Model Comparison

- PPCs for checking model fit. But how do we compare different models (regardless of fit)?
- Compare models by their out-of-sample predictive power
- Leave-one-out cross validation
  - Fit model n times, leaving out one observation
  - Evalute predcitive accuracy on left out observations
  - Typically used for parameter tuning
- Can use ideas from LOO-CV to compare models

Measures of Predictive Accuracy

- Point prediction: predict a single unknown future observation
- Measures of predictive accuracy are called *scoring functions* (e.g. MSE)
- Probabilistic predictions account for uncertainty
  - **Scoring rules**: measure of accuracy based on probabilistic predictions
  - Examples: quadratic, logarithmic and zero-one scores

Log-likelihood as a scoring rule

- As a measure of accuracy, we call it the "log predictive density"
- $\log(p(y|\theta))$ is proportional to MSE if the data are normal with constant variance

Chosen Model

- Connection to Kullback-Leibler divergence
  - KL: $-E_p log(q(y)/p(y)) = -E_p log(q(y)) + E_p log(p(y))$
  - Asymptotically, the lowest KL model is the one with the highest expected log predictive density

Fwd-KL

True Model
(unknown)

Out-of-sample prediction

Ideal: Predict fit on new data.

$$\log p_{post}(\tilde{y}) = \log E_{post}\, p(\tilde{y} \mid \theta) = \int p(\tilde{y} \mid \theta)\, p(\theta \mid y)\, d\theta$$

Expected Log Pointwise Predictive Density

(ELPPD) (one obs)

$$E_f \log p_{post}(\tilde{y}) = \int \log p_{post}(\tilde{y}) f(\tilde{y}) d\tilde{y}$$

True model $f(\tilde{y})$

ELPD: $\sum E_f \log p_{post}(\tilde{y})$

$$\widehat{ELPD}_{LOO} : \sum_{i=1}^{n} \log p(y_i \mid y_{-i})$$

$$p(y_i \mid y_{-i}) = \int p(y_i \mid \theta)\, p(\theta \mid y_{-i})\, d\theta$$

problem: Computation

Need: $p(\theta \mid y_{-i}) \quad \forall_i$

Have: $p(\theta \mid y)$

Importance Sampling

$$p(y_i \mid y_{-i}) = \int p(y_i \mid \theta)\, p(\theta \mid y_{-i})\, d\theta =$$

I.W.

$$\int p(y_i|\theta)\left(\frac{p(\theta|y_{-i})}{p(\theta|y)}\right)p(\theta|y)\,d\theta$$

When $y_1, \dots, y_n$ are iid.

$$\left(\frac{p(\theta|y_{-i})}{p(\theta|y)}\right) = \frac{1}{p(y_i|\theta)}$$

$$w_{is} = \frac{1}{p(y_i|\theta^s)}$$

LOO Importance Sampling

- When the data are conditionally independent, $r_i^s = \frac{1}{p\left(y_i|\theta^s\right)} \propto \frac{p\left(\theta^s|y_{-i}\right)}{p(\theta^s|y)}$ to get importance sampling estimates
- $p\left(\tilde{y}_i \mid y_{-i}\right) \approx \frac{\sum_{s=1}^{S} r_i^s p\left(\tilde{y}_i|\theta^s\right)}{\sum_{s=1}^{S} r_i^s}$
- $p\left(y_i \mid y_{-i}\right) \approx \frac{1}{\frac{1}{S}\sum_{s=1}^{S} \frac{1}{p\left(y_i|\theta^s\right)}}$

https://link.springer.com/article/10.1007/s11222-016-9696-4

Pareto Smoothed Importance Sampling

1. Fit the generalized Pareto distribution to the 20% largest importance ratios r_s
2. Replacing the 20% largest ratios by the expected values of the order statistics of the fitted generalized Pareto distribution
3. Truncate the weights to ensure finite variance (see paper)

The above steps must be performed for each data point i.

$$\widehat{\text{elpd}}_{\text{psis-loo}} = \sum_{i=1}^{n} \log\left(\frac{\sum_{s=1}^{S} w_i^s p\left(y_i|\theta^s\right)}{\sum_{s=1}^{S} w_i^s} \right)$$

```r
library(loo)
nb_model <- cmdstan_model("nb_model.stan")
nb_fit = nb_model$sample(
               data=list(n1=num_bachelors, n2=num_no_bachelors,
               y1=bachelors_data, y2=no_bachelors_data),
               refresh=0)
```

```
Running MCMC with 4 parallel chains...

Chain 1 finished in 0.4 seconds.
Chain 2 finished in 0.4 seconds.
Chain 3 finished in 0.4 seconds.
Chain 4 finished in 0.4 seconds.

All 4 chains finished successfully.
Mean chain execution time: 0.4 seconds.
Total execution time: 0.5 seconds.
```

```r
loo_compare(list("pois"=pois_fit$loo(),
            "zip"=zip_fit$loo(),
            "nb"=nb_fit$loo()))
```

```
     elpd_diff se_diff
zip   0.0       0.0
nb   -3.2       2.4
pois -8.6       4.6
```

Galaxies Example

- Velocities in km/sec of 82 galaxies (Corona Borealis region).
- Multimodality in such surveys is evidence for voids and superclusters in the far universe.
- Statistical question: how many clusters are there in this dataset?

Model Comparison

```r
library("loo")
galaxy_speeds <- log(MASS::galaxies)
galaxy_results <- list()

## Run model for each of k clusters
mix_model <- cmdstanr::cmdstan_model("mix_model.stan")
for(k in 1:5) {
  galaxy_results[[k]] <-
    mix_model$sample(
      data=list(K=k, N=length(galaxy_speeds), y=galaxy_speeds),
      refresh=0, show_messages=FALSE)
}
```

```
Warning: 63 of 4000 (2.0%) transitions ended with a divergence.
See https://mc-stan.org/misc/warnings for details.
```

```
Warning: 16 of 4000 (0.0%) transitions hit the maximum treedepth limit of 10.
See https://mc-stan.org/misc/warnings for details.
```

Model Comparison

```r
loo_compare(galaxy_results[[1]]$loo(),
            galaxy_results[[2]]$loo(),
            galaxy_results[[3]]$loo(),
            galaxy_results[[4]]$loo(),
            galaxy_results[[5]]$loo())
```

```
       elpd_diff se_diff
model3   0.0       0.0  
model5  -0.4       1.4  
model4  -1.0       0.7  
model2 -10.7       3.7  
model1 -41.1       8.7  
```

Model Comparison

```r
galaxy_results[[4]]$loo()
```

```
Computed from 4000 by 82 log-likelihood matrix

         Estimate   SE
elpd_loo     32.0  9.5
p_loo         9.2  1.3
looic       -64.0 18.9
------
Monte Carlo SE of elpd_loo is 0.1.

Pareto k diagnostic values:
                         Count Pct.    Min. n_eff
(-Inf, 0.5]   (good)     81    98.8%   160
 (0.5, 0.7]   (ok)        1     1.2%   953
   (0.7, 1]   (bad)       0     0.0%   <NA>
   (1, Inf)   (very bad)  0     0.0%   <NA>
```