***Correlation between Code coverage and Cyclomatic complexity***

***Calculating Spearman Correlation***

Spearman’s Rank correlation coefficient is one of the most-prominent technique which can be used to find out the strength and correlation between two variables.

***Method used to calculate the Spearman correlation***

* Create a table from your data and get the ordered pairs of two variables.
* Rank the two data sets. Ranking is achieved by giving the ranking '1' to the biggest number in a column, '2' to the second biggest value and so on. The smallest value in the column will get the lowest ranking. This should be done for both sets of measurements or the variables used to find the correlation for.
* Tied scores are given the mean (average) rank.
* Find the difference in the ranks (d).
* Square the differences (d²) To remove negative values and then sum them
* Calculate the coefficient (***Rs***) using the formula mentioned below.

When written in mathematical notation the Spearman Rank formula looks like this:



Here,

ρ= Spearman rank correlation

di= the difference between the ranks of corresponding variables

n= number of observations

We have used the formula of spearmen correlation coefficient by considering the statement coverage values as X- values and the number of bugs as Y- values.

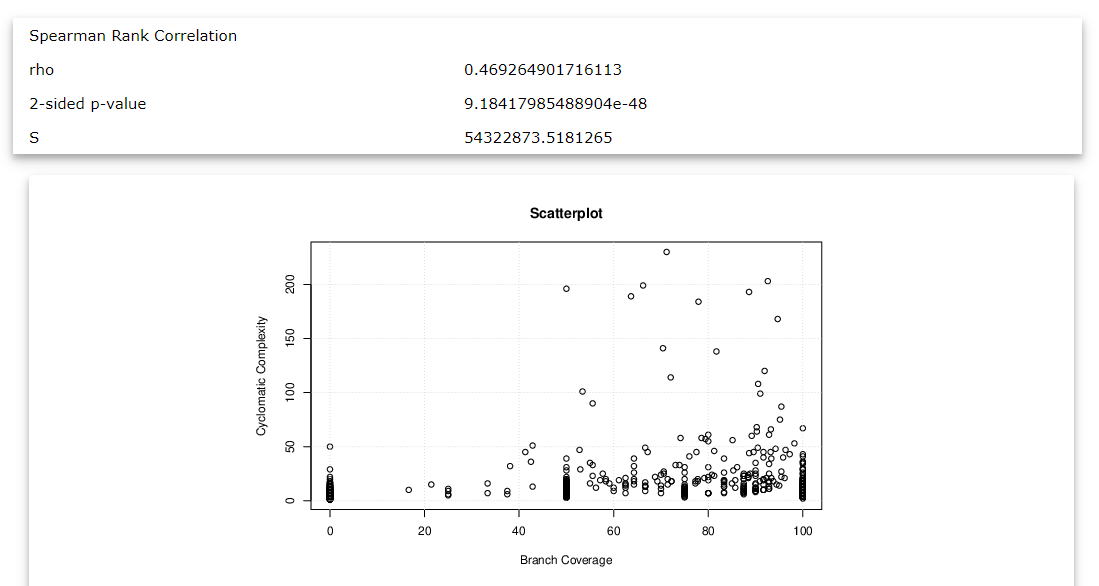
***Calculated Spearman Correlation Coefficient values***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Project*** | ***Statement coverage (1)*** | ***Branch Coverage (2)*** | ***Average Cyclomatic complexity (3)*** | ***Spearman Correlation (1&3)*** | ***Spearman Correlation (2&3)*** |
| ***Apache Commons Collections*** | ***86%*** | ***49%*** | ***14*** | ***-0.278*** | ***0.469*** |
| ***Apache HttpComponents Client*** | ***53%*** | ***23%*** | ***10*** | ***0.0458*** | ***0.497*** |
| ***JFreeChart*** | ***72%*** | ***35%*** | ***24*** | ***-0.506*** | ***0.488*** |
| ***Apache Commons IO*** | ***90%*** | ***45%*** | ***16*** | ***-0.400*** | ***-0.0369*** |
| ***Apache Commons Configurations*** | ***91%*** | ***56%*** | ***17*** | ***-0.369*** | ***0.533*** |

