

# Using BaseR

Mike Keating

## Task 1: Basic Vector Practice

### Question 1:

Create two vectors named pre and post. One vector corresponding to the pre measurements and one to the post measurements.. Create two vectors named pre and post. One vector corresponding to the pre measurements and one to the post measurements.

```
pre <- c(130, 128, 116, 124, 133, 134, 118, 126, 114, 127,  
        141, 138, 128, 140, 137, 131, 120, 128, 139, 135)  
  
post <- c(114, 98, 113, 99, 107, 116, 113, 111, 119, 117,  
        101, 119, 130, 122, 106, 106, 124, 102, 117, 113)
```

### Question 2:

Assign names to the vector elements using the paste() function. Note that names() can be overwritten by a character vector.

```
names <- paste("Subject", 1:20, sep="_")  
# Assign names  
names(pre) <- names  
names(post) <- names
```

### Question 3:

Calculate the change in blood pressure for each patient.

```
# Change in blood pressure
diff_bp <- pre - post
diff_bp
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
16	30	3	25	26	18	5
Subject_8	Subject_9	Subject_10	Subject_11	Subject_12	Subject_13	Subject_14
15	-5	10	40	19	-2	18
Subject_15	Subject_16	Subject_17	Subject_18	Subject_19	Subject_20	
31	25	-4	26	22	22	

#### Question 4:

Calculate the average decrease in blood pressure across all patients.

```
avg_change_all <- mean(diff_bp)
```

#### Question 5

Determine which patients experienced a decrease in blood pressure after treatment (a positive change). Use the `which()` function to just return the indices (and names) associated with this type of change.

```
index_patients_decreased_bp <- which(diff_bp > 0)
```

#### Question 6

Subset the vector of differences to only return those that have a positive change

```
patients_decreased_bp <- diff_bp[index_patients_decreased_bp]
```

#### Question 7

Calculate the average decrease in blood pressure for those where the blood pressure decreased (positive change).

```
mean_patients_decreased_bp <- mean(patients_decreased_bp)
paste(mean_patients_decreased_bp)
```

```
[1] "20.6470588235294"
```

## Task 2: Basic Data Frame Practice

### Question 1

Create a data frame object with four columns corresponding to your data above: patient, pre\_bp, post\_bp, and diff\_bp.

```
bp_df <- data.frame("Patient" = names,
                    "pre_bp" = pre,
                    "post_bp"=post,
                    "diff_bp"=diff_bp,
                    row.names = NULL)
```

### Question 2

Return only rows where the diff\_bp column is negative.

```
subset(bp_df, diff_bp < 0)
```

	Patient	pre_bp	post_bp	diff_bp
9	Subject_9	114	119	-5
13	Subject_13	128	130	-2
17	Subject_17	120	124	-4

### Question 3

Add a new column to the data frame corresponding to TRUE if the post\_bp is less than 120.

```
bp_df$post_bp_normal <- ifelse(bp_df$post_bp < 120, TRUE, FALSE)
```

Let's quickly check our logic worked.

```
tail(bp_df)
```

	Patient	pre_bp	post_bp	diff_bp	post_bp_normal
15	Subject_15	137	106	31	TRUE
16	Subject_16	131	106	25	TRUE
17	Subject_17	120	124	-4	FALSE
18	Subject_18	128	102	26	TRUE
19	Subject_19	139	117	22	TRUE
20	Subject_20	135	113	22	TRUE

#### Question 4

Finally, print the data frame out nicely in your final document by modifying the code below appropriately.

```
knitr::kable(bp_df)
```

Patient	pre_bp	post_bp	diff_bp	post_bp_normal
Subject_1	130	114	16	TRUE
Subject_2	128	98	30	TRUE
Subject_3	116	113	3	TRUE
Subject_4	124	99	25	TRUE
Subject_5	133	107	26	TRUE
Subject_6	134	116	18	TRUE
Subject_7	118	113	5	TRUE
Subject_8	126	111	15	TRUE
Subject_9	114	119	-5	TRUE
Subject_10	127	117	10	TRUE
Subject_11	141	101	40	TRUE
Subject_12	138	119	19	TRUE
Subject_13	128	130	-2	FALSE
Subject_14	140	122	18	FALSE
Subject_15	137	106	31	TRUE
Subject_16	131	106	25	TRUE
Subject_17	120	124	-4	FALSE
Subject_18	128	102	26	TRUE
Subject_19	139	117	22	TRUE
Subject_20	135	113	22	TRUE

## List Practice

### Question 1

Create a new data frame with these data that is similar to the data frame from task 2 (including the new column). That is, include a patient, pre, post, diff, and normal (less than 120) column using the data above. Name this new data frame `bp_df_placebo`.

