Linting for Visualization

Towards a Practical Automated Visualization Guidance System

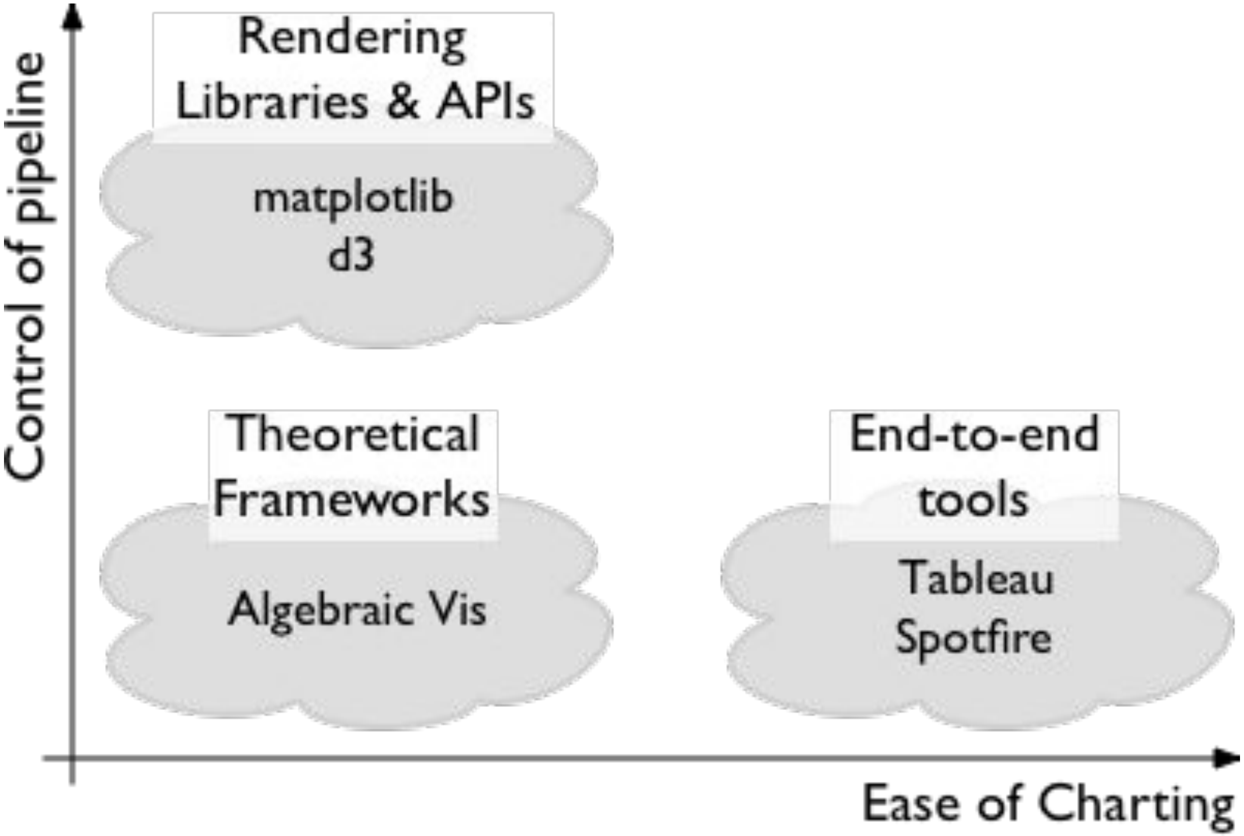
Andrew McNutt & Gordon Kindlmann

Overview/Main Points

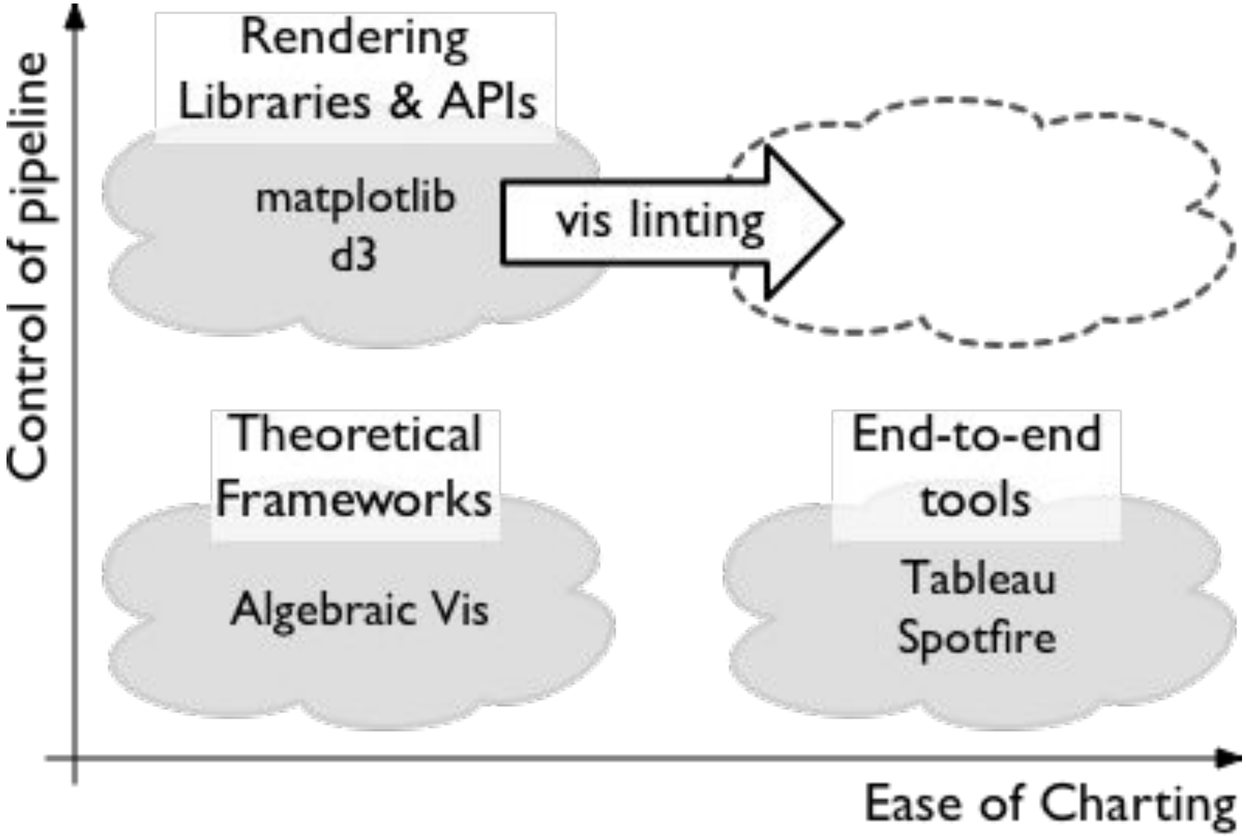
Recommendation systems can be sometimes be too limited, instead consider **guidance systems**.

Linting as a form of analysis based on predefined rules.

Guidance systems can be a springboard for new types of analysis.



Vis software often forces chart makers to sacrifice either configuration or automation



We propose a new route via automated guidance using the language of linting

What Is Linting?

A form of analysis based on a collection of predefined tests.

(In contrast with ad-hoc unit testing)

Like a spell checker for code

Visualization Lint

(or vislint)

Why Linting+Vis?

MECHANIZED INSIGHTS

Automate best practices that are locked away in big lists of guidelines

LEARNING

Helps novice chart makers learn best practices

EXOTIC CHARTS (“XENOGRAPHICS”)

Recommendation systems ignore charts they don’t know about

IGNORABLE & UPDATABLE

The advice given by lint can be programmatically ignored and expanded

Our work: vislint-mpl

Prototype visualization linter in matplotlib (mpl)

- Implements around fifteen rules
- Linting is a good **cultural fit** for mpl

Envisioning linting as springboard for novel types of analysis

- Enable automation of contemporary theory (like algebraic-vis)
- Enable checks that would be difficult for a human (like counting line crossings in a graph or checking for colorblind friendly images)

