Documentation for Trendline and Uncertainty Calculator Webpage

Shafiq R

August 15, 2024

1 Introduction

This document provides a detailed description of the HTML, CSS, and JavaScript code used to create a web-based calculator for generating trendlines, calculating uncertainties, and providing centroid coordinates. The webpage also supports downloading the generated chart as JPG and PNG images.

2 Code Overview

2.1 HTML Structure

The HTML code sets up the basic structure of the webpage, including input fields, buttons, and a canvas element.

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Trendline and Uncertainty Calculator</title>
    <style>
        body { font-family: Arial, sans-serif; margin: 20px; }
        .container { max-width: 600px; margin: auto; padding: 20px; border:
            1px solid #ccc; border-radius: 8px; background-color: #f9f9f9;
            }
        input[type="text"] { width: 100%; padding: 8px; margin: 8px 0; box-
            sizing: border-box; }
        button { padding: 10px; width: 100%; background-color: #4CAF50;
            color: white; border: none; border-radius: 8px; cursor: pointer;
             margin-top: 10px; }
        button:hover { background-color: #45a049; }
        #chartContainer { margin-top: 20px; }
        canvas { max-width: 100%; }
        .results { margin-top: 20px; font-family: "Courier New", monospace;
            }
    </style>
</head>
<body>
    <div class="container">
        <h2>Trendline and Uncertainty Calculator</h2>
        <label for="xData">Enter X data points (comma-separated):</label>
        <input type="text" id="xData" placeholder="e.g., 1, 2, 3, 4, 5">
        <label for="yData">Enter Y data points (comma-separated):</label>
        <input type="text" id="yData" placeholder="e.g., 2, 4, 6, 8, 10">
        <button onclick="generateTrendline()">Generate Trendline</button>
        <div id="chartContainer">
            <canvas id="myChart"></canvas>
        </div>
        <div id="results" class="results"></div>
        <button id="downloadJPG" style="display:none;" onclick="
            downloadImage('image/jpeg', 'trendline.jpg')">Download JPG</
            button>
        <button id="downloadPNG" style="display:none;" onclick="
            downloadImage('image/png', 'trendline.png')">Download PNG</
            button>
    </div>
    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/html2canvas/0.4.1/
        html2canvas.min.js"></script>
    <script>
        function generateTrendline() {
            const xData = document.getElementById('xData').value.split(',')
                .map(Number);
            const yData = document.getElementById('yData').value.split(',')
                .map(Number);
```

```javascript
if (xData.length !== yData.length || xData.length === 0) {
    alert("X and Y data points must have the same length and
        cannot be empty!");
    return;
}
const n = xData.length;
const sumX = xData.reduce((a, b) => a + b, 0);
const sumY = yData.reduce((a, b) => a + b, 0);
const sumXY = xData.map((x, i) => x * yData[i]).reduce((a, b)
    => a + b, 0);
const sumX2 = xData.map(x => x * x).reduce((a, b) => a + b, 0);
const gradient = (n * sumXY - sumX * sumY) / (n * sumX2 - sumX
    * sumX);
const intercept = (sumY - gradient * sumX) / n;
const yFit = xData.map(x => gradient * x + intercept);
const residuals = yData.map((y, i) => y - yFit[i]);
const sumRes2 = residuals.map(r => r * r).reduce((a, b) => a +
    b, 0);
const sigma2 = sumRes2 / (n - 2);
const deltaX = xData.map(x => x - sumX / n);
const sumDeltaX2 = deltaX.map(dx => dx * dx).reduce((a, b) => a
    + b, 0);
const uncertaintyGradient = Math.sqrt(sigma2 / sumDeltaX2);
const uncertaintyIntercept = Math.sqrt(sigma2 * (1/n + (sumX *
    sumX) / (n * sumDeltaX2)));
const centroidX = sumX / n;
const centroidY = sumY / n;
document.getElementById('results').innerHTML = `
    <p>Trendline Equation: y = ${gradient.toFixed(2)}x + ${
        intercept.toFixed(2)}</p>
    <p>Uncertainty in Gradient: ${uncertaintyGradient.toFixed
        (2)}</p>
    <p>Uncertainty in Intercept: ${uncertaintyIntercept.toFixed
        (2)}</p>
    <p>Equation for Uncertainty in Gradient:  _m = sqrt(      /
        ( x )  )</p>
    <p>Equation for Uncertainty in Intercept:  _b = sqrt(
        * (1/n + ( x )   / (n *    ( x )  )))</p>
    <p>Centroid Coordinates: (X: ${centroidX.toFixed(2)}, Y: ${
        centroidY.toFixed(2)})</p>
`;
const ctx = document.getElementById('myChart').getContext('2d')
    ;
new Chart(ctx, {
    type: 'scatter',
    data: {
        datasets: [{
            label: 'Data Points',
            data: xData.map((x, i) => ({ x: x, y: yData[i] })),
            backgroundColor: 'rgba(75, 192, 192, 1)',
        },
        {
            label: 'Trendline',
            type: 'line',
            data: xData.map(x => ({ x: x, y: gradient * x +
                intercept })),
            borderColor: 'rgba(255, 99, 132, 1)',
            borderWidth: 2,
```

```
                fill: false,
                showLine: true,
            }]
        },
        options: {
            responsive: true,
            scales: {
                x: { title: { display: true, text: 'X Values' } },
                y: { title: { display: true, text: 'Y Values' } }
            }
        }
    });
    document.getElementById('downloadJPG').style.display = 'block';
    document.getElementById('downloadPNG').style.display = 'block';
}
function downloadImage(type, filename) {
    html2canvas(document.querySelector('canvas')).then(canvas => {
        const link = document.createElement('a');
        link.href = canvas.toDataURL(type);
        link.download = filename;
        link.click();
    });
}
</script>
</body>
</html>
```

3   Equations and Calculations

3.1   Trendline Equation

The trendline equation is determined using the least squares method:

$$y = mx + b$$

where m is the gradient and b is the intercept.

3.2   Uncertainty Calculations

3.2.1   Uncertainty in Gradient

$$\sigma_m = \sqrt{\frac{\sigma^2}{\sum(\Delta x)^2}}$$

where σ^2 is the variance of the residuals and $\sum(\Delta x)^2$ is the sum of the squared deviations of x from the mean.

3.2.2   Uncertainty in Intercept

$$\sigma_b = \sqrt{\sigma^2\left(\frac{1}{n} + \frac{(\sum x)^2}{n\sum(\Delta x)^2}\right)}$$

where σ^2 is the variance of the residuals, n is the number of data points, and $\sum(\Delta x)^2$ is the sum of the squared deviations of x from the mean.

4 Usage

- **Enter Data Points**: Input X and Y data points as comma-separated values.

- **Generate Trendline**: Click the "Generate Trendline" button to calculate the trendline, uncertainties, and centroid coordinates.

- **Download Chart**: Click "Download JPG" or "Download PNG" to save the chart image in the specified format.