Calculating $Cov(\overline{\log \lambda_{it}}, \overline{\tau_{it}})$

January 14, 2022

1. Calculate individual log lambdas for every farmer i in every year t: $\log \lambda_{it}$
   (a) Create dataframe of expenditures `exp` indexed by `j`: unique ID given by ID + state + year, `m` = 1, `t` = year
   (b) For household composition dataframe `hh`, use log of household size
   (c) `cfe.Result(y=exp, z=hh)`
2. Merge payments data with ARMS data for farmer i in year t based on zip code z
   (a) Sum payments τ_{iztp} from all programs p for each farmer i living in zip code z in year t to get τ_{izt}
      - Note that τ_{izt} could be the total payments to farmer i living in zip code z in year t, or it could be $(\tau_{iztp_1}, ..., \tau_{iztp_n})$ for n different payment categories
   (b) Calculate the mean $\overline{\tau_{zt}}$ and add it to ARMS observations in zip code z in year t (since we have little individual information to use for matching beyond zip code)
   (c) Use self-reported payments data already in ARMS for farmer i in year t and each zip code z to get τ_{it}
3. Calculate $Cov(\overline{\log \lambda_{it}}, \tilde{\tau_{it}})$ where $\tilde{\tau_{it}}$ is self-reported payments (τ_{it}) instrumented by administrative payment means ($\overline{\tau_{zt}}$)

   Indices of the dataframe `df_analysis`:

   - `j`: unique ID given by ID + state + year
   - `t`: year
   - `Zip`: zip code

   Columns of the dataframe `df_analysis`:

   - `loglambda`
   - `constant` = 1
   - `YEAR`: year dummies
   - `PLC_self`: self-reported Price Loss Coverage payments
   - `DCP_ACRE_self`: self-reported Direct and Counter-cyclical/Average Crop Revenue Election payments
   - `ARC_self`: self-reported Agriculture Risk Coverage payments
   - `MFP_self`: self-reported Market Facilitation Program payments
   - `PLC_admin_mean`: mean of administrative payments to farmers in a given zip code for Price Loss Coverage payments
   - `DCP_ACRE_admin_mean`: mean of administrative payments to farmers in a given zip code for Direct and Counter-cyclical/Average Crop Revenue Election payments

- `ARC_admin_mean`: mean of administrative payments to farmers in a given zip code for Agriculture Risk Coverage payments
- `MFP_admin_mean`: mean of administrative payments to farmers in a given zip code for Market Facilitation Program payments

(a) Use the administrative payments data as instruments

- `Y = df_analysis['loglambda']`
- `X = df_analysis[['PLC_self','DCP_ACRE_self','ARC_self', 'MFP_self']]`
- `Z = df_analysis[['constant', YEAR, 'PLC_admin_mean', 'DCP_ACRE_admin_mean', 'ARC_admin_mean', 'MFP_admin_mean']]`
- `from statsmodels.sandbox.regression.gmm import IV2SLS`
- `results_iv = IV2SLS(Y,X,Z).fit()`
- `results_iv.params`

*Note that these preliminary results do not include the year 2012 yet (waiting for the correct 2012 file to be uploaded to the Data Enclave)

`ARC_self`: $-2.273460 * 10^{-6}(5.991604 * 10^{-7})$
`DCP_ACRE_self`: $-3.880741 * 10^{-6}(4.257992 * 10^{-7})$
`MFP_self`: $-8.506738 * 10^{-7}(4.066743 * 10^{-7})$
`PLC_self`: $-6.625808 * 10^{-6}(9.641586 * 10^{-7})$

(b) Correlation between individual self-reported payments and means of administrative payments at the zip code level (using `np.corrcoef`)

*Note that these preliminary results do not include the year 2012 yet (waiting for the correct 2012 file to be uploaded to the Data Enclave)

- ARC: 0.3433
- DCP_ACRE: 0.3068
- MFP: 0.3751
- PLC: 0.2792