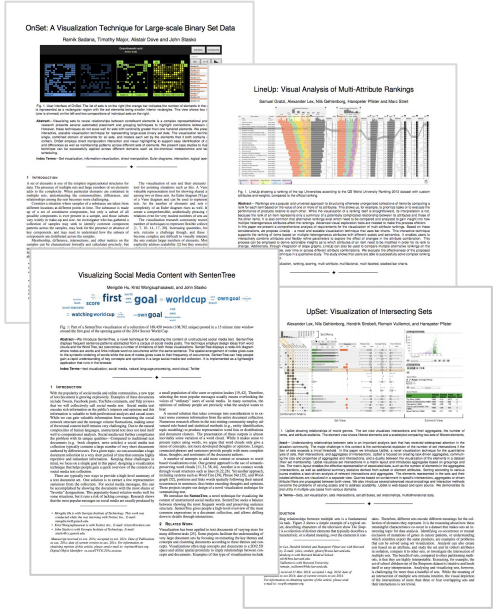
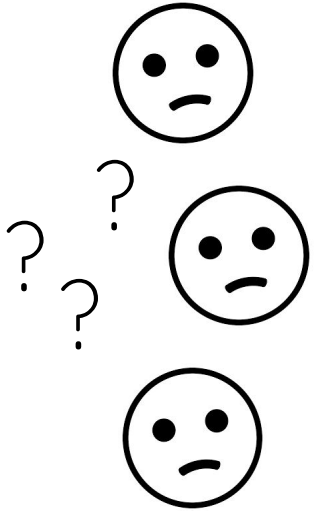
Candela:
An interactive visualization component library for data science

Jeff Baumes
Director of Data and Analytics, Kitware

Lots of research in data visualization, relatively few usable tools





Kitware

Candela: a suite of low-barrier-to-entry modern visualizations

Goal 1. Easy to write

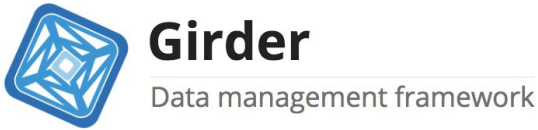
```
export default class MyComp extends VisComponent {
 constructor (el, options) {
   super(el);
   // Setup code here
 }

 render () {
   // Render into this.el here
 }
}
```

Goal 2. Easy to use

```
<body>
  <div id="vis" style="width:400px;height:400px"></div>
  <script src="https://unpkg.com/@candela/all" charset="utf-8"></script>
  <script>
    var el = document.getElementById('vis');
    var vis = new components.ScatterPlot(el, options);
    vis.render();
  </script>
</body>
```

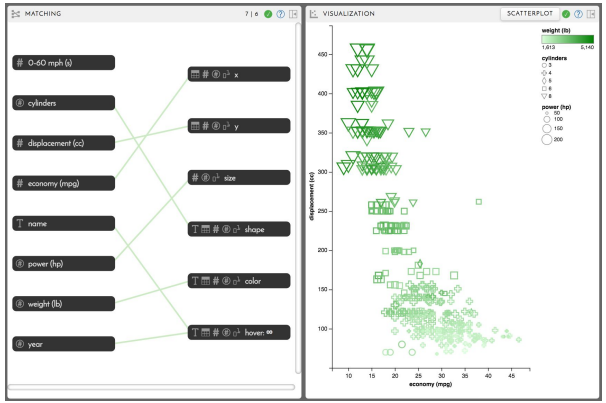
Kitware

Same components available in a variety of environments



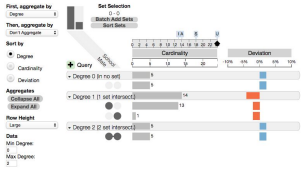
Options specification enables auto UI generation

```
options: [
  {
    name: 'data',
    type: 'table',
    format: 'objectlist'
  },
  {
    name: 'x',
    type: 'string',
    format: 'text',
    domain: {
      mode: 'field',
      from: 'data',
      fieldTypes: [...]
    }
  },
  ...
]
```

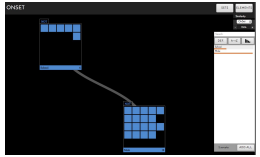


Normalization of API for diverse vis

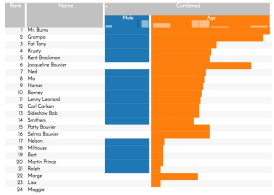
```
new UpSet({
  data: simpsons,
  sets: [“Male”,”School”]
});
```



