# Podstawy języka R - zadania (4)

Tomasz Owczarek, Mateusz Naramski

2024/2025, semestr letni

## Zbiór danych movies.csv

Pod zmienną `mov` wczytaj dane z pliku *movies.csv*. Zamień kolumny `genre` i `year` na zmienne czynnikowe. Po wykonaniu polecenia `str()` efekt powinien być następujący:

```
## 'data.frame':    1565 obs. of  11 variables:
## $ title : chr "Glitter" "Soul Survivors" "Megiddo: The Omega Code 2" "On the
##    Line" ...
## $ genre : Factor w/ 4 levels "Action","Adventure",..: 4 4 1 3 1 1 3 3 3 3 ...
## $ director : chr "Vondie Curtis-Hall" "Stephen Carpenter" "Brian
##    Trenchard-Smith" "Eric Bross" ...
## $ year : Factor w/ 16 levels "2001","2002",..: 1 1 1 1 1 1 1 1 1 1 ...
## $ duration : int 104 84 104 85 85 116 87 93 86 87 ...
## $ gross : int 4273372 3100650 5974653 4356743 12610731 32616869 14249005
##    10097096 23978402 112950721 ...
## $ budget : int 22000000 14000000 22000000 16000000 11000000 72000000 15000000
##    4000000 11000000 72000000 ...
## $ cast_facebook_likes: int 1854 417 4221 2446 3050 14780 2689 955 3850 3287 ...
## $ votes : int 19412 7277 2253 3662 38985 34435 39788 5612 10966 33884 ...
## $ reviews : int 374 245 129 125 878 455 716 65 162 170 ...
## $ rating : num 2.1 3.9 4.1 4.1 4.4 4.5 4.5 4.6 4.6 4.6 ...
```

### Funkcja cut

**51.** Funkcja `cut` tworzy zmienną kategoryczną z poziomami odpowiadającymi przedziałom, w których znajdują się wartości wektora liczbowego. Wymaga podania dwóch argumentów: $x$ (wektor liczbowy) oraz *breaks* (liczba przedziałów lub wektor określający granice przedziałów). Przykład działania:

```r
set.seed(10) # seed dla jednakowych wynikow
x <- rnorm(100, mean = 5, sd = 2) # 100 liczb z rozkladu normalnego
x.interval <- cut(x, breaks = c(0, 2.5, 5, 7.5, 10)) # podzial na 4 ustalone przedzialy
table(x.interval) # sprawdzenie liczebnosci przedzialow
```

```
## x.interval
##  (0,2.5]  (2.5,5]  (5,7.5] (7.5,10]
##       14       42       40        4
```

Utwórz nową kolumnę w ramce `mov` o nazwie *duration.interval*, która będzie zawierała rezultat funkcji `cut` przydzielający każdemu filmowi jeden z następujących przedziałów czasu trwania: 60-90, 90-120 i 120-300. Po tym działaniu wykonanie następującego kodu:

```r
table(mov$duration.interval)
```

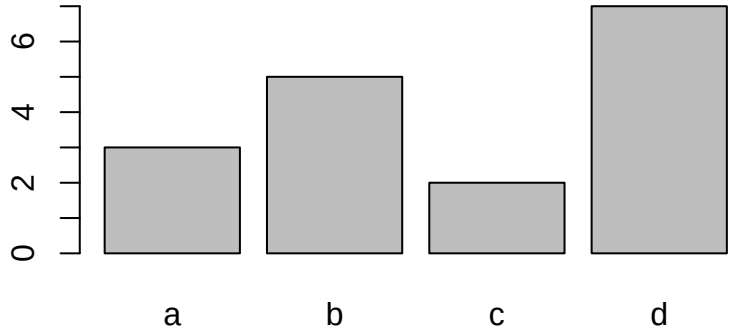
powinno zwrócić taki rezultat:

```
##
##   (60,90]  (90,120] (120,300]
##       227      1036       302
```

### Funkcja barplot

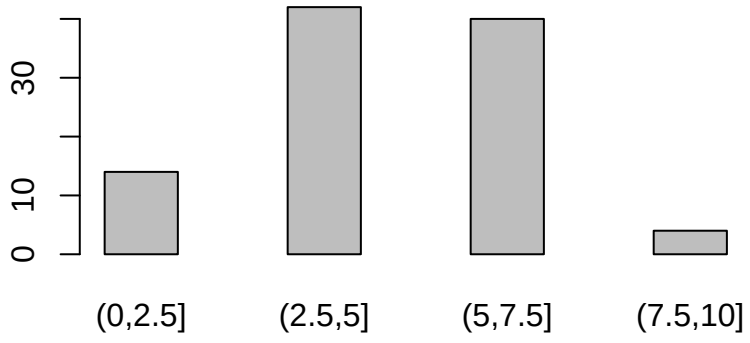
**52.** Funkcja `barplot` tworzy wykres kolumnowy. Potrzebuje wektora liczbowego (opcjonalnie można też podać etykiety kolumn). Przykład działania:

```r
barplot(c(3, 5, 2, 7), names.arg = letters[1:4]) # letters - wektor z literami alfabetu
```

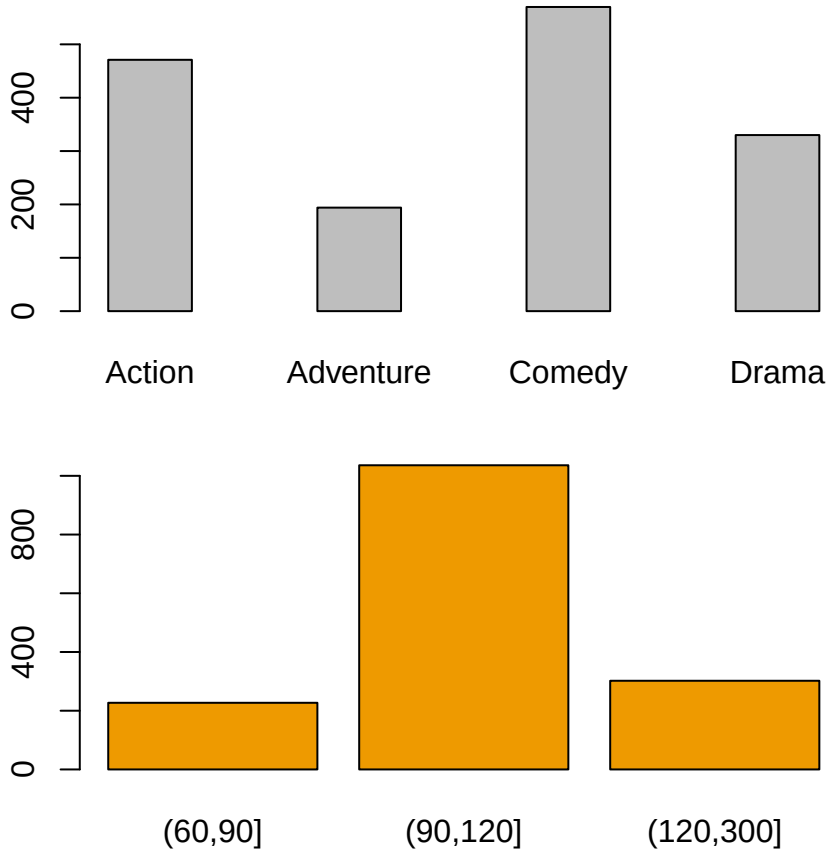



Idealnym wejściem do tej funkcji jest wynik funkcji `table` (nie trzeba wtedy podawać etykiet kolumn, bo te są brane z nazw elementów zwracanych przez funkcję `table`). Przykład działania na wektorze *x.interval* z poprzedniego zadania:

```r
barplot(table(x.interval), space = 1.5) # space pozwala zwezyc kolumny
```



