Data reduction

- Feature selection
  - Lecture 3
- Numerosity reduction
  - Selective (Non-adaptive) algorithms
  - Generalizing (Adaptive) algorithms
- Data compression
  - Next lecture
- Data modelling
  - Next lecture
Nearest Neighbour Rule

For a given query point \( q \), assign the class of the nearest neighbour.

Compute the \( k \) nearest neighbours and assign the class by majority vote.

Nearest Neighbour Issues

- **Expensive**
  - To determine the nearest neighbour of a query point \( q \), must compute the distance to all \( N \) training examples
    - Pre-sort training examples into fast data structures (kd-trees)
    - Compute only an approximate distance (LSH)
    - Remove redundant data (condensing)

- **Storage Requirements**
  - Must store all training data \( P \)
    - Remove redundant data (condensing)
    - Pre-sorting often increases the storage requirements

- **High Dimensional Data**
  - “Curse of Dimensionality”
    - Required amount of training data increases exponentially with dimension
    - Computational cost also increases dramatically
    - Partitioning techniques degrade to linear search in high dimension
Questions

- What distance measure to use?
  - Often Euclidean distance is used
  - Locally adaptive metrics
  - More complicated with non-numeric data, or when different dimensions have different scales

- Choice of $k$?
  - Cross-validation
  - 1-NN often performs well in practice
  - k-NN needed for overlapping classes
  - Re-label all data according to k-NN, then classify with 1-NN
  - Reduce $k$-NN problem to 1-NN through dataset editing

Exact Nearest Neighbour

- Asymptotic error (infinite sample size) is less than twice the Bayes classification error
  - Requires a lot of training data

- Expensive for high dimensional data ($d>20$?)

- $O(Nd)$ complexity for both storage and query time
  - $N$ is the number of training examples, $d$ is the dimension of each sample
  - This can be reduced through dataset editing/condensing
Decision Regions

Each cell contains one sample, and every location within the cell is closer to that sample than to any other sample.

A Voronoi diagram divides the space into such cells.

Every query point will be assigned the classification of the sample within that cell. The decision boundary separates the class regions based on the 1-NN decision rule.

Knowledge of this boundary is sufficient to classify new points. The boundary itself is rarely computed; many algorithms seek to retain only those points necessary to generate an identical boundary.

Condensing

- Aim is to reduce the number of training samples
- Retain only the samples that are needed to define the decision boundary

- Decision Boundary Consistent – a subset whose nearest neighbour decision boundary is identical to the boundary of the entire training set

- Minimum Consistent Set – the smallest subset of the training data that correctly classifies all of the original training data
Condensing

Condensed Nearest Neighbour (CNN) Hart 1968
- Incremental
- Order dependent
- Neither minimal nor decision boundary consistent
- $O(n^3)$ for brute-force method
- Can follow up with reduced NN
  [Gates72]
- Remove a sample if doing so does not cause any incorrect classifications

Proximity Graphs

- Condensing aims to retain points along the decision boundary
- How to identify such points?
  - Neighbouring points of different classes
    \[ \text{NNG} \subseteq \text{MST} \subseteq \text{RNG} \subseteq \text{GG} \subseteq \text{DT} \]
- Proximity graphs provide various definitions of “neighbour”

NNG = Nearest Neighbour Graph
MST = Minimum Spanning Tree
RNG = Relative Neighbourhood Graph
GG = Gabriel Graph
DT = Delaunay Triangulation
**Proximity Graphs: Delaunay**

- The Delaunay Triangulation is the dual of the Voronoi diagram.
- Three points are each others neighbours if their tangent sphere contains no other points.
- Voronoi editing: retain those points whose neighbours (as defined by the Delaunay Triangulation) are of the opposite class.
- The decision boundary is identical.
- Conservative subset.
- Retains extra points.
- Expensive to compute in high dimensions.

**Proximity Graphs: Gabriel**

- The Gabriel graph is a subset of the Delaunay Triangulation.
- Points are neighbours only if their (diametral) sphere of influence is empty.
- Does not preserve the identical decision boundary, but most changes occur outside the convex hull of the data points.
- Can be computed more efficiently.

