Machine Learning in IR

CISC489/689-010, Lecture #21

Monday, May 4th

Ben Carterette

Machine Learning

- Basic idea:
  - Given instances x in a space X and values y
  - "Learn" a function f(x) that maps instances to values
- Recall classification:
  - Given:
    - a description of an instance x in X, where X is the *instance space*
    - a fixed set of classes C = {c_1, c_2, …, c_k}
  - Determine:
    - the class x belongs to: f(x) in C, where f(x) is a *classification function* with domain X and range C
- The goal is to learn functions that are *generalizable*
  - They can correctly predict things about data never seen before

Simple Example

- Non-classification example
- Linear regression:
  - Data is a real "dependent variable" y and a vector of "independent variables" (or "covariates", or "features") **x**
  - Learn linear function $y_i = f(\mathbf{x_i}) = b_0 + b_1 x_{i1} + b_2 x_{i2} + \ldots$
    - Function is "learned" by solving least squares in terms of vector **b:**

$$\min_{\mathbf{b}} \sum (y_i - \mathbf{b}'\mathbf{x}_i)^2$$

  - For a new instance $\mathbf{x_j}$, predict $y = f(\mathbf{x_j}) = b_0 + b_1 x_{j1} + b_2 x_{j2} + \ldots$

Machine Learning Tasks

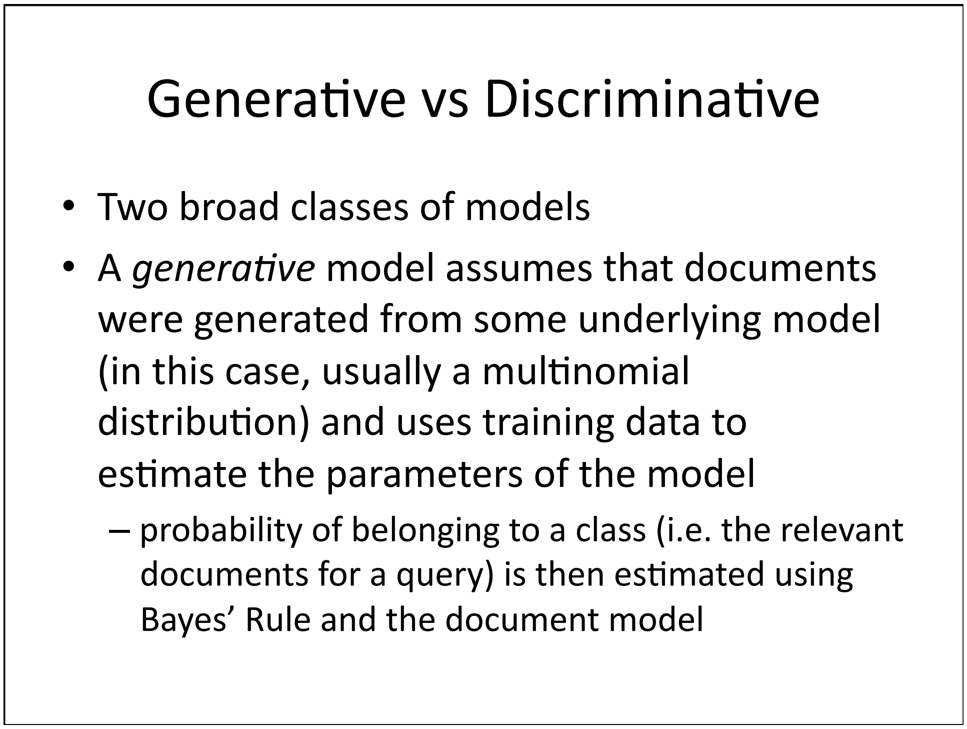
- Regression
  - Function f(X) outputs a real number
  - Linear regression, generalized linear models, neural networks
- Classification
  - Function f(X) outputs a discrete class prediction
  - We've talked about this some: Naïve Bayes classifier
  - Today: SVMs
- Ranking
  - Function f(X) outputs a ranking or partial ranking
  - RankSVM, RankNet, and more

