Genesis Prototype Results

Student Number: 483201
Student Name: Gary Bezuidenhout
School of Electrical & Information Engineering, University of the Witwatersrand

01 November 2017

Introduction

This document outlines the testing procedure for the development of the Genesis prototype. Evaluation of system outputs and performance are assessed to determination whether project requirements have been met. Three tests were carried out to achieve this. These tests, associated results and discussions thereof follow.

1 Testing Procedure

Three tests were carried out for the redeveloped Genesis application namely, the PCA plot generation test, memory usage test and the execution time test. Tests were carried out to evaluate whether requirements were met as well as evaluating performance of the prototype. Due to the fact that the application is a redesigned and reengineered version of the original application, all tests were run against the original.

By approaching testing in this manner the original application can be used as a benchmark in all areas. The results can then be determined based on the outputs produced by the original application. Each test had three test cases, these are highlighted below:

- Test Case 1 : PCA plot generated with just data file and PCA's 1 & 2 selected.
- Test Case 2 : PCA plot generated with data file and phenotype file included.
- Test Case 3 : PCA plot generated with data and phenotype file. Different PCA's selected.

Principal Component Analysis Plots

This test was done to evaluate whether the prototype produced the correct output when provided with a specific input. Output plots are compared to output from original application. Figures 1 to 6 illustrate output from the original application and revised application prototype for the above mentioned test cases.

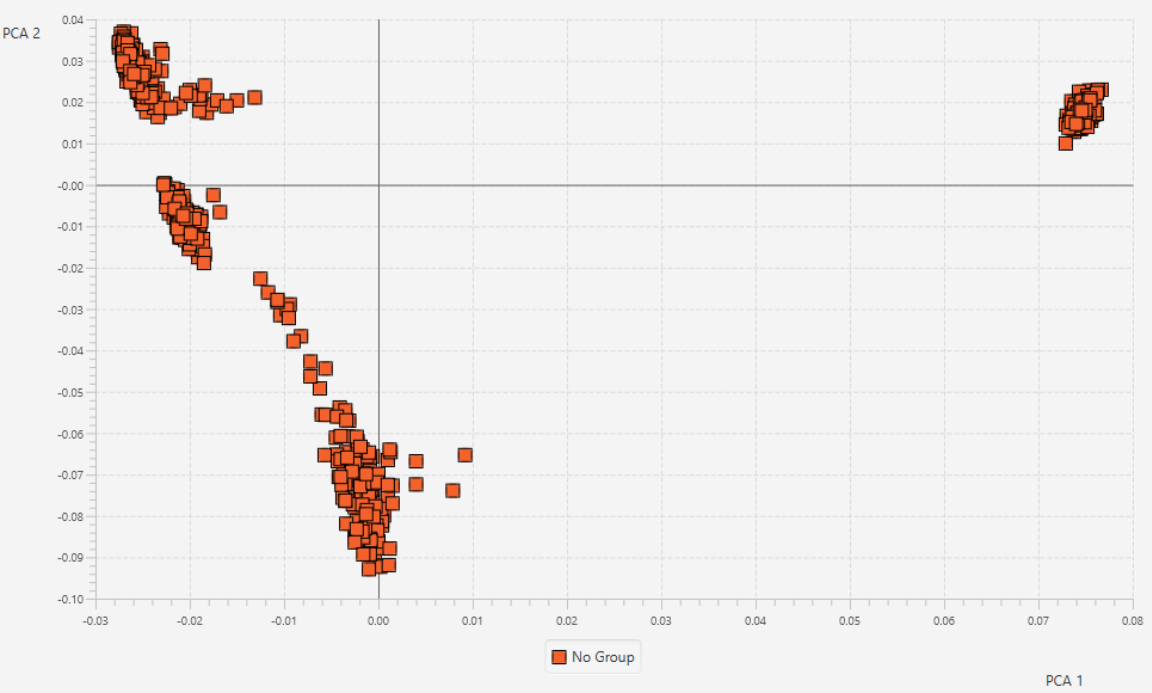

Figure 1: Test Case 1 Output for Genesis Prototype

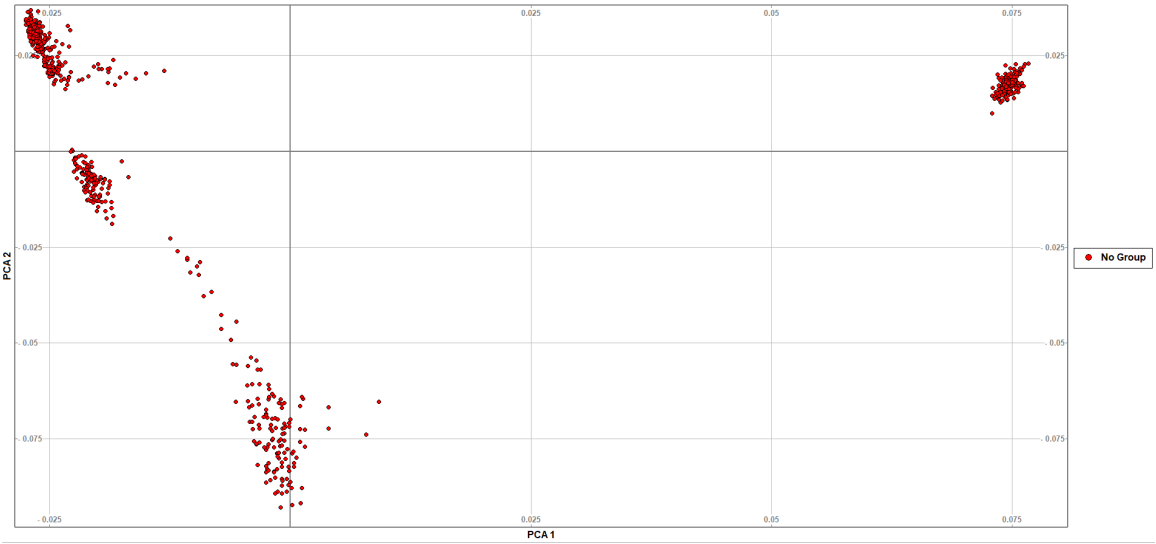

Figure 2: Test Case 1 Output for Original Genesis Application

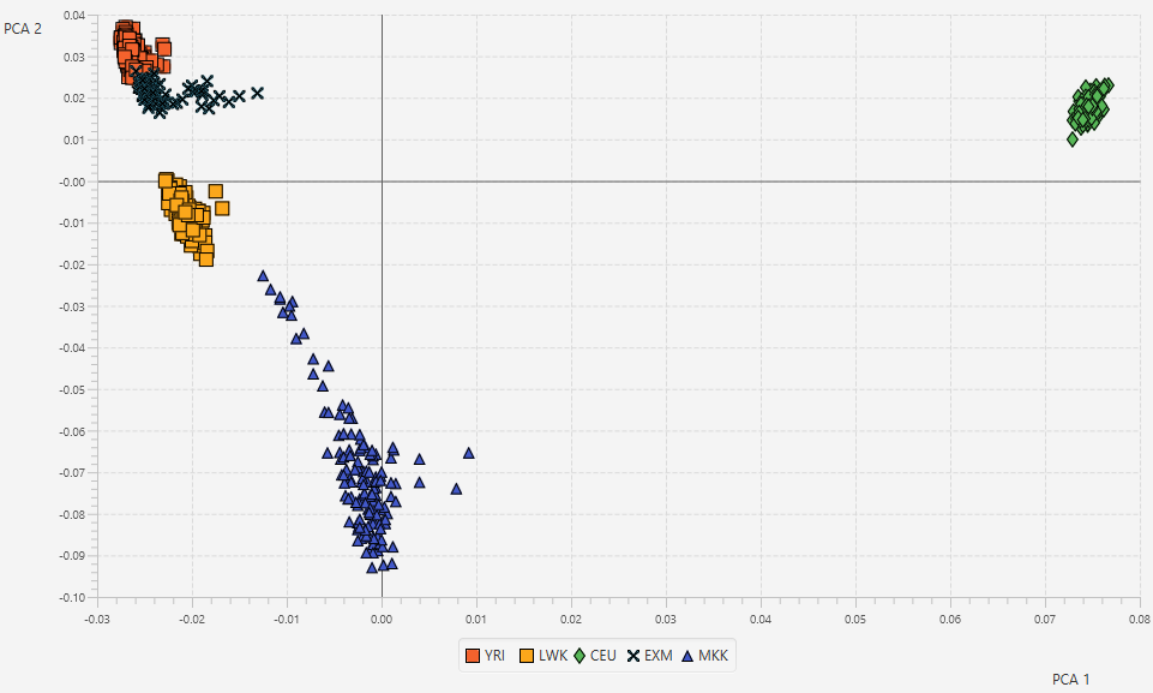

Figure 3: Test Case 2 Output for Genesis Prototype

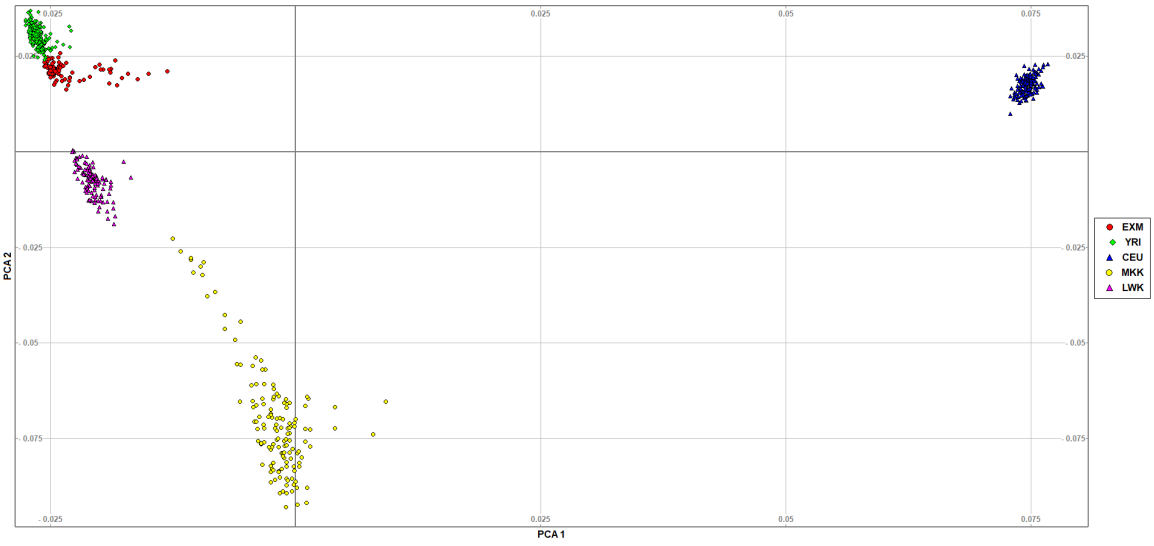

Figure 4: Test Case 2 Output for Original Genesis Application

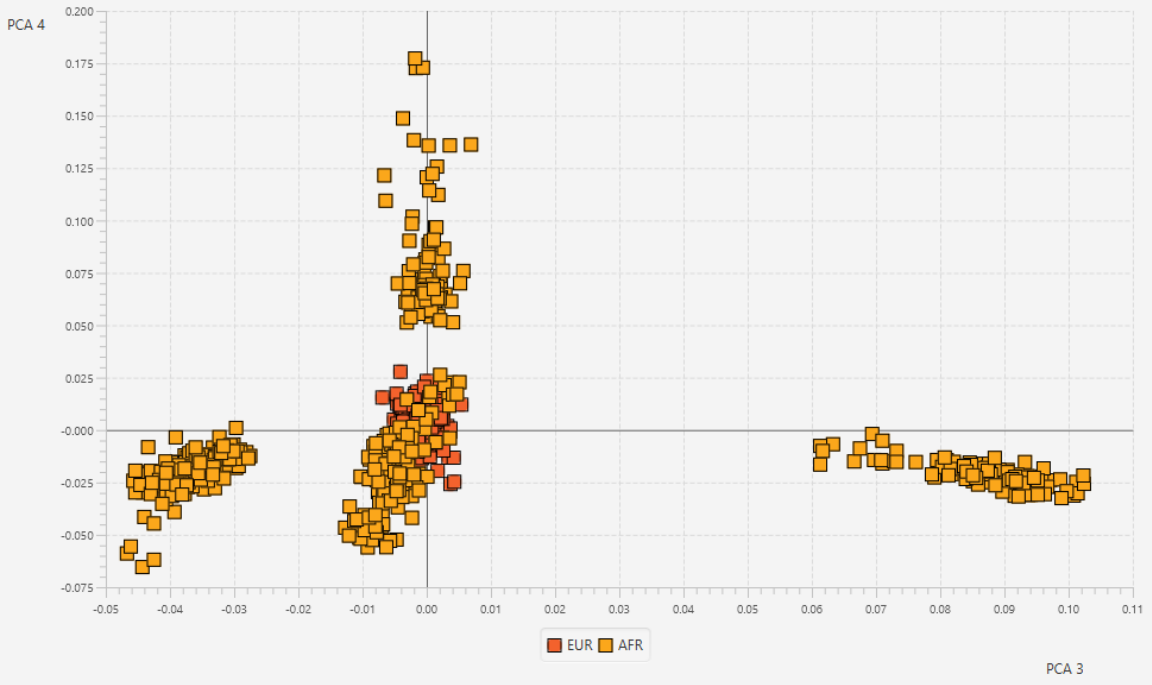

Figure 5: Test Case 3 Output for Genesis Prototype

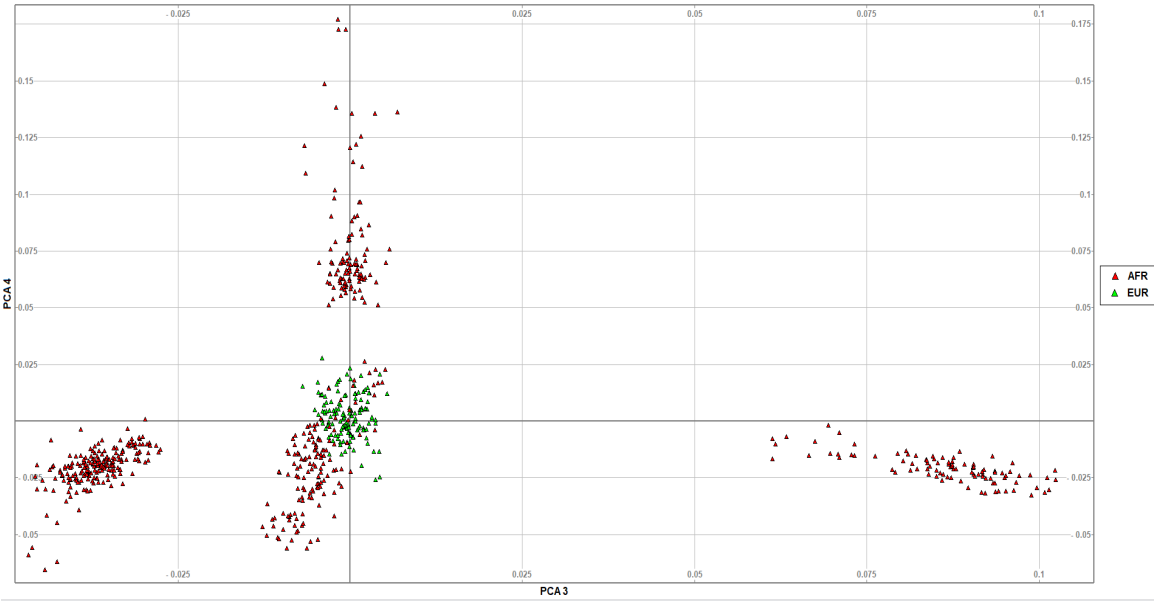

