talk09 练习与作业

目录

0.1 练习和作业说明

将相关代码填写入以 “‘{r} “‘ 标志的代码框中，运行并看到正确的结果；

完成后，用工具栏里的”Knit” 按键生成 PDF 文档；

将 PDF 文档改为：姓名-学号-`talk09` 作业`.pdf`，并提交到老师指定的平台/钉群。

0.2 talk09 内容回顾

- basic plot
- ggplot2

0.2.1 layered grammer (图层语法) 的成分

- 图层 (geom_xxx)
- scale (scale_xxx)
- faceting (facet_xxx)
- 坐标系统

0.2.2 图象类型

- 点图
- bars
- boxplots

0.2.3 其它重要内容（部分需要自学）

- colours
- theme
- 其它图像类型
- 图例（legends）和坐标轴
- 图形注释和其它定制

0.3 练习与作业：用户验证

请运行以下命令，验证你的用户名。

如你当前用户名不能体现你的真实姓名，请改为拼音后再运行本作业！

```
Sys.info()[["user"]]
```

```
## [1] "lucas"
```

```
Sys.getenv("HOME")
```

```
## [1] "/Users/lucas"
```

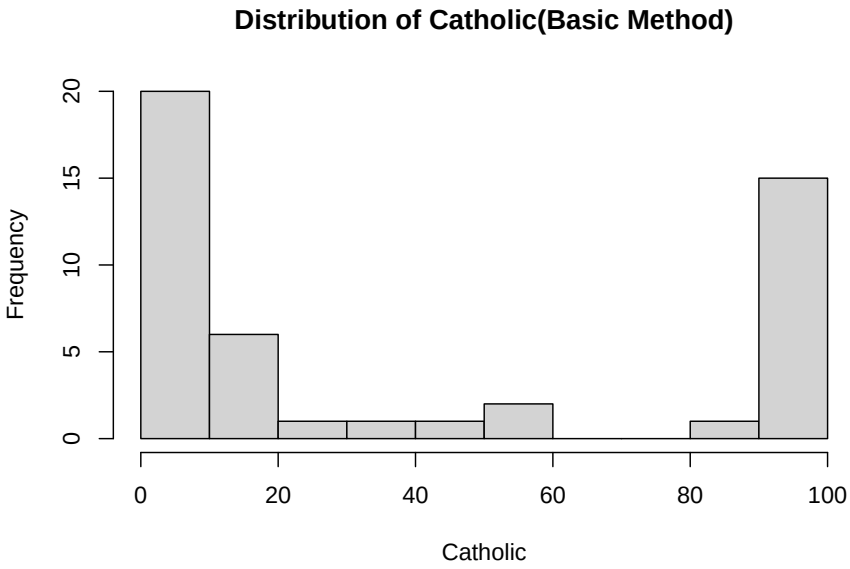
0.4 练习与作业 1：基础做图 & ggplot2

0.4.1 用 swiss 数据做图

1. 用直方图 histogram 显示 Catholic 列的分布情况；
2. 用散点图显示 Eduction 与 Fertility 的关系；将表示两者关系的线性公式、相关系数和 p 值画在图的空白处。

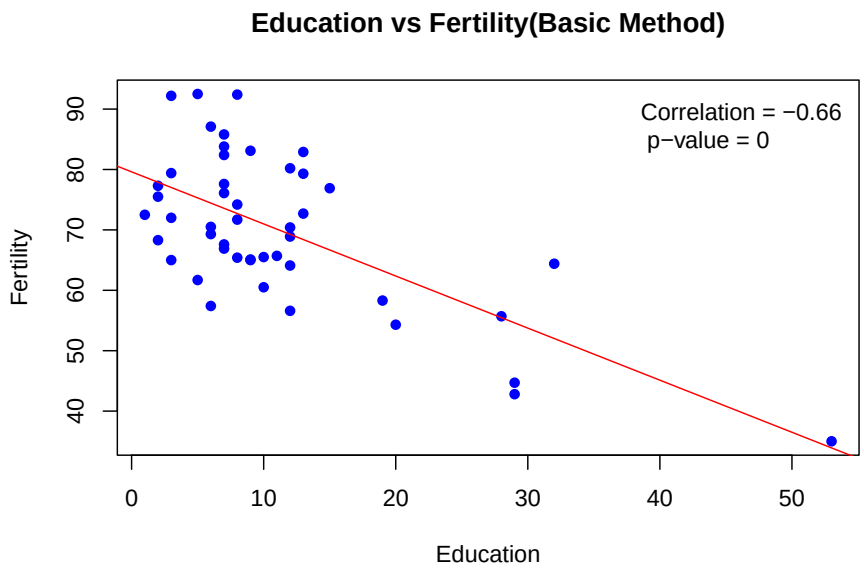
注：每种图提供**基础做图函数**和 **ggplot2** 两个版本！

```
## 代码写这里，并运行；
library(ggplot2)
data(swiss)

# Basic
# Task 01
  hist(
    swiss$Catholic,
    main="Distribution of Catholic(Basic Method)",
    xlab="Catholic")
```



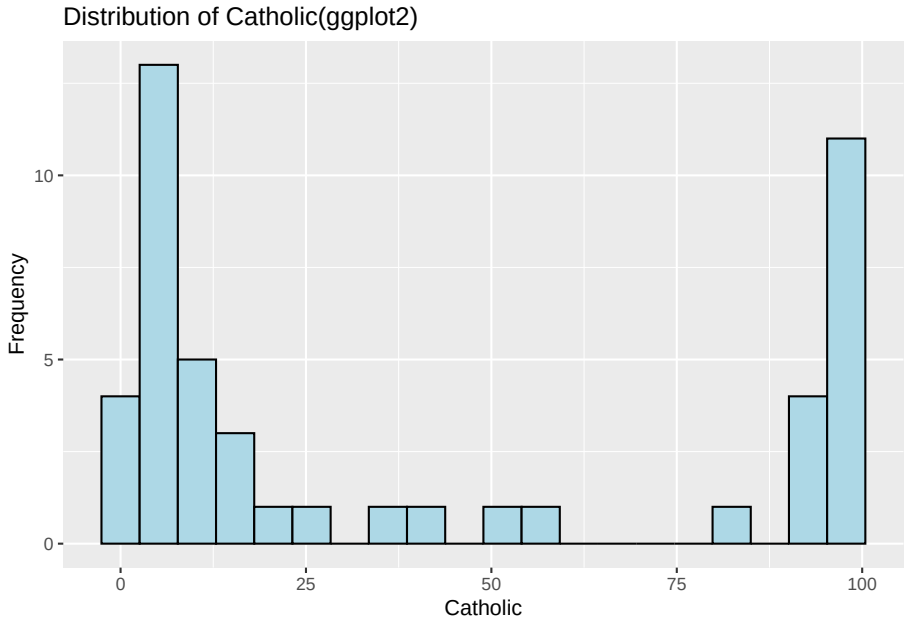

```r
# Task 02
# Plotting Scatter Plots
plot(
  swiss$Education,
  swiss$Fertility,
  main = "Education vs Fertility(Basic Method)",
  xlab = "Education",
  ylab = "Fertility",
  pch = 16,
  col = "blue")

# Adding a linear regression line
abline(
  lm(swiss$Fertility ~ swiss$Education),
  col = "red")

# Add correlation coefficients and p-values
cor_val =
```

```
  cor(swiss$Fertility,
      swiss$Education)
p_val =
  cor.test(swiss$Fertility,
           swiss$Education)$p.value
legend(
  "topright",
  legend = paste("Correlation =",
                 round(cor_val, 2),
                 "\n", "p-value =",
                 round(p_val, 4)),
  bty = "n",
  col = "red")
```




```
# ggplot2
# Task 01
ggplot(
  swiss,
```

