Explorando Dados: Ferramentas e Métodos para Apoiar a Aprendizagem na Construção de Visualizações de Dados

Cassia de Oliveira Fernandez[1,2],

Roseli de Deus Lopes[2] (orientadora), Paulo Blikstein[3] (coorientador)

[1]Insper, Instituto de Ensino e Pesquisa – São Paulo, SP – Brasil

[2]Escola Politécnica – Universidade de São Paulo (USP)
São Paulo, SP – Brasil

[3]Teachers College, Columbia University – New York, NY – EUA

cassia.fernandez@usp.br, roseli.lopes@poli.usp.br, pb@tc.columbia.edu

Resumo. *Este trabalho apresenta contribuições na intersecção entre Ciência da Computação, Educação e Visualização de Dados, com foco no letramento em visualização de dados de estudantes da Educação Básica. São apresentados: (1) um framework para análise e design de ferramentas educacionais de visualização de dados, inspirado em taxonomias da Visualização da Informação; (2) o PlayData, ambiente de programação em blocos open-source para construção de visualizações; (3) método automatizado baseado em aprendizado de máquina para descrição de trajetórias de programação via dados de log coletados no ambiente PlayData; e (4) análise das principais ações (DVP moves) realizadas por aprendizes ao interagir com a ferramenta, com padrões obtidos via Epistemic Network Analysis (ENA).*

Abstract. *This work presents contributions at the intersection of Computer Science, Education, and Data Visualization, focusing on K-12 students' data visualization literacy. We present: (1) a framework for the analysis and design of educational tools for data visualization; (2) PlayData, an open-source block-based programming environment for constructing data visualizations; (3) an automated machine learning method for characterizing programming trajectories from log data; and (4) an analysis of the main DVP (Data Visualization Programming) moves learners engage in, with associated patterns obtained through Epistemic Network Analysis (ENA). Findings illuminate how K-12 students develop computational and data visualization competencies in block-based programming environments.*

1. Introdução e Caracterização do Problema

À medida que grandes volumes de dados tornam-se acessíveis e são utilizados para tomada de decisões, a educação em ciência de dados emerge como um foco crítico na Educação Básica. O letramento em visualização de dados (*visualization literacy*) é um componente central dessa área [Bach et al., 2023; Börner et al., 2019]. Contudo, a maioria dos materiais didáticos concentra-se na leitura de visualizações criadas por outros, e não na sua construção, e estudos indicam que mesmo estudantes de graduação de áreas STEM enfrentam dificuldades na interpretação de visualizações comuns [Maltese et al., 2015].

A maioria das ferramentas educacionais oferece tipos canônicos de visualização baseados em *templates*. Embora facilitem a criação rápida, essas ferramentas obscurecem o processo fundamental de mapeamento de dados para formas representacionais e limitam as possibilidades criativas [Rubin, 2020]. Ferramentas de programação permitem inventar novas formas de visualizar dados e desenvolver competências meta-representacionais [diSessa e Sherin, 2000]; porém, sem suporte de domínio específico, tornam-se complexas demais para uso educacional.

A contribuição central desta tese para a Ciência da Computação reside no desenvolvimento de (1) uma ferramenta de programação baseada em blocos para análise e visualização de dados, e (2) métodos analíticos baseados em aprendizado de máquina para capturar, automatizar e interpretar trajetórias de programação de estudantes, contribuindo para a área de Learning Analytics aplicada à educação em ciência de dados.

2. Objetivos e Contribuições

A tese teve como objetivo avançar no design de ambientes de aprendizagem para aprimorar o letramento em visualização de dados de estudantes da Educação Básica. Quatro questões de pesquisa foram investigadas:

(QP1) Como a literatura de Visualização da Informação pode informar o design de ferramentas educacionais?

(QP2) Como os aprendizes constroem visualizações com o PlayData e quais são as principais ações (*moves*) envolvidas nesse processo?

(QP3) Que padrões podem ser observados nas trajetórias dos estudantes?

(QP4) Quais são os desafios e potencialidades de aprendizagem com o PlayData?

