

# R

*Bill Last Updated:*

*10 April, 2020*



# Contents

<b>Preface: Motivation</b>	<b>5</b>
<b>1 apply, lapply, sapply</b>	<b>7</b>
1.1 apply . . . . .	7
1.2 lapply . . . . .	7
1.3 sapply . . . . .	8
<b>2 C</b>	<b>9</b>
<b>3 GLM and GLiM</b>	<b>11</b>
<b>4 Tidyverse</b>	<b>15</b>
4.1 Slice . . . . .	15
4.2 Group_by . . . . .	16
4.3 Filter . . . . .	17



# Preface: Motivation

All the notes I have done here are about R. While I have tried my best, probably there are still some typos and errors. Please feel free to let me know in case you find one. Thank you!



# Chapter 1

## apply, lapply, sapply

### 1.1 apply

```
m_trying <- matrix(C<-(1:10),nrow=2, ncol=5)
m_trying
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    3    5    7    9
## [2,]    2    4    6    8   10
```

```
## Operating on the columns
```

```
apply(m_trying, 2, sum)
```

```
## [1]  3  7 11 15 19
```

```
## Operating on the rows
```

```
apply(m_trying, 1, sum)
```

```
## [1] 25 30
```

### 1.2 lapply

“lapply returns a list of the same length as X, each element of which is the result of applying FUN to the corresponding element of X.”

lapply operates on lists. Thus, as we can see below, even if m\_trying is not a list, each cell becomes a list.

```
results1<-lapply(m_trying,sum)
str(results1)
```

```
## List of 10
```

```
## $ : int 1
## $ : int 2
## $ : int 3
## $ : int 4
## $ : int 5
## $ : int 6
## $ : int 7
## $ : int 8
## $ : int 9
## $ : int 10
```

```
is.list(results1)
```

```
## [1] TRUE
```

### 1.3 sapply

“sapply() function takes list, vector or data frame as input and gives output in vector or matrix.”

```
results2<-sapply(m_trying, sum)
str(results2)
```

```
## int [1:10] 1 2 3 4 5 6 7 8 9 10
```

```
is.list(results2)
```

```
## [1] FALSE
```

```
is.matrix(results2)
```

```
## [1] FALSE
```

```
is.data.frame(results2)
```

```
## [1] FALSE
```

```
is.vector(results2)
```

```
## [1] TRUE
```



## Chapter 2

# C

```
mydata1<-matrix(runif(4*2),4,2)
mydata1
```

```
##           [,1]      [,2]
## [1,] 0.1969175 0.2054529
## [2,] 0.6757720 0.9610585
## [3,] 0.5566467 0.3911608
## [4,] 0.7862333 0.4072791
```

```
str(mydata1)
```

```
##  num [1:4, 1:2] 0.197 0.676 0.557 0.786 0.205 ...
```

```
mydata2<-c(mydata1)
mydata2
```

```
## [1] 0.1969175 0.6757720 0.5566467 0.7862333 0.2054529 0.9610585 0.3911608
## [8] 0.4072791
```

```
str(mydata2)
```

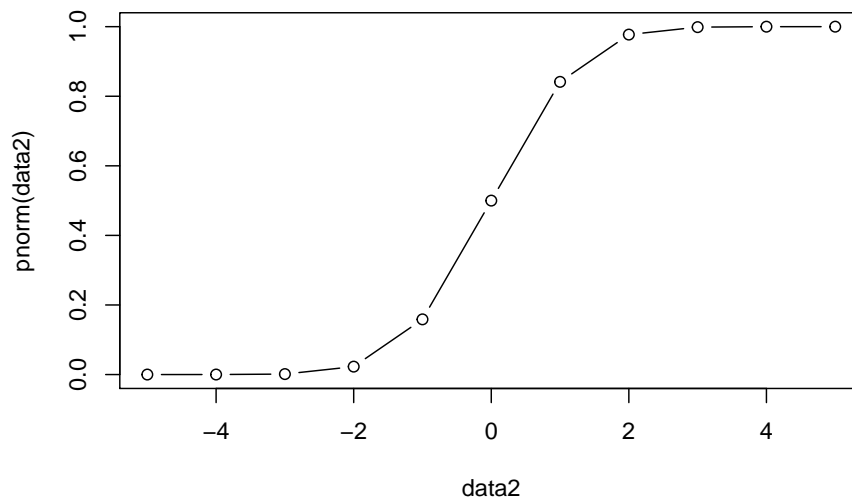
```
##  num [1:8] 0.197 0.676 0.557 0.786 0.205 ...
```



