

# 基于Cox回归模型的子宫内膜异位症术后复发风险研究方案

## 研究方案

### 1. 研究目的

- 主要目的：**利用Cox回归模型分析子宫内膜异位症患者腹腔镜术后复发时间与相关因素（如术前痛经史、孕次、囊肿直径、治疗方案等）之间的关系，评估各因素对复发时间的影响。
- 次要目的：**比较不同促性腺激素释放激素激动剂（GnRHa）方案对术后复发时间的影响。

### 2. 研究设计

- 研究类型：**回顾性队列研究。
- 数据来源：**利用文献中提供的150例子宫内膜异位症患者的数据，包括临床病理资料、治疗方案、随访结果等。
- 研究对象：**2019年1月至2022年6月在宁波市第七医院及复旦大学附属妇产科医院接受腹腔镜手术治疗的子宫内膜异位症患者。
- 分组：**根据术后GnRHa方案分为两组：甲组（醋酸亮丙瑞林微球）和乙组（醋酸戈舍瑞林缓释植入剂）。

### 3. 研究变量

- 因变量：**复发时间（从手术到复发的时间，单位：月）。
- 自变量：**
  - 临床病理特征：年龄、病程、BMI、初潮年龄、痛经史、孕次、产次、吸烟、饮酒、盆腔粘连、合并子宫腺肌病、合并子宫肌瘤、既往宫腔操作史、既往子宫内膜异位症手术史、囊肿最大直径、深部子宫内膜异位灶、后穹窿触痛、ASRM分期、盆腔病变位置、术前血清CA125水平等。
  - 治疗方案：甲组（醋酸亮丙瑞林微球）和乙组（醋酸戈舍瑞林缓释植入剂）。

### 4. 数据处理

- 数据收集：**从文献中提取相关数据，包括患者的临床病理资料、治疗方案、随访结果等。
- 数据清洗：**检查数据的完整性，处理缺失值和异常值。

- **变量编码**：对分类变量进行编码，如痛经史（0=无，1=有）、孕次（0=0次，1=1次，2= $\geq$ 2次）等。

## 5. 统计分析

- **描述性统计**：对患者的基线特征进行描述性统计分析，包括均值、标准差、中位数、四分位数等。
- **单因素Cox回归分析**：对每个自变量与复发时间进行单因素Cox回归分析，初步筛选出与复发时间相关的因素。
- **多因素Cox回归分析**：将单因素分析中显著的因素纳入多因素Cox回归模型，评估各因素对复发时间的独立影响。
- **模型评估**：
  - **比例风险假设检验**：通过Schoenfeld残差检验和图形诊断方法，验证Cox模型的比例风险假设是否成立。
  - **模型拟合优度**：通过计算C-index等指标，评估模型的拟合优度和预测能力。

## 6. 结果呈现

- **风险比 (HR)**：报告各因素的HR及其95%置信区间，评估其对复发时间的影响。
- **生存曲线**：绘制Kaplan-Meier生存曲线，比较不同组别（如不同治疗方案）的生存率差异。
- **模型结果**：以表格形式呈现Cox回归模型的分析结果，包括变量、HR、95%CI、P值等。

## 7. 讨论

- **结果解释**：结合文献和临床背景，讨论各因素对复发时间的影响及其可能的机制。
- **临床意义**：探讨研究结果对临床实践的指导意义，如如何根据患者特征选择更合适的治疗方案。
- **局限性**：分析研究的局限性，如样本量、数据来源、随访时间等对结果的影响。
- **未来研究方向**：提出未来研究的建议，如扩大样本量、延长随访时间、进一步验证模型的预测能力等。

# 研究计划

## 1. 第一阶段：研究准备 (1个月)

- **文献回顾**：进一步查阅相关文献，了解子宫内膜异位症术后复发的研究现状和Cox回归模型的应用。
- **数据收集**：从文献中提取数据，建立数据库。
- **数据清洗与预处理**：检查数据的完整性，处理缺失值和异常值，对变量进行编码。

## 2. 第二阶段：统计分析（2个月）

- 描述性统计分析**：对患者的基线特征进行描述性统计分析。
- 单因素Cox回归分析**：对每个自变量与复发时间进行单因素Cox回归分析。
- 多因素Cox回归分析**：将单因素分析中显著的因素纳入多因素Cox回归模型，进行多因素分析。
- 模型评估**：验证比例风险假设，评估模型的拟合优度和预测能力。

## 3. 第三阶段：结果撰写与讨论（1个月）

- 结果撰写**：撰写研究报告，包括研究背景、方法、结果和讨论。
- 讨论与结论**：结合文献和临床背景，讨论研究结果的意义，提出临床建议和未来研究方向。

## 4. 第四阶段：论文投稿与发表（2个月）

- 论文撰写**：根据研究结果撰写学术论文，投稿至相关期刊。
- 修改与完善**：根据审稿意见进行修改和完善，争取论文发表。

## 注意事项

- 数据质量**：确保数据的准确性和完整性，避免因数据质量问题影响研究结果。
- 模型假设**：在进行Cox回归分析时，需验证比例风险假设是否成立，若不成立，可考虑分层分析或时间依赖协变量。
- 样本量**：虽然文献提供了150例患者的数据，但样本量相对较小，可能会影响模型的稳定性和预测能力，需谨慎解释结果。
- 临床意义**：研究结果需结合临床背景进行解释，避免过度解读统计学结果。

## 模拟文献数据

### 描述特征

请总结这份文献中提到的各个变量的特征，包括均值、标准差、中位数、四分位数等。我的目的是，接下来根据这些变量的特征生成模拟数据

### 回答

根据文献内容，以下是提到的各个变量的特征总结。需要注意的是，文献中主要提供了均值和标准差等信息，而中位数和四分位数等未明确提及，因此在生成模拟数据时，可能需要根据实际情况进行合理假

设或推断。

## 患者基本信息

- **年龄**
  - 甲组：27~37岁，均值为31.64岁，标准差为3.67岁。
  - 乙组：28~36岁，均值为30.52岁，标准差为3.36岁。
- **病程**
  - 甲组：1~4年，均值为2.46年，标准差为0.58年。
  - 乙组：1~4年，均值为2.52年，标准差为0.55年。
- **体质量指数 (BMI)**
  - 复发组：均值为20.37 kg/m<sup>2</sup>，标准差为1.43 kg/m<sup>2</sup>。
  - 未复发组：均值为20.24 kg/m<sup>2</sup>，标准差为1.39 kg/m<sup>2</sup>。
- **初潮年龄**
  - 复发组：均值为13.62岁，标准差为0.75岁。
  - 未复发组：均值为13.76岁，标准差为0.82岁。
- **月经时长**
  - 复发组：<7天的占比54.29%，≥7天的占比45.71%。
  - 未复发组：<7天的占比39.13%，≥7天的占比60.87%。

## 疾病相关特征

- **术前痛经史**
  - 复发组：有痛经史的占比57.14% (20/35)，无痛经史的占比42.86% (15/35)。
  - 未复发组：有痛经史的占比31.30% (36/115)，无痛经史的占比68.70% (79/115)。
- **术前孕次**
  - 复发组：0次占比34.29% (12/35)，1次占比42.86% (15/35)，≥2次占比22.86% (8/35)。
  - 未复发组：0次占比65.22% (75/115)，1次占比24.35% (28/115)，≥2次占比10.43% (12/115)。
- **囊肿最大直径**
  - 复发组：<6cm的占比40.00% (14/35)，≥6cm的占比60.00% (21/35)。
  - 未复发组：<6cm的占比72.17% (83/115)，≥6cm的占比27.83% (32/115)。
- **深部子宫内膜异位灶**
  - 复发组：有深部子宫内膜异位灶的占比51.43% (18/35)。
  - 未复发组：有深部子宫内膜异位灶的占比28.70% (33/115)。
- **后穹窿触痛**
  - 复发组：有后穹窿触痛的占比45.71% (16/35)。
  - 未复发组：有后穹窿触痛的占比24.35% (28/115)。
- **ASRM分期**

