

PCA Analysis of Few Parameters Role in Software Development

Dr Seetharam K
VTU, RLJIT
Bangalore,
India- 561203

Dr.Venkata Subba Reddy
S V University
Tirupathi
India- 517502

Laxmi B.Rananavare
Research Scholar, S V University
Tirupathi
India-517502

ABSTRACT

Principal component analysis is one of the dimension reduction methods with the goal of using the correlation structure among the predictor variables. Qualitative/quantitative measurement of software quality related aspects in all stages of software development are desirable [9,10,11,12]. Any measurement using any element in the software metrics is helpful for analysis in the set of software quality metrics. In this paper seventeen software metric variables [4], are considered. Four cases are carried out using principle component analysis. First analysis is with size as predominant factor. Second analysis is with effort as predominant factor. Third analysis is with duration as predominant factor. Finally all the three association taken together used for analysis of quality performance. The analysis of variables is to identify the dimension that are latent [7,8].

Keywords

Size, Efforts, Duration, Eigen values, Fuzzy logic, Variance.

1. INTRODUCTION

In this paper seventeen variables considered are size, effort, duration, S1 (customer participation), S2 (staff availability), S3(standards use) and S4 (methods use) S5=(Tools use), S6(software logical complexity), S7(requirement volatility), S8(quality requirement), S9(efficiency requirement), S10(Installation requirements), S11(Staff analysis skills), S12(staff application knowledge), S13(staff tool skills), and S14 (staff team skills). Four cases are carried out using Principle component analysis.

2. ANALYSIS

Five different levels are identified (see appendix) for all fifteen variable (size, s1 to s14) separately by means of fuzzy logic[2,16]. Similarly four analysis are carried out.

2.1 The variables are standardized. Then the correlation structure is calculated. Table A1 shows Eigen value for each component along with percentage of the total variance explained by that component.[5,6,14,15]

Eigen value criterion: Retaining nine components with eigen values of up to at least 1.0.

Proportion of variance explained criterion: Components 'size' to 's8' account for a solid 98.562% of the variability, and adding next component gives 99.424% of the variability

Hence nine components are extracted. Table A2 shows component matrix for extracting nine components [1, 3]

2.2 Similar to size, effort factor is taken with s1-s14 to get Eigen and variance value Table B1. Components 'Effort' to 's7' account for a solid 99.58% of the variability and adding next component gives 100% of the variability.

Hence eight components are extracted. Table B2 shows component matrix for extracting eight components.

2.3 Similarly to size, duration factor is taken with s1-s14 to get Eigen and variance value Table C1. Components 'Duration' to 's7' account for a solid 99.59% of the variability and adding next component gives 100% of the variability. Hence eight components are extracted. Table C2 shows component matrix for extracting eight components.

2.4 Taking size, effort, duration, s1-s14 to get Eigen and variance value Table D1. Components 'Size' to 's5' account for a solid 99.62% of the variability and adding next component gives 100% of the variability. Hence eight components are extracted. Table D2 shows component matrix for extracting eight components.

3. CONCLUSION

- Four different cases are considered.
- Analysis is carried out for seventeen metric elements in four groups.
- PCA is done using a covariance matrix,
- Sum of the Eigen values represents the number of variables entered into the PCA,
- Some component's Eigen values in table A1, B1, C1 and D1 are very small and hence neglected.
- The analysis of variables is to identify the dimension that is latent.
- Nine components in A2, eight components in each B2,C2, and D2 tables are retained values.
- Only retained values are used for graphical plot and may be used for Scree plot.
- First three or four elements are more predominant components, remaining are of least effect.

4. REFERENCES

1. Conte, S.,Dunsmore, H.and Shen,V, Software Engineering Metrics and models, Benjamin Cummings, Menlo Park CA, 1986.
2. Guanrong Chen and etal. Introduction to fuzzy sets, fuzzy logic and fuzzy control systems. CRC Press, 2000.
3. E.M.Hall Managing risk: Methods for software system Development Addison-Wesley;1998.
4. Katrina, D.M. Applied statistics for software managers.Prentice Hall PTR, 2002.
5. Daniel T Larose Data Mining: methods and models. Wiley interscience 2006.
6. John D Musa Software Reliability Engineering McGraw Hill 1999.

