# Data Manipulation using dplyr

*Setia*

## Introduction

dplyr adalah package R yang bisa digunakan untuk menangani data terstruktur, yang dikembangkan oleh Hadley Wickham. dplyr sangat powerful dalam manipulasi dan eksplorasi data, dengan menggunakan dplyr, manipulasi data menjadi lebih mudah bagi pengguna R, antara lain dalam melakukan hal-hal sebagai berikut:

a. Select, filter, dan aggregate data
b. Menggunakan window functions
c. Melakukan join pada dataframes
d. Collect data sumber lain ke dalam R

Paket dplyr berisi seperangkat fungsi (atau "verbs") yang melakukan operasi manipulasi data umum seperti filtering untuk baris, memilih kolom tertentu, mengurutkan ulang baris, menambahkan kolom baru dan meringkas data. Selain itu, dplyr berisi fungsi yang berguna untuk melakukan tugas umum lainnya yaitu konsep "split-apply-combine".

Beberapa Fungsi Penting dalam dplyr Berikut ini adalah beberapa verbs (perintah) penting dalam dplyr yang paling sering digunakan:

1. select(): Selecting columns (variables), SELECT
2. filter(): Filter (subset) rows/picks cases based on their values, WHERE
3. group_by(): Group the data, GROUP BY
4. summarise() Summarise (or aggregate) data: reduces multiple values down to a single summary. -
5. arrange(): Sort the data/changes the ordering of the rows ORDER BY
6. join(): Joining data frames (tables) JOIN
7. mutate(): Creating New Variables COLUMN ALIAS

Selain beberapa fungsi penting di atas, terdapat banyak fungsi lain yang juga cukup sering digunakan, antara lain: distinct, anti_join, as.tbl, dan sebagainya.

