Прикладная статистика в R

Лекция 2. Графическое представление данных в R.

Елена Михайловна Парилина

д. ф.-м. н., проф.

2021

Основные подходы к визуализации данных в статистике

Графическое представление данных

Первые шаги по визуализации данных:

1. Выстроим элементы выборки по возрастанию (неубыванию):

$$X_{(1)} \leqslant X_{(2)} \leqslant \ldots \leqslant X_{(n)}.$$

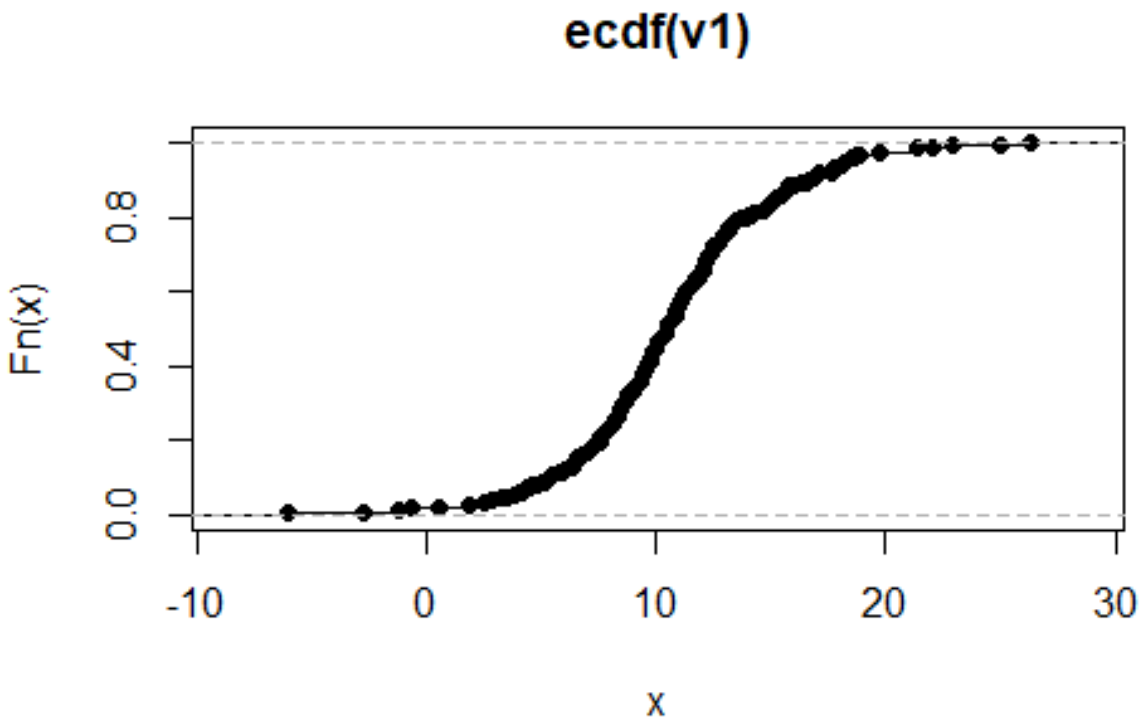
2. Величины $X_{(1)}$, $X_{(2)}$, ..., $X_{(n)}$ образуют *вариационный ряд*.
3. Если предположить, что все элементы вариационного ряда различны, то есть $X_{(1)} < X_{(2)} < \ldots < X_{(n)}$, то можно определить эмпирическую функцию распределения следующим образом:

$$F_n^*(x) = \begin{cases} 0, & \text{если } x < X_{(1)}; \\ \frac{1}{n}, & \text{если } X_{(1)} \leqslant x < X_{(2)}; \\ \frac{2}{n}, & \text{если } X_{(2)} \leqslant x < X_{(3)}; \\ \ldots & \\ \frac{k}{n}, & \text{если } X_{(k)} \leqslant x < X_{(k+1)}; \\ \ldots & \\ 1, & \text{если } x \geqslant X_{(n)}. \end{cases} \tag{1}$$

Данные по доходности фондов в R

Пусть имеется выборка из 259 элементов, содержащая 5-летние доходности фондов.

```
View(Mutual_Funds)
v1<-Mutual_Funds$‘Пятилетняя доходность‘
sort(v1)
plot(ecdf(v1))
```



Продолжаем визуализацию данных

4. Имея вариационный ряд, можно построить *гистограмму*. Возьмем интервал (a, b), где $a < X_{(1)}$ и $X_{(n)} < b$, разобьем этот интервал на непересекающиеся промежутки:

$$a_0 = a < a_1 < a_2 < \ldots < a_m = b,$$

$$(a_{i-1}, a_i], i = 1, \ldots, m.$$

Пусть n_i — количество элементов выборки, попавших в полуинтервал $(a_{i-1}, a_i]$. Тогда

$$n_1 + n_2 + \ldots + n_m = n,$$

$$l_i = a_i - a_{i-1},$$

$$h_i = \frac{n_i}{l_i n}.$$

Продолжаем визуализацию данных

5. Получаем гистограмму:

$$f_n^*(x) = \begin{cases} 0, & \text{если } x \leqslant a_0; \\ h_1, & \text{если } a_0 < x \leqslant a_1; \\ \dots & \\ h_m, & \text{если } a_{m-1} < x \leqslant a_m; \\ 0, & \text{если } x > a_m. \end{cases}$$

Гистограмма $f_n^*(x)$ — эмпирический (построенный по выборке) аналог плотности распределения.

6. Если в знаменателе при вычислении h_i убрать l_i, получится *гистограмма относительных частот*, если, кроме того, в знаменателе убрать n, то получится *гистограмма частот* n_i. Часто при построении гистограммы полагают $l_i = l = const$.

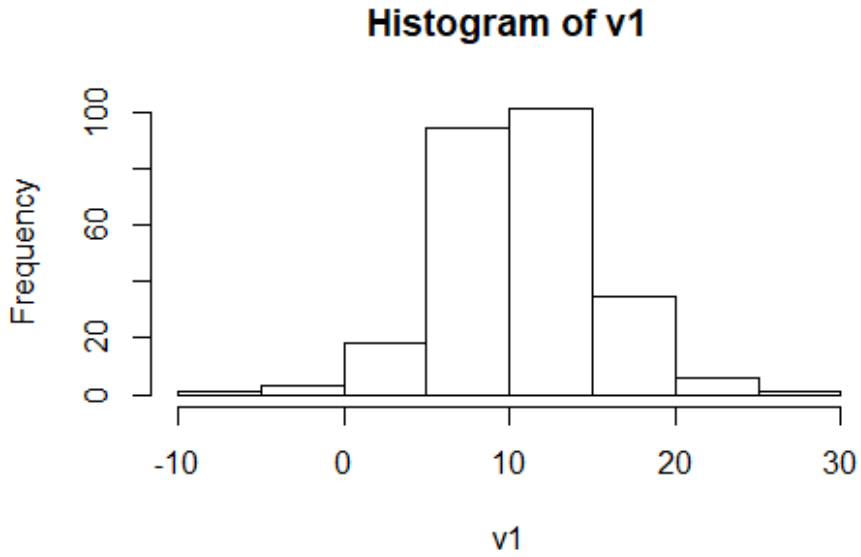
Продолжаем визуализацию данных

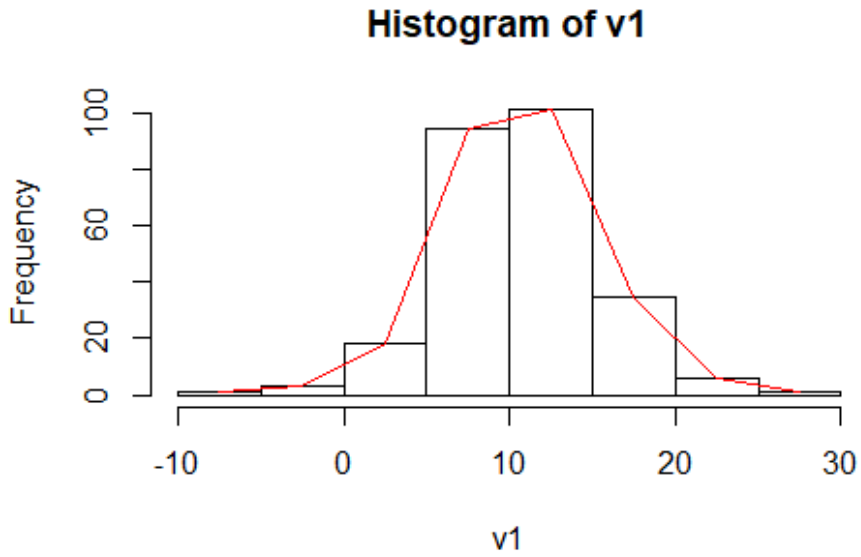
7. Для наглядного представления выборки применяют также *полигон частот*. Полигоном частот называется ломаная с вершинами в точках $(X_i, n_i/b)$, где $b = R/k$, а полигоном относительных частот — ломаная с вершинами в точках $(X_i, n_i/(nb))$.

Данные по доходности фондов в R

```
hist(v1)
h1<-hist(v1)
lines(h1$counts ~ h1$mids, col="red")
```

Можно вывести объект h1, он содержит сведения о границах ($breaks) и серединах ($mids) интервалов, на которые разбиваются исходные данные, частоте ($counts) и относительной частоте ($density) наблюдений на каждом интервале.


Histogram of v1
Frequency
0
20
60
100
-10
0
10
20
30
v1


Histogram of v1
Frequency
0
20
60
100
-10
0
10
20
30
v1

Подробнее о гистограмме в R

Данные по доходности акций в R

```
hist(x = v1,
main = "5-летняя доходность фондов" , # название графика
xlab = "доходность в процентах" , # название оси OX
ylab="Частота" , # название оси OY
border = "gray20" , # установить цвет границ столбцов
col = "gray80" , # установить цвет тени
labels = TRUE, # указывать частоту к каждом столбце
ylim = c(0,120) # изменить шкалу OY
)
```

Подробнее о гистограмме в R

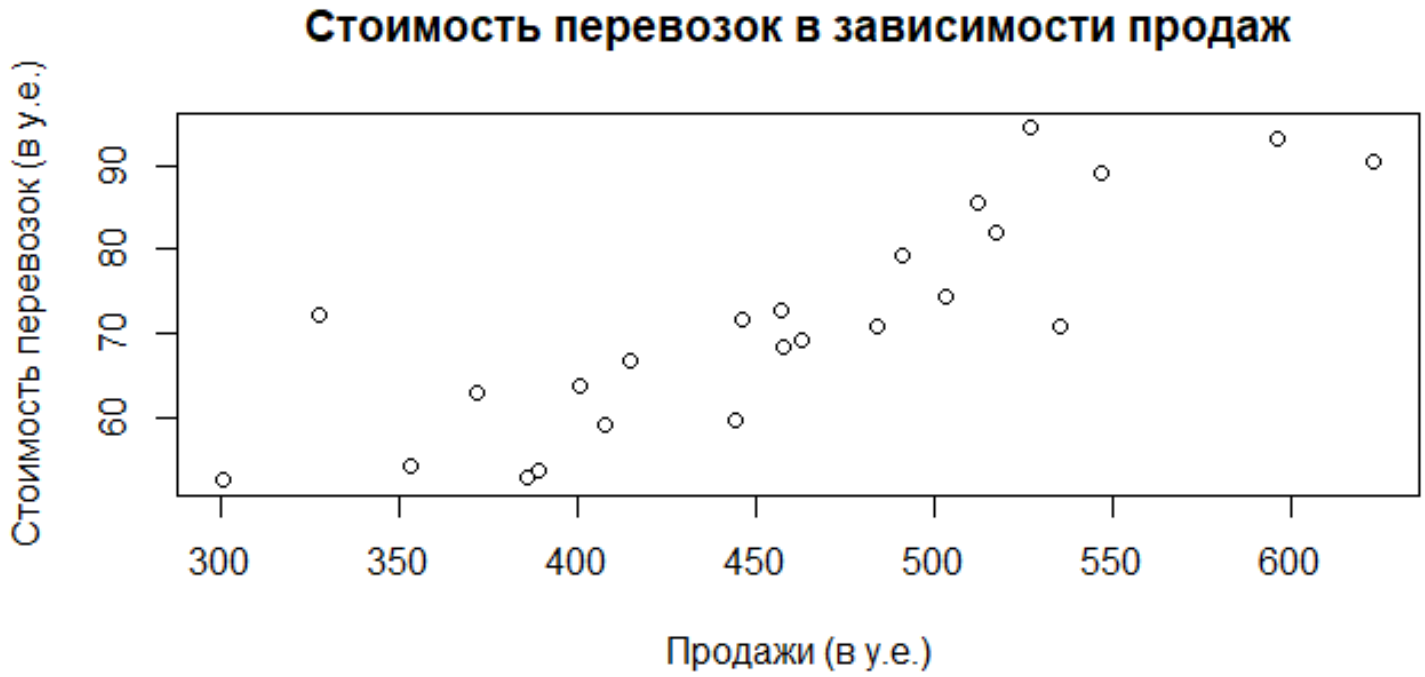
Графическое представление многомерных данных

Данные о стоимости перевозок

Дана выборка наблюдений о стоимости перевозок при заданном объеме продаж и количестве заказов.

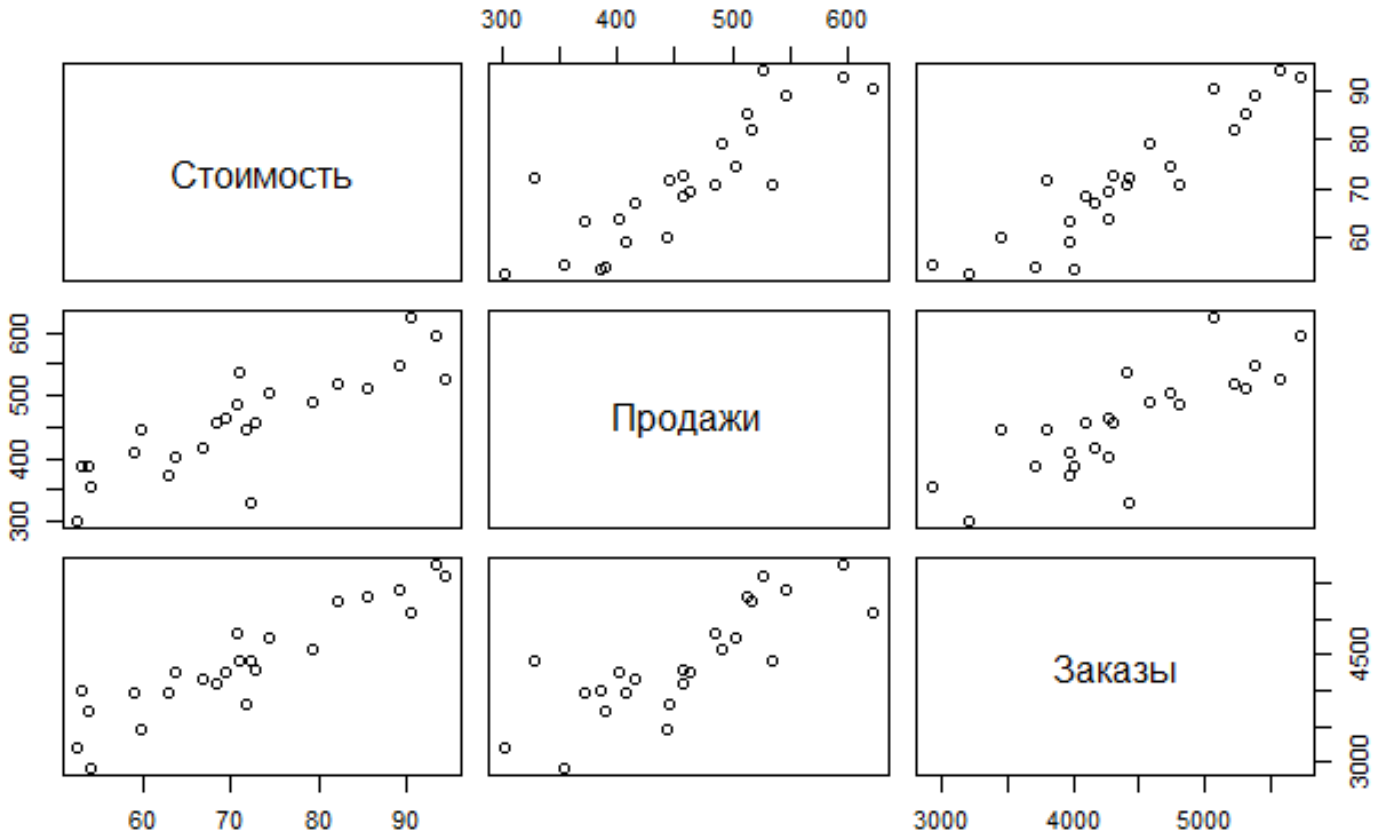
Данные стоимости перевозок в R

```
plot(x = WARECOST$Продажи,
y = WARECOST$Стоимость,
main = "Стоимость перевозок в зависимости продаж" , # название
графика
xlab = "Продажи" , # название оси OX
ylab="Стоимость" , # название оси OY
)
```


Стоимость перевозок в зависимости продаж
Стоимость перевозок (в у.е.)
60
70
80
90
300
350
400
450
500
550
600
Продажи (в у.е.)

Попарное изображение данных выборки

```
pairs(WARECOST)
```



Графический анализ зависимостей

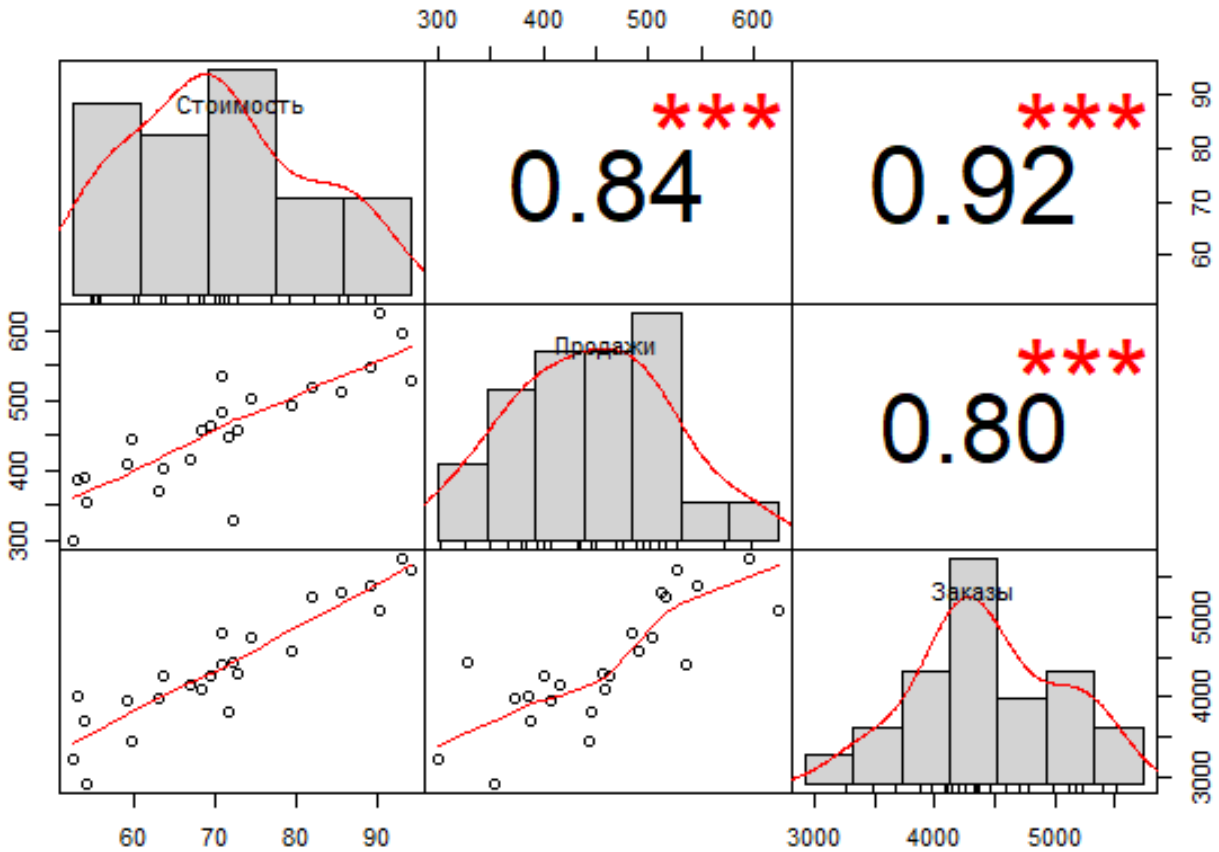
Анализ зависимостей (корреляция данных)

```
install.packages("PerformanceAnalytics")
library("PerformanceAnalytics")
chart.Correlation(WARECOST, histogram=TRUE, pch=10)
```

Пакеты для визуализации корреляции данных

```
install.packages("GGally")
install.packages("corrgram")
install.packages("ellipse")
```

Анализ зависимостей в R



Над диагональю выписаны коэффициенты корреляции, а звездочки указывают на значимость корреляции (символы “***”, “**”, “*”, “.” говорят о статистической значимости для разных уровней).

