

Paper 286-2009

Cross-Validation and Learning Curve Model Comparison with JMP® Genomics and Grid Computing

Stan Martin, Pei-Yi Tan, Glenn Horton, Cheryl Doninger, Tzu-Ming Chu, Shannon Conners, Li Li Li, Padraic Neville, Wenjun Bao, Russ Wolfinger
SAS Institute Inc., Cary, NC

ABSTRACT

The Food and Drug Administration has challenged researchers to demonstrate whether or not gene expression data can reliably predict future disease. JMP® Genomics responded with a suite of predictive modeling tools suitable for short, fat data: tens of observations with tens of thousands of variables. Our approach is to compare a variety of models, as we do not know a priori what is appropriate for a particular data set. The number of types and subtypes of models that are practical to consider is primarily limited by CPU time. SAS® Grid Manager, in conjunction with JMP Genomics, provides the intelligent allocation of distributed computing resources and parallel application execution necessary to provide more accurate and timely predictive modeling. In this paper, we describe our approach: the model suite, learning curves for evaluating the number of observations, cross-validation model comparison, and SAS® Grid Computing.

THE BIRTH AND ADVANCE OF GENOMIC SCIENCES

During the last decade of the 20th century the United States embarked on one of the most audacious and potentially transformative human endeavors in recent memory: the sequencing of the human genome. The project had its genesis at the National Institutes of Health, under the direction of James D. Watson, but it quickly morphed into a competitive race between government and private industry. At the dawn of the 21st century, the first draft of the genome was released. Three years later, in April 2003, a final draft was completed, giving us the ability for the first time to interrogate the mother of all codes: the genetic code that makes us human. Expression technology, the ability to assay the expression of hundreds of thousands of genes at a time, quickly followed suit, and the science continues to evolve at a phenomenally rapid pace, spawning industry after industry, and bringing ever closer the tantalizing prospect of truly personalized medicine. Today genomics labs around the world are churning out reams of both sequence and expression data around the clock. The sheer volume of data is overwhelming. A single run from a next generation sequencing instrument can generate over a terabyte of data. The transformation of this genomic data into actionable clinical information is the great challenge of this century.

JMP GENOMICS: A UNIQUE DISCOVERY PRODUCT FROM SAS

While statisticians and biostatisticians typically handle analyses of clinical genomics data sets intended to support the development and validation of diagnostic tests, many scientists in discovery groups are not statisticians. These biologists and bioinformaticists might not be expert programmers, but they desire statistically rigorous tools to analyze their genomics data sets. In response to this demand, some genomics software applications, such as JMP Genomics from SAS, have incorporated point-and-click applications for building and validating predictive models. JMP Genomics is a comprehensive software application specifically designed for the analysis, manipulation, and visualization of data derived from large-scale genomics experiments. In addition to predictive modeling processes, JMP Genomics has flexible import processes that can accommodate data structures from a wide variety of platforms and perform robust statistical analytics on the resulting data sets. While expression data sets have typically been derived from spotted or manufactured array-based microarray platforms or real-time PCR experiments, bead-based array platforms and summarized data from next generation sequencing runs are also becoming heavily used. JMP Genomics provides clustering tools that can be useful in helping scientists to identify candidate biological pathways affected by disease or treatments. This application also offers whole genome Single Nucleotide Polymorphism (SNP) analysis capabilities as well as powerful and efficient analysis of data derived from gene expression experiments.

ROLE OF PREDICTIVE MODELING IN GENOMICS

An important role of predictive modeling in genomics is to estimate the probability of a disease from genetic or gene expression data. The model is trained on historical data in which the disease state is known, and then applied to patients in whom the risk of disease is unknown. The model is generally not built with biological knowledge because

too little biology is known about the relationship between the genes and the disease. On the contrary, the model is usually trained from statistical associations from which a biologist might hope to discover predictive genetic markers that could then be used in a diagnostic setting.

The lack of a known biological relationship is one of three significant obstacles for creating a model. The others are a weak relationship and too many genes per person in the data. The model builder therefore needs to explore a wide variety of relationships, and to guard against overfitting the model to the over-abundant genetic variables.

OVERFITTING

Overfitting occurs when the model not only fits the signal, but also fits the noise in the historical data. For example, if some genes are partially related to a disease state and some others have nothing to do with the disease, we can imagine training a model to use up all the information in the partially related genes, and then to search the unrelated gene data for an association with the still unexplained occurrence of disease. To the extent that the model incorporates these spurious associations, the model is overfitting. The more people per gene in the historical data, the less likely a spurious association occurs. Unfortunately, gene expression data typically has 50 to 500 times as many gene expression variables than people. If nothing is done to prevent it, a model will be based on spurious associations.

JMP Genomics addresses overfitting first by "honest" cross-validation.

CROSS-VALIDATION

When a model overfits the training data, the error rate of the model will be worse when the model is applied to new data than when applied to the original training data. The error rate on new data is called the generalization error of the model.

Cross-validation is a method of evaluating the generalization error of a model. A number of models are trained, each with a different subset of the data, and then evaluated on the data excluded from the training. The training process is the same for each model, only the training data differs. Notice that the model actually deployed on new data is usually trained with all the available data. Cross-validation does not use that model, even though it is evaluating it.

JMP Genomics offers several methods of cross-validation within its Cross-Validation Model Comparison Analytical Process (CVMC). Stratified 5-fold cross-validation is used in the examples in this paper. In K -fold cross-validation, the available data is partitioned into K subsamples. K models are trained. The training data from the K th model consists of all but the K th subsample of observations. After all K models are trained, each observation will have been used exactly once in evaluating a model. The cross-validation estimate of error is the average of the evaluations. For example, the cross-validation estimate of classification accuracy is the proportion of observations correctly classified, where an observation in the K th subsample is classified using model K . Note the cross-validation estimate is not the error estimate of a particular model. Rather, it estimates the generalization error of the specific modeling algorithm for a given number of observations. Using $K=10$ produces a much more optimistic estimate. For example, it provides a higher accuracy rate than $K=5$. Using $K=2$ tends to produce a more pessimistic estimate. Whether the estimate over- or underestimates the "true" generalization error is usually not known. Without knowing this, cross-validation is still helpful for comparing different models, as is done in CVMC.

