Data Process in R

Class 3

Kai Yao

May, 2016

Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
  - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

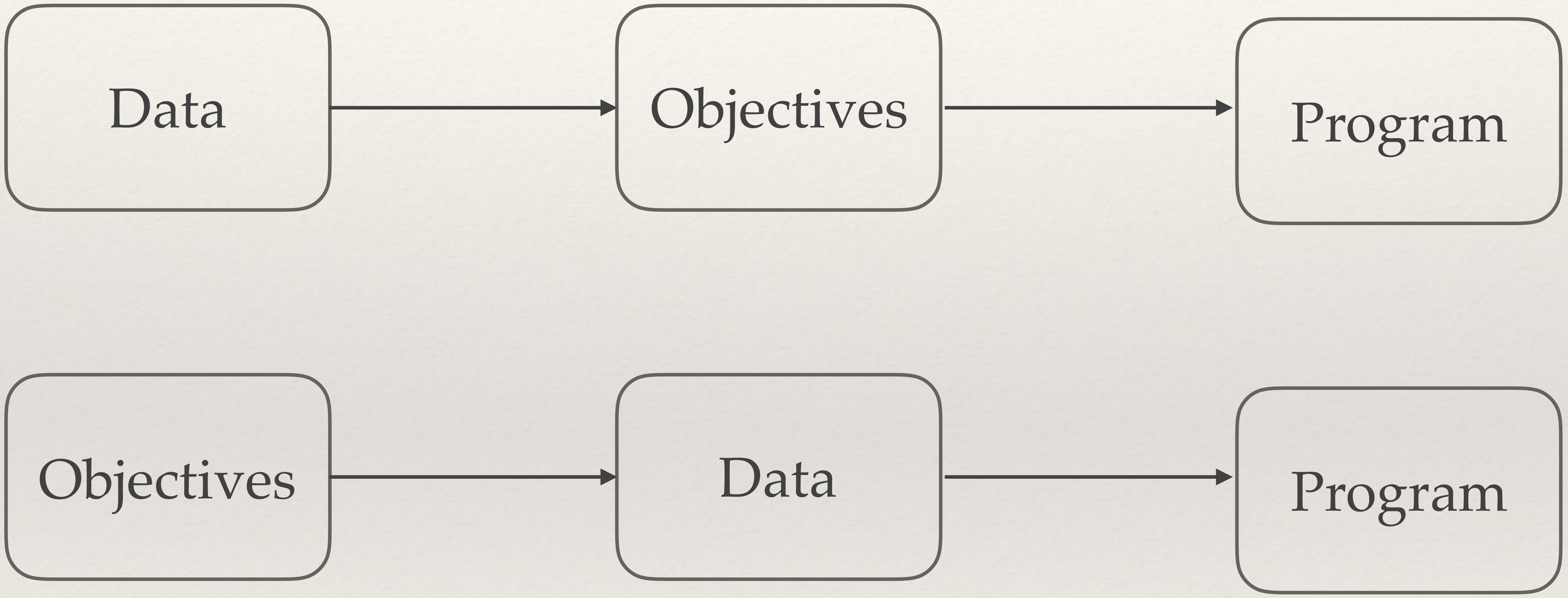
Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
  - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

