I256
Applied Natural Language Processing

Fall 2009

Lecture 11

Classification

Barbara Rosario

Classification

- Define classes/categories
- Label text
- Extract features
- Choose a classifier
  - Naive Bayes Classifier
  - NN (i.e. perceptron)
  - Maximum Entropy
  - ….
- Train it
- Use it to classify new examples

Today

- Algorithms for Classification

- Binary classification
  - Linear and not linear
- Multi-Class classification
  - Linear and not linear

- Examples of classification algorithms for each case

Classification

- In classification problems, each entity in some domain can be placed in one of a discrete set of categories: yes/no, friend/foe, good/bad/indifferent, blue/red/green, etc.
- Given a training set of labeled entities, develop a "rule" for assigning labels to entities in a test set
- Many variations on this theme:
  - binary classification
  - multi-category classification
  - non-exclusive categories
  - ranking
- Many criteria to assess rules and their predictions
  - overall errors
  - costs associated with different kinds of errors

Algorithms for Classification

- It's possible to treat these learning methods as black boxes.
- But there's a lot to be learned from taking a closer look
- An understanding of these methods can help guide our choice for the appropriate learning method (binary or multi-class for example)

Representation of Objects

- Each object to be classified is represented as a pair (x, y):
  - where x is a description of the object
  - where y is a label
- Success or failure of a machine learning classifier often depends on choosing good descriptions of objects
  - the choice of description can also be viewed as a learning problem (feature selections)
  - but good human intuitions are often needed here

Example: Digit Recognition

- Input: images / pixel grids
- Output: a digit 0-9
- Setup:
  - Get a large collection of example images, each labeled with a digit
  - Note: someone has to hand label all this data
  - Want to learn to predict labels of new, future digit images
- Features: The attributes used to make the digit decision
  - Pixels: (6,8)=ON
  - Shape Patterns: NumComponents, AspectRatio, NumLoops
  - …
- Current state-of-the-art: Human-level performance


0


1


2


1


??

Text Classification tasks

Assign the correct **class label** for a given input/object

In basic classification tasks, each input is considered in isolation from all other inputs, and the set of labels is defined in advance.

Examples:

Problem	Object	Label's categories
Tagging	Word	POS
Sense Disambiguation	Word	The word's senses
Information retrieval	Document	Relevant/not relevant
Sentiment classification	Document	Positive/negative
Text categorization	Document	Topics/classes
Author identification	Document	Authors
Language identification	Document	Language

Adapted from: Foundations of Statistical NLP (Manning et al)

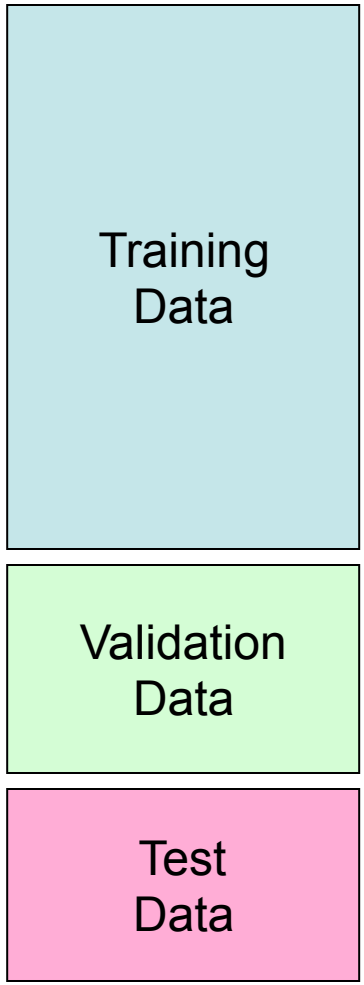
Other Examples of Real-World Classification Tasks

- Fraud detection (input: account activity, classes: fraud / no fraud)
- Web page spam detection (input: HTML/rendered page, classes: spam / ham)
- Speech recognition and speaker recognition (input: waveform, classes: phonemes or words)
- Medical diagnosis (input: symptoms, classes: diseases)
- Automatic essay grader (input: document, classes: grades)
- Customer service email routing and foldering
- Link prediction in social networks
- … many many more

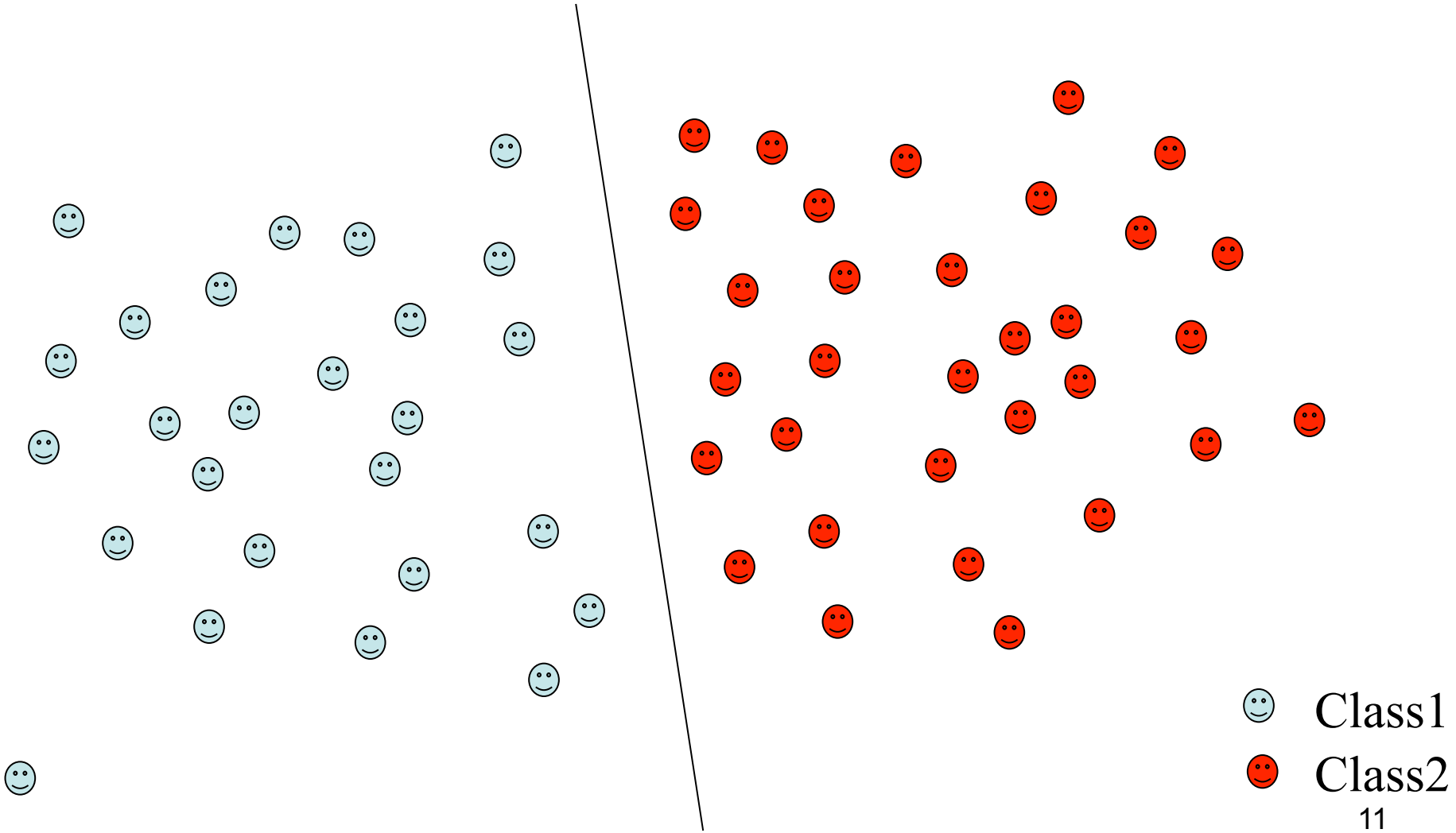
- Classification is an important commercial technology

Training and Validation

- Data: labeled instances, e.g. emails marked spam/ham
  - Training set
  - Validation set
  - Test set
- Training
  - Estimate parameters on training set
  - Tune features on validation set
  - Report results on test set
  - Anything short of this yields over-optimistic claims
- Evaluation
  - Many different metrics
  - Ideally, the criteria used to train the classifier should be closely related to those used to evaluate the classifier
- Statistical issues
  - Want a classifier which does well on *test* data
  - Overfitting: fitting the training data very closely, but not generalizing well
  - Error bars: want realistic (conservative) estimates of accuracy



Intuitive Picture of the Problem



Some Issues

- There may be a simple separator (e.g., a straight line in 2D or a hyperplane in general) or there may not
- There may be “noise” of various kinds
- There may be “overlap”
- Some classifiers explicitly represent separators (e.g., straight lines), while for other classifiers the separation is done implicitly
- Some classifiers just make a decision as to which class an object is in; others estimate class probabilities

Methods

- Binary vs. multi class classification
- Linear vs. non linear

Methods

- **Linear Models**:
  - Perceptron & Winnow (neural networks)
  - Large margin classifier
    - Support Vector Machine (SVM)
- **Probabilistic models**:
  - Naïve Bayes
  - Maximum Entropy Models
- **Decision Models**:
  - Decision Trees
- **Instance-based methods**:
  - Nearest neighbor

Binary Classification: examples

- Spam filtering (spam, not spam)
- Customer service message classification (urgent vs. not urgent)
- Information retrieval (relevant, not relevant)
- Sentiment classification (positive, negative)
- Sometime it can be convenient to treat a multi-way problem like a binary one: one class versus all the others, for all classes