STRATIFICATION AND REPEATED CROSS-VALIDATION

JMP Genomics stratifies the cross-validation of a categorical trait: the proportions of the category within each subsample are as close as possible to the proportions in the original data. Stratified cross-validation tends to reduce the variance of the cross-validation estimates. JMP Genomics also repeats cross-validation by using different partitions in each individual cross-validation run. The resulting scatter of estimates gives an impression of their range—a crude impression of their standard deviation. The impression can be deceptive though, especially for a large K model, because a training data set in one cross-validation run will have many observations in common with a training data set in a repeated run.

CROSS-VALIDATION MODEL COMPARISON RESULTS

Generating cross-validation estimates from several types of models is the purpose of the JMP Genomics Cross-Validation Model Comparison Analytical Process (CVMC). We show two examples, one from a gene expression data with 30 observations, and the other using intensities from Single Nucleotide Polymorphism (SNP) microarray with about 3500 observations.

GENE EXPRESSION DATA

Figure 1 shows CVMC results for gene expression data consisting of 30 observations and 45105 expression measurements. The data originates from a toxicological study: An animal is given a chemical being studied, expression data is subsequently collected, and, months later, the animal is examined for cancer.

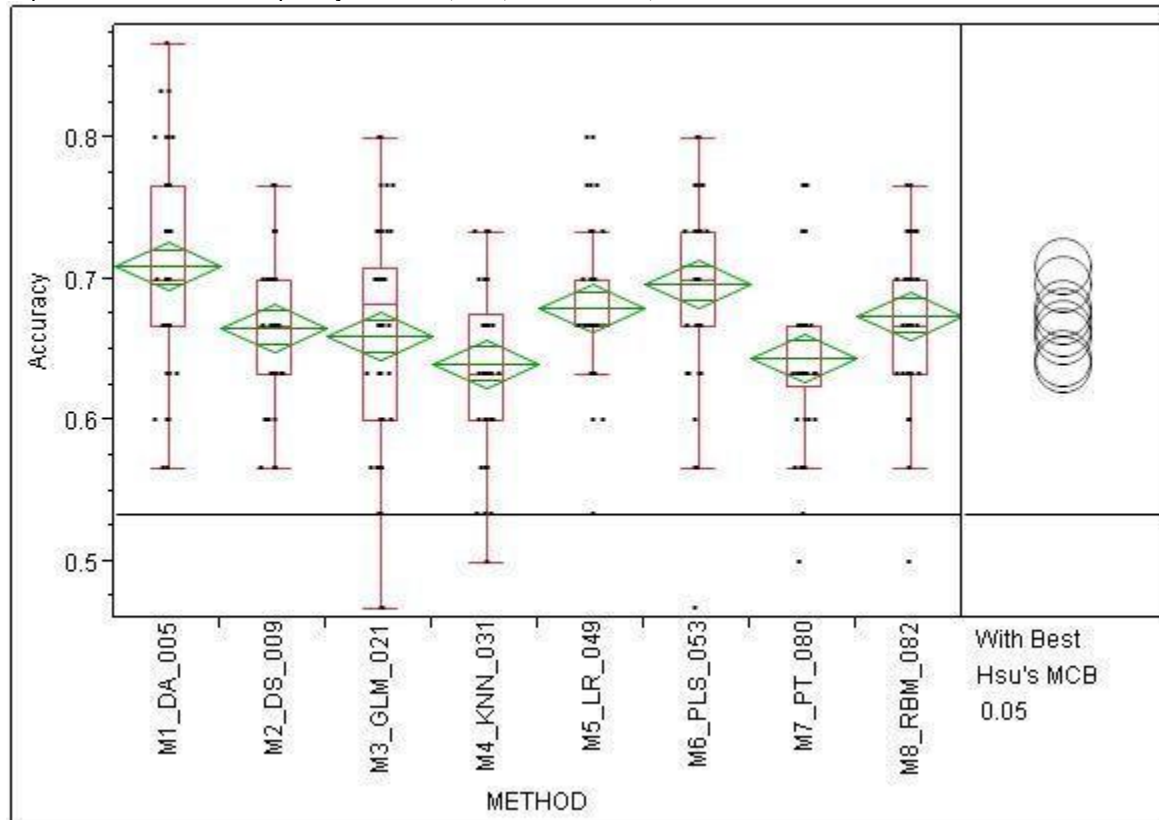


Figure 1. 5-Fold Random Cross-Validated Model Accuracy Comparison

If a model can use expression data to predict whether the animal will get cancer, researchers would have another clue for early detection of cancer and possibly reduce the time and expense of testing for chemical carcinogens.

The vertical axis measures accuracy, or the proportion of observations the model correctly predicts. The horizontal axis shows eight types of models. For each type, 5-fold cross-validation was repeated 50 times. The estimate of accuracy for each of the 50 CV runs is plotted. Because there are only 30 observations, the number of distinct values of accuracy is limited. Ten distinct points are the most that appear for any model.

A box-plot is overlaid on the points to indicate the quartiles of the points. Within the box-plot is a diamond, giving the impression of standard errors. The horizontal line in the middle of the diamond is the average of the 50 runs. It is the estimate of the accuracy to compare other models with.

The model on the left has the highest accuracy. Notice that the range of points includes the averages of the accuracies of each of the other models, suggesting that the model is not a clear winner. The panel on the right contains overlapping circles, one for each model. The center of a circle is the same height as the accuracy for its model. In an idealized situation, two circles only intersect if there is no statistical difference between the models. The present example is not ideal because the cross-validation runs are not independent: different runs have different training data in common.

One of the most important lines on the plot is the horizontal line near the height, 0.57. That marks the accuracy of an estimate with no model: 0.57 of the 30 observations had the same outcome, cancer. A model with less accuracy than this baseline is worse than having no model at all. Fortunately, none of the models shown have this problem, on average.

Figure 2 shows the same results with a different error measure. Instead of accuracy, the vertical axis is the root-mean-square-error (RMSE), sometimes known as the square root of the Brier score. Smaller values of RMSE are better, which is the opposite of accuracy. Notice that there are more points above each model than in the previous plot. RMSE is a more refined measure than a simple proportion of counts. That is why some people prefer it.

