



Bursa Uludağ Üniversitesi

Bilgisayar Mühendisliği Bölümü

R Programlama ve Makine Öğrenmesi Uygulamaları Dersi

Ödev H1O1

Beyzagül ŞAHİN 032290087

Altındaki kod ile gerekli olan paketleri yükledik.

```
install.packages(c('corr', 'tidyverse', 'nycflights13', 'Lahman', 'ggplot2'))
```

ls("package:nycflights13") Paket içeriğini listeler.

```
>
> ls("package:nycflights13")
[1] "airlines" "airports" "flights"  "planes"   "weather"
> View(flights)
>
```

View('flights') komutu ile de veriyi görüntülüyoruz.

	year	month	day	dep_time	sched_dep_time	dep_delay	arr_time	sched_arr_time	arr_delay	carrier	flight	tailnum	origin	dest	air
1	2013	1	1	517	515	2	830	819	11	UA	1545	N14228	EWB	IAH	
2	2013	1	1	533	529	4	850	830	20	UA	1714	N24211	LGA	IAH	
3	2013	1	1	542	540	2	923	850	33	AA	1141	N619AA	JFK	MIA	
4	2013	1	1	544	545	-1	1004	1022	-18	B6	725	N804JB	JFK	BQN	
5	2013	1	1	554	600	-6	812	837	-25	DL	461	N668DN	LGA	ATL	
6	2013	1	1	554	558	-4	740	728	12	UA	1696	N39463	EWB	ORD	
7	2013	1	1	555	600	-5	913	854	19	B6	507	N516JB	EWB	FLL	
8	2013	1	1	557	600	-3	709	723	-14	EV	5708	N829AS	LGA	IAD	
9	2013	1	1	557	600	-3	838	846	-8	B6	79	N593JB	JFK	MCO	
10	2013	1	1	558	600	-2	753	745	8	AA	301	N3ALAA	LGA	ORD	
11	2013	1	1	558	600	-2	849	851	-2	B6	49	N793JB	JFK	PBI	
12	2013	1	1	558	600	-2	853	856	-3	B6	71	N657JB	JFK	TPA	
13	2013	1	1	558	600	-2	924	917	7	UA	194	N29129	JFK	LAX	
14	2013	1	1	558	600	-2	923	937	-14	UA	1124	N53441	EWB	SFO	
15	2013	1	1	559	600	-1	941	910	31	AA	707	N3DUAA	LGA	DFW	
16	2013	1	1	559	559	0	702	706	-4	B6	1806	N708JB	JFK	BOS	

Left_join() fonksiyonu, verisetlerini birleştirmeyi sağlar, Filter fonksiyonu ile de uçuş süresi en yüksek olan uçuşları gösterir.

```
> data1<- left_join(flights,planes,by="tailnum")
>
> data1 %>% filter(arr_time==max(arr_time,na.rm = T)) %>%
+   select(year.y, manufacturer,model)
# A tibble: 1 x 3
  year.y manufacturer model
  <int> <chr>         <chr>
1    2002 BOEING      767-424ER
```

```

>
> data2<- left_join(weather,flights,by=c("origin","year","month","day","hour"))
> data2 %>% select(air_time,temp,wind_speed,humid) %>%
+   correlate()
Correlation computed with
• Method: 'pearson'
• Missing treated using: 'pairwise.complete.obs'
# A tibble: 4 × 5
  term      air_time    temp wind_speed    humid
  <chr>      <dbl>    <dbl>    <dbl>    <dbl>
1 air_time    NA    -0.0367    0.0263  0.0405
2 temp    -0.0367    NA    -0.140    0.0374
3 wind_speed  0.0263 -0.140    NA    -0.187
4 humid      0.0405  0.0374  -0.187    NA
>

```

Right_join fonksiyonu da veri birleştirmek için kullanılır. Filter fonksiyonu, bu kodda kalkış gecikmesi olan uçuşların filtrelenmesini sağlar. Groupby fonksiyonu da her hava yolu için gruptama yapar ve gecikmelerin ortalama ve medyanını hesaplar.

```

> data3<- right_join(airlines,flights,by="carrier")
> data3 %>% filter(dep_delay>0) %>% na.omit() %>% group_by(name) %>% sum
# A tibble: 16 × 3
  name                mean median
  <chr>              <dbl>  <dbl>
1 AirTran Airways Corporation  40.6   16
2 Alaska Airlines Inc.       31.5   12
3 American Airlines Inc.     37.2   16
4 Delta Air Lines Inc.       37.3   16
5 Endeavor Air Inc.          48.5   26
6 Envoy Air                 44.7   27
7 ExpressJet Airlines Inc.   50.2   31
8 Frontier Airlines Inc.     45.2   18
9 Hawaiian Airlines Inc.     44.8    5
10 JetBlue Airways          39.7   20
11 Mesa Airlines Inc.        52.9  29.5
12 SkyWest Airlines Inc.     58     40
13 Southwest Airlines Co.    34.8   15
14 US Airways Inc.          32.9   16
15 United Air Lines Inc.     29.8   12
16 Virgin America           34.2   10

