Introduction to Scientific Computing for Biologists

ISCB20.09 - Data Management with R

An Introduction to `dplyr`

Md. Jubayer Hossain
https://jhossain.me/
jubayer@hdrobd.org

Founder
Health Data Research Organization
Lead Organizer
Scientific Computing for Biologists

03 February 2021

What is `dplyr`?

- The `dplyr` package was developed by Hadley Wickham of RStudio.
- `dplyr` is a new package which provides a set of tools for efficiently manipulating datasets in R.
- `dplyr` is the next iteration of `plyr` , focussing on only data frames.
- With `dplyr` , anything you can do to a local data frame you can also do to a remote database table.

`dplyr` Functionality

- Five basic verbs: `filter`, `select`, `arrange`, `mutate`, `summarise` and `groub_by`
- Can work with data stored in databases and data tables
- Joins: inner join, left join, semi-join, anti-join
- Window functions for calculating ranking, offsets, and more
- Better than plyr if you're only working with data frames (though it doesn't yet duplicate all of the plyr functionality)

Why `dplyr`?

- Great for data exploration and transformation
- Intuitive to write and easy to read, especially when using the "chaining" syntax (covered below)
- Fast on data frames

`dplyr` Grammar

- `select`: return a subset of the column of a data franme, using a flexible notation.
- `filter`: extract a subset of rows from a data frame based on logical conditions.
- `arrange`: reorder rows of data frame
- `mutate`: add new variables/columns or transform existing variables.
- `summarise/summarize`: generate summary statistics of different variables in the data frame, possibly within strata.
- %>% "pipe" operator used to connect multiple verb actions together into a pipeline.

Installing dplyr

```
install.packages('dplyr')
```

Loading Data: The Gapminder Dataset

```r
# Install gapminder dataset
install.packages('gapminder')
```

```r
# Load gapminder dataset
library(gapminder)
```

Exploring the Gapminder Dataset

```
# Examine first few rows
head(gapminder)
```

```
# A tibble: 6 x 6
  country     continent  year lifeExp      pop gdpPercap
  <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
1 Afghanistan Asia       1952    28.8  8425333      779.
2 Afghanistan Asia       1957    30.3  9240934      821.
3 Afghanistan Asia       1962    32.0 10267083      853.
4 Afghanistan Asia       1967    34.0 11537966      836.
5 Afghanistan Asia       1972    36.1 13079460      740.
6 Afghanistan Asia       1977    38.4 14880372      786.
```

Exploring the Gapminder Dataset(Cont. . .)

```
# Examine last few rows
tail(gapminder)
```

```
# A tibble: 6 x 6
  country  continent  year lifeExp      pop gdpPercap
  <fct>    <fct>     <int>   <dbl>    <int>     <dbl>
1 Zimbabwe Africa     1982    60.4  7636524      789.
2 Zimbabwe Africa     1987    62.4  9216418      706.
3 Zimbabwe Africa     1992    60.4 10704340      693.
4 Zimbabwe Africa     1997    46.8 11404948      792.
5 Zimbabwe Africa     2002    40.0 11926563      672.
6 Zimbabwe Africa     2007    43.5 12311143      470.
```

Exploring the Gapminder Dataset(Cont. . .)

```
# Dimensions
dim(gapminder)
```

```
[1] 1704    6
```

```
# Colnames
names(gapminder)
```

```
[1] "country"   "continent" "year"      "lifeExp"   "pop"       "gdpPercap
```

Exploring the Gapminder Dataset(Cont. . .)

```
# Data Structures
str(gapminder)
```

```
tibble [1,704 x 6] (S3: tbl_df/tbl/data.frame)
 $ country  : Factor w/ 142 levels "Afghanistan",..: 1 1 1 1 1 1 1 1 1 1 .
 $ continent: Factor w/ 5 levels "Africa","Americas",..: 3 3 3 3 3 3 3 3 3
 $ year     : int [1:1704] 1952 1957 1962 1967 1972 1977 1982 1987 1992 19
 $ lifeExp  : num [1:1704] 28.8 30.3 32 34 36.1 ...
 $ pop      : int [1:1704] 8425333 9240934 10267083 11537966 13079460 1488
 $ gdpPercap: num [1:1704] 779 821 853 836 740 ...
```

