ITCS 5102 SURVEY OF PROGRAMMING LANGUAGES

TERM PROJECT

Defect Data Analysis and Experiments for Defect Projection Using R

Team Members:
Praveen Kumar Nelli (800936278)
Akhilesh Bollam (800932633)
Navaneeth Reddy Matta (800935534)
Sai Murali Krishna Sriadibhatla (800937317)
Rajeev Reddy Gujjula (800935750)

TABLE OF CONTENTS

1. ABSTRACT

Software development is a complex task which requires good understanding and sound knowledge on the software system. Research Analysts have proposed different measurements in view of measurable parts of the source code entities such as techniques, classes, documents, or modules, and the social structure of a product that extend with an end goal to clarify the connections between software development and software defects. Notwithstanding, these measurements to a great extent disregard the real usefulness, i.e., the calculated concerns, of a software system framework, which are the principle technical concepts that mirror the business rationale or area of the system. The success of software is always on par with the number of bugs that are to be found and rectified, in order to enhance the quality of the software product. Identifying and locating defects in software projects is a difficult work. Especially, when project sizes grow, this task becomes expensive with sophisticated testing and evaluation mechanisms. The maintainers of the software system play a very crucial role in the maintaining the bug reports which contains details of a particular software failure, and detailed description on how these failures have been regenerated in the system. Defect projection is necessary to reduce the risks of the software defects that may lead to the failure of the software product developed. In this paper, we conducted the extensive data analysis and conducted experiments for the defect projection on the three publicly available defect datasets of software products namely Eclipse, JetSpeed2 and Tomcat.

2. INTRODUCTION TO R LANGUAGE

2.1 WHAT IS R?

R is a language and environment for statistical computing and graphics. It is a Data analysis Software used by Data scientists, statisticians, analysts for data visualization, predictive modelling and statistical analysis. R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering …) and graphical techniques, and is highly extensible.

2.2 WHY R?

It's a mature, widely used (around 1-2 million users) and well-supported free and open source software project; it's committed to an annual schedule for major updates (it runs on GNU/linux, unix, MacOS and Windows). R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- an effective data handling and storage facility,
- a suite of operators for calculations on arrays, in particular matrices,
- a large, coherent, integrated collection of intermediate tools for data analysis,
- graphical facilities for data analysis and display either on-screen or on hardcopy, and

- a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.
- It has excellent graphics capabilities vector arithmetic and also has many built-in basic & advanced statistical and numerical analysis tools.

2.2.1 OPEN SOURCE PROJECT – R IS FREE

As R is an open source project it is free of charge. There is no limit on the number of users or license requirements. Unlike other software packages available in the market like SAS and SPSS, R has no subscription fees. R is open for everyone, so we can inspect the code and tinker with it as much as we like. R is available under the GNU General Public License Version 2.

2.2.2 R IS A LANGUAGE, NOT A TOOL

One can analyze data in R by writing functions and scripts instead of point and click functionality available in most of the data analysis software systems. It is an easy to learn language which provides scope for exploration and often leads to data discovery which might not seem possible otherwise. The benefit of having a script is that it documents all our work starting from data access to reporting and can be run n number of times at any point in time. Hence, making it easier to make updates or changes whenever required. The process of automating a sequence of tasks is made easier by the use of scripts and hence can be easily integrated into other processes.

2.2.3 DATA VISUALIZATION AND GRAPHS

The primary design principles on which R was built is clarity of visualization of data through graphs and charts in the data analysis process. R consists of excellent tools for creating bar charts and scatter-plots. In addition to the basic dataviz functionality included with standard R, there are numerous add-on packages to expand R's visualization capabilities. Some packages are for specific disciplines such as biostatistics or finance; others add general visualization features.

2.2.4 A ROBUST AND VIBRANT COMMUNITY

There are over 2 million users of R around the world and thousands of contributors, hence it becomes easier for anyone working on a project to get answers to their questions. There a community of resources available for R language on the web in almost every other domain.

2.3 HISTORY OF R

R began as an experiment in trying to use the methods of Lisp implementers to build a small testbed which could be used to trial some ideas on how a statistical environment

might be built. Early on, the decision was made to use an S-like syntax. Once that decision was made, the move toward being more and more like S has been irresistible.
R is a public domain (a so called "GNU") project which is similar to the commercial S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S, and is much used in as an educational language and research tool. R was created by Ross Ihaka and Robert Gentleman at the University Of Auckland, New Zealand, and is currently developed by the R Development Core Team, of which Chambers is a member. R is named partly after the first names of the first two R authors and partly as a play on the name of S. R has now outgrown its origins and its development is now a collaborative effort undertaken using the Internet to exchange ideas and distribute the results. The focus is now on how the initial experiment can be turned into a viable piece of free software.

2.4 PROGRAMMING FEATURES OF R:

Like other similar languages such as APL and MATLAB, R supports matrix arithmetic. R's data structures include vectors, matrices, arrays, data frames (similar to tables in a relational database) and lists. R's extensible object system includes objects for (among others): regression models, time-series and geo-spatial coordinates. A scalar is represented as a vector with length one. R supports procedural programming with functions and, for some functions, object-oriented programming with generic functions. A generic function acts differently depending on the type of arguments passed to it. In other words, the generic function dispatches the function (method) specific to that type of object. For example, R has a generic print function that can print almost every type of object in R with a simple print (object name) syntax.

