

ЛАБОРАТОРНАЯ РАБОТА №2

Визуализация данных с помощью ggplot2 (функция ggplot) и plotly (функции ggplotly и plot_ly)

Цель работы: продолжить знакомство с элементами грамматики графики пакета ggplot2, научиться использовать функцию ggplot.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Некоторые сведения о ggplot

Функция ggplot пакета ggplot2 является основной функцией для построения графиков. Для построения полноценного графика можно использовать следующий шаблон:

```
ggplot(data = <DATA>) +  
  <GEOM FUNCTION>(mapping = aes)(<MAPPINGS>),  
  stat = <STAT>,  
  position = <POSITION>) +  
  <COORDINATE FUNCTION> +  
  <FACET FUNCTION> +  
  <SCALE FUNCTION> +  
  <THEME FUNCTION>
```

Рисунок 2.1 – Шаблон использования функции ggplot

Более подробную информацию о каждом из объектов можно найти в официальной шпаргалке (cheat sheet) пакета ggplot2 в соответствующем репозитории проекта [GitHub](#).

Пакет plotly

В том случае если необходимо создать интерактивную визуализацию рекомендуется использовать пакет plotly. Пакет plotly тесно интегрирован с пакетом ggplot2, поэтому для визуализации достаточно передать построенный с помощью функции ggplot график в функцию ggplotly, предварительно сохранив предыдущий в переменную.

Plotly не ограничивается функцией ggplotly. У него есть свои конструкции для визуализации, а именно – plot_ly.

Идейно она похожа на ggplot(). В нее тоже можно передать данные, а также информацию об эстетиках, таких, как имя столбца с цветом, группировку, и так далее. Отличием является то, что в plotly мы указываем тип примитивов с помощью параметра type, и объединяем их в один объект с помощью "%>%", а в ggplot'е для каждого типа была своя функция-конструктор, и объединение происходило через "+".

```
plot_ly(economics, x = ~date, color = I("black")) %>%  
  add_lines(y = ~uempmed) %>%  
  add_lines(y = ~psavert, color = I("red"))
```

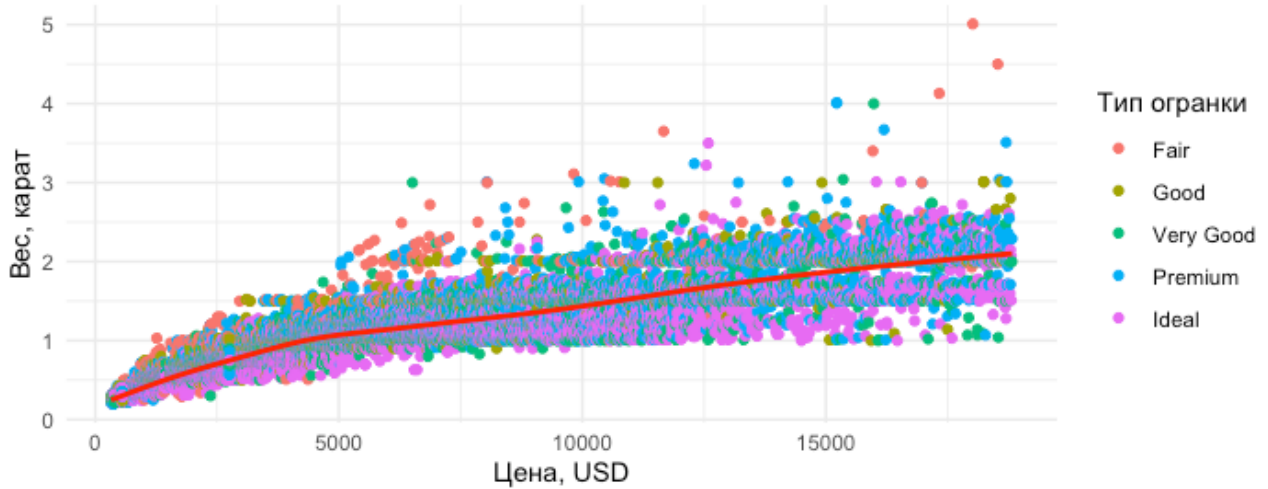
Рисунок 2.2 – Пример работы функции plot_ly

ПРАКТИЧЕСКАЯ ЧАСТЬ

Задание 1 – Используя исходное изображение, повторите его с помощью функции `ggplot` пакета `ggplot2`.

