Model Selection II

- Philosophy of science and multiple alternative models
- Trade-offs
- Likelihood-based metrics
  - Likelihood Ratio Test
  - AIC
- **Bayesian metrics**
  - **DIC**
  - **Predictive Loss**

Model selection

- Focus on choosing between multiple competing models rather than refuting a single null model
- How do we judge models?
  - Complexity
    - Number of parameters
  - Uncertainty
    - Model residuals
    - Parameter error (identifiability)
  - Data as ultimate arbiter
- “Make everything as simple as possible, but not simpler.” - A. Einstein

Likelihood Ratio Test

- $LR = L(x|\theta_{\circ}) / L(x|\theta_{\bullet})$
- $D = -2\ln L(x|\theta_{\circ}) - -2\ln L(x|\theta_{\bullet})$
- The test statistic D is known to be distributed with a χ^2 distribution
- Degrees of freedom = Difference in # of param.
  - Overall, L increases (-lnL declines) with # of param.
  - Penalizes model with more parameters
- p-val = 1-pchisq(D,df)

Akaike Information Criterion

$$AIC = -2\,lnL + 2\mathrm{p}$$

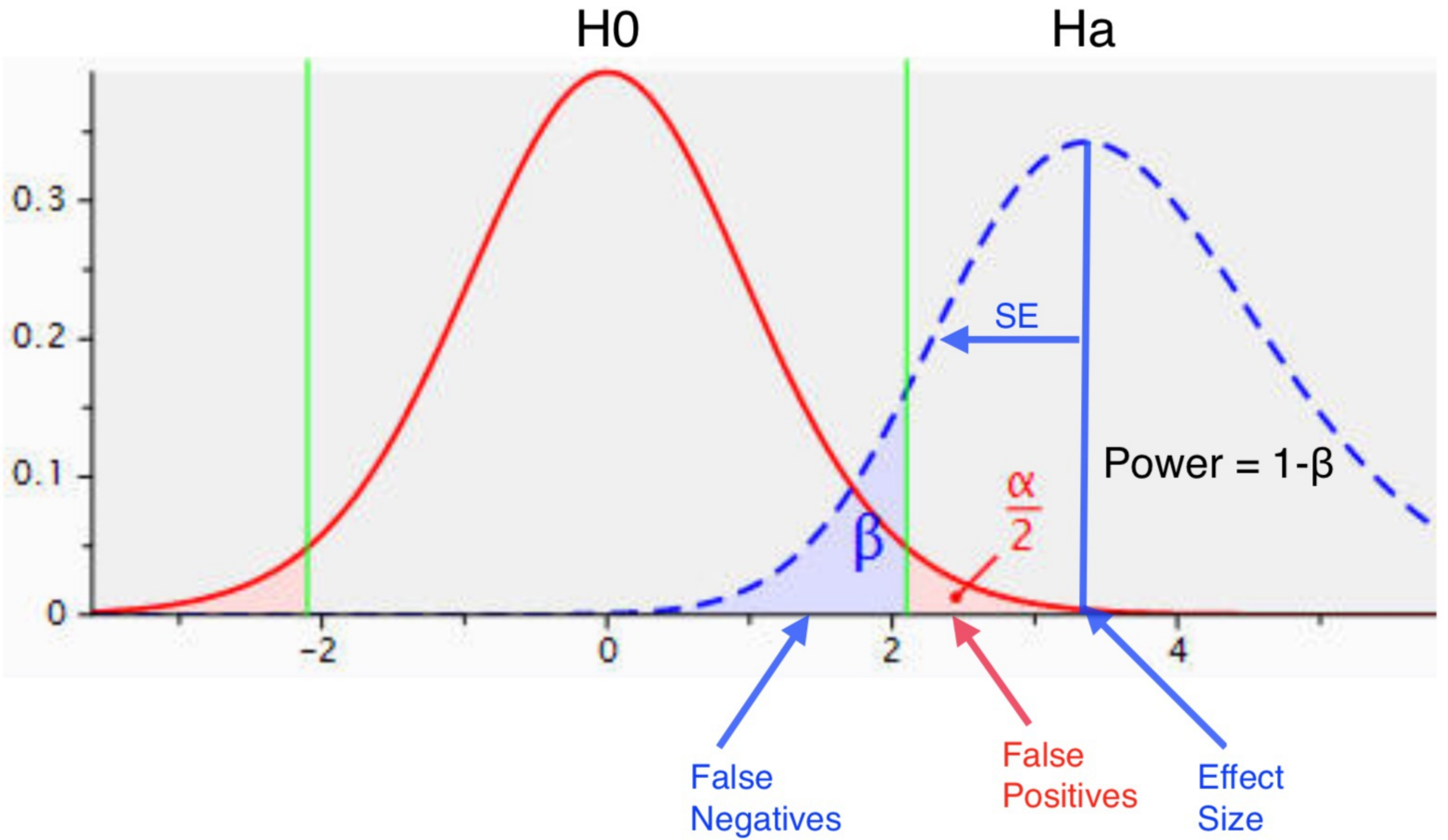
- p = number of parameters in the model
- Based on information theory
- Lowest value “wins”
- No p-value
- Often expressed relative to best model, ∞AIC
- “Rules of thumb”
  - 0-2 = similar   2-5 = weak support   >5 = strong

