STA 3180 Statistical Modelling: Spatial Statistics

STA 3180 Statistical Modelling: Spatial Statistics

Spatial statistics is a branch of statistics that deals with data that has a geographic or spatial component. It is used to analyze and interpret spatial patterns and relationships in data. Spatial statistics can be used to identify clusters, trends, and outliers in data.

Key Concepts

Spatial Autocorrelation - This is a measure of the similarity of values of a variable at different locations. It is used to measure the degree of clustering or dispersion of a variable in space.

Spatial Variograms - This is a graphical representation of the spatial autocorrelation of a variable. It shows how the values of a variable change as the distance between two points increases.

Kriging - This is a method of interpolation that uses a weighted average of nearby sample points to estimate the value of a variable at an unsampled location.

Geostatistics - This is a branch of statistics that deals with the analysis of spatial data. It is used to analyze and interpret spatial patterns and relationships in data.

Definitions

Spatial Data - Data that has a geographic or spatial component, such as latitude and longitude coordinates.

Interpolation - The process of estimating the value of a variable at an unsampled location based on the values of nearby sample points.

Clustering - A pattern in which values of a variable are grouped together in a particular area.

Trend - A pattern in which values of a variable increase or decrease in a particular direction.

Outlier - A value of a variable that lies outside the range of the other values.

Coding Examples

Example 1: Calculating Spatial Autocorrelation

```
Start of Code
// Import libraries
import numpy as np
import pandas as pd
// Load data
data = pd.read_csv('data.csv')
// Calculate spatial autocorrelation
autocorr = np.corrcoef(data['x'], data['y'])
// Print result
print(autocorr)
End of Code
```

Example 2: Plotting a Spatial Variogram

```
Start of Code
// Import libraries
import matplotlib.pyplot as plt
// Load data
data = pd.read_csv('data.csv')
// Calculate distances between points
distances = []
for i in range(len(data)):
	for j in range(i+1, len(data)):
		distances.append(data.iloc[i]['x'] - data.iloc[j]['x'])
// Calculate differences in values
differences = []
for i in range(len(data)):
	for j in range(i+1, len(data)):
		differences.append(data.iloc[i]['y'] - data.iloc[j]['y'])
// Plot variogram
plt.scatter(distances, differences)
plt.xlabel('Distance')
plt.ylabel('Difference')
plt.title('Spatial Variogram')
plt.show()
End of Code
```

Example 3: Kriging

```
Start of Code
// Import libraries
import numpy as np
import pandas as pd
// Load data
data = pd.read_csv('data.csv')
```

```
// Calculate weights
weights = []
for i in range(len(data)):
       for j in range(i+1, len(data)):
              weights.append(1/np.sqrt((data.iloc[i]['x'] - data.iloc[j]['x'])**2 +
              (data.iloc[i]['y'] - data.iloc[j]['y'])**2))
// Calculate kriged value
kriged_value = np.sum(weights * data['y']) / np.sum(weights)
// Print result
print(kriged_value)
End of Code
```