DSDV Midterm Practice

1. 3 Questions - 90 mins
2. Cover from W01 - W05

Critical points of W02 - Perception

1. Perception VS Cognition
2. Dimensions: Hue - Saturation - Value
3. Paint mixing: light mixing (additive) VS ink mixing (subtractive) VS paint mixing (subtractive - wavelength)
4. Colormap: categorical vs ordered, sequential vs diverging, segmented vs continuous, univariate vs bivariate
5. Qualitative data vs. quantitative data (value, saturation, not hue)
6. Bin or not to bin
7. Caution to color blindness: red-green weakness/blindness
8. Color is relative to brightness contrast

Critical points of W03 - Pop out

1. Difference in hue/curvature
2. Task: target detection, boundary detection, region tracking, counting/estimation
3. To find meaning in what we see, we must selectively pay attention to what is important
4. Gestalt principles: patterns that transcend the visual stimuli that produced them, grouping/linking by placing entities in close proximity, Co-modulation of a channel color, shape, size, value, orientation, texture

Critical points of W04 - Data

1. Dataset types: table, network, field continuous, geometry spatial, multidimensional table, tree
2. Data unit types: items, attributes, links, positions, grids
3. Structure vs Unstructured vs Semi-structure data
4. InfoVis vs SciVis

5. Attribute types: categorical, quantitative (nominal vs ordinal vs interval vs ratio equals, not equals/ sign/ plus, minus/multiply, division
6. Sequential vs Diverging data

Critical points of W05 - Mark vs Channel

1. Marks: represent items or links (points, lines, areas, connection, containment)
2. Channels: change appearance based on attribute = visual variable (position, color, shape, tilt, size, volume 3D)
3. Magnitude channel for ordinal, quantitative data VS identity channels for categorical data
4. Expressiveness principle: the visual encoding should express all of, and only, the information in the dataset attributes
5. Effectiveness principle: the importance of the attribute should match the salience of the channel
6. Characteristics: selective, associative, quantitative, order, length
7. Position: Strongest visual variables, Suitable for all data types
8. Length, size: Good for 1D, OK for 2D, Bad for 3D
9. Luminance: OK for quantitative data when length & size are used. Not very many shades are recognizable
10. Color: Good for qualitative data (identity channel)
11. Shape: Great to recognize many classes. No grouping, ordering

Critical points of W06 - Design Guideline

1. The visualization should show all of the data and only the data
2. Use the best visual channel available for the most important aspect of the data
3. Show data variation, not design variation; Clear, detailed, and thorough labeling and appropriate scales; Size of the graphic effect should be directly proportional to the numerical quantities
4. Focus on LIE FACTOR, SCALE DISTORTION, ZERO-STARTING BASELINE, FRAMING, BIASES, AGGREGATED CHARTS, PIE CHARTS

Critical points of W07 - Visualization Design Principles

1. Data-Ink ratio
2. Chart Junk
3. Alignment matter
4. Unjustified 3D

Sample question:

1. Data type
2. Mark, Channel
3. Suggest to design flaws in charts and renovate them
4. Given data, visualize it

Code guidelines:

1. HTML

```
<link rel="stylesheet" href="style.css" />
<script src="https://d3js.org/d3.v4.min.js"></script>
<script src="script.js"></script>

<div class="task">
  <div id="tooltip" class="hidden"></div>
  <script>
    task();
  </script>
</div>
```

2. CSS

```
.task-controls {
  display: flex;
  width: 50%;
  margin-bottom: 10px;
}

.task-controls button {
```

```
  margin: 0;
  margin-right: 20px;
}

.task-controls select {
  margin: 0;
  margin-right: 20px;
}

.task {
  position: relative;
}
```

3. JS

```
// Step 1: Define basic constraints (w, h, p)
// Step 2: Define row converter for CSV
let rowConverter = function (d) {
  return {
  };
};
// Step 3: Driver code // task -> d3.csv -> d3. .. ....
function task() {
  d3.csv("link", rowConverter, function (error, data) {
    if (error) {
      console.log(error);
    } else {
      console.log(data);}});}
// Components: svg (main frame), rect (bar chart), circle (scatter plot), text
(label), g (axis)
// Functions: slice(), filter(), forEach()
// Scale: sclaeOrdinal, scaleLinear, scaleLog
```

```
// Attributes: width, height, x/cx, y/cy, r, fill, fill-opacity, class, text-anchor, font-size
// Update vs Merge transitions
// Predefined code
    // d3.min/max(currentData, function (d) {
    //       return d["attr1"];
    //     })
    // xScale .range([padding, w - padding]) for linear, xScale .rangeRound([0, w - p]) for ordinal
    // yScale .range([h - padding, height]) for linear, yScale .rangeRound([0, h - p]) for ordinal
    // rScale .range(2,5)
    // opacity .range([0.1,1])
    // vertical bar chart: x: xScaleOrdinal(i), y: height - p, width: xScale.bandwidth(), height: yScale(data)
    // horizontal bar chart: x: padding, y: yScaleOrdinal(i), height: yScale.bandwidth(), width: xScale(data)
    // position .attr("text-anchor", ".start mid.dle end.")
    // normal scale return scale(attr1)
    // ordinal scale return scale(i) + scale.bandwidth()/1.5
    // position bottom/top/left/right
    // class name x axis, y axis
    // xScale .attr("transform", "translate(0," + (h - p) + ")")
    // yScale .attr("transform", "translate(" + p + "," + (h - p) + ")")
    // ticks .tickValues([0, 100, 250, 600]);
    // formatted tick .tickFormat(d3.format(".1%"))
    // position .attr("text-anchor", ".start mid.dle end.")
    // xAxis x: w/2, y: h - p * 0.7
    // yAxis x: -h/2, y: p/2
    // yAxis .attr("transform", "rotate(-90)");
```