Uncertainty assessment via iterated simulated learning

P. Richard Hahn

Facebook, July 2015
Bruno de Finetti (1906-1985)

- Trained as a mathematician.
- Worked as an actuary.
- Well-known as a Bayesian statistician and philosopher.
- Posed two questions:
  - **Why** probability models?
  - **Which** probability models?
Outline

- Uncertainty assessment via forward sequential simulation
- Sequential coherence
- Model specification via backward induction
  - Bernoulli likelihood
  - Kernel density model
- Partially coherent max-ent predictive distributions
Representation theorems

If $y_1, y_2, \ldots$ are \textit{infinitely exchangeable} (shuffling indices doesn't change the joint distribution), then (as $n \to \infty$),

$$p(y_{1:n}) \approx \int \prod_{i=1}^{n} f(y_i \mid \theta) \pi(\theta) d\theta. \quad (1)$$

The form of $p(\cdot)$ dictates form of $f(\cdot)$ and $\pi(\theta)$.

With $f(\cdot)$ and $\pi(\theta)$ in hand, learning from data proceeds via Bayes rule.
“Parameter free” sequential predictive Bayes

\[ p(y_{1:n}) = p(y_1)p(y_2 | y_1)p(y_3 | y_{1:2}) \ldots p(y_n | y_{1:(n-1)}). \]

Suppose we are interested in some particular feature \( g \) of the distribution of the data. To obtain one draw from the prior/posterior over this quantity

1. Simulate \( y_{n+1}^* \) from \( p_n(y | y_{1:n}) \).
2. Simulate \( y_{n+2}^* \) from \( p_{n+1}(y | y_{1:n}, y_{n+1}^*) \),
3. Simulate \( y_{n+3}^* \) from \( p_{n+2}(y | y_{1:n}, y_{(n+1):(n+2)}^*) \),
   \[ \ldots \]
4. Extract the desired summary from \( p_N(y | y_{1:n}, y_{(n+1):m}^*) \)
   \[ (m = n + N). \]

Repeating gives draws from the posterior distribution of \( \theta \equiv g[p_m] \).
Example: Bernoulli data

Suppose $Y_i \sim \text{Bernoulli}(\theta)$ with prior $\theta \sim \text{Beta}(\alpha, \beta)$. Integrating over this prior yields the following predictive updates

$$p_t(y_{t+1} \mid y_{1:t}) = \text{Bernoulli} \left( \frac{\alpha_t}{\alpha_t + \beta_t} \right),$$

$$\alpha_t = \alpha_{t-1} + y_t,$$

$$\beta_t = \beta_{t-1} + 1 - y_t.$$  \hspace{1cm} (2)

Now, suppose $n = 10$ observations are observed, and that seven of them are ones: $\sum_{i=1}^{n} y_i = 7$. 
Example: Bernoulli data

The distant-future quantity $\theta$ is uncertain precisely because many different future realizations are possible.
Example: Bernoulli data

By $m = 1000$ we recapitulate the known closed-form posterior nicely.
Specifying joint distributions is hard. If we get it wrong, our inferences are suspect.

Our goal will be to “bake in” the asymptotic behavior of our predictive distribution while retaining the sequential coherence property: for any $h$

$$p_h(y) = \int p_{h+1}(y \mid x)p_h(x)dx.$$  

Our approach will be to start from $p_m(y \mid y_{1:n}, y_{(n+1):m}^*)$ and work backwards.
Example: Bernoulli data

Assume that for a sample of size $m$ and $\bar{y} = m^{-1} \sum_i y_i$, an adequately accurate predictive distribution for $Y_{m+1}$ is Bernoulli($\bar{y}$).

Now, one-by-one integrate out the hypothetical future data, obtaining a sequence of predictives to work with.

