



Eksploracja i wizualizacja danych

Jak?

dowolnie

graficznie

intuicyjnie

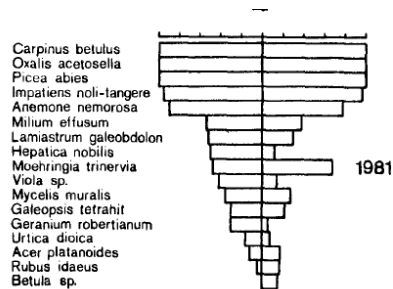


Fig. 3. Species composition of seedlings in the herb layer of a *Tilio-Carpinetum typicum* community. C – control data, E – experimental data (after uprooting of mature plants). After E. Piroznikow (mscr.).

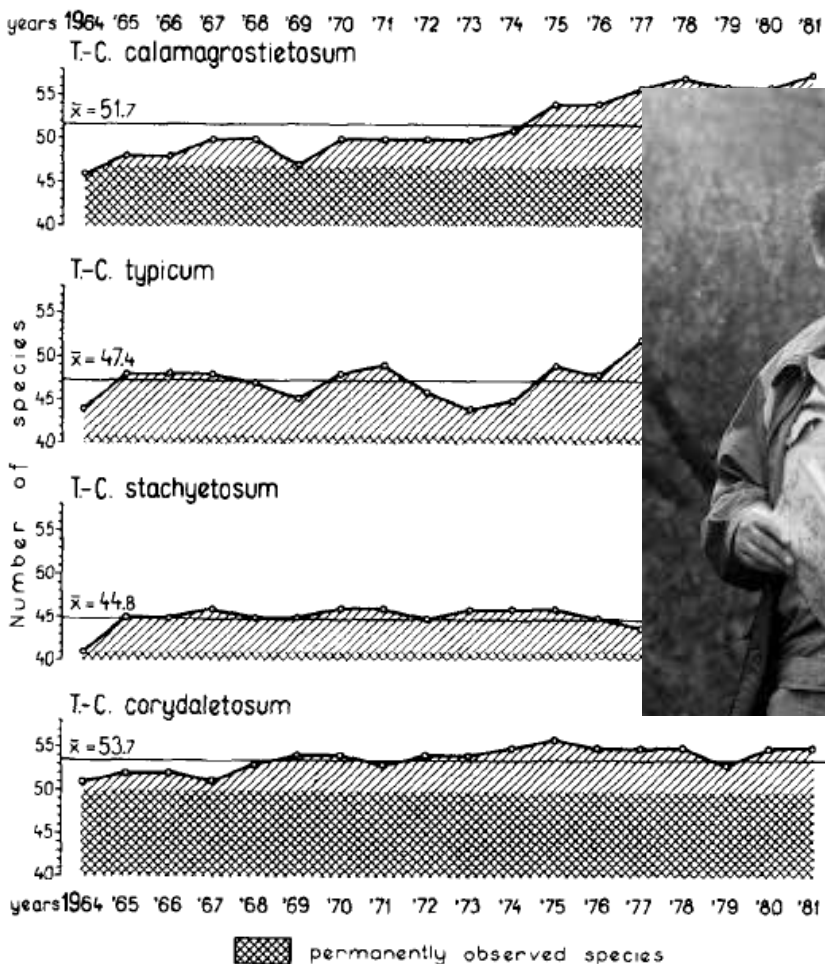


Fig. 4. Changes in the total number of herb layer species in the main subassociations of the *Tilio-Carpinetum* in the course of 18 years. a – community in regeneration, b–d – stable communities under fluctuation. (After Faliński 1986a).

Original article

The utility of ancient forest indicator species in urban environments: A case study from Poznań, Poland

Marcin K. Dyderski^{a,b}, Jarosław Tyborski^c, Andrzej M. Jagodziński^{a,b,*}

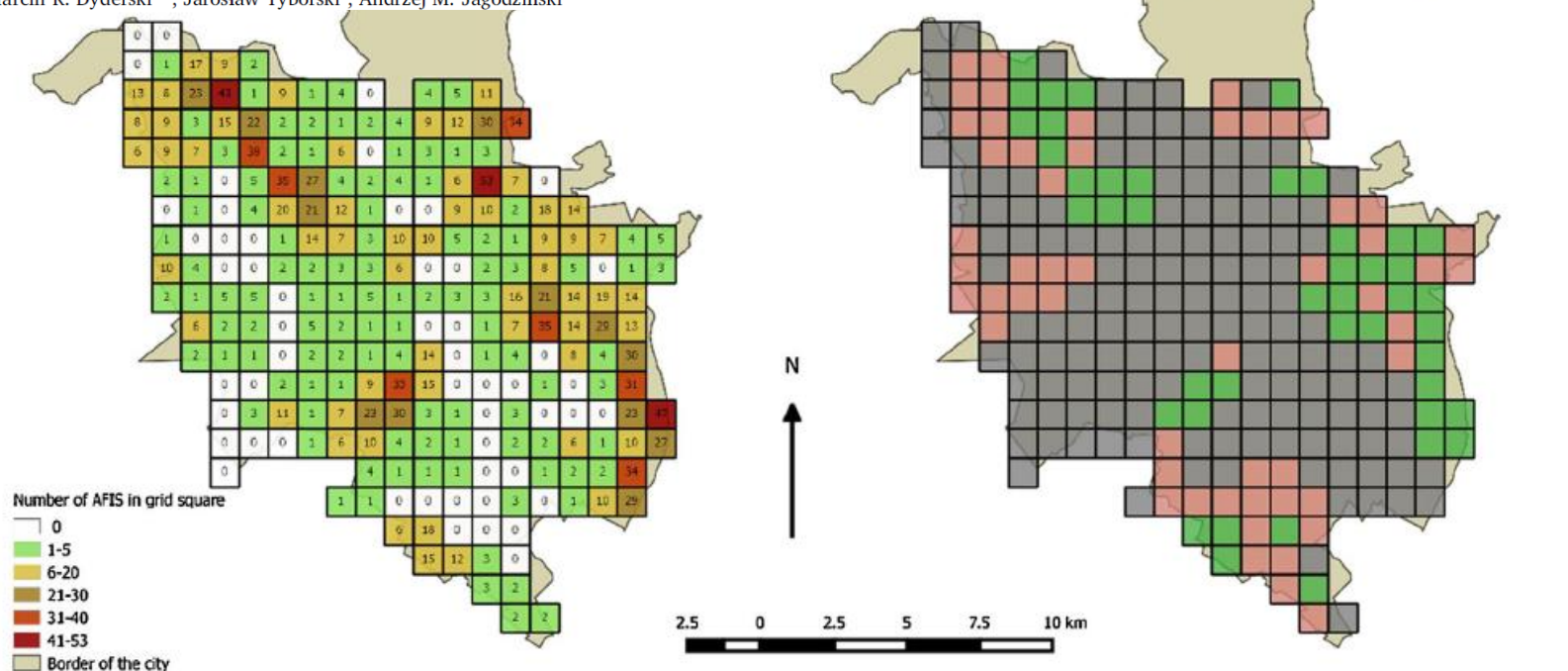


Fig. 3. Spatial distribution of ancient forest indicator species (AFIS) richness in experimental grid squares: 1–number of AFIS per square, 2–distribution of forest categories.

Zaczynamy od zbioru danych

```
eks<-read.csv('datasety/vege_1517_traits.csv',sep=';')
```

dane o gatunkach roślin w Wielkopolskim PN

cechy funkcjonalne, wskaźniki ekologiczne

z czym mamy do czynienia?

```
class(eks)
```

```
[1] "data.frame"
```

```
colnames(eks)
```

str()

str(eks)

```
'data.frame':      312 obs. of  24 variables:
 $ Species      : Factor w/ 312 levels "Acer campestre",...: 1 2 3 4 5 6 7 8 9 10
 ...
 $ hg           : Factor w/ 5 levels "ap","arch","ef",...: 1 4 4 1 1 1 1 5 1 4 ...
 $ class       : Factor w/ 21 levels "0","aln","art vul",...: 14 1 1 14 14 10 3 14 3
 1 ...
 $ stare.lasy  : int  0 0 0 0 0 0 0 1 1 0 ...
 $ L           : int  5 NA 5 4 4 8 8 5 5 NA ...
 $ T           : int  6 NA 6 6 NA NA 7 NA 5 NA ..
```

int - liczby całkowite

num - liczby rzeczywiste

factor - kategorie (uporządkowane)

chr - tekst

Po co nam ta wiedza?

Jeśli wrzucimy jakąś kolumnę do funkcji obsługującej liczby (int/num) i wyskoczy error lub NA to znaczy że mamy do czynienia z innym typem zmiennej. Jest to pierwsza rzecz którą trzeba sprawdzić

summary()

summary(eks)

Species	hg	class
Acer campestre	: 1 ap :149 0	:60
Acer ginnala	: 1 arch: 12 art vul:50	
Acer negundo	: 1 ef : 2 que fag:50	
Acer platanoides	: 1 kn : 37 mol arr:36	
Acer pseudoplatanus	: 1 sp :112 fes bro:15	
Achillea millefolium	: 1 ste med:13	
(Other)	:306	(Other):88

najprostszy sposób na poznanie
właściwości obiektu - dla factorów lub
characterów daje nam liczebność klas
dla numericów - statystyki opisowe
(następny slajd)

summary() to funkcja która obsługuje
różne typy obiektów - będziemy z niej
często korzystać

summary()

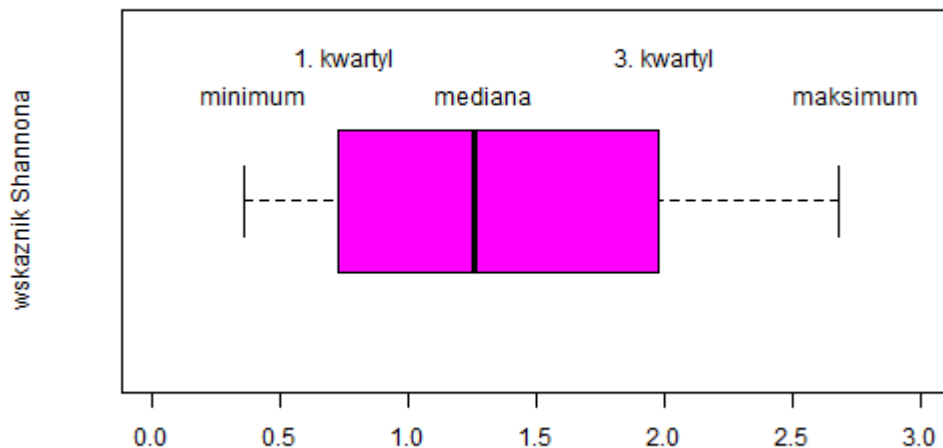
```
summary(eks)
```

	M	SR	N	wh_freq
Min.	: 2.000	Min. :1.00	Min. :1.000	Min. :-2.000
1st Qu.:	4.000	1st Qu.:5.00	1st Qu.:3.000	1st Qu.: -1.859
Median :	5.000	Median :7.00	Median :5.000	Median :-1.245
Mean :	5.108	Mean :5.87	Mean :5.149	Mean :-1.193
3rd Qu.:	6.000	3rd Qu.:7.00	3rd Qu.:7.000	3rd Qu.: -0.614
Max. :	11.000	Max. :9.00	Max. :9.000	Max. : 0.259
NA's :	62	NA's :120	NA's :84	NA's :53

Miary dyspersji a miary położenia

dyspersji - wariancja, SD, SE, rozstęp międzykwartyłowy

położenia - średnia, mediana, moda



Wykres pudełkowy,
pudełko z wąsami,
boxplot

`boxplot(tabela$kolumna)`

```
summary(eks$SLA)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
4.75	18.01	23.89	25.06	29.04	144.78	54

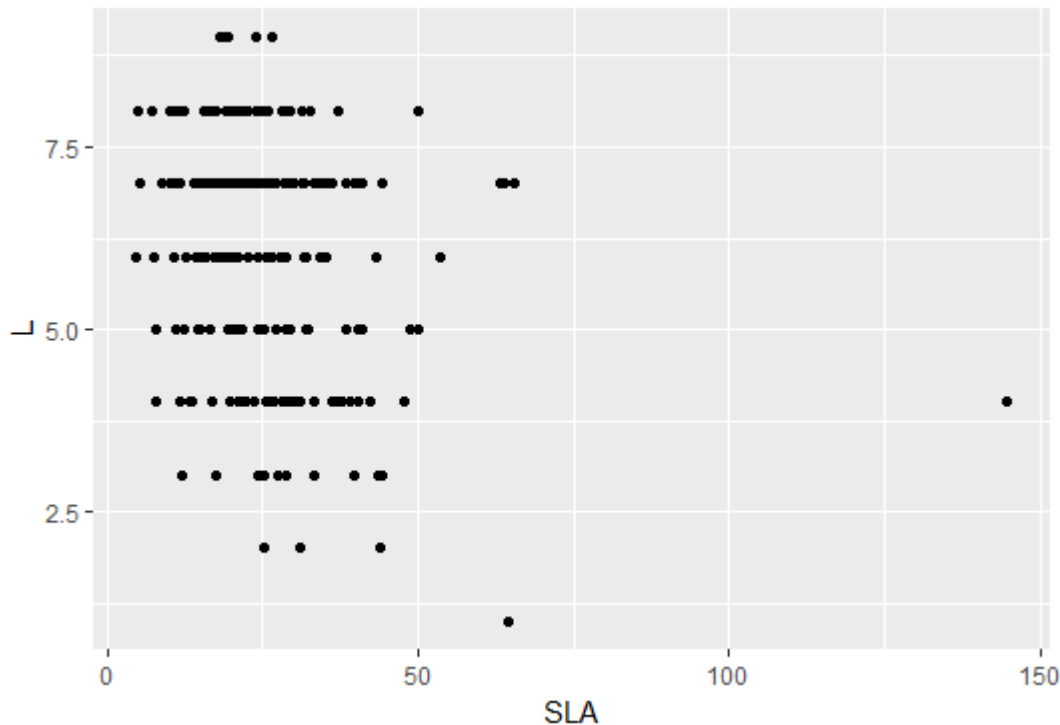
rozkład?

liczba braków danych?

średnia?

Poprawność danych

```
library(ggplot2) #lub library(tidyverse)
ggplot(eks, aes(x=SLA, y=L))+geom_point()
```

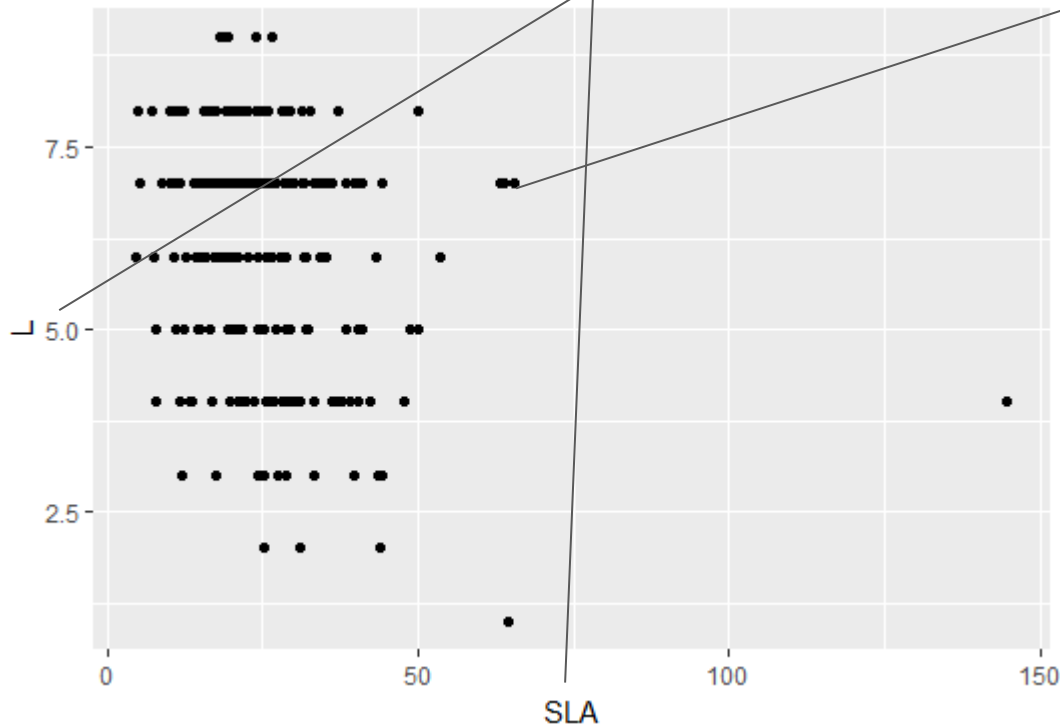


składnia:

ggplot() wywołuje wykres
pierwszy argument (eks) - to nazwa
tabeli z danymi
aes() odpowiada za mapowanie -
wyświetlanie elementów w zależności
od danej zmiennej

samo ggplot() da nam wykres bez
elementów
wywołujemy je za pomocą +geom_*()
każdy typ ma swój geom_*
np. geom_point() to punkty
geom_line() to linie
geom_col() to słupki

```
ggplot(eks, aes(x=SLA, y=L))+geom_point()
```

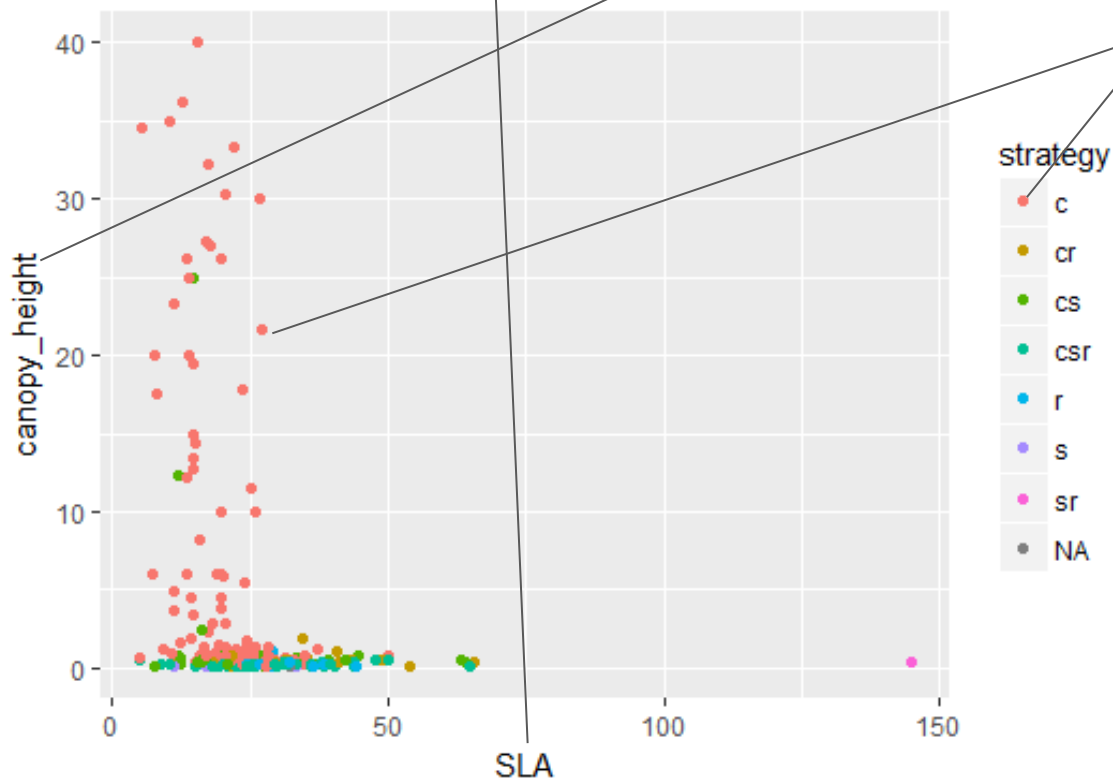


składnia:

ggplot() wywołuje wykres
pierwszy argument (eks) - to nazwa
tabeli z danymi
aes() odpowiada za mapowanie -
wyświetlanie elementów w zależności
od danej zmiennej

samo ggplot() da nam wykres bez
elementów
wywołujemy je za pomocą +geom_*()
każdy typ ma swój geom_*
np. geom_point() to punkty
geom_line() to linie
geom_col() to słupki

