3-2018

How to Explain Empirical Distribution of Software Defects by Severity

Francisco Zapata
The University of Texas at El Paso, fazg74@gmail.com

Olga Kosheleva
The University of Texas at El Paso, olgak@utep.edu

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Comments:
Technical Report: UTEP-CS-18-22

How to Explain Empirical Distribution of Software Defects by Severity

Francisco Zapata[1], Olga Kosheleva[2], and Vladik Kreinovich[3]

[1] Department of Industrial, Manufacturing, and Systems Engineering
University of Texas at El Paso, El Paso, TX 79968, USA
fazg74@gmail.com
[2] Department of Teacher Education
University of Texas at El Paso, El Paso, TX 79968, USA
olgak@utep.edu
[3] Department of Computer Science
University of Texas at El Paso, El Paso, TX 79968, USA
vladik@utep.edu

Abstract. In the last decades, several tools have appeared that, given a software package, mark possible defects of different potential severity. Our empirical analysis has shown that in most situations, we observe the same distribution or software defects by severity. In this paper, we present this empirical distribution, and we use interval-related ideas to provide an explanation for this empirical distribution.

1 Empirical Distribution of Software Defects by Severity

Automatic detection and classification of defects. Software packages have defects of different possible severity. Some defects allow hackers to enter the system can thus, have a potentially high severity. Other defects are minor and maybe not worth the effort need to correct them. For example, if we declare a variable which is never used (or we declare an array of too big size, so that most of its elements are never used), this makes the program not perfect, but does not have any serious negative consequences other than wasting some computer time on this declaration and wasting some computer memory.

In the last decades, several tools have appeared that, given a software package, mark possible defects of different potential severity; see, e.g., [4].

Usually, software defects which are worth repairing are classified into three categories by their relative severity:

- software defects of very high severity;
- software defects of high severity; and
- software defects of medium severity.

Cautious approach. The main objective of this classification is not to miss any potentially serious defects. Thus, in case of any doubt, a defect is classified into the most severe category possible.

As a result, the only time when a defect is classified into medium severity category is when we are absolutely sure that this defect is not of high or of very high severity. If we have any doubt, we classify this defect as being of high or or very high severity.

Similarly, the only time when a defect is classified as being of high severity is when we are absolutely sure that this defect is of very high severity. If there is any doubt, we classify this defect as being of very high severity.

In particular, in situations in which we have no information about severity of different defects, we should classify all of them as of very high severity. As we gain more information about the consequences of different defects, we can start assigning some of the discovered defects to medium or high severity categories. However, since by default we classify a defect as having high severity:

- the number of defects classified as being of very high severity should still be the largest,
- followed by the number of defects classified as being of high severity,
- and finally, the number of defects classified as being of medium severity should be the smallest of the three.

Empirical results. We applied one of the available software packages to detect and classify defects in several software packages. Here are three typical cases, sorted by the overall number of defects:

	Case 1	Case 2	Case 3
Total number of defects	996	1421	1847
Very high severity defects	543	738	1000
High severity defects	320	473	653
Medium severity defects	133	210	244

Analysis of the empirical results: general case. In all the cases, the numbers of very high, high, and medium severity defects can be approximately described by the ratio 5:3:1. In other words:

- the proportion of software defects of very high severity is close to

$$\frac{5}{5+3+1} = \frac{5}{9} \approx 56\%;$$

- the proportion of software defects of high severity is close to

$$\frac{3}{5+3+1} = \frac{3}{9} \approx 33\%;$$

  and
- the proportion of software defects of medium severity is close to

$$\frac{1}{5+3+1} = \frac{1}{9} \approx 11\%.$$

Let us show it on the example of the above three cases.

Case 1. In this case,

$$\frac{1}{9} \cdot 996 = 110\frac{2}{3},$$

we should:

- observe
$$\frac{1}{9} \cdot 996 = 110\frac{2}{3} \approx 111$$
medium severity defects;
- observe
$$\frac{3}{9} \cdot 996 = 3 \cdot \left(\frac{1}{9} \cdot 996\right) = 3 \cdot 110\frac{2}{3} = 332$$
high severity defects, and
- observe
$$\frac{5}{9} \cdot 996 = 5 \cdot \left(\frac{1}{9} \cdot 996\right) = 5 \cdot 110\frac{2}{3} = 553\frac{1}{3} \approx 553$$
high severity defects.