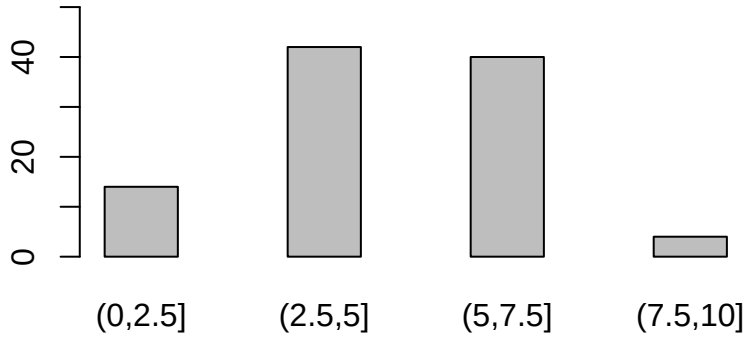

a) Utwórz wykres kolumnowy przedstawiający liczbę filmów poszczególnych gatunków (zwęż kolumny za pomocą argumentu *space*).
b) Utwórz wykres kolumnowy przedstawiający liczbę filmów trwających do 90 minut, między 90 i 120 minut oraz dłuższych (skorzystaj z kolumny *duration.interval* z poprzedniego zadania, kolor kolumn ustaw na *orange2*).



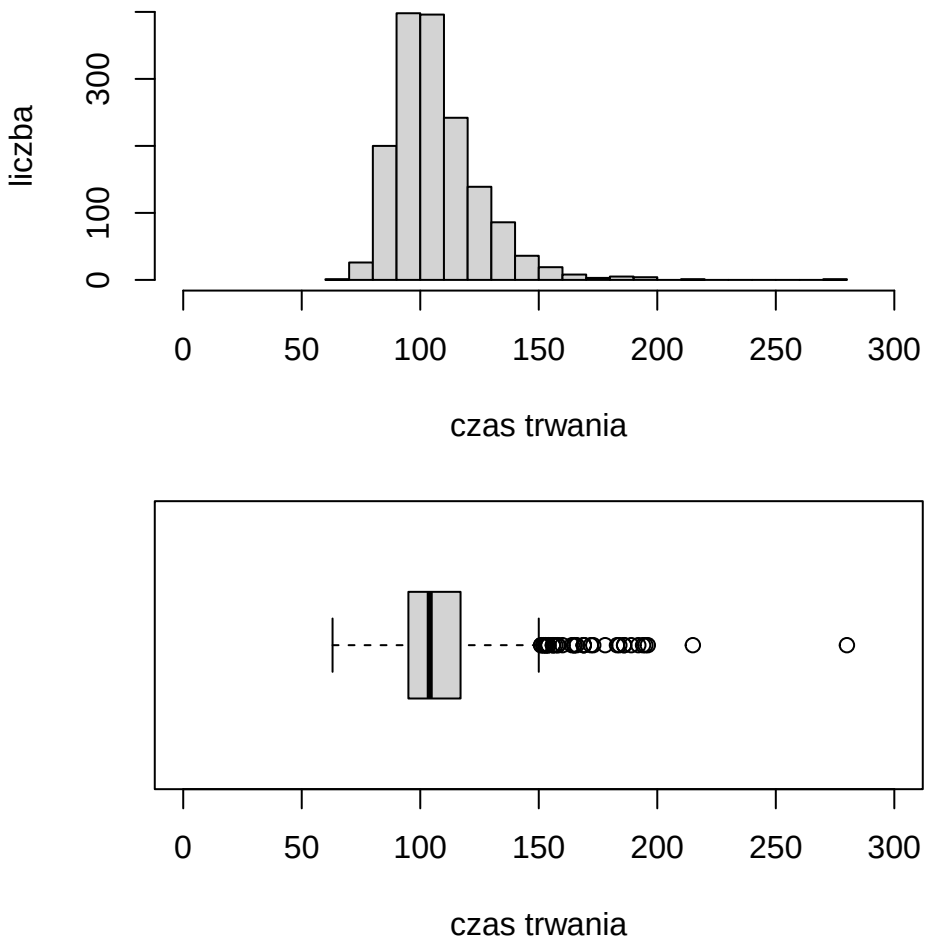

**Parametry wykresów: xlim, ylim**

**53.** Wykresy w R mają granice osi dobierane automatycznie na podstawie danych. Można to zmienić za pomocą argumentów *xlim* i *ylim*, które wymagają podania dwuwartościowego wektora, wyznaczającego dolną i górną granicę osi, np. poniższy kod zmieni granice osi **Y** na 0-50 (*domyślna granica była inna - zobacz wykres z zadania 52*):

```
barplot(table(x.interval), space = 1.5, ylim = c(0, 50)) # ylim - zakres osi
```



