

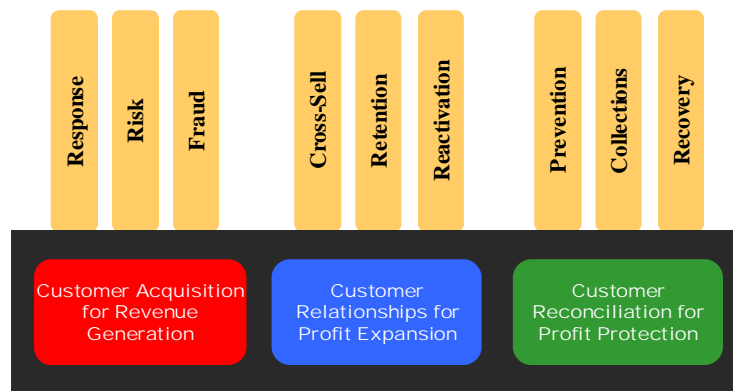
Building Next Generation Predictive Models Using Genetic Algorithm-based Software

GENALYTICS

300 Brickstone Square, Suite 701
Andover, MA 01810
Phone: 978.684.7580
Fax: 978.684.7585
Email: info@genalytics.com
www.genalytics.com

Executive Summary

Technological advances in computer storage have resulted in an enormous increase in the quantity of data being captured and stored by businesses. This data has become a key strategic asset to organizations, imperative to understanding behavior across the customer lifecycle. Analyzing the data for optimal business use has become a critical challenge to firms who are striving to stay ahead of the competition. Whether acquiring new customers, maintaining existing relationships or targeting potentially delinquent customers, predictive modeling is the technique of choice for business analysts and statisticians, used to mountains of data into potential revenue.



Predictive modeling improves performance across the customer lifecycle

Genetic algorithms allow optimized models to be built in far less time than traditional regression analysis techniques, leaving more time for analysts to spend on important tasks like problem definition and interpretation of results.

Standard predictive modeling capabilities using techniques like linear and logistic regression, however, have inherent limitations. They are so labor-intensive that analysts cannot keep up with business demands for data mining; nor has the number of analysts kept pace with the explosive growth of this data. As a result, organizations are exploring new options to address these limitations. Only recently has computing power become inexpensive enough to make machine-learning technologies such as genetic algorithms an economically viable approach to solving this problem. Genetic algorithms allow optimized models to be built in far less time than traditional regression analysis techniques, leaving more time for analysts to spend on important tasks like problem definition and interpretation of results.

By exploring all of the data, it logically follows that GPS will produce better, more predictive models.

Genalytix, Inc. has been applying genetic algorithm-based technologies to predictive modeling since 1998. Genalytix Predictive Suite (GPS), Genalytix' premier software solution, is based on patented genetic algorithms that examine all variables in a data set and through a "trial and error" process, finding the optimal solution to an analytical problem. By exploring all of the data, it logically follows that GPS will produce better, more predictive models. The key benefits are as follows.

Better Modeling Lift

Genalytix' software consistently generates predictive models with 7.5 to 15% improvements in key success factors like response rate or collections.

- The genetic algorithm approach provides better insights into existing datasets by thoroughly exploring 100% of the available variables rather than an analyst-selected subset.
- Models can be updated on a daily basis to reflect the most recent customer or market data.
- The underlying algorithms explicitly optimize results for a specific business problem.

Shorter Development Times

- Genalytix' software typically reduces model development time by 50 to 75% through automated data preparation and exploration.
- GPS software works seamlessly with existing statistical analysis platforms like SAS or SPSS, enabling analytical teams to leverage existing investments without radically changing the way they do business.
- Many analytical teams eliminate predictive modeling backlog and increase the speed to profitable actions by using GPS.

More Time to Discover New Opportunities

Using GPS, analysts have more time to develop new insights and discover innovative market opportunities since the software reduces manual data preparation steps and provides more time for problem definition and model interpretation.

The Conventional Approach: Regression Analysis

The traditional data model development process is very time-consuming. Unfortunately, delays in building models can negatively impact the business in terms of lost business opportunities or less-than-optimal results when the process is hurried. **Figure 1** below presents a “generic” model development lifecycle using traditional regression methodology:

Traditional Regression-Based Approach

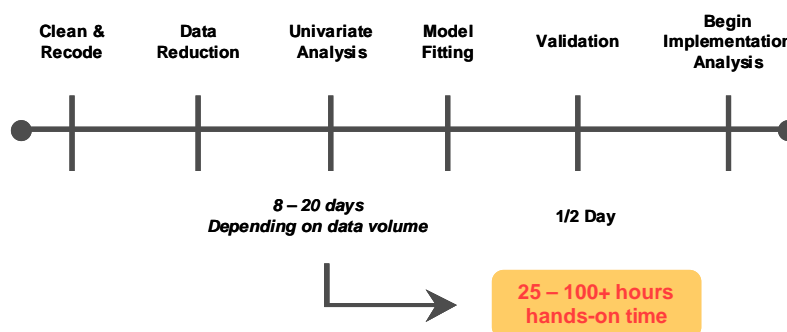


Figure 1. Traditional modeling process

Depending upon the nature of the business problem and the complexity of the data, model development can take four or more weeks to complete and may require more than 100 hours of hands-on statistician time. It is a highly iterative, labor-intensive process. The bulk of the effort (approximately 50 to 75%) is spent on basic tasks like preparing the data, reducing the variables for consideration, selecting variables, and exploring the data. The list below provides a high-level description of these manual tasks:

- *Clean & Recode*: cleanse and sample data; transform variables; substitute missing values; address outliers and sparse data; bin continuous values; convert categorical data to modeling format
- *Data reduction*: review covariance and classes of variables to reduce redundancy for a manageable set of 30-50 elements
- *Univariate analysis*: look for the strongest correlation with the dependent variable, further reducing the number of variables considered for regression
- *Model fitting*: select subsets of variables from previous steps and execute SAS regressions (forward, backward, stepwise); manually review final coefficients and determine incremental impact of adding or dropping variables from equation or adjusting coefficients

By combining the traditional statistical modeling approach with machine-driven genetic algorithms, one can rapidly deploy robust, predictive models that result in significant business impact.