- 复发组：I<sub>II</sub>期占比34.29% (12/35) , III<sub>IV</sub>期占比65.71% (23/35) 。
- 未复发组：I<sub>II</sub>期占比56.52% (65/115) , III<sub>IV</sub>期占比43.48% (50/115) 。

- **盆腔病变位置**

- 复发组：单侧占比45.71% (16/35) , 双侧占比54.29% (19/35) 。
- 未复发组：单侧占比67.83% (78/115) , 双侧占比32.17% (37/115) 。

- **术前血清CA125**

- 复发组：均值为45.67 U/L, 标准差为7.26 U/L。
- 未复发组：均值为43.25 U/L, 标准差为6.21 U/L。

## 治疗后卵巢功能指标

- **血清雌二醇 (E2)**

- 甲组：
  - 治疗前：均值为67.52 pg/ml, 标准差为6.82 pg/ml。
  - 治疗后1年：均值为41.56 pg/ml, 标准差为5.15 pg/ml。
- 乙组：
  - 治疗前：均值为67.43 pg/ml, 标准差为6.59 pg/ml。
  - 治疗后1年：均值为49.57 pg/ml, 标准差为6.34 pg/ml。

- **抗苗勒管激素 (AMH)**

- 甲组：
  - 治疗前：均值为2.33 ng/ml, 标准差为0.60 ng/ml。
  - 治疗后1年：均值为4.18 ng/ml, 标准差为0.74 ng/ml。
- 乙组：
  - 治疗前：均值为2.34 ng/ml, 标准差为0.61 ng/ml。
  - 治疗后1年：均值为3.47 ng/ml, 标准差为0.75 ng/ml。

- **窦卵泡计数 (AFC)**

- 甲组：
  - 治疗前：均值为7.56个, 标准差为1.25个。
  - 治疗后1年：均值为10.42个, 标准差为1.18个。
- 乙组：
  - 治疗前：均值为7.64个, 标准差为1.22个。
  - 治疗后1年：均值为9.54个, 标准差为1.12个。

## 其他变量

- **吸烟**

- 复发组：占比20.00% (7/35) 。
- 未复发组：占比19.13% (22/115) 。

- **饮酒**

- 复发组：占比11.43% (4/35) 。
- 未复发组：占比7.83% (9/115) 。
- 盆腔粘连
  - 复发组：占比65.71% (23/35) 。
  - 未复发组：占比74.78% (86/115) 。
- 合并子宫腺肌病
  - 复发组：占比17.14% (6/35) 。
  - 未复发组：占比13.91% (16/115) 。
- 合并子宫肌瘤
  - 复发组：占比22.86% (8/35) 。
  - 未复发组：占比18.26% (21/115) 。
- 既往宫腔操作史
  - 复发组：占比20.00% (7/35) 。
  - 未复发组：占比15.65% (18/115) 。
- 既往子宫内膜异位症手术史
  - 复发组：占比11.43% (4/35) 。
  - 未复发组：占比12.17% (14/115) 。

临床疗效相关

- 妊娠率
  - 甲组：72.00% (54/75) 。
  - 乙组：56.00% (42/75) 。
- 复发率
  - 甲组：14.67% (11/75) 。
  - 乙组：32.00% (24/75) 。
- 总有效率
  - 甲组：76.00% (57/75) 。
  - 乙组：60.00% (45/75) 。

模拟数据代码

上述是文献中各个变量的具体描述，包括均值、标准差、中位数、四分位数等。  
请按照这份文献中提到的数据以及上述描述，为我生成一份模拟数据，生成的数据格式为csv格式  
要包含这份文献中提到的各种变量名，数据量要达到这篇文献的要求，至少要有150条数据，增加  
一列患者ID，用于编号，从1到150  
此外，具体的变量的变量值你自动生成，要求生成的数据和文献中提到的数据符合且符合上述的描

述

请您根据上述的要求，为我写一份用于生成符合上述要求的python代码

好的，以下是一个Python代码示例，用于生成符合上述要求的模拟数据，并保存为CSV文件。我们将使用numpy和pandas库来生成数据，并确保数据的分布符合文献中的描述。

```

import numpy as np
import pandas as pd

# 设置随机种子以确保结果可复现
np.random.seed(42)

# 患者数量
n = 150

def normalize_probabilities(probs):
    """将概率列表标准化以确保其总和为1。"""
    total = sum(probs)
    return [p / total for p in probs]

# 使用辅助函数调整概率
p_recurrence = normalize_probabilities([0.3429, 0.4286, 0.2286])
p_non_recurrence = normalize_probabilities([0.6522, 0.2435, 0.1043])

# 术前孕次
# 复发组术前孕次: 0次、1次、≥2次的概率
pregnancy_times_recurrence = np.random.choice([0, 1, 2], size=35, p=p_recurrence)
# 未复发组术前孕次: 0次、1次、≥2次的概率
pregnancy_times_non_recurrence = np.random.choice([0, 1, 2], size=115, p=p_non_recurrence)
# 合并复发组和未复发组的术前孕次数据
pregnancy_times = np.concatenate((pregnancy_times_recurrence, pregnancy_times_non_recurrence))

# 患者ID, 从1到150
patient_id = np.arange(1, n + 1)

# 年龄
# 甲组年龄: 正态分布, 均值31.64岁, 标准差3.67岁, 75个样本
age_group_a = np.random.normal(31.64, 3.67, 75)
# 乙组年龄: 正态分布, 均值30.52岁, 标准差3.36岁, 75个样本
age_group_b = np.random.normal(30.52, 3.36, 75)
# 合并甲组和乙组的年龄数据
age = np.concatenate((age_group_a, age_group_b))

# 病程
# 甲组病程: 正态分布, 均值2.46年, 标准差0.58年, 75个样本
duration_group_a = np.random.normal(2.46, 0.58, 75)
# 乙组病程: 正态分布, 均值2.52年, 标准差0.55年, 75个样本
duration_group_b = np.random.normal(2.52, 0.55, 75)
# 合并甲组和乙组的病程数据

```



```

duration = np.concatenate((duration_group_a, duration_group_b))

# 体质量指数 (BMI)
# 复发组BMI: 正态分布, 均值20.37, 标准差1.43, 35个样本
bmi_recurrence = np.random.normal(20.37, 1.43, 35)
# 未复发组BMI: 正态分布, 均值20.24, 标准差1.39, 115个样本
bmi_non_recurrence = np.random.normal(20.24, 1.39, 115)
# 合并复发组和未复发组的BMI数据
bmi = np.concatenate((bmi_recurrence, bmi_non_recurrence))

# 初潮年龄
# 复发组初潮年龄: 正态分布, 均值13.62岁, 标准差0.75岁, 35个样本
menarche_age_recurrence = np.random.normal(13.62, 0.75, 35)
# 未复发组初潮年龄: 正态分布, 均值13.76岁, 标准差0.82岁, 115个样本
menarche_age_non_recurrence = np.random.normal(13.76, 0.82, 115)
# 合并复发组和未复发组的初潮年龄数据
menarche_age = np.concatenate((menarche_age_recurrence, menarche_age_non_recurrence))

# 月经时长
# 复发组月经时长: <7天占比54.29%, ≥7天占比45.71%
period_length_recurrence = np.random.choice([0, 1], size=35, p=[0.5429, 0.4571])
# 未复发组月经时长: <7天占比39.13%, ≥7天占比60.87%
period_length_non_recurrence = np.random.choice([0, 1], size=115, p=[0.3913, 0.6087])
# 合并复发组和未复发组的月经时长数据
period_length = np.concatenate((period_length_recurrence, period_length_non_recurrence))

# 术前痛经史
# 复发组术前痛经史: 无痛经占比42.86%, 有痛经占比57.14%
dysmenorrhea_recurrence = np.random.choice([0, 1], size=35, p=[0.4286, 0.5714])
# 未复发组术前痛经史: 无痛经占比68.70%, 有痛经占比31.30%
dysmenorrhea_non_recurrence = np.random.choice([0, 1], size=115, p=[0.6870, 0.3130])
# 合并复发组和未复发组的术前痛经史数据
dysmenorrhea = np.concatenate((dysmenorrhea_recurrence, dysmenorrhea_non_recurrence))

# 术前孕次
# 复发组术前孕次: 0次、1次、≥2次的概率
p_recurrence = normalize_probabilities([0.3429, 0.4286, 0.2285])
# 未复发组术前孕次: 0次、1次、≥2次的概率
p_non_recurrence = normalize_probabilities([0.6522, 0.2435, 0.1043])
# 复发组术前孕次
pregnancy_times_recurrence = np.random.choice([0, 1, 2], size=35, p=p_recurrence)
# 未复发组术前孕次
pregnancy_times_non_recurrence = np.random.choice([0, 1, 2], size=115, p=p_non_recurrence)