3. INTRODUCTION TO PROJECT

Software analytics speaks to the base segment of the software analysis that generally aims at generating findings, conclusions, and evaluations about software systems and their implementation, composition, behavior, and evolution. Software analytics uses and consolidates methodologies and strategies from statistics, prediction analysis, data mining, and scientific visualization.

Software analytics is used to explore the data in order to predict the defects in the software system which is used to improve the quality of the software. The process of finding the bugs in the software system is one of the cost and time consuming tasks. Reliability models are developed to predict the defects and the failure rates in the software. Large complex multivariate statistical models have been proposed to find a single complexity metric that will account for defects. For the purpose of this project, we have performed study on the publicly available defect data sets. We are interested in the defect-occurrence pattern, which is the rate of defect occurrence as a function of time over the lifetime of a release. We define the lifetime of a release as the duration of time

between when a release becomes generally available and when there are no defect occurrences reported to the software development organization for three consecutive time intervals.

The analysis has been carried out on the defect data sets of three different projects named Eclipse, JetSpeed2 and Tomcat. The In-depth and relevant information cannot be obtained easily by scrutinizing the raw data as such without the sound support of the software analytic technologies. The information obtained by using software analytics is the most relevant information that conveys the proper understanding or knowledge towards performing the given target assignment. In this paper, we focus at answering the following research questions by performing data analysis and machine learning:

1) **How are defect curves of the same product similar to each other?**
2) **What are the modeling qualities of the defect curves using Weibull and Gamma models? What is the best model?**

4. DATASET INFORMATION

The datasets that have been used for this project are publicly available projects from the Apache and Eclipse foundations. Three datasets are provided in the shared project folder and each of the project consists of various versions i.e Eclipse (files E1-E6.txt), Tomcat (files T1-T4.txt) and JetSpeed2 (files J1-J4). Each version file contains a defect curve, i.e. a sequence of numbers of post-release defects per quarter reported for the corresponding version (release) of those software products. The defect data for the project JetSpeed2 has been extracted manually from the raw data present in the Apache foundation.

You might collect data for projects from Apache and Eclipse foundations. Raw bug data of Apache projects could be collected here https://issues.apache.org. Raw bug data of Eclipse projects could be collected here: http://goo.gl/MPM3B0

5. METHODOLOGY

5.1 DATASETS AND ANALYZING SIMILARITY OF DEFECT CURVES FOR THE SOFTWARE PRODUCTS

The datasets that have been used in this paper are the publicly available from the Apache and Eclipse foundations. Each of the projects consists of the various versions i.e. Eclipse (E1-E6), JetSpeed2 (J1-J4) and Tomcat (T1-T4) and each version file contains a defect curve, i.e. a sequence of numbers of post-release defects per quarter reported for the corresponding version of the software products. The defect data for the project JetSpeed2 has been extracted manually from the raw data present in the Apache foundation.

In order to answer the research question Q1.How are defect curves of the same product similar to each other? The Jensen-Shannon Divergence (JSD) of all pairs of normalized (i.e. having sum of 1) defect curves of each product is computed. The Jensen–Shannon divergence is a popular method of measuring the similarity between two

probability distributions. Then box plots are plotted for the JSD scores computed for Eclipse, JetSpeed2 and Tomcat projects.

5.2 JENSEN-SHANNON DIVERGENCE

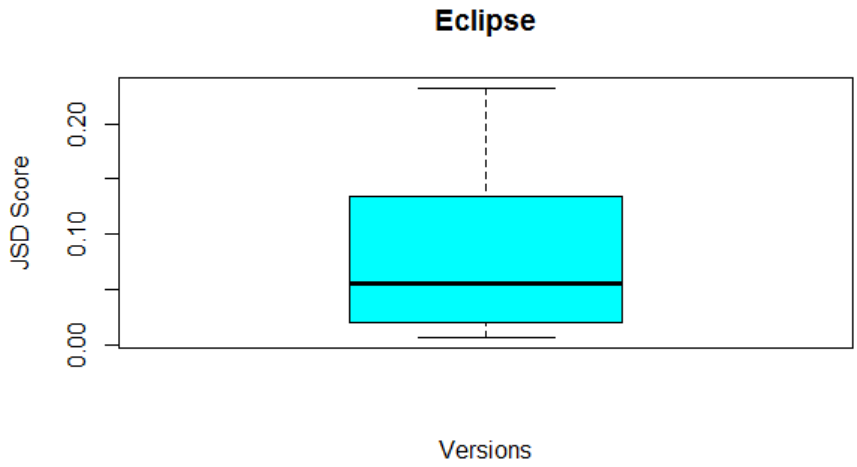
The Jensen-Shanon divergence is a popular method in probability theory and statistics which is used for measuring the similarity between two probability distributions. It is also known as total divergence to the average or information radius (iRad).It is based on Kullback-Leibler divergence, with some notable differences, including that it is symmetric and it is always a finite value. The square root of Jensen-Shanon divergence gives a metric called Jensen-Shanon distance.