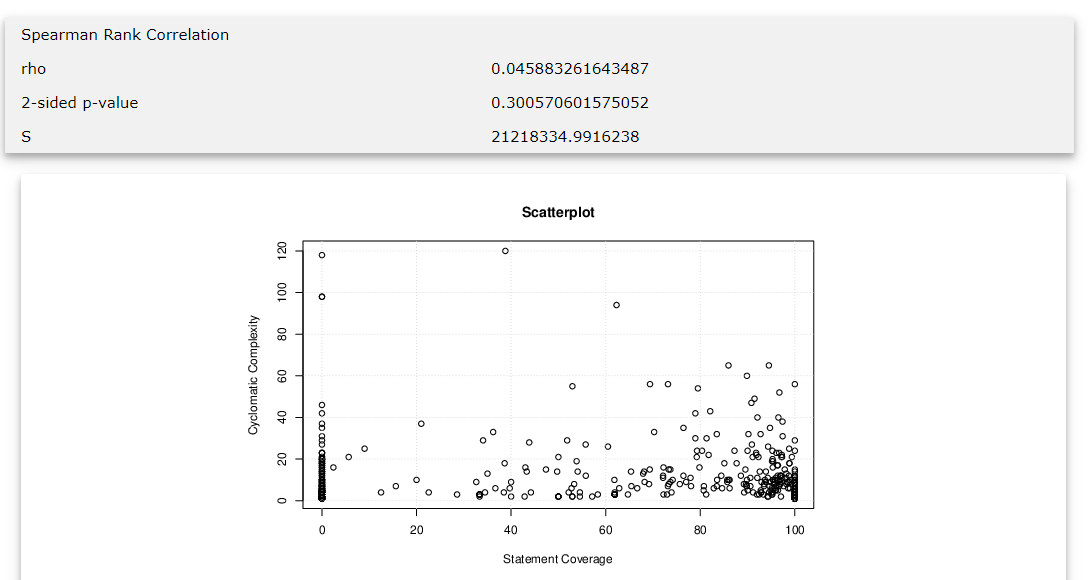
Table-1: Result of Spearman Correlation Coefficient

**Note:** The correlations have been calculated based on the data provided in the folder **correlations ->Metric 1,2 & 4**

**Scatterplots and spearman rank correlation for Apache commons configuration**

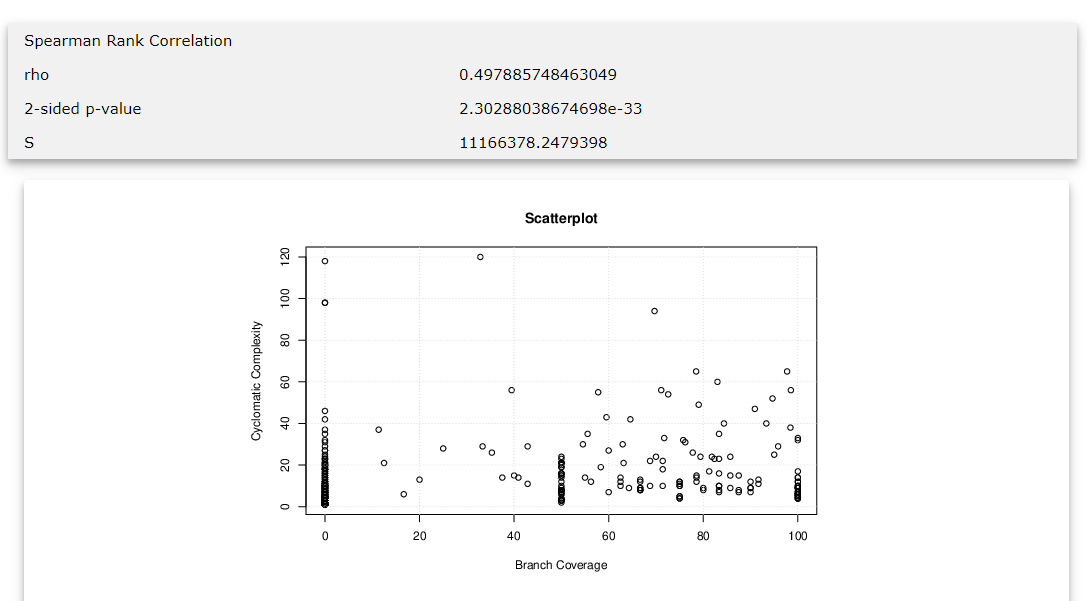


Scatterplot above is mapped between the cyclomatic complexity and branch coverage for Apache Commons Configuration.

