```r
knitr::opts_chunk$set(
    echo = TRUE,
    message = FALSE,
    warning = FALSE
)
options(tinytex.verbose = TRUE)

# 加载数据分析R包
library(tidyverse)

library(readxl) # 数据载入
# library(writexl)
# library(labelled)

library(ggpubr)
library(ggsci)
#library(patchwork)

library(tableone)
library(gtsummary)
#library(flextable)

library(survival)
# library(survminer)
# library(ggsurvfit)

# library(mice)
# library(MatchIt)
# library(Twang)

#library(broom)
# library(caret)
# library(pROC)

#library(devtools)

Sys.setlocale("LC_ALL", 'en_US.UTF-8')
```

代码生成要求

所有任务的共同上下文包括：研究背景和数据字典，
默认数据已经载入完成, 大模型可以看到当前数据的前几行样例(如str)
每个任务独立执行，相互间没有关联
代码生成时可以联网（或者给大模型提供R包的说明文档）

要求：

1. 代码正常运行，输出结果
2. 代码中变量名与数据字典一致
3. 统计分析方法与SAP要求一致（如果没有明确要求，可以自行发挥）

研究背景

本研究为高血压患者队列研究，2666名患者被分为试验组和对照组。入组时收集了人口学、血压、BMI、生活方式等特征。入组后随访患者的收缩压(SBP)和舒张压(DBP)。并且随访患者的心血管事件结局。本研究目标是研究不同干预方式能否影响患者的血压控制，以及降低心血管事件发生率。

数据载入

```
library(readxl)
meta <- readxl::read_xlsx("SPRINT_data/数据字典.xlsx")
baseline <- readxl::read_xlsx("SPRINT_data/baseline.xlsx")
BP <- readxl::read_xlsx("SPRINT_data/BP.xlsx")
events <- readxl::read_xlsx("SPRINT_data/events.xlsx")
stat_set <- readxl::read_xlsx("SPRINT_data/stat_set.xlsx")
```

描述性统计(探索)

Task 1-1

任务：计算两组患者各统计分析集人数
输出：表格

```
library(tidyverse)
stat_set %>%
  left_join(baseline) %>%
  mutate(arm = case_match(arm,
                          0~"对照组",
                          1~"试验组")) %>%
  group_by(arm) %>%
  summarise(`ITT集人数` = sum(ITT),
            `PP集人数` = sum(PP))
```

Task 1-2

任务：统计患者基线特征的缺失值数量
输出：表格

```
colSums(is.na(baseline))
```

Task 1-3

任务：统计患者基线特征的缺失值数量
输出：图片pdf
使用R包：VIM

```
library(VIM)
```

```
pdf("missing.pdf", width = 10, height = 6)
aggr_result <- aggr(baseline,
                    numbers = TRUE,    # 显示缺失值数量
                    prop = FALSE,      # 关闭比例显示（默认显示数量）
                    sortVars = TRUE,   # 按缺失数量排序变量
                    labels = names(baseline) # 显示变量名
)

dev.off()

aggr_result <- aggr(baseline,
                    numbers = TRUE,    # 显示缺失值数量
                    prop = FALSE,      # 关闭比例显示（默认显示数量）
                    sortVars = TRUE,   # 按缺失数量排序变量
                    labels = names(baseline) # 显示变量名
)
```

Task-1-4

任务：展示所有基线连续变量的直方图
变量：基线表中的连续变量
（数据字典是一个meta数据框,包括表名　表中文名 字段名　字段中文名　字段类型 值域字典。选取baseline表中字段类型为numeric的变量）
输出: 图片
使用R包：ggplot2,gridExtra

```
# 加载包
library(ggplot2)
library(gridExtra)

# 识别连续变量
# 从数据字典中筛选基线表的连续变量
continuous_vars <- meta %>%
  filter(表名=="baseline") %>%
  filter(字段类型=="numeric") %>%
  pull(字段名)

# 验证变量类型（双重保险）
continuous_vars <- continuous_vars[sapply(baseline[, continuous_vars],
is.numeric)]

plot_histogram <- function(var_name, data) {
  ggplot(data, aes(x = .data[[var_name]])) +
    geom_histogram(
      bins = 30,                    # 自动优化分组数量
      fill = "#4e79a7",             # 专业学术配色
      color = "white",               # 白色边界增强可读性
      alpha = 0.8                    # 适当透明度
    ) +
    labs(
      title = paste("Distribution of", var_name),
      x = var_name,
```

```
      y = "Frequency"
    ) +
    theme_minimal(base_size = 12) +  # 简洁主题
    theme(
      plot.title = element_text(face = "bold", hjust = 0.5),  # 标题居中加粗
      panel.grid.minor = element_blank()  # 清除次要网格线
    )
}

# 创建图形列表
plot_list <- lapply(continuous_vars, function(var) {
  plot_histogram(var, baseline)
})

# 智能布局（每行最多2图）
grid.arrange(
  grobs = plot_list,
  ncol = min(2, length(plot_list)),  # 自适应列数
  top = textGrob("Histograms of Baseline Continuous Variables",
                 gp = gpar(fontsize = 16, fontface = "bold"))
)
```