Диаграммы рассеяния и гистограммы по категориям

Диаграмма рассеяния (scatterplot)

```
> gf_point(Sepal.Length ~ Sepal.Width, data=iris)
> gf_point(Sepal.Length ~ Sepal.Width | Species, data=iris)
```

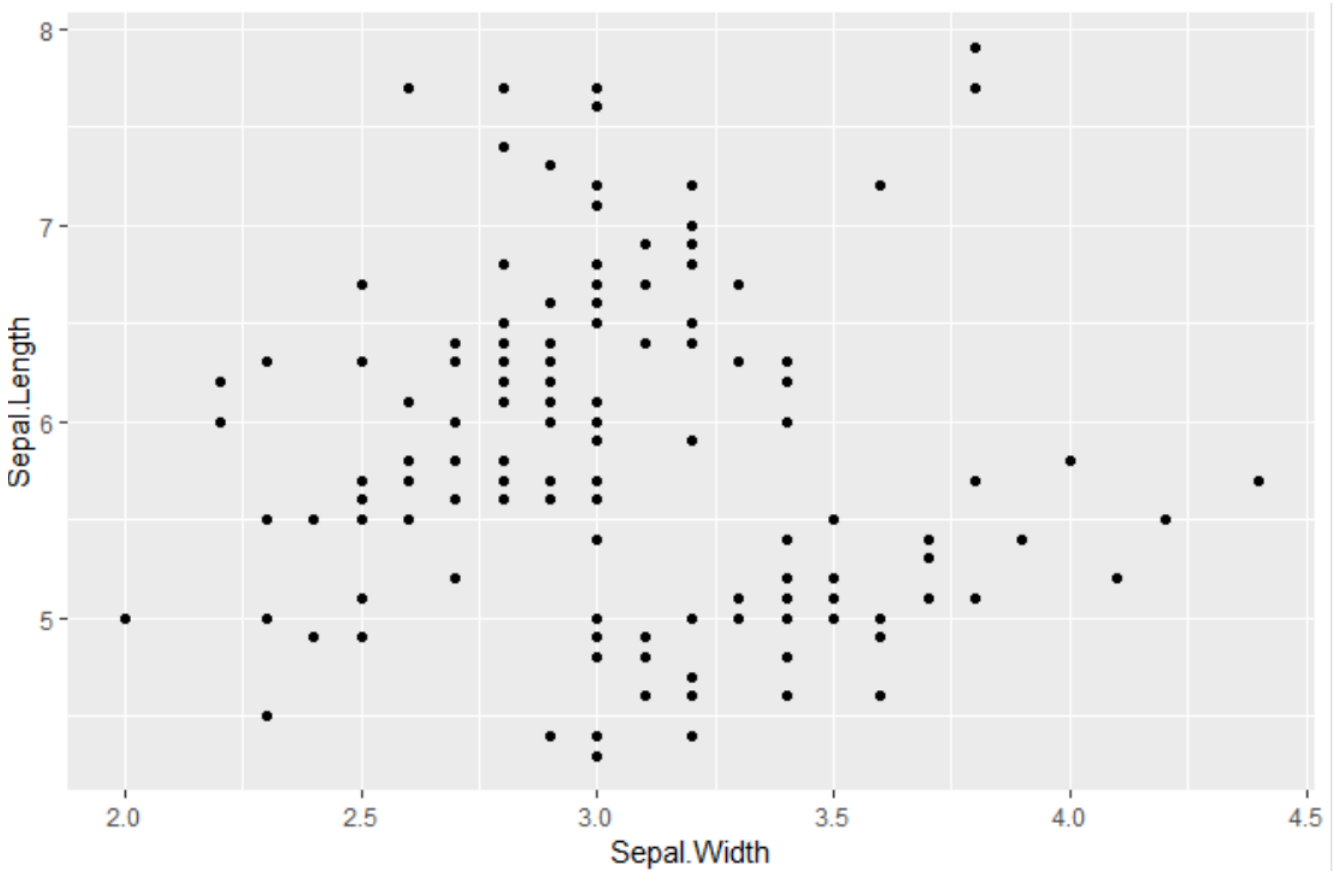


Диаграмма рассеяния (scatterplot)

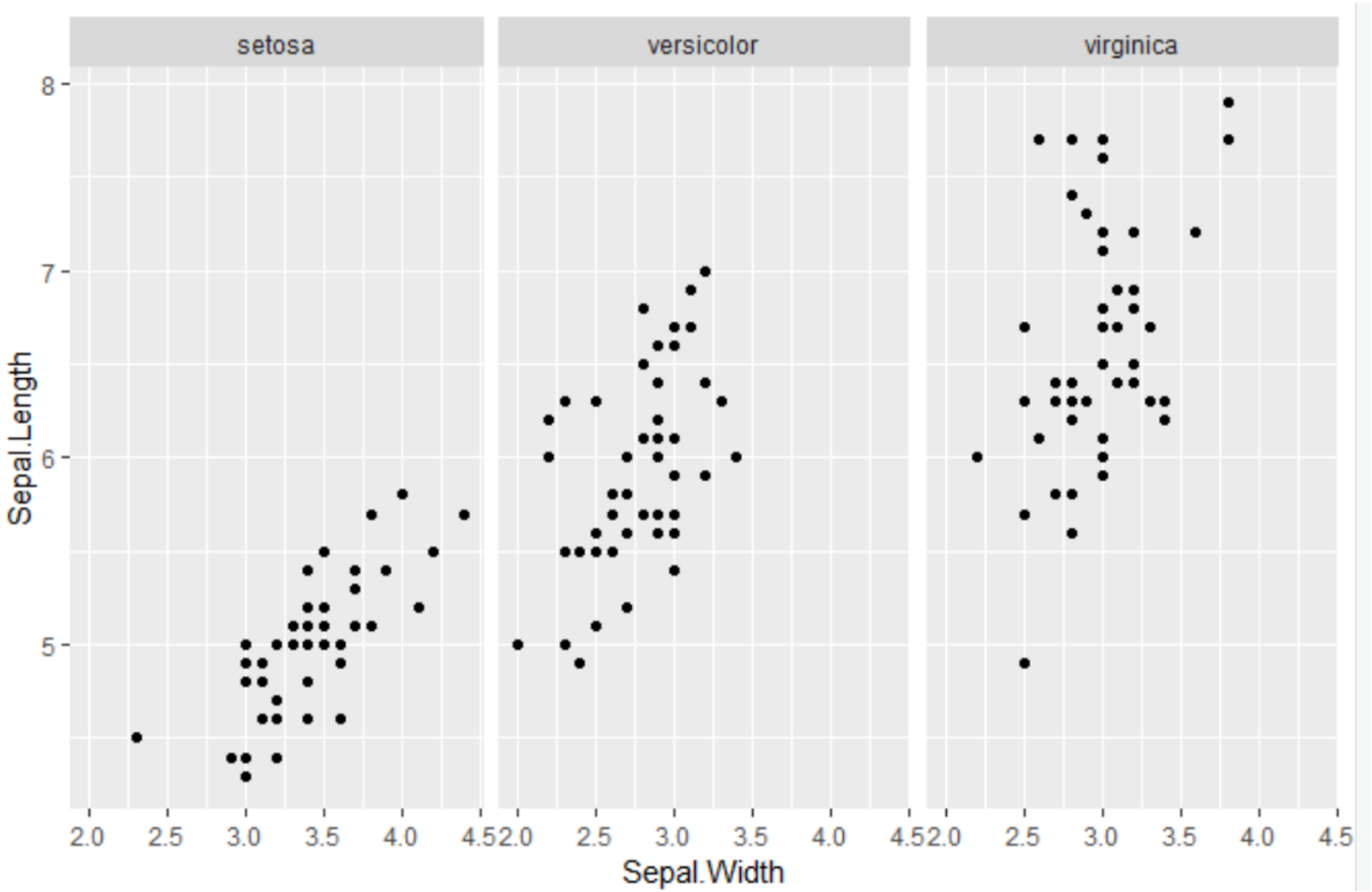


Диаграмма рассеяния (scatterplot)

```
> gf_point(Sepal.Length ~ Sepal.Width, data=iris, color=~Species, shape=~Species)
```

Разбиение выборки на интервалы

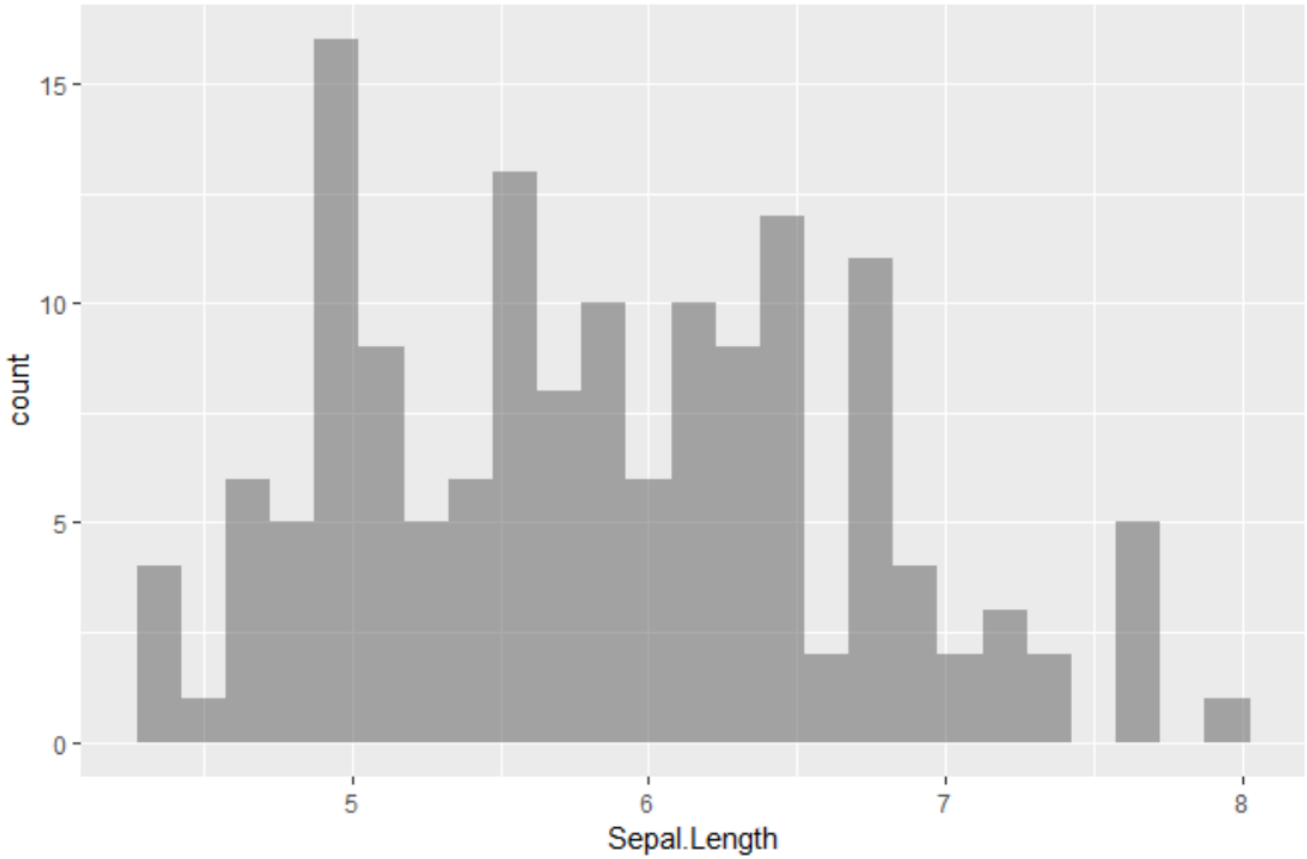
`tally`: создает таблицу значений `iris`;
`cut`: разбивает на интервалы и подсчитывает количество элементов в каждом интервале;

```
> tally(~Species, data=iris)
Species
    setosa versicolor  virginica
        50         50         50
> tally(~cut(Sepal.Length, breaks=2:10), data=iris)
cut(Sepal.Length, breaks = 2:10)
 (2,3]  (3,4]  (4,5]  (5,6]  (6,7]  (7,8]  (8,9] (9,10]
     0      0     32     57     49     12      0      0
> tally(~cut(Sepal.Length, breaks=2:10,right=FALSE), data=iris)
cut(Sepal.Length, breaks = 2:10, right = FALSE)
 [2,3)  [3,4)  [4,5)  [5,6)  [6,7)  [7,8)  [8,9) [9,10)
     0      0     22     61     54     13      0      0
```

