LAB REPORT 1

CSE 564 VISUALIZATION

NAME: ADITYA KUMAR

SBU ID: 112716642

SUBMISSION DATE: 20th Feb 2020

DATASET

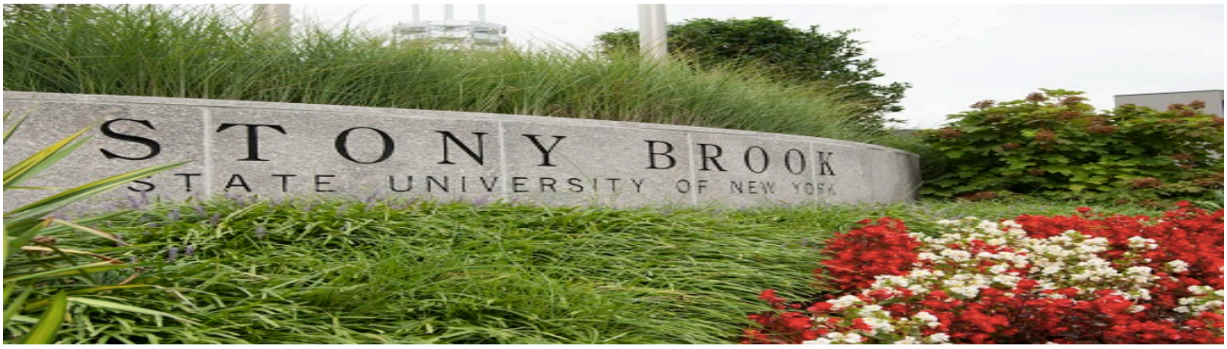
For this lab, FIFA Dataset is used to perform the given tasks. The dataset is retrieved from kaggle. FIFA Dataset contains columns like Player Name, ID, Age, This Dataset is fused with another dataset based upon the region that a Fifa player represents. Based upon region, this dataset is mapped to categories like Population density, GDP, Area and population.

Total No. of Rows – 1553
Total No. of Columns – 21

1553 rows and 15 columns (categories) are used for visualization.

SCREENSHOT OF DASHBOARD

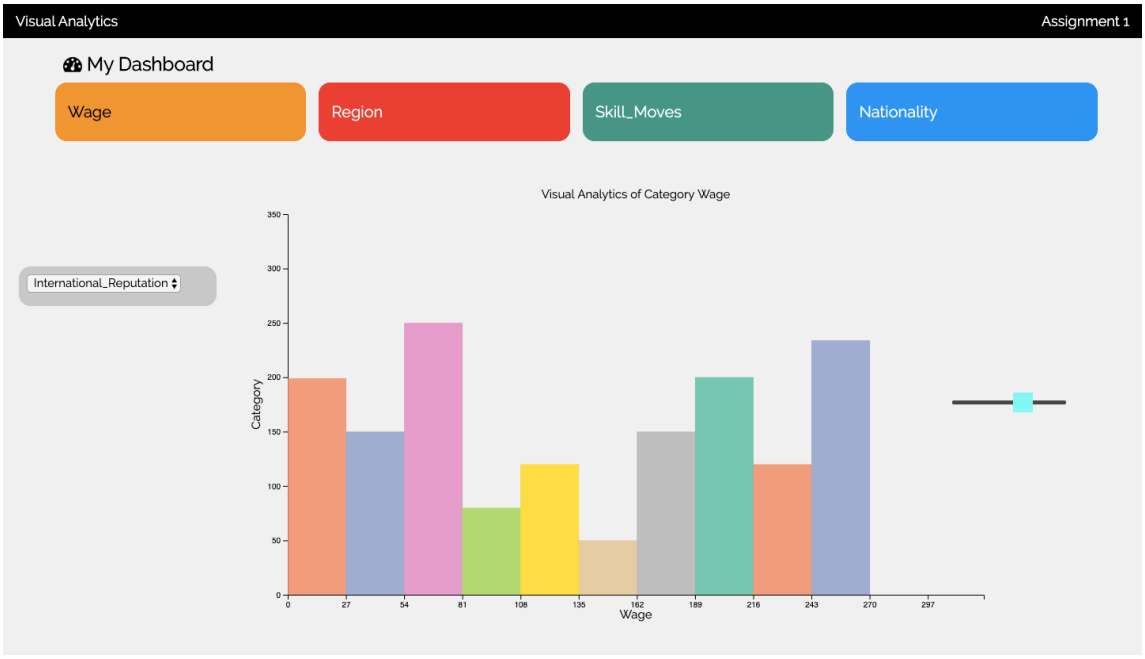
HOME PAGE :