```

```

# 合并复发组和未复发组的术前孕次数据
pregnancy_times = np.concatenate((pregnancy_times_recurrence, pregnancy_times_non_recurrence))

# 囊肿最大直径
# 复发组囊肿最大直径: <6cm占比40.00%, ≥6cm占比60.00%
cyst_diameter_recurrence = np.random.choice([0, 1], size=35, p=[0.4000, 0.6000])
# 未复发组囊肿最大直径: <6cm占比72.17%, ≥6cm占比27.83%
cyst_diameter_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7217, 0.2783])
# 合并复发组和未复发组的囊肿最大直径数据
cyst_diameter = np.concatenate((cyst_diameter_recurrence, cyst_diameter_non_recurrence))

# 深部子宫内膜异位灶
# 复发组深部子宫内膜异位灶: 无占比48.57%, 有占比51.43%
deep_endometriosis_recurrence = np.random.choice([0, 1], size=35, p=[0.4857, 0.5143])
# 未复发组深部子宫内膜异位灶: 无占比71.30%, 有占比28.70%
deep_endometriosis_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7130, 0.2870])
# 合并复发组和未复发组的深部子宫内膜异位灶数据
deep_endometriosis = np.concatenate((deep_endometriosis_recurrence, deep_endometriosis_non_recurrence))

# 后穹窿触痛
# 复发组后穹窿触痛: 无占比54.29%, 有占比45.71%
posterior_fornix_tenderness_recurrence = np.random.choice([0, 1], size=35, p=[0.5429, 0.4571])
# 未复发组后穹窿触痛: 无占比75.65%, 有占比24.35%
posterior_fornix_tenderness_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7565, 0.2435])
# 合并复发组和未复发组的后穹窿触痛数据
posterior_fornix_tenderness = np.concatenate((posterior_fornix_tenderness_recurrence, posterior_fornix_tenderness_non_recurrence))

# ASRM分期
# 复发组ASRM分期: I~II期占比34.29%, III~IV期占比65.71%
asrm_stage_recurrence = np.random.choice([0, 1], size=35, p=[0.3429, 0.6571])
# 未复发组ASRM分期: I~II期占比56.52%, III~IV期占比43.48%
asrm_stage_non_recurrence = np.random.choice([0, 1], size=115, p=[0.5652, 0.4348])
# 合并复发组和未复发组的ASRM分期数据
asrm_stage = np.concatenate((asrm_stage_recurrence, asrm_stage_non_recurrence))

# 盆腔病变位置
# 复发组盆腔病变位置: 单侧占比45.71%, 双侧占比54.29%
pelvic_lesion_location_recurrence = np.random.choice([0, 1], size=35, p=[0.4571, 0.5429])
# 未复发组盆腔病变位置: 单侧占比67.83%, 双侧占比32.17%
pelvic_lesion_location_non_recurrence = np.random.choice([0, 1], size=115, p=[0.6783, 0.3217])
# 合并复发组和未复发组的盆腔病变位置数据
pelvic_lesion_location = np.concatenate((pelvic_lesion_location_recurrence, pelvic_lesion_location_non_recurrence))