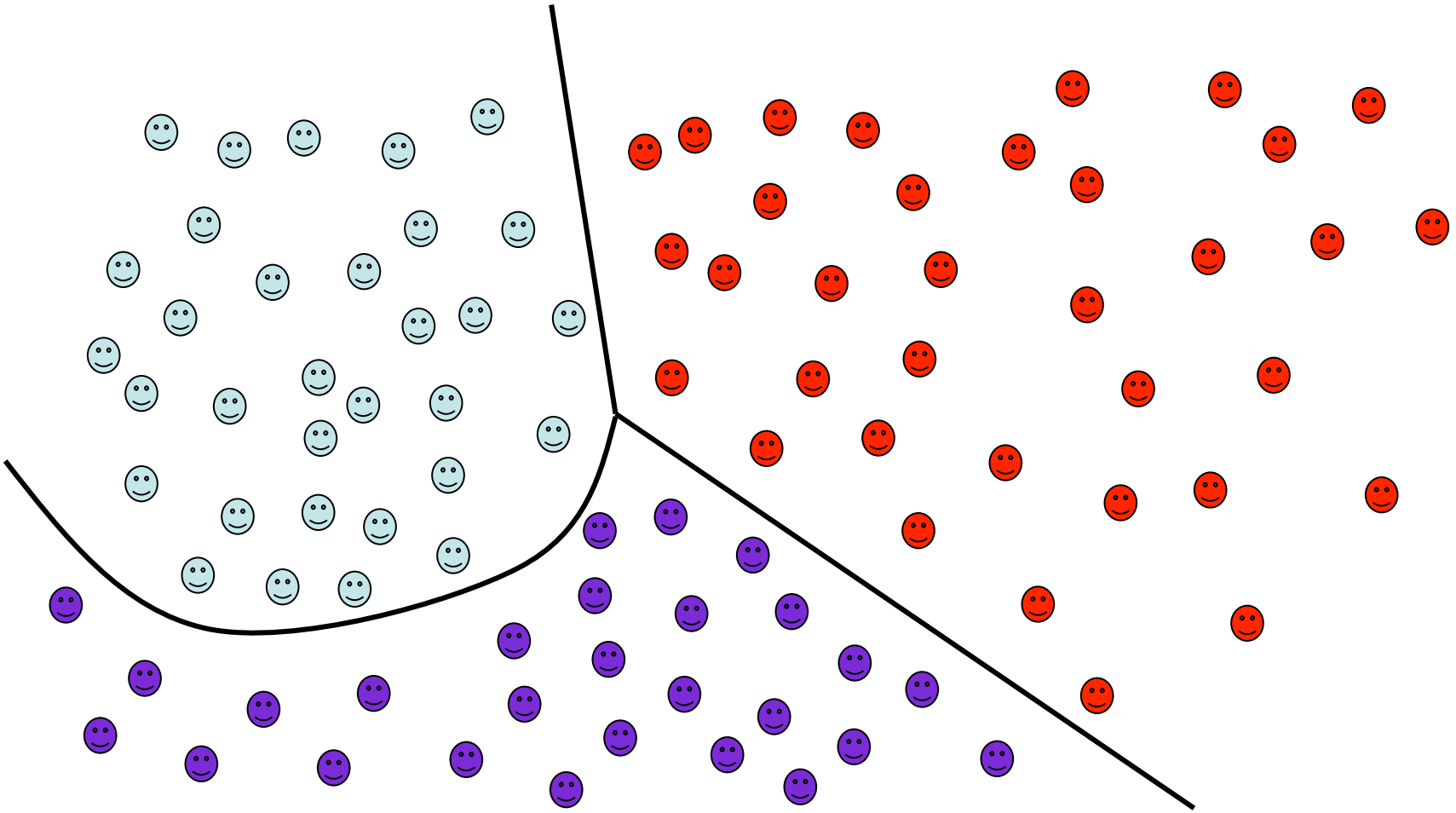
Binary Classification

- Given: some data items that belong to a positive (+1 ) or a negative (-1 ) class
- Task: Train the classifier and predict the class for a new data item
- Geometrically: find a separator

Multi-class classification

- Given: some data items that belong to one of M possible classes
- Task: Train the classifier and predict the class for a new data item
- Geometrically: harder problem, no more simple geometry

Multi-class classification



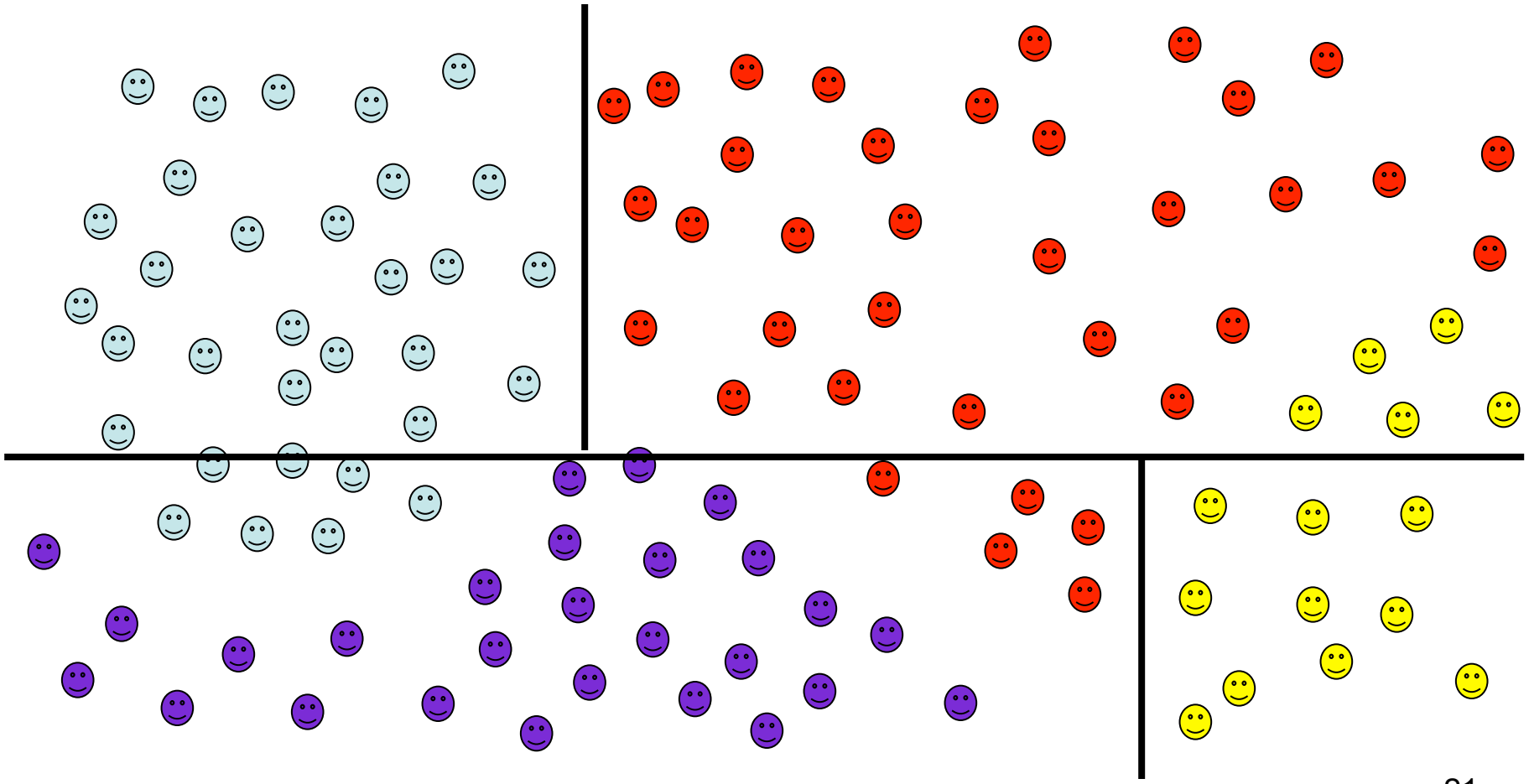
Multi-class classification: Examples

- Author identification
- Language identification
- Text categorization (topics)

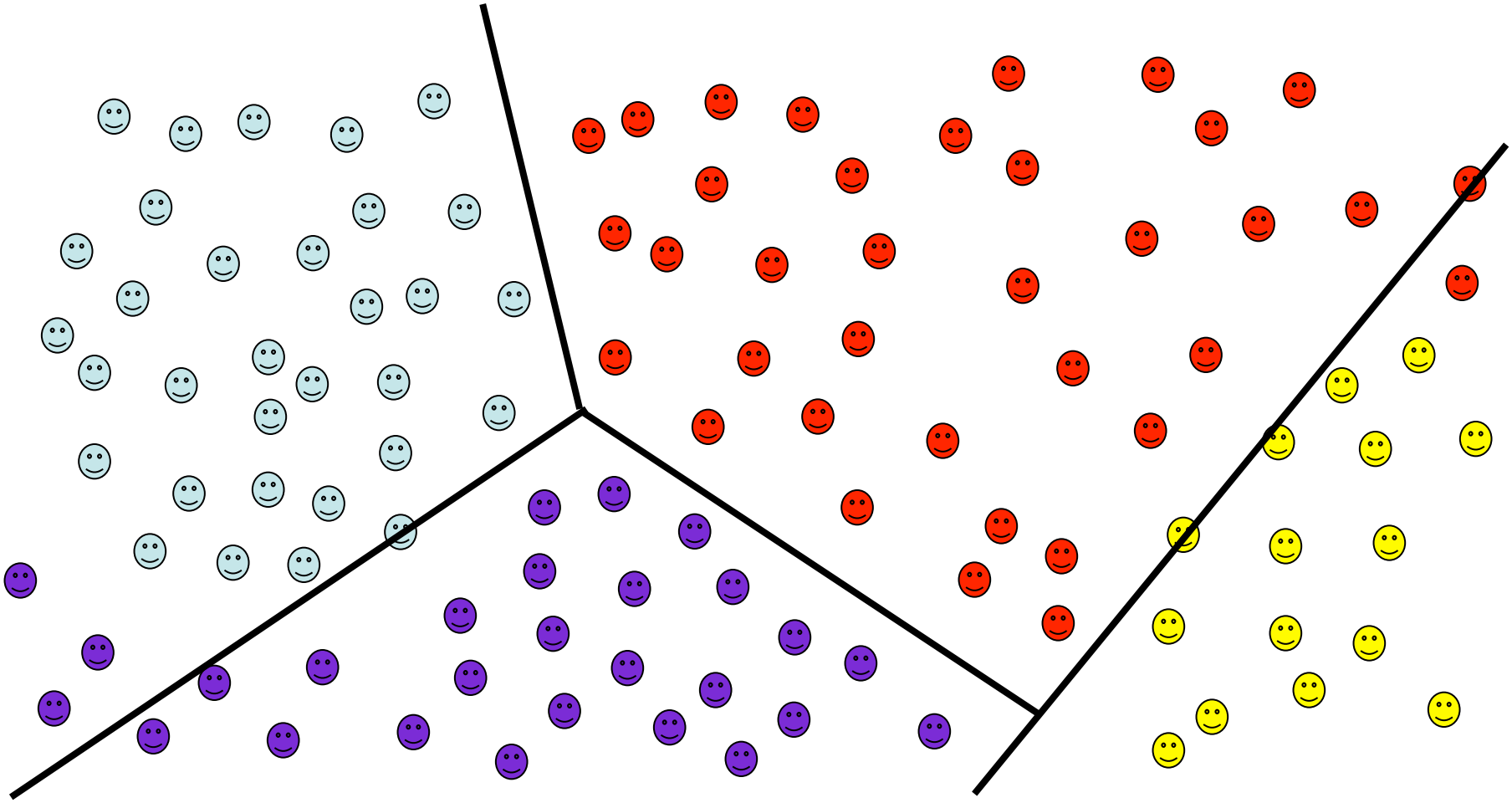
(Some) Algorithms for multi-class classification

