Statistical learning

Russell and Norvig Chapter 20

Statistical learning

- Example:
  Suppose your favorite candy comes in two flavors: cherry and lime (both wrapped in the same opaque wrapper).
  The candy is sold in five types of bags that are indistinguishable:
  - H1: 100% cherry
  - H2: 75% cherry, 25% lime
  - H3: 50% cherry, 50% lime
  - H4: 25% cherry, 75% lime
  - H5: 100% lime
- You open a new bag of candy, and start having some: your data are a vector of observations $\mathbf{d} = d_1,...,d_N$
- Objective: predict the color of the next piece of candy

Statistical Learning

- Given the data we can evaluate the probability of the data under each hypothesis:

$$P(h_i|\mathbf{d}) = \alpha P(\mathbf{d}|h_i)P(h_i)$$

  $P(h_i)$ - the **prior** (say (0.1,0.2,0.4,0.2,0.1) for our example)
  $P(\mathbf{d}|h_i)$ - the **likelihood** of the data
  Typically we make the assumption that the observations are i.i.d. (independent, identically distributed) so:

$$P(\mathbf{d}|h_i) = \prod_j P(d_j|h_i)$$

- For example $P(lime,lime,lime,lime \mid h_3) = 0.5^4$

Maximum a posteriori (MAP)

- The most probable hypothesis (h_{MAP}): the one that maximizes $P(h_i|\mathbf{d})$

$$P(h_i|\mathbf{d}) = \alpha P(\mathbf{d}|h_i)P(h_i)$$

Maximum likelihood

- MAP is chosen to maximize

$$P(h_i|\mathbf{d}) = \alpha P(\mathbf{d}|h_i)P(h_i)$$

- Suppose we have no reason to prefer one hypothesis over another (uniform prior) then MAP reduces to choosing h_i that maximizes $P(\mathbf{d}|h_i)$.
- This is the *maximum likelihood* hypothesis, h_{ML}

Maximum likelihood parameter learning

- Suppose we buy a bag of cherry-lime candy from a manufacturer whose lime-cherry proportions are unknown.
- Hypothesis: h_θ where θ is the proportion of cherry candy
- The likelihood of a dataset:

$$P(\mathbf{d}|h_\theta) = \prod_{j=1}^{N} P(d_j|h_\theta) = \theta^c(1-\theta)^{N-c}$$

  c - number of cherry candy picked

Maximum likelihood parameter learning

- For convenience we work with the log likelihood:

$$L(\mathbf{d}|h_\theta) = \log P(\mathbf{d}|h_\theta) = \sum_{j=1}^{N} \log P(d_j|h_\theta) = c\log\theta + (N-c)\log(1-\theta)$$

- The maximum likelihood value of θ:

$$\frac{dL(\mathbf{d}|h_\theta)}{d\theta} = \frac{c}{\theta} - \frac{N-c}{1-\theta} = 0 \quad \Rightarrow \quad \theta = \frac{c}{N}$$

- The ML hypothesis states that the proportion of cherry candy is equal to the proportion observed so far.

Classification using a probabilistic model

- Task: classify an instance based on a vector of attribute values, $\boldsymbol{x}$

$$c_{MAP} = \arg\max_j P(c_j|\mathbf{x})$$

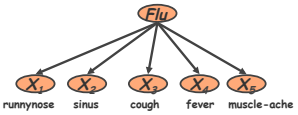
c_j - the possible classes

We model each class separately using a probability distribution $P(\boldsymbol{x} \mid c_j)$

$$c_{MAP} = \arg\max_j \frac{P(\mathbf{x}|c_j)P(c_j)}{P(\mathbf{x})}$$

Need to make simplifying assumptions on the form of $P(\boldsymbol{x}|c_j)$

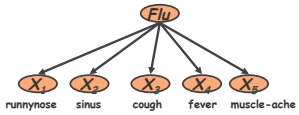
Naïve Bayes classifier



$$P(X_1,\ldots,X_5|C) = P(X_1|C)P(X_2|C)\cdots P(X_5|C)$$

- **Conditional independence assumption:** features are independent of each other given the class

Learning the model

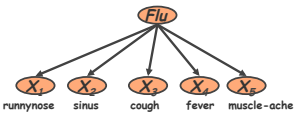


$$P(X_1,\ldots,X_5|C) = P(X_1|C)P(X_2|C)\cdots P(X_5|C)$$

- Estimate parameters using maximum likelihood:

$$P(c_j) = \frac{N(C=c_j)}{N} \qquad P(x_i|c_j) = \frac{N(X_i=x_i, C_j=c_j)}{N(C=c_j)}$$

Learning the model



- What happens if we have no examples where X_5=*true* and *Flu=false*?
- The model gives $P(Flu = false \mid X_5 = true) = 0$
- Solution: pseudo-counts

Pseudo-counts

$$P(x_i|c_j) = \frac{N(X_i=x_i, C_j=c_j)+1}{N(C=c_j)+|X_i|}$$

# of values of X_i

- Pseudo-counts are a way to avoid overfitting -- taking into account what we haven't observed.

Properties of Naïve Bayes

- Number of parameters linear in number of attributes
- Simple learning algorithm: no search involved in finding h_{ML}.
- Training time: linear in number of training examples and number of features.
- Classification time: (of a single example) linear in number of features.
- Performs reasonably well on a variety of problems

Underflow prevention

- Multiplying lots of probabilities, can result in floating-point underflow.
- Since log(*xy*) = log(*x*) + log(*y*), it is better to perform all computations by summing logs of probabilities rather than multiplying probabilities.
- Standard trick for probabilistic models

$$P(X_1, \ldots, X_5|C) = P(X_1|C)P(X_2|C)\cdots P(X_5|C)$$

Text categorization

- How to represent a document?
- Bag of words representation: a document represented by a vector that counts how many times each word appears in it.

# of times word i appears in the text

# of times word i appears in category j

$$P(X_i = x_i|C = c_j) = \frac{N(X_i = x_i, C = c_j) + 1}{N(C = c_j) + |X_i|}$$