Investigation of deforestation in US national forest and its cause

Team members: Judy Jin , Xuhui Liu

2. Question(s) you addressed, why it is important

The questions we want to investigate are:

- What are the current status of deforestation in every US national forest?
- What are the percentage of areas suffering from deforestation in different forest areas of US?
- What are the special characteristics of areas suffering from deforestation?
- What are the factors that contribute most to the deforestation?

These questions are important because deforestation is always an important issue in protecting our environment. The forest areas play an important role in absorbing hazardous elements in the air.

Our expected audiences are those who are concerned with environment issues and wish to slow down the deforestation. They would benefit from our analysis by understanding what are the key factors that contributes to the deforestation. Then, they may be able to take some actions to deal with these factors and slow down the process of deforestation.

3. Background and literature

- K-means to identify deforestation area (https://developers.arcgis.com/python/sample-notebooks/detecting-deforestation-using-kmeans-clustering-on-sentinel-imagery/)
  This article is an example in ArcGis Python API. It introduces a whole process of filtering useful sentinel-2 images and implement a Kmeans algorithm to classify the deforestation areas. It is mainly about identifying the area of deforestation. It is very useful in our project when we want to calculate deforestation score.
- Analyzing the Factors of Deforestation in the Amazon using GIS and Logistic Regression (https://geg.uoguelph.ca/sites/default/files/G3_AODA_W20.pdf)
  This article is by Professor Ben DeVries from University of Guelph. This article is also about investigating what factors may contribute to the deforestation in Amazon. The author also build a predictive model and conclude that road is the main factor that contribute to the deforestation.
- https://earthobservatory.nasa.gov/features/Deforestation/deforestation_update3.php (https://earthobservatory.nasa.gov/features/Deforestation/deforestation_update3.php)
  This article is from NASA earth observatory. The author analyze the reasons of deforestation. From this article, we will have a rough idea about what features may contribute to the deforestation. So, we will be able to decide on what features to include in our analysis model.It mentions human activities such as urbanization, road expansion, and commercial activities. It also mentions natural causes such as climate change and wildfire.
- https://www.fs.usda.gov/speeches/state-forests-and-forestry-united-states-1 (https://www.fs.usda.gov/speeches/state-forests-and-forestry-united-states-1)
  This article is from Tom Tidwell, ChiefWorld Conservation Congress Honolulu, HI in September 4, 2016. The article emphasizes on the great implication of preserving national forests in the US. The author also mentions the great challenges that faced by the US national forests since a century ago including drought, wildfire, and insects. He also talks about the influence of increasing population and urbanization on the deforestation of the US. Final

4. Python libraries or ArcGIS modules you used and why

Python Libraries:

- **Numpy**: We used numpy for some arithmetic operations and to access np.NaN
- **Pandas**: We used pandas for read in addtional dataset and for spatially-enabled dataframe manipulation.
- **Matplotlib**: We used to visualize our perdiction results
- **Sklearn**: We used the linear model to predict our final deforestation result
- **Patsy/statsmodels.api**: We used patsy to plot our statstical summary for our prediction result
- **seaborn**: We used seaborn to plot pairplot for the different features

ArcGIS Modules

- **Geometry**: We used to creat our area of interest
- **Features**: We used when publishing/accessing feature layers on ArcGIS Online
- **Geoenrichment**: We used geoenrichment to further extract our feature set
- **Buffer**: We applied buffering on the deforestation area we found to creat a larger study area.
- **Raster**: We used various raster analyses and the construction of our deforestation model

This list is mainly the same with our project proposal. However, we changed from using machine learning tools to find deforestation situation to using raster functions to construct the deforestation model and calculate the scores due to the limited time we have in this project.

In [1]:

```
%matplotlib inline

import matplotlib.pyplot as plt
import geopandas as gpd
import pandas as pd
import numpy as np
import json
import time

from sklearn.linear_model import LinearRegression
import sklearn
import patsy
import statsmodels.api as sm
import scipy.stats as stats
import seaborn as sns
```

In [2]:

```
import arcgis
from arcgis.gis import GIS
from arcgis.raster.functions import *

from arcgis import geometry
from arcgis.geocoding import geocode
from arcgis.features import GeoAccessor, GeoSeriesAccessor
from arcgis.features import FeatureLayerCollection
from arcgis.features import FeatureLayer
from arcgis.features import use_proximity
from arcgis.features import summarize_data
from arcgis.geoenrichment import *

from IPython.display import display, Image
from ipywidgets import *
import graphviz
```

In [3]:

```
import  getpass
username = input('Enter  username: ')
password = getpass.getpass("Enter  password:  ")
agol = GIS(username=username, password=password)
```

```
Enter  username: dsc170wi22_19
Enter  password:  ........
```

5. Data sources

Since we have three sections in our projects: **Investigation of deforestation**, and **Understand and extract the factors related to deforestation**, and **deforestation perdiction**, we have many dataset to use.

We seperate our data sources to three part:

```
- data for investigation of deforestation (raster data),
- data for feature evalution (mostly vector data),
- addtional data (analysis result data).
```

5.1 Data for investigation of deforestation (raster data)

1. Sentinel-2 satellite imagery (https://registry.opendata.aws/sentinel-2) 10m Multispectral, Multitemporal, 13-band images with visual renderings and indices. This Imagery Layer is sourced from the Sentinel-2 on AWS collections and is updated daily with new imagery. We use this layer as our base data layer to find the define deforestation and calculate scores.
2. World satellite imagery data (https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer/3) provides one meter or better satellite and aerial imagery in many parts of the world and lower resolution satellite imagery worldwide. The map includes 15m TerraColor imagery at small and mid-scales (~1:591M down to ~1:288k) for the world. We use this data to compare our calculated deforestation results and our natrual appearence.

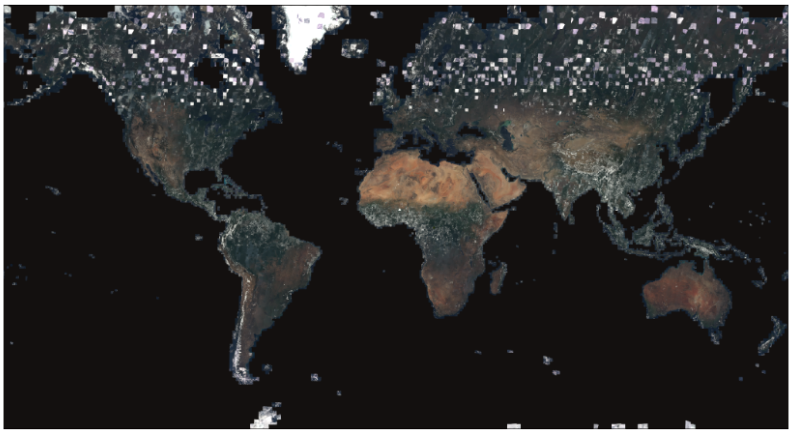
In [4]:

```
# 1.  Sentinel-2 satellite imagery
sentinel = agol.content.search('Sentinel-2 Views', 'Imagery Layer', outside_org=
True)[0].layers[0]

# 2. World satellite imagery data
world = ImageryLayer('https://services.arcgisonline.com/ArcGIS/rest/services/Wor
ld_Imagery/MapServer/3')
```

In [5]:

```
sentinel
```

Out[5]:



5.2 Data for feature evaluation (mostly vector data)

1. Administrative Forest Boundaries\ An area encompassing all the National Forest System lands administered by an administrative unit.\ https://data.fs.usda.gov/geodata/edw/datasets.php?dsetCategory=boundaries (https://data.fs.usda.gov/geodata/edw/datasets.php?dsetCategory=boundaries)

1. Current wildfire point layer\ This layer presents the best-known point and perimeter locations of wildfire occurrences within the United States over the past 7 days. Points mark a location within the wildfire area and provide current information about that wildfire.\ Perimeters are the line surrounding land that has been impacted by a wildfire.\ https://ucsdonline.maps.arcgis.com/home/item.html?id=d957997ccee7408287a963600a77f61f (https://ucsdonline.maps.arcgis.com/home/item.html?id=d957997ccee7408287a963600a77f61f)

1. Fire burned area polygon layer\ It describes the fire burned area in US from 1982 to 2019

1. Beetles point layer\ Data for all 2020 tamarisk beetle presence points

1. US freeways layer\ This layer presents rural and urban interstate highways.

1. US climate data\ Historical U.S. Monthly Average Temperature and Precipitation from Global Historical Climate Network\ Daily (GHCND) from 1981 through 2010

1. Population data in forest service zone by geoenrichment

In [6]:

```
# 1.Administrative Forest Boundaries
forest_layer = agol.content.get('bb5a9047d29443b6ba2b9e202c60c214')
sdf = pd.DataFrame.spatial.from_layer(forest_layer.layers[0])
serviceURL1 = 'https://services1.arcgis.com/eGSDp8lpKe5izqVc/arcgis/rest/services/S_USA.AdministrativeForest/FeatureServer'
forest_collection = FeatureLayerCollection(serviceURL1, gis=agol)

# 2.Current wildfire point layer
serviceURL = 'https://services9.arcgis.com/RHVPKKiFTONKtxq3/arcgis/rest/services/USA_Wildfires_v1/FeatureServer'
fire_collection = FeatureLayerCollection(serviceURL, gis=agol)
fire_sdf = fire_collection.layers[0].query(out_sr=3857).sdf

# # 3.Fire burned area polygon layer (This need time to process, we seperate to
 multiple cells below)

# 4.Beetles point layer
# Data for all 2020 tamarisk beetle presence points
serviceURL2 = 'https://services3.arcgis.com/RSsqTANV8DJhz9AY/arcgis/rest/services/2020_Tamarisk_Beetle_Present/FeatureServer'
beetle_fc = FeatureLayerCollection(serviceURL2, gis=agol)
beetle_sdf = pd.DataFrame.spatial.from_layer(beetle_fc.layers[0])
beetle_fl = beetle_fc.layers[0]

# 5.US freeways layer
# This layer presents rural and urban interstate highways.
serviceURL3 = 'https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/USA_Freeway_System/FeatureServer'
freeway_fc = FeatureLayerCollection(serviceURL3, gis=agol)
freeway_fl = freeway_fc.layers[0]
freeway_sdf = freeway_fl.query(out_sr=3857).sdf

# 6.US climate data
# Historical U.S. Monthly Average Temperature and Precipitation from Global Hist
orical Climate Network
# Daily (GHCND) from 1981 through 2010
serviceURL4 = 'https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/USA_GHCND_ACIS_Monthly/FeatureServer'
climate_fc = FeatureLayerCollection(serviceURL4, gis=agol)
climate_fl = climate_fc.layers[0]
climate_sdf = climate_fl.query(out_sr=3857).sdf

# 7.Population data in forest service zone by geoenrichment
# forest_with_population = enrich(study_areas=forest_buffer_sdf, data_collection
s=["Population"], return_geometry=False)
```

In [7]:

```
# 3.Fire burned area polygon layer (This need time to process, we seperate to mu
ltiple cells)
# It describes the fire burned area in US from 1982 to 2019
burnedURL = 'https://services.arcgis.com/P3ePLMYs2RVChkJx/arcgis/rest/services/MTBS_Polygons_v1/FeatureServer'
burned_area_collection = FeatureLayerCollection(burnedURL, gis=agol)
```

In [8]:

```
feature_set = burned_area_collection.layers[0].query(out_sr=3857)
burned_sdf = feature_set.sdf
```

5.3 Addtional data (other data type and analysis result data)

1. Forest shapefile data (https://ucsdonline.maps.arcgis.com/home/item.html?id=b1dc1eb780e249368ffb81d4272b8c43). This is the shape file for our forest data, mainly same as first dataset in 5.2 but need some special functions to do the AOI clip.
2. Deforestation score data (https://ucsdonline.maps.arcgis.com/home/item.html?id=1b26449c969f4514932525bc0713c5c1). This is calculate using the raster layer on the deforestation score, since the calculation requires a long time frame, we include this calculated results for simple use.
3. Fire severity data (https://ucsdonline.maps.arcgis.com/home/item.html?id=7da22eb584ce4ee49760405185fefdf2) is the data we calculated before, for easier use, we uploaded it into arcgis.