Training and Test Data

- It’s pretty easy to fit a function precisely to any given set of data
- Doing so does not guarantee a *generalizable* function, though
  - Overfitting: no ability to predict anything about new data
- Since the goal is to learn generalizable functions, we need to have some data we can test our machine-learned function on
- *Training* and *test* splits:
  - Given n instances ($\mathbf{x}_i$, y_i), use n-k to learn the function
  - Use the remaining k to test it

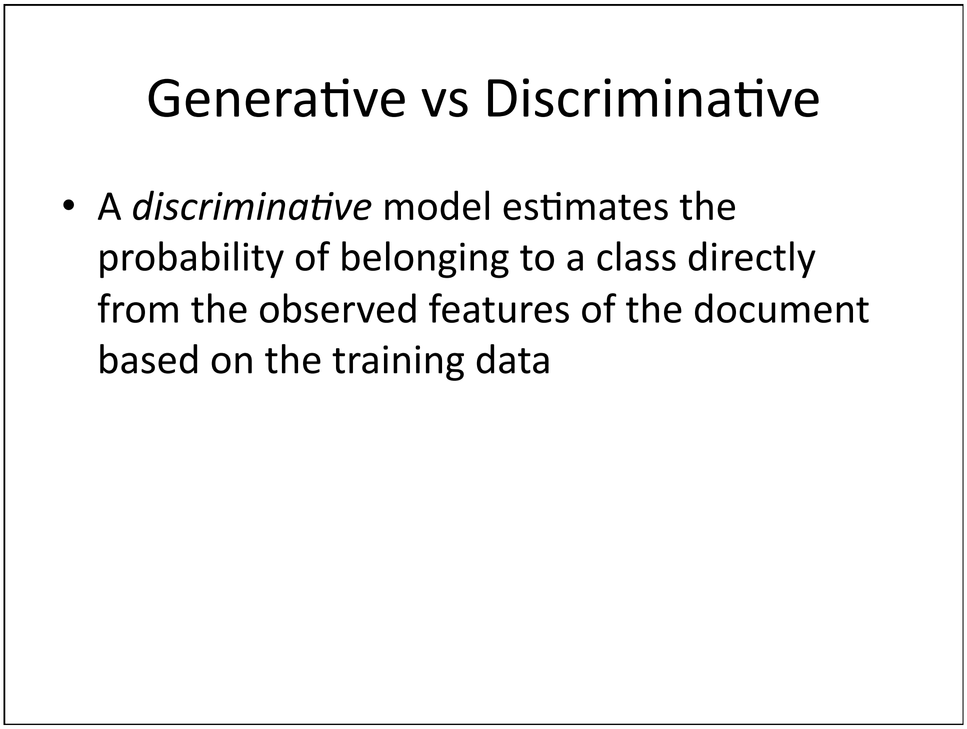
Machine Learning and IR

- Considerable interaction between these fields
  - Rocchio algorithm (60s) is a simple learning approach
  - 80s, 90s: learning ranking algorithms based on user feedback
  - 2000s: text categorization
- Limited by amount of training data
- Web query logs have generated new wave of research
  - e.g., “Learning to Rank”



Generative vs Discriminative

- Two broad classes of models
- A *generative* model assumes that documents were generated from some underlying model (in this case, usually a multinomial distribution) and uses training data to estimate the parameters of the model
  - probability of belonging to a class (i.e. the relevant documents for a query) is then estimated using Bayes' Rule and the document model



Generative vs Discriminative

- A *discriminative* model estimates the probability of belonging to a class directly from the observed features of the document based on the training data

Examples

- Generative classifier:
  - Recall Naïve Bayes

$$P(y_i|\mathbf{x}_i) = P(\mathbf{x}_i|y_i)P(y_i) = P(y_i)\prod_j P(\mathbf{x}_{ij}|y_i)$$

Documents “generated” from class model for class y_i

- Discriminative classifier:
  - Logistic regression

$$P(y_i|\mathbf{x}_i) = \frac{\exp(b_0 + \sum_j b_j \mathbf{x}_{ij})}{1 + \exp(b_0 + \sum_j b_j \mathbf{x}_{ij})}$$

