Overfitting

+ k-Nearest Neighbors
"Mister Rogers" redirects here. For the television series, see *Mister Rogers' Neighborhood*. For the asteroid, see 26858 Misterrogers. For other people, see Frederick Rogers and Rogers (surname).

Fred McFeely Rogers (March 20, 1928 – February 27, 2003) was an American television host, author, producer, and Presbyterian minister.[1] He was the creator, showrunner, and host of the preschool television series *Mister Rogers' Neighborhood*, which ran from 1968 to 2001.

Born in Latrobe, Pennsylvania, near Pittsburgh, Rogers earned a bachelor's degree in music from Rollins College in 1951. He began his television career at NBC in New York, returning to Pittsburgh in 1953 to work for children's programming at NET (later PBS) television station WQED. He graduated from Pittsburgh Theological Seminary with a bachelor's degree in divinity in 1962 and became a Presbyterian minister in 1963. He attended the University of Pittsburgh's Graduate School of Child Development, where he began his 30-year collaboration with child psychologist Margaret McFarland. He also helped develop the children's shows *The Children's Corner* (1955) for WQED in Pittsburgh and *Mister Rogers* (1963) in Canada for the Canadian Broadcasting Corporation. In 1968, he returned to Pittsburgh and adapted the format of his Canadian series to create *Mister Rogers' Neighborhood*. It ran for 33 years and was critically acclaimed for focusing on children's emotional and physical concerns, such as death, sibling rivalry, school enrollment, and divorce.

Rogers died of stomach cancer in 2003, aged 74. His work in children's television has been widely lauded, and he received more than 40 honorary degrees and several awards, including the Presidential Medal of Freedom in 2002 and a Lifetime Achievement Emmy in 1997. He was inducted into the Television Hall of Fame in 1999. Rogers influenced many writers and producers of children's television shows, and his broadcasts provided comfort during tragic events, even after his death.

**Early life**

Rogers was born on March 20, 1928, at 705 Main Street in Latrobe, Pennsylvania, about 40 miles (64 km) outside of Pittsburgh.[2] His father, James Hillis Rogers, was "a very successful businessman"[3] who was president of the McFeely Brick Company, one of Latrobe's most prominent businesses. His mother, Nancy (née McFeely), knitted sweaters for American soldiers from western Pennsylvania who were fighting in Europe and regularly volunteered at the Latrobe Hospital. Initially dreaming of becoming a doctor, she settled
Mr. Rogers was filmed at the WQED station on Fifth Ave (next to CMU)

Mr. Rogers lived in Squirrel Hill (next to CMU)
Mr. Roger’s Neighborhood

• Some of Mr. Roger’s neighbors...
  – Julia Childs (cookbook author)  
    https://www.misterrogers.org/videos/julia-child/
  – Yo-yo Ma (cellist) 2:38  
    https://www.misterrogers.org/videos/yo-yo-ma/
  – Silvia Earle (marine biologist) 3:00  
    https://www.misterrogers.org/videos/sylvia-earle/
  – Wynton Marsalis (trumpet player) 4:00  
    https://www.misterrogers.org/videos/wynton-marsalis/
  – Singing Won’t You Be My Neighbor  
    https://misterrogers.org/videos/wont-you-be-my-neighbor/
Course Staff

Team A (HW2, HW6)
- Sami Kale
- Abuzar Khan
- Bhargav Hadya
- Aditya Bansal
- Erin Gao

Team B (HW3, HW7)
- Chu Weng
- Monica Geng
- Asmita Hajra
- Dishani Lahiri
- Yash Gupta

Team C (HW4, HW8)
- Alex Xie
- Pranay Gundam
- Tanvi Karandikar
- Emaan Ahmed
- Pranit Chawla

Team D (HW5, HW9)
- Abhishek Vijayakumar
- Tara Lakdawala
- Hang Shu
- (Ads) Advaith Sridhar
- Poorvi Hebbar
<table>
<thead>
<tr>
<th>Q:</th>
<th>Why don’t my entropy calculations match those on the slides?</th>
</tr>
</thead>
<tbody>
<tr>
<td>A:</td>
<td>Remember that $H(Y)$ is conventionally reported in “bits” and computed using log base 2. e.g., $H(Y) = - P(Y=0) \log_2 P(Y=0) - P(Y=1) \log_2 P(Y=1)$</td>
</tr>
<tr>
<td>Q:</td>
<td>When and how do we decide to stop growing trees? What if the set of values an attribute could take was really large or even infinite?</td>
</tr>
<tr>
<td>A:</td>
<td>We’ll address this question for discrete attributes today. If an attribute is real-valued, there’s a clever trick that only considers $O(L)$ splits where $L$ = # of values the attribute takes in the training set. Can you guess what it does?</td>
</tr>
</tbody>
</table>
Q: What does decision tree training do if a branch receives no data?

