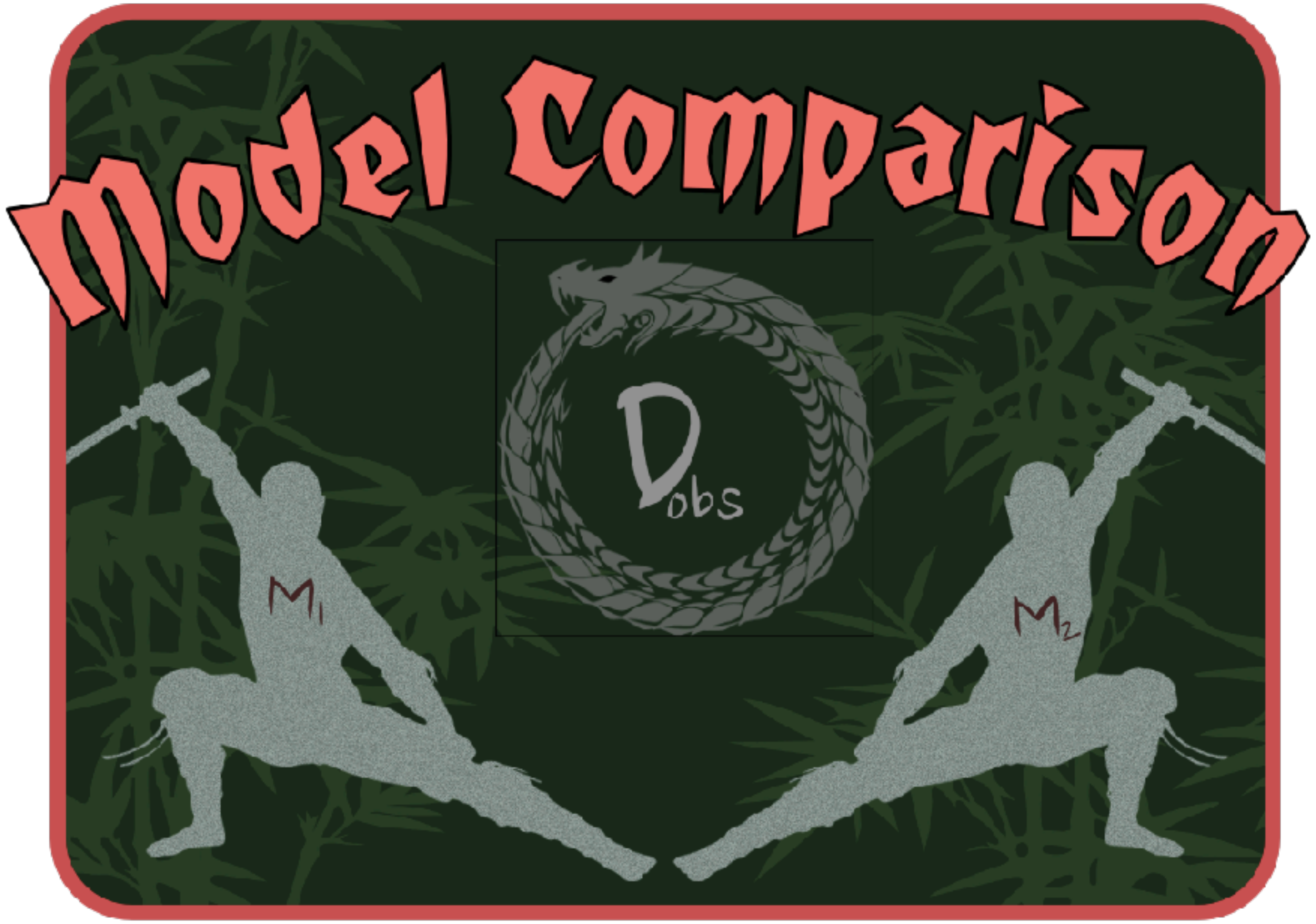


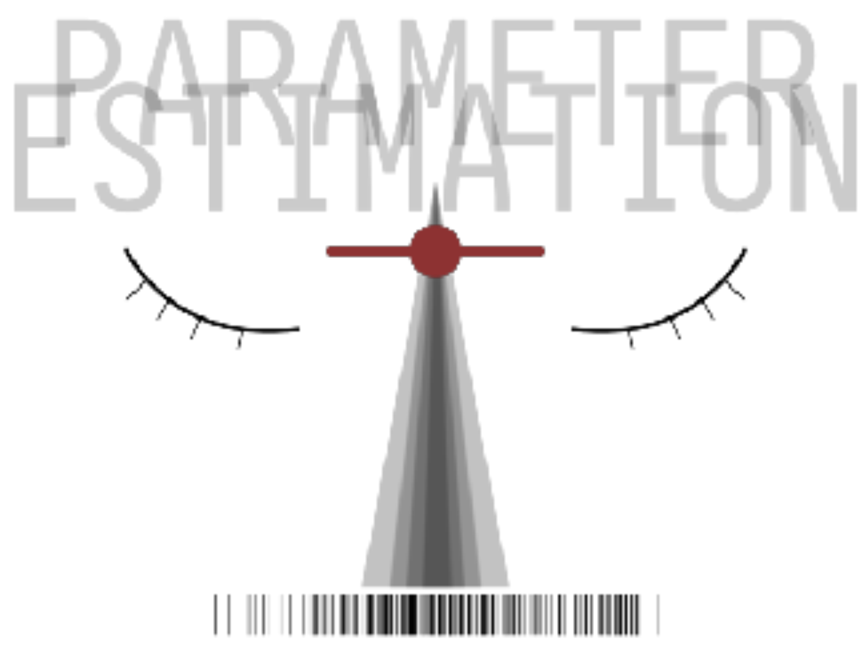
INTRODUCTION TO DATA ANALYSIS

MODEL COMPARISON

RECAP & OUTLOOK

PARAMETER ESTIMATION

- given model M and data D: what are good values of parameters?

