VIS2020

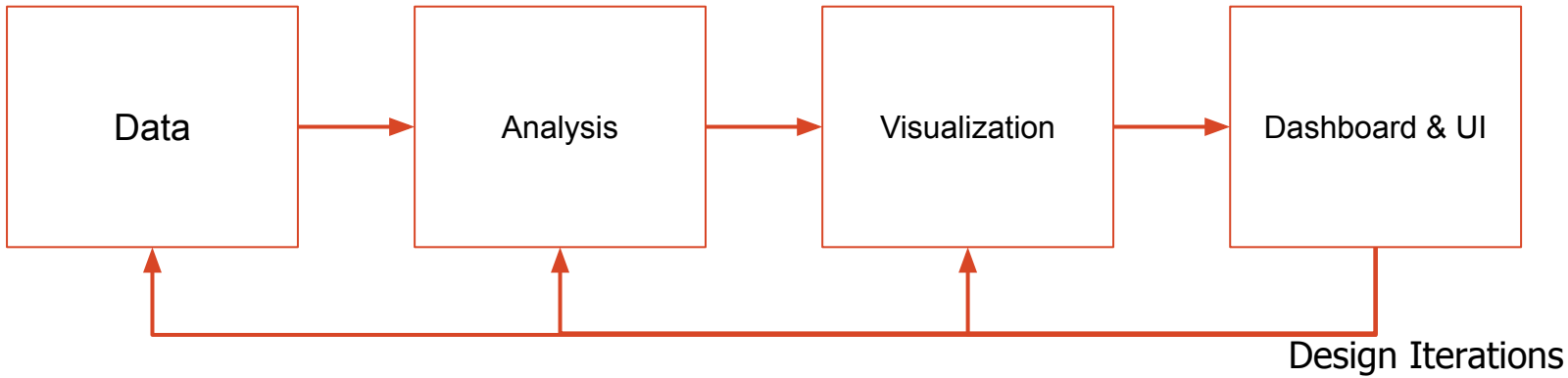
P6: A Declarative Language for Integrating Machine Learning in Visual Analytics

Kelvin Li and Kwan-Liu Ma
University of California, Davis

Motivation

Visual analytics systems are useful.

But building effective visual analytics systems is difficult!



Challenges

- Analyzing and visualizing large datasets
- Integrating machine learning methods with interactive visualizations
  - Multiple programming languages and libraries for data analysis and visualization (e.g., Python and JavaScript)
  - Server-client architecture and communication
  - User interfaces for controlling the parameters of machine learning and visualization methods

Declarative Visualization Languages

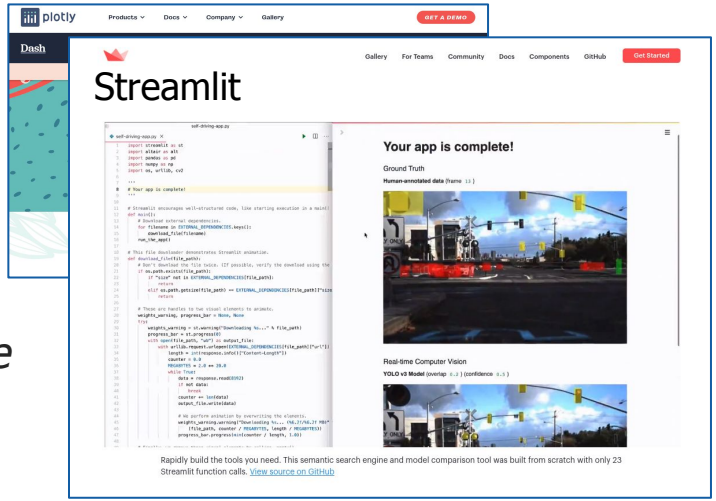
Describe ***what*** the visualization should look like, instead of ***how*** it should be rendered.

- Users focus on design
- Tools generate dataflows

Declarative visual encoding: *ggplot2, D3*

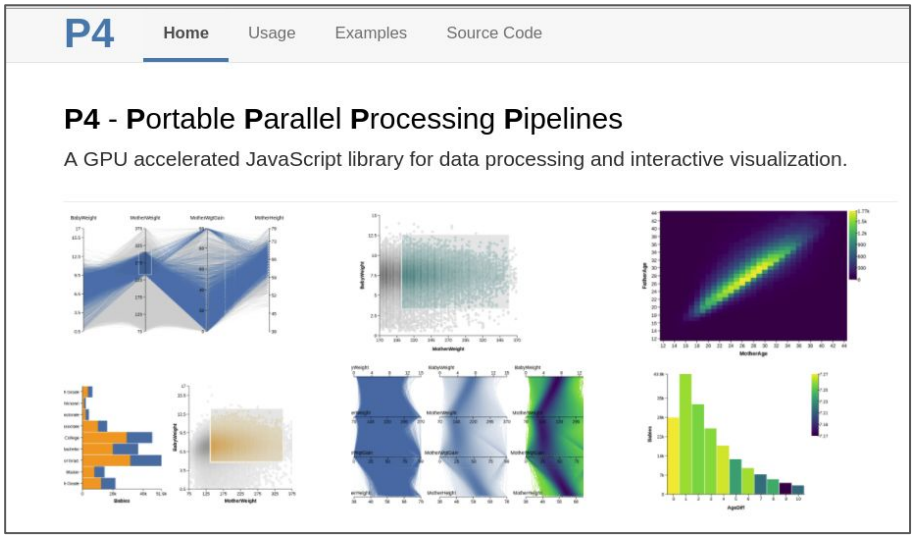
Declarative interactive visualization: *Vega, Vega-Lite*

