National Parks and Biodiversity - Demographics Report

Name: Veronica Alejandro

UT EID: vaa678

Homework 5

This is our last homework. We will combine some of the concepts from our recent lab on visualizing geospatial data with merging data and create an interactive dashboard.

In [1]:
```
!pip3 install vega-datasets
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: vega-datasets in /Users/vero/Library/Python/3.9/lib/python/site-packages
(0.9.0)
Requirement already satisfied: pandas in /Users/vero/Library/Python/3.9/lib/python/site-packages (from veg
a-datasets) (1.5.3)
Requirement already satisfied: pytz>=2020.1 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om pandas->vega-datasets) (2022.7.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /Users/vero/Library/Python/3.9/lib/python/site-pa
ckages (from pandas->vega-datasets) (2.8.2)
Requirement already satisfied: numpy>=1.20.3 in /Users/vero/Library/Python/3.9/lib/python/site-packages (f
rom pandas->vega-datasets) (1.24.1)
Requirement already satisfied: six>=1.5 in /Library/Developer/CommandLineTools/Library/Frameworks/Python3.
framework/Versions/3.9/lib/python3.9/site-packages (from python-dateutil>=2.8.1->pandas->vega-datasets)
(1.15.0)

[notice] A new release of pip available: 22.3.1 -> 23.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -m pip install --upgrade pip
```

In [2]:
```
!pip3 install altair
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: altair in /Users/vero/Library/Python/3.9/lib/python/site-packages (4.2.2)
Requirement already satisfied: numpy in /Users/vero/Library/Python/3.9/lib/python/site-packages (from alta
ir) (1.24.1)
Requirement already satisfied: jinja2 in /Users/vero/Library/Python/3.9/lib/python/site-packages (from alt
air) (3.1.2)
Requirement already satisfied: entrypoints in /Users/vero/Library/Python/3.9/lib/python/site-packages (fro
m altair) (0.4)
Requirement already satisfied: toolz in /Users/vero/Library/Python/3.9/lib/python/site-packages (from alta
ir) (0.12.0)
Requirement already satisfied: jsonschema>=3.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages
(from altair) (4.17.3)
Requirement already satisfied: pandas>=0.18 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om altair) (1.5.3)
Requirement already satisfied: attrs>=17.4.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages (f
rom jsonschema>=3.0->altair) (22.2.0)
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in /Users/vero/Library/Pytho
n/3.9/lib/python/site-packages (from jsonschema>=3.0->altair) (0.19.3)
Requirement already satisfied: python-dateutil>=2.8.1 in /Users/vero/Library/Python/3.9/lib/python/site-pa
ckages (from pandas>=0.18->altair) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om pandas>=0.18->altair) (2022.7.1)
Requirement already satisfied: MarkupSafe>=2.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages
(from jinja2->altair) (2.1.2)
Requirement already satisfied: six>=1.5 in /Library/Developer/CommandLineTools/Library/Frameworks/Python3.
framework/Versions/3.9/lib/python3.9/site-packages (from python-dateutil>=2.8.1->pandas>=0.18->altair) (1.
15.0)

[notice] A new release of pip available: 22.3.1 -> 23.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -m pip install --upgrade pip
```

```
In [3]: !pip3 install geopandas
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: geopandas in /Users/vero/Library/Python/3.9/lib/python/site-packages (0.12.
2)
Requirement already satisfied: fiona>=1.8 in /Users/vero/Library/Python/3.9/lib/python/site-packages (from
geopandas) (1.9.3)
Requirement already satisfied: pyproj>=2.6.1.post1 in /Users/vero/Library/Python/3.9/lib/python/site-packa
ges (from geopandas) (3.5.0)
Requirement already satisfied: pandas>=1.0.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages (f
rom geopandas) (1.5.3)
Requirement already satisfied: shapely>=1.7 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om geopandas) (2.0.1)
Requirement already satisfied: packaging in /Users/vero/Library/Python/3.9/lib/python/site-packages (from
geopandas) (23.0)
Requirement already satisfied: cligj>=0.5 in /Users/vero/Library/Python/3.9/lib/python/site-packages (from
fiona>=1.8->geopandas) (0.7.2)
Requirement already satisfied: importlib-metadata in /Users/vero/Library/Python/3.9/lib/python/site-packag
es (from fiona>=1.8->geopandas) (6.0.0)
Requirement already satisfied: click~=8.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages (from
fiona>=1.8->geopandas) (8.1.3)
Requirement already satisfied: munch>=2.3.2 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om fiona>=1.8->geopandas) (2.5.0)
Requirement already satisfied: attrs>=19.2.0 in /Users/vero/Library/Python/3.9/lib/python/site-packages (f
rom fiona>=1.8->geopandas) (22.2.0)
Requirement already satisfied: click-plugins>=1.0 in /Users/vero/Library/Python/3.9/lib/python/site-packag
es (from fiona>=1.8->geopandas) (1.1.1)
Requirement already satisfied: certifi in /Users/vero/Library/Python/3.9/lib/python/site-packages (from fi
ona>=1.8->geopandas) (2022.12.7)
Requirement already satisfied: numpy>=1.20.3 in /Users/vero/Library/Python/3.9/lib/python/site-packages (f
rom pandas>=1.0.0->geopandas) (1.24.1)
Requirement already satisfied: python-dateutil>=2.8.1 in /Users/vero/Library/Python/3.9/lib/python/site-pa
ckages (from pandas>=1.0.0->geopandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /Users/vero/Library/Python/3.9/lib/python/site-packages (fr
om pandas>=1.0.0->geopandas) (2022.7.1)
Requirement already satisfied: six in /Library/Developer/CommandLineTools/Library/Frameworks/Python3.frame
work/Versions/3.9/lib/python3.9/site-packages (from munch>=2.3.2->fiona>=1.8->geopandas) (1.15.0)
Requirement already satisfied: zipp>=0.5 in /Users/vero/Library/Python/3.9/lib/python/site-packages (from
importlib-metadata->fiona>=1.8->geopandas) (3.11.0)

[notice] A new release of pip available: 22.3.1 -> 23.1
[notice] To update, run: /Library/Developer/CommandLineTools/usr/bin/python3 -m pip install --upgrade pip
```

In [4]:

```
# importing packages

# data wrangling
import pandas as pd
import numpy as np

# plotting
import altair as alt

# working with geographic data
import json
import geopandas as gpd
from vega_datasets import data
```

Loading in a shape file

This zip file contains a *shape* file. This is one of the types of geometry data supported by geopandas (doc)

In [5]:

```
# importing shapefile - We can use a zip file that contains a shape file by reading it from a URL
gdf = gpd.read_file('nps_boundary.zip')
```

In [6]:

```
gdf.sample(5)
```

Out[6]:

	UNIT_CODE	GIS_Notes	UNIT_NAME	DATE_EDIT	STATE	REGION	GNIS_ID	UNIT_TYPE	CREATED
401	HAFE	Lands - http://landsnet.nps.gov/tractsnet/docu...	Harpers Ferry National Historical Park	2020-09-29	WV	NC	1556735	National Historical Park	La
371	WHSA	Lands - http://landsnet.nps.gov/tractsnet/docu...	White Sands National Park	2020-01-22	NM	IM	914261	National Park	La
203	TUIN	Lands - http://landsnet.nps.gov/tractsnet/docu...	Tuskegee Institute National Historic Site	2016-01-14	AL	SE	158362	National Historic Site	La
21	BOHA	Lands - http://landsnet.nps.gov/tractsnet/docu...	Boston Harbor Islands National Recreation Area	2006-08-14	MA	NE	606990	National Recreation Area	La
384	TUSK	Lands - http://landsnet.nps.gov/tractsnet/docu...	Tule Springs Fossil Beds National Monument	2020-08-31	NV	PW	2767392	National Monument	La

Q1 - Overlay the National Parks and Monuments on a county map of the USA

Using the geopandas data frame `gdf`, filter out just the national parks and monuments and overlay them on a county map of the USA. Use the Vega data set to map the counties, using this page as a reference. Create a tool tip that shows the park name (`UNIT_NAME`) and region (`REGION`) of the park, title your chart, and encode the color with the Region of the park using the `dark2` colorscheme. See this page for a reference on changing the color scheme.

This would be an appropriate visualization if our user wanted to understand the landsize of the parks and how the regions were labeled.

In [7]:
```
gdf = gdf[(gdf['UNIT_TYPE'] == 'National Monument') | (gdf['UNIT_TYPE'] == 'National Park')]
```

```
gdf.head()
```

Out[7]:

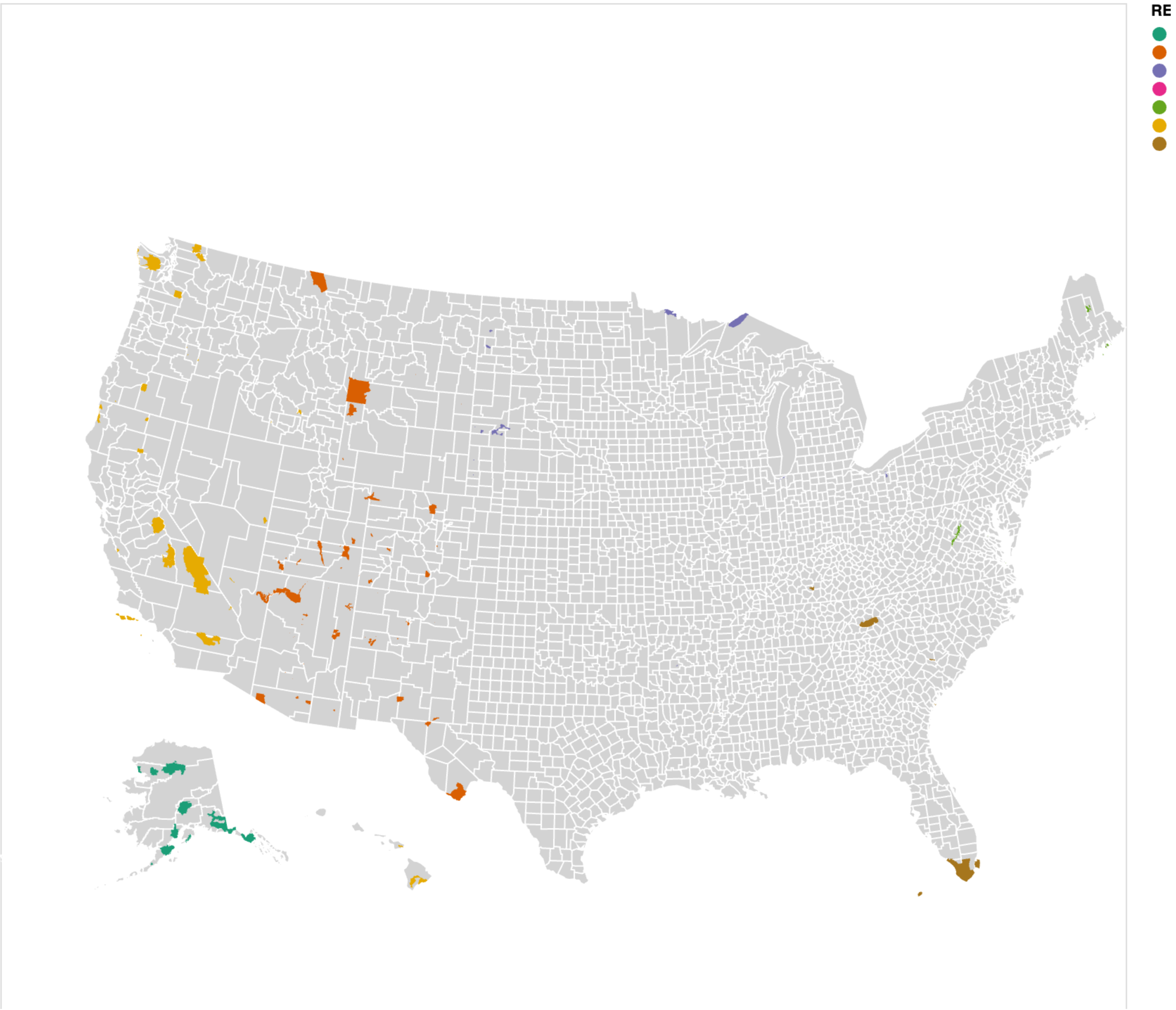
	UNIT_CODE	GIS_Notes	UNIT_NAME	DATE_EDIT	STATE	REGION	GNIS_ID	UNIT_TYPE	CREATED_B
2	LIBI	Lands - http://landsnet.nps.gov/tractsnet/docu...	Little Bighorn Battlefield National Monument	2005-02-23	MT	IM	806297	National Monument	Lanc
3	CAVO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Capulin Volcano National Monument	2004-07-22	NM	IM	936867	National Monument	Lanc
5	GICL	Lands - http://landsnet.nps.gov/tractsnet/docu...	Gila Cliff Dwellings National Monument	2004-05-10	NM	IM	928945	National Monument	Lanc
6	TUZI	Lands - http://landsnet.nps.gov/tractsnet/docu...	Tuzigoot National Monument	2003-11-21	AZ	IM	35601	National Monument	Lanc
8	CASA	Lands - http://landsnet.nps.gov/tractsnet/docu...	Castillo de San Marcos National Monument	2005-02-28	FL	SE	307911	National Monument	Lanc

In [8]:

```
 # your code here
from vega_datasets import data
counties = alt.topo_feature(data.us_10m.url, 'counties')

countiesgraph = alt.Chart(counties).mark_geoshape(
    fill='lightgray',
    stroke='white'
).project('albersUsa').properties(
    width=800,
    height=800
)
```

```
basemap = alt.Chart(gdf).mark_geoshape(
).encode(
    tooltip = ['UNIT_NAME', 'REGION'],
    color=alt.Color('REGION', scale = alt.Scale(scheme='dark2'))
).properties(width=800, height=800)

q1 = countiesgraph + basemap
q1
```

```
Out[8]:
```