Supplements

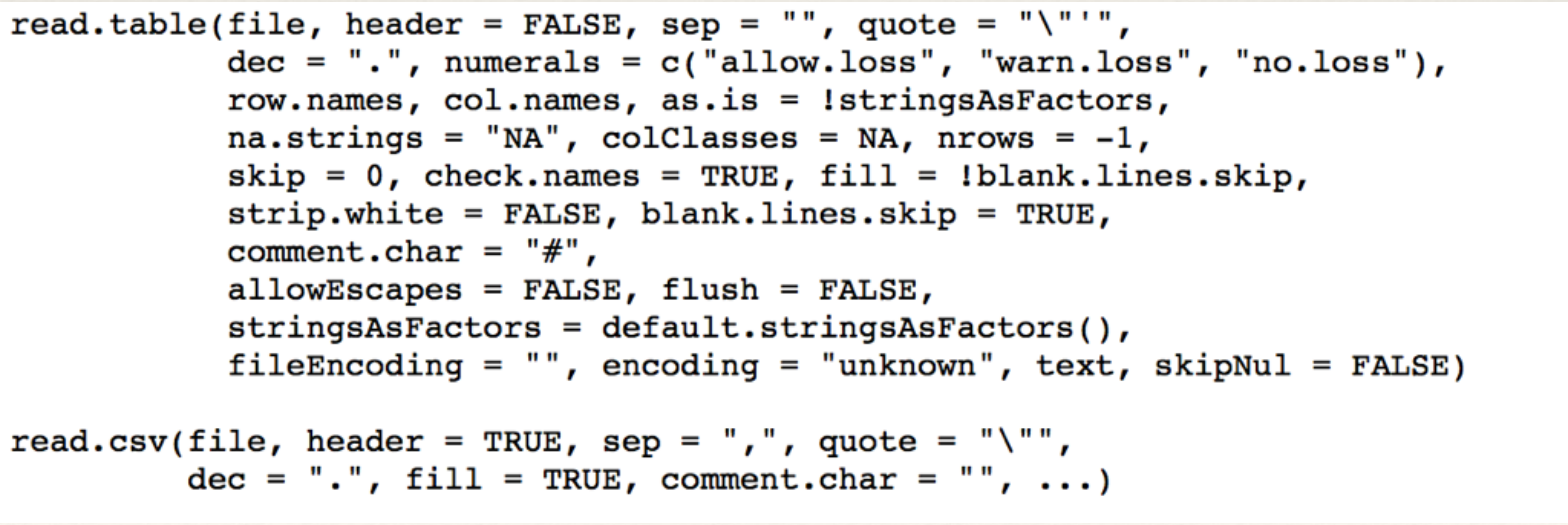
- Variable Definition
- Calculation Order
- Temporary Variable
- Vector Computation
- Multi Conditions
- Multi Loops

Data Process



Read

- File type: .txt, .csv
- Functions: read.table, read.csv

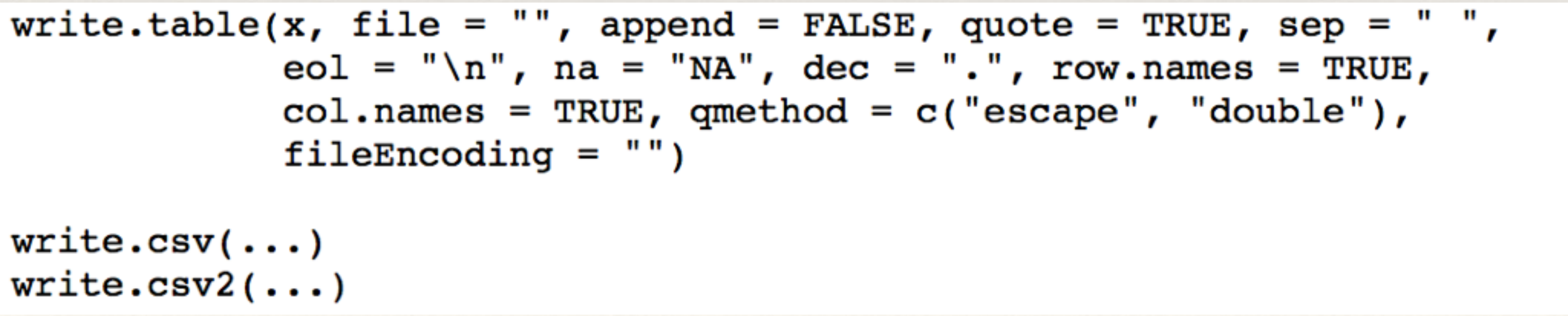


```
read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", numerals = c("allow.loss", "warn.loss", "no.loss"),
           row.names, col.names, as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)

read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", ...)
```

