Clarification questions

Please ask any question related to this assignment in the class or in the Canvas forum. To be fair to everyone, it is best to ask assignment question openly rather than in email. Do not share your code with anyone. Do not post your code online. Do not ask for help in any public discussion forum or online website. You can search over the internet and read books. Let me know if you find any error in the assignment question as soon as possible. DO NOT POST/SHARE OR DESCRIBE YOUR OUTPUT TO ANY OTHER STUDENT. You can always ask questions about the methods, definitions or approaches to solve a question. DO NOT ASK ME WHETHER MY OUTPUT LOOKS CORRECT.

Assignment 3

Note: For q1-q3, no external library except for d3 is allowed. **What to submit?**

Q1: q1.html (d3 code) and q1.jpg (screenshot of the visuzliation)

Q2: q2.html (d3 code), q2.jpg (screenshot of the visuzliation)

Q3: q3.html (d3 code), q3.jpg (screenshot of the visualization after clicking on a circle showing the change of color)

Q4: q4.py (python code), q4.jpg (screenshot of the visualization after clicking on a circle showing the change of color)

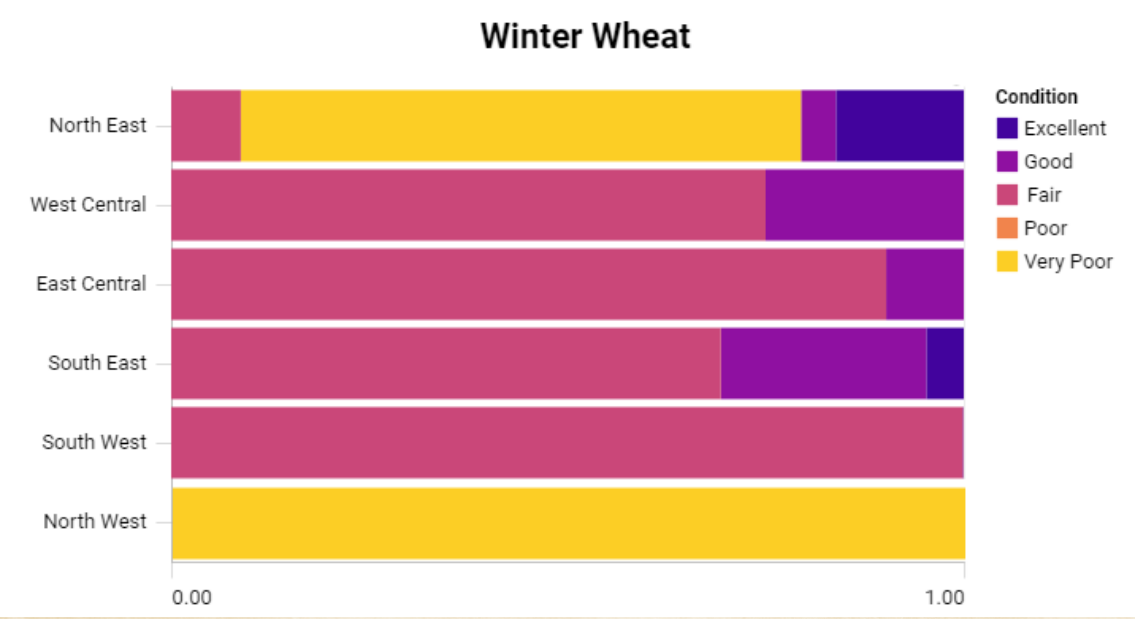
Q1. Bar Chart in D3, Mark : 30 [Use of AI-generated code allowed]

You are given a data file that contains conditions of Canola crop in July 2024 (conditions-canola-july-7-2024.csv). The data has been taken from

`https://dashboard.saskatchewan.ca/agriculture/crop-conditions/conditions-canola`

Your task is to create a bar chart in d3 as follows. Note that your chart will look different as the one you see below is not for Canola.

Your visualization should approximately match the text ordering, colors, and axis labels as shown in the example chart.



Q2. Cartography - D3.js (30 Marks) [Use of AI-generated code allowed]

This question is related to our class topic - geospatial visualization. Your task is to create a proportional symbol map (see lecture slides for definition), as follows.

1. Generate a map of Saskatoon from this link: `http://geojson.io/#map=2/20.0/0.0`
2. Generate coordinates (latitude and longitude) for different regions. You can find the regions in this link `https://dashboard.saskatchewan.ca/agriculture`. Write a d3 code to visualize the regions. At this time, your code should produce a visualization similar to this:

3. Create a proportional symbol map from the data by drawing **RED** circles on the generated coordinates. The radii of the circles must be proportional to the **sum of fair, good, and excellent values**.

Q3. Interactive Visualization (30 Marks) [Use of AI-generated code allowed]

Create a copy of the html solution that you wrote for Q2. Add mouse click event on the html code so that if the user clicks on a circle, then that circle changes its color to **BLUE**. If the user clicks on the same circle again, then the circle must change its color back to **RED**.

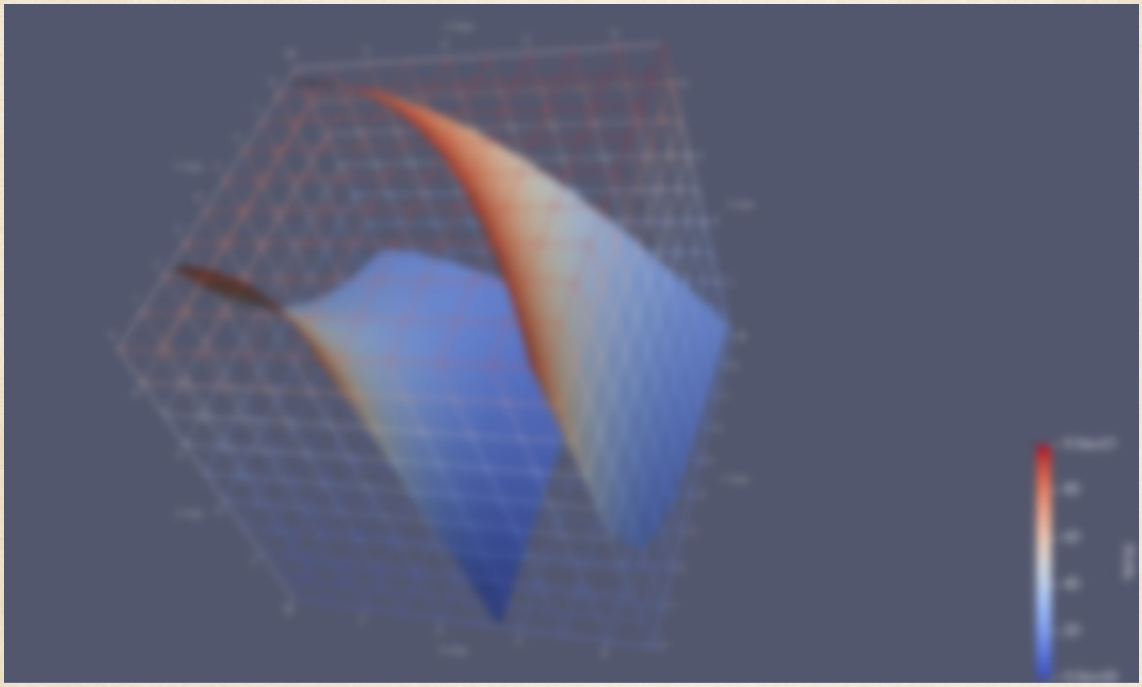
Q4. Paraview (10 Marks) [Use of AI-generated help allowed]

Send me an email if you face trouble installing Paraview or if you do not have sufficient computing resources to complete Q4.

The hypothesis of an industrial simulation is that the temp and humidity of each point are $i^2 + j + k$ and $i + ij + k^2$, respectively. We are interested to identify the space with humidity between [30, 85.5]. We also want to see the temperature of the points at these humidity thresholds.

Install vtk in python. Edit the code on page 4 such that temp and humidity of each point become $i^2 + j + k$ and $i + ij + k^2$, respectively.

Install paraview. Load the data in paraview and generate the isosurfaces for the two humidity thresholds 30 and 85.5. Color the surfaces with temperature. Your visualization should be as follows. We have deliberately blurred the image.



```
import numpy as np
import vtk

def create_some_vti(vti_file, dimensions):
    # Generate some data
    data_array = np.random.rand(*dimensions)

    # Create a structured points object
    structured_points = vtk.vtkStructuredPoints()
    structured_points.SetDimensions(dimensions[0], dimensions[1], dimensions[2])

    # Create VTK array to hold the data
    vtk_data_array1 = vtk.vtkDoubleArray()
    vtk_data_array1.SetName("temp")
    vtk_data_array1.SetNumberOfComponents(1)
    vtk_data_array1.SetNumberOfTuples(dimensions[0] * dimensions[1] * dimensions[2])

    # Populate the VTK array with the data
    for i in range(dimensions[0]):
        for j in range(dimensions[1]):
            for k in range(dimensions[2]):
                vtk_data_array1.SetTuple1(i * dimensions[1] *
                dimensions[2] + j * dimensions[2] + k, 1)

    # Assign the VTK array to the structured points object


    vtk_data_array2 = vtk.vtkDoubleArray()
    vtk_data_array2.SetNumberOfComponents(1)
    vtk_data_array2.SetName("humidity")
    vtk_data_array2.SetNumberOfTuples(dimensions[0] * dimensions[1] * dimensions[2])

    # Populate the VTK array with the data
    for i in range(dimensions[0]):
        for j in range(dimensions[1]):
            for k in range(dimensions[2]):
                vtk_data_array2.SetTuple1(i * dimensions[1] *
                dimensions[2] + j * dimensions[2] + k, 1)

    # Assign the VTK array to the structured points object
    structured_points.GetPointData().AddArray(vtk_data_array1)
    structured_points.GetPointData().AddArray(vtk_data_array2)

    # Write the structured points object to a .vti file
    writer = vtk.vtkXMLImageDataWriter()
    writer.SetFileName(vti_file)
    writer.SetInputData(structured_points)
    writer.Write()

# Example usage
vti_file = 'data.vti'
dimensions = (10, 10, 10)  # Example dimensions for the  data
create_some_vti(vti_file, dimensions)
```