Uncertainty Visualization

Matthew Kay

2018-10-16

Contents

Preface

This book[1] aims to provide the *why* and the *how* of uncertainty visualization. The *why* is based on what we currently know about how people interpret and make decisions from uncertainty visualizations, and the *how* is structured around generating uncertainty visualizations using the R programming language and the grammar of graphics.[2]

[1] or what it will be—it is a work in progress

[2] as implemented in the *ggplot2* package (Wickham et al., 2018)

Chapter 1

Introduction

TBD

Chapter 2

Why uncertainty visualization is hard

- Efficient encodings for uncertainty can be hard to find
  - try putting mean, variance, and interval estimation in one plot + doing this when useful channels are already used up
- Even if you get the encoding right have to contend with making sure people understand it
  - explain a CDF to someone (N.B.: sometimes this works! task has an effect here)
- Even if people understand have to deal with how people perceive uncertainty
  - linear-in-log-odds and other perceptual models of probability
- Even if people correctly perceive the uncertainty have to deal with how people make decisions under uncertainty
- Even if you solve all these problems in one context a new context may cause your solution to break
  - see e.g. deterministic construal errors
- Plus, you still have to build the damn thing
  - But some strategies for building uncertainty vis using the grammar of graphics helps a lot here (the **how** of this book)

Chapter 3

Foundations: Uncertainty as we'll see it

- primarily Bayesian perspective
  - unifies aleatory and epistemic
  - gives us one framework to work in
  - simplifies building things
  - simplifies communicating things (see e.g. confidence fallacy)
- will still occasionally touch on frequentist, usually as a subsection at the end of each chapter:
  - tips on how to translate the visualization techniques presented into that mode
  - at times, visualizations unique to the frequentist approach that might be useful

Chapter 4

Foundations: Uncertainty and the grammar of graphics

- basic strategies for applying uncertainty visualization within the grammar of graphics
- visual channels, marks, etc. (and mapping onto ggplot)
- simple prediction grid example

Chapter 5

Foundations: General strategies

Dunno if this goes before or after all the specific examples

- think about task
  - what do people actually need from the representation to do the task? densities? intervals? (arbitrary or known in advance?) point estimates?
- employ frequency framing where possible
  - *(maybe a table here translating common non-frequency representations into frequency representations)*
  - quantile dotplots
  - representative sampling in scatterplots (need a better name for this)
  - spaghettis
  - pixelated maps (also needs a name)
- use animation / HOPs
- anticipate deterministic construal errors
  - think about task context, common elements of that context with plausibly similar abstract representations
  - when animating / using metaphor, think about the mechanism - do we think of it as random?

Chapter 6

Discrete data

- categorical
  - proportions
    - product plots / mosaic plots
  - icon arrays
    - unit visualizations
  - HOPs-like stuff
  - entropy-based approaches
- ordinal / integer
  - proportions
  - histograms
  - CDFs
  - dotplots
  - …

Chapter 7

Continuous data

not sure of the organization here, especially within multivariate…

- univariate
  - densities, gradients
  - CDFs
  - intervals
  - quantile dotplots
  - HOPs
- multivariate
  - the joint case
    - gradients
    - contour plots
  - the conditional case
    - gradients
    - bands
  - generalizing quantile dotplots to multiple dimensions: representative sampling approaches
    - scatterplots
    - spaghetti plots
  - HOPs
- frequentist addendum
  - interpreting in a Bayesian mode: advantages and pitfalls (and what it reveals)
    - *this maybe goes elsewhere*
  - those double-sided p value plot things

Chapter 8

Dealing with multiple uncertainties

- a note on comparisons
  - univariate marginals might bite you…
- aleatory / epistemic (or data space / parameter space)
  - frankenplots maybe (product plots / mosaic plots / unit visualizations)
  - animation
    - NYT needle
    - Galton boards
- mixing discrete and continuous
  - color with densities
  - HOPs
  - some hurricane stuff

Chapter 9

Maps and glyphs

Not sure this is a separate chapter but putting this here for now…

- always remember to ask if you need a map in the first place
- uncertainty glyphs
  - blur, and problems with it
  - problems with glyphs more generally
- bivariate maps
- VSUPs
  - connect back to LLO and/or regularization (I think LLO…)
- pixelation
- HOPs

