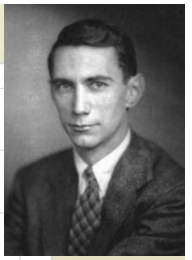
# Fundamentals of Machine Learning

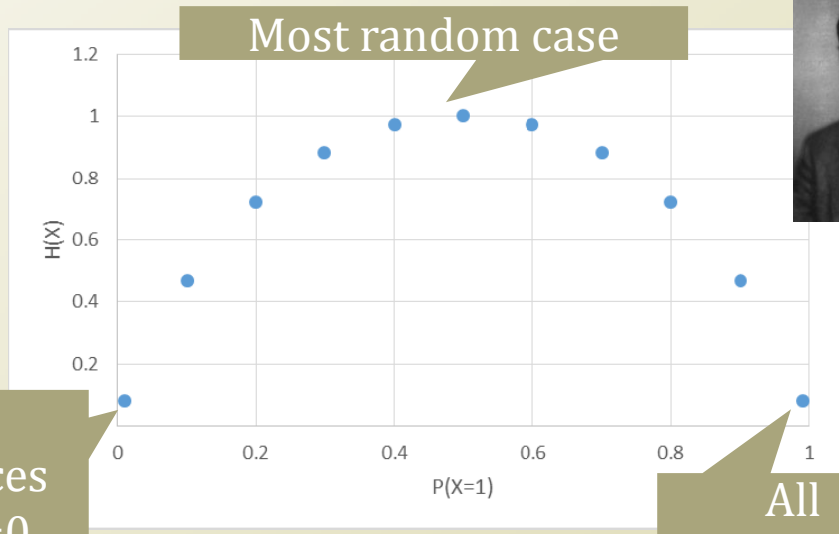
Il-Chul Moon
Dept. of Industrial and Systems Engineering
KAIST
icmoon@kaist.ac.kr

# Weekly Objectives

- Learn the most classical methods of machine learning
  - Rule based approach
  - Classical statistics approach
  - Information theory appraoch
- Rule based machine learning
  - How to find the specialized and the generalized rules
  - Why the rules are easily broken
- Decision Tree
  - How to create a decision tree given a training dataset
  - Why the tree becomes a weak learner with a new dataset
- Linear Regression
  - How to infer a parameter set from a training dataset
  - Why the feature engineering has its limit

