
Class Prediction: K-Nearest Neighbors

Contents at a glance

I.	Definition and Applications.....	2
II.	How do I access the Class Prediction tool?	2
III.	How does the Class Prediction work?.....	4
IV.	How does GeneSpring® determine which genes are best discriminator/predictor?	5
	a. Prediction strength calculation	5
V.	How does the algorithm predict test samples?	7
	a. Summary of the K-nearest algorithm.....	7
	b. Interpreting the Prediction Results window	7
	c. Figure illustrating prediction and k-nearest neighbors concepts	9
VI.	What is the purpose of cross-validation and how does it work?	10
VII.	Most commonly asked questions about Class Predictor.....	11

I. Definition and Applications.

Class prediction is a supervised learning method where the algorithm learns from samples with known class membership (**training set**) and establishes a prediction rule to classify new samples (**test set**). This method can be used, for instance, to predict cancer types using genomic expression profiling.

- Predict the class/phenotype/parameter of a sample
- Identify genes that discriminate well among classes
- Identify samples that could be potential outliers

This technique is best used with at least 20 samples or conditions per class. In GeneSpring, there are two class prediction algorithms that can be used to achieve the objectives mentioned above: K-Nearest Neighbors and Support Vector Machines (SVM). This analysis guide will focus on the K-Nearest Neighbors algorithm.

II. How to access the Class Prediction tool.

Before using the Class Prediction tool, you must assign class membership to **every** sample in your training set (set of samples with known class membership) by defining parameter values in the Change Parameter window. Samples belonging to the same class must be designated with the same parameter value. For example, all leukemia samples in our training set have been assigned as ALL or AML under the parameter Leukemia Type (as in data set published in Golub et al, 1999). Samples with unknown designation (N/A) should be removed from the experiment.

Experiment Parameters for AML ALL training set

Please select values for experimental parameters.

Warning: Modifying parameters may invalidate existing experiment trees built from this experiment.

Parameter Name	Leukemia	Cell type	FAB
Parameter Units			
Numeric	no	no	no
Logarithmic	no	no	no
1: data_set_ALL_AML_train.txt column 3	ALL	B-cell	NA
2: data_set_ALL_AML_train.txt column 5	ALL	T-cell	NA
3: data_set_ALL_AML_train.txt column 7	ALL	T-cell	NA
4: data_set_ALL_AML_train.txt column 9	ALL	B-cell	NA
5: data_set_ALL_AML_train.txt column 11	ALL	B-cell	NA
6: data_set_ALL_AML_train.txt column 13	ALL	T-cell	NA
7: data_set_ALL_AML_train.txt column 15	AML	B-cell	NA
8: data_set_ALL_AML_train.txt column 17	ALL	B-cell	NA
9: data_set_ALL_AML_train.txt column 19	ALL	T-cell	NA
10: data_set_ALL_AML_train.txt column 21	ALL	T-cell	NA
11: data_set_ALL_AML_train.txt column 23	AML	T-cell	NA
12: data_set_ALL_AML_train.txt column 25	AML	B-cell	NA
13: data_set_ALL_AML_train.txt column 27	AML	B-cell	NA

