Generating LED from NHN 1:50,000

Created: January 29, 2015

This document provides step-by-step instructions for creating Lake-Edge Density (LED) criterion layer from NHN 1:50,000 scale. NHN 1:50,000 is the hydrographic layer of the GeoBase and can be downloaded from the ftp site. The script presented here allow to download the NHN data according to a specified studyregion, extract the zip features of interest (waterbodies and islands) and generate LED from them using ArcPY geoprocessing tools. The script can either work using Python IDLE or the Python editor for Windows PythonWin. To load your script into Python IDLE, right click on the .py script and select “Edit using IDLE”. Make sure the parameters setting is properly filled (see instruction below). When the parameters have been filled and the script is ready to be launched, go to the Run menu and select Run Module command.

PythonWIN is a Windows IDE and GUI framework for Python. It has an [integrated debugger](http://docs.activestate.com/activepython/2.5/pywin32/html/pythonwin/doc/debugger/index.html) and a rich Python editing environment. PythonWin can be downloaded from <http://starship.python.net/crew/mhammond/win32/Downloads.html>. A shortcut can be created during the installation process to add “Edit using PythonWin” in the right clic menu. The same attention should be put on setting the parameters properly before launching the script.

## SETTING UP PARAMETERS

**Step 1 - Setting up project directories**

The first step in the preparation of the criteria that will serve in the representation analysis is to create a project directory along with sub-directories that will contain input data and the Python script. Note that the output directory and sub-directories are created by the script.

- StudyRegionName

- index

- output

- proj

- shp

The “index” folder contains the workunits index available on GeodataBase website: <http://www.geobase.ca/geobase/en/data/nhn/national-index.html>. The use of the index will limit the processing extent. The“output” folder will contains all resulting output. The “proj” folder contains the projection file used in the resulting output. To create such a file, you need to copy the .prj file from a shapefile that uses the proper projection, paste it into the proj directory and rename the file to access in subsequent step. The “shp” folder contains the shapefile delineating the study region.

**Step 2 – Specifying the location of the catchments and criteria**

The second step is to specify the path where the script is and derived the location of sub-directories. To derive the location of sub-directories, the script should be stored in the StudyRegionName folder.

# Extract the working directory defined by the location of the script.

currentDir = os.getcwd()

## Location NHN index workunits shapefile

NHNindex = currentDir + "/index/NHN\_INDEX\_WORKUNIT.shp"

## Location Study Region shapefile

studyregion = currentDir + "/shp/NWBLCC\_AoA\_fda\_diss.shp"

## Location output coordinate system

outCS = currentDir + "/proj/NWB\_LCC\_proj.prj"

**Step 3 – Specifying the parameters to access the GeoBase ftp site.**

A gdb has more flexible properties and settings than a shapefile. It allows to control more parameters which is the reason why we use in the intermediate step between the input and output file. A gdb output name and the unique catchment identifier from the catchments input table need to be defined prior to run the script.

####################################

## SET FTP PARAMETERS

####################################

# Geobase FTP site address

ftpSite = "ftp2.cits.rncan.gc.ca"

# Specify the location of the GDB. You can change it if you want other GIS format

ftpDir = "/pub/geobase/official/nhn\_rhn/gdb\_en/"

# Logon to the FTP site

ftp = FTP(ftpSite)

ftp.login('anonymous', 'beacons')

**Step 4 – Specifying the LED parameters.**

Lake-edge density (LED) quantifies the linear density of terrestrial/aquatic edge and represents the abundance of habitat along large waterbodies (lakes and wide rivers). LED units are measured in km/km2 within a define km2 circular moving window. Prior to generate LED, the cell resolution of the output, the search radius for the moving windows and the final output name need to be set.

####################################

## SET LED PARAMETERS

####################################

## Set a buffer around the study region to allow the complete calculation of circular moving windows on cell located on the edge of the study region.

buffer = 3000

## Set the cell size resolution of the output file

cellRes = 250

## Set the search radius of the circular moving window

searchRadius = 2820.94461 # within 25ha

## Set LED output name

outLED = "led\_250\_25.tif"

## GENERATING REPRESENTATION ANALYSIS FUNCTIONS

ALL FUNCTION USED IN THIS SCRIPT ARE HARDCODED. THE PARAMETERS ARE TAKEN FROM THE PARAMETERS SETTING SECTION SPECIFIED IN STEP 1 TO 6. NOTHING SHOULD BE CHANGED IN THE FUNCTIONS WHILE RUNNING THE SCRIPT.

**Step 5 – Preparing the study region.**

In step 5, the prepStudyRegion function buffers the study region to make sure the extracting of NHN based on the study region will be sufficient to allow a full coverage for the circular moving window. The function also converts the study region to raster to create an anchor point and fix the cell size that will be use later in the creation of LED. The function returns the converted study region object that will be used in subsequent step.

rastExtent = prepStudyRegion(studyregion,str(buffer),tempOutputDir,cellRes,outCS)

**Step 6 – Downloading NHN 1:50,000 from Geobase ftpsite**

In step 6, the downloadZipNHN function intersects the study region with the NHN workunits index to identify a list of workunits intersecting the study region. The list is then use to download those workunits and save time processing.

downloadZipNHN(NHNindex,tempOutputDir,studyregion)

prepCatch(catchments,tempOutputDir,indicatorsGDB,n,method,outCatch)

**Step 7 – Unzipping gdb workunits**

In step 7, the unzipNHN function unzips all zip gdb downloaded from the Geobase ftp site.

unzipNHN(zipFileOutputDir,unzipOutputDir)

**Step 8 – Generating reclassified criterion SHP**

In step 8, the genWaterlineFromNHN function loops through unzip gdb workunits, extracts the features of interest (waterbodies and island), converts them into line and merges them together to create an overall waterline features needed to derive the lake-edge density. The function returns the waterline objet, which correspond to the location of the merge waterline feature.

waterLine = genWaterlineFromNHN(unzipOutputDir,tempOutputDir)

**Step 9 – Tabulating categorical criterion per catchment**

In step 9, the genLED function generates the LED criterion and clips it to the study region. The function uses cell resolution and search radius set at the beginning of the script, plus the two objects (rastExtent and waterLine) created in previous step.

genLED(waterLine,outLED,rastExtent,cellRes,searchRadius)