

Assignment 1

1 Blood types genetics

Ex 1.1

$$\Omega = \{I^A, I^A i, I^B I^B, I^B i, I^A I^B, ii\}$$

Ex 1.2

$$P(ii) = 0.25$$

$$P(I^A i, ii) = 0.75$$

Ex 1.3

$$P(C = I^A I^B) = 0.25$$

Ex 1.4

Phenotypes: AB, A, B

Distribution: AB = 0.25; A = 0.375; B = 0.375

Ex 1.5

$$P(I^B i, I^B I^B \mid B = I^A I^B) = 0.375$$

1.1 Some simulation in R

Ex 2.1

```
sample(c("ii", "iB", "iA", "BB", "AA", "AB"), size = 1, replace = TRUE)
```

Ex 2.2

```
offspringG <- function(Parent1, Parent2){
  Allele1 <- sample(unlist(strsplit(Parent1, split = "")), replace = TRUE, size=1)
  Allele2 <- sample(unlist(strsplit(Parent2, split = "")), replace = TRUE, size=1)
  return(paste(Allele1, Allele2, sep = ""))
}

offspringP <- function(Genotype){
  Phenotype <- ""
  if (Genotype == "ii"){
    Phenotype <- "Type 0"
  }
  if (Genotype == "AB" | Genotype == "BA"){
    Phenotype <- "Type AB"
  }
}
```

```

if (Genotype == "iB" | Genotype == "Bi" | Genotype == "BB"){
  Phenotype <- "Type B"
}
if (Genotype == "iA" | Genotype == "Ai" | Genotype == "AA"){
  Phenotype <- "Type A"
}
return(Phenotype)
}

```

Ex 2.3

```

offspringiAiA <- c()
for(i in 1:1000){
  offspringiAiA[i] <- offspringG("iA", "iA")
}
sum(offspringiAiA == "ii")/length(offspringiAiA)

```

```
## [1] 0.248
```

```
sum(offspringiAiA != "AA")/length(offspringiAiA)
```

```
## [1] 0.744
```

Ex 2.4

```

secondGen <- c()

for(i in 1:1000){
  A <- offspringG("iA", "iB")
  B <- offspringG("AB", "AB")
  C <- offspringP(offspringG(A, B))
  secondGen[i] <- C
}

```

```
secondGen
```

```

##      [1] "Type B" "Type B" "Type A" "Type B" "Type AB" "Type B" "Type A"
##      [8] "Type A" "Type B" "Type A" "Type B" "Type AB" "Type B" "Type A"
##     [15] "Type B" "Type A" "Type A" "Type AB" "Type A" "Type A" "Type A"
##     [22] "Type B" "Type B" "Type AB" "Type A" "Type A" "Type A" "Type B"
##     [29] "Type A" "Type A" "Type A" "Type A" "Type A" "Type A" "Type B"
##     [36] "Type AB" "Type AB" "Type AB" "Type B" "Type B" "Type B" "Type B"
##     [43] "Type A" "Type A" "Type A" "Type B" "Type A" "Type AB" "Type A"
##     [50] "Type B" "Type AB" "Type A" "Type A" "Type A" "Type A" "Type B"
##     [57] "Type AB" "Type A" "Type AB" "Type B" "Type A" "Type A" "Type AB"
##     [64] "Type A" "Type B" "Type B" "Type A" "Type B" "Type B" "Type AB"
##     [71] "Type B" "Type AB" "Type A" "Type B" "Type B" "Type B" "Type A"
##     [78] "Type AB" "Type B" "Type AB" "Type B" "Type A" "Type A" "Type B"
##     [85] "Type A" "Type B" "Type A" "Type B" "Type AB" "Type A" "Type B"
##     [92] "Type A" "Type A" "Type B" "Type A" "Type A" "Type A" "Type B"
##     [99] "Type B" "Type A" "Type B" "Type B" "Type AB" "Type B" "Type B"
##    [106] "Type B" "Type A" "Type B" "Type B" "Type A" "Type B" "Type AB"
##    [113] "Type B" "Type A" "Type A" "Type AB" "Type AB" "Type A" "Type AB"
##    [120] "Type B" "Type A" "Type A" "Type A" "Type B" "Type B" "Type B"