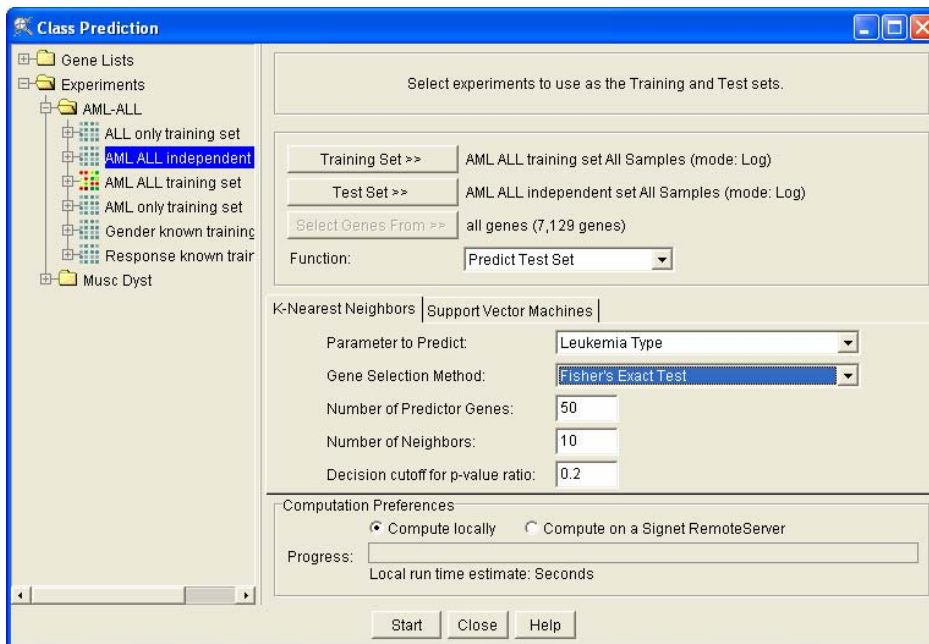
Buttons: Import Parameter..., New Parameter..., Delete Parameter, Replace Text..., Extract Subvalues, Fill Down, Fill Sequence Down, Set Value Order..., Inspect...

Buttons: Save, Save As..., Cancel, Help

Fig 1: Each sample must be assigned to a class (ALL or AML in that example) in the Experiment Parameters window

After saving the experiment parameter settings, make sure to set the class parameter (Leukemia) as continuous element in the Experiment Interpretation window. You may then proceed with the class prediction:

1. Go to **Tools** ⇒ select **Class Prediction** ⇒ select the **K-Nearest Neighbors** tab (see Fig. 2)
2. In the Class Prediction window, open the Experiments folder on the left-hand side of the window, and select the training set. Click on “Training Set >>” to assign training set. Do the same for test set and gene list.
3. Set appropriate prediction rules, such as the gene selection method, parameter to predict, number of predictor genes, number of neighbors (see details in following sections). Select **Predict Test Set** to start the prediction algorithm.



} These 4 settings make up the Prediction Rule

Fig 2: Class Prediction window

Class Prediction window:

Training Set: Sample set for which true class membership is known. Gene expression data from these samples are used to determine the predictor genes.

Test Set: Sample set for which class membership will be predicted based on learned rule from training set. True class membership for these samples may or may not be known.

Select Genes From: Gene list from which predictor genes will be selected. This should be a set of reliable genes to improve accuracy of prediction.

Parameter to Predict: Parameter name containing parameter values defining class membership of samples.

Gene Selection Method: The method that will be used to select predictor genes.

Number of Predictors: The number of best predictor genes that the algorithm will use for the prediction rule.

Number of neighbors: K number of training samples that is nearest to the test sample, based on Euclidean distance of normalized expression intensity (K-nearest neighbor method). The number of neighbors should not exceed the number of samples or conditions in the smallest class. As a rule of thumb, set it to 2/3 of the number of samples in the smallest class. See details in section V.

Decision cutoff for P-value ratio: A rule indicating how the algorithm should make a prediction for the test sample. A p-value ratio of 0.2 (equivalent to 1/5) indicates that the algorithm will make a prediction if the p value (probability that the test sample is predicted as belonging to one class by chance) of the first best class is at least 5 times smaller than the p-value of the next best class. If the actual p value ratio is less than the cutoff, a prediction will be made, if the ratio is higher, no prediction will be made. Setting the p value cutoff to 1 will force the algorithm to always make a prediction but may result in more prediction errors.

Crossvalidate Training Set: This option tests how well the prediction rule – number of predictor genes, decision cutoff p value ratio, and number of neighbors – is at discriminating between classes in the training set.

Predict Test Set: This option becomes enabled after a test set has been assigned in the Predict Parameter window. This will predict the test samples and generate a list of predictor genes.

III. How does Class Prediction work?

We will refer to the Acute Lymphoblastic Leukemia (ALL) and Acute Myeloid Leukemia (AML) data set published by Golub et al (1999) to illustrate all concepts regarding the Class Predictor. This data set contains a training set with 27 ALL and 11 AML samples, and a test set with 20 and 14 ALL and AML samples respectively.

