CHAPTER 4
DATA ANALYSIS AND INTERPRETATION

4.1 INTRODUCTION

In this chapter, an attempt has been made to identify the factors that influence the level of quality of open source software project management in the study area. For the purpose of this study, a field survey method was employed to collect first-hand information from 400 open source software volunteers. The volunteers have been chosen randomly from Chennai and Coimbatore cities. The collected data were arranged into simple tabular form. The level of quality of the open source software was selected as dependent variable. The independent variables selected for the study are gender of the volunteers, age group of the volunteers, marital status, type of family, educational status, type of working industry, position in the organization, experience in it sector, need for participating in open source software community, extent of using open source software, annual income and income from open source software development activities.

The data thus collected from the primary sources were analyzed by using simple statistical tools like percentage, average, range, standard deviation, two-way tables and chi-square. In addition to these tests, multivariate techniques like Multiple Regression, Multi-discriminant analysis, factor analysis, cluster analysis and structured equation model were used appropriately.
The analytical chapter has been presented under three sections viz., Section–I contains result and discussions based on univariate tools. Section–II highlights the analysis and interpretation based on percentage analysis. In Section–III the problems are studied with the help of Henry Garrett ranking method and the interpretations were done based on the result of the Garrett ranking.

SECTION - I : 4.2 CHI-SQUARE ANALYSIS

4.2.1 Gender and Level of Quality

Developing countries like India is giving equal opportunity for male and female. There is no discrimination in providing education and employment opportunity for female gender. Females are given more priority in IT and IT enabled courses as well as employment. In this study, an attempt was made to identify the extent of using open source software system and its quality level. For the purpose of this study, gender has been classified into two strata viz., male and female. The sample consists of 127(31.8%) respondents belonging to male category and 273 (68.3%) respondents belonging to female category. The distribution of sample respondents according to the gender and their level of quality maintained in open source software are shown in the following Table 4.1.

Table 4.1 Gender and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Gender</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Male</td>
<td>127</td>
<td>31.8</td>
<td>67.1</td>
<td>44</td>
<td>82</td>
</tr>
<tr>
<td>2.</td>
<td>Female</td>
<td>273</td>
<td>68.3</td>
<td>68.3</td>
<td>54</td>
<td>82</td>
</tr>
<tr>
<td>Total</td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It could be observed from the above table that the quality of open source software maintained among the male respondents ranged between 44 and 82 with an average of 67.1. The level of quality maintained among the female respondents ranged between 54 and 82 with an average of 68.3. From the above analysis, it was concluded that the high quality maintained in open source software was among the female respondents.

With a view to find the degree of association between gender of the respondents and the level of quality maintained in open source software, a two-way table was prepared and it is shown in the following Table 4.2 and Figure 4.1.

Table 4.2 Gender and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Gender</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low (%)</td>
<td>Medium (%)</td>
</tr>
<tr>
<td>1.</td>
<td>Male</td>
<td>35 (27.6%)</td>
<td>58 (45.7%)</td>
</tr>
<tr>
<td>2.</td>
<td>Female</td>
<td>62 (22.7%)</td>
<td>121 (44.3%)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

Figure 4.1 Gender of the respondents
It could be observed from the above table that the percentage of high level quality maintained in open source software was the highest (33.0%) among the female respondents and the same was the lowest (26.8%) among the male respondents. The percentage of medium level quality maintained in open source software was the highest (45.7%) among the male respondents and the same was the lowest (44.3%) among the female respondents. The percentage of low level quality maintained in open source software was the highest (27.6%) among the male respondents and the same was the lowest (22.7%) among the female respondents.

In order to find the relationship between the gender of the respondents and level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the test is shown in the following Table 4.3.

\[ H_0 : \text{There is no significant relationship between gender of the respondents and their level of quality maintained in open source software.} \]

\[ H_1 : \text{There is a significant relationship between gender of the respondents and their level of quality maintained in open source software.} \]

**Table 4.3 Gender and level of quality (Chi-square test)**

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated $\chi^2$ Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gender</td>
<td>1.949</td>
<td>5.991</td>
<td>2</td>
<td>Not Significant</td>
</tr>
</tbody>
</table>
It is divulged from the above table that the calculated chi-square value is lesser than the table value and the result is not significant at 5% and 1% level. Hence, the hypothesis, “Gender of the respondents and their level of quality maintained in open source software” are associated does not hold good. From the analysis, it is found that there is no relationship between gender and their level of quality maintained in open source software.

### 4.2.2 Age Group and Level of Quality

Age is an important factor to utilize the open source software at the maximum level. It is presumed that young people may utilize the open source software system very easily at the high rate of speed. For the purpose of this study, age has been studied under three classification viz., below 30 years, 30-40 years and above 40 years. The sample consists of 168 (42.0%) respondents belonging to below 30 years age group, 191 (47.8%) respondents belonging to 30-40 years age group and 41(10.3%) respondents belonging to above 40 years age group. The distribution of sample respondents according to the age of the respondents and their level of quality maintained in the open source software are shown in the following Table 4.4 and Figure 4.2.

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Age</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Below 30 years</td>
<td>168</td>
<td>42.0</td>
<td>68.3</td>
<td>52-82</td>
<td>6.6</td>
</tr>
<tr>
<td>2.</td>
<td>30-40 years</td>
<td>191</td>
<td>47.8</td>
<td>67.9</td>
<td>44-82</td>
<td>6.7</td>
</tr>
<tr>
<td>3.</td>
<td>Above 40 years</td>
<td>41</td>
<td>10.3</td>
<td>66.4</td>
<td>54-78</td>
<td>6.4</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>400</strong></td>
<td></td>
<td><strong>100.0</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It could be seen from the above table that the level of quality maintained in open source software by the respondents belonging to the age group of below 30 years ranged between 52 and 82 with an average of 68.3. The respondents with the age group between 30 and 40 years revealed their level of quality in open source software ranged between 44 and 82 with an average of 67.9. On the other hand, the level of quality maintained in open source software by the respondents belonging to above 40 years age group ranged between 54 and 78 with an average of 66.4. It was found from the analysis that the young aged respondents (below 30 years) have maintained high level of quality in open source software than the other age group category.

With a view to find the degree of association between age of the respondents and their level of quality maintained in open source software, a two-way table was prepared and it is exhibited in the following table.

**Figure 4.2 Age group of the respondents**
It could be surmised from the above table that the percentage of high level of quality maintained in open source software was the highest (32.5%) among the respondents of 30-40 years age group and the same was the lowest (22.0%) among the respondents of above 40 years age group. The percentage of medium level of quality maintained in open source software was the highest (50.6%) among the respondents of below 30 years age category and the same was the lowest (31.7%) among the respondents of above 40 years age category. The percentage of low level of quality maintained in open source software was the highest (46.3%) among the respondents of above 40 years of age category and the same was the lowest (17.9%) among the respondents of below 30 years age category.

In order to find the relationship between the age of the respondents and their level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following Table 4.5.
H₀ : There is no significant relationship between age of the respondents and their level of quality maintained in open source software.

H₁ : There is a significant relationship between age of the respondents and their level of quality maintained in open source software.

Table 4.6 Age and level of quality (Chi-square test)

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated $\chi^2$ Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>15.450</td>
<td>13.276</td>
<td>4</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

It is evident from the above table that the calculated chi-square value is greater than the table value and the result is significant at 5% level. Hence, the hypothesis “Age of the respondents and the level of quality maintained in open source software” are associated, holds good. From the analysis, it is concluded that there is a close relationship between age of the respondents and their level of quality maintained in open source software.

4.2.3 Marital Status and Level of Quality

Marriage is an auspicious function occurring in human life. Married people are highly responsible for their family and to the society. Hence, married people are given a special respect in all the ritual functions. An attempt was made to know about the level of quality maintained in open source software among marital status of the respondents. For the purpose of this study, marital status has been studied under two categories namely married and unmarried. The sample consists of 258 (64.5%) married respondents and 142 (35.5%) unmarried category. The distribution of sample respondents according to the marital status of the respondents and their level of quality maintained in open source software are shown in the following table.
Table 4.7 Marital status and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Marital status</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Married</td>
<td>258</td>
<td>64.5</td>
<td>67.8</td>
<td>44 82</td>
<td>6.7</td>
</tr>
<tr>
<td>2.</td>
<td>Unmarried</td>
<td>142</td>
<td>35.5</td>
<td>68.1</td>
<td>52 82</td>
<td>6.6</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.3 Marital status of the respondents

It could be found from the above table that the level of quality maintained in open source software by the married respondents ranged between 44 and 82 with an average of 67.8 and the level of quality maintained in open source software by the unmarried respondents ranged between 52 and 82 with an average of 68.1. From the analysis, it was concluded that unmarried respondents have maintained the high level of quality in open source software.
With a view to find the degree of association between the marital status and the level of quality in open source software, a two-way table was prepared and it is shown in the following Table 4.8.

**Table 4.8 Marital status and level of quality  (Two-way table)**

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Marital status</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Married</td>
<td>65</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td>(25.2%)</td>
<td>(44.2%)</td>
<td>(30.6%)</td>
</tr>
<tr>
<td>2.</td>
<td>Unmarried</td>
<td>32</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>(22.5%)</td>
<td>(45.8%)</td>
<td>(31.7%)</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It is highlighted from the above table that the percentage of high level of quality maintained in open source software was the highest (31.7%) among the unmarried respondents and same was the lowest (30.6%) among the married respondents. The percentage of medium level of quality maintained in open source software was the highest (45.8%) among the unmarried respondents and was the lowest (44.2%) among the married respondents. On the other hand, the percentage of the low level of quality maintained in open source software was the highest (25.2%) among the married respondents and same was the lowest (22.5%) among the respondents of unmarried category.

In order to find the relationship between the marital status of the respondents and the level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following Table 4.9.
There is no significant relationship between respondents’ marital status and their level of quality maintained in open source software.

There is a significant relationship between marital status and their level of quality maintained in open source software.

Table 4.9 Marital status and level of quality (Chi-square test)

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated $\chi^2$ Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marital status</td>
<td>0.352</td>
<td>5.991</td>
<td>2</td>
<td>Not Significant</td>
</tr>
</tbody>
</table>

It is explained from the above table that the calculated chi-square value is less than the table value and the result is not significant at 1% level and 5% level. Hence, the hypothesis “marital status of the respondents and the level of quality maintained in open source software” are associated does not hold good. From the analysis, it is concluded that there is no relationship between the marital status of the respondents and their level of quality maintained in open source software.

4.2.4 Type of Family and Level of Quality

Type of family plays a predominant role in quality maintenance of open source software system. In the joint family system, more number of family members take different responsibilities. Hence, it is quite simple and easy for them to maintain the open source software system than the nuclear family members. Type of family is an important factor to know about the use of open source system and their quality. For the purpose of this study, the type of family has been classified into two categories namely nuclear type family and joint family. The sample consists of 263 (65.8%) respondents belonging
to nuclear family and 137 (34.3%) respondents belonging to joint family. The
distribution of sample respondents according to the type of family and their
level of quality maintained in open source software are shown in the
following Table 4.10.

Table 4.10 Type of family and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Type of family</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>1.</td>
<td>Nuclear family</td>
<td>263</td>
<td>65.8</td>
<td>67.9</td>
<td>44</td>
<td>82</td>
</tr>
<tr>
<td>2.</td>
<td>Joint family</td>
<td>137</td>
<td>34.3</td>
<td>68.9</td>
<td>52</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.4 Type of family of the respondents

It was learnt from the above table that the level of quality maintained in open source software by the respondents having nuclear family
ranged between 44 and 82 with an average of 67.9 and the same with the respondents having joint family ranged between 52 and 82 with an average of 68.9. It was concluded from the analysis that the respondents belonging to joint family have maintained high quality in open source software than the nuclear family.

With a view to find the degree of association between the type of family and level of quality maintained in open source software, a two-way table was prepared and it is exhibited in the following Table 4.11.

Table 4.11 Type of family and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Type of Family</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium (45.6%)</td>
</tr>
<tr>
<td>1.</td>
<td>Nuclear family</td>
<td>65</td>
<td>120 (45.6%)</td>
</tr>
<tr>
<td>2.</td>
<td>Joint family</td>
<td>32</td>
<td>59 (43.1%)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

The above table reveals that the percentage of high level of quality maintained in open source software by the respondents was the highest (33.6%) among the respondents belonging to joint family and the same was the lowest (29.7%) among the respondent belonging to nuclear family. Similarly, the percentage of medium level of quality maintained in open source software by the respondents was the highest (45.6%) among the respondents belonging to nuclear family and the same was the lowest (43.1%) among the respondents belonging to joint family. On the other hand, the percentage of low level of quality maintained in open source software was the highest (24.7%) among the respondents belonging to nuclear family and same was the lowest (23.4%) among the respondents belonging to joint family.
In order to find the relationship between respondents’ type of family and their level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the chi-square test is shown in the following Table 4.12.

H₀ : There is no significant relationship between type of family and their level of quality maintained in open source software.

H₁ : There is a significant relationship between type of family and their level of quality maintained in open source software.

| Table 4.12 Type of family and level of quality (Chi-square test) |
|---------------|------------------|----------------|--------|----------------|
| Factor        | Calculated $\chi^2$ Value | Table Value | D.F   | Remarks        |
| Type of family| 0.647            | 5.991        | 2     | Not Significant |

It is evident from the above table that the calculated chi-square value is lesser than the table value and the result is not significant at 1% level and 5% level. Hence, the hypothesis “the respondents’ type of family and the level of quality maintained in open source software” are associated, does not hold good. From the analysis, it is concluded that there is no relationship between the type of family of the respondents and their level of quality in open source software.

4.2.5 Formal Education Level and Level of Quality

Education helps to acquire knowledge on various dimensions regarding open source software and sharpens their mind to focus on the level of quality. For the purpose of the study, the formal education has been
classified into three categories namely certificate / diploma, bachelor and master. The sample consists of 32 (8.0%) respondents were certificate / diploma holders, 137 (34.3%) respondents were graduates and 231 (57.8%) respondents were qualified at master degree level. The distribution of sample respondents according to the educational qualification of the respondents and their level of quality in open source software are shown in the following table.

Table 4.13  Formal education level and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Formal education level</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Certificate / Diploma</td>
<td>32</td>
<td>8.0</td>
<td>69.2</td>
<td>54 82</td>
<td>6.8</td>
</tr>
<tr>
<td>2.</td>
<td>Bachelor</td>
<td>137</td>
<td>34.3</td>
<td>67.3</td>
<td>54 82</td>
<td>6.5</td>
</tr>
<tr>
<td>3.</td>
<td>Master</td>
<td>231</td>
<td>57.8</td>
<td>68.1</td>
<td>44 82</td>
<td>6.7</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It could be observed from above table that the level of quality maintained in open source software by the respondents having certificate / diploma level education ranged between 54 and 82 with an average of 69.2. Similarly, the level of quality maintained in open source software by the respondents having bachelor degree ranged between 54 and 82 with an average of 67.3. On the other hand, the level of quality maintained in open source software by the respondents having master degree ranged between 44 and 82 with an average of 68.1. From the analysis, it is concluded that the respondents having certificate / diploma level education were maintained high level quality in open source software.
With a view to find the degree of association between the formal education and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.14.

Table 4.14 Formal education level and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Formal education level</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Certificate / Diploma</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(21.9)</td>
<td>(31.3)</td>
</tr>
<tr>
<td>2.</td>
<td>Bachelor</td>
<td>40</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(29.2)</td>
<td>(48.9)</td>
</tr>
<tr>
<td>3.</td>
<td>Master</td>
<td>50</td>
<td>102</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(21.6)</td>
<td>(44.2)</td>
</tr>
<tr>
<td>Total</td>
<td>97</td>
<td>179</td>
<td>124</td>
</tr>
</tbody>
</table>
It is learned from the above table that the percentage of high level of quality maintained in open source software was the highest (46.9%) among the respondents having certificate / diploma level education and the same was the lowest (21.9%) among the respondents having bachelor level education. The percentage of medium level quality maintained in open source software was the highest (48.9%) among the respondents having bachelor level education and the same was the lowest (31.3%) among the respondents of certificate / diploma level education. On the other hand, the percentage of low level quality maintained in open source software was the highest (29.2%) among the respondents of bachelor level education and the same was the lowest (21.6%) among the respondents having master level education.