```

```

## [127] "Type A" "Type AB" "Type A" "Type AB" "Type B" "Type AB" "Type A"
## [134] "Type AB" "Type B" "Type A" "Type A" "Type A" "Type A" "Type B"
## [141] "Type A" "Type AB" "Type B" "Type A" "Type B" "Type B" "Type B"
## [148] "Type A" "Type AB" "Type A" "Type A" "Type A" "Type A" "Type B"
## [155] "Type B" "Type B" "Type A" "Type A" "Type A" "Type B" "Type AB"
## [162] "Type B" "Type B" "Type A" "Type AB" "Type A" "Type AB" "Type B"
## [169] "Type AB" "Type B" "Type A" "Type A" "Type B" "Type A" "Type AB"
## [176] "Type A" "Type A" "Type A" "Type AB" "Type B" "Type B" "Type AB"
## [183] "Type B" "Type B" "Type B" "Type B" "Type A" "Type AB" "Type A"
## [190] "Type B" "Type B" "Type AB" "Type B" "Type B" "Type B" "Type B"
## [197] "Type A" "Type A" "Type A" "Type AB" "Type B" "Type A" "Type B"
## [204] "Type B" "Type A" "Type B" "Type AB" "Type B" "Type A" "Type AB"
## [211] "Type A" "Type A" "Type AB" "Type A" "Type AB" "Type AB" "Type AB"
## [218] "Type AB" "Type AB" "Type A" "Type AB" "Type A" "Type AB" "Type AB"
## [225] "Type B" "Type A" "Type B" "Type B" "Type A" "Type A" "Type B"
## [232] "Type A" "Type B" "Type B" "Type AB" "Type AB" "Type B" "Type B"
## [239] "Type B" "Type A" "Type A" "Type A" "Type B" "Type B" "Type A"
## [246] "Type B" "Type A" "Type B" "Type AB" "Type AB" "Type A" "Type A"
## [253] "Type B" "Type A" "Type B" "Type B" "Type AB" "Type B" "Type A"
## [260] "Type AB" "Type AB" "Type AB" "Type A" "Type AB" "Type AB" "Type AB"
## [267] "Type A" "Type AB" "Type A" "Type A" "Type B" "Type AB" "Type B"
## [274] "Type AB" "Type B" "Type B" "Type B" "Type B" "Type A" "Type B"
## [281] "Type B" "Type B" "Type B" "Type A" "Type A" "Type A" "Type B"
## [288] "Type B" "Type AB" "Type AB" "Type B" "Type A" "Type AB" "Type B"
## [295] "Type A" "Type AB" "Type B" "Type B" "Type B" "Type B" "Type AB"
## [302] "Type B" "Type AB" "Type A" "Type B" "Type B" "Type AB" "Type A"
## [309] "Type B" "Type B" "Type A" "Type AB" "Type A" "Type B" "Type AB"
## [316] "Type AB" "Type B" "Type A" "Type AB" "Type B" "Type B" "Type B"
## [323] "Type B" "Type A" "Type AB" "Type A" "Type A" "Type A" "Type AB"
## [330] "Type A" "Type B" "Type B" "Type A" "Type A" "Type A" "Type A"
## [337] "Type A" "Type B" "Type B" "Type AB" "Type A" "Type B" "Type A"
## [344] "Type B" "Type A" "Type AB" "Type AB" "Type AB" "Type AB" "Type B"
## [351] "Type AB" "Type A" "Type AB" "Type B" "Type AB" "Type AB" "Type B"
## [358] "Type B" "Type AB" "Type A" "Type A" "Type AB" "Type B" "Type A"
## [365] "Type A" "Type B" "Type A" "Type B" "Type A" "Type A" "Type A"
## [372] "Type A" "Type AB" "Type AB" "Type A" "Type A" "Type B" "Type A"
## [379] "Type AB" "Type B" "Type B" "Type B" "Type A" "Type B" "Type A"
## [386] "Type A" "Type B" "Type B" "Type AB" "Type A" "Type B" "Type B"
## [393] "Type B" "Type A" "Type A" "Type A" "Type A" "Type AB" "Type A"
## [400] "Type B" "Type AB" "Type B" "Type A" "Type A" "Type AB" "Type B"
## [407] "Type B" "Type AB" "Type A" "Type B" "Type AB" "Type A" "Type B"
## [414] "Type A" "Type B" "Type B" "Type B" "Type A" "Type A" "Type AB"
## [421] "Type A" "Type A" "Type A" "Type AB" "Type AB" "Type A" "Type B"
## [428] "Type A" "Type A" "Type B" "Type A" "Type AB" "Type A" "Type AB"
## [435] "Type B" "Type A" "Type AB" "Type AB" "Type A" "Type A" "Type A"
## [442] "Type B" "Type AB" "Type B" "Type A" "Type AB" "Type AB" "Type AB"
## [449] "Type B" "Type A" "Type B" "Type AB" "Type A" "Type AB" "Type A"
## [456] "Type A" "Type B" "Type AB" "Type A" "Type B" "Type AB" "Type A"
## [463] "Type B" "Type A" "Type A" "Type AB" "Type A" "Type A" "Type B"
## [470] "Type B" "Type A" "Type A" "Type B" "Type B" "Type B" "Type A"
## [477] "Type AB" "Type AB" "Type A" "Type AB" "Type A" "Type A" "Type A"
## [484] "Type B" "Type B" "Type B" "Type AB" "Type A" "Type B" "Type B"
## [491] "Type AB" "Type A" "Type B" "Type A" "Type AB" "Type B" "Type B"
## [498] "Type B" "Type AB" "Type B" "Type A" "Type A" "Type AB" "Type A"