A: Then we hit the base case and create a leaf node. So the real question is what does majority vote do when there is no data? Of course, there is no majority label, so (if forced to) we could just return one randomly.

Q: What do we do at test time when we observe a value for a feature that we didn’t see at training time.

A: This really just a variant of the first question. That said, a real DT implementation needs to elegantly handle this case. We could do so by either (a) assuming that all possible values will be seen at train time, so there should be a branch for all attributes even if the partition of the dataset doesn’t include them all or (b) recognize the unseen value at test time and return some appropriate label in that case.
Reminders

• Syllabus changes:
  1. no more exit polls for homework (folding those questions into in-class polls)
  2. no more solution sessions (folding correct answers into Gradescope rubric items, ask followup questions in office hours)

• Homework 2: Decision Trees
  – Out: Wed, Jan. 25
  – Due: Fri, Feb. 3 at 11:59pm
EMPIRICAL COMPARISON OF SPLITTING CRITERIA
Experiments: Splitting Criteria

Bluntine & Niblett (1992) compared 4 criteria (random, Gini, mutual information, Marshall) on 12 datasets

Medical Diagnosis Datasets: (4 of 12)
- **hypo**: data set of 3772 examples records expert opinion on possible hypo-thyroid conditions from 29 real and discrete attributes of the patient such as sex, age, taking of relevant drugs, and hormone readings taken from drug samples.
- **breast**: The classes are reoccurrence or non-reoccurrence of breast cancer sometime after an operation. There are nine attributes giving details about the original cancer nodes, position on the breast, and age, with multi-valued discrete and real values.
- **tumor**: examples of the location of a primary tumor
- **lymph**: from the lymphography domain in oncology. The classes are normal, metastases, malignant, and fibrosis, and there are nineteen attributes giving details about the lymphatics and lymph nodes

### Table 1. Properties of the data sets

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Classes</th>
<th>Attr.</th>
<th>Training Set</th>
<th>Test Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>hypo</td>
<td>4</td>
<td>29</td>
<td>1000</td>
<td>2772</td>
</tr>
<tr>
<td>breast</td>
<td>2</td>
<td>9</td>
<td>200</td>
<td>86</td>
</tr>
<tr>
<td>tumor</td>
<td>22</td>
<td>18</td>
<td>237</td>
<td>102</td>
</tr>
<tr>
<td>lymph</td>
<td>4</td>
<td>18</td>
<td>103</td>
<td>45</td>
</tr>
<tr>
<td>LED</td>
<td>10</td>
<td>7</td>
<td>200</td>
<td>1800</td>
</tr>
<tr>
<td>mush</td>
<td>2</td>
<td>22</td>
<td>200</td>
<td>7924</td>
</tr>
<tr>
<td>votes</td>
<td>2</td>
<td>17</td>
<td>200</td>
<td>235</td>
</tr>
<tr>
<td>votesI</td>
<td>2</td>
<td>16</td>
<td>200</td>
<td>235</td>
</tr>
<tr>
<td>iris</td>
<td>3</td>
<td>4</td>
<td>100</td>
<td>50</td>
</tr>
<tr>
<td>glass</td>
<td>7</td>
<td>9</td>
<td>100</td>
<td>114</td>
</tr>
<tr>
<td>xd6</td>
<td>2</td>
<td>10</td>
<td>200</td>
<td>400</td>
</tr>
<tr>
<td>pole</td>
<td>2</td>
<td>4</td>
<td>200</td>
<td>1647</td>
</tr>
</tbody>
</table>

Table from Bluntine & Niblett (1992)
Experiments: Splitting Criteria

Table 3. Error for different splitting rules (pruned trees).