## Examples

Pada bagian ini, kita akan mendemonstrasikan beberapa penggunaan verbs di dalam dplyr untuk melakukan manipulasi data dengan menggunakan data set airlines dan flight yang diambil dari package nycflights13. Package tersebut berisi data lengkap untuk seluruh 336.776 penerbangan yang berangkat dari kota New York selama tahun 2013. Data tersebut berasal dari US Bureau of Transportation Statistics.

```
#install.packages("nycflights13")
library("nycflights13")
```

```
## Warning: package 'nycflights13' was built under R version 3.4.4
```

```
data(flights)
head(flights)
```

```
## # A tibble: 6 x 19
##    year month   day dep_t~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##   <int> <int> <int>  <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
```

```
## 1  2013     1     1    517   515  2.00   830   819  11.0 UA     1545 N142~
## 2  2013     1     1    533   529  4.00   850   830  20.0 UA     1714 N242~
## 3  2013     1     1    542   540  2.00   923   850  33.0 AA     1141 N619~
## 4  2013     1     1    544   545 -1.00  1004  1022 -18.0 B6      725 N804~
## 5  2013     1     1    554   600 -6.00   812   837 -25.0 DL      461 N668~
## 6  2013     1     1    554   558 -4.00   740   728  12.0 UA     1696 N394~
## # ... with 7 more variables: origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```

**select**

Perintah select digunakan untuk memilih beberapa kolom/variabel dalam suatu tabel atau data set. Penggunaannya dalam R adalah sebagai berikut

select() syntax : select(data , ....) data : Data Frame .... : Variables by name or by function

Isikan bagian ( . . . ) dengan suatu daftar nama variabel/kolom yang akan dipilih, tanpa tanda kurung dan tanda petik dengan dipisahkan oleh tanda koma. Gunakan simbol minus ( - ) untuk membuang kolom/variabel. Contoh: Dari data flights, pilihlah variabel atau kolom berikut: year, month, dat, arr_delay dan dep_delay. Untuk melakukannya, ketik perintah dengan menggunakan verb select berikut:

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
sample_n(flights,10)
```

```
## # A tibble: 10 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     4    22   1653    1700 - 7.00  1842   1819  23.0  US     2136
##  2  2013     5    15   1057    1100 - 3.00  1237   1303 -26.0  EV     5596
##  3  2013    12    10   2101    2059   2.00  2340   2314  26.0  MQ     3473
##  4  2013     1    14   1853    1900 - 7.00  2135   2146 -11.0  DL      947
##  5  2013     2    27   1934    1845  49.0   2121   2058  23.0  DL     2131
##  6  2013    11    11    628     630 - 2.00   901    854   7.00 DL      575
##  7  2013     1    15    656     705 - 9.00  1012    940  32.0  MQ     4534
##  8  2013     1     3   1625    1629 - 4.00  1811   1806   5.00 EV     4645
##  9  2013     4    10   2105    2030  35.0   2301   2150  71.0  WN      184
## 10  2013     2    25   1202    1207 - 5.00  1509   1503   6.00 UA     1461
## # ... with 8 more variables: tailnum <chr>, origin <chr>, dest <chr>,
## #   air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour
## #   <dttm>
```

```
## select coloumn ##

flights[,which(names(flights)==c("year","month", "day", "arr_delay","dep_delay"))]
```

```
## Warning in names(flights) == c("year", "month", "day", "arr_delay",
## "dep_delay"): longer object length is not a multiple of shorter object
## length
```

```
## # A tibble: 336,776 x 4
##     year month   day arr_delay
##    <int> <int> <int>     <dbl>
##  1  2013     1     1     11.0
##  2  2013     1     1     20.0
##  3  2013     1     1     33.0
##  4  2013     1     1    -18.0
##  5  2013     1     1    -25.0
##  6  2013     1     1     12.0
##  7  2013     1     1     19.0
##  8  2013     1     1    -14.0
##  9  2013     1     1    - 8.00
## 10  2013     1     1      8.00
## # ... with 336,766 more rows
```

```
## atau
flights[,c("year","month", "day", "arr_delay","dep_delay")]
```

```
## # A tibble: 336,776 x 5
##     year month   day arr_delay dep_delay
##    <int> <int> <int>     <dbl>     <dbl>
##  1  2013     1     1     11.0       2.00
##  2  2013     1     1     20.0       4.00
##  3  2013     1     1     33.0       2.00
##  4  2013     1     1    -18.0      -1.00
##  5  2013     1     1    -25.0      -6.00
##  6  2013     1     1     12.0      -4.00
##  7  2013     1     1     19.0      -5.00
##  8  2013     1     1    -14.0      -3.00
##  9  2013     1     1    - 8.00     -3.00
## 10  2013     1     1      8.00     -2.00
## # ... with 336,766 more rows
```

```
# Using  dplyr:
select(flights, year,month, day, arr_delay, dep_delay)
```

```
## # A tibble: 336,776 x 5
##     year month   day arr_delay dep_delay
##    <int> <int> <int>     <dbl>     <dbl>
##  1  2013     1     1     11.0       2.00
##  2  2013     1     1     20.0       4.00
##  3  2013     1     1     33.0       2.00
##  4  2013     1     1    -18.0      -1.00
##  5  2013     1     1    -25.0      -6.00
##  6  2013     1     1     12.0      -4.00
##  7  2013     1     1     19.0      -5.00
##  8  2013     1     1    -14.0      -3.00
##  9  2013     1     1    - 8.00     -3.00
## 10  2013     1     1      8.00     -2.00
## # ... with 336,766 more rows
```

```r
# or
select(flights, year:day, arr_delay, dep_delay)
```

```
## # A tibble: 336,776 x 5
##     year month   day arr_delay dep_delay
##    <int> <int> <int>     <dbl>     <dbl>
##  1  2013     1     1     11.0       2.00
##  2  2013     1     1     20.0       4.00
##  3  2013     1     1     33.0       2.00
##  4  2013     1     1    -18.0      -1.00
##  5  2013     1     1    -25.0      -6.00
##  6  2013     1     1     12.0      -4.00
##  7  2013     1     1     19.0      -5.00
##  8  2013     1     1    -14.0      -3.00
##  9  2013     1     1    - 8.00     -3.00
## 10  2013     1     1      8.00     -2.00
## # ... with 336,766 more rows
```

```r
## drop Variable

newdata = select(flights, -arr_delay, -c(year:day))
head(newdata)
```

```
## # A tibble: 6 x 15
##   dep_t~ sche~ dep_~ arr_~ sche~ carr~ flig~ tail~ orig~ dest  air_~ dist~
##    <int> <int> <dbl> <int> <int> <chr> <int> <chr> <chr> <chr> <dbl> <dbl>
## 1    517   515  2.00   830   819 UA     1545 N142~ EWR   IAH     227  1400
## 2    533   529  4.00   850   830 UA     1714 N242~ LGA   IAH     227  1416
## 3    542   540  2.00   923   850 AA     1141 N619~ JFK   MIA     160  1089
## 4    544   545 -1.00  1004  1022 B6      725 N804~ JFK   BQN     183  1576
## 5    554   600 -6.00   812   837 DL      461 N668~ LGA   ATL     116   762
## 6    554   558 -4.00   740   728 UA     1696 N394~ EWR   ORD     150   719
## # ... with 3 more variables: hour <dbl>, minute <dbl>, time_hour <dttm>
```

```r
newdata = select(flights, -c(arr_delay,year:day))
head(newdata)
```

```
## # A tibble: 6 x 15
##   dep_t~ sche~ dep_~ arr_~ sche~ carr~ flig~ tail~ orig~ dest  air_~ dist~
##    <int> <int> <dbl> <int> <int> <chr> <int> <chr> <chr> <chr> <dbl> <dbl>
## 1    517   515  2.00   830   819 UA     1545 N142~ EWR   IAH     227  1400
## 2    533   529  4.00   850   830 UA     1714 N242~ LGA   IAH     227  1416
## 3    542   540  2.00   923   850 AA     1141 N619~ JFK   MIA     160  1089
## 4    544   545 -1.00  1004  1022 B6      725 N804~ JFK   BQN     183  1576
## 5    554   600 -6.00   812   837 DL      461 N668~ LGA   ATL     116   762
## 6    554   558 -4.00   740   728 UA     1696 N394~ EWR   ORD     150   719
## # ... with 3 more variables: hour <dbl>, minute <dbl>, time_hour <dttm>
```

```r
newdata = select(flights, starts_with("Y"))
head(newdata)
```

```
## # A tibble: 6 x 1
##    year
##   <int>
## 1  2013
## 2  2013
```

```
## 3  2013
## 4  2013
## 5  2013
## 6  2013
```

```
head(select(flights, starts_with("arr")))
```

```
## # A tibble: 6 x 2
##   arr_time arr_delay
##      <int>     <dbl>
## 1      830      11.0
## 2      850      20.0
## 3      923      33.0
## 4     1004     -18.0
## 5      812     -25.0
## 6      740      12.0
```

```
head(select(flights, -starts_with("arr")))
```

```
## # A tibble: 6 x 17
##    year month   day dep_t~ sche~ dep_~ sche~ carr~ flig~ tail~ orig~ dest
##   <int> <int> <int>  <int> <int> <dbl> <int> <chr> <int> <chr> <chr> <chr>
## 1  2013     1     1    517   515  2.00   819 UA     1545 N142~ EWR   IAH
## 2  2013     1     1    533   529  4.00   830 UA     1714 N242~ LGA   IAH
## 3  2013     1     1    542   540  2.00   850 AA     1141 N619~ JFK   MIA
## 4  2013     1     1    544   545 -1.00  1022 B6      725 N804~ JFK   BQN
## 5  2013     1     1    554   600 -6.00   837 DL      461 N668~ LGA   ATL
## 6  2013     1     1    554   558 -4.00   728 UA     1696 N394~ EWR   ORD
## # ... with 5 more variables: air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>
```

```
head(select(flights, -contains("time")))
```

```
## # A tibble: 6 x 13
##    year month   day dep_d~ arr_~ carr~ flig~ tail~ orig~ dest  dist~  hour
##   <int> <int> <int>  <dbl> <dbl> <chr> <int> <chr> <chr> <chr> <dbl> <dbl>
## 1  2013     1     1   2.00  11.0 UA     1545 N142~ EWR   IAH    1400  5.00
## 2  2013     1     1   4.00  20.0 UA     1714 N242~ LGA   IAH    1416  5.00
## 3  2013     1     1   2.00  33.0 AA     1141 N619~ JFK   MIA    1089  5.00
## 4  2013     1     1  -1.00 -18.0 B6      725 N804~ JFK   BQN    1576  5.00
## 5  2013     1     1  -6.00 -25.0 DL      461 N668~ LGA   ATL     762  6.00
## 6  2013     1     1  -4.00  12.0 UA     1696 N394~ EWR   ORD     719  5.00
## # ... with 1 more variable: minute <dbl>
```

```
# rename

head(rename(flights, bulan=month))
```

```
## # A tibble: 6 x 19
##    year bulan   day dep_t~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##   <int> <int> <int>  <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
## 1  2013     1     1    517   515  2.00   830   819  11.0 UA     1545 N142~
## 2  2013     1     1    533   529  4.00   850   830  20.0 UA     1714 N242~
## 3  2013     1     1    542   540  2.00   923   850  33.0 AA     1141 N619~
## 4  2013     1     1    544   545 -1.00  1004  1022 -18.0 B6      725 N804~
## 5  2013     1     1    554   600 -6.00   812   837 -25.0 DL      461 N668~
## 6  2013     1     1    554   558 -4.00   740   728  12.0 UA     1696 N394~
```

```
## # ... with 7 more variables: origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```

Others: starts_with(): Starts with a prefix ends_with(): Ends with a prefix contains(): Contains a literal string matches(): Matches a regular expression num_range(): Numerical range like x01, x02, x03. one_of(): Variables in character vector. everything(): All variables.

### Filter

Filter digunakan untuk memilih baris atau cases pada suatu tabel atau data frame dengan kondisi yang ditentukan. Penggunaannya dalam R adalah sebagai berikut: filter(.data, . . . )

Isilah bagian ( . . . ) dengan logic atau kondisi yang diinginkan dengan menggunakan operator logika.

Contoh: Dari data flights, tampilkan hanya data penerbangan yang mengalami keterlambatan berangkat (dep_delay) lebih dari 1000 menit.

