Documentation for FiTABM

Phoebe Pearce

31 March 2017

An agent-based model, implemented in R, for modelling the effect of feed-in tariffs on installation of solar panels by households in Great Britain, 2010-2021.

This document aims to give an overview of how to get set up with the model and the ways in which the key functions in the model can be used (i.e. what their arguments are and what they can do).

Getting started

If you don't have R installed, do that: https://cran.rstudio.com/

To make your life easier, also download RStudio: https://www.rstudio.com/products/rstudio/download/

Although FiTABM was written in RStudio, there is no reason it won't work in a different IDE/from the command line.

Once you have R/RStudio and the right packages:

1. Download the FiTABM repository from https://github.com/phoebe-p/FiTABM (Clone or download -> Download ZIP), or create a clone if you prefer
2. Extract the zip file into your preferred location (best to put the files in their own folder)
3. If you have RStudio, open FiTABM.Rproj in RStudio; you'll automatically be in the correct working directly (where all the data files you need are). Otherwise, navigate to the folder where the data files are (where you extracted the .zip to).
4. The functions which form the basis of the program rely on the following R packages:

   readr, dplyr, purrr, ggplot2, stringr, reshape2, lubridate, magrittr

   You can install these yourself, e.g by running the following line of R code:

   install.packages(c("readr", "dplyr", "purrr", "ggplot2", "stringr", "reshape2", "lubridate", "magrittr"))
5. To actually load the functions into your global environment, run all the contents of 01-required_functions.R and 02-run_functions.R (Ctrl/Cmd + A, Ctrl/Cmd + Enter in RStudio). All the functions you need are now in your Global Environment.
6. Now you can start running simulations!

File list

1. FiTABM.Rproj – RStudio project
2. 01-required_functions.R – contains most of the necessary functions, except those for running in batches & actually executing the model.
3. 02-run_functions.R – contains the functions for individual and batch runs; all functions called by the individual run function are defined in 01-required_functions.R
4. Documentation – this document
5. In the folder *Data*:
   - all_inst_1.csv and all_inst_2.csv – data on individual PV installations in the UK. Big files!

- electricityprices.csv – estimated electricity prices in the UK (annual, 2010-2016)
- FiT_levels.csv – Real feed-in tariff (FiT) levels in Great Britain (monthly, 2010-2016)
- LF_mean.csv – Solar PV load factors per GB region
- mean-electricity.csv – Mean electricity consumption, depending on household size and income
- median-electricity.csv – Median electricity consumption, depending on household size and income
- owner_occupiers.csv – number of households which occupy a home they own, in Great Britain (annual, 2010-2016)
- population_mid2012.csv – population estimates per GB region, from 2012
- PV_cost_data_est.csv – PV cost data used in the model (monthly, 2010-2016)
- real_dep_cap.csv – Deployment caps implement on FiT scheme from October 2016

6. README.md

Historical simulations

load_data()

Usage:

load_data(start_date, end_date, FiT_end_date, FiT_type, red_frac, init_fit, final_fit, exp_tar, dep_caps, cap)

Required arguments:

None

Default arguments:

- start_date = "1apr2010"
- end_date = "1sep2016"
- FiT_end_date = end_date
- FiT_type = "real_h"
- red_frac = 0.03
- init_fit = 49.43
- final_fit = 4.18
- exp_tar = 4
- dep_caps = F
- cap = no default

Arguments:

- *start_date* is when the simulation starts. Passed as a string (NOT a date), in the format "1jan2000"
- *end_date* is when the simulation ends. Passed as a string (NOT a date), in the format "1jan2000"
- *FiT_end_date* is when the feed-in tariffs end (so no new registrations, but existing installations keep receiving their FiTs). Passed as a string (NOT a date), in the format "1jan2000"
- *FiT_type* is the degression strategy. This can have the following values:
  - "real_h" - sets the FiTs as they actually were in Great Britain 2010-2016 for $<$ 10kW solar systems.
  - "perc_red" - the generation tariff reduces by a fixed % every month from some starting value (export tariff fixed)
  - "ann_perc_red" - the generation tariff reduces by a fixed % every year from some starting value (export tariff fixed)
  - "linear" - the generation tariff reduces linearly every month between an initial and final value (export tariff fixed)

If you are setting "real_h", you don't need to set anything else, it happens automatically. Has to be passed as a string (so "real_h", not real_h)

- *red_frac* is the fractional (not percentage!) reduction for FiT types "perc_red" and "ann_perc_red". Don't set for the other FiT types.
- *init_fit* is the initial generation tariff (needed for all FiT types except "real_h") in p/kWh
- *final_fit* is the final generation tariff (only need for FiT type "linear") in p/kWh
- *exp_tar* is the export tariff in p/kWh. Needed for all FiT types except "real_h" and assumed to be constant.
- *dep_caps* is a Boolean (TRUE (T) or FALSE (F)). Set to true if you want there to be deployment caps.
- *cap* is how much you want the deployment cap to be per quarter in MW. Only set if you're using dep_caps = T.

batch_run_func()

Usage:

batch_run_func(w, t, number_of_agents, number_of_runs, plot_u, plot_cost, plot_prod, save_name)

Required arguments:

None

Default arguments:

- w = c(0.27, 0.25, 0.05, 0.43)
- t = 0.74
- number_of_agents = 5000
- number_of_runs = 10
- plot_u = T
- plot_cost = T
- plot_prod = T
- save_name = no default

Arguments:

- *w* are the partial utility weights, in the order income, social, economical, capital cost. The defaults are the results of calibration.
- *t* is the adoption threshold, again the result of calibration.
- *number_of_agents* is how many agents the model generates and uses for each model run.
- *number_of_runs* is how many times the model is repeated to produce an average results.
- *plot_u*, *plot_cost*, and *plot_prod* accept TRUE or FALSE. They don't affect how the model runs, but do (T) or don't (F) plot the partial utilities, private and subsidy cost, and production over time.
- *save_name* is how you want any saved results to be identified. If you don't enter anything, your results won't automatically be saved (but they will still be in the workspace after the model is finished running).