

ASSIGNMENT 1

Output on the Console window

PART A

1. Load these files into working directory, one by one

```
> data1 <- read.csv("data/201808.csv", skip = 7)
> data2 <- read.csv("data/201809.csv", skip = 7)
> data3 <- read.csv("data/201810.csv", skip = 7)
> data4 <- read.csv("data/201811.csv", skip = 7)
> data5 <- read.csv("data/201812.csv", skip = 7)
> data6 <- read.csv("data/201901.csv", skip = 7)
> data7 <- read.csv("data/201902.csv", skip = 7)
> data8 <- read.csv("data/201903.csv", skip = 7)
> data9 <- read.csv("data/201904.csv", skip = 7)
> data10 <- read.csv("data/201905.csv", skip = 7)
> data11 <- read.csv("data/201906.csv", skip = 7)
> data12 <- read.csv("data/201907.csv", skip = 7)
> data13 <- read.csv("data/201908.csv", skip = 7)
> data14 <- read.csv("data/201909.csv", skip = 7)
> data15 <- read.csv("data/201910.csv", skip = 7)
> data16 <- read.csv("data/201911.csv", skip = 7)
> data17 <- read.csv("data/201912.csv", skip = 7)
> data18 <- read.csv("data/202001.csv", skip = 7)
> data19 <- read.csv("data/202002.csv", skip = 7)
```

2. Concatenate all the data of these file into one data frame

```
> list.files()
[1] "201808.csv" "201809.csv" "201810.csv" "201811.csv" "201812.csv"
[6] "201901.csv" "201902.csv" "201903.csv" "201904.csv" "201905.csv"
[11] "201906.csv" "201907.csv" "201908.csv" "201909.csv" "201910.csv"
[16] "201911.csv" "201912.csv" "202001.csv" "202002.csv"
```

The screenshot shows the RStudio interface. The main window displays a data table with 14 rows and 5 columns: Date, Minimum.temperature, Maximum.temperature, Rainfall.mm., and Evaporati. The console shows the following code and output:

```
> library(data.table)
> x <- mydf %>% rowwise() %>%
+   mutate( min_value = min(c(X3pm_wind_speed__km_h_,X9am_wind_speed__km_h_)))
> View(x)
> class(mydf$X3pm_wind_speed__km_h_ )
[1] "numeric"
> #check problems while loading and parsing the data
> stop_for_problems(mydf)
> #check problems while loading and parsing the data
> problems(mydf)
[1] row      col      expected actual
<0 rows> (or 0-length row.names)
> mydf <- ldply(list.files(), read.csv, skip=7, header=TRUE)
> View(mydf)
>
```

The file explorer on the right shows the file structure, including folders like VirtualBox VMs, UC, Public, Projects, Pictures, OneDrive, Music, Movies, Library, IntroDataScience, HelloWorld, Downloads, Documents, Desktop, Creative Cloud Files, Applications, and anaconda3.

3. Check problems while loading and parsing the data

```
> #check problems while loading and parsing the data
> stop_for_problems(mydf)
> #check problems while loading and parsing the data
> problems(mydf)
[1] row      col      expected actual
<0 rows> (or 0-length row.names)
```

PART B

1. Remove the variables, which have no data at all

RStudio

Project: (None)

Assignment1.R* x mydf x

Filter

	Date	Minimum.temperature	Maximum.temperature	Rainfall.mm.	Direction
1	1/08/2018	7.6	15.4	0.0	NW
2	2/08/2018	-3.8	14.3	0.0	NNW
3	3/08/2018	-3.6	19.5	0.0	NW
4	4/08/2018	3.7	12.8	13.8	NNW
5	5/08/2018	-1.0	15.0	0.0	NW
6	6/08/2018	1.2	13.7	0.0	NW
7	7/08/2018	2.4	9.7	6.6	WNW
8	8/08/2018	2.6	12.1	0.0	WNW
9	9/08/2018	1.6	13.7	0.0	NNW
10	10/08/2018	-2.5	15.6	0.2	NNW
11	11/08/2018	0.4	16.4	0.0	NW
12	12/08/2018	1.4	11.7	3.0	WNW
13	13/08/2018	2.4	13.8	0.0	WNW
14	14/08/2018	2.8	16.4	0.0	N

Showing 1 to 14 of 578 entries, 19 total columns

Console Terminal x Jobs x

```
~/IntroDataScience/Assignment1/data/
44      1018.1
45      1014.9
46      1005.0
47      1018.3
48      1020.6
49      1013.2
50      1010.7
51      1018.1
52      1024.4
[ reached 'max' / getOption("max.print") -- omitted 526 rows ]
> ##### PART B #####
> #1.Remove the variables, which have no data at all
> mydf <- mydf[ , colSums(is.na(mydf)) < nrow(mydf)]
> View(mydf)
> |
```

Environment History Connections

R | Global Environment

- max_wind... 19 obs. of 3 variab...
- min_wind... 19 obs. of 3 variab...
- monthly_... 19 obs. of 3 variab...
- monthly_... 19 obs. of 3 variab...
- monthly_... 19 obs. of 3 variab...
- monthly_... 19 obs. of 3 variab...
- mydata 19 obs. of 8 variab...
- mydf 578 obs. of 19 variab...
- remove_n... 578 obs. of 19 variab...

