Appendices to "Bayesian hierarchical stacking: Some models are (somewhere) useful"

Yuling Yao, Pirš Gregor, Aki Vehtari, Andrew Gelman

The appendices contain five parts: (A) an illustrative theory example, (B) proofs of theorems, (C) guidance on software implementation in `R` and `Stan`, (D) recommendations on prior choices and (E) details of the numerical experiments.

The indices of equations and graphs in the appendices follow the main manuscript.

Appendix A: A theoretical example

Before theorem proofs, we first consider a toy example. It can be solved with a closed form solution and illustrates how Theorems 1–4 apply.

As shown in Figure 10, the true data generating process (DG) of the outcome is $y \sim \text{uniform}(-3, 1)$, and there are two given (pre-trained) models with spike-and-slab predictive distributions

$$M_1 : y \sim .99\ \text{uniform}(-4, 0) + .01\ \text{uniform}(0, 2),$$
$$M_2 : y \sim .99\ \text{uniform}(0, 2) + .01\ \text{uniform}(-4, 0),$$

which yield piece-wise constant predictive densities

$$p_1(y) = 0.99/4\mathbb{1}(y \in [-4, 0]) + 0.01/2\mathbb{1}(y \in [0, 2]),$$
$$p_2(y) = 0.99/2\mathbb{1}(y \in [0, 2]) + 0.01/4\mathbb{1}(y \in [-4, 0]).$$

Using our notation in Section 3.2, the region in which M_1 predominates is $\mathcal{J}_1 = [-4, 0]$, and M_2 outperforms on $\mathcal{J}_2 = (0, 2]$ (the conventions send the tie $\{0\}$ to M_1). We count their masses with respect to the true DG: $\Pr(\mathcal{J}_1) = 3/4$ and $\Pr(\mathcal{J}_2) = 1/4$.

Complete-pooling stacking solves

$$\max_{\mathbf{w}\in\mathcal{S}_2} \int_{-3}^{1} 1/4 \log\Big((w_1 0.99/4 + w_2 0.01/4)\mathbb{1}(y \in [-4, 0]) + (w_2 0.99/2 + w_1 0.01/2)\mathbb{1}(y \in [0, 2])\Big) dy.$$

The exact optimal weight is $w_1 = 0.755$, close to the mass $\Pr(\mathcal{J}_1) = 0.75$ and is irrelevant to the height of each regions. For instance, if the right bump in M_2 shrinks to the interval $(0, 1)$ (i.e., $y \sim 0.99\ \text{uniform}(0, 1) + 0.01\ \text{uniform}(-4, 0)$), then the winning margin therein is twice as big, while the winning probability as well as the stacking weight remains nearly unchanged.

At the pointwise level, stacking behaves as a plurality voting system: as long a model "wins" a sub-region (subject to a prefixed threshold L in condition (19)), *the winner take all* and its winning margin no longer matters.

By contrast, likelihood-based model averaging techniques such as Bayesian model averaging (BMA, Hoeting et al., 1999) and pseudo-Bayesian model averaging (Yao et al., 2018) are analogies of *proportional representation*: every count of the winning margin matters. For illustration, we vary the slab probability δ in Model 1 and 2:

$$\begin{aligned} M_1 \mid \delta: &\quad y \sim (1-\delta) \times \text{uniform}(-4, 0) + \delta \times \text{uniform}(0, 2), \\ M_2 \mid \delta: &\quad y \sim (1-\delta) \times \text{uniform}(0, 2) + \delta \times \text{uniform}(-4, 0). \end{aligned}$$

The left column in Figure 11 visualizes the predictive densities from these two models at $\delta = 0.2$, 0.33, and 0.45.

When the slab probability δ increases from 0 to 0.5, these two models are closer and closer to each other, measured by a smaller $\text{KL}(M_1, M_2)$. The $(0.5, 1)$ counterpart is similar, though not exactly symmetric. We compute stacking weight and the expected pseudo-BMA weight with sample size n: $w_1^{\text{BMA}}(n, \delta) = \left(1 + \exp\left(n\,\mathbb{E}_{y|\delta} \log p_2(y) - n\,\mathbb{E}_{y|\delta} \log p_1(y)\right)\right)^{-1}$.

Interestingly, pseudo-BMA weight $w_1^{\text{BMA}}(n, \delta)$ is strictly decreasing as a function of $\delta \in (0, 1)$. This is because when $\delta \to 0^+$, log predictive density of model 2 in the left part $\log(\delta/4) \to \infty$ can be arbitrarily small, and the influence of this bad region dominates the overall performance of model 2. By contrast, stacking weight is monotonic non-increasing on $(0, 0.5)$ (strictly decreasing on $(0, 1/3)$, and remains flat afterwards)—the opposite direction of BMA. Stacking simply recognizes model 1 winning the $[-3, 0]$ interval and does not haggle over how much it wins.

In addition, when $\delta = 1/3$, M_1 becomes a uniform density on $[-4, 2]$. When $\delta \in (1/3, 1/2)$, model 2 is not only strictly worse than model 1 but also provides no extra information for model averaging. Hence stacking assigns it weight zero.

The first two panels in Figure 12 show the KL divergence from model 1 or from model 2 to the data generating process and the KL divergence between model 1 and model 2. The third panel is the largest separation constant L for which the separation condition (19) holds. The last two panels show the stacking epld gain (compared with

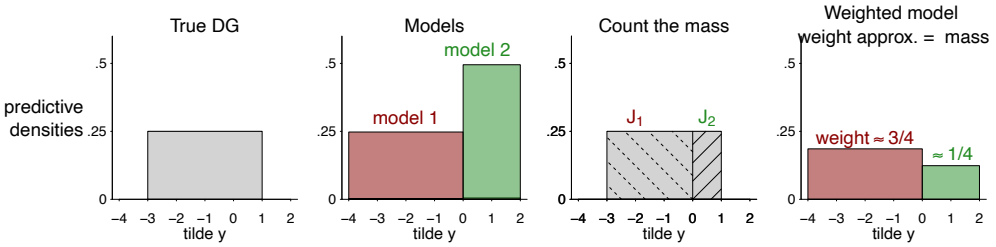

Figure 10: *The true data is generated from uniform*$(-3, 1)$ *and there are two models with spikes and slabs on intervals* $(-4, 0)$ *and* $(0, 2)$ *respectively.* $\mathcal{J}_1$ *and* $\mathcal{J}_2$ *in Theorem 1 are* $[-4, 0]$ *and* $(0, 2]$*, with DG probabilities 3/4 and 1/4. The stacking weights are approximately these two probabilities, and irrelevant to how high the winning margins are.*