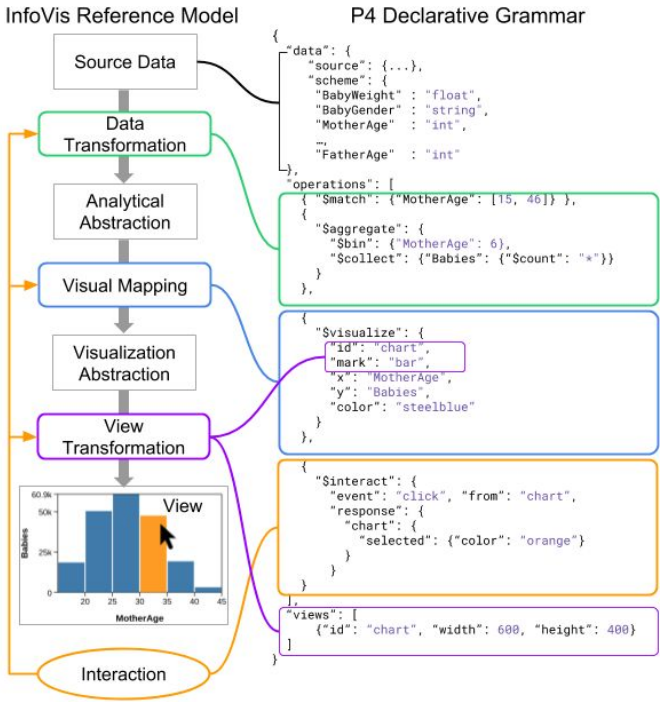
Dash



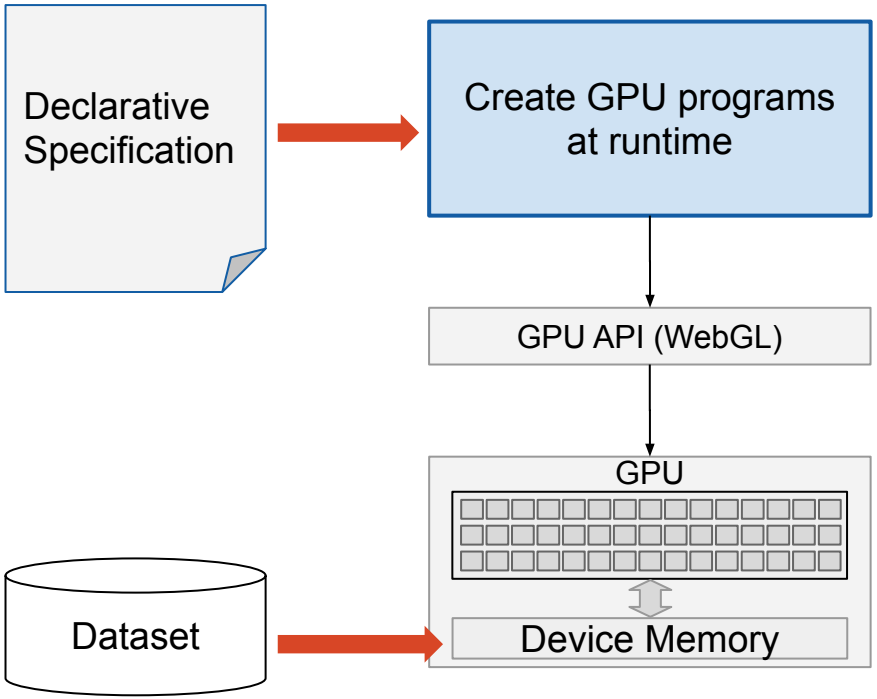
Declarative Visualization and GPU Computing for the Web

https://jpkli.github.io/p4/



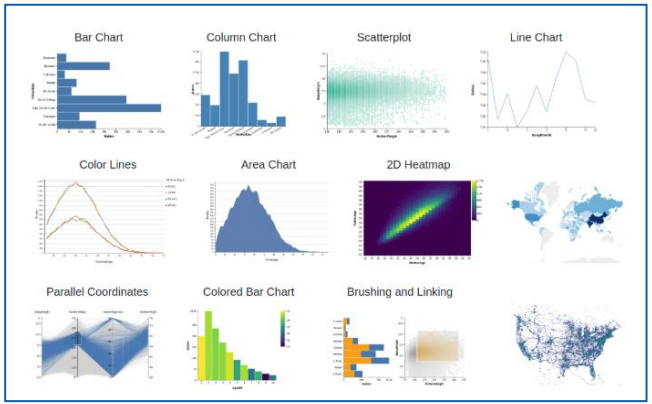


P4 Framework



Jianping Kelvin Li and Kwan-Liu Ma. "P4: Portable parallel processing pipelines for interactive information visualization." *TVCG 2018*.

