

talk09 练习与作业

目录

0.1 练习和作业说明	1
0.2 talk09 内容回顾	1
0.3 练习与作业：用户验证	2
0.4 练习与作业 1：基础做图 & ggplot2	3
0.5 练习与作业 2：多图组合，将多个图画在一起	20
0.6 练习与作业 3：作图扩展	26

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 grammar (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] "mingyuwang"
```

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

```
## [1] "C:/Users/rhong/Documents"
```

```
library("tidyverse")
library("ggsignif")
library("viridis")
library("cowplot")
library("gridExtra")
library("patchwork")
library("lattice")
```

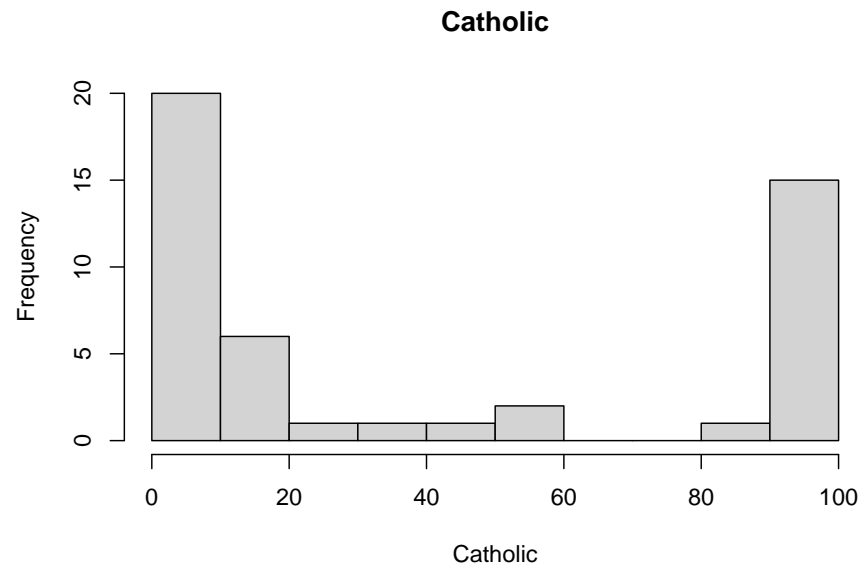
0.4 练习与作业 1: 基础做图 & ggplot2

0.4.1 用 `swiss` 数据做图

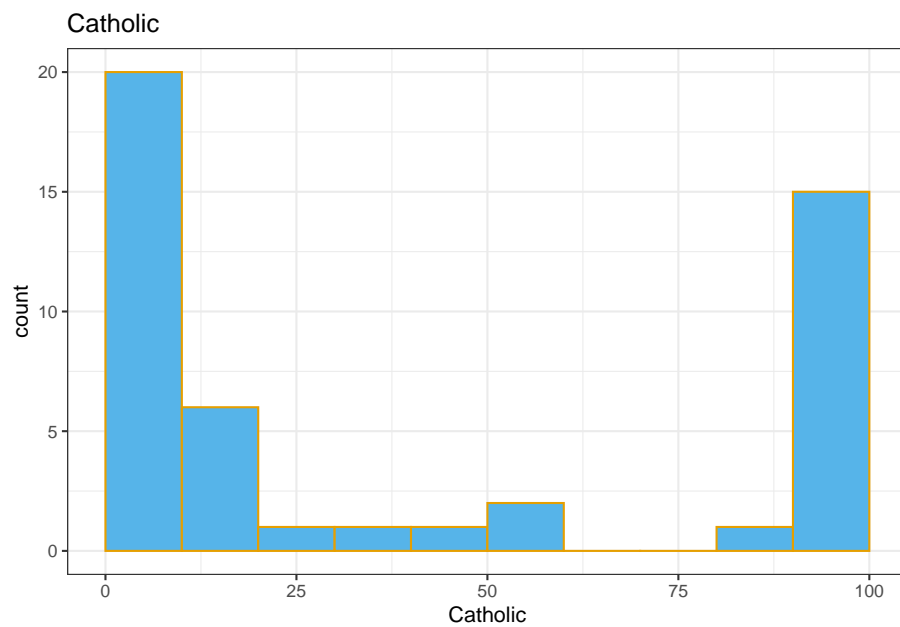
1. 用直方图 `histogram` 显示 `Catholic` 列的分布情况;
2. 用散点图显示 `Eduction` 与 `Fertility` 的关系; 将表示两者关系的线性公式、相关系数和 `p` 值画在图的空白处。

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;
# 1. 用直方图 histogram 显示 Catholic 列的分布情况;
# 1.1 基础做图函数
# 修改标题和 x 轴标签
hist(swiss$Catholic, main = "Catholic", xlab = "Catholic")
```



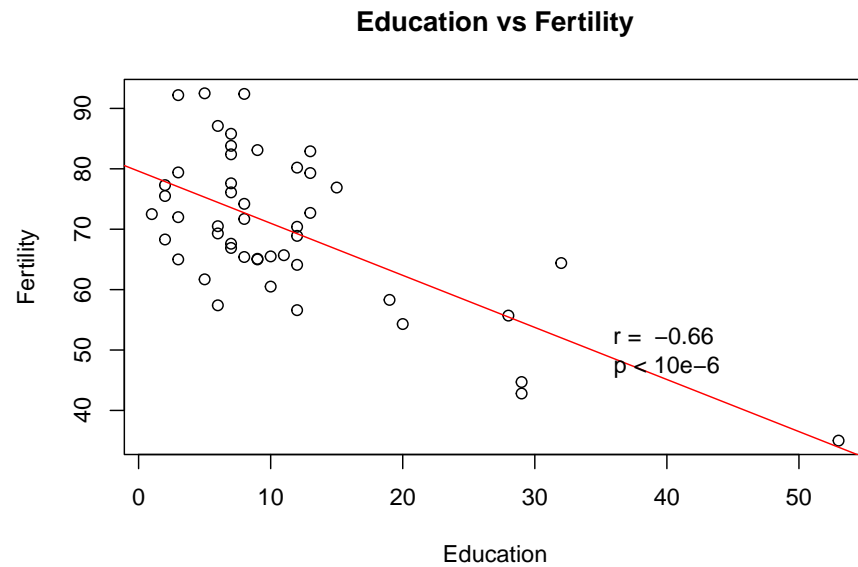
```
# 1.2 ggplot2
# 定制每个区间的范围
ggplot(swiss, aes(x = Catholic)) +
  geom_histogram(breaks = seq(0, 100, 10),
    color = "#E69F00", fill = "#56B4E9") +
  labs(title = "Catholic", x = "Catholic") +
  theme_bw()
```