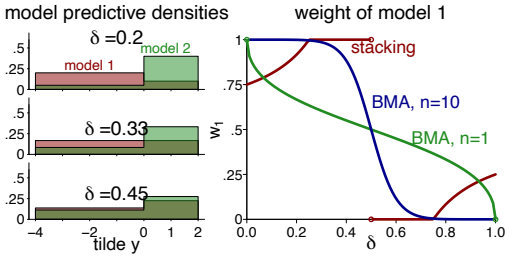

Figure 11: *Left: Pointwise predictive density $p(\tilde{y}|M_1$ or $M_2)$ when the slab probability δ is chosen 0.2, 1/3 and 0.45. Right: Weight of model 1 in complete-pooling stacking (not defined at $\delta = 0.5$) and pseudo-BMA (sample size n=1 or 10, not defined at $\delta = 0$ or 1) as a function of the slab probability δ. They evolve in the opposite direction. Besides, stacking weights are more polarized when models are more similar.*

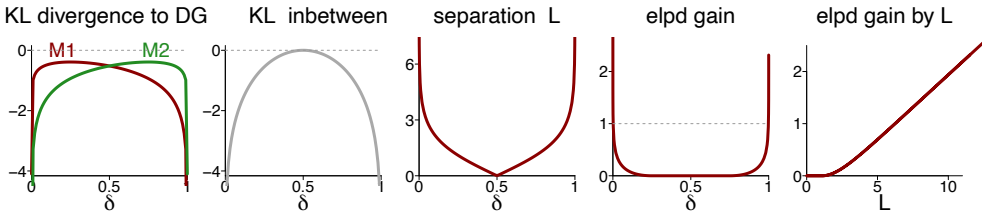

Figure 12: *From left: (1) KL divergence between model 1 or model 2 and data generating process. (2) KL divergence between model 1 and model 2. (3) Separation constant L. (4) Stacking elpd gain compared with the best individual model. (5) Stacking elpd gain as a function of L.*

the best individual model) as a function of δ and L. This constructive example reflects the worst case for it matches the theoretical lower bound $g^*(L, K, \rho, \epsilon) = \log(\rho) + (1 - \rho)(\log(1 - \rho) - \log(K - 1))$ (here $L = L, K = 2, \rho = 1/4, \epsilon = 0$) in Theorem 3.

When $\delta \in [1/3, 1/2)$, Model 2 still wins on the interval $\mathcal{J}_2 = (0, 2]$ with the separation constant $\epsilon = 0$ and $L \leq \log 2$ (the winning margin is maximized at $\delta = 1/3$). Nevertheless, a zero stacking weight and a non-zero winning area do not contradict Theorem 1. Indeed, Theorem 2 precisely bounds the mass of the winning region when stacking weight is zero. We provide self-contained theorem proofs in the next section.

Loosely speaking, BMA computes the probability of a model being *true* (if one model *has to* be true), while stacking (through the approximation $\Pr(\mathcal{J}_k)$) computes the probability of a model being the *best*.

Appendix B: Proofs of theorems

For brevity, in later proofs we will use the abbreviation for the posterior pointwise conditional predictive density from the k-th model:

$$p_k(\tilde{y}|\tilde{x}) \coloneqq p(\tilde{y}|\tilde{x}, M_k) = \int p(\tilde{y}|\tilde{x}, \theta_k) p(\theta_k|\mathcal{D}) d\theta_k, \quad k = 1, \ldots, K.$$

This subscript index k should not be confused with the notation t as in $p_t(\tilde{y}|\tilde{x})$ or $p_t(\tilde{y}, \tilde{x})$: the unknown conditional or joint density of the true data generating process. The subscript letter t is always reserved for "*true*".

Recall that in this section $\mathbf{w}^{\text{stacking}}$ refers to the complete-pooling stacking in the population:

$$\mathbf{w}^{\text{stacking}} \coloneqq \arg\max_{\mathbf{w} \in \mathcal{S}_K} \text{elpd}(\mathbf{w}),$$
$$\text{elpd}(\mathbf{w}) = \int_{\mathcal{X} \times \mathcal{Y}} \log \left(\sum_{k=1}^{K} w_k p(\tilde{y}|M_k, \tilde{x}) \right) p_t(\tilde{y}, \tilde{x}) d\tilde{y} d\tilde{x}.$$

Theorem 1. *We call K predictive densities $\{p(\tilde{y} = \cdot|\tilde{x} = \cdot, M_k)\}_{k=1}^K$ to be locally separable with a constant pair $L > 0$ and $0 < \epsilon < 1$ with respect to the true data generating process $p_t(\tilde{y}, \tilde{x})$, if*

$$\sum_{k=1}^{K} \int_{(\tilde{x}, \tilde{y}) \in \mathcal{J}_k} \mathbb{1}\Big(\log p(\tilde{y}|\tilde{x}, M_k) < \log p(\tilde{y}|\tilde{x}, M_{k'}) + L, \ \forall k' \neq k \Big) p_t(\tilde{y}, \tilde{x}) d\tilde{y} d\tilde{x} \leq \epsilon.$$

For a small ϵ and a large L, the stacking weights that solve (3) is approximately the proportion of the model being the locally best model:

$$\mathbf{w}_k^{\text{stacking}} \approx \mathbf{w}_k^{\text{approx}} \coloneqq \Pr(\mathcal{J}_k) = \int_{\mathcal{J}_k} p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{y} d\tilde{x}.$$

in the sense that the objective function is nearly optimal:

$$|\text{elpd}(\mathbf{w}^{\text{approx}}) - \text{elpd}(\mathbf{w}^{\text{stacking}})| \leq \mathcal{O}(\epsilon + \exp(-L)).$$

Proof. The expected log predictive density of the weighted prediction $\sum_k w_k p_k(\cdot|x)$ (as a function of $\mathbf{w}$) is

$$\begin{aligned}
\text{elpd}(\mathbf{w}) &= \int_{\mathcal{X} \times \mathcal{Y}} \log \left(\sum_{l=1}^{K} w_l p_l(\tilde{y}|\tilde{x}) \right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= \sum_{k=1}^{K} \int_{\mathcal{J}_k} \log \left(\sum_{l=1}^{K} w_l p_l(\tilde{y}|\tilde{x}) \right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= \sum_{k=1}^{K} \int_{\mathcal{J}_k} \log \left(w_k p_k(\tilde{y}|\tilde{x}) + \sum_{l \neq k} w_l p_l(\tilde{y}|\tilde{x}) \right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y}
\end{aligned}$$

$$= \sum_{k=1}^{K} \int_{\mathcal{J}_k} \left(\log\left(w_k p_k(\tilde{y}|\tilde{x})\right) + \log\left(1 + \sum_{l \neq k} \frac{w_l p_l(\tilde{y}|\tilde{x})}{w_k p_k(\tilde{y}|\tilde{x})}\right)\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y}.$$

The expression is legit for any simplex vector $\mathbf{w} \in \mathcal{S}_K$ that does not contain zeros. We will treat zeros later. For now we only consider a dense weight: $\{\mathbf{w} \in \mathcal{S}_K : w_k > 0, k = 1, \dots K\}$.

