Evolutionary Computing in Data Mining A. Ghosh and L C. Jain (editors), pp211–235, 2004 Springer.

10. Genetic Programming in Data Mining for Drug Discovery

W. B. Langdon and S. J. Barrett

Data Exploration Sciences, GlaxoSmithKline, Research and Development, Greenford, Middlesex, UK. http://www.cs.ucl.ac.uk/staff/W.Langdon

Summary.

Genetic programming (GP) is used to extract from rat oral bioavailability (OB) measurements simple, interpretable and predictive QSAR models which both generalise to rats and to marketed drugs in humans. Receiver Operating Characteristics (ROC) curves for the binary classifier produced by machine learning show no statistical difference between rats (albeit without known clearance differences) and man. Thus evolutionary computing offers the prospect of *in silico* ADME screening, e.g. for "virtual" chemicals, for pharmaceutical drug discovery.

The discovery, development and approval of a new drug treatment is a major undertaking (see Table 10.1). Only a small fraction of the drug discovery projects undertaken eventually lead to a successful medicine. Even successful programmes can take in the region of 12–15 years.

The discovery of new chemical entities with appropriate biological activity is a multi-stage and iteratively focussed search process in which many thousands of chemicals are measured firstly for primary activity against some (often novel) disease/therapy related target. The initial active subset subsequently becomes slimed down to select suitable candidates for use within the human body and worthy of expensive further development. The drug discovery process can be thought of as a funnel. The mouth of the funnel is wide and covers many diverse molecules. Gradually the funnel narrows and the later stages concentrate upon fewer more similar molecules.

As the directed discovery cycle continues, the criteria for progression become more stringent and complex. This means smaller numbers or classes of molecules are passed to the succeeding stages. Initialy molecules need only show some hint of activity against the target in relatively cheap *in vitro* tests. Later (early development) stages progress to more expensive and time consuming *in vitro* and *in vivo* measurements. *In vivo* measurements are required to demonstrate good bodily Absorption, Distribution, Metabolism, Excretion and Toxicity (ADMET) properties.

ADMET testing includes satisfying aspects relating to: 1) metabolism by, or inhibition of, critical metabolic enzymes (such as cytochrome P450) and 2) the molecule's ability to reach and stay in areas of the body required to enable sufficient drug effect to occur before its metabolism/excretion. (These properties are collectively known as good pharmacokinetics and bioavailability.)

Even an approximate *in silico* (computational) method that can be applied at an earlier step is very useful. Since it can be used by medicinal

Table 10.1. Discovery and early development cascade. At each successive stage in the cascade there is some increase in knowledge, but although the data becomes of higher quality it relates to a smaller (more specialised) chemical space as the decision about which chemicals to take forward becomes focussed to a smaller number of molecular classes and more complex as more factors are introduced.

Exploratory Screening
High/ultra-High Throughput Screening HTS/u-HTS

Primary screening wells contain a single concentration of test chemical and the target, together with reagents for a "bio-assay". The assay is designed to show if the chemical directly binds to the target at all, or can promote some bioactivity via interaction with the target. Many tens of thousands or even hundreds of thousands of very diverse chemicals are tested.

IC50/EC50 and early selectivity assays. More refined measurements of primary target binding/potency involve testing at a number of chemical concentrations to determine the concentration that is needed to reach 50% of maximum inhibition/activity.

Another set of assays (also known as initial selectivity assays) are designed to test non-target specific binding/activity for avoiding other (unwanted) effects. Thousands to many tens of thousands of chemicals are tested, depending upon earlier "hit" rates, the number of molecular classes with promise for required activity or modifiability, their collective content (the initial Structure-Activity Relations information,"early SAR") and the initial importance of specific selectivities.

"Early Lead"/"Back-up". Selection of promising molecular classes with the necessary potency and selectivity and which its feasible to mass manufacture. Early lead/back-up chemicals should allow for their "optimisation" as a drug.

Lead Optimisation

Chemical Programme of Modification

Using "*in silico*" virtual compound screening (i.e. selecting promising chemicals based on computer models) and/or more traditional QSAR/"rational" library design methods and combinatorial chemistry techniques many thousands of chemicals similar to the lead molecule class are identified and made in the hope they will have analogous and maybe improved properties.

Molecular class-focussed SAR screening with 1^{st} and 2^{nd} assays

Pharmacological characterisation to improve potency and/or selectivity.

Initial key and "scale-feasible" ADMET related testing *in vitro*

Permeability, p450 interactions, solubility, etc.

Results are feed back to the "Chemical Modification" stage, leading to an iterative cycle of chemical design, synthesis and testing.

"Development Candidate" with good potency, selectivity and initial ADMET

Further *in vivo* and more realistic and extensive ADMET and pharmacokinetic testing, including bioavailability measurements.

Again results are fed back to an increasingly more fine-tuned "Chemical Modification" stage.

Finally a compound is fit to be forwarded to first time in man (toxicity, dose-ranging) studies and subsequent clinical trials, plus supporting knowledge to be used in developing formulations/treatments.

chemists to assist them to decide which molecules to progress. The time and expense required for *in vivo* testing and its inherently low through-put nature make measuring ADMET properties of the many thousands of chemicals at the mouth of the funnel infeasible. However poor ADMET characteristics cause a high failure-rate of molecules in the later stages of drug discovery. An approximate screen helps ensure better quality molecules advance to ADMET measurement from earlier stages. That is, *in silico* screening helps to ensure the later stages of the drug discovery funnel are not clogged with chemicals which will ultimately have to be rejected due to their poor ADMET characteristics.

We illustrate the use of genetic programming in drug discovery by using it to evolve simple, biologically interpretable *in silico* models of bioavailability. Section 10.4 summarises GP, while Section 10.5 introduces ROC curves. Sections 10.6–10.7 and 10.8 describes the rat and human drug datasets and method. Our results (10.9), particularly Figure 10.15 on page 230, are discussed in Section 10.10 before we conclude with Section 10.11. However first we give the background of using computational, data mining and machine learning techniques in drug discovery (Sections 10.1 and 10.2) and describe bioavailability (Section 10.3).

10.1 Computational Drug Discovery

There is a long history of efforts to improve earlier decision-making in the drug discovery process. This has largely involved Quantitative Structure-Activity Relationship (QSAR) modelling employing traditional multivariate statistical techniques (see for example [10.1], while [10.2] gives a current review) and some quite fundamental indicators have arisen [10.3].

QSAR models link chemical's structure to their pharmacological activity. They are applicable to both library design (i.e. selection of which chemicals to keep in a drug discovery "library" or warehouse) and "virtual screening" (see [10.4]). The predictive performance of QSAR models is typically highly related to the number and diversity of chemicals that are used in modelling. This in turn is curtailed by the extent of biological testing that has been done. (Mostly the data has not been gathered directly to support the modelling process.)

In more recent years "machine learning" approaches have increasingly been experimented with (see [10.5, 10.6]) and applied in this area. Although evolutionary computing techniques, principally genetic algorithms, have been used for some while [10.7], e.g. in library design [10.8], newer paradigms such as genetic programming have only more recently been experimented with [10.9, 10.10] including the prediction of specific properties [10.11, 10.12, 10.13, 10.14, 10.15]. Although there is interest in other computational techniques [10.16].

10.2 Evolutionary Computing for Drug Discovery

Evolutionary computing and in particular genetic programming [10.17, 10.18] is increasingly being used to analyse bioinformatics data. For example recently two workshop series have started (BioGEC in the USA [10.19, 10.20] and EvoBIO in Europe [10.21]). Recent work also includes Kell [10.22], Moore [10.23], Iba [10.24] and Koza [10.25].

