Machine Learning and NeuroEngineering

机器学习与神经工程

Lecture 14 – Hierarchical Modeling

Quanying Liu （刘泉影）

SUSTech, BME department

Email: liuqy@sustech.edu.cn

BMEB212, Spring term, 2021

Lecture 11 – Gibbs Sampling & JAGS

- What is Gibbs Sampling?
  - Motivations
  - Gibbs Sampling: a Bivariate Example
  - Gibbs Sampling vs Metropolis-Hasting Sampling
- JAGS
  - Installation
  - Scripting for JAGS
- Examples of JAGS: Revisiting Some Known Models
  - Bayesian Modeling of Signal-Detection Theory (SDT)
  - A Bayesian Approach to a High-Threshold Model (1HT)
  - A Bayesian Approach to Multinomial Processing Tree (MPT)

Motivations for Gibbs Sampling

Sampling from the *joint posterior* for all the parameters may be unachievable in many situations;

$$P(x1, x2, \ldots xn)$$

We can often easily sample from the posterior for one parameter given knowledge of the other parameter values.

- *Gibbs Sampler* samples from *conditional distributions*;
- By iterating through the parameters, sampling each conditional upon the others being constant, the Gibbs sampler manages to provide us with posterior distributions for each parameter.

Gibbs Sampler

1. Initialise $x_{0,1:n}$.
2. For $i = 0$ to $N - 1$
   - Sample $x_1^{(i+1)} \sim p(x_1 | x_2^{(i)}, x_3^{(i)}, \ldots, x_n^{(i)})$.
   - Sample $x_2^{(i+1)} \sim p(x_2 | x_1^{(i+1)}, x_3^{(i)}, \ldots, x_n^{(i)})$.

   $$\vdots$$

   - Sample $x_j^{(i+1)} \sim p(x_j | x_1^{(i+1)}, \ldots, x_{j-1}^{(i+1)}, x_{j+1}^{(i)}, \ldots, x_n^{(i)})$.

   $$\vdots$$

   - Sample $x_n^{(i+1)} \sim p(x_n | x_1^{(i+1)}, x_2^{(i+1)}, \ldots x_{n-1}^{(i+1)})$.

C Andrieu et al (2003) An Introduction to MCMC for Machine Learning

Gibbs Sampling: a Bivariate Example

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \triangleq \frac{1}{(2\pi)^{D/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

Let us consider a 2d example.

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

$$x^{(i+1)} \sim f(x|y^{(i)}) = N\left(\rho\left(\frac{\sigma_x}{\sigma_y}\right)y^{(i)}, \sqrt{\sigma_x^2(1-\rho^2)}\right)$$

$$y^{(i+1)} \sim f(y|x^{(i+1)}) = N\left(\rho\left(\frac{\sigma_y}{\sigma_x}\right)x^{(i+1)}, \sqrt{\sigma_y^2(1-\rho^2)}\right)$$

```
#gibbs sampling
sxt1mr <- sqrt(sigx^2*(1-rho^2))
syt1mr <- sqrt(sigy^2*(1-rho^2))
rxy <- rho*(sigx/sigy)
ryx <- rho*(sigy/sigx)
xsamp <- ysamp <- rep(0,nsamples)
xsamp[1] <- -2
ysamp[1] <- 2
for (i in c(1:(nsamples-1))) {
   xsamp[i+1] <- rnorm(1, mean=rxy*ysamp[i], sd=sxt1mr)
   ysamp[i+1] <- rnorm(1, mean=ryx*xsamp[i+1], sd=syt1mr)
}
```

SEE CODE: Lecture11_1_gibbs.R

Gibbs Sampling vs Metropolis-Hastings Sampling

Gibbs sampling only applies to:

1) multiple variables
2) the conditional distributions for each single variable are known

M-H sampling can **also** operate in univariate situations

C Andrieu et al (2003)

Gibbs sampling do not reject samples. Acceptance probability is 1.

M-H sampling reject the proposals with certain probability. Acceptance probability is

$$\min\left\{1, \frac{p(x^\star)q\left(x^{(i)} \mid x^\star\right)}{p\left(x^{(i)}\right)q\left(x^\star|x^{(i)}\right)}\right\}$$

JAGS - installation

JAGS stands for ‘Just Another Gibbs Sampler’.

To install JAGS:

Download the relevant files from this website:

http://mcmc-jags.sourceforge.net/

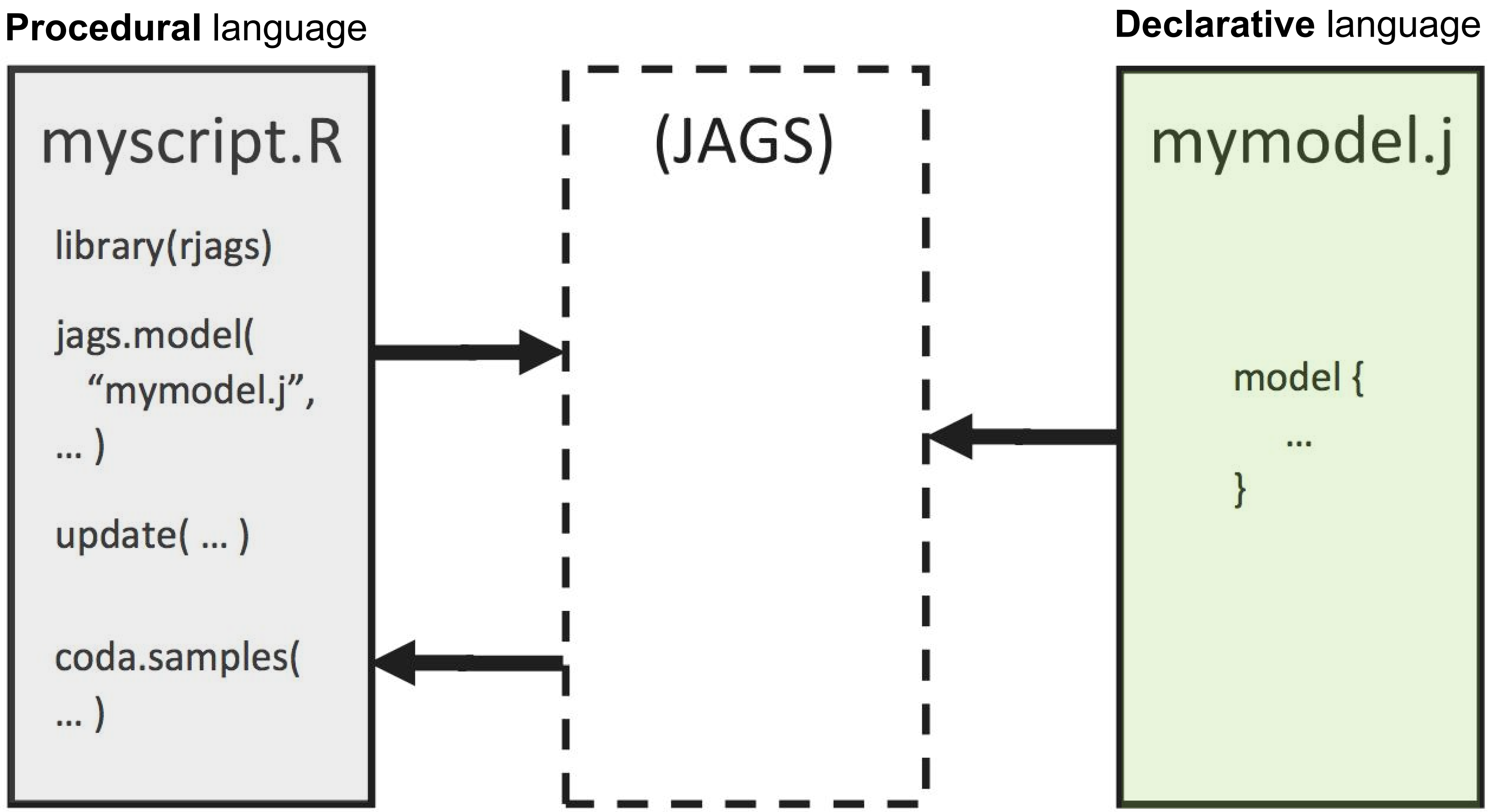
Run the installer

To call JAGS using R:

install.packages("rjags")

require(rjags) or library(rjags)