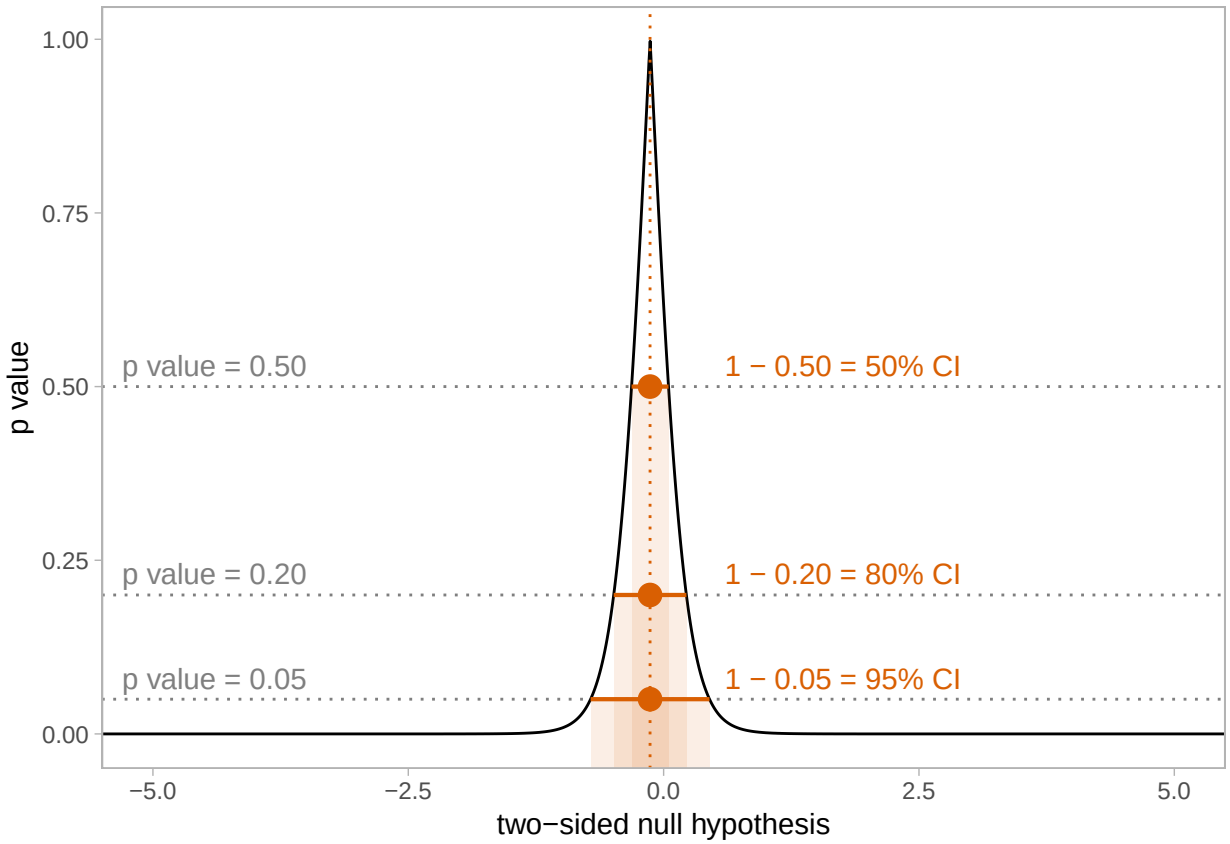
Scratch space

```
library(tidyverse)
library(modelr)
library(broom)
library(ggstance)
```

```
set.seed(12345)
x = rnorm(10)
ci_color = "#d95f02"

intervals = data_frame(
    # conf.level = seq(.7, .99, by = .01),
    conf.level = c(.50, .80, .95),
    alpha = 1 - conf.level,
    test = map(conf.level, ~ tidy(t.test(x, conf.level = ..1)))
  ) %>%
  unnest()

crossing(
    null = seq(-6, 6, length.out = 1001)
  ) %>%
  mutate(test = map(null, ~ tidy(t.test(x, mu = ..1)))) %>%
  unnest() %>%
  ggplot() +
  geom_line(aes(x = null, y = p.value)) +
  geom_hline(aes(yintercept = alpha), color = "gray50", data = intervals, linetype = "dotted") +
  geom_rect(aes(xmin = conf.low, xmax = conf.high, ymax = alpha), ymin = -1, fill = ci_color, alpha = .
    data = intervals, color = NA, size = .75) +
  geom_pointrangeh(aes(xmin = conf.low, xmax = conf.high, x = estimate, y = alpha),
    data = intervals, color = ci_color, size = .75) +
  geom_vline(aes(xintercept = estimate), data = intervals, color = ci_color, linetype = "dotted") +
  geom_text(aes(y = alpha, label = paste0("p value = ", format(alpha, digits = 2, nsmall = 2))),
    x = -5.3, vjust = -.5, hjust = 0, data = intervals, color = "gray50") +
  geom_text(aes(y = alpha, label = paste0("1 - ", format(alpha, digits = 2, nsmall = 2), " = ", round((
    x = 0.6, vjust = -.5, color = ci_color, hjust = 0, data = intervals) +
  theme_light() +
  theme(panel.grid = element_blank()) +
  xlab("two-sided null hypothesis") +
  ylab("p value") +
  coord_cartesian(xlim = c(-5, 5))
```


1.00
0.75
0.50
0.25
0.00
p value
p value = 0.50
p value = 0.20
p value = 0.05
1 − 0.50 = 50% CI
1 − 0.20 = 80% CI
1 − 0.05 = 95% CI
−5.0
−2.5
0.0
2.5
5.0
two−sided null hypothesis

Bibliography

Wickham, H., Chang, W., Henry, L., Pedersen, T. L., Takahashi, K., Wilke, C., and Woo, K. (2018). *ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. R package version 3.0.0.