In the last few years we have used genetic programming in a series of data mining and modelling experiments. We used GP to fuse together models generated by other machine learning techniques (artificial neural networks, decision trees and naive Bayes [10.26, 10.27, 10.28]). This technique has been used for predicting inhibition of human cytochrome P450 2D6 (an important enzyme involved with the metabolism of many drugs) and compounds which might be starting points in drug based disease treatments [10.12, 10.13]. In a recent comparison, GP was shown to evolve models that best extrapolated from the available training data [10.14] as well as being understandable. Initial experiments have also used GP for both feature selection and model building of gene expression data [10.29].

10.3 Oral Bioavailability

The preferred method of introducing a drug into the body is for the patient to swallow it by mouth (orally). Hence a very important QSAR problem is the prediction of human oral bioavailability, %F. (%F is the percentage of an orally-administered dose reaching the blood stream.) Although some progress has been made, this task has proven particularly difficult. Typically false positive rates are quite high [10.4, 10.30, 10.31, 10.32] due to :

- The complex nature of processes underlying bioavailability. Bioavailability represents, in essence, the overall product of retained drug integrity within the body (e.g. avoiding high rates of digestion, metabolism in the liver and excretion via the kidneys). And the drug's ability to pass through bodily barriers while retaining its activity until the site of action (the target) is reached. Bioavailability is also a dose-dependent feature for many drugs. Obviously there are many physical and chemical processes involved.
- Modelling bioavailability is also hard due to the restricted set of classes of molecules for which %F measurements are available. This is because oral bioavailability is usually only measured late in the development process. So %F in man is normally only available for successfully marketed drugs or for proto-drugs which failed near the end of the drug discovery process.

Earlier efforts at examining molecular properties in relation to human oral bioavailability were naturally restricted to human data, but more recently progress has been made in identifying the importance of certain molecular properties from more voluminous rat data [10.33, 10.34].

10.4 Genetic Programming

Essentially genetic programming evolves a population of initialy randomly created programs using a fitness function to select the better ones to be parents for children in succeeding generations. The processes of sexual recombination and mutation are used to ensure, that while children have some similarity with previous generations, they are different from them. Using "survival of the fittest" better programs are produced.

It is common in GP to represent programs as parse trees (Lisp S-expressions). (Figure 10.8 an page 226 contains an example.) Parse trees have the great advantage that if one starts with two syntactically correct programs they can be used as parents of new children by exchanging subtrees between them (crossing over) and automatically the new programs will also be syntactically correct. Similarly various pruning and grafting operations can be implemented to mutate trees in such a way that they also give syntactically correct offspring. With a little practise, one can become adept at interpreting small trees and so extracting biological inferences from automatically created models.

GP can be thought of the use of genetic algorithms [10.35] to search the space of programs for one of the huge number of programs which satisfy some user requirement. Here the computer programs are all functional models of chemical properties. In addition to making accurate predictions of chemical properties, we will also want our models to be readily interpretable. It is difficult to quantify how interpretable a model is to a particular chemist or biologist, however, as a first step, it is reasonable to prefer smaller models. For an introduction into the long history of research into this aspect of genetic programming see [10.36].

10.5 Receiver Operating Characteristics

This section gives some of the background of Receiver Operating Characteristics (ROC) curves, while Section 10.5.1 uses ROC curves to explain why fitness $= \frac{1}{2}$True Positive rate $+ \frac{1}{2}$True Negative rate may be a good choice. That is, to explain why we use the average error rate rather than the error rate directly. (Note the two need not be the same when there are different numbers of training examples in each class. This is quite common.)

ROC curves are a good way to show the trade off a classifier makes between catching positive examples and raising false alarms [10.37]. Figure 10.9 (page 226) shows some ROC curves. ROC curves plot a classifier's true positive rate (i.e. fraction of positive examples correctly classified) against its false positive rate (i.e. fraction of negative examples which it gets wrong). All ROC curves lie in the unit square and must pass through two points 0,0 and 1,1. The origin 0,0 corresponds to when the classifiers sensitivity is so low that it always says no. I.e. it never detects any positive examples. While

1,1 means the classifier is so sensitive that it always says yes. I.e. it never makes a mistake on positive examples, but gets all negative cases wrong. A classifier that randomly guesses has an ROC which lies somewhere along the diagonal line connecting 0,0 and 1,1. A better than random classifier, has an ROC above the diagonal. Since an ideal classifier detects all positive cases but does not raise any false alarms its ROC curve goes through 1,0 (the top left corner of unit square).

A worse than random classifier has an ROC curve lying below the diagonal. It can be converted into better than random by inverting its output. This has the effect of rotating its ROC curve by 180 degrees, so that the ROC curve now lies above the diagonal.

Scott [10.38] suggests that the "Maximum Realisable Receiver Operating Characteristics" for a combination of classifiers is the convex hull of their individual ROCs, cf. also [10.39]. In [10.38] Scott proves a nice result: it is always possible to form a composite classifier whose ROC lies at any chosen point within the convex hull of the original classifier's ROC. (See Figures 10.1 and 10.2.) Since Scott's classifier is formed by random combination of the outputs, it is not acceptable where decisions have to justified individually, rather than on average across a group. E.g. some medical applications. Moreover the convex hull is not always the best that can be achieved [10.40]. Indeed we have shown GP can in some cases do better, including on Scott's own benchmarks [10.27] and several real world pharmaceutical classification tasks [10.11].

It can be shown that the overall discriminative ability of a classifier (as measured by its Wilcox statistic) is equal to the area under its ROC curve [10.41]. Thus we have used the area under the ROC (AUROC) to decide which classifiers to allow to breed and have children in the next generation. That is, we have used AUROC as the fitness measure.

10.5.1 Simple use of ROC as the Objective to be Maximised

It is not necessary to measure each Receiver Operating Characteristics curve completely during evolution. Instead some computational savings can be made by basing the fitness function on measuring only one point on the ROC. We can still form the convex hull of the ROC, but now it is always a quadrilateral (see Figure 10.3). Given a little geometry, it can be shown that the area of the quadrilateral (i.e. the AUROC of the convex hull) is $\frac{1}{2}$True Positive rate $+ \frac{1}{2}$True Negative rate. This gives a very simple formula for fitness calculation which automatically takes into account class imbalances.

Note again geometry tells us that two points that are equally distant from the diagonal crossing the ROC square from 0,0 to 1,1 will give rise to quadrilaterals with the same area and hence have the same fitness. Another way of looking at this is to consider the cost associated with the classifier.