As contribuições originais da tese incluem:

(1) Um framework para análise e design de ferramentas educacionais de visualização de dados, estruturado em quatro dimensões: objetivos suportados, expressividade, abstração e transparência para mapeamentos de dados.

(2) O PlayData, ambiente de programação em blocos *open-source* para construção de visualizações de dados, baseada no Scratch.

(3) Um método automatizado para coleta, análise e representação de trajetórias de programação com base em dados de log granulares coletados no ambiente de programação.

(4) A definição e caracterização de seis ações centrais (DVP moves) com padrões temporais obtidos via clustering K-means e ENA.

3. Trabalhos Relacionados

Ferramentas como CODAP [Finzer, 2013] e Google Spreadsheets oferecem interfaces baseadas em templates que facilitam visualizações canônicas, mas limitam a expressividade e a transparência. Ferramentas de programação geral como o Scratch permitem alta expressividade, porém exigem programar todos os elementos do zero. Nossa revisão de programação em blocos voltadas à visualização de dados revelou que nenhum ambiente combinava alta expressividade com transparência nos mapeamentos de

dados especificamente para uso educacional. Exemplos das ferramentas estudadas são apresentados na Figura 1.

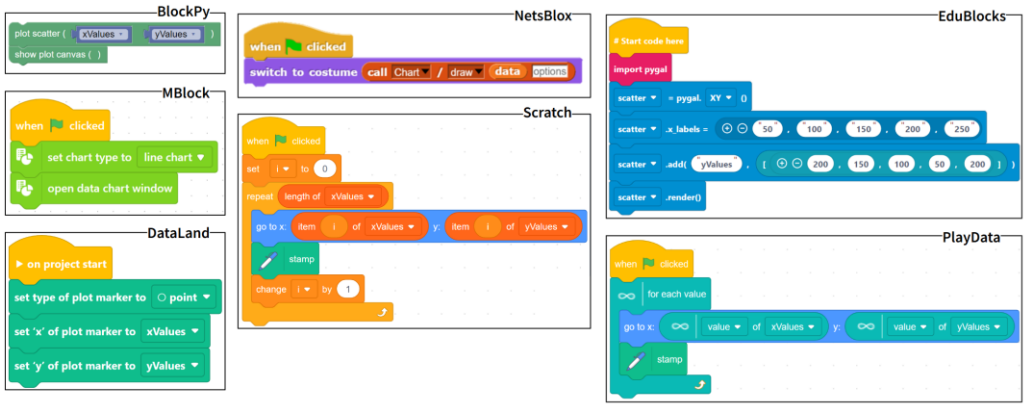

Figura 1. Comparação do código para criar gráficos de dispersão em diferentes ambientes de programação por blocos (BlockPy, NetsBlox, EduBlocks, MBlock, Scratch, DataLand e PlayData).

Na área de análise de trajetórias de programação, trabalhos anteriores como os de Huang e Parker [2023] exploram técnicas de *learning analytics* para caracterizar processos de programação, mas focam em métricas do produto final. Nossa abordagem, baseada em janelas temporais de eventos de log, captura a natureza iterativa e processual das tarefas de programação voltadas para visualização de dados, representando uma inovação metodológica relevante.

4. Metodologia de Pesquisa

4.1. Desenvolvimento da ferramenta PlayData

O desenvolvimento da ferramenta adotou a metodologia de Design-Based Research (DBR) [Barab e Squire, 2004], com ciclos iterativos de design, implementação, análise e redesign (Figura 2). Cinco estudos principais foram conduzidos entre 2021 e 2023:

- Estudo 1, 2021: Sessões individuais online com 4 estudantes dos Anos Finais do Ensino Fundamental (EFII);
- Estudo 2, 2021: Parceria com professores para implementação de uma unidade sobre mudanças climáticas no EFII;
- Estudo 3, 2022: Sessões em duplas com 4 estudantes do Ensino Médio;
- Estudo 4, 2022: Implementação em escola com 16 estudantes do EFII;
- Estudo 5, 2023: Workshop de 3 dias com 7 estudantes de 12–13 anos.

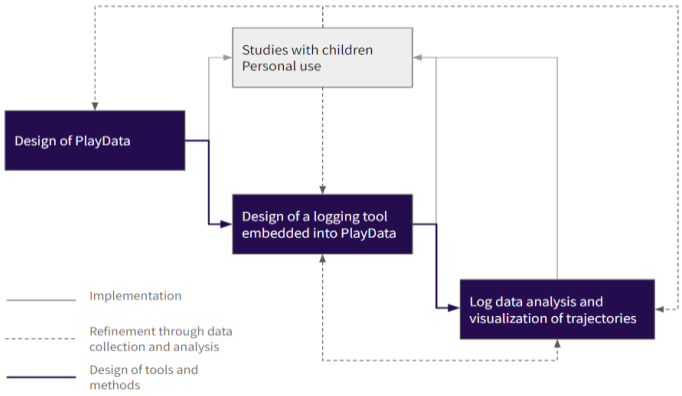

Figura 2. Processo iterativo de design das ferramentas e métodos desenvolvidos durante a pesquisa, seguindo a abordagem DBR.

4.2. Desenvolvimento do método para caracterização de trajetórias de programação

Foi adotada uma abordagem multi-método que combina análise qualitativa, técnicas de machine learning e análise de redes epistêmicas (*Epistemic Network Analysis, ENA*) para investigar como aprendizes constroem visualizações de dados utilizando o ambiente PlayData. Os dados coletados incluem: (a) logs de granularidade fina do PlayData, registrando cada interação com os blocos; (b) gravações de vídeo e tela; e (c) artefatos finais criados pelos estudantes.

A análise dividiu-se em quatro etapas principais:

1. **Análise Qualitativa de Vídeo:** A pesquisa começou com a microanálise de **18** horas de gravações de vídeo e capturas de tela de estudantes trabalhando em duplas. A partir dessa observação detalhada, foram identificadas indutivamente as seis categorias principais de ações (DVP moves): criação de programa, rearranjo de programa, seleção de parâmetros, inspeção de dados, inspeção de saída e design visual.
2. **Análise de Agrupamento (*Clustering*) para Automatização:** Para escalar a análise, o estudo utilizou os **logs granulares** capturados pelo ambiente PlayData, que registrou 64 tipos de eventos distintos (como adicionar blocos ou alterar parâmetros). Os logs foram segmentados em janelas de 1 minuto e processados utilizando o algoritmo de *clustering k-means*. Esse processo permitiu categorizar automaticamente cada minuto de atividade dos alunos em um dos seis DVP moves, atingindo um índice de concordância de 0,77 em relação à codificação manual humana para uma base amostral.
3. **Análise de Redes Epistêmicas Ordenadas (ONA):** Com os dados categorizados, os pesquisadores aplicaram a técnica de Ordered Network Analysis (ONA), uma evolução da Epistemic Network Analysis (ENA). Essa técnica foi utilizada para mapear as conexões cronológicas entre os eventos de baixo nível (como conectar blocos ou executar o programa) dentro de cada ação, revelando a dinâmica e a estrutura lógica das práticas de programação, análise de dados e comunicação visual dos estudantes.
4. **Análise Agregada e Temporal:** Por fim, os métodos permitiram uma visão macro das trajetórias de aprendizagem, incluindo como a distribuição dessas ações evolui ao longo do tempo e como os padrões de interação mudam do início

ao fim de uma única sessão de programação, identificando, por exemplo, a transição de desafios técnicos para decisões de design comunicativo.

Essa combinação de métodos permitiu que o trabalho emulasse a profundidade da análise de vídeo qualitativa em um corpus de dados muito maior, oferecendo uma fundação para compreender de forma granular processos de aprendizagem em tarefas de programação abertas.

5. Resultados

5.1. Framework e PlayData

O framework estruturado em quatro dimensões foi aplicado a sete ferramentas, revelando que ambientes de programação por blocos existentes concentravam-se em quadrantes de baixa expressividade ou alta abstração. O PlayData preenche o quadrante de alta expressividade com transparência, com abstração intermediária obtida via a implementação de novos blocos que apoiam a análise e visualização de dados no ambiente. O PlayData está publicamente disponível em playdatalab.github.io.