```

```

## [505] "Type AB" "Type AB" "Type A" "Type A" "Type A" "Type AB" "Type AB"
## [512] "Type B" "Type B" "Type A" "Type B" "Type A" "Type B" "Type B"
## [519] "Type B" "Type B" "Type B" "Type AB" "Type B" "Type A" "Type A"
## [526] "Type A" "Type AB" "Type B" "Type AB" "Type A" "Type B" "Type B"
## [533] "Type AB" "Type B" "Type B" "Type A" "Type A" "Type AB" "Type A"
## [540] "Type A" "Type B" "Type AB" "Type B" "Type A" "Type AB" "Type A"
## [547] "Type B" "Type B" "Type AB" "Type B" "Type A" "Type AB" "Type A"
## [554] "Type A" "Type B" "Type B" "Type B" "Type A" "Type B" "Type B"
## [561] "Type AB" "Type B" "Type B" "Type B" "Type A" "Type B" "Type A"
## [568] "Type B" "Type AB" "Type AB" "Type A" "Type AB" "Type AB" "Type A"
## [575] "Type AB" "Type A" "Type AB" "Type A" "Type B" "Type AB" "Type AB"
## [582] "Type B" "Type A" "Type A" "Type AB" "Type B" "Type AB" "Type AB"
## [589] "Type AB" "Type B" "Type AB" "Type A" "Type A" "Type B" "Type B"
## [596] "Type B" "Type A" "Type B" "Type B" "Type AB" "Type B" "Type B"
## [603] "Type A" "Type B" "Type B" "Type AB" "Type A" "Type A" "Type B"
## [610] "Type A" "Type B" "Type A" "Type AB" "Type AB" "Type A" "Type B"
## [617] "Type B" "Type AB" "Type A" "Type A" "Type AB" "Type AB" "Type AB"
## [624] "Type A" "Type A" "Type B" "Type A" "Type B" "Type B" "Type B"
## [631] "Type AB" "Type A" "Type AB" "Type B" "Type AB" "Type A" "Type B"
## [638] "Type A" "Type B" "Type A" "Type B" "Type A" "Type AB" "Type A"
## [645] "Type B" "Type B" "Type AB" "Type A" "Type A" "Type AB" "Type A"
## [652] "Type B" "Type AB" "Type A" "Type B" "Type A" "Type AB" "Type A"
## [659] "Type AB" "Type B" "Type AB" "Type B" "Type AB" "Type AB" "Type AB"
## [666] "Type AB" "Type A" "Type B" "Type A" "Type AB" "Type AB" "Type B"
## [673] "Type AB" "Type A" "Type A" "Type B" "Type A" "Type B" "Type B"
## [680] "Type A" "Type AB" "Type A" "Type B" "Type A" "Type B" "Type A"
## [687] "Type AB" "Type AB" "Type AB" "Type B" "Type AB" "Type A" "Type A"
## [694] "Type A" "Type B" "Type A" "Type B" "Type B" "Type AB" "Type B"
## [701] "Type AB" "Type AB" "Type AB" "Type B" "Type A" "Type AB" "Type B"
## [708] "Type A" "Type B" "Type A" "Type AB" "Type A" "Type B" "Type AB"
## [715] "Type AB" "Type B" "Type A" "Type B" "Type A" "Type B" "Type AB"
## [722] "Type AB" "Type B" "Type B" "Type B" "Type B" "Type B" "Type B"
## [729] "Type A" "Type A" "Type B" "Type B" "Type AB" "Type A" "Type A"
## [736] "Type B" "Type B" "Type B" "Type AB" "Type B" "Type AB" "Type AB"
## [743] "Type A" "Type B" "Type AB" "Type A" "Type AB" "Type B" "Type B"
## [750] "Type A" "Type AB" "Type AB" "Type A" "Type B" "Type AB" "Type B"
## [757] "Type A" "Type B" "Type A" "Type A" "Type A" "Type AB" "Type AB"
## [764] "Type B" "Type B" "Type B" "Type A" "Type B" "Type A" "Type B"
## [771] "Type B" "Type A" "Type B" "Type AB" "Type B" "Type B" "Type B"
## [778] "Type A" "Type AB" "Type B" "Type AB" "Type AB" "Type B" "Type A"
## [785] "Type AB" "Type B" "Type B" "Type A" "Type A" "Type B" "Type B"
## [792] "Type B" "Type AB" "Type A" "Type B" "Type B" "Type A" "Type AB"
## [799] "Type A" "Type A" "Type AB" "Type A" "Type B" "Type AB" "Type B"
## [806] "Type AB" "Type A" "Type B" "Type B" "Type B" "Type A" "Type A"
## [813] "Type B" "Type A" "Type B" "Type B" "Type AB" "Type AB" "Type A"
## [820] "Type A" "Type A" "Type AB" "Type AB" "Type A" "Type A" "Type A"
## [827] "Type AB" "Type AB" "Type AB" "Type B" "Type B" "Type A" "Type AB"
## [834] "Type B" "Type A" "Type AB" "Type AB" "Type A" "Type A" "Type A"
## [841] "Type B" "Type A" "Type B" "Type A" "Type AB" "Type A" "Type AB"
## [848] "Type AB" "Type B" "Type A" "Type A" "Type B" "Type A" "Type B"
## [855] "Type A" "Type A" "Type B" "Type B" "Type A" "Type AB" "Type AB"
## [862] "Type A" "Type AB" "Type A" "Type B" "Type B" "Type AB" "Type A"
## [869] "Type A" "Type B" "Type AB" "Type A" "Type B" "Type B" "Type B"
## [876] "Type AB" "Type A" "Type B" "Type B" "Type B" "Type A" "Type B"