```
# I'm going to start the patient numbering after the previous names
names_placebo <- paste("Subject", 21:30, sep="_")
pre_bp_placebo <- c(138, 135, 147, 117, 152, 134, 114, 121, 131, 130)
post_bp_placebo <- c(105, 136, 123, 130, 134, 143, 135, 139, 120, 124)

bp_df_placebo <- data.frame("Patient" = names_placebo,
                           "pre_bp" = pre_bp_placebo,
                           "post_bp" = post_bp_placebo,
                           row.names = NULL)

# Let's make the diff_bp column a different way, this time from existing dataframe
bp_df_placebo$diff_bp <- bp_df_placebo$pre_bp - bp_df_placebo$post_bp

# Normal bp
bp_df_placebo$post_bp_normal <- ifelse(bp_df_placebo$post_bp < 120, TRUE, FALSE)

head(bp_df_placebo)
```

	Patient	pre_bp	post_bp	diff_bp	post_bp_normal
1	Subject_21	138	105	33	TRUE
2	Subject_22	135	136	-1	FALSE
3	Subject_23	147	123	24	FALSE
4	Subject_24	117	130	-13	FALSE
5	Subject_25	152	134	18	FALSE
6	Subject_26	134	143	-9	FALSE

### Question 2

Now create and store a list with two elements:

- 1st element named `treatment` and contains the first data frame you created.
- 2nd element named `placebo` and contains the second data frame you created.

```
bp_list <- list("treatment" = bp_df,
               "placebo" = bp_df_placebo)
```

### Question 3

Access the first list element using three different types of syntax.

```
# By index
bp_list[1]
```

```
$treatment
  Patient pre_bp post_bp diff_bp post_bp_normal
1 Subject_1   130   114     16             TRUE
2 Subject_2   128    98     30             TRUE
3 Subject_3   116   113      3             TRUE
4 Subject_4   124    99     25             TRUE
5 Subject_5   133   107     26             TRUE
6 Subject_6   134   116     18             TRUE
7 Subject_7   118   113      5             TRUE
8 Subject_8   126   111     15             TRUE
9 Subject_9   114   119     -5             TRUE
10 Subject_10  127   117     10             TRUE
11 Subject_11  141   101     40             TRUE
12 Subject_12  138   119     19             TRUE
13 Subject_13  128   130     -2            FALSE
14 Subject_14  140   122     18            FALSE
15 Subject_15  137   106     31             TRUE
16 Subject_16  131   106     25             TRUE
17 Subject_17  120   124     -4            FALSE
18 Subject_18  128   102     26             TRUE
19 Subject_19  139   117     22             TRUE
20 Subject_20  135   113     22             TRUE
```

```
# By index in the other direction
bp_list[-2]
```

```
$treatment
  Patient pre_bp post_bp diff_bp post_bp_normal
1 Subject_1   130   114     16             TRUE
2 Subject_2   128    98     30             TRUE
```

3	Subject_3	116	113	3	TRUE
4	Subject_4	124	99	25	TRUE
5	Subject_5	133	107	26	TRUE
6	Subject_6	134	116	18	TRUE
7	Subject_7	118	113	5	TRUE
8	Subject_8	126	111	15	TRUE
9	Subject_9	114	119	-5	TRUE
10	Subject_10	127	117	10	TRUE
11	Subject_11	141	101	40	TRUE
12	Subject_12	138	119	19	TRUE
13	Subject_13	128	130	-2	FALSE
14	Subject_14	140	122	18	FALSE
15	Subject_15	137	106	31	TRUE
16	Subject_16	131	106	25	TRUE
17	Subject_17	120	124	-4	FALSE
18	Subject_18	128	102	26	TRUE
19	Subject_19	139	117	22	TRUE
20	Subject_20	135	113	22	TRUE

```
# By name
bp_list$treatment
```

	Patient	pre_bp	post_bp	diff_bp	post_bp_normal
1	Subject_1	130	114	16	TRUE
2	Subject_2	128	98	30	TRUE
3	Subject_3	116	113	3	TRUE
4	Subject_4	124	99	25	TRUE
5	Subject_5	133	107	26	TRUE
6	Subject_6	134	116	18	TRUE
7	Subject_7	118	113	5	TRUE
8	Subject_8	126	111	15	TRUE
9	Subject_9	114	119	-5	TRUE
10	Subject_10	127	117	10	TRUE
11	Subject_11	141	101	40	TRUE
12	Subject_12	138	119	19	TRUE
13	Subject_13	128	130	-2	FALSE
14	Subject_14	140	122	18	FALSE
15	Subject_15	137	106	31	TRUE
16	Subject_16	131	106	25	TRUE
17	Subject_17	120	124	-4	FALSE
18	Subject_18	128	102	26	TRUE
19	Subject_19	139	117	22	TRUE

```
20 Subject_20      135      113      22      TRUE
```

#### Question 4

In one line, access the placebo data frame, pre\_bp column.

```
bp_list$placebo$pre_bp
```

```
[1] 138 135 147 117 152 134 114 121 131 130
```

### Task 4: Control Flow Practice

#### Question 1

Classify bp as optimal, borderline, and high.

```
bp_list$treatment$status <- character(20)
bp_list$placebo$status <- character(10)
```

#### Question 2

For the non-placebo data frame (within the list), create a for loop and use if/then/else logic to create the status column's values.

```
for (i in 1:nrow(bp_list$treatment)) {
  post_bp <- bp_list$treatment[i, "post_bp"]
  if (post_bp > 130) {
    bp_list$treatment[i, "status"] <- "high"
  }
  else if (post_bp > 120) {
    bp_list$treatment[i, "status"] <- "borderline"
  }
  else {
    bp_list$treatment[i, "status"] <- "optimal"
  }
}

head(bp_list$treatment)
```



	Patient	pre_bp	post_bp	diff_bp	post_bp_normal	status
1	Subject_1	130	114	16	TRUE	optimal
2	Subject_2	128	98	30	TRUE	optimal
3	Subject_3	116	113	3	TRUE	optimal
4	Subject_4	124	99	25	TRUE	optimal
5	Subject_5	133	107	26	TRUE	optimal
6	Subject_6	134	116	18	TRUE	optimal

### Question 3

Repeat for the placebo dataframe (within the list)

```
for (i in 1:nrow(bp_list$placebo)) {
  post_bp <- bp_list$placebo[i, "post_bp"]
  if (post_bp > 130) {
    bp_list$placebo[i, "status"] <- "high"
  }
  else if (post_bp > 120) {
    bp_list$placebo[i, "status"] <- "borderline"
  }
  else {
    bp_list$placebo[i, "status"] <- "optimal"
  }
}

head(bp_list$placebo)
```

	Patient	pre_bp	post_bp	diff_bp	post_bp_normal	status
1	Subject_21	138	105	33	TRUE	optimal
2	Subject_22	135	136	-1	FALSE	high
3	Subject_23	147	123	24	FALSE	borderline
4	Subject_24	117	130	-13	FALSE	borderline
5	Subject_25	152	134	18	FALSE	high
6	Subject_26	134	143	-9	FALSE	high

## Task 5: Function Writing

### Question 1

```
get_stat <- function(df_list, stat = "mean"){

  my_func <- get(stat)
  cols <- c("post_bp", "pre_bp", "diff_bp")
  values <- c()
  names <- c()
  j <- 1
  for (i in 1:length(df_list)){

    for (col in cols){
      name <- paste( col, "_",names(df_list[i]),"_", stat, sep="")
      value <- my_func(df_list[[i]][,col])
      values[j] <- value
      names [j] <- name
      j = j + 1
    }

  }
  stats <- values
  names(stats) <- names
  return(stats)
}
```

Without specifying stat:

```
result <- get_stat(bp_list)
knitr::kable(result)
```

	x
post_bp_treatment_mean	112.35
pre_bp_treatment_mean	129.35
diff_bp_treatment_mean	17.00
post_bp_placebo_mean	128.90
pre_bp_placebo_mean	131.90
diff_bp_placebo_mean	3.00

Specifying various stats:

```
result <- get_stat(bp_list, "var")
knitr::kable(result)
```

	x
post_bp_treatment_var	74.76579
pre_bp_treatment_var	64.55526
diff_bp_treatment_var	153.68421
post_bp_placebo_var	124.98889
pre_bp_placebo_var	149.87778
diff_bp_placebo_var	341.33333

```
result <- get_stat(bp_list, "sd")
knitr::kable(result)
```

	x
post_bp_treatment_sd	8.646721
pre_bp_treatment_sd	8.034629
diff_bp_treatment_sd	12.396944
post_bp_placebo_sd	11.179843
pre_bp_placebo_sd	12.242458
diff_bp_placebo_sd	18.475209

```
result <- get_stat(bp_list, "max")
knitr::kable(result)
```

	x
post_bp_treatment_max	130
pre_bp_treatment_max	141
diff_bp_treatment_max	40
post_bp_placebo_max	143
pre_bp_placebo_max	152
diff_bp_placebo_max	33

```
result <- get_stat(bp_list, "min")
knitr::kable(result)
```

	x
post_bp_treatment_min	98
pre_bp_treatment_min	114
diff_bp_treatment_min	-5
post_bp_placebo_min	105
pre_bp_placebo_min	114
diff_bp_placebo_min	-21