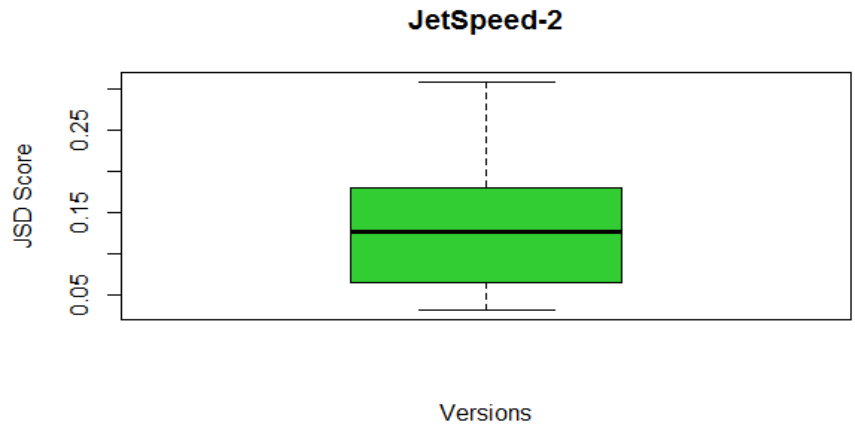
Consider the set M_+^1 (A) of probability distributions where A is a set provided with some σ-algebra of measurable subsets. In particular we can take A to be a finite or countable set with all subsets being measurable.

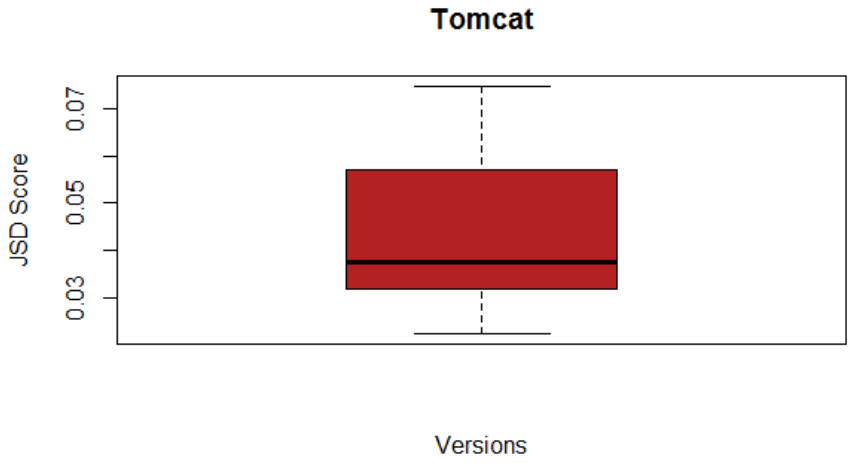
The Jensen–Shannon divergence (JSD) M_+^1 (A) * M_+^1 (A) -> [0, ∞) is a symmetrized and smoothed version of the Kullback-Leibler divergence D(P || Q). It is defined by

$$JSD(P \parallel Q) = (1/2)\, D(P \parallel M) + (1/2)\, D(Q \parallel M)$$

where $M = 1/2\,(P+Q)$







Box plots of Jensen-Shannon Divergence Score of all defect curves for the software products

5.3 ANALYZING MODELS THAT ARE BEST FIT FOR THE DEFECT CURVES

The research question Q2.What are the modeling quality (i.e. the goodness of fit) of the defect curves using Weibull and Gamma models? What is the best model? It is answered by computing the fitness of the curves using the Weibull and Gamma models.

Weibull Distribution model: The probability density function of a Weibull random variable is:

$$f(x;\, \lambda, k) = \begin{cases} \left(\frac{k}{\lambda}\right)\left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^{k}}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

where k > 0 is the shape parameter and λ> 0 is the scale parameter of the distribution.

Gamma Distribution model: The gamma distribution is a two-parameter family of continuous probability distributions. The probability density function using the Shape-scale parameterization is:

$$f(x; k, \theta) = \frac{x^{k-1} e^{-\frac{x}{\theta}}}{\theta^{k}\, \tau(k)} \; for\; x > 0 \; and \; k, \theta > 0$$

Here τ(k) is the gamma function evaluated at k. Then R2 and AIC scores are computed in order to analyze the quality of the fitted models.

Shape: In probability theory and statistics, a shape parameter is a kind of numerical parameter of a parametric family of probability distributions. A shape parameter is any parameter of a probability distribution that is neither a location parameter nor a scale

parameter nor a function of either or both of these only, such as a rate parameter. Such a parameter must affect the shape of a distribution rather than simply shifting it as a location parameter does or stretching/shrinking it as a scale parameter does.

A Scale parameter is a kind of numerical parameter of a parametric family of probability distribution in probability theory and statistics. If a family of probability distribution is such that there is a parameter s and other parameters θ for which the cumulative distribution function satisfies then s is called a scale parameter, since its value determines the "scale" or statistical dispersion of the probability distribution. If s is large, then the distribution will be more spread out otherwise it will be more concentrated.

