Text Classification

COMP90042

Natural Language Processing

Lecture 4

Semester 1 Week 2

Jey Han Lau

THE UNIVERSITY OF MELBOURNE

Outline

- Fundamentals of classification
- Text classification tasks
- Algorithms for classification
- Evaluation

Classification

- Input
  - A document *d*
    - Often represented as a vector of *features*
  - A fixed output set of classes $C = \{c_1, c_2, \ldots \mathrm{c}_\mathrm{k}\}$
    - Categorical, not continuous (regression) or ordinal (ranking)
- Output
  - A predicted class $c \in C$

Text Classification Tasks

- Some common examples
  - Topic classification
  - Sentiment analysis
  - Native-language identification
  - Natural language inference
  - Automatic fact-checking
  - Paraphrase
- Input may not be a long document
  - E.g. sentence or tweet-level sentiment analysis

Topic Classification

Is the text about **acquisitions** or **earnings**?

LIEBERT CORP APPROVES MERGER
Liebert Corp said its shareholders approved the merger of a wholly-owned subsidiary of Emerson Electric Co. Under the terms of the merger, each Liebert shareholder will receive .3322 shares of Emerson stock for each Liebert share.

ANSWER: ACQUISITIONS

Topic Classification

- Motivation: library science, information retrieval
- Classes: Topic categories, e.g. “jobs”, “international news”
- Features
  - Unigram bag of words (BOW), with stop-words removed
  - Longer *n*-grams (bigrams, trigrams) for phrases
- Examples of corpora
  - Reuters news corpus (RCV1; NLTK)
  - Pubmed abstracts
  - Tweets with hashtags

Sentiment Analysis

What is the sentiment of this tweet?

> anyone having problems with Windows 10? may be coincidental but since i downloaded, my WiFi keeps dropping out. Itunes had a malfunction

ANSWER: NEGATIVE

Sentiment Analysis

- Motivation: opinion mining, business analytics
- Classes: Positive/Negative/(Neutral)
- Features
  - *N*-grams
  - Polarity lexicons
- Examples of corpora
  - Movie review dataset (in NLTK)
  - SEMEVAL Twitter polarity datasets

Native-Language Identification

What is the **native language** of the writer of this text?

> Based on the feedback given, how students revised their writing will be analyzed as well. However, since whether teachers tell their student to revise or not can depend on teachers, it is unsure the following analysis can be taken.

PollEv.com/jeyhanlau569

Native-Language Identification

- Motivation: forensic linguistics, educational applications
- Classes: first language of author (e.g. Indonesian)
- Features
  - Word *N*-grams
  - Syntactic patterns (POS, parse trees)
  - Phonological features
- Examples of corpora
  - TOEFL/IELTS essay corpora

Natural Language Inference

What is the relationship between the first and second sentence (entailment vs. contradiction)?

1: A man inspects the uniform of a figure in some East Asian country.

2: The man is sleeping

ANSWER: CONTRADICTION

Natural Language Inference

- AKA textual entailment
- Motivation: language understanding
- Classes: entailment, contradiction, neutral
- Features
  - Word overlap
  - Length difference between the sentences
  - *N*-grams
- Examples of corpora
  - SNLI, MNLI

Building a Text Classifier

1. Identify a task of interest
2. Collect an appropriate corpus
3. Carry out annotation
4. Select features
5. Choose a machine learning algorithm
6. Train model and tune hyper-parameters using held-out development data
7. Repeat earlier steps as needed
8. Train final model
9. Evaluate model on held-out test data

Algorithms for Classification

Choosing a Classification Algorithm

- Bias vs. Variance
  - Bias: assumptions we made in our model
  - Variance: sensitivity to training set
- Underlying assumptions, e.g., independence
- Complexity
- Speed

