

Assignment 7

Max Wagner

March 17, 2016

Part I, base v custom

```
numbers <- runif(100, 0, 10)

# mean without using mean
eValue <- function(vector) {
  return(sum(vector) / length(vector))
}

# test if its the same thing
eValue(numbers) == mean(numbers)

## [1] TRUE

print(c("custom:", eValue(numbers)))

## [1] "custom:"          "4.85561225134879"

print(c("base:", mean(numbers)))

## [1] "base:"             "4.85561225134879"

# sd without using sd()
std <- function(vector) {
  return(sqrt(sum((vector - eValue(vector))^2) / (length(vector) - 1)))
}

# test if we get the same, if they are slightly different print the values
std(numbers) == sd(numbers)

## [1] TRUE

print(c("custom:", std(numbers)))

## [1] "custom:"          "2.95212221055843"

print(c("base:", sd(numbers)))

## [1] "base:"            "2.95212221055843"
```

Part II, rolling stats

```

rolling <- function(vector, stats = list()){
  if (length(stats) > 0) { # a stats list exists
    oldn <- stats$n
    stats$n <- stats$n + length(vector) # new n
    stats$allmeans <- c(stats$allmeans, eValue(vector)) # list of means
    stats$mean <- ((stats$mean * oldn) + (eValue(vector) * length(vector))) / stats$n # new mean from t
    stats$allsds <- c(stats$allsds, std(vector)) # list of sds
    stats$sd <- sqrt(sum(stats$allsds) / length(stats$allsds)) # new sd from the list of sds
  } else { # no stats list is input
    stats$n <- length(vector)
    stats$mean <- eValue(vector)
    stats$sd <- std(vector)
    stats$allmeans <- stats$mean
    stats$allsds <- stats$sd
  }
  return(list(n = stats$n, mean = stats$mean, sd = stats$sd,
             allmeans = stats$allmeans, allsds = stats$allsds))
}

```

Let's test with a few different samples

```

vector <- runif(100, 0, 10)
vector2 <- runif(50, 0, 10)
vector3 <- runif(200, 0, 10)
stats <- c()
stats <- rolling(vector);stats

```

```

## $n
## [1] 100
##
## $mean
## [1] 5.169953
##
## $sd
## [1] 3.035704
##
## $allmeans
## [1] 5.169953
##
## $allsds
## [1] 3.035704

```

```

stats <- rolling(vector2, stats);stats

```

```

## $n
## [1] 150
##
## $mean
## [1] 5.15452
##
## $sd
## [1] 1.74267

```

```
##  
## $allmeans  
## [1] 5.169953 5.123653  
##  
## $allsds  
## [1] 3.035704 3.038095
```

```
stats <- rolling(vector3, stats);stats
```

```
## $n  
## [1] 350  
##  
## $mean  
## [1] 5.019318  
##  
## $sd  
## [1] 1.741974  
##  
## $allmeans  
## [1] 5.169953 5.123653 4.917916  
##  
## $allsds  
## [1] 3.035704 3.038095 3.029624
```