Визуализация данных: базовые функции R и пакет ggplot2

Г. Мороз

презентация доступна:

R для визуализации? Совсем не обязательно...

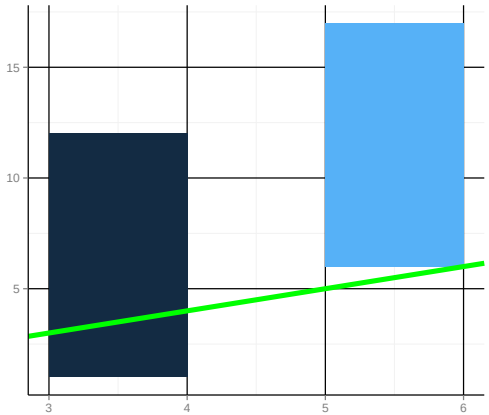
Взято отсюда. Куча ресурсов, которые скоро устареют.

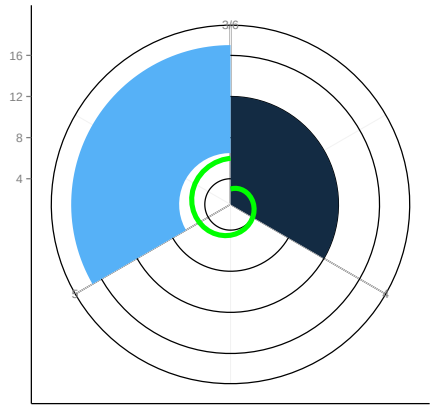
- **Matplotlib**
- **Bokeh**
- AnyChart
- Chart Tool
- Chart.js
- Chartbuilder
- Chartbuilder 2.0
- ChartGo
- Chiasm
- D3plus
- Datahero
- Datamatic
- Datavisual
- Datawrapper
- Diagrammer
- Diychart
- Dygraphs
- Echarts
- Envision.js
- filtergraph
- Flare
- Google Charts
- Highcharts
- iCharts
- Infogr.am
- JS Charts
- JavaScript InfoVis Toolkit
- Livegap Charts
- Lyra
- Plotly
- **Processing**
- Qlik
- Raw
- Lumira
- Slemma
- Spotfire
- Sprites
- Tableau
- VIDI
- Vega
- Visage
- Vizydrop
- Weave
- Zingchart

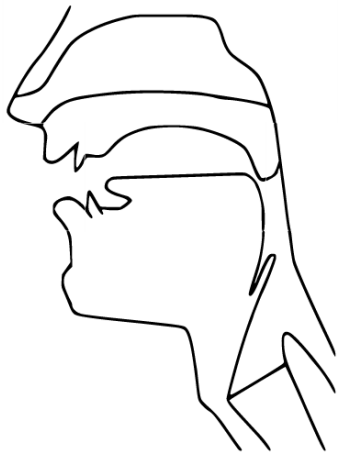
презентация доступна:

Элементы визуализации

- система координат







Элементы визуализации

- позиция
- длина
- форма
- угол
- направление
- размер
- цвет
- ...

Данные

В данной презентации все примеры будут приводиться на примере датасета из работы [Chi-kuk 2007] (доступна по ссылке http://goo.gl/Zjr9aF). В работе исследовались речь 7 гомосексуальных и 7 гетеросексуальных носителей кантонского диалекта языка юэ. В датасете есть следующие переменные:

- долгота s (s.duration.ms)
- долгота гласных (vowel.duration.ms)
- среднее значении ЧОТ (average.f0.Hz)
- диапозон ЧОТ (f0.range.Hz)
- сколько носителей воспринимает говорящего как гомосексуала (perceived.as.homo)
- сколько носителей воспринимает говорящего как гетеросексуала (perceived.as.hetero)
- ориентация говорящего (orientation)
- возраст говорящего (age)

