Анализ данных для лингвистов

Г. А. Мороз

Оглавление

1 **О курсе** **5**

2 **Распределения** **7**

2.1 Распределения в R 7
2.2 Дискретные переменные 10
2.3 Числовые переменные 16

3 **Метод максимального правдоподобия** **19**

3.1 Оценка вероятности 19
3.2 Функция правдоподобия 21
3.3 Пример с непрерывным распределением 23
3.4 Метод максимального правдоподобия (MLE) 25

Глава 1

О курсе

Материалы для курса Анализа данных для лингвистов, Школа лингвистики НИУ ВШЭ.

- запись лекции 2020.01.13[1]
- запись лекции 2020.01.15[2]
- запись лекции 2020.01.20[3]

[1] https://youtu.be/HmLcBJnfipk
[2] https://youtu.be/V_c_K_wBMuY
[3] https://youtu.be/cRc5z9F3XNw

Глава 2

Распределения

```
library(tidyverse)
```

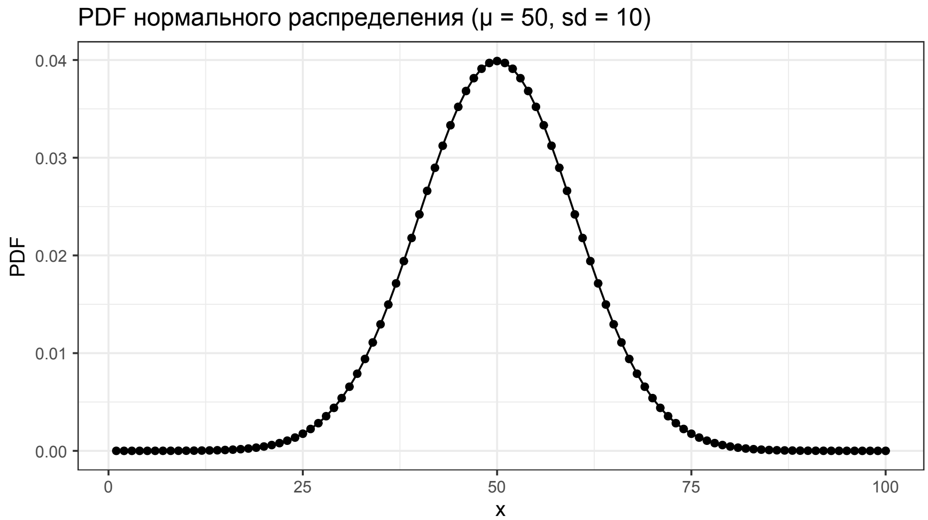
2.1 Распределения в R

В R встроено какое-то количество известных распределений. Все они представлены четырьмя функциями:

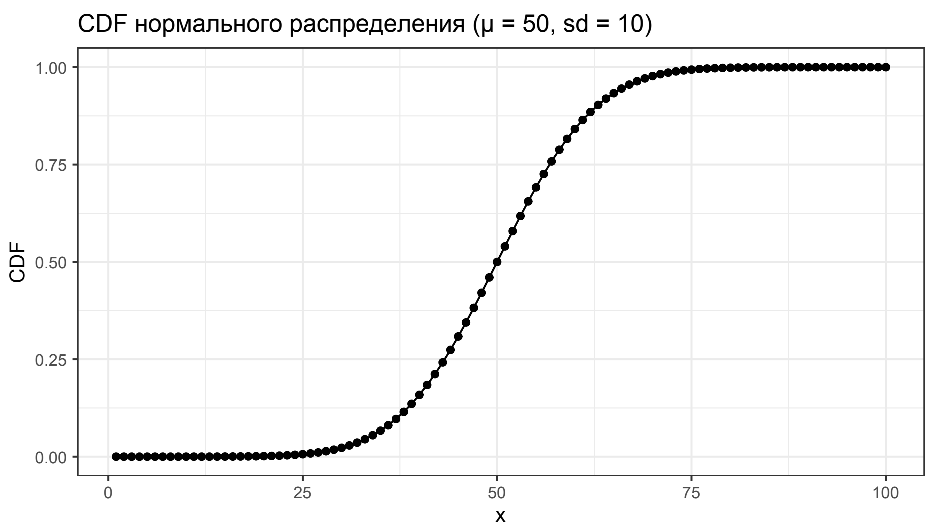
- `d...` (функция плотности, probability density function),
- `p...` (функция распределения, cumulative distribution function) — интеграл площади под кривой от начала до указанной квантили
- `q...` (обратная функции распределения, inverse cumulative distribution function) — значение *p*-той квантили распределения
- и `r...` (рандомные числа из заданного распределения).

Рассмотрим все это на примере нормального распределения.

```
tibble(x = 1:100,
       PDF = dnorm(x = x, mean = 50, sd = 10)) %>%
  ggplot(aes(x, PDF))+
  geom_point()+
  geom_line()+
  labs(title = "PDF нормального распределения (μ = 50, sd = 10)")
```

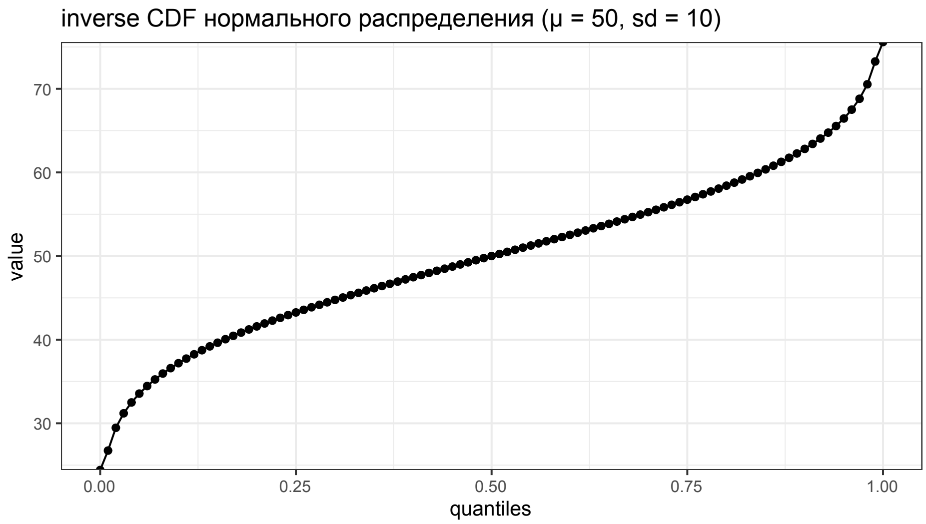



```
tibble(x = 1:100,
       CDF = pnorm(x, mean = 50, sd = 10)) %>%
  ggplot(aes(x, CDF))+
  geom_point()+
  geom_line()+
  labs(title = "CDF нормального распределения (μ = 50, sd = 10)")
```

