

https://dspace.org/

Using optimisation techniques to granulise rough set partitions

Crossingham, Bodie

2009-01-26T08:58:40Z
http://hdl.handle.net/10539/5967

Downloaded from a DSpace repository. https://dspace.org/

Using Optimisation Techniques to Granulise Rough Set Partitions

Bodie Crossingham* and Tshilidzi Marwala†

*b.crossingham@ee.wits.ac.za
†t.marwala@ee.wits.ac.za

Abstract. This paper presents an approach to optimise rough set partition sizes using various optimisation techniques. Three optimisation techniques are implemented to perform the granularisation process, namely, genetic algorithm (GA), hill climbing (HC) and simulated annealing (SA). These optimisation methods maximise the classification accuracy of the rough sets. The proposed rough set partition method is tested on a set of demographic properties of individuals obtained from the South African antenatal survey. The three techniques are compared in terms of their computational time, accuracy and number of rules produced when applied to the Human Immunodeficiency Virus (HIV) data set. The optimised methods results are compared to a well known non-optimised discretisation method, equal-width-bin partitioning (EWB). The accuracies achieved after optimising the partitions using GA, HC and SA are 66.89%, 65.84% and 65.48% respectively, compared to the accuracy of EWB of 59.86%. In addition to rough sets providing the plausabilities of the estimated HIV status, they also provide the linguistic rules describing how the demographic parameters drive the risk of HIV.

Keywords: Bioinformatics application, HIV modelling, Evolutionary optimisation techniques, Rough set theory

1. INTRODUCTION

Rough set theory (RST) was introduced by Zdzislaw Pawlak in the early 1980s [1]. RST is a mathematical tool which deals with vagueness and uncertainty. It is of fundamental importance to artificial intelligence (AI) and cognitive science and is highly applicable to this study of performing the task of machine learning and decision analysis. The advantages of rough sets as with many other AI techniques are that they do not require rigid *a priori* assumptions on the mathematical nature of such complex relationships as do commonly used multivariate statistical techniques [2, 3]. RST is based on the assumption that the information of interest is associated with *some information* of its universe of discourse [4, 5]. The main concept of rough set theory is an indiscernibility relation (indiscernibility meaning indistinguishable from one another). Rough set theory handles inconsistent information using two approximations, namely the upper and lower approximation.

Rough sets have been used in many real-life applications, and these range from biomedical applications to fault diagnosis [6, 7]. Rough set theory is primarily limited to binary-concept, providing either *yes/no* results in decision processes or *positive/negative* in classification processes [8].

The rough sets are discretised into four partitions for the purpose of comparison. Four partitions are chosen as they provide a good balance between accuracy achieved

and computational time. The rough sets are first partitioned using the equal-width-bin (EWB) method, and an accuracy of 59.86% is achieved. After which, the partition sizes are optimised using a genetic algorithm, hill climbing and simulated annealing. Very limited work has been done on the optimisation of partition sizes. Crossingham *et al* [9] have investigated using a GA to discretise the partition size of a rough set. Chen *et al* have used a GA to optimise the number of partitions,but for a given number of partitions, the optimal technique to partition the cuts within each granule is yet to be investigated. This paper compares various optimisation techniques to determine the optimal method to use to discretise rough sets partitions.

The genetic algorithm was developed extensively by John Holland in mid 70's [10]. It is inspired by the principles of genetics and evolutionary biology, it uses techniques such as inheritance, mutation, selection, and crossover. The GA employs the principle of survival of the fittest in its search process, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (mutated) to form a new population. A new population is then used in the next iteration of the algorithm. The process terminates when either a predefined set number of generations are executed or once a satisfactory fitness level is reached. Its must be noted that if the algorithm terminates after reaching a maximum number of generations, the optimal solution may not have been reached. The fitness/evaluation function is the only part of the GA that has any knowledge about the problem.

Hill climbing is another optimisation technique used to partition the rough sets. Hill climbing works on the premise of obtaining the state of a current node, and then moving towards a state which is better than the current one. Depending on the variant of hill climbing implemented, either the closest node is chosen for its state to be evaluated, which is referred to as simple hill climbing or all successors are compared and the closest solution is chosen, which is called steepest ascent hill climbing. For the purpose of this paper, the later is chosen. The algorithm is called gradient ascent, as the objective function is maximised, where as, if the function were to be minimised, the algorithm would be gradient descent.