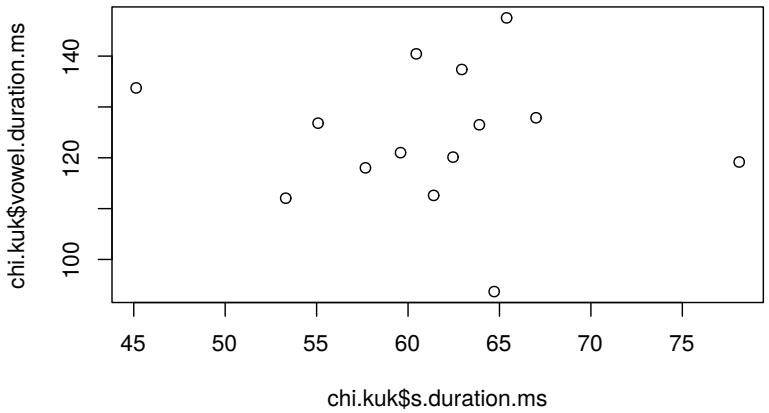
В R визуализация реализована по-разному:

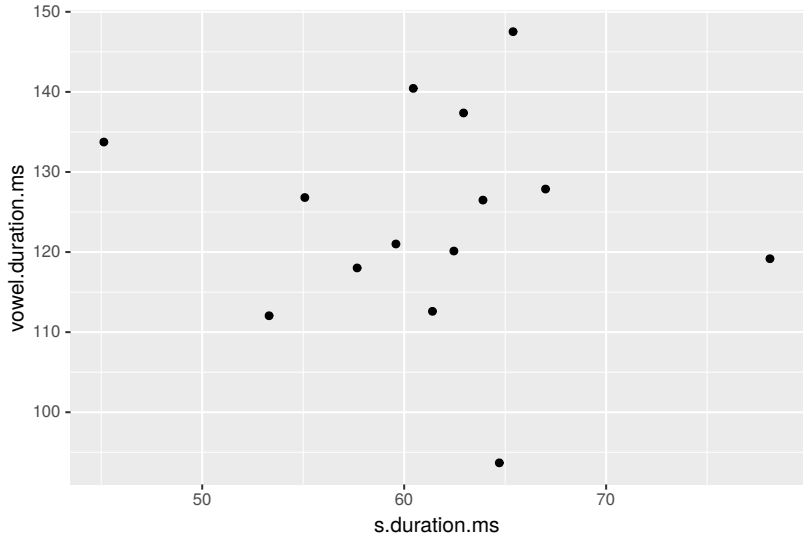
- core R (всегда будет слева)
- библиотекой ggplot2 (всегда будет справа)

Все примеры ggplot2 в данной презентации реализованы в связке с пакетом dplyr. Так что, для того чтобы код из данной презентации работал следует включить библиотеки и скачать датасет:

```
library(ggplot2)
library(dplyr)
chi.kuk <- read.csv("http://goo.gl/Zjr9aF")              # закачивает датасет
```