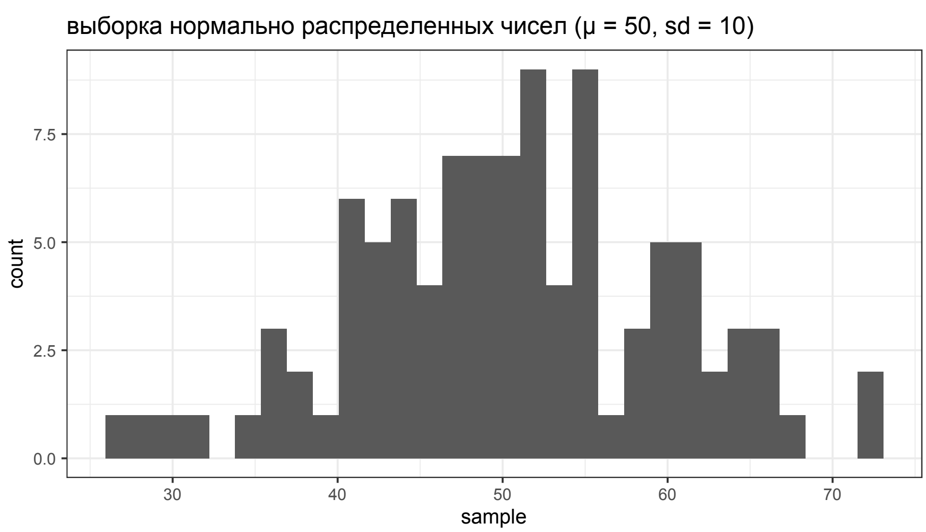



```
tibble(quantiles = seq(0, 1, by = 0.01),
       value = qnorm(quantiles, mean = 50, sd = 10)) %>%
  ggplot(aes(quantiles, value))+
```

```
  geom_point()+
  geom_line()+
  labs(title = "inverse CDF нормального распределения (μ = 50, sd = 10)")
```




```
tibble(sample = rnorm(100, mean = 50, sd = 10)) %>%
  ggplot(aes(sample))+
  geom_histogram()+
  labs(title = "выборка нормально распределенных чисел (μ = 50, sd = 10)")
```



Если не использовать `set.seed()`, то результат работы рандомизатора нельзя будет повторить.

Какое значение имеет 25% квантиль нормального распределения со средним в 20 и стандартным отклонением 90? Ответ округлите до трех знаков после запятой.

Данные из базы данных фонетических инвентарей PHOIBLE [@phoible], достаточно сильно упрощая, можно описать нормальным распределением со средним 35 фонем и стандартным отклонением 13. Если мы ничего не знаем про язык, оцените с какой вероятностью, согласно этой модели произвольно взятый язык окажется в промежутке между 25 и 50 фонемами? Ответ округлите до трех знаков после запятой.

Какие есть недостатки у модели из предыдущего задания?

2.2 Дискретные переменные

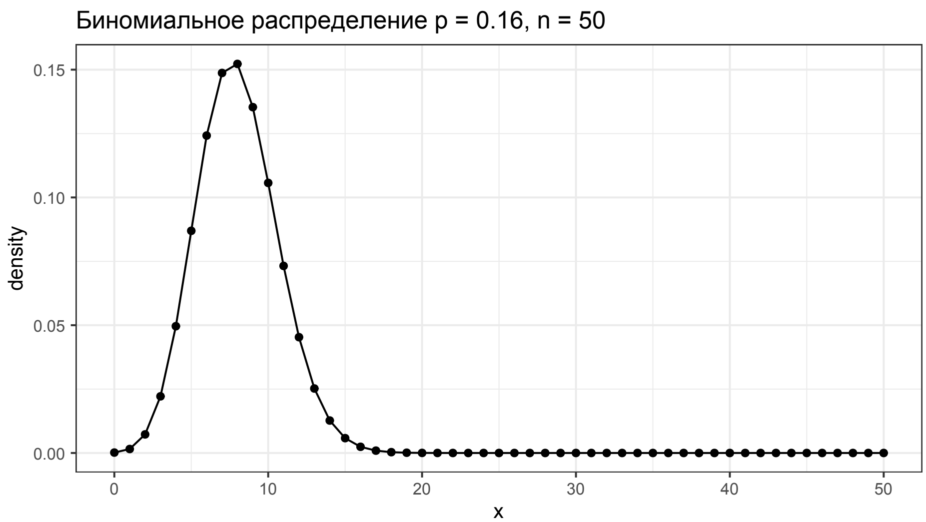
2.2.1 Биномиальное распределение

Биномиальное распределение — распределение количетсва успехов эксперементов Бернулли из *n* попыток с вероятностью успеха *p*.

$$P(k|n,p) = \frac{n!}{k!(n-k)!} \times p^k \times (1-p)^{n-k} = \binom{n}{k} \times p^k \times (1-p)^{n-k}$$

$$0 \leq p \leq 1; n, k > 0$$

```
tibble(x = 0:50,
       density = dbinom(x = x, size = 50, prob = 0.16)) %>%
  ggplot(aes(x, density))+
  geom_point()+
  geom_line()+
  labs(title = "Биномиальное распределение p = 0.16, n = 50")
```



Немного упрощая данные из статьи [@rosenbach03: 394], можно сказать что носители британского английского предпочитают *s*-генитив (90%) *of*-генитиву (10%). Какова вероятность, согласно этим данным, что в интервью британского актера из 118 контекстов будет 102 *s*-генитивов? Ответ округлите до трёх или менее знаков после запятой.

А какое значение количества *s*-генитивов наиболее ожидаемо, согласно этой модели?

