

Numerical Descriptive Measures

Manideep Bangaru

21/11/2020

To clean the environment variables

```
rm(list=ls(all=T))
```

Defining a Vector

```
vec <- c(20,34,56,23,45,22,60,23,56,78,23,45)
```

Mean of the vector

$$\frac{\sum_{i=1}^n x_i}{n}$$

```
mean(vec)
```

```
## [1] 40.41667
```

Median of the vector

$$\frac{(n+1)}{2} \text{ ranked value}$$

```
median(vec)
```

```
## [1] 39.5
```

- For symmetrical distributed data mean, median and mode are almost equal in value
- For asymmetrical distributed data, following relationship holds good approximately
- Mode = 3 * Median - 2 * mean (or)
- Mean - Mode = 3 * (Mean - Median)
- above relation is called as empirical relation. Using this if two measures are known, it is easy to find out the third measure

Harmonic Mean of a vector

$$\frac{n}{\sum_{i=1}^n \frac{1}{x_i}}$$

```
hm_ <- function(vvector) {  
  return(length(vec)/sum(1/vec))  
}
```

```
hm_(vec)
```

```
## [1] 32.88149
```

Geometric Mean of a vector

$$\bar{X}_g = (X_1 * X_2 * X_3 * \dots * X_n)^{\frac{1}{n}}$$

```
gm_ <- function(vvector) {  
  return(prod(vvector)^(1/length(vvector)))  
}
```

```
gm_(vec)
```

```
## [1] 36.39962
```

Variation and shape

Range of a vector

$$\text{Range} = (x_{\max} - x_{\min})$$

```
range_ <- function(vcetor) {  
  return(max(vcetor)-min(vcetor))  
}
```

```
range_(vec)
```

```
## [1] 58
```

Variance of a vector

$$\sigma^2 = \frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n}$$

```
variance_ <- function(vcetor){  
  var_ <- 0  
  for (ele in 1:length(vcetor)){  
    var_ = var_ + (vcetor[ele] - mean(vcetor))^2  
  }  
  return (var_/length(vcetor)-1)  
}
```

```
variance_(vec)
```

```
## [1] 334.9097
```

Standard deviation of a vector

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n}}$$

```
stddev_ <- function(vcetor){  
  return (variance_(vcetor) ^ 0.5)  
}
```

```
stddev_(vec)
```

```
## [1] 18.30054
```

Coefficient of Variation

It measures the scatter in the data with respect to the mean

$$\left(\frac{\sigma}{\mu}\right) * 100$$

```
CoeffVar_ <- function(vcetor){  
  return (stddev_(vcetor)/mean(vcetor)) *100  
}
```

```
CoeffVar_(vec)
```

```
## [1] 0.4527968
```

Z – Score

$$Z = \left(\frac{x - \mu}{\sigma}\right)$$

- a Z score of 0 indicates that the value is same as the mean
- Helps in identifying outliers, less than -3 and greater than +3 are considered to be outliers

```
zscore <- function(vvector){
  temp_ <- c()
  for (ele in 1:length(vvector)){
    temp_ <- append(temp_, (vvector[ele]-mean(vvector))/stddev_(vvector))
  }
  return (temp_)
}
```

```
zscore(vec)
```

```
## [1] -1.1156320 -0.3506272 0.8515232 -0.9517024 0.2504480 -1.0063456
## [7] 1.0700960 -0.9517024 0.8515232 2.0536736 -0.9517024 0.2504480
```

Skewness

- Mean < Median → left skewed or negative skew
- Mean = Median → symmetrical distribution (zero skewness)
- Mean > Median → right skewed or positive skew

Kurtosis

- A distribution that has a sharper-rising center peak than the peak of a normal distribution has positive kurtosis, a kurtosis value that is greater than zero, and is called **leptokurtic**
- A distribution that has a slower-rising (flatter) center peak than the peak of a normal distribution has negative kurtosis, a kurtosis value that is less than zero, and is called **platykurtic**
- A **leptokurtic distribution** has a **higher concentration** of values near the mean of the distribution compared to a normal distribution, while a **platykurtic distribution** has a **lower concentration** compared to a normal distribution

Quartiles

Quartiles split the data into 4 equal parts

$$Q1 = \frac{n+1}{4} \quad Q2 = 2\left(\frac{n+1}{4}\right)$$

$$Q3 = 3\left(\frac{n+1}{4}\right) \quad Q4 = 4\left(\frac{n+1}{4}\right)$$

Percentile

Percentile divides the data into 100 equal parts

The Interquartile Range (IQR)

The interquartile range is the difference between third quartile (**Q3**) and first quartile (**Q1**)

$$IQR = Q3 - Q1$$

The Empirical rule

The Empirical rule states that :

In a Normal distribution,

- Approximately, 68% of the values are within $\pm 1\sigma$
- Approximately, 95% of the values are within $\pm 2\sigma$
- Approximately, 99.7% of the values are within $\pm 3\sigma$

Chebyshev's theorem

For heavily skewed datasets that do not appear to be normally distributed, you should use chebyshev's theorem:

Regardless of the shape, the percentage of values that are found within distances of k standard deviations from the mean must be at least

$$\left(1 - \frac{1}{k^2}\right) * 100$$

The Covariance and the Coefficient of Correlation

The Covariance

It measures the strength of a linear relationship between two numerical variables

$$\text{Sample, } \text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{n-1}$$

$$\text{Population, } \text{cov}(X, Y) = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{n}$$

```
vec1 <- c(23,45,34,23,34,56,78,65,45,34)
vec2 <- c(65,54,34,45,23,45,67,88,96,33)
```

```
cov_ <- function(vcetor_x,vcetor_y,sample){
  numerator <- 0
  for (i in 1:length(vcetor_x)){
    numerator <- numerator + (vcetor_x[i]-mean(vcetor_x))*(vcetor_y[i]-mean(vcetor_y))
  }
  if (sample==T){
    return (numerator/(length(vcetor_x)-1))
  }else if (sample==F){
    return (numerator/length(vcetor_x))
  }else{
    return ('Check your parameters !!!')
  }
}
```

```
cov_(vec1,vec2,T) # sample Variance
```

```
## [1] 196.7778
```

```
cov_(vec1,vec2,F) # Population Variance
```

```
## [1] 177.1
```

The Coefficient of Correlation

It measures the relative strength of a linear relationship between two numerical variables

- Values ranges between -1 to +1
- -1, perfect negative correlation
- +1, perfect positive correlation

$$\text{sample correlation, } \mathbf{r} = \frac{\text{Cov}(X,Y)}{S_X S_Y}$$

```
vec1 <- c(23,45,34,23,34,56,78,65,45,34)
vec2 <- c(65,54,34,45,23,45,67,88,96,33)
```

```
corr_ <- function(vcetor_x,vcetor_y){
  return(cov_(vcetor_x,vcetor_y,F)/(stddev_(vcetor_x)*stddev_(vcetor_y)))
}
```

```
corr_(vec1,vec2)
```

```
## [1] 0.4569768
```