This effort takes the statistician away from the critical challenges inherent in defining the business problem, interpreting the results or implementing the model for business benefit.

A Better Approach: The Genetic Algorithm Method

By combining the traditional statistical modeling approach with machine-driven genetic algorithms, one can rapidly deploy robust, predictive models that result in significant business impact.

This alternative strategy combines the traditional approach with automation from Genalytix's software, Genalytix Predictive Suite (GPS). Adopting this approach, statisticians have reduced risk and improved response by 7.5% to 15%, while decreasing skilled statistician time and effort by 80%. These savings have given statisticians time to explore other aspects of customer data to gain new business insights. **Figure 2** below illustrates the potential savings in development time with the GPS approach:

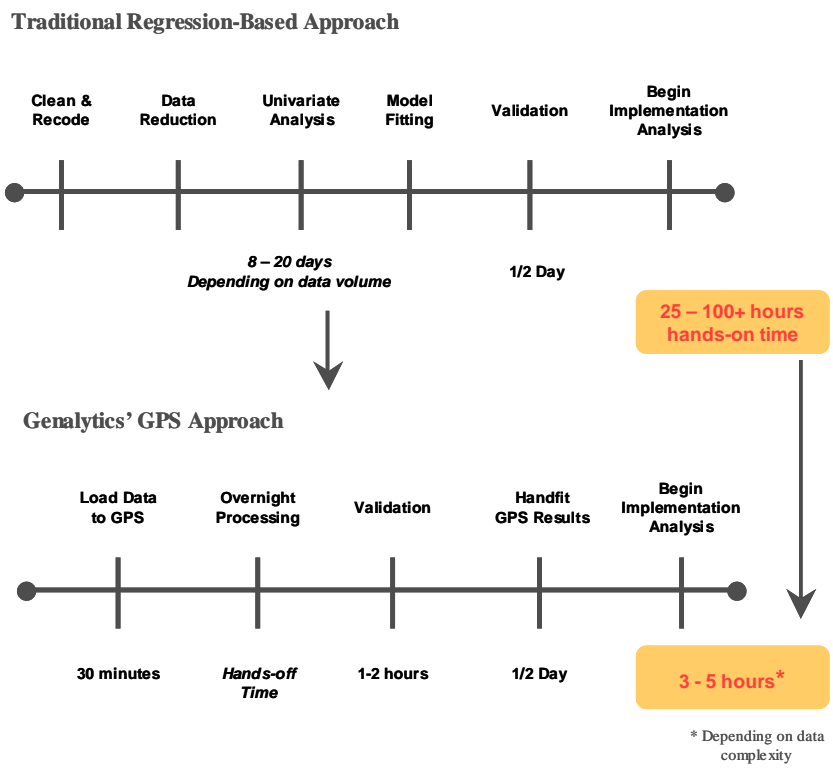


Figure 2. Time Line Using Genetic Algorithms

Genetic algorithms have the following advantages over other predictive modeling applications:

- They are robust, assumption-free, and perform well on large and small samples
- They can evaluate far more combinations of data attributes than regression-based models
- They allow for fine-tuning optimal responders into top or bottom deciles based on unique fitness functions
- They can run thousands of iterations of model builds to develop an optimal result for champion/challenger testing
- They hold up over longer periods of time before decay sets in
- They offer clear, understandable results

GPS MODEL™ is a predictive intelligence application that dynamically generates highly optimized models. By using patented genetic algorithms, GPS MODEL can evaluate more combinations and patterns of data than other products on the market today.

Breeding Better Solutions

The application of genetic algorithms to predictive modeling is based on Darwin's principle of "survival of the fittest". The genetic algorithm will breed an initial "generation" of random models; each model is then tested for fitness against user-defined criteria. The best models are more likely to be selected for breeding and will survive while the weaker models die out. By applying the basic genetic principles of cloning, mating and mutation, the genetic algorithm increases the diversity of models evaluated. This search of possible outcomes is repeated for thousands or even tens of thousands of generations, with the best models surviving. Ultimately, the more "evolved" model "wins" and represents the best solution to the business analytic problem. The remainder of this section describes genetic algorithms and the evolutionary process.

Genetic Encoding

Begin by thinking of the business problem as being "genetically encoded" into 1's and 0's. It all begins at the gene level. Envision a series of genes, or gene groups, representing everything about an independent (predictive) variable based upon its type:

- Continuous variables are represented as genes for selection, coefficients, outlier trimming, transformation and normalization
- Categorical variables are represented as genes for selection, contrasts for value combination, and coefficients
- Data variables are represented as genes for selection, transformation (duration) and coefficients

Begin by thinking of the business problem as being "genetically encoded" into 1's and 0's.

- Interaction variables are represented as genes for selection, interaction operator and coefficients

Figure 3 shows an example of a continuous variable, which has selection, transformation, outlier and coefficient genes.

