

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
	The Next Release Problem.....	7
	The Gap between Abstraction and Reality.....	9
IV	BACKGROUND	12
	Time Series Models.....	12
	Trends and Stationarity	13
V	METHODS	16
	Data Methods	16
	Modeling Methods	21
VI	RESULTS	26
	Data Results	26
	Modeling Results	28
VII	DISCUSSION	43
VIII	THREATS TO VALIDITY	45
IX	FUTURE WORK.....	46
X	CONCLUSION.....	49
	REFERENCES	50
	APPENDIXES	49
	Appendix A—Time Series Data Plots	52
	Appendix B—Stationarity Testing Results	58
	Appendix C—Exploratory Modeling Results	65

LIST OF TABLES¹

Table		Page
1	Results of sampling example issue data	19
2	Date ranges of data collected, and the number issues that resulted.	26
3	Sliding windows sizes to be used for each sampling period.....	27
4	Parameter values selected for final modeling.	32
5	A comparison of the final modeling results across datasets	42

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

LIST OF FIGURES²

Figure		Page
1	An explanatory model.....	7
2	Applying defect prediction in the Next Release Problem.....	10
3	An overview of data methods	16
4	Sampling issue data	19
5	An illustration of time-windowing	21
6	Plot of the none-valid proportion.....	30
7	Plot of the 95% in-interval proportion.....	31
8	Distributions for actual and predicted number of bugs, MongoDB <i>core server</i> dataset	34
9	Histogram of prediction errors, MongoDB <i>core server</i> dataset	34
10	Q-Q plot of forecast mean errors, MongoDB <i>core server</i> dataset	35
11	Distributions for actual and predicted number of bugs, Hibernate <i>orm</i> dataset	36
12	Histogram of prediction errors, Hibernate <i>orm</i> dataset	37
13	Q-Q plot of forecast mean errors, Hibernate <i>orm</i> dataset	37
14	Distributions for actual and predicted number of bugs, NetBeans <i>platform</i> dataset	38
15	Histogram of prediction errors, NetBeans <i>platform</i> dataset	39

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

16	Q-Q plot of forecast mean errors, NetBeans <i>platform</i> dataset	39
17	Distributions for actual and predicted number of bugs, NetBeans <i>java</i> dataset	41
18	Histogram of prediction errors, NetBeans <i>java</i> dataset	41
19	Q-Q plot of forecast mean errors, NetBeans <i>java</i> dataset	42
20	Undifferenced time series data from the Hibernate <i>orm</i> dataset.....	47