- Linear
  - Parallel class separators: Decision Trees
  - Non parallel class separators: Naïve Bayes and Maximum Entropy
- Non Linear
  - K-nearest neighbors

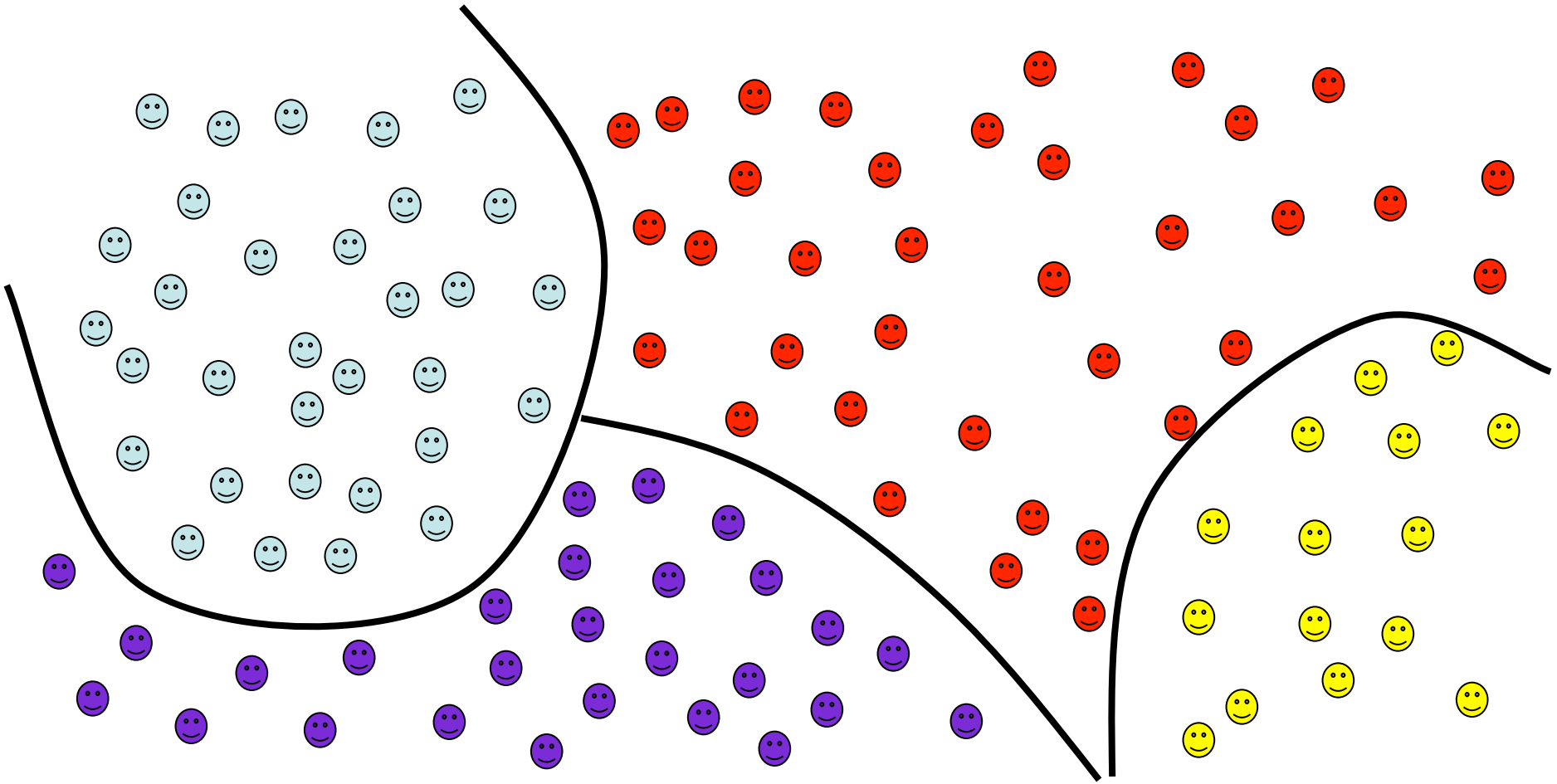
Linear, parallel class separators (ex: Decision Trees)



Linear, NON parallel class separators (ex: Naïve Bayes)



Non Linear (ex: *k* Nearest Neighbor)



Decision Trees

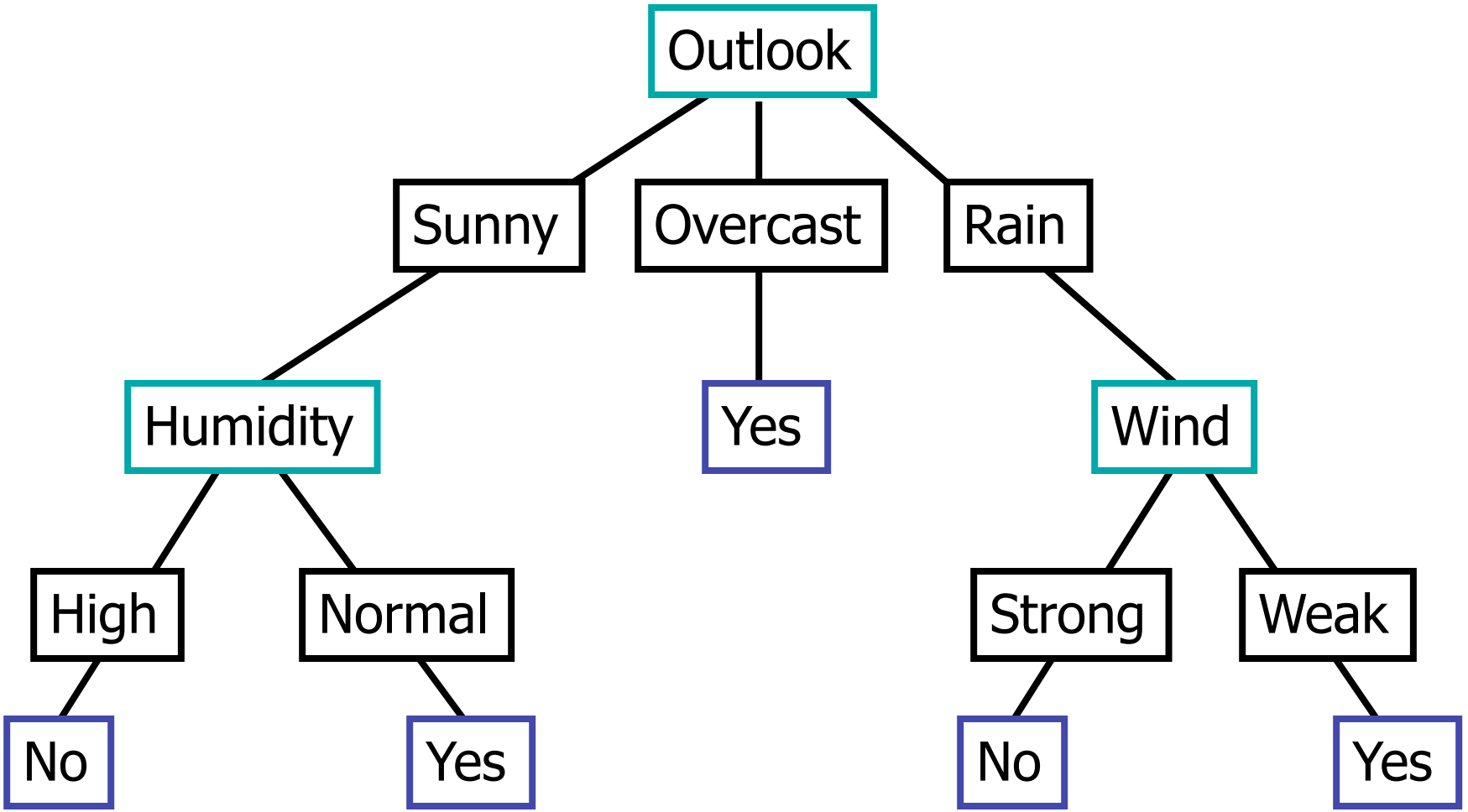
- *Decision tree* is a classifier in the form of a tree structure, where each node is either:
  - *Decision node* - specifies some test to be carried out on a single attribute-value, with one branch and sub-tree for each possible outcome of the test
  - *Leaf node* - indicates the value of the target attribute (class) of examples
- A decision tree can be used to classify an example by starting at the root of the tree and moving through it until a leaf node, which provides the classification of the instance.