Interactive data visualizations



P6's Goal

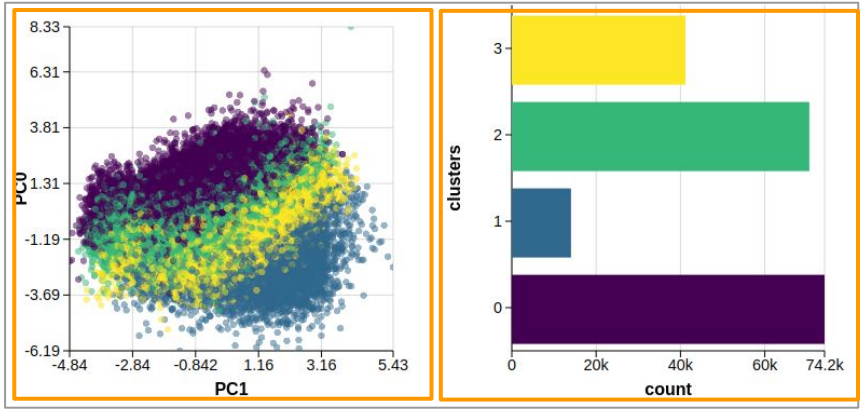
Lower the "threshold" for building visual analytics systems.

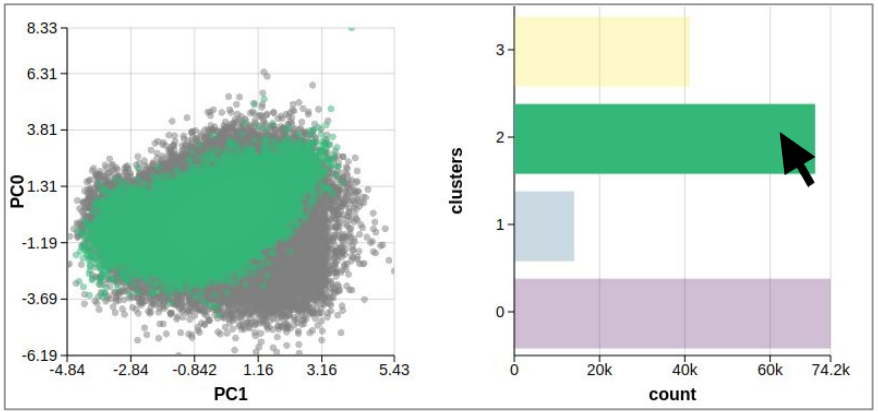
- A declarative visual analytics language
- An efficient system framework
  - Integrate machine learning and interactive visualizations
  - Facilitate visual analysis of large datasets on the web

```
let app = p6(config)
  data({url: 'data/UsBirth.csv'})
  analyze({
    PC: {
      algorithm: 'PCA', n_components: 2
    },
    clusters: {
      algorithm: 'KMeans', n_clusters: 4
    }
  })
  visualize({
   c1: {
     mark: 'circle',
     x: 'PC1', y: 'PC0',
     color: 'clusters',
     size: 5, opacity: 0.25
   },
```

```
app.interact({
   event: 'click',
   from: 'c2',
   response: {
     c1: { unselected: {color: 'gray'} },
     c2: { unselected: {opacity: 0.25} }
   }
 })
```