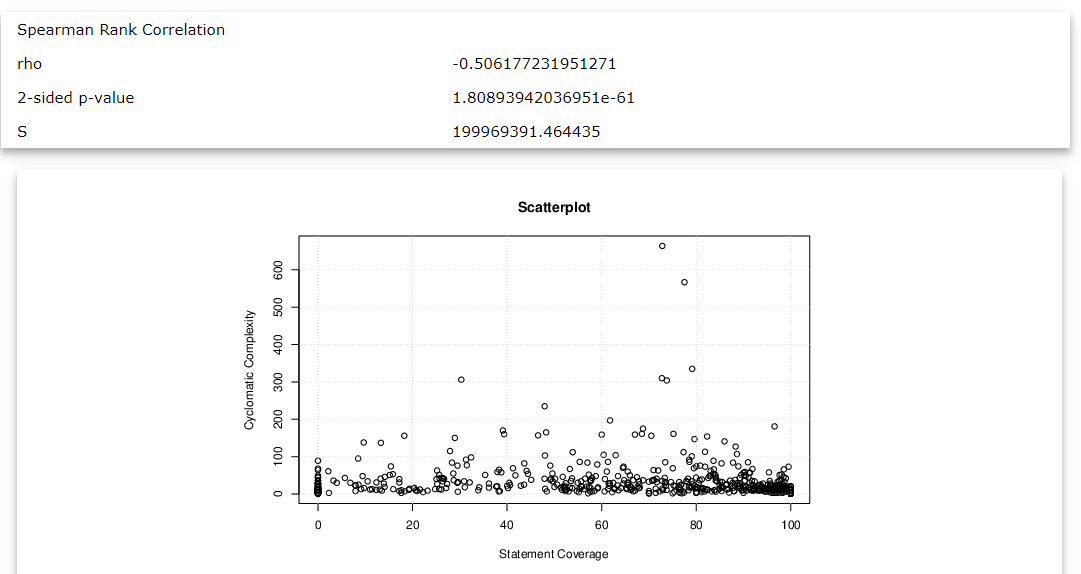


Scatterplot above is mapped between the cyclomatic complexity and statement coverage for Apache Commons Configuration.

