

# Математические пакеты

## Основы работы с диаграммами в R

Сучков Андрей Игоревич

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»

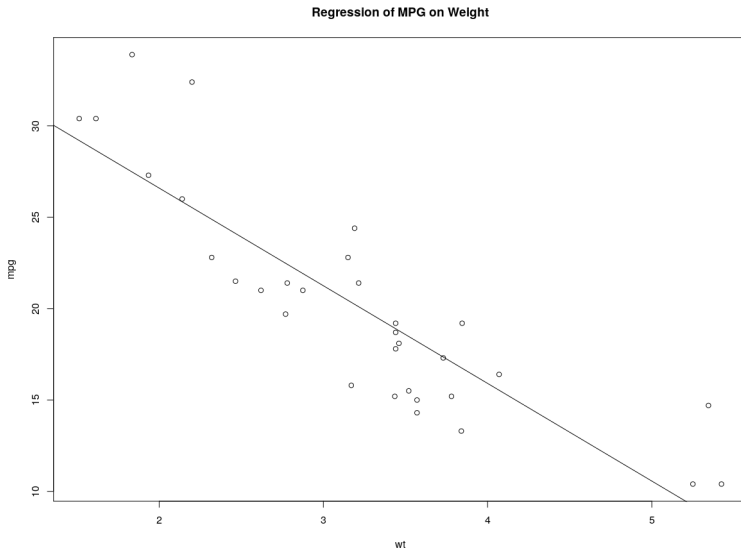
14 ноября 2020 г.

### Листинг 1: Пример построения диаграмм

```
1 attach(mtcars)
2   plot(wt, mpg)
3   abline(lm(mpg ~ wt))
4   title("Regression of MPG on Weight")
5 detach(mtcars)
```

# Работа с диаграммами

## Результат построения диаграммы



### Листинг 2: Сохранение в формате .pdf

```
1 pdf("my-figure.PDF")
2   attach(mtcars)
3     plot(wt, mpg)
4     abline(lm(mpg ~ wt))
5     title("Regression of MPG on Weight")
6   detach(mtcars)
7 dev.off()
```

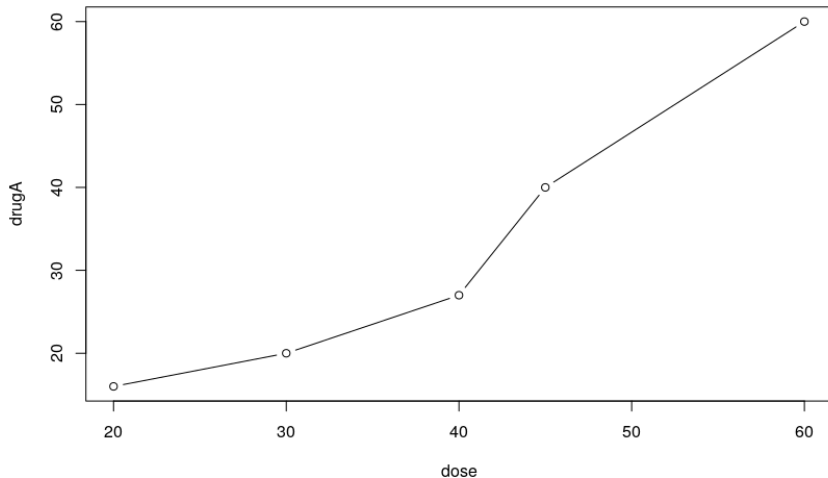
Можно сохранять в других форматах: png, jpeg, bmp, tiff, xfig ...

### Листинг 3: Реакция пациента на два лекарства в пяти дозировках

```
1 dose <- c(20, 30, 40, 45, 60)
2 drugA <- c(16, 20, 27, 40, 60)
3 drugB <- c(15, 18, 25, 31, 40)
4
5 plot(dose, drugA, type="b")
```