Consider a surrogate objective function (the first term in the integral above):

$$\begin{aligned}
\text{elpd}^{\text{surrogate}}(\mathbf{w}) &= \sum_{k=1}^{K} \int_{\mathcal{J}_k} \log\left(w_k p_k(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= \sum_{k=1}^{K} \int_{\mathcal{J}_k} \left(\log w_k + \log p_k(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= \sum_{k=1}^{K} \log w_k \int_{\mathcal{J}_k} p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} + \sum_{k=1}^{K} \int_{\mathcal{J}_k} \log p_k(\tilde{y}|\tilde{x}) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= \sum_{k=1}^{K} \left(\Pr(\mathcal{J}_k) \log w_k\right) + \text{constant}.
\end{aligned}$$

Ignoring the constant term above (the expected cross-entropy between each conditional prediction and the true DG), to maximize the surrogate objective function is equivalent to maximizing $\sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log w_k$, we call this function elbo($\mathbf{w}$), the *evidence lower bound*. To optimize $\text{elpd}^{\text{surrogate}}$ is equivalent to optimizing elbo. We show that this elbo function has a closed form optimum. Using Jensen's inequality,

$$\begin{aligned}
\text{elbo}(\mathbf{w}) &= \sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log w_k \\
&= \sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log \frac{w_k}{\Pr(\mathcal{J}_k)} + \sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log \Pr(\mathcal{J}_k) \\
&\leq \log\left(\sum_{k=1}^{K} \Pr(\mathcal{J}_k) \frac{w_k}{\Pr(\mathcal{J}_k)}\right) + \sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log \Pr(\mathcal{J}_k) \\
&= \sum_{k=1}^{K} \Pr(\mathcal{J}_k) \log \Pr(\mathcal{J}_k).
\end{aligned}$$

The equality is attained at $w_k = \Pr(\mathcal{J}_k),\ k = 1, \dots, K$, which reaches our definition of $\mathbf{w}^{\text{approx}}$ in Theorem 1.

What remains to be proved is that the surrogate objective function is close to the actual objective. We divide each set $\mathcal{J}_k$ into two disjoint subsets $\mathcal{J}_k = \mathcal{J}_\circ \cup \mathcal{J}_k^\bullet$, for

$$\mathcal{J}_k^\circ \coloneqq \{(\tilde{x}, \tilde{y}) \in \mathcal{J}_k : \log p(\tilde{y}|\tilde{x}, M_k) < \log p(\tilde{y}|\tilde{x}, M_{k'}) + L\};$$

$$\mathcal{J}_k^\bullet \coloneqq \{(\tilde{x},\tilde{y}) \in \mathcal{J}_k : \log p(\tilde{y}|\tilde{x}, M_k) \geq \log p(\tilde{y}|\tilde{x}, M_{k'}) + L\}\,.$$

The separation condition ensures $\sum_{k=1}^K \Pr(\mathcal{J}_k^\circ) \leq \epsilon$.

Let $\Delta(\mathbf{w}) = \text{elpd}(\mathbf{w}) - \text{elpd}^{\text{surrogate}}(\mathbf{w})$. For any fixed simplex vector $\mathbf{w}$, this absolute difference of the objective function is bounded by

$$\begin{aligned}
|\Delta(\mathbf{w})| &= \left|\sum_{k=1}^K \int_{\mathcal{J}_k} \left(\log\left(1 + \sum_{l\neq k} \frac{w_l p_l(\tilde{y}|\tilde{x})}{w_k p_k(\tilde{y}|\tilde{x})}\right)\right) p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y}\right| \\
&\leq \sum_{k=1}^K \int_{\mathcal{J}_k} \left|\log\left(1 + \sum_{l\neq k} \frac{w_l p_l(\tilde{y}|\tilde{x})}{w_k p_k(\tilde{y}|\tilde{x})}\right)\right| p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y} \\
&= \sum_{k=1}^K \left(\int_{\mathcal{J}_k^\circ} + \int_{\mathcal{J}_k^\bullet}\right) \left|\log\left(1 + \sum_{l\neq k} \frac{w_l p_l(\tilde{y}|\tilde{x})}{w_k p_k(\tilde{y}|\tilde{x})}\right)\right| p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y} \\
&\leq \sum_{k=1}^K \int_{\mathcal{J}_k^\circ} \log(1 + \sum_{l\neq k} \frac{w_l}{w_k}) p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y} + \sum_{k=1}^K \int_{\mathcal{J}_k^\bullet} \sum_{l\neq k} \frac{w_l}{w_k} \frac{p_l(\tilde{y}|\tilde{x})}{p_k(\tilde{y}|\tilde{x})} p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y} \\
&\leq \left(\sum_{k=1}^K \sum_{l\neq k} \frac{w_l}{w_k}\right) \left(\sum_{k=1}^K \int_{\mathcal{J}_k^\circ} p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y} + \sum_{k=1}^K \int_{\mathcal{J}_k^\bullet} \frac{p_l(\tilde{y}|\tilde{x})}{p_k(\tilde{y}|\tilde{x})} p_t(\tilde{y}|\tilde{x})p(\tilde{x})d\tilde{x}d\tilde{y}\right) \\
&\leq \left(\sum_{k=1}^K \frac{1-w_k}{w_k}\right)(\epsilon + \exp(-L)).
\end{aligned}$$

The second inequality used $\log(1+x) \leq x$ for $x \geq 0$.

The exact optima of objective function is $\mathbf{w}^{\text{stacking}}$. Using the inequality above twice,

$$\begin{aligned}
0 \leq \text{elpd}(\mathbf{w}^{\text{stacking}}) - \text{elpd}(\mathbf{w}^{\text{approx}}) &\leq |\text{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{stacking}}) - \text{elpd}(\mathbf{w}^{\text{stacking}})| \\
&\quad + |\text{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}}) - \text{elpd}(\mathbf{w}^{\text{approx}})| \\
&\quad + \text{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{stacking}}) - \text{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}}) \\
&\leq |\Delta(\mathbf{w}^{\text{approx}})| + |\Delta(\mathbf{w}^{\text{stacking}})| \\
&\leq \sum_{k=1}^K \left(\frac{1-w_k^{\text{approx}}}{w_k^{\text{approx}}} + \frac{1-w_k^{\text{stacking}}}{w_k^{\text{stacking}}}\right)(\epsilon + \exp(-L)).
\end{aligned}$$

It has almost finished the proof except for the simplex edge where w_k^{stacking} or w_k^{approx} attains zero.

Without loss of generality, if $w_1^{\text{approx}} = 0, w_k^{\text{approx}} \neq 0, \forall k \neq 1$, which means $p(\tilde{y}|M_k, \tilde{x})$ is always inferior to some other models. This will only happen if $p(\tilde{y}|M_k, \tilde{x})$ is almost sure zero (w.r.t $p_t(\tilde{y}|\tilde{x})p(\tilde{x})$) hence we can remove model 1 from the model list, and the same $\mathcal{O}(\epsilon + \exp(-L))$ bound applies to remaining model $2, \ldots, K$. If there are more than one zeros, repeat until all zeros have been removed.

Next, we deal with $w_1^{\text{stacking}} = 0, w_k^{\text{stacking}} \neq 0, \forall k \neq 1$. If $w_1^{\text{approx}} = 0$, too, then we have solved in the previous paragraph. If not, Theorem 2 shows that w_1^{approx} has to be a small order term:

$$\Pr(\mathcal{J}_1) \leq (1 + (\exp(L) - 1)(1 - \epsilon) + \epsilon)^{-1} < \exp(-L) + \epsilon.$$

We leave the proof of this inequality in Theorem 2.

