ANNOUNCEMENTS —

1. HW6 due next Monday, 5pm / late due, next Wed, 5pm
2. HW3 grades are (re)posted

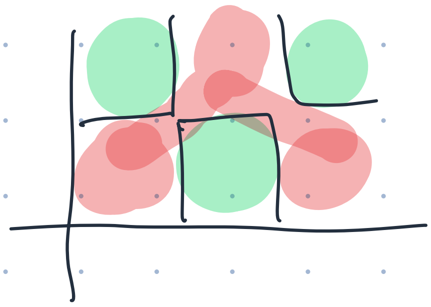
Random thought: When do you think "group" exams would succeed?

(a) if everyone in the group was an expert in ONE topic | (b) more of a "jack-of-all-trades" situation?

(Also, Kehlani for Slope day?!)

Recap: Decision trees

Goal: Have a non-linear classifier, akin to region-splitting



yes/no questions

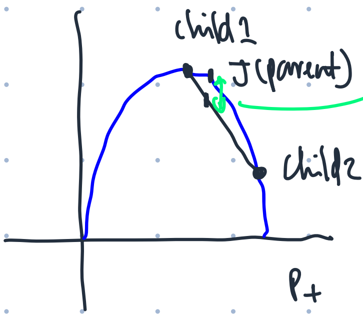
$J(R)$ measures the impurity in a region,

our goal:

$$\max \; J(\text{parent}) - \overset{\text{w. avg}}{J}(\text{children})$$

is latitude $> 30°$

Entropy metric – measures impurity



"info gain"

parent entropy always above w. avg of child entropies

$\Rightarrow$ splitting by entropy always results in lower entropy.

$$J(R) = \sum_{y \in \{-1, 1\}} - P_y \log P_y$$

Alternative to not computing log, Gini impurity:

$$J_{gini}(R) = \sum_{y \in \{+1, -1\}} P_y (1 - P_y)$$

OR SPECIFIC INSTANCES

Today: ① variants of decision trees

② problems with decision trees

③ (As usual,) ask if we can do better?

Categorical attributes

Classify CS3780/5780 students into grad/ugrad classes

If we split on netID, we'd be asking 399 questions.

Data:

NetID
TF352
KS999
WSL92
LJL2
⋮

In the worst case, I'd be asking 2^{400} questions!

↓

splitting on "netId" results in pure nodes

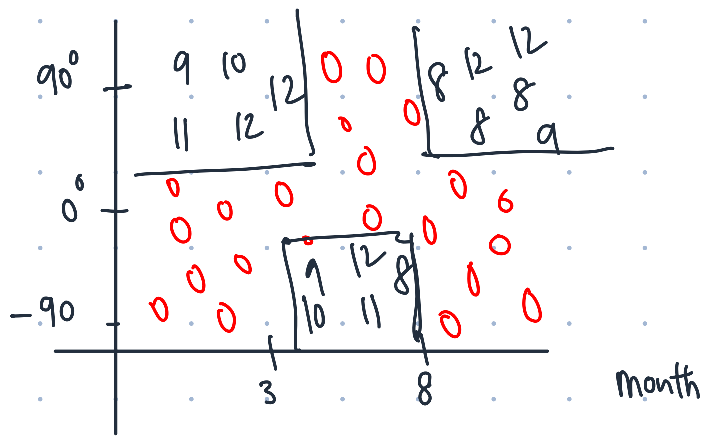
⇒ max decrease in "J"

Takeaway: ⊕ Decision trees kind of give you nice way of handling Categorical attributes

⊖ highly branching attributes are not favorable → computation → overfitting.

Regression trees

Instead of predicting sled/not, we wish to predict snowfall in inches.