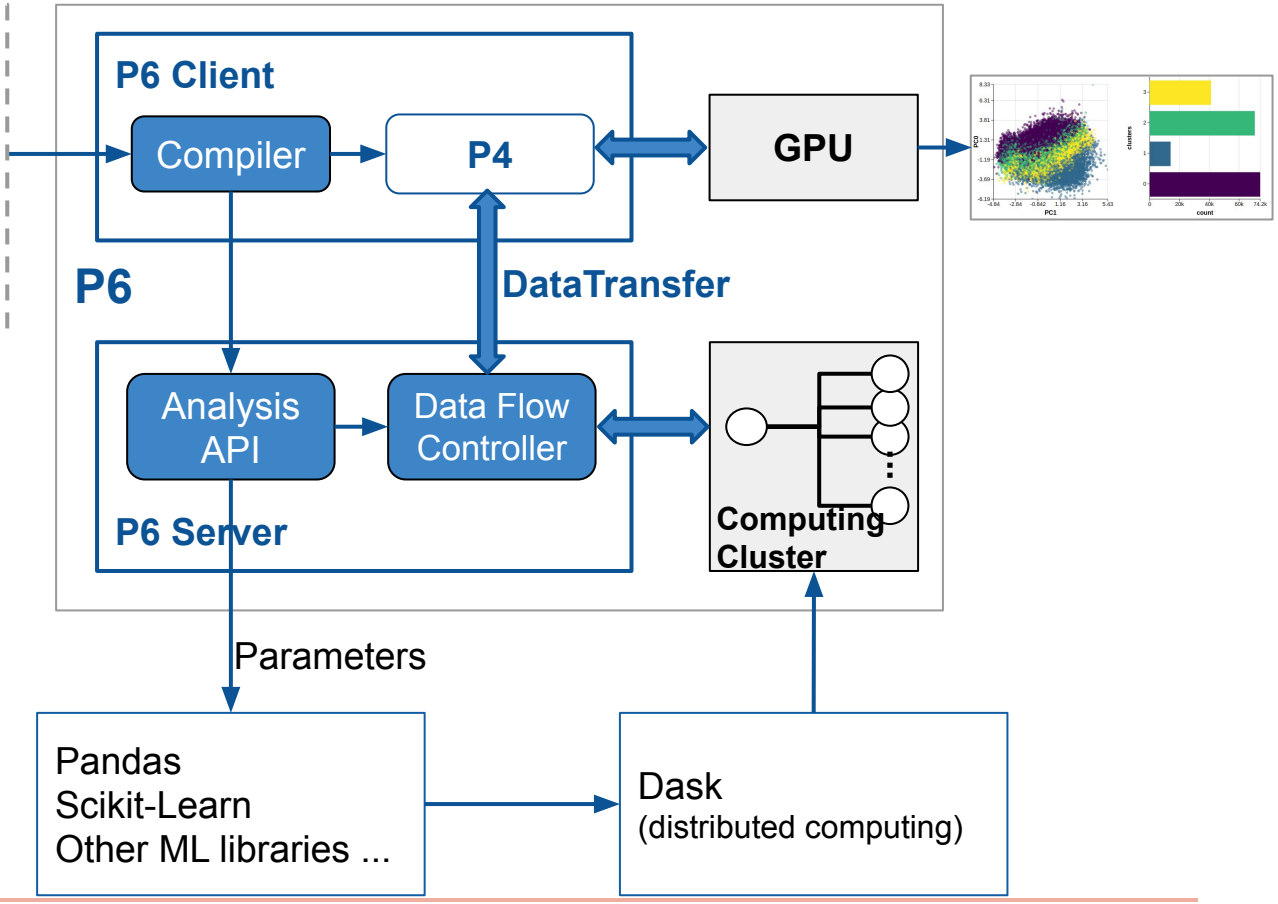
```
  })
```



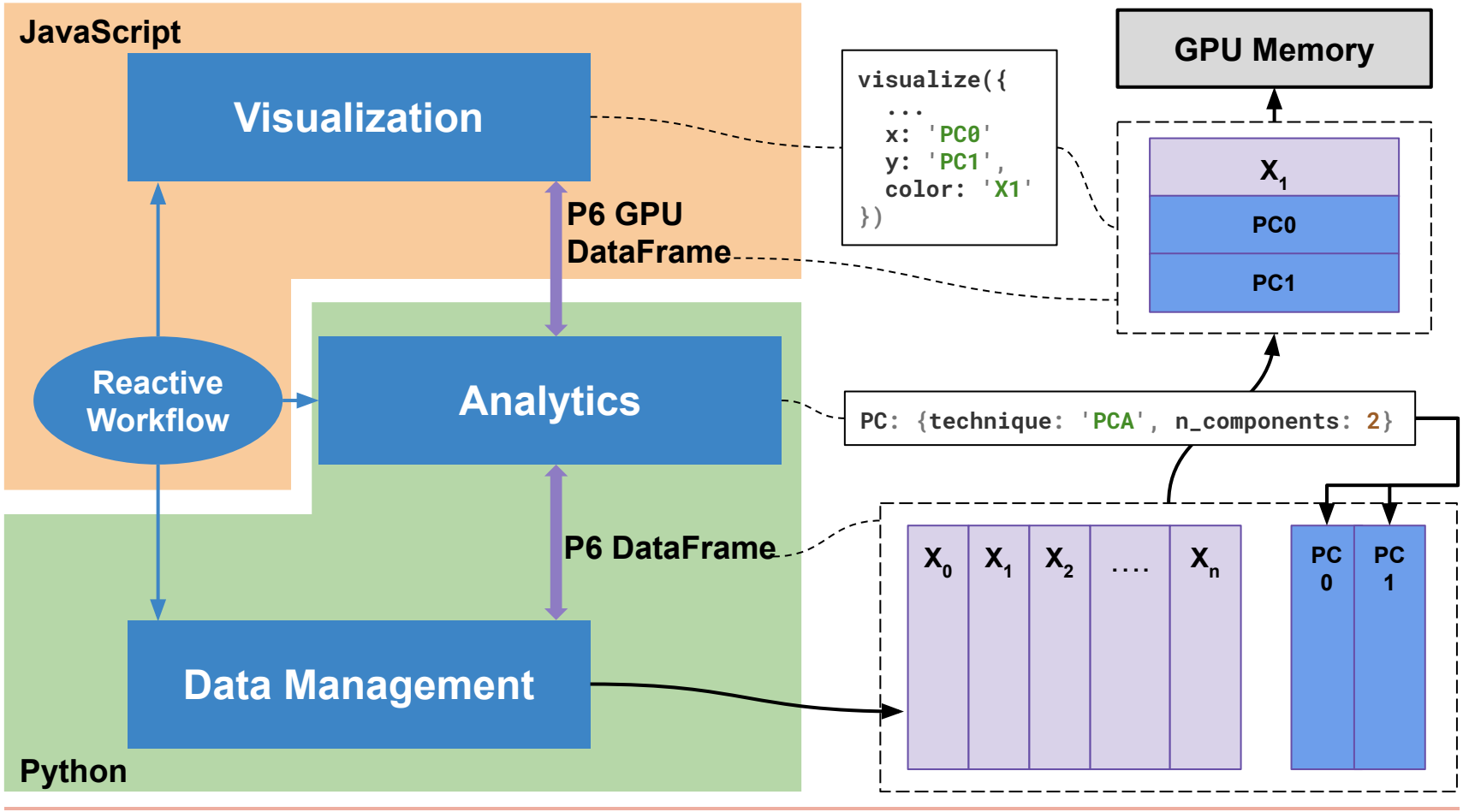


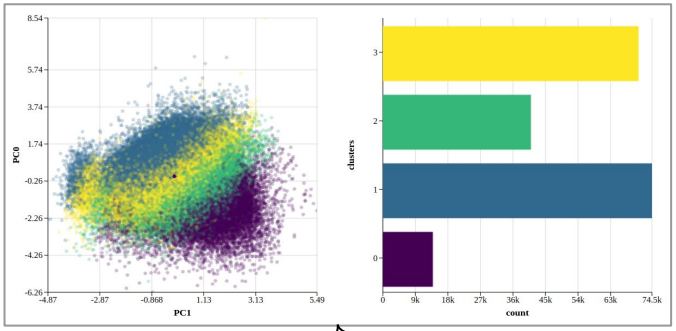
Declarative Specification

```
p6(config).data({...})
.analyze({...})
.view({...})
.visualize({...})
.interact({...})
```



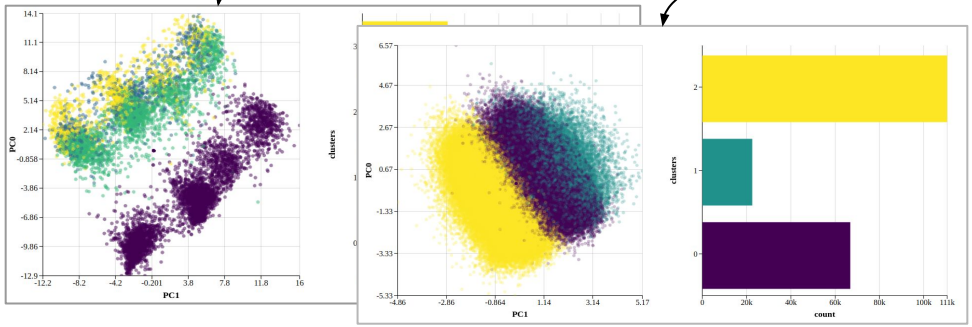
P6 automatically create efficient programs to execute operations across server and client!