The contribution of the first model in the surrogate model is at most $\Pr(\mathcal{J}_1) \log \Pr(\mathcal{J}_1)$. After we remove the first model from the model list, with the surrogate model elpd changes by at most a small order term, not affecting the final bound. Because the separation condition with constant (ϵ, L) applies to model $1, \ldots, K$, and due to lack of a competition source, the same separation condition applies to model $2, \ldots, K$ and the same bound applies.

□

Theorem 2. *When the separation condition (19) holds, and if the k-th model has zero weight in stacking, $w_k^{\text{stacking}} = 0$, then the probability of its winning region is bounded by:*

$$\Pr(\mathcal{J}_k) \leq (1 + (\exp(L) - 1)(1 - \epsilon) + \epsilon)^{-1}.$$

The right hand side can be further upper-bounded by $\exp(-L) + \epsilon$.

Proof. Without loss of generality, assume $w_1^{\text{stacking}} = 0$. Let $p_0(\tilde{y}|\tilde{x}) = \sum_{k=2}^{K} \mathbf{w}^{\text{stacking}} p_k(\tilde{y}|\tilde{x})$. Consider a constrained objective $\widetilde{\text{elpd}}(w_1) = \mathbb{E}(\log(w_1 p_1(\tilde{y}|\tilde{x}) + (1 - w_1)p_0(\tilde{y}|\tilde{x})))$ where the expectation is over both $\tilde{y}$ and $\tilde{x}$ as before. Because the max is attained at $w_1 = 0$ and because $\log(\cdot)$ is a concave function, the derivative at any $w_1 \in [0, 1]$ is

$$\frac{d}{dw_1}\widetilde{\text{elpd}}(w_1) = \mathbb{E}_{\tilde{y},\tilde{x}}\left(\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{w_1 p_1(\tilde{y}|\tilde{x}) + (1 - w_1)p_0(\tilde{y}|\tilde{x})}\right) \leq 0.$$

That is

$$\begin{aligned}
0 \geq & \mathbb{E}\left(\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{p_0(\tilde{y}|\tilde{x})}\right) \\
= & \Pr(\mathcal{J}_1)\, \mathbb{E}[\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{p_0(\tilde{y}|\tilde{x})}|\mathcal{J}_1] + (1 - \Pr(\mathcal{J}_1))\, \mathbb{E}[\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{p_0(\tilde{y}|\tilde{x})}|\mathcal{J}_0] \\
\geq & \Pr(\mathcal{J}_1)\, \mathbb{E}[\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{p_0(\tilde{y}|\tilde{x})}|\mathcal{J}_1] - (1 - \Pr(\mathcal{J}_1)).
\end{aligned}$$

Rearranging this inequality arrives at

$$\begin{aligned}
1 \geq & \Pr(\mathcal{J}_1)\left(1 + \mathbb{E}[\frac{p_1(\tilde{y}|\tilde{x}) - p_0(\tilde{y}|\tilde{x})}{p_0(\tilde{y}|\tilde{x})}|\mathcal{J}_1]\right) \\
\geq & \Pr(\mathcal{J}_1)(1 + (\exp(L) - 1)(1 - \epsilon) + \epsilon).
\end{aligned}$$

As a result, the model that has stacking weight zero cannot have a large probability to predominate all other models,

$$\Pr(\mathcal{J}_1) \leq (1 + (\exp(L) - 1)(1 - \epsilon) + \epsilon)^{-1} < \exp(-L) + \epsilon.$$

□

Theorem 3. *Let* $\rho = \sup_{1 \leq k \leq K} \Pr(\mathcal{J}_k)$*, and two deterministic functions* g *and* g^* *by*

$$\begin{aligned} &g(L, K, \rho, \epsilon) = L(1-\rho)(1-\epsilon) - \log K \\ \leq &g^*(L, K, \rho, \epsilon) = L(1-\rho)(1-\epsilon) + \rho \log(\rho) + (1-\rho)(\log(1-\rho) - \log(K-1)). \end{aligned}$$

Assuming the separation condition (19) *holds for all* $k = 1, \ldots, K$*, then the utility gain of stacking is further lower-bounded by*

$$\mathrm{elpd}_{\mathrm{stacking}} - \mathrm{elpd}_k \geq \max\left(g^*(L, K, \rho) + \mathcal{O}(\exp(-L) + \epsilon), 0\right).$$

Proof. As before, we consider the approximate weights: $w_k^{\mathrm{approx}} = \Pr(\mathcal{J}_k)$, and the surrogate elpd $\mathrm{elpd}^{\mathrm{surrogate}}(\mathbf{w}) = \sum_{k=1}^K \int_{\mathcal{J}_k} \log\left(w_k p_k(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y}$.

$$\begin{aligned} &\mathrm{elpd}^{\mathrm{surrogate}}(\mathbf{w}^{\mathrm{approx}}) - \mathrm{elpd}_k \\ =&\sum_{l=1}^K \int_{\mathcal{J}_l} \log\left(\Pr(\mathcal{J}_l) p_l(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} - \sum_{l=1}^K \int_{\mathcal{J}_l} \log\left(p_k(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\ =&\sum_{l=1}^K \int_{\mathcal{J}_l} \left(\log \Pr(\mathcal{J}_l) + \log p_l(\tilde{y}|\tilde{x}) - \log p_k(\tilde{y}|\tilde{x})\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\ =&\sum_{l=1}^K \Pr(\mathcal{J}_l) \log \Pr(\mathcal{J}_l) + \sum_{l=1}^K \mathbb{1}(l \neq k) \int_{\mathcal{J}_l} \log(p_l(\tilde{y}|\tilde{x}) - \log p_k(\tilde{y}|\tilde{x})) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\ =&\sum_{l=1}^K \Pr(\mathcal{J}_l) \log \Pr(\mathcal{J}_l) + \sum_{l=1}^K \mathbb{1}(l \neq k) \left(\int_{\mathcal{J}_l^\circ} + \int_{\mathcal{J}_l^\bullet}\right) \log(p_l(\tilde{y}|\tilde{x}) - \log p_k(\tilde{y}|\tilde{x})) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\ \geq&\sum_{l=1}^K \Pr(\mathcal{J}_l) \log \Pr(\mathcal{J}_l) + (1-\epsilon)(1-\rho) L - \epsilon \\ \geq&\rho \log \rho + (1-\rho) \log \frac{1-\rho}{K-1} + (1-\epsilon)(1-\rho) L - \epsilon \\ =&g^*(L, K, \rho, \epsilon) - \epsilon. \end{aligned}$$

The last inequality comes from the fact that, under the constraint of $\max_k \Pr(\mathcal{J}_k) = \rho$, the entropy $\sum_{k=1}^K \Pr(\mathcal{J}_k) \log \Pr(\mathcal{J}_k)$ attains its minimal when each of the $\Pr(\mathcal{J}_k)$ term equals $(1-\rho)/(K-1)$ except for the largest term ρ. This inequality is due to the convexity of $x \log x$.