Exploring the Gapminder Dataset(Cont. . .)

```
# Summary
summary(gapminder)
```

```
       country        continent        year         lifeExp
 Afghanistan:  12   Africa  :624   Min.   :1952   Min.   :23.60
 Albania    :  12   Americas:300   1st Qu.:1966   1st Qu.:48.20
 Algeria    :  12   Asia    :396   Median :1980   Median :60.71
 Angola     :  12   Europe  :360   Mean   :1980   Mean   :59.47
 Argentina  :  12   Oceania : 24   3rd Qu.:1993   3rd Qu.:70.85
 Australia  :  12                  Max.   :2007   Max.   :82.60
 (Other)    :1632
      pop              gdpPercap
 Min.   :6.001e+04   Min.   :   241.2
 1st Qu.:2.794e+06   1st Qu.:  1202.1
 Median :7.024e+06   Median :  3531.8
 Mean   :2.960e+07   Mean   :  7215.3
```

Command Structure (for all dplyr verbs)

- first argument is a data frame
- return value is a data frame
- nothing is modified in place
- Note: dplyr generally does not preserve row names

Load `dplyr` Package

```
# Load dplyr
library(dplyr)
```

```
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
```

`select`: Pick Single Column by Name

```
select(gapminder, country)
```

```
# A tibble: 1,704 x 1
   country
   <fct>
 1 Afghanistan
 2 Afghanistan
 3 Afghanistan
 4 Afghanistan
 5 Afghanistan
 6 Afghanistan
 7 Afghanistan
 8 Afghanistan
 9 Afghanistan
10 Afghanistan
# ... with 1,694 more rows
```

`select`: Pick Multiple Columns by Name

```
select(gapminder, country, continent, year)
```

```
# A tibble: 1,704 x 3
   country     continent  year
   <fct>       <fct>     <int>
 1 Afghanistan Asia       1952
 2 Afghanistan Asia       1957
 3 Afghanistan Asia       1962
 4 Afghanistan Asia       1967
 5 Afghanistan Asia       1972
 6 Afghanistan Asia       1977
 7 Afghanistan Asia       1982
 8 Afghanistan Asia       1987
 9 Afghanistan Asia       1992
10 Afghanistan Asia       1997
# ... with 1,694 more rows
```

`select`: Removing Single Column

```
select(gapminder, - gdpPercap)
```

```
# A tibble: 1,704 x 5
   country     continent  year lifeExp      pop
   <fct>       <fct>     <int>   <dbl>    <int>
 1 Afghanistan Asia       1952    28.8  8425333
 2 Afghanistan Asia       1957    30.3  9240934
 3 Afghanistan Asia       1962    32.0 10267083
 4 Afghanistan Asia       1967    34.0 11537966
 5 Afghanistan Asia       1972    36.1 13079460
 6 Afghanistan Asia       1977    38.4 14880372
 7 Afghanistan Asia       1982    39.9 12881816
 8 Afghanistan Asia       1987    40.8 13867957
 9 Afghanistan Asia       1992    41.7 16317921
10 Afghanistan Asia       1997    41.8 22227415
# ... with 1,694 more rows
```

select: Removing Multiple Columns

```
select(gapminder, -c(pop,  gdpPercap))
```

```
# A tibble: 1,704 x 4
   country     continent  year lifeExp
   <fct>       <fct>     <int>   <dbl>
 1 Afghanistan Asia       1952    28.8
 2 Afghanistan Asia       1957    30.3
 3 Afghanistan Asia       1962    32.0
 4 Afghanistan Asia       1967    34.0
 5 Afghanistan Asia       1972    36.1
 6 Afghanistan Asia       1977    38.4
 7 Afghanistan Asia       1982    39.9
 8 Afghanistan Asia       1987    40.8
 9 Afghanistan Asia       1992    41.7
10 Afghanistan Asia       1997    41.8
# ... with 1,694 more rows
```

select: Select Column Using : (Range)

```
select(gapminder, country:year)
```

```
# A tibble: 1,704 x 3
   country     continent  year
   <fct>       <fct>     <int>
 1 Afghanistan Asia       1952
 2 Afghanistan Asia       1957
 3 Afghanistan Asia       1962
 4 Afghanistan Asia       1967
 5 Afghanistan Asia       1972
 6 Afghanistan Asia       1977
 7 Afghanistan Asia       1982
 8 Afghanistan Asia       1987
 9 Afghanistan Asia       1992
10 Afghanistan Asia       1997
# ... with 1,694 more rows
```