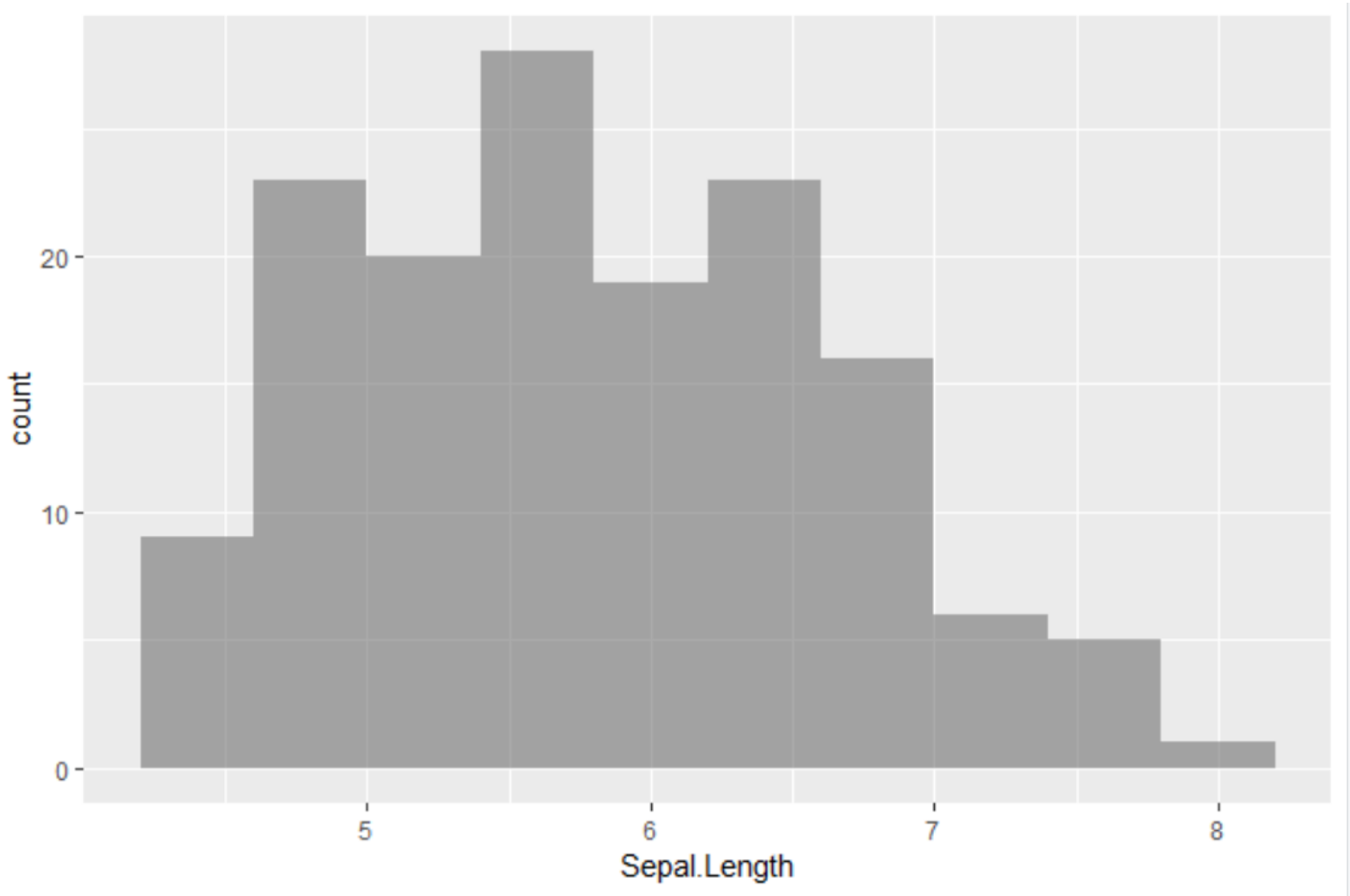
Гистограммы

```
> gf_histogram(~Sepal.Length, data=iris)
> gf_histogram(~Sepal.Length, data=iris,bins=10)
> gf_histogram(~Sepal.Length, data=iris,breaks=c(4,5,5.5,6,6.5,7,8,10),color="black",fill="blue")
> gf_dhistogram(~Sepal.Length, data=iris,breaks=c(4,5,5.5,6,6.5,7,8,10),color="black",fill="blue")
```

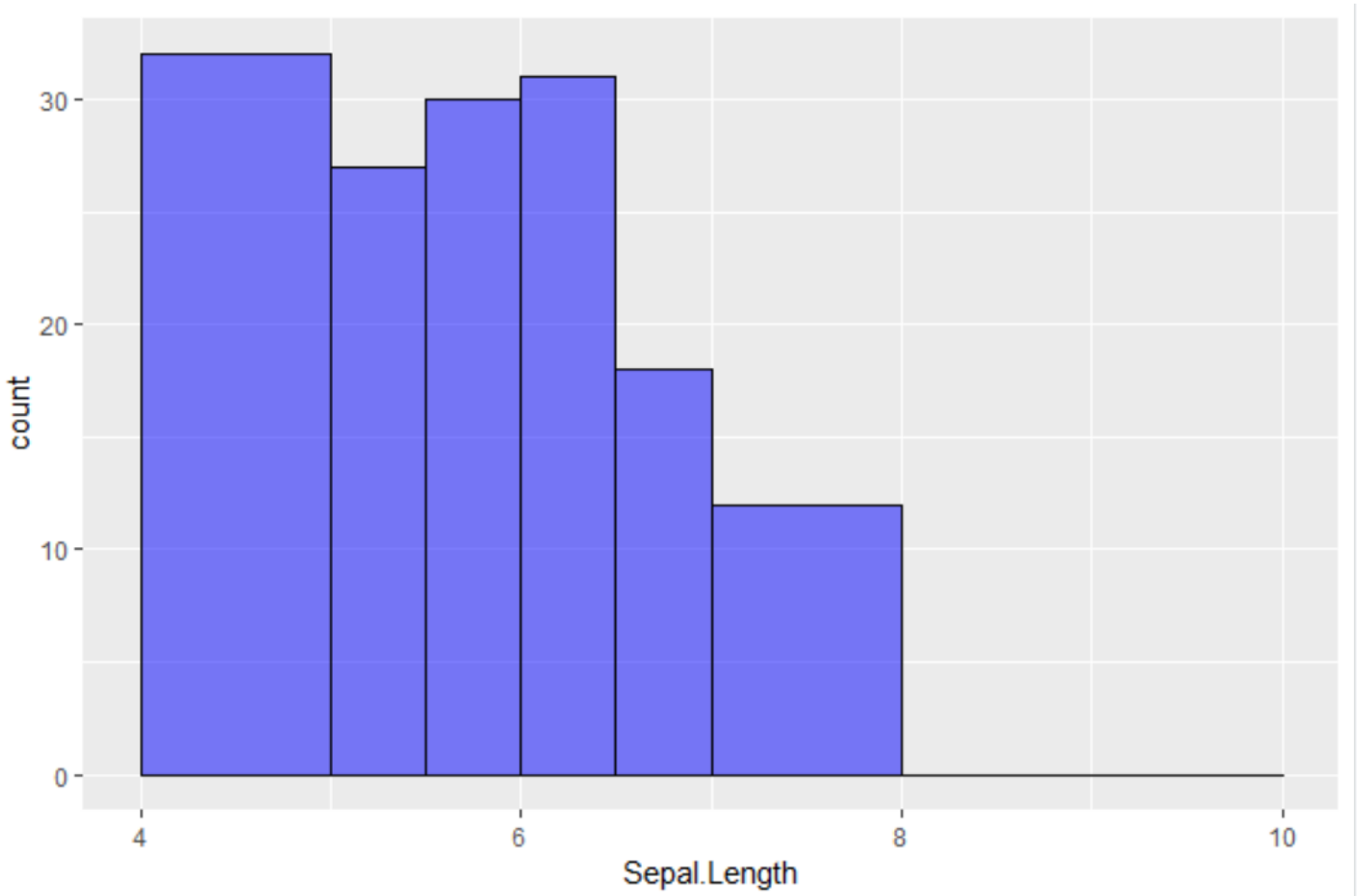
Ниже приведены 4 рисунка с соответствующими гистограммами:



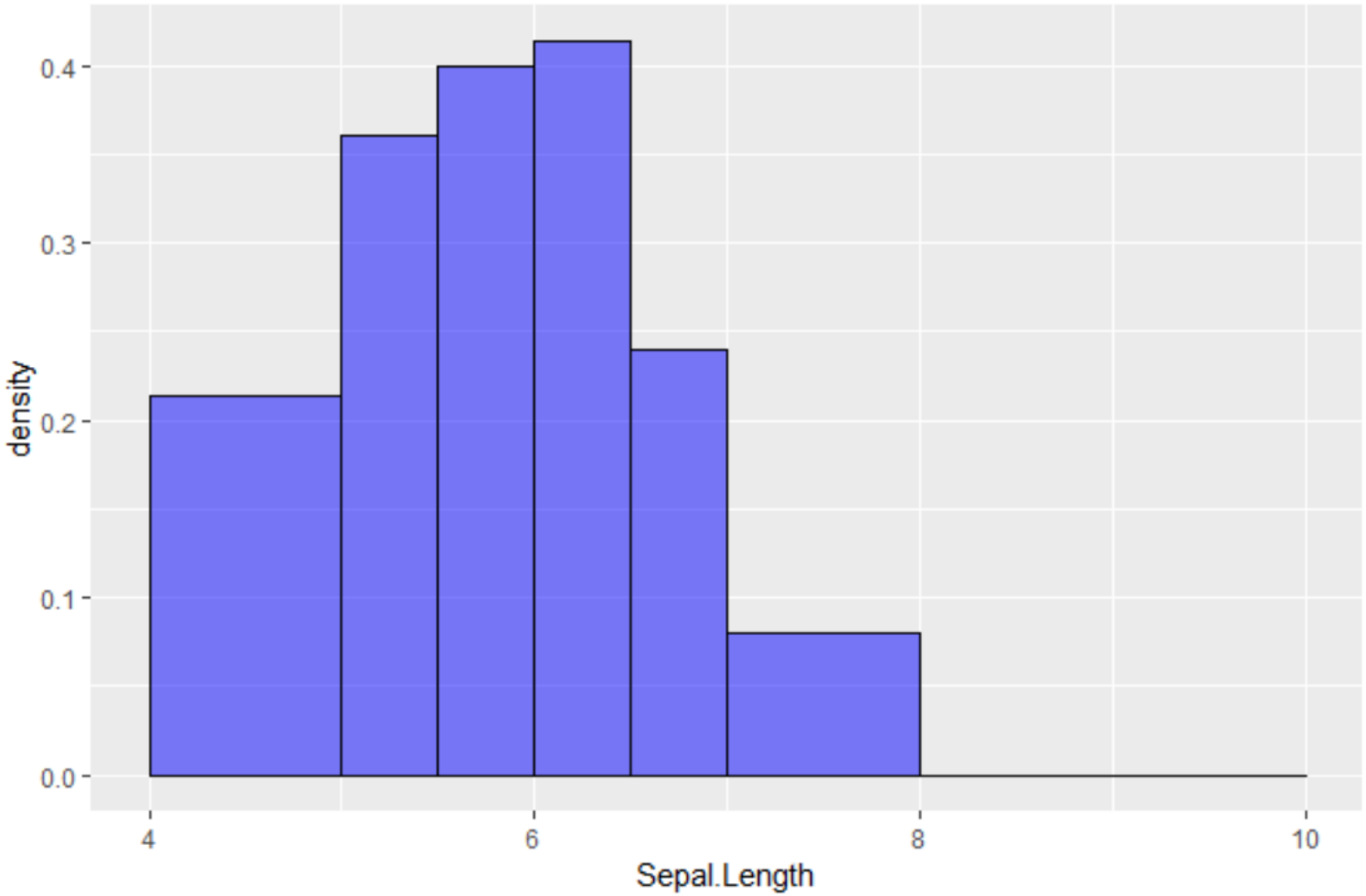
Гистограммы



Гистограммы

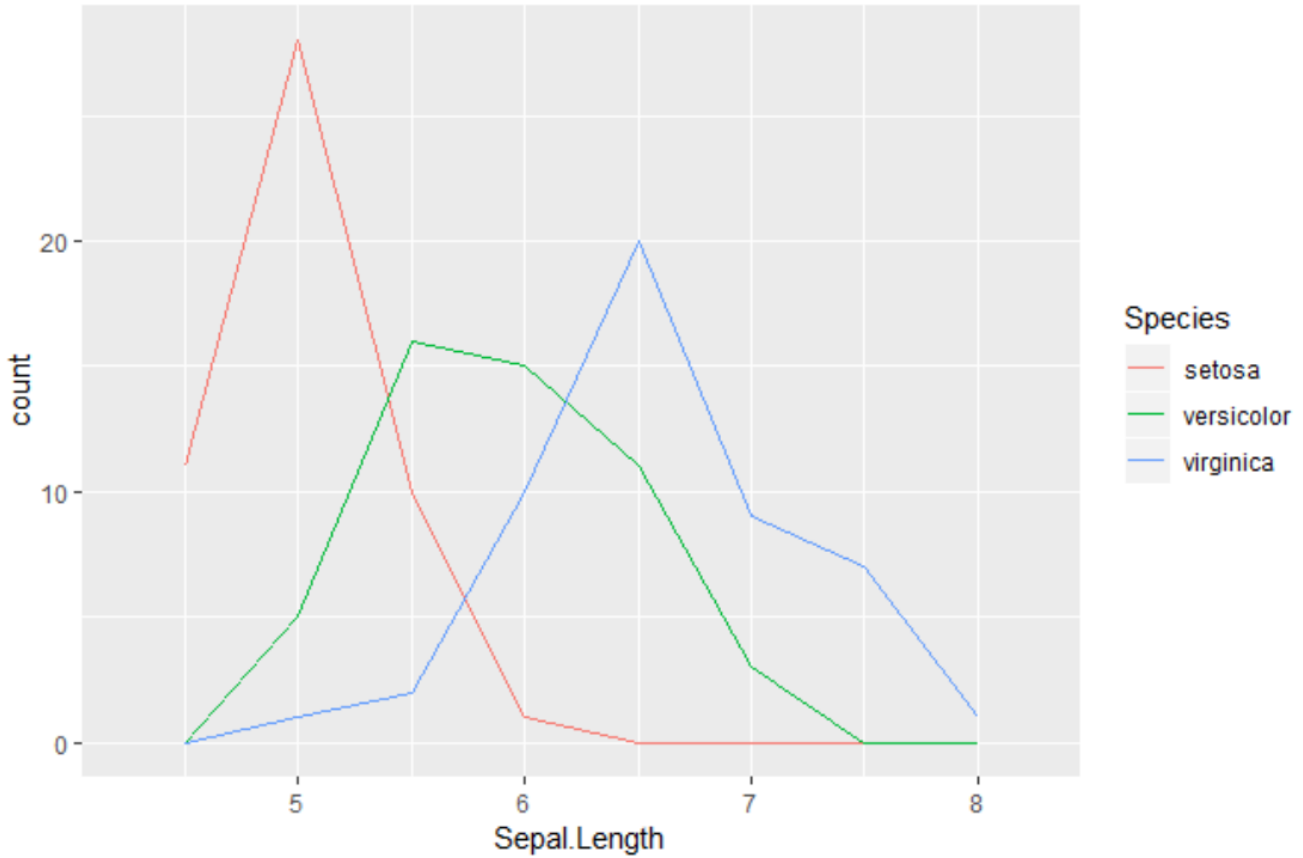


Гистограммы



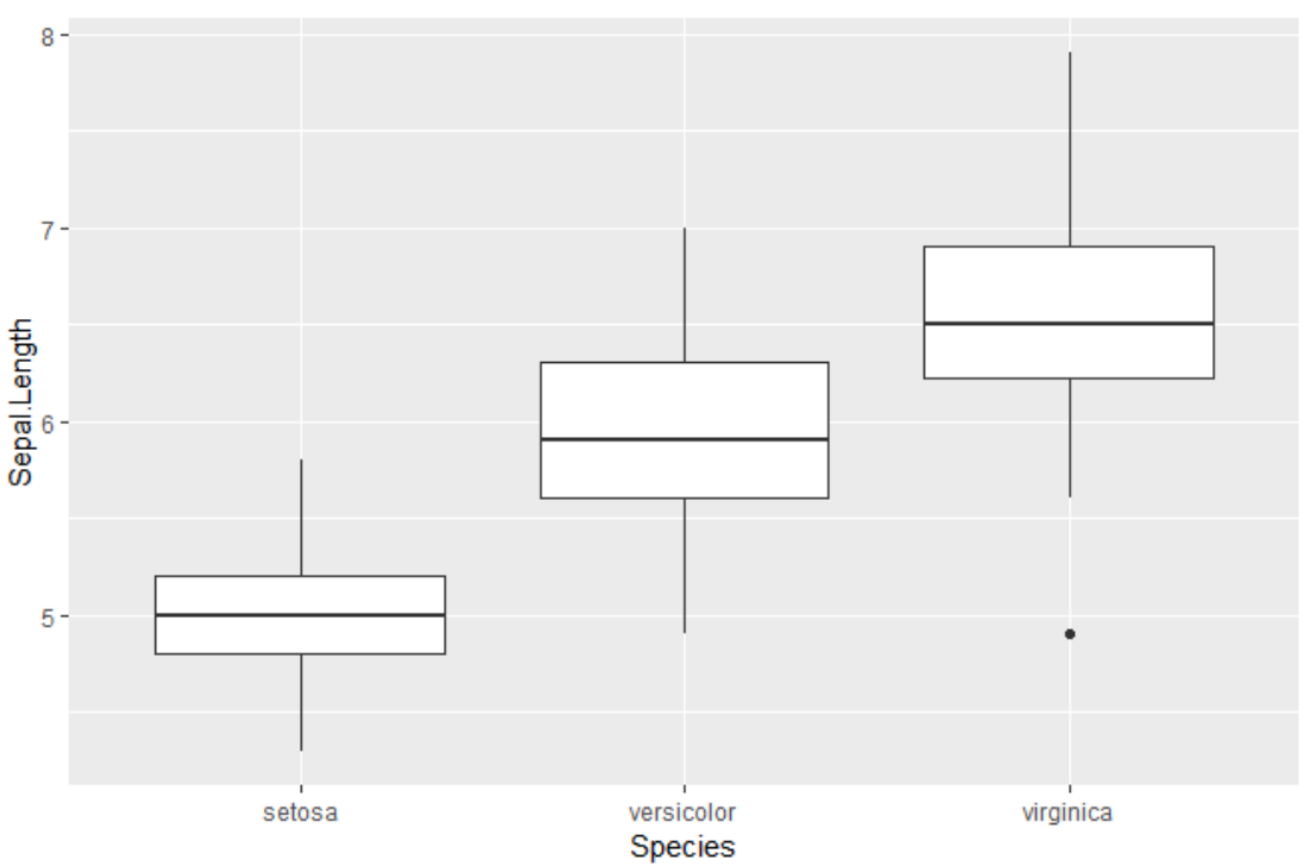
Полигон частот

```
> gf_freqpoly(~Sepal.Length, color=~Species, data=iris, binwidth = 0.5)
```



"Ящик с усами"(boxplot)

```
> gf_boxplot(Sepal.Length~Species, data=iris)
```



"Ящик с усами" (boxplot)