The **Class Predictor** is designed to predict the class membership of an uncharacterized sample or condition based on a set of rules learned from the training set. It works in two steps:

1. Determines best predictor genes using the training set (see section IV)
2. Using predictor genes, predicts test samples using the k-nearest neighbor algorithm (see section V).

IV. How does GeneSpring determine which genes are best predictors?

Prediction strengths are evaluated for all genes on selected gene list using data from training samples. All genes are evaluated independently and ranked by their power to discriminate each class from all others using information from that gene alone.

For each gene, the best decision cutoff point for predicting each group versus all others is determined, by evaluating the highest relative class abundance on one side of a cutoff mark in comparison to the other side of the cutoff mark (see Fig. 5).

a. Prediction strength calculation using Fisher's Exact Test:

1. The class prediction isolates a gene.
2. For each sample, it calculates the probability of obtaining the observed number of samples from each class above and below that cutoff mark (Fig. 5) by chance, using Fisher's exact test (hypergeometric distribution).
3. Selects the smallest p-value calculated in step 2 and converts it into prediction strength by taking negative natural log of the p-value.

4. Repeats steps 1 to 3 until prediction strengths for all genes on selected gene list are calculated.
5. Ranks the genes according to their predictive strength for each class.
6. Genes with highest predictive strength for each class are selected equally to generate a final list of best predictor genes. The final number of best predictors is user-specified.

b. Prediction strength calculation using Golub Method:

1. For each gene, the Golub Method calculates the difference in means between groups divided by the sum of the standard deviations.
2. This ratio that is calculated above is the prediction strength for the gene. The best predictor genes have large between-group variability and small within-group variability.
3. Genes with highest predictive strength for each class are selected equally to generate a final list of best predictor genes. The final number of best predictors is user-specified.

Each sample from the training and test set will be represented as a vector mapped in terms of the normalized expression intensity of these final predictor genes.

Figure 5 and 6 below show the expression intensity of 1 gene across each ALL and AML samples in the training set. Gene shown in figure 5 is an example of a good predictor. A cutoff mark can be drawn that distinguishes the expression intensity for ALL versus AML samples. Thus, if a test sample has an expression intensity for this gene that falls above the cutoff line, the predictor will classify it as being an AML sample.

Figure 6 shows the expression intensity for a gene that is not a good predictor. No cutoff can distinguish the AML from the ALL group.