select: Select Single Column Using "contains"

```
select(gapminder, contains("c"))
```

```
# A tibble: 1,704 x 3
   country     continent gdpPercap
   <fct>       <fct>         <dbl>
 1 Afghanistan Asia           779.
 2 Afghanistan Asia           821.
 3 Afghanistan Asia           853.
 4 Afghanistan Asia           836.
 5 Afghanistan Asia           740.
 6 Afghanistan Asia           786.
 7 Afghanistan Asia           978.
 8 Afghanistan Asia           852.
 9 Afghanistan Asia           649.
10 Afghanistan Asia           635.
# ... with 1,694 more rows
```

select: Select Multiple Columns Using "contains"

```
select(gapminder, contains("c"), contains('g'))
```

```
# A tibble: 1,704 x 3
   country     continent gdpPercap
   <fct>       <fct>         <dbl>
 1 Afghanistan Asia           779.
 2 Afghanistan Asia           821.
 3 Afghanistan Asia           853.
 4 Afghanistan Asia           836.
 5 Afghanistan Asia           740.
 6 Afghanistan Asia           786.
 7 Afghanistan Asia           978.
 8 Afghanistan Asia           852.
 9 Afghanistan Asia           649.
10 Afghanistan Asia           635.
# ... with 1,694 more rows
```

select: Select Column Using "starts_with"

```
select(gapminder, starts_with('c'))
```

```
# A tibble: 1,704 x 2
   country     continent
   <fct>       <fct>
 1 Afghanistan Asia
 2 Afghanistan Asia
 3 Afghanistan Asia
 4 Afghanistan Asia
 5 Afghanistan Asia
 6 Afghanistan Asia
 7 Afghanistan Asia
 8 Afghanistan Asia
 9 Afghanistan Asia
10 Afghanistan Asia
# ... with 1,694 more rows
```

select: Select Column Using "ends_with"

```
select(gapminder, ends_with('p'))
```

```
# A tibble: 1,704 x 3
   lifeExp      pop gdpPercap
     <dbl>    <int>     <dbl>
 1    28.8  8425333      779.
 2    30.3  9240934      821.
 3    32.0 10267083      853.
 4    34.0 11537966      836.
 5    36.1 13079460      740.
 6    38.4 14880372      786.
 7    39.9 12881816      978.
 8    40.8 13867957      852.
 9    41.7 16317921      649.
10    41.8 22227415      635.
# ... with 1,694 more rows
```

Chaining Method: The Pipe(%>%) Operator

```
gapminder %>%
  select(country, continent, year) %>%
  head()
```

```
# A tibble: 6 x 3
  country     continent  year
  <fct>       <fct>     <int>
1 Afghanistan Asia       1952
2 Afghanistan Asia       1957
3 Afghanistan Asia       1962
4 Afghanistan Asia       1967
5 Afghanistan Asia       1972
6 Afghanistan Asia       1977
```

The Count Verb

```
gapminder %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1  1704
```

Count Variable

```
gapminder %>%
  count(country)
```

```
# A tibble: 142 x 2
   country         n
 * <fct>       <int>
 1 Afghanistan    12
 2 Albania        12
 3 Algeria        12
 4 Angola         12
 5 Argentina      12
 6 Australia      12
 7 Austria        12
 8 Bahrain        12
 9 Bangladesh     12
10 Belgium        12
```

Count and Sort

```
gapminder %>%
  count(country, sort = TRUE)
```

```
# A tibble: 142 x 2
   country         n
   <fct>       <int>
 1 Afghanistan    12
 2 Albania        12
 3 Algeria        12
 4 Angola         12
 5 Argentina      12
 6 Australia      12
 7 Austria        12
 8 Bahrain        12
 9 Bangladesh     12
10 Belgium        12
```

Count Population

```
gapminder %>%
  select(country, pop) %>%
  count(country, wt = pop, sort = TRUE)
```

```
# A tibble: 142 x 2
   country                 n
   <fct>               <dbl>
 1 China          11497920623
 2 India           8413568878
 3 United States   2738534790
 4 Indonesia       1779874000
 5 Brazil          1467745520
 6 Japan           1341105696
 7 Pakistan        1124200629
 8 Bangladesh      1089064744
 9 Germany          930564520
```

