

CSDN新首页上线啦，邀请你来立即体验！(http://blog.csdn.net/)

CSDN

博客 (//blog.csdn.net?ref=toolbar) 学院 (//edu.csdn.net?ref=toolbar)

下载 (//download.csdn.net?ref=toolbar) GitChat (//gitbook.cn/?ref=csdn)

更多 ▾

搜索

登录 (https://passport.csdn.net/mobile/login?ref=toolbar) 注册 (https://passport.csdn.net/account/mobile/register?ref=toolbar&action=mobileRegister)

## Spark加载PMML进行预测

原创

2016年11月25日 22:28:05

5313

### 软件版本：

CDH:5.8.0 , CDH-hadoop :2.6.0 ; CDH-spark :1.6.0

### 目标：

使用Spark 加载PMML文件到模型，并使用Spark平台进行预测（这里测试使用的是Spark on YARN的方式）。  
具体小目标：  
1. 参考https://github.com/jpmml/jpmml-spark 实现，能运行简单例子；  
2. 直接读取HDFS上面的输入数据文件，使用PMML生成的模型进行预测；  
（第1点和第2点的不一样的地方体现在输入数据的构造上，可以参看下面的代码）

### 具体步骤：

1. 准备原始数据，原始数据包括PMML文件，以及测试数据；分别如下：

```
[html]
1. <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2. <PMML version="4.2" xmlns="http://www.dmg.org/PMML-4_2">
3.   <Header description="linear SVM">
4.     <Application name="Apache Spark MLlib"/>
5.     <Timestamp>2016-11-16T22:17:47</Timestamp>
6.   </Header>
7.   <DataDictionary numberOfFields="4">
8.     <DataField name="field_0" optype="continuous" dataType="double"/>
9.     <DataField name="field_1" optype="continuous" dataType="double"/>
10.    <DataField name="field_2" optype="continuous" dataType="double"/>
11.    <DataField name="target" optype="categorical" dataType="string"/>
12.  </DataDictionary>
13.  <RegressionModel modelName="linear SVM" functionName="classification" normalizationMethod="none">
14.    <MiningSchema>
15.      <MiningField name="field_0" usageType="active"/>
16.      <MiningField name="field_1" usageType="active"/>
17.      <MiningField name="field_2" usageType="active"/>
18.      <MiningField name="target" usageType="target"/>
19.    </MiningSchema>
20.    <RegressionTable intercept="0.0" targetCategory="1">
21.      <NumericPredictor name="field_0" coefficient="-0.36682158807862086"/>
22.      <NumericPredictor name="field_1" coefficient="3.8787681305811765"/>
23.      <NumericPredictor name="field_2" coefficient="-1.6134308474471166"/>
24.    </RegressionTable>
25.    <RegressionTable intercept="0.0" targetCategory="0"/>
26.  </RegressionModel>
27. </PMML>
```

以上pmml文件是由一个svm模型构建的，其输入有三个字段，有一个目标输出，代表类别；  
输入测试数据，如下：

**fansy1990** (http://blog....)

+ 关注

(http://blog.csdn.net/fansy1990)

码云

原创

粉丝

喜欢

未开通

265

1324

0

(https://gi  
utm\_sourc

- 他的最新文章

更多文章 (http://blog.csdn.net/fansy1990)
- SparkSQL read Elasticsearch ClassNotFoundException (http://blog.csdn.net/fansy1990/article/details/78652768)
- Java Web提交任务到Spark Standalone集群并监控 (http://blog.csdn.net/fansy1990/article/details/78551986)
- TensorFlowOnSpark stuck (http://blog.csdn.net/fansy1990/article/details/78402457)
- Centos6安装TensorFlow及TensorFlowOnSpark (http://blog.csdn.net/fansy1990/article/details/78370648)
- Spark应用HanLP对中文语料进行文本挖掘--聚类 (http://blog.csdn.net/fansy1990/article/details/77577061)

