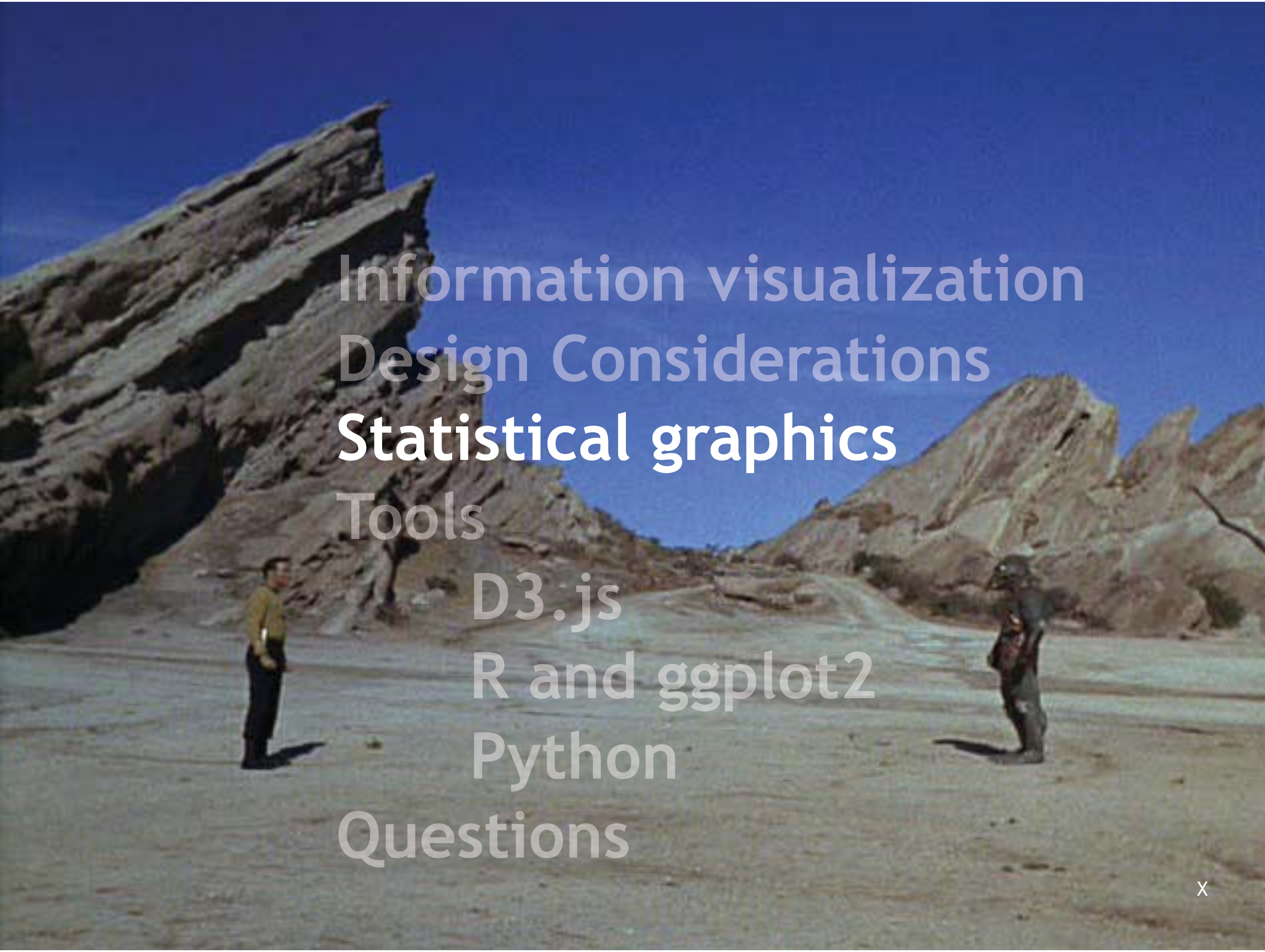
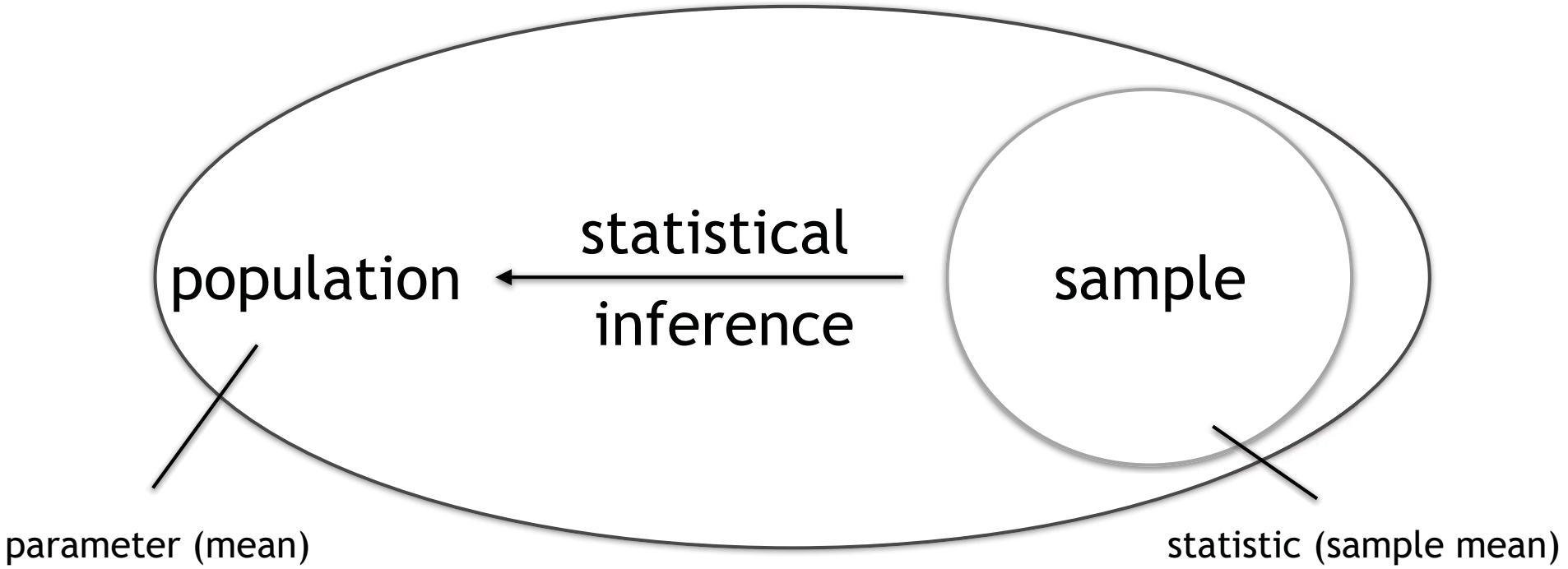

Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions
X


testing
significance
sum
dichotomous
normality
correlation
continuous
thickness
paired
hypothesis
scatter
kurtosis
death
independent
order
medians
assumptions
standard
rank
means
categorical
type
rate
mean
linear
calculate
ordered
ttest
statistically
data
coefficient
normally
test
variable
observations
nominal
reject
statistics
alcohol
median
error
association
variables
tests
ordinal
wilcoxon
nonparametric
regression
normal
parametric
consumption
statistical
proportion
significant

X

Statistics is the science of collecting, **organizing**, **analyzing**, and **interpreting** data in order to make decisions

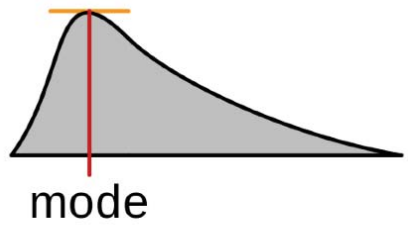


Descriptive statistics summarize data, e.g., one number stands for a group of numbers.

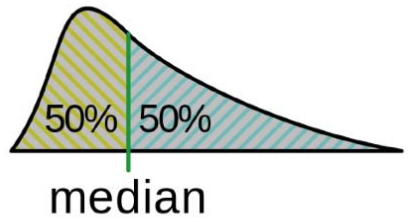
Descriptive vs. inferential statistics

Measures of central tendency & dispersion

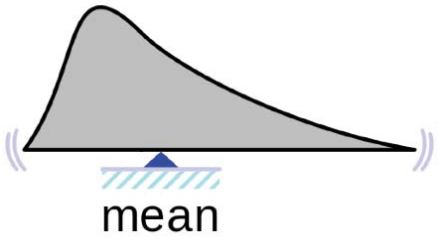
Mode
Most frequently occurring observation(s)
0,1,1,2,2,3,4 (modes = 1 and 2)
Bimodal distribution with two modes or peaks



Median (useful for skewed distributions or with outliers)
Robust to outliers
Middle-most observation of ordered data
0,1,1,2,2,3,4 (median = 2)



Mean (best for symmetric distributions without outliers)
Sensitive to outliers
Sum of all observations divided by n



By Cmglee - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=38969094

Range
Difference between largest and smallest observations in a set.

Variance
Way to ascertain how individual values are located around the mean.

$$s^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \overline{x})^2$$

X

Measures of order/rank

Order statistics

The *k*th **order statistic** of a sample is equal to its *k*th-smallest value

- first order statistic: $X_{(1)} = \min\{ X_1, \ldots, X_n \}$
- *n*th order statistic: $X_{(n)} = \max\{ X_1, \ldots, X_n \}.$
- sample range: $\mathrm{Range}\{ X_1, \ldots, X_n \} = X_{(n)} - X_{(1)}.$

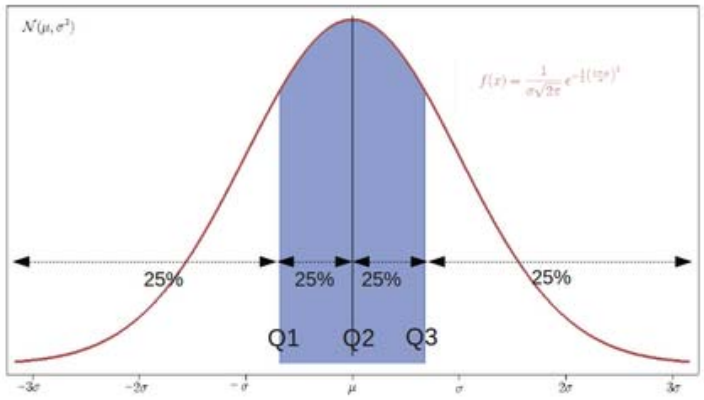
Quantiles

Cutpoints dividing the distribution or observations into intervals of equal probability

p^{th} percentile: p% of the observations fall at or below, and (100 - p)% above it.

