




Bayesian Statistics and Hierarchical Bayesian Modeling for Psychological Science

Lecture 13

Lei Zhang

Social, Cognitive and Affective Neuroscience Unit (SCAN-Unit)
Department of Basic Psychological Research and Research Methods

https://github.com/lei-zhang/BayesCog_Wien

lei.zhang@univie.ac.at
lei-zhang.net
 @lei_zhang_lz



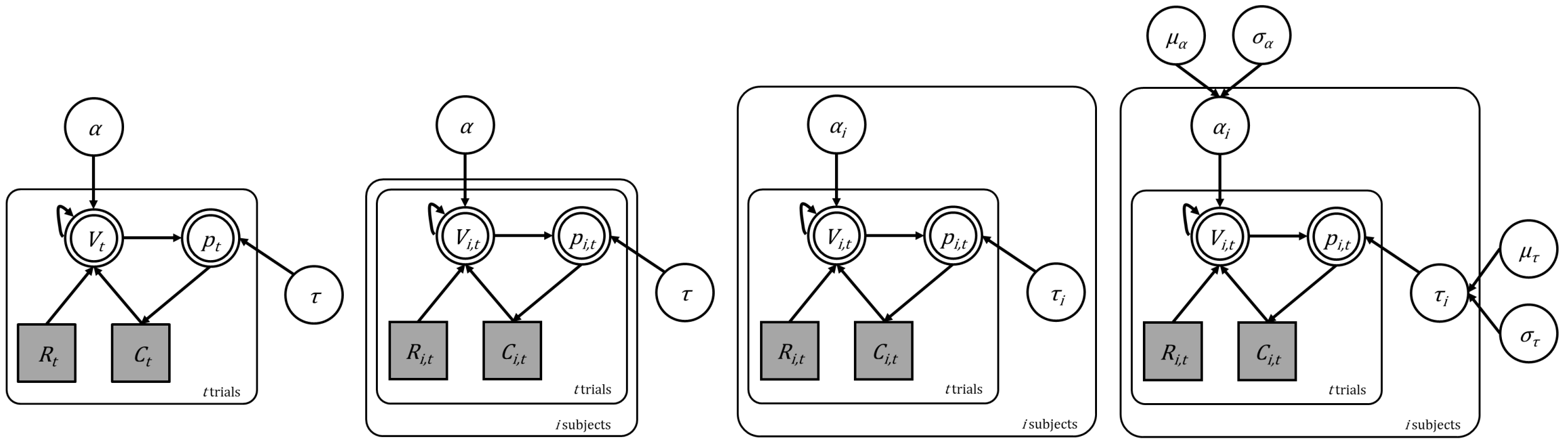
universität
wien
Fakultät für Psychologie

Review comments

- Summary
 - high-level summary
 - does not have to be very detailed, e.g. N, re-describe the procedure
- Major
 - Justification of design/analysis
 - unclear intro
 - missing literatures
 - alternative interpretation
- Minor
 - typo
 - inconsistency
 - improper analysis