- 相关推荐
- PMML(一):初探 (http://blog.csdn.net/li taoshoujiao/article/details/8536268)
- 走出“搜索引擎营销”三个误区 (http://blog.csdn.net/ywhxu09/article/details/7004162)
- Weka生成和加载PMML文件 (http://blog.csdn.net/hanphy/article/details/51900774)
- Spark读写Hive添加PMML支持 (http://blog.csdn.net/fansy1990/article/details/53444781)

**[plain]**

```

1. field_0,field_1,field_2
2. 98,97,96
3. 1,2,7

```

这个数据由列名和数据组成，这里需要注意，列名需要和pmml里面的列名对应；

2. 把<https://github.com/jpmml/jpmml-spark>工程下载到本地，并添加如下代码：

**[java]**

```

1. package org.jpmmml.spark;
2.
3. import org.apache.hadoop.conf.Configuration;
4. import org.apache.hadoop.fs.FileSystem;
5. import org.apache.hadoop.fs.Path;
6. import org.apache.spark.SparkConf;
7. import org.apache.spark.api.java.JavaSparkContext;
8. import org.apache.spark.ml.Transformer;
9. import org.apache.spark.sql.*;
10. import org.jpmmml.evaluator.Evaluator;
11.
12. public class SVMEvaluationSparkExample {
13.
14.     static
15.     public void main(String... args) throws Exception {
16.
17.         if(args.length != 3){
18.             System.err.println("Usage: java " + SVMEvaluationSparkExample.class.getName() + " <PMML file> <input data> <output data>");
19.             System.exit(-1);
20.         }
21.         /**
22.          * 根据pmml文件，构建模型
23.          */
24.         FileSystem fs = FileSystem.get(new Configuration());
25.         Evaluator evaluator = EvaluatorUtil.createEvaluator(fs.open(new Path(args[0])));
26.
27.         TransformerBuilder modelBuilder = new TransformerBuilder(evaluator)
28.             .withTargetCols()
29.             .withOutputCols()
30.             .exploded(true);
31.
32.         Transformer transformer = modelBuilder.build();
33.
34.         /**
35.          * 利用DataFrameReader从原始数据中构造 DataFrame对象
36.          * 需要原始数据包含列名
37.          */
38.         SparkConf conf = new SparkConf();
39.         try(JavaSparkContext sparkContext = new JavaSparkContext(conf)){
40.
41.             SQLContext sqlContext = new SQLContext(sparkContext);
42.
43.             DataFrameReader reader = sqlContext.read()
44.                 .format("com.databricks.spark.csv")
45.                 .option("header", "true")
46.                 .option("inferSchema", "true");
47.             DataFrame dataframe = reader.load(args[1]); // 输入数据需要包含列名
48.
49.             /**
50.              * 使用模型进行预测
51.              */
52.             dataframe = transformer.transform(dataframe);
53.
54.             /**
55.              * 写入数据
56.              */
57.             DataFrameWriter writer = dataframe.write()
58.                 .format("com.databricks.spark.csv")
59.                 .option("header", "true");
60.
61.             writer.save(args[2]);
62.         }
63.     }
64. }
65. }

```

这个代码主要实现的是小目标1，即参考jpmml-spark工程给的示例，编写代码；代码有四个部分，第一部分读取HDFS上面的PMML文件，然后构建模型；第二部分使用DataFrameReader根据输入数据构建

**住人集装箱****博主专栏**

mahout算法源码分析  
(<http://blog.csdn.net/column>)

8 35614

(<http://blog.csdn.net/column/detail>)  
JavaWeb invoke Spark

(<http://blog.csdn.net/column>)  
2 3575

(<http://blog.csdn.net/column/detail>)

**水连接器研发生产**

¥7.50/只

已售0件

**他的热门文章**

HBase表管理系统 (<http://blog.csdn.net/fansy1990/article/details/51494095>)

19731

Eclipse调用hadoop2运行MR程序 (<http://blog.csdn.net/fansy1990/article/details/22896249>)

18810

基于HBase的冠字号查询系统2--实现部分 (<http://blog.csdn.net/fansy1990/article/details/51583401>)

14439

Hadoop k-means 算法实现 (<http://blog.csdn.net/fansy1990/article/details/8028546>)

13919

基于HBase的冠字号查询系统1--理论部分 (<http://blog.csdn.net/fansy1990/article/details/51583080>)

13722

DataFrame数据结构；第三部分，使用模型对构造的DataFrame数据进行预测；第四部分，把预测的结果写入HDFS。

注意里面在构造数据的时候.option("header","true")是一定要加的，原因如下：1) 原始数据中确实有列名；2) 如果这里不加，那么将读取不到列名的相关信息，将不能和模型中的列名对应；(当然，下面有其他方法处理这种情况)。

3. 上传测试数据以及pmml文件到HDFS，进行测试，代码如下：

[plain]

```
1. spark-submit --master yarn --class org.jpmmml.spark.SVMEvaluationSparkExample /opt/tmp/example-1.0-SNAPSHOT.jar hdfs://quickstart.cloudera:8020/tmp/svm/part-00000 sample_test_data.txt sample_out00
```

其中，example-1.0-SNAPSHOT.jar 是编译后的jar包；/tmp/svm/part-00000时svm模型的pmml文件；sample\_test\_data.txt 是测试数据；sample\_out00是输出目录；

查看结果：

**File: /user/root/sample\_out00/part-00001**

Goto :

[Go back to dir listing](#) /blog.csdn.net/  
[Advanced view/download options](#)

```
field_0,field_1,field_2,target
98,97,96,1
1,2,7,0
```

根据输出的结果，也可以看出预测结果是对的。

4. 如何实现小目标2呢？

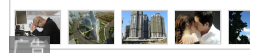
编写代码：

[java]

```
1.  /*
2.  * Copyright (c) 2015 Villu Ruusmann
3.  *
4.  * This file is part of JPMMML-Spark
5.  *
6.  * JPMMML-Spark is free software: you can redistribute it and/or modify
7.  * it under the terms of the GNU Affero General Public License as published by
8.  * the Free Software Foundation, either version 3 of the License, or
9.  * (at your option) any later version.
10. *
11. * JPMMML-Spark is distributed in the hope that it will be useful,
12. * but WITHOUT ANY WARRANTY; without even the implied warranty of
13. * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
14. * GNU Affero General Public License for more details.
15. *
16. * You should have received a copy of the GNU Affero General Public License
17. * along with JPMMML-Spark. If not, see <http://www.gnu.org/licenses/>.
18. */
19. package org.jpmmml.spark;
20.
21. import org.apache.hadoop.conf.Configuration;
22. import org.apache.hadoop.fs.FileSystem;
23. import org.apache.hadoop.fs.Path;
24. import org.apache.spark.SparkConf;
25. import org.apache.spark.api.java.JavaRDD;
26. import org.apache.spark.api.java.JavaSparkContext;
27. import org.apache.spark.api.java.function.Function;
28. import org.apache.spark.ml.Transformer;
29. import org.apache.spark.sql.*;
30. import org.apache.spark.sql.types.DataTypes;
31. import org.apache.spark.sql.types.StructField;
32. import org.apache.spark.sql.types.StructType;
33. import org.dmg.pmml.FieldName;
34. import org.jpmmml.evaluator.Evaluator;
35.
36. import java.util.ArrayList;
37. import java.util.List;
38.
39. //import org.jpmmml.evaluator.FieldValue;
40.
41. public class EvaluationSparkExample {
42.
43.     static
```