Naïve Bayes

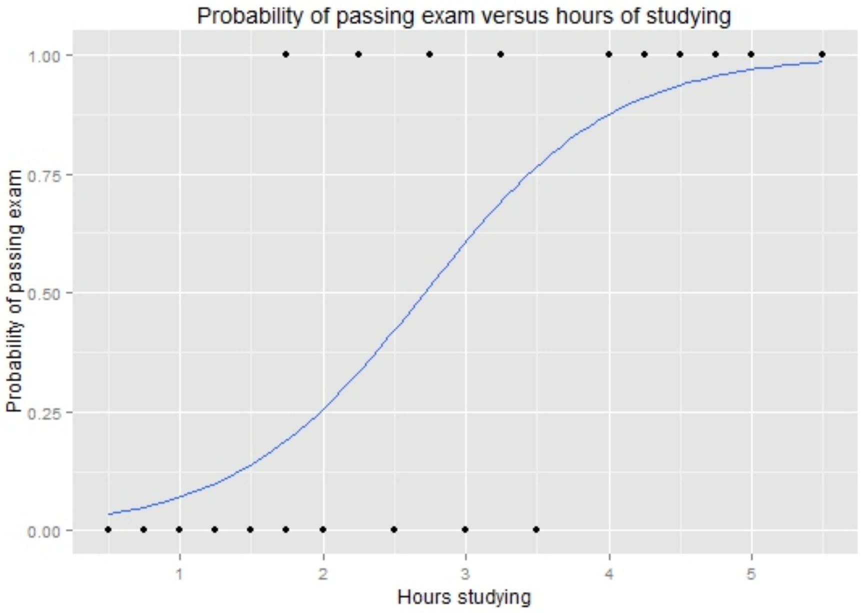
- Finds the class with the highest likelihood under Bayes law
  - $P(C|F) \propto P(F|C)P(C)$
  - i.e. probability of the class times probability of features given the class
- Naïvely assumes features are independent

$$p\left(c_n \middle| f_1 \ldots f_m\right) = \prod_{i=1}^{m} p\left(f_i \middle| c_n\right) p(c_n)$$

Naïve Bayes

- Pros:
  - Fast to train and classify
  - robust, low-variance → good for low data situations
  - optimal classifier if independence assumption is correct
  - extremely simple to implement.
- Cons:
  - Independence assumption rarely holds
  - low accuracy compared to similar methods in most situations
  - smoothing required for unseen class/feature combinations

Logistic Regression



- A classifier, despite its name
- A linear model, but uses *softmax* “squashing” to get valid probability

$$p\left(c_n \middle| f_1 \ldots f_m\right) = \frac{1}{Z} \bullet \exp\left(\sum_{i=0}^{m} w_i f_i\right)$$

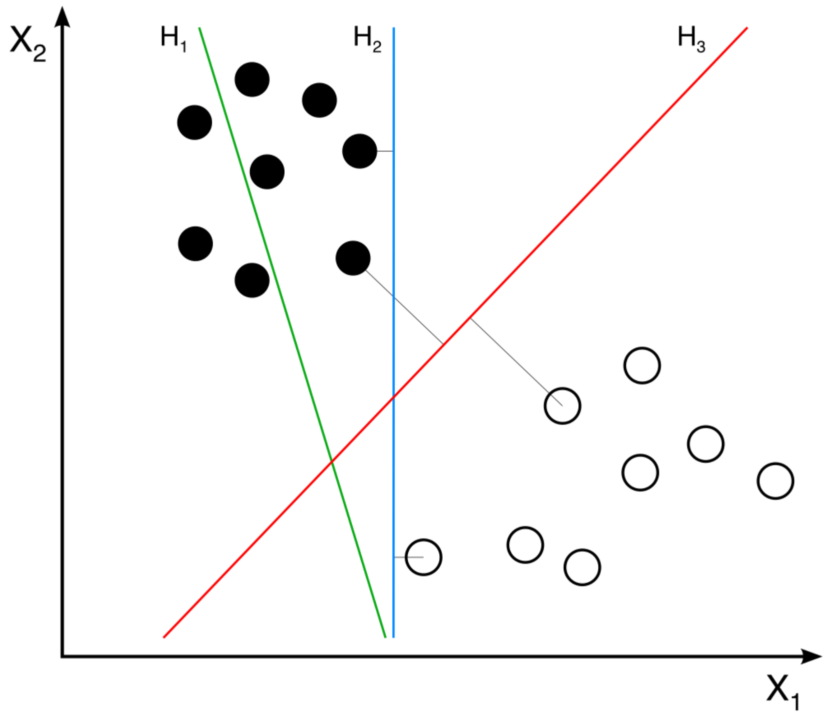
- Training maximizes probability of training data subject to regularization which encourages low or sparse weights