**Scatterplots and spearman rank correlation for Apache commonsIO**

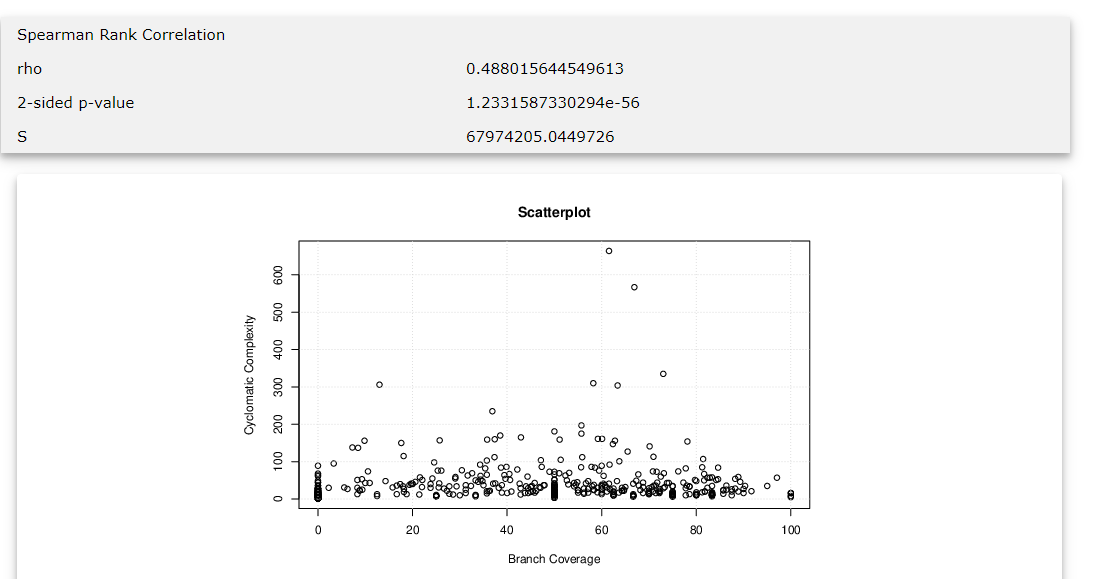


Scatterplot above is mapped between the cyclomatic complexity and branch coverage for Apache Commons IO

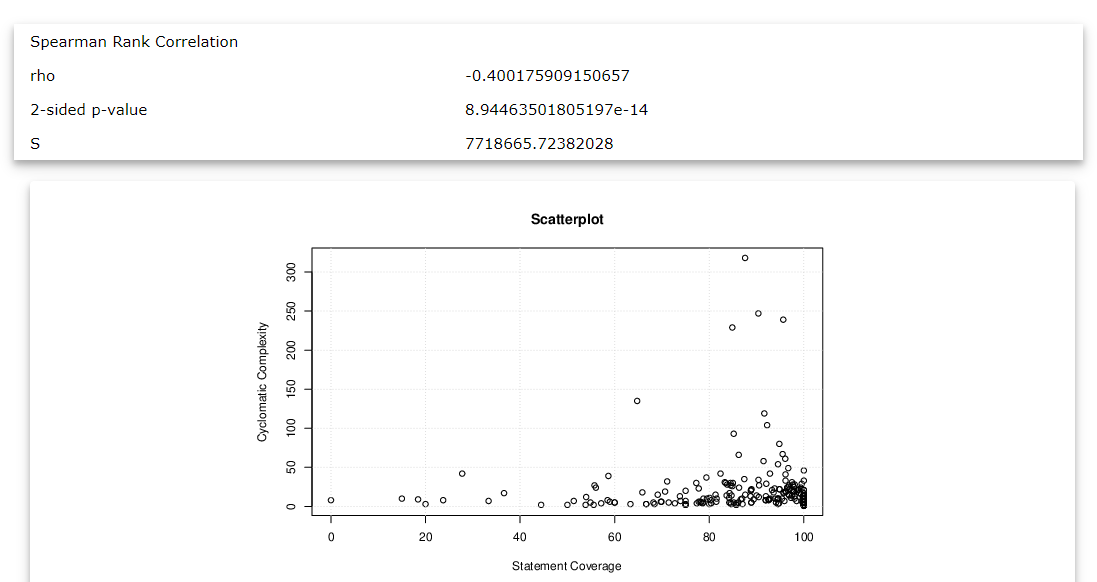


Scatterplot above is mapped between the cyclomatic complexity and statement coverage for Apache Commons IO

**Scatterplots and spearman rank correlation for JfreeChart**

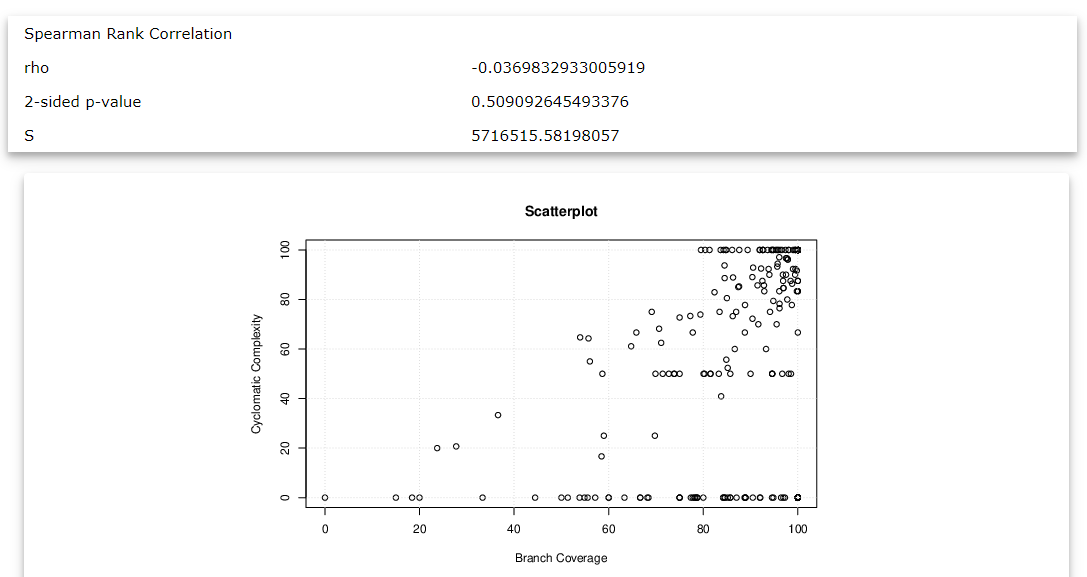


Scatterplot above is mapped between the cyclomatic complexity and branch coverage for JFreechart

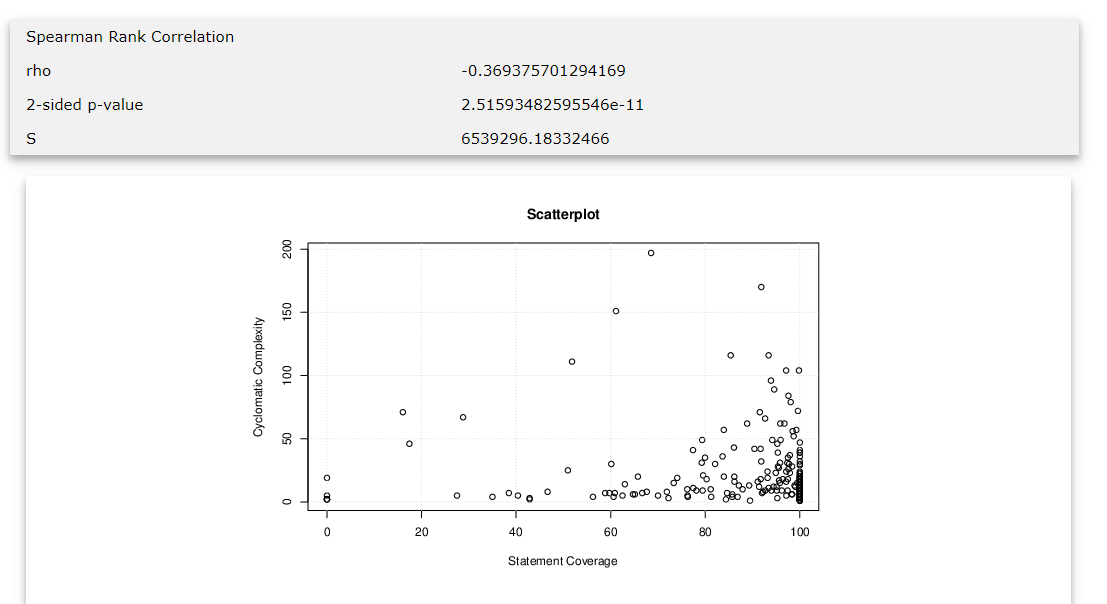


Scatterplot above is mapped between the cyclomatic complexity and statement coverage for JFreechart

