POSTAL LOCATION "NEAREST-NEIGHBOR" CALCULATION AND RESULTS

I. PREFACE

I-A. Purpose:

USPS-LR-N2011-1/4 describes a technique for ascertaining the nearest neighbor to a postal location (based on latitude/longitude data available in the USPS Enterprise Data Warehouse (EDW). This analysis focused on Locations that accrued less than $100,000 in Walk-In Revenue (WIR) in FY 2010; these will be referred to hereafter as "LT100K". For purposes of this library reference, a "location" is a Post Office, station, branch, or contract postal unit (CPU), each of which is collectively referred herein as a "Location", and a "neighbor" consists of the nearest Post Office, station, branch, or contract postal unit.

I-B. Predecessor Documents:

There are no predecessor documents to USPS-LR-N2011-1/4.

I-C. Corresponding Non-Public or Public Document.

There are no corresponding public or non-public documents to USPS-LR-N2011-1/4.

I-D. Methodology:

1. Cleansed (removed missing or NULL data) available latitude/longitude data for all Locations, and ran a Cartesian product algorithm (described below) to identify the nearest neighbor for each Location.

I-E. Inputs/Outputs:

The following inputs are used for this process:

- Excel spreadsheet containing Unit Finance Numbers, latitudes and longitudes of 13,494 retail locations with FY2010 WIR <$100K for which latitude and longitude data was available. There were 16,312 total retail locations with FY2010 WIR <$100k, and 2,818 of these did not have latitude and longitude data available, so these 2,818 locations were excluded from the analysis.
- Excel spreadsheet identifying all LT100K Locations.

The following outputs are created as a result of this process:

- Excel spreadsheet containing Unit Finance Number, “nearest neighbor” Unit Finance Number, distance in miles, and “LT100K” indicator for all Locations for which latitude/longitude data were available.

II. ORGANIZATION

USPS-LR-N2011-1/4 consists of:

- 4 Excel Workbooks with six Excel files (4 input, 1 output, 1 reference), and
- SAS Program documentation with `.sas` files for SAS program codes.

USPS-LR-N2011-1/4 Excel Workbooks comprise 1 worksheet with latitude/longitude data and 3 worksheets that identify LT100K Locations stratified by retail reporting technology (POS, IRT or eMOVES). The final worksheet is the output of the process, as described above in section I-E.

USPS-LR-N2011-1/4 SAS documentation includes a general description of the programming procedures associated with the SAS program codes, as indicated below under section III.B. *USPS-LR-N2011-1/4 SAS programs* contains `.sas` files of the SAS program codes.

III. DOCUMENTATION

III.A. Table Of Contents for USPS-LR-N2011-1/4 Excel Workbooks.

Excel spreadsheet “FDB CMN Core.xlsx” contains one tab, “FDB_CMN_CORE”, which contains Unit Finance Numbers and latitude/longitude data retrieved from EDW. The original table from which these data were extracted is EDWFcltyProdView.FDB_Cmn_Core.

Excel spreadsheet “FY2010 Offices under $100,000 by Technology.xls” contains three tabs, “POS”, “IRT”, and “Emoves (Manual)” which collectively identified LT100K Locations across the three reporting technologies.

Excel spreadsheet “Nearest neighbor data.xls” contains the output of the process, directly from SAS.

Excel spreadsheet “Nearest neighbor graph data.xls” is built from “Nearest neighbor data.xls”; Location names and other data have been added on one of the tabs to assist in verification. This spreadsheet was used to prepare the “quarter-circle” graphic that visually summarized these results in the powerpoint slide USPS-LR-N2011-1_4.ppt.

III. B. SAS Program Documentation for USPS-LR-N2011-1/4

Programs are developed in SAS for processing on a PC or server.

1. General Objective:

The SAS program provides the entire execution stream of the process, as described in detail below and in the comments within the code itself:

2. General Programming Structure:

Step0 *Import latitude/longitude data was retrieved from the EDW table, EDWFcltyProdView.FDB_Cmn_Core, as well as the three spreadsheet tabs with Small post offices identified.*

The SAS program imported the latitude/longitude data, which was then sanitized and put into SAS table `fdb_cmn_core`.

Next, the three worksheets that identify LT100K Locations were imported. These were placed into SAS tables `pos_small`, `irt_small`, and `emoves_small`, and then concatenated into SAS table `small_post_offices`.

Step1 *Converted latitude/longitude data into radians, as these are necessary for use of SAS trigonometric functions. Any post offices with missing latitude/longitude data were deleted.*

Step2 *Made two copies of the latitude/longitude data for use in the Cartesian product step (Step4 below).*

Step3 *Ran the N(N-1) product (Cartesian, omitting 'diagonal' entries) routine to figure out which neighbors exist, and how close they are. Since the latitude/longitude data are occasionally suspect, omitted any 'neighbors' with a distance of less than 1/10-th of a mile, which is 176 yards (= 1760/10). This sometimes happens when Locations are co-located, or (as alluded to above) one or both of the latitude/longitude data points is not entirely accurate.*

3.1 Used one copy of the latitude/longitude data as the driver table, over which the master loop was run.

3.2 Looped over the other copy of the latitude/longitude data as the inner, or slave, loop.

3.3 For each pair of latitudes/longitudes where the unit finance numbers were not identical, computed the great circle distance (GCD) using standard

spherical geometry and a standard Earth radius (this is the average of the equatorial and the polar radius of the Earth, which differ by approximately 1%) to calculate arc distance between the pair of points. Kept track of the smallest distance calculated through the entire inner loop, and the corresponding unit finance number of the second Location. The mathematics of this calculation are described in section 3 below.

3.3 At the bottom of the inner loop, output the nearest neighbor identified.

Step4 *Marked LT100K Locations.*

4.1 Using the list of LT100K Locations identified, performed a simple table join to mark those unit finance numbers that are LT100K.

Step5 *Exported results to Excel.*

3. General Methods and Procedures Employed :

Programs were developed in SAS for processing on a PC or server.

The algorithm to calculate the distance between two Postal Locations utilized the *Great Circle Distance* formula. The formula is based upon getting the angle subtended by the two Locations on an idealized Earth's surface (a sphere); once this angle is computed, the arc length is (trivially) the angle multiplied by the radius of the arc – in this case, the average radius of the Earth in miles.

The underlying algorithm to calculate the GCD is as follows:

To convert a value in degrees (X) to a value in radians (R), use the formula

$R = X * (2\pi / 360)$

Let (l_1, f_1) and (l_2, f_2) be the (latitude, longitude) pairs of two Locations, *already converted to radians*, as above.

Let $\Delta f = f_1 - f_2$ be the difference between the two longitudes. Then a standard result of spherical geometry is that the angle θ that is subtended (or spanned) by the two points is

$\theta = \cos^{-1}(\sin(l_1)\sin(l_2) + \cos(l_1)\cos(l_2)\cos(\Delta f))$

where functions $\cos^{-1}$, sin and cos are standard trigonometric functions.

Now let r be the radius of the Earth in miles. Choose an average value, since equatorial and polar radii differ by a small amount. In the case of this process, r = 3959.9 miles. Then the great-circle distance, which is the arc-length along the surface of the Earth, is

GCD = r*q

4. List of SAS Programs:

The only SAS program involved in this process is called "`Facilities Nearest Neighbors.sas`".