CS 360: Machine Learning

Prof. Sara Mathieson

Fall 2019

HAVERFORD COLLEGE

Admin

- **Midterm 1 Thursday** (TODAY in lab + take home)
  - In-lab: may use self-created "cheat-sheet" + calculator
  - Take-home due Tuesday by 6pm (in my office)

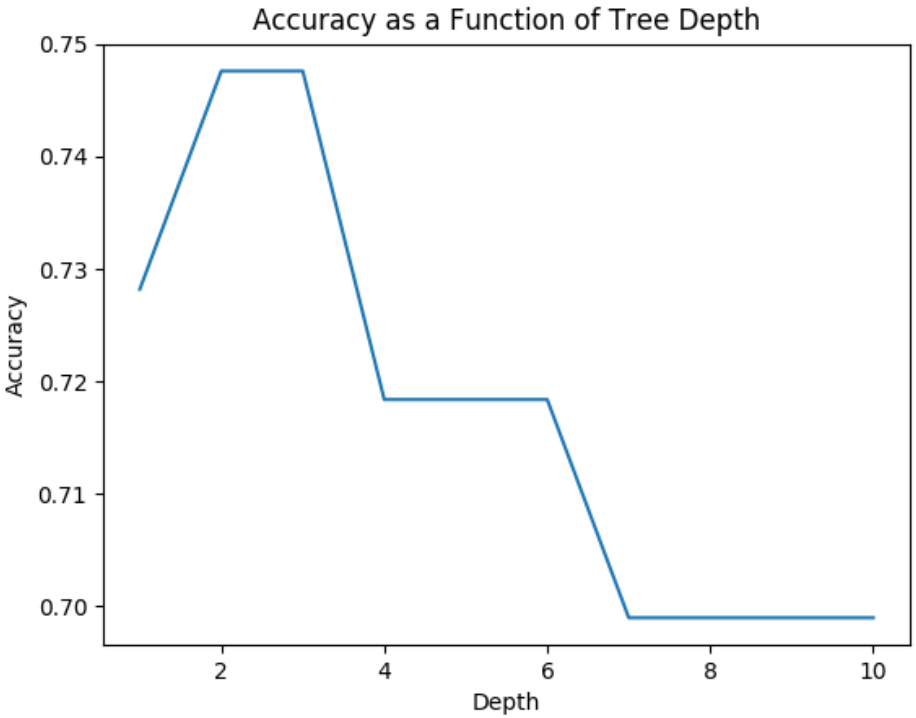
- No office hours this Friday or this coming Monday
  - Feel free to make an appointment to talk about material or anything else about the course

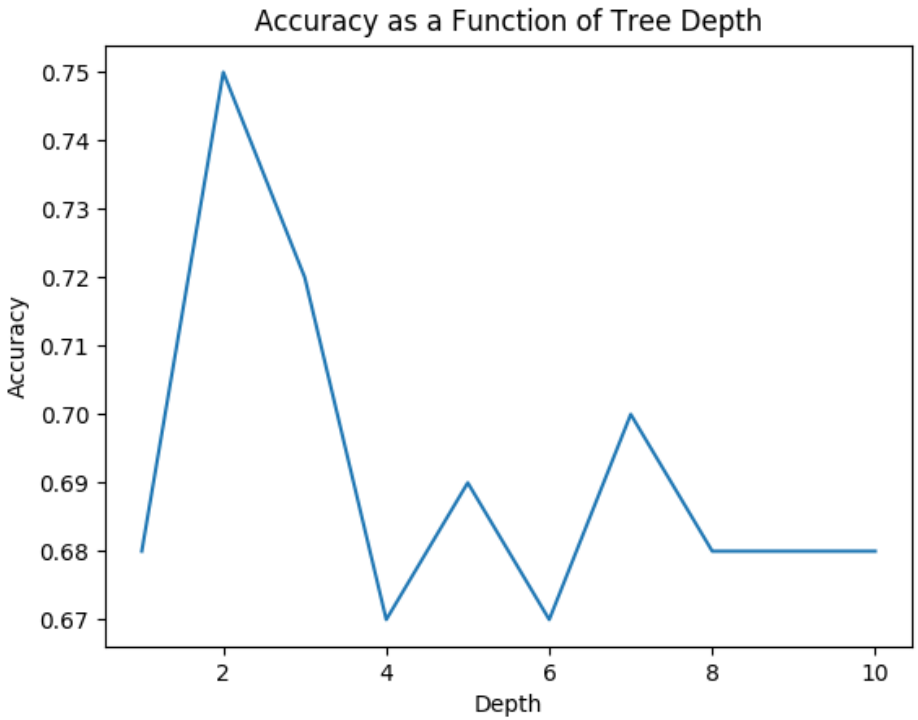
- **Lab 5 due October 22** (Tuesday after fall break)

