Machine learning

TM the Great Unread

DTL|Digital Arts Initiative
Interacting Minds Centre|Aarhus University

July 29, 2016

Classification

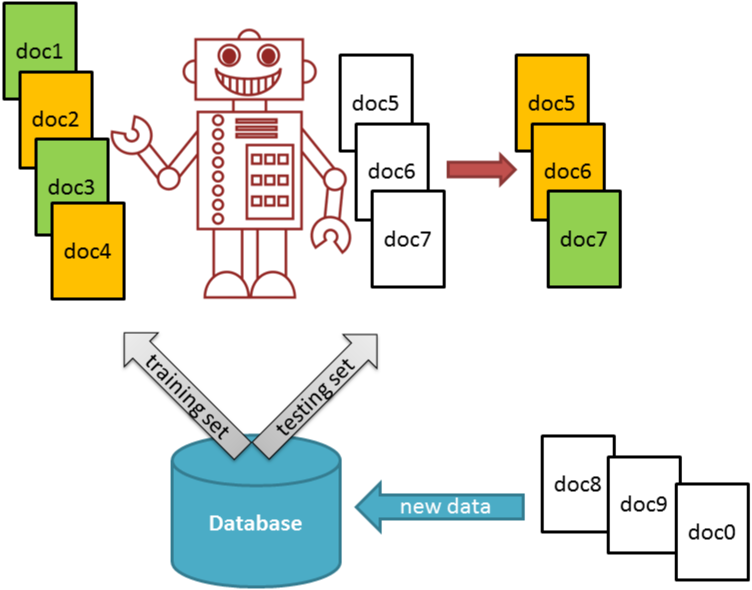
Given labeled data (supervised learning), a classification algorithm will output a solution that categorizes new examples $\rightarrow$ associate labels with subsets of the data

While clustering (unsupervised learning) searches for groups within the corpus, classification learns to map a collection of documents onto a categorical class values or labels $\rightarrow$ find mapping function

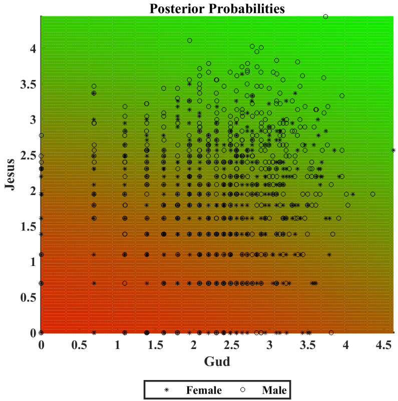
Data (features) with class values ($\sim$ labeled data), excellent opportunity to make use of metadata

Vast majority of models are **black box models**

Workflow: separate data set in training and test subsets (training, test, and validation) $\rightarrow$ train model $\rightarrow$ test model $\rightarrow$ apply model to new data


doc1
doc2
doc3
doc4
doc5
doc6
doc7
doc5
doc6
doc7
training set
testing set
Database
new data
doc8
doc9
doc0

classification in the humanities



Types of classifiers

Binary and multiclass classification problems [1]

Naive Bayes probabilistic classifier that is fast and popular for in text categorization, but assumes independence between features (naive)

Neural network broad framework for machine learning, which is very extremely flexible. Training can be very slow, but classification fast. Prone to overfitting

Decision Tree versatile and creates sets of rules (binary decisions) that are simple and can be understood (leaves are classes and branches features) $\rightarrow$ white box method

Support Vector Machines works on small datasets (typically binary) with high dimensional data (features $>$ objects) and very memory efficient (only uses the support vectors). Bad performance on noisy data (overlapping classes)

[1] Can be advantageous to reformulate multiclass problems as binary

Training

Labeled data the correct class information is available

- metadata is readily available, e.g. author, genre, year of publication
- labels from an external source/databases, e.g. reviews, ratings, reads
- annotate data (expert or raters)

Evaluation

Estimate performance (error rate) of a classifier (the lower the error the better). Often several classifiers are compared

Most metrics are developed for binary classification problems

Confusion matrix: table for describing performance of classifier on training and/or testing data

		True	
		positive	negative
Predicted	positive	TP	FP
	negative	FN	TN

True Positive: Correctly assigns positive class membership
True Negative: Correctly rejects class membership
False Positive: Fail to rejects class membership
False Negative: Reject class membership incorrectly

We train a Naive Bayes classifier on 1500 verses of the KJV Bible labeled with collection data (NT: New Testament OT: Old Testament)

Confusion matrix for binary classification problem:

	NT	OT
NT	644	89
OT	106	661

, verses: $644 + 106 + 89 + 661 = 1500$

Accuracy

Measures in how many cases the predicted class conformed with the correct class: $\dfrac{TP + NP}{TP + TN + FP + FN}$

	NT	OT
NT	644	89
OT	106	661

, accuracy: $\dfrac{(644 + 661)}{1500} = 0.87$ (87%)

Precision

Measures the number of selected verses that are relevant, i.e., how certain are we that a classified verse is correctly classified ($\sim$ how many time did the model positively predict a class): $\dfrac{TP}{TP + FP}$

	NT	OT
NT	644	89
OT	106	661

, $precision_{NT}$: $\dfrac{(644)}{644 + 89} = 0.88$

For each class label: How many of the items that got the label should have gotten it? How many should have gotten other labels?

Recall

Recall measures the number of relevant verses that are selected, i.e., how good is the classifier at detecting verses within a given class: $\dfrac{TP}{TP + FN}$

	NT	OT
NT	644	89
OT	106	661

, $recall_{NT}$: $\dfrac{(644)}{644 + 106} = 0.86$

For each class label: How many items that should have gotten the label did get it? How many were missed?

F-score

Composite (general) measure of a classifier's accuracy

$$F_1 = 2 \times \frac{percision \times recall}{precision + recall}$$

$$F_1 : 2 \times \frac{.88 \times .86}{.88 + .86} = 0.87$$

F is the harmonic mean of precision and recall.

Validation

If the model gets enough data, it can basically memorize the data set (overfitting) $\rightarrow$ need to test the model on held-out data

When building a predictive model, we need a way to evaluate the capability of the model on unseen data:

- Data Split (conventional validation)
- Cross validation
- Bootstrap