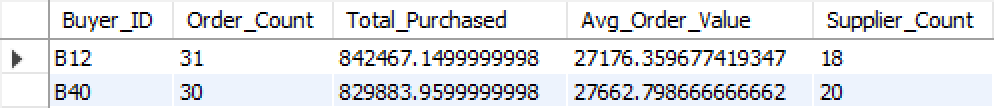
**反向保理潜在客户识别的简单规则**

**第一阶段：识别核心企业和上下游供应链企业**

1. ​**识别核心企业**​：通过总采购金额、订单数量等指标筛选出核心企业（B12和B40）。

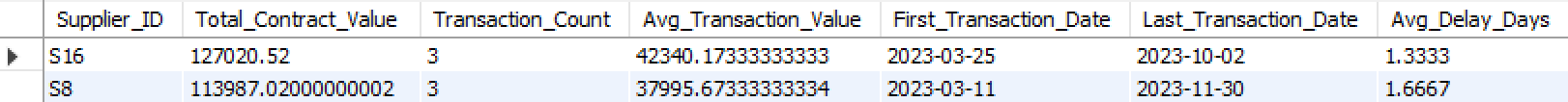


1. ​**检查供应商的双重身份**​：确定哪些供应商同时也是买家，为多级供应链分析做准备。

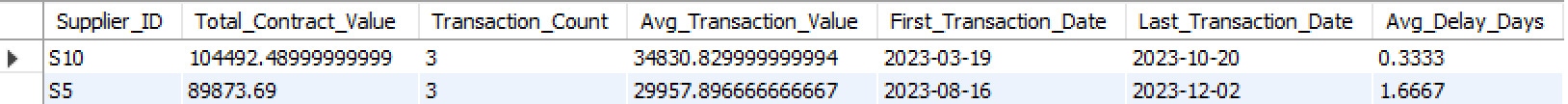
