Podstawy języka R - zadania (6 - dplyr)

Tomasz Owczarek, Mateusz Naramski

2024/2025, semestr letni

Zbiór danych movies.csv

Pod zmienną `mov` wczytaj dane z pliku *movies.csv*. Zamień kolumnę *genre* na zmienną czynnikową.

Funkcje `select`, `filter`, `arrange`

101. Korzystając z funkcji `select` i `filter` z pakietu `dplyr` wyświetl:

a) tytuły, gatunki, rok i czas trwania filmów, które trwają dłużej niż 180 minut
b) tytuły, rok, przychód i oceny filmów, których przychód jest większy niż 600mln lub ocena jest wyższa niż 8,5

```
##                                         title     genre year duration
## 1                                 Pearl Harbor    Action 2001      184
## 2                               Gods and Generals     Drama 2003      280
## 3  The Lord of the Rings: The Return of the King    Action 2003      192
## 4                                         Troy Adventure 2004      196
## 5                               Kingdom of Heaven    Action 2005      194
## 6                                     Grindhouse    Action 2007      189
## 7                                       Watchmen    Action 2009      215
## 8                                       Margaret     Drama 2011      186
## 9                                     Iron Man 3    Action 2013      195
## 10          The Hobbit: The Desolation of Smaug Adventure 2013      186
## 11           Batman v Superman: Dawn of Justice    Action 2016      183
```

```
##                                       title year     gross rating
## 1         The Lord of the Rings: The Two Towers 2002 340478898    8.7
## 2 The Lord of the Rings: The Return of the King 2003 377019252    8.9
## 3                               The Dark Knight 2008 533316061    9.0
## 4                                        Avatar 2009 760505847    7.9
## 5                                     Inception 2010 292568851    8.8
## 6                                  The Avengers 2012 623279547    8.1
## 7                                  Interstellar 2014 187991439    8.6
## 8                                Jurassic World 2015 652177271    7.0
```

102. Korzystając z funkcji `select`, `filter` i `arrange` z pakietu `dplyr` wyświetl:

a) tytuły, rok i oceny filmów z lat 2011-2016, których oceny wyniosły przynajmniej 8,2, uporządkuj wyniki wg ocen rosnąco
b) tytuły, rok, gatunek i budżet **komedii**, których budżet wyniósł więcej niż 90mln, uporządkuj wyniki malejąco wg budżetu

```
##                         title year rating
## 1                     Warrior 2011    8.2
## 2 Captain America: Civil War 2016    8.2
## 3                 Inside Out 2015    8.3
## 4      The Dark Knight Rises 2012    8.5
## 5           Django Unchained 2012    8.5
## 6                    Whiplash 2014    8.5
## 7               Interstellar 2014    8.6
```

```
##                     title year  genre    budget
## 1          Evan Almighty 2007 Comedy 175000000
## 2        How Do You Know 2010 Comedy 120000000
## 3 Fun with Dick and Jane 2005 Comedy 100000000
## 4     Sex and the City 2 2010 Comedy 100000000
## 5         Little Fockers 2010 Comedy 100000000
## 6           Dark Shadows 2012 Comedy 100000000
## 7           The Campaign 2012 Comedy  95000000
```

Funkcje `group_by` i `summarise`

103. Korzystając z funkcji `group_by` i `summarise` z pakietu `dplyr` wyświetl:

a) średni czas trwania i średnią ocenę filmów z poszczególnych lat
b) średni budżet i średni przychód filmów z poszczególnych gatunków

```
## # A tibble: 16 x 3
##     year mean.duration mean.rating
##    <int>         <dbl>       <dbl>
##  1  2001          104.        6.08
##  2  2002          102.        6.12
##  3  2003          108.        6.04
##  4  2004          107.        6.28
##  5  2005          108.        6.24
##  6  2006          105.        6.17
##  7  2007          109.        6.35
##  8  2008          106.        6.23
##  9  2009          106.        6.24
## 10  2010          105.        6.21
## 11  2011          108.        6.26
## 12  2012          109.        6.42
## 13  2013          111.        6.39
## 14  2014          112.        6.51
## 15  2015          113.        6.37
## 16  2016          115.        6.33
```

```
## # A tibble: 4 x 3
##   genre     mean.budget mean.gross
##   <fct>           <dbl>      <dbl>
## 1 Action      83862739.  94909692.
## 2 Adventure   77065464. 104017836.
## 3 Comedy      26224175.  41515367.
## 4 Drama       24878564.  31614972.
```