# Работа с диаграммами

## Результат построения диаграммы



# Работа с диаграммами

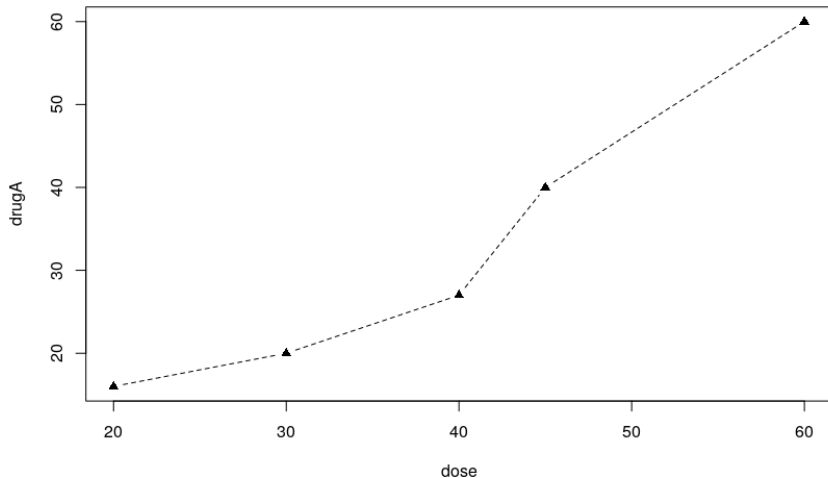
## Простой пример

### Листинг 4: Реакция пациента на два лекарства в пяти дозировках

```
1 opar <- par(no.readonly=TRUE)
2 par(lty = 2, pch = 17)
3 ## Equivalent
4 ## par(lty = 2)
5 ## par(pch = 17)
6 plot(dose, drugA, type = "b")
7 par(opar)
8
9 ## Another way
10 plot(dose, drugA, type = "b", lty = 2, pch = 17)
```

# Работа с диаграммами

## Результат построения диаграммы





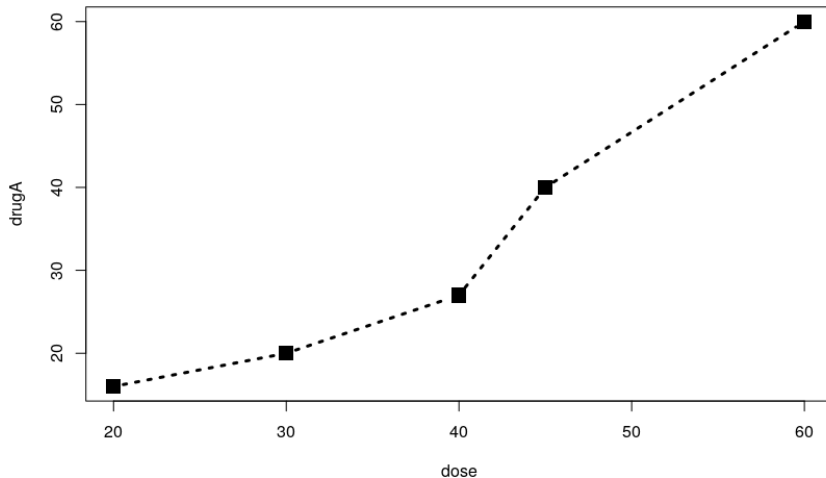
# Графические параметры

## Символы и линии

- `pch` – Определяет тип символа
- `cex` – Определяет размер символа. `cex` – это число, обозначающее, как символы должны быть масштабированы по отношению к размеру по умолчанию. 1 = размер по умолчанию, 1.5 – на 50% крупнее, 0.5 – на 50% мельче и т. д.
- `lty` – Определяет тип линии
- `lwd` – Определяет толщину линии по сравнению с толщиной линии по умолчанию (1). Например, `lwd = 2` делает линию в два раза толще, чем по умолчанию
- Программный код, объединяющий все эти параметры:  
`plot(dose, drugA, type="b", lty=3, lwd=3, pch=15, cex=2)`

# Графические параметры

Результат построения диаграммы



# Графические параметры

## Цвета

- `col` – Цвет элементов на графике. Для некоторых функций (таких как `lines` и `pie`) можно указывать вектор из значений, которые используются по очереди. Например, если `col=c("red", "blue")` и изображены три линии, первая будет красной, вторая – синей и третья – красной
- `col.axis` – Цвет значений осей
- `col.lab` – Цвет подписей осей
- `col.main` – Цвет заголовков
- `col.sub` – Цвет подзаголовков
- `fg` – Цвет графика
- `bg` – Цвет фона

В R цвета можно обозначать номером, названием, в шестнадцатеричной системе, а также в системах RGB или HSV.