Examples

Linting can help with adherence to best practices

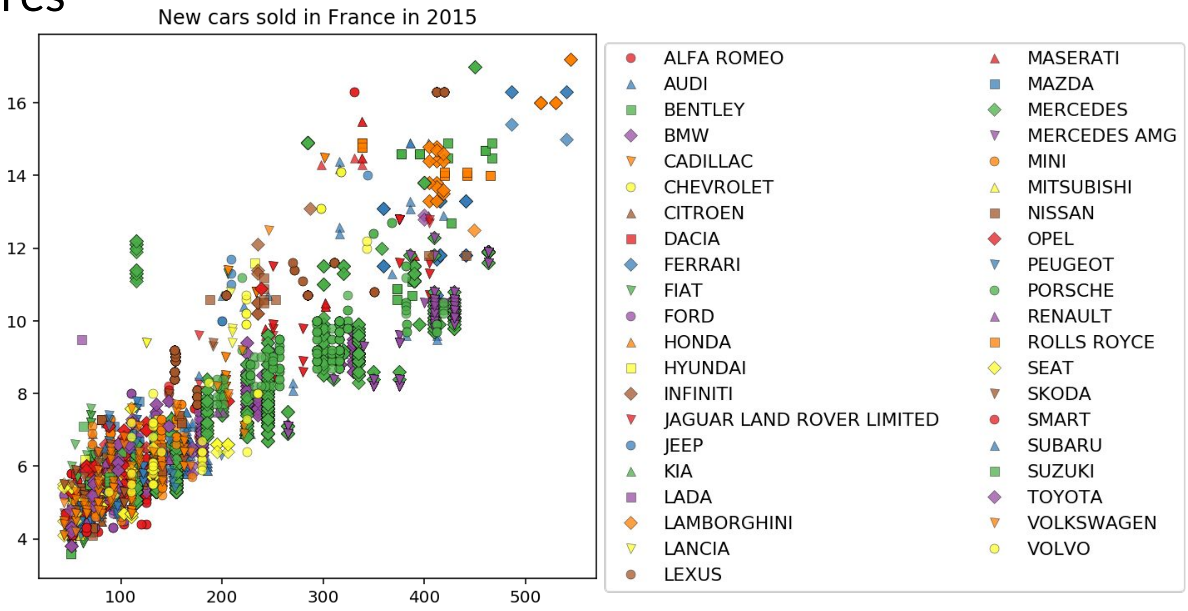
vislint-mpl catches these failures

`require-axes-labels`

`maxcolors`

`no-indistinguishable-series`

`representation-invariance`



(A real chart found in the wild, see paper for details)

Linting can enable difficult to see checks

This inverted sunburst fails a number of vislint-mpl rules:

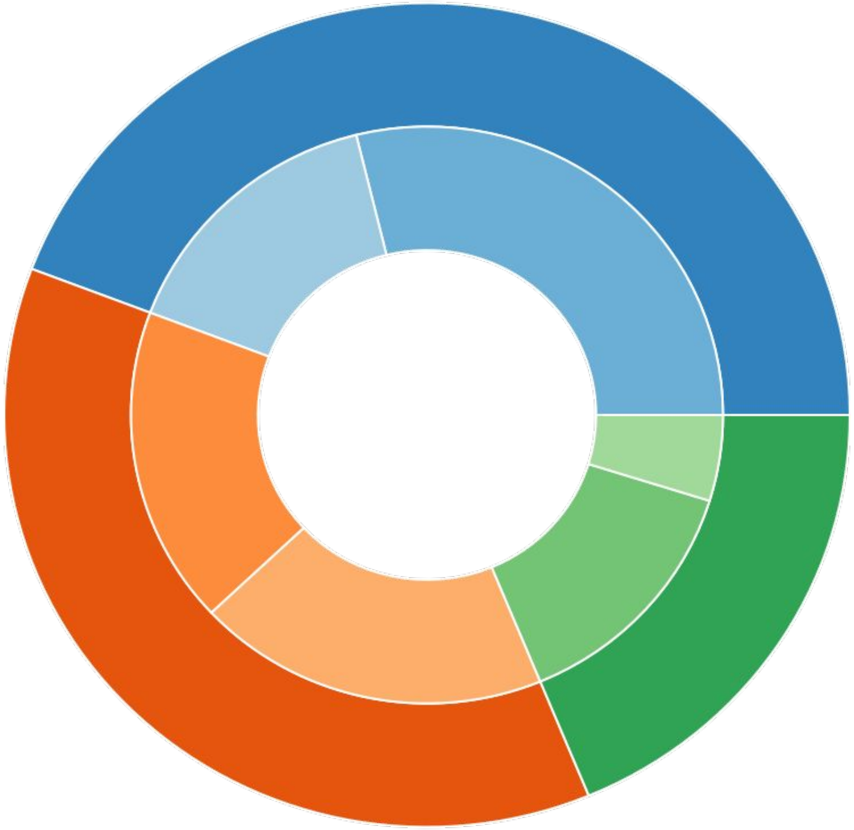
`representation-invariance`

`require-axes`

`require-legend`

`no-pie`

`no-radial`



(Src: MatplotLib Docs)

Linting can enable difficult to see checks

This inverted sunburst fails a number of vislint-mpl rules:

`representation-invariance`

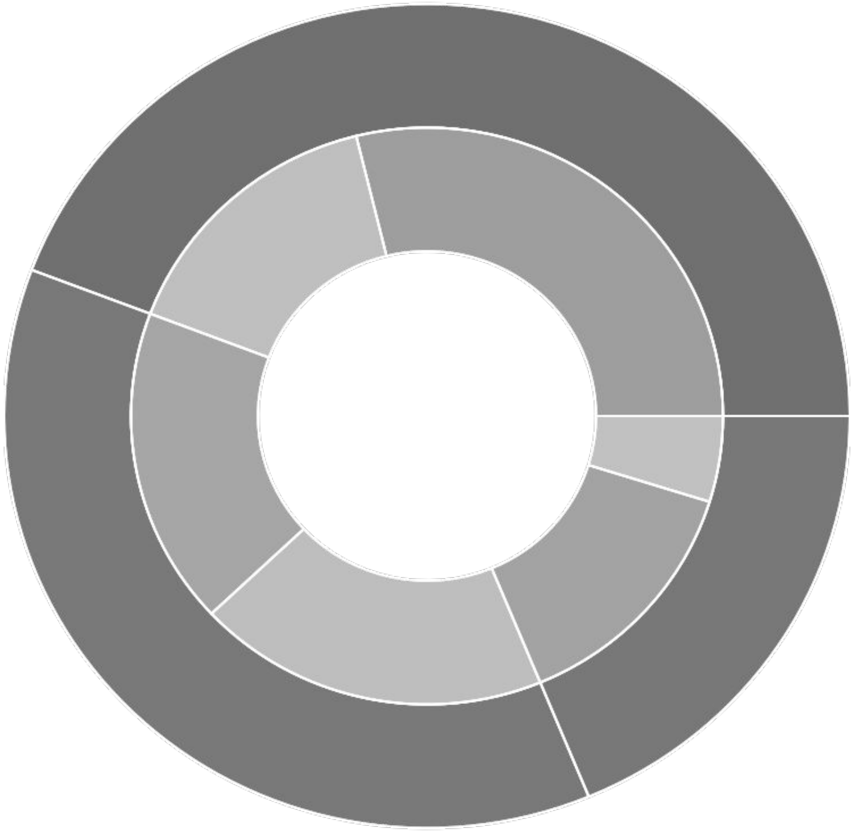
`require-axes`

`require-legend`

`no-pie`

`no-radial`

`printable-colors`



(Src: MatplotLib Docs)

Future Work

Develop additional lint-rules and cover chart more types

Consider implementing a linter for a different plotting domain, such as ggplot, react-vis, or vega/altair.

Create linting systems for the Jupyter notebook system

Important Ideas

Recommendation systems can be sometimes be too limited, instead consider guidance systems.

Linting as a form of analysis based on predefined rules.

Guidance systems can be a springboard for new types of analysis.

We raise a question:

How should vis software be architected to further the aims of guidance &/ recommendation?

(Should our aims for guidance affect software design at all?)