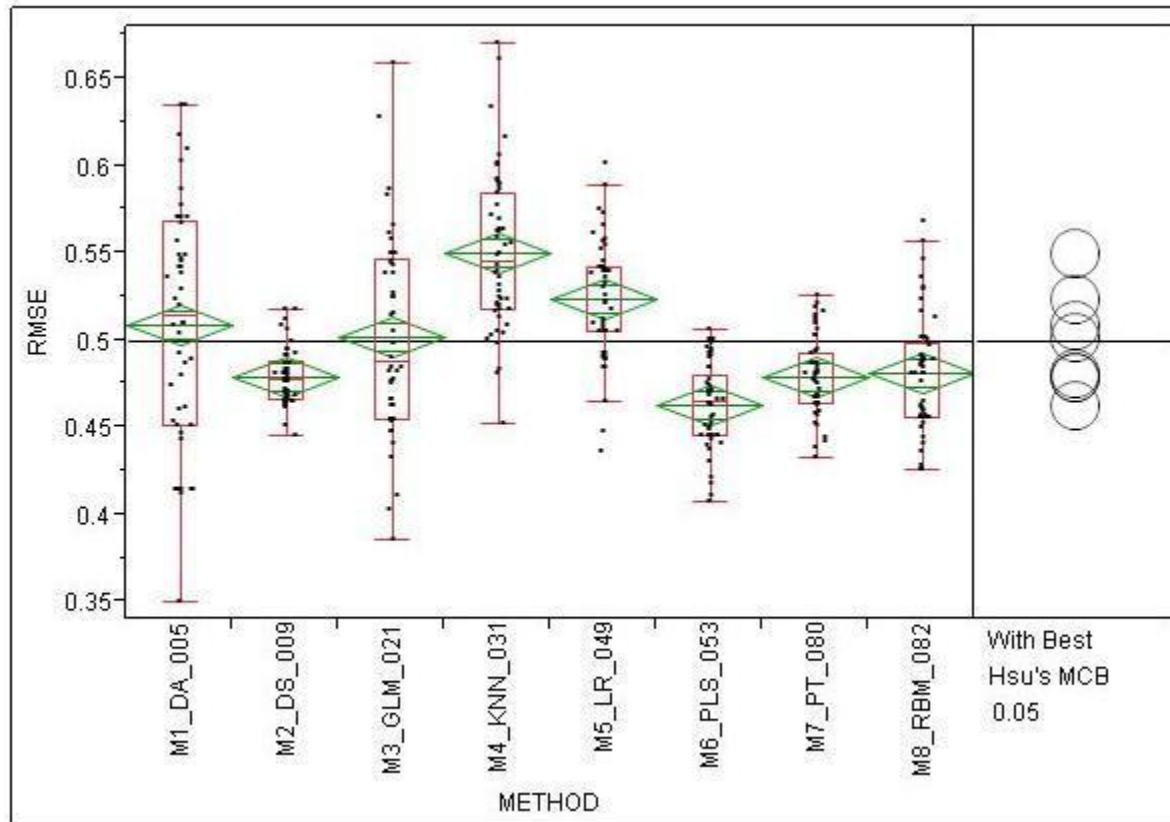


Figure 2. 5-Fold Random Cross-Validated Model RMSE Comparison

Next notice that the average RMSE for four of the models are above the baseline. Those models are overfitting so much they are worse than having no model. Even the first model, the model that has the best CV accuracy, is worse than having no model on this plot. If the purpose of model is limited to classification accuracy, then perhaps the first model can be used. However, if the model will be used to estimate the probability of disease, then the best model on the RMSE plot should be used.

Another type plot, called the Reliability plot, reveals the problem. The vertical axis ranges from 0 to 1. It represents the proportion of observations with cancer. The horizontal axis ranges from 0 to 1 and is divided into 10 equally spaced bins. Each bin collects those observations whose predicted probability of cancer is within the range of the bin. The point plotted in the bin is the fraction of proportion of observations with cancer among those observations whose predicted probability of cancer is within the bin range.

The reliability plot in Figure 3 shows the CV results from the first model, which is the model with the best accuracy. Going from left to right, points are plotted with a higher probability of cancer, according to the model. Notice that the actual proportion of cancer patients does not increase much with the probability.

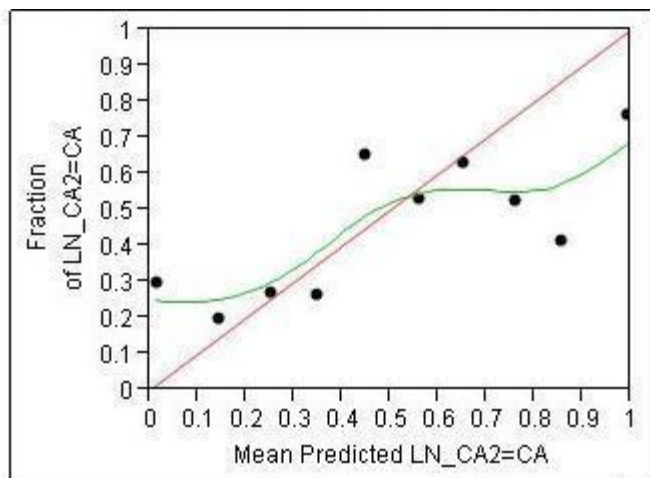


Figure 3. Reliability Diagram from Discriminant Analysis Model

Figure 4 shows the results from the model with the best average cross-validated RMSE. The proportion of cancer patients increases with the probability of cancer. This model is much more reliable for statements about the probability of cancer.

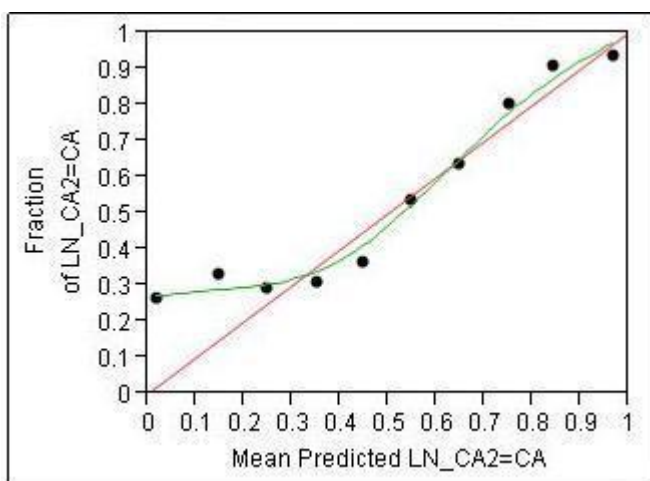


Figure 4. Reliability Diagram from Partial Least Squares Model

LEARNING CURVES

Some types of models need more observations than others to achieve good results. A learning curve is a plot of model error against the number of observations in the training data. Generally, the curve starts relatively steep, slowly flattens out, and might, eventually, turn back from whence it came. The slope of the curve indicates the improvement in prediction for each additional observation in the training sample. After drawing one curve, others can be drawn using different samples for training data. If the curves are approximately parallel, then the dependence between error and training size is fairly stable over training samples. On the other hand, if the curves seem unrelated to each other over a range of initial training sample sizes, the generalization error from models trained with these sample sizes is unreliable.

