Presentation: Learning Behavior and Cognitive Bias from Web Logs

Rashmi Jayathirtha Rao, Christopher Stewart, Arnulfo Perez and Siva Meenakshi Renganathan, The Ohio State University

I. Introduction

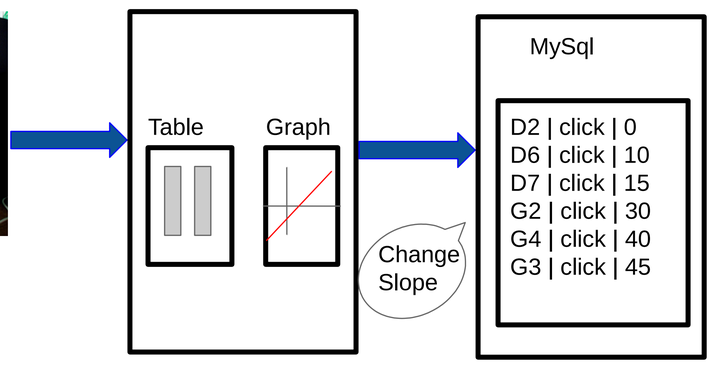
- An approach to model and analyze learning behaviour of students in a graph based e-learning tool.
- Derived models to classify the students and identify their computational thinking levels as well as cognitive preference towards either graph or data based learning.

Students who can link algebraic functions to their corresponding graphs perform well in STEM courses and is an indicator of computational thinking.

It is challenging to assess whether students are linking the concepts.

Traditional methods to analyse computational thinking- Tests, video analyses, interviews, surveys.etc

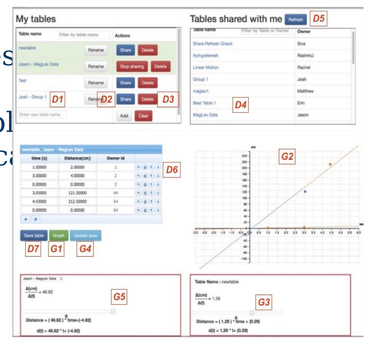
- We use web logs collected by a smart classroom web server to infer learning behavior
- We collect the clickstream, i.e, click and mouse movement data as users interact with the web portal
- Research* indicates that there are more than 67% correlation between eye movement and mouse movement.



II. Design and Setup

The portal

- Moodle- login
- Perform activities
- Collect the data
- Enter data in tabl
- Visualize graphica



Currently used by Grove City Schools

Published in FiE 2017

Over 1400 students have used the portal since 2016

Learning Analytics:

- Learning Analytics: the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environments in which it occurs[1]
- Major steps in LA process[1]:
  - Data Collection; Analytics and action; Post processing
  - The challenge is to define the right Objective / Indicator / Metric (OIM) triple before starting the LA2
  - Objective: Understand computational thinking/learning behaviour in terms of linking concepts of graph and data.
  - Indicator: Actions with the web components
  - Metric: Results from traditional analysis methods-participation levels

Our Data

- Source #1: Summer institute with 18 Math Instructors
- Source #2: Grove City students; 15 classes; 365 students

III. Feature Engineering

Web Resources: Web resource is a facility provided in a web application. To do meaningful analysis divide the web elements into data components and graph components this group.

Label	Component Name	Associated Behaviour
D1	My Tables - Table Name	Data
D2	My Tables - Share	Data
D3	My Tables - Delete	Data
D4	Tables Shared with me-Table name	Data
D5	Tables Shared with me-Refresh	Data
D6	Input Table	Data
D7	Save Table	Data
G1	Graph Button	Graphical
G2	Graph - SVG	Graphical
G3	Slider- Primary Table	Graphical
G4	Update Axes	Graphical
G5	Slider- Secondary Table	Graphical

Predictive features:

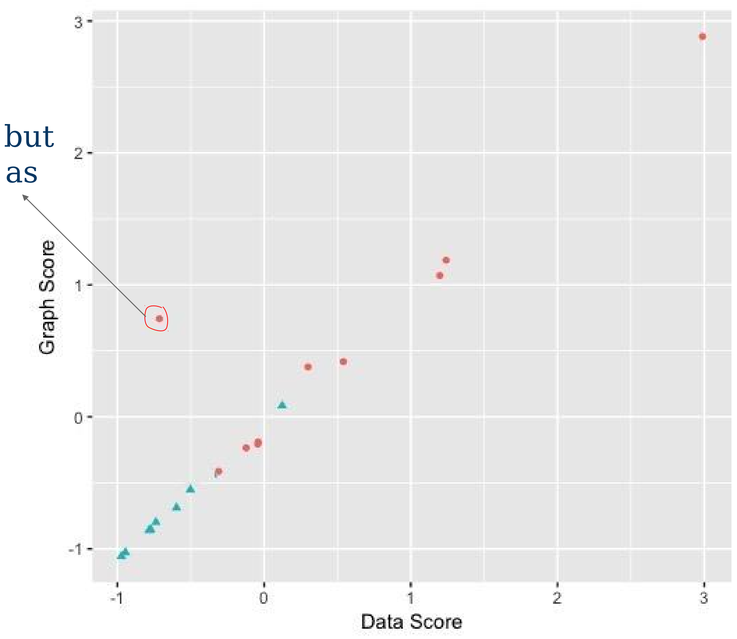
- Inter data interactions
- Inter graph interactions
- Intra data interactions
- Intra graph interactions
- Intra data time spent
- Intra graph time spent
- Data count in patterns
- Graph count in patterns

Trying to predict participation levels each student as “high” or “less” participating

