

Procvičování 7 - řešení

Načtete si pracovní prostředí z <http://www.sci.muni.cz/~syrovat/svratka.RData>
a z <http://www.sci.muni.cz/~syrovat/becva.RData>.

```
load(url("http://www.sci.muni.cz/~syrovat/svratka.RData"))
load(url("http://www.sci.muni.cz/~syrovat/becva.RData"))
```

1. Vytvořte sekvenci celých čísel od 1 do 10. Zjistěte délku vytvořeného vektoru, jeho minimální a maximální hodnotu a součet všech hodnot vektoru.

```
vec <- 1:10
length(vec)

## [1] 10

min(vec)

## [1] 1

max(vec)

## [1] 10

sum(vec)

## [1] 55
```

2. Vytvořte vektor obsahující hodnoty 1 až 10, 21 až 30, 41 až 50 v uvedeném pořadí.

```
c(1:10, 21:30, 41:50)

## [1] 1 2 3 4 5 6 7 8 9 10 21 22 23 24 25 26 27 28 29 30 41 42 43
## [24] 44 45 46 47 48 49 50
```

3. Vytvořte sekvenci čísel od 0 do 0.3, jejíž délka bude rovna 100.

```
seq(0, 0.3, length = 100)

## [1] 0.000000 0.003030 0.006061 0.009091 0.012121 0.015152 0.018182
## [8] 0.021212 0.024242 0.027273 0.030303 0.033333 0.036364 0.039394
## [15] 0.042424 0.045455 0.048485 0.051515 0.054545 0.057576 0.060606
## [22] 0.063636 0.066667 0.069697 0.072727 0.075758 0.078788 0.081818
## [29] 0.084848 0.087879 0.090909 0.093939 0.096970 0.100000 0.103030
## [36] 0.106061 0.109091 0.112121 0.115152 0.118182 0.121212 0.124242
## [43] 0.127273 0.130303 0.133333 0.136364 0.139394 0.142424 0.145455
## [50] 0.148485 0.151515 0.154545 0.157576 0.160606 0.163636 0.166667
## [57] 0.169697 0.172727 0.175758 0.178788 0.181818 0.184848 0.187879
## [64] 0.190909 0.193939 0.196970 0.200000 0.203030 0.206061 0.209091
```

```
## [71] 0.212121 0.215152 0.218182 0.221212 0.224242 0.227273 0.230303
## [78] 0.233333 0.236364 0.239394 0.242424 0.245455 0.248485 0.251515
## [85] 0.254545 0.257576 0.260606 0.263636 0.266667 0.269697 0.272727
## [92] 0.275758 0.278788 0.281818 0.284848 0.287879 0.290909 0.293939
## [99] 0.296970 0.300000
```

4. Vytvořte vektor obsahující všechna lichá čísla od -9 do 9.

```
seq(-9, 9, 2)
## [1] -9 -7 -5 -3 -1 1 3 5 7 9
```

5. Vytvořte vektor obsahující násobky 7 od 7 do 70.

```
seq(7, 70, 7)
## [1] 7 14 21 28 35 42 49 56 63 70
```

6. Vytvořte vektor obsahující hodnoty 1 až 5, které se budou 10 krát opakovat (1,2,3,4,5,1,2,3,4,5,1,2,...).
Odpovídá délka vektoru násobku 10 a 5, jak by měla? Přímou se zeptejte, odpovědí je buď TRUE, nebo FALSE.

```
vec <- rep(1:5, 10)
length(vec) == 50
## [1] TRUE
```

7. Vytvořte vektor obsahující textové řetězce "pejsek", "kocicka", "dort", každý z nich se opakující 5 krát (nejprve "pejsek" 5 krát, pak "kocicka" 5 krát a nakonec "dort" 5 krát).

```
rep(c("pejsek", "kocicka", "dort"), each = 5)
## [1] "pejsek" "pejsek" "pejsek" "pejsek" "pejsek" "kocicka" "kocicka"
## [8] "kocicka" "kocicka" "kocicka" "dort" "dort" "dort" "dort"
## [15] "dort"
```

8. Vytvořte vektor obsahující hodnoty 1 až 5, každá z nich se bude dvakrát za sebou opakovat, a to celé se bude opakovat třikrát: (tohle je výsledek: 1 1 2 2 3 3 4 4 5 5 1 1 2 2 3 3 4 4 5 5 1 1 2 2 3 3 4 4 5 5 3 3 4 4 5 5)

```
rep(1:5, each = 2, 3)
## [1] 1 1 2 2 3 3 4 4 5 5 1 1 2 2 3 3 4 4 5 5 1 1 2 2 3 3 4 4 5 5 3 3 4 4 5 5
```

