



CEFET/RJ

Pesquisa em Computação



Eduardo Ogasawara
eduardo.ogasawara@cefet-rj.br
<https://eic.cefet-rj.br/~eogasawara>

Ciência e tecnologia

- Ciência busca conhecimento e explicações
 - Constrói teorias para explicar fatos observados
- Tecnologia aplica conhecimento para resolver problemas práticos
 - Seu foco não é explicar fenômenos naturais, mas sim desenvolver soluções que impactam o mundo
- Na Computação, ciência e tecnologia caminham juntas:
 - Trabalhos científicos explicam como e por que algo funciona
 - Trabalhos tecnológicos resultam em ferramentas, algoritmos e processos
- Exemplo:
 - Ciência: A conjectura do favo de mel
 - Tecnologia: Estrutura espacial hexagonal hierarquia do Uber

Discrete & Computational Geometry (2001) 25:1–22
DOI: 10.1007/s004540010001

Discrete & Computational
Geometry

The Honeycomb Conjecture

T. C. Hales
Department of Mathematics, University of Michigan
Ann Arbor, MI 48109, USA

Abstract. This article gives a proof of the classical honeycomb conjecture: any partition of the plane into regions of equal area has perimeter at least that of the regular hexagonal honeycomb tiling.



1. Introduction

Around 30 B.C., Marcus Terentius Varro, in his book on agriculture, wrote about the hexagonal form of the bee's honeycomb [2]. There were two competing theories of the hexagonal structure. One theory held that the hexagons better accommodated the bee's cell. The other theory, supported by the mathematicians of the day, was that the structure was explained by an isoperimetric property of the hexagonal honeycomb. Varro wrote, "Does not the cluster in the comb have six angles? ... The geometer proves that this hexagon inscribed in a circle figure encloses the greatest amount of space."

The origin of this problem is somewhat obscure. Varro was aware of it long before Pappus of Alexandria, who mentions it in his 10th book [2]. Much later, Brook Taylor (1686) and Zeno's paradox were used to prove Varro's claim (see [10]). However, only fragments of Zeno's work remain, and it is not known whether the honeycomb is discussed there.

The argument of Pappus is incomplete. In fact, it involves nothing more than a comparison of three suggestive cases. It was known to the Pythagoreans that only three regular



Tipos de Pesquisa em Computação

- A pesquisa pode ser classificada pela:
 - Fonte de dados
 - Primária
 - Secundária
 - Métodos de investigação
 - Quantitativa
 - Qualitativa
 - Analítica

Pesquisa primária vs. secundária

- Pesquisa Primária
 - Produz novos dados (experimentos, entrevistas, observações)
 - Metodologias: qualitativa, quantitativa, analítica
 - Exemplo:
 - Desenvolvimento de um novo algoritmo de aprendizado de máquina
 - Novo método de normalização de séries temporais
- Pesquisa Secundária
 - Sistematiza o conhecimento existente
 - Inclui revisões sistemáticas e meta-análises
 - ◆ Exemplo:
 - Revisão sistemática sobre técnicas de aprendizado de máquina
 - Revisão sistemática sobre métodos de normalização para séries temporais

Pesquisa quantitativa vs. qualitativa

- Pesquisa Quantitativa

- Trabalha com dados numéricos
- Utiliza métodos estatísticos
- Exemplo:
 - Avaliação da acurácia de um modelo de IA com métricas como precisão e recall

- Pesquisa Qualitativa

- Trabalha com dados textuais ou subjetivos
- Utiliza métodos descritivos e exploratórios
- ◆ Exemplo:
 - Entrevistas com programadores sobre a adoção de uma nova tecnologia

Pesquisa Analítica

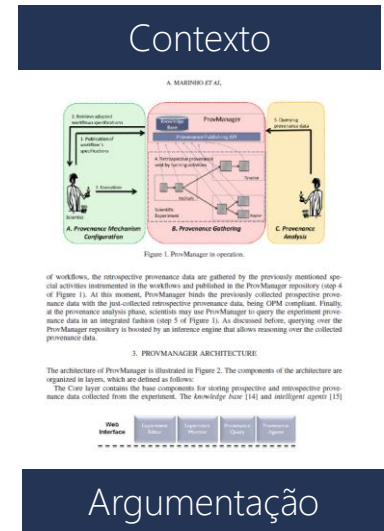
- Constrói teorias e apresenta provas matemáticas
- Comum em áreas como criptografia, algoritmos e lógica formal
- Exemplo:
 - Demonstração da complexidade do algoritmo de Dijkstra