We assume the costs can be represented as α = cost of a false positive (false alarm) and β = cost of missing a positive (false negative). Let p be the

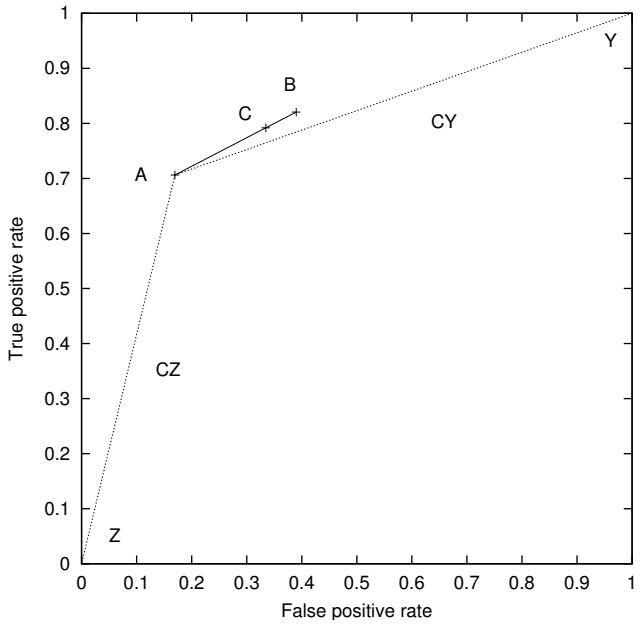

Fig. 10.1. Given two classifiers (A and B) a composite classifier (C) can always be formed by returning the result of A a fixed fraction of the time and the prediction given by B otherwise. The Receiver Operating Characteristics of C will lie on a straight line connecting A and B. By combining with the classifier which always says no (Z) a composite CZ can be constructed between a real classifier and the origin (Z). Similarly, a classifier which always says yes (Y) can always be used to give a classifier (CY) between a real classifier and the 1,1 corner.

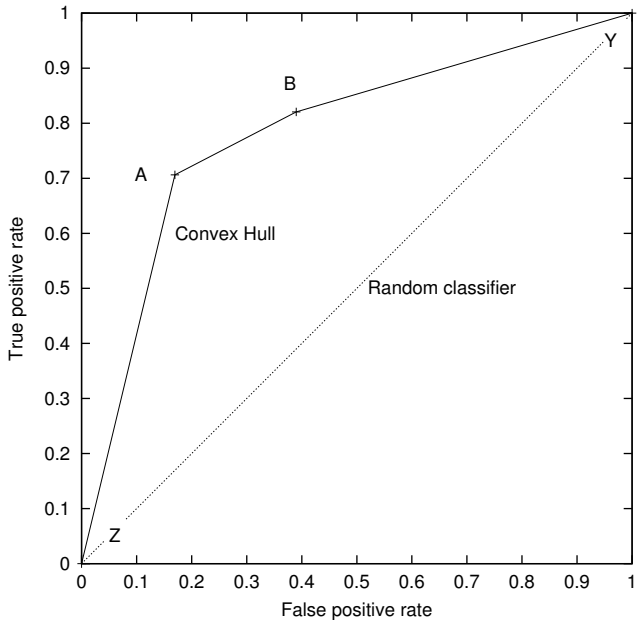

Fig. 10.2. A composite classifier can always be formed which will have an ROC lying between the convex hull of all available classifiers and the diagonal line between 0,0 and 1,1. (Genetic programming and other techniques can sometimes fuse classifiers to yield improved classifiers which lie above the convex hull.)

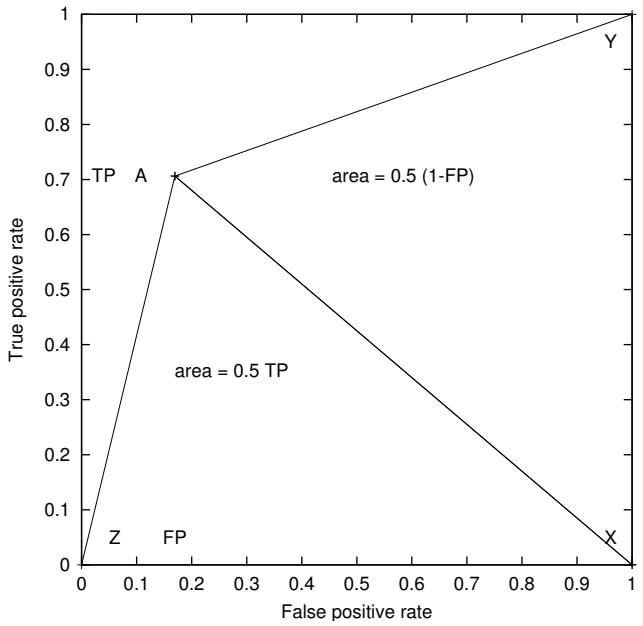

Fig. 10.3. With only one measured point on the ROC, the convex hull reduces to a quadrilateral (ZAYX). ZAYX is composed of triangles ZAX and XAY. ZAX's area is $\frac{1}{2}$TP. XAY's area is $\frac{1}{2}(1 - \text{FP})$.

proportion of positive cases. Then the average cost of classification at point x, y in the ROC space is $(1-p)\alpha x + p\beta(1-y)$.

Lines of equal cost are parallel and straight. Their gradient is $\alpha/\beta \times (1-p)/p$. If the cost of error on the two classes are equal ($\alpha = \beta$) and 50% are positive ($p = 0.5$), the gradient is 1 and the lines are at 45 degrees, i.e. parallel to the diagonal. Note that our GP fitness function ($\frac{1}{2}$True Positive rate + $\frac{1}{2}$True Negative rate) also rewards equally points that are equally distant from the diagonal. In other words it treats errors on both classes as if they were being equally important. It is a simple alternative to error rate, and it deals with the common case that more training data is available for one class than the other.

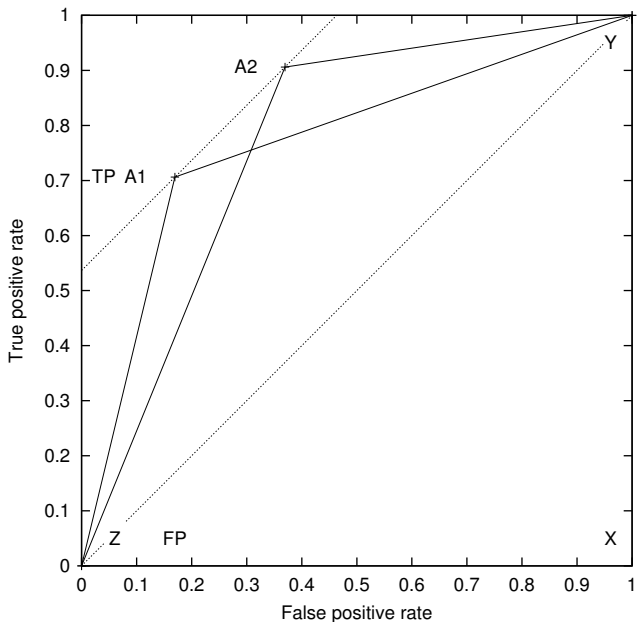

Fig. 10.4. With only one measured point on the ROC, the convex hull reduces to a quadrilateral (ZAYX) whose area is $0.5\text{TP} + 0.5(1 - \text{FP})$. Since A_1 and A_2 are the same distance from the diagonal ZY, the triangles ZA_1Y and ZA_2Y have the same area. Thus the quadrilaterals ZA_1YX and ZA_2YX have the same area.

10.6 The Bioavailability Data

In high throughput screening (HTS) [10.11, 10.12, 10.13] and IC50 [10.14] experiments the chemical properties of many thousands of chemicals were measured in solution. In contrast, oral bioavailability is measured in living

organisms. This severely limits the number of measurements. In fact only 481 data points on human subjects were initialy available. These are data for marketed drugs.

The available data were randomly partitioned into 321 to be used for training and 160 held back as a holdout set. I.e. to assess how well the evolved models perform on data which they were not trained on.

The chemicals selected for measurement are naturally a highly biased sample. The sample consists only of chemicals which made it through the drug discovery process. There are two things we would like to know about any predictive classifier; how well it will work on chemicals like those on which it was trained and secondly (and much more difficult), how well will it extrapolate outside the training domain.