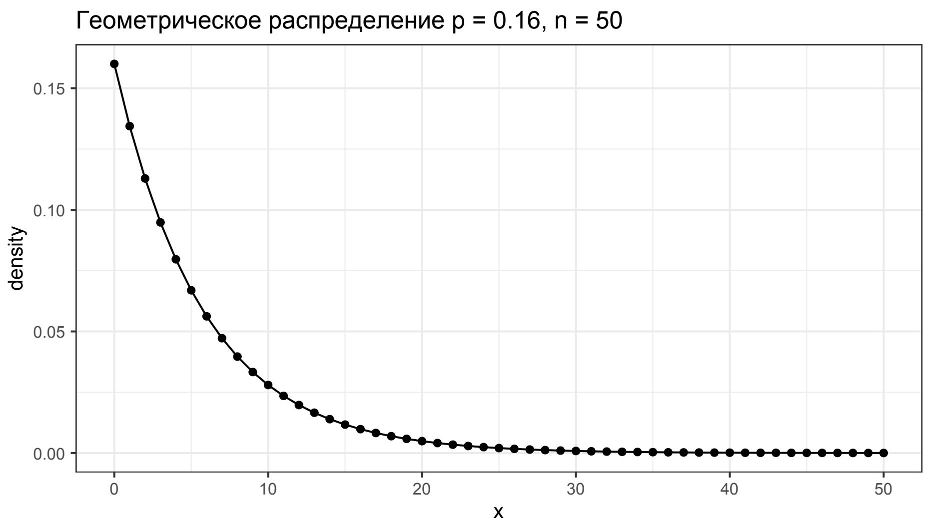
2.2.2 Геометрическое распределение

Геометрическое распределение — распределение количетсва эксперементов Бернулли с вероятностью успеха p до первого успеха.

$$P(k|p) = (1 - p)^k \times p$$

$$k \in \{1, 2, \dots\}$$

```
tibble(x = 0:50,
       density = dgeom(x = x, prob = 0.16)) %>%
  ggplot(aes(x, density))+
  geom_point()+
  geom_line()+
  labs(title = "Геометрическое распределение p = 0.16, n = 50")
```



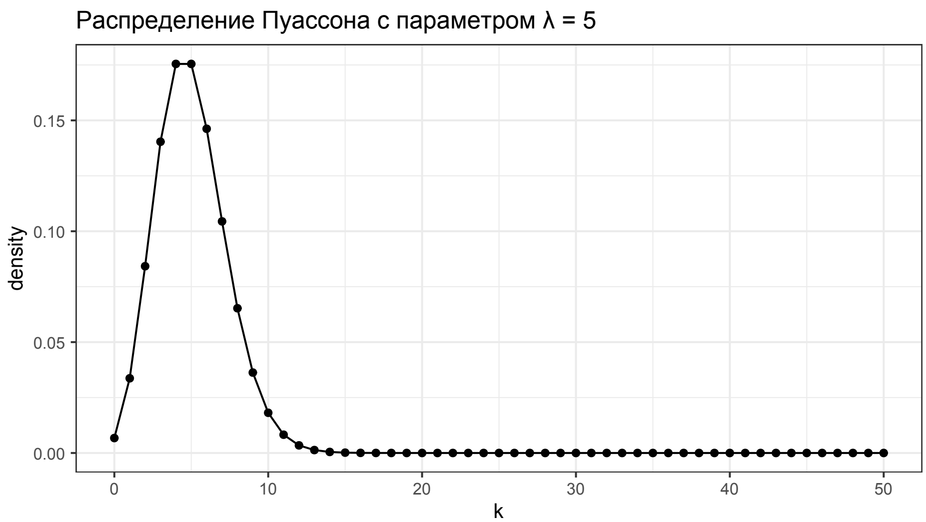
Приняв модель из [@rosenbach03: 394], какова вероятность, что в интервью с британским актером первый *of*-генитив будет третьим по счету?

2.2.3 Распределение Пуассона

Распределение дискретной переменной, обозначающей количество случаев k некоторого события, которое происходит с некоторой заданной частотой λ.

$$P(\lambda) = \frac{e^{-\lambda} \times \lambda^k}{k!}$$

```
tibble(k = 0:50,
       density = dpois(x = k, lambda = 5)) %>%
  ggplot(aes(k, density))+
  geom_point()+
  geom_line()+
  labs(title = "Распределение Пуассона с параметром λ = 5")
```

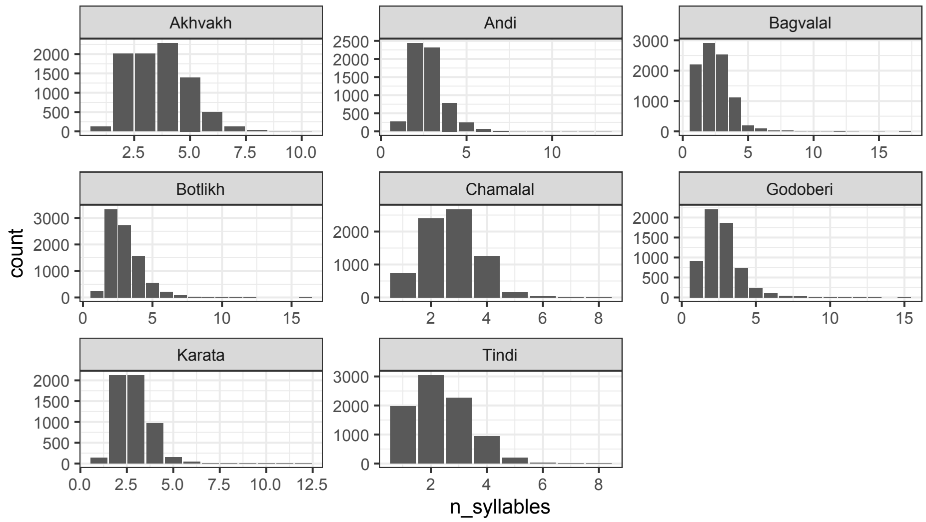


Параметр λ в модели Пуассона одновременно является и средним, и дисперсией.

Попробуем воспользоваться распределением Пуассона для моделирования количества слогов в андийском языке. Количество слогов – это всегда натуральное число (т. е. не бывает 2.5 слогов, не бывает -3 слогов и т. д., но в теории может быть 0 слогов), так что модель Пуассона здесь применима. Согласно модели Пуассона все слова независимо друг от друга получают сколько-то слогов согласно распределению Пуассона. Посмотрим на данные:

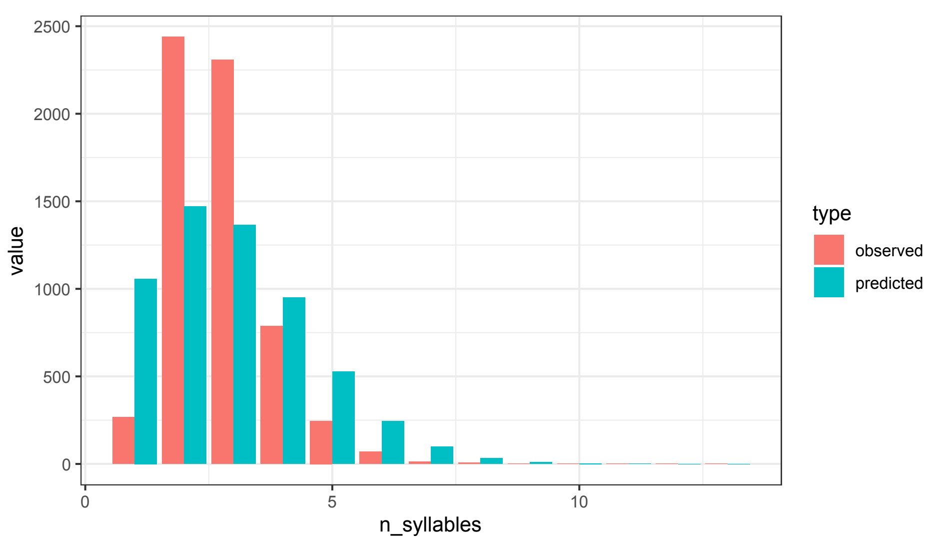
```
andic_syllables <- read_csv("https://raw.githubusercontent.com/agricolamz/2021_da4l/master/data/andic_syllables.csv")

andic_syllables %>%
  ggplot(aes(n_syllables, count))+
  geom_col()+
  facet_wrap(~language, scales = "free")
```

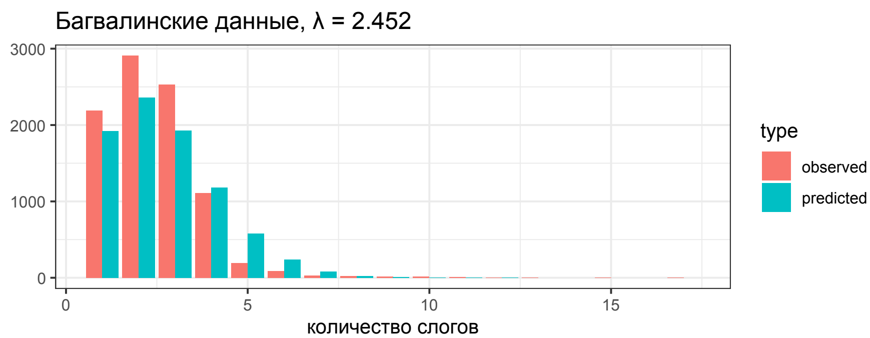


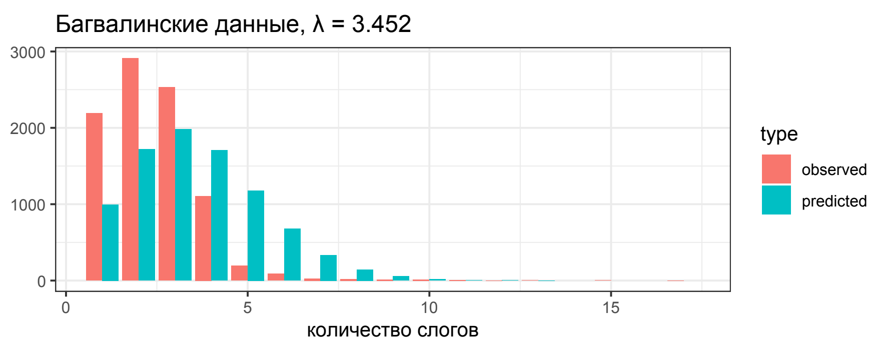
Птичка напела (мы научимся узнавать, откуда птичка это знает на следующем занятии), что андийские данные можно описать при помощи распределения Пуассона с параметром $\lambda = 2.783$.

```
andic_syllables %>%
  filter(language == "Andi") %>%
  rename(observed = count) %>%
  mutate(predicted = dpois(n_syllables, lambda = 2.783)*sum(observed)) %>%
  pivot_longer(names_to = "type", values_to = "value", cols = c(observed, predicted)) %>%
  ggplot(aes(n_syllables, value, fill = type))+
  geom_col(position = "dodge")
```



На графиках ниже представлены предсказания трех Пуассоновских моделей, какая кажется лучше?





Выше было написано:

> Согласно модели Пуассона все слова **независимо друг от друга** получают сколько-то слогов согласно распределению Пуассона.

Какие проблемы есть у предположения о независимости друг от друга количества слогов разных слов в словаре?

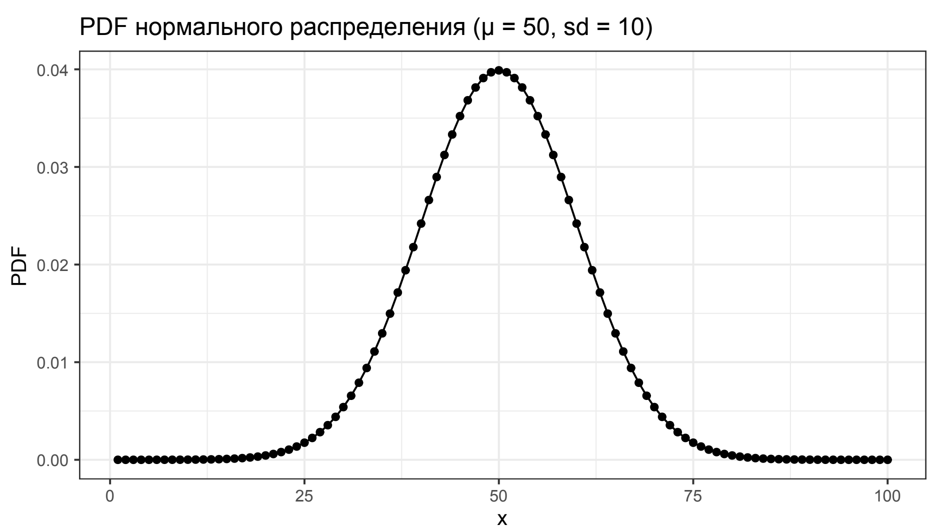
2.3 Числовые переменные

2.3.1 Нормальное распределение

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} \times e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

