

Asymmetric Neighbourhood Selection and Support Aggregation for Effective Classification

Gongde Guo, Hui Wang, and David Bell

School of Information and Software Engineering, University of Ulster
Newtownabbey, BT37 0QB, N.Ireland, UK
{G.Guo, H.Wang, DA.Bell}@ulst.ac.uk

Abstract: The k -Nearest-Neighbours (k NN) is a simple but effective method for classification. The success of k NN in classification depends on the selection of a “good value” for k . To reduce the bias of k and take account of the different roles or influences that features play with respect to the decision attribute, we propose a novel asymmetric neighbourhood selection and support aggregation method in this paper. Our aim is to create a classifier less biased by k and to obtain better classification performance.

Experimental results show that the performance of our proposed method is better than k NN and is indeed less biased by k after saturation is reached. The classification accuracy of the proposed method is better than that based on symmetric neighbourhood selection method as it takes into account the different role each feature plays in the classification process.

1 Introduction

k -Nearest-Neighbours (k NN) is a non-parametric classification method which is simple but effective in many cases [Hand *et al.*, 2001]. For a data record t to be classified, its k nearest neighbours are retrieved, and this forms a *neighbourhood of t* . Majority voting among the data records in the neighbourhood is used to decide the classification for t . However, to apply k NN we need to choose an appropriate value for k , and the success of classification is very much dependent on this value. In a sense the k NN method is biased by k . There are many ways of choosing the k value, but a simple one is to run the algorithm many times with different k values and choose the one with the best performance. This is a pragmatic approach, but it lacks theoretical justification.

In order for k NN to be less dependent on the choice of k , Wang [wang, 2002] proposed to look at multiple sets of nearest neighbours rather than just one set of k nearest neighbours as we know for a data record t each neighbourhood bears certain support for different possible classes. The proposed formalism is based on probability, and the idea is to aggregate the support for various classes to give a more reliable support value, which better reveals the true class of t . However, in practice the given dataset is usually a sample of the underlying data space, and with limited computing time it is impossible to gather all the neighbourhoods to calculate the support for classifying a new data record. In a sense, the classification accuracy of the CPT method in [Wang, 2002] depends on a number of chosen neighbourhoods and this number is limited. Moreover, for most datasets in practice, features always play different roles with respect to decision attribute. Distinguishing different

influences of features on the decision attribute is a critical issue and many solutions have been developed to choose and weigh the features [Wang *et al.*, 1998, Liu *et al.*, 1998, Kononenko, 1994]. In this paper, we propose an asymmetric neighbourhood selection method based on information entropy, which takes into account the different role each feature plays to the decision attribute. Based on these specific neighbourhoods, we propose a simple aggregation method. It aggregates all the support of a set of chosen neighbourhoods to various classes for classifying a new data record in the spirit of k NN.

2 Aggregation Problem

Let Ω be a finite set called a frame of discernment. A *mass* function is $m: 2^\Omega \rightarrow [0,1]$ such that

$$\sum_{x \subseteq \Omega} m(X) = 1 \quad (2.1)$$

The mass function is interpreted as a *representation* (or *measure*) of *knowledge* or *belief* about Ω , and $m(A)$ is interpreted as a degree of support for A for $A \subseteq \Omega$ [Bell *et al.*, 1996].

To extend our knowledge to an event, A , that we cannot evaluate explicitly for m , Wang [Wang, 2002] defines a new function $G: 2^\Omega \rightarrow [0,1]$ such that for any $A \subseteq \Omega$

$$G(A) = \sum_{x \subseteq \Omega} m(X) \frac{|A \cap X|}{|X|} \quad (2.2)$$

This means that the knowledge of event A may not be known explicitly in the representation of our knowledge, but we know explicitly some events X that are related to it (i.e., A overlaps with X or $A \cap X \neq \emptyset$). Part of the knowledge about X , $m(X)$, should then be shared by A , and a measure of this part is $|A \cap X| / |X|$.

The mass function can be interpreted in different ways. In order to solve the *aggregation problem*, one interpretation is made by Wang as follows.

Let S be a finite set of class labels, and Ω be a finite dataset each element of which has a class label in S . The labelling is denoted by a function $f: \Omega \rightarrow S$ so that for $x \in \Omega$, $f(x)$ is the class label of x .

Consider a class $c \in S$. Let $N = |\Omega|$, $N_c = |\{x \in \Omega \mid f(x) = c\}|$, and $M_c = \sum_{x \subseteq \Omega} P(c \mid X)$.

