Chapter 13 - Iterating with purrr

Michael Beebe

---

1. map

Create a list called myList. in this list store:
- A matrix of the numbers 1-10 in 2 rows
- The even numbers from 0-10
- The odd numbers from 1-9
- A matrix of the numbers from 1-9 in 3 rows

```
myList <- list(
  matrix(data = 1:10, nrow = 2),
  c(2, 4, 5, 6, 8, 10),
  c(1, 3, 5, 7, 9),
  matrix(data = 1:9, nrow = 3)
)

myList
```

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

Map the mean and sum functions to your list

```
map(myList, mean)
```

```
## [[1]]
## [1] 5.5
```

```
##
## [[2]]
## [1] 5.833333
##
## [[3]]
## [1] 5
##
## [[4]]
## [1] 5
```

```
map(myList, sum)
```

```
## [[1]]
## [1] 55
##
## [[2]]
## [1] 35
##
## [[3]]
## [1] 25
##
## [[4]]
## [1] 45
```

Use the identical function to determine if map and lapply are the same when applied to myList

```
lapply(myList, mean)
```

```
## [[1]]
## [1] 5.5
##
## [[2]]
## [1] 5.833333
##
## [[3]]
## [1] 5
##
## [[4]]
## [1] 5
```

```
lapply(myList, sum)
```

```
## [[1]]
## [1] 55
##
## [[2]]
## [1] 35
##
## [[3]]
## [1] 25
##
## [[4]]
## [1] 45
```

2. map with Specified Types

Using your list - myList - find the following:
- Use map_int to find number of rows or length of each list element

```
map_int(myList, length)
```

```
## [1] 10  6  5  9
```

- Use map_dbl to find the standard deviation

```
map_dbl(myList, sd)
```

```
## [1] 3.027650 2.857738 3.162278 2.738613
```

- Use map_chr to find the class of each element in your list

```
map_chr(myList, typeof)
```

```
## [1] "integer" "double"  "double"  "integer"
```

- Use map_lgl to determine if the number of rows in each element of your list is < 3

```
check <- function(x) is.list(nrow(x)) < 3
map_lgl(myList, check)
```

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

- Create the data frame and list on page 184 of your text. Use map_df to call the function with the list of lengths

```
buildDF <- function(x) {
  data.frame(A=1:x, B=x:1)
}

listOfLengths <- list(3, 4, 1, 5)

listOfLengths %>%
  map(buildDF)
```

```
## [[1]]
##   A B
## 1 1 3
## 2 2 2
## 3 3 1
##
## [[2]]
##   A B
## 1 1 4
```

```
## 2 2 3
## 3 3 2
## 4 4 1
##
## [[3]]
##   A B
## 1 1 1
##
## [[4]]
##   A B
## 1 1 5
## 2 2 4
## 3 3 3
## 4 4 2
## 5 5 1
```

3. Iterating over a data.frame

Use the mtcars built in data set
Find the mean of all of the numeric data in the dataset

```
mtcars %>%
  map(mean)
```

```
## $mpg
## [1] 20.09062
##
## $cyl
## [1] 6.1875
##
## $disp
## [1] 230.7219
##
## $hp
## [1] 146.6875
##
## $drat
## [1] 3.596563
##
## $wt
## [1] 3.21725
##
## $qsec
## [1] 17.84875
##
## $vs
## [1] 0.4375
##
## $am
## [1] 0.40625
##
## $gear
## [1] 3.6875
```

```
##
## $carb
## [1] 2.8125
```

Find the standard deviation of all of the numeric data in the dataset

```
mtcars %>%
  map(sd)
```

```
## $mpg
## [1] 6.026948
##
## $cyl
## [1] 1.785922
##
## $disp
## [1] 123.9387
##
## $hp
## [1] 68.56287
##
## $drat
## [1] 0.5346787
##
## $wt
## [1] 0.9784574
##
## $qsec
## [1] 1.786943
##
## $vs
## [1] 0.5040161
##
## $am
## [1] 0.4989909
##
## $gear
## [1] 0.7378041
##
## $carb
## [1] 1.6152
```