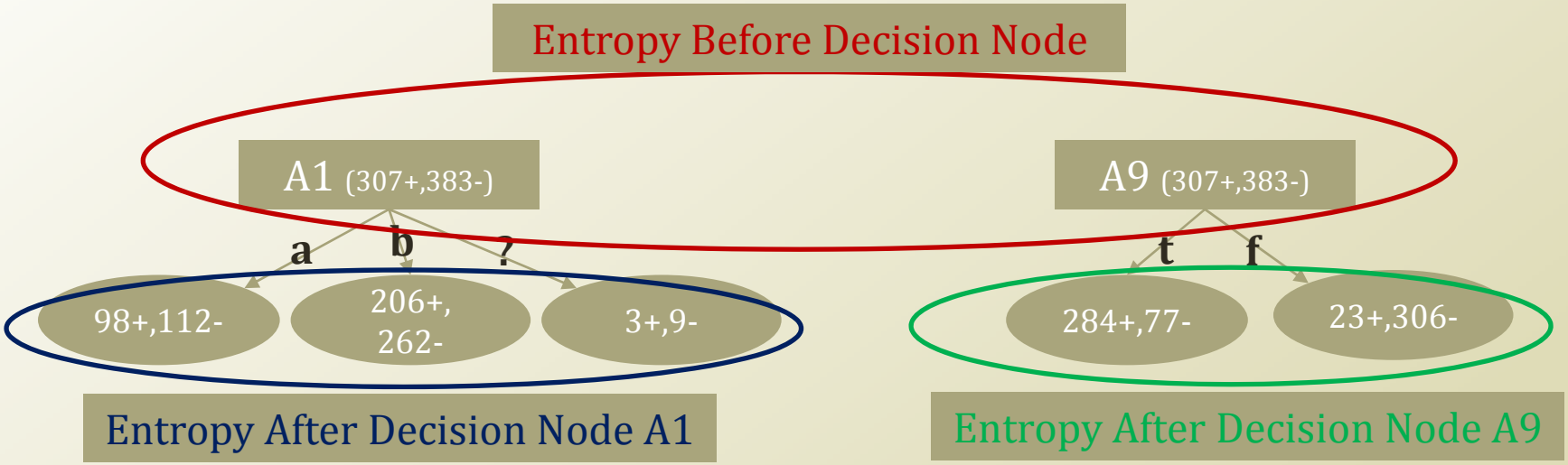
# Entropy






- Better attribute to check?
  - Reducing the most uncertainty
  - Then, how to measure the uncertainty of a feature variable
- Entropy of a random variable
  - Features are random variables
  - Higher entropy means more uncertainty
  - $H(X) = -\sum_X P(X = x) log_b P(X = x)$
- Conditional Entropy
  - We are interested in the entropy of the class given a feature variable
  - Need to introduce a given condition in the entropy
  - $H(Y|X) = \sum_X P(X = x) H(Y|X = x)$
    $= \sum_X P(X = x)\{-\sum_Y P(Y = y|X = x) log_b P(Y = y|X = x)\}$

# Information Gain

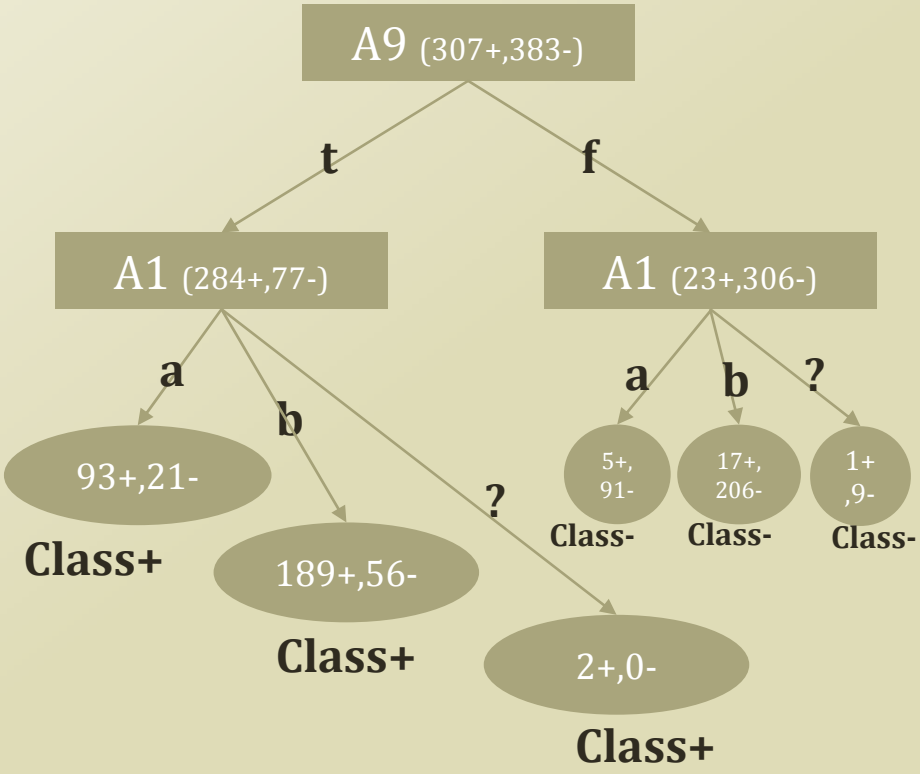



- Let's calculate the entropy values
  - $H(Y) = -\sum_{Y\in\{+,-\}} P(Y=y)\log_2 P(Y=y)$
  - $H(Y|A1) = \sum_{X\in\{a,b,?\}} \sum_{Y\in\{+,-\}} P(A1=x, Y=y)\log_2 \frac{P(A1=x)}{P(A1=x,Y=y)}$
  - $H(Y|A9) = \sum_{X\in\{t,f\}} \sum_{Y\in\{+,-\}} P(A9=x, Y=y)\log_2 \frac{P(A9=x)}{P(A9=x,Y=y)}$
- What's the difference before and after?
  - $IG(Y,A_i) = H(Y) - H(Y|A_i)$
- Who is the winner?