7. R.Kelley Pace and Ronald Berry, sparse spatial autoregressions, statistics and probability letters, vol. 33No.3,pp 291-297, May 5, 1997 Data set available from stat lib,http://lib.stat. cmu.edu/datasets/houses.zip.
8. Richard A Jhonson and Dean Wichern, Applied multivariate statistical analysis, Prentice Hall, upper Saddle River , NJ,1998.
9. Kemerer.C.F. and S.Slaughter “An Empirical Approach to Studying SoftwareEvolution“, IEEE Transactions on Software Engineering, Volume 25, Number 4, July/August 1999, pp.493-509.
10. Kemeerer.C.F, and S.Slaughter, “Determinants of Software Maintenance Profiles: An Empirical Investigation,” Software Maintenance Research and Practice, Volume 9, 1997, pp.235-251.
11. Gerg Goth, “Software-as-a-service: The Spark That Will Change Software Engineering?” IEEE distribute systems, July 2008.
12. Tsun Chowe, Dac-Buu Cao A survey study of critical success factors in agile software projects Journal of system and software 81 (2008) 961-971.
13. Xuemei Zhang, Hoang Pham An analysis of factors affecting software reliability The journal of systems and software pp43-56, 200
14. Seetharam.k, Chandrakanth G Pujari, Ranking of Tools use, software logical complexity, Requirement volatility, Quality requirements, Efficiency requirements in software development IEEE Explorer Journal 2008 pp1608-1616.
15. Seetharam.K,Chandrakanth G Pujari, Evaluation of Installation requirements, staff analysis skills, Staff application knowledge, staff tool skill, & staff team skill in software development; cluster analysis http://bit.ly/dzScyk
16. U.S.Census,Bureau, Urban area criteria for census 2000, Federal Register,Vol 67,No 51,March 15 2002.

5. APPENDIX

Details of factors Details about variable:[4]

S1: customer participation: how actively customer took part in development work:

1 = very low; none

2 = low; passive; client defined or approved <30% of all functions

3 = normal; client defined and approved 30-70% of all functions

4 = high; active; client defined and approved all of most important functions, and over 70% of others

5 = very high; client participated very actively; Most functions were slightly volatile and require changes

S2: staff availability: availability of software personnel during project:

1 = very low; big problems with key personnel availability; lots of simultaneous customer and maintenance responsibilities; special know-hold required

2 = low; personnel involved in some other simultaneous projects and/or maintenance responsibilities

3 = normal; key members involve in only one other project

4 = high; project members involved almost full-time

5 = very high; qualified personnel available when needed full-time participation

S3: standards use: level and use if standards:

1 = very low; standards developed during project

2 = low; some standards, but not familiar ones; more must be developed for some tasks

3 = nominal; generally known standards applied in environment before; some tailoring needed

4 = high detailed standards applied in same environment for some time

5 = very high; stable and detailed to team; use controlled

S4: method use: level and use of methods: meeting; used by individuals

1 = very low;no modern design methods

2 = low; use beginning; traditional concepts employed; mostly (structural analysis & design, top-down design etc).

3 = nominal; generally known methods used

4 = High Methods integrated in detail and most activities are covered. Support Existing used by everyone.

5 = Very high methods used during entire life cycle.

S5: Tools use: levels and use of tools:

1.Very low; Minimal tools; editors, compilers, and testing tools.

2 Low basic tools: interpreters, editors, compilers, debuggers, databases, and libraries.

3 Nominal ; development environment , DBMS, and support for most phases

4 High modern tools like CASE, project planners, application generators, and standardized interfaces, between phases

5 Very high; integrated CASE environment over entire lifecycle; all tools support each other.

S6: Software Logical Complexity

Computing, I/O needs, and user interface requirements;

1 Very low; only routines; no need for user interface; simple databases.

2.Low ; functionally clear; no algorithmic tasks; database solution clear.

