The use of Hyperspectral data for tree species discrimination: Combining binary classifiers

by

Xolani Dastile

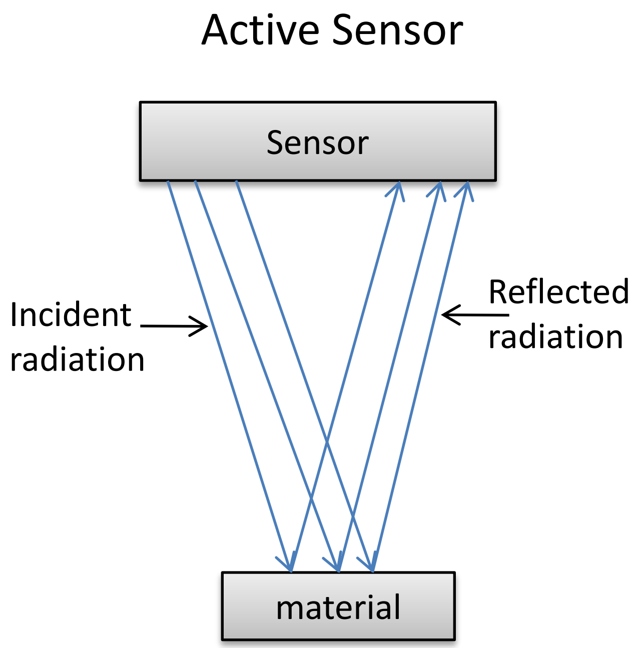
supervised
by
Professor G. Jager
Doctor P. Debba

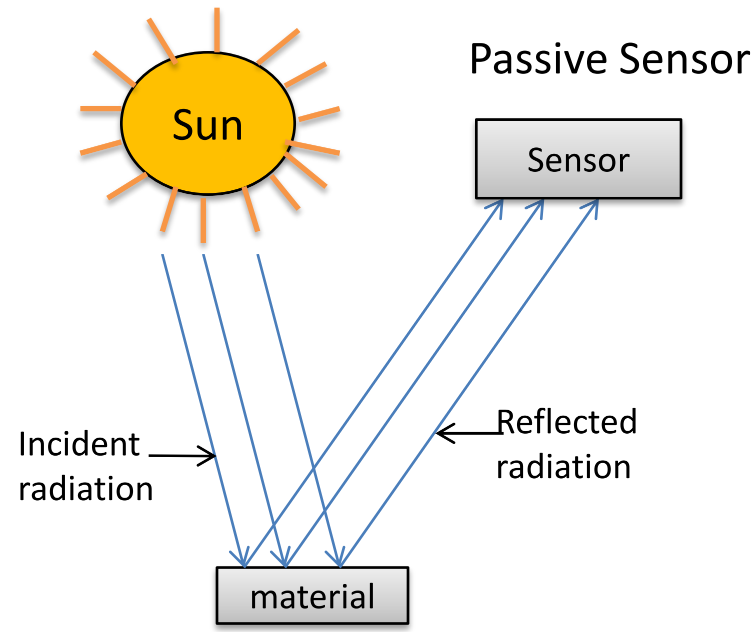
1. Outline

- Hyperspectral Remote sensing
- Data description
- Classification
  - Classifiers: Nearest neighbour and Neural Networks
  - Estimate of the error probability
- Binary classifiers
- Combining binary classifiers: Error Correcting Output Codes
- Discussion
- References

2. Hyperspectral Remote Sensing

- Hyperspectral remote sensors record reflectances in many narrow and closely spaced bands.
- Reflectance is a ratio of the reflected radiation to the incident radiation *i.e* reflectance = $\frac{R_i}{R_r}$ (R_i incident radiation, R_r reflected radiation).