描述性统计(基线表)

Task 1-5

任务：统计两组患者的基线特征
变量：基线表中变量
方法：收缩压、舒张压作为非正态分布处理
输出：表格
使用R包：tableone

```
# 加载必要的包
library(tableone)
library(flextable)
library(officer)

# 定义变量
vars <- c("age", "sex", "SBP", "DBP", "BMI", "smoking")  # 需分析的变量
catVars <- c("sex", "smoking")           # 指定分类变量（需转为因子）
nonnormalVars <- c("SBP", "DBP")         # 指定非正态分布变量

# 转换变量类型
# baseline$arm <- as.factor(baseline$arm) # 分组变量转为因子
# baseline[catVars] <- lapply(baseline[catVars], as.factor) # 分类变量转为因子

# 创建基线表对象
table1 <- CreateTableOne(
  vars = vars,
  strata = "arm",           # 按治疗组分层
  data = baseline,
  factorVars = catVars,     # 分类变量列表
```

```
  includeNA = FALSE,        # 不单独显示缺失值
  test = TRUE
)

# 打印结果（显示所有分类水平）
print(table1,
      showAllLevels = TRUE,  # 显示分类变量的所有水平
      nonnormal = nonnormalVars)
```

Task 1-6

任务：统计两组患者的基线特征
人群：ITT
基线变量：年龄、性别、BMI、吸烟
输出：表格
使用R包：gtsummary

```
# 加载所需包
library(gtsummary)
library(dplyr)

# 筛选ITT患者
itt_baseline <- baseline %>%
  left_join(stat_set) %>%
  filter(ITT==1)

itt_baseline_table <- itt_baseline %>%
  select(arm, age, sex, BMI, smoking) %>%   # 选择基线变量
  mutate(arm = dplyr::case_match(arm,               # 分类变量编码
                           0~"对照组",
                           1~"试验组"))   %>%
  mutate(sex = case_match(sex,
                           0~"女性",
                           1~"男性")) %>%
  mutate(smoking = case_match(smoking,
                              0~"从未吸烟",
                              1~"曾经吸烟",
                              2~"已经戒烟")) %>%
  tbl_summary(
    by = arm,                                 # 按分组变量统计
    label = list(                               # 设置变量显示名称
      age ~ "年龄",
      sex ~ "性别",
      BMI ~ "BMI",
      smoking ~ "吸烟状态"
    ),
    statistic = list(                           # 自定义统计量格式
      all_continuous() ~ "{mean} ({sd})",    # 连续变量：均值±标准差
      all_categorical() ~ "{n} ({p}%)"       # 分类变量：频数（百分比）
    ),
    digits = list(                              # 设置小数位数
      all_continuous() ~ 1,                   # 连续变量保留1位小数
```

```
    all_categorical() ~ 0                    # 分类变量取整数
  ),
  missing = "no"                             # 不显示缺失值统计
) %>%
add_p() %>%                                  # 添加组间比较p值
add_overall() %>%                            # 添加总人群列
modify_header(                               # 修改表头
  label ~ "**变量**",
  p.value ~ "**P值**"
) %>%
modify_caption("**表1. ITT人群基线特征**")  # 添加表格标题

# 打印结果
itt_baseline_table
```