104. Korzystając z funkcji `group_by` i `summarise` z pakietu `dplyr` wyświetl średnią liczbę recenzji filmów z 2010 i 2011 roku w podziale na lata i na gatunki (użyj funkcji `filter`, zeby ograniczyć rekordy tylko do lat 2010 i 2011).

```
## # A tibble: 8 x 3
## # Groups:   year [2]
##    year genre     mean.reviews
##   <int> <fct>            <dbl>
## 1  2010 Action            707.
## 2  2010 Adventure         651.
## 3  2010 Comedy            316.
## 4  2010 Drama             486.
## 5  2011 Action            702
## 6  2011 Adventure         510.
## 7  2011 Comedy            389.
## 8  2011 Drama             494.
```

Funkcja `mutate`

105. Korzystając z funkcji `mutate` wyświetl:

a) tytuły, rok oraz 2 nowe kolumny: *budget_mln* i *gross_mln* zawierające budżet i przychód filmów w milionach (zaokrąglone do dwóch miejsc po przecinku) (*użyj mutate do stworzenia nowych kolumn, select do wyboru kolumn, przekaż wynik do funkcji* **`head`** *z n = 15*)
b) tytuły i zysk 10 filmów z największym zyskiem (*gross - budget*), uporządkowane malejąco wg zysku; kolumnę z zyskiem nazwij *profit* (*skorzystaj dodatkowo z funkcji* **`select`** *i* **`arrange`***, a wynik przekaż do funkcji* **`head`** *z n = 10*)

```
##                         title year budget_mln gross_mln
## 1                     Glitter 2001       22.0      4.27
## 2              Soul Survivors 2001       14.0      3.10
## 3  Megiddo: The Omega Code 2 2001       22.0      5.97
## 4                  On the Line 2001       16.0      4.36
## 5                      Jason X 2001       11.0     12.61
## 6                       Driven 2001       72.0     32.62
## 7         Freddy Got Fingered 2001       15.0     14.25
## 8                     The Wash 2001        4.0     10.10
## 9                 Corky Romano 2001       11.0     23.98
## 10             Dr. Dolittle 2 2001       72.0    112.95
## 11               Black Knight 2001       50.0     33.42
## 12                  The Animal 2001       22.0     55.76
## 13             Ghosts of Mars 2001       28.0      8.43
## 14                Harvard Man 2001        5.5      0.06
## 15           Say It Isn't So 2001       25.0      5.52
```

```
##                                title    profit
## 1                             Avatar 523505847
## 2                     Jurassic World 502177271
## 3                       The Avengers 403279547
## 4                    The Dark Knight 348316061
## 5                    The Hunger Games 329999255
## 6                           Deadpool 305024263
## 7  The Hunger Games: Catching Fire 294645577
## 8                     American Sniper 291323553
## 9                        Finding Nemo 286838870
## 10                            Shrek 2 286471036
```

106. Korzystając z funkcji `mutate` wyświetl tytuły, oceny, czas trwania i nową kolumnę o nazwie *duration2*, która będzie zawierała oznaczenie przedziału czasu trwania filmów (ustal następujące przedziały:

do 90 minut, 90-120 minut, 120 - 180 minut, powyżej 180 minut) dla 10 filmów z najwyższymi ocenami (*użyj funkcji* ***mutate*** *i* ***cut****, żeby utworzyć kolumnę duration2, oraz* ***arrange*** *do posortowania filmów i* ***head*** *dla ograniczenia liczby wierszy do pierwszych 10*)

```
##                                          title rating duration  duration2
## 1                                The Dark Knight    9.0      152  (120,180]
## 2  The Lord of the Rings: The Return of the King    8.9      192 (180, Inf]
## 3                                    Inception    8.8      148  (120,180]
## 4          The Lord of the Rings: The Two Towers    8.7      172  (120,180]
## 5                                  Interstellar    8.6      169  (120,180]
## 6                                   The Prestige    8.5      130  (120,180]
## 7                          The Dark Knight Rises    8.5      164  (120,180]
## 8                               Django Unchained    8.5      165  (120,180]
## 9                                       Whiplash    8.5      107   (90,120]
## 10                                        WALL·E    8.4       98   (90,120]
```

Zadania różne

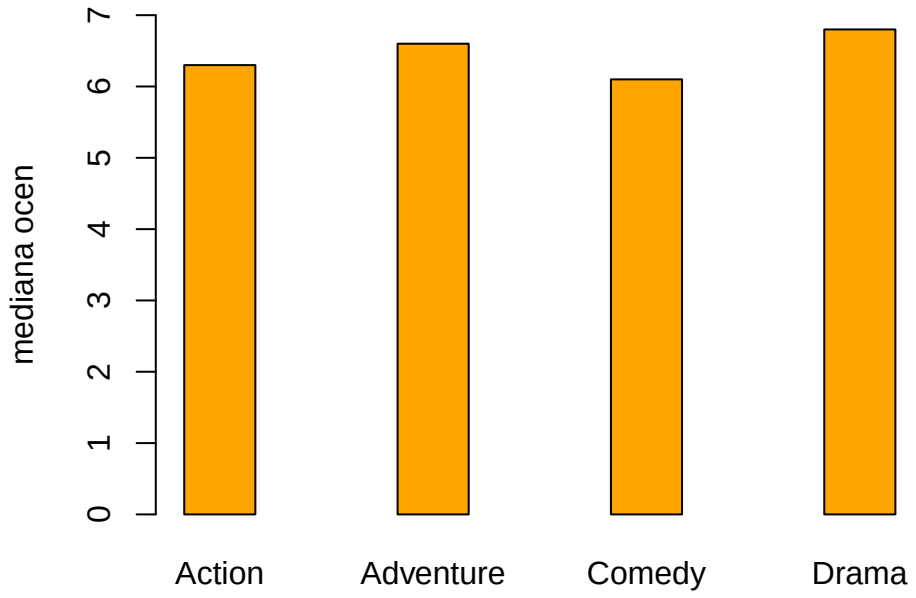
107. Utwórz nową ramkę `mov.action10` zawierająca tytuły, reżyserów, lata i przychód 10 filmów akcji z największym przychodem (uporządkowane malejąco wg przychodu).

```
##                                        title          director year     gross
## 1                                     Avatar     James Cameron 2009 760505847
## 2                              Jurassic World   Colin Trevorrow 2015 652177271
## 3                                The Avengers       Joss Whedon 2012 623279547
## 4                             The Dark Knight Christopher Nolan 2008 533316061
## 5                       Avengers: Age of Ultron       Joss Whedon 2015 458991599
## 6                         The Dark Knight Rises Christopher Nolan 2012 448130642
## 7  Pirates of the Caribbean: Dead Man's Chest    Gore Verbinski 2006 423032628
## 8                                   Iron Man 3       Shane Black 2013 408992272
## 9                  Captain America: Civil War     Anthony Russo 2016 407197282
## 10                                 Spider-Man         Sam Raimi 2002 403706375
```