```

```
## [883] "Type AB" "Type AB" "Type B" "Type AB" "Type AB" "Type AB" "Type AB" "Type A"
## [890] "Type B" "Type A" "Type B" "Type AB" "Type A" "Type A" "Type A" "Type B"
## [897] "Type B" "Type A" "Type AB" "Type B" "Type B" "Type A" "Type A" "Type B"
## [904] "Type B" "Type B" "Type A" "Type B" "Type B" "Type A" "Type A" "Type AB"
## [911] "Type A" "Type B" "Type A" "Type A" "Type B" "Type B" "Type B" "Type B"
## [918] "Type AB" "Type A" "Type AB" "Type B" "Type A" "Type A" "Type A" "Type B"
## [925] "Type A" "Type A" "Type A" "Type B" "Type A" "Type A" "Type A" "Type B"
## [932] "Type A" "Type A" "Type AB" "Type B" "Type B" "Type B" "Type AB" "Type B"
## [939] "Type AB" "Type A" "Type A" "Type B" "Type AB" "Type A" "Type A" "Type B"
## [946] "Type B" "Type B" "Type A" "Type A" "Type A" "Type B" "Type B" "Type B"
## [953] "Type AB" "Type A" "Type A" "Type B" "Type A" "Type B" "Type B" "Type A"
## [960] "Type A" "Type A" "Type A" "Type AB" "Type A" "Type B" "Type B" "Type B"
## [967] "Type B" "Type B" "Type B" "Type B" "Type AB" "Type AB" "Type AB" "Type B"
## [974] "Type B" "Type AB" "Type A" "Type B" "Type AB" "Type B" "Type B" "Type B"
## [981] "Type B" "Type A" "Type AB" "Type A" "Type A" "Type A" "Type A" "Type AB"
## [988] "Type B" "Type B" "Type A" "Type B" "Type B" "Type B" "Type AB" "Type AB"
## [995] "Type B" "Type A" "Type B" "Type B" "Type B" "Type B" "Type B" "Type B"
```

2 A case study of DNA sequence

Ex 3.1

$\Omega = \{AA, AT, AC, AG, TA, TT, TC, TG, CA, CT, CC, CG, GA, GT, GC, GG\}$

Ex 3.2

$P(\text{ACG}) = (0.25 \times 0.25 \times 0.25) \times 3 = 0.047$

Ex 3.3

$\Omega = \{\emptyset, A, T, C, AA, AT, AC, AAA, \dots\}$

$P(\text{length } 10 \text{ G}^C) = (0.75)^{10} = 0.05$

Ex 3.4

1: $P(\text{CG}|\text{C}) = 0.5 \times 0.1 = 0.05$

2: $P(\text{CG}|\text{C}) = 0.3 \times 0.4 = 0.12$

2.1 Some simulations in R

Ex 4.1

```
ACGsim <- function(reps){
  count = 0

  for(i in 1:reps){
    sqce_5 <- paste(sample(c("A", "T", "G", "C"), replace = TRUE, size = 5), collapse = "")

    if(grepl("ACG", sqce_5, fixed = TRUE)){
      count <- count + 1
    }
  }
  return(count/reps)
}
ACGsim(10000)
```

```
## [1] 0.0481
```