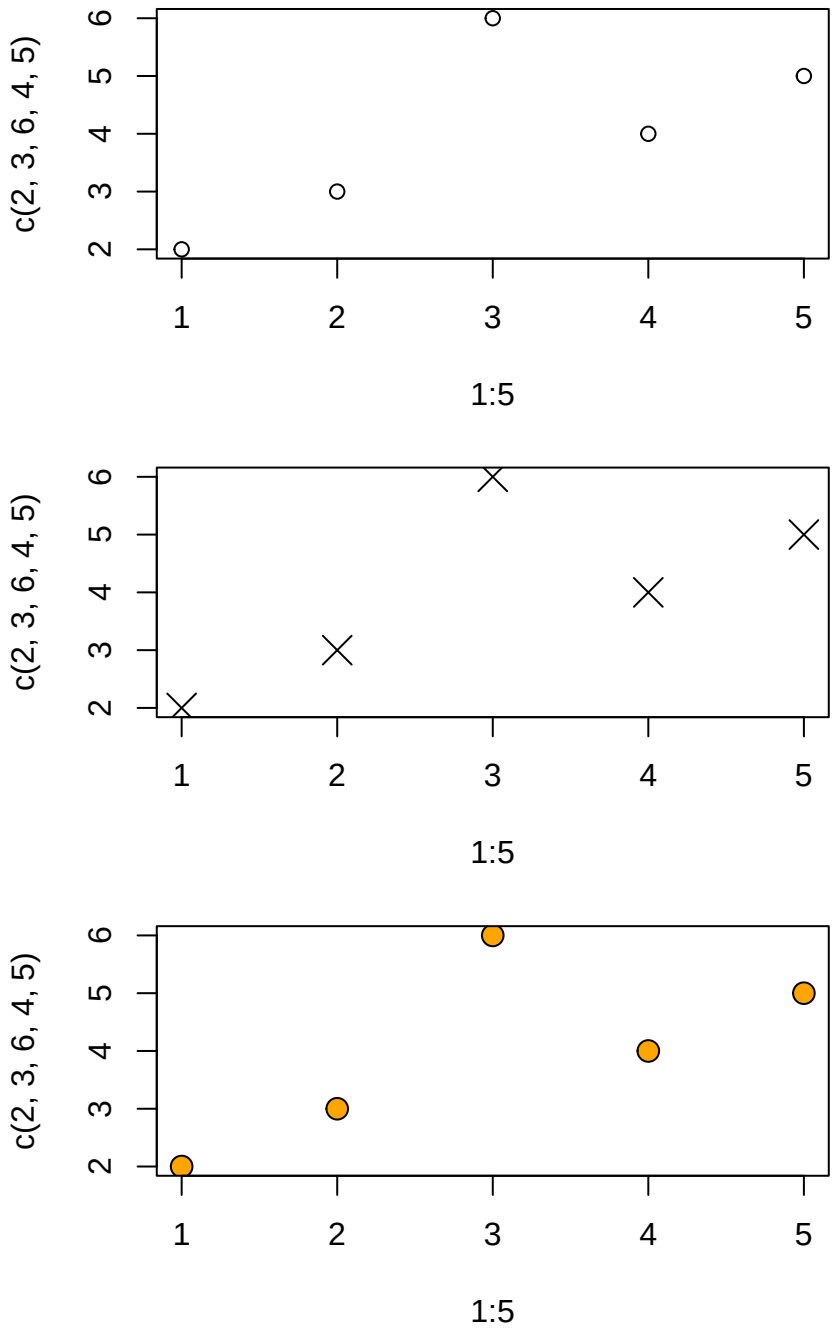

a) Utwórz histogram czasu trwania filmów (ustaw *breaks* na 20 a granice osi **X** na 0-300, zmień opisy osi zgodnie ze wzorem, usuń tytuł wykresu)
b) Utwórz poziomy wykres pudełkowy czasu trwania filmów (ustaw granice osi **Y** na 0-300, zmień opisy osi zgodnie ze wzorem)



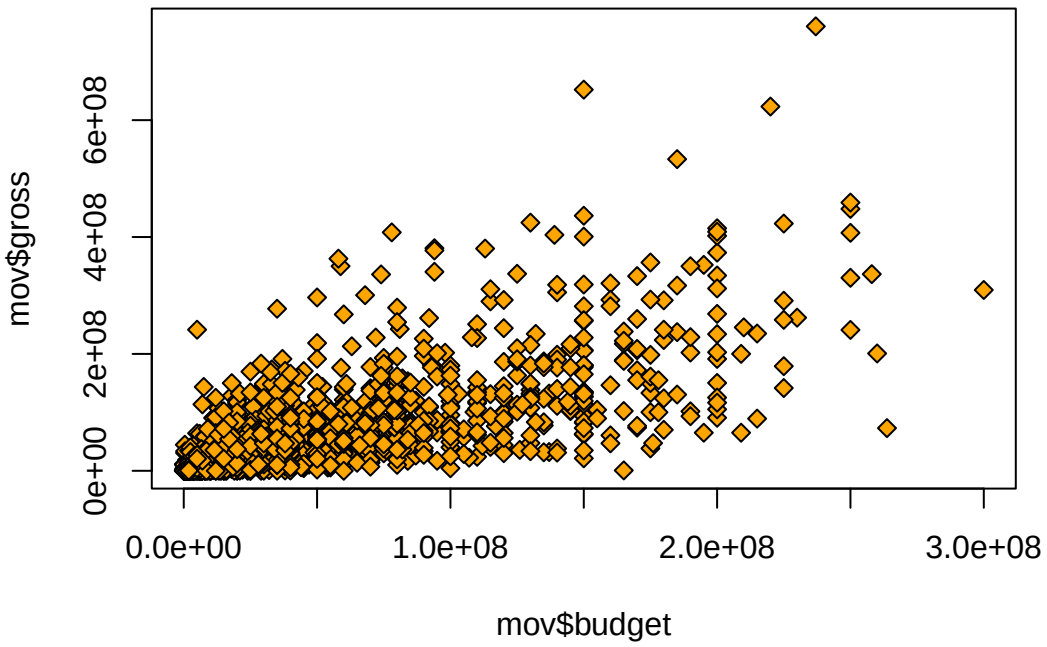

**Kształty (pch) i rozmiar (cex) punktów**

**54.** Punkty na wykresach punktowych mogą przyjmowac różne kształty (argument *pch*) i wielkość (argument *cex*). Domyślnym kształtem jest puste kółko, które ma numer 1, numery innych kształtów można sprawdzić np. tutaj. Kształty o numerach 21-25 mają dodatkowy argument *bg*, który pozwala na zmianę koloru wypełnienia. Przykłady:

```
plot(1:5, c(2, 3, 6, 4, 5)) # bez zmian
plot(1:5, c(2, 3, 6, 4, 5), pch = 4, cex = 2) # zmiana ksztaltu i wielkosci
plot(1:5, c(2, 3, 6, 4, 5), pch = 21, cex = 1.5, bg = "orange") # ksztalt, wielkosc,
                                                                # wypelnienie
```




Utwórz wykres punktowy przedstawiający budżet i przychód filmu. Kształt punktu to romb z wypełnieniem (23), wybierz samodzielnie ciekawy kolor wypełnienia.

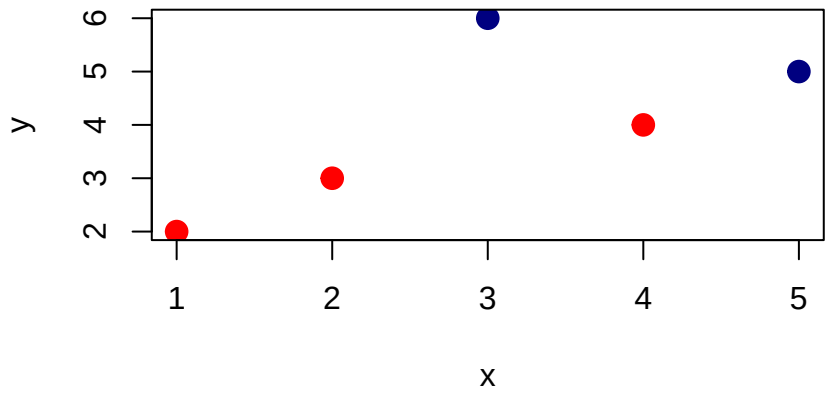



**Kolorowanie wg zmiennej kategorycznej i legenda wykresu**

**55.** Zmienną kategoryczną (*factor*) można użyć do kolorowania punktów. Domyślnie `R` dobiera własne kolory, ale można tym sterować za pomocą następującej konstrukcji: `wektor_kolorow[factor]`, np.

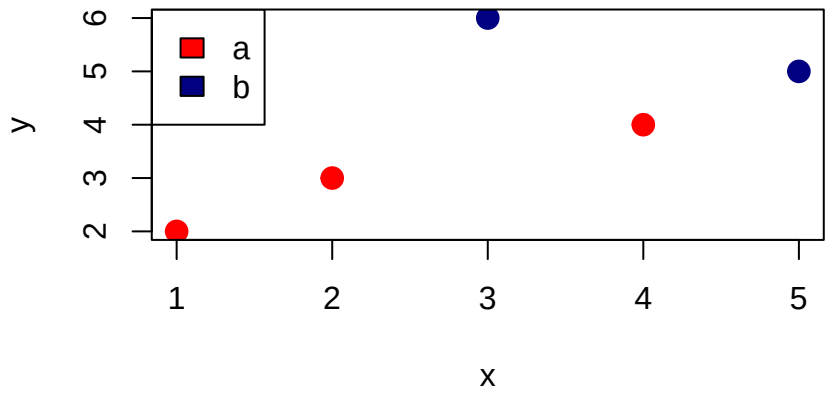
```
x <- 1:5
y <- c(2, 3, 6, 4, 5)
type <- factor(c("a", "a", "b", "a", "b"))

plot(x, y, pch = 19, col = c("red", "navy")[type], cex = 1.5) # kolory wg type
```



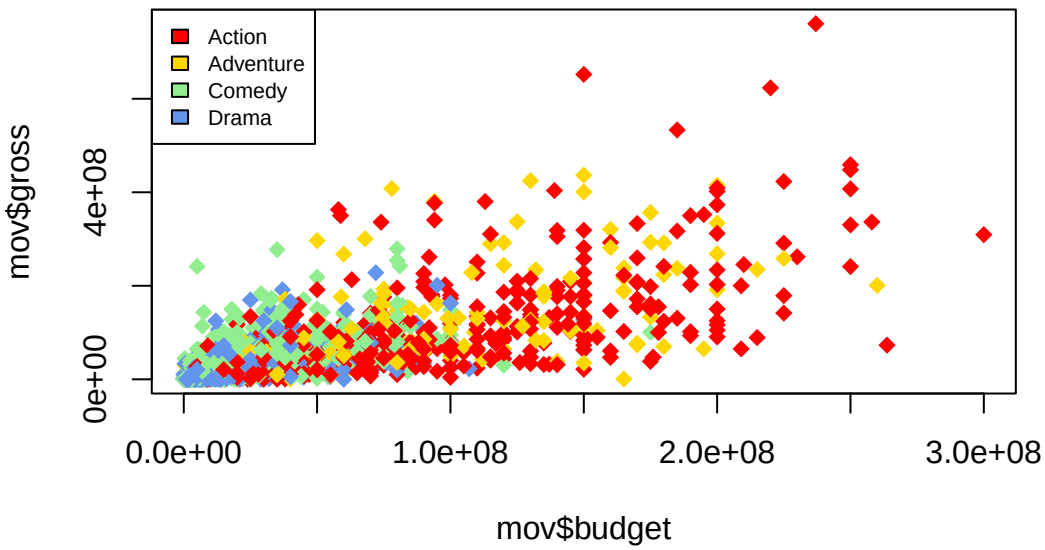

W takiej sytuacji dobrze jest dodać legendę do wykresu za pomocą funkcji `legend`. Ma ona wiele opcji, ale najważniejsze to położenie (można użyć współrzędnych lub opisu, np. “topleft”), elementy legendy (zwykle są to poziomy factora) oraz kolory. Legenda zostanie dołożona do wyświetlanego wykresu:

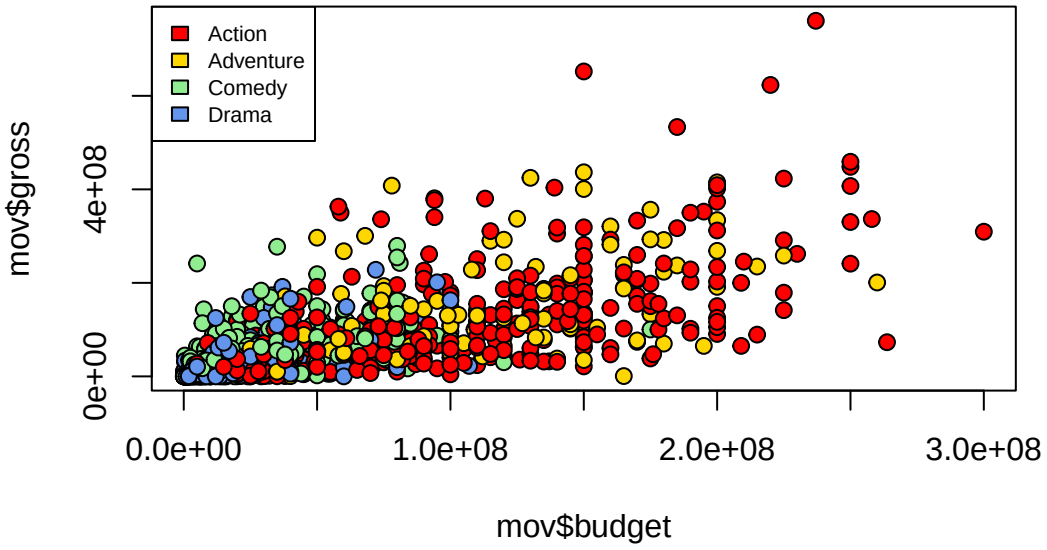
```
legend("topleft", legend = levels(type), fill = c("red", "navy"))
```




a) Utwórz wykres punktowy przedstawiający budżet i przychód filmu, kolorami rozróżnij gatunki filmów. Kształt punktu to pełny romb (18), jako kolory wybierz kolejno: *red*, *gold1*, *lightgreen*, *cornflowerblue*, dołóż legendę w lewym górnym rogu.

b) Utwórz taki sam wykres, ale tym razem wybierz jako kształt koło z wypełnieniem (21) i to wypełnienie punktów ma rozróżniać gatunki (możesz zastosować te same kolory co wcześniej lub inne, wybrane przez siebie).