```
ggplot(eks, aes(x=SLA, y=canopy_height, col=strategy))+geom_point()
```



Filozofia ggplot

`ggplot(dane, aes(zmienne))+elementy:`

`+geom_point()` - punkty

`+geom_col()` - kolumny

`+geom_smooth()` - linie regresji

`aes` - aesthetics - elementy do pokazania na wykresach:

`x,y` - osie; `col` - kolor linii, `fill` - wypełnienie, `shape` - kształt, `size`...

Bardzo dobra dokumentacja

<http://ggplot2.tidyverse.org/index.html>

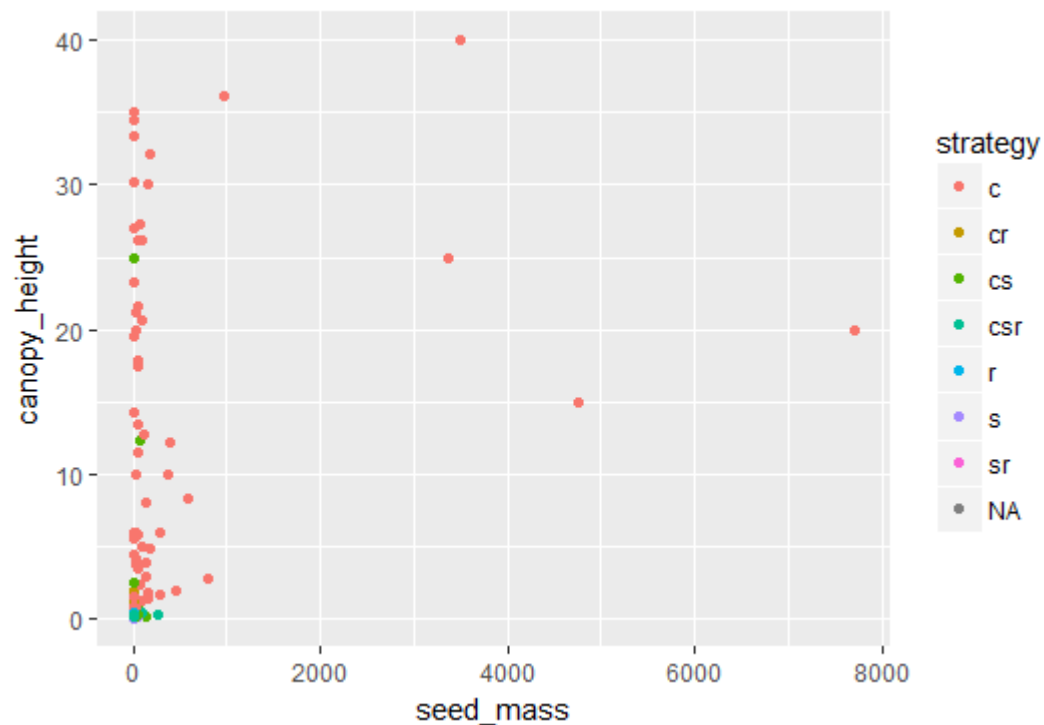
<http://r-statistics.co/ggplot2-cheatsheet.html>

<https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>

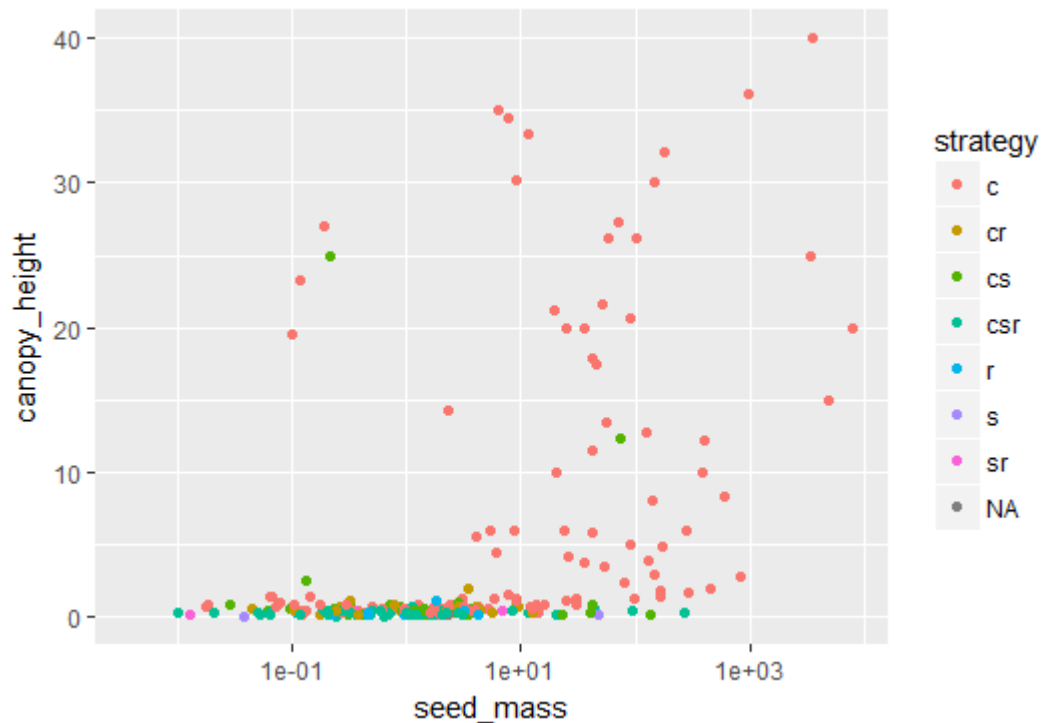
różnica między `library(ggplot2)` a `library(tidyverse)`:

`ggplot2` to tylko obrazki, `tidyverse` oprócz `ggplot2` zawiera wiele innych pakietów
wygodniej wczytywać `tidyverse`, ale czasem mogą być konflikty między pakietami


```
ggplot(eks, aes(x=seed_mass, y=canopy_height, col=strategy))+geom_point()
```

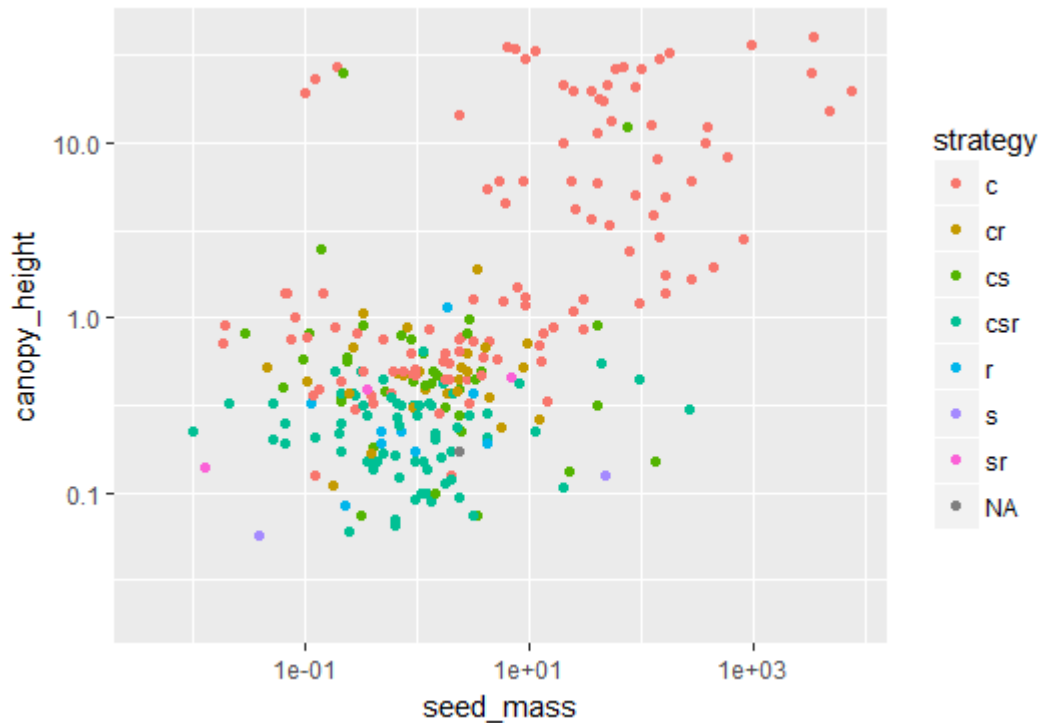


```
ggplot(eks, aes(x=seed_mass, y=canopy_height,  
col=strategy))+geom_point()+scale_x_log10()
```



oś logarytmiczna
transformacja logarytmiczna
logarytm

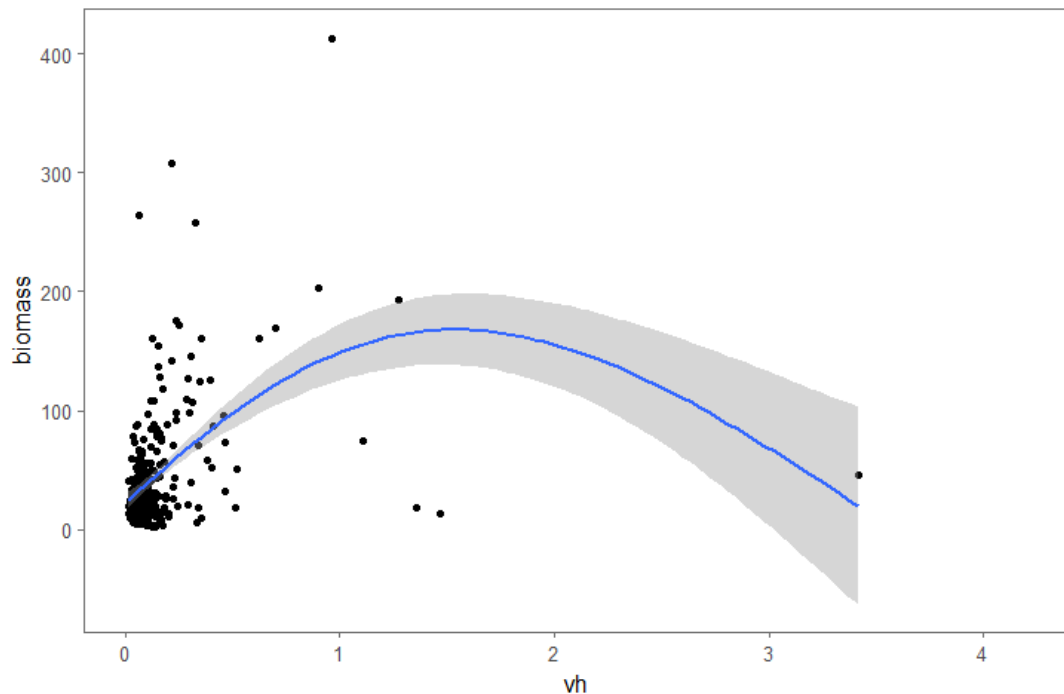
```
ggplot(eks, aes(x=seed_mass, y=canopy_height,  
col=strategy))+geom_point()+scale_x_log10()+scale_y_log10()
```



Poprawność danych

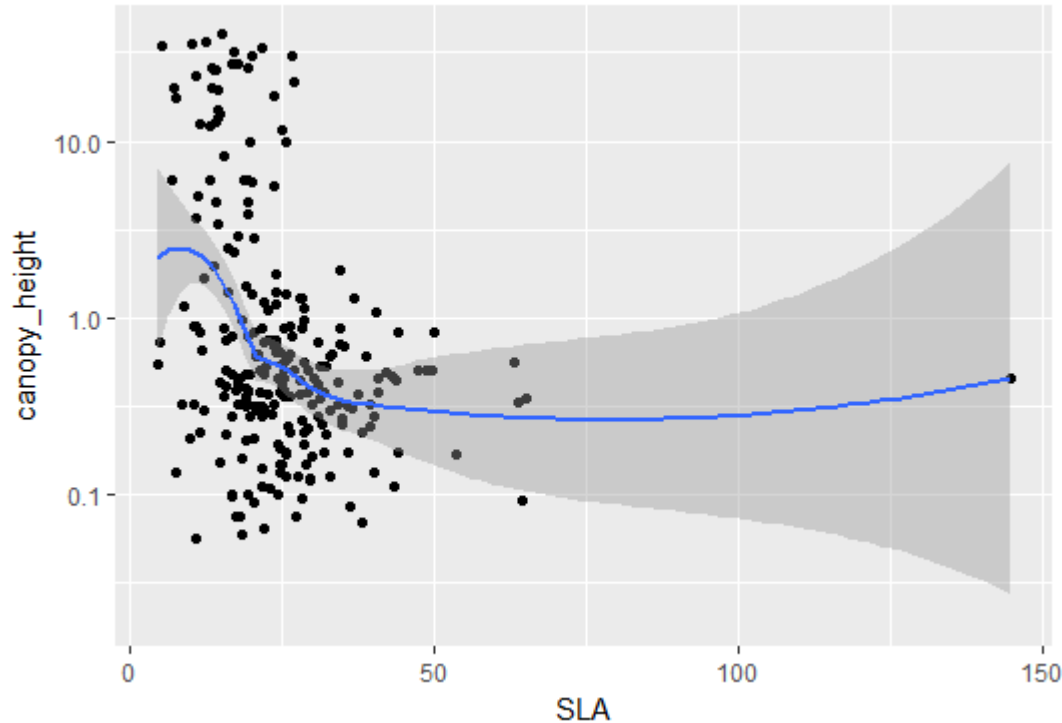
sprawdzamy

```
ggplot(dane, aes(x=vh,y=biomass))+geom_point()+geom_smooth()
```

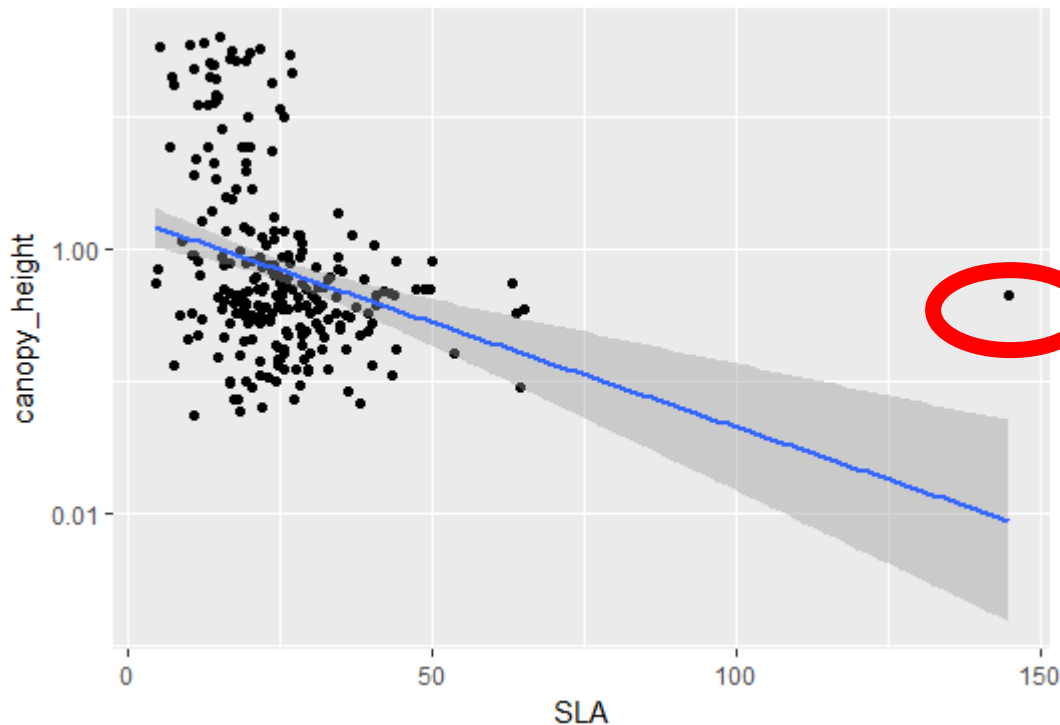


na razie dopasowujemy linie trendu
bez refleksji statystycznych
środa i czwartek dostarczy ich aż
nadto

```
ggplot(eks, aes(x=SLA, y=canopy_height))+geom_point()  
+scale_y_log10()+geom_smooth()
```



```
ggplot(eks, aes(x=SLA, y=canopy_height))+geom_point()  
+scale_y_log10()+geom_smooth(method='lm')
```



aby trend był linowy
musimy zdefiniować jaki model
ma być użyty

ggplot może dopasować
modele liniowe (lm), nieliniowe
(nls), uogólnione modele
liniowe (glm), modele
addytywne (gam), czy
przybliżenia (loess),

linia trendu, linia na wykresie

co to jest za obserwacja?

E:/Nauka/stat_narz/R/BSS/bssR - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

Species hg class stare.lasy L T C M SR N wh_freq wh_sev herb_freq

1	Acer campestre	ap	que fag	0	5	6	4	5	7	6	-1.912	0.246
2	Acer ginnala	kn	0	0	NA	NA	NA	NA	NA	NA	NA	NA
3	Acer negundo	kn	0	5	6	6	6	7	7		-1.525	0.336
4	Acer platanoides	ap	que fag	0	4	6	4	NA	NA	NA	-1.963	0.231
5	Acer pseudoplatanus	ap	que fag	0	4	NA	4	6	NA	7	-1.969	0.229
6	Achillea millefolium	ap	mol arr	0	8	NA	NA	4	NA	5	-0.612	0.451
7	Achillea nobilis	ap	art vul	0	8	7	7	4	8	1	NA	NA
8	Adoxa moschatellina	sp	que fag	1	5	NA	5	6	7	8	-1.938	0.261
9	Aegopodium podagraria	ap	art vul	1	5	5	3	6	7	8	-1.519	0.394
10	Aesculus hippocastanum	kn	0	0	NA	NA	NA	NA	NA	NA	NA	NA
11	Agrimonia eupatoria	ap	tri ger	0	7	6	4	4	8	4	-0.622	0.251
12	Aegrotia exilis	ap	mol arr	0	7	NA	7	NA	4	4	0.000	0.251

Showing 1 to 13 of 312 entries

Console R Markdown