```
flights[which(flights$dep_delay > 1000),]
```

```
## # A tibble: 5 x 19
##    year month   day dep_t~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##   <int> <int> <int>  <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
## 1  2013     1     9    641   900  1301  1242  1530  1272 HA       51 N384~
## 2  2013     1    10   1121  1635  1126  1239  1810  1109 MQ     3695 N517~
## 3  2013     6    15   1432  1935  1137  1607  2120  1127 MQ     3535 N504~
## 4  2013     7    22    845  1600  1005  1044  1815   989 MQ     3075 N665~
## 5  2013     9    20   1139  1845  1014  1457  2210  1007 AA      177 N338~
## # ... with 7 more variables: origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```

```
# other examples #

filter(flights, dep_delay > 1000)
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
## # A tibble: 5 x 19
##    year month   day dep_t~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##   <int> <int> <int>  <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
## 1  2013     1     9    641   900  1301  1242  1530  1272 HA       51 N384~
## 2  2013     1    10   1121  1635  1126  1239  1810  1109 MQ     3695 N517~
## 3  2013     6    15   1432  1935  1137  1607  2120  1127 MQ     3535 N504~
## 4  2013     7    22    845  1600  1005  1044  1815   989 MQ     3075 N665~
## 5  2013     9    20   1139  1845  1014  1457  2210  1007 AA      177 N338~
## # ... with 7 more variables: origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
```

```
filter(flights, origin == "JFK")
```

```
## # A tibble: 111,279 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    542     540   2.00   923    850  33.0  AA     1141
##  2  2013     1     1    544     545 - 1.00  1004   1022 -18.0  B6      725
##  3  2013     1     1    557     600 - 3.00   838    846 - 8.00 B6       79
##  4  2013     1     1    558     600 - 2.00   849    851 - 2.00 B6       49
##  5  2013     1     1    558     600 - 2.00   853    856 - 3.00 B6       71
```

```
##  6  2013     1     1    558     600 - 2.00   924    917   7.00 UA      194
##  7  2013     1     1    559     559   0      702    706 - 4.00 B6     1806
##  8  2013     1     1    606     610 - 4.00   837    845 - 8.00 DL     1743
##  9  2013     1     1    611     600  11.0    945    931  14.0  UA      303
## 10  2013     1     1    613     610   3.00   925    921   4.00 B6      135
## # ... with 111,269 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
filter(flights, origin %in% c("JFK","EWR") )
```

```
## # A tibble: 232,114 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    517     515   2.00   830    819  11.0  UA     1545
##  2  2013     1     1    542     540   2.00   923    850  33.0  AA     1141
##  3  2013     1     1    544     545  -1.00  1004   1022 -18.0  B6      725
##  4  2013     1     1    554     558  -4.00   740    728  12.0  UA     1696
##  5  2013     1     1    555     600  -5.00   913    854  19.0  B6      507
##  6  2013     1     1    557     600  -3.00   838    846 - 8.00 B6       79
##  7  2013     1     1    558     600  -2.00   849    851 - 2.00 B6       49
##  8  2013     1     1    558     600  -2.00   853    856 - 3.00 B6       71
##  9  2013     1     1    558     600  -2.00   924    917   7.00 UA      194
## 10  2013     1     1    558     600  -2.00   923    937 -14.0  UA     1124
## # ... with 232,104 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
filter(flights, !origin %in% c("JFK","EWR") )
```

```
## # A tibble: 104,662 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    533     529   4.00   850    830  20.0  UA     1714
##  2  2013     1     1    554     600 - 6.00   812    837 -25.0  DL      461
##  3  2013     1     1    557     600 - 3.00   709    723 -14.0  EV     5708
##  4  2013     1     1    558     600 - 2.00   753    745   8.00 AA      301
##  5  2013     1     1    559     600 - 1.00   941    910  31.0  AA      707
##  6  2013     1     1    600     600   0      851    858 - 7.00 B6      371
##  7  2013     1     1    600     600   0      837    825  12.0  MQ     4650
##  8  2013     1     1    602     610 - 8.00   812    820 - 8.00 DL     1919
##  9  2013     1     1    602     605 - 3.00   821    805  16.0  MQ     4401
## 10  2013     1     1    623     610  13.0    920    915   5.00 AA     1837
## # ... with 104,652 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
filter(flights, origin %in% c("JFK","EWR") & month >5 )
```

```
## # A tibble: 135,677 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013    10     1    447     500 -13.0    614    648 -34.0  US     1877
##  2  2013    10     1    522     517   5.00   735    757 -22.0  UA      252
##  3  2013    10     1    536     545 - 9.00   809    855 -46.0  AA     2243
##  4  2013    10     1    539     545 - 6.00   917    933 -16.0  B6     1403
```

```
##  5  2013    10     1    544     550 - 6.00   912    932 -20.0  B6      939
##  6  2013    10     1    549     600 -11.0    653    716 -23.0  EV     5716
##  7  2013    10     1    551     600 - 9.00   727    730 - 3.00 UA      279
##  8  2013    10     1    551     600 - 9.00   655    708 -13.0  B6     2180
##  9  2013    10     1    553     600 - 7.00   829    856 -27.0  B6      601
## 10  2013    10     1    554     600 - 6.00   757    843 -46.0  UA     1014
## # ... with 135,667 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
filter(flights, origin %in% c("JFK","EWR") | carrier =="UA" )
```

```
## # A tibble: 240,158 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    517     515   2.00   830    819  11.0  UA     1545
##  2  2013     1     1    533     529   4.00   850    830  20.0  UA     1714
##  3  2013     1     1    542     540   2.00   923    850  33.0  AA     1141
##  4  2013     1     1    544     545  -1.00  1004   1022 -18.0  B6      725
##  5  2013     1     1    554     558  -4.00   740    728  12.0  UA     1696
##  6  2013     1     1    555     600  -5.00   913    854  19.0  B6      507
##  7  2013     1     1    557     600  -3.00   838    846 - 8.00 B6       79
##  8  2013     1     1    558     600  -2.00   849    851 - 2.00 B6       49
##  9  2013     1     1    558     600  -2.00   853    856 - 3.00 B6       71
## 10  2013     1     1    558     600  -2.00   924    917   7.00 UA      194
## # ... with 240,148 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
 filter(flights, grepl("JB", tailnum))
```

```
## # A tibble: 54,691 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    544     545  -1.00  1004   1022 -18.0  B6      725
##  2  2013     1     1    555     600  -5.00   913    854  19.0  B6      507
##  3  2013     1     1    557     600  -3.00   838    846 - 8.00 B6       79
##  4  2013     1     1    558     600  -2.00   849    851 - 2.00 B6       49
##  5  2013     1     1    558     600  -2.00   853    856 - 3.00 B6       71
##  6  2013     1     1    559     559   0      702    706 - 4.00 B6     1806
##  7  2013     1     1    600     600   0      851    858 - 7.00 B6      371
##  8  2013     1     1    601     600   1.00   844    850 - 6.00 B6      343
##  9  2013     1     1    613     610   3.00   925    921   4.00 B6      135
## 10  2013     1     1    615     615   0     1039   1100 -21.0  B6      709
## # ... with 54,681 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

## Arrange

Perintah arrange digunakan untuk melakukan pengurutan (sorting) dari cases baik secara ascending ataupun descending. Penggunaannya dalam R adalah sebagai berikut:

arrange(.data, . . . )

Isilah bagian ( . . . ) dengan suatu daftar nama variabel yang akan menjadi dasar pengurutan, secara default, pengurutan dilakukan secara ascending, jika descending maka gunakan desc.

Contoh: Dari data flights, urutkanlah data penerbangan secara menurun (descending) berdasarkan variabel dep_delay.