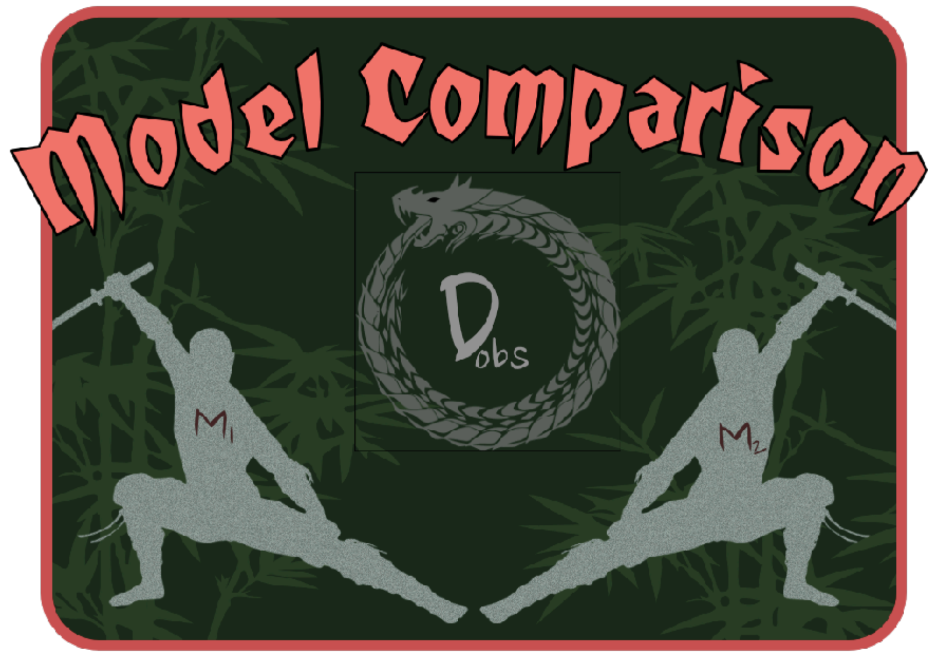


HYPOTHESIS TESTING

- given model M: is a specific null assumption about some parameter's value compatible with data D?

MODEL COMPARISON

- how much better an explanation of data D is one model, relative to another model?



WHAT MAKES A MODEL 'GOOD'?

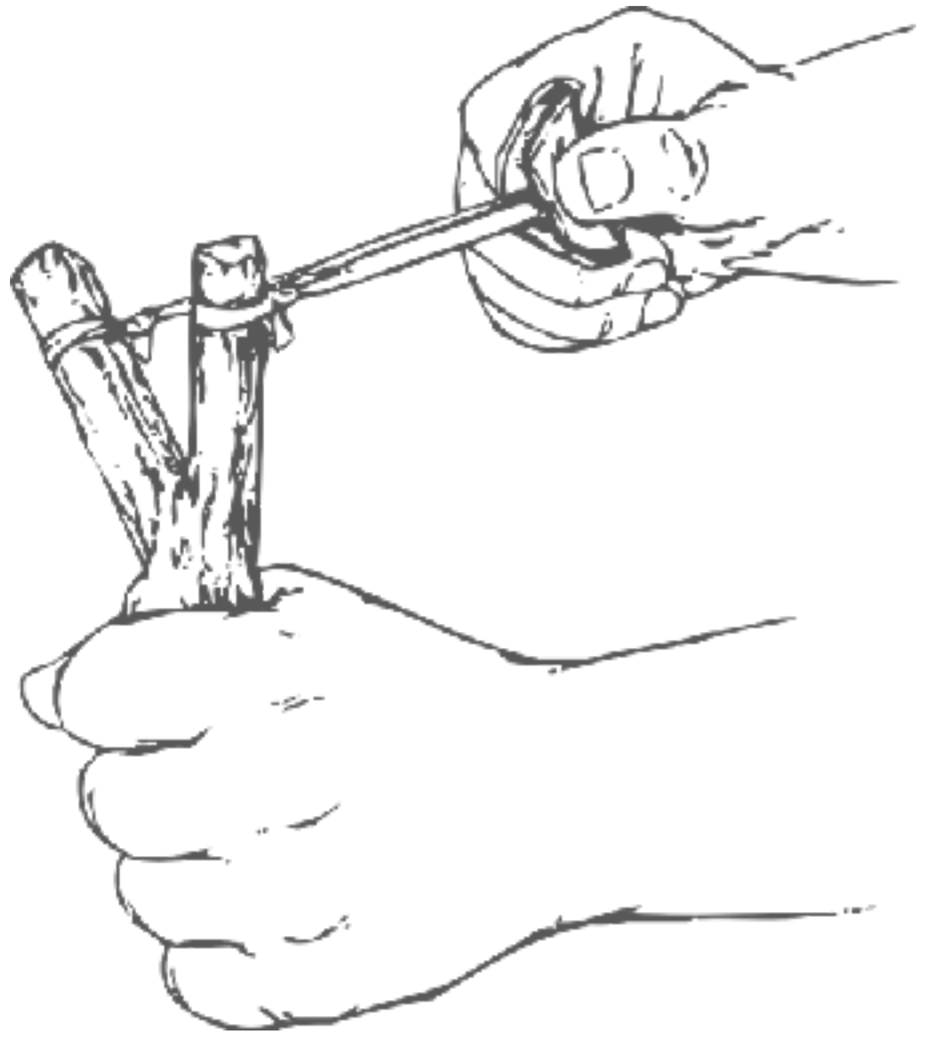
GOOD EXPLANATION

- model M is a good model of data D to the extent that it explains D well
- a good explanation of D is a view of the world that makes D less puzzling
  - the higher $P(D \mid M)$, the better M explains D

