Tutorial 14: Classification -- K-Nearest Neighbors (KNN) and Linear Discriminant Analysis (LDA)
What is KNN?

The K-nearest neighbors algorithm is a method for classifying objects based on closest training examples in the feature space.

**KNN algorithm:**
K-Nearest Neighbors (KNN) classification divides data into a test set and a training set. For each row of the test set, the K nearest (in Euclidean distance) training set objects are found, and the classification is determined by majority vote with ties broken at random. If there are ties for the Kth nearest vector, all candidates are included in the vote.
Launch KNN in ArrayTrack

First select the experiment, right-click, then choose “Select datasets” ⇒ “Select raw datasets”. In the pop-up window, click “OK” button.
Launch KNN in ArrayTrack

The selected data are highlighted, then right-click, select “Analysis” ⇒ K-Nearest Neighbors.
Launch KNN in ArrayTrack

Click the triangle button to select the gene list. In this example select “95gene_KEGG0.01”. User can also load external gene list by click “Browse” button.
Launch KNN in ArrayTrack

The gene list is selected, then click “OK” button.

The next window shows the data is being exported to do “Build KNN”. In the Action Select window, make sure “Model build” is selected, then click “OK” button.
Launch KNN in ArrayTrack

In the Sample Classes Assignment window, the left panel lists all the unassigned samples, and the right panel shows the classes of assigned samples. At the top of left side there are two check boxes: Column-wised Samples and Sort.

- Column-wised samples checkbox: In the table view of data set, one column represents one sample if checked; otherwise, one row represents one sample.
- Sort button: Sort unassigned samples for ease of selection.
Launch KNN in ArrayTrack

If “Reverse Selection” button is clicked, then all the samples will be selected. User can also select by patterns by typing the character in the text box, then all the samples that meet the pattern criteria will be highlighted.

--New Class button: assign the highlighted samples in the left panel into a new class.
--Right arrow button: Assign highlighted unassigned samples (in left panel) to the class highlighted in the right panel.
--Left Arrow Button: Remove selected samples from right panel.
--Reset button: Clears all assigned samples.
Launch KNN in ArrayTrack

In the “Select by pattern” textbox, type “N0”, all the sample with “N0” suffix will be highlighted, then click “New Class” button.
Launch KNN in ArrayTrack

Now, all the “N0” data is assigned to class 1 in the right panel. In the “Select by pattern” textbox, type “N1”, all the sample with “N1” suffix will be highlighted, then click “New Class” button to assign them to class2.
Launch KNN in ArrayTrack

The right panel lists the assigned samples. Class 1 has 22 samples and class 2 has 50 samples. Click “Next” button.
Launch KNN in ArrayTrack

KNN option dialogue:

1. Number of neighbors (K):
   • Predefined: select the number of Neighbors.
   • Dynamic search: specify a range to search for an optimal K using the Leave-one-out validation method.

2. Validation options:
   • Fitting
   • Leave-one-out
   • K-fold
   • Monte Carlo
   • Customized
Launch KNN in ArrayTrack

ArrayTrack gives a performance report and a confusion table when building a KNN model. There are 4 validation methods available in ArrayTrack for a predefined K specified by the user:

- **Leave-one-out**: Uses a single sample as the validation data with the rest of the samples as training data for testing the model. Each sample is used exactly once as the validation data.
- **K-Fold**: Partitions the original sample into K subsamples. Each random subsample is used as the validation data while the rest is used as training data once each. The user can change the value of K, amount of runs, and the random seed. Additional runs may result in different subsample partitions, giving the total algorithm higher accuracy.
- **Monte Carlo**: Randomly selects the specified number of samples as test set, and builds the KNN on remaining samples. The user can change the number of test samples, runs, and the random seed.
- **Customized**: Allows the user to specify the training samples and the test samples.

Additionally, ArrayTrack supports searching for an optimal K within a range specified by the user using the Leave-one-out validation method.
Launch KNN in ArrayTrack

In the KNN options window, the default $K=5$. Select “Fitting” for validation method. Then click “Finish” button.
Classification GUI

The left panel displays the tree hierarchically. There are six types of tree nodes: Project, Data, Files, Model, Classifiers and Predictions. The right panel displays the model information report.

Classification GUI

The number of neighbors: K = 5
Validation Method: Fitting

Confusion Table:

<table>
<thead>
<tr>
<th>Actual</th>
<th>Class 1 (Negative)</th>
<th>Class 2 (Positive)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class 1 (Negative)</td>
<td>20</td>
<td>2</td>
</tr>
<tr>
<td>Class 2 (Positive)</td>
<td>4</td>
<td>46</td>
</tr>
</tbody>
</table>

Performance Report:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>91.67%</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>92.00%</td>
</tr>
<tr>
<td>Specificity</td>
<td>90.91%</td>
</tr>
<tr>
<td>Prevalence</td>
<td>69.44%</td>
</tr>
<tr>
<td>Positive predictiv</td>
<td>95.83%</td>
</tr>
<tr>
<td>Negative predictiv</td>
<td>83.33%</td>
</tr>
<tr>
<td>False positive rate</td>
<td>9.09%</td>
</tr>
<tr>
<td>False negative rate</td>
<td>8.00%</td>
</tr>
<tr>
<td>Matthews' correlation coefficient</td>
<td>0.81</td>
</tr>
<tr>
<td>Area under ROC curve</td>
<td>0.91</td>
</tr>
</tbody>
</table>

Prediction:

<table>
<thead>
<tr>
<th>Name of Sample</th>
<th>Class ID (Actual)</th>
<th>Class ID (Predicted)</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAA1G1H1J1H1</td>
<td>2</td>
<td>2</td>
<td>1.0</td>
</tr>
<tr>
<td>TAA1G1H2J2H1</td>
<td>2</td>
<td>2</td>
<td>1.0</td>
</tr>
<tr>
<td>TAA1G1H3J3H1</td>
<td>2</td>
<td>2</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Model Information Report

The report includes the model name, the optimum number of neighbors (K), validation strategy, confusion table, performance metrics (if number of classes is less than three), and predictions. The reported performance metrics include accuracy, sensitivity, specificity, prevalence, positive predictivity, negative predictivity, false positive rate, false negative rate, Mathew’s correlation coefficient (MCC), and area under ROC curve (AUC).