Training Examples

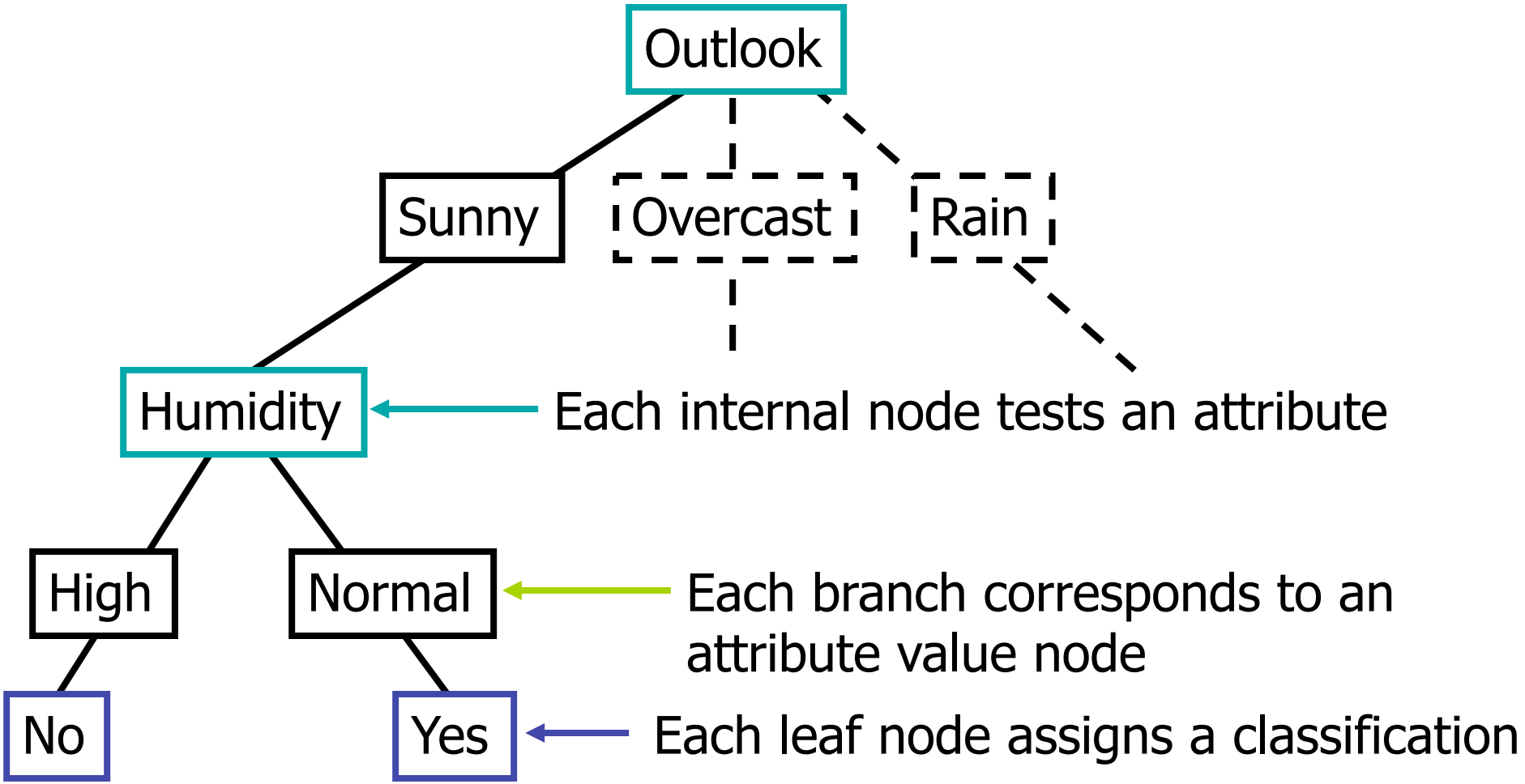
Goal: learn when we can play Tennis and when we cannot

Day	Outlook	Temp.	Humidity	Wind	**Play Tennis**
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Weak	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cold	Normal	Weak	Yes
D10	Rain	Mild	Normal	Strong	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

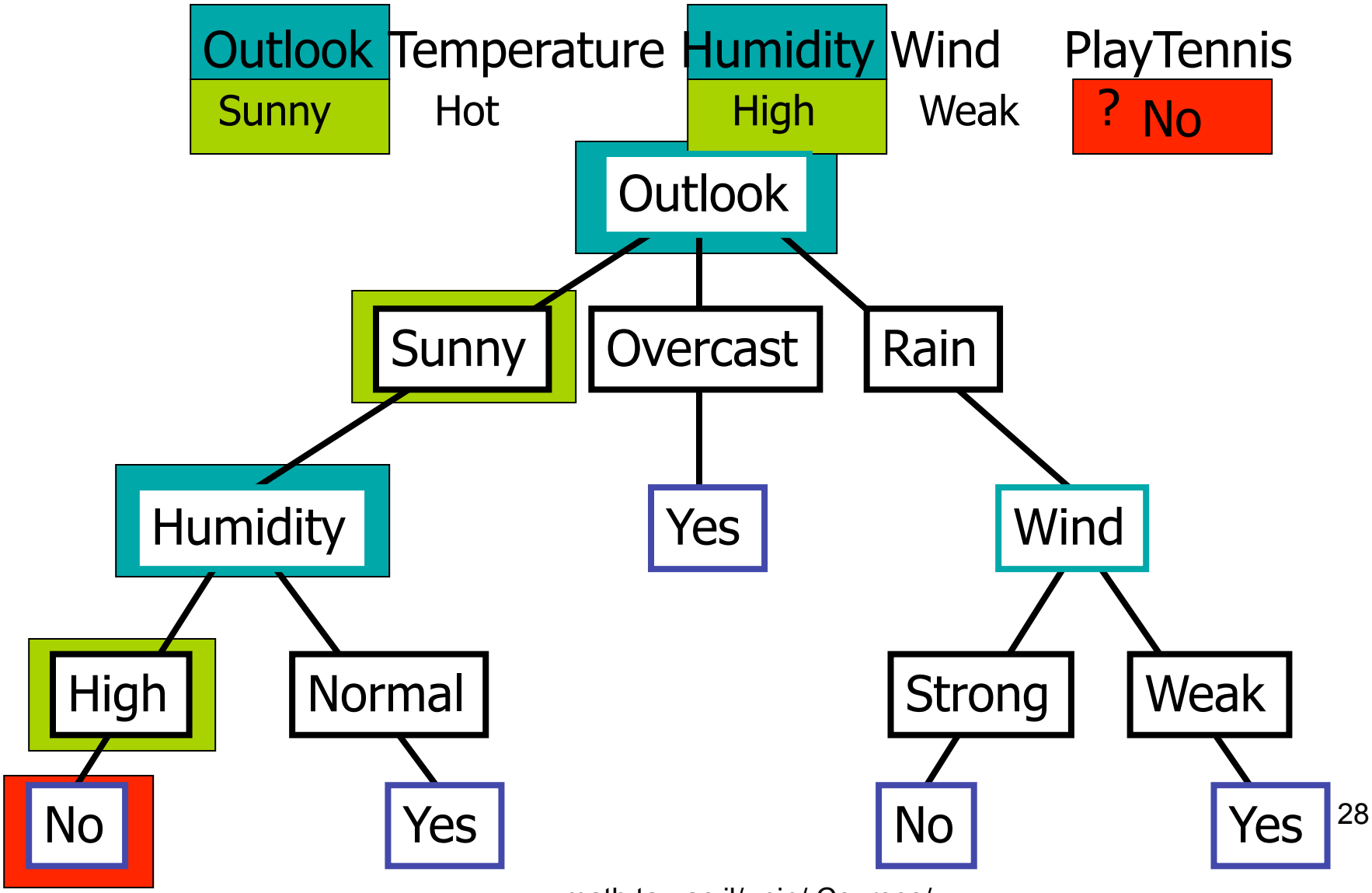
Decision Tree for PlayTennis



Decision Tree for PlayTennis



Decision Tree for PlayTennis



www.math.tau.ac.il/~nin/ Courses/
ML04/DecisionTreesCLS.pp

Decision Tree for Reuter classification

```
<REUTERS NEWID="11">
<DATE>26-FEB-1987 15:18:59.34</DATE>
<TOPICS><D>earn</D></TOPICS>
<TEXT>
<TITLE>COBANCO INC &lt;CBCO> YEAR NET</TITLE>
<DATELINE>    SANTA CRUZ, Calif., Feb 26 - </DATELINE>
<BODY>Shr 34 cts vs 1.19 dlrs
    Net 807,000 vs 2,858,000
    Assets 510.2 mln vs 479.7 mln
    Deposits 472.3 mln vs 440.3 mln
    Loans 299.2 mln vs 327.2 mln
    Note: 4th qtr not available. Year includes 1985
extraordinary gain from tax carry forward of 132,000 dlrs,
or five cts per shr.
 Reuter
</BODY></TEXT>
</REUTERS>
```

Figure 16.3 An example of a Reuters news story in the topic category "earnings." Parts of the original have been omitted for brevity.

Decision Tree for Reuter classification

```
<REUTERS NEWID="11">
<DATE>26-FEB-1987 15:18:59.34</DATE>
<TOPICS><D>earn</D></TOPICS>
<TEXT>
<TITLE>COBANCO INC &lt;CBCO> YEAR NET</TITLE>
<DATELINE>    SANTA CRUZ, Calif., Feb 26 - </DATELINE>
<BODY>Shr 34 cts vs 1.19 dlrs
    Net 807,000 vs 2,858,000
    Assets 510.2 mln vs 479.7 mln
    Deposits 472.3 mln vs 440.3 mln
    Loans 299.2 mln vs 327.2 mln
    Note: 4th qtr not available. Year includes 1985
extraordinary gain from tax carry forward of 132,000 dlrs,
or five cts per shr.
 Reuter
</BODY></TEXT>
</REUTERS>
```

Figure 16.3 An example of a Reuters news story in the topic category "earnings." Parts of the original have been omitted for brevity.