2. 用散点图显示 *Education* 与 *Fertility* 的关系并将表示两者关系的线性公式、相关系数和 *p* 值

2.1 基础绘图函数

```
plot(swiss$Education, swiss$Fertility,  
     xlab = "Education", ylab = "Fertility",  
     main = "Education vs Fertility")  
abline(lm(swiss$Fertility ~ swiss$Education), col = "red")  
# 相关系数和 p 值画在图的空白处  
text(40, 50,  
     paste("r = ",  
           round(cor(swiss$Education, swiss$Fertility), 2),  
           "\n p < 10e-6 "))
```



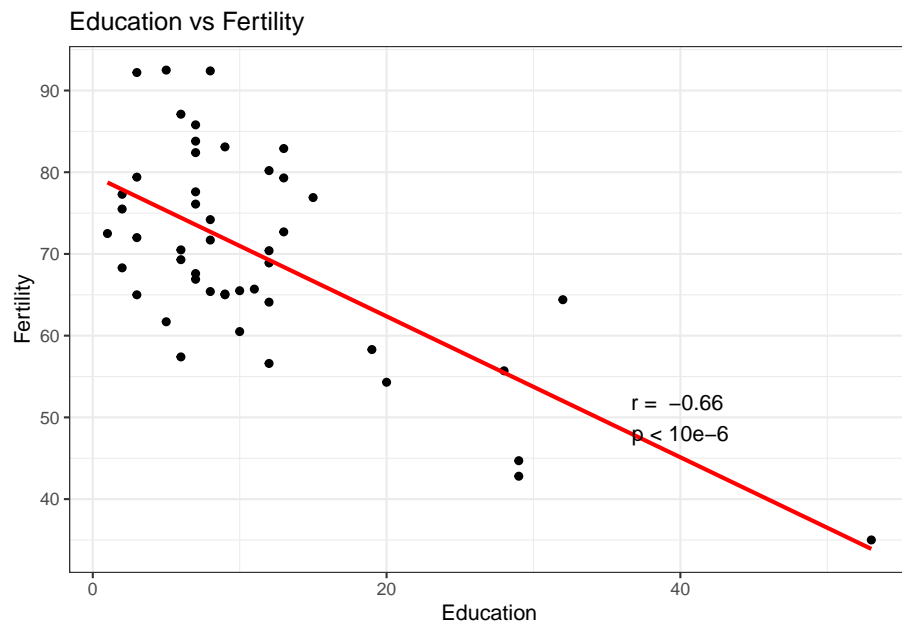
```
cor.test(swiss$Education, swiss$Fertility)$p.value
```

```
## [1] 3.658617e-07
```

```
# 2.2 ggplot2
```

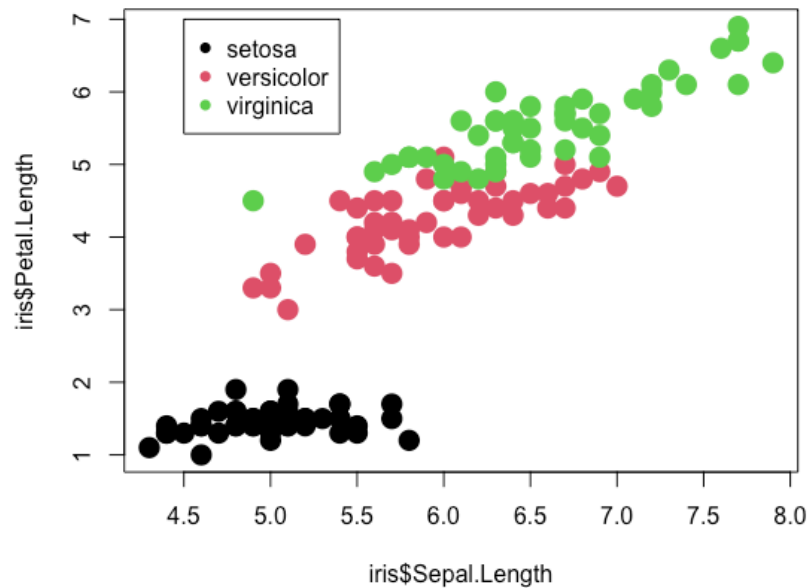
```
ggplot(swiss, aes(x = Education, y = Fertility)) +  
  geom_point() +  
  geom_smooth(method = "lm", se = FALSE, color = "red") +  
  labs(title = "Education vs Fertility", x = "Education", y = "Fertility") +  
  theme_bw() +  
  annotate("text", x = 40, y = 50,  
    label = paste("r = ",  
      round(cor(swiss$Education, swiss$Fertility), 2),  
      "\n p < 10e-6 "))
```

```
## `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；

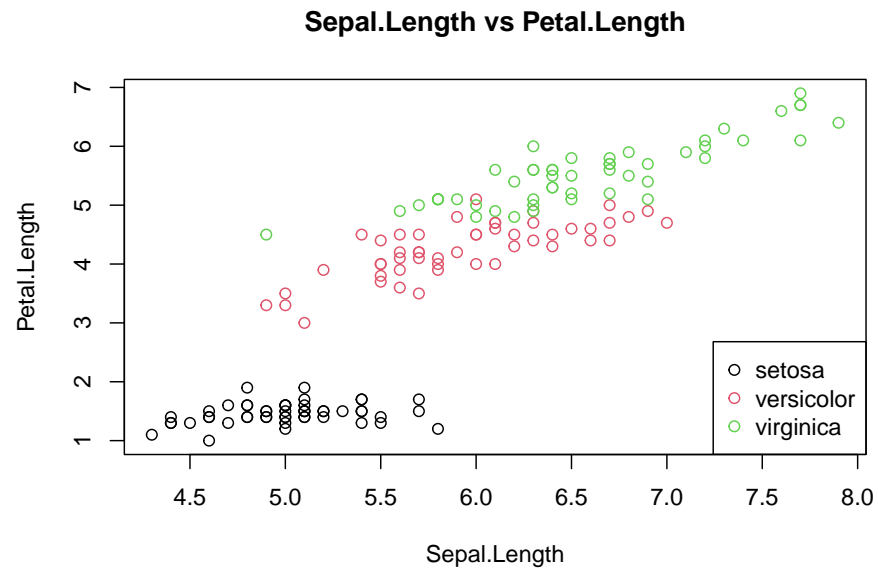


如下图所示：

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

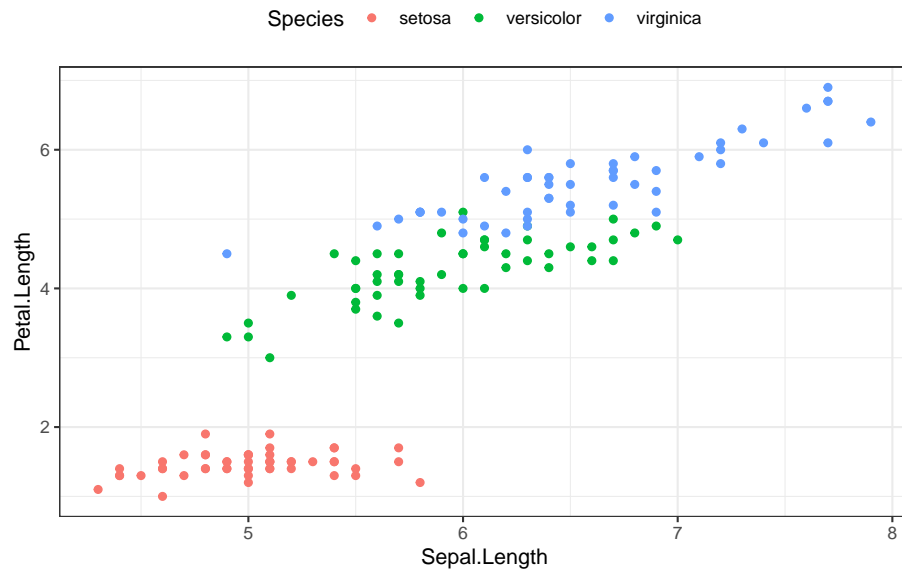
注：每种图提供基础做图函数和 ggplot2 两个版本！

