

Um Método de Extração de Valores Referência para Métricas de Softwares Orientados por Objetos

Tarcísio G. S. Filó¹, Mariza A. S. Bigonha¹, Kécia Aline Marques Ferreira²

¹Programa de Pós-Graduação em Ciências da Computação (PPGCC)
Departamento de Ciência da Computação - Universidade Federal de Minas
Gerais (UFMG) - Belo Horizonte - MG - Brasil

²Departamento de Computação - Centro Federal de Educação Tecnológica de Minas
Gerais (CEFET-MG) - Belo Horizonte - MG - Brasil

{tfilo,mariza}@dcc.ufmg.br, kecia@decom.cefetmg.br

Nível: Mestrado

Ano de ingresso no programa: 2012

Época prevista de conclusão: março de 2014

Data da aprovação da proposta de dissertação: março de 2013

Eventos Relacionados: SBES

Resumo. Valores referência para métricas de softwares orientados por objetos são úteis em diversos campos da Engenharia de Software, tais como processos de filtragem em estratégias de detecção de bad smells e identificação de medições anômalas em um processo de medição de produto, parte integrante de um programa de qualidade de software. Nesse trabalho apresentamos um método de extração de valores referência para métricas de softwares orientados por objetos, bem como mostramos a sua aplicação na identificação de valores referência para 17 métricas em um dataset de 111 softwares open-source. Os resultados preliminares sugerem que os valores referência propostos estão de acordo com os bons princípios de projetos orientados por objetos. Os próximos passos do trabalho consistem em avaliar empiricamente a eficácia dos valores referência na avaliação da qualidade do software.

Palavras-chave: métricas, valores referência, qualide de software, medição de software.

Abstract. Thresholds for object-oriented software metrics are helpful in severous fields of Software Engineering, such as filtering mechanisms in bad smells detection strategies and identification of anomalous measurements in a software quality program. In this work, we present a method to extract thresholds of object-oriented software metrics and show its application in the thresholds identifying thresholds for 17 metrics of an 111 open-source software systems dataset. Preliminary results suggest that the proposed thresholds are in accordance with well-established design principles. The next steps are related to the empirical evaluation of the thresholds effectiveness in software quality evaluation.

Keywords: metrics, thresholds, software quality, software measurement.

1. Introdução

As pesquisas relacionadas com valores referência para métricas de software orientado por objetos vem sendo realizadas a partir de diferentes metodologias de pesquisa e apresentam valores referência de formatos e significados distintos. Além disso, os valores referência de diversas métricas propostas na literatura ainda não são conhecidos. Uma consequência direta desse problema diz respeito à inexistência de condições adequadas para avaliar quantitativamente e de forma mais eficaz a qualidade dos softwares. Valores referência permitem responder questões simples como “Quais métodos do sistema possuem uma quantidade excessiva de parâmetros?” ou “Quais classes do sistema possuem muitos métodos?”. Portanto, conhecer esses valores é essencial para que as métricas de software possam ser usadas na solução desse problema.

Nesse contexto, o trabalho de pesquisa proposto tem por objetivo contribuir para a solução desse problema, catalogando e avaliando valores referência para um conjunto de 18 métricas de softwares orientados por objetos, a partir de um método sólido e embasado em dados estatísticos para derivação desses valores. Com valores referência catalogados, vislumbra-se que as métricas possam ser efetivamente aplicadas na indústria de software, e, conseqüentemente, a qualidade dos softwares possa ser avaliada de forma automática.

2. Solução Proposta

Nesta seção é descrita a metodologia a ser aplicada na derivação de valores referência para as métricas de softwares orientados por objetos coletadas no *Qualitas.class Corpus* [Terra et al. 2013], que é uma versão compilada do *Qualitas Corpus* proposto por Tempero *et al.* (2010), no qual são disponibilizados arquivos XML referentes aos 111 sistemas contendo medições de 18 métricas de softwares.

2.1. Preparação dos Dados

Neste trabalho, foi desenvolvida uma ferramenta que lê os arquivos XML disponibilizados no *Qualitas.class Corpus* e alimenta um banco de dados *Mysql* com o objetivo de estruturar e normalizar as medidas. Realizada a conversão por meio dessa ferramenta, têm-se uma forma rápida e ágil de agregar e sumarizar os dados.

