# Conditional Probabilities

## Definition

The conditional probability that event $E$ occurs given that event $F$ occurs is

$$\Pr(E \mid F) = \frac{\Pr(E \cap F)}{\Pr(F)}.$$

The conditional probability is only well-defined if $\Pr(F) > 0$.

- By conditioning on $F$ we restrict the sample space to the set $F$.
- $\Pr(E \mid F)$ defines a proper probability function on the sample space $F$.

# Bayes' Law

**Theorem (Bayes' Law)**

*Assume that* $E_1, E_2, \ldots, E_n$ *are mutually disjoint sets such that* $\cup_{i=1}^{n} E_i = \Omega$, *then*

$$\Pr(E_j \mid B) = \frac{\Pr(E_j \cap B)}{\Pr(B)} = \frac{\Pr(B \mid E_j)\Pr(E_j)}{\sum_{i=1}^{n} \Pr(B \mid E_i)\Pr(E_i)}.$$

# Application: Naive Bayes Classifier

Automatic classifications of objects to categories

- Junk mail filter
- Classify text documents into one of several subjects/topics - politics, business, sport, science, religion,....
- Classify to genres - "editorials", "movie-reviews", "news",...
- Classify opinions: like, hate, neutral,...
- Identify the language of a document: English, French, Chinese,...
- recommendation systems
- ...

# Bayes Classifier

- We have a training set of classified documents.
- We assume that the category of a document can be deduced from the words used in the document.
- Training Phase: Learn from the training set the conditional probabilities that a given word appears in a document of a given category - Supervised learning
- Classification Case: Compute the conditional probability that a new document belongs to a given category conditioned on the words that appear (or don't appear) in the document.

# The "bag of words" model

- A document is represented by the set of words in the document
- We ignore locality relation and number of occurrences of words.
- We clean the documents by removing HTML commands, stop words, "s"'s and "ing"'s , etc. ("tokenizing"), so we are left with the keywords of the document.

# Bayes Classifier

- $X_w^d$ - the event "word $w$ appears in document $d$". Classes $C_1, C_2, \ldots$

$$Pr(d \in C_i \mid \cap_{w \in d} X_w^d) = \frac{Pr((d \in C_i) \cap (\cap_{w \in d} X_w^d))}{Pr(\cap_{w \in d} X_w^d)}$$

$$= \frac{Pr(\cap_{w \in d} X_w^d \mid d \in C_i) Pr(d \in C_i)}{\sum_j Pr(\cap_{w \in d} X_w^d \mid d \in C_j) Pr(d \in C_j)}$$

- The classification of $d$ is

$$\arg\max_{C_i \in \mathcal{C}} Pr(d \in C_i \mid \cap_{w \in d} X_w^d)$$

- It's the maximum likelihood category

# The "Naive" assumption

- Problem; Assume $n$ categories and a dictionary of $k$ keywords. Need to estimate $k = n \cdot 2^k$ probabilities:

$$Pr(\cap_{w \in d} X_w^d \mid d \in C_i)$$

- Practical solution: Assume that occurrences of words are independent

$$Pr(\cap_{w \in d} X_w^d \mid d \in C_i) = \prod_{w \in d} Pr(X_w^d \mid d \in C_i)$$

$$Pr(d \in C_i \mid \cap_{w \in d} X_w^d) = \frac{\prod_{w \in d} Pr(X_w^d \mid d \in C_i) Pr(d \in C_i)}{\sum_j \prod_{w \in d} Pr(X_w^d \mid d \in C_j) Pr(d \in C_j)}$$

- The classification of $d$ is

$$\arg\max_{C_i \in \mathcal{C}} Pr(d \in C_i \mid \cap_{w \in d} X_w^d)$$

# Naive Bayes Classifier

Using the training data:

- For each category $C \in \mathcal{C}$ and keyword $w$ we compute

$$Pr(\text{page includes } w \mid \text{page in } c) = \frac{|\{d \mid d \in C \cap w \in d\}|}{|\{d \mid d \in C\}|}$$

- For each category $C \in \mathcal{C}$, compute

$$Pr(\text{page in } C) = \frac{|\{d \mid d \in C\}|}{|\{\text{all pages}\}|}$$

# Naive Bayes Classifier

- We are looking for the category $C_{i^*}$ that maximizes

$$Pr(d \in C_i \mid \cap_w X_w^d) = \frac{\prod_w Pr(X_w^d \mid d \in C_i) Pr(d \in C_i)}{\sum_j \prod_w Pr(X_w^d \mid d \in C_j) Pr(d \in C_j)}$$

,

- We only need to consider the nomirators. The classification of $d$ is

$$\arg\max_{C_i \in \mathcal{C}} Pr(d \in C_i \mid \cap_w X_w^d) =$$

$$\arg\max_{C_i \in \mathcal{C}} \prod_w Pr(X_w^d \mid d \in C_i) Pr(d \in C_i)$$

# Two technical problems

- Underflow prevention: Multiplying lots of probabilities can result in very small quantities and floating-point underflow.
- Solution: We do all computations in log's. The classification of $d$ is

$$\arg\max_{C_i \in \mathcal{C}} Pr(d \in C_i \mid \cap_w X_w^d) =$$

$$\arg\max_{C_i \in \mathcal{C}} [\log Pr(d \in C_i) + \sum_w \log Pr(X_w^d \mid d \in C_i)]$$

- Zero probability events: What if $w \in d$ but we have seen no training documents in category $C$ with keyword $w$?
- In that case $Pr(X_w^d \mid d \in C_i) = 0$, cannot take the log
- Zero probabilities cannot be conditioned away, no matter the other evidence!
- Solution: Smoothing

$$Pr(X_w^d \mid C) = \frac{|\{d \mid (d \in C) \cap (w \in d)\}| + 1}{|\{d \mid d \in C\}| + k}$$

- No zero probability events.

# Naive Bayes Classifier Algorithm

**Training phase:** $TD$ set of classified documents, $n$ categories.

- For each category $C \in \mathcal{C}$ and keyword $w$ compute

$$Pr(\text{page includes } w \mid \text{page in } C) = \frac{|\{d \mid d \in C \cap w \in d\}| + 1}{|\{d \mid d \in C\}| + k}$$

- For each category $C \in \mathcal{C}$, compute

$$Pr(\text{page in } C) = \frac{|\{d \mid d \in C\}|}{|TD|}$$

- Store $n + nk$ probabilities.

To **classify** a new document $d$, compute

$$\arg\max_{C_i \in \mathcal{C}} [\log Pr(d \in C_i) + \sum_{w \in d} \log Pr(X_w^d \mid d \in C_i)]$$

Execute $O(n(|d| + 1))$ operations