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

- \( LR = \frac{L(x|\theta_1)}{L(x|\theta_0)} \)
- \( D = -2\ln L(x|\theta_1) - -2\ln L(x|\theta_0) \)
- The test statistic \( D \) is known to be distributed with a \( \chi^2 \) distribution
- Degrees of freedom = Difference in \# of param.
  - Overall, \( L \) increases (-\( \ln L \) declines) with \# of param.
  - Penalizes model with more parameters
- \( p\text{-val} = 1 - p\text{chisq}(D, df) \)
Akaike Information Criterion

\[ AIC = -2 \ln L + 2p \]

- \( p \) = number of parameters in the model
- Based on information theory
- Lowest value “wins”
- No \( p \)-value
- Often expressed relative to best model, \( \infty \text{AIC} \)
- “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)
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

\[ DIC = \bar{D} + p_D \]

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

- \( p_D \) = 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

Hierarchical

Independent

\[ \mu \]

\[ \theta_1 \theta_2 \theta_3 \]

\[ Y_1 Y_2 Y_3 \]

\[ p=2 \]

\[ 2 < p < 8 \]

\[ p=6 \]

p=2
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 \( \bar{D}(\theta) = \sum D(\theta_i)/n_g \)

• \( DIC = 2\bar{D}(\theta) - D(\bar{\theta}) \)
<table>
<thead>
<tr>
<th>Model</th>
<th>DIC</th>
<th>pD</th>
<th>ΔDIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>flat</td>
<td>221.10</td>
<td>2.06</td>
<td>82.00</td>
</tr>
<tr>
<td>linear</td>
<td>174.40</td>
<td>3.12</td>
<td>35.30</td>
</tr>
<tr>
<td>quadratic</td>
<td>139.10</td>
<td>4.15</td>
<td>0.00</td>
</tr>
<tr>
<td>cubic</td>
<td>141.40</td>
<td>5.27</td>
<td>2.30</td>
</tr>
</tbody>
</table>
Watanabe-Akaike (WAIC)

\[
\text{WAIC} = -2 \sum_{i=1}^{n} \log \int [y_i | \theta] [\theta | y] d\theta + 2p_{D,2}
\]

\[
p_{D,2} = \sum_{i=1}^{n} \text{var}_{\theta | y} (\log[y_i | \theta])
\]

- 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])
   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
- \( P = \) total predictive variance
- 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
Predictive Loss Algorithm

• For every MCMC step
  – Generate pseudodata at same points/covariates 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: $\text{Var}[y_{rep}]$
  – $P = \text{sum of } \text{Var}[y_{rep}] \text{ over all points}$
  – $G = \sum (E[y_{rep}] - y_{obs})^2$
## Predictive Loss: Quadratic

<table>
<thead>
<tr>
<th>model</th>
<th>P</th>
<th>G</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>flat</td>
<td>10065.16</td>
<td>8596.29</td>
<td>18661.45</td>
</tr>
<tr>
<td>linear</td>
<td>1546.03</td>
<td>1215.26</td>
<td>2761.3</td>
</tr>
<tr>
<td>quadratic</td>
<td>378.68</td>
<td>272.7</td>
<td>651.38</td>
</tr>
<tr>
<td>cubic</td>
<td>410.45</td>
<td>271.3</td>
<td>681.74</td>
</tr>
</tbody>
</table>

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
constraining a statistical optimization problem (i.e., penalization or shrinkage)
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