- **Lab 2** has been graded

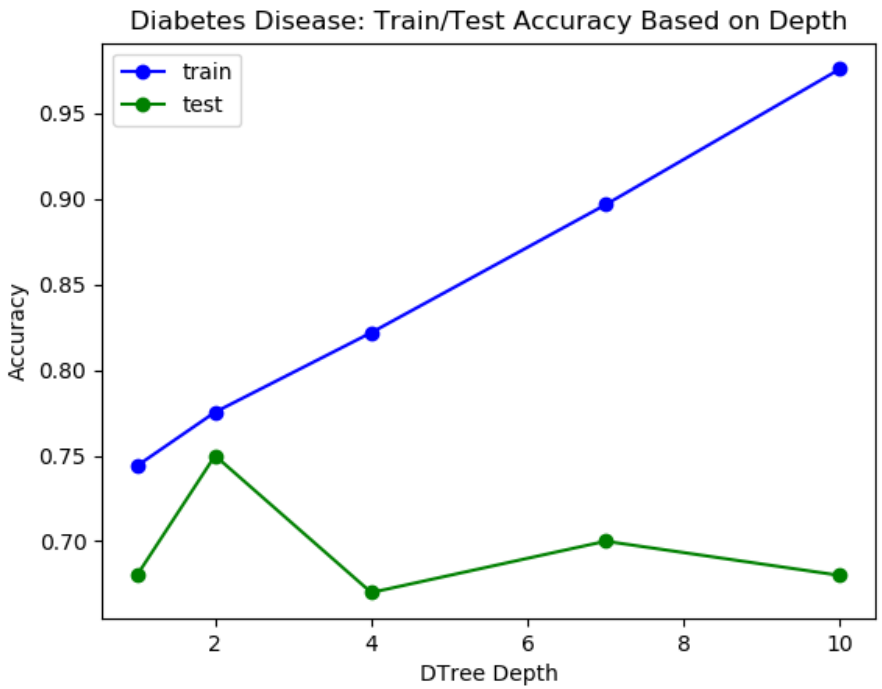
Lab 2 followups

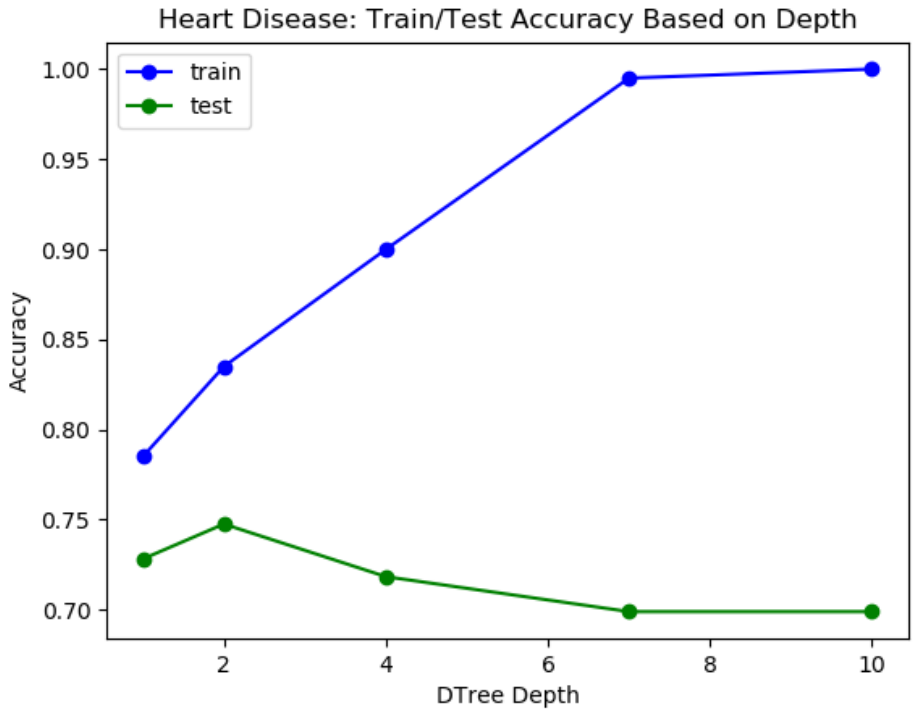
Lab 2 depth: Gareth





Lab 2 depth: Tessa





Lab 2 highlights

- Excellent TDD and style: Russell
- Excellent analysis: Rudy

Considerations when using a decision tree in medical applications

Advantages

- Very fast, gives a quick guide
- Could be more objective across different patient circumstances

Potential issues:

- Requires lots of historical data
- Decision trees work with how symptoms and other factors were historically measured
- If a new type of scan, xray, etc becomes available, we won't have historical data to incorporate into the algorithm
- Decision trees do not elegantly handle continuous features
- Patient may have a condition not seen in the training data
- 75% is not an optimal accuracy!

Outline for October 3

- Review
  - Loss functions & bias-variance tradeoff
  - Decision Trees
    - Revisit entropy intuition
    - High-level implementation, recursive aspect, splitting
    - Continuous -> Binary features
  - Linear Regression
    - SGD with varying alpha
    - Closed form vs. SGD
    - Polynomial regression (did last time!)
- Begin: Logistic Regression

Outline for October 3

- Review