3. Data description

- Aim: Assess tree species diversity in Kruger National Park
- Study: Record hyperspectral measurements of leaf samples with Analytical Spectral Device (ASD) spectrometer
- The hyperspectral data consists of 2101 spectral bands (400nm-2500nm) for seven plant tree species in the area.

class 1	*Lonchocarpus Capassa*	*LC*	25 samples
class 2	*Combretum Apiculatum*	CA	23 samples
class 3	*Combretum Heroense*	*CH*	20 samples
class 4	*Combretum Zeyherrea*	*CZ*	19 samples
class 5	*Gymnospora Buxifolia*	*GB*	21 samples
class 6	*Gymnospora Senegalensis*	*GS*	18 samples
class 7	*Terminalia Sericia*	*TS*	22 samples

4. Reflectance spectra for *CA* and *CH* species

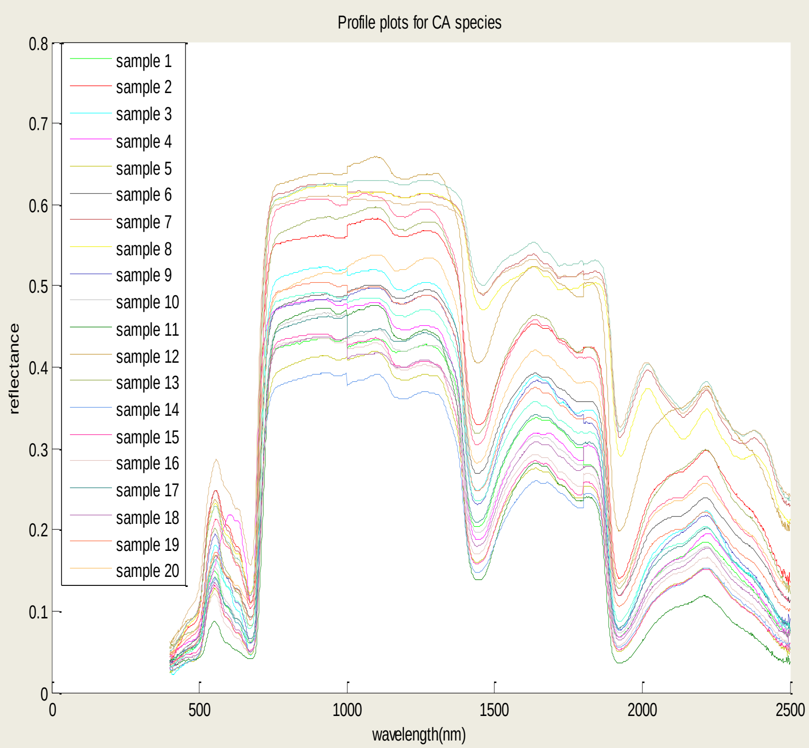


Figure: Reflectance spectra of the samples for Combretum Apiculatum

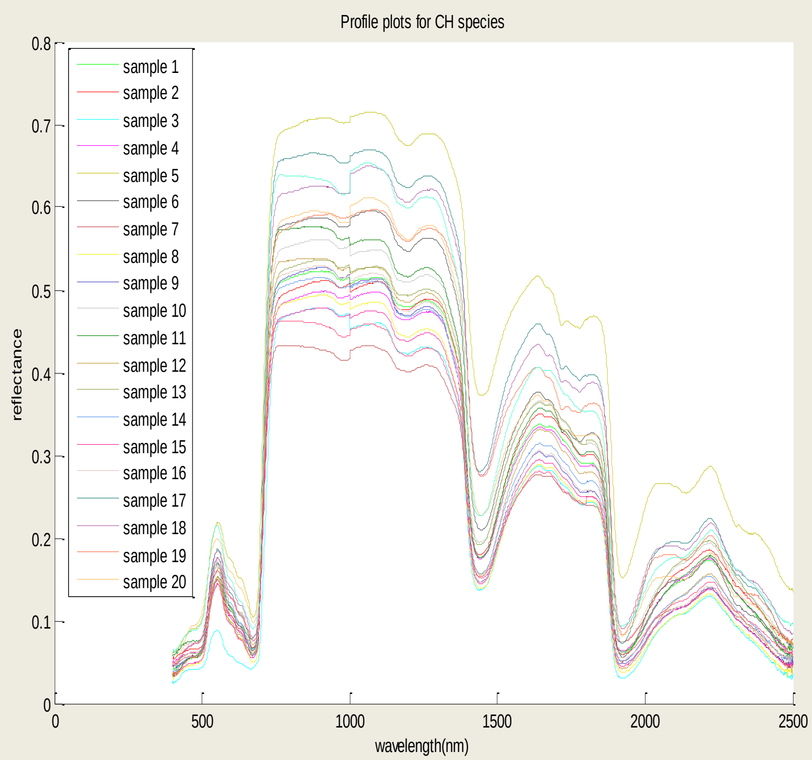
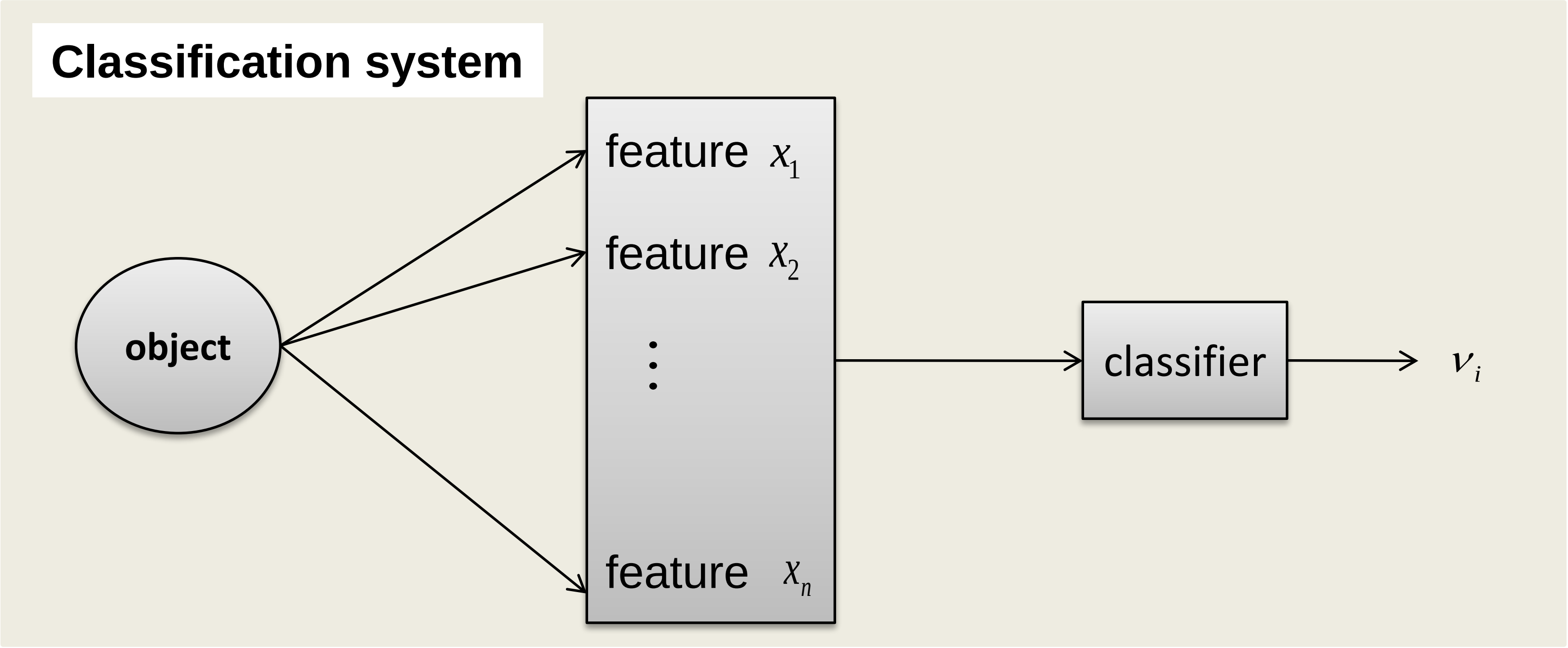


Figure: Reflectance spectra of the samples for Combretum Heroense(CH)

Note: high within-class variability, low between-class variability

5. Classification

- Aim of classification: Assign object into one class v_i of a set of given classes $\{ v_1, v_2, \ldots, v_c \}$.
- Classification = supervised learning: training data with known classes available.



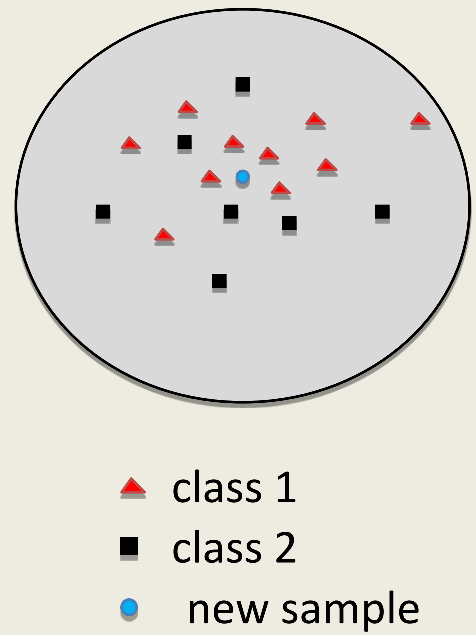
6. Classifiers: K-nearest neighbour classifier

- Given learning task $\{(x^1,t_1),(x^2,t_2),\ldots,(x^p,t_p)\}$
  ($x^i \in R^n$ feature vectors,
  $t_i \in \{ v_1,\ldots, v_c\}$ class labels.)

- For a new object $x \in R^n$:
  + determine k closest samples
  + Assign to x the class of the majority of the k closest samples

- Closeness is measured e.g. by using Euclidean distance

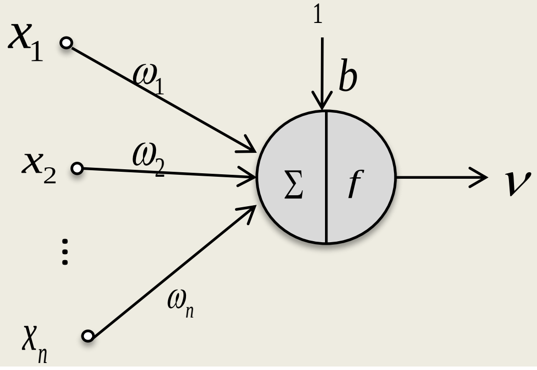
$$d\left(x^i, x\right) = \sqrt{\left(x_1^i - x_1\right)^2 + \left(x_2^i - x_2\right)^2 + \ldots + \left(x_n^i - x_n\right)^2}$$



For 5-nearest neighbour classification: assign new sample to class 1.

6. Classifiers: Neural networks (I)

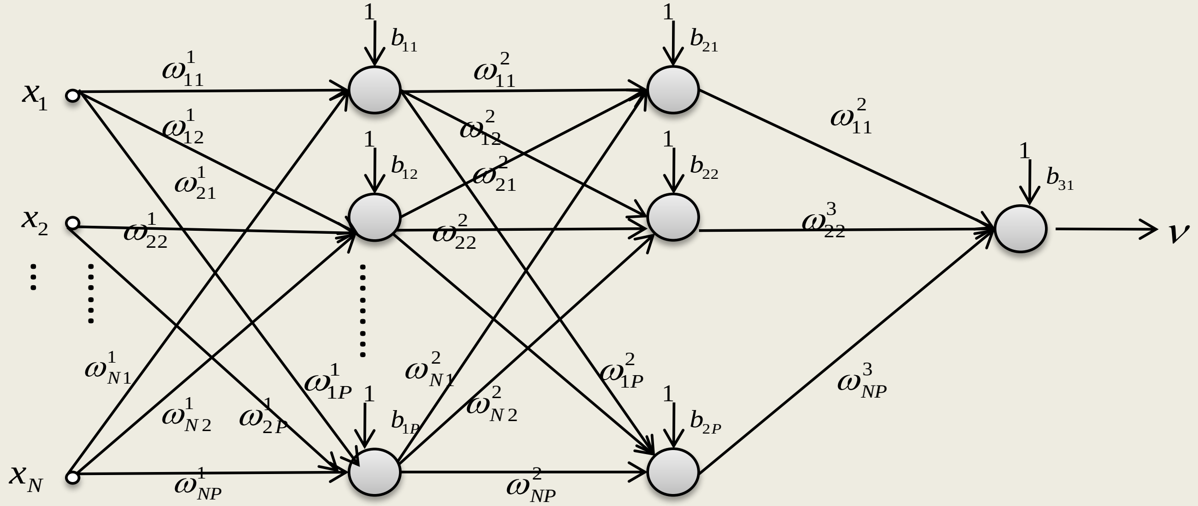
Single artificial neuron:



$$b+\omega_1 x_1+\omega_2 x_2+\ldots+\omega_n x_n=b+\sum_{k=1}^{n}\omega_k x_k$$

$$f\left(b+\sum_{k=1}^{n}\omega_k x_k\right)=v$$

Multi-layer feedforward neural network:



- x_i - inputs
- ω_i - weights
- b - bias
- f -transfer function e.g.

$$f(x)=\frac{1}{1+e^{-x}}$$

- v - output

6. Classifiers: Neural networks (II)

- Parameters: weights and biases
- Initial parameter values assigned randomly.
- "Optimal" parameters minimize the error function

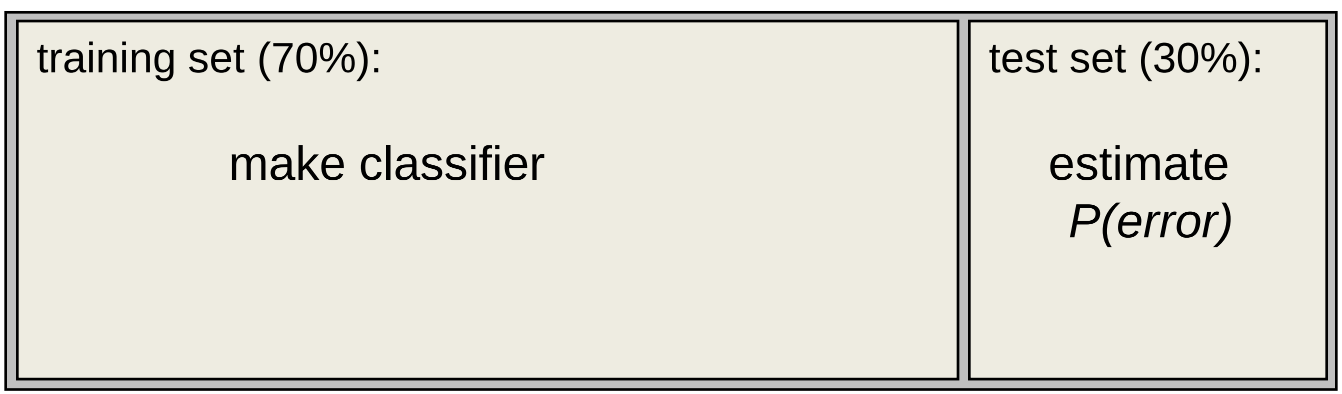
$$E = \frac{1}{2}\sum_{k=1}^{n}\left(y_k(\omega, b) - t_k\right)^2$$

$y_k(\omega,b)$ network output, t_k target

- Find optimal parameters by using back-propagation training (steepest descent algorithm)

7. Error probability estimate

- Error probability = probability of misclassifying an object.
- For estimation:
  - Split data set into two independent sets (random split): training set and test set.
  - Construct classifier on training set.
  - Estimate error probability (proportion of misclassified samples) on test set.



8. Results of Seven-class classifiers

1-Nearest Neighbour:

Sets	10-experiments										mean
train	0	0	0	0	0	0	0	0	0	0	**0**
test	0.386	0.409	0.341	0.409	0.318	0.386	0.296	0.455	0.341	0.455	**0.380**

5-Nearest Neighbour:

Sets	10-experiments										mean
train	0.250	0.279	0.308	0.298	0.327	0.356	0.289	0.231	0.289	0.327	**0.300**
test	0.477	0.431	0.568	0.432	0.477	0.341	0.568	0.636	0.477	0.523	**0.493**

Neural Network (2 hidden layers, resilient backpropagation):

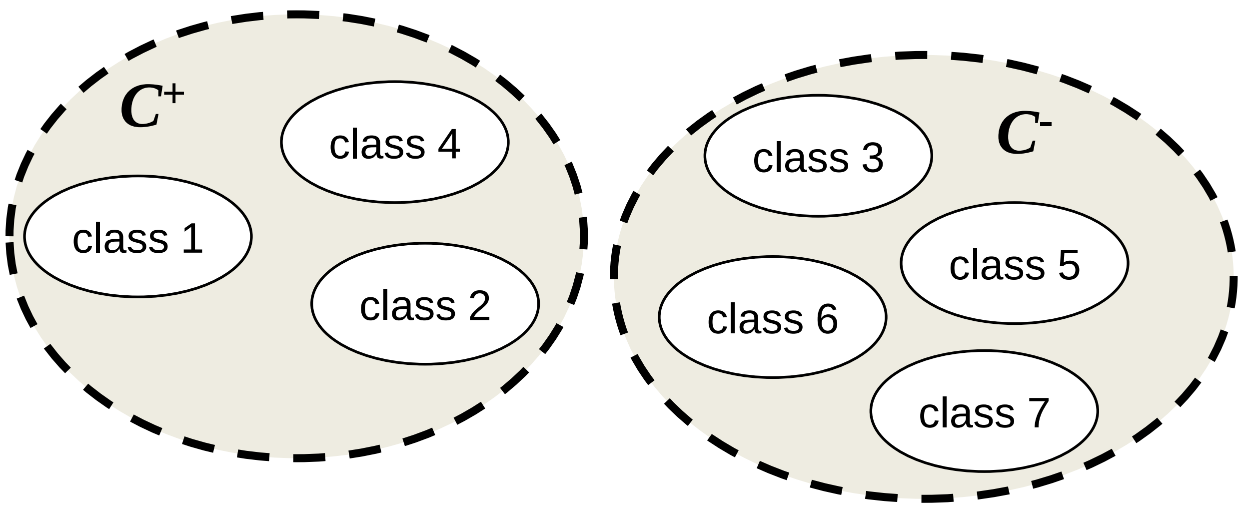
Sets											mean
train	0.058	0.058	0.048	0.077	0.058	0.077	0.077	0.058	0.077	0.077	**0.067**
test	0.318	0.273	0.273	0.250	0.205	0.273	0.250	0.205	0.273	0.341	**0.266**

Results:

- Neural Network superior
- For nearest neighbour: Increasing neighbour numbers does not lead to better classification results (small data set!)
- Seven-class prediction is **NOT POSSIBLE**!

9. Binary Classifiers

- Binary classification - classification with only two classes.
- A multiclass problem is decomposed into a set of binary classification problems by forming metaclasses C^+ and C^-.
- The binary classifiers are combined to obtain a multiclass predictor.
- Use **Error Correcting Output Codes** (ECOC) for combination.



10. Results of some binary classifiers

- 8 binary classifiers.

	f1	f2	f3	f4	f5	f6	f7	f8
C+	1	1,7	1,6	1,6,7	1,5	1,5,7	1,5,6	1,5,6,7
C-	2,3,4,5,6,7	2,3,4,5,6	2,3,4,5,7	2,3,4,5	2,3,4,6,7	2,3,4,6	2,3,4,7	2,3,4

- 10 experiments:

1-Nearest Neighbour results

	f1	f2	f3	f4	f5	f6	f7	f8
Min	0.046	0.046	0.136	0.114	0.136	0.114	0.136	0.136
Mean	**0.132**	**0.125**	**0.175**	**0.168**	**0.182**	**0.175**	**0.189**	**0.189**
max	0.182	0.182	0.205	0.250	0.250	0.250	0.227	0.300

Neural Network results (1 hidden layer with 10 hidden neurons)

	f1	f2	f3	f4	f5	f6	f7	f8
Min	0	0.023	0.046	0.091	0.046	0.046	0.023	0.068
Mean	**0.067**	**0.059**	**0.155**	**0.159**	**0.159**	**0.148**	**0.116**	**0.134**
max	0.114	0.136	0.205	0.227	0.205	0.250	0.205	0.205

Results:

- Neural network classifiers are better than Nearest neighbour classifiers.
- Misclassification rates between 6% and 19% on average.

11. Error Correcting Output Codes (I)

- Code classes: metaclass C^{+} by 1; metaclass C^{-} by 0.

  $\Rightarrow$ each classifier is represented by binary (column) vector of a code matrix M.

  Example: $(0,1,1,0,1,1,1)'$ represents $C^{+}=\{2,3,5,6,7\}$ vs. $C^{-}=\{1,4\}$

- Exhaustive code: $2^{(7\text{-}1)} - 1 = 63$ different binary classifiers.
- Each row of M represents a class and each column represents a binary classifier.

  Example: 4 classes and 7 classifiers

  $\Rightarrow$ code matrix :

$$
\begin{array}{c c}
 & \begin{array}{ccccccc} f_1 & f_2 & f_3 & f_4 & f_5 & f_6 & f_7 \end{array} \\
\begin{array}{c} class\,1 \\ class\,2 \\ class\,3 \\ class\,4 \end{array} &
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 0
\end{bmatrix}
\end{array}
$$

11. Error correcting Output Codes II

- Evaluate all binary classifiers for sample x:
  $\Rightarrow$ binary vector $\lambda = [f_1(x), f_2(x), \ldots, f_7(x)]$
- Ideally: for sample of class k, $f_i(x)=1$ if class k is in metaclass C_i^+,
  else $f_i(x)= 0$.
  $\Rightarrow$ compare λ with rows M_i of M.
- determine Hamming distances $d_H(\lambda, M_i)$, row M_i with smallest d_H wins.

Example:
4 classes
7 binary classifiers

λ	1	0	1	1	0	0	1	d_H
class 1	1	1	1	1	1	1	1	3
class 2	0	0	0	0	1	1	1	5
class 3	0	0	1	1	0	0	1	1
class 4	0	1	0	1	0	1	0	5
	f_1	f_2	f_3	f_4	f_5	f_6	f_7	

decision: class 3

- d_m = minimum Hamming distance between pair of rows of M,
  $\Rightarrow$ ECOC can correct up to $\left\lfloor \frac{d_m - 1}{2} \right\rfloor$ single bit errors.

12. Results of Error Correcting Output Codes (I)

ECOC: 1-Nearest Neighbour binary classifiers

Sets	10-experiments										mean
train	0	0	0	0	0	0	0	0	0	0	0
test	0.386	0.409	0.341	0.409	0.318	0.386	0.296	0.455	0.341	0.455	**0.380**

7-class
0.380

ECOC: 5-Nearest Neighbour binary classifiers

Sets	10-experiments										mean
train	0.269	0.289	0.307	0.307	0.337	0.365	0.289	0.240	0.269	0.365	**0.304**
test	0.500	0.432	0.568	0.432	0.523	0.386	0.546	0.636	0.523	0.545	**0.509**

7-class
0.493

ECOC: Neural Network binary classifiers

Sets	10-experiments										mean
train	0	0.001	0	0.001	0.001	0.001	0.001	0	0	0	**0.001**
test	0.136	0.114	0.227	0.227	0.250	0.136	0.159	0.136	0.159	0.159	**0.170**

7-class
0.266

Results:

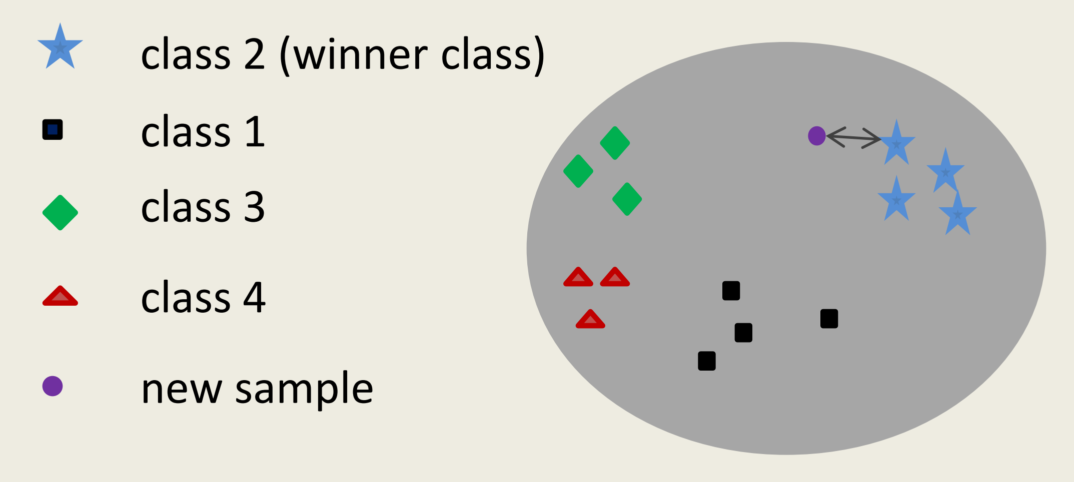
- approx. 10% improvement for Neural Network classifiers
- no improvement for Nearest Neighbour classifiers

13. ECOC and 1-Nearest neighbour binary classifiers

- $M_{ik} = 1$ if and only if class i is in metaclass C_k^+ of binary classifier f_k.
- if class i "wins" the nearest neighbour competition,
  then each binary classifier f_k that has class i in metaclass C_k^+ returns a 1
  else it returns a 0.
- Hence λ and the i-th row of code matrix M are identical.

Example:

$$\begin{array}{r c} & \begin{array}{cccc} f_1 & f_2 & f_3 & f_4 \end{array} \\ \begin{array}{r} class\,1 \\ class\,2 \\ class\,3 \\ class\,4 \end{array} & \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$



$$f_1: \quad C_1^+ = \{1\}, C_1^- = \{2,3,4\}, \quad classification: 0$$
$$f_2: \quad C_2^+ = \{1,4\}, C_2^- = \{2,3\}, \quad classification: 0$$
$$f_3: \quad C_3^+ = \{1,2\}, C_3^- = \{3,4\}, \quad classification: 1$$
$$f_4: \quad C_4^+ = \{1,2,3\}, C_4^- = \{4\}, \quad classification: 1$$

Result: No improvement for 1-NN with ECOC!

14. Results of Error Correcting Output Codes (II) (non-exhaustive: without "bad" binary classifiers)

- Idea: Delete "bad" binary classifiers ($P(\text{error}) \geq 15\%$)

Non-exh. ECOC with 1-Nearest Neighbour binary classifiers

Sets	10-experiments										mean
test	0.386	0.409	0.341	0.409	0.318	0.386	0.296	0.455	0.341	0.455	**0.380**

exh. ECOC
0.380

Non-exh. ECOC with 5-Nearest Neighbour binary classifiers

Sets	10-experiments										mean
test	0.409	0.409	0.477	0.341	0.546	0.341	0.659	0.614	0.409	0.523	**0.473**

exh. ECOC
0.509

Non-exh. ECOC with Neural Network binary classifiers

Sets	10-experiments										mean
test	0.091	0.136	0.182	0.205	0.114	0.136	0.068	0.068	0.091	0.068	**0.116**

exh. ECOC
0.17

Result:
- Further improvement (5%) when we use ECOC without "bad" classifiers!
- Note: we used the same test set to remove "bad" binary classifiers and also to test the classifier $\Rightarrow$ error probability estimate may be overly optimistic.

15. Discussion

- Classification of data is difficult because within class variability is large compared to the between class variability. This implies that classes overlap.
- 7-class classifiers (Neural Network and Nearest Neighbour) perform poorly.
- Way out :

  - use binary classifiers

  - combine the binary classifiers
- Error Correcting Output Codes (ECOC) improves classification when using Neural Networks but not for Nearest Neighbour.
- There may be improvement when using "good" binary classifiers for ECOC

15.References

[1] Michie D, Spiegelhater D.J and Taylor C.C. (1994) . Machine Learning, Neural and Statistical Classification. Ellis Horwood Limited

[2] Bishop C.M. (1995) . Neural Networks for Pattern Recognition. Oxford University Press

[3] Gordon A.D. (1995) . Classification. Chapman and Hall Limited

[4] Mitchell T.M. (1997) . Machine Learning. The McGraw-Hill Companies

[5] Riedmiller M and Braun H. (1993). *A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*. IEEE. *pg* 587

[6] Lorena A.C, de Carvalho A,C.P.L and Gama.J.M.P. (2009). A review on the combination of binary classifiers in multiclass problems. Springer science and Business Media B.V

[7] Dietterich T.G and Bakiri G.(1995). Solving Multiclass Learning Problem via Error-Correcting Output Codes. AI Access Foundation and Margan kaufmann