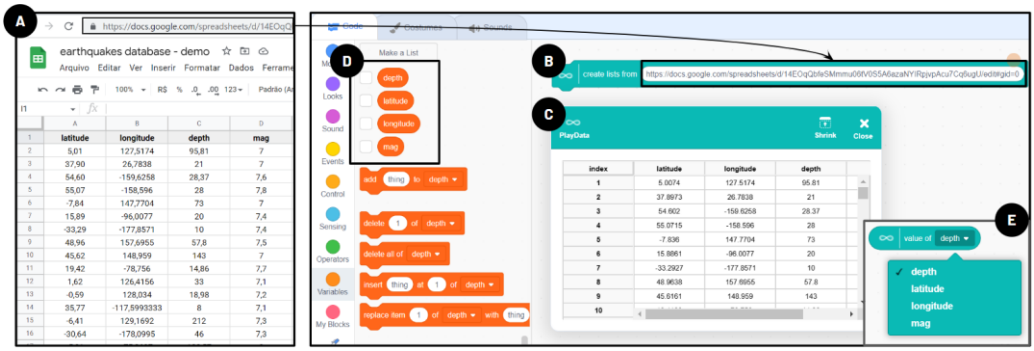

Figura 3. Interface do PlayData com importação de dataset do Google Sheets (A), bloco de importação (B), tabela embutida (C), blocos-lista gerados automaticamente (D) e menu suspenso com colunas do dataset (E).

O PlayData incorpora blocos especializados para: iteração sobre listas de dados (‘for each value’), mapeamento de valores a propriedades visuais (posição, cor, tamanho), transformações de dados (média, ordenação, filtragem) e importação de datasets do Google Sheets com geração automática de blocos-lista (Figura 3). O sistema de log registra cada interação com granularidade de evento. Exemplos de visualizações criadas com o PlayData são apresentadas na Figura 4.

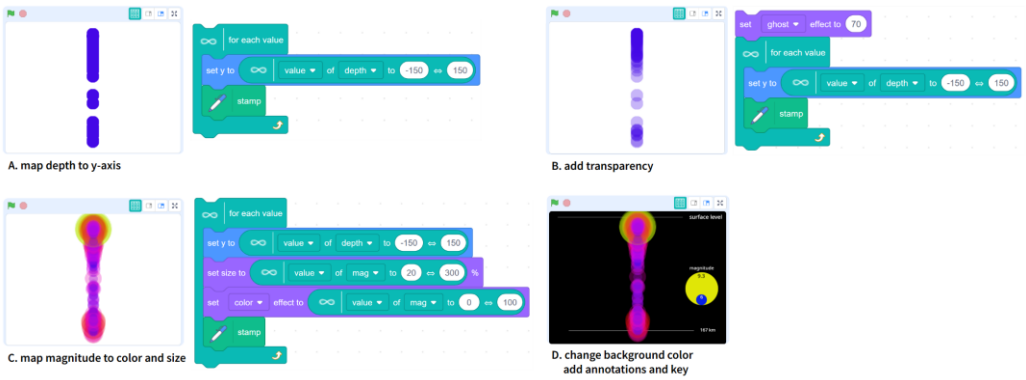

Figura 4. Construção de uma visualização de dados sísmicos com PlayData: mapeamento de profundidade ao eixo Y (A), transparência (B), magnitude a cor e tamanho (C) e anotações ao background (D).

5.2. DVP Moves

A análise combinada de vídeo e dados de log identificou seis DVP moves, cada um com assinatura distinta de eventos nas redes ENA: *Program Creation* (criação do código inicial; dominante no início das sessões); *Program Rearrangement* (associado com depuração; 21% do tempo no Dia 1, reduzindo a 3%); *Data Inspection* (análise dos dados na tabela); *Output Inspection* (observação do resultado visual por mais de 10s); *Selection of Parameters* (ajuste iterativo de parâmetros; dominante na fase intermediária); *Visual Design* (adição de títulos, legendas e anotações; dominante na fase final). Para cada ação, foi construída uma rede baseada em ONA para representar como os eventos que a compõem se relacionam entre si (Figura 5). Além disso, com base nessa categorização foram construídas trajetórias para cada uma das sessões de programação analisadas (Figura 6).

