

# A Soft Computing Approach for the Design of Novel Pharmaceuticals

Robert Kewley, Mark J. Embrechts, *Member, IEEE*, and Curt Breneman

**Abstract** -- The pace of technological advancement in today's society has generated an enormous demand for methods facilitating the intelligent design of new pharmaceuticals, chemical compounds, and materials. This paper's authors have developed computationally intelligent data mining and molecular modeling technologies for the automated design and understanding of complex molecular structures. The Transferable Atom Equivalent methodology for calculating derived molecular properties generates a large set of potential predictors for a set of molecules. Novel neural network based "data strip mining" techniques extract predictive models from this set. These models may be used to screen candidate pharmaceuticals prior to expensive and time-consuming laboratory testing.

**Index Terms** — Neural Networks, Data Mining, Pharmaceuticals, Molecules, Bootstrapping, Sensitivity Analysis, Data Strip Mining

## I. INTRODUCTION

The aim of synthetic pharmaceutical design is to target a select number of novel candidate molecules with desirable properties. In a traditional framework, the development of a new compound, such as a pharmaceutical drug, requires a lengthy, costly and laborious discovery regimen in addition to a rigorous development and testing process. A model to predict certain pharmaceutical properties from a benchmark database with known properties can streamline this process. The Transferable Atom Equivalent methodology developed by one of the authors [1] provides a way to quickly and inexpensively generate a large set of predictive properties. "Data strip mining" provides a method to extract useful predictive models from this data set, even though there are few observations of the predicted property. These methodologies are combined to produce excellent results on two challenging data sets.

## II. MODELING ELECTRON DENSITIES TO GENERATE PREDICTIVE DATA

The Transferable Atom Equivalent (TAE) methodology relies on creating computer models of new molecules from a set of atomic building blocks. When these blocks of electron density are combined and allowed to fit into the

environment of a new molecule, a highly descriptive picture of the molecule and its properties emerges. Just as individual human beings change as surroundings and associates change, so too with atoms. The carbon atoms in diamond, in graphite, or in polyethylene are actually all different atom types. Likewise, groups of atoms in one molecule don't behave the same way as the same atoms do in another molecule where their environment is different. The same is true with other kinds of atoms such as nitrogen or oxygen. As a consequence, using traditional means, it has been difficult to accurately compute interesting molecular characteristics of even small molecules, and virtually impossible to do so with complex molecules of possible pharmacological importance.

Changes in the behavior of atoms (and, therefore the performance of a molecule) are not beyond computational prediction. Through the TAE method, which accounts for an immense range of possible influences on electron density, one can describe with remarkable speed and accuracy what will result when specific atoms and groups are combined to form a complex new molecule. An even more important aspect of TAE modeling is that the pharmaceutical activity of a molecule can be broken down into atomic or regional contributions. The information obtained in this way can be used to design new molecules with a tailored set of desirable properties. During the past decade, there has been increasing interest in applying molecular modeling methods to the design of new pharmaceutical agents and polymeric materials. One of the major challenges in this area has been to find ways of rapidly computing the characteristics of the electron density cloud, which surrounds all molecules. The reason why this is so important is that the distribution of electron density is a rich source of information about how that molecule will interact with other molecules. The exquisite specificity of some kinds of intermolecular interactions, "molecular recognition," is what makes enzymes and other catalysts work, and is fundamental to life itself. Pharmaceutical agents work by enhancing or blocking these recognition events, depending upon the desired outcome of the therapy.

The importance of gaining an understanding of the subtle factors which cause one molecule to bind strongly to another molecule while virtually ignoring all others is at the root of the quest for accurate molecular modeling methods. Aside from applications in the health sciences, an understanding of such interactions allows the designers of polymers to work smarter as well. The characteristics of

---

Robert Kewley is an instructor in the Department of Systems Engineering at the United States Military Academy.

bulk materials are also governed by how their constituent molecules interact with each other. Until the advent of computer-assisted modeling methods, the only way that new compounds could be discovered was by analogy, insight and serendipity. These tools have served us well in the past, but it is now possible to accelerate the discovery process by using methods that concentrate on the analysis of molecular electron densities.

