CS Education Research at Virginia Tech

Cliff Shaffer

Department of Computer Science

Virginia Tech

Goals of the Talk

1. Introduce ongoing activities in the Department
2. Describe what we do, what is difficult, and how it is valuable
3. To relate some past successes and failures

CS@VT Role in Digital Education

- A Major Player in the Field
  - Competing with MIT, CMU, Perdue, GaTech, Duke
- SIGCSE’09
  - Largest departmental contingent?
  - 7 CS faculty
  - About that many students
  - About that many CS Alumni (faculty)

A Sampler of VT Work

- CITIDEL/Ensemble (part of NSDL) – Ed Fox
- Collections
  - ETD/Syllabus Repository/AlgoViz Wiki
- Web-CAT – Steve Edwards
- Cyber Arts – Steve Harrison, Yong Cao
- Middle School Math – Deborah Tatar
- Algorithm Visualization – Cliff Shaffer
- Visual Debugging – Godmar Back

Questions to Consider

- What are the (some) Goals?
- Is it scientific research?
- Is it Computer Science?

Goals

- Improve education
  - Course (and courseware) development
  - Improve understanding
  - Improve proficiencies (programming)

Is it Scientific Research?

- Does it have Measurable Effects?
- Is it Reproducible?
- Is it Novel?

Does it Have Measurable Effects?

- We want results to be quantifiable
- Performance scores:
  - Absolute: Difference between pretest and posttest
  - Relative: Performance gains in various treatments
- No Significant Difference
- Hawthorne Effect

Is it Reproducible?

- Isolate confounding influences
  - Instructor
  - Environment
  - Multiple treatments intermingled

Is it Novel?

- Computers are still relatively new in education
- Even in CS, we have only “recently” had the opportunity to use computers for education in ways other than as targets for programming exercises
- Which leads to the next issue…

Is it Computer Science?

- Some scenarios
  - What if I were a Chemistry Professor?
  - Instructional Technology Professor?
  - Engineering Education Professor?
  - CS Professor?
- In each case, the work is clearly of service to the discipline
- Note: ACM and IEEE both have Transactions in Education journals

Courseware Development

- Courseware development work is interdisciplinary:
  - Domain content (CS in our case)
  - Education/Instructional Technology
  - Human Factors/HCI
  - Software Engineering

Improving Data Structures

- Problem: Improve the retention/success rate in CS2606/CS3114
    - Key feature of the course is difficult projects
    - So, I focus on improving success rate in projects
- Interventions
    - Pairs programming
    - Project management
    - Increase student/Instructor interactions

Pairs Programming

- CS2606, 2007, 2 sections, no control
- Assigned partners, switch each project. Self-selected at end (but partners generally required)
- 3-4 week projects (4 total), fairly difficult
- “eXtreme Programming” style interactions encouraged

Pairs Programming: Outcomes

- Result: No difference detected in success rates or other outcome from prior semesters
- Somewhat contradictory of prior literature.
- Appears not to help or hurt students, in general. What about individuals?
- Mixed reception by students
- Hypothesis: Some benefit, some don’t
  - Used “free choice” in future sections
  - No differences detected

Scheduling

- Managing large-scale projects involves scheduling activities
  - It is human nature to work better toward intermediate milestones.
- The same concepts can/should be applied to mid-sized projects encountered in class.
  - For any project that needs more than a week of active work to complete, break into parts and design a schedule with milestones and deliverables.

