

Foundations of R Software
Prof. Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture - 44
Data frames: Combining and Merging

Hello friends, welcome to the course Foundations of R Software. Now, in this lecture we are once again going to consider the topic of Data frame and we are going to learn that how you can combine different data frames in a single data frame. Now, it is possible that suppose you are getting some data frame from different sources and you need to merge them together.

Now, this combining or merging that can be done in different ways, the first option is this whatever the data frame suppose 1 comes here, 2 comes here and then both are merged horizontally. Second option is data 1 comes here, data frame 2 comes here and both are merged vertically.

And third option is that in case if the two data frame they have some variable in common then possibly what would you like to have? You would you would not like to have that the same variable should occur in both the data frames when you are trying to combine it.


So, you would like to use it only once and if there are more than one such variables or you have a choice that you want to combine the data set with respect to which variable that has to be controlled. So, these are different types of questions which arise when you are trying to combine different data sets or data frame. So, these are the type of questions which we are going to understand in the lecture today.


And after this I will stop with the topic of data frame also. Well, as I said in the last lecture towards the end that ok, in data frame there are many many operations and it is not practically possible for me to cover here all the operation. But I have chosen some common operation to give you feeling to give you an idea that how the spreadsheets can be handled in the R software, but now the success of using the data frame and its related topics, its command etc. that depends completely on you.


(Refer Slide Time: 02:31)

Data Frames: Combining

□ There are three main techniques :

`cbind()` - combining the columns of two data frames side-by-side. 

`merge()` - joining two data frames using a common column. 

`rbind()` - stacking two data frames on top of each other, appending one to the other. 

So, we today begin our lecture and we try to take some examples to explain that how you can combine the data. So, you see, when you want to combine the data frames then there are three possible ways and for that we have three different commands, one is that you want to combine the two data frames with respect to their columns and the two data frames are side by side just like this. And second option is that if both the data frames have got variable which is common between the two then you would like to join them here like this.

See, whatever is the common variable here that comes over here, right. So, the common variable does not occur in two data frame, but it appears only once. And third option is that you can simply stack them together over each other, right and such that one is appending other. So, it is like this one, right. So, now, how to get it done and what are the commands that is what we have to understand.

So, when you want to combine the two data frames side by side horizontally then our command is `cbind` `cbind` all in lower case alphabets, right and within parentheses we try to give the name of the data frame with some other options. Similarly, when you want to combine the two data frames using a common column then our command is `merge`, `merge` all in a lower case alphabets and within the brackets within the parentheses you have to write down the name of the data frames and then you have to give other options, right.

Similarly, when you want to combine the data frames such that they are stacking over each other like this one in a vertical direction I would say then the command here is `rbind` `r b i n d` all in lower case alphabet.

(Refer Slide Time: 04:36)

Data Frames: Combining

The command `cbind` horizontally merges two data frames side by side.

Example: Create two data frames as follows:

```
df1=data.frame(state=c("UP", "MP", "AP", "JK"),
               popnsize=c(1000, 2000, 3000, 4000))
df2=data.frame(state=c("UP", "MP", "AP", "JK"),
               samplesize=c(100, 200, 300, 400),
               surveycompleted=c("Yes", "No", "Yes", "No"))
```

So, now we try to take here some examples and through those examples I try to explain you the application of these three commands, right. So, first I try to take the here command `c bind` which horizontally merges two data frames side by side, right. I am clicking here example of two data frame, but you can take actually more. So, that is why I am saying here two. So, for that, what we try to do?

We simply try to create here two data frames and so that we can understand that what is really happening. So, I try to create here a data frame whose name is `df 1`; that means, `d` means data frame is `f` and `1` is `df 1`, right. And in which I try to take here two variables here say one is here is state or two columns here one here is a state and another here is population size `p o p n s i z e`.

So, it is like this type of data set which is trying to indicate here that these are here states and then here is the population size. So, I am trying to take the states here “UP”, “MP”, “AP”, “JK” that is Uttar Pradesh, Madhya Pradesh, Andhra Pradesh and Jammu and Kashmir, right and then their corresponding populations are here 1000, 2000, 3000, 4000 respectively, right.

