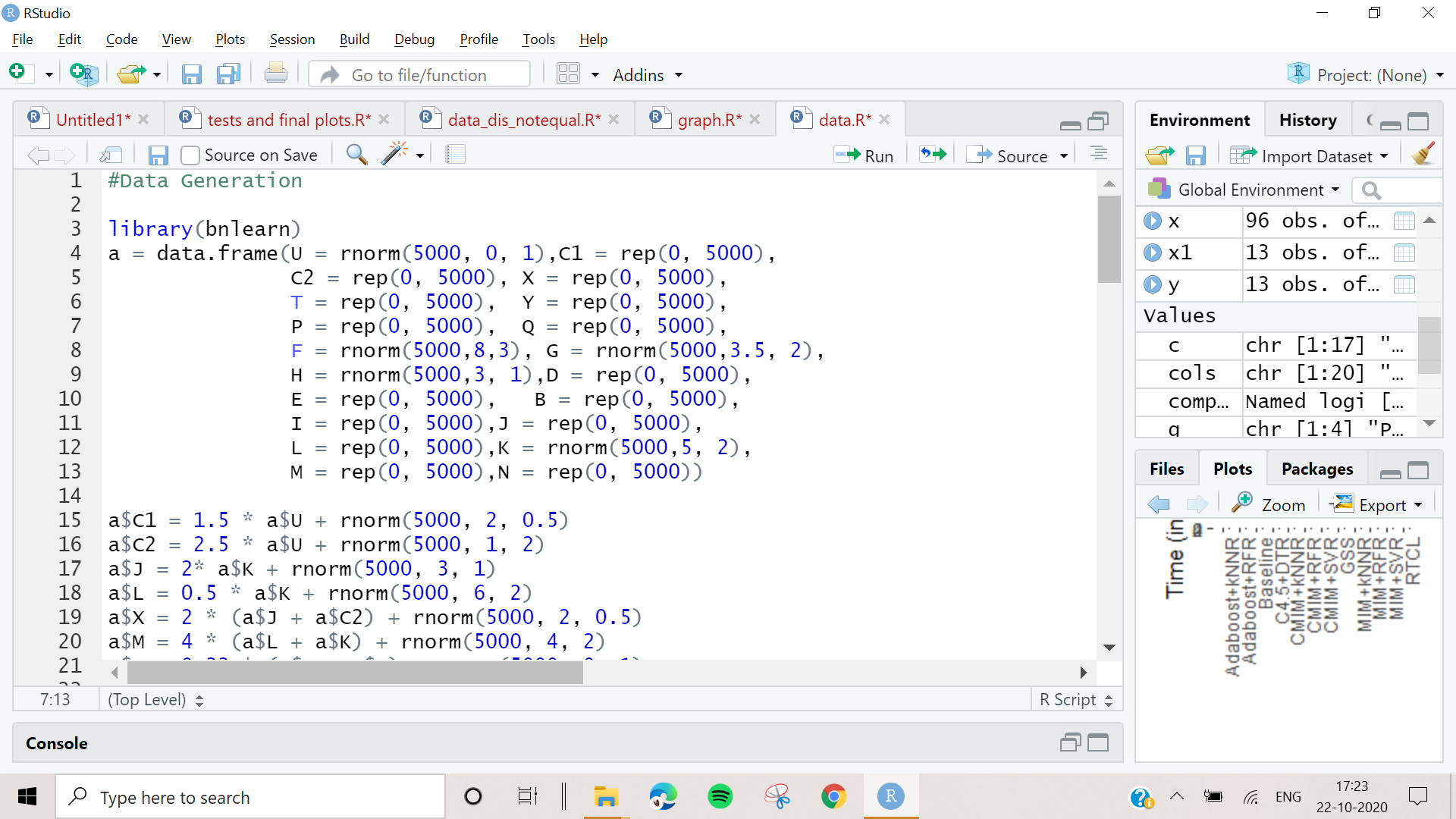
**STEPS TO EXECUTE THE CODE:**

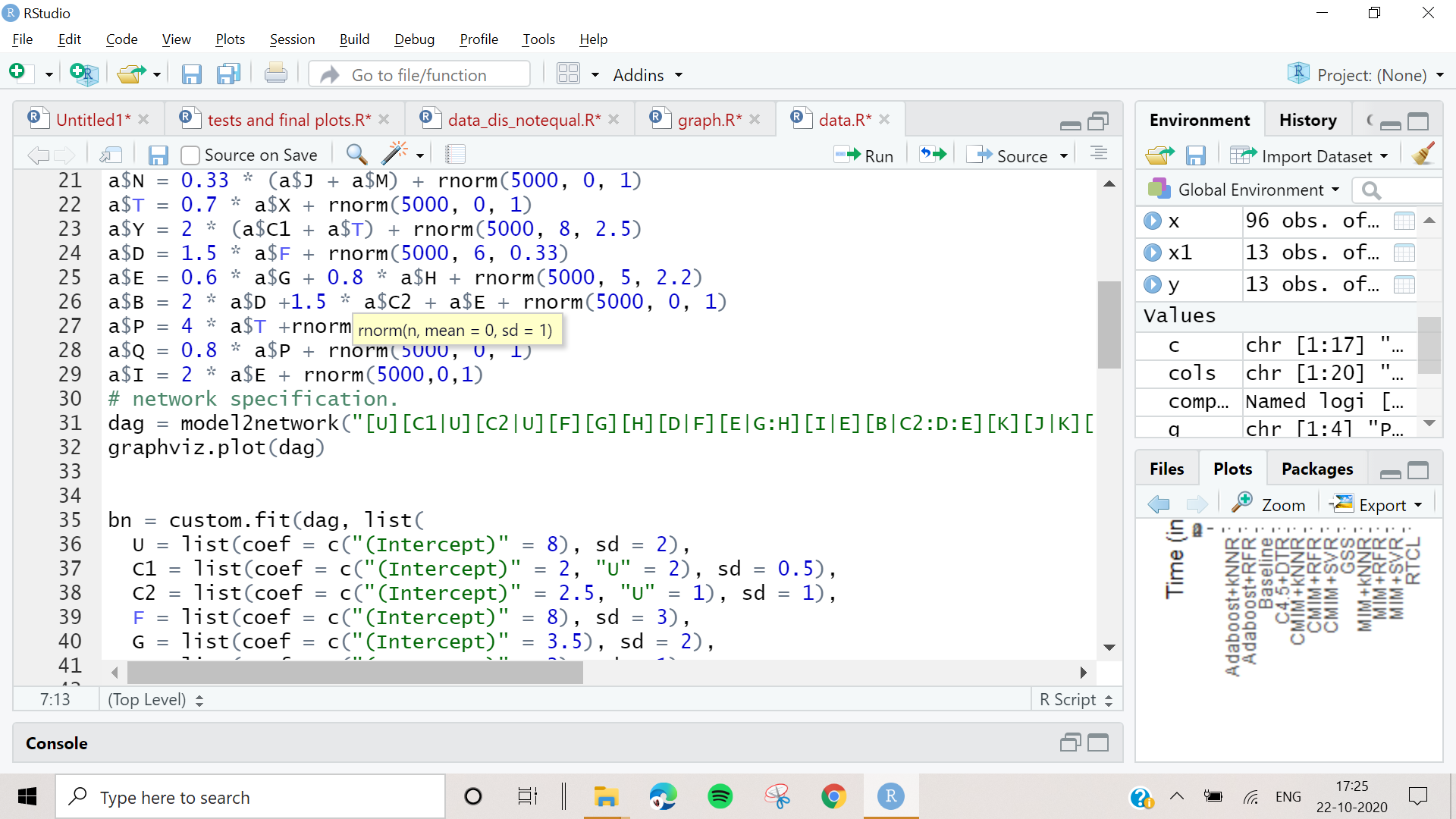
**DATSET GENERATION:**

1) Install packages **bnlearn, pclag**

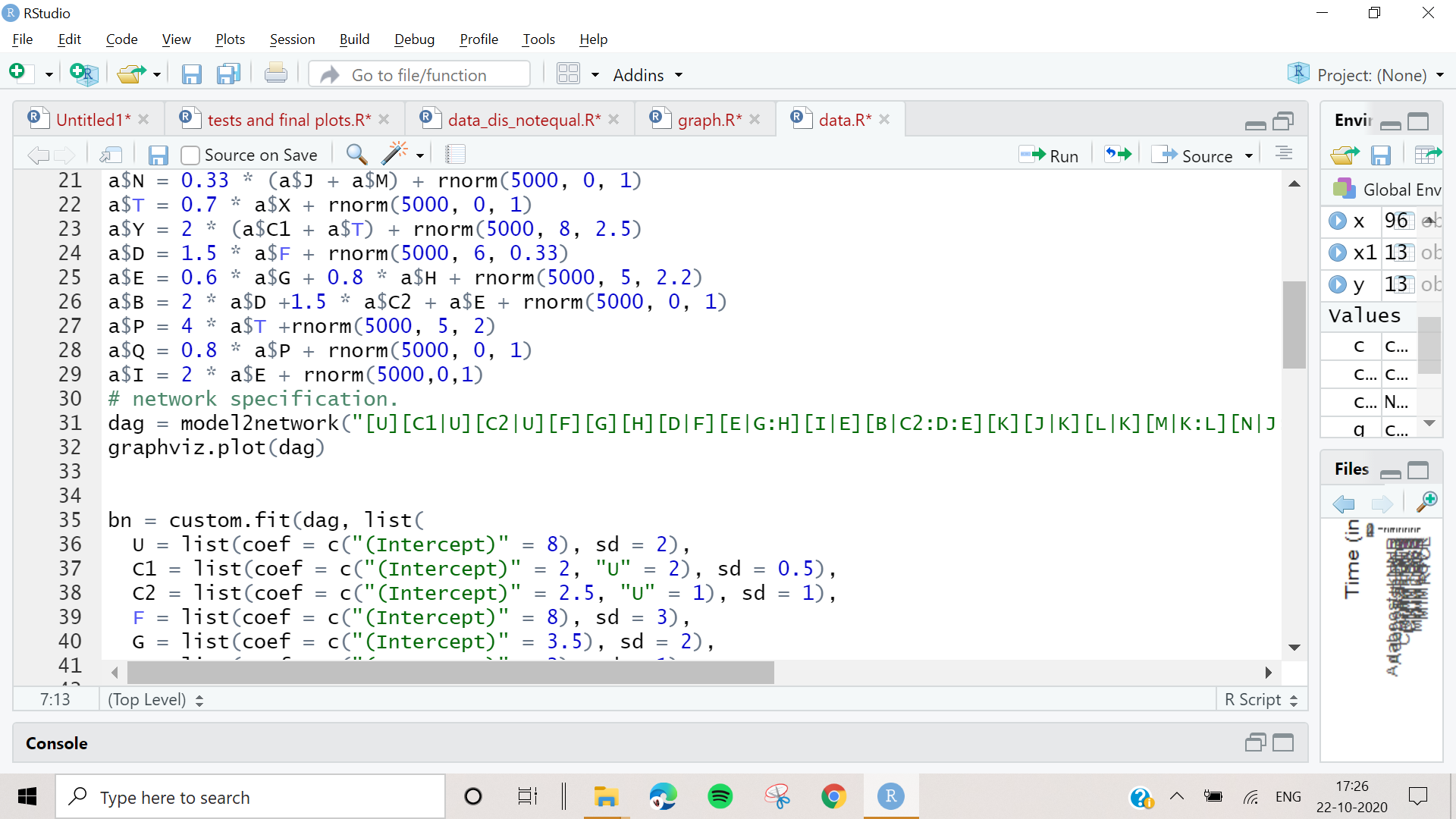
2) If you want to visualize the input to the graph, use **data.R.**