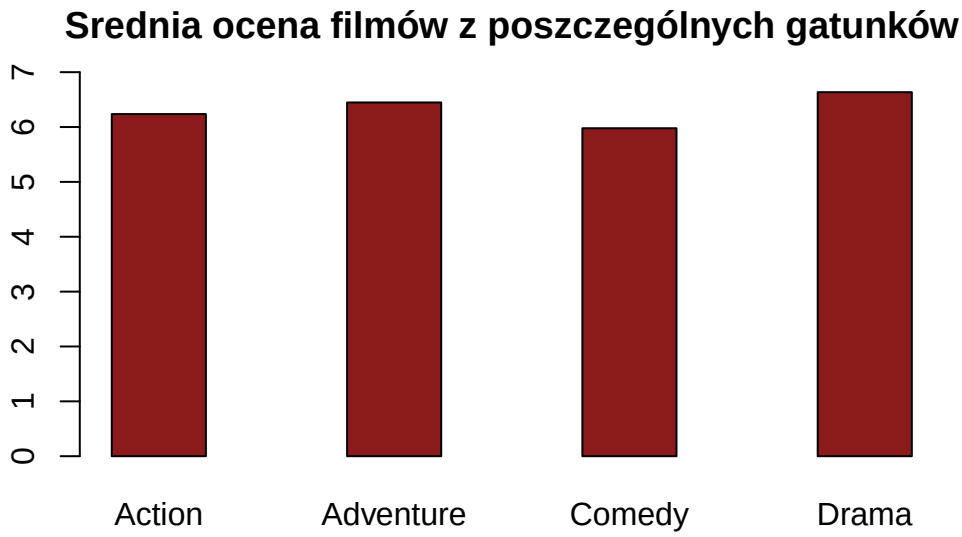



**Zadania różne**

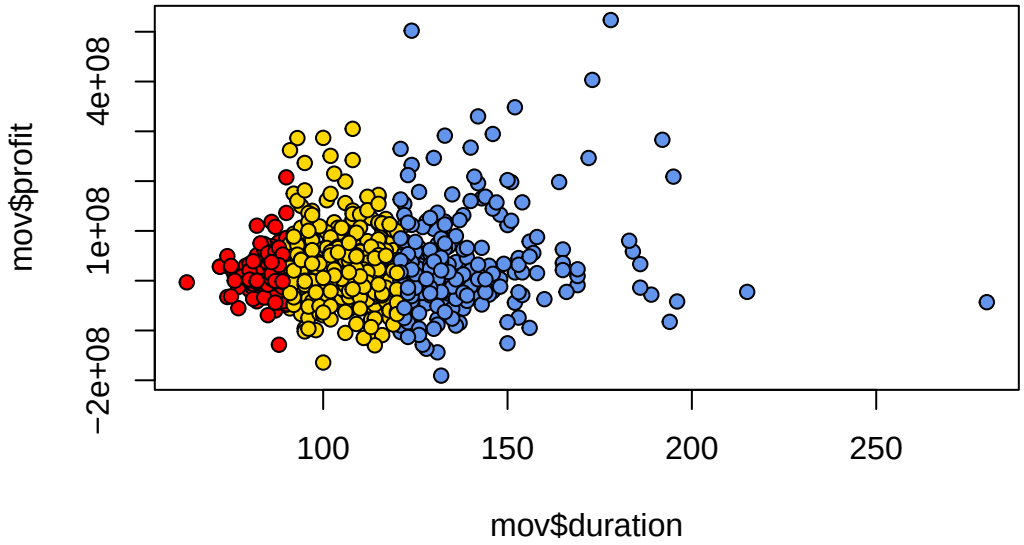
**56.** Wyświetl tytuł, przychód (*gross*) i rok 10 filmów z największym przychodem (uporządkowane malejąco wg przychodu).

```
##                                          title     gross year
## 946                                     Avatar 760505847 2009
## 1507                          Jurassic World 652177271 2015
## 1266                            The Avengers 623279547 2012
## 840                           The Dark Knight 533316061 2008
## 1519                Avengers: Age of Ultron 458991599 2015
## 1267                  The Dark Knight Rises 448130642 2012
## 415                                 Shrek 2 436471036 2004
## 1357       The Hunger Games: Catching Fire 424645577 2013
## 629  Pirates of the Caribbean: Dead Man's Chest 423032628 2006
## 1057                           Toy Story 3 414984497 2010
```

**57.** Za pomocą wykresu kolumnowego przedstaw średnią ocenę filmów z poszczególnych gatunków (*do wyliczenia średniej użyj funkcji* `tapply`). Filmy z których gatunków są przeciętnie najlepiej oceniane?




**58.** Dodaj do ramki `mov` nową kolumnę *profit*, która będzie zawierała zysk filmu w USA, czyli różnicę między przychodem (*gross*) a budżetem. Następnie utwórz wykres punktowy przedstawiający zależność między czasem trwania filmów a ich zyskiem. Użyj symbolu kółka z wypełnieniem, kolorem wypełnienia rozróżnij filmy wg kolumny *duration.interval*.

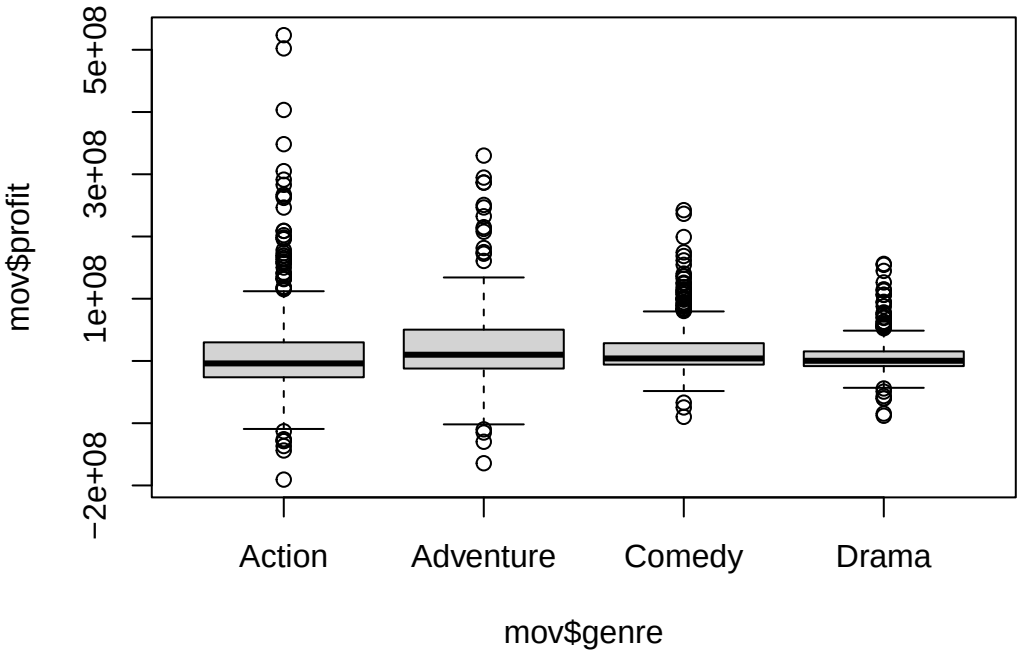



**59.** Wyświetl tytuł, przychód, gatunek, reżysera, rok i zysk 10 filmów z największą stratą (czyli z najmniejszym zyskiem). Wyniki mają być uporządkowane rosnąco wg zysku.

```
##                     title    gross     genre        director year     profit
## 1225           John Carter 73058679    Action  Andrew Stanton 2012 -190641321
## 389      The Polar Express   665426 Adventure Robert Zemeckis 2004 -164334574
## 1196            Battleship 65173160    Action      Peter Berg 2012 -143826840
## 1301              47 Ronin 38297305    Action     Carl Rinsch 2013 -136702695
## 1302 Jack the Giant Slayer 65171860 Adventure    Bryan Singer 2013 -129828140
## 1467     Jupiter Ascending 47375327    Action  Lana Wachowski 2015 -128624673
## 1074       Mars Needs Moms 21379315    Action     Simon Wells 2011 -128620685
## 1315       The Lone Ranger 89289910    Action  Gore Verbinski 2013 -125710090
## 1476                   Pan 34964818 Adventure      Joe Wright 2015 -115035182
## 1559               Warcraft 46978995    Action    Duncan Jones 2016 -113021005
```