The actual numbers of defects of different severity are very close to these numbers:

	actual number	predicted number
Very high severity defects	543	553
High severity defects	320	332
Medium severity defects	133	111

The match is up to 20% accuracy, which for the problem of predicting number of software defects is very good.

Case 2. In this case,

$$\frac{1}{9} \cdot 1421 = 157\frac{8}{9},$$

we should:

- observe
$$\frac{1}{9} \cdot 1421 = 157\frac{8}{9} \approx 158$$
medium severity defects;
- observe
$$\frac{3}{9} \cdot 1421 = 3 \cdot \left(\frac{1}{9} \cdot 1421\right) = 3 \cdot 157\frac{8}{9} = 471\frac{8}{3} \approx 474$$
high severity defects, and

– observe

$$\frac{5}{9} \cdot 1421 = 5 \cdot \left(\frac{1}{9} \cdot 1421\right) = 5 \cdot 157\frac{8}{9} = 785\frac{40}{9} \approx 789$$

high severity defects.

The actual numbers of defects of different severity are very close to these numbers:

	actual number	predicted number
Very high severity defects	738	789
High severity defects	473	474
Medium severity defects	210	158

The match is also up to ≈20% accuracy.

Case 3. In this case,

$$\frac{1}{9} \cdot 1847 = 205\frac{2}{9},$$

we should:

– observe

$$\frac{1}{9} \cdot 1847 = 205\frac{2}{9} \approx 205$$

medium severity defects;

– observe

$$\frac{3}{9} \cdot 1847 = 3 \cdot \left(\frac{1}{9} \cdot 1847\right) = 3 \cdot 205\frac{2}{9} = 615\frac{6}{9} \approx 616$$

high severity defects, and

– observe

$$\frac{5}{9} \cdot 1847 = 5 \cdot \left(\frac{1}{9} \cdot 1847\right) = 5 \cdot 205\frac{2}{9} = 1025\frac{10}{9} \approx 1026$$

high severity defects.

The actual numbers of defects of different severity are very close to these numbers:

	actual number	predicted number
Very high severity defects	1000	1026
High severity defects	653	616
Medium severity defects	244	206

The match is also up to 20% accuracy.

2 How to Explain the Empirical Distribution

What we want: a brief reminder. We want to find the three frequencies:

- the frequency p_1 of defects of medium severity;
- the frequency p_2 of defects of high severity, and
- the frequency p_3 of defects of very high severity.

All we know is that $p_1 < p_2 < p_3$.

What we do. We will use ideas related to interval uncertainty and interval computations (see, e.g., [2, 3, 5]) to explain the above empirical dependence.

First idea: let us use intervals instead of exact numbers. In principle, these frequencies can somewhat change from one example to another – as we have seen in the above examples. So, instead of selecting single values p_1, p_2, and p_3, we should select three *regions* of possible values, i.e., we should select:

- an interval $\left[\underline{F}_1, \overline{F}_1\right]$ of possible values of p_1;
- an interval $\left[\underline{F}_2, \overline{F}_2\right]$ of possible values of p_2; and
- an interval $\left[\underline{F}_3, \overline{F}_3\right]$ of possible values of p_3.

To guarantee that $p_1 < p_2$, we want to make sure that every value from the first interval $\left[\underline{F}_1, \overline{F}_1\right]$ is smaller than or equal to any value from the second interval $\left[\underline{F}_2, \overline{F}_2\right]$. To guarantee this, it is sufficient to require that the largest value $\overline{F}_1$ from the first interval is smaller than or equal to the smallest value of the second interval:

$$\overline{F}_1 \le \underline{F}_2.$$

Similarly, to guarantee that $p_2 < p_3$, we want to make sure that every value from the second interval $\left[\underline{F}_2, \overline{F}_2\right]$ is smaller than or equal to any value from the third interval $\left[\underline{F}_3, \overline{F}_3\right]$. To guarantee this, it is sufficient to require that the largest value $\overline{F}_2$ from the first interval is smaller than or equal to the smallest value of the third interval:

$$\overline{F}_2 \le \underline{F}_3.$$

First idea expanded: let us make these intervals as wide as possible. We decided to have intervals of possible values of p_i instead of exact values of the frequencies. To fully follow this idea, let us make these intervals as wide as possible, i.e., let us make sure that it is not possible to increase one of the intervals without violating the above inequalities.