In order to find the relationship between the formal education level of the respondents and their level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the chi-square test is shown in the following Table 4.15.

\[ H_0 : \text{There is no significant relationship between formal education level and their level of quality maintained in open source software.} \]

\[ H_1 : \text{There is a significant relationship between formal education level and their level of quality maintained in open source software.} \]

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated $\chi^2$ Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formal education level</td>
<td>10.979</td>
<td>9.488</td>
<td>4</td>
<td>Significant at 5% level</td>
</tr>
</tbody>
</table>
It is evident from the above table that the calculated chi-square value is greater than the table value and the result is significant at 5% level. Hence, the hypothesis “formal education level of the respondents and the level of quality maintained in open source software” are associated holds good. From the analysis, it is concluded that there is a close relationship between the formal education level of the respondents and their level of quality maintained in open source software.

### 4.2.6 Working Domain and Level of Quality

The working domain is a prominent factor to enforce their method of providing the quality of open source software. For the purpose of this study, the working segment has been classified into four categories namely IT enabled industry, software development, consultant and others (education, training, etc.). The sample consists 106 (26.5%) respondents working in IT enabled industry, 170 (42.5%) respondents working in software development, and 34 (8.5%) respondents working as consultant and 90 (22.5%) respondents working other segments such as education, training etc. The distribution of sample respondents according to the working domain of the respondents and their level of quality in open source software are shown in the following Table 4.16.

**TABLE 4.16 Working domain and level of quality**

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Working domain</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>IT enabled industry</td>
<td>106</td>
<td>26.5</td>
<td>67.8</td>
<td>44</td>
<td>6.8</td>
</tr>
<tr>
<td>2.</td>
<td>Software development</td>
<td>170</td>
<td>42.5</td>
<td>68.0</td>
<td>52</td>
<td>6.3</td>
</tr>
<tr>
<td>3.</td>
<td>Consultant</td>
<td>34</td>
<td>8.5</td>
<td>67.0</td>
<td>54</td>
<td>8.1</td>
</tr>
<tr>
<td>4.</td>
<td>Others (education, training etc.)</td>
<td>90</td>
<td>22.5</td>
<td>68.1</td>
<td>55</td>
<td>6.5</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>400</strong></td>
<td><strong>100.0</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It could be observed from the above table that the level of quality maintained in open source software among the respondents who are working in IT enabled industry ranged between 44 and 82 with an average of 67.8. The level of quality maintained in open source software by the respondents working in software development ranged between 52 and 82 with an average of 68.0. On the other hand, the level of quality maintained in open source software among the respondents who are working as consultant ranged between 54 and 82 with an average of 67.0. The level of quality maintained in open source software by the respondents working in other category such as education, training, hardware etc. ranged between 55 and 82 with an average of 68.1. From the analysis, it is concluded that the respondents who are working in other industry have maintained high level of quality in open source software.

With a view to find the degree of association between the working domain and the level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.17.
Table 4.17 Working domain and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Working Domain</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>IT enabled industry</td>
<td>27</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(25.5%)</td>
<td>(37.7%)</td>
</tr>
<tr>
<td>2.</td>
<td>Software development</td>
<td>38</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(22.4%)</td>
<td>(45.3%)</td>
</tr>
<tr>
<td>3.</td>
<td>Consultant</td>
<td>15</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(44.1%)</td>
<td>(26.5%)</td>
</tr>
<tr>
<td>4.</td>
<td>Others (education, training, hardware, etc.,)</td>
<td>17</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(18.9%)</td>
<td>(58.9%)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It could be determined from the above table that the percentage of high level of quality maintained in open source software was the highest (36.8%) among the respondents working in IT enabled industry and the same was the lowest (22.2%) among the respondents working in other industry. The percentage of medium level of quality maintained in open source software was the highest (58.9%) among the respondents working in other industry and the same was the lowest (26.5%) among the respondents working as consultant. On the other hand, the percentage of low level quality maintained in open source software was the highest (44.1%) among the respondents working as consultant and the lowest (18.9%) among the respondents working in IT enabled industry.

In order to find the relationship between the working domain of the respondents and their level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following Table 4.18.
\( H_0 \): There is no significant relationship between working industry segment and their level of quality maintained in open source software.

\( H_1 \): There is a significant relationship between working industry segment and their level of quality maintained in open source software.

**Table 4.18 Working domain and level of quality (Chi-square test)**

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated ( \chi^2 ) Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working domain</td>
<td>18.167</td>
<td>16.811</td>
<td>6</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

It is examined from the table that the calculated chi-square value is greater than the table value and the result is significant at 1% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The hypothesis “working domain of the respondents and their level of quality maintained in open source software”, are associated holds good. From the analysis, it is concluded that there is a close relationship between working domain of the respondents and their level of quality maintained in open source software.

**4.2.7 Position in the Organization and Level of Quality**

Occupation status indicates the authority and responsibility of an individual in an organization. The occupational position highlights the extensive usage of open source software and its level of maintenance. For the purpose of this study, the occupational position of the respondents has been studied under three categories namely team leader / project manager, software support engineers and soft programmers. The sample consists of 70 (17.5%)
respondents are working as team leader / project manager, 196 (49.0%) respondents are software support engineers and 134 (33.5%) respondents are software programmers. The distribution of sample respondents according to the occupational position in the organization and their level of quality maintained in open source software are shown in the following Table 4.19.

Table 4.19  Position in the organization and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Position in the organization</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Team leader / Project Manager</td>
<td>70</td>
<td>17.5</td>
<td>67.2</td>
<td>44</td>
<td>82</td>
</tr>
<tr>
<td>2.</td>
<td>Software support engineers</td>
<td>196</td>
<td>49.0</td>
<td>68.0</td>
<td>52</td>
<td>82</td>
</tr>
<tr>
<td>3.</td>
<td>Software programmers</td>
<td>134</td>
<td>33.5</td>
<td>68.1</td>
<td>54</td>
<td>82</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It could be observed from above table that the level of quality maintained in open source software among the respondents of team leader / project manager ranged between 44 and 82 with an average of 67.2. The level of quality maintained in open source software by the respondents occupying the position as software support engineers ranged between 52 and 82 with an average of 68.0. On the other hand, the level of quality maintained in open source software among the respondents possessing the status as software programmers ranged between 54 and 82 with an average of 68.1. From the analysis, it is concluded that the respondents working as software programmers have maintained high level of quality in open source software.

With a view to find the degree of association between the position in the organization and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.20.
Table 4.20  Position in the organization and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Position in the organization</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Team leader / Project Manager</td>
<td>26</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(37.1)</td>
<td>(24.3)</td>
</tr>
<tr>
<td>2.</td>
<td>Software support engineers</td>
<td>40</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(20.4)</td>
<td>(51.5)</td>
</tr>
<tr>
<td>3.</td>
<td>Software programmers</td>
<td>31</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(23.1)</td>
<td>(45.5)</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>97</strong></td>
<td><strong>179</strong></td>
</tr>
</tbody>
</table>

It is learned from the above table that the percentage of high level of quality maintained in open source software was the highest (38.6%) among the respondents working as team leader / project manager and the same was the lowest (28.1%) among the respondents working as software support...
engineers. The percentage of medium level of quality maintained in open source software was the highest (51.5%) among the respondents of software support engineers and the same was the lowest (24.3%) among the respondents working as team leader / project manager. On the other hand, the percentage of low level of quality maintained in open source software was the highest (37.1%) among the respondents working as team leader / project manager and the same was the lowest (20.4%) among the respondents working as software support engineers.

In order to find the relationship between the respondents position in the organization and their level of quality maintained in open source software, the following hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following Table 4.21.

\[ H_0 \]: There is no significant relationship between position in the organization and their level of quality maintained in open source software.

\[ H_1 \]: There is a significant relationship between position in the organization and their level of quality maintained in open source software.

<table>
<thead>
<tr>
<th>Table 4.21 Position in the organization and level of quality (Chi-square test)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Factor</strong></td>
</tr>
<tr>
<td>Position in the organization</td>
</tr>
</tbody>
</table>

It is highlighted from the table that the calculated chi-square value is greater than the table value and the result is significant at 5% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The
hypothesis “respondents position in the organization and their level of quality maintained in open source software”, are associated holds good. From the analysis, it is concluded that there is a close relationship between respondents position in the organization and their level of quality maintained in open source software.

4.2.8 Experience in IT Sector and Level of Quality

Experience gives practical knowledge to an individual, longer period gives rich experience and it moulds everyone in novel ways and makes everyone to be exposed to new ideas in an innovative way. For the purpose of this study, experience gained by the respondents in IT sector has been classified into four strata viz., below 5 years, 6-10 years, 11-15 years and above 15 years. The sample consists of 139 (34.8%) respondents who have below 5 years experience, 119 (29.8%) respondents who have between 6 and 10 years of experience, 90 (22.5%) respondents who have between 11 and 15 years of experience and 52 (13.0%) respondents who have above 15 years experience. The distribution of sample respondents according to the experience gained in IT sector and their level of quality maintained in open source software are shown in the following table.

Table 4.22 Experience in it sector and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Experience in IT sector</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min</td>
<td>Max</td>
</tr>
<tr>
<td>1.</td>
<td>Below 5 years</td>
<td>139</td>
<td>34.8</td>
<td>67.9</td>
<td>52</td>
<td>82</td>
</tr>
<tr>
<td>2.</td>
<td>6-10 years</td>
<td>119</td>
<td>29.8</td>
<td>68.8</td>
<td>54</td>
<td>82</td>
</tr>
<tr>
<td>3.</td>
<td>11-15 years</td>
<td>90</td>
<td>22.5</td>
<td>66.9</td>
<td>44</td>
<td>82</td>
</tr>
<tr>
<td>4.</td>
<td>Above 15 years</td>
<td>52</td>
<td>13.0</td>
<td>67.7</td>
<td>54</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It could be identified from the above table that the level of quality maintained in open source software by the respondents who have below 5 years experience in IT sector ranged between 52 and 82 with an average of 67.9. The level of quality maintained in open source software by the respondents having 6-10 years of experience in IT sector ranged between 54 and 82 with an average of 68.8. The level of quality maintained in open source software by the entrepreneurs having 11-15 years of experience in IT sector ranged between 44 and 82 with an average of 66.9. On the other hand, the level of quality maintained in open source software by the respondents having above 15 years of experience in IT sector ranged between 54 and 82 with an average of 67.7. From the analysis, it is concluded that the respondents who have gained 6-10 years of experience in IT sector have maintained high level of quality in open source software.

Figure 4.8  Experience in it sector
With a view to find the degree of association between the respondents’ experience in IT sector and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.23.

**Table 4.23 Experience in it sector and level of quality (Two-way table)**

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Experience in IT sector</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Below 5 years</td>
<td>23</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(16.5)</td>
<td>(54.7)</td>
</tr>
<tr>
<td>2.</td>
<td>6-10 years</td>
<td>24</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(20.2)</td>
<td>(42.0)</td>
</tr>
<tr>
<td>3.</td>
<td>11-15 years</td>
<td>38</td>
<td>26</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(42.2)</td>
<td>(28.9)</td>
</tr>
<tr>
<td>4.</td>
<td>Above 15 years</td>
<td>12</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(23.1)</td>
<td>(51.9)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It could be seen from the above table that the percentage of high level of quality maintained in open source software was the highest (37.8%) among the respondents who have 6-10 years experience in IT sector and the same was the lowest (25.0%) among the respondents who have above 15 years of experience in IT sector. The percentage of medium level of quality maintained in open source software was the highest (54.7%) among the respondents who have below 5 years of experience in IT sector and the same was the lowest (28.9%) among the respondents who have 11-15 years of experience in IT sector. On the other hand, the percentage of low level of quality maintained in open source software was the highest (42.2%) among the respondents who have 11-15 years of experience in IT sector and same was the lowest (16.5%) among the respondents who have below 5 years of experience in IT sector.
In order to find the relationship between the experience gained by the respondents in IT sector and the level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the test is shown in the following Table 4.24.

\[ H_0 \]: There is no significant relationship between experience of the respondents in IT sector and their level of quality maintained in open source software.

\[ H_1 \]: There is a significant relationship between experience of the respondents in IT sector and their level of quality maintained in open source software.

**Table 4.24 Experience in IT sector and level of quality (Chi-square test)**

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated $\chi^2$ Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experience in IT sector</td>
<td>27.891</td>
<td>16.811</td>
<td>6</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

It could be noted from the above table that the calculated chi-square value is greater than the table value and the result is significant at 1% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The hypothesis “experience in IT sector and the level of quality maintained in open source software by the respondents” are associated, holds good. From the analysis, it is concluded that there is a close relationship between experience in IT sector and their level of quality in open source software.
4.2.9 Need for Participating in Open Source Software Community and Level of Quality

Need is the most powerful factor in human life. Abraham Maslow has rightly pointed out the power of the need, based on hierarchical model. If the needs are fulfilled the tension will be released, otherwise the tension remains constant and the individual is unable to concentrate in his/her job perfectly. For the purpose of this study, need for participating in open source software community has been classified into four strata viz., personal needs, company needs, personal & company needs and community needs. The sample consists of 84 (21.0%) respondents are having personal needs, 58 (14.5%) respondents are having company needs, 152 (38.0%) respondents are having personal & company needs and 106 (26.5%) respondents are having community needs. The distribution of sample respondents according to the need for participating in open source community and their level of quality in open source software are shown in the following Table 4.25.

Table 4.25 Need for participating in open source software community and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Needs</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Personal needs</td>
<td>84</td>
<td>21.0</td>
<td>67.6</td>
<td>54 82</td>
<td>6.8</td>
</tr>
<tr>
<td>2.</td>
<td>Company needs</td>
<td>58</td>
<td>14.5</td>
<td>68.7</td>
<td>52 82</td>
<td>6.5</td>
</tr>
<tr>
<td>3.</td>
<td>Personal &amp; company needs</td>
<td>152</td>
<td>38.0</td>
<td>68.0</td>
<td>54 82</td>
<td>6.5</td>
</tr>
<tr>
<td>4.</td>
<td>Community needs</td>
<td>106</td>
<td>26.5</td>
<td>67.6</td>
<td>44 82</td>
<td>6.9</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It could be identified from the above table that the level of quality maintained in open source software by the respondents who participated in
open source software community for their personal needs ranged between 54 and 82 with an average of 67.6. The level of quality maintained in open source software by the respondents who participated in open source software community for their company needs ranged between 52 and 82 with an average of 68.7. The level of quality maintained in open source software by the respondents who participated in open source software community for their personal and company needs ranged between 54 and 82 with an average of 68.0. On the other hand, the level of quality maintained in open source software by the respondents who participated in open source software community for their community needs ranged between 44 and 84 with an average of 67.6. From the analysis, it is concluded that the respondents who participated in open source software community for their company needs have maintained high level of quality in open source software.

![Figure 4.9 Need for participating in open source software community](image)

**Figure 4.9 Need for participating in open source software community**

With a view to find the degree of association between the need for participating in open source software community and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.26.
Table 4.26 Need for participating in open source software community and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Needs</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Personal needs</td>
<td>30 (35.7)</td>
<td>41 (48.8)</td>
</tr>
<tr>
<td>2.</td>
<td>Company needs</td>
<td>12 (20.7)</td>
<td>27 (46.6)</td>
</tr>
<tr>
<td>3.</td>
<td>Personal &amp; company needs</td>
<td>27 (17.8)</td>
<td>65 (42.8)</td>
</tr>
<tr>
<td>4.</td>
<td>Community needs</td>
<td>28 (26.4)</td>
<td>46 (43.4)</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It could be observed from the above table that the percentage of high level of quality maintained in open source software was the highest (39.5%) among the respondents who participated in open source software community for their personal & company needs and the same was the lowest (15.3%) among the respondents who participated in open source software community for their personal needs. The percentage of medium level of quality maintained in open source software was the highest (48.8%) among the respondents who participated in open source software community for their personal needs and the same was the lowest (42.8%) among the respondents who participated in open source software community for their personal and company needs. On the other hand, the percentage of low level of quality maintained in open source software was the highest (35.7%) among the respondents who participated in open source software community for their personal needs and same was the lowest (17.8%) among the respondents who participated in open source software community for their personal & company needs.
In order to find the relationship between the need for participating in open source software community and the level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the test is shown in the following Table 4.27.

\( H_0 \): There is no significant relationship between need for participating in open source software Community and their level of quality maintained in open source software.

\( H_1 \): There is a significant relationship between need for participating in open source software community and their level of quality maintained in open source software.

Table 4.27 Need for participating in open source software community and level of quality (Chi-square test)

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated ( \chi^2 ) Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Needs</td>
<td>18.358</td>
<td>16.811</td>
<td>6</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

It is divulged from the above table that the calculated chi-square value is greater than the table value and the result is significant at 1% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The hypothesis “need for participating in open source software community and the level of quality maintained in open source software by the respondents” are associated, holds good. From the analysis, it is concluded that there is a close relationship between need for participating in open source software community and their level of quality maintained in open source software.
4.2.10 Extent of Using Open Source Software and Level of Quality

An attempt was made to know the extent of using open source software by the selected sample respondents. For the purpose of this study, the extent of using open source software has been classified into five strata viz., office automation tool, communication tool, testing tool, development tool and version control tool. The sample consists of 40 (10.0%) respondents who are using open source software for the purpose of office automation tool, 42 (10.5%) respondents who are using open source software communication tool, 32 (8.0%) respondents who are using open source software for testing tool, 214 (53.5%) respondents who are using for development tool and 72 (18.0%) respondents who are using version control tool. The distributions of sample respondents according to the extent of using open source software and their level of quality maintained in open source software are shown in the following Table 4.28.