2.1 Run the following code:

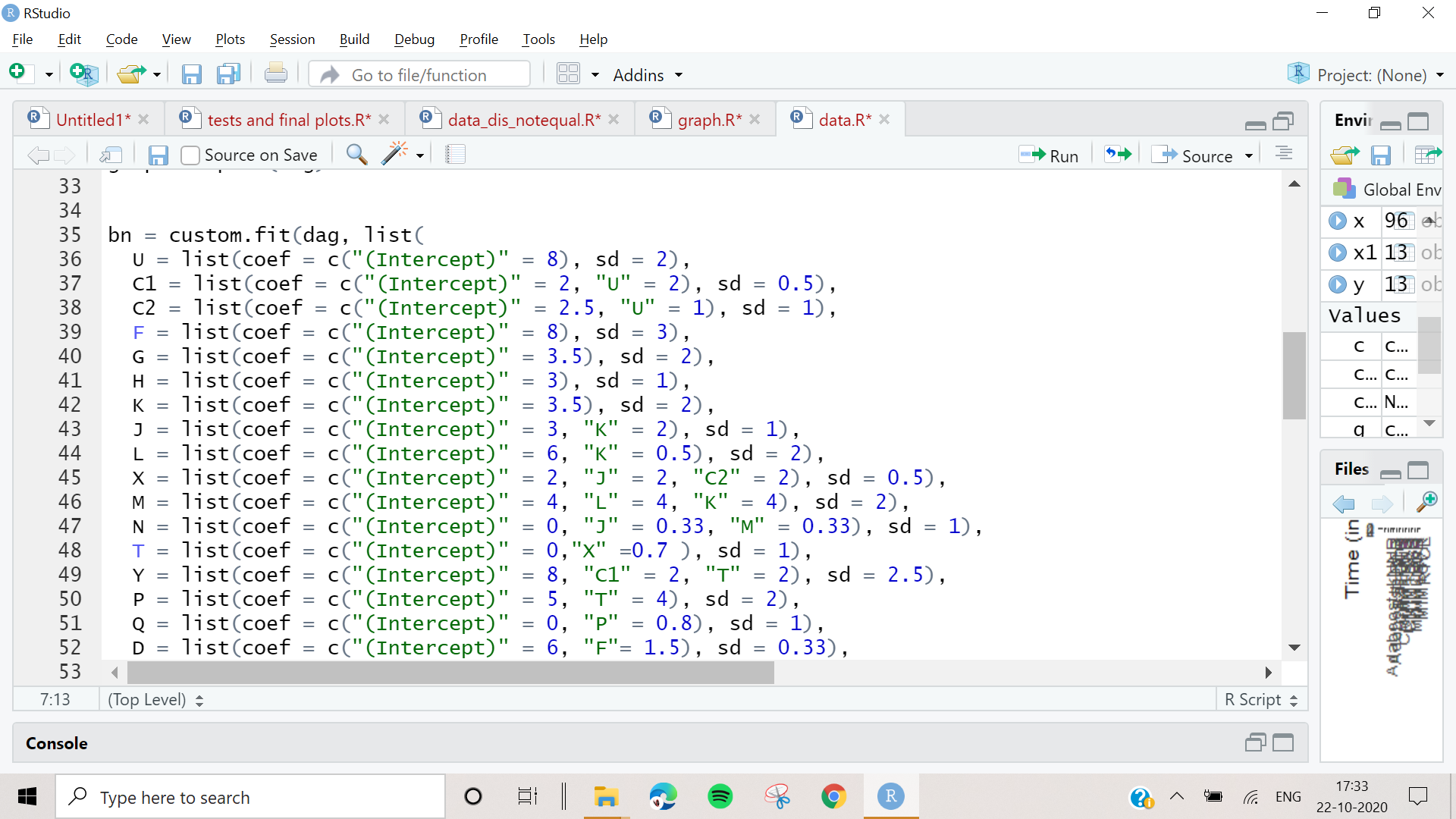


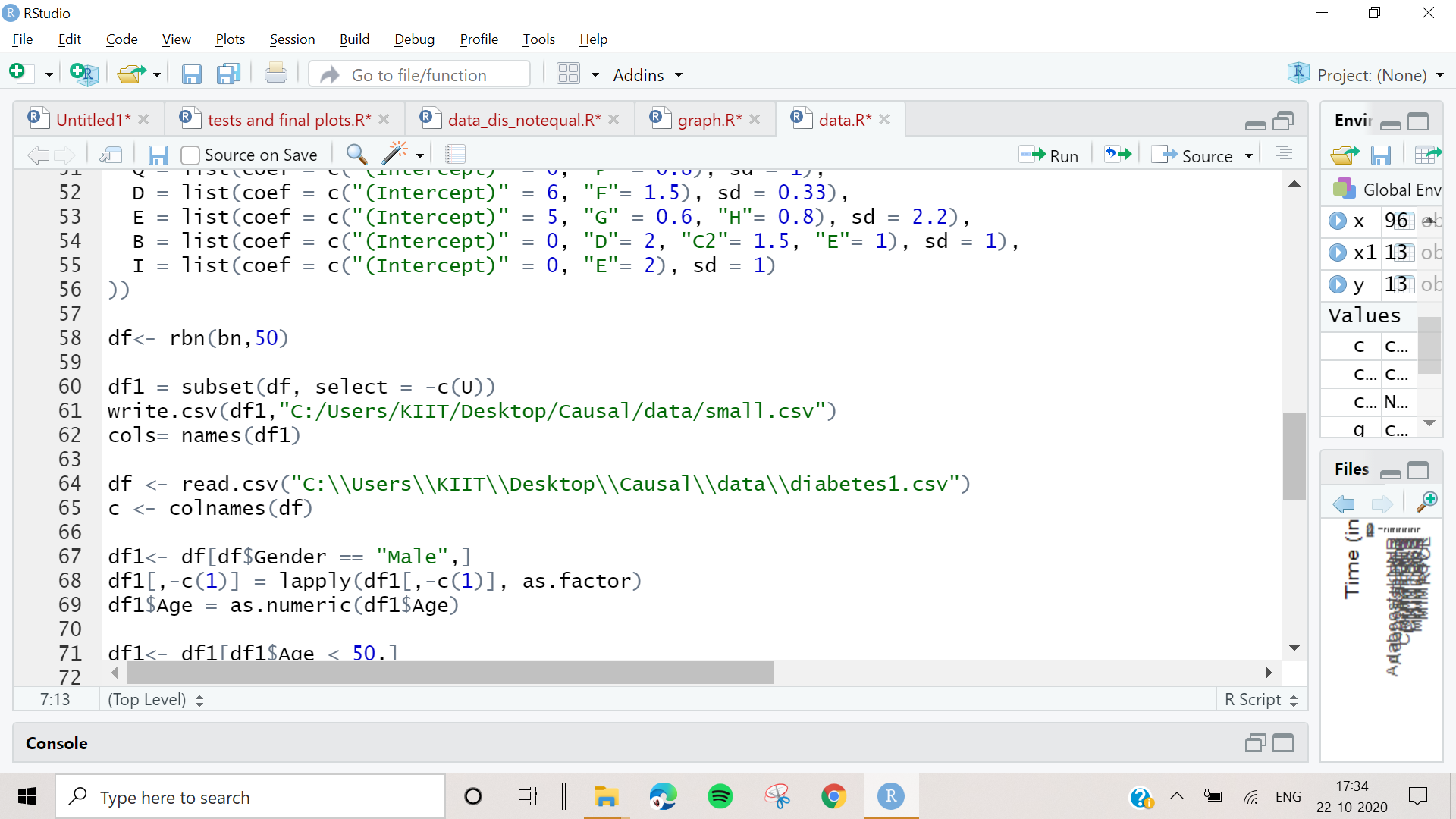


2.2 To visualize the Ground truth Graph in R use:



2.3. Once Confirmed with the dataset distribution run:





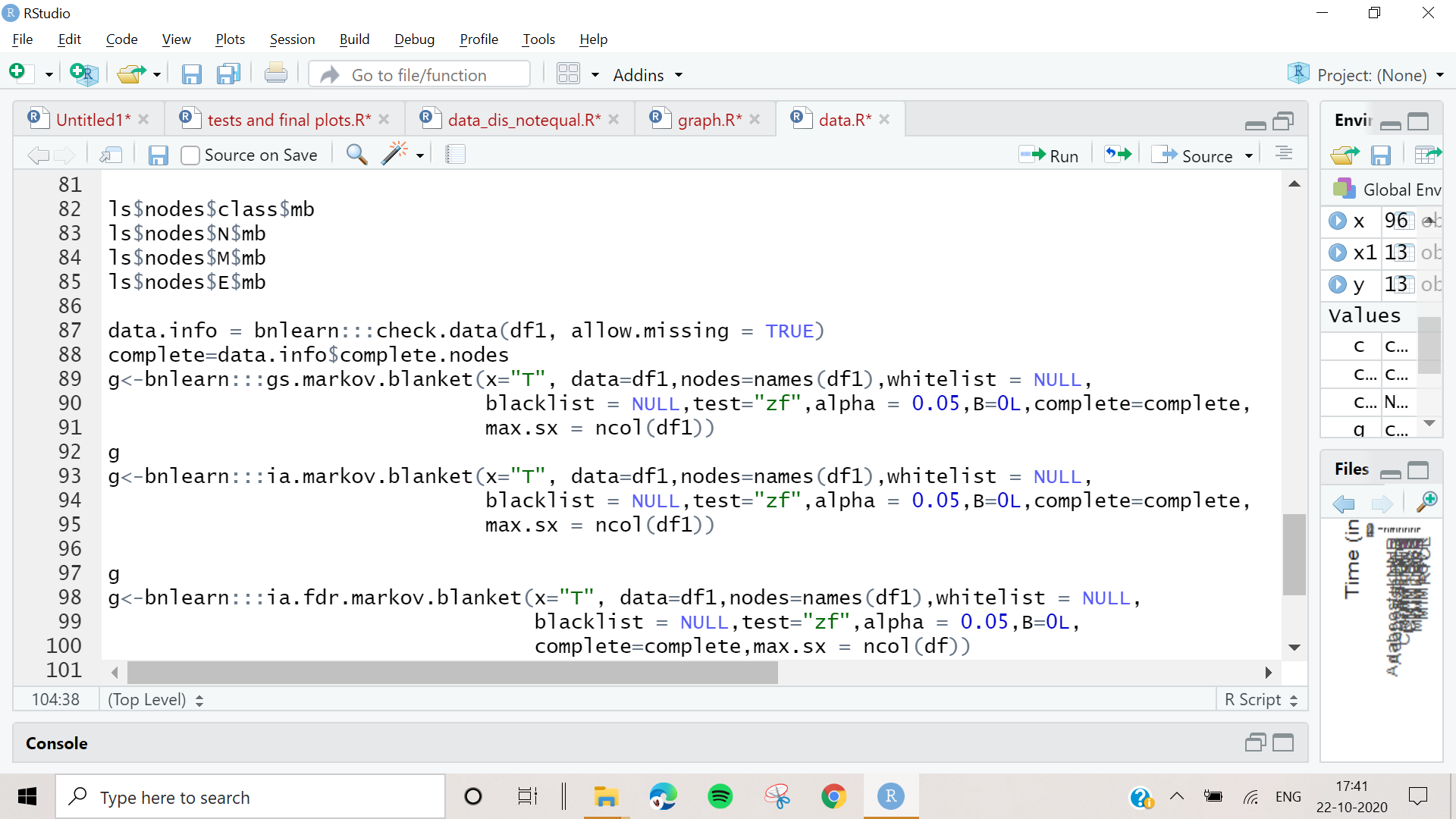
Rbn(bn, sample\_size)