```

```
# 术前血清CA125
# 复发组术前血清CA125: 正态分布, 均值45.67 U/L, 标准差7.26 U/L, 35个样本
ca125_recurrence = np.random.normal(45.67, 7.26, 35)
# 未复发组术前血清CA125: 正态分布, 均值43.25 U/L, 标准差6.21 U/L, 115个样本
ca125_non_recurrence = np.random.normal(43.25, 6.21, 115)
# 合并复发组和未复发组的术前血清CA125数据
ca125 = np.concatenate((ca125_recurrence, ca125_non_recurrence))

# 血清雌二醇 (E2)
# 甲组治疗前E2: 正态分布, 均值67.52 pg/ml, 标准差6.82 pg/ml, 75个样本
e2_before_group_a = np.random.normal(67.52, 6.82, 75)
# 乙组治疗前E2: 正态分布, 均值67.43 pg/ml, 标准差6.59 pg/ml, 75个样本
e2_before_group_b = np.random.normal(67.43, 6.59, 75)
# 合并甲组和乙组的治疗前E2数据
e2_before = np.concatenate((e2_before_group_a, e2_before_group_b))

# 甲组治疗后1年E2: 正态分布, 均值41.56 pg/ml, 标准差5.15 pg/ml, 75个样本
e2_after_group_a = np.random.normal(41.56, 5.15, 75)
# 乙组治疗后1年E2: 正态分布, 均值49.57 pg/ml, 标准差6.34 pg/ml, 75个样本
e2_after_group_b = np.random.normal(49.57, 6.34, 75)
# 合并甲组和乙组的治疗后1年E2数据
e2_after = np.concatenate((e2_after_group_a, e2_after_group_b))

# 抗苗勒管激素 (AMH)
# 甲组治疗前AMH: 正态分布, 均值2.33 ng/ml, 标准差0.60 ng/ml, 75个样本
amh_before_group_a = np.random.normal(2.33, 0.60, 75)
# 乙组治疗前AMH: 正态分布, 均值2.34 ng/ml, 标准差0.61 ng/ml, 75个样本
amh_before_group_b = np.random.normal(2.34, 0.61, 75)
# 合并甲组和乙组的治疗前AMH数据
amh_before = np.concatenate((amh_before_group_a, amh_before_group_b))

# 甲组治疗后1年AMH: 正态分布, 均值4.18 ng/ml, 标准差0.74 ng/ml, 75个样本
amh_after_group_a = np.random.normal(4.18, 0.74, 75)
# 乙组治疗后1年AMH: 正态分布, 均值3.47 ng/ml, 标准差0.75 ng/ml, 75个样本
amh_after_group_b = np.random.normal(3.47, 0.75, 75)
# 合并甲组和乙组的治疗后1年AMH数据
amh_after = np.concatenate((amh_after_group_a, amh_after_group_b))

# 窦卵泡计数 (AFC)
# 甲组治疗前AFC: 正态分布, 均值7.56个, 标准差1.25个, 75个样本
afc_before_group_a = np.random.normal(7.56, 1.25, 75)
# 乙组治疗前AFC: 正态分布, 均值7.64个, 标准差1.22个, 75个样本
afc_before_group_b = np.random.normal(7.64, 1.22, 75)
```

```
# 合并甲组和乙组的治疗前AFC数据
afc_before = np.concatenate((afc_before_group_a, afc_before_group_b))

# 甲组治疗后1年AFC: 正态分布, 均值10.42个, 标准差1.18个, 75个样本
afc_after_group_a = np.random.normal(10.42, 1.18, 75)
# 乙组治疗后1年AFC: 正态分布, 均值9.54个, 标准差1.12个, 75个样本
afc_after_group_b = np.random.normal(9.54, 1.12, 75)
# 合并甲组和乙组的治疗后1年AFC数据
afc_after = np.concatenate((afc_after_group_a, afc_after_group_b))

# 吸烟
# 复发组吸烟: 无占比80.00%, 有占比20.00%
smoking_recurrence = np.random.choice([0, 1], size=35, p=[0.8000, 0.2000])
# 未复发组吸烟: 无占比80.87%, 有占比19.13%
smoking_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8087, 0.1913])
# 合并复发组和未复发组的吸烟数据
smoking = np.concatenate((smoking_recurrence, smoking_non_recurrence))

# 饮酒
# 复发组饮酒: 无占比88.57%, 有占比11.43%
drinking_recurrence = np.random.choice([0, 1], size=35, p=[0.8857, 0.1143])
# 未复发组饮酒: 无占比92.17%, 有占比7.83%
drinking_non_recurrence = np.random.choice([0, 1], size=115, p=[0.9217, 0.0783])
# 合并复发组和未复发组的饮酒数据
drinking = np.concatenate((drinking_recurrence, drinking_non_recurrence))

# 盆腔粘连
# 复发组盆腔粘连: 无占比34.29%, 有占比65.71%
pelvic_adhesion_recurrence = np.random.choice([0, 1], size=35, p=[0.3429, 0.6571])
# 未复发组盆腔粘连: 无占比25.22%, 有占比74.78%
pelvic_adhesion_non_recurrence = np.random.choice([0, 1], size=115, p=[0.2522, 0.7478])
# 合并复发组和未复发组的盆腔粘连数据
pelvic_adhesion = np.concatenate((pelvic_adhesion_recurrence, pelvic_adhesion_non_recurrence))

# 合并子宫腺肌病
# 复发组合并子宫腺肌病: 无占比82.86%, 有占比17.14%
adenomyosis_recurrence = np.random.choice([0, 1], size=35, p=[0.8286, 0.1714])
# 未复发组合并子宫腺肌病: 无占比86.09%, 有占比13.91%
adenomyosis_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8609, 0.1391])
# 合并复发组和未复发组的合并子宫腺肌病数据
adenomyosis = np.concatenate((adenomyosis_recurrence, adenomyosis_non_recurrence))

# 合并子宫肌瘤
```

```

# 复发组合并子宫肌瘤：无占比77.14%，有占比22.86%
myoma_recurrence = np.random.choice([0, 1], size=35, p=[0.7714, 0.2286])
# 未复发组合并子宫肌瘤：无占比81.74%，有占比18.26%
myoma_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8174, 0.1826])
# 合并复发组和未复发组的合并子宫肌瘤数据
myoma = np.concatenate((myoma_recurrence, myoma_non_recurrence))

# 既往宫腔操作史
# 复发组既往宫腔操作史：无占比80.00%，有占比20.00%
uterine_cavity_operation_recurrence = np.random.choice([0, 1], size=35, p=[0.8000, 0.2000])
# 未复发组既往宫腔操作史：无占比84.35%，有占比15.65%
uterine_cavity_operation_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8435, 0.1565])
# 合并复发组和未复发组的既往宫腔操作史数据
uterine_cavity_operation = np.concatenate((uterine_cavity_operation_recurrence, uterine_cavity_operation_non_recurrence))

# 既往子宫内膜异位症手术史
# 复发组既往子宫内膜异位症手术史：无占比88.57%，有占比11.43%
endometriosis_surgery_recurrence = np.random.choice([0, 1], size=35, p=[0.8857, 0.1143])
# 未复发组既往子宫内膜异位症手术史：无占比87.83%，有占比12.17%
endometriosis_surgery_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8783, 0.1217])
# 合并复发组和未复发组的既往子宫内膜异位症手术史数据
endometriosis_surgery = np.concatenate((endometriosis_surgery_recurrence, endometriosis_surgery_non_recurrence))

# 妊娠率
# 甲组妊娠率：未妊娠占比28.00%，妊娠占比72.00%
pregnancy_rate_group_a = np.random.choice([0, 1], size=75, p=[0.2800, 0.7200])
# 乙组妊娠率：未妊娠占比44.00%，妊娠占比56.00%
pregnancy_rate_group_b = np.random.choice([0, 1], size=75, p=[0.4400, 0.5600])
# 合并甲组和乙组的妊娠率数据
pregnancy_rate = np.concatenate((pregnancy_rate_group_a, pregnancy_rate_group_b))

# 复发率
# 甲组复发率：未复发占比85.33%，复发占比14.67%
recurrence_rate_group_a = np.random.choice([0, 1], size=75, p=[0.8533, 0.1467])
# 乙组复发率：未复发占比68.00%，复发占比32.00%
recurrence_rate_group_b = np.random.choice([0, 1], size=75, p=[0.6800, 0.3200])
# 合并甲组和乙组的复发率数据
recurrence_rate = np.concatenate((recurrence_rate_group_a, recurrence_rate_group_b))

# 总有效率
# 甲组总有效率：无效占比24.00%，有效占比76.00%
total_effectiveness_group_a = np.random.choice([0, 1], size=75, p=[0.2400, 0.7600])
# 乙组总有效率：无效占比40.00%，有效占比60.00%
total_effectiveness_group_b = np.random.choice([0, 1], size=75, p=[0.4000, 0.6000])

```

```

total_effectiveness_group_b = np.random.choice([0, 1], size=75, p=[0.4000, 0.6000])
# 合并甲组和乙组的总有效率数据
total_effectiveness = np.concatenate((total_effectiveness_group_a, total_effectiveness_group_b))

# 创建DataFrame, 将所有生成的数据组合成一个表格
data = pd.DataFrame({
    'PatientID': patient_id,
    'Age': age,
    'Duration': duration,
    'BMI': bmi,
    'MenarcheAge': menarche_age,
    'PeriodLength': period_length,
    'PreoperativeDysmenorrhea': dysmenorrhea,
    'PreoperativePregnancyTimes': pregnancy_times,
    'CystDiameter': cyst_diameter,
    'DeepEndometriosisFocus': deep_endometriosis,
    'PosteriorFornixTenderness': posterior_fornix_tenderness,
    'ASRMStage': asrm_stage,
    'PelvicLesionLocation': pelvic_lesion_location,
    'PreoperativeCA125': ca125,
    'E2Before': e2_before,
    'E2After': e2_after,
    'AMHBefore': amh_before,
    'AMHAfter': amh_after,
    'AFCBefore': afc_before,
    'AFCAfter': afc_after,
    'Smoking': smoking,
    'Drinking': drinking,
    'PelvicAdhesion': pelvic_adhesion,
    'Adenomyosis': adenomyosis,
    'Myoma': myoma,
    'UterineCavityOperation': uterine_cavity_operation,
    'EndometriosisSurgery': endometriosis_surgery,
    'PregnancyRate': pregnancy_rate,
    'RecurrenceRate': recurrence_rate,
    'TotalEffectiveness': total_effectiveness
})

# 保存为CSV文件
data.to_csv('simulated_data.csv', index=False)

# 打印提示信息
print("模拟数据已生成并保存为 'simulated_data.csv'")