The Learning Curve Model Comparison Analytical Process (LCMC) in JMP Genomics plots a point to mark the error estimate from the repeated cross-validation of a model. The same model is rebuilt with different sizes of training data, producing the different points along the curve. Larger training data sets include all the observations of smaller data sets. Different learning curves are made for the same type of model by using different subsets of the available data for the training data sets. The envelope of these curves is narrow or wide depending on whether the model is

reproducible on different data or not.

Learning curves are not informative unless there are enough observations to show a difference in model prediction. The Wellcome Trust Case Control Consortium (WTCCC) has provided 2519 SNP microarray intensities for about 2000 cases with Coronary Artery Disease and about 1500 control observations. Figure 5 shows learning curves for three types of models run on these data. The vertical axis represents RMSE. Smaller values represent better models. The three thick lines are the average learning curves for the three models. For each, the RMSE moves downward, left to right, from small training data sets to larger ones. The learning curve with the highest RMSE, becomes horizontal around 1000 observations, indicating that more data is superfluous. It even rises a little at the end, suggesting the model might be overfitting with so much data. The bottom two curves are smoother, and continue to descend at the largest data sizes, suggesting that more training data would improve the models. The model with the middle RMSE at small data sizes becomes the preferred model when there are more than 1500 observations. In general, different models require different amounts of training data to achieve comparable fit, and some that require more data can end up fitting better.

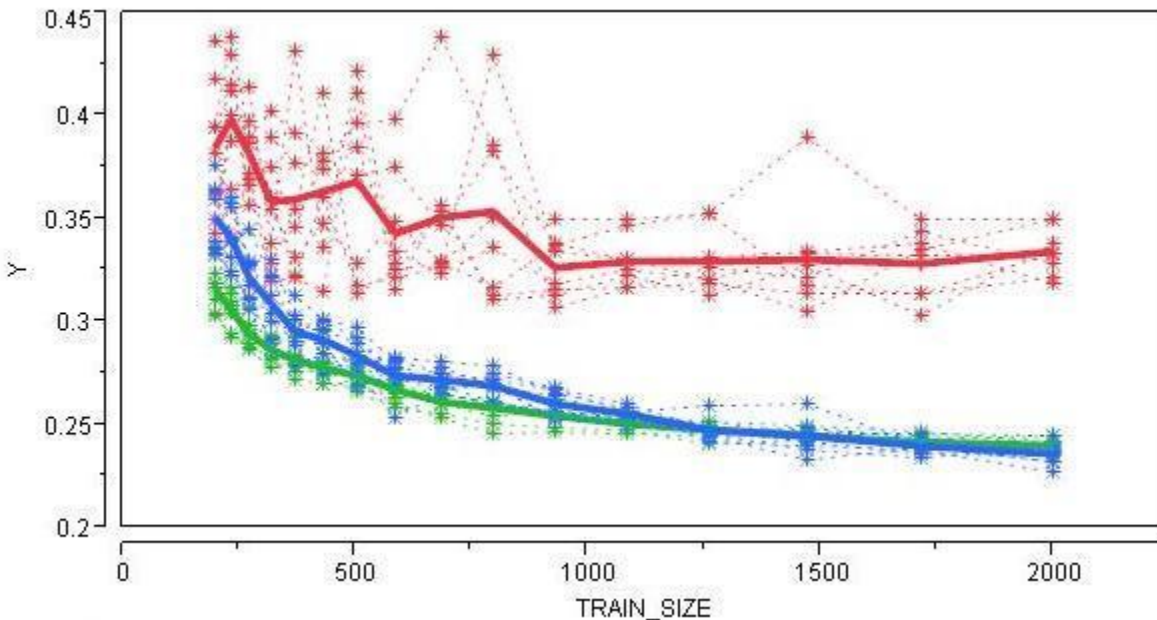


Figure 5. Average RMSE Fixed Test and Models Individuals Learning Curve

The individual learning curves are shown as thin, dotted lines. The individual curves of the highest RMSE model vary greatly in height compared to the individual curves of the two models with smaller RMSEs. Individual curves that vary greatly are a warning that the model predictions on new data will be much different than expected. In this example, the model with the worst RMSE also has the worst variability. This is a coincidence: The best model can have the worst variability.