Once the descriptors have been determined through the TAE method, and a predictive model has been built, thousands of new potential molecules, chemically similar to those of the benchmark data set, are scanned from large databases and are evaluated for their chemical properties based on the predictive model. The aim is to target a few novel molecules with potentially attractive pharmaceutical properties that can then be tested further in the traditional way in the laboratory. Neural network based data mining techniques help extract information used to select these novel molecules.

### III. NEURAL NETWORKS FOR DATA STRIP MINING

**Data mining** is the automated discovery of non-obvious, novel, and potentially useful information from large databases. This paper's authors define the **standard data mining problem** as a multivariate regression or classification problem for which the analyst has many candidate inputs or **descriptors** from which to choose. As the number of potential inputs to the model becomes large, and as the number of data observations becomes small, the problem of mining pharmaceutical data takes on an additional challenge. For most modeling methodologies, there are no longer enough data points to support the number of inputs. The most obvious solution would be to get more data, but this is not always so simple. Experiments may be very expensive, or the conditions under which the data was taken may no longer be available. In this case, the modeler must get as much information as possible out of the data and descriptors available. The problem is pivoted in the sense that there are more descriptors than data points to support them. The authors of this paper term modeling this type of pivoted problem **data strip mining** [2]. The procedure depends on two subordinate techniques, **neural sensitivity analysis** and **model bootstrapping**. These techniques may be combined to reduce the number of inputs to the model and improve its predictive capabilities at the same time.

#### A. Neural Sensitivity Analysis

**Neural sensitivity analysis** is the analysis of the system responses generated in the output layer of a neural network by changing the inputs (descriptors) going into the input layer. Recent neural network developments have shown how weight analysis and sensitivity analysis on a trained neural network can be efficiently applied to determine the

most important sensor variables as an alternative to a statistical factor analysis [3].

This paper proposes the following methodology for performing neural sensitivity analysis. Once a network has been trained on a large set of input variables, calculate an average value across all data points for each input descriptor. Then, holding all descriptors but one at a time at their average levels, vary the one input over its entire range and compute the variability produced in the net outputs. Analysis of this variability may be done for several different networks, each trained from a different weight initialization. The algorithm will then rank the variables from highest to lowest according to the mean variability produced in the output.

This sensitivity analysis procedure has two parameters. The first is the number of levels at which to evaluate each input during the analysis. All inputs but one will be held at their average values, and the net will be evaluated with this input at  $l$  values. If five values are selected, these values will be 0.0, 0.25, 0.5, 0.75, and 1.0. If only two levels are used, they will be the extreme points. The addition of other levels allows consideration of variance over the entire range of values. The analysis in this paper is done with  $l = 6$ .

The second parameter for this procedure is the number of sensitivity runs  $r$ . This is the number of differently initialized trained nets on which to perform sensitivity analysis. Different random initializations of a network will yield different sensitivity levels for the variables. As the number of networks used increases, so does the confidence level in the rankings. If these rankings are to be used to eliminate variables from the model, a higher degree of confidence will ensure that those variables with the smallest effect on the output are eliminated. This leads to better decisions in the variable reduction. The analysis in this paper is done with  $r = 5$ .

#### B. Model Bootstrapping

**Model bootstrapping** allows simultaneous improvement in the mean and variance of the estimate of a model's predictive ability using small data sets. As long as the net structure has enough complexity, a neural net can be trained to produce any desirable error level on the training set. Therefore, any measure of this error is a poor criterion for the ability of the net to predict outputs for data points it has not seen before. In order to determine a net's ability to generalize, it must be evaluated on a validation dataset which was not used during the training. If the number of data points is large, it is possible to have a large number of data points in the validation set and a good estimate of the mean prediction error. However, if the number of data points is small, the estimate of the mean prediction error produced by evaluation on the validation set will be highly variable. Resampling from this small data set allows better estimates of prediction error. Neural bootstrapping,

motivated by jackknife [4] and bootstrap [5] techniques, is used to break the data into many combinations of validation and training sets. For each combination, the net trains using the data in the training set to a certain error level then generates observations of prediction error using data in the validation set. By this method, a large number of prediction error observations may be taken from a small data set. This allows statistical statements about the mean and variance of prediction errors to be made with greater confidence.