Decision Tree for Reuter classification

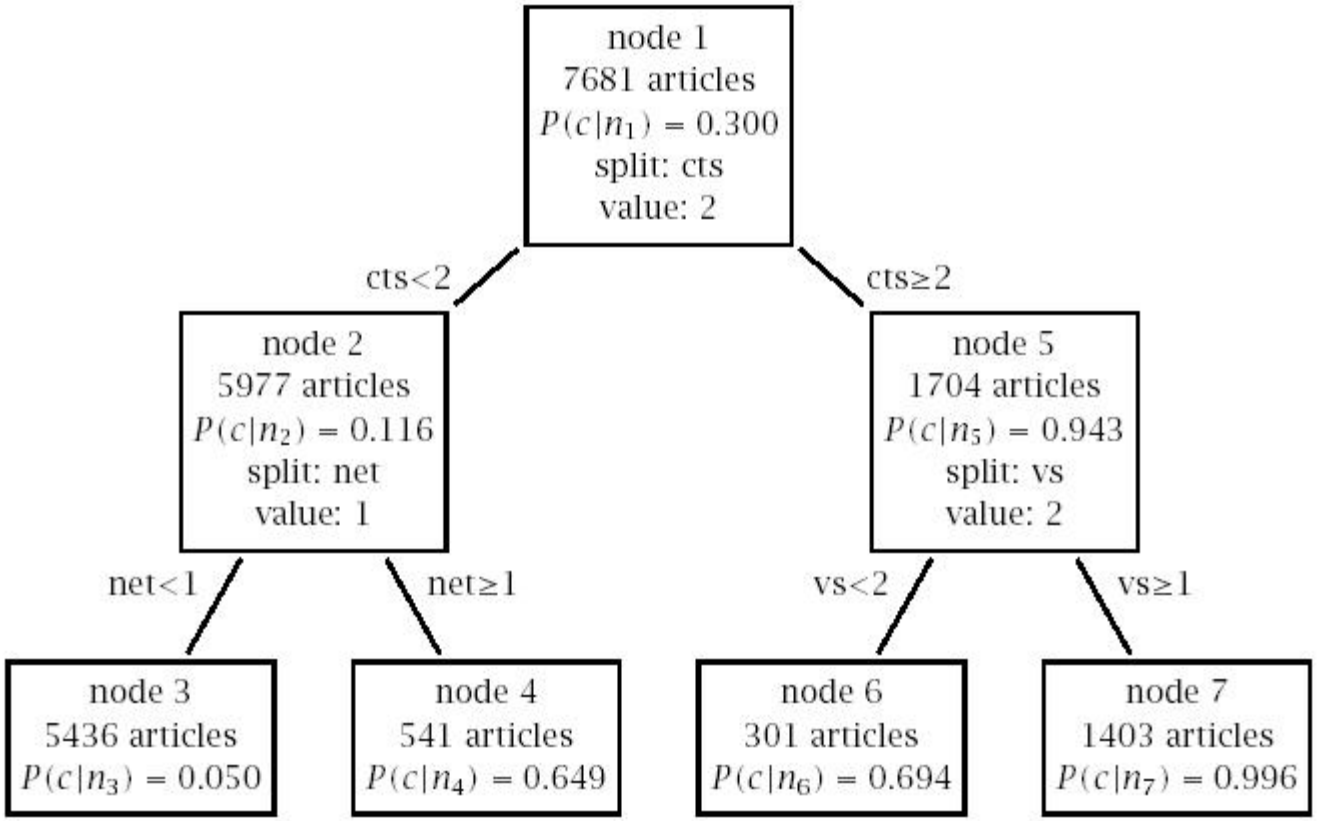


Figure 16.1 A decision tree. This tree determines whether a document is part of the topic category "earnings" or not. $P(c|n_i)$ is the probability of a document at node n_i to belong to the "earnings" category c.

Questions is of the form: Is the statement true?

- Is continuous variable X ≤ c ?
- Does categorical variable D take on levels i, j, or k?
  - e.g.Is geographic region 1, 2, 4, or 7? •
- Standard split:
  - If answer to question is YES a case goes left; otherwise it goes right
  - This is the form of all primary splits
- Question is formulated so that only two answers possible
  - Called binary partitioning
  - In CART the YES answer always goes left

From An Intro to Cart: http://www.researchmethods.org/CARTIntroTutorial.pdf

Heart Attack Risk Tree

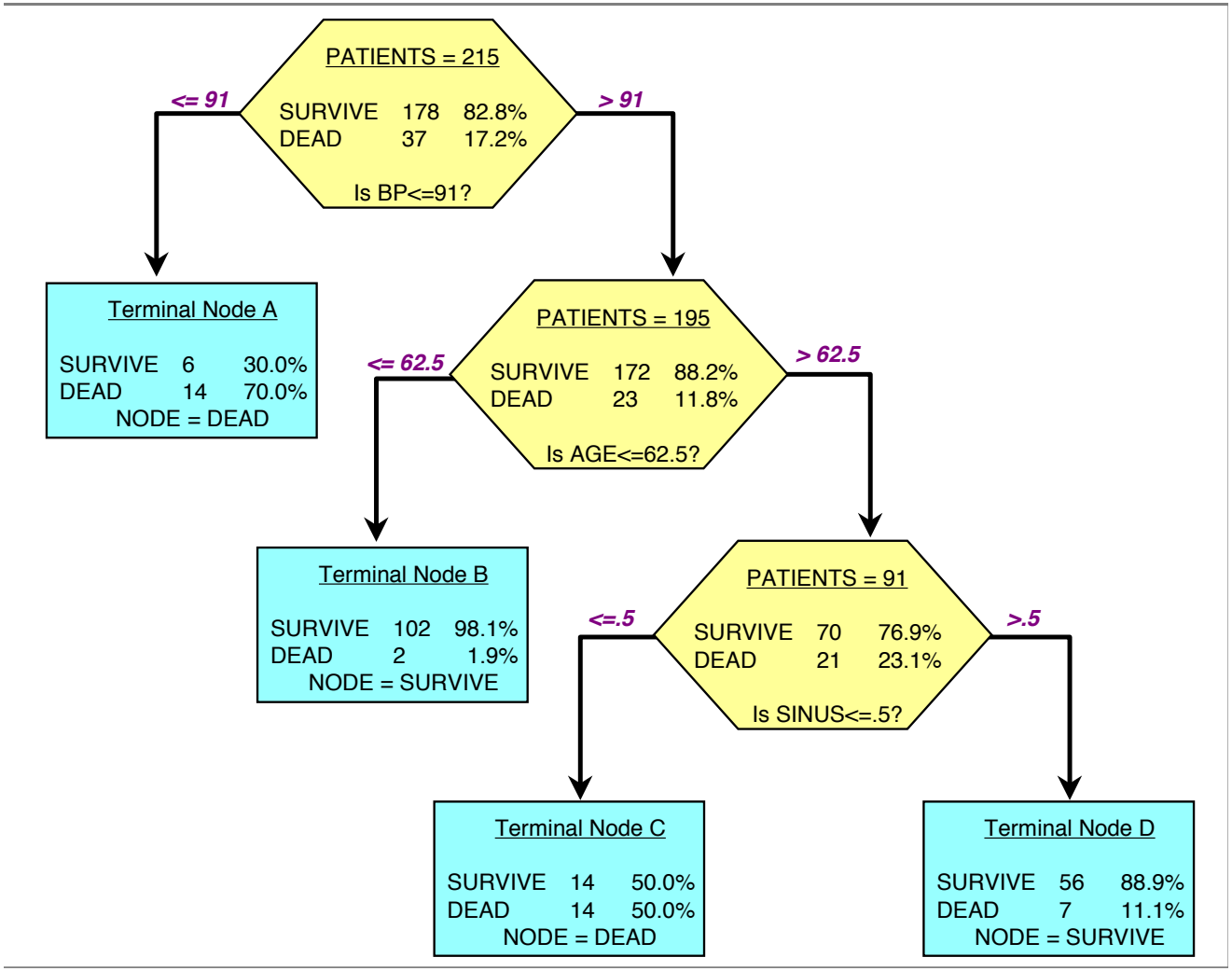
Example of a CLASSIFICATION tree

PATIENTS = 215

- **Dependent variable is categorical (SURVIVE, DIE)**

Terminal Node A
PATIENTS = 195

- **Want to predict class membership**



From An Intro to Cart: http://www.researchmethods.org/CARTIntroTutorial.pdf

Building Decision Trees

- Once we have a decision tree, it is straightforward to use it to assign labels to new input values.
- How we can build a decision tree that models a given training set?

Building Decision Trees

- The central focus of the decision tree growing algorithm is selecting which attribute to test at each node in the tree. The goal is to select the attribute that is most useful for classifying examples.
- Top-down, greedy search through the space of possible decision trees.
  - That is, it picks the best attribute and never looks back to reconsider earlier choices.

Building Decision Trees

- Splitting criterion
  - Finding the features and the values to split on
    - for example, why test first “cts” and not “vs”?
    - Why test on “cts < 2” and not “cts < 5” ?
  - Split that gives us the *maximum information gain* (or the *maximum reduction of uncertainty*)
- Stopping criterion
  - When all the elements at one node have the same class, no need to split further
- In practice, one first builds a large tree and then one prunes it back (to avoid overfitting)

- See *Foundations of Statistical Natural Language Processing*, Manning and Schuetze for a good introduction

Decision Trees: Strengths

- Decision trees are able to generate understandable rules.
- Decision trees perform classification without requiring much computation.
- Decision trees are able to handle both continuous and categorical variables.
- Decision trees provide a clear indication of which features are most important for prediction or classification.