CSE 564 VIZUALIZATION ASSIGNMENT 1

Dashboard

DASHBOARD



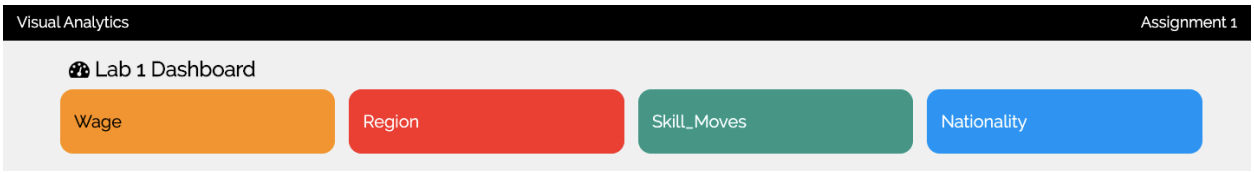
TASKS

1. **Present a menu to allow users to select a variable and update chart**

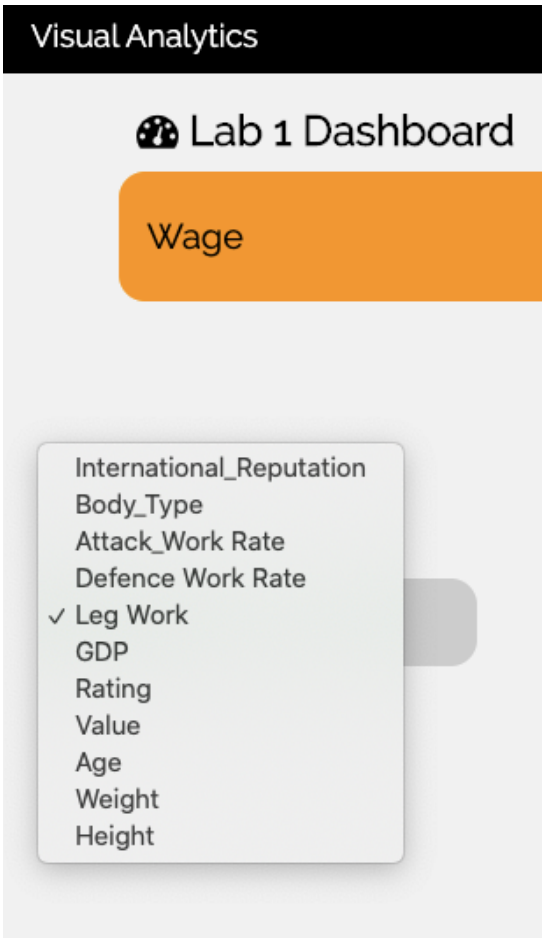
   Two types of menu are presented for this task. One is the global menu (Wage, Region, Skill_Moves, Nationality) which gives the visual analytics of some most popular categories. Other menu is presented in the form of dropdown, which contains rest of the categories.