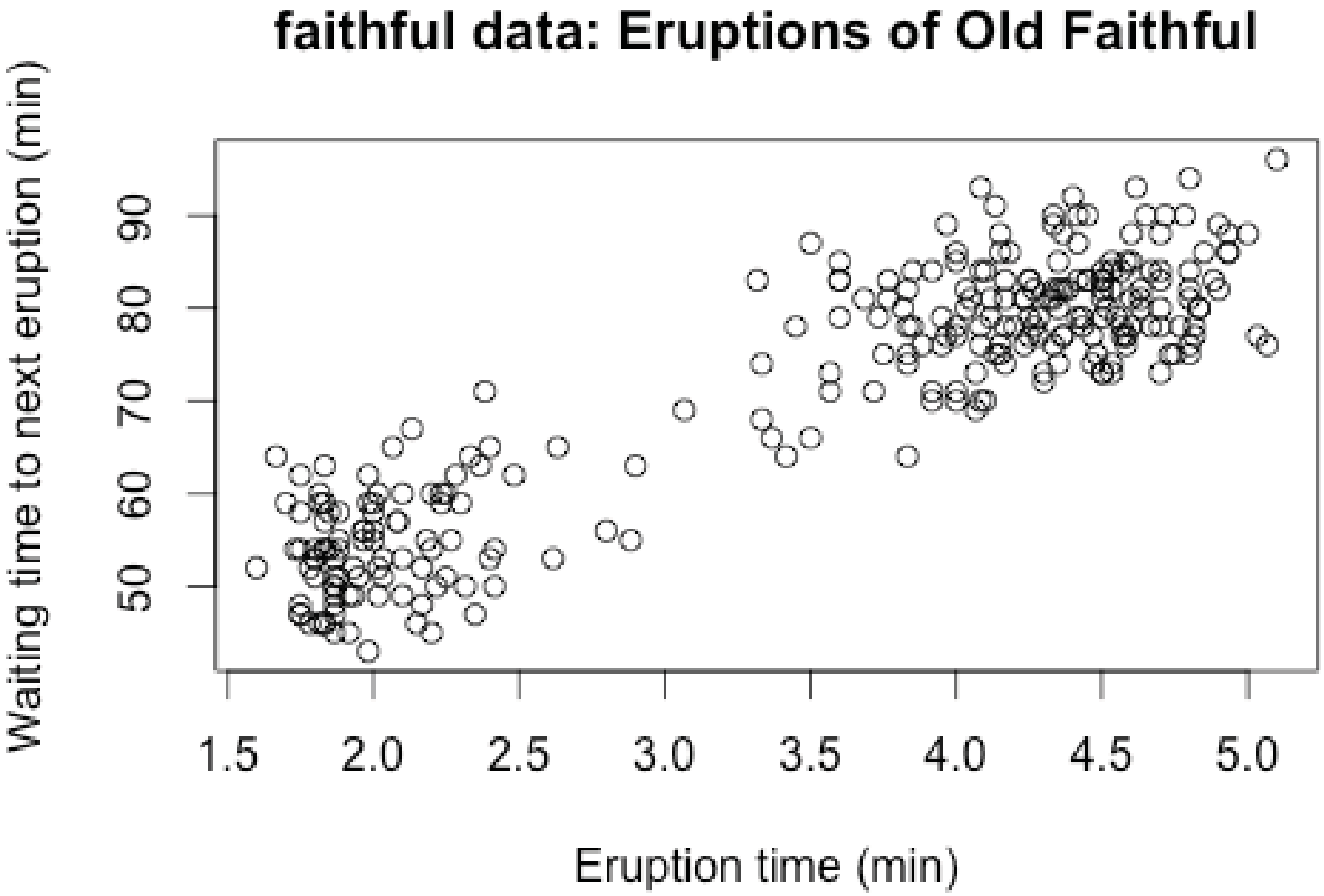
Robust to outliers.

- **Median** : 2-quantile
- **Quartiles**: 4-quantiles
- **Interquartile range**: IQR = Q3 − Q1
- **Quintiles:** 5-quantiles

