

USING TIME SERIES MODELS FOR DEFECT PREDICTION
IN SOFTWARE RELEASE PLANNING

A Thesis

Presented to

The Graduate Faculty

Central Washington University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

Computational Science

by

James W. Tunnell

June 2015

CENTRAL WASHINGTON UNIVERSITY

Graduate Studies

We hereby approve the thesis of

James W. Tunnell

Candidate for the degree of Master of Science

APPROVED FOR THE GRADUATE FACULTY

Dr. John K. Anvik, Committee Chair

Dr. Yvonne Chueh

Dr. Kathryn Temple

Dean of Graduate Studies

ABSTRACT

USING TIME SERIES MODELS FOR DEFECT PREDICTION IN SOFTWARE RELEASE PLANNING

by

James W. Tunnell

June 2015

To produce a high-quality software release, sufficient time should be allowed for testing and fixing defects. Otherwise, there is a risk of slip in the development schedule and/or software quality. A time series model is used to predict the number of bugs created during development. The model depends on the previous numbers of bugs created. The model also depends, in an exogenous manner, on the previous numbers of new features resolved and improvements resolved. This model structure would allow hypothetical release plans to be compared by assessing their predicted impact on testing and defect-fixing time. The VARX time series model was selected as a reasonable approach. The accuracy of the model varies for different sampling periods, window sizes, and degree of differencing.

ACKNOWLEDGEMENTS

The author is grateful to Dr. John Anvik, for his advice and patience, to Dr. Yvonne Chueh for her help with exploratory data analysis, and to Dr. Kathryn Temple for her guidance with time series modeling.

TABLE OF CONTENTS

Chapter		Page
I	INTRODUCTION	1
II	LITERATURE REVIEW	3
III	MOTIVATION	6
IV	BACKGROUND	12
V	METHODS	16
VI	RESULTS	26
VII	DISCUSSION	43
VIII	THREATS TO VALIDITY	45
IX	FUTURE WORK.....	48
X	CONCLUSION.....	52
	REFERENCES	53
	APPENDIXES	55

LIST OF TABLES¹

Table		Page
1	The results of sampling example issues.....	19
2	The date ranges of data collected, and the number issues that resulted.....	27
3	The sliding windows sizes to be used for each sampling period	28
4	The parameter values selected from exploratory modeling.	32
5	A comparison of the final modeling results across datasets	42
6	A comparison of the results for full and restricted sample ranges.....	49

¹ The tables shown in the Appendixes are not listed here.

LIST OF FIGURES²

Figure	Page
1 Using an explanatory model for defect prediction.....	7
2 Applying the defect prediction model to estimate overall cost	10
3 An overview of data methods	16
4 Sampling example issue data	19
5 An illustration of time-windowing	21
6 The none-valid proportion, using the MongoDB <i>core server</i> dataset	30
7 The in-interval proportion, using the MongoDB <i>core server</i> dataset.....	31
8 The actual and predicted distributions of the number of bugs.....	34
9 Histogram of forecast mean errors over sliding window.....	34
10 Q-Q plot of forecast mean errors	35
11 The actual and predicted distributions of the number of bugs.....	36
12 Histogram of forecast mean errors over sliding window.....	37
13 Q-Q plot of forecast mean errors	37
14 The actual and predicted distributions of the number of bugs.....	38
15 Histogram of forecast mean errors over sliding window.....	39
16 Q-Q plot of forecast mean errors	39
17 The actual and predicted distributions of the number of bugs.....	40
18 Histogram of forecast mean errors over sliding window.....	41
19 Q-Q plot of forecast mean errors	42

² The figures shown in the Appendixes are not listed here.

LIST OF FIGURES (CONTINUED)

Figure		Page
20	Forecast errors by window reveal the location of an outlier.....	49
21	Undifferenced time series data from the Hibernate <i>orm</i> dataset.....	50