Uncertainty for the Novice

Fuzz: A Visualization Toolkit

Abigail Daken
Center for Integrative Environmental Research
University of Maryland
2202 Van Munching Hall
College Park, Maryland
aadaken@umd.edu

Eylul Dogruel
Computer Science Department University of Maryland
A.V. Williams Building
College Park, Maryland
eylul@cs.umd.edu

Justin Grimes
Center for Information Policy and E-Government
University of Maryland
2118G Hornbake Buidling
College Park, Maryland
jgrimes2@umd.edu

Michael Lam
Computer Science Department University of Maryland
A.V. Williams Building
College Park, Maryland
lam@cs.umd.edu

Thomas Lotze
Applied Mathematics Department
University of Maryland
Mathematics Building
College Park, Maryland
lotze@math.umd.edu

ABSTRACT

Scientific visualizations often include uncertainty, as do visualizations meant for other specialized users, but there is currently no standard software toolkit for visualizing uncertainty. Furthermore, previous studies of user reactions to visualization of uncertain data have focused on users familiar with uncertainty. Here we present Fuzz, a toolkit to visualize uncertain data in several types of visualizations. These tools are intended to be intuitive and understandable to inexperienced users, and to help such users understand uncertainty. An additional interactive element, the confidence level slider, is introduced as well. The results of a pilot user study with non-scientists is also presented. The idea for this project came from State of the USA, a non-profit developing a website allowing users to access and explore reliable non-partisan data defining the state of the nation.

1. INTRODUCTION

Much of the recent work in visualizing uncertainty has focused on uncertainty in map based displays, screen representations of 3D objects, and networks and trees with uncertain structure. Each of these research areas is driven by burgeoning areas with specialized users: GIS systems, 3D medical imaging and rendering, terrorist tracking, and study of social networks. They are also driven by the engaging and challenging nature of the problems.

Researchers have thoroughly categorized and explored options for representing data uncertainty on 1D or 2D graphs, for instance [18, 8]. Scientists and others who deal with uncertainty regularly use many of these techniques as a matter of course. However, the reaction of users who are not familiar with the concept of data uncertainty to these techniques has not been well studied. These users are novices in the sense that they are not familiar with uncertainty and its visual representation, and in fact may not use data visualizations of any kind regularly. Many of them are specialists in other areas.

The spread of web access makes a wide variety of data available to everyone. This presents an opportunity for better informed decision making, but to exploit the opportunity fully, the uncertainty of that data must be communicated even to novices. Visualization is a powerful tool to understand data, and provides the opportunity to help novice understand its uncertainty as well. Combining data visualization with user interaction broadens this opportunity considerably. In the process, users may develop a better understanding of uncertainty.

This project was developed in collaboration with the non-profit organization State of the USA [7]. Their mission is to make reliable nonpartisan national metrics data widely available to local and national policy makers, journalists, the well informed public, and ultimately students at every educational level. To make this effort more specific, a focus on visualizing uncertainty for local policy makers was selected. The result, Fuzz, is a Java toolkit to assist developers in representing uncertain data for novices.

Fuzz provides two visualizations of data, each with two methods to visualize uncertainty. Screen captures of each of these options are shown in following sections. For bar charts, uncertainty can be represented either with standard error bars or by fading out the top of the bar. For time series visualizations, uncertainty can be represented by bars for each data point covering the confidence interval for that point, or by a region encompassing the confidence interval for each point. Time series data can also be used to forecast future values. Fuzz shows the uncertainty of forecast values using the same methods as for time series data.

A number of controls are available for each visualization. Most importantly, users control the displayed confidence intervals with a confidence level slider. This allows the visualization to provide the maximum amount of information with the least visual clutter. Users can also adjust the data range shown, whether grid lines are displayed, and the visibility and thickness of the line connecting the data points.

Users can also control which data are displayed on the visualization. The plot quickly becomes cluttered when viewing multiple data series with overlapping confidence intervals. To address this, users have the option of showing each data series on a separate set of axes, with the axes coordinated between plots for easy comparison.

Section 2 reviews published work on uncertainty and its visualization. Section 3 discusses the design process and design trade offs for Fuzz. Section 4 reviews the mathematic background for statistical uncertainty. Section 5 lays out the implementation of these principles in the toolkit. A pilot user study with reactions from graduate students of public policy and other fields is described in section 6, along with the results. Section 7 suggests directions for future development of Fuzz.

2. RELATED WORK

2.1 Sources and Types of Uncertainty

Uncertainty can enter a data set at any point between its collection and its final visualization [15, 13]. At the collection step, instruments measure physical quantities to a particular precision and accuracy, and characteristics of populations generally come from sampling a subset of the population. Measured data is then often transformed, for instance through interpolation or subsampling, or through combination with other data as when quantities are calculated on a per capita basis. Lastly, the visualization of the data may itself introduce uncertainty, since rendering algorithms generally trade off speed and accuracy [12]. Even a simple 2D time series chart usually includes a line between points, which is effectively a linear interpolation.

The origin of the uncertainty is just one way to classify it. Meredith Skeels et al present a classification system designed to aid visualization [17]. They base their system on previous classifications and on in-depth discussions with experienced users of uncertainty. They identify uncertainty at three levels: measurement precision, completeness, and inference. Completeness includes missing values, sampling and aggregation. Inferences include predictions, modeling and projection into the past. They also identify two types of uncertainty that span these levels: the credibility of the source and the level of disagreement, which sometimes also affects credibility. Thomson [18] presents a review of other uncertainty classifications.

While all these types of uncertainty are important, our work deals only with numerical uncertainty, specifically in statistical data. Many of the national metrics that State of the USA is interested in are collected from survey data. Such data have completeness uncertainty according to Skeels' classification. The addition of forecasting leads to inference uncertainty. Many of the user study participants were public policy students who clearly understand source credibility, and routinely take it into account when examining data.