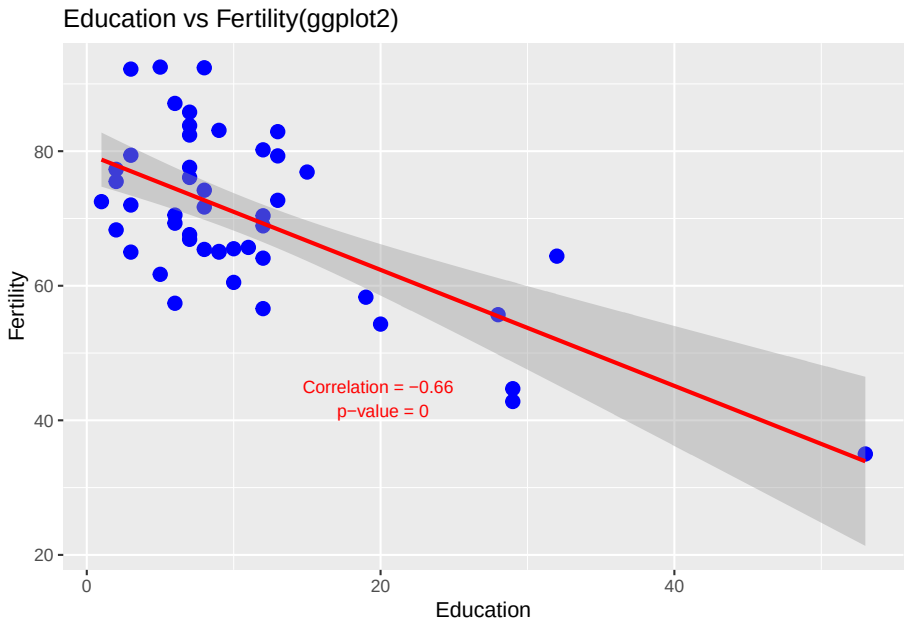
```
  aes(x = Catholic)) +
  geom_histogram(
    fill = "lightblue",
    color = "black",
    bins = 20) +
  labs(
    title = "Distribution of Catholic(ggplot2)",
    x = "Catholic",
    y = "Frequency")
```




```
# Task 02
# Plotting Scatterplots with ggplot2
ggplot(
  swiss,
  aes(x = Education,
      y = Fertility)) +
  geom_point(
    color = "blue",
```

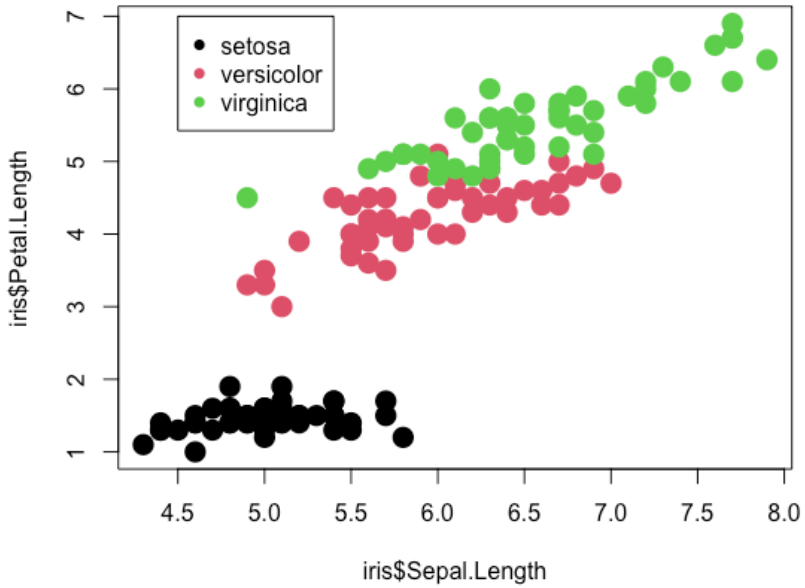
```
    size = 3) +
  geom_smooth(
    method = "lm",
    color = "red") +
  labs(
    title = "Education vs Fertility(ggplot2)",
    x = "Education",
    y = "Fertility") +
  annotate(
    "text",
    x = 20,
    y = 35,
    label =
      paste(
        "Correlation =",
        round(cor_val, 2),
        "\n",
        "p-value =",
        round(p_val, 4)),
    color = "red",
    vjust = -1,
    size = 3)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



0.4.2 用 `iris` 作图

1. 用散点图显示 `Sepal.Length` 和 `Petal.Length` 之间的关系；按 `species` 为散点确定颜色，并画出 legend 以显示 `species` 对应的颜色；



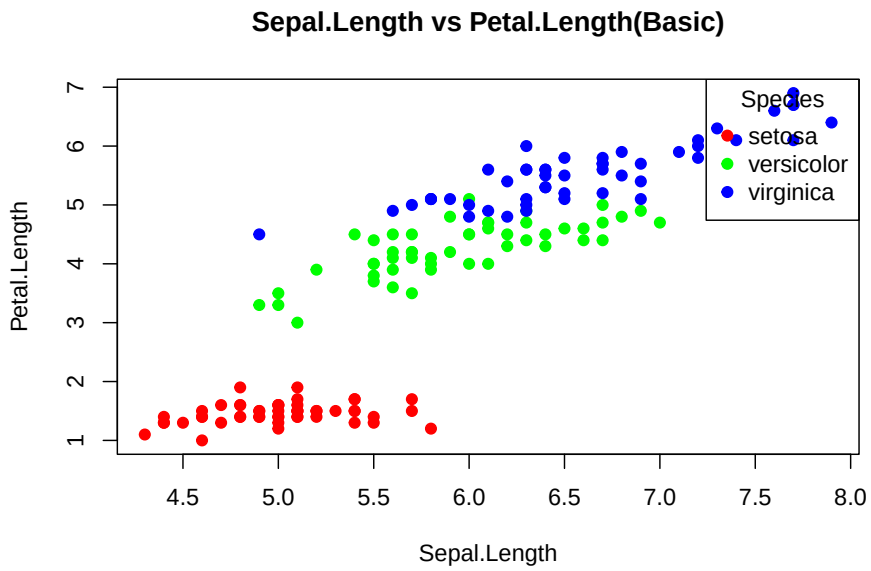
如下图所示：

2. 用 boxplot 显示 `species` 之间 `Sepal.Length` 的分布情况；

注：每种图提供**基础做图函数**和 **ggplot2** 两个版本！

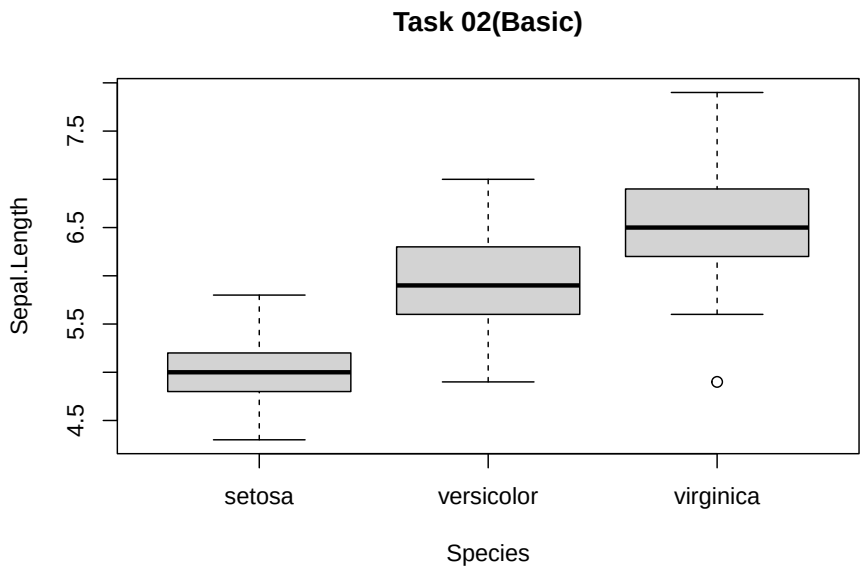
```
## 代码写这里，并运行；
data(iris)
library(ggplot2)