The final optimisation technique investigated to discretise the rough set partitions is simulated annealing. Simulated annealing (SA) was invented in 1983 by Kirkpatrick, Gelatt and Vecchi [11]. SA is an iterative procedure that continuously updates one candidate solution until a termination condition is reached [11], it is a technique that mathematically mirrors the cooling of a set of atoms to a state of minimum energy. The SA algorithm replaces the current solution by a random nearby solution, chosen with a probability that depends on the difference between the corresponding function values and a global parameter T (temperature). SA is based on the manner in which liquids freeze or metals recrystalise in the process of annealing. SA is a generalisation of a Monte Carlo method and relies on the Metropolis acceptance criterion [12]. SA is advantageous as its implementation allows the solution to move further away, and not only closer to the solution, this prevents the solution getting stuck at a local optimum rather than the desired global optimum. These techniques will be explained in more detail in section 3.

The remainder of the paper is organised into four major sections. First, RST and the computation of its accuracy is explained. Secondly the three optimisation methods im-

TABLE 1. Information table of the HIV data.

	Race	Mothers Age	Education	Gravidity	Parity	Fathers Age		HIV Status
$Obj^{(1)}$	2	19	13	1	1	22		1
$Obj^{(2)}$	3	22	6	2	1	25		1
$Obj^{(3)}$	1	35	6	1	0	33		0
.	.	.	.	.	.	.		.
$Obj^{(i)}$	2	27	9	3	2	30		0

plemented are summarised and compared. After that, the proposed methods are applied to the Human Immunodeficiency Virus (HIV) data set, and the results obtained are used for comparison, and finally a conclusion is given.

2. ROUGH SET THEORY (RST)

The main goal of rough sets is to synthesize approximations of concepts from the acquired data. Two approximations, namely the upper and lower approximation, are formed to deal with inconsistent information. These approximations along with other concepts that are fundamental to RST theory are given below.

2.1. Information table and information system

The data is represented using an information table, an example for the HIV data set for the *ith* object is given in Table 1. Each row in the information table represents a case, event or an object. Each column represents a condition attribute or variable. The *HIV Status* is the outcome (also called the *concept* or *decision attribute*) of each object.

An information system (Λ), is defined as a pair (**U**,A) where **U** is a finite set of objects called the universe and A is a non-empty finite set of attributes [13]. I.e. Λ = (**U**,A).

Every attribute $a \in A$ has a value which must be a member of a value set V_a of the attribute a.

$$a : \mathbf{U} \rightarrow V_a \tag{1}$$

Any subset B of A determines a binary relation *I(B)* on **U**, which is called an indiscernibility relation. The main concept of rough set theory is an indiscernibility relation (indiscernibility meaning indistinguishable from one another). Given an information system (Λ) and a subset $B \subseteq A$, B determines a binary relation (*B-indiscernibility relation*) *I(B)* on **U**:

$$(x,y) \in I(B) \text{ iff } a(x) = a(y) \tag{2}$$

RST offers a tool to deal with indiscernibility, the way in which it works is, for each concept/decision X, the greatest definable set containing X and the least definable set containing X are computed. These sets are the lower and upper approximation.

2.2. Lower and upper approximations

The lower approximation is defined as *the collection of cases whose equivalence classes are fully contained in the set of cases we want to approximate* [6]. The lower approximation of set X is denoted $\underline{B}X$ and mathematically it is represented as: $\underline{B}X = \{x \in \mathbf{U}: B(x) \subseteq X\}$

The upper approximation is defined as *the collection of cases whose equivalence classes are at least partially contained in the set of cases we want to approximate* [6]. The upper approximation of set X is denoted $\overline{B}X$ and is mathematically represented as: $\overline{B}X = \{x \in \mathbf{U}: B(x) \cap X \neq \emptyset\}$

It is through these lower and upper approximations that any rough set is defined. Lower and upper approximations are defined differently in literature, but it follows that a crisp set is only defined for $\overline{B}X = \underline{B}X$. It must be noted that for most cases in RST, reducts are generated to enable us to discard functionally redundant information [1]. Although reducts are one of the main advantages of RST, it is ignored for the purpose of this paper, i.e. the comparison of optimisation techniques to discretise rough set partitions.