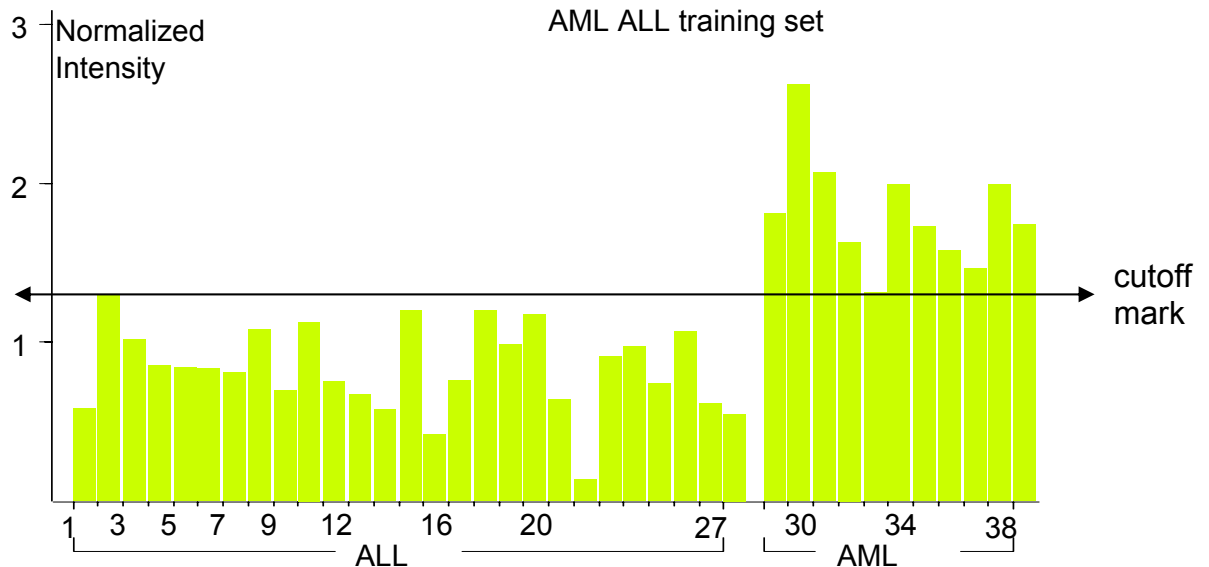


Fig 5: Example of a good predictor gene

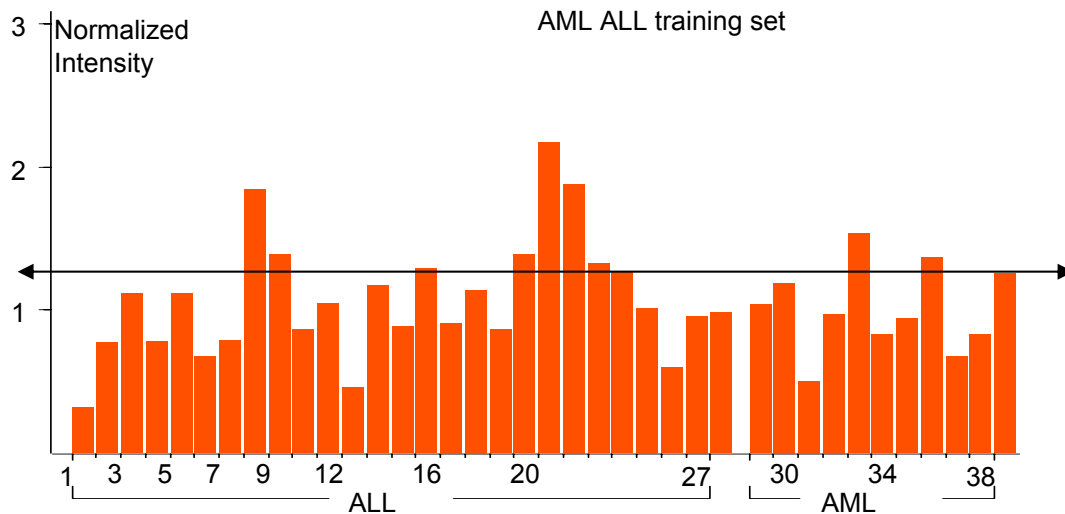


Fig 6: Example of a bad predictor gene

V. How does the algorithm predict test samples?

The algorithm uses genes with highest prediction strength (determined using data from training set) and k-nearest neighbor rule to classify the test sample. The number of k-nearest neighbors is user-defined.

Each sample from the test set is classified by finding the k-nearest neighboring training samples based on the Euclidean distance of normalized expression intensity. The class membership of k-nearest neighbors is enumerated and p values are computed to determine the likelihood of seeing at least the observed number of neighbors from each class relative to the whole training set by chance. The class with the smallest p value is compared to the class with the second smallest p value.

a. K-nearest Neighbor algorithm:

1. Counts the k-nearest samples (in Euclidean distance) in the training set to the new sample to be classified.
2. Determines the proportion of neighbor samples from each class and makes a 'vote' for each class.
3. Calculates p-values for the likelihood of observed representation of each class.
4. Computes the ratio between the p-value of the most highly represented class and the p-value of the next most highly represented class.
5. Allows "no prediction" result if differential between p-values is above Decision cutoff for P-value ratio

Training Set: AML ALL training set All Samples (mode: Log)
 Test Set: AML ALL independent set All Samples (mode: Log)
 Initial Gene List: all genes (7,129 genes)

