

dsge

October 30, 2024

1 DSGE Modelling with Python

```
[1]: import os, sys
import pandas as pd

path = os.getcwd()
working_dir = os.path.abspath(os.path.join(os.getcwd(), "../.."))
if working_dir.endswith("Framework"):
    sys.path.append(working_dir)
    os.chdir(working_dir)

# print the current work dir
print('Working directory: ', os.getcwd())
```

Working directory: /home/alexei/work/Framework

Read model file and create model object.

```
[2]: from snowdrop.src import driver

# Path to model file
file_path = os.path.abspath(os.path.join(working_dir, 'snowdrop/models/TOY/
↪ JLMP98.yaml'))

# Create model object
model = driver.importModel(fname=file_path, use_cache=True)
```

```
=====
Monetary policy model example
=====
```

Model:

```
name: "Monetary policy model example"
file: "/home/alexei/work/Framework/snowdrop/models/TOY/JLMP98.yaml"
Non-Linear Model
```

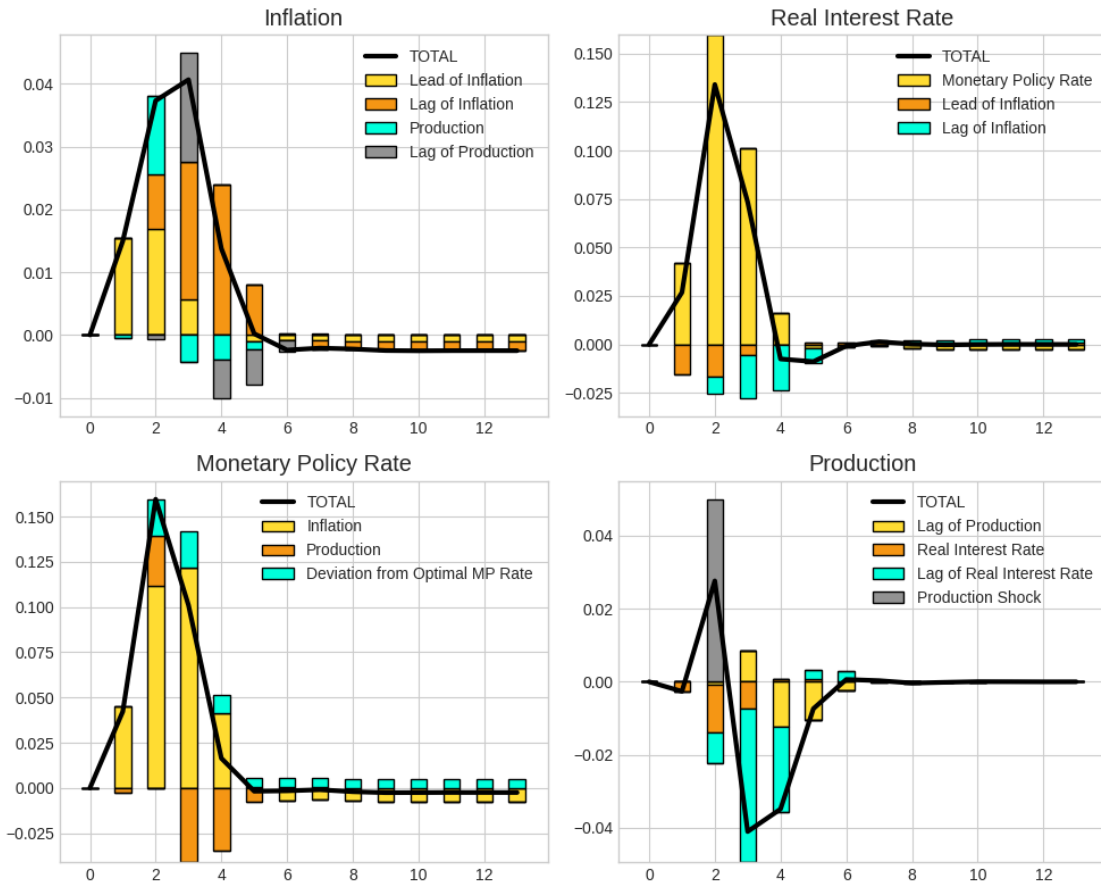
Transition Equations:

```
1      0.000 : PDOT = p_pdot1*PDOT(+1) + (1-p_pdot1)*PDOT(-1) +  
p_pdot2*(g**2/(g-Y) - g) + p_pdot3*(g**2/(g-Y(-1)) - g)  
2      0.000 : RR = RS - p_pdot1*PDOT(+1) - (1-p_pdot1)*PDOT(-1)  
3     -0.000 : RS = p_rs1*PDOT + Y + ers  
4      0.000 : Y = p_y1*Y(-1) - p_y2*RR - p_y3*RR(-1) + ey
```

Set model parameters. Specify shocks and exogenous variables. Run simulations.

```
[6]: # Model parameters  
cal = {'g':0.049,'p_pdot1':0.414,'p_pdot2': 0.196,'p_pdot3': 0.276,  
      'p_rs1':3,'p_y1':0.304,'p_y2':0.098,'p_y3':0.315}  
model.calibration['parameters'] = list(cal.values())  
  
# Shocks  
model.options["periods"] = [2]  
model.options["shock_values"] = [0.05]  
  
# Exogenous variables. Revision of Monetary Policy Rate.  
from snowdrop.src.utils.util import getExogenousSeries  
  
exog_data = {'ers': pd.Series([0,0.02,0.02,0.01,0.005],[1,2,3,4,5])}  
#exog_data = {'ers': pd.Series([0]*5,[1,2,3,4,5])}  
model.symbolic.exog_data = exog_data  
model.calibration["exogenous"] = getExogenousSeries(model)  
  
# List of variables for which decomposition plots are produced  
decomp = ['PDOT','RR','RS','Y']  
  
# Function that runs simulations, model parameters estimation, MCMC sampling,  
→ etc...  
rng_date,yy = driver.  
→ run(model=model,decomp_variables=decomp,Output=False,Plot=True)
```

Plotting Decomposition of Endogenous Variables



[]: