This is instruction for running codes of Guo et al. (2025) " **Modeling single-cell heterogeneity in signaling dynamics of macrophages reveals principles of information transmission** "

Note 1: Matlab codes are tested in Matlab 2024a (Mac)
Note 2: To execute each section, set the values in the "if" statements to 1.
Note 3: This script includes parallel programming. Please adjust the parameters to suit your server/computer configuration.
Note 4: please download the source data from here. Then change the path to the folders (for datasets, for saving figures) accordingly.
Note 5: The Monolix (the software for parameter estimation) files *.mlxtran are in the data filefoler, raw_data2023/SAEM_proj_2023/

Figure S1:
(1) run_me_publish_nc_2025.m
run section "parameter sens analysis" for both run the simulation and visualizing the results..

Figure 2:
(1) run_me_publish_nc.m
run section "rescaling the data to SI for monolix software input" to rescale the experimental data to S.I. units and change the data format to monolix software input format.
(2) run monolix codes in the following order:
(2-a) run XGESN2023001.mlxtran to estimate the parameter for the ODE parameter distribution in the core module.
(2-b) run XGESN2023002.mlxtran to estimate the parameter for the ODE parameter distribution in the TNF module, with the core module population distribution fixed.
(2-c) run XGESN2023003.mlxtran to estimate the parameter for the ODE parameter distribution in the LPS module, with the core module population distribution fixed.
(2-d) run XGESN2023004.mlxtran to estimate the parameter for the ODE parameter distribution in the CpG module, with the core module population distribution fixed.
(2-e) run XGESN2023005.mlxtran to estimate the parameter for the ODE parameter distribution in the PolyIC module, with the core module population distribution fixed.
(2-f) run XGESN2023006.mlxtran to estimate the parameter for the ODE parameter distribution in the Pam3CSK module, with the core module population distribution fixed.
(3) run_me_publish_nc.m
run section "calculate signaling codon" to calculate the signaling codons and save the results.
(4) run_me_publish_nc.m
run section "Heatmaps of traj, W-dist of signaling codon distri." to get Figure 2ABC
(5) to calculate the machine learning classification based on the signaling codons
(5-a) run_me_publish_nc.m
run section "codon prep for MI calculation & machine learning format" to get the machine learning format signaling codons dataset, which is the input for the classification algorithm.
(5-b) run "python_ML_classify_signalingcodon/cross_exp_fitting.ipynb" to to train the Random Forest model and make predictions to get the the confusion matrix.
(5-c) run_me_publish_nc.m

run section "stimulation classification" to get the results for machine learning classification results, to get Figure 2D

Figure S2:
(1) run_me_publish_nc.m
run section "metrics of good fitting, RMSD, signaling codon distri, and W-dist" to get figure S2A-E.
(2) to calculate the machine learning classification based on the signaling codons
(2-a) run_me_publish_nc.m
run section "codon prep for MI calculation & machine learning format" to get the machine learning format signaling codons dataset, which is the input for the classification algorithm.
(2-b) run "python_ML_classify_signalingcodon/cross_exp_fitting.ipynb" to to train the Random Forest classifier and make predictions to get the the confusion matrix.
(2-c) run_me_publish_nc.m
run section "stimulation classificatio" to get the results for machine learning classification results, to get Figure 2F-G

Figure 3:
(1) run_me_publish_nc.m
Run section "[simulation] doses study" to get the simulated NFKB trajectories under stimulation of extended 21 doses of five ligand.
(2) run_me_publish_nc.m
Run section "[visualization] for extended doses study of each ligand" to visualize the extended doses stimulation, for Figure 3A.
(3) run_me_publish_nc.m
Run section "codon prep for MI calculation & machine learning format" to transfer the extended doses signaling codons into Mutual information calculation format.
(4) MI_codes/run_me_MI_publication_nc.m
Run section "[MI calculation] doses study", to calculate the MI and visualization, to get Figure 3B.

Figure S3:
(1) run_me_publish_nc.m
Run section "[simulation] sampling single ligand" to get the simulated NFkB trajectories for five single ligand stimulation, and saved in "Sim5_codon_all5dose_metric.mat".
(2) run_me_publish_nc.m
Run section "[visualization] single-ligand sampling results" to visualize the sampling results for Figure S3A-C, S3E
(3) to calculate the machine learning classification based on the signaling codons from sampling results (Fgiure S3D)
(3-a) run_me_publish_nc.m
run section "codon prep for MI calculation & machine learning format" to get the machine learning format signaling codons dataset, which is the input for the classification algorithm.
(3-b) run "python_ML_classify_signalingcodon/cross_exp_fitting.ipynb" to to train the Random Forest model and make predictions to get the the confusion matrix.
(3-c) run_me_publish_nc.m

run section "stimulation classification" to get the results for machine learning classification results, to get Figure S3D

(4) run_me_publish_nc.m

Run section "[visualization] extended doses signaling codon distribution.m" to get Figure S3F.

(5) run_me_publish_nc.m

Run section "run extend dose sim to compare with Adelaja 2021" "quantitative assessing extend dose sim and experiments" to get Figure S3E-G

Figure 4

(1) run_me_publish_nc.m

Run section "[simulation] & [data preparation] MI loss within the network" to get sampled NFKB trajectories under different denoise strategies. And transfer the corresponding signaling codons into mutual information calculation format.

(2) MI_codesrun_me_MI_publication_nc.m

Run section "[MI calculation] denoise netowrk" to calculate the MI and visualization, to get Figure 4B-C, S4A

Run section "wt (wild type) vs Ikbo (IkBa mutant)", to get Figure 4G.

(3) run_me_publish_nc.m

Run section "[calculation & visualization] coefficient of variance of parameter distribution" to calculate the CV for different parameters and plot them, for Figure 4E.

Figure S4

(1) Figure S4A, refer to Figure 4, step (3)

Figure 5

(1) run_me_publish_nc.m

Run section "[Simulation] IkBamut compare with Ade2021 dataset" to get sampled NFKB trajectories for IkBa promoter mutant (Sjroen syndrome). And transfer the corresponding signaling codons into mutual information calculation format.

(2) run_me_publish_nc.m

Run section "ML classification: simulation and ade2021 exp IkBamut" to get the machine learning classifier results for simulation and experimental data (Adelaja 2021) comparison

Figure S5

(2) run_me_publish_nc.m

Run section "ML classification: simulation and ade2021 exp IkBamut" to get the heatmaps of trajectories and signaling codon distribution for WT and IkBa promoter mutant sample dataset, to get Figure S5A-B

Figure 6 & S6:

(1) run_me_publish_nc.m

Run section "[simulation] single cell specificity" to get the simulated NFkB trajectories for five single-ligand stimulation for each single cell and saved in "Sim8_5_signle_ligand_codon_metric_r3.mat" for wild type and in "Sim16_IkBao_matching_5_signle_ligand_codon_metric_p25x_r1.mat" for IkBa promoter mutant.

(2) run_me_publish_nc.m
Run section "[visualization] draw single cell specificity" to calculate and visualize the single-cell stimulus specificity results from the simulated trajectories, for Figure 6 & Figure S6

Figure 7 & S7
(1) run_me_publish_nc.m
Run section [visualization] draw single cell confusion" to calculate and visualize the single-cell stimulus confusion results from the simulated trajectories, for Figure 7 & Figure S7