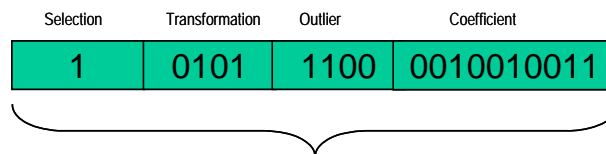


Figure 3. Continuous variable gene group example

Each chromosome represents a model equation or potential solutions to the business problem.

Genes are combined to form a chromosome. Chromosomes include gene groups for all variables on the input dataset, along with genes for interactions that are a combination of variables based upon different operators. Each chromosome represents a model equation or potential solution to the business problem. **Figure 4** illustrates the relationship between gene groups (variables) and a model chromosome:

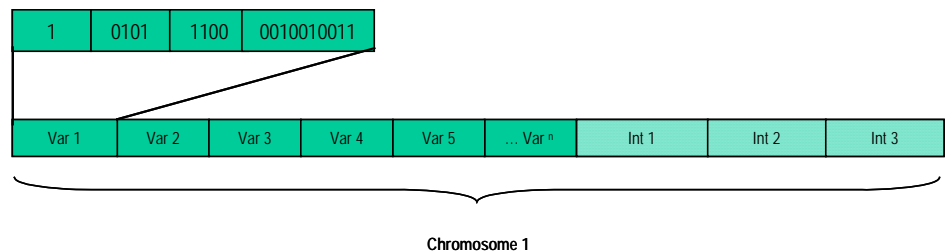


Figure 4. Chromosome is made up of gene groups for all variables and interactions

The genetic algorithm will adjust the variables selected at random as a means of ensuring stability and avoiding the potential of overfitting of solutions.

In addition to gene groups for each variable, the genetic algorithm automatically appends genes for interaction variables based on chromosome configuration parameters. These new variables help the genetic algorithm discover subtle interactions that greatly enhance the overall model performance.

While the chromosome contains genes for all available variables, the genetic algorithm will adjust the variables selected at random as a means of ensuring stability and avoiding the potential for overfitting of solutions. The modelers can use the variable limit defaults or apply their own criteria. **Figure 5** illustrates the “model” chromosome, identifying those variables that are part of the model equation.



Figure 5. Each model chromosome uses only selected variables

How The Evolutionary Process Works

Once the basic structure of the chromosomes is complete, the genetic algorithm will create an initial population of models, randomly populating them with 1’s and 0’s. The default initial population will contain 100 chromosomes. Once this initial population is created, each model is tested for fitness based on user selected fitness metrics, which include both binary and continuous metrics. The “most fit” models are more likely to survive and be selected for breeding. The probability of mating is based on the models fitness. Imagine a roulette wheel with each model assigned a portion based on a model’s proportion of total fitness. **Figure 6** illustrates a model fitness table and corresponding “roulette wheel”:

Model	Fitness	% Total
1	0.20	9%
2	0.32	14%
3	0.45	19%
4	0.62	26%
5	0.75	32%

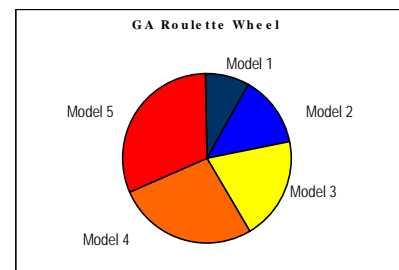


Figure 6. Genetic Algorithm “Roulette Wheel”

Note that Model 5 has the highest fitness and, therefore, has the largest portion of the “roulette wheel”. As the genetic algorithm selects models for mating, those with the higher fitness are more likely to be selected. Models 4 and 5, for example, represent almost 60% of total fitness and are the best models to date.

Following fitness evaluation, the genetic algorithm will apply the basic principles of genetics to breed the next generation of models. These consist of cloning, mating, mutating, breeding and introducing viruses.

Cloning is the process of copying a chromosome from one generation to the next.

Cloning

Cloning is the process of copying a chromosome from one generation to the next. With the default option, the genetic algorithm clones the best model, the “king” model, to the next generation.

Mating

For mating, the genetic algorithm will select and combine pairs of models through the genetic process of “crossover”. In this process, portions of the first chromosome are combined with sections of the second chromosome to create a new pair of chromosomes for the next generation. The rate of crossover is controlled by the genetic algorithm and automatically adjusted based on learning, or lack thereof.

Figure 7 provides an example of the crossover process:

```

110101010100101010101101010101010101010...
00000101010111101001010010111010101001011...

... becomes

110101010100101010101101010101010101010...
00000101010110101010110101011010101001011...

```

Figure 7. Example of genetic crossover

Mutation

Following crossover, each of the new chromosomes is subjected to mutations. During mutation, each bit has a small chance of flipping from a 1 to a 0, or a 0 to a 1. As with crossover, the genetic algorithm controls the mutation rate and will adjust it as necessary. **Figure 8** illustrates the mutation process:

```

00000101010111101001010010111010101001011...

... becomes

00000101010101101001110110111011101000011

```

Figure 8. Example of genetic chromosome mutation

During the evolutionary process, the genetic algorithm will breed new generations of models through cloning, mating and mutation, and will also introduce viruses to help improve model lift.

Viruses

For each generation, a random set of models is introduced as viruses and may be selected for crossover during the subsequent generation process. Introduction of viruses helps enhance the overall model fitness and helps avoid “local optimum”. The genetic algorithm controls the virus rate and will adjust this rate to help the learning process.

Breeding

During the evolutionary process, the genetic algorithm will breed new generations of models through cloning, mating and mutation, and will also introduce viruses to help improve model lift. The resulting generation of models may contain a) one model chromosome that is the king model, b) 22 models created through crossover and mutation, and c) two random model chromosomes introduced as viruses. **Figure 9** below illustrates a potential composition of 25 chromosome models following breeding. The genetic algorithm controls the evolutionary parameters for crossover, mutation and viruses, and will adjust these parameters based on learning, or lack thereof.

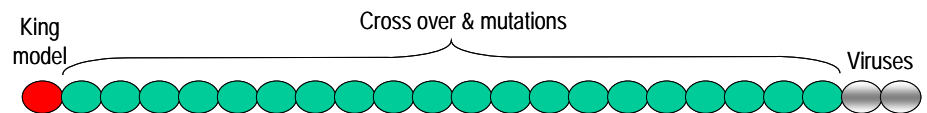


Figure 9. Result of generation breeding

Using the basic genetic processes—cloning, crossover, mutation and viruses—the genetic algorithm breeds new generations of models.

Putting it all Together

Using the basic genetic processes—cloning, crossover, mutation and viruses—the genetic algorithm breeds new generations of models. **Figure 10** illustrates this process. For each generation, the genetic algorithm evaluates each model against the user-defined fitness metric. The better models are more likely to be selected for breeding. The end result: a “king” model that represents the best-evolved model to address the business problem.

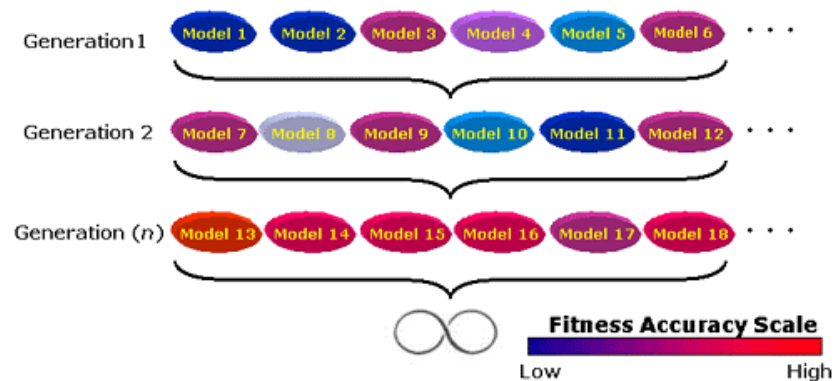


Figure 10. The evolutionary process

During this breeding process, the genetic algorithm applies different data manipulation techniques behind the scenes to help improve the model lift. These techniques, each of which is described in more detail in the following section, include:

- Outlier trimming
- Data transformations (e.g., square, square root) and normalizations (transforming continuous variables to values between 0 and 1 to account for different ranges across different variables)
- Continuous variable bucketing
- Categorical variable combinations
- Variable reduction
- Interaction detection
- Multiple fitness evaluation metrics

The Results

After the modeler has run for an adequate number of generations (typically around 5,000 to 10,000), the analyst has a series of standard tools at his or her disposal. Standard lift charts can help show how well the model performed against existing models. An equation is also produced for use in production scoring to evaluate outcomes.

Conclusion

In many cases, Genalytics' predictive modeling software outperforms linear or logistic regressions by 7.5 to 15 percent when predicting key performance metrics like response rate or risk assessment. This is because the software:

- Explores a broader range of outcomes and as a result can identify patterns undetectable by other modeling techniques
- Considers 100% of the available data rather than a subset of the variables selected by the analyst

In many cases, predictive modeling software based on genetic algorithms outperforms linear or logistic regressions by 7.5 to 15 percent.

- Automatically creates new virtual variables that are transformations or combinations of existing variables that expand data exploration
- Reduces model decay by updating models in days rather than weeks or months

The basic idea is to leverage the automation of the genetic algorithm for data mining, and then use the results from the genetic algorithm as input into final validation and model fitting.

A New Approach: Combining Traditional Techniques with Genetic Algorithms

This section redefines the model development lifecycle based on the combination of traditional techniques with the automated activities from GPS MODEL. The basic idea is to leverage the automation of the genetic algorithm for data mining, and then use the results from the genetic algorithm as input into final validation and model fitting.

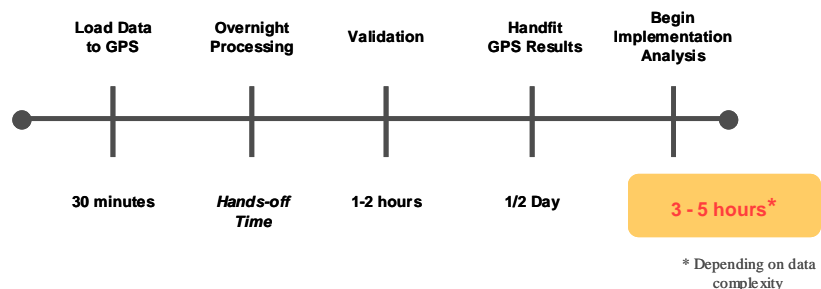


Figure 11. Traditional modeling process

Once the data is loaded and modeling parameters are set, the genetic algorithm will attempt to optimize the outcome of the model by considering **all** (as well as combinations of) attributes and data transformations. This optimization is at the core of the genetic algorithm approach, where millions of model parameters are evaluated in an “evolutionary” process. Tens, or even hundreds of thousands of generations of competing models, each passing results and discoveries on to the next, evolve into the best solution for the business problem.

Once the genetic algorithm is completed, statisticians can apply their expertise to produce a final model that is robust, explicable and that satisfies business and regulatory policies. Here, the statistician will input the variables from the genetic algorithm into other statistical procedures in SAS or SPSS.

The genetic algorithm automates many of the activities in the data mining process, most of which are time-consuming and mundane.

Genetic Algorithm Processing—Time-saving Techniques

One primary benefit of this alternative approach is that the genetic algorithm automates many of the activities in the data mining process, most of which are time-consuming and mundane. At a high level, the genetic algorithm approach is a two-step process: Data Load and Sampling, and Overnight Processing. The level of human control over this process can be tailored based on the business problem as well as statistical analyst preferences. The data and modeling parameters can be adjusted during the evolutionary process so that the analyst can explore different aspects of the data. The remainder of this section describes the genetic algorithm processing activities.

Data Load & Sampling

Regardless of the data mining technique, analysts must understand their data. While this is true with the genetic algorithm, it is not as critical since the genetic algorithm can process 100% of the data to find the best solutions.

To begin the modeling process, the analyst will load data directly into the genetic algorithm. Depending on the nature of the business problem, the analyst can apply business rules to subset the number of observations that are available for the sampling process. During sampling, the analyst identifies the appropriate percentages for training and validation sets as well as category cutoffs for alphanumeric and numeric data. For alphanumeric data, if a particular variable has more values than the cutoff, it is ignored. For numeric data, if a specific variable has more values than the cutoff, it is treated as continuous data; otherwise, it's treated as categorical numeric. The analyst has the ability to switch numeric usage from categorical to continuous.

The genetic algorithm sampling process will calculate different metadata for each variable depending on type—categorical or continuous. These metrics include MIN, MAX, MEDIAN, MEAN, % MISSING, # CATEGORIES, CATEGORY VALUES, PEARSON CORRELATION and CHI-SQUARE. This metadata provides an initial view into the data and a mechanism to ensure that the data was loaded properly.

The data loaded into the genetic algorithm may not be appropriate for the business problem at hand, either because of the nature of the data (sparsely populated or future information) or because it doesn't conform to the firm's business practices. Regardless, the analyst can drop variables from the search space to:

- Restrict the genetic algorithm to only those variables that would satisfy business policy reviews

- Soft start with only the variables that pass a statistical threshold (Chi-square or Pearson)
- Open the search space to all variables, with final reduction of variables completed in the hands-on validation step following overnight processing
- Force or restrict the genetic algorithm from evaluating different variables from an R&D perspective

In most cases, a variation of these is used and even modified during processing to help gain additional insight into the data.

Once the data is loaded and sampled, it's time for the statistical analyst to define the genetic algorithm modeling parameters.

Overnight Processing

Once the data is loaded and sampled, it's time for the statistical analyst to define the genetic algorithm modeling parameters. These include:

Population: defines how the initial population of chromosomes is created. Initial population options include:

- Random
- Distributed, where two models are built for every variable in the dataset with +1 and -1 coefficients; after fitness evaluation, the best N models are used for generation 0 and the breeding begins

Fitness: defines the genetic algorithm fitness function the genetic algorithm uses to evaluate the models; as a general practice, analysts typically start several models with different fitness functions

Interactions: defines the search space on the chromosome for the genetic algorithm to create and evaluate interactions; the larger the search space, the more likely interactions will enter into the model

Data transformations: identifies whether data transformations are to be applied to variables

Variable limits: limits (by default) the number of variables allowed in the model; the analyst can override this depending on the business objective

Noticeably missing are controls for the evolutionary process (crossover, mutation and virus rates). The genetic algorithm starts with the default values defined within the configuration properties. Unique to GPS MODEL, these parameters are automatically adjusted by the genetic algorithm.

After these modeling parameters are set the model can be started, and the genetic algorithm is left to "do its thing". Occasional monitoring of the evolutionary process is suggested to ensure that no "future" or restricted information is in the model. Nothing is worse than letting a model run overnight, only to find it has created a "perfect" model with one or two variables that should have been eliminated.

While processing, the genetic algorithm will automatically apply different data preparation functions against the data and evaluate the results against the fitness function.

While processing, the genetic algorithm will automatically apply different data preparation functions against the data (so the statistician doesn't have to) and evaluate the results against the fitness function.

Missing Value Substitution

Virtually all datasets have missing data. Regardless of the approach, the statistician must evaluate the data and select the appropriate value to substitute for missing values. The genetic algorithm will automatically handle missing values, both for continuous and categorical data.

For continuous variables, the genetic algorithm will automatically substitute the computed median value (default). The statistician can override the median either globally or at the variable level, depending upon the business problem. For categorical data, the genetic algorithm will treat missing values as a separate category with the statistician having options to restrict the genetic algorithm from combining missing values with other values.

Outlier Trimming

Extreme values in a dataset can badly skew the resulting models. Traditionally, analysts have used their own judgment in handling these outlying values. Genalytics' software can automatically calculate optimal outlier trimming values for each variable based on improvement in the model's overall fitness.

Data Transformations

Traditional modeling requires that data be normally distributed. A substantial manual effort can be spent in transforming data to fit the business problem and traditional techniques. Unfortunately, analysts often search through a small number of variables and transformations. This is especially true of datasets with a larger number of variables.

The genetic algorithm will automatically apply different transformations against continuous data and evaluate the result against the overall model fitness. These transformations include square, square root, exp, log, and normalizations. Normalizations transform the data to values between 0 and 1 based on the relationship of that observation to the overall range of raw data values.

Continuous Variable Bucketing

Numeric data can be treated as either categorical or continuous. When treated as continuous, the genetic algorithm will define and evaluate different coefficients that are applied to the raw value. This works well when there is a linear relationship between the continuous variable and dependent variable. However, when there is a non-linear relationship, the genetic algorithm can automatically bucket continuous values into different ranges of values and assign different coefficients for each of

these buckets. The modeler has the option to turn on bucketing for individual variables.

Categorical Variable Representation and Combinations

This is another time-consuming process in traditional statistical modeling. Besides identifying hundreds or thousands of categorical variables, the analyst must examine the composition of each category and determine appropriate “binning,” or combinations of categories. An analyst may choose, for example, to have the categories of “some college” and “college graduate” combined based on their correlation to the dependent variable.

Besides the time-consuming manual manipulation of categorical data, there is a subtler problem with this traditional approach. Category combinations are often defined prior to the modeling process based on simple correlations with the dependent variable. A better place to calculate the optimal binning is in the modeling step where other variables are competing to explain the relationship. Back to our example, “some college” and “college graduate” may appear in a simple correlation to act as the same category, but after “age” has entered into the equation, the residuals may reveal these are best treated as two distinct categories.

The genetic algorithm software automatically detects categorical variables and dynamically bins categories in the model-building step. The analyst controls key parameters such as how thin the data in a particular category can be and still be allowed to exist as a standalone category, whether a value must combine with a “next nearest neighbor”, or whether an extreme value must remain un-combined.

Interactions

Similar to data transformations, a traditional modeler will put substantial effort into discovering and evaluating interactions between variables to fit the predictive relationship. Since this process can take a significant amount of time, the modeler will usually search through a small subset of variables for possible interactions. For example, a dataset with 2,000 variables has approximately one million possible two-way interactions.

Within the genetic algorithm, the modeler can define the search space available for interactions. This search space will contain interaction gene types with genes for selection, coefficient and interaction operator. The genetic algorithm will automatically define and evaluate different interactions. Depending on how aggressively the analyst directs the genetic algorithm to search for interactions, the vast majority of possible interactions may be explored.

(Of course, there is no point in “reinventing the wheel”. Should the user know the key interactions (e.g., line of credit utilization), they should be created prior to the

The genetic algorithm software automatically detects categorical variables and dynamically bins categories in the model-building step.

modeling process and made available for consideration by the genetic algorithm during the evolutionary process.)

Variable Reduction and Coefficients

In traditional statistical modeling, this is the easy part of the process. Having spent many days or weeks preparing the data, the remaining variables (typically between 20 and 30) are input into regression procedures to determine the final variable selection and coefficients. After building the initial model, the analyst iterates through the process, changing transformations or adding/removing variables from the equation. This process continues until a satisfactory model is produced, or the time available to build the model expires.

The analyst can control the number of variables for a model or use the default values.

Conversely, the genetic algorithm generates superior results by using a different methodology. During the breeding process, the genetic algorithm will automatically assign different coefficients for a variable and will assign either a 0 or 1 to the selection gene. If 1, the variable is part of the model. If 0, it is not. The analyst can control the number of variables for a model or use the genetic algorithm default values. To help ensure more robust models, the genetic algorithm will always opt for a model that has fewer variables (assuming identical fitness measures).

Multiple Fitness Evaluation Functions

The genetic algorithm has multiple fitness functions, both for binary and continuous outcomes. In addition to the traditional fitness functions, the analyst can fine-tune the outcome based on unique fitness functions using the genetic algorithm function called *lift*. Using the lift function, the genetic algorithm rewards itself based upon user-selected fitness values for each quantile in the output reporting set. **Figure 12** illustrates three lift functions:

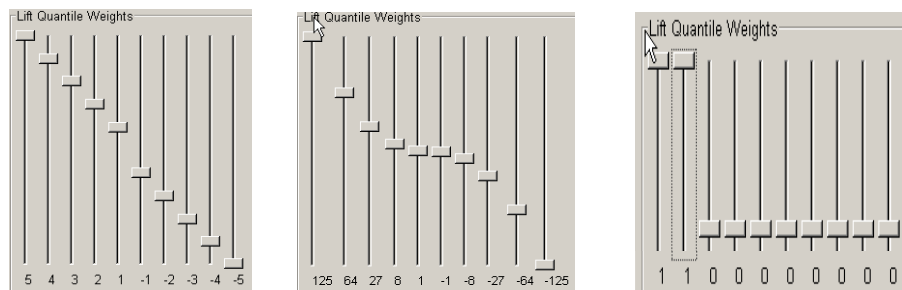


Figure 12. Genetic Algorithm LIFT functions: Linear, Cubes and Upper 20%

Applying a Linear Lift function to a binary model, the genetic algorithm would assign 5 points for every event in the top decile, 4 points for every event in the second decile, and -5 points for every event in the bottom decile. The overall fitness for this model would be the sum of all weighted events. Each of the three lift function examples above would be applied in the same manner.

Since the genetic algorithm processes data very quickly, analysts will typically start several models with different fitness metrics focusing on different aspects of the customer file.

In addition to the default lift fitness functions, the statistical analyst can use the lift slide bars and adjust the weights based on the specific business challenge. For example, if the marketing organization wanted to mail up to 30% of the mail file, but the final percentage wouldn't be determined until after the model was executed, the lift function could be: 125, 120, 115, 30, 25, 20, 15, 10, 5, 0. The genetic algorithm would focus its effort in moving events from the bottom seven deciles into the top three, with a slightly linear relationship between the deciles within the two major groups.

Since the genetic algorithm processes data very quickly, analysts will typically start several models with different fitness metrics focusing on different aspects of the customer file. Then, after all models are complete, the analyst will use the superset of variables in the subsequent model validation and fitting activities.

Proprietary Algorithms

The automated processing identified above is only part of the story. There are several other proprietary methods that the genetic algorithm uses to help avoid *overfitting* and *local optimum*. *Overfitting* can occur when a predictive modeling technique learns so much from the training dataset that it does not hold up during validation. *Local optimum* can occur when continued breeding with better models does not improve upon the best solution to date. The end result is a model that is robust and holds up over time.

Hands-on Validation & Final Model Fitting

Now that the genetic algorithm has completed and the time-consuming and mundane work is done, the statistical analysts can apply their expertise. While the time required for this step is significantly reduced, it is still the critical step in model development. The final model must be one that makes sense and satisfies all business objectives and policy requirements.

The analyst reviews the variables identified in the genetic algorithm model(s) to ensure that no future or restricted variables were allowed to enter the genetic algorithm search space, and that the variables entering the model and data transformations applied are explicable. The analyst may run a few correlation or frequency reports for validation, and may adjust different data transformations or totally eliminate variables from further consideration.

For those variables that remain, the statistical analyst then inputs them into SAS or SPSS statistical procedures to generate the final model.

Results/Summary

The process described here combining traditional methods with genetic algorithms results in a robust model built in a fraction of the time that passes statistician,

business objective and policy reviews. The following section presents various case studies where this alternative approach was applied to generate models with improved lift in a few days, versus existing models built the traditional way that required weeks or even months to create.

CASE STUDIES – The New Approach at Work

In general, clients built models with 10% to 20% improvement in marketing lift or reduction in credit risk.

The case studies below are for illustrative purposes, with the client names and actual benefits generalized for confidentiality. In general, clients built models with 10% to 20% improvement in marketing lift or reduction in credit risk, with business benefit measured in terms of increased marketing revenue, decreased credit losses, and reduced operational expenses. While the benefit is dependent upon the nature of business challenges, these clients typically achieved multi-million dollar improvements.

In many situations, the genetic algorithm was used for R&D to search for new variables that existed within their current environment, or new data sources that statisticians were not familiar with or didn't have time to evaluate (or both). These data sources included personal demographics, credit bureaus, and the U.S. Census. One goal of this R&D was to find nuggets of data to improve model results or to segment populations for further modeling.

Each case study outlines the business challenge addressed as well as the business value attained.

Case #1: Front-end Risk

Business Challenge

To improve risk assessment capabilities, this team sought new analytic methods to decrease risk and quickly determine the value of segmenting risk based on new, powerful indicators. The current market stage risk model did not measure the impact of adding these new indicators to segment risk. The goal was to attain a 5% reduction in risk and/or 5% increase in mail volume.

Business Value

Improved performance: exceeded expectations, achieving 10% reduction in risk – twice the project goal

Immediate results: initial model results were produced in hours

Increased productivity: dramatically shortened analysis cycle with final model produced within 24 hours; internal resources expected to take one to two months to complete project

Accelerated learning: new data discoveries validated segmentation approach and revealed new risk indicators that greatly contributed to the improved results; in addition, served as filter in identifying variables for further investigation

Results

The resulting model is robust in a lower-risk segment. Even without the demographic indicators, the model reduced losses by 15% through the sixth decile.

Case #2: Next-best Product

Business Challenge

Credit cards are marketed with different reward programs or affiliations. The marketing objective is to better understand a prospect's preference for one product versus the other so that the appropriate offer can be made. The approach: build a matrix of scores where each individual will have an overall likelihood of response score, along with a separate score for each product. The highest product score represents the next-best product recommendation for that prospect. Prospects are selected for individual marketing efforts, using business rules and optimization techniques that can factor in logistic and economic constraints, along with business policies.

To build this matrix of scores requires multiple models to be built in a short period of time. For the initial phase, or proof-of-concept, the statistical analysts built three different models: one for overall response and two product preference models. As a proof-of-concept for the genetic algorithm approach, the analysts built similar models using existing processes and techniques.

Business Value

Improved performance: three models were built with an average 10% improvement over existing models

Immediate results: all three models were created in less than one week

Increased productivity: existing process required almost 15 days for only one model

Accelerated learning: preference models built with genetic algorithm approach were significantly better than those built manually; identified attributes that had not been considered in prior models, as these appeared statistically insignificant

Results

Three models were produced in less than one week, all with a minimum of 10% increase in accuracy.

Case #3: New Market Opportunity

Business Challenge

A financial services company was looking to explore a new market opportunity. The program under consideration involved the promotion of debt consolidation loans to a new prospect base. While the initial marketing campaign had established sufficient results to justify further investigation into the program, it was important that a more responsive segment be identified to support future efforts. The challenge was to

Improved performance: three models were built with an average 10% improvement over existing models.

The genetic algorithm found data attributes that were not considered statistically significant and/or were weighted differently in prior models, contributing to improved model accuracy.

The sheer volume of data presents one of the biggest challenges in the traditional modeling process.

select a smaller, more precisely targeted population to yield higher response rates and program adoption. The significant size of the prospect database, combined with over 500 data attributes, would require multiple models to be built quickly.

