# Using Machine Learning to Predict the Effect of Warfarin on Heart Patients

November 2004

Lara J Rennie

**Abstract**

In this study several machine learning approaches were compared to the accuracy of more traditional ways of predicting the effect of a dose of Warfarin, an anticoagulant, in heart-valve transplant patients. The twin motivations for this project derived from its potential contribution to the field of time-series machine learning, as well as the medical applications. A new 'two-layer' approach was attempted to account for the fact that the Warfarin problem consists of multiple, potentially related data-sets. Many different attribute combinations were attempted to provide the best representation of the data and any temporal patterns observed that could help with prediction. Its value in a medical sense derived from the desirability of an accurate web-based system with which self-management of patients could be facilitated. Machine learning was considered a viable solution to the difficulty of Warfarin dose prediction as machine learning algorithms endeavour to cope in a heuristic manner with problems in real data sets such as non-linearity and noise. When tested on the data of heart-valve transplant patients, it was found that the effect of a Warfarin dosage could be predicted with the most accuracy by machine learning algorithms learning on the history of multiple patients. However, the best performing algorithm and attributes differed from patient to patient, making a one-fits-all solution unlikely. The potential for machine learning solutions to out-perform physicians was demonstrated, meaning further work to increase their accuracy would be recommended in this area.

# Contents

# Chapter 1

# Introduction

Predicting the effect of the drug Warfarin, an anticoagulant, is a difficult and risky task. It is hoped that a machine learning solution may be able to improve on the current level of accuracy achieved. The field of machine learning evolved out of Artificial Intelligence in the late 1970s (Briscoe & Caelli 1996), with the first practical algorithms of ID3 (Quinlan 1986) and AQ11 (Michalski & Larson 1978). It is a branch of computer science that deals with constructing models based on known data, in order to explain its structure or variability, usually for the purpose of classifying or predicting the value of future data. The machine learning system usually produces either an equation, a set of rules or a decision tree, which can be used to explain and predict the data.

Data Mining is a closely related field, in that it can be defined as the extraction of previously unknown relationships from a data-set, often in order to predict future data values. machine learning investigates ways in which the extraction of these relationships can be automated in an efficient and accurate manner. Machine learning has an advantage over pure statistical methods, in that its different algorithms endeavour to cope with issues such as nonlinearity, noisy data and a lack of prior knowledge of the domain, albeit with varying degrees of success. Most machine learning algorithms work by following heuristics to allow them to efficiently search a set of hypotheses that may explain the data in order to find a good hypothesis in an acceptable length of time.

One approach favoured by some algorithms is that of instance-based learning, where the known example with attributes most similar to those of the given example is used to predict other information about the given example. Neural networks and genetic algorithms have also been developed, to simulate the reasoning of the human brain and the 'survival of the fittest' idea of evolution, respectively. However, the main drawback of neural networks is their inability to explain the structure of the data, even if they can often predict the value of an attribute of a given example accurately (Witten & Frank 2000). This lack of reasoning behind a particular classification can also be said of some of the results derived from instance-based learning.

There has been a growing interest in machine learning as its potential and successes in a variety of applications has become recognised. It has been successfully applied to a variety of domains, including both identifying and classifying astronomical bodies (Langley & Simon 1995) and helping epidemiologists understand the dynamics of tuberculosis epidemics (Getoor, Rhee, Koller & Small 2004). Recent research has been interested in applications of machine learning to time-series domains. Time-series domains can be defined as those applications where the history of the data can be used to predict future values. One difficulty with the data of such domains is that any attempts to model this history in a form such that it can be used to predict the future can result in data-sets with enormous numbers of attributes. Large volumes of real-world data are required to test and assess the various methods of modelling time-series data. These real world data-sets would have properties of that make machine learning difficult, namely noise, irrelevant attributes, and missing data.

The Warfarin data-set represents a time-series problem as it examines the different dosages and INR values recorded for a patient over time. It also contains all the properties mentioned

in the previous paragraph. Noise is inherent in the data-set as it is impossible to control a patient's life-style, so confounding factors, including non-compliance with their Warfarin therapy, are extremely likely. Other errors or missing data derive or arise from the fact that the data have been obtained from hand-written doctor records of a patient. Errors when performing data-input are also possible. It is therefore ideally suited as a data-set to examine time-series data with machine learning. In this problem, the history of dosages for heart-transplant patients is known, along with the corresponding International Normalised Ratio (INR), which measures the time the blood takes to clot and can be compared both between patients and across different countries. From this, it is hoped some machine learning algorithm will be able to predict either the effect of the next dosage on the INR, or the optimal next dosage for a particular INR reading, with some degree of accuracy.

## 1.1 The Importance of Correct Warfarin Prescriptions

The development of an accurate method for calculating Warfarin dosages is crucial, both because of the potential danger to the patient if the dosage is incorrect, and because of its wide use, with 13,891,000 prescriptions filled in 2003 in the United States alone (Marketos 2004). Warfarin is taken by heart-valve transplant patients as a blood-thinner, because after such operations there is a very real danger of blood clots occurring on the heart-valve. Currently 6,000 patients in the United Kingdom and 60,000 in the United States undergo heart-valve transplants every year (Bloomfield 2002). A safe Warfarin dosage range, defined as being when the INR reading is within some target range, endeavours to minimise the chance of clotting, while still ensuring the patient has enough clotting ability so that he or she does not bleed to death.

Incorrect Warfarin dosages can have a drastic effect on a patient as Warfarin has a very narrow therapeutic index. The target INR range differs depending on the type of valve replacement, and to some degree also on the patient. A target INR range of 2-3 is generally recommended, but for bileaflet mechanical valves in the aortical position, this range is usually even more precise, at 2.5-3.0. For bileaflet valves in the mitral position, a range of 3-3.5 is usually targeted, whereas for other heart-valves, an INR reading between 3.0 and 4.5 is desired (Bloomfield 2002).

However, studies show that only 50-75% of INR readings are likely to fall into the desired range for a particular patient (Gallus, Baker, Chong, Ockelford & Street 2000). Between 1.1% and 2.7% of patients managed by anti-coagulant clinics suffer major bleeding (Gallus et al. 2000). Warfarin therapy mismanagement can also have a large financial impact, shown by the case in 2002 where a Philadelphian hospital was sued for US $447,500 after the deaths of three patients were linked to overdoses of Warfarin (American Medical Association 2002).

## 1.2 Why Accurate Warfarin Dosing is Difficult

Determining the appropriate dosage of Warfarin is extremely difficult for many reasons. One of the most important reasons is the way in which the drug effect is measured. Warfarin has a half-life of 36 hours (Gallus et al. 2000), so it takes approximately 4-6 days to achieve new steady-state plasma concentrations after dose adjustments. Furthermore, the anticoagulant response monitored is only an indirect measure of the drug's actual effect, which is to reduce the ability of the body to make the prothrombin complex necessary for blood clotting (Roland & Tozer 1995). For these reasons, the maximum response to a dose is not visible for at least one or two days after ingestion, making it difficult to accurately adjust the last dosage given after a worrying INR reading.

Furthermore, there is a large variance in individual responses to the drug due to its complex pharmacology (Gage, Fihn & White 2000). The response of an individual to a particular dose is affected by many factors. These include an individual's age, weight and gender, lifestyle habits such as alcohol and tobacco consumption, and even environmental factors. Patient compliance is also an issue, as is some acquired tolerance for Warfarin (Roland & Tozer 1995).

The general health of the patient can also affect one's response to Warfarin, a factor of particular importance as most Warfarin consumers are elderly (Gallus et al. 2000). For example, acute viral hepatitis causes excessive anticoagulant response, so less Warfarin should be prescribed (Roland & Tozer 1995).

Another concern for elderly patients is that Warfarin interacts with a variety of other medications and foodstuffs. It is sensitive to vitamin K intake, and hence may be affected by foodstuffs high in this, such as green tea, lettuce and broccoli (Northwestern Memorial Hospital 2003). Any changes to the way in which Warfarin is formulated also affect the required dosage; a study performed on the generic substitution of Warfarin formulations in 1998 in the United States, where Warfarin brands were replaced by a generic formulation, discovered that apparent sensitivity to Warfarin in all patients decreased (Halkin, Shapiro, Kurnik, Loebstein, Shalev & Kokia 2003).

The genetic makeup of an individual has also been shown to be a factor. A regression model has been built taking into account the phenotype (observed nature) of the CYP2C9 gene, responsible for metabolising a wide range of drugs including Warfarin. Along with age, weight and gender, this model was able to explain up to 40% of variability in Warfarin response (Michaud, Morin, Brouillette, Roy, Verret, Noel, Taillon, O'Hara, Gossard, Champagne, Vanier & Turgeon 2004). In addition, cytochrome P450 genetic mutations have been shown to account for most of those patients who require very small dosages of Warfarin (Joffe, Xu, Johnson, Longtine, Kucer & Goldhaber 2004).

The complex nature of these factors and how they interact, along with the impossibility of accurately measuring them for a particular patient, means that accurate Warfarin dosage has proved to be very difficult. Time-series machine learning analysis could help, however, as it is hoped that a patient's history would implicitly represent those factors affecting their response to Warfarin. If such a history could be appropriately modelled, it could be used to predict the current dosage instead of explicitly attempting to model individual factors. An automated model would also be desirable if it were sufficiently accurate, in that it would allow patients to adjust their own dosage based on self-testing, from their own home, through a secure Internet server. More detail on the benefits of this is found in Chapter 3.

In Chapter 2, we first describe the current state of research into time-series domains of machine learning. Current methods for calculating Warfarin prescriptions, including any involving machine learning, along with any research into improving these, are also detailed in Chapter 2. Chapter 3 contains the aims of this research project. Chapter 4 provides an explanation of the different ways in which the patient's history will be represented, along with a rationale for and description of the various algorithms experimented with. Results from experiments on actual patient data and a discussion of these are shown in Chapter 5, followed finally by what has been learned from this research and opportunities for further research.

# Chapter 2

# Relevant Research

Relevant research with regard to this project can be divided into two main areas. Firstly, as the Warfarin data-set is an example of time-series data, one must examine how machine learning algorithms are currently being applied to time-series data in a variety of domains. There also exists a body of research that attempts to improve Warfarin dosage accuracy, through either automated or other methods. Previous attempts to apply machine learning to the Warfarin problem are discussed here.

## 2.1 Machine Learning and Time-Series Data

Time series data differ from other forms of data in that there are likely to be potentially useful relationships between different attributes, because the attributes are just instances of the same data. Some specific new algorithms have been developed in an attempt to compensate for the propensity of most machine learning algorithms to ignore these. In Kolarik & Rudorfer's (1994) study, a back-propagation neural network with one hidden layer was used to model time-series data. However, the number of hidden units to be used must be experimented with to find a number suitable to the data set. The best results were achieved by performing a logarithmic transformation on the time-series data beforehand, after which the neural network outperformed traditional regression tools. For this reason neural networks will be trialled for the Warfarin data set.