And similarly, I try to take here one more data frame and I try to create it using two different variable the three different variable and one of them is going to be common. So, this is df 2 in which I try to take here the state as say “UP”, “MP”, “AP”, “JK”. So, this is like this one that here I have here three columns, one here is a state and then I try to take here sample size. So, this is here in the second column it is sample size and then in the third column this is survey completed, right.

So you can imagine that it is a sort of data set and yeah means it is related to a survey in these four states where they are trying to specify the population size and the sample size, right. So, and then these states are “UP”, “MP”, “AP”, “JK”, sample sizes from states are 100, 200, 300, 400 and then there is a status that is string in terms of “YES” or “NO” so, “YES”, “NO”, “YES”, “NO”, right. So, if you try to look here that there is a common variable here is state this is here common, right.

(Refer Slide Time: 07:24)

```
Data Frames: Combining  
> df1  
  state popnsize  
1    UP    1000  
2    MP    2000  
3    AP    3000  
4    JK    4000  
  
> df2  
  state samplesize surveycompleted  
1    UP         100             Yes  
2    MP         200             No  
3    AP         300             Yes  
4    JK         400             No
```

So, now if you try to see this data frame 1 and data frame 2 that is df 1 and df 2 will look like this, right. So, you can see here this state here is common in both df 1 and df 2.

(Refer Slide Time: 07:40)

Data Frames: Combining

```
> cbind(df1,df2)
```

	state	popnsiz	state	samplesize	surveycompleted
1	UP	1000	UP	100	Yes
2	MP	2000	MP	200	No
3	AP	3000	AP	300	Yes
4	JK	4000	JK	400	No

df1 (points to first two columns)
df2 (points to last three columns)

Now, we try to make here different operations. So, first I try to take here c bind. So, c bind and then you have to simply write down here df 1 comma df 2. So, now you can see here whatever was your here df 1 this is coming here like this and then you have here df 2. So, this is your here df 1 and this is your here df 2 and if you want to have a look you can have a look here, right.

So, now, if you try to see both these data frame they are combined horizontally and this c bind operation is not trying to consider because the state is in both the data frame. So, it has to be only once or twice. So, it is just copy and paste that is all, right.

(Refer Slide Time: 08:33)

Data Frames: Combining

```
R Console
```

```
> df1
```

	state	popnsiz
1	UP	1000
2	MP	2000
3	AP	3000
4	JK	4000

```
> df2
```

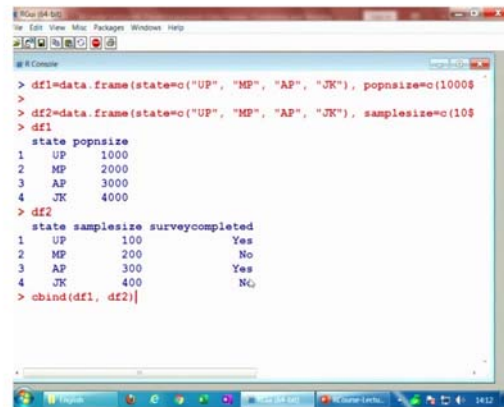
	state	samplesize	surveycompleted
1	UP	100	Yes
2	MP	200	No
3	AP	300	Yes
4	JK	400	No

```
> cbind(df1,df2)
```

	state	popnsiz	state	samplesize	surveycompleted
1	UP	1000	UP	100	Yes
2	MP	2000	MP	200	No
3	AP	3000	AP	300	Yes
4	JK	4000	JK	400	No

And if you try to look here at the screenshot of this outcome it will look like this, this is your here df 1, this is your here df 2 and you can see here, now here this is your here df 1 from here and this is your here df 2 from here, right. So, you can see that both the data frames are joined horizontally ok. So, let us try to first make these operations on the R software, so that I can show you. So, I try to create here this both this data frames, right.

