Preventing Overfitting in Decision Trees

Deeper trees ⇒ lower training error

Training error = 0: Is this model perfect?

Not perfect!

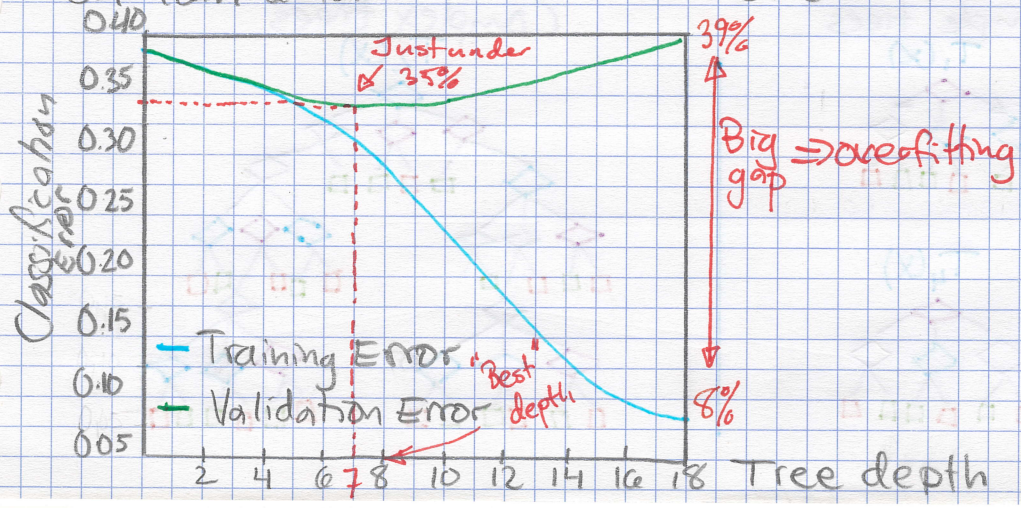
Why training error reduces with depth?

Feature split selection algorithm

- Given a subset of data M (a node in a tree)
- For each feature $h_i(x)$:
  1.) Split data of M according to feature $h_i(x)$
  2) Compute classification error split
- Chose feature $h^*(x)$ with lowest classification error

By design, each split reduces training error

Decision trees overfitting on loan data



Principle of Occam's razor: Simpler trees are better

Symptoms: S_1 and S_2

Diagnosis 1: 2 diseases
Two diseases D_1 and D_2 where
D_1 explains S_1, D_2 explains S_2

OR

Diagnosis 2: 1 disease
Disease D_3 explains both symptoms S_1 and S_2

} Simpler

Occam's Razor for decision trees
When two trees have similar classification error on the validation set, pick the simpler one.

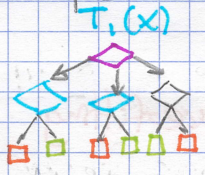
Complexity	Train Error	Validation Error	
Simple	0.23	0.24	
Moderate	0.12	0.15	Same validation error
Complex	0.07	0.15	
Super complex	0 ← bad!	0.18	→ Overfit

pick ↑

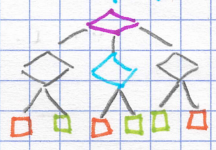
Modified tree learning problem
Find a "simple" decision tree with low classification error

Simple trees

$T_1(X)$

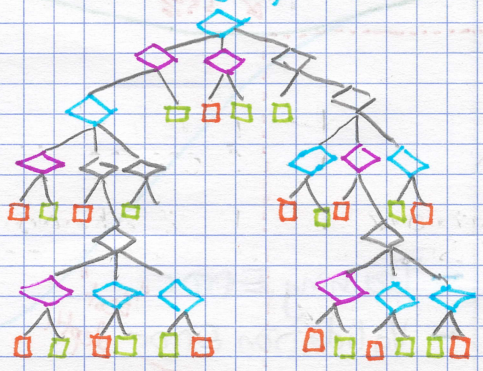


$T_4(X)$



Complex trees

$T_2(X)$



How do we pick simpler trees?

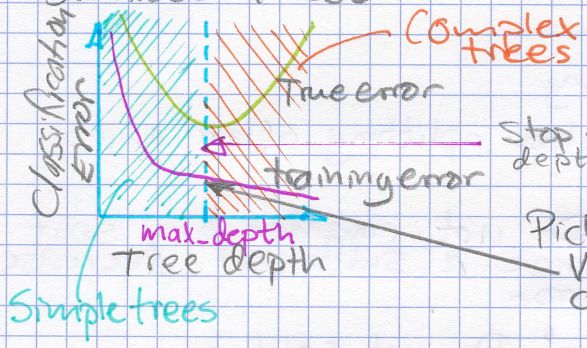
1. **Early Stopping**: Stop learning algorithm *before* tree become too complex
2) **Pruning**: Simplify tree *after* learning algorithm terminates

Early Stopping in learning decision trees

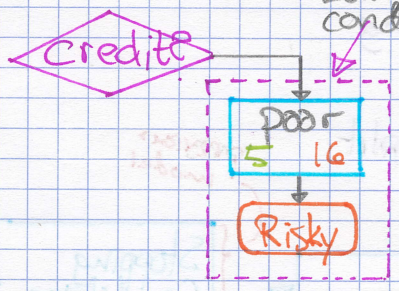
Deeper trees → Increasing complexity

Early stopping condition 1: *Limit the depth of a tree*

Restrict tree learning to shallow trees



Early stopping condition 2: *Use classification error to limit depth of tree*



No split improves classification error → Stop!

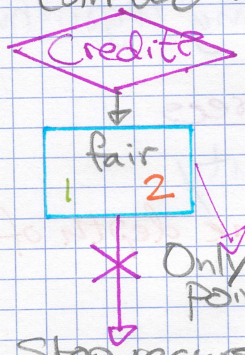
Splits for Credit=poor	Classification error
no split	0.24
split on term	0.24
split on income	0.24

Practical note about stopping when classification error doesn't decrease

1) Typically, add magic parameter ϵ
   - Stop if error doesn't decrease more than ϵ
2) Some pitfalls to this rule (see pruning)
3) Very useful in practice

Early stopping condition 3. Stop if number of data points contained in a node is too small

Can we trust nodes with very few points?



Stop when data points in a node $<= N_{min}$

Early stopping summary

1) **Limit tree depth** Stop splitting after a certain depth
2) **Classification error** Do not consider any split that does not cause a sufficient decrease in classification error
3) **Minimum node "size"** Do not split an intermediate node which contains too few data points

Greedy decision tree learning

- Step1: Start with an empty tree
- Step2: Select a feature to split data
- For each split of the tree
  - Step3: If nothing more to, make predictions (majority)
  - Step4: Otherwise, go to Step2 & continue (recurse) on this split

previous model

Stopping Conditions 1 & 2

or

Early stopping conditions 1, 2 & 3

Recursion