```
## 代码写这里，并运行；  
# 散点图  
# 1. 基础做图函数  
plot(iris$Sepal.Length, iris$Petal.Length,  
      col = iris$Species, xlab = "Sepal.Length",  
      ylab = "Petal.Length",  
      main = "Sepal.Length vs Petal.Length")  
legend("bottomright", legend = levels(iris$Species), col = 1:3, pch = 1)
```

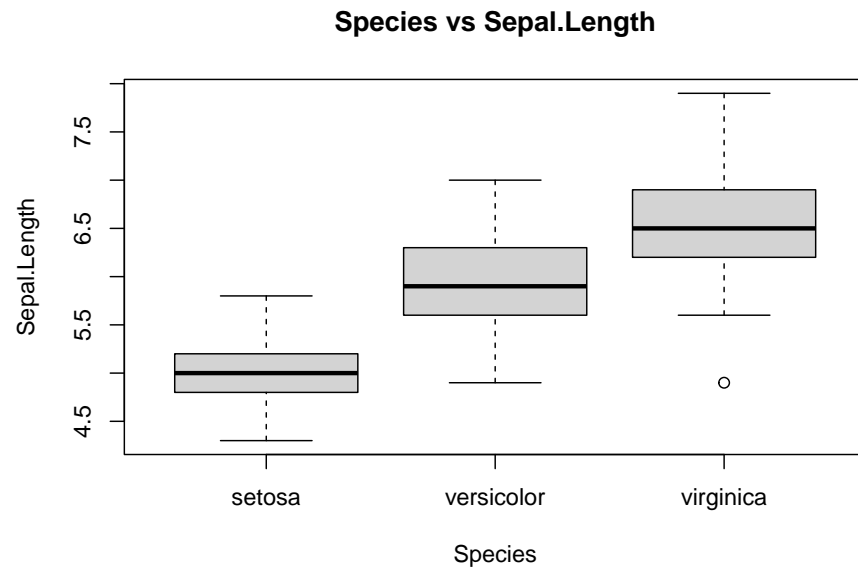



```
# 2. ggplot2
ggplot(iris, aes(x = Sepal.Length, y = Petal.Length, color = Species)) +
  geom_point() +
  labs(title = "Sepal.Length vs Petal.Length",
       x = "Sepal.Length", y = "Petal.Length") +
  theme_bw() +
  theme(legend.position = "top")
```

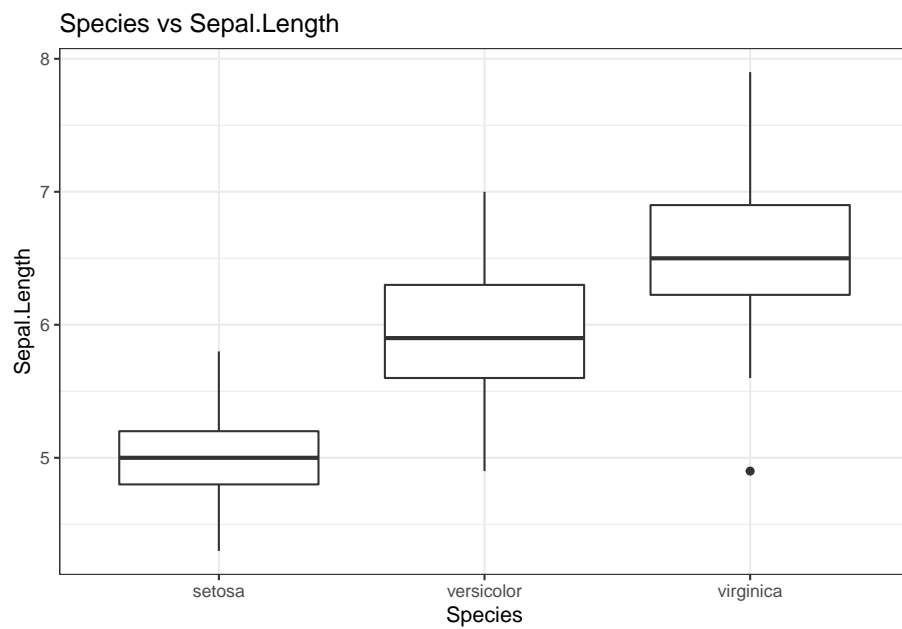
Sepal.Length vs Petal.Length



```
# boxplot
# 1. 基础做图函数
boxplot(iris$Sepal.Length ~ iris$Species,
        xlab = "Species", ylab = "Sepal.Length",
        main = "Species vs Sepal.Length")
```