```

数据说明：

- PatientID：患者编号，从1到150。
- Age：年龄，根据均值和标准差生成。
- Duration：病程，根据均值和标准差生成。
- BMI：体质量指数，根据均值和标准差生成。
- MenarcheAge：初潮年龄，根据均值和标准差生成。
- PeriodLength：月经时长，根据比例生成。
- PreoperativeDysmenorrhea：术前痛经史，0表示无，1表示有。
- PreoperativePregnancyTimes：术前孕次，0、1、 $\geq 2$ 。
- CystDiameter：囊肿最大直径，根据比例生成。
- DeepEndometriosisFocus：深部子宫内膜异位灶，0表示无，1表示有。
- PosteriorFornixTenderness：后穹窿触痛，0表示无，1表示有。
- ASRMStage：ASRM分期，0表示I<sub>II</sub>期，1表示III<sub>IV</sub>期。
- PelvicLesionLocation：盆腔病变位置，0表示单侧，1表示双侧。
- PreoperativeCA125：术前血清CA125，根据均值和标准差生成。
- E2Before：治疗前血清雌二醇（E2）。
- E2After：治疗后1年血清雌二醇（E2）。
- AMHBefore：治疗前抗苗勒管激素（AMH）。
- AMHAfter：治疗后1年抗苗勒管激素（AMH）。
- AFCBefore：治疗前窦卵泡计数（AFC）。
- AFCAfter：治疗后1年窦卵泡计数（AFC）。
- PregnancyRate：妊娠率，0表示未妊娠，1表示妊娠。
- RecurrenceRate：复发率，0表示未复发，1表示复发。
- TotalEffectiveness：总有效率，0表示无效，1表示有效。

# 分析思路

请根据上述数据，给出分析上述数据的思路

分析上述模拟数据的思路可以从以下几个方面展开，结合文献中的研究目的和统计分析方法，重点关注子宫内膜异位症患者腹腔镜术后复发风险及相关因素，以及不同促性腺激素释放激素激动剂（GnRHa）方案的应用效果。以下是一个详细的分析思路：

## 1. 数据预处理

### 1.1 数据清洗

检查缺失值：确认数据中是否存在缺失值，如果有，决定是否需要填补或删除。

检查异常值：检查数据中是否存在异常值，例如年龄、病程、BMI等是否在合理范围内。

数据类型检查：确认各列数据类型是否正确，如分类变量是否为整数或字符串，连续变量是否为数值类型。

## 1.2 数据转换

分类变量编码：将分类变量（如术前痛经史、盆腔病变位置等）转换为数值编码，便于后续分析。

标准化/归一化：对连续变量（如年龄、病程、BMI等）进行标准化或归一化处理，以便在模型中使用。

## 2. 描述性统计分析

### 2.1 基本统计量

- 计算均值、中位数、标准差、最小值、最大值：对连续变量（如年龄、病程、BMI、E2、AMH、AFC等）进行描述性统计。
- 计算频率和百分比：对分类变量（如术前痛经史、术前孕次、囊肿最大直径、深部子宫内膜异位灶等）进行频率统计。

### 2.2 数据可视化

- 绘制直方图：观察连续变量的分布情况。
- 绘制箱线图：检查连续变量的异常值。
- 绘制条形图：展示分类变量的频率分布。
- 绘制散点图：观察连续变量之间的关系。

## 3. 探索性数据分析

### 3.1 相关性分析

- 计算相关系数：分析连续变量之间的相关性，如E2、AMH、AFC等之间的相关性。
- 绘制热力图：可视化相关系数矩阵。

### 3.2 分组比较

- 复发组与未复发组的比较：比较复发组和未复发组在各变量上的差异，使用t检验或卡方检验。
- 不同GnRHa方案的比较：比较甲组（醋酸亮丙瑞林微球）和乙组（醋酸戈舍瑞林缓释植入剂）在卵巢功能指标（E2、AMH、AFC）、妊娠率、复发率等方面的差异。

## 4. 回归分析



## 4.1 单因素分析

- Logistic回归：以术后复发（0=未复发，1=复发）为因变量，分别将术前痛经史、术前孕次、囊肿最大直径、深部子宫内膜异位灶、后穹窿触痛、ASRM分期、盆腔病变位置等作为自变量，进行单因素Logistic回归分析。
- 线性回归：以卵巢功能指标（E2、AMH、AFC）为因变量，分别将术前痛经史、术前孕次、囊肿最大直径等作为自变量，进行线性回归分析。

## 4.2 多因素分析

- 多因素Logistic回归：将单因素分析中显著的变量纳入多因素Logistic回归模型，分析术后复发的独立危险因素。
- 多因素线性回归：将单因素分析中显著的变量纳入多因素线性回归模型，分析影响卵巢功能指标的独立因素。

# 5. 模型评估

## 5.1 模型拟合优度

- Logistic回归模型：计算模型的拟合优度指标，如 $R^2$ 、AIC、BIC等。
- 线性回归模型：计算模型的拟合优度指标，如 $R^2$ 、调整 $R^2$ 、F统计量等。

## 5.2 模型诊断

- 残差分析：检查残差是否呈正态分布，是否存在异方差性。
- 影响点分析：检查是否存在影响模型结果的异常数据点。

# 6. 结果解释与讨论

## 6.1 结果解释

- 术后复发风险因素：根据多因素Logistic回归分析结果，解释哪些因素是术后复发的独立危险因素。
- 不同GnRHa方案的效果：根据线性回归和卡方检验结果，解释不同GnRHa方案在改善卵巢功能、提高妊娠率、降低复发率方面的效果差异。

## 6.2 讨论

- 与文献对比：将分析结果与文献中的结论进行对比，讨论一致性或差异性。
- 临床意义：讨论分析结果的临床意义，提出可能的解释和建议。

## 7. 报告撰写

### 7.1 报告结构

- 引言：介绍研究背景和目的。
- 方法：详细描述数据来源、数据处理、分析方法。
- 结果：展示分析结果，包括描述性统计、相关性分析、回归分析等。
- 讨论：解释结果，讨论其临床意义和可能的解释。
- 结论：总结研究的主要发现和建议。

通过上述分析思路，可以全面地分析和解释模拟数据，为临床决策提供科学依据。

## 分析代码

请根据上述数据，给出分析上述数据的代码

分析上述模拟数据的思路可以分为几个主要步骤，包括数据预处理、描述性统计分析、探索性数据分析、回归分析以及结果解释与讨论。以下是详细的分析思路：

### 1. 数据预处理

#### 1.1 数据加载

加载生成的CSV文件，检查数据的基本结构和完整性。

```
import pandas as pd

# 加载数据
# 从CSV文件中读取数据，并将其存储在DataFrame对象中
data = pd.read_csv('simulated_data.csv')

# 查看数据的前几行
# 打印数据的前五行，以便快速查看数据的结构和内容
print(data.head())

# 检查数据的基本信息
# 打印数据的基本信息，包括每列的数据类型、非空值数量等
print(data.info())

# 检查缺失值
# 计算并打印每列中缺失值的数量，以便识别数据中的缺失情况
print(data.isnull().sum())
```

## 1.2 数据类型检查

确保各列数据类型正确，分类变量应为category类型，连续变量应为float或int类型。

```
# 转换数据类型
# 定义需要转换为分类数据类型的列名列表
categorical_columns = [
    'PeriodLength', # 月经时长
    'PreoperativeDysmenorrhea', # 术前痛经史
    'PreoperativePregnancyTimes', # 术前孕次
    'CystDiameter', # 囊肿最大直径
    'DeepEndometriosisFocus', # 深部子宫内膜异位灶
    'PosteriorFornixTenderness', # 后穹窿触痛
    'ASRMStage', # ASRM分期
    'PelvicLesionLocation', # 盆腔病变位置
    'Smoking', # 吸烟
    'Drinking', # 饮酒
    'PelvicAdhesion', # 盆腔粘连
    'Adenomyosis', # 合并子宫腺肌病
    'Myoma', # 合并子宫肌瘤
    'UterineCavityOperation', # 既往宫腔操作史
    'EndometriosisSurgery', # 既往子宫内膜异位症手术史
    'PregnancyRate', # 妊娠率
    'RecurrenceRate', # 复发率
    'TotalEffectiveness' # 总有效率
]

# 遍历每个需要转换的列名
for col in categorical_columns:
    # 将指定列的数据类型转换为'category'，以便后续分析中更好地处理分类数据
    data[col] = data[col].astype('category')
```