  - Loss functions & bias-variance tradeoff
  - Decision Trees
    - Revisit entropy intuition
    - High-level implementation, recursive aspect, splitting
    - Continuous -> Binary features
  - Linear Regression
    - SGD with varying alpha
    - Closed form vs. SGD
    - Polynomial regression (did last time!)
- Begin: Logistic Regression

Loss functions

$\hat{y}$ = prediction, y = truth.

- "zero/one" loss: $\ell(y, \hat{y}) = \begin{cases} 0 & \text{if } y = \hat{y} \\ 1 & \text{o.w.} \end{cases}$

(binary / multi-class classification)

- "squared" loss: $\ell(y, \hat{y}) = (\hat{y} - y)^2$

(regression)

$$\boxed{J(\vec{w}) = \frac{1}{2} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}$$

cost function for regression.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$$

prediction function true

$$E[MSE] = \underbrace{E[(\hat{f} - f)^2]}_{\text{reducible}} + \underbrace{Var(\varepsilon)}_{\text{noise}}$$

irreducible

$$\underbrace{y = f(x) + \varepsilon}_{\text{assumption.}}$$

$$E[MSE] = \underbrace{bias(\hat{f})^2 + Var(\hat{f})}_{\text{bias-variance tradeoff.}} + Var(\varepsilon)$$

as model flexibility ↑

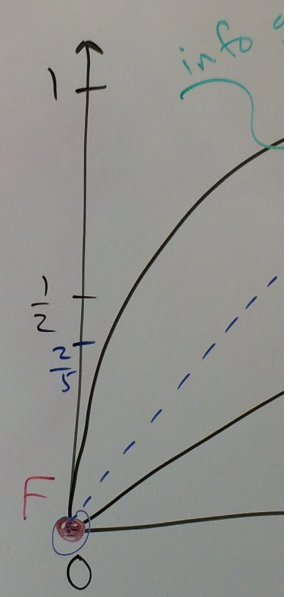
- bias ↓
- Var ↑

Dtree: depth.

~~lin~~ reg: deg of poly.

NB: } modified features.

