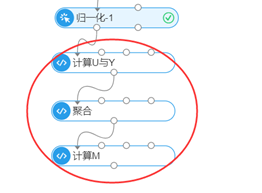
2.3 时间序列分解

这一步新用到的PAI组件有：

“数据预处理”--“归一化”：可以对数据进行归一化处理。

**1.时间序列分解计算**

接下来进行时间序列分解，要求的参数有U,T,C,S,R。其中，U直接取训练集的平均dQ即可，T则需要计算移动平均。从左侧拖出3个“SQL”脚本，按如下方式连接并重命名：



在“计算U与Y”输入如下代码：

select a.monthno

,a.dQ\_adj

,a.U

,(a.dQ\_adj/a.U) as Y

from

(SELECT monthno

,dQ as dQ\_adj

,avg(dQ) over(partition by 1) AS U

from ${t1}

) a

这段代码根据归一化之后的dQ与其均值U计算出了Y。

右键单击“计算U与Y”，选择“执行该节点”。再右键单击“计算U与Y”，左键单击“查看数据”



这里已经计算出了Y，接下来要对Y求移动平均。

在“聚合”输入如下代码：

select \* from

( select monthno,lag(a.y,1) over (partition by 1 order by monthno) as y from ${t1} a

union all

select monthno,lag(b.y,2) over (partition by 1 order by monthno) as y from ${t1} b

union all

select monthno,lag(c.y,3) over (partition by 1 order by monthno) as y from ${t1} c

union all

select monthno,lag(d.y,4) over (partition by 1 order by monthno) as y from ${t1} d

union all

select monthno,lag(e.y,5) over (partition by 1 order by monthno) as y from ${t1} e

union all

select monthno,lag(f.y,6) over (partition by 1 order by monthno) as y from ${t1} f

union all

select monthno,lag(g.y,7) over (partition by 1 order by monthno) as y from ${t1} g

union all

select monthno,lag(h.y,8) over (partition by 1 order by monthno) as y from ${t1} h

union all

select monthno,lag(i.y,9) over (partition by 1 order by monthno) as y from ${t1} i

union all

select monthno,lag(j.y,10) over (partition by 1 order by monthno) as y from ${t1} j

union all

select monthno,lag(k.y,11) over (partition by 1 order by monthno) as y from ${t1} k

union all

select monthno,lag(l.y,12) over (partition by 1 order by monthno) as y from ${t1} l

union all

select monthno,lag(m.y,13) over (partition by 1 order by monthno) as y from ${t1} m

) t1;

在“计算M”输入如下代码：

select monthno,avg(y) as m

from ${t1}

where monthno >=14

group by monthno

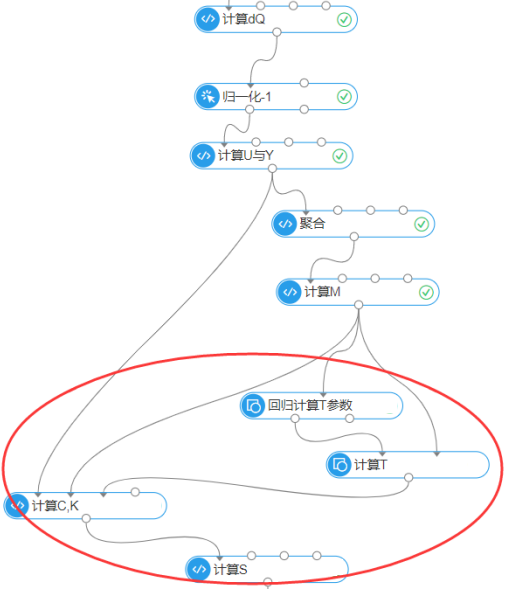
这两段代码实现了PAI机器学习平台中的移动平均。SQL脚本中的lag函数可以取某字段前第几行的数据，通过13次的union再对monthno聚合取y的平均，就可以计算出y的移动平均。每一段被union的select语句都选择了往前推1,2，...，13期的y，最后通过一个select avg(y) group by monthno语句将其聚和，便可以算出移动平均。Monthno>14是因为计算移动平均需要前13个月的y，所以monthno从1到13的结果都是没有意义的。