Write

- txt, csv
- x: (output data, notice the format)
- file: file name of the output data (notice the directory)
- row.names: whether keep names of rows
- col.names: whether keep names of columns



```
write.table(x, file = "", append = FALSE, quote = TRUE, sep = " ",
            eol = "\n", na = "NA", dec = ".", row.names = TRUE,
            col.names = TRUE, qmethod = c("escape", "double"),
            fileEncoding = "")

write.csv(...)
write.csv2(...)
```

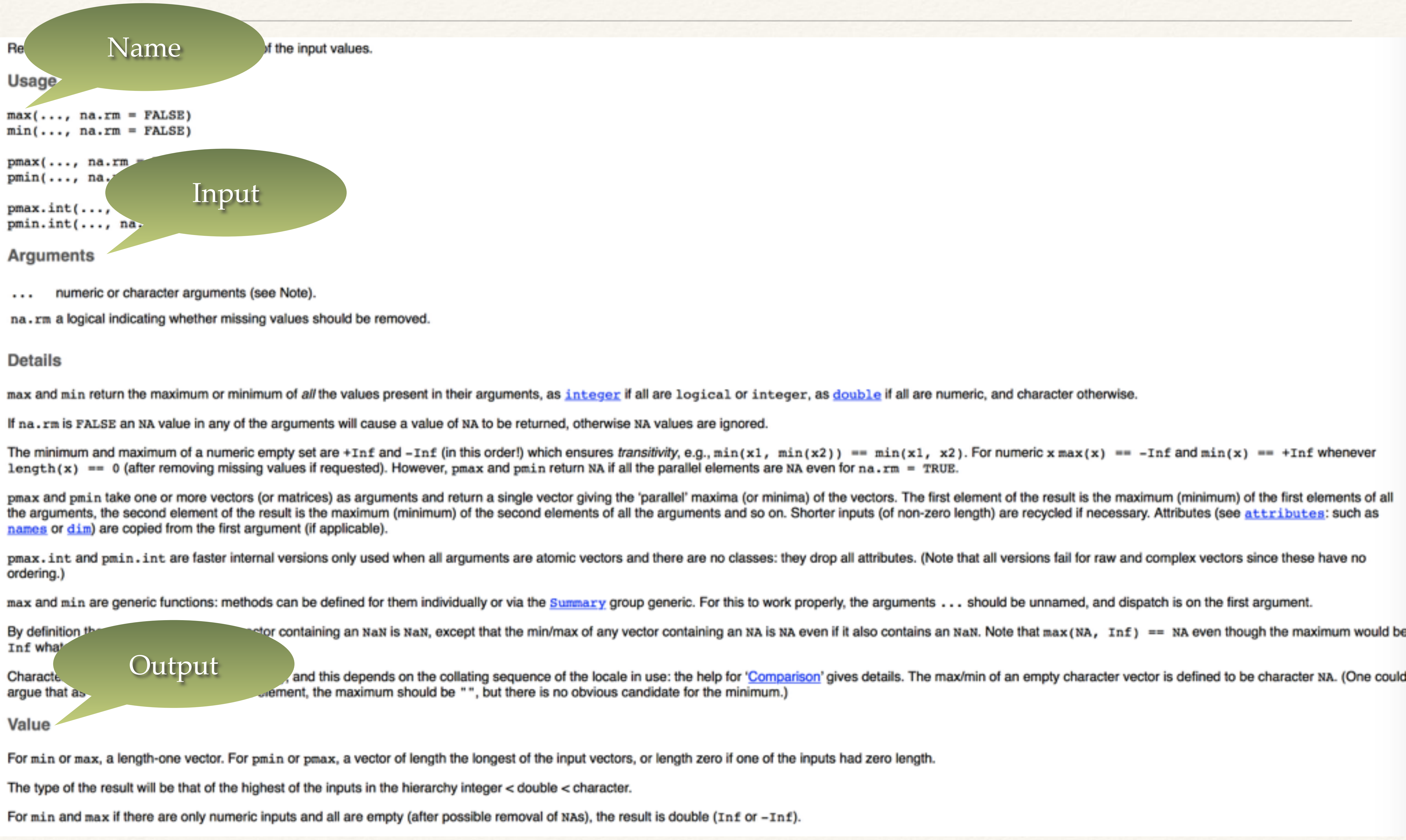
Frequency Distribution

- Min
- Max
- Median
- Mean

Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
    - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

How to Define Function



Re f the input values.