# Basic
# Task 01
colors =
  c("setosa" = "red",
    "versicolor" = "green",
    "virginica" = "blue")
plot(
  iris$Sepal.Length,
  iris$Petal.Length,
```

```
  pch = 19,
  col = colors[iris$Species],
  xlab = "Sepal.Length",
  ylab = "Petal.Length",
  main = "Sepal.Length vs Petal.Length(Basic)"
)
legend(
  "topright",
  legend = levels(iris$Species),
  col = colors,
  pch = 19,
  title = "Species"
)
```



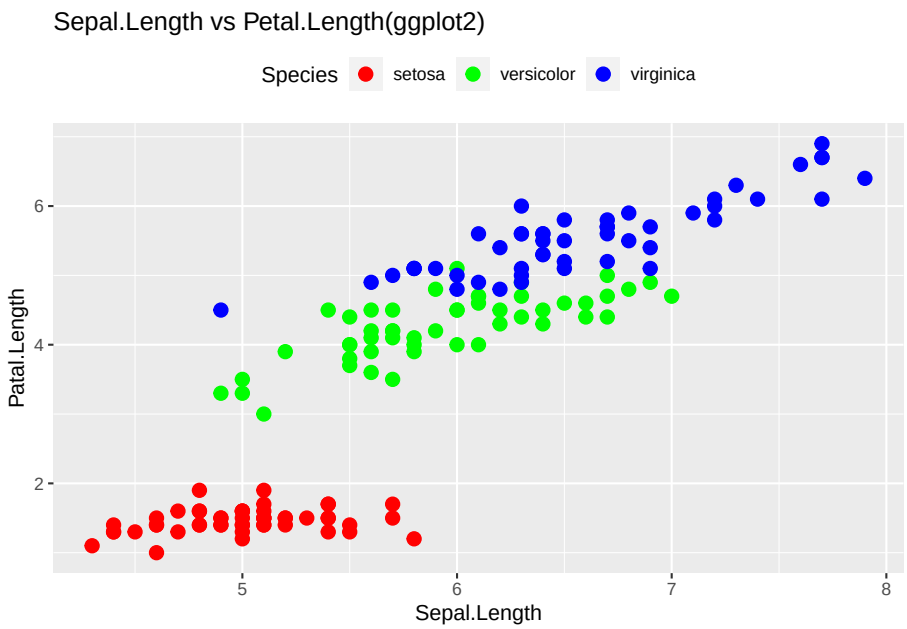

```
# Task 02
boxplot(
  Sepal.Length ~ Species,
  data = iris,
```

```
  xlab = "Species",
  ylab = "Sepal.Length",
  main = "Task 02(Basic)"
)
```




```
# ggplot
# Task 01
ggplot(
  iris,
  aes(
    x = Sepal.Length,
    y = Petal.Length,
    color = Species
  )
) +
  geom_point(size = 3) +
  labs(
    title =
```

```
    "Sepal.Length vs Petal.Length(ggplot2)",
  x = "Sepal.Length",
  y = "Patal.Length"
) +
scale_color_manual(
  values =
    c("setosa" = "red",
      "versicolor" = "green",
      "virginica" = "blue")
) +
theme(
  legend.position = "top"
)
```

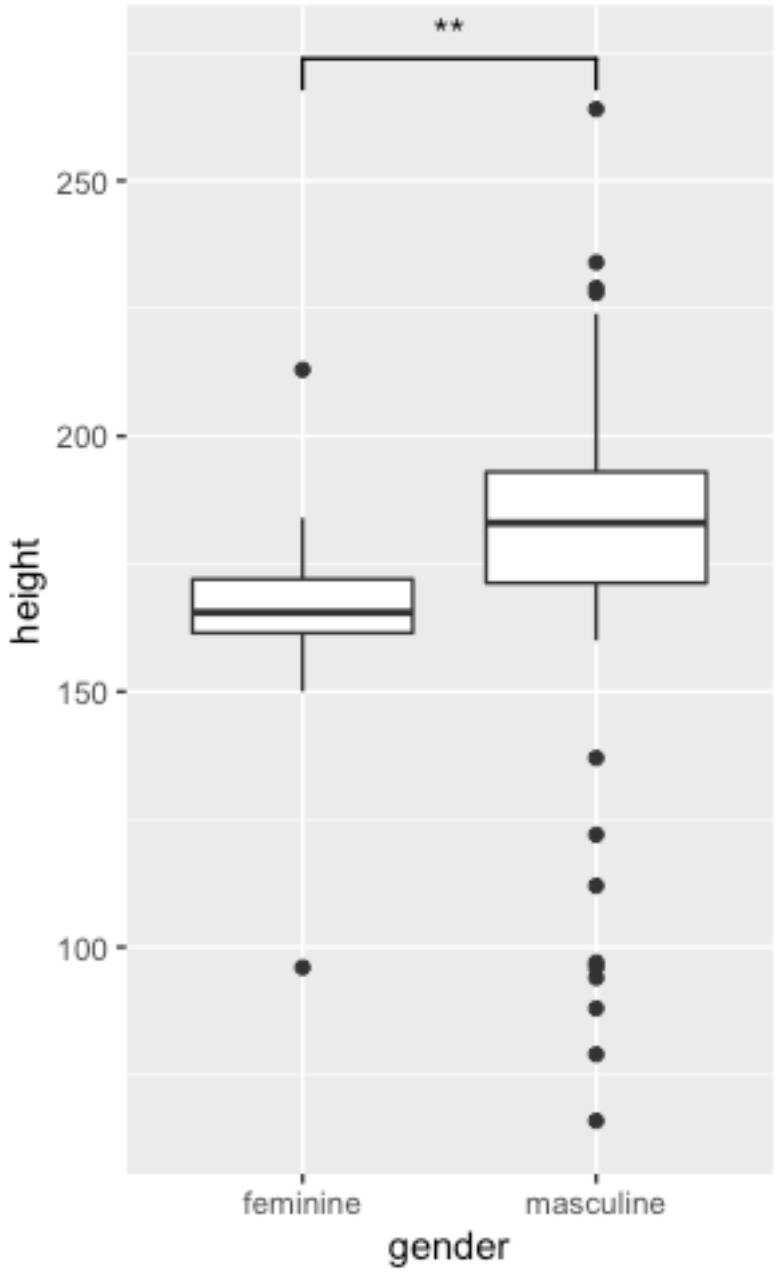


0.4.3 用 ggplot 作图：boxplot

用 `starwars` 的数据作图，画 `boxplot` 显示身高 `height` 与性别 `gender` 的关系。要求：

1. `height` 为 NA 的，不显示；
2. 用 `ggsignif` 包计算 `feminine` 和 `masculine` 两种性别的身高是否有显著区别，并在图上显示。
3. 将此图的结果保存为变量 `p1` ，以备后面使用；

最终结果如图所示：