拓展训练价格



广告

```
44. public void main(String... args) throws Exception {
45.
46.     if(args.length != 3){
47.         System.err.println("Usage: java " + EvaluationSparkExample.class.getName() + " <PMML file>
48.
49.         System.exit(-1);
50.     }
51.
52.     /**
53.      * 构造模型
54.      */
55.     FileSystem fs = FileSystem.get(new Configuration());
56.     Evaluator evaluator = EvaluatorUtil.createEvaluator(fs.open(new Path(args[0])));
57.
58.     TransformerBuilder modelBuilder = new TransformerBuilder(evaluator)
59.         .withTargetCols()
60.         .withOutputCols()
61.         .exploded(true);
62.     Transformer transformer = modelBuilder.build();
63.
64.     /**
65.      * 构造列名,schema
66.      */
67.     List<StructField> fields = new ArrayList<>();
68.     for (FieldName fieldName: evaluator.getActiveFields()) {
69.         fields.add(DataTypes.createStructField(fieldName.getValue(), DataTypes.StringType, true));
70.     }
71.     StructType schema = DataTypes.createStructType(fields);
72.
73.     /**
74.      * 原始数据构造成DataFrame
75.      */
76.     SparkConf conf = new SparkConf();
77.     final String splitter = ",";
78.     try{JavaSparkContext sparkContext = new JavaSparkContext(conf)){
79.         JavaRDD<Row> data = sparkContext.textFile(args[1]).map(new Function<String, Row>() {
80.             @Override
81.             public Row call(String line) throws Exception {
82.                 String[] lineArr = line.split(splitter,-1);
83.                 return RowFactory.create(lineArr);
84.             }
85.         });
86.
87.         SQLContext sqlContext = new SQLContext(sparkContext);
88.         DataFrame dataframe = sqlContext.createDataFrame(data, schema);
89.
90.         /**
91.          * 预测,并生成新的DataFrame
92.          */
93.         dataframe = transformer.transform(dataframe);
94.
95.         /**
96.          * 把评估后的数据写入HDFS,不要写入列名
97.          */
98.         DataFrameWriter writer = dataframe.write()
99.             .format("com.databricks.spark.csv");
100.         writer.save(args[2]);
101.
102.     }
103. }
104. }
```

加入CSDN，享受更精准的内容推荐，与5000万程序员共同成长！

内容举报

返回顶部

登录 注册 X (https://passpc

这个代码和上一个代码的不同之处只是从原始测试数据中构造DataFrame不同，这里使用的PMML模型中的列名信息，代码参考：<http://spark.apache.org/docs/1.6.0/sql-programming-guide.html#interoperating-with-rdds>；同时，这时，原始测试数据就不需要再添加列名信息了。由于在代码中，在输出的时候也把列名信息给去掉了，所以只输出数据。运行后，其结果如下所示：  
**File: /user/root/sample\_out02/part-00000**

Goto :

[Go back to dir listing](#) /blog.csdn.net/  
[Advanced view/download options](#)

98,97,96,1

其调用代码如下所示：

```
[plain]
1. spark-submit --master yarn --class org.jpmmml.spark.EvaluationSparkExample /opt/tmp/example-1.0-SNAPSHOT.jar hdfs://quickstart.cloudera:8020/tmp/svm/part-00000 sample_test_data1.txt sample_out02
```

其中，sample\_test\_data1.txt是没有列名的数据。

分享，成长，快乐

转载请注明blog地址：<http://blog.csdn.net/fansy1990> (<http://blog.csdn.net/fansy1990>)



## 相关文章推荐

### PMML(一):初探 (<http://blog.csdn.net/litaoshoujiao/article/details/8536268>)

1.简介 PMML全称预言模型标记语言（Predictive Model Markup Language），利用XML描述和存储数据挖掘模型，是一个已经被W3C所接受的标准。MML是一种基于XM...

litaoshoujiao (<http://blog.csdn.net/litaoshoujiao>) 2013年01月23日 23:46 7395

### 走出“搜索引擎营销”三个误区 (<http://blog.csdn.net/ywhxu09/article/details/7004162>)

一说起搜索引擎营销(付费竞价排名)，更多从业者会将其与排名、流量、甚至销量挂钩，孰不知，搜索引擎营销也与品牌息息相关。 1、搜索引擎营销三个误区 经过长时间的观察，发现众网站在搜索引擎营销...

ywhxu09 (<http://blog.csdn.net/ywhxu09>) 2011年11月23日 14:32 190



### 惊呆了！微博和阿里背后的数据库有多厉害？

想不到！数据库作为最关键的基础设施，渗透技术领域的方方面面，我阿里和微博的师哥们是这么分享的...