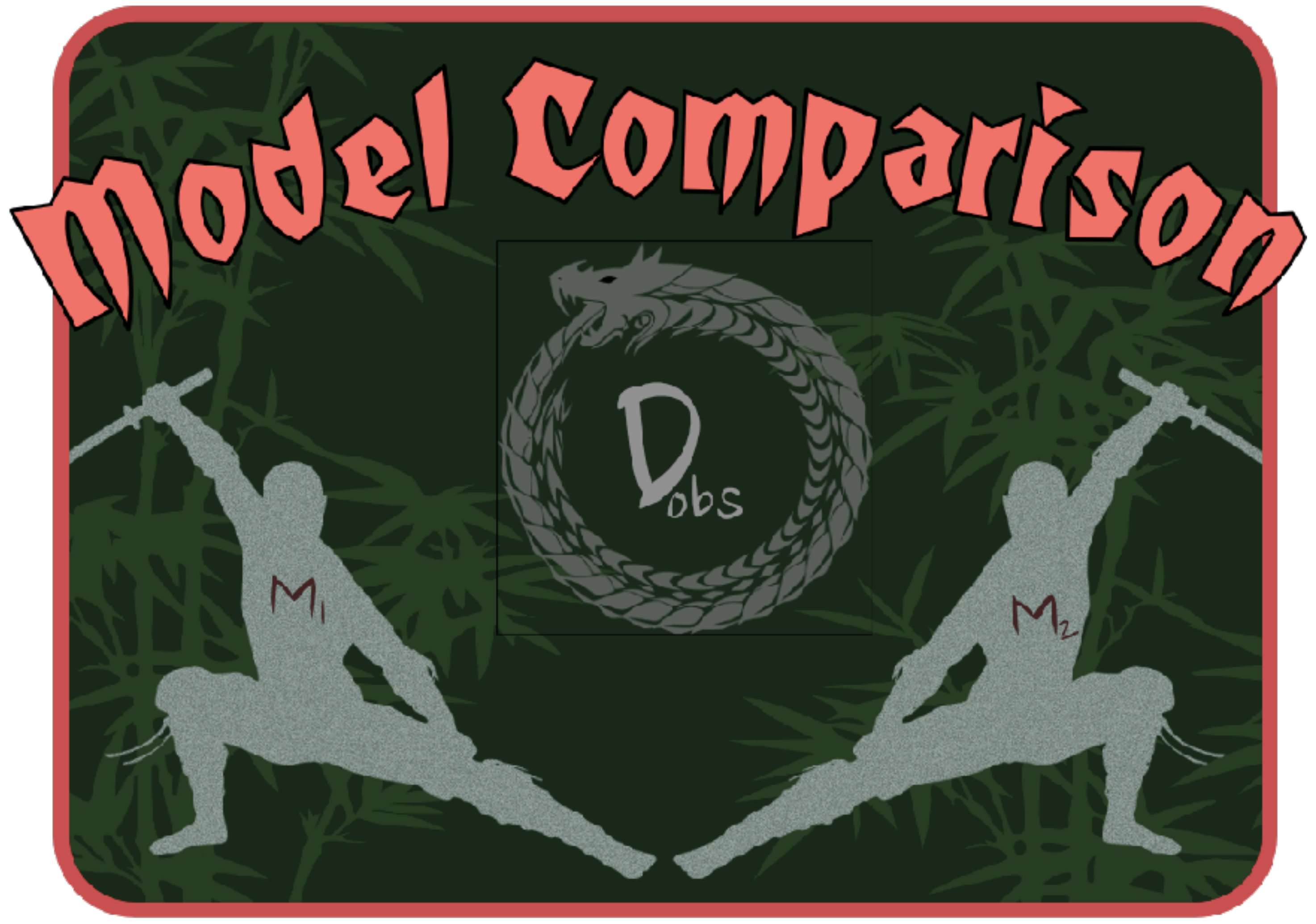
SIMPLICITY :: ECONOMY :: PARSIMONY

- model M is a good model of data D to the extent that it is simple
- we want our explanations to be austere, with few postulates, no magic ingredients and a lean mechanism / functional form
  - the fewer (powerful) parameters M has, the better

LEARNING GOALS



- understand the differences between estimation, testing and model comparison
- understand the idea behind and become able to apply the covered methods:
  - Akaike information criterion
  - likelihood-ratio test
  - Bayes factor
- become familiar with pro's and con's of each of these methods

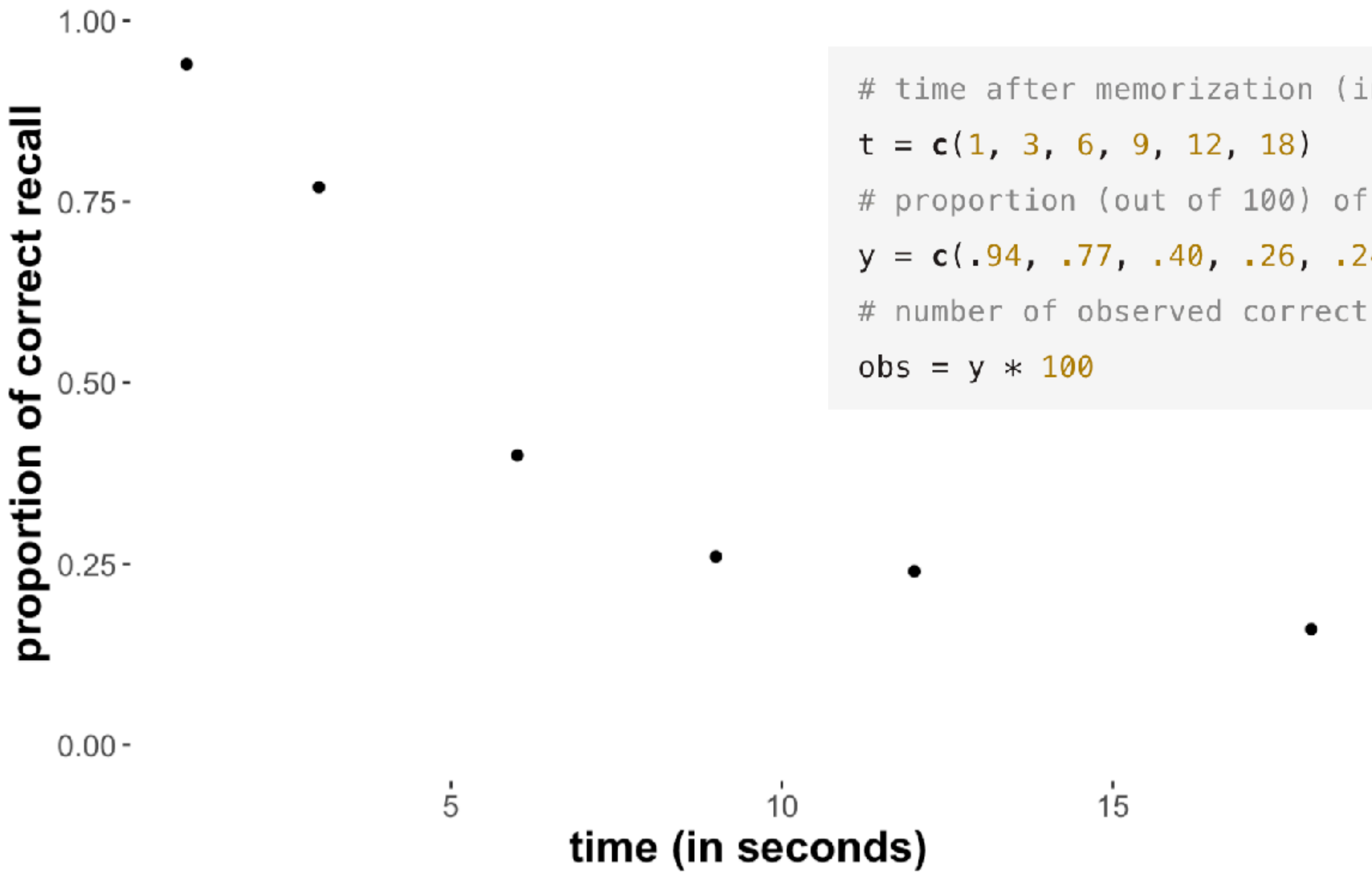


case study

time-course of recall rates

FORGETTING DATA

- 100 binary measurements (correct / incorrect recall) at different times after memorization