Green lines denote "Tomek links."
Proximity Graphs: RNG

- The Relative Neighbourhood Graph (RNG) is a subset of the Gabriel graph.
- Two points are neighbours if the "lune" defined by the intersection of their radial spheres is empty.
- Further reduces the number of neighbours.
- Decision boundary changes are often drastic, and not guaranteed to be training set consistent.

Additionally, there is a mention of a Matlab demo, indicating a practical application or demonstration of the concepts discussed.
Dataset Reduction: Editing

- Training data may contain noise, overlapping classes
  - starting to make assumptions about the underlying distributions
- Editing seeks to remove noisy points and produce smooth decision boundaries – often by retaining points far from the decision boundaries
- Results in homogenous clusters of points

Wilson Editing

- Wilson 1972
- Remove points that do not agree with the majority of their k nearest neighbours

Earlier example

Overlapping classes

Wilson editing with k=7
Multi-edit

- Multi-edit [Devijer & Kittler ‘79]
- Repeatedly apply Wilson editing to random partitions
- Classify with the 1-NN rule

1. **Diffusion:** divide data into $N \geq 3$ random subsets
2. **Classification:** Classify $S_i$ using 1-NN with $S_{(i+1) \mod N}$ as the training set ($i = 1..N$)
3. **Editing:** Discard all samples incorrectly classified in (2)
4. **Confusion:** Pool all remaining samples into a new set
5. **Termination:** If the last $I$ iterations produced no editing then end; otherwise go to (1)

Combined Editing/Condensing

- First edit the data to remove noise and smooth the boundary
- Then condense to obtain a smaller subset
Where are we?

- Simple method, pretty powerful rule
- Can be made to run fast
- Requires a lot of training data

- Edit to reduce noise, class overlap
- Condense to remove redundant data

Questions

- What distance measure to use?
  - Often Euclidean distance is used
  - Locally adaptive metrics
  - More complicated with non-numeric data, or when different dimensions have different scales

- Choice of $k$?
  - Cross-validation
  - 1-NN often performs well in practice
  - $k$-NN needed for overlapping classes
  - Re-label all data according to $k$-NN, then classify with 1-NN
  - Reduce $k$-NN problem to 1-NN through dataset editing
Numerosity reduction - overview

- Selective algorithms
  - CNN – selects instances near the decision boundary
  - RNN – subset selected by CNN is further reduced
  - IB3 – similar to CNN, statistical reduction (confidence)
  - DROP3 – deletes instances, that will not change the classification of other instances

- Adaptive algorithms
  - Prototype – creates new strategic data points
  - Chen – new instances as centroids of original data
  - RSP1 – Chen including class ballancing

RNN

subset selected by CNN is further reduced

IB3

similar to CNN, statistical reduction (confidence)
**DROP3**

deletes instances, that will not change the classification of other instances

---

**Prototype**

creates new strategic data points
Numerosity reduction and class balancing

- **Motivation:**
  - Balance the training data
  - Remove noisy examples lying on the wrong side of the decision border

- **Methods:**
  - Over-sampling method: Smote
  - Data cleaning methods: Tomek links, and Edited Nearest Neighbor Rule

- Adapted from: Study of the Behavior of Several Methods for Balancing Machine Learning Training Data, Gustavo E. A. Batista, University of Ottawa
Balancing aware reduction methods

- Baseline Methods
  - Random over-sampling
  - Random under-sampling
- Under-sampling Methods
  - Tomek links
  - Condensed Nearest Neighbor Rule
  - One-sided selection
  - CNN + Tomek links
  - Neighborhood Cleaning Rule
- Over-sampling Methods
  - Smote
- Combination of Over-sampling method with Under-sampling method
  - Smote + Tomek links
  - Smote + ENN

Baseline Methods

- Baseline methods
  - Random over-sampling
    - random replication of minority class examples
    - Can increase the likelihood of occurring overfitting
  - Random over-sampling
    - random elimination of majority class examples
    - Can discard potentially useful data that could be important for the induction process
Tomek links

- To remove both noise and borderline examples
- Tomek link
  - \( E_i, E_j \) belong to different classes, \( d(E_i, E_j) \) is the distance between them.
  - A \( (E_i, E_j) \) pair is called a Tomek link if there is no example \( E_l \), such that \( d(E_l, E_i) < d(E_i, E_j) \) or \( d(E_l, E_j) < d(E_i, E_j) \).