JAGS - Scripting



JAGS - Scripting

myscript.R

```r
require(rjags)

N <- 1000
x <- rnorm(N, 0, 2)

myj <- jags.model("mymodel.j",
                  data = list("xx" = x, "N" = N))
update(myj,n.iter=1000)
mcmcfin<-coda.samples(myj,c("mu", "tau"),5000)

summary(mcmcfin)
plot(mcmcfin)
```

mymodel.j

```
#Gaussian
model {
  #model the data
        for (i in 1:N) {
              xx[i] ~ dnorm(mu, tau)
        }

        #priors for parameters
        mu ~ dunif(-100,100)
        tau <- pow(sigma, -2)
        sigma ~ dunif(0, 100)
}
```

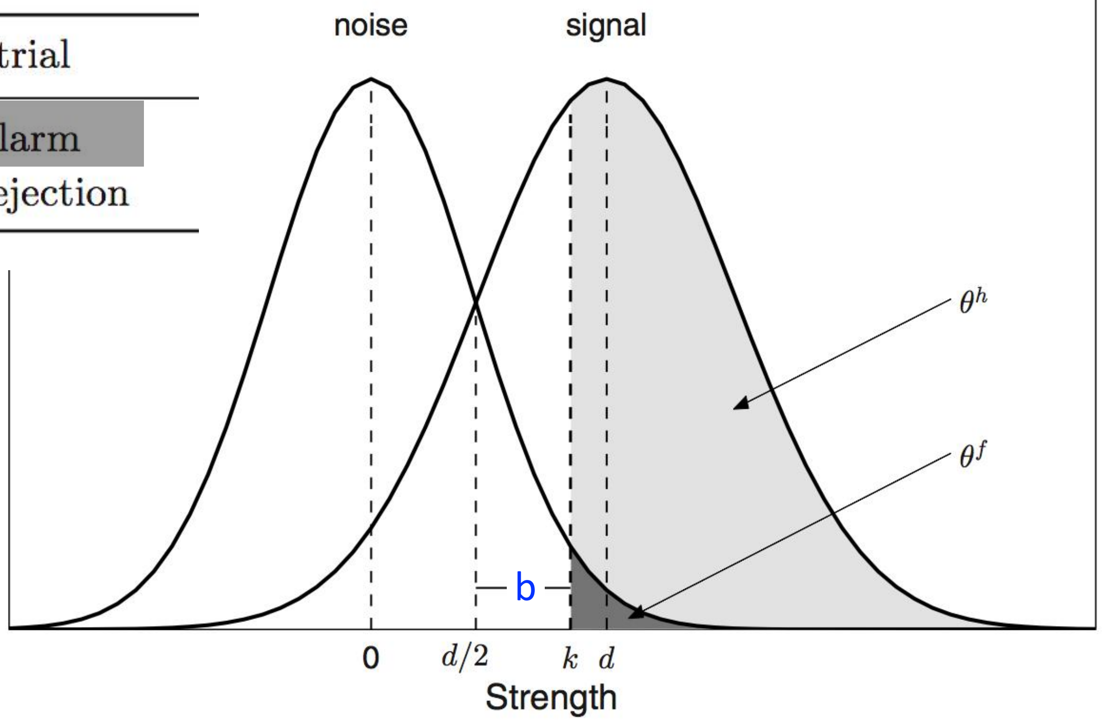
Examples of JAGS - SDT

Revisit SDT we have learned in Lecture 9

Table 11.1 Basic signal detection theory data and terminology.

	Signal trial	Noise trial
Yes response	Hit	False alarm
No response	Miss	Correct rejection

		Test Item	
		Old	New
Response	"Old"	60%	11%
	"New"	40%	89%



Equal-variance Gaussian signal detection theory framework.

Examples of JAGS – Scripting for SDT

SDT.R

```
library(rjags)
h <- 60
f <- 11
sigtrials <- noistrials <- 100
oneinit <- list(d=0, b=0)
myinits <- list(oneinit)[rep(1,4)]
myinits <- lapply(myinits,FUN=function(x) lapply(x,
FUN=function(y) y+rnorm(1,0,.1)))
sdtj <- jags.model("SDT.j", data = list("h"=h, "f"=f,
                        "sigtrials"=sigtrials,"noistrials"=noistrials),
               inits=myinits, n.chains=4)
update(sdtj,n.iter=1000)
parameters <- c("d", "b", "phih", "phif")
mcmcfin<-coda.samples(sdtj, parameters, 5000)

summary(mcmcfin)
plot(mcmcfin)
gelman.plot(mcmcfin)
```

SDT.j

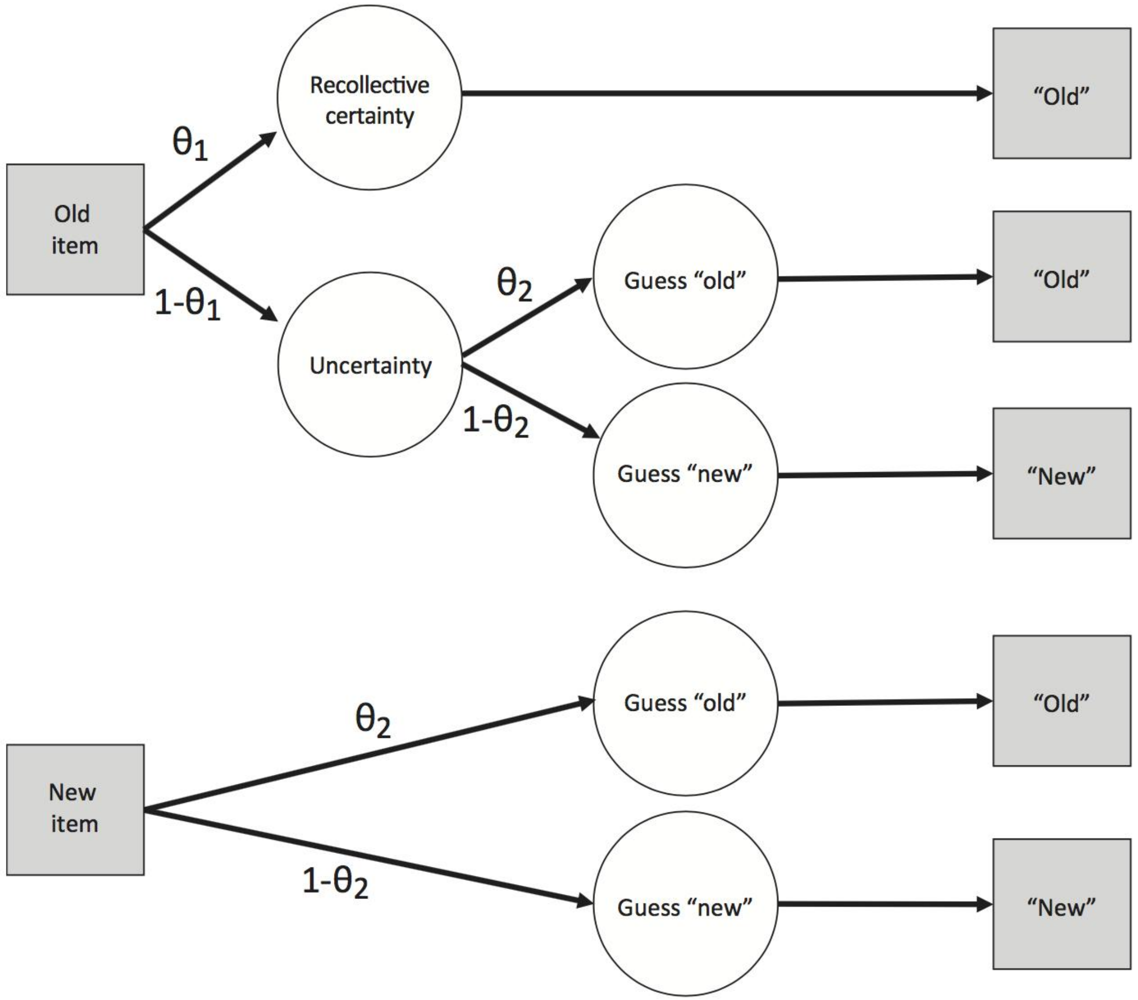
```
model{
# priors for discriminability and bias
   d ~ dnorm(1,1)
   b ~ dnorm(0,1)

   # express as areas under curves
   phih <- phi(d/2-b)   #normal cdf
   phif <- phi(-d/2-b)

   # Observed hits and false alarms
   h ~ dbin(phih, sigtrials)
   f ~ dbin(phif, noistrials)
}
```

Examples of JAGS – A High-Threshold Model

The high-threshold(1HT) model of recognition memory



Unlike signal-detection theory, the 1HT model does not contain a response criterion (d/2+b).

1HT model relies on two parameters:

1. the probability of being in the certain state, captured by the parameter θ1,
2. the probability of guessing "old", described by θ2, when in the state of uncertainty.

$$p(hit) = \theta_1 + (1 - \theta_1)\theta_2,$$

$$p(FA) = \theta_2,$$

Examples of JAGS – Scripting for 1HT

1HT.R

```
library(rjags)
#provide data from experiment
h <- 60
f <- 11
sigtrials <- noistrials <- 100

#define JAGS model
onehtj <- jags.model("1HT.j",
              data = list("h"=h, "f"=f,
                    "sigtrials"=sigtrials,
                    "noistrials"=noistrials),
               n.chains=4)

# burnin
update(onehtj,n.iter=1000)

# perform MCMC
parameters <- c("th1", "th2", "predh", "predf")
mcmcfin<-coda.samples(onehtj, parameters, 5000)
```

1HT.j

```
# High-threshold model
model{
   # priors for MPT parameters
   th1 ~ dbeta(1,1)
   th2 ~ dbeta(1,1)

   # predictions for responses
   predh <- th1+(1-th1)*th2
   predf <- th2

   # Observed responses
   h ~ dbin(predh, sigtrials)
   f ~ dbin(predf, noistrials)
}
```

Lecture 14 – Hierarchical Modeling

- What is Hierarchical Modeling?
  - Motivations → data from multiple participants
  - Frequentist vs Bayesian

- Bayesian Hierarchical Modeling
  - Graphical Models
    - Nodes, arrows, plate, subscript
  - Example: Hierarchical Modeling of SDT
  - Example: Hierarchical Modeling of Forgetting
  - Example: Hierarchical Modeling of Intertemporal Preferences
    - Present Subjective Value
    - The neural correlates of Subjective Value during intertemporal choice

Hierarchical Modeling

Hierarchical Modeling or Multilevel Modeling

is a statistically rigorous way

to make scientific inferences

about a population (or specific object)

based on many individuals (or observations).

We have learnt two approaches to account for data from multiple participants:

1) fitting a model to individual participants;

2) fitting aggregate data (i.e. Vincent averaging);

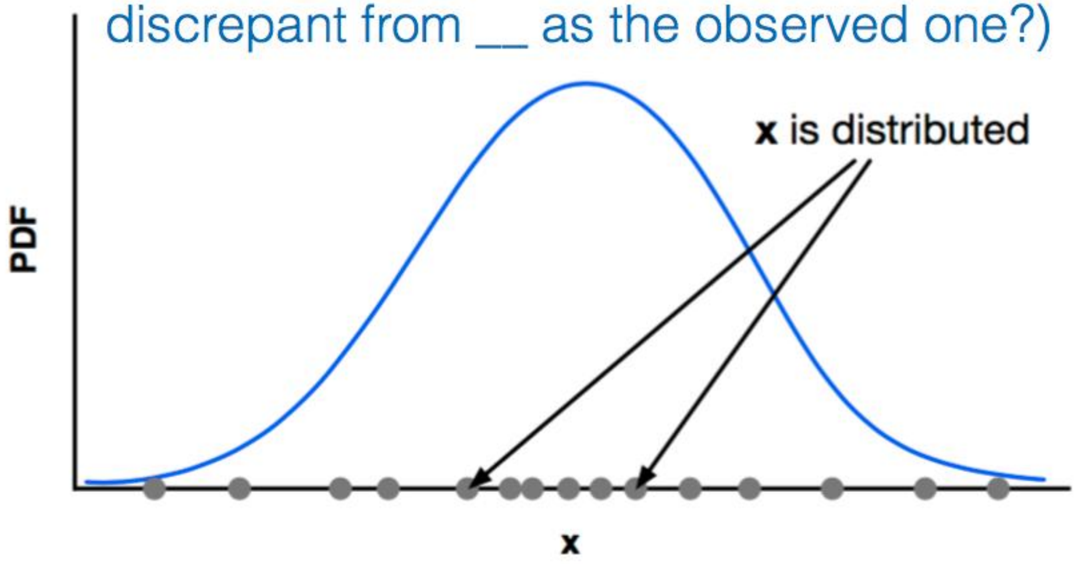
Hierarchical models exploit some degree of dependence between participants.

Frequentist multi-level modeling techniques exist,

but we will discuss the Bayesian approach today.

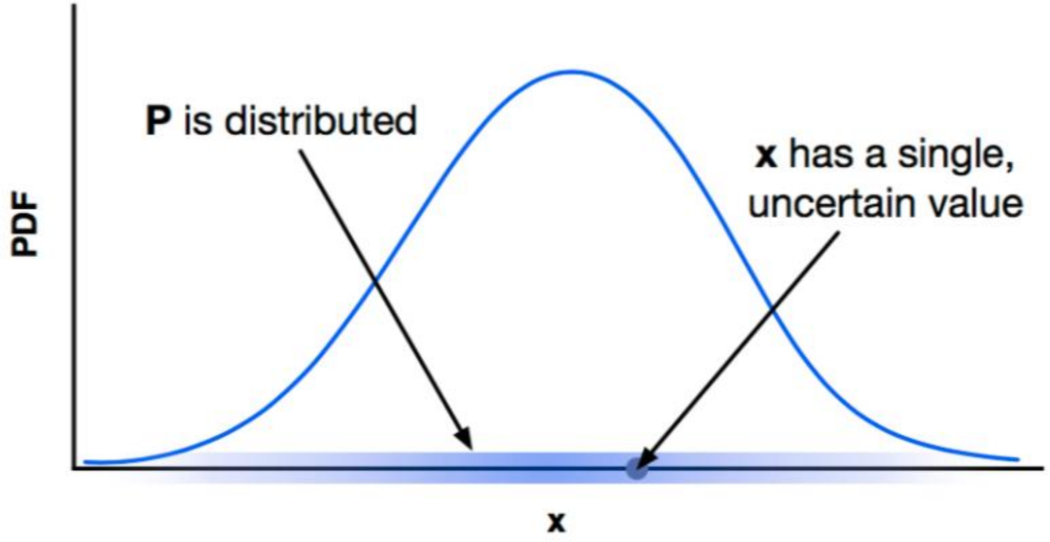
Frequentist: variability of sample

(If __ is the true value, what fraction of many hypothetical datasets would be as or more discrepant from __ as the observed one?)



Bayesian: uncertainty of inference

(What's the probability that __ is the true value given the current data?)

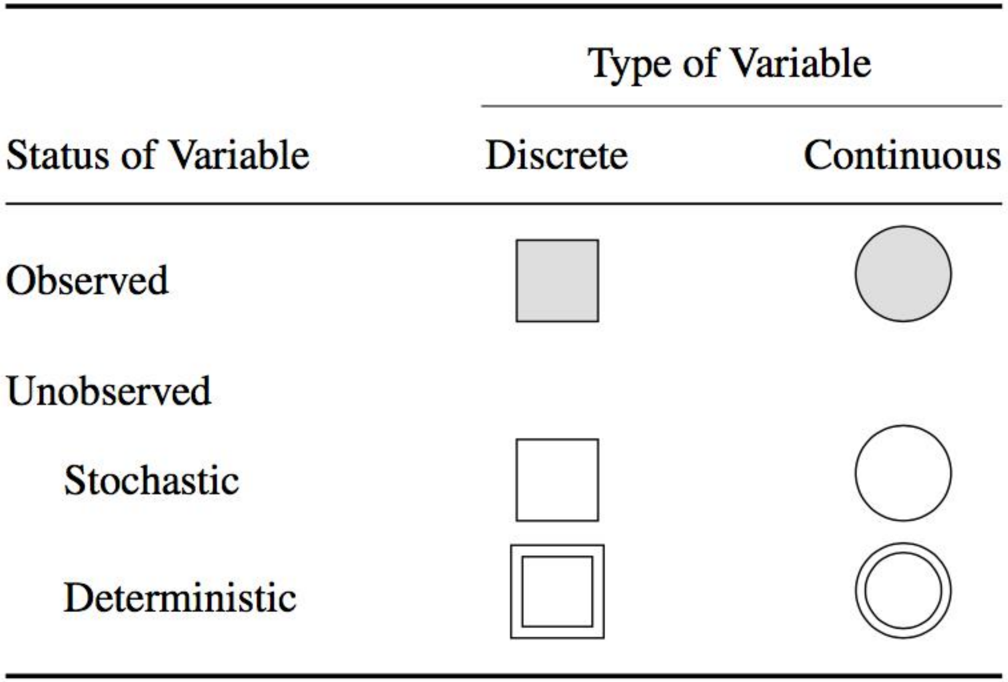


Graphical Models

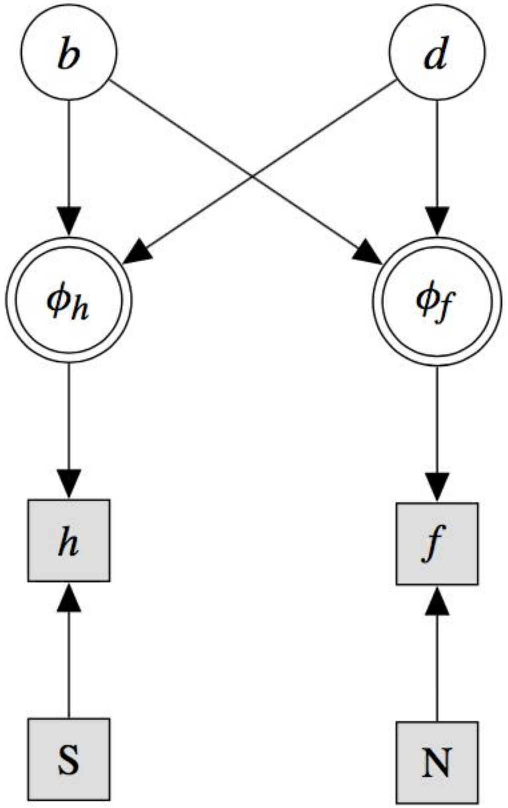
Graphical Models is a useful way of picturing and conceptualizing Bayesian Models.

- **Nodes**: variables (i.e. data, model parameters, model predictions)
- **Arrows**: dependencies between variables

Notation for nodes used in graphical models



Graphical model for the SDT model in Lecture10



$$d \sim Gaussian(1, 1)$$

$$b \sim Gaussian(0, 1)$$

$$\phi_h \leftarrow \Phi\left(\tfrac{1}{2}d - b\right)$$

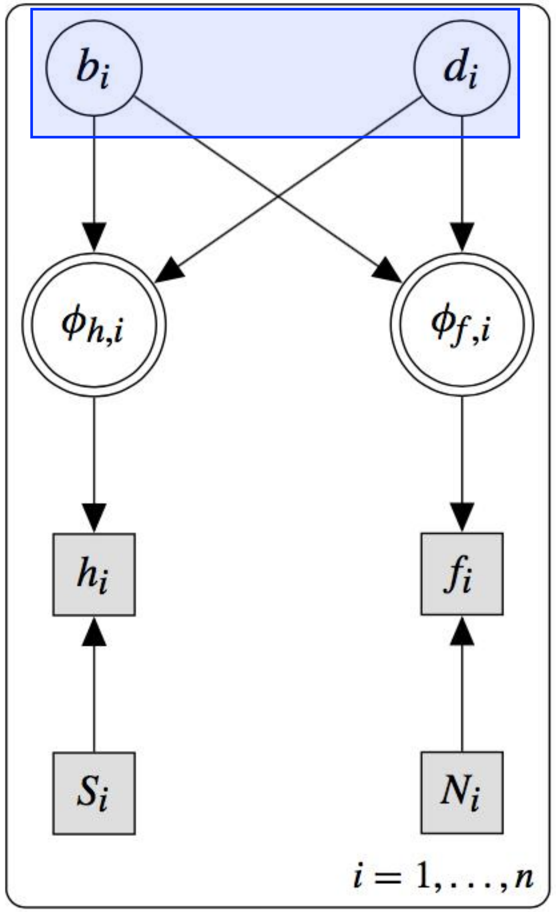
$$\phi_f \leftarrow \Phi\left(-\tfrac{1}{2}d - b\right)$$

$$h \sim Binomial(\phi_h, S)$$

$$f \sim Binomial(\phi_f, N)$$

Graphical Model for SDT on each subject

Graphical model for a SDT model that is applied to a number of different participants (or conditions).



Prior

$$d_i \sim Gaussian(1, 1)$$

$$b_i \sim Gaussian(0, 1)$$

$$\phi_{h,i} \leftarrow \Phi\left(\tfrac{1}{2}d_i - b_i\right)$$

$$\phi_{f,i} \leftarrow \Phi\left(-\tfrac{1}{2}d_i - b_i\right)$$

$$h_i \sim Binomial(\phi_{h,i}, S_i)$$

$$f_i \sim Binomial(\phi_{f,i}, N_i)$$

A plate signals that all variables enclosed within it are replicated across a number of participants (or conditions).

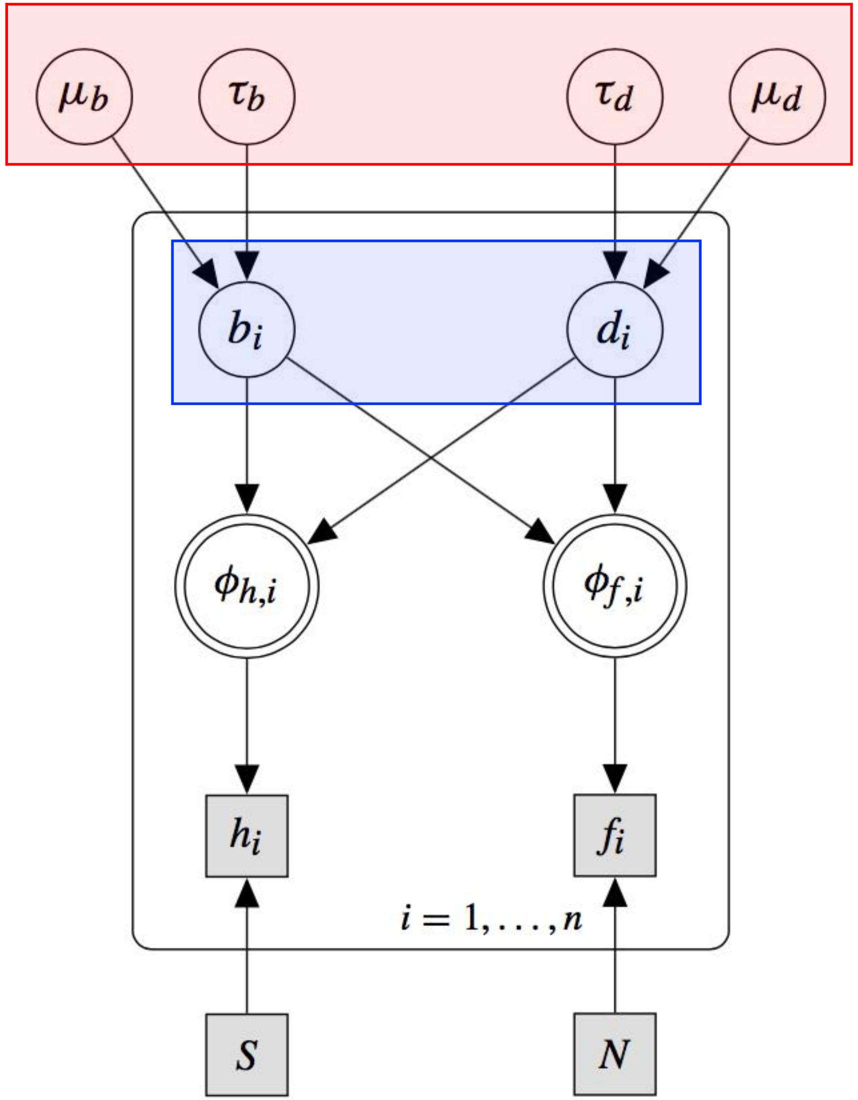
The subscript *i* represents the i^{th} participant (or condition).

