Manual for the Eject! code in Matlab

To use the model, download all .m-files and the excel file containing topography data, and store them in one folder on your computer. In the following, we will shortly describe how to use the different functions.

eject.m

This function calculates the trajectory of a clast, given the input parameters thetadeg (ejection angle in degrees), vi (initial velocity) and diam (diameter of the clast). Please note that, so far, only the shape option "high cube" is available (see Mastin, 2001; 2011). You can use this function directly in your command window or call it in a different script. You can/need to also adjust various parameters used by the function that are defined within the eject.m-script. Currently, some of them are adapted to the case of Ruapehu volcano, namely the temperature at sea level and the density of ballistic clasts. The *eject* function in turn uses the functions *draghicube.m, rk.m, splint.m, spline.m* and *derivs.m.* You do not have to change these to use *eject* on different volcanoes than Ruapehu.

distances.m

This function is used to compare flight trajectories to the topography of the volcano in order to determine the flight distance of ballistic clasts. You can calculate the flight distance for any number of clasts at the same time. A single trajectory consists of x- (horizontal distance from ejection point) and corresponding z-values (elevation of the clast at point x). The input for the *distances* function is x (matrix with x-values, where each row represents one clast), z_ballistic (matrix with z-values, where each row represents one clast) and direction (column vector with one value for each clast). We decided to distinguish between four directions in which the clast can fly and assigned a number to each direction: towards the north (1), south (2), east (3) and west (4). For each direction, an elevation profile of Ruapehu is stored in the excel file *Topography.xlsx*. This is imported into Matlab and, depending on the direction value, the trajectory is compared to the corresponding topography profile. You need to adapt this to your volcano, and also adjust the file path for the import in the *distances.m*-script to your own storage system.

For example, you could now write a script that calculates trajectories with *eject*, stores these in a matrix and then calls *distances* in order to get the flight distances. In a separate resource, we will describe how to utilise *eject* and *distances* in a probabilistic procedure (as used in Strehlow et al., 2017).

References:

Mastin, LG (2001) A simple calculator of ballistic trajectories for blocks ejected during volcanic eruptions: U.S. Geological Survey Open-File Report 01-45, 16p; https://pubs.usgs.gov/of/2001/0045/.

Mastin, L (2011) eject model. https://vhub.org/resources/455.

Strehlow, K, Sandri, L, Gottsmann, JH, Kilgour, G, Rust, AC, Tonini, R (2017) Phreatic eruptions at crater lakes: occurrence statistics and probabilistic hazard forecast. Journal of Applied Volcanology, 6 (4), DOI 10.1186/s13617-016-0053-2.