Recap Exercise

In what follows, we will be considering the following corpus of sentences:

D1 The use of valid arguments developed with reasoning and evidence.

D2 Valid arguments are persuasive by reinforcing reason to an argument.

D3 Reasoning consists in deriving truths from truths.

1 Information Retrieval (10 mins)

Compute a term-by-document TFIDF matrix, and then rank D1-D3 by cosine similarity w.r.t. the following two queries

1. argument
2. reason

Hint: Use *stemming* and case normalization!

2 k-NN Classification (10 mins)

document	class
D1	reason
D2	argument
D3	reason

Use k-NN and *cosine similarity* to classify the following two documents:

D4 Arguments can be won by reason and reason alone.

D5 Arguments cannot be useless arguments.

To what value should we set (hyper) parameter k? Justify.

3 Evaluation (5 mins)

Suppose our gold standard looks like this:

document	class
D1	reason
D2	argument
D3	reason
D4	argument
D5	reason

What would be the precision, recall and F1 scores of your k-NN algorithm?

4 Decision Trees (10 mins)

document	length	repetitive	class
D1	>7	no	reason
D2	>7	yes	argument
D3	$=7$	no	reason
D4	>7	no	argument
D5	<7	yes	reason

Build a decision tree classifier from these labeled examples using (1) the ID3 algorithm and (2) information gain.

Do you agree with these features? Which could we observe instead?

How could we proceed to take into account cosine similarity as an additional feature?

5 Appendix - Theoretical Concepts

5.1 Cosine Similarity

$$sim_C(\vec{q_j}, \vec{d_k}) = \frac{\vec{q_j} \cdot \vec{d_k}}{||\vec{q_j}|| \cdot ||\vec{d_k}||} = \frac{\sum_i q_{i,j} \cdot w_{i,k}}{\sqrt{\sum_i q_{i,j}^2} \cdot \sqrt{\sum_i w_{i,k}^2}}$$

where:

- $df_i = \#$ of documents in which term i occurs,
- $idf_i = \log_{10}\left(\frac{|D|}{df_i}\right)$,
- $tf_{i,j} = \#$ times term i occurs in document j,
- $w_{i,j} = tf_{i,j} \cdot idf_i$.

5.2 Information Gain

$$InfGain(S, X) \;=\; H(S) \;-\; \left[\sum_{x \in Values(X)} \left(\frac{|S_x|}{|S|} \;\cdot\; H(S_x)\right)\right]$$

where:

- S denotes the set of examples, $\vec{x}$ an example, $X \in F$ a feature and x a value
- $S_x = \{\vec{x} \in S \mid X(\vec{x}) = x\}$: examples $\vec{x}$ where feature X has value x,
- $H(S) = -\sum\{p(c_i) \cdot \log_2 p(c_i) \mid 1 \leq i \leq |C|\}$ is the *Shannon entropy* of sample S.

5.3 ID3 Algorithm

```
procedure DECISON-TREE($S, F, c_m$)
    if $S = \emptyset$ then return $c_m$;                                  ▷ stop condition 1
    else if every $\vec{x} \in S$ classified $c$ then return $c$;          ▷ stop condition 2
    else if $F = \emptyset$ then return $c_m$;                             ▷ stop condition 3
    else
        $X_{best} \leftarrow$ CHOOSE-ATT($F, S$);                          ▷ rank $F$ by information gain
        $tree \leftarrow$ new tree rooted at $X_{best}$;
        $c_m \leftarrow$ MAJ-VAL($S$);                                     ▷ choose the most frequent class in $S$
        for $x$ in $X_{best}$ do
            $S_x \leftarrow \{\vec{x} \in S \mid \vec{x}$ has $x\}$;
            $subtree \leftarrow$ DECISION-TREE($S_x, F \setminus X_{best}, c_m$);   ▷ recursive call
            add branch to $tree$ with label $x$ and subtree $subtree$;
        end for
        return $tree$
    end if
end procedure
```

where:

- S is a sample, c_m is the most frequent class in S, and F the attributes,
- the procedure CHOOSE-ATT(S, F) ranks F w.r.t. S by *information gain*,
- the procedure MAJ-VAL(S) returns *most frequent class* in S.

This is a *recursive* procedure that learns the smallest decision tree consistent with S (Occam's razor).

5.4 Evaluation

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN} \qquad F1 = \frac{2PR}{P + R}$$

where:

- $TP = \#$ of true predictions, $FP = \#$ of false predictions,
- $FP = \#$ of *missed* predictions (happens when the classifier returns $\perp$: classifiers need not be *total* functions).