This graphical model would be implemented by running the SDT model on the data from each subject.

It will yield a large number of independent parameter estimates to capture and characterize *individual differences*.

Hierarchical SDT model

Parent distribution



$$\mu_b \sim Gaussian(0, \epsilon)$$

$$\tau_b \sim Gamma(\epsilon, \epsilon)$$

$$\mu_d \sim Gaussian(0, \epsilon)$$

$$\tau_d \sim Gamma(\epsilon, \epsilon)$$

$$d_i \sim Gaussian(\mu_d, \tau_d)$$

$$b_i \sim Gaussian(\mu_b, \tau_b)$$

$$\phi_{h,i} \leftarrow \Phi\left(\tfrac{1}{2}d_i - b_i\right)$$

$$\phi_{f,i} \leftarrow \Phi\left(-\tfrac{1}{2}d_i - b_i\right)$$

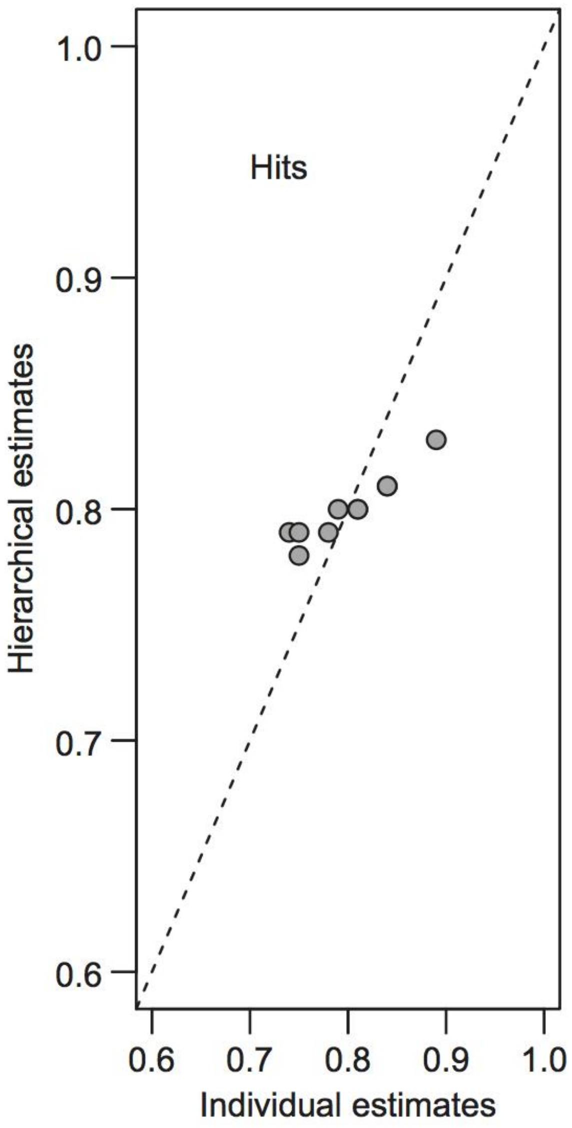
$$h_i \sim Binomial(\phi_{h,i}, S)$$

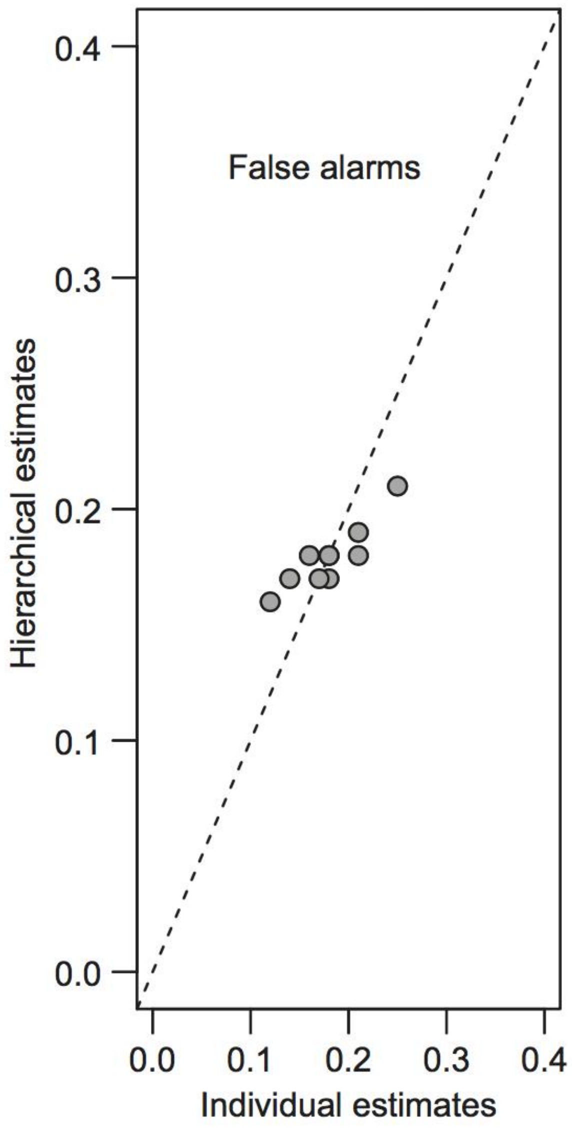
$$f_i \sim Binomial(\phi_{f,i}, N)$$

Lecture14_1_SDThierarch.j

```
# Hierarchical Signal Detection Theory
model{  # parent distributions for priors
            mud ~ dnorm(0,epsilon)
            mub ~ dnorm(0,epsilon)
            taud ~ dgamma(epsilon,epsilon)
            taub ~ dgamma(epsilon,epsilon)
#modeling all n subjects
            for (i in 1:n) {
# priors for discriminability and bias
                        d[i] ~ dnorm(mud,taud)
                        b[i] ~ dnorm(mub,taub)
# predictions for hits and false alarms
                        phih[i] <- phi( d[i]/2 - b[i])
                        phif[i] <- phi(-d[i]/2 - b[i])
# Observed hits and false alarms
                        h[i] ~ dbin(phih[i],sigtrials)
                        f[i] ~ dbin(phif[i],noistrials)
            }
}
```

Shrinkage





Shrinkage: the attenuation of individual differences in hierarchical models

RMSD is lower for hierarchical estimates.

Stein's Paradox:

The best estimate of a person's true ability is not their own performance,

but an adjusted measures that brings an individual's performance estimate more in line with the observations of all other individuals.

Hierarchical modeling for forgetting

We forget.

Sometimes we forget in seconds. Sometimes we take several decades to forget.

We have intense research interest over its causes, neural mechanisms, as well as the shape of forgetting.