	Condition	True Value	Prediction	P value ratio	ALL votes	ALL P value	AML votes	AML P value
1	data_set_ALL_AML_independent.txt column 3	ALL	ALL	0.0178	10	0.0178	0	1
2	data_set_ALL_AML_independent.txt column 5	ALL	ALL	0.0178	10	0.0178	0	1
3	data_set_ALL_AML_independent.txt column 15	ALL	ALL	0.0178	10	0.0178	0	1
4	data_set_ALL_AML_independent.txt column 7	ALL	ALL	0.0178	10	0.0178	0	1
5	data_set_ALL_AML_independent.txt column 17	ALL	ALL	0.0178	10	0.0178	0	1
6	data_set_ALL_AML_independent.txt column 19	ALL	ALL	0.0178	10	0.0178	0	1
7	data_set_ALL_AML_independent.txt column 23	ALL	ALL	0.0178	10	0.0178	0	1
8	data_set_ALL_AML_independent.txt column 11	ALL	ALL	0.0178	10	0.0178	0	1
9	data_set_ALL_AML_independent.txt column 37	ALL	ALL	0.129	9	0.127	1	0.982
10	data_set_ALL_AML_independent.txt column 39	ALL	ALL	0.0178	10	0.0178	0	1
11	data_set_ALL_AML_independent.txt column 41	ALL	ALL	0.0178	10	0.0178	0	1
12	data_set_ALL_AML_independent.txt column 21	ALL	ALL	0.0178	10	0.0178	0	1
13	data_set_ALL_AML_independent.txt column 9	ALL	ALL	0.129	9	0.127	1	0.982
14	data_set_ALL_AML_independent.txt column 13	ALL	ALL	0.0178	10	0.0178	0	1

32 correct predictions, 1 incorrect predictions, 1 not predicted

Save Predictor Genes Hide Details Copy to Clipboard Get Text Description... Close

Fig 7: Prediction Results window

b. Interpreting the Prediction Results window

Condition: Name of a sample/condition

True Value: Known class membership of a test sample. This value may or may not be available.

Prediction: Predicted class membership of a sample.

P value ratio: Ratio of the smallest p value over the second smallest p value. a. If calculated p value ratio is less than the **Decision cutoff for P-value ratio**, make prediction for the class with the lowest p value (the numerator class)

- i. If **True Value** of sample matches with predicted class ⇒ consider this Correct Prediction
- ii. If **True Value** of sample does not match with predicted class ⇒ consider this Incorrect Prediction
- iii. If calculated p value ratio is more than the **Decision cutoff for P-value ratio**, do not make prediction ⇒ consider this Not Predicted

ALL votes: The number of nearest training samples to the test sample with ALL phenotype. Each neighboring training sample contributes one vote for its own class membership. Thus, a neighboring ALL sample will contribute one vote for ALL class.

ALL P value: The probability that this many K-nearest neighbors with ALL phenotype were found by chance.

AML votes: The number of nearest training samples to the test sample with AML phenotype.

AML P value: The probability that this many K-nearest neighbors with AML phenotype were found by chance.



Correct Predictions: The predicted class membership matches with the known class membership for that sample.

Incorrect Predictions: The predicted class membership does not match with the known class membership for that sample.

Not Predicted: The ratio of the smallest p value class over the second smallest p value class is larger than the user-specified **Decision cutoff for P-value ratio**. This is equivalent to saying there is not enough evidence to predict one class over another class.

Save Predictor Genes: Saves the list of predictor genes and their associated predictor strengths.

c. Illustration of the prediction and k-nearest neighbors concepts:

The settings for the **Predict Parameter Window** for this example are as follows:

Number of Predictor genes: 3

Number of neighbors: 6

Decision cutoff P-value ratio: 0.2

There are 27 ALL training samples (represented by red circles) and 11 AML samples (represented by yellow spheres)