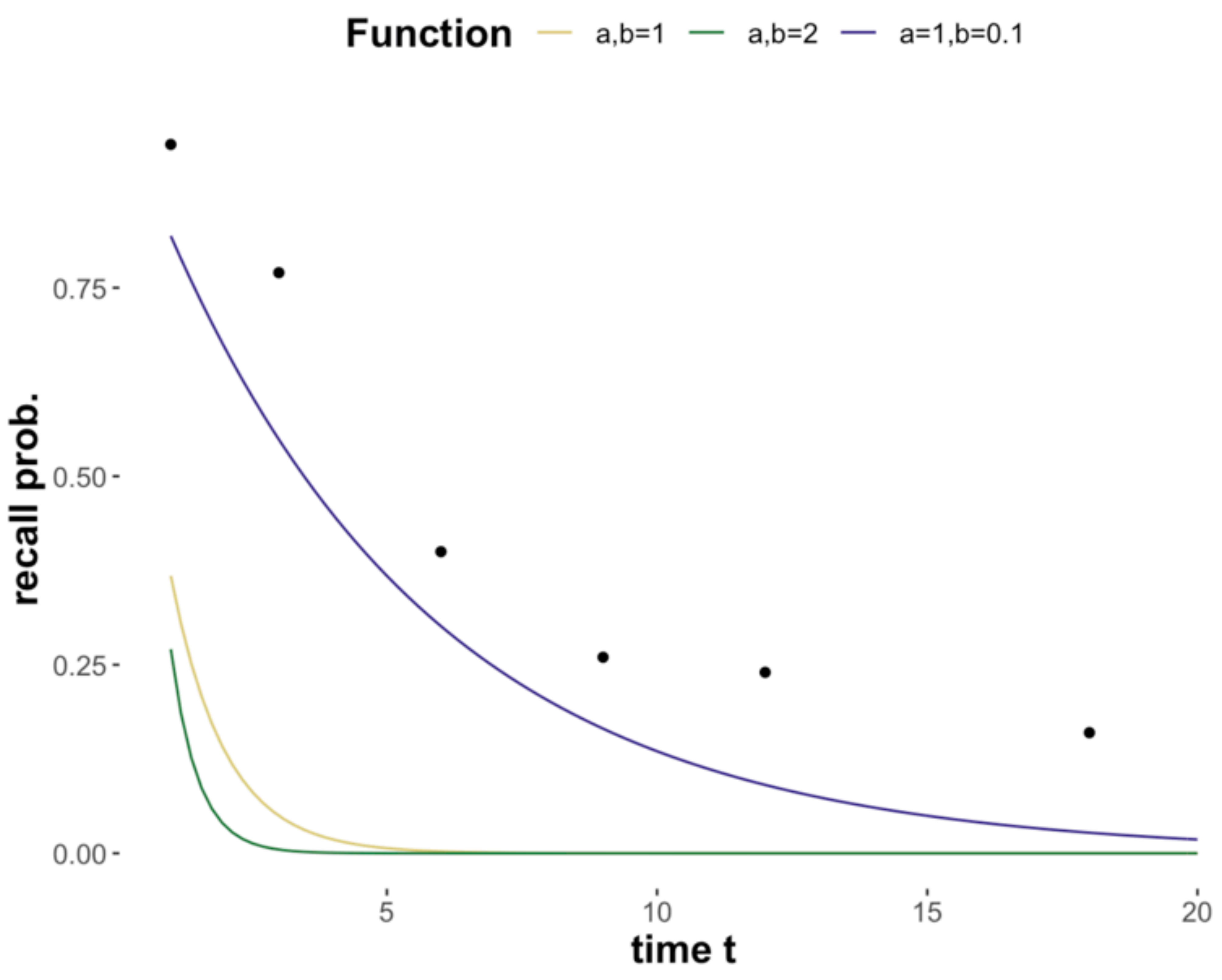

```
# time after memorization (in seconds)
t = c(1, 3, 6, 9, 12, 18)
# proportion (out of 100) of correct recall
y = c(.94, .77, .40, .26, .24, .16)
# number of observed correct recalls (out of 100)
obs = y * 100
```

RECALL MODELS

EXPONENTIAL MODEL

$$P(D = \langle k, N \rangle \mid \langle a, b \rangle) = \text{Binom}(k, N, \underline{a \exp(-bt)})$$

