

# LearnComputationalModeling

Krisma Adiwibawa

## Chapter 2: From Words to Models

### 2.2.2 The Random Walk Model

Listing 2.1 A simple random-walk model

```
#random-walk model
nreps <- 10000 #number of random-walks (decisions)
nsamples <- 2000 #number of times that evidence is being sampled for each decision

drift <- 0.0 #noninformative stimulus
           #drift rate: amount of evidence available during sampling
sdrw <- 0.3 #noise in the evidence as standard deviation
criterion <- 3 #response criterion (the distance of the 2 boundaries from origin)

latencies <- rep(0, nreps) #vector containing zeros
responses <- rep(0, nreps) #vector containing zeros
evidence <- matrix(0, nreps, nsamples+1) #a matrix containing zeros
for (i in c(1:nreps)) {
  evidence[i,] <- cumsum(c(0, rnorm(nsamples, drift, sdrw))) #evidence accumulation
  p <- which(abs(evidence[i,])>criterion)[1] #the step at which decision is made
  ↪ (the column)
  responses[i] <- sign(evidence[i,p]) #return +1 or -1
  latencies[i] <- p
}

#View(evidence)
```

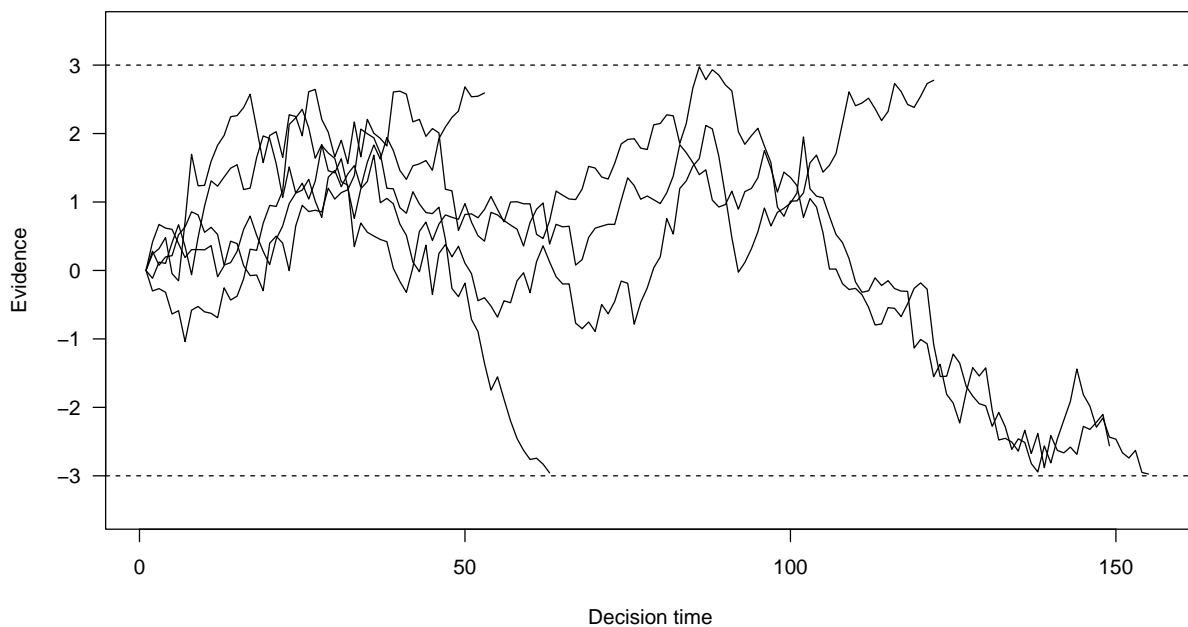
Listing 2.2 Plot up to 5 random-walk paths

```
#plot up to 5 random-walk paths
tbpn <- min(nreps, 5) #to-be-plotted-number
plot(1:max(latencies[1:tbpn])+10, # create x-axis as needed (what the longest
  ↪ latency?
```

```

type="n", #create empty plot
las=1, #make the y-axis labels horizontal
ylim=c(-criterion-.5, criterion+.5), #set y-axis limit, give extra room above
  ↪ and below criterion
ylab="Evidence", xlab="Decision time")
for (i in c(1:tbpn)) {
  lines(evidence[i, 1:(latencies[i]-1)]) #this is the values in row i, which
  ↪ is the evidence sampling. -1 means plot the line just before the
  ↪ decision
}
abline(h=c(criterion, -criterion), lty="dashed") #create a dashed line for
  ↪ criterion (upper & lower decision threshold)

```



Listing 2.3 Plot distribution of response latencies from random-walk

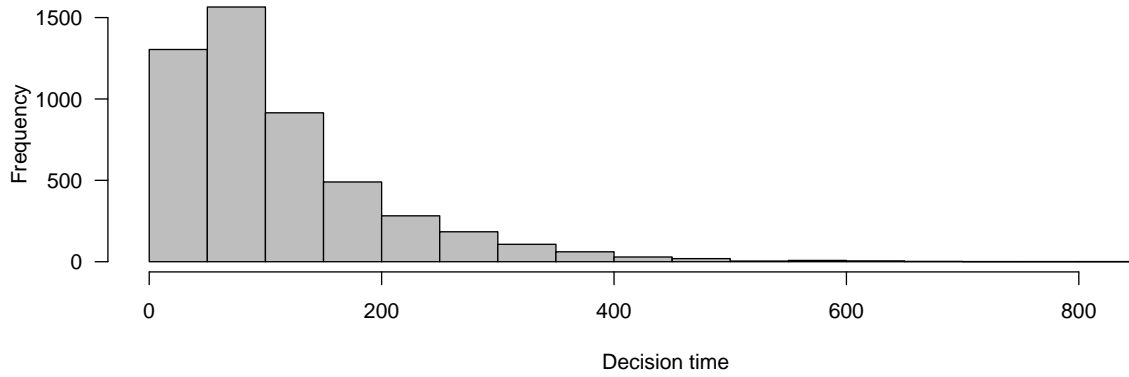
```

#plot histogram latencies
par(mfrow=c(2,1)) #split the plotting area into 2 rows and 1 column
toprt <- latencies[responses>0] #latencies that corresponds to +1
topprop <- length(toprt)/nreps #proportion of top responses relative to all trials
hist(toprt, col="gray",
  xlab="Decision time", xlim=c(0, max(latencies)),
  main=paste("Top responses (", as.numeric(topprop),
    ") m=", as.character(signif(mean(toprt),4)),
    sep=""), las=1)
botrt <- latencies[responses<0] #latencies that corresponds to -1
botprop <- length(botrt)/nreps

```

```
hist(botrt, col="gray",
     xlab="Decision time", xlim=c(0, max(latencies)),
     main=paste("Bottom responses (", as.numeric(botprop),
               ") m=", as.character(signif(mean(botrt), 4))
     )
)
```

**Top responses (0.4978) m=112.7**



**Bottom responses ( 0.5022 ) m= 112.7**

