PCC170 - Projeto e Análise de Experimentos Computacionais

Marco Antonio M. Carvalho

Departamento de Computação
Instituto de Ciências Exatas e Biológicas
Universidade Federal de Ouro Preto

Conteúdo

Fonte

Este material é parcialmente baseado no conteúdo de

- Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G. C., and Stewart, W. R. (1995). *Designing and Reporting on Computational Experiments with Heuristic Methods*. Journal of Heuristics, 1:9–32.
- da Costa, C. R., & Longo, H. (2011). *Condução de Experimentos Computacionais com Métodos Heurísticos*.
- Rardin, R. L. and Uzsoy, R. (2001). *Experimental Evaluation of Heuristic Optimization Algorithms: A Tutorial*. Journal of Heuristics, 7(3):261–304.
- Weber, L.M., Saelens, W., Cannoodt, R. et al. Essential guidelines for computational method benchmarking. Genome Biol 20, 125 (2019).

Licença

Benchmark computacional

Definição

Estudos de benchmarking são realizados por pesquisadores de computação para comparar o desempenho de diferentes métodos, usando conjuntos de dados de referência e uma variedade de critérios de avaliação.

Benchmark computacional

Diretrizes

1. Defina o propósito e o escopo do *benchmark*.
2. Inclua todos os métodos relevantes.
3. Selecione (ou projete) conjuntos de **dados representativos**.
4. Escolha valores de parâmetros e versões de software apropriados.
5. Avalie os métodos de acordo com as principais métricas quantitativas de desempenho.
6. Avalie medidas secundárias, incluindo requisitos computacionais.
7. Interprete os resultados e forneça recomendações.
8. Publique os resultados em um formato acessível.
9. Projete o *benchmark* para permitir **futuras extensões**.
10. Siga as melhores práticas de pesquisa **reproduzível**.

Propósito e o escopo

A finalidade e o escopo de um *benchmark* devem ser claramente definidos no início do estudo e orientarão fundamentalmente o projeto e a implementação.

Em geral, podemos definir três tipos amplos de estudos de benchmarking:

1. Aqueles realizados por desenvolvedores de métodos, para demonstrar os méritos de sua abordagem;
2. Estudos neutros realizados para comparar métodos sistematicamente para uma determinada análise; ou
3. Aqueles organizados na forma de um desafio comunitário.

Benchmark computacional

Propósito e o escopo

Ao propormos um novo método, o foco do *benchmark* é avaliar os méritos relativos do novo método.

Isso pode ser alcançado, por exemplo, comparando com os métodos atuais de melhor desempenho (se conhecidos), um método simples como *baseline* e quaisquer métodos amplamente utilizados.

Benchmarks realizados para introduzir um novo método devem discutir o que o novo método oferece em comparação com o estado da arte atual, como descobertas que de outra forma não seriam possíveis.

Seleção de métodos

A seleção de métodos concorrentes deve garantir uma avaliação precisa e imparcial dos méritos relativos da nova abordagem, em comparação com o estado da arte atual.

Em áreas dinâmicas, os desenvolvedores de métodos devem estar preparados para atualizar seus *benchmarks* ou projetá-los para permitir extensões facilmente à medida que novos métodos surgem.

Seleção de dados

A seleção de conjuntos de dados de referência é uma escolha crítica de projeto.

Se conjuntos de dados acessíveis ao público adequados não puderem ser encontrados, eles precisarão ser gerados ou construídos.

Incluir uma variedade de conjuntos de dados garante que os métodos possam ser avaliados sob uma ampla gama de condições.

Há quatro tipos de instâncias de teste:

1. Instâncias reais;
2. Variações de instâncias reais;
3. Bibliotecas públicas de referência;
4. Instâncias geradas aleatoriamente.

Seleção de dados

Em geral, os melhores conjuntos de instâncias de testes são aqueles com instâncias reais.

Entretanto, pode ser difícil obtê-las, pois empresas ou instituições podem não tornar os dados públicos, ou só aceitam, caso dados importantes sejam omitidos.

As bibliotecas públicas de referência são muito utilizadas, e existem muitos repositórios de instâncias de teste.

Seleção de dados

Pesquisas feitas no início do ciclo de vida de um problema muitas vezes apresentam poucas instâncias.

Como a pesquisa continua, os conjuntos de instâncias de teste utilizados pelos pioneiros das pesquisas tendem a se tornar coleções de referências clássicas, utilizadas por todos os pesquisadores que trabalham sobre o mesmo problema.

Todavia, esses conjuntos podem apresentar algumas falhas graves.

Benchmark computacional

Seleção de dados

Pode ocorrer que as instâncias geradas e escolhidas sejam as que resultaram em bons resultados para os algoritmos desenvolvidos.