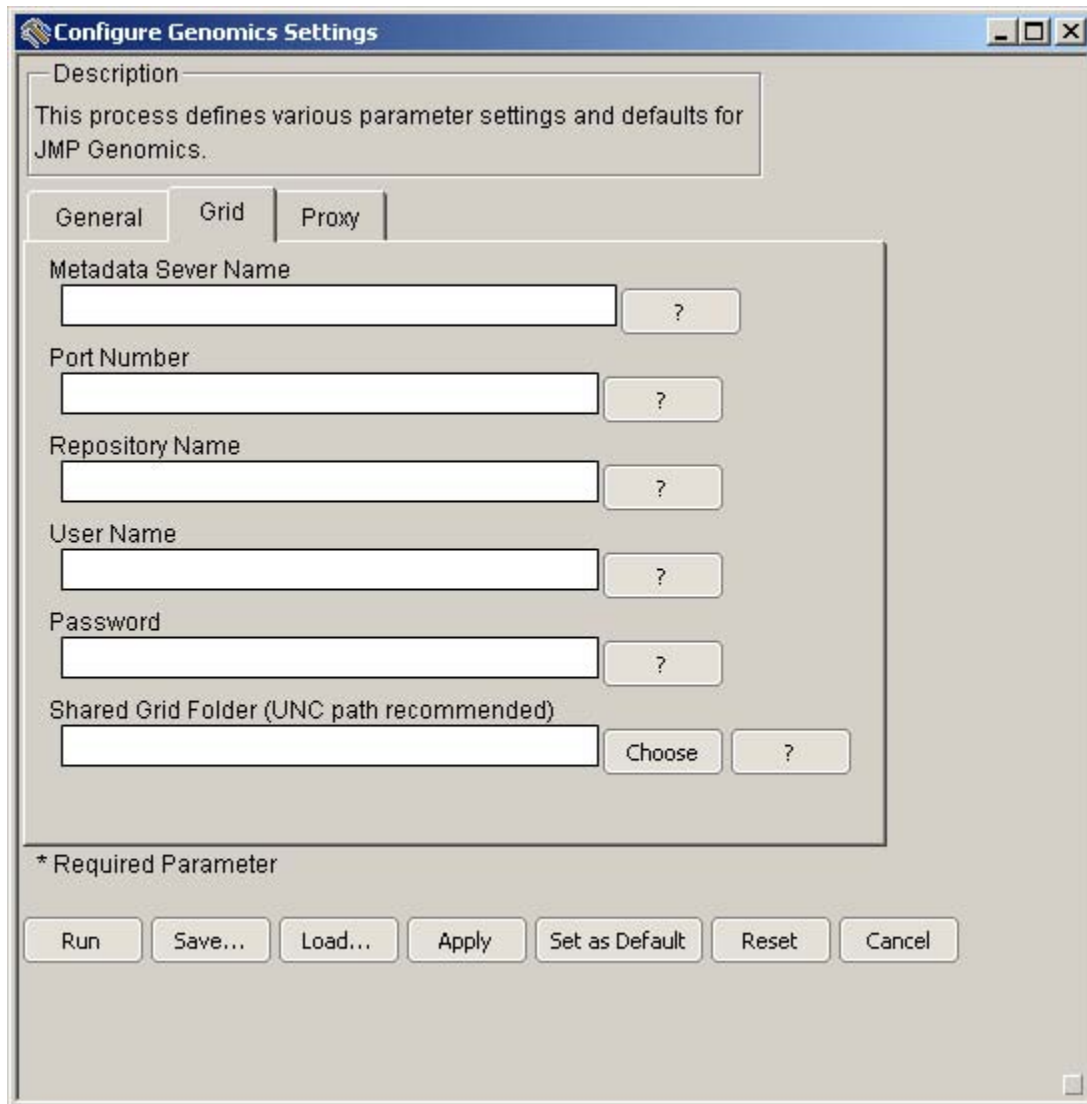
MODEL SPACE

Having cross-validation and learning curve tools, the question is now which models to consider. The quick answer is "all of them." However, so little is known about the dependence of phenotypes on genetics information that no modeling relationship should be overlooked. "All of them," though, are too many: Suppose there are 10 algorithms, each with 10 parameter settings, and five different preliminary variable selection methods. Ten iterations of 5-fold cross-validation would require 25,000 models being fit to the data. The ability to train this many depends on the available computing resources. Since the computational capabilities of standard issue research computers are limited, it is imperative to develop parallelized algorithms and implement them in such a fashion that multiple "clusters" of computers can be used to solve the problem.

PARALLELIZATION IN JMP GENOMICS USING SAS GRID MANAGER

JMP Genomics addresses the need to parallelize by effectively harnessing the power of SAS Grid Computing. SAS

Grid Computing enables processes that can be run independently, such as modeling within CVMC and LCMC, to run on multiple computers at once. This is accomplished by virtue of the SAS Grid Manager product, which enables a single “master control node” to direct computational nodes to perform the parallelized operations. JMP Genomics enables the user to configure their Grid environment using the **Grid** tab, which is located on the “Configure Genomics Settings” dialog box. The **Grid** tab is shown in Figure 6.



The screenshot shows the 'Configure Genomics Settings' dialog box with the 'Grid' tab selected. The dialog has a title bar with the text 'Configure Genomics Settings' and standard window controls. Below the title bar is a 'Description' box containing the text: 'This process defines various parameter settings and defaults for JMP Genomics.' Below the description are three tabs: 'General', 'Grid', and 'Proxy'. The 'Grid' tab is active and contains several input fields, each with a '?' button to its right: 'Metadata Server Name', 'Port Number', 'Repository Name', 'User Name', 'Password', and 'Shared Grid Folder (UNC path recommended)'. The 'Shared Grid Folder' field has a 'Choose' button and a '?' button. At the bottom of the dialog, there is a legend '* Required Parameter' and a row of buttons: 'Run', 'Save...', 'Load...', 'Apply', 'Set as Default', 'Reset', and 'Cancel'.

Figure 6: The Grid Tab of Configure Genomics Settings Dialog Box.

The Cross Validation Model Comparison dialog box (Figure 7) has the option **Use Grid Computing**. When this option is selected and an authenticated grid is present, the process branches so that the various models are distributed to the computational nodes in the grid.