```
flights[order(-flights$dep_delay),]
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##    <int> <int> <int> <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
##  1  2013     1     9   641   900  1301  1242  1530  1272 HA       51 N384~
##  2  2013     6    15  1432  1935  1137  1607  2120  1127 MQ     3535 N504~
##  3  2013     1    10  1121  1635  1126  1239  1810  1109 MQ     3695 N517~
##  4  2013     9    20  1139  1845  1014  1457  2210  1007 AA      177 N338~
##  5  2013     7    22   845  1600  1005  1044  1815   989 MQ     3075 N665~
##  6  2013     4    10  1100  1900   960  1342  2211   931 DL     2391 N959~
##  7  2013     3    17  2321   810   911   135  1020   915 DL     2119 N927~
##  8  2013     6    27   959  1900   899  1236  2226   850 DL     2007 N376~
##  9  2013     7    22  2257   759   898   121  1026   895 DL     2047 N671~
## 10  2013    12     5   756  1700   896  1058  2020   878 AA      172 N5DM~
## # ... with 336,766 more rows, and 7 more variables: origin <chr>, dest
## #   <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   time_hour <dttm>
```

```
# dplyr:

 arrange(flights, dep_time)
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_t~ sched~ dep_de~ arr_~ sche~ arr_de~ carr~ flig~
##    <int> <int> <int>  <int>  <int>   <dbl> <int> <int>   <dbl> <chr> <int>
##  1  2013     1    13      1   2249   72.0    108  2357   71.0  B6       22
##  2  2013     1    31      1   2100  181      124  2225  179    WN      530
##  3  2013    11    13      1   2359    2.00   442   440    2.00 B6     1503
##  4  2013    12    16      1   2359    2.00   447   437   10.0  B6      839
##  5  2013    12    20      1   2359    2.00   430   440 - 10.0  B6     1503
##  6  2013    12    26      1   2359    2.00   437   440 -  3.00 B6     1503
##  7  2013    12    30      1   2359    2.00   441   437    4.00 B6      839
##  8  2013     2    11      1   2100  181      111  2225  166    WN      530
##  9  2013     2    24      1   2245   76.0    121  2354   87.0  B6      608
## 10  2013     3     8      1   2355    6.00   431   440 -  9.00 B6      739
## # ... with 336,766 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
 arrange(flights, desc(dep_time))
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_t~ sched~ dep_de~ arr_~ sche~ arr_de~ carr~ flig~
##    <int> <int> <int>  <int>  <int>   <dbl> <int> <int>   <dbl> <chr> <int>
##  1  2013    10    30   2400   2359    1.00   327   337 - 10.0  B6      839
##  2  2013    11    27   2400   2359    1.00   515   445   30.0  B6      745
##  3  2013    12     5   2400   2359    1.00   427   440 - 13.0  B6     1503
##  4  2013    12     9   2400   2359    1.00   432   440 -  8.00 B6     1503
##  5  2013    12     9   2400   2250   70.0     59  2356   63.0  B6     1816
##  6  2013    12    13   2400   2359    1.00   432   440 -  8.00 B6     1503
```

```
##  7  2013    12    19   2400   2359    1.00   434   440 -  6.00 B6     1503
##  8  2013    12    29   2400   1700  420      302  2025  397    AA     2379
##  9  2013     2     7   2400   2359    1.00   432   436 -  4.00 B6      727
## 10  2013     2     7   2400   2359    1.00   443   444 -  1.00 B6      739
## # ... with 336,766 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
arrange(flights, dep_time, dep_delay)
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013    11    13      1    2359   2.00   442    440   2.00 B6     1503
##  2  2013    12    16      1    2359   2.00   447    437  10.0  B6      839
##  3  2013    12    20      1    2359   2.00   430    440 -10.0  B6     1503
##  4  2013    12    26      1    2359   2.00   437    440 - 3.00 B6     1503
##  5  2013    12    30      1    2359   2.00   441    437   4.00 B6      839
##  6  2013     4     5      1    2359   2.00   410    339  31.0  B6      727
##  7  2013     5    25      1    2359   2.00   336    341 - 5.00 B6      727
##  8  2013     6    20      1    2359   2.00   340    350 -10.0  B6      745
##  9  2013     7    27      1    2359   2.00   345    340   5.00 B6      839
## 10  2013     7    28      1    2359   2.00   423    350  33.0  B6      745
## # ... with 336,766 more rows, and 8 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>
```