2.2. Data-Fitting

Um problema importante em Estatística é obter informação sobre a distribuição seguida pelo conjunto de dados analisados [Gibbons and Chakraborti 2003]. Para esse propósito, a ferramenta chamada EasyFit (2013), por meio do teste de aderência de *Kolmogorov-Smirnov*, realiza a seleção da distribuição apropriada aos dados. Selecionar a distribuição apropriada aos dados é o procedimento de escolher uma distribuição que tenha melhor ajuste para um conjunto de dados. O objetivo dessa etapa da pesquisa é estabelecer a distribuição apropriada a cada métrica de software estudada. Segundo Baxter *et al.* (2006), explorar as distribuições dos valores das métricas de software é fundamental para avançar no entendimento das estruturas internas de softwares. A partir da distribuição é possível identificar e entender suas características, como por exemplo, se o valor médio é ou não representativo para a análise.

2.3. Análise dos Dados

EasyFit gera um histograma de probabilidade dos dados ajustados a distribuição. A partir dessa visualização gráfica e do conhecimento das características da distribuição mais aderente aos dados, é possível derivar valores referência para as métricas. Na abordagem proposta por Ferreira *et al.* (2012), quando a métrica apresenta uma distribuição que possui valor médio representativo, como por exemplo, a distribuição de *Poisson*, este valor representa o valor típico da métrica. Caso contrário, são identificadas três faixas de valores: *Bom/Frequente*, *Regular/Ocasional* e *Ruim/Raro*. A faixa *Bom/Frequente* corresponde a valores com alta frequência, caracterizando os valores mais comuns da métrica, na prática. Segundo Ferreira *et al.*, esses valores não expressam necessariamente as melhores práticas da Engenharia de Software, e sim um padrão seguido pela maioria dos softwares. A faixa *Ruim/Raro* corresponde a valores com baixa frequência, e a faixa *Regular/Ocasional* é intermediária, e correspondem a valores que não são nem muito frequentes nem raros. A frequência é estabelecida por meio de uma análise gráfica do histograma de probabilidade dos dados ajustados a distribuição. Neste trabalho, pretende-se estender o método de Ferreira *et al.*, tornando a identificação dos limiares mais precisa, reproduzível e automatizável.

3. Avaliação dos Resultados

Ferreira *et al.* (2012) avaliaram os valores referência propostos por meio de estudos de casos, considerando que a sua aplicação pode resultar em duas situações possíveis: o valor referência avalia corretamente ou não o método, a classe ou o pacote. A avaliação correta ocorre quando a faixa em que a métrica cai reflete a situação real do artefato avaliado. Nos estudos de caso, consideram-se duas situações em que não há a avaliação correta do valor referência: *falso positivo*, quando o valor referência gera uma classificação ruim e a inspeção qualitativa não identifica problemas estruturais, ou *falso negativo*, quando o valor não classifica a classe como ruim, mas a inspeção qualitativa identifica problemas em sua estrutura. A identificação de casos *falsos negativos* demanda uma avaliação qualitativa de um número muito grande de classes. Considerando que a derivação é feita de forma quantitativa, por meio da análise de distribuição das medidas coletadas de uma grande quantidade de sistemas, uma inspeção manual é inviável, dada a dimensão da amostra. Por essa razão, os casos *falsos positivos* foram avaliados por Ferreira *et al.* a partir da inspeção manual apenas das classes identificadas nas faixas *Ruim/Raro* estabelecidas para as métricas, com o objetivo de avaliar a eficácia do valor referência para identificar problemas. Em nosso trabalho propomos a condução de 4 estudos de casos com os seguintes objetivos:

- **Estudo de Caso 1:** avaliar artefatos de software classificado como *ruim* por meio dos valores referência, para verificar a presença de casos *falsos positivos*.
- **Estudo de Caso 2:** avaliar o software JHotDraw, de boa qualidade, de forma a verificar a sua classificação como *Bom/Frequente* por meio dos valores referência. Se isso se confirmar, acredita-se que a aplicação dos valores referência não levem a resultados *falsos negativos*.
- **Estudo de Caso 3:** avaliar um software proprietário modelado de forma anêmica, que não se aproveita dos recursos da orientação por objetos, e verificar se a avaliação dos valores referência propostos classificarão seus atributos de qualidade como *Ruim/Raro*. Projetos modelados de forma anêmica pregam que não se

deve colocar a lógica de negócio nos objetos de domínio, e sim em um conjunto de objetos de serviço [Fowler 2003].

- **Estudo de Caso 4:** avaliar um software proprietário sabidamente bem projetado, de forma a avaliar se os valores referência classificarão seus atributos de qualidade como *Bom/Frequente* ou *Regular/Ocasional*.

4. Estado Atual do Trabalho

Para a definição dos valores referência foi realizada uma revisão de literatura referente à métricas de softwares orientados por objetos e a trabalhos relacionados à extração de valores referência. Toda a parte de preparação dos dados (Seção 2.1), *data-fitting* (Seção 2.2) e análise dos dados (Seção 2.3) foi concluída. Dessa forma, já foram identificados os valores referência para 18 métricas de softwares orientados por objetos. A Tabela 1 sumariza 5 dos 18 valores referência propostos. Na Seção 4.1 exemplificamos o processo de derivação para a métrica número de métodos.

