## Introduction

In order to facilitate geocoding of protected health information (PHI) and personally identifiable information (PII) for research purposes, our team has implemented a version of DEGAUSS in the Minerva HPC environment. At this time, using DeGAUSS or geocoding using an off-line geocoder on a Mount Sinai managed computer system are the only options for geocoding PHI/PII. Online geocoders, like Google Maps, the US Census API, OpenStreetMaps, and local software (like ArcGIS) that utilizes the internet to geocode are not allowed due to potential exposures of PHI/PII.

DeGAUSS is a command-line tool that is run while connected to the Minerva HPC environment (a HIPAA compliant computing environment operated by the Icahn School of Medicine at Mount Sinai). Researchers will submit a small csv dataset, containing a unique linkable variable and the address to be geocoded. Output from this initial process is the initial dataset with geocoded variables appended including match quality metrics (match score, geographic level of match) and latitude/longitude values when the match is of sufficient quality. Additionally, a second stage process can be performed to append geomarker information to the data. Census block groups and census tracts are two of the most common geomarkers, but deprivation index data, road proximity, daily traffic data, and vegetation indexes are also available. Information on the available geomarkers can be found here (https://degauss.org/available_images.html). Let Dave Buckler and Alexis Zebrowski know if there are specific images you want or need.

## About DEGAUSS

DeGAUSS (Decentralized Geomarker Assessment for Multi-Site Studies) is a decentralized method for geocoding and deriving community and individual level environmental characteristics while maintaining the privacy of protected health information. It is a standalone and versatile software application based on containerization. This means that Geomarker assessment is reproducible, standardized, and can be computed on at scale. Importantly, DeGAUSS is executable on a local machine – it does not require extensive computational resources and PHI is never exposed to a third party or the internet, making it ideal for Geomarker assessment in a multi-site study.

## Preparing the data

Addresses must be stored as a CSV file and follow these formatting requirements:

- Other columns may be present, but it is recommended to only include address and an optional identifier column (e.g., **id**). Fewer columns will increase geocoding speed.
- Address data must be in one column called **address**.
- Separate the different address components with a space
- Do not include apartment numbers or "second address line" (but its okay if you can't remove them)
- ZIP codes must be five digits (i.e. 32709) and not "plus four" (i.e. 32709-0000)
- Do not try to geocode addresses without a valid 5 digit zip code; this is used by the geocoder to complete its initial searches and if attempted, it will likely return incorrect matches
- Spelling should be as accurate as possible, but the program does complete "fuzzy matching" so an exact match is not necessary
- Capitalization does not affect results
- Abbreviations may be used (i.e. St. instead of Street or OH instead of Ohio)
- Use Arabic numerals instead of written numbers (i.e. 13 instead of thirteen)
- Address strings with out of order items could return NA (i.e. 3333 Burnet Ave Cincinnati 45229 OH)

## Transferring to Minerva (if necessary)

- If the data already lives in a project folder within the Minerva HPC, we recommend creating a 'geocoding' subfolder to contain the files generated in this process.
- If the data is not on Minerva, please use Globus to securely transfer the data to a project folder on the server. More information on how to setup and transfer data with Globus can be found here: https://labs.icahn.mssm.edu/minervalab/documentation/services/globus-high-assurance-hipaa-file-manager/

## Geocoding

1. Connect to the Minerva HPC environment using SSH

```
ssh -Y username@minerva.hpc.mssm.edu
```

2. Navigate to the geocoding subfolder within your project folder

```
cd /sc/arion/projects/emhsr/emhsr-ehr/projects/yourProjectFolder/geocoding
```

3. Make sure you are in the same folder as your address list before running the geocoder script

```
ls -a
```

4. Run the geocoder script. The last argument in the command below if the name of your csv file containing the addresses to be geocoded. The addresses must be in one column named 'address'

```
sh /sc/arion/projects/emhsr/emhsr_general/dataLibrary/geocoder/geocode.sh addressFile.csv
```

5. When geocoding has completed, you will see a summary indicating the percentage records successfully geocoded. The geocoded results will be in a file with name structure: *addressFile*_geocoded_*versionNumber*.csv

## Geomarker Assessment

- If you want to append Geomarker ids, like census tracts or block groups, you will need to run an additional step where you feed the result file from the previous step into another geocoding process outlined below.
- If more than 500 records did not geocode, you have to filter out those records prior to Geomarker Assessment. This can be done using statistical software like R or Python.
- Geomarker assessment can also be perfomed on other data, as long as the file to be assessed has 2 columns, lat and lon, with valid numerical values.

6. If you are not already there, navigate to the geocoding subfolder in you project folder
7. Run the script file for Geomarker you want to assess. The last 2 arguments are the gocoded address file 9from the previous process) and the Census Year you want the Geomarkers from. For block group data the options are 2020, 2010, 2000 or 1990.

   ```
   sh /sc/arion/projects/emhsr/emhsr_general/dataLibrary/geocoder/geocode_blockgroup.sh
   geocodedAddressFile.csv 2010
   ```

8. Review and interpret the results of your geocoding.

Notes from experience:

- Excel will truncate block group and census tract id numbers. Read the csv files directly into your statistical analytic software.

## Interpreting geocoding results

The geocoder's output file includes the following columns:

- matched_street, matched_city, matched_state, matched_zip: matched address componets (e.g., matched_street is the street the geocoder matched with the input address); can be used to investigate input address misspellings, typos, etc.
- precision: The qualitative precision of the geocode. The value will be one of:
  - range: interpolated based on address ranges from street segments
  - street: center of the matched street
  - intersection: intersection of two streets
  - zip: centroid of the matched zip code
  - city: centroid of the matched city
- score: The percentage of text match between the given address and the geocoded result, expressed as a number between 0 and 1. A higher score indicates a closer match. Note that each score is relative within a precision method (i.e. a score of 0.8 with a precision of rangeis not the same as a score of 0.8 with a precision of street).
- lat and lon: geocoded coordinates for matched address
- geocode_result: A qualitative summary of the geocoding result. The value will be one of
- po_box: the address was not geocoded because it is a PO Box
- cincy_inst_foster_addr: the address was not geocoded because it is a known institutional address, not a residential address
- non_address_text: the address was not geocoded because it was blank or listed as "foreign", "verify", or "unknown"
- imprecise_geocode: the address was geocoded, but results were suppressed because the precision was intersection, zip, or city and/or the score was less than 0.5.
- geocoded: the address was geocoded with a precision of either range or street and a score of 0.5 or greater.

### Missing geocoding results

Geocodes with a resulting precision of intersection, zip, or city are returned with a missing lat and lon because they are likely too inaccurate and/or too imprecise to be used for further analysis.