<table>
<thead>
<tr>
<th>Data Set</th>
<th>GINI</th>
<th>Info. Gain</th>
<th>Marsh.</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>hypo</td>
<td>1.01 ± 0.29</td>
<td>0.93 ± 0.22</td>
<td>1.27 ± 0.47</td>
<td>7.44 ± 0.53</td>
</tr>
<tr>
<td>breast</td>
<td>28.66 ± 3.87</td>
<td>28.49 ± 4.28</td>
<td>27.15 ± 4.22</td>
<td>29.65 ± 4.97</td>
</tr>
<tr>
<td>tumor</td>
<td>60.88 ± 5.44</td>
<td>62.70 ± 3.89</td>
<td>61.62 ± 3.98</td>
<td>67.94 ± 5.68</td>
</tr>
<tr>
<td>lymph</td>
<td>24.44 ± 6.92</td>
<td>24.00 ± 6.87</td>
<td>24.33 ± 5.51</td>
<td>32.33 ± 11.25</td>
</tr>
<tr>
<td>LED</td>
<td>33.77 ± 3.06</td>
<td>32.89 ± 2.59</td>
<td>33.15 ± 4.02</td>
<td>38.18 ± 4.57</td>
</tr>
<tr>
<td>mush</td>
<td>1.44 ± 0.47</td>
<td>1.44 ± 0.47</td>
<td>7.31 ± 2.25</td>
<td>8.77 ± 4.65</td>
</tr>
<tr>
<td>votes</td>
<td>4.47 ± 0.95</td>
<td>4.57 ± 0.87</td>
<td>11.77 ± 3.95</td>
<td>12.40 ± 4.56</td>
</tr>
<tr>
<td>votes1</td>
<td>12.79 ± 1.48</td>
<td>13.04 ± 1.65</td>
<td>15.13 ± 2.89</td>
<td>15.62 ± 2.73</td>
</tr>
<tr>
<td>iris</td>
<td>5.00 ± 3.08</td>
<td>4.90 ± 3.08</td>
<td>5.50 ± 2.59</td>
<td>14.20 ± 6.77</td>
</tr>
<tr>
<td>glass</td>
<td>39.56 ± 6.20</td>
<td>50.57 ± 6.73</td>
<td>40.53 ± 6.41</td>
<td>53.20 ± 5.01</td>
</tr>
<tr>
<td>xd6</td>
<td>22.14 ± 3.23</td>
<td>22.17 ± 3.36</td>
<td>22.06 ± 3.37</td>
<td>31.86 ± 3.62</td>
</tr>
<tr>
<td>pole</td>
<td>15.43 ± 1.51</td>
<td>15.47 ± 0.88</td>
<td>15.91 ± 1.15</td>
<td>26.38 ± 6.92</td>
</tr>
</tbody>
</table>

Key Takeaway: GINI gain and Mutual Information are statistically indistinguishable!

Table from Bluntine & Niblett (1992)
Experiments: Splitting Criteria

**Table 4. Difference and significance of error for GINI splitting rule versus others.**

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Info. Gain</th>
<th>Marsh.</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>hypo</td>
<td>-0.06 (0.82)</td>
<td>0.26 (0.99)</td>
<td>6.43 (1.00)</td>
</tr>
<tr>
<td>breast</td>
<td>-0.17 (0.23)</td>
<td>-1.51 (0.94)</td>
<td>0.99 (0.72)</td>
</tr>
<tr>
<td>tumor</td>
<td>1.81 (0.84)</td>
<td>0.74 (0.39)</td>
<td>7.06 (0.99)</td>
</tr>
<tr>
<td>lymph</td>
<td>-0.44 (0.83)</td>
<td>0.11 (0.05)</td>
<td>7.89 (0.99)</td>
</tr>
<tr>
<td>LED</td>
<td>0.12 (0.17)</td>
<td>0.08</td>
<td></td>
</tr>
<tr>
<td>mush</td>
<td>0.00 (0.00)</td>
<td>5.86</td>
<td></td>
</tr>
<tr>
<td>votes</td>
<td>-8.11 (5.15)</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>votes</td>
<td>-1.34</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>iris</td>
<td>0.50</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>glass</td>
<td>0.96</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>xd6</td>
<td>0.07</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>pole</td>
<td>0.43</td>
<td>0.61</td>
<td></td>
</tr>
</tbody>
</table>

**Key Takeaway:** GINI gain and Mutual Information are statistically indistinguishable!

Results are of the form A.AA (B.BB) where:
1. **A.AA** is the **average difference in errors** between the two methods.
2. **B.BB** is the **significance** of the difference according to a two-tailed **paired t-test**

Table from Bluntine & Niblett (1992)
INDUCTIVE BIAS
(FOR DECISION TREES)
In-Class Exercise

Which of the following trees would be learned by the decision tree learning algorithm using “error rate” as the splitting criterion? (Assume ties are broken alphabetically.)
Background: Greedy Search