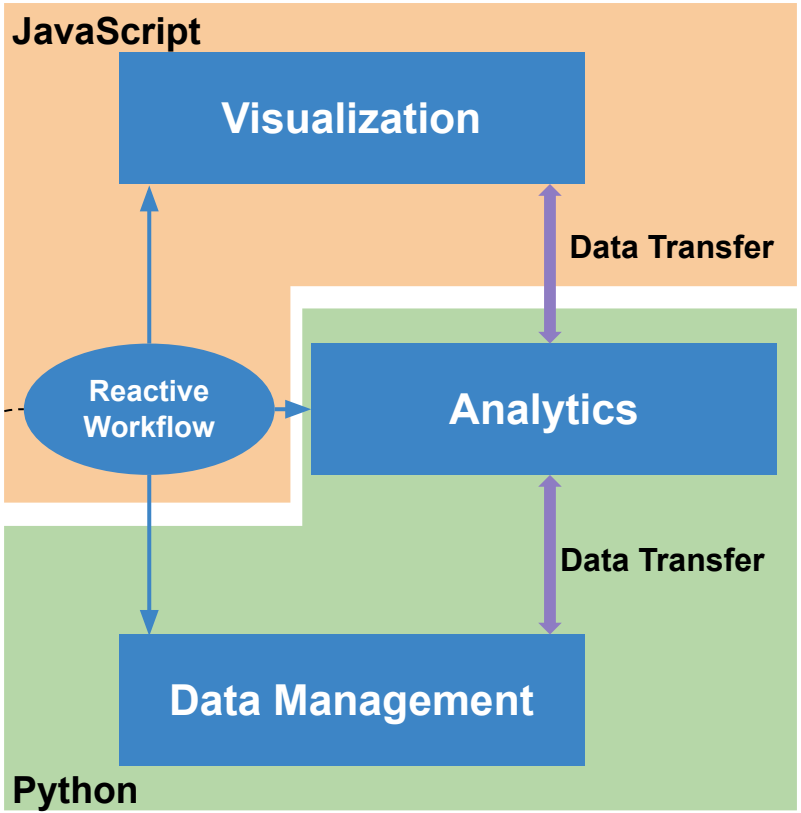

JavaScript
Visualization
P6 GPU DataFrame
Reactive Workflow
Analytics
P6 DataFrame
Data Management
Python
visualize({
...
x: 'PC0'
y: 'PC1',
color: 'X1'
})
GPU Memory
X1
PC0
PC1
PC: {technique: 'PCA', n_components: 2}
X0
X1
X2
....
Xn
PC 0
PC 1

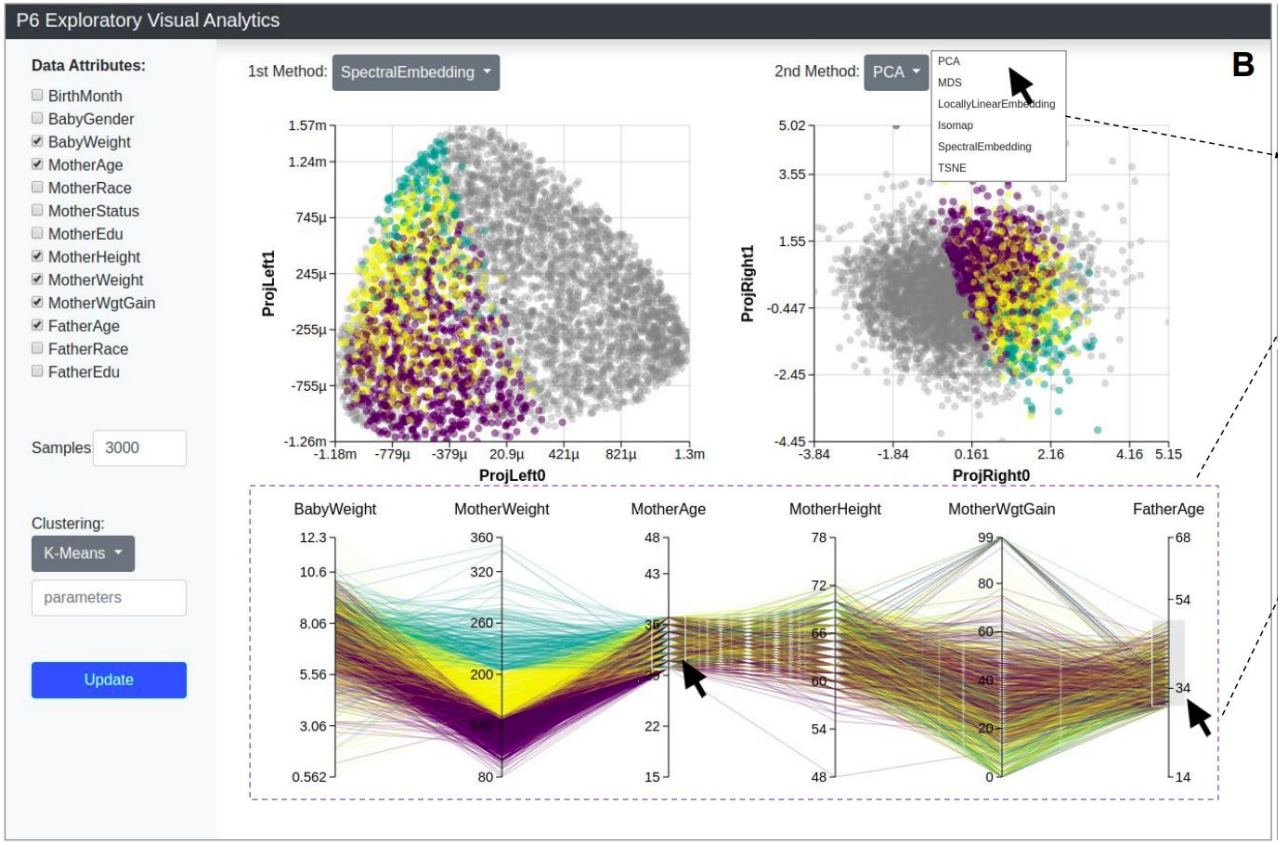



```
app.analyses.PC.algorithm = 'Isomap'
```

```
app.analyses.clusters.n_clusters = 3
```