108. Utwórz ramkę `mov.genre.rating` zawierającą średnią, medianę, maksimum i minimum oceny filmów z poszczególnych gatunków.

```
## # A tibble: 4 x 5
##   genre     mean.rating median.rating max.rating min.rating
##   <fct>           <dbl>         <dbl>      <dbl>      <dbl>
## 1 Action           6.24           6.3        9          2.1
## 2 Adventure        6.45           6.6        8.6        2.3
## 3 Comedy           5.98           6.1        7.9        1.9
## 4 Drama            6.64           6.8        8.5        2.1
```

109. Korzystając z ramki `mov.genre.rating` z poprzedniego zadania utwórz następujący wykres kolumnowy (użyj argumentu `names_arg` dla opisania kolumn, zakres osi Y zmień do 0-7):



110. Korzystając z funkcji z pakietu `dplyr` utwórz ramkę danych z dwiema kolumnami: rokiem i średnią ocen filmów z danego roku. Pierwszych 5 rekordów tej ramki poniżej.

```
## # A tibble: 5 x 2
##    year mean.rating
##   <int>       <dbl>
## 1  2001        6.08
## 2  2002        6.12
## 3  2003        6.04
## 4  2004        6.28
## 5  2005        6.24
```

111. Korzystając z funkcji z pakietu `dplyr` utwórz ramkę danych z czterema kolumnami: rokiem, gatunkiem, średnią oceną filmów danego gatunku w danym roku oraz liczbą filmów danego gatunku w danym roku. Pierwszych 8 rekordów tej ramki poniżej.

```
## # A tibble: 8 x 4
## # Groups:   year [2]
##    year genre     mean.rating     n
##   <int> <fct>           <dbl> <int>
## 1  2001 Action           5.81    31
## 2  2001 Adventure        6.5      8
## 3  2001 Comedy           5.94    48
## 4  2001 Drama            6.56    24
## 5  2002 Action           5.99    33
## 6  2002 Adventure        6.19    11
## 7  2002 Comedy           5.85    39
## 8  2002 Drama            6.55    32
```

112. Wyświetl 10 reżyserów, których średnia zysku filmów jest najwyższa, średni zysk tych filmów w milionach oraz liczbę filmów, które wyreżyserowali (uwzględnij tylko tych reżyserów, którzy wyreżyserowali przynajmniej 5 filmów).

W tym celu potokowo:

- za pomocą funkcji `mutate` dodaj do ramki `mov` kolumnę *profit* z zyskiem filmów,
- za pomocą funkcji `group_by` pogrupuj rekordy po reżyserach,
- za pomocą kolumny `mutate` dodaj kolumnę *dir.n.films* z liczbą filmów poszczególnych reżyserów (liczenie rekordów - funkcja `n()`),
- za pomocą funkcji `filter` wybierz tylko te rekordy, które w kolumnie *dir.n.films* mają liczbę przynajmniej 5,
- za pomocą funkcji `summarise` oblicz średnią zyskowność filmów (w milionach) i średnią z liczby filmów poszczególnych reżyserów (ten drugi wynik będzie po prostu liczbą filmów danego reżysera),
- za pomocą funkcji `arrange` uporządkuj filmy malejąco wg średniej zyskowności,
- za pomocą funkcji `head` ogranicz wynik do 10 pierwszych rekordów,
- na samym końcu zamień wynik na ramkę danych za pomocą funkcji *as.data.frame()*.

```
##             director mean.profit n
## 1   Francis Lawrence   151.10039 5
## 2  Christopher Nolan   113.09953 7
## 3      Todd Phillips    88.13840 6
## 4        Jon Favreau    78.81825 5
## 5        Michael Bay    63.00626 9
## 6          Tim Story    57.61687 6
## 7     Clint Eastwood    53.44307 7
## 8         Tim Burton    49.32441 5
## 9      Jason Reitman    42.55219 5
## 10       Tyler Perry    41.55629 6
```

Piraci!

Pakiet `yarrr` zawiera kilka ciekawych zbiorów danych o piratach (*dane są sztuczne, ale dość zabawne*).

