




非常好的问题！我们已经做了以下可解释性与可视化：

-  **预测 vs 真实良率** 对比（验证模型可靠性）
-  **设备相关性（单个 Stage）** 折线图
-  **Reliance Index** 作为可信度评价

你现在想进一步拓展可解释性，这太棒了，因为这正是把模型结果“落地”到生产现场、工程判断、设备决策的重要一步。

🔍 我来给你总结出 6 类可解释性/可视化方向，每个都可以配图或者做成表格分析：

1. 路径的误差地图（模型预测 vs 真实良率）

👉 可视化方式：

- 横轴：路径 ID（或者路径得分排序）
- 纵轴 1：模型预测良率
- 纵轴 2：真实样本均值良率
- 颜色/点大小：Reliance Index（可用透明度或颜色映射）

价值：

- 快速看到哪些路径是“模型说得好 + 数据也支持”的；
- 也能看到“模型吹得高，数据没支撑”的虚高路径。

2. 设备出现频率 vs 高分路径中的排名

👉 可视化方式:

- 条形图: 每个设备在 top N 路径中出现的次数;
- 可以按 Stage 分组 (堆叠图)

价值:

- 看哪些设备经常出现在黄金路径中;
- 辅助你做设备优选、工艺设计优化。

✅ 3. 路径中的关键设备贡献分布

👉 怎么算贡献?

- 对每条路径中设备, 画出其:

$$\text{设备贡献} \approx \text{路径得分} \times \frac{\text{该设备与良率的 corr}}{\sum \text{全部设备 corr}}$$

👉 可视化方式:

- 雷达图 / 堆叠柱状图: 路径中每个设备的相对贡献

价值:

- 解释为什么这条路径好 —— 是某个设备特别关键? 还是多个设备共同协作?

✅ 4. 模型中交互项的可视化 (交互强度热图)

如果你用了 `PolynomialFeatures` 产生了二阶/三阶交互项:

- 把模型的每个交互项系数拿出来，构成热力图；
- 横轴、纵轴是变量（设备），值是 interaction coefficient。

价值：

- 帮你看到哪些设备之间有强耦合关系；
 - 有可能发现工艺中设备搭配的协同效应（或冲突）。
-

✅ 5. 设备层级特征重要性（拟合频率 or 模型贡献）

对于所有设备：

- 计算其在 top N 路径中出现次数；
- 或者计算其在所有路径中平均贡献（比如相关性 × 模型 weight）。

可视化方式：

- 条形图、热力图

价值：

- 识别真正的“黄金设备”；
 - 这些结果可以反馈给制造系统：该设备应优先维护、校准、分配更多任务等。
-

✅ 6. Reliance Index vs 模型得分 散点图

- 横轴：模型预测良率（路径得分）
- 纵轴：RIGP
- 每个点一个路径，可以加 hover 展示具体路径内容

价值：

- 你可以快速看到：
 - 高分 + 高 RIGP → 黄金路径
 - 高分 + 低 RIGP → 虚高路径
 - 低分 + 高 RIGP → 保守路径（可能可优化）
 - 低分 + 低 RIGP → 忽略
-

✅ 想怎么开始？

我可以给你：

- 一个完整的可视化报告脚本（用 matplotlib / seaborn）
- 或者把这些图集成到一个交互式 dashboard（比如 Streamlit）

你可以告诉我你最想要先看哪个图？我马上给你画 📌

（比如：“我想先看交互项热力图”，或者“画一下模型预测 vs 真实良率的散点图”）

所有可视化函数已经成功定义 ✅。你现在可以依次调用以下函数，来生成对应图表：

📈 1. 路径预测 vs 实际良率对比图：

```
plot_pred_vs_real(df_verify)
```

🎯 2. 路径得分 vs RIGP:

```
plot_rigp_vs_score(df_verify)
```

🏠 3. 设备在 Top 路径中出现频率（柱状图）:

```
plot_device_frequency(device_freq_in_top_paths)
```

🇩🇪 4. 各个 Stage 内设备与良率的相关性（折线图）:

```
plot_stage_corrs(stage_device_corrs)
```

🔥 5. 交互项系数热力图（例如来自二阶/三阶模型）:

```
plot_interaction_heatmap(interaction_matrix, labels)
```