<table>
<thead>
<tr>
<th>Actual</th>
<th>Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Class 1</td>
</tr>
<tr>
<td>Class 1</td>
<td>TN</td>
</tr>
<tr>
<td>Class 2</td>
<td>FN</td>
</tr>
</tbody>
</table>
Launch KNN in ArrayTrack

In slide #14, user can select validation option “Leave-one-out”, then click “Finish” button.
Launch KNN in ArrayTrack

Here is the model information for “Leave-one-out cross-validation”.

The model name can be changed by right-click and choose “Rename”.
Launch KNN in ArrayTrack

In slide #14, user can select “K-fold” for validation method. The default values for folds, random seed and runs are already filled. If user doesn’t want to change the default value, click “Finish” button.
Launch KNN in ArrayTrack

Here is the model information for “K-Fold cross-validation”.

![Cross Validation and Fitting Report](image)

**Name:** 5-Fold cross-validation

The number of neighbors: K = 5

Validation Method: 5-Fold cross-validation, run: 1

**Confusion Table:**

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class 1 (Negative)</td>
<td>18</td>
</tr>
<tr>
<td>Class 2 (Positive)</td>
<td>6</td>
</tr>
</tbody>
</table>

**Performance Report:**

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>86.11%</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>88.00%</td>
</tr>
<tr>
<td>Specificity</td>
<td>81.82%</td>
</tr>
<tr>
<td>Prevalence</td>
<td>69.44%</td>
</tr>
<tr>
<td>Positive predictivity value</td>
<td>91.67%</td>
</tr>
<tr>
<td>Negative predictivity value</td>
<td>75.00%</td>
</tr>
<tr>
<td>False positive rate</td>
<td>18.18%</td>
</tr>
<tr>
<td>False negative rate</td>
<td>12.00%</td>
</tr>
<tr>
<td>Matthews’s correlation coefficient</td>
<td>0.68</td>
</tr>
<tr>
<td>Area under ROC curve</td>
<td>0.85</td>
</tr>
</tbody>
</table>

**Prediction:**

<table>
<thead>
<tr>
<th>Name of Sample</th>
<th>Class ID (Actual)</th>
<th>Class ID (Predicted)</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>(TAA_L_6FL_1_N1)</td>
<td>2</td>
<td>2</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Predict Unknown Samples

Once the model is built, it can be used to predict the classes of unknown samples based on the KNN algorithm. It can be invoked in two ways:

• For samples already in ArrayTrack database panel.
• For unknown samples that are stored in a tab-delimited text file, right click on the classifier and select the “Load data & predict…” command from popup menu.

(see next slides for detail)
Predict Unknown Samples

For samples already in ArrayTrack database panel, select experiment “Classification-Testing Set”, right-click, choose “Select datasets” ⇒”select raw datasets”. In the pop-up window, click “OK” button.
Predict Unknown Samples

With all the data highlighted, right-click, choose “analysis” ⇒ “K-Nearest Neighbors”.
Predict Unknown Samples

Select the same gene list “95gene_KEGGP0.01”, click “OK” button.

Select “Predict”, then click “OK” button.
Predict Unknown Samples

Select all the samples, click arrow button.
Predict Unknown Samples

Make sure the Model KNN1 is selected, click “OK” button.
Predict Unknown Samples

Prediction result: the right panel displays the report for Prediction 1.
Calculate Prediction Performance

Right-click on a prediction and select “Calculate performance” to proceed to the Test Sample Classes Assignment dialogue. The performance can be calculated only when the test sample identity is known. The prediction also can be renamed or deleted by right-clicking the prediction.

Assign a class for each test sample: In the “Select by pattern” textbox, type “N0”, all the sample with “N0” suffix will be highlighted, then click “New Class” button.
Calculate Prediction Performance

In the “Select by pattern” textbox, type “N1”, all the sample with “N1” suffix will be highlighted, then click “New Class” button.
Calculate Prediction Performance

All the samples with “N0” suffix are assigned to class 1, and samples with “N1” suffix are assigned to class 2. Click “Finish” button. The performance report — (Confusion table and performance metrics table) will be added to “Prediction Results”.
Calculate Prediction Performance

The prediction performance report is displayed.
Linear Discriminant Analysis (LDA)

(LDA) uses an ANOVA-like approach in grouping data samples by attempting to minimize the variance within each group, and maximize the variance between groups.
Launch LDA

The procedures of doing LDA is similar with KNN.
Select the dataset (see slide #3), right-click, then choose “Analysis” ⇒ LDA.

Select gene list “20Gene_KEGGP0.01_subset”. The number of genes in the list has to be much smaller than the number of samples.
Launch LDA

LDA Options with default parameter. Click “Finish” button.
Save and Open Model

User can save or rename the model by right-clicking the model and choose the right options.

If user try to close KNN without saving the model, a pop-up window will ask user to save it. Click “Yes” button to save to the local drive.
Save and Open Model

To open a saved model, click “File” menu and select “load model file”.