Files Plots Packages Help Vi

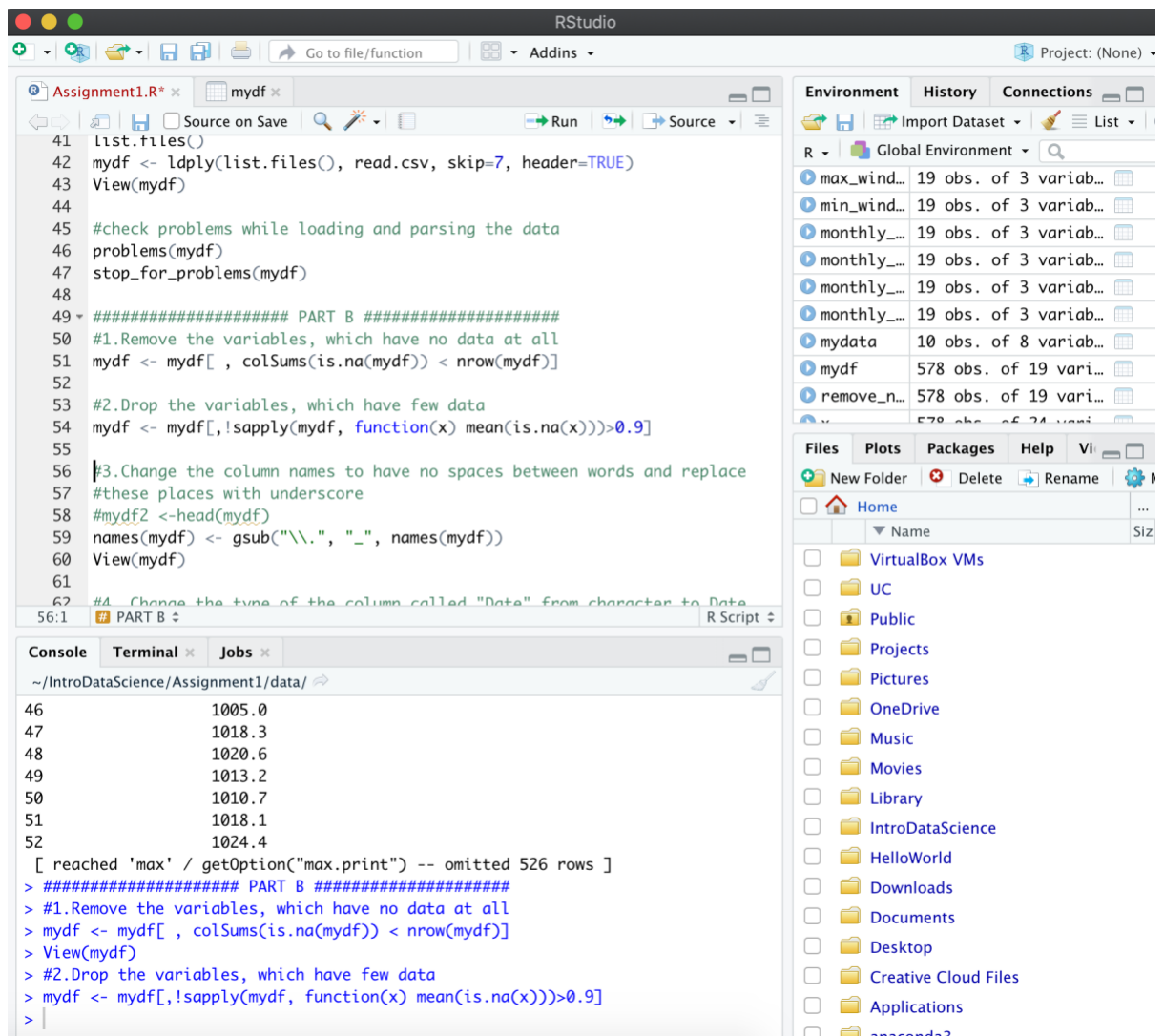
New Folder Delete Rename

Home

- VirtualBox VMs
- UC
- Public
- Projects
- Pictures
- OneDrive
- Music
- Movies
- Library
- IntroDataScience
- HelloWorld
- Downloads
- Documents
- Desktop
- Creative Cloud Files
- Applications
- anaconda3