Task 1-7

任务：统计两组患者的基线特征
人群：PP
基线变量：年龄、性别、BMI、吸烟
方法：连续变量采用中位数、上下四分位数进行描述;展示缺失值
输出：表格
使用R包：gtsummary

```
# 加载所需包
library(gtsummary)
library(dplyr)

# 筛选PP患者
itt_baseline <- baseline %>%
  left_join(stat_set) %>%
  filter(PP==1)

pp_baseline_table <- itt_baseline %>%
  select(arm, age, sex, BMI, smoking) %>%   # 选择基线变量
  mutate(arm = case_match(arm,              # 分类变量编码
                          0~"对照组",
                          1~"试验组"))   %>%
  mutate(sex = case_match(sex,
                          0~"女性",
                          1~"男性")) %>%
  mutate(smoking = case_match(smoking,
                              0~"从未吸烟",
                              1~"曾经吸烟",
                              2~"已经戒烟")) %>%
  tbl_summary(
    by = arm,                               # 按分组变量统计
    label = list(                           # 设置变量显示名称
      age ~ "年龄",
      sex ~ "性别",
      BMI ~ "BMI",
      smoking ~ "吸烟状态"
    ),
```

```
    statistic = list(                          # 自定义统计量格式
      all_continuous() ~ "{median} ({p25}, {p75})",    # 连续变量：均值±标准差
      all_categorical() ~ "{n} ({p}%)"         # 分类变量：频数（百分比）
    ),
    digits = list(                             # 设置小数位数
      all_continuous() ~ 1,                   # 连续变量保留1位小数
      all_categorical() ~ 0                    # 分类变量取整数
    ),
    missing = "ifany"                              # 不显示缺失值统计
  ) %>%
  add_p() %>%                                  # 添加组间比较p值
  add_overall() %>%                            # 添加总人群列
  modify_header(                               # 修改表头
    label ~ "**变量**",
    p.value ~ "**P值**"
  ) %>%
  modify_caption("**表1. ITT人群基线特征**")  # 添加表格标题

# 打印结果
pp_baseline_table
```

Task 1-8

任务：统计两组患者的基线特征
人群：年龄>60岁人群
基线变量：年龄、性别、BMI>35、吸烟
方法：连续变量采用中位数、上下四分位数进行描述;展示缺失值
输出：表格
使用R包：gtsummary

```
# 加载所需包
library(gtsummary)
library(dplyr)

# 筛选PP患者
itt_baseline <- baseline %>%
  left_join(stat_set) %>%
  filter(age> 60)

pp_baseline_table <- itt_baseline %>%
  select(arm, age, sex, BMI, smoking) %>%   # 选择基线变量
  mutate(arm = case_match(arm,               # 分类变量编码
                          0~"对照组",
                          1~"试验组"))    %>%
  mutate(sex = case_match(sex,
                          0~"女性",
                          1~"男性")) %>%
  mutate(smoking = case_match(smoking,
                              0~"从未吸烟",
                              1~"曾经吸烟",
                              2~"已经戒烟")) %>%
  mutate(BMI = ifelse(BMI>35,">35","<=35")) %>%
  tbl_summary(
    by = arm,                                # 按分组变量统计
```

```
    label = list(                              # 设置变量显示名称
      age ~ "年龄",
      sex ~ "性别",
      BMI ~ "BMI",
      smoking ~ "吸烟状态"
    ),
    statistic = list(                          # 自定义统计量格式
      all_continuous() ~ "{median} ({p25}, {p75})",    # 连续变量：均值±标准差
      all_categorical() ~ "{n} ({p}%)"         # 分类变量：频数（百分比）
    ),
    digits = list(                             # 设置小数位数
      all_continuous() ~ 1,                   # 连续变量保留1位小数
      all_categorical() ~ 0                    # 分类变量取整数
    ),
    missing = "ifany"                              # 不显示缺失值统计
  ) %>%
  add_p() %>%                                  # 添加组间比较p值
  add_overall() %>%                            # 添加总人群列
  modify_header(                               # 修改表头
    label ~ "**变量**",
    p.value ~ "**P值**"
  ) %>%
  modify_caption("**表1. ITT人群基线特征**")  # 添加表格标题

# 打印结果
pp_baseline_table
```

纵向随访数据

Task 2-1

任务：绘制两组患者血压随时间变化图
变量：收缩压
输出：折线图, 横轴为随访次数，纵轴为收缩压, 展示均值、标准误
使用R包：ggplot2, ggpubr

```
library(ggplot2) # 绘图包
library(ggpubr)

follow_up <- BP %>%
  left_join(baseline %>% select(ID, arm))

ggline(                              # 类型
       data = follow_up,             # 数据
       group = "arm",                # 分组
       x = "VISIT", y = "SBP",       # 轴
       add = "mean_se")              # 内容
```