Akaike information criterion (AIC) is a measure of the relative quality of a statistical model, for a given set of data. As such, AIC provides a means for model selection. AIC deals with the trade-off between the goodness of fit of the model and the complexity of the model. In the general case, the AIC is:

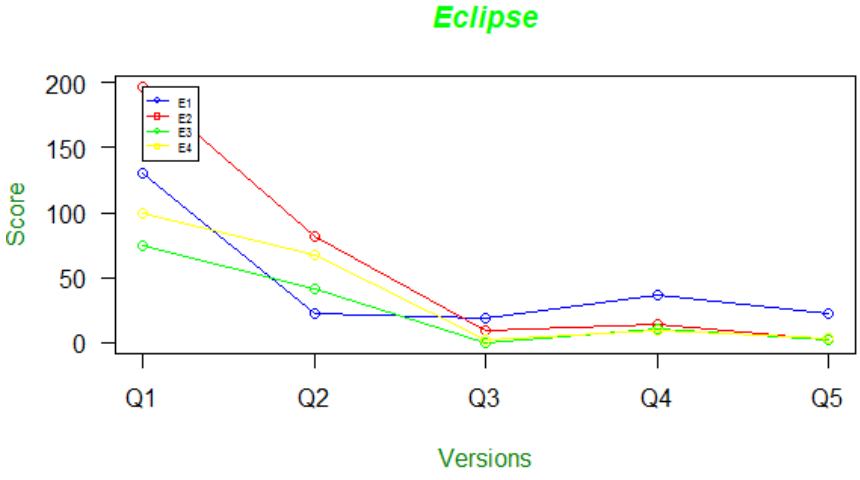
$$AIC = 2k-2\ln(L)$$

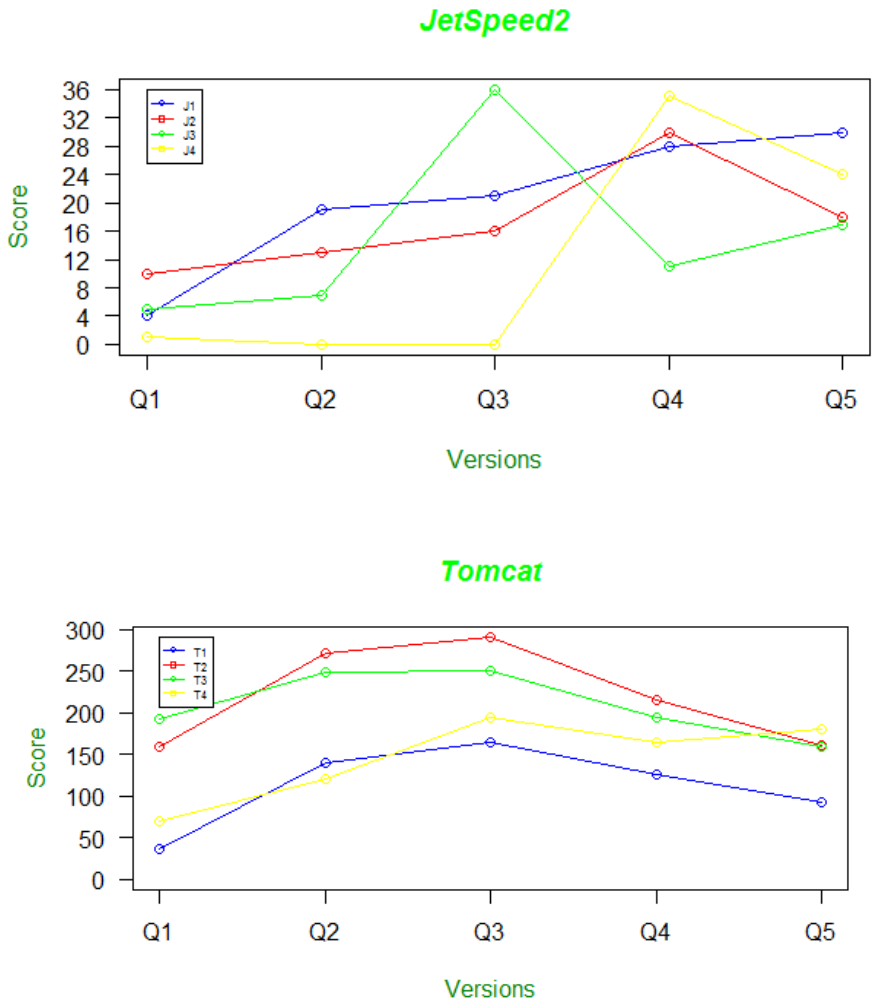
where k is the number of parameters in the statistical model and L is the maximized value of the likelihood function for the estimated model.

R^2 Score indicates how well data points fit a statistical model sometimes simply a line or curve. It is a statistic used in the context of statistical models whose main purpose is either the prediction of future outcomes or the testing of hypotheses, on the basis of other related information. It provides a measure of how well observed outcomes are replicated by the model, as the proportion of total variation of outcomes explained by the model. It is computed as

$$R^2 = 1-\text{var}(y' - y)/\text{var}(y)$$

where y is the actual defect counts and y' is the corresponding fitted values.
We used Non-Linear Least Squares method to find out the best fit of the defect curve to the distribution. The fitness is given by R^2 and AIC values. R^2 is the similarity between the two curves. So higher the value of R^2 higher the fit and lower the AIC values highest is the fit.





Actual curves for Eclipse, JetSpeed2 and Tomcat projects

6. RESULTS AND DISCUSSIONS:

Defect Curves Similarity: The box plots clearly shows that the lower the JSD scores higher the similarity, therefore Eclipse project has the lowest JSD Score when compared to the other projects, which clearly states that the defect curves of various versions in the Eclipse project are more similar to each other.

Model that best fits to the Defect curve: From the results present in Table which represents the AIC and R^2 scores of the various models. The model that has the lower AIC score and the higher R^2 Score clearly depicts the model that best fits the curve. In order to know which model best fit the curve from Table, we can say that among the

various distribution models to predict the best fit to the curve, Gamma model has the lower AIC scores and higher R2 Scores when compared to other models for the projects.

