Classification

Compiled by Sujatha Das G & Cornelia Caragea

August 19, 2014

Credits for slides: Allan, Arms, Manning, Lund, Noble, Page, Jurafsky.

Terminology Used in Classification Problems

- Features are attributes of a given problem
- Training/Labeled examples with known labels
- Unlabeled/Test examples on which labels need to be predicted
- Set of "classes" (prediction variable, categoric not numeric)

Training: Learn a model/hypothesis/function based on training data

Testing: Predict classes for test examples using the learnt model

Examples are represented using "features", model is made up of "parameters" .

Learning is the process of estimating parameters for each feature!

Example Classification Problems

- Email filtering: spam / non spam
- Email foldering/tagging: Work, Friends, Family, Hobby
- Research articles by subject area: Law, Politics, Computer Science, Biology
- Document by type: research article, thesis, slides, CV
- Tumor: malignant / benign
- Medical diagnosis: Not ill, Cold, Flu

Data Representation

- N = number of training examples
- x's = "input" variable / features
- y's = "output" variable / "target" variable
- (x,y) – one training example
- $(x^{(i)},y^{(i)})$ – the i^{th} training example
- (x, ?) – test example

Data Representation

http://www.d.umn.edu/~padhy005/Chapter5.html

Independent Variables			Dependent Variable
Outlook	**Humidity**	**Number of batsmen in team > 6**	**Final Outcome**
Sunny	High	Yes	Won
Overcast	High	No	Lost
Sunny	Low	No	Lost
Sunny	High	No	Won
Overcast	Low	Yes	Lost
Sunny	Low	Yes	Won
Sunny	Low	No	Lost
Sunny	High	No	Won
Sunny	Low	Yes	Won
Sunny	Low	Yes	Won

Tree-based Classifiers

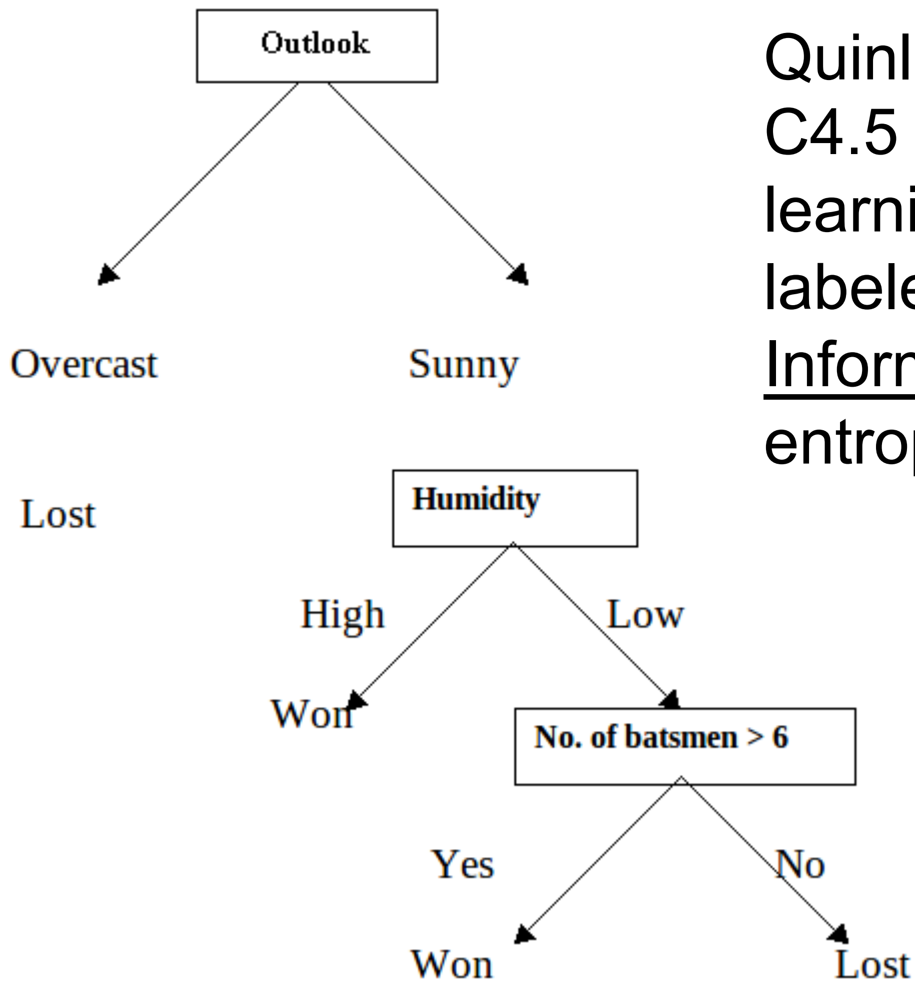
- Conjunctive Rules
  - If the outlook=sunny and humidity=high and #batsmen<6, outcome=lost…
- Usually not possible to manually build an accurate rule set for large problems .