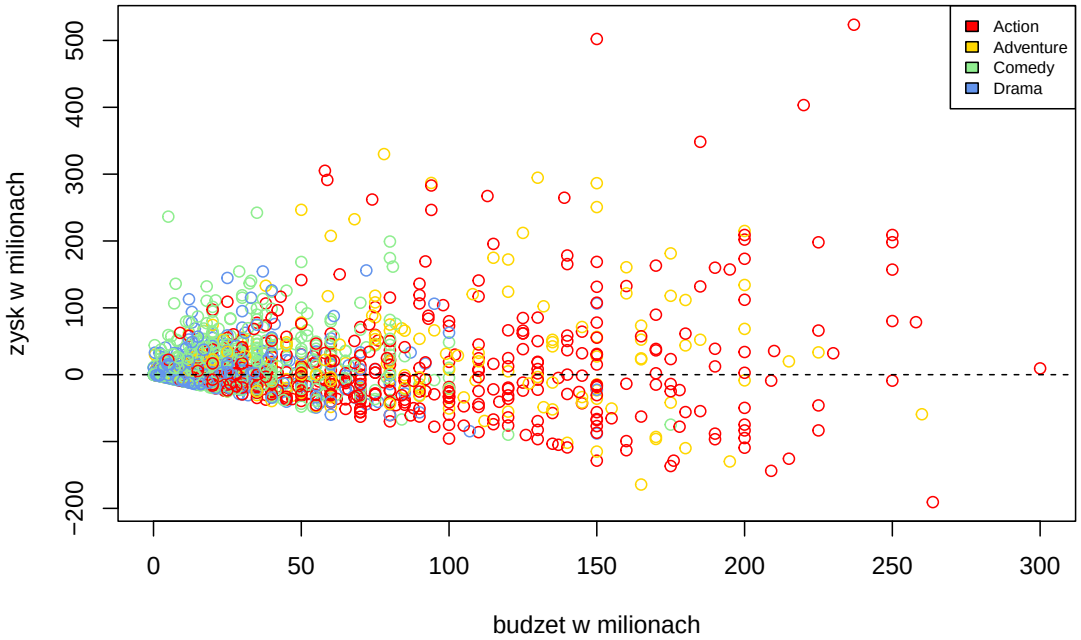
**60.** Pracuj na ramce `mov`.

a) Przedstaw za pomocą wykresu pudełkowego rozkłady zyskowności filmów poszczególnych gatunków.
b) Z wykresu wynika, że zróżnicowanie zyskowności jest różne dla poszczególnych gatunków. Dla których getunków jest wyraźnie wyższe? Uzasadnij odpowiedź wyliczając odchylenie standardowe zyskowności filmów poszczególnych gatunków.



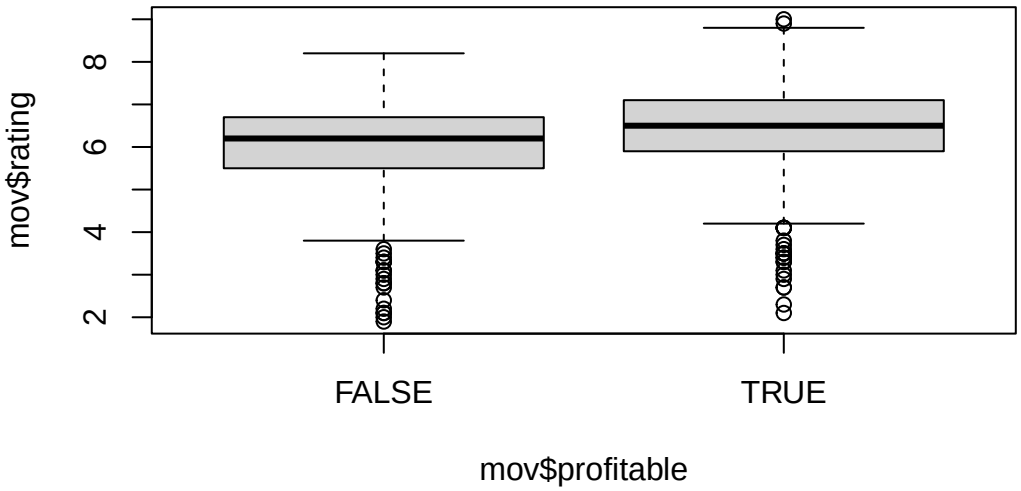

```
##    Action Adventure    Comedy     Drama
##  77735305  75478980  37712527  31492653
```

**61.** Utwórz wykres punktowy przedstawiający zależność między budżetem i zyskiem filmu (*wartości przedstaw w milionach, czyli podziel przez 1E6*), kolorami rozróżnij gatunki (*możesz określić własne kolory lub skorzystać z tych z zadania 55.*). Dodaj legendę i linię na wysokości 0 wskazującą granicę między filmami, które przyniosły zysk i tymi, które przyniosły straty. Opisz odpowiednio osie.




**62.** Dodaj do ramki `mov` nową kolumnę o nazwie *profitable*, która będzie zawierała wartości TRUE, jeśli zysk był dodatni (czyli film przynajmniej się zwrócił*) i FALSE w przeciwnym wypadku (*wskazówka: porównaj kolumny gross i budget - w efekcie otrzymasz wektor logiczny*). Zamień tę kolumnę na *factor*. Utwórz wykres pudełkowy średniej oceny filmów zyskownych i niezyskownych. Na jego podstawie odpowiedz, czy zyskowne filmy są przeciętnie lepiej oceniane?

* *Żeby film zwrócił się studiu zwykle musi zarobić przynajmniej dwukrotność swojego budżetu, ale tutaj mamy przychód tylko z USA, bez reszty świata, więc można sobie pozwolić na takie uproszczenie*




**63.** Sprawdź udział filmów opłacalnych i nieopłacalnych w poszczególnych gatunkach (*skorzystaj z funkcji* `prop.table` *i* `table` *z odpowiednią wartością argumentu margin*). Filmy których gatunków są najczęściej opłacalne?

```
##
##            Action Adventure    Comedy     Drama
##   FALSE 0.5456476 0.4020619 0.4228070 0.4939394
##   TRUE  0.4543524 0.5979381 0.5771930 0.5060606
```

## Zbiór danych bikes.csv

Pod zmienną `bikes` wczytaj dane z pliku *bikes.csv*. Zawierają one informacje o liczbie dziennych wypożyczeń rowerów miejskich w latach 2011 i 2012 w Dystrykcie Kolumbii (Washington D.C.) w USA. Poleceniami `str` i `summary` sprawdź strukturę i kompletność danych. Poszczególne kolumny oznaczają:

- instant - kolejny numer dnia
- dteday - data
- season - pora roku
- yr - rok
- mnth - miesiąc
- holiday - czy ten dzień był dniem wolnym
- weekday - dzień tygodnia
- workingday - czy dzień był dniem roboczym
- weathersit - warunki pogodowe
- temp, atemp - znormalizowana temperatura i odczuwana temperatura
- hum - znormalizowana wilgotność
- windspeed - znormalizowana prędkość wiatru
- casual, registered, cnt - liczba wypożyczeń przez niezarejestrowanych, zarejestrowanych i wszystkich użytkowników

Większość zmiennych kategorycznych jest zakodowana za pomocą liczb całkowitych, więc przed analizą dobrze zamienić je na factory z odpowiednimi etykietami.