```
arrange(flights, dep_time, desc(dep_delay))
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##    <int> <int> <int> <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
##  1  2013     4    10     1  1930 271     106  2101 245   UA     1703 N332~
##  2  2013     5    22     1  1935 266     154  2140 254   EV     4361 N272~
##  3  2013     6    24     1  1950 251     105  2130 215   AA      363 N546~
##  4  2013     7     1     1  2029 212     236  2359 157   B6      915 N653~
##  5  2013     1    31     1  2100 181     124  2225 179   WN      530 N550~
##  6  2013     2    11     1  2100 181     111  2225 166   WN      530 N231~
##  7  2013     3    18     1  2128 153     247  2355 172   B6       97 N760~
##  8  2013     6    25     1  2130 151     249    14 155   B6     1371 N607~
##  9  2013     2    24     1  2245  76.0   121  2354  87.0 B6      608 N216~
## 10  2013     1    13     1  2249  72.0   108  2357  71.0 B6       22 N206~
## # ... with 336,766 more rows, and 7 more variables: origin <chr>, dest
## #   <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   time_hour <dttm>
```

## Summarise

Summarise digunakan untuk membuat suatu ringkasan atau membuat data agregat. Penggunaannya di R adalah sebagai berikut:

```
summarise(flights, mean_dep_delay= mean(dep_delay,na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   mean_dep_delay
```

```
##            <dbl>
## 1           12.6
```

```
res1 <- summarise(flights, mean_dep_delay= mean(dep_delay,na.rm = TRUE),
                  med_arr_time= median(arr_time,na.rm = TRUE))

summarise_at(flights, vars(air_time, distance), funs(n(), mean, median, sd))
```

```
## # A tibble: 1 x 8
##   air_time_n distance_n air_time_mean distan~ air_ti~ dista~ air_t~ dista~
##        <int>      <int>         <dbl>   <dbl>   <dbl>  <dbl>  <dbl>  <dbl>
## 1     336776     336776            NA    1040      NA    872    NaN    733
```

```
summarise_at(flights, vars(air_time, distance), funs(n(),
    missing = sum(is.na(.)), mean(., na.rm = TRUE),
    median(.,na.rm=T), sd(.,na.rm = T)))
```

```
## # A tibble: 1 x 10
##   air_time_n distance_n air_t~ dista~ air_t~ dist~ air_~ dist~ air_~ dist~
##        <int>      <int>  <int>  <int>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     336776     336776   9430      0    151  1040   129   872  93.7   733
```

```
summarise_at(flights,vars(distance), function(x) var(x - mean(x, na.rm=T)))
```

```
## # A tibble: 1 x 1
##   distance
##      <dbl>
## 1   537631
```

```
summarise_if(flights, is.numeric, funs(n(),
     mean(., na.rm=T),median(., na.rm=T)))
```

```
## # A tibble: 1 x 42
##   year_n month_n  day_n dep_ti~ sched_~ dep_d~ arr_t~ sched~ arr_d~ fligh~
##    <int>   <int>  <int>   <int>   <int>  <int>  <int>  <int>  <int>  <int>
## 1 336776  336776 336776  336776  336776 336776 336776 336776 336776 336776
## # ... with 32 more variables: air_time_n <int>, distance_n <int>, hour_n
## #   <int>, minute_n <int>, year_mean <dbl>, month_mean <dbl>, day_mean
## #   <dbl>, dep_time_mean <dbl>, sched_dep_time_mean <dbl>, dep_delay_mean
## #   <dbl>, arr_time_mean <dbl>, sched_arr_time_mean <dbl>, arr_delay_mean
## #   <dbl>, flight_mean <dbl>, air_time_mean <dbl>, distance_mean <dbl>,
## #   hour_mean <dbl>, minute_mean <dbl>, year_median <dbl>, month_median
## #   <dbl>, day_median <dbl>, dep_time_median <int>, sched_dep_time_median
## #   <dbl>, dep_delay_median <dbl>, arr_time_median <int>,
## #   sched_arr_time_median <dbl>, arr_delay_median <dbl>, flight_median
## #   <dbl>, air_time_median <dbl>, distance_median <dbl>, hour_median
## #   <dbl>, minute_median <dbl>
```

### Mutate

Mutate digunakan untuk membuat variabel baru, yang bisa jadi merupakan hasil penghitungan berdasarkan variabel lama. Penggunaannya di R adalah sebagai berikut:

mutate(.data, . . . )

Pada bagian ( . . . ) diisi oleh suatu ekspresi aritmatika ataupun logika yang mendefinisikan variabel baru. Misalkan dari data flights, buatlah sebuah variabel baru yang berasal dari variabel air_time¬ dimana

ubahlah satuannya dari menit menjadi jam, tanpa merubah file asli.

```
 mutate(flights, speed=distance/hour)
```

```
## # A tibble: 336,776 x 20
##     year month   day dep_t~ sched_~ dep_d~ arr_~ sched~ arr_d~ carr~ flig~
##    <int> <int> <int>  <int>   <int>  <dbl> <int>  <int>  <dbl> <chr> <int>
##  1  2013     1     1    517     515   2.00   830    819  11.0  UA     1545
##  2  2013     1     1    533     529   4.00   850    830  20.0  UA     1714
##  3  2013     1     1    542     540   2.00   923    850  33.0  AA     1141
##  4  2013     1     1    544     545  -1.00  1004   1022 -18.0  B6      725
##  5  2013     1     1    554     600  -6.00   812    837 -25.0  DL      461
##  6  2013     1     1    554     558  -4.00   740    728  12.0  UA     1696
##  7  2013     1     1    555     600  -5.00   913    854  19.0  B6      507
##  8  2013     1     1    557     600  -3.00   709    723 -14.0  EV     5708
##  9  2013     1     1    557     600  -3.00   838    846 - 8.00 B6       79
## 10  2013     1     1    558     600  -2.00   753    745   8.00 AA      301
## # ... with 336,766 more rows, and 9 more variables: tailnum <chr>, origin
## #   <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute
## #   <dbl>, time_hour <dttm>, speed <dbl>
```