The mass function for c is defined as $m_c: 2^\Omega \rightarrow [0,1]$ such that, for $A \subseteq \Omega$,

$$m_c(A) = \frac{P(c \mid A)}{\sum_{x \subseteq \Omega} P(c \mid X)} = \frac{P(c \mid A)}{M_c} \quad (2.3)$$

clearly $\sum_{x \subseteq \Omega} m_c(X) = 1$, and if the distribution over Ω is uniform, then

$M_c = \frac{N_c}{N} (2^N - 1)$. Based on the mass function, the aggregation function for c is

defined as $G_c: 2^\Omega \rightarrow [0,1]$ such that, for $A \subseteq \Omega$

$$G_c(A) = \sum_{x \subseteq \Omega} m_c(X) \frac{|A \cap X|}{|X|} \quad (2.4)$$

When A is singleton, denoted as a , equation 2.4 can be changed to equation 2.5.

$$G_c(a) = \sum_{x \in \Omega} m_c(x) \frac{|a \cap X|}{|X|} \quad (2.5)$$

If the distribution over Ω is uniform then, for $a \in \Omega$ and $c \in S$, $G_c(a)$ can be represented as equation 2.6.

$$G_c(a) = P(c|a) \mathbf{a}_c + \mathbf{b} \quad (2.6)$$

Let C_N^n be the combinatorial number representing the number of ways of picking n unordered outcomes from N possibilities, then, $\mathbf{a}_c = \frac{1}{M_c} \sum_{i=1}^N \frac{1}{i^2} (C_{N-1}^{i-1} - C_{N-2}^{i-2})$ and

$$\mathbf{b} = \frac{N_c}{M_c} \sum_{i=1}^N \frac{1}{i^2} (C_{N-2}^{i-2}).$$

Let t be a data record to be classified. If we know $P(c|t)$ for all $c \in S$ then we can assign t to the class c that has the largest $P(c|t)$. Since the given dataset is usually a sample of the underlying data space we may never know the true $P(c|t)$. All we can do is to approximate $P(c|t)$.

Equation 2.6 shows the relationship between $P(c|t)$ and $G_c(t)$, and the latter can be calculated from some given events. If the set of events is complete, i.e., 2^Ω , we can accurately calculate $G_c(t)$ and hence $P(c|t)$; otherwise if it is partial, i.e., a subset of 2^Ω , $G_c(t)$ is a approximate and so is $P(c|t)$.

From equation 2.5 we know that the more we know about a the more accurate $G_c(a)$ (and hence $P(c|a)$) will be. As a result, we can try to gather as many relevant events about a as possible. In the spirit of the k NN method we can deem the neighbourhood of a as relevant. Therefore we can take neighbourhoods of t as events. But in practice, the more neighbourhoods chosen for classification, the more computing time it will take. With limited computing time, the choice of the more relevant neighbourhoods is not trivial. This is one reason that motivated us to seek a series of more relevant neighbourhoods to aggregate the support for classification. Also in the spirit of k NN, for a data record t to be classified, the closer a tuple is to t , the more contribution the tuple donates for classifying t . Based on this understanding, to limit the number of neighbourhoods (for example, k) chosen for aggregation, we choose a series of specific neighbourhoods, which we think are relevant to a data item to be classified, for classification. Moreover, for computational simplicity, we modify equation 2.4 to equation 2.7 only keeping the core of aggregation and the spirit of k NN.

$$G_c'(t) = \frac{1}{k} \sum_{i=0}^{k-1} P(c|A_i), t \in A_i \quad (2.7)$$

Given a data record t to be classified, we choose k neighbourhoods, A_0, A_1, \dots, A_{k-1} which satisfies $t \in A_0$ and $A_0 \subset A_1 \subset \dots \subset A_{k-1}$. According to equation 2.7 we calculate $G_c'(t)$ for all classes $c \in S$, and classify t as c_i with maximal $G_{c_i}'(t)$, where $c_i \in S$.

Example 1. Given some examples with known classification $S = \{+, -\}$ shown in Figure 2.1, three neighbourhoods around t are denoted as A_0 (blank), A_1 (striped),

In practice we cannot collect all neighbourhoods to gather the support for classification, so methods to consider the contributions different features make to the decision attribute and select the more relevant neighbourhoods in the process of picking up neighbourhoods are important.