3 Nominal; Functionally typical; Normal, standard database; no algorithms

4. High; processing more demanding; database large and complex. New requirements for user interface.

5. Very high; functionally and technical difficult solution. User interface very complex. Distributed database.

S7:Requirements volatility

1 Very low; no new features, std components; conversions only.

2. Low ;some changes to specifications; some new or adapted functions; some minor changes in data contents.

3.Nominal; more hanges to specifications.< 15% new or modified functions.

4.High ; some major changes affecting total architecture.15-30% of functions new or modified.

5.Very high; new requirements added continuously; Lots of rework; more than 30% new or modified functions compared to original requirements

S8:Quality requirements: quality goals of software:

1.Very low; no quality requirements. quick and dirty allowed.
2.Low ;basic requirements satisfied.
3.Nominal:proper documentation of critical features. Design and implementation tested, walkthroughs; maintenance work planned.
4.High;formal reviews and inspections between all phases. Attention to usability.
5. Very high quantified and quality requirements.
S9: Efficiency requirements: Efficiency goals of software.
1.Very Low :No efficiency requirements needing attention or planning.
2.Low: Efficiency goals easy to reach: requirements below average.
3.Nominal: capacity level of software stable and predictable.
4.High:Specific peaks in capacity, response time, transaction process, and turn around time reached by specific design and implementation techniques.
5.Very high: efficiency essential: strict efficiency goals needing continuous attention and specific skills.
S10: Installation requirements
1=Very low; No training needs; < 10 users.
2= Low ; some training ; about 10 users; creation of basic data only minor
3=Nominal; typical training; 10-50 users. Some conversion of old data.
4=High ;large scale training for several organizations/platforms in ; <1000 users; extra software for conversions; possible parallel runs ;
5=Very high;> 1000 users; Long expected life time; several user organizations; several different platforms.
S11:staff analysis skills ; Analysis skills of project staff at kick off.
1 Very low; No experience in requirements Analysis or similar projects.
2.Low ; < 30 % of project staff with analysis and design experience in similar projects.
3 Nominal; 30-70%of project staff with analysis experience; one experienced member.
4. High; Most members of staff with experience in specification and analysis; Analysis professional in charge.
5.Very high; Project staff composed of first –class professionals. Members have strong vision and experience with requirements analysis.
S12: Staff application knowledge: Knowledge of application domain in project team.(supplier and customer)
1=very low: team application experience < 6 months on average.
2= low; application experience low; some members have experience ;6-12 months on average.
3=Nominal ;Application experience good;1-3 years on average.
4=High; Application experience good both at supplier and customer sites. 3-6 years on average.;Business dynamics known.
5=Very high; Both supplier and customer know application area well, including the business; > 6 years average experience.
S13= staff tool skill: Experience level of project team (supplier and customer) with development and documentation tools at project kick off;

1=Very low; Team has no experience in necessary tools; team's average experience < 6 months..
2=.Low; Tools experience < average; some members have experience with some tools; (6-12 months)
3=.Nominal: tools experience good in about half the team; some members know development and documentation tools well; 1-3 years on average.
4=High; most team members know tools well; some members can help others; 3-6 years on average.
5= Very high; team knows all tools well; support available for specific needs of project; >6 years average experience.
S14: staff team skill: Ability of project team to work effectively according to best project practices: 1=Very Low; scattered team; minimal project and management skills
2=Low; some members with previous experience on similar projects; not united as a group
3=Nominal: most members with experience on similar projects; commitment on project goals good; no motivation to utilize real team spirit
4=.High; group very active and knows how to exploit team effectiveness
5=Very high; very anticipatory team ; team can solve in an innovative way most personal and team problems; superior spirit.