2.3. Rough membership functions and rough set accuracy

A rough membership function is a function $\mu_X^B : U \rightarrow [0,1]$ that, when applied to object x, quantifies the degree of relative overlap between the set X and the indiscernibility set $[x]$ to which x belongs. This membership function is a measure of the plausibility of which an object x belongs to set X. This membership function is defined as:

$$\mu_X^B(x) = \frac{|[x]_B \cap X|}{|[x]_B|} \tag{3}$$

The results are illustrated using a receiver operating characteristic (ROC) curve. The ROC curve displays the relationship between sensitivity (true-positive rate) and 1-specificity (false-positive rate) across all possible threshold values that define the positivity of being infected with HIV. The area under curve (AUC) of the respective ROC curves is used as the performance criterion. The higher the AUC, the better the classification accuracy is. The AUC is the accuracy used to optimise the rough set partition sizes.

2.4. Rough set rule extraction

Rough set analysis involves generating decision rules, these rules are extracted from the approximation sets. Once the rules are extracted, they can be presented in an *if*-CONDITION(S)-*then*-DECISION format.

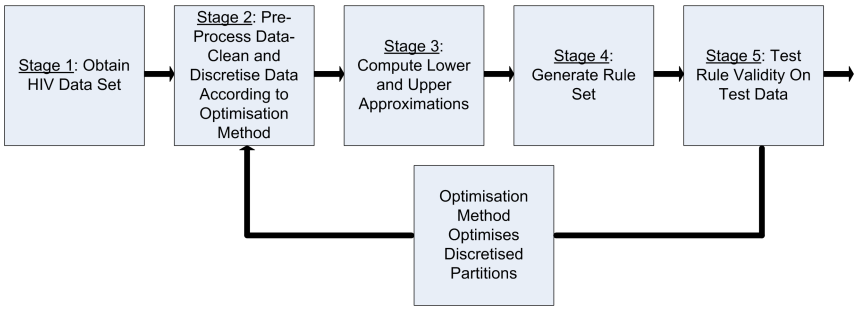


FIGURE 1. Block diagram of the sequence of events in modelling HIV

3. OPTIMISATION METHODS

The input data into the rough set is discretised into four partitions. Various approaches to perform this granularisation are; equal width intervals, equal frequency intervals and Ent-MDLPC [14]. This paper proposes an approach to determine the sizes of the partitions by optimising the cuts of the four chosen intervals or bins. Figure 1 illustrates how the optimisation techniques work in conjunction with the rough set. It must be noted; the choice of a particular optimisation technique is dependent upon the application, different optimisation techniques are better suited to some problems yet, for other applications it may not be.

3.1. Genetic algorithms (GA)

Genetic algorithms (GAs) are population based search methods. GAs are popular and widely used due to their ease of implementation, intuitiveness and the their ability to solve highly nonlinear optimisation problems. A GA is a stochastic search procedure for combinatorial optimisation problems based on the mechanism of natural selection [15]. Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover[1]. The fitness function measures the quality of the represented solution. In this case the fitness function is to maximise the accuracy of the rough set. The pseudo-code for the genetic algorithm is given below;

1. Initialise a population of chromosomes
2. Evaluate each chromosome (individual) in the population
   (a) Create new chromosomes by mating chromosomes in the current population (using crossover and mutation)
   (b) Delete members of the existing population to make way for the new members
   (c) Evaluate the new members and insert them into the population

[1] For a complete explanation of GAs, refer to [16]

3. Repeat stage 2 until some termination condition is reached, in this case until 100 generations are reached.
4. Return the best chromosome as the solution

For the particular test set as explained in section 4, the best results are obtained by implementing tournament selection, a boundary mutation and uniform crossover. An initial population of 20 members is selected and the termination function is 100 generations, at which point the final solution is the individual with the highest fitness value. In order to prevent premature convergence to a local optimum, the mutation diversification mechanism is implemented. Other mechanisms such as elitism can also be implemented in an attempt to improve the accuracy. The drawback of using GAs is that they are much slower than most traditional methods, i.e. a good initial guess will allow traditional optimisation techniques to converge quickly towards the solution. Although GAs will always approximate a solution, it must be noted, due to their stochastic nature, this approximation is only an estimate, whereas with traditional methods, if they can find an optimum, they will find it exactly [10].

