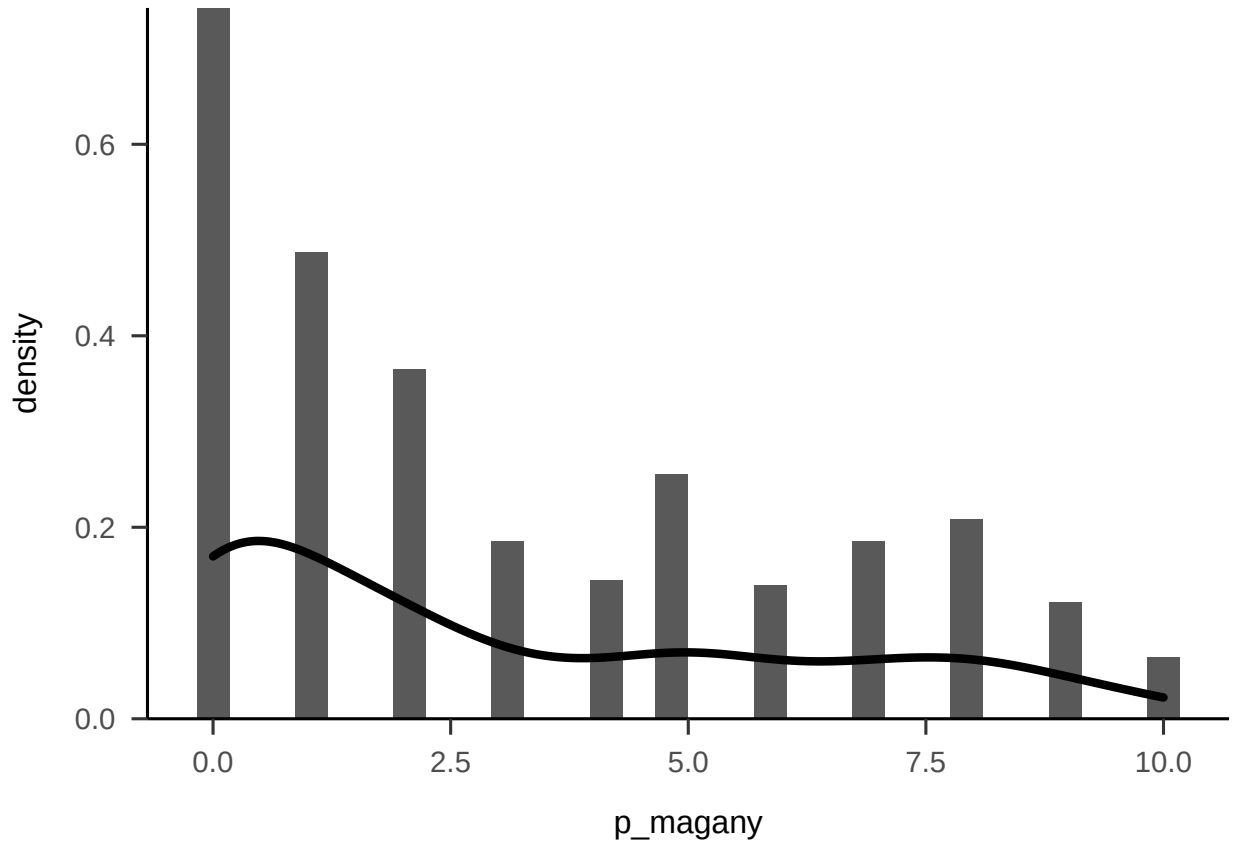

```
##
## [[24]]
```



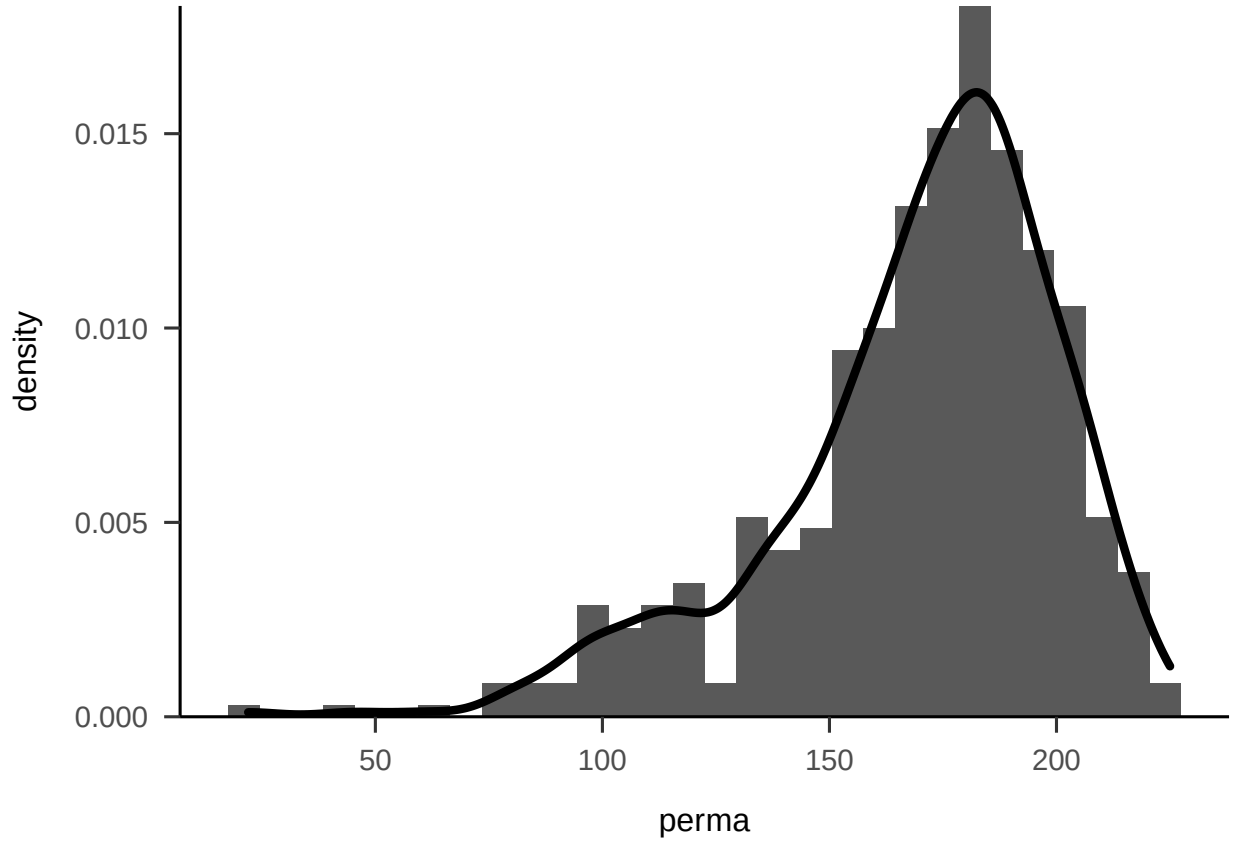

```
##
## [[25]]
```




```
##
## [[26]]
```




```
##
## [[27]]
```



Looking at the skewness and the kurtosis values. I will use 1.5 and -1.5 as arbitrary cutoff values for indicating a non normal distribution.

```
tibble(
  variable = interval_vars,
  skewness = map_dbl(variable, ~ skew(processed[[.x]])),
  kurtosis = map_dbl(variable, ~ kurtosi(processed[[.x]])),
  non_normal = case_when(skewness > 1.5 | skewness < -1.5 | kurtosis > 1.5 | kurtosis < -1.5 ~ "not nor
                         TRUE ~ "normal")
  ) %>%
  kable(
    format = "latex",
    # booktabs = TRUE,
    # escape = FALSE,
    col.names = c("Variable", "Skewness", "Kurtosis", "Normally distributed"),
    align = c("l", "c", "c", "c"),
    caption = "Investigation of the Skewness and Kurtosis Values of the interval Variables"
  ) %>%
  row_spec(row = 0, align = "c") %>%
  kable_styling(full_width = TRUE)
```

Based on visual investigation and the kurtosis and skewness values the *eletkora* variables seems like the most normally distributed variable. Whereas, the *p_ert_cel* and the *p_telj* variables are not normally distributed according to the cutoff values.