scaterplot

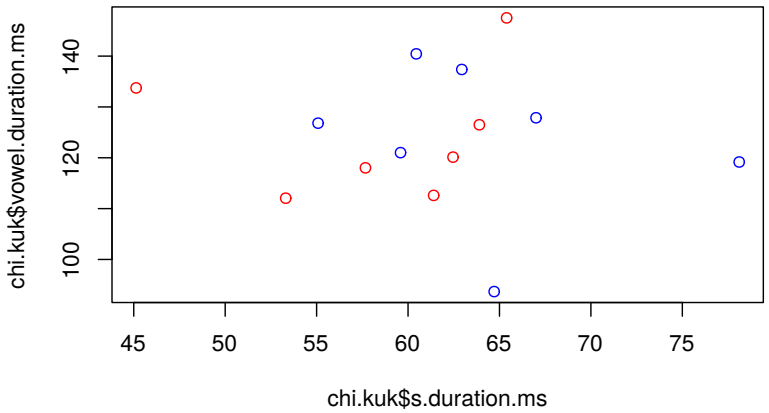


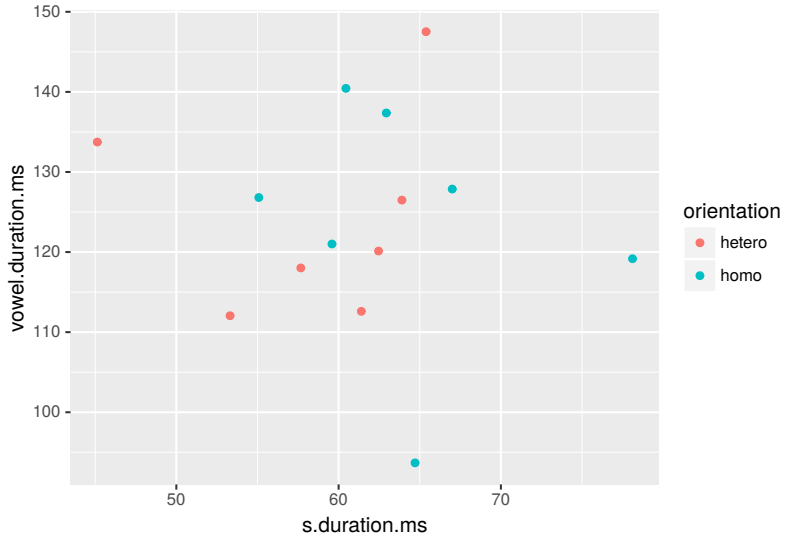



```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms)

# dplyr, ggplot2
chi.kuk %>%
  ggplot(aes(s.duration.ms, vowel.duration.ms)) +
  geom_point()
```

scaterplot: цвет

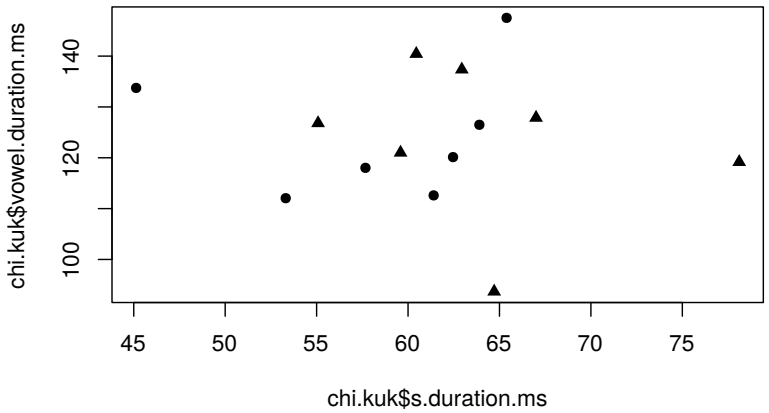


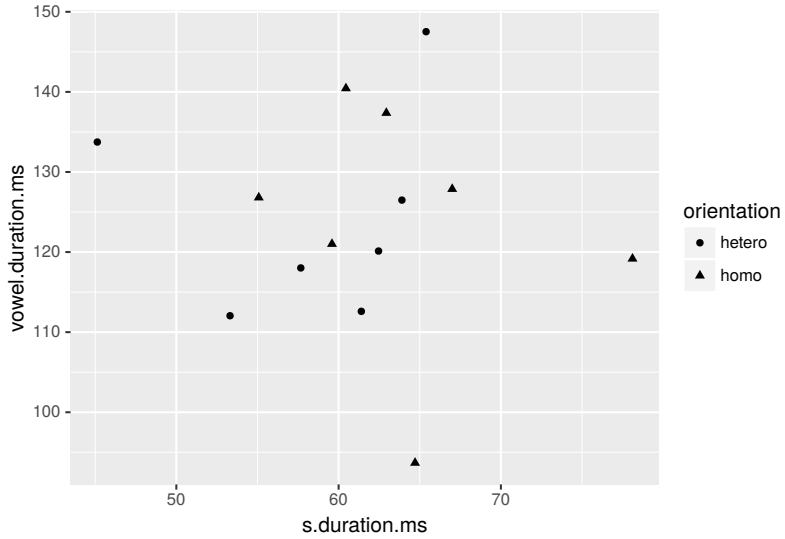



```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
   col = c("red", "blue")[as.numeric(chi.kuk$orientation)])

# dplyr, ggplot2
chi.kuk %>%
   ggplot(aes(s.duration.ms, vowel.duration.ms, color = orientation)) +
   geom_point()
```