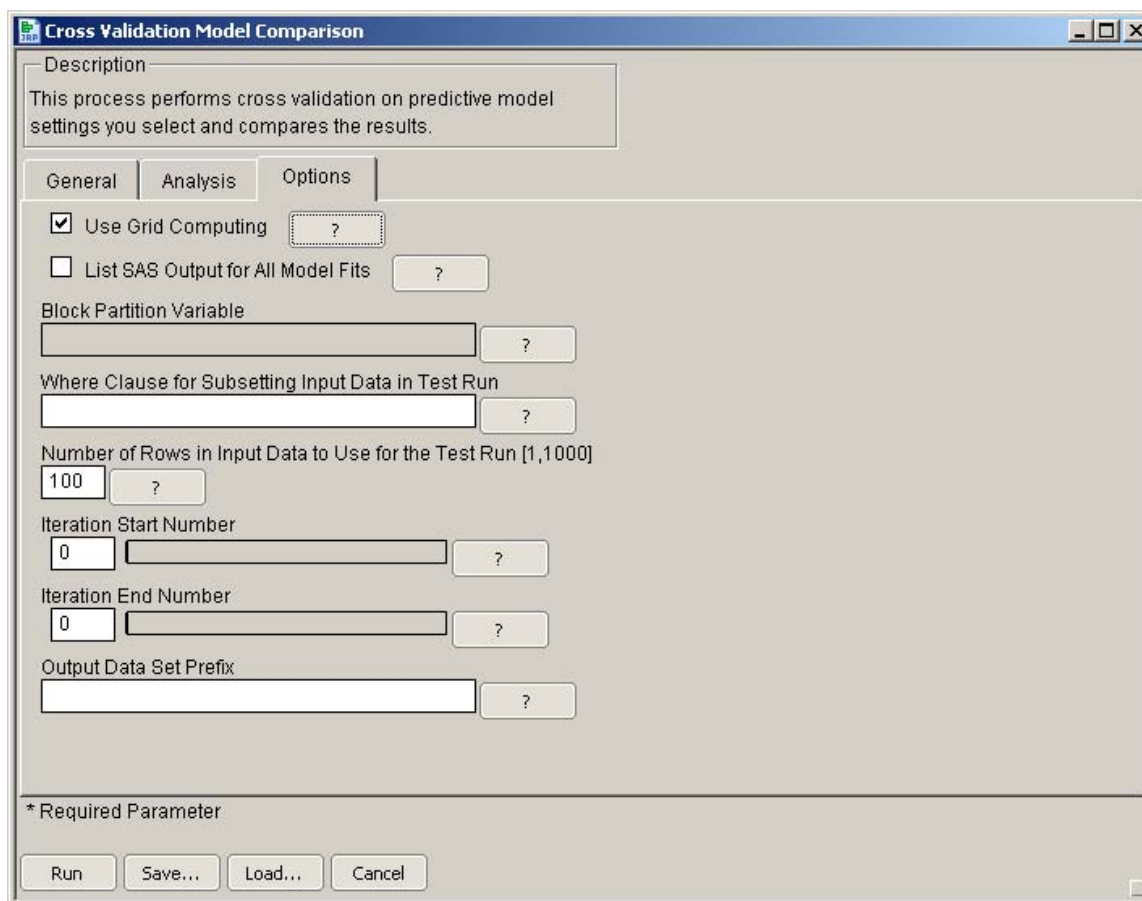


Figure 7: Options Tab of Cross Validation Model Comparison Dialog Box with Use Grid Computing Check Box Selected.

The distribution algorithm counts the number models to be evaluated, and the number of available nodes in the grid. It then attempts to make a connection to each node. When a connection is successful, an RSUBMIT statement is executed that sends one of the models to be evaluated to that node. Once a node is in use, it cannot be connected to a second time, so the algorithm moves on to the next node. Once a job is finished, a node becomes available. This process continues until all requested models are evaluated. The results are then collated in a special folder designated as the "Share Grid Folder" and surfaced to the user in JMP with the graphics used in JMP's internal one-way modeling platform.

BENCHMARKING USING THE MICROARRAY QUALITY CONSORTIUM AND WELCOME TRUST DATA SETS.

To quantify the computational advantage gained by implementing a parallelized model comparison program we ran several benchmarks. In the first case, we ran CVMC with 84 predictive models. The analysis parameter conditions for CVMC were Stratified Random as the holdout method, the holdout size was $K=5$ and the number of random holdout iterations equaled 50. Our test box was a single computational node, (Dell Optiplex 755 with Intel Dual Core CPU E6550 running at 2.33 GHz clock speed and equipped with 4 Gigabytes of RAM, with JMP Genomics installed on Windows XP Service Pack 3. On this node CVMC ran for over 95 hours. Using a grid of nine computational nodes with similar specifications, CVMC finished in 11.5 hours.

In the second case, the input table for predictive models contained 30 observations and 45105 variables. Two predictive models were used in the CVMC. The first model was the Partial Least Squares that defined 120 as the maximum number of filtered predictor, t -test as the statistical testing method for continuous predictors, and the proportional as assumed prior proportions of each of the classes. The second model was the Partition Trees that also defined 120 as maximum number of filtered predictor, t -test as statistical testing method for continuous predictors, and the proportional as assumed prior proportions of each of the classes, plus we chose Forest as the automated

model type with 100 as maximum number of trees for four being the maximum number of variables to consider for splitting a node. The CVMC selected Random Partition as the holdout method, specified the holdout size of the number of folds or groups in the partition (K fold) as 5, and then changed the number of random holdout iterations as 10, 25, 50, 75 and 100 with and without the active grid. Figure 8 shows the results of this comparison. As indicated in the graph, the increase in speed very closely approximated a linear relationship.

Figure 8 shows a comparison of the amount of time required for various iterations when the grid was used, and when it was not used. The red dots indicate time required when not using the grid. The triangles indicate time required with the grid. Triangle are color coded based on the number of nodes in the grid. Green = 10 nodes. Light Blue = 12 nodes. Dark Blue = 13 nodes.

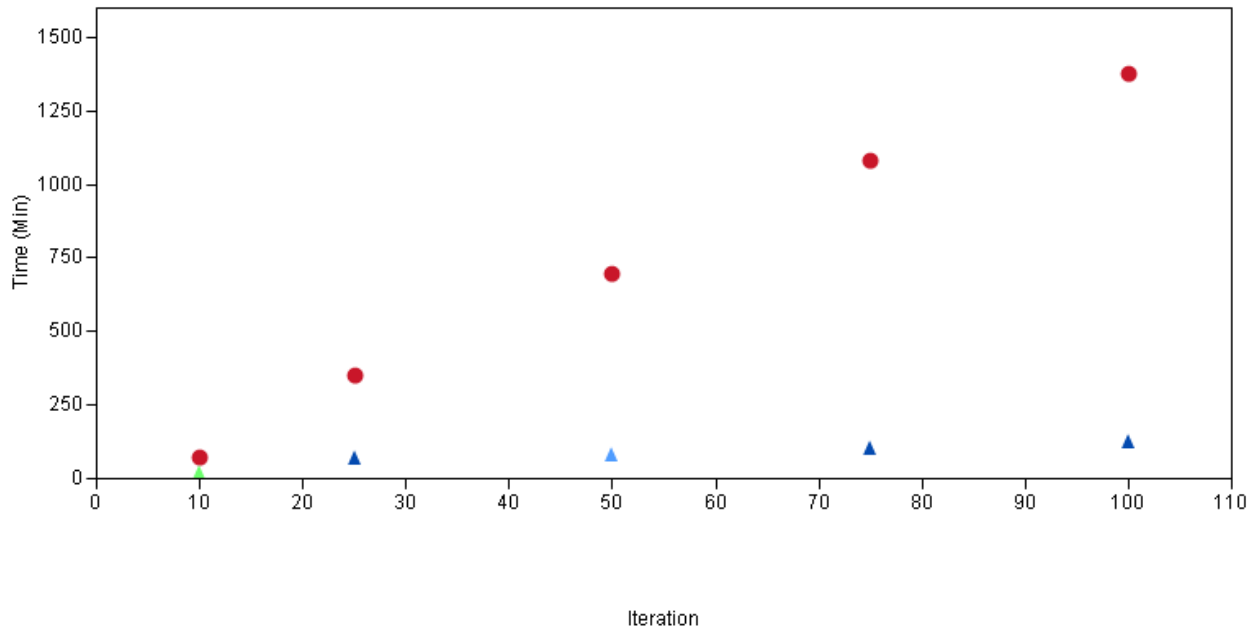


Figure 8. A Comparison of the Amount of Time Required for Various Iterations When the Grid Was Used and When It Was Not Used.