([http://www.baidu.com/cb.php?c=IgF\\_pyfqHmknjTzrjb0IZ0qnK9ujYzP1nsrjD10Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1Y3PAfdmyc1nH0dmHmknH6k0AwY5HDdnHn4rH63PHR0IgF\\_5y9YIZ0IQzq-uZR8mLPbUB48ugfEpZNGXy-jULNzTvRETVNzpyN1gvw-IA7GUatLPjqdIAdxTvqdThP-5yF\\_UvTkn0KzujYk0AFV5H00TZcqN0KdpyfqHRLPjnvnfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnWD4rjR](http://www.baidu.com/cb.php?c=IgF_pyfqHmknjTzrjb0IZ0qnK9ujYzP1nsrjD10Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1Y3PAfdmyc1nH0dmHmknH6k0AwY5HDdnHn4rH63PHR0IgF_5y9YIZ0IQzq-uZR8mLPbUB48ugfEpZNGXy-jULNzTvRETVNzpyN1gvw-IA7GUatLPjqdIAdxTvqdThP-5yF_UvTkn0KzujYk0AFV5H00TZcqN0KdpyfqHRLPjnvnfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnWD4rjR))

### Weka生成和加载PMML文件 (<http://blog.csdn.net/hanphy/article/details/51900774>)

网络上太多示例展示了Weka怎么样调用数据分类算法，但想想我如何针对一个训练好的分类模型进行重用呢。所以必须要“导出”。导出模型，一个标准的方式就是用PMML了。...

hanphy (<http://blog.csdn.net/hanphy>) 2016年07月13日 19:22 1375

### Spark读写Hive添加PMML支持 (<http://blog.csdn.net/fansy1990/article/details/53444...>)

软件版本：CDH：5.8.0；Hadoop：2.6.0；Spark：1.6.0；Hive：1.1.0；JDK：1.7；SDK：2.10.6 (Scala) 目标：在Spark加载PMML文件处理...

fansy1990 (<http://blog.csdn.net/fansy1990>) 2016年12月03日 15:49 1887

## 分类解读Spark下的39个机器学习库 (<http://blog.csdn.net/sparkexpert/article/details/4...>)



转自小象学院的文章 ( <http://xxwenda.com/article/584> ) , 后续准备逐个试验一下。当然有不少已经测试过的。 Apache Spark 本身 1.MLib AM...

 sparkexpert (<http://blog.csdn.net/sparkexpert>) 2015年11月05日 09:01  2664





## XGBoost模型文件转化为PMML (<http://blog.csdn.net/Sinsa110/article/details/522022...>)


运用java包和命令行讲XGBoost模型转化为PMML通用模型文件。 前期准备 下载jpmml-xgboost, <https://github.com/jpmml/jpmml-xgboost/>...

 Sinsa110 (<http://blog.csdn.net/Sinsa110>) 2016年08月13日 22:54  4299


## JPMML解析Random Forest模型并使用其预测分析 (<http://blog.csdn.net/c1481118216/...>)

准备pmml文件, 数据集文件如果没有可以藏考我的博客: R训练Random Forest并转换成PMML导入Jar包maven 的pom.xml文件中添加jpmml的依赖 org.jpml...

 c1481118216 (<http://blog.csdn.net/c1481118216>) 2017年07月05日 20:03  1515





### Delphi7高级应用开发随书源码 (<http://download.csdn.net/download/c...>)

(<http://download.csdn.net/download/c...>) 2003年04月30日 00:00 676KB 



## 机器学习算法线部署方法 (<http://blog.csdn.net/u012294181/article/details/54564391>)

本文由携程技术中心投递, ID: ctriptech。作者: 潘鹏举, 携程酒店研发BI经理, 负责酒店服务相关的业务建模工作, 主要研究方向是用机器学习实现业务流程自动化、系统智能化、效率最优化, 专注于算法实践...

 u012294181 (<http://blog.csdn.net/u012294181>) 2017年01月15日 20:25  2560

## 将python或R生成的模型存为PMML供java调用 (<http://blog.csdn.net/u010035907/artic...>)

查看jpmml的说明文档: <https://github.com/jpmml/jpmml-evaluator> 其它参考资料 1、XGBoost模型文件转化为PMML 2、JPMML Example ...