Large problems (lot of attributes, lot of instances)

- What if the attributes are numeric?
- What if we need a probability of winning vs losing?

Decision Trees and other variants capture express a model as a set of predicate rules/conjunctions of features

How can automatically learn a decision tree?



Quinlan proposed the ID3 and C4.5 techniques for automatically learning Decision Trees from labeled data using concepts from Information Theory such as entropy and information gain.

http://www.d.umn.edu/~padhy005/Chapter5.html

Bayesian Methods

- Learning and classification methods based on Probability Theory.
- Bayes theorem plays a critical role in probabilistic learning and classification.

- Build a *generative model* that approximates how data is produced
- Use *prior* probability of each category given no information about an item.
- Categorization produces a *posterior* probability distribution over the possible categories given a description of an item.

Generative Probabilistic Classifiers

- Assume a simple (usually unrealistic for the purpose of tractability) probabilistic method by which the data was generated.
- Each class has a different parameterized generative model that characterizes that class.
- **Training**: Use the data for each category to estimate the parameters of the generative model for that category.
- **Testing**: Use Bayesian analysis to determine the category model that most likely generated a specific test instance.

Bayes Theorem

$$P(c|x) = \frac{P(x|c)\,P(c)}{P(x)}$$

Need to work with probability distributions

For example, Multinomial for words, Gaussian for numeric...

In Text classification, terms distributions are usually expressed using multinomial distributions

Next slides by Jurafsky and Manning

http://web.stanford.edu/class/cs124/lec/

Text Classification: definition

- *Input*:
  - a document d
  - a fixed set of classes $C = \{c_1, c_2, \ldots, c_J\}$

- *Output*: a predicted class $c \in C$

Positive or negative movie review?

- unbelievably disappointing

- Full of zany characters and richly applied satire, and some great plot twists

- this is the greatest screwball comedy ever filmed

- It was pathetic. The worst part about it was the boxing scenes.

The bag of words representation

$\gamma($

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun… It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.

$)=c$

The bag of words representation

$\gamma($

great	2
love	2
recommend	1
laugh	1
happy	1
...	...

$)=c$

Naïve Bayes Classifier (I)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(c \mid d)$$

MAP is "maximum a posteriori" = most likely class

$$= \underset{c \in C}{\operatorname{argmax}} \frac{P(d \mid c)P(c)}{P(d)}$$

Bayes Rule

$$= \underset{c \in C}{\operatorname{argmax}} P(d \mid c)P(c)$$

Dropping the denominator

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(d \mid c)P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c)P(c)$$

Document d represented as features x1..xn

Multinomial Naïve Bayes Independence Assumptions

$$P(x_1, x_2, \ldots, x_n \mid c)$$

- **Bag of Words assumption**: Assume position doesn't matter
- **Conditional Independence**: Assume the feature probabilities $P(x_i|c_j)$ are independent given the class c.

$$P(x_1, \ldots, x_n \mid c) = P(x_1 \mid c) \bullet P(x_2 \mid c) \bullet P(x_3 \mid c) \bullet \ldots \bullet P(x_n \mid c)$$