Isto pode resultar em instâncias com padrões ocultos, i.e., algum pesquisador construiu instâncias em que foram obtidas boas soluções para seus algoritmos, mas não se sabe se estas instâncias são realmente relevantes.

Benchmark computacional

Seleção de dados

Instâncias geradas artificialmente, de forma totalmente artificial e com propriedades controladas por parâmetros gerais, é a maneira mais fácil e rápida de se obter um conjunto de instâncias.

Isto permite a produção de diversas populações de instâncias, eventualmente englobando características que muitas vezes não são encontradas em instâncias reais.

Essa abordagem permite que sejam geradas instâncias em que uma solução ótima é conhecida, o que facilita a avaliação do desempenho das heurísticas.

Seleção de dados

Entretanto, estas instâncias podem gerar conclusões totalmente distorcidas em relação ao mundo real e muitas vezes são criticadas por retratarem situações irreais e serem mais fáceis ou difíceis de serem resolvidas que os problemas reais.

Se as instâncias são geradas pelo pesquisador, então o processo de geração deve ser claramente descrito, **cada decisão deve ser justificada** e, se possível, amarrada à literatura.

Instâncias geradas aleatoriamente sem um processo justificado e sem soluções ótimas conhecidas são **irrelevantes**.

Parametrização

As configurações de parâmetros podem ter um impacto crucial no desempenho.

Alguns métodos têm um grande número de parâmetros, e o ajuste de parâmetros para valores ideais pode exigir um esforço significativo.

Para um *benchmark* neutro, uma faixa de valores de parâmetros deve ser idealmente considerada para cada método, embora as compensações precisem ser consideradas em relação ao tempo disponível e aos recursos computacionais.

Parametrização

A seleção dos valores dos parâmetros deve obedecer ao princípio da neutralidade, ou seja, certos métodos não devem ser favorecidos em relação a outros por meio de ajustes de parâmetros mais extensos.

Os valores podem ser selecionados durante o trabalho exploratório inicial, no entanto, pode ser introduzido **viés** ao ajustar os parâmetros do novo método mais **extensivamente** (*overfitting*).

A estratégia de seleção de parâmetros deve ser discutida de forma transparente durante o relato dos resultados.

Mais detalhes na aula "Definição de parâmetros de métodos de otimização".

Indicadores de desempenho

A avaliação dos métodos dependerá de uma ou mais métricas de desempenho quantitativas e qualitativas.

É um grande desafio avaliar a qualidade das soluções encontradas por heurísticas, pois geralmente os problemas são NP-Difíceis, e muitas vezes para estes problemas, métodos exatos não produzem soluções de qualidade em tempo viável.

O **balizamento externo** é obrigatório e pode incluir: cálculo da solução exata para pequenas instâncias; uso de limites inferiores ou superiores; construção de instâncias a partir de valores ótimos conhecidos; estimativa estatística de valores ótimos conhecidos; e comparação dos melhores valores encontrados.

Mais detalhes na aula "Análise do desempenho de métodos heurísticos".

Benchmark computacional

Interpretação de dados

Os resultados devem ser claramente interpretados a partir da perspectiva do público-alvo.

Os resultados devem ser resumidos na forma de recomendações.

Uma classificação geral dos métodos, ou classificações separadas para vários critérios de avaliação, podem fornecer uma visão geral útil.

Interpretação de dados

Não havendo um "vencedor" claro em todos os indicadores, uma estratégia informativa é usar *rankings* para identificar um conjunto de métodos de alto desempenho e destacar os diferentes pontos fortes e compensações entre esses métodos.

Sem uma discussão transparente das limitações do benchmark, corre-se o risco de enganar os leitores; em casos extremos, isso pode até prejudicar o campo de pesquisa mais amplo, orientando os esforços de pesquisa em direções erradas.

Publicação de resultados

A estratégia de publicação e relato deve enfatizar a clareza e a acessibilidade.

As visualizações que resumem várias métricas de desempenho podem ser altamente informativas para o público.

Representações gráficas adicionais e complementares são uma maneira útil de envolver o leitor.

Para *benchmarks* extensos, o material suplementar *on-line* permite que os leitores explorem os dados interativamente e em maior profundidade, além de ajudar nas extensões.

Extensões

A extensão e a reprodução de um *benchmark* estão intimamente relacionadas.

A criação de repositórios públicos contendo código e dados permite que outros pesquisadores desenvolvam os resultados para incluir novos métodos ou conjuntos de dados, ou tentar diferentes configurações de parâmetros ou procedimentos de pré-processamento.

Além de dados brutos e código, é útil distribuir dados pré-processados e/ou de resultados, especialmente para *benchmarks* computacionalmente intensivos.

