Forecasting by Exponential Smoothing

Ananda Biswas

2023-06-14

```
our_data <- c(100, 110, 120, 115, 125, 135, 130, 140, 145, 150, 155, 160)

print(length(our_data))
```

```
## [1] 12
```

```
print(our_data)
```

```
##  [1] 100 110 120 115 125 135 130 140 145 150 155 160
```

```
test_values_of_alpha <- seq(from = 0, to = 1, by = 0.15)

test_values_of_alpha <- test_values_of_alpha[-1]

print(length(test_values_of_alpha))
```

```
## [1] 6
```

```
print(test_values_of_alpha)
```

```
## [1] 0.15 0.30 0.45 0.60 0.75 0.90
```

```
estimates <- c()

for (i in 1:length(test_values_of_alpha)) {

    alpha <- test_values_of_alpha[i]

    estimates <- append(estimates, our_data[1], after = length(estimates))

    for (j in 2:length(our_data)) {

        next_estimate <- alpha * our_data[j - 1] + (1 - alpha) * estimates[length(estimates)]

        estimates <- append(estimates, next_estimate, after = length(estimates))

    }
}
```

```
estimated_values <- matrix(data = estimates, ncol = length(test_values_of_alpha),
    nrow = 12, byrow = FALSE)

print(estimated_values)
```

```
##           [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
##  [1,] 100.0000 100.0000 100.0000 100.0000 100.0000 100.0000
```

```
##  [2,] 100.0000 100.0000 100.0000 100.0000 100.0000 100.0000
##  [3,] 101.5000 103.0000 104.5000 106.0000 107.5000 109.0000
##  [4,] 104.2750 108.1000 111.4750 114.4000 116.8750 118.9000
##  [5,] 105.8837 110.1700 113.0613 114.7600 115.4688 115.3900
##  [6,] 108.7512 114.6190 118.4337 120.9040 122.6172 124.0390
##  [7,] 112.6885 120.7333 125.8885 129.3616 131.9043 133.9039
##  [8,] 115.2852 123.5133 127.7387 129.7446 130.4761 130.3904
##  [9,] 118.9924 128.4593 133.2563 135.8979 137.6190 139.0390
## [10,] 122.8936 133.4215 138.5410 141.3591 143.1548 144.4039
## [11,] 126.9595 138.3951 143.6975 146.5437 148.2887 149.4404
## [12,] 131.1656 143.3765 148.7836 151.6175 153.3222 154.4440
```

```
error_matrix <- matrix(data = 0, nrow = 12, ncol = length(test_values_of_alpha),
    byrow = TRUE)

for (i in 1:length(test_values_of_alpha)) {

    for (j in 1:12) {
        error_matrix[j, i] <- (estimated_values[j, i] - our_data[j])^2
    }
}

error_matrix <- round(error_matrix, digits = 4)

print(error_matrix)
```

```
##           [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
##  [1,]   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
##  [2,] 100.0000 100.0000 100.0000 100.0000 100.0000 100.0000
##  [3,] 342.2500 289.0000 240.2500 196.0000 156.2500 121.0000
##  [4,] 115.0256  47.6100  12.4256   0.3600   3.5156  15.2100
##  [5,] 365.4310 219.9289 142.5338 104.8576  90.8447  92.3521
##  [6,] 689.0002 415.3852 274.4427 198.6972 153.3340 120.1435
##  [7,] 299.6877  85.8717  16.9042   0.4076   3.6263  15.2404
##  [8,] 610.8197 271.8109 150.3397 105.1724  90.7052  92.3446
##  [9,] 676.3928 273.5942 137.9150  82.8490  54.4789  35.5331
## [10,] 734.7580 274.8459 131.3097  74.6644  46.8574  31.3163
## [11,] 786.2672 275.7239 127.7459  71.5097  45.0417  30.9093
## [12,] 831.4219 276.3392 125.8068  70.2669  44.5934  30.8687
```

```
sum_of_errors <- c()

for (i in 1:length(test_values_of_alpha)) {
    temp <- sum(error_matrix[, i])

    sum_of_errors <- append(sum_of_errors, temp, after = length(sum_of_errors))
}

print(sum_of_errors)
```

```
## [1] 5551.0541 2530.1099 1459.6734 1004.7848  789.2472  684.9180
```

```
print(min(sum_of_errors))
```

```
## [1] 684.918
```

```r
sprintf("best value for alpha is %.2f", test_values_of_alpha[which.min(sum_of_errors)])
```

```
## [1] "best value for alpha is 0.90"
```

```r
alpha = 0.9

final_estimates <- c()

final_estimates <- append(final_estimates, our_data[1], after = length(final_estimates))

for (i in 2:12) {

    new_value <- alpha * our_data[i - 1] + (1 - alpha) * final_estimates[length(final_estimates)]

    final_estimates <- append(final_estimates, new_value, after = length(final_estimates))
}

final_estimates <- data.frame(Months = month.name, Estimates = final_estimates)

print(final_estimates)
```

```
##       Months Estimates
## 1    January  100.0000
## 2   February  100.0000
## 3      March  109.0000
## 4      April  118.9000
## 5        May  115.3900
## 6       June  124.0390
## 7       July  133.9039
## 8     August  130.3904
## 9  September  139.0390
## 10   October  144.4039
## 11  November  149.4404
## 12  December  154.4440
```

```r
alpha = 0.9

prediction <- alpha * 160 + (1 - alpha) * 154.444

prediction
```

```
## [1] 159.4444
```