Logistic Regression

- Pros:
  - Unlike Naïve Bayes not confounded by diverse, correlated features → better performance
- Cons:
  - Slow to train;
  - Feature scaling needed
  - Requires a lot of data to work well in practice
  - Choosing regularisation strategy is important since overfitting is a big problem

Support Vector Machines

- Finds hyperplane which separates the training data with maximum margin
- Pros:
  - Fast and accurate linear classifier
  - Can do non-linearity with kernel trick
  - Works well with huge feature sets
- Cons:
  - Multiclass classification awkward
  - Feature scaling needed
  - Deals poorly with class imbalances
  - Interpretability



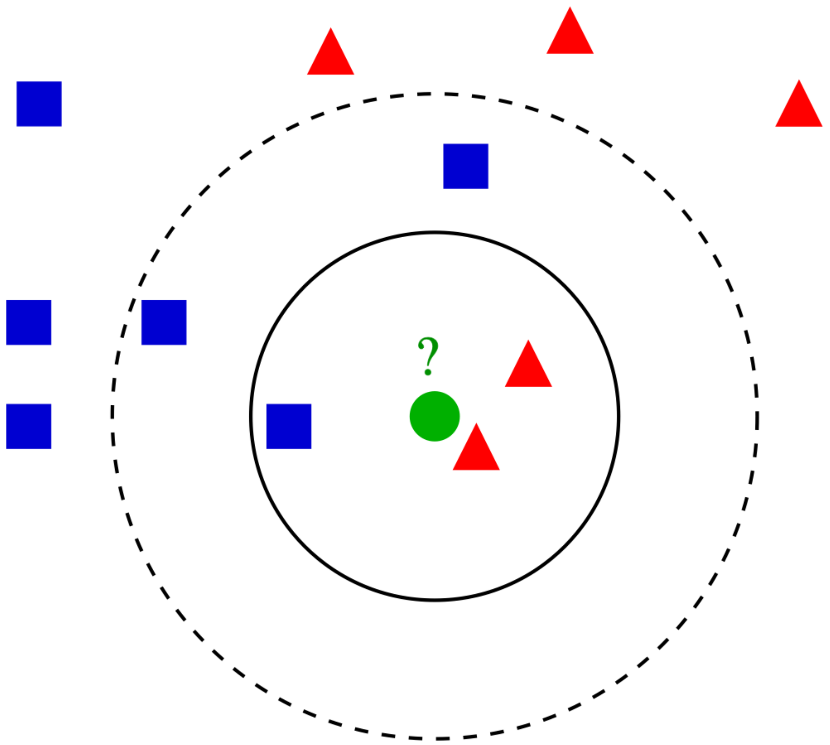
Prior to deep learning, SVM is very popular for NLP, why?