Our motivation in proposing the asymmetric neighbourhood selection method is an attempt to improve classification accuracy by selecting a given number of neighbourhoods with information for classification as possible. In this paper, we use the entropy measure of information theory in the process of neighbourhood selection. We propose a neighbourhood-expansion method by which the next neighbourhood is generated by expanding the previous one. Obviously, the previous one is covered by the later one. In each neighbourhood expansion process, we calculate the entropy of each candidate and select one with minimal entropy as our next neighbourhood. The smaller the entropy of a neighbourhood, the more unbalanced there is in the class distribution of the neighbours, and the more relevant the neighbours are to the data to be classified. More details of our algorithm are presented below.

Let C be a finite set of class labels denoted as $S=(c_1, c_2, \dots, c_m)$, and Ω be a finite dataset denoted as $\Omega=\{d_1, d_2, \dots, d_N\}$. Each element d_i in Ω denoted as $d_i=(d_{i1}, d_{i2}, \dots, d_{in})$ has a class label in S . t is a data record to be classified denoted as $t=(t_1, t_2, \dots, t_n)$. Let $N=|\Omega|$, $N_{c_i} = |\{x \in \Omega: f(x)=c_i\}|$ to all of $c_i \in S$.

Firstly, we project dataset Ω into n -dimensional space. Each data is a point in the n -dimensional space. Then we partition the n -dimensional space into a multi-grid. The partitioning algorithm of our multi-grid is described as follows:

For each dimension of n -dimensional space, if attribute a_i is ordinal, we partition $\text{dom}(a_i)=|a_{imax}-a_{imin}|$ into h equal intervals. h is an option, its value depends on concrete application domains. We use symbol Δ_i to represent the length of each grid of i^{th} attribute, in which $\Delta_i=|a_{imax}-a_{imin}|/h$. If attribute a_i is nominal, its discrete values provide a natural partitioning. At the end of the partitioning process all the data in dataset Ω are distributed into this multi-grid.

Assume A_j is the i^{th} neighbourhood and $G^i=(g_1^i, g_2^i, \dots, g_n^i)$ is the corresponding grid in n -dimensional space, for any ordinal attribute a_j , g_j^i is a interval denoted as $g_j^i=[g_{j1}^i, g_{j2}^i]$. The set of all the data covered by grid $(g_1^i, \dots, [g_{j1}^i - \Delta_j, g_{j2}^i], \dots, g_n^i)$ as well as the set of all the data covered by grid $(g_1^i, \dots, [g_{j1}^i, g_{j2}^i + \Delta_j], \dots, g_n^i)$ will be the candidates for the next neighbourhood selection. If attribute a_j is nominal, g_j^i is a set denoted as $g_j^i = \{g_{j1}^i, g_{j2}^i, \dots, g_{jq}^i\}$. For every element $x \in \text{dom}(a_j)$, where $x \notin g_j^i$, the set of all the data covered by grid $(g_1^i, \dots, g_j^i \cup \{x\}, \dots, g_n^i)$ will be the candidates for the next neighbourhood selection.

Given a set of label-known samples, the algorithm to classify a new data record t is described as follows:

Suppose that a data record $t=(t_1, t_2, \dots, t_n)$ to be classified initially falls into grid $G^0=(g_1^0, g_2^0, \dots, g_n^0)$ of n -dimensional space, i.e., $t \in G^0$. To grid G^0 , if feature t_j is

ordinal, g_j^0 represents a interval, denoted as $g_j^0=[g_{j1}^0, g_{j2}^0]$, where $g_{j1}^0=t_j-|\Delta_j|/2, g_{j2}^0=t_j+|\Delta_j|/2$. Obviously, t_j satisfies $g_{j1}^0 \leq t_j \leq g_{j2}^0$; if feature t_j is nominal, g_j^0 is a set, denoted as $g_j^0 = \{g_{jq}^0\}$, where $t_j = g_{jq}^0$. All the data covered by grid G^0 make up of a set denoted by A_0 , which is the first neighbourhood of our algorithm. The detailed neighbourhood selection and support aggregation algorithm for classification is described as follows:

1. Set $A_0 = \{d_i | d_i \in G^0\}$
2. For $i=1$ to $k-1$
 - { Find i^{th} neighbourhood A_i with minimal entropy E^i among all the candidates expanding from A_{i-1} }
3. Calculate $G_c(t) = \frac{1}{k} \sum_{i=0}^{k-1} (|A_i^c| / |A_i|)$ for all $c \in S$
4. Classify t for c that has the largest $G_c(t)$