Tomek links – instances from dominant class removed
Remember - Condensed Nearest Neighbor Rule (CNN rule)

- To pick out points near the boundary between the classes
- Algorithm:
  - Let \( E \) be the original training set
  - Let \( E' \) contains all positive examples from \( S \) and one randomly selected negative example
  - Classify \( E \) with the 1-NN rule using the examples in \( E' \)
  - Move all misclassified example from \( E \) to \( E' \)
- Sensitive to noise. Noisy examples are likely to be misclassified, many of them will be added to the training set.

One-sided selection vs CNN+Tomek links

- One-sided selection
  - Tomek links + CNN
  - CNN + Tomek links
- Finding Tomek links is computationally demanding, it would be computationally cheaper if it was performed on a reduced data set.
Neighborhood Cleaning Rule

- To remove majority class examples
- Emphasize more data cleaning than data reduction
- Algorithm:
  - Find three nearest neighbors for each example $E_i$ in the training set
  - If $E_i$ belongs to majority class, & the three nearest neighbors classify it to be minority class, then remove $E_i$
  - If $E_i$ belongs to minority class, and the three nearest neighbors classify it to be majority class, then remove the three nearest neighbors

Smote: Synthetic Minority Over-sampling Technique

- To form new minority class examples by interpolating between several minority class examples that lie together.
- In "feature space" rather than "data space"
- Algorithm: For each minority class example, introduce synthetic examples along the line segments joining any/all of the $k$ minority class nearest neighbors.
- Note: Depending upon the amount of over-sampling required, neighbors from the $k$ nearest neighbors are randomly chosen.
- For example: if we are using 5 nearest neighbors, if the amount of over-sampling needed is 200%, only two neighbors from the five nearest neighbors are chosen and one sample is generated in the direction of each.
Smote: Synthetic Minority Oversampling Technique

- Synthetic samples are generated in the following way:
  - Take the difference between the feature vector (sample) under consideration and its nearest neighbor.
  - Multiply this difference by a random number between 0 and 1.
  - Add it to the feature vector under consideration.

Consider a sample (6,4) and let (4,3) be its nearest neighbor.

(6,4) is the sample for which k-nearest neighbors are being identified.

(4,3) is one of its k-nearest neighbors.

Let:

\[
\begin{align*}
  f_{1_1} &= 6 \quad f_{2_1} = 4 \quad f_{2_1} - f_{1_1} = -2 \\
  f_{1_2} &= 4 \quad f_{2_2} = 3 \quad f_{2_2} - f_{1_2} = -1
\end{align*}
\]

The new samples will be generated as

\[
(f_1', f_2') = (6,4) + \text{rand}(0-1) \times (-2,-1)
\]

\text{rand}(0-1) generates a random number between 0 and 1.

Smote + Tomek links

- Problem with Smote: might introduce the artificial minority class examples too deeply in the majority class space.
- Tomek links: data cleaning
- Instead of removing only the majority class examples that form Tomek links, examples from both classes are removed.
Smote + Tomek links

ENN removes any example whose class label differs from the class of at least two of its three nearest neighbors.

ENN remove more examples than the Tomek links does

ENN remove examples from both classes
Experimental Evaluation

- 10 methods
- 13 UCI data sets which have different degrees of imbalance

Data balance concerns

Facts:
- In spite of a large degree of imbalance, the data sets Letter-a and Nursery obtained almost 100% AUC

Conclude:
- domains with non-overlapping classes do not seem to be problematic for learning no matter the degree of imbalance
- But when allied to highly overlapped classes, it can significantly decrease the number of minority class examples correctly classified.
- The relationship between training set size and performance
  - For small imbalanced data sets, when a large degree of class overlapping exists and the class is further divided into subclusters, the minority class is poorly represented by an excessively reduced number of examples
  - For large data sets, the effect of these complicating factors seems to be reduced, the minority class is better represented by a larger number of examples.

Table 3: Data sets summary descriptions.

<table>
<thead>
<tr>
<th>Data set</th>
<th>#Examples</th>
<th>#Attributes (quant., qual.)</th>
<th>Class (min., maj.)</th>
<th>Class % (min., maj.)</th>
<th>Majority Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pima</td>
<td>708</td>
<td>8 (8.0)</td>
<td>(1, 0)</td>
<td>(34.77%, 65.23%)</td>
<td>65.23%</td>
</tr>
<tr>
<td>German</td>
<td>1000</td>
<td>20 (7, 13)</td>
<td>(bad, good)</td>
<td>(80.00%, 70.00%)</td>
<td>70.00%</td>
</tr>
<tr>
<td>Post-operative</td>
<td>90</td>
<td>8 (4, 5)</td>
<td>(sleep, remainder)</td>
<td>(26.67%, 73.33%)</td>
<td>73.33%</td>
</tr>
<tr>
<td>Haberman</td>
<td>306</td>
<td>3 (3.0)</td>
<td>(Die, Survive)</td>
<td>(26.47%, 73.53%)</td>
<td>73.53%</td>
</tr>
<tr>
<td>Splice-oo</td>
<td>3176</td>
<td>60 (0.60)</td>
<td>(0, remainder)</td>
<td>(24.00%, 75.91%)</td>
<td>75.91%</td>
</tr>
<tr>
<td>Splice-ce</td>
<td>3176</td>
<td>60 (0.60)</td>
<td>(0, remainder)</td>
<td>(25.00%, 75.01%)</td>
<td>75.01%</td>
</tr>
<tr>
<td>Vehicle</td>
<td>546</td>
<td>18 (18.0)</td>
<td>(vm, remainder)</td>
<td>(23.52%, 76.48%)</td>
<td>76.48%</td>
</tr>
<tr>
<td>Letter-vowel</td>
<td>20000</td>
<td>16 (16.0)</td>
<td>(all vowels, remainder)</td>
<td>(19.39%, 80.61%)</td>
<td>80.61%</td>
</tr>
<tr>
<td>New-thyroid</td>
<td>215</td>
<td>5 (5.0)</td>
<td>(hypo, remainder)</td>
<td>(16.28%, 83.72%)</td>
<td>83.72%</td>
</tr>
<tr>
<td>Ecoli</td>
<td>396</td>
<td>7 (7.0)</td>
<td>(bel, remainder)</td>
<td>(10.42%, 89.58%)</td>
<td>89.58%</td>
</tr>
<tr>
<td>Statlog</td>
<td>648</td>
<td>30 (30.0)</td>
<td>(a, remainder)</td>
<td>(9.73%, 90.27%)</td>
<td>90.27%</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>9 (9.0)</td>
<td>(ve-win-frac-proc, remainder)</td>
<td>(7.94%, 92.06%)</td>
<td>92.06%</td>
</tr>
<tr>
<td>Letter-a</td>
<td>12060</td>
<td>16 (16.0)</td>
<td>(0, remainder)</td>
<td>(3.95%, 96.05%)</td>
<td>96.05%</td>
</tr>
<tr>
<td>Nursery</td>
<td>12060</td>
<td>8 (8.0)</td>
<td>(not recogn, remainder)</td>
<td>(3.55%, 96.45%)</td>
<td>96.45%</td>
</tr>
</tbody>
</table>
# Over sampling results

- results for the original and over sampled data sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Pruning</th>
<th>Original</th>
<th>Over Rand</th>
<th>Over Smote</th>
<th>Smote+Tomek</th>
<th>Smote+ENN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pima</td>
<td>yes</td>
<td>81.33(5.11)</td>
<td>85.32(4.17)</td>
<td>85.49(5.17)</td>
<td>84.46(5.81)</td>
<td>83.09(1.77)</td>
</tr>
<tr>
<td>German</td>
<td>no</td>
<td>79.04(5.84)</td>
<td>84.65(3.80)</td>
<td>80.74(5.43)</td>
<td>81.76(4.78)</td>
<td>80.91(4.36)</td>
</tr>
<tr>
<td>Post-operative</td>
<td>no</td>
<td>85.64(1.44)</td>
<td>85.66(4.31)</td>
<td>84.51(4.55)</td>
<td>84.02(3.94)</td>
<td>83.90(3.70)</td>
</tr>
<tr>
<td>Haberman</td>
<td>no</td>
<td>71.03(15.03)</td>
<td>71.76(13.43)</td>
<td>68.19(26.02)</td>
<td>68.81(36.74)</td>
<td>68.72(31.91)</td>
</tr>
<tr>
<td>Splice-je</td>
<td>no</td>
<td>97.76(5.06)</td>
<td>98.29(9.47)</td>
<td>98.46(9.87)</td>
<td>98.26(9.51)</td>
<td>97.00(7.94)</td>
</tr>
<tr>
<td>Splice-ei</td>
<td>no</td>
<td>99.59(0.40)</td>
<td>99.86(0.44)</td>
<td>99.02(0.44)</td>
<td>99.37(0.44)</td>
<td>99.87(0.94)</td>
</tr>
<tr>
<td>Vehicle</td>
<td>no</td>
<td>98.49(0.84)</td>
<td>99.14(0.73)</td>
<td>98.90(0.98)</td>
<td>98.90(0.98)</td>
<td>97.92(1.09)</td>
</tr>
<tr>
<td>Letter-vowel</td>
<td>no</td>
<td>90.45(0.90)</td>
<td>90.13(0.75)</td>
<td>90.04(0.85)</td>
<td>90.04(0.85)</td>
<td>88.22(0.90)</td>
</tr>
<tr>
<td>New-thyroid</td>
<td>no</td>
<td>98.97(0.63)</td>
<td>98.89(0.42)</td>
<td>98.90(0.20)</td>
<td>98.90(0.20)</td>
<td>98.94(0.22)</td>
</tr>
<tr>
<td>E-Coli</td>
<td>no</td>
<td>94.78(0.38)</td>
<td>94.80(2.68)</td>
<td>94.91(1.84)</td>
<td>94.91(1.84)</td>
<td>94.90(1.72)</td>
</tr>
<tr>
<td>Satimage</td>
<td>no</td>
<td>93.78(1.91)</td>
<td>95.34(1.25)</td>
<td>95.43(1.03)</td>
<td>95.43(1.03)</td>
<td>95.67(1.18)</td>
</tr>
<tr>
<td>Flag</td>
<td>no</td>
<td>45.00(5.18)</td>
<td>79.15(28.72)</td>
<td>73.04(20.16)</td>
<td>72.49(23.88)</td>
<td>62.21(36.66)</td>
</tr>
<tr>
<td>Glass</td>
<td>no</td>
<td>43.11(4.16)</td>
<td>69.29(6.91)</td>
<td>66.91(27.02)</td>
<td>66.91(27.02)</td>
<td>64.44(20.81)</td>
</tr>
<tr>
<td>Letter-a</td>
<td>no</td>
<td>99.61(0.40)</td>
<td>99.77(0.30)</td>
<td>99.91(0.12)</td>
<td>99.91(0.12)</td>
<td>99.91(0.12)</td>
</tr>
<tr>
<td>Nursery</td>
<td>no</td>
<td>99.75(0.11)</td>
<td>99.90(0.01)</td>
<td>99.91(0.01)</td>
<td>99.90(0.01)</td>
<td>99.75(0.61)</td>
</tr>
</tbody>
</table>

# Under sampling results

- results for the original and under sampled data sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Pruning</th>
<th>Original</th>
<th>Under Rand</th>
<th>Under Smote</th>
<th>Smote+Tomek</th>
<th>Smote+ENN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pima</td>
<td>yes</td>
<td>81.64(8.87)</td>
<td>79.60(6.26)</td>
<td>80.55(6.51)</td>
<td>77.89(5.37)</td>
<td>81.61(4.48)</td>
</tr>
<tr>
<td>German</td>
<td>no</td>
<td>81.49(4.20)</td>
<td>80.98(5.82)</td>
<td>81.71(3.69)</td>
<td>83.11(4.05)</td>
<td>82.13(4.81)</td>
</tr>
<tr>
<td>Post-operative</td>
<td>no</td>
<td>79.85(3.05)</td>
<td>79.85(5.50)</td>
<td>79.48(5.01)</td>
<td>78.87(4.27)</td>
<td>79.20(3.15)</td>
</tr>
<tr>
<td>Haberman</td>
<td>no</td>
<td>85.43(3.32)</td>
<td>82.53(9.59)</td>
<td>81.70(4.00)</td>
<td>85.90(3.99)</td>
<td>82.96(3.32)</td>
</tr>
<tr>
<td>Splice-je</td>
<td>no</td>
<td>88.36(1.28)</td>
<td>92.07(12.09)</td>
<td>91.49(8.23)</td>
<td>91.49(8.23)</td>
<td>92.10(7.32)</td>
</tr>
<tr>
<td>Splice-ei</td>
<td>no</td>
<td>97.46(1.10)</td>
<td>97.77(0.40)</td>
<td>97.97(0.12)</td>
<td>97.97(0.12)</td>
<td>97.97(0.12)</td>
</tr>
<tr>
<td>Vehicle</td>
<td>no</td>
<td>98.56(0.40)</td>
<td>99.17(0.56)</td>
<td>98.82(0.53)</td>
<td>99.18(0.45)</td>
<td>98.90(0.39)</td>
</tr>
<tr>
<td>Letter-vowel</td>
<td>no</td>
<td>97.80(0.40)</td>
<td>98.64(0.60)</td>
<td>98.42(1.03)</td>
<td>98.41(0.90)</td>
<td>98.71(0.97)</td>
</tr>
<tr>
<td>New-thyroid</td>
<td>no</td>
<td>98.70(0.40)</td>
<td>98.84(0.37)</td>
<td>98.97(0.37)</td>
<td>98.97(0.37)</td>
<td>98.97(0.37)</td>
</tr>
<tr>
<td>E-Coli</td>
<td>no</td>
<td>99.26(0.28)</td>
<td>98.29(0.31)</td>
<td>98.39(0.22)</td>
<td>98.39(0.22)</td>
<td>98.29(0.19)</td>
</tr>
<tr>
<td>Satimage</td>
<td>no</td>
<td>94.87(5.00)</td>
<td>94.79(1.04)</td>
<td>94.54(1.04)</td>
<td>94.98(3.85)</td>
<td>94.39(0.74)</td>
</tr>
<tr>
<td>Flag</td>
<td>no</td>
<td>99.56(0.28)</td>
<td>98.52(0.46)</td>
<td>99.65(0.35)</td>
<td>99.65(0.35)</td>
<td>99.65(0.35)</td>
</tr>
<tr>
<td>Glass</td>
<td>no</td>
<td>99.78(0.82)</td>
<td>99.55(0.21)</td>
<td>97.75(0.35)</td>
<td>99.50(0.08)</td>
<td>99.50(0.08)</td>
</tr>
<tr>
<td>Letter-a</td>
<td>no</td>
<td>97.80(0.22)</td>
<td>98.84(1.03)</td>
<td>98.57(1.01)</td>
<td>98.57(1.01)</td>
<td>98.57(1.01)</td>
</tr>
<tr>
<td>Nursery</td>
<td>no</td>
<td>98.78(0.79)</td>
<td>98.84(1.03)</td>
<td>98.57(1.01)</td>
<td>98.57(1.01)</td>
<td>98.57(1.01)</td>
</tr>
</tbody>
</table>
Over/under sampling

Facts:
- Pruning rarely leads to an improvement in AUC for the original and balanced data sets.
- All best results (results in bold) were obtained by the over-sampling methods.
- Over-sampling methods are better ranked than the under-sampling methods.
- Smote+Tomek and Smoke+ENN are generally ranked among the best for data sets with a small number of positive examples.

Explanation:
- The loss of performance is directly related to the lack of minority class examples in conjunction with other complicating factors.
- Over sampling is the methods that most directly attack the problem of the lack minority class examples.

Results for **pruned** decision tree as classifier

<table>
<thead>
<tr>
<th>Data set</th>
<th>P2°</th>
<th>P3°</th>
<th>P4°</th>
<th>P5°</th>
<th>P6°</th>
<th>P7°</th>
<th>P8°</th>
<th>P9°</th>
<th>P10°</th>
<th>P11°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prima</td>
<td>RdOvr</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>Tnk</td>
<td>NCL</td>
<td>Original</td>
<td>RdUdr</td>
<td>CNN+Tnk</td>
<td>CNN</td>
<td>OSS</td>
</tr>
<tr>
<td>German</td>
<td>RdOvr</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>Smt</td>
<td>NCL</td>
<td>Original</td>
<td>RdUdr</td>
<td>CNN+Tnk</td>
<td>OSS</td>
<td>Original</td>
</tr>
<tr>
<td>Post-operative</td>
<td>RdOvr</td>
<td>Smt+ENN</td>
<td>Smt</td>
<td>NCL</td>
<td>Original</td>
<td>RdUdr</td>
<td>CNN+Tnk</td>
<td>OSS</td>
<td>Original</td>
<td></td>
</tr>
<tr>
<td>Haberman</td>
<td>Smt+ENN</td>
<td>Smt+Tnk</td>
<td>RdOvr</td>
<td>NCL</td>
<td>Original</td>
<td>RdUdr</td>
<td>Tnk</td>
<td>OSS</td>
<td>CNN</td>
<td>Original</td>
</tr>
<tr>
<td>Splice-in</td>
<td>RdOvr</td>
<td>Original</td>
<td>Tnk</td>
<td>Smt</td>
<td>CNN</td>
<td>NCL</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>CNN+Tnk</td>
<td>RdUdr</td>
</tr>
<tr>
<td>Splice-out</td>
<td>Smt</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>CNN+Tnk</td>
<td>OSS</td>
<td>RdOvr</td>
<td>Tnk</td>
<td>CNN</td>
<td>NCL</td>
<td>Original</td>
</tr>
<tr>
<td>Vehicle</td>
<td>RdOvr</td>
<td>Smt</td>
<td>Smt+Tnk</td>
<td>OSS</td>
<td>CNN</td>
<td>Original</td>
<td>CNN+Tnk</td>
<td>Tnk</td>
<td>NCL</td>
<td>Smt+ENN</td>
</tr>
<tr>
<td>Letter-vowel</td>
<td>Smt+ENN</td>
<td>Smt+Tnk</td>
<td>RdOvr</td>
<td>NCL</td>
<td>Original*</td>
<td>CNN</td>
<td>NCL</td>
<td>Original*</td>
<td>CNN+Tnk</td>
<td>RdUdr</td>
</tr>
<tr>
<td>New-thyroid</td>
<td>Smt+ENN</td>
<td>Smt+Tnk</td>
<td>RdOvr</td>
<td>RdUdr</td>
<td>CNN</td>
<td>Original</td>
<td>Tnk</td>
<td>CNN+Tnk</td>
<td>NCL</td>
<td>OSS</td>
</tr>
<tr>
<td>E. coli</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>RdOvr</td>
<td>NCL</td>
<td>Tnk</td>
<td>RdUdr</td>
<td>Original</td>
<td>OSS</td>
<td>CNN+Tnk</td>
<td>CNN</td>
</tr>
<tr>
<td>Satimage</td>
<td>Smt+ENN</td>
<td>Smt+Tnk</td>
<td>RdOvr</td>
<td>NCL</td>
<td>Tnk</td>
<td>Original*</td>
<td>OSS</td>
<td>CNN+Tnk</td>
<td>RdUdr</td>
<td>OSS</td>
</tr>
<tr>
<td>Flag</td>
<td>RdOvr</td>
<td>Smt+ENN</td>
<td>Smt+Tnk</td>
<td>CNN+Tnk</td>
<td>Smt+Tnk</td>
<td>RdUdr</td>
<td>CNN*</td>
<td>OSS</td>
<td>Tnk</td>
<td>Original</td>
</tr>
<tr>
<td>Glass</td>
<td>Smt+ENN</td>
<td>RdOvr</td>
<td>NCL</td>
<td>Smt+Tnk</td>
<td>Original</td>
<td>Tnk</td>
<td>RdUdr</td>
<td>CNN+Tnk</td>
<td>OSS</td>
<td>CNN</td>
</tr>
<tr>
<td>Letter-a</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>RdOvr</td>
<td>OSS</td>
<td>Original</td>
<td>Tnk</td>
<td>CNN+Tnk</td>
<td>NCL</td>
<td>CNN</td>
<td>RdUdr</td>
</tr>
<tr>
<td>Nursey</td>
<td>RdOvr</td>
<td>Tnk</td>
<td>Original</td>
<td>NCL</td>
<td>CNN*</td>
<td>OSS*</td>
<td>Smt+Tnk</td>
<td>Smt+ENN</td>
<td>CNN+Tnk</td>
<td>RdUdr</td>
</tr>
</tbody>
</table>

- Light gray color: results obtained with over sampling methods
- Dark gray color: results obtained with the original data sets
- Methods marked with an asterisk obtained statistically inferior results when compared to the top ranked method
Results for **unpruned** decision tree as classifier