Let $p_m = \Pr(Y = 1 \mid y_{1:m})$ and $p_m(a) = \Pr(Y = 1 \mid X = a)$. Then

$$p_{m-1} = p_m(1)p_{m-1} + p_n(0)(1 - p_m)$$

$$= \left( \frac{m-1}{m} \bar{y}_{m-1} + \frac{1}{m} \right) p_{m-1} + \frac{(m-1)}{m} \bar{y}_{m-1}(1 - p_{m-1}),$$

$$= \bar{y}_{m-1}.$$
Bernoulli posteriors

```r
for (j in 1:MCsamples){
    ybar <- datamean
    n <- nobs
    for (h in 1:m){
        ytemp <- rbinom(1,1,ybar) # draw from predictive
        ybar <- ytemp/(n+1) + n*ybar/(n+1) # update statistic
        n <- n+1 # increment iteration counter
    }
    theta[j] <- ybar # pick off future parameter
}
```
Remarks

- The Gaussian case with known variance is easy to work out.

- Closed form solutions exist for the NEF-QVF distributions.

- Prior free!

- We obtain the usual Bayesian posterior by seeding with pseudo-observations.

- The simulation is trivially parallelizable.

Can we do anything more interesting?
Kernel density estimators

Recall kernel density estimators of the form

$$K_n^\tau(y) = \sum_{i=1}^{n} \phi(y \mid y_i, \tau),$$

where $\phi(y \mid \mu, \tau)$ is a normal density function with center $\mu$ and "bandwidth" (variance) $\tau$.

This is going to be our $p_m$ predictive density: if we had $m$ observations, we think a kernel density estimate would be reliable.
Kernel density estimators

Begin by considering the marginalization consistency criterion applied to a KDE at sample size $m$.

$$p_{m-1}(y) = \int K_{m}^{\tau}(y)p_{m-1}(x)dx.$$  

Now “peel off” the $m$th observation $x \equiv y_m$.

$$p_{m-1}(y) = \frac{m-1}{m}K_{m-1}^{\tau}(y) + \frac{1}{m} \int \phi(y \mid x, \tau)p_{m-1}(x)dx.$$
Kernel density estimators

Next, substitute into the recursion:

\[
\frac{m - 1}{m} K_{m-1}^{\tau}(y) + \frac{1}{m} \int \phi(y \mid x, \tau) \left[ \frac{m - 1}{m} K_{m-1}^{\tau}(x) + \frac{1}{m} \int \phi(x \mid x', \tau) p_{m-1}(x') dx' \right] dx
\]

and simplify

\[
\frac{m - 1}{m} K_{m-1}^{\tau}(y) + \frac{m - 1}{m^2} K_{m-1}^{2\tau}(y) + \frac{1}{m^2} \int \int \phi(y \mid x, \tau)\phi(y \mid x, \tau) p_{m-1}(x') dx' dx
\]

and again

\[
\text{same} + \frac{1}{m^2} \int \phi(y \mid x, 2\tau) p_{m-1}(x) dx.
\]
Putting it all together we arrive at

\[ p_{m-1}(y) = \sum_{j=1}^{\infty} \frac{m-1}{mj} K_j^{m-1}(y). \]

This can be expressed as an expectation

\[ p_{m-1}(y) = E K_{m-1}^{Z_{m-1}}(y). \]

where \( Z \sim \text{Geometric}(\rho) \), with \( \rho = \frac{m-1}{m} \).
Stepping back further towards the present...

Because our derivation involved summation and convolution, we can repeat it for each term in that infinite sum, by considering the mappings

\[ m \to m - 1, \]
\[ \tau \to j \tau. \] (3)

Substitution and iteration yields

\[ p_{m-n+1} = \sum_{q=1}^{\infty} \cdots \sum_{k=1}^{\infty} \sum_{j=1}^{\infty} \frac{m - 1}{m^j} \frac{m - 2}{(m - 1)^k} \cdots \frac{m - n}{(m - n + 1)^q} K_{m-n+1}^{q \cdots k \tau}(y). \]
A lognormal approximation

However, this expression depends only on the product,

$$\chi = Z_1 \times Z_2 \times \ldots Z_n,$$

of Geometric random variables $Z_h$ with parameters $\frac{m-h}{m-h+1}$.

For a large number of factors, a lognormal approximation applies:

$$p_{m-1}(y) = EK_{m-1}^{\chi \tau}(y),$$

with $\chi \sim \text{log-Normal}(\mu, \sigma)$ where $\mu$ and $\sigma$ are known functions of $m$ and $n$. 
The $n + 1$ predictive density bandwidth has a more diffuse log-normal mixing distribution, which collapses to an approximate point mass at $\tau$ by the $m$th predictive.
Varying bandwidth kernels (marginally)
Simulation is easy. At step $t$,

1. Select a location parameter $u$ at random among the previous $n + t - 1$ data points (of which $t - 1$ are simulated).

2. Next, draw a scale parameter $s$ from the sample size-appropriate log-normal distribution.

3. Finally, draw (pseudo-)observation $y_{n+t}^*$ from $N(u, \tau(s + 1))$.

Repeat $N$ times. Store the resulting KDE as one posterior draw.
Example: synthetic data, \( n = 50 \)
Example: synthetic data, $n = 500$
Example: the galaxy data
Advantages

- No MCMC required.

- The posterior mean is just the one-step-ahead predictive (by construction).

- A simple application of Azuma-Hoeffding inequality gives explicit posterior concentration guarantees.

- Still prior free!

- But we can seed with pseudo-observations. Much easier to understand compared to DP mixture model prior.

- The simulation is trivially parallelized.
More details and discussion are in the paper (at my research page):  

*Hahn, P. Richard.* Model specification via sequential coherence and backward induction.

I’m working out extensions to the regression case and also hierarchical models ("partial exchangeability").

The stuff I am most optimistic about is something I call partially coherent inference.
Partially coherent predictive distributions

When coherence is imposed via the density function itself, one has sequential coherence for any measurable function \( s(\cdot) \):

\[
E_h[s(Y)] = E_n[E_{h+1}(s(Y) \mid X)].
\]

If one demands coherence only for a certain (vector-valued) function \( s(\cdot) \), model specification becomes easier.

To derive our simulation distribution sequence, we start from

\[
E_m[s(Y) \mid y_{1:m}] = \hat{s}(y_{1:m})
\]

where \( \hat{s}(y_{1:m}) \) is some consistent estimator.
When $\hat{s}$ is an empirical moment estimator, the coherence condition is especially simple:

$$E_{m-1}[s(Y) \mid y_1:(m-1)] = E_{m-1}[\hat{s}(y_1:m)]$$

$$= \frac{m-1}{m}\hat{s}(y_1:(m-1)) + \frac{1}{m}E_{m-1}[\hat{s}(Y_m)],$$

Thus, we need only restrict $Y_m$ such that $E_{m-1}[s(Y)] = \hat{s}(y_1:(m-1))$.

A similar logic holds all the way down to $Y_{n+1}$, the first simulated data point.
Max-ent predictive distributions for conservative inference

So, partially coherent inference reduces to a sequence of simulations from distributions with specified moment constraints.

A natural choice is the **maximum entropy distribution**, subject to the given constraints — the *least informative* distribution with the desired moments.

The general form of a max-ent distribution is

\[
f(y) \propto \exp \left\{ \sum_j \lambda_j s_j(y) \right\},
\]

where \(s_j(y)\) is the \(j\)th element of the vector \(s(y)\).
This is different than usual Bayes

Even the simple case of $s(X) = (X, X^2)$ is different: we get a sequence of normal distributions, Bayes gives a sequence of t-distributions.
This is a cousin of the bootstrap

If at step $h$ we sample uniformly from the discrete sample $(y_{1:n}, y_{(n+1):h}^*)$, we clearly satisfy all moment constraints.

This is similar to, but not the same as, the bootstrap. (Nor is it exactly the “Bayesian bootstrap”.)
Statistical uncertainty as missing data

If we have a predictive model for our future/missing/unobserved data, then we can forward simulate to calculate uncertainty intervals.

“Population quantities are uncertain precisely because many different future realizations are possible.

(Partial) coherence is an intuitive criterion that restricts our class of predictive models, and guarantees that our uncertainty intervals are well-defined.