   SNAPSHOTS OF MENU

   GLOBAL MENU



DROPDOWN MENU



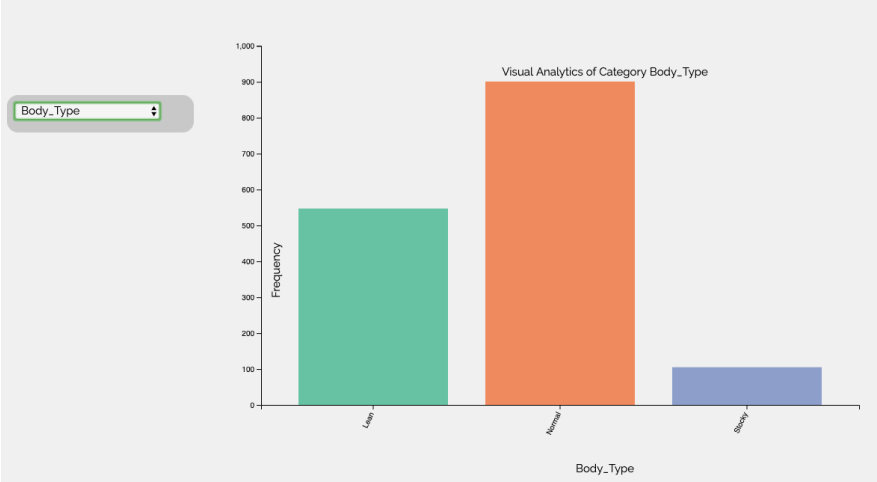
CODE SNIPPET

```
var dropdownChange = function() {
      var new_feature = d3.select(this).property('value');
    feature_name = new_feature;
      nBin = 10;
    slider.value = 10;

      if(categorical_features.includes(new_feature)){
          renderGraph(new_feature, DATA, "CATEGORICAL");
      } else {
          renderGraph(new_feature, DATA, "NUMERICAL", (16 - nBin) );
      }

      console.log("change "+ new_feature);
  };

  var dropdown = d3.select("select")
                  .on("change", dropdownChange)
                  .attr("id", "select_list");

  dropdown.selectAll("option")
          .data(categorical_features.concat(numerical_features))
          .enter().append("option")
          .attr("value", function (d) { return d; })
          .text(function (d) { return d; });


  dropdown.selectAll("option")
          .data(numerical_features)
          .enter().append("option")
          .attr("value", function (d) { return d; })
          .text(function (d) { return d; });

select_list = document.getElementById("select_list");
```

2. **Draw a bar chart if a categorical variable is selected**

   Snapshot depicts the bar graph after a categorical variable Body Type is selected.

   SNAPSHOT



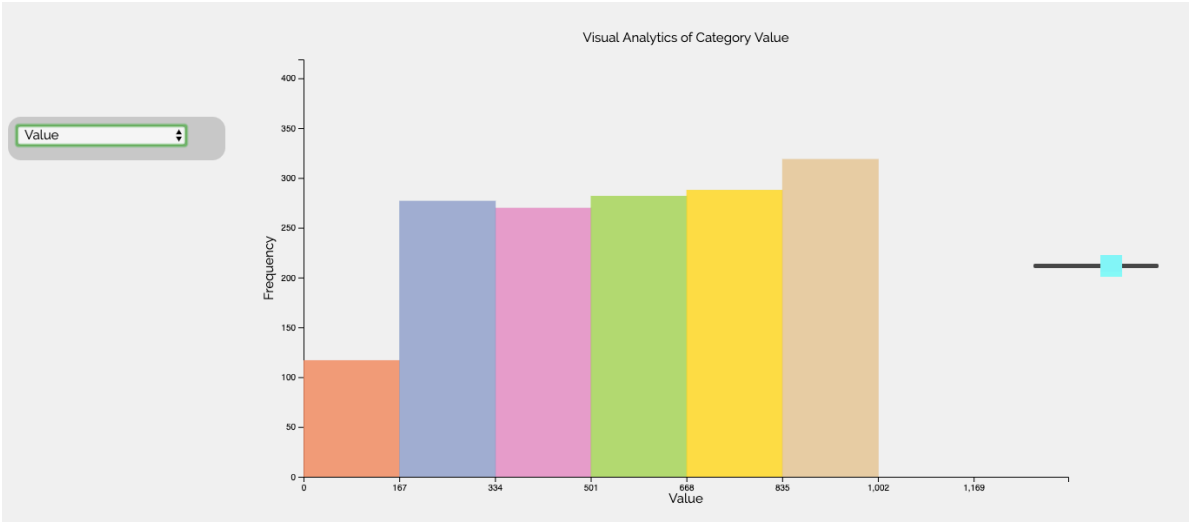
   CODE SNIPPET

```
function renderGraph(feature_name, data, type,nBin){

    if(type == "CATEGORICAL"){
        slider.style.visibility = "hidden";
        renderBarGraph(feature_name, data);
```

3. **Draw a histogram if a numerical variable is selected**

   Snapshot depicts the histogram after a numerical variable Player's value is selected.

SNAPSHOT



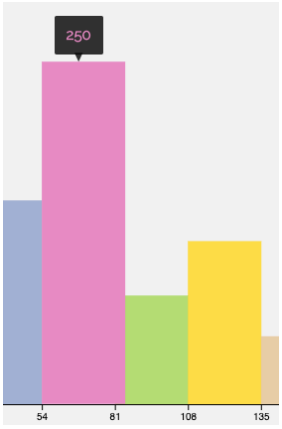
CODE SNIPPET

```
if(categorical_features.includes(new_feature)){
    renderGraph(new_feature, DATA, "CATEGORICAL");
} else {
    renderGraph(new_feature, DATA, "NUMERICAL", (TOTAL - nBin) );
}
```

4. **On mouse-over display the value of the bar on top of the bar**

   Tip appears as we hover over the bar.

   SNAPSHOT



CODE SNIPPET

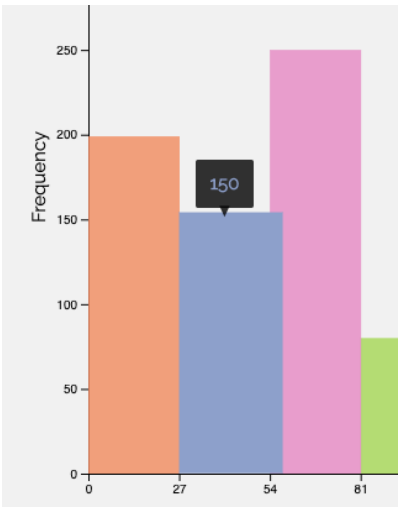
```
var tip = d3.tip()
  .attr('class', 'd3-tip')
  .offset([-10, 0])
  .html(function(d) {
    return "<span style='color:red'>" + d.value + "</span>";
})
```

```
.on('mouseover', tip.show)
.on('mouseout', tip.hide)
```

5. **On mouse-over also make the bar wider and higher to focus on it**

   Opaque attribute is changes over mouse events.

   SCREENSHOT



CODE SNIPPET

```
.on('mouseenter', function (s, i) {
    d3.select(this).raise();

    d3.select(this)
        .transition()
        .duration(200)
        .attr('opacity', 1)
        .attr('width', rectWidth + 10)
        .attr('height', (s) => height - yScale(s.value)
        .style("transform", "scale(1,0.979)"); // Doubt

})
.on('mouseleave', function (actual, i) {
    d3.select(this)
    .attr("opacity", 0.6)
    .transition()
    .duration(200)
    .attr('width',rectWidth)
    .attr('height', (s) => height - yScale(s.value))
    .style("transform", "scale(1,1)");
```

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

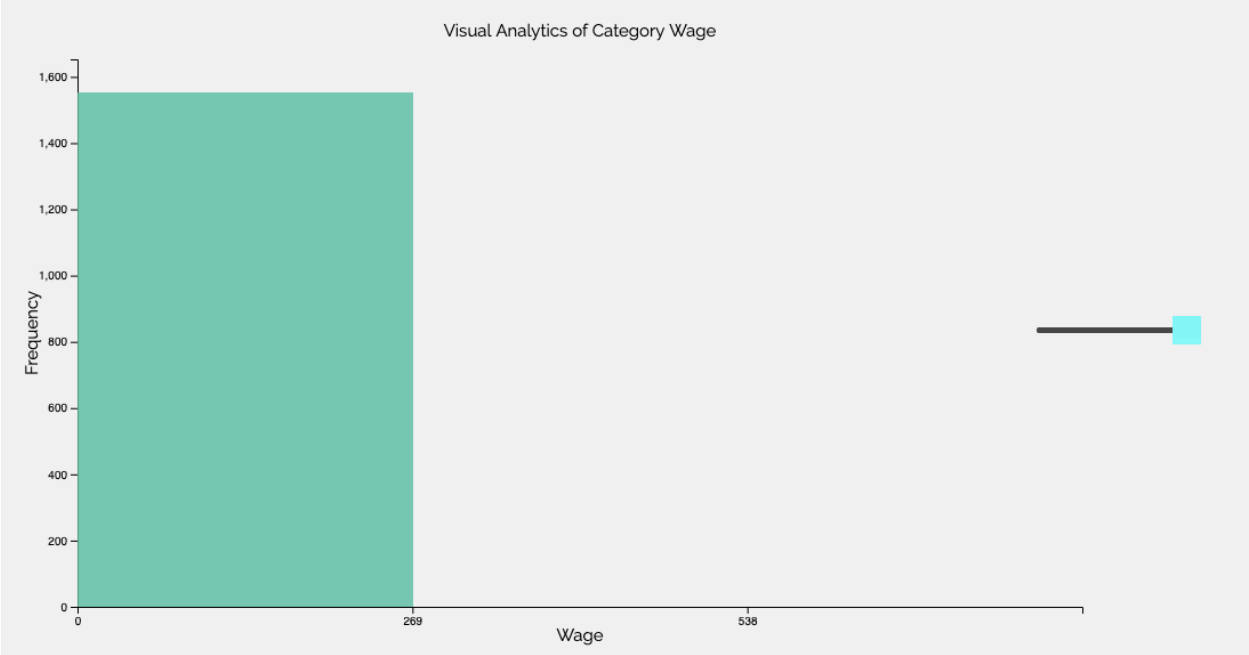
   Implemented in 2 ways:

   1st Method: Slider appears on screen, moving it left increases the no. of bins as per the question.

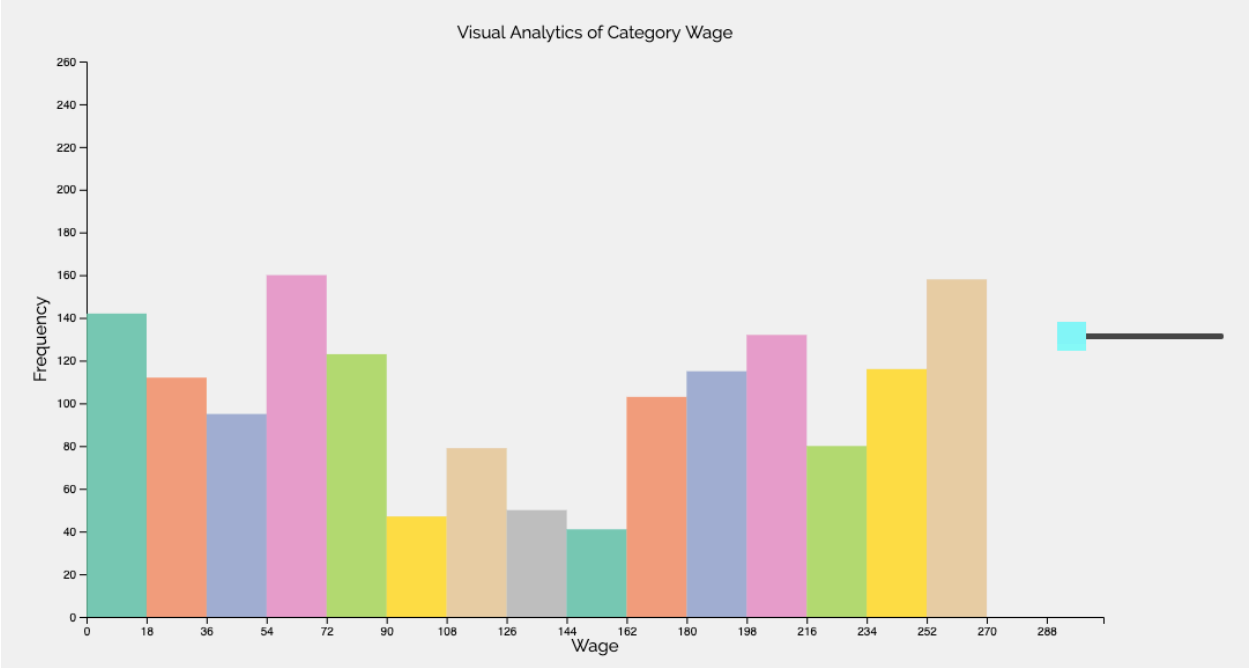
   2nd Method: To make something elegant, Mousedown and drag events are used to adjust the bin width.

   SNAPSHOT

   When Slider is at Extreme Right



When Slider is at Extreme Left



CODE SNIPPET

```
function WhichButton(event){

  if(event.which == '1')
  {
      if(nBin<15)
      {
          nBin = nBin + 1;
          slider.value = 16 - nBin;
      }
      update_histogram(feature_name, DATA, nBin);
  }
    else if(event.which== '2')
    {
        if(nBin>1)
        {
        nBin = nBin - 1;
        slider.value = 16 - nBin;
        }
    }
    update_histogram(feature_name, DATA, nBin);
```

YOUTUBE LINK

https://www.youtube.com/watch?v=dFikOnFiLBc