CONTENTS

This directory contains the following folders:

1) Figure 1 – Code to fit median-effect equation to experimentally measure dose-response curve to estimate IC50 and m values.

2) Figure 3 – Codes to predict plasma dilution curves and estimate NT50 values.

3) Figure 4 – Codes to predict SAR-CoV-2 viral dynamics and estimate viral dynamic parameters.

4) Figure 5 – Codes to predict the efficacies of COVID-19 vaccines

REQUIREMENTS

The codes can be run in MATLAB or Monolix. The MATLAB codes were compiled using MATLAB R2017b. The Monolix codes were run on Monolix 2020R1.

USAGE

1) Figure 1/Figure1A_dose response curves_data fitting/:

A) Data_BD236.xlsx - This excel sheet contains the experimentally measured dose-response curve of BD-236 antibody

B) Median_effect_equation.m - This code is to fit the median-effect equation to the experimental data.
Instructions: Change the line: "data = xlsread('Data_BD236.xlsx')" for the code to read the desired dataset in '.xlsx' format and estimate the dose-response curve parameters IC_{50} and *m*.

2) Figure3/ Figure3B_plasma dilution_data fitting/:

A) Master_plot_plasma.xlsx - This excel sheet contains the raw data (plasma dilution curve of three patients).

B) Plasma_dilution_fits.m - This code fits the standard dose-response curve to the data
Instructions: Change the line: "Name = 'Master_plot_plasma.xlsx'" for the code to fit the dose-response curve to the desired plasma dilution dataset in '.xlsx' format.

3) Figure 3/ Figure3B_3C_predict plasma dilution_Bliss and Loewe/:

A) Predict_Plasmadilution_Loewe.m - This code computes the plasma dilution curve using Loewe additivity.

B) Predict_Plasmadilution_Bliss.m - This code computes the plasma dilution curve using Bliss Independence.

Instructions: In the above, change the values of N, D0, IC50, or m to predict the plasma dilution curve for the desired parameter values.

C) Fig_CI_30.mat: Plasma dilution curves simulations assuming Loewe additivity and overall NAb concentration of 30 μg/ml.

D) Plotting_Fig3B.m - This code plots plasma dilution curves saved in Fig_CI_30.mat (grey lines), as well as fits to the patient data (blue lines). See Fig 3C in the manuscript.

E) Computing_NT50.m - This code computes NT50 of all the predicted plasma dilution curve. This code was used to predict Fig. 3C.

Instructions: Change the line: "Data = importdata('Fig_CI_30.mat')" for the code to estimate NT50 values of the desired dataset in '.mat' format.

4) Figure4/ Figure4B/:

A) Neant_Kissler_Wolfel_input_monolix.csv - The digitized viral dynamics data from three papers that is input into monolix.

Instructions: Change this file to fit the model to the desired dataset.

B) vdm.txt - The viral dynamics model written in mlx format.

C) run.mlxtran - The main Monolix format that internally calls both the files above and runs NLME fitting.

Instructions: Load the above files into Monolix to perform fitting.

D) run - The sub-folder that is the actual Monolix output which was used in the manuscript. Once Monolix converges with respect to population distribution estimates of fit parameters, multiple individuals who are sampled from the distributions are simulated (in the sub-folder '\run\IndividualParameters').

E) Sel_Indiv_Params.csv – This file compiles the list of individuals whose parameters when input into the viral dynamics model best-captured individual clinical data from 'Neant_Kissler_Wolfel_input_monolix.csv' (Figure S7 in the paper).

5) Figure4/ Figure4C/:

A) Viral_dynamics_Fig4C.m - This code predicts SARS-CoV-2 viral dynamics for different efficacy values as shown in Fig. 4B.

Instructions: Change the values of 'epi_CI' to predict viral dynamics for the desired efficacy. Changes the viral dynamic parameters on lines 10-18 to predict viral dynamics for the desired parameter values.

6) Figure4/ Figure4D/:

A) m_IC50_2.mat - This file contains NAbs with distinct m and IC50 values sampled from the landscape in Fig. 2A.

B) Viral_dynamics_compute_peak_viral_load_Fig4C.m - For different NAb combinations sampled from the landscape and for different viral dynamics parameters, this code predicts how the peak viral load varies for a fixed overall antibody concentration, D_0, as shown in Fig 4C.

Instructions: Change the parameter values on lines 61-70 to predict viral dynamics for the desired parameter values.

C) V_0.01.mat - This file is created by running the 'Viral_dynamics_compute_peak_viral_load_Fig4C.m' code. This file has viral dynamics of all 500 virtual patients at D0 = 0.01 μg/ml

7) Figure5:
(Run the codes in the following order. Note that there will run-to-run variability due to stochasticity.)

1) A_Model_Protection_final.m - This code will generate a file 'V_varyFixedparms_sample2_v1.mat' containing 10,000 virtual patients.

Instructions: Change the parameter values on lines 60-66 to predict viral dynamics for the desired parameter values.

2) *B_scaling_Model_Protection_final.m* - This code generates a file 'V_varyFixedparms_sample2_v1_Scale_10.mat' containing 10,000 virtual patients. The IC50 values are scaled with omega.

Instructions: Change the values of 'Scale' to predict the protection curve for different omega values.

3) *C_ModelPredict_Protection_final.m* - This code computes protection as a function of scaled NT50, as shown in Figure 5.

REFERENCE

Pranesh Padmanabhan, Rajat Desikan, and Narendra M. Dixit. Modeling how antibody responses may determine the efficacy of COVID-19 vaccines. Nature Computational Science 2021.