```
E:/Nauka/stat_narz/R/BSS/bssR/  
warning messages:  
1: Removed 61 rows containing non-finite values (stat_smooth).  
2: Removed 61 rows containing missing values (geom_point).  
> ggplot(eks, aes(x=SLA, y=canopy_height))+geom_point()+scale_y_log10()  
  (+geom_smooth(method='lm'))  
Warning messages:  
1: Removed 61 rows containing non-finite values (stat_smooth).  
2: Removed 61 rows containing missing values (geom_point).  
> view(eks)  
>
```

Environment History Connections

Global Environment

Data

eks 312 obs. of 24 variables

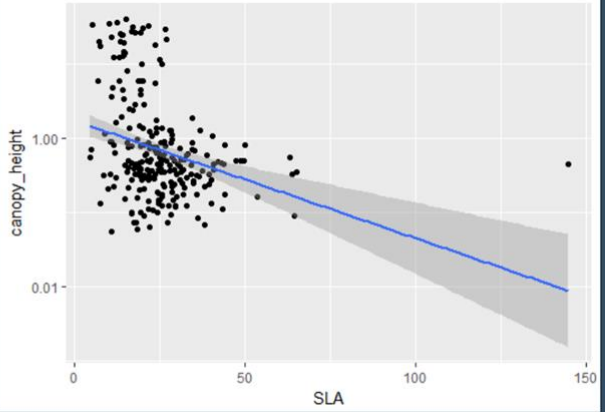
species : Factor w/ 312 levels "Acer camp...
hg : Factor w/ 5 levels "ap","arch","ef",...
class : Factor w/ 21 levels "0","aln","ar...
stare.lasy : int 0 0 0 0 0 0 1 1 0 ...
L : int 5 NA 5 4 4 8 8 5 5 NA ...

Files Plots Packages Help Viewer

Zoom Export Publish

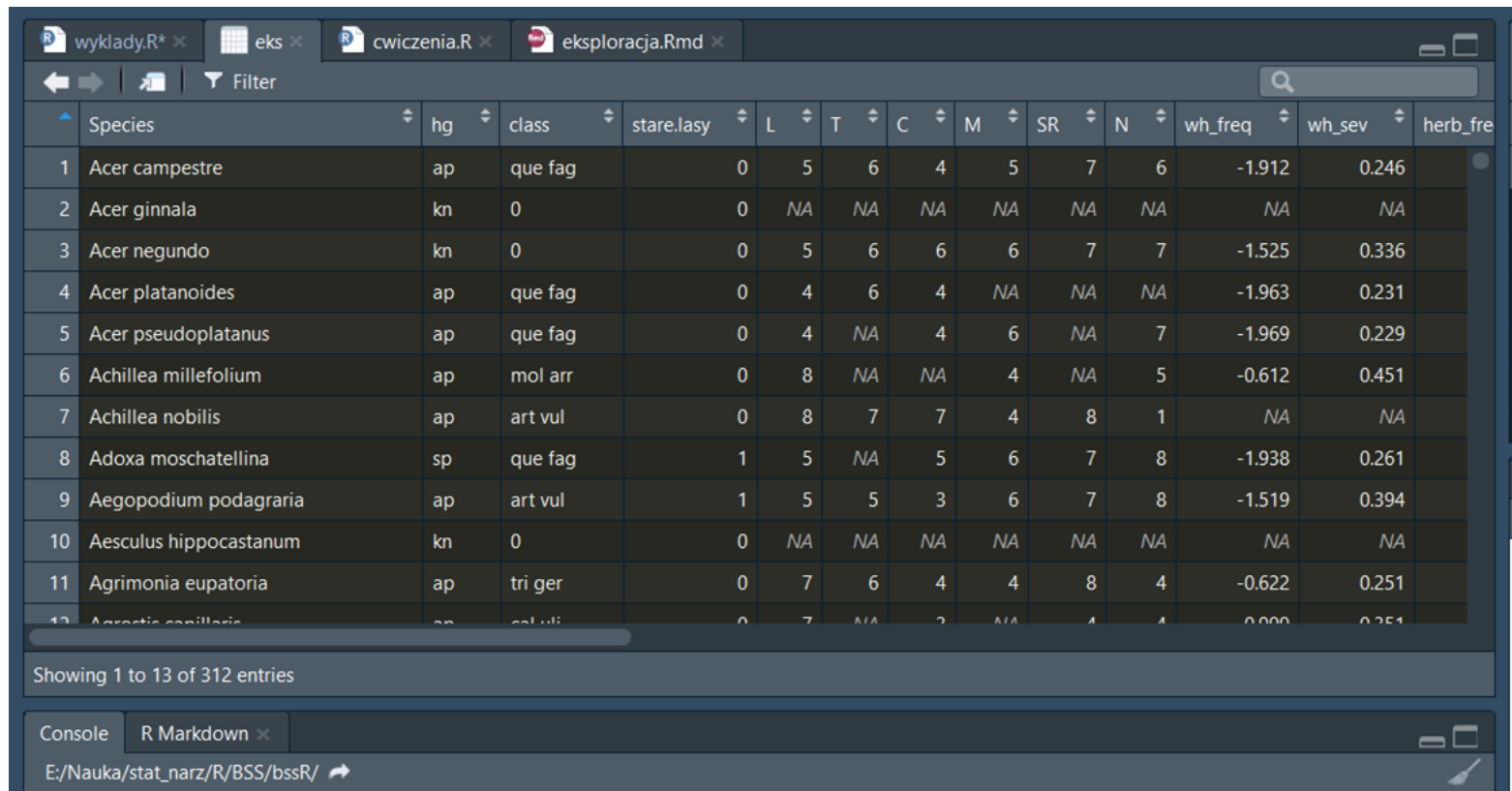
canopy_height

SLA



10:44 13.04.2018

co to jest za obserwacja?



wyklady.R* × eks × cwiczenia.R × eksploracja.Rmd ×

Filter

	Species	hg	class	stare.lasy	L	T	C	M	SR	N	wh_freq	wh_sev	herb_fre
1	Acer campestre	ap	que fag	0	5	6	4	5	7	6	-1.912	0.246	
2	Acer ginnala	kn	0	0	NA	NA	NA	NA	NA	NA	NA	NA	
3	Acer negundo	kn	0	0	5	6	6	6	7	7	-1.525	0.336	
4	Acer platanoides	ap	que fag	0	4	6	4	NA	NA	NA	-1.963	0.231	
5	Acer pseudoplatanus	ap	que fag	0	4	NA	4	6	NA	7	-1.969	0.229	
6	Achillea millefolium	ap	mol arr	0	8	NA	NA	4	NA	5	-0.612	0.451	
7	Achillea nobilis	ap	art vul	0	8	7	7	4	8	1	NA	NA	
8	Adoxa moschatellina	sp	que fag	1	5	NA	5	6	7	8	-1.938	0.261	
9	Aegopodium podagraria	ap	art vul	1	5	5	3	6	7	8	-1.519	0.394	
10	Aesculus hippocastanum	kn	0	0	NA	NA	NA	NA	NA	NA	NA	NA	
11	Agrimonia eupatoria	ap	tri ger	0	7	6	4	4	8	4	-0.622	0.251	
12	Agrostis canadensis	ap	caluli	0	7	NA	2	NA	4	4	0.000	0.251	

Showing 1 to 13 of 312 entries

Console R Markdown ×

E:/Nauka/stat_narz/R/BSS/bssR/ ↗

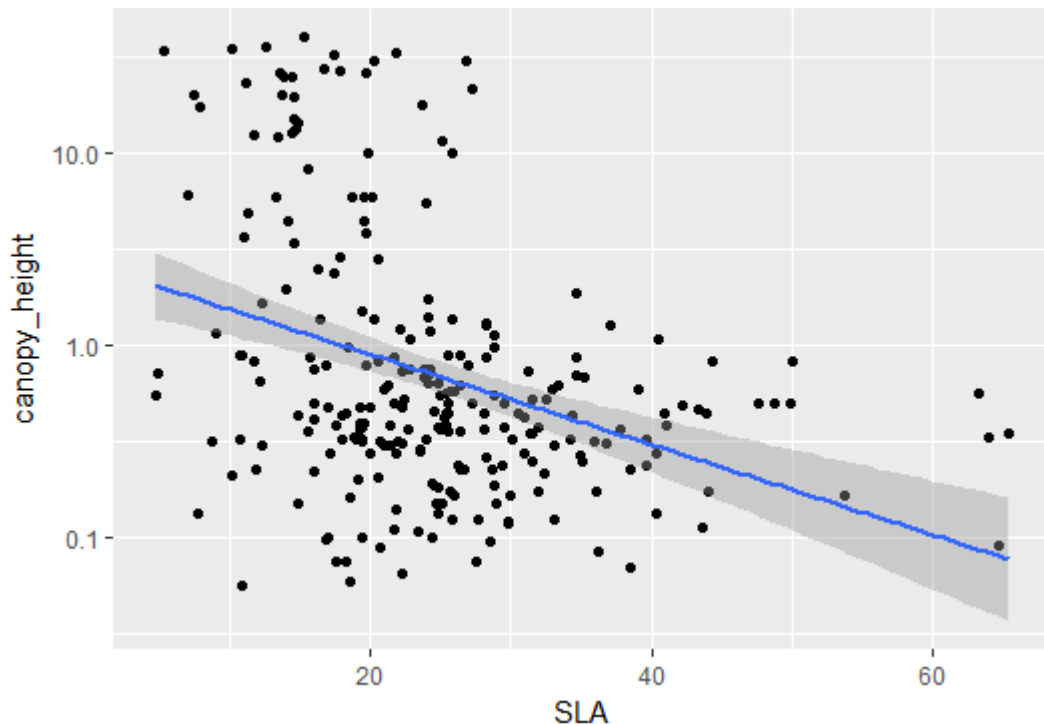
leaf_mass	leaf_size	SLA	growth_form	seed_mass	seed_number_per_shoot	reprod_B	strategy
33.0250000	4675.00000	144.77500	Therophyte	6.8926667	2.030317e+03	s (by seed/by spore)	sr
8.3400000	NA	65.36000	Therophyte	4.4090000	7.027778e+04	s (by seed/by spore)	cr
7.6890571	806.00000	64.63000	Hemicryptophyte	0.9610000	7.655378e+02	sv (by seed and vegetatively)	csr
19.7466667	1048.00000	63.94000	Hemicryptophyte	0.2039500	6.923568e+05	sv (by seed and vegetatively)	cs
29.4000000	1814.75000	63.18000	Hemicryptophyte	0.2331636	9.385806e+03	sv (by seed and vegetatively)	cs
8.7462351	488.00000	53.68108	Therophyte	0.3852000	7.782644e+03	s (by seed/by spore)	cr
NA	NA	50.00000	Hemicryptophyte	13.5575000	1.625833e+04	s (by seed/by spore)	c
8.8386437	536.50000	49.90000	Hemicryptophyte	0.1817455	9.484508e+03	sv (by seed and vegetatively)	csr
98.9792285	4966.00000	48.73250	Hemicryptophyte	1.0212000	1.955248e+04	s (by seed/by spore)	cr
NA	NA	47.59239	Hemicryptophyte	0.3163200	7.208058e+03	s (by seed/by spore)	csr
20.4262209	3886.54545	44.35000	Hemicryptophyte	2.7852545	1.653929e+03	sv (by seed and vegetatively)	cs
10.3735000	380.37440	44.00500	Therophyte	0.0460000	1.226250e+03	s (by seed/by spore)	cr

	Species	hg	class	stare.lasy	L	T	C	M	SR	N	wh_freq	wh_sev	herb_f
147	Impatiens parviflora	kn	art vul	0	4	6	5	5	NA	6	-1.700	0.347	
122	Galeopsis pubescens	ap	art vul	0	7	5	4	5	NA	6	-1.697	0.451	
189	Oxalis acetosella	sp	0	1	1	NA	3	5	4	6	-1.914	0.275	
186	Myosoton aquaticum	ap	art vul	0	7	5	3	8	7	8	-1.001	0.474	
168	Lycopus europaeus	ap	aln	0	7	6	5	9	7	7	-0.971	0.249	
278	Stellaria media	ap	ste med	0	6	NA	NA	NA	7	8	-0.877	0.858	
24	Arctium tomentosum	ap	art vul	0	8	5	7	5	8	9	-0.576	0.655	
210	Poa nemoralis	ap	que fag	1	5	NA	5	5	5	4	-1.871	0.282	
156	Lapsana communis	ap	art vul	0	5	6	3	5	NA	7	-1.222	0.776	
185	Mycelis muralis	sp	art vul	1	4	6	2	5	NA	6	-1.941	0.271	
43	Brachypodium sylvaticum	sp	que fag	1	3	5	3	5	6	6	-1.864	0.290	
217	Polypodium minus	sp	hid	0	7	6	3	8	5	8	-0.000	0.217	

Błąd czy nie błąd?

Co by było gdyby?

```
ggplot(subset(eks,SLA<100), aes(x=SLA,  
y=canopy_height)) +geom_point()+scale_y_log10()  
+geom_smooth(method='lm')
```



subset - podzbiór
funkcja subset daje nam część tabeli
która spełnia podane warunki

tutaj daje wszystkie wiersze dla których w
kolumnie SLA <100

można łączyć warunki używając
operatorów logicznych, np.:
subset(eks, SLA<100&canopy_height>2)

& - i

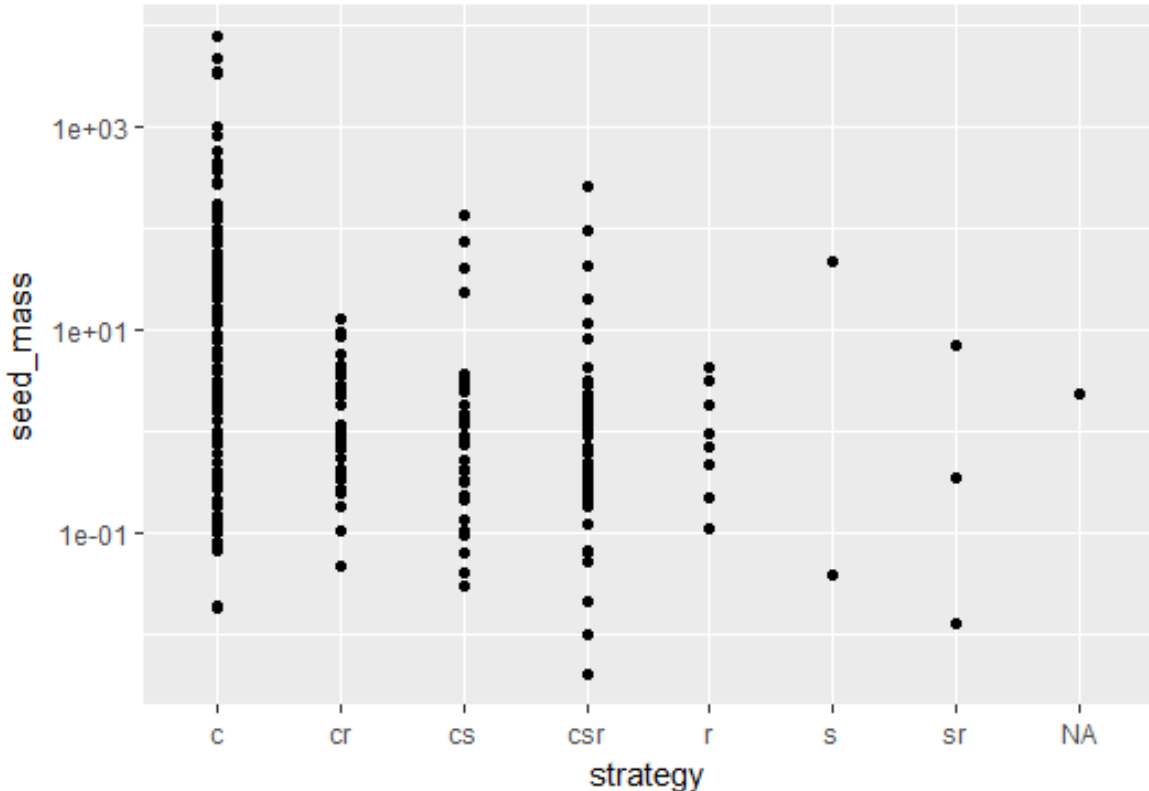
| - lub

! - nie, np. strategy!='c' - wszystkie bez c

== - równa się, strategy=='c' - tylko c

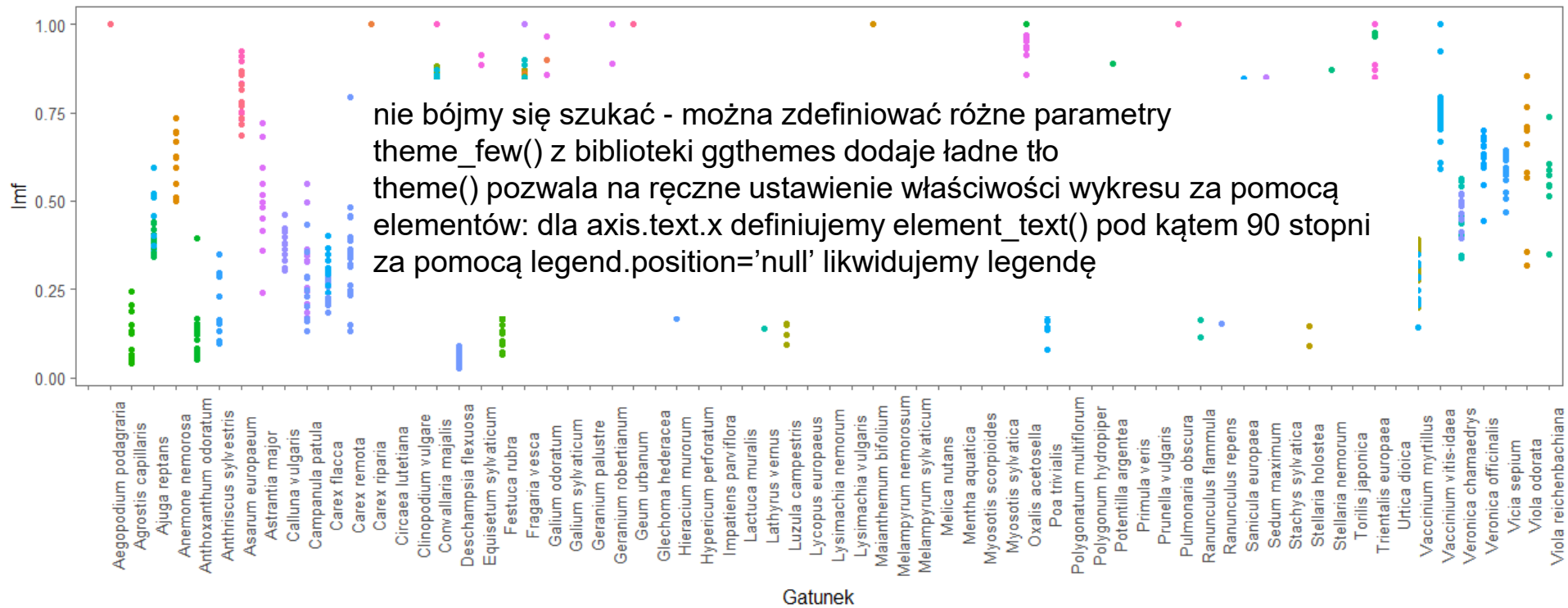
wyłącz obserwację, podzbiór, filtrowanie, selekcja,

```
ggplot(eks, aes(x=strategy, y=seed_mass))  
+geom_point()+scale_y_log10()
```

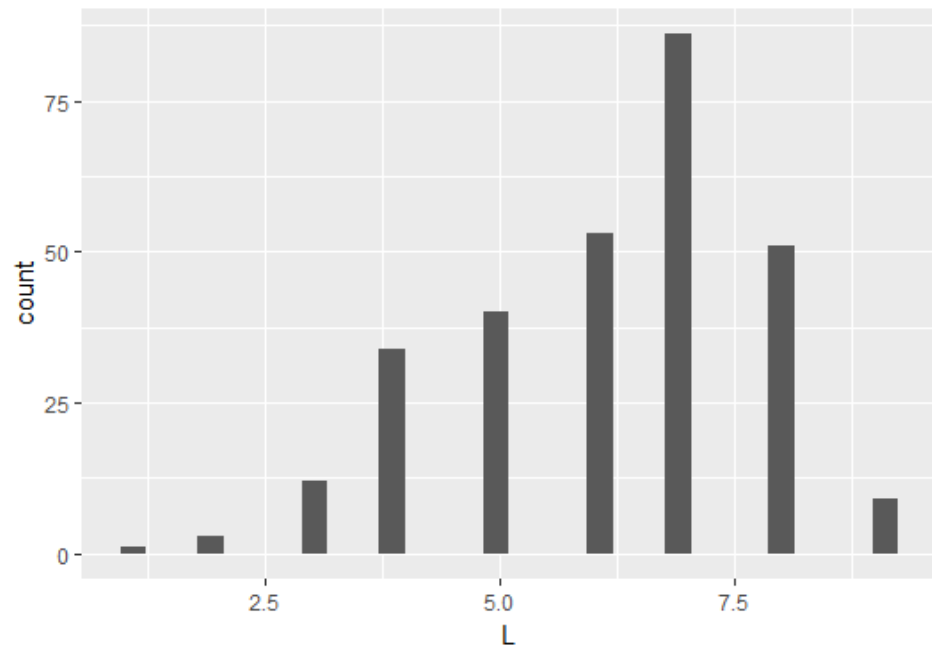


Zmienne kategoryczne
Zmienne tekstowe
Zmienne czynnikowe
Factor
Kategoria
Czynnik

```
ggplot(baza, aes(x=Gatunek,col=Kod.miejsca,  
y=lmf,label=Kod))+geom_point()+theme_few()+theme(axis.text.x=element_text(angle=90),legend.position='null')
```

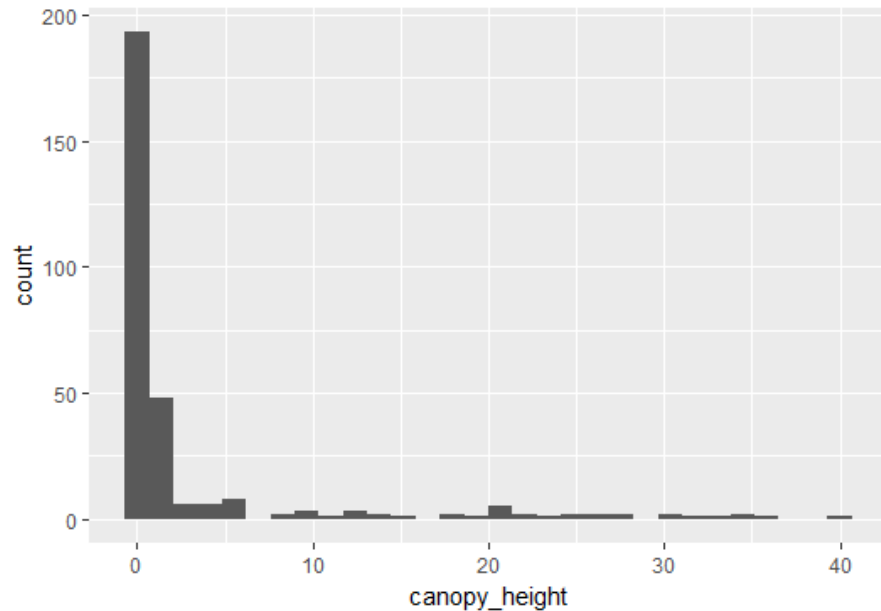


Histogram



```
ggplot(eks, aes(x=L))+geom_histogram()
```

```
ggplot(eks, aes(x=canopy_height))+geom_histogram()
```

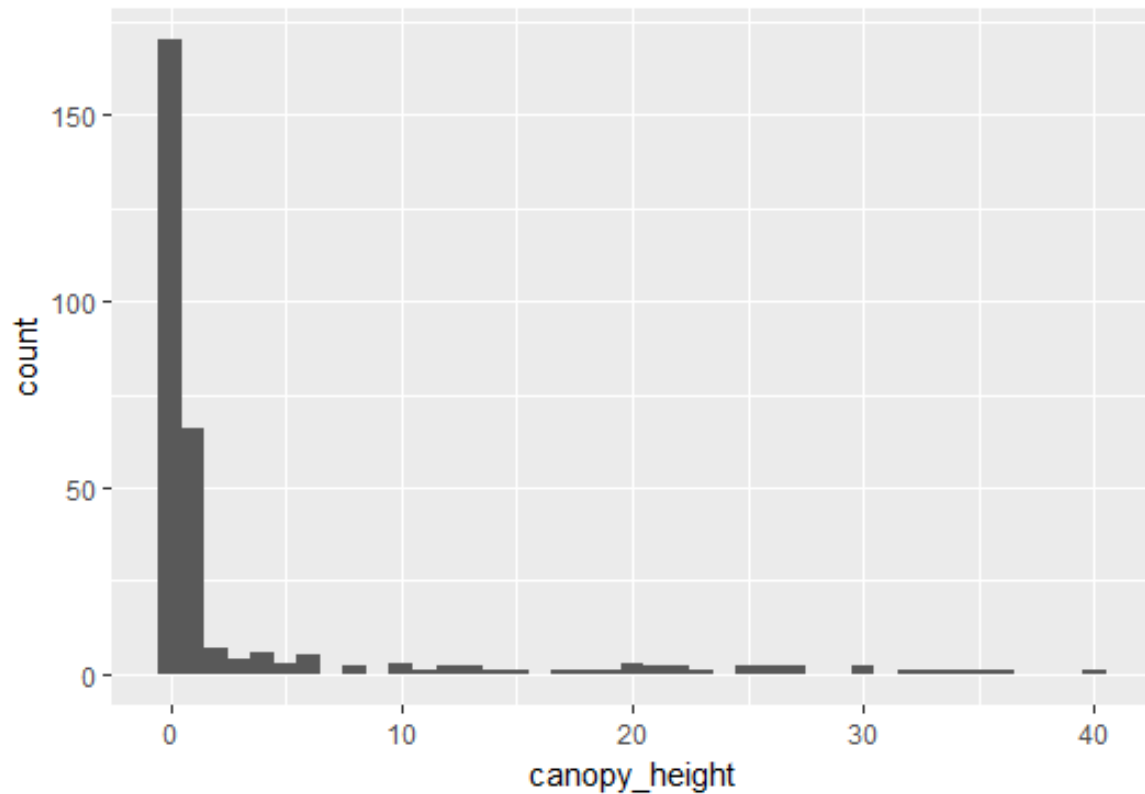