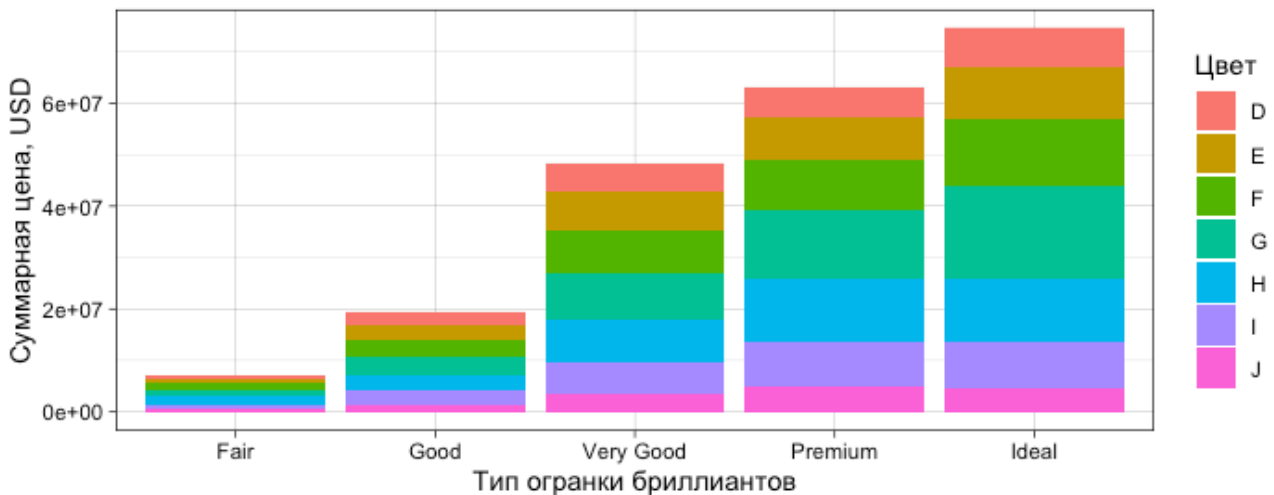
Вариант – 1 (diamonds)

Соотношение цены и веса бриллиантов



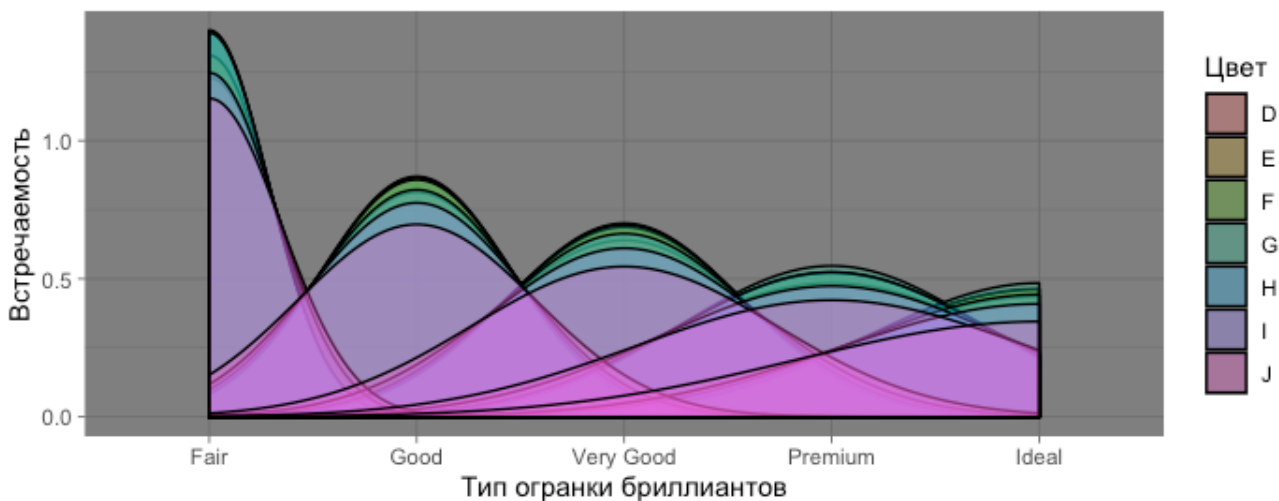
Вариант – 2 (diamonds)

