**数据框架(Data Frames):**

创建新数据集：使用c()，matrix()，和data.frame()函数。

数据框架的结构：利用mode()和str()函数查看数据框架的结构。

访问数据框架：使用$，[]，和[[]]进行数据访问。

**NA值处理:**

使用subset()函数选择数据子集。

使用complete.cases()函数移除含有NA的行。

**数据框架合并:**

使用rbind()和cbind()函数添加行或列。

使用merge()函数合并具有至少一个共同列名的两个数据框架。

**dplyr包的使用:**

filter()：根据条件选择行。

select()：从数据框中选择变量。

mutate()：基于现有列计算新列。

summarise()：计算变量的单数摘要。

arrange()：更改行的顺序。

**数据预处理:**

数据导入：使用readxl包和read\_excel()函数导入数据。

数据筛选与选择：使用管道操作符|>和filter()，select()函数。

数据变换：使用mutate()和ifelse()函数处理和转换数据。

分组操作：使用group\_by()和summarise()函数进行分组摘要统计。

**dplyr与基础R的对比:**

使用tapply()函数在基础R中进行类似dplyr的操作。

**性能比较:**

使用rbenchmark包比较dplyr和基础R的性能。

**教训和结论:**

数据框架和tibbles是R中处理矩形数据集的有用数据结构。

dplyr可以轻松地对数据框架进行数据操作。

**图表基础:**

创建基本图表，设置坐标轴限制，添加图例，使用数学符号和表达式。

**调整图表样式:**

设置持久的默认图表参数，如边距和背景色。

**保存图表:**

将图表保存为多种格式，如PDF或PNG。

**颜色和调色板:**

使用R的颜色系统，添加透明度，更改颜色调色板。

**ggplot2包:**

利用ggplot2包，根据图形的语法来创建图表，包括使用数据、美学和几何对象。

**ggplot2构建块:**

使用ggplot2包中的数据、美学、几何体、分面、统计、坐标和主题。

**ggplot2绘图示例:**

使用ggplot2创建点图、盒图、直方图、密度图和条形图。

**地图绘制:**

使用maps包和ggplot2在地图上绘制数据。

**其他ggplot2功能:**

使用主题、分面和颜色来增强图表的表现力。

**交互式图表:**

使用plotly包将ggplot2图表转换为交互式图表。

**动画图表:**

使用gganimate包制作动态图表。

**其他资源:**

提供学习ggplot2和R数据可视化的其他资源链接。

**总结:**

介绍了在R中使用plot函数和ggplot2包创建和自定义高质量图表的方法。

**R与其他语言的接口：**介绍了如何将R与C、Python等语言结合使用。

**版本控制与GitHub：**讨论了使用GitHub进行版本控制的方法。

**编写自己的R包：**提供了如何使用devtools和roxygen2来创建和文档化R包的指南。

**数据导入：**使用readr和rio包从不同格式导入数据。

**列表和Tibbles：**介绍了列表的使用和tibbles中列表作为列的应用。

**整洁数据：**遵循整洁数据的原则（每个变量形成一列，每个观察形成一行，每个单元格一个值）。

**使用tidyr重塑数据：**使用pivot\_longer()和pivot\_wider()函数转换数据框的形状。

**分类数据处理：**包括使用factors和levels处理分类数据，使用forcats包和table()函数处理和整理分类数据。

**字符数据处理：**使用stringr包处理字符串，包括使用正则表达式、字符串转换和文件名处理。

**日期和时间数据处理：**介绍了如何使用lubridate包处理日期和时间数据，例如创建和解析日期。

**数据整理技巧：**包括如何有效地使用factors和stringr函数进行数据整理和分析。

**循环（For, While, Repeat）：**解释了如何使用不同类型的循环结构。

**条件判断（If-Else语句）：**展示了如何根据条件执行不同的代码块。

**算术和布尔运算符：**介绍了在R中进行基本数学和逻辑操作的方法。

**函数编写：**解释了如何定义函数，包括默认参数值、return语句和匿名函数。

**环境和作用域：**讨论了R中的环境和作用域，如何影响变量的访问。

**递归和排序函数：**介绍了递归函数的使用，包括快速排序算法的示例。

**其他编程技巧：**包括如何编写自定义操作符和处理替换函数。

**对象大小：**使用lobstr包探索对象大小，包括obj\_size()和obj\_sizes()函数。

**内存使用：**讨论了R中内存分配的重要性，包括在RStudio IDE中的内存使用。

**向量赋值和修改时复制：**解释了R中向量赋值的内部工作原理。

**列表和数据框：**讨论列表和数据框的使用及其对内存的影响。

**代码性能分析：**介绍了使用Rprof进行代码性能分析的方法，以及如何使用它来优化代码。

**功能编程的基础**：定义了功能编程的概念，包括R中的函数如何作为一等公民操作。

**Base R的\*apply函数族：**讨论了lapply、sapply、vapply等函数的用法。

**purrr包：**介绍了purrr包，它提供了一组用于功能编程的一致且易于使用的工具。

**各种map函数：**详细说明了如何使用purrr包中的各种map函数（如map()、map2()、pmap()）来处理列表和向量。

**S3类和泛型函数：**介绍了S3类的创建和管理，以及如何使用和编写S3泛型函数。

**S4类：**讨论了S4类的创建和特性，包括如何定义和使用S4对象。

**继承和多态性：**探讨了在R中使用继承和多态性的方法。

**S3和S4的比较：**对比了S3和S4系统的特点和应用场景。

**其他面向对象系统：**简要介绍了R中其他的面向对象编程系统，如R6。