数据集内没有双重身份的厂商，没有多级供应链。

1. ​**分析核心企业的供应商**​：分别对B12和B40的供应商进行分析，按交易总额排序，找出主要供应商。

B12的核心供应商为：S16和S8



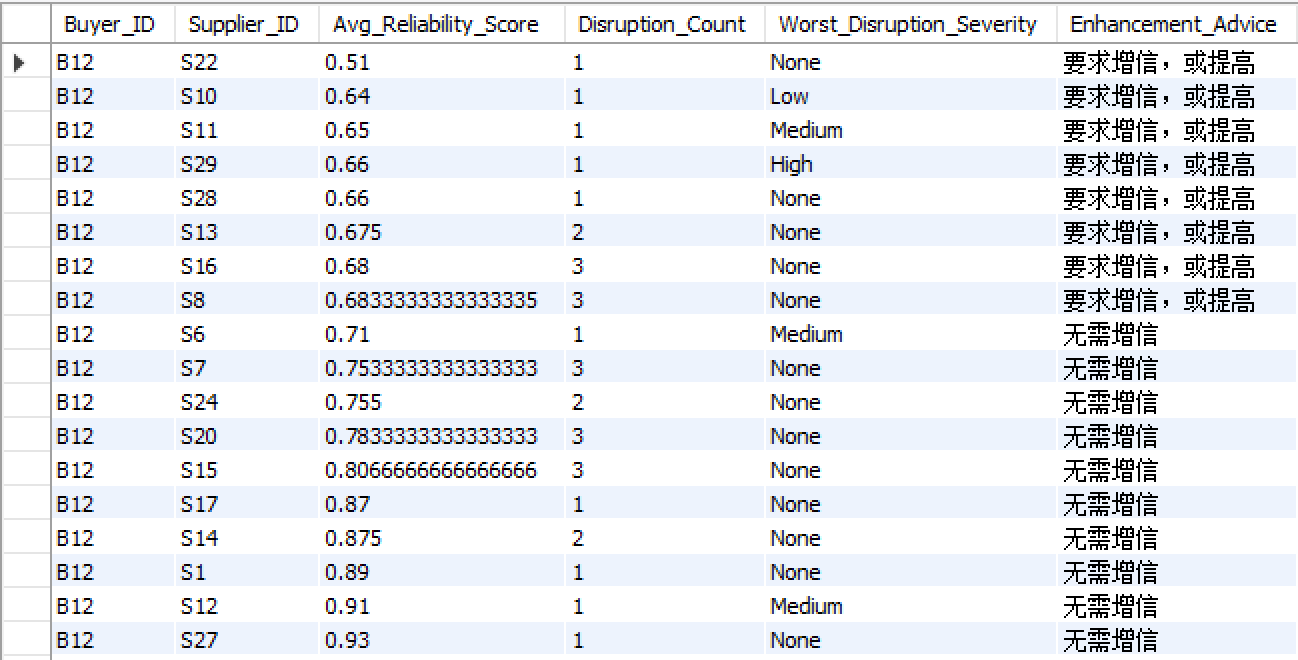
B40的核心供应商为：S10和S5



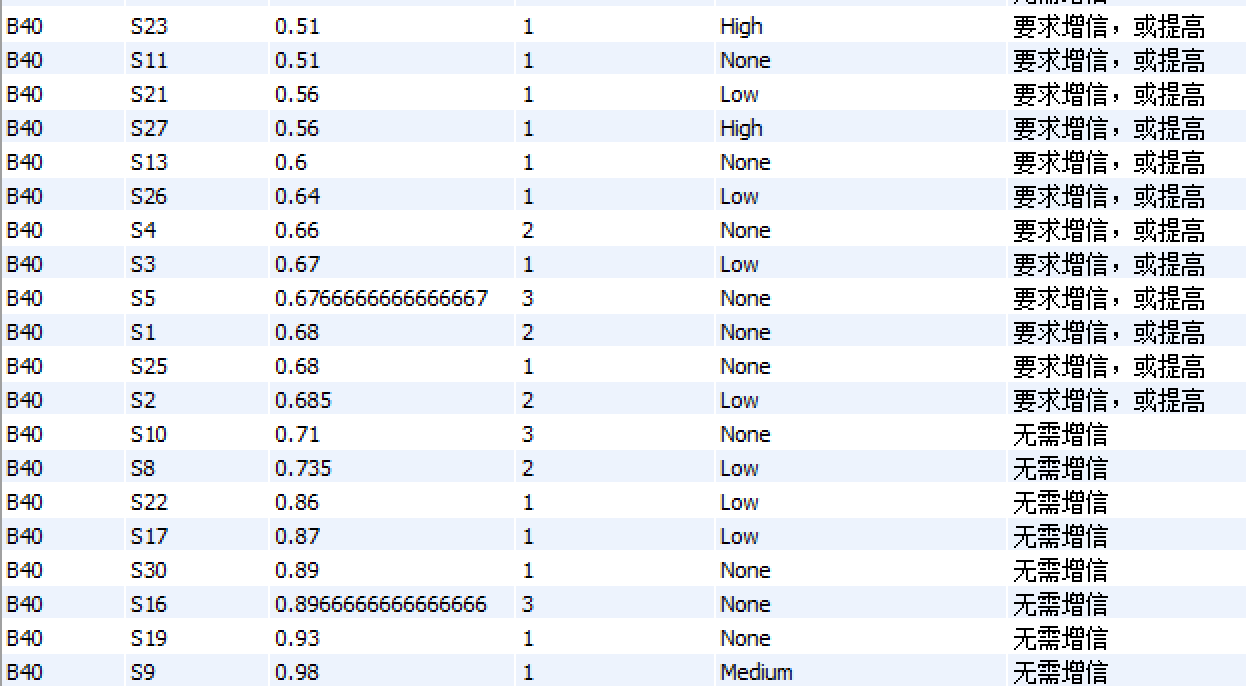
**第二阶段：供应商风险评估与融资策略匹配**

1. ​**供应商可靠性评估**​：计算每个供应商的平均可靠性得分，并根据得分给出增信建议。

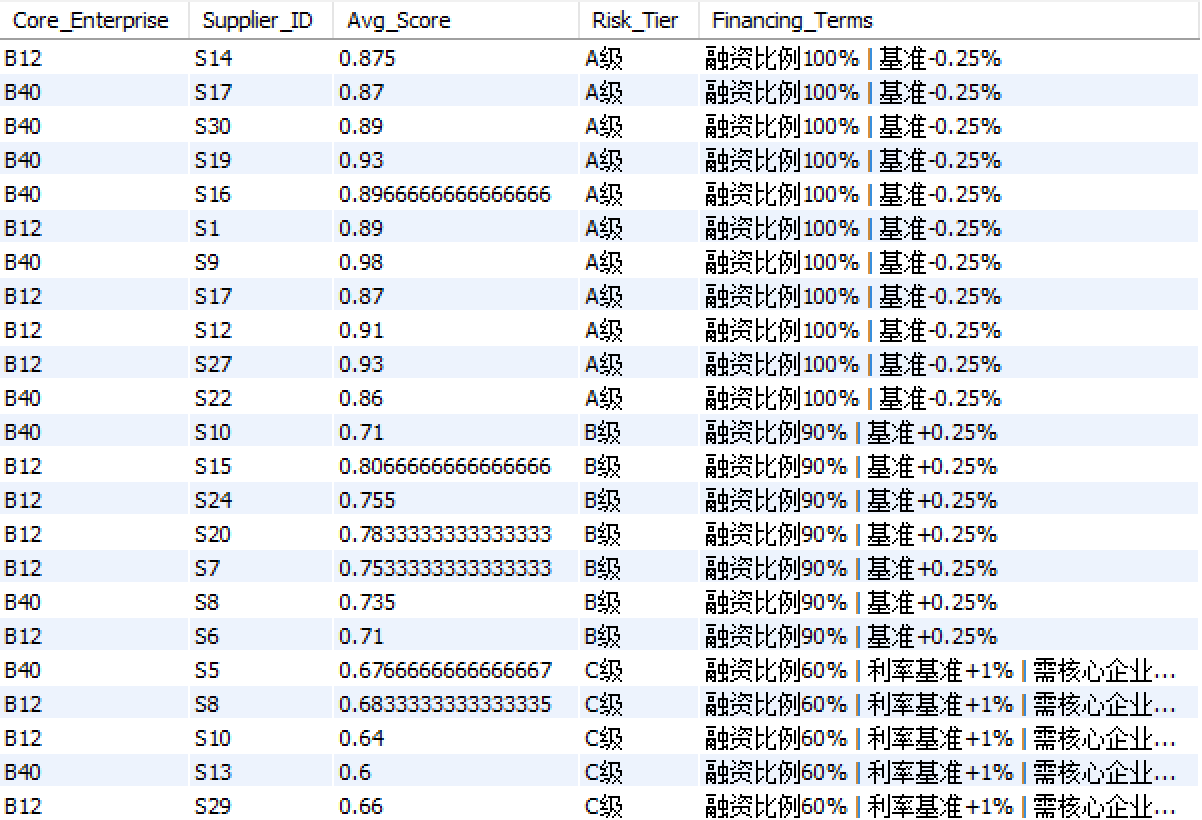
B12供应商风险评估



B40供应商风险评估

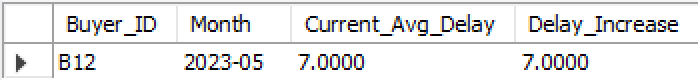


1. ​**风险分级与融资策略**​：根据可靠性得分将供应商分为A、B、C三级，并为每级设计不同的融资方案。



**第三阶段：持续监控**

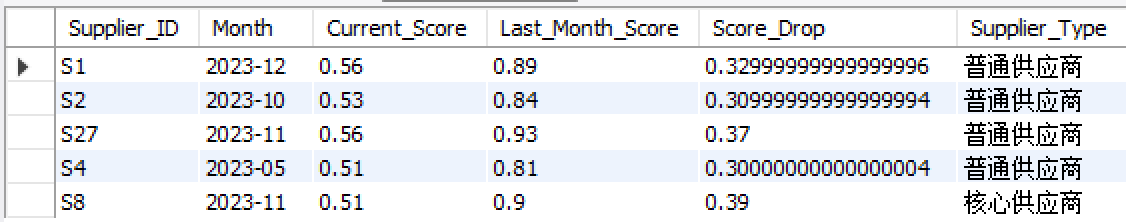
1. ​**监控核心企业付款延迟**​：计算每个核心企业每月的平均付款延迟，并监控延迟的环比增加情况（超过5天触发预警）。



仅有核心企业B12在2023年5月触发付款延迟预警。

1. ​**监控供应商可靠性变化**​：计算每个供应商每月平均可靠性得分的环比变化，若单月下降超过0.25则触发预警。

预警名单如下



*Mysql代码*

-- 第一步：识别核心企业和上下游供应链企业 (按总采购金额排序)

-- 1.1 按照订单总额、订单次数、平均订单价值、供应商数量来识别核心企业

SELECT

Buyer\_ID,

COUNT(DISTINCT `Order\_ID`) AS Order\_Count, -- 订单总数，反映业务频繁度

SUM(`Order\_Value\_USD`) AS Total\_Purchased, -- 总采购金额，核心指标

AVG(`Order\_Value\_USD`) AS Avg\_Order\_Value, -- 平均订单价值

COUNT(DISTINCT `Supplier\_ID`) AS Supplier\_Count -- 供应商数量，反映供应链广度

FROM raws

GROUP BY Buyer\_ID

HAVING COUNT(DISTINCT `Order\_ID`) >= 30 and SUM(`Order\_Value\_USD`) > 80000

ORDER BY

Total\_Purchased DESC; -- 按总采购额降序排列，排名最高的就是最核心的企业

-- 至此我们找到两个关键的核心企业，b12和b40

-- 还要检查供应商是否同时有买家和供应商双重身份，从而决定下一步是否梳理多级供应商

SELECT Supplier\_ID

FROM raws

WHERE Supplier\_ID IN (SELECT DISTINCT Buyer\_ID FROM raws);

-- 这里的buyers和suppliers是带前缀编码的字符串，所以肯定不会重复。

-- 1.2 查询核心企业(B12和B40)的主要供应商及合作情况

SELECT

`Supplier\_ID`,

SUM(`Order\_Value\_USD`) AS Total\_Contract\_Value, -- 与该供应商的总交易额

COUNT(`Order\_ID`) AS Transaction\_Count, -- 交易次数

AVG(`Order\_Value\_USD`) AS Avg\_Transaction\_Value, -- 平均交易额

MIN(`Order\_Date`) AS First\_Transaction\_Date, -- 首次合作时间

MAX(`Order\_Date`) AS Last\_Transaction\_Date, -- 最近合作时间

AVG(`Delay\_Days`) AS Avg\_Delay\_Days -- 平均交付延迟天数(评估供应稳定性)

FROM

raws

WHERE

Buyer\_ID = 'B12' -- 指定核心企业ID

GROUP BY

`Supplier\_ID`

ORDER BY

SUM(`Order\_Value\_USD`) DESC; -- 按交易总额降序，排名最高的就是最重要的战略供应商即S16和S8

SELECT

`Supplier\_ID`,

SUM(`Order\_Value\_USD`) AS Total\_Contract\_Value, -- 与该供应商的总交易额

COUNT(`Order\_ID`) AS Transaction\_Count, -- 交易次数

AVG(`Order\_Value\_USD`) AS Avg\_Transaction\_Value, -- 平均交易额

MIN(`Order\_Date`) AS First\_Transaction\_Date, -- 首次合作时间

MAX(`Order\_Date`) AS Last\_Transaction\_Date, -- 最近合作时间

AVG(`Delay\_Days`) AS Avg\_Delay\_Days -- 平均交付延迟天数(评估供应稳定性)

FROM

raws

WHERE

Buyer\_ID = 'B40'

GROUP BY

`Supplier\_ID`

ORDER BY

SUM(`Order\_Value\_USD`) DESC; -- 按交易总额降序，排名最高的就是最重要的战略供应商即S10和S5

-- 1.3 评估所有供应商的可靠性，这里设置简单规则，如果主供应商平均可靠性分数低于0.70，则要求企业增信

-- 1.3 评估所有供应商的可靠性

SELECT

`Buyer\_ID`,

`Supplier\_ID`,

AVG(`Supplier\_Reliability\_Score`) AS Avg\_Reliability\_Score,

COUNT(CASE WHEN `Disruption\_Type` IS NOT NULL THEN 1 END) AS Disruption\_Count,

MAX(`Disruption\_Severity`) AS Worst\_Disruption\_Severity,

-- 增加一个简单规则

CASE

WHEN AVG(`Supplier\_Reliability\_Score`) < 0.7 THEN '要求增信，或提高'

