Top-down particle filtering for Bayesian decision trees

Balaji Lakshminarayanan\textsuperscript{1}, Daniel M. Roy\textsuperscript{2} and Yee Whye Teh\textsuperscript{3}

Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Introduction

Input: attributes $X = \{x_i\}_{i=1}^N$, labels $Y = \{y_i\}_{i=1}^N$ (i.i.d)

$y_i \in \{1, \ldots, K\}$ (classification) or $y_i \in \mathbb{R}$ (regression)

Goal: Model $p(y|x)$
Input: attributes $X = \{x_i\}_{i=1}^N$, labels $Y = \{y_i\}_{i=1}^N$ (i.i.d)

$y_i \in \{1, \ldots, K\}$ (classification) or $y_i \in \mathbb{R}$ (regression)

Goal: Model $p(y|x)$

Assume $p(y|x)$ is specified by decision tree $T$

Bayesian decision trees:

- Posterior: $p(T|Y, X) \propto p(Y|T, X) p(T|X)

- Prediction: $p(y^*|x^*) = \sum_T p(T|Y, X)p(y^*|x^*, T)$
Example: Classification tree

\[ x_1 > 0.5 \]
\[ x_2 > 0.35 \]

\( \theta \): Multinomial parameters at leaf nodes
Example: Regression tree

\[ \theta_0 \]

\[ x_1 > 0.5 \]

\[ \theta_{10} \]

\[ x_2 > 0.35 \]

\[ \theta_{11} \]

\[ \theta : \text{Gaussian parameters at leaf nodes} \]
Motivation

- Classic non-Bayesian induction algorithms (e.g. CART) learn a single tree in a top-down manner using greedy heuristics (post-pruning and/or bagging necessary)
- MCMC for Bayesian decision trees: [Chipman et al., 1998]: local Monte Carlo modifications to the tree structure (less prone to over fitting but slow to mix)
- **Our contribution:** Sequential Monte Carlo (SMC) algorithm that approximates the posterior, in a top-down manner
- **Take home message:** SMC provides better computation vs predictive performance tradeoff than MCMC
Bayesian decision trees: likelihood

$$p(T|Y, X) \propto p(Y|T, X) p(T|X)$$
Likelihood

- Assume \( x_n \) falls in the \( j^{th} \) leaf node of \( \mathcal{T} \)
- Likelihood for \( n^{th} \) data point:  
  \[
p(y_n \mid x_n, \mathcal{T}, \theta) = p(y_n \mid \theta_j, x_n)
  \]

\[
p(Y \mid \mathcal{T}, X, \Theta) = \prod_{n} p(y_n \mid x_n, \mathcal{T}, \theta) = \prod_{j \in \text{leaves}(\mathcal{T})} \prod_{n \in N(j)} p(y_n \mid \theta_j)
\]

Better: integrate out \( \theta_j \), use marginal likelihood

Classification: Dirichlet - Multinomial
Regression: Normal - Normal Inverse Gamma
Likelihood

- Assume $x_n$ falls in the $j^{th}$ leaf node of $T$
- Likelihood for $n^{th}$ data point: $p(y_n | x_n, T, \theta) = p(y_n | \theta_j, x_n)$

$$p(Y | T, X, \Theta) = \prod_{n} p(y_n | x_n, T, \theta) = \prod_{j \in \text{leaves}(T)} \prod_{n \in N(j)} p(y_n | \theta_j)$$

- Better: integrate out $\theta_j$, use marginal likelihood

$$p(Y | T, X) = \prod_{j \in \text{leaves}(T)} \int_{\theta_j} \prod_{n \in N(j)} p(y_n | \theta_j) p(\theta_j) d\theta_j$$

- Classification: Dirichlet - Multinomial
- Regression: Normal - Normal Inverse Gamma
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Bayesian decision trees: prior

\[ p(T \mid Y, X) \propto p(Y \mid T, X) \cdot p(T \mid X) \]

\( \text{likelihood} \quad \text{prior} \)
Partial trees

0. Start with empty tree.
1. Choose to split root node with feature 1 and threshold 0.5.

\[ \epsilon : x_1 > 0.5 \]
Partial trees

2. Choose to not split node 0.

\[ \epsilon : x_1 > 0.5 \]
3. Choose to split node 1 with feature 2 and threshold 0.35.
4. Choose to not split node 10.
5. Choose to not split node 11.
Sequence of random variables for a tree

1. $\rho_\epsilon = 1, \kappa_\epsilon = 1, \tau_\epsilon = 0.5$
2. $\rho_0 = 0$
3. $\rho_1 = 1, \kappa_1 = 2, \tau_1 = 0.35$
4. $\rho_{10} = 0$
5. $\rho_{11} = 0$
.Sequential prior over decision trees

- Probability of split (assuming a valid split exists):

\[ p(j \text{ split}) = \alpha_s \cdot \left(1 + \text{depth}(j)\right)^{-\beta_s} \quad \alpha_s \in (0, 1), \quad \beta_s \in [0, \infty) \]

- \( \kappa_j, \tau_j \) sampled uniformly from the range of valid splits
Sequential prior over decision trees

- Probability of split (assuming a valid split exists):
  \[ p(j \text{ split}) = \alpha_s \cdot \left(1 + \text{depth}(j)\right)^{-\beta_s} \]
  \[ \alpha_s \in (0, 1), \quad \beta_s \in [0, \infty) \]

- \( \kappa_j, \tau_j \) sampled uniformly from the range of valid splits

- Prior distribution:
  \[ p(T, \kappa, \tau | X) = \prod_{j \in \text{leaves}(T)} p(j \text{ not split}) \]
  \[ \times \prod_{j \in \text{nonleaves}(T)} p(j \text{ split}) p(\kappa_j, \tau_j) \]
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Bayesian decision trees: posterior

\[ p(T|Y, X) \propto p(Y|T, X) p(T|X) \]

\[ \text{likelihood} \quad \text{prior} \]
SMC algorithm for Bayesian decision trees