 u010035907 (<http://blog.csdn.net/u010035907>) 2017年05月27日 10:33  3090



## PMML模型文件在机器学习的实践经验 (<http://blog.csdn.net/hopeztm/article/details/7...>)

算法工程师和业务开发工程师, 所掌握的技能容易在长期的工作中出现比较深的鸿沟, 算法工程师辛辛苦苦调参的成果, 业务工程师可能不清楚如何使用, 如何为线上决策给予支持。本文介绍一种基于PMML的模型上线方法。...

 hopeztm (<http://blog.csdn.net/hopeztm>) 2017年10月23日 18:33  821

## Apache Spark 2.0 : 机器学习模型持久化 (<http://blog.csdn.net/A3301/article/details/5...>)


在即将发布的Apache Spark 2.0中将会提供机器学习模型持久化能力。机器学习模型持久化 (机器学习模型的保存和加载) 使得以下三类机器学习场景变得容易: 数据科学家开发ML模型并移交给工程...

 A3301 (<http://blog.csdn.net/A3301>) 2016年11月19日 11:59  1801

## SPARK模型实例: 两种方法实现随机森林模型 (MLlib和ML) (<http://blog.csdn.net/dahu...>)




SPARK模型实例，基于HiveSQL，实现随机森林模型的训练和预测

 dahunbi (<http://blog.csdn.net/dahunbi>) 2017年06月02日 17:34 1282


## 使用Spark构建聚类模型 (<http://blog.csdn.net/lovebyz/article/details/51290679>)

将使用一个模型（推荐模型）的输出作为另外一个模型（聚类模型）的输入 import org.apache.spark.mllib.clustering.KMeans import org.apache...

 lovebyz (<http://blog.csdn.net/lovebyz>) 2016年05月01日 15:53 2193

## libsvm savemodel and loadmodel (<http://blog.csdn.net/DreamD1987/article/detail...>)

savemodel 和 loadmodel的c代码如下： #include "svm.h" #include "mex.h" #include "svm\_model\_matlab.h" stati...

 DreamD1987 (<http://blog.csdn.net/DreamD1987>) 2014年06月16日 11:15 1380


## 分享Spark MLlib训练的广告点击率预测模型 ([http://blog.csdn.net/LW\\_GHY/article/deta...](http://blog.csdn.net/LW_GHY/article/deta...))

2015年，全球互联网广告营收接近600亿美元，比2014年增长了近20%。多家互联网巨头都依赖于广告营收，如谷歌，百度，Facebook，互联网新贵们也都开始试水广告业，如Snapchat, Pin...

 LW\_GHY ([http://blog.csdn.net/LW\\_GHY](http://blog.csdn.net/LW_GHY)) 2017年01月14日 14:05 2632

## SparkML之回归(一)线性回归 (<http://blog.csdn.net/legotime/article/details/51836008>)

-----目录-----...

 legotime (<http://blog.csdn.net/legotime>) 2016年07月06日 07:36 3946


## Java集成Weka做逻辑回归（Logistic Regression）（续）(<http://blog.csdn.net/hanphy...>)

从网上找样本数据太不好找了，尤其是想看看多分类的那种数据；而且数据量都偏小，不好玩。得，还是自己造数据，当然规则则自己拟。自己造数据，生成arff文件。...

 hanphy (<http://blog.csdn.net/hanphy>) 2016年07月13日 10:14 1050


## Spark之导出PMML文件（Python）(<http://blog.csdn.net/jclian91/article/details/787...>)

本文将介绍如何在Spark中导出PMML文件（Python语言）。

 jclian91 (<http://blog.csdn.net/jclian91>) 2017年12月04日 12:56 48

## 基于spark用线性回归（linear regression）进行数据预测 (<http://blog.csdn.net/wtt56111...>)

ubuntu+spark+scala实现线性回归（linear regression）算法（代码+数据）

 wtt561111 (<http://blog.csdn.net/wtt561111>) 2017年03月08日 13:05 2141