```
new_flights<-mutate(flights, air_time_hours = air_time / 60)
head(new_flights)
```

```
## # A tibble: 6 x 20
##    year month   day dep_t~ sche~ dep_~ arr_~ sche~ arr_~ carr~ flig~ tail~
##   <int> <int> <int>  <int> <int> <dbl> <int> <int> <dbl> <chr> <int> <chr>
## 1  2013     1     1    517   515  2.00   830   819  11.0 UA     1545 N142~
## 2  2013     1     1    533   529  4.00   850   830  20.0 UA     1714 N242~
## 3  2013     1     1    542   540  2.00   923   850  33.0 AA     1141 N619~
## 4  2013     1     1    544   545 -1.00  1004  1022 -18.0 B6      725 N804~
## 5  2013     1     1    554   600 -6.00   812   837 -25.0 DL      461 N668~
## 6  2013     1     1    554   558 -4.00   740   728  12.0 UA     1696 N394~
## # ... with 8 more variables: origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>,
## #   air_time_hours <dbl>
```

```
## Other examples


## Keep only the new variable
new_flights<-transmute(flights, air_time_hours = air_time / 60)
```

### Group_by

Group by digunakan untuk mengelompokkan data berdasarkan satu variabel kategori atau lebih. Biasanya digunakan dengan dikombinasikan dengan verbs lainnya seperti summarise dan mutate. Penggunaan dasar dalam R adalah sebagai berikut:

group_by(.data, . . . )

Dari data flights, misal kita ingin mengetahui rata-rata keterlambatan keberangkatan (dep_delay) secara bulanan.

```
dt2 = summarise_at(group_by(flights, origin), vars(arr_delay, dep_delay), funs(n(), mean(., na.rm = TRU
dt2
```

```
## # A tibble: 3 x 5
##   origin arr_delay_n dep_delay_n arr_delay_mean dep_delay_mean
##   <chr>        <int>       <int>          <dbl>          <dbl>
## 1 EWR         120835      120835           9.11           15.1
## 2 JFK         111279      111279           5.55           12.1
## 3 LGA         104662      104662           5.78           10.3
```

```
summarise(group_by(flights,carrier), mean_dep_delay = mean(dep_delay,na.rm = TRUE))
```

```
## # A tibble: 16 x 2
##    carrier mean_dep_delay
##    <chr>            <dbl>
##  1 9E               16.7
##  2 AA                8.59
##  3 AS                5.80
##  4 B6               13.0
##  5 DL                9.26
##  6 EV               20.0
##  7 F9               20.2
##  8 FL               18.7
##  9 HA                4.90
## 10 MQ               10.6
## 11 OO               12.6
## 12 UA               12.1
## 13 US                3.78
## 14 VX               12.9
## 15 WN               17.7
## 16 YV               19.0
```

**Join**

Perintah join() digunakan untuk menggabungkan dua dataset, baik dengan berdasarkan variable tertentu yang unik (seperti ID) atau tidak. Terdapat lima fungsi join:

inner_join(x, y, by = ) : menghasilan baris yang sesuai ID nya dari kedua set data.

left_join(x, y, by = ): mengembalikan semua baris dari tabel kiri, meskipun tidak ada yang cocok di tabel kanan.

right_join(x, y, by = ) : mengembalikan semua baris dari tabel kanan, meskipun tidak ada yang cocok di tabel kanan.

full_join(x, y, by = ) : mengembalikan semua baris dari kedua tabel, meskipun tidak ada yang cocok di tabel kanan.

anti_join(x, y, by = ): mengembalikan semua baris yang tidak cocok di kedua tabel

Contoh berikut menggunakan illustrasi data frame sebagai berikut:

```
df1 <- data.frame(ID = c(1, 2, 3, 4, 5),
                  w = c('a', 'b', 'c', 'd', 'e'),
                  x = c(1, 1, 0, 0, 1),
                  y=rnorm(5),
                  z=letters[1:5])
df2 <- data.frame(ID = c(1, 7, 3, 6, 8),
                  a = c('z', 'b', 'k', 'd', 'l'),
                  b = c(1, 2, 3, 0, 4),
```

```
                  c =rnorm(5),
                  d =letters[2:6])


df3 = inner_join(df1, df2, by = "ID")
df3
```

```
##   ID w x          y z a b          c d
## 1  1 a 1 -2.5514858 a z 1 -0.1409228 b
## 2  3 c 0 -0.8453897 c k 3  1.6116492 d
```

```
left_join(df1, df2, by = "ID")
```

```
##   ID w x          y z    a  b          c    d
## 1  1 a 1 -2.5514858 a    z  1 -0.1409228    b
## 2  2 b 1  0.2400879 b <NA> NA         NA <NA>
## 3  3 c 0 -0.8453897 c    k  3  1.6116492    d
## 4  4 d 0 -0.4762697 d <NA> NA         NA <NA>
## 5  5 e 1  0.1549297 e <NA> NA         NA <NA>
```