Table 4.28 Extent of using open source software and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Extent of using</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Ave (Min, Max)</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Office automation tool</td>
<td>40</td>
<td>10.0</td>
<td>67.6 (54, 82)</td>
<td>6.5</td>
</tr>
<tr>
<td>2.</td>
<td>Communication tool</td>
<td>42</td>
<td>10.5</td>
<td>68.7 (44, 82)</td>
<td>8.3</td>
</tr>
<tr>
<td>3.</td>
<td>Testing tool</td>
<td>32</td>
<td>8.0</td>
<td>67.9 (55, 78)</td>
<td>5.5</td>
</tr>
<tr>
<td>4.</td>
<td>Development tool</td>
<td>214</td>
<td>53.5</td>
<td>68.4 (54, 82)</td>
<td>6.6</td>
</tr>
<tr>
<td>5.</td>
<td>Version control tool</td>
<td>72</td>
<td>18.0</td>
<td>66.2 (54, 82)</td>
<td>6.1</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It could be identified from the above table that the respondents who utilize the office automation tool for the development of open source software ranged between 54 and 82 with an average of 67.6. The respondents who are
using the communication tool for the development of open source software ranged between 44 and 82 with an average of 68.7 and the respondents who are utilizing the testing tool for the development of open source software ranged between 55 and 78 with an average of 67.9. The respondents who are using the development tool for the development of open source software ranged between 54 and 82 with an average of 68.4. On the other hand, the respondents who are utilizing the version control tool for the development of open source software ranged between 54 and 82 with an average of 66.2. From the analysis, it is concluded that the respondents using open source software for the purpose of communication tool have maintained the open source software at high level.

With a view to find the degree of association between the extent of using open source software and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.29.

![Figure 4.10 Extent of using open source software](image-url)
Table 4.29  Extent of using open source software and level of quality
(Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Extent of using</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(       )</td>
<td>(      )</td>
</tr>
<tr>
<td>1.</td>
<td>Office automation tool</td>
<td>11 (27.5%)</td>
<td>17 (42.5%)</td>
</tr>
<tr>
<td>2.</td>
<td>Communication tool</td>
<td>9 (21.4%)</td>
<td>20 (47.6%)</td>
</tr>
<tr>
<td>3.</td>
<td>Testing tool</td>
<td>6 (18.8%)</td>
<td>15 (46.9%)</td>
</tr>
<tr>
<td>4.</td>
<td>Development tool</td>
<td>47 (22.0%)</td>
<td>97 (45.3%)</td>
</tr>
<tr>
<td>5.</td>
<td>Version control tool</td>
<td>24 (33.3%)</td>
<td>30 (41.7%)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It could be identified from the above table that the percentage of high level of quality maintained in open source software was the highest (34.4%) among the respondents who are utilizing the testing tool for the development of open source software and the same was the lowest (25.0%) among the respondents who are using the version control tool for the development of open source software. The percentage of medium level of quality maintained in open source software was the highest (47.6%) among the respondents who are utilizing the communication tool for the development of open source software and the same was the lowest (41.7%) among the respondents who are using the version control tool for the development of open source software. On the other hand, the percentage of low level of quality maintained in open source software was the highest (33.3%) among the respondents who are utilizing the version control tool for the development of open source software and same was the lowest (18.8%) among the respondents who are utilizing the testing tool for the development of open source software.
In order to find the relationship between the extent of using open source software and the level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following table.

H₀ : There is no significant relationship between extent of using open source software and their level of quality maintained in open source software.

H₁ : There is a significant relationship between extent of using open source software and their level of quality maintained in open source software.

| Table 4.30 Extent of using open source software and level of quality (Chi-square test) |
| Factor | Calculated $\chi^2$ Value | Table Value | D.F | Remarks |
| Extent of using | 1.656 | 15.507 | 8 | Not Significant |

It could be seen from the above table that the calculated chi-square value is lesser than the table value and the result is not significant. Hence, the null hypothesis is accepted and the alternative hypothesis is rejected. The hypothesis “Extent of using open source software and the level of quality maintained in open source software by the respondents” are associated, does not hold good. From the analysis, it is concluded that there is no relationship between extent of using open source software and their level of quality in open source software.
4.2.11 Annual Income and Level of Quality

Income is one of the important factors in human life. The needs and wants of the human beings can be very easily fulfilled only through the income. The income also determines the standard of living of the individual and holds the vital role to recognize and respect the person in the society. For the purpose of this study, annual income generated by the respondents has been classified into four strata viz., less than 5 lakhs, 5-7 lakhs, 7-10 lakhs and above 10 lakhs. The sample consists of 132 (33.0%) respondents who are earning less than Rs.5 lakhs per year, 112 (28.0%) respondents who are earning Rs.5-7 lakhs per year, 100 (25.0%) respondents who are earning Rs.7-10 lakhs per year and 56 (14.0%) respondents who are earning above Rs.10 lakhs per year. The distributions of sample respondents according to the annual income and their level of quality maintained in open source software are shown in the following table.

Table 4.31 Annual income and level of quality

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Annual salary income</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Less than 5 lakhs</td>
<td>132</td>
<td>33.0</td>
<td>68.2</td>
<td>54-82</td>
<td>6.5</td>
</tr>
<tr>
<td>2.</td>
<td>5-7 lakhs</td>
<td>112</td>
<td>28.0</td>
<td>67.2</td>
<td>44-82</td>
<td>7.1</td>
</tr>
<tr>
<td>3.</td>
<td>7-10 lakhs</td>
<td>100</td>
<td>25.0</td>
<td>68.1</td>
<td>52-82</td>
<td>6.6</td>
</tr>
<tr>
<td>4.</td>
<td>Above 10 lakhs</td>
<td>56</td>
<td>14.0</td>
<td>68.3</td>
<td>54-82</td>
<td>6.1</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>400</td>
<td>100.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
It could be identified from the above table that the level of quality maintained in open source software by the respondents who are earning less than Rs.5 lakhs ranged between 54 and 82 with an average of 68.2. The level of quality maintained in open source software by the respondents who are earning 5-7 lakhs ranged between 44 and 82 with an average of 67.2. The level of quality maintained in open source software by the respondents who are earning Rs.7-10 lakhs ranged between 52 and 82 with an average of 68.1. On the other hand, the level of quality maintained in open source software by the respondents who are earning above Rs.10 lakhs ranged between 54 and 82 with an average of 68.3. From the analysis, it is concluded that the respondents who earning above Rs.10 lakhs income per year maintaining high level of quality in open source software.

With a view to find the degree of association between the respondents annual income and level of quality maintained in open source software, a two-way table was prepared and it is displayed in the following Table 4.32.

**Figure 4.11 Annual income of the respondents**
Table 4.32 Annual salary income and level of quality (Two-way table)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Annual salary income</th>
<th>Level of quality</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td>1.</td>
<td>Less than 5 lakhs</td>
<td>21 (15.9)</td>
<td>57 (43.2)</td>
</tr>
<tr>
<td>2.</td>
<td>5-7 lakhs</td>
<td>41 (36.6)</td>
<td>54 (48.2)</td>
</tr>
<tr>
<td>3.</td>
<td>7-10 lakhs</td>
<td>24 (24.0)</td>
<td>42 (42.0)</td>
</tr>
<tr>
<td>4.</td>
<td>Above 10 lakhs</td>
<td>11 (19.6)</td>
<td>26 (46.4)</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>97</td>
<td>179</td>
</tr>
</tbody>
</table>

It could be observed from the above table that the percentage of high level of quality maintained in open source software was the highest (40.9%) among the respondents who earn less than Rs.5 lakhs per annum and the same was the lowest (15.2%) among the respondents who earn Rs.5-7 lakhs per annum. The percentage of medium level of quality maintained in open source software was the highest (48.2%) among the respondents who earn Rs.5-7 lakhs per annum and the same was the lowest (42.0%) among the respondents who earn Rs.7-10 lakhs per annum. On the other hand, the percentage of low level of quality maintained in open source software was the highest (36.6%) among the respondents who earn Rs.5-7 lakhs per annum and same was the lowest (15.9%) among the respondents who earn less than Rs.5 lakhs income per annum.

In order to find the relationship between the respondents’ annual income and the level of quality maintained in open source software, the following null hypothesis was framed and tested with the help of Chi-square test. The result of the chi-square test is shown in the following Table 4.33.
H₀ : There is no significant relationship between the respondents annual income and their level of quality maintained in open source software.

H₁ : There is a significant relationship between the respondents annual income and their level of quality maintained in open source software.

### Table 4.33 Annual income and level of quality (Chi-square test)

<table>
<thead>
<tr>
<th>Factor</th>
<th>Calculated χ² Value</th>
<th>Table Value</th>
<th>D.F</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annual salary income</td>
<td>25.579</td>
<td>13.276</td>
<td>4</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

It could be observed from the above table that the calculated chi-square value is greater than the table value and the result is significant at 1% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The hypothesis “annual salary income and the level of quality maintained in open source software by the respondents” are associated, holds good. From the analysis, it is concluded that there is a close relationship between the respondents annual income and their level of quality maintained in open source software.

#### 4.2.12 Additional Income Earned from Open Source Software Development Activities and Level of Quality

The respondents working in software industries are busily engaged apart from their regular working hours. They get the job assignment through voluntary participation in virtual network. For each job, they are collecting the service charges separately. In this chapter, an attempt was made to know
the additional income earned by the respondents through open source software development activities. For the purpose of this study, it has been classified into four strata viz., less than Rs.3 lakhs, Rs.3-5 lakhs, Rs.5-7 lakhs and above Rs.7 lakhs. The sample consists of 36 (35.0%) respondents who are earning less than Rs.3 lakhs additional income from open source software development activities, 27 (26.2%) respondents who are earning Rs.3-5 lakhs additional income from open source software development activities, 22 (21.4%) respondents who are earning Rs.5-7 lakhs additional income from open source software development activities and 18 (17.5%) respondents who are earning above Rs.7 lakhs additional income from open source software development activities. The distributions of sample respondents according to the additional income earned from open source software development activities and their level of quality maintained in open source software are shown in the following Table 4.34.

**Table 4.34 Additional income earned from open source software development activities and level of quality**

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Additional Income Earned</th>
<th>No. of Respondents</th>
<th>%</th>
<th>Average</th>
<th>Range Min</th>
<th>Range Max</th>
<th>S.D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Less than 3 lakhs</td>
<td>36</td>
<td>35.0</td>
<td>67.9</td>
<td>54</td>
<td>82</td>
<td>5.9</td>
</tr>
<tr>
<td>2.</td>
<td>3-5 lakhs</td>
<td>27</td>
<td>26.2</td>
<td>67.4</td>
<td>44</td>
<td>82</td>
<td>8.0</td>
</tr>
<tr>
<td>3.</td>
<td>5-7 lakhs</td>
<td>22</td>
<td>21.4</td>
<td>67.8</td>
<td>55</td>
<td>74</td>
<td>5.4</td>
</tr>
<tr>
<td>4.</td>
<td>Above 7 lakhs</td>
<td>18</td>
<td>17.5</td>
<td>66.2</td>
<td>54</td>
<td>78</td>
<td>8.3</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>103</strong></td>
<td><strong>100.0</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

It could be seen from the above table that the level of quality maintained in open source software by the respondents who have earned less than 3 lakhs additional income from open source software development activities ranged between 54 and 82 with an average of 67.9 and the level of quality maintained in open source software by the respondents who are
earning Rs.3-5 lakhs additional income from open source software development activities ranged between 44 and 82 with an average of 67.4.

![Figure 4.12 Additional income earned from open source software development activities](image)

The level of quality maintained in open source software by the respondents who are earning Rs.5-7 lakhs additional income from open source software development activities ranged between 55 and 74 with an average of 67.8. On the other hand, the level of quality maintained in open source software by the respondents who are earning above Rs.7 lakhs additional income from open source software development activities ranged between 54 and 78 with an average of 66.2. From the analysis it is concluded that the respondents who are earning below 3 lakhs income from open source software development activities have maintained high level of quality in open source software.

With a view to find the degree of association between the additional income earned by the respondents from open source software development activities and level of quality maintained, a two-way table was prepared and it is displayed in the following Table 4.35.
It could be observed from the above table that the percentage of high level of quality maintained in open source software was the highest (55.6%) among the respondents who earned Rs.3-5 lakhs additional income from open source software development activities and the same was the lowest (16.7%) among the respondents who earned less than Rs.3 lakhs additional income from open source software development activities. The percentage of medium level of quality maintained in open source software was the highest (61.1%) among the respondents who earned less than Rs.3 lakhs income from open source software development activities and the same was the lowest (22.2%) among the respondents who earned 3-5 lakhs income from open source software development activities. On the other hand, the percentage of low level of quality maintained in open source software was the highest (44.4%) among the respondents who earned above Rs.7 lakhs additional income from open source software development activities and same was the lowest (22.2%) among the respondents who earned less than 3 lakhs and also Rs.3-5 lakhs additional income from open source software development activities.
In order to find the relationship between additional income earned from open source software development activities and the level of quality maintained, the following null hypothesis was framed and tested with the help of Chi-square test, and the result of the test is shown in the following Table 4.36.

\( H_0 \): There is no significant relationship between additional income earned from open source software development activities and their level of quality.

\( H_1 \): There is a significant relationship between additional income earned from open source software development activities and their level of quality.

<table>
<thead>
<tr>
<th>Table 4.36 Range of income from open source software development activities and level of quality (Chi-square test)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factor</td>
</tr>
<tr>
<td>Range of income</td>
</tr>
</tbody>
</table>

It could be noted from the above table that the calculated chi-square value is greater than the table value and the result is significant at 1% level. Hence, the null hypothesis is rejected and the alternative hypothesis is accepted. The hypothesis “additional income earned from open source software development activities and the level of quality maintained in open source software by the respondents” are associated, holds good. From the analysis, it is concluded that there is a close relationship between additional income earned from open source software development activities and their level of quality maintained in open source software.
SECTION - II : 4.3 PERCENTAGE ANALYSIS

4.3.1 Location of the Respondents

An attempt was made to know about the locality of the respondents. For this purpose, the location of the respondents was studied under two strata viz., Chennai and Coimbatore. The details are furnished in the following Table 4.37.

Table 4.37 Location of the respondents

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Location</th>
<th>No. of Respondents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Chennai</td>
<td>200</td>
<td>50.0</td>
</tr>
<tr>
<td>2.</td>
<td>Coimbatore</td>
<td>200</td>
<td>50.0</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
</tr>
</tbody>
</table>

It could be observed from the above table that 50.0 per cent of the respondents belonged to Chennai and remaining 50 per cent of the respondents belonged to Coimbatore. From the above analysis, it is found that (50%) of the respondents belonged to Chennai and also Coimbatore.

4.3.2 Usage of Open Source Software in the Work Environment

In the work environment, the respondents were asked the usage of open source software or not. For this purpose, it was studied under three strata viz., yes, uncertain and no. The details are furnished in the following Table 4.38.
Table 4.38 Usage of open source software in the work environment

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Opinion</th>
<th>No. of Respondents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Yes</td>
<td>296</td>
<td>74.0</td>
</tr>
<tr>
<td>2.</td>
<td>Uncertain</td>
<td>64</td>
<td>16.0</td>
</tr>
<tr>
<td>3.</td>
<td>No</td>
<td>40</td>
<td>10.0</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
</tr>
</tbody>
</table>

It is evident from the above table that 74.0 per cent of the respondents were using the open source software in the work environment, 16.0 per cent of the respondents have opined as uncertain about the usage of open source software in the work environment and 10.0 per cent of the respondents were not using the open source software in the work environment. From the above analysis, it is noted that majority (74.0%) of the respondents were using the open source software in the work environment.

Figure 4.13 Location of the respondents
4.3.3 Level of Involvement in Open Source Software Development

An attempt was made to identify the level of involvement in open source software development. For this purpose, the level of involvement was studied under four strata viz., fully, partly, seldom and passive. The details are furnished in the following Table 4.39.