```
# 2. ggplot2
ggplot(iris, aes(x = Species, y = Sepal.Length)) +
  geom_boxplot() +
  labs(title = "Species vs Sepal.Length",
       x = "Species", y = "Sepal.Length") +
  theme_bw()
```

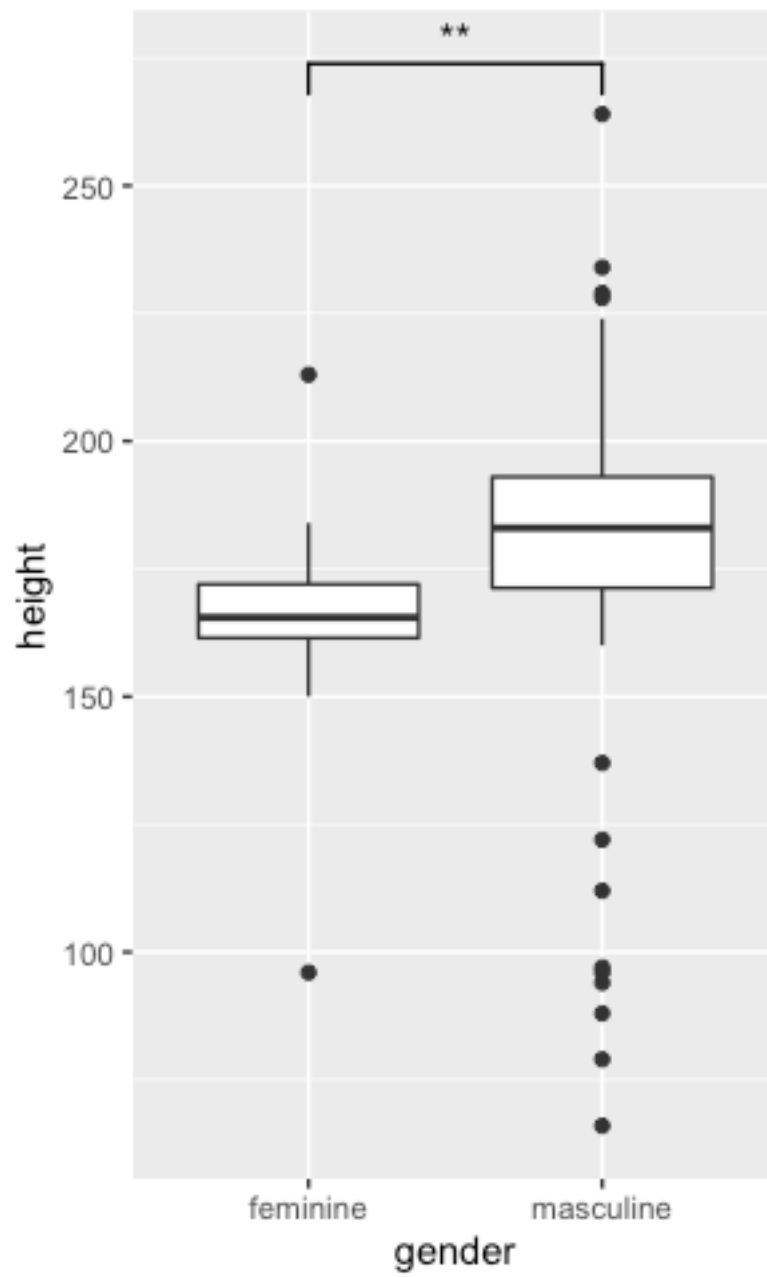


0.4.3 用 ggplot 作图: boxplot

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 `ggsigif` 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；

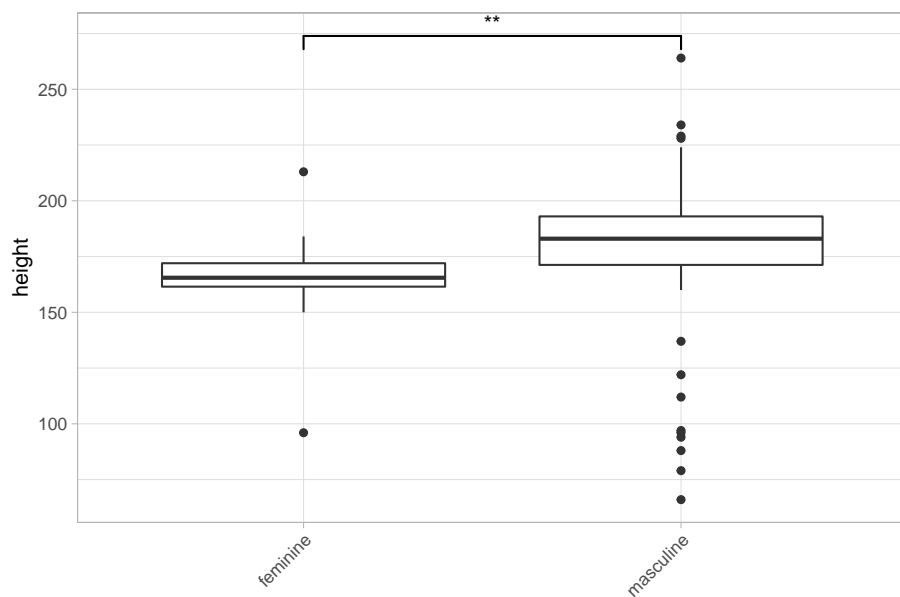
最终结果如图所示：