Task 2-2

任务：统计两组患者各血压随访时间点的人数
人群：PP
变量：人数
输出：表格

```
follow_up_n <- BP %>%
  left_join(baseline %>% select(ID, arm), by = "ID") %>%
  mutate(arm = case_match(arm,                # 分类变量编码
                          0~"对照组",
                          1~"试验组"))   %>%
  group_by(VISIT, arm) %>%
  summarize(
    n = n()  # 计算各访视点每组人数
  ) %>%
  ungroup() %>%
  pivot_wider(names_from = VISIT, values_from = n , names_prefix = "V")

follow_up_n
```

Task 2-3

任务：绘制两组患者相对于基线的改变量随时间变化图
变量：收缩压
输出：折线图, 横轴为随访次数，纵轴为收缩压相对于基线的改变量, 展示均值、标准误
使用R包：ggplot2, ggpubr

```
library(ggplot2) # 绘图包
library(ggpubr)

# 计算相对于基线的改变量
follow_up <- BP %>%
  left_join(baseline %>% select(ID, arm)) %>%
  group_by(ID) %>%
    arrange(VISIT,.by_group = TRUE) %>%
  mutate(SBP_change = SBP- first(SBP))

ggline(                              # 类型
       data = follow_up,             # 数据
       group = "arm",                # 分组
       x = "VISIT", y = "SBP_change",      # 轴
       add = "mean_se")              # 内容
```

血压结局比较

Task 3-1

任务：比较两组患者6个月、1年时，血压相对于基线的变化量。
人群：ITT
变量：收缩压、舒张压在6个月+/-14天，1年+/-30天，相对于基线的改变量。
方法：使用均值+-标准差描述各终点，使用t检验进行组间比较，计算组间差的mean difference 和95%置信区间
缺失值：不插补
输出：表格, 展示缺失值
使用R包：gtsummary

```
# 计算入组天数
x <- BP %>%
  left_join(baseline %>% select(ID, date_enroll), by = "ID") %>%
  mutate(days = day(days(date(date_visit)-date(date_enroll))))

BP_1y <- x %>% filter(days > (365-30) & days < (365+30) )%>%
  select(ID, SBP_1y = SBP, DBP_1y = DBP)

BP_6m <- x %>% filter(days> (182-14) & days< (182+14)) %>%
  select(ID, SBP_6m = SBP, DBP_6m = DBP)

# 计算收缩压、舒张压相对于基线的该变量
SBP_outcome <- baseline %>%
  select(ID, arm, SBP, DBP) %>%
  left_join(BP_1y) %>%
  left_join(BP_6m) %>%
  mutate(arm = case_match(arm,              # 分类变量编码
                          0~"对照组",
                          1~"试验组")) %>%
  mutate(SBP_change_1y = SBP_1y-SBP,
         DBP_change_1y = DBP_1y-DBP,
         SBP_change_6m = SBP_6m-SBP,
         DBP_change_6m = DBP_6m-DBP)

# ITT人群筛选
SBP_outcome_itt <- SBP_outcome %>%
  left_join(stat_set) %>%
  filter(ITT==1)

# 生成统计表格
# 计算组间差及置信区间
result_table <- SBP_outcome_itt %>%
  select(arm, SBP_change_6m, DBP_change_6m, SBP_change_1y, DBP_change_1y) %>%
  tbl_summary(
    by = arm,
    label = list(
      SBP_change_6m ~ "6个月收缩压变化(mmHg)",
      DBP_change_6m ~ "6个月舒张压变化(mmHg)",
      SBP_change_1y ~ "1年收缩压变化(mmHg)",
      DBP_change_1y ~ "1年舒张压变化(mmHg)"
    ),
```

```
    statistic = all_continuous() ~ "{mean} ± {sd}",
    digits = all_continuous() ~ 1,
    missing = "ifany"
  ) %>%
  add_difference()

# 打印结果
result_table
```