Another algorithm, developed by Geurts (2001), works by using regression trees to model piecewise functions to represent temporal signals. Geurts (2001) argues against the use of traditional machine learning algorithms for temporal data, arguing their use sacrifices interpretability of classification rules. He also argues that some temporal rules cannot easily be represented traditionally, giving the example of trying to discover a rule which asks for three consecutive values to be within a certain amount. However, it is not certain that his solution would be suitable for the Warfarin problem. Firstly, this method did not achieve better results than naïve sampling when tested on a data set with more than one attribute. The Warfarin problem fits into this category, as every data point has an INR value and a dosage, as well as existing at some point in time. Furthermore, the algorithm was not convincingly verified on real data, with two out of the three datasets on which the algorithm is tested being artificially developed for research in this context. Their method of analysing the success of the algorithm was also questionable because cross-validation was used. With cross-validation, a random subset of the data is extracted, and the rest used to train the algorithm. This will most likely be repeated several times, with a different subset of data selected for testing each time. This means that for a data-point $n$ in the testing set, it is quite likely that values of the patient history that occurred after $n$ will be used to train the algorithm that then predicts the value for $n$. In Geurts's (2001) study, the accuracy with which this $n$ can be predicted in such a way is being used as a statistical measure of the accuracy of the algorithm. When using data from a patient history to train a model, future data cannot be used to help the doctor decide

how much to prescribe at the current time. This means that the only way to measure the accuracy of a machine learning technique in this case is to look at the number of successes it achieves in predicting a data point after being trained only on a history of the data-points preceding it.

Other suggested methods for temporal data include that of Kadous (2002), who attempted to first extract 'metafeatures' from a temporal data series, and then train different machine learning algorithms on this. This is principally used, however, for finding similarities between behaviour of a certain value during two different spans of time, in order to classify such spans of time. Nevertheless, this means that if a span in time can be classified correctly, the value of individual points within this can be derived. It also attempts to use global attributes (such as gender in our domain) and aggregate features (attributes derived by combining other attributes). However, a major drawback is that again it only really looks at the value of one attribute over time, and in our domain we have two, namely INR and dosage, which are related to each other. It also does not outperform base-line methods on the two data-sets on which it was tested, and fares badly under noise.

Hidden Markov Models are the usual method for performing time-series analysis, and a novel approach combining these with providing reward notification on correct behaviour was developed by Wierstra & Wiering (2004). This was tested on datasets with more than one attribute for each data point in the past. It is designed to be suited to stochastic and noisy environments. However, these were not considered to be so useful for this domain, being useful more in domains such as speech recognition. Hidden Markov Models rely on gaining a large amount of state information from the history. Although values immediately surrounding each data-point do have some influence on the value of the current data-point, this domain does not really seem to have recurring chains of values from which further values along the chain can be predicted.

A recent survey on new methods for modelling time-series data was performed by Keogh & Kasetty (2002). This evaluated 56 papers, and considered all to be lacking in several key areas. The main problems were that they were tested on a mean of 1.85 data sets, and hence were not proved to be widely applicable, and secondly were usually only compared to one other algorithm. In our study it was hence deemed necessary to try and use multiple data sets where possible, albeit for the same domain. It must be remembered however that the primary aim is not to determine a generic time-series solution, as different algorithms are suited to different domains; instead, the best solution for the Warfarin time-series data is desired. Many different algorithms will also be examined to avoid the problems of lack of comparative information found in other studies.

## 2.2 Current Methods for Warfarin Dosage

It is important to examine current Warfarin prediction methods, firstly as they may offer insight into a machine learning solution, particularly with regard to attributes selection, and secondly as a means of comparison. Warfarin is currently prescribed in a variety of different ways. Although computer support is beginning to be recognised as useful, many clinics and physicians still use paper-based nomograms or loading tables.

### 2.2.1 Non-computerised dosage calculation

The simplest mechanism for prescribing Warfarin is a "loading table" of rules specifying what dosage is needed following a given INR reading. This, however, does not acknowledge individual differences in Warfarin response. Current advice is to adjust the dosage by 5-20% of the total weekly dose, depending on the current INR, the previous dose, and any reasons identified that might explain the undesirable current INR reading (Jaffer & Bragg 2003, Horton & Bushwick 1999). Other policies also exist for determining whether the INR is such that the dosage should be changed. These include altering dosage either after two consecutive out-of-range INR readings, or if the INR reading exceeded the target range by a given amount. An example of a loading table used by some physicians for dose adjustment is shown in Table 2.1.

| INR Goal 2.0 - 3.0 | INR Goal 2.5-3.5 | Action |
|---|---|---|
| $\geq 6.3$ | $\geq 6.3$ | Stop Warfarin, evaluate for bleeding and call physician |
| 4.5 - 6.2 | 4.5 - 6.2 | Stop Warfarin for 1-2 days, then decrease total weekly dose by 20%. Repeat INR in 1 week. |
| 3.6-4.4 | 4.1-4.4 | Decrease total weekly dose by 15-20%. Repeat INR in 1-2 weeks. |
| 3.2-3.5 | 3.7-4.0 | Decrease total weekly dose by 10-15% or maintain same dose. Repeat INR in 1-2 weeks. |
| 1.9-3.1 | 2.4-3.6 | No change. Repeat INR in 1-2 weeks. If stable patient (2 consecutive goal INRs) repeat INR in 3-6 weeks. |
| 1.3-1.8 | 1.6-2.3 | Increase total weekly dose by 10-15%. Repeat INR in 1 week. |
| $< 1.3$ | $< 1.6$ | Increase total weekly dose by 15-20%. Repeat INR in 1 week. |

Table 2.1: Warfarin dose adjustment protocol used by Clarian Health Family Practice Centre Anticoagulation Clinic (Clarian Health 2004)

Graphical "nomograms" have also been developed to help with dose adjustment, such as that by Dalere (1999) shown in Figure 2.1. This was developed by first constructing a model of Warfarin activity (using pharmacodynamic and pharmacokinetic variables), and then plotting the expected behaviour of the INR for certain values of these variables when the dosage is varied. The pharmacodynamic variables were then varied to produce other curves, to try and model individual responses to Warfarin. The nomogram is used by finding the curve on which the current INR reading and dosage lie, and following this curve until the desired INR reading is reached. At this point the desired dosage can be found by reading the matching value on the x-axis. A study was performed on 111 patients, divided into three groups depending on whether their Warfarin treatment was to be managed by this nomogram, an experienced physician or the Bayesian Regression computer programme described in Section 2.2.2. This showed the nomogram to be significantly better than other methods when the mean amount of prediction error was compared. However, this paper did not give the percentage of time during which the patient was in range, making further comparisons difficult.

In general, success rates for physicians, using either nomograms, dosage adjustment tables or their own experience, have not been particularly high. A study by Schaufele, Marciello & Burke (2000) demonstrated this by analysing 181 patients receiving Warfarin treatment over a four-month period through a rehabilitation centre. Only 38% of all INR readings were found to be within the target range. Most physicians, however, achieve a 50-75% success rate for a particular patient (Gallus et al. 2000). This is still relatively low, especially when combined with the fact that between 1.1% and 2.7% of patients managed by anti-coagulant clinics suffer major bleeding (Gallus et al. 2000).

### 2.2.2 Current Technological Support

The potential of computer support for Warfarin dosage has been recognised by many researchers. It is hoped it may improve accuracy of dosage, cut costs, decrease physician workload, increase convenience for patients and help with keeping an accurate patient medical history. Originally the potential of computers was limited to making recommendations of dosage using a formula derived by clinicians, such as in the system developed by Wilson & James (1984). Twenty years after the development of this system, the real potential for computer support is still being ignored by some,

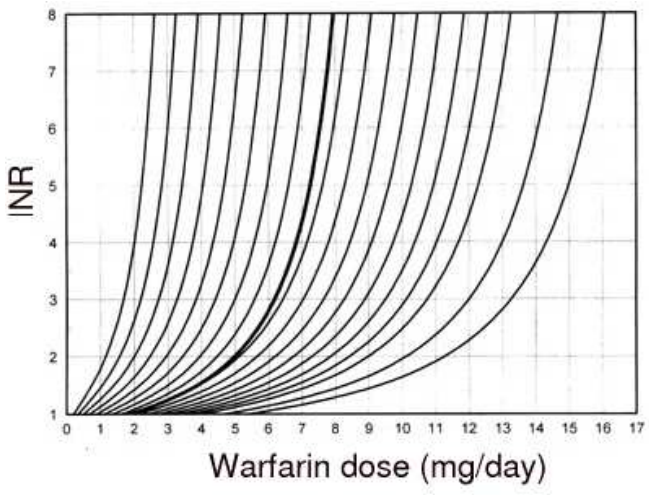


Figure 2.1: Graphical nomogram for prescribing Warfarin

with the development of similar systems simply calculating results using theoretically calculated algorithms. An example of such a system developed in 2003 was that developed by Kelly, Sweigard, Shields & Schneider (2003) for patients taking Warfarin after knee or hip replacement surgery. Their Virtual Anticoagulation Clinic was supervised by a specifically trained nurse or assistant who monitored patients and followed clinical decisions, especially important from a safety point of view. Warfarin dosage was recommended based on some theoretically calculated clinical decision support algorithms. This was achieving 'safe' INR values 70% of the time for its 1,928 patients a year after the system was first implemented. However, this system made little attempt to individualise dosage recommendations for a given INR reading. Computerised pharmacokinetic and pharmacodynamic models and simulations have also been used, achieving at least similar, if not better, accuracy than manual dosing when examining the amount of time spent in range by the patients (Vadher, Patterson & Leaning 1997, Abbrecht, O'Leary & Behrendt 1982, Ageno & Turpie 1998). The BAP-PC computer decision support system, which has been used in the United Kingdom by many primary care centers since 1998 and implements the "Coventry Model", achieved a mean of 58% of in-range INR readings (Oppenkowski, Murray, Sandhar & Fitzmaurice 2003). A survey of three currently-used computerised systems showed them to be particularly effective when patients had a higher than usual INR target range, as physicians tended to be too conservative and under-dosed their patients (Poller, Wright & Rowlands 1993).

The computational power of computers has also been exploited in several systems using Bayesian probability. One of these uses a Bayesian forecasting model to try and reach a therapeutic Warfarin level after total hip arthroplasty, and found a significant improvement in the computer-assisted group (Motykie, Mokhtee, Zebala, Caprini, Kudrna & Mungall 1999). Another computer system implemented Bayesian regression with some success (Svec, Coleman, Mungall & Ludden 1985), but when tested against the graphical nomograms discussed above in Section 2.2.1, produced worse results (Dalere 1999).

A unique addition to the literature on anticoagulant management was research by Good, Hahn, Edison & Qin (2002). This used a run-to-run control algorithm used typically for semiconductor manufacturing and achieved good results in maintaining a patient in its desired therapeutic range. However, significant input by physicians is necessary to determine whether an unexpected response to a dosage is the result of a temporary aberration or a permanent lifestyle change.

### Machine Learning Solutions

There have been some attempts to utilise the machine learning capabilities of computers in Warfarin treatment. The worth of machine learning has been proved in the prescription of other drugs, as shown in a study by Floares, Floares, Cucu, Marian & Lazar (2004), where neural networks were used successfully to compute an optimal dosage regimen for chemotherapy patients. These produced better results than the other conventional or artificial intelligence approaches reported in the study.

Mayo (2002) researched machine-learning solutions to Warfarin drug prescription, but there were some concerns with the conclusions reached. Firstly, all data came from one patient, and moreover did not take into account that this patient was growing from adolescence to adulthood. Additionally, it is never made clear whether the success rate claimed is from testing on new data, or is just an indication of how well the machine learning model fitted the training data. The attributes and machine learning parameters used were also not precisely specified.

