MSMS - 106

Ananda Biswas

✎ **Consider 1, 2, 3, . . . , 100 as population observations. Take a sample of size 40 (without replacement) by *sample()* function and compute mean, median, variance, mean absolute deviation about mean, median absolute deviation about median, skewness and kurtosis. Implement the functions on your own and use functions provided by R to verify results.**

⊕ *Getting a sample*

```
my_sample <- sample(1:100, size = 40, replace = FALSE)
my_sample
```

```
##  [1] 65 23 14 49 44 57 39 26 34 83 60 48 85 92 21 75 11 58 66 46 89 82 77 62 29
## [26] 98 78 67 43  6  2 33 35 42 96 41 30  4 24 32
```

⊕ *Mean*

To compute mean, we add all the observations and divide the sum by number of observations i.e. $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.

```
my_mean_function <- function(x){

  sample_sum <- 0

  for (i in 1:length(x)) {
    sample_sum <- sample_sum + x[i]
  }

  return(sample_sum / length(x))
}
```

```
my_mean_function(my_sample); mean(my_sample)
```

```
## [1] 49.15
## [1] 49.15
```

⊘ Results matched !!

Median

To compute median, we first sort the sample values and return the middle most value.

```
my_median_function <- function(x){

  for (i in 1:(length(x)-1)) {
    for (j in (i+1):length(x)) {

      if(x[i] > x[j]){
        x[c(i, j)] <- x[c(j, i)]
      }
    }
  }

  if(length(x) %% 2 == 0){
    return((x[length(x)/2] + x[length(x)/2 + 1]) / 2)
  } else{
    return(x[(length(x) + 1) / 2])
  }
}
```

```
my_median_function(my_sample); median(my_sample)

## [1] 45
## [1] 45
```

Results matched !!

Variance

To compute variance, we get the squared deviations, their sum and then we divide the sum by $(n-1)$ i.e. $\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$.

```
my_sample_central_moments_function <- function(x, r){
  temp <- 0

  for (i in 1:length(x)) {
    temp <- temp + (x[i] - my_mean_function(x))^r
  }

  return(temp / (length(x) - 1))
}
```

```
my_sample_central_moments_function(my_sample, 2); var(my_sample)

## [1] 732.8487
## [1] 732.8487
```

Results matched !!

➔ *Mean Absolute Deviation about Mean*

We first obtain the deviations $(x_i - \bar{x})\ \forall i$ and then we calculate their absolute values. Lastly we calculate their mean.

```
deviations_about_mean <- my_sample - my_mean_function(my_sample)

absolute_deviations_about_mean <- c()

for (i in 1:length(deviations_about_mean)) {
  if(deviations_about_mean[i] < 0){
    absolute_deviations_about_mean[i] <- deviations_about_mean[i] * (-1)
  }
  else{
    absolute_deviations_about_mean[i] <- deviations_about_mean[i]
  }
}
```

```
absolute_deviations_about_mean

##  [1] 15.85 26.15 35.15  0.15  5.15  7.85 10.15 23.15 15.15 33.85 10.85  1.15
## [13] 35.85 42.85 28.15 25.85 38.15  8.85 16.85  3.15 39.85 32.85 27.85 12.85
## [25] 20.15 48.85 28.85 17.85  6.15 43.15 47.15 16.15 14.15  7.15 46.85  8.15
## [37] 19.15 45.15 25.15 17.15
```

```
my_mean_function(absolute_deviations_about_mean)

## [1] 22.7225
```

➔ *Median Absolute Deviation about Median*

We first obtain the deviations $(x_i - \widetilde{x})\ \forall i$ and then we calculate their absolute values. Lastly we calculate their median.

```
deviations_about_median <- my_sample - my_median_function(my_sample)

absolute_deviations_about_median <- c()

for (i in 1:length(deviations_about_median)) {
  if(deviations_about_median[i] < 0){
    absolute_deviations_about_median[i] <- deviations_about_median[i] * (-1)
  }
  else{
    absolute_deviations_about_median[i] <- deviations_about_median[i]
  }
}
```

```
absolute_deviations_about_median

##  [1] 20 22 31  4  1 12  6 19 11 38 15  3 40 47 24 30 34 13 21  1 44 37 32 17 16
## [26] 53 33 22  2 39 43 12 10  3 51  4 15 41 21 13
```

```
my_median_function(absolute_deviations_about_median)

## [1] 20.5
```

Skewness

We compute $\frac{m_3}{m_2^{\frac{3}{2}}}$ where m_2 and m_3 are 2nd and 3rd order central moments respectively.

```
my_skewness_function <- function(x){
  return(my_sample_central_moments_function(x, 3) / my_sample_central_moments_function(x, 2)^(3/2))
}
```

```
my_skewness_function(my_sample)

## [1] 0.1109031
```

Kurtosis

We compute $\frac{m_4}{m_2^2} - 3$ where m_4 is the 4th order central moment.

```
my_kurtosis_function <- function(x){
  return(my_sample_central_moments_function(x, 4) / my_sample_central_moments_function(x, 2)^2 - 3)
}
```

```
my_kurtosis_function(my_sample)

## [1] -1.03632
```