**Feb 4th, 2019**
**590 DV**
**Lecture 3 Notetaking**

Please read the note respectively with the lecture slides.

**Warm-up activity:**

Interactive 35 yrs of American death visualization
The heat map shows a trend a decrease in death
Method:
Census to collect the data, adjusted to the older population and geographic variation, collected from death certificate(vague), Institute for health metrics and evaluation.
Room for improvement, negative and positive:
Interactive, play button
Informative data for each county
Difficult to drill down to the type of death, specific reason why a region is hot in the heat map
Recall two weeks ago, Geographical map distort your sense of population, assuming larger county has the same weight as east count, unfair towards the west
The audience can only read before understanding the visualization
Popular Data journalism, the 538 website
County by county data that's too much comparing to state to state
Provide a way to Seek out information on a specific area

**Today's topic:**

How drawing on computer works?
Bits and bytes
More than 90% of modern displays are driven by a grid of pixels
Tv HD resolution, Collums and rows of pixels
Red green light/ pixel
Smooth picture are all pixels putting together

Vector display graphics, drawing lines between endpoints
Starwar arcade, float values storing in computer
Why did the vector display go away? Investigate by drawing a line, check slides:
Fill in the pixels the line touches, unreasonable representation since it's not exactly horizontal
Lines got curves and direction
Increase the resolution to 16X16,3232, starting to look like a line but more data, waste of computer storage
A grid of data to a grid of screen pixels not always 1 to 1 representation

*Raster data:* an image file on a computer is raster.
Display of raster data
Every pixel has a color, GIF JPG PNG

Very basic Compression technique, store the adjacent white pixels only once

Raster codec, lossy compression degrade your image through multiple encodings
Example: unloading an mp4 to youtube multiple times
Raster data is not always an image, volume pixel

*Vector data:*
Every component defines as drawing component
A component has attributes like color, common format SVG PDF EPS
Render a list of location to the monitor, one to one mapping

Raster is less memory than vector if a line covers fewer than 40 pixels:
Vector, X0,y0,x1,y1,width, 5bytes, 40bits
Raster, 1bit
Vector becomes efficient if we increase the resolution with more pixels
Zoomed in, outside information discarded, raster becomes valuable again

Tables showing comparative data sizes:
No matter how many lines are dumping into the image, raster representation size stays the same
Fixed size of 5mb for vector data regardless of the pixels
Within 1 million lines, a lot cheaper to store in raster

Circle: takes up less storage than a line, floating values of center coordinates and radius
Similar data table

“Fractal”
Image raster of the vector data
“Menger sponge”
25.6million points vs 921600 pixels

Hard to represent nature in vector representation
Nature resolved in atoms, no way to store those infinite atoms

*Text:* most texts are stored as vector data, constantly resizing our text
Rendering engine and text font, vector requires a rendering process
Fontscript.ipynb test it on your own (download from course website)

*Geographical Data:*
1.State Boundaries, vector
2.height map, terrain map, raster
3.population density, raster
4.capitol city, vector

*Mutation*, take our data change into the format of the data we want to work with
Mathematical operations
Smoothing, create the more useful shape of the data without dumping

*Histogram aggregating*
Figuring out the bins of histograms
Creating a grid into 2-D
Splitting into meaning buckets
(Vertical bins of number of circles, just counting the numbers, or looking at the sum, even average)

*Splitting operations:* Characteristic we use to split the data

**Python basics reviews from previous weeks**

**Code demonstration will be posted on the website along with assignment number 2**