Table A1	Eigen	%	cumulative
component	value	variance	%
size	5.374	33.172	33.1724
s1	3.288	20.296	53.4682
s2	2.4822	15.322	68.7904
s3	1.528	9.432	78.2224
s4	1.296	7.9999	86.2223
s5	0.7065	4.3613	90.5836
s6	0.5327	3.2883	93.8719
s7	0.5195	3.2066	97.0785
s8	0.2404	1.4837	98.5622
S9	0.1396	0.8616	99.4238

Table B1	Eigen	%	cumulative
component	value	variance	%
Effort	5.5778	37.19	37.1852
s1	3.5092	23.39	60.5796
s2	2.2317	14.88	75.4579
s3	1.5225	10.15	85.6076
s4	0.783	5.22	90.8273
s5	0.5534	3.689	94.5163
s6	0.4279	2.853	97.3691
s7	0.3318	2.212	99.5812
s8	0.0628	0.419	99.9999

Table C1	Eigen	%	cumulative
component	value	variance	%
Duration	5.8246	38.8306	38.8306
s1	3.2592	21.7278	60.5584
s2	2.2134	14.7558	75.3142
s3	1.5277	10.185	85.4992
s4	0.7414	4.9429	90.4421
s5	0.6325	4.2168	94.6589
s6	0.3961	2.641	97.2999
s7	0.3439	2.293	99.5929
s8	0.0611	0.4071	100

Table D1	Eigen	%	cumulative
Component	value	variance	%
Size	6.8566	40.3331	40.3331
Effort	3.6614	21.5376	61.8707

Duration	2.5001	14.7064	76.5771
s1	1.5572	9.1601	85.7372
s2	0.8481	4.9887	90.7259
s3	0.6862	4.0364	94.7623
s4	0.4749	2.7935	97.5558
s5	0.351	2.0647	99.6205
s6	0.0645	0.3796	100.0001

Table A2	size	s1	s2	s3	s4	s5	s6	s7	s8
size	-0.3134					0.3775			-0.306
s1		-0.32		0.38			-0.42		
s2		-0.47						-0.379	
s3	0.3165				0.4348			0.3276	
s4	0.3762			0.313					
s5			0.367	0.379					0.3687
s6		-0.43			0.3137		0.338	0.3041	0.3294
s7				0.352			0.3	-0.35	
s8	0.3175						-0.343		
s9	0.3252			0.405					
s10	0.3365					0.5019			
s11			0.345		-0.718	-0.384			
s12				-0.42					
s13			0.586						
s14	0.3455						0.508	0.4262	

Table B2	Effort	s1	s2	s3	s4	s5	s6	s7
Effort		0.36					-0.33	
s1			-0.5	0.4		0.493		
s2		0.46						0.483
s3					0.48		-0.54	
s4	0.371			0.31				
s5		-0.4		0.33				
s6			-0.4					-0.68
s7				0.38		0.372		
s8		0.33				0.36	0.32	
s9				0.43			-0.34	
s10	0.316				-0.56			
s11	0.363							
s12				-0.39	0.305			
s13			-0.5				-0.35	0.347
s14	0.335				-0.38	0.468		

Table C2	Duration	s1	s2	s3	s4	s5	s6	s7
Duration	-0.3363					0.342		
s1			0.4306	0.4002		0.433		
s2		-0.489						-0.45
s3							0.35	
s4	0.3633			0.306				
s5		0.3749		0.349				
s6		-0.3162	0.3656			-0.25		0.6
s7				0.3861		-0.33		
s8		-0.3177					0.42	
s9				0.42			-0.31	
s10	0.3143				0.543	0.336		
s11	0.3515							-0.35
s12		-0.3		-0.3953				
s13			0.5379				-0.37	
s14	0.3358				0.37	-0.33	-0.37	

Table D2	S	E	Duration	s1	s2	s3	s4	s5
S			0.3172					
E								
Duration	-0.3445					-0.31		
s1			0.3931	0.4425		-0.34		
s2		0.475						0.47
s3						-0.42	-0.5	
s4	0.3511							
s5		-0.317		0.3121				
s6			0.3176			0.326		-0.63
s7				0.4101				
s8		0.3597						
s9				0.3711	0.343			
s10				-0.3006	0.581			
s11	0.3082							
s12		0.3277		-0.3506	-0.329			
s13			0.5272					
s14						0.413	-0.34	