Comparação e Validação de Contribuições Científicas em Computação

- Ao apresentar uma nova contribuição científica, é fundamental contextualizá-la e avaliá-la de forma adequada. Dependendo da natureza do trabalho, a validação pode seguir diferentes abordagens:
 - Apresentação de Algo Diferente: Quando não há trabalhos semelhantes, é necessário construir uma argumentação forte
 - Apresentação de Algo Melhor: Se há soluções existentes, a nova abordagem deve ser comparada com benchmarks reconhecidos
 - Apresentação de uma Prova: Em áreas teóricas, a validade se dá por meio de demonstrações matemáticas
 - Apresentação de um Artefato Computacional: Quando o foco é um software ou framework, a avaliação pode envolver testes práticos
 - Apresentação de Artigos de Dados: Trabalhos que organizam e disponibilizam datasets devem descrever claramente seu impacto
- Pergunta-chave para cada caso:
 - Como posso demonstrar o valor da minha pesquisa?

Apresentação de algo diferente

- Pesquisa que não tem trabalhos relacionados direto
- Avaliação qualitativa quando não há dados quantitativos disponíveis
- Estudos de caso são comuns, pois fornecem evidências qualitativas, mas não provas estatísticas
- Exemplo:
 - Desenvolvimento de um novo paradigma de programação sem comparação direta com modelos existentes



of workflows, the introspective provenance data are gathered by the previously mentioned agent activities instrumented in the workflow and published in the ProManager repository (step 4 of Figure 1). At this moment, ProManager finds the previously collected prospective provenance data with the just-collected introspective provenance data, being GPM compliant. Finally, at the provenance analysis phase, scientists may use ProManager to query the experiment provenance data in an integrated fashion (step 5 of Figure 1). As discussed before, querying over the ProManager repository is boosted by an inference engine that allows reasoning over the collected provenance data.

5. EVALUATION

ProManager was designed to support the provenance management of experiments in an integrated way, allowing all information that is collected individually for each system in a distributed and heterogeneous environment. In this paper, we evaluate if the scientist can take advantage of ProManager to enhance the provenance analysis of an experiment that is executed in a complex heterogeneous environment. With this objective in mind, we planned an observational study to assess the performance and to obtain the evaluation of scientists using two different provenance analysis approaches: decentralized approach, in which scientists use the provenance mechanisms available in existing SWMS where experiment workflow fragments are executed, and the ProManager approach or centralized approach, in which scientists rely on the provenance information collected by ProManager to make their analysis. The following subsections present the evaluation plan, its execution, and the obtained results.

5.1. Planning

A hypothetical experiment, named CHExp (Complex Heterogeneous Experiment), was defined to be performed in a heterogeneous environment. According to this scenario, the experiment is instrumented into two workflows, instrumented in two distinct SWMS: ViTrails and Kepler. The conceptual model of CHExp can be seen in the activity diagram of Figure 11. CHExp consists of

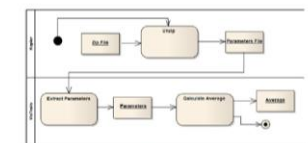


Figure 11. Diagram of CHExp.

Copyright © 2011 John Wiley & Sons, Ltd.
Concurrency Comput. Pract. Exper. (2011)
DOI: 10.1002/cpe

Apresentação de algo melhor

- Exige comparação com a literatura
- Pode usar benchmarks ou criar seus próprios testes
- Se não houver benchmarks disponíveis, o autor pode criar experimentos específicos, mas isso exige cuidado para evitar viés na avaliação
- Importante definir métricas claras
- Exemplo:
 - Um novo algoritmo de aprendizado de máquina testado em datasets padrão como MNIST e ImageNet

Comparação

Table 1. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using monthly aggregated data. The table shows the performance of different methods across various metrics.

Table 2. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using daily aggregated data. The table shows the performance of different methods across various metrics.

Table 3. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using hourly aggregated data. The table shows the performance of different methods across various metrics.

Table 4. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 15-minute aggregated data. The table shows the performance of different methods across various metrics.