Ex 4.2

```
Gsim <- function(reps, length = 10){
  sqces_with_G <- 0
  for(i in 1:reps){
    seq <- sample(c("A", "C", "T", "G"), replace = TRUE, size = length)
    if ("G" %in% seq){
      sqces_with_G <- sqces_with_G + 1
    }
  }
  sqces_with_no_G <- reps - sqces_with_G
  return(sqces_with_no_G/reps)
}

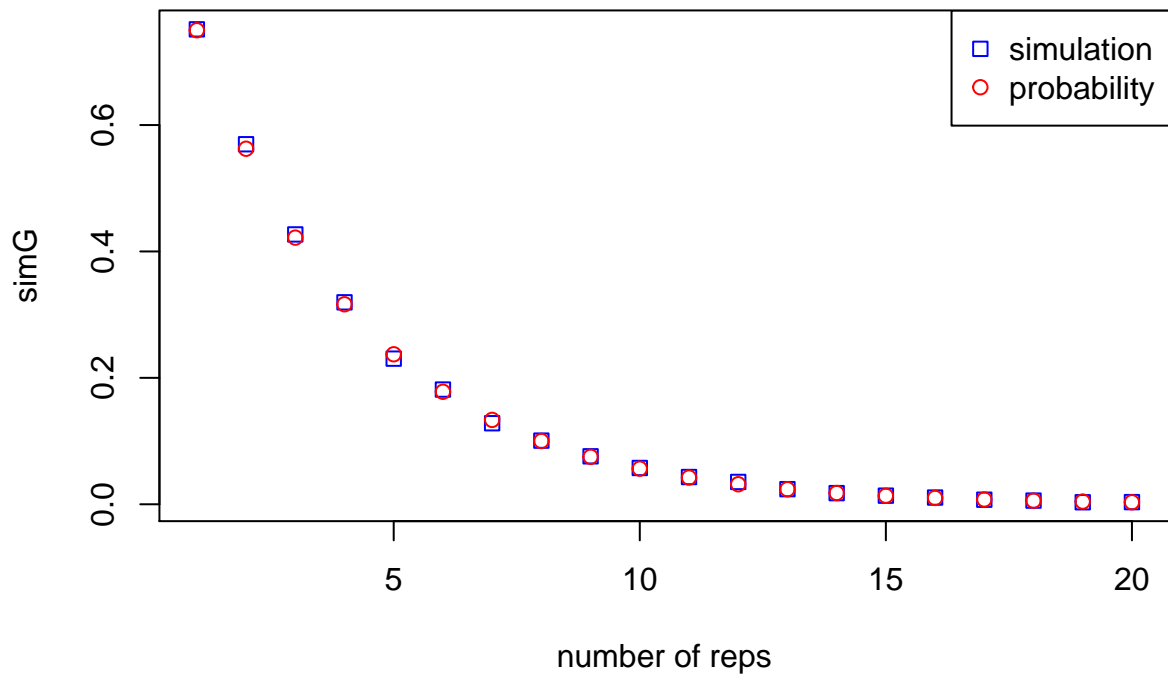
Gsim(10000)
```

```
## [1] 0.0567
```

Ex 4.3

```
simG <- c()
pG <- c()
for(i in 1:20){
  simG[i] <- Gsim(10000, length = i)
  pG[i] <- dbinom(i, size = i, prob = 0.75)
}

plot(1:20, simG, col = "blue", pch = 0, xlab = "number of reps")
points(1:20, pG, col = "red", pch = 1)
legend("topright", legend = c("simulation", "probability"),
      col = c("blue", "red"), pch = c(0, 1))
```