- Границами ящика служат первый и третий квартили (25-й и 75-й процентили соответственно), линия в середине ящика — медиана (50-й процентиль). Концы усов — края статистически значимой выборки (без выбросов), и они могут определяться несколькими способами. Наиболее распространённые значения, определяющие длину «усов»:
  1. Минимальное и максимальное наблюдаемые значения данных по выборке (в этом случае выбросы отсутствуют).
  2. Разность первого квартиля и полутора межквартильных расстояний:

$$X_1 = Q_1 - k(Q_3 - Q_1),$$

сумма третьего квартиля и полутора межквартильных расстояний:

$$X_2 = Q_3 + k(Q_3 - Q_1),$$

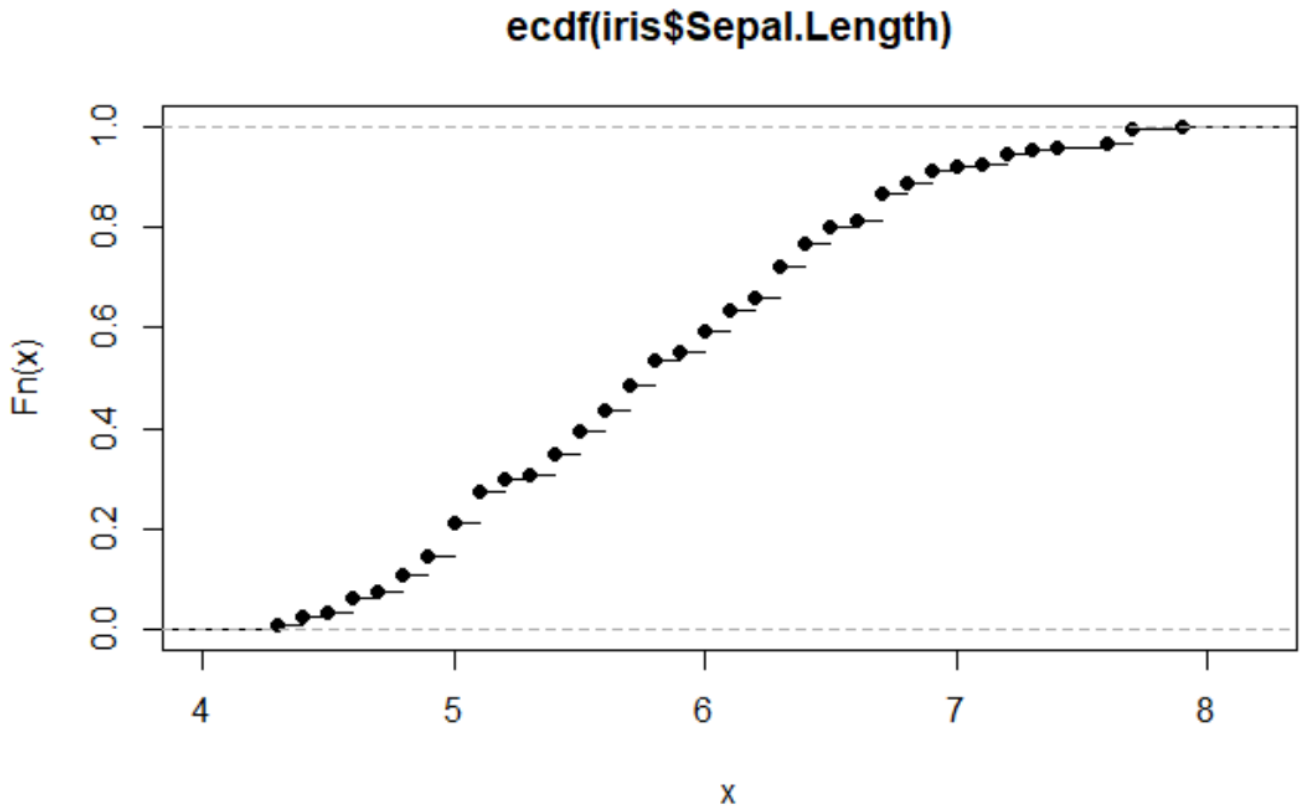
где X_1 — нижняя граница уса, X_2 — верхняя граница уса, Q_1 — первый квартиль, Q_3 — третий квартиль, k — коэффициент, наиболее часто употребляемое значение которого равно 1,5.

"Ящик с усами" (boxplot)

- Данные, выходящие за границы усов (выбросы), отображаются на графике в виде точек, маленьких кружков или звёздочек. Иногда на графике отмечают среднее арифметическое и его доверительный интервал («зарубка» на ящике).
- В связи с тем, что не существует единого общего согласия относительно того, как конкретно строить «ящик с усами», при виде такого графика необходимо искать информацию в документации.

`hist(iris$Sepal.Length)`: гистограмма `iris$Sepal.Length`;
`boxplot(iris$Sepal.Length)`: boxplot `iris$Sepal.Length`;

`ecdf(iris$Sepal.Length)`: эмпирическая функция распределения переменной `iris$Sepal.Length`
`plot(ecdf(iris$Sepal.Length))`: график эмпирической функции распределения переменной `iris$Sepal.Length`



Графическое представление временных рядов

Пример

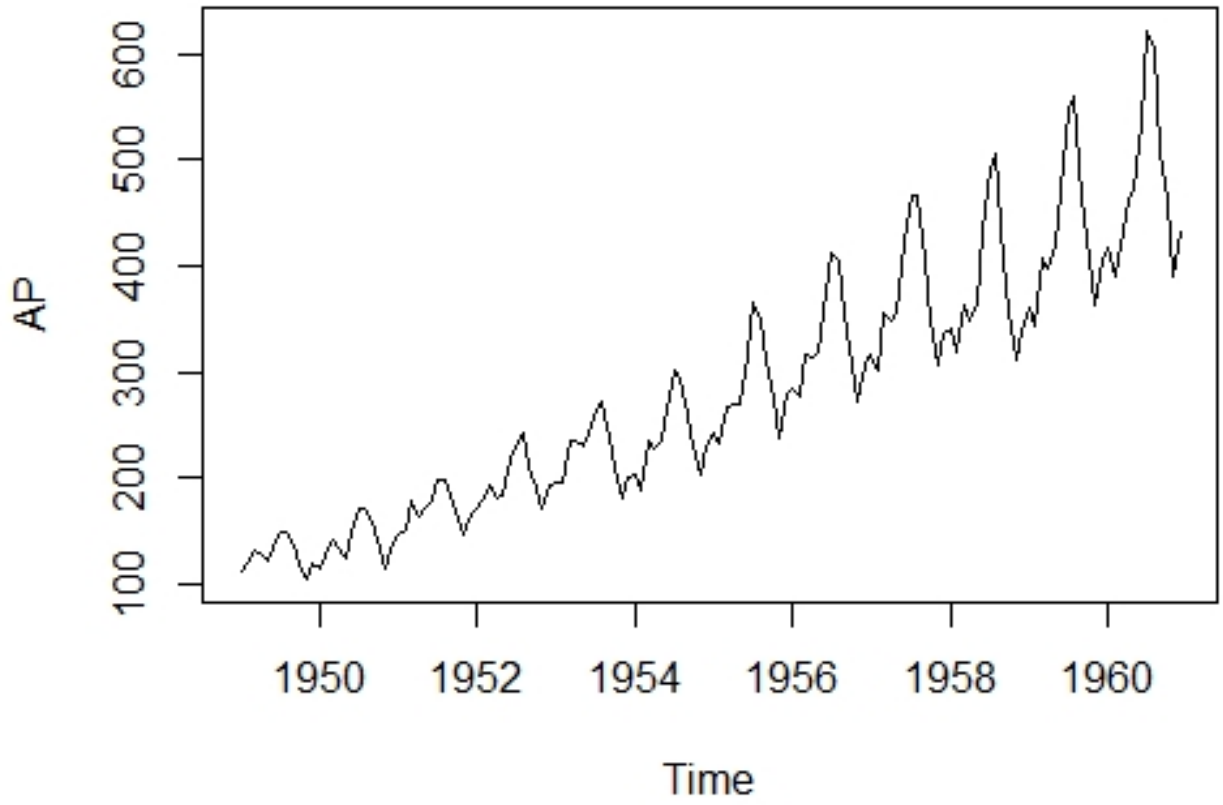
Количество международных пассажирских перевозок авиакомпании Pan Am (в тысячах) за месяц на территории Соединенных Штатов Америки было получено от Федерального управления гражданской авиации за период 1949–1960 годов (Brown, 1963). Компания использовала данные для прогнозирования будущего спроса, прежде чем заказывать новые самолеты и тренировать экипаж.

Данные доступны в виде временных рядов в R и иллюстрируют несколько важных концепций, возникающих при исследовательском анализе временных рядов.

Функции в R

```
data(AirPassengers)
AP <- AirPassengers
AP
```

```
> plot(AP)
> data(AirPassengers)
> AP <- AirPassengers
> AP
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
1949 112 118 132 129 121 135 148 148 136 119 104 118
1950 115 126 141 135 125 149 170 170 158 133 114 140
1951 145 150 178 163 172 178 199 199 184 162 146 166
1952 171 180 193 181 183 218 230 242 209 191 172 194
1953 196 196 236 235 229 243 264 272 237 211 180 201
1954 204 188 235 227 234 264 302 293 259 229 203 229
1955 242 233 267 269 270 315 364 347 312 274 237 278
1956 284 277 317 313 318 374 413 405 355 306 271 306
1957 315 301 356 348 355 422 465 467 404 347 305 336
1958 340 318 362 348 363 435 491 505 404 359 310 337
1959 360 342 406 396 420 472 548 559 463 407 362 405
1960 417 391 419 461 472 535 622 606 508 461 390 432
>
```



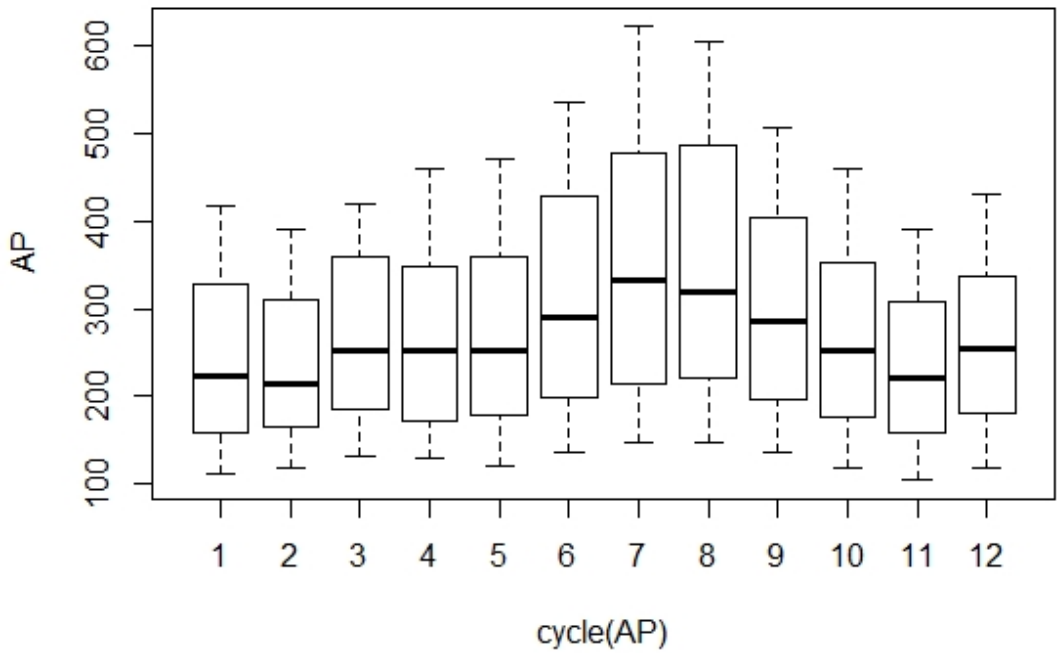
Здесь наблюдается четкий возрастающий тенденция. Существует также сильная сезонная закономерность, размер которой увеличивается по мере увеличения уровня ряда.

Ящики с усами (Boxplot) для временного ряда

Сезонные эффекты можно увидеть на графике (больше людей путешествовали в летние месяцы с июня по сентябрь).