Table 5. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 5-minute aggregated data. The table shows the performance of different methods across various metrics.

Table 6. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 1-minute aggregated data. The table shows the performance of different methods across various metrics.

Table 7. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 30-second aggregated data. The table shows the performance of different methods across various metrics.

Table 8. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 10-second aggregated data. The table shows the performance of different methods across various metrics.

Table 9. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 5-second aggregated data. The table shows the performance of different methods across various metrics.

Table 10. Comparison of temporal aggregation methods for predicting sea surface temperature (SST) using 1-second aggregated data. The table shows the performance of different methods across various metrics.

Apresentação de uma prova

- Construção de uma teoria baseada em definições formais
- Uso comum em áreas matematicamente rigorosas
- Atualmente, experimentação também é necessária para demonstrar aplicabilidade
- Exemplo:
 - Demonstração formal da convergência de um algoritmo de otimização

Formalização

optimum solution for the probabilities $\alpha_0, \alpha_1, \dots, \alpha_{i-1}$ and $\beta_1, \dots, \beta_{i-1}$, its right subtree is an optimum solution for the probabilities $\alpha_0, \dots, \alpha_i$ and $\beta_{i+1}, \dots, \beta_n$. Therefore, we can get a bottom up algorithm for building an optimal binary search tree for a set of probabilities $(\alpha_0, \dots, \alpha_n, \beta_1, \dots, \beta_n)$. We can build up optimal trees $T_{i,j}$ for all the probabilities $\alpha_0, \dots, \alpha_j$ and β_1, \dots, β_j where $i \leq j$ starting from the smallest intervals and working toward the largest.

Let $P_{i,j}$ and $W_{i,j}$ denote the weighted path length and the total weight of an optimal binary search tree for all words $K_i < X < K_{j+1}$ where $i < j$. Let $R_{i,j}$ denote the index of the root of this tree when $i < j$. The following formulae determine the cubic time algorithm:

$$P_{i,i} = W_{i,i} = \alpha_i \quad \text{for } 0 \leq i \leq n, \quad (1)$$

$$W_{i,j} = W_{i,j-1} + \beta_j + \alpha_j, \quad (2)$$

$$P_{i,j} = W_{i,j} + \min_{i \leq k < j} (P_{i,k-1} + P_{k,j}) \quad \text{for } 0 \leq i < j \leq n. \quad (3)$$

Since we choose $R_{i,j}$ from among $j - i$ pairs for each i, j such that $0 \leq i < j \leq n$ the algorithm runs in $O(n^3)$ time, as there are only $(n+1)(n+2)/2$ choices of $0 \leq i < j \leq n$, the space required being $O(n^2)$.

2.1.2. The monotonicity of roots and consequent $O(n^2)$ algorithm

Knuth [85] observed that the $R_{i,j}$'s satisfy the condition $R_{i,j-1} \leq R_{i,j} \leq R_{i+1,j}$. We will look at the proof in the next subsection. This condition means that we only have to search all the indices between $R_{i,j-1}$ and $R_{i+1,j}$ to compute $R_{i,j}$. The running time

Prova

Lemma 2.2. If w satisfies the quadrangle inequality and is monotone, then the function c defined above also satisfies the quadrangle inequality, i.e. $c(i,j) + c(i',j') \leq c(i,j') + c(i',j)$ for $i \leq i' \leq j \leq j'$.

Proof. We use induction on the length $l = j' - i$ to prove the result. This inequality is trivially true if $i = i'$ or $j = j'$. This proves the quadrangle inequality for c for $l \leq 1$. For the induction step we distinguish two cases $i' = j$, $i' < j$.

Case 1: $i < i' = j < j'$. In this case the quadrangle inequality for c reduces to

$$c(i,j) + c(j,j') \leq c(i,j') \quad (9)$$

Let $k = R(i,j')$. We distinguish two symmetric subcases: $k \leq j$, $k \geq j$.