Loading in a csv a species data

In [9]:

```
df_species = pd.read_csv('numspecies.csv')
df_species = df_species.fillna( value = 0 ) # Fill the missing species counts with zeros for visualization
df_species.sample(5)
```

Out[9]:

	UNIT_CODE	All Types	Amphibian	Bird	Fish	Mammal	Reptile	Vascular Plant	Fungi	Insect	Invertebrate	Spider/Scorpion	Nonva
8	CARE	1566	6.0	248	15.0	73	21.0	1203	0.0	0.0	0.0	0.0	
42	PEFO	853	8.0	244	0.0	62	21.0	516	1.0	0.0	0.0	0.0	
11	CONG	2321	43.0	200	65.0	39	50.0	884	279.0	617.0	15.0	21.0	
9	CAVE	1536	15.0	367	5.0	92	62.0	995	0.0	0.0	0.0	0.0	
44	REDW	6310	33.0	501	247.0	154	39.0	2257	1363.0	744.0	334.0	7.0	

Q2 - Change data to Long format

Remember long form vs wide form formats? It turns out our species data is wide form. Convert it to long form and save the new dataframe to `df_species_long`. Name your new columns `Species Type` and `Species Count` so they look pretty when we plot them later.

In [10]:

```
 # your code here
df_species_long = df_species.melt('UNIT_CODE', var_name = 'Species Type', value_name = 'Species Count')
```

In [11]:

```
# sanity check
df_species_long.sample(5)
```

Out[11]:

	UNIT_CODE	Species Type	Species Count
453	BLCA	Insect	0.0
662	SAGU	Nonvascular Plant	0.0
755	HALE	Crab/Lobster/Shrimp	18.0
554	VOYA	Invertebrate	7.0
540	LAVO	Invertebrate	18.0

Q3 - Get lat/long and area from park shape

We have the shape file for the national parks, which contains the polygon for the park boundary, but we decide for the visualization we want to encode the location with a circle whose size is constant. We also are interested in comparing the area of the park with the species diversity to discover trends. To accomplish these tasks we need to calculate the centroid of the park to get a single lat/long and calculate the area of the polygon in acres (a convenient unit for the USA).

To do math in geopandas you have to translate between "EPSG:4269" and "EPSG:5070" (USA Albers projection)

Add the `Acres` column to the `gdf` geopandas dataframe, and replace the `geometry` column with the centroid. Use the intro to geopandas documentation to help with the syntax.

Check that everything looks OK by plotting the lat/long of the parks on top of the county map as before. Include a tooltip that shows the region, name of park, and Acres. Use the `dark2` color scheme.

I'm helping you out with the coordinate reference systems here; note that there are many many map reference coordinates and we can see the reference for a given geopandas dataframe with the `crs` property.

In [12]:

```
# look at the coordinate reference of the data - this needs to be translated to do math, but then put back
gdf.crs
```

Out[12]:

```
<Geographic 2D CRS: EPSG:4269>
Name: NAD83
Axis Info [ellipsoidal]:
- Lat[north]: Geodetic latitude (degree)
- Lon[east]: Geodetic longitude (degree)
Area of Use:
- name: North America - onshore and offshore: Canada - Alberta; British Columbia; Manitoba; New Brunswick;
Newfoundland and Labrador; Northwest Territories; Nova Scotia; Nunavut; Ontario; Prince Edward Island; Que
bec; Saskatchewan; Yukon. Puerto Rico. United States (USA) - Alabama; Alaska; Arizona; Arkansas; Californi
a; Colorado; Connecticut; Delaware; Florida; Georgia; Hawaii; Idaho; Illinois; Indiana; Iowa; Kansas; Kent
ucky; Louisiana; Maine; Maryland; Massachusetts; Michigan; Minnesota; Mississippi; Missouri; Montana; Nebr
aska; Nevada; New Hampshire; New Jersey; New Mexico; New York; North Carolina; North Dakota; Ohio; Oklahom
a; Oregon; Pennsylvania; Rhode Island; South Carolina; South Dakota; Tennessee; Texas; Utah; Vermont; Virg
inia; Washington; West Virginia; Wisconsin; Wyoming. US Virgin Islands. British Virgin Islands.
- bounds: (167.65, 14.92, -40.73, 86.45)
Datum: North American Datum 1983
- Ellipsoid: GRS 1980
- Prime Meridian: Greenwich
```

In [13]:

```
gdf = gdf.to_crs( 'EPSG:5070')
```

In [14]:

```
gdf.crs
```

Out[14]:

```
<Projected CRS: EPSG:5070>
Name: NAD83 / Conus Albers
Axis Info [cartesian]:
- X[east]: Easting (metre)
- Y[north]: Northing (metre)
Area of Use:
- name: United States (USA) - CONUS onshore - Alabama; Arizona; Arkansas; California; Colorado; Connecticu
t; Delaware; Florida; Georgia; Idaho; Illinois; Indiana; Iowa; Kansas; Kentucky; Louisiana; Maine; Marylan
d; Massachusetts; Michigan; Minnesota; Mississippi; Missouri; Montana; Nebraska; Nevada; New Hampshire; Ne
w Jersey; New Mexico; New York; North Carolina; North Dakota; Ohio; Oklahoma; Oregon; Pennsylvania; Rhode
Island; South Carolina; South Dakota; Tennessee; Texas; Utah; Vermont; Virginia; Washington; West Virgini
a; Wisconsin; Wyoming.
- bounds: (-124.79, 24.41, -66.91, 49.38)
Coordinate Operation:
- name: Conus Albers
- method: Albers Equal Area
Datum: North American Datum 1983
- Ellipsoid: GRS 1980
- Prime Meridian: Greenwich
```

In [15]:

```python
# Find the area in acres
gdf['Acres'] = gdf.area / 4047
```

In [16]:

```python
# sanity check
# The value from Rocky Mountain National Park webpage is 265,807 acres and our estimate is off probably due
# projection we used was not locally optimal, or maybe the shape file wasn't exact. If we were a GIS class
# this would probably bother us, but for now we'll call this close enough (your number should be about 0.5
gdf[gdf['UNIT_CODE'] == 'ROMO']['Acres']
```

Out[16]:

```
415    267032.175621
Name: Acres, dtype: float64
```

In [17]:

```python
# your code here to convert the polygons data to point data
gdf['geometry'] = gdf.centroid
```

In [18]:

```python
gdf = gdf.to_crs( 'EPSG:4269') # This takes us back to lat/long space
gdf.sample(5)
```