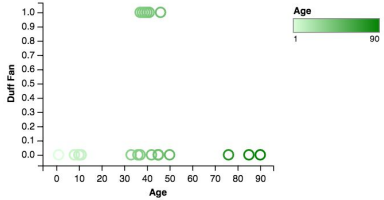
```
new OnSet({
  data: simpsons,
  sets: [“Male”,”School”]
});
```



```
new LineUp({
  data: simpsons,
  fields:
    [“Name”,”Male”, ”Age”]
});
```

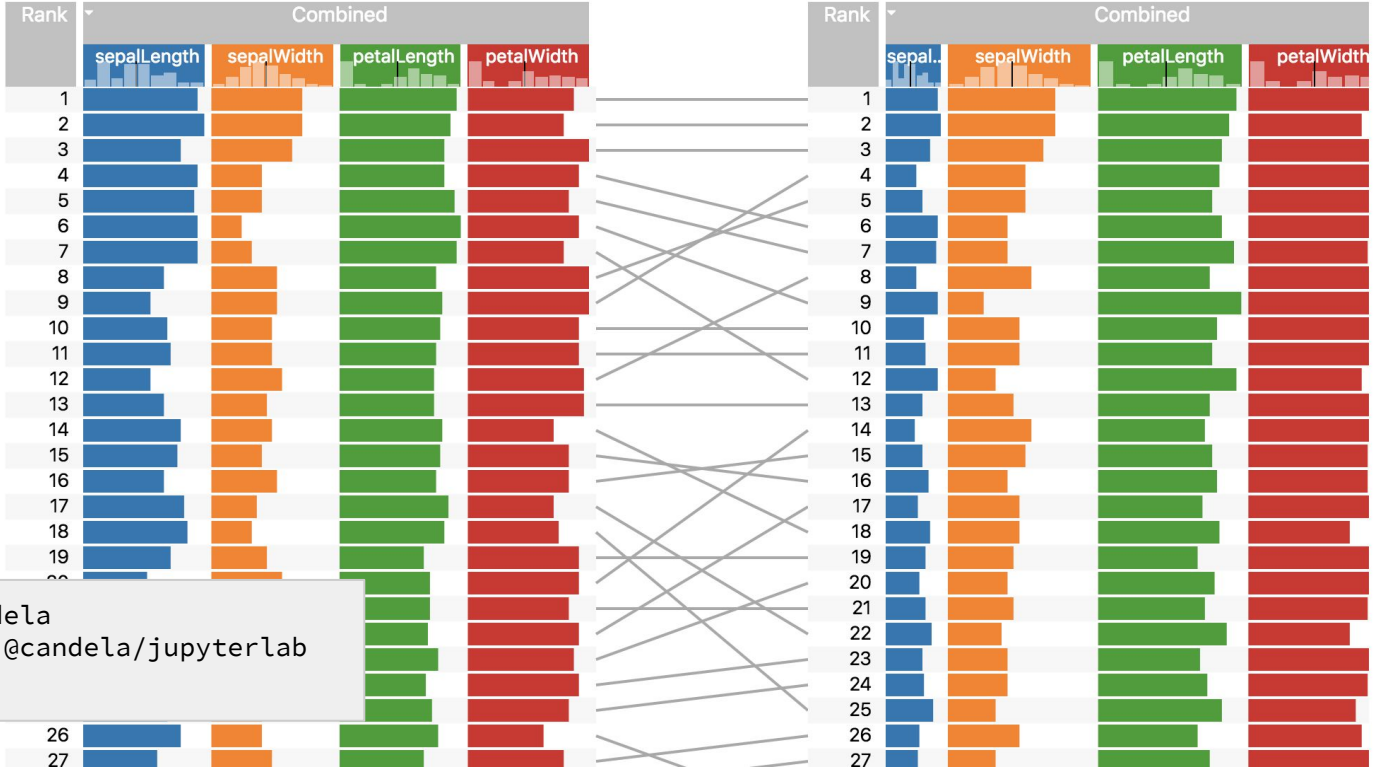


```
new ScatterPlot({
  data: simpsons,
  x: “Age”,
  y: “Duff Fan”,
  color: “Age”
}); // Uses Vega
```



In JupyterLab output cells

```
import requests
data = requests.get('https://raw.githubusercontent.com/vega/vega-datasets/gh-pages/data/iris.json').json()
pycandela.components.LineUp(
    data=data, color='species', fields=['sepalLength', 'sepalWidth', 'petalLength', 'petalWidth'], width=800, height=800)
```