Versions	AIC		R^2	
	Weibull	Gamma	Weibull	Gamma
E1	100.49829	81.76729	-0.10979932	0.8847013
E2	96.63743	53.93029	0.02583854	0.9947824
E3	72.84157	49.05161	0.08493834	0.9620674
E4	85.82247	55.85391	0.18319072	0.9774884
E5	99.37555	62.37531	0.01345181	0.9898744
E6	43.38528	19.24029	-0.27484018	0.9995991
J1	98.39184	60.57714	-0.17696595	0.9213518
J2	85.71320	61.88704	-0.28760671	0.7728033
J3	87.24628	70.76061	-0.18585146	0.5804936
J4	87.44659	65.00155	-0.01327446	0.8281104
T1	278.56945	151.69950	0.22471835	0.9929323
T2	306.78178	197.78666	0.32337139	0.9879267
T3	304.54911	153.18625	0.34488460	0.9975214
T4	298.01770	211.33785	0.05668024	0.9542637

TABLE: Measuring Model Quality using AIC and R^2 Scores

7. CONCLUSION:

This paper discusses about the Defect Projection model which uses the defect curves of various datasets such as Eclipse, JetSpeed2 and Tomcat. The similarity between the curves of various versions in a product is determined. The curves are then fit into a statistical distribution (like Weibull, Gamma models) and the best fit to the curve is determined. In this project for the given scale and shape factors of the projects Gamma distribution was determined to be the best fit to the curve.

