**Patient Matching Problem User Guide (PMP\_.py)**

**Purpose & Background**

After a patient has been given a propensity score based on various characteristics (i.e. age, gender, existing conditions, etc.) they should be matched with a control patient that has a similar propensity score so that outcomes can be measured fairly. The purpose of this problem is to minimize the total difference in score between each patient and control match. The problem is formulated and solved using minimum cost flow. Currently, at MARQUI there is already a program that does this; however, it does not account for patients and controls that have multiple cases. This does not allow for statistical independence. This python code ensures that if there are patients with multiple cases within the patient group, or multiple cases within the control group, or a case in the patient group and a case in the controls group only one of the cases is used.

**Inputs**

To run this program, you will need three inputs following this format:

1. **“PMP\_Patients.csv”**

Table

Description automatically generated

This input should have a header (does not have to be this same titles as shown) and three columns. The first column is each patient’s individual demographic ID. There may be repeating demographic IDs if a patient has multiple cases. This should be an integer. The second column is each cases unique ID. This should not repeat and should be an integer. Finally, the third column is the propensity score assigned to that case or patient. This can be any number, does not have to be an integer.

1. **“PMP\_Controls.csv”**

Table

Description automatically generated

This input has the same set up as PMP\_Patients.csv with the controls’ information.

1. ﻿The program will print **“How many controls would you like to match each patient?”** Input how many controls you would like to match with each patient as an integer (1, 2, 3, etc.)

**Outputs**

1. **“PMP\_Matches.csv”**

Table

Description automatically generated

The program will output one match on each row. Column one is the patient’s demographic ID, column 2 is the patient’s unique case ID, column three is the matched control’s demographic ID, column 4 is the matched control’s unique case ID. If you ask for more than one match per patient each match will be in a new row. The figure above had 2 control matches for each patient.

1. **Optimal Cost**

The program will print the optimal cost which is the minimized sum of the difference between patient propensity scores and control propensity scores.

1. **Run time**

The program will print the amount of time it took to run the program.

**How to:**

1. Ensure “PMP.py”, “PMP\_Patients.csv”, “PMP\_Controls.csv”, and “PMP\_Matches.csv” are stored in the same location.
2. Run the Code
3. The program will ask “﻿How many controls would you like to match each patient?” Respond with an integer (1, 2, 3, etc.). Ensure there are enough controls to match the number of times you want to match with each patient. For example, ensure there are at least 5,000 controls for 1,000 patients with 5 matches.
4. Let the program run. A program with 1000 patients and 10,000 controls will take over 12 hours. A program with 150 patients and 600 controls should take about 12 seconds.
5. Once the program is done running the matches will be in “PMP\_Matches.csv”.