Table 4.39 Level of involvement in open source software development

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Level of involvement</th>
<th>No. of Respondents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Fully</td>
<td>99</td>
<td>24.8</td>
</tr>
<tr>
<td>2.</td>
<td>Partly</td>
<td>160</td>
<td>40.0</td>
</tr>
<tr>
<td>3.</td>
<td>Seldom</td>
<td>53</td>
<td>13.3</td>
</tr>
<tr>
<td>4.</td>
<td>Passive</td>
<td>88</td>
<td>22.0</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>400</strong></td>
<td><strong>100.0</strong></td>
</tr>
</tbody>
</table>
It could be inferred from the above table that 24.8 per cent of the respondents were involved fully for the development of open source software, 40.0 per cent of the respondents were partly involved for the development of open source software, 13.3 per cent of the respondents were involved rarely for the development of open source software and 22.0 per cent of the respondents were shown passive involvement for the development of open source software. From the above analysis, it is noted that majority (40.0%) of the respondents were involved in part-time for the development of open source software.

4.3.4 Experience in Open Source Software Activity

Experience makes everyone to learn a lot from the open source software activity. For this purpose, the experience gained in open source software activity was studied under five strata viz., below 2 years, 2-5 years, 5-10 years, 10-15 years and above 15 years. The details are furnished in the following Table 4.40.
Table 4.40 Experience gained in open source software activity

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Experience</th>
<th>No. of Respondents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Below 2 years</td>
<td>101</td>
<td>25.3</td>
</tr>
<tr>
<td>2.</td>
<td>2-5 years</td>
<td>208</td>
<td>52.0</td>
</tr>
<tr>
<td>3.</td>
<td>5-10 years</td>
<td>47</td>
<td>11.8</td>
</tr>
<tr>
<td>4.</td>
<td>10-15 years</td>
<td>34</td>
<td>8.5</td>
</tr>
<tr>
<td>5.</td>
<td>Above 15 years</td>
<td>10</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>400</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Figure 4.16 Experience gained in open source software Activity

It could be observed from the above table that 25.3 per cent of the respondents have gained below 2 years experience in open source software activity, 52.0 per cent of the respondents have gained 2-5 years experience in open source software activity, 11.8 per cent of the respondents have gained 5-10 years experience in open source software activity, 8.5 per cent of the respondents have gained 10-15 years experience in open source software activity, and 2.5 per cent of the respondents have gained above 15 years experience in open source software activity. From the above analysis, it is
noted that majority (52.0%) of the respondents have gained 2-5 years experience in open source software activity.

4.4 MULTIPLE REGRESSION ANALYSIS

The regression is a statistical relationship between two or more variables. When there are two or more independent variables, the analysis that describes such relationship is the multiple regression. This analysis is adopted where there is one dependent variable that is presumed to be a function of two or more independent variables. In multiple regression, a linear composite of explanatory variables is formed, in such a way that it has maximum correlation with an active criterion variable. The main objective of using this technique is to predict the variability of the dependent variable, based on its co-variance with all the independent variables. It is useful to predict the level of dependent phenomenon through Multiple Regression Analysis models, if the levels of independent variables were given. The linear multiple regression problem is to estimate coefficients of \( \beta_1, \beta_2, \ldots, \beta_j \) and \( \beta_0 \) such that the expression,

\[
Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_j X_j
\]

provides a good estimate of an individual \( Y \) score based on the \( X \) scores.

In the following analysis, the relationship between the level of quality maintained in open source software by the respondents and twelve independent factors were studied. It was found that out of twelve factors, only ten factors were closely associated with the quality of open source software by the selected sample respondents.

Selected twelve independent factors are.

1. Gender
2. Age Group
3. Marital Status  
4. Type of Family  
5. Educational Status  
6. Working Industry  
7. Working Status  
8. Experience  
9. Need for Participate  
10. Extent of using Open Source Software  
11. Annual Salary  
12. Income from OSS Development Activities  

Table 4.41 Multiple regression analysis

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Variables</th>
<th>Unstandardized coefficients</th>
<th>Standardized coefficients</th>
<th>T</th>
<th>Sig.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Constant)</td>
<td>3.395</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Gender</td>
<td>0.167</td>
<td>0.098</td>
<td>2.799</td>
<td>1%</td>
</tr>
<tr>
<td>2</td>
<td>Age Group</td>
<td>0.475</td>
<td>0.314</td>
<td>9.545</td>
<td>1%</td>
</tr>
<tr>
<td>3</td>
<td>Marital Status</td>
<td>-0.027</td>
<td>-0.036</td>
<td>-0.865</td>
<td>NS</td>
</tr>
<tr>
<td>4</td>
<td>Type of Family</td>
<td>-0.015</td>
<td>-0.023</td>
<td>-0.504</td>
<td>NS</td>
</tr>
<tr>
<td>5</td>
<td>Educational Status</td>
<td>0.206</td>
<td>0.193</td>
<td>5.323</td>
<td>1%</td>
</tr>
<tr>
<td>6</td>
<td>Working Domain</td>
<td>-0.047</td>
<td>-0.080</td>
<td>-2.402</td>
<td>5%</td>
</tr>
<tr>
<td>7</td>
<td>Working Status</td>
<td>0.186</td>
<td>0.271</td>
<td>5.840</td>
<td>1%</td>
</tr>
<tr>
<td>8</td>
<td>Experience</td>
<td>0.082</td>
<td>0.139</td>
<td>3.022</td>
<td>1%</td>
</tr>
<tr>
<td>9</td>
<td>Need to Participate</td>
<td>0.206</td>
<td>0.229</td>
<td>6.317</td>
<td>1%</td>
</tr>
<tr>
<td>10</td>
<td>Extent of using Open Source Software</td>
<td>0.188</td>
<td>0.187</td>
<td>4.335</td>
<td>1%</td>
</tr>
<tr>
<td>11</td>
<td>Annual Salary</td>
<td>-0.062</td>
<td>-0.036</td>
<td>-0.834</td>
<td>NS</td>
</tr>
<tr>
<td>12</td>
<td>Income from OSS Development activities</td>
<td>0.672</td>
<td>0.393</td>
<td>8.458</td>
<td>1%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R-Value</th>
<th>R² -Value</th>
<th>Degree of freedom – V₁</th>
<th>Degree of freedom – V₂</th>
<th>F Value</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.900</td>
<td>0.810</td>
<td>12</td>
<td>386</td>
<td>126.26</td>
<td>1% Level</td>
</tr>
</tbody>
</table>
The multiple linear regression co-efficient (dependent variable) is found to be statistically a good fit as $R^2$ is 0.810. It shows that the independent variables contribute about 90.0 per cent of the variation in the level of quality maintained by the selected sample respondents and this is statistically significant at 1% level and 5% level respectively.

The table indicated that the co-efficient of gender, age group, educational status, working status, experience, need for participate, extent of using open source software and income from OSS development activities are positively associated with the quality of open source software project management. On the other hand, the co-efficient of working domain is negatively associated. Further, it indicated that the contribution of working location, gender, age group, educational status, working status, experience, need to participate, extent of using open source software and income from OSS development activities are statistically significant implying that their influence on quality of open source software project management is stronger than the other variables.

Thus from the above analysis, the following observation could be made. The level of quality of open source software project management is positively associated with their gender, age group, educational status, working status, experience, need to participate, extent of using open source software and income from OSS development activities in the study area.

4.5 DISCRIMINANT FUNCTION ANALYSIS

Discriminant analysis is a technique designed to characterize the relationship between a set of variables, often called the response are predictor variables, and a grouping variables with a relatively small number of
categories. To do so, discriminant function creates a linear combination of the predictors that best characterizes the differences among the groups. The technique is related to both regression and multi-variate analysis of variants, and as such it is another general linear model technique. Discriminant attempts to find the linear combinations of predictor variables i.e., X are low level quality maintainers of open source software, Y are the high level quality maintainers of open source software. How do respondents maintain the quality against the various variables of utilizing the Open Source Software? Do all the factors selected in the analysis differ among the groups? In general, what are all the variables which significantly discriminate the respondents of one group from other group? Discriminant function analysis answers these questions in three stages viz., 1. Construction of discriminant function 2. Classification 3. Interpretation.

Opinions of respondents towards extent of utilizing the open source software were collected. In the study area 400 respondents were divided into two groups, one showing low level of utilization of open source software and the other showing high level of utilization of open source software. The difference of opinion of the respondents in one group with the other was studied with the help of discriminant function analysis. For the purpose of this study twelve variables were selected.

1. Gender
2. Age Group
3. Marital Status
4. Type of Family
5. Educational Status
The discriminant function analysis was attempted to construct a function with these and other variables, so that the respondents belonging to these two groups are differentiated at the maximum. The linear combination of variables is known as discriminant function and its parameters are called discriminant function coefficients. In constructing this discriminant function, all the variables which contribute more to differentiate these two groups were examined.

Mahalanobis Minimum $D^2$ Method is based on the generalized squared Euclidean distance that adjusts for unequal variances in the variables. The major advantage of this procedure is that it is computed in the original space of the predictor (independent) variables rather than as a collapsed version which is used in the other method.

Generally, all the variables selected will not contribute to explain the maximum discriminatory power of the function. So, a selection rule is applied based on certain criteria to include those variables which best discriminate. Stepwise selection method was applied in constructing discriminant function which selects one variable at a time to include in the
Before entering into the function, the variables are examined for inclusion in the function.

The variables which have maximum $D^2$ value, if entered into the function is selected for inclusion in the function. Once entered, any variable already in the equation is again considered for removal based on certain removal criteria. Likewise, at each step, the next best discriminating variable is selected and included in the function and any variable already included in the function is considered for removal, based on the selection and removal criteria respectively.

### 4.5.1 Discriminant Analysis for the Problem Under Study

Since discriminant function analysis involved classification problem also, to ascertain the efficiency of the discriminant function analysis all the variables which satisfy the entry and removal criteria were entered into the function. Normally, the criteria used to select the variables for inclusion in the function is minimum 'F', to enter into the equation (i.e) F statistic calculated for the qualified variable to enter into the function is fixed as $\geq 1$.

Similarly, any variable entered in the equation will be removed from the function if ‘F’ statistic for the variable calculated is $<1$. The two groups are defined as

- **Group 1** - Low level of quality maintainer of open source software
- **Group 2** - High level of quality maintainer of open source software

The mean and standard deviation for these groups and for the entire samples are given for each variable considered in the analysis.
Table 4.42 Group means (Between low and high utilizer groups)

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Factor</th>
<th>LOW</th>
<th></th>
<th>HIGH</th>
<th></th>
<th>TOTAL</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>1</td>
<td>Gender</td>
<td>1.753</td>
<td>0.433</td>
<td>1.640</td>
<td>0.481</td>
<td>1.683</td>
<td>0.466</td>
</tr>
<tr>
<td>2</td>
<td>Age Group</td>
<td>1.640</td>
<td>0.648</td>
<td>1.708</td>
<td>0.651</td>
<td>1.683</td>
<td>0.650</td>
</tr>
<tr>
<td>3</td>
<td>Marital Status</td>
<td>1.393</td>
<td>0.490</td>
<td>1.332</td>
<td>0.472</td>
<td>1.355</td>
<td>0.479</td>
</tr>
<tr>
<td>4</td>
<td>Type of Family</td>
<td>1.333</td>
<td>0.473</td>
<td>1.348</td>
<td>0.477</td>
<td>1.343</td>
<td>0.475</td>
</tr>
<tr>
<td>5</td>
<td>Educational Status</td>
<td>2.373</td>
<td>0.691</td>
<td>2.572</td>
<td>0.599</td>
<td>2.498</td>
<td>0.641</td>
</tr>
<tr>
<td>6</td>
<td>Working Domain</td>
<td>2.213</td>
<td>1.053</td>
<td>2.304</td>
<td>1.107</td>
<td>2.270</td>
<td>1.086</td>
</tr>
<tr>
<td>7</td>
<td>Working Status</td>
<td>2.013</td>
<td>0.723</td>
<td>2.248</td>
<td>0.667</td>
<td>2.160</td>
<td>0.697</td>
</tr>
<tr>
<td>8</td>
<td>Experience</td>
<td>2.187</td>
<td>1.058</td>
<td>2.108</td>
<td>1.026</td>
<td>2.138</td>
<td>1.037</td>
</tr>
<tr>
<td>9</td>
<td>Need to Participate</td>
<td>2.707</td>
<td>1.078</td>
<td>2.696</td>
<td>1.081</td>
<td>2.700</td>
<td>1.078</td>
</tr>
<tr>
<td>10</td>
<td>Extent of using Open Source</td>
<td>3.593</td>
<td>1.193</td>
<td>3.588</td>
<td>1.190</td>
<td>3.590</td>
<td>1.190</td>
</tr>
<tr>
<td></td>
<td>Software</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Annual Salary</td>
<td>2.287</td>
<td>1.025</td>
<td>2.148</td>
<td>1.063</td>
<td>2.200</td>
<td>1.050</td>
</tr>
<tr>
<td>12</td>
<td>Income from OSS Development</td>
<td>0.373</td>
<td>0.931</td>
<td>0.688</td>
<td>1.205</td>
<td>0.570</td>
<td>1.119</td>
</tr>
</tbody>
</table>

The overall step wise D.F.A. results after all significant discriminators have been included in the estimation of discriminated function is given in the following Table 4.43.

Table 4.43 Summary table between low and high groups

<table>
<thead>
<tr>
<th>Step</th>
<th>Variable entered</th>
<th>Wilk’s lambda</th>
<th>Minimum D²</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Working Status</td>
<td>0.973</td>
<td>0.116</td>
<td>*</td>
</tr>
<tr>
<td>2</td>
<td>Educational Qualification</td>
<td>0.951</td>
<td>0.217</td>
<td>*</td>
</tr>
<tr>
<td>3</td>
<td>Gender</td>
<td>0.933</td>
<td>0.306</td>
<td>*</td>
</tr>
<tr>
<td>4</td>
<td>Income from OSSD</td>
<td>0.921</td>
<td>0.367</td>
<td>*</td>
</tr>
</tbody>
</table>

* Significant at 1% level

The summary table indicated that the variable working status entered in step 1, educational qualification entered in step 2, gender in step 3 and variable Income from OSSD entered in the step 4. The variables working status, educational qualification, gender and income from OSSD are significant at 1% level. All the variables are significant discriminator’s based
on their Wilk’s Lambda and $D^2$ value. The multivariate aspect of the model is given in the following Table 4.44.

**Table 4.44 Canonical discriminant function (Between low and high utilizer groups)**

<table>
<thead>
<tr>
<th>Canonical correlation</th>
<th>Wilks Lambda</th>
<th>Chi-square</th>
<th>D.F.</th>
<th>Sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.282</td>
<td>0.921</td>
<td>32.791</td>
<td>4</td>
<td>Significant at 1% level</td>
</tr>
</tbody>
</table>

The canonical correlation is 0.282 when squared is 0.079 that is 7.9% of the variance in the discriminant group can be accounted for by this model, Wilk’s Lambda and chi-square value suggested that D.F. is significant at 1% level.

The variables given above are identified finally by the D.F.A. as the eligible discriminating variables. Based on the selected variables, the corresponding D.F. coefficients are calculated. They are given in the following Table 4.45.

**Table 4.45 Discriminant function coefficients (Between low and high utilizer groups)**

<table>
<thead>
<tr>
<th>Gender (X1)</th>
<th>-1.002</th>
</tr>
</thead>
<tbody>
<tr>
<td>Educational Status (X2)</td>
<td>0.895</td>
</tr>
<tr>
<td>Working Status (X3)</td>
<td>0.846</td>
</tr>
<tr>
<td>Income from OPSS (X4)</td>
<td>0.367</td>
</tr>
<tr>
<td>Constant</td>
<td>-2.585</td>
</tr>
</tbody>
</table>

\[ Z = -2.585 - 1.002 \text{ (Gender)} + 0.895 \text{ (Educational Status)} + 0.846 \text{ (Working Status)} + 0.367 \text{ (Income from OPSSD)} \]
Using this D.F. coefficients and variables, discriminating scores for 2 groups are found out which are called group centroids or group means.

For low maintainers of open source software (\(Z_1\)) it is \(-0.378\)

For high maintainers of open source software (\(Z_2\)) it is \(+0.227\)

Discriminating factor is the weighted average of \(Z_1\) and \(Z_2\).

\[
(\text{ie.}) \quad Z = \frac{(150 \times Z_1) + (250 \times Z_2)}{150 + 250}
\]

If it is represented diagrammatically it will be

Thus, to classify the respondents as to low or high level maintainers of open source software, the Z score for the respondents is found out by using the equation. If the score found out for any respondents is \(Z_0\) and if the value is \(>Z\) (i.e. \(Z_0 > Z\)) then it is classified into high maintainers of open source software and if \(Z_0 < Z\) then (i.e. \(Z_0 < Z\)) it is classified in the low maintainers of open source software.
Now the questions remain to be answered are

1. How efficient are the discriminating variables in the D.F.A.?
2. How efficient the D.F. itself is?

The first question cannot be answered directly however, the discriminating power or the contribution of each variable to the function can sufficiently answer the question. For this, the following table is considered.