ELSE '无需增信'

END AS Enhancement\_Advice

FROM

raws

WHERE

`Buyer\_ID` IN ('B40', 'B12')

GROUP BY

`Buyer\_ID`, `Supplier\_ID` -- 分别计算给B12和B40供货的供应商是否需要增信

ORDER BY

`Buyer\_ID`, Avg\_Reliability\_Score ASC; -- 按分数升序排列，风险高的（分数低的）排在前面

-- 2 简单规则的供应商风险分级与融资策略匹配

SELECT

`Buyer\_ID` AS Core\_Enterprise,

`Supplier\_ID`,

AVG(`Supplier\_Reliability\_Score`) AS Avg\_Score,

-- 风险分级

CASE

WHEN AVG(`Supplier\_Reliability\_Score`) >= 0.85 THEN 'A级'

WHEN AVG(`Supplier\_Reliability\_Score`) BETWEEN 0.7 AND 0.84 THEN 'B级'

ELSE 'C级'

END AS Risk\_Tier,

-- 差异化融资方案

CASE

WHEN AVG(`Supplier\_Reliability\_Score`) >= 0.85 THEN '融资比例100% | 基准-0.25%'

WHEN AVG(`Supplier\_Reliability\_Score`) BETWEEN 0.7 AND 0.84 THEN '融资比例90% | 基准+0.25%'

ELSE '融资比例60% | 利率基准+1% | 需核心企业担保'

END AS Financing\_Terms

FROM raws

WHERE `Buyer\_ID` IN ('B12','B40')

GROUP BY `Buyer\_ID`, `Supplier\_ID`;

-- 3 持续监控核心企业和供应商交易状况

-- 3.1 监控核心企业付款延迟恶化

SELECT

Buyer\_ID,

Month,

Current\_Avg\_Delay,

Delay\_Increase

FROM (

SELECT

Buyer\_ID,

DATE\_FORMAT(Order\_Date, '%Y-%m') AS Month,

AVG(Delay\_Days) AS Current\_Avg\_Delay,

AVG(Delay\_Days) - LAG(AVG(Delay\_Days), 1) OVER (

PARTITION BY Buyer\_ID

ORDER BY DATE\_FORMAT(Order\_Date, '%Y-%m')

) AS Delay\_Increase

FROM raws

WHERE Buyer\_ID IN ('B12','B40')

GROUP BY Buyer\_ID, DATE\_FORMAT(Order\_Date, '%Y-%m')

) AS monthly\_delays

WHERE Delay\_Increase >= 5; -- 付款延迟按月环比增加超过5天就触发预警

-- 3.2 监控供应商可靠性（简单规则月平均可靠性下降0.25触发）

-- 3.2 监控供应商可靠性

SELECT

Supplier\_ID,

Month,

Current\_Score,

Last\_Month\_Score,

Score\_Drop,

-- 核心供应商标识

CASE

WHEN Supplier\_ID IN ('S16', 'S8', 'S10', 'S5') THEN '核心供应商'

ELSE '普通供应商'

END AS Supplier\_Type

FROM (

SELECT

Supplier\_ID,

DATE\_FORMAT(Order\_Date, '%Y-%m') AS Month,

AVG(Supplier\_Reliability\_Score) AS Current\_Score,

LAG(AVG(Supplier\_Reliability\_Score), 1) OVER (

PARTITION BY Supplier\_ID

ORDER BY DATE\_FORMAT(Order\_Date, '%Y-%m')

) AS Last\_Month\_Score,

(LAG(AVG(Supplier\_Reliability\_Score), 1) OVER (

PARTITION BY Supplier\_ID

ORDER BY DATE\_FORMAT(Order\_Date, '%Y-%m')

) - AVG(Supplier\_Reliability\_Score)) AS Score\_Drop,

Buyer\_ID

FROM raws

WHERE Buyer\_ID IN ('B12', 'B40')

GROUP BY Supplier\_ID, DATE\_FORMAT(Order\_Date, '%Y-%m'), Buyer\_ID

) AS supplier\_scores

WHERE Score\_Drop > 0.25; -- B12和B40的供应商单月可靠性下降超过0.25触发