9. Vytvořte sekvenci celých čísel od 1 do 5, v níž jednička bude jednou, dvojka dvakrát, ... pětka pětkrát (1, 2, 2, 3, 3, 3, ...).

```
rep(1:5, 1:5)
## [1] 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5
```

10. Vytvořte vektor, v němž se budou opakovat jména druhů dataframu *spe* každé tolikrát, kolik bylo jedinců daného druhu na 1. lokalitě dataframu *spe* (*s1*).

```
vec <- rep(names(spe), spe[1, ])
```

11. Aplikujte na výsledek předchozího bodu funkci `table()`. Co je výsledkem?

```
table(vec)
## vec
## Apsetrif Cricbici Eukicoer Eukilobi Nanoreag Orthobum Orthrubi Pararufi
##          1          1          1          3          7          28          30          2
## Rheofusc Synosemi  corysp. cricbigr crictrgr micrchgr mictrasp rheotasp
##          3          120         15          2          2          25          3          2
## tanybrun tanytasp thellasp thieGrGe tvetbaca tvetdive
##          7          4          5          5          4          4
```

12. Spojte do jednoho dataframu *both* měření hloubky, rychlosti proudu a Froudeho čísla dataframů *env* a *becva*. Přidejte do tohoto dataframu i abundance druhu *Tvetenia calvescens* (*tvetbaca*). Samozřejmě tak, aby abundance pocházely ze stejných vzorků jako měřené proměnné. Vlastně chceme spojit stejné proměnné naměřené na dvou řekách do jednoho dataframu.

```
names(env)
## [1] "gr_env" "depth" "velocity" "froude"
names(becva)
## [1] "vel" "froude" "depth" "tvetbaca"
names(env)[3] <- "vel"
both <- rbind(env[, 2:4], becva[, 1:3])
both$tvetbaca <- c(spe$tvetbaca, becva$tvetbaca)
```

13. V dataframu *both* vytvořte novou proměnnou *loc*, odlišující měření na řece Svatce od řeky Bečvy. Sami uvažte, jakého má být typu (třídy), a jakých by měla nabývat hodnot. V dalších krocích ji použijeme v grafu a pro analýzu.

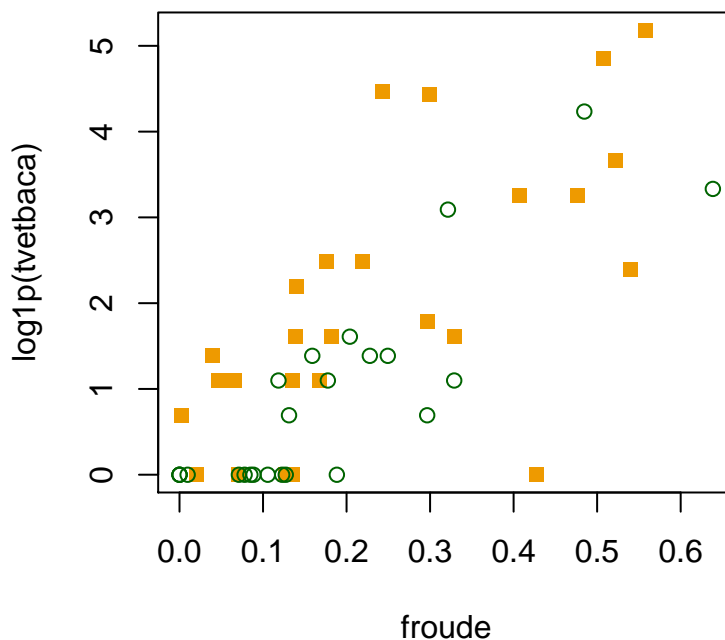
```
both$loc <- factor(rep(c("svratka", "becva"), c(nrow(env), nrow(becva))))
both
```

