**Methods**

*arcpy Script*

The script takes as input a line shapefile and a DEM. The two must be in the same projected coordinate system, and the DEM must cover the full extent of the line. The user has the option to split the input line at its vertices or select a sampling distance for the input lines to be split at in increments. The script loads each line in the line shapefile into WKT geometry. If the user selects a sampling distance, the interpolate function is used to capture the spatial coordinates along the line in intervals of the sampling distance. Each point along the line at the specified sampling distance is added to an output line string and saved into a shapefile called LineSplit.shp.

A blue circle on a black background

Description automatically generated

When sampling along the line at a given distance, there will be some discrepancies in the line split if a vertex is present within the interval. This can result in some line segments that are shorter than the sampling distance. When tested on a 8724.45-meter jogging-pace GPS track with a frequency of position capture every 1 second and a sampling distance of 0.5 meters, this resulted in 17,449 segments. Of these segments, 1,492, or 8.6% were calculated at less than 0.499 meters in length. Nine of the segments, or less than 0.1%, were shorter than 0.4 meters.

With the input line split into segments, a field is added to store the bearing of each segment. The line bearing is calculated using the arcpy function to Calculate Geometry Attributes. The Feature to Raster tool is applied to convert the bearing data to raster. The projection of this bearing raster is set to the same coordinate system as that of the inputs. The cell size is set to the same as input DEM, and the snap raster, mask, and extent are also set to the same as the input DEM.