```
## 代码写这里，并运行；
library(ggplot2)
```

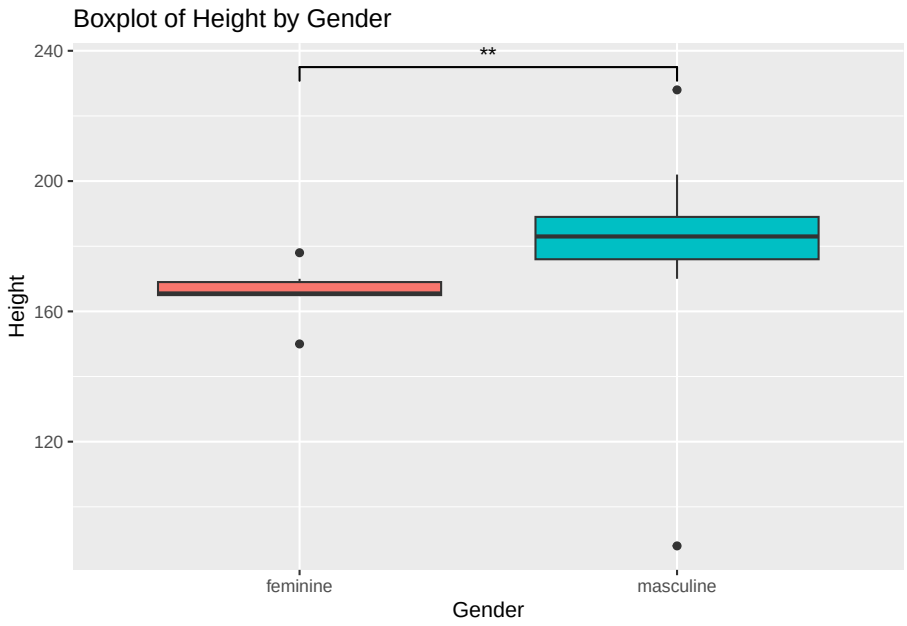
```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.9.2     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts
```

```r
library(ggsignif)

# Load the data
data("starwars")
starwars_filtered =
  na.omit(starwars)

# Initializing the plot
# Ignore 'NA' in "height"
p1 =
  ggplot(
    starwars_filtered,
    aes(
      x = gender,
      y = height,
      fill = gender
    )
  ) +
  geom_boxplot() +
  labs(
    title = "Boxplot of Height by Gender",
```

```
    x = "Gender",
    y = "Height"
  ) +
  theme(
    legend.position = "none"
  )

# Calculate height
p1_with_signif =
  p1 +
  geom_signif(
    comparisons =
      list(
        c("feminine", "masculine")
      ),
    map_signif_level = TRUE
  )

# Save it
p1 = p1_with_signif

print(p1)
```

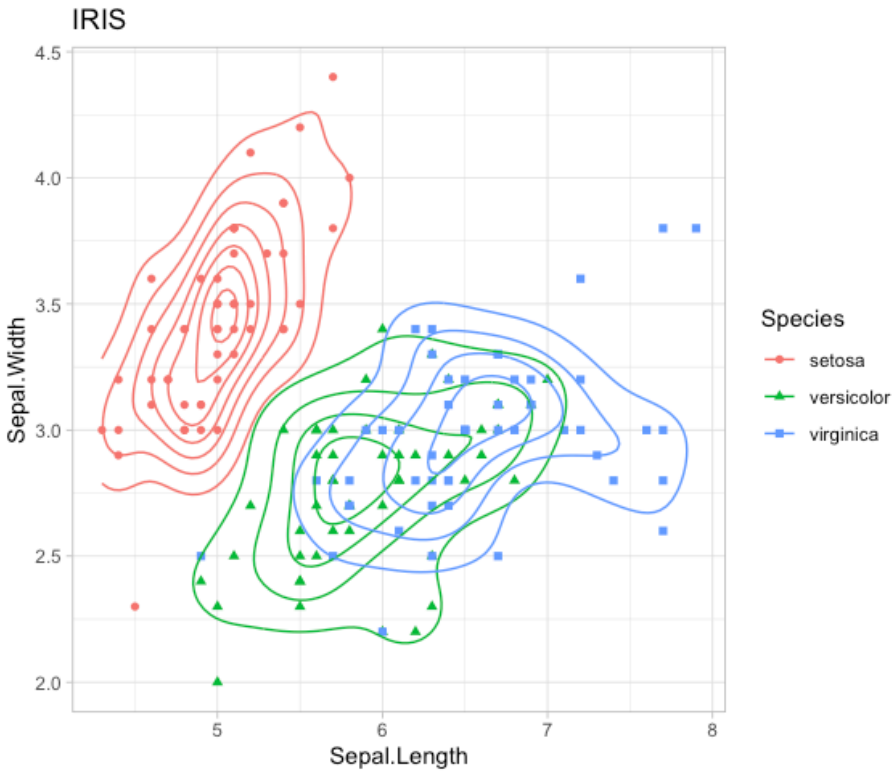
```
## Warning in wilcox.test.default(c(150, 165, 178, 170, 166, 165), c(172, 202, :
## cannot compute exact p-value with ties
```



0.4.4 用 ggplot 作图：使用 iris 做图

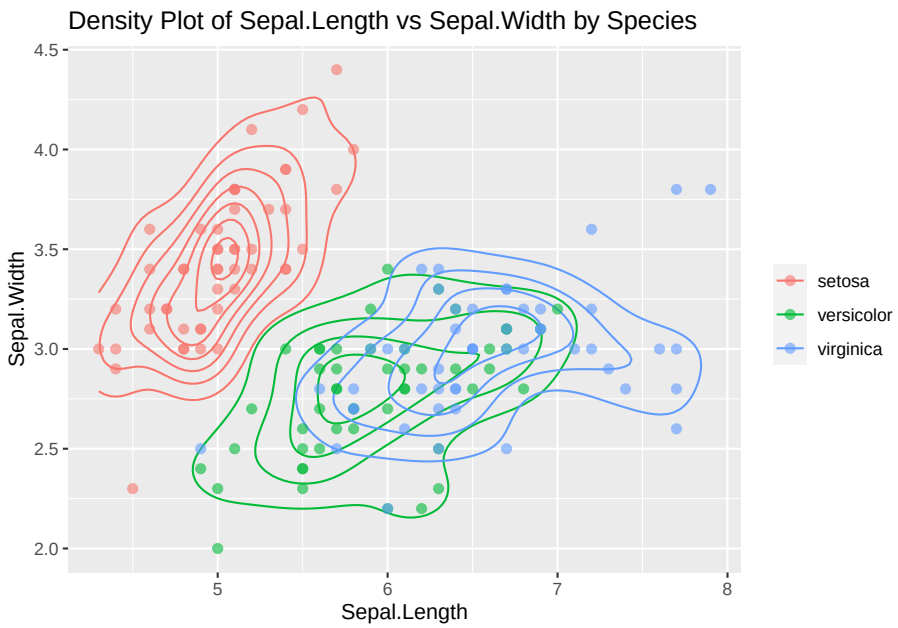
用 `geom_density2d` 显示 `Sepal.Length` 和 `Sepal.Width` 之间的关系，同时以 `Species` 为分组，结果如图所示：

将此图的结果保存为变量 **p2** ，以备后面使用；




```
## 代码写这里，并运行；
library(ggplot2)
library(tidyverse)