Training a Classifier

- Generative:
  - Training works by estimating model probabilities from training data
  - For Naïve Bayes, we estimate $P(x_{ij} \mid y_i)$ using the frequency of x_{ij} in documents with class y_i
- Discriminative:
  - Training works by finding feature weights **b**
  - Usually by solving a minimization problem defined by the data
  - For logistic regression, the minimization problem is

$$\min_{\mathbf{b}} \prod_i \left(\frac{\exp(b_0 + \sum_j b_j \mathbf{x}_{ij})}{1 + \exp(b_0 + \sum_j b_j \mathbf{x}_{ij})}\right)^{y_i} \left(\frac{1}{1 + \exp(b_0 + \sum_j b_j \mathbf{x}_{ij})}\right)^{1-y_i}$$

Generative vs. Discriminative

- All of the probabilistic retrieval models presented so far fall into the category of *generative models*
- Generative models perform well with low numbers of training examples
- Discriminative models usually have the advantage given enough training data
  - Can also easily incorporate many features

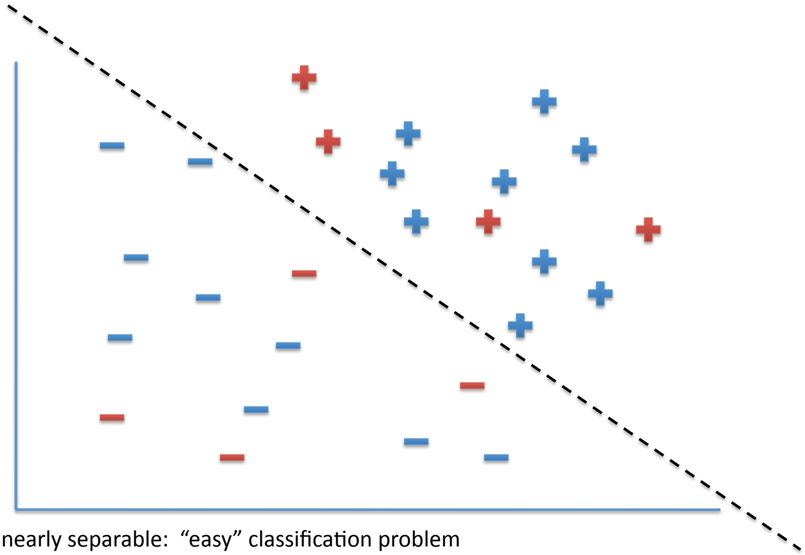
Discriminative Models for IR

- Discriminative models can be trained using explicit relevance or class judgments or click data in query logs
  - Click data is much cheaper, more noisy
  - e.g. Ranking Support Vector Machine (SVM) takes as input *partial rank* information for queries
    - partial information about which documents should be ranked higher than others
    - More on Wednesday

Support Vector Machines

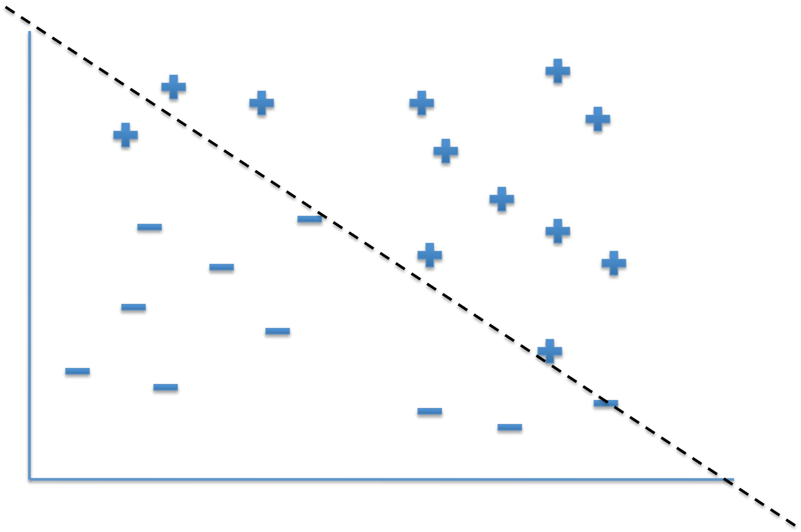
- SVMs are a popular type of discriminative model
- Their goal is to find a hyperplane separating positive instances from negative instances
- Additionally, this hyperplane has maximum distance from the most difficult-to-classify points
  - *Maximum-margin classifier*
- If the data is linearly separable, this hyperplane is optimal

Illustration

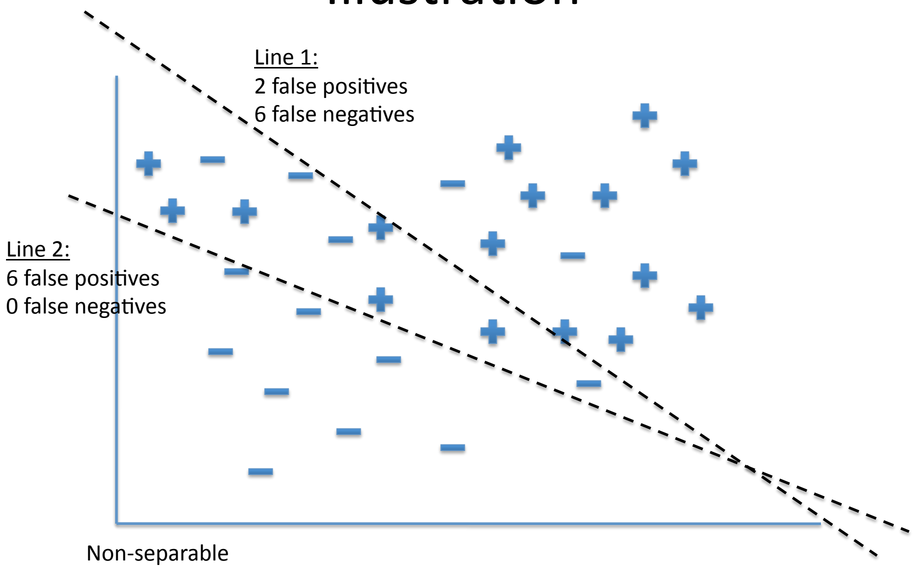


Linearly separable: “easy” classification problem

Illustration



Illustration



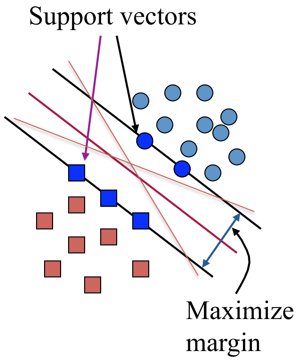
Line 1:
2 false positives
6 false negatives

Line 2:
6 false positives
0 false negatives

Non-separable

SVM Idea

- There are many possible separating hyperplanes
- To find the optimal one, locate the points that are closest to the decision boundary
- These points are the *support vectors*
- Find the hyperplane **w** that has maximum distance from the support vectors



SVM Optimization

- Finding the hyperplane can be expressed as a quadratic optimization problem

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\mathbf{w}'\mathbf{w}$$

$$\text{s.t. } y_i(\mathbf{w}'x_i + b) \geq 1 \text{ for all } (x_i, y_i)$$

$\mathbf{w}$ is the hyperplane

b is a bias term (intercept)

x_i is an instance (a feature vector)

y_i is the true class (1 or -1)

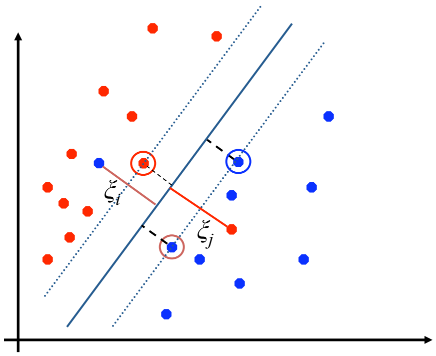
Soft-Margin SVM

- In text classification problems, the data is seldom linearly separable
  - Why? Choice of features to describe it, but also disagreement between people about which class something belongs to
- Modify optimization problem to include *slack variables*
- These slack variables allow for some errors in placing the separating hyperplane

Soft-Margin Optimization

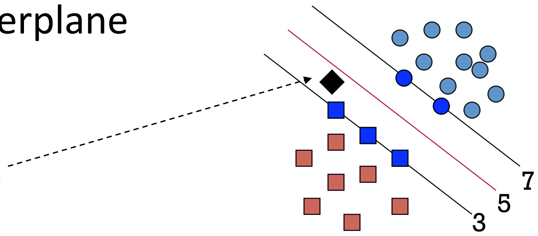
$$\min_{\mathbf{w},b} \quad \frac{1}{2}\mathbf{w}'\mathbf{w} + C\sum \zeta_i$$
$$\text{s.t. } y_i(\mathbf{w}'x_i + b) \geq 1 - \zeta_i$$

ζ_i is a slack variable for instance i



Classification with SVMs

- Solving the minimization problem produces vector **w**, intercept b
- Classification function is $f(x_i) = sign\left(\sum w_i x_{ij} + b\right)$
- The confidence in the prediction can be determined by the distance from the separating hyperplane



Score > *t*: yes

Score < -*t*: no

Else: don't know

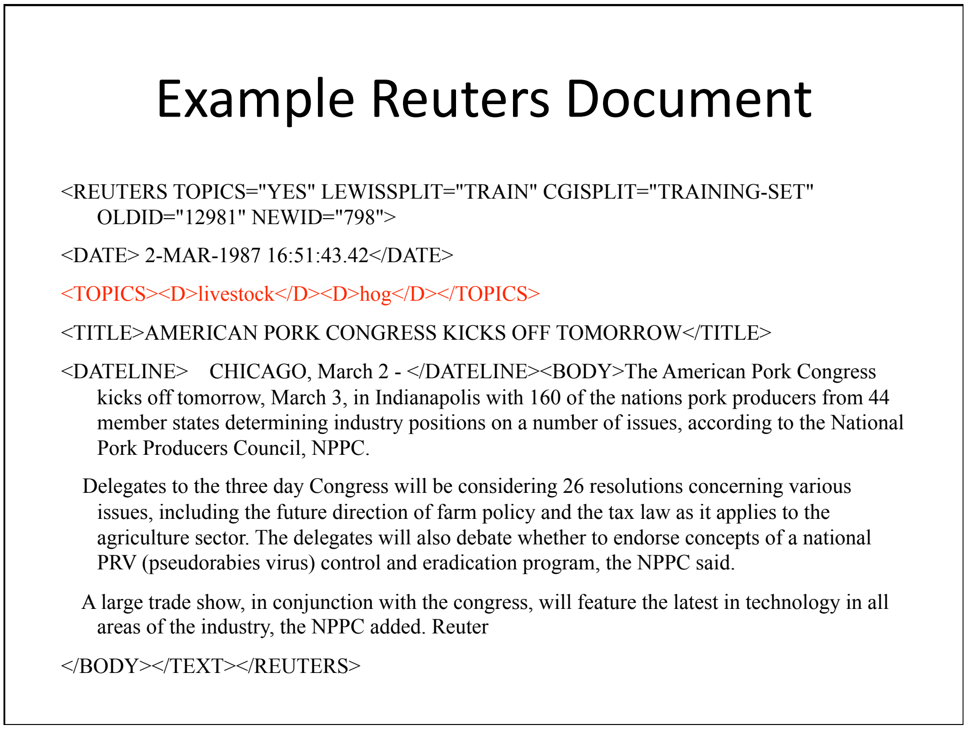
SVM Evaluation

- How do SVM and Naïve Bayes compare for a standard text classification task?
- Data: Reuters collection
  - 21,578 documents (9,603 for training, 3,299 for validation)
  - 118 categories (documents can be in more than one)
  - Only about 10 of the 118 are large

Common categories (#train, #test)

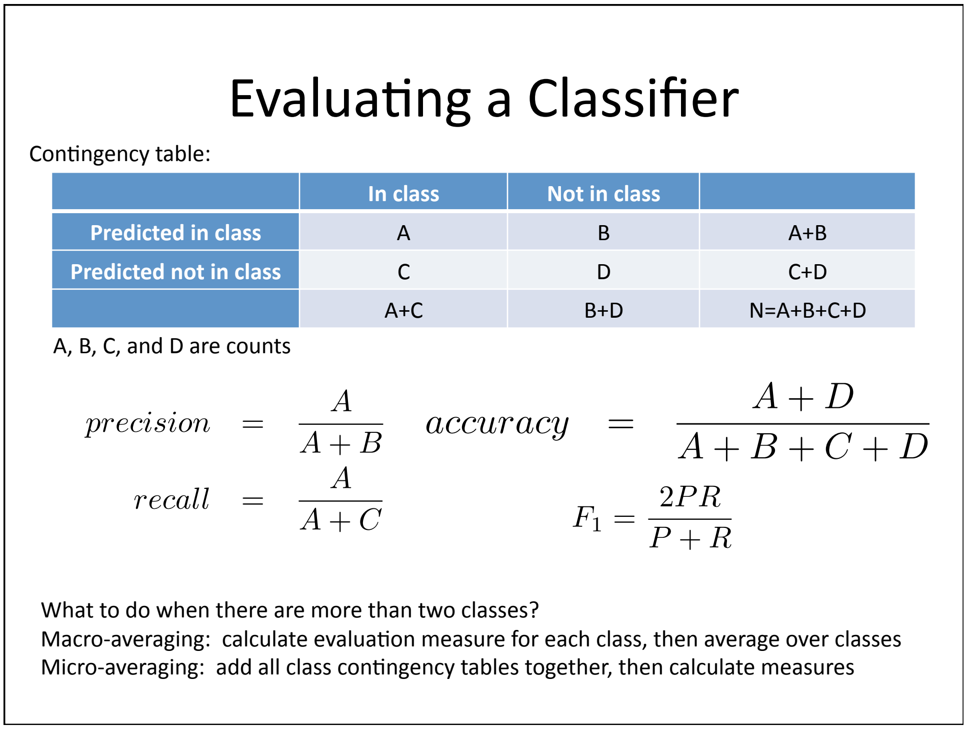
- Earn (2877, 1087)
- Acquisitions (1650, 179)
- Money-fx (538, 179)
- Grain (433, 149)
- Crude (389, 189)

- Trade (369,119)
- Interest (347, 131)
- Ship (197, 89)
- Wheat (212, 71)
- Corn (182, 56)



Example Reuters Document

```
<REUTERS TOPICS="YES" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET" OLDID="12981" NEWID="798">

<DATE> 2-MAR-1987 16:51:43.42</DATE>

<TOPICS><D>livestock</D><D>hog</D></TOPICS>

<TITLE>AMERICAN PORK CONGRESS KICKS OFF TOMORROW</TITLE>

<DATELINE>    CHICAGO, March 2 - </DATELINE><BODY>The American Pork Congress kicks off tomorrow, March 3, in Indianapolis with 160 of the nations pork producers from 44 member states determining industry positions on a number of issues, according to the National Pork Producers Council, NPPC.

  Delegates to the three day Congress will be considering 26 resolutions concerning various issues, including the future direction of farm policy and the tax law as it applies to the agriculture sector. The delegates will also debate whether to endorse concepts of a national PRV (pseudorabies virus) control and eradication program, the NPPC said.

  A large trade show, in conjunction with the congress, will feature the latest in technology in all areas of the industry, the NPPC added. Reuter

</BODY></TEXT></REUTERS>
```



Evaluating a Classifier

Contingency table:

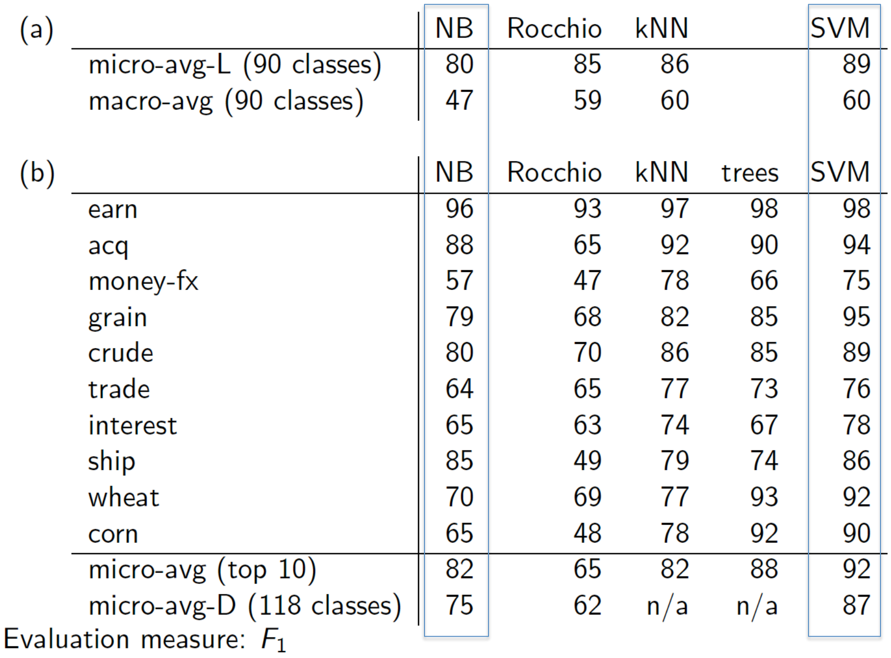
	In class	Not in class	
Predicted in class	A	B	A+B
Predicted not in class	C	D	C+D
	A+C	B+D	N=A+B+C+D

A, B, C, and D are counts

$$precision = \frac{A}{A+B} \qquad accuracy = \frac{A+D}{A+B+C+D}$$

$$recall = \frac{A}{A+C} \qquad F_1 = \frac{2PR}{P+R}$$

What to do when there are more than two classes?
Macro-averaging: calculate evaluation measure for each class, then average over classes
Micro-averaging: add all class contingency tables together, then calculate measures



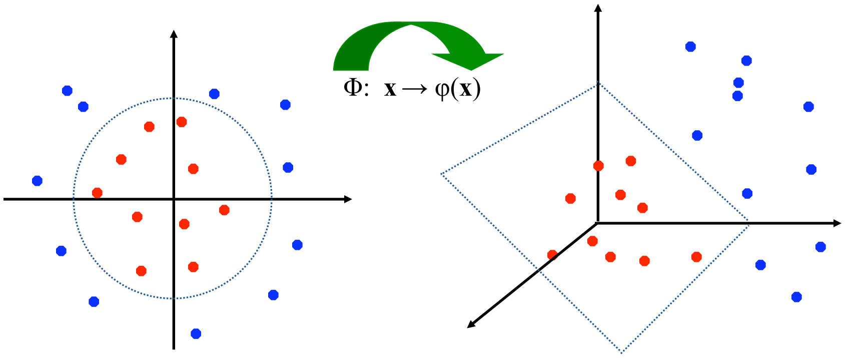
(a)	NB	Rocchio	kNN		SVM
micro-avg-L (90 classes)	80	85	86		89
macro-avg (90 classes)	47	59	60		60

(b)	NB	Rocchio	kNN	trees	SVM
earn	96	93	97	98	98
acq	88	65	92	90	94
money-fx	57	47	78	66	75
grain	79	68	82	85	95
crude	80	70	86	85	89
trade	64	65	77	73	76
interest	65	63	74	67	78
ship	85	49	79	74	86
wheat	70	69	77	93	92
corn	65	48	78	92	90
micro-avg (top 10)	82	65	82	88	92
micro-avg-D (118 classes)	75	62	n/a	n/a	87

Evaluation measure: F_1

Non-Linear SVMs

- We have only discussed the linear case, but SVMs are not restricted to that case
- If the data is not linearly separable, map it into a higher-dimensional space where it might become linearly separable



Feature Selection

- Recall that Naïve Bayes classification required some feature selection to perform well
- Discriminative models are generally more robust to "noninformative" features
  - They will simply end up with near-zero weights
- In practice, feature selection is still useful to reduce the number of weights that must be estimated