spark-ml功能拾遗

# 管道函数和模型参数CV调试

**import** org.apache.spark.ml.tuning.{CrossValidator, CrossValidatorModel, ParamGridBuilder}  
**import** org.apache.spark.ml.param.Param  
**import** org.apache.spark.ml.regression.LinearRegressionModel  
**import** org.apache.spark.sql.DataFrame  
**import** org.apache.spark.sql.functions.\_  
**import** org.apache.spark.ml.regression.LinearRegression  
**import** org.apache.spark.ml.evaluation.RegressionEvaluator

**val** lrg = **new** LinearRegression()  
 .setMaxIter(maxIterations)  
 .setFeaturesCol("features")  
 .setLabelCol("label")  
 .setPredictionCol("prediction")  
  
**val** paramGrid = **new** ParamGridBuilder()  
 .addGrid(lrg.*standardization*, *Array*(**true**, **false**))  
 .addGrid(lrg.*regParam*, *Array*(0.1, 0.2, 0.3, 0.4, 0.5))  
 .addGrid(lrg.*elasticNetParam*, *Array*(0.0, 0.5, 1.0))  
 .build()  
  
**val** cv = **new** CrossValidator()  
 .setEstimator(lrg)  
 .setEvaluator(**new** RegressionEvaluator)  
 .setEstimatorParamMaps(paramGrid)  
 .setNumFolds(10)  
  
**val** cvModel: CrossValidatorModel = cv.fit(enhanceDF)  
*outputrdd*.put(rddTableName + "\_cvModel", cvModel)  
  
*println*("模型系数")  
featuresCols.zip(  
 cvModel.bestModel.asInstanceOf[LinearRegressionModel].coefficients.toArray  
).foreach {  
 **case** (feature, coefficient) =>  
 *println*(s"**$**feature:**$**coefficient")  
}  
  
*/\*\* 3）打印最优模型的性质并查看训练集的效果 \*/  
println*("最优模型的参数:")  
cvModel.bestModel.*params*.foreach {  
 param: Param[\_] =>  
 *println*(param.name)  
}  
*println*("最优模型的评价指标:", cvModel.avgMetrics.mkString(", "))  
  
**val** trainAndDevelopDF = cvModel  
 .transform(enhanceDF)  
 .withColumn("residual", *col*("label") - *col*("prediction"))  
**val** train\_loss = *mse*(trainAndDevelopDF, "label", "prediction")  
*println*("训练集和开发集的loss", train\_loss)