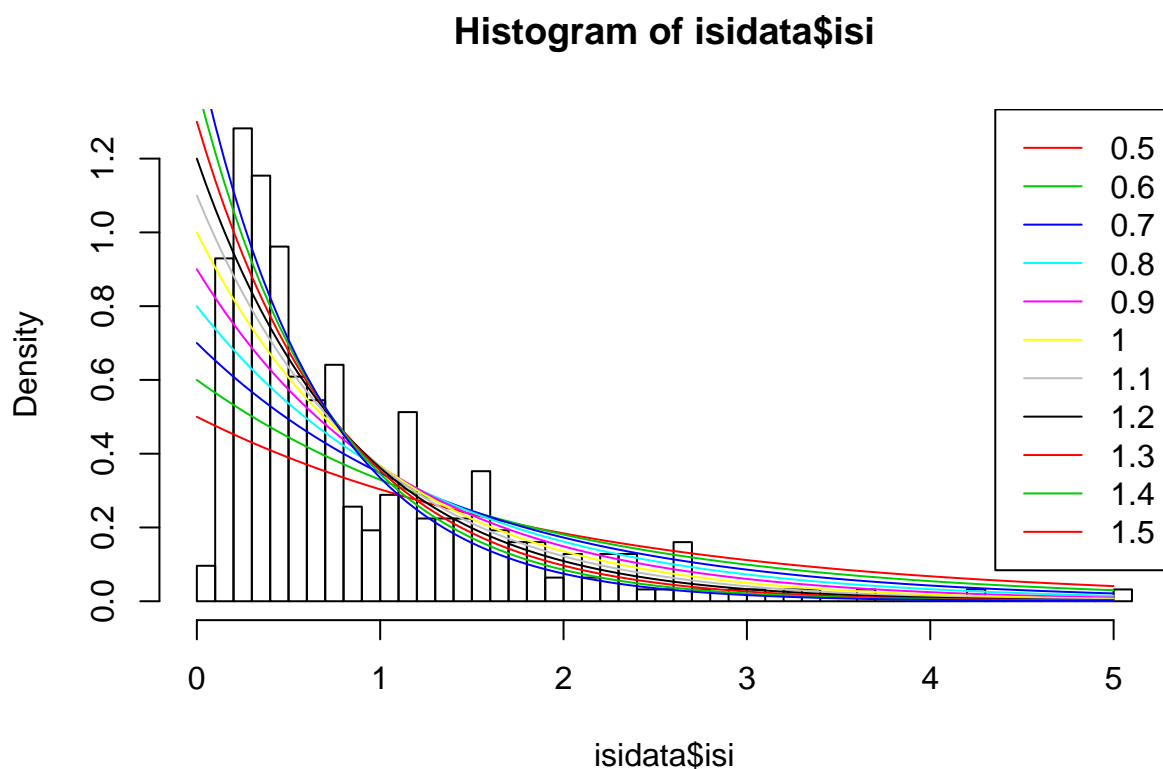
3 A case study of neuronal data

Ex 5.1 and 5.2

```
isidata <- read.table("neuronspikes.txt", col.names = "isi")
hist(isidata$isi, breaks = 50, probability = TRUE)

for(i in seq(0.5, 1.5, 0.1)){
  curve(dexp((x), rate = i), from = 0, to = 5, col= (i*10)-3, add = TRUE)
}

legend("topright", legend=c(seq(0.5,1.5,0.1)),
      col=c(2:length(seq(0.5,1.5,0.1))), lty = 1)
```



A λ value of 1.5 fits the data the best.

Ex 5.3

```
pexp(1, rate = 1.5)
```

```
## [1] 0.7768698
```

```
pexp(1.5, rate = 1.5) - pexp(0.5, rate = 1.5)
```

```
## [1] 0.3669673
```

4 Brain cell database

Ex 6.1

The option `na.strings` converts empty strings to NA values.

Ex 6.2

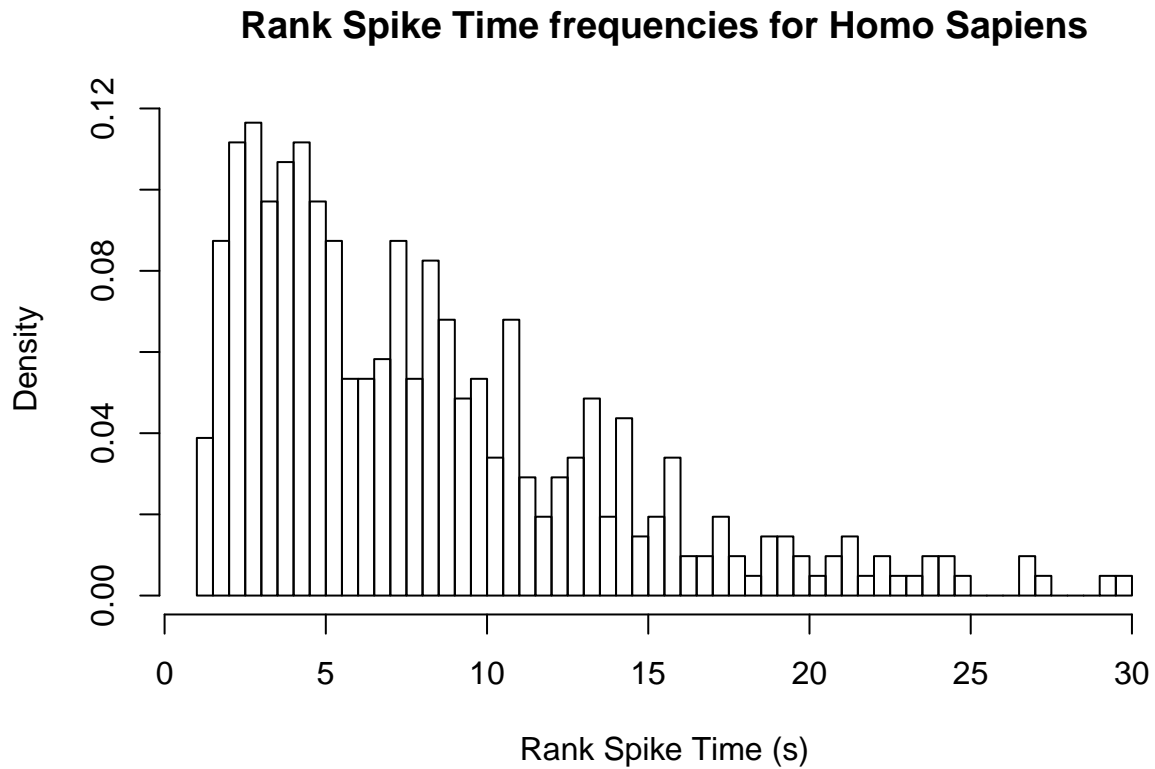
The donor species column indicates which values are humans and which are mice.

```
sum(cells$donor__species == "Homo Sapiens") / length(cells[,1])
```

```
## [1] 0.1770253
```