# Графические параметры

## Характеристики текста

- `sex` – Число, определяющее, как отображаемый на диаграмме текст будет масштабирован относительно размера по умолчанию (1). 1.5 – на 50% больше, 0.5 – на 50% меньше и т. д.
- `sex.axis` – Размер значений на осях по отношению к `sex`
- `sex.lab` – Размер названий осей по отношению к `sex`
- `sex.main` – Размер заголовков по отношению к `sex`
- `sex.sub` – Размер подзаголовков по отношению к `sex`

# Графические параметры

## Характеристики текста

- `font` – Число, которое определяет шрифт для текста на диаграмме. 1 = обычный, 2 = полужирный, 3 = курсив, 4 = полужирный курсив, 5 = символы (в кодировке Adobe)
- `font.axis` – Шрифт значений на осях
- `font.lab` – Шрифт для подписей по осям
- `font.main` – Шрифт для заголовков
- `font.sub` – Шрифт для подзаголовков
- `ps` – Размер точки в шрифте (приблизительно 0.3 мм)
- `family` – Семейство шрифтов. Стандартные значения – `serif`, `sans` и `mono`

# Графические параметры

## Размеры диаграммы и полей

- `pin` – Размер диаграммы (ширина, высота) в дюймах
- `mai` – Числовой вектор, задающий размеры полей, где параметры `s(низ, лево, верх, право)` измеряются в дюймах (1 дюйм = 2.5 см)
- `mar` – Числовой вектор, задающий размеры полей, где параметры `s(низ, лево, верх, право)` измеряются в числе строк. По умолчанию это `s(5, 4, 4, 2) + 0.1`

# Графические параметры

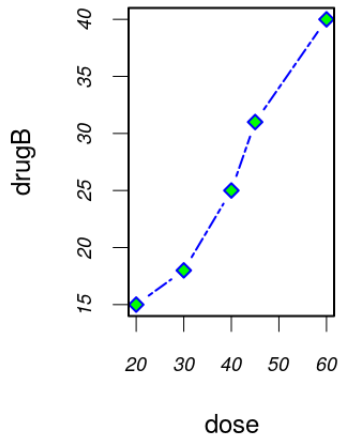
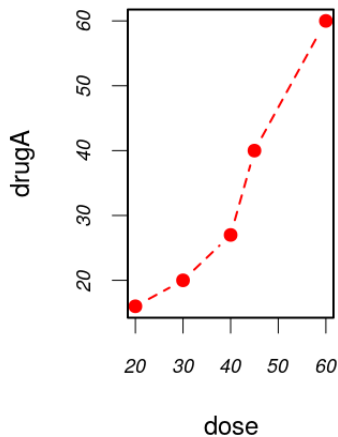
Использование графических параметров для определения внешнего вида диаграммы

## Листинг 5: Пример работы вышеописанных параметров

```
1 dose <- c(20, 30, 40, 45, 60)
2 drugA <- c(16, 20, 27, 40, 60)
3 drugB <- c(15, 18, 25, 31, 40)
4 opar <- par(no.readonly=TRUE)
5 par(pin=c(2, 3))
6 par(lwd=2, cex=1.5)
7 par(cex.axis=.75, font.axis=3)
8 plot(dose, drugA, type="b", pch=19, lty=2, col="red")
9 plot(dose, drugB, type="b", pch=23, lty=6, col="blue",
10      bg="green")
11 par(opar)
```

# Графические параметры

Результат работы скрипта





# Объединение диаграмм

Функция `par` и параметр `mfrow`

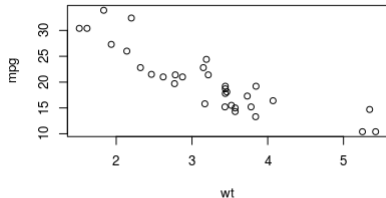
## Листинг 6: Использование `par` с параметром `mfrow`

```
1 attach(mtcars)
2   opar <- par(no.readonly=TRUE)
3   par(mfrow=c(2,2))
4   plot(wt,mpg, main="Scatter plot for wt & mpg")
5   plot(wt,disp, main="Scatter plot for wt & mpg")
6   hist(wt, main="Distribution of values for wt")
7   boxplot(wt, main="Box plot for wt")
8   par(opar)
9 detach(mtcars)
```