① Instead of a majority vote, we will use the avg in the region

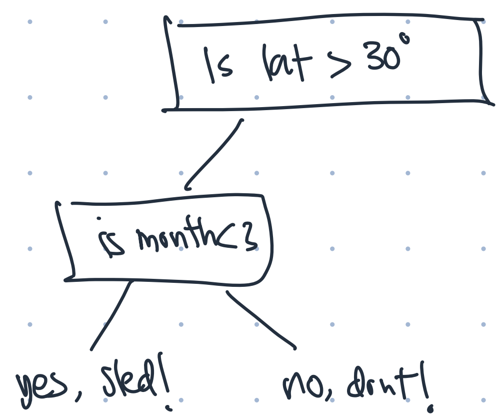
$$\hat{y} = \frac{1}{|R|} \sum_{j \in R} y^{(j)}$$

② "J" → $J_{sq} = \frac{1}{|R|} \sum_{j \in R} (y^{(j)} - \hat{y})^2$

Why does CS3780/5780 like DT?

Rakin — They're non linear!

Jay — Interpretability (for free!!)



Issues with Decision trees — Bias/variance

Q. Do DT suffer from high bias?

⇒ I can ask questions (as many as I want) to get EVERY training pt. correctly classified!

[stopping criteria is that every leaf is a single, pure node.

⇒ LOW BIAS! ← good :)

Q. Do DT suffer from high variance?

overfitting to the training data ⇒ HIGH VARIANCE! ← BAD! :C

⇒ Q. How do we fix this "high variance" problem?

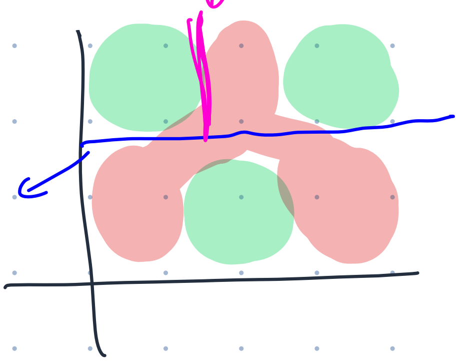
(1) Set a min leaf size — don't split if $|R| < 20$

(2) set a max depth — if tree has > 3 levels, don't grow!

(3) threshold on max nodes — 20 nodes/less in the whole tree

(4) threshold on "info gain" we talked about → Don't

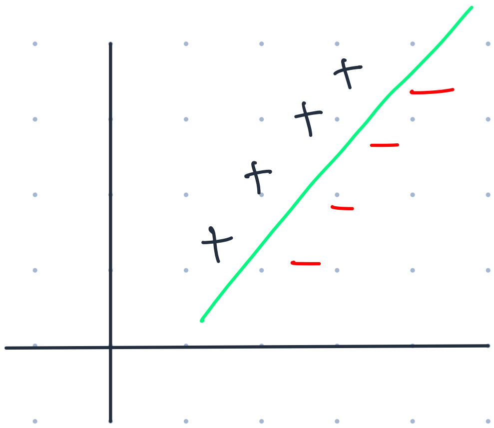
high info gain



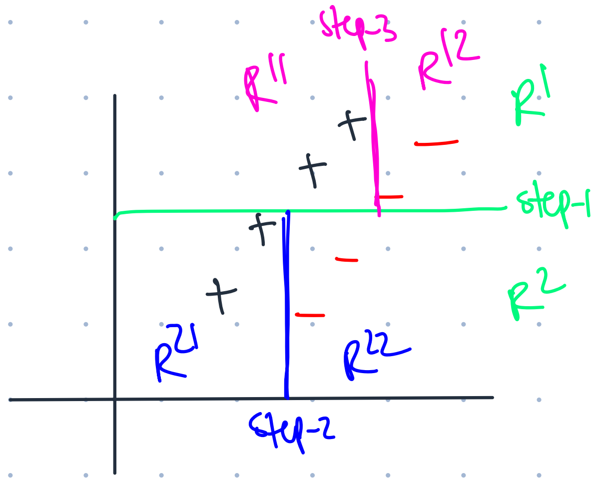
doesn't get me high info gain

Jay: (5) Grow out the whole tree, then remove nodes based on X42 on validation set!

"line drawing"



logistic regression



decision tree

Takeaway: Can't model additive structures

The good, the bad of DECISION TREES

In the wild, DT have poor predictive accuracy

Idea : what if we could train multiple models and use the "aggregate" prediction?