filter: Equality("==")

```
filter(gapminder, country == "Bangladesh")
```

```
# A tibble: 12 x 6
   country    continent  year lifeExp      pop gdpPercap
   <fct>      <fct>     <int>   <dbl>    <int>     <dbl>
 1 Bangladesh Asia       1952    37.5 46886859      684.
 2 Bangladesh Asia       1957    39.3 51365468      662.
 3 Bangladesh Asia       1962    41.2 56839289      686.
 4 Bangladesh Asia       1967    43.5 62821884      721.
 5 Bangladesh Asia       1972    45.3 70759295      630.
 6 Bangladesh Asia       1977    46.9 80428306      660.
 7 Bangladesh Asia       1982    50.0 93074406      677.
 8 Bangladesh Asia       1987    52.8 103764241     752.
 9 Bangladesh Asia       1992    56.0 113704579     838.
10 Bangladesh Asia       1997    59.4 123315288     973.
11 Bangladesh Asia       2002    62.0 135656790    1136.
```

filter: Inequality("!=")

```
filter(gapminder, country != "Bangladesh")
```

```
# A tibble: 1,692 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.
 2 Afghanistan Asia       1957    30.3  9240934      821.
 3 Afghanistan Asia       1962    32.0 10267083      853.
 4 Afghanistan Asia       1967    34.0 11537966      836.
 5 Afghanistan Asia       1972    36.1 13079460      740.
 6 Afghanistan Asia       1977    38.4 14880372      786.
 7 Afghanistan Asia       1982    39.9 12881816      978.
 8 Afghanistan Asia       1987    40.8 13867957      852.
 9 Afghanistan Asia       1992    41.7 16317921      649.
10 Afghanistan Asia       1997    41.8 22227415      635.
# ... with 1,682 more rows
```

filter: Greater(">")

```
filter(gapminder, gdpPercap > 800)
```

```
# A tibble: 1,460 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1957    30.3  9240934      821.
 2 Afghanistan Asia       1962    32.0 10267083      853.
 3 Afghanistan Asia       1967    34.0 11537966      836.
 4 Afghanistan Asia       1982    39.9 12881816      978.
 5 Afghanistan Asia       1987    40.8 13867957      852.
 6 Afghanistan Asia       2007    43.8 31889923      975.
 7 Albania     Europe     1952    55.2  1282697     1601.
 8 Albania     Europe     1957    59.3  1476505     1942.
 9 Albania     Europe     1962    64.8  1728137     2313.
10 Albania     Europe     1967    66.2  1984060     2760.
# ... with 1,450 more rows
```

filter: Greater or Equal(">=")

```
filter(gapminder, gdpPercap >= 800)
```

```
# A tibble: 1,460 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1957    30.3  9240934      821.
 2 Afghanistan Asia       1962    32.0 10267083      853.
 3 Afghanistan Asia       1967    34.0 11537966      836.
 4 Afghanistan Asia       1982    39.9 12881816      978.
 5 Afghanistan Asia       1987    40.8 13867957      852.
 6 Afghanistan Asia       2007    43.8 31889923      975.
 7 Albania     Europe     1952    55.2  1282697     1601.
 8 Albania     Europe     1957    59.3  1476505     1942.
 9 Albania     Europe     1962    64.8  1728137     2313.
10 Albania     Europe     1967    66.2  1984060     2760.
# ... with 1,450 more rows
```

filter: Less("<")

```
filter(gapminder, gdpPercap < 800)
```

```
# A tibble: 244 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.
 2 Afghanistan Asia       1972    36.1 13079460      740.
 3 Afghanistan Asia       1977    38.4 14880372      786.
 4 Afghanistan Asia       1992    41.7 16317921      649.
 5 Afghanistan Asia       1997    41.8 22227415      635.
 6 Afghanistan Asia       2002    42.1 25268405      727.
 7 Bangladesh  Asia       1952    37.5 46886859      684.
 8 Bangladesh  Asia       1957    39.3 51365468      662.
 9 Bangladesh  Asia       1962    41.2 56839289      686.
10 Bangladesh  Asia       1967    43.5 62821884      721.
# ... with 234 more rows
```

filter: Less or Equal("<=")

```
filter(gapminder, gdpPercap <= 800)
```

```
# A tibble: 244 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1952    28.8  8425333      779.
 2 Afghanistan Asia       1972    36.1 13079460      740.
 3 Afghanistan Asia       1977    38.4 14880372      786.
 4 Afghanistan Asia       1992    41.7 16317921      649.
 5 Afghanistan Asia       1997    41.8 22227415      635.
 6 Afghanistan Asia       2002    42.1 25268405      727.
 7 Bangladesh  Asia       1952    37.5 46886859      684.
 8 Bangladesh  Asia       1957    39.3 51365468      662.
 9 Bangladesh  Asia       1962    41.2 56839289      686.
10 Bangladesh  Asia       1967    43.5 62821884      721.
# ... with 234 more rows
```