Finally, using the proof of Theorem 1, the error from the surrogate is bounded,

$$|\mathrm{elpd}^{\mathrm{surrogate}}(\mathbf{w}^{\mathrm{approx}}) - \mathrm{elpd}^{\mathrm{stacking}}(\mathbf{w}^{\mathrm{stacking}})| \leq \mathcal{O}(\exp(-L) + \epsilon).$$

Hence the overall utility is bounded,

$$\begin{aligned}
&\mathrm{elpd}_{\text{stacking}} - \mathrm{elpd}_k \\
&= \left(\mathrm{elpd}_{\text{stacking}} - \mathrm{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}})\right) + \left(\mathrm{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}}) - \mathrm{elpd}_k\right) \\
&\geq g^*(L, K, \rho) + \mathcal{O}(\exp(-L) + \epsilon).
\end{aligned}$$

Because selection is always a specials case of averaging, the utility is further bounded below by 0.

To replace $g^*(\cdot)$ with the looser bound $g(\cdot)$, we only need to ensure $\rho \log \rho + (1-\rho)\log\frac{1-\rho}{K-1} \geq -\log K$, for the range $\rho \in [1/K, 1), K \geq 2$. The proof is elementary. For any fixed $K \geq 2$, let $h(\rho) = \rho\log\rho + (1-\rho)\log\frac{1-\rho}{K-1} + \log K$. It is increasing on $\rho \in [1/K, 1)$, for $\frac{d}{d\rho}h(\rho) = \log\frac{(K-1)\rho}{1-\rho} \geq 0$. Hence, $h(\rho)$ attains minimum at $\rho = 1/K$, at which $h(1/K) = 0$. □

From the constrictive example (Appendix A), $g^*(\cdot)$ is a tight bound. We use the looser bound $g(\cdot)$ in the main paper for its simpler form.

Theorem 4. *Under the strong separation assumption*

$$\sum_{k=1}^{K} \int_{\tilde{x}\in\mathcal{I}_k} \int_{\tilde{y}\in\mathcal{Y}} \mathbb{1}\Big(\log p(\tilde{y}|M_k, x) < \log p(\tilde{y}|M_{k'}, x) + L,\ \forall k' \neq k^*(x)\Big) p_t(\tilde{y}|x, D) d\tilde{y} d\tilde{x} \leq \epsilon,$$

and if the sets $\{\mathcal{I}_k\}$ are known exactly, then we can construct pointwise selection

$$p(\tilde{y}|x, \text{pointwise selection}) = \sum_{k=1}^{K} \mathbb{1}(x \in \mathcal{I}_k) p(\tilde{y}|x, M_k).$$

Its utility gain is bounded from below by

$$\mathrm{elpd}_{\text{pointwise selection}} - \mathrm{elpd}_{\text{stacking}} \geq -\log\rho_{\mathcal{X}} + \mathcal{O}(\exp(-L) + \epsilon).$$

Proof.

$$\begin{aligned}
&\mathrm{elpd}_{\text{pointwise selection}} - \mathrm{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}}) \\
&= \sum_{l=1}^{K} \int_{\mathcal{I}_l} \left(\log p_l(\tilde{y}|\tilde{x}) - \log\left(\Pr(\mathcal{I}_l) p_l(\tilde{y}|\tilde{x})\right)\right) p_t(\tilde{y}|\tilde{x}) p(\tilde{x}) d\tilde{x} d\tilde{y} \\
&= -\sum_{l=1}^{K} \Pr(\mathcal{I}_l) \log \Pr(\mathcal{I}_l) \\
&\geq -\sum_{l=1}^{K} \Pr(\mathcal{I}_l) \log \rho_x \\
&= -\log \rho_x.
\end{aligned}$$

Finally, from the proof of Theorem 1,

$$|\text{elpd}^{\text{surrogate}}(\mathbf{w}^{\text{approx}}) - \text{elpd}^{\text{stacking}}(\mathbf{w}^{\text{stacking}})| \leq \mathcal{O}(\exp(-L) + \epsilon).$$

□

We close this section with two remarks. First, Yao (2019) approximates the probabilistic stacking weights under the strong separation condition (23). The result therein can be viewed as a special case of Theorem 4 in the present paper as $\Pr(\mathcal{J}_k) \approx \Pr(\mathcal{I}_k)$ under assumption (23).

Second, most proofs only use the concavity of the log scoring rule. Therefore, some proprieties of stacking weights could be extended to other concave scoring rules, too.

Appendix C: Software implementation in Stan

We summarize our formulation of hierarchical stacking by pseudo code 1.

Algorithm 1: Hierarchical stacking

Data: y: outcomes; x: input on which the stacking weights vary, z: other inputs;
$p_{k,-i}$: approximate leave-one-out predictive densities of the k-th model and i-th data.
Result: input-dependent stacking weight $\overline{\mathbf{w}}(x) : \mathcal{X} \to \mathcal{S}_K$; combined model.

1 Sample from the joint densities $p(\alpha, \mu, \sigma|\mathcal{D})$ in hierarchical stacking model (11);
2 Compute posterior mean of $w_k(\tilde{x})$ at any $\tilde{x}$, and make predictions $p(\tilde{y}|\tilde{x}, \tilde{z})$ by (12).

To code the basic additive model, we prepare the input covariate $X = (X_{\text{discrete}}, X_{\text{continuous}})$, where X_{discrete} is discrete dummy variable, and $X_{\text{continuous}}$ are remaining features (already rectified as in (16)). The dimension of these two parts are $d_{\text{continuous}}$ and d_{discrete}.

Here we use the "grouped hierarchical priors" (Section 6.3) with only two groups, distinguishing between continuous and discrete variables. We discuss more on the hyper prior choice in the next section.

$$w_{1:K}(x) = \text{softmax}(w^*_{1:K}(x)), \quad w^*_k(x) = \sum_{m=1}^{M} \alpha_{mk} f_m(x) + \mu_k, \quad k \le K-1, \quad w^*_K(x) = 0,$$

$$\alpha_{mk} \mid \sigma_{k1} \sim \text{normal}(0, \sigma_{k1}), \ k = 1, \dots, K-1, \ m = 1, \dots, d_{\text{discrete}},$$

$$\alpha_{mk} \mid \sigma_{k2} \sim \text{normal}(0, \sigma_{k2}), \ k = 1, \dots, K-1, \ m = d_{\text{discrete}}+1, \dots, d_{\text{discrete}} + d_{\text{continuous}},$$

$$\mu_k \sim \text{normal}(\mu_0, \tau_\mu), \quad \sigma_{k1} \sim \text{normal}^+(0, \tau_{\sigma 1}), \sigma_{k2} \sim \text{normal}^+(0, \tau_{\sigma 2}), \quad k = 1, \dots, K-1.$$