# Draw the plot
p2 =
  ggplot(
    iris,
    aes(
      x = Sepal.Length,
      y = Sepal.Width,
      color = Species
    )
  ) +
```

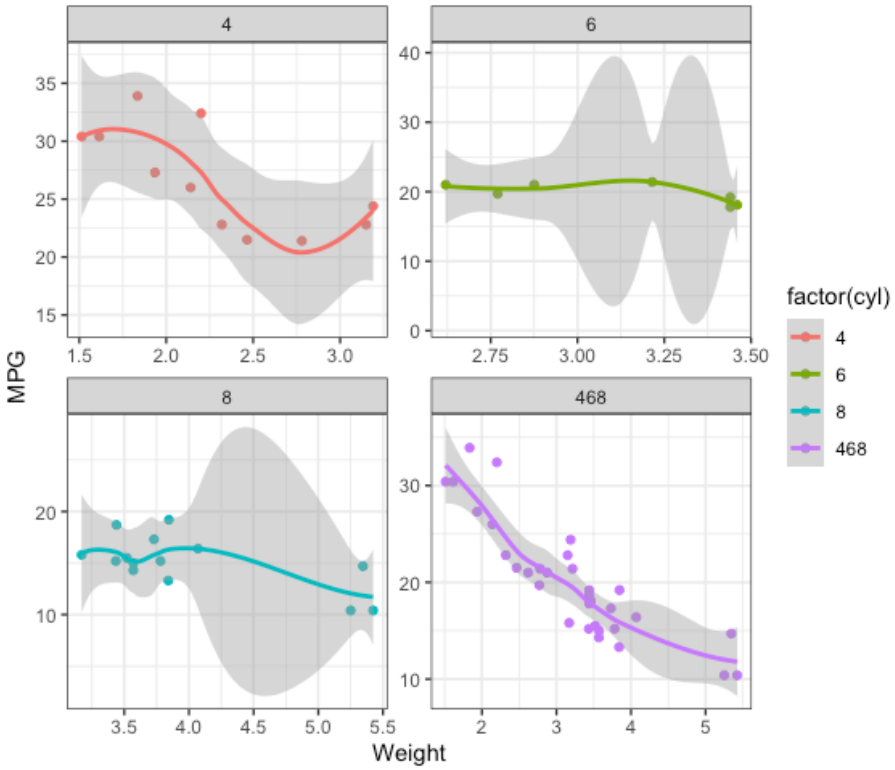
```
  geom_density2d() +
  geom_point(
    aes(color = Species),
    size = 2,
    alpha = 0.6) +
  labs(
    title = "Density Plot of Sepal.Length vs Sepal.Width by Species",
    x = "Sepal.Length",
    y = "Sepal.Width"
  ) +
  theme(
    legend.title = element_blank()
  )

print(p2)
```



0.4.5 用 ggplot 作图：facet

用 mtcars 作图，显示 wt 和 mpg 之间的关系，但用 cyl 将数据分组；见下图：

将此图的结果保存为变量 p3 ，以备后面使用；

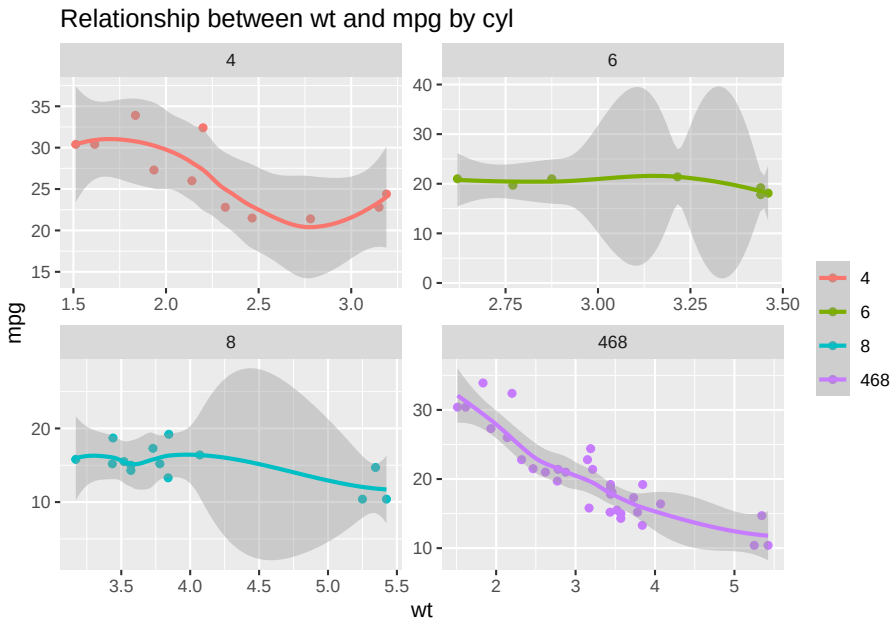


注此图中的 468 组为所有数据合在一起的结果。

```
## 代码写这里，并运行；
library(ggplot2)
library(tidyverse)
library(dplyr)

# Load data
data(mtcars)
```

```r
# Create new data that combines all data
mtcars_modified =
  mtcars %>%
  mutate(cyl = ifelse(cyl %in% c(4, 6, 8), 468, cyl))
mtcars_filtered =
  mtcars_modified %>%
  filter(cyl == 468)
merged_mtcars =
  bind_rows(mtcars, mtcars_filtered)

# Creating facet graphics
p3 =
  ggplot(merged_mtcars, aes(x = wt, y = mpg)) +
  geom_point(aes(color = factor(cyl))) +
  geom_smooth(aes(color = factor(cyl))) +
  labs(title = "Relationship between wt and mpg by cyl", x = "wt", y = "mpg") +
  theme(legend.title = element_blank()) +
  facet_wrap(. ~ cyl,
             ncol = 2,
             dir = "h",
             scales = "free",
             drop = FALSE)


# Save it
p3
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



0.4.6 用 ggplot 作图：facet 2

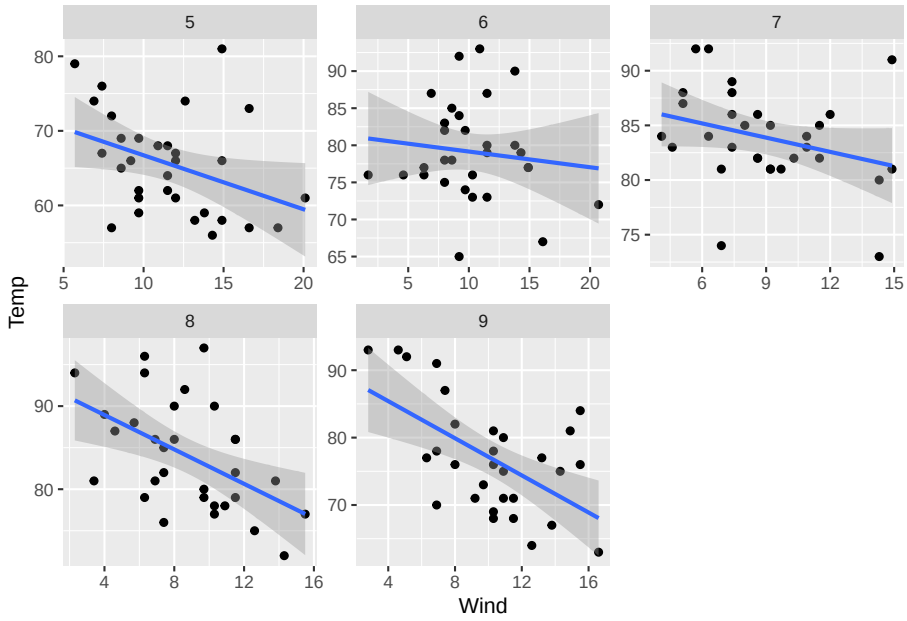
用 airquality 作图，显示 Wind 和 Temp 之间的关系，用 Month 将数据分组；得到的子图按 2 行 3 列组织。

注画点线图，并增加 smooth 图层。

```
## 代码写这里，并运行；
library(ggplot2)

# Creating Scatterplots
ggplot(
  airquality,
  aes(
    x = Wind,
    y = Temp
  )
) +
```

```
  # Draw the dots
  geom_point() +
  # Add smooth layer and
  # fit using a linear model
  geom_smooth(
    method = "lm"
  ) +
  # Grouping with Month and
  # organizing subgraphs by 2 rows and 3 columns
  facet_wrap(
    ~ Month,
    ncol = 3,
    scales = "free"
  )
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

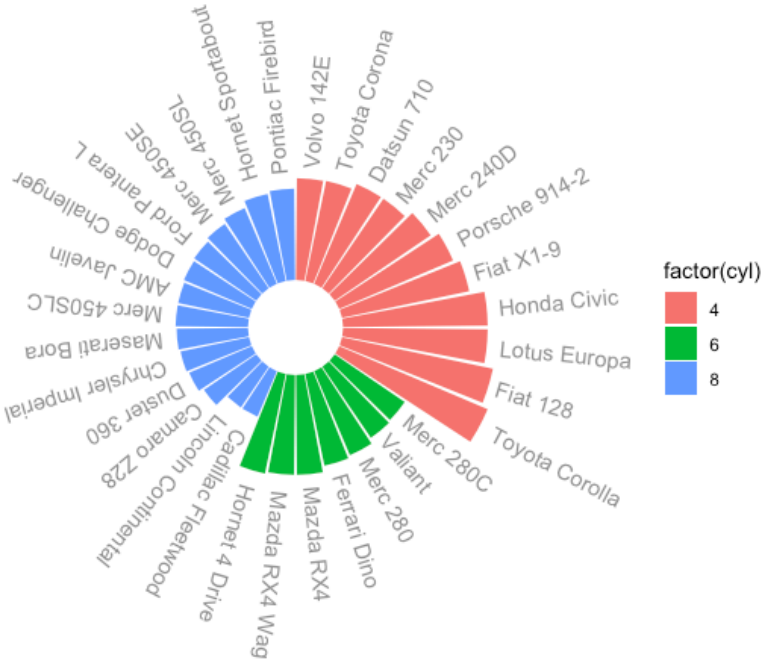


0.4.7 用 ggplot 作图：用 mtcars 做 polar 图

用 mtcars 的 mpg 列做如下图，要求：先按 cyl 排序；每个 cyl 组内按 mpg 排序；将此图的结果保存为变量 p4 ，以备后面使用；

提示

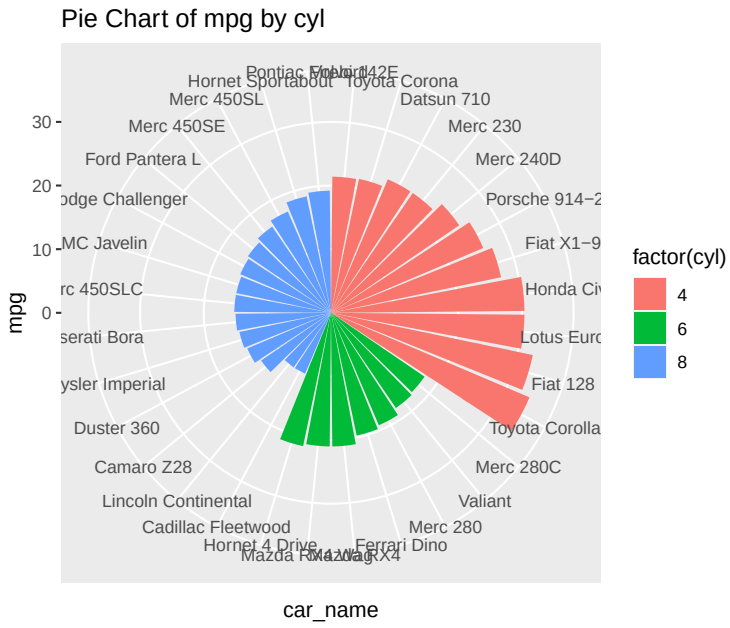
1. 先增加一列,用于保存 rowname：`mtcars %>% rownames_to_column()`
   注：将行名变为列，列名为 `rowname`
2. 完成排序
3. 更改 rowname 的 factor
4. 计算每个 rowname 的旋转角度：`mutate( id = row_number(), angle = 90 - 360 * (id - 0.5) / n() )`




```
## 代码写这里，并运行；
```

```
# THERE ARE STILL BUGS HERE!!!!!!
# 1. Can't insert the hole in the center of the plot
# 2. Can't rotate the label of each tab




# 导入 ggplot2 和 dplyr 包
library(ggplot2)
library(tidyverse)
library(dplyr)

data(mtcars)

# Add a column to save row names
mtcars_named =
  mtcars %>% rownames_to_column(var = "car_name")

# Sorted data
mtcars_arranged =
  mtcars_named %>%
  arrange(cyl, mpg)

# Factors for changing line names
mtcars_arranged$car_name =
  factor(
    mtcars_arranged$car_name,
    levels = mtcars_arranged$car_name)

# Calculate the rotation angle for each line name
mtcars_arranged =
```

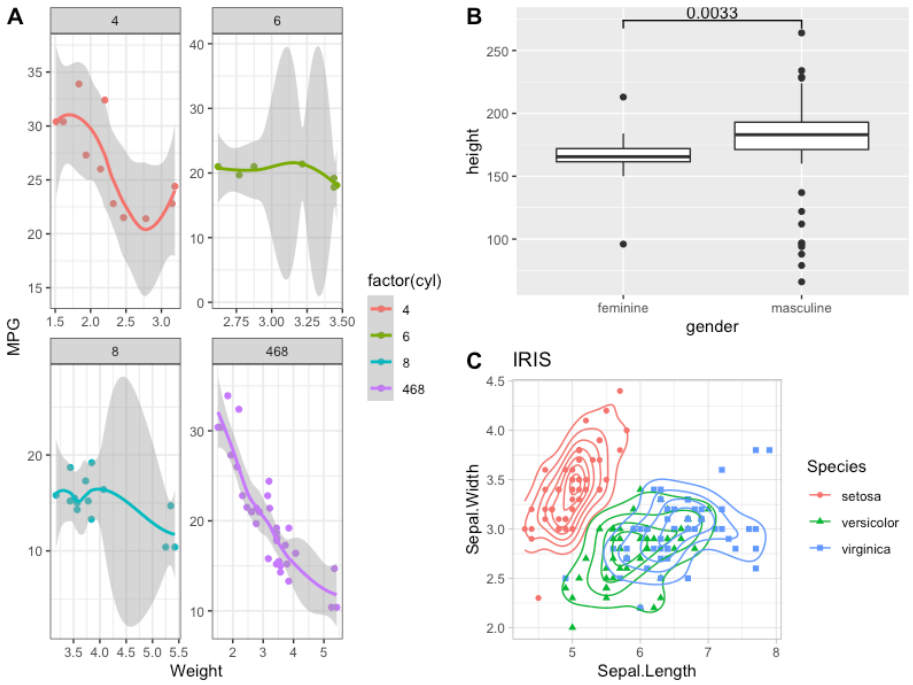
```r
  mtcars_arranged %>%
  mutate(id = row_number(),
         angle = 90 - 360 * (id - 0.5) / n())

# Creating a Coordinate Chart
p4 =
  ggplot(
    mtcars_arranged,
    aes(
      x = car_name,
      y = mpg,
      fill = factor(cyl)
    )
  ) +
  geom_bar(stat = "identity") +
  coord_polar(theta = "x") +



  # THE BUG MAY BE FIXED BY FOLLOWING CODES

  # scale_x_discrete(labels = mtcars_arranged$car_name) +
  # theme(axis.title.x = element_text(angle = (mtcars_arranged$angle))) +



  labs(title = "Pie Chart of mpg by cyl")


# Save to p4
p4
```



0.5 练习与作业 2：多图组合，将多个图画在一起

0.5.1 用 `cowplot::ggdraw` 将 p1, p2 和 p3 按下面的方式组合在一起

注：需要先安装 `cowplot` 包




