Causal Inference in Python: A Vignette

Laurence Wong

April 25, 2015

Installation

CausalInference can be installed using `pip`, and will run provided the necessary dependencies are in place. On Ubuntu systems, the following commands should take care of all the essential steps if you are starting from scratch:

```
$ sudo apt-get update
$ sudo apt-get install python-pip python-numpy python-scipy
$ sudo pip install causalinference
```

Minimal Example

The main object of interest in CausalInference is the class `CausalModel`. It takes as inputs three NumPy arrays: `Y`, an N-vector of observed outcomes; `D`, an N-vector of treatment status indicators; and `X`, an N-by-K matrix of covariates. The following code snippet illustrates how the class can be invoked using random data simulated from NumPy's `random` module.

```
>>> import numpy as np
>>> from causalinference import *
>>> Y = np.random.rand(1000)
>>> D = np.random.randint(0, 2, 1000)
>>> X = np.random.rand(1000, 3)
>>> causal = CausalModel(Y, D, X)
```

Class Attributes and Methods

Once an instance of the class `CausalModel` has been created, it will contain a number of attributes and methods that are relevant for conducting a causal analysis. Tables 1 and 2 contain a brief description of these attributes and methods.

Attribute	Description
`N, N_c, N_t, K`	Integers indicating sample sizes and number of covariates.
`covariates`	Dictionary-like object containing summary statistics for the covariate variables.
`pscore`	Dictionary-like object containing propensity score data, including estimated logistic regression coefficients, predicted propensity score, maximized log-likelihood, and the lists of the linear and quadratic terms that are included in the regression.
`cutoff`	Floating point number specifying the cutoff point for trimming on propensity score.
`blocks`	Either an integer indicating the number of equal-sized blocks to stratify the sample into, or a list of ascending numbers specifying the boundaries of the strata.
`strata`	List-like object containing the list of stratified propensity bins.
`est`	Dictionary-like object containing treatment effect estimates for each estimator used.

Table 1: Attributes of the class CausalModel. Invoking `print` on any of the dictionary- or list-like attribute above yields customized summary tables. Note that some attributes are only created after the relevant methods have been called.

Method	Description
`restart`	Reinitializes data to original inputs, and drop any estimated results.
`propensity`	Estimates via logit the propensity score using specified linear and quadratic terms.
`propensity_s`	Estimates via logit the propensity score using the covariate selection algorithm of Imbens and Rubin (2015).
`trim`	Trims data based on propensity score using the threshold specified by the attribute `cutoff`.
`trim_s`	Trims data based on propensity score using the cutoff selected by the procedure of Crump, Hotz, Imbens, and Mitnik (2008).
`stratify`	Stratifies the sample based on propensity score as specified by the attribute `blocks`.
`stratify_s`	Stratifies the sample based on propensity score using the bin selection procedure suggested by Imbens and Rubin (2015).
`blocking`	Estimates average treatment effects using regression within blocks.
`matching`	Estimates average treatment effects using matching with replacement.
`weighting`	Estimates average treatment effects using the Horvitz-Thompson weighting estimator modified to incorporate covariates.
`ols`	Estimates average treatment effects using least squares.

Table 2: Methods of the class CausalModel. Invoke `help` on any of the above methods for more detailed documentation.