Author Contribution Statement

Chen Feng Tsai, Erich Congo Strange, Kalie Knecht, and Gavril Moniaga * Confidence interval code in confidence.py: Gavril * Wrote test suite for confidence.py, and added assert statements in confidence.py: Congo * Contributed to docstrings in confidence.py: Congo * Ensured all documents adhered to Pep8 and Pep257 style guidelines: Congo * Contributed to debugging process for Analysis Part I: Congo * Analysis part I and testing of overflow bug: Chen * Analysis parts II-IV: Kalie * Brief user guide: Kalie * makefile for pdf rendering of notebooks: Kalie

Release Notes

Confidence Interval Code

- Developed `cibin` code base which finds the 2-sided 1-alpha confidence bounds for the average treatment effect. The method in Li and Deng is implemented to calculate the two sided confidence bounds.
- Helper functions developed to assist in calculating the two sided confidence bound
    - `N_generator` to generate the tables algebraically ocnsistent with data from an experiment with binary outcomes
    - `filterTable` to check whether summary table Nt of binary outcomes is consistent with observed counts.
    - `potential_outcomes` to make a 2xN table of potential outcomes from the 2x2 summary table Nt
- Docstrings written for main confidence interval code and all helper functions
- All functions contain input validation - ValueErrors for improper input

Test Suite

- All functions get tested for various types of erroneous inputs
- Accuracy of tau_twosided_ci is tested by comparing results to examples from Method 3 in Li & Ding's paper, as well as ensuring that with small numbers, the lower bound and upper bound are the same whether exact ==True or exact== False
- potential_outcomes is tested against manual calculations

Use Guide

- Preliminary use guide in `docs/01-Getting-Started.ipynb` to assist the user in implementing the `cibin` code base. The notebook contains some background information on the mathematics being implemented as well as sample code.