Comparison of Interpolation Methods for Predicting Temperature (Lab 3 - Part 2)

GIS 5571: ArcGIS I
University of Minnesota

Luke Zaruba
November 30, 2022

In [1]:
```
import arcpy
import requests
import pandas as pd
```

Cleaning Data

In [2]:
```
url = "https://ndawn.ndsu.nodak.edu/table.csv?station=78&station=111&station=98&station=174&station=142&station=138&station=161&station=9&station=10&station=118&station
```

In [3]:
```
# Get Data as CSV
response = requests.get(url)

# Write Data into CSV File
file_name = 'raw_ndawn_temp.csv'
csv = open(file_name,'w')
csv.write(response.text)
csv.close()

# Take CSV and read into DF
raw_df = pd.read_csv(file_name, header=3, skiprows=[4])
raw_df.rename(columns={'Unnamed: 0':'Station Name', 'deg':'Lat', 'deg.1':'Lon', 'Degrees F':'Max', 'Degrees F.1':'Min', 'Degrees F.2':'Avg'}, inplace=True)
raw_df.head()
```

Out[3]:

	Station Name	Lat	Lon	ft	Unnamed: 4	Unnamed: 5	Unnamed: 6	Max	Unnamed: 8	Min	Unnamed: 10	Avg	Unnamed: 12
0	Ada	47.32119	-96.51406	910	2022	10	28	62.636	NaN	39.546	NaN	51.091	NaN
1	Ada	47.32119	-96.51406	910	2022	10	29	60.584	NaN	30.619	NaN	45.602	NaN
2	Ada	47.32119	-96.51406	910	2022	10	30	58.784	NaN	22.091	NaN	40.438	NaN
3	Ada	47.32119	-96.51406	910	2022	10	31	64.022	NaN	35.074	NaN	49.548	NaN
4	Ada	47.32119	-96.51406	910	2022	11	1	70.610	NaN	27.538	NaN	49.074	NaN

In [4]:
```
# Create Copy of Data with Relevant Columns
columns = ['Station Name', 'Lat', 'Lon', 'Max', 'Min', 'Avg']

cleaned_df = raw_df[columns].copy()

# Aggregate Data
agg_functions = {'Lat':'first', 'Lon':'first', 'Max':'mean', 'Min':'mean', 'Avg':'mean'}
agg_df = cleaned_df.groupby(cleaned_df['Station Name']).aggregate(agg_functions)

agg_df.head()
```

Out[4]:

	Lat	Lon	Max	Min	Avg
Station Name					
Ada	47.321190	-96.514060	38.195467	21.400933	29.798400
Adams	48.499880	-98.075880	34.255733	16.855167	25.555700
Alamo	48.546520	-103.471860	32.763867	12.436800	22.600533
Alvarado	48.245942	-97.021532	36.927167	19.318267	28.123000
Amidon	46.488440	-103.316290	37.425933	16.728133	27.077267

In [5]:
```
# Export Aggregated DF to CSV
agg_df.to_csv('aggregated_ndawn_temps.csv')
```

Converting Data To Feature Class

In [6]:
```
csv_path = r"C:\gitFiles\GIS5571\Lab3\Part 2\Lab3_2_APRX\aggregated_ndawn_temps.csv"
temperature_features = arcpy.management.XYTableToPoint(csv_path, 'station_temperatures', 'Lon', 'Lat')
```

Interpolation

In [7]:
```
# Function to Iterate through Various Z Variables and Create Interpolated Surfaces using Many Methods
def runInterpolation(fc, zList):
    for z in zList:
        # IDW
        outIDW = arcpy.sa.Idw(fc, z)
        outIDW.save(f'idw_{z}')

        # Kriging
        outKriging = arcpy.sa.Kriging(fc, z, arcpy.sa.KrigingModelOrdinary())
        outKriging.save(f'kriging_{z}')

        # Natural Neighbor
        outNatNeighbor = arcpy.sa.NaturalNeighbor(fc, z)
        outNatNeighbor.save(f'natNeighbor_{z}')
```

In [8]:
```
runInterpolation(temperature_features, ['Max', 'Min', 'Avg'])
```