Stan code for hierarchical stacking. Besides advantage listed in this paper, another benefit of stacking now being a Bayesian model is the automated inference in generic computing programs, such as `Stan` (Stan Development Team, 2020). The following `Stan` program is one example of stacking with a linear additive form.

```
data {
  int<lower=1> N; // number of observations
  int<lower=1> d; //number of input variables
  int<lower=1> d_discrete; // number of discrete dummy inputs
  int<lower=2> K;  // number of models
  //when K=2, replace softmax by inverse-logit for higher efficiency
  matrix[N,d] X;   // predictors
  //including continuous and discrete in dummy variables, no constant
  matrix[N,K] lpd_point;  //the input pointwise  predictive density
  real<lower=0> tau_mu;
```

```
  real<lower=0> tau_discrete;//global regularization for discrete x
  real<lower=0> tau_con;//overall regularization for continuous x
}

transformed data {
  matrix[N,K] exp_lpd_point = exp(lpd_point);
}

parameters {
  vector[K-1] mu;
  real mu_0;
  vector<lower=0>[K-1] sigma;
  vector<lower=0>[K-1] sigma_con;
  vector[d-d_discrete] beta_con[K-1];
  vector[d_discrete] tau[K-1]; // using non-centered parameterization
}

transformed parameters {
  vector[d] beta[K-1];
  simplex[K] w[N];
  matrix[N,K] f;
  for (k in 1:(K-1))
    beta[k] = append_row(mu_0*tau_mu + mu[k]*tau_mu + sigma[k]*tau[k],
                         sigma_con[k]*beta_con[k]);
  for (k in 1:(K-1))
    f[,k] = X * beta[k];
  f[,K] = rep_vector(0, N);
  for (n in 1:N)
    w[n] = softmax(to_vector(f[n, 1:K]));
}

model{
  for (k in 1:(K-1)){
    tau[k] ~ std_normal();
    beta_con[k] ~ std_normal();
  }
  mu ~ std_normal();
  mu_0 ~ std_normal();
  sigma ~ normal(0, tau_discrete);
  sigma_con ~ normal(0,tau_con);
  for (i in 1:N)
    target += log(exp_lpd_point[i,] * w[i]); //log likelihood
}

//optional block: needed if an extra layer of LOO (eq.28) is called to
    evaluate the final stacked prediction.
generated quantities {
  vector[N] log_lik;
  for (i in 1:N)
    log_lik[i] = log(exp_lpd_point[i,] * w[i]);
```

```
}
```

To run this stacking program on model fits, we can fit all individual models in Stan, and extract their leave-one-out likelihoods $\{p_{k,-i}\}$. In R, we use the efficient leave-one-out approximation package loo (Vehtari et al., 2020):

```
library("loo")  # https://mc-stan.org/loo/
lpd_point <- matrix(NA, nrow(X), K)
for (k in 1:K) {
  fit_stan <- stan(stan_model = model_k, data = ...)
  # input x may differ in models
  log_lik <- extract_log_lik(fit_stan, merge_chains = FALSE)
  lpd_point[,k] <- loo(log_lik,
                       r_eff = relative_eff(exp(log_lik)))$pointwise
}
```

Finally, we run hierarchical stacking as a regular Bayesian model in Stan.

```
library("rstan")  # https://mc-stan.org/rstan/
# save the stan code above to a  file "stacking.stan".
stan_data <- list(X = X, N=nrow(X), d=ncol(X), d_discrete=d_discrete,
                  lpd_point=lpd_point, K=ncol(lpd_point), tau_mu = 1,
                  tau_sigma = 1, tau_discrete= 0.5, tau_con = 1)
fit_stacking <- stan("stacking.stan",  data = stan_data)
w_fit <- extract(fit_stacking, pars = 'w')$w  # posterior simulation
    of pointwise stacking weights.
```

Appendix D: Prior recommendations

We believe the prior specification should follow the general principle of the weakly-informative prior[2]. In the context of the additive model (Section 2.4), some weakly-informative prior heuristics imply

- We would like to use a half-normal instead of a too wide half-Cauchy or inverse-gamma for the model-wise scale parameter (i.e., $\sigma_k \sim \text{normal}^+(0, \tau_\sigma)$). This is not only because generally, we prefer half-normal for its lighter right tail in hierarchical models, but also because we know that the complete-pooling stacking ($\sigma_k \equiv 0$) is often a rational solution in many problems, to begin with.

  On the contrary, a wide $\sigma_k^2 \sim \text{InvGamma}(10^{-2}, 10^{-4})$ seems a popular choice in the mixture of experts, which we do not recommend.

- When the number of features M is large, it is sensible to first standardize feature such that $\text{Var}(f_m(x)) = 1,\ 1 \le m \le M$, and scale the hyper-parameter to control $\text{Var}(\sum_{m=1}^M \alpha_{mk} f_m(x))$. With independent inputs, it leads to $\tau_\sigma = \mathcal{O}(\sqrt{1/M})$.

- When there are a small number of features and no extra information to incorporate, we often first standardize all features and use a half-normal$(0, 1)$ prior on model-wise scale σ_k (i.e., $\tau_\sigma := 1$). The half-normal$(0, 1)$ has been used as a default informative prior for group-level scale in some applied regression tasks.

- The structure of the prior matters more than the scale of the prior. Hierarchical stacking is typically not sensitive to the difference between a half-normal$(0, 1)$ or half-normal$(0, 2)$ hyper-prior on σ_k, although this sensitivity can be checked. But it would be sensitive to the structure of priors, such as feature-model decomposition, correlated priors, and horseshoe priors, as we have discussed in Section 2.4.

Second, instead of recommending a static default prior, we would rather adopt the attitude that the prior is part of the model and can be checked and improved. Because of our full-Bayesian formulation of hypercritical stacking, we do not have to reinvent model checking tools. When there are concerns on the prior specification, we would like to run prior predictive checks, sensitivity analysis by influence function or importance sampling, and select, stacking, or hierarchically stack a sequence of priors based upon an extra layer of (approximate) leave-one-out cross validation (28).

Choice of features and exploratory data analysis

How to construct features $f_m(x)$ on which model weights can vary is variable selection problem in a regression (13). In ordinary statistical modeling, we often start variable selection by exploratory data analysis. Here we cannot directly associate model weights w_{ki} with observable quantities. Nevertheless, we can use the paired pointwise log predictive density difference $\Delta_{ki} = (\log p_{k,-i} - \log p_{K,-i})$ as an exploratory approximation to the trend of $\alpha_k(x_i)$. A scatter plot of Δ_{ki} against x may suggest which margin of x

[2]For example, see https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations.

is likely important. For example, the dependence of Δ_{ki} on whether x_i is in the bulk or tail is an evidence for our previous recommendation of the rectified features.

As more variables x are allowed to vary in the stacking model, model averaging is more prone to over-fitting. Pointwise stacking typically has a large noise-to-signal ratio not only due to model similarity, but also a high variance of pointwise model evaluation: the approximate leave-one-out cross validation possesses Monte Carlo errors; even if we run exact leave-one-out, or use an independent validation set in lieu of leave-one-out, we only observe one y_i for one x_i (if x is continuous) such that $\log p_{k,-i}$ is at best an one-sample-estimate of $\mathbb{E}_{\tilde{y}|x_i}(\log p(\tilde{y}|x_i, M_k))$ with non-vanishing variance. If $f_m(\cdot)$ is flexible enough, then the sample optimum of no-pooling stacking (7) always degenerates to pointwise model selection that pointwisely picks the model that "best" fits current realization of y_i: $w_{\arg\max_k p_{k,-i}}(x_i) = 1$, which is purely over-fitting.