Figure 9 gives us a plot of versus actual results when using the grid environment.

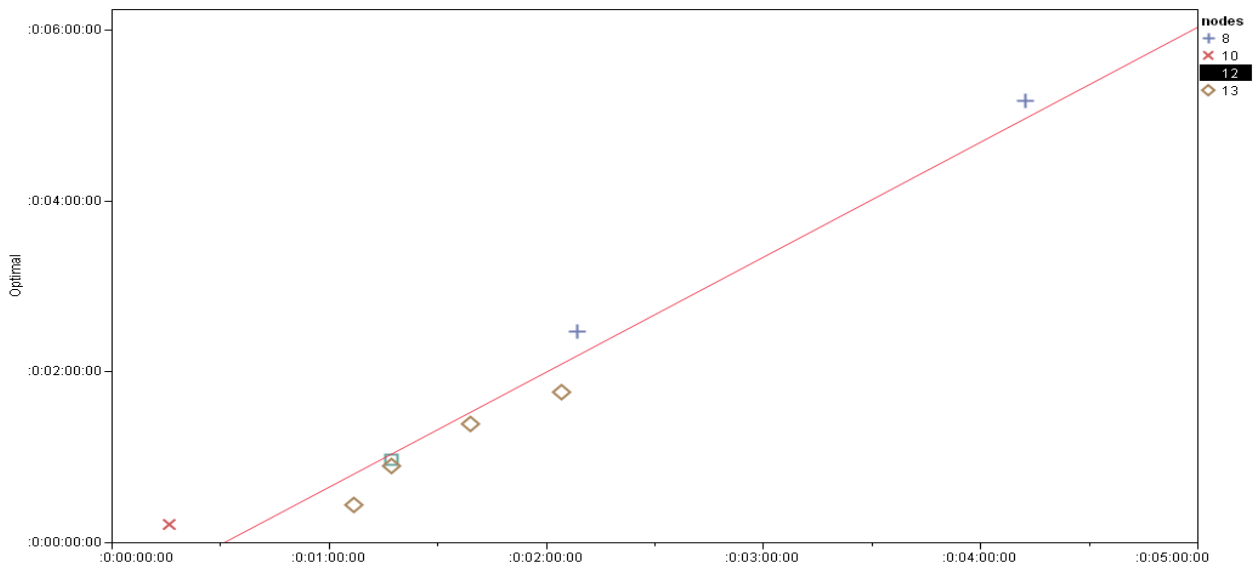


Figure 9. Plot of Optimal Versus Actual Results When Using the Grid Environment.