Decision Trees: weaknesses

- Decision trees are prone to errors in classification problems with many classes and relatively small number of training examples.
  - Since each branch in the decision tree splits the training data, the amount of training data available to train nodes lower in the tree can become quite small.
- Decision tree can be computationally expensive to train.
  - Need to compare all possible splits
  - Pruning is also expensive

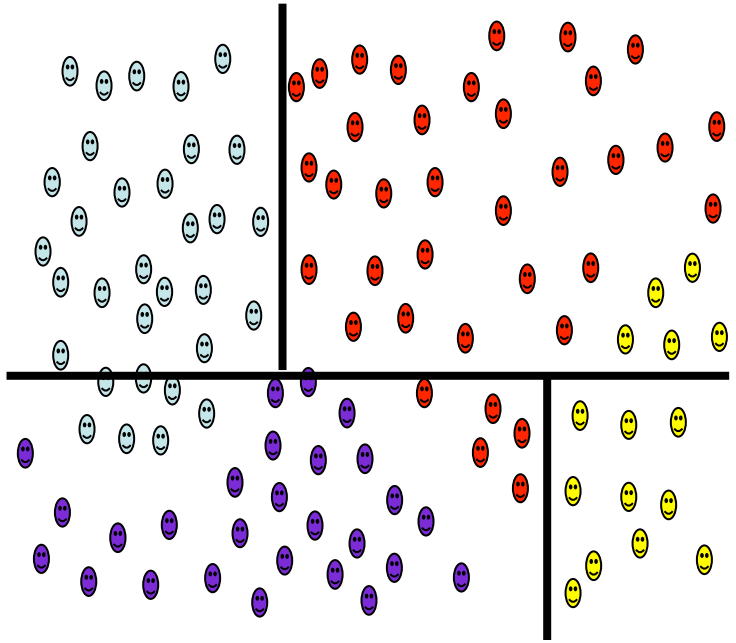
Decision Trees: weaknesses

- Most decision-tree algorithms only examine a single field at a time. This leads to rectangular classification boxes that may not correspond well with the actual distribution of records in the decision space.
  - The fact that decision trees require that features be checked in a specific order limits their ability to exploit features that are relatively independent of one another.
  - Naive Bayes overcomes this limitation by allowing all features to act "in parallel."

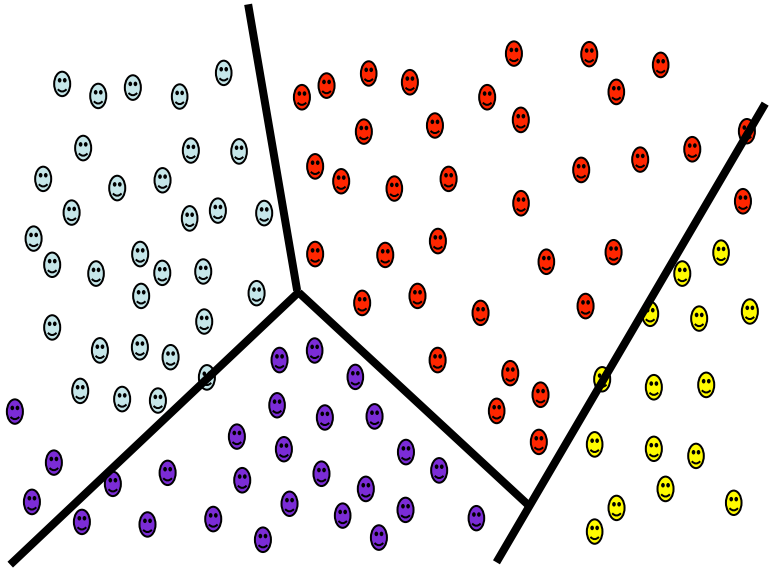
Naïve Bayes

More powerful that Decision Trees

Decision Trees



Naïve Bayes



Every feature gets a say in determining
which label should be assigned to a given input value.

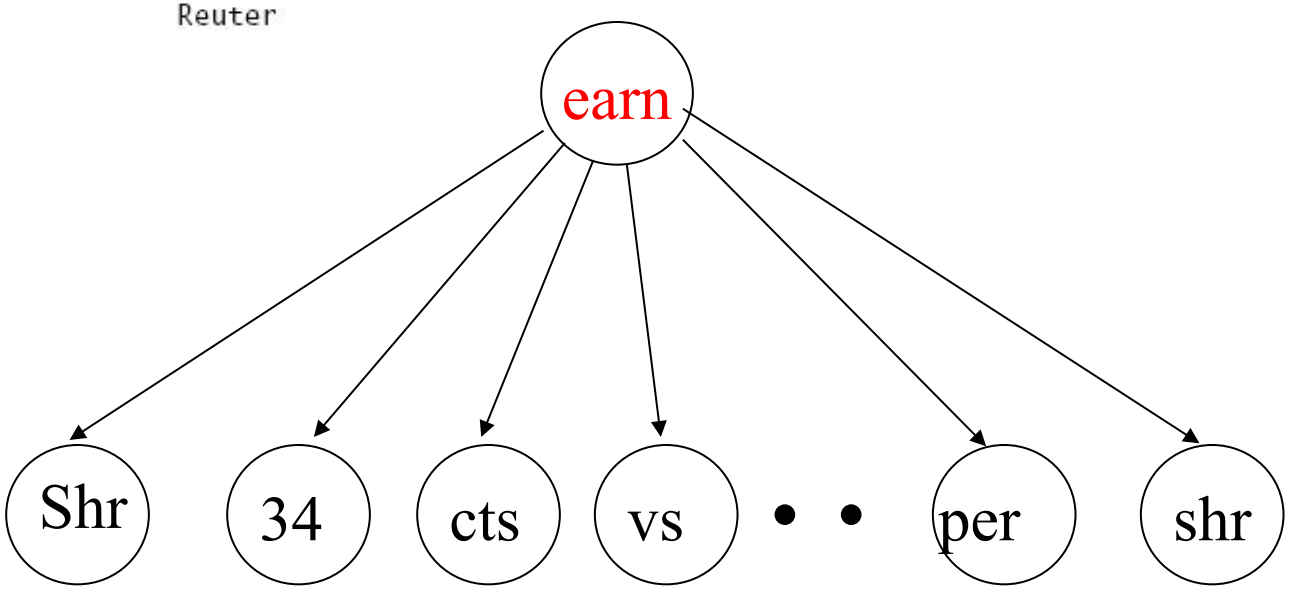
Naïve Bayes for text classification

```
<REUTERS NEWID="11">
<DATE>26-FEB-1987 15:18:59.34</DATE>
<TOPICS><D>earn</D></TOPICS>
<TEXT>
<TITLE>COBANCO INC &lt;CBCO> YEAR NET</TITLE>
<DATELINE>    SANTA CRUZ, Calif., Feb 26 - </DATELINE>
<BODY>Shr 34 cts vs 1.19 dlrs
    Net 807,000 vs 2,858,000
    Assets 510.2 mln vs 479.7 mln
    Deposits 472.3 mln vs 440.3 mln
    Loans 299.2 mln vs 327.2 mln
    Note: 4th qtr not available. Year includes 1985
extraordinary gain from tax carry forward of 132,000 dlrs,
or five cts per shr.
 Reuter
</BODY></TEXT>
</REUTERS>
```

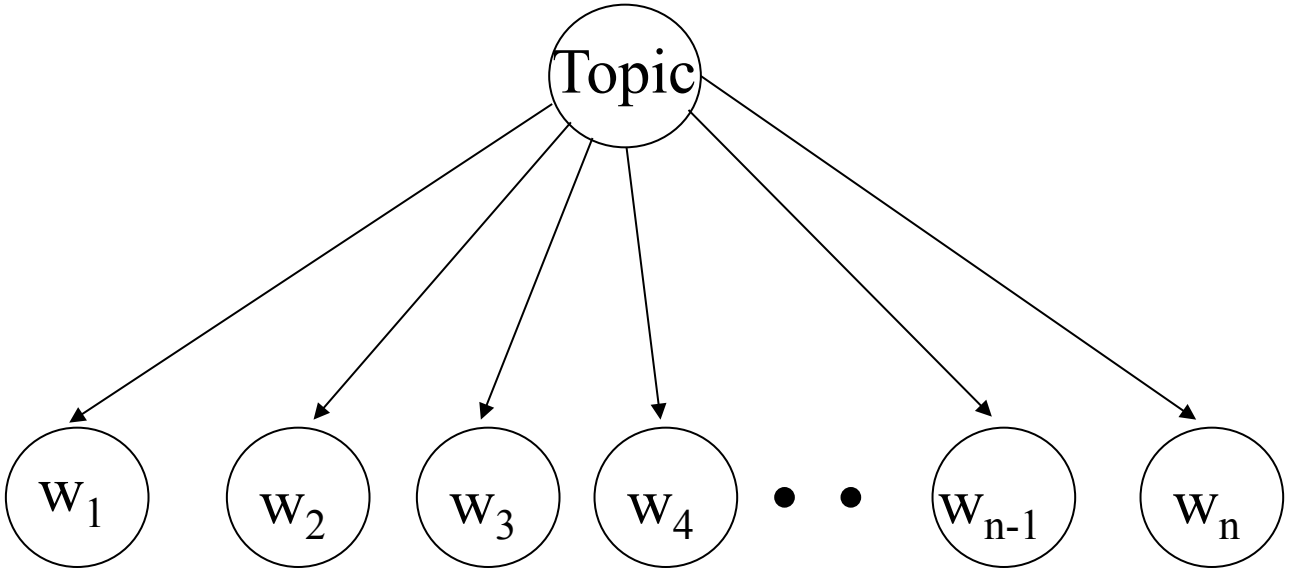
Figure 16.3 An example of a Reuters news story in the topic category "earnings." Parts of the original have been omitted for brevity.