Case 1.1: $k \leq j$. We have

$$\begin{aligned} c(i,j) + c(j,j') &\leq w(i,j) + c(i,k-1) + c(k,j) + c(j,j') \\ &\quad \text{(by definition of } c(i,j)) \end{aligned} \quad (10)$$

$$\begin{aligned} &\leq w(i,j') + c(i,k-1) + c(k,j) + c(j,j') \\ &\quad \text{(by monotonicity of } w) \end{aligned} \quad (11)$$

$$\begin{aligned} &\leq w(i,j') + c(i,k-1) + c(k,j') \\ &\quad \text{(by the induction hypothesis)} \end{aligned} \quad (12)$$

$$= c(i,j') \quad \text{(by definition of } c(i,j') \text{ and } k). \quad (13)$$

Case 1.2: $k \geq j$. As this case is symmetric to case 1.1 the proof is similar.

Case 2: $i < i' < j < j'$. Let $y = R(i',j)$ and $z = R(i,j')$. We have to distinguish two symmetric cases: $z \leq y$ or $z \geq y$. We only consider the case $z \leq y$. We note that $z \leq y < j$ by the definition of y and $i < z$ by the definition of z . We have

$$c(i',j') + c(i,j) \quad (14)$$

Apresentação de um artefato computacional

- O foco está na criação de um produto novo (software, ferramenta, framework)
 - Comum em TCCs e projetos aplicados
 - Pode não exigir rigor científico na apresentação dos resultados, mas deve demonstrar impacto prático
- Exemplo:
 - Desenvolvimento de um framework para detecção de anomalias em séries temporais

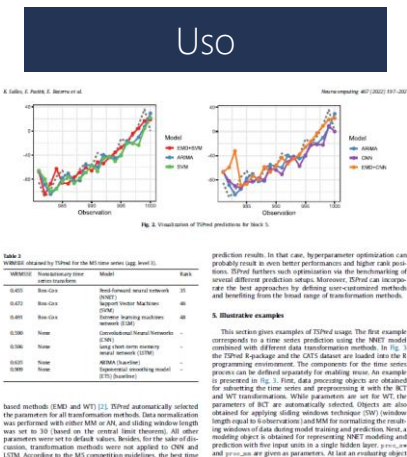


Table 2. Results obtained by TSPred for the 100 time series (log level 1).

Series	Series type	Model	Rank
0.001	Non-Stationary	Best forward neural network (NN)	25
0.001	Non-Stationary	Support Vector Machines (SVM)	48
0.001	Non-Stationary	Ensemble learning methods (ensemble)	48
0.001	Non-Stationary	Conventional neural networks (CNN)	-
0.001	Non-Stationary	Long short-term memory (LSTM)	-
0.001	Non-Stationary	ARIMA (ARIMA)	-
0.001	Non-Stationary	Exponential smoothing (ETS)	-

5. Illustrative examples

This section gives examples of TSPred usage. The first example corresponds to a time series prediction using the NNBT model combined with different data transformation methods. In Fig. 3, the TSPred framework and the GTS dataset are loaded into the R programming environment. The components for the time series prediction can be defined separately for training and testing. An example is presented in Fig. 3. First, data preprocessing objects are obtained by submitting the time series and preprocessing it with the RCT and NBT transformation. While parameters are set for the NBT, the parameters of RCT are automatically selected. Objects are also obtained for applying sliding window technique (SW) (window length equal to 10 observations) and LSTM for forecasting the resulting windows of data during model training and prediction. Next, a modeling object is obtained for representing NNBT modeling and prediction with five input series in a single hidden layer. $p=0.01$, $w=10$ and $g=10$ are given as parameters. At last an evaluating object

Apresentação de artigos de dados

- Documento revisado por pares que descreve um conjunto de dados
- Valoriza o esforço para organizar e descrever dados
- Exemplo:
 - Google's Open Images Dataset: um dataset publicado para treinamentos de modelos de visão computacional



Received: 2020-07-15; Accepted: 2020-12-15; Published: 2021-01-15

Page 1 of 1

Table 1 Overview of data description

Label	Name of data description	The type	Source ID
Data 1	Integrated dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 2	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 3	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 4	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 5	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 6	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 7	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 8	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 9	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222
Data 10	Healthcare dataset	dataset	https://www.kaggle.com/datasets/1122222222

Data description
The data description is a document that provides information about the data used in the study. It includes details about the data source, the data collection process, the data cleaning process, and the data analysis process. The data description is an essential part of the research paper, as it allows readers to understand the data used in the study and to evaluate the results of the study.