<table>
<thead>
<tr>
<th>Table 7: Performance ranking for original and balanced data sets for unpruned decision trees.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data set</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>IM</td>
</tr>
<tr>
<td>German</td>
</tr>
<tr>
<td>Post-operative</td>
</tr>
<tr>
<td>Haberman</td>
</tr>
<tr>
<td>Splice-ie</td>
</tr>
<tr>
<td>Splice-ei</td>
</tr>
<tr>
<td>Vehicle</td>
</tr>
<tr>
<td>Letter-vowel</td>
</tr>
<tr>
<td>New-thyroid</td>
</tr>
<tr>
<td>E-Cell</td>
</tr>
<tr>
<td>Satimage</td>
</tr>
<tr>
<td>Flag</td>
</tr>
<tr>
<td>Glass</td>
</tr>
<tr>
<td>Letter-a</td>
</tr>
<tr>
<td>Nursery</td>
</tr>
</tbody>
</table>

**Conclusions from experimental results**

- **Class imbalance does not systematically hinder the performance of learning systems.**
- Besides class imbalance, the degree of data overlapping among the classes is another factor that lead to the decrease in performance of learning algorithms.
- **Experiments show that in general, over-sampling methods provide more accurate results than under-sampling methods.**
- Random over-sampling is very competitive to more complex over-sampling methods.
References


Some results on medium datasets

<table>
<thead>
<tr>
<th></th>
<th>Execution Time(sec)</th>
<th>% Reduction</th>
<th>C4.5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>% Ac. Trn</td>
</tr>
<tr>
<td>C4.5</td>
<td>1</td>
<td></td>
<td>98.82</td>
</tr>
<tr>
<td>Cnn (*)</td>
<td>4</td>
<td>97.32</td>
<td>93.68</td>
</tr>
<tr>
<td>Drop1 (*)</td>
<td>254</td>
<td>96.72</td>
<td>95.19</td>
</tr>
<tr>
<td>Drop2 (*)</td>
<td>215</td>
<td>89.57</td>
<td>95.42</td>
</tr>
<tr>
<td>Drop3 (*)</td>
<td>338</td>
<td>96.25</td>
<td>95.14</td>
</tr>
<tr>
<td>Emm</td>
<td>139</td>
<td>5.82</td>
<td>99.26</td>
</tr>
<tr>
<td>Jb2 (*)</td>
<td>2</td>
<td>97.78</td>
<td>93.06</td>
</tr>
<tr>
<td>Jcf (*)</td>
<td>386</td>
<td>90.04</td>
<td>97.85</td>
</tr>
<tr>
<td>Mcs</td>
<td>101</td>
<td>2.66</td>
<td>98.97</td>
</tr>
<tr>
<td>Multiled</td>
<td>1778</td>
<td>13.99</td>
<td>99.77</td>
</tr>
<tr>
<td>Renn</td>
<td>489</td>
<td>6.47</td>
<td>99.31</td>
</tr>
<tr>
<td>Run (*)</td>
<td>13017</td>
<td>96.88</td>
<td>99.16</td>
</tr>
<tr>
<td>Shrink</td>
<td>206</td>
<td>4.89</td>
<td>99.23</td>
</tr>
<tr>
<td>Vem</td>
<td>94</td>
<td>1.04</td>
<td>98.84</td>
</tr>
<tr>
<td>Jb3 (*)</td>
<td>42</td>
<td>71.67</td>
<td>94.94</td>
</tr>
<tr>
<td>Rnnb (*)</td>
<td>34525</td>
<td>90.02</td>
<td>98.11</td>
</tr>
<tr>
<td>Enmsr (*)</td>
<td>37802</td>
<td>90.02</td>
<td>97.99</td>
</tr>
<tr>
<td>GGA</td>
<td>66157</td>
<td>62.53</td>
<td>98.72</td>
</tr>
<tr>
<td>SGA</td>
<td>54656</td>
<td>62.91</td>
<td>98.38</td>
</tr>
<tr>
<td>CHC (*)</td>
<td>8072</td>
<td>99.29</td>
<td>98.29</td>
</tr>
<tr>
<td>PBIL (*)</td>
<td>32942</td>
<td>73.13</td>
<td>98.71</td>
</tr>
</tbody>
</table>

From: Using Evolutionary Algorithms as Instance Selection for Data Reduction in KDD: An Experimental Study
Final remarks

- As you see, non method is universal
- You should be aware of
  - noise
  - data balance
- Condensing, editing, oversampling can help you to prepare good training sample
- Always use independent testing data!