3.2. Hill climbing (HC)

Hill climbing is an optimisation technique that belongs to the group of local search algorithms, meaning the algorithm moves from solution to solution in the search space until an optimal solution is found. The algorithm tries to maximise the fitness function (accuracy) by iteratively comparing two solutions, it adopts the best solution and continues the comparison (i.e. move further up the hill). This iteration terminates when there are no better solutions on either side of the current solution (i.e. it has reached the peak). There are several variants or methods of hill climbing, the first and most basic form is simple hill climbing, in this method the first closest node is chosen for evaluation. A second variant is called steepest ascent hill climbing, in this method all successors are compared and the closest to the solution is chosen. Other variants that can be investigated include next-ascent hill climbing and zero-temperature Monte Carlo hill climbing [17]. In this paper, steepest ascent hill climbing is implemented, a major disadvantage of hill climbing is that it only finds the local optimum. There are other local search algorithms that overcome this and these include: stochastic hill climbing, random walks and iterated hill-climbing. The advantages of hill climbing are that it requires no memory as there is no backtracking and it is trivial to implement.

3.3. Simulated annealing (SA)

Simulated annealing is an algorithm that locates a good approximation to the global optimum of a given function. It originated as a generalisation to the Monte Carlo method and relies on the Metropolis algorithm[2]. As is the case with GA and HC, SA continu-

[2] See [12] for more information regarding the Metropolis algorithm

ously updates the solution until a termination criteria is reached. SA is a well established stochastic technique originally developed to model the natural process of crystallisation and later adopted as an optimisation technique [12]. The SA algorithm replaces a current solution with a "nearby" random solution with a probability that depends on the difference between the corresponding function values and the temperature (T). T decreases throughout the process, so as T starts approaching zero, there is less random changes in the solution. As with the case of greedy search methods, SA keeps moving towards the best solution, except that it has the advantage of reversal in fitness. That means it can move to a solution with worse fitness than it currently has, but the advantage of that is that it ensures the solution is not found at a local optimum, but rather a global optimum. This is the major advantage that SA has over most other methods, but once again its drawback is its computational time, the SA algorithm will find the global optimum if specified but it can approach infinite time in doing so. The probability of accepting the reversal is given by Botlzman's equation [18]:

$$P(\Delta E) \propto e^{-\frac{\Delta E}{T}} \tag{4}$$

Where ΔE is the difference in energy (fitness) between the old and new states, and T is the temperature of the system. The rate at which temperature decreases depends on the cooling schedule chosen. The following cooling model is used [18]:

$$T(k) = \frac{T(k-1)}{1+\sigma} \tag{5}$$

Where $T(k)$ is the current temperature, $T(k-1)$ is the previous temperature, and σ dictates the cooling rate. It must be noted, the precision of the numbers used in the implementation of SAs can have a significant effect on the outcome. A method to improve the computational time of SA is to implement either very fast simulated re-annealing (VFSR) or adaptive simulated annealing (ASA) [19].

4. EXPERIMENTAL INVESTIGATION ON HIV DATA SET

Human Immunodeficiency Virus (HIV) is well known for being the cause that leads to the development of Acquired Immunodeficiency Syndrome (AIDS). In the last 20 years, over 60 million people have been infected with HIV, and of those cases, 95% are in developing countries [20]. In 2006 alone, an estimated 39.5 million people around the world were living with HIV, with 27.5 million of those people living in Sub-Saharan Africa. During last year, AIDS claimed an estimated 2.9 million lives [21]. Because AIDS is killing people in the prime of their working and parenting lives, it represents a grave threat to economic development. In the worst affected countries, the epidemic has already reversed many of the development achievements of the past generation [21]. The proposed method is tested on demographic data obtained from the South African antenatal sero-prevalence survey of 2001. As with many surveys, there is missing and/or incorrect data. This data needs to be cleaned and irregularities removed before any processing can be performed on it. The first irregularity would be the case of missing data, the second being false information. Such an instance of false information would