Figure 6: Test Case 3 Output for Original Genesis Application

Memory Usage

To evaluate the performance of the revised application prototype, memory usage was the first measure that was considered. Each one of the previously mentioned tests were carried out three times and the usage for each application was noted. The average was determined and this data is displayed in Table 1 and Figure 7, respectively. It was seen that the original Genesis application performed much better in terms of memory usage. This could be due to the difference in memory usage between SWT and JavaFX.

Memory Usage Data (MB)			
	Test 1	Test 2	Test 3
Genesis 1.0	52.1 54.6 49.2	53.3 54.2 53.9	53.4 55.6 54.1
Genesis 2.0	141.2 141.7 139.8	145.9 144.6 146.6	155.6 153.5 151.2

Table 1: Table displaying values of Memory Usage for each application

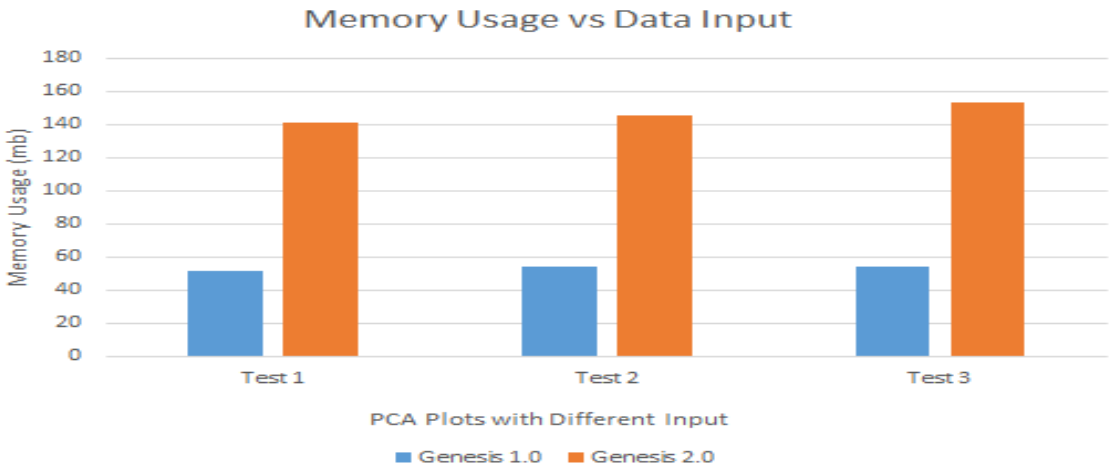

Figure 7: Memory Usage Results for all Test Cases

Execution Time (ms)			
	Test 1	Test 2	Test 3
Genesis 1.0	7158 7265 7214	8603 8692 8790	9875 9514 9635
Genesis 2.0	7082 7159 6987	8205 8536 8397	9136 9460 9257

Table 2: Execution Time Results for all Test Cases

Execution Time

The second measure considered for the performance evaluation of the revised application prototype was execution time. A process similiar to the process followed for the determination of memory usage, was adopted for the determination of execution time. Three values were noted for each test and the average was determined, Figure 8 illustrates the average of these results graphically. The values recorded for each test are shown in Table 2.

