HU Extension **Assignment 08** E-63 Big Data Analytics

**Problem1**

Calculate histogram of values for “Per capita total spending on health (PPP int. $)” using R and save output as “”

This is for bucket size 200. The same process was repeated for the 500 and 1000 bucket sizes and the files attached.

**Input values and create histogram**

var values= [44,577,330,3254,168,991,1287,239,3441,4388,579,1988,1083,57,1523,786,4025,378,65,275,233,972,…….,84,77,229,1449,483,1029,199,473,124,519,1544,3480,83,8362,1188,184,240,589,215,122,90];

var data = d3.layout.histogram()

.bins(x.ticks(42))

(values);

**Add margins and height and width**

var margin = {top: 10, right: 30, bottom: 40, left: 40},

width = 960 - margin.left - margin.right,

height = 500 - margin.top - margin.bottom;

**Set X and Y Axis and format**

var x = d3.scale.linear()

.domain([0, 8400])

.range([0, width]);

var y = d3.scale.linear()

.domain([0, d3.max(data, function(d) { return d.y; })])

.range([height, 0]);

var xAxis = d3.svg.axis()

.scale(x)

.orient("bottom");

var yAxis = d3.svg.axis()

.scale(y)

.ticks(8)

.orient("left");

**Define svg element**

var svg = d3.select("body").append("svg")

.attr("width", width + margin.left + margin.right)

.attr("height", height + margin.top + margin.bottom)

.append("g")

.attr("transform", "translate(" + margin.left + "," + margin.top + ")");

**Define bar elements in svg, bind histogram data and format**

var bar = svg.selectAll(".bar")

.data(data)

.enter().append("g")

.attr("class", "bar")

.attr("transform", function(d) { return "translate(" + x(d.x) + "," + y(d.y) + ")"; });

bar.append("rect")

.attr("x", 1)

.attr("width", x(data[0].dx) - 1)

.attr("height", function(d) { return height - y(d.y); });

**Add the x axis and x-label**

svg.append("g")

.attr("class", "x axis")

.attr("transform", "translate(0," + height + ")")

.call(xAxis);

svg.append("text")

.attr("class", "xlabel")

.attr("text-anchor", "middle")

.attr("x", width / 2)

.attr("y", height + margin.bottom-5)

.text("Spending on health per capita");

**Add the y axis and y-label**

svg.append("g")

.attr("class", "y axis")

.attr("transform", "translate(0,0)")

.call(yAxis);

svg.append("text")

.attr("class", "ylabel")

.attr("y", 0 - margin.left) // x and y switched due to rotation

.attr("x", 0 - (height / 2))

.attr("dy", "1em")

.attr("transform", "rotate(-90)")

.style("text-anchor", "middle")

.text("Frequency(Number of Countries");

**Use style tags to define hover function for bars**

.bar rect {

fill: steelblue;

shape-rendering: crispEdges;

}

.bar rect:hover{

fill: rgba(0,0,0,.8);