## 2. 描述性统计分析

### 2.1 基本统计量

计算连续变量的均值、标准差、最小值、最大值等。

```
# 基本统计量
# 打印数据集中所有数值型列的基本统计信息，包括均值、标准差、最小值、最大值、四分位数等
print(data.describe())

# 分类变量的频率统计
# 遍历每个分类变量列
for col in categorical_columns:
    # 打印每个分类变量的频率统计信息，即每个类别的出现次数
    print(f"{col}:\n{data[col].value_counts()}\n")
```

## 2.2 数据可视化

绘制直方图、箱线图、条形图等，观察数据分布和异常值。

```
import matplotlib.pyplot as plt
import seaborn as sns

# 直方图
# 绘制数据集中所有数值型列的直方图，以观察其分布情况
data.hist(figsize=(12, 10))
# 显示直方图
plt.show()

# 箱线图
# 创建一个新的图形窗口，设置图形大小
plt.figure(figsize=(12, 8))
# 使用Seaborn绘制箱线图，展示指定数值型列的分布和异常值
sns.boxplot(data=data[['Age', 'Duration', 'BMI', 'MenarcheAge', 'PreoperativeCA125', 'E2Before']],
# 设置x轴标签的旋转角度，以便更好地显示
plt.xticks(rotation=45)
# 显示箱线图
plt.show()

# 条形图
# 遍历每个分类变量列
for col in categorical_columns:
    # 创建一个新的图形窗口，设置图形大小
    plt.figure(figsize=(6, 4))
    # 使用Seaborn绘制条形图，展示每个分类变量的频率分布
    sns.countplot(x=col, data=data)
    # 设置图形的标题为当前列名
    plt.title(col)
    # 显示条形图
    plt.show()
```

## 3. 探索性数据分析

### 3.1 相关性分析

计算连续变量之间的相关性，并绘制热力图。

```
# 相关性分析
# 计算数据集中所有数值型列之间的相关系数矩阵
correlation_matrix = data.corr()
# 打印相关系数矩阵，以便查看各变量之间的相关性
print(correlation_matrix)

# 热力图
# 创建一个新的图形窗口，设置图形大小
plt.figure(figsize=(12, 10))
# 使用Seaborn绘制热力图，展示相关系数矩阵
# annot=True表示在每个单元格中显示相关系数的数值
# cmap='coolwarm'设置热力图的颜色映射
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
# 显示热力图
plt.show()
```

### 3.2 分组比较

比较复发组和未复发组在各变量上的差异，使用t检验或卡方检验。

```

from scipy import stats

# 分组
# 将数据按复发率分为复发组和未复发组
recurrence_group = data[data['RecurrenceRate'] == 1]
non_recurrence_group = data[data['RecurrenceRate'] == 0]

# t检验
# 定义需要进行t检验的连续变量列名列表
continuous_columns = ['Age', 'Duration', 'BMI', 'MenarcheAge', 'PreoperativeCA125', 'E2Before',
# 遍历每个连续变量列
for col in continuous_columns:
    # 对复发组和未复发组的当前列进行独立样本t检验
    t_stat, p_value = stats.ttest_ind(recurrence_group[col], non_recurrence_group[col])
    # 打印t检验的统计量和p值
    print(f"{col}: t-statistic={t_stat}, p-value={p_value}")

# 卡方检验
# 遍历每个分类变量列, 排除PregnancyRate, RecurrenceRate, TotalEffectiveness
for col in categorical_columns[:-3]:
    # 对当前分类变量与复发率进行卡方检验
    chi2_stat, p_value, dof, expected = stats.chi2_contingency(pd.crosstab(data[col], data['Recu
    # 打印卡方检验的统计量和p值
    print(f"{col}: chi2-statistic={chi2_stat}, p-value={p_value}")

```

## 4. 回归分析

### 4.1 单因素Logistic回归

以术后复发（0=未复发，1=复发）为因变量，分别将术前痛经史、术前孕次、囊肿最大直径等作为自变量，进行单因素Logistic回归分析。

```

import statsmodels.api as sm
from statsmodels.formula.api import logit

# 确保 RecurrenceRate 是数值型
# 将 'RecurrenceRate' 列的数据类型转换为整数型，以便进行回归分析
data['RecurrenceRate'] = data['RecurrenceRate'].astype(int)

# 单因素Logistic回归
# 遍历数据集中除最后三个列（PregnancyRate, RecurrenceRate, TotalEffectiveness）以外的所有列
for col in data.columns[:-3]:
    # 跳过 'PatientID' 列，因为它不是一个有意义的自变量
    if col != 'PatientID':
        # 跳过分类变量，因为它们需要特殊处理
        if data[col].dtype == 'category':
            continue
        # 使用Logistic回归分析当前列与复发率之间的关系
        model = logit(f"RecurrenceRate ~ {col}", data=data).fit()
        # 打印回归模型的摘要信息，包括系数、标准误差、p值等
        print(f"{col}:\n{model.summary()}\n")

```

## 代码说明

数据类型转换：

将 RecurrenceRate 转换为数值型（int）。

跳过分类变量（category 类型），因为这些变量不能直接用于 logit 函数。

循环逻辑：

遍历所有列，跳过 PatientID 和分类变量。

对每个连续变量进行单因素Logistic回归分析。

示例输出

运行上述代码后，你将得到每个连续变量的单因素Logistic回归分析结果。例如：



Age:  
Optimization terminated successfully.  
Current function value: 0.554545  
Iterations 7

Logit Regression Results						
=====						
Dep. Variable:	RecurrenceRate		No. Observations:		150	
Model:	Logit		Df Residuals:		148	
Method:	MLE		Df Model:		1	
Date:	Mon, 05 Feb 2024		Pseudo R-squ.:		0.0633	
Time:	14:30:00		Log-Likelihood:		-83.182	
converged:	True		LL-Null:		-88.725	
Covariance Type:	nonrobust		LLR p-value:		0.001462	
=====						
	coef	std err	z	P> z	[0.025	0.975]
-----						
Intercept	-2.1973	0.646	-3.400	0.001	-3.467	-0.928
Age	0.0657	0.027	2.432	0.015	0.013	0.118
=====						

注意事项  
如果需要对分类变量进行Logistic回归分析，可以使用 pandas.get\_dummies 将分类变量转换为虚拟变量（dummy variables）。

确保数据中没有缺失值，否则可能会影响回归分析的结果。

希望这个修正后的代码能够解决你的问题！

4.2 多因素Logistic回归

将单因素分析中显著的变量纳入多因素Logistic回归模型，分析术后复发的独立危险因素。

```
# 多因素Logistic回归
significant_vars = ['PreoperativeDysmenorrhea', 'PreoperativePregnancyTimes', 'CystDiameter', 'I
formula = f"RecurrenceRate ~ {' + '.join(significant_vars)}"
model = logit(formula, data=data).fit()
print(model.summary())
```

5. 模型评估

5.1 模型拟合优度

计算模型的拟合优度指标，如AIC、BIC等。

```
# Logistic回归模型拟合优度
print(f"AIC: {model.aic}")
print(f"BIC: {model.bic}")
```

## 5.2 残差分析

检查残差是否呈正态分布，是否存在异方差性。

```
import statsmodels.api as sm
from statsmodels.formula.api import logit
import seaborn as sns
import matplotlib.pyplot as plt

# 假设你已经拟合了一个Logit模型
# 例如：
# model = logit("RecurrenceRate ~ Age + Duration + BMI", data=data).fit()

# 获取残差
resid_dev = model.resid_dev # 偏差残差
resid_response = model.resid_response # 响应残差

# 残差直方图
sns.histplot(resid_dev, kde=True)
plt.title("Deviance Residuals Histogram")
plt.show()

# 影响点分析
influence = model.get_influence()
cooks_d = influence.cooks_distance[0]
sns.scatterplot(x=model.fittedvalues, y=resid_dev, size=cooks_d, sizes=(20, 200), alpha=0.7)
plt.title("Deviance Residuals vs Fitted Values with Cook's Distance")
plt.xlabel("Fitted Values")
plt.ylabel("Deviance Residuals")
plt.show()
```