2.2 Visualizing Uncertainty

While most of the literature focuses on displaying uncertainty for users who are not novices, many of the design principles are similar. For instance, Andrej Cedilnik and Penny Rheingans [5] present four excellent principles for the design of uncertainty displays:

1. Uncertainty should not overshadow the data values.
2. Uncertainty should be "perceptually normalized", with the same amount of visual "energy" present at every place.
3. Uncertainty representations should be inherently meaningful, having some intuitive way of presenting the uncertainty.
4. The generation of uncertainty visualization should not slow down the visualization.

They also show a nice method for visualizing uncertainty for geographic data that may be applicable to other kinds of data displays, though it is not currently part of Fuzz.

There are a limited number of methods available for showing uncertainty [13]:

1. Add glyphs, e.g. error bars.
2. Modify glyphs, e.g. extend scatterplot points to cover confidence interval.
3. Add geometry, e.g. add surfaces representing the confidence interval around rendered surfaces.
4. Modify geometry, e.g. make uncertain surfaces bumpier or shinier.
5. Animation, e.g. more uncertain data blinks off more often or for more time, or jiggles more.
6. Sonification, e.g. mousing over a data point gives a pitch proportional to its uncertainty.

One might be able to imagine additional options, such as haptification, if one has the appropriate output devices available. In keeping with the nature of the intended audience of Fuzz, these options were not explored.

As mentioned above, much recent work has focused on geographic information and 3D rendering. Older references such as [20] give examples of uncertainty representations specifically for 1D and 2D visualizations. Unfortunately, they were even more strongly focused on expert users than recent work.

3. DESIGN

Because of its focus on users who are novices at using visualizations, Fuzz is limited to relatively simple and familiar visualizations: time series plots and bar charts. Trend forecasting is included because it is important in the policy arena. The overall design goal for Fuzz was to find ways to visualize uncertain data that balance familiarity and cognitive effectiveness.

For any user, it is clearly important to represent uncertainty in such a way that data with high uncertainty is intuitively perceived as uncertain. In addition, the targeted users are consumers of data and not data producers. Therefore, more certain data should draw the attention of users from less certain data. The users should base any conclusions or cognitive models drawn from the data more heavily on the more certain data. In contrast, for a researcher attempting to eliminate areas of uncertainty, it may make sense to emphasize uncertain data.

Familiarity and viewing comfort were also important in selecting visualizations. Fuzz is intended to encourage novice users to take uncertainty into account when they look at data, which means that user preference is important. If users are not comfortable with the tools provided, they will simply view the data another way, one less likely to show them uncertainty.

3.1 Participatory Design

Designing for novice users calls for a participatory design approach [16]. In this case, a public policy graduate student volunteered to provide feedback early in the design of the toolkit. During the design process, this interim user provided essential feedback to the creative process. This helped focus the effort on what worked in the design and on the questions users might seek to answer with such visualizations. The user provided input at the conceptual, paper prototype, and alpha version stages of Fuzz. For example, the early paper prototypes included displaying uncertainty using choropleth maps with mixed uncertainty techniques including hatching. The user expressed concerns over the usefulness and success of such a method, and the effort was dropped.

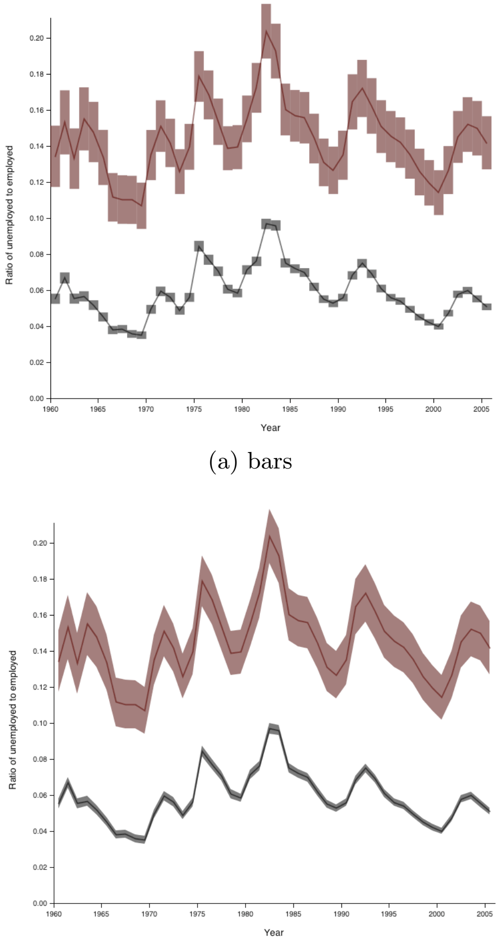
3.2 Paper Prototyping

Since this project was focused on visualizations rather than software, the paper prototyping stage of the design process was somewhat unusual, focusing on impressions of different possible ways of displaying uncertainty, rather than software controls and interactions. Drawings and images of alternative visualizations of several mockup data sets were created to show different features of the visualization. These images were discussed with the interim user, focusing on their effectiveness in relaying data and uncertainty. Some of the interface controls were integrated into the layout at this stage but they were not evaluated. Instead, after the elimination of several visualizations and based on the feedback, the controls and the other elements of the toolkit were built around the visualizations and tested during the alpha version phase.

The most important lesson learned from this stage was that users expect to be able to see the values, or at least the ranges of the uncertainty clearly. In addition, data glyphs with gradient transparency seemed to be distracting on time series displays, though they were not a problem in categorical bar charts.

3.3 Visualization Details

Uncertainty in time series charts is represented in two ways. The first method is using an extended bar, covering the confidence interval, to represent each point (see Figure 1(a)). The second is using a region around the line (see Figure 1(b)). The confidence level is selected with a slider, which allows users a way to play with uncertainty and increase their understanding. Griethe at al assert that fuzziness is easily recognized as uncertainty, but present no evidence [8]. It would be interesting to test this assertion for bar charts or time series, though it may also be important to preserve the ability of the location of the edges to communicate the confidence limits.