}

**Output**

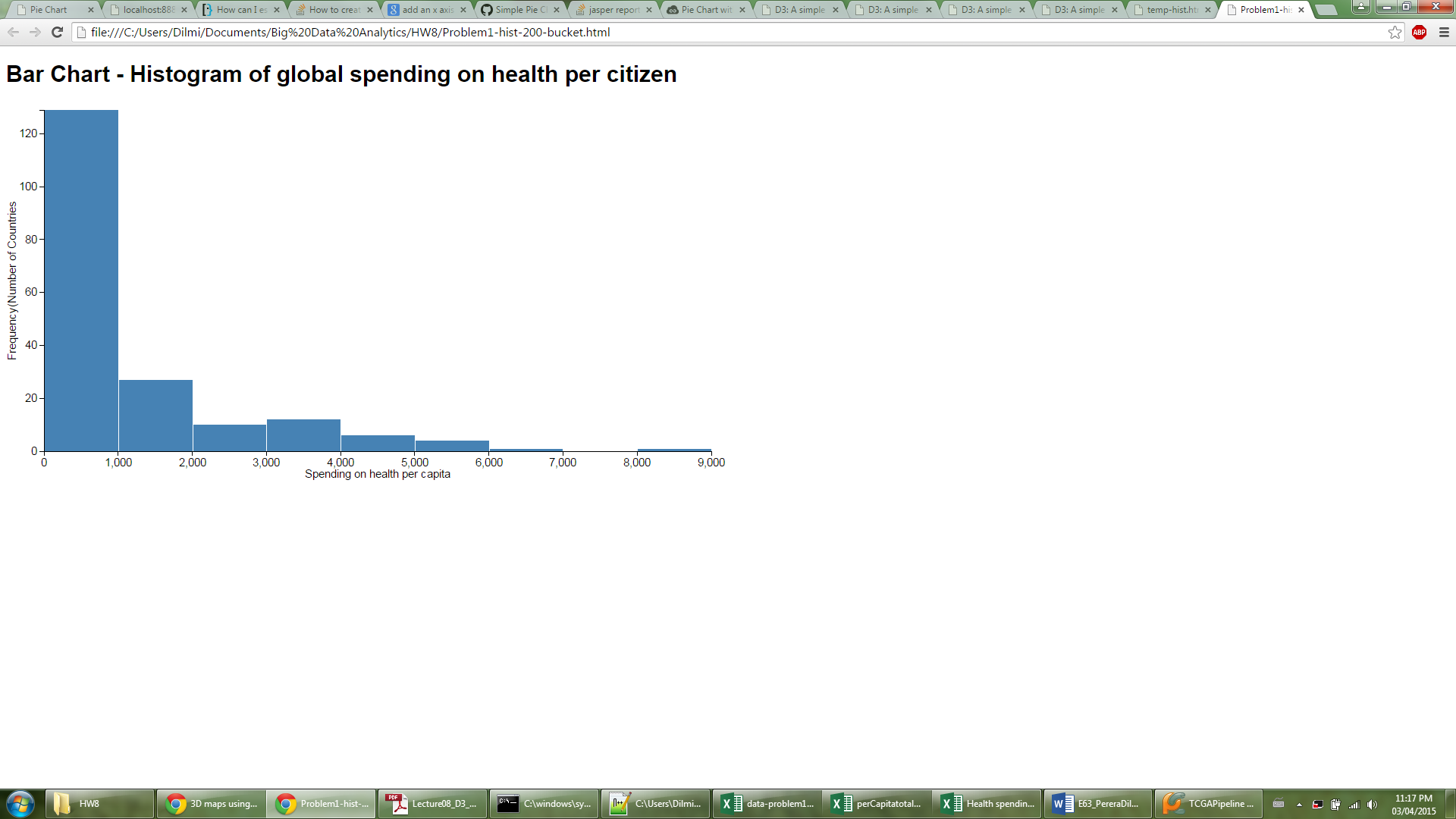
200 bucket



500 bucket



1000 bucket



**Problem 2**

**Complete html doc is attached. I have highlighted the important steps here. Used bucket size of 1000**

data = [{"label":"0-1000","value":67.89},

{"label":"1000-2000","value":14.21},

{"label":"2000-3000","value":5.26},

{"label":"3000-4000","value":6.32},

{"label":"4000-5000","value":3.16},

{"label":"5000-6000","value":2.11},

{"label":">6000","value":1.06}];

**Set boarders, height width etc and input svg element.**

var w = 500,

h = 500,

r = 200,;

var vis = d3.select("body")

.append("svg:svg")

.data([data])

.attr("width", w)

.attr("height", h)

.append("svg:g") //make a group to hold pie chart

.attr("transform", "translate(" + r + "," + r + ")") //move the center of the pie

**Used input colour range to get different colours**

var color = d3.scale.category20c();

**create <path> elements using arc data**

var arc = d3.svg.arc().outerRadius(r);

**Input data into a pie chart layout**

var pie = d3.layout.pie.value(function(d) { return d.value; });

**Bind data to pie chart and format output**

var arcs = vis.selectAll("g.slice") .data(pie).enter

.append("svg:g")

.attr("class", "slice");

**Set the color for each slice to be chosen from the color variable**

arcs.append("svg:path")

.attr("fill", function(d, i) { return color(i); } ) .attr("d", arc);

**Set text lables to each slice and format**

arcs.append("svg:text")