### Obróbka danych

**64.** Kolumna *yr* jest zakodowana jako 0 dla 2011 i 1 dla 2012 roku. Wykonaj poniższy kod, który zamieni ją na *factor* z odpowiednimi poziomami:

```
bikes$yr <- factor(bikes$yr, levels = c(0, 1), labels = c(2011, 2012))
```

W podobny sposób zamień na zmienną czynnikową kolumnę *season*, cyfry 1:4 oznaczają w niej kolejno zimę, wiosnę, lato i jesień (skorzystaj z angielskich nazw: "winter", "spring", "summer", "autumn"). Polecenie `table` na tej kolumnie powinno po tym zwrócić następujący wynik:

```
##
## winter spring summer autumn
##    181    184    188    178
```

**65.** Zamień na zmienną czynnikową kolumnę *weekday*. Cyfry 1:6 oznaczają dni od poniedziałku do soboty, a cyfra 0 to niedziela. Jako etykiety dni ustaw angielskie skróty nazw dni: "Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun" (poziomy tego factora mają być w tej kolejności). Polecenie `table` na tej kolumnie powinno po tym zwrócić następujący wynik:

```
##
## Mon Tue Wed Thu Fri Sat Sun
## 105 104 104 104 104 105 105
```

**66.** Zamień na zmienną czynnikową kolumnę *weathersit*. Cyfry 1:3 oznaczają w niej kolejno: ładną pogodę ("clear"), mgłę ("mist") oraz lekki deszcz lub śnieg ("rain/snow"). Polecenie `table` na tej kolumnie powinno po tym zwrócić następujący wynik:

```
##
##     clear      mist rain/snow
##       463       247        21
```

**67.** Zamień na factory kolumny *mnth*, *holiday* i *workingday* pozostawiając domyślne poziomy jako wartości factora. Po wykonaniu zadań 64-67 polecenie `str(bikes)` powinno zwrócić następujący wynik:

```
## 'data.frame':    731 obs. of  16 variables:
## $ instant : int 1 2 3 4 5 6 7 8 9 10 ...
## $ dteday : chr "2011-01-01" "2011-01-02" "2011-01-03" "2011-01-04" ...
## $ season : Factor w/ 4 levels "winter","spring",..: 1 1 1 1 1 1 1 1 1 1 ...
## $ yr : Factor w/ 2 levels "2011","2012": 1 1 1 1 1 1 1 1 1 1 ...
## $ mnth : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
## $ holiday : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
## $ weekday : Factor w/ 7 levels "Mon","Tue","Wed",..: 6 7 1 2 3 4 5 6 7 1 ...
## $ workingday: Factor w/ 2 levels "0","1": 1 1 2 2 2 2 2 1 1 2 ...
## $ weathersit: Factor w/ 3 levels "clear","mist",..: 2 2 1 1 1 1 2 2 1 1 ...
## $ temp : num 0.344 0.363 0.196 0.2 0.227 ...
## $ atemp : num 0.364 0.354 0.189 0.212 0.229 ...
## $ hum : num 0.806 0.696 0.437 0.59 0.437 ...
## $ windspeed : num 0.16 0.249 0.248 0.16 0.187 ...
## $ casual : int 331 131 120 108 82 88 148 68 54 41 ...
## $ registered: int 654 670 1229 1454 1518 1518 1362 891 768 1280 ...
## $ cnt : int 985 801 1349 1562 1600 1606 1510 959 822 1321 ...
```

### Zadania różne

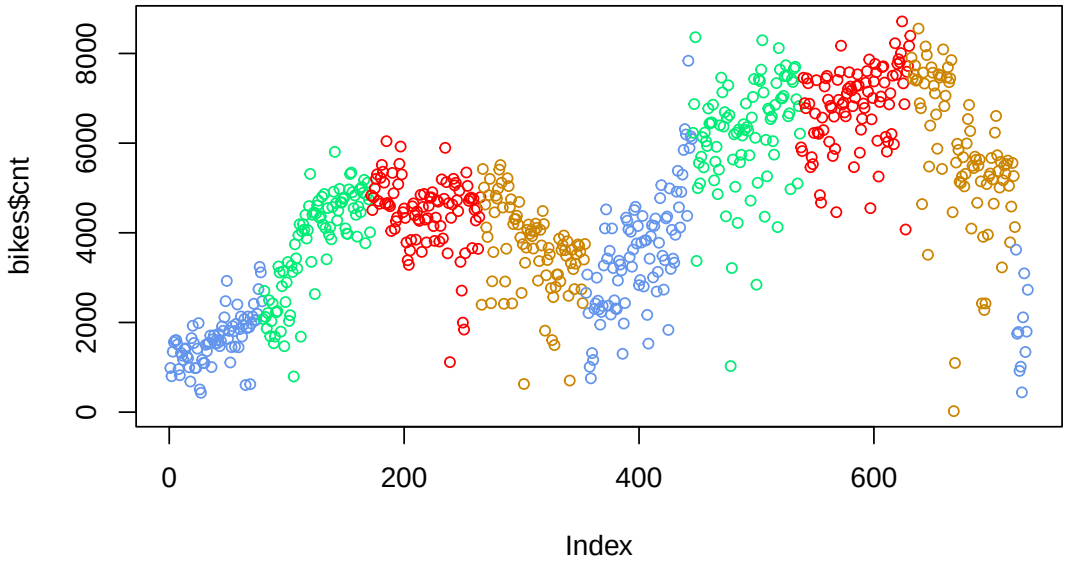
**68.** W zbiorze danych znajduje się liczba wypożyczeń przez użytkowników z abonamentem (*registered*), bez abonamentu (*casual*) oraz łączna liczba wypożyczeń (*cnt*). Sprawdź, czy suma *registered* i *casual* jest w każdym dniu równa liczbie *cnt.*

```
##
## TRUE
##  731
```

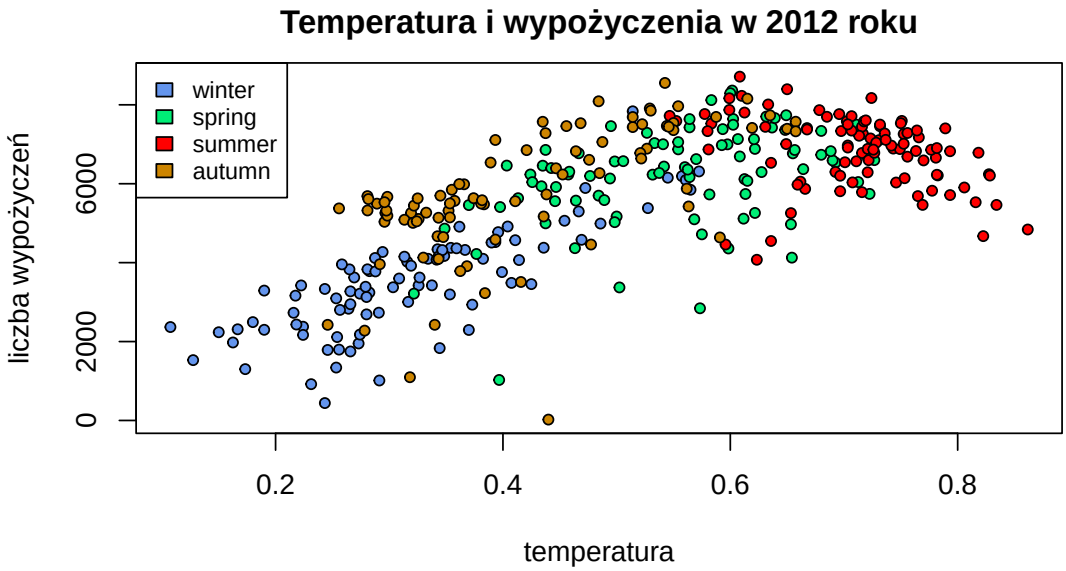
**69.** Sprawdź średnią liczbę wypożyczeń w poszczególnych latach.

```
##     2011     2012
## 3405.762 5599.934
```