The value of neural bootstrapping is demonstrated on pharmaceutical a data set which included 66 data points, each of them molecules. Each molecule had 480 descriptors as potential inputs to the model. It also had a target predicted value for concentration required to inhibit binding with the Cholecystokinin (CCK) molecule in the human blood stream. The analyst would like to determine the ability of two different network structures to predict this value. The first structure, after some variable removal, contains 103 variables. The second structure contains 12 variables. Training one network on 44 variables, leaving 22 variables in the validation set, yields the results in figures 1 and 2. As performance measure we will define  $Q^2$  (Eq. 1), where a lower  $Q^2$  means better performance. One can see that the 12 variable network seems to offer better predictive performance. The one-tail t-test for difference in mean prediction error (table 1) only allows us to conclude that the 12 variable structure produces smaller mean prediction errors with at a 0.12 level of significance. This is not strong evidence of better performance.

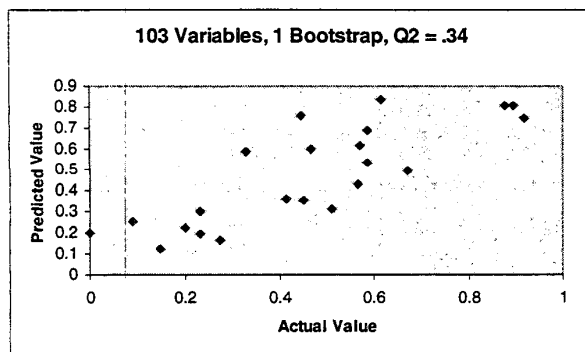


Figure 1. Plot of predictions by 103-variable network evaluated on one validation set of 22 data points, after training on 44 data points.

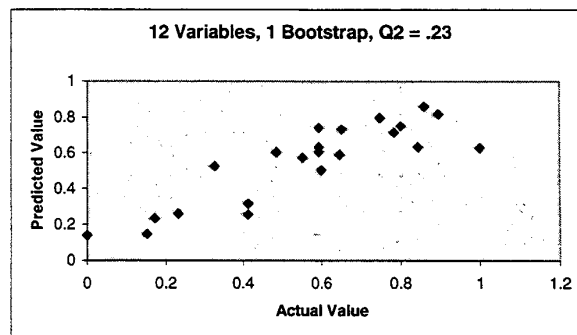


Figure 2. Plot of predictions by 12-variable network evaluated on one validation set of 22 data points, after training on 44 data points.

Table 1. Results of one tail t-test of the hypothesis that the 12 variable network produced smaller mean prediction errors than the 103-variable network using the results of one training session. We may only conclude the hypothesis is true with a 0.12 level of significance.

	103 Vars	12 Vars
Mean	0.125418182	0.094959091
Variance	0.006479235	0.007386356
Observations	22	22
Hypothesized Mean Difference	0	
df	42	
t Stat	1.213274552	
P(T<=t) one-tail	0.115902195	

Bootstrap training on 33 networks, each with a different combination of 60 training points and 6 validation points yields the results in figures 3 and 4. Visual inspection of these graphs points to better performance with 12 variables. The one tail t-test on these results (table 2) allows us to conclude that the 12 variable structure produces smaller mean prediction errors with a 0.0006 level of significance. This is convincing evidence to choose the 12 variable structure. Further inspection of the data also shows that doing one bootstrap produced a mean prediction error of 0.095 for the 12 variable set. Doing 33 bootstraps produced a mean prediction error of 0.069. The inclusion of 60 data points in each training set, as opposed to 44, allowed better prediction on the validation set. Therefore, neural bootstrapping has allowed us to enjoy the benefits of smaller estimates of mean prediction error and greater statistical confidence in detecting performance differences for different network structures.

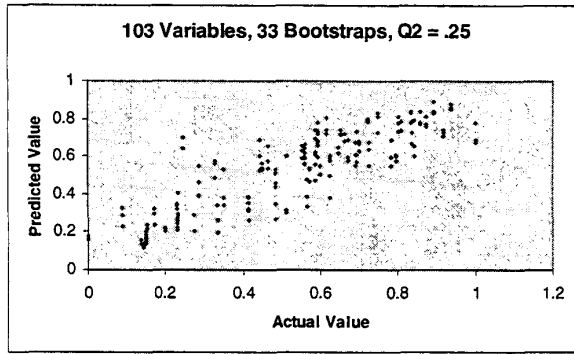


Figure 3. Plot of predictions by 103-variable network bootstrap evaluated on 33 validation sets of 6 data points, after training on 60 data points.