右键单击“计算U与Y”，左键单击“从此处开始执行”。

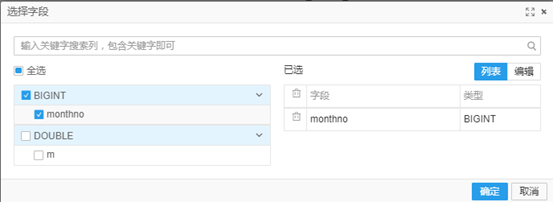
在得到了移动平均后，可以计算该商品的T,C,S,R等指标。

从左侧拖出一个“线性回归”，一个“预测”与2个“SQL脚本”，按如下方式连接并重命名。

|  |  |  |  |
| --- | --- | --- | --- |
| **组件类型** | **重命名** | **输入端口** | **上游端口** |
| 线性回归 | 回归计算T参数 | 1 | 计算M--1 |
| 预测 | 计算T | 1 | 回归计算T参数--1 |
| 2 | 计算M--1 |
| SQL脚本 | 计算C,K | 1 | 计算U与Y--1 |
| 2 | 计算M--1 |
| 3 | 计算T--1 |
| SQL脚本 | 计算S | 1 | 计算C,K--1 |



“回归计算T参数”的“选择特征列”选择‘monthno’：

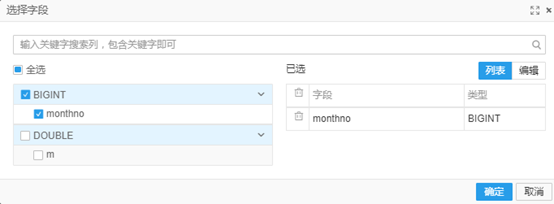


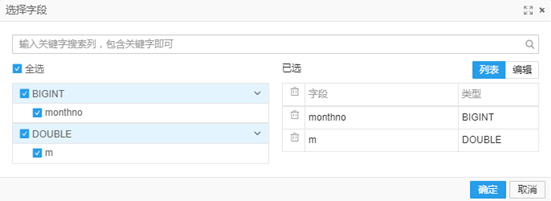
“选择标签列”选择‘m’：



右键单击“回归计算T参数”，左键单击“执行该节点”。

待运行结束，在“计算T”的“特征列”选择‘monthno’，“原样输出列”选择‘monthno’与‘m’两个字段：





在“计算C,K”输入如下代码：

select a.MONTHNO

,c.u

,a.prediction\_result as T

,b.m/a.prediction\_result as C

,c.Y/b.M as K

from

(select \* from ${t3}) a

left outer join

(select monthno

,m

from ${t2}) b

on a.monthno = b.monthno

left outer join

(select monthno

,u

,Y

from ${t1}

where monthno >=14) c

on a.monthno = c.monthno

这段代码对3张表的数据进行了计算与汇总。

鼠标右键单击“计算C,K”，左键单击“查看数据”



可以看出，T是一个下降的序列，也就是说该产品的销量呈现出了一个下降趋势。

表a取了“计算T”中的参数T，表b取了“计算M”中的移动平均M，根据公式可以用M计算出C与K，表c取了“计算U与Y”中的参数U和Y。至此，已经有了参数U,T,C,接下来根据K来计算S。

在“计算S”输入如下代码：

select monthno\_in1y

,avg(K) as S

from

(select monthno

,monthno %13 as monthno\_in1y

,K

from ${t1}) a

group by monthno\_in1y

这段代码根据monthno，计算了每一个monthno是每年内的第几个月，将其命名为monthno\_in1y，然后对monthno\_in1y聚合求K的平均，便可以得到参数S。

右键单击“计算T,”左键单击“从此处开始执行”。

待其运行结束，右键单击“计算S”，左键单击“查看数据”



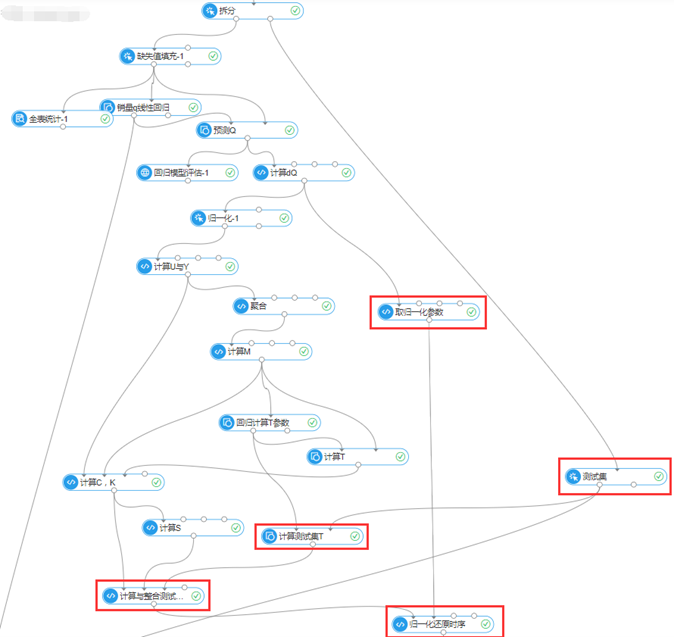
可以看出，这张表里只有monthno\_in1y所对应的S，后面在计算测试集的S时，需要将相应的monthno计算出其对应的monthno\_in1y再回到这张表中取出S。