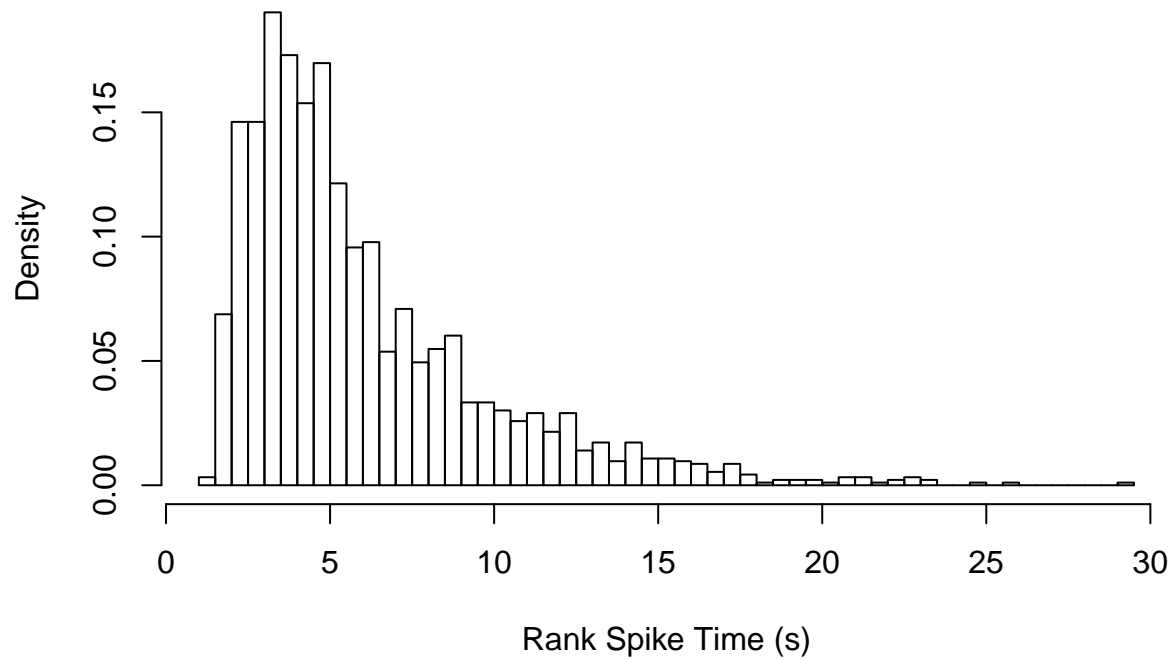

Ex 6.3

```
hist(cells$ef_peak_t_ramp[cells$donor_species == "Homo Sapiens"],  
      proba = TRUE, xlab = "Rank Spike Time (s)",  
      breaks = 50, main = "Rank Spike Time frequencies for Homo Sapiens")
```



```
hist(cells$ef_peak_t_ramp[cells$donor_species == "Mus musculus"],  
      proba = TRUE, xlab = "Rank Spike Time (s)",  
      breaks = 50, main = "Rank Spike Time frequencies for Mus musculus")
```