```
right_join(df1, df2, by = "ID")
```

```
##   ID    w  x          y    z a b          c d
## 1  1    a  1 -2.5514858    a z 1 -0.1409228 b
## 2  7 <NA> NA         NA <NA> b 2 -1.5034741 c
## 3  3    c  0 -0.8453897    c k 3  1.6116492 d
## 4  6 <NA> NA         NA <NA> d 0 -1.2045745 e
## 5  8 <NA> NA         NA <NA> l 4 -0.5560177 f
```

```
full_join(df1, df2, by = "ID")
```

```
##   ID    w  x          y    z    a  b          c    d
## 1  1    a  1 -2.5514858    a    z  1 -0.1409228    b
## 2  2    b  1  0.2400879    b <NA> NA         NA <NA>
## 3  3    c  0 -0.8453897    c    k  3  1.6116492    d
## 4  4    d  0 -0.4762697    d <NA> NA         NA <NA>
## 5  5    e  1  0.1549297    e <NA> NA         NA <NA>
## 6  7 <NA> NA         NA <NA>    b  2 -1.5034741    c
## 7  6 <NA> NA         NA <NA>    d  0 -1.2045745    e
## 8  8 <NA> NA         NA <NA>    l  4 -0.5560177    f
```

```
anti_join(df1, df2, by = "ID")
```

```
##   ID w x          y z
## 1  2 b 1  0.2400879 b
## 2  4 d 0 -0.4762697 d
## 3  5 e 1  0.1549297 e
```

## Piping

Paket magrittr adalah sebuah paket dengan dua tujuan, yaitu untuk mengurangi waktu pengembangan dan untuk meningkatkan keterbacaan dan kemampuan pemeliharaan kode. Paket magrittr menyediakan suatu fungsi baru yang disebut sebagai “pipa”, yaitu operator %>%, dimana dengan operator tersebut Anda dapat mengumpankan nilai ke depan menjadi sebuah ekspresi atau fungsi.

Contoh: Masih dengan data flights, kita ingin menampilkan penerbangan pada bulan ke 5 dan tanggal 17

untuk sebagain airlines/carrier ('UA', 'WN', 'AA', 'DL'). Variabel yang akan ditampilkan hanyalah carrier, dep_delay, air_time, dan distance. Data akan diurutkan sesuai dengan carrier nya serta dibuat sebuah variabel baru yang merupakan air time dalam jam air_time_hours. Berikut implementasinya:

```
new_flights2 <- flights %>%
  filter(month == 5, day == 17, carrier %in% c('UA', 'WN', 'AA', 'DL')) %>%
  select(carrier, dep_delay, air_time, distance) %>%
  arrange(carrier) %>%
  mutate(air_time_hours = air_time / 60)

head(new_flights2)
```

```
## # A tibble: 6 x 5
##   carrier dep_delay air_time distance air_time_hours
##   <chr>       <dbl>    <dbl>    <dbl>          <dbl>
## 1 AA          -7.00      142     1089           2.37
## 2 AA          -9.00      186     1389           3.10
## 3 AA          -6.00      143     1096           2.38
## 4 AA          -4.00      114      733           1.90
## 5 AA          -2.00      146     1085           2.43
## 6 AA          -7.00      119      733           1.98
```

```
dt = sample_n(select(flights, arr_time, carrier),10)

flights %>% select(arr_time, carrier)
```

```
## # A tibble: 336,776 x 2
##    arr_time carrier
##       <int> <chr>
##  1      830 UA
##  2      850 UA
##  3      923 AA
##  4     1004 B6
##  5      812 DL
##  6      740 UA
##  7      913 B6
##  8      709 EV
##  9      838 B6
## 10      753 AA
## # ... with 336,766 more rows
```

```
flights %>% select(arr_time, carrier) %>% sample_n(10)
```

```
## # A tibble: 10 x 2
##    arr_time carrier
##       <int> <chr>
##  1     1049 F9
##  2     1102 WN
##  3     2249 MQ
##  4     1550 UA
##  5     2356 EV
##  6     1025 DL
##  7     1459 EV
##  8     1128 WN
##  9     1818 UA
## 10      924 MQ
```

```r
dt = flights %>% group_by(carrier) %>%
  summarise_at(vars(dep_delay, air_time, distance), funs(n(), mean(., na.rm = TRUE)))
dt
```

```
## # A tibble: 16 x 7
##    carrier dep_delay_n air_time_n distance_n dep_delay_mean air_ti~ dista~
##    <chr>         <int>      <int>      <int>          <dbl>   <dbl>  <dbl>
##  1 9E            18460      18460      18460          16.7     86.8    530
##  2 AA            32729      32729      32729           8.59   189     1340
##  3 AS              714        714        714           5.80   326     2402
##  4 B6            54635      54635      54635          13.0    151     1069
##  5 DL            48110      48110      48110           9.26   174     1237
##  6 EV            54173      54173      54173          20.0     90.1    563
##  7 F9              685        685        685          20.2    230     1620
##  8 FL             3260       3260       3260          18.7    101      665
##  9 HA              342        342        342           4.90   623     4983
## 10 MQ            26397      26397      26397          10.6     91.2    570
## 11 OO               32         32         32          12.6     83.5    501
## 12 UA            58665      58665      58665          12.1    212     1529
## 13 US            20536      20536      20536           3.78    88.6    553
## 14 VX             5162       5162       5162          12.9    337     2499
## 15 WN            12275      12275      12275          17.7    148      996
## 16 YV              601        601        601          19.0     65.7    375
```