```

```
> flights %>% filter(dep_delay>0) %>% summarise(mean=mean(dep_delay))
# A tibble: 1 x 1
  mean
  <dbl>
1 39.4
```

View(People) : Bu kod people verisini gösterir.

playerID	birthYear	birthMonth	birthDay	birthCity	birthCountry	birthState	deathYear	deathMonth	deathDay	deathCountry
1 aardsda01	1981	12	27	Denver	USA	CO	NA	NA	NA	NA
2 aaronha01	1934	2	5	Mobile	USA	AL	2021	1	22	USA
3 aaronto01	1939	8	5	Mobile	USA	AL	1984	8	16	USA
4 aasedo01	1954	9	8	Orange	USA	CA	NA	NA	NA	NA
5 abadan01	1972	8	25	Palm Beach	USA	FL	NA	NA	NA	NA
6 abadfe01	1985	12	17	La Romana	D.R.	La Romana	NA	NA	NA	NA
7 abadijo01	1850	11	4	Philadelphia	USA	PA	1905	5	17	USA
8 abbated01	1877	4	15	Latrobe	USA	PA	1957	1	6	USA
9 abbeybe01	1869	11	11	Essex	USA	VT	1962	6	11	USA
10 abbeych01	1866	10	14	Falls City	USA	NE	1926	4	27	USA
11 abbotco01	1995	9	20	San Diego	USA	CA	NA	NA	NA	NA
12 abbotda01	1862	3	16	Portage	USA	OH	1930	2	13	USA
13 abbotfr01	1874	10	22	Versailles	USA	OH	1935	6	11	USA
14 abbotgl01	1951	2	16	Little Rock	USA	AR	NA	NA	NA	NA
15 abbotje01	1972	8	17	Atlanta	USA	GA	NA	NA	NA	NA
16 abbotji01	1967	9	19	Flint	USA	MI	NA	NA	NA	NA
17 abbotku01	1969	6	2	Zanesville	USA	OH	NA	NA	NA	NA
18 abbotky01	1968	2	18	Newburyport	USA	MA	NA	NA	NA	NA
19 abbotod01	1886	9	5	New Eagle	USA	PA	1933	4	13	USA
20 abbotpa01	1967	9	15	Van Nuys	USA	CA	NA	NA	NA	NA
21 aberal01	1927	7	31	Cleveland	USA	OH	1993	5	20	USA
22 abercda01	1850	1	2	Fort Towson	USA	OK	1939	11	11	USA
23 sharrca01	1980	7	15	Columbus	USA	CA	NA	NA	NA	NA

Showing 1 to 22 of 21,010 entries, 26 total columns

Oyuncuların ödülleri sayıları hesaplanır ve en çok ödül alan oyuncu bulunur.

```
> t<-table(AwardsPlayers$playerID) %>% as.data.frame()
> names(t)<-c("playerID","num.of.aw")
> data1<-People %>% left_join(t,by="playerID") # hepsine na.omit() uygulamak?
> data1 %>% filter(num.of.aw==max(num.of.aw,na.rm=T)) %>%
+   select(nameFirst,nameLast)
  nameFirst nameLast
1   Barry   Bonds
```

Verilen aralıktaki oyuncuları listeler.

```
> People %>% mutate(BMI=weight/(height^2)*703) %>% filter(BMI>=25 & BMI<29.9) %>% nrow()
[1] 9719
```

En yüksek maaş alan oyuncunun bilgilerini gösterir.

```
> slice(data2,which.max(data2$salary)) #max salary
  yearID teamID lgID playerID salary num.of.aw
1  2009   NYA   AL rodrial01 33000000      31
```