```
ggplot(eks, aes(x=M))+geom_histogram()
```

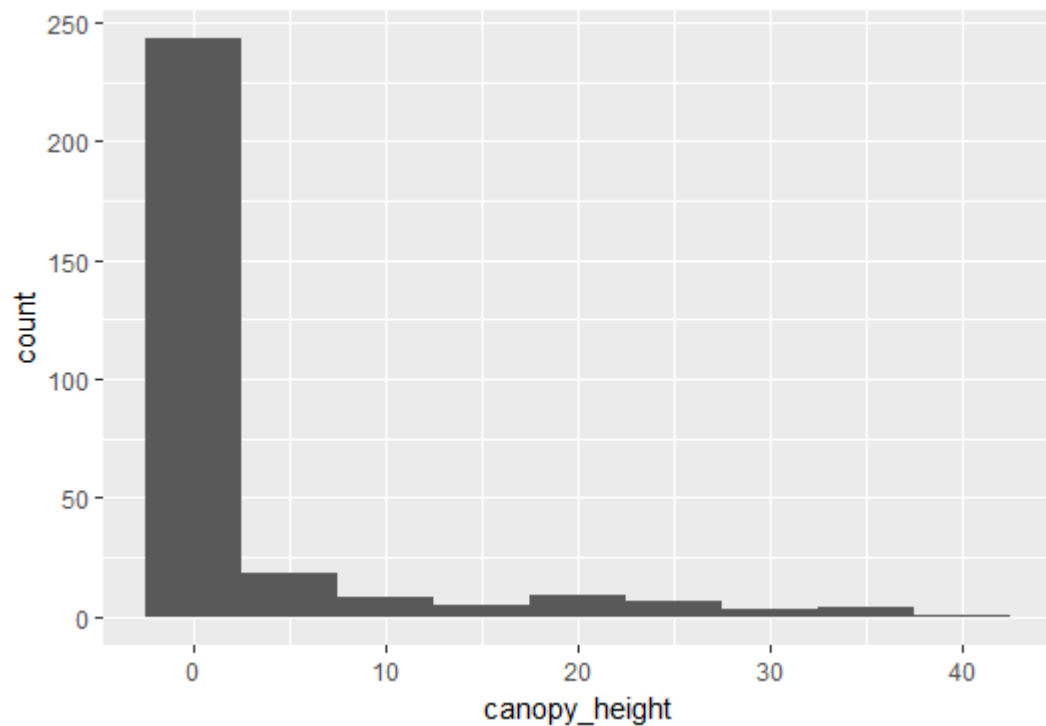
```
geom_histogram(binwidth=...)
```



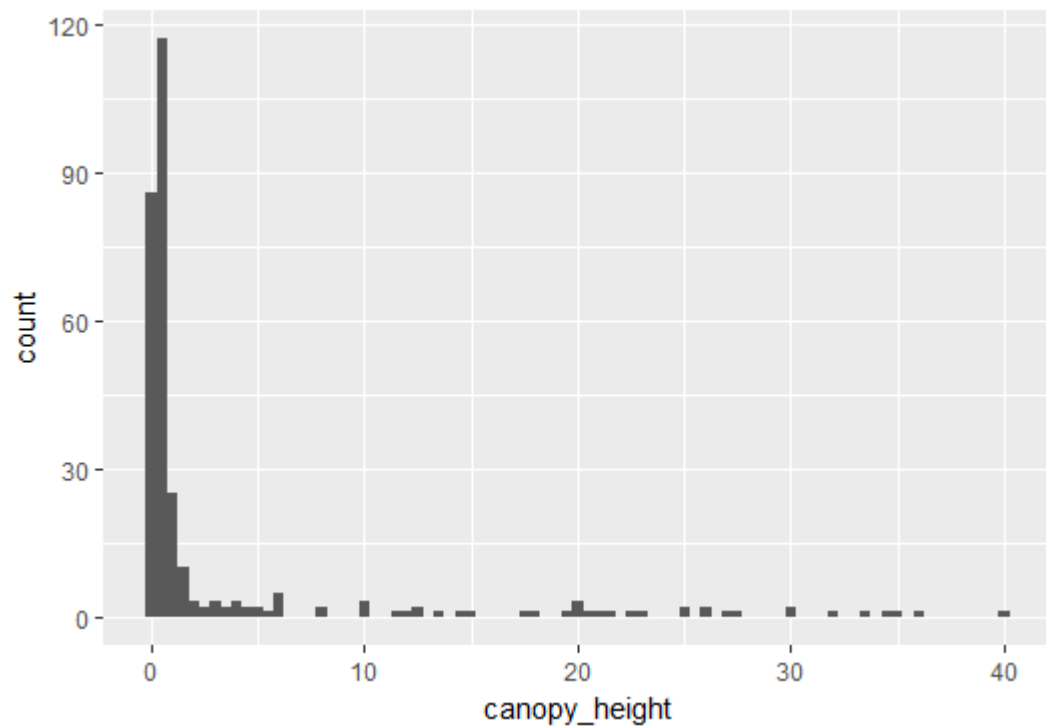
```
ggplot(eks, aes(x=canopy_height))+geom_histogram(binwidth = 1)
```



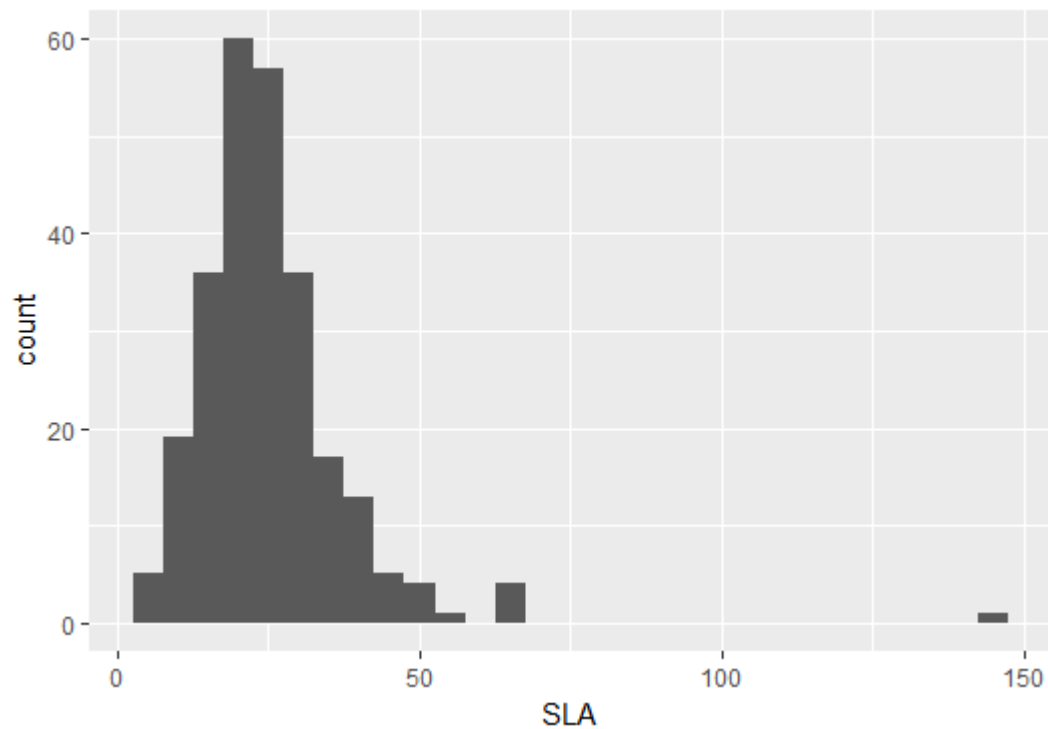
```
ggplot(eks, aes(x=canopy_height))+geom_histogram(binwidth = 5)
```



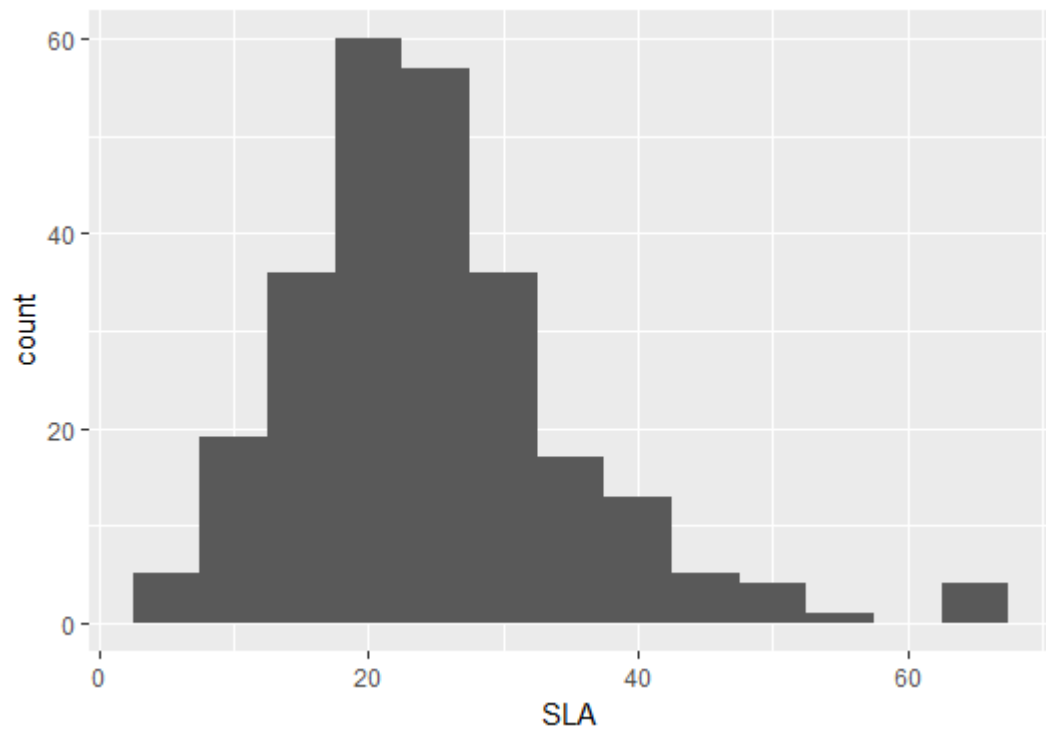
```
ggplot(eks, aes(x=canopy_height))+geom_histogram(binwidth = .5)
```



```
ggplot(eks, aes(x=SLA))+geom_histogram(binwidth = 5)
```



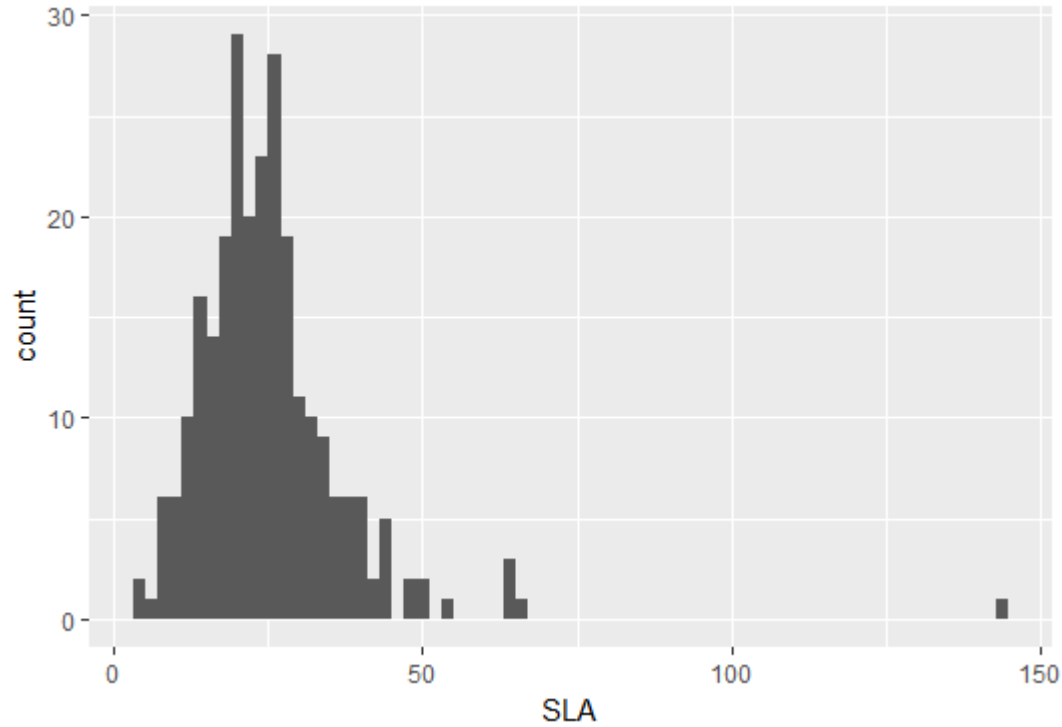
```
ggplot(subset(eks,SLA<100), aes(x=SLA))+geom_histogram(binwidth = 5)
```



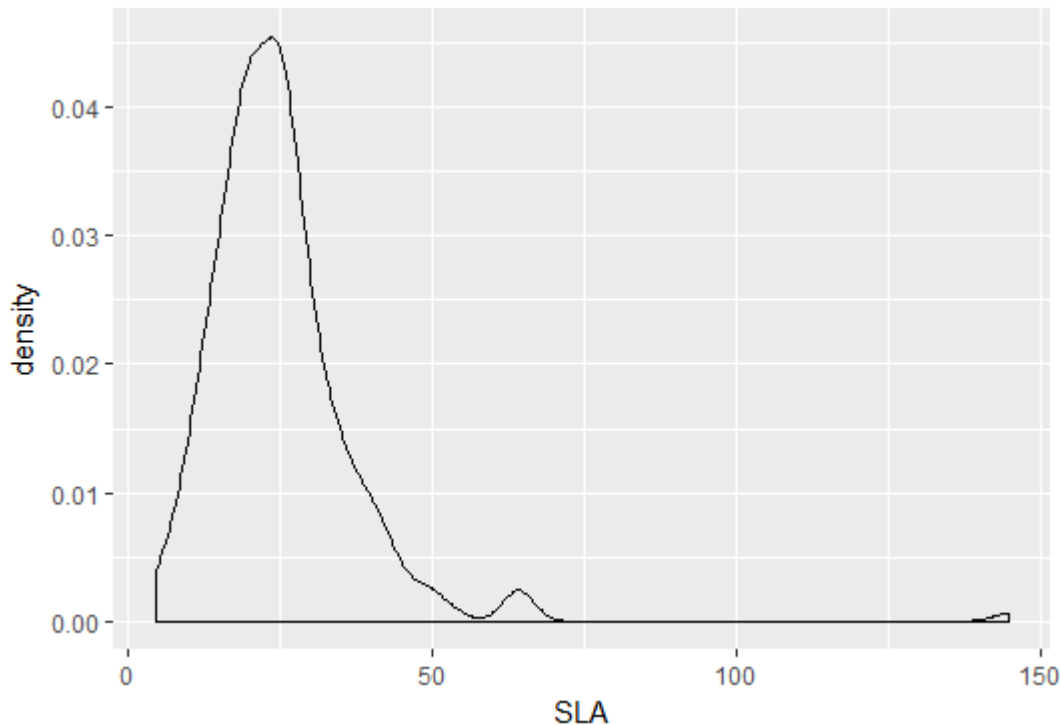
Gęstość prawdopodobieństwa

Prawdopodobieństwo, że losowo wybrany element będzie miał wartość z danego przedziału

```
ggplot(eks, aes(x=SLA))+geom_histogram(binwidth=2)
```

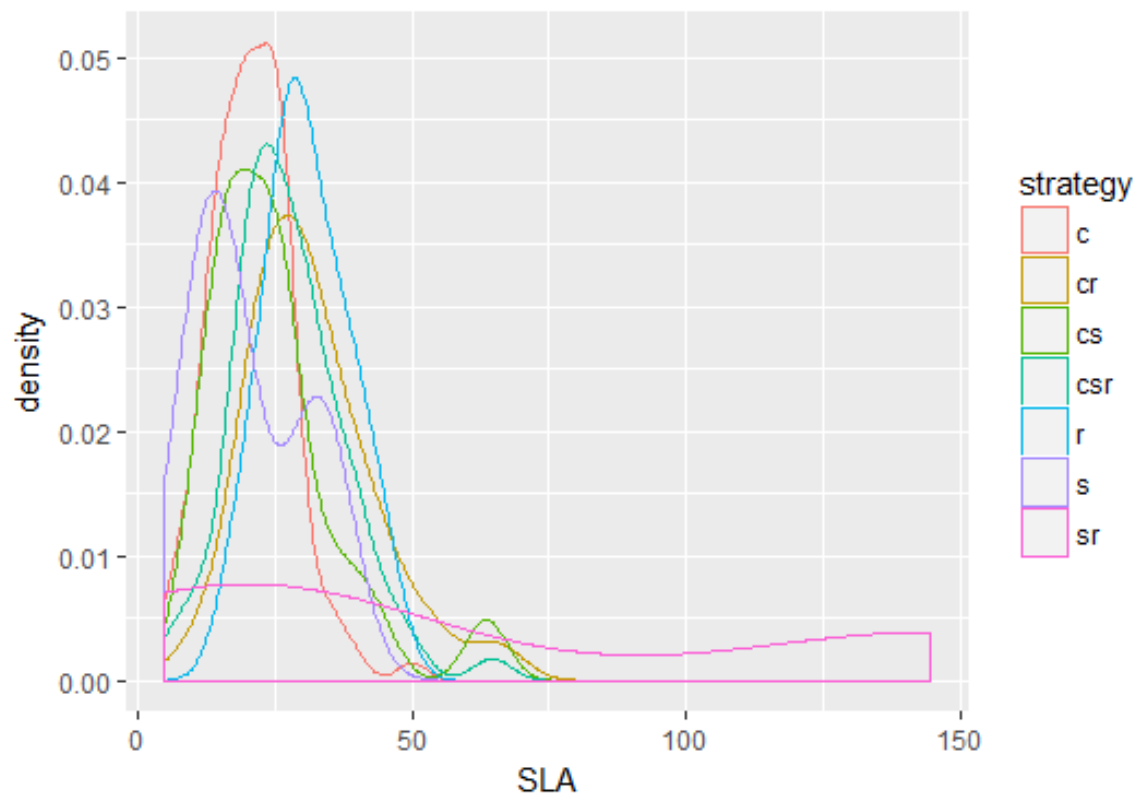


```
ggplot(eks, aes(x=SLA))+geom_density()
```



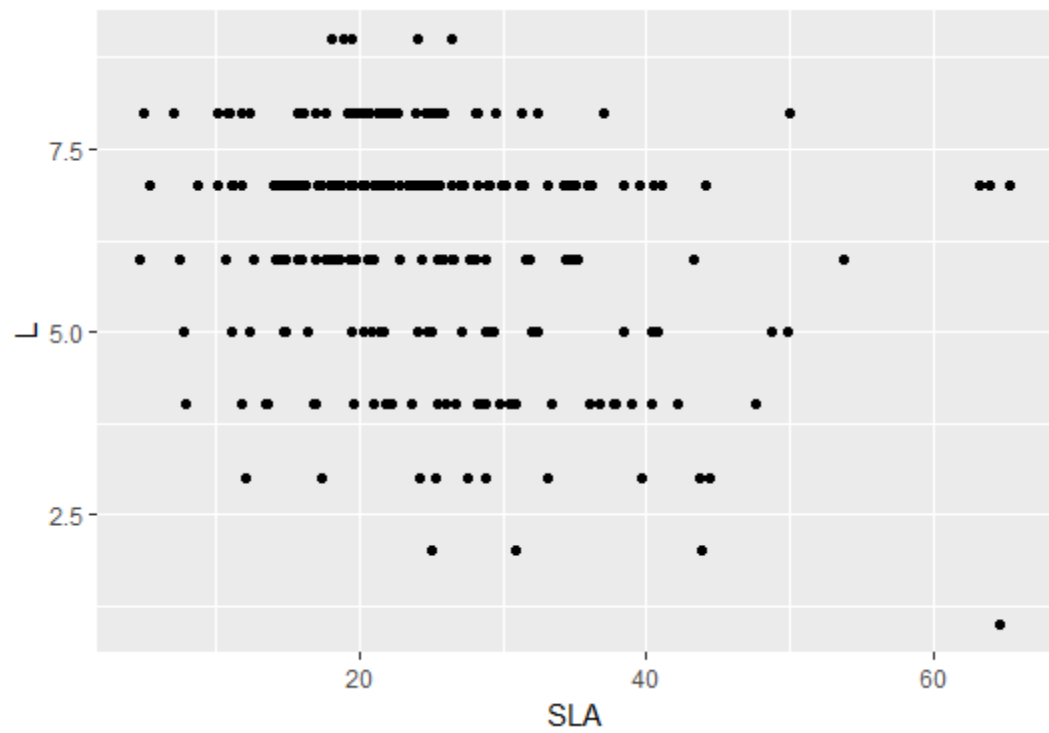
geom_density() to jądrowy
estymator gęstości
prawdopodobieństwa - warto
poczytać o nim więcej


```
ggplot(subset(eks, !is.na(strategy)), aes(x=SLA,col=strategy))+geom_density()
```

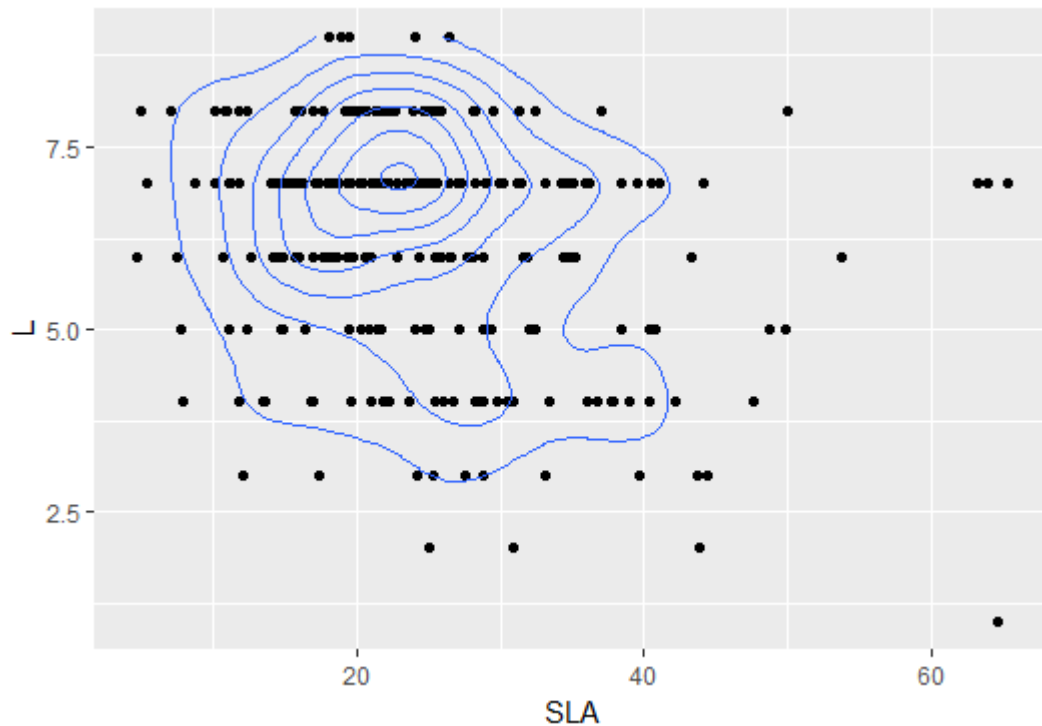


Dwuwymiarowy estymator gęstości

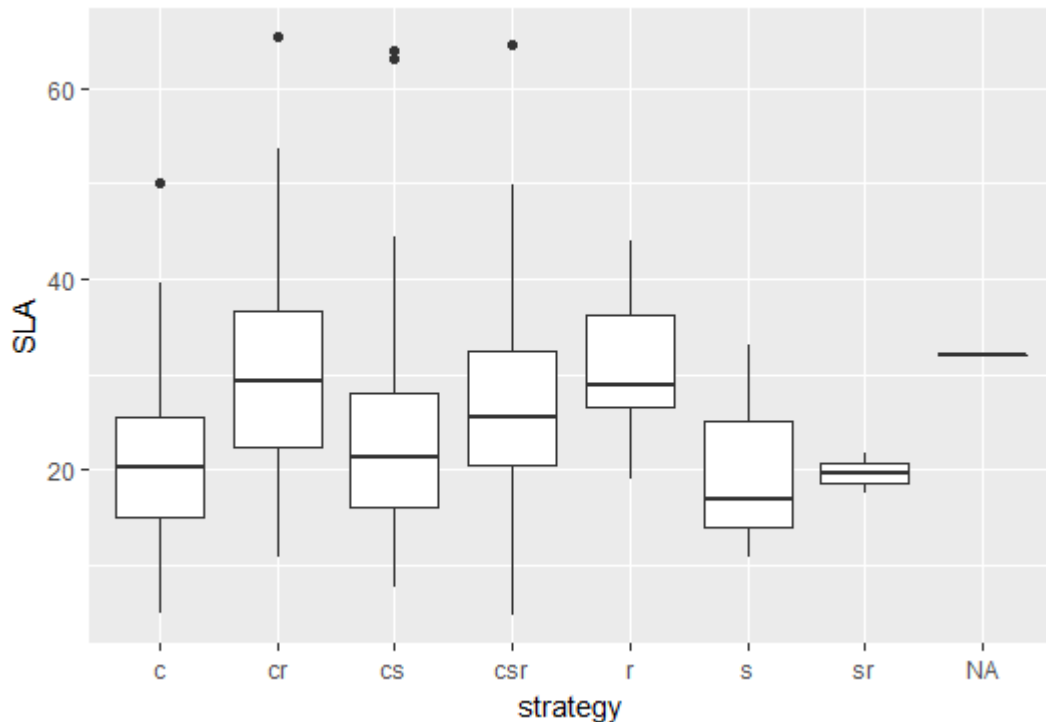
+geom_density_2d()