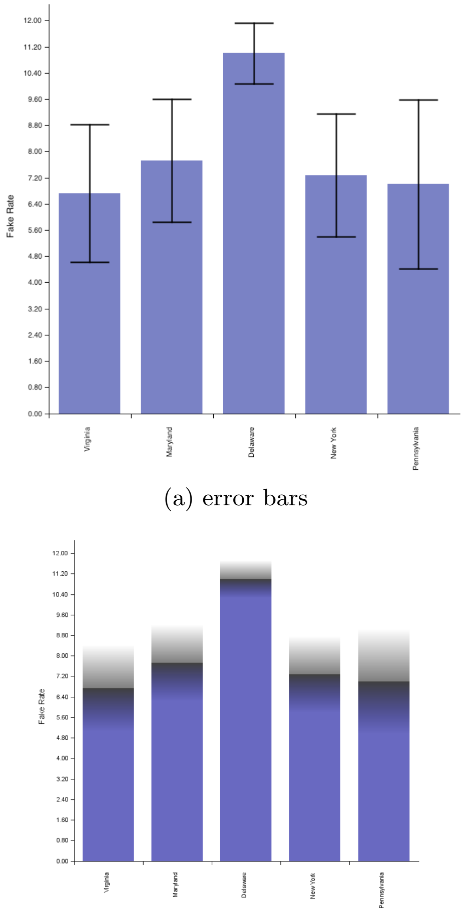


(a) bars

(b) regions

Figure 1: Uncertainty visualization for time series. (a) Bars are more useful for reading off the confidence limits. (b) Regions are attractive to users.

On bar charts, Fuzz provides options for uncertainty to be represented with standard error bars or with a fading glyph overlaid on the top of the bar (see Figures 2(a) and 2(b)). It seemed that fading would be more intuitive for novices than



(a) error bars

(b) fading

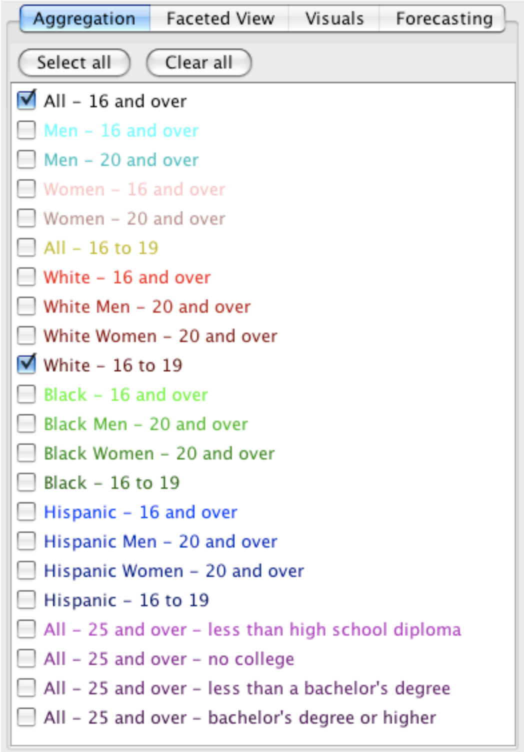
Figure 2: Uncertainty visualization for bar charts were not tested in the pilot user study. (a) Standard error bars may be more familiar. (b) Fading in uncertainty region may intuitively seem like uncertainty.

the traditional error bars; this assumption will be tested in a follow up study. Wainer [20] also shows a confidence aperture plot, as well as a number of methods to show more than one level of confidence at once. It seemed that a single confidence interval would be far less confusing. The interactivity of a computer graphic, as opposed to a printed one, allows similar functionality with the confidence level slider.

The upper and lower limits of the confidence bound are difficult to see in the fading bar representation. One alternative is a glyph which covers the confidence interval but does not intersect the category (horizontal) axis. This is similar to one of the options for the time series plot. However, discussions with the interim user confirmed our supposition that the familiar bar chart might be preferable for categorical data with no intrinsic ordering.

A fading-edge bar for time series charts was considered, but feedback on the paper prototype indicated that it was unpleasant to view. This may have been because finding the edge of the glyph is difficult and draws perceptual attention; if so, it would strike a user much like clutter, as well as drawing undue attention to less certain points. It would be interesting to see if there were circumstances where this technique works well, perhaps with more experienced users.

Displaying multiple time series with uncertainty at once often leads to occlusion and visual clutter. The "display on separate axes" feature was created to addresses this problem. It is based on Edward Tufte's principle of small multiples[19]. The small multiples concept refers to a series of similar pictures, usually visualizations of quantitive data. Through repetition, the series helps the user to understand patterns and differences in the data. Each aggregation is separately graphed on its own axis and arranged in consecutive order on a grid layout. For easy comparison, the axes have the same scale and extent for each aggregation.



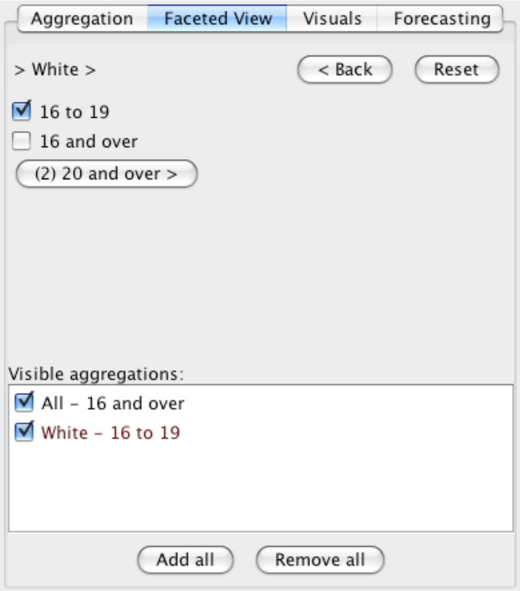
☑ All – 16 and over
☐ Men – 16 and over
☐ Men – 20 and over
☐ Women – 16 and over
☐ Women – 20 and over
☐ All – 16 to 19
☐ White – 16 and over
☐ White Men – 20 and over
☐ White Women – 20 and over
☑ White – 16 to 19
☐ Black – 16 and over
☐ Black Men – 20 and over
☐ Black Women – 20 and over
☐ Black – 16 to 19
☐ Hispanic – 16 and over
☐ Hispanic Men – 20 and over
☐ Hispanic Women – 20 and over
☐ Hispanic – 16 to 19
☐ All – 25 and over – less than high school diploma
☐ All – 25 and over – no college
☐ All – 25 and over – less than a bachelor's degree
☐ All – 25 and over – bachelor's degree or higher

Figure 3: Aggregation panel allows users to select which data to view. Options for selecting data are likely to vary depending on the data set.