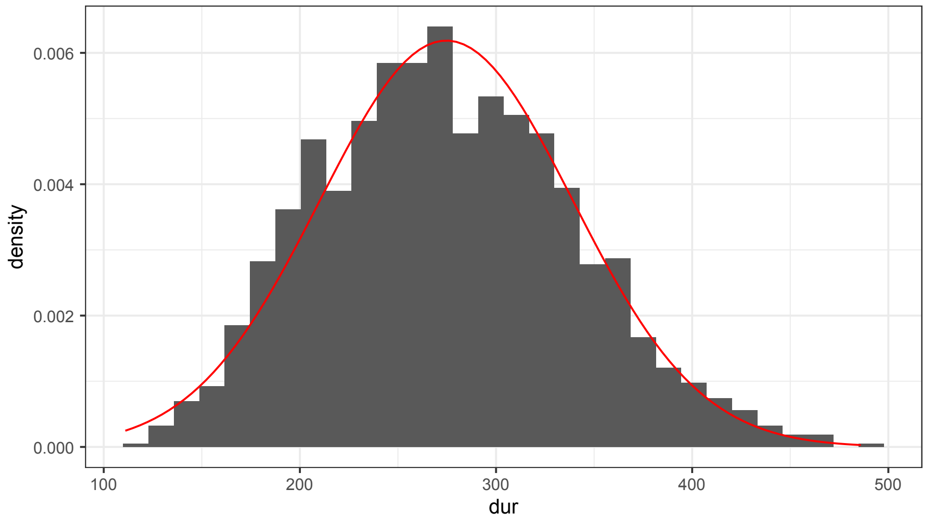
$$\mu \in \mathbb{R}; \sigma^2 > 0$$

```
tibble(x = 1:100,
       PDF = dnorm(x = x, mean = 50, sd = 10)) %>%
  ggplot(aes(x, PDF))+
  geom_point()+
  geom_line()+
  labs(title = "PDF нормального распределения (μ = 50, sd = 10)")
```



Птичка напела, что длительность гласных американского английского из (Hillenbrand et al., 1995) можно описать нормальным распределением с параметрами $\mu = 274.673$ и $\sigma = 64.482$. Посмотрим, как можно совместить данные и это распределение:

```
vowels <- read_csv("https://raw.githubusercontent.com/agricolamz/2021_da4l/master/data/phonTools_hillenbrand_1995.csv")
vowels %>%
  ggplot(aes(dur)) +
  geom_histogram(aes(y =..density..)) + # обратите внимание на аргумент ..density..
  stat_function(fun = dnorm, args = list(mean = 274.673, sd = 64.482), color = "red")
```

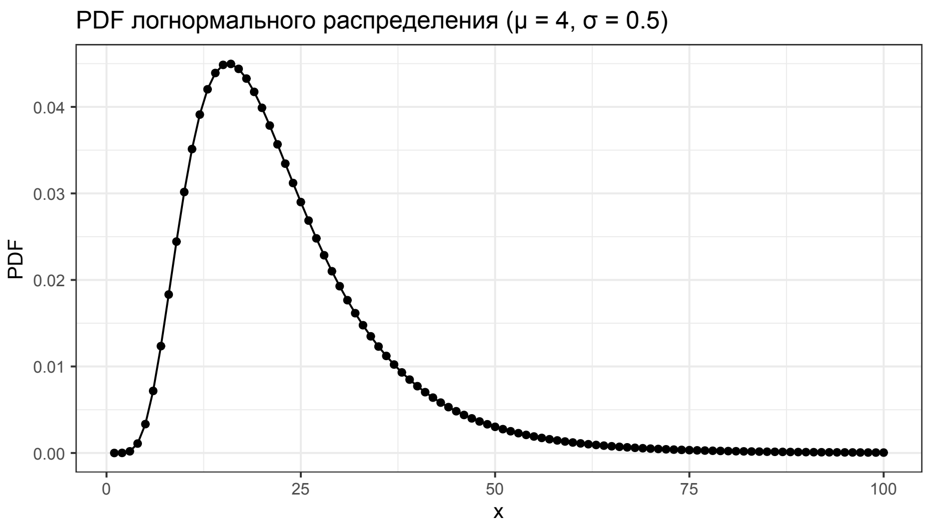


2.3.2 Логнормальное распределение

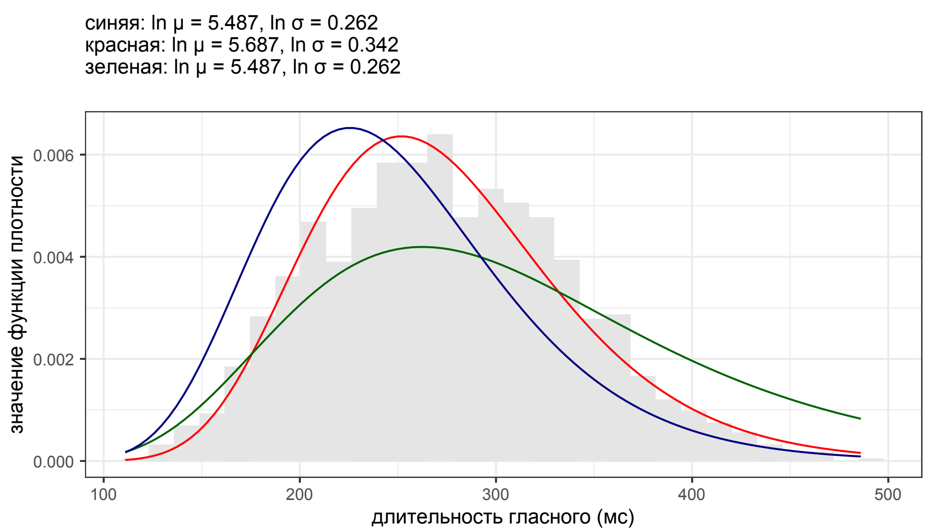
$$P(x) = \frac{1}{\sqrt{x\sigma2\pi}} \times e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}}$$

$$\mu \in \mathbb{R}; \sigma^2 > 0$$

```
tibble(x = 1:100,
       PDF = dlnorm(x = x, mean = 3, sd = 0.5)) %>%
  ggplot(aes(x, PDF))+
  geom_point()+
  geom_line()+
  labs(title = "PDF логнормального распределения (μ = 4, σ = 0.5)")
```



Какая из логнормальных моделей для длительности гласных американского английского из [@hillenbrand95] лучше подходит к данным? Попробуйте самостоятельно построить данный график.



2.3.3 Что еще почитать про распределения?

Люди придумали очень много разных распределений. Стоит, наверное, также понимать, что распределения не существуют отдельно в вакууме: многие из них математически связаны друг с другом. Про это можно посмотреть вот здесь[1] или здесь[2].

