Interpretable ML

Variable Importance

Introduction

Algorithm 1: Grow a Random Forest

```
Set number of trees B;
Set predictor subset size m;
Define stopping criteria;
for b = 1 to B do
    draw a bootstrap sample from the training data;
    assign sampled data to root node;
    if stopping criterion is reached then
        end splitting;
    else
        draw a random sample m from the p predictors;
        find the optimal split point among m;
        split node into two subnodes at this split point;
        for each node of the current tree do
            continue tree growing process;
        end
    end
end
```

Interpretable ML

Interpreting Random Forests

- Inspect each tree of the forest
  - Inefficient for 500+ trees
- Variable importance
  - Summary of "effect size"
- Partial dependence plots
  - Graphical representation of "effect structure"
- ...

Variable Importance

Variable importance with CART

$$\mathcal{I}_\ell^2(T) = \sum_{t=1}^{J-1} \hat{\imath}_t^2 I(v(t) = \ell)$$

- Sum of squared improvements $\hat{\imath}^2$ over all internal nodes with predictor X_ℓ
  - Regression: Overall reduction in RSS caused by X_ℓ
  - Classification: Overall reduction of impurity caused by X_ℓ

Importance with Random Forests

$$\mathcal{I}_\ell^2 = \frac{1}{M} \sum_{m=1}^{M} \mathcal{I}_\ell^2(T_m)$$

- Average improvement caused by predictor X_ℓ over all trees

Variable Importance

Permutation feature importance (Fisher et al. 2018)

1. Estimate the original model error $e_{orig}(\hat{f}) = L(Y, \hat{f}(X))$
2. For each feature $j \in 1, \ldots, p$
    1. Generate feature matrix $X_{perm\,j}$ by permuting the values of feature X_j in X
    2. Estimate error $e_{perm} = L(Y, \hat{f}(X_{perm\,j}))$ based on the predictions of the permuted data
    3. Calculate permutation feature importance $FI_j = \frac{e_{perm}(\hat{f})}{e_{orig}(\hat{f})}$ or via $FI_j = e_{perm}(\hat{f}) - e_{orig}(\hat{f})$
3. Output FI for all variables