KNN: }



Outline for October 3

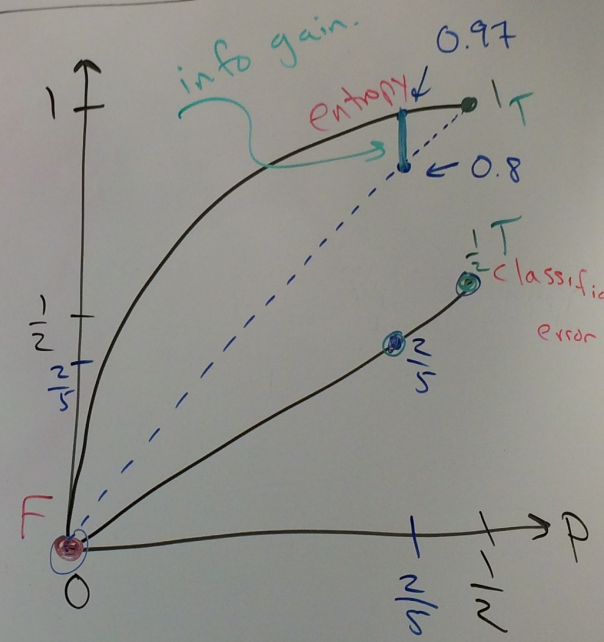
- Review
  - Loss functions & bias-variance tradeoff
  - Decision Trees
    - Revisit entropy intuition
    - High-level implementation, recursive aspect, splitting
    - Continuous -> Binary features
  - Linear Regression
    - SGD with varying alpha
    - Closed form vs. SGD
    - Polynomial regression (did last time!)
- Begin: Logistic Regression

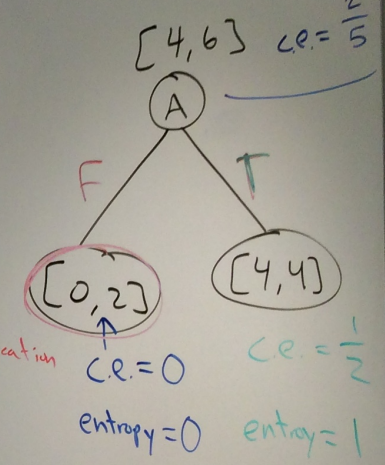
$\text{ar}(\hat{f}) + \text{Var}(\varepsilon)$

nce tradeoff.

Dtrees

one more try at entropy:





→ entropy: $-\left(\frac{2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}\right) \approx 0.97$

<u>weighted average</u>

c.e. $\Rightarrow 0 \cdot \frac{2}{10} + \frac{1}{2} \cdot \frac{8}{10} = \frac{2}{5}$

entropy $\Rightarrow 0 \cdot \frac{2}{10} + 1 \cdot \frac{8}{10} = \frac{4}{5}$

info gain $= H(Y) - H(Y|X) = 0.97 - 0.8$

$= 0.17$

implementation

dictionary to split

	A
x_1	F
x_2	T
⋮	F
	⋮
x_{10}	F

groups = { F : [x_1, x_4 ··· x_{10}],
T : [x_2, x_3, x_7 ···] }

recursive step

for v in {F, T}:

list of Examples

np = Partition(groups[v], feats − A)

recursive

child = DecisionTree(np, depth+1)

Self.children[v] = child.

age	label
21	Y
18	N
21	N
17	Y

17	18	21	21
Y	N	N	Y

ED

N

outlook

S R W

17	18	21
Y	N	None

age ≤ 17.5

age ≤ 19.5

cont → binary.

Outline for October 3

- Review
  - Loss functions & bias-variance tradeoff
  - Decision Trees
    - Revisit entropy intuition
    - High-level implementation, recursive aspect, splitting
    - Continuous -> Binary features
  - Linear Regression
    - SGD with varying alpha
    - Closed form vs. SGD
    - Polynomial regression (did last time!)
- Begin: Logistic Regression

SGD with varying α

* common choice : $\alpha = \frac{1}{t}$, where t is the iteration.