```
## 代码写这里，并运行；
```

```
(p1 <- ggplot(filter(starwars, !is.na(height) & !is.na(gender)),
```

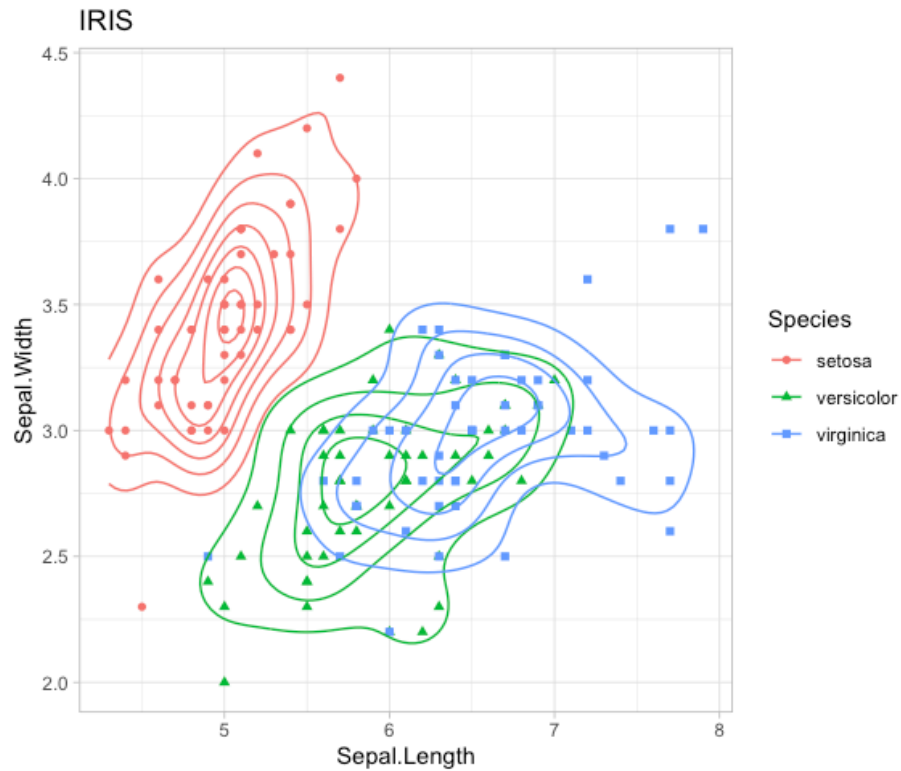
```
aes(x = gender, y = height)) +  
geom_boxplot() +  
geom_signif(comparisons = list(c("feminine", "masculine")),  
  map_signif_level = TRUE) +  
scale_fill_viridis(discrete = TRUE, alpha = 0.6) +  
theme_light() +  
theme(  
  legend.position = "none",  
  axis.text.x = element_text(angle = 45, hjust = 1)  
) +  
xlab("")
```



0.4.4 用 ggplot 作图：使用 iris 做图

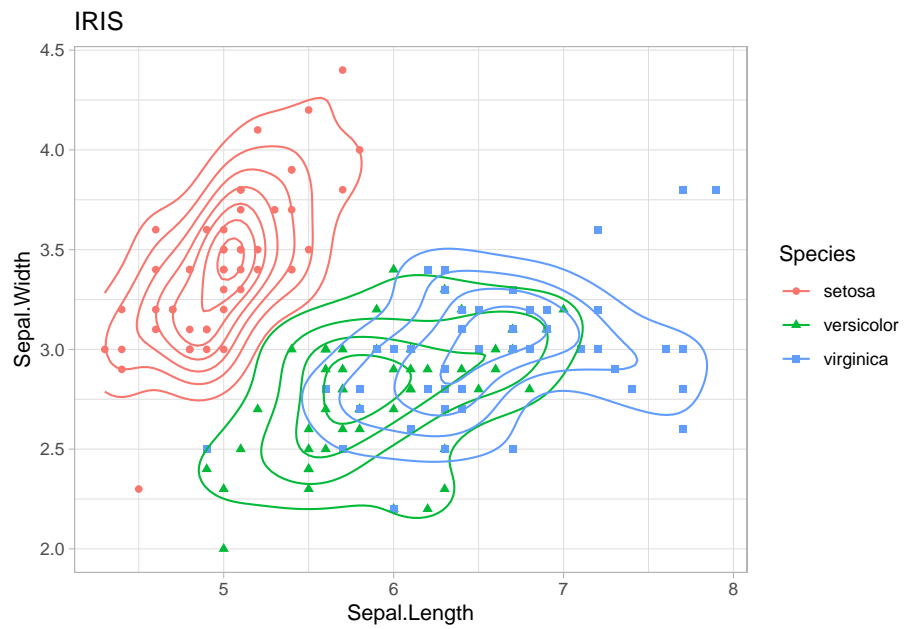
用 `geom_density2d` 显示 `Sepal.Length` 和 `Sepal.Width` 之间的关系，同时以 `Species` 为分组，结果如图所示：

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；

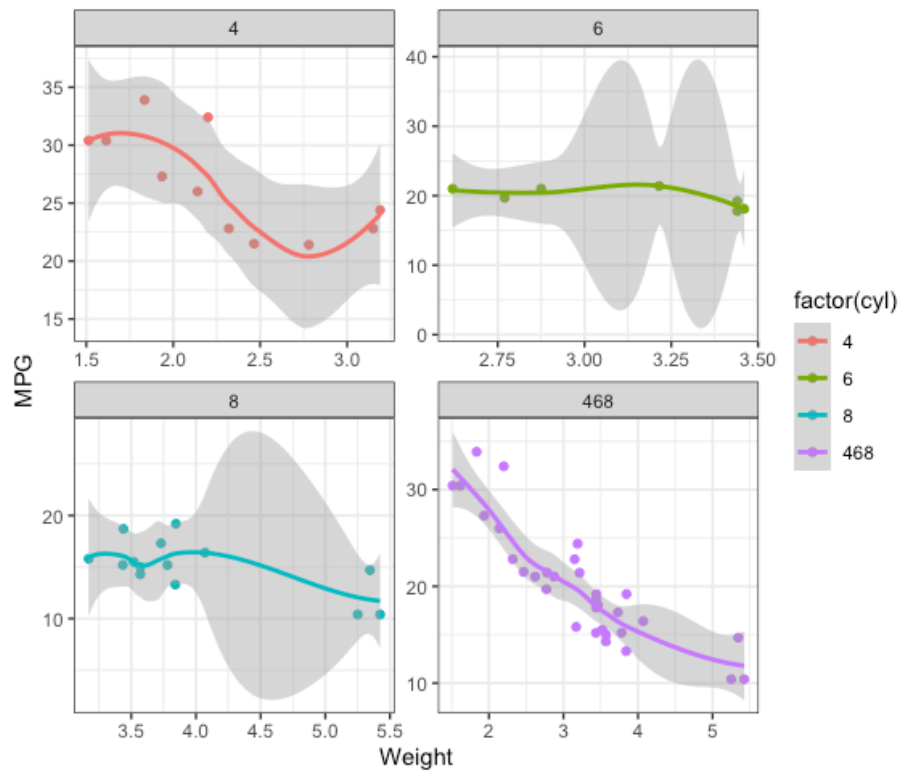
```
(p2 <- ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width,
  color = Species, shape = Species)) +
  geom_point() +
  geom_density2d() +
  theme_light() +
  xlab("Sepal.Length") +
  ylab("Sepal.Width") +
  labs(title = "IRIS"))
```



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：

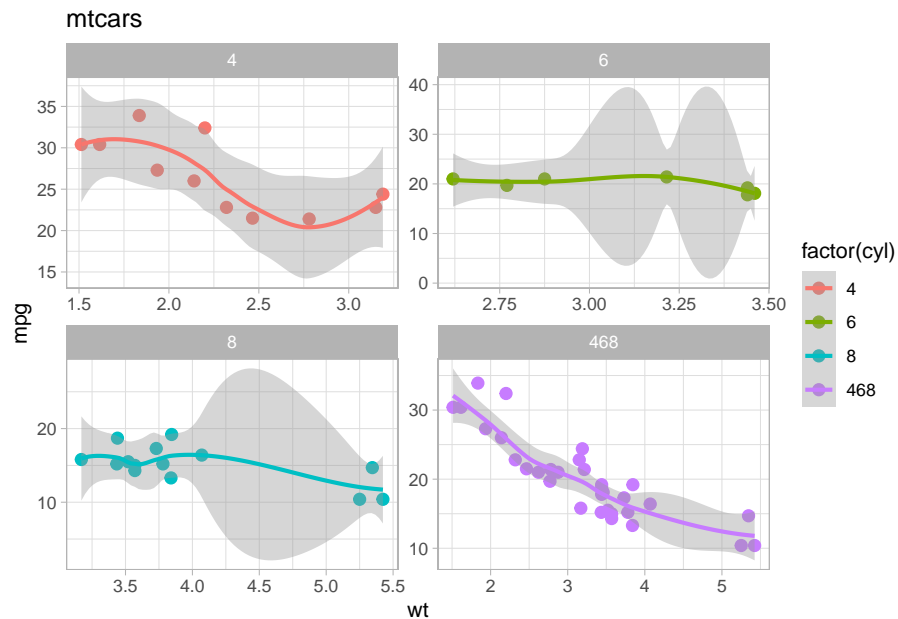
将此图的结果保存为变量 `p3`，以备后面使用；



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；
# mtcars 增加 cyl，加一组 468
my_mtcars <- bind_rows(mtcars, mutate(mtcars, cyl = 468))
(p3 <- ggplot(my_mtcars, aes(x = wt, y = mpg, color = factor(cyl))) +
  geom_point(size = 2.5) +
  geom_smooth(method = "loess") +
  facet_wrap(~ cyl, scales = "free", nrow = 2) +
  # facet_grid(. ~ cyl, scales = "free", margins = TRUE) +
  theme_light() +
  xlab("wt") +
  ylab("mpg") +
  labs(title = "mtcars"))
```