P-value

- Probability of obtaining a test statistic at least as extreme as the one that was actually observed, assuming that the null hypothesis is true.
- **Not** the probability that the null hypothesis is true
  - P-value can be close to zero when the posterior probability of the null is close to 1
- **Not** the probability of falsely rejecting the null hypothesis
- **Not** biological significance

Power

- Probability of correctly rejecting the null hypothesis
- Requires that some explicit alternative hypothesis is stated
  - Parameter values
  - Variance
  - Sample size
- Often calculated as a function of sample size
- For complex models, calculate through simulation

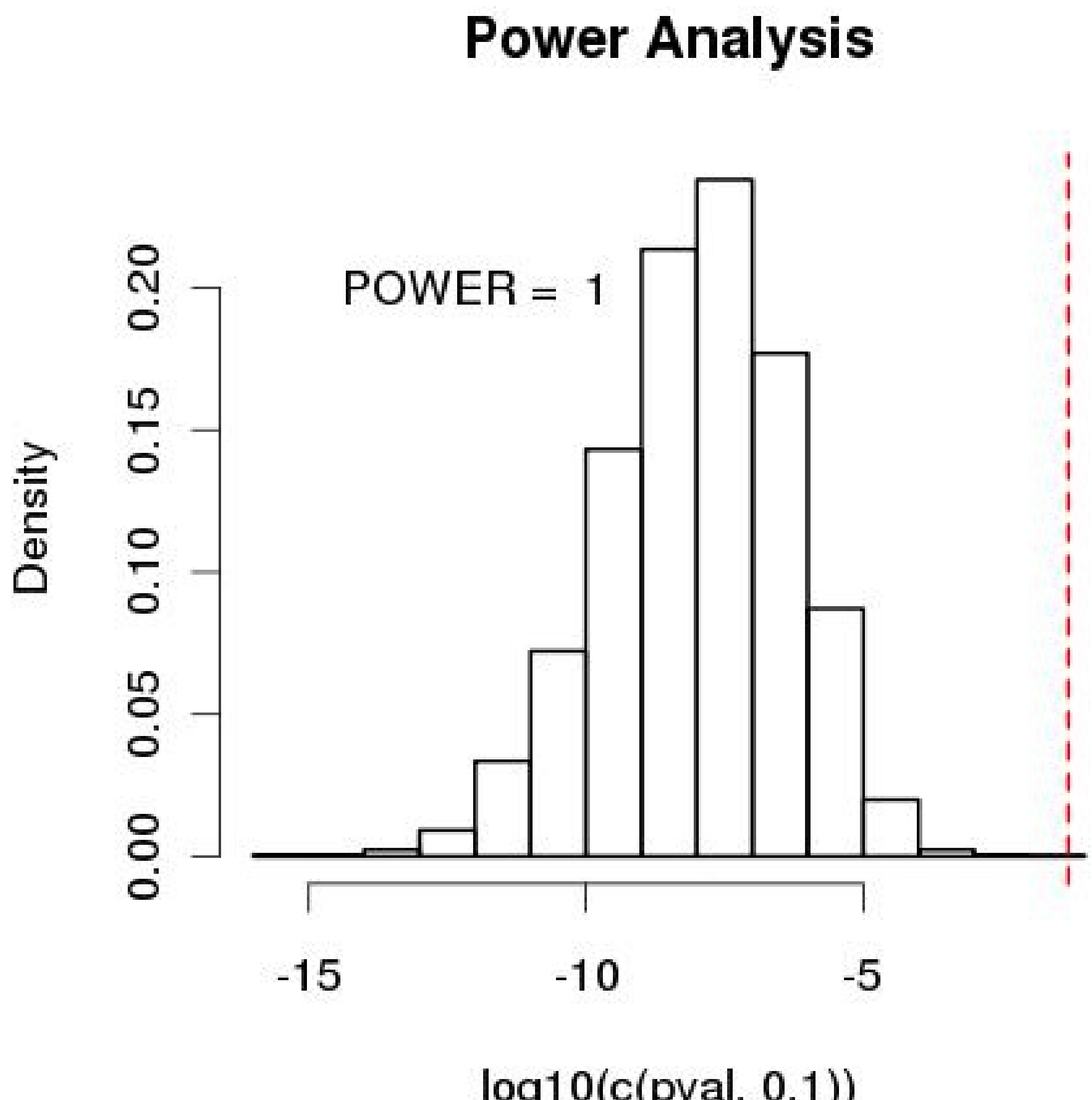


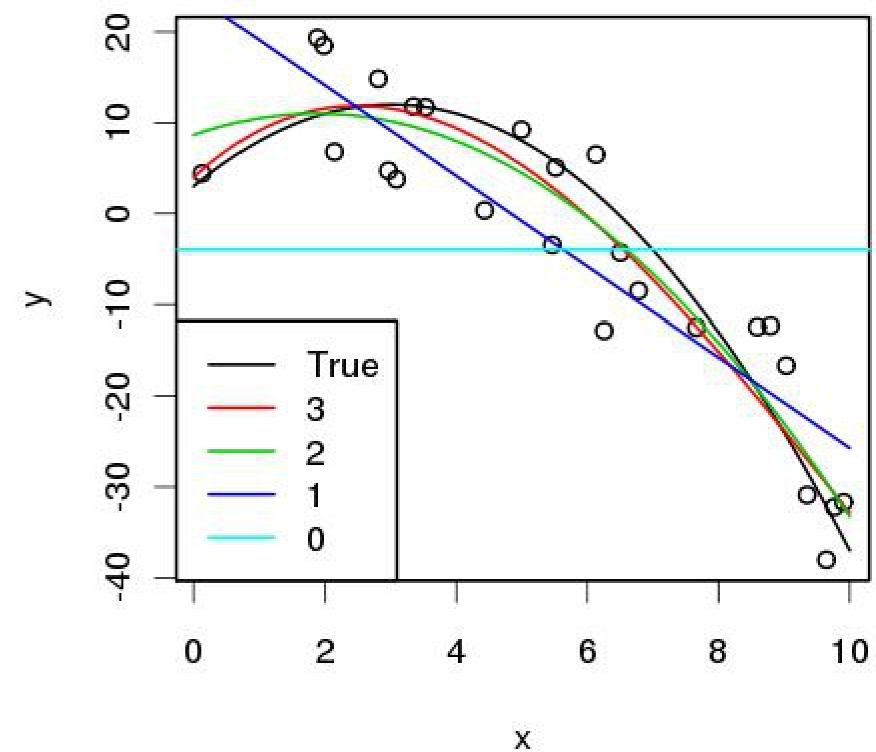
Power = f(effect size, SE)

Generic Example

```
LnL.A = function(theta){
    -sum(dnorm(y,f(x,theta),sd)))
}
lnL.0 = function(mu){
    -sum(dnorm(y,mu,sd))
}
for(i in 1:nsim){
    Ey = f(x,theta)             ## process model
    y = rnorm(N,Ey,sd)      ## data model
    outA = optim(ic,lnL.A) ##fit of alternative
    out0 = optim(ic,lnL.0)  ##fit of null
    pval[i] = 1-pchisq(2*(outA$value-out0$value),df)
}
power = sum(pval < 0.05)/nsim
```

Example: Quadratic vs Linear LRT





- Results **specific** to parameter values and sample size chosen

Deviance Information Criterion

$$DIC = \bar{D} + p_D$$

$$\bar{D} = E[D(\theta)] = \frac{1}{n_g}\sum D_i \qquad\qquad p_D = \bar{D} - D(\bar{\theta})$$

- p_D = effective number of parameters
- Easily calculated from MCMC
- Averages over parameter distribution rather than just single maximum
- Applicable when the number of parameters is ambiguous
- Lowest score “wins”

Hierarchical Models

Common mean

μ

$Y_1 \quad Y_2 \quad Y_3$

$p=2$

Hierarchical

μ

$\theta_1 \quad \theta_2 \quad \theta_3$

$Y_1 \quad Y_2 \quad Y_3$

$2 < p < 8$

Independent

$\mu \quad \mu \quad \mu$

$Y_1 \quad Y_2 \quad Y_3$

$p=6$

DIC computation

```
dic.samples(model, n.iter, ...)
```

- For each MCMC iteration
  - Calculate and store deviance: $D(\theta_i) = -2\ln L(y|\theta_i)$
- After MCMC
  - Calculate posterior means for parameters $\bar{\theta}$
  - Calculate D at $\bar{\theta}$: $D(\bar{\theta}) = -2\ln L(y|\bar{\theta})$
  - Calculate $\overline{D(\theta)} = \sum D(\theta_i)/n_g$
- $DIC = 2\overline{D(\theta)} - D(\bar{\theta})$

Model	DIC	pD	ΔDIC
flat	221.10	2.06	82.00
linear	174.40	3.12	35.30
quadratic	139.10	4.15	0.00
cubic	141.40	5.27	2.30

Watanabe-Akaike (WAIC)

$$\text{WAIC} = -2\sum_{i=1}^{n} \log \underbrace{\int [y_i \mid \boldsymbol{\theta}][\boldsymbol{\theta} \mid \mathbf{y}]d\boldsymbol{\theta}}_{\text{Posterior Predictive Distribution}} + 2p_{\text{D},2}$$

$$p_{\text{D},2} = \sum_{i=1}^{n} \text{var}_{\boldsymbol{\theta} \mid \mathbf{y}}(\log \underbrace{[y_i \mid \boldsymbol{\theta}]}_{\mathbf{L}})$$

- Fully Bayesian
- Both elements in sum approximated using MCMC samples

```
model  <- jags.model(mod,data = data,
                     n.chains=chains,quiet=TRUE)

samps  <- coda.samples(model,
                     variable.names=c("like"),
                     n.iter=iters, progress.bar="none")

# Compute DIC

dic    <- dic.samples(model,n.iter=iters)

DIC  <- sum(dic$dev)+sum(dic$pen)

# Compute WAIC

like          <- rbind(samps[[1]],samps[[2]])
  # Combine samples from the two chains

fbar          <- colMeans(like)

Pw            <- sum(apply(log(like),2,var))

WAIC <- -2*sum(log(fbar))+2*Pw
```