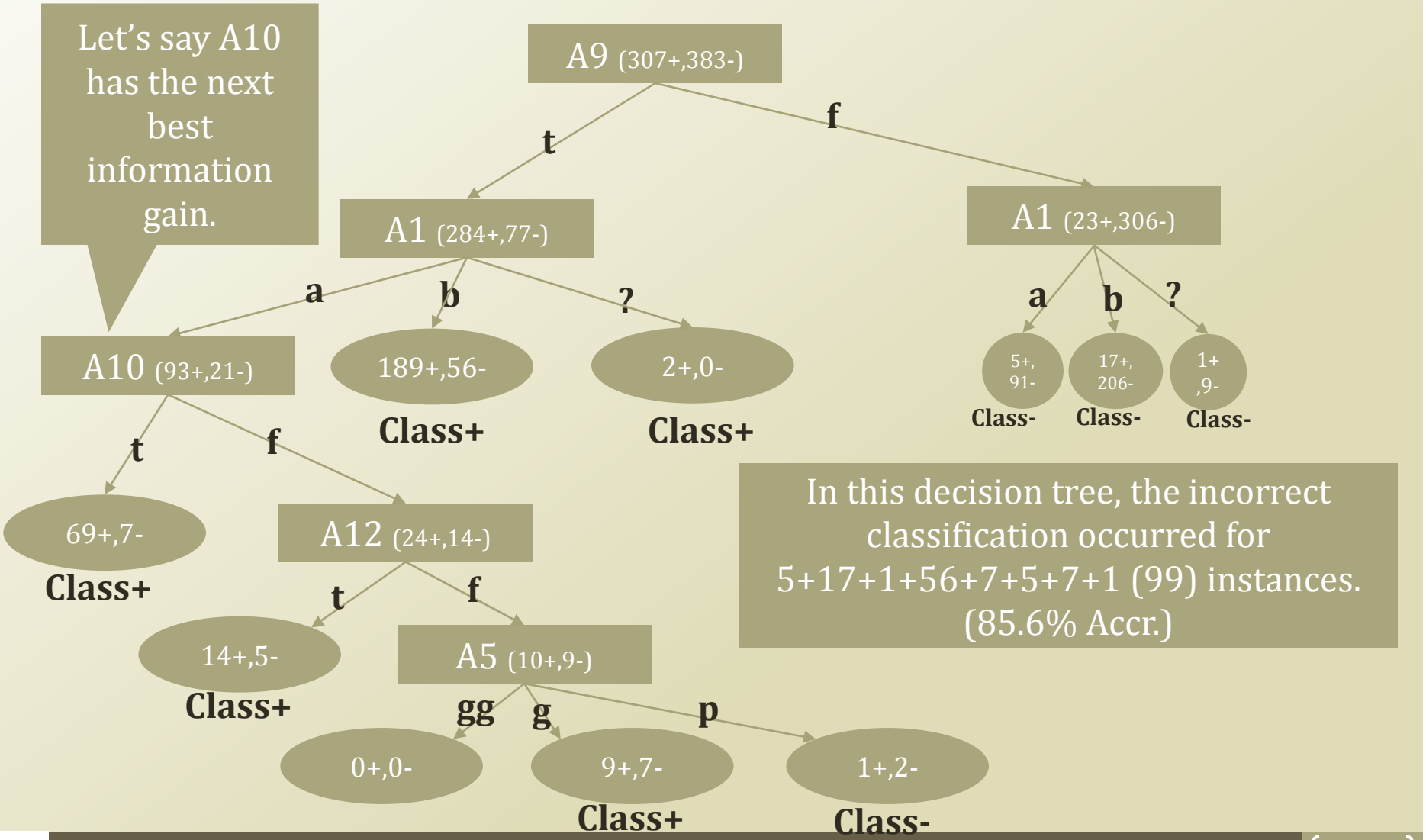
# Top-Down Induction Algorithm

- Many, many variations in learning a decision tree
  - ID3, C4.5 CART….
- One example: ID3 algorithm
- ID3 algorithm
  - Create an initial open node
  - Put instances in the initial node
  - Repeat until no open node
    - Select an open node to split
    - Select a best variable to split
    - For values of the selected variable
      - Sort instances with the value of the selected variable
      - Put the sorted items under the branch of the value of the variable
      - If the sorted items are all in one class
        - Close the leaf node of the branch