Narayanan & Lucas (1993) also attempted a machine learning solution to predicting INR levels after a given dosage, by using a genetic algorithm to select variables with which to train a neural network. However, no comparisons were offered to other solutions, examining only the benefits of the genetic algorithm addition to the existing neural network. Neural networks have been investigated by Byrne, Cunningham, Barry, Graham, Delaney & Corrigan (2000) and found to be twice as accurate as physicians at predicting the result of a given dosage. The benefits of extracting rules for Warfarin dosage from ensemble learning have been researched by Wall, Cunningham, Walsh & Byrne (2003). It is hence hoped that these and other machine learning techniques will be able to perform well in predicting the effect of a given Warfarin dose for a particular patient in our study.

### Self-management of Warfarin

There has also been some recent research, with the development of computerised solutions to Warfarin prescription, into the benefits of letting the patient use such systems from their own home. Self-management of Warfarin, allowing INR measurements to be taken and the dosage adjusted more frequently, has been proved by Sidhu & O'Kane (2001) to result in a patient's INR reading being within the desired range 76.5% of the time. This is significantly different from the 63.8% accuracy achieved for patients managed conventionally. Many other studies have confirmed these findings, such as that by Cromheecke, Levi, Colly, de Mol, Prins, Hutten, Mak, Keyzers & Buller (2000), which showed self-management to perform at least as well as specialist management clinics, and resulted in greater patient satisfaction. Surveys of studies by Hirsh, Fuster, Ansell & Halperin (2003) and Ansell, Hirsh, Dalen, Bussey, Anderson, Poller, Jacobsen, Deykin & Matchar (2001) showed this to be a general trend.

However, in many of these self-management trials, including that by Sidhu & O'Kane (2001), only a very simple protocol was used by patients to adjust their Warfarin dose (Table 2.2. Sidhu & O'Kane (2001) also observed that not all patients could be successfully trained to manage their own therapy, and only two-thirds of those successfully trained were able to manage their own anti-coagulation for a period of two years. It is hoped a computerised system would enable more patients to self-manage their dosage regimes, and hence capitalise on the advantages of frequent measurement. It would also hopefully be more accurate than the simple protocol used here.

| INR Value Obtained | Action Taken by Patient |
|---|---|
| $< 1.5$ | Contact doctor for advice |
| $1.5 - 1.9$ | Increase dose of Warfarin by 1 mg daily |
| $2.0 - 2.5$ | Same dose of Warfarin |
| $2.6 - 4.0$ | Decrease dose of Warfarin by 1 mg daily |
| $> 4.0$ | Contact doctor for advice |

Table 2.2: Protocol used in Self-Management Patient Study (Sidhu & O'Kane 2001)

# Chapter 3

# Motivations

The main goal of this research is to find the machine learning solution that can predict the result of a given Warfarin dosage with the greatest accuracy. This involves examining different attribute selection, using different algorithms, stratification of data and a novel two-layer approach, discussed in Section 4.2.3.

It is then an aim of this study to compare this best solution with the accuracy of the physician for the patients used in this study, that of a graphical nomogram technique, and in a general sense with the accuracy reported in the literature.

Another ambition of this research is to examine the Warfarin problem in the context of time-series data. An interesting feature of the Warfarin problem is that there are multiple related data-sets available, all with more than one attribute for each data-point. Current time-series research, as detailed in the previous chapter, focusses on understanding a sole data-set, usually containing the value of only one attribute over time. With the Warfarin problem, both the reaction of a patient to previous doses as well as the reaction of other patients to Warfarin doses might help with prediction. This could be particularly important when the patient does not have a long history of taking Warfarin. In this research the effect of other patient histories on predicting the dosage for a particular patient is hence to be examined.

Furthermore, accuracy is not the only attribute of a final machine learning solution that should be considered. A crucial part of the motivation for this project is to prove the viability of a web-based system based on the results of this research. It is hence important that the solution be one that can be calculated relatively quickly. It also must not require expert input once it is running, although some monitoring would be undertaken as a safety precaution. This would improve on current patient self-management detailed in the previous chapter as it would be able to adjust dosages based on individual responses to Warfarin, and also decrease stress for the patient as they would not have to be solely responsible for deciding how to change their dosage. A study published by Walker, Machin, Baglin, Barrowcliffe, Colvin, Greaves, Ludlam, Mackie, Preston & Rose (1998) delineated some guidelines for Warfarin dosage computer systems, and should a solution derived from this research ever be implemented in a clinical trial, these would have to be followed. These specified that the computer needed to alert trained staff should patient INR levels be dangerously high or low, or should the patient stop checking their INR, and allow clinicians to override computer recommendations.

Even should the system be used in clinics in conjunction with physicians rather than as a web-based service, this would still be useful in providing recommendations to doctors, especially those less experienced with Warfarin dosage. The physician could adhere to or override this dosage if he or she felt it necessary. It would also be valuable in giving physicians more confidence to prescribe higher doses, as has been shown in previous studies involving computer support (Walton, Dovey, Harvey & Freemantle 1999).

It is to be noted that in this research we will be examining the ability of a solution to predict the effect of a given dosage of Warfarin. Although it will not therefore give a recommendation on what Warfarin dose should be prescribed, should it be sufficiently succesful at predicting the

effect of dosages, it should be relatively simple to extract a range of values that is expected to give a safe INR reading. From this, a mid-point or similar can be offered as a recommended dosage. However, recommendations of dosage are outside the scope of this research.

# Chapter 4

# Experimental Design

Some visualisation and simple regression on patient data were first undertaken to see whether the effort of a machine learning solution was worthwhile. There seemed to be low correlation between the INR levels and corresponding dosages, with a Pearson correlation factor ranging from 1% to 68% for different patients. This made it more likely that a non-linear solution would be needed.

Following this, several steps had to be followed to try to find a suitable machine learning solution.

- Decide what the machine is to be trained to predict (the 'Learning Task')
- Choose the data set and how it is to be used
- Select different machine learning algorithms for trialling
- Select the attributes to model the patient and patient history
- Run experiments
- Perform comparisons

Before full experiments were run on patient data, many machine learning solutions were briefly investigated. This involved researching possible candidate algorithms, performing small trial runs on data subsets and estimating which algorithm parameters might give the best results. The WEKA tool-bench (Witten & Frank 2000) was used as a source of implementations of all algorithms considered.

## 4.1 The Learning Task

Two options immediately presented themselves when considering the form that the output from the machine learning algorithms should take. The first of these was a numerical value of the expected INR reading after a given Warfarin dosage. However, this severely restricted the number of machine learning algorithms that can be used, because the vast majority of machine learning algorithms produce a nominal classification as their output. Since the actual value of INR is less important than its relative position to the therapeutic index, it was decided that the end result of a particular Warfarin dosage could be usefully classified as either "low", "in range" or "high". The discretisation of real values into nominal values is known as 'binning'.

## 4.2 Possible Approaches

Several approaches were proposed as possible solutions, within which different algorithms and attributes could be used. These approaches principally differ in the source of data used for training

the system as well as whether some form of ensemble learning is to be used. These vary from learning from one's own history, to learning from a vast data-set comprising everyone's history, to learning using the 'Two layer' approach detailed below in Section 4.2.3.

### 4.2.1 Learning from One's Personal History

The simplest solution possible would be predicting a patient's response to Warfarin solely by examining this patient's history of interaction with the drug. This would have the advantage that any model of the patient built by the machine learning algorithm would be individualised as much as possible for that patient. However, this solution would obviously not be ideal if the patient did not have a long history on which the algorithm could train itself. This is the most difficult time of prediction for human physicians as well, and is also a more dangerous time for the patient.

### 4.2.2 Learning from Multiple Patients' Data

Alternatively, data from all patients could be used to train an algorithm, from which predictions for an individual patient could be made. In such a system, any data point, no matter to which patient it belonged, is used by the algorithm. This would have the advantage that the algorithm could be used to predict data points for a patient when the patient's own history was very small. It would also possibly add accuracy to predictions if dealing with an INR value that was poorly represented in the history of this particular patient but had occurred more often in the general population. However, it has the disadvantage that the data from other patients may not in fact be useful when considering the patient under study.

### 4.2.3 An Ensemble Approach using Multiple Patient Data

A 'Two layer' approach has been trialled in this study as a variation on the types of ensemble learning often practised in machine learning. Ensemble learning is where the results of independent or iterative learning procedures are combined in some way to obtain a final result. In this case, ideas from two popular ensemble learning techniques, 'bagging' (Breiman 1996) and 'stacking' (Wolpert 1992), were combined. Bagging typically splits up one dataset into random subsets, and trains individual machine learning algorithms on each subset. Stacking is where the outcomes of multiple machines trained on the same data are passed through a second machine, which learns to predict a final classification based on the predictions of these machines.

In this case, individual learning algorithms were applied to the datasets of different patients. Hence each patient's history, including that of the patient currently under study, was modelled separately. The algorithms and attributes chosen for each patient were those that performed best after experiments on learning from only that patient's personal history. Following this, each patient model was given the same data-point, and their predictions of the resultant INR level for this data-point fed into the 'second-layer' algorithm. This was then trained to use these predictions as a means for predicting the real INR value for this patient. The second-layer algorithm was varied to try to improve performance, trialling most of the candidate machine learning algorithms detailed below.

A representation of this system can be found in Figure 4.1.

It must be kept in mind, however, that if one is predicting the INR value for data-point $n$ of a certain patient, only data-points which occurred before this data-point in time could be used to train the machine learning algorithm building the patient model for this patient. All data-points could be used to train the other patient models, however, as there is no temporal interaction between patients.

It was hoped that this approach would achieve a good success rate as it combined the desirable attributes of the other approaches previously outlined. It was hoped that the use of multiple models would minimise the chance of error as the 'opinions' of multiple machines were being sought before a final prediction is made. It also adapted to an individual patient, yet used the

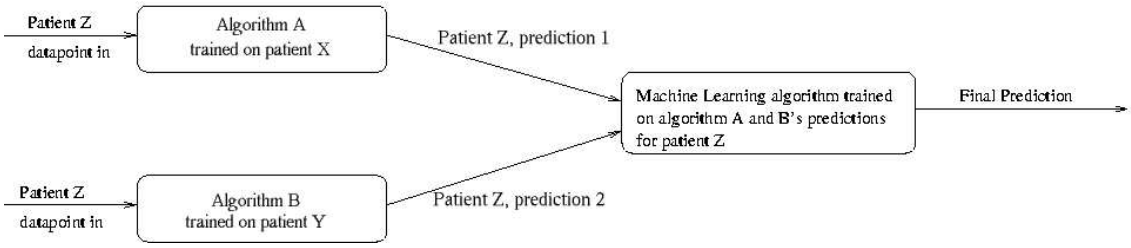


Figure 4.1: The 'Two layer' approach

data of the whole population to help it make predictions for INR values that were rare for the individual patient.

Elements of 'bagging' can be seen in that the data set of multiple histories is divided among different algorithms, but in this case the division of data-points is done in an 'intelligent' fashion, depending on to whom they belong. Similarly, the idea of a second machine, learning on the results of a first layer of machines, derives from the 'stacking' concept.

## 4.3 Candidate Machine Learning Algorithms

Most Machine Learning algorithms can be classified as being either instance-based, rule-based, neural networks, genetic algorithms, or hybrids of these approaches. Representative algorithms from each of these were trialled, along with two algorithms that are harder to classify in such a manner, namely Support Vector Machines (SVM), which is in fact based on regression, and Fuzzy Lattice Reasoning (FLR). All algorithms mentioned in previous research were tested. The machine learning algorithms below were all trialled for both the case of learning only from one's own data, and learning from multiple patients' data.

### 4.3.1 J4.8

J4.8 is the last public version available of the C4.5 top-down decision tree learner. This is a variant of ID3, which determines at each step the most predictive attribute, and splits a node based on this attribute. Each node represents a decision point over the value of some attribute. J4.8 attempts to account for noise and missing data. It also deals with numeric attributes, by determining where thresholds for decision splits should be placed. In our domain, this might see a node split according to values of the attribute "`currentDosage`", stating if it is below a certain value to follow a particular branch of the decision tree, otherwise to follow the other branch. The end result is a set of rules obtained from the decision tree, with some pruning of rules that are more complex than necessary. Rules are pruned in an attempt to combat noise, and so the rules learned do not over-represent the training set. In this way disjunctive rules can be learned, where different combinations of attributes may give the same classification.

The main parameters that can be altered for this algorithm are the confidence threshold, the minimum number of instances per leaf and the number of folds for reduced error pruning. This last variable is only relevant if reduced error pruning is to be used, instead of trying to make an error estimate based on the training data. The algorithm was trialled with the default values of 0.25 and 2 for the first two of these. Reduced-error pruning was not used since it works by dividing the data set into the number of folds for error pruning. All these subsets except one are then used for training, and the subset that is left out is used to validate the generated rules when transforming the tree into a rule set. With small patient histories, such as when we are predicting early data points, this may result in only two or three data-points being used to validate the rules. In addition, it leaves less data available to build the tree.

It was hoped this algorithm would be useful as it has been successful in many domains, and it has been designed to cope with problems such as noise. Its rule-based nature also means that it provides explanations for its decisions, something which may have given physicians more confidence in its predictions.

### 4.3.2 Non-Nested Generalised Exemplars (NNge)

This algorithm also generates rules for classification of a data set. However, it derives from instance-based approaches as it works by placing instances from the data-set into an instance space, and then tries to generalise groups of instances into rules. This algorithm was implemented firstly in its default form, but also with a limit of ten placed on the number of neighbours it will examine to attempt to generalise from. This causes the algorithm to try harder to make generalisations, but may make it more vulnerable to noise.

It was hoped that NNge would be useful as it is designed to avoid the specificity bias and hence potential over-fitting of Nearest Neighbour(IBk), as it generalises the data. It has been proved in some situations, normally those with both large and small disjuncts, to outperform J4.8 and nearest neighbour. It also reduces classification time, important in a web-based system. However, current implementations are somewhat vulnerable to noise (Martin 1995).

### 4.3.3 Nearest Neighbour (IBk)

Instance-based learners are 'lazy' classifiers, in that they delay the effort of classifying data until test instances are actually given. For every test instance, its similarity to other instances is calculated, and the classes of the $n$ most similar instances are used to predict the class of the test instance. Three different variations of IBk were used. Two of these used three neighbours for classification to try to reduce the effect of noise, which is likely to adversely affect classification accuracy if only one instance is used. In one case, the neighbours were weighted by the inverse of their 'distance' from the test instance, so that closer instances had more influence over the classification of the test instance. The third variation used this inverse weighting, but over the five nearest neighbours. However, high values of $n$ are clearly not desirable with small data-sets, as then the $n$-nearest neighbours may in fact be so different as to be completely irrelevant. The inverse weighting assists in this regard, as any neighbours that are in fact too disparate will have very low weight.

Instance-based learning is simple, independent of the order in which examples are given to the algorithm, and often works well. The success rate of this algorithm may suffer, however, for attribute values that are sparsely represented in the data-set, particularly a problem when classifying early values in a patient's history.

### 4.3.4 Ripple-Down Rule Learner (Ridor)

Ridor is another algorithm that produces a set of rules. It works by generating a default rule, after which exceptions are generated in a tree-like fashion. This has been used with success in large databases previously, such as those containing patient information used for diagnosing thyroid disorder (Gaines & Compton 1995).

### 4.3.5 Repeated Incremental Pruning to Produce Error Reduction (RIPPER)

This algorithm, proposed by Cohen (1995), was only used in the implementation of the two layer approach as a second-layer algorithm, in an attempt to find an algorithm that would increase the system's accuracy. The algorithm starts off with an empty rule-set, and 'grows' a rule by adding conditions until the rule is perfect. It then incrementally prunes rules to avoid over-fitting, optimises and sometimes deletes rules.

This algorithm was trialled since Cohen's (1995) paper showed RIPPER to be efficient on noisy datasets and to often achieve higher accuracy than C4.5, the decision tree learner on which J4.8 is based.

### 4.3.6 Genetic Algorithms

Genetic algorithms follow the basic idea of starting with a series of hypotheses, modifying each generation of hypotheses to generate new ones, while maintaining in a probabilistic fashion the most promising from the old generation, and stopping either after a certain number of generations or when the success level of a hypothesis reaches some predetermined threshold.

In the WEKA suite, genetic algorithms are only available when implementing Bayesian Network classifiers, as it allows one to find an optimal network using a genetic algorithm search. These were run twice, with varying parameters. The initial case had an initial population size of 100, with 200 descendants and ran for 50 runs with cross-over and mutation of hypotheses allowed. The 'fitness' of a particular hypothesis was judged using a Bayesian metric. A more complex version of the algorithm had an initial population size of 1000, with 2000 descendants and ran for 500 generations. This is supposed to increase the likelihood of finding the best network structure, but may in fact take too much time to be viable. However, it must be remembered that this time is only taken during training, and not classification, so this may not be too problematic. This would be true even in the case of an Internet system, as training can be performed while the patient is offline. With this network, an estimator must also be specified from the two options available, and the 'Simple Estimator' was used in both cases. These calculate the conditional probability tables for the network (Bouckaert 2004).

### 4.3.7 Neural Networks

Neural networks derive from attempts to simulate the theoretical workings of the human brain. Artificial neural networks consist of layers of perceptrons, which consider weighted inputs and based on these give a single output. Backpropagation is a valuable addition to neural networks, featuring a supervised mode of learning and a feedforward architecture (Hussain & Ishak 2004). Supervised learning means that weightings of connections are adjusted according to the accuracy with which the existing network classifies the input, so the network is trying to match the output with a known target value. The reference to a feedforward architecture simply means that output from the input layer becomes the input to the next layer, and so on. Changes to weights, determined from discrepancies between the actual and desired output, are 'backpropagated' from the output layer to the previous layer, which in turn feeds the adjustments back until the first layer is reached.

However, neural networks are typically somewhat of a 'black box' approach in that they do not help understanding of the domain to any great extent, and often find a locally optimum solution rather than a global best solution. Despite this, it is hoped that their expressive nature and ability to learn very complex nonlinear functions will help them predict Warfarin dosage accurately. They have shown promise in this domain before, as noted in Section 2.2.2.

In this study, three different neural networks were trialled. The first, and simplest, of these, had one hidden layer, whereas the second had two hidden layers. The third one had a much higher 'momentum value' than usual to try to alleviate the potential problem of reaching only a local optimal solution, rather than a global optimum.

### 4.3.8 Fuzzy Lattice Reasoning (FLR)

Fuzzy Lattice Reasoning is a branch of 'fuzzy logic', which tries to make predictions based on the premise that everything is a matter of degree. It hence tries to maintain levels of uncertainty with respect to each candidate class for the current data. Theorems show that it can, in principle, be used to model any continuous system, and it has the ability to model non-linearity (Kosoko & Isaka 1993).

However, this could not be used in those cases where nominal attributes were present because it works by defining a scale between values, so that it can assign a probability distribution with respect to a set of possible values. It hence needs numeric or ordinal values for all its attributes.

### 4.3.9 SMO

Support vector machines is a machine learning approach based on multiple regression, blending this with instance-based learning. These select a small number of critical boundary instances for each class and build functions to discriminate between the classes based on these. Non-linearity can be learned as it first applies a non-linear "kernel" function to the attributes, and then a linear solution based on these modified attributes is learned. Although initial training is often slow, classification is generally sufficiently quick.

## 4.4 Attributes Used

The choice of which attributes to include for each dosage was not a straight-forward decision. Theoretically, every previous dosage and INR reading could be used as attributes, but this is obviously not desirable in that many of these would be irrelevant and perhaps even confounding factors, computational and storage load would be heavy, and different size histories would be difficult to compare. When selecting variables it must be remembered that variables that may not add understanding in themselves can be useful if selected in combination with other variables. Furthermore, combinations of variables may be useful in decreasing the number of attributes used. Domain understanding is valuable in attribute selection (Guyon & Elisseeff 2003).

The attributes used aim to represent the patient's history of Warfarin interaction, and thereby implicitly represent those factors which influence the effect of Warfarin. Instead, partial instance memory could be used, which would see only the most recent data points provided for algorithms to learn on. However, it was decided that all data should be used, and attributes utilised as a means of indicating relevance of previous data to the current data-point.

### 4.4.1 Global Attributes

Global attributes are those attributes which are constant for all data-points for a patient. They are hence by definition only relevant for the learning approach where one machine has to use data from all patients to train on. The only global attribute provided is the gender, which is not actually known for all patients. This is specified as being either female, male or missing, and used as an attribute when learning on multiple patient histories.

### 4.4.2 Representing the Temporal Nature of the Data

Many different combinations of attributes were trialled to represent the patient's history before a given data-point. It should first be realised that the points in a patient's data set do not have the same temporal gap. However, almost all are between 3-5 weeks apart, and the Warfarin from a specific dose is virtually all exhumed from the body or processed in a week. In the interests of simplicity the differences here were hence ignored. However, a patient's history is being used to represent implicitly not only constant attributes of the patient, such as their genetic makeup, but also varying attributes such as diet and general lifestyle choices. This means that if two data points are separated by a greater amount of time, there is more chance that there has been a lifestyle change, and hence accuracy of prediction of the next data-point may be reduced.

The first type of attributes used was previous dosage and INR pairs. When learning from only one patient's data, up to three previous dosages and INR values were provided. More data was provided when working with a training set based on multiple patients, with anywhere from one to twelve previous dosages used. These extra data were provided here since the data set contained multiple patient histories. The data may not therefore all be relevant, and hence may have lowered

```
@relation Warfarin

@attribute dosage real
@attribute when real
@attribute INR {LOW, IN, HIGH}

@data

38.50,1,LOW              %most recent
35.00,1,LOW
35.00,2,IN
38.50,2,IN
35.00,3,LOW
38.50,3,HIGH
35.00,4,IN
37.30,4,HIGH
...
```

(a) This shows a data file built from data from only this patient with an explicit ranking attribute to show which data is more relevant

```
@relation Warfarin

@attribute gender {M, F, MSG}
@attribute dosage real
@attribute prvDsg1 real
@attribute prvINR1 {LOW, IN, HIGH}
@attribute prvDsg2 real
@attribute prvINR2 {LOW, IN, HIGH}
@attribute avDsgtot real
@attribute avINRtot real
@attribute INR {LOW, IN, HIGH}

@data
M,35.00,35.00,IN,34.00,LOW,32.50,2.8,LOW
F,28.00,35.00,HIGH,33.33,HIGH,30.00,3.0,IN
...
```

(b) This shows a data file built from data of multiple patients, using a combination of previous data-points and an average of a patient's history

Figure 4.2: Sample data files used for training different algorithms

accuracy. It was hence hypothesised that increasing the number of attributes would, in providing more information about a data-point, help to combat this.

