1. True Positive (TP): This represents a faulty module that is correctly predicted as faulty.
2. True Negative (TN): A non-faulty module correctly predicted as non-faulty.
3. False Positive (FP): A non-faulty module incorrectly predicted as faulty.
4. False Negative (FN): A faulty module incorrectly predicted as non-faulty.

In my project, I conducted an analysis of various software fault datasets obtained from the PROMISE repository. For each of these datasets, I collected and calculated the following metrics:

1. Non-Commented Lines of Code (# non-commented-LOC): total number of lines of code in the software project, excluding lines that are comments.
2. Total Number of Modules: This metric quantifies the total number of modules or components in the software.
3. Total Number of Faulty Modules: I determined the count of modules that were identified as faulty in each dataset.
4. Percentage of Faulty Modules: To gain insight into the prevalence of faulty modules within each dataset, I calculated the percentage of faulty modules using the formula:
5. % of Faulty Modules = (Number of Faulty Modules / Total Number of Modules) \* 100%

Source Lines Of Code (SLOC): This metric family is centered around counting the lines in source code files. It includes various sub-metrics, such as:

* Physical LOC (SLOCP
* Blank LOC (BLOC
* Comment LOC (CLOC
* Logical LOC (SLOC-L)

I have made a table depicting the metrics with abbreviations and full-forms of the names as the same are there in downloaded .csv files for bug-data which I had used in this project from PROMISE repo.

|  |  |
| --- | --- |
| **Short name of Columns in Metrics** | **The name of the module.** |
| VERSION | The version of the software. |
| WMC | Weighted Methods per Class. |
| DIT | Depth of Inheritance Tree. |
| NOC | Number of Children. |
| CBO | Coupling between Objects. |
| RFC | Response for a Class. |
| LCOM | Lack of Cohesion in Methods. |
| CA | Afferent Couplings. |
| CE | Efferent Couplings. |
| NPM | Number of Public Methods. |
| LCOM3 | Lack of Cohesion in Methods 3. |
| LOC | Lines of Code. |
| DAM | Data Access Metric. |
| MOA | Measure of Aggregation. |
| MFA | Measure of Functional Abstraction. |
| CAM | Cohesion Among Methods in Class. |
| IC | Inheritance Cohesion. |
| CBM | Coupling Between Methods. |
| AMC | Average Method Complexity. |
| mAX\_CC | Maximum McCabe's Cyclomatic Complexity. |
| AVG\_CC | Average McCabe's Cyclomatic Complexity. |
| BUG | A column related to bugs. |