Table 1: Investigation of the Skewness and Kurtosis Values of the interval Variables

Variable	Skewness	Kurtosis	Normally distributed
eletkora	-0.1136826	-0.3196599	normal
jollet	-0.8250476	0.3261614	normal
savor	-0.7557138	0.4474827	normal
a_vhat	-0.8247275	0.5626726	normal
onreg	-0.5432614	-0.3902600	normal
rezil	-0.4662640	0.2435979	normal
m_flow	-0.8048933	0.2601981	normal
g_jerz	-0.3665192	-0.1867937	normal
g_jpszi	-0.5940671	0.1905535	normal
g_jszoc	-0.2737659	-0.5526795	normal
g_jspir	-0.4873621	-0.4809700	normal
g_jerzpsz	-0.5003305	0.0032951	normal
g_jspszoc	-0.3764639	-0.4724553	normal
pik_mm	-0.5946799	-0.0185863	normal
pik_av	-0.7076777	0.0514150	normal
pik_onr	-0.7801368	-0.0102156	normal
pik_rez	-0.7476450	-0.0444034	normal
p_poz_erz	-1.2110062	1.4076572	normal
p_elmely	-0.7712823	0.2846074	normal
p_poz_kapc	-1.0872805	1.0259594	normal
p_ert_cel	-1.3202785	2.5672759	not normal
p_telj	-1.2119993	2.1056079	not normal
p_boldog	-1.2583395	1.3092972	normal
p_egeszs	-1.0415275	1.3671302	normal
p_neg_erz	0.8239769	0.2866230	normal
p_magany	0.6599909	-0.8935592	normal
perma	-1.0938015	1.4907642	normal

4. Running Kolmogorov-Smirnov tests

I am running two-sided tests with a normal distribution as the cumulative distribution function.

```
tibble(
  variable = interval_vars,
  kolmogorov_res = map(interval_vars, ~ ks.test(processed[[.x]], y = "pnorm")),
  test_statistic = map_chr(kolmogorov_res, "statistic"),
  p = map_dbl(kolmogorov_res, "p.value")
  ) %>%
  mutate(p = if_else(p == 0, "p < 0.01", as.character(p))) %>%
  select(variable, test_statistic, p) %>%
    kable(
    format = "latex",
    # booktabs = TRUE,
    # escape = FALSE,
    col.names = c("Variable", "Test statistic", "p"),
    align = c("l", "c", "c"),
    caption = "Results of the Kolmogorov-Smirnov Tests"
  ) %>%
  row_spec(row = 0, align = "c") %>%
  kable_styling(full_width = TRUE)
```

All tests are significant.

5. On which scales do we find the largest difference between male and female? What are the corresponding standardised effectsizes (in Cohens' d and eta square)?

For all of the variables I used the mean of the variable by gender group for the comparison. Also, I am running type 2 ANOVAs to calculate the eta square from.

```
# creating a function for calculating the mean difference
gender_mean_diff <- function(var) {
  processed %>%
    group_by(neme) %>%
    summarise(mean = mean(.data[[var]], na.rm = TRUE)) %>%
    spread(neme, mean) %>%
    mutate(diff = no - ferfi) %>%
    pull(diff)
}

# creating function to calculate pooled SD
gender_sd_pooled <- function(var) {
  processed %>%
    group_by(neme) %>%
    summarise(sd = sd(.data[[var]], na.rm = TRUE)) %>%
    spread(neme, sd) %>%
    mutate(sd_pooled = sqrt((ferfi^2 + no^2)/2)) %>%
    pull(sd_pooled)
}
```