When specifying a patient's dosage, the amount of Warfarin prescribed over a week was used. However, since only the total amount of Warfarin for the week was used, this meant that any variance over the amount taken each day was ignored. This may in fact be of some importance. Where the total weekly dosage varied, such as in the case of patients being prescribed differing dosages on alternate days, an average weekly dose was used.

Previous INR values could be provided as either nominal or numeric values, and both were trialled. Although the numeric value did give more information, if this did not ensure greater accuracy then the simpler approach of using a nominal value would be preferred.

Average values of INR or dosage were also used when learning off multiple patient histories, either alone or in conjunction with previous INR values and dosages. These averages were either calculated over the total history or over a number of previous data-points.

When learning from only the patient's own history, an explicit indicator of history was also tried. This was achieved by dividing the history chronologically into a given number of parts, and assigning each part a number such that the more recent dosages had a lower number. The data-point to be predicted was given a value of '1'. The idea behind this was to explicitly tell the algorithm which of the data-points upon which it was trained were more relevant, as the most recent data-points would also have a value of '1', and the largest numbers would correspond to the oldest data-points in the history. Example data files, using different attributes, are shown in Figure 4.2.

## 4.5 Evaluation Method

In this section we outline the steps undertaken for analysis of the different methods of predicting the effect of a Warfarin dosage.

### 4.5.1 Data source

The patient data used originated from the Christchurch Public Hospital, and was provided by Mr David Shaw, a cardiothoracic surgeon. Although data for over 70 patients were initially provided, not all of these could be used. Firstly, patients could be divided into two groups, namely initial-state patients (within two years of their heart-valve operation) or steady-state patients. The problem of predicting their response to Warfarin differs significantly between groups, and because of the time constraints placed on this study it was decided to restrict it to steady-state patients only. Some patients also had to be excluded because their history was of insufficient length.

Furthermore, for systems based on data from more than one patient, all patients studied had to have the same therapeutic range. This is because nominal classification was used, and hence a response to a particular Warfarin dosage was classified as low, in range or high. If the ranges were different for different patients, these would cease to have any common meaning.

Data were originally provided in the form of photocopies of hand-written doctor notes, and were manually entered into a spreadsheet before use. From this, ten patients were selected at random on which to perform experiments. Dosages recorded are those given over a week.

### 4.5.2 Experimental Process

Once the data were in a spreadsheet, the data for those patients selected for further analysis were extracted and converted into two files per patient, containing their INR history as nominal and numeric values respectively. From here, scripts were written so that the spreadsheet data could be converted to data files in the `arff` format required by the WEKA tool-bench, covering all options of attributes detailed in Section 4.4. Following this, experiments were scripted to use the candidate machine learning algorithms previously detailed in combination with the different attributes and approaches available, and the success rates for each scheme collated. Prediction of data-point values began on the seventh data point for each patient.

### 4.5.3 Comparisons Made

Because of the temporal nature of the data, for all but the 'two layer' approach, cross-validation could not be used to judge the success of a particular algorithm. Cross-validation is typically used as it trains the data set on a certain amount of the data, and then tries to predict the value of the rest of the data. However, this would result in effectively using future data to predict previous data-points. Instead, each different machine learning scheme was tested on each data point individually, using only the patient history up to this point to train the algorithm. The percentage of times that the algorithm correctly predicted the INR outcome as low, in range or high, was used to denote the success rate of a particular machine learning scheme. To make the data more meaningful, particularly in cases where one Warfarin prediction result was over-represented in the data, all success rates were compared to a 'base accuracy'. This calculated a prediction based on the most likely event if no information was known about a data-point. The reported success rate when comparing solutions can hence be represented as the gain (or loss) in percentage accuracy over the base accuracy.

#### 'Two layer' Approach

With the 'two layer' approach, however, the machine performing the final decision on the INR outcome of a Warfarin dosage is learning only from the nominal INR values given to it by its subsidiary machines. This means the temporal nature of the domain can be ignored when testing this machine, and so cross-validation is able to be used.

For the purposes of this study, a particular form of cross-validation called 'Leave-one-out' cross-validation was used. This set the number of folds to the number of training instances, so the classifier is built this number of times, each time using all but one of the data-points for training. This last data-point is then tested. This makes maximum use of the available data and removes the random nature of other cross-validation sub-sampling.

### Naïve Bayes

Naïve Bayes is a statistical measure which is usually used as a benchmark for machine learning studies. This calculates probabilistically the most likely class, based on the equation below.

$$P[H|E] = \frac{P[E|H]P[H]}{P[E]}$$

$P[H|E]$ is the probability of the hypothesis $H$ occurring given that event $E$ has occurred. Hence the probabilities of the reading being in range, low or high can be calculated based on the attribute values and what has previously occurred for these readings. However, it assumes independence of attributes, something which is not the case for this domain. This classifier was used to see if the more sophisticated machine learning algorithms provided any information gain over this simplistic method.

### The Physicians

In order to compare the results of our machine learning algorithms with those achieved by physicians, it is assumed that the physicians are aiming to achieve an in-range INR value with every Warfarin dosage prescribed. In this way, the number of times that the INR value was in fact in range can be used as the number of times that the physician correctly predicted the resultant INR value after a Warfarin dose.

### Previous Research

The graphical nomogram detailed in Dalere's (1999) study is to be used to predict the INR after a particular Warfarin dosage, and this is to be converted to a nominal value depending on its position relative to the target therapeutic index. The success rate of the graphical nomogram for each patient can hence be determined.

# Chapter 5

# System Evaluation and Discussion

## 5.1 Different Machine Learning Approaches

### 5.1.1 Learning from an Individual History Only

Results from learning in this manner showed that different algorithms and attributes suited different patient histories. Table 5.1 corresponds each patient with the most successful algorithm in predicting their history. Accuracies achieved varied from patient to patient, and a significant improvement could be noted over time for patients with long histories. For example, the accuracy for patient 31 reached a maximum of 53% over the whole data-set, but when only the accuracy over the last 12 data-points was considered, achieved 67%. The best performing algorithm and attributes was not necessarily the same for these two situations, however. The best accuracy achieved by each algorithm for each patient over the different parameters and attribute combinations trialled is reported in Appendix A.

The first question that must be asked when examining the results above is if the application of machine learning algorithms to a problem is in fact justified, or if a simpler model such as regression would not work just as well. The data in this case are non-linear, as was discussed in Chapter 4 and hence multiple regression is really the only possibility beyond machine learning algorithms. In fact, the SMO machine learning algorithm is basically an efficient way of implementing multiple regression, so its success rate can be considered approximately representative of this. SMO proved to be satisfactory for one patient. However, it is more interesting that the best machine learning algorithm and attribute method did vary between patients, suggesting that there is no 'one-fits-all' solution. This is not surprising, considering the wide variability in patient response to Warfarin. This would imply, however, that for every new patient different algorithms would have to be trialled, before one that models the patient well is chosen. This also restricts the ability of any system to accurately predict early values in a patient's history.

The rules generated by the most successful algorithms may offer insight into the reactions of individual patients to Warfarin. These get more complex as more data-points are provided on which the algorithms can learn. However, some do not make intuitive sense and may be a result of noise, such as those generated to model the data of patient 2, shown in full in Appendix B. For this patient, a rule was formed by NNge stating that if the dosage is 14.0, the previous dosage is also 14, and the previous INR reading is 3.2, then the current INR reading is predicted to be low. Surprisingly, if all the same conditions hold with the previous INR reading of 3.1, then the current INR reading is predicted to be high. Rules such as this lower the credibility of such solutions in the eyes of physicians. It should also be noted that in order for the algorithm to predict an 'in-range' reading, it must have seen at least one example of this. For patient 2, eight data-points were needed for training before the first such instance was encountered. This means that if the algorithm was being used in reverse, to predict a dosage that would give an 'in-range' reading, it would not have been able to do so until at least this point in the patient's history.

| Patient ID | Best algorithms | Most predictive attributes | Best accuracy achieved |
|---|---|---|---|
| 2 | NNge | Real values of INR and dosage for previous dose | 73% (100% over last 6 datapoints) |
| 5 | Neural network, IB3, IB5 and support machine vectors (SMO) | Algorithm dependent, varied from history grouping of different sizes for SMO, history grouping in groups of 5 for neural network, and previous dosages and INR readings (either nominal or real) for instance based learning | 82% |
| 31 | Neural Network | Current dosage only | 53% (50% overall and 75% over last 12 datapoints for SMO with previous dosage and real INR value) |
| 39 | Neural Network | Current dosage, attribute ranking data-point by grouping history in groups of 2 or 5 | 77% |
| 40 | Neural Network, SMO, fuzzy lattice reasoning (FLR) and NNge and IBk | No attributes beyond the current dose, and history grouping attribute of various sizes (all but IBk), or previous 3 dosages and real INR values (all) | 80% |
| 44 | Fuzzy Lattice Reasoning | Attribute ranking data-point by grouping history in groups of 1 or 3 | 55% (67% over last 12 datapoints) |
| 48 | IB5 | History grouped in groups of 3 | 70% |

Table 5.1: Best algorithms when learning from an individual's history

### 5.1.2 Learning from Multiple Histories

Again, the most predictive attributes and most accurate algorithm varied between patients. However, there was less variance here than for individual histories, as is seen in Table 5.2. A fuller account, showing best accuracy achieved by each algorithm for each patient over the different parameters and attribute combinations trialled, is reported in Appendix A.2. The success rate of the best solution varied between 61% and 90%. Most patients did, however, need a larger history than when trained on only their own data to enable an accurate prediction to be made. There was also more consensus on the most successful algorithms, with Ridor, NNge and a Bayesian Network formed by genetic search the most successful algorithms on more than one occasion each.

This is potentially a more useful solution, in that theoretically a patient does not need to have an extensive history to make predictions. However, this would be most useful if it was the case that the same algorithm could be applied to this same data to achieve accurate predictions for all patients. This could not really be said to be the case, although only three algorithms featured as optimal solutions for particular patients. In addition, the best results were usually obtained using six to nine datapoints, either by taking the average of them, their specific values or a combination of both. This means the patient would have to have to have a sufficiently large history before any predictions could be made.

| Patient ID | Best algorithms | Most predictive attributes | Best accuracy achieved |
|---|---|---|---|
| 2 | Ridor | Previous real INR value and dosage, and average of each over last 6 datapoints | 80% |
| 5 | Bayesian Network with Genetic Search | Previous six real INR values and dosages and the average INR and dosage | 90% |
| 31 | Ridor | Previous seven real INR values and dosages, and average of each over last 3 datapoints | 71% |
| 39 | Ridor and NNge | Previous three nominal INR values and dosages with the average of each over the last four datapoints, and previous four real INR values and dosages with the average of each over the total history respectively | 86% |
| 40 | Bayesian Network with Genetic Search | Previous four real INR values and dosages | 83% |
| 44 | NNge | Previous three doses and real INR values, and average of each over last 9 datapoints | 61 % |
| 48 | NNge | Previous real INR value and dosage, and average of each over last 9 datapoints | 84% |

Table 5.2: Best algorithms when learning from multiple patient histories

This could be improved by using an attribute, such as a nominal patient ID, to weight each data-point in the training set in some fashion so those data-points belonging to the individual that we are trying to predict for are given more importance. This may solve the problem noted that, in some cases, data from other patients may in fact not be very helpful.

### 5.1.3 "Two Layer" Approach

The first important thing to note is the choice of algorithm with which to represent each patient history in the first layer. Although in general the best performing algorithm and attributes were chosen, if for a patient with a long history the best overall accuracy was not very high, such as in the case of patient 31, the best performing solution over the last year was chosen instead.