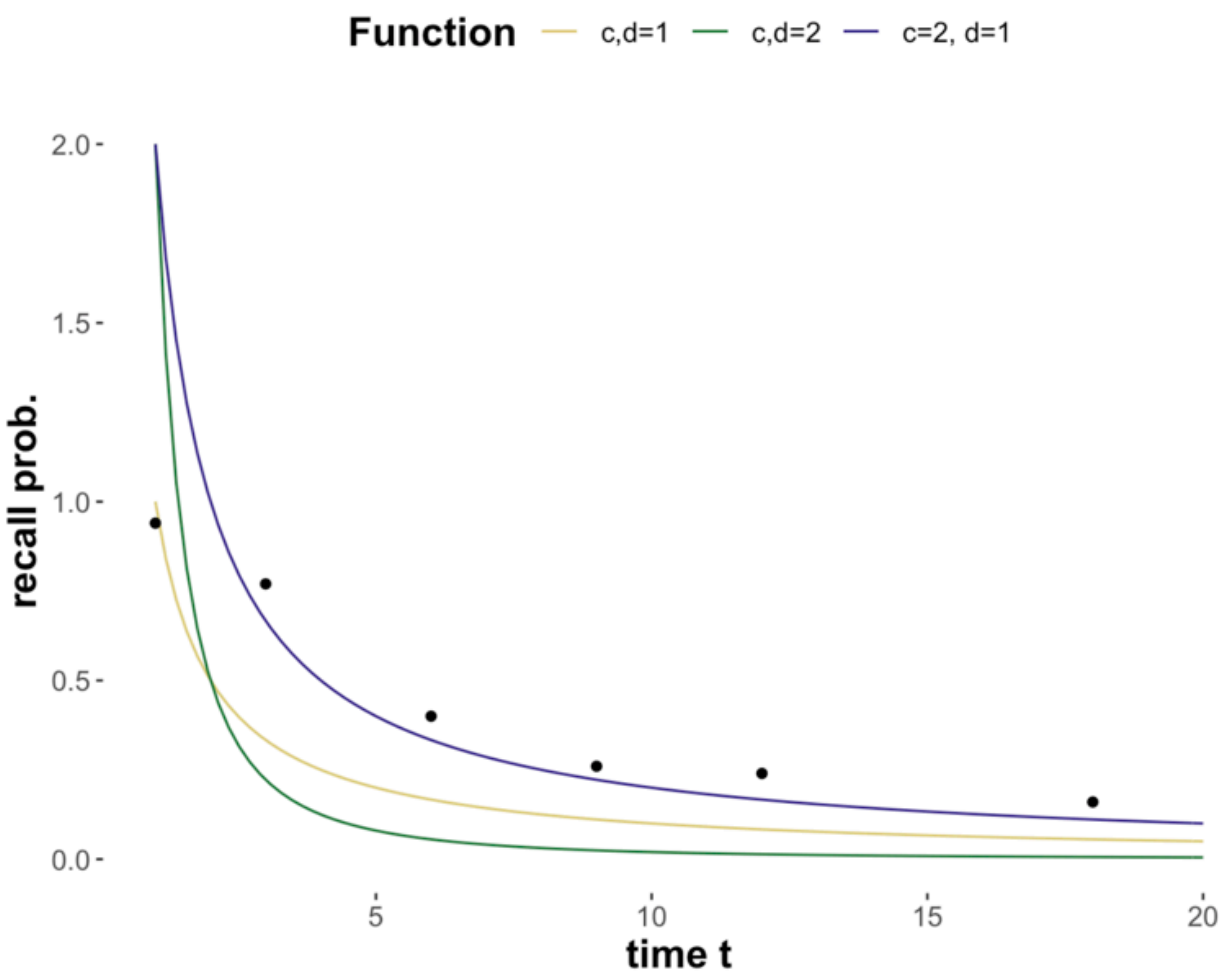
with $a, b > 0$



POWER MODEL

$$P(D = \langle k, N \rangle \mid \langle c, d \rangle) = \text{Binom}(k, N, \underline{c\ t^{-d}})$$

with $c, d > 0$

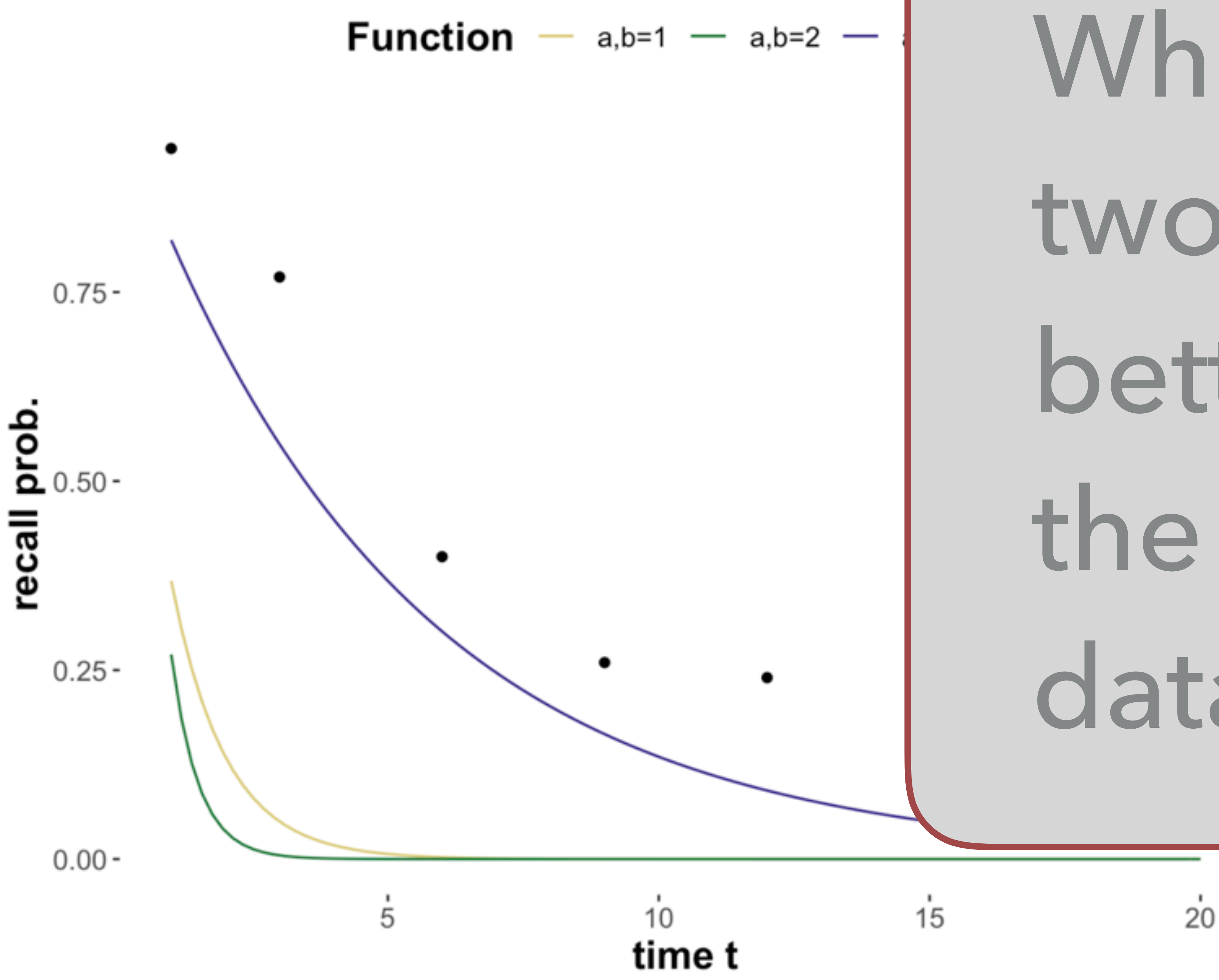


RECALL MODELS

EXPONENTIAL MODEL

$$P(D = \langle k, N \rangle \mid \langle a, b \rangle) = \text{Binom}(k, N, \underline{a \exp(-bt)})$$

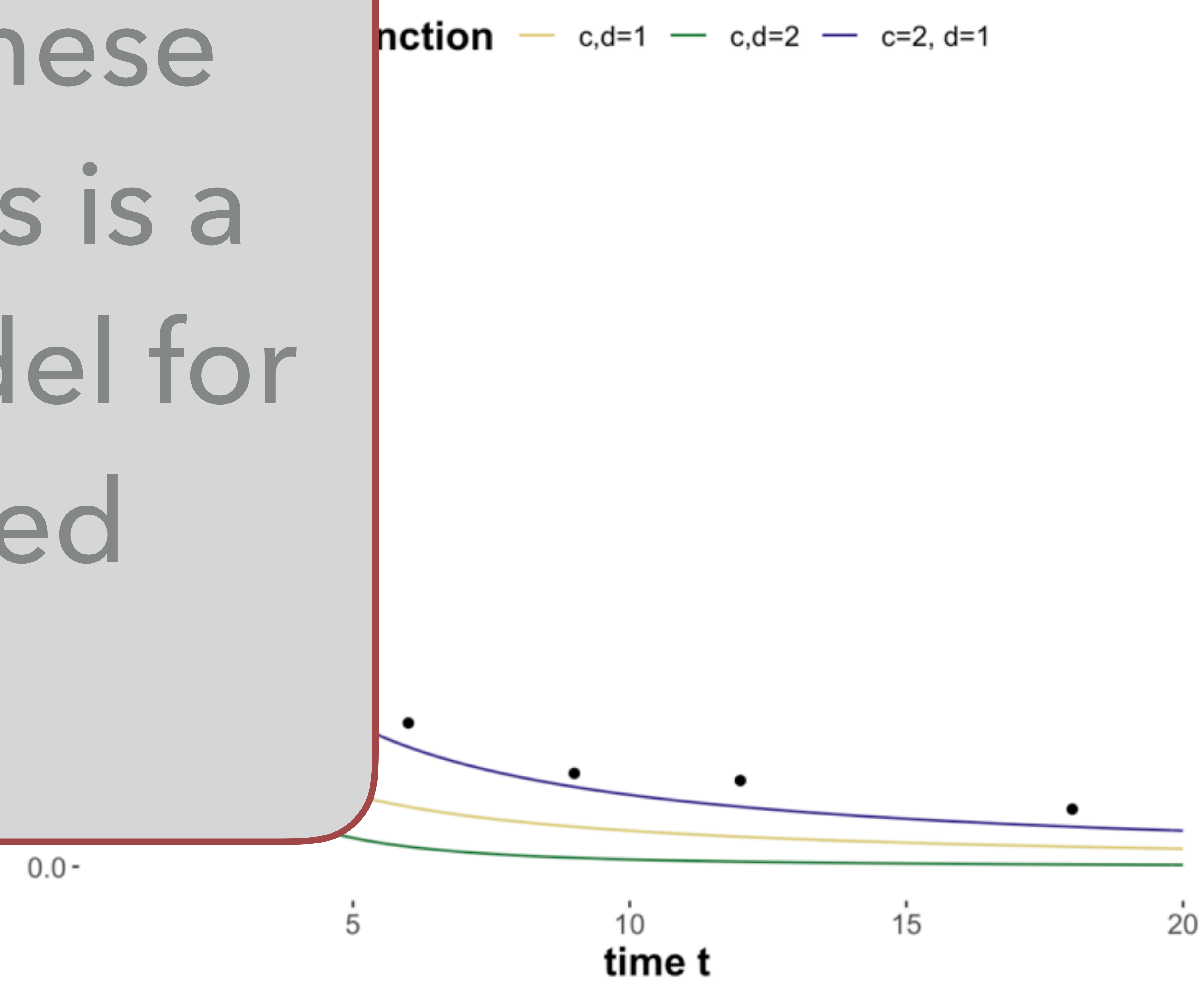
with $a, b > 0$



POWER MODEL

$$P(D = \langle k, N \rangle \mid \langle c, d \rangle) = \text{Binom}(k, N, \underline{c\ t^{-d}})$$

with $c, d > 0$



Which of these two models is a better model for the observed data?

Akaike information criterion

AKAIKE INFORMATION CRITERION

- M_i is a (frequentist) model with likelihood function $P(D \mid \theta_i, M_i)$
- k free parameters in parameter vector θ_i
- $\hat{\theta}_i = \arg\max_{\theta_i} P(D_{\text{obs}} \mid \theta_i, M_i)$ is the MLE for observed data D_{obs}
- the AIC-score (where lower is better) is defined as:

$$\text{AIC}(M_i, D_{\text{obs}}) = 2k - 2 \log P(D_{\text{obs}} \mid \hat{\theta}_i, M_i)$$

[penalty for complexity]

[how surprising is the data for the best parameter of the model?]