*There used to be 21 variables, down to 19 variables now

- Drop the variables, which have few data



3. Change the column names to have no spaces between words and replace these places with underscore

RStudio

Assignment1.R* x mydf x

Filter

	Date	Minimum_temperature	Maximum_temperature	Rainfall_mm_	Directi
1	1/08/2018	7.6	15.4	0.0	NW
2	2/08/2018	-3.8	14.3	0.0	NNW
3	3/08/2018	-3.6	19.5	0.0	NW
4	4/08/2018	3.7	12.8	13.8	NNW
5	5/08/2018	-1.0	15.0	0.0	NW
6	6/08/2018	1.2	13.7	0.0	NW
7	7/08/2018	2.4	9.7	6.6	WNW
8	8/08/2018	2.6	12.1	0.0	WNW
9	9/08/2018	1.6	13.7	0.0	NNW
10	10/08/2018	-2.5	15.6	0.2	NNW
11	11/08/2018	0.4	16.4	0.0	NW
12	12/08/2018	1.4	11.7	3.0	WNW
13	13/08/2018	2.4	13.8	0.0	WNW
14	14/08/2018	2.8	16.4	0.0	N

Showing 1 to 14 of 578 entries, 19 total columns

Console Terminal x Jobs x

~/IntroDataScience/Assignment1/data/

```

51          1018.1
52          1024.4
[ reached 'max' / getOption("max.print") -- omitted 526 rows ]
> ##### PART B #####
> #1.Remove the variables, which have no data at all
> mydf <- mydf[ , colSums(is.na(mydf)) < nrow(mydf)]
> View(mydf)
> #2.Drop the variables, which have few data
> mydf <- mydf[,!sapply(mydf, function(x) mean(is.na(x)))>0.9]
> #3.Change the column names to have no spaces between words and replace
> #these places with underscore
> #mydf2 <-head(mydf)
> names(mydf) <- gsub("\\.", "_", names(mydf))
> View(mydf)
>

```

- Change the type of the column called "Date" from character to Date data type


```

> #4. Change the type of the column called "Date" from character to Date
> #data type
> mydf$Date <- as.Date(paste(mydf$Date), format= "%d/%m/%Y")
> class(mydf$Date)
[1] "Date"

```
- Add two new columns for the month and year of the data in each file, you may extract the contents of this column from the Date column. Please note that the data are collected for the 19 months across 3 years

	Year	Month	Date	Minimum_temperature	Maximum_temperature	Rainfall_mm
1	2018	08	01	7.6	15.4	0.0
2	2018	08	02	-3.8	14.3	0.0
3	2018	08	03	-3.6	19.5	0.0
4	2018	08	04	3.7	12.8	13.8
5	2018	08	05	-1.0	15.0	0.0
6	2018	08	06	1.2	13.7	0.0
7	2018	08	07	2.4	9.7	6.6
8	2018	08	08	2.6	12.1	0.0
9	2018	08	09	1.6	13.7	0.0
10	2018	08	10	-2.5	15.6	0.2
11	2018	08	11	0.4	16.4	0.0
12	2018	08	12	1.4	11.7	3.0
13	2018	08	13	2.4	13.8	0.0

6. Change the type of the "Month" and "Year" columns from character to Ordinal with levels as the number of months in a year (12) and no of year(3)

```
> #6. Change the type of the "Month" and "Year" columns from character to
> #Ordinal with levels as the number of months in a year (12) and no of year(3)
> mydf$Year <- as.numeric(mydf$Year)
> levels(mydf$Year)=c(2018, 2019, 2020)
> mydf$Month <- as.numeric(mydf$Month)
> levels(mydf$Month)=c(1,2,3,4,5,6,7,8,9,10,11,12)
> levels(mydf$Year)
[1] 2018 2019 2020
> levels(mydf$Month)
[1] 1 2 3 4 5 6 7 8 9 10 11 12
> |
```

7. For all the numeric columns, replace the remaining NAs with the median of the values in the column, if exist

am_Temperature	X9am_relative_humidity__	X9am_cloud_amount__oktas_	X9am_wind_d
10.9	54	8	WNW
3.3	87	1	
3.7	84	8	
7.8	77	8	NNW
5.0	85	8	
9.2	72	8	N
4.9	90	5	NW
6.5	65	8	NW
6.3	94	7	SSW
0.5	99	8	
8.9	63	1	SE
6.4	72	8	W
7.6	76	8	NNW
9.6	66	8	NW
9.9	62	8	N