scaterplot: форма

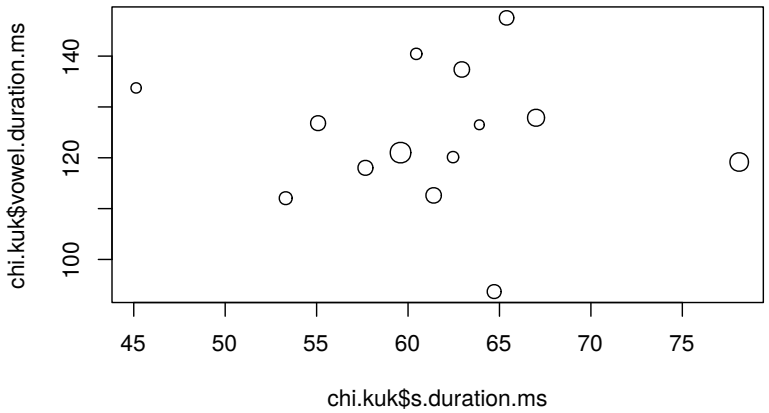


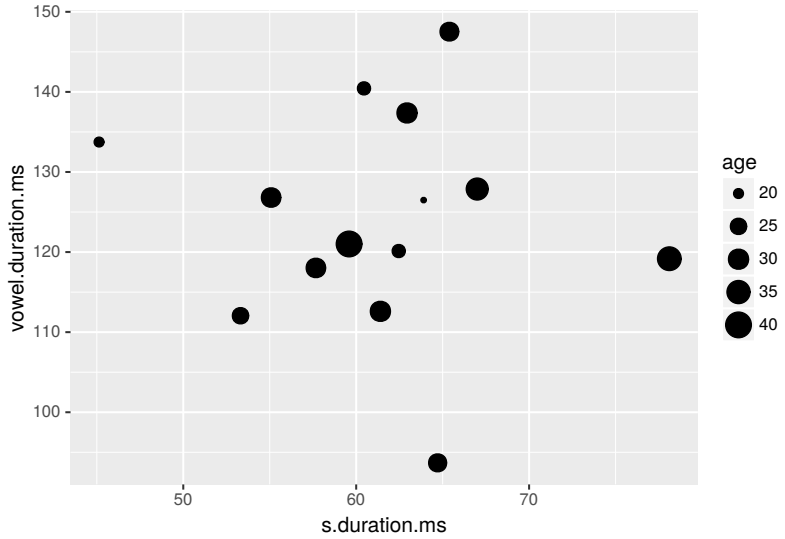



```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
   pch = c(16, 17)[as.numeric(chi.kuk$orientation)])

# dplyr, ggplot2
chi.kuk %>%
   ggplot(aes(s.duration.ms, vowel.duration.ms, shape = orientation)) +
   geom_point()
```