Furthermore, in the case of patient 5, the Naïve Bayes approach was more successful than any other algorithm, and hence this was used instead of a real machine learning solution to model this patient.

Results from the two-layer approach were not as promising as initially hoped. Accuracies of the best second-layer algorithm ranged from 44% to 85% depending on the patient. Possible ways of improving it are discussed below. Many machine learning algorithms for the second layer machine were trialled, yet the best result that could be achieved for a particular patient was usually achieved by many different algorithms. This suggests that the choice of algorithm for this second machine may not be so important. This has in fact been found to be generally the case with stacking, so the second layer machine is usually chosen for simplicity (Wolpert 1992). The results achieved by each second-layer algorithm for each patient are shown in Table 5.3. Where different parameters were trialled for an algorithm, the best result achieved is recorded.

The two-layer approach was only moderately successful. In addition, it depends on individual

| Patient ID | J48 | Naïve Bayes | SMO | IBk | Neural Network | NNge | Ridor | RIPPER |
|---|---|---|---|---|---|---|---|---|
| 2 | 45% | 63% | 54% | 54% | 63% | 36% | 63% | 54% |
| 5 | 81% | 73% | 81% | 81% | 81% | 63% | 81% | 81% |
| 31 | 39% | 40% | 31% | 40% | 33% | 38% | 39% | 44% |
| 39 | 85% | 62% | 54% | 62% | 54% | 54% | 46% | 77% |
| 40 | 80% | 80% | 80% | 80% | 80% | 80% | 80% | 80% |
| 44 | 38% | 45% | 42% | 47% | 42% | 38% | 35% | 38% |
| 48 | 57% | 61% | 65% | 52% | 57% | 57% | 43% | 61% |

Table 5.3: Performance of second-layer algorithms per patient for 'two-layer' approach

models being built for each patient. This means for new patients an algorithm will be trained on their data as a component of the final solution. This may compromise the accuracy of early solutions for a new patient, with the lack of their own data to train upon, but the use of data from other patients should help compensate for this.

The choice of which algorithm to use for a first-layer machine modelling a particular patient is also an interesting one, as the one which was most accurate over the whole data series for that patient is not necessarily the optimal solution. Instead, a different solution, such as the most accurate over the past year, may be a better option. Furthermore, although a static choice of algorithm for the second-layer machine may be desirable from a simplicity point of view, this may not be the most accurate solution.

The two-layer approach could be improved by making a finite number of first layer machines. The data of new patients could then be incorporated into machines of a patient deemed sufficiently similar, either through machine learning techniques or a manual decision. One reason for the lack of accuracy of this approach might be that some of the machines predicting data for a patient were irrelevant, and hence confusing, for a particular patient. Some sort of filtering could hence be done to restrict the first-layer machines to only those that truly help, such as picking the best $n$ machines based on their accuracy over the last $k$ data-points.

In addition, to help the second layer machine make its decision, the gender of patients used to train each first layer machine could be given with the prediction of each first layer machine.

An analysis of the rules generated by the most successful second layer machine algorithms is also interesting. For patient 39, the two-layer approach was more successful than when learning on its own history, with an 85% success rate. The algorithm to achieve this result was J4.8, and the resulting tree was, surprisingly, based entirely on the predictions of the machine that learned to model the history of patient 5. The rules dictated that if machine 5 predicted it to be low, it would be in range; if it predicted it in range, it would be high; and if it predicted it high it would be in range. This does not, however, allow any predictions of a low reading. The success here may be because of the high prediction rates achieved by most of the machine learning solutions for patient five.

The 82% result for patient 5 is also less useful when it is examined more closely. Any rule-based algorithm that achieved this success rate did so by learning the rule that patient 5 is always above range. This is clearly not a useful rule in our domain, as it is possible for every patient to be brought into the therapeutic index. However, the instance-based approaches also achieved this accuracy, and they may be able to better model low and in-range doses as well. Most of the algorithms for patient 40 also came up with single rules classifying everything as above range. NNge, however, came up with potentially more sensible rules and achieved the same accuracy, as shown in Table 5.4.

## 5.2 Attribute Selection

The most helpful attributes differed depending on each machine learning algorithm, the individual patient and the scheme employed. However, it seems that when learning on multiple histories,

| Prediction | First-layer machine prediction | | | | | | |
|---|---|---|---|---|---|---|---|
| | 2 | 5 | 31 | 39 | 40 | 44 | 48 |
| HIGH | any | $\{LOW, HIGH\}$ | $\{HIGH\}$ | $\{IN, HIGH\}$ | $\{HIGH\}$ | $\{HIGH\}$ | $\{IN, HIGH\}$ |
| IN | $\{HIGH\}$ | $\{LOW\}$ | $\{HIGH\}$ | $\{LOW\}$ | $\{HIGH\}$ | $\{LOW\}$ | $\{LOW\}$ |

Table 5.4: Rules generated by NNge for patient 40, Two-layer scheme

a longer history is required, but still no longer than nine data-points, roughly equivalent to nine months for most patients. When learning on an individual's history, the use of only an attribute that ranked the data-points in terms of which were the most useful (recent) data-points was more successful than explicitly specifying previous dosages in some cases.

It must also be remembered that increasing the number of dosages required as attributes, either as a specific value or averaged together, also decreases the number of datapoints on which training and testing can take place. For example, the seventh datapoint can be predicted by a scheme based on only the four previous dosages, but not by one dependent on the eight previous dosages. This means that schemes requiring more attributes may be unfairly advantaged as their accuracy is calculated over less datapoints. In addition, these are later datapoints in the history so the scheme is learning from more data, even if it is not learning off more datapoints.

Where the choice of the best algorithm for an individual patient was uncertain, the performance of the last six to twelve was sometimes looked at. This was firstly done when the size of the history ensured that all attribute combinations would be possible for all of these datapoints, so a fair comparison over the same number of datapoints could be made. In addition, looking at the more recent datapoints shows the algorithms' most recent performances, after they have had plenty of data on which to learn. The algorithms performing well here are hence the algorithms that have been able to sufficiently model the entire patient history.

The representation of INR values was varied between real and nominal values. In most cases, it did not seem the extra information provided by real values was capitalised on by the machine learning systems for extra predictive power. Table 5.5 shows the success rates for each algorithm when trained on data-points with between one to three previous dosages for patient 48, in order to show the comparative performance of using nominal and real INR values. As previously discussed in Chapter 4, Section 4.3.8, FLR can only be performed when real INR values are used.

| Algorithm | 1 previous dose | | 2 previous doses | | 3 previous doses | |
|---|---|---|---|---|---|---|
| | Nominal | Real | Nominal | Real | Nominal | Real |
| Bayesian Network | 48% | 48% | 48% | 43% | 48% | 39% |
| Neural Network | 52% | 57% | 57% | 52% | 65% | 57% |
| SMO | 57% | 57% | 61% | 52% | 65% | 57% |
| IBk | 52% | 52% | 57% | 48% | 48% | 48% |
| NNge | 52% | 43% | 61% | 35% | 48% | 43% |
| Ridor | 43% | 39% | 48% | 43% | 43% | 57% |
| J4.8 | 43% | 43% | 43% | 43% | 43% | 48% |

Table 5.5: Success rates when using nominal or real INR values to represent own patient history (patient 48)

One might have expected that with more attributes, or previous data, given to the machine, accuracy would only increase. However, this is not the case. Although to a certain extent adding several more attributes does help, too long a history can be a hindrance. This may be because older data-points are in fact mostly irrelevant, with large lifestyle changes having occurred since. A typical example of this is shown in Table 5.6, where for patient 31 the change in accuracy rate over different numbers of previous data-points when learning on multiple histories is shown. Previous INR values here are represented as nominal values. One should also note the non-linearity in performance as the number of previous dosages used increase. This is typical when machine

learning parameters are modified because of stochastic effects.

| Algorithm | Number of previous dosages as attributes | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Bayesian Network | 41% | 45% | 45% | 49% | 44% | 45% | 39% | 41% | 46% |
| Neural Network | 42% | 44% | 40% | 41% | 43% | 39% | 41% | 45% | 29% |
| SMO | 49% | 48% | 51% | 52% | 50% | 45% | 47% | 49% | 51% |
| IBk | 42% | 47% | 44% | 45% | 42% | 37% | 39% | 46% | 37% |
| NNge | 45% | 42% | 48% | 40% | 35% | 35% | 34% | 36% | 44% |
| Ridor | 38% | 33% | 36% | 34% | 33% | 30% | 37% | 36% | 40% |
| J4.8 | 36% | 41% | 38% | 38% | 29% | 31% | 31% | 33% | 29% |

Table 5.6: Accuracy of different algorithms when increasing number of previous data-points used (learning on multiple histories, patient 31)

## 5.3 Comparative Analysis

Comparisons were made between different ways of predicting the effect of a given Warfarin dosage by examining the percentage of correct predictions made over the datapoints. However, predictions are only made from at least the 7th data-point, and on some occasions slightly later than this, if the attributes required dictate this. This allows sufficient history to be used to make predictions.

Table 5.7 shows a comparison of the best machine learning solutions with graphical nomograms, the accuracy of the physician and a Naïve Bayesian prediction. The machine learning solutions compared are the best algorithm combination when learning on the patient's own history, the best when learning on multiple histories, and the best two-layer result obtained. Further information on the rationale for a comparison with Naïve Bayes in particular was provided in Chapter 4, Section 4.5.3. The 'base accuracy' to which each solution is compared is the percentage of the most popular class over the patient's history, or the accuracy that would be achieved if the most common result was predicted in every case.

| Patient ID | Gain over Base Accuracy | | | | | |
|---|---|---|---|---|---|---|
| | Individual patient history | Multiple Patient Histories | Two Layer Approach | Graphical Nomogram | Doctor | Naïve Bayes |
| 2 | 10 | **17** | 0 | -18 | 0 | 4 |
| 5 | 0 | 8 | 0 | -19 | -64 | **9** |
| 31 | 5 | **23** | -4 | -8 | 0 | -4 |
| 39 | 23 | **32** | 31 | 7 | 0 | 15 |
| 40 | 0 | **3** | 0 | -20 | -60 | -13 |
| 44 | 13 | **19** | 5 | 3 | -9 | 12 |
| 48 | 22 | **36** | 17 | 4 | 0 | 17 |

Table 5.7: Comparison of the success in predicting the effect of Warfarin dosages of the best solution for each approach

One-way ANOVA was performed on the raw percentage success for each patient of the five different prediction methods. A significant difference was found ($F_{5,36} = 7.198, p < 0.001$ ). Post-hoc analysis using paired t-tests with a Bonferroni correction[1] was hence applied. This showed a weakly significant difference between Naïve Bayes and the machine learning solutions, apart from the case of the two layer approach from which Naïve Bayes was not significantly different. There was a weakly significant difference in accuracy between the two-layer approach,

[1] At the 5% level the Bonferroni value is calculated as 0.05/number of tests, which is 15 in this case

with a mean accuracy of 66.5%, and the nomogram, with a mean accuracy of 52.5%:$t_6 = 4.5, p = 0.0041$. More crucially, strongly significant differences were observed between the machine learning solution learning only the individual's history, achieving a mean best success rate of 70%, and the nomogram's accuracy (52%): $t_6 = 8.17, p < 0.001$. There was also a significant difference between the machine learning solutions learning on the patient's own history and that learning on multiple patients, which achieved a mean accuracy of 79%: $t_6 = 4.8, p = 0.0029$. When learning on the history of multiple patients, the best machine learning solution was significantly superior to both the nomogram ($t_6 = 11.2, p < 0.0001$) and the doctors' accuracy (41%): $t_6 = 4.94, p = 0.0026$.