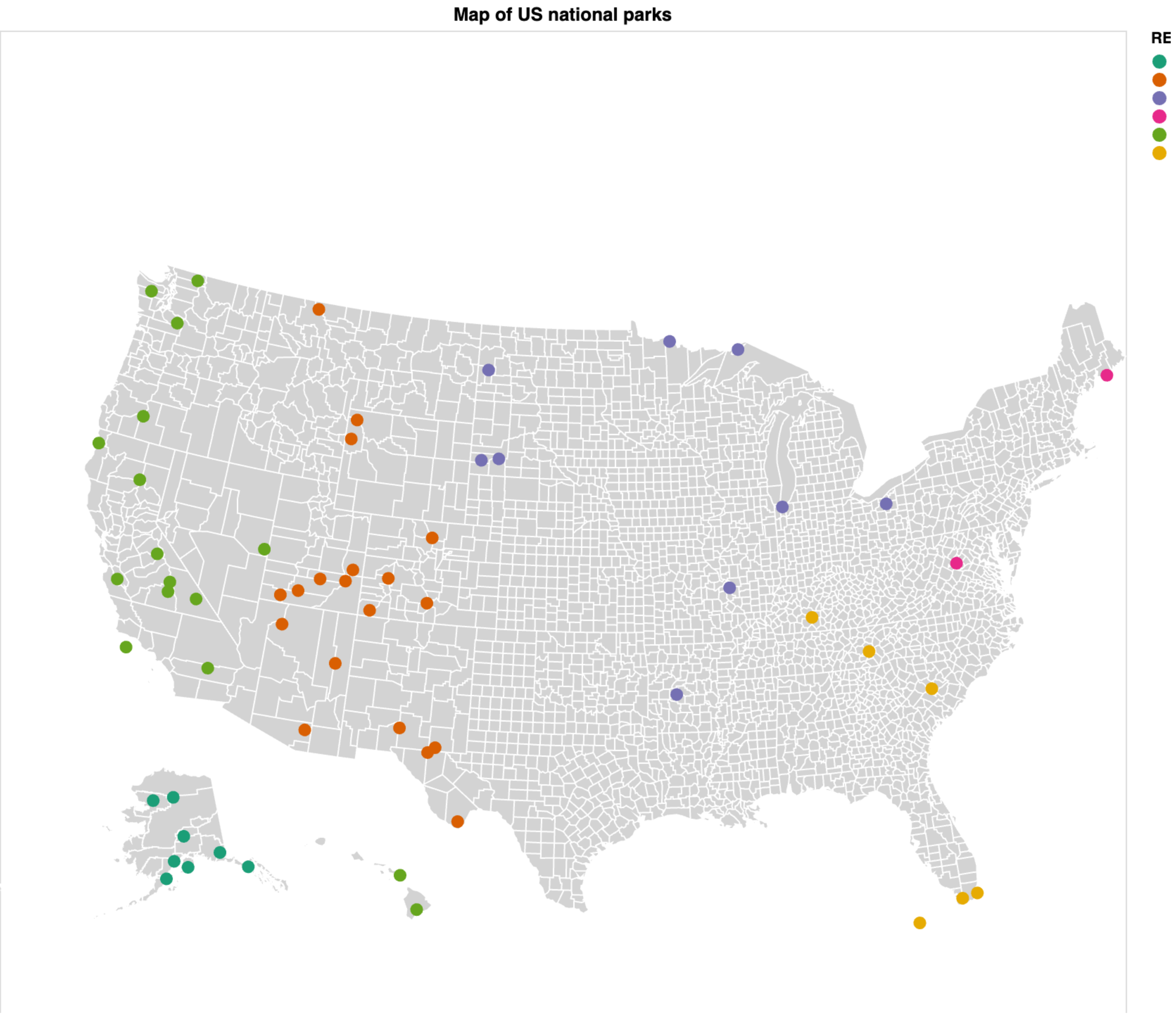
Out[18]:

	UNIT_CODE	GIS_Notes	UNIT_NAME	DATE_EDIT	STATE	REGION	GNIS_ID	UNIT_TYPE	CREATED_
160	CAGR	Lands - http://landsnet.nps.gov/tractsnet/docu...	Casa Grande Ruins National Monument	2020-04-29	AZ	IM	44811	National Monument	La
171	ORCA	Lands - http://landsnet.nps.gov/tractsnet/docu...	Oregon Caves National Monument and Preserve	2015-02-26	OR	PW	1147260	National Monument	La
51	SEQU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Sequoia National Park	2008-11-26	CA	PW	266000	National Park	La
27	ELMO	Lands - http://landsnet.nps.gov/tractsnet/docu...	El Morro National Monument	2006-12-11	NM	IM	906062	National Monument	La
101	JEFF	Lands - http://landsnet.nps.gov/tractsnet/docu...	Gateway Arch National Park	2010-10-28	MO	MW	765817	National Park	La

In [19]:

```
basemap = alt.Chart(gdf[gdf['UNIT_TYPE'] == 'National Park']).mark_geoshape(
).encode(
    tooltip = ['UNIT_NAME', 'REGION', 'Acres'],
    color=alt.Color('REGION', scale = alt.Scale(scheme='dark2'))
).properties(width=800, height=800)

q3 = countiesgraph + basemap
q3 = q3.properties(
    title = 'Map of US national parks'
)
q3
```

Out[19]:



Q4 - Merge Geo Data and and Species Data

We've been using the `UNIT_NAME` for the park name in the plots, and we want to use that same name, and the acerage when plotting the species data. Add these two columns to our `df_species_long` dataframe using a inner join, renaming the `UNIT_NAME` to `Park Name`, and `REGION` to `Region` for pretty visualization. The key for the merge is `UNIT_CODE`. Save this new dataframe as `df_species_final`

In [20]:

```
# your code here
df_species_final = gdf.merge(df_species_long, how = 'inner', on = 'UNIT_CODE')
df_species_final = df_species_final.rename(columns = {'UNIT_NAME': 'Park Name', 'REGION': 'Region'})
df_species_final
```

Out[20]:

	UNIT_CODE	GIS_Notes	Park Name	DATE_EDIT	STATE	Region	GNIS_ID	UNIT_TYPE	CREATED_BY
0	DRTO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Dry Tortugas National Park	2007-11-07	FL	SE	307632	National Park	Lands
1	DRTO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Dry Tortugas National Park	2007-11-07	FL	SE	307632	National Park	Lands
2	DRTO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Dry Tortugas National Park	2007-11-07	FL	SE	307632	National Park	Lands
3	DRTO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Dry Tortugas National Park	2007-11-07	FL	SE	307632	National Park	Lands
4	DRTO	Lands - http://landsnet.nps.gov/tractsnet/docu...	Dry Tortugas National Park	2007-11-07	FL	SE	307632	National Park	Lands
...	...	...	...	...	...	...	...	...	...
820	SAGU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Saguaro National Park	2021-03-29	AZ	IM	10683	National Park	Lands
821	SAGU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Saguaro National Park	2021-03-29	AZ	IM	10683	National Park	Lands
822	SAGU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Saguaro National Park	2021-03-29	AZ	IM	10683	National Park	Lands
823	SAGU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Saguaro National Park	2021-03-29	AZ	IM	10683	National Park	Lands
824	SAGU	Lands - http://landsnet.nps.gov/tractsnet/docu...	Saguaro National Park	2021-03-29	AZ	IM	10683	National Park	Lands

825 rows × 19 columns

Q5 - Compare the Acres to Species Count

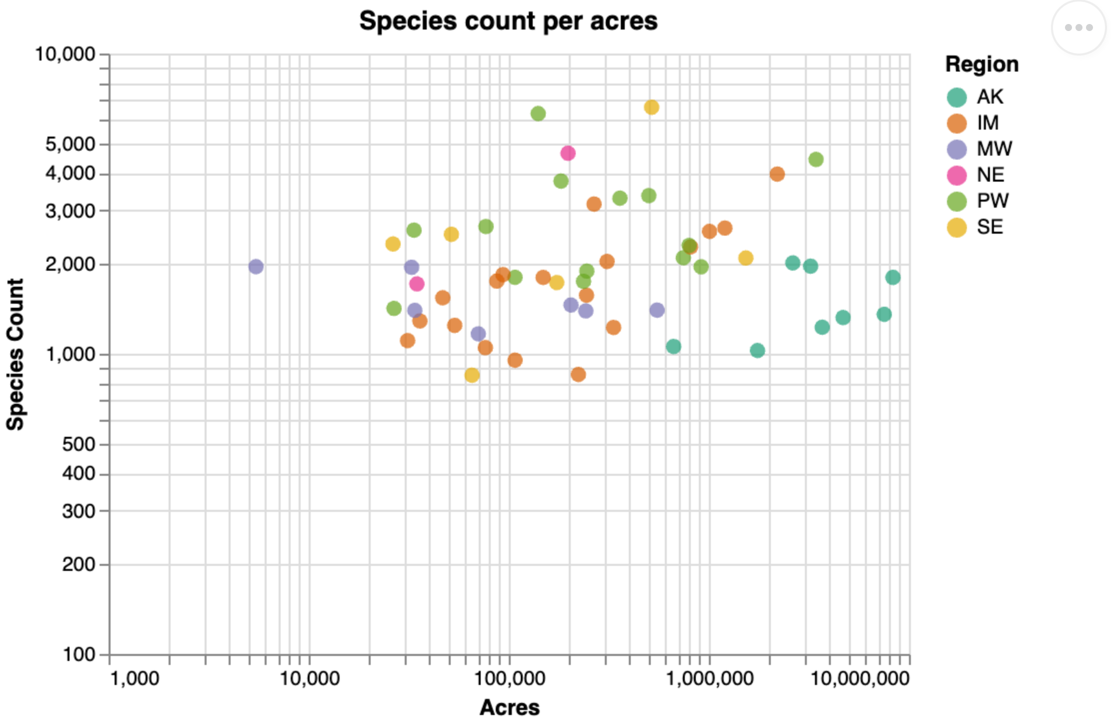
On a log/log scale compare the total species count (only the rows where "Species Type" is "All Types") with the park size. Encode the region using the same color scale as above. Add a tool tip to see the park name, Species Count, and Acres.

Add a `selection_multi` that will highlight the selected data by turning unselected data light gray.

In [21]:

```
# your code here
df_species_all = df_species_final[df_species_final['Species Type'] == 'All Types']

multi = alt.selection_interval()

q5 = alt.Chart(df_species_all).mark_circle(size = 60).encode(
    x = alt.X('Acres:Q', scale = alt.Scale(type = 'log')),
    y = alt.Y('Species Count:Q', scale = alt.Scale(type = 'log')),
    color = alt.condition(multi, alt.Color('Region:N', scale = alt.Scale(scheme = 'dark2')), alt.value('lig
    tooltip=['Park Name', 'Species Count', 'Acres']
).properties(
    title = 'Species count per acres'
)

q5.add_selection(
    multi
)
```

Out[21]:



Q6 - Create a rug plot of Species Diversity

Look at how the different Species categores (all Species types that are NOT 'All Types' are distributed across all the parks.

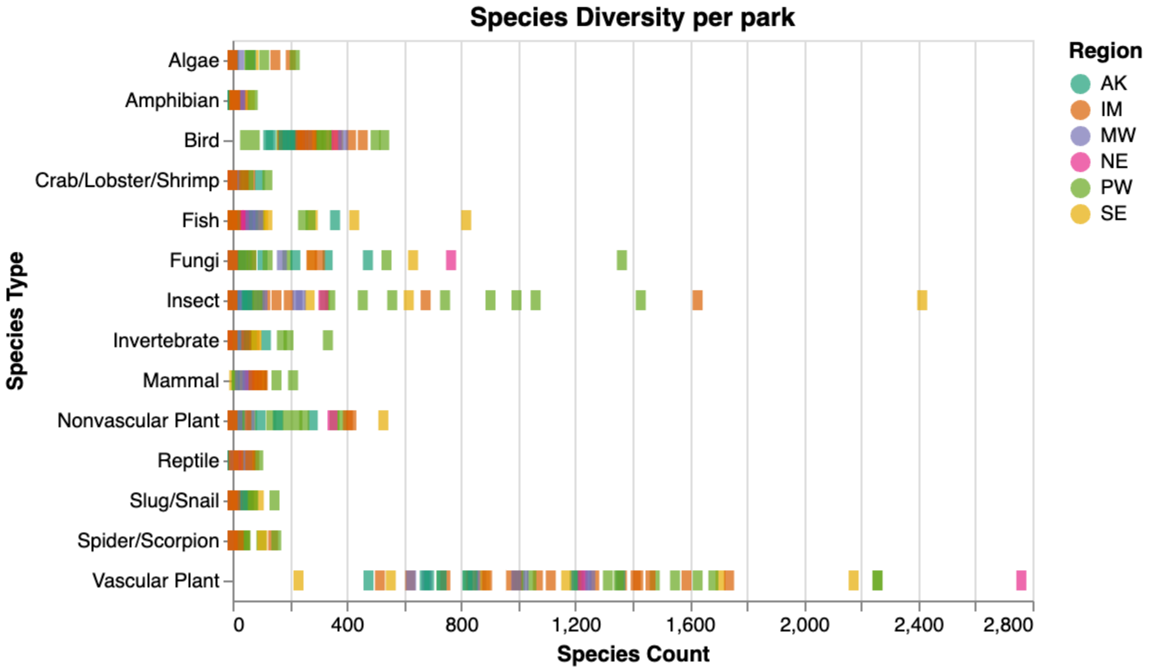
Use the same color encoding for the Region as the prior plots.

Add a `selection_multi` that will highlight the selected data for a park by turning unselected data light gray. Also, make the tick a little bigger when it is selected to help find the park in the other categories.

In [22]:

```
# your code here
species_diff = df_species_final[df_species_final['Species Type'] != 'All Types']
unique_species = list(species_diff['Species Type'].unique())

selection = alt.selection_multi(on = 'mouseover')

q6 = alt.Chart(species_diff).mark_tick(size = 15, thickness = 5).encode(
    x = 'Species Count',
    y = 'Species Type',
    color = alt.condition(selection, alt.Color('Region:N', scale = alt.Scale(scheme = 'dark2')), alt.value
    tooltip = ['Park Name'],
    size = alt.condition(selection, alt.value(10), alt.value(2))
).properties(
    title = 'Species Diversity per park'
)

q6.add_selection(
    selection
)
```

Out[22]:



Q7 - Create a sorted bar chart of Species Count with a Region Drop down selector

Create a bar chart with horizontal bars that shows the total species count per park with the color channel encoding the Region.

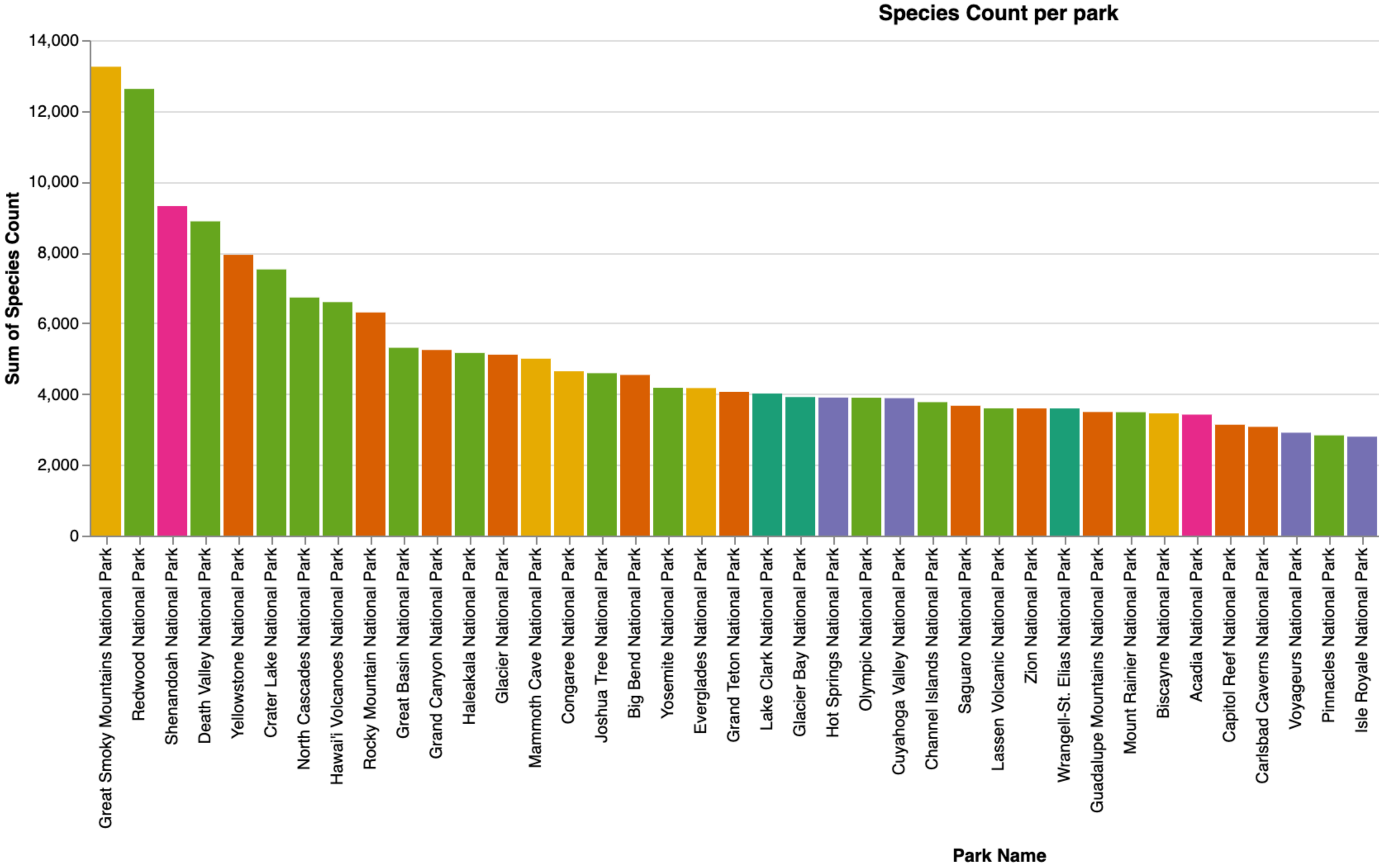
Add a drop down selector so that you can filter the bar chart by just one region. See this documentation for an example of adding a drop down menu and using it to filter your selections.

In addition to the drop down, include a `selection_multi` that will highlight the selected data by turning unselected data light gray.

In [23]:

```
# your code here

regions = list(df_species_final['Region'].unique())
dropdown = alt.binding_select(options = regions, name = "Region")
selection = alt.selection_single(fields=['Region'], bind=dropdown)


q7 = alt.Chart(df_species_final).mark_bar().encode(
    x = alt.X('Park Name', sort = '-y'),
    y = alt.Y('sum(Species Count)'),
    color = alt.condition(
    selection,
    alt.Color('Region:O', scale = alt.Scale(scheme = 'dark2')),
    alt.value('lightgray'))
).properties(
    title = 'Species Count per park'
)

q7.add_selection(
    selection
)
```

Out[23]:



Region SE

Q8 - Bring all the charts together for a Dashboard

Bring all 4 charts and the region filter together. Any click on one chart should highlight that data in the other charts. The colors for the regions should not change depending on the region selection and be consistent across all graphs. When plotting your geopandas data frame, filter out only the rows that have species data before making the map. I used the documentation on customizing titles to change the font size and name my dashboard (all of the titles in the component graphs I changed to be `subtitles`)

Change your color scheme to `tableau10`.

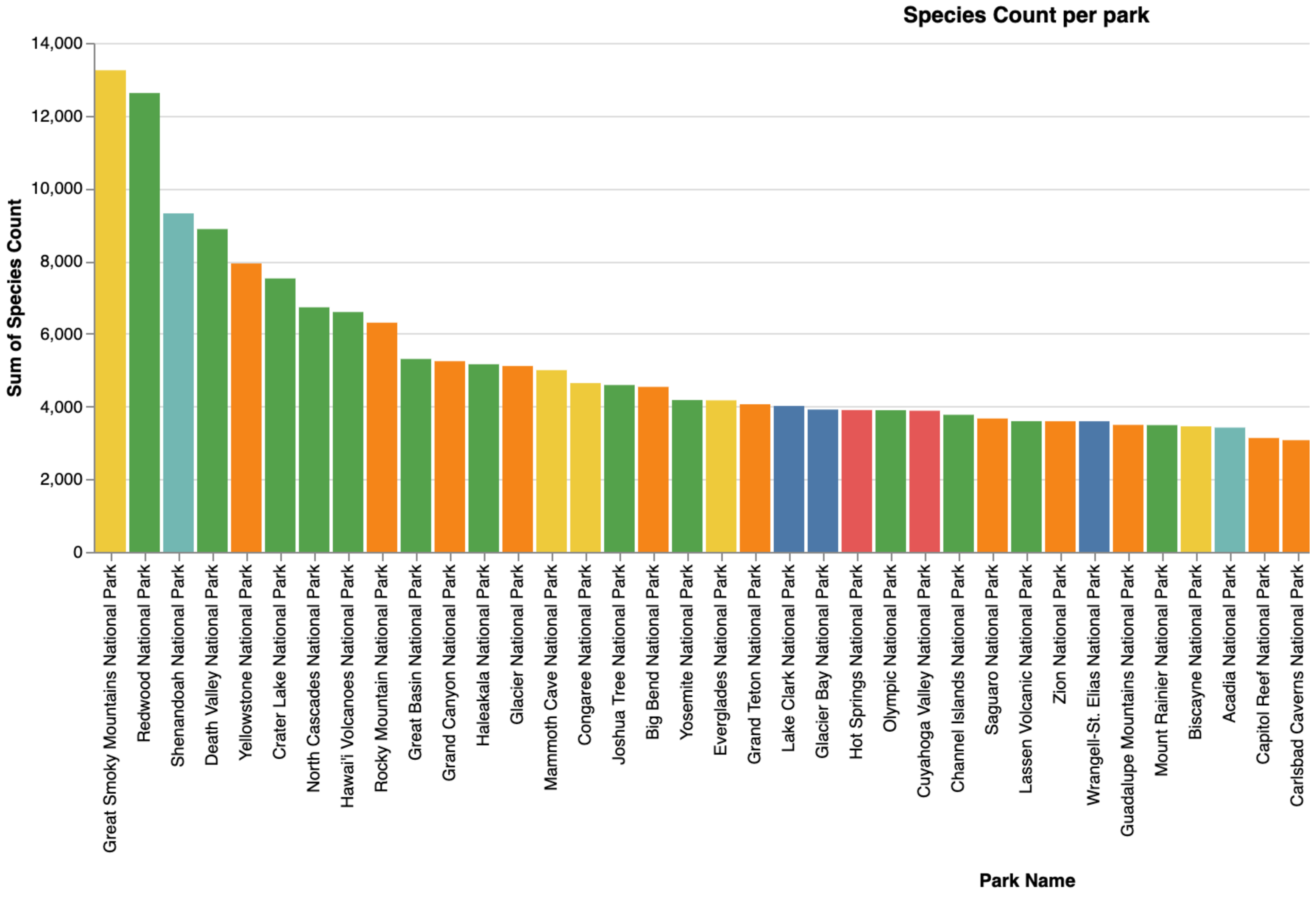
Hint: There will be only one `selection_multi` that links all the selection highlighting in the charts together.

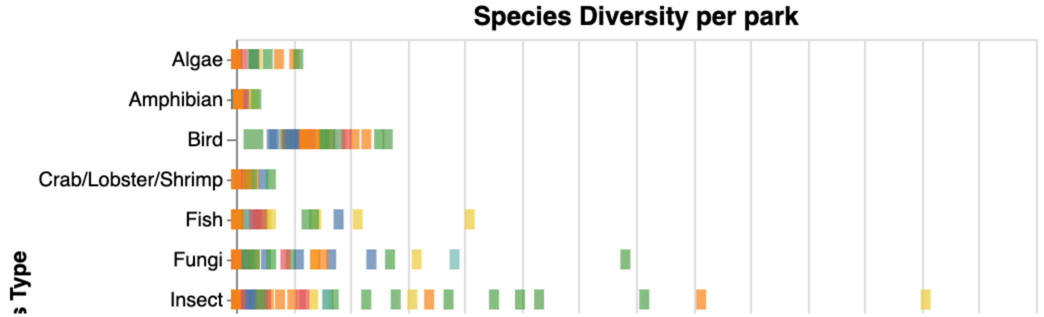
In [24]:

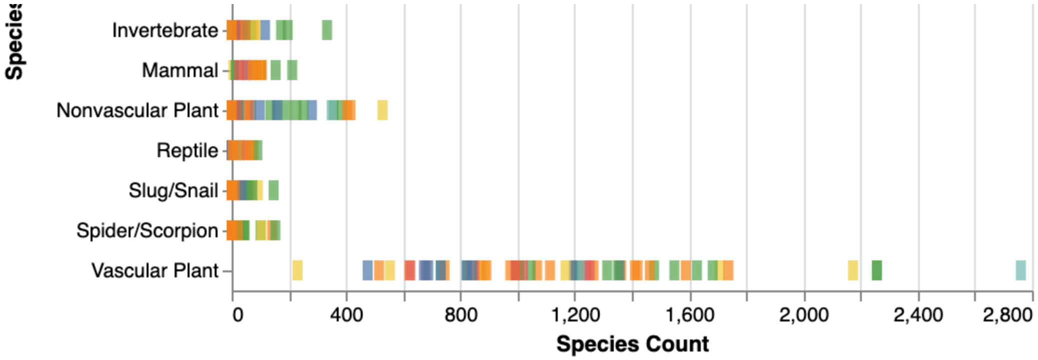
```
# regions = list(df_species_final['Region'].unique())
# labels = [region + ' ' for region in regions]
# dropdown = alt.binding_select(options = regions + [None], labels = labels + ['All'],name = "Region")
# selector = alt.selection_single(fields=['Region'], bind=dropdown)
selector = alt.selection_multi(fields = ['Region'])
color = alt.condition(selector, alt.Color('Region:O', scale = alt.Scale(scheme = 'tableau10')),
                      alt.value('lightgray'))

countiesgraph = alt.Chart(counties).mark_geoshape(
    fill='lightgray',
    stroke='white'
).project('albersUsa').properties(
    width=800,
    height=800
)

basemap = alt.Chart(df_species_final).mark_geoshape(
).encode(
    tooltip = ['Park Name', 'Region', 'Acres'],
    color=color
).properties(width=800, height=800)

q3 = countiesgraph + basemap
q3 = q3.add_selection(
selector).properties(
    title = 'Map of US national parks'
)


q5 = alt.Chart(df_species_all).mark_circle(size = 60).encode(
    x = alt.X('Acres:Q', scale = alt.Scale(type = 'log')),
    y = alt.Y('Species Count:Q', scale = alt.Scale(type = 'log')),
    color = color,
    tooltip=['Park Name', 'Species Count', 'Acres']
).add_selection(
selector).properties(
    title = 'Species count per acres'
)


q6 = alt.Chart(species_diff).mark_tick(size = 15, thickness = 5).encode(
    x = 'Species Count',
    y = 'Species Type',
    color = color,
    tooltip = ['Park Name'],
    size = alt.condition(selector, alt.value(10), alt.value(2))
).add_selection(
selector
).properties(
    title = 'Species Diversity per park'
)

q7 = alt.Chart(df_species_final).mark_bar().encode(
    x = alt.X('Park Name', sort = '-y'),
    y = alt.Y('sum(Species Count)'),
    color = color
).add_selection(
selector).properties(
    title = 'Species Count per park'
)

q8 = q7 & q6 & q5 & q3
q8 = q8.properties(
    title = 'National Parks and Biodiversity Demographics Dashboard'
)
q8
```

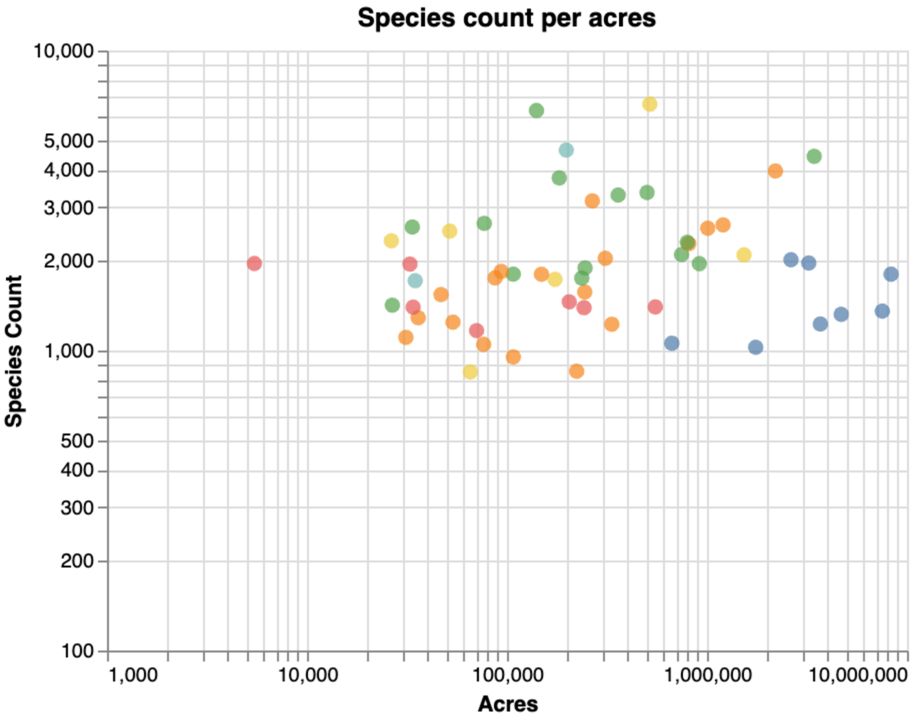
Out[24]:

National Parks and Biodiversity Demographics Dashboard



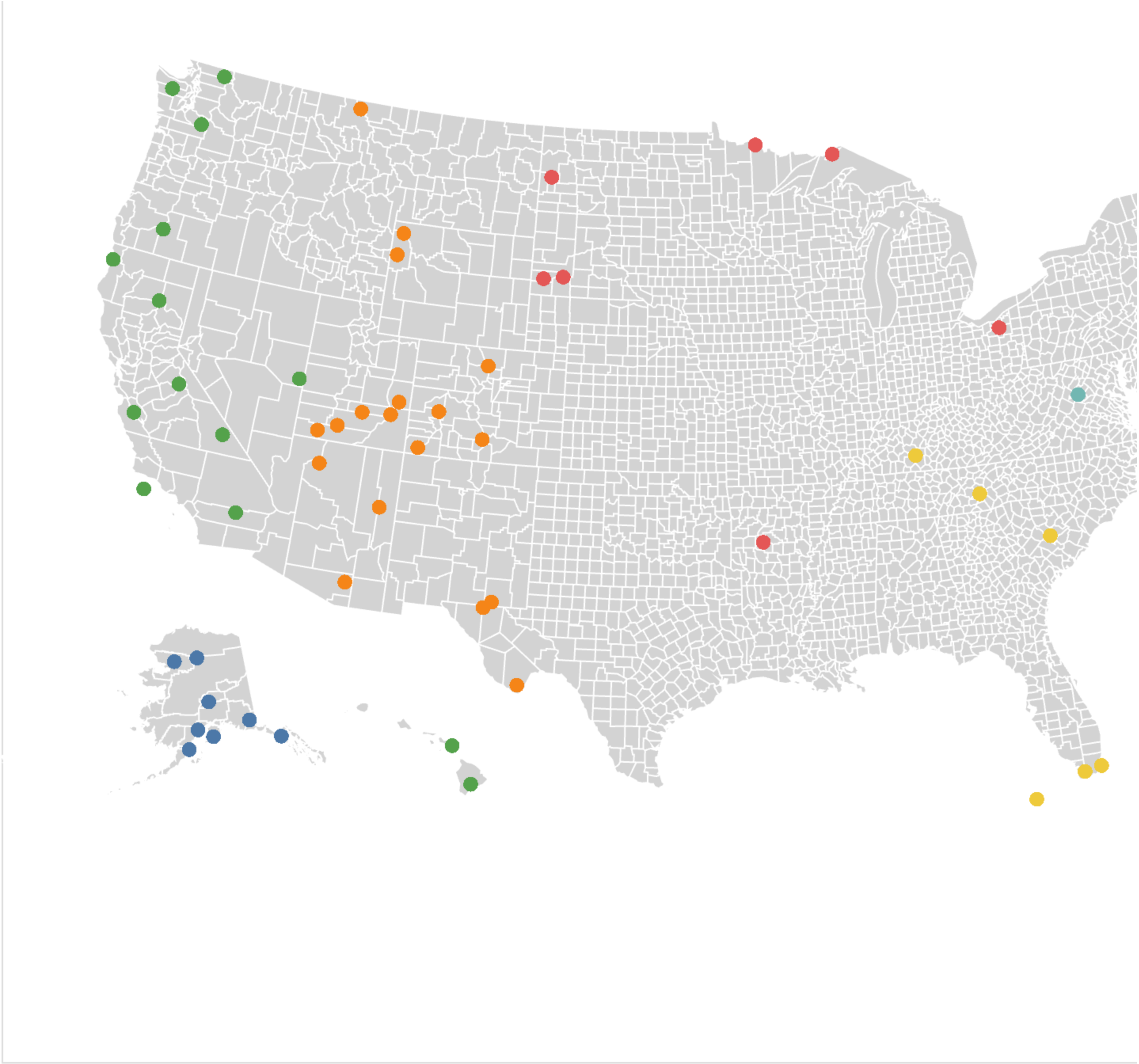






Species count per acres

Map of US national parks



```
In [ ]:
```

```
In [ ]:
```