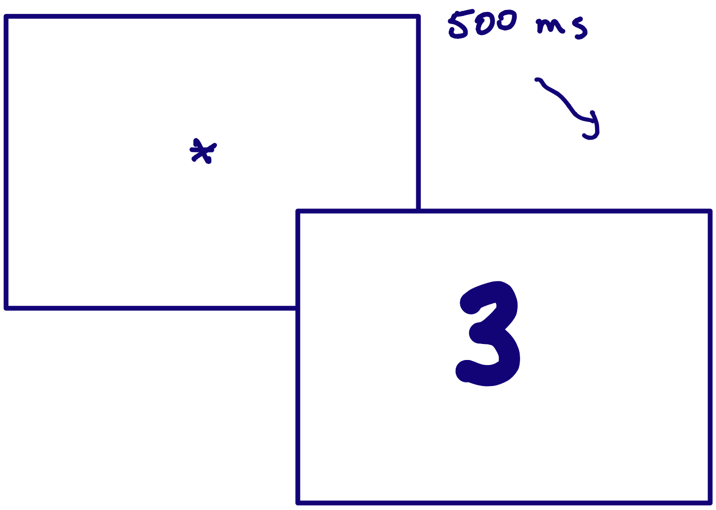
# Lecture 6 - Characteristics of Response Time Distributions

Let's look at a typical experiment designed to investigate how people make decisions with numbers:




Press "a" if less than 5
Press "l" if greater than 5

Question: what happens to RTs as the presented number gets closer to the comparison standard 5?

Partial answer: mean RT increases

↳ numerical distance effect (Moyer & Landauer, 1967)

↳ but what happens to the distribution of RTs?

Goal: let's learn how to model RT distributions!

# Sequential sampling models

↳ "accumulator model"

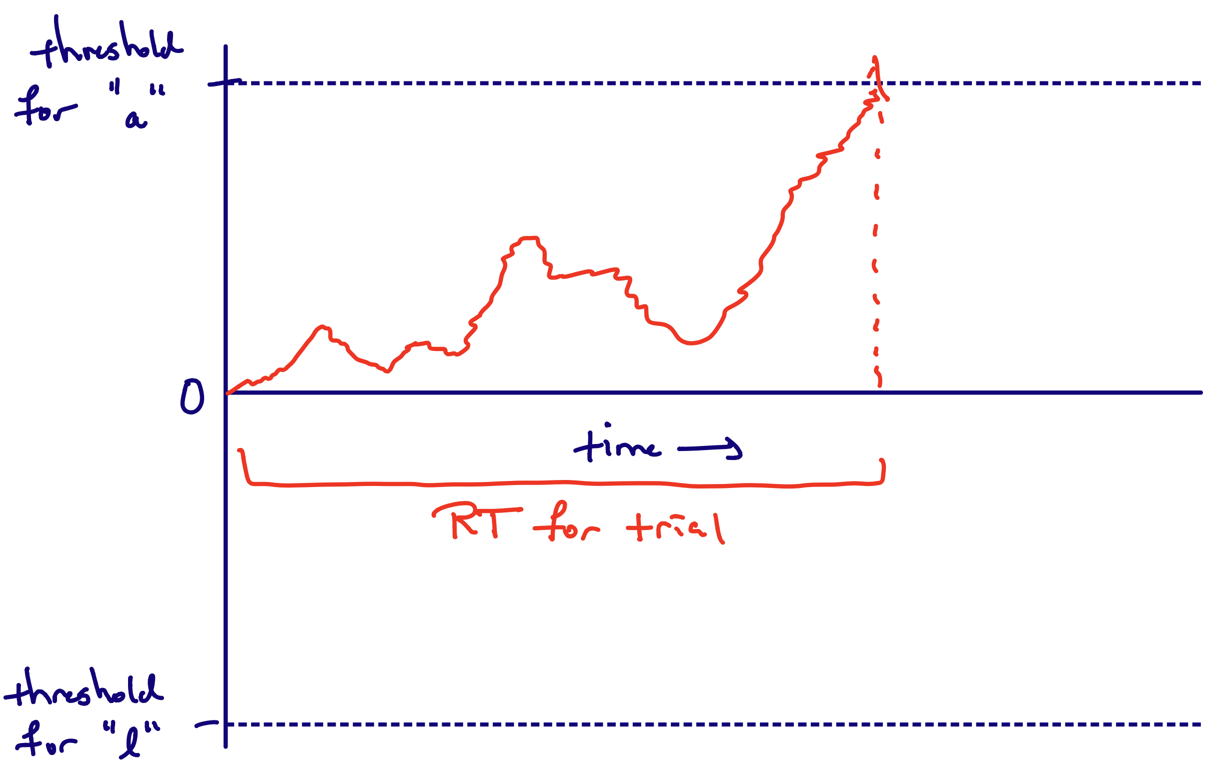
↳ "random walk model"

Basic idea:

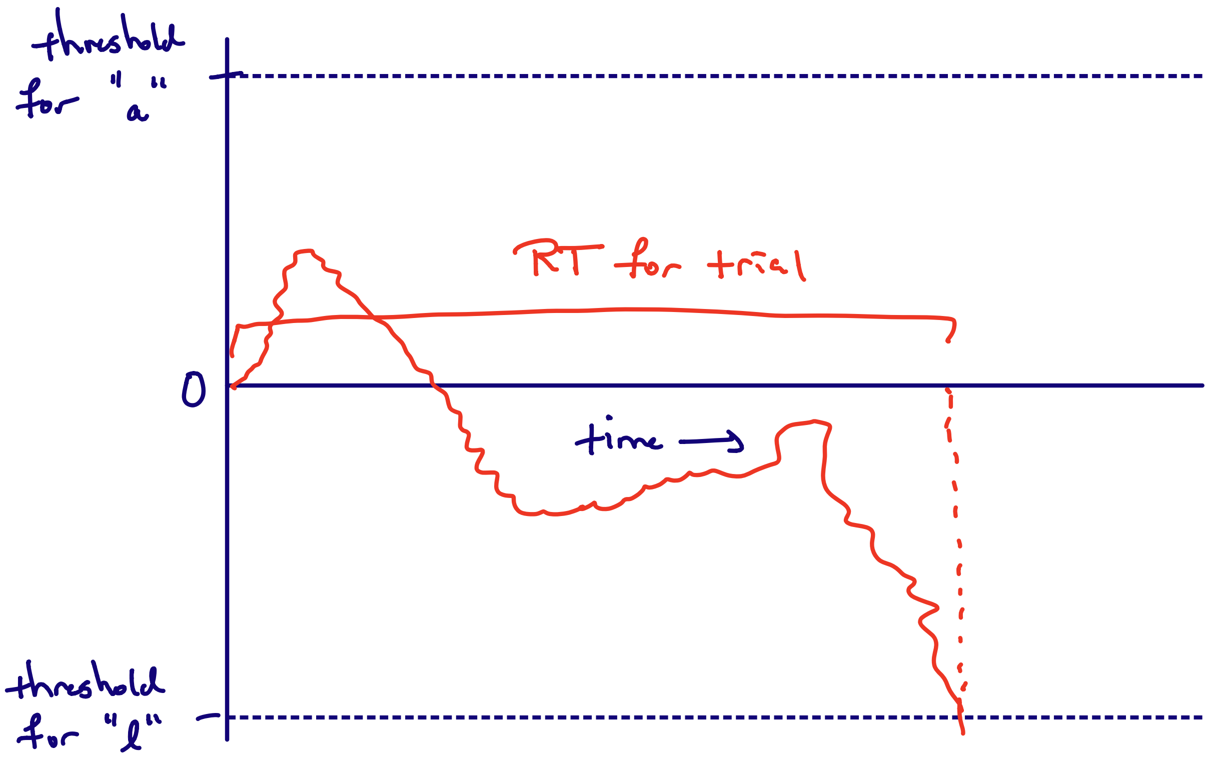
* when stimulus is presented, we quickly and repeatedly sample information from the stimulus

* at each discrete time step, we accumulate evidence
  ↳ this information is noisy – sometimes it moves us closer to a decision, sometimes farther away

* once we reach a certain threshold, the accumulated information is sufficient to trigger a decision



Sometimes we make errors and make the wrong choice!




At its simplest, this is a two parameter model:

Parameter 1: response threshold

↳ amount of evidence required to trigger a decision

Parameter 2: drift rate

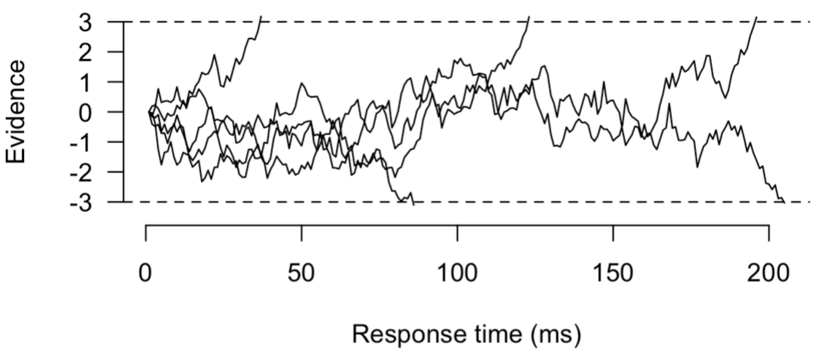
↳ average rate of evidence accumulation

↳ drift rate = 0 → noninformative stimulus

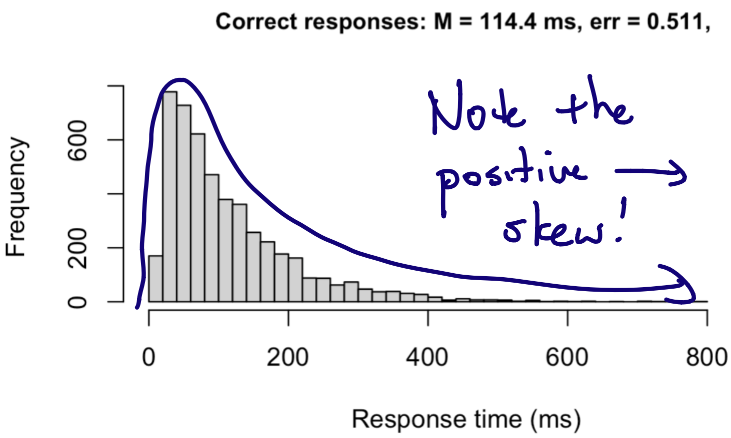
drift rate > 0 → stimulus is informative