Learning the Multinomial Naïve Bayes Model

- First attempt: maximum likelihood estimates
  - simply use the frequencies in the data

$$\hat{P}(c_j) = \frac{doccount(C = c_j)}{N_{doc}}$$

$$\hat{P}(w_i \mid c_j) = \frac{count(w_i, c_j)}{\sum_{w \in V} count(w, c_j)}$$

Dan Jurafsky

Laplace (add-1) smoothing for Naïve Bayes

$$\hat{P}(w_i \mid c) = \frac{count(w_i, c) + 1}{\sum_{w \in V} \left(count(w, c) + 1 \right)}$$

$$= \frac{count(w_i, c) + 1}{\left(\sum_{w \in V} count(w, c) \right) + |V|}$$

Multinomial Naïve Bayes: Learning

- From training corpus, extract *Vocabulary*

- Calculate $P(c_j)$ terms
  - For each c_j in C do

    $docs_j \leftarrow$ all docs with class $= c_j$

    $$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$

- Calculate $P(w_k \mid c_j)$ terms
  - $Text_j \leftarrow$ single doc containing all $docs_j$
  - For each word w_k in *Vocabulary*

    $n_k \leftarrow$ # of occurrences of w_k in $Text_j$

    $$P(w_k \mid c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha |Vocabulary|}$$

Naive Bayes is Not So Naive

- Naïve Bayes: First and Second place in KDD-CUP 97 competition, among 16 (then) state of the art algorithms

  Goal: Financial services industry direct mail response prediction model: Predict if the recipient of mail will actually respond to the advertisement – 750,000 records.

- Robust to Irrelevant Features

  Irrelevant Features cancel each other without affecting results

- Very good in domains with many <u>equally important</u> features
- A good baseline for text classification!
- Very Fast: Learning with one pass of counting over the data; testing linear in the number of attributes, and document collection size
- Low Storage requirements

Discriminative Classifiers

Examples: Probabilistic (Logistic Regression) and Support Vector Machines

- Typically, more accurate on classification tasks, model class-conditional distribution directly
- No assumptions on underlying distributions, able to incorporate arbitrary features (that link an observation with a class)
  - “word=Africa and isCapitalized, feature value=1”

Feature-based Linear Classifiers

- Each (feature, class) has associated weight parameter
- Typically, parameter estimation involves optimization of a loss function and is more tricky involving numerical optimization techniques
- Prediction depends on a function of the dot product between the weight vector and the feature vector

- For example in Logistic Regression

$$P(c \mid d, \lambda) = \frac{\exp \sum_i \lambda_i f_i(c, d)}{\sum_{c'} \exp \sum_i \lambda_i f_i(c', d)}$$