filter: Logical AND("&")

```
filter(gapminder, country=="Bangladesh" & gdpPercap > 800)
```

```
# A tibble: 4 x 6
  country    continent  year lifeExp       pop gdpPercap
  <fct>      <fct>     <int>   <dbl>     <int>     <dbl>
1 Bangladesh Asia       1992    56.0 113704579      838.
2 Bangladesh Asia       1997    59.4 123315288      973.
3 Bangladesh Asia       2002    62.0 135656790     1136.
4 Bangladesh Asia       2007    64.1 150448339     1391.
```

filter: Logical OR("|")

```
filter(gapminder, country == "Bangladesh" |  gdpPercap > 800)
```

```
# A tibble: 1,468 x 6
   country     continent  year lifeExp      pop gdpPercap
   <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
 1 Afghanistan Asia       1957    30.3  9240934      821.
 2 Afghanistan Asia       1962    32.0 10267083      853.
 3 Afghanistan Asia       1967    34.0 11537966      836.
 4 Afghanistan Asia       1982    39.9 12881816      978.
 5 Afghanistan Asia       1987    40.8 13867957      852.
 6 Afghanistan Asia       2007    43.8 31889923      975.
 7 Albania     Europe     1952    55.2  1282697     1601.
 8 Albania     Europe     1957    59.3  1476505     1942.
 9 Albania     Europe     1962    64.8  1728137     2313.
10 Albania     Europe     1967    66.2  1984060     2760.
# ... with 1,458 more rows
```

filter: The "%in%" Operator

```
filter(gapminder, country %in% c("Bangladesh", "Australia"))
```

```
# A tibble: 24 x 6
   country   continent  year lifeExp      pop gdpPercap
   <fct>     <fct>     <int>   <dbl>    <int>     <dbl>
 1 Australia Oceania    1952    69.1  8691212    10040.
 2 Australia Oceania    1957    70.3  9712569    10950.
 3 Australia Oceania    1962    70.9 10794968    12217.
 4 Australia Oceania    1967    71.1 11872264    14526.
 5 Australia Oceania    1972    71.9 13177000    16789.
 6 Australia Oceania    1977    73.5 14074100    18334.
 7 Australia Oceania    1982    74.7 15184200    19477.
 8 Australia Oceania    1987    76.3 16257249    21889.
 9 Australia Oceania    1992    77.6 17481977    23425.
10 Australia Oceania    1997    78.8 18565243    26998.
# ... with 14 more rows
```

mutate: Creating New Column

```
gapminder %>%
  mutate(gdp = gdpPercap * pop) %>%
  head()
```

```
# A tibble: 6 x 7
  country     continent  year lifeExp      pop gdpPercap          gdp
  <fct>       <fct>     <int>   <dbl>    <int>     <dbl>        <dbl>
1 Afghanistan Asia       1952    28.8  8425333      779.  6567086330.
2 Afghanistan Asia       1957    30.3  9240934      821.  7585448670.
3 Afghanistan Asia       1962    32.0 10267083      853.  8758855797.
4 Afghanistan Asia       1967    34.0 11537966      836.  9648014150.
5 Afghanistan Asia       1972    36.1 13079460      740.  9678553274.
6 Afghanistan Asia       1977    38.4 14880372      786. 11697659231.
```

mutate: Creating New Column (Cont..)

```
# GDP in Million
gapminder %>%
  mutate(gdp = gdpPercap * pop / 10^6) %>%
  head()
```

```
# A tibble: 6 x 7
  country     continent  year lifeExp      pop gdpPercap    gdp
  <fct>       <fct>     <int>   <dbl>    <int>     <dbl>  <dbl>
1 Afghanistan Asia       1952    28.8  8425333       779.  6567.
2 Afghanistan Asia       1957    30.3  9240934       821.  7585.
3 Afghanistan Asia       1962    32.0 10267083       853.  8759.
4 Afghanistan Asia       1967    34.0 11537966       836.  9648.
5 Afghanistan Asia       1972    36.1 13079460       740.  9679.
6 Afghanistan Asia       1977    38.4 14880372       786. 11698.
```

arrange : Reorder Rows

```
gapminder %>%
  select(country, pop) %>%
  arrange(pop) %>%
  head()
```

```
# A tibble: 6 x 2
  country                   pop
  <fct>                   <int>
1 Sao Tome and Principe   60011
2 Sao Tome and Principe   61325
3 Djibouti                63149
4 Sao Tome and Principe   65345
5 Sao Tome and Principe   70787
6 Djibouti                71851
```