Table 4.46 Relative discriminating index (Between low and high maintainers of open source software groups)

<table>
<thead>
<tr>
<th>Variables</th>
<th>Group 1 Mean X1</th>
<th>Group 2 Mean X2</th>
<th>Unstandardized dic. Coeff. (kj)</th>
<th>Ij = ABS (Kj) Mean (Xjo − xji)</th>
<th>Rj = Ij / sum Ij *100</th>
</tr>
</thead>
<tbody>
<tr>
<td>X1</td>
<td>1.640</td>
<td>1.708</td>
<td>0.895</td>
<td>0.535</td>
<td>20.52</td>
</tr>
<tr>
<td>X2</td>
<td>2.213</td>
<td>2.304</td>
<td>0.846</td>
<td>0.375</td>
<td>14.38</td>
</tr>
<tr>
<td>X3</td>
<td>2.013</td>
<td>2.248</td>
<td>0.367</td>
<td>0.172</td>
<td>6.61</td>
</tr>
<tr>
<td>X4</td>
<td>1.753</td>
<td>1.640</td>
<td>-2.585</td>
<td>1.524</td>
<td>58.49</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>2.606</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>100.00</strong></td>
</tr>
</tbody>
</table>

### 4.5.2 Relative Discriminating Index

For each variable, the respective D.F. co-efficient, its mean for each group and R_j are given. R_j called relative discriminating index is calculated from the discriminant function coefficient and group means. R_j tells how much each variable is contributing to the function. By looking at this column, it is found that ‘Income from open source software’ is maximum discriminating variable and ‘Working Status’ is the least discriminating variable.
The second question is answered by reclassifying the already grouped individuals into low or high level maintainer of open source software the D.F.(Z) defined in the equation. This reclassification is called predictor group membership. In short, the efficiency of the D.F. is how correctly it predicts the respondents into respective groups.

Table 4.47  Classification results  (Between low and high maintainer of open source software group)

<table>
<thead>
<tr>
<th>Actual group</th>
<th>No. of cases</th>
<th>Predicted group membership</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td>Group I</td>
<td>Group II</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Low maintainer of open source software)</td>
<td>150</td>
<td>110</td>
<td>73.3%</td>
<td>40</td>
</tr>
<tr>
<td>Group 2</td>
<td>Group I</td>
<td>Group II</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(High maintainer of open source software)</td>
<td>250</td>
<td>113</td>
<td>45.2%</td>
<td>137</td>
</tr>
</tbody>
</table>

Per cent of grouped cases correctly classified : 61.8%.

The above table gives the results of the reclassification. The function, using the variables selected in the analysis classified 61.8% of the cases correctly in the respective groups.

Discriminate Function Analysis was applied to the respondents based on the low and high level maintainers of open source software level. The following factors significantly discriminate the two groups. They are

1. Gender (at 1% level)
2. Educational Status (at 1% level)
3. Working Status (at 1% level)
4. Additional Income from OSSD (at 1% level)
4.6 FACTOR ANALYSIS

Factor Analysis is a method used to transform a set of variables into a small number of linear composites, which have a maximum correlation with original variables. Factor analysis is used to study a complex product (or) services, in order to identify the major characteristics or factors considered important by the respondents. The purpose of factor analysis is to determine whether the responses of several statements favoured by the respondents are significantly correlated. If the responses to the several statements are significantly correlated, it is considered that the statement measures some factors common to all of them.

Factor analysis can only be applied to continuous variables (or) interval scaled variables. Factor analysis is like Regression analysis as it tries to ‘best fit’ the factors to a scatter diagram of data in such a way that the factors explain the variance associated with the responses to each statements. Factor analysis was conducted by the researcher in the present research in the following stages.

1. Desk Research
2. Formulation of questionnaire
3. Collection of data
4. Feeding and processing the input
5. Analyzing the output
6. Identification of factors and naming them
7. Conclusion
4.6.1 Factors Chosen for the Analysis

The level of quality maintained by the volunteers of open source software development was studied by selecting various parameters. Factor analysis, detailed analysis and discussions were done at various stages.

4.6.2 Statistics Associated with Factor Analysis

Bartlett’s Test of Sphericity - Bartlett’s test of sphericity can be used to test the null hypothesis that means that the variable chosen are not correlated with the sample population. The test of sphericity is based on the chi-square transformation of the determination of the correlation matrix. A large value of test statistics favours the rejection of null hypothesis.

Kaiser-Mayer-Olkin Measure of Sampling - This index compares the magnitude of the observed correlation co-efficient to the magnitude of partial correlation co-efficient. Instant small values indicate that the correlation between pairs of variables cannot be explained by other variables and that factor analysis for evaluating a particular aspect will not be appropriate.

Eign Values and Communalities – A factor’s Eigen value or latent route is the sum of squared of its factor loading. It helps us to understand how well a given factor fits the data gathered from all sample respondents on all the statements. Communalities were the sum of squares of a statement’s factor loading, i.e., it explains how much each variables accounts for the factors taken together.

All 20 items given in the questionnaire were selected for factor analysis by using principle component extraction with an orthogonal (Varimax) rotation. The number of factors is unconstrained. For the sake of convergent validity, 0.50 was used as a factor loading cut-off point.
The Factor Matrix is a matrix of loading and correlations between the variable and factors. Pure variables have loading of 0.5 and greater or only one factor. Complex variables may have high loading on more than one factor and they make the interpretation of the output difficulty. The researcher rotated the components seven times to get the significant variables under three factors.

Table 4.49 shows the reliability statistics and proves the data could support 94.1 percentage reliable to do this analysis. Table 4.50 indicates that the Kaiser-Meyer-Olkin (KMO) measures of sampling adequacy in the study are 84.3. This is good result, as it exceeds 0.5 Bartlett’s Test of Sphericity which is 0.000, meaning that factors that form the variables are adequate.

Table 4.48 Reliability statistics

<table>
<thead>
<tr>
<th>Cronbach's Alpha</th>
<th>No. of Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9413</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 4.49 KMO and bartlett’s test

<table>
<thead>
<tr>
<th>Kaiser-Meyer-Olkin Measure of Sampling Adequacy.</th>
<th>0.843</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bartlett's Test of Sphericity</td>
<td></td>
</tr>
<tr>
<td>Approx. Chi-Square</td>
<td>7188.06</td>
</tr>
<tr>
<td>Df</td>
<td>190</td>
</tr>
<tr>
<td>Sig.</td>
<td>0.000</td>
</tr>
</tbody>
</table>

The following table shows the variables with the corresponding extraction communality factor value.
Table 4.50 Variable with extracted communality factor value – significance of open source software and volunteer community

<table>
<thead>
<tr>
<th>Variables</th>
<th>Initial</th>
<th>Extraction</th>
</tr>
</thead>
<tbody>
<tr>
<td>FACTOR 1</td>
<td>1.000</td>
<td>0.636</td>
</tr>
<tr>
<td>FACTOR 2</td>
<td>1.000</td>
<td>0.768</td>
</tr>
<tr>
<td>FACTOR 3</td>
<td>1.000</td>
<td>0.698</td>
</tr>
<tr>
<td>FACTOR 4</td>
<td>1.000</td>
<td>0.738</td>
</tr>
<tr>
<td>FACTOR 5</td>
<td>1.000</td>
<td>0.895</td>
</tr>
<tr>
<td>FACTOR 6</td>
<td>1.000</td>
<td>0.872</td>
</tr>
<tr>
<td>FACTOR 7</td>
<td>1.000</td>
<td>0.650</td>
</tr>
<tr>
<td>FACTOR 8</td>
<td>1.000</td>
<td>0.673</td>
</tr>
<tr>
<td>FACTOR 9</td>
<td>1.000</td>
<td>0.822</td>
</tr>
<tr>
<td>FACTOR 10</td>
<td>1.000</td>
<td>0.757</td>
</tr>
<tr>
<td>FACTOR 11</td>
<td>1.000</td>
<td>0.569</td>
</tr>
<tr>
<td>FACTOR 12</td>
<td>1.000</td>
<td>0.751</td>
</tr>
<tr>
<td>FACTOR 13</td>
<td>1.000</td>
<td>0.651</td>
</tr>
<tr>
<td>FACTOR 14</td>
<td>1.000</td>
<td>0.836</td>
</tr>
<tr>
<td>FACTOR 15</td>
<td>1.000</td>
<td>0.696</td>
</tr>
<tr>
<td>FACTOR 16</td>
<td>1.000</td>
<td>0.758</td>
</tr>
<tr>
<td>FACTOR 17</td>
<td>1.000</td>
<td>0.722</td>
</tr>
<tr>
<td>FACTOR 18</td>
<td>1.000</td>
<td>0.783</td>
</tr>
<tr>
<td>FACTOR 19</td>
<td>1.000</td>
<td>0.739</td>
</tr>
<tr>
<td>FACTOR 20</td>
<td>1.000</td>
<td>0.742</td>
</tr>
</tbody>
</table>

**Extraction Method:** Principal Component Analysis.

where,

Factor 1    - Have the rights to see the source code of the program
Factor 2    - Know more about how a particular program works
Factor 3    - Liberty to modify the OSS we use
Factor 4    - Provide alternatives to proprietary software
Factor 5 - Quality software product
Factor 6 - Ease of Use
Factor 7 - Interface support with proprietary software applications
Factor 8 - Hardware compatibility
Factor 9 - Support for mobile applications
Factor 10 - Main infrastructure for Cloud Computing
Factor 11 - Documentation Guidance
Factor 12 - Global Volunteer Community
Factor 13 - Geographical Virtual Support
Factor 14 - OSS Community Develops Innovative Software
Factor 15 - OSSD is a way to become a better programmer
Factor 16 - OSSPM is the most efficient way to develop software
Factor 17 - OSS Technologies empowers individuals and communities
Factor 18 - Opportunity to interact with like-minded programmers
Factor 19 - As a user of OSS, want to give something back to the Community
Factor 20 - Public Secure Network Communication

4.6.3 Total Variance Explained

The following table revealed that the extraction has been undertaken by using principal-component method and the initial Eigen values are formulated from the communalities table and the same has been developed as extraction sums of squared loadings with percentage of variance and the relative cumulative percentage. From the initial Eigen values and the extraction sums of squared loadings values, the rotation sums of squared loadings has been formulated and shown in the following table.
### Table 4.51 Total variance – level of satisfaction

<table>
<thead>
<tr>
<th>Component</th>
<th>Initial Eigen values</th>
<th>Extraction Sum./s of Squared Loadings</th>
<th>Rotation Sums of Squared Loadings</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
<td>% of Variance</td>
<td>Cumulative %</td>
</tr>
<tr>
<td>1</td>
<td>9.893</td>
<td>49.467</td>
<td>49.467</td>
</tr>
<tr>
<td>2</td>
<td>1.957</td>
<td>9.785</td>
<td>59.252</td>
</tr>
<tr>
<td>4</td>
<td>1.353</td>
<td>6.764</td>
<td>73.777</td>
</tr>
<tr>
<td>5</td>
<td>0.695</td>
<td>4.477</td>
<td>78.254</td>
</tr>
<tr>
<td>6</td>
<td>0.654</td>
<td>3.269</td>
<td>81.523</td>
</tr>
<tr>
<td>7</td>
<td>0.534</td>
<td>2.670</td>
<td>84.193</td>
</tr>
<tr>
<td>8</td>
<td>0.475</td>
<td>2.377</td>
<td>86.570</td>
</tr>
<tr>
<td>9</td>
<td>0.459</td>
<td>2.293</td>
<td>88.862</td>
</tr>
<tr>
<td>10</td>
<td>0.400</td>
<td>2.001</td>
<td>90.863</td>
</tr>
<tr>
<td>11</td>
<td>0.337</td>
<td>1.687</td>
<td>92.550</td>
</tr>
<tr>
<td>12</td>
<td>0.283</td>
<td>1.417</td>
<td>93.966</td>
</tr>
<tr>
<td>13</td>
<td>0.275</td>
<td>1.376</td>
<td>95.343</td>
</tr>
<tr>
<td>14</td>
<td>0.252</td>
<td>1.260</td>
<td>96.602</td>
</tr>
<tr>
<td>15</td>
<td>0.177</td>
<td>0.884</td>
<td>97.486</td>
</tr>
<tr>
<td>16</td>
<td>0.147</td>
<td>0.734</td>
<td>98.220</td>
</tr>
<tr>
<td>17</td>
<td>0.121</td>
<td>0.604</td>
<td>98.824</td>
</tr>
<tr>
<td>18</td>
<td>0.093</td>
<td>0.467</td>
<td>99.291</td>
</tr>
<tr>
<td>19</td>
<td>0.080</td>
<td>0.402</td>
<td>99.693</td>
</tr>
<tr>
<td>20</td>
<td>0.061</td>
<td>0.307</td>
<td>100.000</td>
</tr>
</tbody>
</table>

**Extraction Method:** Principal Component Analysis

The extraction process has been carried out by using principal-component method, and it is found from the rotation sums of squared loadings and the total sum of 20 variables has been extracted and the same has been grouped into 4 components which have eigen value of more than one. It ranges from component No. 1 to component No. 4 with the cumulative percentage from 49.467 per cent to 73.777 per cent. The percentage of variance ranges from 49.467% to 6.764%. For the third component of initial Eigen values, the total, percentage of variance and the cumulative percentage values are 1.353, 6.764% and 73.777 respectively. The extracted sum of squared loadings for the same is 1.353, 6.764% and 73.777% respectively. The rotation sum of squared loadings for the above is 2.166, 10.832 and 73.777 respectively.
From the analysis, it is inferred that the factor analysis has been supported upto 73.777% in this study. This is a good result and made the study reliable to the analysis.

The following table has been formulated by using ‘Principal-Component Method’ for extraction of variables into components and Varimax with Kaiser Normalization has been undergone by using ‘rotation method’. All the twenty variables have been grouped into four components and each component consists of sets of factors and the analysis has been made to identify the influence of one variable over another.

### 4.6.4 Rotated Component Matrix

The Rotated Component Matrix is discussed in the following table. After a factor solution has been obtained, in which all variables have a significant loading on a factor, the researchers attempted to assign some meaning to the pattern of factor loadings. Variables with higher loadings are considered more important and have greater influence on the name or label selected to represent a factor. Researcher examined all the underlined variables for a particular factor and placed greater emphasis on those variables with higher loadings to assign a name or label to a factor that accurately reflected the variables loading on that factor. The names or labels are not derived or assigned by the factor analysis computer programme; rather, the label is intuitively developed by the factor analyst based on its appropriateness for representing the underlying dimension of a particular factor. All four factors have given appropriate names on the basis of the variable represented in each case.
Table 4.52  Rotated component matrix$^a$

<table>
<thead>
<tr>
<th>Variable No.</th>
<th>Component</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Factor 9</td>
<td>0.801</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 20</td>
<td>0.789</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 18</td>
<td>0.757</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 8</td>
<td>0.750</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 10</td>
<td>0.651</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 19</td>
<td>0.635</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 7</td>
<td>0.618</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 13</td>
<td>0.594</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 12</td>
<td>0.550</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 2</td>
<td>0.809</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 4</td>
<td>0.789</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 17</td>
<td>0.707</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 11</td>
<td>0.598</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 3</td>
<td>0.590</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 16</td>
<td></td>
<td>0.847</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 15</td>
<td></td>
<td>0.744</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 14</td>
<td></td>
<td>0.695</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 1</td>
<td></td>
<td>0.621</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Factor 6</td>
<td></td>
<td></td>
<td>0.925</td>
<td></td>
</tr>
<tr>
<td>Factor 5</td>
<td></td>
<td></td>
<td></td>
<td>0.896</td>
</tr>
</tbody>
</table>

**Extraction Method**: Principal Component Analysis

**Rotation Method**: Varimax with Kaiser Normalization

a. Rotation converged in 9 iterations

The above table shows the Rotated Component Matrix, in which the extracted factors are assigning a new naming related together. From the above table, it is noted that all the loading factors which are having the loading value less than 0.5 are rejected from the analysis.
a). Factor 1 is the most important factor which explained 26.717% of the variation. The factors as Support for mobile applications (0.801), Public Secure Network Communication (0.789), Opportunity to interact with like-minded programmers (0.757), Hardware compatibility (0.750), Main infrastructure for Cloud Computing (0.651), As a user of OSS, want to give something back to the Community (0.635), Interface support with proprietary software applications (0.618), Geographical Virtual Support (0.594) and Global Volunteer Community (0.550) are highly correlated with each other. These statements reflect the strength of open source software. Hence, the researcher names this segment respondents are ‘Impact of open source software’.

b). The second kind of factors explained 18.811% of the variances. In this segment, the researchers took the five important variables such as Know more about how a particular program works (0.809), Provide alternatives to proprietary software (0.789), OSS Technologies empowers individuals and communities (0.707), Documentation Guidance (0.598) and Liberty to modify the OSS we use (0.590). These statements are embossed the easy of assessing the OSS and liberty to modify the OSS. Hence, the researcher coined the name as ‘Operative Function of the OSS Development’.

c). This factor explained 17.417% of the variations. In this segment, the researchers took the four important variables. (1) OSSPM is the most efficient way to develop software (0.847), (2) OSSD is a way to become a better programmer (0.744), (3) OSS Community Develops Innovative Software (0.695) and (4) Have the rights to see the source code of the program
These statements are focused upon the development of the OSS methodology. Hence, the researcher named as ‘Efficiency of OSS Development’.

d. The fourth factor explained 10.832% of the variations. The extracted statements are Ease of Use (0.925) and Quality software product (0.896), these statements indicates that the quality of the OSS. Hence, the researcher named this segment as ‘Reliability of OSS’.