##	depth	vel	froude	tvetbaca	loc
## s1	0.395	0.274	0.139193	4	svratka
## s2	0.422	0.358	0.175951	11	svratka
## s3	0.496	0.310	0.140536	8	svratka
## s4	0.291	0.078	0.046165	2	svratka
## s5	0.320	0.092	0.051925	2	svratka
## s6	0.328	0.126	0.070242	0	svratka
## s7	0.278	0.490	0.296715	5	svratka
## s8	0.213	0.618	0.427528	0	svratka
## s9	0.243	0.508	0.329023	4	svratka
## s10	0.353	0.758	0.407331	25	svratka
## s11	0.236	0.334	0.219511	11	svratka
## s12	0.245	0.810	0.522477	38	svratka
## s13	0.184	0.088	0.065500	2	svratka
## s14	0.155	0.224	0.181655	4	svratka
## s15	0.279	0.496	0.299809	83	svratka
## s16	0.501	0.372	0.167799	2	svratka
## s17	0.454	0.286	0.135520	0	svratka
## s18	0.200	0.340	0.242733	86	svratka
## s19	0.162	0.050	0.039662	3	svratka
## s20	0.290	0.804	0.476675	25	svratka
## s21	0.138	0.590	0.507082	127	svratka
## s22	0.340	0.246	0.134698	2	svratka
## s23	0.344	0.234	0.127380	0	svratka
## s24	0.300	0.958	0.558432	177	svratka
## s25	0.139	0.630	0.539509	10	svratka
## s26	0.124	0.002	0.001813	1	svratka
## s27	0.478	0.044	0.020319	0	svratka
## A01	0.360	0.354	0.188373	0	becva
## A02	0.180	0.236	0.177599	2	becva
## A03	0.050	0.074	0.105660	0	becva
## A04	0.810	0.448	0.158928	3	becva
## A05	0.430	0.609	0.296516	1	becva
## A06	0.160	0.800	0.638551	27	becva
## A07	0.340	0.155	0.084871	0	becva
## A08	0.400	0.252	0.127214	0	becva
## A09	0.130	0.011	0.009741	0	becva
## A10	0.640	0.222	0.088599	0	becva
## A11	0.990	0.383	0.122899	0	becva
## A12	0.410	0.457	0.227871	3	becva
## A13	0.170	0.415	0.321358	21	becva
## A15	0.190	0.179	0.131112	1	becva
## A16	0.070	0.169	0.203940	4	becva
## A17	0.100	0.480	0.484626	68	becva
## A18	0.220	0.174	0.118441	2	becva
## A19	0.320	0.126	0.071115	0	becva
## A20	0.210	0.472	0.328850	2	becva
## SH1	0.150	0.000	0.000000	0	becva
## SH2	0.280	0.000	0.000000	0	becva
## SH5	0.090	0.000	0.000000	0	becva
## SH7	0.740	0.210	0.077942	0	becva

```
## SH8 0.600 0.605 0.249371      3      becva
```

14. Zobrazte bodový graf závislosti log abundance *T. calvescens* na Froudeho čísle a odlište symboly vzorky ze Svratky od Bečvy.

```
par(mar = c(4, 4, 1, 1))
plot(log1p(tvetbaca) ~ froude, data = both, pch = c(1, 15)[loc], col = c("darkgreen",
"orange2")[loc])
```



15. Vytvořte lineární model závislosti log abundance *T. calvescens* na Froudeho čísle a lokalitě (m.tvet<- lm(log1p(tvetbaca) froude + loc, data= both)).

```
m.tvet <- lm(log1p(tvetbaca) ~ froude + loc, data = both)
```

16. Vytvořte dataframe s proměnnými *loc* a *froude* pro predikování modelovaných abundancí. V proměnné *loc* se bude 100 krát opakovat jméno první řeky, pak 100 krát jméno druhé řeky. Proměnná *froude* bude obsahovat dvakrát za sebou sekvenci hodnot Froudeho čísla od jeho minimální hodnoty po maximální o délce 100 (celkem tedy bude mít délku = 2 x 100).

```
pred.data <- data.frame(loc = rep(levels(both$loc), each = 100), froude = rep(seq(min(both$froude)
max(both$froude), length = 100), 2))
```

Asi už tušíte, proč to děláme. Abychom do grafu mohli zobrazit modelovaný vztah závislosti (log) abundance *T. calvescens* na hydraulických podmínkách potažmo Froudeho čísla jako hladkou čáru, musíme napredikovat její (log) abundance pro dostatečný počet bodů rovnoměrně rozptýlených na

gradientu Froudeho čísla. A protože máme dvě lokality, na nichž vztah abundance *T. calvescens* a Froudeho čísla může být jiný, musíme to udělat pro obě lokality zvlášť.