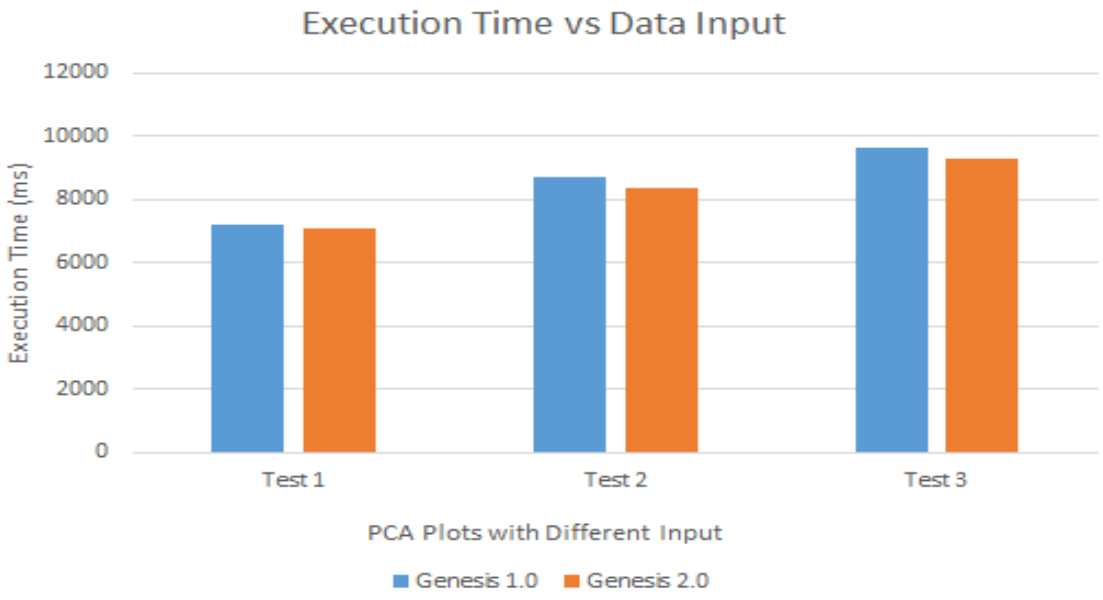

Figure 8: Execution Results for all Test Cases

2 Output of Required & Additional Genesis Features

The following images, illustrate selected functionality required for the Genesis application. These include modification of an icon, as well as adding a label and a line to a plot. An additional feature is also shown, a tooltip was implemented to the revised application prototype. When a user hovers over a point the tooltip will appear displaying information related to that specific point/node.