Naïve Bayes for text classification

```
<TOPICS><D>earn</D></TOPICS>
<TEXT>
<TITLE>COBANCO INC &lt;CBCO> YEAR NET</TITLE>
<DATELINE>    SANTA CRUZ, Calif., Feb 26 - </DATELINE>
<BODY>Shr 34 cts vs 1.19 dlrs
    Net 807,000 vs 2,858,000
    Assets 510.2 mln vs 479.7 mln
    Deposits 472.3 mln vs 440.3 mln
    Loans 299.2 mln vs 327.2 mln
    Note: 4th qtr not available. Year includes 1985
extraordinary gain from tax carry forward of 132,000 dlrs,
or five cts per shr.
 Reuter
```

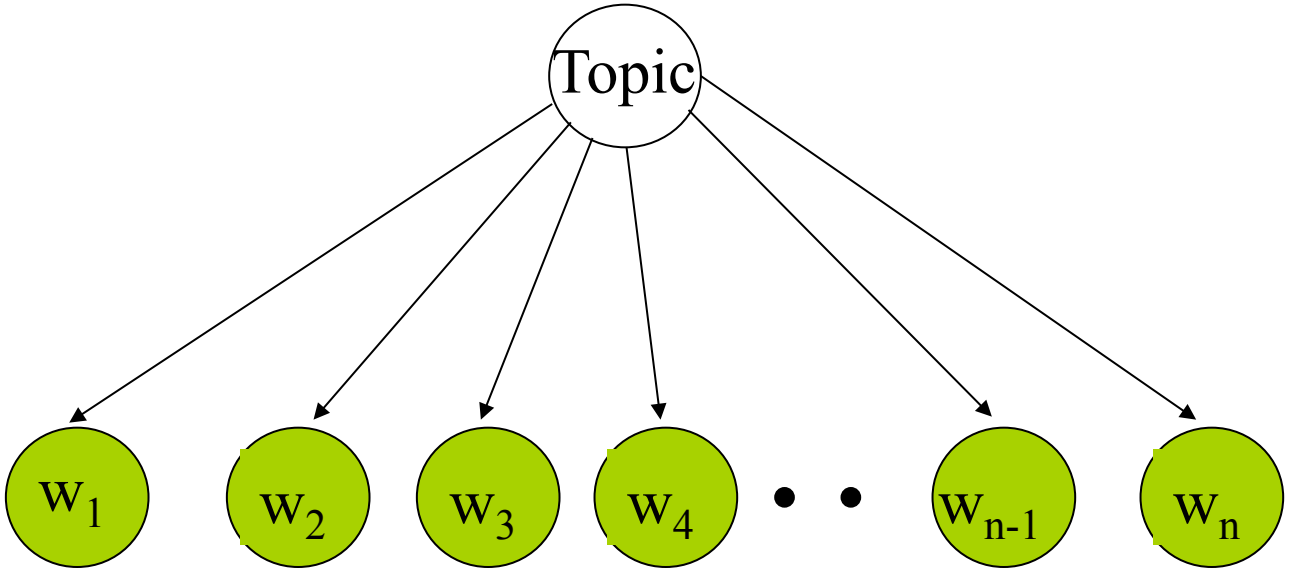


Naïve Bayes for text classification



- The words depend on the topic: $P(w_i|\ \text{Topic})$
  - $P(\text{cts} \mid \text{earn}) > P(\text{tennis} \mid \text{earn})$
- Naïve Bayes (aka independence) assumption: all words are independent given the topic
- From training set we learn the probabilities $P(w_i|\ \text{Topic})$ for each word and for each topic in the training set

Naïve Bayes for text classification



- To: Classify new example
- Calculate P(Topic | w_1, w_2, … w_n) for each topic
- Bayes decision rule:
  - Choose the topic T' for which
  - P(T' | w_1, w_2, … w_n) > P(T | w_1, w_2, … w_n) for each T≠ T'

Naïve Bayes: Strengths

- Very simple model
  - Easy to understand
  - Very easy to implement
- Can scale easily to millions of training examples (just need counts!)
- Very efficient, fast training and classification
- Modest space storage
- Widely used because it works really well for text categorization
- Linear, but non parallel decision boundaries

Naïve Bayes: weaknesses

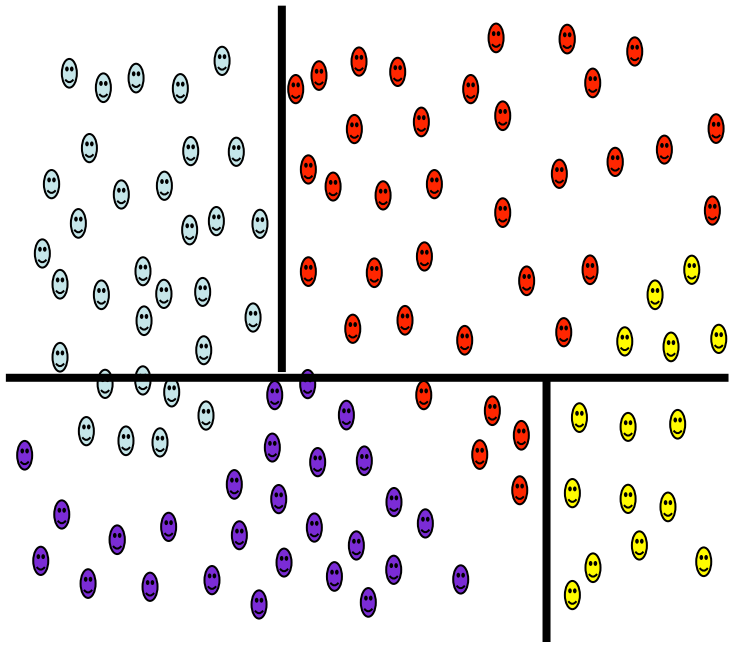
- Naïve Bayes independence assumption has two consequences:
  - The linear ordering of words is ignored (*bag of words* model)
  - The words are independent of each other given the class: False
    - *President* is more likely to occur in a context that contains *election* than in a context that contains *poet*
- Naïve Bayes assumption is inappropriate if there are strong conditional dependencies between the variables
- (But even if the model is not "right", Naïve Bayes models do well in a surprisingly large number of cases because often we are interested in *classification accuracy* and not in accurate *probability estimations*)
- Does not optimize prediction accuracy

The naivete of independence

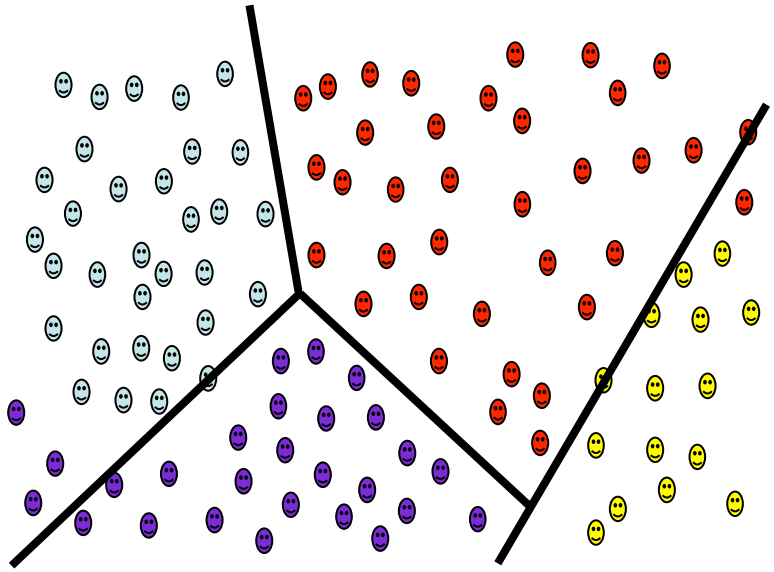
- Naïve Bayes assumption is inappropriate if there are strong conditional dependencies between the variables
- One problem that arises is that the classifier can end up "double-counting" the effect of highly correlated features, pushing the classifier closer to a given label than is justified.
- Consider a name gender classifier
- For example, the features ends-with(a) and ends-with(vowel) are dependent on one another, because if an input value has the first feature, then it must also have the second feature. For features like these, the duplicated information may be given more weight than is justified by the training set.

Maximum Entropy

Decision Trees



Naïve Bayes &
Maximum Entropy

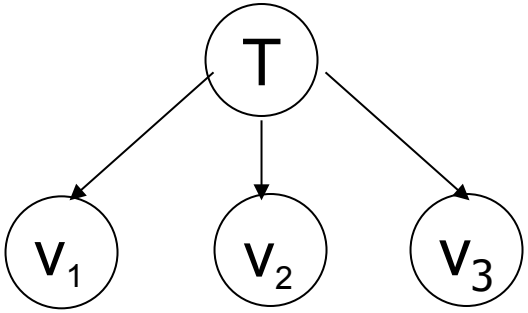


Maximum Entropy: Multi class, linear classifier
Difference with Naïve Bayes:

Maximum Entropy is a conditional model

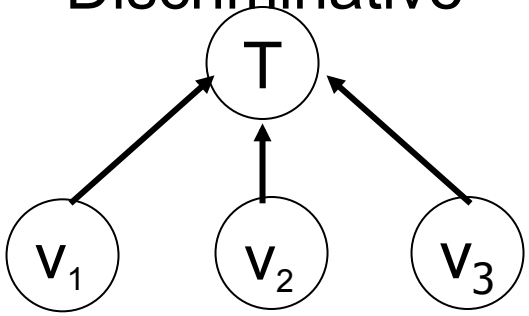
Generative vs Conditional Classifiers

Generative



- Builds a model that predicts ***P(input, label)*** ---the joint probability of a *(input, label)* pair.
- Examples:
- Naïve Bayes

Discriminative



- Aka discriminative
- Builds a model that predicts ***P(label | input)*** — the probability of a label *given* the input value.
- Examples:
- Maximum Entropy

Generative vs Conditional

- Generative P(input, label): can be used to answer the following questions:

1. What is the most likely label for a given input?
2. How likely is a given label for a given input?
3. What is the most likely input value?
4. How likely is a given input value?
5. How likely is a given input value with a given label?
6. What is the most likely label for an input that might have one of two values (but we don't know which)?