Even in companion with hierarchical priors, we do not expect to include too many features on which stacking weights depend on. In our experiments, an additive model with discrete variables and rectified continuous variables without interaction is often adequate. After standardizing all features such that $\mathrm{Var}(f_m(x)) = 1$, we typically use a generic informative prior setting $\tau_\mu = \tau_\sigma = 1$ in experiments. With a moderate or large number of features/cells, M, it is sensible to scale the hyperprior $\tau_\sigma = \mathcal{O}(\sqrt{1/M})$, or adopt other feature-wise shrinkage priors such as horseshoe for better regularization.

Appendix E: Experiment details

The replication code for experiments is available at
https://github.com/yao-yl/hierarchical-stacking-code.

Well-switch. Vehtari et al. (2017) and Gelman et al. (2020) used the same pointwise pattern (first panel in Figure 2) in our well-switch example to demonstrate the heterogeneity of model fit. The input contains both continuous $x_{\text{con}} \in \mathbb{R}^D$ and categorical $x_{\text{cat}} \in \{1, \ldots, 8\}$. As per previous discussion (16), we convert all continuous inputs x_{con} into two parts $x^+_{\text{con},j} := (x_{\text{con},j} - \text{median}(x_{\text{con},j}))_+$ and $x^-_{\text{con},j} := (x_{\text{con},j} - \text{median}(x_{\text{con},j}))_-$. We then model the unconstrained weight by a linear regression

$$\alpha_k(x) = \sum_{j=1}^{D} \left(\beta_{2j-1,k} x^+_{\text{con},j} + \beta_{2j,k} x^-_{\text{con},j}\right) + z_k[x_{\text{cat}}],\ k = 1, \ldots, 4; \quad \alpha_5(x) = 0. \tag{29}$$

And place a default prior on parameters and hyper-parameters.

$$z_k[j] \sim \text{normal}(\mu_k, \sigma_k), \quad \beta_j, \mu_k \sim \text{normal}(0, 1), \quad \sigma_k \sim \text{normal}^+(0, 1).$$

Gaussian process regression. We use training data $\{x_i, y_i\}$ from Neal (1998) (file `odata.txt` in our repo). Yao et al. (2020) use same setting to explain the benefit of complete-pooling stacking. The training size is $n = 100$. We generate additional test data for model evaluation. The univariate input x is distributed normal$(0, 1)$, and the corresponding outcome y is also Gaussian. The true but unknown conditional mean is

$$\mathbb{E}_{\text{true}}(y|x) = f_{\text{true}}(x) = 0.3 + 0.4x + 0.5\sin(2.7x) + 1.1/(1 + x^2).$$

In the data generating process, with probability 0.95, y is a realization from $y|f_{\text{true}} = \text{normal}(\text{mean} = f_{\text{true}},\ \text{sd} = 0.1)$. With probability 0.05, y is considered an outlier and the standard deviation is inflated to 1: $y|f_{\text{true}} = \text{normal}(f_{\text{true}}, 1)$. This outlier probability is independent of location x, and the observational noises are mutually independent.

To infer the parameter $\theta = (a, \rho, \sigma)$ in the first level GP model

$$y_i = f(x_i) + \epsilon_i,\ \epsilon_i \sim \text{normal}(0, \sigma),\ f(x) \sim \mathcal{GP}\left(0, a^2 \exp\left(-\frac{(x - x')^2}{\rho^2}\right)\right).$$

We integrate out all local $f(x_i)$ and obtain the marginal posterior distribution $\log p(\theta|y) = -\frac{1}{2} y^T \left(K(x, x) + \sigma^2 I\right)^{-1} y - \frac{1}{2}\log|K(x, x) + \sigma^2 I| + \log p(\theta) + \text{constant}$, where K is squared-exponential-kernel, and $p(\theta)$ is the prior for which we choose an elementwise half-Cauchy$(0, 3)$. Using initialization $(\log\rho, \log a, \log\sigma) = (1, 0.7, 0.1)$ and $(-1, -5, 2)$ respectively, we find two posterior modes of hyper-parameter $\theta = (a, \rho, \sigma)$.

The posterior multimodality relies on the particular realization of x and y. We have tried other randomly generated training datasets, among which only Neal (1998)'s original data realization can give rise to two distinct modes. We then consider three standard mode-based approximate inference:

- Type-II MAP: The value $\hat{\theta}$ that maximizes the marginal posterior distribution. We further draw $f|\hat{\theta}, y$.
- Laplace approximation. First compute Σ: the inverse of the negative Hessian matrix of the log posterior density at the local mode $\hat{\theta}$, draw z from $\text{MVN}(0, I_3)$, and use $\theta(z) = \hat{\theta} + \text{V}\Lambda^{1/2}z$ as the approximate posterior samples around the mode $\hat{\theta}$, where the matrices V, Λ are from the eigen-decomposition $\Sigma = \text{V}\Lambda^{1/2}\text{V}^T$.
- Importance resampling. First draw z from uniform$(-4, 4)$, resample z without replacement with probability proportional to $p\left(\theta(z)|y\right)$, and use the kept samples of $\theta(z)$ as an approximation of $p(\theta|y)$.

With two local modes $\hat{\theta}_1, \hat{\theta}_2$, we either obtain two MAPs, or two nonoverlapped draws, $(\theta_{1s})_{s=1}^S, (\theta_{2s})_{s=1}^S$. We evaluate the predictive distribution of f, $p_k(f|y,\theta) = \int p(f|y,\theta)q(\theta|\hat{\theta}_k)d\theta,\ k = 1, 2$, where $q(\theta|\hat{\theta}_k)$ is a delta function at the mode $\hat{\theta}_k$, or the draws from the Laplace approximation and importance resampling expanded at $\hat{\theta}_k$.

In the model averaging phase, we form the model weight in GP prior stacking by

$$w_1(x) = \text{invlogit}(\alpha(x)),\ \alpha(x) \sim \mathcal{GP}(0, \mathcal{K}(x)),\ \mathcal{K}(x_i, x_j) = a\exp(-\left((x_i - x_j)/\rho\right)^2).$$

Because input x is distributed normal$(0, 1)$, the length scale ρ should be constrained on a similar scale. We use the following hyperprior for GP prior stacking:

$$\rho \sim \text{Inv–Gamma}(4, 1), \quad a \sim \text{normal}(0, 1).$$

The Inv–Gamma$(4, 1)$ prior puts 98% of mass on the interval $0.1 < \rho < 1.2$.

Election polling. In the election example, we conduct a back-test for one-week-ahead forecasts. For example, if there are 20 polls between Aug 1 and Aug 7, we first fit each model on the data prior to Aug 1 and forecast for each of the 20 polls in this week. Next, we move on to forecasting for the week between Aug 8 and Aug 14. We use this step-wise approach for both, fitting the candidate models and stacking.