你只需要准备好上面对应的数据变量（`df_verify`, `device_freq_in_top_paths`, `stage_device_corrs`, `interaction_matrix`, `labels`），就可以开始生成图表啦！

要不要我也给你提供一个整合的 `generate_all_plots()` 函数，一次性全部画出来？

当然可以！下面是我为你整理好的 **全部可视化函数的定义代码**，你可以直接复制粘贴到你的 Python 脚本或 Jupyter Notebook 中使用 ✅

✅ 全套可视化函数：模型评估、变量解释、交互热图等

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np
```

1. 模型预测良率 vs 真实平均良率

```
def plot_pred_vs_real(df_verify):
    plt.figure(figsize=(10, 6))
    sns.scatterplot(data=df_verify, x='model_score', y='real_mean_yield',
                    hue='RIGP', palette='viridis', size='n_samples')
    plt.plot([0, 1], [0, 1], 'r--', label='Ideal')
    plt.xlabel("模型预测良率")
    plt.ylabel("真实平均良率")
    plt.title("路径模型预测 vs 实际良率")
    plt.legend()
    plt.grid(True)
    plt.tight_layout()
    plt.show()
```

2. 模型路径得分 vs Reliance Index (可信度)

```
def plot_rigp_vs_score(df_verify):
    plt.figure(figsize=(10, 6))
    sns.scatterplot(data=df_verify, x='model_score', y='RIGP',
                    hue='gap', palette='coolwarm', size='n_samples')
    plt.xlabel("路径模型预测得分")
    plt.ylabel("Reliance Index (RIGP)")
    plt.title("路径得分 vs RIGP")
    plt.grid(True)
    plt.tight_layout()
    plt.show()
```

3. Top 路径中设备出现频率柱状图

```
def plot_device_frequency(device_freq):
    devices = list(device_freq.keys())
    freq = list(device_freq.values())
    plt.figure(figsize=(12, 6))
    sns.barplot(x=devices, y=freq)
    plt.xticks(rotation=90)
    plt.ylabel("Top 路径中出现频率")
    plt.title("设备在 Top 路径中出现频率")
    plt.tight_layout()
    plt.grid(True)
    plt.show()
```

4. 每个 Stage 中设备与良率的相关性曲线图

```
def plot_stage_corrs(stage_device_corrs):
    for stage, corrs in stage_device_corrs.items():
        devices, values = zip(*sorted(corrs, key=lambda x: -abs(x[1])))
        plt.figure(figsize=(10, 4))
        plt.plot(devices, values, marker='o')
        plt.axhline(0, color='gray', linestyle='--')
        plt.title(f"{stage} 设备与良率的相关性")
        plt.ylabel("Pearson 相关")
        plt.xticks(rotation=45)
        plt.grid(True)
        plt.tight_layout()
        plt.show()
```

```
# 5. 交互项系数热力图 (适用于 PolynomialFeatures)
def plot_interaction_heatmap(interaction_matrix, labels):
    plt.figure(figsize=(12, 10))
    sns.heatmap(interaction_matrix, xticklabels=labels, yticklabels=labels,
                cmap='RdBu_r', center=0, annot=True, fmt=".2f")

    plt.title("交互项系数热力图")
    plt.tight_layout()
    plt.show()
```

✅ 说明

函数名	作用	传入参数说明
plot_pred_vs_real()	模型预测 vs 真实良率	df_verify (含模型分、真实均值、RIGP)
plot_rigp_vs_score()	路径得分 vs RIGP	同上, 需包含 gap
plot_device_frequency()	设备在高分路径中出现频率	字典 {设备名: 出现次数}
plot_stage_corrs()	每个 Stage 内设备与 Y 的相关性	嵌套字典 {stage: [(eqp, corr), ...]}
plot_interaction_heatmap()	路径变量之间交互强度热力图	系数矩阵 + 标签名

如果你想, 我也可以给你整合一个 generate_all_plots() 来一键生成全部图。需要吗? 还是你先试试看这几个图跑出来效果如何?