3.4 Controls

The visualizations in Fuzz are controlled by two tabbed panels. The visualization panel (Figure 5) controls how uncertainty is displayed, the confidence level, zooming and various graph options. The forecasting panel (Figure 6) controls which forecast to use, how many years of data to use for the forecast, and how far into the future to forecast. It also gives information about the quality of fit for each forecasting option (see section 4).

Separate panels can make controls hard to find, but they also maximize use of screen space for the visualization itself. The interim user gave us valuable feedback about the layout



☑ 16 to 19
□ 16 and over

Visible aggregations:

☑ All – 16 and over
☑ White – 16 to 19

Figure 4: For many national indicators, data can be subdivided by several categories. The faceted aggregation panel allows users to see the structure of the data set when selecting which data to view. Options for selecting data are likely to vary depending on the data set.

of various functions at the alpha prototype phase.

In the sample implementation presented here, two other tabbed panels (see Figures 3 and 4) control which data are displayed. This functionality is likely to vary depending on the nature of the data set in question.

4. STATISTICAL BACKGROUND

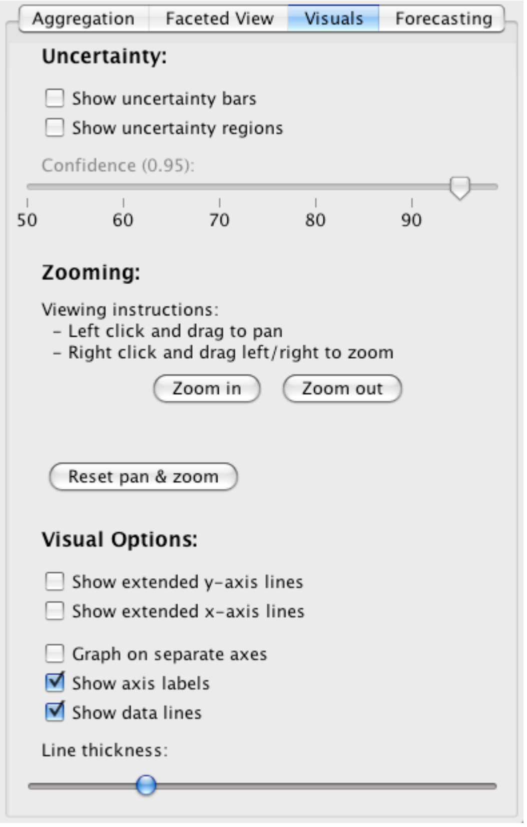
4.1 Numerical Uncertainty

Statistical uncertainty is that based on measurement error due to sampling from a population. Since the measurement deals with the proportion of a population, a Wald interval was used to create confidence intervals for each value, using a Gaussian normal distribution with standard error $\sqrt{\hat{p}(1-\hat{p})/n}$, where $\hat{p}$ is the estimated proportion (e.g., the proportion of unemployed) and n is the number of people in the sample.

Since there are multiple series being considered, each with multiple data points, we must also consider that the user is performing multiple testing, considering each point on each series as a test [20]. Thus, the confidence intervals are modified by a Bonferroni correction to account for the km tests, where m is the number of series and k is the number of data points in each series. This correction is known to be conservative, and alternatives are discussed in Section 7.

4.2 Forecasting

There are four types of forecasts provided: linear, quadratic, exponential, and logarithmic (see Figure 7). All use basic least-squares regression [14] to model the data as a function of time. Linear regression forecasts the data y using time x as a predictor, estimating coefficients in the equation $y = a + bx + \epsilon$. Quadratic regression uses the model $y = a + bx + cx^2 + \epsilon$. Exponential regression uses the model $y = ae^{bx}\epsilon$. Logarithmic regression uses the model $y = a + bln(x) + \epsilon$.



Uncertainty:

□ Show uncertainty bars
□ Show uncertainty regions

Confidence (0.95):

Zooming:

Viewing instructions:
- Left click and drag to pan
- Right click and drag left/right to zoom

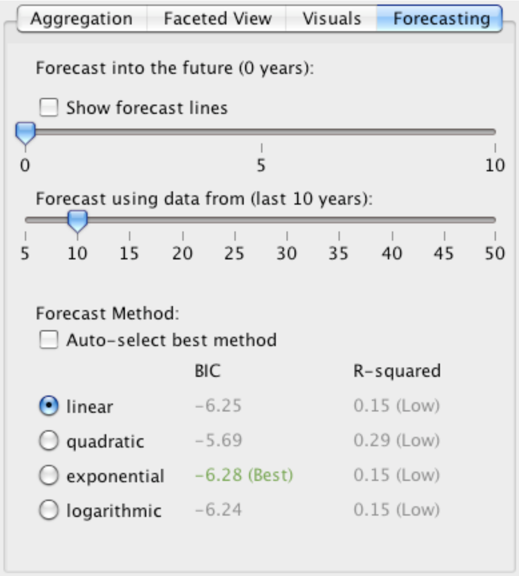
Visual Options:

□ Show extended y-axis lines
□ Show extended x-axis lines

□ Graph on separate axes
☑ Show axis labels
☑ Show data lines

Line thickness:

Figure 5: Users control the way information is displayed using the visual options panel.



