BAGGING PSEUDO CODE

Bagging algorithm: Training

```
Input: Dataset D, type of BLs, number of bootstraps M
for m = 1 → M do
    Draw a bootstrap sample D^[m] from D
    Train BL on D^[m] to obtain model b̂^[m]
end for
```

- **Input:** Dataset $\mathcal{D}$, type of BLs, number of bootstraps M
- **for** $m = 1 \to M$ **do**
  - Draw a bootstrap sample $\mathcal{D}^{[m]}$ from $\mathcal{D}$
  - Train BL on $\mathcal{D}^{[m]}$ to obtain model $\hat{b}^{[m]}$
- **end for**

Bagging algorithm: Prediction

- **Input:** Obs. $\mathbf{x}$, trained BLs $\hat{b}^{[m]}$ (as scores $\hat{f}^{[m]}$, hard labels $\hat{h}^{[m]}$ or probs $\hat{\pi}^{[m]}$)
- Aggregate/Average predictions

$$\hat{f}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M}\left(\hat{f}^{[m]}(\mathbf{x})\right)$$ (regression / decision score, use $\hat{f}_k$ in multi-class)

$$\hat{h}(\mathbf{x}) = \arg\max_{k \in \mathcal{Y}} \sum_{m=1}^{M} \mathbb{I}\left(\hat{h}^{[m]}(\mathbf{x}) = k\right)$$ (majority voting)

$$\hat{\pi}_k(\mathbf{x}) = \begin{cases} \frac{1}{M}\sum_{m=1}^{M}\hat{\pi}_k^{[m]}(\mathbf{x}) & \text{(probabilities through averaging)} \\ \frac{1}{M}\sum_{m=1}^{M}\mathbb{I}\left(\hat{h}^{[m]}(\mathbf{x}) = k\right) & \text{(probabilities through class frequencies)} \end{cases}$$

MINI BENCHMARK

- Bagged ensembles with 100 BLs each on `spam`:
  Bagging seems especially helpful for less stable learners like CART