4.6.5 Conclusion

The present study has highlighted the significance of open source software and volunteer community into four categories. The volunteers were named the first factor as ‘Impact of open source software’. Second kind of volunteers has been named as ‘Operative Function of the Development’. Third factor was named by the researcher as ‘Efficiency of OSS Development’. The researchers named the fourth following factor as ‘Reliability of OSS’ which is essential for every volunteers.

4.7 CLUSTER ANALYSIS

Cluster analysis is a multivariate statistical technique which groups unknown number of persons / objects / occasions into groups such that the members of each group are having similar characteristics/ attributes (Donald R. Cooper, 2003). The primary objective of Cluster Analysis is to define the structure of the data and identifying the most similar observations to place them into groups. The different groups to be determined in Cluster Analysis are not pre-defined as in Discriminate Analysis. This analysis is ideally suited to segmentation applications in open source software project management. The method of clustering may be either hierarchical or
non-hierarchical or both. The outcome of this analysis is much superior when the results from the hierarchical order are used for the analysis along with the non-hierarchical. Thus, hierarchical and non-hierarchical techniques should be viewed as complementary clustering techniques rather than as competing techniques (Sharma 1996). In this study, the researcher has used both hierarchical and non-hierarchical clustering techniques.

Cluster analysis is typically applied to data recorded on interval scale or continuous scaled variables. This analysis is applied to a large set of data which may consist of many variables. Cluster analysis determines internal homogeneity. i.e., similarities exist among the respondents or items and external heterogeneity i.e. the differences exist across different groups of respondents or items. This analysis helps in grouping the objects or persons based on the variables considered in an analysis or research.

4.7.1 Steps involved in Cluster Analysis

Six steps are basic to the application of most cluster studies.

4.7.1.1 Selection of the sample to be clustered

The sample may be persons (buyers, medical patients, employees etc.) inventory, products events and any other objects. Generally large set of data are more suitable for cluster analysis.

4.7.1.2 Definition of the variables

The variables on which to measure the objects, events or people have to be decided. The set of variables selected should describe the similarity between objects in terms that are relevant to the release management problem. The variables should be selected based on past research theory or a consideration of the hypothesis being tested.
4.7.1.3 Selecting a distance or similarity measure

The computation of similarities among the entities is very essential in order to group the similar objects together. Similarities can be measured either through correlation or distance measures. The most common approach is to measure similarity in terms of the distance between pairs of objects. Various distance measures are used in the Cluster studies.

4.7.1.4 Selecting a clustering procedure

Clustering procedures can be hierarchical, non-hierarchical or application of both the methods for better formation of clusters. Hierarchical clustering is characterized by the development of a hierarchy or tree like structure. Hierarchical methods can be agglomerative or divisive. Agglomerative methods are commonly used in marketing research. Agglomerative clustering starts with each object in a separate cluster. Clusters are formed by grouping objects into bigger and bigger clusters. This process is continued until all objects are members of a simple cluster. Agglomerative method includes linkage methods, error sum of squares or variance methods and centroid methods. Under the linkage method, the researcher have used average linkage method,

4.7.1.5 Selection of mutually exclusive clusters

The number of clusters is selected, based on the Dendogram and the Agglomeration Schedule. The agglomeration schedule shows all possible solutions from cluster 1 to (n-1) clusters, where n is the number of respondents / cases. Going up from the bottom of the agglomeration schedule, the difference in the values of co-efficient column is considered to decide the number of clusters. The decision of selecting the number of clusters is based on the volume of difference in the value of co-efficient column one by one from the bottom of the agglomeration schedule. When the difference in the
values of co-efficients starts decreasing, the number of clusters is stopped with that points where it starts decreasing.

4.7.1.6 Identifying the clustering variables and assessing the validity of clustering

The variables to be grouped in each cluster can be identified with the help of the final cluster centers table. The final cluster center table shows mean values of all the variables in each cluster which helps in the selection of the grouping variables based on the mean values. Case listing cluster membership shows the list of cases coming under each cluster. The ANOVA table shows the F values and variables which are statistically significant.

4.7.2 Procedures Followed to Classify Segments

The interval scaled data (five point scaling: 5-strongly agree, 4-agree, 3-neither agree nor disagree, 2-disagree, 1-strongly disagree) collected from 400 respondents on various demographic characteristics were analysed in 6 stages.

Step - 1: The sample selected for cluster analysis included the people who the respondents involved in open source software project management activities. The researcher took the sample of 400 respondents with various demographic characteristics in Chennai and Coimbatore.

Step -2: The most important part in the clustering problem is selecting the variables on which the clustering is based. The researcher selected the Release Management Activities as the variables which included Testing Process, Defect Handling, Release Activities and Release Management.

Step -3: The similarities among the cases / entities were computed either through correlations, distance measures and other techniques.
Among the various distances measures Squared Euclidean Distance Measure was adopted to compute the similarity between 2 cases in this study.

**Step-4:** In the clustering procedure, Hierarchical Clustering Method was adopted for the I stage. In this method, Agglomerative Method was used with average linkage between groups method. As the agglomeration schedule for 400 was very large, the values from last 20 cases were given in the table and the remaining were listed in the Appendix 2.

**Table 4.53 Agglomeration Schedule**

<table>
<thead>
<tr>
<th>Stage</th>
<th>Cluster combined</th>
<th>Coefficients</th>
<th>Stage cluster first appears</th>
<th>Next stage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cluster 1</td>
<td>Cluster 2</td>
<td>Cluster 1</td>
<td>Cluster 2</td>
</tr>
<tr>
<td>380</td>
<td>122</td>
<td>190</td>
<td>62.00</td>
<td>0</td>
</tr>
<tr>
<td>381</td>
<td>1</td>
<td>155</td>
<td>62.20</td>
<td>378</td>
</tr>
<tr>
<td>382</td>
<td>1</td>
<td>355</td>
<td>62.67</td>
<td>381</td>
</tr>
<tr>
<td>383</td>
<td>1</td>
<td>249</td>
<td>63.49</td>
<td>382</td>
</tr>
<tr>
<td>384</td>
<td>1</td>
<td>63</td>
<td>64.09</td>
<td>383</td>
</tr>
<tr>
<td>385</td>
<td>30</td>
<td>36</td>
<td>64.50</td>
<td>373</td>
</tr>
<tr>
<td>386</td>
<td>1</td>
<td>32</td>
<td>64.77</td>
<td>384</td>
</tr>
<tr>
<td>387</td>
<td>1</td>
<td>122</td>
<td>64.94</td>
<td>386</td>
</tr>
<tr>
<td>388</td>
<td>1</td>
<td>30</td>
<td>65.32</td>
<td>387</td>
</tr>
<tr>
<td>389</td>
<td>1</td>
<td>234</td>
<td>67.55</td>
<td>388</td>
</tr>
<tr>
<td>390</td>
<td>1</td>
<td>213</td>
<td>67.77</td>
<td>389</td>
</tr>
<tr>
<td>391</td>
<td>1</td>
<td>273</td>
<td>68.29</td>
<td>390</td>
</tr>
<tr>
<td>392</td>
<td>1</td>
<td>312</td>
<td>69.23</td>
<td>391</td>
</tr>
<tr>
<td>393</td>
<td>1</td>
<td>363</td>
<td>70.07</td>
<td>392</td>
</tr>
<tr>
<td>394</td>
<td>1</td>
<td>142</td>
<td>70.83</td>
<td>393</td>
</tr>
<tr>
<td>395</td>
<td>1</td>
<td>147</td>
<td>72.92</td>
<td>394</td>
</tr>
<tr>
<td>396</td>
<td>1</td>
<td>49</td>
<td>73.62</td>
<td>395</td>
</tr>
<tr>
<td>397</td>
<td>1</td>
<td>170</td>
<td>74.60</td>
<td>396</td>
</tr>
<tr>
<td>398</td>
<td>1</td>
<td>159</td>
<td>77.06</td>
<td>397</td>
</tr>
<tr>
<td>399</td>
<td>1</td>
<td>199</td>
<td>85.01</td>
<td>398</td>
</tr>
</tbody>
</table>

(The first part of the agglomeration schedule from cases 1 to 379 is given in the Appendix 1)

**Cluster method:** Average linkage between groups method (Hierarchical Clustering Method)

**Distance method:** Squared Euclidean Distance measure
In the above agglomeration schedule, the figures from top to bottom (stage 1 to 399) indicates the sequence in which the cases get combined with other until all 400 cases are combined together in one cluster at the last stage (stage 399). To identify the number of clusters, the co-efficient values (i.e. difference between rows) in column 4 is considered. The figures of co-efficient values were seen from the last row upwards to have the lowest possible number of clusters for interpretation. The difference in the value of co-efficients from stage 399 and stage 398 is 7.95 (85.01-77.06) indicating the 1st cluster. The procedure is continued till the differences between the 2 stages get reduced in order to identify the number of clusters. In the next stage the difference between stage 398 and 397 is 2.46 (77.06-74.60) which is low but again the difference between 397 and 396 is 0.98 (74.60-73.62) indicating the decreasing trend with more difference. But the difference between stages 396 and 395 is 0.71 (73.62-72.92) and stages 395 and 394 is 2.09 (72.92-70.83) indicating the increasing trend with more differences. Hence, it is better to stop with the stage 395 and 394 with the difference of 2.09 indicating a four cluster solution with maximum differences in the value of co-efficients. It was decided consequently to have 4 clusters from the agglomeration schedule.

**Step -5 :** After deciding the number of clusters as 4, the non-hierarchical k-means (quick clustering) Clustering Method was used to find out the psychographic variables in each cluster. The output initial cluster centers, final cluster centers and Anova tables are interpreted to decide the variables in each cluster.
Table 4.54 initial cluster centres

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Variables</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>X1</td>
<td>2.0</td>
</tr>
<tr>
<td>2</td>
<td>X2</td>
<td>4.0</td>
</tr>
<tr>
<td>3</td>
<td>X3</td>
<td>1.0</td>
</tr>
<tr>
<td>4</td>
<td>X4</td>
<td>1.0</td>
</tr>
<tr>
<td>5</td>
<td>X5</td>
<td>3.0</td>
</tr>
<tr>
<td>6</td>
<td>X6</td>
<td>1.0</td>
</tr>
<tr>
<td>7</td>
<td>X7</td>
<td>4.0</td>
</tr>
<tr>
<td>8</td>
<td>X8</td>
<td>2.0</td>
</tr>
<tr>
<td>9</td>
<td>X9</td>
<td>3.0</td>
</tr>
<tr>
<td>10</td>
<td>X10</td>
<td>4.0</td>
</tr>
<tr>
<td>11</td>
<td>X11</td>
<td>2.0</td>
</tr>
<tr>
<td>12</td>
<td>X12</td>
<td>2.0</td>
</tr>
<tr>
<td>13</td>
<td>X13</td>
<td>1.0</td>
</tr>
<tr>
<td>14</td>
<td>X14</td>
<td>1.0</td>
</tr>
<tr>
<td>15</td>
<td>X15</td>
<td>1.0</td>
</tr>
<tr>
<td>16</td>
<td>X16</td>
<td>3.0</td>
</tr>
<tr>
<td>17</td>
<td>X17</td>
<td>1.0</td>
</tr>
<tr>
<td>18</td>
<td>X18</td>
<td>1.0</td>
</tr>
<tr>
<td>19</td>
<td>X19</td>
<td>1.0</td>
</tr>
<tr>
<td>20</td>
<td>X20</td>
<td>4.0</td>
</tr>
<tr>
<td>21</td>
<td>X21</td>
<td>3.0</td>
</tr>
<tr>
<td>22</td>
<td>X22</td>
<td>4.0</td>
</tr>
<tr>
<td>23</td>
<td>X23</td>
<td>2.0</td>
</tr>
<tr>
<td>24</td>
<td>X24</td>
<td>2.0</td>
</tr>
<tr>
<td>25</td>
<td>X25</td>
<td>1.0</td>
</tr>
<tr>
<td>26</td>
<td>X26</td>
<td>4.0</td>
</tr>
<tr>
<td>27</td>
<td>X27</td>
<td>2.0</td>
</tr>
<tr>
<td>28</td>
<td>X28</td>
<td>4.0</td>
</tr>
<tr>
<td>29</td>
<td>X29</td>
<td>5.0</td>
</tr>
<tr>
<td>30</td>
<td>X30</td>
<td>2.0</td>
</tr>
<tr>
<td>31</td>
<td>X31</td>
<td>2.0</td>
</tr>
<tr>
<td>32</td>
<td>X32</td>
<td>2.0</td>
</tr>
<tr>
<td>33</td>
<td>X33</td>
<td>1.0</td>
</tr>
<tr>
<td>34</td>
<td>X34</td>
<td>2.0</td>
</tr>
<tr>
<td>35</td>
<td>X35</td>
<td>3.0</td>
</tr>
<tr>
<td>36</td>
<td>X36</td>
<td>1.0</td>
</tr>
<tr>
<td>37</td>
<td>X37</td>
<td>1.0</td>
</tr>
<tr>
<td>38</td>
<td>X38</td>
<td>2.0</td>
</tr>
<tr>
<td>39</td>
<td>X39</td>
<td>1.0</td>
</tr>
<tr>
<td>40</td>
<td>X40</td>
<td>3.0</td>
</tr>
</tbody>
</table>

The above table shows the initial cluster formations for 40 variables selected with their mean scores.
**Table 4.55  Final cluster centres**