- Non-linear kernel trick works well for text
- Feature scaling is not an issue for NLP
- NLP datasets are usually large, which favours SVM
- NLP problems often involve large feature sets

PollEv.com/jeyhanlau569

K-Nearest Neighbour

- Classify based on majority class of *k*-nearest training examples in feature space
- Definition of nearest can vary
  - Euclidean distance
  - Cosine distance

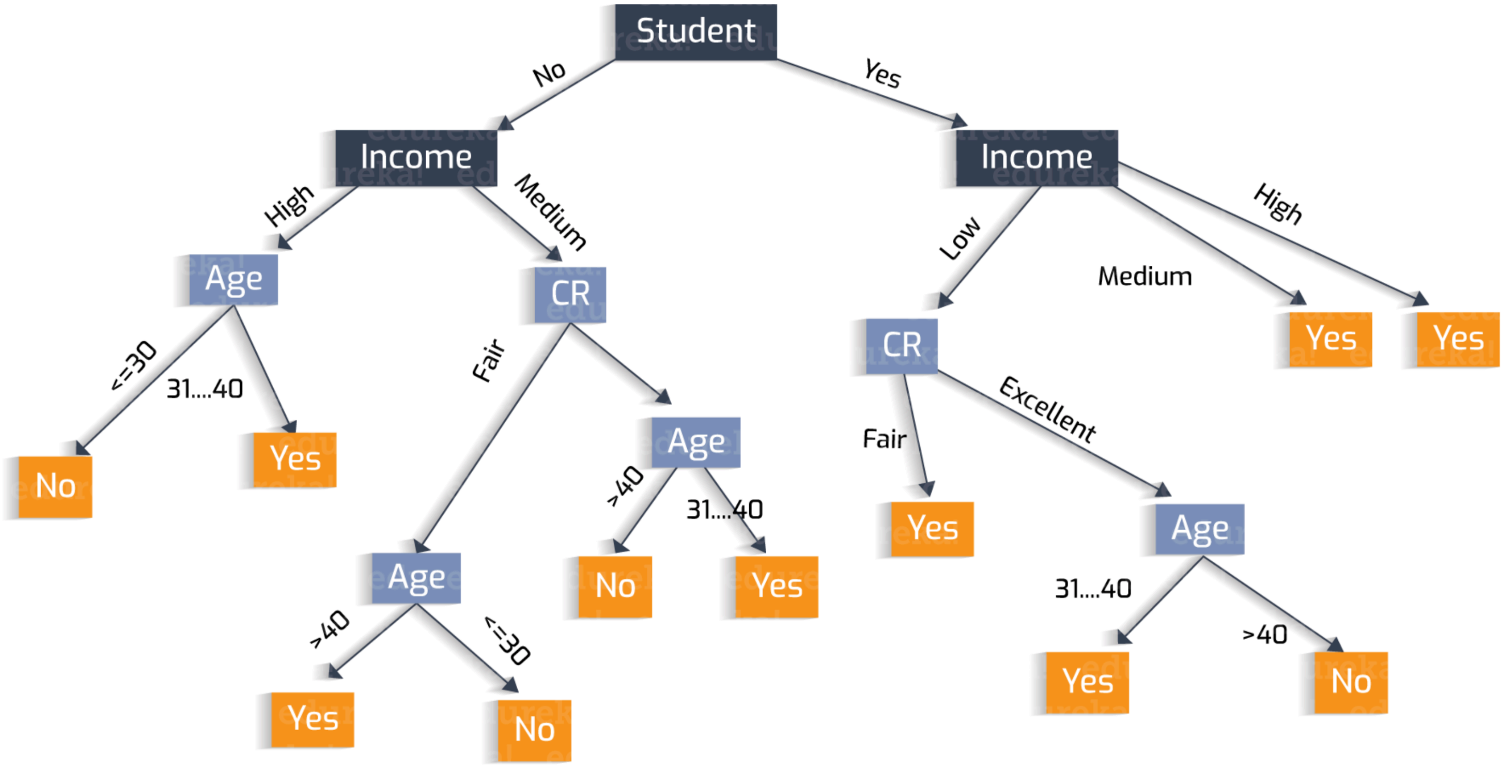


K-Nearest Neighbour

- Pros:
    - Simple but surprisingly effective
    - No training required
    - Inherently multiclass
    - Optimal classifier with infinite data
- Cons:
    - Have to select *k*
    - Issues with imbalanced classes
    - Often slow (for finding the neighbours)
    - Features must be selected carefully