Forecast into the future (0 years):

□ Show forecast lines

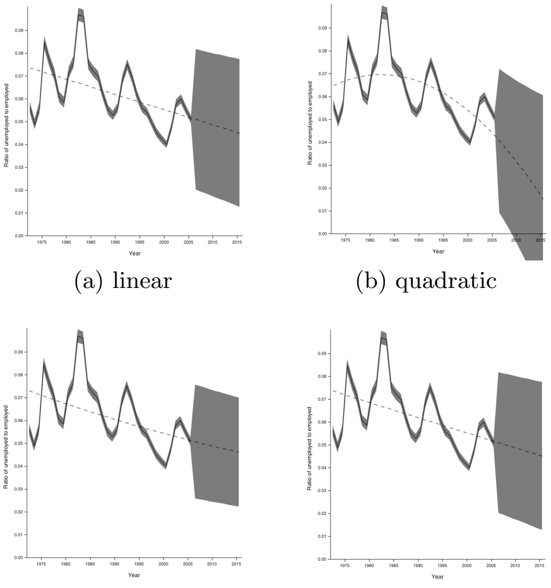
Forecast using data from (last 10 years):

Forecast Method:

□ Auto-select best method

	BIC	R-squared
● linear	-6.25	0.15 (Low)
○ quadratic	-5.69	0.29 (Low)
○ exponential	-6.28 (Best)	0.15 (Low)
○ logarithmic	-6.24	0.15 (Low)

Figure 6: Users control forecasting using the forecasting options panel.



(a) linear

(b) quadratic

(c) exponential

(d) logarithmic

Figure 7: Forecast uncertainty for time series: (a) linear, (b) quadratic, (c) exponential, and (d) logarithmic. Uncertainty is shown with regions in this example, but bars also work with forecasting. The large uncertainty of the forecast reflects the fact that this particular data is poorly predicted by all of the models because it depends more on other variables than on time.

Each of these models is ranked using the Bayesian Information Criterion (BIC). BIC is a measure of the relative worth of the model, and is calculated as

$$BIC = nln(\frac{RSS}{n}) + kln(n) \qquad (1)$$

where n is the number of observations, RSS is the residual sum of squares of the model, and k is the number of parameters used in the model. The goodness of fit (RSS) is thus balanced with the complexity of the model (k) in such a way that as n increases, the model with the lowest BIC is the model with the correct form. It is commonly used as a way of comparing models and choosing which to use.

We also provide a calculation of R^2. R^2 is the proportion of variance in the data which is explained by the model, and is calculated as

$$R^2 = 1 - \frac{SS_{err}}{SS_{tot}} \qquad (2)$$

where $SS_{err} = \sum_i (y_i - \hat{y}_i)^2$ and $SS_{tot} = \sum_i (y_i - \bar{y})^2$. For more details on the calculation of R^2, see [14]. It is often used as a general guide for how useful a model is for explaining the data. An R^2 value less than 0.7 is generally considered to not be very useful, while one greater than 0.85 is generally considered to be very useful. R^2 should not be used as the measure to compare between models, however, as more complex models will always have a better R^2, regardless of whether the additional complexity accurately captures the model variance. For this reason, the BIC is used to compare models and the R^2 is provided as a guide for usefulness. For instance, in Figure 6, the quadratic model has the lowest BIC and is therefore the best of these four. However, the R^2 is only .72, indicating that even this is not a particularly good model of the data.

We provide uncertainty limits for the forecasts using the standard normal form for linear regression,

$$P(y_0 \in x_0'\hat{\beta} \pm t_{\alpha/2,n-k-1} s\sqrt{1 + x_0'(X'X)^{-1}x_0}) \geq \alpha \qquad (3)$$

where s is the estimated standard error of ϵ from the regression, n is the number of values for training, k is the number of parameters in the model, and x_0 is the data for which the forecast is made. For more details, see Rencher [14]. A Bonferroni correction accounts for multiple testing of several forecasts. Improved methods for modeling the confidence intervals for forecasts are discussed in Section 7.

5. IMPLEMENTATION

The Fuzz toolkit and sample visualizations are built in Java. Java has several advantages: 1) it has a strong following among the software development community, 2) there are several strong precedents of Java-based visualization projects [21, 10], and 3) Java toolkits for visualization and statistics are available [1, 9, 11, 3]. The AWT/Swing, Piccolo [1], and Prefuse [9] libraries were used to expedite the interface development, and the Jama [11] and JSC [3] libraries were used for statistical routines.

In order to maximize the potential for code reuse and future extension, we strove to achieve maximum separation between generic visualization code and the needs of our specific data sets. The effective result is a strong distinction between graphics and data structure code components as well as between the various types of visualizations. See Figure 8 for a class diagram overview of all of the code components.

5.1 Graphics

For each type of visualization (currently time series and bar charts), Fuzz provides graph, panel, and axes objects. It also provides objects for all of the other pieces of the different visualization types (ex. points, lines, and regions for time series, or bars for bar charts).

The graph object for each visualization provides generic routines for the layout and display of that kind of visualization, and extends the PCanvas object from the Piccolo toolkit. The panel object for each visualization wraps the corresponding graph object in an AWT/Swing panel with an accompanying control panel with options appropriate to the visualization. The panel object contains most of the interface callbacks and uses methods from the graph object to update the visualization (see Figures 5 and 6 for screen shots of the various tabs of the options panel). The axes object for each visualization handles the display of graphical axes and labels around the visualizations.

