Leibniz-Universität Hannover
Institut für Kartographie & Geoinformatik
Prof. Dr. Martin Werner

Big Geospatial Data (Summer Term 2017)
Tutorial 3

Aufgabe 7: Point cloud processing with Hadoop Streaming

Points clouds, e.g. recorded with a LiDAR mobile mapping system, often contain several millions or even billions of points. Thus, a distributed processing is benefitial. In this exercise, we want to implement a simple point cloud processing to identify elevated objects like trees and reduce the size of the point cloud using a grid.

- Run the virtual machine image from the USB stick or provided on the lecture webpage *http://martinwerner.de/big_geospatial_data/index.html*
- Make yourself familiar with the Hadoop Streaming API following the steps 1 and 2 from the README.html under /home/bgd/hadoop/markt-hadoop-streaming/
- Compute the mean height of the points in the point cloud *marktplatz.dat* using Hadoop MapReduce.
  Hint: Characters can be removed from a string in Python using *translate*
  Example for ! and ?: stringVar.translate(None, '!?')
- Classify the points based on their height in a *map* step: if the Z value is higher than 110m, emit a new pair with key *high*. If the Z value is higher than the mean, but lower than 110m, assign the key *tree*. In the *reduce* step, consider only the *tree* points and assign the points to the grid in *grid.dat* based on their distance. Emit only the first point per grid center (see also step 3 in README.html).

  Useful commands:

  - Read text file: `numpy.genfromtxt(filename, delimiter = ' ')`
  - Build point: `numpy.array([[float(x), float(y)]])`
  - Compute all distances point to list: `scipy.spatial.distance.cdist(list, point)`