Usage

```
max(..., na.rm = FALSE)
min(..., na.rm = FALSE)

pmax(..., na.rm =
pmin(..., na.

pmax.int(...,
pmin.int(..., na.
```

Arguments

`...`	numeric or character arguments (see Note).
`na.rm`	a logical indicating whether missing values should be removed.

Details

`max` and `min` return the maximum or minimum of *all* the values present in their arguments, as `integer` if all are `logical` or `integer`, as `double` if all are numeric, and character otherwise.

If `na.rm` is `FALSE` an `NA` value in any of the arguments will cause a value of `NA` to be returned, otherwise `NA` values are ignored.

The minimum and maximum of a numeric empty set are `+Inf` and `-Inf` (in this order!) which ensures *transitivity*, e.g., `min(x1, min(x2)) == min(x1, x2)`. For numeric `x` `max(x) == -Inf` and `min(x) == +Inf` whenever `length(x) == 0` (after removing missing values if requested). However, `pmax` and `pmin` return `NA` if all the parallel elements are `NA` even for `na.rm = TRUE`.

`pmax` and `pmin` take one or more vectors (or matrices) as arguments and return a single vector giving the 'parallel' maxima (or minima) of the vectors. The first element of the result is the maximum (minimum) of the first elements of all the arguments, the second element of the result is the maximum (minimum) of the second elements of all the arguments and so on. Shorter inputs (of non-zero length) are recycled if necessary. Attributes (see `attributes`: such as `names` or `dim`) are copied from the first argument (if applicable).

`pmax.int` and `pmin.int` are faster internal versions only used when all arguments are atomic vectors and there are no classes: they drop all attributes. (Note that all versions fail for raw and complex vectors since these have no ordering.)

`max` and `min` are generic functions: methods can be defined for them individually or via the `Summary` group generic. For this to work properly, the arguments `...` should be unnamed, and dispatch is on the first argument.

By definition th ctor containing an `NaN` is `NaN`, except that the min/max of any vector containing an `NA` is `NA` even if it also contains an `NaN`. Note that `max(NA, Inf) == NA` even though the maximum would be `Inf` wha

Characte , and this depends on the collating sequence of the locale in use: the help for 'Comparison' gives details. The max/min of an empty character vector is defined to be character `NA`. (One could argue that as element, the maximum should be `""`, but there is no obvious candidate for the minimum.)

Value

For `min` or `max`, a length-one vector. For `pmin` or `pmax`, a vector of length the longest of the input vectors, or length zero if one of the inputs had zero length.

The type of the result will be that of the highest of the inputs in the hierarchy integer < double < character.

For `min` and `max` if there are only numeric inputs and all are empty (after possible removal of `NA`s), the result is double (`Inf` or `-Inf`).

How to Define Function

```
Function_Name = function(arg1, arg2…)
{
    #body of the function
    return(result);
}
```

The red parts are defined by programer!

How to Use Own Function

```
Function_Name = function(arg1, arg2)

{

    #body of the function

    return(result);

}

re = Function_Name(var1,var2)
```

Notice: relationships between var1, var2, re and arg1, arg2, result

Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
  - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