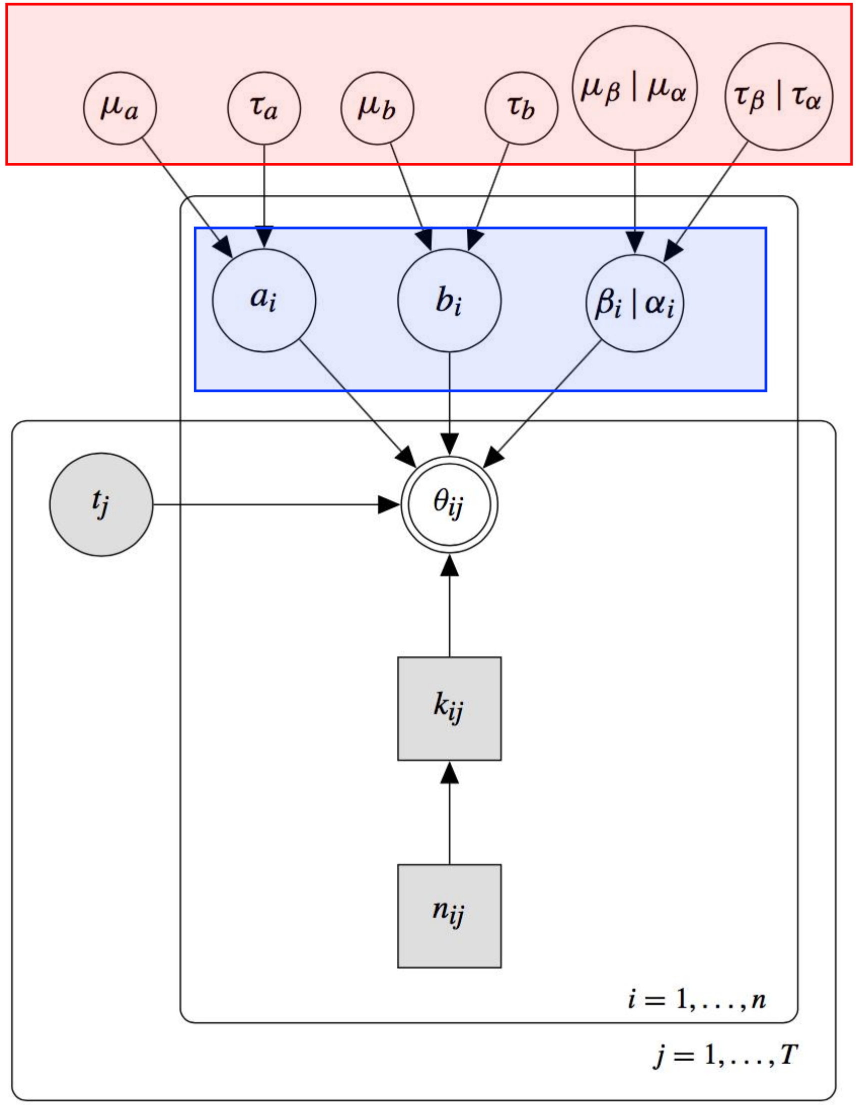
Two candidate functions to describe the shape of the forgetting function (or retention curve):

exponential function $$\theta_t = a + (1 - a) \times b \times e^{-\alpha \times t},$$

power function $$\theta_t = a + (1 - a) \times b \times (1 + t)^{-\beta}.$$

Hierarchical modeling for forgetting

Parent distribution



$\mu_{\alpha|\beta} \sim \text{Uniform}(0, 1)$

$\tau_{\alpha|\beta} \sim \text{Gamma}(\epsilon, \epsilon)$

$\mu_a \sim \text{Uniform}(0, 1)$

$\tau_a \sim \text{Gamma}(\epsilon, \epsilon)$

$\mu_b \sim \text{Uniform}(0, 1)$

$\tau_b \sim \text{Gamma}(\epsilon, \epsilon)$

$a_i \sim \text{Gaussian}(\mu_a, \tau_a) \;\; 0 < a_i < 1$

$b_i \sim \text{Gaussian}(\mu_b, \tau_b) \;\; 0 < b_i < 1$

$\alpha_i \sim \text{Gaussian}(\mu_\alpha, \tau_\alpha) \;\; 0 < \alpha_i < 1$

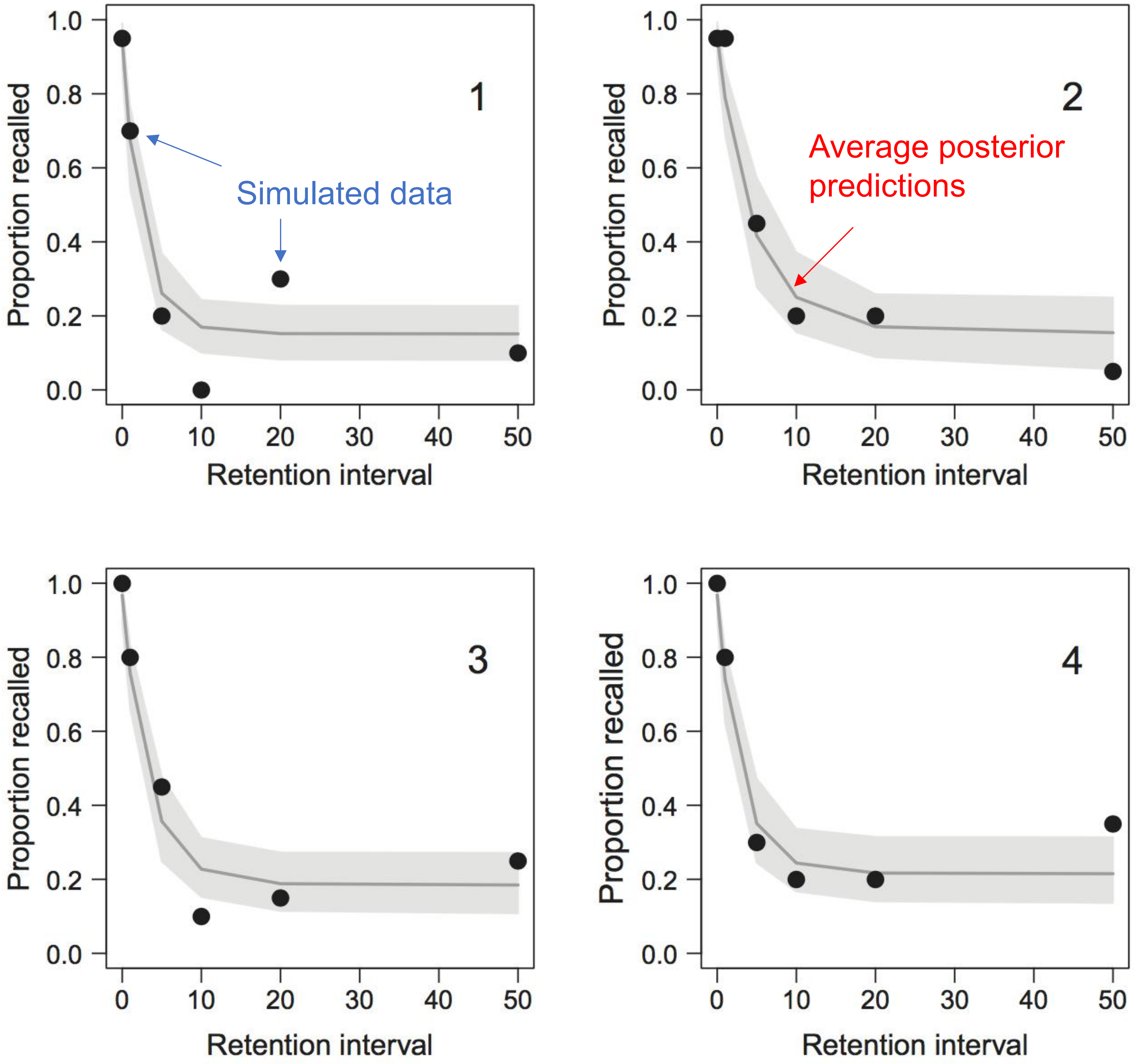
$\beta_i \sim \text{Gaussian}(\mu_\beta, \tau_\beta) \;\; 0 < \alpha_i < 1$

$$\theta_{i,j} \leftarrow \begin{cases} a_i + (1 - a_i) \times b_i \times \exp(-\alpha_i t_j) \\ a_i + (1 - a_i) \times b_i \times (1 + t_j)^{-\beta_i} \end{cases}$$