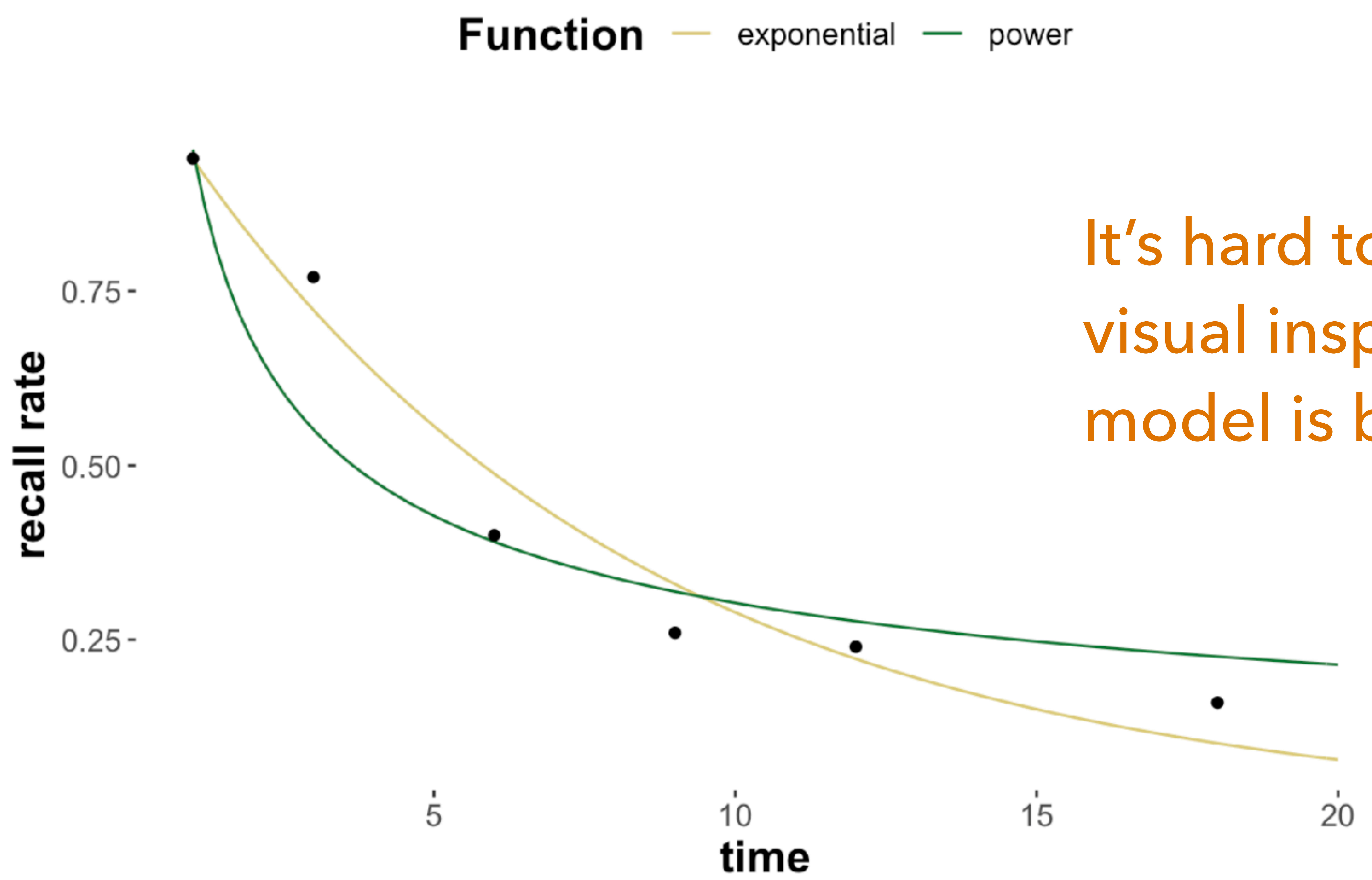
COMPUTING AIC-SCORES :: STEP 1 :: GETTING MLEs

```r
# generic neg-log-LH function (covers both models)
nLL_generic <- function(par, model_name) {
  w1 <- par[1]
  w2 <- par[2]
  # make sure paramters are in acceptable range
  if (w1 < 0 | w2 < 0 | w1 > 20 | w2 > 20) {
    return(NA)
  }
  # calculate predicted recall rates for given parameters
  if (model_name == "exponential") {
    theta <- w1*exp(-w2*t)  # exponential model
  } else {
    theta <- w1*t^(-w2)     # power model
  }
  # avoid edge cases of infinite log-likelihood
  theta[theta <= 0.0] <- 1.0e-4
  theta[theta >= 1.0] <- 1-1.0e-4
  # return negative log-likelihood of data
  - sum(dbinom(x = obs, prob = theta, size = 100, log = T))
}
# negative log likelihood of exponential model
nLL_exp <- function(par) {nLL_generic(par, "exponential")}
# negative log likelihood of power model
nLL_pow <- function(par) {nLL_generic(par, "power")}
```

```r
# getting the best fitting values
bestExpo <- optim(nLL_exp, par = c(1,0.5))
bestPow  <- optim(nLL_pow, par = c(0.5,0.2))
MLEstimates = data.frame(model = rep(c("exponential", "power"), each = 2),
                         parameter = c("a", "b", "c", "d"),
                         value = c(bestExpo$par, bestPow$par))
knitr::kable(MLEstimates)
```

model	parameter	value
exponential	a	1.0701722
exponential	b	0.1308151
power	c	0.9531330
power	d	0.4979154

INSPECTING EACH MODEL'S BEST PREDICTION



It's hard to say from visual inspection which model is better.

COMPUTING AIC-SCORES :: STEP 2 :: CALCULATE AIC FROM MLE

```
get_AIC <- function(optim_fit) {
  2 * length(optim_fit$par) + 2 * optim_fit$value
}
AIC_scores <- tibble(
  AIC_exponential = get_AIC(bestExpo),
  AIC_power = get_AIC(bestPow)
)
AIC_scores
```

```
## # A tibble: 1 x 2
##   AIC_exponential AIC_power
##             <dbl>     <dbl>
## 1            41.3      57.5
```

$$\text{AIC}(M_i, D_{\text{obs}}) = 2k - 2 \log P(D_{\text{obs}} \mid \hat{\theta}_i, M_i)$$

Exponential model has lower AIC score, so it comes up as "better" under this approach.

AKAIKE WEIGHTS

$$w_{\text{AIC}}(M_i, D) = \frac{\exp(-0.5 * \Delta_{\text{AIC}}(M_i, D))}{\sum_{j=1}^{k} \exp(-0.5 * \Delta_{\text{AIC}}(M_j, D))}$$

$$\Delta_{\text{AIC}}(M_i, D) = \text{AIC}(M_i, D) - \min_j \text{AIC}(M_j, D)$$

$$P(M_i \mid D) \approx w_{\text{AIC}}(M_i, D)$$

```
delta_AIC_power <- AIC_scores$AIC_power - AIC_scores$AIC_exponential
delta_AIC_exponential <- 0
Akaike_weight_exponential <- exp(-0.5 * delta_AIC_exponential) /
  (exp(-0.5 * delta_AIC_exponential) + exp(-0.5 * delta_AIC_power))
Akaike_weight_exponential
```

```
## [1] 0.9996841
```

Based on the quantitative (approximate) interpretation of Akaike weights, we would conclude that the evidence in favor of the exponential model is very strong.

Likelihood Ratio Test

NESTED [FREQUENTIST] MODELS

- LR-test first and foremost applies to the comparison of nested models
- (simpler) model M_i is nested under (more complex) model M_j if M_i is like M_j, but fixes some of M_j's free parameters to specific values
  - M_i is the nested model
  - M_j is the nesting model or encompassing model

NESTING EXPONENTIAL MODEL

$$P(D = \langle k, N \rangle \mid \langle a, b \rangle) = \text{Binom}(k, N, a \exp(-bt))$$

with $a, b > 0$

NESTED EXPONENTIAL MODEL

$$P(D = \langle k, N \rangle \mid b) = \text{Binom}(k, N, 1.1 \exp(-bt))$$

with $b > 0$

LR-TEST FOR NESTED MODELS

- let M_0 be nested under M_1
- let d be the number of parameters free in M_1 but fixed in M_0
- the test statistic is the likelihood ratio:

$$\text{LR}(M_1, M_0) = -2\log\left(\frac{P_{M_0}(D_{\text{obs}} \mid \hat{\theta}_0)}{P_{M_1}(D_{\text{obs}} \mid \hat{\theta}_1)}\right)$$

