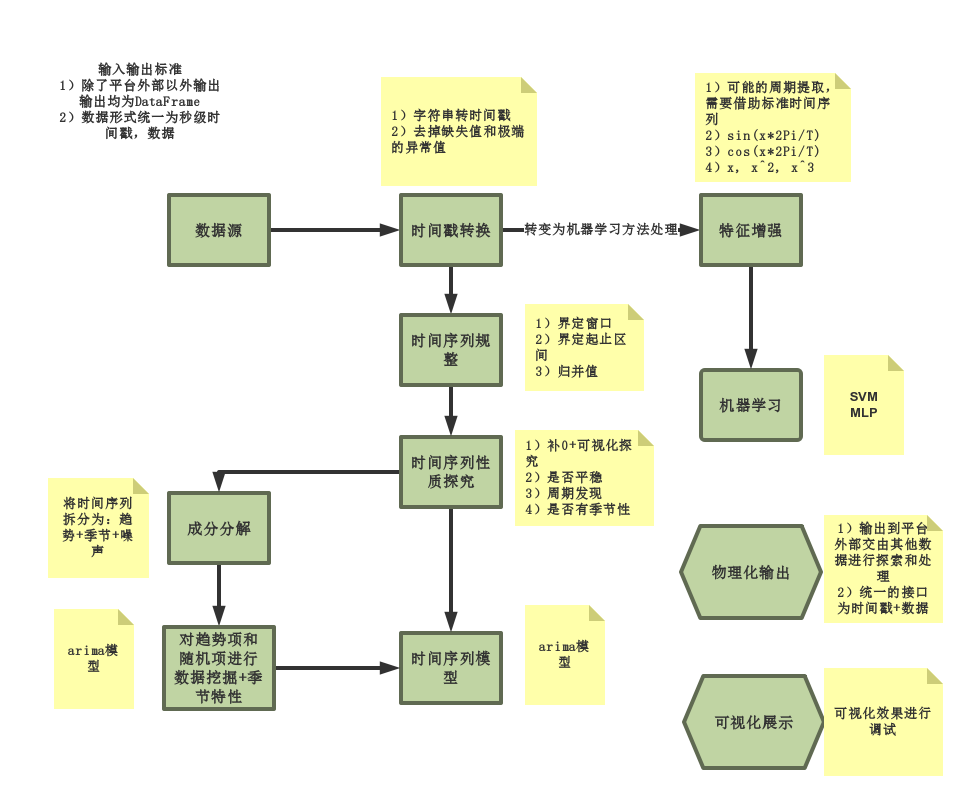
时序数据挖掘

# 时序数据挖掘示意图



# 流程注意事项

## 数据预处理

### 时间字符串转时间戳 + 排序

SQL算子就可以做到，如果是毫秒时间则需要写一个udf

### 时间分箱 + SQL过滤频数过低的箱子

### 过滤极端值

根据分位数，过滤掉前后若干百分比的数据。

### 数量数据预处理

功能简介：将输入数据进行预处理并输出

输入：DataFrame，假定时间列已经被转为了毫秒级的时间戳，且已经过滤了极端值

功能点：

1）log变换

2）diff

3）模型需要有变换和反向变换两种操作

## 基于时间序列的数据挖掘方法

arima

## 模型评价

### 模型评价标准

模型评价要数据拆分，评价标准和实际应用场景有关系。

有两种方式：

1）一种场景是找规律，一次性找到大致的规律，之后规律不变一直用，以这个为标准可以预测今后相当长的时间内的数据，且预测差异和时间无关。

2）另一种场景是利用当前的数据预测一个近期内的数据，数据还会一直来，只是当前不知道，需要预测而已。当下一批数据来了之后就可以继续预测再下一阶段的数据。此时的预测和时间远近有关，越近越准确，越远越不准确。

### 数据拆分

由于数据的顺序对模型有关系，因此不能随机拆分。

应该每次选取若干时间序列，将拆分出来前后两部分，时间靠前的作为训练数据，时间靠后的作为预测数据。

特征增强

1）T可以同时取多个

2）sin和cos一定要同时有

3）也可以加sin平方等

## 模型评价

# 项目落实

现场项目实施的评比标准：部署快、预测准、性能高。

## 基于机器学习的路径做一条快速调试实现的分析流

### 功能简介

目标是做到快，且能达到一定的精准度，先保底。

### 实现流程

实现流程。

1）数据源

2）将时间转为长整型的时间

3）异常值过滤，如果数据没有极端不合理的值的话，这一个可以不要。

4）将数据持久化到hdfs，再从hdfs拉取到本地。

5）本地进行数据增强，并基于R进行模型选择。

6）将基于R的模型选择结果做成spark版本的特征增强

7）基于特征增强的数据利用线性回归模型进行粗调，对“标准化”、“reg”确定每组对应的收敛次数（确保模型收敛）

8）通过cv模型选择进行粗调。

9）通过模型预测新数据。（注意这里预测所用的模型包括两个部分：特征增强 + 线性回归）

### 所需工具及其包

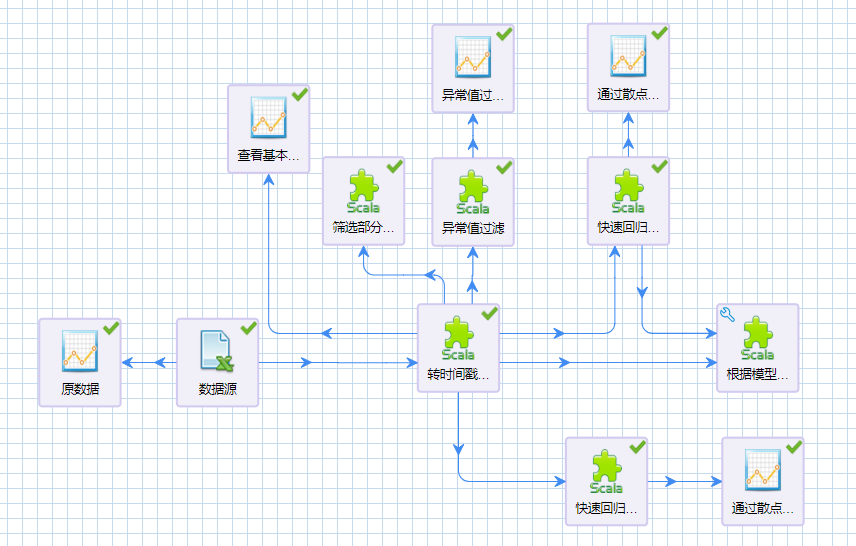
intellij + scala插件 + 工程项目 + spark源码

RGui + RStudio + 写的R脚本

平台部署：无

### 分析流实现和R中的调试过程

1）分析流



2）在R中的调试过程

1.从hdfs拉取数据，并结合列名做成csv形式

2.读取数据

3.特征增强

4.回归并进行初步的特征筛选

5.通过逐步回归进行筛选

6.结合可视化查看初步效果，不行就变一下特征增强过程

7.打印筛选的特征

### 初步效果

训练集拟合效果：



测试集预测效果



## 基于时间序列做一个预测分析流

### 功能简介

目标是作为机器学习搞到瓶颈时候的另一条路，另外也可以结合机器学习，如：利用时间序列预测机器学习未能识别的残差，或者利用机器学习预测时间序列的趋势项，这个暂时不考虑，不过接口都是DataFrame应该好做。

注意此处的模型评价不能再采用随机拆分，而应该是随机取一段数据，将其根据前后拆分为训练集和测试集。

注意该方式很多都是单节点进行的，都需要collect到R，基于R进行处理在推到sparkR中。因此性能会有很大影响。

### 实现流程

1）将数据按一个合适的窗口分箱，选择每个窗口的均值，如果数据比较好，是连在一起的，不分箱也可以。

2）将数据持久化到hdfs，再从hdfs拉取到本地

3）基于R进行基本的探索：几阶平稳，acf和pcf，周期性

4）进行数据挖掘：

第一种方式是：arima模型进行预测并查看预测效果

第二种方式是：arima模型用于预测数据的一个分量，该方法可以是：周期分解后预测分解的趋势项或者机器学习方法后预测机器学习方法的残差项。

5）利用arima模型预测。注意预测步骤需要结合第1步的分箱。

### 所需工具及其包

intellij + scala插件 + 工程项目 + spark源码

RGui + RStudio + 写的R脚本 + tseries + graphic

平台部署：无

### 初步效果