$t = 1$

while not converged:

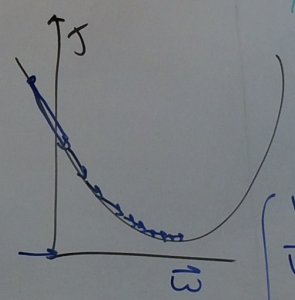
One SGD iter

$\alpha = \frac{1}{t}$

for $i = 1 \dots n$:

$\vec{w} \leftarrow \vec{w} - \alpha(\hat{y}_i - y)\vec{x}_i$

$t \mathrel{+}= 1$



$t = 5$ $\quad \vec{w} = \begin{bmatrix} 0.5 \\ -2 \\ 6.1 \\ 0.2 \end{bmatrix}$ $\quad p = 3$

$\vec{x} = [\dots x_j \dots]$

$\vec{x}_i = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 4 \end{bmatrix} \begin{matrix} x_{i1} \\ x_{i2} \\ x_{i3} \end{matrix}$

$\vec{x}_i$

$y_i = 7$

$\begin{bmatrix} | \\ \vec{w} \\ | \end{bmatrix} \leftarrow \begin{bmatrix} 0.5 \\ -2 \\ 6.1 \\ 0.2 \end{bmatrix} - \frac{1}{5}\left(\begin{bmatrix} 0.5 & -2 & 6.1 & 0.2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 4 \end{bmatrix} - 7 \right) \begin{bmatrix} 1 \\ 2 \\ 0 \\ 4 \end{bmatrix}$

$\boxed{p=1}$ $x_i = 4$, $d=3$

$$\vec{x}_i = \begin{bmatrix} 1 \\ 4 \\ 16 \\ 64 \end{bmatrix} = \phi(x_i)$$

$$\hat{h}(x) = 7 + 0.8x + 2x^2 + 3x^3$$

$7 \quad 7^2 \quad 7^3$

<u>test time</u>

$$x_{test} = 7$$

$$\vec{x}_{test} = \begin{bmatrix} 1 \\ 7 \\ 7^2 \\ 7^3 \end{bmatrix} \Rightarrow \vec{w} \cdot \vec{x}_{test}$$

$$\begin{bmatrix} | & - & \vec{x}_1^{\,T} & - \\ | & & & \\ | & & & \\ | & & & \end{bmatrix}$$

X

Pros and Cons

Gradient Descent

- requires multiple iterations
- need to choose α
- works well when p is large
- can support online learning

Normal Equations

- non-iterative
- no need for α
- slow if p is large
  - matrix inversion is $O(p^3)$

Slide: modified from Jessica Wu
Based on slide by Eric Eaton

Outline for October 3

- Review
  - Loss functions & bias-variance tradeoff
  - Decision Trees
    - Revisit entropy intuition
    - High-level implementation, recursive aspect, splitting
    - Continuous -> Binary features
  - Linear Regression
    - SGD with varying alpha
    - Closed form vs. SGD
    - Polynomial regression (did last time!)
- Begin: Logistic Regression

Why is linear regression a bad choice for classification?

Case Study: you need to identify the medical condition of a patient in the emergency room on the basis of their symptoms.

Possible conditions (*y*) are:

- Stroke
- Drug overdose
- Epileptic seizure

1) If you were forced to use linear regression for this problem, how could you encode *y* to make it real-valued?

2) What issues arise with making y real-valued?

3) What if you just had two outcomes (i.e. stroke and drug overdose) -- why is linear regression still not a good choice?

Why is linear regression a bad choice for classification?

Case Study: you need to identify the medical condition of a patient in the emergency room on the basis of their symptoms.

Possible conditions (*y*) are:

- Stroke
- Drug overdose
- Epileptic seizure

1) If you were forced to use linear regression for this problem, how could you encode *y* to make it real-valued?

You could choose stroke=0, drug overdose=1, epileptic seizure=2 (or some permutation)

2) What issues arise with making y real-valued?

3) What if you just had two outcomes (i.e. stroke and drug overdose) -- why is linear regression still not a good choice?

Why is linear regression a bad choice for classification?

Case Study: you need to identify the medical condition of a patient in the emergency room on the basis of their symptoms.

Possible conditions (*y*) are:

- Stroke
- Drug overdose
- Epileptic seizure

1) If you were forced to use linear regression for this problem, how could you encode *y* to make it real-valued?

You could choose stroke=0, drug overdose=1, epileptic seizure=2 (or some permutation)

2) What issues arise with making y real-valued?

Assumes some *ordering* of the outcomes that is probably not there!