Although a comparative analysis does make the machine learning approaches look promising, particularly that built on multiple patient data, some reservations must be mentioned. Most importantly, this was a study based on pre-existing data. This means that the machine learning algorithms were not having to cope with keeping the patients in range. This meant for some patients there were very few examples of a particular type. For example, patient 5 was the easiest for the machine learning algorithms to correctly predict, but this was most likely because most of the readings for that patient were high. This means the machine learning solutions, in most cases, did not learn the effect of Warfarin on this patient as such, but merely that for this patient the response to Warfarin was likely to put them above range. This is clearly not so useful in a clinical situation where a dosage is desired that will put the patient in range.

Other factors that should be remembered include that experiments were not performed on a large number of patients. Furthermore, the physician for each patient was not necessarily the same, and a lack of experience of some could result in worse performances than is typical.

### 5.3.1 Confounding factors

Some sources of noise must be considered when examining these results. However, these affect the original data and are not part of machine learning experiments performed on the data. These hence only affect overall accuracies obtained, and should not have much, if any, effect on a comparative analysis.

Such noise can be caused by dramatic dietary or other lifestyle changes, or bleeding events, illness, noncompliance, or medicine changes. Another significant problem is that the INR reading itself is not always accurate, with a standard deviation of 0.2 observed (Gage et al. 2000).

There is, however, one effect of noise that must be considered when comparing the accuracy of a physician to that of machine learning. This is that the physician may be able to more accurately specify something as a result of noise by discussion with the patient after an unusual result. Gage et al. (2000) recommend that if the INR values for a patient have been previously stable, and the latest INR value is more than 0.2 below or 0.4 above the therapeutic index, then the source of such a difference be investigated. If none can be found, then the dosage should be changed, but the next INR reading appointment made sooner than it would be otherwise. It would be desirable for any end solution system to ask a physician, after a surprising reading, if any reason for this is immediately obvious. This may enhance its accuracy, as it would be able to assign less importance to this data-point when making later predictions. An end-system in a clinic would also want to deal with rescheduling the next appointment of a patient following such an event.

It should be mentioned also that for the purposes of this study, the percentage of in-range INR readings is used for comparative purposes. However, other methods are possible. The recommendation by Azar, Deckers, Rosendaal, van Bergen, van der Meer, Jonker & Briet (1994) is that the percentage of total observation time spent in range, assuming INR changing linearly between readings, should be taken as a means of comparison. However, the INR reading may not in fact change in a linear fashion, especially in the case of steady-state patients where INR readings are relatively far apart (typically three-four weeks).

# Chapter 6

# Conclusion

In this research, machine learning systems outperformed traditional solutions with regards to predicting the effect on a given individual of a particular Warfarin dosage. This study has hence given hope to the idea that a machine learning approach to Warfarin could indeed be valuable in helping doctors to prescribe accurate Warfarin dosage regimens. Although the lack of clinical trials means that it cannot be emphatically stated that the best machine learning solution outstripped the performance of physicians, it does seem that the ability of the best solution for each patient to predict the INR reading was superior to that of the patients' physicians. It seems likely that a system could be built that could use this predictive ability to prescribe an optimum Warfarin dose. Such a system would indeed be a valuable asset, not only to physicians but also in contributing to the viability of patient self-management of their Warfarin regime. This would improve the quality of life for such patients.

Best results were obtained by the machine learning approach involving learning on data from all patients. This is also useful in that it may be able to be used on patients with a short history, as plenty of data from other patients are available. However, the precise algorithm and attributes to be used for best results varied between patients, although Ridor, NNge or a Bayesian network generated by genetic search were usually the most successful.

The 'two layer' approach, a novel contribution to this problem of multiple, hypothetically related temporal datasets, did not prove to be as successful as initially hoped. However, improvements may help these become more useful, as discussed in Chapter 5, Section 5.1.3.

Further research would be useful to try and improve the results found in this study, and clinical trials would be a necessity before any certain conclusions about the true viability of a machine learning solution to Warfarin prescription can be made.

## 6.1 Future Research

Future research could take two main directions. The first would relate to the Warfarin problem directly. Most importantly, one could investigate ways to convert the current prediction of the effect of a dosage into a recommendation of what dosage to give. One solution would be to use the midpoint of all dosage values that the solution predicts will end up in range, but this may not prove to be the best.

Further work could also look at various ways to improve accuracy, including other ensemble approaches. "Bagging" is commonly used, and sees multiple machines learning on different random subsets of the data, and then voting on the outcome of a class. This is similar to what has actually been implemented in the 'two layer' approach during the course of this research, except that bagging would see the data for each machine come from the same data-set and be randomly chosen. Bagging is effective in many domains, but usually gives only a modest improvement in accuracy. "Boosting", on the other hand, where misclassified instances are given greater weight and the machine re-trained on them, is less likely to work, but when it does it usually produces

a greater improvement than does bagging (Bauer & Kohavi 1999). However, boosting is unlikely to assist the Warfarin problem in that boosting is badly affected by noise, as it assigns more importance to unusual instances. Such instances are often 'noisy' and the extra weight thus placed on them is hence usually undesirable.

The accuracy over early data-points for a patient could also be examined in more detail, comparing the ability of machine learning solutions to predict the effect of a Warfarin dose with little history for a particular patient. This could be applied not only to the early history of steady-state patients, but also to unstable patients in their first few months of Warfarin treatment. This is particularly important, since it is difficult for physicians to accurately prescribe Warfarin at this time, as well as more dangerous for the patients.

More detailed research on other time-series machine learning methods and their application to the Warfarin problem could also be performed, especially with the case of Hidden Markov Models, which were mentioned in Section 2.1, Chapter 2. Some parameter fine-tuning to the methods implemented in this study could also be worthwhile. Genetic algorithms were also not fully explored in the context of this study, with genetic search being restricted to finding an optimal Bayesian network only.

Initial state patients would also be another interesting area to study, as they are usually harder to keep within their therapeutic index than their steady-state counterparts.

Extensions to the advice that an end-system is capable of giving could also be researched, such as how much vitamin K to prescribe to patients who are suffering a bleed.

The other main direction in which this work could be taken is that of time-series research. The algorithms and attributes useful here could be trialled in other domains. It would be interesting to apply the two-layer approach, in particular, to another domain with multiple related data-sets. Examples of this could include domains as diverse as predicting seismic activity looking at a number of different volcanoes and their activity over time, to the price of shares of the same industry over time. Other medical applications are also numerous.

## 6.2 Acknowledgements

I would firstly like to thank my supervisor, Dr Brent Martin, for his unfailing enthusiasm, support, advice and friendship. The honours class members have all been great, with special thanks to Karthik Nilakant for proof-reading my work and to James Mitchell for helping me to understand the intricacies and power of `bash` scripting. Thanks also to my family, including my aunt Dr Helen Winter, for proof-reading, listening to my seminar and helping me out whenever I needed it. A final thanks must go to the staff of Countdown who keep it open 24 hours a day, seven days a week for those students in need of refreshment, or further inspiration.

# Bibliography


Abbrecht, P. H., O'Leary, T. J. & Behrendt, D. M. (1982), 'Evaluation of a computer-assisted method for individualised anticoagulation: retrospective and prospective studies with a pharmacodynamic model', *Clinical Pharmacology and Therapeutics* **32**(1), 129–136.

Ageno, W. & Turpie, A. (1998), 'A randomized comparison of a computer-based dosing program with a manual system to monitor oral anticoagulant therapy', *Thrombosis Research* **91**(5), 237–240.

American Medical Association (2002), 'Philadelphia hospital sued over death linked to lab error', *American Medical News* .

Ansell, J., Hirsh, J., Dalen, J., Bussey, H., Anderson, D., Poller, L., Jacobsen, A., Deykin, D. & Matchar, D. (2001), 'Managing oral anticoagulant therapy', *Chest - The Cardiopulmonary and Critical Care Journal* **119**(1 Suppl), 22S–38S.

Azar, A. J., Deckers, J. W., Rosendaal, F. R., van Bergen, P. F., van der Meer, F. J., Jonker, J. J. & Briet, E. (1994), 'Assessment of therapeutic quality control in a long-term anticoagulant trial in post-myocardial infarction patients', *Thrombosis and haemostasis* **72**(3), 347–351.

Bauer, E. & Kohavi, R. (1999), 'An empirical comparison of voting classification algorithms: Bagging, boosting and variants', *Machine Learning* **36**, 105–142.

Bloomfield, P. (2002), 'Choice of heart valve prosthesis', *Heart* **87**, 583–589.

Bouckaert, R. (2004), 'Bayesian network classifiers in weka', `http://www.cs.waikato.ac.nz/~remco/weka_bn/weka_bn.html`

Breiman, L. (1996), 'Bagging predictors', *Machine Learning* **26**(2), 123–140.

Briscoe, G. & Caelli, T. (1996), *Symbolic Machine Learning*, Vol. 1 of *A Compendium of Machine Learning*, Ablex Publishing Corporation.

Byrne, S., Cunningham, P., Barry, A., Graham, I., Delaney, T. & Corrigan, O. I. (2000), Using neural nets for decision support in prescription and outcome prediction in anticoagulation drug therapy, *in* N. Lavrac & S. Miksch, eds, 'Proceedings of the Fifth Workshop on Intelligent Data Analysis in Medicine and Pharmacology', Berlin.

Clarian Health (2004), 'Protocol for warfarin adjustment', `http://www.iufammed.iupui.edu/patient_care/anticoagulation_clinic/down% loads/FPC_INR_Goal_Action.pdf`

Cohen, W. (1995), Fast effective rule induction, *in* 'Machine Learning: Proceedings of the Twelfth International Conference', pp. 115–123.

Cromheecke, M., Levi, M., Colly, L., de Mol, B. J., Prins, M. H., Hutten, B. A., Mak, R., Keyzers, K. & Buller, H. (2000), 'Oral anticoagulation self-management and management by a specialist anticoagulation clinic: a randomised cross-over comparison', *The Lancet* **356**(9224), 97–102.

Dalere, G. (1999), 'A graphic nomogram for warfarin dosage adjustment', *Pharmacotherapy* **19**(4), 461–467.

Floares, A. G., Floares, C., Cucu, M., Marian, M. & Lazar, L. (2004), 'Optimal drug dosage regimens in cancer chemotherapy with neural networks', *Journal of Clinical Oncology* **22**(14S), 2134.

Gage, B. F., Fihn, S. D. & White, R. H. (2000), 'Management and dosing of warfarin therapy', *The American Journal of Medicine* **109**(6), 481–488.

Gaines, B. & Compton, P. (1995), 'Induction of ripple-down rules applied to modelling large databases', *Journal of Intelligent Information Systems* **5**(3), 211–228.

Gallus, A., Baker, R., Chong, B., Ockelford, P. & Street, A. (2000), 'Consensus guidelines for warfarin therapy', *Medical Journal of Australia* **172**, 600–605.

Getoor, L., Rhee, J., Koller, D. & Small, P. (2004), 'Understanding tuberculosis epidemiology using structured statistical models', *Artificial Intelligence in Medicine* **30**(3).