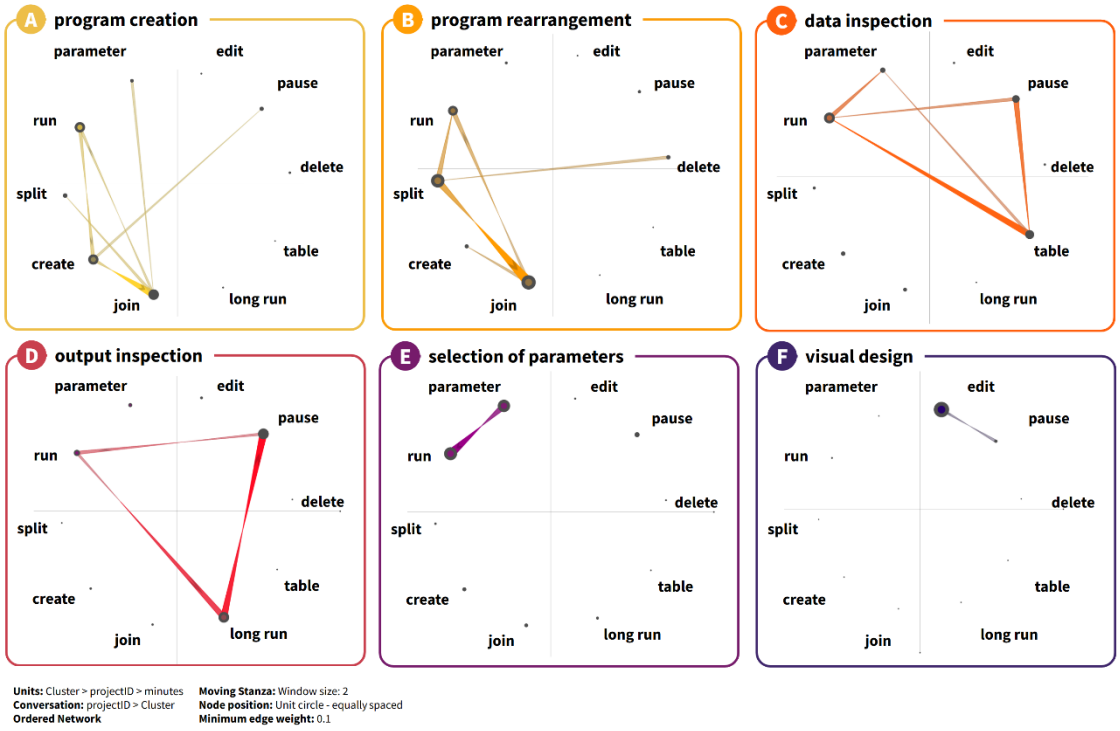

Figura 5. ONAs para cada ação (DVP move) identificada.