In addition to the training data we also had access to two further datasets. A further 124 human records, and data from rats. There was almost no overlap between the drugs in the human datasets and the chemicals in the rat dataset. However the rat dataset was naturally more extensive. Before using the rat data, chemicals known to have a specific bioavailability difference (related to "clearance") in humans to that in rats where excluded, leaving 2013 chemicals. Figure 10.5 shows, as expected, there are systematic differences between the rat and the human data.

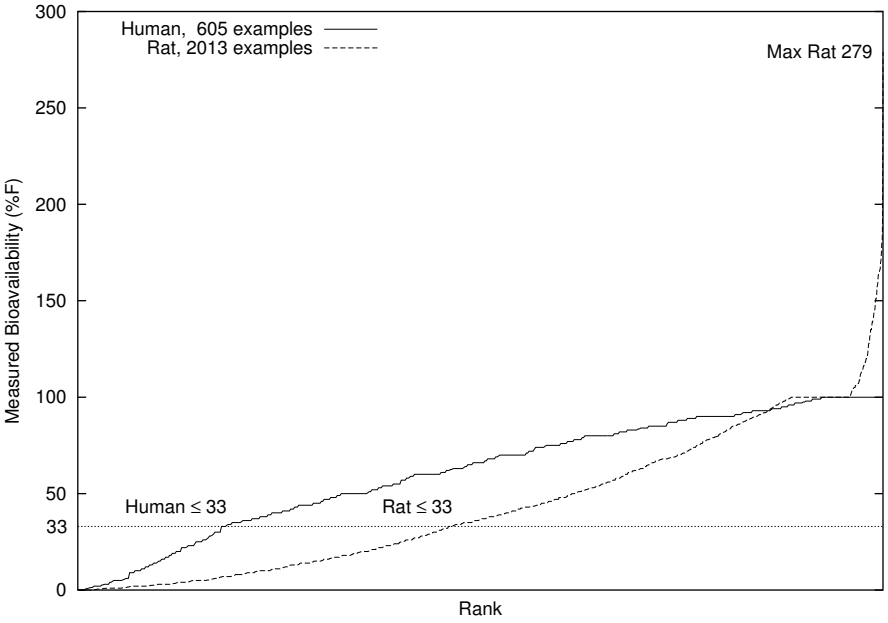

Fig. 10.5. Human and rat bioavailability data. Note only 19% of human data has a bioavailability of 33 or below, while 47% of the rat dataset is in class 0.

10.7 Chemical Features

An implicit modelling goal is that the model should be readily applicable to novel chemicals. Indeed we may wish to use the evolved model to predict the chemical properties of chemicals that have yet to be synthesised. I.e. "virtual chemicals" which exist only in the computer so far. Therefore the models must be based on chemical formulae, rather than three dimensional shapes (conformations). Since chemistry (particularly biochemistry) is inherently three dimensional this is a fundamental restriction. Nonetheless, as we shall see, predictions can still be made.

Chemical formulae can be viewed as graphs with labelled nodes (the atoms) and labelled edges (the bonds) (see Figure 10.6). However it can be inconvenient to work with such graphs. Instead it is common practise in Cheminformatics not to work from the chemical formulae directly but instead to precalculate chemical "features". The pharmaceutical industry has considerable experience with designing features. Simple features include, the presence or absence of charged atoms, aromatic rings, specific groups and metallic atoms.

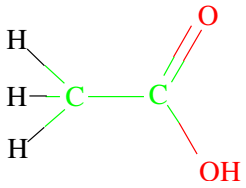

Fig. 10.6. Chemical structure of Acetic Acid. Acetic Acid's chemical formula is CH_3CO_2H. While its SMILES (Simplified Molecular Input Line Entry Specification) representation is CC(=O)O. In the SMILES representation all hydrogen atoms are omitted, branches are shown with parenthesis and double bonds with "=".

A total of 83, numerical and categorical, chemical features from a diverse array of families (electronic, structural, topological/shape, physico-chemical, etc.) were computed for each chemical, starting from a SMILES[1] representation of it's primary chemical structure (2-d chemical formula).

Of the 83 features, all but 7 had previously been used in when modelling chemical interaction with a P450 cell wall enzyme [10.14].

10.8 Genetic Programming Configuration

The genetic programming system is deliberately simple. For example the GP uses a single type. So categorical and integer as well as continuous measure-

[1] `http://www.daylight.com/dayhtml/smiles/`

ments and features are converted to single precision floating point numbers before the GP is run. The GP is summarised in Table 10.2.

10.8.1 Function Set

The functions used for combining feature values and numerical values are the four binary arithmetic functions $(+ - * /)$ and a four input "if". Since functions are initialy used randomly, division is "protected". This means division by zero is trapped and the value 1 is returned rather than attempting to perform division by zero.

IFLTE evaluates its first two arguments. If the first is less than or equal to the second, IFLTE returns the value of its third argument. Otherwise it returns the value of its fourth argument.

10.8.2 Terminal Set

The terminals or leaves of the trees being evolved by the GP are either pre-calculated compound features (cf. Section 10.7) or constants (see Table 10.2). GProc does not use "ephemeral random constants". Instead all the numeric values used by the evolved expressions are chosen from the fixed initial set. The random constants are drawn from a very non-uniform distribution of both positive and negative values, with about 500 values lying between -10 and +10. This is generated using a "tangent" distribution [10.42]. Random values are uniformly generated in the range $0 \ldots \pi$ and then scaled by multiplying by 10. Duplicates are discarded. Finally the integers 0 to 9 are also included to assist with categorical data. Figure 10.7 shows the distribution of constants.

10.8.3 GP Genetic Operations and other Parameters

Following [10.44] and others, we use a high mutation rate and a mixture of different mutation operators. To avoid bloat, we also use size fair crossover [10.43] and limit the maximum model size to 63, see Table 10.2.

10.8.4 GP Fitness Function

Each individual returns a real number. This is treated as if it was a prediction of the true percentage bioavailability. However first it is truncated to force it to lie in the range $0 \ldots 100$. There are two components of fitness 1) error squared and 2) $\frac{1}{2}\mathrm{TP} + \frac{1}{2}(1 - \mathrm{FP})$ (cf. Section 10.5.1).

Error squared is simply the sum of the squared difference between the (truncated) value returned by the individual and the measured bioavailability across all the training compounds, divided by the number of training compounds.

Table 10.2. GP Parameters

Objective:	Evolve a predictive model of human bioavailability
Function set:	MUL ADD DIV SUB IFLTE
Terminal set:	83 features, 0 0 1 2 3 4 5 6 7 8 9 plus 1000 unique random constants
Fitness:	$100,000 \times \left(\frac{1}{2}\text{TP} + \frac{1}{2}(1 - \text{FP})\right) - \sum \lvert a - actual \rvert^2$/num chemicals Measured on 321 (59- and 262+) drugs selected for training
Selection:	generational (non elitist), tournament size 7
Wrapper:	Force into range 0..100 (i.e. if $a < 0, a = 0$ if $a > 100, a = 100$) Iff $a > 33$ Predict ok.
Pop Size:	500
Max model size:	63
Initial pop:	Each individual comprises one tree each created by ramped half-and-half (2:6) (half terminals are constants)
Parameters:	50% size fair crossover, crossover fragments $\leq$ 30 [10.43] 50% mutation (point 22.5%, constants 22.5%, shrink 2.5% subtree 2.5%)
Termination:	generation 50