scaterplot: размер

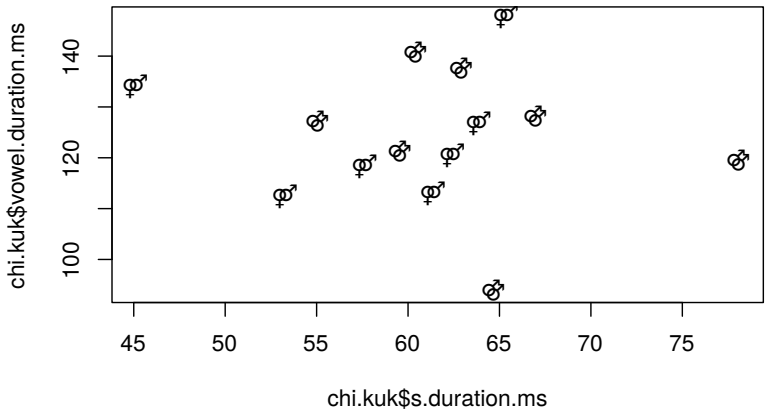


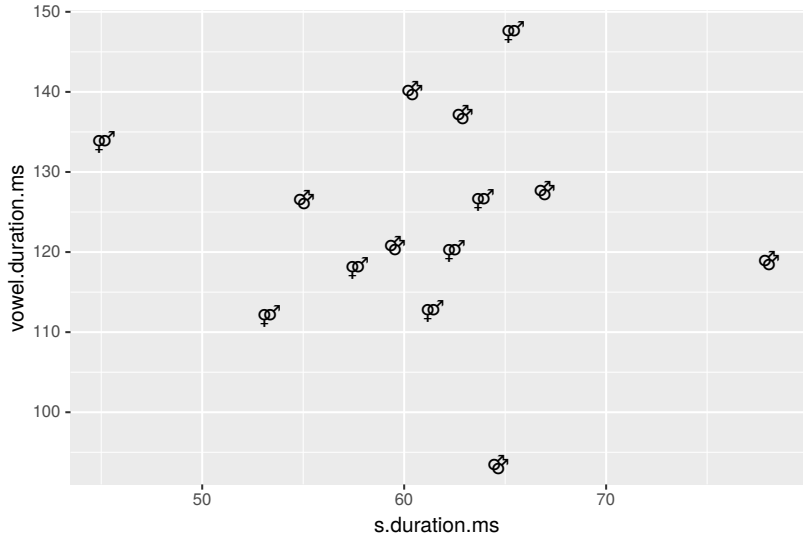



```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
  cex = chi.kuk$age/20)

# dplyr, ggplot2
chi.kuk %>%
  ggplot(aes(s.duration.ms, vowel.duration.ms, color = orientation)) +
  geom_point()
```

scaterplot: текст

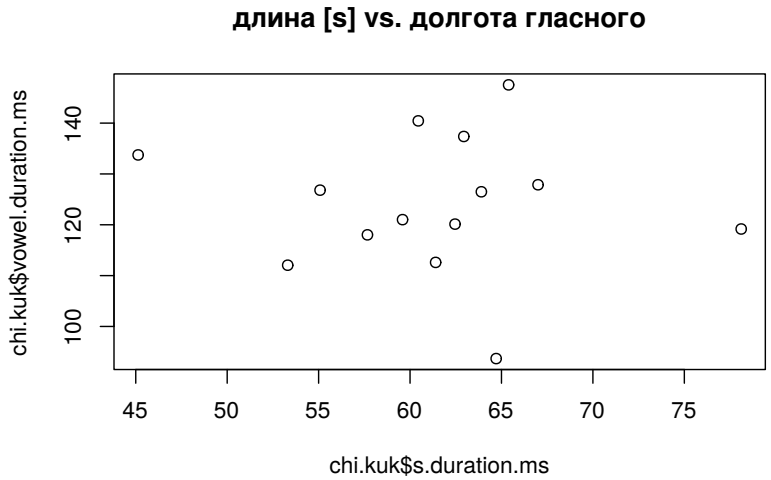


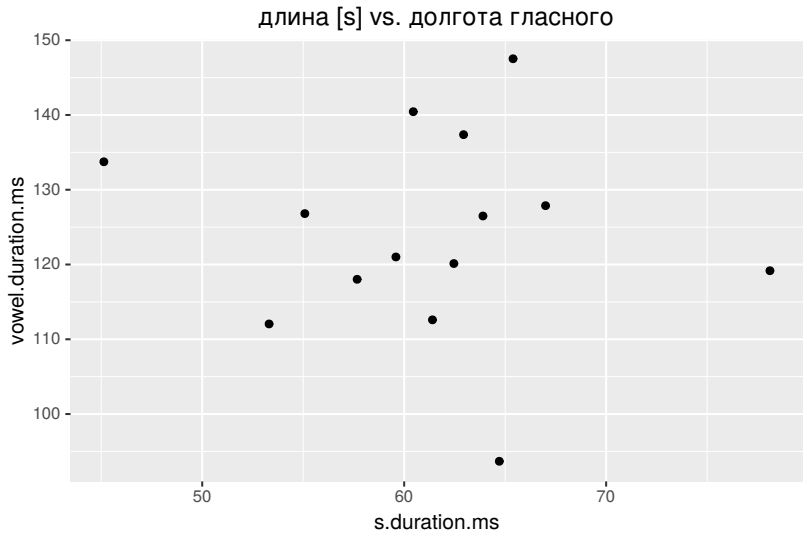



```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
   pch = c("⚤", "⚣")[as.numeric(chi.kuk$orientation)])

# dplyr, ggplot2
levels(chi.kuk$orientation) <- list("homo"="⚣", "hetero"="⚤")
chi.kuk %>%
   ggplot(aes(s.duration.ms, vowel.duration.ms, label = orientation)) +
   geom_text()
```