In above algorithm, the entropy E^i is defined as follows:

$$E^i = I_{A_i}(c_1^i, c_2^i, \dots, c_m^i) \frac{|A_i|}{|\Omega|} \quad (3.1)$$

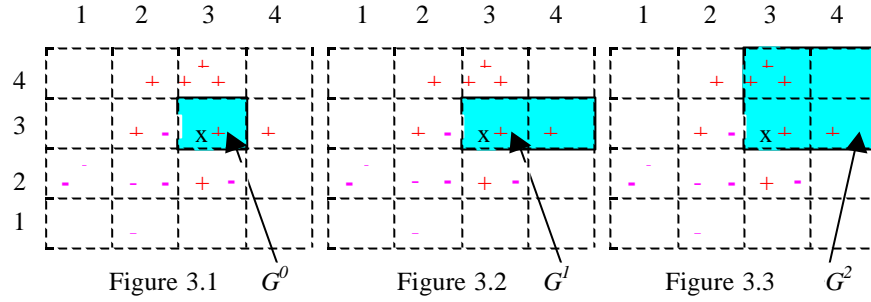
$$I_{A_i}(c_1^i, c_2^i, \dots, c_m^i) = -\sum_{j=1}^m p_j \log_2(p_j), \text{ where } p_j = \frac{|\{d_j^i | d_j^i \in A_i, f(d_j^i) = c_j^i\}|}{|A_i|} \quad (3.2)$$

Suppose that A_i and A_j are two neighbourhoods of t having the same amount of entropy, i.e., $I_{A_i}(c_1^i, c_2^i, \dots, c_m^i) = I_{A_j}(c_1^j, c_2^j, \dots, c_m^j)$, if $|A_i| < |A_j|$, we believe that A_i is more relevant to t than A_j , so in this case, we prefer to choose A_i to be our next neighbourhood. Also, if two neighbourhoods A_i and A_j of t have the same number of data tuples, we prefer to choose the one with minimal entropy as our next neighbourhood. According to equation 3.2, the smaller a neighbourhood's entropy is, the more unbalanced its class distribution is, and consequently the more information it has for classification. So, in our algorithm, we adopt equation 3.1 to be the criterion for neighbourhood selection. In each expanding process, we select a candidate with minimal E^i as our next neighbourhood.

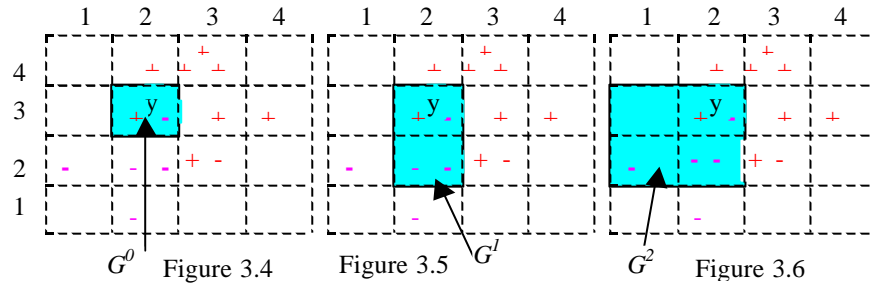
To grasp the idea here, the best way is by means of an example, so we graphically illustrate the asymmetric neighbourhood selection method here. For simplicity, we describe our asymmetric neighbourhood selection method in 2-dimensional space.

Example 2. Suppose that a data record x to be classified locates at grid [3,3] in Figure 3.1. We collect all the data, which are covered by grid [3,3] (G^0), into a set called A_0 as our first neighbourhood. Then we try to expand our neighbourhood one step in each of 4 different directions respectively (up, down, left, right) and choose a candidate having minimal E^i as our new expanded area, e.g. G^1 . Then we look up, down, left, right again and select a new area (e.g. G^2 in Figure 3.3). All the data covered by the expanded area make up of the next neighbourhood called A_i and so

on. At the end of the procedure, we obtain a series of asymmetric neighbourhoods e.g. A_2, A_3, \dots , as in Figure 3.1 to Figure 3.3.



If the data record y to be classified locates at grid [2,3] in Figure 3.4, the selection process of 3 asymmetric neighbourhoods is demonstrated by Figure 3.4 to Figure 3.6. The support aggregation method is demonstrated by **Example 1** in the previous section.