# Implementation in R

```
# build a random walk accumulator model
reps = 10000
samples = 5000 # think of these as milliseconds (1 sample per millisecond)

# parameters of model
driftRate = 0.0
threshold = 3

# build empty structures for storing RTs, responses, and accumulated evidence
RTs = numeric(reps)
responses = numeric(reps)
evidence = matrix(0, reps, samples + 1)

# run the simulation
for (i in 1:reps){
  evidence[i, ] = cumsum(c(0, rnorm(n = samples, mean = driftRate, sd = 0.3)))
  p = which(abs(evidence[i,]) > threshold)[1]
  responses[i] = sign(evidence[i,p])
  RTs[i] = p
}
```

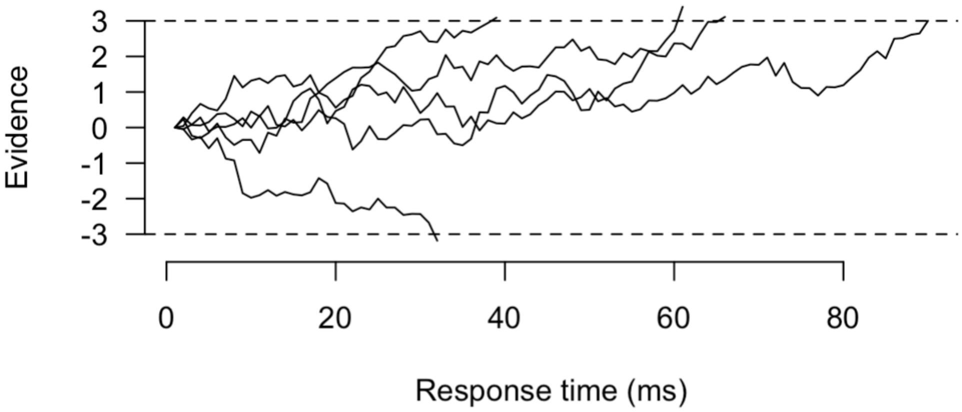



```
# plot some of the random walks and a histog
par(mfrow = c(2,1), cex.main = 0.9)

howmany = 5
plot(1:max(RTs[1:howmany])+10,
     type = "n",
     las = 1, bty="n",
     ylim = c(-threshold-0.5, threshold+0.5
     xlab = "Response time (ms)",
     ylab = "Evidence"
     )
for (i in 1:howmany){
  lines(evidence[i, 1:(RTs[i])])
}
abline(h = c(threshold, -threshold), lty=2)


# plot histograms of RTs
topRT = RTs[responses > 0]
topErr = 1- length(topRT)/reps
hist(topRT, breaks=30,
     xlim = c(0, max(RTs)),
     xlab = "Response time (ms)",
     main = sprintf("Correct responses: M = %.1f ms, err = %.3f, ", mean(topRT), topErr)
     )
```



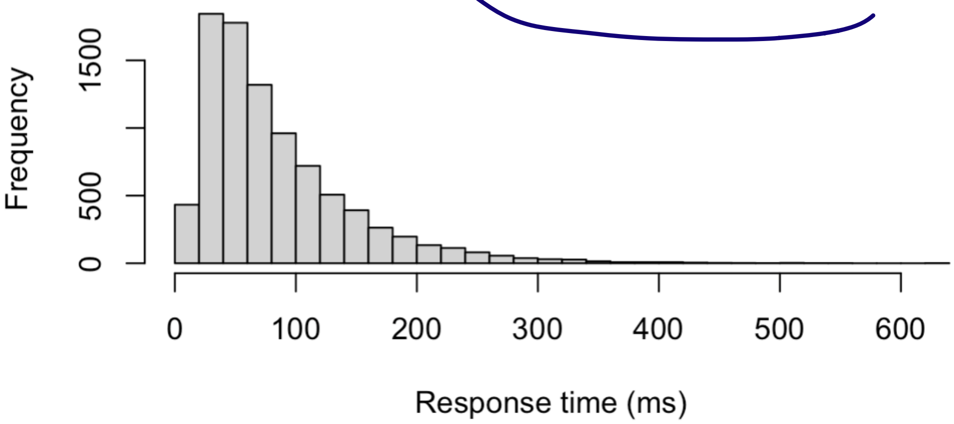

Note: if drift rate = 0, error rate ≈ 50%

What happens if we increase drift rate?
↳ try driftrate = 0.03




Correct responses: M = 84.8 ms, err = 0.105,

decreased error rate and faster RTs.




In today's homework, we will systematically investigate how threshold and drift rate affect the mean RT and error rates.