产品的季节波动S的最大值是1.7  最小值是0.6，说明产品的季节性波动幅度还是比较大的。结合定价策略，可以在季节波动的波峰提升利润。

**2.预测时间序列销量**

从左侧拖出一个“拆分”，一个“预测”和3个“SQL脚本”，按如下方式连接并重命名：

|  |  |  |  |
| --- | --- | --- | --- |
| **组件类型** | **重命名** | **输入端口** | **上游端口** |
| SQL脚本 | 取归一化参数 | 1 | 计算dQ--1 |
| 拆分 | 测试集 | 1 | 拆分--2 |
| 预测 | 计算测试集T | 1 | 回归计算T参数--1 |
| 2 | 测试集--1 |
| SQL脚本 | 计算与整合测试集T,C,K,S,R | 1 | 计算C,K--1 |
| 2 | 计算S--1 |
| 3 | 计算测试集T--1 |
| SQL脚本 | 归一化还原时序 | 1 | 计算与整合测试集T,C,K,S,R--1 |
| 2 | 取归一化参数--1 |



在“取归一化参数”输入如下代码：

select monthno

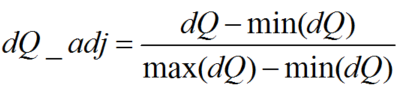
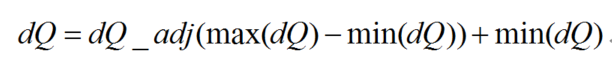
,dq

,max(dq) over(partition by 1) AS max

,min(dq) over(partition by 1) AS min

, 1 as justforjoin

from ${t1}

这段代码计算了dQ的最大值与最小值。根据公式，要对该序列进行还原，需要的逆变换公式为： 

justforjoin字段只是为了方便后面跟测试集的预测数据进行聚合而设置的，没有其他含义。

右键单击“取归一化参数”，左键单击“执行该节点”

在“测试集”的“参数设置”选择“按阀值拆分”，“拆分列”选择‘monthno’，“阀值”为‘44’。



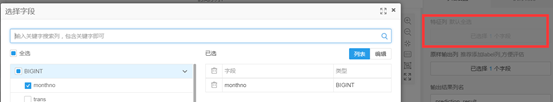
一般来说，在周期取为一个月时，短期预测为对一到两个月以后的预测。通常预测的距离越远预测就越不准确。在本实验中，我们选取4个月作为测试集，前两个月可以看作是短期预测，后两个月可以看作中长期预测。

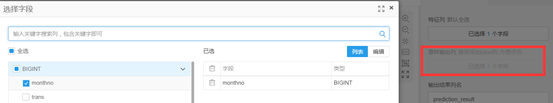
将阀值设为44以后，输出桩1输出的数据为第40、41、42、43个月的销售数据，我们将这四个月的数据选为测试集，来测试模型预测的准确度。

