Importance sampling and approximate LOO-CV

ISTA 410 / INFO 510: Bayesian Modeling and Inference

U. of Arizona School of Information

March 17, 2021

Outline

Today:

- Approximate leave-one-out cross-validation
- Detecting influential outliers with Pareto k values

Leave-one-out cross-validation

LOO cross-validation

Idea behind cross-validation:

- Hold out some of your data for evaluation
- Fit the model on the remaining data
- Evaluate the model by estimating lppd on the held-out data
- Repeat, with different partitionings

Types of cross-validation

k-fold CV:

- Partition the data set into k equal subsets
- Each subset gets a turn as the hold-out set
- Problem: dependent on the (arbitrary) partition

Leave-one-out CV:

- Each observation gets a turn as the hold-out set
- Exhaustive: no arbitrary choices involved in choosing hold-out sets
- Problem: many re-fits required

Importance sampling

Importance sampling

Importance sampling:

- Method related to rejection sampling and Metropolis
- Goal: calculate an average of a quantity h over some probability distribution, when we can only sample from an approximation to that distribution
  - Our case: want to calculate expected log score on ith observation over $p(\theta|y_{-i})$, the posterior with that observation dropped
  - But we don't want to calculate every one of those posteriors, so we use the full $p(\theta|y)$ as an approximation

Importance sampling in general

Idea: we want to calculate the average of $h(\theta)$, where θ follows a probability distribution $p(\theta)$. If we had a sample $\{\theta_s\}$ from $p(\theta)$, we could just evaluate h and average:

$$E[h(\theta)] \approx \frac{1}{S} \sum_s h(\theta_s)$$

But suppose our sample $\{\theta_s\}$ comes instead from an approximate distribution $q(\theta)$. Then the above doesn't work; but, if we re-weight each term in the sum we can recover a good approximation.

Importance sampling in general

Define the *importance ratio* or *importance weight*:

$$w(\theta_s) = \frac{p(\theta_s)}{q(\theta_s)}$$

then,

$$E[h(\theta)] \approx \frac{\sum_s h(\theta_s) w(\theta_s)}{\sum_s w(\theta_s)}$$

i.e. just a weighted average, weighted by the importance ratios.

Idea: samples with a high $q(\theta_s)$ are more "important" to the distribution we are trying to target

Importance sampling for LOO-CV

In LOO-CV we are trying to estimate $\log p(y_i|y_{-i})$, where y_{-i} denotes the set of observed y values without y_i.

We calculate importance weights for each sample θ_s:

$$w(\theta_s) = \frac{1}{p(y_i|\theta_s)}$$

and get an estimate for the lppd for that observation:

$$\mathrm{lppd} \approx \frac{\sum_s p(y_i|\theta_s)w(\theta_s)}{\sum_s w(\theta_s)}$$

Then our estimated out-of-sample log score is the sum of the above over observations i.

Smoothing

Importance weights can be unreliable:

- If one or a few importance weights are much larger than the others, they can dominate the estimate and make it inaccurate
- So, we want to "smooth" the estimate

Under some standard conditions, largest importance weights should follow a *generalized Pareto distribution*:

$$p(w|u,\sigma,k) = \sigma^{-1}(1 + k(r-u)\sigma^{-1})^{-\frac{1}{k}-1}$$

- u, σ – location and scale
- k – shape; controls the weight of the tail

Smoothing

In Pareto-smoothed importance sampling:

- the largest 20% of importance weights are used to fit the parameters for a generalized Pareto distribution
- those weights are then replaced by quantiles from the same distribution

Still doesn't work very well if the Pareto distribution's shape is bad:

- $k > 0.5$: distribution has infinite variance
- In practice, still usually ok if $k < 0.7$; for larger k, can't necessarily trust approximation

Detecting high-influence points

When are the importance ratios really big?

$$w_(\theta_s) = \frac{1}{p(y_i|\theta_s)}$$

When the posterior distribution assigns low probability to an observation.

If k is large for the Pareto distribution fitted for y_i, that indicates that these weights are really big – suggesting that the model cannot accomodate that point well.

- use `az.loo(trace, pointwise = True)`

Let's see it...

Applying WAIC and LOO-CV

We now have two numerical tools for estimating out-of-sample deviance: WAIC and LOO-CV.

- In ordinary linear models, LOO-CV and WAIC perform pretty similarly. LOO-CV has higher variance, WAIC higher bias as estimates of the KL divergence.
- In practice differences are usually small; best practice is to compute both. If there are large differences, this may indicate that one or both are unreliable
- Computational problems with both can sometimes be resolved by using a more robust model

Summary

Today:

- Importance sampling
- Pareto smoothed LOO-CV

Going forward:

- ~ 3 weeks: return to linear models: causal inference, interactions, multilevel regression
- ~ 1 week: Gaussian processes
- ~ 2 weeks: time series models, HMM, Kalman filters