```
ggplot(subset(eks,SLA<100),  
aes(x=SLA,y=L))+geom_point()+geom_density_2d()
```



Boxploty - wykresy pudełkowe/pudełka z wąsami

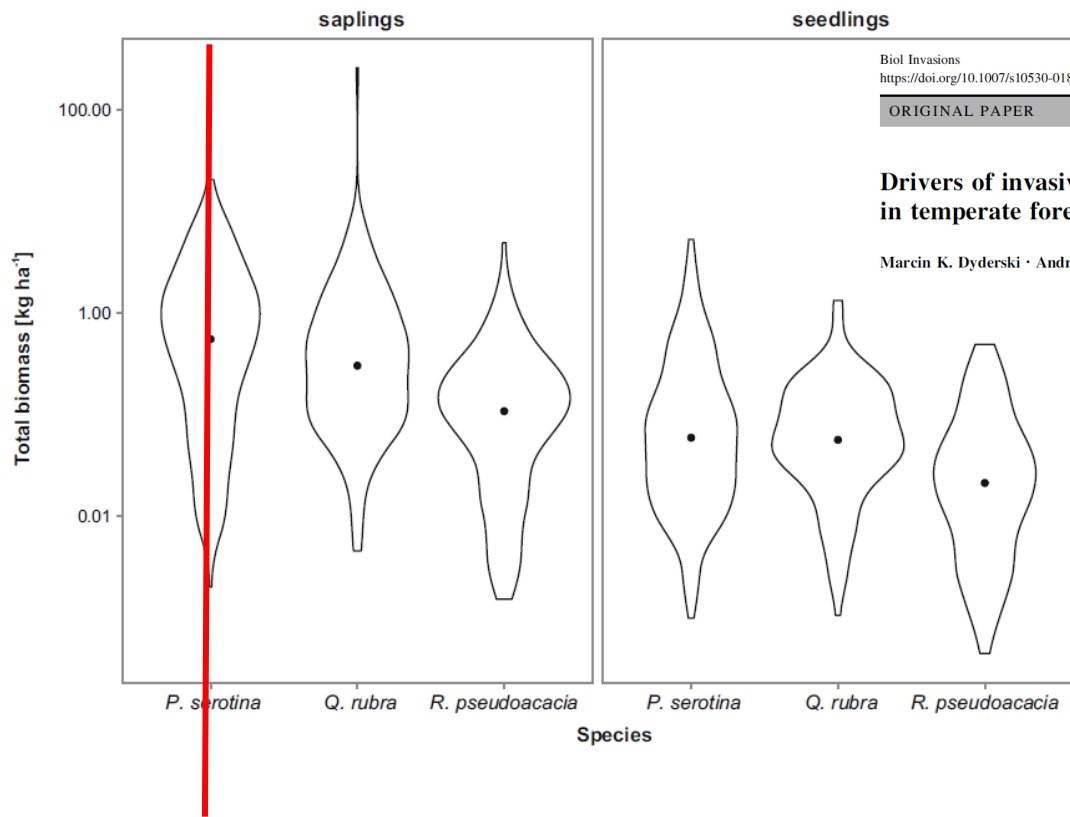


wykres pudełkowy opiera się o pięć liczb Tukeya - min, 1. i 3. kwartyl, medianę i max
kropki to obserwacje odstające - $>1,5$ rozstępu międzykwartylowego

Ważne:
wiele osób interpretuje to jako średnią i zakres SE/SD + min/max - tak jest w niektórych programach okienkowych!
trzeba zawsze napisać co jest na boxplocie, bo nie jest to oczywiste

```
ggplot(subset(eks,SLA<100), aes(x=strategy,y=SLA))+geom_boxplot()
```

Violinplots - wykresy skrzypcowe



Biol Invasions
<https://doi.org/10.1007/s10530-018-1706-3>

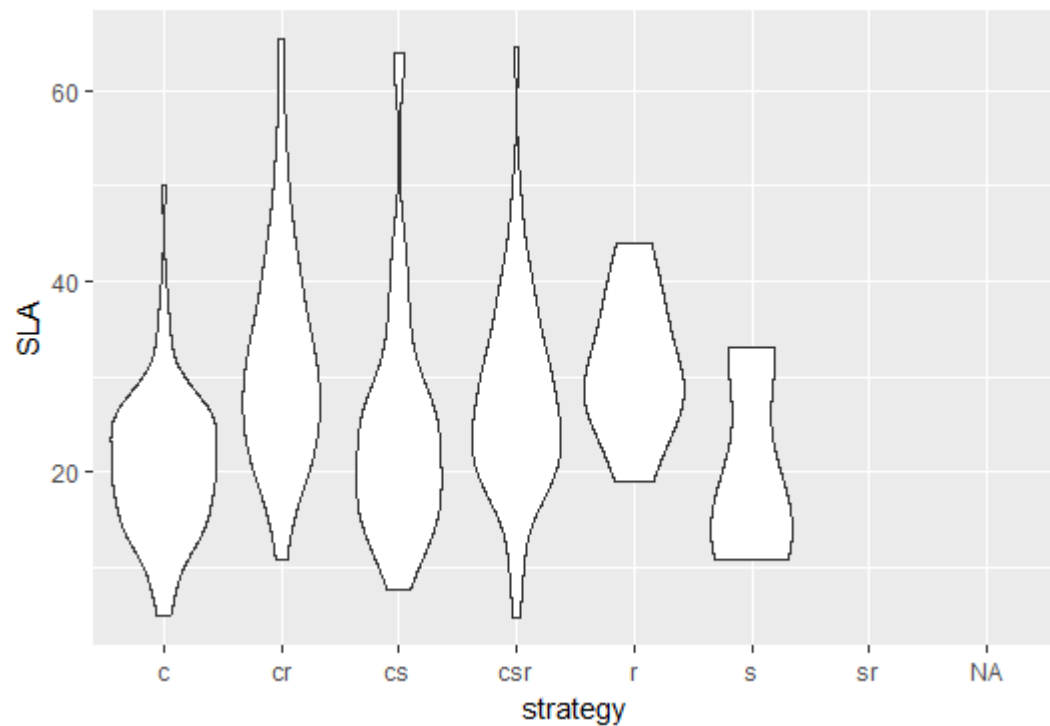


ORIGINAL PAPER

Drivers of invasive tree and shrub natural regeneration in temperate forests

Marcin K. Dyderski · Andrzej M. Jagodziński

```
ggplot(subset(eks,SLA<100), aes(x=strategy,y=SLA))+geom_violin()
```



Zawsze patrz na dane!

obrazki są podstawą pracy w analizie danych

nawet jak umiesz interpretować cyfry, łatwiej jest dojrzeć historię stojącą za danymi...

nawet jak trzeba zastosować mało standardowe sposoby prezentacji danych...

Przykład

Climate change, tourism and historical grazing influence the distribution of *Carex lachenalii* Schkuhr – A rare arctic-alpine species in the Tatra Mts



Patryk Czortek ^{a,*}, Anna Delimat ^b, Marcin K. Dyderski ^{c,d}, Antoni Zięba ^e,
Andrzej M. Jagodziński ^{c,d}, Bogdan Jaroszewicz ^a

^a Białowieża Geobotanical Station, Faculty of Biology, University of Warsaw, Sportowa 19, 17-230 Białowieża, Poland

^b W. Szafer Institute of Botany, Polish Academy of Sciences, Lubicz 46, 31-512 Kraków, Poland

^c Institute of Dendrology, Polish Academy of Sciences, Parkowa 5, 62-035 Kórnik, Poland

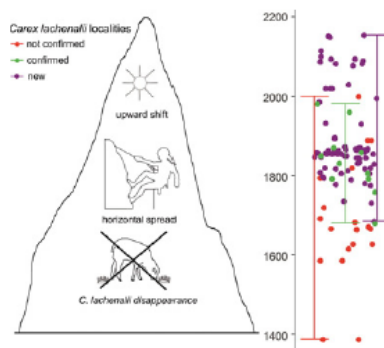
^d Department of Game Management and Forest Protection, Faculty of Forestry, Poznań University of Life Sciences, Wojska Polskiego 71c, 60-625 Poznań, Poland

^e Tatra National Park, Kuźnice 1, 34-500 Zakopane, Poland

HIGHLIGHTS

- We assessed niche shift of a model arctic-alpine species.
- Vegetation pattern shows competition – mediated retreat from lower elevations.
- Climate warming allowed colonization of higher elevations, lacking in competitors.
- Modern habitat changes both threatens and promotes rare mountain plant species.

GRAPHICAL ABSTRACT



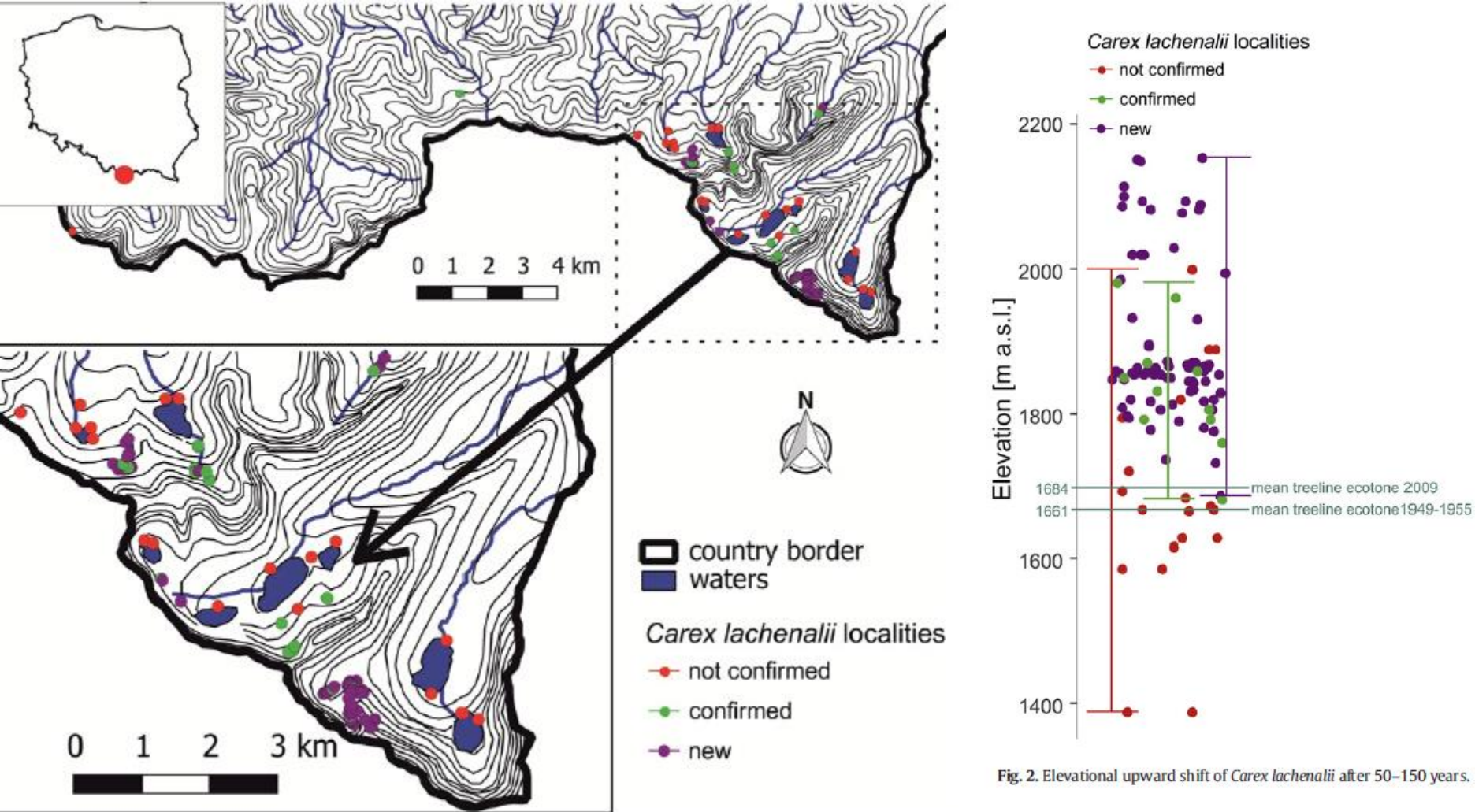


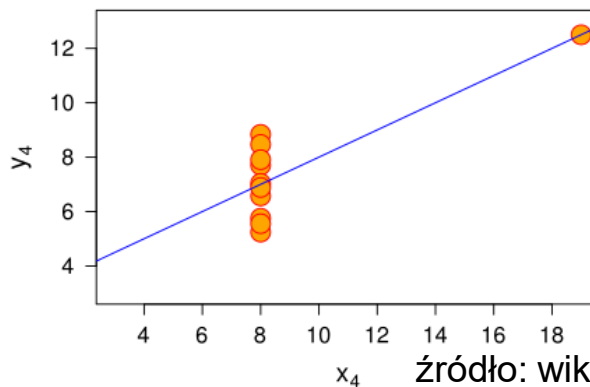
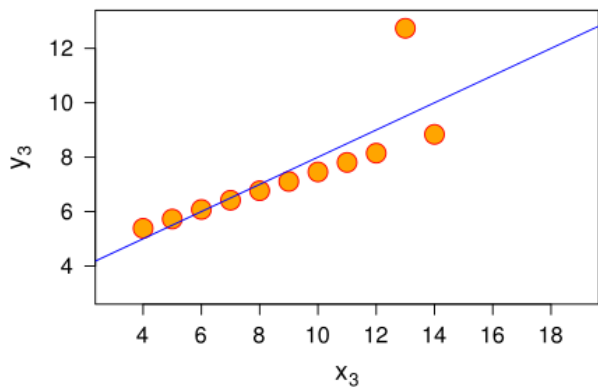
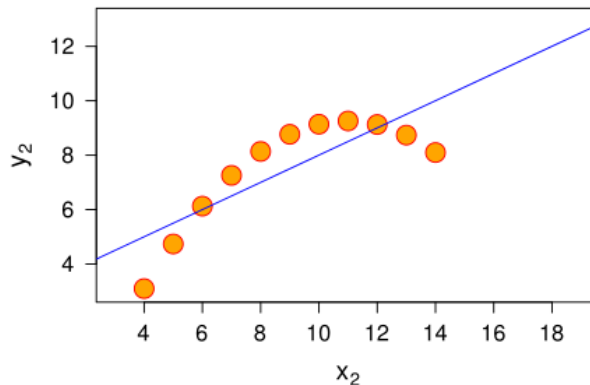
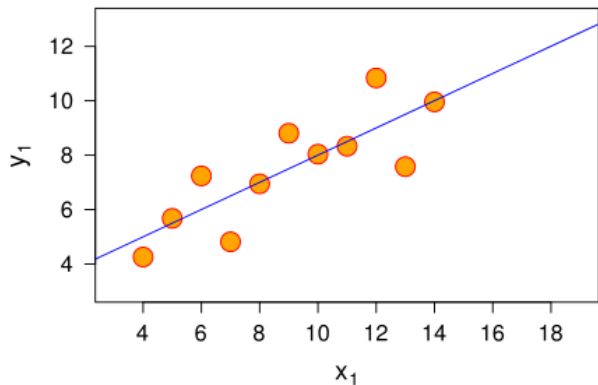
Fig. 2. Elevational upward shift of *Carex lachenalii* after 50–150 years.

Kwadrat Ascombe'a

średnia $y=7,5$ średnia $x=9$
współczynnik $r^2=0,816$
równanie regresji:
 $y=3+0,5 \cdot x$

nie
ufaj
samym
statystykom

liczby
mogą
kłamać



Dzielenie na klasy/przedziały

`cut()`

`cut(zmienna, breaks, labels)`

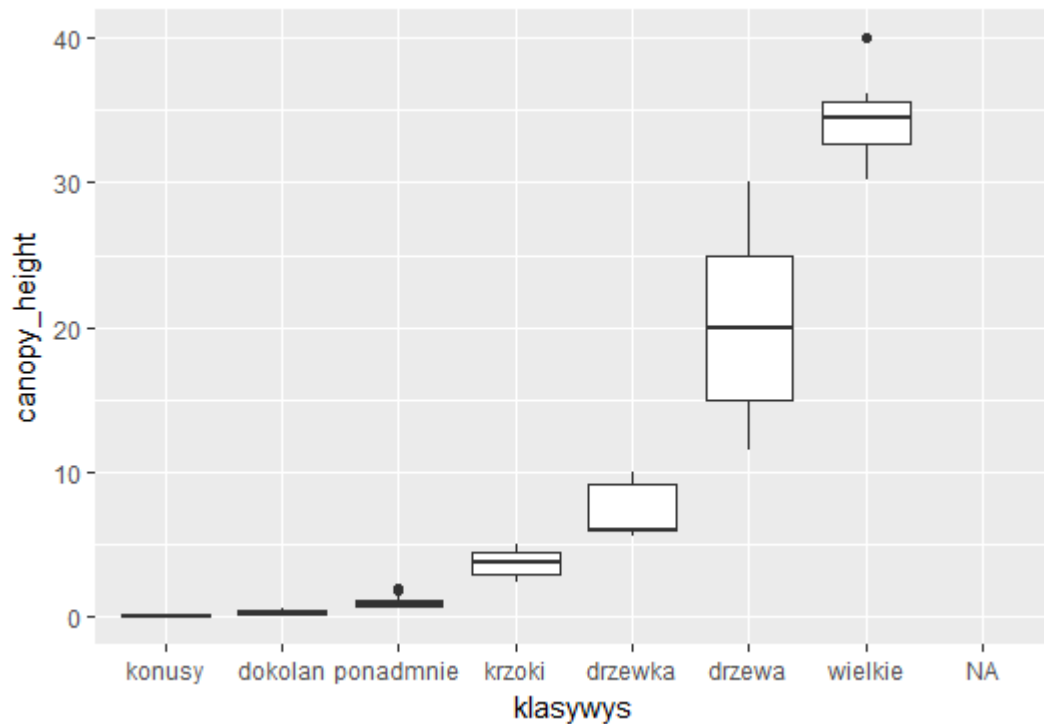
`eks$klasywys<-`

`cut(eks$canopy_height,breaks=c(0,.1,.5,2,5,10,30,50),labels=c('konusy','dokolan','ponadmnienie','krzoki','drzewka','drzewa','wielkie'))`

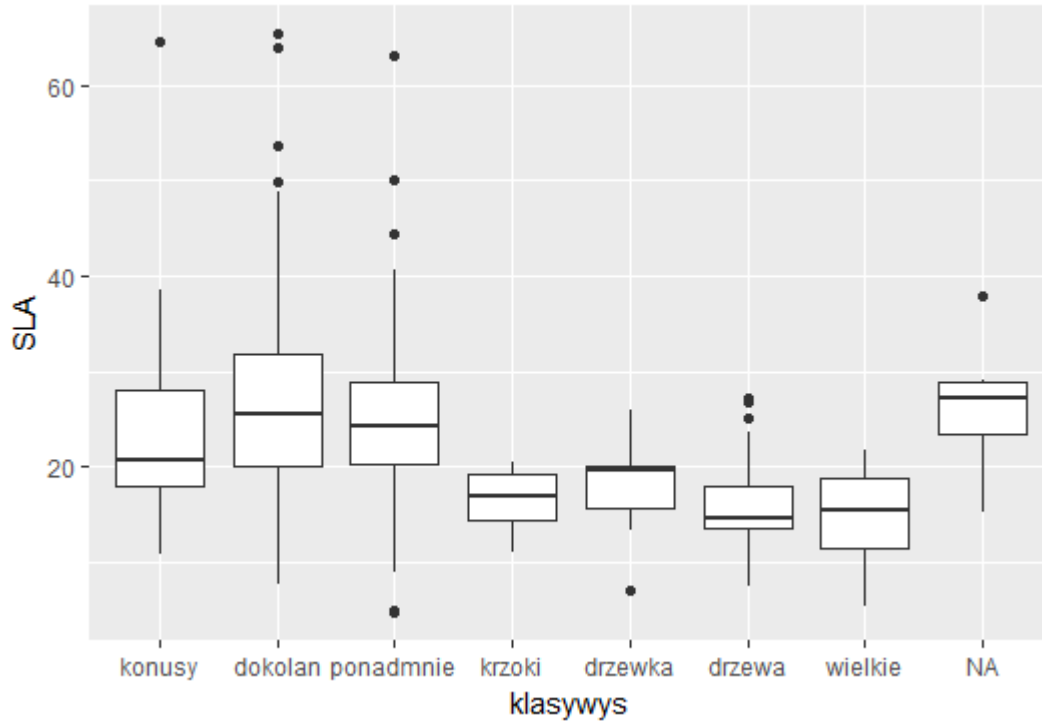
nowa zmienna

breaks musi się zaczynać czymś mniejszym niż `min(zmienna)` i kończyć większym niż `max(zmienna)`