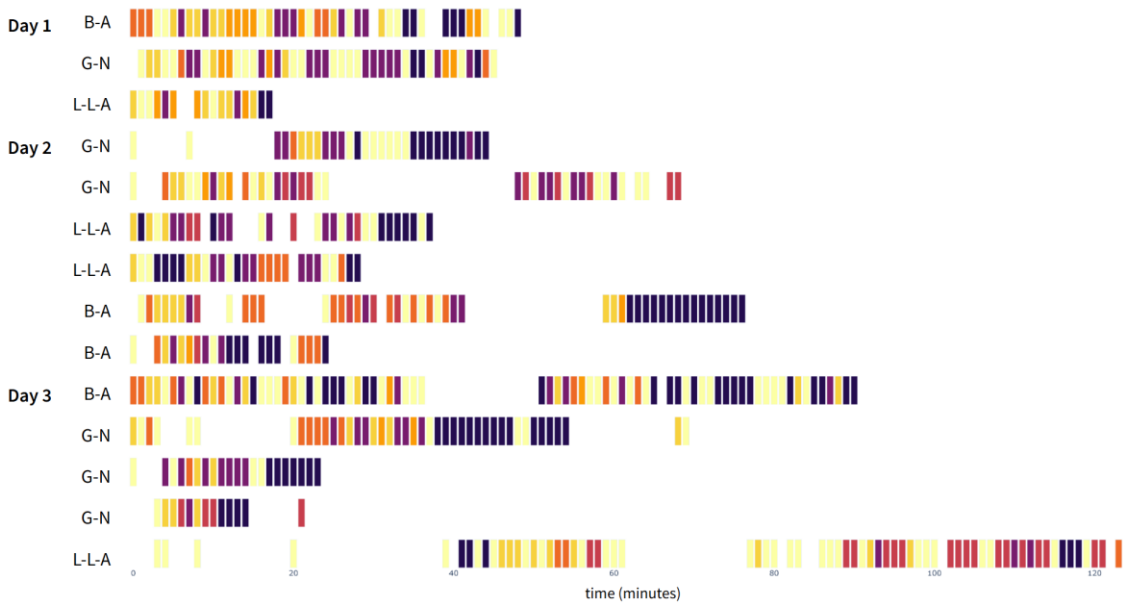

Figura 6. Representação temporal das trajetórias com base na análise de clusters. Espaços vazios representam momentos de inatividade. As letras na esquerda representam as iniciais dos estudantes envolvidos na sessão de programação.

5.3. Padrões nas Trajetórias

A análise temporal revelou padrões consistentes a respeito das trajetórias de aprendizagem dos estudantes: início dominado por inspeção de dados e criação de código; fase intermediária por seleção de parâmetros (processo altamente iterativo); e fase final por design visual (Figura 7). Ao longo do workshop de 3 dias, o tempo de debug reduziu de 21% para 3%, enquanto design visual e inspeção de output aumentaram significativamente.

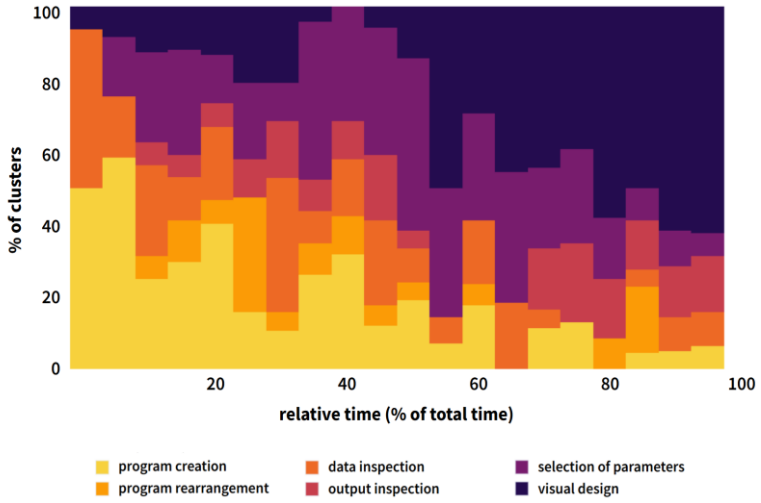

Figura 7. Distribuição dos DVP moves ao longo do tempo relativo das sessões (% do tempo total). A transição de program creation/data inspection (início) para selection of parameters (meio) e visual design (fim) é claramente visível.

6. Contribuição para a Ciência da Computação

Do ponto de vista do desenvolvimento de sistemas, o PlayData é um sistema de software original open-source que estende o Scratch com primitivas computacionais especializadas, incluindo módulo de importação de dados em tempo real (via integração com Google Sheets), bem como um sistema de logging granular de eventos.

Do ponto de vista de aprendizado de máquina e Educational Data Mining, a metodologia de caracterização automatizada de trajetórias de programação constitui uma inovação metodológica: representa trajetórias como sequências de vetores de contagem de eventos em janelas temporais, aplica K-means para codificação automática de ações (com obtenção de resultados compatíveis com análises manuais qualitativas) e usa ENA para descrição de padrões em tais ações. A abordagem é generalizável a outros domínios com ambientes de programação que suportem logging granular.

Do ponto de vista de HCI e design de sistemas educacionais, o framework desenvolvido contribui com vocabulário preciso para comparar ferramentas em dimensões computacionalmente relevantes. A identificação dos DVP moves caracteriza as práticas computacionais envolvidas em tarefas de visualização de dados com programação, com implicações para currículos e ferramentas de ciência de dados para a Educação Básica.