Task 3-2

任务：比较两组患者6个月、1年时，血压相对于基线的变化量。
人群：ITT
变量：收缩压、舒张压在6个月和1年时，相对于基线的改变量。
方法：使用均值+-标准差描述各终点，使用t检验进行组间比较，计算组间差的mean difference 和95%置信区间
缺失值处理: LOCF (对于6个月的终点，使用 182+14天前的最后一次观察值， 对于1年的终点，使用365+14天前的最后一次观察值)
输出：表格
使用R包：gtsummary

```
# 计算入组天数
x <- BP %>%
  left_join(baseline %>% select(ID, date_enroll), by = "ID") %>%
  mutate(days = day(days(date(date_visit)-date(date_enroll))))

BP_1y <- x %>% filter(days <= (365+14) )%>%
  group_by(ID) %>%
  slice_max(days, n=1, with_ties=FALSE) %>%
  select(ID, SBP_1y = SBP, DBP_1y = DBP)

BP_6m <- x %>% filter(days <=  (182+14)) %>%
    group_by(ID) %>%
  slice_max(days, n=1, with_ties=FALSE) %>%
  select(ID, SBP_6m = SBP, DBP_6m = DBP)

# 计算收缩压、舒张压相对于基线的该变量
SBP_outcome <- baseline %>%
  select(ID, arm, SBP, DBP) %>%
  left_join(BP_1y) %>%
  left_join(BP_6m) %>%
  mutate(arm = case_match(arm,              # 分类变量编码
                          0~"对照组",
                          1~"试验组")) %>%
  mutate(SBP_change_1y = SBP_1y-SBP,
         DBP_change_1y = DBP_1y-DBP,
         SBP_change_6m = SBP_6m-SBP,
         DBP_change_6m = DBP_6m-DBP)

# ITT人群筛选
SBP_outcome_itt <- SBP_outcome %>%
  left_join(stat_set) %>%
  filter(ITT==1)
```

```r
# 生成统计表格
# 计算组间差及置信区间
result_table <- SBP_outcome_itt %>%
  select(arm, SBP_change_6m, DBP_change_6m, SBP_change_1y, DBP_change_1y) %>%
  tbl_summary(
    by = arm,
    label = list(
      SBP_change_6m ~ "6个月收缩压变化(mmHg)",
      DBP_change_6m ~ "6个月舒张压变化(mmHg)",
      SBP_change_1y ~ "1年收缩压变化(mmHg)",
      DBP_change_1y ~ "1年舒张压变化(mmHg)"
    ),
    statistic = all_continuous() ~ "{mean} ± {sd}",
    digits = all_continuous() ~ 1,
    missing = "ifany"
  ) %>%
  add_difference()

# 打印结果
result_table
```

Task 3-3

任务：分析血压变化量的影响因素。
变量：1年时，收缩压相对于基线的改变量。
方法：多元线性回归，模型包含治疗组、年龄、性别、基线BMI、基线SBP
缺失值处理: LOCF (对于1年的终点，使用 365+14天前的最后一次观察值)
输出：多元线性回归表
使用R包：gtsummary

```r
# 计算入组天数
x <- BP %>%
  left_join(baseline %>% select(ID, date_enroll), by = "ID") %>%
  mutate(days = day(days(date(date_visit)-date(date_enroll))))

BP_1y <- x %>% filter(days <= (365+14) )%>%
  group_by(ID) %>%
  slice_max(days, n=1, with_ties=FALSE) %>%
  select(ID, SBP_1y = SBP)

# 计算收缩压、舒张压相对于基线的该变量
SBP_outcome <- baseline %>%
  select(ID, arm ,age, sex, BMI, SBP) %>%
  left_join(BP_1y) %>%
  mutate(sex = case_match(sex,
                          0~"女性",
                          1~"男性")) %>%
  mutate(arm = case_match(arm,                # 分类变量编码
                          0~"对照组",
                          1~"试验组")) %>%
  mutate(SBP_change_1y = SBP_1y-SBP)
```

```r
# 步骤3：多元线性回归建模
model <- lm(
  SBP_change_1y ~ arm + age + sex + BMI + SBP,
  data = SBP_outcome
)

# 用gtsummary输出专业结果
tbl_regression(
  model,
  label = list(
    arm ~ "治疗组",
    age ~ "年龄(岁)",
    sex ~ "性别",
    BMI ~ "基线BMI(kg/m²)",
    SBP ~ "基线收缩压(mmHg)")) %>%
  bold_p() %>%
  modify_caption("**表1：收缩压变化的多元线性回归分析**")
```

生存事件结局

Task 4-1

任务：derive数据。根据事件表，计算患者的生存结局。
events表中包含患者的各类事件0: 结束随访, 1:心血管死亡, 2:卒中, 3:心梗, 4:心衰, 5:非心源性死亡, 6:失访
根据这张表，计算患者的生存结局表survival_data, 结构如下：
包含字段
ID: 患者ID
days: 入组后天数
status: 生存结局 1:死亡, 0:删失