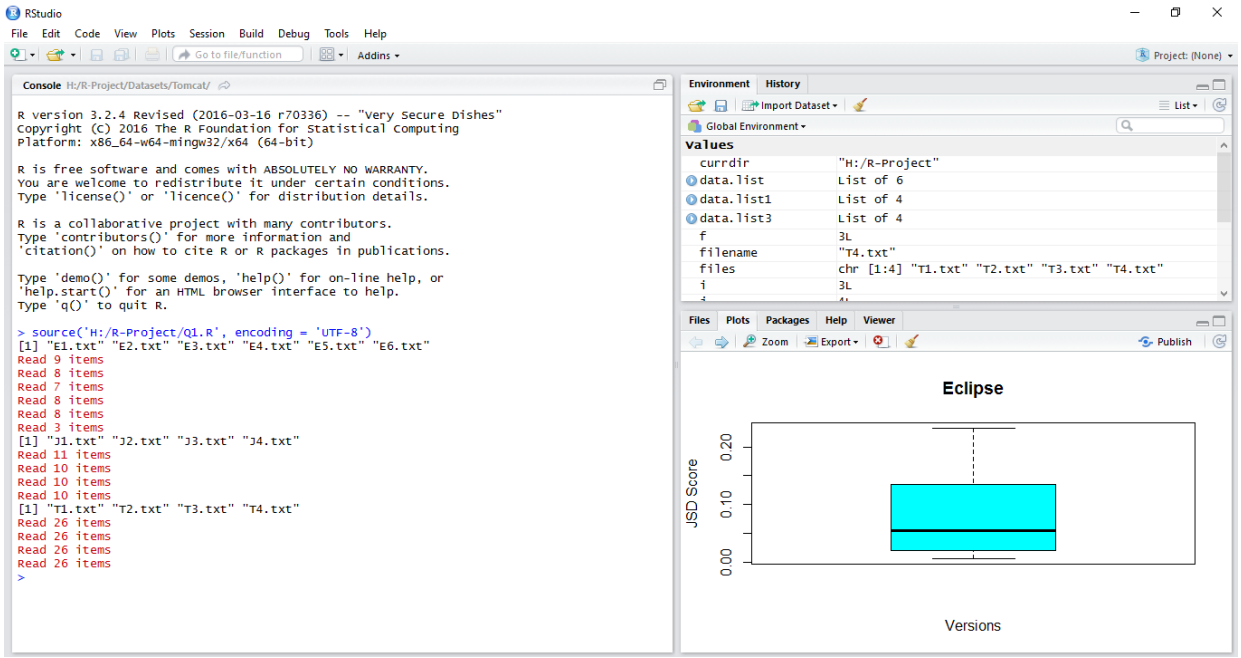
8. STEP WISE EXECUTION

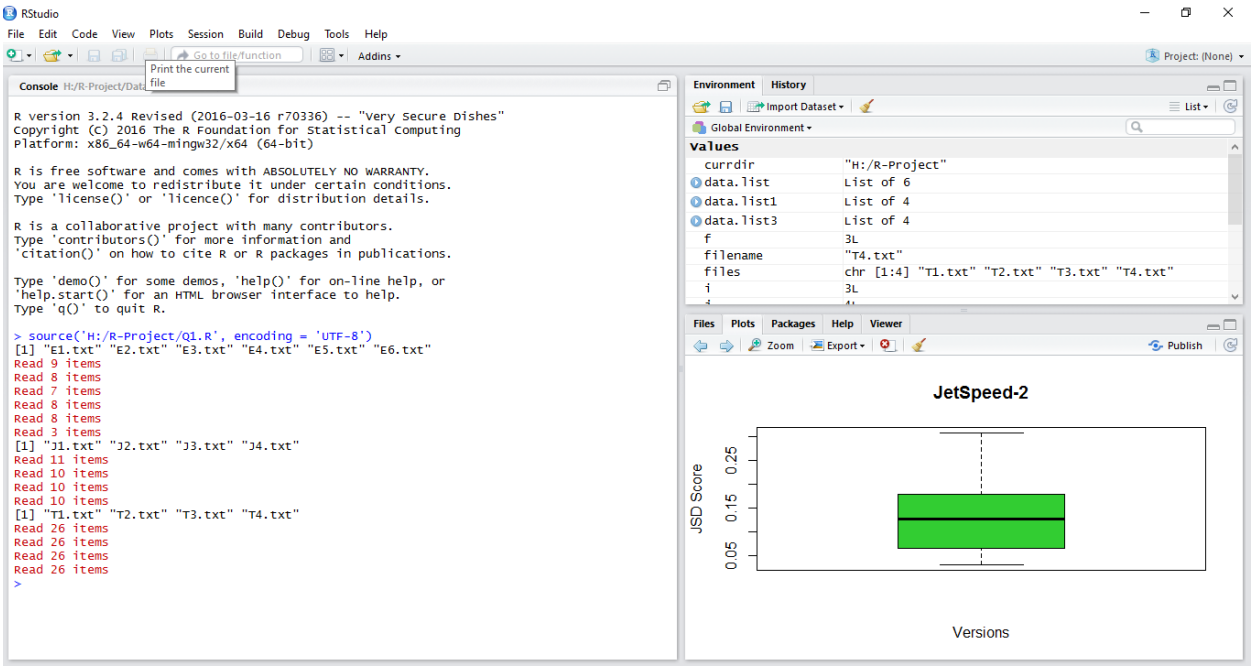
We have two programs (namely Q1.R and Q2.R) to address the questions about

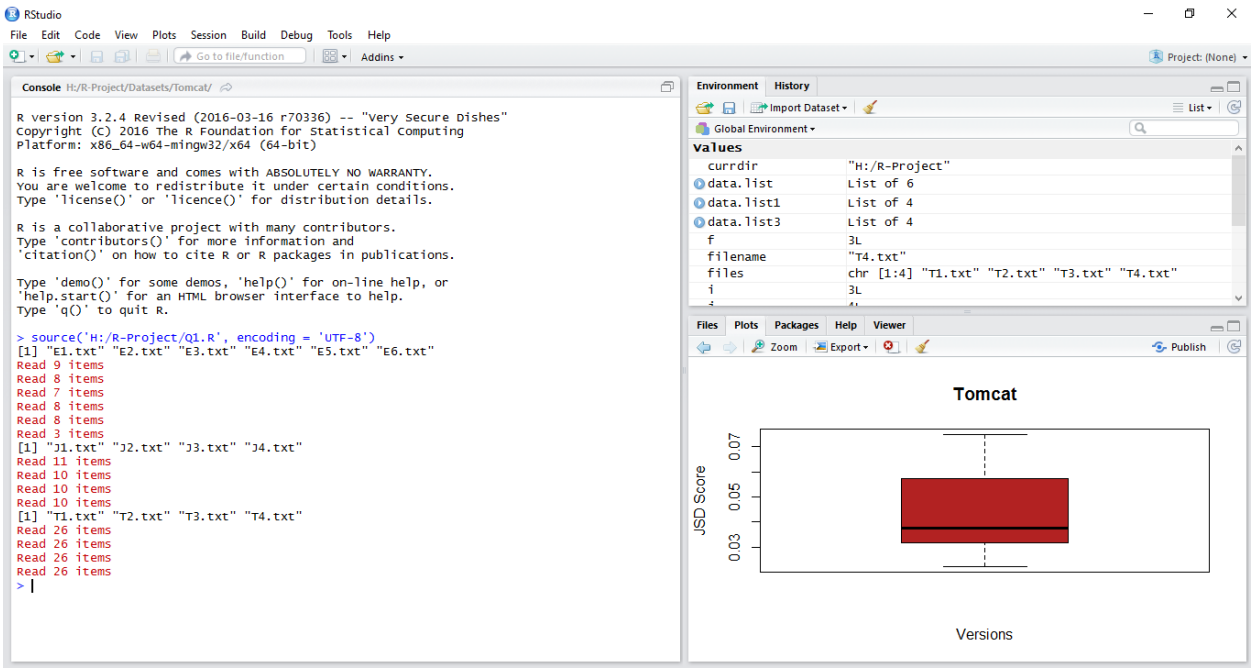
a. How are defect curves of the same product similar to each other?
b. What are the modeling qualities of the defect curves using Weibull and Gamma models? What is the best model?

To execute the first program:

Go to R Studio -> Code (tab at the top) -> Source file and the select the file Q1.R in the project directory

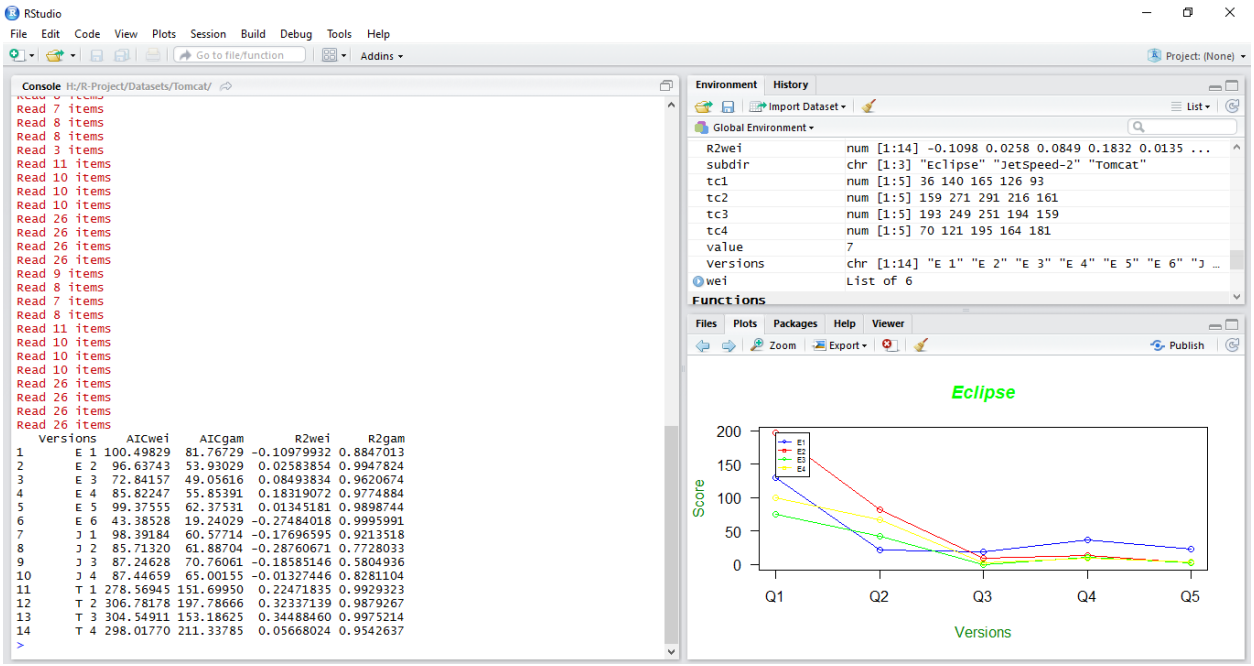


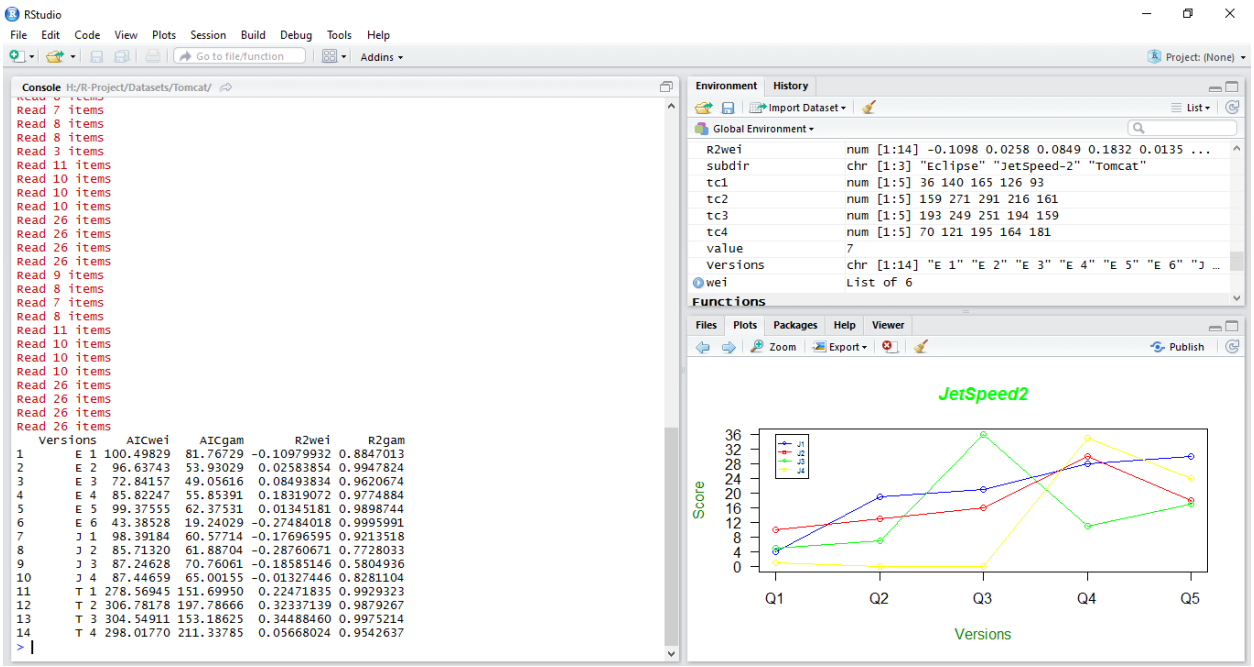

RStudio
File Edit Code View Plots Session Build Debug Tools Help
Print the current file
Addins
Project: (None)
Console H:/R-Project/Dat
R version 3.2.4 Revised (2016-03-16 r70336) -- "Very Secure Dishes"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
> source('H:/R-Project/Q1.R', encoding = 'UTF-8')
[1] "E1.txt" "E2.txt" "E3.txt" "E4.txt" "E5.txt" "E6.txt"
Read 9 items
Read 8 items
Read 7 items
Read 8 items
Read 8 items
Read 3 items
[1] "J1.txt" "J2.txt" "J3.txt" "J4.txt"
Read 11 items
Read 10 items
Read 10 items
Read 10 items
[1] "T1.txt" "T2.txt" "T3.txt" "T4.txt"
Read 26 items
Read 26 items
Read 26 items
Read 26 items
>
Environment History
Import Dataset
List
Global Environment
Values
currdir "H:/R-Project"
data.list List of 6
data.list1 List of 4
data.list3 List of 4
f 3L
filename "T4.txt"
files chr [1:4] "T1.txt" "T2.txt" "T3.txt" "T4.txt"
i 3L
Files Plots Packages Help Viewer
Zoom Export Publish
JetSpeed-2
JSD Score
0.05 0.15 0.25
Versions

