

CSE564: Visualization and Visual Analytics Lab 1

Report

by

Rutvik Parekh

112687483

1. Dataset

The dataset selected for analysis is related to various statistics pertaining to top 5000 movies on IMDb. It contains various numerical as well as categorical variables. I have selected 10 numerical and 5 categorical variables for analysis.

The numerical variables selected are:

- 1) Number of critics for review
- 2) Duration of the movie
- 3) Actor 1 Facebook likes
- 4) Actor 2 Facebook likes
- 5) Actor 3 Facebook likes
- 6) Director Facebook likes
- 7) Movie Facebook likes
- 8) IMDb score of the movie
- 9) Budget of the movie
- 10) Box-office gross of the movie

The categorical variables selected are:

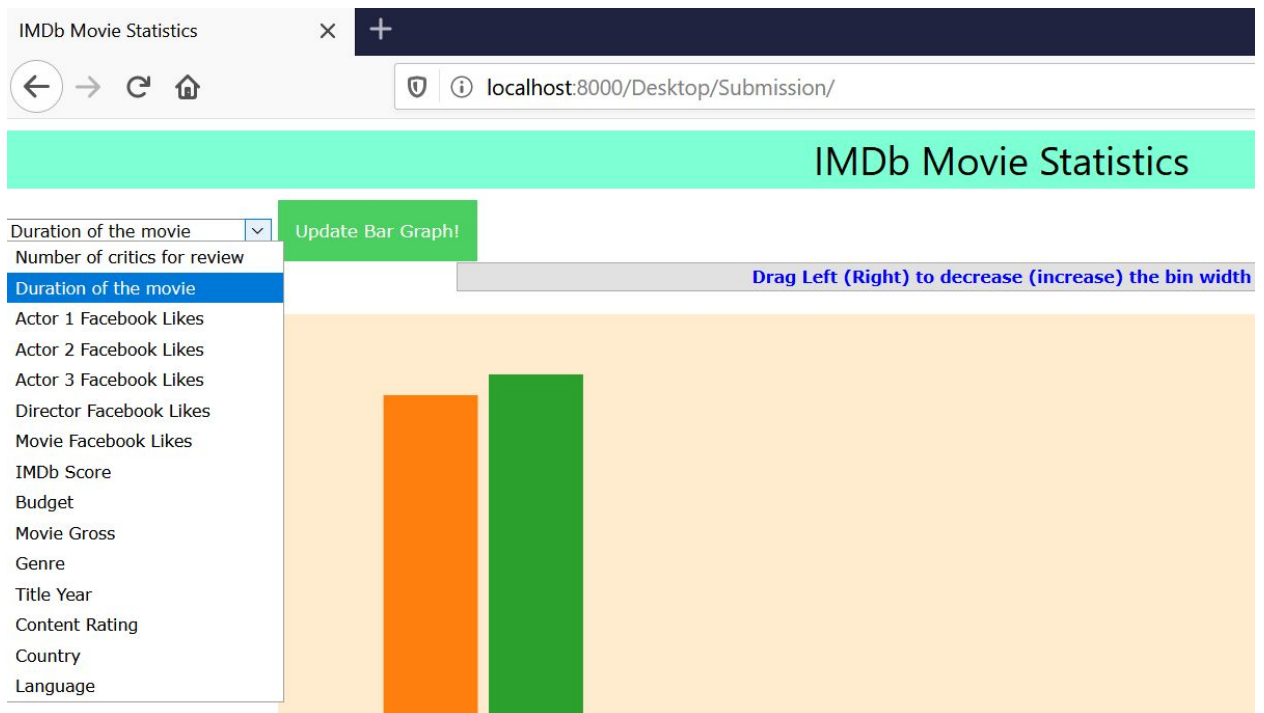
- 1) Genre of the movie
- 2) Year in which the movie released
- 3) Content rating of the movie
- 4) Country in which the movie was made
- 5) Language in which the movie was made

For the categorical variables, preliminary analysis was done using Python code. The counts of the five categorical variables were counted using Python.

