7: Catchup Session & very short intro to other classifiers

Machine Learning and Real-world Data (MLRD)

Simone Teufel (based on slides by Paula Buttery)
What happens in a catchup session?

- Lecture and practical session as normal.
- New material is non-examinable.
- Time for you to catch-up or attempt some starred ticks.
- Demonstrators help as per usual.
Naive Bayes is a probabilistic classifier

- Given a set of input features a probabilistic classifier provide a distribution over classes.
- That is, for a set of observed features $O$ and classes $c_1...c_n \in C$ gives $P(c_i|O)$ for all $c_i \in C$
- For us $O$ was the set all the words in a review $\{w_1, w_2, ..., w_n\}$ where $w_i$ is the $i$th word in a review, $C = \{\text{POS}, \text{NEG}\}$
- We decided on a single class by choosing the one with the highest probability given the features:

$$\hat{c} = \underset{c \in C}{\text{argmax}} \ P(c|O)$$
An SVM is a popular non-probabilistic classifier

- A Support Vector Machine (SVM) is a non-probabilistic binary linear classifier
- SVMs assign new examples to one category or the other
- SVMs can reduce the amount of labeled data required to gain good accuracy
- SVMs can be efficiently adapted to perform a non-linear classification
SVMs find hyper-planes that separate classes

- Our classes exist in a multidimensional feature space
- A linear classifier will separate the points with a hyper-plane
SVMs find a maximum-margin hyperplane in noisy data

There are many possible hyperplanes
SVMs find the best hyperplane such that the distance from it to the nearest data point from each class is maximised
i.e. the hyperplane that passes through the widest possible gap (hopefully helps to avoid over-fitting)
SVMs can be very efficient and effective

- Efficient when learning from a large number of features (good for text)
- Effective even with relatively small amounts of labelled data (we only need points close to the plane to calculate it)
- We can choose how many points to involve (size of margin) when calculating the plane (tuning vs. over-fitting)
- Can separate non-linear boundaries by changing the feature space (using a kernel function)
Choice of classifier will depend on the task

Comparison of a SVM and Naive Bayes on the same task:
- 2000 imdb movie reviews, 400 kept for testing
- preprocess with improved tokeniser (lowercased, removed uninformative words, dealt with punctuation, lemmatised words)

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>Naive Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy on train</td>
<td>0.98</td>
<td>0.96</td>
</tr>
<tr>
<td>Accuracy on test</td>
<td>0.84</td>
<td>0.80</td>
</tr>
</tbody>
</table>

- But from Naive Bayes I know that character, good, story, great, ... are informative features
- SVMs are more difficult to interpret
Decision tree can be used to visually represent classifications

- Simple to interpret
- Can mix numerical and categorical data
- You specify the parameters of the tree (maximum depth, number of items at leaf nodes—both change accuracy)
- But finding the optimal decision tree can be NP-complete
Information gain can be used to decide how to split

- Information gain is defined in terms of entropy $H$

**Entropy of tree node:**

$$H(n) = - \sum_{p} p_i \log_2 p_i$$

where $p_i$ are the probabilities of each class at node $n$

- Information gain $I$ is the reduction in entropy of $n$ achieved by learning the state of the random variable $D$.

**Information gain:**

$$I(n, D) = H(n) - H(n|D)$$

where $H(n|D)$ is the weighted entropy of the daughter nodes if we split on $D$. 
Information gain can be used to decide how to split

- bad <= 0.0157
  - entropy = 0.9999
  - samples = 1600
  - value = [809, 791]
  - class = neg
- waste <= 0.022
  - entropy = 0.952
  - samples = 1014
  - value = [377, 637]
  - class = pos
- many <= 0.0094
  - entropy = 0.9238
  - samples = 944
  - value = [320, 624]
  - class = pos
- strong <= 0.0257
  - entropy = 0.6924
  - samples = 70
  - value = [57, 13]
  - class = neg
- suppose <= 0.0144
  - entropy = 0.907
  - samples = 428
  - value = [290, 138]
  - class = neg
- perfect <= 0.0343
  - entropy = 0.9432
  - samples = 366
  - value = [234, 132]
  - class = neg
- life <= 0.0194
  - entropy = 0.8309
  - samples = 586
  - value = [432, 154]
  - class = pos
- show <= 0.0287
  - entropy = 0.4237
  - samples = 58
  - value = [53, 5]
  - class = neg
- move <= 0.0127
  - entropy = 0.9183
  - samples = 12
  - value = [4, 8]
  - class = pos
- together <= 0.0284
  - entropy = 0.339
  - samples = 143
  - value = [134, 9]
  - class = neg
- reason <= 0.0084
  - entropy = 0.9968
  - samples = 15
  - value = [8, 7]
  - class = neg

(...) (...) (...) (...) (...)
Results on the 2000 movie reviews:

<table>
<thead>
<tr>
<th></th>
<th>SVM</th>
<th>Naive Bayes</th>
<th>DTREE (max depth 7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy on train</td>
<td>0.98</td>
<td>0.96</td>
<td>0.80</td>
</tr>
<tr>
<td>Accuracy on test</td>
<td>0.84</td>
<td>0.80</td>
<td>0.69</td>
</tr>
</tbody>
</table>
Today

■ Pretester for Tick 1 cuts out Midnight on Friday
■ Friday session last chance to get help
■ Get in touch with me if you are behind
■ New topic starts on Monday—try to have ticks 1–6 by end of today