IV. Early Results

High: graph score+data score >=-0.8
Less: graph score+data score <-0.8
17 out of 18 users classified accurately

Low data but classified as “high”



Results on classroom data

✔ Found classes with many computational thinking oriented students

✔ Identified students with possible cognitive bias
- 77 students out of 365

Mean graph score for workshop data	SD in graph score for workshop data	Mean graph score for classroom data	SD in graph score for classroom data
0.1861	1.751	0.1769	1.819

Mean data Score for workshop data	SD in data score for workshop data	Mean data score for classroom data	SD in data score for classroom
0.2733	1.8724	0.3016	1.7957

VI. References

[1] J. Wing, “A definition of computational thinking,” 2016.
[2] S. I. Swaid, “Bringing computational thinking to stem education,” in 6th International Conference on Applied Human Factors and Ergonomics and the Affiliated Conferences, 2015.
[3] S. Bagley and J. M. Rabin, “Students’ use of computational thinking in linear algebra,” November 2015.
[4] Arnulfo Perez, Kathy Malone, Siva Meenakshi Renganathan,Kimberly Groshong , “Computer modeling and programming in algebra,” 8th International Conference on Computer Supported Education, 2010.
[5] R. Rao, “Modeling learning behaviour and cognitive bias from web logs,” Master’s thesis, The Ohio State University, http://library.ohio-state.edu/record=b8245815 ~ S7, 2017
[6] J. Kelley, C. Stewart, N. Morris, D. Tiwari, Y. He, and S. Elnikety. Measuring and managing answer quality for online data-intensive services. In IEEE Conference on Autonomic Computing, 2015.
[7] Siva Renganathan, Christopher Stewart, Arnulfo Perez, Rashmi Rao and Bailey Braaten. Preliminary Results on an Interactive Learning Tool for Early Algebra Education. In IEEE Frontiers in Education, 2017
[8] N. Morris, S. M. Renganathan, C. Stewart, R. Birke, and L. Chen. Sprint ability: How well does your software exploit bursts in processing capacity? In IEEE Conference on Autonomic Computing, 2016.
[9] R. B. Myerson. Refinements of the nash equilibrium concept. International journal of game theory, 7(2), 1978.
[10] J. L. Sanchez-Lopez, R. A. S. Fern´andez, H. Bavle, C. Sampedro,M. Molina, J. Pestana, and P. Campoy. Aerostack: An architecture and open-source software framework for aerial robotics. In International Conference on Unmanned Aircraft Systems, 2016.
[11] Stackoverflow.com. Mobile developer surveys, 2013–2018.
[12] C. Stewart, M. Leventi, and K. Shen. Empirical examination of a collaborative web application. In IISWC, 2008.
[13]Stewart, Christopher, Aniket Chakrabarti, and Rean Griffith. "Zoolander: Efficiently Meeting Very Strict, Low-Latency SLOs." In ICAC, vol. 13, pp. 265-277. 2013.
[14] C. Stewart and K. Shen. Performance modeling and system management for multi-component online services. In USENIX Symposium on Networked Systems Design and Implementation, May 2005.
[15] S. Tomforde, S. Rudolph, K. L. Bellman, and R. P. W´urtz. An organic computing perspective on self-improving system interweaving at runtime. In IEEE Conference on Autonomic Computing, 2016.
[16] Z. Wang, X. Liu, A. Zhang, C. Stewart, X. Zhu, and T. Kelly. Autoparam: Automated control of application-level performance in virtualized server environments. In FeBid, 2007.
[17] D. Weyns and M. U. Iftikhar. Model-based simulation at runtime for self-adaptive systems. In IEEE Conference on Autonomic Computing, 2016.
[18] Z. Xu, N. Deng, C. Stewart, and X. Wang. Cadre: Carbon-aware data replication for geo-diverse services. In IEEE Conference on Autonomic Computing, 2015.
[19] C. Stewart, T. Kelly, and A. Zhang. Exploiting nonstationarity for performance prediction. In Proceedings of the 2nd ACM SIGOPS/EuroSys European Conference on Computer Systems, 2007
[20] C. Stewart, A. Chakrabarti, and R. Griffith, “Zoolander: Efficiently meeting very strict, low-latency SLOs,” in International Conference on Autonomic Computing, 2013.
[21] J. Kelley and C. Stewart and N. Morris and D. Tiwari and Y. He and S. Elnikety, “Obtaining and managing answer quality for online data intensive services,” in ACM Transactions on Modeling and Performance Evaluation of Computing Systems, 2017.
[22] N. Deng, C. Stewart, D. Gmach, and M. Arlitt. Policy and mechanism for carbon-aware cloud applications. In Proceedings of Network Operations and Management Symposium (NOMS’12). IEEE, 2012.
[23] M.-P. Hosseini, A. Hajisami, and D. Pompili, “Real-time epileptic seizure detection from eeg signals via random subspace ensemble learning,’’ in Proc. IEEE Int. Conf. Auto. Comput. (ICAC), Jul. 2016, pp. 209–218.
[24] C. Krupitzer, F. M. Roth, C. Becker, M. Weckesser, M. Lochau, and A. Schurr. Fesas ide: An integrated development environment for autonomic computing. In Autonomic Computing (ICAC), 2016 IEEE International Conference on, pages 15–24. IEEE, 2016.
[25] B. H. Cheng, R. De Lemos, H. Giese, P. Inverardi, J. Magee, J. Andersson, B. Becker, N. Bencomo, Y. Brun, B. Cukic, et al. Software engineering for self-adaptive systems: A research roadmap. In Software engineering for self-adaptive systems, pages 1–26. Springer, 2009.