A quick introduction to dplyr

```
library(dplyr)
```

dplyr is a Cran Package with tools for manipulation of data. It is very extensive and has methods to deal with different kinds of data structures, but there are **5 functions** that you can learn quickly and use on your R data frame objects. They are:

1. select(): focus on a subset of variables
2. filter(): focus on a subset of rows
3. mutate(): add new columns
4. summarise(): get summary statistics
5. arrange(): re-order the rows

You can use column names as parameters without having to use the $ operator, which makes code more readable!

1. select()

basic pseudocode: select(your_dataframe, columns you want to select)

example:

```
head(iris) #built-in R data for example
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
#I only want to analyse Species and Sepal Length

myiris <- select(iris, Species, Sepal.Length)

head(myiris)
```

```
##   Species Sepal.Length
## 1  setosa          5.1
## 2  setosa          4.9
## 3  setosa          4.7
## 4  setosa          4.6
## 5  setosa          5.0
## 6  setosa          5.4
```

```
#notice it changes the column ordering to what you chose too
```

2. filter()

My favorite: only get the data that follows a certain condition

basic pseudocode: filter(your_dataframe, true/false condition)

example:

```
#I only want to analyse Species virginica and Petal length > 5

myiris2 <- filter(iris,Species == "virginica",Petal.Length > 5)

head(myiris2)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
## 1          6.3         3.3          6.0         2.5 virginica
## 2          5.8         2.7          5.1         1.9 virginica
## 3          7.1         3.0          5.9         2.1 virginica
## 4          6.3         2.9          5.6         1.8 virginica
## 5          6.5         3.0          5.8         2.2 virginica
## 6          7.6         3.0          6.6         2.1 virginica
```

3. mutate()

basic pseudocode: mutate(your_dataframe, new column to be added)
example:

```
#I will add a column of random uniform draws

myiris3 <- mutate(iris, runif(length(Species)))

head(myiris3)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
##   runif(length(Species))
## 1              0.9416964
## 2              0.9365674
## 3              0.3823653
## 4              0.2321037
## 5              0.6233210
## 6              0.2651123
```

4. summarise()

summarise(factors to group, summary statistics to use)

```
summarise(group_by(iris,Species),mean(Sepal.Length))
```

```
## # A tibble: 3 × 2
##      Species `mean(Sepal.Length)`
##       <fctr>                <dbl>
## 1     setosa                5.006
## 2 versicolor                5.936
## 3  virginica                6.588
```

```r
summarise(group_by(iris,Species),sd(Sepal.Length))
```

```
## # A tibble: 3 × 2
##      Species `sd(Sepal.Length)`
##       <fctr>              <dbl>
## 1     setosa          0.3524897
## 2 versicolor          0.5161711
## 3  virginica          0.6358796
```

5. arrange()

arrange(yourdata, columns that should be sorted by, in order

```r
head(arrange(iris, Sepal.Length))
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          4.3         3.0          1.1         0.1  setosa
## 2          4.4         2.9          1.4         0.2  setosa
## 3          4.4         3.0          1.3         0.2  setosa
## 4          4.4         3.2          1.3         0.2  setosa
## 5          4.5         2.3          1.3         0.3  setosa
## 6          4.6         3.1          1.5         0.2  setosa
```

```r
head(arrange(iris, Sepal.Length, Petal.Length))
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          4.3         3.0          1.1         0.1  setosa
## 2          4.4         3.0          1.3         0.2  setosa
## 3          4.4         3.2          1.3         0.2  setosa
## 4          4.4         2.9          1.4         0.2  setosa
## 5          4.5         2.3          1.3         0.3  setosa
## 6          4.6         3.6          1.0         0.2  setosa
```

```r
head(arrange(iris, desc(Sepal.Length)))
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
## 1          7.9         3.8          6.4         2.0 virginica
## 2          7.7         3.8          6.7         2.2 virginica
## 3          7.7         2.6          6.9         2.3 virginica
## 4          7.7         2.8          6.7         2.0 virginica
## 5          7.7         3.0          6.1         2.3 virginica
## 6          7.6         3.0          6.6         2.1 virginica
```