Greedy Search:
- At each node, selects the edge with lowest (immediate) weight
- Heuristic method of search (i.e. does not necessarily find the best path)
- Computation time: linear in max path length

Goal:
- Search space consists of nodes and weighted edges
- Goal is to find the lowest (total) weight path from root to a leaf
Background: Greedy Search

Goal:
- Search space consists of nodes and weighted edges
- Goal is to find the lowest (total) weight path from root to a leaf

Greedy Search:
- At each node, selects the edge with lowest (immediate) weight
- **Heuristic** method of search (i.e. does not necessarily find the best path)
- Computation time: linear in max path length
Greedy Search:
• At each node, selects the edge with lowest (immediate) weight
• Heuristic method of search (i.e. does not necessarily find the best path)
• Computation time: \textbf{linear} in max path length

Goal:
• Search space consists of nodes and weighted edges
• Goal is to find the lowest (total) weight path from root to a leaf

Start State

States

End State

Goal
Background: Global Search

Goal:
- Search space consists of nodes and weighted edges
- Goal is to find the lowest (total) weight path from root to a leaf

Global Search:
- Compute the weight of the path to every leaf
- **Exact** method of search (i.e. guaranteed to find the best path)
- Computation time: exponential in max path length
Decision Tree Learning as Search

1. **search space**: all possible decision trees
2. **node**: single decision tree
3. **edge**: connects one full tree to another, where child has one more split than parent
4. **edge weight**: (negative) splitting criterion
5. **DT learning**: greedy search, maximizing our splitting criterion at each step
Big Question: How is it that your ML algorithm can generalize to unseen examples?
Question: Which tree does ID3 find?

ID3 = Decision Tree Learning with Mutual Information as the splitting criterion
Question: Which tree does ID3 find?

Definition:
We say that the inductive bias of a machine learning algorithm is the principal by which it generalizes to unseen examples.

Inductive Bias of ID3:
Smallest tree that matches the data with high mutual information attributes near the top.

Occam’s Razor: (restated for ML)
Prefer the simplest hypothesis that explains the data.
Decision Tree Learning Example

In-Class Exercise

Suppose you had an algorithm that found the tree with lowest training error that was as small as possible (i.e. exhaustive global search), which tree would it return?

(Assume ties are broken by choosing the smallest.)
OVERFITTING
(FOR DECISION TREES)
Question:
Which of the following would generalize best to unseen examples?
A. Small tree with low training accuracy
B. Large tree with low training accuracy
C. Small tree with high training accuracy
D. Large tree with high training accuracy

Answer:
75%
Overfitting and Underfitting

**Underfitting**
- The model...
  - is too simple
  - is unable captures the trends in the data
  - exhibits too much bias
- *Example*: majority-vote classifier (i.e. depth-zero decision tree)
- *Example*: a toddler (that has not attended medical school) attempting to carry out medical diagnosis

**Overfitting**
- The model...
  - is too complex
  - is fitting the noise in the data or fitting “outliers”
  - does not have enough bias
- *Example*: our “memorizer” algorithm responding to an irrelevant attribute
- *Example*: medical student who simply memorizes patient case studies, but does not understand how to apply knowledge to new patients
Overfitting

• Given a hypothesis $h$, its…
  ... error rate over all training data: $\text{error}(h, D_{\text{train}})$
  ... error rate over all test data: $\text{error}(h, D_{\text{test}})$
  ... true error over all data: $\text{error}_{\text{true}}(h)$

• We say $h$ overfits the training data if…
  $\text{error}_{\text{true}}(h) > \text{error}(h, D_{\text{train}})$

• Amount of overfitting =
  $\text{error}_{\text{true}}(h) – \text{error}(h, D_{\text{train}})$

In practice, $\text{error}_{\text{true}}(h)$ is unknown

Slide adapted from Tom Mitchell
Overfitting in Decision Tree Learning

Figure from Tom Mitchell
How to Avoid Overfitting?

For Decision Trees...

1. Do not grow tree beyond some maximum depth
2. Do not split if splitting criterion (e.g. mutual information) is below some threshold
3. Stop growing when the split is not statistically significant
4. Grow the entire tree, then prune
Reduced-Error Pruning

Split data into training and validation set

- Create tree that classifies training set correctly
- Do until further pruning is harmful:
  1. Evaluate impact on validation set of pruning each possible node (plus those below it)
  2. Greedily remove the one that most improves validation set accuracy

- produces smallest version of most accurate subtree
- What if data is limited?
Effect of Reduced-Error Pruning

![Graph showing the effect of reduced-error pruning on accuracy vs. size of tree (number of nodes).](image)
Effect of Reduced-Error Pruning

IMPORTANT!
Shortly, we’ll learn that doing pruning on test data is the **wrong** thing to do.