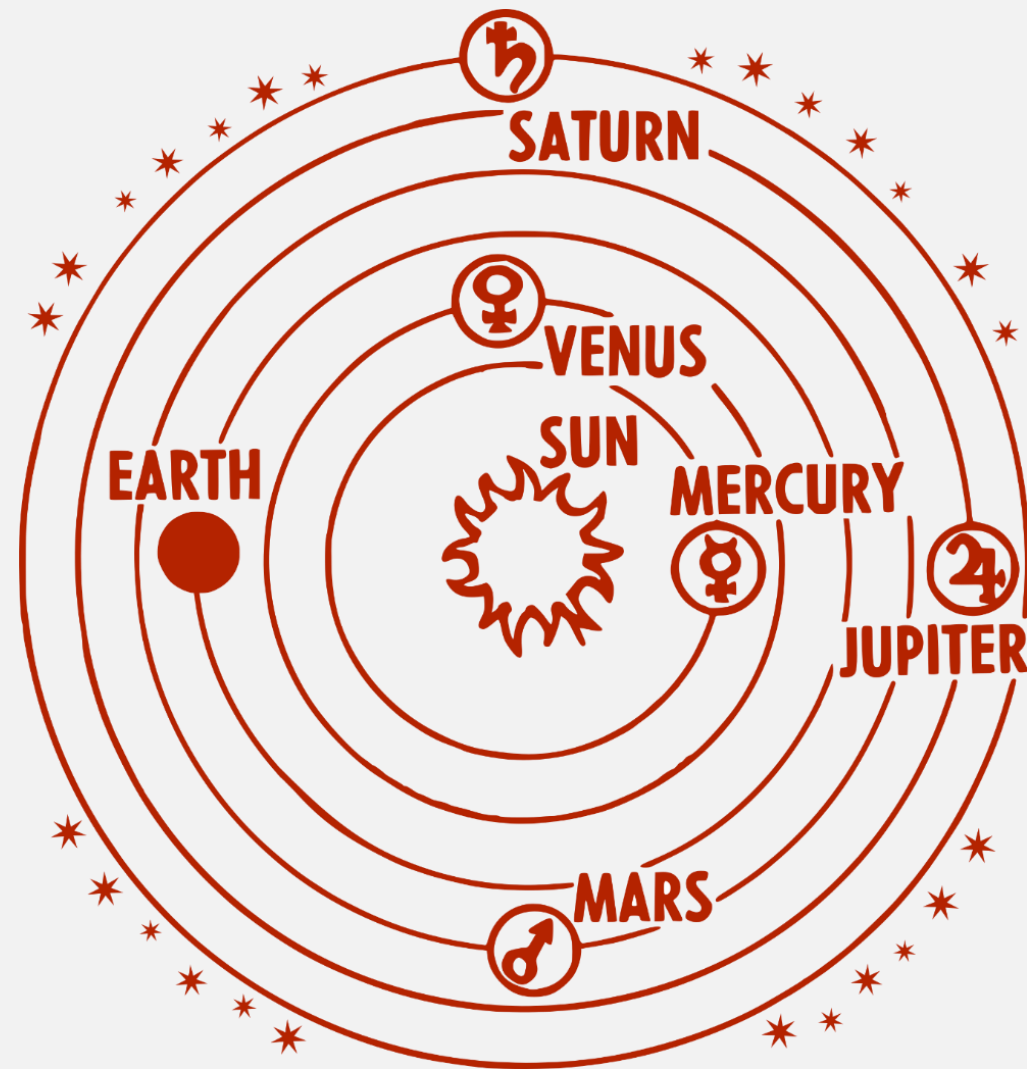
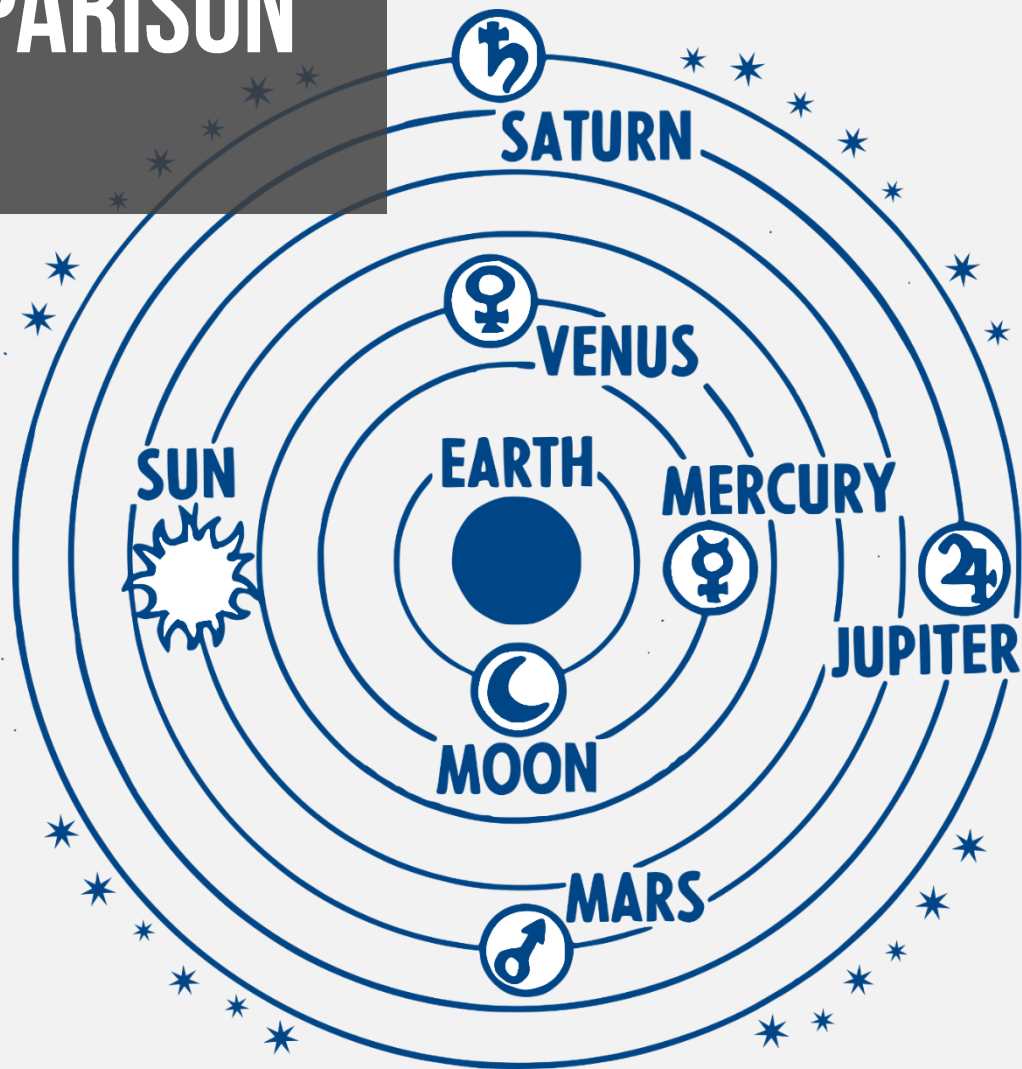
More on review comments