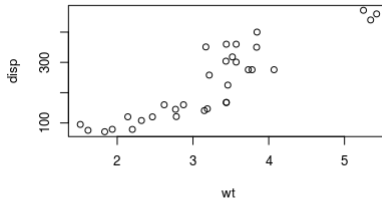
# Объединение диаграмм

Результат работы программы

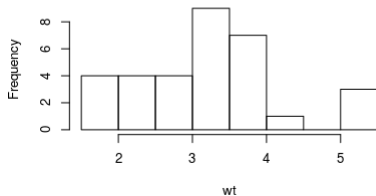
Scatter plot for wt & mpg



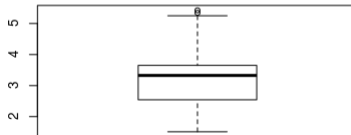
Scatter plot for wt & mpg



Distribution of values for wt



Box plot for wt



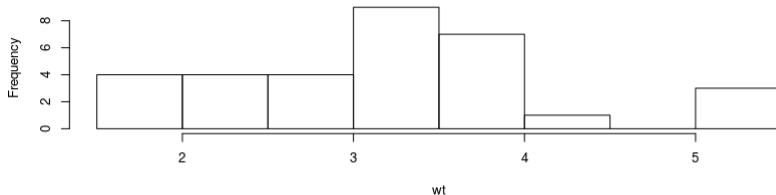
### Листинг 7: Использование layout

```
1 attach(mtcars)
2   layout(matrix(c(1,1,2,3), 2, 2, byrow=TRUE))
3   hist(wt)
4   hist(mpg)
5   hist(displ)
6 detach(mtcars)
```

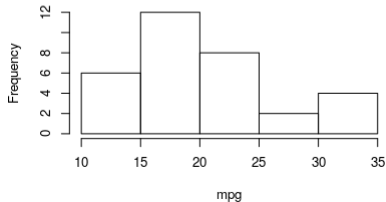
# Объединение диаграмм

Результат работы программы

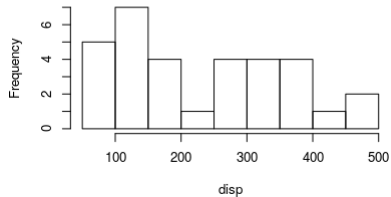
Histogram of wt



Histogram of mpg



Histogram of disp



# Объединение диаграмм

## Полный контроль над расположением диаграмм

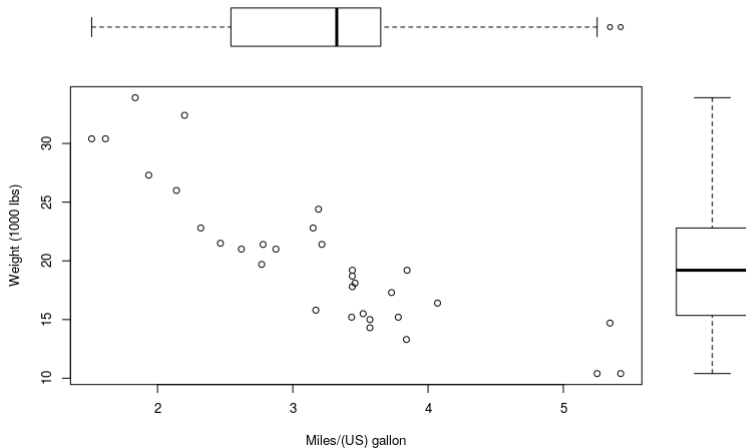
Листинг 8: Диаграмма рассеяния с двумя диаграммами размахов на полях

```
1 opar <- par(no.readonly=TRUE)
2
3 par(fig=c(0, 0.8, 0, 0.8))
4 plot(mtcars$wt, mtcars$mpg,
5       xlab="Miles/(US) gallon",
6       ylab="Weight (1000 lbs)")
7
8 par(fig=c(0, 0.8, 0.55, 1), new=TRUE)
9 boxplot(mtcars$wt, horizontal=TRUE, axes=FALSE)
10
11 par(fig=c(0.65, 1, 0, 0.8), new=TRUE)
12 boxplot(mtcars$mpg, axes=FALSE)
13
14 mtext("Improved scatter plot", side=3, outer=TRUE, line=-3)
15 par(opar)
```