输出：survival_data.xlsx, 以及相应的数据字典。

```r
# 通过长表分析 编码复合终点
outcome <- events %>%
  left_join(baseline %>% select(ID, arm, date_enroll)) %>%
  mutate(days = day(days(date(date_event)-date(date_enroll)))) %>%
  mutate(event = dplyr::case_match(event,
                                   c(1,5)~1, # 把 心血管死亡/非心源性死亡编码为死亡。
                                   c(0,6)~0,    # 把结束随访、失访编码为删失
                                   .default = NA)) %>%
  filter(!is.na(event)) %>%  #去除无关事件
  group_by(ID) %>%
  arrange(days, .by_group = TRUE) %>%  # 按患者分组，按入组后时间排序
  summarise(days = first(days),        # 提取第一个发生的事件
            status = first(event))

head(outcome)
#writexl::write_xlsx(outcome,"survival_data.xlsx")
```

Task 4-2

任务: derive数据。根据事件表，计算患者的MACE事件
events表中包含患者的各类事件0: 结束随访, 1:心血管死亡, 2:卒中, 3:心梗, 4:心衰, 5:非心源性死亡, 6:失访
变量：MACE事件包括心血管死亡、心梗、卒中、心衰；删失事件包括结束随访、失访、非心源性死亡)
计算患者的MACE事件表 MACE_data, 结构如下：
包含字段
ID: 患者ID
days: 入组后天数
status: 生存结局 1:死亡, 0:删失
输出：MACE_data.xlsx, 以及相应的数据字典。

```
# 通过长表分析 编码复合终点
MACE <- events %>%
  left_join(baseline %>% select(ID, arm, date_enroll)) %>%
  mutate(days = day(days(date(date_event)-date(date_enroll)))) %>%
  mutate(event = dplyr::case_match(event,
                                   c(1,2,3,4)~1, # 把 MACE为死亡。
                                   c(0,5,6)~0,    # 把结束随访、非心源性死亡、失访编码为删失
                                   .default = NA)) %>%
  filter(!is.na(event)) %>%  #去除无关事件
  group_by(ID) %>%
  arrange(days, .by_group = TRUE) %>%  # 按患者分组，按入组后时间排序
  summarise(days = first(days),         # 提取第一个发生的事件
            status = first(event))

head(MACE)
#writexl::write_xlsx(MACE,"MACE_data.xlsx")
```

Task 4-3

任务: 绘制两组患者的生存曲线
人群：ITT
变量：总生存OS (终点事件包括心血管死亡、非心源性死亡；删失事件包括结束随访、失访)
方法：KM曲线
使用R包：survival, survminer

```
library(survival)
library(survminer)


# 通过长表分析 编码复合终点
outcome <- events %>%
  left_join(baseline %>% select(ID, arm, date_enroll)) %>%
  mutate(days = day(days(date(date_event)-date(date_enroll)))) %>%
  mutate(event = dplyr::case_match(event,
                                   c(1,5)~1, # 把 心血管死亡/非心源性死亡编码为死亡。
                                   c(0,6)~0,    # 把结束随访、失访编码为删失
                                   .default = NA)) %>%
```

```
  filter(!is.na(event)) %>%  #去除无关事件
  group_by(ID) %>%
  arrange(days, .by_group = TRUE) %>%  # 按患者分组，按入组后时间排序
  summarise(days = first(days),        # 提取第一个发生的事件
            status = first(event))

outcome2 <- outcome %>%
  left_join(stat_set) %>%
  filter(ITT==1) %>%
  left_join(baseline %>% select(ID,arm))

# 拟合生存曲线
sfit <- survfit( Surv(time=days, event=status) ~ arm,  # 曲线公式：  结局 ~ 分组 ，结局必须是一个Surv构造的生存变量 y ~ x1 + x2
                 data = outcome2 )     # 曲线数据

ggsurvplot(sfit)
```