Reprodutibilidade

A alteração da versão de um compilador ou sistema operacional pode resultar em diferentes sequências de operações que influenciam no resultado final.

Portanto, quando é necessária a reprodução do experimento, isto não significa que é uma reprodução exata dos resultados.

Os experimentos serão reproduzíveis se os resultados dos dados originais obtidos são consistentes com o do novo experimento e apresentam as mesmas conclusões.

Benchmark computacional

Reprodutibilidade

A reprodutibilidade do código e das análises de dados foi reconhecida como um "padrão mínimo" de verificação de trabalhos publicados.

O acesso ao código e aos dados permitiu anteriormente que os desenvolvedores de métodos descobrissem possíveis erros em *benchmarks* publicados devido a subutilização dos métodos ou dados utilizados.

A descrição dos algoritmos, detalhes de implementação, do ambiente computacional, dos valores dos parâmetros e as versões de *software* utilizados devem ser claramente relatados para garantir a reprodutibilidade, ainda que aproximada.

Reprodutibilidade

O relato do modelo experimental inclui, além de sua descrição:

- A quantidade, descrição e geração das instâncias testadas;
- A quantidade de testes repetidos por instância;
- Quais e quantas sementes foram utilizadas;
- Critérios de parada;
- Os critérios para encontrar uma solução inicial;
- Os valores dos parâmetros.

Reprodutibilidade

A descrição do algoritmo é um dos requisitos mais importantes na reprodução de um experimento. Deve-se dar a descrição completa do algoritmo:

- Linguagem de programação;
- Versão do compilador e opções utilizadas;
- Técnicas de pré-processamento;
- Algoritmo para obtenção da solução inicial;
- Bibliotecas utilizadas.

Reprodutibilidade

Como o ambiente de teste, que é o computador, pode influenciar no desempenho do algoritmo, alguns itens devem ser documentados:

- Modelo e marca dos processadores;
- Quantidade, tipos e velocidades dos processadores;
- Tamanho e configuração das memórias RAM e cache;
- Sistema operacional e versão.

Reprodutibilidade

Alguns padrões, se utilizados, poderão tornar os experimentos irreproduzíveis:

Relatar somente o valor da solução: Torna o experimento irreproduzível em um sentido limitado, pois não fornece informações sobre a variabilidade dos resultados, e, portanto, sobre uma margem de erro.

Relatar somente a porcentagem sobre a melhor solução calculada: Não representa muito, devido ao fato de que se forem calculadas várias soluções, não há como afirmar quais são as melhores. Deve-se sempre fornecer a instância e o valor encontrado.

Reprodutibilidade

Relatar a porcentagem sobre uma estimativa da solução ótima esperada: Para instâncias geradas aleatoriamente, estes dados serão irreproduzíveis se as estimativas não forem definidas ou se o método de obtenção não for especificado.

Relatar a porcentagem excedente do limite inferior: É reproduzível somente se o limite inferior puder ser calculado facilmente ou se for possível de fazer um cálculo aproximado.

Benchmark computacional

Reprodutibilidade

Relatar o percentual de melhora de alguma heurística: É reproduzível somente se a heurística for completamente especificada ou tiver seu código e soluções disponibilizados.

Não disponibilizar instâncias: Sem os dados de entrada, não há extensão ou reprodução de experimentos.

Benchmark computacional

Algumas bibliotecas de instâncias

- OR Library;
- ESICUP;
- DEIS;
- CsPLib;
- SCOOP;
- Matrix Market;
- OPTSICOM;
- TSPLIB;
- TSP;
- DIMACS;

Benchmark computacional

Algumas bibliotecas de instâncias

- *Kaggle*;
- *Loggi Benchmark for Urban Deliveries*;
- *Network Repository*;
- The house of graphs;
- BPPLIB;
- Bin Packing;
- *ekhoda's list of Optimization Problem Libraries*;
- 2E-LRP;
- VRP-REP;
- VeRoLog.

Leitura recomendada

- Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G. C., and Stewart, W. R. (1995). *Designing and Reporting on Computational Experiments with Heuristic Methods*. Journal of Heuristics, 1:9–32.
- da Costa, C. R., & Longo, H. (2011). *Condução de Experimentos Computacionais com Métodos Heurísticos*.
- Rardin, R. L. and Uzsoy, R. (2001). *Experimental Evaluation of Heuristic Optimization Algorithms: A Tutorial*. Journal of Heuristics, 7(3):261–304.
- Weber, L.M., Saelens, W., Cannoodt, R. et al. Essential guidelines for computational method benchmarking. Genome Biol 20, 125 (2019).

Dúvidas?