Rank Spike Time frequencies for *Mus musculus*

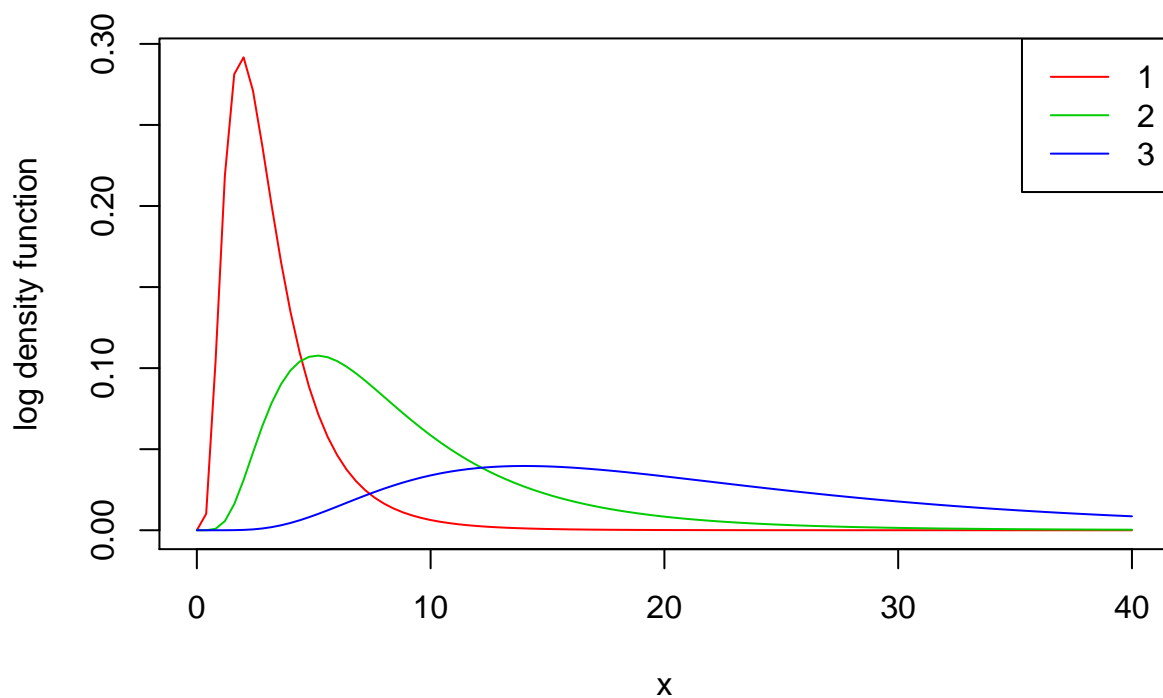


Ex 6.4

```
first_plot = FALSE

for(i in 1:3){
  curve(dlnorm(x, sdlog = 0.6, meanlog = i), from = 0, to = 40,
        add = first_plot, col = i+1, ylab = "log density function")
  first_plot = TRUE
}

legend("topright", legend = 1:3, col = 2:4, lty = 1)
```



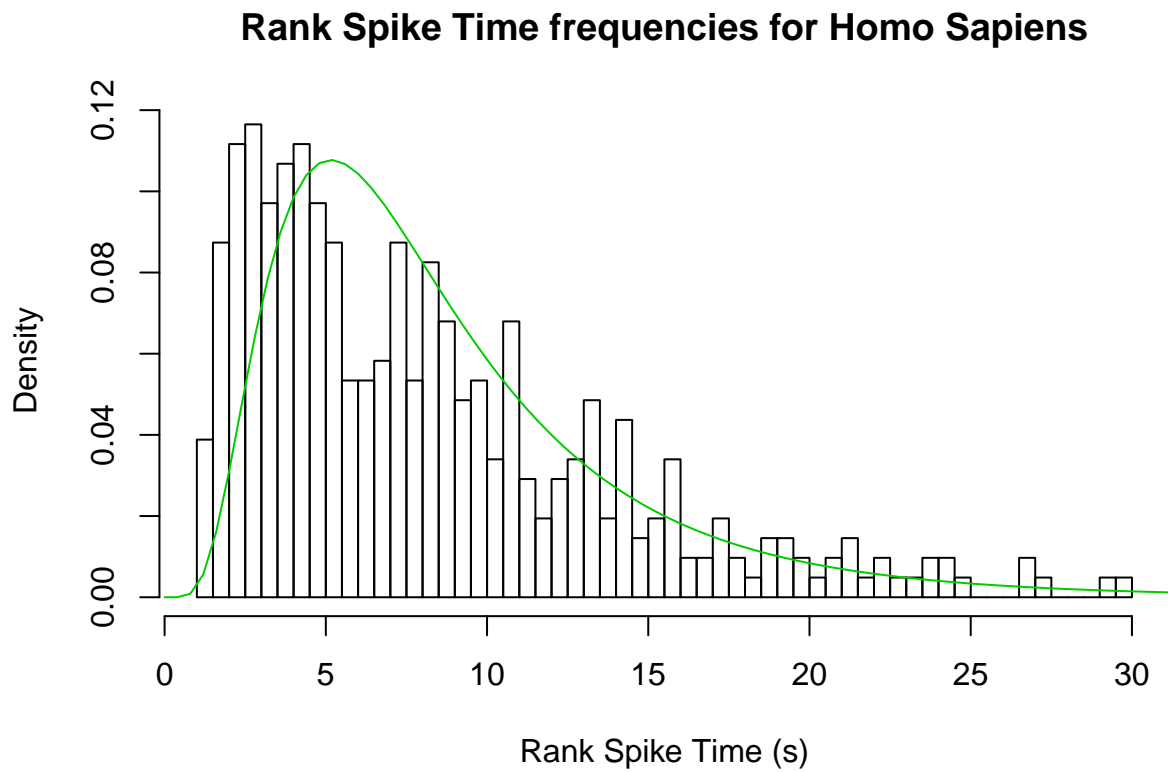
Ex 6.5

meanlog = 2 is best, based on the fit of the corresponding density function in the plot.

Ex 6.6

```
hist(cells$ef_peak_t_ramp[cells$donor_species == "Homo Sapiens"],
     proba = TRUE, xlab = "Rank Spike Time (s)",
     breaks = 50, main = "Rank Spike Time frequencies for Homo Sapiens")

curve(dlnorm(x, sdlog = 0.6, meanlog = 2),
     from = 0, to = 40, add = TRUE,
     col = 3, ylab = "log density function")
```



Yes, the density fits the data.

Ex 6.7

```
sum(cells$donor__species == "Homo Sapiens" & cells$donor__sex == "Male")
```

```
## [1] 234
```

```
sum(cells$donor__species == "Homo Sapiens" & cells$donor__sex == "Female")
```

```
## [1] 179
```