Task 4-4

任务: 绘制两组患者的MACE事件累积风险曲线
人群：ITT
变量：MACE (终点事件包括心血管死亡、心梗、卒中、心衰；删失事件包括结束随访、失访、非心源性死亡)
方法: 绘制累积风险曲线, 使用lancet配色, 增加risk table, y轴范围为0-0.1
输出：pdf图
使用R包：survival, survminer

```
library(survival)
library(survminer)

# 通过长表分析 编码复合终点
MACE <- events %>%
  left_join(baseline %>% select(ID, arm, date_enroll)) %>%
  mutate(days = day(days(date(date_event)-date(date_enroll)))) %>%
  mutate(event = dplyr::case_match(event,
                                   c(1,2,3,4)~1, # 把 MACE为死亡。
                                   c(0,5,6)~0,   # 把结束随访、非心源性死亡、失访编码为删失
                                   .default = NA)) %>%
  filter(!is.na(event)) %>%  #去除无关事件
  group_by(ID) %>%
  arrange(days, .by_group = TRUE) %>%  # 按患者分组，按入组后时间排序
  summarise(days = first(days),        # 提取第一个发生的事件
            status = first(event))

# 筛选ITT人群，合并基线表需要的变量
MACE2 <- MACE %>%
  left_join(stat_set) %>%
  filter(ITT==1) %>%
  left_join(baseline %>% select(ID,arm))

# 拟合生存曲线
```

```
sfit <- survfit( Surv(time=days, event=status) ~ arm,  # 曲线公式：  结局 ~ 分组 ，结局必须是一个Surv构造的生存变量 y ~ x1 + x2
                 data = MACE2)    # 曲线数据

p2 <- ggsurvplot(sfit,
           palette="lancet",  # 杂志配色
           fun = function(x){1-x}, # 把结果转化为累积风险
           censor = FALSE,   # 是否对删失事件画+号
           # conf.int = TRUE,  # 是否曲线画置信区间
           pvalue.size=1, pval.coord = c(1,0.38), # p值显示大小，p值位置
           risk.table = TRUE, risk.table.height=0.2, # 是否显示risk table ，risk table的高度。
           # risk table是指：各时间点的两组各自 可能发生事件(处于risk中)的人数，不是两组各有多少人！
           break.x.by = 6,  xlim= c(0,30), ylim=c(0,0.1), # 坐标轴设置
           legend.title="",legend.labs=c("对照组","试验组"), # 图例设置
           tables.theme = theme_cleantable()) # risktable 格式

p2
ggsave("survival2.pdf")
```

Task 4-5

任务: 比较两组患者的MACE事件风险
人群：ITT
events表中包含患者的各类事件0: 结束随访, 1:心血管死亡, 2:卒中, 3:心梗, 4:心衰, 5:非心源性死亡, 6:失访
变量：MACE (终点事件包括心血管死亡、心梗、卒中、心衰；删失事件包括结束随访、失访、非心源性死亡)
方法: cox回归, 模型包括 治疗组、年龄、性别、基线SBP
输出：表格
使用R包：survival

```
# 加载必要包
library(survival)
library(dplyr)
library(tidyr)

# 通过长表分析 编码复合终点
MACE <- events %>%
  left_join(baseline %>% select(ID, arm, date_enroll)) %>%
  mutate(days = day(days(date(date_event)-date(date_enroll)))) %>%
  mutate(event = dplyr::case_match(event,
                                   c(1,2,3,4)~1, # 把 MACE为死亡。
                                   c(0,5,6)~0,    # 把结束随访、非心源性死亡、失访编码为删失
                                   .default = NA)) %>%
  filter(!is.na(event)) %>%  #去除无关事件
  group_by(ID) %>%
  arrange(days, .by_group = TRUE) %>%  # 按患者分组，按入组后时间排序
  summarise(days = first(days),         # 提取第一个发生的事件
            status = first(event))

# 筛选ITT人群，合并基线表需要的变量
```

```r
MACE2 <- MACE %>%
  left_join(stat_set) %>%
  filter(ITT==1) %>%
  left_join(baseline %>% select(ID,arm, age, sex, SBP))


# 构建Cox比例风险模型[1,4](@ref)
cox_model <- coxph(
  formula = Surv(days, status) ~ arm + age + sex + SBP,
  data = MACE2
)

# 模型结果整理[4,7](@ref)
results_table <- broom::tidy(cox_model, exponentiate = TRUE, conf.int = TRUE) %>%
  mutate(across(c(estimate, conf.low, conf.high), ~ round(., 2)),
         p.value = format.pval(p.value, digits = 2)) %>%
  select(term, estimate, conf.low, conf.high, p.value) %>%
  rename(
    Variable = term,
    HR = estimate,
    `95% CI Lower` = conf.low,
    `95% CI Upper` = conf.high,
    `P-value` = p.value
  )

# 输出结果表格
print(results_table)
```