Hitomi Nakamura
GIS 5577 Final Project

Introduction

Wildfires are uncontrolled fires that wipe out large areas of land. There are both natural and human-related causes for wildfire ignition. For example, the four major natural causes of ignition are lightning, sparks from rockfalls, spontaneous combustion, and volcanic eruption. On the other hand, the most common human-related causes for fire ignition include arson, discarded cigarettes, powerline arcs, and sparks from equipment.

If a wildfire strikes in a forest, an entire ecosystem consisting of animals, insects, birds, bacteria, and trees will be lost. Also, heavy air and water pollution, as well as soil degradation, will occur. If a wildfire strikes an area inhabited by people, the fires will destroy houses and other infrastructure.

The United States is no exception. Wildfires are one of the most common natural disasters in America. Specifically, it is a critical problem for those who live in California and southern areas like Nevada.

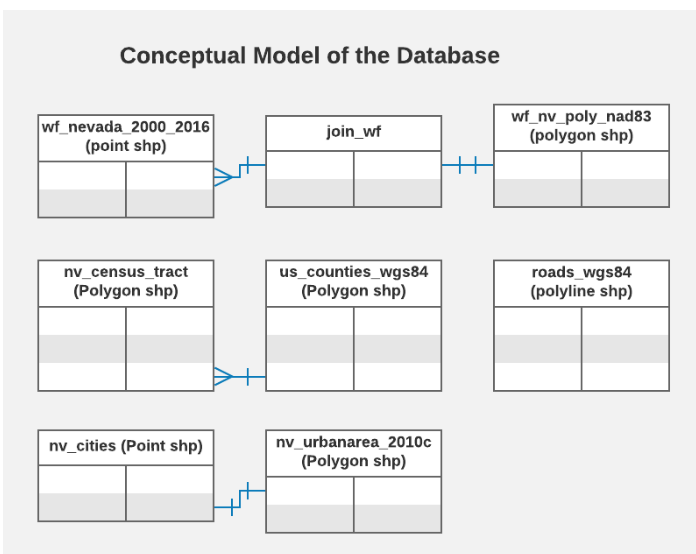
The primary scope of this project is to analyze wildfires that occurred in Nevada from 2000 to 2016. In this project, I decided to solve and visualize the following questions:

1. What is the wildfire distribution in Nevada?
2. How many wildfire locations are reported each year?
3. What is the total area size damaged each year?
4. Which census tract has the most wildfire damage area size (km^2)?
5. Which highways are potentially affected by wildfire?
6. Which cities may experience a larger impact resulting from wildfire?

This paper will introduce the conceptual model of the database, dataset, methodology, data analysis result, limitations of the project, and lessons I learned from the project.

Conceptual Model of the Database

The chart shows the conceptual model of the database that I used for this project.



There are seven original tables and one table was created with INNER JOIN. As you can tell from the model, the table has one-to-one and many-to-one relationships. Other than that, there will be a spatial relationship/join.

Dataset

The dataset that I used for this project are as follows:

- Census Tract shapefile, County shapefile, and Urban Area shapefile from the Census Bureau
- Wildfire Point shapefile from USGS Federal Wildland Fire Occurrence Data
- Wildfire Polygon shapefile from the USGS ScienceBase-Catalog
- Highway road shapefile and the US city point shapefile from the GIS 5577 class data

**Please look at the Database Schema for more details*

Methodology

After downloading shapefiles from the websites, I used shp2pgsql in a command line to import the dataset with NAD 83 projection.

Example:
```
shp2pgsql -W LATIN1 -s 4269 -I
C:\Users\nakam031\Downloads\tl_2017_32_tract\tl_2017_32_tract.shp
nv_census_tract | psql -h gis5777.csaba3m4f8xj.us-east-1.rds.amazonaws.com -d
nakam031 -p 5432 -U nakam031
```

After importing all the tables to Postgresql, I modified column names and dropped unnecessary columns in the tables to create clean tables with meaningful column names. For example, column name ALAND and AWATER in census tract (nv_census_tract) was changed to land_sqm, water_sqm. The reasoning behind why I changed the name for these columns is so that the unit is clear without going through the metadata.