### 代码说明

偏差残差 (Deviance Residuals) :

`model.resid_dev` 提供了偏差残差，这是逻辑回归中常用的残差类型。

偏差残差定义为： $DevianceResidual = sign(y_i - \hat{y}_i) - 2\log(\hat{y}_i y_i (1 - \hat{y}_i) (1 - y_i))$ 。

响应残差 (Response Residuals) :

`model.resid_response` 提供了响应残差，这是实际值与预测值的差。

响应残差定义为： $ResponseResidual = y_i - \hat{y}^i$ 。

Cook's Distance：

`influence.cooks_distance[0]` 提供了Cook's Distance，用于识别影响点。

### 示例输出

运行上述代码后，你将得到以下两个图表：

偏差残差直方图：展示偏差残差的分布情况。

偏差残差与拟合值的散点图：展示偏差残差与拟合值的关系，并用Cook's Distance标记影响点。

### 注意事项

如果你使用的是其他类型的回归模型（如线性回归），可以直接使用 `model.resid` 获取残差。

对于逻辑回归，建议使用偏差残差或响应残差进行分析，因为这些残差类型更适合逻辑回归模型。

## 6. 结果解释与讨论

### 6.1 结果解释

根据模型输出的结果，解释哪些变量是术后复发的独立危险因素，以及不同GnRHa方案在改善卵巢功能、提高妊娠率、降低复发率方面的效果差异。

### 6.2 讨论

将分析结果与文献中的结论进行对比，讨论一致性或差异性，并提出可能的解释和建议。

## 7. 报告撰写

### 7.1 报告结构

- 引言：介绍研究背景和目的。
- 方法：详细描述数据来源、数据处理、分析方法。
- 结果：展示分析结果，包括描述性统计、相关性分析、回归分析等。
- 讨论：解释结果，讨论其临床意义和可能的解释。
- 结论：总结研究的主要发现和建议。

通过上述分析思路，可以全面地分析和解释模拟数据，为临床决策提供科学依据。

# 模拟复发时间数据

在前文的模拟数据的代码上继续修改：

上述是生成一份模拟数据的代码，现在请您基于上述代码进行修改。

RecurrenceRate：复发率，0表示未复发，1表示复发。

RecurrenceRate为0，则为未复发组，RecurrenceRate为1，则为复发组。

我想要增加一列新变量，这个变量的具体名称你来取名字，这个新变量的具体含义是记录患者手术之后复发的时间，针对于复发组的患者，有一个具体的复发时间，而针对于未复发组的患者，他们没有复发时间，直接将随访时间填入这些未复发患者组，随访时间为3年。要求，复发组和未复发组的生存时间要有统计学差异，此外，甲组和乙组的复发时间最好也要有统计学差异，但是差异要小于复发组和未复发组的生存时间的统计学差异

我之所以想这样生成数据，是为了接下来使用Cox风险回归模型，请你根据我的上述要求，基于上述代码写一份python代码，要求代码完整

特别要注意，复发组的复发时间为一个具体的数值，而非复发组为的复发时间为随访时间，这两者一定要对应，千万不要随机对应

```

import numpy as np
import pandas as pd
from scipy import stats

# 设置随机种子以确保结果可复现
np.random.seed(42)

# 患者数量
n = 150

# 定义一个函数来生成具有统计学差异的生存时间
def generate_survival_time(mean, std, n_samples):
    return np.random.normal(mean, std, n_samples)

# 使用辅助函数调整概率
def normalize_probabilities(probs):
    """将概率列表标准化以确保其总和为1。"""
    total = sum(probs)
    return [p / total for p in probs]

# 复发组和未复发组的生存时间参数
recurrence_mean = 1.5 # 复发组平均生存时间为1.5年
recurrence_std = 0.5 # 复发组生存时间标准差为0.5年
non_recurrence_mean = 3.0 # 未复发组生存时间为3年
non_recurrence_std = 0.0 # 未复发组生存时间标准差为0年

# 生成复发组和未复发组的生存时间
recurrence_time = generate_survival_time('recurrence', recurrence_mean, recurrence_std, 0.8)
non_recurrence_time = generate_survival_time('non_recurrence', non_recurrence_mean, non_recurrence_std, 0.2)

# 患者ID, 从1到150
patient_id = np.arange(1, n + 1)

# 年龄
age_group_a = np.random.normal(31.64, 3.67, 75)
age_group_b = np.random.normal(30.52, 3.36, 75)
age = np.concatenate((age_group_a, age_group_b))

# 病程
duration_group_a = np.random.normal(2.46, 0.58, 75)
duration_group_b = np.random.normal(2.52, 0.55, 75)
duration = np.concatenate((duration_group_a, duration_group_b))

```

```

# 体质指数 (BMI)
bmi_recurrence = np.random.normal(20.37, 1.43, 35)
bmi_non_recurrence = np.random.normal(20.24, 1.39, 115)
bmi = np.concatenate((bmi_recurrence, bmi_non_recurrence))

# 初潮年龄
menarche_age_recurrence = np.random.normal(13.62, 0.75, 35)
menarche_age_non_recurrence = np.random.normal(13.76, 0.82, 115)
menarche_age = np.concatenate((menarche_age_recurrence, menarche_age_non_recurrence))

# 月经时长
period_length_recurrence = np.random.choice([0, 1], size=35, p=[0.5429, 0.4571])
period_length_non_recurrence = np.random.choice([0, 1], size=115, p=[0.3913, 0.6087])
period_length = np.concatenate((period_length_recurrence, period_length_non_recurrence))

# 术前痛经史
dysmenorrhea_recurrence = np.random.choice([0, 1], size=35, p=[0.4286, 0.5714])
dysmenorrhea_non_recurrence = np.random.choice([0, 1], size=115, p=[0.6870, 0.3130])
dysmenorrhea = np.concatenate((dysmenorrhea_recurrence, dysmenorrhea_non_recurrence))

# 术前孕次
p_recurrence = normalize_probabilities([0.3429, 0.4286, 0.2285])
p_non_recurrence = normalize_probabilities([0.6522, 0.2435, 0.1043])
pregnancy_times_recurrence = np.random.choice([0, 1, 2], size=35, p=p_recurrence)
pregnancy_times_non_recurrence = np.random.choice([0, 1, 2], size=115, p=p_non_recurrence)
pregnancy_times = np.concatenate((pregnancy_times_recurrence, pregnancy_times_non_recurrence))

# 囊肿最大直径
cyst_diameter_recurrence = np.random.choice([0, 1], size=35, p=[0.4000, 0.6000])
cyst_diameter_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7217, 0.2783])
cyst_diameter = np.concatenate((cyst_diameter_recurrence, cyst_diameter_non_recurrence))

# 深部子宫内膜异位灶
deep_endometriosis_recurrence = np.random.choice([0, 1], size=35, p=[0.4857, 0.5143])
deep_endometriosis_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7130, 0.2870])
deep_endometriosis = np.concatenate((deep_endometriosis_recurrence, deep_endometriosis_non_recurrence))

# 后穹窿触痛
posterior_fornix_tenderness_recurrence = np.random.choice([0, 1], size=35, p=[0.5429, 0.4571])
posterior_fornix_tenderness_non_recurrence = np.random.choice([0, 1], size=115, p=[0.7565, 0.2435])
posterior_fornix_tenderness = np.concatenate((posterior_fornix_tenderness_recurrence, posterior_fornix_tenderness_non_recurrence))

# ASRM分期