$k_{ij} \sim Binomial(\theta_{ij}, n_{ij})$

Lecture14_2_hierarchforgexp.j

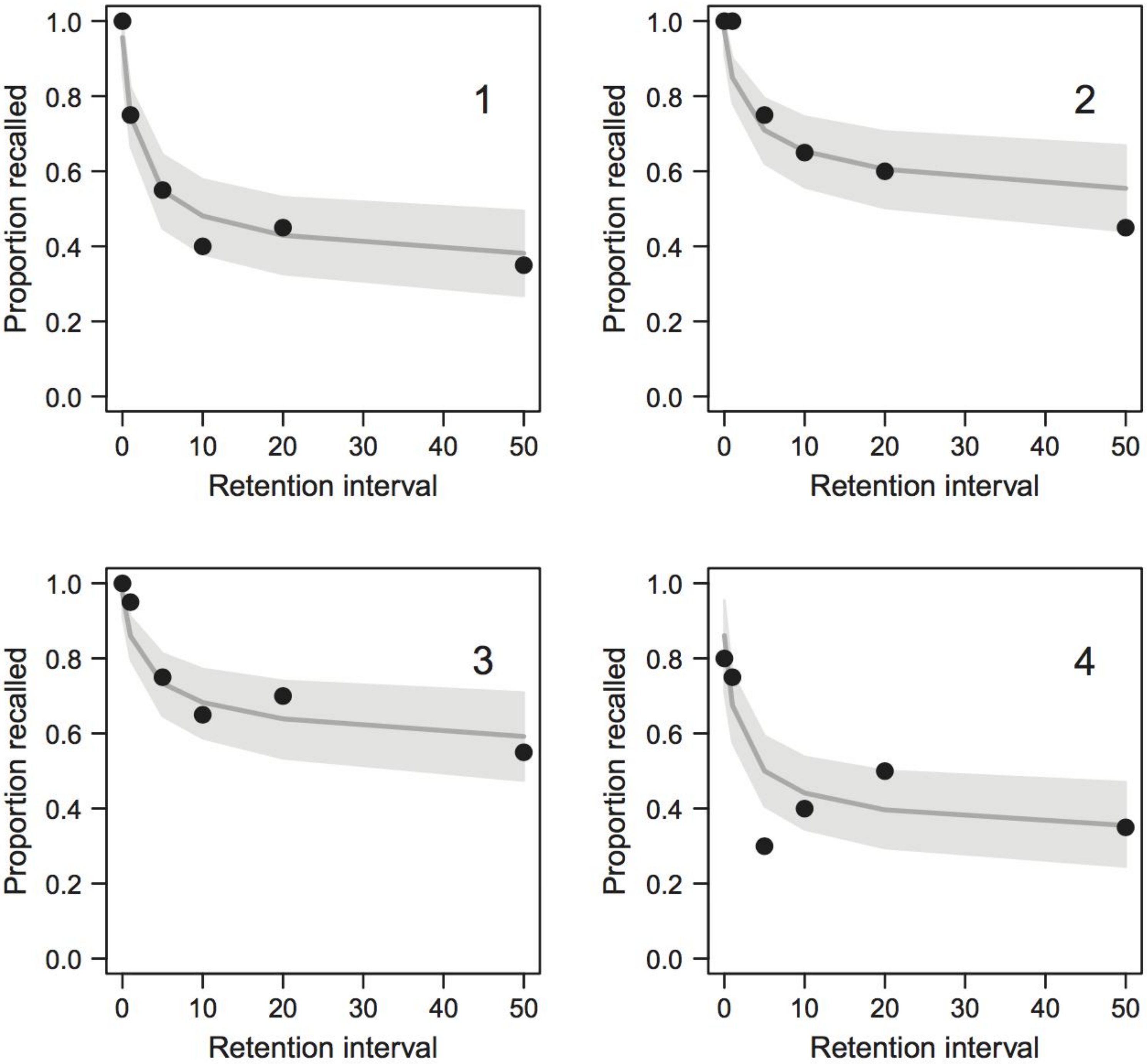
```
# hierarchical exponential forgetting model
model{  # Priors for parent Distributions
  mualpha  ~ dunif(0,1)
  taualpha ~ dgamma(epsilon + 0.1,epsilon)
  mua      ~ dunif(0,1)
  taua     ~ dgamma(epsilon + 0.1,epsilon)
  mub      ~ dunif(0,1)
  taub     ~ dgamma(epsilon + 0.1,epsilon)
# individual sampled parameters
  for (i in 1:ns){
    alpha[i] ~ dnorm(mualpha,taualpha)T(0,1)
    a[i]     ~ dnorm(mua,taua)T(0,1)
    b[i]     ~ dnorm(mub,taub)T(0,1)  }
# predictions for each subject at each lag
  for (i in 1:ns){
    for (j in 1:nt){
       theta[i,j] <- a[i]+(1-a[i])*b[i]*exp(-alpha[i]*t[j]) } }
# observed data
  for (i in 1:ns){
    for (j in 1:nt){
      k[i,j] ~ dbin(theta[i,j],n) }}
}
```



Results of a run of the hierarchical exponential forgetting model.

Each panel shows simulated data (large plotting symbols) and average posterior predictions (solid gray line).

The central 95% of the posterior predictive distribution (gray shaded area) for each subject.



SEE CODE:
Lecture11_3_hierarchforgpow.R
for the power model

Model comparisons
→ AIC or BIC …

Hierarchical modeling for Inter-Temporal Preferences

Let’s do a task together!

Inter-Temporal Preferences ...

扫一扫二维码打开或分享给好友

Hierarchical modeling for Inter-Temporal Preferences

A task: choose 1 or 2

Q1 (1) 5*yuan* now vs (2) 10*yuan* 2weeks later

Q2 (1) 12*yuan* now vs (2) 20*yuan* 3weeks later

Q3 (1) 15*yuan* now vs (2) 30*yuan* 4weeks later

Q4 (1) 8*yuan* now vs (2) 50*yuan* 10weeks later

Q5 (1) 15*yuan* now vs (2) 50*yuan* 5weeks later

Q6 (1) 5*yuan* now vs (2) 200*yuan* 6weeks later

Q7 (1) 10*yuan* now vs (2) 500*yuan* 10weeks later

Q8 (1) 50*yuan* now vs (2) 8,000*yuan* 1year later

Hierarchical modeling for Inter-Temporal Preferences

A task: choose 1 or 2

Q9 (1) 10*yuan* now vs (2) 50*yuan* 8weeks later

Q10 (1) 20*yuan* now vs (2) 50*yuan* 6weeks later

Q11 (1) 20*yuan* now vs (2) 30*yuan* 5weeks later

Q12 (1) 8*yuan* now vs (2) 50*yuan* 10weeks later

Q13 (1) 4*yuan* now vs (2) 100*yuan* 1years later

Q14 (1) 50*yuan* now vs (2) 150*yuan* 4weeks later

Q15 (1) 20*yuan* now vs (2) 1,000*yuan* 8weeks later

Q16 (1) 25*yuan* now vs (2) 10,000*yuan* 2years later

Hierarchical modeling for Inter-Temporal Preferences

We value the present more than the future. --> 'intertemporal ' preferences

Given the intertemporal choices are at the heart of many policy decisions, such as the retirement saving plans, long-term investments, climate change...

Modeling the decision-making under intertemporal preferences:

Present Subjective Value (PSV) of the option 1 and option 2

$$V^B = B \times \frac{1}{1 + kD}$$

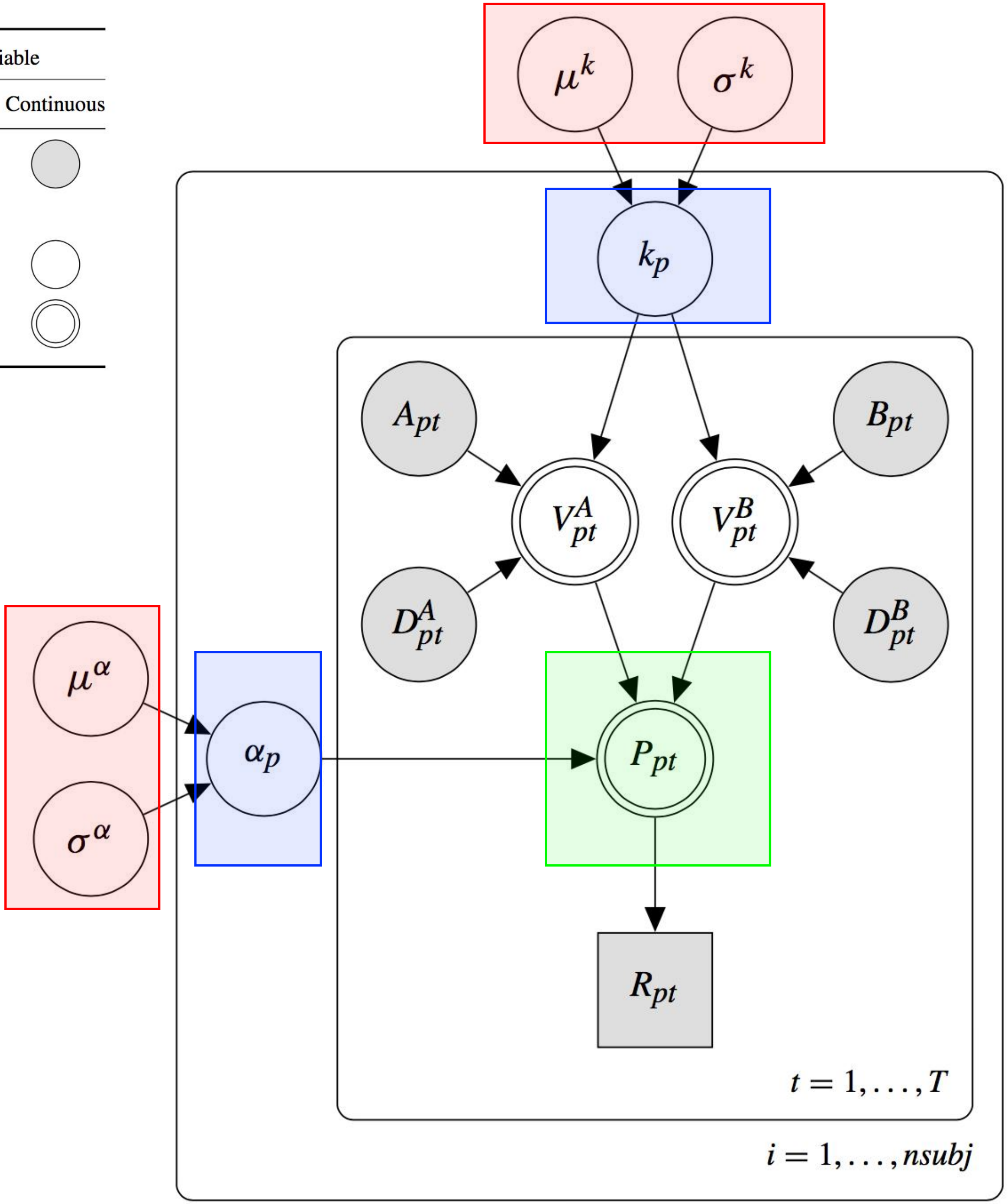
B: monetary amount
D: delay
k: discount function

Choosing Between option 1 and option 2

$$P(choose\ V^B) = \Phi\left(\frac{V^B - V^A}{\alpha}\right)$$

V^A, V^B : PSV of option1 and option2
α : temperature
ϕ: cumulative normal distribution function

	Type of Variable	
Status of Variable	Discrete	Continuous
Observed	■	●
Unobserved		
Stochastic	□	○
Deterministic	⧈	◎



$$\mu^k \sim \text{Gaussian}(0, 0.01)$$

$$\sigma^k \sim \text{Uniform}(0, 100)$$

$$\mu^\alpha \sim \text{Uniform}(0, 5)$$

$$\sigma^\alpha \sim \text{Uniform}(0, 5)$$

$$k_p \sim \text{Gaussian}(\mu^k, \sigma^k) \;\; k_p > 0$$

$$\alpha_p \sim \text{Gaussian}(\mu^\alpha, \sigma^\alpha) \;\; \alpha_p > 0$$

$$V^A_{pt} \leftarrow A_{pt}/(1 + k_p \times D^A_{pt})$$

$$V^B_{pt} \leftarrow B_{pt}/(1 + k_p \times D^B_{pt})$$

$$P_{pt} \leftarrow \Phi\left(\frac{V^B_{pt} - V^A_{pt}}{\alpha_p}\right)$$

$$R_{pt} \sim Bernoulli(P_{pt})$$

Lecture14_4_hierarchicalITC.j

```
model{
# k (steepness of hyperbolic discounting)
    groupkmu        ~ dnorm(0, 1/100)
    groupksigma     ~ dunif(0, 100)
# comparison acuity (alpha)
    groupALPHAmu        ~ dunif(0,5)
    groupALPHAsigma     ~ dunif(0,5)
# Participant-level parameters
    for (p in 1:nsubj){
      k[p]       ~ dnorm(groupkmu, 1/(groupksigma^2)) T(0,)
      alpha[p]    ~ dnorm(groupALPHAmu, 1/(groupALPHAsigma^2)) T(0,)
      for (t in 1:T) {
# calculate present subjective value for each reward
          VA[p,t] <- A[p,t] / (1+k[p]*DA[p,t])
          VB[p,t] <- B[p,t] / (1+k[p]*DB[p,t])
# Psychometric function yields predicted choice
          P[p,t] <-  phi( (VB[p,t]-VA[p,t]) / alpha[p] )
# Observed responses
          R[p,t] ~ dbern(P[p,t])
     }  }
}
```

$$\mu^k \sim \text{Gaussian}(0, 0.01)$$
$$\sigma^k \sim \text{Uniform}(0, 100)$$
$$\mu^\alpha \sim \text{Uniform}(0, 5)$$
$$\sigma^\alpha \sim \text{Uniform}(0, 5)$$

$$k_p \sim \text{Gaussian}(\mu^k, \sigma^k) \;\; k_p > 0$$
$$\alpha_p \sim \text{Gaussian}(\mu^\alpha, \sigma^\alpha) \;\; \alpha_p > 0$$

$$V^A_{pt} \leftarrow A_{pt}/(1 + k_p \times D^A_{pt})$$
$$V^B_{pt} \leftarrow B_{pt}/(1 + k_p \times D^B_{pt})$$

$$P_{pt} \leftarrow \Phi\left(\frac{V^B_{pt} - V^A_{pt}}{\alpha_p}\right)$$

$$R_{pt} \sim \mathit{Bernoulli}(P_{pt})$$

Lecture 14 – Hierarchical Modeling

- What is Hierarchical Modeling?
  - Motivations → data from multiple participants
  - Frequentist vs Bayesian

- Bayesian Hierarchical Modeling
  - Graphical Models
    - Nodes, arrows, plate, subscript
  - Example: Hierarchical Modeling of SDT
  - Example: Hierarchical Modeling of Forgetting
  - Example: Hierarchical Modeling of Intertemporal Preferences
    - Present Subjective Value
    - The neural correlates of Subjective Value during intertemporal choice

Possible course projects

- **Implement your task with pygame, and collect behavioral data, and build a model to fit the data**

  The task can be from previous literature.

  The parameters of your model should be meaningful and interpretable.

- **EEG data analysis: EEG signal processing, feature extraction**

  EEG data can be from some open datasets, or previous literature, or from my lab.

  Correlate the EEG features with your model parameters

- **Deep learning applications in EEG data analysis**

  EEG denoise by CNN / RNN / transformer

  EEG data generation by GAN / VAE (32 channel → 64channel, or generate a bad channel)

  end-to-end classification with EEG data

Inter-Temporal Preferences ...
扫一扫二维码打开或分享给好友

Homework 4

Modelling the Inter-Temporal Preferences

DDL: May 10, 2021，上课前
把code和结果图 做成pdf, 发邮件给曲由之 12031145@mail.sustech.edu.cn

Requirements

1. Fit your own data, and get your discount k
2. Apply the hierarchical models to all the data from your classmates
3. Compare the k values from **girls** and **boys** (t-test)
4. Investigate the relationship between k and the **personal characteristics** (correlation analysis)
5. Investigate whether **being single** will impact k values

Textbooks

- Chapter 9 (Hierarchical Modeling)

Course Project

- Find your teammates, **form a team**, discuss about your course project, and read 4 papers relevant to your course project (until May 10)
- **Do** the course project (until May 24)
- **Finalize** the course project (until May 26)
- The final presentation will be a **poster** presentation next to 大榕树 (on June 1)