be if gravidity was zero and parity was at least one. Gravidity is defined as the number of times that a woman has been pregnant, and parity is defined as the number of times that she has given birth. Therefore it is impossible for a woman to have given birth, given she has not been pregnant. Once these irregularities were removed from the data set, only 12945 cases remained from the initial total of 13087. Of the 12945 cases, the data sets were balanced and then split into training and testing data using the ratio of 70% to 30% respectively. The performance of the rough set model was validated using the testing data. The six demographic variables considered are: *race, age of mother, education, gravidity, parity* and, *age of father*, with the outcome or decision being either HIV positive or negative[3].

Before the optimisation techniques are applied to the rough set, the accuracy of the rough set is computed using EWB partitioning and the resulting accuracy is 59.86%. The GA is run with tournament selection, a boundary mutation and uniform crossover. An initial population of 20 members is selected and the termination function is 100 generations. HC is implemented with a steepest ascent algorithm. SA has the cooling model as given in 5, it is run with a random generator with the bounds of the maximum and minimum input values, an initial temperature of 1, a stopping temperature of $1e^{-8}$. The SA algorithm has a maximum number of consecutive rejections of 200 and a maximum number of successes within one temperature set to 10. The results for each method are stated below, and receiver operation characteristic (ROC) curves and confusion matrices are given. The ROC curves for each method are displayed in Fig. 2:

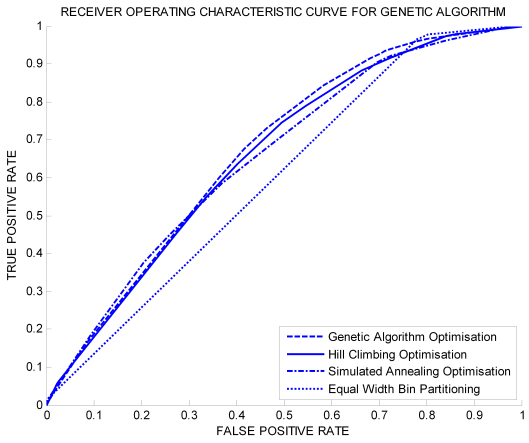


FIGURE 2. Receiver operating characteristic (ROC) curves for each method

The ROC curve is chosen to illustrate the results as they allow the performance of the classifiers to be evaluated on how well they predict. The area under curve (AUC) of the respective ROC curves is used as the performance criterion. The higher the AUC, the better the classification accuracy is. Tabulated in Table 2 is the performance of the three optimisers.

The results indicate that for the particular HIV data set, partitioning the data using

[3] For detailed information on the demographic data used refer to [22]

TABLE 2. Results obtained for each optimiser

	Number of Rules	Computational Time (min)	AUC
Genetic Algorithm	234	1255	0.6689
Hill Climbing	132	3393	0.6584
Simulated Annealing	137	3551	0.6548
Equal Width Bin	41	2	0.5986

TABLE 3. Confusion matrix of results

Genetic Algorithm			**Hill Climbing**			**Simulated Annealing**			**Equal Width Bin**		
	PP	PN		PP	PN		PP	PN		PP	PN
AP	658	168	AP	607	209	AP	535	287	AP	789	39
AN	436	365	AN	409	411	AN	350	440	AN	659	168
62.88%			**62.23%**			**60.55%**			**57.82%**		

genetic algorithm optimisation will yield the greatest AUC, i.e. the best performance. It also computes the granularisation in the shortest amount of time. Although the GA produced a marginally more accurate classification, it cannot be stated that the GA is always the optimal classifier. Measures are taken to ensure neither the GA or SA get stuck at local optima but rather global, it is due to this that similar results are achieved. The confusion matrices of the four methods are given in Table 3, the accuracy of each method is given under their confusion matrix. A confusion matrix is a visualisation tool, which contains information about actual and predicted classifications done by a classification system.

