Reproducibility(?) Review Proposal

- Proposal to evaluate reproducibility of submissions to an annual CS / engineering conference
  - Conference is ACM sponsored & published
  - Accepts 25 to 30 papers / year, perhaps 70% include a computational element
  - Program committee has expressed support
- Inspired by SIGMOD "repeatability & workability" evaluation procedures
  - Bonnet et al, SIGMOD Record, DOI: 10.1145/2034863.2034873
- This conference has much more homogenous computational efforts than SIGMOD
  - Typically a handful of plots generated by a few hundred lines of Matlab that runs in a few hours on a laptop

Procedure

- ## Repeatability Evaluation Committee (REC)

  - Get recommendations for postdocs / senior grad students from members of the program committee (PC)

  - Papers go through normal PC review process

  - Authors of accepted papers are invited to submit a repeatability package (RP) at time of final paper submission

  - Authors and REC are provided evaluation criteria in advance

- ## RP contains a document, software and data

  - Document explains what elements of the paper are repeatable, system requirements and a procedure for installation, execution and extraction of results

  - Software can be provided by: link to public repository, archive file, VM, AMI, runmycode.org, …?

Evaluation Criteria

- Three criteria, each rated 0-4
  - "Repeatable" if average score of 2, all scores > 0
  - Not clear how to combine scores from different reviewers
  - Not clear what elements of the reviews should be public
  - If not repeatable, no effect on the paper
  - If repeatable, the instruction document must be included in ACM DL supplemental material, small software and data could be included

- Criteria 1: Coverage
  - 0: no computational elements are repeatable
  - 1: at least one repeatable element
  - 2: majority of elements are repeatable
  - 3: all repeatable and/or most extensible
  - 4: all extensible

Evaluation Criteria

- ## Criteria 2: Instructions
  - 0: none included
  - 1: installation instructions but little else
  - 2: for every computational element that is repeatable there is a specific instruction explaining how to repeat it
  - 3: there is a single command that almost exactly recreates each repeatable element
  - 4: additional explanations of design decisions, extensions, …

- ## Criteria 3: Quality(?)
  - 0: No evidence of documentation or testing
  - 1: The purpose of almost all files is documented
  - 2: Almost all elements within source code and all data file formats are documented
  - 3: At least some components of the code have some testing
  - 4: Significant unit and system test coverage