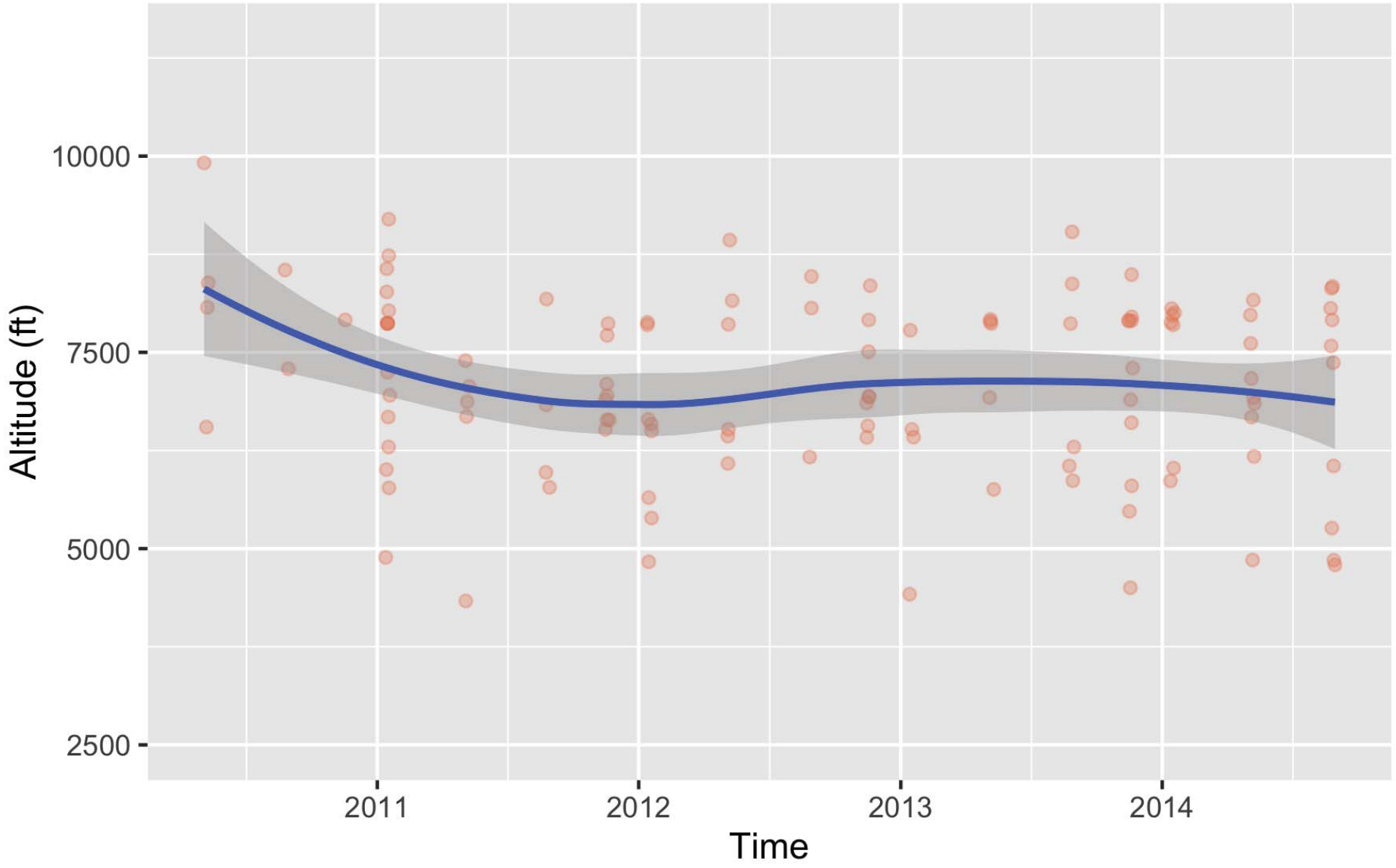


X

Viewing data distribution with scatterplots



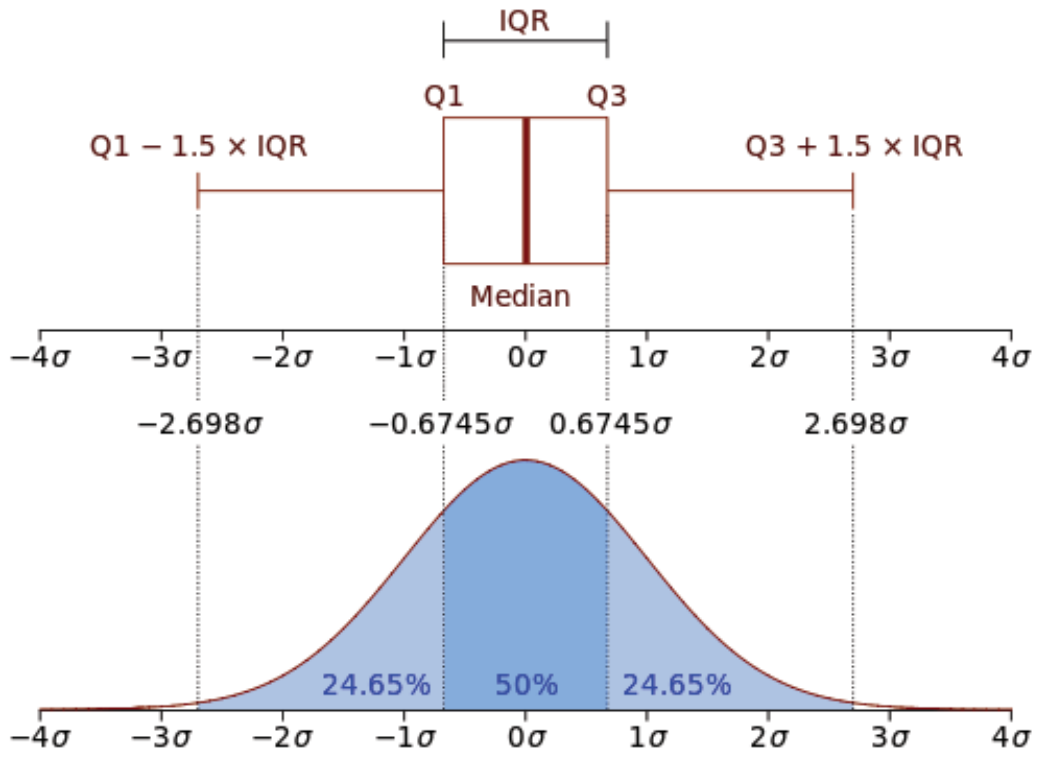
Visualizing Regression with scatterplots



X

Box plot / box-and-whisker plot [Tuckey 1969]

Shows IQR median, skew, tails, outliers



Quartiles: cutpoints at Q1, Q2 and Q3

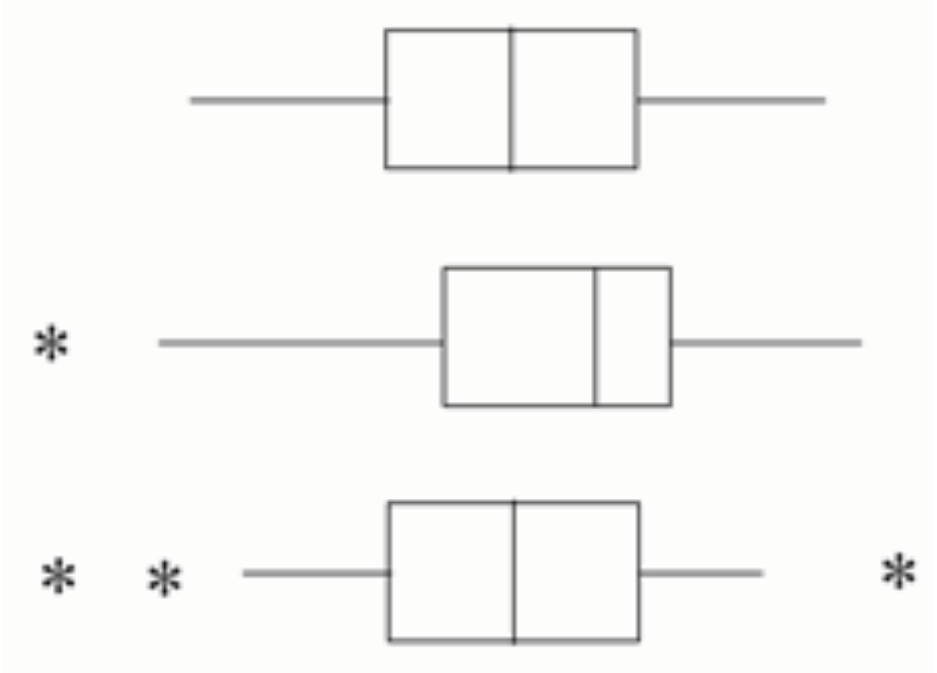
Q1: 25% of observations below and 75% above Q1

Median or Q2: 50% of observations below Q2

Q3: 75% of the observations below and 25% above Q3

IQR: inter-quartile range containing 50% of the observations

Distributions and box plots



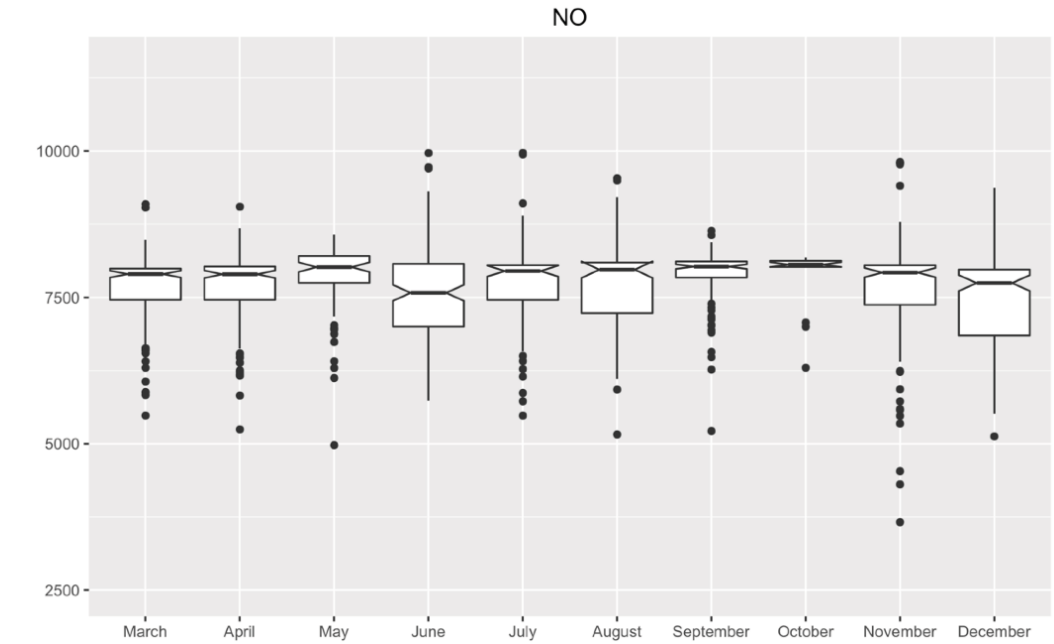
normal distribution

skew to the left

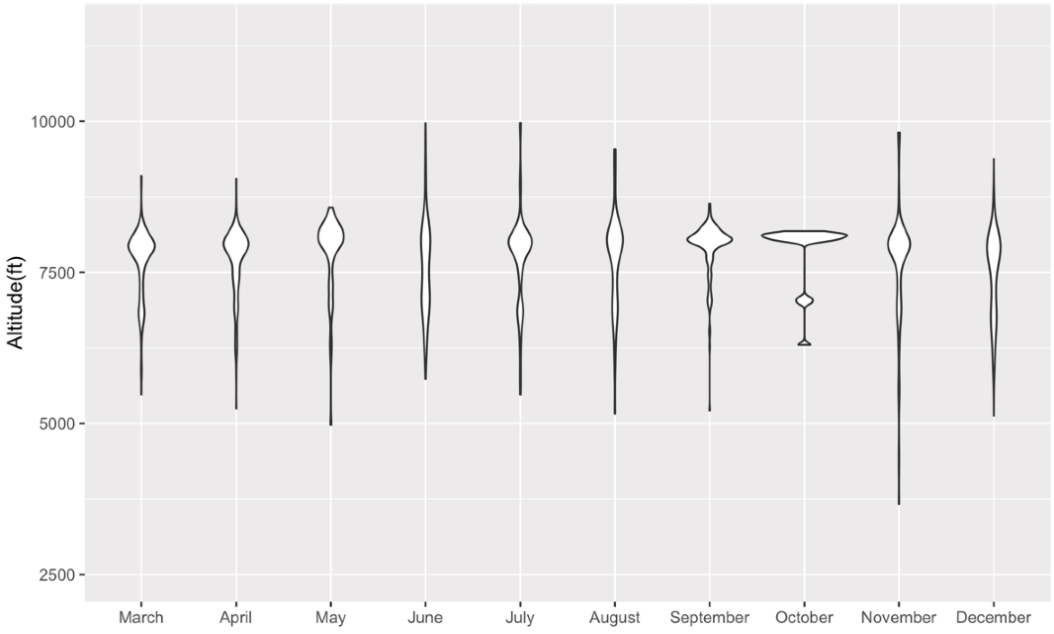
centered with some outliers

Distributions and box plots

Box plot with **notches**



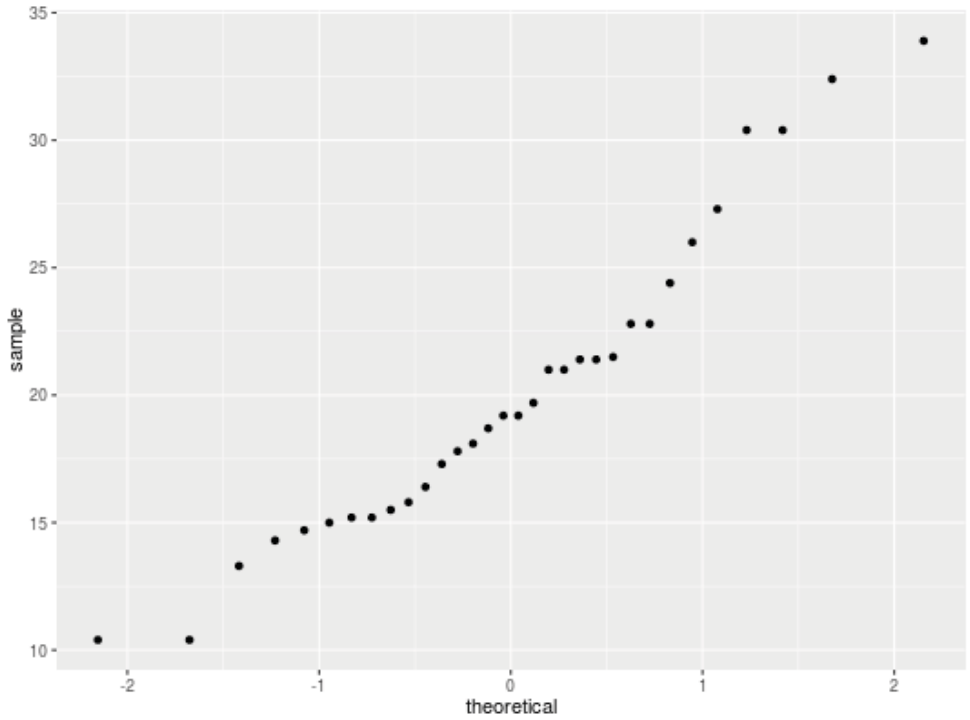
Violin plot box plot with probability density



X

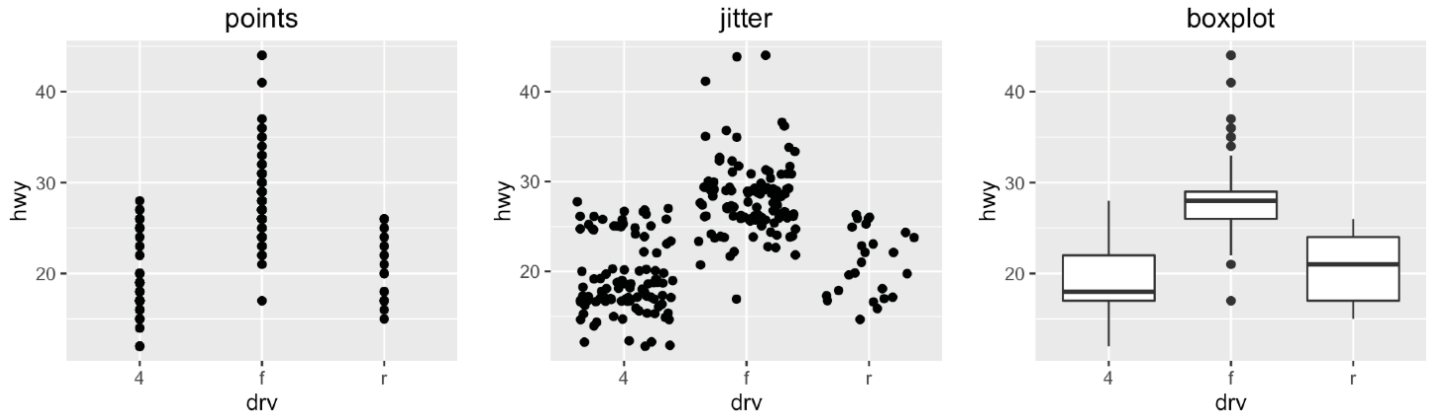
Comparing distributions with Q-Q plots

Quantile-quantile plot (Q-Q plot): plot quantiles against each other

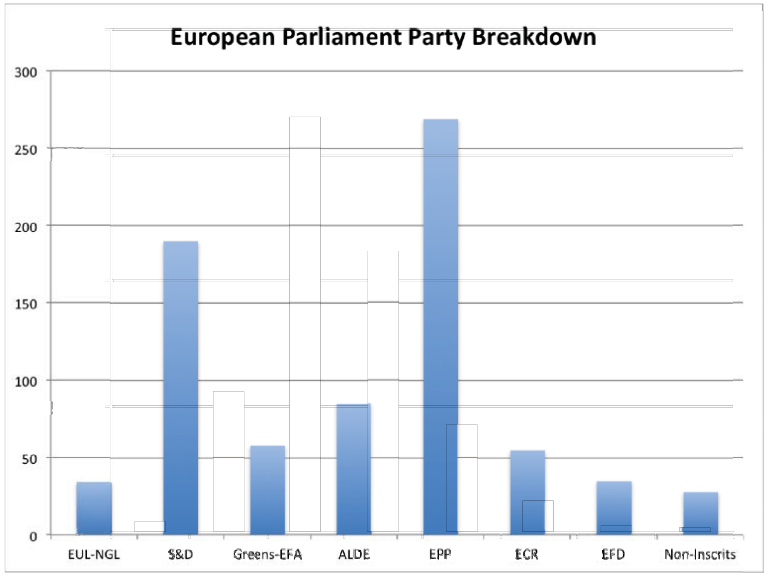


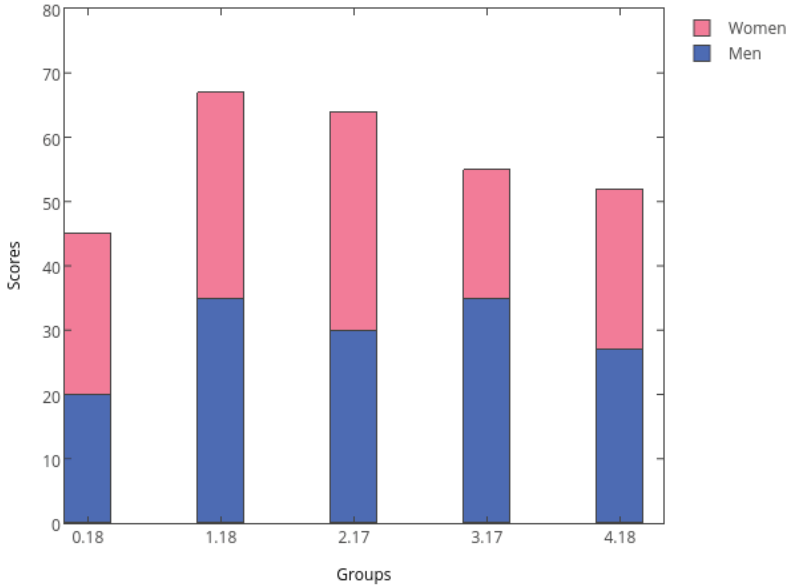
Compare to a normal distribution: linearity suggests data is normally distributed.

Box plots like showing the data



Comparing frequency of a categorical variable





Categories

Use bar charts to compare categorical variables

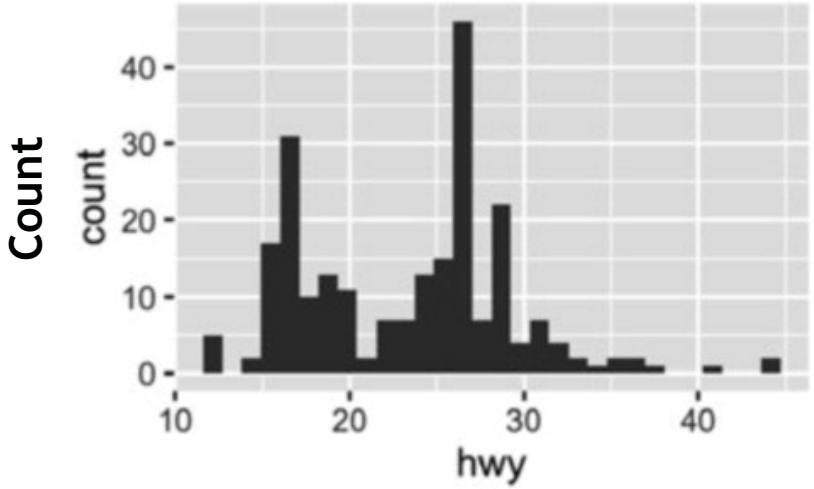
Use stacked bar graphs to show frequency of subgroups

Bars are separated not to imply continuity of categories!
Bars can be re-arranged!