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

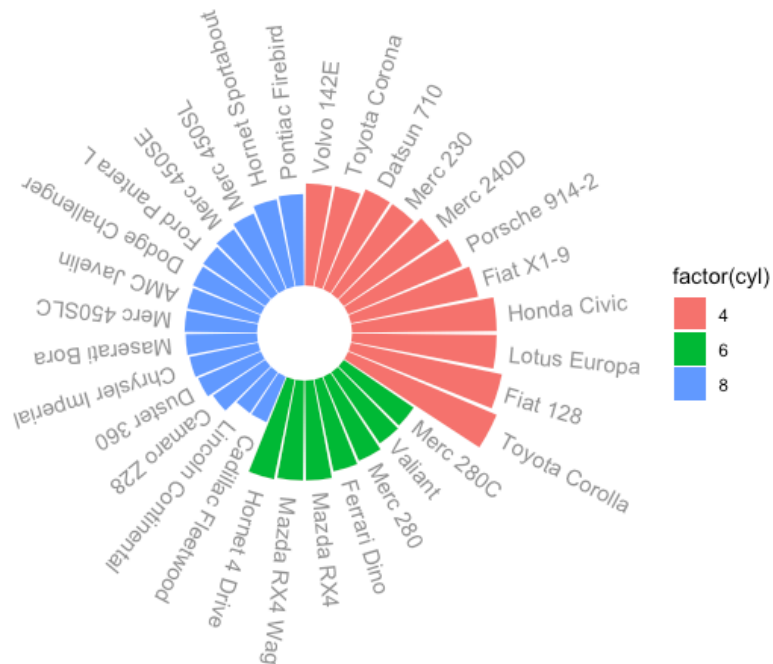


0.4.6 用 ggplot 作图：用 mtcars 做 polar 图

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())`



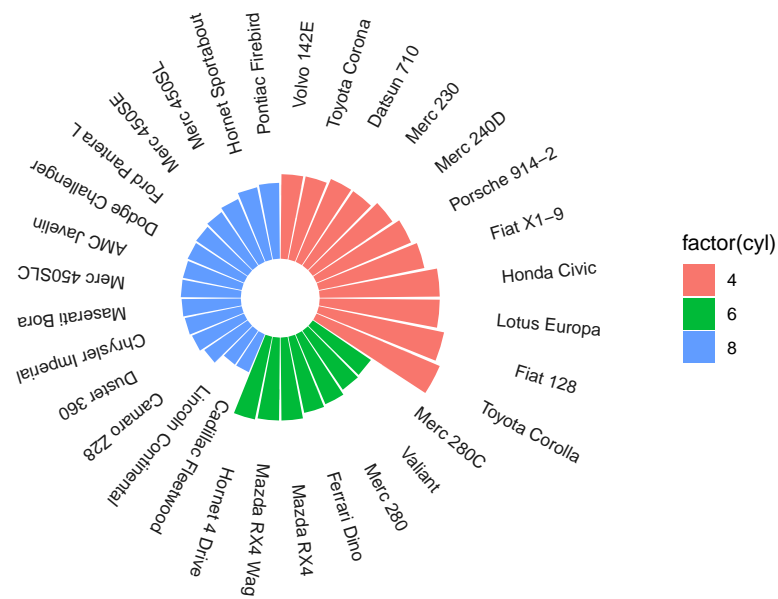
```
## 代码写这里，并运行；
mdf_mtcars <- mtcars %>% rownames_to_column() %>%
  arrange(cyl, mpg) %>%
  mutate(rowname = factor(rowname, levels = rowname)) %>%
  mutate(id = row_number(),
         angle = 90 - 360 * (id - 0.5) / n()) %>%
  select(rowname, mpg, cyl, angle)

(p4 <- ggplot(mdf_mtcars, aes(x = rowname)) +
  geom_bar(stat = "identity", aes(y = mpg, fill = factor(cyl))) +
  # Limits of the plot = very important.
  # The negative value controls the size of the inner circle,
  # the positive one is useful to add size over each bar
  ylim(-10, 65) +
  coord_polar(start = 0, direction = 1, clip = "on") +
  # 清空坐标轴
```

```

theme_minimal() +
theme(
  axis.text = element_blank(),
  axis.title = element_blank(),
  panel.grid = element_blank(),
  plot.margin = unit(rep(-1,4), "cm")
) +
# Add the labels, using the mdf_mtcars dataframe
geom_text(data = mdf_mtcars,
  aes(x = rowname, y = mpg + 28, label = rowname),
  color = "black", alpha = 0.9, size = 3,
  angle = mdf_mtcars$angle))

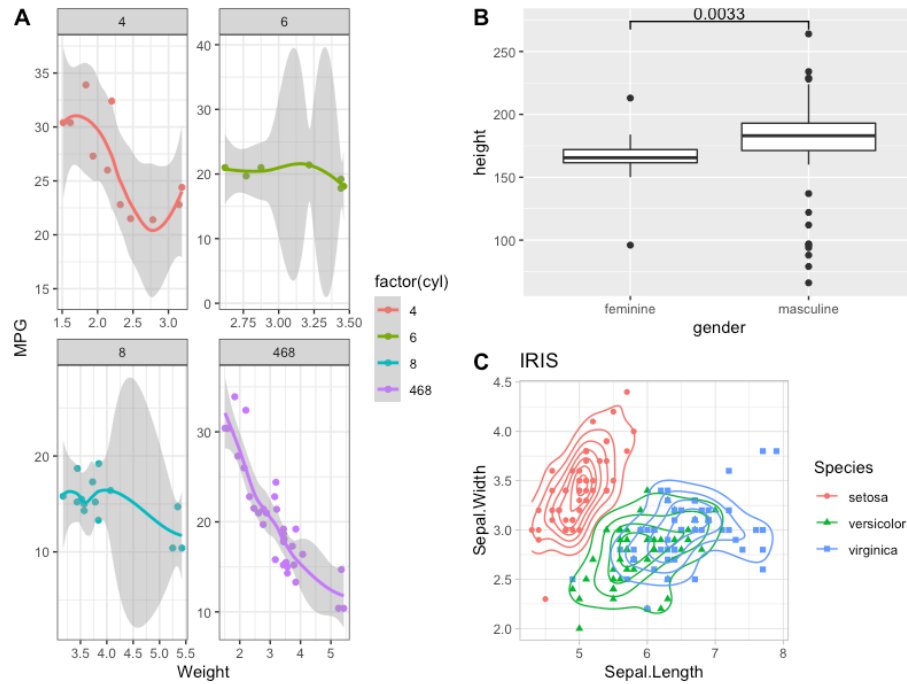
```