C

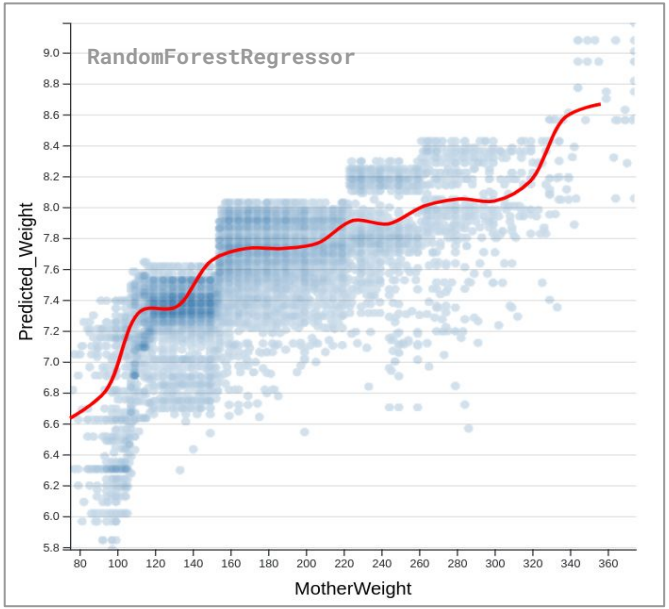
```
dropDownMenuL.on('change', evt => {
 app.analyses.ProjLeft
   .features = dropDownMenuL.value
})

dropDownMenuR.on('change', evt => {
 app.analyses.ProjRight
   .features = dropDownMenuR.value
})

app.vis.ChartBottom
  .y = selectedDataAttributes
```

D

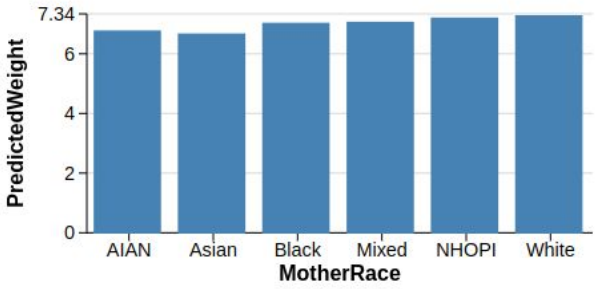
```
app.interact({
  event: 'brush',
  from: [
    'ChartLeft',
    'ChartRight',
    'ChartBottom'
  ],
  response: {
    ChartLeft: {
      unselected: {color: 'gray'}
    },
    ChartRight: {
      unselected: {color: 'gray'}
    },
    ChartBottom: {
      unselected: {opacity: 0}
    }
  }
})
```

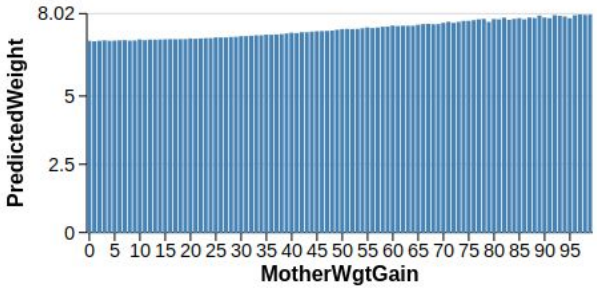
```
// train and save a model
app.gridSearch({
   $BabyWeightRegressor: {
     module: 'ensemble',
     method: 'RandomForestRegressor',
```

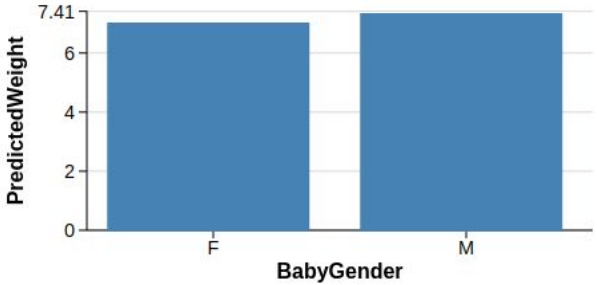
```
app.data({url: '/data/Pregnancies.csv'})
 .analyze({
   Predicted_Weight: '$BabyWeightRegressor',
 })
 .visualize({
   c1: [
     {
       mark: 'circle', color: 'steelblue',
       x: 'MotherWeight', y: 'Predicted_Weight',
       size: 8, opacity: 'auto',
     },
     {
       $transform: {
         $aggregate: {
           $bin: 'MotherWeight',
           $collect: {Predicted_Weight: '$avg'}
         }
       },
       mark: 'spline', color: 'red',
       x: 'MotherWeight', y: 'Predicted_Weight',
       size: 3, opacity: 1
     }
   ]
 })
```

```
app.data({url: 'data/Pregnancy.csv'})
  .analyze({
    PredictedWeight: '$BabyWeightRegressor',
  })
  .visualize({
    $rows: {
      $select: {
        model: '$BabyWeightRegressor',
        attribute: 'feature_importances_',
        sort: 'desc',
        limit: 3
      },
      $transform: {
        $aggregate: {
          $group: '$select',
          $collect: {
            PredictedWeight: {$avg: 'PredictedWeight'}
          }
        }
      },
      mark: 'bar',
      x: '$select',
      height: 'PredictedWeight',
      color: 'steelblue'
    }
  })
  .execute()
```

