



**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO  
NACIONAL DE MÉXICO



**TECNOLÓGICO NACIONAL DE MÉXICO INSTITUTO  
TECNOLÓGICO DE TIJUANA**

**SUBDIRECCIÓN ACADÉMICA**

**DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN**

**SEMESTRE FEBRERO-JUNIO 2022**

**CARRERA**

**Ingeniería en informática**

**MATERIA**

**Datos masivos**

**TÍTULO**

**Práctica#4**

**Integrantes:**

**Munguía silva Edgar Geovanny #17212344**

**Alicia Guadalupe Pérez López#18210514**

**NOMBRE DEL MAESTRO**

**Jose Christian Romero Hernandez**

**Tijuana Baja California 01 de Mayo del 2022**

## Input.

```
package org.apache.spark.examples.mllib

import org.apache.spark.{SparkConf, SparkContext}
// $example on$
import org.apache.spark.mllib.linalg._
import org.apache.spark.mllib.stat.Statistics
import org.apache.spark.rdd.RDD
object CorrelationsExample {

def main(){

    val conf = new SparkConf().setAppName("CorrelationsExample")
    val sc = new SparkContext(conf)

    // $example on$
    val seriesX: RDD[Double] = sc.parallelize(Array(1, 2, 3, 3, 5)) //
a series
    // must have the same number of partitions and cardinality as
seriesX
    val seriesY: RDD[Double] = sc.parallelize(Array(11, 22, 33, 33,
555))

    // compute the correlation using Pearson's method. Enter "spearman"
for Spearman's method. If a
    // method is not specified, Pearson's method will be used by
default.
    val correlation: Double = Statistics.corr(seriesX, seriesY,
"pearson")
    println(s"Correlation is: $correlation")

    val data: RDD[Vector] = sc.parallelize(
Seq(
        Vectors.dense(1.0, 10.0, 100.0),
        Vectors.dense(2.0, 20.0, 200.0),
        Vectors.dense(5.0, 33.0, 366.0))
    ) // note that each Vector is a row and not a column

    // calculate the correlation matrix using Pearson's method. Use
"spearman" for Spearman's method
    // If a method is not specified, Pearson's method will be used by
default.
```

```
val correlMatrix: Matrix = Statistics.corr(data, "pearson")
println(correlMatrix.toString)
// $example off$

sc.stop()
}
```

### Output.

```
Correlation is: 0.8500286768773001
1.0          0.9788834658894731  0.9903895695275673
0.9788834658894731  1.0          0.9977483233986101
0.9903895695275673  0.9977483233986101  1.0
```