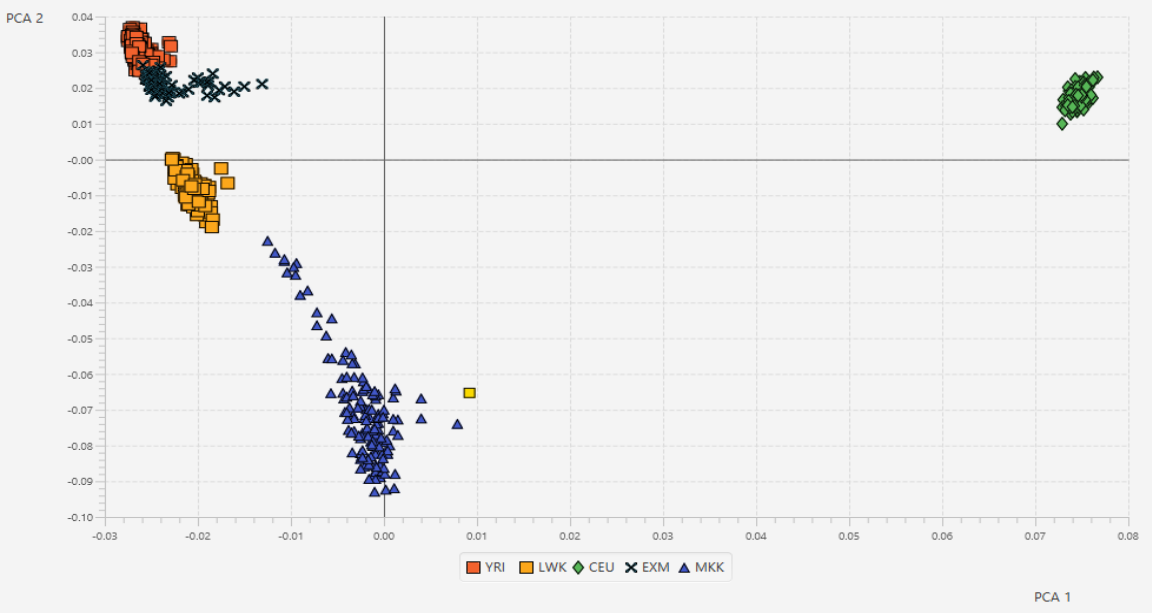

Figure 9: Screenshot displaying icon modification feature

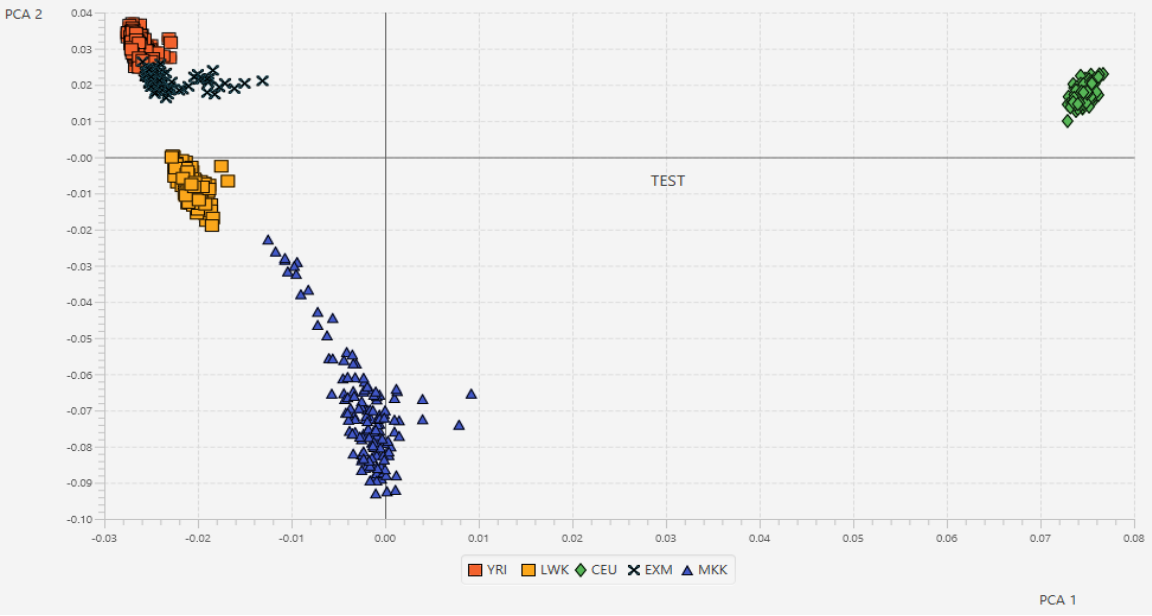

Figure 10: Screenshot displaying label addition feature

