# TECNOLÓGICO NACIONAL DE MEXICO

# INSTITUTO TECNOLOGICO DE TIJUANA

## SUBDIRECCIÓN ACADÉMICA

## DEPARTAMENTO DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

## SEMESTRE FEBRERO-JUNIO 2022

## MATERIA:

Datos masivos.

## UNIDAD 2

**Practica 7**

## DOCENTE:

JOSE CHRISTIAN ROMERO HERNANDEZ

## INTEGRANTES:

López Higuera Saúl Alfredo #18210493

Ramos Rivera Manuel Isaí #17212931

Tijuana BC 20 de mayo del 2022

//Importar las librerías necesarias

```
import org.apache.spark.ml.classification.NaiveBayes
import org.apache.spark.ml.evaluation.MulticlassClassificationEvaluator
import org.apache.spark.sql.SparkSession
```

//Cargar los datos especificando la ruta del archivo

```
val data = spark.read.format("libsvm").load("C:/spark/spark-2.4.8-bin-
hadoop2.7/data/mllib/sample_libsvm_data.txt")

println ("Numero de lineas en el archivo de datos:" + data.count ())
```

//Mostrar las primeras 20 líneas por defecto

```
data.show()
```

```
scala> data.show()
+-----+--------------------+
|label|            features|
+-----+--------------------+
|  0.0|(692,[127,128,129...|
|  1.0|(692,[158,159,160...|
|  1.0|(692,[124,125,126...|
|  1.0|(692,[152,153,154...|
|  1.0|(692,[151,152,153...|
|  0.0|(692,[129,130,131...|
|  1.0|(692,[158,159,160...|
|  1.0|(692,[99,100,101,...|
|  0.0|(692,[154,155,156...|
|  0.0|(692,[127,128,129...|
|  1.0|(692,[154,155,156...|
|  0.0|(692,[153,154,155...|
|  0.0|(692,[151,152,153...|
|  1.0|(692,[129,130,131...|
|  0.0|(692,[154,155,156...|
|  1.0|(692,[150,151,152...|
|  0.0|(692,[124,125,126...|
|  0.0|(692,[152,153,154...|
|  1.0|(692,[97,98,99,12...|
|  1.0|(692,[124,125,126...|
+-----+--------------------+
only showing top 20 rows
```

//Divida aleatoriamente el conjunto de datos en conjunto de entrenamiento y conjunto de prueba de acuerdo con los pesos proporcionados. También puede especificar una seed

```
val Array (trainingData, testData) = data.randomSplit (Array (0.7, 0.3), 100L)
```

// El resultado es el tipo de la matriz, y la matriz almacena los datos de tipo DataSet
//Incorporar al conjunto de entrenamiento (operación de ajuste) para entrenar un modelo bayesiano

```
val naiveBayesModel = new NaiveBayes().fit(trainingData)
```

//El modelo llama a transform() para hacer predicciones y generar un nuevo DataFrame.

```
val predictions = naiveBayesModel.transform(testData)
```

//Salida de datos de resultados de predicción

```
predictions.show()
```

```
scala> predictions.show()
22/05/18 18:02:12 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeSystemBLAS
22/05/18 18:02:12 WARN BLAS: Failed to load implementation from: com.github.fommil.netlib.NativeRefBLAS
+-----+--------------------+--------------------+----------+----------+
|label|            features|       rawPrediction|probability|prediction|
+-----+--------------------+--------------------+----------+----------+
|  0.0|(692,[122,123,124...|[-190595.07825499...|  [1.0,0.0]|       0.0|
|  0.0|(692,[123,124,125...|[-246607.82713076...|  [1.0,0.0]|       0.0|
|  0.0|(692,[123,124,125...|[-199533.44171742...|  [1.0,0.0]|       0.0|
|  0.0|(692,[124,125,126...|[-275837.93657958...|  [1.0,0.0]|       0.0|
|  0.0|(692,[125,126,127...|[-258745.16054128...|  [1.0,0.0]|       0.0|
|  0.0|(692,[126,127,128...|[-279335.13066876...|  [1.0,0.0]|       0.0|
|  0.0|(692,[126,127,128...|[-136866.59032154...|  [1.0,0.0]|       0.0|
|  0.0|(692,[126,127,128...|[-208662.62448642...|  [1.0,0.0]|       0.0|
|  0.0|(692,[127,128,129...|[-211024.47289349...|  [1.0,0.0]|       0.0|
|  0.0|(692,[127,128,129...|[-182940.32083349...|  [1.0,0.0]|       0.0|
|  0.0|(692,[152,153,154...|[-93356.564312516...|  [1.0,0.0]|       0.0|
|  0.0|(692,[153,154,155...|[-260165.09822408...|  [1.0,0.0]|       0.0|
|  0.0|(692,[153,154,155...|[-207398.84940196...|  [1.0,0.0]|       0.0|
|  0.0|(692,[155,156,180...|[-229364.82012475...|  [1.0,0.0]|       0.0|
|  0.0|(692,[234,235,237...|[-95021.471295301...|  [1.0,0.0]|       0.0|
|  1.0|(692,[100,101,102...|[-143311.69653049...|  [0.0,1.0]|       1.0|
|  1.0|(692,[123,124,125...|[-99743.193683546...|  [0.0,1.0]|       1.0|
|  1.0|(692,[124,125,126...|[-127676.82708555...|  [0.0,1.0]|       1.0|
|  1.0|(692,[125,126,127...|[-102406.94970386...|  [0.0,1.0]|       1.0|
|  1.0|(692,[125,126,153...|[-81780.123925676...|  [0.0,1.0]|       1.0|
+-----+--------------------+--------------------+----------+----------+
only showing top 20 rows
```

//Evaluación de la precisión del modelo

```
val evaluator = new
MulticlassClassificationEvaluator().setLabelCol("label").setPredictionCol("p
rediction").setMetricName("accuracy")
```

// Precisión

```
val precision = evaluator.evaluate (predictions)
```

```
scala> val precision = evaluator.evaluate (predictions)
precision: Double = 1.0
```

//Imprimir la tasa de error

```
println ("tasa de error =" + (1-precision))
```

```
scala> println ("tasa de error =" + (1-precision)
tasa de error =0.0

scala>
```