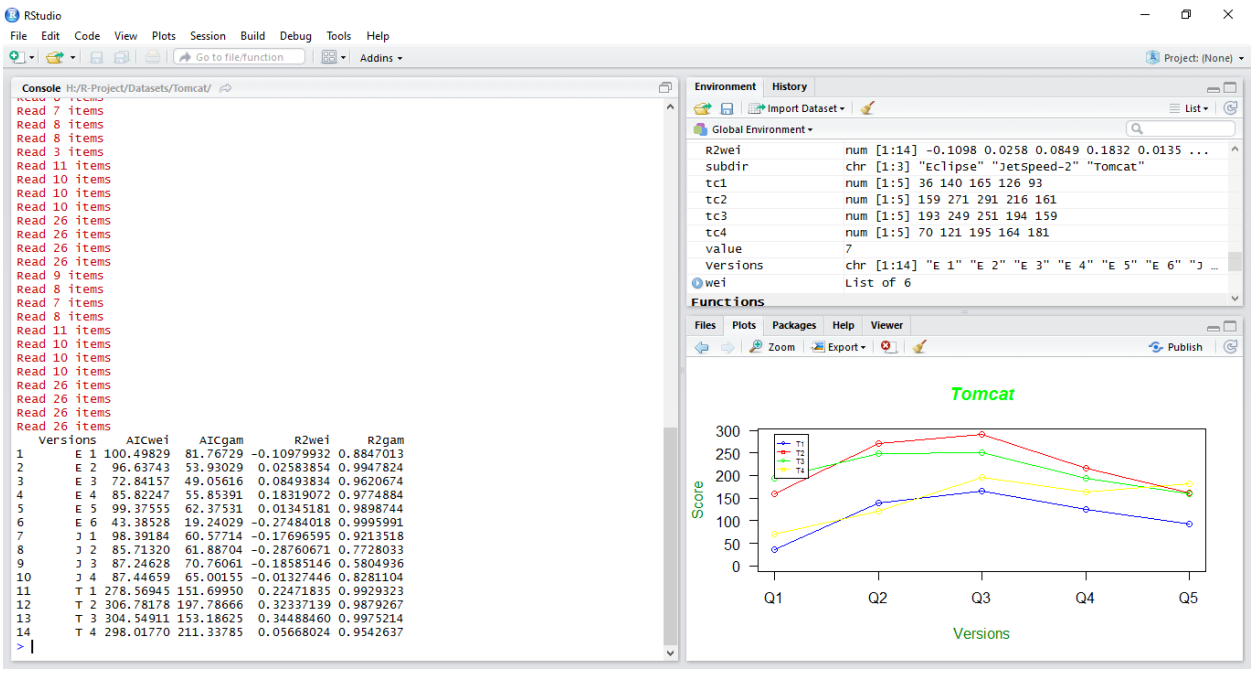

RStudio
File Edit Code View Plots Session Build Debug Tools Help
Addins
Project: (None)
Console H:/R-Project/Datasets/Tomcat/
R version 3.2.4 Revised (2016-03-16 r70336) -- "Very Secure Dishes"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
> source('H:/R-Project/Q1.R', encoding = 'UTF-8')
[1] "E1.txt" "E2.txt" "E3.txt" "E4.txt" "E5.txt" "E6.txt"
Read 9 items
Read 8 items
Read 7 items
Read 8 items
Read 8 items
Read 3 items
[1] "J1.txt" "J2.txt" "J3.txt" "J4.txt"
Read 11 items
Read 10 items
Read 10 items
Read 10 items
[1] "T1.txt" "T2.txt" "T3.txt" "T4.txt"
Read 26 items
Read 26 items
Read 26 items
Read 26 items
>
Environment History
Import Dataset
List
Global Environment
Values
currdir "H:/R-Project"
data.list List of 6
data.list1 List of 4
data.list3 List of 4
f 3L
filename "T4.txt"
files chr [1:4] "T1.txt" "T2.txt" "T3.txt" "T4.txt"
i 3L
Files Plots Packages Help Viewer
Zoom Export Publish
Tomcat
JSD Score
0.03 0.05 0.07
Versions

To execute the second program:

Go to R Studio -> Code (tab at the top) -> Source file and the select the file Q2.R in the project directory







9. REFERENCES:

[1] Tse-Hsun Chen, Stephen W. Thomas, Meiyappan Nagappan, Ahmed E. Hassan: Explaining Software Defects Using Topic Models.

[2] Tim Klinger P. Santhanam Tung Thanh Nguyen, Evelyn Duesterwald and Tien N. Nguyen: Characterizing defect trends in software support.

[3] T Patrick Knab,Martin Pinzger, Abraham Bernstein: Predicting defect densities in source code files with decision tree learners.

[4] "Divergence measures based on the shannon entropy". IEEE Transactions on Information Theory