右键单击“测试集”，左键单击“执行该节点”。

在“计算测试集T”的“字段设置”中，“特征列”与“原样输出列”均选择‘monthno’。







在“计算与整合测试集T,C,K,S,R”输入如下代码：

select distinct monthno

,u\_test,t\_test,s\_test,c\_test,r\_test

,(u\_test\*t\_test\*s\_test\*c\_test\*r\_test) as dq\_adj\_p

from

(select c.monthno

,a.U as u\_test,b.s\_test

,avg(a.c) over(partition by 1) AS c\_test

,c.t\_test,1 as r\_test

from

(select monthno

,C,U

,1 as justforjoin

from ${t1}) a

left outer join

(select monthno

,monthno %13 as monthno\_in1y

,prediction\_result as t\_test

,1 as justforjoin

from ${t3}) c

on a.justforjoin = c.justforjoin

left outer join

(select monthno\_in1y

,s as s\_test

from ${t2}

) b

on b.monthno\_in1y = c.monthno\_in1y) d

这段代码整合了训练集中的参数，并计算了测试集相应的参数。

在“归一化还原时序”输入如下代码：

select distinct a.monthno

,a.dq\_adj\_p

,b.max

,b.min

,a.dq\_adj\_p\*(b.max-b.min)+b.min as dq\_p

from

(select \* from ${t1}) a

left outer join

(select \* from ${t2}) b

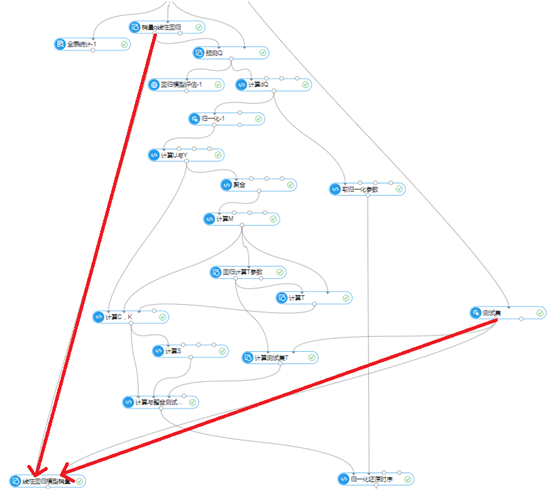
on a.r\_test = b.justforjoin

这段代码根据公式 还原了序列。

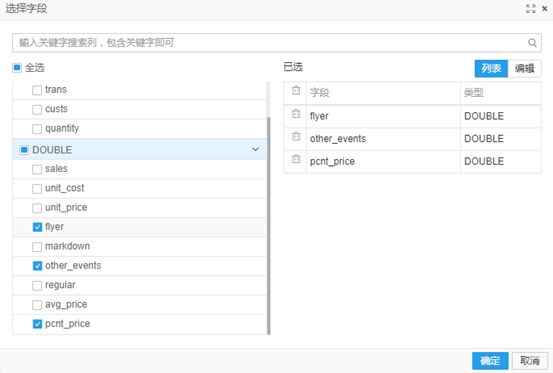
右键单击“计算测试集T”，左键单击“从此处开始执行”。

**3.预测线性回归销量**

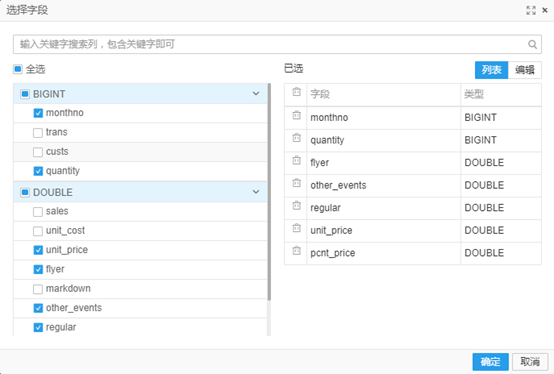
从左侧拖出一个“预测”，重命名为“线性回归模型销量”按如下方式连接。



左键单击“线性回归模型销量”，在右侧字段设置中，“特征列”选择‘flyer’，‘other\_events’,’pcnt\_price’。



“原样输出列”选择‘monthno’，‘quantity’，‘flyer’，‘other\_events’，‘regular’，‘unit\_price’，‘pcnt\_price’7个字段。



右键单击“线性回归模型销量”，左键单击“执行该节点”。

**4.计算结果与有效性评估**

从左侧拖出一个“SQL脚本”，按如下方式重命名并连接。



随后输入如下代码：

select \*

,abs((quantity-quantity\_p)/quantity) as error

from

(select a.monthno

,a.prediction\_result as quantity\_linnermodel

,b.dq\_p as quantity\_timeseries

,a.prediction\_result + b.dq\_p as quantity\_p

,a.quantity

from (select \* from ${t1}) a

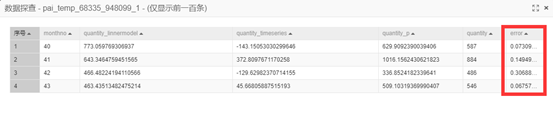
left outer join (select \* from ${t2}) b

on a. monthno = b.monthno )c

这段代码计算了线性模型预测的销量与时序分解模型预测的销量之和，并计算了模型的误差。

执行该节点，待运行结束后，右键单击“预测销量”，左键单击“查看数据”：

左键单击‘monthno’右侧的小三角，使其按月份排序。



从模型的预测准确率来看：

第一个月的误差在10%以内，为7.309%，预测的前两个月误差较小，到了第3个月误差较大，说明短期内模型的预测还是有一定效果的，但是无法保证长期预测准确率。由于长期预测准确率的下降，模型重跑的调度周期设为一个月比较好。