.attr("transform", function(d) {

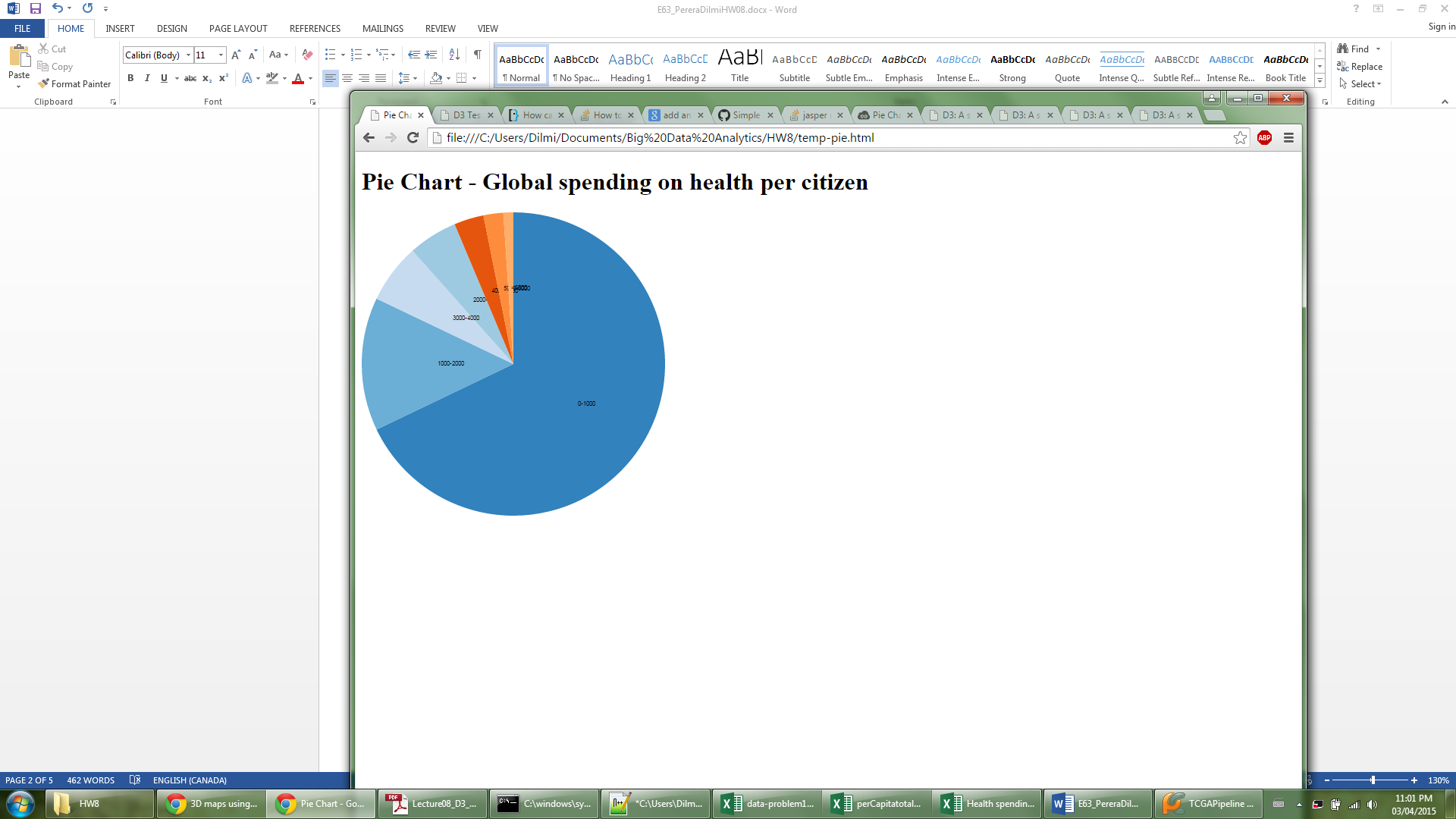
d.innerRadius = 0;

d.outerRadius = r;

return "translate(" + arc.centroid(d) + ")"; })

.attr("text-anchor", "rad").text(function(d, i) { return data[i].label; });

Output



**Problem 3**

**Complete html doc is attached. I have highlighted the important steps here**

**Select 20 different countries and extract relavant values into an array**

var dataset = [[8362,24.22,79,"United States"],[215,12.24,76,"Vietnam"],[5426,41.59,81,"Norway"],

[5394,40.7,83,"Switzerland"],[132,6.49,65,"India"],[4404,19.8,81,"Canada"],[4388,48.53,81,"Austria"],

[4332,36.01,81,"Germany"],[4021,34.47,83,"France"],[998,43.089,70,"Russia"],[3704,31.73,81,"Ireland"],

[3480,27.43,82,"United Kingdom"],[3441,29.91,82,"Australia"],

[16,0.5,62,"Eritrea"],[27,32.9,50,"Congo, Dem Rep"],

[713,1.47,74,"Colombia"],[34,4.57,65,"Myanmar"],[36,1.61,64,"Madagascar"],[37,0.19,58,"Niger"],[379,14.15,75,"China"]

];

**Set boarders, height width etc and input svg element**

var margin = {top: 100, right: 300, bottom: 100, left: 100};

var w = 1000; var h = 1000;

var svg = d3.select("body").append("svg")

.attr("width", w + margin.left + margin.right)

.attr("height", h + margin.top + margin.bottom)

.append("g")

.attr("transform", "translate(" + margin.left + "," + margin.top + ")");

**Input and format circle elements to have**

svg.selectAll("circle")

.data(dataset).enter().append("circle")

**Circles x position as per capita total spending**

.attr("cx", function(d) { return d[0]; })

**Circles y position as doctors per 10000 population**

.attr("cy", function(d) { return d[1]; })

**radius LE-45**

.attr("r", function(d) {

return d[2]-45; })

**color – determined by LE buckets of size 5**

.attr("fill", function(d) {

if(d[2]>45 && d[2]<=50 ){return "Green";}

else if (d[2]>50 && d[2]<=55 ){return "Magenta";}

else if (d[2]>55 && d[2]<=60 ){return "Blue";}

else if (d[2]>60 && d[2]<=65 ){return "Red";}

else if (d[2]>65 && d[2]<=70 ){return "Yellow";}

else if (d[2]>70 && d[2]<=75 ){return "Orange";}

else if (d[2]>75 && d[2]<=80 ){return "Pink";}

else if (d[2]>85 && d[2]<=85 ){return "Purple";}

else{return "Black";}} );

**Input Text (Country Name) to name the circles**

svg.selectAll("text")

.data(dataset)

.enter()

.append("text")

.text(function(d) {

return d[3]; })

.attr("x", function(d) {return d[0]; })

.attr("y", function(d) {return d[1]; })

.attr("font-family", "sans-serif")

.attr("font-size", "11px")

.attr("fill", "Black");

Output