2. Tasks

The following section contains implementation details related to all the tasks, code snippets and images from the demo.

a. present a menu to allow users to select a variable and update chart

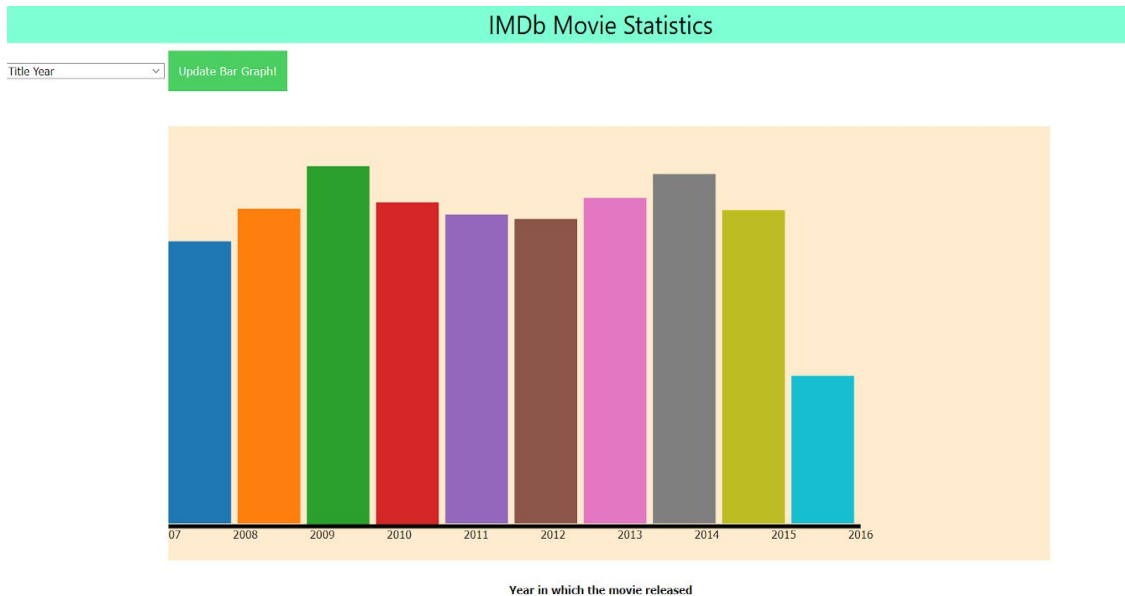


As you can see in the image, a **menu** is presented to the user to select the variable and a button is provided for him to update the bar chart. In case of **numerical variables**, it will be updated to a **histogram** and in case of **categorical variables**, it will be updated to a **bar chart**.

```
<select id="dataVariable">
  <option value="numCritics" selected="selected">Number of critics for review</option>
  <option value="duration">Duration of the movie</option>
  <option value="actor1FbLikes">Actor 1 Facebook Likes</option>
  <option value="actor2FbLikes">Actor 2 Facebook Likes</option>
  <option value="actor3FbLikes">Actor 3 Facebook Likes</option>
  <option value="directorFbLikes">Director Facebook Likes</option>
  <option value="movieFbLikes">Movie Facebook Likes</option>
  <option value="imdbScore">IMDb Score</option>
  <option value="budget">Budget</option>
  <option value="movieGross">Movie Gross</option>
  <option value="genre">Genre</option>
  <option value="titleYear">Title Year</option>
  <option value="contentRating">Content Rating</option>
  <option value="country">Country</option>
  <option value="language">Language</option>
</select>
<button class="dropbtn" id="variable">Update Bar Graph!</button>
```

In the above code snippet, the dropdown menu is implemented as a 'select' tag in HTML with various options in it. Along with it, a button is provided to provide the value of the selected option for updating the bar chart or histogram.

b. draw a bar chart if a categorical variable is selected

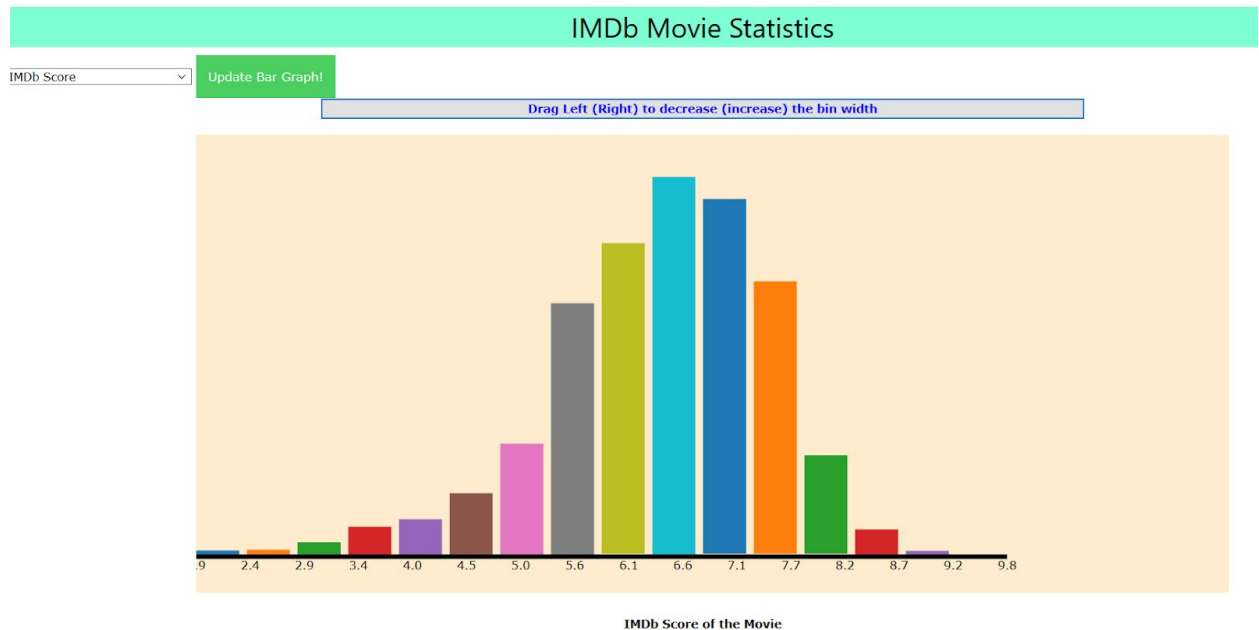


As can be seen in the image, a bar graph is presented if a categorical variable is selected. In this example, the bar graph shows the number of movies vs. the year in which it was released.

```
function drawHistOrBarGraph(data, categoricalCount, bins, isCategorical){  
    if(isCategorical){  
        document.getElementById("bar").innerHTML = '';  
        generateBarGraph(categoricalCount, data);  
    }  
    else{  
        var binWidth = (d3.max(data) - d3.min(data))/(bins-1);  
  
        frequency = Array.apply(null, Array(bins)).map(Number.prototype.valueOf,0);  
        binValues = [];  
  
        data.forEach(function(d){  
            frequency[Math.floor((d - d3.min(data))/binWidth)]++;  
        })  
  
        var min = d3.min(data);  
        for(i=0; i<bins; i++){  
            var end = (min+binWidth);  
            binValue = (min+end)/2;  
            binValue = binValue.toFixed(1);  
            binValues.push(binValue);  
            min = end;  
        }  
  
        document.getElementById("bar").innerHTML = '';  
        generateBarGraph(frequency, binValues);  
        resizeBins();  
    }  
}
```

In the above code snippet, I have implemented a function `drawHistOrBarGraph()` which will draw a bar graph if a categorical variable is selected and a histogram if a numerical variable is selected.

- c. draw a histogram if a numerical variable is selected (bin it into a fixed range (equi-width) of your choice)

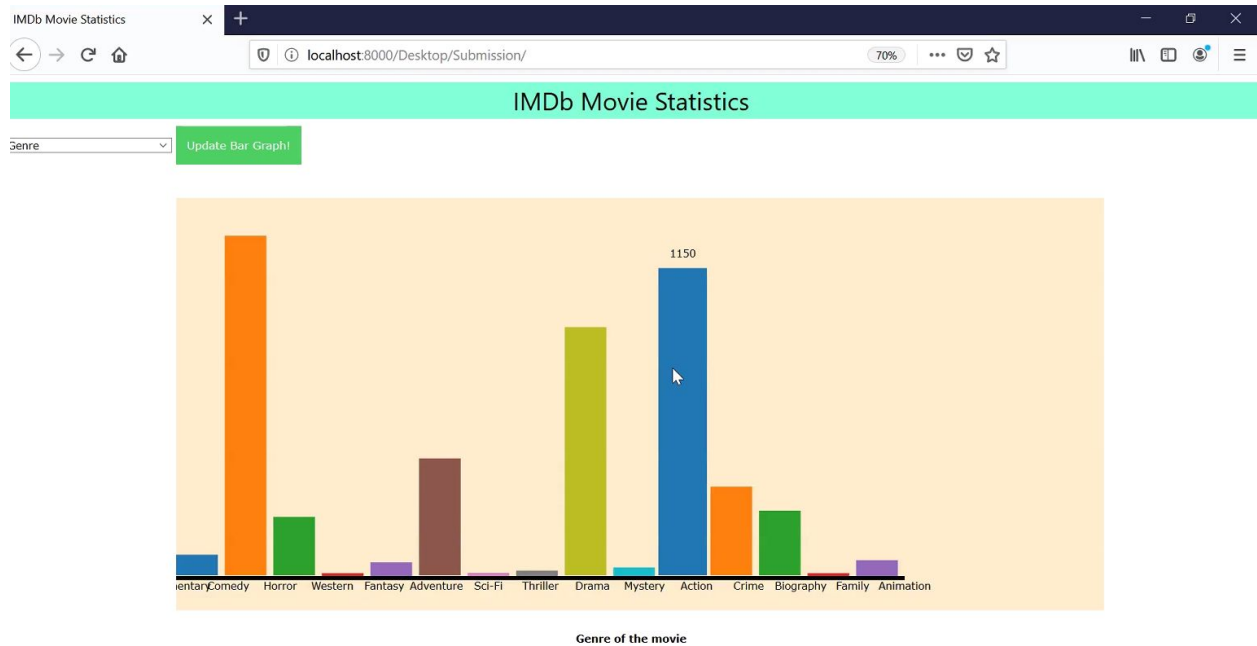


In the above image, a numerical variable is selected and a histogram is plotted.

```
function generateBarGraph(frequency, xValues){  
    var x = d3.scale.ordinal()  
        .rangeRoundBands([0, w], 0.5);  
  
    var y = d3.scale.linear()  
        .range([h, 0]);  
  
    x.domain(xValues).rangePoints([0, w]);  
  
    y.domain([0, d3.max(frequency, function(d) {  
        return d;  
    })]);  
}
```

The above code snippet shows the function `generateBarGraph()` which does the work of generating bar graph or histogram by using various features of `d3.js`. In this function, all the important features of `d3.js` are used such as `.attr()`, `.domain()`, `.range()`, `.enter()`. to display the bar graph from the provided data.

- d. on mouse-over display the value of the bar on top of the bar
- e. on mouse-over also make the bar wider and higher to focus on it



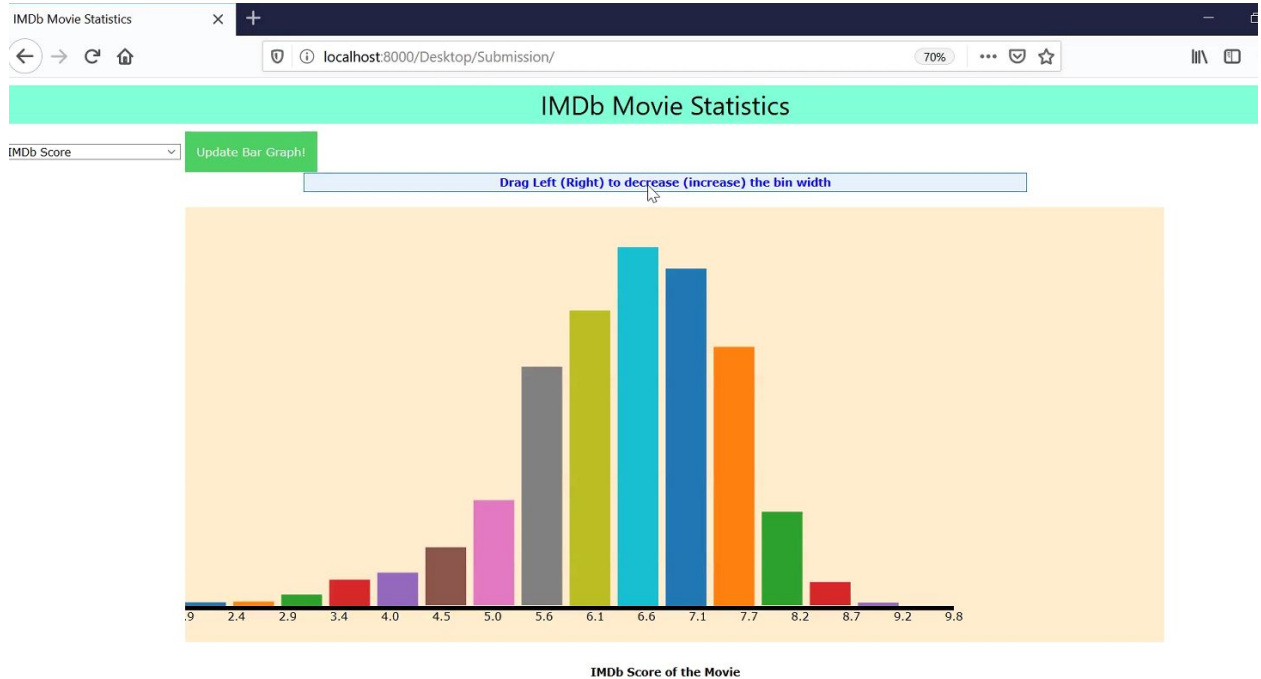
As we can see from the image above, the value of the bar is displayed on top of the bar if the mouse is hovering over it and also the width and the height of the bar is increased.

```
.on("mouseover", function(d,i) {
  d3.select(this)
    .attr("x",i * (w / frequency.length) - 5)
    .attr("y", parseInt(d3.select(this).attr("y"))) - 15)
    .attr("width",w / frequency.length - padding + 10)
    .attr("height",parseInt(d3.select(this).attr("height")) + 15);

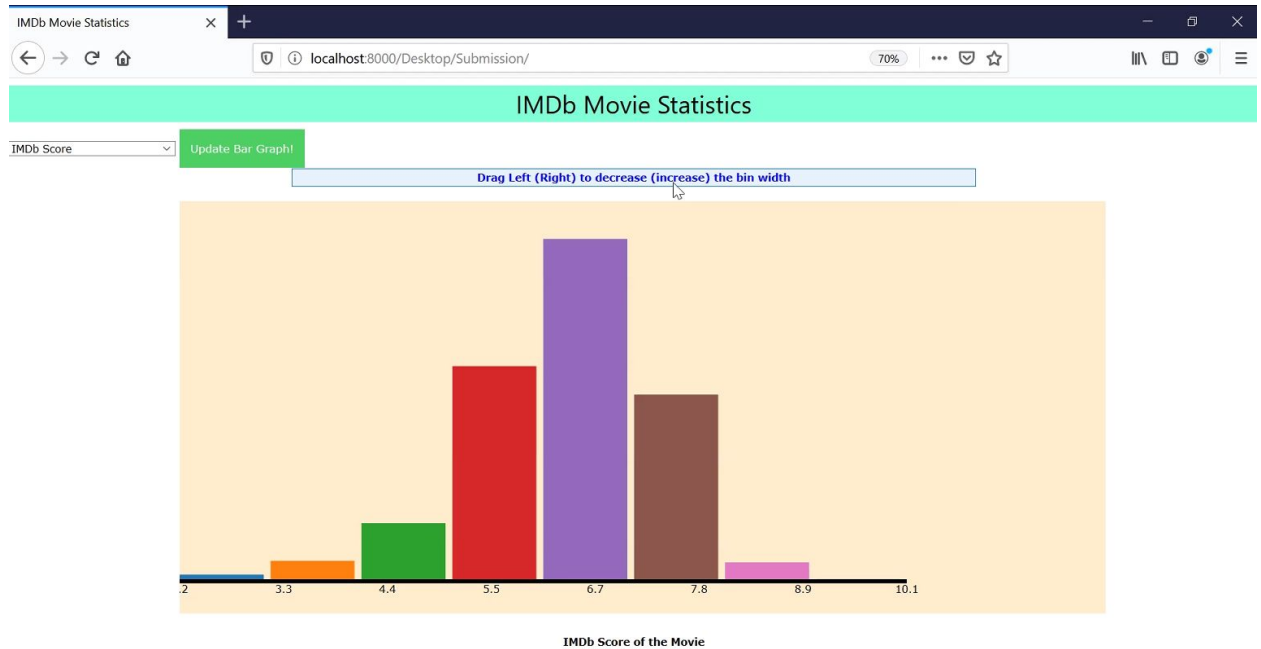
  tip.html( "<span style='color:black'>" + d + "</span>");
  tip.show();
})
```

The above code snippet shows that an event can be added such that if we take the mouse over to a single bar in the graph, it shows the value of the bar on top of the bar. The code **increases the width and the height of the selected bar on triggering of this event**. A **tool-tip** is added which can show the **value of the bar on top of the bar**.

f. mouse (with left mouse button down) move left (right) should decrease (increase) bin width/size (for numerical variables only)



As we can see from the above image, the slider on top of the histogram is used to drag the mouse left/right to decrease/increase the bin size. It currently shows the frequency distribution of the IMDb scores of the movies. On dragging the mouse right, we can obtain the below image.



```
function draggableMove() {
  if(d3.mouse(sec.node())[0] + 20 < horizontal && bins < 20){
    bins = bins + 1;
    drawHistOrBarGraph(data, null, bins, false);
    horizontal = d3.mouse(sec.node())[0];
  }
  else if(d3.mouse(sec.node())[0] - 20 > horizontal && bins > 5){
    bins = bins - 1;
    drawHistOrBarGraph(data, null, bins, false);
    horizontal = d3.mouse(sec.node())[0];
  }
}
```

The above code snippet shows the function which is executed when the 'mousedown' event is triggered. It checks the position of the mouse pointer and compares it with the previous position. If the mouse has moved to the left, the number of bins are increased (reducing the bin width). If the mouse has moved to the right, the number of bins are decreased (increasing the bin width).

3. Youtube Link for the voice-narrated video:

https://www.youtube.com/watch?v=iYVGU0_X-e0

4. Running the project:

- 1) Open the python command prompt
- 2) Run the following command on the python command prompt: `python -m http.server 8000`
- 3) Open the link in either Mozilla Firefox or Google Chrome: <https://localhost:8000/>
- 4) Unzip the project folder and navigate to the project through <https://localhost:8000/>
- 5) Open index.html