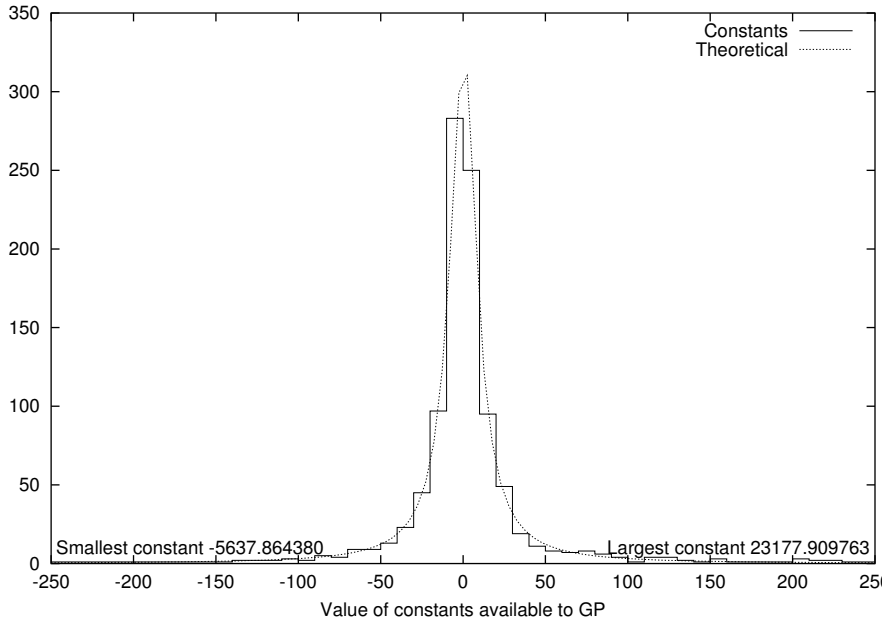

Fig. 10.7. Distribution of fixed constants. Theoretical line ($\frac{1}{r\pi}\frac{1}{1+(x/r)^2}$, $r = 10$) is derived from derivative of inverse of tangent function used to randomly chose the constants. Scaling factor r means we expect 50% of constants to lie in $-r \ldots r$, 10% to lie outside $-6.314r \ldots 6.314r$, 1% to lie outside $-63.66r \ldots 63.66r$, 0.1% to lie outside $-636.6r \ldots 636.6r$ etc.

To calculate the true positive (TP) and false positive (FP) rates, the (truncated) value returned by the tree is compared with the threshold 33. Values ≤ 33 are treated as predicting poor bioavailability (negative) while values above 33 indicate positives. The individual's overall fitness is given by the weighted sum of the two components: $f = 100,000 \times \left(\frac{1}{2}\mathrm{TP} + \frac{1}{2}(1 - \mathrm{FP})\right) + \frac{\sum |error|^2}{\text{number cases}}$. (100,000 was chosen as the weighting factor empirically. A Pareto approach might have been used instead [10.42].)

10.9 Experiments

We conducted two experiments, both of five runs. The first used 321 compounds randomly drawn from the first 481 human data records. The second set used 1342 records randomly chosen from the 2013 records for rats. Apart from the training data, the second set of runs were identical to the first, cf. Table 10.2.

10.9.1 Training on 321 Human records

The fittest models in generation 50 of the five runs were compared. The model shown in Figure 10.8 was chosen as the best overall. This was primarily because the difference between its (ROC) performance on the training and 160 human test records was the smallest of the five. A small difference suggests that the model does not over fit the training data and so may generalise to other chemicals. While the test data was used for model selection, the model's performance was later tested on a further 124 holdout records. No significant difference was found. We are reasonable confident that the model's ROC ("605" in Figure 10.9) is a fair indication of its likely performance on *similar* data.

After model selection, it was tested on the 2013 compounds whose bioavailability in rats was known. Its performance was significantly worse.

10.9.2 Training on 1342 Rat Records

Again the five fittest models evolved at the end of the five runs where compared, and again the one (shown in Figure 10.10) was chosen since it had the smallest difference between training and test performance. Its performance on its training data is noticeably worse than that trained on the human data set. However (see Figure 10.11) its performance does not fall away when tested on data from another species (i.e. human). That is, our second experiments automatically produced a simpler model with wider applicability than the first but at the cost of lower predictive performance.

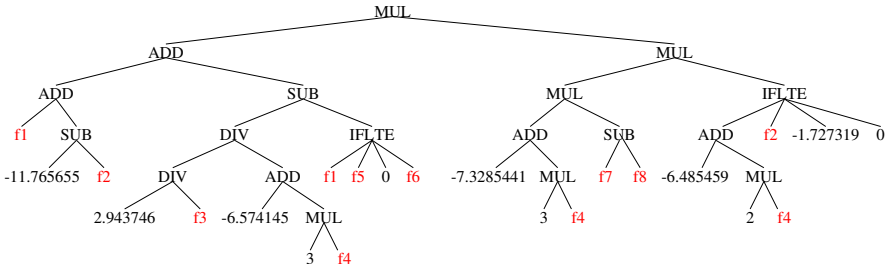

Fig. 10.8. Evolved model of bioavailability created using human training data. The model has been simplified to remove redundant code, without affecting its performance. Final size 55.

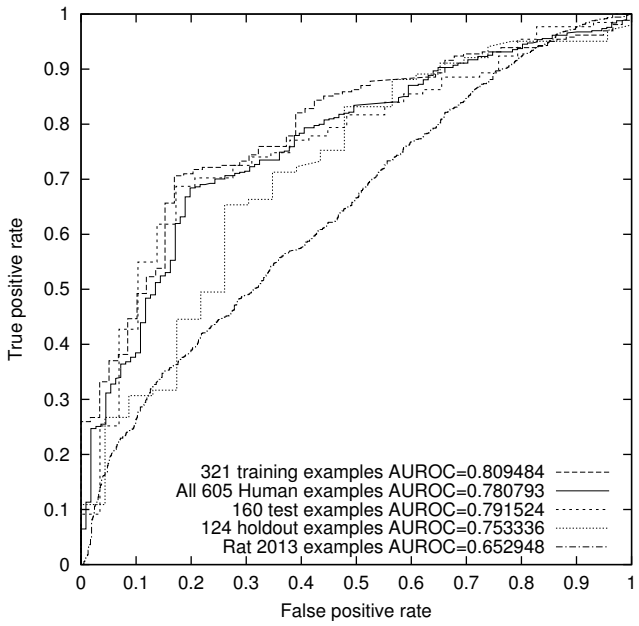

Fig. 10.9. Receiver Operating Characteristics of evolved model of bioavailability created using human training data. (Model is shown in Figure 10.8. It has been simplified by hand). Performance across all human data includes (20%, 68%) while for the same (68%) true positive rate the classifier only achieves FP=51% across all rat data. There is no statistical significance [10.41] between the area under the curves (AUROC) for the four human datasets. However the difference between AUROCs of the combined human and rat curves is unlikely to be due to chance fluctuations.

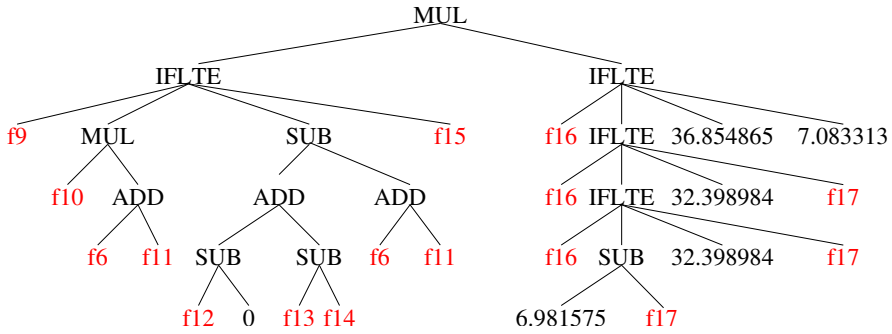

Fig. 10.10. Evolved model of bioavailability created using rat training data. Size 35. Only feature f6 is used by both this model and that shown in Figure 10.8. One can readily see that the model falls into the product of left and right components. The left evaluated to either f11+f12+f13-f14-f16 or f15 depending if f9≤f10(f6+f11). While the right hand side comes to either a small value (7.083313) or a large (36.854865) value depending only on f16 v. f17. The RHS can be simplified to if f16> 6.981575−f17 then 7.083313 otherwise 36.854865, with only marginal effect.