In [9]:

```
# 1. Forest shapefile data.
forest_item = agol.content.get('b1dc1eb780e249368ffb81d4272b8c43')
tem_data = forest_item.get_data()
gdf = gpd.read_file(tem_data)

# 2. Deforestation score data
deforestation_item = agol.content.get('1b26449c969f4514932525bc0713c5c1')
tem_data = deforestation_item.get_data()
deforestation_scores = pd.read_csv(tem_data)
deforestation_scores.head()

# 3. New fire data
fire_item = agol.content.get('7da22eb584ce4ee49760405185fefdf2')
tem_data = fire_item.get_data()
new_fire = pd.read_csv(tem_data)
```

5.4 Discussion about data

After proposal, we decided not to use the use the Landsat data but only sentinel2 data because we think sentinel 2 have better NVDI accuracy and those two dataset serve as a duplicate. Moreover, we adopted the world imagery data to serve as a manual scrutinization of our deforestation states. We also included more vector data for feature investigation.

- Sentinel-2: We may experience different capture date for different parts of the US national forest. One may come from summer while other may come from winter, which the vegetation may not be the same. Moreover, the image may also not reflect the current states of the national forest since several wildfire had happened near some forest, which may not be captured by sentinel-2.
- Wildfire: The current wildfire layer may not be consistent with our salliet image since the salliet image was taken in different time, which may not reflect the damage of current wildfire yet.
- Beetles: The beetles data is not specific enough. The beetles data only include the tamarisk beetles but not mountain beetles.

6. Data cleaning

Our main data cleaning was focusing on the feature extraction part, which we want to align our feature to the correct area of interest (national forest). Specifically, we need to creat buffer and fill our area of interest with information from selected feature layers. We have done much more work than we expected in this part. It is time consuming and tedious. For the raster layer data, we did a small amount of data cleaning, for example remapping for the NVDI score. However, we put all those in the analysis part since those require additional procedures.

In [12]:

```
# Data Cleaning we have done

# 1.Create a forest service zone as our AIO
# Since the forest polygon is complex and hard to perform spatial operations on, we need to create a buffer zone around each forest
# boundary. In addition, learning information around each forest is also meaning ful.

# forest_buffer = use_proximity.create_buffers(forest_collection.layers[0], dist ances=[20], units = 'Miles')
# forest_buffer_sdf = forest_buffer.query().sdf
# forest_buffer_fl = forest_buffer_sdf.spatial.to_featurelayer(title ='forest_bu ffer', gis=agol, tags="forest")

# create buffer requires a lot of time, to make it quicker, we read in the buffe r we created before
serviceURL5 = 'https://services1.arcgis.com/eGSDp8lpKe5izqVc/arcgis/rest/service s/a2bff6/FeatureServer'
collection = FeatureLayerCollection(serviceURL5, gis=agol)
forest_buffer_sdf = collection.layers[0].query(out_sr=3857).sdf
forest_buffer_fl = forest_buffer_sdf.spatial.to_featurelayer(title ='forest_buff er', gis=agol, tags="forest")

# 2.SpatialJoin the current wildfire data with AIO
# Count how many wildfires and their severities in forest service zone
sum_fields = ['DAILYACRES Sum']
forest_wildfire_sum = summarize_data.aggregate_points(
point_layer=fire_collection.layers[0], polygon_layer=forest_buffer_fl.layers[0], keep_boundaries_with_no_points=True, summary_fields=sum_fields)

forest_with_wildfire_sdf = forest_wildfire_sum['aggregated_layer'].query().sdf

forest_with_wildfire_cleaned = forest_with_wildfire_sdf[["forestname", "sum_dail yacres", "Point_Count", "SHAPE"]]
forest_with_wildfire_cleaned.columns = ["FORESTNAME", "Fire_severity", "Num_fir e", "Service_zone_shape"]
forest_with_wildfire_cleaned["Fire_severity"] = forest_with_wildfire_cleaned["Fi re_severity"].fillna(0)
#forest_with_wildfire_cleaned.head(3)

# 3.SpatialJoin the burned area data with AIO
# Count the burned area scaled by time in forest service zone
# For simplification, we find center of each polygon (there is some issue in pol ygon join)
burned_sdf_point = burned_sdf.assign(SHAPE = burned_sdf["SHAPE"].apply(lambda x: geometry.Point({"x":x.centroid[0],"y":x.centroid[1], "spatialReference":{"wkid": 102100}})))

# Scale the burned area by time
date_time_series = burned_sdf_point["StartDate"].apply(lambda x: pd.to_datetime( x))
time_delta_series = date_time_series.apply(lambda x: pd.Timedelta(pd.to_datetime ('2020-01-01') - x).days)
influence_series = time_delta_series.apply(lambda x: 2**(-x/1000))

burned_sdf_point["Area_adjusted_bytime"] = influence_series*burned_sdf_point["Ac res"]
```

```
forest_buffer_with_burned_area_nogroupby = forest_buffer_sdf.spatial.join(burned
_sdf_point, how = "left", op = "contains")

forest_buffer_with_burned_area = forest_buffer_with_burned_area_nogroupby.groupb
y("forestname").sum().reset_index()

forest_buffer_with_burned_area_plot = forest_buffer_with_burned_area.merge(fores
t_with_wildfire_cleaned, left_on = "forestname", right_on = "FORESTNAME")

forest_buffer_with_burned_area_plot["SHAPE"] = forest_buffer_with_burned_area_pl
ot["Service_zone_shape"]

# 4.SpatialJoin the beetles data with AIO
# Count how many beetles in forest service zone
sum_fields = ['ID Sum']
forest_beetle_sum = summarize_data.aggregate_points(
point_layer=beetle_fl, polygon_layer=forest_buffer_fl.layers[0],
keep_boundaries_with_no_points=True, summary_fields=sum_fields)

forest_with_beetle_sdf = forest_beetle_sum['aggregated_layer'].query().sdf

forest_with_beetle_cleaned = forest_with_beetle_sdf[["forestname", "Point_Count"
]]
forest_with_beetle_cleaned.columns = ["FORESTNAME", "Num_beetle"]
#forest_with_beetle_cleaned.head(3)

# 5.SpatialJoin the freeway data with AIO
# Count how many freeways cross forest service zone
sdf_with_freeways = forest_buffer_sdf.spatial.join(freeway_sdf, how = "left", op
= "intersects")

forest_with_freeway_cleaned = sdf_with_freeways.groupby("forestname").count()[[
"fid"]].reset_index()
forest_with_freeway_cleaned.columns = ["FORESTNAME", "Num_freeway"]
forest_with_freeway_cleaned.head(3)

# 6.SpatialJoin the climate data with AIO
# Calculate the mean precipitation and temperature in forest service zone
sum_fields = ['Mean_T_f_07_Jul Mean', 'Mean_mmPr_07_Jul Mean']
forest_climate_mean = summarize_data.aggregate_points(
point_layer=climate_fl, polygon_layer=forest_buffer_fl.layers[0],
keep_boundaries_with_no_points=True, summary_fields=sum_fields)

forest_with_climate_sdf = forest_climate_mean['aggregated_layer'].query().sdf

forest_with_climate_cleaned = forest_with_climate_sdf[["forestname", "mean_mean_
t_f_07_jul", "mean_mean_mmpr_07_jul"]]
forest_with_climate_cleaned.columns= ["FORESTNAME","Tempetrature", "Precipitatio
n"]
#climate_fl1 = climate_sdf1.spatial.to_featurelayer(title ='climate in US', gis=
agol, tags="climate")
# forest_with_climate_cleaned.head(3)

# 7.Geoenrich forest service zone with population data
forest_with_population = enrich(study_areas=forest_buffer_sdf, data_collections=
["Population"], return_geometry=False)

forest_with_population_cleaned = forest_with_population[["forestname", "ACSTOTPO
```

```
P"]]
forest_with_population_cleaned.columns = ["FORESTNAME", "Population"]
#forest_with_population_cleaned.head(3)

# 8.Merge all the information into a single sedf
df = forest_with_population_cleaned.merge(forest_with_beetle_cleaned).merge(fore
st_with_freeway_cleaned).merge(forest_with_freeway_cleaned).merge(forest_with_cl
imate_cleaned).merge(forest_with_wildfire_cleaned)
forest_polygon = sdf[["FORESTNAME","SHAPE"]]
forest_info = df.merge(forest_polygon)
forest_info.head(3)
```

Out[12]:

	FORESTNAME	Population	Num_beetle	Num_freeway	Tempetrature	Precipitation	Fire_sev
0	Superior National Forest	116675	0	1	65.119379	3.669830	
1	Fremont-Winema National Forest	88825	0	1	63.740621	0.356109	
2	Stanislaus National Forest	145984	0	1	69.344986	0.211570	

7. Descriptive statistics for the data

We plot some of our feature data and give a statistical summary below. There is an evidence of spatial auto-correlation. Areas that are close to each other tend to have similar value in population, temperature, precipitation, fire severity, beetles, and number of freeways crossing them.

In [13]:

```
# Map for forest service zone
m = agol.map("USA",zoomlevel = 4)
m.add_layer(forest_buffer)
m
```

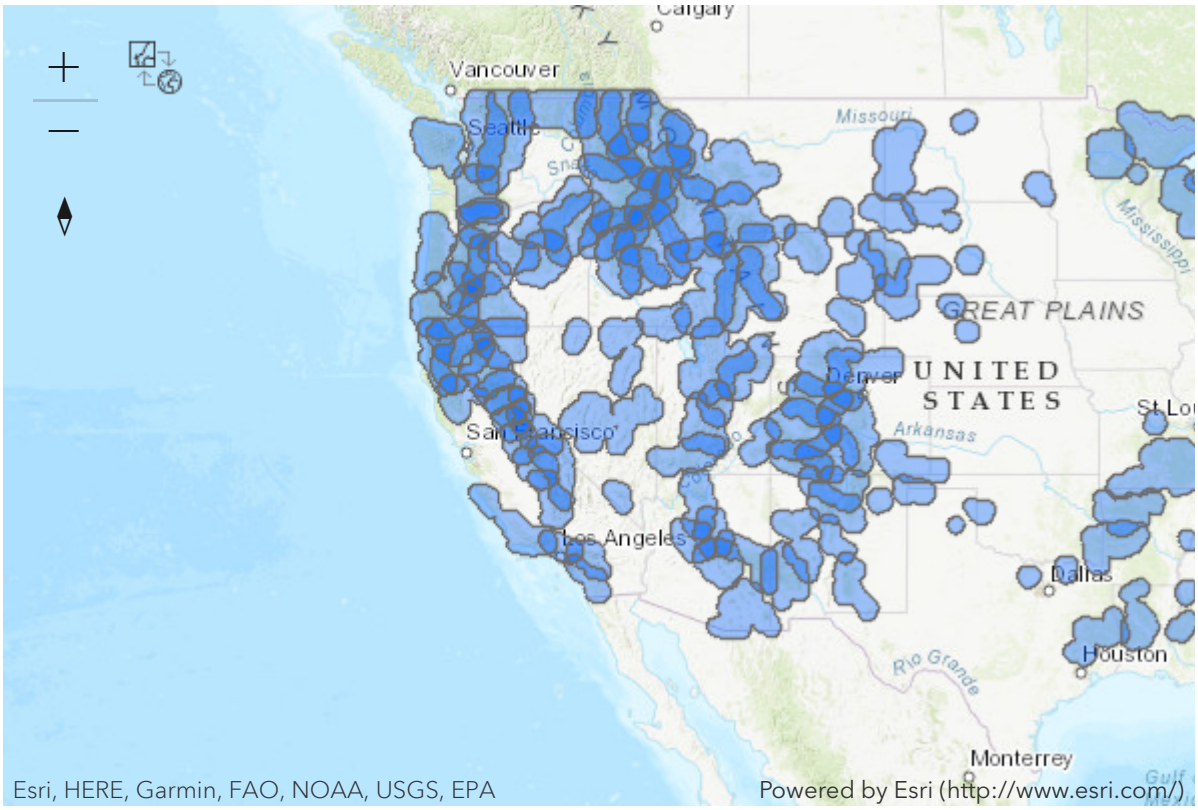
In [14]:

```
# Map for current wildfire severity in forest service zone
m1 = agol.map("USA", zoomlevel = 4)
forest_with_wildfire_cleaned.spatial.plot(map_widget=m1,renderer_type='c',method
='esriClassifyNaturalBreaks', class_count=10, col='Fire_severity', alpha=0.7, cm
ap = "OrRd")
m1
```

We plot some of our feature data and give a statistical summary below. There is an evidence of spatial auto-correlation. Areas that are close to each other tend to have similar value in population, temperature, precipitation, fire severity, beetles, and number of freeways crossing them.

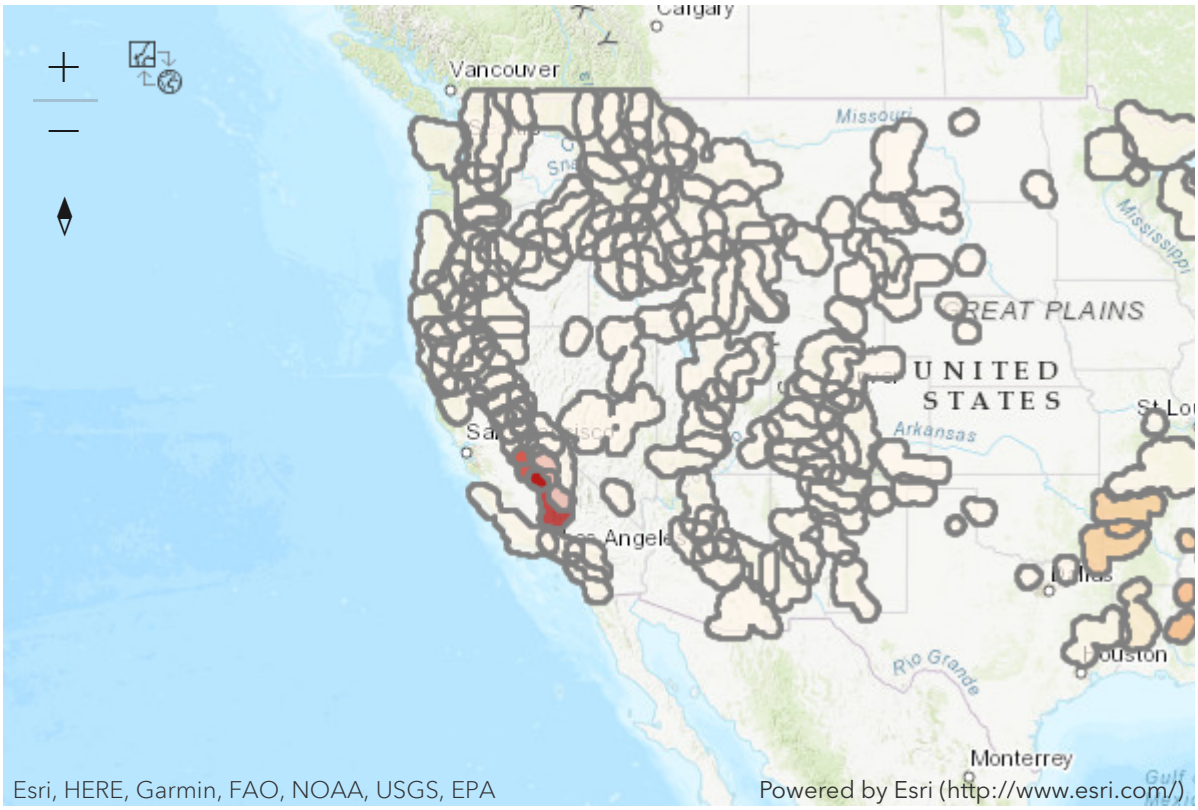
In [13]:

```
# Map for forest service zone
m = agol.map("USA",zoomlevel = 4)
m.add_layer(forest_buffer)
m
```



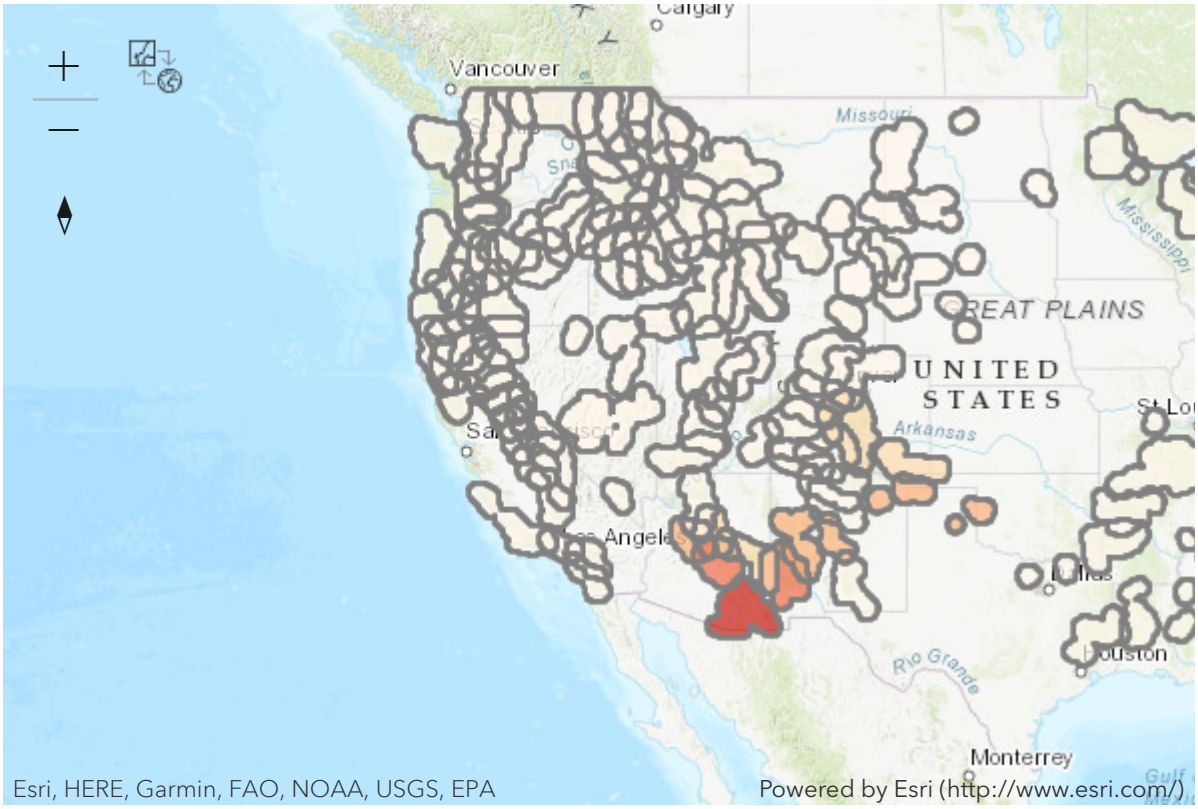
In [14]:

```
# Map for current wildfire severity in forest service zone
m1 = agol.map("USA", zoomlevel = 4)
forest_with_wildfire_cleaned.spatial.plot(map_widget=m1,renderer_type='c',methoc
m1
```



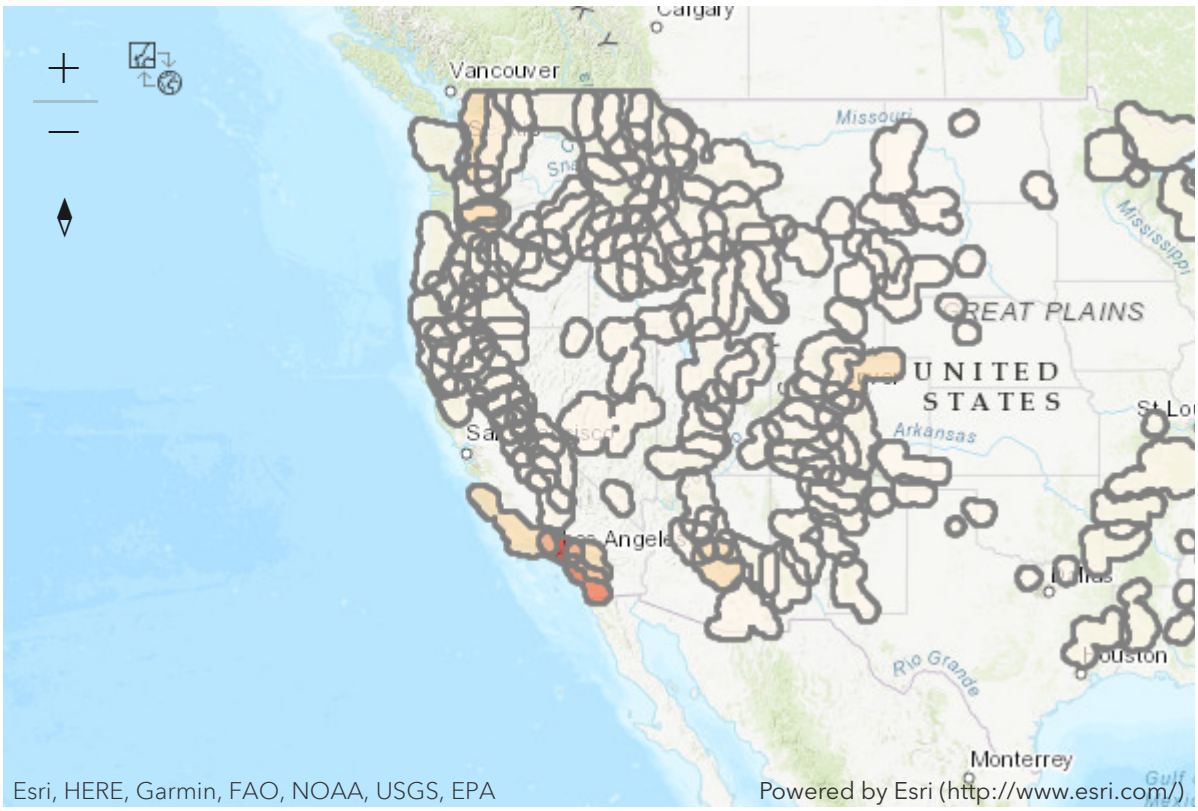
In [15]:

```
# Map for the beetles in forest service zone
m2 = agol.map("USA", zoomlevel = 4)
df_to_plot = forest_with_beetle_cleaned.merge(forest_with_wildfire_cleaned)
df_to_plot["SHAPE"] = df_to_plot["Service_zone_shape"]
df_to_plot.spatial.plot(map_widget=m2,renderer_type='c',method='esriClassifyNatu
m2
```



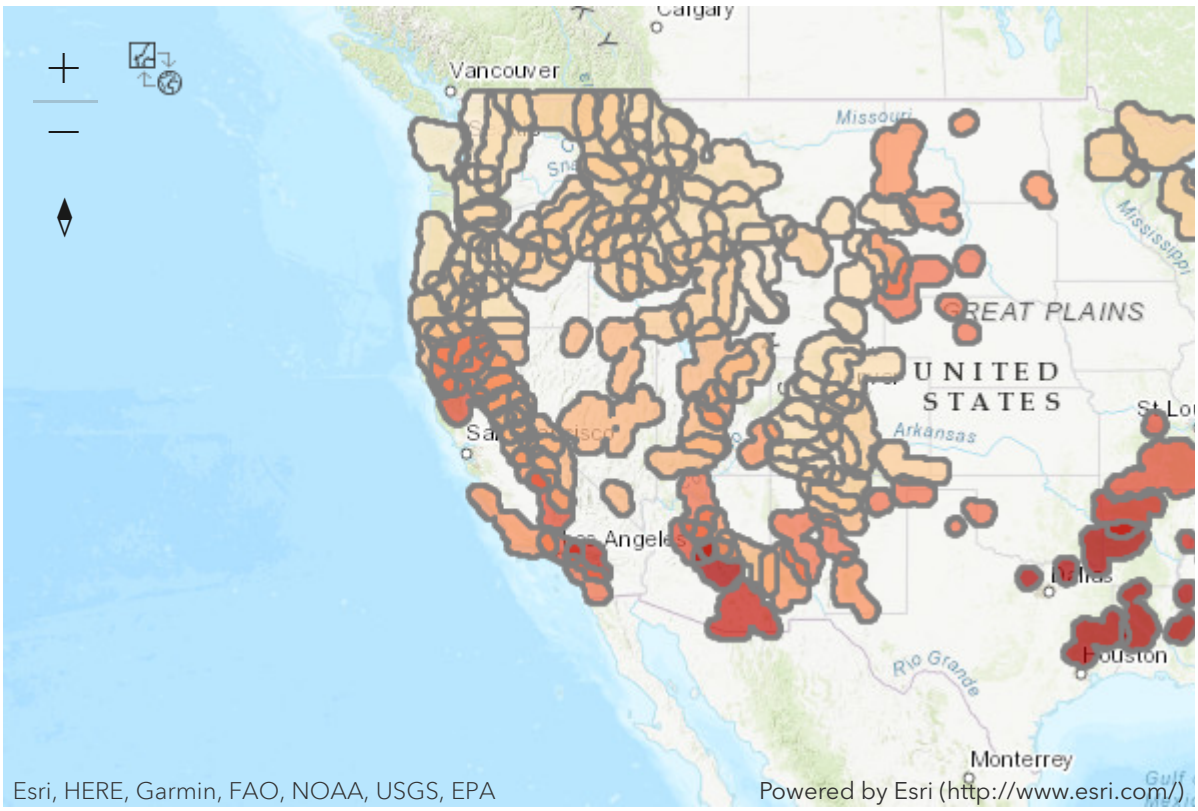
In [16]:

```
# Map for the number of freeways crossing forest service zone
m3 = agol.map("USA", zoomlevel = 4)
df_to_plot1 = forest_with_freeway_cleaned.merge(forest_with_wildfire_cleaned)
df_to_plot1["SHAPE"] = df_to_plot1["Service_zone_shape"]
df_to_plot1.spatial.plot(map_widget=m3,renderer_type='c',method='esriClassifyNat
m3
```



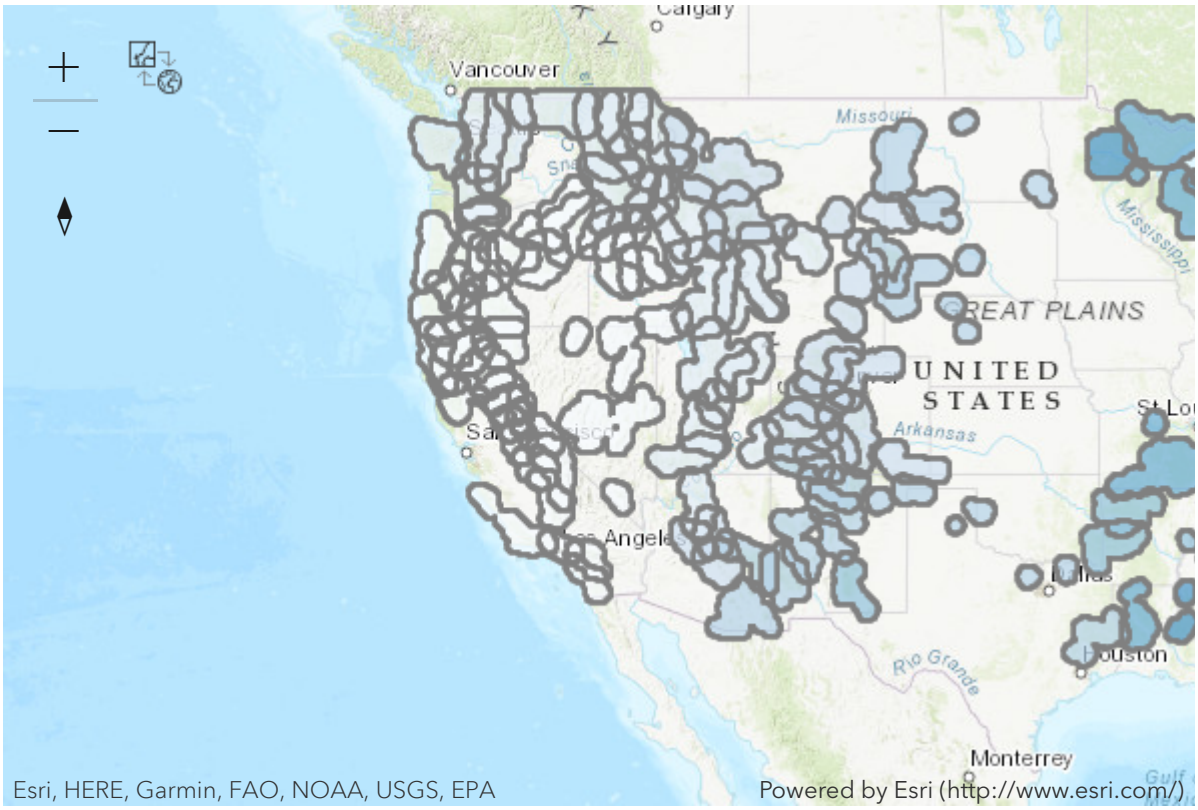
In [17]:

```
# Map for the average temperature in forest service zone
m4 = agol.map("USA", zoomlevel = 4)
df_to_plot2 = forest_with_climate_cleaned.merge(forest_with_wildfire_cleaned)
df_to_plot2["SHAPE"] = df_to_plot2["Service_zone_shape"]
df_to_plot2.spatial.plot(map_widget=m4,renderer_type='c',method='esriClassifyNat
m4
```



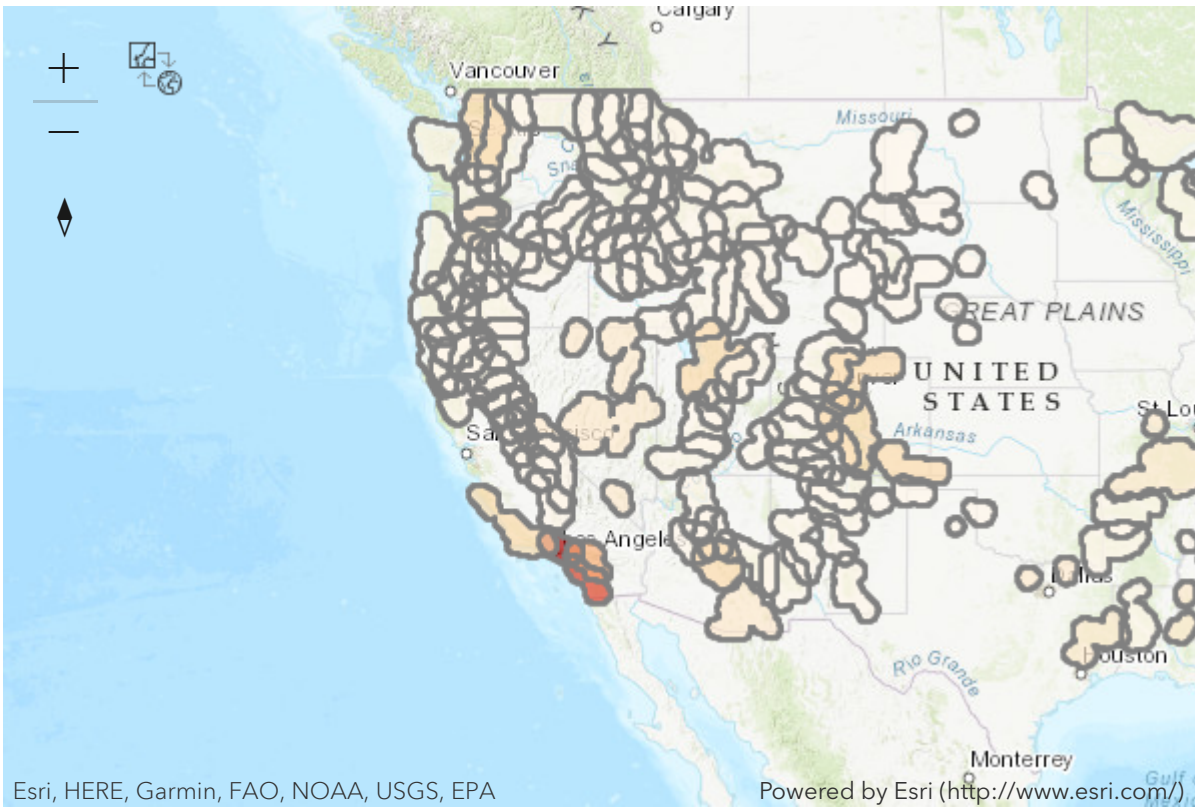
In [18]:

```
# Map for the average precipitation in forest service zone
m5 = agol.map("USA", zoomlevel = 4)
df_to_plot2.spatial.plot(map_widget=m5,renderer_type='c',method='esriClassifyNat
m5
```



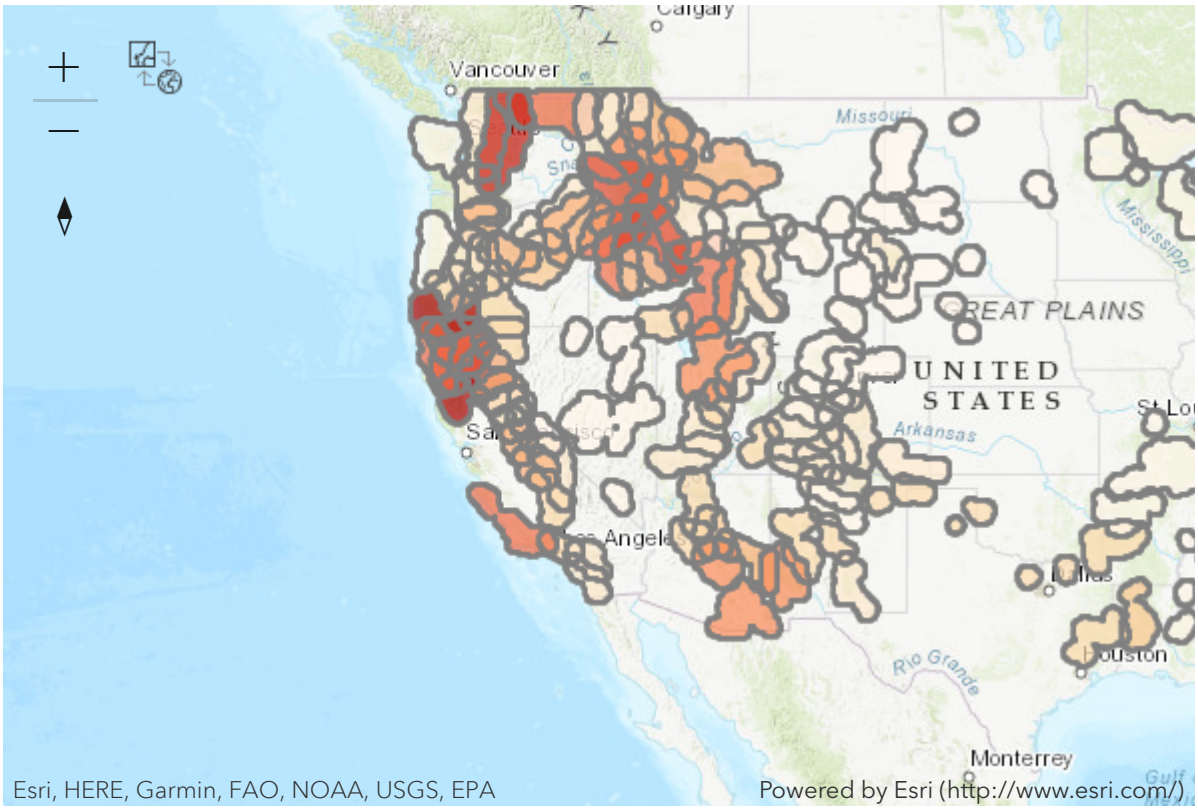
In [19]:

```
# Map for population in forest service zone
m6 = agol.map("USA", zoomlevel = 4)
df_to_plot3 = forest_with_population_cleaned.merge(forest_with_wildfire_cleaned)
df_to_plot3["SHAPE"] = df_to_plot3["Service_zone_shape"]
df_to_plot3.spatial.plot(map_widget=m6,renderer_type='c',method='esriClassifyNat
m6
```



In [20]:

```
# Map for the burned area in forest service zone
m7 = agol.map("USA", zoomlevel = 4)
forest_buffer_with_burned_area_plot.spatial.plot(map_widget=m7,renderer_type='c'
m7
```



In [21]:

```
# what are the summary statistics for each feature
forest_info.describe()
```

Out[21]:

	Population	Num_beetle	Num_freeway	Tempetrature	Precipitation	Fire_severity	Nu
count	1.120000e+02	112.000000	112.000000	112.000000	112.000000	112.000000	112.(
mean	9.022142e+05	1.125000	3.348214	69.028087	1.853655	3679.089286	3.(
std	1.530918e+06	4.431917	4.220734	6.284971	1.595966	15444.813449	7.4
min	1.485900e+04	0.000000	1.000000	56.416229	0.014739	0.000000	0.(
25%	1.474578e+05	0.000000	1.000000	64.454030	0.503199	0.000000	0.(
50%	3.389630e+05	0.000000	2.000000	67.581811	1.321901	0.000000	1.(
75%	1.079576e+06	0.000000	4.000000	73.674665	3.118663	66.500000	3.(
max	1.099693e+07	29.000000	28.000000	82.919089	6.625921	96682.000000	47.(

8. Analysis

- Investigation of deforestation

In [20]:

```
# Map for the burned area in forest service zone
m7 = agol.map("USA", zoomlevel = 4)
forest_buffer_with_burned_area_plot.spatial.plot(map_widget=m7,renderer_type='c'
,method='esriClassifyNaturalBreaks', class_count=10, col='Area_adjusted_bytime',
alpha=0.7, cmap = "OrRd")
m7
```

In [21]:

```
# what are the summary statistics for each feature
forest_info.describe()
```

Out[21]:

	Population	Num_beetle	Num_freeway	Tempetrature	Precipitation	Fire_severity	
count	1.120000e+02	112.000000	112.000000	112.000000	112.000000	112.000000	11
mean	9.022142e+05	1.125000	3.348214	69.028087	1.853655	3679.089286	
std	1.530918e+06	4.431917	4.220734	6.284971	1.595966	15444.813449	
min	1.485900e+04	0.000000	1.000000	56.416229	0.014739	0.000000	
25%	1.474578e+05	0.000000	1.000000	64.454030	0.503199	0.000000	
50%	3.389630e+05	0.000000	2.000000	67.581811	1.321901	0.000000	
75%	1.079576e+06	0.000000	4.000000	73.674665	3.118663	66.500000	
max	1.099693e+07	29.000000	28.000000	82.919089	6.625921	96682.000000	4

8. Analysis

- Investigation of deforestation
  - Define area of interest
  - Define deforestation
  - Evaluate deforestation
  - Calculate deforestation scores
  - Generalize model
- Deforestation prediction
  - Data merge/cleaning
  - Model training/perdiction on deforestation scores using selected features
  - Understand each fators importance
- Reflection on our analysis

8.1 Investigation of deforestation

First, we want to find the quantitative analysis on the deforestation states of all the US national forests. To acquire this result, we first perform a case study on one of the national forest and work out the procedure of defining and calculating the deforestation scores, and then we perform a generalization on every forests to acquire of quantitative results.

8.1.1 Define area of Interests

We want to find the deforestation status for the national foresta in US. We define these two function to

In [22]:

```
def area_interest(i):
    extent_array = gdf.iloc[i:i+1,:].to_crs(3857).total_bounds
    study_area_extent = {'xmin': extent_array[0], 'ymin': extent_array[1], 'xma
x': extent_array[2], 'ymax': extent_array[3], 'spatialReference': {'latestwkid':
3857, 'wkid': 102100}}
    #sedf = GeoAccessor.from_geodataframe(gdf.iloc[i:i+1,:].to_crs(3857), column
_name="geometry")
    #study_area_geom = sedf.geometry.iloc[0]
    study_area_geom = sdf["SHAPE"].iloc[i]
    return study_area_extent,study_area_geom
def clip_image(i, image):
    extent_array = gdf.iloc[i:i+1,:].to_crs(3857).total_bounds
    study_area_extent = {'xmin': extent_array[0], 'ymin': extent_array[1], 'xma
x': extent_array[2], 'ymax': extent_array[3], 'spatialReference': {'latestwkid':
3857, 'wkid': 102100}}
    #sedf = GeoAccessor.from_geodataframe(gdf.iloc[i:i+1,:].to_crs(3857), column
_name="geometry")
    #study_area_geom = sedf.geometry.iloc[0]
    study_area_geom = sdf["SHAPE"].iloc[i]
    new_image = image
    new_image.extent = study_area_extent
    image_clip = clip(raster=image, geometry=study_area_geom)
    return study_area_extent,study_area_geom,image_clip
```

8.1.2 Define deforestation (From lecture)

Originally, we proposed to find deforestation area by employing different methods including machine learning and k-means. However, after a series of investigation, we want to simplify our project and mainly focus on using NVDI as our primary indicator for deforestation.

From lecture, we know that NDVI (http://desktop.arcgis.com/en/arcmap/latest/manage-data/raster-and-images/band-arithmetic-function.htm) is a good indicator for the vegatation status, which we would love to use to evaluate the deforestation status.

The value of NDVI:

```
NDVI = ((NIR - Red)/(NIR + Red))
```

Normalized Difference Vegetation Index (NDVI) quantifies vegetation by measuring the difference between near-infrared (which vegetation strongly reflects) and red light (which vegetation absorbs).

NDVI always ranges from -1 to +1. But there isn't a distinct boundary for each type of land cover.

In agriculture, farmers use NDVI for precision farming and to measure biomass. In forestry, foresters use NDVI to quantify forest supply and leaf area index.

Furthermore, NASA states that NDVI is a good indicator of drought. When water limits vegetation growth, it has a lower relative NDVI and density of vegetation.

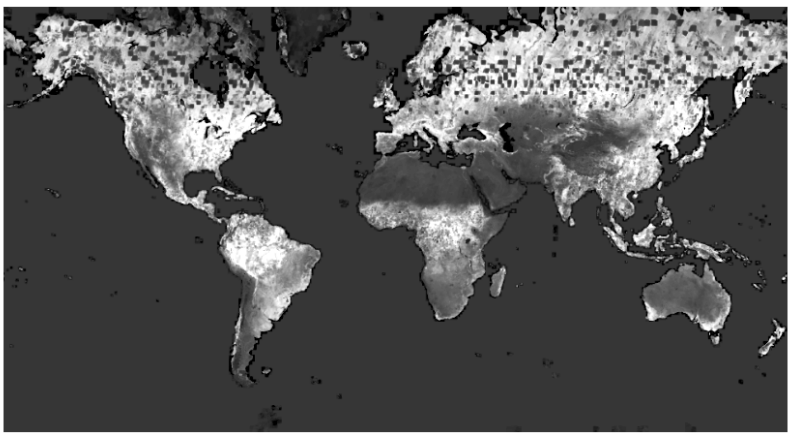
In reality, there are hundreds of applications where NDVI and other remote sensing applications is being applied to in the real world.

After exploring the functions in sentinel-2, we decided to use the NDVI Raw function to calculate our nvdi base raster layer. We also tryied NVDI color function for visualization, but it cannot procede to furture numerical calculations, so we commented them out below.

In [23]:

```
# nvdi_color_sentinel = apply(sentinel, 'NDVI Colormap')
# nvdi_color_sentinel

nvdi_sentinel = apply(sentinel, 'NDVI Raw')
nvdi_sentinel
```

Out[23]:



8.1.3 Evaluate deforestation: Case study 1 (Superior national forest)

We use our first national forest to evaluate deforestation and define how to calculate our deforestation score. We first clip the image to the show the satelite imagery on the first forest.

In [24]:

```
study_area_extent,study_area_geom,image_clip_world = clip_image(0,world)
# area_map = agol.map("USA",4)
# area_map.add_layer(image_clip_world)
# area_map.legend = True
# area_map
image_clip_world
```

Out[24]:

From the above map, we notice that the center part of the ground have experience some deforestation. We want to get a quantitative evaluation on the deforestation status, then we clip our nvdi base raster on this area to show the NVDI index on our national forest.

In [25]:

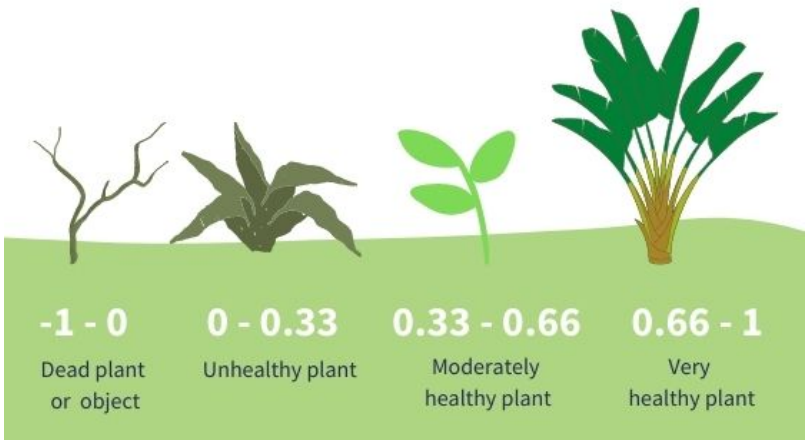
```
study_area_extent,study_area_geom,image_clip = clip_image(0,nvdi_sentinel)
# area_map_nvdi = agol.map("USA",4)
# area_map_nvdi.add_layer(image_clip)
# area_map_nvdi.legend = True
# area_map_nvdi
image_clip
```

Out[25]:

We notice that due to some reason, the map cannot show the NVDI clearly, then we will try to evaluation numerically by remapping.

8.1.3 Evaluation (Cont.): Remap NVDI

According to NDVI, the Foundation for Remote Sensing Phenology (https://www.usgs.gov/special-topics/remote-sensing-phenology/science/ndvi-foundation-remote-sensing-phenolog), NDVI values range from +1.0 to -1.0. Areas of barren rock, sand, or snow usually show very low NDVI values (for example, 0.1 or less). Sparse vegetation such as shrubs and grasslands or senescing crops may result in moderate NDVI values (approximately 0.2 to 0.5). High NDVI values (approximately 0.6 to 0.9) correspond to dense vegetation such as that found in temperate and tropical forests or crops at their peak growth stage. According to 5 Things To Know About NDVI (Normalized Difference Vegetation Index) (https://up42.com/blog/tech/5-things-to-know-about-ndvi), there are severl cutoffs to determine the status of the plants shown below.



Thus, we remap our NVDI calculation based on the following cateria,

NVDI	Deforestation Severity
-1 to 0.1	Severe deforestation
0.1 to 0.33	Medium deforestation
0.33 to 0.66	Some deforestation
0.66 to 1	Negligible/No deforestation

and remap using color

	R	G	B
Red	255	0	0
Severe	255	100	0
Orange	255	150	0
Medium	255	200	0
Yellow	255	255	0
Some	200	255	0
Chartreuse	150	255	0
Negligible/No	100	255	0
Green	0	255	0

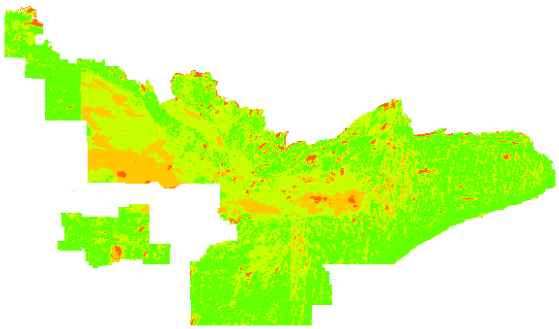
In [26]:

```
clrmap4 =  [[0, 255, 100, 0], [1, 255, 200, 0], [2, 200, 255, 0], [3, 100, 255,
0]]#, [5, 0, 255, 0]]
deforestation_areas = colormap(remap(image_clip,
                               input_ranges=[-1,  0.1,  # Severe deforestation
                                             0.1, 0.33,  # Medium deforestation
                                             0.33, 0.66,  # Some deforestation
                                             0.66, 1.00], # Negligible/No deforest
ation
                               output_values=[0, 1, 2, 3]),
                         colormap=clrmap4)
```

In [27]:

```
deforestation_areas
```

Out[27]:



8.1.3 Evaluation (Cont.): Calculate histogram

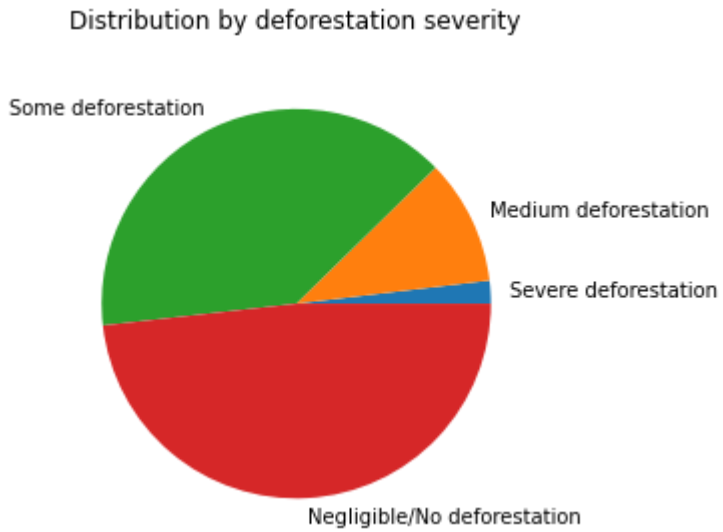
After invetigate some of the forest, we decided to chosse pixel value as 1000 since it has the most accuract result.

In [28]:

```
pixx = 1000
pixy = 1000

res = deforestation_areas.compute_histograms(study_area_extent, pixel_size={'x':
pixx, 'y':pixy})

numpix = 0
histogram = res['histograms'][0]['counts']
for i in histogram:
    numpix += i
```

In [29]:

```
# plt.plot(histogram)
plt.title('Distribution by deforestation severity',y = 1.1)#, y=-0.1)
plt.pie(histogram, labels=['Severe deforestation','Medium deforestation',
                           'Some deforestation','Negligible/No deforestation']);
plt.axis('equal');
```



8.1.4 Define and calculate deforestation score

Since we want to have a numerical value to acess the deforestation status for different national forest, we calculate the score using formula

```
score = severe deforestation (%) * 1 + medium deforestation (%) * 0.5 + so
me deforestation (%) * 0.2
```

In [30]:

```
def deforest_score(hist):
    return 1*hist[0]/np.sum(hist) + 0.5*hist[1]/np.sum(hist) + 0.2*hist[2]/np.su
m(hist)

deforestation_score = deforest_score(histogram)
deforestation_score
```

Out[30]:

```
0.15003711528578773
```

8.1.5 Generalize model: Calculate deforestation score for different national forest

In [31]:

```
clrmap4 =  [[0, 255, 100, 0], [1, 255, 200, 0], [2, 200, 255, 0], [3, 100, 255,
0]]
def national_forest_deforestation(forest_id, nvdi_layer, pixx=1000, pixy=1000):
    """
    Find the deforestation score as we defined before
    """
    study_area_extent,study_area_geom,image_clip = clip_image(forest_id,nvdi_lay
er)
    deforestation_areas = colormap(remap(image_clip,
                                         input_ranges=[-1,  0.1,  # Severe defor
estation
                                           0.1, 0.33,  # Medium deforestation
                                           0.33, 0.66,  # Some deforestation
                                           0.66, 1.00], # Negligible/No deforest
ation
                                         output_values=[0, 1, 2, 3]),
                                   colormap=clrmap4)
#     pixx = (study_area_extent['xmax'] - study_area_extent['xmin']) / 1000.0
#     pixy = (study_area_extent['ymax'] - study_area_extent['ymin']) / 1000.0

    res = deforestation_areas.compute_histograms(study_area_extent, pixel_size={
'x':pixx, 'y':pixy})
    numpix = 0
    histogram = res['histograms'][0]['counts']
    if res['histograms'][0]['max'] == 2.5:
        histogram.append(0)
    for i in histogram:
        numpix += i

    score = deforest_score(histogram)
    return deforestation_areas, histogram, score
```

In [32]:

```
def plot_deforestaion_histogram(histogram, forest_name):
    plt.title(forest_name + ': Distribution by deforestation severity',y = 1.1)
#, y=-0.1)
    plt.pie(histogram, labels=['Severe deforestation','Medium deforestation',
                               'Some deforestation','Negligible/No deforestatio
n']);
    plt.axis('equal');
```

Since this calculation of scores needs a lot of time, we skip this step and read in our previously calculated scores.

In [33]:

```
# index = np.arange(len(gdf))
# scores = []
# deforestation_layer = []
# histogram_list = []
# for i in index:
#     if i == 56 or i == 106 or i ==107:
#         score = np.nan
#         histogram = np.nan
#         deforestation_areas = np.nan
#     else:
#         try:
#             deforestation_areas, histogram, score = national_forest_deforestat
ion(i,nvdi_sentinel)
#         except:
#             score = np.nan
#             histogram = np.nan
#             deforestation_areas = np.nan
#         if i%10 == 0:
#             print(str(i)+" images processed")
#     scores.append(score)
#     histogram_list.append(histogram)
#     deforestation_layer.append(deforestation_areas)
```

In [34]:

```
# calculated_results = pd.DataFrame({"forest_id":index,"deforestation_score":sco
res})
# calculated_results.to_csv("forest_deforestation_all.csv",index = False)
# calculated_results
deforestation_scores
```

Out[34]:

	forest_id	deforestation_score
0	0	0.150037
1	1	0.280932
2	2	0.315803
3	3	0.298468
4	4	0.404013
...	...	...
107	107	NaN
108	108	0.280268
109	109	0.390748
110	110	0.206412
111	111	0.412516

112 rows × 2 columns

8.1.6 Compare our model with image

In [35]:

```
def human_evaluate(i):
    study_area_extent,study_area_geom,image_clip_world = clip_image(i,world)
    deforestation, hist, score = national_forest_deforestation(i, nvdi_sentinel)
    return image_clip_world, deforestation, hist, score
```

In [36]:

```
image_clip_world, deforestation, hist, score = human_evaluate(36)
image_clip_world
```

Out[36]:

In [37]:

```
deforestation
```

Out[37]:

In [38]:

```
score
```

Out[38]:

```
0.058603015674354844
```

In [39]:

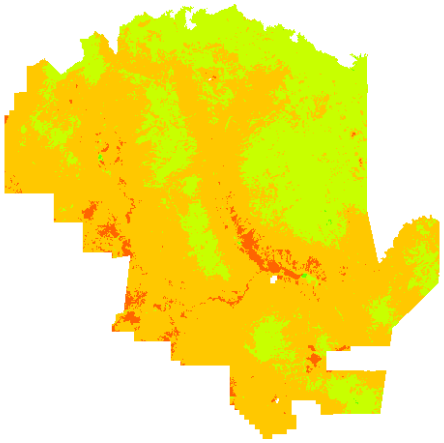
```
image_clip_world, deforestation, hist, score = human_evaluate(111)
image_clip_world
```

Out[39]:

In [53]:

```
deforestation
```

Out[53]:



In [41]:

```
score
```

Out[41]:

0.41251571608183796

8.1.7 Potential Problems

1. Capture dates from sentinel-2: We may experience different capture date for different parts of the US national forest. One may come from summer while other may come from winter, which the vegetation may not be the same. Moreover, the image may also not reflect the current states of the national forest since several wildfire had happened near some forest, which may not be captured by sentinel-2.
2. Pixel problem: We noticed that there are several different resolution for the sentinel-2 data. In our experiment, we found that for some forest, the score calculate varies a lot if we use different pixel size. We suspect that the issue come from different resolution image were taken on different seasons.

Although there are problems, we believe that our results remain a good indicator on the deforestation status for each national forest since the score align with the natural look of the ground. Moreover, we will investigate more to the previous problems in the future to refine our results.

8.2 Deforestation prediction

Since now we have all the feature data we want to analyze on and we also have our deforestation score calculated. We can now perform our prediction and analysis on answering the questions `What are the special characteristics of areas suffering from deforestation?` and `What are the factors that contribute most to the deforestation?` .

8.2.1 Merge Data

`new_fire` is just a added feature that was not in the forest_info_sdf yet. We added it by another csv as explained in section 5.3.\ `forest_info` is the forest data as indicated in section 6. However since some of the data will change as time goes, we want to include the original data we did before, we upload a old version of forest info into arcGIS, and we will use that version for perdiciton. \ `deforestation_scores` is the calculated deforestation score as explained in section 5.3 and 8.\ We join our deforestation score with features together here

In [42]:

```
# Read in old forest info data
forestinfoURL = 'https://services1.arcgis.com/eGSDp8lpKe5izqVc/arcgis/rest/servi
ces/ae2986/FeatureServer'
forest_info_fc = FeatureLayerCollection(forestinfoURL, gis=agol)
forest_info_fl = forest_info_fc.layers[0]
forest_info_sdf = forest_info_fl.query(out_sr=3857).sdf
```

In [43]:

```
forest_info_sdf = forest_info_sdf.merge(new_fire, left_on = "forestname", right_
on = "FORESTNAME")
new_forest_label = deforestation_scores.merge(forest_info_sdf, left_on = 'forest
_id', how = 'left', right_index = True)
new_forest_label.head(2)
```

Out[43]:

	forest_id	deforestation_score	FID_x	forestname	population	num_beetle	num_freewa	ter
0	0	0.150037	1	Superior National Forest	116675	0	1	6
1	1	0.280932	2	Fremont-Winema National Forest	88825	0	1	6

8.2.2 Model Data Cleaning

With labels from Judy and features from Xuhui, we can build prediction model now.

In [44]:

```
new_forest_label_clean = new_forest_label[['forestname','population','num_beetl
e','num_freewa','tempetratu','precipitat','fire_sever','num_fire',"Area_adjusted
_bytime", 'deforestation_score']]
new_forest_label_clean = new_forest_label_clean.dropna()
new_forest_label_clean.head()
```

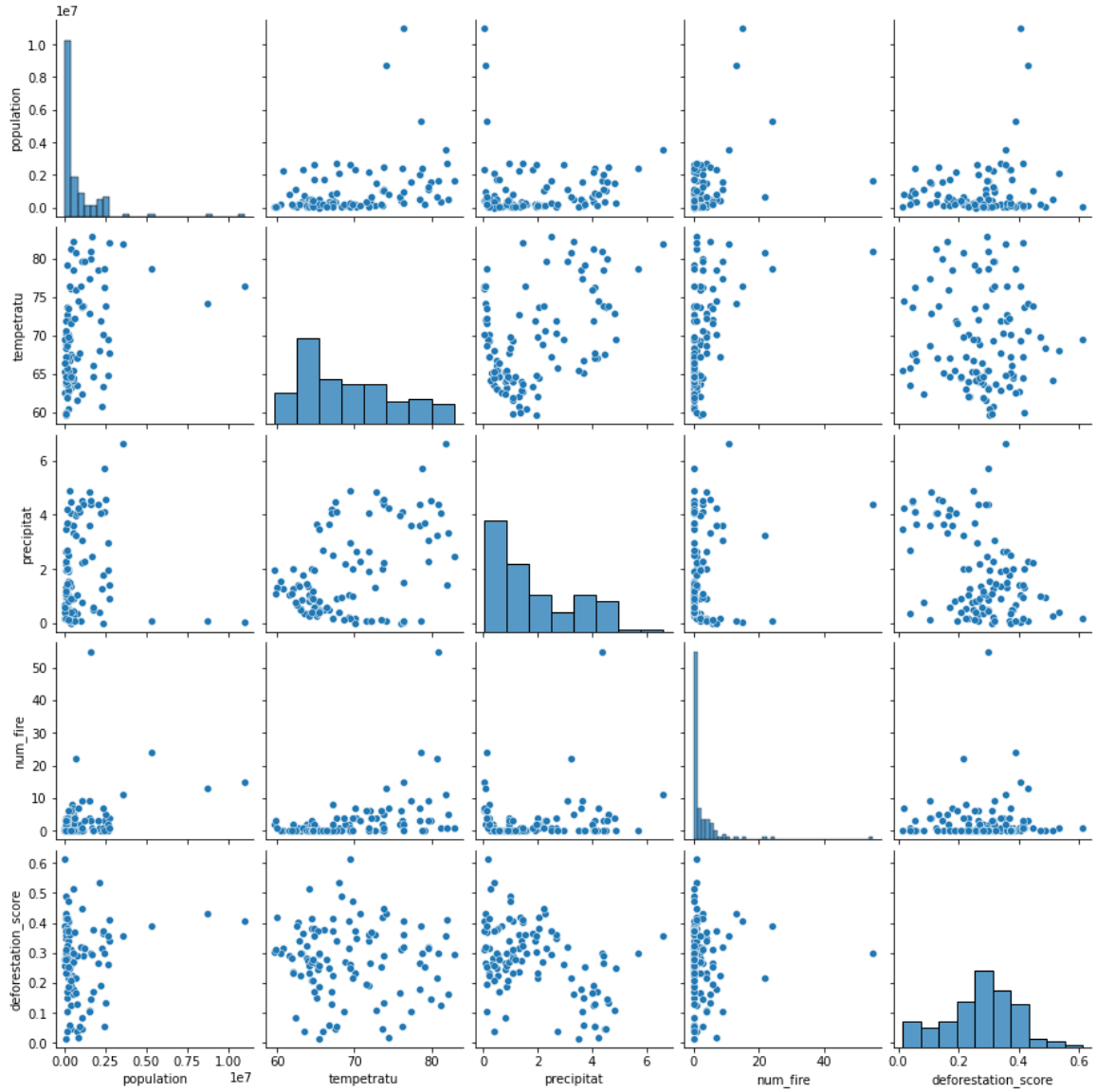
Out[44]:

	forestname	population	num_beetle	num_freewa	tempetratu	precipitat	fire_sever	num_fi
0	Superior National Forest	116675	0	1	65.119379	3.669830	0.0	
1	Fremont-Winema National Forest	88825	0	1	63.740621	0.356109	9754.0	
2	Stanislaus National Forest	145984	0	1	69.344986	0.211570	0.0	
3	Okanogan-Wenatchee National Forest	364271	0	2	63.098937	0.864127	0.0	
4	Medicine Bow-Routt National Forest	202269	0	2	62.619650	1.432083	1.0	

First, we want to have a rough idea about the correlation between each variable.

In [45]:

```
sns.pairplot(new_forest_label_clean, vars=('population','tempetratu','precipitat','num_fire','deforestation_score'));
```



We first drop outliers

In [46]:

```
Q3 = new_forest_label_clean['deforestation_score'].describe()['75%']
Q1 = new_forest_label_clean['deforestation_score'].describe()['25%']
lower_out = Q1 - 1.5* (Q3-Q1)
upper_out = Q3 + 1.5* (Q3-Q1)
```

In [47]:

```
new_forest_label_clean = new_forest_label_clean[(new_forest_label_clean['defores
tation_score']<upper_out) & (new_forest_label_clean['deforestation_score']>lower
_out)]
```

Then, we standardize all variables.

In [48]:

```
normalized_df=(new_forest_label_clean-new_forest_label_clean.mean())/new_forest_
label_clean.std()
new_noramlized_df = pd.concat([new_forest_label_clean[['forestname','deforestati
on_score']],normalized_df[['fire_sever','num_beetle','num_fire','num_freewa','po
pulation','precipitat','tempetratu', "Area_adjusted_bytime"]]],axis = 1)
new_noramlized_df.head()
```

Out[48]:

	forestname	deforestation_score	fire_sever	num_beetle	num_fire	num_freewa	population
0	Superior National Forest	0.150037	-0.228690	-0.258767	-0.435308	-0.564328	-0.520075
1	Fremont-Winema National Forest	0.280932	0.920934	-0.258767	-0.128032	-0.564328	-0.538002
2	Stanislaus National Forest	0.315803	-0.228690	-0.258767	-0.128032	-0.564328	-0.501209
3	Okanogan-Wenatchee National Forest	0.298468	-0.228690	-0.258767	-0.435308	-0.330813	-0.360700
4	Medicine Bow-Routt National Forest	0.404013	-0.228572	-0.258767	-0.281670	-0.330813	-0.464979

8.2.3 Build Prediction Model: linear model

In [49]:

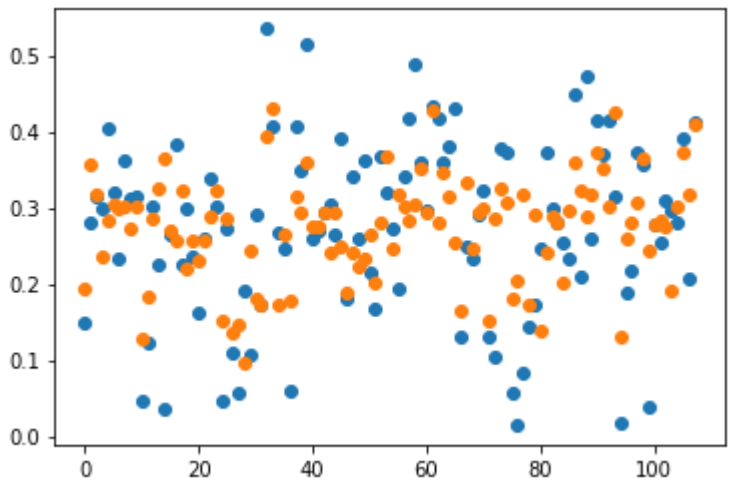
```
lr = LinearRegression()
lr.fit(new_forest_label_clean[['population','num_beetle','num_freewa','tempetrat
u',
                               'precipitat','fire_sever','num_fire', "Area_adjus
ted_bytime"]],
       new_forest_label_clean[['deforestation_score']])

y_predicted = lr.predict(new_forest_label_clean[['population','num_beetle','num_
freewa','tempetratu',
                               'precipitat','fire_sever','num_fire', "Area_adjus
ted_bytime"]])
sklearn.metrics.r2_score(new_forest_label_clean['deforestation_score'], y_predic
ted)
```

Out[49]:

```
0.3600601003214797
```

In [50]:

```
plt.scatter(np.arange(len(y_predicted)),new_noramlized_df[['deforestation_score'
]])
plt.scatter(np.arange(len(y_predicted)),y_predicted);
```



Understand how each feature contribute to the linear model

In [51]:

```
outcome_2, predictors_2 = patsy.dmatrices('deforestation_score ~ population+num_
beetle+num_freewa+tempetratu+precipitat+fire_sever+num_fire+Area_adjusted_bytim
e', new_forest_label_clean)
mod_2 = sm.OLS(outcome_2, predictors_2)
res_2 = mod_2.fit()
```

In [52]:

```
print(res_2.summary())
```

```
                            OLS Regression Results
==============================================================================
============
Dep. Variable:     deforestation_score   R-squared:
0.360
Model:                             OLS   Adj. R-squared:
0.308
Method:                  Least Squares   F-statistic:
6.963
Date:                 Fri, 18 Mar 2022   Prob (F-statistic):
3.00e-07
Time:                         03:09:05   Log-Likelihood:
104.09
No. Observations:                  108   AIC:
-190.2
Df Residuals:                       99   BIC:
-166.0
Df Model:                            8
Covariance Type:             nonrobust
==============================================================================
=====================
                             coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
---------------------
Intercept                  0.2628      0.136      1.926      0.057
-0.008       0.534
population             2.996e-08    1.4e-08      2.145      0.034
2.25e-09    5.77e-08
num_beetle                 0.0053      0.002      2.366      0.020
0.001       0.010
num_freewa                -0.0080      0.005     -1.652      0.102
-0.018       0.002
tempetratu                 0.0017      0.002      0.794      0.429
-0.003       0.006
precipitat                -0.0474      0.009     -5.023      0.000
-0.066      -0.029
fire_sever             4.455e-06   1.52e-06      2.940      0.004
1.45e-06    7.46e-06
num_fire                  -0.0031      0.002     -1.534      0.128
-0.007       0.001
Area_adjusted_bytime -1.624e-07   7.63e-08     -2.129      0.036
-3.14e-07   -1.11e-08
==============================================================================
===========
Omnibus:                        6.219   Durbin-Watson:
1.917
Prob(Omnibus):                  0.045   Jarque-Bera (JB):
5.802
Skew:                          -0.456   Prob(JB):
0.0550
Kurtosis:                       3.676   Cond. No.
2.65e+07
==============================================================================
===========

Notes:
[1] Standard Errors assume that the covariance matrix of the errors
is correctly specified.
```

```
[2] The condition number is large, 2.65e+07. This might indicate th
at there are
strong multicollinearity or other numerical problems.
```

8.3 Reflection on our analysis:

Our analysis does not deviate from our project proposal in general.

Firstly, in our project proposal, we proposed that we need to collect features for each forest area. We accomplished that by spatial joining our forest service zone with different feature layers that contain information about forest service zone such as wildfire, beetles, freeways, temperature, rainfall, and population.

Secondly, in our project proposal, we also proposed that we need to calculate the deforestation score for each US forest area. We originally plan to use method like Kmeans or deep learning to predict proportional of deforestation area in each US national forest. However, we find that it is very difficult to find a high resolution image for each area of interest. In addition, it is extremely time-consuming to run deep learning model on such a huge areas. Thus, we resort to another method which is using ndvi value from sentinel-2 image to predict the deforestation score for each area of interest. It turns out that the result from ndvi value is pretty good as well.

Thirdly, in our project proposal, we mentioned that we want to use a machine learning model to analyze potential cause of deforestation. We complete this step by building a linear regression model. The independent variables are the features we collected and the dependent variable is the deforestation score we computed. Our liner model has a r square of 36 percent which implies a correlation of 0.6, which is not a bad score. We also look at the confidence interval and p-value for each the regression coefficient for each feature so that we can understand whether each feature is positively or negatively correlated with the deforestation score. Thus, we can understand how each factors are associated with deforestation score.

In a short conclusion, we successfully finished all the tasks mentioned in the proporsal.

9 Summary of our result

1. The regression coefficient of precipitation is negative with p-value approximately equals 0. So, we are very confidence that the precipitation is negatively correlated with deforestation score which means more precipitation are related to less deforestation. Since the p-value is the smallest among all features, we would say that the precipitation is the most important factor that contributes to the deforestation
2. The regression coefficient of fire severity is positive with p-value equals $0.004 < 0.05$. So, we are confidence that the fire severity is positively correlated with deforestation score which means the more severe fire in forest are related to more deforestation. Fire Severity is the second most important factor influencing deforestation
3. The regression coefficient of beetles is positive with p-value equals $0.02 < 0.05$. So, we are confidence that the number of beetles is positively correlated with deforestation score which means the more beetles in forest are related to more deforestation. Beetles number is the third most important factor influencing deforestation
4. The regression coefficient of population is positive with p-value equals $0.034 < 0.05$. So, we are confidence that the population is positively correlated with deforestation score which means the more population in forest service zone are related to more deforestation. Population is the fourth most important factor influencing deforestation (may due to agriculture and log demand)
5. The 95% confidence interval of regression coefficient of temperature is -0.016 to 0.037 which contains 0. So, we are 95% confident that temperature in forest service zone is not related to deforestation
6. The 95% confidence interval of regression coefficient of number of freeways crossing forest service zone is -0.075 to 0.007 which contains 0. So, we are 95% confident that number of freeways crossing forest service zone is not related to deforestation. It is wierd that number of freeway cross forest service zone does not impact deforestation. This is anti-intuitive
7. The regression coefficient of precipitation is negative with p-value approximately equals $0.036 < 0.05$. So, the burned area even negatively associated with deforestation with statistical significance. It is couter-intuitive.(The forest may regrow rapidly with grass)

10 Discussion

1. Our first two literatures focus on identifying the deforestation area from a satelliet image. Instead of using a K-means or deep learning model to identify the deforestation area, we used the ndvi value. Our result is not bad compared with a satelliet image. Basically, we just offer a simpler but as effective method to identify the area of deforestation.
2. The second part of the second literature and the last two literatures focus on analyzing what factors may contribute to the deforestation. In the last literature, the author points out that drought, wildfire, insects and population increase contribute to the deforestation in US. Our analysis agrees with his argument since our analysis shows that fire severity, beetles, precipitation and population are all correlated with deforestation score with statistical significance (P value less than 0.05). However, in Professor Ben DeVries's analysis, he proposed that the road construction is most heaviliy correlated with deforestation in Amazon rain forest. This does not disagree with our finding because our study area is US national forest which is different from Amazon rain forest.
3. When we are spatial joining the burned area polygon layer with forest service zone polygon layer, problems always occur. This may related to conflict of boundary point or arising of invalid geometries after spatial operation. In order to resolve the problem, we have to convert the polygon layer to a point layer (using the centroid of each polygon). By doing this, our prediction accuracies may decrease, but we saved a lot of computational time and address the error.
4. The size of the buffer zone also needs more consideration. The 20-miles buffer zone is just choosed at our discretion.
5. Capture dates from sentinel-2: We may experience different capture date for different parts of the US national forest. One may come from summer while other may come from winter, which the vegetation may not be the same. Moreover, the image may also not reflect the current states of the national forest since several wildfire had happened near some forest, which may not be captured by sentinel-2.
6. Pixel problem: We noticed that there are several different resolution for the sentinel-2 data. In our experiment, we found that for some forest, the score calculate varies a lot if we use different pixel size. We suspect that the issue come from different resolution image were taken on different seasons.
7. Our algorithm to rescale the burned area by time may not be the optimal. We need to study some enviromental science paper to create the best rescaling algorithm.

11 Conclusions and future work

We believe that we successfully address our initial research question. We use ndvi to calculate percentage of deforestation areas in each US national forest. We also spatial join different feature layers with our study areas. At last, we build a linear regression model to understand how each feature may contribute to the deforestation in each US national forest. The only thing that we wish to do but we didn't do is to understand the cause of deforestation. Our model only suggests a corrleation between each factor and deforestation score, but correlation does not imply causal relationship. In order to investigate causal relationship, more advanced regression technique is required and we also need to collect more features on each area of interest.

Our approach can also be used to analyze topics like desertification and factors that may contribute to it. In a broader sense, our approach can be used to analyze any abnormality in the sentinel-2 image and what factors may contribute to it.

We expect our result to be used by those people who are in charge of the preservation of US national forest and all the people who are concerned about the deforestation problem in the US national forest. By looking at our result, the decision-maker can better understand what factors may contribute to the deforestation and make better decisions regarding the preservation of US national forests.

Contribution

- Judy mainly works on undertanding the deforestation status.
- Xuhui mainly works on finding related factors on deforestation.
- We perform the training and prediction tasks together and help each other during our discussion.