Geurts, P. (2001), 'Pattern extraction for time series classification', *Lecture Notes in Computer Science* **2168**, 115–127.
*citeseer.ist.psu.edu/geurts01pattern.html

Good, R., Hahn, J., Edison, T. & Qin, S. J. (2002), Drug dosage adjustment via run-to-run control, *in* 'Proceedings of the American Control Conference', Anchorage, Alaska, pp. 4044–4049.

Guyon, I. & Elisseeff, A. (2003), 'An introduction to variable and feature selection', *Journal of Machine Learning Research* **3**, 1157–1182.

Halkin, H., Shapiro, J., Kurnik, D., Loebstein, R., Shalev, V. & Kokia, E. (2003), 'Increased Warfarin doses and decreased international normalised ratio response after nationwide generic switching', *Clinical Pharmacology and Theurapeutics* **74**(3), 215–221.

Hirsh, J., Fuster, V., Ansell, J. & Halperin, J. (2003), 'American Heart Association/American College of Cardiology Foundation guide to Warfarin therapy', *Journal of American College of Cardiology* **41**, 1633–1652.

Horton, J. & Bushwick, B. (1999), 'Warfarin therapy: Evolving strategies in anticoagulation', *American Academy of Family Physicians* pp. 635–648.

Hussain, W. & Ishak, W. (2004), 'Notes on neural network learning and training', `http://www.generation5.org/content/2004/NNTrLr.asp`

Jaffer, A. & Bragg, L. (2003), 'Practical tips for warfarin dosing and monitoring', *Cleveland Clinic Journal of Medicine* **70**(4), 361–371.

Joffe, H. V., Xu, R., Johnson, F. B., Longtine, J., Kucer, N. & Goldhaber, S. Z. (2004), 'Warfarin dosing and cytochrome p450 2c9 polymorphisms', *Thrombosis and Haemostasis* **91**(6), 1123–8.

Kadous, M. W. (2002), Temporal Classification: Extending the Classification Paradigm to Multivariate Time Series, PhD thesis, University of New South Wales Computer Science and Engineering.

Kelly, J., Sweigard, K., Shields, K. & Schneider, D. (2003), 'Safety, Effectiveness and Efficiency: a Web-based virtual anticoagulation clinic', *Joint Commission Journal on Quality and Safety* **29**(12), 646–651.

Keogh, E. & Kasetty, S. (2002), The need for time series data mining benchmarks: A survey and empirical demonstration, *in* 'Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', pp. 102–111.

Kolarik, T. & Rudorfer, G. (1994), Time series forecasting using neural networks, *in* 'Proceedings of the international conference on APL : the language and its applications', ACM Press, pp. 86–94.

Kosoko, B. & Isaka, S. (1993), 'Fuzzy logic', *Scientific American* .

Langley, P. & Simon, H. A. (1995), 'Applications of machine learning and rule induction', *Communications of the ACM* **38**(11), 55–64.

Marketos, M. (2004), 'The top 200 generic drugs in 2003 (by units)', *Drug Topics* **148**, 82.

Martin, B. (1995), Instance-based learning: Nearest neighbour with generalisation, Master's thesis, Department of Computer Science, University of Waikato, New Zealand.

Mayo, M. (2002), An adaptive computer-based system for the prescription of warfarin, Master's thesis, Department of Computer Science, University of Canterbury.

Michalski, R. S. & Larson, J. B. (1978), Selection of most representative training examples and incremental generation of vl1 hypothesis: The underlying methodology and the descriptions of programs esel and aq11, Technical Report 867, Department of Computer Science, University of Illinois, Urbana, Illinois.

Michaud, V., Morin, N., Brouillette, D., Roy, D., Verret, L., Noel, N., Taillon, I., O'Hara, G., Gossard, D., Champagne, M., Vanier, M. & Turgeon, J. (2004), 'Comparison of genotypic and phenotypic strategies for individualised therapy with the narrow therapeutic drug warfarin', *Clinical Pharmacology & Therapeutics* **75**(2), 60–60.

Motykie, G. D., Mokhtee, D., Zebala, L. P., Caprini, J. A., Kudrna, J. C. & Mungall, D. R. (1999), 'The use of a bayesian forecasting model in the management of warfarin therapy after total hip arthroplasty', *Arthroplasty Journal* **14**(8), 988–93.

Narayanan, M. & Lucas, S. (1993), 'A genetic algorithm to improve a neural network to predict a patient's response to warfarin', *Methods of Information in Medicine* **32**(1), 55–58.

Northwestern Memorial Hospital (2003), 'Warfarin sodium medication instructions', `http://www.nmh.org/patient_ed_pdfs/pt_ed_warfarin.pdf`

Oppenkowski, T. P., Murray, E. T., Sandhar, H. & Fitzmaurice, D. A. (2003), 'External quality assessment for warfarin dosing using computerised decision support software', *Journal of Clinical Pathology* **56**, 605–607.

Poller, L., Wright, D. & Rowlands, M. (1993), 'Prospective comparative study of computer programs used for management of warfarin', *Journal of Clinical Pathology* **46**, 299–303.

Quinlan, J. R. (1986), 'Induction of decision trees', *Machine Learning* **1**, 81–106.

Roland, M. & Tozer, T. (1995), *Clinical Pharmacokinetics: Concepts and Applications*, 3 edn, Williams and Wilkins, Philadelphia.

Schaufele, M. K., Marciello, M. A. & Burke, D. T. (2000), 'Dosing practices of physicians for anticoagulation with warfarin during inpatient rehabilitation', *American Journal of Physical Medicine and Rehabilitation* **79**(1), 69–74.

Sidhu, P. & O'Kane, H. O. (2001), 'Self-managed anticoagulation: results from a two-year prospective randomized trial with heart valve patients', *The Annals of Thoracic Surgery* **72**(5), 1523–1527.

Svec, J. M., Coleman, R. W., Mungall, D. R. & Ludden, T. M. (1985), 'Bayesian pharmacokinetic/pharmacodynamic forecasting of prothrombin response to warfarin therapy: preliminary evaluation', *Therapeutic Drug Monitoring* **7**(2), 174–180.

Vadher, B., Patterson, D. L. H. & Leaning, M. (1997), 'Evaluation of a decision support system for initiation and control of oral anticoagulation in a randomised trial', *British Medical Journal* **314**, 1252–1256.

Walker, I. D., Machin, S., Baglin, T. P., Barrowcliffe, T. W., Colvin, B. T., Greaves, M., Ludlam, C. A., Mackie, I. J., Preston, F. E. & Rose, P. E. (1998), 'Guidelines on oral anticoagulation', *British Journal of Haemotology* **101**, 374–387.

Wall, R., Cunningham, P., Walsh, P. & Byrne, S. (2003), 'Explaining the output of ensembles in medical decision support on a case by case basis', *Artificial Intelligence in Medicine* **28**(2), 191–206.

Walton, R., Dovey, S., Harvey, E. & Freemantle, N. (1999), 'Computer support for determining drug dosage: systematic review and meta-analysis', *British Medical Journal* **318**(7189), 984–990.

Wierstra, D. & Wiering, M. (2004), Utile distinction hidden markov models, *in* 'International Conference on Machine Learning', Banff, Canada.

Wilson, R. & James, A. H. (1984), 'Computer assisted management of warfarin treatment', *British Medical Journal* **289**(6442), 422–4.

Witten, I. H. & Frank, E. (2000), *Data Mining*, Morgan Kaufmann Publishers, San Francisco, California.

Wolpert, D. H. (1992), 'Stacked generalisation', *Neural Networks* **5**, 241–259.

# Appendix A

# Best Algorithm Results by Patient

## A.1 Learning on individual history only

| Patient ID | Naïve Bayes | Bayesian Network | Neural Network | SMO | IBk | FLR | NNge | Ridor | J48 |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 67% | 73% | 67% | 67% | 64% | 45% | 73% | 50% | 67% |
| 5 | 91% | 73% | 82% | 82% | 82% | 73% | 73% | 73% | 45% |
| 31 | 44% | 44% | 53% | 51% | 47% | 42% | 49% | 51% | 51% |
| 40 | 60% | 60% | 80% | 80% | 80% | 80% | 80% | 60% | 60% |
| 48 | 57% | 52% | 65% | 65% | 70% | 57% | 61% | 65% | 57% |
| 39 | 69% | 54% | 77% | 62% | 62% | 54% | 69% | 62% | 69% |
| 44 | 52% | 48% | 48% | 45% | 53% | 55% | 53% | 53% | 52% |

Attributes trialled:

- Current dosage, and a 'ranking' attribute that ranks data in terms of how recent it is, in groups of size $n$, where $n$ varied from 1 to 10
- Current dosage provided only
- Current dosage, and previous $k$ data-points where $k$ varied from 1 to 3

INR values were represented in both a nominal and real fashion.

## A.2 Learning on multiple histories

| Patient ID | Naïve Bayes | Bayesian Network | Neural Network | SMO | IBk | NNge | Ridor | J48 |
|---|---|---|---|---|---|---|---|---|
| 2 | 70% | 78% | 67% | 67% | 71% | 73% | 80% | 78% |
| 5 | 88% | 90% | 88% | 78% | 78% | 78% | 83% | 75% |
| 31 | 53% | 50% | 56% | 55% | 50% | 53% | 71% | 60% |
| 40 | 67% | 83% | 80% | 67% | 75% | 75% | 75% | 67% |
| 44 | 55% | 48% | 54% | 51% | 55% | 61% | 50% | 53% |
| 48 | 65% | 67% | 63% | 64% | 75% | 84% | 74% | 58% |
| 39 | 57% | 83% | 83% | 83% | 83% | 86% | 86% | 78% |

Attributes trialled:

- Current dosage and previous $n$ data-points where $n$ varied from 1 to 9
- Current dosage and average over $k$ data-points where $k$ varied from 1 to 9

- Current dosage and average over the total history
- Current dosage and combinations of the previous $n$ data-points with the average over $k$ datapoints

INR values were represented in both a nominal and real fashion.

# Appendix B

# Generated Rules

## B.1 Rules for best individual solution for patient 2

Note that these rules, generated by the NNge algorithm, may not cover all possible values of input, in which case the 'closest' rule is applied. History size is the number of data-points on which the algorithm has been trained.

| Size of history | Prediction | Current Dosage | Previous Dosage | Previous INR |
|---|---|---|---|---|
| 5 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| 6 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| 7 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| | IN | 21 | 14 | 1.4 |
| 8 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| | IN | 21 | $14 \leq dosage \leq 21$ | $1.4 \leq INR \leq 2.1$ |
| 9 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| | IN | 21 | $14 \leq dosage \leq 21$ | $1.4 \leq INR \leq 2.2$ |
| 10 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| | | 21 | 21 | 2.3 |
| | IN | 21 | $14 \leq dosage \leq 21$ | $1.4 \leq INR \leq 2.2$ |
| 11 | No change | | | |
| 12 | No change | | | |
| 13 | HIGH | $14 \leq dosage \leq 35$ | $28 \leq dosage \leq 35$ | $3.4 \leq INR \leq 5.8$ |
| | | 14 | 14 | 3.1 |

| | | | | |
|---|---|---|---|---|
| | LOW | 14 | 14 | 3.2 |
| | | 14 | 14 | 1.3 |
| | | 21 | 21 | $2.3 \leq INR \leq 2.5$ |
| | IN | 21 | $14 \leq dosage \leq 21$ | $1.4 \leq INR \leq 2.2$ |
| 14 | No change | | | |
| 15 | No change | | | |