# Dynamic Visualizations (II)

Riccardo Pucella

October 26, 2015

#### Last time

Interactions for dynamic views:

- switching: moving "laterally" across views
- filtering: which data rows do we consider?

Remaining main interaction:

- refining

Sometimes referred to as drill-down, or cross tabulation.

- Split data into groups according to criteria
  - e.g., age groups
- Aggregate independently over each group
  - e.g., sum over the various age groups
- Visualize results for groups
- Can let splitting criteria be interactively chosen

Facebook users: 4600

Twitter users: 5600

		Male	Female
Facebook users:	4600	2000	2600
Twitter users:	5600	2600	3000

<35 >35

Facebook users: 4600 3500 1100

Twitter users: 5600 4000 1600

<35 >35

Facebook users: 4600 3500 1100

Twitter users: 5600 4000 1600

<35 >35

Facebook users:

4600 3500

1100

Twitter users:

5600

4000

1600

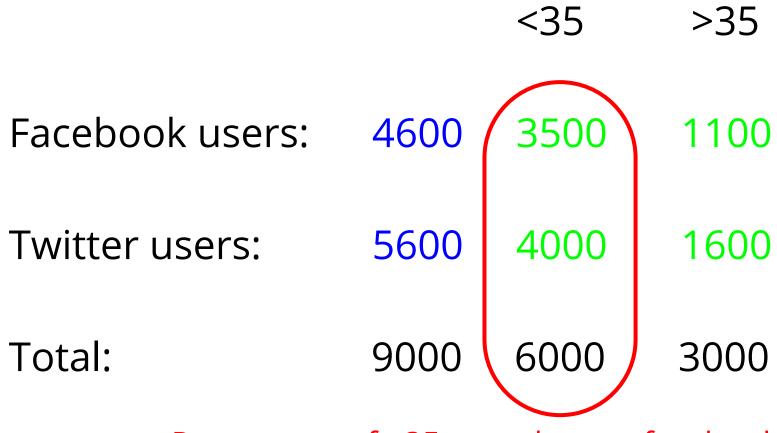
Percentage of facebook users < 35 y.o. 76% Percentage of facebook users > 35 y.o. 24%

<35	>35

Facebook users: 4600 3500 1100

Twitter users: 5600 4000 1600

Total: 9000 6000 3000



Percentage of <35 y.o. who use facebook 58% Percentage of >35 y.o. who use facebook 66%

### **Basic aggregation**

(Variant from last time)

```
var all = 0, count = 0;
GLOBAL.data.forEach(function(r) {
    all += 1;
    c = r[media_col];
    if (c==="Yes") {
       count += 1;
    }
})
// returns # rows and count for specific media
```

## **Aggregation + filtering**

(Variant from last time)

```
var all = 0, count = 0;
GLOBAL.data.forEach(function(r) {
   if (keepSelectorsRow(r)) {
      all += 1;
      c = r[media_col];
   if (c==="Yes") {
      count += 1;
      }
   }
})
// returns # rows and count for specific media
```

## **Aggregation + filtering**

(Variant from last time)

```
var all = 0, count = 0;
GLOBAL.data.forEach(function(r) {
   if (keepSelectorsRow(r)) {
     all += 1;
     c = r[media_col]
     if (c==="Yes") {
       count +=
                    Check that row r is kept by the
                           various selector
// returns #
```

## **Aggregation + refining**

```
var all = 0, counts = \{\};
GLOBAL.data.forEach(function(r) {
  c = r[media_col];
  if (c==="Yes") {
    all += 1;
    var group = r[split_col];
    if (group in counts) {
      counts[group] += 1;
    } else {
      counts[group] = 1;
```

// returns media count and media count for each group

## Refining in practice

- Refining is sometimes selector-based
  - "choose a variable by which to group the data"
- Can attach refining to chart elements
  - "click on element to refine"

## Aggregation + filtering + refining

```
var all = 0, counts = \{\};
GLOBAL.data.forEach(function(r) {
  if (keepSelectorsRow(r)) {
    c = r[media_col];
    if (c==="Yes") {
      all += 1;
      var group = r[split_col];
      if (group in counts) {
        counts[group] += 1;
      } else {
        counts[group] = 1;
```

## Aggregation + filtering + refining

## Aggregation + filtering + refining

Note that original *aggregation* + *filtering* can be done with this function:

```
var result = countSplitByColumn
        (GLOBAL.data,
        function(r) { return keepSelectorsRow(r); },
        media_col);
var all = result.all;
var count = result.counts.Yes;
```

## More interactions: tooltips

- On mouseover, create a tooltip
  - shape
  - data to put in tooltip (details)
- On mouseout, remove tooltip

- Some subtleties
  - clicks
  - getting the right data
  - creating versus reusing tooltip