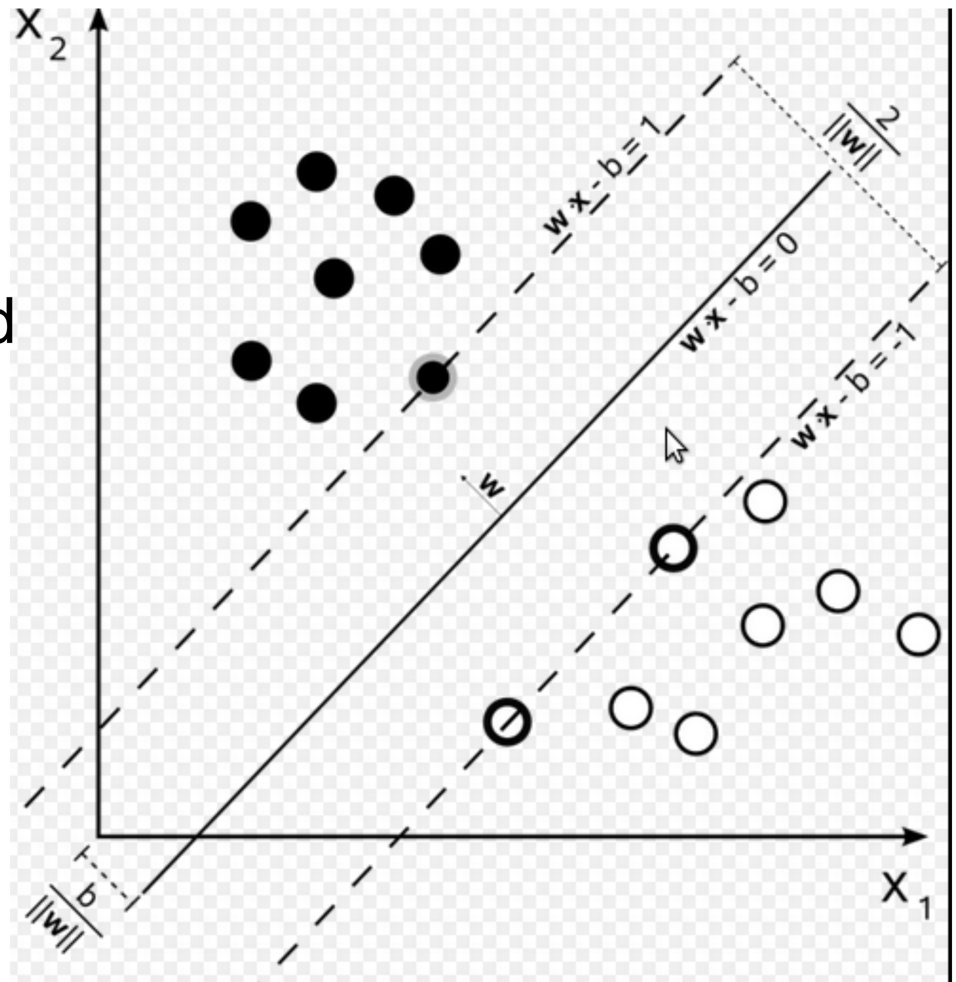
Parameter Estimation

Logistic Regression:

Find the weight vector that maximizes a quantity called conditional log likelihood

SVM:

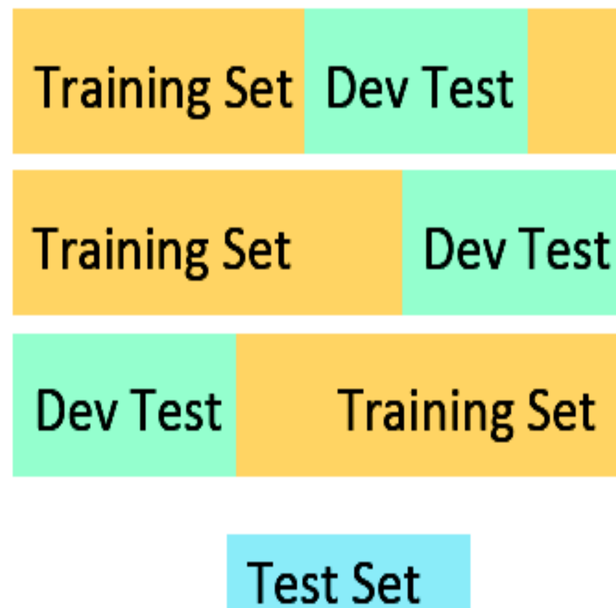
Find the weight vector that maximizes the distance between hyperplanes



Dan Jurafsky

Development Test Sets and Cross-validation

- Metric: P/R/F1 or Accuracy
- Unseen test set
  - avoid overfitting ('tuning to the test set')
  - more conservative estimate of performance
- Cross-validation over multiple splits
    - Handle sampling errors from different datasets
  - Pool results over each split
  - Compute pooled dev set performance



Confusion matrix c

- For each pair of classes <c_1,c_2> how many documents from c_1 were incorrectly assigned to c_2?
  - $c_{3,2}$: 90 wheat documents incorrectly assigned to poultry

Docs in test set	Assigned UK	Assigned poultry	Assigned wheat	Assigned coffee	Assigned interest	Assigned trade
True UK	95	1	13	0	1	0
True poultry	0	1	0	0	0	0
True wheat	10	90	0	1	0	0
True coffee	0	0	0	34	3	7
True interest	-	1	2	13	26	5
True trade	0	0	2	14	5	10

Per class evaluation measures

Recall:

Fraction of docs in class *i* classified correctly: $\dfrac{c_{ii}}{\sum_j c_{ij}}$

Precision:

Fraction of docs assigned class *i* that are actually about class *i*: $\dfrac{c_{ii}}{\sum_j c_{ji}}$

Accuracy: (1 - error rate)

Fraction of docs classified correctly: $\dfrac{\sum_i c_{ii}}{\sum_j \sum_i c_{ij}}$