- Conditional P(label | input): can be used to answer the following questions:

1. What is the most likely label for a given input?
2. How likely is a given label for a given input?

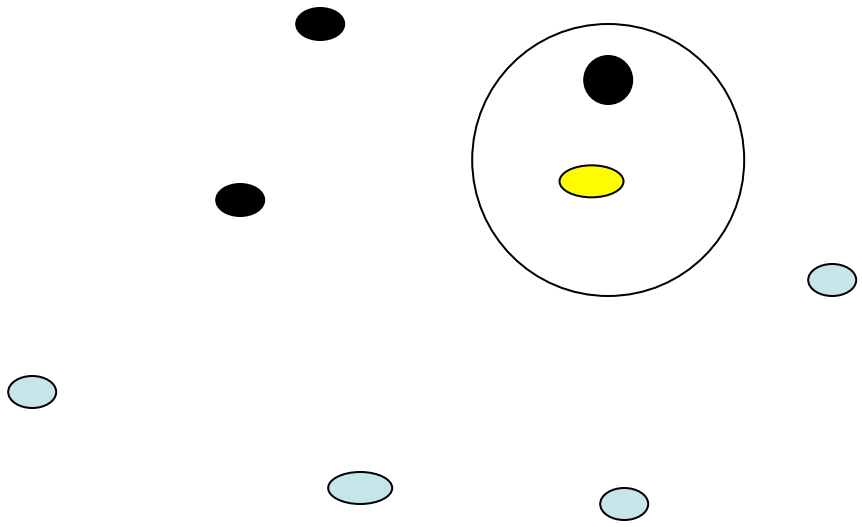
Generative vs Conditional

- **Generative P(input, label) --- Conditional P(label | input)**
- Generative models are strictly more powerful than conditional models, since we can calculate the conditional probability *P(label|input)* from the joint probability *P(input, label)*, but not vice versa
- This additional power comes at a price. Because the model is more powerful, it has more "free parameters" which need to be learned.
- Thus, when using a more powerful model, we end up with less data that can be used to train each parameter's value, making it harder to find the best parameter values.
- As a result, a generative model may not do as good a job at answering questions 1 and 2 as a conditional model, since the conditional model can focus its efforts on those two questions.
- However, if we do need answers to questions like 3-6, then we have no choice but to use a generative model.

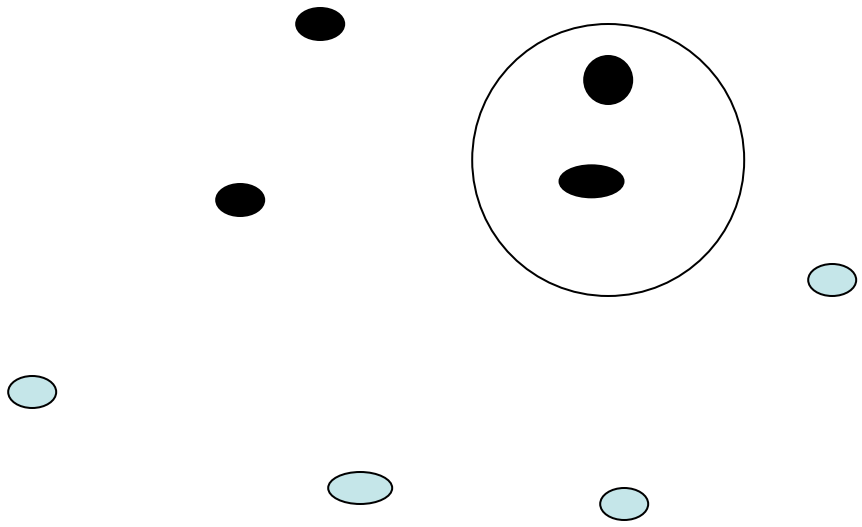
k Nearest Neighbor Classification

- **Instance-based method**
- Nearest Neighbor classification rule: to classify a new object, find the object in the training set that is *most similar* (i.e. closest in the vector space*).*
- Then assign the category of this nearest neighbor
- K Nearest Neighbor (KNN): consult k nearest neighbors. Decision based on the majority category of these neighbors. More robust than k = 1 (nearest Neighbor)
- Example of similarity measure often used in NLP is cosine similarity

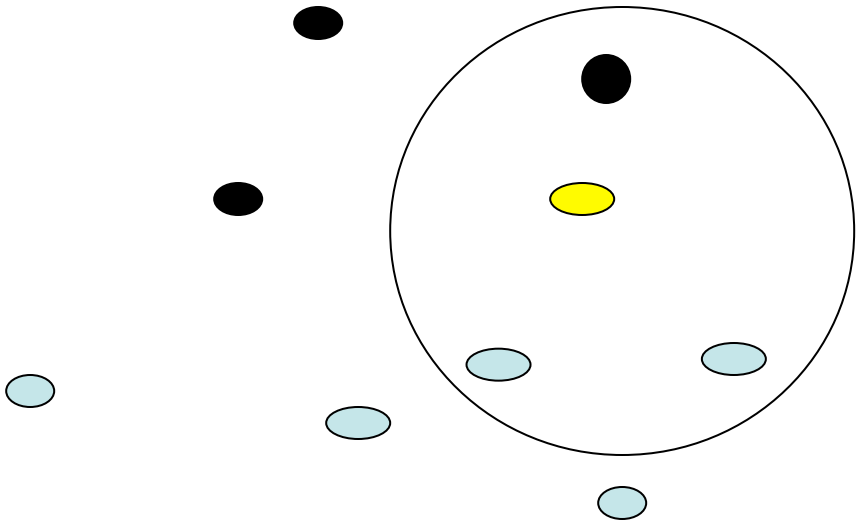
1-Nearest Neighbor



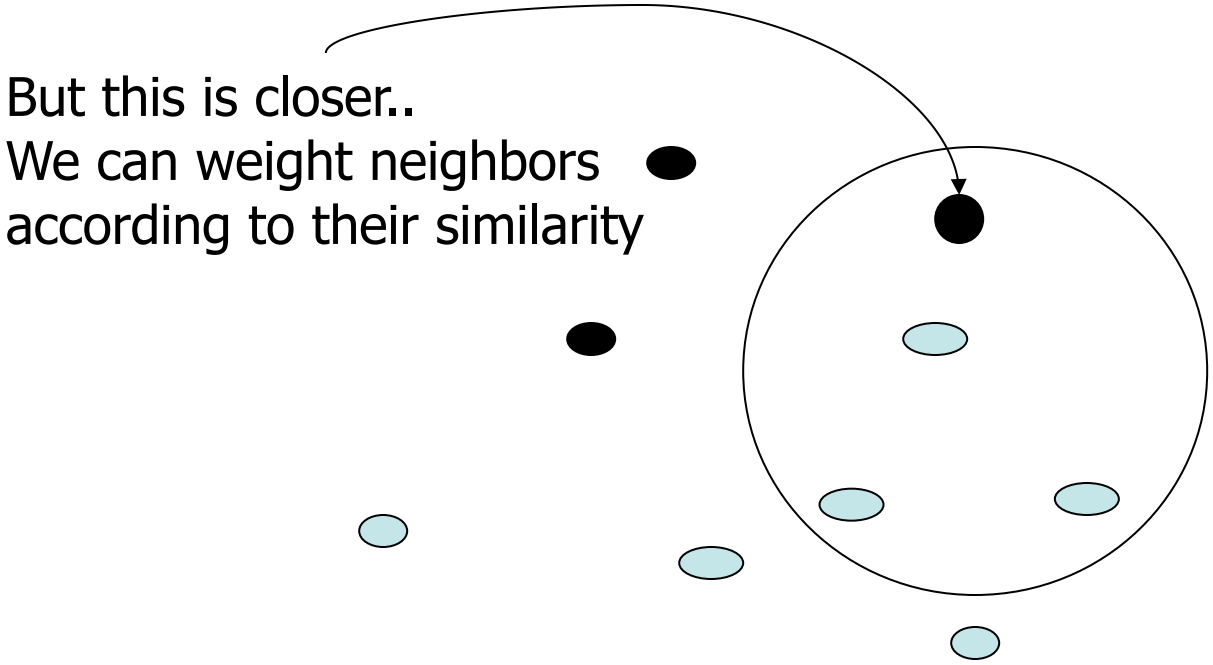
1-Nearest Neighbor



3-Nearest Neighbor



3-Nearest Neighbor



Assign the category of the majority of the neighbors

k Nearest Neighbor Classification

- Strengths
  - Robust
  - Conceptually simple
  - Often works well (used in computer vision a lot)
  - Powerful (arbitrary decision boundaries)
  - Very fast training (just build data structures)
- Weaknesses
  - Performance is very dependent on the similarity measure used (and to a lesser extent on the number of neighbors k used)
  - Finding a good similarity measure can be difficult
  - Computationally expensive for prediction
  - Not best accuracy among classifiers

What do models tell us?

- Descriptive models vs. explanatory models.
- Descriptive models capture patterns and correlations in the data but they don't provide any information about *why* the data contains those patterns
- Explanatory models attempt to capture properties and relationships that cause the linguistic patterns.

What do models tell us?

Google hits	*adore*	*love*	*like*	*prefer*
absolutely	289,000	905,000	16,200	644
definitely	1,460	51,000	158,000	62,600
ratio	198:1	18:1	1:10	1:97

- For example, we might introduce the abstract concept of "polar adjective", as one that has an extreme meaning, and categorize some adjectives like *adore* and *detest* as polar.
- Our explanatory model would contain the constraint that *absolutely* can only combine with polar adjectives, and *definitely* can only combine with non-polar adjectives.

What do models tell us?

- Most models that are automatically constructed from a corpus are descriptive models
- If our goal is to understand the linguistic patterns, then we can use this information about which features are related as a starting point for further experiments designed to tease apart the relationships between features and patterns.
- On the other hand, if we're just interested in using the model to make predictions (e.g., as part of a language processing system), then we can use the model to make predictions about new data without worrying about the details of underlying causal relationships.

Summary

- Algorithms for Classification
- Linear versus non linear classification

- Binary Linear
  - Perceptron, Winnon, Large margin classifier
- Binary Non Linear
  - Kernel methods

- Multi-class Linear
  - Decision Trees, Naïve bayes, Maximum Entropy
- Multi-class Non Linear
  - K-nearest neighbors

Methods

- **Linear Models**:
  - Perceptron & Winnow (neural networks)
  - Large margin classifier
    - Support Vector Machine (SVM)
- **Probabilistic models**:
  - Naïve Bayes
  - Maximum Entropy Models
- **Decision Models**:
  - Decision Trees
- **Instance-based methods**:
  - Nearest neighbor

Classification

- Define classes/categories
- Label text
- Extract features
- Choose a classifier
  - Naive Bayes Classifier
  - NN (i.e. perceptron)
  - Maximum Entropy
  - ….
- Train it
- Use it to classify new examples

How do you choose the classifier?

- Binary vs. multi-class
- Try different ones (linear –not linear) and evaluate
- Pros and cons
  - For example: do you care about being fast at decision time? (*k*-Nearest Neighbor is not…)
  - Fast training? (decision tree training is slow)
  - Do you care about interpretable rules (decision trees) or only classification accuracy?