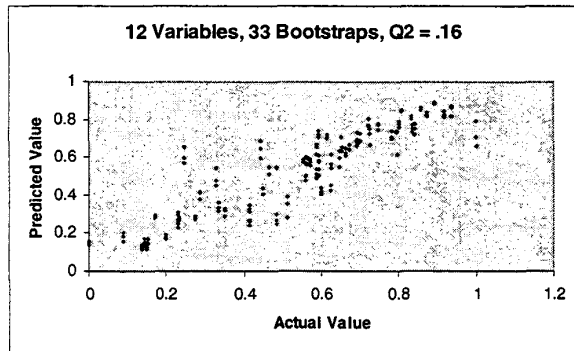


Figure 4. Plot of predictions by 12-variable network bootstrap evaluated on 33 validation sets of 6 data points, after training on 60 data points.

Table 2. Results of one tail t-test of the hypothesis that the 12 variable network produced smaller mean prediction errors than the 103-variable network using the results of 33 bootstraps. We may now conclude the hypothesis is true with a 0.0006 level of significance.

	103 Vars	12 Vars
Mean	0.094244444	0.068862626
Variance	0.006763167	0.00527881
Observations	198	198
Hypothesized Mean Difference	0	
df	388	
t Stat	3.254666013	
P(T<=t) one-tail	0.000617444	

In retrospect, doing traditional training on one network with one validation set would cause us to weakly recommend a 12 variable structure with an estimated mean prediction error of 0.095. Neural bootstrapping allows us to strongly recommend a 12 variable structure with a mean prediction error of 0.069. The users of this network will have much greater confidence in the network's predictive abilities given the evidence produced by bootstrap runs.

#### 1) Network Evaluation Criteria

Multiple regression models typically evaluate model adequacy by the adjusted coefficient of determination or the

mean square error in the model [6]. These statistics may not be calculated for strip mining models where the number of variables exceeds the number of data points. The coefficient of determination (without adjustment),  $R^2$ , corresponds to the square of the correlation coefficient between the actual and predicted values. This may be calculated regardless of the number of variables in the system. However, it should be estimated for a validation set, not used in model building, to determine a model's ability to predict. This can often lead to confusion since many model building methodologies evaluate  $R^2$  for the training set along the way. To ease this confusion, this paper will refer to a  $Q^2$  performance statistic, which is simply one minus the correlation coefficient squared evaluated on the **validation** set:

$$Q^2 = 1 - R^2 = 1 - \left( \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2} \sqrt{\sum(y-\bar{y})^2}} \right)^2 \quad (1)$$

This statistic will be used in conjunction with the bootstrap to evaluate models. Its advantages are that it eliminates the confusion between analysis on the validation and training sets and that it is not specific to one particular model building paradigm.

#### 2) Ideal Training Error Stop Point

In the conduct of these bootstrap sessions, the analyst must determine at what error level to stop training the net on the training set and evaluate it on the validation set. The authors call this value the **ideal training error stop point**. This stop point is the innovation which allows a network with many free parameters to train on a small data set and still give good prediction. Stopping training too early will not allow the network to fully learn the complexity required for prediction. Stopping training too late will allow the network to fit the output to the data so well that it no longer models the general trend in the output but fits the output curve to the individual data points exactly, even in the presence of data error. On-line observation of network errors on the validation and training sets confirm that test error decreases continually to a certain point, at which it begins to increase. The training error level at which the test error is minimized is the optimal training error stop point for this network. This point will vary for different splits of validation and training sets.

One could simply monitor test error on line, record the optimal point, and evaluate the network using that error level. However, if the goal of model evaluation is to estimate a network's ability to predict outputs for data it has not seen before, this scheme will bias that estimate toward an optimal test error for each split of the training and validation sets. A better method is to record the optimal test error for several different splits of the data set, calculate the average training error stop point, and use this for all of the splits. Then evaluate the network on different data splits than those used to calculate this error. This produces a better estimate of the net's ability to generalize.