7. Impacto e Perspectivas Futuras

O PlayData foi utilizado em workshops com estudantes e professores no Brasil, EUA, Itália e Japão, e está disponível publicamente como ferramenta *open-source* traduzida para o português, inglês, espanhol e japonês. A metodologia de learning analytics é aplicável a qualquer ambiente de programação com logging granular, enquanto os *DVP moves* fornecem vocabulário teórico para avaliação formativa automatizada do engajamento em tarefas de ciência de dados.

Nossas perspectivas futuras incluem: (1) realização de estudos em maior escala; (2) desenvolvimento de *dashboards* para professores baseados nos DVP moves; (3) exploração de algoritmos para segmentação dinâmica de trajetórias (*versus* janelas fixas); e (4) extensão da ferramenta PlayData para acessibilidade e colaboração.

8. Conclusão

A tese apresentou contribuições originais para a Ciência da Computação na intersecção com a Educação em Computação e Ciência de Dados. O framework desenvolvido fornece uma estrutura conceitual para a análise e o *design* de ferramentas educacionais voltadas à visualização de dados. O PlayData, por sua vez, propõe uma forma de equilibrar expressividade, transparência e abstração no *design* de tais ferramentas para estudantes da Educação Básica.

A metodologia de *learning analytics* desenvolvida, que combina logs granulares, K-means e ENA com análises qualitativas aprofundadas, oferece uma contribuição técnica inovadora para análise automatizada de trajetórias de programação. Os DVP moves fornecem um modelo descritivo do processo de construção de visualizações, revelando como os estudantes engajam em práticas autênticas de *design* via programação. Os resultados demonstram que a programação por blocos voltada à visualização de dados é um caminho promissor para o desenvolvimento simultâneo de competências

computacionais e de letramento em dados, e que a visualização de trajetórias temporais pode apoiar educadores e pesquisadores na compreensão dos processos envolvidos em atividades abertas de programação.

Referências

Bach, B. et al. (2023) “Challenges and opportunities in data visualization education”, IEEE Transactions on Visualization and Computer Graphics, v. 29, n. 1, p. 700–710.

Barab, S. e Squire, K. (2004) “Design-based research: Putting a stake in the ground”, Journal of the Learning Sciences, v. 13, n. 1, p. 1–14.

Börner, K. et al. (2019) “A multi-level typology of abstract visualization tasks”, IEEE TVCG, v. 25, n. 1, p. 1197–1202.

Brennan, K. e Resnick, M. (2012) “New frameworks for studying CT development”, Proceedings of the Annual Meeting of AERA.

diSessa, A. A. e Sherin, B. L. (2000) “Meta-representation: An introduction”, Journal of Mathematical Behavior, v. 19, n. 4, p. 385–398.

Finzer, W. (2013) “The data science education dilemma”, Technology Innovations in Statistics Education, v. 7, n. 2.

Glazer, N. (2011) “Challenges with graph interpretation”, Studies in Science Education, v. 47, n. 2, p. 183–210.

Huang, R. e Parker, M. C. (2023) “Automated characterization of CT practices in block-based programming”, Proceedings of ICER 2023.

Maltese, A. V. et al. (2015) “What is (or should be) scientific literacy?”, Journal of Research in Science Teaching, v. 52, n. 7.

Méndez, G. et al. (2017) “Bottom-up vs. top-down strategies in data visualization design”, IEEE CG&A, v. 37, n. 4, p. 12–18.

Parsons, P. (2022) “Data visualization as a design discipline”, Design Studies, v. 79, 101084.

Rubin, A. (2020) “Learning to reason from samples”, Educational Studies in Mathematics, v. 103, n. 2, p. 201–222.

Shaffer, D. W. et al. (2016) “Epistemic network analysis: A worked example of theory-based learning analytics”. In: Handbook of Learning Analytics. SoLAR.

Wise, A. F. (2020) “Educating data scientists for a new generation of data”, Journal of the Learning Sciences, v. 29, n. 1, p. 165–181.