דף נוסחאות – הבחינה הסופית

372-1-3105 : מדעי הנתונים ובינה עסקית - תש"ף, סמסטר ב'

Information Theory

- Conditional Entropy $H(Y/X) = - \Sigma\ p(x,y)*\log p\ (y/x)$
- Mutual Information $I(X;Y) =$
  $$\sum_{x,y} p(x,y) \bullet \log \frac{p(y/x)}{p(y)}$$

Conditional Mutual Information $I(X;Y/Z)$

$$= \sum_{x,y} p(x,y,z) \bullet \log \frac{p(x,y/z)}{p(x/z) \bullet p(y/z)}$$

- Fano's Inequality: $H\ (Y/ X_1 \ldots X_n) \leq H\ (P_e) + P_e \log_2 (m-1)$

Classification and Decision Trees

- Confidence Interval for an Error Rate:

$$Err_{Test} \pm z_\alpha \sqrt{\frac{Err_{Test}(1-Err_{Test})}{n}}$$

- Confidence Interval for a difference between error rates:

$$\hat{d} \pm z_\alpha \sqrt{\frac{Err_{Test1}(1-Err_{Test1})}{n_1} + \frac{Err_{Test2}(1-Err_{Test2})}{n_2}}$$

- Expected information needed to classify a tuple in D (before using A):
  $$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$
- Expected information needed to classify a tuple in D (after using A):

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

- Information Gain:
  $Gain(A) = Info(D) - Info_A(D)$
- Expected number of records in C_i, for class j:

$$e'_{ij} = \frac{e_j}{\sum_{j=1}^{c} e_j} \sum_{j=1}^{c} o_{ij}$$

- Chi-Square Statistic:

$$\sum_{j=1}^{c} \sum_{i=1}^{v} \frac{(o_{ij} - e'_{ij})^2}{e'_{ij}} \Big|_{H_0} \sim \chi_\alpha^{\ 2}((v-1)(c-1))$$

- Apparent (pessimistic) error rate:
  $$q = \frac{N - n_C + 0.5}{N}$$
- Entropy induced by threshold T:
  $$E(A,T;S) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2)$$
- Split Information:

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2 \left(\frac{|D_j|}{|D|} \right)$$

- Gini index: $$gini(T) = 1 - \sum_{j=1}^{n} p_j^2$$
- Gini split (T):
  $$gini_{split}(T) = \frac{N_1}{N} gini(T_1) + \frac{N_2}{N} gini(T_2)$$
- Twoing Splitting Rule:
  $$\frac{p_L p_R}{4} \left[\sum_j \left| p(j/t_L) - p(j/t_R) \right| \right]^2$$
- Cost-complexity function (CART):
  $$R_\alpha(T) = R(T) + \alpha \cdot \left| \tilde{T} \right|$$

IFN

- IFN Conditional mutual information at a node z:

MI ($A_{i'}$; A_i / z) =

$$\sum_{j=0}^{M_i-1} \sum_{j'=0}^{M_{i'}-1} P(V_{ij};V_{i'j'};z) \bullet \log \frac{P(V_{i'j'}^{ij}/z)}{P(V_{i'j'}/z) \bullet P(V_{ij}/z)}$$

- IFN Likelihood-Ratio Statistic:

$$G^2(A_{i'};A_i/z)=2\bullet(\ln 2)\bullet E^*\bullet MI(A_{i'};A_i/z)$$

$$G^2|_{H_0}\sim\chi^2((NI_{i'}(z)-1)\cdot(NT_i(z)-1))$$

- Conditional Mutual Information in a Layer *i'*:

$$MI(A_{i'};A_i)=\sum_{\substack{z\in Layer_{i'}\\ Split(z)=true}} MI(A_{i'};A_i/z)$$

- IFN Connection Weight:

$$w_z^{ij}=P(V_{ij};z)\bullet\log\frac{P(V_{ij}/z)}{P(V_{ij})}$$

  - Conditional Mutual Information (Split):

$$\sum_{t=0}^{M_i-1}\sum_{y=1}^{2}P(S_y;C_t;z)\bullet\log\frac{P(S_y;C_t\ /S,z)}{P(S_y/S,z)\bullet P(C_t/S,z)}$$

