

SAVEETHA SCHOOL OF ENGINEERING

SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES

ITA 0443 - STATISTICS WITH R PROGRAMMING FOR REAL TIME PROBLEM

DAY 2 – LAB EXERCISES

Reg No:192125027

Name:D.Takeshbabu

RESHAPE FUNCTION IN R

RESHAPE FUNCTION IN R

Exercise: 1

1. Construct the following data frame 'country'.

Program:

```
country<-data.frame(c("A","B","C"),c(100,200,120),c(2000,7000,15000))  
colnames(country)<- c("countries","population_in_million","gdp_percapita")  
print(country)
```

Output:

```
countries population_in_million gdp_percapita
```

1	A	100	2000
---	---	-----	------

2	B	200	7000
---	---	-----	------

3	C	120	15000
---	---	-----	-------

2. Reshape in R from wide to long:

Reshape the above data frame from wide to long format in R.

data frame "country" is passed to reshape function

idvar is the variable which need to be left unaltered which is "countries"

varying are the ones that needs to converted from wide to long

v.names are the values that should be against the times in the resultant data frame.

new.row.names is used to assign row names to the resultant dataset

direction is, to which format the data needs to be transformed

Program:

```
country1<- reshape(data=country, idvar="countries",  
varying = c("population_in_million","gdp_percapita"),  
v.name=c("value"),  
times=c("population_in_million","gdp_percapita"),  
new.row.names = 1:1000,  
direction="long")  
print(country1)
```

Output:

countries time value

1 A population_in_million 100

2 B population_in_million 200

3 C population_in_million 120

4 A gdp_percapita 2000

5 B gdp_percapita 7000

6 C gdp_percapita 15000

3.Reshape in R from long to wide:

data (country_w_to_L) which is in long format, is passed to reshape function

idvar is the variable which need to be left unaltered, which is “countries”

timevar are the variables that needs to converted to wide format

v.names are the value variable

direction is, to which format the data needs to be transformed

Program:

```
country2 <- reshape(data=country1,idvar="countries",
```

```
v.names = "value",
```

```
timevar = "time",
```

```
direction="wide")
```

```
print(country2)
```

Output:

```
countries value.population_in_million value.gdp_percapita
```

```
1      A           100           2000
```

```
2      B           200           7000
```

```
3      C           120          15000
```

MELTING AND CASTING IN R

Exercises :

Melt airquality data set and display as a long – format data ?

Melt airquality data and specify month and day to be “ID variables” ?

Cast the molten airquality data set .

Use cast function appropriately and compute the average of Ozone, Solar.R , Wind and temperature per month ?

Program:

```
names(airquality) <- tolower(names(airquality))
head(airquality)
aql <- melt(airquality, id.vars = c("month", "day"))
head(aql)
```

Output:

```
ozone solar.r wind temp month day
```

```
1  41   190 7.4  67   5  1
```

```
2  36   118 8.0  72   5  2
```

```
3  12   149 12.6 74   5  3
```

```
4  18   313 11.5 62   5  4
```

```
5  NA    NA 14.3 56   5  5
```

```
6  28    NA 14.9 66   5  6
```

```
## month day variable value
```

```
## 1  5  1 ozone  41
```

2 5 2 ozone 36

3 5 3 ozone 12

4 5 4 ozone 18

5 5 5 ozone NA

6 5 6 ozone 28

FILE MANUPULATION IN R

Exercise

1.Consider the following data present. Create this file using windows notepad . Save the file as input.csv using the save As All files(.) option in notepad.

Use appropriate R commands to read input.csv file.

Analyze the CSV File and compute the following.

Get the maximum salary

Get the details of the person with max salary

Get all the people working in IT department

Get the persons in IT department whose salary is greater than 600

Get the people who joined on or after 2014

Get the people who joined on or after 2014 and write the output onto a file called output.csv

program:-

```
print(getwd())  
setwd("/web/com")  
print(getwd())  
data <- read.csv("input.csv")  
print(data)  
data <- read.csv("input.csv")  
sal <- max(data$salary)  
print(sal)  
data <- read.csv("input.csv")  
sal <- max(data$salary)  
retval <- subset(data, salary == max(salary))  
print(retval)  
data <- read.csv("input.csv")  
retval <- subset( data, dept == "IT")  
print(retval)  
data <- read.csv("input.csv")  
info <- subset(data, salary > 600 & dept == "IT")  
print(info)  
data <- read.csv("input.csv")  
retval <- subset(data, as.Date(start_date) > as.Date("2014-01-01"))  
print(retval)
```

output:-

```
id, name, salary, start_date, dept  
1  1  Rick   623.30  2012-01-01  IT
```

2	2	Dan	515.20	2013-09-23	Operations
3	3	Michelle	611.00	2014-11-15	IT
4	4	Ryan	729.00	2014-05-11	HR
5	NA	Gary	843.25	2015-03-27	Finance
6	6	Nina	578.00	2013-05-21	IT
7	7	Simon	632.80	2013-07-30	Operations
8	8	Guru	722.50	2014-06-17	Finance

[1] 843.25

	id	name	salary	start_date	dept
--	----	------	--------	------------	------

5	NA	Gary	843.25	2015-03-27	Finance
---	----	------	--------	------------	---------

	id	name	salary	start_date	dept
--	----	------	--------	------------	------

1	1	Rick	623.3	2012-01-01	IT
---	---	------	-------	------------	----

3	3	Michelle	611.0	2014-11-15	IT
---	---	----------	-------	------------	----

6	6	Nina	578.0	2013-05-21	IT
---	---	------	-------	------------	----

	id	name	salary	start_date	dept
--	----	------	--------	------------	------

1	1	Rick	623.3	2012-01-01	IT
---	---	------	-------	------------	----

3	3	Michelle	611.0	2014-11-15	IT
---	---	----------	-------	------------	----

	id	name	salary	start_date	dept
--	----	------	--------	------------	------

3	3	Michelle	611.00	2014-11-15	IT
---	---	----------	--------	------------	----

4	4	Ryan	729.00	2014-05-11	HR
---	---	------	--------	------------	----

5	NA	Gary	843.25	2015-03-27	Finance
---	----	------	--------	------------	---------

8	8	Guru	722.50	2014-06-17	Finance
---	---	------	--------	------------	---------