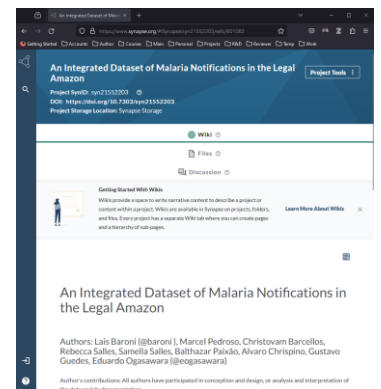
Data collection
The data collection process involves gathering data from various sources. In this study, data was collected from a variety of sources, including medical records, patient interviews, and public health data. The data collection process was designed to ensure that the data was accurate and complete.

Data cleaning
The data cleaning process involves removing any data that is missing, incorrect, or otherwise problematic. In this study, data was cleaned using a variety of techniques, including data imputation, data deletion, and data transformation. The data cleaning process was designed to ensure that the data was accurate and complete.

Data analysis
The data analysis process involves using statistical and other analytical techniques to examine the data. In this study, data was analyzed using a variety of techniques, including descriptive statistics, inferential statistics, and machine learning. The data analysis process was designed to identify patterns and relationships in the data.

Data visualization
The data visualization process involves creating visual representations of the data. In this study, data was visualized using a variety of techniques, including bar charts, line graphs, and scatter plots. The data visualization process was designed to make the data easier to understand and to highlight key findings.

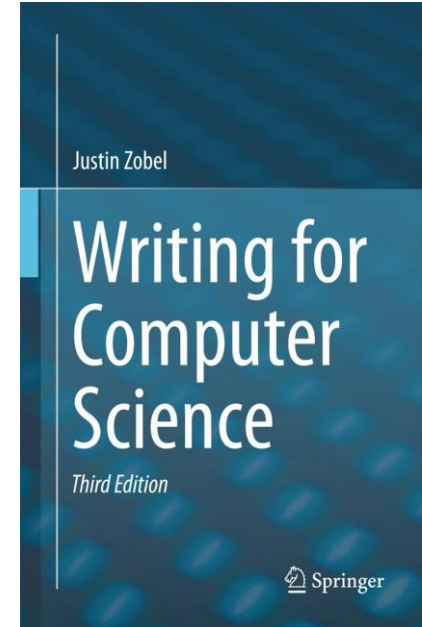
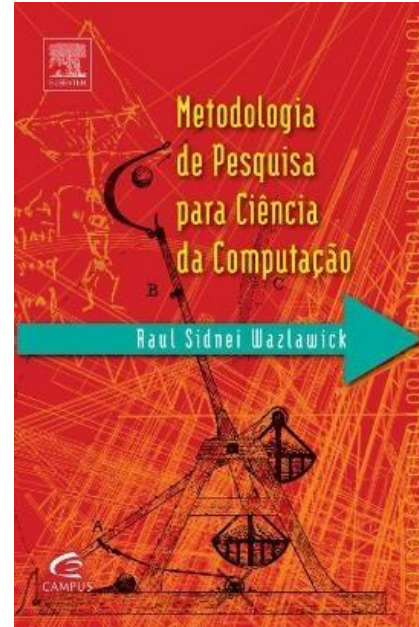
Data interpretation
The data interpretation process involves drawing conclusions from the data. In this study, data was interpreted using a variety of techniques, including statistical inference, machine learning, and expert judgment. The data interpretation process was designed to identify the most likely causes and effects of the data.



Considerações

- A pesquisa em Computação pode assumir diversas abordagens
- Escolher a metodologia correta depende do problema estudado
- Trabalhos podem focar na criação de conhecimento, na análise de dados, ou na prototipagem de soluções
 - Apesar do foco, pode apresentar em menor grau as outras abordagens
 - Exemplo:
 - Um artigo pode combinar aspectos de pesquisa analítica (prova formal), quantitativa (avaliação experimental) e tecnológica (implementação de ferramenta)

Referências



- [1] D. G. Perovano, Manual de metodologia da pesquisa científica. Editora Intersaberes, 2016.
- [2] A. L. Cervo, P. A. Bervian, e R. da Silva, Metodologia Científica. Pearson Universidades, 2006.
- [3] R. Wazlawick, 2017, Metodologia de Pesquisa para Ciência da Computação. Elsevier Brasil.
- [4] J. Zobel, 2015, Writing for Computer Science. Springer.