Table 2: Results of the Kolmogorov-Smirnov Tests

Variable	Test statistic	p
eletkora	1.000000	$p < 0.01$
jollet	0.961250	$p < 0.01$
savor	0.967250	$p < 0.01$
a_vhat	0.970097	$p < 0.01$
onreg	0.933250	$p < 0.01$
rezil	0.947250	$p < 0.01$
m_flow	0.984184	$p < 0.01$
g_jerz	0.959250	$p < 0.01$
g_jpszi	0.963250	$p < 0.01$
g_jszoc	0.927250	$p < 0.01$
g_jspir	0.941941	$p < 0.01$
g_jerzpsz	0.963185	$p < 0.01$
g_jspszoc	0.937941	$p < 0.01$
pik_mm	0.937250	$p < 0.01$
pik_av	0.951250	$p < 0.01$
pik_onr	0.902213	$p < 0.01$
pik_rez	0.917250	$p < 0.01$
p_poz_erz	0.996650	$p < 0.01$
p_elmely	1.000000	$p < 0.01$
p_poz_kapc	0.995968	$p < 0.01$
p_ert_cel	0.998000	$p < 0.01$
p_telj	0.998000	$p < 0.01$
p_boldog	0.965250	$p < 0.01$
p_egeszs	0.994000	$p < 0.01$
p_neg_erz	0.959250	$p < 0.01$
p_magany	0.585345	$p < 0.01$
perma	1.000000	$p < 0.01$

```
tibble(
  variable = interval_vars,
  mean_diff = map_dbl(variable, ~ gender_mean_diff(.x)),
  sd_pooled = map_dbl(variable, ~ gender_sd_pooled(.x)),
  cohens_d = mean_diff / sd_pooled,
  anova_res = map(variable,
                  ~ aov(processed[[.x]] ~ processed[["neme"]], data = processed)),
  eta_squared = map_dbl(anova_res, ~ lsr::etaSquared(.x, type = 2, anova = FALSE)[[1]])
  ) %>%
  select(variable, mean_diff, cohens_d, eta_squared) %>%
    kable(
    format = "latex",
    # booktabs = TRUE,
    # escape = FALSE,
    col.names = c("Variable", "Mean difference", "Cohens' d", "Eta-squared"),
    align = c("l", "c", "c", "c"),
    caption = "The Corresponding Raw and Standardised Effect Sizes For Each intervalum Variable"
  ) %>%
  row_spec(row = 0, align = "c") %>%
  kable_styling(full_width = TRUE)
```

References

Vargha, András, Virág Zábó, Regina Török, and Attila Oláh. 2020. “A jóllét és a Mentális Egészség mérése: A Mentális Egészség Teszt.” *Mentálhigiéné és Pszichoszomatika* 21 (3): 281–322.

Table 3: The Corresponding Raw and Standardised Effect Sizes For Each intervalum Variable

Variable	Mean difference	Cohens' d	Eta-squared
eletkora	0.6600000	0.0562524	0.0007936
jollet	0.1213304	0.1144186	0.0032753
savor	0.3306676	0.3304086	0.0266712
a_vhat	0.1800000	0.1923067	0.0091972
onreg	0.1146696	0.0938674	0.0022067
rezil	0.0253368	0.0247916	0.0001542
m_flow	0.1720028	0.1937621	0.0093357
g_jerz	0.1456000	0.1413822	0.0049923
g_jpszi	0.0980000	0.0990841	0.0024582
g_jszoc	0.1490000	0.1265134	0.0040014
g_jspir	0.2970000	0.2491063	0.0153369
g_jerzpsz	0.1218000	0.1254269	0.0039332
g_jspszoc	0.2230000	0.1966731	0.0096156
pik_mm	0.0679968	0.1085427	0.0029485
pik_av	0.1020000	0.1640537	0.0067101
pik_onr	0.1119952	0.1417594	0.0050188
pik_rez	0.0506632	0.0682053	0.0011663
p_poz_erz	0.9640000	0.1833292	0.0083656
p_elmely	0.6320000	0.1388376	0.0048150
p_poz_kapc	0.8120000	0.1461005	0.0053292
p_ert_cel	0.8280000	0.1774774	0.0078442
p_telj	0.3200000	0.0736748	0.0013606
p_boldog	0.2400000	0.1131828	0.0032051
p_egeszs	1.0960000	0.2107213	0.0110226
p_neg_erz	-0.1880000	-0.0329020	0.0002716
p_magany	0.0280000	0.0091516	0.0000210
perma	5.0520000	0.1619439	0.0065397