```
ggplot(eks, aes(x=klasywys,y=canopy_height))+geom_boxplot()
```



```
ggplot(subset(eks,SLA<100), aes(x=klasywys,y=SLA))+geom_boxplot()
```



dwuwymiarowe tablice kontyngencji

12

```
> table(eks$class, eks$stare.lasy)
```

	0	1
0	53	7
aln	4	0
art vul	46	4
aspl	1	0
bid	4	0
cal uli	10	1
epi	4	0
fes bro	15	0
koe cor	11	0
mol arr	36	0
mon car	1	0
phr	3	0
pol poe	1	0
que fag	19	31
que rob-pet	11	1
rha pru	12	0
sal	1	0
sch car	2	0
ste med	13	0
tri ger	8	1
vac pic	8	4

Podsumowanie liczebności klas
Tablice liczebności
funkcja table()

table(zmienna1, zmienna2)

Liczebność wg jednej zmiennej – table(zmienna)
Ile elementów danej klasy

Łączenie grup

```
> table(eks$class)
```

0	aln	art	vul	aspl	bid			
60	4		50	1	4			
cal	uli	epi	fes	bro	koe	cor	mol	arr
11	4		15		11		36	
mon	car	phr	pol	poe	que	fag	que	rob-pet
1	3		1		50		12	
rha	pru	sal	sch	car	ste	med	tri	ger
12	1		2		13		9	
vac	pic							
12								

```
> |
```


łączenie grup

mam 21 `length(unique(eks$class))` grup danych

chcę mieć mniej:

- zmiana ręczna w excelu (rozważ co zajmie więcej czasu)

- `plyr::mapvalues(zmienna, from, to)`

mapvalues(wektor.do.zamiany, wzorce, zamienniki)

```
eks$typ<-plyr::mapvalues(eks$class,
```

```
unique(eks$class),
```

```
c('las','pozostałe','murawołąki','okrajki','okrajki','murawołąki','murawołąki','las','mura  
wołąki','las','las','okrajki','okrajki','okrajki','murawołąki','okrajki','pozostałe','murawołą  
ki','pozostałe','las','pozostałe'))
```

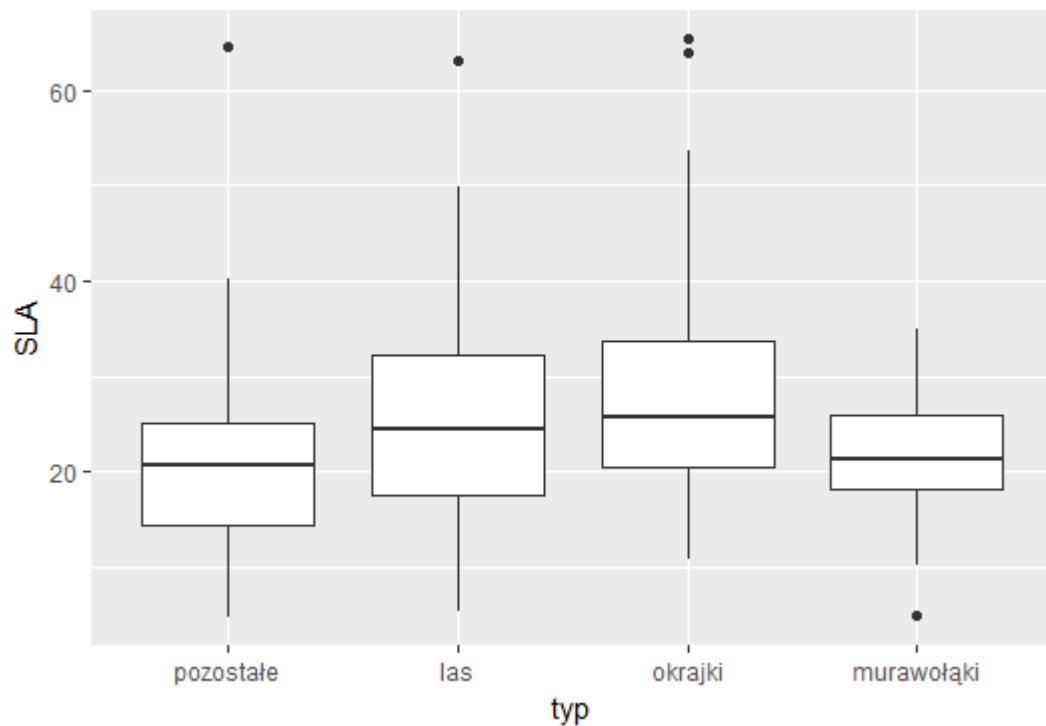
1. element – co ma być zamienione?
wektor ze wszystkimi pozycjami

2. element – wzorzec, lista pozycji z
których każdy po kolei będzie zastąpiony

3. element – zamienniki, na co będzie
zastąpiony każdy z kolei element ze
wzorca

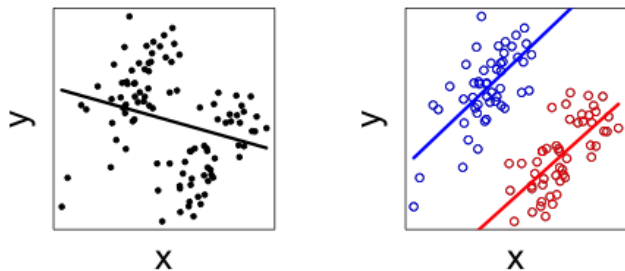
wpisane z ręki, odpowiada kolejności w unique(eks\$class)

```
ggplot(subset(eks,SLA<100), aes(x=typ,y=SLA))+geom_boxplot()
```

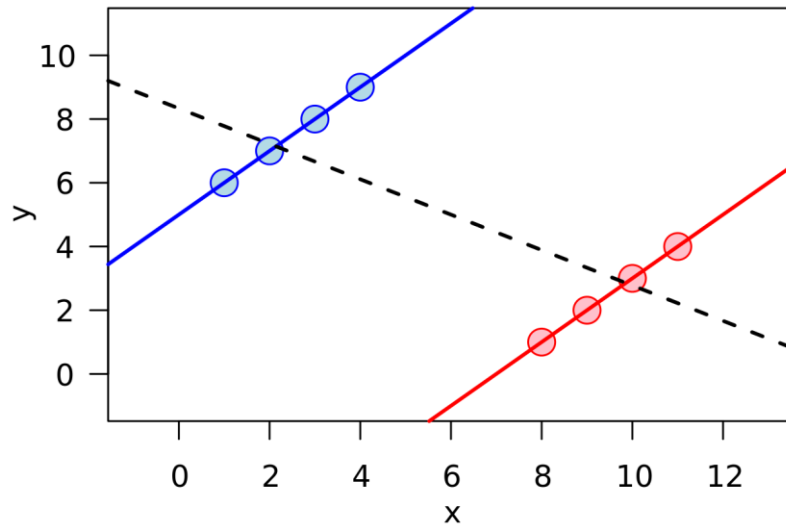


a co jeśli sprawdzę w ten sposób
seed_mass?

Wykresy panelowe - czy w grupach jest trend?

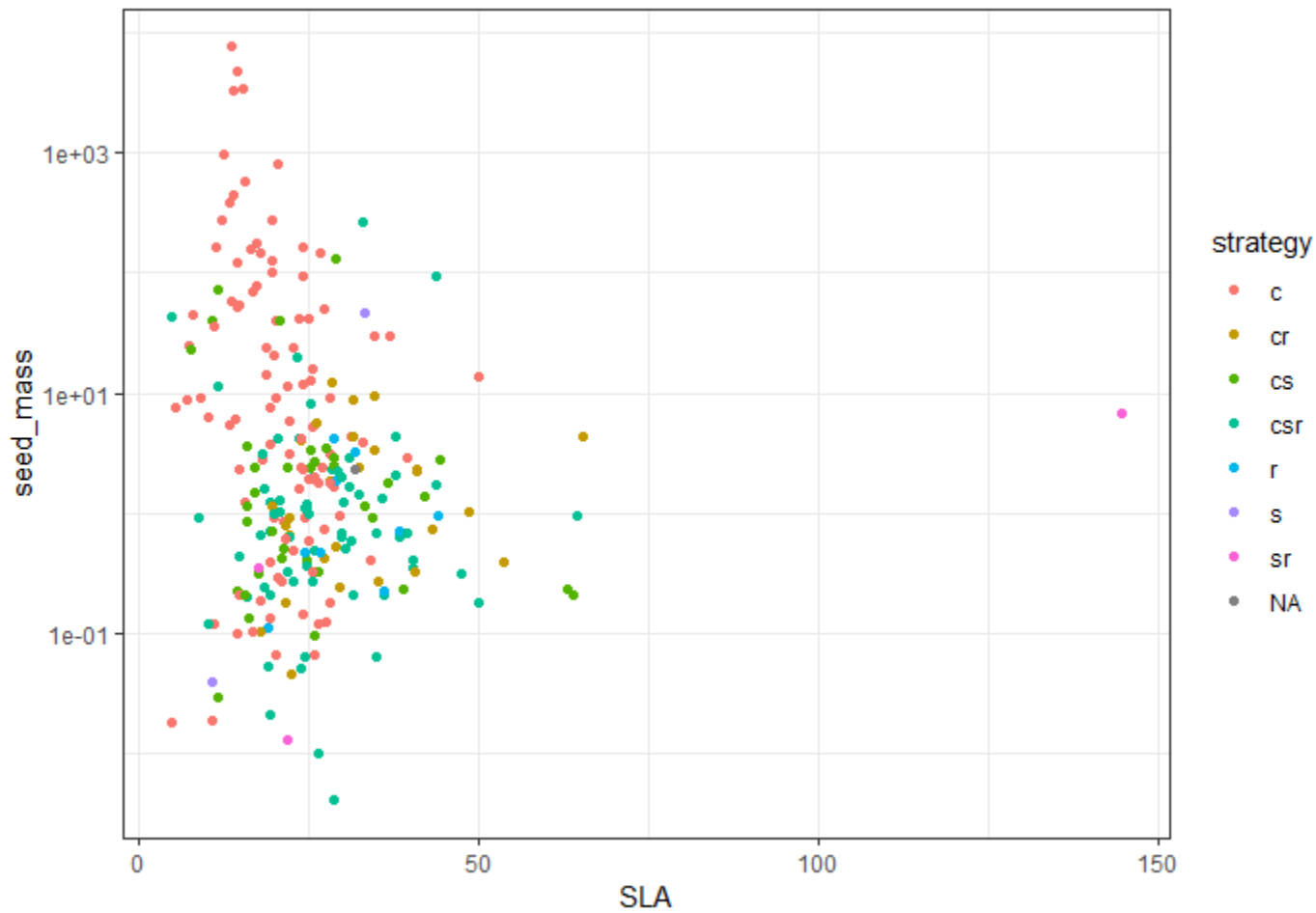


<https://normaldeviate.wordpress.com/2013/06/20/simpsons-paradox-explained/>

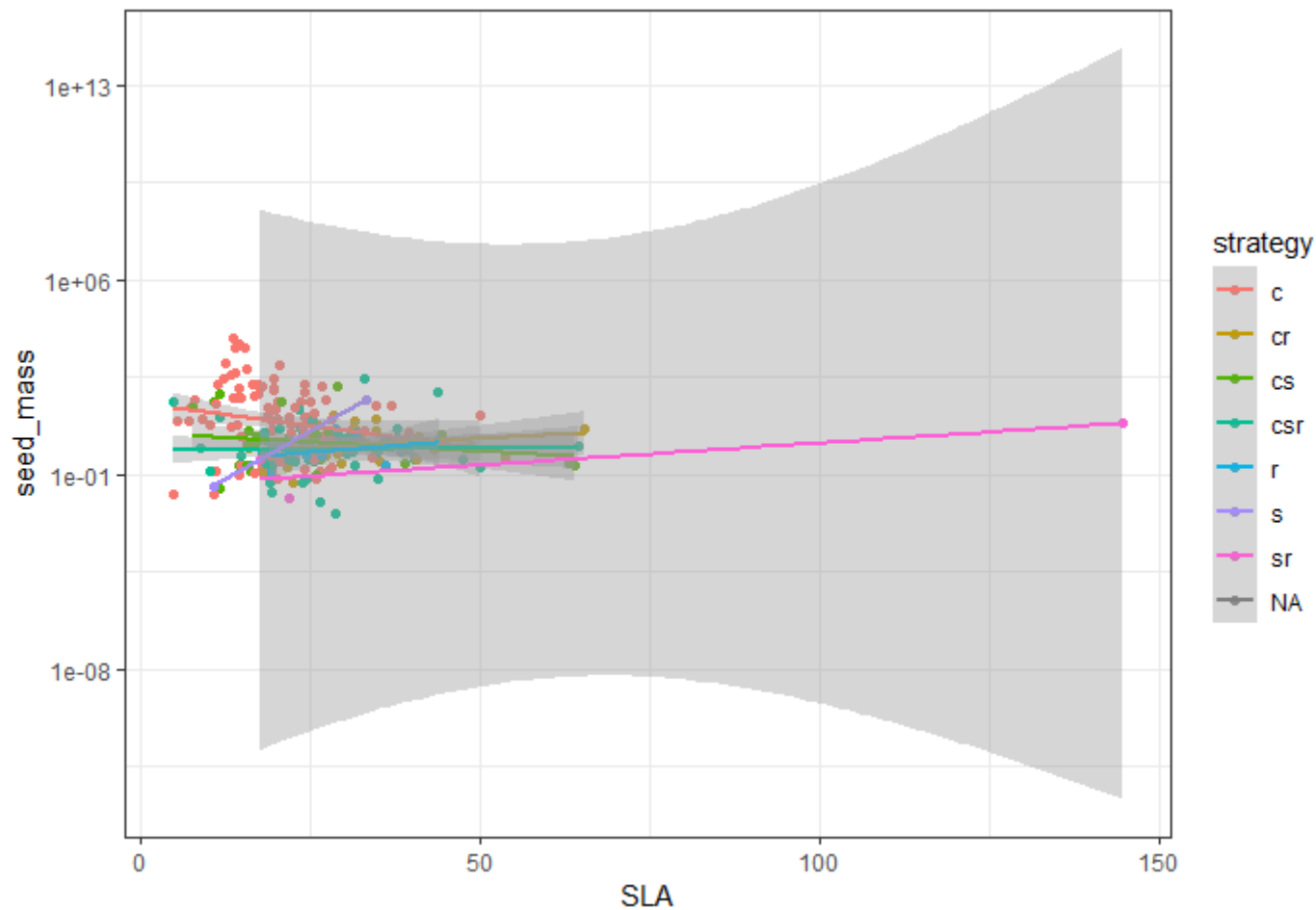


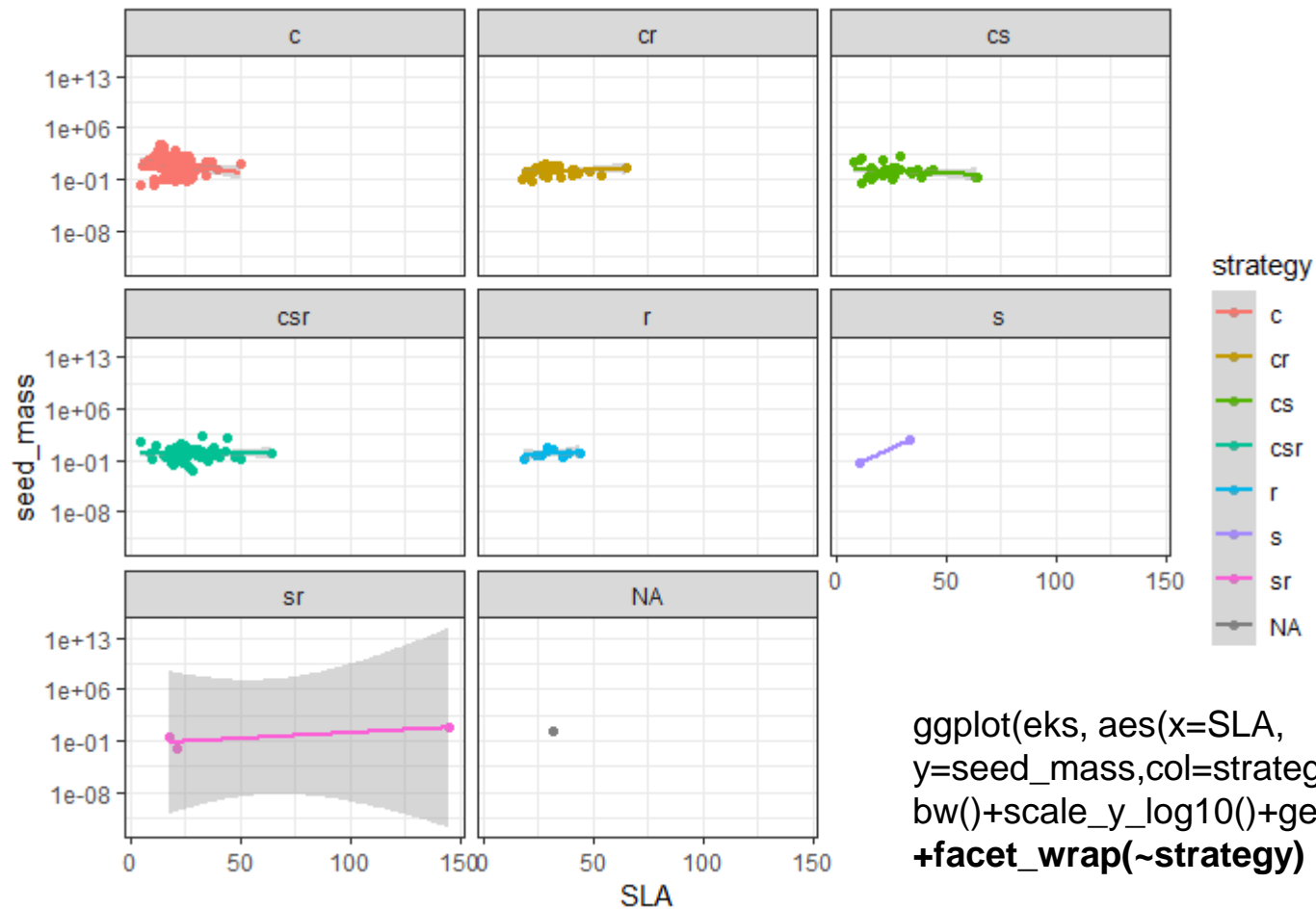
wikimedia.org

```
ggplot(eks, aes(x=SLA, y=seed_mass,col=strategy))+geom_point()+theme_bw()+scale_y_log10()
```



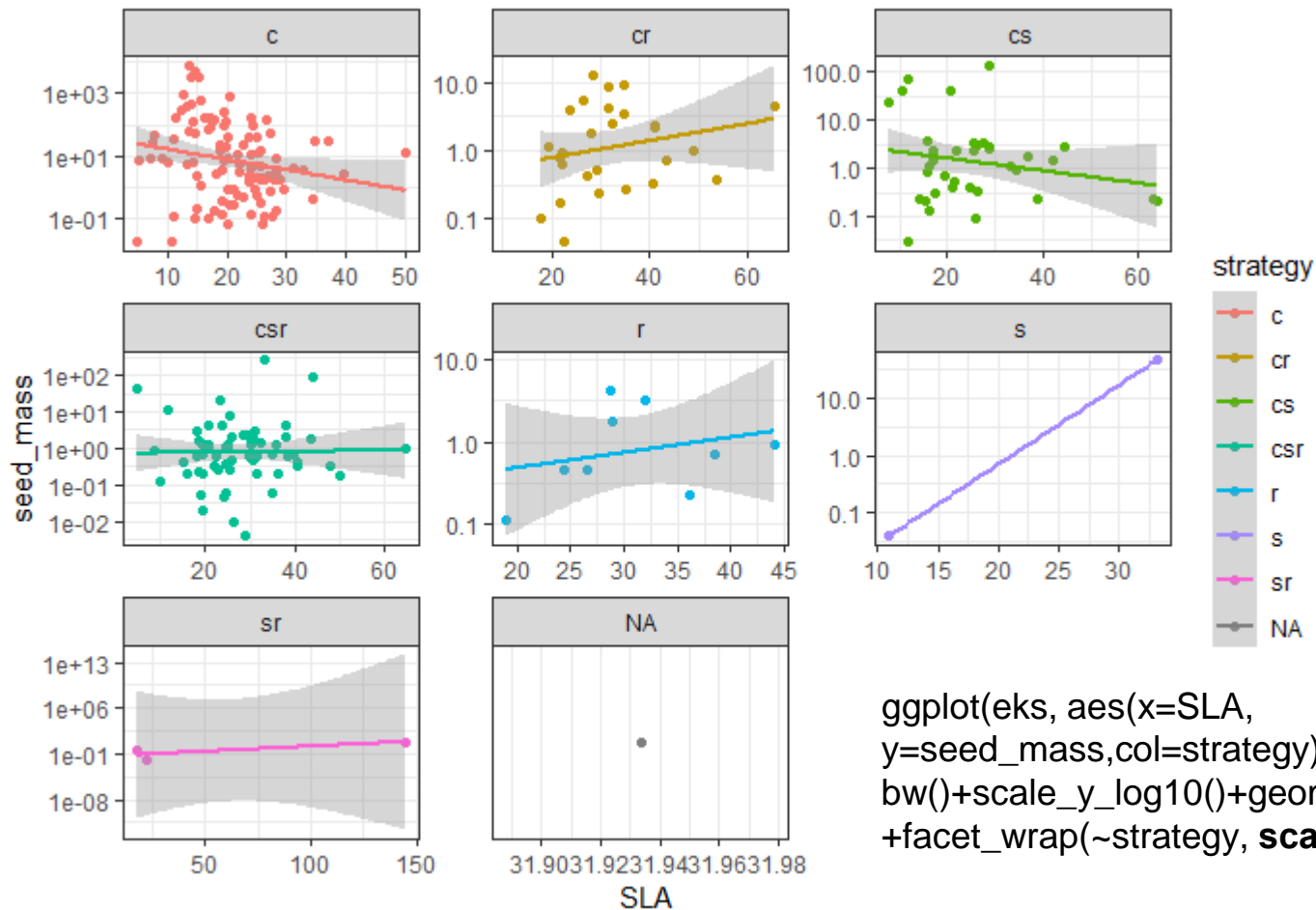
+geom_smooth(method='lm')