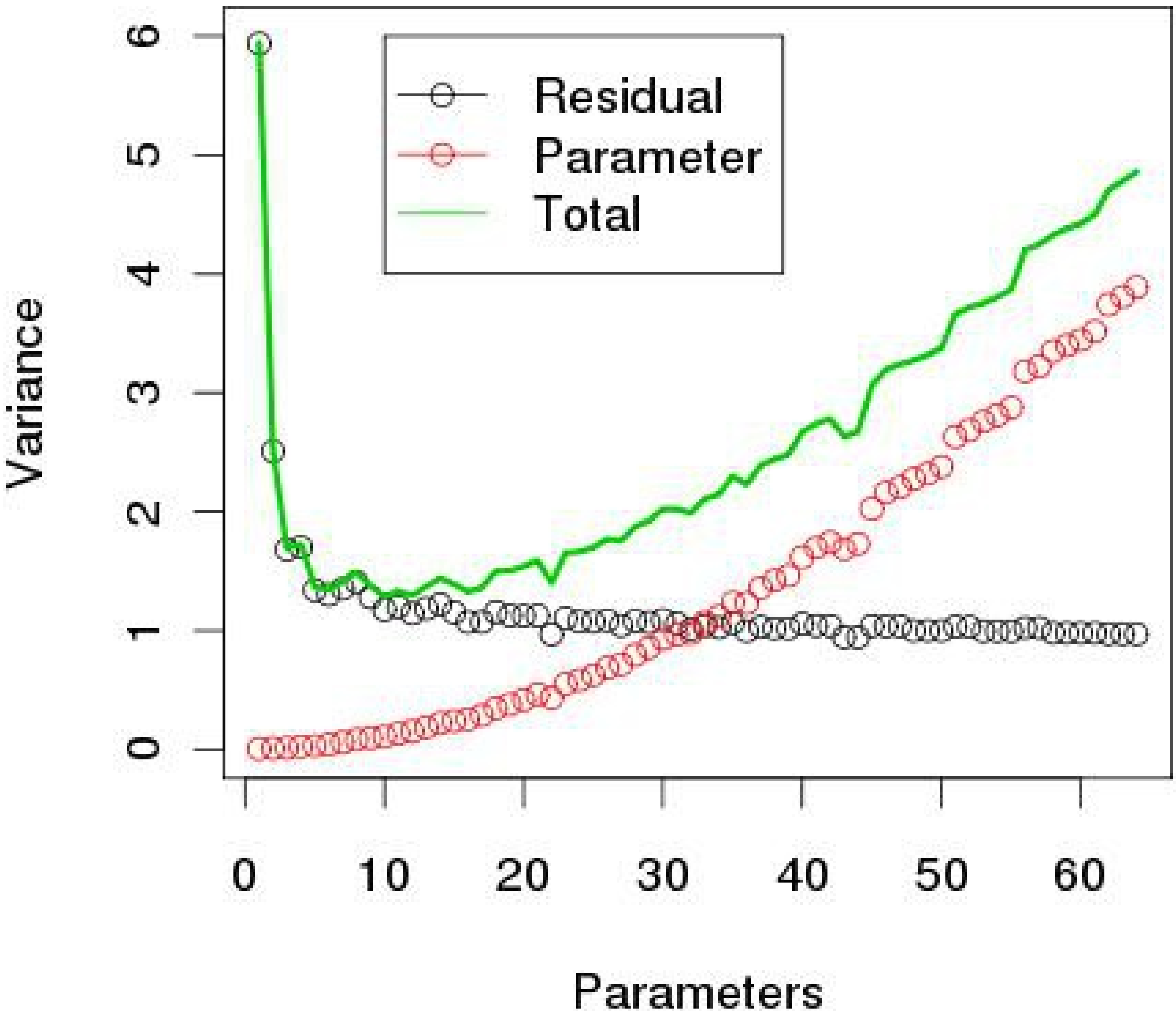
```
 # simple logistic regression model
for(i in 1:n){
    Y[i]          ~ dbern(pi[i])
    logit(pi[i]) <- beta[1]+ X[i]*beta[2]
    like[i]      <- dbin(Y[i],pi[i],1)
     # For WAIC computation
  }
for(j in 1:2){beta[j] ~ dnorm(0,0.01)}
 }
```

Predictive Loss

$$D_{pl} = G + P$$

- G = total residual SS $\sum (E[y_{rep}] - y_{obs})^2$
- P = total predictive variance $\sum var[y_{rep}]$
- Given y_{obs}, predict replicate y_{rep}
  - i.e. predictions made for same points as observations
- Focused on prediction, easily calc from MCMC
- Does not require model dimension

UNCERTAINTY



P/n

G/n

Predictive Loss Algorithm

- For every MCMC step
  - Generate pseudodata <u>at same points/covariates</u> as the original data (otherwise equiv. PI calc.)
- From posterior predictive distribution
  - Calculate posterior mean for each point: $E[y_{rep}]$
  - Calculate residual variance for each point: $Var[y_{rep}]$
  - P = sum of $Var[y_{rep}]$ over all points
  - G = $\sum \left(E[y_{rep}] - y_{obs}\right)^2$

Predictive Loss: Quadratic

model	P	G	D
flat	10065.16	8596.29	18661.45
linear	1546.03	1215.26	2761.3
quadratic	378.68	272.7	651.38
cubic	410.45	271.3	681.74

Note: sqrt of P/n and G/n are the predictive SD and residual SD respectively

Bayes Factor

$$BF = \frac{p(M_1|y)/p(M_2|y)}{p(M_1)/p(M_2)}$$

- Require assigning a prior probability to each model
- Hard to calculate except in limited cases
- Asymptotically tends to select too simple
- I have not seen BF used much recently

Reversible Jump MCMC

- Considers the number of terms in a nested model to be unknown
- Will add and remove terms within the MCMC step
- Generates a posterior probability for each model
- Prediction automatically averages over models
- “in fashion”

Bayesian Model Averaging

- Make predictions using all of your alternative models

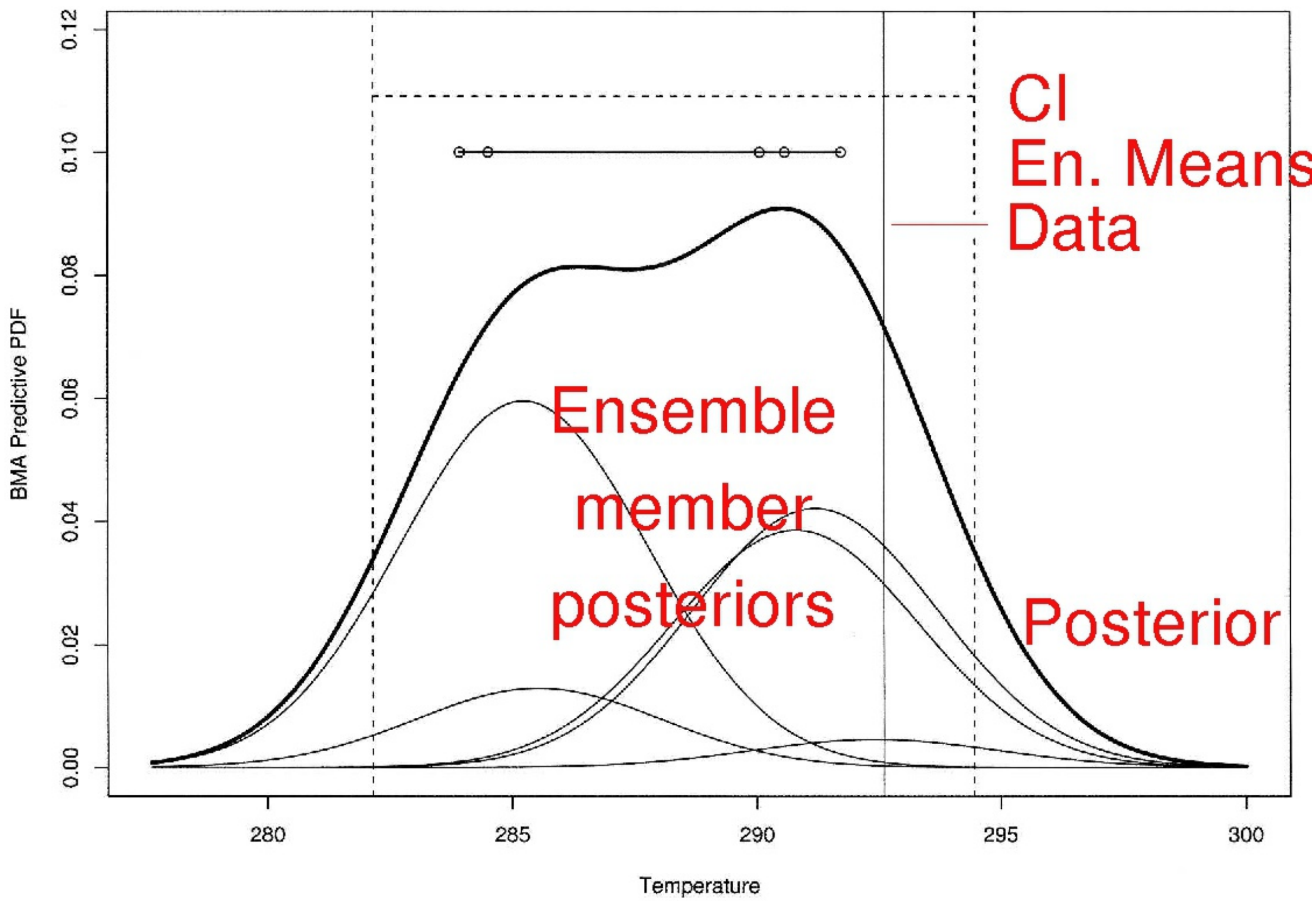
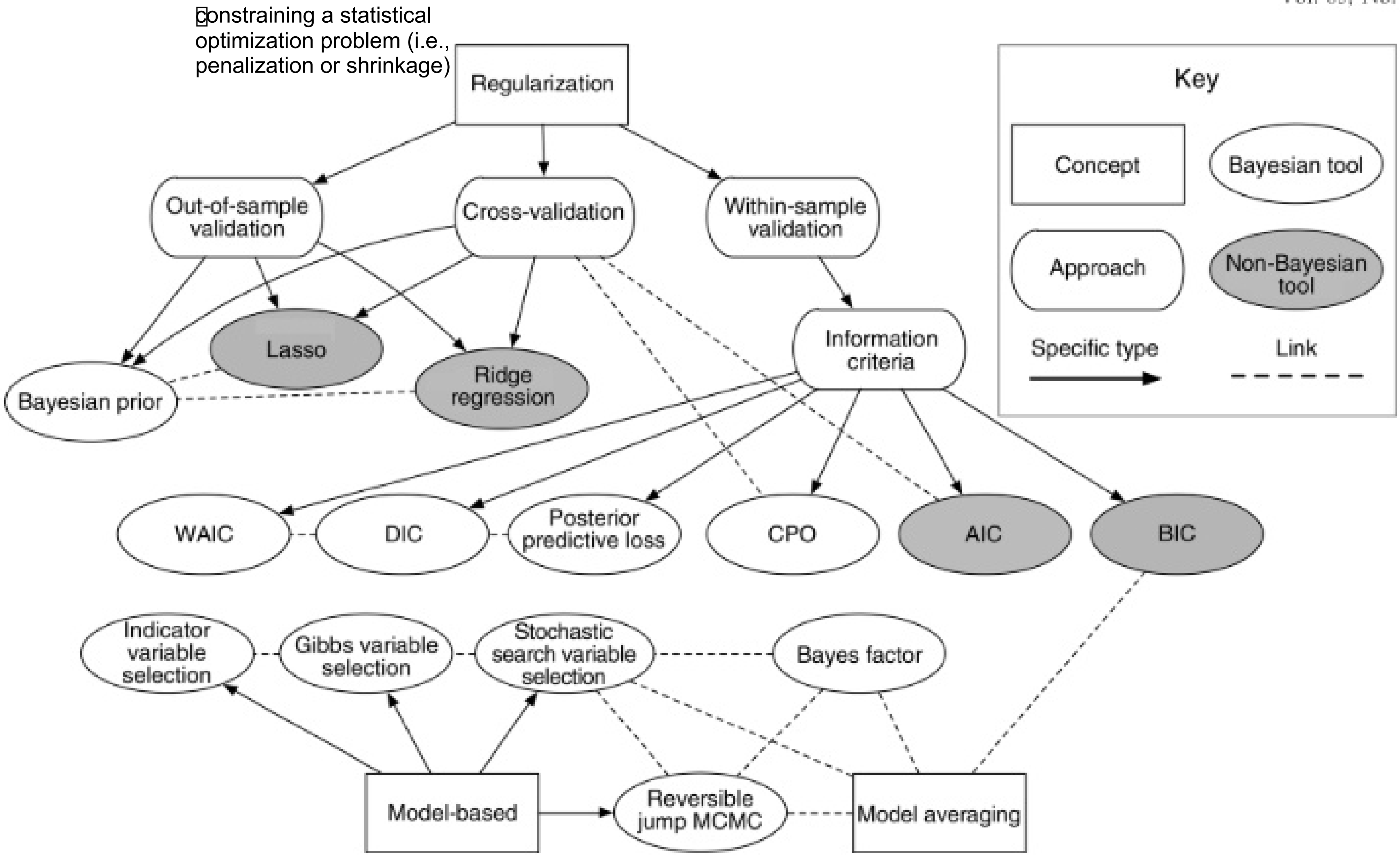


FIG. 3. BMA predictive PDF (thick curve) and its five components (thin curves) for the 48-h surface temperature forecast at Packwood, WA, initialized at 0000 UTC on 12 Jun 2000. Also shown are the ensemble member forecasts and range (solid horizontal line and bullets), the BMA 90% prediction interval (dotted lines), and the verifying observation (solid vertical line).


constraining a statistical
optimization problem (i.e.,
penalization or shrinkage)
Regularization
Key
Concept
Bayesian tool
Approach
Non-Bayesian tool
Specific type
Link
Out-of-sample validation
Cross-validation
Within-sample validation
Lasso
Ridge regression
Bayesian prior
Information criteria
WAIC
DIC
Posterior predictive loss
CPO
AIC
BIC
Indicator variable selection
Gibbs variable selection
Stochastic search variable selection
Bayes factor
Model-based
Reversible jump MCMC
Model averaging

As of now we can...

- Fit models (Likelihood and Bayes)
- Construct Confidence intervals
- Test Hypotheses / compare models
- Make predictions that propagate uncertainty

What's left?

- Exploration of common and more advance models
  - Useful approaches/models for certain types of problems