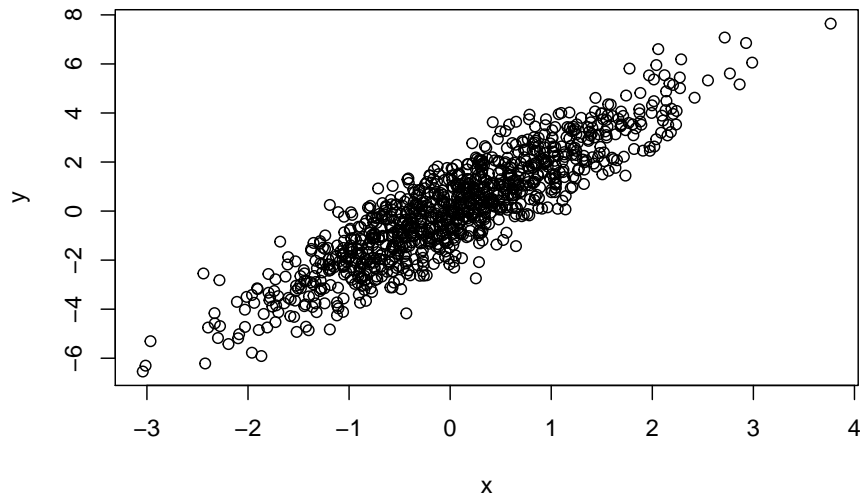
## Chapter 3

# GLM and GLiM

```
data2<-seq(-5,5,1)  
plot(data2,pnorm(data2),type = "b")
```

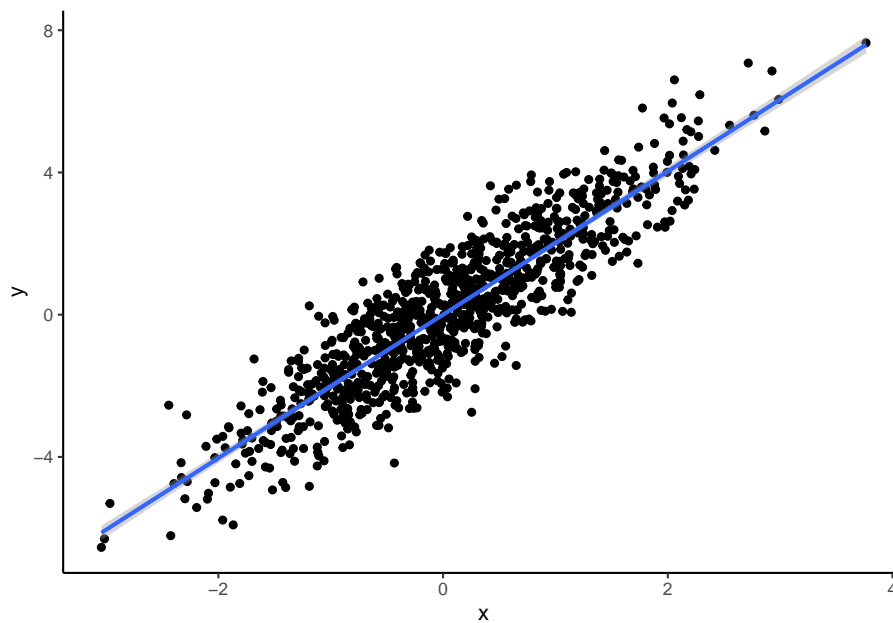


```
b_1<-2  
x<-rnorm(1000)  
y=b_1*x+rnorm(1000)  
plot(x,y)
```



```
data_1<-as.data.frame(cbind(y,x))

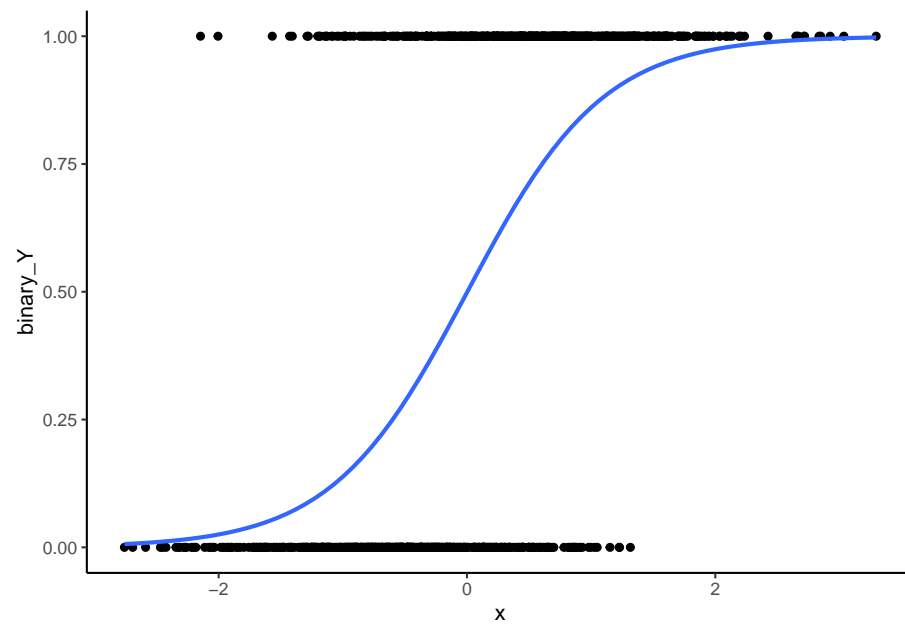
library(ggplot2)
ggplot(data_1,aes(x, y)) +
  geom_point() +
  geom_smooth(method = "lm")+theme_classic()
```



```
x<-rnorm(1000)
y=b_1*x+rnorm(1000)
mu=exp(y)/(1+exp(y))

u<-runif(1000)
binary_Y<-ifelse(mu>u,1,0)
data_2<-as.data.frame(cbind(binary_Y,x))

library(ggplot2)
ggplot(data_2,aes(x, binary_Y)) +
  geom_point() +
  geom_smooth(method = "glm", method.args = list(family = "binomial"),
             se = FALSE)+theme_classic()
```



## Chapter 4

# Tidyverse

### 4.1 Slice

<https://dplyr.tidyverse.org/reference/slice.html>

mtcars

##	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
## Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
## Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
## Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
## Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
## Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
## Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
## Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
## Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
## Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
## Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
## Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
## Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
## Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
## Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
## Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
## Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
## Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
## Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
## Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
## Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
## Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
## Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
## AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2

```
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41 0 0   3   4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05 0 0   3   2
## Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90 1 1   4   1
## Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70 0 1   5   2
## Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90 1 1   5   2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50 0 1   5   4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50 0 1   5   6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60 0 1   5   8
## Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60 1 1   4   2
```

```
library(tidyverse)
slice(mtcars,1L)
```

```
##   mpg cyl disp  hp drat   wt  qsec vs am gear carb
## 1  21   6  160 110  3.9 2.62 16.46 0  1    4    4
```

```
slice(mtcars,n())
```

```
##   mpg cyl disp  hp drat   wt  qsec vs am gear carb
## 1 21.4   4  121 109 4.11 2.78 18.6  1  1    4    2
```

Group\_by just group row with same values (w.r.t a certain column) together. It does not reduce row number.

## 4.2 Group\_by

```
by_cyl<-group_by(mtcars,cyl)
```

```
by_cyl
```

```
## # A tibble: 32 x 11
## # Groups:   cyl [3]
##   mpg   cyl  disp    hp  drat    wt  qsec    vs  am  gear  carb
## * <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  21     6   160   110   3.9  2.62  16.5    0   1    4     4
## 2  21     6   160   110   3.9  2.88  17.0    0   1    4     4
## 3 22.8    4   108    93   3.85  2.32  18.6    1   1    4     1
## 4 21.4    6   258   110   3.08  3.22  19.4    1   0    3     1
## 5 18.7    8   360   175   3.15  3.44  17.0    0   0    3     2
## 6 18.1    6   225   105   2.76  3.46  20.2    1   0    3     1
## 7 14.3    8   360   245   3.21  3.57  15.8    0   0    3     4
## 8 24.4    4   147.    62   3.69  3.19  20      1   0    4     2
## 9 22.8    4   141.    95   3.92  3.15  22.9    1   0    4     2
## 10 19.2    6   168.   123   3.92  3.44  18.3    1   0    4     4
## # ... with 22 more rows
```



```
nrow(mtcars)
```

```
## [1] 32
```

```
nrow(by_cyl)
```

```
## [1] 32
```

## 4.3 Filter

You can also use **filter** to do the same job as **slice**. The only difference is that you need to provide a variable name.

```
filter(mtcars, row_number()==1L)
```

```
##   mpg cyl disp  hp drat   wt  qsec vs am gear carb
## 1  21   6  160 110  3.9 2.62 16.46  0  1    4    4
```

```
filter(mtcars, row_number()==n())
```

```
##   mpg cyl disp  hp drat   wt  qsec vs am gear carb
## 1 21.4   4  121 109 4.11 2.78 18.6  1  1    4    2
```