Business Value

Improved performance: 20% improved model performance set a strong business case for the new program

Immediate results: initial model results were produced in 48 hours; internal resources expected project completion to take six weeks, and analysis would not have begun for over a month due to resource constraints

Accelerated learning: new data discoveries – the genetic algorithm found data attributes that were not considered statistically significant and/or were weighted differently in prior models, which contributed to improved model accuracy

Results

Three robust models were built concurrently in less than 48 hours. The results from the models showed a 20% increase in accuracy, which led to comparable increases in the gross response rate.

Case #4: Demographic Data R&D

Business Case

The sheer volume of data presents one of the biggest challenges in the traditional modeling process; there is such a wealth of internal data that statistical analysts cannot adequately explore all attributes. Add to that the ever-growing list of external data sources, and the task becomes prohibitive. For response models, the client wanted to explore data beyond the standard extract file it had been using for all modeling projects. The existing extract contained almost 150 attributes that included past marketing activity, credit bureau attributes, and high-level demographics. Two demographic files were appended to this standard extract, which expanded the number of attributes to almost 1,200.

Business Value

Improved performance: models with demographic data performed almost 15% better than traditional models, and almost 5% better than the models built using the genetic algorithm alone

Immediate results: response models were built in less than three days

Increased productivity: given time and resource constraints, these analyses could not otherwise have been completed

Accelerated learning: new data discoveries – the genetic algorithm found data attributes that were not considered statistically significant, thus eliminating them from consideration earlier in the process

Results

Data R&D improved response rate models by almost 15%.

GENALYTICS GPS MODEL

The software platform behind the new approach is Genalytics' Predictive Suite. Running on a 100% Java platform, GPS is scalable, distributed, and easy to install in a number of platforms including Linux, Solaris and Windows. GPS can run as a standalone or an embedded application, taking even greater advantage of your current CRM infrastructure and data warehouse investments. Genalytics' software requires a relational database with a JDBC driver that supports JDBC 2.0. Examples of supported databases include Oracle, DB2, Informix, SQL Server, Interbase and MySQL.

GPS is scaleable, and can be installed on anything from a single server for small models to as many servers as required in your production environment. Typical server requirements include a Dual CPU, Pentium 4 or better. The software is generally distributed across multiple servers; throughput is a function of the modeling workload and size of the datasets. The technical environment used to build models described in the case studies included three (3) dual-processor WinTel servers (2.8-3.0 GHz), each with 1 GB memory and 40 GB storage. **Figure 13** below provides a high-level architecture for GPS MODEL:

GPS is scalable, and can be installed on anything from a single server for small models to as many servers as required in your production environment.

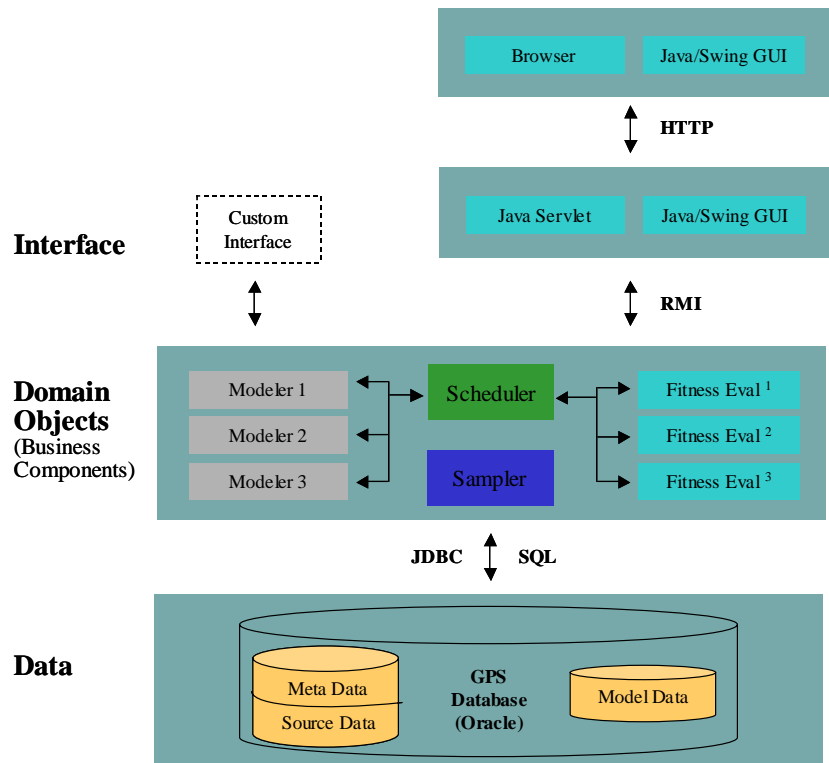


Figure 13. GPS high-level architecture

User Interface

Each user accesses Genalytics Predictive Suite through a single user interface component running on his or her workstation. The user interface software can run on any workstation capable of running Java.

Domain Objects

There are four main server GPS components:

Sampler: used to load and sample data into the GPS database

Modeler: creates and breeds chromosomes via one or more modeler components

Fitness Evaluators: used to determine each chromosome’s fitness level; to improve performance, training data is loaded into memory and chromosomes are evaluated in parallel

Scheduler: coordinates and distributes activities within the GPS environment

Database

Typically an Oracle database, it houses all source, model metadata and results data.

Bibliography

An Introduction to Genetic Algorithms, Melanie Mitchell, MIT Press 1996

Data Mining Cookbook: Modeling Data for Marketing, Risk and Customer Relationship Management, Olivia Parr Rud, John Wiley & Sons; 2000

Mastering Data Mining, The Art and Science of Customer Relationship Management, Michael J. A. Berry, Gordon S. Linoff; John Wiley & Sons; 2000