A modification to this procedure would be to undertrain or overtrain the network by a certain amount. One could then calculate a standard deviation for the training error stop point, and adjust the average value by positive or negative multiples of this standard deviation. Some empirical evidence indicates that test error decreases more steeply as it approaches its optimum and then begins a gentle increase. Therefore, there is a smaller penalty for overtraining than undertraining.

#### IV. RESULTS

The sensitivity analysis and bootstrap procedures may be combined into a methodology which efficiently builds a very good predictive model with a minimum number of input variables. These techniques are applied with good results to two different molecular data sets to predict two different pharmaceutical properties. Generating multiple models with bootstrap training increases both the predictive capabilities of the model and the confidence in the estimate of those capabilities. Next, performing sensitivity analysis on multiple differently initialized neural networks increases the confidence in the rankings of the most sensitive variables. Finally, iterative elimination of descriptors, based on the sensitivity analysis, yields better predictive abilities for the model. These strong performance indicators are observed in both data sets.

The data strip mining methodology produced excellent prediction in the CCK data set. Figure 6 shows the increasing predictive ability of a network as excess predictors are removed from the input set. Prediction improves until the input set contains 12 variables. After this point, removing additional variables discards data necessary for prediction. This methodology also worked well for a much more challenging data set related to cancer prediction. The original 400 variable structure offered no predictive ability. However, stripping the data set of predictors allowed prediction to progressively improve to a point where  $Q^2 = 0.49$ , an indicator of moderate predictive ability. The improvement in  $Q^2$  as variables are removed is plotted in figure 5.

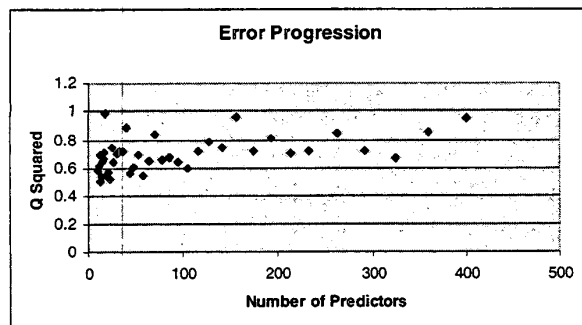


Figure 5. Note the improvement in  $Q^2$  for a data set related to cancer prediction as variables are removed from the network.

#### V. CONCLUSION

This paper demonstrated the combined power of generating molecular data with the TAE methodology and using data strip mining to extract predictive information from that data. The data strip mining methodology is used to solve data strip mining problems, which call for the development of a model from a data set containing too few observations to support the number of potential input variables. Despite a much larger number of free parameters than data points, calculation of an ideal training error stop point during backpropagation allowed the model to give good prediction. Neural sensitivity analysis was used as a method to determine the most important descriptors for the system, allowing some of the least sensitive descriptors to be eliminated. The authors used neural bootstrapping as a method to accurately evaluate different neural models of the same system when the number of data points from which to draw training set and validation set samples is very small. All of these methods are combined and automated in a data strip mining methodology. This methodology works well for two very difficult problems from pharmaceutical chemistry for which other methodologies have failed to provide sufficient performance.

#### VI. REFERENCES

- [1] Breneman, C.M., T. R. Thompson, M. Rhem, and M. Dung, "Electron Density Modeling of Large Systems Using the Transferable Atom Equivalent Method," *Computers & Chemistry*, (19), 161 (1995).
- [2] R. Kewley, M. J. Embrechts, and C. Breneman, "Neural network analysis for data strip mining problems," in *Intelligent Engineering Systems through Artificial Neural Networks*, (8), Cihan Dagli et al., eds., 391 - 396, ASME Press (1998).
- [3] Bigus, J.P. *Data Mining with Neural Networks*. McGraw Hill (1996).
- [4] Tukey, J. "Bias and confidence interval in not quite large samples," *Annals of Statistics*, (29), 614 (1958).
- [5] Efron, B. "Bootstrap methods: another look at the jackknife," *Annals of Statistics*, (4), 1-26 (1979).
- [6] J. Neter, M. Kutner, C. Nachtsheim, and W. Wasserman. *Applied Linear Regression Models*. Irwin (1996).