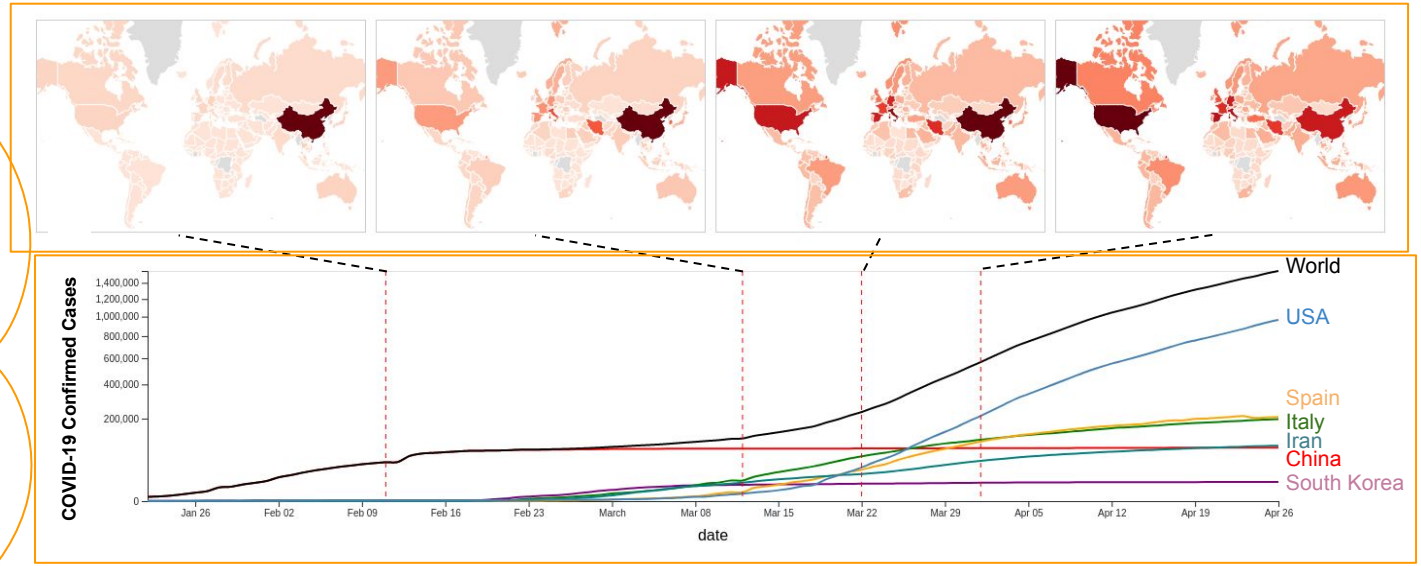






```
let app = p6(config1).data({
 url: 'https://raw.githubuserco/CSSE...',
 type: 'CSV'
})
.preprocess({...})
.analyze({
  ChangePoints: {
   algorithm: 'CPD',
   attribute: 'World', n: 4,
   method:'Window', width: 5
  }
})
.visualize({
  lineChart: {
   mark: 'spline', size: 2, x: 'date',
   y: {columns: topCountries,
    exponent: 0.5}
 }
}).execute();

let changePoints = app.result('json')
 .filter(d => d.ChangePoints !== 0)
 .map(d => new Date(d.date));

app.annotate({
   id: 'lineChart', mark: 'rule',
   size: 1, color: 'red',
   x: changePoints
 });

let geoMaps = p6({...})
.data({...}).preprocess({...})
.visualize({
   $cols: {
     $select: changePoints
     …
})
```

COVID-19 Global Cases



Change point detection for analyzing time-series data

Summary

P6 aims to lower the threshold for building visual analytics systems by providing a declarative visual analytics language.

Future work:

- Provide debugging and development tools for using P6
- Collect user feedback to improve the P6 toolkit

Codes & Demos: https://github.com/jpkli/p6

Contact: jpkelvinli@gmail.com

Thank You !

Acknowledgements

This research is sponsored in part by the U.S. National Science Foundation through grant IIS-1741536 and IIS-1528203, and also by the U.S.Department of Energy through grant DE-SC0014917.

VIS2020

Acknowledgements

This research is sponsored in part by the U.S. National Science Foundation through grant IIS-1741536 and IIS-1528203, and also by the U.S.Department of Energy through grant DE-SC0014917.