0.5 练习与作业 2: 多图组合, 将多个图画在一起

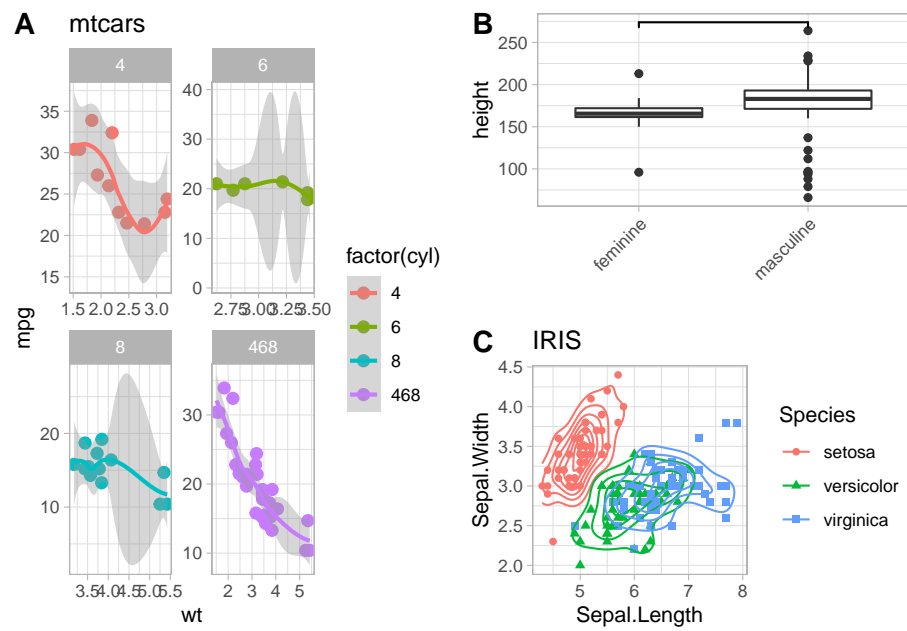
0.5.1 用 `cowplot::ggdraw` 将 p1, p2 和 p3 按下面的方式组合在一起

注：需要先安装 `cowplot` 包



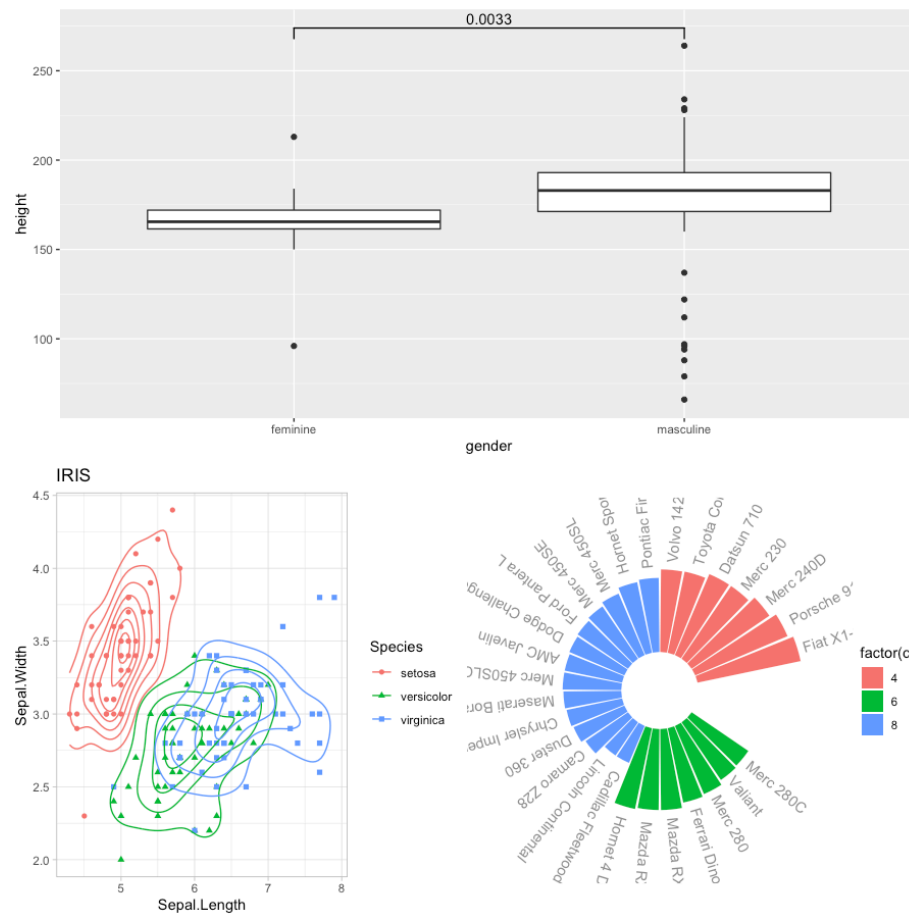
代码写这里，并运行；

```
ggdraw() +
  draw_plot(p3, x = 0, y = 0, width = 0.5, height = 1) +
  draw_plot(p2, 0.5, 0, 0.5, 0.5) +
  draw_plot(p1, 0.5, 0.5, 0.5, 0.5) +
  draw_plot_label(c("A", "B", "C"), c(0, 0.5, 0.5),
    c(1, 1, 0.5), size = 15)
```