```
ggplot(eks, aes(x=SLA,
y=seed_mass,col=strategy))+geom_point()+theme_
bw()+scale_y_log10()+geom_smooth(method='lm')
+facet_wrap(~strategy)
```

facet_wrap, podział na grupy, panele,



czasem potrzebujemy
równych skal -
scales='fixed', albo tylko
wolnej osi y -
scales='free_y'

możemy też zdefiniować
ncol=3 albo nrow=3

```
ggplot(eks, aes(x=SLA,  
y=seed_mass,col=strategy))+geom_point()+theme_  
bw()+scale_y_log10()+geom_smooth(method='lm')  
+facet_wrap(~strategy, scales='free')
```

wielkość osi na wykresie, dopasowanie paneli

Wykres średnia + SE

```
eks%>%filter(!is.na(SLA))%>%group_by(typ)%>%summarise(srednia=mean(SLA))
```

```
# A tibble: 4 x 2
```

	typ	srednia
	<fctr>	<dbl>
1	pozostałe	20.91537
2	las	25.80841
3	okrajki	29.56142
4	murawołąki	21.83004

SE a SD

$SE = SD / \sqrt{n}$

```
se<-function(x) sd(x)/sqrt(length(x))
```

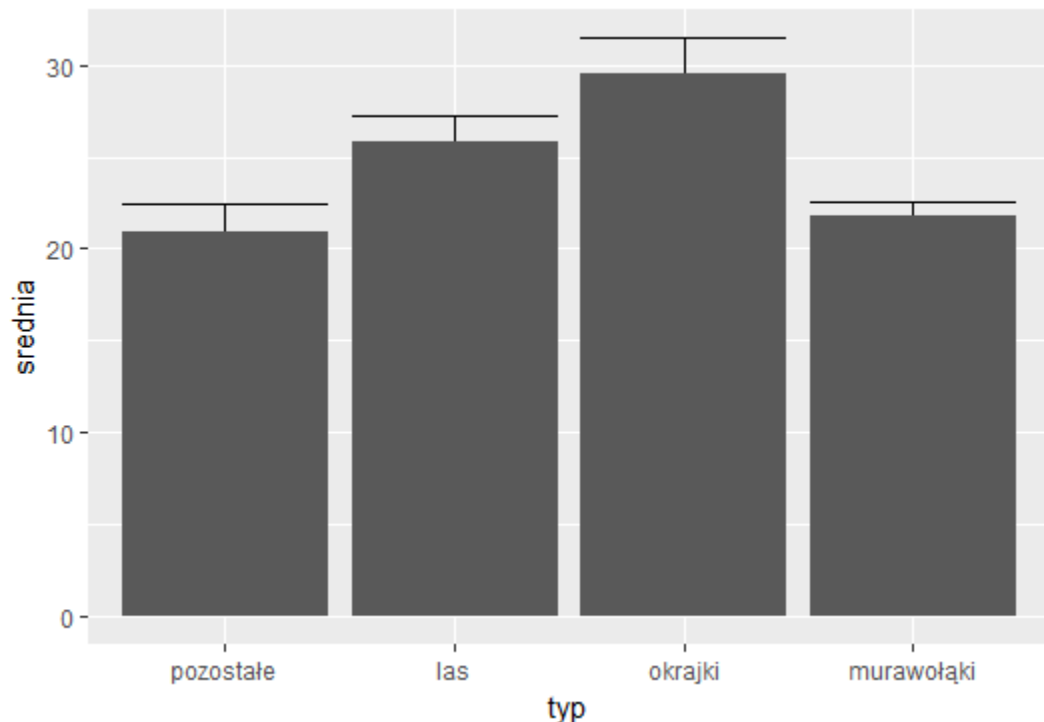
```
se<-function(x) sd(x,na.rm=T)/sqrt(length(x[!is.na(x)]))
```

```
eks%>%filter(!is.na(SLA))%>%group_by(typ)%>%summarise(srednia=mean(SLA),se=se(SLA))
```

```
# A tibble: 4 x 3
```

	typ	srednia	se
	<fctr>	<dbl>	<dbl>
1	pozostałe	20.91537	1.5595037
2	las	25.80841	1.4097452
3	okrajki	29.56142	1.9128191
4	murawołąki	21.83004	0.7576967

```
df<-  
eks%>%filter(!is.na(SLA))%>%group_by(typ)%>%summarise(srednia=mean(SLA),  
se=se(SLA))  
ggplot(df, aes(x=typ,y=srednia)) + geom_errorbar(aes(ymin=srednia-se,  
ymax=srednia+se)) + geom_col()
```



trochę szaro...

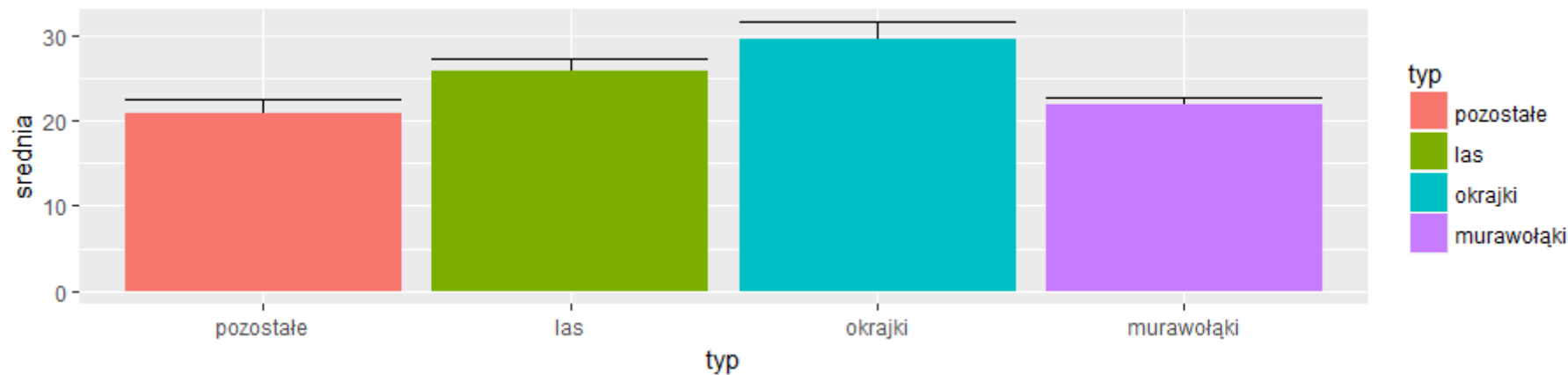
fill = kolor wypełnienia

col = kolor obramowania

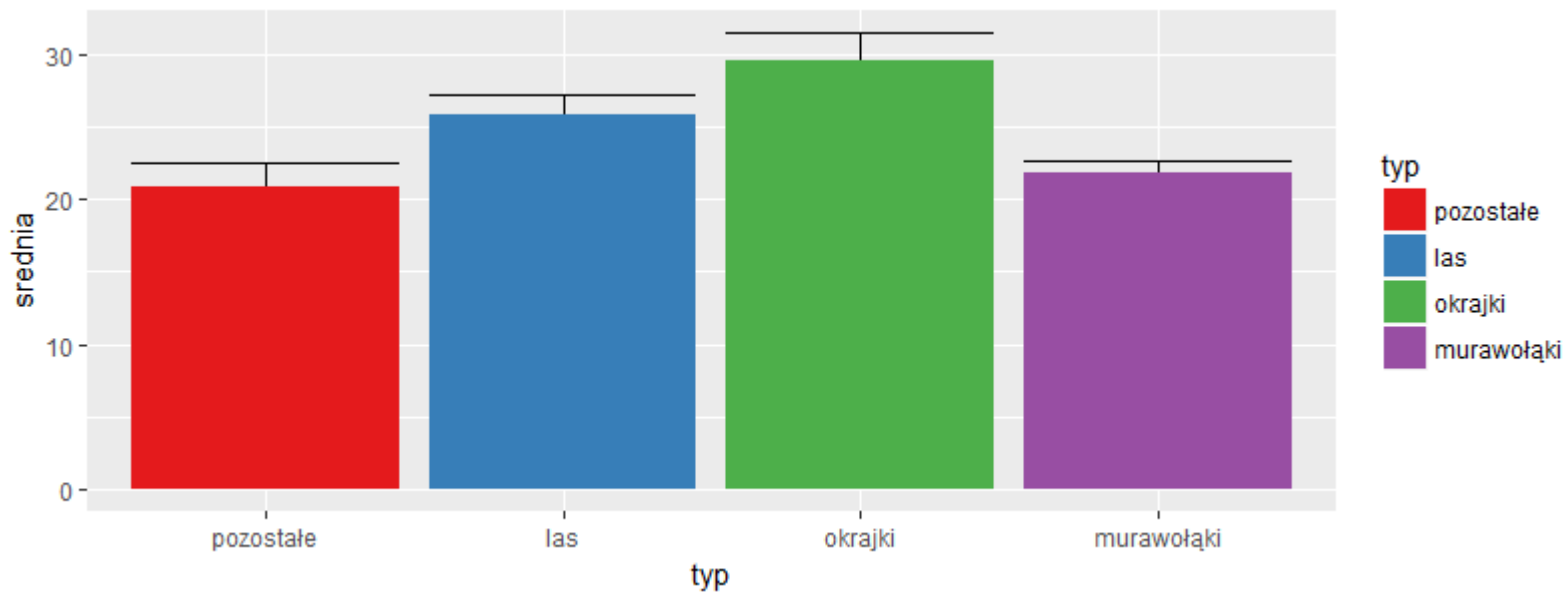
domyślnie col nie jest czarne

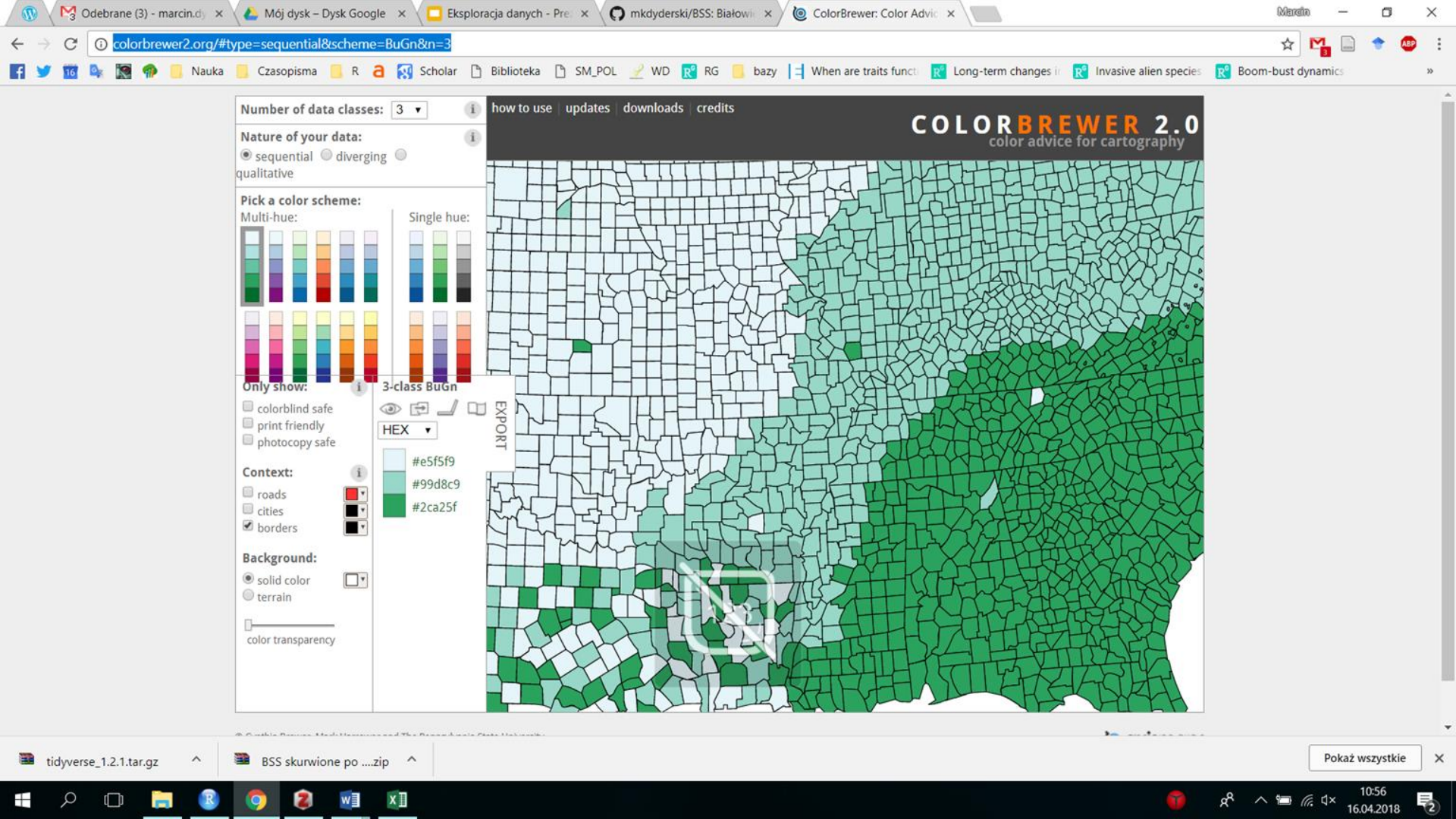
skale kolorów

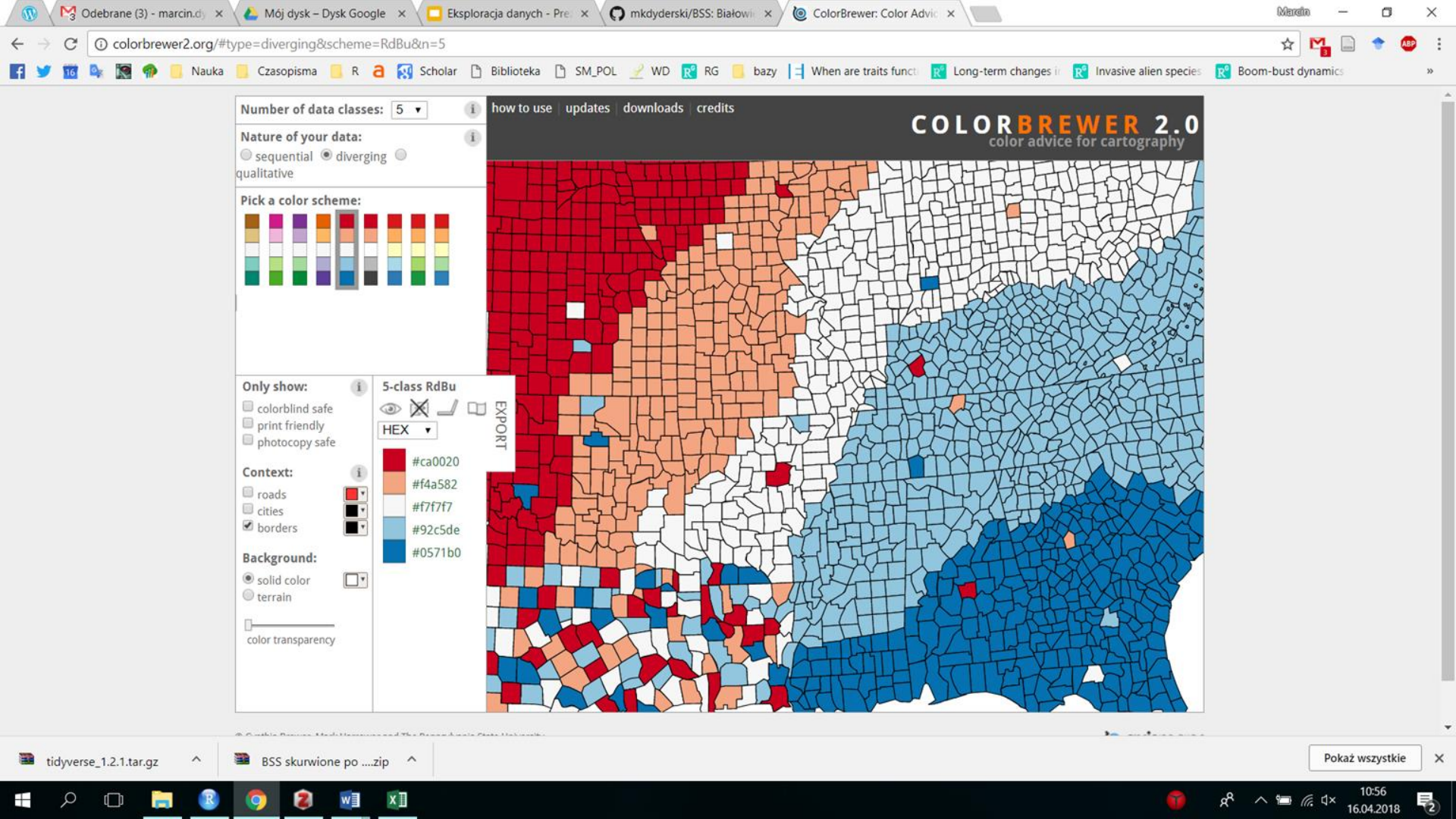
```
ggplot(df, aes(x=typ,y=srednia,fill=typ))+geom_errorbar(aes(ymin=srednia-se,ymax=srednia+se)) +geom_col()
```

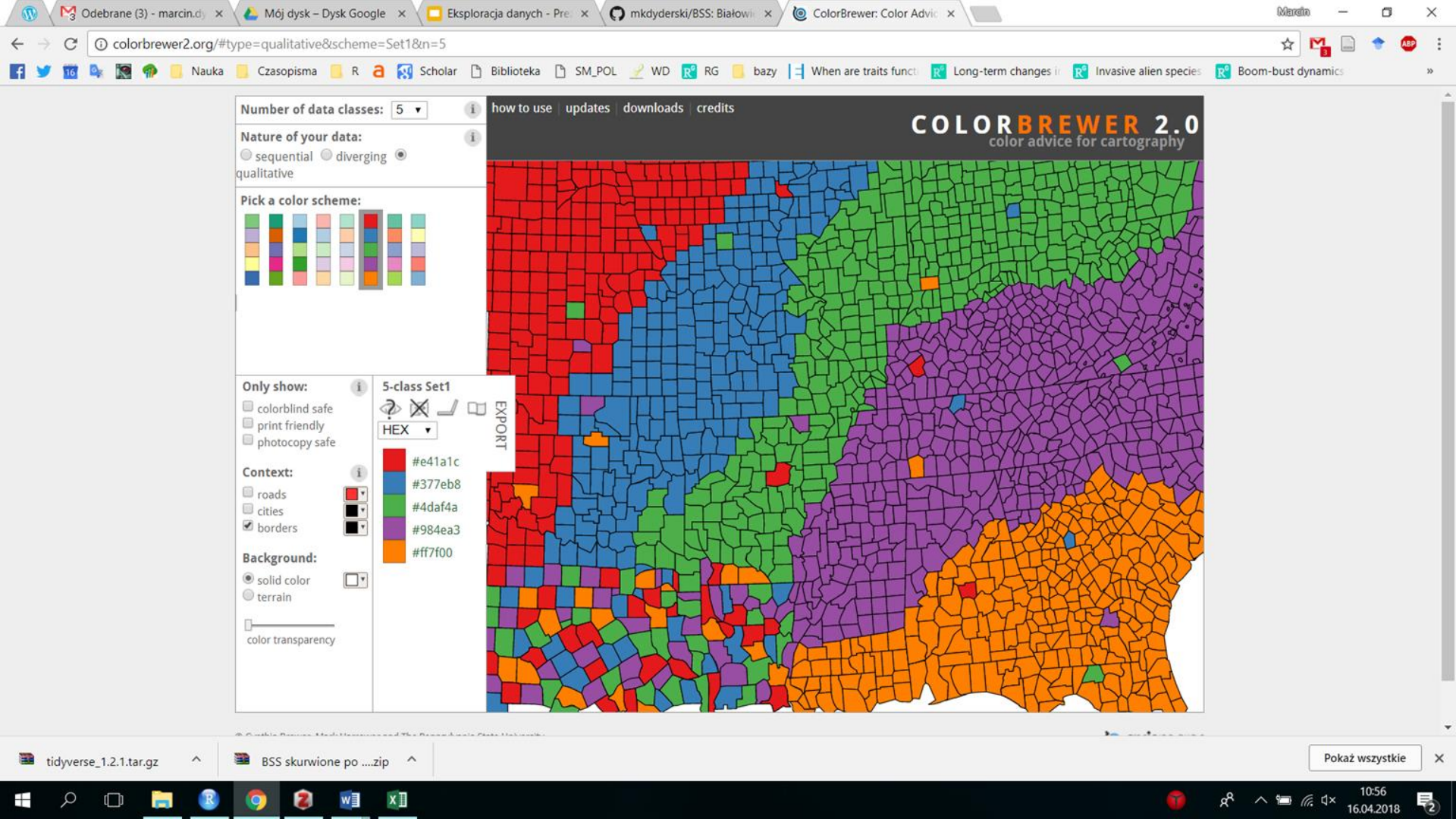