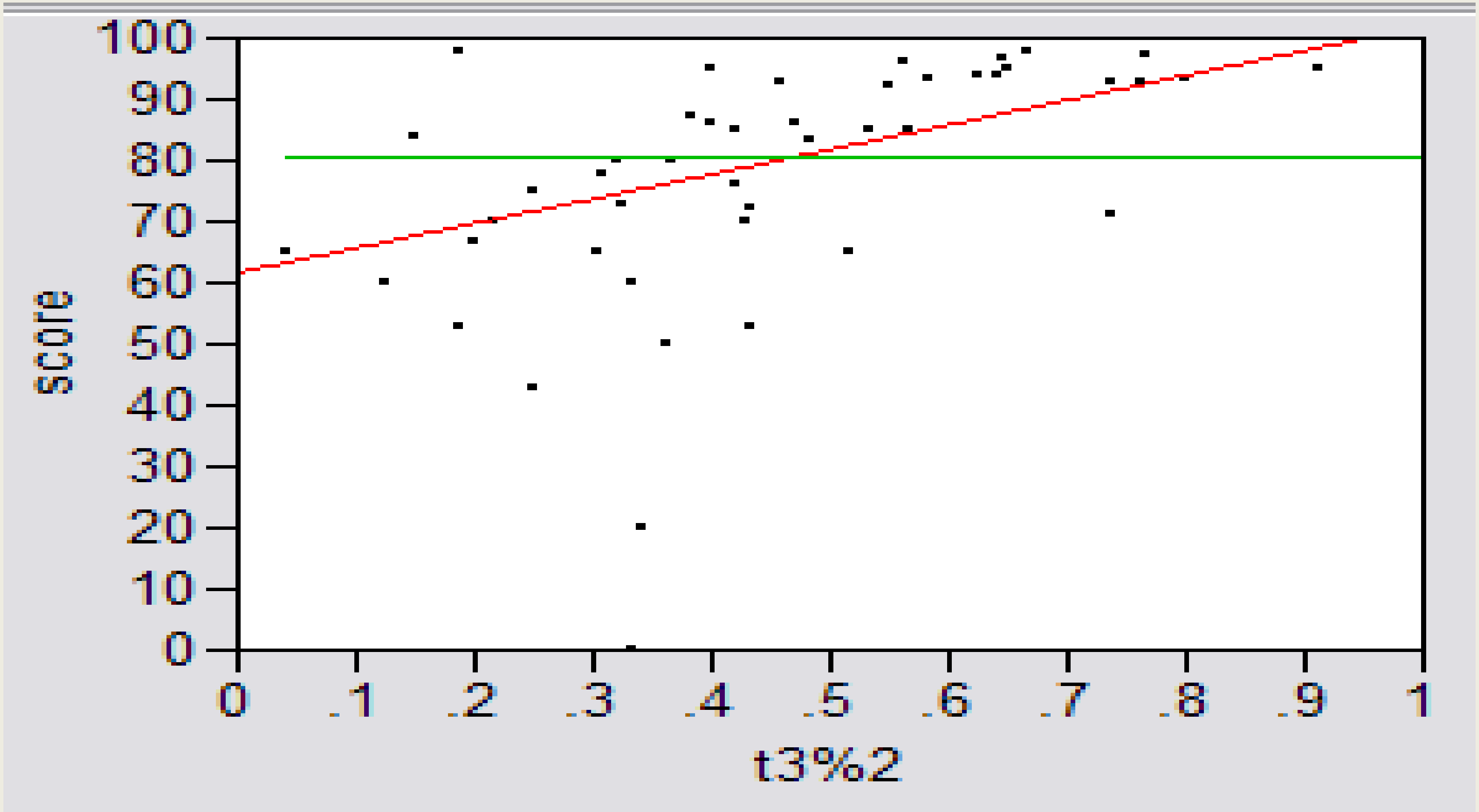
Scheduling

- CS2606, CS3114, 2007-2009 several sections, no control
- Require students to plan interim due dates, predict times required, weekly reports of time spent
- Mixed reaction from students. Some anecdotal evidence of appreciation afterward
- No recognizable change in outcomes

Real Results #1

- CS2606, Fall 2006
- 3-4 week projects
- Kept schedule information:
  - Estimated time required
  - Milestones, estimated times for each
  - Weekly estimates of time spent.

Real Results #2



Real Results #3

- Results were significant:
  - 90% of scores below median were students who did less than 50% of the project prior to the last week.
  - Few did poorly who put in > 50% time early
  - Some did well who didn’t put in >50% time early, but most who did well put in the early time

Real Results #4

- Correlations:
  - Strong correlation between early time and high score
  - No correlation between time spent and score
  - No correlation between % early time and total time

What is the Mechanism?

- Correlations are not causal
  - Do they behave that way because they are good, or does behaving that way make them good?
- Spreading projects over time allows the “sleep on it” heuristic to operate
- Avoiding the “zombie” effect makes people more productive (and cuts time requirements)

CS 2606/CS 3114

- We know scheduling works, but how do we change behavior?
- Old:
  - 50+ students
  - Little interaction with instructor/TA as needed
  - Solo programming
- New:
  - 14 students
  - Meet with instructor for each project
  - Pairs if desired
  - Schedule sheets

Outcome

- No recognizable difference

Algorithm Visualization: Features

- Pseudocode display
- Back Button
- Animation vs. “next” step

Tutorials vs. AVs

- Integrated text and activities (applets)
- Guide questions/directed activity
- Built-in quizzing (future)
- Explanatory applets vs. “analysis” applets
- Takes a long time to develop (several students over two years)
- In progress:
  - Hashing
  - Memory management
  - Search Trees

AVs: Hashing Tutorial

- Section 1: Standard lecture and textbook for one week
- Section 2: In-class tutorial use for one week (same material)
- Student reaction: Universally positive for tutorial
- Section 2 had significantly better score in post-test

AV Community (AlgoViz)

- NSF CCLI grant, connections to NSDL/Ensemble project
- Problem:
  - – Some identifiable successes for AVs
  - – Have High faculty and student favorability ratings
  - – But AVs have little overall impact on education
- Solution
  - – Build a community of users/developers
  - – Better disseminate best practices information

AlgoViz Wiki Catalog Data

- A collection of links to nearly 450 Avs
- Some results:
- Topical Distribution
- Who/where
- Quality
- Access Stability

NSDL Project Proposal

- Create a new model of "dissemination" to lower barriers to access
- Move away from the "digital library" model of users coming to collections
- Notification via social networks
- Focus on "community-driven" content development
  - Discussion, review, ratings
  - Think Amazon, but we have critical mass issues