Showing 1 to 15 of 578 entries, 21 total columns

```

Console Terminal Jobs
~/IntroDataScience/Assignment1/data/
> levels(mydf$Month)
[1] 1 2 3 4 5 6 7 8 9 10 11 12
> mydf$X9am_cloud_amount__oktas_[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
[1] 8
> mydf$X9am_cloud_amount__oktas_[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> mydf$X3pm_cloud_amount__oktas_[is.na(mydf$X3pm_cloud_amount__oktas_)] <-
+ median(mydf$X3pm_cloud_amount__oktas_, na.rm = TRUE)
> mydf$Speed_of_maximum_wind_gust__km_h_[is.na(mydf$Speed_of_maximum_wind_gust__k
m_h_)] <-
+ median(mydf$Speed_of_maximum_wind_gust__km_h_, na.rm = TRUE)
> |

```

RStudio

Assignment1.R* mydf

Filter

pm_relative_humidity__	X3pm_cloud_amount__oktas_	X3pm_wind_direction	X3pm_wind
32	6	NW	19
43	6	NE	9
35	4	NNW	39
52	5	WNW	22
48	6	NW	24
44	8	WNW	31
54	6	WNW	44
60	8	WNW	35
50	6	N	15
42	6	NW	28
45	8	NNW	37
39	1	WNW	31
54	6	WNW	35
49	6	NNW	28
46	4	NW	37

Showing 1 to 15 of 578 entries, 21 total columns

Console Terminal Jobs

```
~/IntroDataScience/Assignment1/data/
> levels(mydf$Month)
[1] 1 2 3 4 5 6 7 8 9 10 11 12
> mydf$X9am_cloud_amount__oktas_[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
[1] 8
> mydf$X9am_cloud_amount__oktas_[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> mydf$X3pm_cloud_amount__oktas_[is.na(mydf$X3pm_cloud_amount__oktas_)] <-
+ median(mydf$X3pm_cloud_amount__oktas_, na.rm = TRUE)
> mydf$Speed_of_maximum_wind_gust__km_h_[is.na(mydf$Speed_of_maximum_wind_gust__k
m_h_)] <-
+ median(mydf$Speed_of_maximum_wind_gust__km_h_, na.rm = TRUE)
> |
```

8.

The screenshot displays the RStudio environment. The main editor shows a data frame 'mydf' with the following columns: 'gust', 'Speed_of_maximum_wind_gust_kmh', 'Time_of_maximum_wind_gust', and 'X9am_Tempe'. The data is displayed in a table with 15 rows shown (out of 578 total entries). The console shows the following R code and output:

```
~/IntroDataScience/Assignment1/data/
> levels(mydf$Month)
[1] 1 2 3 4 5 6 7 8 9 10 11 12
> mydf$X9am_cloud_amount__oktas[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
[1] 8
> mydf$X9am_cloud_amount__oktas[is.na(mydf$X9am_cloud_amount__oktas_)] <-
+ median(mydf$X9am_cloud_amount__oktas_, na.rm = TRUE)
> mydf$X3pm_cloud_amount__oktas[is.na(mydf$X3pm_cloud_amount__oktas_)] <-
+ median(mydf$X3pm_cloud_amount__oktas_, na.rm = TRUE)
> mydf$Speed_of_maximum_wind_gust__km_h[is.na(mydf$Speed_of_maximum_wind_gust__k
m_h_)] <-
+ median(mydf$Speed_of_maximum_wind_gust__km_h_, na.rm = TRUE)
> |
```

The Environment pane on the right lists the following objects:

- data8: 31 obs. of 21 varia...
- data9: 30 obs. of 21 varia...
- df: 336776 obs. of 19 v...
- m1: num [1:2, 1:3] 3 9 ...
- m2: num [1:3, 1:3] 5 2 ...
- m3: num [1:3, 1:2] 5 0 ...
- m4: num [1:2, 1:2] 21 6...
- mat: int [1:8, 1:10] 11 ...
- mydata: 10 obs. of 8 variab...
- mydf: 578 obs. of 21 vari...

The right sidebar shows the 'The R Datasets Package' documentation for version 4.0.4, including a link to the 'DESCRIPTION file' and a 'Help Pages' section.