Where sample\_size can be any number

**RCTL:**

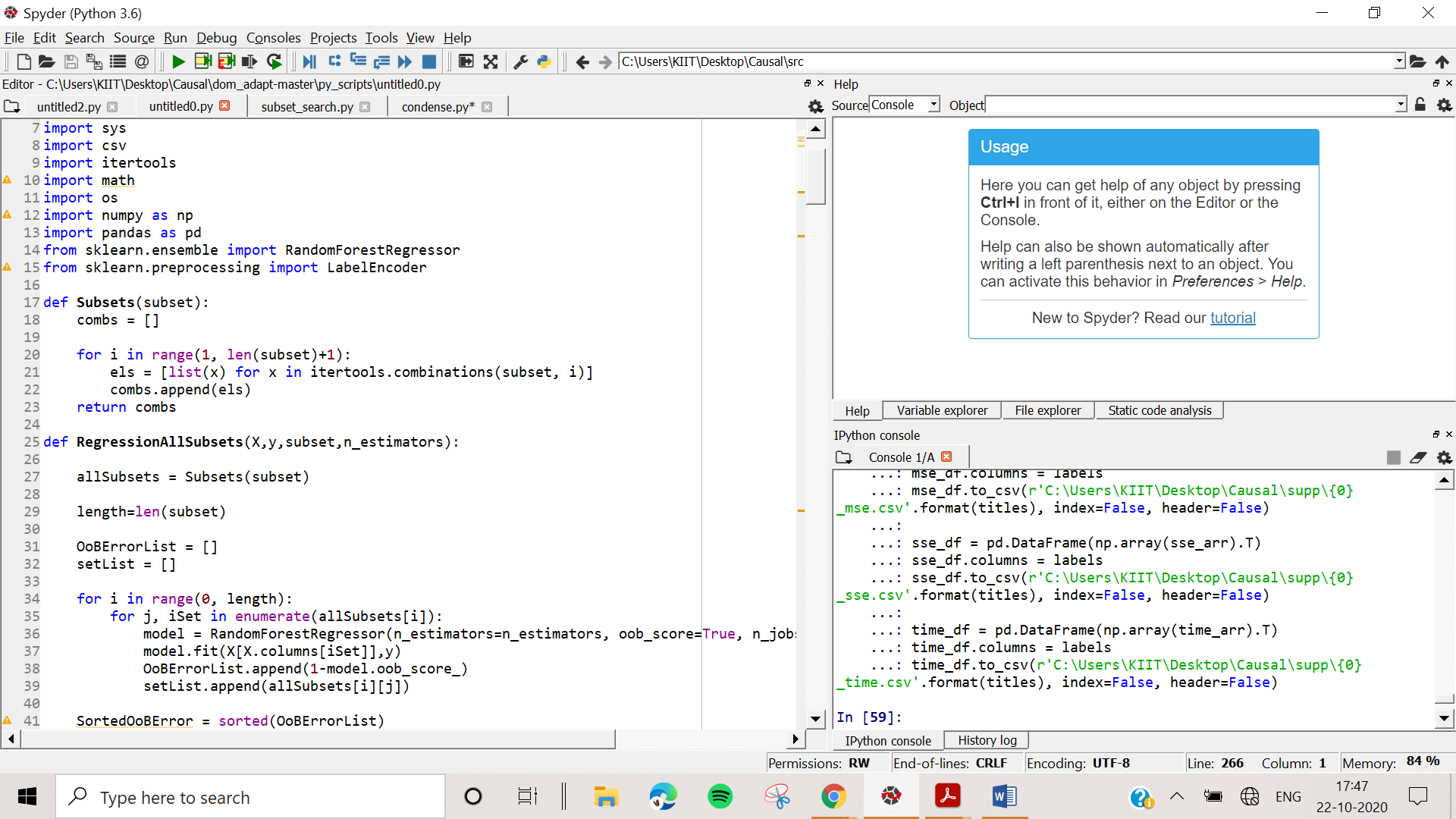
3) But, if you want to just generate the graph and get the dataset, run **graph.R** which only entails steps 2.2 and 2.3

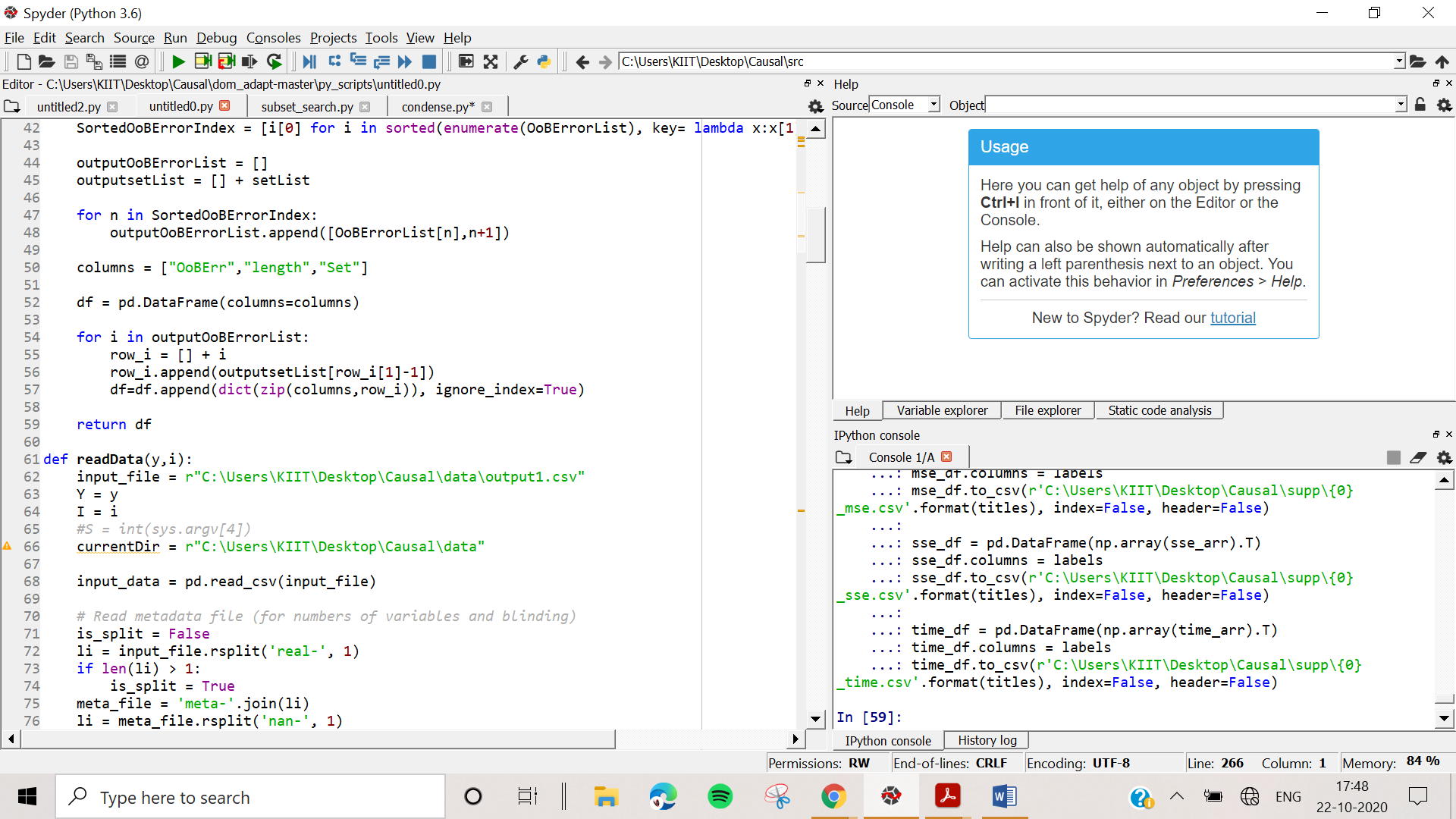
4) After getting the dataset choose any one Markov blanket discovery algorithms and run it Markov Blanket