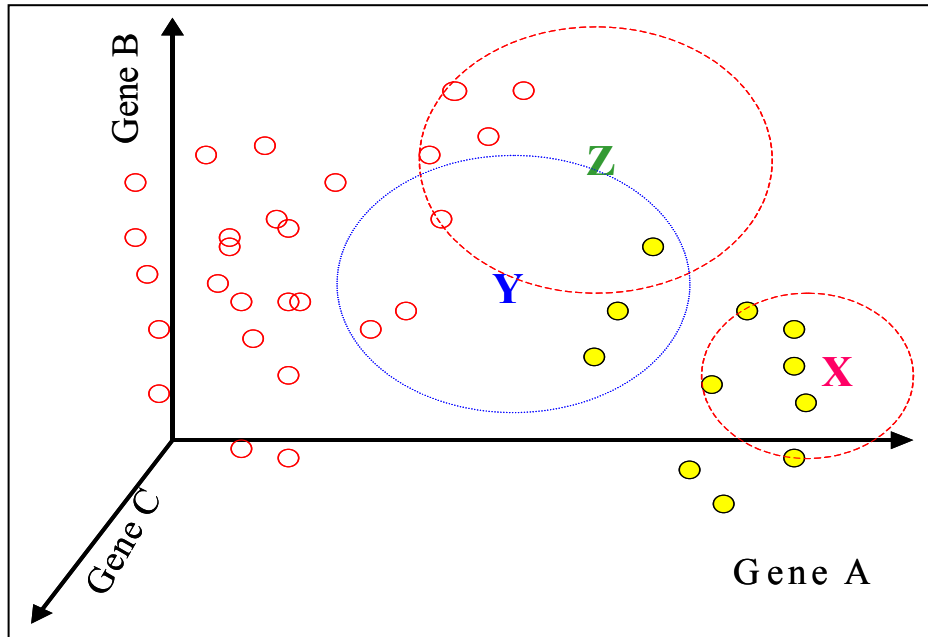


Fig 9: K-nearest neighbors: Each sample is plotted according to its expression intensity for 3 best predictor genes (gene A, B, and C)

Test Sample	ALL vote	ALL p-value	AML vote	AML p-value	P value ratio	Prediction
X	0	1	6	0	0 (0.0 is less than decision cutoff p-value ratio, 0.2 ⇒ predicts sample X as AML)	AML
Y	3	.953**	3	.221**	.232 (0 .232 is higher than decision cutoff p-value ratio, 0.2 ⇒ do not make prediction for sample Y)	Not predicted
Z	5	.429	1	.893	.480 (0.480 is higher than decision cutoff p-value ratio, 0.2 ⇒ do not make prediction for sample Z)	Not predicted

** Notice that both AML and ALL class contributed 3 votes, but their p-values are different. This is because p-value calculation is based on the number of neighbors from each class **relative** to its proportion in the entire training set, not on the absolute number of neighboring samples.

VI. What is the purpose of cross-validation and how does it work?

Cross validation tests how well predictor genes and the prediction rules are at discriminating between classes.

The cross-validation method, also known as drop-one-out or jack-knife approach, removes one sample from the training set and uses it as a test sample. The remaining training samples are used to predict the removed test sample**.

Example of Cross-validation in AML ALL data set (38 training samples total):

1. Remove one leukemia sample
2. Predict the membership of the removed leukemia sample using the prediction rule and data from the remaining 37 training samples
3. Return removed sample back to training set. Remove another sample
4. Repeat step 2 and 3 until all samples have been predicted

** This leave-one-out approach detects samples that have different expression from other samples in the same group. Thus, potential outlier samples can be detected during crossvalidation.