<table>
<thead>
<tr>
<th>S.No.</th>
<th>Variables</th>
<th>Cluster</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X1</td>
<td></td>
<td>2.1</td>
<td>1.8</td>
<td>2.1</td>
<td>1.7</td>
</tr>
<tr>
<td>2</td>
<td>X2</td>
<td></td>
<td>3.5</td>
<td>1.8</td>
<td>2.2</td>
<td>2.1</td>
</tr>
<tr>
<td>3</td>
<td>X3</td>
<td></td>
<td>1.4</td>
<td>1.9</td>
<td>2.1</td>
<td>1.3</td>
</tr>
<tr>
<td>4</td>
<td>X4</td>
<td></td>
<td>1.9</td>
<td>1.8</td>
<td>2.2</td>
<td>1.3</td>
</tr>
<tr>
<td>5</td>
<td>X5</td>
<td></td>
<td>2.1</td>
<td>1.7</td>
<td>2.3</td>
<td>1.3</td>
</tr>
<tr>
<td>6</td>
<td>X6</td>
<td></td>
<td>2.7</td>
<td>1.8</td>
<td>2.3</td>
<td>2.9</td>
</tr>
<tr>
<td>7</td>
<td>X7</td>
<td></td>
<td>3.7</td>
<td>1.8</td>
<td>2.5</td>
<td>2.6</td>
</tr>
<tr>
<td>8</td>
<td>X8</td>
<td></td>
<td>2.0</td>
<td>1.6</td>
<td>2.4</td>
<td>1.0</td>
</tr>
<tr>
<td>9</td>
<td>X9</td>
<td></td>
<td>2.3</td>
<td>1.8</td>
<td>2.4</td>
<td>1.6</td>
</tr>
<tr>
<td>10</td>
<td>X10</td>
<td></td>
<td>3.1</td>
<td>1.8</td>
<td>2.3</td>
<td>1.3</td>
</tr>
<tr>
<td>11</td>
<td>X11</td>
<td></td>
<td>2.0</td>
<td>1.7</td>
<td>2.8</td>
<td>3.1</td>
</tr>
<tr>
<td>12</td>
<td>X12</td>
<td></td>
<td>2.5</td>
<td>1.7</td>
<td>2.6</td>
<td>2.7</td>
</tr>
<tr>
<td>13</td>
<td>X13</td>
<td></td>
<td>1.2</td>
<td>1.9</td>
<td>2.7</td>
<td>2.0</td>
</tr>
<tr>
<td>14</td>
<td>X14</td>
<td></td>
<td>1.1</td>
<td>2.1</td>
<td>2.6</td>
<td>2.0</td>
</tr>
<tr>
<td>15</td>
<td>X15</td>
<td></td>
<td>1.7</td>
<td>2.2</td>
<td>2.4</td>
<td>1.9</td>
</tr>
<tr>
<td>16</td>
<td>X16</td>
<td></td>
<td>2.2</td>
<td>2.1</td>
<td>2.5</td>
<td>1.7</td>
</tr>
<tr>
<td>17</td>
<td>X17</td>
<td></td>
<td>2.1</td>
<td>2.0</td>
<td>2.7</td>
<td>1.9</td>
</tr>
<tr>
<td>18</td>
<td>X18</td>
<td></td>
<td>1.9</td>
<td>2.0</td>
<td>2.9</td>
<td>2.9</td>
</tr>
<tr>
<td>19</td>
<td>X19</td>
<td></td>
<td>2.2</td>
<td>1.9</td>
<td>2.5</td>
<td>3.0</td>
</tr>
<tr>
<td>20</td>
<td>X20</td>
<td></td>
<td>2.9</td>
<td>1.9</td>
<td>2.7</td>
<td>2.7</td>
</tr>
<tr>
<td>21</td>
<td>X21</td>
<td></td>
<td>2.6</td>
<td>1.9</td>
<td>2.5</td>
<td>1.6</td>
</tr>
<tr>
<td>22</td>
<td>X22</td>
<td></td>
<td>3.3</td>
<td>2.0</td>
<td>2.4</td>
<td>2.1</td>
</tr>
<tr>
<td>23</td>
<td>X23</td>
<td></td>
<td>1.9</td>
<td>1.9</td>
<td>2.5</td>
<td>1.6</td>
</tr>
<tr>
<td>24</td>
<td>X24</td>
<td></td>
<td>1.7</td>
<td>2.1</td>
<td>2.2</td>
<td>1.3</td>
</tr>
<tr>
<td>25</td>
<td>X25</td>
<td></td>
<td>1.3</td>
<td>2.1</td>
<td>2.1</td>
<td>1.7</td>
</tr>
<tr>
<td>26</td>
<td>X26</td>
<td></td>
<td>3.2</td>
<td>2.0</td>
<td>2.5</td>
<td>3.3</td>
</tr>
<tr>
<td>27</td>
<td>X27</td>
<td></td>
<td>1.9</td>
<td>1.9</td>
<td>2.6</td>
<td>2.0</td>
</tr>
<tr>
<td>28</td>
<td>X28</td>
<td></td>
<td>3.6</td>
<td>1.9</td>
<td>2.4</td>
<td>1.4</td>
</tr>
<tr>
<td>29</td>
<td>X29</td>
<td></td>
<td>4.8</td>
<td>1.9</td>
<td>2.6</td>
<td>2.1</td>
</tr>
<tr>
<td>30</td>
<td>X30</td>
<td></td>
<td>2.2</td>
<td>1.9</td>
<td>2.6</td>
<td>2.3</td>
</tr>
<tr>
<td>31</td>
<td>X31</td>
<td></td>
<td>2.1</td>
<td>1.9</td>
<td>2.3</td>
<td>3.4</td>
</tr>
<tr>
<td>32</td>
<td>X32</td>
<td></td>
<td>1.9</td>
<td>2.0</td>
<td>2.4</td>
<td>2.3</td>
</tr>
<tr>
<td>33</td>
<td>X33</td>
<td></td>
<td>1.4</td>
<td>1.9</td>
<td>2.5</td>
<td>2.3</td>
</tr>
<tr>
<td>34</td>
<td>X34</td>
<td></td>
<td>2.1</td>
<td>1.9</td>
<td>2.5</td>
<td>2.0</td>
</tr>
<tr>
<td>35</td>
<td>X35</td>
<td></td>
<td>2.5</td>
<td>1.9</td>
<td>2.4</td>
<td>2.4</td>
</tr>
<tr>
<td>36</td>
<td>X36</td>
<td></td>
<td>2.6</td>
<td>2.1</td>
<td>2.6</td>
<td>3.3</td>
</tr>
<tr>
<td>37</td>
<td>X37</td>
<td></td>
<td>1.4</td>
<td>2.1</td>
<td>2.5</td>
<td>2.0</td>
</tr>
<tr>
<td>38</td>
<td>X38</td>
<td></td>
<td>2.1</td>
<td>2.0</td>
<td>2.4</td>
<td>1.7</td>
</tr>
<tr>
<td>39</td>
<td>X39</td>
<td></td>
<td>1.9</td>
<td>1.9</td>
<td>2.3</td>
<td>3.0</td>
</tr>
<tr>
<td>40</td>
<td>X40</td>
<td></td>
<td>2.3</td>
<td>1.9</td>
<td>2.7</td>
<td>3.0</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>2.3</strong></td>
<td><strong>1.9</strong></td>
<td><strong>2.5</strong></td>
<td><strong>2.1</strong></td>
<td></td>
</tr>
</tbody>
</table>

The final cluster centers table given above. It contains the mean values for each variable in each cluster. As the mean values with the respected scores more than the mean score were being selected in each cluster. Hence, in cluster I the variables X2, X6, X7, X10, X20, X21, X22, X26, X28, X29,
X35 and X36 were selected and in cluster 2, the variables selected were X14, X15, X16, X17, X18, X22, X24, X25, X26, X32, X36, X37 and X38. The variables X11, X12, X13, X14, X17, X18, X20, X27, X29, X30, X36 and X40 were included in the III cluster with the mean value greater than 2.5. Finally, the cluster 4, the variables X6, X7, X11, X12, X18, X19, X20, X26, X30, X31, X32, X33, X35, X36, X39 and X40 are selected with the mean value greater than 2.1. From the above table, the variables in each cluster were identified for the four cluster segments.

The researcher found four cluster indexes, which is according to Weingessel et al (1999); Calinski and Harabasz (1974); Ratkowsky and Lance (1978); Scott and Symons (1971); TraceW (Edwards and Cavalli-Sforza 1965) and the Likelihood Index (Wedel and Kamakura 1998).

The number of respondents in each cluster segment is shown in the following Table 4.56.

<table>
<thead>
<tr>
<th>Segment</th>
<th>No. of Respondents</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality assurance testers</td>
<td>20</td>
<td>5.0</td>
</tr>
<tr>
<td>Defect Handlers</td>
<td>179</td>
<td>44.7</td>
</tr>
<tr>
<td>Project Managers</td>
<td>194</td>
<td>48.5</td>
</tr>
<tr>
<td>Communicators</td>
<td>7</td>
<td>1.8</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>400</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Table 4.58, shows the number of respondents in each segment out of the 400 respondents. The I cluster is grouped by 20 respondents (5.0%), II cluster by 179 respondents (44.7%), III cluster by 194 respondents (48.5%) and IV cluster is grouped by 7 respondents (1.8%).
Step-6: The variables in each cluster segment were identified based on the mean values in the final cluster center table. The number of respondents in each cluster was also found as given in the following Table 4.57.

Table 4.57 Cluster formation with variables and mean values

<table>
<thead>
<tr>
<th>Cluster-1 : Quality Assurance Testers (Cronbach’s Alpha=0.886, 5.0% variance)</th>
<th>Cluster-2 : Defect Handlers (Cronbach’s Alpha=0.914, 44.8% variance)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X2 (3.5)</td>
<td>X14 (2.1)</td>
</tr>
<tr>
<td>X6 (2.7)</td>
<td>X15 (2.2)</td>
</tr>
<tr>
<td>X7 (3.7)</td>
<td>X16 (2.1)</td>
</tr>
<tr>
<td>X10 (3.1)</td>
<td>X17 (2.0)</td>
</tr>
<tr>
<td>X20 (2.9)</td>
<td>X18 (2.0)</td>
</tr>
<tr>
<td>X21 (2.6)</td>
<td>X22 (2.0)</td>
</tr>
<tr>
<td>X22 (3.3)</td>
<td>X24 (2.1)</td>
</tr>
<tr>
<td>X26 (3.2)</td>
<td>X25 (2.1)</td>
</tr>
<tr>
<td>X28 (3.6)</td>
<td>X26 (2.0)</td>
</tr>
<tr>
<td>X29 (4.8)</td>
<td>X32 (2.0)</td>
</tr>
<tr>
<td>X35 (2.5)</td>
<td>X36 (2.1)</td>
</tr>
<tr>
<td>X36 (2.6)</td>
<td>X37 (2.1)</td>
</tr>
<tr>
<td>X38 (2.0)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cluster-3 : Project Managers (Cronbach’s Alpha=0.936, 48.5% variance)</th>
<th>Cluster-4 : Communicators (Cronbach’s Alpha=0.946, 1.8% variance)</th>
</tr>
</thead>
<tbody>
<tr>
<td>X11 (2.8)</td>
<td>X6 (2.9)</td>
</tr>
<tr>
<td>X12 (2.6)</td>
<td>X7 (2.6)</td>
</tr>
<tr>
<td>X13 (2.7)</td>
<td>X11 (3.1)</td>
</tr>
<tr>
<td>X14 (2.6)</td>
<td>X12 (2.7)</td>
</tr>
<tr>
<td>X17 (2.7)</td>
<td>X18 (2.9)</td>
</tr>
<tr>
<td>X18 (2.9)</td>
<td>X19 (3.0)</td>
</tr>
<tr>
<td>X20 (2.7)</td>
<td>X20 (2.7)</td>
</tr>
<tr>
<td>X27 (2.6)</td>
<td>X26 (3.3)</td>
</tr>
<tr>
<td>X29 (2.6)</td>
<td>X30 (2.3)</td>
</tr>
<tr>
<td>X30 (2.6)</td>
<td>X31 (3.4)</td>
</tr>
<tr>
<td>X36 (2.6)</td>
<td>X32 (2.3)</td>
</tr>
<tr>
<td>X40 (2.7)</td>
<td>X33 (2.3)</td>
</tr>
<tr>
<td></td>
<td>X35 (2.4)</td>
</tr>
<tr>
<td></td>
<td>X36 (3.3)</td>
</tr>
<tr>
<td></td>
<td>X39 (3.0)</td>
</tr>
<tr>
<td></td>
<td>X40 (3.0)</td>
</tr>
</tbody>
</table>

Note: Figures in parenthesis is percentage
From the above table, it is observed that the respondents’ release management activities play a great role as quality assurance testers from the first cluster of respondents. The respondents in the second cluster opined as Defect handlers, Project managers in the third cluster and Communicators in fourth cluster. Among the four clusters, the first and fourth cluster respondents are found to be small in terms of carrying very less number of Software developers.

To study which of the variables is statistically significant across the 4 clusters, Anova was employed and the result obtained is given in the following Table 4.58.

**Table 4.58 Anova release management activities variables - cluster**

<table>
<thead>
<tr>
<th>S.No</th>
<th>Variables</th>
<th>Cluster Mean square</th>
<th>df</th>
<th>Error Mean square</th>
<th>df</th>
<th>F</th>
<th>Sig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X1</td>
<td>4.389</td>
<td>3</td>
<td>.321</td>
<td>396</td>
<td>13.694</td>
<td>1%</td>
</tr>
<tr>
<td>2</td>
<td>X2</td>
<td>18.975</td>
<td>3</td>
<td>.294</td>
<td>396</td>
<td>64.493</td>
<td>1%</td>
</tr>
<tr>
<td>3</td>
<td>X3</td>
<td>4.663</td>
<td>3</td>
<td>.361</td>
<td>396</td>
<td>12.930</td>
<td>1%</td>
</tr>
<tr>
<td>4</td>
<td>X4</td>
<td>8.334</td>
<td>3</td>
<td>.295</td>
<td>396</td>
<td>28.211</td>
<td>1%</td>
</tr>
<tr>
<td>5</td>
<td>X5</td>
<td>10.013</td>
<td>3</td>
<td>.293</td>
<td>396</td>
<td>34.199</td>
<td>1%</td>
</tr>
<tr>
<td>6</td>
<td>X6</td>
<td>12.889</td>
<td>3</td>
<td>.439</td>
<td>396</td>
<td>29.381</td>
<td>1%</td>
</tr>
<tr>
<td>7</td>
<td>X7</td>
<td>29.447</td>
<td>3</td>
<td>.359</td>
<td>396</td>
<td>82.065</td>
<td>1%</td>
</tr>
<tr>
<td>8</td>
<td>X8</td>
<td>18.612</td>
<td>3</td>
<td>.283</td>
<td>396</td>
<td>65.735</td>
<td>1%</td>
</tr>
<tr>
<td>9</td>
<td>X9</td>
<td>12.917</td>
<td>3</td>
<td>.329</td>
<td>396</td>
<td>39.240</td>
<td>1%</td>
</tr>
<tr>
<td>10</td>
<td>X10</td>
<td>15.151</td>
<td>3</td>
<td>.367</td>
<td>396</td>
<td>41.304</td>
<td>1%</td>
</tr>
<tr>
<td>11</td>
<td>X11</td>
<td>37.444</td>
<td>3</td>
<td>.358</td>
<td>396</td>
<td>104.671</td>
<td>1%</td>
</tr>
<tr>
<td>12</td>
<td>X12</td>
<td>25.037</td>
<td>3</td>
<td>.459</td>
<td>396</td>
<td>54.574</td>
<td>1%</td>
</tr>
<tr>
<td>13</td>
<td>X13</td>
<td>30.582</td>
<td>3</td>
<td>.301</td>
<td>396</td>
<td>101.553</td>
<td>1%</td>
</tr>
<tr>
<td>14</td>
<td>X14</td>
<td>16.834</td>
<td>3</td>
<td>.304</td>
<td>396</td>
<td>55.423</td>
<td>1%</td>
</tr>
<tr>
<td>15</td>
<td>X15</td>
<td>4.711</td>
<td>3</td>
<td>.502</td>
<td>396</td>
<td>9.391</td>
<td>1%</td>
</tr>
</tbody>
</table>
The Anova table helped in identifying which of the 40 variables were significant across the 4 clusters. The last column in the table indicates that all the variables are significant at the 1 per cent level.