113.

a) Zainstaluj i załaduj pakiet `yarrr`.
b) Pod zmienną `pir` podstaw dane z ramki `pirates` z tego pakietu.
c) Zamień wszystkie kolumny znakowe z ramki `pir` na factory.

Po tym wszystkim ramka `pir` *powinna mieć następującą strukturę:*

```
## 'data.frame':    1000 obs. of  17 variables:
## $ id : int 1 2 3 4 5 6 7 8 9 10 ...
## $ sex : Factor w/ 3 levels "female","male",..: 2 2 2 1 1 2 1 1 1 2 ...
## $ age : num 28 31 26 31 41 26 31 31 28 30 ...
## $ height : num 173 209 170 144 158 ...
## $ weight : num 70.5 105.6 77.1 58.5 58.4 ...
## $ headband : Factor w/ 2 levels "no","yes": 2 2 2 1 2 2 2 2 2 2 ...
## $ college : Factor w/ 2 levels "CCCC","JSSFP": 2 2 1 2 2 1 2 2 2 2 ...
## $ tattoos : num 9 9 10 2 9 7 9 5 12 12 ...
## $ tchests : num 0 11 10 0 6 19 1 13 37 69 ...
## $ parrots : num 0 0 1 2 4 0 7 7 2 4 ...
## $ favorite.pirate: Factor w/ 6 levels "Anicetus","Blackbeard",..: 5 5 5 5 4 5 2
##    4 1 5 ...
## $ sword.type : Factor w/ 4 levels "banana","cutlass",..: 2 2 2 4 2 2 2 2 2 2
##    ...
## $ eyepatch : num 1 0 1 1 1 1 0 1 0 1 ...
## $ sword.time : num 0.58 1.11 1.44 36.11 0.11 ...
## $ beard.length : num 16 21 19 2 0 17 1 1 1 25 ...
## $ fav.pixar : Factor w/ 15 levels "A Bug's Life",..: 8 15 6 6 6 7 3 6 13 7 ...
## $ grogg : num 11 9 7 9 14 7 9 12 16 9 ...
```

Zadania różne

114. Korzystając z funkcji z pakietu `dplyr` wyświetl *id*, płeć (*sex*) i wiek (*age*) 10 najstarszych piratów (malejąco wg wieku).

```
##     id    sex age
## 1  705 female  46
## 2   15 female  45
## 3   95 female  45
## 4  651 female  44
## 5  774 female  43
## 6  848 female  43
## 7    5 female  41
## 8  330 female  41
## 9  639 female  41
## 10 656 female  41
```

115. Korzystając z funkcji z pakietu `dplyr` wyświetl *id*, płeć (*sex*) i długość brody (*beard.length*) 10 piratów z najdłuższą brodą (malejąco wg długosci brody).

```
##     id   sex beard.length
## 1  320  male           40
```

```
## 2  360  male           37
## 3  157  male           35
## 4  286  male           34
## 5  716 other           34
## 6  220  male           33
## 7  440  male           33
## 8  955  male           33
## 9  872  male           32
## 10  78  male           31
```

116. Korzystając z funkcji z pakietu `dplyr` wyświetl *id*, płeć (*sex*), opaskę (*eyepatch*) i liczbę papug (*parrots*) 10 piratów z największą liczbą papug, ale tylko tych z przepaską na oku (*eyepatch == 1*) (malejąco wg liczby papug).

```
##     id    sex eyepatch parrots
## 1  622 female        1      27
## 2   93   male        1      24
## 3   95 female        1      21
## 4   96 female        1      20
## 5  307 female        1      18
## 6   61 female        1      17
## 7  500 female        1      16
## 8  975 female        1      16
## 9  995   male        1      16
## 10 139   male        1      15
```