En fazla ödül alan oyuncunun bilgilerini gösterir.

```
> slice(data2,which.max(data2$num.of.aw)) #max num of awards
  yearID teamID lgID playerID salary num.of.aw
1   1986   PIT   NL bondsba01  60000         47

~

> table(AwardsPlayers$awardID) %>% as.data.frame() %>% arrange(desc(Freq))
  Var1 Freq
1      TSN All-Star 1525
2  Baseball Magazine All-Star 1520
3      Gold Glove 1204
4      Silver Slugger 792
5    Most Valuable Player 208
6    Rookie of the Year 154
7    TSN Pitcher of the Year 151
8      Cy Young Award 126
9  Reliever of the Year Award 94
10     TSN Player of the Year 92
11 TSN Major League Player of the Year 89
12     TSN Fireman of the Year 88
13     Babe Ruth Award 78
14     World Series MVP 71
15    Lou Gehrig Memorial Award 69
16     All-Star Game MVP 62
17      Hutch Award 55
18    Roberto Clemente Award 54
19     Hank Aaron Award 50
20      NLCS MVP 49
21     ALCS MVP 43
22    Pitching Triple Crown 39
23    Comeback Player of the Year 36
24    TSN Player of the Year 36

> diamonds %>% mutate(t=x^2-sqrt(y)+(1/z)) %>% filter(t==min(t)) %>% select(depth)
# A tibble: 1 x 1
  depth
  <dbl>
1  62.6
```

Filtrelenme işlemiyle belirli bir renk ve berraklık seviyesindeki en düşük fiyatı bulur.

```

> diamonds %>% mutate(discount = case_when(
+   cut=="Fair" ~ price*0.01,
+   cut=="Good" ~ price*0.02,
+   cut=="Very Good" ~ price*0.025,
+   cut=="Premium" ~ price*0.03,
+   cut=="Ideal" ~ price*0.03,
+ ),new.price=price-discount) %>% filter(color=="E" & clarity=="SI2" & new.price==min(new.price))
# A tibble: 1 x 12
  carat cut    color clarity depth table price     x     y     z discount new.price
  <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>   <dbl>   <dbl>
1  0.23 Ideal E      SI2     61.5   55   326  3.95  3.98  2.43    9.78    316.

> diamonds %>% mutate(discount= case_when(
+   cut=="Fair" ~ price*0.1,
+   cut=="Good" ~ price*0.12,
+   cut=="Very Good" ~ price*0.15,
+   cut=="Premium" ~ price*0.18,
+   cut=="Ideal" ~ price*0.18,
+ ),new.price=price-discount) %>% filter(color=="E"& clarity=="SI1") %>% arrange(new.price)
# A tibble: 2,426 x 12
  carat cut    color clarity depth table price     x     y     z discount new.price
  <dbl> <ord> <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>   <dbl>   <dbl>
1  0.21 Premium E      SI1     59.8   61   326  3.89  3.84  2.31    58.7    267.
2  0.26 Very Good E      SI1     62     54   384  4.08  4.11  2.54    57.6    326.
3  0.28 Good    E      SI1     62.6   60   373  4.12  4.15  2.59    44.8    328.
4  0.27 Very Good E      SI1     61.2   57   407  4.2   4.23  2.58    61.0    346.
5  0.31 Premium E      SI1     62.7   58   471  4.32  4.35  2.72    84.8    386.
6  0.24 Ideal   E      SI1     62.5   56   486  4.01  3.99  2.5     87.5    399.
7  0.3 Ideal   E      SI1     62.1   58   499  4.27  4.29  2.66    89.8    409.
8  0.3 Ideal   E      SI1     61.1   57   499  4.3   4.34  2.64    89.8    409.
9  0.3 Ideal   E      SI1     62.4   57   499  4.27  4.29  2.67    89.8    409.
10 0.3 Ideal   E      SI1     61.6   54   499  4.32  4.35  2.67    89.8    409.
# i 2,416 more rows
# i Use `print(n = ...)` to see more rows

```

```

> cl<-diamonds$clarity %>% as.factor()
> cl %>% levels()
[1] "I1"    "SI2"   "SI1"   "VS2"   "VS1"   "VVS2"  "VVS1"  "IF"

```

```

> cl %>% nlevels()
[1] 8

```

Elmasları kesim türüne göre gruplar ve her kesim türü için ortalama fiyatı hesaplar.

```

> diamonds %>% group_by(cut) %>% summarise(mean.pr=mean(price))
# A tibble: 5 x 2
  cut    mean.pr
  <ord>   <dbl>
1 Fair    4359.
2 Good    3929.
3 Very Good 3982.
4 Premium 4584.
5 Ideal    3458.

```

```
> diamonds %>% filter(cut=="Ideal") %>% summarise(mean(price))  
# A tibble: 1 × 1  
  `mean(price)`  
    <dbl>  
1      3458.
```