4 Evaluation and Experiment

For experimentation we used 7 public datasets available from the UC Irvine Machine Learning Repository. General information about these datasets is shown in Table 1. The datasets are relatively small but scalability is not an issue when datasets are indexed.

Table 1. General information about the datasets

Dataset	NA	NN	NO	NB	NE	CD
Iris	4	0	4	0	150	50:50:50
Wine	13	0	13	0	178	59:71:48
Hear	13	3	7	3	270	120:150
Aust	14	4	6	4	690	383:307
Diab	8	0	8	0	768	268:500
Vote	18	0	0	18	232	108:124
TTT	9	9	0	0	958	332:626

In Table 1, the meaning of the title in each column is follows: NA-Number of attributes, NN-Number of Nominal attributes, NO-Number of Ordinal attributes, NB-Number of Binary attributes, NE-Number of Examples, and CD-Class Distribution.

We used the asymmetric neighbourhood selection algorithm introduced in the previous section to select a series of neighbourhoods on 7 public datasets. The experimental results are graphically illustrated in Figure 4.1. For each value of k , $nokNN$ (we use the notation $nokNN$ in this paper to label our method) represents the average classification accuracy of aggregating k neighbourhoods' support, and kNN represents the average classification accuracy of the k^{th} neighbourhood. A comparison of asymmetric $nokNN$ and kNN is shown in Table 2.

Figure 4.1 A Comparison of Asymmetric $nokNN$ and Asymmetric kNN

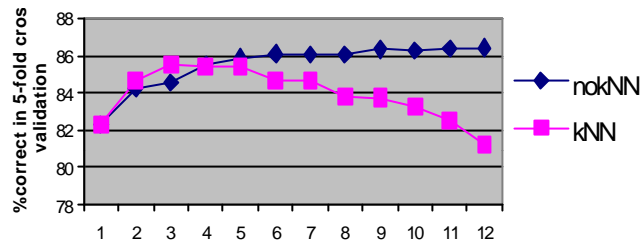
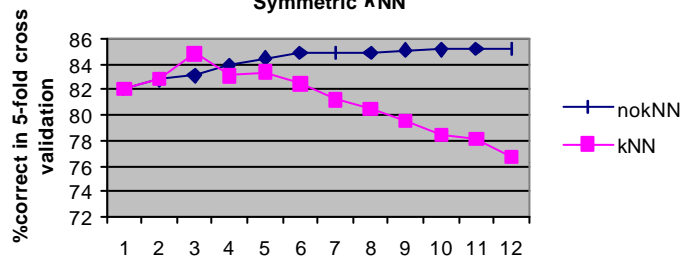


Table 2. A comparison of asymmetric kNN and asymmetric $nokNN$ in 5-fold cross validation

Dataset	Asymmetric kNN				$NokNN$
	Worst case		Best case		All of 12
	k	%correct	k	%correct	%correct
Iris	12	90.67	2	97.33	96.67
Wine	1	83.71	4	96.07	95.51
Hear	12	59.63	3	81.85	83.70
Aust	1	83.04	9	85.51	85.22
Diab	1	68.75	11	76.43	74.48
Vote	8	85.78	3	92.67	91.38
TTT	10	75.78	8	79.65	78.08
Average		78.19		87.07	86.43

From the experimental results it is clear that kNN performance varies when different neighbourhoods are used while $nokNN$ performance improves with increasing number of neighbourhoods but stabilises after a certain stage. Furthermore the performance of $nokNN$ is obviously better than that of kNN after stabilisation for each k . The experiment further shows that the stabilised performance of $nokNN$ is comparable to the best performance of kNN within 12 neighbourhoods.

Figure 4.2 A Comparison of Symmetric $nokNN$ and Symmetric kNN



To further verify our aggregation method, we also developed a symmetric neighbourhood selection algorithm, which in each neighbourhood selection process all features are expanded in the same ratio as its domain interval, seeing Figure 2.1.

Figure 4.2 and Table 3 show that similar results are obtained while using the symmetric neighbourhoods selection method.

Table 3. A comparison of symmetric k NN and symmetric no kNN in 5-fold cross validation

Dataset	Symmetric k NN				no kNN
	Worst case		Best case		All of 12
	k	%correct	K	%correct	%correct
Iris	12	74.00	1	96.67	96.67
Wine	4	91.01	3	93.82	95.51
Hear	12	55.56	3	75.93	75.93
Aust	12	78.55	3	85.07	83.91
Diab	1	68.62	3	75.52	75.00
Vote	1	85.34	5	92.67	91.38
TTT	12	75.99	9	79.12	77.97
Average		75.58		85.54	85.19

A comparison of asymmetric no kNN with symmetric no kNN in classification performance is shown in Figure 4.3 and a comparison of asymmetric k NN with symmetric k NN in classification performance is shown in Figure 4.4.