```
## 代码写这里，并运行；
library(ggplot2)
library(cowplot)
```

```
##
## Attaching package: 'cowplot'

## The following object is masked from 'package:lubridate':
##
##     stamp
```

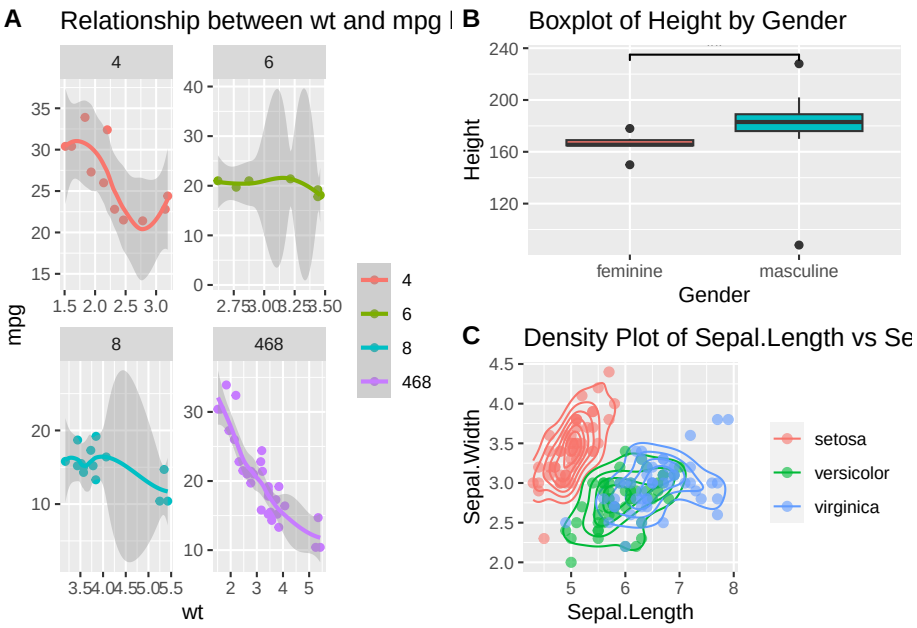
```
# Combine the plots
combined_plot_01 =
  ggdraw() +
  draw_plot(
    plot = p3,
    x = 0,
```

```
    y = 0,
    width = 0.5,
    height = 1
  ) +
  draw_plot(
    plot = p1,
    x = 0.5,
    y = 0.5,
    width = 0.5,
    height = 0.5
  ) +
  draw_plot(
    plot = p2,
    x = 0.5,
    y = 0,
    width = 0.5,
    height = 0.5
  ) +
  draw_plot_label(
    label = c("A", "B", "C"),
    x = c(0, 0.5, 0.5),
    y = c(1, 1, 0.5),
    size = 13
  )
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

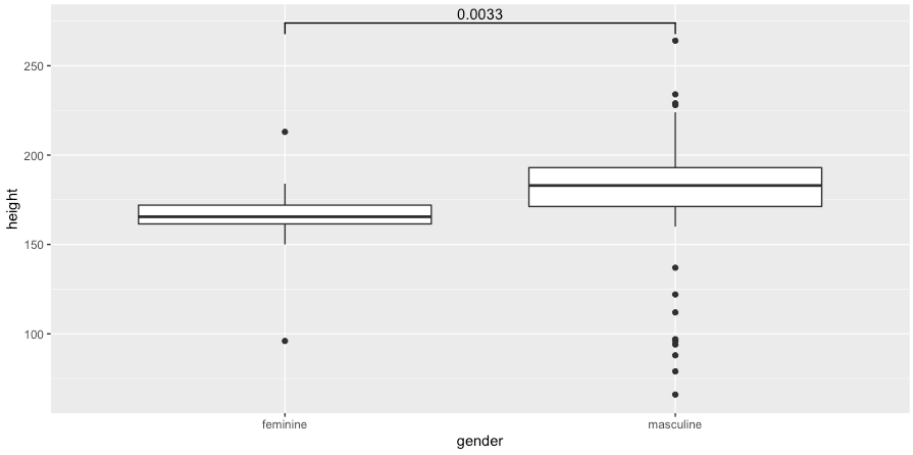
```
## Warning in wilcox.test.default(c(150, 165, 178, 170, 166, 165), c(172, 202, :
## cannot compute exact p-value with ties
```