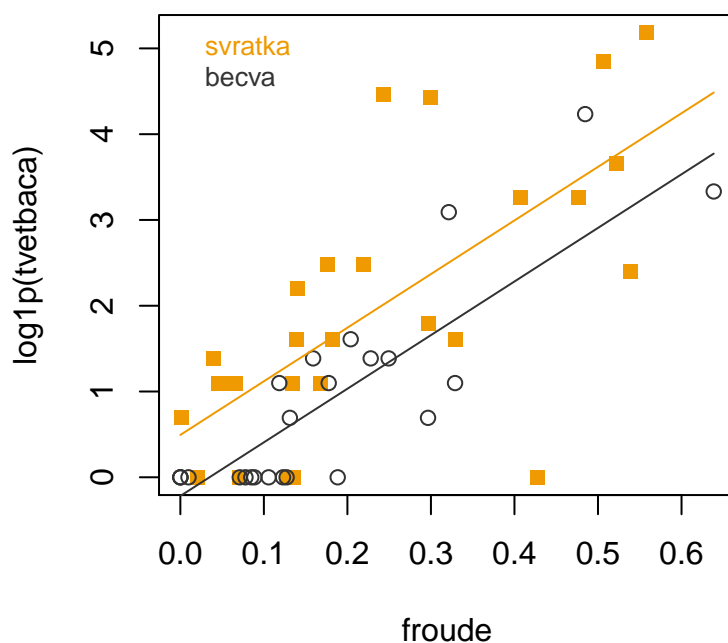
Dataframe *pred.data* obsahuje sekvenci Froudeho čísel od nejmenšího naměřeného po největší naměřené dvakrát za sebou - pro řeku Svatku a Bečvu. Teď můžeme **R** požádat o predikování (log) abundancí *T. calvescens* podle modelu *m.tvet* pro všechny kombinace Froudeho čísel a řek dataframu *pred.data*, tedy pro celou sekvenci a řeku Svatku a pro celou sekvenci a řeku Bečvu.

```
pred.data$tvetbaca <- predict(m.tvet, newdata = pred.data)
```

Argumentem *newdata=* **R** říkáme, že hodnoty prediktorů, podle nichž bude predikovat, má brát z dataframu *pred.data*, nikoliv z *both* (z něhož bral proměnné při stavbě modelu).

Nyní máme napredikované hodnoty a můžeme malovat. Nejprve zobrazíme bodový graf s vlastními naměřenými hodnotami, opět odlišíme obě řeky. A pak do něj pomocí funkce *lines()* vložíme modelované závislosti. Nejprve pro řeku Svatku a pak pro Bečvu:

```
par(mar = c(4, 4, 1, 1))
plot(log1p(tvetbaca) ~ froude, data = both, pch = c(1, 15)[loc], col = c("orange2",
  "orange2")[loc])
lines(tvetbaca ~ froude, data = pred.data, col = "orange2", subset = loc ==
  "svratka")
lines(tvetbaca ~ froude, data = pred.data, col = "grey20", subset = loc == "becva")
legend("topleft", text.col = c("orange2", "grey20"), legend = c("svratka", "becva"),
  bty = "n", cex = 0.8)
```



Posledním příkazem jsem do grafu přidal legendu.

Všimněte si argumentu *subset=*. Pomocí něj můžeme u některých grafických a statistických funkcí omezit použité pozorování pouze na ta, která splňují nějakou podmínku. V tomto případě, že jsou predikované pro řeku Svatku, resp. Bečvu. Samozřejmě bychom mohli zobrazovaná data omezit obyčejným subscriptem (obrázek je stejný a není zobrazen):

```
par(mar = c(4, 4, 1, 1))
plot(log1p(tvetbaca) ~ froude, data = both, pch = c(1, 15)[loc], col = c("grey20",
  "orange2")[loc])
lines(tvetbaca[loc == "svratka"] ~ froude[loc == "svratka"], data = pred.data,
  col = "orange2")
lines(tvetbaca[loc == "becva"] ~ froude[loc == "becva"], data = pred.data, col = "grey20")
```

Model to rozhodně není ideální. Lepší by bylo modelovat vztah zobecněným modelem s poissonovým nebo negativně binomickým rozložením, které berou v úvahu povahu abundancí, tedy to, že jsou to nezáporné počty, s jejichž průměrem roste i rozptyl.