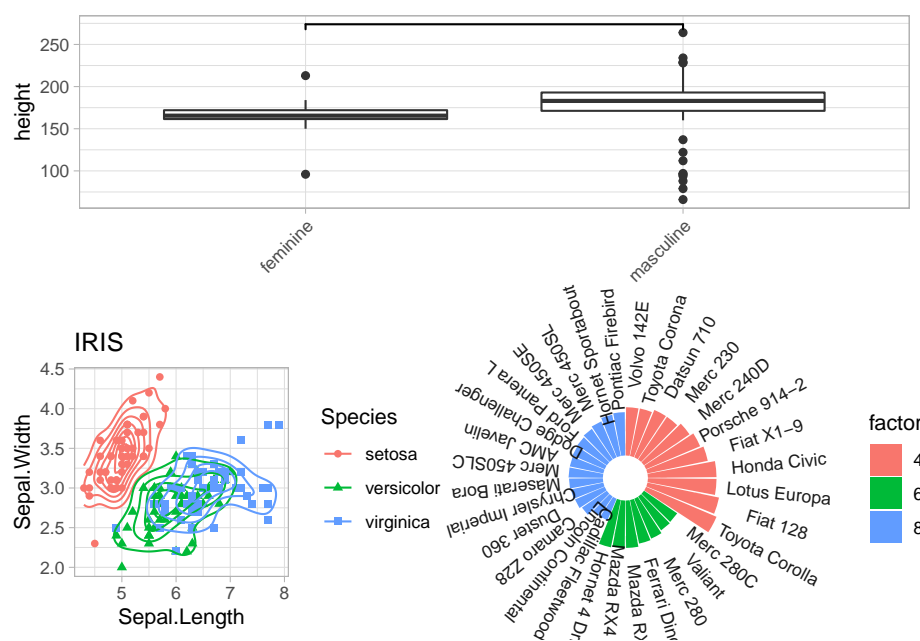


0.5.2 用 `gridExtra::grid.arrange()` 函数将 p1, p2, p4 按下面的方式组合在一起

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

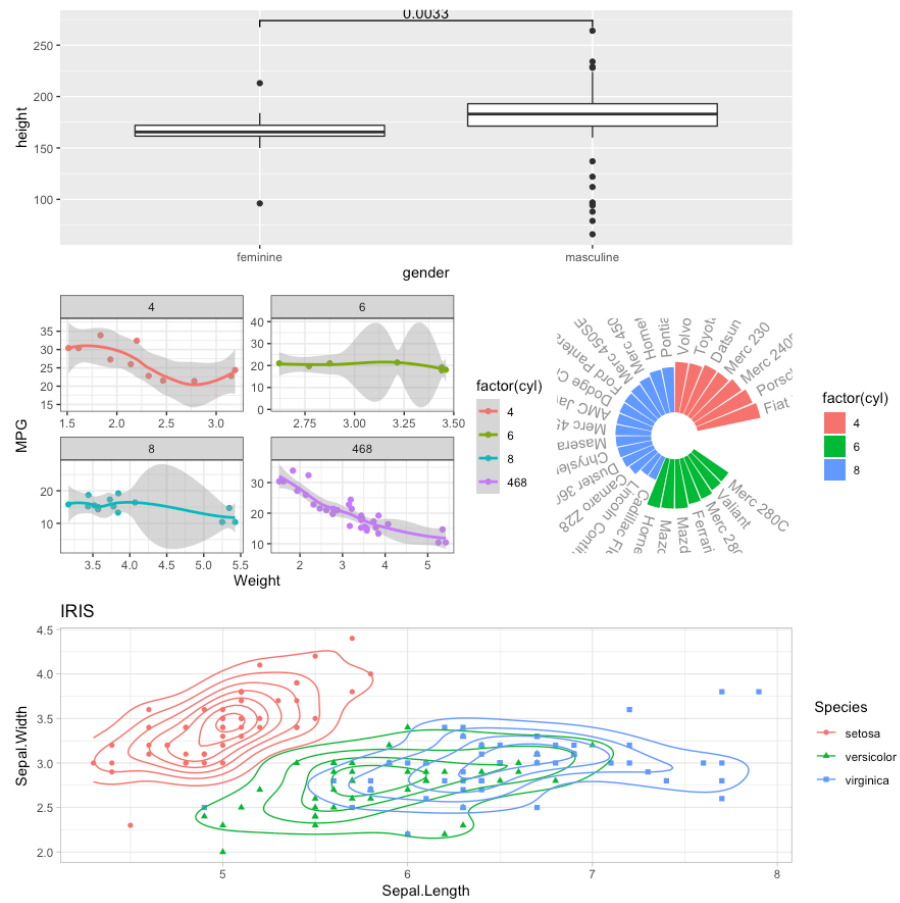


```
## 代码写这里，并运行；
gridExtra::grid.arrange(p1, p2, p4,
  nrow = 2, layout_matrix = rbind(
    c(1, 1),
    c(2, 3)
  ))
```



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;
```

```
p1 / (p2 | p3) / p4 +
```

```
plot_annotation(
```

```
  title = "A title",
```

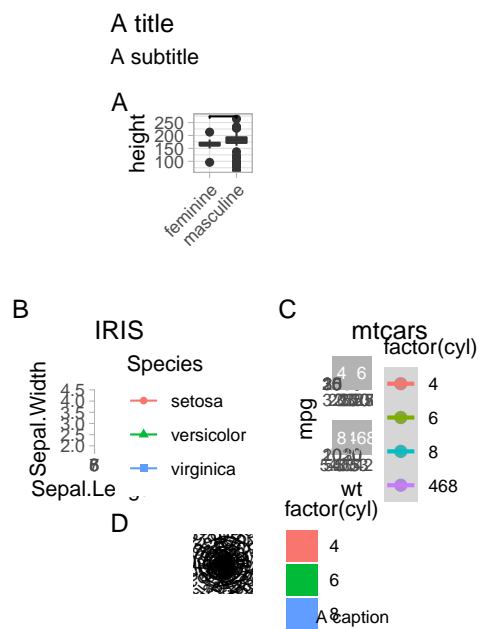
```
  subtitle = "A subtitle",
```

```
  caption = "A caption",
```

```
  tag_levels = "A"
```

```
)
```

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



0.6 练习与作业 3：作图扩展

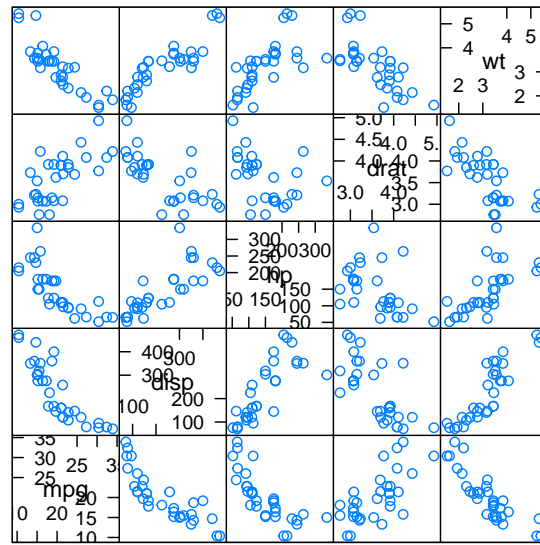
0.6.1 scatterplot

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；

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



Scatter Plot Matrix