arrange : Reorder Rows(Descending)

```
gapminder %>%
  select(country, pop) %>%
  # descending order
  arrange(desc(pop)) %>%
  head()
```

```
# A tibble: 6 x 2
  country        pop
  <fct>        <int>
1 China   1318683096
2 China   1280400000
3 China   1230075000
4 China   1164970000
5 India   1110396331
6 China   1084035000
```

group_by: Grouping Data

```
gapminder %>%
  group_by(continent) %>%
  head()
```

```
# A tibble: 6 x 6
# Groups:   continent [1]
  country     continent  year lifeExp      pop gdpPercap
  <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
1 Afghanistan Asia       1952    28.8  8425333      779.
2 Afghanistan Asia       1957    30.3  9240934      821.
3 Afghanistan Asia       1962    32.0 10267083      853.
4 Afghanistan Asia       1967    34.0 11537966      836.
5 Afghanistan Asia       1972    36.1 13079460      740.
6 Afghanistan Asia       1977    38.4 14880372      786.
```

group_by and summarize

```
gapminder %>%
  summarize(total_pop = sum(pop))
```

```
# A tibble: 1 x 1
    total_pop
        <dbl>
1 50440465801
```

Aggregate and summarize

```
gapminder %>%
  summarize(total_population = sum(pop),
            avg_gdppercap = mean(gdpPercap))
```

```
# A tibble: 1 x 2
  total_population avg_gdppercap
             <dbl>          <dbl>
1      50440465801          7215.
```

summarise: Sum

```
gapminder %>%
  group_by(continent) %>%
  summarise(pop = sum(pop))
```

```
# A tibble: 5 x 2
  continent         pop
* <fct>           <dbl>
1 Africa     6187585961
2 Americas   7351438499
3 Asia      30507333901
4 Europe     6181115304
5 Oceania     212992136
```

summarise: Sum

```
gapminder %>%
  group_by(continent) %>%
  # In Million
  summarise(pop = sum(pop) / 10^6)
```

```
# A tibble: 5 x 2
  continent    pop
* <fct>      <dbl>
1 Africa     6188.
2 Americas   7351.
3 Asia      30507.
4 Europe     6181.
5 Oceania     213.
```

summarise: Maximum

```
gapminder %>%
  group_by(continent) %>%
  summarise(max_liexp = max(lifeExp))
```

```
# A tibble: 5 x 2
  continent max_liexp
* <fct>         <dbl>
1 Africa         76.4
2 Americas       80.7
3 Asia           82.6
4 Europe         81.8
5 Oceania        81.2
```

Summary Functions

```
mean()
sum()
median()
min()
max()
n()
```

Aggregate within Groups

```
gapminder %>%
  group_by(continent) %>%
  summarize(total_pop = sum(pop),
            avg_lifeexp = mean(lifeExp))
```

```
# A tibble: 5 x 3
  continent   total_pop avg_lifeexp
* <fct>           <dbl>       <dbl>
1 Africa     6187585961        48.9
2 Americas   7351438499        64.7
3 Asia      30507333901        60.1
4 Europe     6181115304        71.9
5 Oceania     212992136        74.3
```

The top_n Verb

```
gapminder %>%
  group_by(continent) %>%
  top_n(1, pop)
```

```
# A tibble: 5 x 6
# Groups:   continent [5]
  country       continent  year lifeExp        pop gdpPercap
  <fct>         <fct>     <int>   <dbl>      <int>     <dbl>
1 Australia     Oceania    2007    81.2   20434176    34435.
2 China         Asia       2007    73.0 1318683096     4959.
3 Germany       Europe     2007    79.4   82400996    32170.
4 Nigeria       Africa     2007    46.9  135031164     2014.
5 United States Americas   2007    78.2  301139947    42952.
```

rename: Renaming Column

```
gapminder %>%
  rename(population = pop) %>%
  head()
```

```
# A tibble: 6 x 6
  country     continent  year lifeExp population gdpPercap
  <fct>       <fct>     <int>   <dbl>      <int>     <dbl>
1 Afghanistan Asia       1952    28.8    8425333      779.
2 Afghanistan Asia       1957    30.3    9240934      821.
3 Afghanistan Asia       1962    32.0   10267083      853.
4 Afghanistan Asia       1967    34.0   11537966      836.
5 Afghanistan Asia       1972    36.1   13079460      740.
6 Afghanistan Asia       1977    38.4   14880372      786.
```