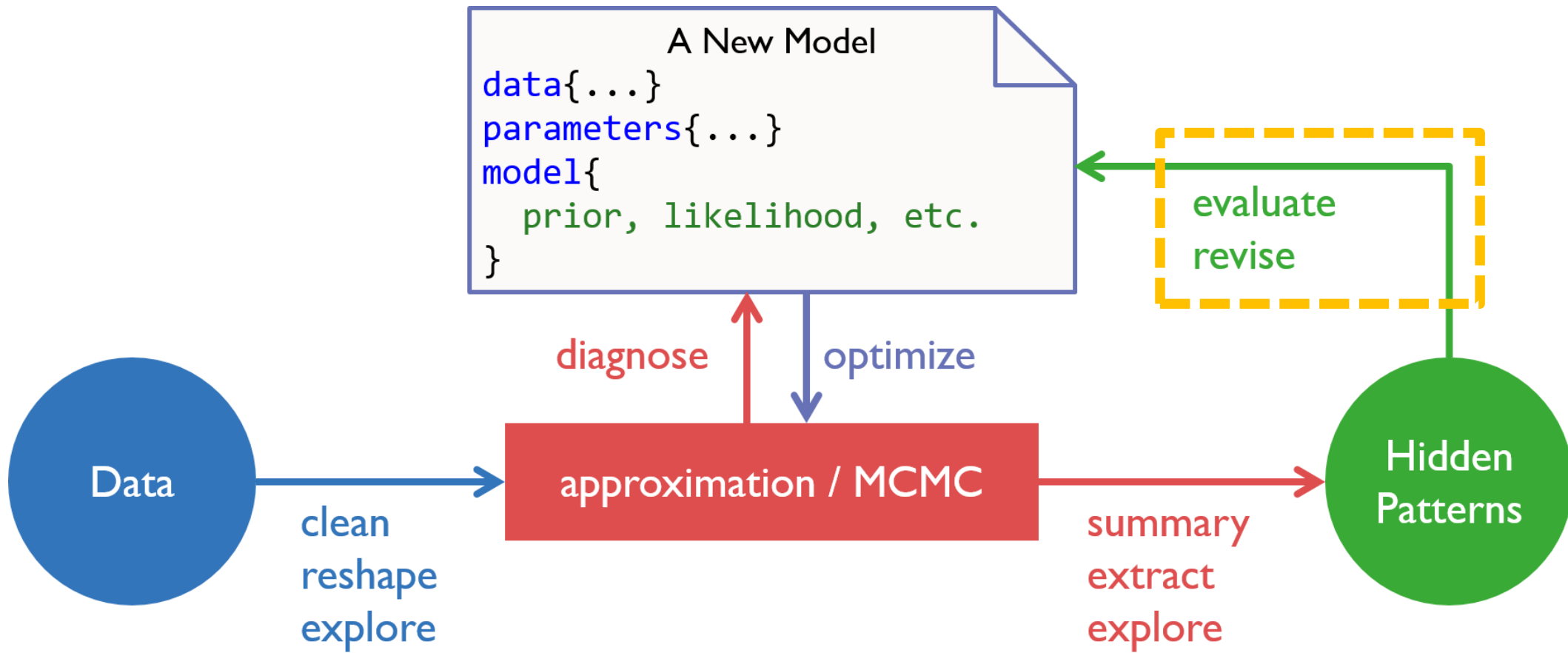
- Wording
 - you should do xx → the authors need to do xx
- When to add reference?
 - only for new citations
- Something okay
 - methods at the very end (depends on the journal they submitted)
 - student sample re: generalizability



constraint	reparameterization
$\theta \in (-\infty, +\infty)$	$\theta = \mu_\theta + \sigma_\theta \tilde{\theta}$
$\theta \in [0, N]$	$\theta = \text{Probit}^{-1}(\mu_\theta + \sigma_\theta \tilde{\theta}) \times N$
$\theta \in [M, N]$	$\theta = \text{Probit}^{-1}(\mu_\theta + \sigma_\theta \tilde{\theta}) \times (N-M) + M$
$\theta \in (0, +\infty)$	$\theta = \exp(\mu_\theta + \sigma_\theta \tilde{\theta})$

MODEL COMPARISON





Model Comparison

cognitive model

statistics

computing

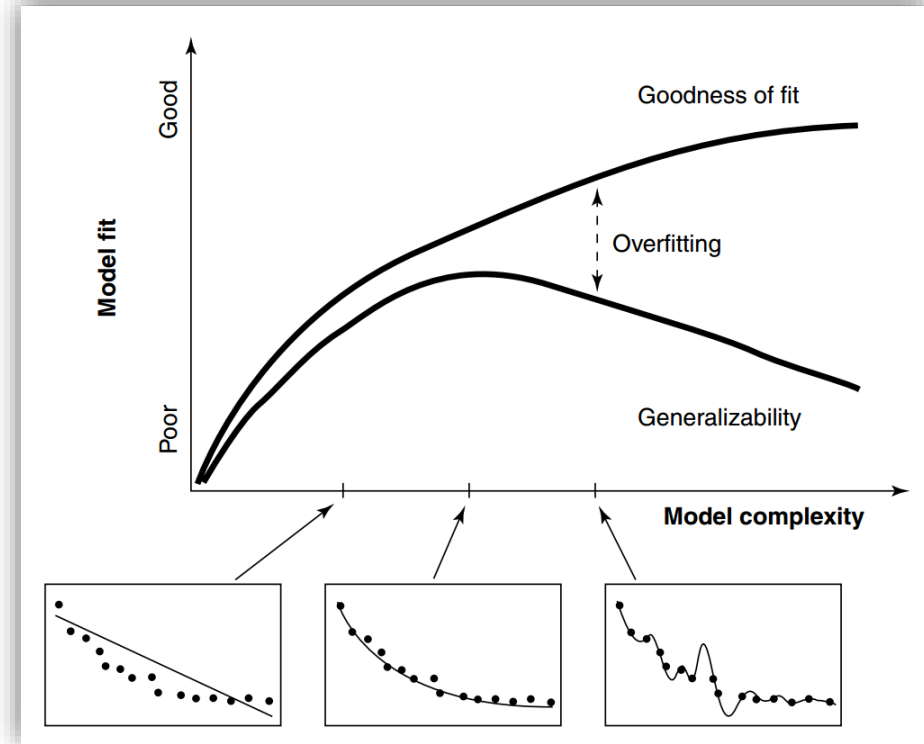
Which model provides the best **fit**?



Which model represents the best **balance** between model fit and model complexity?

Ockham's razor:

Models with fewer assumptions are to be preferred



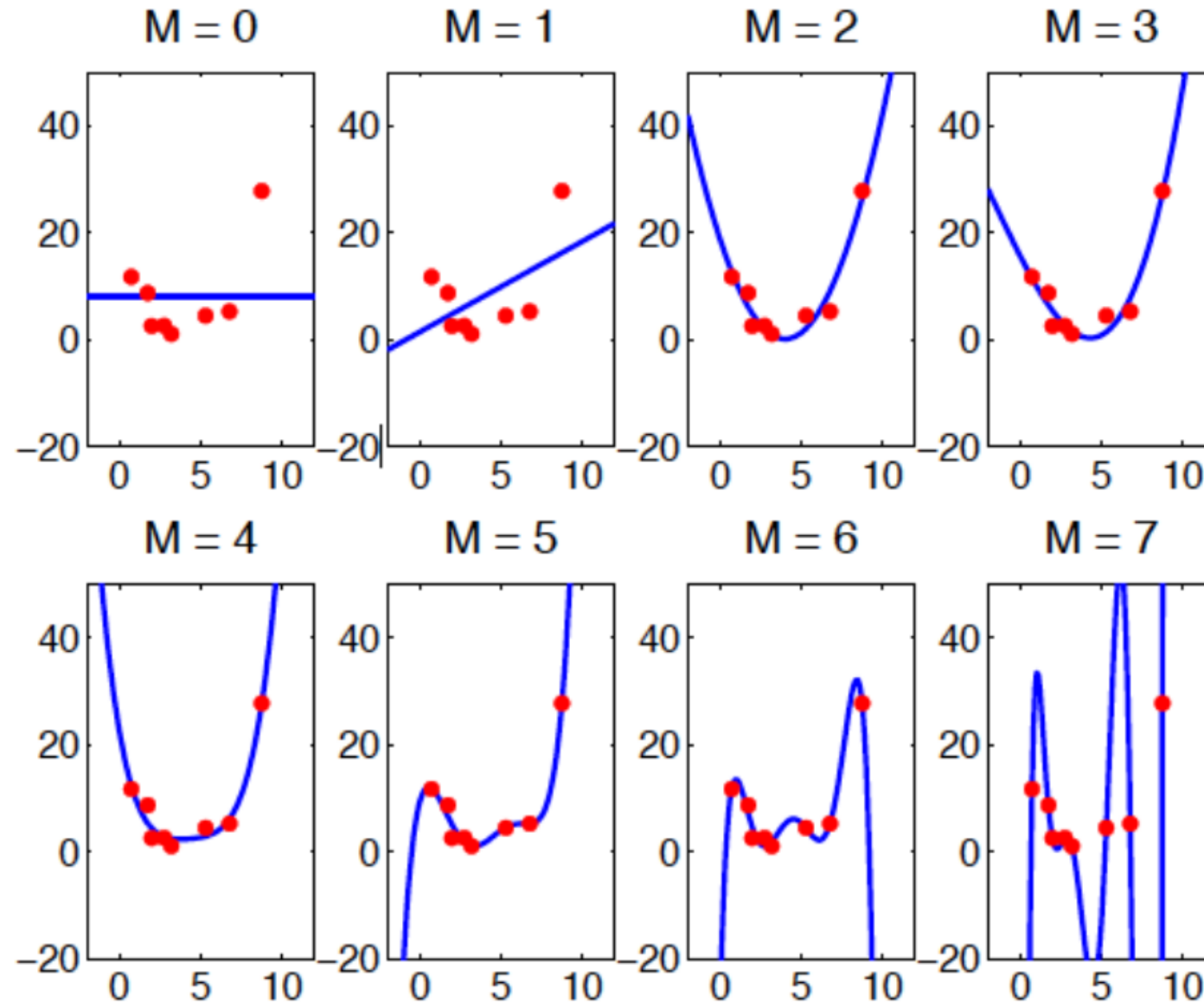
- overfitting: learn **too much** from the data
- underfitting: learn **too little** from the data

Which model has the highest predictive power?