```
ggplot(df, aes(x=typ,y=srednia,fill=typ))  
+geom_errorbar(aes(ymin=srednia-se,ymax=srednia+se))  
+geom_col()+scale_fill_brewer(palette='Set1')
```









Analiza danych zagregowanych

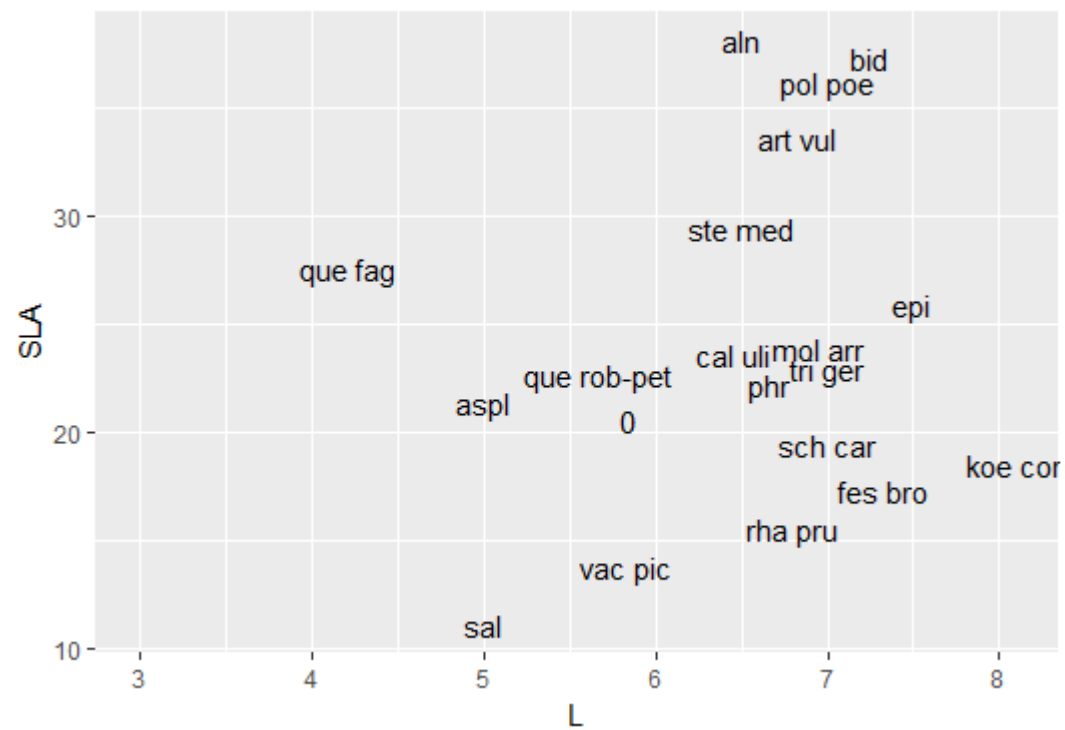
```
eks2<-eks%>%filter(!is.na(class))%>%group_by(class)  
%>%summarise_all(mean,na.rm=T)
```

#grupujemy aby dla każdej klasy mieć średnie wartości

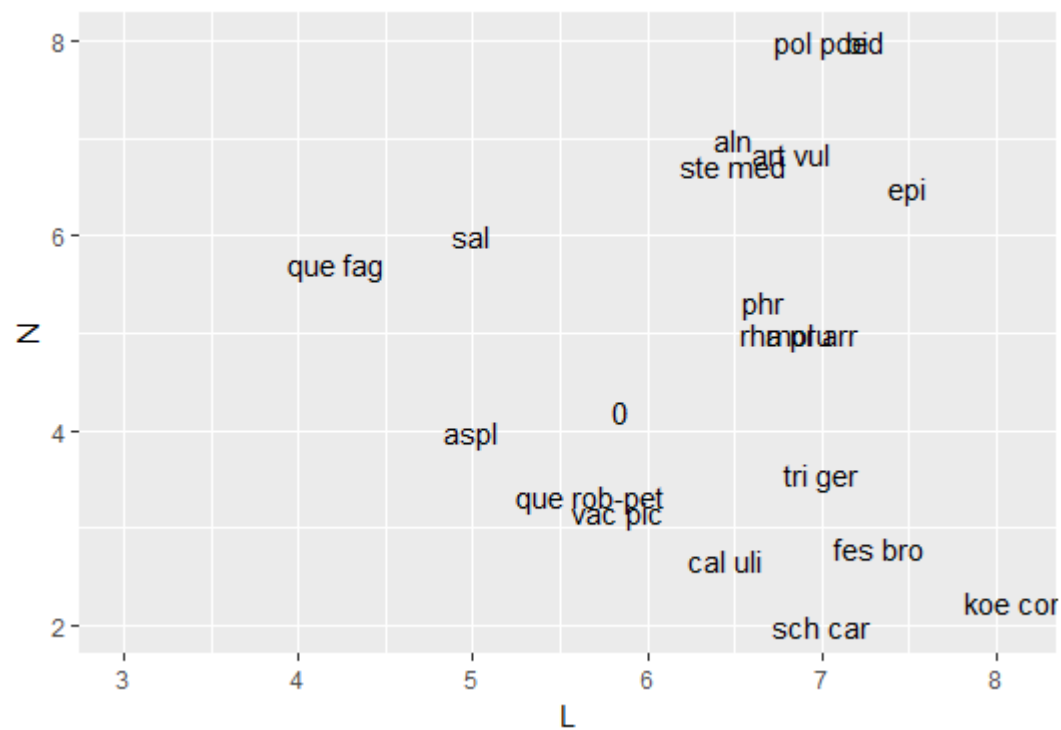
```
ggplot(eks2, aes(x=L,y=SLA))+geom_text(aes(label=class))
```

#geom_text – rysuje zamiast punktów tekst

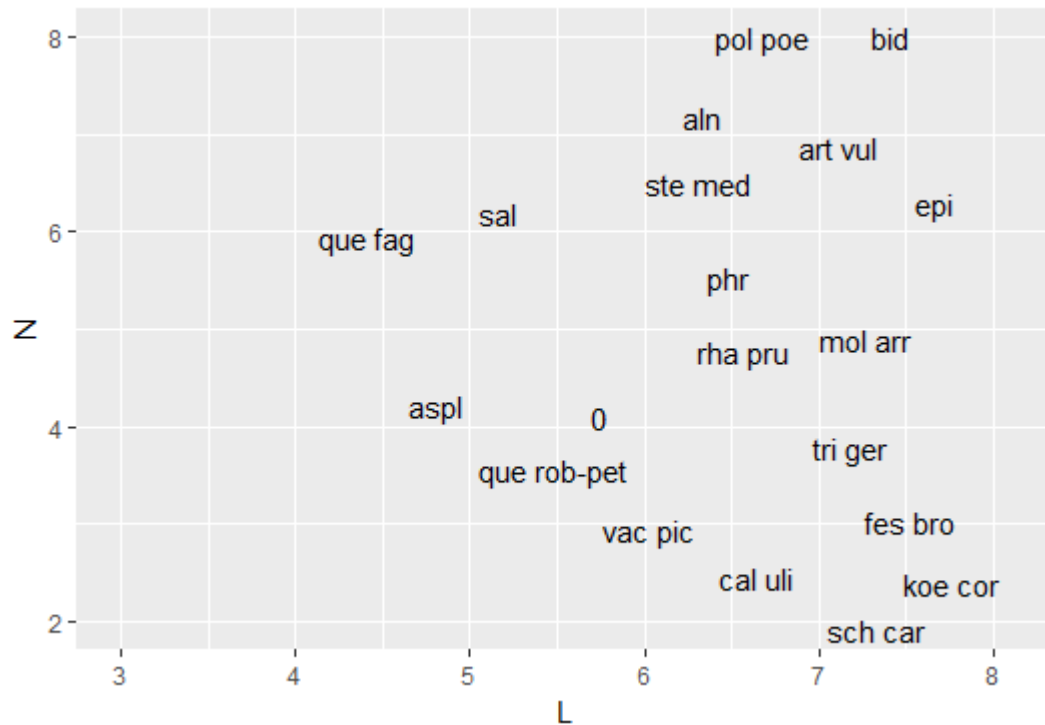
#etykiety na wykresie,



brzydkie?



```
library(ggrepel)  
ggplot(eks2, aes(x=L,y=N))+geom_text_repel(aes(label=class))
```



ggrepel porządkuje etykiety
jak jest ich za dużo to
dorabia strzałki

Zastosowania - interakcja z ggplot

włączanie “rurek” w ggplot():

```
ggplot(subset(eks,SLA<100), aes(x=strategy,y=SLA))+geom_violin()
```

```
ggplot(eks%>%filter(SLA<100), aes(x=strategy,y=SLA))+geom_violin()
```

Poza ggplot

ggplot pracuje na data.frame

wszystko co da się przekształcić w data.frame da się pokazać ggplotem

przekształcenia są często opakowane w gotowe funkcje:

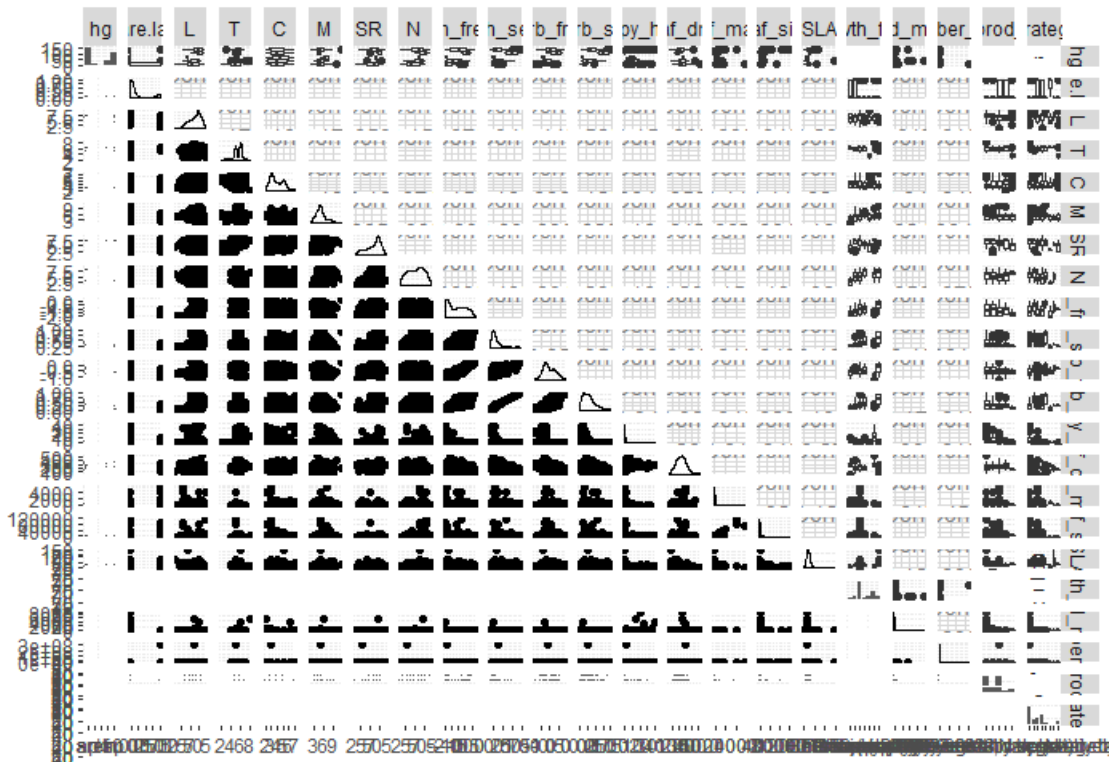
pakiet ggfortify - <https://github.com/sinhrks/ggfortify>

pakiet GGally - <https://ggobi.github.io/ggally/>

do eksploracji - ggpairs()

wymagająca obliczeniowo, istnieje też bazowa funkcja pairs()

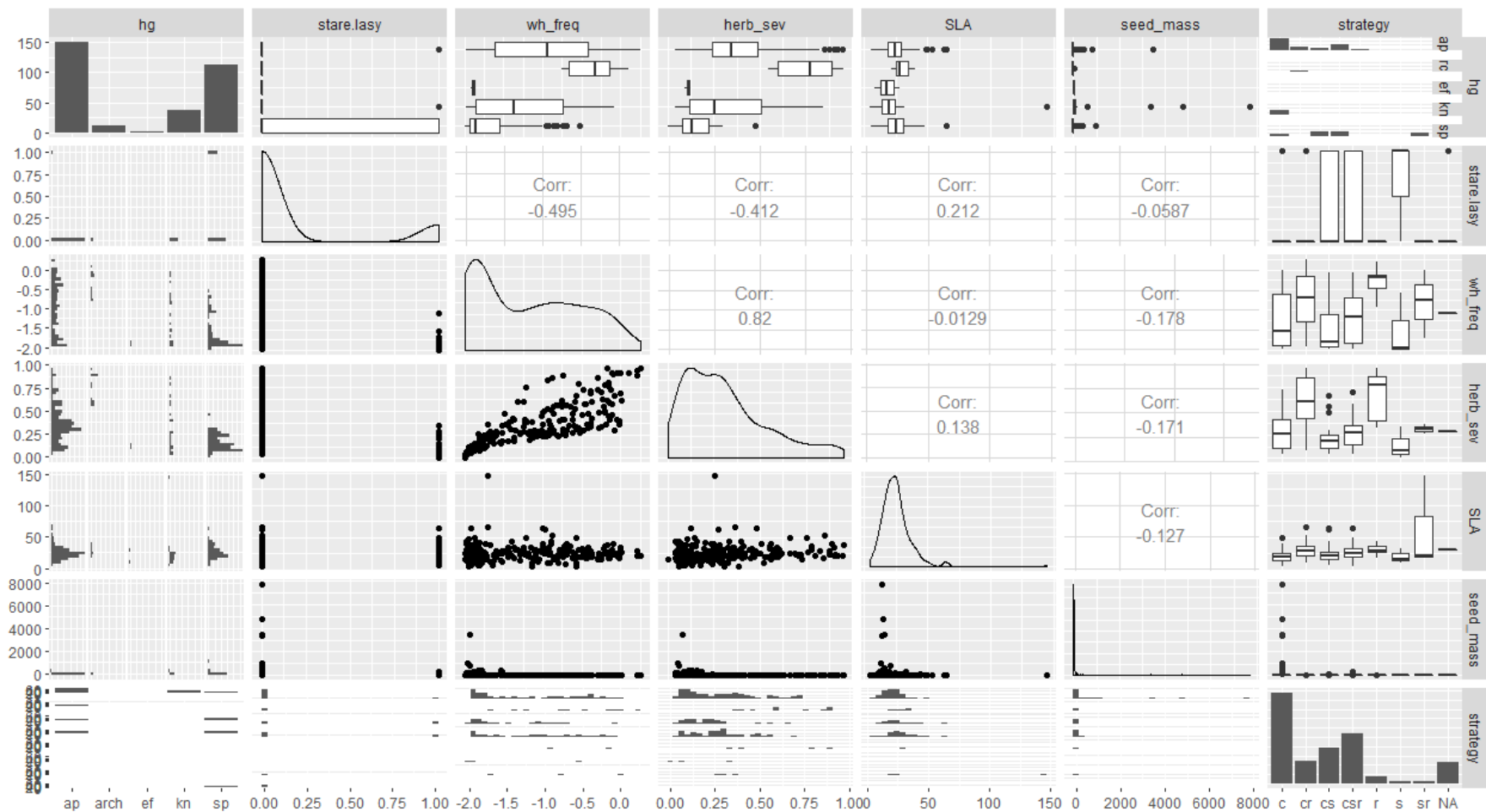

```
library(GGally)
ggpairs(eks[,c(2,4:24)])
```

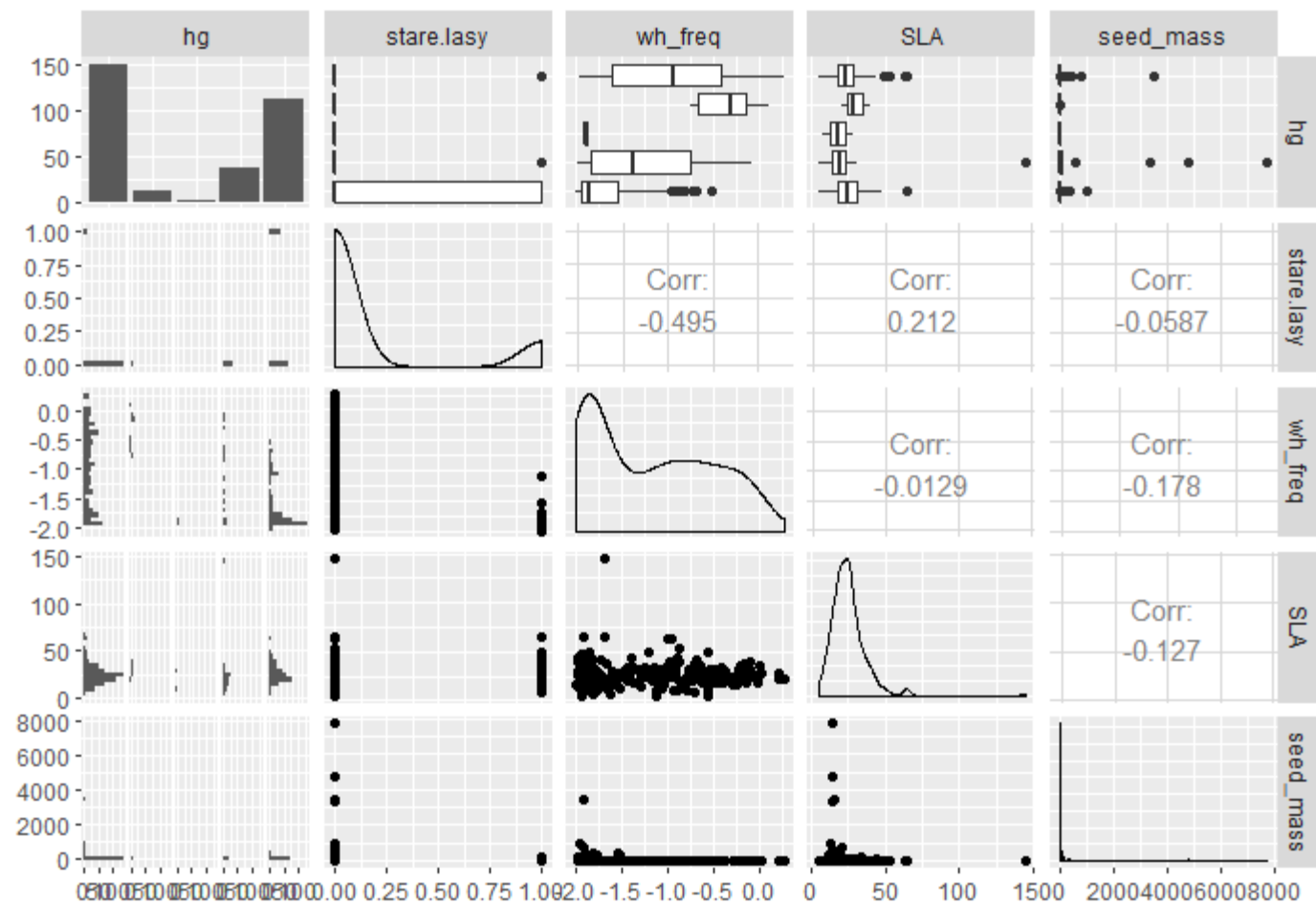


widać niewiele
liczy się długo

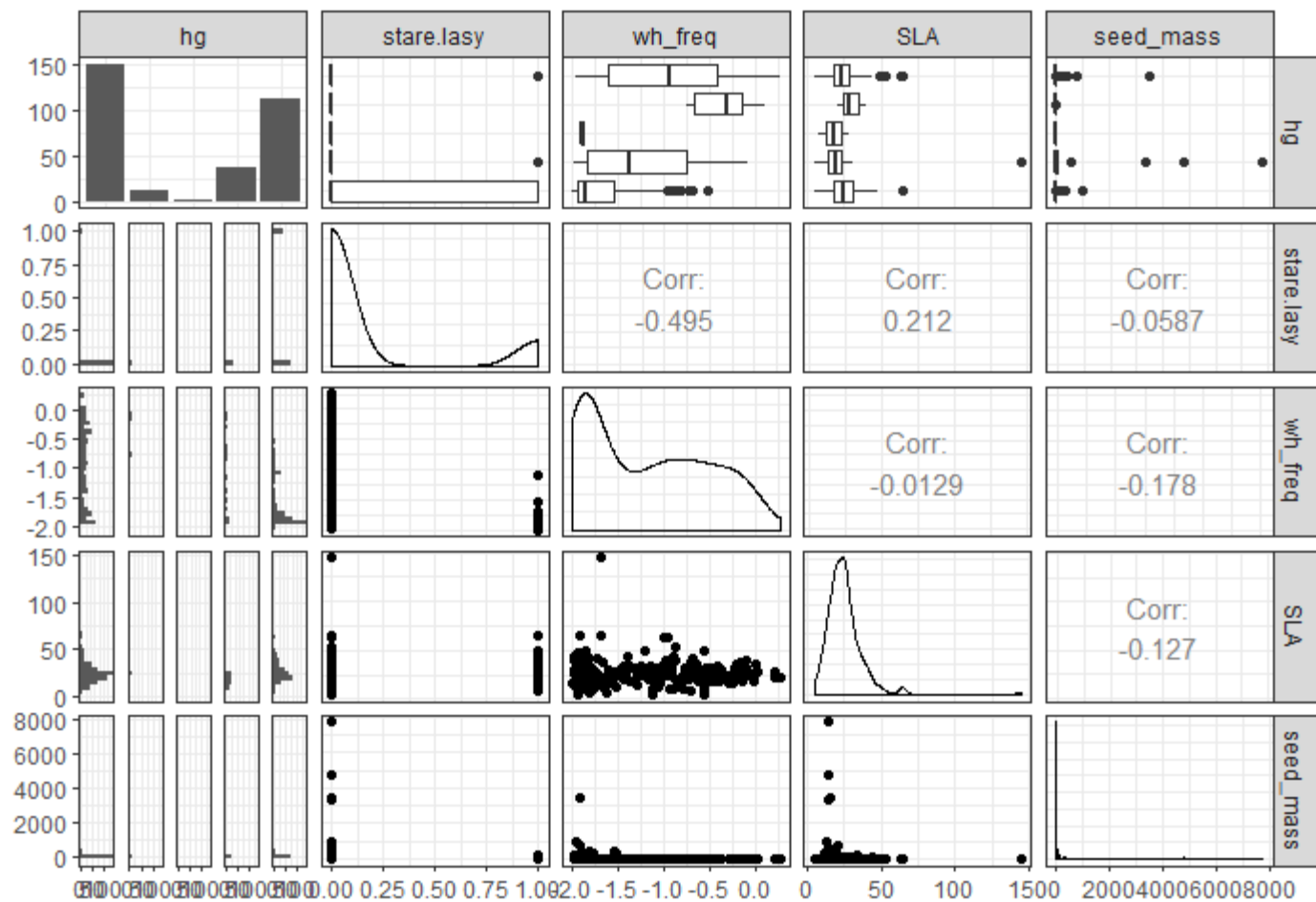
zobaczmy na kilku
kolumnach

```
library(GGally); ggpairs(eks[,c(2,4,11,14,19,21,24)])
```





ggpairs(eks[,c(2,4,11,19,21)])



`ggpairs(eks[,c(2,4,11,19,21)])+theme_bw()`



BIAŁOWIESKA SZKOŁA STATYSTYKI