scaterplot: заголовок



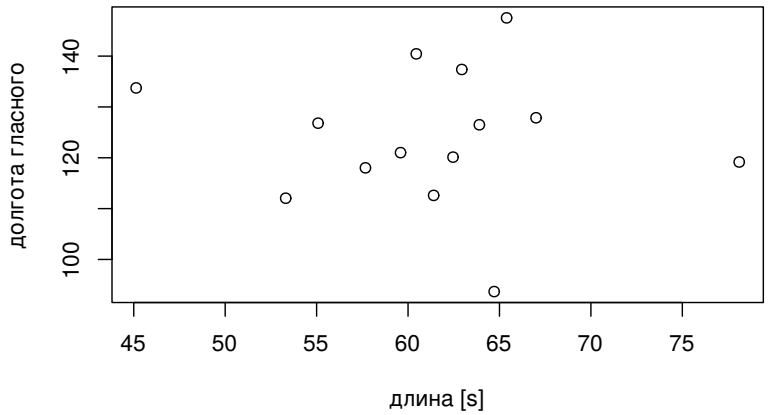


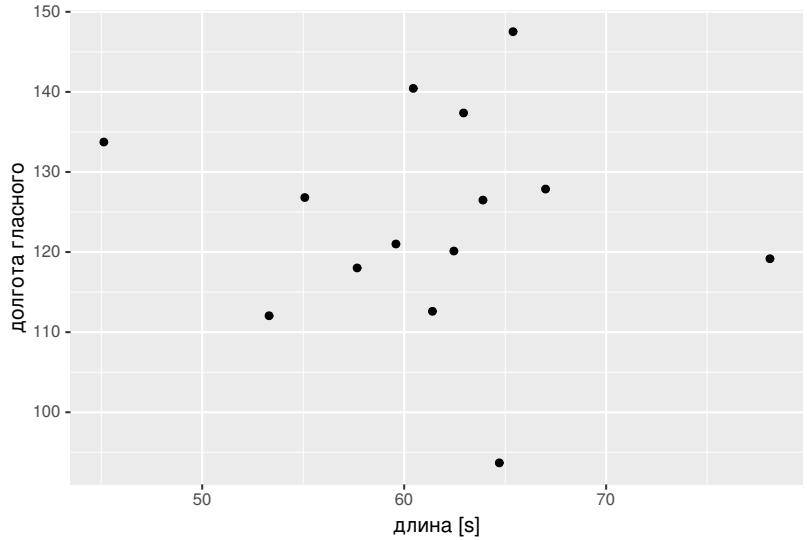

```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
   main = "длина [s] vs. долгота гласного")

# dplyr, ggplot2
chi.kuk %>%
   ggplot(aes(s.duration.ms, vowel.duration.ms)) +
   geom_point() +
   ggtitle("длина [s] vs. долгота гласного")
```

презентация доступна:

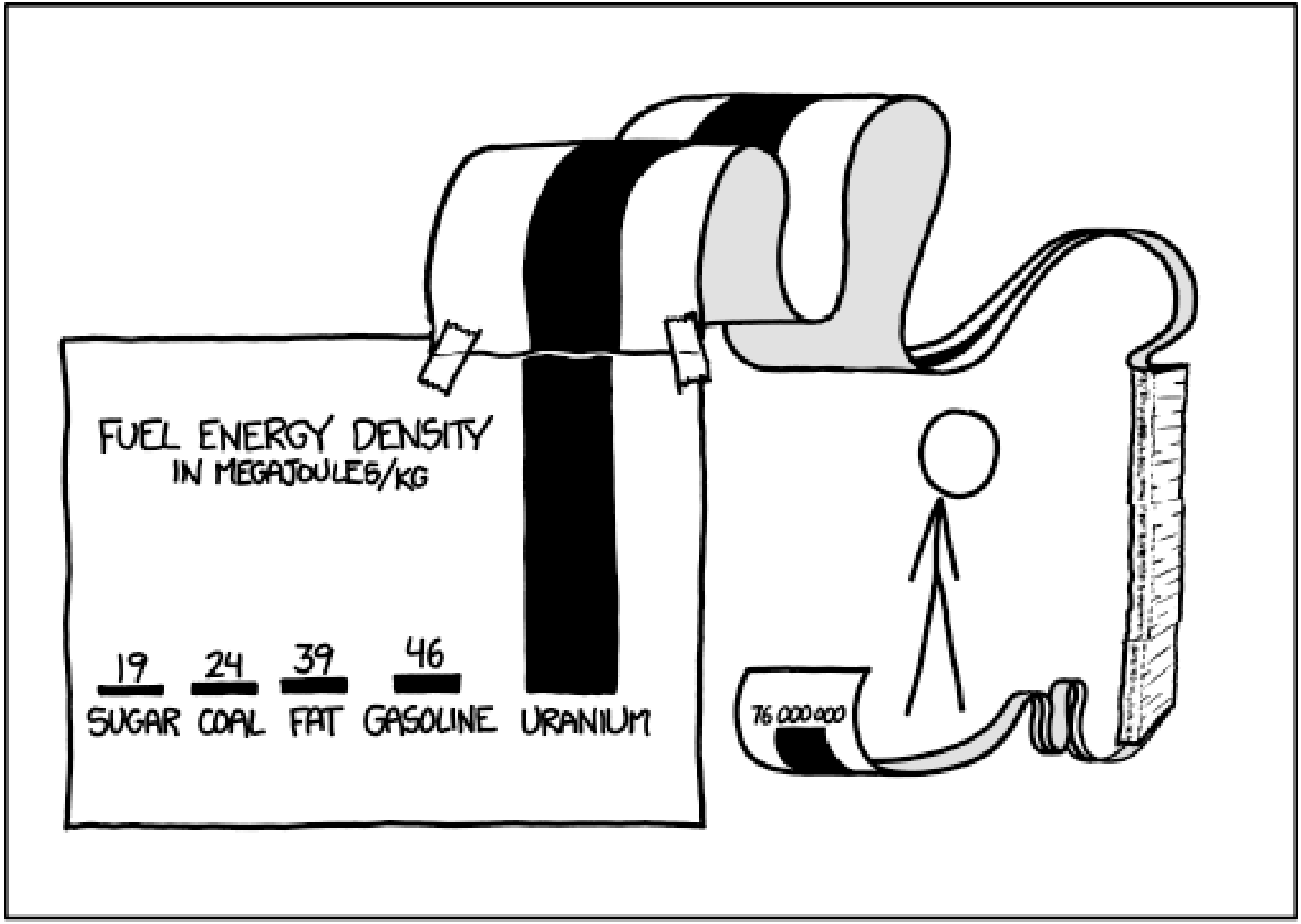
scaterplot: подписи осей






```
# base R
plot(chi.kuk$s.duration.ms, chi.kuk$vowel.duration.ms,
  xlab = "длина [s] ylab = "долгота гласного")

# dplyr, ggplot2
chi.kuk %>%
  ggplot(aes(s.duration.ms, vowel.duration.ms)) +
  geom_point() +
  xlab("длина [s]")+
  ylab("долгота гласного")
```



SCIENCE TIP: LOG SCALES ARE FOR QUITTERS WHO CAN'T FIND ENOUGH PAPER TO MAKE THEIR POINT *PROPERLY.*

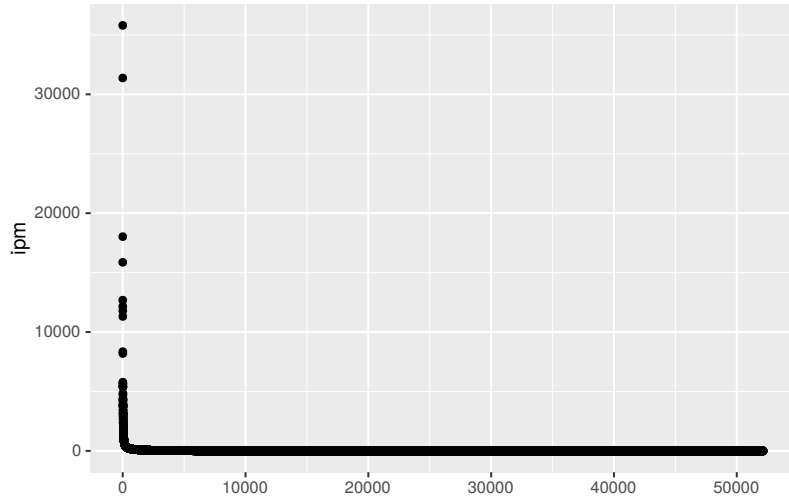
scaterplot: логарифмирование осей

Воспользуемся частотным словарем [Ляшевская, Шаров 2009] (топ-50000 лемм СЛРЯ) и посмотрим на параметр частоты слова (Freq.imp., среднее на миллион словоупотреблений).

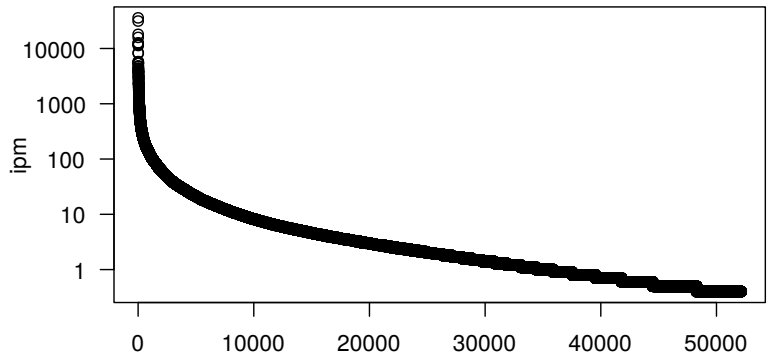
```
freq <- read.csv("https://goo.gl/TlX7xW", sep = "\t")
```

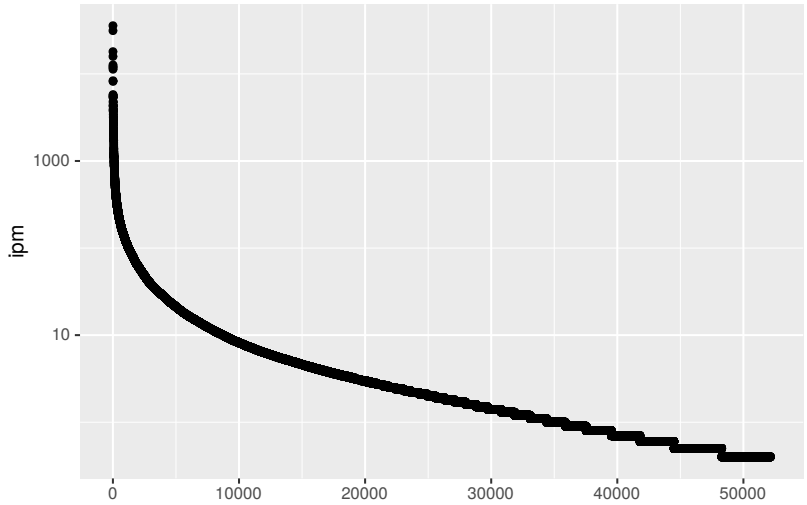
Если оси не логарифмировать, то получится следующее:

```
freq %>%
  ggplot(aes(1:52138, Freq.ipm.)) +
  geom_point() +
  xlab("") +
  ylab("ipm")
```



scaterplot: логарифмирование осей






```
# base R
plot(1:52138, freq$Freq.ipm.,
   xlab = NA, ylab = "ipm",
   las = 1,                          # поворот значений на оси y
   log = "y")

# dplyr, ggplot2
freq %>%
   ggplot(aes(1:52138, Freq.ipm.))+
   geom_point()+
   xlab("")+
   ylab("ipm")+
   scale_y_log10()
```

презентация доступна:

Спасибо что долистали!

Пишите письма

agricolamz@gmail.com

презентация доступна: