Evaluation – next steps

Lift and Costs

Outline

- Lift and Gains charts
- *ROC
- Cost-sensitive learning
- Evaluation for numeric predictions
- MDL principle and Occam's razor

Direct Marketing Paradigm

- Find most likely prospects to contact
- Not everybody needs to be contacted
- Number of targets is usually much smaller than number of prospects

- Typical Applications
  - retailers, catalogues, direct mail (and e-mail)
  - customer acquisition, cross-sell, attrition prediction
  - …

Direct Marketing Evaluation

- **Accuracy on the entire dataset is not the right measure**
- Approach
  - develop a target model
  - score all prospects and rank them by decreasing score
  - select top P% of prospects for action
- How to decide what is the best selection?

Model-Sorted List

Use a model to assign score to each customer
Sort customers by decreasing score
Expect more targets (hits) near the top of the list

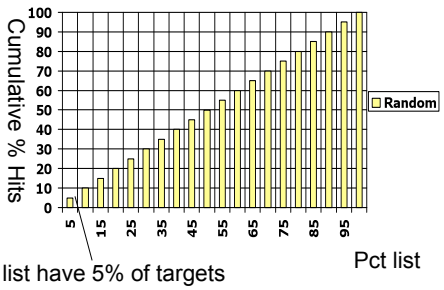
No	Score	Target	CustID	Age	
1	0.97	Y	1746	…	
2	0.95	N	1024	…	
3	0.94	Y	2478	…	
4	0.93	Y	3820	…	
5	0.92	N	4897	…	
…	…		…	…	
99	0.11	N	2734	…	
100	0.06	N	2422		

3 hits in top 5% of the list

If there 15 targets overall, then top 5 has 3/15=20% of targets

CPH (Cumulative Pct Hits)

Definition:
CPH(P,M)
= % of all targets in the first P% of the list scored by model M
CPH frequently called Gains

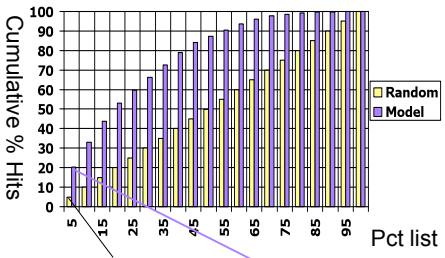


5% of random list have 5% of targets

Q: What is expected value for CPH(P,Random) ?

A: Expected value for CPH(P,Random) = P

CPH: Random List vs Model-ranked list



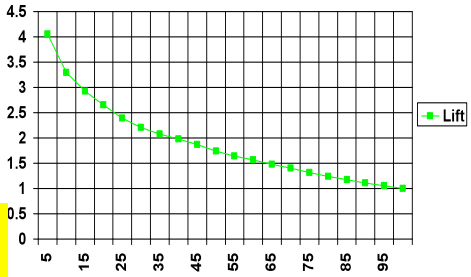
5% of random list have 5% of targets,

but 5% of model ranked list have 21% of targets CPH(5%,model)=21%.

Lift

$$\mathbf{Lift}(P,M) = \mathbf{CPH}(P,M) \ / \ P$$

Lift (at 5%)
= 21% / 5%
= 4.2
better
than random



P -- percent of the list

Note: Some (including Witten & Eibe) use "Lift" for what we call CPH.

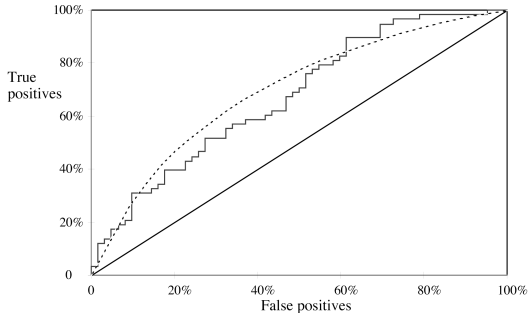
Lift Properties

- ***Q: Lift(P,Random) =***
  - ***A:*** 1 (expected value, can vary)
- ***Q: Lift(100%, M) =***
  - ***A:*** 1 (for any model M)
- ***Q: Can lift be less than 1?***
  - ***A***: yes, if the model is inverted (all the non-targets precede targets in the list)
- Generally, a better model has higher lift

*ROC curves

- *ROC curves* are similar to gains charts
  - Stands for "receiver operating characteristic"
  - Used in signal detection to show tradeoff between hit rate and false alarm rate over noisy channel
- Differences from gains chart:
  - *y* axis shows percentage of true positives in sample *rather than absolute number*
  - *x* axis shows percentage of false positives in sample *rather than sample size*

*A sample ROC curve

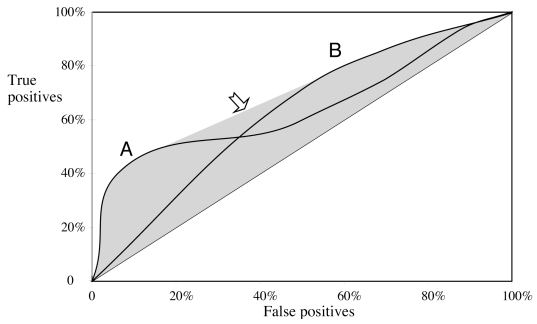


- Jagged curve—one set of test data
- Smooth curve—use cross-validation

*Cross-validation and ROC curves

- Simple method of getting a ROC curve using cross-validation:
  - Collect probabilities for instances in test folds
  - Sort instances according to probabilities
- This method is implemented in WEKA
- However, this is just one possibility
  - The method described in the book generates an ROC curve for each fold and averages them

*ROC curves for two schemes



- For a small, focused sample, use method A
- For a larger one, use method B
- In between, choose between A and B with appropriate probabilities

*The convex hull

- Given two learning schemes we can achieve any point on the convex hull!
- TP and FP rates for scheme 1: t_1 and f_1
- TP and FP rates for scheme 2: t_2 and f_2
- If scheme 1 is used to predict $100 \times q$ % of the cases and scheme 2 for the rest, then
  - TP rate for combined scheme: $q \times t_1+(1-q) \times t_2$
  - FP rate for combined scheme: $q \times f_2+(1-q) \times f_2$

Cost Sensitive Learning

- There are two types of errors

		Predicted class	
		Yes	No
Actual class	Yes	TP: True positive	FN: False negative
	No	FP: False positive	TN: True negative

- Machine Learning methods usually minimize FP+FN
- Direct marketing maximizes TP

Different Costs

- In practice, true positive and false negative errors often incur different costs
- Examples:
  - Medical diagnostic tests: does X have leukemia?
  - Loan decisions: approve mortgage for X?
  - Web mining: will X click on this link?
  - Promotional mailing: will X buy the product?
  - …

Cost-sensitive learning

- Most learning schemes do not perform cost-sensitive learning
  - They generate the same classifier no matter what costs are assigned to the different classes
  - Example: standard decision tree learner
- Simple methods for cost-sensitive learning:
  - Re-sampling of instances according to costs
  - Weighting of instances according to costs
- Some schemes are inherently cost-sensitive, e.g. naïve Bayes

*Measures in information retrieval

- Percentage of retrieved documents that are relevant: *precision*=TP/(TP+FP)
- Percentage of relevant documents that are returned: *recall* =TP/(TP+FN)
- Precision/recall curves have hyperbolic shape
- Summary measures: average precision at 20%, 50% and 80% recall (*three-point average recall*)
- *F-measure*=(2×recall×precision)/(recall+precision)

*Summary of measures

	Domain	Plot	Explanation
Lift chart	**Marketing**	**TP** **Subset size**	**TP** **(TP+FP)/(TP+FP+TN+FN)**
ROC curve	**Communications**	**TP rate** **FP rate**	**TP/(TP+FN)** **FP/(FP+TN)**
Recall-precision curve	**Information retrieval**	**Recall** **Precision**	**TP/(TP+FN)** **TP/(TP+FP)**

Evaluating numeric prediction

- Same strategies: independent test set, cross-validation, significance tests, etc.
- Difference: error measures
- Actual target values: $a_1\ a_2 \ldots a_n$
- Predicted target values: $p_1\ p_2 \ldots\ p_n$
- Most popular measure: *mean-squared error*

$$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}$$

  - Easy to manipulate mathematically

Other measures

- The *root mean-squared error* :

$$\sqrt{\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{n}}$$

- The *mean absolute error* is less sensitive to outliers than the mean-squared error:

$$\frac{|p_1-a_1|+\ldots+|p_n-a_n|}{n}$$

- Sometimes *relative* error values are more appropriate (e.g. 10% for an error of 50 when predicting 500)

Improvement on the mean

- How much does the scheme improve on simply predicting the average?
- The *relative squared error* is ($\bar{a}$ is the average):

$$\frac{(p_1-a_1)^2+\ldots+(p_n-a_n)^2}{(\bar{a}-a_1)^2+\ldots+(\bar{a}-a_n)^2}$$

- The *relative absolute error* is:

$$\frac{|p_1-a_1|+\ldots+|p_n-a_n|}{|\bar{a}-a_1|+\ldots+|\bar{a}-a_n|}$$

Correlation coefficient

- Measures the *statistical correlation* between the predicted values and the actual values

$$\frac{S_{PA}}{\sqrt{S_P S_A}}$$

$$S_{PA}=\frac{\sum_i (p_i-\bar{p})(a_i-\bar{a})}{n-1} \qquad S_P=\frac{\sum_i (p_i-\bar{p})^2}{n-1} \qquad S_A=\frac{\sum_i (a_i-\bar{a})^2}{n-1}$$

- Scale independent, between –1 and +1
- Good performance leads to large values!

Which measure?

- Best to look at all of them
- Often it doesn't matter
- Example:

	A	B	C	D
Root mean-squared error	67.8	91.7	63.3	57.4
Mean absolute error	41.3	38.5	33.4	29.2
Root rel squared error	42.2%	57.2%	39.4%	35.8%
Relative absolute error	43.1%	40.1%	34.8%	30.4%
Correlation coefficient	0.88	0.88	0.89	0.91

  - D best
  - C second-best
  - A, B arguable

*The MDL principle

- MDL stands for *minimum description length*
- The description length is defined as:

  space required to describe a theory

  +

  space required to describe the theory's mistakes
- In our case the theory is the classifier and the mistakes are the errors on the training data
- Aim: we seek a classifier with minimal DL
- MDL principle is a *model selection criterion*

Model selection criteria

- Model selection criteria attempt to find a good compromise between:
  A. The complexity of a model
  B. Its prediction accuracy on the training data
- Reasoning: a good model is a simple model that achieves high accuracy on the given data
- Also known as *Occam's Razor* : the best theory is the smallest one that describes all the facts

William of Ockham, born in the village of Ockham in Surrey (England) about 1285, was the most influential philosopher of the 14th century and a controversial theologian.

Elegance vs. errors

- Theory 1: very simple, elegant theory that explains the data almost perfectly
- Theory 2: significantly more complex theory that reproduces the data without mistakes
- Theory 1 is probably preferable
- Classical example: Kepler's three laws on planetary motion
  - Less accurate than Copernicus's latest refinement of the Ptolemaic theory of epicycles

*MDL and compression

- MDL principle relates to data compression:
  - The best theory is the one that compresses the data the most
  - I.e. to compress a dataset we generate a model and then store the model and its mistakes
- We need to compute
  (a) size of the model, and
  (b) space needed to encode the errors
- (b) easy: use the informational loss function
- (a) need a method to encode the model

*MDL and Bayes's theorem

- L[T]="length" of the theory
- L[E|T]=training set encoded wrt the theory
- Description length= L[T] + L[E|T]
- Bayes' theorem gives *a posteriori* probability of a theory given the data:

$$\Pr[T \mid E] = \frac{\Pr[E \mid T]\Pr[T]}{\Pr[E]}$$

- Equivalent to:

$$-\log \Pr[T \mid E] = -\log \Pr[E \mid T] - \log \Pr[T] + \underbrace{\log \Pr[E]}_{\textit{constant}}$$

*MDL and MAP

- MAP stands for *maximum a posteriori probability*
- Finding the MAP theory corresponds to finding the MDL theory
- Difficult bit in applying the MAP principle: determining the prior probability Pr[T] of the theory
- Corresponds to difficult part in applying the MDL principle: coding scheme for the theory
- I.e. if we know a priori that a particular theory is more likely we need less bits to encode it

*Discussion of MDL principle

- Advantage: makes full use of the training data when selecting a model
- Disadvantage 1: appropriate coding scheme/prior probabilities for theories are crucial
- Disadvantage 2: no guarantee that the MDL theory is the one which minimizes the expected error
- Note: Occam's Razor is an axiom!
- Epicurus' *principle of multiple explanations*: keep all theories that are consistent with the data

*Bayesian model averaging

- Reflects Epicurus' principle: all theories are used for prediction weighted according to P[T|E]
- Let I be a new instance whose class we must predict
- Let C be the random variable denoting the class
- Then BMA gives the probability of C given
  - I
  - training data E
  - possible theories T_j

$$\Pr[C \mid I, E] = \sum_j \Pr[C \mid I, T_j] \Pr[T_j \mid E]$$

*MDL and clustering

- Description length of theory: bits needed to encode the clusters
  - e.g. cluster centers
- Description length of data given theory: encode cluster membership and position relative to cluster
  - e.g. distance to cluster center
- Works if coding scheme uses less code space for small numbers than for large ones
- With nominal attributes, must communicate probability distributions for each cluster

Evaluation Summary:

- Avoid Overfitting
- Use Cross-validation for small data
- Don't use test data for parameter tuning - use separate validation data
- Consider costs when appropriate