X

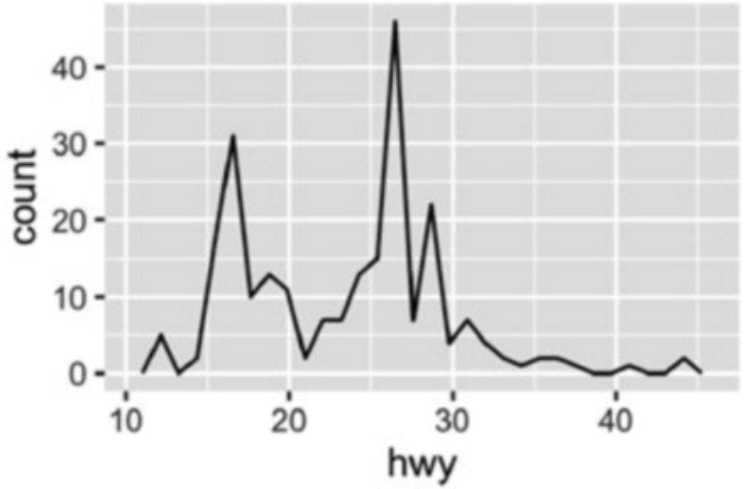
Visualizing the distribution of a continuous variable

Histograms
Experiment with the bin-size!



Bins (intervals)

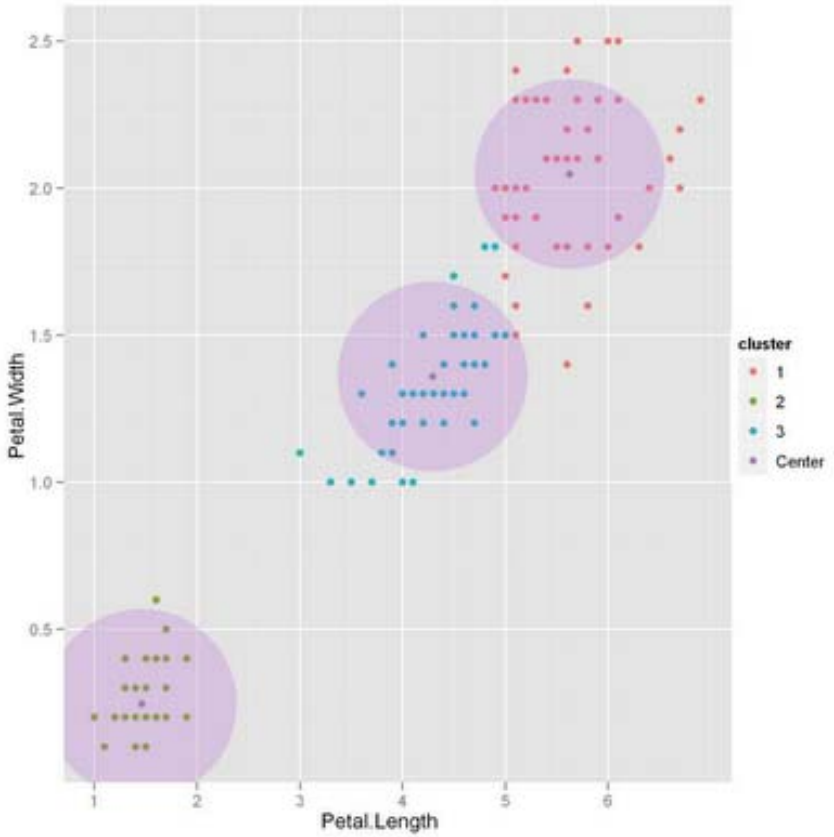
Frequency polygons of frequency or relative frequency



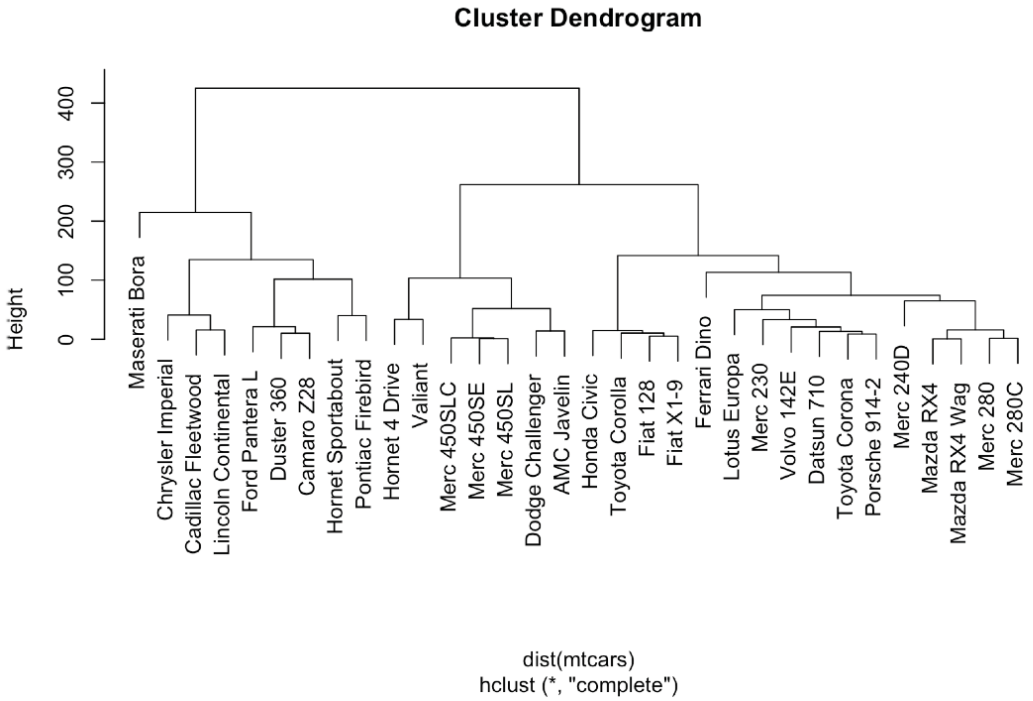
Bars are NOT separated to imply continuity of categories!
Bars CANNOT be re-arranged!