	Condition	True Value	Prediction	P value ratio	ALL votes	ALL P value	AML votes	AML P value
1	data_set_ALL_AML_train.txt column 15	ALL	ALL	0.0152	10	0.0152	0	1
2	data_set_ALL_AML_train.txt column 25	ALL	AML	0.0225	4	0.997	6	0.0224
3	data_set_ALL_AML_train.txt column 27	ALL	ALL	0.0152	10	0.0152	0	1
4	data_set_ALL_AML_train.txt column 17	ALL	ALL	0.0152	10	0.0152	0	1
5	data_set_ALL_AML_train.txt column 31	ALL	ALL	0.0152	10	0.0152	0	1
6	data_set_ALL_AML_train.txt column 37	ALL	ALL	0.116	9	0.114	1	0.985
7	data_set_ALL_AML_train.txt column 53	ALL	ALL	0.0152	10	0.0152	0	1
8	data_set_ALL_AML_train.txt column 55	ALL	ALL	0.0152	10	0.0152	0	1
9	data_set_ALL_AML_train.txt column 3	ALL	ALL	0.0152	10	0.0152	0	1
10	data_set_ALL_AML_train.txt column 33	ALL	ALL	0.0152	10	0.0152	0	1
11	data_set_ALL_AML_train.txt column 35	ALL	ALL	0.116	9	0.114	1	0.985
12	data_set_ALL_AML_train.txt column 43	ALL	ALL	0.0152	10	0.0152	0	1
13	data_set_ALL_AML_train.txt column 45	ALL	ALL	0.0152	10	0.0152	0	1
14	data_set_ALL_AML_train.txt column 49	ALL	ALL	0.0152	10	0.0152	0	1
15	data_set_ALL_AML_train.txt column 51	ALL	ALL	0.116	9	0.114	1	0.985

34 correct predictions, 2 incorrect predictions, 2 not predicted

Hide Details Copy to Clipboard Get Text Description... Close

Fig 10: Detailed **Crossvalidation Results** window. To view an abridged version of the Crossvalidation Result window, select Hide Details.

Interpreting the **Crossvalidation Results** window:

All columns (**Condition, True Value, Prediction, P value ratio, ALL votes, ALL P value, AML votes and AML P value**) are interpreted the same as the **Prediction Results** window. The only difference is that the **True Value** for each of these samples will always be known.

VIII. Most commonly asked questions about Class Prediction:

Q. How can I obtain the name of the best predictor genes?

- A. Set training set **and** test set, and run the Predict Test Set command. Select **Save Predictor Genes** in the **Prediction Results** window, and name the resulting gene list. To view the names of the predictor genes and their associated predictor strength:
1. Open the **Gene List Inspector** window (by double-clicking on the gene list or right-click and choose **Inspect**)
 2. Choose **View** ⇒ **Ordered List**. Predictor genes will be displayed according to prediction strength. Zoom in to view the associated names and associated values.
- To export out the predictor genes names and associated prediction strengths:
1. Drag and drop the gene list into an spreadsheet file
 2. Selects the gene list, go to **Edit** ⇒ **Copy** ⇒ **Copy Annotated Gene List**. Select **Gene List Associated Value** and copy the information to clipboard. Paste this information into a spreadsheet.

Q. Do we use the same algorithm as in Golub et al. (1999)?

- A. No, the algorithm we use is not the same algorithm as in Golub et al. (1999), but our own. The main difference is that we use k-nearest neighbor as the classification rule. Neighbors are samples selected from the training set according to the Euclidean distance of the expression profiles of predictor genes.

Q. What are the benefits of using k-nearest neighbor over other discrimination methods?

- A. It is flexible and can be applied to situations where there are more than 2 classes to discriminate. The decision threshold can be adjusted according to the relative costs of incorrect predictions versus no prediction. We have found that it does better than the predictor mentioned in Golub et al. (1999) publication on the AML ALL dataset mentioned in this document.

Q. What is the minimum number of training samples I should have per class?

- A. The Class Predictor was designed for experiments with 20 or more samples per class. If you have fewer samples than this, you can increase the decision cutoff p-value ratio to decrease the stringency for making prediction.

Q. Do I need a test set to get a list of predictor genes?

- A. No, you can get the list of predictor genes without a true test set by designating your training samples as both the training set and test set. Select **Predict Test Set** once it becomes enabled. In the Results window, select **Save Predictor Genes**.

Q. Does cross validation alter the list of predictor genes?

A. No, the crossvalidation process does not alter the list of predictor genes. Prediction strength for each gene is calculated using all samples in the training set. A user-defined number of best predictor genes are selected and this list of predictor genes are used for both crossvalidation and predict test set process.

References:

Hastie, T. et al. The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2001). Springer.

Golub, T.R. et al. "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring". Science, v 286, pp 531-537 (1999).