This means that we should have no space left between $\overline{F}_1$ and $\underline{F}_2$ – otherwise, we can expand either the first or the second interval. We should therefore have

$$\overline{F}_1 = \underline{F}_2.$$

Similarly, we should have no space left between $\overline{F}_2$ and $\underline{F}_3$ – otherwise, we can expand either the second or the third interval. We should therefore have

$$\overline{F}_2 = \underline{F}_3.$$

Also, we should have $\underline{F}_1 = 0$ – otherwise, we can expand the first interval.

As a result, we get the division of the interval $[0, F]$ of possible frequencies into three sub-intervals:

- the interval $\left[0, \overline{F}_1\right]$ of possible values of the frequency p_1;
- the interval $\left[\overline{F}_1, \overline{F}_2\right]$ of possible values of the frequency p_2; and
- the interval $\left[\overline{F}_2, F\right]$ of possible values of the frequency p_3.

Second idea: since we have to reason to take intervals of different widths, let us take them equal. We have no a priori reason to assume that the three intervals have different widths. Thus, it is reasonable to assume that these three intervals have the exact same width, i.e., that

$$\overline{F}_1 = \overline{F}_1 - \underline{F}_1 = F - \overline{F}_2.$$

From the equality $\overline{F}_2 - \underline{F}_1 = \overline{F}_1$, we conclude that $\overline{F}_2 = 2\overline{F}_1$. Now, from the condition that $F - \overline{F}_2 = \overline{F}_1$, we conclude that

$$F = \overline{F}_1 + \overline{F}_1 = 2\overline{F}_1 + \overline{F}_1 = 3\overline{F}_1.$$

So, we have the following three intervals:

- the interval $\left[0, \overline{F}_1\right]$ of possible values of the frequency p_1;
- the interval $\left[\overline{F}_1, 2\overline{F}_1\right]$ of possible values of the frequency p_2; and
- the interval $\left[2\overline{F}_1, 3\overline{F}_1\right]$ of possible values of the frequency p_3.

Third idea: which value from the interval should we choose. We would like to select a single "typical" value from each of the three intervals.

If we know the probability of different values from each interval, we could select the average value. We do not know these probabilities, so to use this approach, we need to select one reasonable probability distribution on each interval.

A priori, we have no reason to believe that some values from a given interval are more probable than others. Thus, it is reasonable to conclude that all the values within each interval are equally probable – i.e., that on each of the three intervals, we have a uniform distribution.

Comment. This conclusion can be viewed as a particular case of *Laplace Indeterminacy Principle* – and of its natural generalization, the Maximum Entropy approach; see, e.g., [1].

Now, we are ready to produce the desired probabilities. For the uniform distribution on an interval, the mean value, as one can clearly check, is the midpoint of the interval. So:

- as the estimate for p_1, we select the midpoint of the first interval $\left[0, \overline{F}_1\right]$, i.e., the value

$$p_1 = \frac{0 + \overline{F}_1}{2} = \frac{\overline{F}_1}{2};$$

– as the estimate for p_2, we select the midpoint of the second interval $\left[\overline{F}_1, 2\overline{F}_1\right]$, i.e., the value

$$p_2 = \frac{\overline{F}_1 + 2\overline{F}_1}{2} = 3 \cdot \frac{\overline{F}_1}{2};$$

– finally, as the estimate for p_1, we select the midpoint of the third interval $\left[2\overline{F}_1, 3\overline{F}_1\right]$, i.e., the value

$$p_3 = \frac{2\overline{F}_1 + 3\overline{F}_1}{2} = 5 \cdot \frac{\overline{F}_1}{2}.$$

Conclusion. We see that $p_2 = 3p_1$ and $p_3 = 5p_1$. So, we indeed have an explanation for the empirical ratios 1:3:5 between the the frequencies of software flaws of different severity.

Acknowledgments

This work was supported in part by the US National Science Foundation grant HRD-1242122.

References

1. E. T. Jaynes and G. L. Bretthorst, *Probability Theory: The Logic of Science*, Cambridge University Press, Cambridge, UK, 2003.
2. L. Jaulin, M. Kiefer, O. Didrit, and E. Walter, *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control, and Robotics*, Springer, London, 2001.
3. G. Mayer, *Interval Analysis and Automatic Result Verification*, de Gruyter, Berlin, 2017.
4. Mitre Corp., *Common Weakness Enumerication: A Community-Developed List of Software Weakness Types*, https://cwe.mitre.org/cwss/cwss_v1.0.1.html, accessed on March 2, 1018.
5. R. E. Moore, R. B. Kearfott, and M. J. Cloud, *Introduction to Interval Analysis*, SIAM, Philadelphia, 2009.