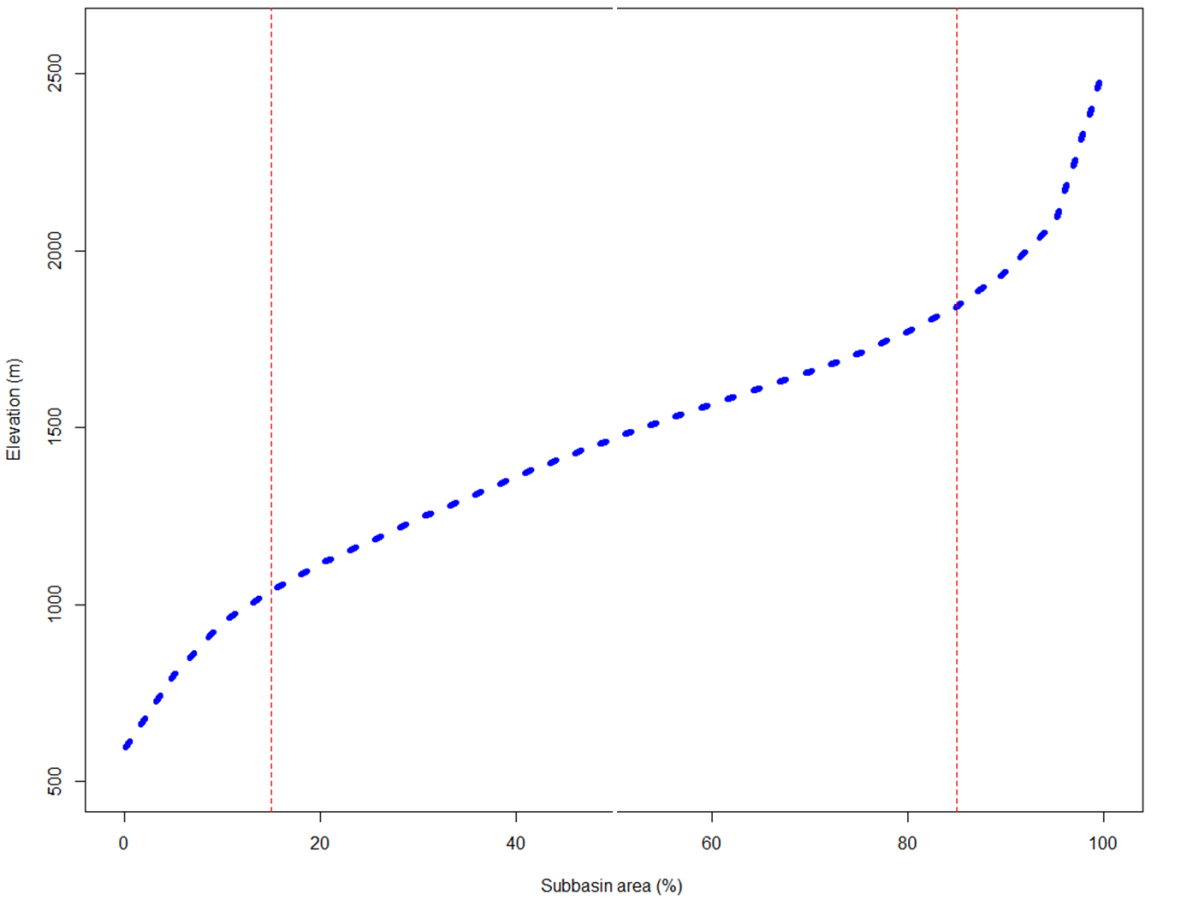
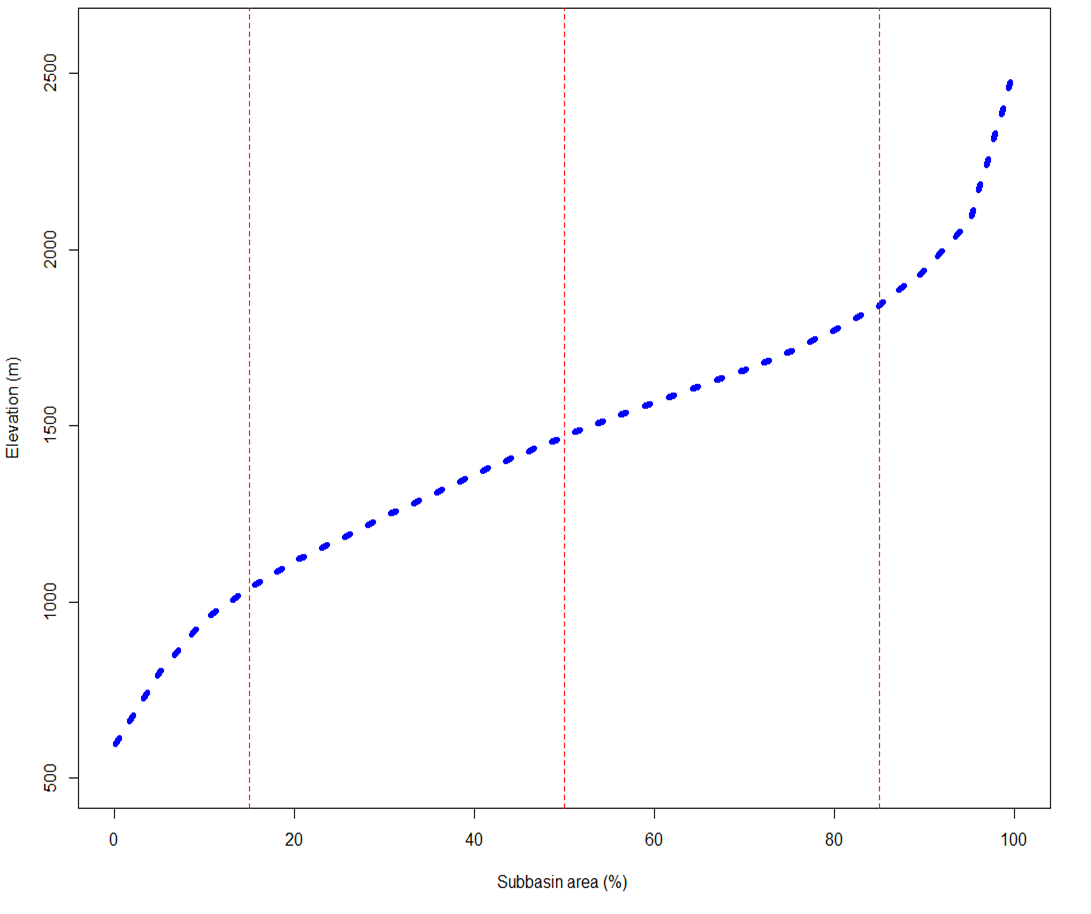
**A Local Topographic Classification Method to Derive Subgrid Land Units for the ACME Land Surface Model**

Following Tesfa et al. (2014), which demonstrated improved scalability using subbasin-based land surface modeling approach compared to the standard rectangular grid-based approach, the ACME land model will adopt watersheds/subbasins as the basic computational units. For more realistic representation of orographic effects on the hydrological cycle, each subbasin will be further divided into multiple subgrid units (land units) based on topographic properties such as elevation, slope and aspect. Exploratory analysis of elevation-area profiles of subbasins shows three distinct elevation bands: (1) lower fast rising, (2) middle slow rising, and (3) higher fast rising elevation bands (see Figure 1). This enables us to develop a local land unit derivation method to discretize each subbasin into distinct subgrid units (land units) for the ACME land surface model.

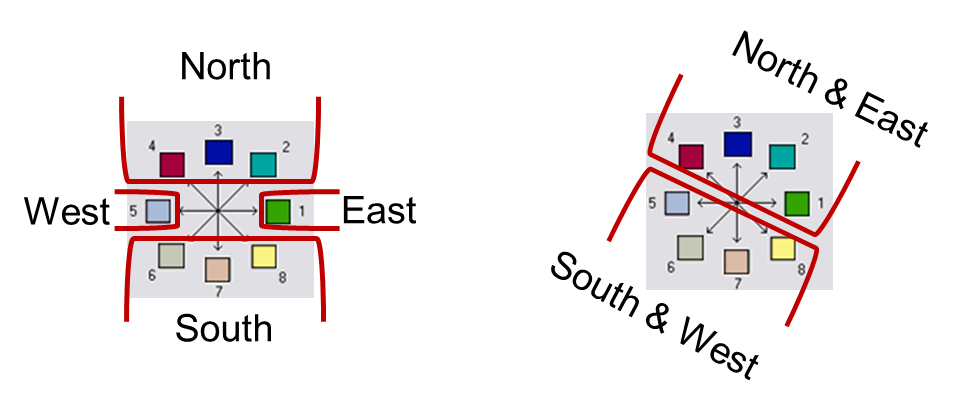
****

**Figure 1**: Elevation-area profile plot showing the three distinct elevation bands (lower fast rising, middle slow rising, and higher fast rising).

The local land unit derivation method involves: (1) derivation of elevation-area profile for each subbasin using elevation values extracted from a Digital Elevation Model (DEM), (2) dividing the subbasin into distinct elevation classes using the pattern of the corresponding elevation-area profile as shown in Figure 2, (3) classifying the subbasin into two topographic aspect classes (Figure 3) (south and west facing and north and east facing slopes), (4) combining the classes resulted from elevation and aspect classification, and (5) merging all classes covering less than certain area threshold value into the neighboring classes to derive the final land units.



**Figure 2:** An example of initial elevation zones generated based on the pattern of elevation-area profile of a subbasin.



**Figure 3:** Classification of topographic aspect

More specifically, in this method, elevation values are first extracted for each subbasin from a 90 meter resolution DEM and used to derive an elevation-area profile. The subbasin is then discretized into four initial elevation classes using the minimum elevation, maximum elevation, and the 15, 50, and 85 percentiles as class breaks, which are obtained from the corresponding elevation-area profile (Figure 2). These initial elevation classes identify the areas representing the lower fast rising, two middle slow rising, and one higher fast rising elevation bands. To minimize the number of land units in subbasins with flat topography, a minimum of 100 m elevation range criterion is applied for each elevation band, where if an elevation band has less than 100 m elevation range, the band is merged to the neighboring elevation band reducing the number of elevation classes per subbasin to three. Algorithm 1 summarizes the method implemented to derive topographic elevation land units using the local land unit derivation method. The algorithm assures that each elevation band has minimum elevation range of 100 meters and the number of elevation bands for each subbasin is less than or equal to four. The land units resulted from the elevation classification algorithm (Algorithm 1) are combined with the classes of topographic aspect of the corresponding subbasin. Finally, small land units covering less than five percent of the subbasin area are merged to the neighboring land units to derive the final land unit dataset. The method has been implemented in Python using ArcGIS functions.

**Algorithm 1**

*For each Subbasin* ***S****:*

*Generate Elevation-area profile curve Get minimum, maximum, and 15, 50, and 85 percentile elevation values as initial elevation band threshold values (****TV****)*

*Calculate elevation ranges (****ER****) between each consecutive* ***TVs*** *For each elevation range (****ER****):*

*If* ***ER*** *< 100 m:*

*Combine class to next or previous class and update the corresponding* ***TVs***

*Determine final class thresholds*

*Classify* ***S*** *into elevation bands based on the final threshold values*