Decision tree

- Construct a tree where nodes correspond to tests on individual features
- Leaves are final class decisions
- Based on greedy maximization of mutual information

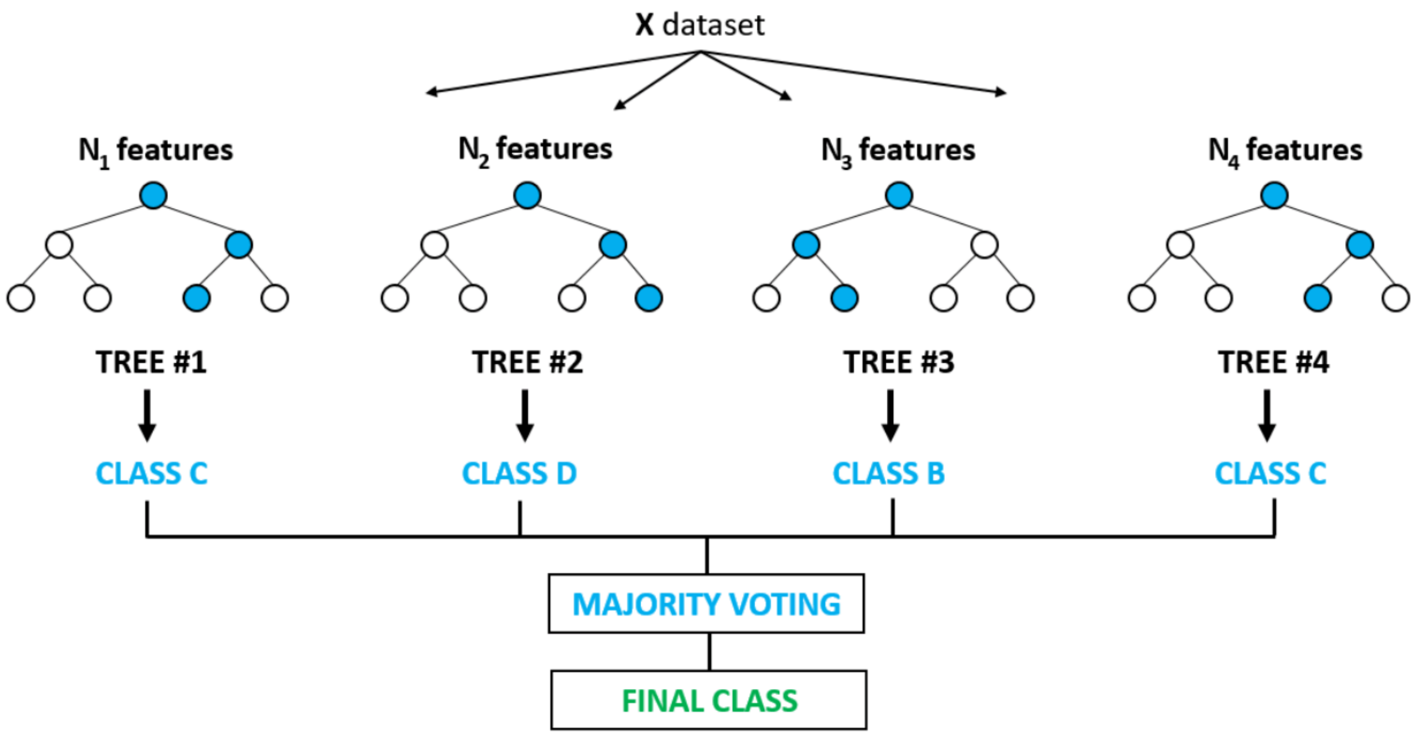


Decision tree

- Pros:
    - Fast to build and test
    - Feature scaling irrelevant
    - Good for small feature sets
    - Handles non-linearly-separable problems
- Cons:
    - In practice, not that interpretable
    - Highly redundant sub-trees
    - Not competitive for large feature sets