We have "m" total IID random variables, $x^{(j)}$s,

we want to compute

$$Var\left[\frac{1}{m}\sum_{j=1}^{m} x^{(j)}\right] = \frac{\sigma^2}{m}$$, because $x^{(j)}$s are independent

Goal — variance reduction
Idea — as $m\uparrow$ (# predictors) var $\downarrow$ (Y_m)

} $x^{(j)}$ = error rate of my predictor $h^{(j)}$

Is the assumption of independence among predictors reasonable?

→ maybe not always

BREAK THE ASSUMPTION, $Corr(x^{(j)}, x^{(k)}) = \rho$

ID $x^{(j)}$s (not necessarily ind),

$$Var\left[\frac{1}{m}\sum_{j=1}^{m} x^{(j)}\right] = \sigma^2\rho + (1-\rho)\frac{\sigma^2}{m}.$$

$\rho = 0$ when decorrelated $\frac{\sigma^2}{m}$

$\rho = 1$ we have σ^2

Ways to ensemble?

Ensemble "m" predictors & hope var goes down!

1) Train different algorithms — SVM, DT, LR, NB — aggregate them!

Computationally expensive!

2) Go and collect different datasets — D^1, D^2, D^3 — train SVM on D^1, D^2, D^3 seperately, aggregate

Could be infeasible

similar to this, but doesn't require additional data collection! } "Bagging"
(Eg. RF)

Ensembling from bias-reduction perspection = "boosting"
(Eg. AdaBoost, XGBoost)

BOOTSTRAPPING + AGGREGATION (AKA BAGGING)

Idea – If we had $D^1, D^2, D^3, \ldots, D^m$, we could train a model on each of these seperately, then aggregate!

$D = \{(x^{(j)}, y^{(j)}) \mid 1 \leq j \leq n\}$, $(x^{(j)}, y^{(j)}) \sim \mathcal{P}$ ← true distribution

1. Bootstrapping: we assume $\underline{\underline{D = \mathcal{P}}}$ ⇒ we can sample from D, more imp, we can sample as many times as we want.

→ Sample, WITH REPLACEMENT, 'n' samples, to get $Z^{(1)}$

matt – if we are assuming $D = \mathcal{P}$, sampling w/ repl makes sense for the assumpt. to hold!

Repeat to get $Z^{(2)}, Z^{(3)}, \ldots, Z^{(m)}$

2. Aggregation: Given $Z^{(1)}, \ldots, Z^{(m)}$, train some model on each, to get $h^{(1)}, \ldots, h^{(m)}$

aggregate hypothesis

$$h(x) = \frac{1}{m} \sum_{j=1}^{m} h^{(j)}(x)$$

Jay →

Niceties of BAGGING!

(1) Variance of average error of "m" correlated models,

$$\mathrm{Var}\left[\frac{1}{m}\sum_{j=1}^{m} x^{(j)}\right] = \sigma^2\rho + (1-\rho)\frac{\sigma^2}{m}.$$

Thought — If we trained DT on D, D, D .... D m times then "ρ" b/w $h^{(1)}, h^{(2)}, \ldots, h^{(m)}$ would be high

corr

But, for bootstrap samples, ρ is lower!

If we sample a lot of $z^{(i)}$s, the m↑, makes $(1-\rho)\frac{\sigma^2}{m}$ ↓

We are reducing the variance by training on a subset of the data, bias ↑.

(2) Out-of-bag error.

In sampling $z^{(i)}$, 37% of D doesn't get sampled!

free 'val' set

Almost

Random Forests = DT + Bagging

DT ← low bias, high variance

DT + Bagging = ALMOST RF

making DT ← Bagging – increases bias, decreases variance!
good choice to bag with!

In DT construction, one of the attributes could be really important, meaning all $h^{(1)}, \dots, h^{(m)}$ would likely split on that feature (eg. HP pokemon)

↓

At the first step, all DT are correlated!

→ At each split, I am only going to use a random subset of features

↓

drives ρ, further down, since we're cutting down on features, we ↑ bias!

The good and the bad (of BAGGING)

Additive modeling is still issue!