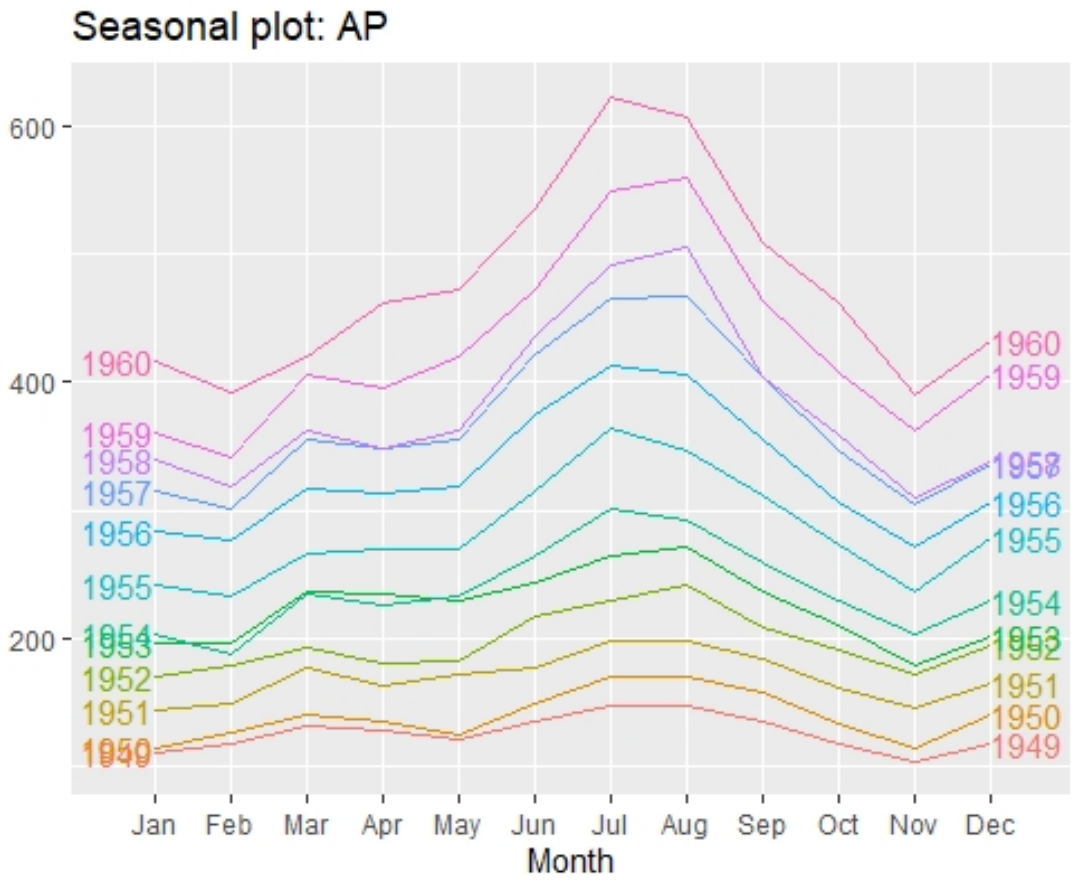
Функции в R

```
boxplot(AP~ cycle(AP))
```



Функции в R

```
install.packages("forecast")
library(forecast)
ggseasonplot(AP, year.labels=TRUE, year.labels.left=TRUE)
```



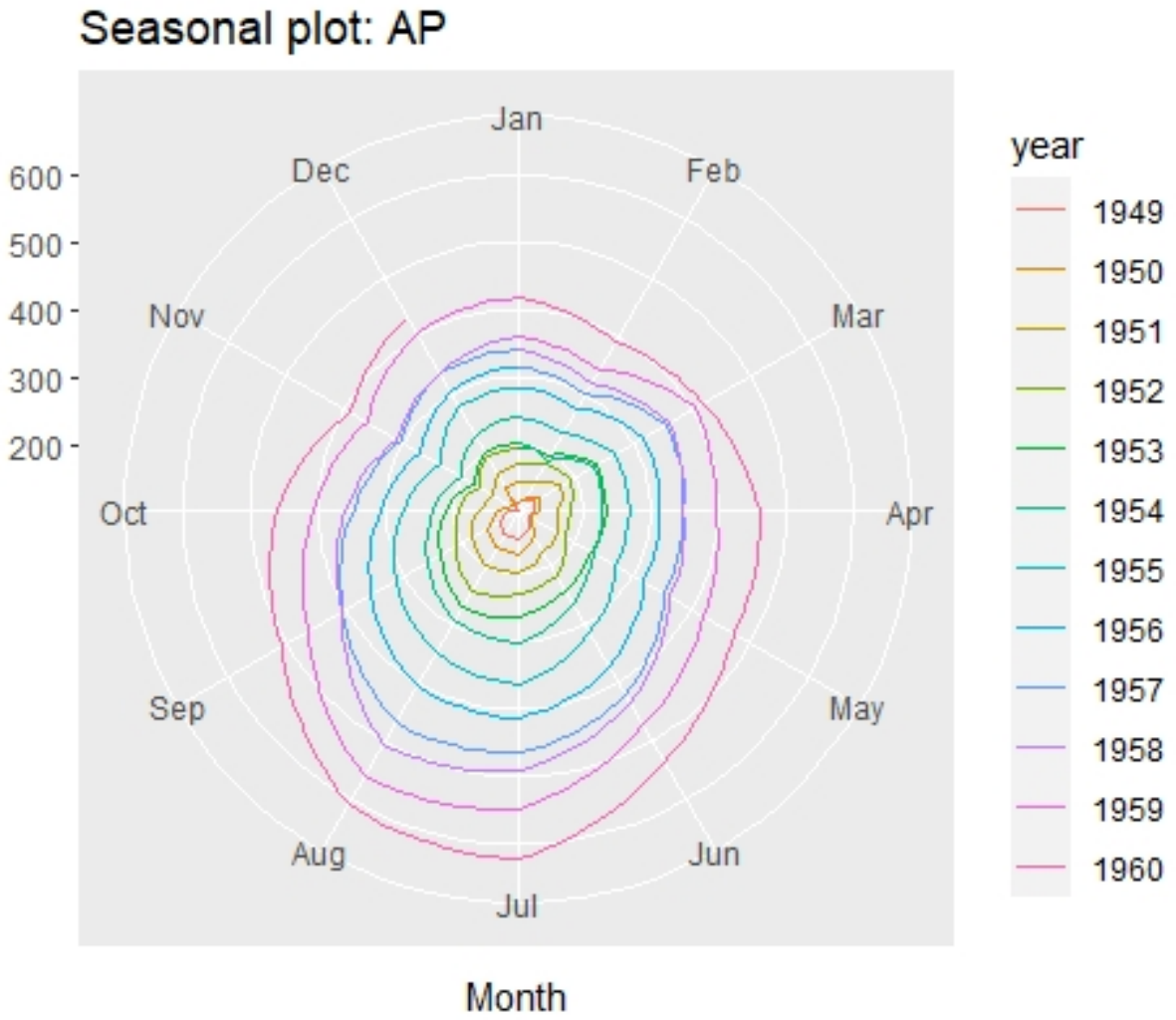
Это в точности те же данные, которые были показаны ранее. Сезонный график позволяет более четко увидеть лежащую в основе сезонную модель и особенно полезен для определения периодов, в которые она меняется.

Вопрос

Какие выводы мы можем сделать, исходя из этого графика?

Полезный вариант сезонного графика использует полярные координаты. Функция `polar = TRUE` делает ось временного ряда круговой, а не горизонтальной, как показано ниже.

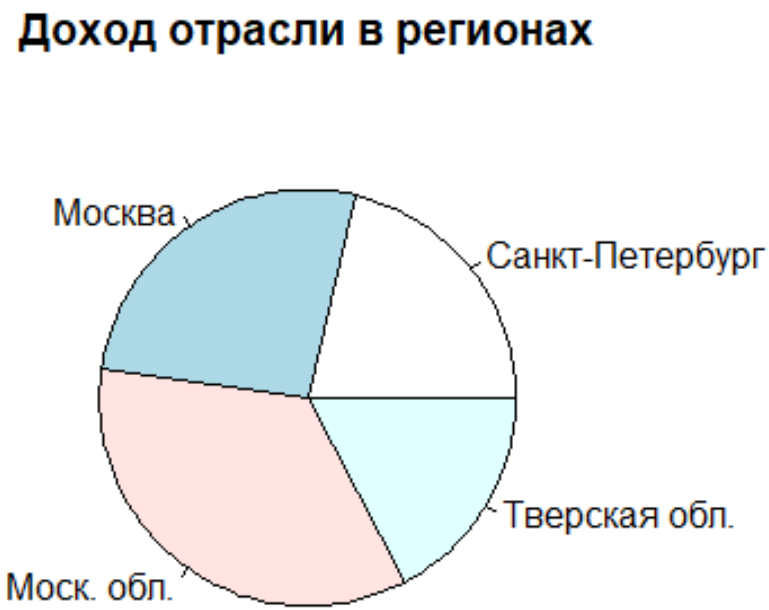
Круговые диаграммы временных рядов



Круговые диаграммы

Функции в R

```
slices <- c(10, 12.4, 16.4, 8.1)
lbls <- c("Санкт-Петербург" , "Москва" , "Моск. обл." , "Тверская
обл.")
pie(slices, labels = lbls, main="Доход отрасли в регионах")
```

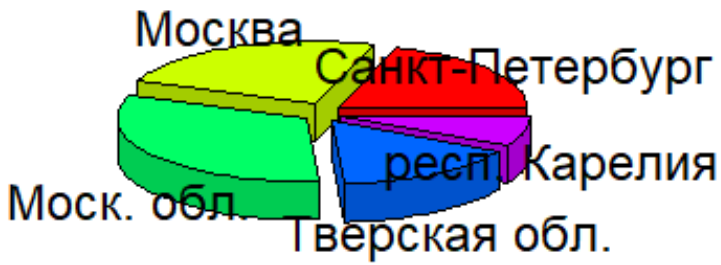


Функции в R (Круговые диаграммы с процентами)

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c("Санкт-Петербург" , "Москва" , "Моск. обл." , "Тверская
обл." , "Респ. Карелия")
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct) # вычисляем проценты lbls <-
paste(lbls,"%" , sep=" ") # добавляем проценты к меткам
pie(slices,labels = lbls, col=rainbow(length(lbls)), main="Доход
отрасли в регионах")
```

Функции в R (Круговые диаграммы в 3D)

```
install.packages("plotrix")
library(plotrix)
slices <- c(10, 12, 16, 8, 4)
lbls <- c("Санкт-Петербург" , "Москва" , "Моск. обл." , "Тверская
обл." , "Респ. Карелия")
pie3D(slices,labels=lbls,explode=0.1, main="Доход отрасли в
регионах")
```



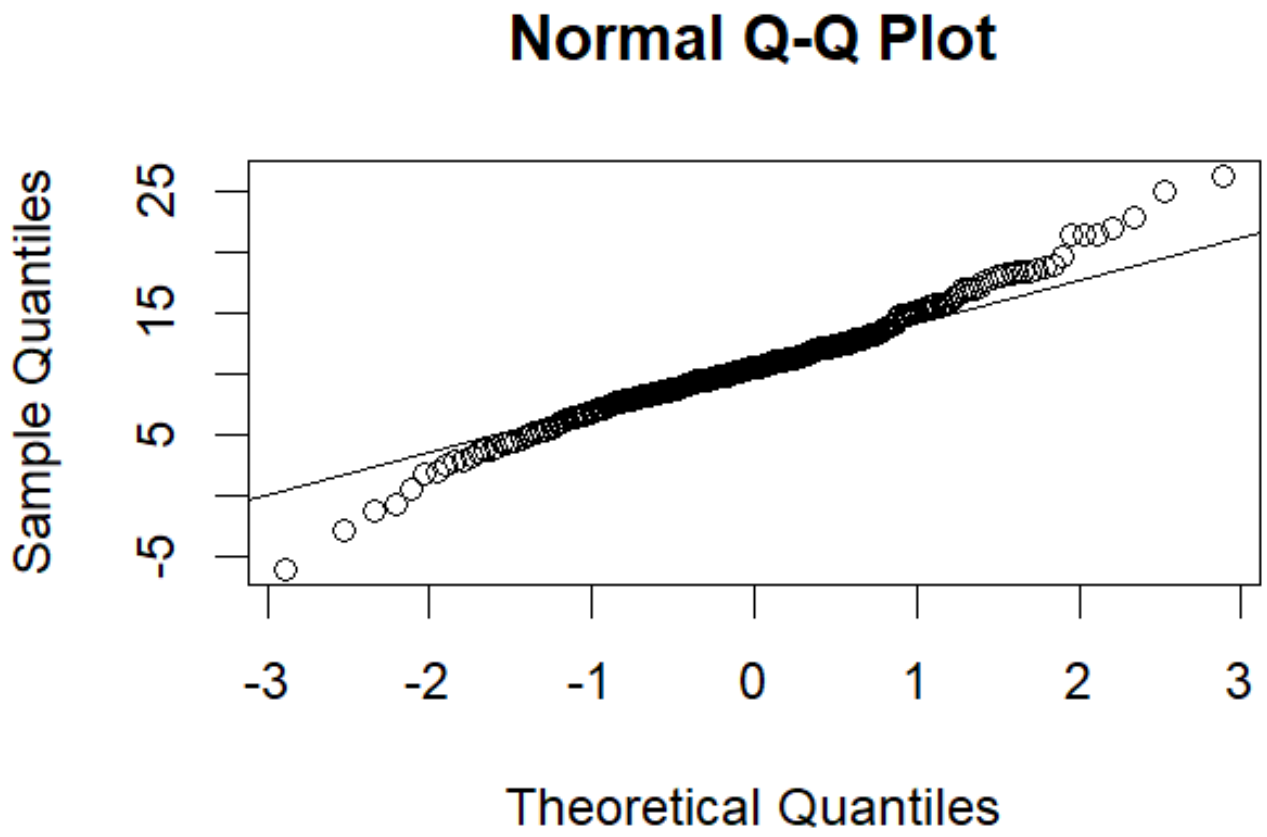
Проверка предположений о нормальности (графическая)

Чтобы проверить предположение о нормальности переменной (нормальность означает, что данные подчиняются нормальному распределению, также известному как распределение Гаусса), мы обычно используем гистограммы и/или QQ-графики[1]. Гистограммы были представлены ранее, научимся сейчас строить и анализировать QQ-графики:

Функции в R (qq-plot)

```
qqnorm(v1)
qqline(v1)
```

[1] В Лекции 3 мы обсудим нормальное распределение и как проверить предположение о нормальности в R.

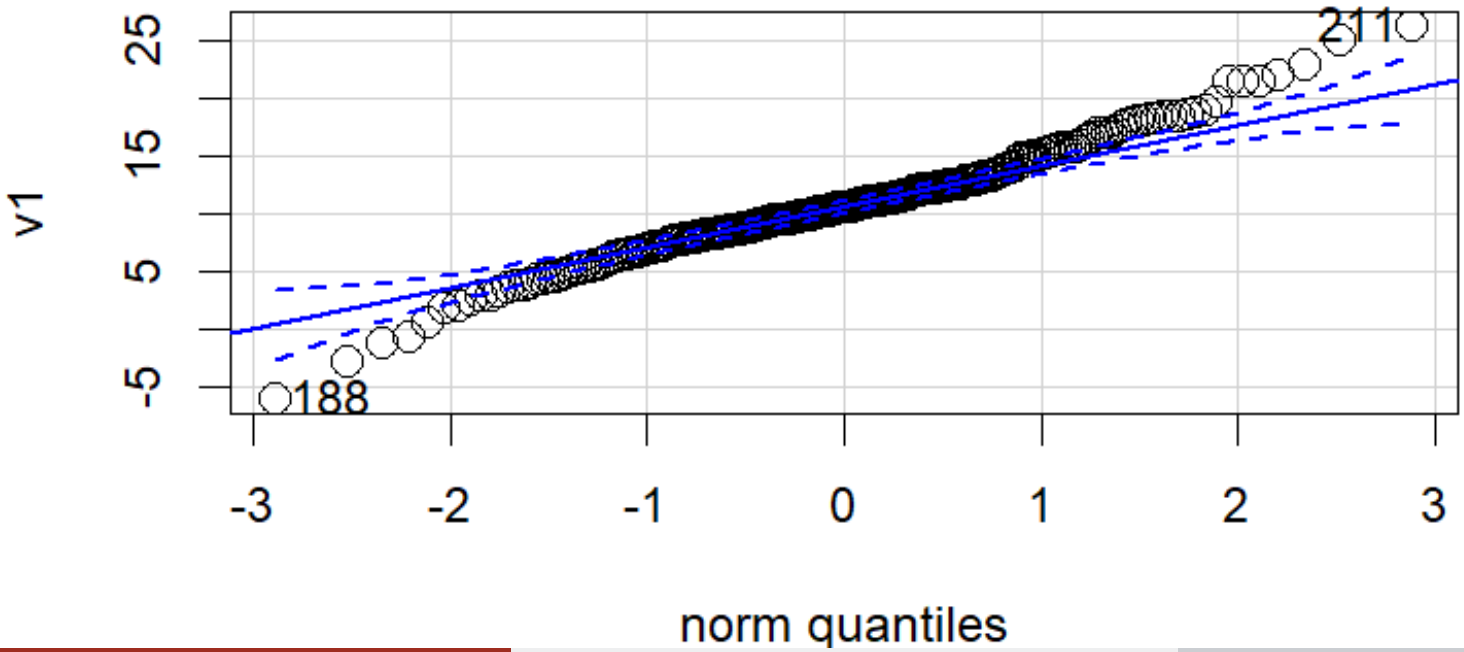

Normal Q-Q Plot
Sample Quantiles
-5
5
15
25
-3
-2
-1
0
1
2
3
Theoretical Quantiles

Или QQ-график с доверительным интервалом с функцией `qqPlot()` из пакета `car`:

Функции в R (пакет car)

```
library(car)
qqPlot(v1)
```

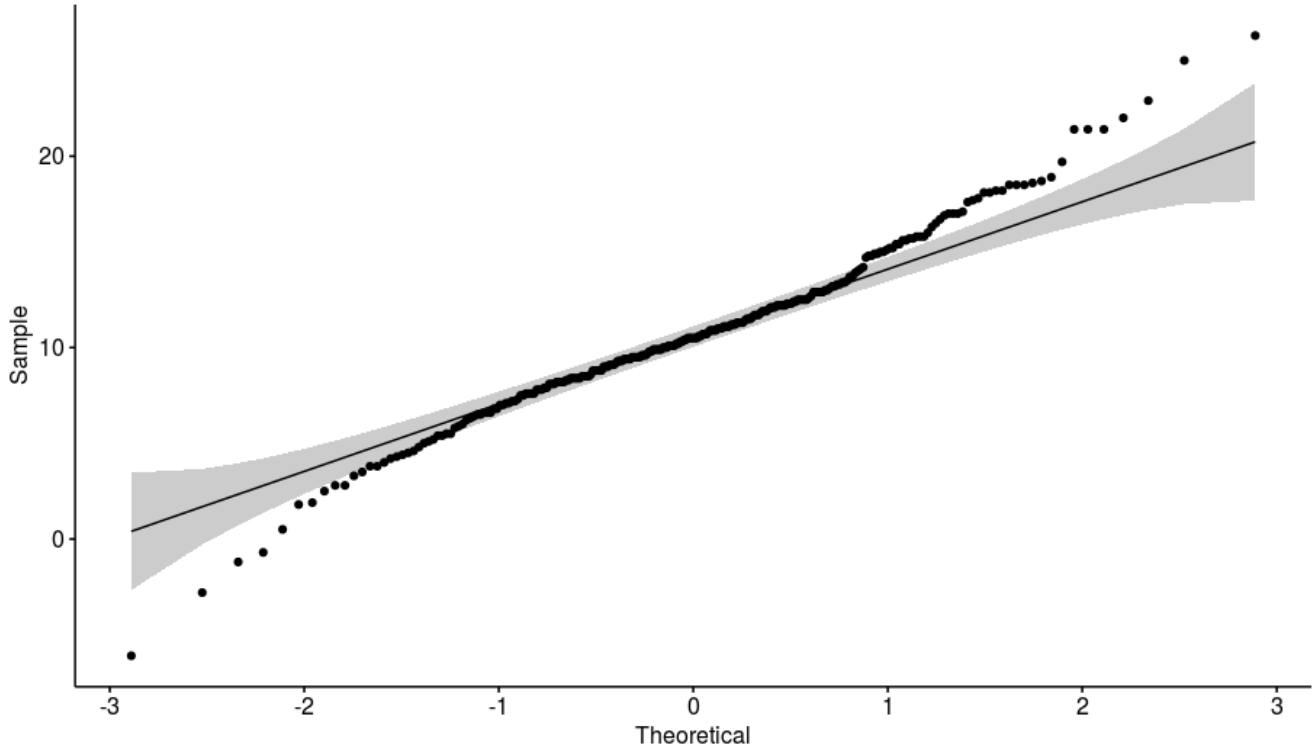
Вывод: `[1] 188 211` # выбросы (наиболее сильные)



Можно попробовать прологарифмировать данные и опять построить QQ-график:

Функции в R (пакет ggpubr)

```
library(ggplot2)
library(ggpubr)
ggqqplot(v1)
```



Итоги

Что мы узнали на Лекции 2?

- Какие возможности имеются в R для графического представления данных.
- Что такое эмпирическая функция распределения, гистограмма, диаграмма рассеяния, полигон частот, а также, как все это изображать в R.
- Как графически исследовать зависимости между данными.
- Что такое ящик с усами.
- Как изображать временные ряды разными способами.
- Какие гипотезы можно выдвинуть при графическом анализе данных.

Что мы узнаем на Лекции 3?

Мы узнаем,

- как проверять гипотезы о распределении данных в R.
- как проверять гипотезы об однородности данных, если имеется две и более выборок.

Спасибо за внимание и до встречи на Лекции 3!