**Scatterplots and spearman rank correlation for Apache HttpComponents-client**

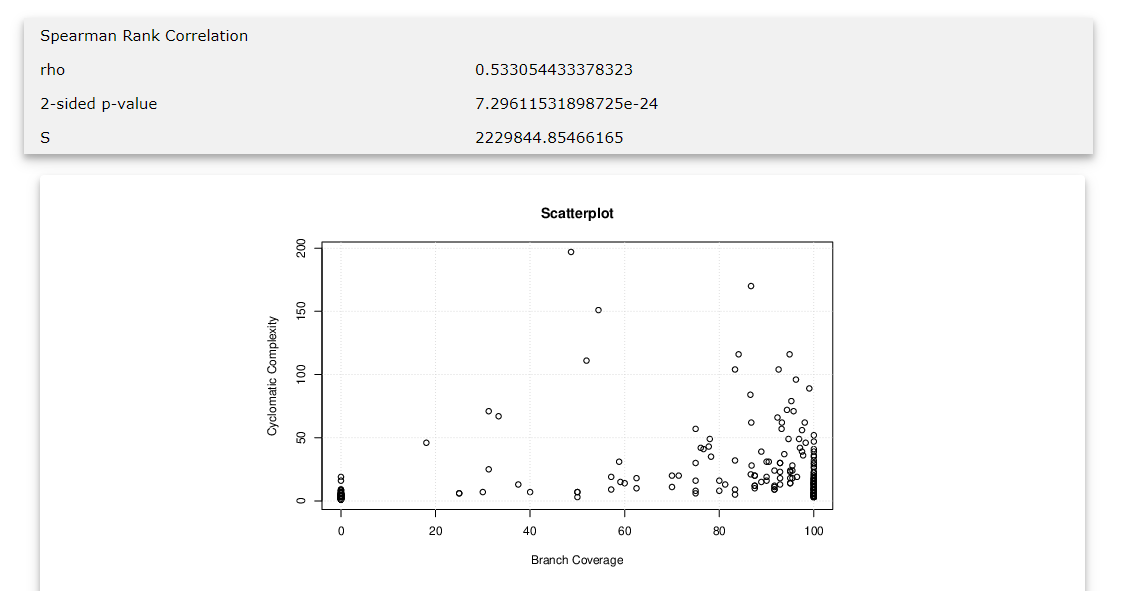


Scatterplot above is mapped between the cyclomatic complexity and branch coverage for Apache HTTPComponents-Client

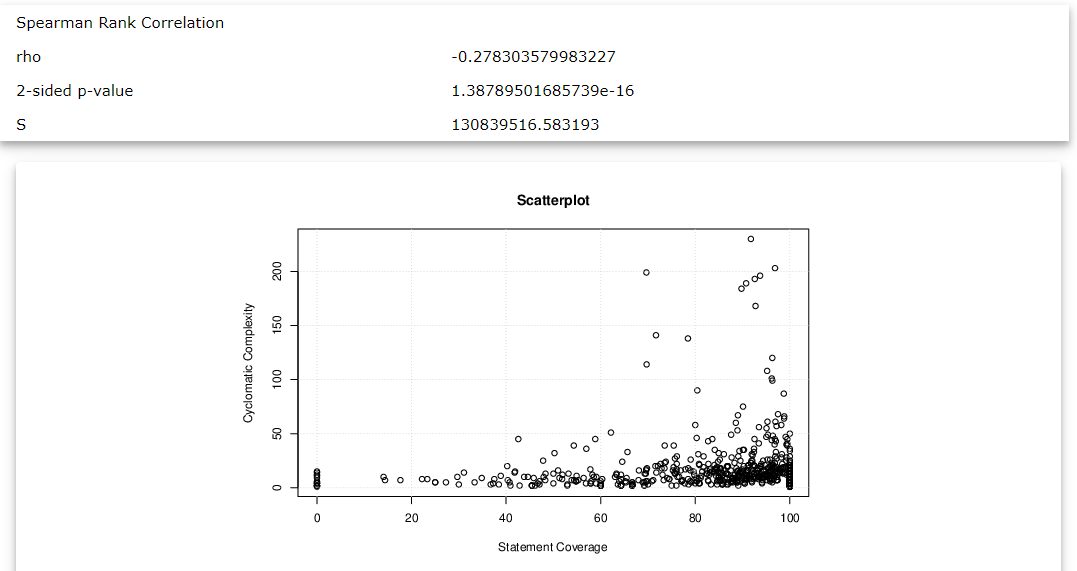


Scatterplot above is mapped between the cyclomatic complexity and statement coverage for Apache HTTPComponents-Client

**Scatterplots and spearman rank correlation for Apache commons collections**



Scatterplot above is mapped between the cyclomatic complexity and branch coverage for Apache commons collections



Scatterplot above is mapped between the cyclomatic complexity and statement coverage for Apache commons collections