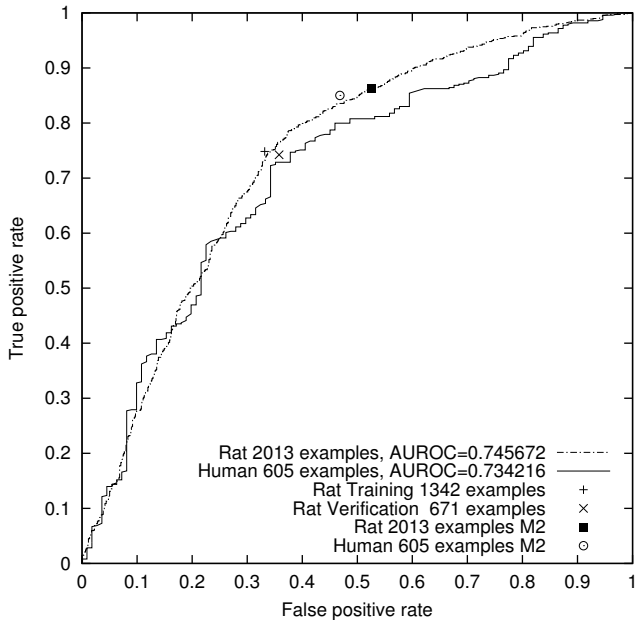

Fig. 10.11. Receiver Operating Characteristics of evolved model of bioavailability created using rat training data (cf. Figure 10.10). This model achieves a false positive rate of 32% for a true positive rate of 70%. Only a single point on the ROC (shown with +) is used to assess the performance of the classifiers as they are evolved. (The performance of this classifier on the rat validation data is shown with ×.) Hanley's statistical significance test [10.41] shows the difference in the AUROC of the rat and human curves can be explained as chance fluctuations. The two M2 points refer to the simplified model shown in Figure 10.15.

10.9.3 Simplification of Evolved Model of Rat Bioavailability

As Figure 10.10 makes clear the GP model evolved using the rat data is unnecessarily complex. The first of three stages of simplification was to replace the large nested if statement on the right hand side (RHS) by a single if (actually by `(IFLTE f16 (SUB 6.981575 f17) 36.854865 7.083313)`, cf. Figure 10.12). In the second stage, the new model was used to seed a new GP run. In order to encourage the evolution of still simpler models, their maximum size was reduced and the fraction of shrink mutation was greatly increased. Details are given in Table 10.3.

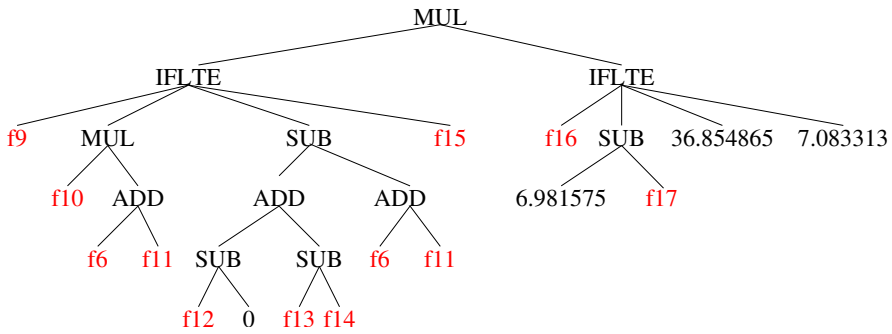

Fig. 10.12. Seed loaded into GP population for automatic simplification. Size 27. (Produced by shrinking RHS of model evolved using rat data, cf. Figure 10.10).

Table 10.3. GP Parameters used to simplify model, cf. Table 10.2. (All parameters were as the first set of GP runs on rat training data except where given.)

Objective:	Simplify best model evolved using rat bioavailability data
Selection:	Elitist. (Generational and tournament size 7, as before)
Max model size:	41
Initial pop:	100% seeded with model shown in Figure 10.12. Each new subtree inserted by subtree mutation is created with ramped half-and-half as before but with max depth range (1:2), rather than (2:6).
Parameters:	10% size fair crossover, 90% mutation (point 4.5%, constants 4.5%, shrink 76.5%, subtree 4.5%)

After 50 generations, this run produced a simplified model of similar performance. The RHS was unchanged but the LHS was simplified. In the new model the the first argument of the if (f9) was replaced by f15, two of the features in the second argument were replaced and the large subtree which had been the third argument of the if was replaced by zero, see Figure 10.13. In the third stage the model was further simplified by hand.

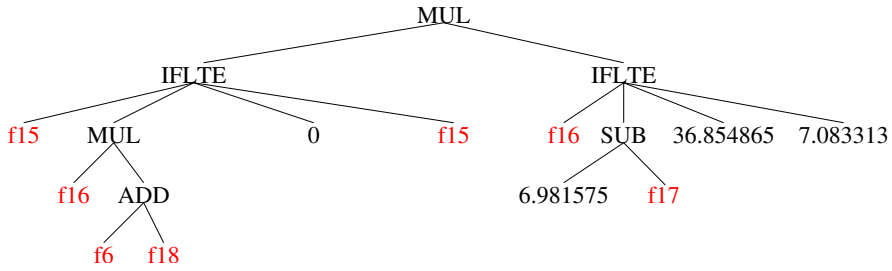

Fig. 10.13. Automatic simplification of seed model (cf. Figure 10.12) produced at generation 50 by GP. Size 17.

The model was rewritten to reverse the order of the multiplication and the ifs (making no semantic difference). The subexpression f16 × (f6+f18) was replaced by zero and the two remaining constants were rounded to the nearest integer. Yielding if(f_1(f15, f16, f17) $>$ 33 then bioavailability is ok. Where f_1 = if f15 $\leq$ 0 then 0 else if f16 $<$ (7 − f17) then f15 × 37 else f15 × 7, cf. Figure 10.14. Which in turn can be simplified to if f15$>$ 0 and f16 $<$ (7 − f17) then predict bioavailability is ok. This final model (M2, Figure 10.15) has a true positive rate of 85% (human) 88% (rat) with a corresponding false positive rate of 47% (human) 55% (rat). These points are plotted as M2 on Figure 10.11.

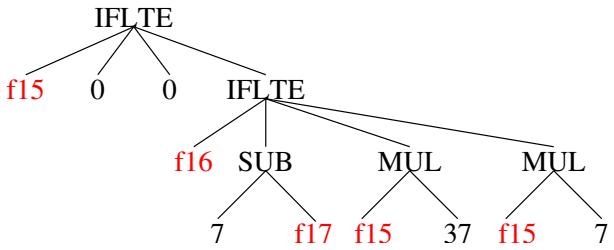

Fig. 10.14. Simplification of compacted model slimed by GP (cf. Figure 10.13).

