Why build Models?

- To predict (identify) something
  - Diagnosis
  - Best therapy
  - Prognosis
  - Cost
- To understand something
  - Structure of model may correspond to structure of reality

Where do models come from?

- Pure induction from data
- Even so, need some “space” of models to explore
- Maximum A-posteriori Probability (MAP)
  \[ P(h_i|d) = \alpha P(d|h_i)P(h_i) \]
- Maximum Likelihood (ML)
  \[ P(h_i|d) = \alpha P(d|h_i) \]
  - Assumes uniform priors over all hypotheses in the space
- A-priori knowledge, expressed in
  - Structure of the space of models
  - \( P(h_i) \)
  - Adjustments to observed data

An Example

(Russell & Norvig)

- Surprise Candy Corp. makes two flavors of candy: cherry and lime
- Both flavors come in the same opaque wrapper
- Candy is sold in large bags, which have one of the following distributions of flavors, but are visually indistinguishable:
  - \( h_1 \): 100% cherry
  - \( h_2 \): 75% cherry, 25% lime
  - \( h_3 \): 50% cherry, 50% lime
  - \( h_4 \): 25% cherry, 75% lime
  - \( h_5 \): 100% lime
- Relative prevalence of these types of bags is \((0.1, 0.2, 0.4, 0.2, 0.1)\)
- As we eat our way through a bag of candy, predict the flavor of the next piece; actually a probability distribution.
Bayesian Learning

- Calculate the probability of each hypothesis given the data
  \[ P(h_i|d) = \alpha P(d|h_i)P(h_i) \]
- To predict the probability distribution over an unknown quantity, \( X \),
  \[ P(X|d) = \sum_i P(X|d, h_i)P(h_i|d) = \sum_i P(X|h_i)P(h_i|d) \]
- If the observations \( d \) are independent, then
  \[ P(d|h_i) = \prod_j P(d_j|h_i) \]
- E.g., suppose the first 10 candies we taste are all lime
  \[ P(d|h_3) = 0.5^{10} \approx 0.001 \]

Learning Hypotheses and Predicting from Them

- (a) probabilities of \( h_i \) after \( k \) lime candies; (b) prob. of next lime

- MAP prediction: predict just from most probable hypothesis
  - After 3 limes, \( h_5 \) is most probable, hence we predict lime
  - Even though, by (b), it’s only 80% probable

Observations

- Bayesian approach asks for prior probabilities on hypotheses!
- Natural way to encode bias against complex hypotheses: make their prior probability very low
- Choosing \( h_{MAP} \) to maximize \( P(h_i|d) = \alpha P(d|h_i)P(h_i) \)
  - is equivalent to minimizing \( -\log P(d|h_i) - \log P(h_i) \)
  - but as we know that entropy is a measure of information, these two terms are
    - # of bits needed to describe the data given hypothesis
    - # bits needed to specify the hypothesis
  - Thus, MAP learning chooses the hypothesis that maximizes compression of the data; Minimum Description Length principle
- Regularization is similar to 2nd term—penalty for complexity
- Assuming uniform priors on hypotheses makes MAP yield \( h_{ML} \), the maximum likelihood hypothesis, which maximizes \( P(h_i|d) = \alpha P(d|h_i) \)

Learning More Complex Hypotheses

- Input:
  - Set of cases, each of which includes
    - numerous features: categorical labels, ordinals, continuous
    - these correspond to the independent variables
- Output:
  - For each case, a result, prediction, classification, etc., corresponding to the dependent variable
    - In regression problems, a continuous output
    - a designated feature the model tries to predict
    - In classification problems, a discrete output
    - the category to which the case is assigned
- Task: learn function \( f(\text{input})=\text{output} \)
  - that minimizes some measure of error
### Linear Regression

- General form of the function
  \[ y = f(x_1, x_2, \ldots, x_n) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n \]

- For each case:
  \[ \hat{y}_i = f(x_{1,i}, x_{2,i}, \ldots, x_{n,i}) = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \cdots + \beta_n x_{n,i} \]

- Find \( \beta_j \) to minimize some function of \( (y_i - \hat{y}_i) \) over all \( y_i \)
  - e.g., mean squared error: \( \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n} \)

### Logistic Regression

- Logistic function:
  \[ f(x) = \frac{1}{1 + e^{-x}} \]

- E.g., how risk factors contribute to probability of death
  - \( \beta_i \) are the log odds ratios \( \log \frac{P(y_i = 1 | x_i)}{P(y_i = 0 | x_i)} \)

### More sophisticated models

- Nearest Neighbor Methods
- Classification Trees
- Artificial Neural Nets
- Support Vector Machines
- Bayes Networks (much on this, later)
- Rough Sets, Fuzzy Sets, etc. (see 6.873/HST951 or other ML classes)

### How?

- Given: pile of training data, all cases labeled with gold standard outcome
- Learn “best” model
- Gather new test data, also all labeled with outcomes
- Test performance of model on new test data
- Simple, no?