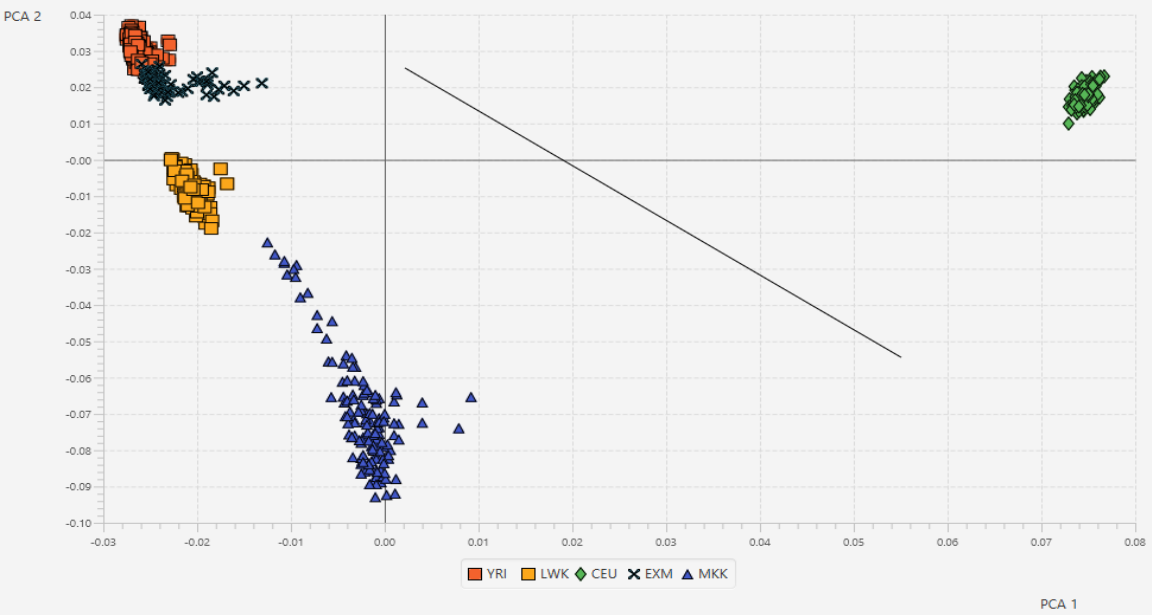

Figure 11: Screenshot displaying line addition feature

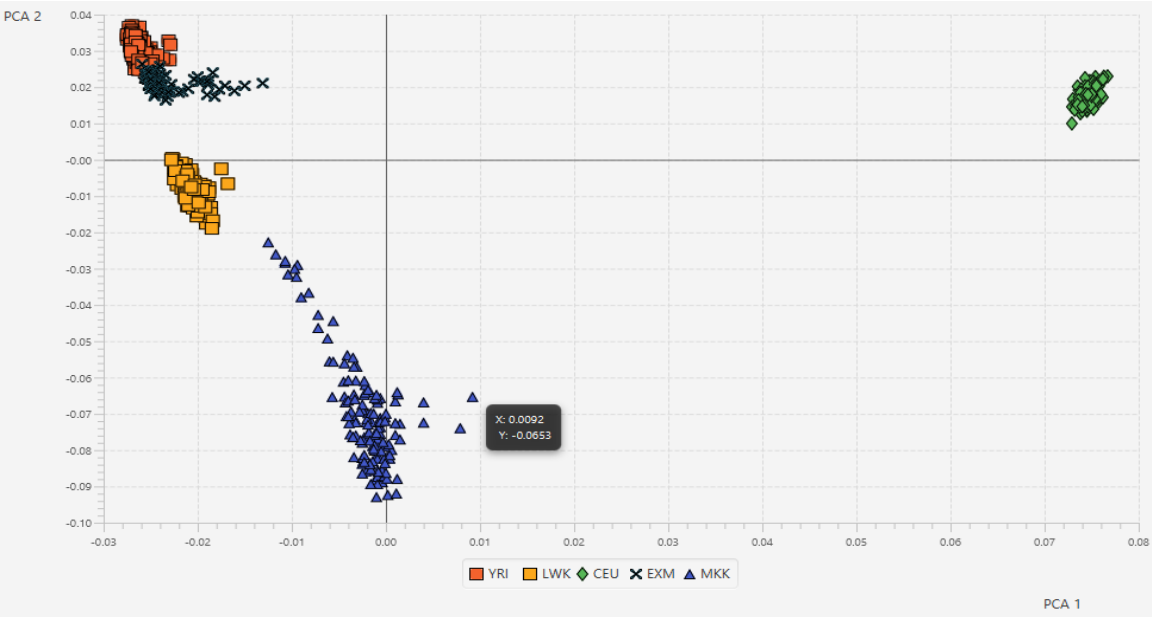

Figure 12: Screenshot displaying new tooltip feature