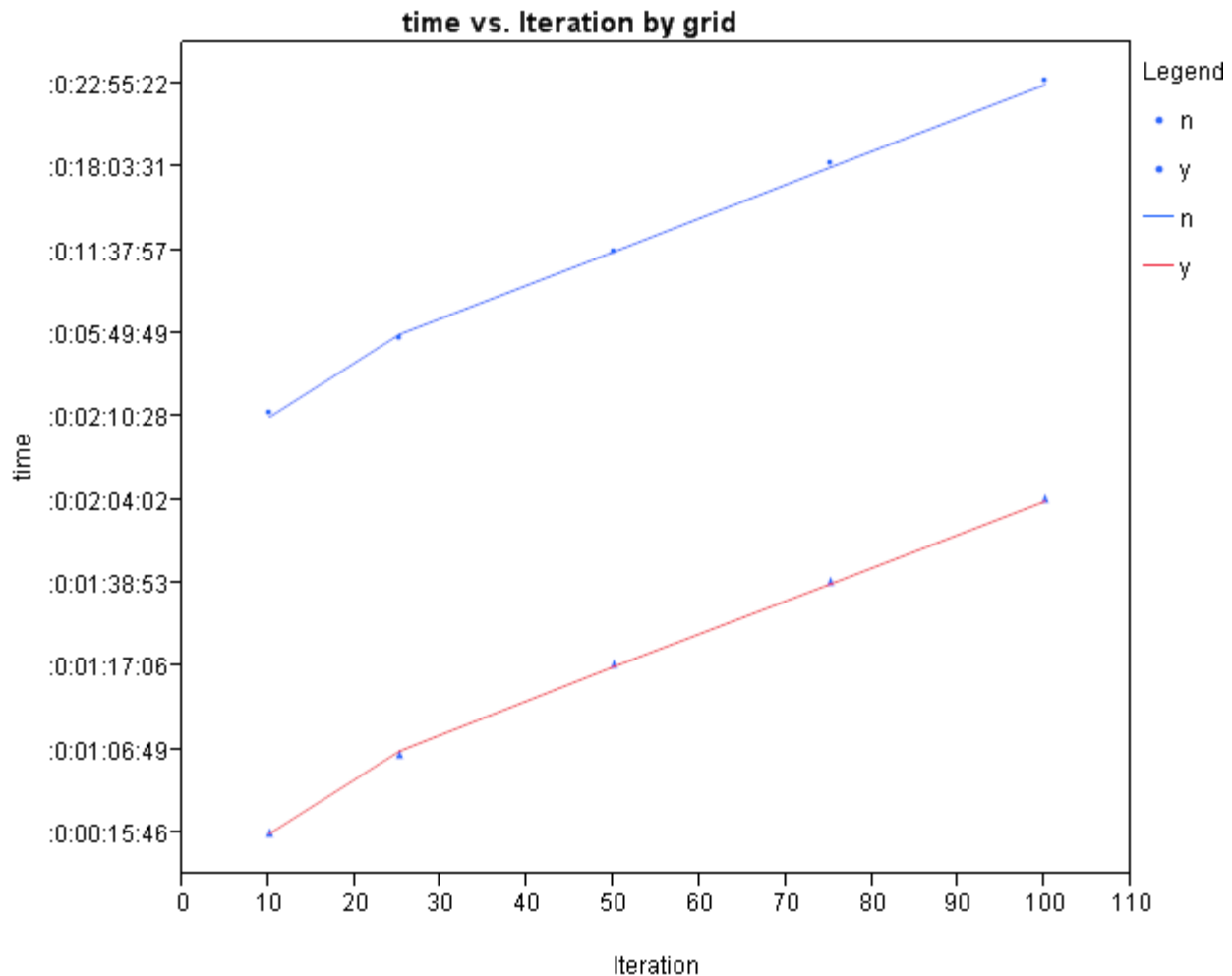


Figure 10 shows a comparison of the amount of time required for various iterations when the grid was used, and when it was not used. The blue line indicates time required when not using the grid. The red line indicates time required with the grid.

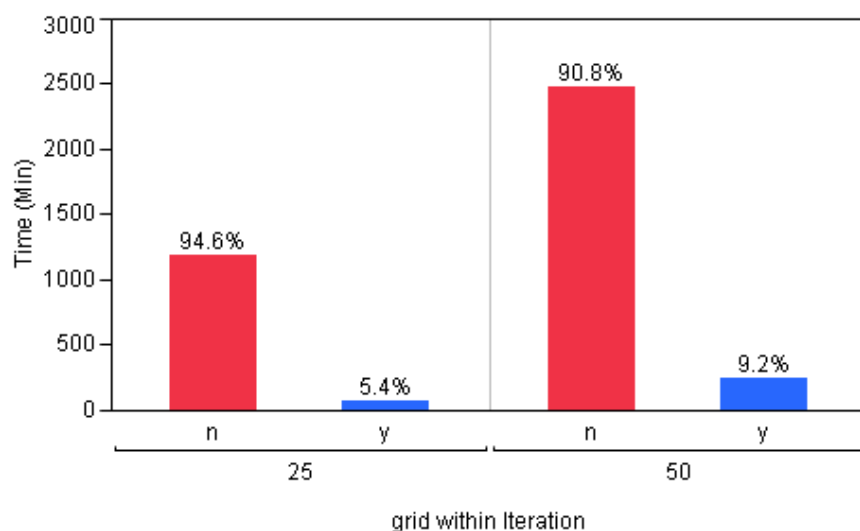
A third case used eight predictive models:

1. The Discriminant Analysis model defined 30 as the maximum number of filtered predictor, *t*-test as the statistical testing method for continuous predictors, specified quadratic metric for computing distances between groups and the proportional as assumed prior proportions of each of the classes and selected stepwise as variable selection method with significance level for significance variables as 0.05 and significance level for retaining variables as 0.05.
2. The Distance Scoring model defined 30 as maximum number of filtered predictor, *t*-test as the statistical testing method for continuous predictors, specified the Euclid method for computing distances between observations with Mean Class Centroid as nominal dependent variables, selected proportional as assumed prior proportions of each of the classes, and Gaussian as threaded kernel function for computing posterior probabilities.
3. The GLM Select model defined 30 as the maximum number of filtered predictor, *t*-test as the statistical testing method for continuous predictors, selected proportional as assumed prior proportions of each of the classes and using Stepwise as the model selection method with SBC for the criterion for stopping model selection.
4. The K Nearest Neighbors Model defined 30 as maximum number of filtered predictor, *t*-test as the statistical testing method for continuous predictors, specified 5 as number of nearest neighbors, used Mahalanobis as distance metric, and selected proportional as the assumed prior proportions of each of the classes.
5. The Logistic Regression model defined 120 as maximum number of filtered predictor, *t*-test as the statistical

testing method for continuous predictors, used Penalized as the variable selection method with $-\log 10$ equal to 1 as controlling the severity of the penalty, and selected proportional as assumed prior proportions of each of the classes.

6. The Partial Least Squares model defined 120 as maximum number of filtered predictor, t -test as statistical testing method for continuous predictors, selected proportional as the assumed prior proportions of each of the classes and specified 10 as number of PLS components.
7. The Partition Tree model defined 120 as maximum number of filtered predictor, t -test as the statistical testing method for continuous predictors, and proportional as the assumed prior proportions of each of the classes, and chose Forest as the automated model type with 100 as maximum number of trees for four maximum number of variables to consider for splitting a node.
8. The Radial Basis Machine model defined 60 as the maximum number of filtered predictor, t -test as the statistical testing method for continuous predictors, and proportional as the assumed prior proportions of each of the classes.

Eight nodes were used on the grid. Each model was run for 25 and 50 iterations with and without the grid nodes. Figure 11 shows these results.



grid ■ n ■ y

Figure 11. Time Necessary to Run 25 and 50 Iterations on Eight Models.

The presence of the grid sped up compute time by 94.6% when running 25 iterations on 8 models. Compute time was improved by 90.8% when running 50 iterations on 8 models.

CONCLUSION

These results confirmed our instinct that modeling problems that have thus far been intractable become tractable when one adopts a parallelization strategy. Grid-enabled JMP Genomics 4.0, the first release of the software that includes parallelization, combined with SAS Grid Manager represents a quantum leap forward in terms of computational tractability. Future releases will refine and improve this paradigm, and it is hoped that those problems that are currently not addressed due to their computationally intense nature will be able to be quickly and efficiently addressed using JMP Genomics.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Stan Martin
 Email: Stan.Martin@jmp.com

Name: Pei-Yi Tan
Email : Pei-Yi.Tan@sas.com
Name: Glenn Horton
Email: Glenn.Horton@sas.com
Name: Cheryl Doninger
Email : Cheryl.Doninger@sas.com
Name : Tzu-Ming Chu
Email: Tzu-Ming.Chu@jmp.com
Name: Shannon Conners
Email: Shannon.Conners@jmp.com
Name: Li C Li
Email : LiC.Li@sas.com
Name: Padraic Neville
Email : Padraic.Neville@sas.com
Name: Wenjun Bao
Email: Wenjun.Bao@jmp.com
Name: Russ Wolfinger
Email: Russ.Wolfinger@jmp.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.