PART C

1. Print the summary of 'Minimum_temperature', '9am_temperature', 'Speed_of_maximum_wind_gust_(km/h)'


```

> #1a. Print the summary of 'Minimum_temperature'
> summary(mydf$Minimum_temperature)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-6.40   2.00   8.00   7.84  13.60  26.70
> #1b. Print the summary of '9am_temperature'
> summary(mydf$X9am_Temperature)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.50   9.20  14.70  14.09  18.60  34.50
> #1b. Print the summary of '9am_temperature'
> summary(mydf$X9am_Temperature)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.50   9.20  14.70  14.09  18.60  34.50
> #1c. Print the summary of 'Speed_of_maximum_wind_gust_(km/h)'
> summary(mydf$Speed_of_maximum_wind_gust__km_h_)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
15.00  35.00  43.00  44.43  52.00 117.00
>

```

2. Extracting the average of minimum temperature per month and year monthly

```

> #2.Extracting the average of minimum temperature per month and year
> #monthly
> aggregate( Minimum_temperature ~ Month + Year , mydf , mean)
  Month Year Minimum_temperature
1     8 2018         0.76451613
2     9 2018         1.82000000
3    10 2018         7.29677419
4    11 2018        10.47666667
5    12 2018        13.74516129
6     1 2019        17.67741935
7     2 2019        12.90714286
8     3 2019        12.00000000
9     4 2019         7.46666667
10    5 2019         3.33870968
11    6 2019        -0.08333333
12    7 2019         0.66774194
13    8 2019         0.10967742
14    9 2019         2.12333333
15   10 2019         6.21290323
16   11 2019         9.48666667
17   12 2019        13.09677419
18    1 2020        15.24193548
19    2 2020        15.06896552
> #yearly
> aggregate( Minimum_temperature ~ Year , mydf , mean)
  Year Minimum_temperature
1 2018         6.829412
2 2019         7.061370
3 2020        15.158333
>

```

3. Extracting the average of speed of maximum wind gust by direction of maximum wind gust

The screenshot displays the RStudio interface. At the top, there are three tabs: 'Assignment1.R', 'average_windgust', and 'mydf'. Below the tabs is a toolbar with navigation icons and a 'Filter' button. The main area shows a data table with two columns: 'Direction_of_maximum_wind_gust' and 'Speed_of_maximum_wind_gust_kmh'. The table contains 17 rows of data, indexed 1 to 17. Below the table, it says 'Showing 1 to 17 of 17 entries, 2 total columns'. At the bottom, the 'Console' tab is active, showing the following R code and its output:

```
~/IntroDataScience/Assignment1/data/
+ aggregate(Speed_of_maximum_wind_gust_kmh ~ Direction_of_maximum_wind_gust
, mydf , mean)
> View(average_windgust)
> view(average_windgust)
> #3. Extracting the average of speed of maximum wind gust by direction
> #of maximum wind gust
> average_windgust <-
+ aggregate(Speed_of_maximum_wind_gust_kmh ~ Direction_of_maximum_wind_gust
, mydf , mean)
> view(average_windgust)
```

4. Which month was dry, if any? And in which year?

```

> #4. Which month was dry, if any? And in which year?
> monthly_rainfall <- aggregate(Rainfall__mm_ ~ Month + Year , mydf , FUN = sum
)
> if (0.0 %in% monthly_rainfall$Rainfall__mm_){
+   driest_month <-
+     summarise(monthly_rainfall, year= Year[which.min(monthly_rainfall$Rainfall
__mm_)]),
+     driest_month= Month[which.min(Rainfall__mm_)],
+     driest_value= min(Rainfall__mm_)
+ } else {
+   print("There is no dry month")
+ }
[1] "There is no dry month"
>

```

5. What about the humidity, which month in the ACT has the highest humidity level in 2019

```

> mydf_2019 <- filter(mydf, mydf$Year == "2019")
>
> aggregate(X9am_relative_humidity____+X3pm_relative_humidity____
~ Month + Year , mydf_2019 , FUN = sum )
  Month Year X9am_relative_humidity____ + X3pm_relative_humidity____
1     1 2019                        2971
2     2 2019                        2661
3     3 2019                        3503
4     4 2019                        3508
5     5 2019                        4253
6     6 2019                        4348
7     7 2019                        4195
8     8 2019                        3786
9     9 2019                        2894
10    10 2019                        2618
11    11 2019                        2113
12    12 2019                        2063
> summarise(monthly_humidity,
+   max_month= Month[which.max(`X9am_relative_humidity____ + X3pm_relative_humidity____`)],
+   max_value= max(`X9am_relative_humidity____ + X3pm_relative_humidity____`))
  max_month max_value
1         6     4348
>

```

6. For 2019, extract the minimum, maximum and average temperature, wind speed and humidity per month and per quarter in 2019 only
7. Plot the histogram/bar-charts for each variable of the previous question