I created a new table join_wf from joining the table wildfire point table (wf_nevada_2000_2016) and wildfire polygon table (wf_nv_poly_nad83). The table was joined using INNER JOIN of fire code. I joined these two tables since the wildfire point table contained more information such as causation of the fire and class size of the fire. This was necessary for data visualization. I created this table to use it as a data hub so that I do not have to keep using a sub query or common table expression.

```
--create table join_wf by joining point shp and polygon shp
DROP TABLE IF EXISTS
CREATE TABLE join_wf AS
SELECT py.gid AS py_gid, py.fire_code_ AS py_firecode,
 py.fire_dsc_1 AS py_date, py.fire_cause py_cause,
 py.gis_acres AS py_acres, py.geom AS py_geom,
 pt.gid AS pt_gid, pt.firename AS pt_firename, pt.firecode AS pt_firecode,
 pt.cause AS pt_cause, pt.sizeclass AS pt_sizeclass, pt.firetype AS pt_type,
 pt.startdated AS pt_startdate, pt.totalacres AS pt_totacres, pt.geom AS
pt_geom
 FROM wf_nv_poly_nad83 py
 INNER JOIN wf_nevada_2000_2016 pt ON (py.fire_code_=pt.firecode)
WHERE py.fire_dsc_1=pt.startdated
```

I modified column names and dropped some of the columns from the urban area table. For example, I renamed NAME10 to ua_name and UATYP10 to area_type. Also, I dropped LSAD10 since it was the same as UATYP10 but in a different format. Then, I created a new table that only shows the urban area in Nevada (nv_urbanarea_2010).

```
--create a new table of urban area
CREATE TABLE nv_urbanarea_2016 AS (
SELECT * FROM us_urbanarea_2016 WHERE lower(NAME10) LIKE '%nv%')
--ALTER TABLE name
ALTER TABLE IF EXISTS nv_urbanarea_2016
rename to nv_urbanarea_2010
```

ua_name is a string with both the urban area name and state name. I decided to divide this column into two columns: urban_name and state_name. My approach was to create a new table nv_urbanarea_2010c using common table expression and LEFT JOIN.

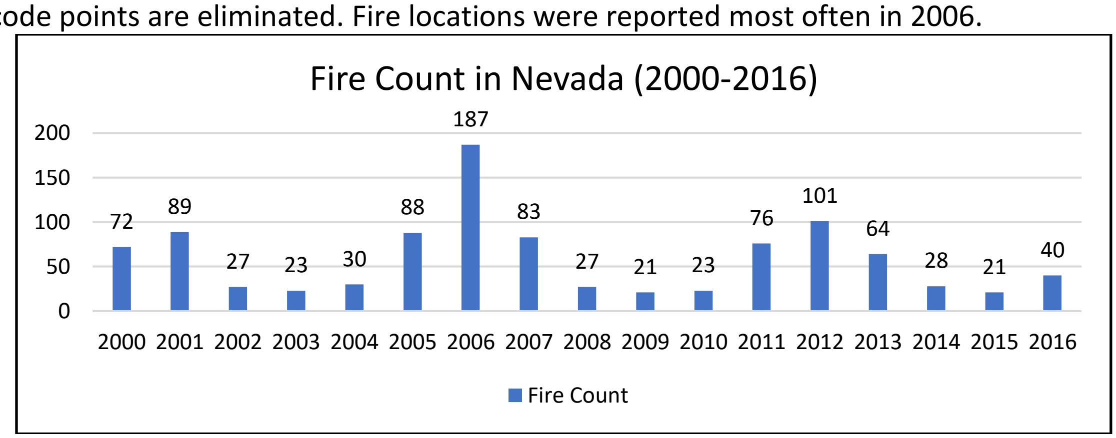
```
--create a table WITH separate urban area name and state
CREATE TABLE nv_urbanarea_2010c AS(
WITH urbanarea AS (
SELECT split_part(ua_name,',', 1) AS urban_name,
      split_part(ua_name,',',2) AS state_name, gid FROM nv_urbanarea_copy)
SELECT u.gid, u.urban_area, c.urban_name, c.state_name, u.affgeoid10,
u.geoid10,
      u.area_type, u.land_sqm, u.water_sqm, u.geom
      FROM nv_urbanarea_2010 u
      left join urbanarea c ON (u.gid=c.gid))
```

Data Analysis and the Result

All the following maps are projected in NAD 83 UTM zone 11 N.

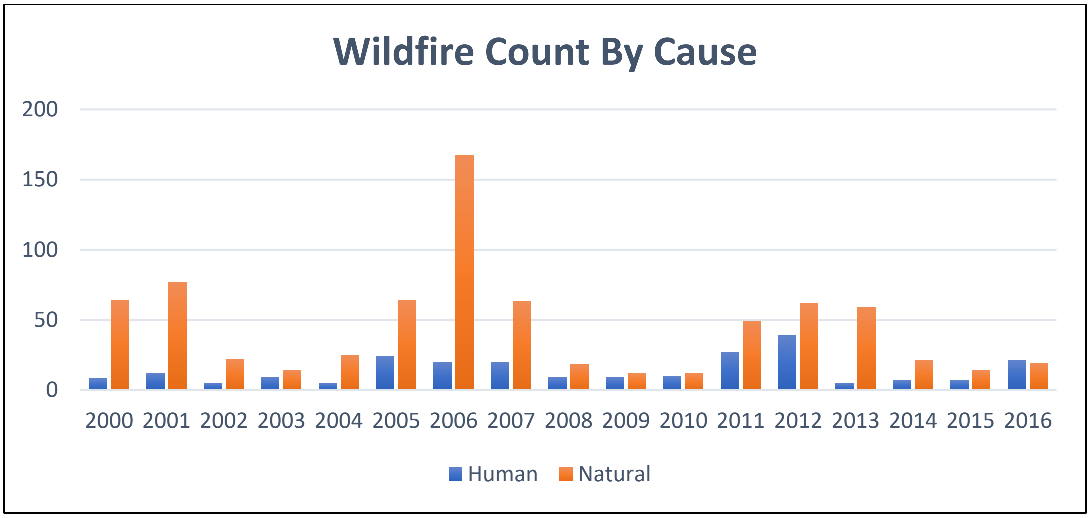
1. **What is the wildfire distribution in Nevada?**
   Map 1 shows the wildfire distribution in Nevada by class size. To show the overlapped area in the inset map, transparency is set to 40%. From the map, we can tell that class size G fires are concentrated in the northern and southwest areas of Nevada. From the inset map, we could tell that same areas are experiencing wildfire multiple times with different class size from 2000 to 2016.

2. **How many wildfire locations are reported each year?**
   Graph 1 shows the number of fire locations reported in each year from 2000 to 2016. The data is from join_wf table which means one count per fire code. Also, undefined fire code points are eliminated. Fire locations were reported most often in 2006.



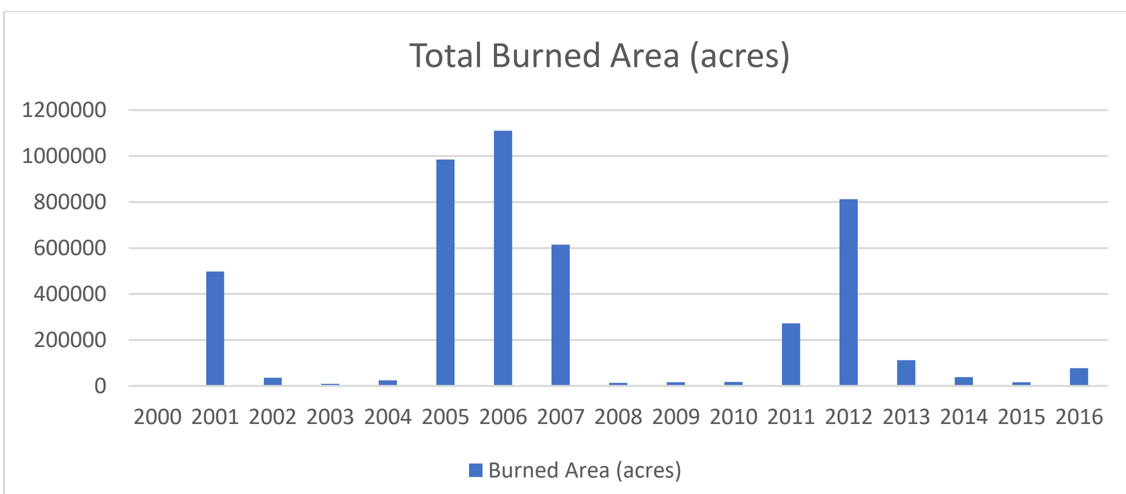
Graph 1

Graph 2 shows the number of fire locations reported by cause from 2000 to 2016. The data is from the join_wf table as well and with one count per fire code. From this graph, we can tell that most of the fires are naturally caused each year with the exception of 2016.



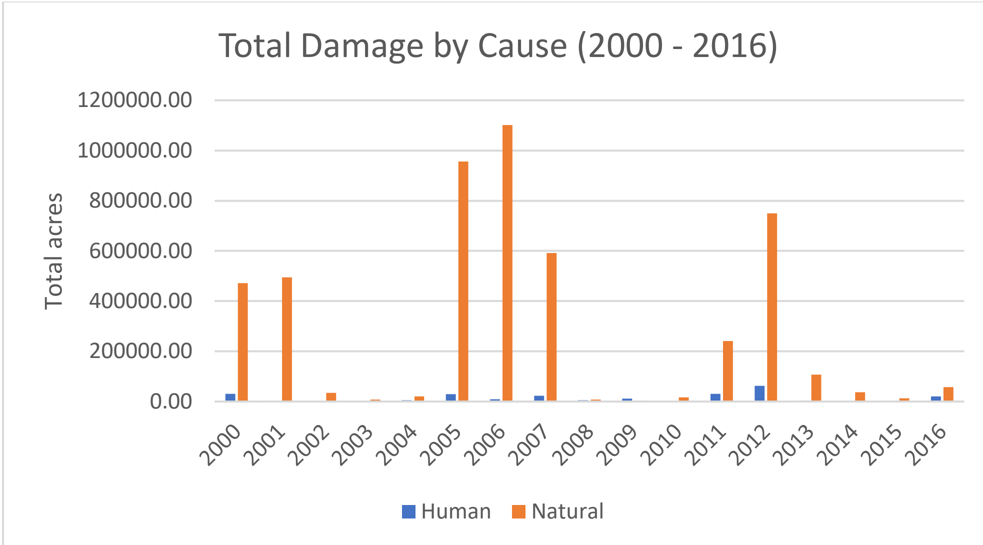
Graph 2

3. **What is the total area size damaged each year?**
Graph 3 shows the total burned area in Nevada from 2000 to 2016. As you would assume from Graph 1, 2006 has the largest burned area. However, a higher number of reported locations does not necessarily mean that the burned area is larger. For example, there were 88 points in 2005 and 101 points in 2012 (Graph 1). However, if we look at Graph 3, 2005 has a larger burned area than 2012. From this result, we can conclude that individual fire was larger in 2005 in comparison to the fire in 2012.



Graph 3

Graph 4 shows the total damage by cause from 2000 to 2016. From this graph, we can tell that naturally-caused wildfires are both more powerful and spread larger.



Graph 4

4. **Which census tract has the most wildfire damage area size (km^2)?**
   To conduct this analysis, I used a spatial join ST_INTERSECTS to find wildfire points that overlap with each census tract. I then summed the burned area in sqkm (*Ⅰ). Map 2 shows the result of the query. Census tracts with grey polygons mean that there are no wildfires reported in that tract. Northern Nevada and one census tract in the southeast have the largest damage size. This result is identical to the result in Map 1.

5. **Which highways are potentially affected by wildfire?**
   For this analysis, I used ST_BUFFER and ST_INTERSECTS. I created a 100 m and 200 m buffer of the wildfire points and obtained highways that intersected with the buffer. Since the highway data was in WGS 84, I had to use ST_TRANSFORM as well (*ⅠⅠ). All the highways within 100 m and 200 m are in northern Nevada. From the spatial analysis, we can assume that Highway US Route 93 has the highest potential to be influenced by wildfire.

6. **Which cities may experience a larger impact resulting from wildfire?**
   This analysis was done by using ST_INTERSECTS and ST_BUFFER as well. I created a 2 km buffer of the urban area polygons and calculated the total area size (km^2) of the overlapping wildfire points (*ⅠⅠⅠ). We can see that both Incline Village and Reno in west Nevada have a higher impact from wildfire (Map 4). *This analysis takes only the spatial relations into consideration and not the population.*

Limitations and Challenges for the Project

From this project, I learned the importance of knowing the relationship between tables. For example, the result will be different depending on how we join the tables (INNER JOIN or LEFT JOIN). Also, providing a meaningful name for all the columns save a lot of time during the analysis phase since I do not have to keep going back to the metadata to find which columns mean what. Another lesson I learned is that it is important to give a detailed comment to all the codes that I have written. This allows me to understand the code, my logic, and my thoughts that I had during the moment in which I wrote the code.

Originally, I was planning to implement a population change element in the project. However, I soon realized that I cannot infer the relationship between the population change and wildfires. In other words, I cannot conclude that wildfires caused the population to change in the counties even though I could see the population change. To simplify the project, I decided not to add the population element to this project. Instead, I conducted a pure spatial analysis.

In the analysis 5 and 6, I tried to simply pick wildfire polygons that intersected with highways or urban areas. However, I found ring self-intersection errors in the polygon geometry. This is a geometry problem and I found the error by using ST_isValid and ST_isValidReason. This error usually happens when polygons are created from raster data. In order to fix this problem, I searched for possible solutions online.

The first approach was to use ST_SimplifyPreserveTopology. This returns a simplified version of the given geometry using the Douglas-Peucker algorithm. However, I still found some ring self-intersection errors.

```
UPDATE join_wf
SET py_geom = st_SimplifyPreserveTopology(py_geom,0.0001)
WHERE st_isvalid(py_geom)=false;
```

The second approach was to use ST_MakeValid. This function attempts to make an invalid geometry valid without losing vertices. This function worked and was able to use the ST_Intersects function to the wildfire polygons (*IV).

gid integer	geom geometry	pt_startdate date	name character varying (120)
17372	010500002...	2003-07-17	State Route 225
27264	010500002...	2006-07-02	State Route 375
29247	010500002...	2010-07-01	State Route 168
19562	010500002...	2007-05-18	Interstate Route 80
20536	010500002...	2007-05-30	State Route 341
20716	010500002...	2004-07-14	US Route 50, US Route 395
19150	010500002...	2003-07-10	US Route 395
20615	010500002...	2006-06-26	US Route 50

Table 1 Result using the ST_MakeValid

I found the difficulty of both searching online and reading the Postgresql's tutorials and references whenever I faced an error. Specifically, Postgresql's documentation was difficult to follow.

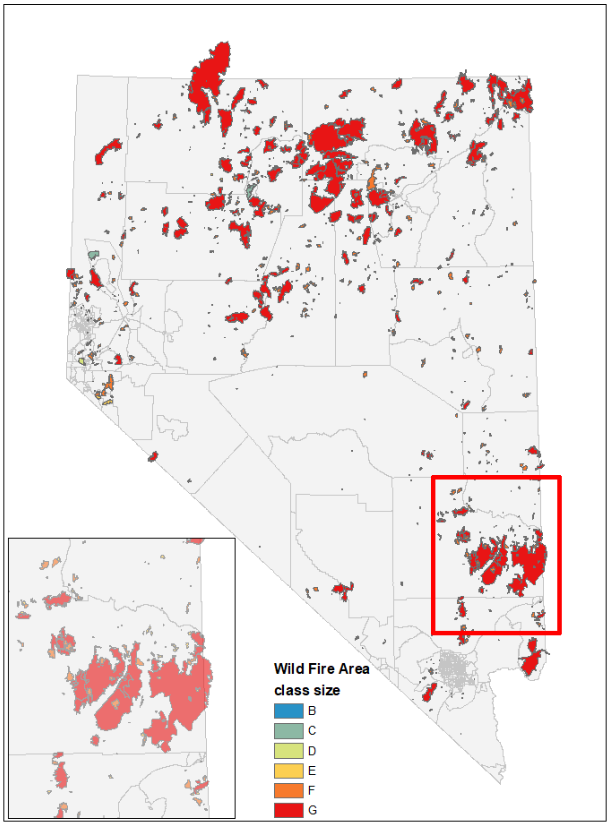
Further study could be implement landcover data to find which common land types experience wildfire.

Appendix

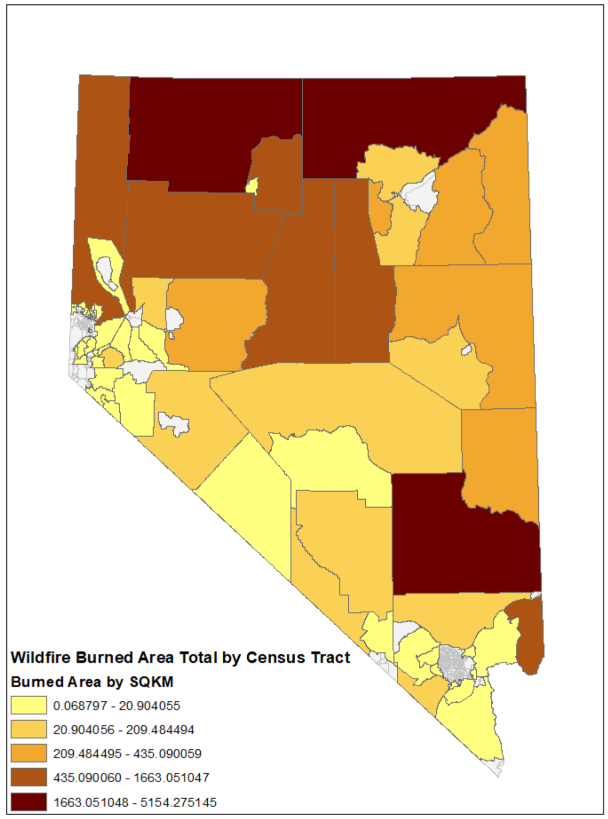
Code Examples for spatial analysis:

```
I. --spatial join with wildfire locations and census tract
--Create a choropleth map shows wildfire damaged area (sqkm) by census tract
(Map 2)
WITH census_tract AS (SELECT c.name AS countyname,
t.geoid AS tract_id, t.countyfp AS county_fp, ST_transform(t.geom,26911) AS
tract_geom
FROM nv_census_tract t, us_counties_wgs84 c
WHERE c.statefp = '32' AND t.countyfp = c.countyfp)
SELECT sum(pt_totacres)*0.00404686 AS wf_areasqkm, c.countyname, c.tract_id,
c.tract_geom
FROM join_wf j, census_tract c
WHERE ST_intersects(c.tract_geom,ST_transform(j.pt_geom,26911))
GROUP BY c.countyname, c.tract_id, c.tract_geom

II. --Find roads that intersects with 200m buffer of wildfire points
--In order to find the 100m buffer intersection change 200 to 100 (Map 3)
WITH nv_road AS (
SELECT * FROM roads_wgs84
WHERE lower(state)='nv'),
point_buffer AS (
SELECT ST_buffer(ST_transform(pt_geom,26911),200) AS buffergeom,
pt_sizeclass, pt_startdate
FROM join_wf)
SELECT r.gid, ST_transform(r.geom,26911) geom, p.pt_sizeclass,p.
pt_startdate,r.name
FROM nv_road r, point_buffer p
WHERE
ST_intersects(ST_transform(p.buffergeom,26911),ST_transform(r.geom,26911))

III. --create a buffer of urban area and check which urban area has the most
burned area in radius 2km (Map 4)
WITH urban_buffer AS (--create a 2km buffer
SELECT ST_buffer(ST_transform(geom,26911),2000) AS ubuffer_geom, geom,
gid, area_type,urban_name
FROM nv_urbanarea_2010c)
--get the sum of area burned in sqkm
SELECT sum(p.pt_totacres)*0.00404686 AS total_sqkm, u.geom,
u.area_type, u.urban_name
FROM join_wf p, urban_buffer u
WHERE
ST_intersects(ST_transform(u.ubuffer_geom,26911),ST_transform(p.pt_geom,26911
))
GROUP BY u.area_type,u.urban_name, u.geom

IV. --The code to fix the ring self-intersection error
UPDATE join_wf
SET py_geom=ST_MakeValid(py_geom)
```

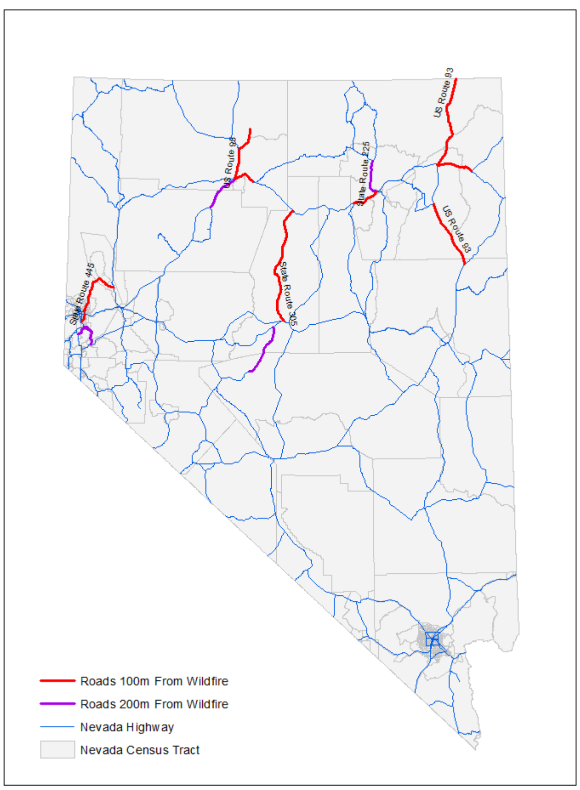
Hitomi Nakamura
GIS 5577 Final Project



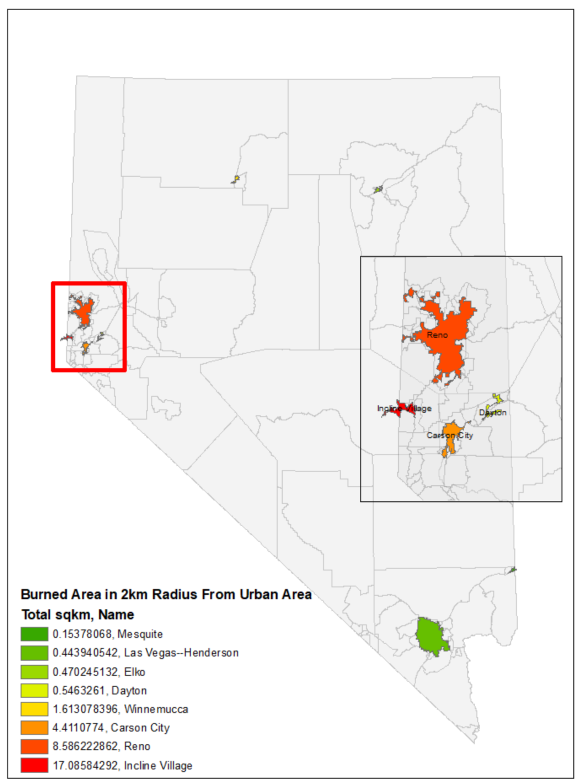
Map 1



Map 2



Map 3



Map 4