Table 1. Valores Referência Identificados.

Métrica	Bom/Frequente	Regular/Ocasional	Ruim/Raro
Acoplamento Aferente	$m \leq 7$	$7 < m \leq 39$	$m > 39$
Acoplamento Eferente	$m \leq 6$	$6 < m \leq 16$	$m > 16$
Número de Atributos	$m \leq 3$	$3 < m \leq 8$	$m > 8$
Número de Métodos	$m \leq 6$	$6 < m \leq 14$	$m > 14$
Complexidade de McCabe	$m \leq 2$	$2 < m \leq 4$	$m > 4$

4.1. Número de Métodos

O Gráfico de Frequência Relativa Acumulada, Figura 1(a), sugere uma distribuição de cauda-pesada. Isso significa que há um grande número de classes com poucos métodos e um pequeno número de classes com muitos métodos. Como mostrado na Figura 1(b), o conjunto de valores dessa métrica possui melhor ajuste à distribuição *Weibull*. A Figura 1(c) mostra esses dados em escala logarítmica, sugerindo uma lei de potência. Foram identificados os valores que representam o 70° e 90° percentil dos dados, que correspondem aos valores 6 e 14 (Tabela 1). Os valores 70% e 90% foram identificados por meio de uma análise visual no Gráfico de Frequência Relativa Acumulada da Figura 1(a). Os pontos marcados com a cor verde/quadrado e vermelha/losango representam as regiões identificadas nessa análise. Além disso, determinam a escolha desses percentis os conceitos das faixas *Bom/Frequente*, *Regular/Ocasional* e *Ruim/Raro*, que devem caracterizar, respectivamente, valores muito frequentes, que ocorrem ocasionalmente e raros.

5. Contribuições Pretendidas

As principais contribuições pretendidas, oriundas da pesquisa proposta, incluem:

1. Uma revisão bibliográfica sobre métricas de softwares orientados por objetos.
2. Revisão bibliográfica sobre valores referência para métricas de softwares OO.
3. Um catálogo de distribuições estatísticas que possuam melhor encaixe em relação as 18 métricas de softwares orientados por objetos, permitindo identificar características acerca de como os softwares estão sendo construídos.

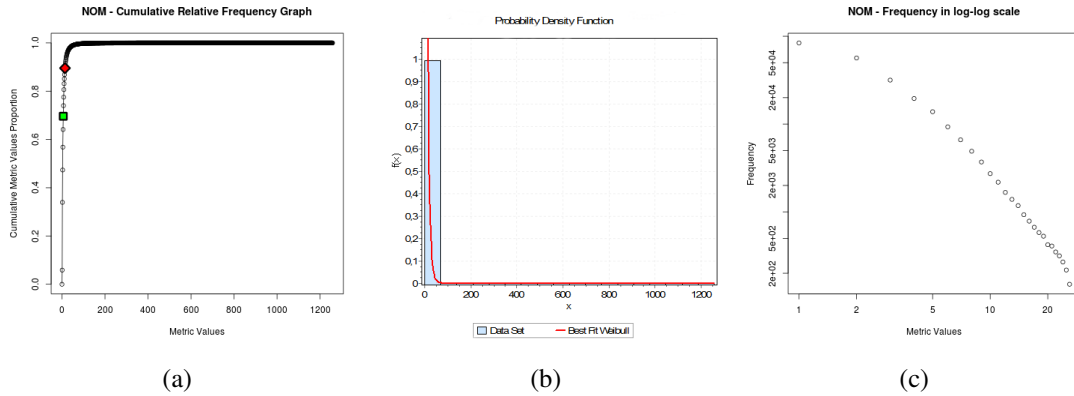


Figure 1. Número de Métodos: (a) Gráfico de Frequência Relativa Acumulada (b) Função de Densidade de Probabilidade ajustada à Distribuição Weibull (c) Histograma em escala log-log.

4. Extração de valores referência para 18 métricas de softwares orientados por objetos a partir de um *DataSet* de 111 softwares *open-source*.
5. Ampliação do universo de métricas com valores referência derivados por meio da metodologia estabelecida por Ferreira *et al.* (2012), permitindo realizar uma análise comparativa em termos de resultados com outros trabalhos.
6. Avaliação dos valores referência em softwares abertos.
7. Avaliação dos valores referência em softwares proprietários bem e mal projetados, estendendo a validação proposta a softwares fora do universo de código aberto.