- Importance sampler: Draw $T^{(c)} \sim q(\cdot)$

\[
p(Y|X) = \sum_{T} p(Y, T|X) \approx \sum_{c=1}^{C} \frac{1}{C} \frac{p(T^{(c)})}{q(T^{(c)})} p(Y|X, T^{(c)}) w^{(c)}
\]
SMC algorithm for Bayesian decision trees

- Importance sampler: Draw $T^{(c)} \sim q(\cdot)$

$$p(Y|X) = \sum_{T} p(Y, T|X) \approx \sum_{c=1}^{C} \frac{1}{C} \frac{p(T^{(c)})}{q(T^{(c)})} \frac{p(Y|X, T^{(c)})}{w^{(c)}}$$

- Normalize: $\bar{w}^{(c)} = \frac{w^{(c)}}{\sum_{c'} w^{(c')}}$

- Approximate posterior:

$$p(T|Y, X) \approx \sum_{c} \bar{w}^{(c)} \delta(T = T^{(c)})$$
SMC algorithm for Bayesian decision trees (contd.)

- Sequential importance sampler (SIS):

\[
p(T_n) = p(T_0) \prod_{n'=1}^{n} p(T_{n'}|T_{n'-1}) \\
q(T_n) = q_0(T_0) \prod_{n'=1}^{n} q_{n'}(T_{n'}|T_{n'-1})
\]

\[
p(Y|X, T_n) = p(Y|X, T_0) \frac{p(Y|X, T_1)}{p(Y|X, T_0)} \cdots \frac{p(Y|X, T_n)}{p(Y|X, T_{n-1})}
\]
SMC algorithm for Bayesian decision trees (contd.)

- Sequential importance sampler (SIS):

\[
p(T_n) = p(T_0) \prod_{n'=1}^{n} p(T_{n'}|T_{n'-1}) \quad q(T_n) = q_0(T_0) \prod_{n'=1}^{n} q_{n'}(T_{n'}|T_{n'-1})
\]

\[
p(Y|X, T_n) = \frac{p(Y|X, T_0) p(Y|X, T_1)}{p(Y|X, T_0)} \cdots \frac{p(Y|X, T_n)}{p(Y|X, T_{n-1})}
\]

\[
w = \frac{1}{C} \frac{p(T_n)}{q(T_n)} p(Y|X, T_n)
\]

\[
= w_0 \prod_{n'=1}^{n} \frac{p(T_{n'}|T_{n'-1})}{q_{n'}(T_{n'}|T_{n'-1})} \frac{p(Y|X, T_{n'})}{p(Y|X, T_{n'-1})}
\]

- Sequential Monte Carlo (SMC): SIS + adaptive resampling steps
- Every node is processed just once: no multi-path issues
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Experimental setup

- Datasets:

- 70% - 30% train-test split

- Numbers averaged across 10 different initializations
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
SMC design choices

- Proposals
  - *prior* proposal: \( q_n(\rho_j, \kappa_j, \tau_j) = p(\rho_j, \kappa_j, \tau_j) \)
  - *optimal* proposal:
    \[
    q_n(\rho_j = \text{stop}) \propto p(\text{not split})p(Y_{N(j)}|X_{N(j)}),
    
    q_n(\rho_j = \text{split}, \kappa_j, \tau_j) \propto p(\text{split})p(\kappa_j, \tau_j) 
    \times p(Y_{N(j0)}|X_{N(j0)}) p(Y_{N(j1)}|X_{N(j1)}) .
    \]

- Set of nodes considered for expansion at iteration \( n \)
  - *node-wise*: next node
  - *layer-wise*: all nodes at depth \( n \)

- Multinomial resampling
Effect of SMC design choices

Figure: Results on *magic-04* dataset
Effect of irrelevant features on SMC design choices

*madelon*: $N = 2.6K$, $D = 500$, $K = 2$

(96% of the features are irrelevant)

Figure: Results on *madelon* dataset
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Predictive performance vs computation: SMC vs MCMC

- Fix hyper parameters $\alpha = 5$, $\alpha_s = 0.95$, $\beta_s = 0.5$
- MCMC [Chipman et al., 1998]: one of the 4 proposals:
  - $grow$
  - $prune$
  - $change$
  - $swap$
- MCMC averages predictions over all previous trees
- Vary number of particles in SMC, number of MCMC iterations and compare runtime vs performance
Predictive performance vs computation: SMC vs MCMC

Figure: Results on magic-04 dataset
SMC (prior, node-wise) is at least an order of magnitude faster than MCMC
Outline

Introduction

Sequential prior over decision trees

Bayesian inference: Top-down particle filtering

Experiments
  Design choices in the SMC algorithm
  SMC vs MCMC

Conclusion
Conclusion

- SMC for fast Bayesian inference for decision trees
  - mimick the top-down generative process of decision trees
  - use ‘local’ likelihoods + resampling steps to guide tree growth
  - For a fixed computational budget, SMC outperforms MCMC
Conclusion

- SMC for fast Bayesian inference for decision trees
  - mimick the top-down generative process of decision trees
  - use ‘local’ likelihoods + resampling steps to guide tree growth
  - For a fixed computational budget, SMC outperforms MCMC

- Future directions
  - Particle-MCMC for Bayesian Additive Regression Trees
  - Mondrian process prior: projective and exchangeable prior for decision trees [Roy and Teh, 2009]
Thank you!

Code available at
http://www.gatsby.ucl.ac.uk/~balaji