<table>
<thead>
<tr>
<th>S.No</th>
<th>Variables</th>
<th>Cluster Mean square</th>
<th>df</th>
<th>Error Mean square</th>
<th>df</th>
<th>F</th>
<th>Sig.</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>X16</td>
<td>6.254</td>
<td>3</td>
<td>.485</td>
<td>396</td>
<td>12.896</td>
<td>1%</td>
</tr>
<tr>
<td>17</td>
<td>X17</td>
<td>14.790</td>
<td>3</td>
<td>.482</td>
<td>396</td>
<td>30.659</td>
<td>1%</td>
</tr>
<tr>
<td>18</td>
<td>X18</td>
<td>26.862</td>
<td>3</td>
<td>.271</td>
<td>396</td>
<td>99.264</td>
<td>1%</td>
</tr>
<tr>
<td>19</td>
<td>X19</td>
<td>14.642</td>
<td>3</td>
<td>.333</td>
<td>396</td>
<td>43.932</td>
<td>1%</td>
</tr>
<tr>
<td>20</td>
<td>X20</td>
<td>21.806</td>
<td>3</td>
<td>.451</td>
<td>396</td>
<td>48.389</td>
<td>1%</td>
</tr>
<tr>
<td>21</td>
<td>X21</td>
<td>13.043</td>
<td>3</td>
<td>.241</td>
<td>396</td>
<td>54.103</td>
<td>1%</td>
</tr>
<tr>
<td>22</td>
<td>X22</td>
<td>11.976</td>
<td>3</td>
<td>.390</td>
<td>396</td>
<td>30.693</td>
<td>1%</td>
</tr>
<tr>
<td>23</td>
<td>X23</td>
<td>12.195</td>
<td>3</td>
<td>.338</td>
<td>396</td>
<td>36.090</td>
<td>1%</td>
</tr>
<tr>
<td>24</td>
<td>X24</td>
<td>3.183</td>
<td>3</td>
<td>.319</td>
<td>396</td>
<td>9.972</td>
<td>1%</td>
</tr>
<tr>
<td>25</td>
<td>X25</td>
<td>4.527</td>
<td>3</td>
<td>.312</td>
<td>396</td>
<td>14.504</td>
<td>1%</td>
</tr>
<tr>
<td>26</td>
<td>X26</td>
<td>13.476</td>
<td>3</td>
<td>.383</td>
<td>396</td>
<td>35.185</td>
<td>1%</td>
</tr>
<tr>
<td>27</td>
<td>X27</td>
<td>18.830</td>
<td>3</td>
<td>.275</td>
<td>396</td>
<td>68.442</td>
<td>1%</td>
</tr>
<tr>
<td>28</td>
<td>X28</td>
<td>23.918</td>
<td>3</td>
<td>.298</td>
<td>396</td>
<td>80.161</td>
<td>1%</td>
</tr>
<tr>
<td>29</td>
<td>X29</td>
<td>54.156</td>
<td>3</td>
<td>.422</td>
<td>396</td>
<td>128.322</td>
<td>1%</td>
</tr>
<tr>
<td>30</td>
<td>X30</td>
<td>14.456</td>
<td>3</td>
<td>.391</td>
<td>396</td>
<td>36.929</td>
<td>1%</td>
</tr>
<tr>
<td>31</td>
<td>X31</td>
<td>8.459</td>
<td>3</td>
<td>.289</td>
<td>396</td>
<td>29.242</td>
<td>1%</td>
</tr>
<tr>
<td>32</td>
<td>X32</td>
<td>6.747</td>
<td>3</td>
<td>.327</td>
<td>396</td>
<td>20.629</td>
<td>1%</td>
</tr>
<tr>
<td>33</td>
<td>X33</td>
<td>15.665</td>
<td>3</td>
<td>.226</td>
<td>396</td>
<td>69.359</td>
<td>1%</td>
</tr>
<tr>
<td>34</td>
<td>X34</td>
<td>12.078</td>
<td>3</td>
<td>.362</td>
<td>396</td>
<td>33.319</td>
<td>1%</td>
</tr>
<tr>
<td>35</td>
<td>X35</td>
<td>7.603</td>
<td>3</td>
<td>.502</td>
<td>396</td>
<td>15.152</td>
<td>1%</td>
</tr>
<tr>
<td>36</td>
<td>X36</td>
<td>11.582</td>
<td>3</td>
<td>.449</td>
<td>396</td>
<td>25.771</td>
<td>1%</td>
</tr>
<tr>
<td>37</td>
<td>X37</td>
<td>10.398</td>
<td>3</td>
<td>.350</td>
<td>396</td>
<td>29.668</td>
<td>1%</td>
</tr>
<tr>
<td>38</td>
<td>X38</td>
<td>5.251</td>
<td>3</td>
<td>.337</td>
<td>396</td>
<td>15.559</td>
<td>1%</td>
</tr>
<tr>
<td>39</td>
<td>X39</td>
<td>8.644</td>
<td>3</td>
<td>.299</td>
<td>396</td>
<td>28.933</td>
<td>1%</td>
</tr>
<tr>
<td>40</td>
<td>X40</td>
<td>20.586</td>
<td>3</td>
<td>.309</td>
<td>396</td>
<td>66.518</td>
<td>1%</td>
</tr>
</tbody>
</table>
The validity and stability of the clusters were checked by splitting the sample into two halves of 200 each and repeating the same procedure of cluster analysis in 2 stages (hierarchical and non-hierarchical). The result shows the 4 cluster solution on both the samples.

4.8 STRUCTURAL EQUATION MODELLING (SEM)

In order to ascertain the quality maintenance of the open source software, various attributes which affecting and encouraging the volunteers were studied. Here the attributes include Software runs in multi Operating System environments, Software is written in a known programming language, Software should be modular, Software should be secure and reliable, Software application should be used in known native language, Software project has large and active developer community, Software project has large and active user/support community and Software development is supported by a Business Enterprise. All these were studied with the help of structural equation model.

Structured Equation Modelling is used to test and eliminate causal relationship using a combination of statistical data and qualitative caused assumptions. It is considered the best approach because SEM unlike other methods does not have limitation on the number of variables. There is no difficulty in hypothesis testing in SEM because it takes the confirmatory approach rather than the exploratory approach. Many sub-criteria are considered under each criterion. The response is arrived for all the sub-criteria from the people involved in the decision making process.

The significance of the criteria as well as the sub-criteria is tested. This is the reason why the relative weightage arrived from SEM is considered more valid than through any other approach. This model also takes measurement error into account when analyzing the data statistically. SEM is capable of estimating or assessing measurement error. It can incorporate both
observed and latent variables. SEM models require less reliance on basic statistical methods.

### 4.8.1 Research Model and Hypothesis Formulation

The research hypotheses have been defined on the basis of the constructs outlined above and using previous research on open source software volunteers. The following figure is a graphic presentation of the developed hypothetical model. On the basis of above presented model, the following hypotheses are proposed.

### 4.8.2 Hypothesis of the Study

Quality attributes are having positive impact with their maintenance of open source software project management activities.

![Figure 4.17 Hypotheses model](image)

Figure 4.17 Hypotheses model
4.8.3 Validity of the Measurements

In Structural Equation Modeling, the Confirmatory Factor Model is imposed on the data. In this case, the purpose of structural equation modeling is twofold. First, it aims to obtain estimates of the parameters of the model, i.e. the factor loadings, the variances and covariances of the factor, and the residual error variances of the observed variables. The second purpose is to assess the fit of the model, i.e. to assess whether the model itself provides a good fit to the data. The ability of SEM to produce a meaningful identification of the correlations between factors is a key strength.

To obtain unstandardized and standardized regression weights, a variance estimate for the residual errors, and the squared multiple correlation of the dependent variable ‘Quality of OSS’. In this case, the calculated value of chi-square test is 119.129 on 20 degrees of freedom, which gives a p-value of 0.00 and this model is a good fit for the analysis. The real strength of SEM is to estimate more complicated path models, with intervening variables between the independent and dependent variables, and latent factor as well.

4.8.4 Maximum Likelihood Estimates

The below table shows the regression coefficient of the exogenous variables. It is noted that the critical ratio of all the selected attribute variables are above the table value 3.707 and it is significant at 1 per cent level. Among the selected eight attributes, all the attributes are the most influenced factors to do their maintenance of quality of OSS in a successful way.
Table 4.59 Regression weights

<table>
<thead>
<tr>
<th>Measured Variable</th>
<th>Latent Variable</th>
<th>Estimate</th>
<th>S.E.</th>
<th>C.R.</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality attribute 8</td>
<td>Quality attribute</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality attribute 7</td>
<td>Quality attribute</td>
<td>1.308</td>
<td>0.228</td>
<td>5.748</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 6</td>
<td>Quality attribute</td>
<td>1.735</td>
<td>0.287</td>
<td>6.040</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 5</td>
<td>Quality attribute</td>
<td>2.390</td>
<td>0.378</td>
<td>6.319</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 4</td>
<td>Quality attribute</td>
<td>1.146</td>
<td>0.211</td>
<td>5.426</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 3</td>
<td>Quality attribute</td>
<td>2.098</td>
<td>0.337</td>
<td>6.232</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 2</td>
<td>Quality attribute</td>
<td>1.838</td>
<td>0.303</td>
<td>6.070</td>
<td>***</td>
</tr>
<tr>
<td>Quality attribute 1</td>
<td>Quality attribute</td>
<td>1.126</td>
<td>0.209</td>
<td>5.373</td>
<td>***</td>
</tr>
</tbody>
</table>

4.8.5 Model Fit Summary

The following table shows that CMIN for the ‘default model’. A significant chi-square indicates satisfactory model fit.

4.8.5.1 CMIN

CMIN is a chi-square statistics comparing the default model and the independence model with the saturated model. The above table infers that the default model has been associated as 5.956 per cent with saturated model and other side, the independence model has been associated as 28.894 per cent with saturated model.

Table 4.60 CMIN

<table>
<thead>
<tr>
<th>Model</th>
<th>NPAR</th>
<th>CMIN</th>
<th>DF</th>
<th>P</th>
<th>CMIN/DF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default model</td>
<td>16</td>
<td>119.129</td>
<td>20</td>
<td>.000</td>
<td>5.956</td>
</tr>
<tr>
<td>Saturated model</td>
<td>36</td>
<td>.000</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Independence model</td>
<td>8</td>
<td>809.031</td>
<td>28</td>
<td>.000</td>
<td>28.894</td>
</tr>
</tbody>
</table>
4.8.5.2 RMR, GFI

The Root Mean Square Residual is the mean absolute value of the covariance residuals, which reflect the difference between observed and model-estimated covariance. Specifically, RMR is the co-efficient which results from taking the square root of the mean of the squared residuals. The closer is RMR is to 0, the better the model fit. The GFI is the goodness-of-fit index and is equal to 1-(chi-square for the default model / chi-square for the null model).

<table>
<thead>
<tr>
<th>Model</th>
<th>RMR</th>
<th>GFI</th>
<th>AGFI</th>
<th>PGFI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default model</td>
<td>.036</td>
<td>.936</td>
<td>.884</td>
<td>.520</td>
</tr>
<tr>
<td>Saturated model</td>
<td>.000</td>
<td>1.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Independence model</td>
<td>.196</td>
<td>.557</td>
<td>.430</td>
<td>.433</td>
</tr>
</tbody>
</table>

From the above table, it is indicated that the model is good fit by the influence of RMR value which is closer to one, i.e., 0.036. GFI (Goodness of Fit Index) refers to 93.6 per cent has been fitted in Default model for the proportion of variance-covariance matrix. On the other hand, 55.7 per cent fit in Independence model.

4.8.5.3 Baseline comparisons

The NFI, Normed Fit Index, also known as (Δ1), was developed as the alternative to CFI, Comparative Fit Index, is also known as the Bentler Comparative Fit Index, compares the existing model fit with the null model which assumes the latent variables correlates with the independent variables.
Table 4.62 Baseline comparisons

<table>
<thead>
<tr>
<th>Model</th>
<th>NFI</th>
<th>RFI</th>
<th>IFI</th>
<th>TLI</th>
<th>CFI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delta1</td>
<td>rho1</td>
<td>Delta2</td>
<td>rho2</td>
<td></td>
</tr>
<tr>
<td>Default model</td>
<td>.853</td>
<td>.794</td>
<td>.874</td>
<td>.822</td>
<td>.873</td>
</tr>
<tr>
<td>Saturated model</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td></td>
</tr>
<tr>
<td>Independence model</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
<td>.000</td>
</tr>
</tbody>
</table>

From the above table, it is noted that the model fit indices are good fit with the evidence of NFI (0.853) and CFI (0.873) which is greater than 0.9.

4.8.5.4 RMSEA

Root Mean Square Error of Approximation is the popular measure of fit, because it does not require comparison with the null model. It is one of the fit indexes less affected by sample size. There is good model fit if RMSEA less than or equal to 0.05.

Table 4.63 RMSEA

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSEA</th>
<th>LO 90</th>
<th>HI 90</th>
<th>PCLOSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default model</td>
<td>.001</td>
<td>.093</td>
<td>.131</td>
<td>.000</td>
</tr>
<tr>
<td>Independence model</td>
<td>.264</td>
<td>.249</td>
<td>.280</td>
<td>.000</td>
</tr>
</tbody>
</table>

It could be noted from the above table that the RMSEA value is 0.001 which is lesser than 0.05 and the model resulted as good fit.
4.8.6 Results

Table 4.64 Bootstrapping

<table>
<thead>
<tr>
<th>Paths</th>
<th>Estimates</th>
<th>SE</th>
<th>Mean</th>
<th>‘t’ Value</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality Attribute 8 &lt;-&gt; Quality of OSS</td>
<td>0.000</td>
<td>0.000</td>
<td>1.000</td>
<td>13.708</td>
<td>H₁ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 7 &lt;-&gt; Quality of OSS</td>
<td>0.247</td>
<td>0.012</td>
<td>1.355</td>
<td>12.919</td>
<td>H₂ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 6 &lt;-&gt; Quality of OSS</td>
<td>0.377</td>
<td>0.019</td>
<td>1.796</td>
<td>12.177</td>
<td>H₃ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 5 &lt;-&gt; Quality of OSS</td>
<td>0.539</td>
<td>0.027</td>
<td>2.487</td>
<td>9.994</td>
<td>H₄ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 4 &lt;-&gt; Quality of OSS</td>
<td>0.356</td>
<td>0.018</td>
<td>1.238</td>
<td>13.315</td>
<td>H₅ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 3 &lt;-&gt; Quality of OSS</td>
<td>0.507</td>
<td>0.025</td>
<td>2.218</td>
<td>11.050</td>
<td>H₆ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 2 &lt;-&gt; Quality of OSS</td>
<td>0.425</td>
<td>0.021</td>
<td>1.954</td>
<td>12.058</td>
<td>H₇ Accepted</td>
</tr>
<tr>
<td>Quality Attribute 1 &lt;-&gt; Quality of OSS</td>
<td>0.391</td>
<td>0.020</td>
<td>1.212</td>
<td>13.360</td>
<td>H₈ Accepted</td>
</tr>
</tbody>
</table>

4.8.6.1 Structural equations: methodology and technical application

The following path analysis is used to prove the selected hypotheses.
Figure 4.18 Resulted hypotheses model

4.8.6.2 Testing of hypotheses

The following table represents the results of the testing of the hypotheses.
Table 4.65 Testing of hypotheses

<table>
<thead>
<tr>
<th>Hypotheses</th>
<th>Hypothetical Relationship</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1 : There is a positive effect of quality attribute 1 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H2 : There is a positive effect of quality attribute 2 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H3 : There is a positive effect of quality attribute 3 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H4 : There is a positive effect of quality attribute 4 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H5 : There is a positive effect of quality attribute 5 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H6 : There is a positive effect of quality attribute 6 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H7 : There is a positive effect of quality attribute 7 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
<tr>
<td>H8 : There is a positive effect of quality attribute 8 and quality of OSS</td>
<td>Positive</td>
<td>Confirmed</td>
</tr>
</tbody>
</table>

4.8.7 Discussion of the Result

From the path diagram, measured variables with latent variable of successful maintenance of the quality of open source software is having positive relationship and also significant at 1 per cent and 5 per cent level. The analysis of the model, from the viewpoint of the pioneer of maintaining the quality of OSS in a successful way, suggests that all the measured variables that is, the different quality attributes are significantly associated on good quality of open source software project management.
SECTION III : 4.9 HENRY GARRETT RANKING TECHNIQUE

4.9.1 Problems Faced by the Respondents

The open source software volunteers are facing the quality maintenance problems when releasing the OSS project management activities. So, the OSS users expectation and need based maintenance is very difficult. For the purpose of this study, the problems such as “Installation guidance needed for all the software released”, “Unable to identify the stable release of a software”, “Software should define the HAL”, “Need adequate support for previous released software”, “Software should be updated without uninstalling the existing application” and “Platform Independent compatibility is needed” were chosen. The problems are studied with the help of Henry Garrett Ranking Technique and the details are shown in the following Table 4.66.

<table>
<thead>
<tr>
<th>No.</th>
<th>Problems</th>
<th>Total Score</th>
<th>Mean Score</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Installation guidance needed for all the software released</td>
<td>20495</td>
<td>51.2</td>
<td>V</td>
</tr>
<tr>
<td>2</td>
<td>Unable to identify the stable release of a software</td>
<td>19297</td>
<td>48.2</td>
<td>II</td>
</tr>
<tr>
<td>3</td>
<td>Software should define the Hardware Accessibility List (HAL)</td>
<td>18654</td>
<td>46.6</td>
<td>I</td>
</tr>
<tr>
<td>4</td>
<td>Need adequate support for previous released software</td>
<td>19260</td>
<td>48.2</td>
<td>IV</td>
</tr>
<tr>
<td>5</td>
<td>Software should be upgraded without uninstalling the existing application</td>
<td>18410</td>
<td>46.0</td>
<td>VI</td>
</tr>
<tr>
<td>6</td>
<td>Platform Independent compatibility is needed</td>
<td>24231</td>
<td>60.6</td>
<td>III</td>
</tr>
</tbody>
</table>

Table 4.66 Problems faced by the respondents
It was learnt from the above table that the problem “Software should define the HAL” was ranked in the first with the total Garrett score of 18654 points. It is followed by the “Unable to identify the stable release of a software” and “Platform Independent compatibility is needed” were ranked second and third with the Garrett scores of 19297 and 24231 points respectively. The problems such as, “Need adequate support for previous released software” and “Installation guidance needed for all the software released” were ranked fourth and fifth with the Garrett scores of 19260 and 20495 points respectively. The last rank was placed to the problem “Software should be updated without uninstalling the existing application” with the Garrett score of 18410 points. It is found from the analysis that maximum of the respondents reported that the problems such as “Software should define the HAL” and “Unable to identify the stable release of a software” are the main problems to be faced at the time of maintaining the quality of open source software.