Only using A1 and A9, we have 21+56+0+5+17+1 (100) instances classified inaccurately. (85.5% Accr.)



# If you want more….

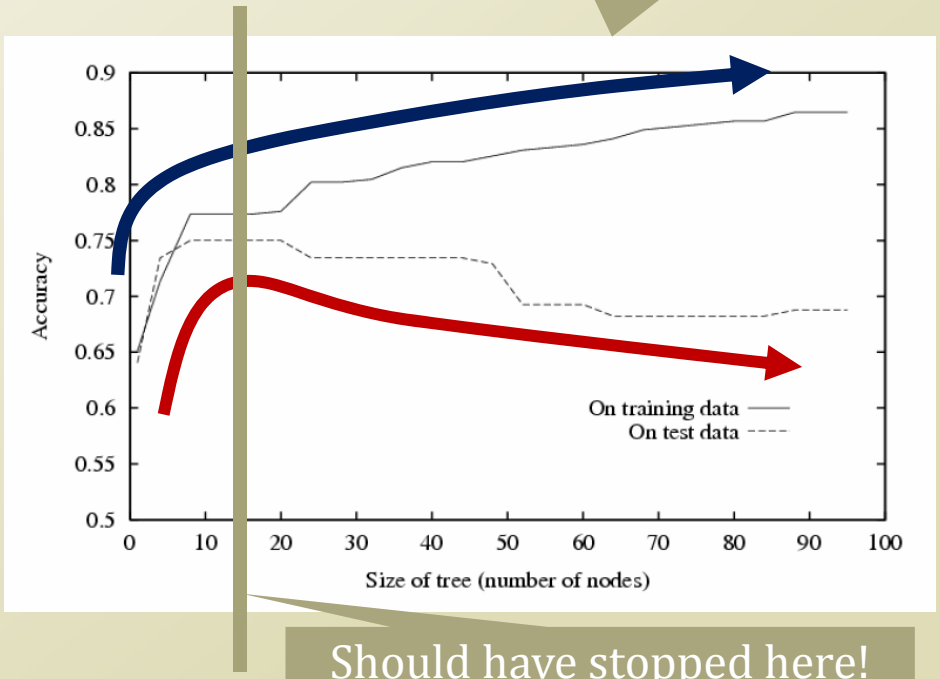



In this decision tree, the incorrect classification occurred for 5+17+1+56+7+5+7+1 (99) instances. (85.6% Accr.)

# Problem of Decision Tree

- We did better in the given dataset!
  - Only in the given experience, a.k.a. Training dataset
- What if we deploy the created decision tree in the field?
  - World has so much noise and inconsistencies.
  - The training dataset will not be a perfect sample of the real world
    - Noise
    - Inconsistencies

Typical result of decision tree




Should have stopped here!

**Knowing when to stop is a pretty difficult task. How to do it?**
- Pruning by divided dataset?
- Path length penalty?

# Why we are not interested in these?

- Rule based machine learning algorithms
  - Easy to implement
  - Easily interpretable
    - Particularly, decision tree
- Their weaknesses
  - Fragile
    - Assume the perfect world in the dataset
    - Any new observations, contradicting to the training, will cause problems
  - Convergence
    - Convergence only guaranteed in the perfect dataset
    - Once there is a noise, there is a possibility that the true hypothesis can be ruled out.
    - Also, very hard to tell when to stop in some cases
- Still used in many places
  - Easy → Wide audience and users → Many applications → Better result???
- Need a white knight as a savior
  - Should be able to handle noisy datasets
  - Robust to errors

Believe the small dataset?
(5/6→ Head with 83.3% prob?)