We use two variants of hierarchical stacking with discrete inputs—first with independent priors from Eq. (9) and second with correlated priors from Eq. (15). We place default priors on the hyperparameters in both variants:

$$\mu_k \sim \text{normal}(0, 1), \quad \sigma_k \sim \text{normal}^+(0, 1).$$

We are evaluating all combining methods on the same data, therefore we can compare them pointwisely by selecting a reference model—in our case this is the proposed hierarchical stacking and set it to be zero in all visualisations. For each combination method and each poll i, we compute the pointwise difference in elpds: $\text{elpd\_diff}_i^{\text{M}_j} = \text{elpd}_i^{\text{M}_j} - \text{elpd}_i^{\text{M}_{\text{ref}}}$, where M_j is the j-th model and M_{ref} is the reference model. Then we report the mean of this differences over all polls in the test data, $\text{elpd\_diff}^{\text{M}_j} = \frac{1}{N}\sum_i \text{elpd\_diff}_i^{\text{M}_j}$, where N is the number of all polls.

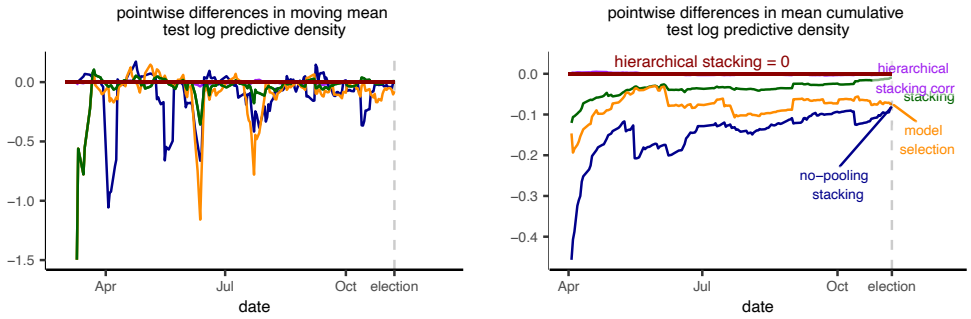

Figure 13: *Same pointwise model comparisons as in Figure 7, except this time all model averaging and selection methods are trained using the previous 60 days rather than 4 weeks on each backtesting day. The pattern remains similar.*

In the main text, to account for non-stationarity discussed in Section 2.5, we only use the last four weeks prior to prediction day for training model averaging. In the end we obtain a trajectory of this back-testing performance of hierarchical stacking, complete-pooling, and no-pooling stacking and single model selection. The time window of four-week is a relatively ad-hoc choice and we did not tune it. Figure 13 displays the result when trained with the previous 60 days rather than four weeks on each backtesting day. The pattern remains similar.

Additionally, we are interested in how models perform depending on time, as there are few polls available in the early days of the election year, and then their number continuously increases toward election day. This results in noisier observations in the beginning. To suitably evaluate the combining methods, we compute the cumulative mean elpd at each day d, $\text{elpd}_d^{*,\text{M}_j} = \frac{1}{N_d}\sum_{j\le d}\text{elpd}_j^{\text{M}_j}$, where N_d is the number of conducted polls prior to or on day d. Then we compute the pointwise differences between these cumulative mean elpds of each method and the reference method: $\text{elpd\_diff}_d^{*,\text{M}_j} = \text{elpd}_d^{*,\text{M}_j} - \text{elpd}_d^{*,\text{M}_{\text{ref}}}$.

To get the elpd of a state, we take the average of all elpds in that state, for example $\text{elpd}_{\text{NY}} = \frac{1}{N_{\text{NY}}}\sum_{i\in A_{\text{NY}}}\text{elpd}_i$, where N_{NY} is the number of polls conducted in New York, and A_{NY} is the set of indexes of polls in New York. Figure 14 displays the state-level log predictive density of the combined model in six representative states.

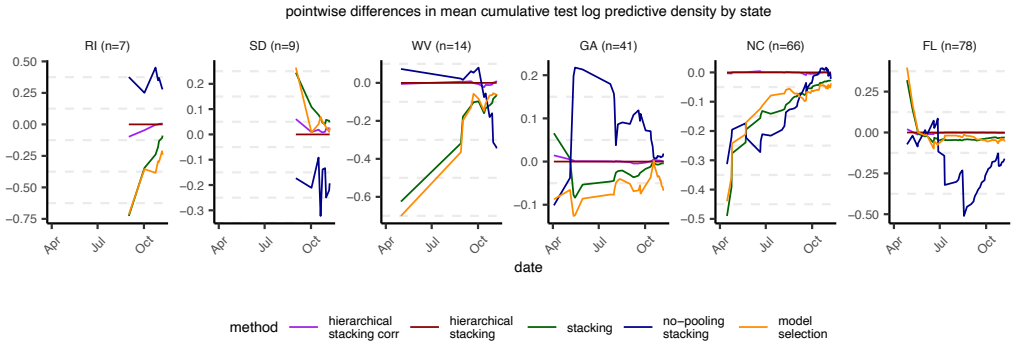

Figure 14: *Log predictive density of the combined model in three small states (in the number of available state polls n, RI, SD, WV) and three swing states (GA, NC, FL). We fix the uncorrelated hierarchical stacking to be constant zero as a reference. The number in the bracket is the total number of polls in that state. With a large number of state polls available, for example, close to election day in Florida and North Carolina, no-pooling stacking performs well. With fewer polls, no-pooling stacking is unstable, as can be seen in Rhode Island, South Dakota, West Virginia, and the early part of Georgia plots. Hierarchical stacking alleviates this instability, while retaining enough flexibility for a good performance with large data come in.*

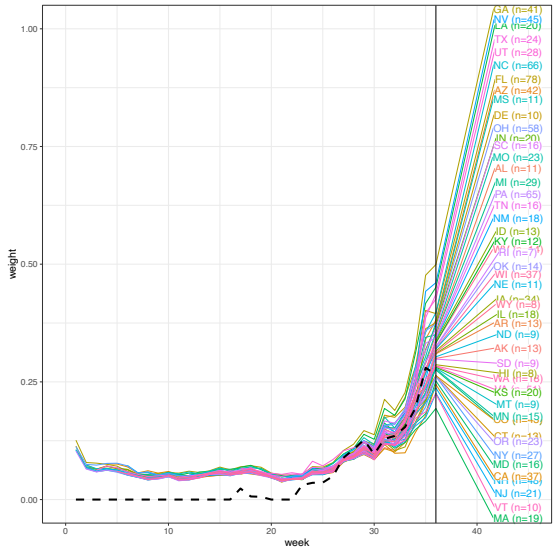

Figure 15: *In the election pooling example, the posterior mean of hierarchical stacking weight of Model 1 (the fundamental model) in different time and states. Apart from improving predictions, this input-varying stacking weight is also informative for model understanding: which and when states are more sensitive to macro fundamentals.*