# Объединение диаграмм

Результат работы программы

Improved scatter plot



# Столбчатые диаграммы

## Простые столбчатые диаграммы

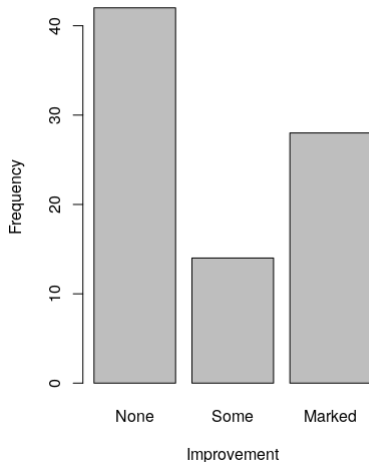
Листинг 9: Диаграмма рассеяния с двумя диаграммами размахов на полях

```
1 counts <- table(Arthritis$Improved)
2
3 par(mfrow=c(1,2))
4 barplot(counts,
5         main="Vertical bar plot",
6         xlab="Improvement", ylab="Frequency")
7
8 barplot(counts,
9         main="Horizontal bar plot",
10        xlab="Frequency", ylab="Improvement",
11        horiz=TRUE)
```

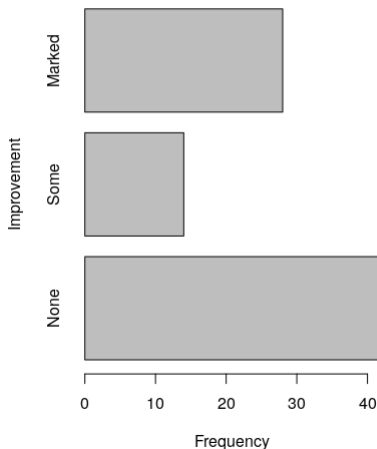
# Столбчатые диаграммы

Результат работы программы

**Vertical bar plot**



**Horizontal bar plot**





# Столбчатые диаграммы

## Простые столбчатые диаграммы

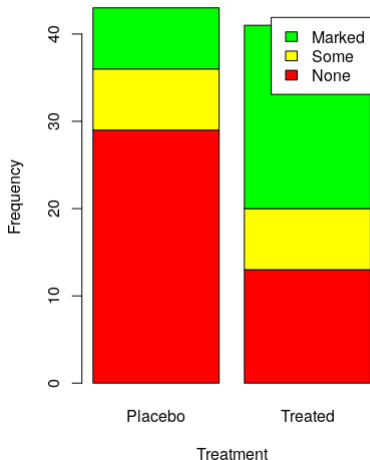
### Листинг 10: Построение столбчатых диаграмм

```
1 counts <- table(Arthritis$Improved, Arthritis$Treatment)
2
3 par(mfrow=c(1,2))
4 barplot(counts,
5         main="Composite bar plot",
6         xlab="Treatment", ylab="Frequency",
7         col=c("red", "yellow", "green"),
8         legend=rownames(counts))
9
10 barplot(counts,
11         main="Composite bar plot",
12         xlab="Treatment", ylab="Frequency",
13         col=c("red", "yellow", "green"),
14         legend=rownames(counts), beside=TRUE)
```

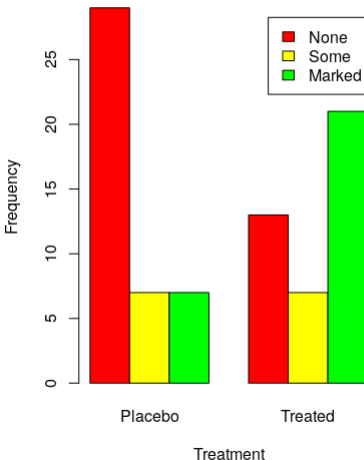
# Столбчатые диаграммы

Результат работы программы

Composite bar plot



Composite bar plot



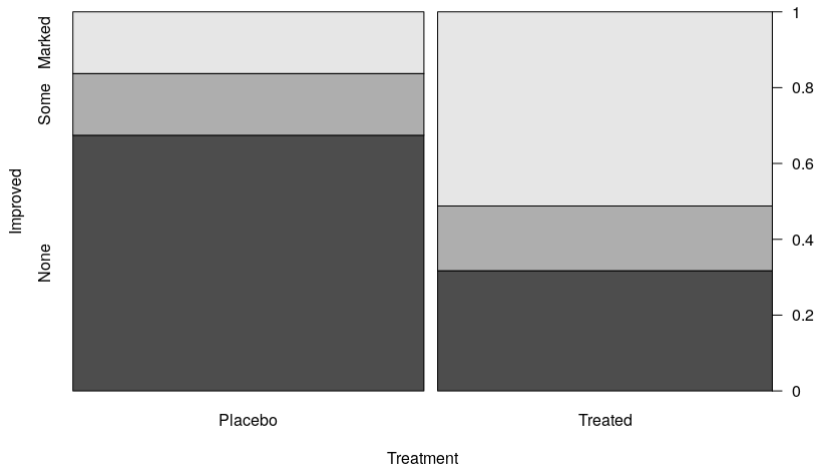
### Листинг 11: Построение спинограммы

```
1 attach(Arthritis)
2 counts <- table(Treatment, Improved)
3 spine(counts, main="Spinogram example")
4 detach(Arthritis)
```