[1] http://www.math.wm.edu/~leemis/chart/UDR/UDR.html

[2] https://en.wikipedia.org/wiki/Relationships_among_probability_distributions

Глава 3

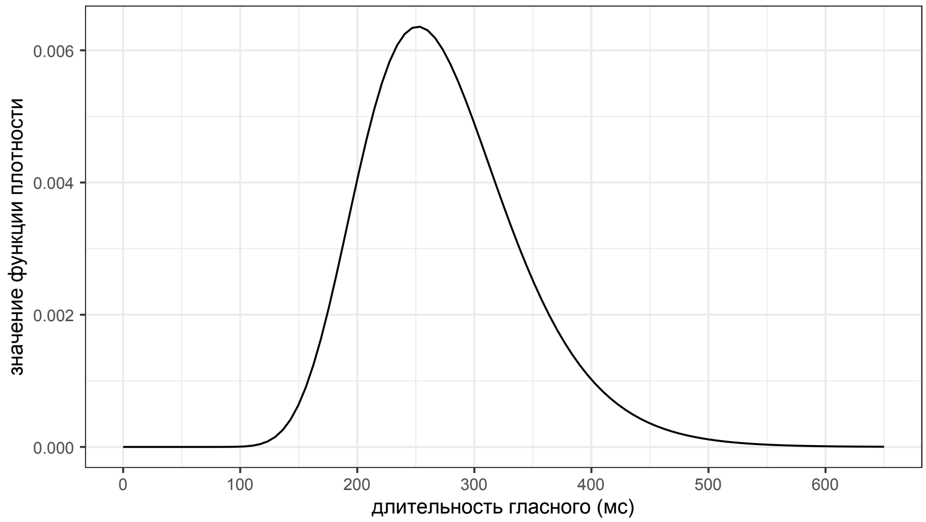
Метод максимального правдоподобия

3.1 Оценка вероятности

```
library(tidyverse)
```

Когда у нас задано некоторое распределение, мы можем задавать к нему разные вопросы. Например, если мы верим что длительность гласных американского английского из (Hillenbrand et al., 1995) можно описать логнормальным распределением с параметрами $\ln \mu = 5.587$ и $\ln \sigma = 0.242$, то мы можем делать некотрые предсказания относительно интересующей нас переменной.

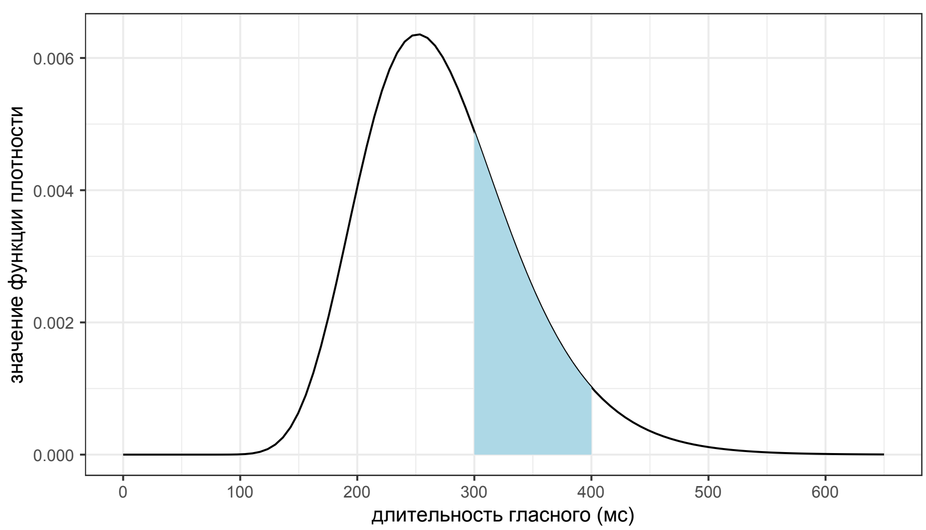
```
ggplot() +
  stat_function(fun = dlnorm, args = list(mean = 5.587, sd = 0.242))+
  scale_x_continuous(breaks = 0:6*100, limits = c(0, 650))+
  labs(x = "длительность гласного (мс)",
       y = "значение функции плотности")
```



Если принять на веру, что логнормальное распределение с параметрами $\ln \mu = 5.587$ и $\ln \sigma = 0.242$ описывает данные длительности гласных американского английского из [@hillenbrand95], то какова вероятность наблюдать значения между 300 и 400 мс? То же самое можно записать, используя математическую нотацию:

$$P\left(X \in [300,\ 400] | X \sim \ln \mathcal{N}(\ln \mu = 5.587, \ln \sigma = 0.242)\right) = ??$$

Ответ округлите до трех и меньше знаков после запятой.



Если принять на веру, что биномиальное распределение с параметрами $p = 0.9$ описывает, согласно [@rosenbach03: 394] употребление *s*-генитивов в британском английском, то какова вероятность наблюдать значения между 300 и 350 генитивов в интервью, содержащее 400 генитивных контекстов? То же самое можно записать, используя математическую нотацию:

$$P\left(X \in [300,\ 350] | X \sim Binom(n = 400, p = 0.9)\right) = ??$$

Ответ округлите до трех и меньше знаков после запятой.

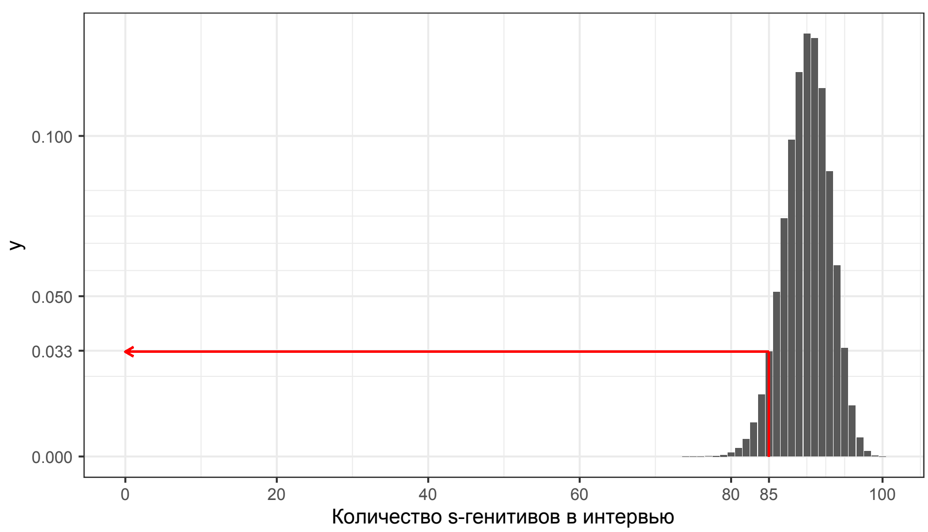
3.2 Функция правдоподобия

Если при поиске вероятностей, мы предполагали, что данные нам **неизвестны**, а распределение и его параметры **известны**, то функция правдоподобия позволяет этот процесс перевернуть, запустив поиск параметров распределения, при изветсных данных и семье распределения:

$$L\left(X \sim Distr(...)|x\right) = ...$$

Таким образом получается, что на основании функции плотности мы можем сравнивать, какой параметр лучше подходит к нашим данным.

Для примера рассмотрим наш s-генетив: мы провели интервью и нам встретилось 85 *s*-генетив из 100 случаев всех генетивов. Насколько хорошо подходит нам распределение с параметром *p* = 0.9?