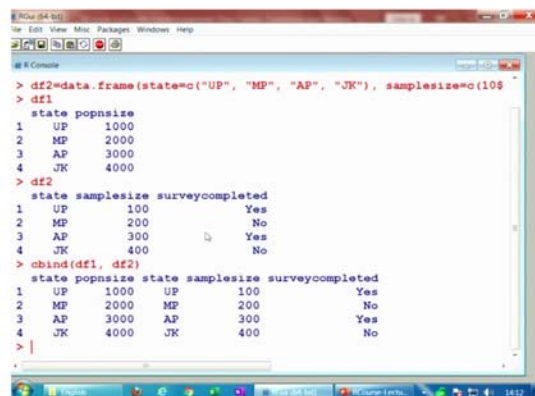
(Refer Slide Time: 09:11)



```
> df1=data.frame(state=c("UP", "MP", "AP", "JK"), popsize=c(1000$
>
> df2=data.frame(state=c("UP", "MP", "AP", "JK"), samplesize=c(10$
> df1
  state popsize
1    UP    1000
2    MP    2000
3    AP    3000
4    JK    4000
> df2
  state samplesize surveycompleted
1    UP         100             Yes
2    MP         200             No
3    AP         300             Yes
4    JK         400             No
> cbind(df1, df2)
```

So, you can see here df 1 is here like this and df 2 here is like this and then when you are trying to combine them so using the function c bind. So, this is say df 1 comma df 2 and then it comes here like this. So, you can see here this is the same operation which I showed you on the screen shot, right ok.

(Refer Slide Time: 09:25)



```
> df2=data.frame(state=c("UP", "MP", "AP", "JK"), samplesize=c(10$
> df1
  state popsize
1    UP    1000
2    MP    2000
3    AP    3000
4    JK    4000
> df2
  state samplesize surveycompleted
1    UP         100             Yes
2    MP         200             No
3    AP         300             Yes
4    JK         400             No
> cbind(df1, df2)
  state popsize state samplesize surveycompleted
1    UP    1000    UP         100             Yes
2    MP    2000    MP         200             No
3    AP    3000    AP         300             Yes
4    JK    4000    JK         400             No
> |
```

(Refer Slide Time: 09:38)

Data Frames: Merging

□ The command `merge` horizontally merges two data frames by common columns or row names.

Example: Create two data frames as follows:

```
df1=data.frame(state=c("UP", "MP", "AP", "JK"),  
               popsize=c(1000,2000,3000,4000))  
  
df2=data.frame(state=c("UP", "MP", "AP", "JK"),  
               samplesize=c(100,200,300,400),  
               surveycompleted=c("Yes", "No", "Yes", "No"))
```

Variable "state" is common between the two data frames and we want to merge the two data frames with respect to `state`.

Now, after this I try to take here second command which is about merge. So, now, the advantage in merge is that it also works just like c bind, but it takes care of the common columns or the row names, right. So, this merge will merge the two data sets horizontally and it will use the common column or row names. For example, in the same example which you have just taken there is common column name state in this df 1 and df 2 here, right. So, same data frame which we have just created.

(Refer Slide Time: 10:19)

Data Frames: Merging

□ The command `merge` horizontally merges two data frames by common columns or row names.

Usage : `merge(x, y, ...)`

Arguments :

`x, y` : data frames, or objects to be coerced to one.

`by, by.x, by.y` : specifications of the columns used for merging.

`sort` : logical.

`no.dups` : logical indicating that suffixes are appended in more cases to avoid duplicated column names in the result.

So, now if you try to see how you can merge this together so, first you have to write down here `merge m e r g e` which is the command for merging the two data set in the R software. And then inside the parenthesis you have to write down the two data sets separated by commas and after this you have to specify that which of the column of this data frame has to be used for merging.

It is like here `by, by dot x, by dot y` and then after that if you want to give here the option here `sort`; that means, whether you want to sort the data after this or not. So, this is going to give you a true or false and similarly there is here another command here `no dot dups n o dot d u p s`. So, this is also a logical variable indicating that the suffix are appended in more cases to avoid duplicated columns in the result, right. So, this will take care of the duplication. So, anyway I will try to make this presentation and lecture simple.

(Refer Slide Time: 11:18)

```
Data Frames: Merging
> df1
  state popsize
1    UP    1000
2    MP    2000
3    AP    3000
4    JK    4000

> df2
  state samplesize surveycompleted
1    UP         100             Yes
2    MP         200             No
3    AP         300             Yes
4    JK         400             No

> merge(df1, df2, by="state")
  state popsize samplesize surveycompleted
1    AP    3000         300             Yes
2    JK    4000         400             No
3    MP    2000         200             No
4    UP    1000         100             Yes
```

So, I will not take many option, but I will leave it up to you that first you understand how the merging is being done then you have to experiment with the different other options. So, now, if you try to see here this is your here `df 1` which we have just created and this is your here `df 2` which we have just created in which this state is a common name common column name.

So, now I try to use here the command here `merge m e r g e` and then I write down here `df 1, df 2` and after that I use the option here `by` and then I try to specify here the name of

the column with respect to we want to merge it, right. So, I try to give it here the name “state”, s t a t e within the double quotes exactly in the same way as it is given, right now you will see here what will happen here.

Now, state is coming here only once and then this population size which is here in df 1 this is coming here and then after this whatever is the sample size and the outcome of the survey whether survey completed or not this is coming here. So, you can see here this is how the merging has been done whereas, if you try to look into the other case here in this case the state was getting repeated in both the data frame.

(Refer Slide Time: 12:50)

Data Frames: Merging

```
R Console
> df1
  state popsize
1  UP    1000
2  MP    2000
3  AP    3000
4  JK    4000
>
> df2
  state samplesize surveycompleted
1  UP         100             Yes
2  MP         200             No
3  AP         300             Yes
4  JK         400             No
>
> merge(df1,df2,by="state")
  state popsize samplesize surveycompleted
1  AP    3000         300             Yes
2  JK    4000         400             No
3  MP    2000         200             No
4  UP    1000         100             Yes
>
```

So, this is the advantage and if you try to see it on the console also, it will look like this is your here df 1, this is your here df 2 and this is your here the outcome of merge command where this state is appearing only here once. And after that this is here the merge command of df 1 and df 2 of those column which are not common, right. So, let us try to have a look on this operation also in the R software, right ok.

(Refer Slide Time: 13:31)

```

> df1
  state popsize
1  UP     1000
2  MP     2000
3  AP     3000
4  JK     4000
> df2
  state samplesize surveycompleted
1  UP         100             Yes
2  MP         200             No
3  AP         300             Yes
4  JK         400             No
> merge(df1, df2, by="state")
  state popsize samplesize surveycompleted
1  AP     3000         300             Yes
2  JK     4000         400             No
3  MP     2000         200             No
4  UP     1000         100             Yes
>

```

So, let me try to copy here this command. So, you already have this df 1 and df 2. So, you can see here and now if you try to merge here, you can see here that state is coming here because it was common and after this population size from df 1 and sample size and survey computed from df 2 they are coming here. So, you can see that it is not a very difficult operation the only thing is that you have to understand how it is going to work ok.

(Refer Slide Time: 13:57)

Data Frames: Combining vertically

The command `rbind` stacks two data frames on top of each other, appending one to the other

Example: Create two data frames as follows:

```

df1=data.frame(state=c("UP", "MP", "AP", "JK"), popsize=c(1000,2000,3000,4000))
df2=data.frame(state=c("Bihar", "Delhi", "Punjab"), popsize =c(100,200,300))

```

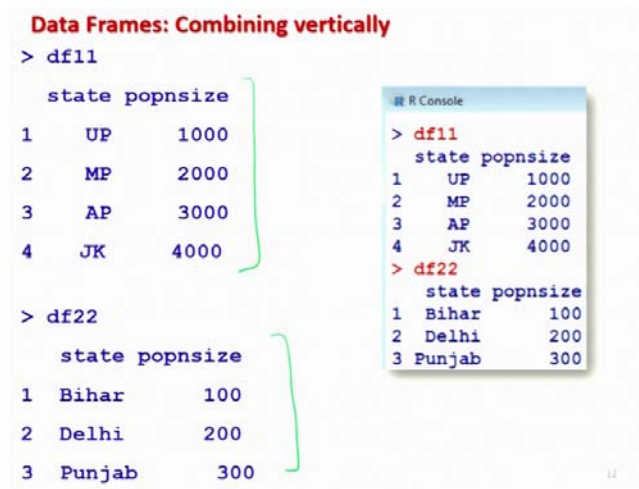
Now, after this I try to take the last option that is r bind. So, r bind is used to join the data frames vertically; that means, this will stick the two data frames on top of each other

appending one to another, right it is just like this, right. So, we try to consider here three these two data frames and now they are different than others I have created it artificially, so that I can explain you this concept easily.

So, I try to consider here data frame by creating by considering two variables and creating it. So, I try to take here state which are “UP”, “MP”, “AP” and “JK” and their population size are 1000, 2000, 3000, 4000 and this data frame is like a df 1. And then after that I have considered here one more data frame df 2 which is constructed by considering the state as “Bihar”, “Delhi” and “Punjab” and population size is 100, 200, 300.

So, you can see here the difference between this case and another case is; that means, earlier you had a data frame 1 like this and data frame 2 like this which has more number of column, but now I wanted to make it the same number of columns, so that they can be joined vertically without any mistake. So, that is why I have to create these two data frame separately.

(Refer Slide Time: 15:24)



Data Frames: Combining vertically

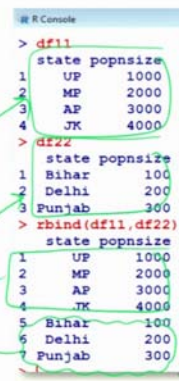
```
> df1
  state popsize
1   UP    1000
2   MP    2000
3   AP    3000
4   JK    4000

> df22
  state popsize
1 Bihar    100
2 Delhi    200
3 Punjab   300
```

The screenshot shows an R console window with the title "R Console". It displays the creation of two data frames, df1 and df22. df1 has columns 'state' and 'popsize' with rows for UP (1000), MP (2000), AP (3000), and JK (4000). df22 has columns 'state' and 'popsize' with rows for Bihar (100), Delhi (200), and Punjab (300). Green brackets are drawn around the data rows of both data frames to highlight their structure.

(Refer Slide Time: 15:33)

```
Data Frames: Combining vertically  
> rbind(df11,df22)  
  state popnsiz  
1    UP    1000  
2    MP    2000  
3    AP    3000  
4    JK    4000  
5  Bihar    100  
6  Delhi    200  
7  Punjab    300
```

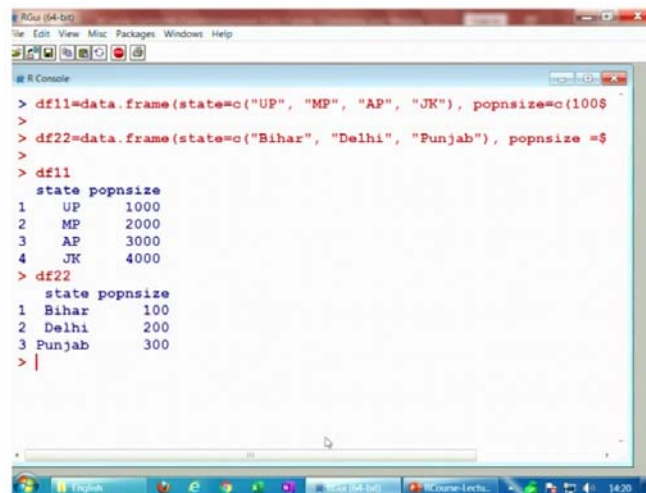


```
# R Console  
> df11  
  state popnsiz  
1    UP    1000  
2    MP    2000  
3    AP    3000  
4    JK    4000  
> df22  
  state popnsiz  
1  Bihar    100  
2  Delhi    200  
3  Punjab    300  
> rbind(df11,df22)  
  state popnsiz  
1    UP    1000  
2    MP    2000  
3    AP    3000  
4    JK    4000  
5  Bihar    100  
6  Delhi    200  
7  Punjab    300
```

So, now if you try to see these two data frames will look like this, right and now in case if you try to use here the command here r bind then you can see here that they are joined together.

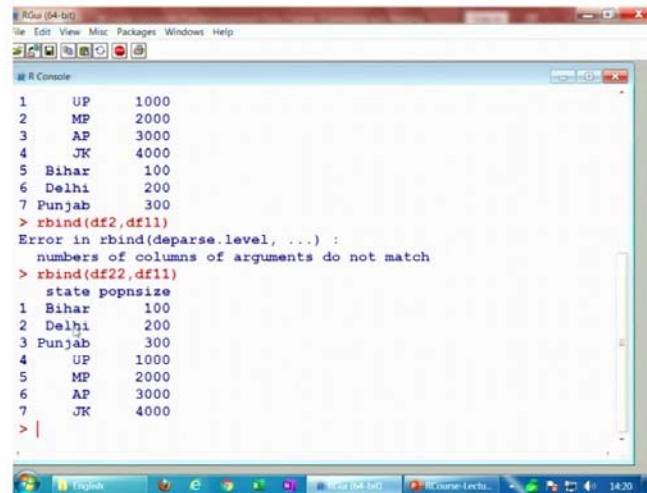
And it is like this, you can see here this is your here df 1, this is your here df 2 and if you try to see here this is your here df 1 from here and this is your here df 2 from here, right and this is here the outcome. So, you can see here both the data frames are joined together. And now this is a new data frame in which you can make different types of operation, right.

(Refer Slide Time: 16:20)



```
RGui (64-bit)  
File Edit View Misc Packages Windows Help  
# R Console  
> df11=data.frame(state=c("UP", "MP", "AP", "JK"), popnsiz=c(1000  
>  
> df22=data.frame(state=c("Bihar", "Delhi", "Punjab"), popnsiz =3  
>  
> df11  
  state popnsiz  
1    UP    1000  
2    MP    2000  
3    AP    3000  
4    JK    4000  
> df22  
  state popnsiz  
1  Bihar    100  
2  Delhi    200  
3  Punjab    300  
> |
```

(Refer Slide Time: 16:37)



```
RGui (64-bit)
File Edit View Misc Packages Windows Help

# R Console

1   UP    1000
2   MP    2000
3   AP    3000
4   JK    4000
5   Bihar  100
6   Delhi  200
7   Punjab 300
> rbind(df2,df11)
Error in rbind(deparse.level, ...) :
  numbers of columns of arguments do not match
> rbind(df22,df11)
  state popsize
1   Bihar    100
2   Delhi    200
3   Punjab   300
4     UP     1000
5     MP     2000
6     AP     3000
7     JK     4000
> |
```

So, let me try to show you these two examples on the R console also, so that you can understand what is really happening. So, this is your here df 1 1 and this is your here df 2 2, right and then I try to merge it using the command r bind. Try to see here it is like that this first four values in the df 1 1 they are coming here and this three values in the df 2 2 they are coming here in this at the end.

So, they are stacked together and if you try to actually if you try to change the order here then you will see what will happen, I try to change it here, here 1 1 and here 2 2, right. So, now, you can this is 2 2, right. So, if you try to see here now this has been reversed mean earlier Bihar, Delhi and Punjab we are coming in the bottom, now they are coming in the top and after that yeah. So, that is pretty straight forward now, right, ok.

So, now we come to an end to this lecture here and as you can see that was a pretty small lecture and we have learnt a very simple operation that how you are going to merge two different data frames together. Now, it is your turn try to take some more data frames more than two and try to see try to make some columns common and then try to see how you can handle them. And similarly try to look into the help and try to see there are various options which are given here which can be used to handle more complicated situation.

My objective was very simple that I wanted to give you an idea that number 1 the merging is possible in R software when you are trying to deal with data frames and you can do it very easily without any problem.

So, now, I will stop in this lecture with the topic of data frame also. So, as I said in the beginning there are many things which are left, but now I will leave it up to you that how much you want to learn and depending on your needs actually you can choose what commands are going to be useful for you, you try to practice them. And I will see you in the next lecture till then goodbye.