Стоимость бриллиантов в зависимости от огранки



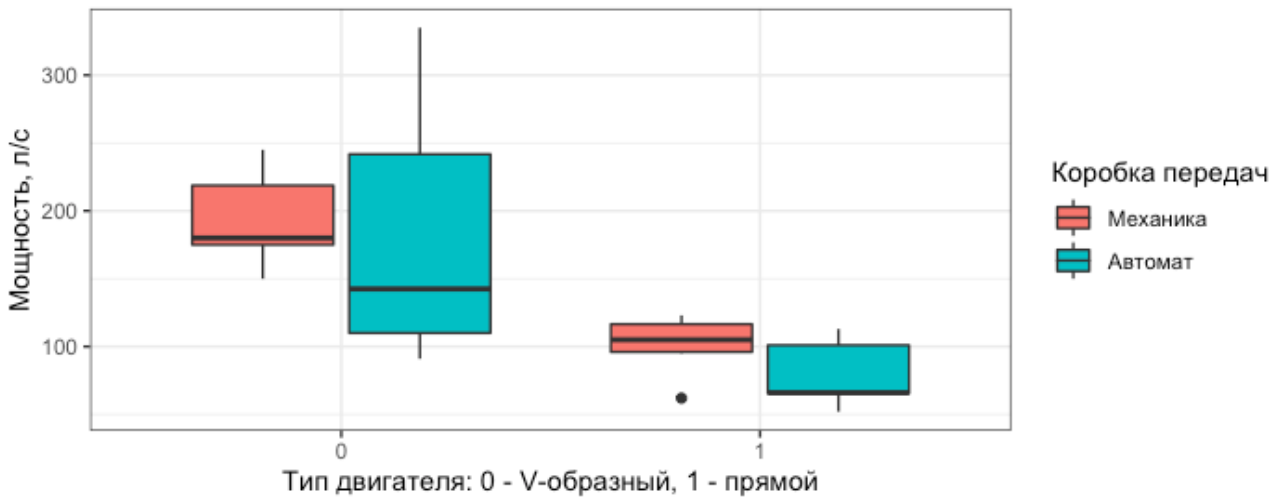
Вариант – 3 (diamonds)

Встречаемость бриллиантов по цветам в зависимости от огранки



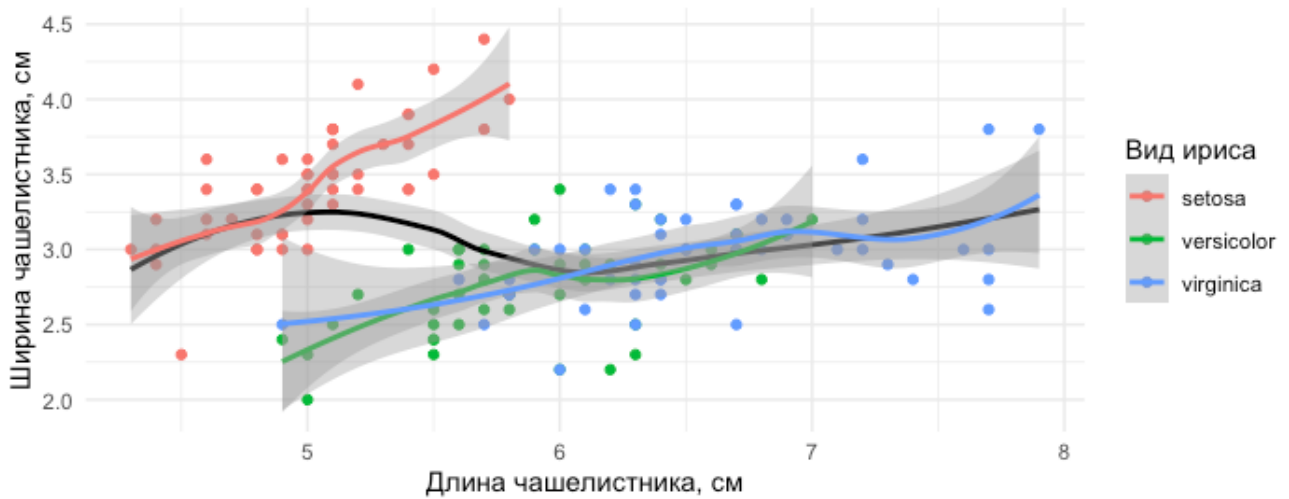
Вариант – 4 (mtcars)