Note, apart from re-arranging the if and multiplication operators, each simplification made by hand resulted in a slight reduction in classification performance. In principle, if one could establish a quantifiable trade off between model complexity and accuracy (perhaps based on information theory) these steps could have been automated.

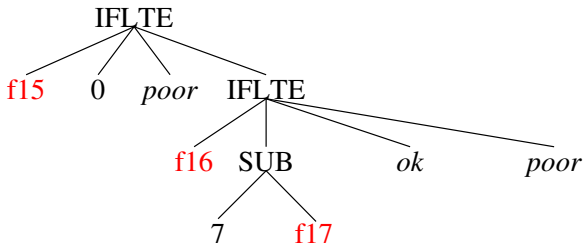

Fig. 10.15. Predicting bioavailability (M2 production version).

10.10 Discussion

The results of the previous section indicate that we can automatically evolve predictive models of bioavailability for chemicals like those for which we already have measurements.

A, possibly simplistic, explanation for the difference between the results when training with human data and rat data is simply the size and nature of the training data. When the volume of data is small its easy to get higher performance in the limited domain. However, in these cases, machine learning models are liable not to extrapolate out of their training domain. Thus the first model works on human data but fails on the rat data. In contrast, Figure 10.5 (page 221) suggests that the rat data is more diverse (and thus harder to learn) but also covers the space occupied by marketed drugs as a subset. Since modelling the rat data is harder, performance is somewhat reduced but, importantly, it does not fall off on the human data set. We tried to confirm these assumptions using a cluster analysis but this was somewhat inconclusive. Obtaining identical performance on different species with different chemicals is very encouraging.

It is critical that models are able to predict properties of chemicals, when we do not know them. Our experiments tend to reinforce the view that this is possible only where chemicals are like those we have already met. Accuracy will fall away with dissimilar chemicals. Nonetheless in drug discovery we are dealing with a very limited part of the whole of chemical space and so we may hope for useful extrapolation.

The simplicity of the evolved model (Figure 10.15) suggests that our initial assumption that many mechanisms are involved in ensuring molecules reach the blood stream was too cautious. The simplicity hints that perhaps either there are few dominating mechanisms or that many mechanisms are similar. A second, less encouraging, possibility is that few mechanisms are involved with the observed chemicals because the chemicals we have measurements for, are too similar to each other. That is, a more complex model would be needed to cover the whole chemical space of potential drugs. Also the model suggests that the GSK features are indeed suitable for modelling important drug properties.

Other pharmaceutical uses of data mining might be to highlight which events in patient histories are important and to sift through records to differentiate drug interactions from the normal background prevalence of unrelated or unchanged diseases. Following the success of the Human Genome project, it is anticipated that DNA data (e.g. gene expression levels) will soon lead to the rapid identification of patients with genetic predispositions to both disease and adverse reactions to drug treatments. However, at least at present, DNA chip data presents a difficult data mining problem.

10.11 Conclusion

We have used genetic programming (GP) to automatically create interpretable predictive models of a small number of very complex biological interactions of great interest to medicinal and computational chemists who search for new drug treatments. Particularly, we have found a simple predictive model of human oral bioavailability (Figure 10.15). While the models are not (and probably can never be) 100% accurate, they use readily available data and so can be used to guide the choice of which molecules to forward to the next (more expensive) stage in the drug selection process. Notably the models can make *in silico* predictions about "virtual" chemicals, e.g. to decide if they are to be synthesised.

Since the models are simple and presented as mathematical functions, they can be readily ported into other tools, e.g. spreadsheets, database queries and intranet pages. Little more than "cut and paste" may be required. Run time of the GP part of the model (once produced) is unlikely to be an issue.

Acknowledgements

We would like to thank Sandeep Modi and Chris Luscombe.

References

10.1 David Livingstone. *Data Analysis for Chemists – Applications to QSAR and Chemical product Design*. Oxford University Press, 1995.

10.2 Han van de Waterbeemd and Eric Gifford. ADMET *in silico* modelling: towards prediction paradise? *Nature Reviews Drug Discovery*, 2(3):192–204, March 2003.

10.3 Christopher A. Lipinski, Franco Lombardo, Beryl W. Dominy, and Paul J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 23(1-3):3–25, 15 January 1997.

10.4 Sean Ekins, Chris L. Waller, Peter W. Swaan, Gabriele Cruciani, Steven A. Wrighton, and James H. Wikel. Progress in predicting human ADME parameters *in silico*. *Journal of Pharmacological and Toxicological Methods*, 44(1):251–272, July-August 2000.

10.5 Journal of chemical information and computational sciences.
10.6 Brent L. Podlogar and Ingo Muegge. "Holistic" *in silico* methods to estimate the systemic and CNS bioavailabilities of potential chemotherapeutic agents. *Current Topics in Medicinal Chemistry*, 1(4):257–275, 2001.
10.7 Gareth Jones. Genetic and evolutionary algorithms. In *Encyclopedia of Computational Chemistry*. John Wiley & Sons, Ltd., September 1998.
10.8 Robert P. Sheridan, Sonia G. SanFeliciano, and Simon K. Kearsley. Designing targeted libraries with genetic algorithms. *Journal of Molecular Graphics and Modelling*, 18(4-5):320–334, August-October 2000.
10.9 Orazio Nicolotti, Valerie J. Gillet, Peter J. Fleming, and Darren V. S. Green. Multiobjective optimization in quantitative structure-activity relationships: Deriving accurate and interpretable QSARs. *Journal of Medicinal Chemistry*, 45(23):5069–5080, 2002.
10.10 Mark Kotanchek, Arthur Kordon, Guido Smits, Flor Castillo, R. Pell, M. B. Seasholtz, L. Chiang, P. Margl, P. K. Mercure, and A. Kalos. Evolutionary computing in Dow Chemical. In Lawrence "Dave" Davis and Rajkumar Roy, editors, *GECCO-2002 Presentations in the Evolutionary Computation in Industry Track*, pages 101–110, New York, New York, 11-13 July 2002.
10.11 W. B. Langdon, S. J. Barrett, and B. F. Buxton. Genetic programming for combining neural networks for drug discovery. In Rajkumar Roy, Mario Köppen, Seppo Ovaska, Takeshi Furuhashi, and Frank Hoffmann, editors, *Soft Computing and Industry Recent Applications*, pages 597–608. Springer-Verlag, 10–24 September 2001. Published 2002.
10.12 William B. Langdon, S. J. Barrett, and B. F. Buxton. Combining decision trees and neural networks for drug discovery. In James A. Foster, Evelyne Lutton, Julian Miller, Conor Ryan, and Andrea G. B. Tettamanzi, editors, *Genetic Programming, Proceedings of the 5th European Conference, EuroGP 2002*, volume 2278 of *LNCS*, pages 60–70, Kinsale, Ireland, 3-5 April 2002. Springer-Verlag.
10.13 W. B. Langdon, S. J. Barrett, and B. F. Buxton. Comparison of adaboost and genetic programming for combining neural networks for drug discovery. In Günther R. Raidl, Stefano Cagnoni, Juan Jesús Romero Cardalda, David W. Corne, Jens Gottlieb, Agnès Guillot, Emma Hart, Colin G. Johnson, Elena Marchiori, Jean-Arcady Meyer, and Martin Middendorf, editors, *Applications of Evolutionary Computing, EvoWorkshops2003: EvoBIO, EvoCOP, EvoIASP, EvoMUSART, EvoROB, EvoSTIM*, volume 2611 of *LNCS*, pages 87–98, University of Essex, UK, 14-16 April 2003. Springer-Verlag.
10.14 W. B. Langdon, S. J. Barrett, and B. F. Buxton. Predicting biochemical interactions – human P450 2D6 enzyme inhibition. In *Proceedings of the 2003 Congress on Evolutionary Computation CEC2003*, Canberra, 8-12 December 2003. IEEE Press.
10.15 William Bains, Richard Gilbert, Lilya Sviridenko, Jose-Miguel Gascon, Robert Scoffin, Kris Birchall, Inman Harvey, and John Caldwell. Evolutionary computational methods to predict oral bioavailability QSPRs. *Current Opinion in Drug Discovery and Development*, 5(1):44–51, January 2002.
10.16 Balaji Agorama, Walter S. Woltosza, and Michael B. Bolger. Predicting the impact of physiological and biochemical processes on oral drug bioavailability. *Advanced Drug Delivery Reviews*, 50(Supplement 1):S41–S67, 1 October 2001.
10.17 John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
10.18 Wolfgang Banzhaf, Peter Nordin, Robert E. Keller, and Frank D. Francone. *Genetic Programming – An Introduction; On the Automatic Evolution of*

