

Documentation for Trendline Calculator Webpage

Shafiq R

August 30, 2024

1 Introduction

This document provides comprehensive documentation for the HTML and JavaScript code that forms a Trendline Calculator webpage. This webpage enables users to input a series of X and Y values, compute a linear regression trendline, and display the output in both text and graphical formats. Additionally, the page includes a refresh button to reset inputs and outputs for new calculations.

2 Code Explanation

2.1 HTML Structure

The HTML structure defines the layout and essential components of the webpage. The webpage consists of a head section for metadata and script inclusion, and a body section that houses the input fields, buttons, and output display areas.

2.1.1 Head Section

Listing 1: HTML Head Section

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Trendline Calculator</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
```

- **DOCTYPE Declaration:** The ‘`!DOCTYPE html`’ declaration defines the document type and version of HTML. Here, it indicates that the document follows the HTML5 standard.
- **HTML Element:** The ‘`<html lang="en">`’ tag is the root element of the webpage and indicates that the language of the content is English (‘`lang="en"`’).
- **Head Element:** The ‘`<head>`’ section contains metadata about the document, including:
 - **Character Set:** The ‘`<meta charset="UTF-8">`’ tag specifies the character encoding for the webpage, ensuring it supports a wide range of characters, including special symbols.
 - **Viewport Settings:** The ‘`<meta name="viewport" content="width=device-width, initial-scale=1.0">`’ tag controls the layout on mobile devices by setting the viewport width to match the device’s width and an initial scale of 1.0 for proper rendering.
 - **Title:** The ‘`<title>`’ tag sets the title of the webpage, displayed in the browser tab.
 - **Script Inclusion:** The ‘`<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>`’ tag includes the Chart.js library, a JavaScript library for data visualization, which is used to create the graphical chart.

2.1.2 Body Section

Listing 2: HTML Body Section

```
<body>
  <h1>Trendline Calculator</h1>

  <label for="xValues">X Values (comma separated):</label>
  <input type="text" id="xValues">

  <label for="yValues">Y Values (comma separated):</label>
  <input type="text" id="yValues">

  <button onclick="generateTrendline()">Generate Trendline</button>
  <button onclick="refreshPage()">Refresh Page</button> <!-- Refresh Button -->

  <canvas id="myChart"></canvas>

  <p id="output"></p>

  <button id="downloadJPG" style="display:none;">Download JPG</button>
  <button id="downloadPNG" style="display:none;">Download PNG</button>
</body>
```

- **Header:** The ‘`<h1>Trendline Calculator</h1>`’ tag creates a prominent header for the webpage.
- **Input Fields:**
 - The ‘`<label for="xValues">X Values (comma separated):</label>`’ tag associates a label with the X values input field.
 - The ‘`<input type="text" id="xValues">`’ tag creates a text input field where users can enter the X values as a comma-separated list.
 - Similarly, the Y values input field is created with ‘`<label for="yValues">Y Values (comma separated):</label>`’ and ‘`<input type="text" id="yValues">`’.
- **Buttons:**
 - The ‘`<button onclick="generateTrendline()">Generate Trendline</button>`’ tag creates a button that triggers the ‘`generateTrendline()`’ JavaScript function when clicked.
 - The ‘`<button onclick="refreshPage()">Refresh Page</button>`’ tag creates a button that triggers the ‘`refreshPage()`’ function to reload the page, resetting all inputs and outputs.
- **Canvas Element:** The ‘`<canvas id="myChart"></canvas>`’ tag provides a drawing area for the Chart.js library to render the trendline and data points.
- **Output Paragraph:** The ‘`<p id="output"></p>`’ tag is a placeholder where the text output, including the trendline equation and uncertainties, will be displayed.
- **Download Buttons:** The ‘`<button id="downloadJPG" style="display:none;">Download JPG</button>`’ and ‘`<button id="downloadPNG" style="display:none;">Download PNG</button>`’ tags create buttons for downloading the chart as an image. These buttons are initially hidden and are only displayed after the trendline is generated.

2.2 JavaScript Functions

The JavaScript code implements the core functionality of the Trendline Calculator, including data processing, trendline calculation, chart generation, and page refresh.

2.2.1 Generating the Trendline

Listing 3: Generate Trendline Function

```
function generateTrendline() {
  // Get the X and Y values from the input fields
  const xData = document.getElementById('xValues').value.split(',').map(Number);
```

```

const yData = document.getElementById('yValues').value.split(',').map(Number);

// Calculate the means of X and Y
const meanX = xData.reduce((a, b) => a + b, 0) / xData.length;
const meanY = yData.reduce((a, b) => a + b, 0) / yData.length;

// Calculate the slope (gradient) and intercept
const numerator = xData.map((x, i) => (x - meanX) * (yData[i] - meanY)).reduce((a, b) => a + b, 0);
const denominator = xData.map(x => (x - meanX) ** 2).reduce((a, b) => a + b, 0);
const gradient = numerator / denominator;
const intercept = meanY - gradient * meanX;

// Calculate uncertainties
const yFit = xData.map(x => gradient * x + intercept);
const residuals = yData.map((y, i) => y - yFit[i]);
const residualSumSquares = residuals.map(r => r ** 2).reduce((a, b) => a + b, 0);
const residualVariance = residualSumSquares / (xData.length - 2);

const uncertaintyGradient = Math.sqrt(residualVariance / denominator);
const uncertaintyIntercept = Math.sqrt(
  residualVariance * (1 / xData.length + (meanX ** 2 / denominator))
);

// Calculate the centroid
const centroidX = meanX;
const centroidY = meanY;

// Display the output
document.getElementById('output').innerHTML = `
  Trendline: y = ${gradient.toFixed(2)}x + ${intercept.toFixed(2)}<br>
  Uncertainty in Gradient: ${uncertaintyGradient.toFixed(2)}<br>
  Uncertainty in Intercept: ${uncertaintyIntercept.toFixed(2)}<br>
  Centroid: (${centroidX.toFixed(2)}, ${centroidY.toFixed(2)})<br>
`;

// Generate the chart
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',
      data: [
        { x: Math.min(...xData), y: gradient * Math.min(...xData) + intercept },
        { x: Math.max(...xData), y: gradient * Math.max(...xData) + intercept },
      ],
      type: 'line',
      borderColor: 'rgba(255, 99, 132, 1)',
      fill: false,
    }
  ],
  options: {
    scales: {
      x: {
        type: 'linear',
        position: 'bottom'
      }
    }
  }
});

// Show download buttons
document.getElementById('downloadJPG').style.display = 'inline';
document.getElementById('downloadPNG').style.display = 'inline';
}

```

• Input Retrieval:

- The function begins by retrieving the X and Y values from the respective input fields using ‘document.getElementById(‘xValues’).value’ and ‘document.getElementById(‘yValues’).value’. The values are then split into arrays of numbers using ‘split(‘,’’).map(Number)’.

- **Mean Calculation:**

- The mean of the X and Y values is calculated using the ‘reduce’ method, which sums all elements and then divides by the number of elements.

- **Slope and Intercept Calculation:**

- The slope (gradient) is computed using the least squares method, which minimizes the sum of the squared residuals (differences between observed and predicted Y values). The intercept is then calculated from the slope and means.

- **Uncertainty Calculation:**

- Uncertainties in the slope and intercept are estimated using the residual sum of squares and the number of data points. These uncertainties indicate the reliability of the trendline’s parameters.

- **Centroid Calculation:**

- The centroid (mean point) is calculated and displayed, representing the average X and Y values.

- **Output Display:**

- The calculated trendline, uncertainties, and centroid are displayed in the webpage’s output section using the ‘innerHTML’ property.

- **Chart Generation:**

- A scatter plot of the data points and the trendline is generated using Chart.js. The chart is displayed in the canvas element.

- **Download Buttons:**

- The download buttons for saving the chart as JPG or PNG are made visible after the trendline is generated.

2.2.2 Page Refresh Function

Listing 4: Refresh Page Function

```
function refreshPage() {
    location.reload(); // Refresh the webpage
}
```

- The ‘refreshPage()’ function simply reloads the current webpage, resetting all inputs and outputs to their default states. This is done using the ‘location.reload()’ method.