# Столбчатые диаграммы

Результат работы программы

Spinogram example



# Круговые диаграммы

## Построение

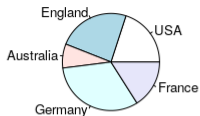
### Листинг 12: Построение круговых диаграмм

```
1 layout(matrix(c(1,1,2,3), 2, 2, byrow=TRUE))
2
3 slices <- c(10, 12, 4, 16, 8)
4 lbls <- c("USA", "England", "Australia",
5           "Germany", "France")
6
7 pie(slices, labels = lbls, main="Simple pie chart")
8
9 pct <- round(slices/sum(slices)*100)
10 lbls2 <- paste(lbls, " ", pct, "%", sep="")
11 pie(slices, labels=lbls2, col=rainbow(length(lbls2)),
12     main="Pie chart with percent")
13
14 mytable <- table(state.region)
15 lbls3 <- paste(names(mytable), "\n", mytable, sep="")
16 pie(mytable, labels = lbls3,
17     main="Pie chart for table\n(with sample sizes)")
```

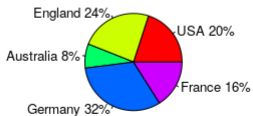
# Круговые диаграммы

## Результат работы программы

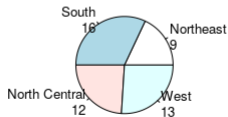
**Simple pie chart**



**Pie chart with percent**



**Pie chart for table  
(with sample sizes)**



### Листинг 13: Построение гистограмм

```
1 par(mfrow=c(2,2))
2
3 hist(mtcars$mpg)
4
5 hist(mtcars$mpg,
6       breaks=12, col="red", xlab="Miles/(US) gallon",
7       main="Colored bar chart with 12 bars")
8
9 hist(mtcars$mpg, freq=FALSE,
10      breaks=12, col="red", xlab="Miles/(US) gallon",
11      main="Histogram, brush plot\nand distribution density
12           curve")
13
14 rug(jitter(mtcars$mpg))
15 lines(density(mtcars$mpg), col="blue", lwd=2)
```

### Листинг 14: Построение гистограмм (продолжение)

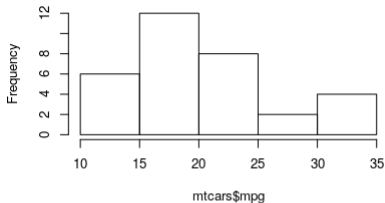
```
1 x <- mtcars$mpg
2 h <- hist(x, breaks=12, col="red", xlab="Miles/(US) gallon",
3         main="Boxed histogram with bell curve")
4
5 xfit <- seq(min(x), max(x), length=40)
6 yfit <- dnorm(xfit, mean=mean(x), sd=sd(x))
7 yfit <- yfit * diff(h$mids[1:2]) * length(x)
8
9 lines(xfit, yfit, col="blue", lwd=2)
10 box()
```



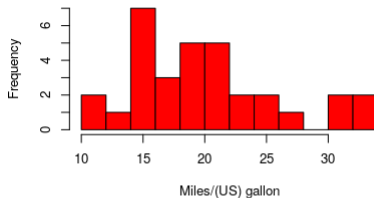
# Гистограммы

Результат работы программы

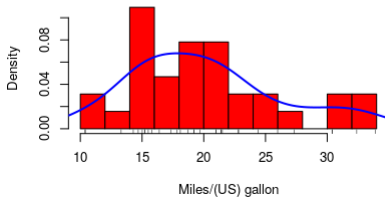
**Histogram of mtcars\$mpg**



**Colored bar chart with 12 bars**



**Histogram, brush plot  
and distribution density curve**



**Boxed histogram with bell curve**