117. Korzystając z funkcji z pakietu `dplyr` wyświetl *id*, płeć (*sex*), ulubionego pirata (*favorite.pirate*), broń (*sword.type*) i liczbę zdobytych skrzynek ze skarbami (*tchests*) 10 piratów z największą liczbą zdobytych skrzynek, ale tylko tych, których ulubionym piratem jest *Czarnobrody* (*Blackbeard*) (malejąco wg liczby zdobytych skrzynek).

```
##     id    sex favorite.pirate sword.type tchests
## 1  718 female      Blackbeard     banana     139
## 2  493   male      Blackbeard    cutlass     134
## 3  240   male      Blackbeard    cutlass     125
## 4   85 female      Blackbeard    cutlass     122
## 5   18 female      Blackbeard    cutlass     107
## 6  748 female      Blackbeard    cutlass      90
## 7  373 female      Blackbeard    cutlass      84
## 8  703  other      Blackbeard    cutlass      80
## 9  502   male      Blackbeard    cutlass      72
## 10 694 female      Blackbeard      sabre      62
```

118. Korzystając z funkcji z pakietu `dplyr` wyświetl w postaci `tibble` średnią liczbę skrzynek, którą zdobyli absolwenci każdej z pirackich uczelni (*college*) oraz liczbę tych absolwentów.

```
## # A tibble: 2 x 3
##   college mean.tchests     n
##   <fct>          <dbl> <int>
## 1 CCCC            19.4   658
## 2 JSSFP           28.9   342
```

119. Korzystając z funkcji z pakietu `dplyr` wyświetl w postaci `tibble` średnią liczbę skrzynek, którą zdobyli absolwenci każdej z pirackich uczelni (*college*) oraz liczbę tych absolwentów z dodatkowym podziałem wg płci.

```
## # A tibble: 6 x 4
## # Groups:   college [2]
##   college sex    mean.tchests     n
##   <fct>   <fct>         <dbl> <int>
## 1 CCCC    female         21.4   228
## 2 CCCC    male           18.2   397
## 3 CCCC    other          21.0    33
## 4 JSSFP   female         28.5   236
## 5 JSSFP   male           28.4    93
## 6 JSSFP   other          40.5    13
```

120. Korzystając z funkcji z pakietu `dplyr` wyświetl w postaci `tibble` dla każdej płci: średni wiek, średnią długość brody, średnią liczbę kubków wypijanego rumu (*grogg*).

```
## # A tibble: 3 x 4
##   sex    mean.age mean.beard mean.grogg
##   <fct>     <dbl>      <dbl>      <dbl>
## 1 female     29.9      0.399      10.3
## 2 male       25.0     19.4         9.97
## 3 other      27       14.9        10.7
```

Pytania

Skorzystaj z funkcji z pakietu `dplyr`, żeby odpowiednio podsumować dane i szybko odpowiedzieć na nastepujące pytania:

121. Piraci której płci są przeciętnie najwyżsi?

122. Którzy piraci mają przeciętnie najwięcej tatuaży - z bandaną czy bez (*headband*)?

123. Piraci używający której broni zdobyli średnio najwięcej skrzynek ze skarbami?

124. Czy posiadanie opaski ma związek z czasem wydobycia broni (*sword.time*)?

Zadanie dodatkowe

125! Pracuj na danych `imdb`. Za pomocą funkcji z pakietu `dplyr` wyświetl tytuły oraz przychody filmów, które miały największy przychód w poszczególnych latach (jedna ramka danych z 11 wierszami i trzema kolumnami, filmy uporządkowane wg roku, poniżej ramka z pierwszą literą tytułów i pierwszą cyfrą przychodów - *dla sprawdzenia*). Uwaga - dozwolone jest użycie jedynie funkcji, które poznaliśmy na zajęciach.

```
##    Year Title Revenue.Millions.
## 1  2006  P...              4...
## 2  2007  S...              3...
## 3  2008  T...              5...
## 4  2009  A...              7...
## 5  2010  T...              4...
## 6  2011  H...              3...
## 7  2012  T...              6...
## 8  2013  T...              4...
## 9  2014  A...              3...
## 10 2015  S...              9...
## 11 2016  R...              5...
```