# Chapter 3 MATLAB Program Instruction

## 1 Introduction

The MATLAB program is for **ECO 529 Chapter 3: Endogenous Risk Dynamics**. It includes six scripts

- `Lec3Main.m`: the main program, solving for the equilibrium and calling other scripts to display the results.
- `BasicFig.m`: displays variables $q, \psi, \sigma^q, C/N, \sigma^\eta, \mu^\eta$ w.r.t state variable $\eta$.
- `DistImpulseRespon.m`: simulates the distributional impulse response of an economy. Starting from the stationary distribution, we impose a shock (1% quantile of original Brownian shock, i.e. $\mathrm{d}Z_t = -2.32\mathrm{d}t$) for a period of $\mathrm{d}t = 1$, then investigate the density transition period by solving Kolmogorov forward equation. Fan charts and 2D/3D density diffusion figures are plotted in this script.
- `DistImpulseRespon_degen.m`: imposes same impulse as `DistImpulseRespon.m` for a system starting from a degenerated state (the default starting point is the median of the stationary distribution).
- `DistImpulseResponDiff.m`: simulates distributional impulse response (difference to unshocked paths). Suppose we have two systems staying in the same stationary distribution. We impose a shock (1% quantile of original Brownian shock, i.e. $\mathrm{d}Z_t = -2.32\mathrm{d}t$) for a period of $\mathrm{d}t = 1$ to System 1, while don't impose it on System 2. After that, both systems experience a same sequence of Brownian shock. We simulate the density transition paths with Monte Carlo simulation and investigate their difference.
- `MakeMovie.m`: makes 2D and 3D density diffusion movie for the distributional impulse response. (The default setting is starting from the stationary distribution).

with six functions

- `payoff_policy_growth.m`: iterates value functions with the backward method for the equilibrium.[1]
- `KFE.m`: solves time-invariant or time-dependent Kolmogorov forward equation with finite difference method.
  - Use `[pdf_stat] = KFE(X,MU,S)` to solve for time-invariant KFE, and it returns the stationary distribution `pdf_stat`;
  - Use `[pdf_stat,cdf] = KFE(X,MU,S,T,F0)` to solve for time-dependent KFE, and it returns stationary distribution `pdf_stat` and cumulative density `cdf` along the time grid `T`.
- `KFE_pdepe.m`: (alternative solver) solves time-dependent Kolmogorov forward equation using the MATLAB build-in solver `pdepe` for 1-D parabolic-elliptic PDE.
- `PercentileLine.m`: computes density percentiles along the shock and recovery path.
- `FanChart.m`: plots fan chart for the distributional impulse response.
- `FanChartPath.m`: plots fan chart for the distributional impulse response (difference to unshocked path). [2]

## 2 Implement

One can directly run `Lec3Main.m` to implement all functions, or go over the program section by section following the steps:

Step 0 Initialize parameters in `Lec3Main.m`.

Step 1 Set Eta grid, initialize variables, and guess the terminal condition of value functions in `Lec3_Main.m`.

[1] We thank Yuliy for providing this function.

[2] The input of `FanChartPath.m` is stimulated paths by Monte Carlo Simulation, while the input of `FanChart.m` is probability distribution function or cumulative distribution function getting from solving the Kolmogorov forward equation. `FanChartPath.m` is only used in `DistImpulseResponDiff.m`.

Step 2 Solve the model via the iterative method.

Once the model is solved, `BasicFig.m` will be called by main function `Lec3Main.m` to plot the solution. Then essential parameters $\eta, \sigma^\eta, \mu^\eta$ are save as `Eta_S_MU.mat` for further usage, and the workspace is cleared.

Step 3 Impose an impulse to the system and investigate its response, plot fan charts, and make density diffusion movie.

- `DistImpulseRespon.m` is called to simulate the shock and recovery path of a system starting from the stationary distribution. The 2D/3D density diffusion figures and fan charts are plotted, and the solution is saved as `solution_DIR.mat` in the end, which will be called later to make diffusion movies.
- `DistImpulseRespon_degen.m` is called to simulate the shock and recovery path of a system starting from a degenerate distribution.
- `DistImpulseResponDiff.m` is called to simulate the difference between shocked and unshocked paths.
- Finally, `MakeMovie.m` is called to make 2D/3D density diffusion movie.[3]

# 3 Compatibility

The optimal MATLAB platform version are `MATLAB 2019a` or later version. If you find compatibility issues with earlier versions, please let us know.

[3]Note it may take more than 20 minutes (depends on your machine), so we suggest skipping this step at your first attempt.