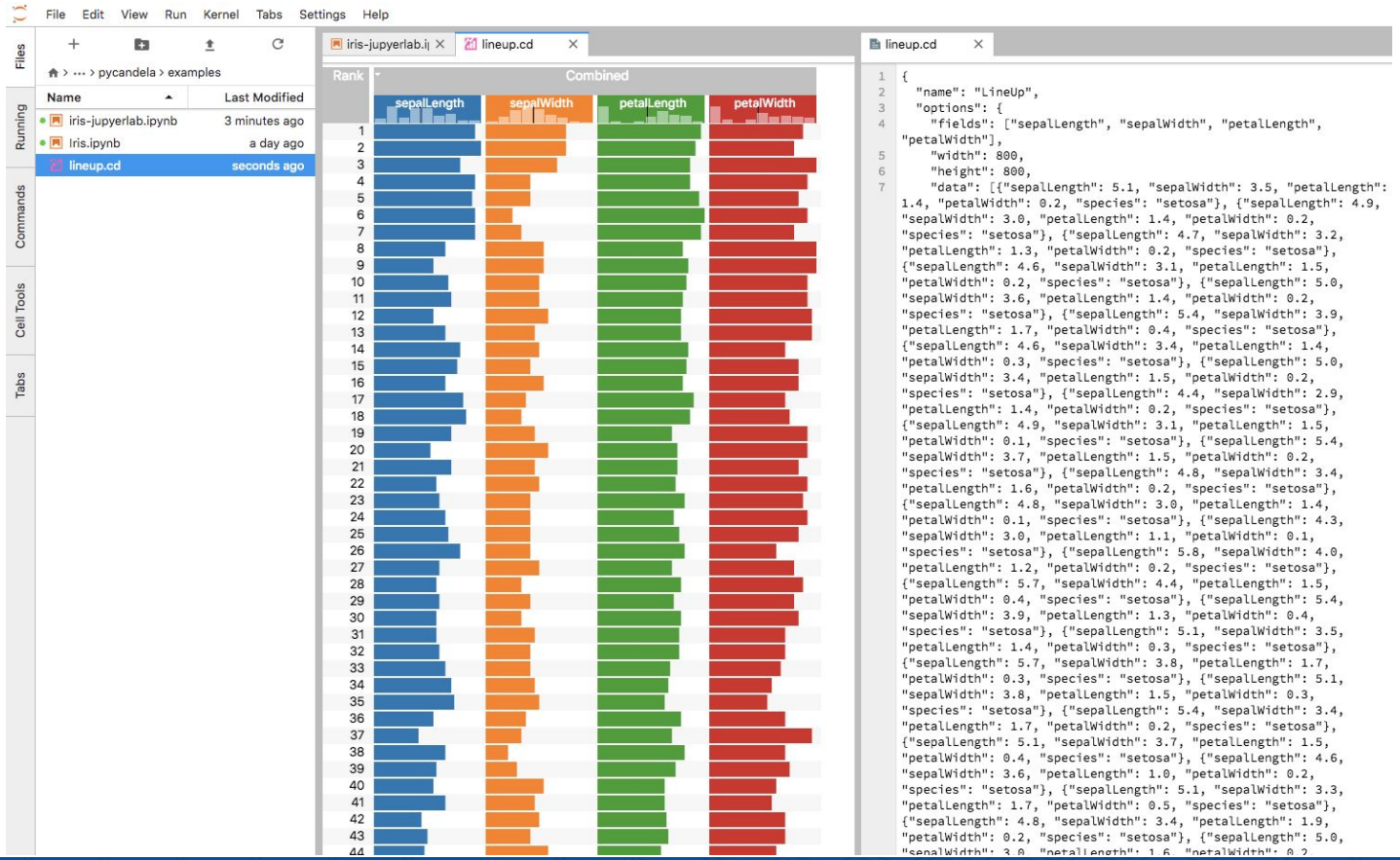
Example uses **LineUp**: Alexander Lex et al. IEEE InfoVis 2013 Best Paper Award

```
$ pip install jupyterlab pycandela
$ jupyter labextension install @candela/jupyterlab
$ jupyter lab
```

In JupyterLab file viewer

Save to *.cd file, recognized as Candela mime type

Open in JSON editor or Candela viewer, edit side by side



In Observable

Observable hosts reactive notebooks in Javascript

Example uses **Vega-Lite**: Arvind Satyanarayan et al. IEEE InfoVis 2017 Best Paper Award

https://beta.observablehq.com/@jeffbaumes/candela

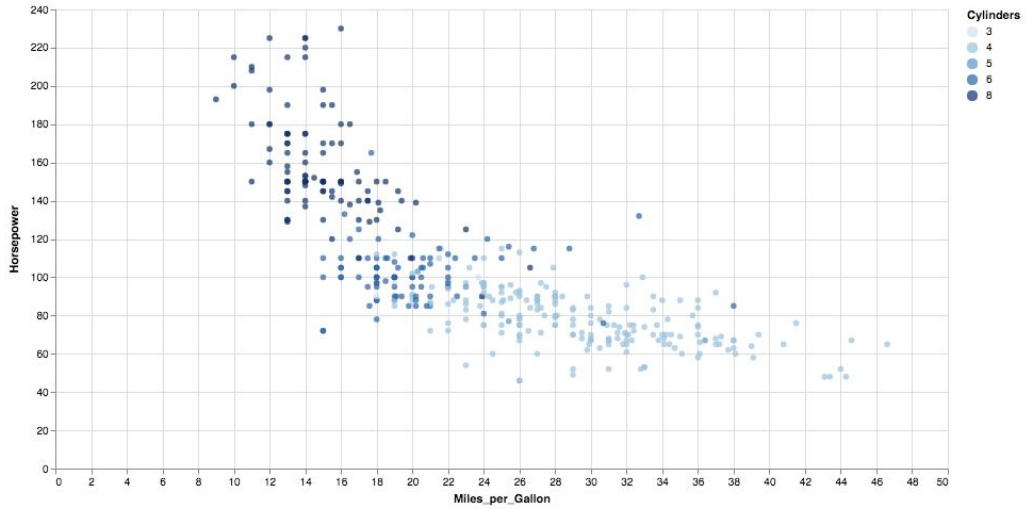
```
candela = require('@candela/all')
```

data

```
▸Array(406) [Object, Object, Object, Object, Object, Object, Object, Object, Object, Object, Object, Object, Objec
```

```
data = require(`@observablehq/cars`)
```

vis

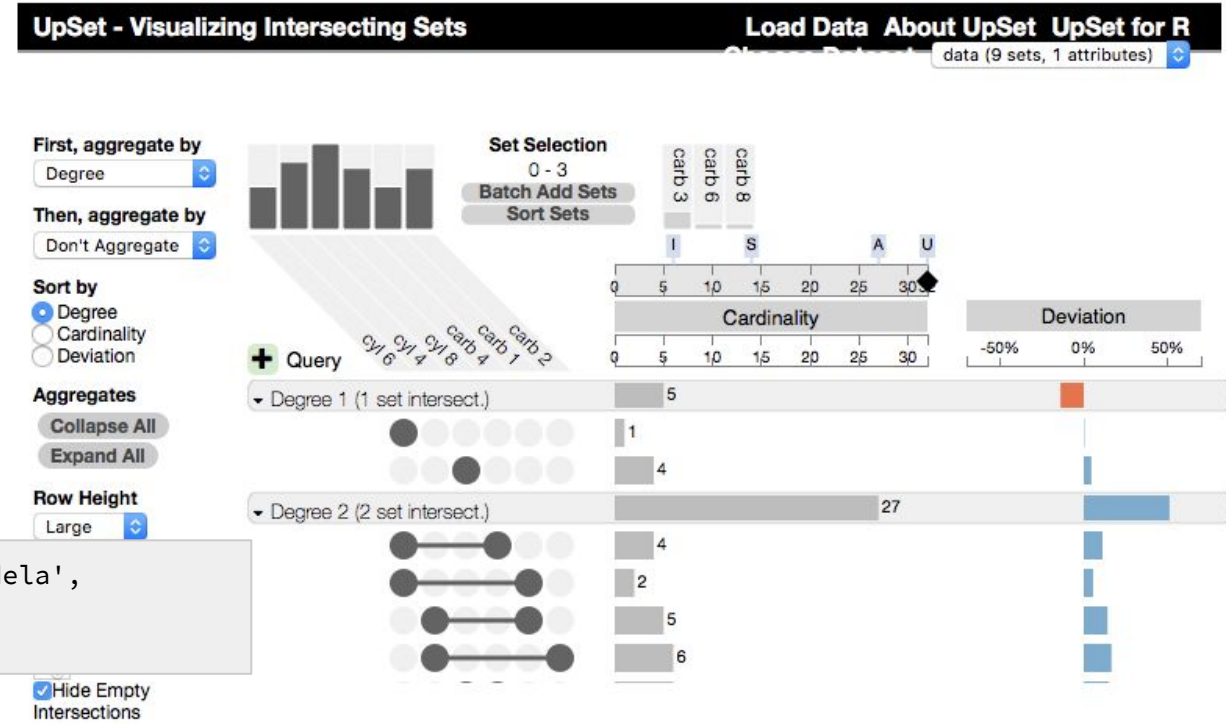



```
vis = {
  const el = html`<div style="width:${width}px;height:${width/2}px"></div>`
  yield el
  const vis = new candela.components.ScatterPlot(el, {
    data: data,
    x: 'Miles_per_Gallon',
    y: 'Horsepower',
    color: 'Cylinders',
    colorType: 'ordinal'
  });
  vis.render();
  return vis;
}
```

