**Code and screenshots**

Code :

import org.apache.spark.sql.functions.\_

import org.apache.spark.ml.feature.{VectorAssembler}

import org.apache.spark.ml.Pipeline

import org.apache.spark.ml.regression.{LinearRegression}

import org.apache.spark.ml.tuning.{CrossValidator, CrossValidatorModel, ParamGridBuilder}

import org.apache.spark.ml.evaluation.{RegressionEvaluator}

import org.apache.spark.ml.param.ParamMap

import org.apache.spark.sql.types.{DoubleType}

val data = spark.read.format("csv").option("header", "true").load("hdfs://10.128.0.5/Bigdata/happiness.csv")

val rank\_score = data.select(col("Happiness Rank").cast(DoubleType), col("Happiness Score").cast(DoubleType))

val Array(trainingData, testData) = rank\_score.randomSplit(Array(0.8, 0.2), 1111)

val assembler = new VectorAssembler()

.setInputCols(Array("Happiness Score"))

.setOutputCol("assembled-features")

val lr = new LinearRegression().setFeaturesCol("assembled-features").setLabelCol("Happiness Rank")

val pipeline = new Pipeline().setStages(Array(assembler, lr))

val evaluator = new RegressionEvaluator().setLabelCol("Happiness Rank").setPredictionCol("prediction").setMetricName("r2")

val cross\_validator = new CrossValidator().setEstimator(pipeline).setEvaluator(evaluator).setEstimatorParamMaps(new ParamGridBuilder().build).setNumFolds(3)

val cvModel = cross\_validator.fit(trainingData)

val predictions = cvModel.transform(testData)

predictions

.select(col("Happiness Rank"), col("Happiness Score"), col("prediction"))

.write

.format("csv")

.save("hdfs://10.128.0.5/Bigdata/happiness/output/")

val r2 = evaluator.evaluate(predictions)

println("r-squared on test data =" + r2)

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedGraphical user interface, text

Description automatically generatedGraphical user interface, text

Description automatically generated