Each visualization type also has a representative container class. For time series visualizations, there is a time series object that serves as a container for collections of point, line, and region objects. This reflects the multiple layers of abstraction inherent in a visualization; a graph is composed

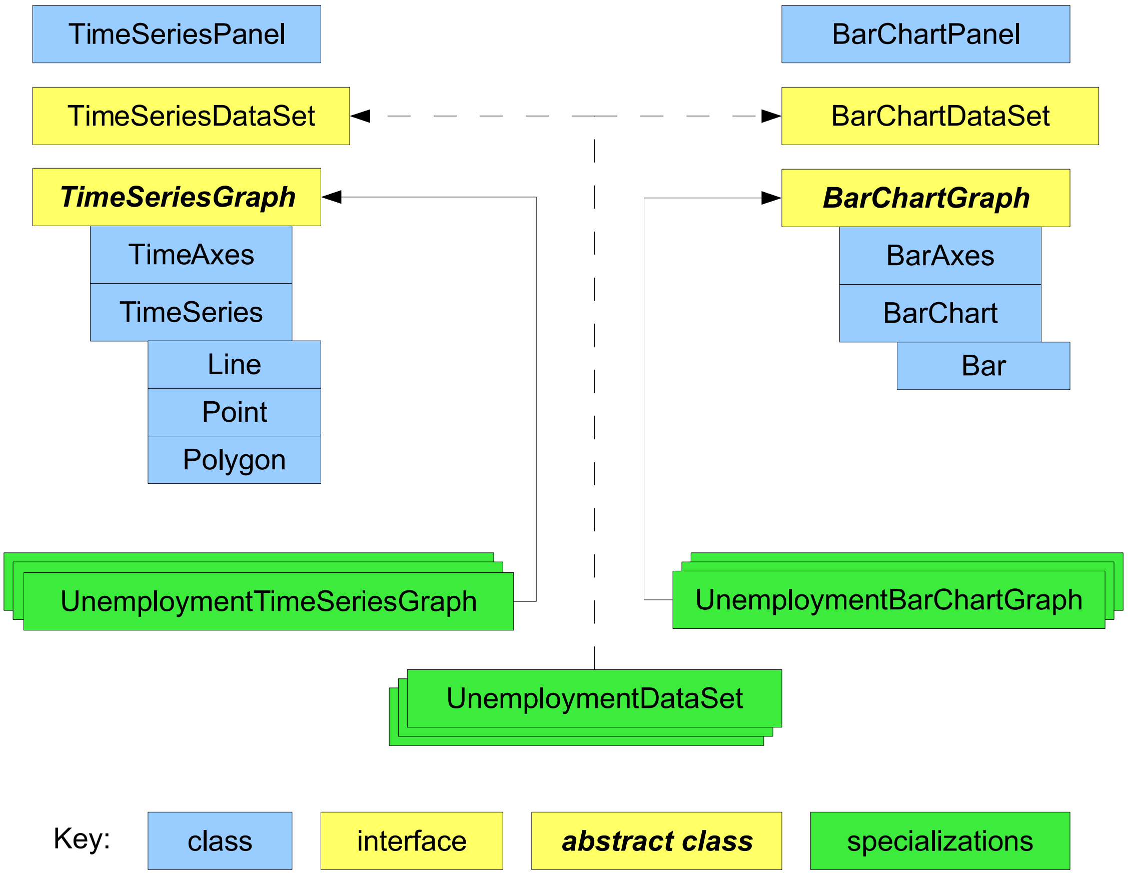


Figure 8: The structure of the code library provides clear separation between generic objects (blue) and dataset-specific objects (green) via well-defined interfaces and abstract classes (yellow), as well as between the visualization types: time series (left) vs. bar charts (right)

of multiple time series, and each time series is further composed of many individual graphical elements. For bar charts, there is no intermediate abstraction, so the bar graph object directly contains a collection of bar objects. That bar class handles all layout and rendering for a single bar. All of the various pieces of a visualization extend the PNode object from the Piccolo toolkit to provide basic panning, zooming, and efficient redrawing.

To add a new type of visualization, the developer must perform the following steps:

1. Write graph component classes that extend the Piccolo PNode object and handle the drawing of individual visualization elements (depending on the type of visualization, it may be helpful to set up container relationships between appropriate classes, as with the time series objects).

2. Write a graph object that 1) extends the Piccolo PCanvas object, 2) contains references to various necessary component objects, and 3) handles the high-level layout of these objects.

3. Decide on a sufficiently generic data structure interface (see Section 5.2) that can handle a wide variety of data sets, and write a specification of any needed methods as a Java interface (data set classes must implement this interface to be used by the new visualization).

5.2 Data Structures

For each visualization (time series, bar chart, etc.), there is a single, inherent, generic data type. For a time series the data type is a mapping from integers (years) to doubles (data), while for a bar chart the data type is a mapping from character strings (nominal categories) to doubles (data). Since the project deals with uncertainty, there are also structures for mapping keys to standard error rates. The visualization objects interact only with these types of data via certain method signatures. These methods are formally defined as Java interfaces. This enabled multiple data set objects to conform to a common interface. These data set objects contain customized import routines and implement the data structure interfaces to allow the rest of the visualization objects to use them.

Unfortunately, it is not completely possible to separate the visualization code from specific data sets because of the data-specific nature of coloring and labeling options, and it was necessary to declare the generic graph objects to be "abstract" (meaning in Java that they cannot be used directly). To create a "concrete" graph object that is useful for a specific data set, the developer must subclass the abstract graph object and fill in the needed information and initialization routines.

To add support for a new data set, the developer must perform the following steps:

1. Write a data set object with code to import the desired data, implementing the appropriate interface methods for all desired visualizations.

2. Write a graph object for each desired visualization type that extends the abstract graph class for that visualization type and specifies all of the necessary coloring and labeling information.

We believe that our implementation provides a simple and standard way to add visualizations to a Java application with minimal effort. Other toolkits have done this in the past, but ours is the first to seamlessly add uncertainty visualization as a standard part of a visualization.

5.3 Integration

After support for the data set is added, the addition of visualizations into a Java application is very simple:

1. Create a panel object of the desired visualization type.

2. Connect the panel to an instance of the desired data set.

3. Add the panel to the application's interface.

We have demonstrated this process by creating several visualizations for various data sets. Our primary example is a time series-based visualization of unemployment data. This example allows the user to explore graphs such as those in Figure 1. The visual and forecasting options for this example are shown in Figures 5 and 6. We used this example for our user evaluation.

6. EVALUATION

To evaluate this toolkit a pilot user evaluation experiment was conducted in the form of a small sample usability test. The goal of this evaluation was to test the effectiveness of the toolkit in describing, exploring, and explaining uncertainty in the visualization of information.

6.1 Methodology

The user evaluation process was semi-structured, with each participant being evaluated individually. The evaluation included a task based analysis and a post evaluation survey.

Fuzz was designed for policy makers who are novice users of visualizations. Thus, recruitment of study participants focused on those that had backgrounds in policy making or policy-related research. Email advertisements were sent to graduate students studying public policy, computer science, information science, and journalism. Five participants were selected from the responses.

Evaluation sessions were conducted either in the participant's work environment or in a convenient conference room. Evaluation sessions lasted approximately thirty minutes. A test laptop with Fuzz loaded on it was provided to the participant to perform the evaluation.

Each evaluation session began with a brief explanation of Fuzz and of the displayed data set. For our evaluation session, users were asked to explore time series data on unemployment from the Bureau of Labor Statistics. This data set shows various nationwide unemployment rates by sex, race and age. The data set was selected for relevance and because its uncertainty is statistical, directly related to sample size.

After a few minutes to explore Fuzz, participants completed four tasks designed to emulate typical user activities. Participants were encouraged to think aloud when conducting tasks and to ask questions at any time during the evaluation.

Listed below are the tasks that each participant was asked to complete, stated in the form of questions:

1. What is the highest and lowest unemployment rate over the years for women 20+ and when did it occur?

2. In a particular time period, find two aggregations that look like they do not intersect, but their confidence intervals overlap.

3. What is the 70% confidence bounds intervals for White 16-19 for 1997? What is the 50% confidence bounds intervals for Hispanics 16-19 for 2001? Which of the aggregate values appears to have the overall most uncertainty over the total time period?

4. Assuming policies have been consistent since 2000 and will continue through 2010: do you think that unemployment, for Black men 20 and over, will be below 6% in 2010? How confident are you in that answer? Why do you think this is the case?

Each task required the participants to either explore the data set to determine the uncertainty for a particular group, or to construct a hypotheses based on the information displayed. Quantitative data such as time or number of clicks to complete tasks was not collected during the pilot study. It focused instead on understanding participants' reasoning and logic in relation to the features of Fuzz.

Once the tasks were completed, each participant filled out a short questionnaire with generic software usability questions and questions about specific features of Fuzz.

6.2 Results

Most participants were able to successfully complete all four of tasks provided. Several participants required clarification of the questions or explanation of particular terms.

In terms of facilitating sense-making, many participants went well beyond merely answering the task questions. Often participants would commit or pose questions theorizing what the displayed results meant. For example, when one participant was exploring the uncertainty of the data they visually realized that blacks and hispanics contained immense uncertainty in comparison to other aggregations. The participant posed questions to themselves such as “why is that?” and then began to reason an answer from this. This reflects a growing understanding of the relationship between statistical uncertainty and sample size.

Participants were split three to two about whether uncertainty regions or uncertainty bars were more effective for displaying uncertainty. During the evaluation participants used the bar display to determine exact confidence interval values for particular data points. They used regions for an appealing general overview of uncertainty ranges. One of the participants specifically mentioned that “bars are much more helpful for guessing confidence intervals” but then quizzically followed that by mentioning later that “bars freak me out a little”. When encouraged to elaborate the participant responded that it was perhaps merely a personal choice.

Results from testing reaffirmed the potential advantage of using small multiples with the “graph on separate axes” feature. In the version of Fuzz used for the evaluation, the “graph on separate axis” feature was called “grid view.” The change in terminology was a direct result of the user evaluation. Participants had difficulty independently discovering and understanding the feature, because of the name and because of placement of the control on the screen. Participants did find the feature particularly useful to display multiple time series. Once participants were shown the feature, encouraging responses included “cool, that makes it so much easier to see” and “wow, that’s just what I was looking for.” There is a problem with readability of labels and axes as the number of graphs increases.

The zooming feature was quite problematic in use. Due to poor design choices, most participants failed to discover the feature at all. During the course of the evaluation most participants eventually commented on their desire to zoom. Even when shown the zooming tool, almost all participants continued to struggle with the mouse based controls. Several of the participants mentioned that they wanted “smart” zooming, i.e. the ability to zoom in on a particular selected region, line or interval rather than simply zooming directly to the center of the visualization. Based on this feedback, better zooming has been implemented, and will be a priority for future development.

Several participants also uncovered bugs in the y axis scaling, which caused certain values to be repeated. The y axis scaling automatically readjusts as time series are added to or removed from the plot. This turned out to confuse participants and make it harder to comparing time series.

The study uncovered one issue that is important to both usability and uncertainty. Often participants would make changes to the confidence slider on the “visuals” tab but forget that setting when moving to other tasks. While it is necessary to split features between tabs to conserve screen space, this problem seemed counterproductive to the goal of uncertainty literacy. This issue was addressed by providing a status bar in the visualization which always displays the confidence setting.

Another problem was with the forecasting feature. Several participants did not discover and use the slider for “number of years to use for forecast.” Participants forgot to increment this value, or confused this slider with the year filtering slider on the visualization tab.

7. FUTURE WORK

User evaluation is one of the critical areas missing from the literature reviewed on visualizing uncertainty. The user study presented here is a small pilot and further evaluation is needed. This more extensive user evaluation experiment would include more participants and more tasks.

The more extensive study would be most useful if a wider variety of uncertainty visualizations were available, as well as more data sets. In particular, the bar chart fading bar technique might be considerably improved. In addition, it may be possible to combine the cognitive mapping of visual fuzziness to uncertainty with the desire for sharp-edged glyphs to indicate the confidence limits. One might, for instance, have glyphs with fuzzy boundaries in one direction and sharp boundaries in the other.