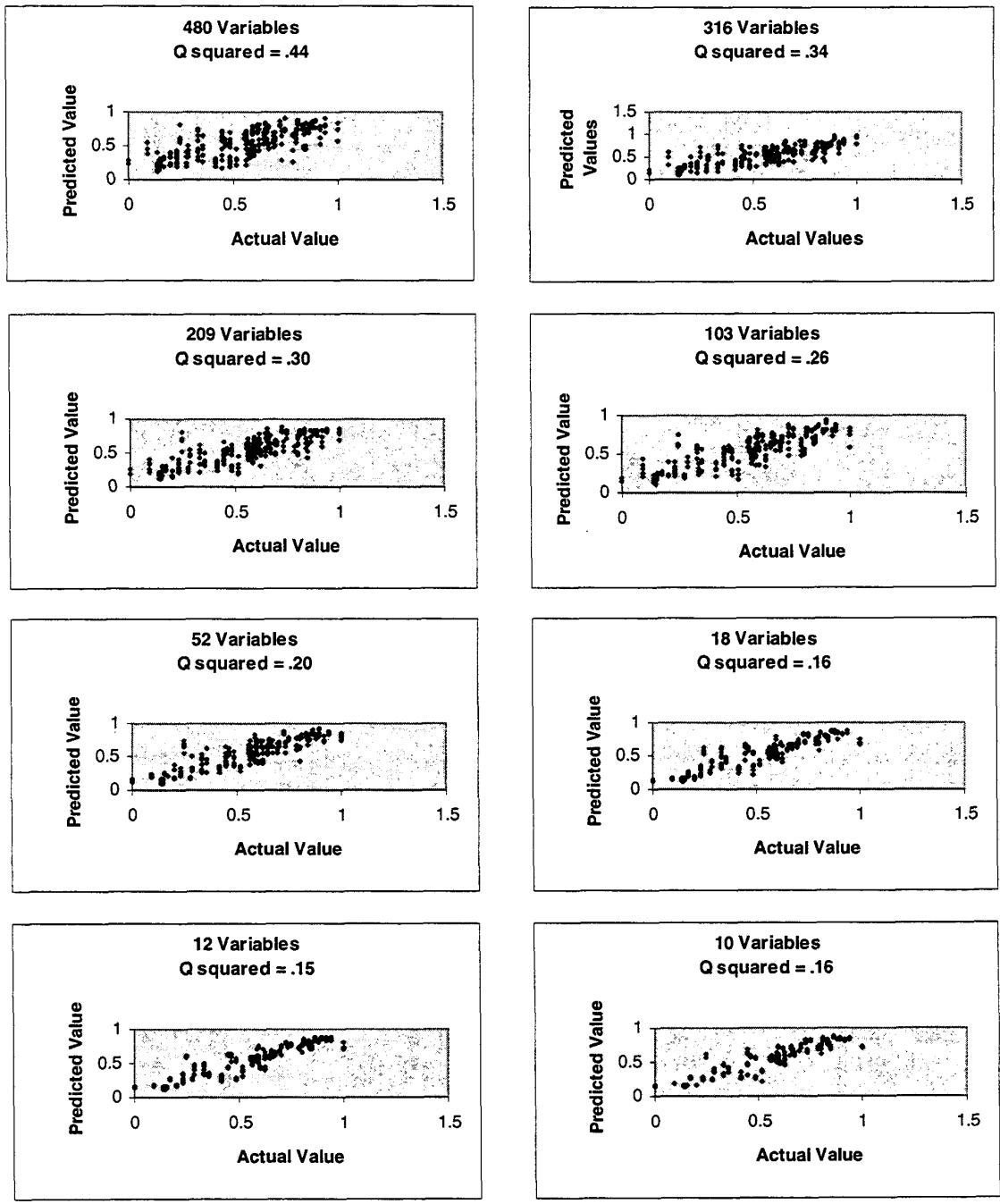


Figure 6. The progressive improvement of predictive models as predictor variables are removed from the CCK data set.