Instead, use a third "validation" dataset.
Decision Trees (DTs) in the Wild

• DTs are one of the most popular classification methods for practical applications
  – Reason #1: The learned representation is easy to explain a non-ML person
  – Reason #2: They are efficient in both computation and memory

• DTs can be applied to a wide variety of problems including classification, regression, density estimation, etc.

• Applications of DTs include...
  – medicine, molecular biology, text classification, manufacturing, astronomy, agriculture, and many others

• Decision Forests learn many DTs from random subsets of features; the result is a very powerful example of an ensemble method (discussed later in the course)
DT Learning Objectives

You should be able to...

1. Implement Decision Tree training and prediction
2. Use effective splitting criteria for Decision Trees and be able to define entropy, conditional entropy, and mutual information / information gain
3. Explain the difference between memorization and generalization [CIML]
4. Describe the inductive bias of a decision tree
5. Formalize a learning problem by identifying the input space, output space, hypothesis space, and target function
6. Explain the difference between true error and training error
7. Judge whether a decision tree is "underfitting" or "overfitting"
8. Implement a pruning or early stopping method to combat overfitting in Decision Tree learning
REAL VALUED ATTRIBUTES
Fisher Iris Dataset

Fisher (1936) used 150 measurements of flowers from 3 different species: Iris setosa (0), Iris virginica (1), Iris versicolor (2) collected by Anderson (1936)

<table>
<thead>
<tr>
<th>Species</th>
<th>Sepal Length</th>
<th>Sepal Width</th>
<th>Petal Length</th>
<th>Petal Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4.3</td>
<td>3.0</td>
<td>1.1</td>
<td>0.1</td>
</tr>
<tr>
<td>0</td>
<td>4.9</td>
<td>3.6</td>
<td>1.4</td>
<td>0.1</td>
</tr>
<tr>
<td>0</td>
<td>5.3</td>
<td>3.7</td>
<td>1.5</td>
<td>0.2</td>
</tr>
<tr>
<td>1</td>
<td>4.9</td>
<td>2.4</td>
<td>3.3</td>
<td>1.0</td>
</tr>
<tr>
<td>1</td>
<td>5.7</td>
<td>2.8</td>
<td>4.1</td>
<td>1.3</td>
</tr>
<tr>
<td>1</td>
<td>6.3</td>
<td>3.3</td>
<td>4.7</td>
<td>1.6</td>
</tr>
<tr>
<td>1</td>
<td>6.7</td>
<td>3.0</td>
<td>5.0</td>
<td>1.7</td>
</tr>
</tbody>
</table>

Full dataset: https://en.wikipedia.org/wiki/Iris_flower_data_set
Fisher Iris Dataset

Fisher (1936) used 150 measurements of flowers from 3 different species: Iris setosa (0), Iris virginica (1), Iris versicolor (2) collected by Anderson (1936)

<table>
<thead>
<tr>
<th>Species</th>
<th>Sepal Length</th>
<th>Sepal Width</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4.3</td>
<td>3.0</td>
</tr>
<tr>
<td>0</td>
<td>4.9</td>
<td>3.6</td>
</tr>
<tr>
<td>0</td>
<td>5.3</td>
<td>3.7</td>
</tr>
<tr>
<td>1</td>
<td>4.9</td>
<td>2.4</td>
</tr>
<tr>
<td>1</td>
<td>5.7</td>
<td>2.8</td>
</tr>
<tr>
<td>1</td>
<td>6.3</td>
<td>3.3</td>
</tr>
<tr>
<td>1</td>
<td>6.7</td>
<td>3.0</td>
</tr>
</tbody>
</table>

Deleted two of the four features, so that input space is 2D

Full dataset: https://en.wikipedia.org/wiki/Iris_flower_data_set
Fisher Iris Dataset

![Fisher Iris Dataset Graph](image-url)
K-NEAREST NEIGHBORS
K-Nearest Neighbor: Pseudocode

def train(\mathcal{D}):
    store \mathcal{D}

def predict(x'):
    find the nearest neighbors to $x'$ in $\mathcal{D}$, $x^{(i)}$
    return $y^{(i)}$ and return the most common label among them