cognitive model

statistics

computing

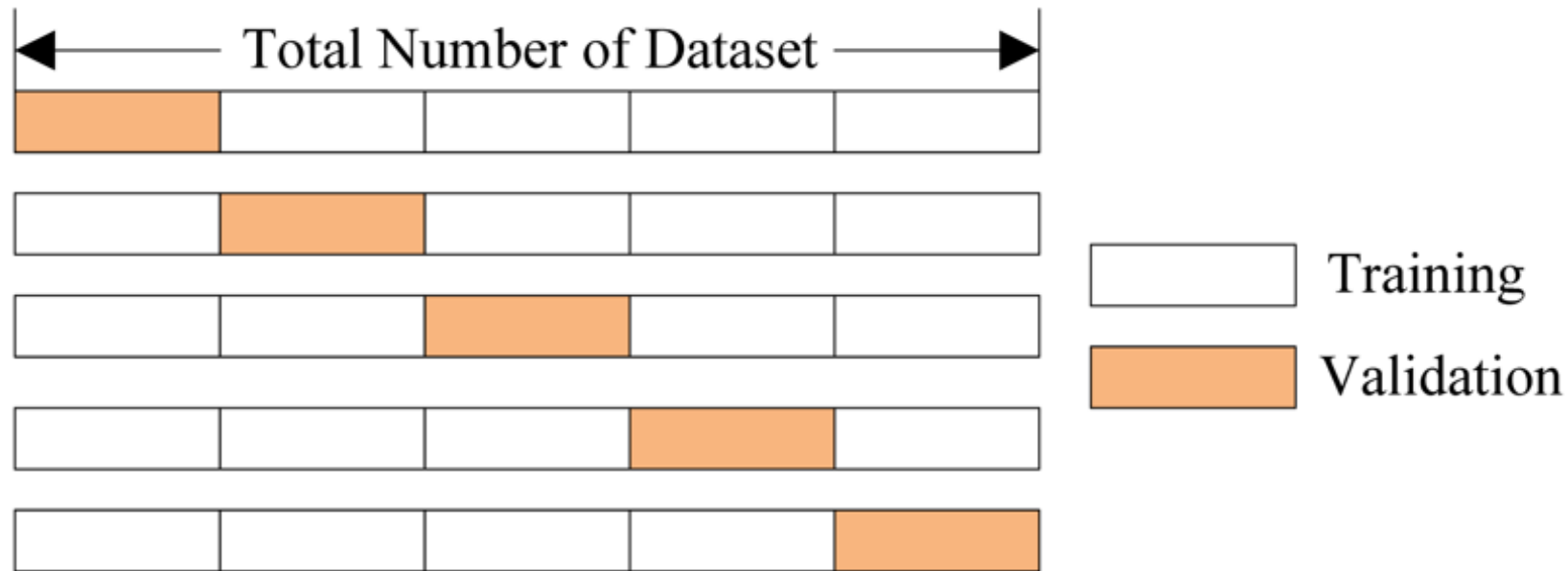


Focusing on Predictive Accuracy

cognitive model

statistics

computing



- Nothing prevents you from doing that in a Bayesian context but holding out data makes your posterior distribution more diffuse
- Bayesians usually condition on *all* the data and evaluate how well a model is expected to **predict out of sample** using "information criteria": model with the **highest expected log predictive density (ELPD)** for new data

Information Criteria

cognitive model

statistics

computing

AIC – Akaike information criterion

DIC – Deviance Information Criterion

WAIC – Widely Applicable Information Criterion

finding the model that has
the highest out-of-sample
predictive accuracy

approximation to LOO

BIC – Bayesian Information Criterion

finding the “true” model

Compute WAIC from Likelihood

cognitive model

statistics

computing

$$\text{WAIC} = -2 \widehat{\text{elpd}}_{\text{waic}}$$

expected log pointwise predictive density

$$\widehat{\text{elpd}}_{\text{waic}} = \widehat{\text{lpd}} - \widehat{p}_{\text{waic}}$$

$\widehat{\text{lpd}}$ = computed log pointwise predictive density

$$= \sum_{i=1}^n \log \left(\frac{1}{S} \sum_{s=1}^S p(y_i | \theta^s) \right).$$

```
lpd <- log(colMeans(exp(log_lik)))
```

estimated effective number of parameters

$$\widehat{p}_{\text{waic}} = \sum_{i=1}^n V_{s=1}^S (\log p(y_i | \theta^s))$$

```
p_waic <- colVars(log_lik)
```

*IC comparisons

cognitive model

statistics

computing

		No pooling ($\tau = \infty$)	Complete pooling ($\tau = 0$)	Hierarchical model (τ estimated)
AIC	$-2 \text{ lpd} = -2 \log p(y \hat{\theta}_{\text{mle}})$	54.6	59.4	
	k	8.0	1.0	
	$\text{AIC} = -2 \widehat{\text{elpd}}_{\text{AIC}}$	70.6	61.4	
DIC	$-2 \text{ lpd} = -2 \log p(y \hat{\theta}_{\text{Bayes}})$	54.6	59.4	57.4
	p_{DIC}	8.0	1.0	2.8
	$\text{DIC} = -2 \widehat{\text{elpd}}_{\text{DIC}}$	70.6	61.4	63.0
WAIC	$-2 \text{ lppd} = -2 \sum_i \log p_{\text{post}}(y_i)$	60.2	59.8	59.2
	$p_{\text{WAIC } 1}$	2.5	0.6	1.0
	$p_{\text{WAIC } 2}$	4.0	0.7	1.3
	$\text{WAIC} = -2 \widehat{\text{elppd}}_{\text{WAIC } 2}$	68.2	61.2	61.8
LOO-CV	-2 lppd		59.8	59.2
	$p_{\text{loo-cv}}$		0.5	1.8
	$-2 \text{ lppd}_{\text{loo-cv}}$		60.8	62.8

Recording the Log-Likelihood in Stan

cognitive model

statistics

computing

```
generated quantities {  
  ...  
  real log_lik[nSubjects];  
  ...  
  
  { # local section, this saves time and space  
    for (s in 1:nSubjects) {  
      vector[2] v;  
      real pe;  
  
      log_lik[s] = 0;  
      v = initV;  
  
      for (t in 1:nTrials) {  
        log_lik[s] = log_lik[s] + categorical_logit_lpmf(choice[s,t] | tau[s] * v);  
  
        pe = reward[s,t] - v[choice[s,t]];  
        v[choice[s,t]] = v[choice[s,t]] + lr[s] * pe;  
      }  
    }  
  }  
}
```

The {loo} Package

cognitive model

statistics

computing

```
> library(loo)
> LL1    <- extract_log_lik(stanfit)
> loo1   <- loo(LL1)    # PSIS leave-one-out
> waic1  <- waic(LL1)   # WAIC
```

