Lesson 1 Solutions

Stefanie Molin

April 12, 2017

1. Calculate the average (`mean()`) and standard deviation (`sd()`) of the `speed` column in the `cars` dataframe.

```
mean(cars$speed)
```

```
## [1] 15.4
```

```
sd(cars$speed)
```

```
## [1] 5.287644
```

2. Create a new column in the dataframe `cars` called `time` defined as `dist`/`speed`

```
cars$time <- cars$dist/cars$speed
head(cars)
```

```
##   speed dist      time
## 1     4    2 0.5000000
## 2     4   10 2.5000000
## 3     7    4 0.5714286
## 4     7   22 3.1428571
## 5     8   16 2.0000000
## 6     9   10 1.1111111
```

3. A Fibonacci sequence is a series of numbers in which each number is the sum of the two preceding numbers (1, 1, 2, 3, 5, 8, ...). Write a function `fibonacci()` that takes one argument `n`, the size of the sequence you want to print and outputs a Fibonacci sequence of that length. To make this easier, you can assume that the user always properly implements this function (they always provide an n of 1 or greater).

   Extra credit: Have the function handle cases for all numeric values of n and notify the user of an error i.e. if n <= 0 is given.

   Hint: Intialize a vector using `numeric(n)` to store your sequence, and have your function *return* the vector.

```
fibonacci <- function(n){

  # handle missing n input
  if(missing(n)){
    stop("Please provide the length of the sequence you want.")
  }

  # handle invalid inputs
  if(n <= 0){
    stop("Not a valid input for n. Values must be greater than or equal to 1.")
  }

  # initalize a vector of size n
  fibonacci <- numeric(n)

  # handle special cases of n = 1 and 2
  if(n >= 1){
    fibonacci[1] <- 1

    if(n >= 2){
      fibonacci[2] <- 1

      if(n >= 3){
        # loop through for a series of length n
        for(i in 3:n){
          fibonacci[i] <- fibonacci[i - 2] + fibonacci[i - 1]
        }
      }
    }
  }

  return(fibonacci)
}

# check that we can't break the function
fibonacci()
```

```
## Error in fibonacci(): Please provide the length of the sequence you want.
```

```
fibonacci(-1)
```

```
## Error in fibonacci(-1): Not a valid input for n. Values must be greater than or equal to 1.
```

```
# check fibonacci on valid inputs
fibonacci(1)
```

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

```
fibonacci(2)
```

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

```
fibonacci(5)
```

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

```
fibonacci(6)
```

```
## [1] 1 1 2 3 5 8
```