```

```

asrm_stage_recurrence = np.random.choice([0, 1], size=35, p=[0.3429, 0.6571])
asrm_stage_non_recurrence = np.random.choice([0, 1], size=115, p=[0.5652, 0.4348])
asrm_stage = np.concatenate((asrm_stage_recurrence, asrm_stage_non_recurrence))

# 盆腔病变位置
pelvic_lesion_location_recurrence = np.random.choice([0, 1], size=35, p=[0.4571, 0.5429])
pelvic_lesion_location_non_recurrence = np.random.choice([0, 1], size=115, p=[0.6783, 0.3217])
pelvic_lesion_location = np.concatenate((pelvic_lesion_location_recurrence, pelvic_lesion_locat:

# 术前血清CA125
ca125_recurrence = np.random.normal(45.67, 7.26, 35)
ca125_non_recurrence = np.random.normal(43.25, 6.21, 115)
ca125 = np.concatenate((ca125_recurrence, ca125_non_recurrence))

# 血清雌二醇 (E2)
e2_before_group_a = np.random.normal(67.52, 6.82, 75)
e2_before_group_b = np.random.normal(67.43, 6.59, 75)
e2_before = np.concatenate((e2_before_group_a, e2_before_group_b))

e2_after_group_a = np.random.normal(41.56, 5.15, 75)
e2_after_group_b = np.random.normal(49.57, 6.34, 75)
e2_after = np.concatenate((e2_after_group_a, e2_after_group_b))

# 抗苗勒管激素 (AMH)
amh_before_group_a = np.random.normal(2.33, 0.60, 75)
amh_before_group_b = np.random.normal(2.34, 0.61, 75)
amh_before = np.concatenate((amh_before_group_a, amh_before_group_b))

amh_after_group_a = np.random.normal(4.18, 0.74, 75)
amh_after_group_b = np.random.normal(3.47, 0.75, 75)
amh_after = np.concatenate((amh_after_group_a, amh_after_group_b))

# 窦卵泡计数 (AFC)
afc_before_group_a = np.random.normal(7.56, 1.25, 75)
afc_before_group_b = np.random.normal(7.64, 1.22, 75)
afc_before = np.concatenate((afc_before_group_a, afc_before_group_b))

afc_after_group_a = np.random.normal(10.42, 1.18, 75)
afc_after_group_b = np.random.normal(9.54, 1.12, 75)
afc_after = np.concatenate((afc_after_group_a, afc_after_group_b))

# 吸烟
smoking_recurrence = np.random.choice([0, 1], size=35, p=[0.8000, 0.2000])

```

```

smoking_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8087, 0.1913])
smoking = np.concatenate((smoking_recurrence, smoking_non_recurrence))

# 饮酒
drinking_recurrence = np.random.choice([0, 1], size=35, p=[0.8857, 0.1143])
drinking_non_recurrence = np.random.choice([0, 1], size=115, p=[0.9217, 0.0783])
drinking = np.concatenate((drinking_recurrence, drinking_non_recurrence))

# 盆腔粘连
pelvic_adhesion_recurrence = np.random.choice([0, 1], size=35, p=[0.3429, 0.6571])
pelvic_adhesion_non_recurrence = np.random.choice([0, 1], size=115, p=[0.2522, 0.7478])
pelvic_adhesion = np.concatenate((pelvic_adhesion_recurrence, pelvic_adhesion_non_recurrence))

# 合并子宫腺肌病
adenomyosis_recurrence = np.random.choice([0, 1], size=35, p=[0.8286, 0.1714])
adenomyosis_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8609, 0.1391])
adenomyosis = np.concatenate((adenomyosis_recurrence, adenomyosis_non_recurrence))

# 合并子宫肌瘤
myoma_recurrence = np.random.choice([0, 1], size=35, p=[0.7714, 0.2286])
myoma_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8174, 0.1826])
myoma = np.concatenate((myoma_recurrence, myoma_non_recurrence))

# 既往宫腔操作史
uterine_cavity_operation_recurrence = np.random.choice([0, 1], size=35, p=[0.8000, 0.2000])
uterine_cavity_operation_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8435, 0.1565])
uterine_cavity_operation = np.concatenate((uterine_cavity_operation_recurrence, uterine_cavity_operation_non_recurrence))

# 既往子宫内膜异位症手术史
endometriosis_surgery_recurrence = np.random.choice([0, 1], size=35, p=[0.8857, 0.1143])
endometriosis_surgery_non_recurrence = np.random.choice([0, 1], size=115, p=[0.8783, 0.1217])
endometriosis_surgery = np.concatenate((endometriosis_surgery_recurrence, endometriosis_surgery_non_recurrence))

# 妊娠率
pregnancy_rate_group_a = np.random.choice([0, 1], size=75, p=[0.2800, 0.7200])
pregnancy_rate_group_b = np.random.choice([0, 1], size=75, p=[0.4400, 0.5600])
pregnancy_rate = np.concatenate((pregnancy_rate_group_a, pregnancy_rate_group_b))

# # 复发率
# recurrence_rate_group_a = np.random.choice([0, 1], size=75, p=[0.8533, 0.1467])
# recurrence_rate_group_b = np.random.choice([0, 1], size=75, p=[0.6800, 0.3200])
# recurrence_rate = np.concatenate((recurrence_rate_group_a, recurrence_rate_group_b))

```



```

# 总有效率
total_effectiveness_group_a = np.random.choice([0, 1], size=75, p=[0.2400, 0.7600])
total_effectiveness_group_b = np.random.choice([0, 1], size=75, p=[0.4000, 0.6000])
total_effectiveness = np.concatenate((total_effectiveness_group_a, total_effectiveness_group_b))

# 复发率
recurrence_rate = np.random.choice([0, 1], size=n, p=[0.8, 0.2])

# 复发时间
recurrence_time = np.where(recurrence_rate == 1, generate_survival_time(1.5, 0.5, n), 3)

# # 生存时间
# survival_time = np.concatenate((recurrence_time, non_recurrence_time))

# 创建DataFrame, 将所有生成的数据组合成一个表格
data = pd.DataFrame({
    'PatientID': patient_id,
    'Age': age,
    'Duration': duration,
    'BMI': bmi,
    'MenarcheAge': menarche_age,
    'PeriodLength': period_length,
    'PreoperativeDysmenorrhea': dysmenorrhea,
    'PreoperativePregnancyTimes': pregnancy_times,
    'CystDiameter': cyst_diameter,
    'DeepEndometriosisFocus': deep_endometriosis,
    'PosteriorFornixTenderness': posterior_fornix_tenderness,
    'ASRMStage': asrm_stage,
    'PelvicLesionLocation': pelvic_lesion_location,
    'PreoperativeCA125': ca125,
    'E2Before': e2_before,
    'E2After': e2_after,
    'AMHBefore': amh_before,
    'AMHAfter': amh_after,
    'AFCBefore': afc_before,
    'AFCAfter': afc_after,
    'Smoking': smoking,
    'Drinking': drinking,
    'PelvicAdhesion': pelvic_adhesion,
    'Adenomyosis': adenomyosis,
    'Myoma': myoma,
    'UterineCavityOperation': uterine_cavity_operation,

```

```
'EndometriosisSurgery': endometriosis_surgery,
'PregnancyRate': pregnancy_rate,
'RecurrenceRate': recurrence_rate,
'TotalEffectiveness': total_effectiveness,
#     'SurvivalTime': survival_time # 新增列
'Recurrence_time': recurrence_time # 新增列
})

# 保存为CSV文件
data.to_csv('simulated_data2.csv', index=False)

# 打印提示信息
print("模拟数据已生成并保存为 'simulated_data2.csv'")
```