In RStudio

```{r}
candela('UpSet', data=mtcars, id='_row', fields=list('cyl', 'carb'))
```

Example uses **UpSet**: Alexander Lex et al. IEEE InfoVis 2014



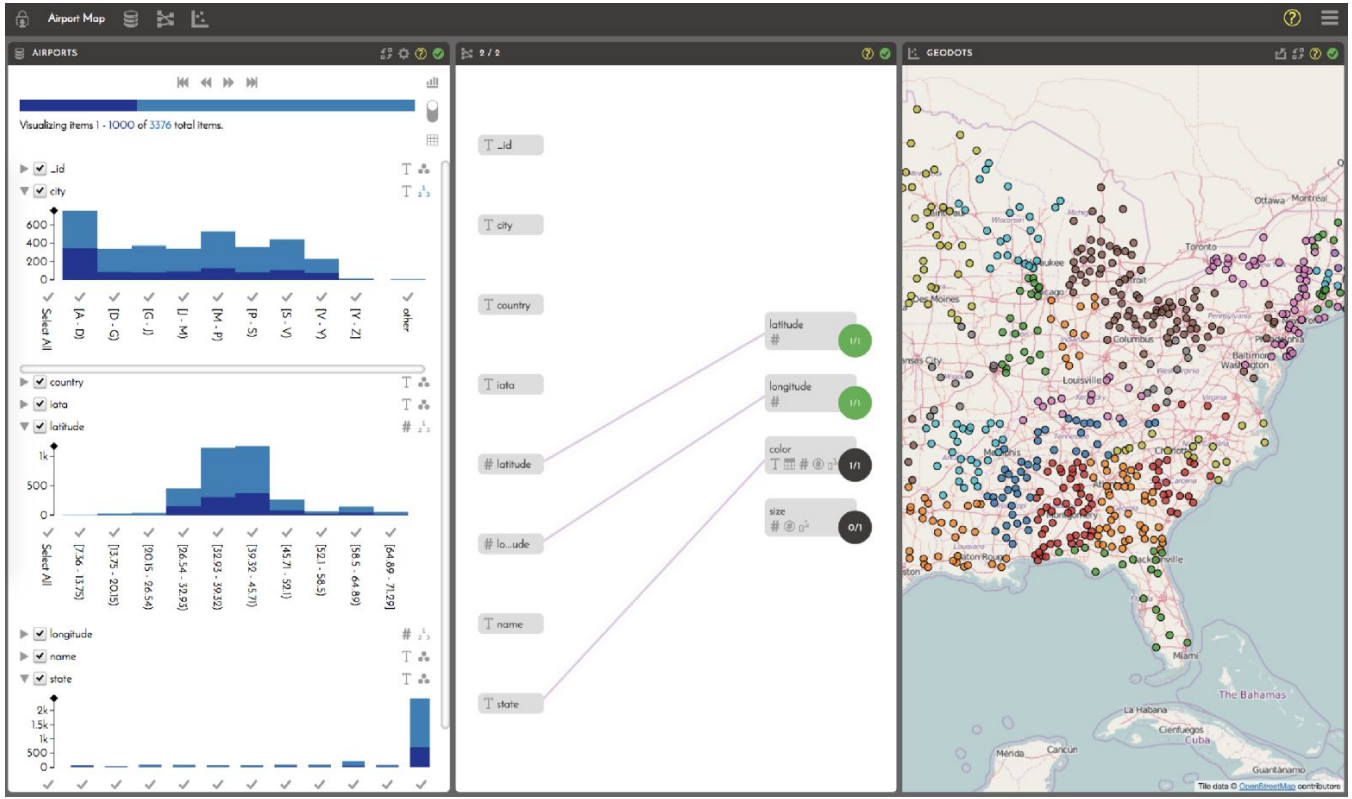

```
devtools::install_github('Kitware/candela',
  subdir='R/candela',
  dependencies = TRUE)
```

In ResonantLab

User/data/session management backed by Girder

Data-to-visual matching interface

Example uses **GeoJS**: High-performance geospatial visualization



https://resonantlab.kitware.com

candela.readthedocs.io

try it out
ask a question
contribute

Kitware