6. Trabalhos Relacionados

Alves *et al.* (2010) propuseram um método empírico para determinar valores referências a partir de um conjunto de softwares. Nesse método, as medições coletadas são agrupadas e agregadas gerando informações no seguinte formato: o valor 17 da métrica *McCabe* representa 0,658% de todos os sistemas. Esses valores são ordenados e os valores referência são identificados pela escolha do 70%, 80% e 90% percentil do conjunto, caracterizando os seguintes perfis de qualidade: *low risk* (entre 0 - 70%), *moderate risk* (70 - 80%), *high risk* (80 - 90%) and *very high risk* (> 90%). Nesse trabalho, não se considera a distribuição apropriada aos dados na definição dos percentis que definem os perfis e não há uma avaliação empírica do modelo de risco proposto.

Ferreira *et al.* (2012) identificam valores referência para 6 métricas de softwares orientados por objetos por meio da medição das estruturas de um conjunto de 40 softwares de código aberto desenvolvidos em Java, de diferentes tamanhos, domínios e tipos, e uma análise estatística dos dados baseada em uma distribuição estatística apropriada. A análise utilizada nesse método é essencialmente visual, sendo de difícil reprodução. Além disso, foi aplicado a uma amostra pequena de softwares.

Oliveira *et al.* (2014) propuseram o conceito de valores referência relativos para avaliar métricas aderentes a uma distribuição de cauda-pesada. Os valores referência são relativos pois devem ser seguidos pela maioria das entidades, mas considera-se natural que um determinado número de entidades não siga os limites definidos. O formato proposto dos valores referência é o seguinte: $p\%$ das entidades devem ter $M \leq k$, aonde M é uma

métrica de código-fonte, k é o limite definido p é a porcentagem mínima. Neste trabalho, definimos valores referência absolutos. As abordagens não são exclusivas pois possuem diferentes cenários de utilização, que se complementam para os esforços correntes em aplicar as métricas de forma efetiva na indústria de software.

7. Conclusões

Essa proposta de dissertação de mestrado tem por objetivo a definição de um método para extração de valores referência para métricas de softwares orientados por objetos. O método proposto consiste na análise estatística da distribuição de um conjunto de medidas extraídas de um *dataset* de softwares, apresentando três faixas de valores: *Bom/Frequente*, *Regular/Ocasional* e *Ruim/Raro*. Nosso método não aplica técnicas estatísticas demasiadamente complexas, é reproduzível e o aplicamos a uma grande quantidade de softwares para extração dos valores referência propostos.

Este artigo relata o método proposto e os resultados preliminares do trabalho. Uma análise preliminar sugere que os valores referência indicados estão de acordo com os bons princípios de projetos orientados por objetos. Na continuidade do trabalho, pretendemos avaliar os valores referência por meio de estudos de caso, de forma a verificar se eles de fato auxiliam a identificação de problemas estruturais de software.

References

- Alves, T. L., Ypma, C., and Visser, J. (2010). Deriving metric thresholds from benchmark data. In *ICSM*, pages 1–10. IEEE Computer Society.
- Baxter, G., Freen, M., Noble, J., Rickerby, M., Smith, H., Visser, M., Melton, H., and Tempero, E. (2006). Understanding the shape of java software. In *OOPSLA '06: Proceedings of the 21st annual ACM SIGPLAN*, pages 397–412, NY, USA. ACM.
- EasyFit (2013). Easyfit. Disponível em: <http://www.mathwave.com/products/easyfit.html>. Acesso em Setembro de 2013.
- Ferreira, K. A., Bigonha, M. A., Bigonha, R. S., Mendes, L. F., and Almeida, H. C. (2012). Identifying thresholds for object-oriented software metrics. *Journal of Systems and Software*, 85(2):244–257.
- Fowler, M. (2003). Anemic domain model. Disponível em: <http://www.martinfowler.com/bliki/AnemicDomainModel.html>. Acesso em Setembro de 2013.
- Gibbons, J. D. and Chakraborti, S. (2003). *Nonparametric Statistical Inference (Statistics: a Series of Textbooks and Monographs)*. CRC, 4 edition.
- Oliveira, P., Valente, M. T., and Lima, F. P. (2014). Extracting relative thresholds for source code metrics. In *CSMR-WCRE, 2014 Software Evolution Week - IEEE Conference on*, pages 254–263.
- Tempero, E., Anslow, C., Dietrich, J., Han, T., Li, J., Lumpe, M., Melton, H., and Noble, J. (2010). Qualitas corpus: A curated collection of java code for empirical studies. In *2010 Asia Pacific Software Engineering Conference (APSEC2010)*, pages 336–345.
- Terra, R., Miranda, L. F., Valente, M. T., and Bigonha, R. S. (2013). Qualitas.class Corpus: A compiled version of the Qualitas Corpus. *Software Engineering Notes*, 38(5):1–4.