3) What if you just had two outcomes (i.e. stroke and drug overdose) -- why is linear regression still not a good choice?

Why is linear regression a bad choice for classification?

Case Study: you need to identify the medical condition of a patient in the emergency room on the basis of their symptoms.

Possible conditions (*y*) are:

- Stroke
- Drug overdose
- Epileptic seizure

1) If you were forced to use linear regression for this problem, how could you encode *y* to make it real-valued?

You could choose stroke=0, drug overdose=1, epileptic seizure=2 (or some permutation)

2) What issues arise with making y real-valued?

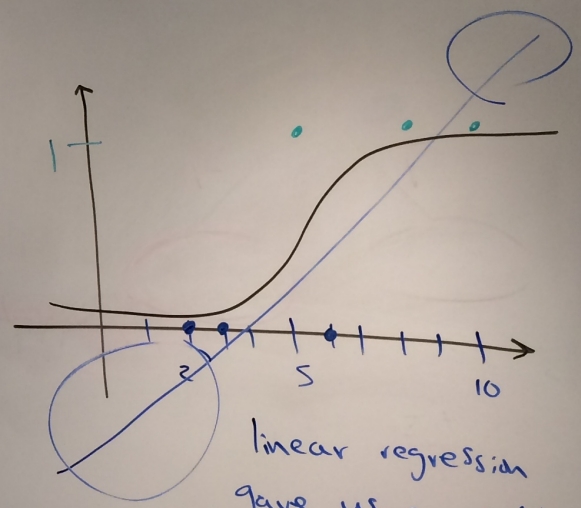
Assumes some *ordering* of the outcomes that is probably not there!

3) What if you just had two outcomes (i.e. stroke and drug overdose) -- why is linear regression still not a good choice?

The range of a linear function (i.e. *y* values) is $[-\infty, \infty]$, but we want $[0, 1]$

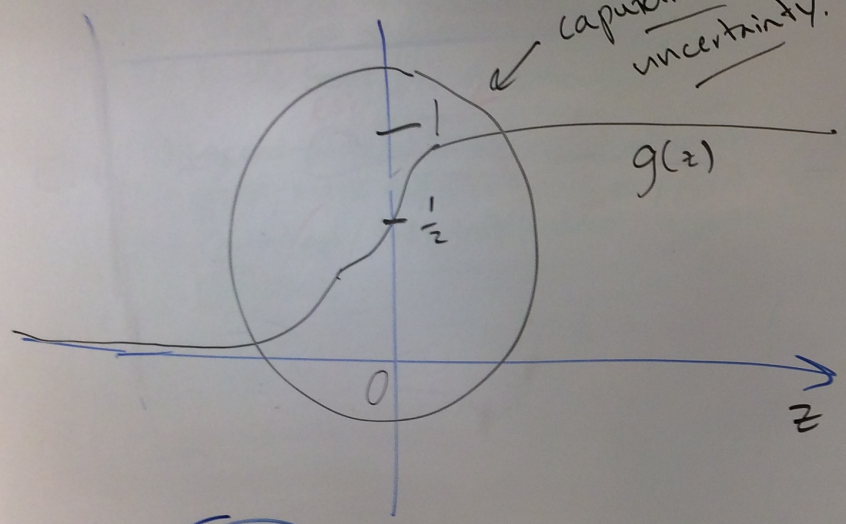
Back to binary classification...

x	y
8	1
6	0
2	0
5	1
10	1
3	0



linear regression gave us a weight on each feature.

Logistic Regression

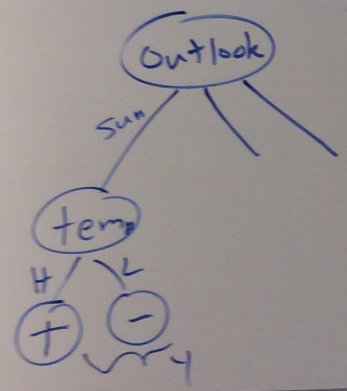


$$\boxed{g(z) = \frac{1}{1+e^{-z}}}$$

$z \to \infty$, $g(z) \to$

$z \to -\infty$, $g(z) \to$

Exercise!



Logistic (sigmoid) function