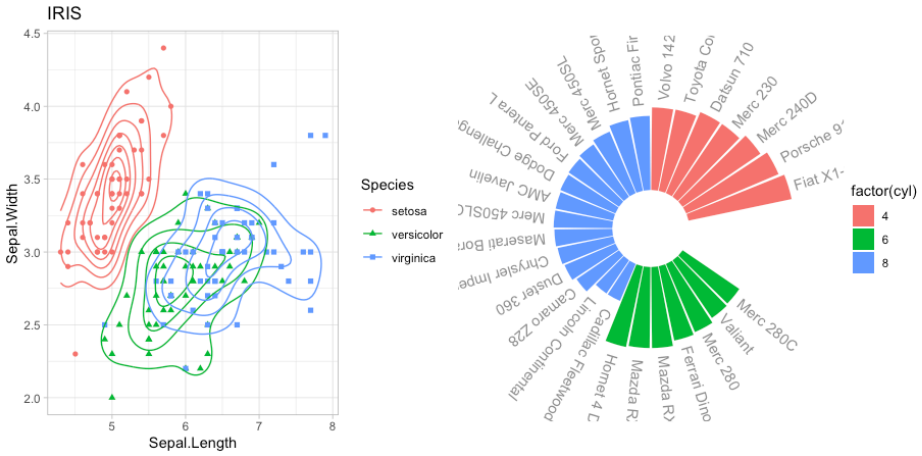
```
ggdraw(combined_plot_01)
```



0.5.2 用 `gridExtra::grid.arrange()` 函数将 `p1`, `p2`, `p4` 按下面的方式组合在一起

注：1. 需要安装 `gridExtra` 包；2. 请为三个 panel 加上 A, B, C 字样的标签。



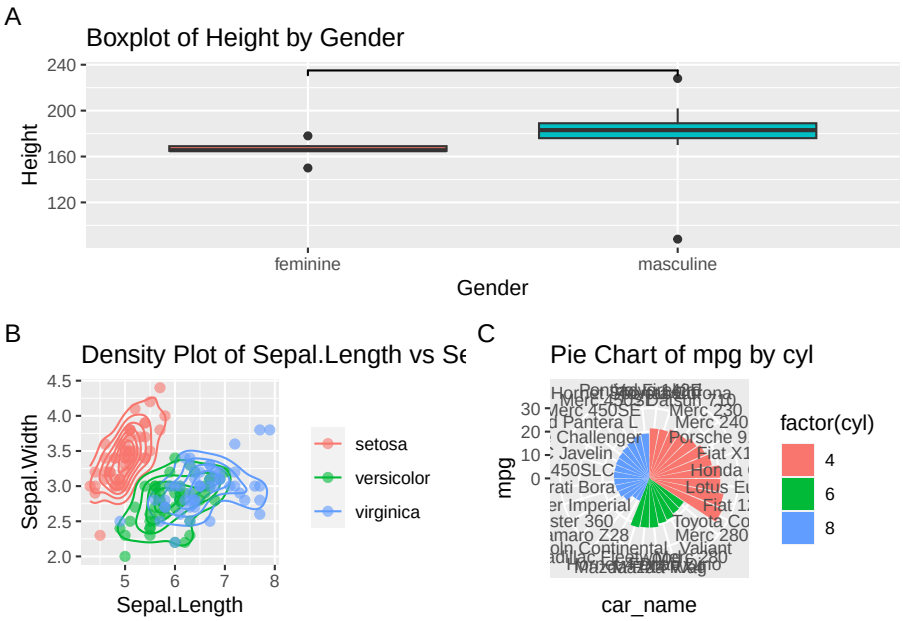



```
## 代码写这里，并运行；
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

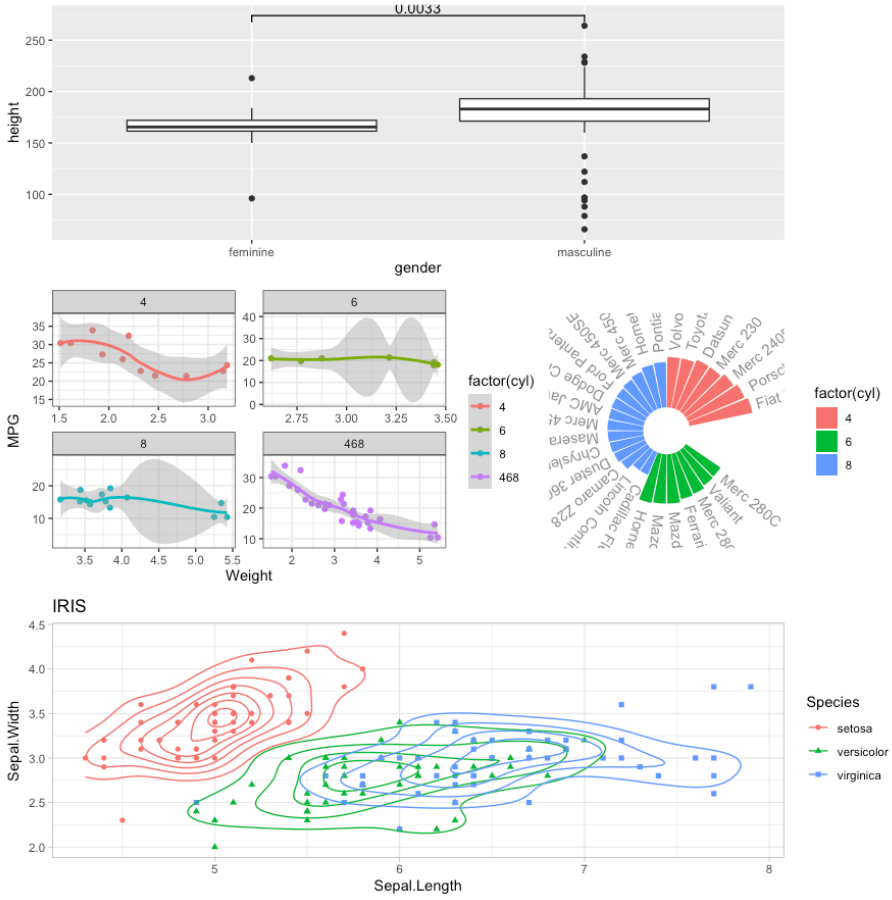
```r
# Add the tags
p1 =
  p1 +
  labs(tag = "A")
p2 =
  p2 +
  labs(tag = "B")
p4 =
  p4 +
  labs(tag = "C")

# Draw the plot
combined_plot_02 =
  grid.arrange(
    p1,p2,p4,
    ncol = 2,
    layout_matrix =
      cbind(c(1, 2), c(1, 3))
  )
```

```
## Warning in wilcox.test.default(c(150, 165, 178, 170, 166, 165), c(172, 202, :
## cannot compute exact p-value with ties
```



0.5.3 用 patchwork 包中的相关函数将 p1, p2, p3, p4 按下面的方式组合在一起

注：1. 需要安装 patchwork 包；2. 为四个 panel 加上 A, B, C, D 字样的标签。



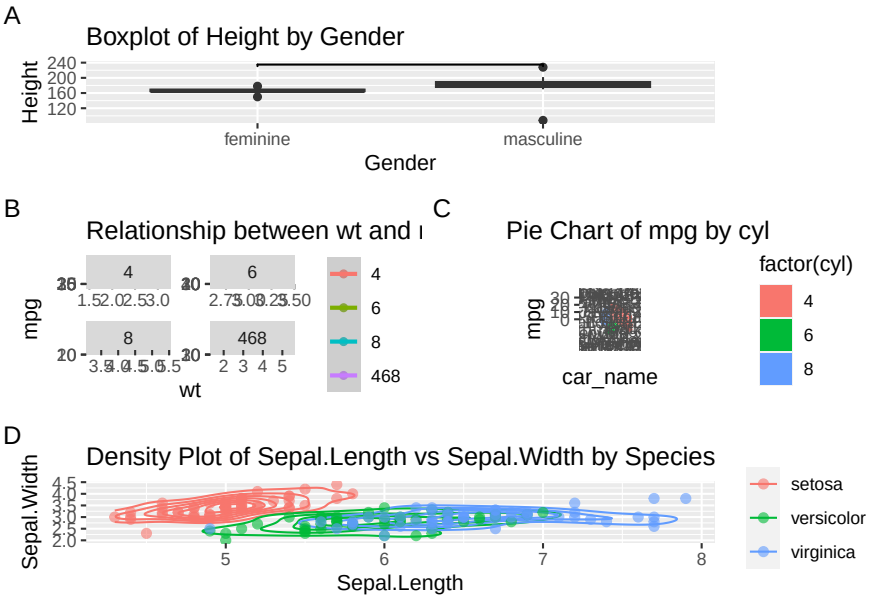

```
## 代码写这里，并运行;

library(ggplot2)
library(patchwork)
```

```
##
## Attaching package: 'patchwork'

## The following object is masked from 'package:cowplot':
##
##     align_plots
```

```
# Add the tags
p1 =
  p1 +
  labs(tag = "A")
p2 =
  p2 +
  labs(tag = "D")
p3 =
  p3 +
  labs(tag = "B")
p4 =
  p4 +
  labs(tag = "C")

# Draw the plot
layout_03 =
  "
  AAAA
  BBCC
  DDDD
"

combined_plot_03 =
  p1/p3/p4/p2 +
  plot_layout(
    design = layout_03,
    ncol = 2,
    nrow = 3,
    heights = c(1, 1, 1, 1),
    widths = c(2, 1, 1, 2)
  )

print(combined_plot_03)
```

```
## Warning in wilcox.test.default(c(150, 165, 178, 170, 166, 165), c(172, 202, :
## cannot compute exact p-value with ties
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



0.6 练习与作业 3：作图扩展

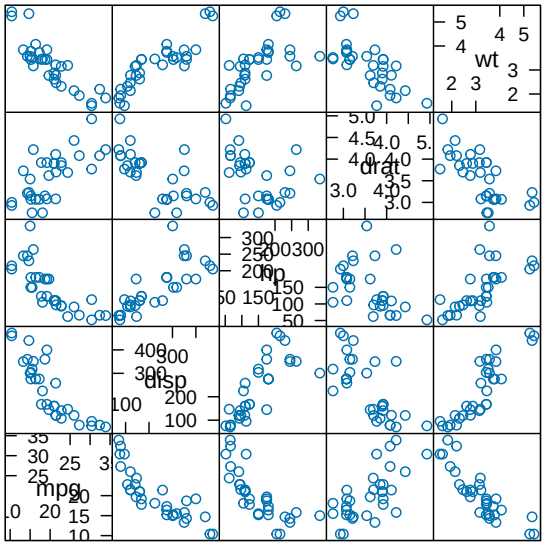
0.6.1 scatterplot

安装 lattice 包，并使用其 splom 函数作图：

```
lattice::splom( mtcars[c(1,3,4,5,6)] )
```

```
## 代码写这里，并运行；
library(lattice)
library(tidyverse)
```

```
splom(mtcars[c(1, 3, 4, 5, 6)])
```



Scatter Plot Matrix