Random Forests

- An *ensemble* classifier
- Consists of decision trees trained on different subsets of the training and feature space
- Final class decision is majority vote of sub-classifiers

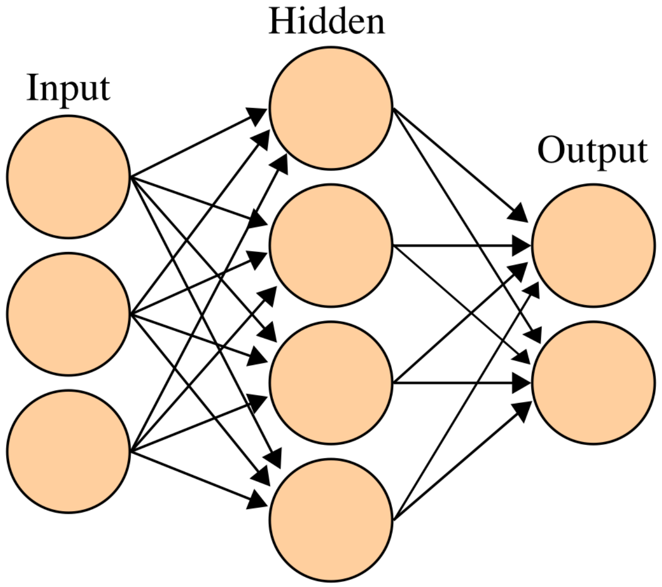


Random Forests

- Pros:
  - Usually more accurate and more robust than decision trees
  - Great classifier for medium feature sets
  - Training easily parallelised
- Cons:
  - Interpretability
  - Slow with large feature sets

Neural Networks

- An interconnected set of nodes typically arranged in layers
- Input layer (features), output layer (class probabilities), and one or more hidden layers
- Each node performs a linear weighting of its inputs from previous layer, passes result through activation function to nodes in next layer



Neural Networks

- Pros:
  - Extremely powerful, dominant method in NLP and vision
  - Little feature engineering
- Cons:
  - Not an off-the-shelf classifier
  - Many hyper-parameters, difficult to optimise
  - Slow to train
  - Prone to overfitting

Hyper-parameter Tuning

- Dataset for tuning
    - Development set
    - Not the training set or the test set
    - *k*-fold cross-validation
- Specific hyper-parameters are classifier specific
    - E.g. tree depth for decision trees
- But many hyper-parameters relate to regularisation
    - Regularisation hyper-parameters penalise model complexity
    - Used to prevent overfitting
- For multiple hyper-parameters, use grid search

Evaluation

Evaluation: Accuracy

	Classified As	
Class	A	B
A	79	13
B	8	10

Accuracy = correct classifications/total classifications

= (79 + 10)/(79 + 13 + 8 + 10)

= 0.81

0.81 looks good, but most common class baseline accuracy is

= (79 + 13)/(79 + 13 + 8 + 10) = 0.84

Evaluation: Precision & Recall

	Classified As	
Class	A	B
A	79	13
B	8	10

False Positives (fp)

True Positives (tp)

False Negatives (fn)

B as "positive class"

Precision = correct classifications of B (tp)
/ total classifications as B (tp + fp)
= 10/(10 + 13) = 0.43

Recall = correct classifications of B (tp)
/ total instances of B (tp + fn)
= 10/(10 + 8) = 0.56

Evaluation: F(1)-score

- Harmonic mean of precision and recall

$$F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

- Like precision and recall, defined relative to a specific positive class
- But can be used as a general multiclass metric
  - Macroaverage: Average F-score across classes
  - Microaverage: Calculate F-score using sum of counts (= accuracy for multiclass problems)

A Final Word

- Lots of algorithms available to try out on your task of interest (see scikit-learn)
- But if good results on a new task are your goal, then well-annotated, plentiful datasets and appropriate features often more important than the specific algorithm used

Further Reading

- E18 Ch 4.1, 4.3-4.4.1
- E18 Chs 2 & 3: reviews linear and non-linear classification algorithms