COLOR PALETTE GENERATION FOR NOMINAL ENCODINGS

Ketrina Yim, Calvin Ardi, Simon Tan
CS 294-10 | Fall 2008
Problem & Motivation

● Color serves many functions in visualization
  ○ Highlighting certain values
  ○ Creating groupings
  ○ Facilitating data layering
  ○ Improving legibility

● Choosing effective colors can be difficult
  ○ Controlling value
  ○ Maintaining contrast
  ○ Aesthetics

● Color palettes in current visualization software
  ○ User Created - May be overwhelming for beginners
  ○ Presets - Offers little opportunity to customize
Related Work

- **Color Brewer, Colour Scheme Chooser**
- Brath et al., "Visualization for Communication: The Importance of Aesthetic Sizzle"
- Tominski et al., "Task-Driven Color Coding"
- Ventura and Schettini, "Computer-aided Color Coding for Data Display"
- Shulze-Wollgast et al., "Enhancing Visual Exploration by Appropriate Color Coding"
- Campadelli et al., "Color set selection for nominal coding by Hopfield networks"
Related Work

[Schulze-Wollgast et al. 2]
Related Work

[Brewer]
Related Work

US EPA Regional Oxidant Model -- Midwest
Ozone (ppbv): June 26, 1987, 18:00
Related Work

[Bergman et al.]
Related Work

[Rheinigans]

Figure 1. Grey scale.

Figure 2. Saturation scale.

Figure 3. Spectrum scale.

Figure 4. Limited spectrum scale.
Main Challenge: How can we produce palettes with minimal human intervention?

Additional Technical Challenges
- Devising heuristic function to evaluate a palette's effectiveness
- Runtime efficiency
- Developing an extension to prefuse / flare
Similar to Agrawala's simulated annealing for pie-chart labeling
  o Use simulated annealing to generate color palettes iteratively

Use Brewer's Color Use Guidelines for Data Representation to dictate heuristics
  o Which pairs of colors work together?
  o Contrast rules, etc...
  o Validity measured using Brewer's palettes as the epitome of good palette design

User specifications should also affect heuristics
  o Request a colorblind-friendly palette
  o Request a palette with color harmony
Milestones

● M1: Heuristic function for evaluating palettes

● M2: The palette-generating algorithm

● M3: Building interface to display palettes and take user input
Most of the work will be done together, but each member will focus on a particular element:

- Ketrina: Interface
- Simon: Simulated Annealing algorithm
- Calvin: Heuristic function
Thank You

Questions?