Мощность двигателя в зависимости от типа и коробки передач



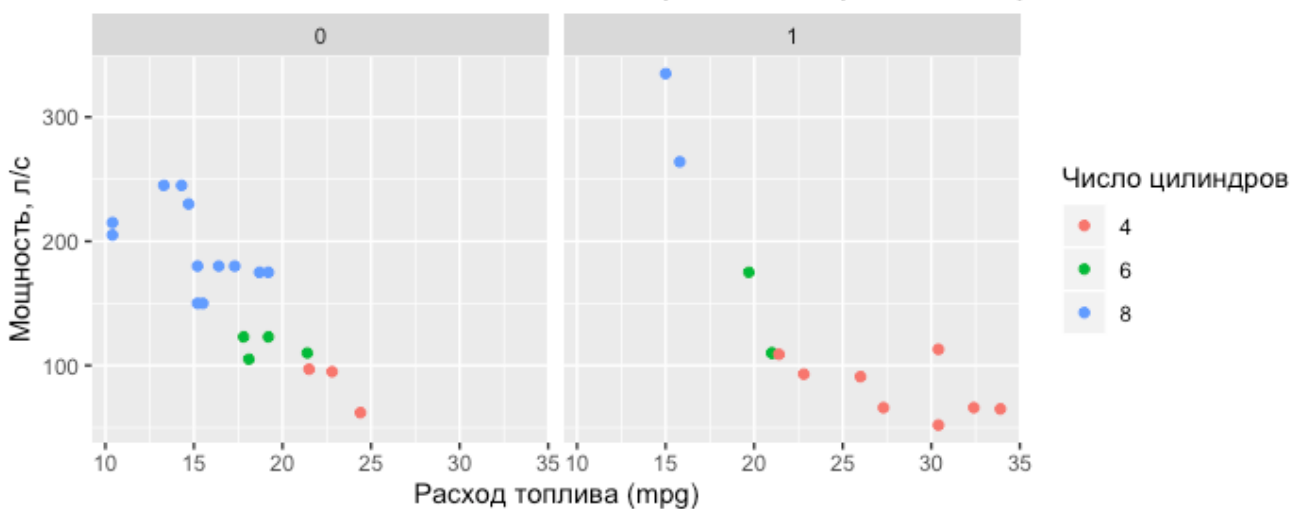
Вариант – 5 (iris)

Связь длины и ширины чашелистника Ирисов Фишера



Вариант – 6 (mtcars)

Расхода топлива и мощность авто с разными коробками передач



Задание 2 – С помощью функции `ggplot` постройте два произвольных графика, показав умение использовать различные элементы грамматики графики, для набора данных по заданному варианту. Дайте описание набору данных и то, что показывает изображенный график. Добавьте интерактивность в графики с использованием функции `ggplotly`. Повторите выполнение графика с помощью функции `plot_ly`.

№	Набор данных	№	Набор данных	№	Набор данных	№	Набор данных
1	CO2	8	Puromycin	15	iris	22	swiss
2	ChickWeight	9	Seatbelts	16	longley	23	trees
3	DNase	10	Theoph	17	mtcars	24	economics
4	LifeCycleSavings	11	ToothGrowth	18	quakes	25	faithfuld
5	Loblolly	12	USArrests	19	rock	26	midwest
6	Orange	13	freeny	20	stack.x	27	mpg
7	OrchardSprays	14	infert	21	stackloss	28	txhousing

Задание 3 – Повторите построение трехмерного чайника. Представьте итоговый скрин изображения.

```
install.packages("readr")
install.packages("plotly")
library(readr)
library(plotly)
teapot <- read_delim("http://0861.ru/r_2019/teapot.csv",
                    ";", escape_double = FALSE, trim_ws = TRUE)
make.fancy.teapot <- function(teapot.coords) {
  is <- (1:(nrow(teapot.coords) / 3)) * 3 - 3
  js <- (1:(nrow(teapot.coords) / 3)) * 3 - 2
  ks <- (1:(nrow(teapot.coords) / 3)) * 3 - 1
  plot_ly(teapot.coords, x = ~x, y = ~y, z = ~z,
          i = is,
          j = js,
          k = ks, type = "mesh3d")
}
```

Контрольные вопросы:

1. Как изменить подпись к легенде?
2. Как изменить подписи к осям координат?
3. Каким образом производится разбивка на фасеты?
4. Каким образом меняется тема оформления графика?
5. Каким образом добавить название к графику?
6. В чем отличие сигнатуры функции plot_ly от ggplot?
7. Каким образом преобразовать исходный статичный график ggplot в интерактивный?
8. За что отвечает аргумент type функции plot_ly?