

# OSM data analysis and visualization for beginners

Martijn van Exel for Mapping USA 2024

# setting the stage

I expect you to have very little or no knowledge of OSM data, visualization techniques, or data analysis.

Some GIS knowledge and understanding of basic GIS concepts is helpful.

For the bonus part of this workshop: Experience with Python, Pandas, Jupyter notebooks is helpful. I'll add bonus resources though!

*There's always more than one way to do things. I will show you **a** way but not **the** way. A big part of the fun in spatial data analysis is exploring different paths.*

# About OpenStreetMap & OSM data

- community driven map of the world
- 40,000 individuals edit the map every month
- freely available

to keep in mind

- Each mapper brings their own experience (or lack thereof) and background
- There is no fixed data model
- OSM data not public domain (!)

# OSM data

OSM has

almost 9 billion **points** (nodes)

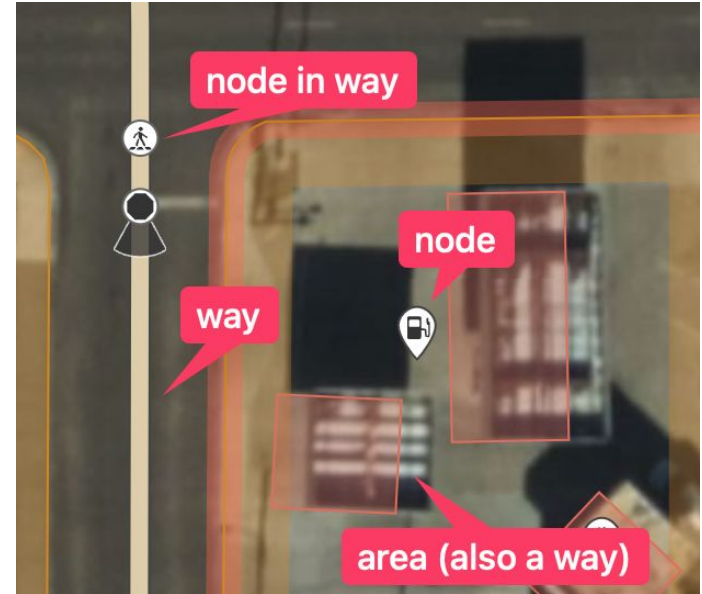
almost a billion **lines** (ways)

Every day, mappers create

2.5 million **new** nodes

250,000 **new** ways

*More than 100 million things change in OSM each month*



metrics about OSM data:

[OSM Stats](#), [Taginfo](#)

# Data from osm.org

The OpenStreetMap project itself makes map data available in a few ways:

First , the **planet** file. This is the entirety of OSM data. It comes in XML and PBF flavors.

Second, **changesets**. This is data about the changes made to OSM: who made the edit, when, which editor they used, among other things.

Third, **diffs**. These are small files with the difference in the OSM map data between two points in time. You can use these to keep an existing copy of local OSM map data up-to-date. We will not cover these today.

# Data from osm.org

There is data you **cannot** get directly from OSM itself, like

**Thematic data**, for example, just the road network

Data for a **smaller area**, like a country

Data in **GIS formats** like Shapefile or GeoJSON

We will look at how and where you can get these types of data.

# Choosing a path

There's two main ways to do data visualization and analysis.

First, a desktop application like **QGIS** or **ArcGIS**

Second, a more programming centric environment like **Jupyter Notebooks**

# Pros and Cons

## Desktop

Intuitive, easy to get started

Working with very large datasets, like all of OSM, can be difficult

Non-map visualization options, like charts and graphs, are limited

You can't use all types of OSM data, for example changesets

## Jupyter Notebooks

Learning curve

Some traditional GIS analysis types are harder

Flexibility

You can use any data and work more easily with large datasets



# We will focus on QGIS today

As time allows...perhaps some notebooks

Let's start with QGIS!

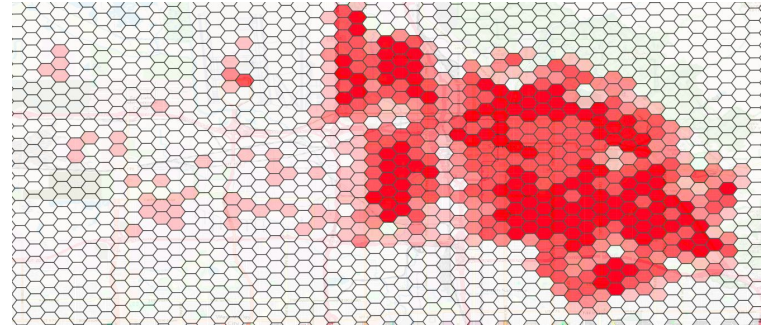
**QGIS** is a **free and open source** desktop GIS application. You can download your copy at [qgis.org](https://qgis.org).

The base application can be extended with **community plugins**



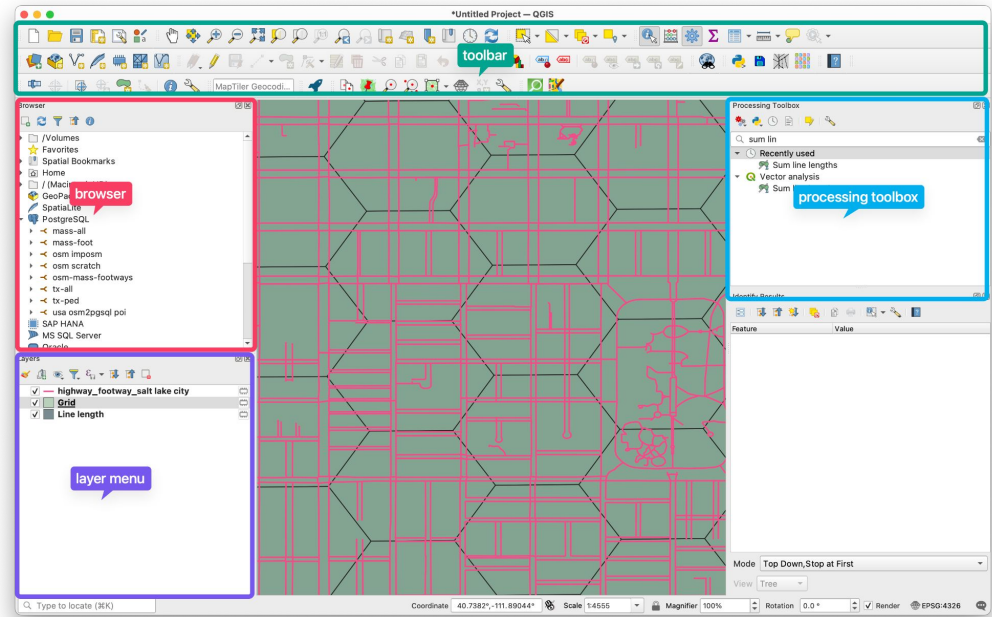
# What we'll do

- Query OSM from within QGIS
- Display the density of pedestrian infrastructure in a city



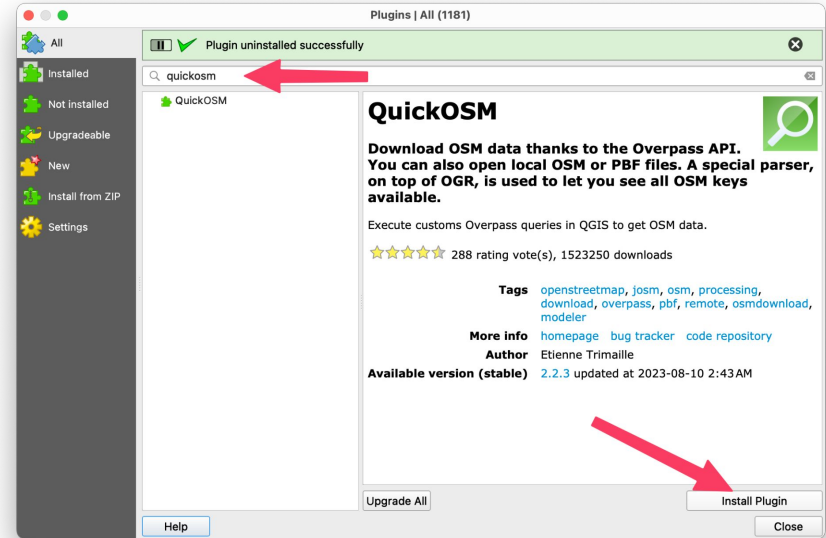
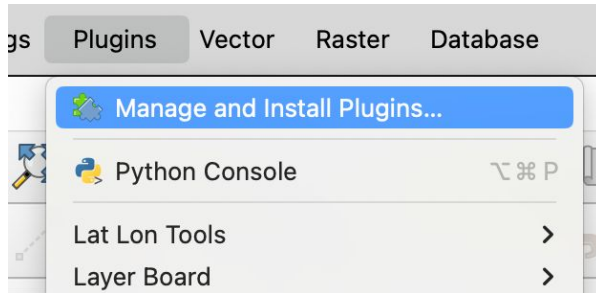
# Quick Tour of QGIS

- **Layer List**, this is where all your data layers live. You can click on them for options like styling. You can rearrange them.
- **Toolbar**, here you will find the most common tools for interacting with the data
- **Processing Toolbox**, for all advanced data manipulation



# Install the QuickOSM plugin

QuickOSM lets you query and download OSM data into a QGIS layer with a simple interface



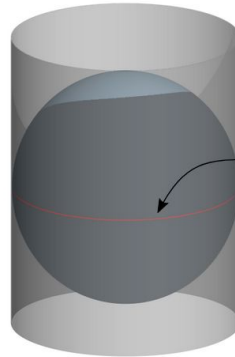
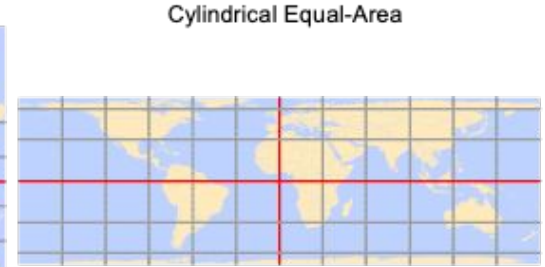
# Set an appropriate Coordinate Reference System

A CRS is a system that uses **coordinates** to establish and represent the positions of points on the Earth's surface.

This involves **projecting** the 3d sphere of the earth onto a flat 2d map.

This introduces distortion in **shape**, **area**, **distance** and **direction**.

[Learn more...](#)

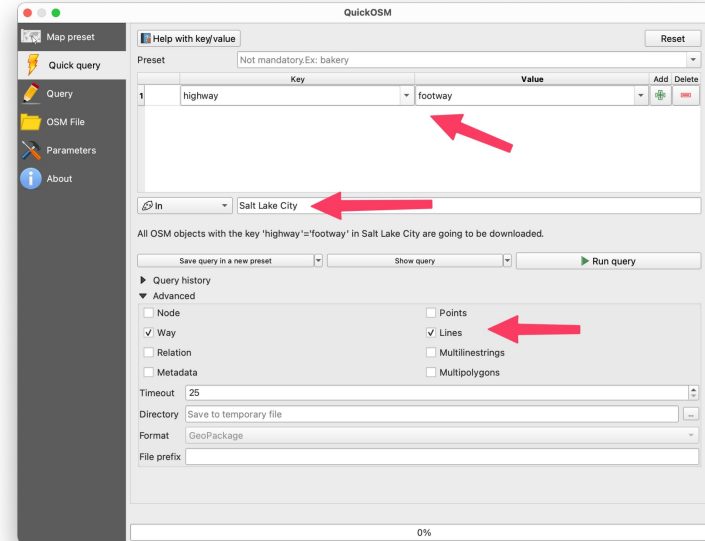
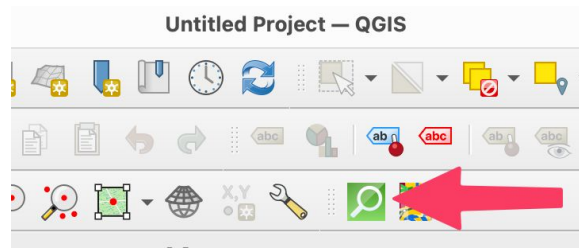


Projected plane touches earth surface **along one circle**

# Query some OSM data!

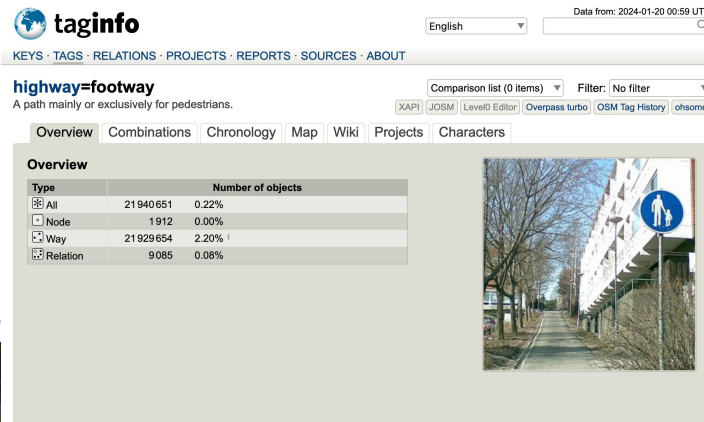
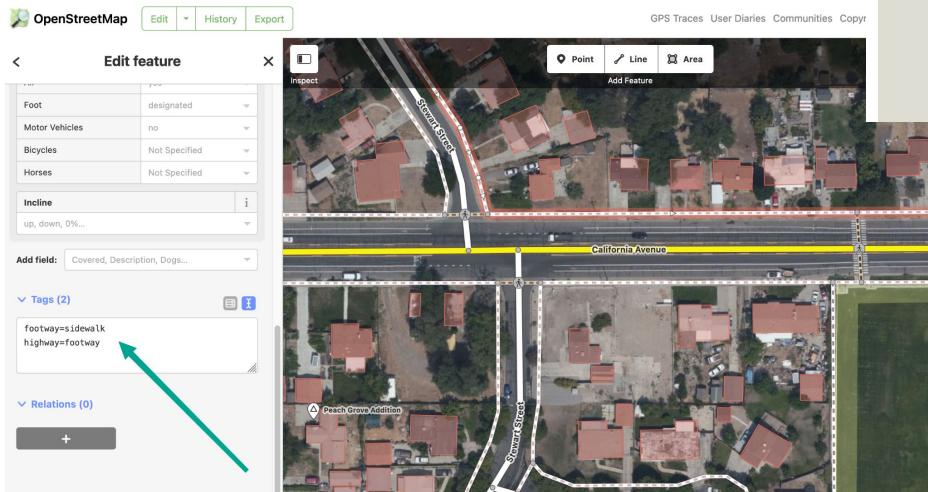
- Click the QuickOSM button
- In the QuickOSM window, we will query:
  - highway=footway
  - Located in Salt Lake City
  - Only ways / lines

highway=footway represents a foot path in OSM.



# Learning about OSM feature types

- [Wiki.osm.org](https://wiki.osm.org)
- [Taginfo](https://taginfo.osm.org)
- Inspecting tags while editing

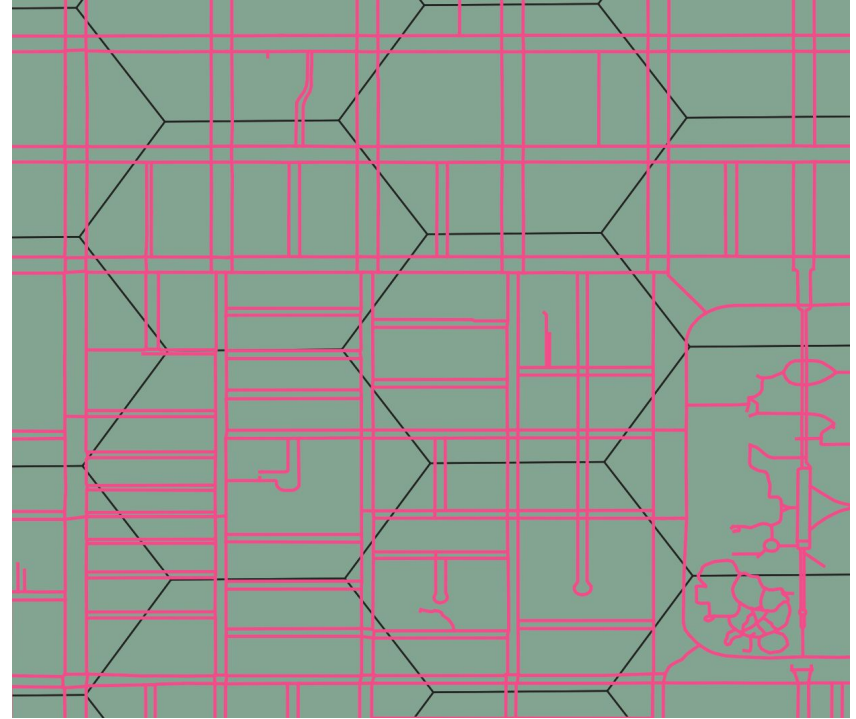


# Create a Grid

For aggregating data so we can visualize

Choosing the **right size** is important

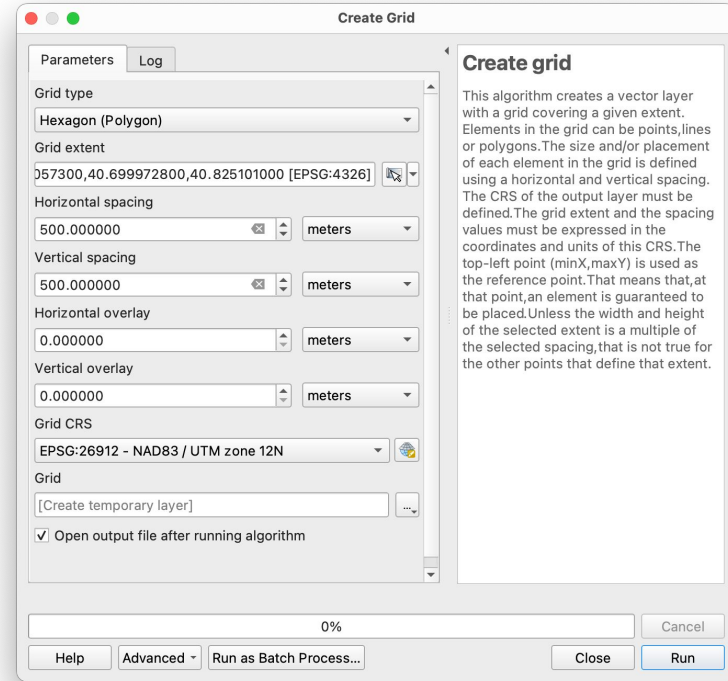
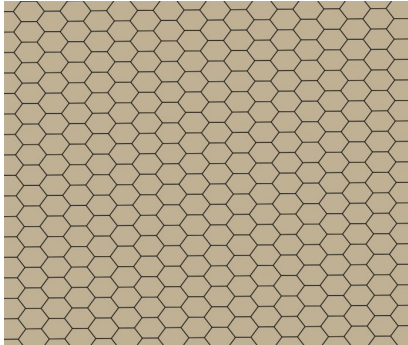
- Appropriate to the scale of your area of interest
- Should have enough density of data to be meaningful





# Create a Grid

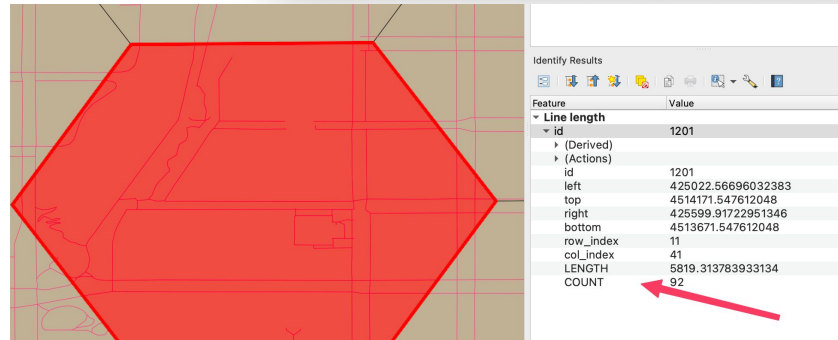
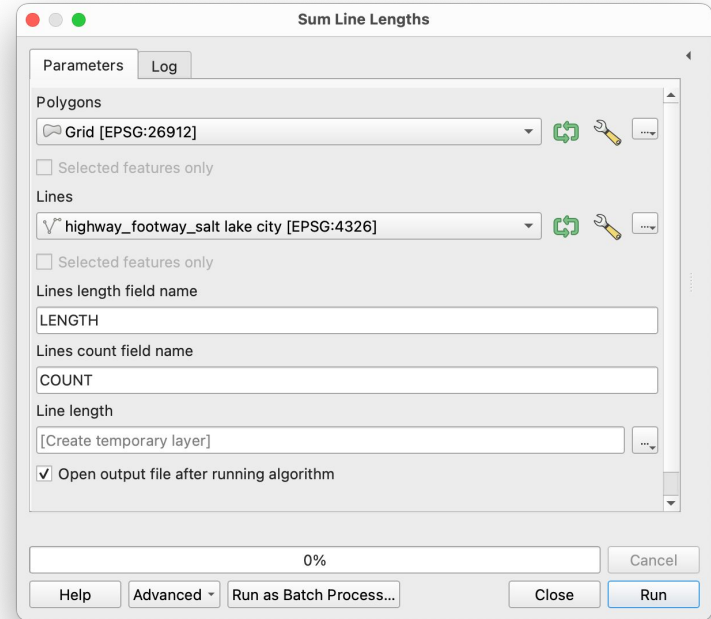
- Processing Toolbox -> Create Grid
- Grid Type: Hexagon
- Grid CRS: EPSG: 26912 [learn more](#)
- H/V Spacing: 500 meters



# Aggregate OSM Data

The length of footway for each cell

- Processing Toolbox -> Sum line lengths
- **Polygons:** Grid layer
- **Lines:** highway\_footway layer
- Optional: **Create Spatial Index** first (essential for large datasets)



# Symbology

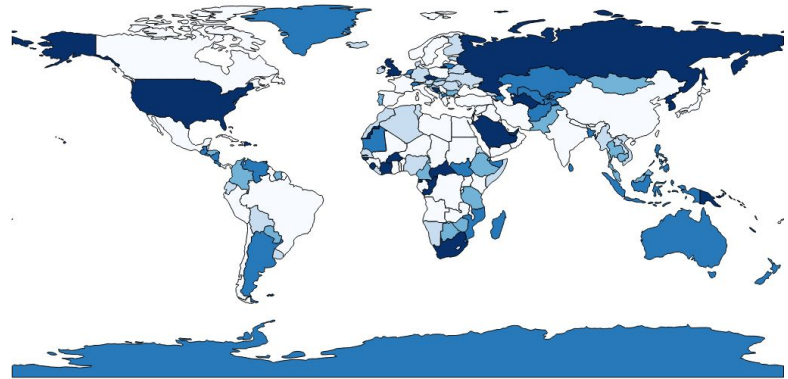
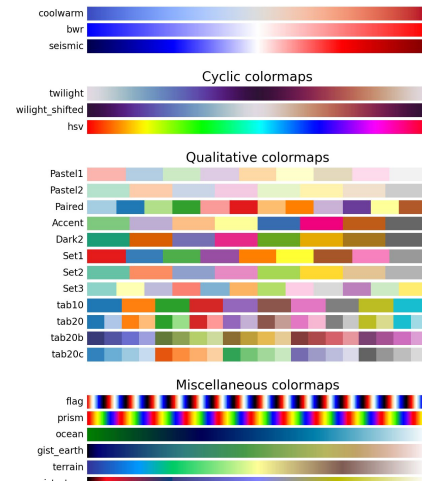
Essential to convey information at a glance.

What we're creating is called a choropleth map. [Learn more...](#)

A choropleth map uses different shading and colors based on quantitative data.

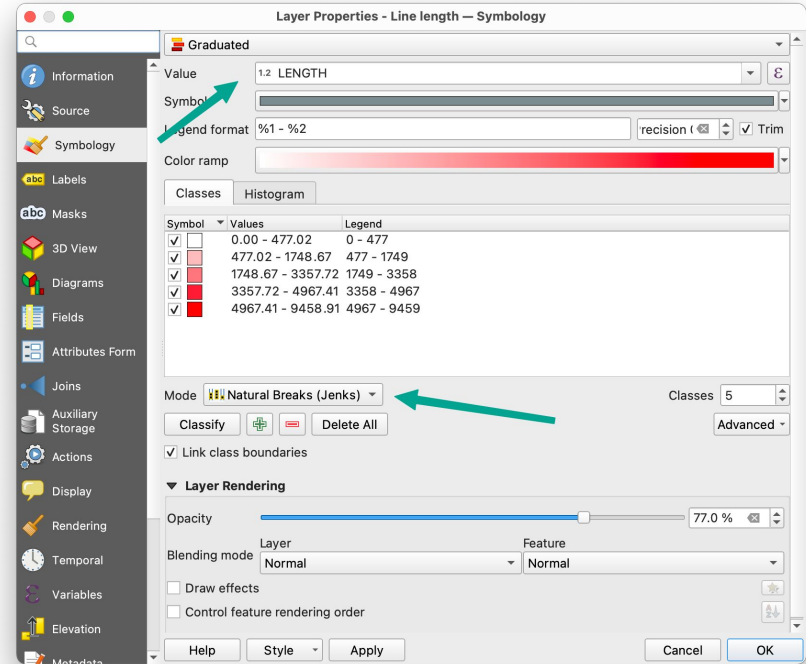
Choosing an **appropriate color ramp** is essential. Do we have **qualitative** or **quantitative** data?

And how to map data values to colors: **classification**.



# Let's style our grid

- Double click on the Line Lengths layer in the layer list
- Select “Symbology” from the left tabs
- We will use the “Natural Breaks” classification method



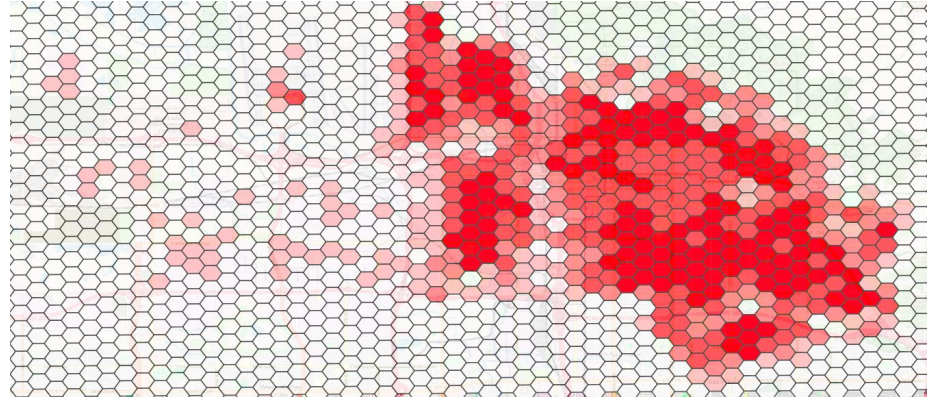
# Result!

A density map of footpaths in Salt Lake City

Are we satisfied with this??

Problems: hard to orient yourself

Let's try something else!

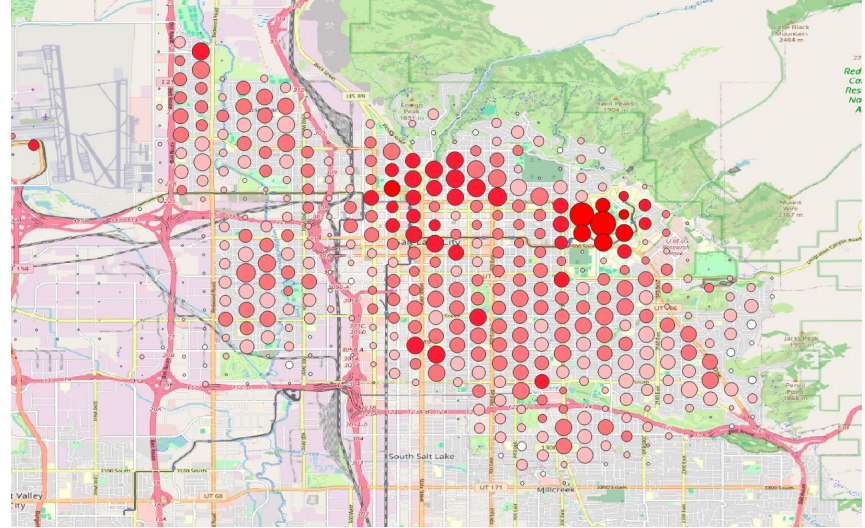


# Point Grid

Perhaps points will be better.

They don't completely cover the underlying base map

You can set the radius to represent the value

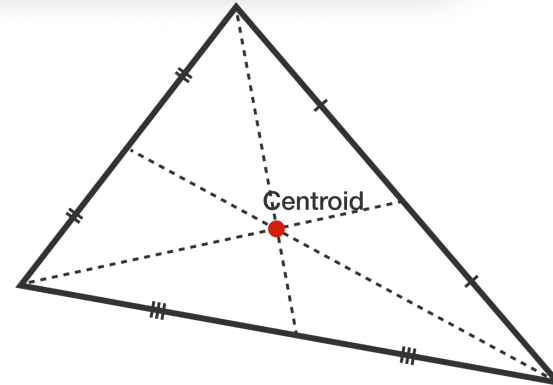
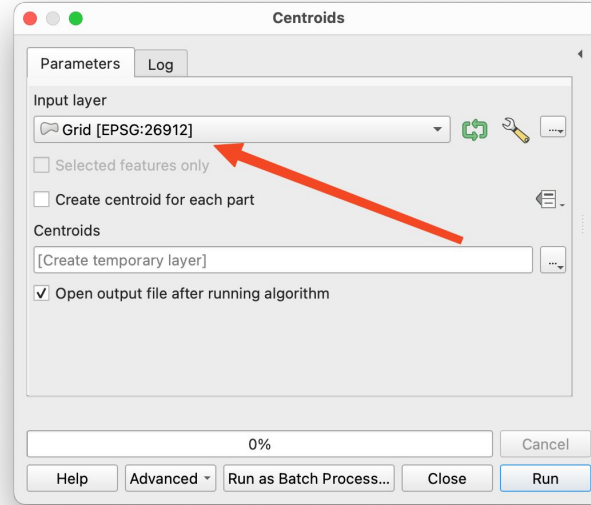


# Creation steps

Create “centroids” for each hex

This will generate a new layer with points at the center of each hexagon

The original values will be copied, so we don't have to perform the sum operation again



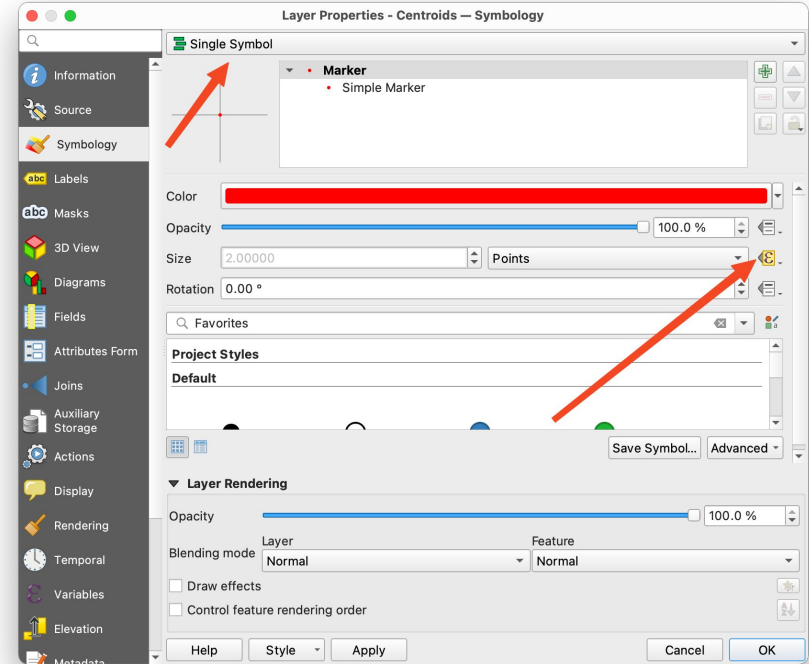
# Styling

We will represent the total length of footpaths in the circle size

Layer -> Symbology

Single symbol, so they are all the same color

Click on the  and select





# Styling

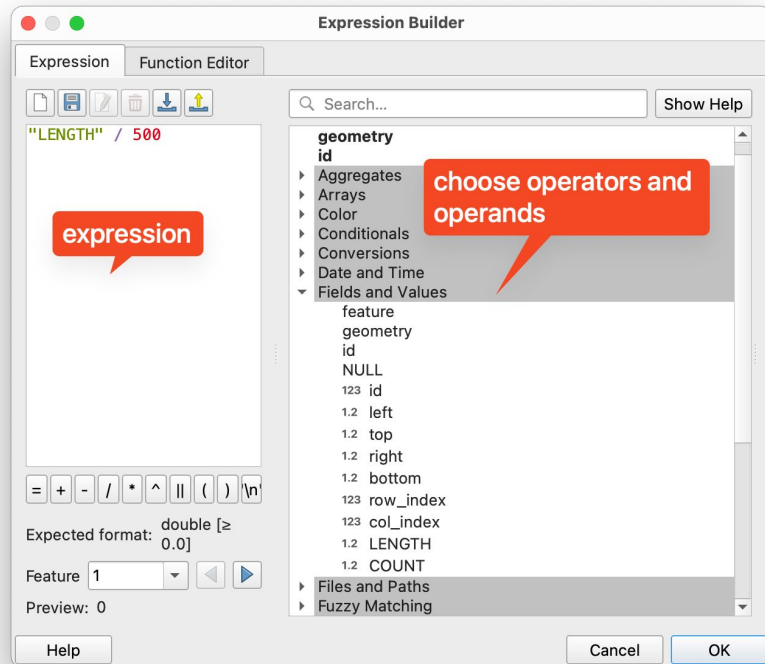
Click on the  and select “Edit...”

Now we can make the size a data-driven parameter.

This may require some experimentation!

In the Expression field, use

"LENGTH" / 500

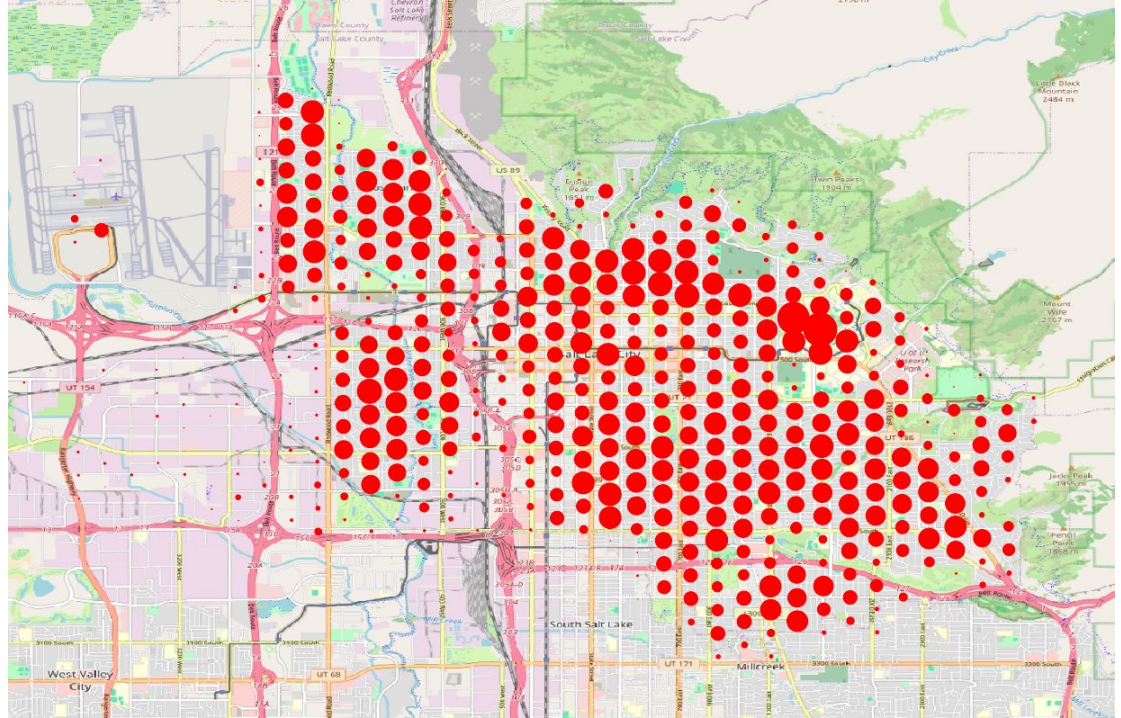


# Result!!

Much better.. What do you think?

## Exercises

- Different query. Shops? Hospitals? Parking?
- Query your own area



# Resources

Thanks to participants for providing some of these awesome learning resources:

- [QGIS tutorials](#)
- [QGIS training manual](#)
- [A gentle introduction to GIS](#)

If you want to start delving into Python and notebooks: [PyGIS](#)