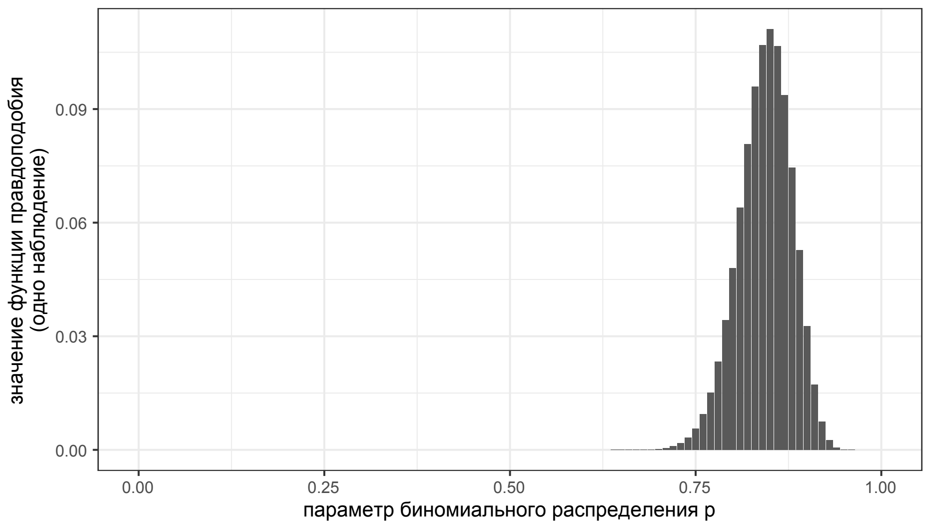
Ответ:

```
dbinom(85, 100, 0.9)
```

```
[1] 0.03268244
```

Представим теперь это как функцию от параметра p:

```
tibble(p = seq(0, 1, by = 0.01)) %>%
  ggplot(aes(p)) +
  stat_function(fun = function(p) dbinom(85, 100, p), geom = "col")+
  labs(x = "параметр биномиального распределения p",
       y = "значение функции правдоподобия\n(одно наблюдение)")
```

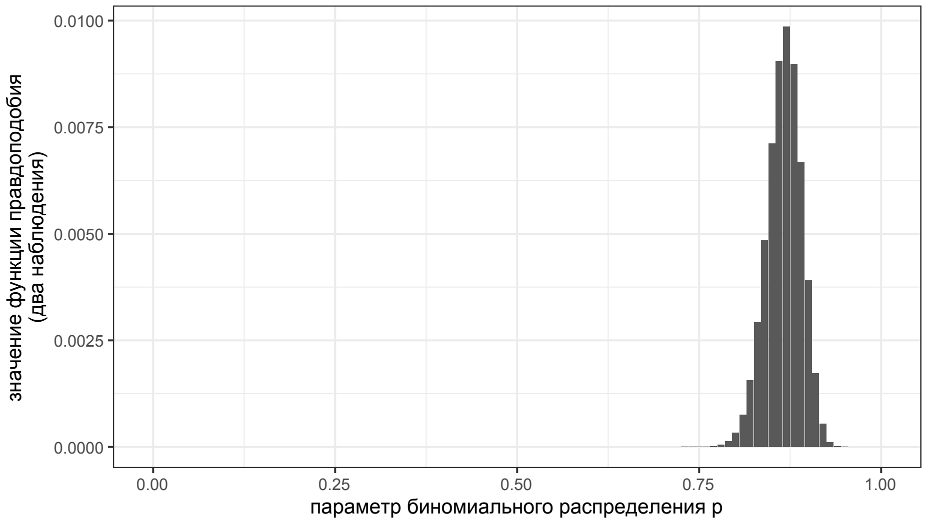


А что если мы располагаем двумя интервью одного актера? В первом на сто генитивов пришлось 85 s-генитивов, а во втором – 89. В таком случае, также как и с вероятностью наступления двух независимых событий, значения функции плотности перемножаются.

```
dbinom(85, 100, 0.9)*dbinom(89, 100, 0.9)
```

```
[1] 0.003917892
```

```
tibble(p = seq(0, 1, by = 0.01)) %>%
  ggplot(aes(p)) +
  stat_function(fun = function(p) dbinom(85, 100, p)*dbinom(89, 100, p), geom = "col")+
  labs(x = "параметр биномиального распределения p",
       y = "значение функции правдоподобия\n(два наблюдения)")
```



В итоге:

- вероятность — P(data|distribution)
- правдоподобие — L(distribution|data)

Интеграл распределения/сумма значений вероятностей равен/на 1. Интеграл распределения/сумма значений правдоподобия может быть не равен/на 1[1].

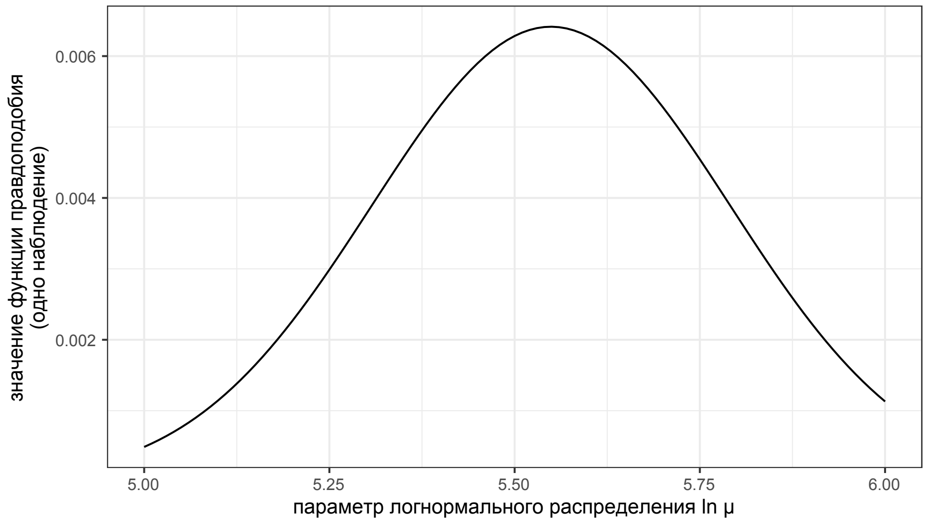
3.3 Пример с непрерывным распределением

Мы уже обсуждали, что длительность гласных американского английского из (Hillenbrand et al., 1995) можно описать логнормальным распределением с параметрами $\ln\mu$ и $\ln\sigma$. Предположим, что $\ln\sigma = 0.342$, построим функцию правдоподобия для $\ln\mu$:

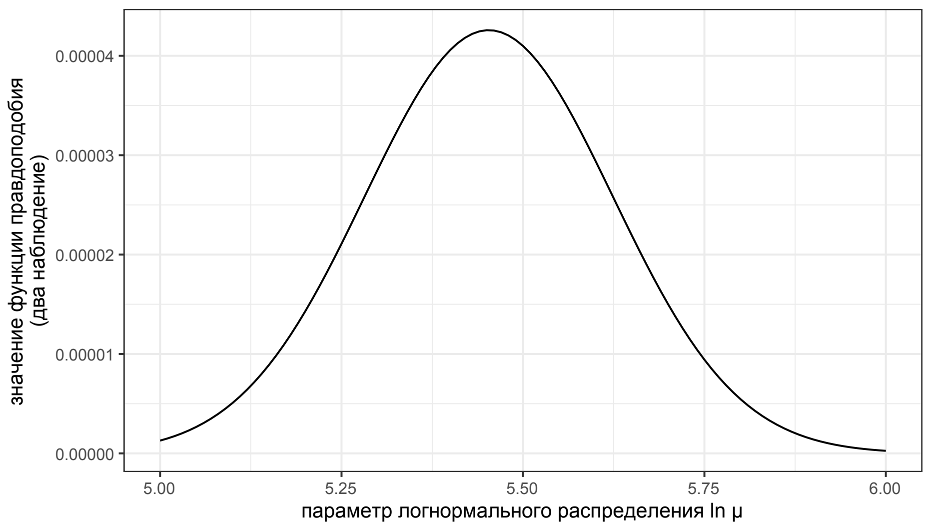
```
vowels <- read_csv("https://raw.githubusercontent.com/agricolamz/2021_da4l/master/data/phonTools_hillenbrand_1995.csv")

tibble(ln_mu = seq(5, 6, by = 0.001)) %>%
  ggplot(aes(ln_mu)) +
  stat_function(fun = function(ln_mu) dlnorm(vowels$dur[1], meanlog = ln_mu, sdlog = 0.242))+
  labs(x = "параметр логнормального распределения ln μ",
       y = "значение функции правдоподобия\n(одно наблюдение)")
```

[1] https://stats.stackexchange.com/a/31241/225843

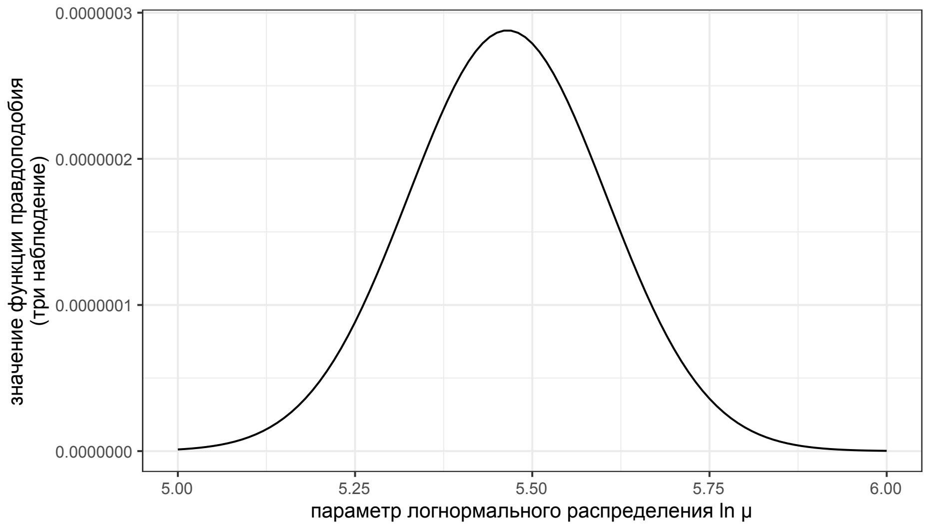



```
tibble(ln_mu = seq(5, 6, by = 0.001)) %>%
  ggplot(aes(ln_mu)) +
  stat_function(fun = function(ln_mu) dlnorm(vowels$dur[1], meanlog = ln_mu, sdlog = 0.242)*dlnorm(vowels$dur[2], meanlog = ln_mu,
  labs(x = "параметр логнормального распределения ln μ",
       y = "значение функции правдоподобия\n(два наблюдение)")
```




```
tibble(ln_mu = seq(5, 6, by = 0.001)) %>%
  ggplot(aes(ln_mu)) +
  stat_function(fun = function(ln_mu) dlnorm(vowels$dur[1], meanlog = ln_mu, sdlog = 0.242)*dlnorm(vowels$dur[2], meanlog = ln_mu,
  labs(x = "параметр логнормального распределения ln μ",
```

```
    y = "значение функции правдоподобия\n(три наблюдение)")
```



Для простоты в начале я использовал зафиксировал один из параметров логнормального распредления: лог стандартное отклонение. Конечно, это совсем необязательно делать: можно создать матрицу значений лог среднего и лог стандартного отклонения и получить для каждой ячейки матрицы значения функции правдоподобия.

3.4 Метод максимального правдоподобия (MLE)

Функция правдподобия позволяет подбирать параметры распределения. Оценка параметров распределения при помощи функции максимального правдоподобия получила названия метод максимального правдоподобия. Его я и использовал ранее для того, чтобы получить значения распределений для заданий из первого занятия:

- данные длительности американских гласных из (Hillenbrand et al., 1995) и логнормальное распределение

```
library(fitdistrplus)
fitdist(vowels$dur, distr = 'lnorm', method = 'mle')
```

```
Fitting of the distribution ' lnorm ' by maximum likelihood
Parameters:
         estimate  Std. Error
meanlog 5.5870359 0.005935135
sdlog   0.2423978 0.004196453
```

- количество андийских слогов в словах и распределение Пуассона

```
andic_syllables <- read_csv("https://raw.githubusercontent.com/agricolamz/2021_da4l/master/data/andic_syllables.csv")

andic_syllables %>%
  filter(language == "Andi") %>%
  uncount(count) %>%
  pull(n_syllables) %>%
  fitdist(distr = 'pois', method = 'mle')
```

```
Fitting of the distribution ' pois ' by maximum likelihood
Parameters:
       estimate Std. Error
lambda 2.782715 0.02128182
```

- Есть и другие методы оценки параметров.
- Метод максимального правдоподобия может быть чувствителен к размеру выборки.

Литература

Hillenbrand, J., Getty, L. A., Clark, M. J., and Wheeler, K. (1995). Acoustic characteristics of American English vowels. *The Journal of the Acoustical society of America*, 97(5):3099–3111.