There are a number of improvements which could be made to the statistical analysis performed in the program. In general, the confidence intervals are overly conservative. Instead of using a confidence interval scaled for making all comparisons, the adjustment could be made dynamically based on how many series are displayed. This would result in smaller and more appropriate confidence intervals. In addition, using the False Discovery Rate correction of Bejamini and Hochberg [2] instead of a Bonferroni correction would result in smaller and less conservative confidence intervals.

The confidence intervals (CI) used in the program are normal approximations. However, since most of the data being analyzed will either be proportions or counts, the CIs should be changed to appropriate CIs for Binomial or Poisson data [4, 6]. In particular, the Wald estimator which we use for proportion CI estimation is known to be chaotic in its coverage and should be replaced.

Only four simple regression models are included in the program. Additional models, including generalized linear regression or seasonal models, would be very good to have included for comparison. Also, allowing for models using multiple variables as predictors would be very useful for judging more complicated policy questions.

In order to be a better general-purpose toolkit for displaying uncertainty, the code should be generalized to import any data set and display two columns as X and Y axes. This poses additional challenges for representing uncertainty, as there may now be uncertainty in both the x and y coordinate of the data. However, this would allow the toolkit to support scattergrams, a very widely-used kind of visualization.

Lastly, data relevant to public policy is typically incomplete. For instance, some years may be missing, or the definitions of survey categories may have changed over time. The national census data is a classic example of the latter. Techniques to visualize such uncertainty would be very valuable.

8. CONCLUSIONS

This work is an effort to determine how best to communicate about uncertainty with non-expert users of visualizations. Interactive information visualizations hold great potential to represent uncertainty intuitively and teach novice users to account for it in drawing conclusions. Participants in the pilot user study easily understood the displayed uncertainty, and gave us valuable feedback about the user interface and controls. Bars with fading edges were distracting on time series plots, and while users liked the look of uncertainty regions around line plots, they were more likely to use a bar representation to answer detailed questions. The confidence level slider allows users to explore uncertainty while maintaining a simple visual display.

9. ACKNOWLEDGMENTS

We would like to thank Scott Gilkeson of State of the USA, our interim user Jeananne Sciabarra, and all our user evaluation participants for their helpful comments at various design stages. Thanks to Inbal Yahav, Ryan Blue, Kyle King, Adam Fuchs, and Ken Knudsen, who reviewed an early draft of this paper. And a special thanks to Dr. Ben Shienderman for his continuous advice during the project.

10. REFERENCES

[1] Bederson, B. Piccolo - a structured 2d graphics framework. http://www.cs.umd.edu/hcil/jazz/. Accessed 6 May 2008.

[2] Benjamini, Y., and Hochberg, Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society B 57* (1995), 289–300.

[3] Bertie, A. Java statistical classes. http://www.jsc.nildram.co.uk/index.htm. Accessed 6 May 2008.

[4] Brown, L. D., Cai, T. T., and DasGupta, A. Interval estimation for a binomial proportion. *Statistical Science 16(2)* (2001), 101–133.

[5] Cedilnik, A., and Rheingans, P. Procedural annotation of uncertain information. In *VIS '00: Proceedings of the conference on Visualization '00* (Los Alamitos, CA, USA, 2000), IEEE Computer Society Press, pp. 77–83.

[6] Dobson, A. J., Kuulasmaa, K., Eberle, E., and Scherer, J. Confidence intervals for weighted sums of poisson parameters. *Statistics in Medicine 10(3)* (1991), 457–462.

[7] Gilkeson, S. State of the usa. http://stateoftheusa.org/. Accessed 6 May 2008.

[8] Griethe, H., and Schumann, H. The visualization of uncertain data: Methods and problems. In *Proceedings SimVis'06* (march 2006).

[9] Heer, J. Prefuse information visualization toolkit. http://prefuse.org/. Accessed 6 May 2008.

[10] Heer, J., and Boyd, D. Vizster - visualizing online social networks. http://jheer.org/vizster/. Accessed 6 May 2008.

[11] Hicklin, J., Moler, C., Webb, P., Boisvert, R. F., Miller, B., Pozo, R., and Remington, K. Jama : A java matrix package. http://math.nist.gov/javanumerics/jama/. Accessed 6 May 2008.

[12] Johnson, C., and Sanderson, A. A next step: Visualizing errors and uncertainty. *Computer Graphics and Applications, IEEE 23*, 5 (Sept.-Oct. 2003), 6–10.

[13] Pang, A. T., Wittenbrink, C. M., and Lodha, S. K. Approaches to uncertainty visualization. *The Visual Computer 13(8)* (1997), 370–390.

[14] Rencher, A. C. *Linear Models in Statistics.* John Wiley & Sons, Inc., 2000.

[15] Riveiro, M. Evaluation of uncertainty visualization techniques for information fusion. In *10th International Conference on Information Fusion* (2007).

[16] Schuler, D. *Participatory Design.* L. Erlbaum Associates, Hillsdale, 1993.

[17] Skeels, M., Lee, B., Smith, G., and Robertson, G. Revealing uncertainty for information visualization. Tech. rep., Microsoft Technical Report, 2008.

[18] Thomson, J., Hetzler, B., MacEachren, A., Gahegan, M., and Pavel, M. A typology for visualizing uncertainty. In *Conference on Visualization and Data Analysis* (January 2005), pp. 16–20.

[19] Tufte, E. *The Visual Display of Quantitative Information.* Graphics Press, Cheshire, 2001.

[20] Wainer, H. Depicting error. *The American Statistician 50(2)* (1996), 101–111.

[21] Wattenberg, M., Kriss, J., ViŐgas, F., van Ham, F., and McKeon, M. Ibm many eyes. http://services.alphaworks.ibm.com/manyeyes/home. Accessed 6 May 2008.