Add

- rbind
- cbind

Delete

- Dataframe[-rownumber,]
- resave to another dataframe

Revise

- Dataframe[i,j] =
- Dataframe$var =
- Dataframe$var[i] =

Search

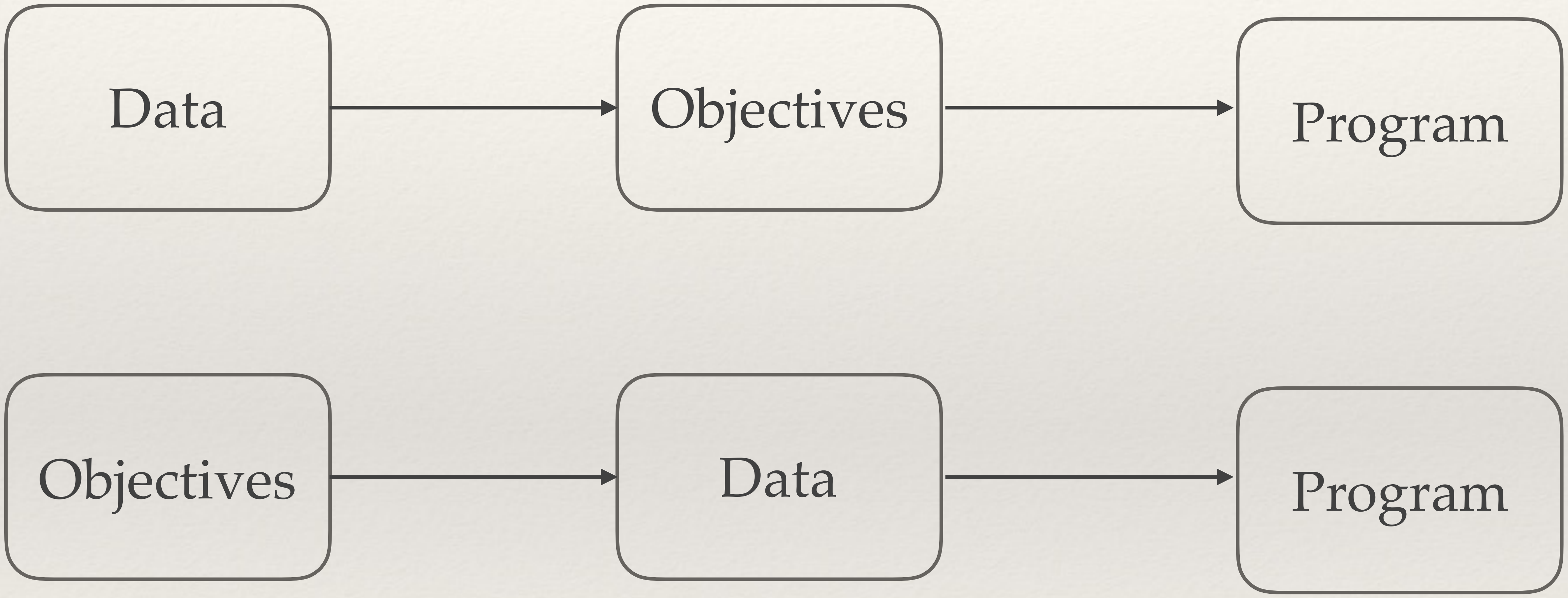
- Dataframe[i,j]
- Dataframe[c(1,2…),c(1,2…)]
- Dataframe[condition,]

  e.g. Dataframe[Dataframe$var>100,]

Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
    - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

Data Process



Data Analysis Steps

1. Brief Overview
2. Data Clean
3. Description Analysis
4. Visualization
5. Modeling

Brief Overview

- The structure of the dataset
- How many columns?
- The meaning of each column
- Possible issues of the data
- Is the format of the columns right?

COMEDY CENTRAL
THE DAILY SHOW
WITH JON STEWART

Data Clean

- missing data
- wrong format
- wrong data
- outlier

Description Analysis

- Analyze data in each column
- min
- max
- median
- standard deviation

Outline

- Review of Last Class
- Create Own Function
- More Operations in Data Frame
  - Add, Delete, Revise, Search
- Data Analysis Steps
- Data Visualization

Data Visualization

- plot
- hist
- sort

Conclusion

- Variable Definition
- Calculation Order
- Temporary Variable
- Vector Computation
- Multi Conditions
- Multi Loops

Next Class

Case Study of Movie Data

In Class Test

Good Luck!