HTML5 Local Storage... R... Testing...

Matthew B Harris (harris112)   follow me on twitter @mattbenh

Problem To Solve

- Data Intensive
  - 11063 Records in test

- Long Page Load Times
  - Max load time over one second for test data
  - Current load time for our site is 11.887 seconds

- Static Data

Things To Learn

- HTML5 Web Storage
  - Bigger Better Cookies

- Debugging Tools
  - Yslow (add on to Firebug)
  - Safari Developer Tools

- R
  - Used in our group
  - Taking class next week

Code!

```javascript
<script type="text/javascript">
    function loadDB(){
        if(typeof(Storage)==="undefined"){
            window.location="WithOutHTML5.jsp";
        }
        else{
            var type = localStorage.getItem('0');
            if(type === 'filled'){
            }
            else{
                window.location="WithOutHTML5.jsp";
            }
        }
    }
    window.onload = loadDB;
</script>
```

↗ Pulling data from local storage

```html
<!-- HTML STUFF -->

<ul name="list" id="list">
    <script type="text/javascript">
        for(var i = 0; i < localStorage.length-1; i++){
            var key = localStorage.key(i);
            var val = localStorage.getItem(key);
            document.write("<li>" + i + " | " + val + "</li>");
        }
    </script>
</ul>
```
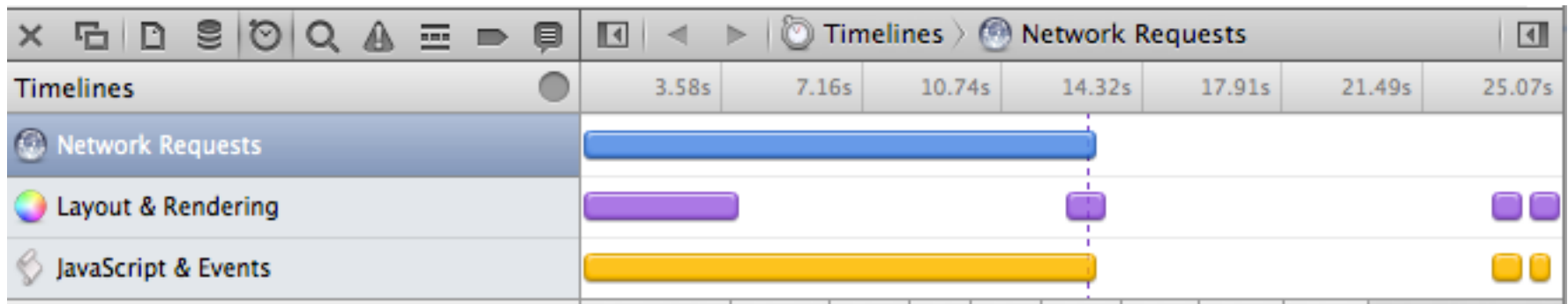
Code!

- ↗ Pushing data into local storage

```javascript
<script type="text/javascript">
    function loadDB(){
        localStorage.setItem('0', 'filled');
        var list = document.getElementById('list').getElementsByTagName("*");
        for(var i = 0; i < list.length; ++i){
            var a = i+1;
            var b = list[i].innerHTML;
            localStorage.setItem(a, b);
        }
    }
    window.onload = loadDB;
</script>
```

Collecting Data

↗ Yslow

↗ Safari dev tools

↗ Data collection not efficient would be better suited for automated collection

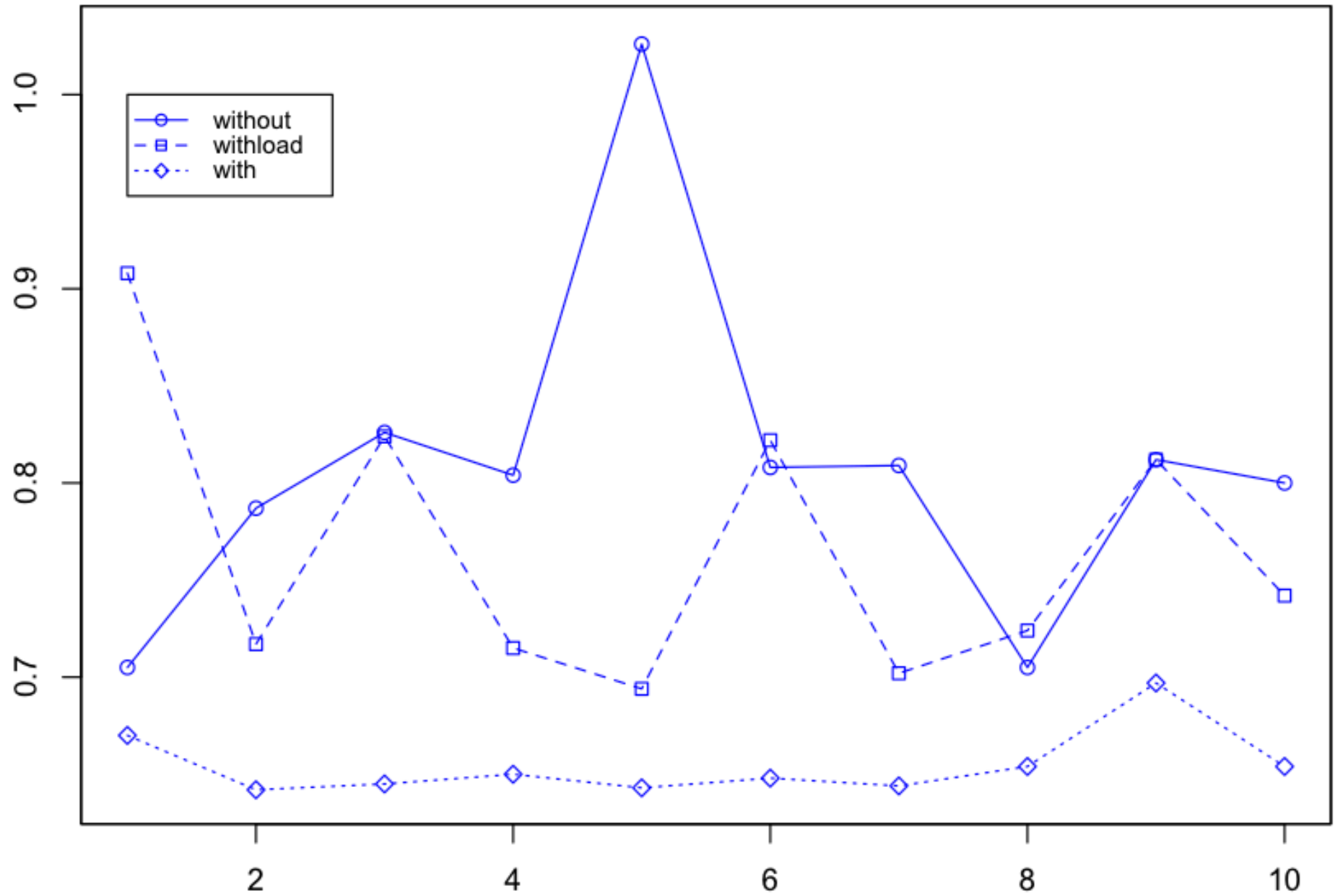↗ Manually cleared cookies and cache after each page load

Raw Data

	WithOutHTML5	WithHTML5	
	Time	First Time	Second Time
0	0.705	0.908	0.67
1	0.787	0.717	0.642
2	0.826	0.824	0.645
3	0.804	0.715	0.65
4	1.026	0.694	0.643
5	0.808	0.822	0.648
6	0.809	0.702	0.644
7	0.705	0.724	0.654
8	0.812	0.812	0.697
9	0.8	0.742	0.654
5NS	without	withload	with
min	0.705	0.694	0.642
Q1	0.787	0.715	0.644
mean	0.806	0.733	0.649
Q3	0.812	0.822	0.654
max	1.026	0.908	0.697
Records	Loaded	11063	

Page Load Times

Legend:
- without (solid line, circle markers)
- withload (dashed line, square markers)
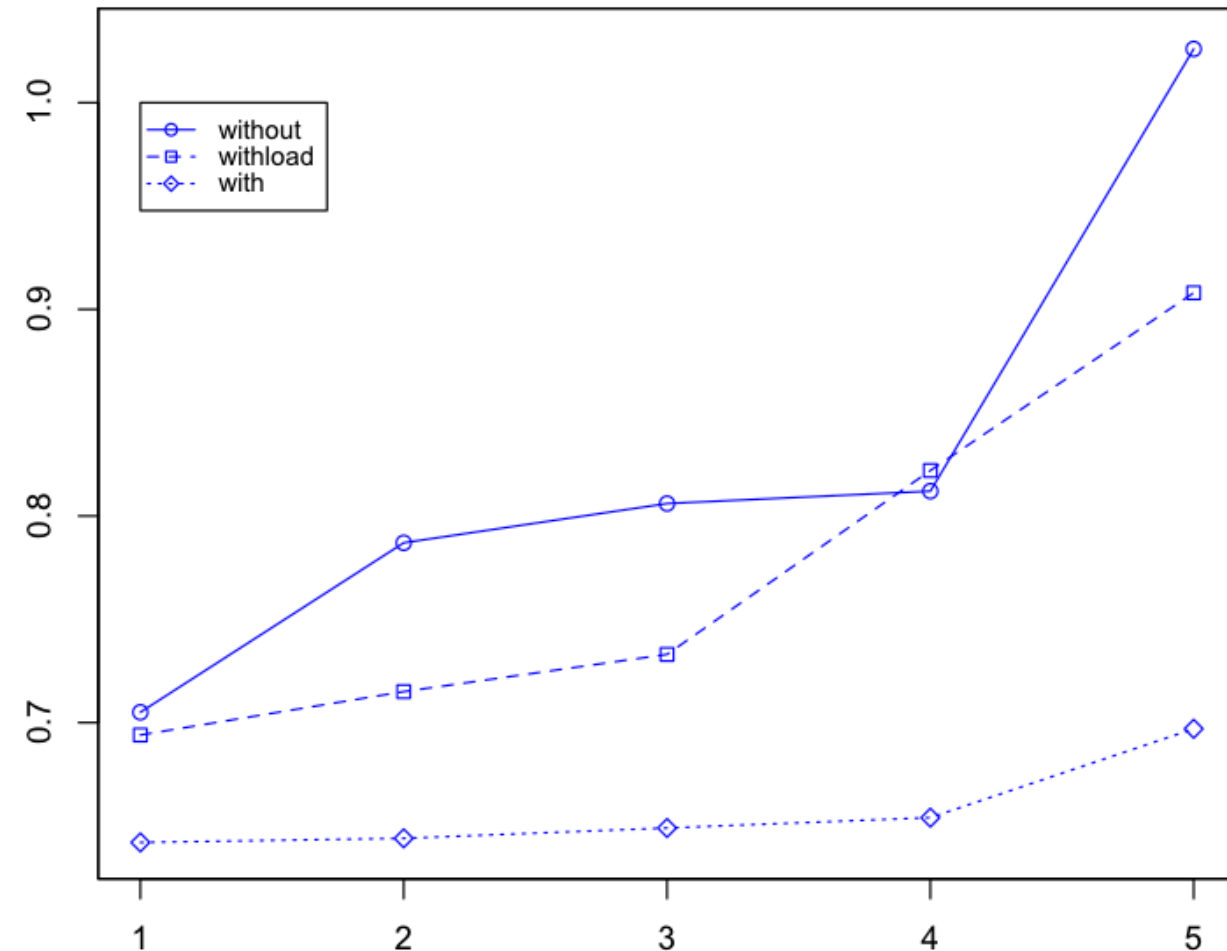- with (dotted line, diamond markers)

```
 1 without <- c(0.705, 0.787, 0.826, 0.804, 1.026, 0.808, 0.809, 0.705, 0.812, 0.800)
 2 withload <- c(0.908, 0.717, 0.824, 0.715, 0.694, 0.822, 0.702, 0.724, 0.812, 0.742)
 3 with <- c(0.67, 0.642, 0.645, 0.650, 0.643, 0.648, 0.644, 0.654, 0.697, 0.654)
 4
 5 plot(without, type="o", col="blue", ylim=c(0.640, 1.03))
 6
 7 box()
 8
 9 lines(withload, type="o", col="red", pch=22, lty=2)
10 lines(with, type="o", col="green", pch=23, lty=3)
11
12 title(main="Page Load Times", col.main="red", font.main=4)
13
14 legend(1, c("without", "withload", "with"), cex=0.8, col=c("blue", "red", "green"), pch=21:23, lty=1:3);
15
```

```
 1 without <- c(0.705, 0.787, 0.806, 0.812, 1.026)
 2 withload <- c(0.694, 0.715, 0.733, 0.822, 0.908)
 3 with <- c(0.642, 0.644, 0.649, 0.654, 0.697)
 4
 5 plot(without, type="o", col="blue", ylim=c(0.640, 1.03))
 6
 7 box()
 8
 9 lines(withload, type="o", col="red", pch=22, lty=2)
10 lines(with, type="o", col="green", pch=23, lty=3)
11
12 title(main="Five Number Summarys", col.main="red", font.main=4)
13
14 legend(1, c("without", "withload", "with"), cex=0.8, col=c("blue", "red", "green"), pch=21:23, lty=1:3);
15
```

Results

Five Number Summarys

- ↗ No HTML
  - ↗ mean load time 0.806s

- ↗ With HTML (no cookie)
  - ↗ mean load time 0.733

- ↗ With HTML (with cookie)
  - ↗ mean load time 0.649s

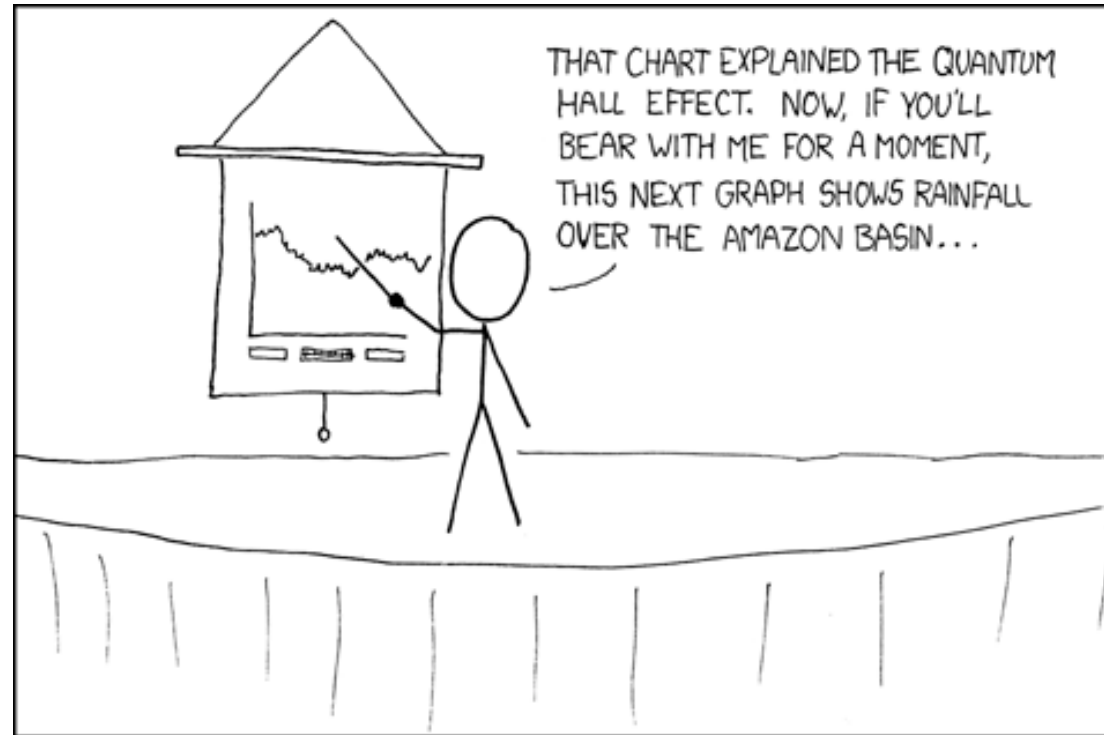- ↗ Speed up for 11063 records
  - ↗ 0.157s

Conclusion

- Yslow – page load times seemed random

- Safari dev tools – made me a believer
    - Sorry FireBug

- R is cooler then MiniTab

- Internet speeds are fast while IO speeds are slow

- HTML5 Web Storage great for saving page state and user data

- Not ready to push whole publications database in to users local storage

- Experiment thoughts
    - Poor sample size
    - Poor amount of results collected
    - Learning tools while collecting results
    - Lack of sleep ☺

Future Work

- Create a test project from scratch, not pieces together parts of current project

- Collect more results from multiple tools

- Collect more data per result (bandwidth, size of data...)

- Review other experiments in similar work

- Use local storage for user data and page state in current projects

THAT CHART EXPLAINED THE QUANTUM HALL EFFECT. NOW, IF YOU'LL BEAR WITH ME FOR A MOMENT, THIS NEXT GRAPH SHOWS RAINFALL OVER THE AMAZON BASIN...

IF YOU KEEP SAYING "BEAR WITH ME FOR A MOMENT", PEOPLE TAKE A WHILE TO FIGURE OUT THAT YOU'RE JUST SHOWING THEM RANDOM SLIDES.

References

- https://addons.mozilla.org/en-US/firefox/addon/yslow/#

- http://www.harding.edu/fmccown/r/

- http://diveintohtml5.info/storage.html

- Countless others found from
  - http://www.google.com