Artificial Neural Networks

- Sigmoid Activation Unit:

$$I_j=\sum_i w_{ij}O_i+\theta_j,$$

$$O_j=\frac{1}{1+e^{-I_j}}.$$

- Error in the output layer:

$$Err_j=O_j(1-O_j)(T_j-O_j),$$

- Error in the hidden layer:

$$Err_j=O_j(1-O_j)\sum_k Err_k w_{jk},$$

- Gradient-Descent Rule:

$$\Delta w_{ij}=(l)Err_jO_i.$$

$$w_{ij}=w_{ij}+\Delta w_{ij}.$$

$$\Delta\theta_j=(l)Err_j.$$

$$\theta_j=\theta_j+\Delta\theta_j.$$

- ReLU activation function:

$$f(x)=x,\ x\geq 0$$

Bayesian Learning

- Naïve Bayes Classifier:

$$C_{NB}=\arg\max_{C_i}\ P(C_i)*\prod_{k=1}^{n}P(x_k|C_i)$$

- m-estimate: $\frac{n_c+mp}{n+m}$

- Laplacian-estimate: $\frac{n_c+1}{n+K}$

- Joint probability in Bayesian network:

$$P(x_1,\ldots,x_n)=\prod_{i=1}^{n}P(x_i\,|\,Parents(X_i))$$

k-Nearest Neighbors, Clustering

- Euclidean distance:

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sqrt{\sum_{t=1}^{T}[x_{ti}-x_{tj}]^2}$$

- Distance-weighted k-NN:

$$\hat{f}(q)=\arg\max_{v\in V}\sum_{i=1}^{k}w_i\delta(v,f(x_i))$$

$$w_i=\frac{1}{d(x_q-x_i)^2}$$

- Distance measure for <u>symmetric</u> binary variables:

$$d(i,j)=\frac{b+c}{a+b+c+d}$$

- Distance measure for <u>asymmetric</u> binary variables:

$$d(i,j)=\frac{b+c}{a+b+c}$$

- Distance measure for nominal variables:

$$d(i,j)=\frac{p-m}{p}$$

- Distance measure for variables of mixed types:

$$d(i,j)=\frac{\sum_{f=1}^{p}\delta_{ij}^{(f)}d_{ij}^{(f)}}{\sum_{f=1}^{p}\delta_{ij}^{(f)}}$$

- Rank for an ordinal variable:

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

- Cluster centroid: $C_m = \frac{\sum_{i=1}^{N} (t_{ip})}{N}$

Kernel-based Methods and SVM

- Nadaraya-Watson Kernel-weighted Average:

$$\hat{f}(x) = \frac{\sum_{i=1}^{N} K_\lambda(x_0, x_i) y_i}{\sum_{i=1}^{N} K_\lambda(x_0, x_i)}$$

$$K_\lambda(x_0\, x_i) = D\left(\frac{\|x - x_0\|}{\lambda}\right)$$

$$D(t) = \begin{cases} 0.75(1 - t^2) \text{ if } |\text{t}| \le 1 \\ 0 \text{ otherwise} \end{cases}$$

- Linear SVM:

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$y_j(w^T x_j + b) \ge 1$$

- Nonlinear SVM:

$$g(x_j) = \sum_{i \in SV} \alpha_i y_i K(x_i, x_j) + b$$

- Polynomial kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^p$$

- Gaussian (Radial-Basis Function (RBF)) kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$$

- Sigmoid:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\beta_0 \mathbf{x}_i^T \mathbf{x}_j + \beta_1)$$

Data Preparation

- min-max normalization:

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

- z-score normalization:

$$v' = \frac{v - mean_A}{stand\_dev_A}$$

- normalization by decimal scaling:

$$v' = \frac{v}{10^j}$$

- Simple Moving Average:

$$\hat{Y}_{t+1} = \frac{Y_t + Y_{t-1} + \ldots + Y_{t-k+1}}{k}$$

- Weighted Moving Average:

$$\hat{Y}_{t+1} = w_t Y_t + w_{t-1} Y_{t-1} + \ldots + w_{t-k+1} Y_{t-k+1}$$

$$where: \quad w_t + w_{t-1} + \ldots + w_{t-k+1} = 1$$

- Exponential Moving Average:

$$F_t = \alpha Y_{t-1} + (1 - \alpha) F_{t-1}$$