Figure 4.3 A Comparison of Asymmetric no kNN and Symmetric no kNN

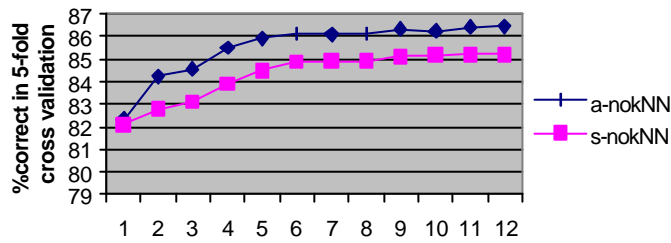
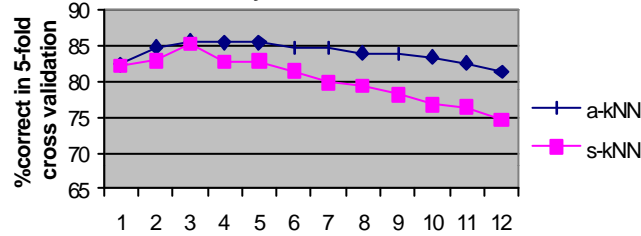


Figure 4.4 A Comparison of Asymmetric k NN and Symmetric k NN



It is obvious that the asymmetric neighbourhood selection method is better than the symmetric neighbourhood selection method for both no kNN and k NN. From the

experimental results it is clear that our hypothesis is correct – the bias of k can be removed by this method.

5 Summary and Conclusion

In this paper we have discussed the existed issues related to the k NN method for classification. In order for k NN to be less dependent on the choice of k , we looked at multiple sets of nearest neighbours rather than just one set of k nearest neighbours. A set of neighbours is called a neighbourhood. For a data record t each neighbourhood bears certain support for different possible classes. Wang addressed a novel formalism based on probability to aggregate the support for various classes to give a more reliable support value, which better reveals the true class of t . Based on [Wang, 2002] method, for specific neighbourhoods using in k NN, which always surround around the data record t to be classified, we proposed a simple aggregation method to aggregate the support for classification. We also proposed an asymmetric neighbourhood selection method based on information entropy which partitions a multidimensional data space into multi-grid and expands neighbourhoods with minimal information entropy in this multi-grid. This method is independent of ‘distance metric’ or ‘similarity metric’ and also locally takes into account the different influence of each feature on the decision attribute.

Experiments on some public datasets shows that using no k NN the classification performance (accuracy) increases as the number of neighbourhoods increases but stabilises soon after a small number of neighbourhoods; using k NN, however, the classification performance varies when different neighbourhoods are used. Experiments also show that the stabilised performance of no k NN is comparable to the best performance of k NN. The comparison of asymmetric and symmetric methods shows that our proposed asymmetric method has better classification performance as it takes into account the different influence of each feature on the decision attribute.

References

- [Bell *et al.*, 1996] Bell, D. Guan, J. and Lee, S. (1996) Generalized Union and Project Operations for Pooling Uncertain and Imprecise Information. *Data & Knowledge Engineering*. 18(1996), pp89-117.
- [Hand *et al.*, 2000] Hand, D., Mannila, H., and Smyth, P. (2001). *Principles of Data Mining*. The MIT Press.
- [Kononenko, 1994] Kononenko, I., (1994) *Estimating attributes: analysis and extensions of RELIEF*. In *Proceedings of the 1994 European Conference on Machine Learning*.
- [Liu, *et al.*, 1998] Liu, H., Motoda, H., (1998) *Feature Extraction Construction and Selection: a data mining perspective*, Kluwer Academic Publishers.
- [Wang *et al.*, 1998] Wang, H., Bell, D., and Murtagh, F. (1998). *Feature Extraction Construction and Selection a Data Mining Perspective* p85-99. Kluwer Academic Publishers.
- [Wang, 2002] Wang, H. (2002) *Nearest Neighbours without k : A Classification Formalism based on Probability*, technical report, Faculty of Informatics, University of Ulster, N.Ireland, UK.