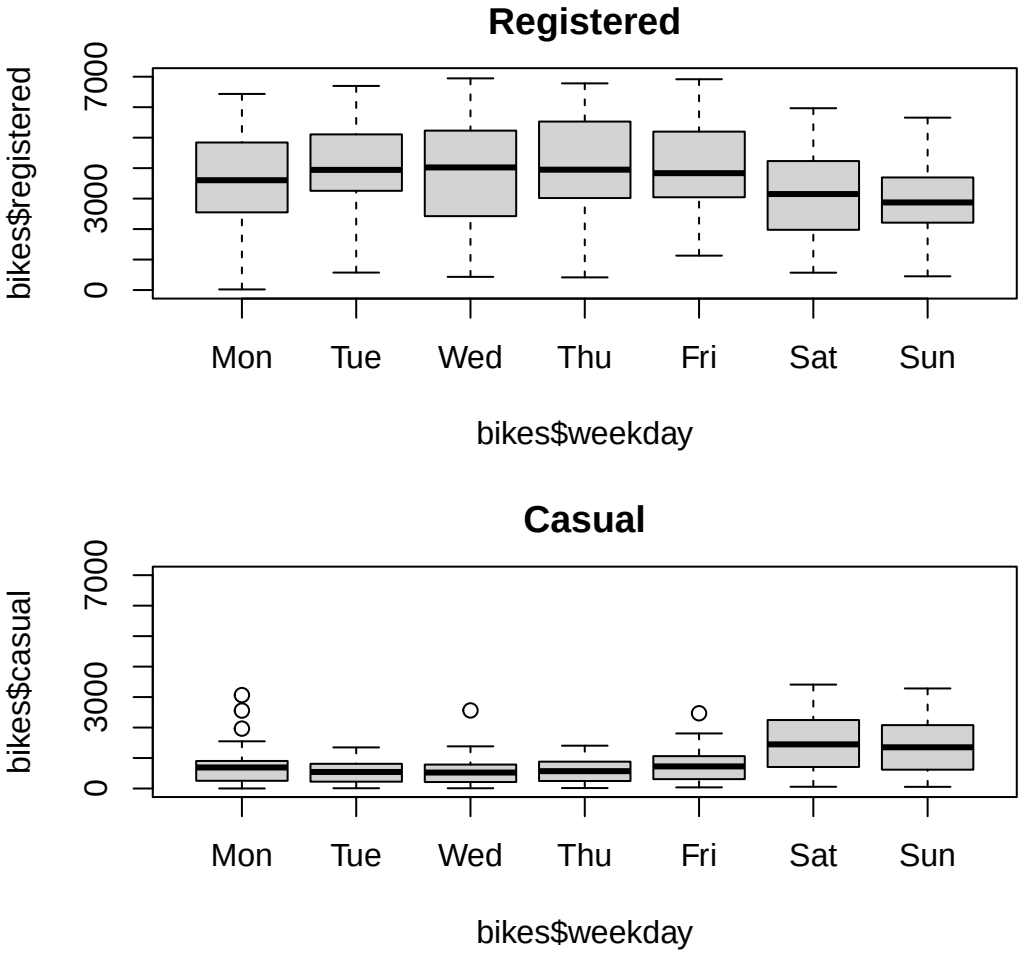
**70.** Przedstaw na wykresie punktowym liczbę wypożyczeń w poszczególnych dniach, kolorami rozróżnij pory roku (użyj kolorów, które kojarzą się z poszczególnymi porami roku, np. "cornflowerblue", "springgreen2", "red", "orange3").




**71.** Przedstaw na wykresie punktowym temperaturę i liczbę wszystkich wypożyczeń w 2012 roku. Użyj ponktu z wypełnieniem, kolorami wypełnienia rozróżnij pory roku, opisz odpowiednio osie, dodaj tytuł wykresu i legendę.



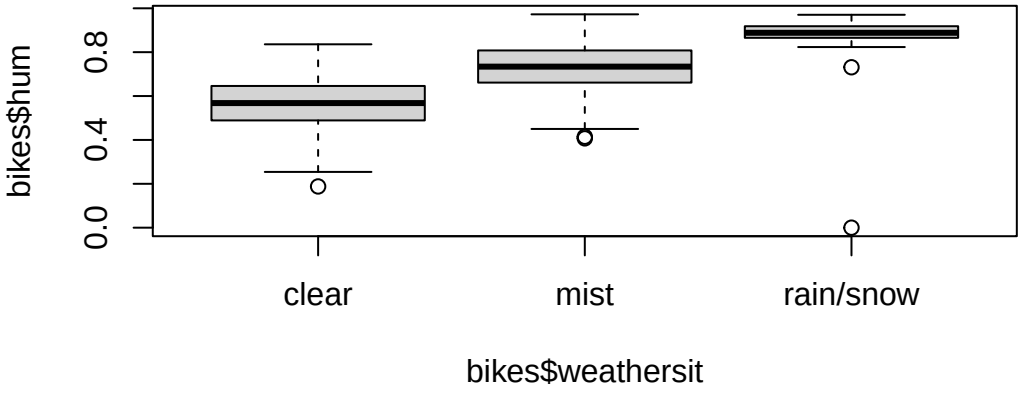

**72.** Przedstaw na wykresie pudełkowym rozkłady liczby wypożyczeń w poszczególnych dniach użytkowników z abonamentem (*registered*), a na drugim podobnym wykresie użytkowników bez abonamentu (*casual*). Na obu wykresach ustaw maksymalną wartość osi **Y** na 7000 (żeby można było łatwo porównać skalę). Jakie są różnice między tymi dwiema grupami?




**73.** Liczba wypożyczeń przez użytkowników *registered* jest zwykle znacznie wyższa niż użytkowników *casual*, jednak jest kilka dni, kiedy tych drugich jest więcej. Sprawdź co to za dni i czym się charakateryzują.

```
##         dteday casual registered holiday workingday weekday
## 93  2011-04-03   1651       1598       0          0     Sun
## 185 2011-07-04   3065       2978       1          0     Mon
## 247 2011-09-04   2521       2419       0          0     Sun
```

**74.** Sprawdź za pomocą wykresu pudełkowego rozkład wilgotności powietrza w zależności od warunków pogodowych. Czy wynik jest zgodny z oczekiwaniami?



## Zadanie dodatkowe

**75!** Pracuj na danych z pliku *imdb.csv* (to dane inne od *movies.csv*). Napisz kod wyświetlający 20 aktorów, którzy wystąpili w największej liczbie filmów z tego zbioru (wyświetl też liczbę tych filmów). Poniżej (jako wskazówka) dane pierwszej trójki.

```
##
## Mark Wahlberg  Hugh Jackman     Brad Pitt
##            15            14            13
```