Computed from 4000 by 20 log-likelihood matrix

	Estimate	SE
elpd_loo	-29.5	3.3
p_loo	2.7	1.0
looic	58.9	6.7

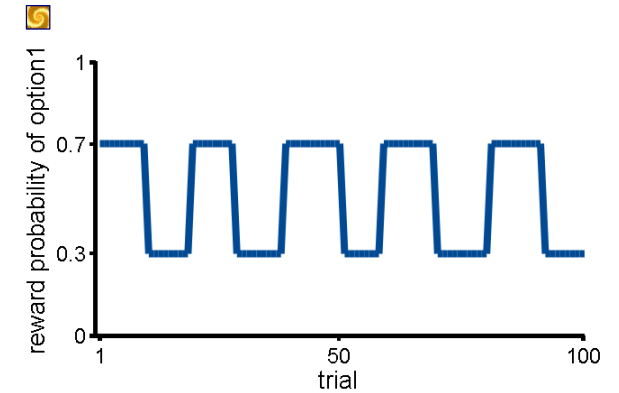
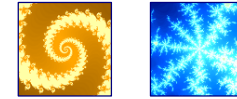
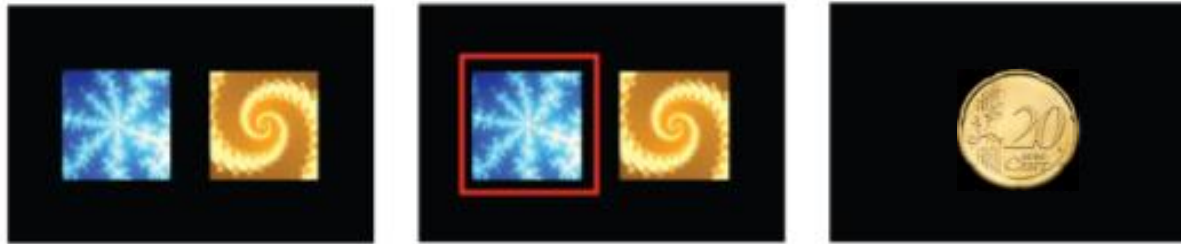
Pareto Smoothed Importance Sampling

Reversal Learning Task

cognitive model

statistics

computing



Fictitious RL
(Counterfactual RL)

Value update:

$$V_{t+1}^c = V_t^c + \alpha * PE$$

$$V_{t+1}^{nc} = V_t^{nc} + \alpha * PEnc$$

Prediction error:

$$PE = R_t - V_t^c$$

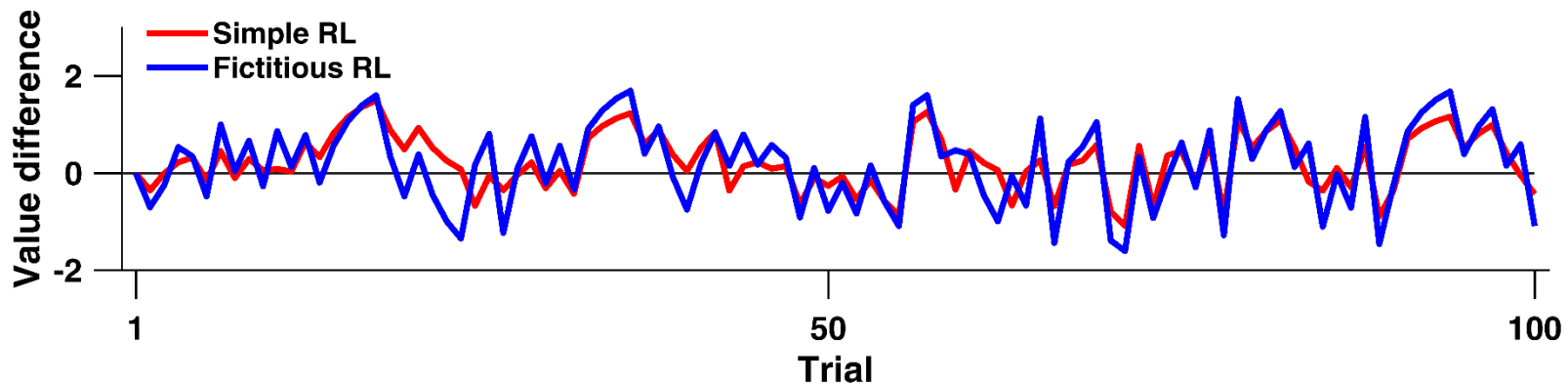
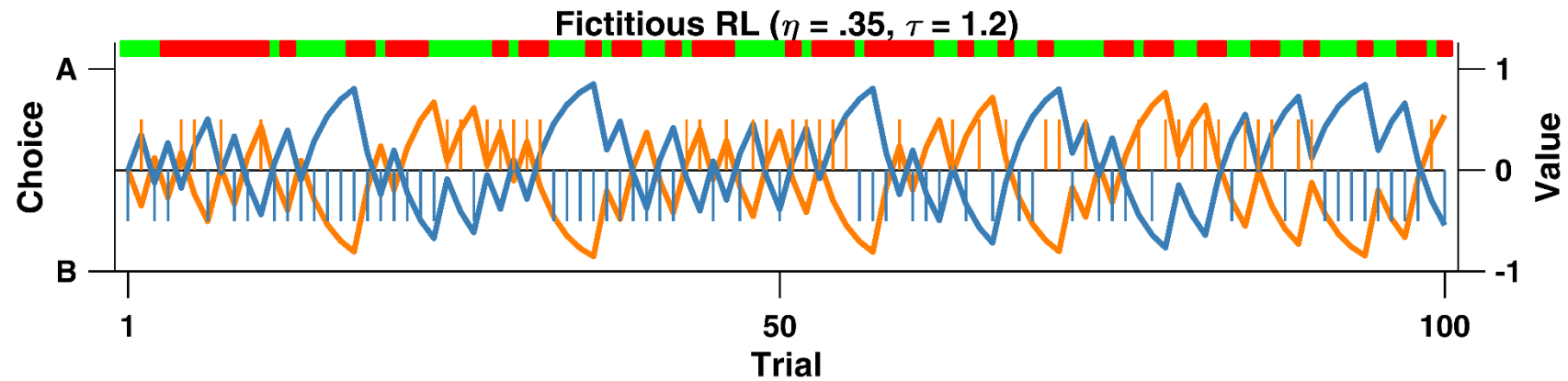
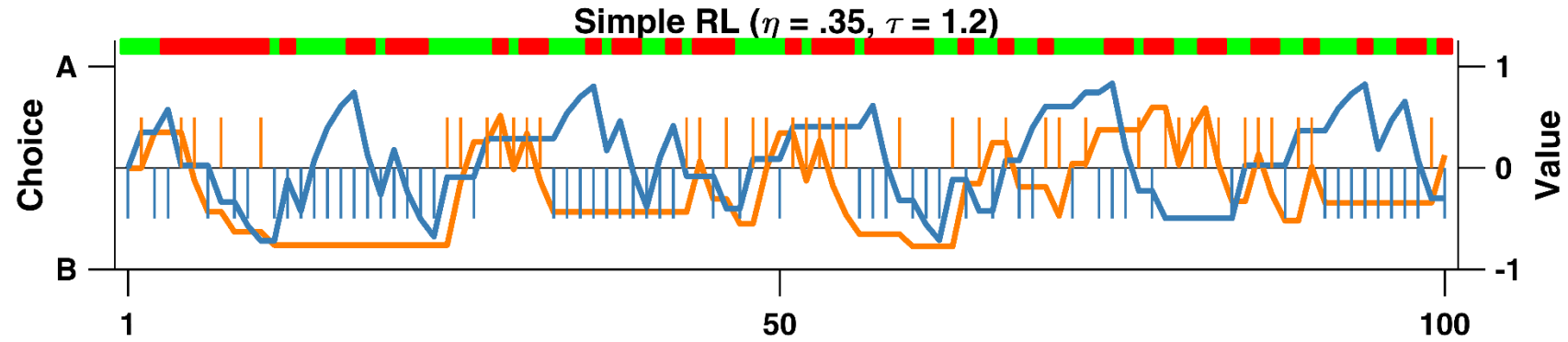
$$PEnc = -R_t - V_t^{nc}$$

More on Fictitious RL

cognitive model

statistics

computing



Exercise XIII

cognitive model

statistics

computing

```
.../08.compare_models/_scripts/compare_models_main.R
```

TASK: (1) complete the fictitious RL model (model2, loglik)
(2) fit and compare the 2 models

Exercise XIII – output

cognitive model

statistics

computing

```
> LL1 <- extract_log_lik(fit_rl1)
> ( loo1 <- loo(LL1) )
Computed from 4000 by 10 log-likelihood matrix
```

	Estimate	SE
elpd_loo	-389.8	15.4
p_loo	3.8	0.8
looic	779.5	30.9

```
> ( loo2 <- loo(LL2) )
Computed from 4000 by 10 log-likelihood matrix
```

	Estimate	SE
elpd_loo	-281.3	17.5
p_loo	3.4	0.5
looic	562.6	35.0

Posterior Predictive Check

cognitive model

statistics

computing

