### Introduction

- **Questions:**
  - Assessment of the expected error of a learning algorithm: Is the error rate of $1 - \text{NN}$ less than $2\%$?
  - Comparing the expected errors of two algorithms: Is k-NN more accurate than MLP?
  - Training/validation/test sets

### Algorithm Preference

**Criteria (Application-dependent):**
- Misclassification error, or risk (loss functions)
- Training time/space complexity
- Testing time/space complexity
- Interpretability
- Easy programmability

---

### Training

1. **Introduction**
2. **Training**
   - Response Surface Design
   - Cross-Validation & Resampling
3. **Measuring Classifier Performance**
4. **Comparing Classifiers**
   - Comparing Two Classifiers
   - Comparing Multiple Classifiers
   - Comparing Over Multiple Datasets
Factors and Response

Algorithm Preference

Criteria (Application-dependent):
- Misclassification error, or risk (loss functions)
- Training time/space complexity
- Testing time/space complexity
- Interpretability
- Easy programmability

Select desired response function measuring the desired criteria.

Response surface design

The need for multiple training/validation sets
\(\{T_i, V_i\}_i\): Training/validation sets of fold \(i\)

K-fold cross-validation: Divide \(X\) into \(k\) sets, \(X_i\)

\[
V_1 = X_1 \\
T_1 = X_2 \cup X_3 \cup \ldots \cup X_k = X - X_1 \\
V_2 = X_2 \\
T_2 = X_1 \cup X_3 \cup \ldots \cup X_k = X - X_2 \\
\vdots \\
V_k = X_k \\
T_k = X - X_k
\]

Each pair of \(T_i\) share \(k - 2\) parts
5x2 Cross-Validation

- Perform 2-Fold Cross-Validation 5 times

\[
\begin{align*}
V_1 &= X_1^{(1)} \quad T_1 = X_2^{(1)} \\
V_1 &= X_1^{(2)} \quad T_1 = X_2^{(2)} \\
V_1 &= X_1^{(3)} \quad T_1 = X_2^{(3)} \\
V_1 &= X_1^{(4)} \quad T_1 = X_2^{(4)} \\
V_1 &= X_1^{(5)} \quad T_1 = X_2^{(5)}
\end{align*}
\]

using 5 different divisions into half

Measuring Classifier Performance

<table>
<thead>
<tr>
<th>True Class</th>
<th>Predicted Class</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>TP: true positive</td>
<td>FN: false negative</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>FP: false positive</td>
<td>TN: true negative</td>
<td></td>
</tr>
</tbody>
</table>

Error rate = \# of errors/\# of instances

= (FN + FP)/N

Recall = \# of found positives / \# of positives

= TP/(TP + FN) = sensitivity = hit rate

Precision = \# of found positives / \# of found

= TP/(TP + FP)

Specificity = TN/(TN + FP)

False alarm rate = FP/(FP + TN) = 1 − Specificity

Receiver Operating Characteristics

(a) Example ROC curve

(b) Different ROC curves for different classifiers

Comparing Classifiers

- Introduction
- Training
  - Response Surface Design
  - Cross-Validation & Resampling
- Measuring Classifier Performance
- Comparing Classifiers
  - Comparing Two Classifiers
  - Comparing Multiple Classifiers
  - Comparing Over Multiple Datasets
McNemar’s Test

\[ H_0: \mu_0 = \mu_1 \]

- Single training/validation set

\[ e_{00}: \text{Number of examples misclassified by both} \]
\[ e_{01}: \text{Number of examples misclassified by 1 but not by 2} \]
\[ e_{10}: \text{Number of examples misclassified by 2 but not by 1} \]
\[ e_{11}: \text{Number of examples correctly classified by both} \]

- Under \( H_0 \) we expect \( e_{01} = e_{10} \)

\[
\frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}} \sim \chi^2_1
\]

- Accept with confidence \( 1 - \alpha \) if \( < \chi^2_{\alpha,1} \)

K-fold Cross-Validated Paired \( t \) Test

- Use K-fold c-v to get K training/validation folds

\[ p^1_i, p^2_i: \text{Errors of classifiers 1 and 2 on fold } i \]
\[ p_i = p^1_i - p^2_i: \text{Paired difference on fold } i \]
\[ H_0: p_i \text{ has mean 0} \]

\[
m = \frac{\sum_{i=1}^{K} p_i}{K}
\]
\[
s^2 = \frac{\sum_{i=1}^{K} (p_i - m)^2}{K - 1}
\]
\[
m\sqrt{\frac{K}{s}} \sim t_{K-1}
\]

- Accept if in \((-t_{\alpha/2,K-1}, t_{\alpha/2,K-1})\)

Comparing \( L > 2 \) Classifiers

- Analysis of variance (Anova)

\[ H_0: \mu_1 = \mu_2 = \ldots = \mu_L \]

- Errors of \( L \) algorithms on \( K \) folds

\[ X_{ij} \sim N(\mu_j, \sigma^2) j = 1, \ldots, L \ i = 1, \ldots, K \]

- Anova constructs two estimates of \( \sigma^2 \)

\[ \hat{\sigma}_b^2 = \frac{\sum_{j=1}^{L} (m_j - m)^2}{L(K-1)} \]

\[ \frac{\sum_{j=1}^{L} \sum_{i=1}^{K} (X_{ij} - m_j)^2}{\hat{\sigma}_w^2} \sim \chi^2_{L,K-1} \]

\[ \frac{\hat{\sigma}_w^2}{\sigma^2} \sim F_{L-1,L(K-1)} \]

Accept \( H_0 \) if \( < F_{\alpha,L-1,L(K-1)} \)
Comparing Over Multiple Datasets

- Comparing two algorithms:
  - **Sign test**: Count how many times A beats B over N datasets, and check if this could have been by chance if A and B did have the same error rate

- Comparing multiple algorithms
  - **Kruskal-Wallis test**: Calculate the average rank of all algorithms on N datasets, and check if these could have been by chance if they all had equal error
  
  If KW rejects, we do pairwise posthoc tests to find which ones have significant rank difference