Computer Programs and its Applications. Morgan Kaufmann, dpunkt.verlag, 1998.

10.19 Wolfgang Banzhaf and James A. Foster, editors. *Biological Applications of Evolutionary Computation (BioGEC 2002)*, New York, 8 July 2002. AAAI.

10.20 Wolfgang Banzhaf and James A. Foster, editors. *Biological Applications of Evolutionary Computation (BioGEC 2003)*, Chigaco, 11 July 2003. AAAI.

10.21 Elena Marchiori and David W. Corne, editors. *EvoBIO, the first European workshop on Evolutionary Bioinformatics*, volume 2611 of *LNCS*, University of Essex, UK, 14-16 April 2003. Springer-Verlag.

10.22 Douglas Kell. Defence against the flood. *Bioinformatics World*, pages 16–18, January/February 2002.

10.23 Jason H. Moore, Joel S. Parker, Nancy J. Olsen, and Thomas M. Aune. Symbolic discriminant analysis of microarray data in automimmune disease. *Genetic Epidemiology*, 23:57–69, 2002.

10.24 Hitoshi Iba and Erina Sakamoto. Inference of differential equation models by genetic programming. In W. B. Langdon, E. Cantú-Paz, K. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. Honavar, G. Rudolph, J. Wegener, L. Bull, M. A. Potter, A. C. Schultz, J. F. Miller, E. Burke, and N. Jonoska, editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 788–795, New York, 9-13 July 2002. Morgan Kaufmann Publishers.

10.25 John R. Koza, William Mydlowec, Guido Lanza, Jessen Yu, and Martin A. Keane. Reverse engineering of metabolic pathways from observed data by means of genetic programming. In *First International Conference on Systems Biology (ICSB)*, Tokyo, 14-16 November 2000.

10.26 William B. Langdon and Bernard F. Buxton. Evolving receiver operating characteristics for data fusion. In Julian F. Miller, Marco Tomassini, Pier Luca Lanzi, Conor Ryan, Andrea G. B. Tettamanzi, and William B. Langdon, editors, *Genetic Programming, Proceedings of EuroGP'2001*, volume 2038 of *LNCS*, pages 87–96, Lake Como, Italy, 18-20 April 2001. Springer-Verlag.

10.27 W. B. Langdon and B. F. Buxton. Genetic programming for combining classifiers. In Lee Spector, Erik D. Goodman, Annie Wu, W. B. Langdon, Hans-Michael Voigt, Mitsuo Gen, Sandip Sen, Marco Dorigo, Shahram Pezeshk, Max H. Garzon, and Edmund Burke, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 66–73, San Francisco, California, USA, 7-11 July 2001. Morgan Kaufmann.

10.28 W. B. Langdon and B. F. Buxton. Genetic programming for improved receiver operating characteristics. In Josef Kittler and Fabio Roli, editors, *Second International Conference on Multiple Classifier System*, volume 2096 of *LNCS*, pages 68–77, Cambridge, 2-4 July 2001. Springer Verlag.

10.29 W. B. Langdon and B. Buxton. Genetic programming for mining DNA chip data from cancer patients. *Genetic Programming and Evolvable Machines*, 2004.

10.30 Fumitaka Yoshida and John G. Topliss. QSAR model for drug human oral bioavailability. *Journal of Medicinal Chemistry*, 43(13):2575–2585, 2000.

10.31 C. Webster Andrews, Lee Bennett, and Lawrence X. Yu. Predicting human oral bioavailability of a compound: Development of a novel quantitative structure-bioavailability relationship. *Pharmaceutical Research*, 17(6):639–644, June 2000.

10.32 Marco Pintore, Han van de Waterbeemd, Nadege Piclin, and Jacques R. Chretien. Prediction of oral bioavailability by adaptive fuzzy partitioning.

European Journal of Medicinal Chemistry, 38(4):427–431, 2003. XVIIth International Symposium on Medicinal Chemistry.

10.33 Daniel F. Veber, Stephen R. Johnson, Hung-Yuan Cheng, Brian R. Smith, Keith W. Ward, and Kenneth D. Kopple. Molecular properties that influence the oral bioavailability of drug candidates. *Journal of Medicinal Chemistry*, 45(12):2615–2623, 2002.

10.34 Arun K. Mandagere, Thomas N. Thompson, and Kin-Kai Hwang. Graphical model for estimating oral bioavailability of drugs in humans and other species from their Caco-2 permeability and *in vitro* liver enzyme metabolic stability rates. *Journal of Medicinal Chemistry*, 45(2):304–311, 2002.

10.35 David E. Goldberg. *Genetic Algorithms in Search Optimization and Machine Learning*. Addison-Wesley, 1989.

10.36 William B. Langdon, Terry Soule, Riccardo Poli, and James A. Foster. The evolution of size and shape. In Lee Spector, William B. Langdon, Una-May O'Reilly, and Peter J. Angeline, editors, *Advances in Genetic Programming 3*, chapter 8, pages 163–190. MIT Press, 1999.

10.37 John A. Swets, Robyn M. Dawes, and John Monahan. Better decisions through science. *Scientific American*, 283(4):70–75, October 2000.

10.38 M. J. J. Scott, M. Niranjan, and R. W. Prager. Realisable classifiers: Improving operating performance on variable cost problems. In Paul H. Lewis and Mark S. Nixon, editors, *Proceedings of the Ninth British Machine Vision Conference*, volume 1, pages 304–315, University of Southampton, UK, 14-17 September 1998.

10.39 Foster Provost and Tom Fawcett. Robust classification for imprecise environments. *Machine Learning*, 42(3):203–231, March 2001.

10.40 Y. Yusoff, J. Kittler, and W. Christmas. Combining multiple experts for classifying shot changes in video sequences. In *IEEE International Conference on Multimedia Computing and Systems*, volume II, pages 700–704, Florence, Italy, 7-11 June 1998.

10.41 James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, April 1982.

10.42 William B. Langdon. *Genetic Programming and Data Structures: Genetic Programming + Data Structures = Automatic Programming!*, volume 1 of *Genetic Programming*. Kluwer, Boston, 1998.

10.43 William B. Langdon. Size fair and homologous tree genetic programming crossovers. *Genetic Programming and Evolvable Machines*, 1(1/2):95–119, April 2000.

10.44 Peter J. Angeline. Multiple interacting programs: A representation for evolving complex behaviors. *Cybernetics and Systems*, 29(8):779–806, November 1998.