Where AP, AN, PP and PN are actual positive, actual negative, predicted positive and predicted negative respectively. As a result of implementing RST on the data set, the rules extracted are explicit and easily interpreted. RST will however compromise accuracy over rule interpretability, and this is brought about in the discretisation process where the granularity of the variables are decreased. The results obtained by optimising the partition sizes clearly produce better accuracies than that of EWB partitioning but there is no individual classifier that is significantly more accurate that the others. Using a method such as hill climbing may not always give the optimal solution as it is prone to finding a local optimum, not a global optimum as mentioned. Genetic algorithms and simulated annealing are better optimisation techniques to be applied to the general problem of discretising rough sets due to their nature of finding a global optimum.

5. CONCLUSION

Rough sets are formulated by discretising the rough set partitions using three optimisation techniques. The techniques used are the genetic algorithm, hill climbing and simulated annealing. The classification accuracy of the rough sets are obtained for the case of each classifiers, and are compared to each other as well as to that of equal-width-bin partitioning. These comparisons are done on HIV data obtained from an antenatal

sero-prevalence survey taken in South Africa in 2001. The genetic algorithm approach produced the best classification accuracy as well as the shortest computational time. Although there is no clear indication as to which classifier is the outright best for the HIV data set, it can be stated that the optimised partitions all produce higher classification accuracies than that of the non-optimised discretisation method of EWB partitioning. It must be noted that depending on the linearity and complexity of the data set used, the optimal technique to be used to discretise the partitions will vary. The rough sets produce a balance between transparency of the rough set model and accuracy of HIV estimation, but it does come at the cost of high computational effort. Recommendations for further improvement are given under the respective sections and these include the implementation of alternative optimisation techniques.

REFERENCES

1. Z. Pawlak, *Rough Sets, Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, 1991, chap. 3, p. 33.
2. G. D. Garson, *Social Science Computer Review* **9**, 399–433 (1991).
3. L. Zeng, *Sociological Methods and Research* pp. 499–524 (1999).
4. J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron, *A Rough Set Perspective on Data and Knowledge*, Oxford University Press: The Handbook of Data Mining and Knowledge Discovery, 1999.
5. Y. Yang, and R. John, "Roughness Bounds in Set-oriented Rough Set Operations," in *2006 IEEE International Conference on Fuzzy Systems*, 2006, pp. 1461–1468.
6. A. Ohrn, and T. Rowland, *American Journal of Physical Medicine and Rehabilitation* **79**, 100–108 (2000).
7. F. E. H. Tay, and L. Shen, *Engineering Applications of Artificial Intelligence* **16**, 39–43 (2003).
8. L. Zhai, L. Khoo, and S. Fok, *Computers and Industrial Engineering* **43**, 661–676 (2002).
9. B. Crossingham, and T. Marwala, *arXiv:0705.2485* (2007).
10. D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, MA, 1989, pp. 1–25.
11. S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Science, Number 4598, 13 May 1983* **220**, 671–680 (1983).
12. N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, *The Journal of Chemical Physics* **21**, 1087–1092 (1953).
13. A. Deja, and P. Peszek, *Fundamenta Informaticae* pp. 387–408 (2003).
14. U. Fayyad, and K. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1993, pp. 1022–1027.
15. S. Malve, and R. Uzsoy, *Computers and Operations Research* **34**, 3016–3028 (2007).
16. R. Hassan, B. Cohanim, and O. de Weck, "A Comparison of Particle Swarm Optimisation and the Genetic Algorithm," in *Proceedings of the 46th American Institute of Aeronautics and Astronautics*, 2005.
17. M. Mitchell, J. Holland, and S. Forest, *Advances in Neural Information Processing Systems 6, J. Cowan, G. Tesauro, and J. Alspector eds.* pp. 51–58 (1994).
18. K. Bryan, P. Cunningham, and N. Bolshkova, *IEEE Transactions on Information Technology in Biomedicine* **10**, 519–525 (2006).
19. R. Salazar, and R. Toral, *arXiv:cond-mat/9706051* (2006).
20. A. Lasry, G. S. Zaric, and M. W. Carter, *European Journal of Operational Research* **180**, 786–799 (2007).
21. Unaids, www.unaids.org/en/HIV_data/2006GlobalReport/default.asp/. Last accessed: 20/3/2007 (2006).
22. R. Department of Health, National hiv and syphilis sero-prevalence survey of women attending public antenatal clinics in south africa, http://www.info.gov.za/otherdocs/2002/hivsurvey01.pdf (2001).