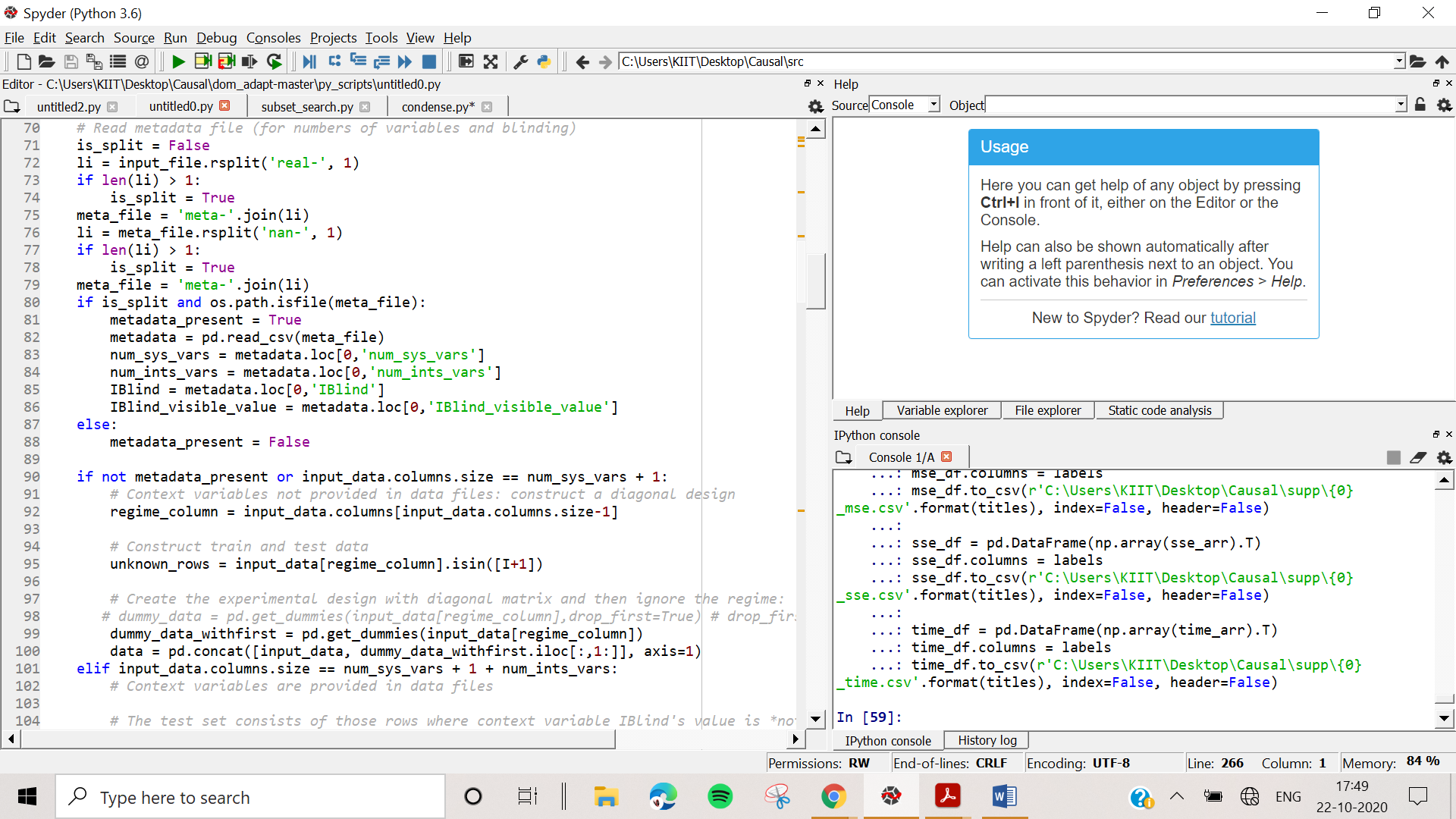


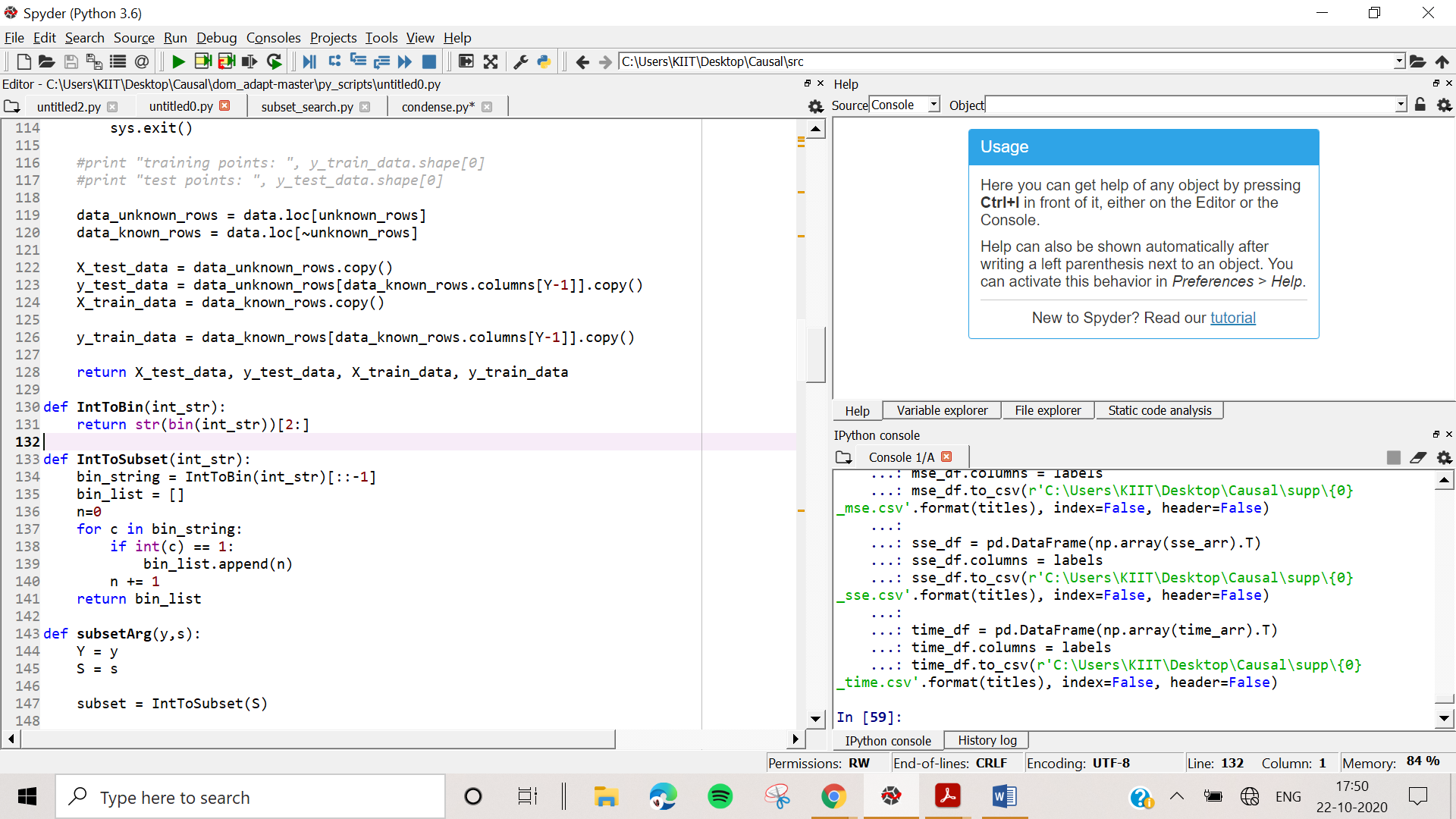
We can use g to get the necessary info, if there are not context variables in the Markov blanket only it is returned. Else we check in the neighbour. For which we need to find all possible subsets of the given set, which can be done by running:

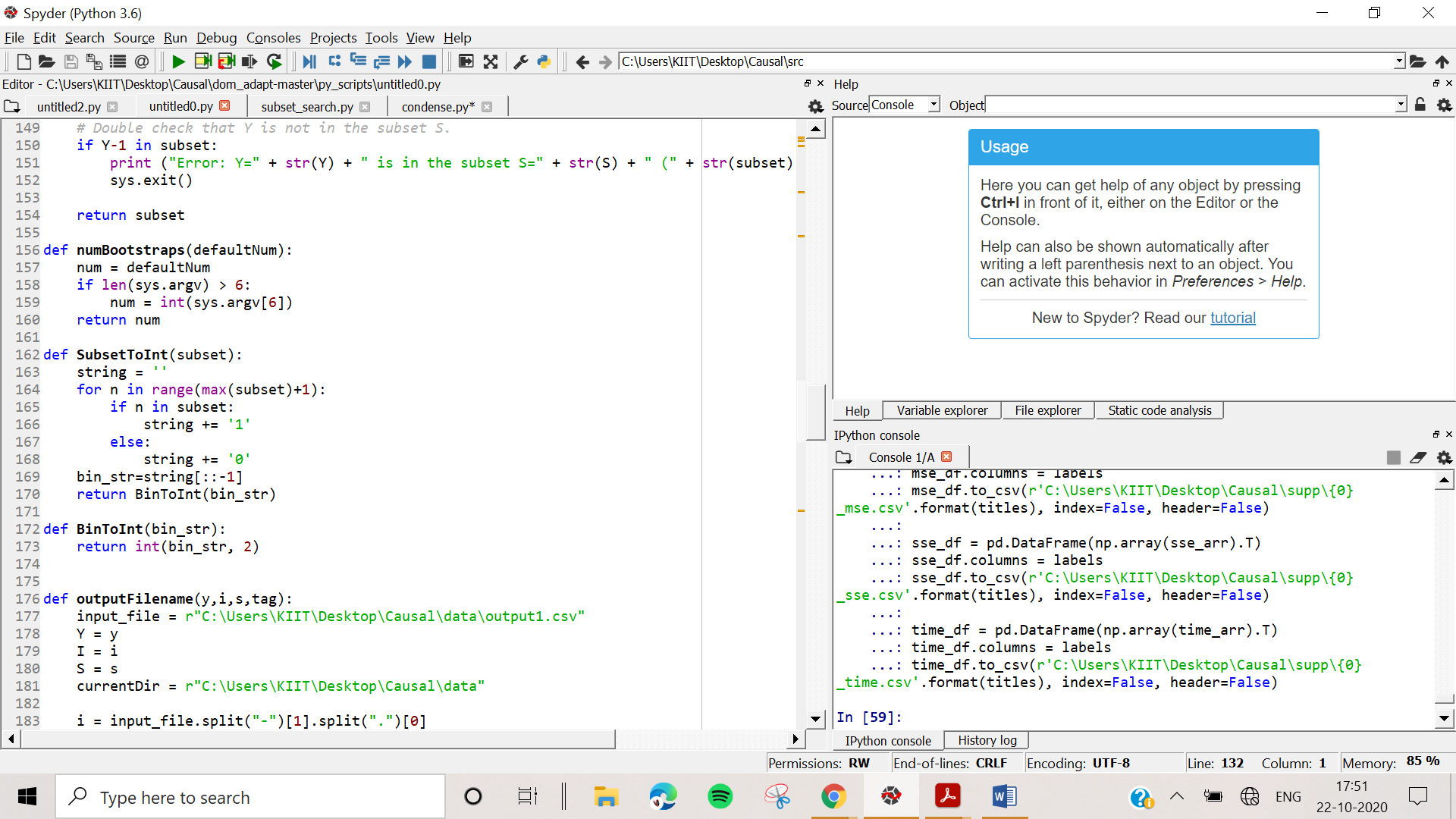
Install all python packages, then using untitled0.py:

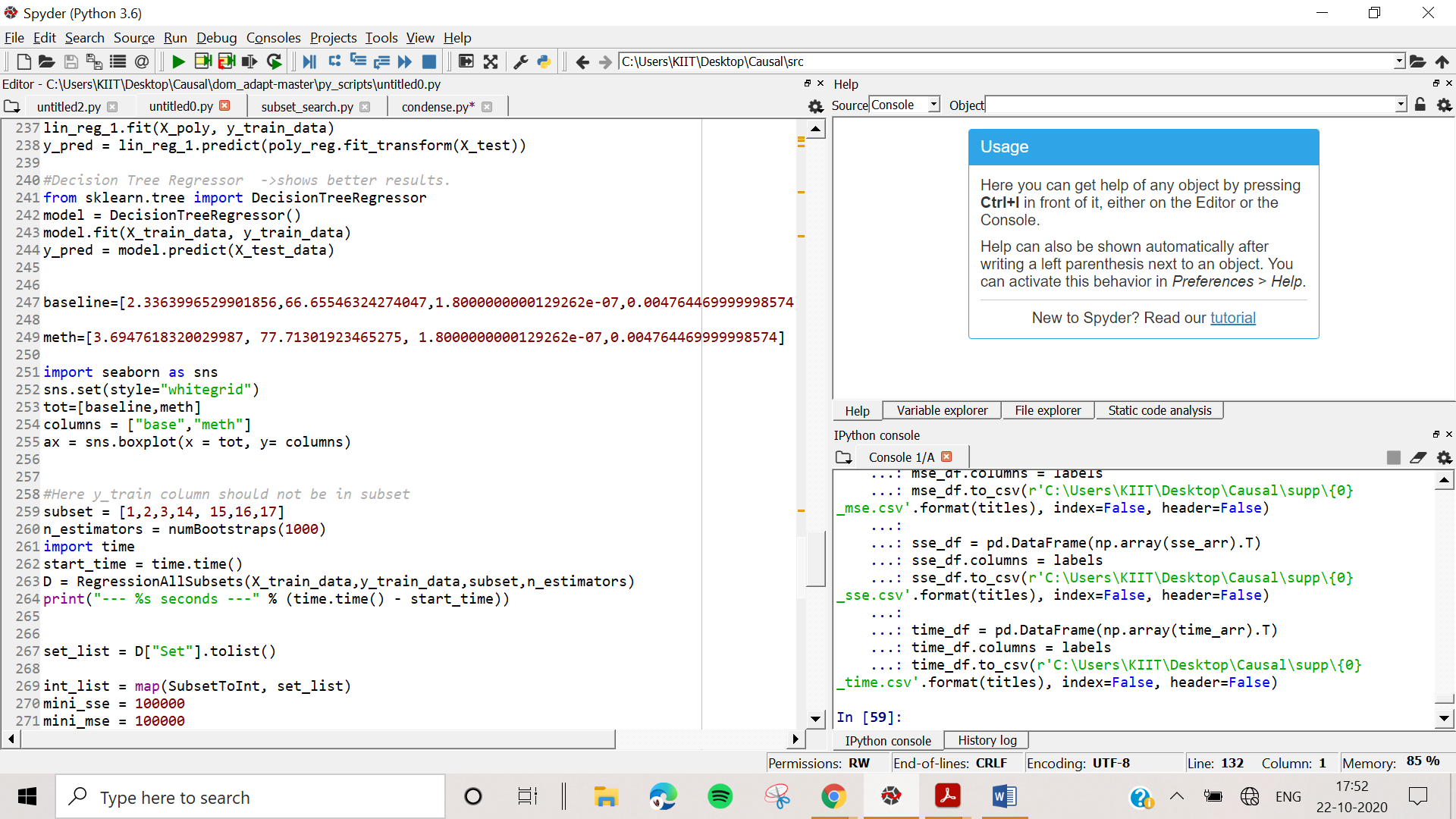






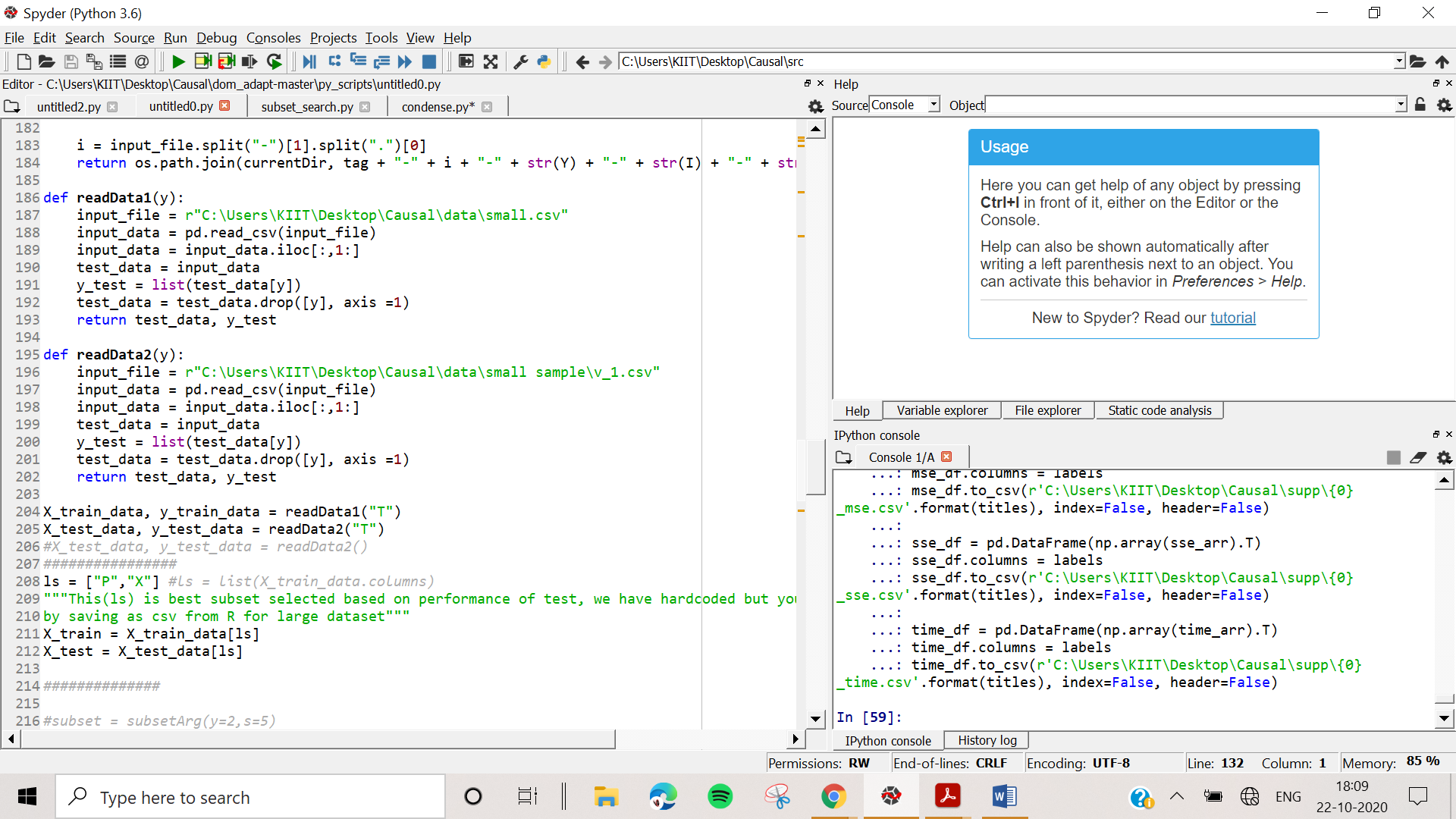






You have all possible subsets in S. Now you can save these to excel or directly import them to R for the ci-tests:

Depending on the type of data, you choose the ci-test. The separating set for the ci-test is used here:

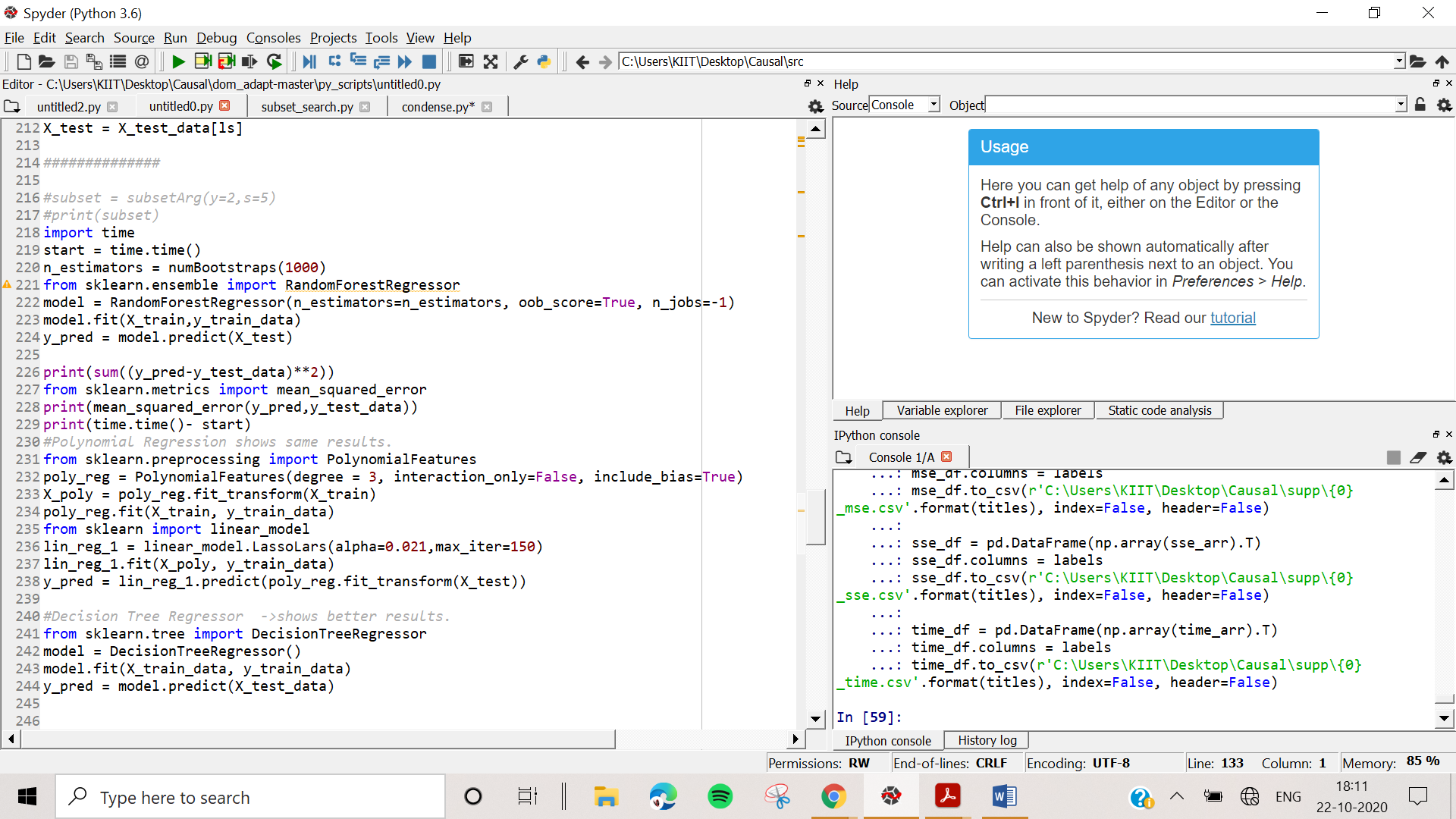


readData1 -> source domain

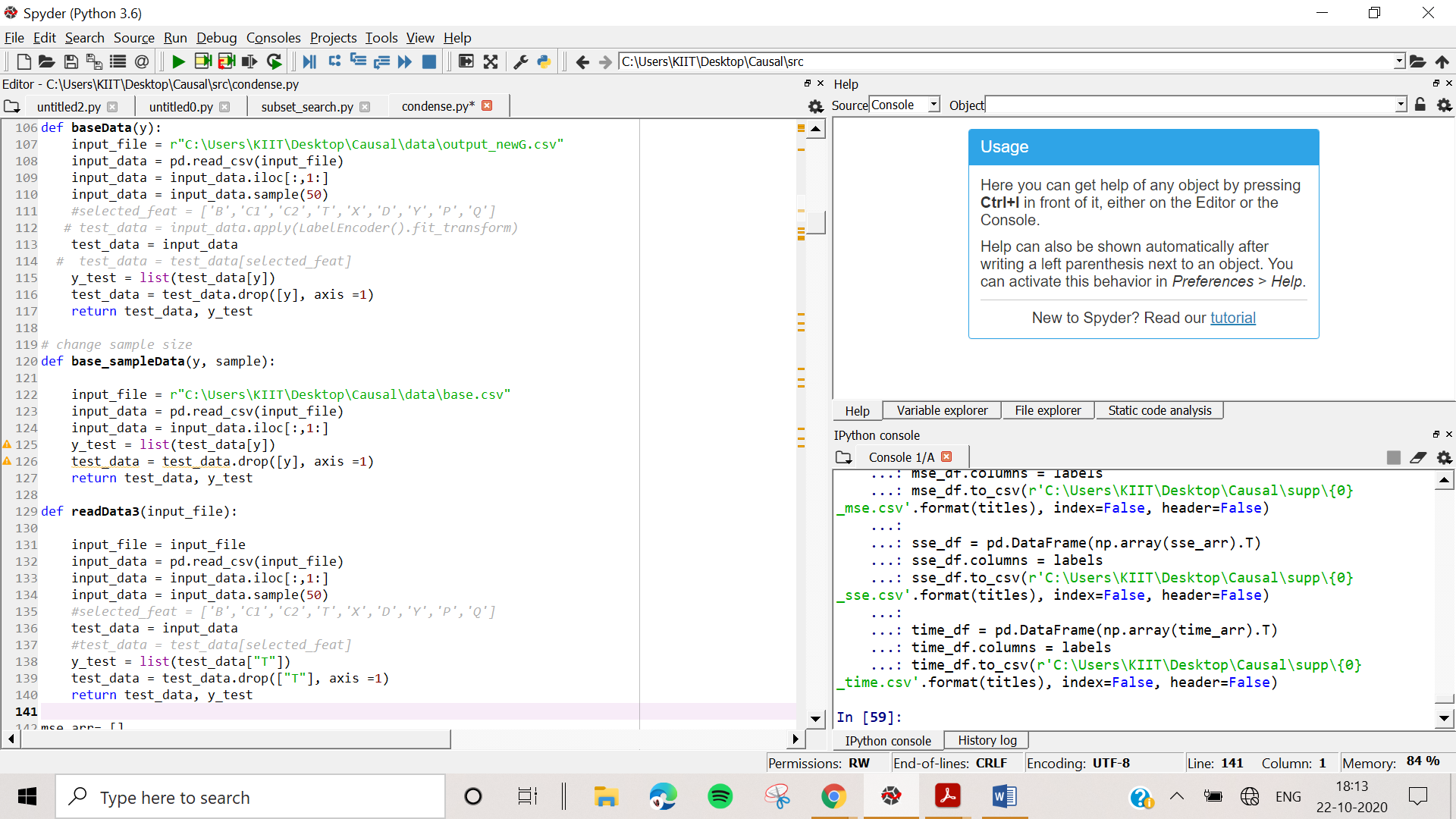
readData2-> target domain

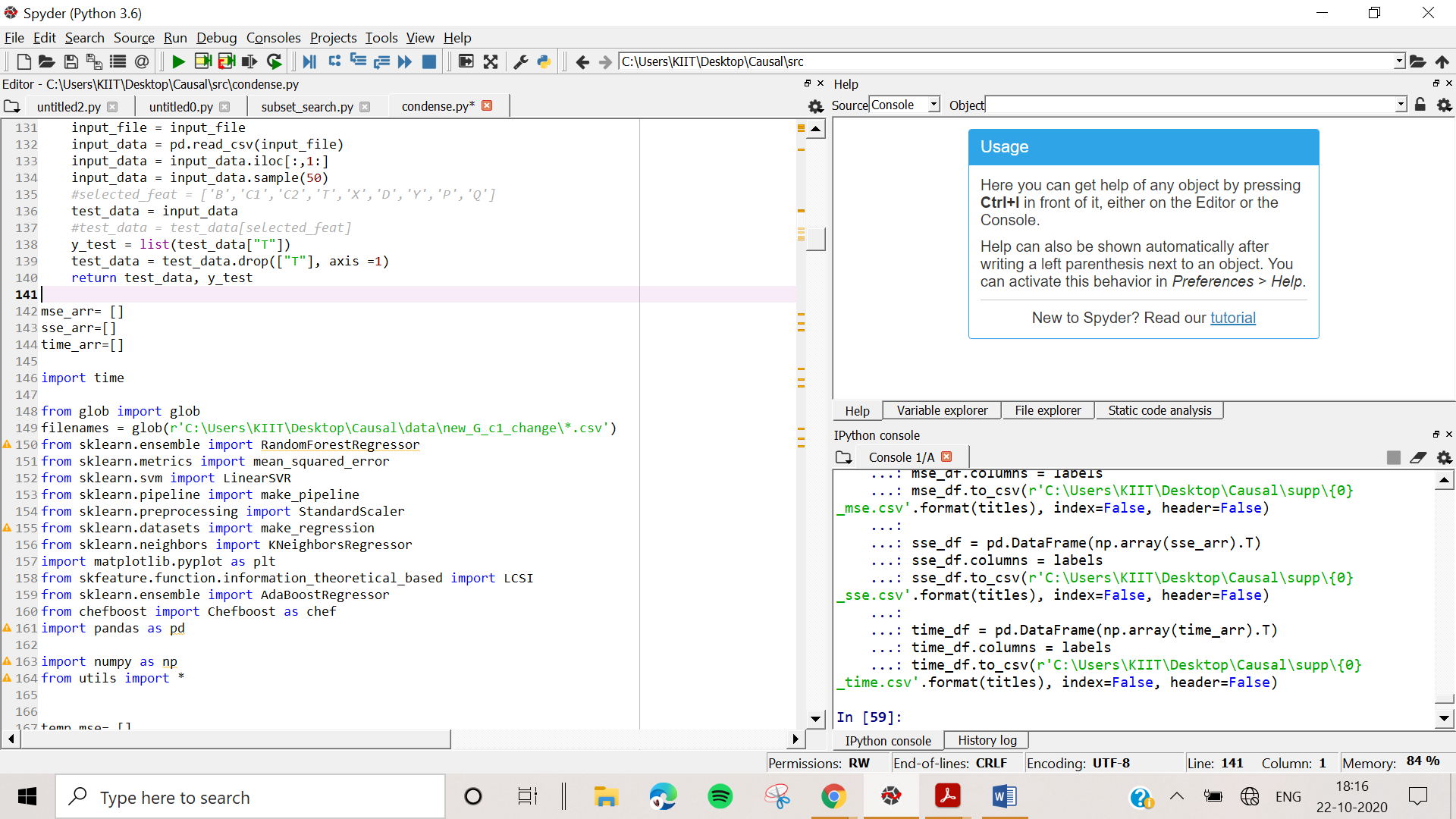
ls = [separating feature names]

Then we predict:



Or if you want to run all approaches we used to compare in the paper, just load the correct datasets and path in **condense.py**:





baseData -> path to source domain

filenames-> path to target domain folder.

Then execute the whole .py, all results will get printed and cataloged