- if M_0 is the true model, the sampling distribution is closely approximated by a χ^2-distribution with d degrees of freedom (for large data)

LR-TEST EXAMPLE

GET MLE FOR NESTED MODEL

```
nLL_expo_nested <- function(b) {
  # calculate predicted recall rates for given parameters
  theta <- 1.1*exp(-b*t)  # one-param exponential model
  # avoid edge cases of infinite log-likelihood
  theta[theta <= 0.0] <- 1.0e-4
  theta[theta >= 1.0] <- 1-1.0e-4
  # return negative log-likelihood of data
  - sum(dbinom(x = obs, prob = theta, size = 100, log = T))
}

bestExpo_nested <- optim(
  nLL_expo_nested,
  par = 0.5,
  method = "Brent",
  lower = 0,
  upper = 20
)
```

TEST STATISTIC FOR OBSERVED DATA

```
LR_observed <- 2 * bestExpo_nested$value - 2 * bestExpo$value
LR_observed
```

```
## [1] 1.098429
```

P-VALUE FOR OBSERVED DATA

```
p_value_LR_test <- 1 - pchisq(LR_observed, 1)
p_value_LR_test
```

```
## [1] 0.2946111
```

No strong evidence in the data against the assumption that the simpler nested model is correct. Therefore, preferring simplicity, the decision is usually to stick with the simpler model.

Bayes Factors

BAYES FACTOR

- Bayesian models (with priors):
  - M_1 has prior $P(\theta_1 \mid M_1)$ and likelihood $P(D \mid \theta_1, M_1)$
  - M_2 has prior $P(\theta_2 \mid M_2)$ and likelihood $P(D \mid \theta_2, M_2)$
- Bayes factor is the factor by which the prior odds need to be adjusted by rational belief update after observing D to arrive at posterior odds

$$\underbrace{\frac{P(M_1 \mid D)}{P(M_2 \mid D)}}_{\text{posterior odds}} = \underbrace{\frac{P(D \mid M_1)}{P(D \mid M_2)}}_{\text{Bayes factor}} \underbrace{\frac{P(M_1)}{P(M_2)}}_{\text{prior odds}}$$

EXPANDING BAYES FACTOR

$$\frac{P(D \mid M_1)}{P(D \mid M_2)} = \frac{\int P(\theta_i \mid M_i)\ P(D \mid \theta_i, M_i)\ \mathrm{d}\theta_i}{\int P(\theta_j \mid M_j)\ P(D \mid \theta_j, M_j)\ \mathrm{d}\theta_j}$$

- Bayes factors look at ex ante (a priori) predictions
- integration over priors → implicit (severe) punishment for model complexity
- calculating Bayes factors is computationally hard for sophisticated models

NOTATION AND INTERPRETATION

$$\mathrm{BF}_{12} = \frac{P(D \mid M_1)}{P(D \mid M_2)}$$

read: "BF in favor of model 1 over model 2"

BF_{12}	**interpretation**
1	irrelevant data
1 - 3	hardly worth ink or breath
3 - 6	anecdotal
6 - 10	now we're talking: substantial
10 - 30	strong
30 - 100	very strong
100 +	decisive (bye, bye M_2!)

BAYESIAN FORGETTING MODELS

EXPONENTIAL MODEL

$$
\begin{aligned}
P(D = \langle k, N \rangle \mid \langle a, b \rangle, M_{\exp}) &= \text{Binom}(k, N, a \exp(-bt)) \\
P(a \mid M_{\exp}) &= \text{Uniform}(a, 0, 1.5) \\
P(b \mid M_{\exp}) &= \text{Uniform}(b, 0, 1.5)
\end{aligned}
$$

POWER MODEL

$$
\begin{aligned}
P(D = \langle k, N \rangle \mid \langle c, d \rangle, M_{\text{pow}}) &= \text{Binom}(k, N, c\, t^{-d}) \\
P(d \mid M_{\text{pow}}) &= \text{Uniform}(c, 0, 1.5) \\
P(c \mid M_{\text{pow}}) &= \text{Uniform}(d, 0, 1.5)
\end{aligned}
$$

```r
# prior exponential model
priorExp = function(a, b){
  dunif(a, 0, 1.5) * dunif(b, 0, 1.5)
}
# likelihood function exponential model
lhExp = function(a, b){
  theta = a*exp(-b*t)
  theta[theta <= 0.0] = 1.0e-5
  theta[theta >= 1.0] = 1-1.0e-5
  prod(dbinom(x = obs, prob = theta, size = 100))
}

# prior power model
priorPow = function(c, d){
  dunif(c, 0, 1.5) * dunif(d, 0, 1.5)
}
# likelihood function power model
lhPow = function(c, d){
  theta = c*t^(-d)
  theta[theta <= 0.0] = 1.0e-5
  theta[theta >= 1.0] = 1-1.0e-5
  prod(dbinom(x = obs, prob = theta, size = 100))
}
```

GRID APPROXIMATION

```
# make sure the functions accept vector input
lhExp = Vectorize(lhExp)
lhPow = Vectorize(lhPow)

# define the step size of the grid
stepsize = 0.01
# calculate the "evidence" aka marginal likelihood
evidence = expand.grid(x = seq(0.005, 1.495, by = stepsize),
                       y = seq(0.005, 1.495, by = stepsize)) %>%
  mutate(lhExp = lhExp(x,y), priExp = 1 / length(x),  # uniform priors!
         lhPow = lhPow(x,y), priPow = 1 / length(x))

paste0("BF in favor of exponential model: ",
            with(evidence, sum(priExp*lhExp)/ sum(priPow*lhPow)) %>% round(2))
```

```
## [1] "BF in favor of exponential model: 1221.39"
```

NAIVE MONTE CARLO SAMPLING

$$P(D, M_i) = \int P(D \mid \theta, M_i)\, P(\theta \mid M_i)\, \mathrm{d}\theta \approx \frac{1}{n} \sum_{\theta_j \sim P(\theta \mid M_i)}^{n} P(D \mid \theta_j, M_i)$$

```r
nSamples = 1000000
a = runif(nSamples, 0, 1.5)
b = runif(nSamples, 0, 1.5)
lhExpVec = lhExp(a,b)
lhPowVec = lhPow(a,b)
paste0("BF in favor of exponential model: ",
            signif(sum(lhExpVec) / sum(lhPowVec)),6)
```

```
## [1] "BF in favor of exponential model: 1250.366"
```