X

Visualizing clusters (multivariate data)



Annotated scatterplot

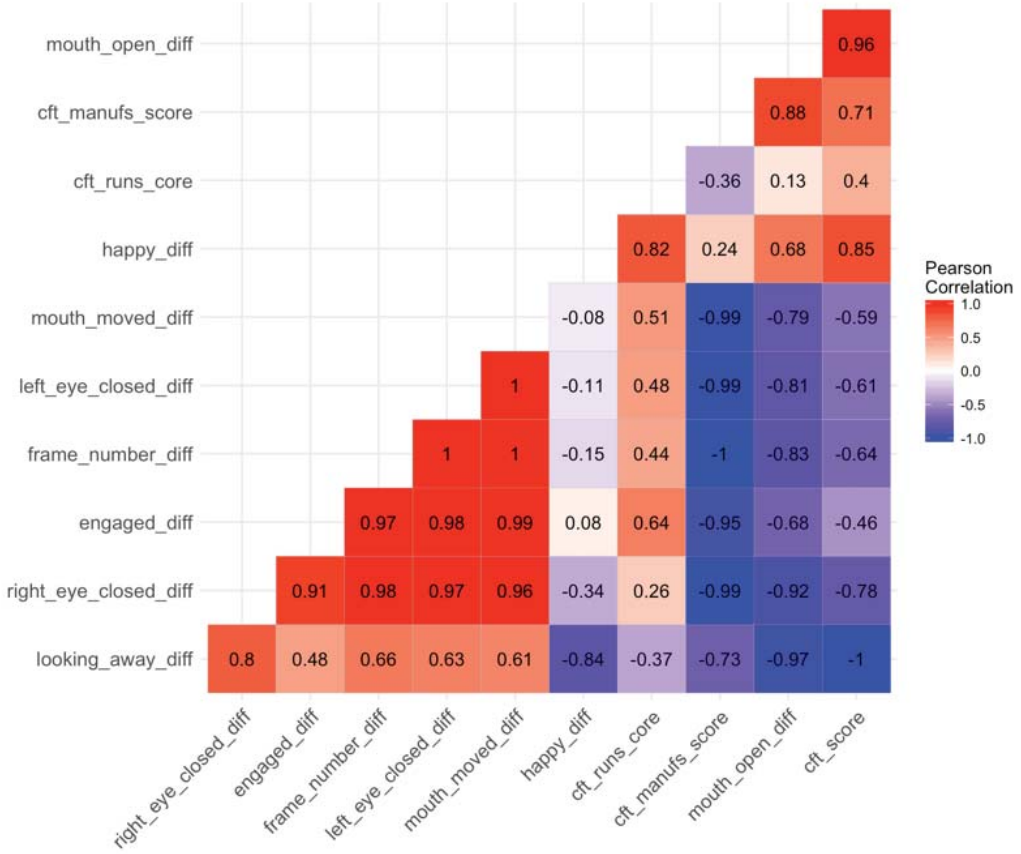


Hierarchical clustering dendrogram

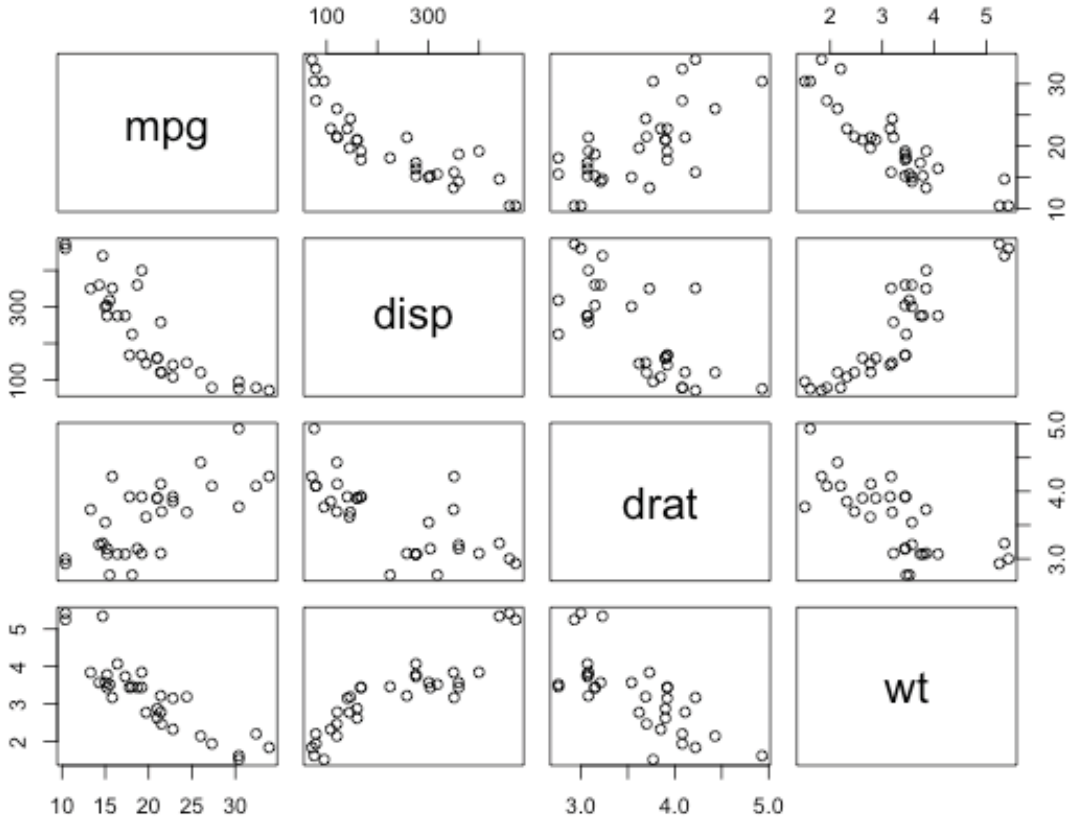
Visualizing correlations

Scatterplot of Pearson's correlation coefficient

Heatmap of Pearson's correlation

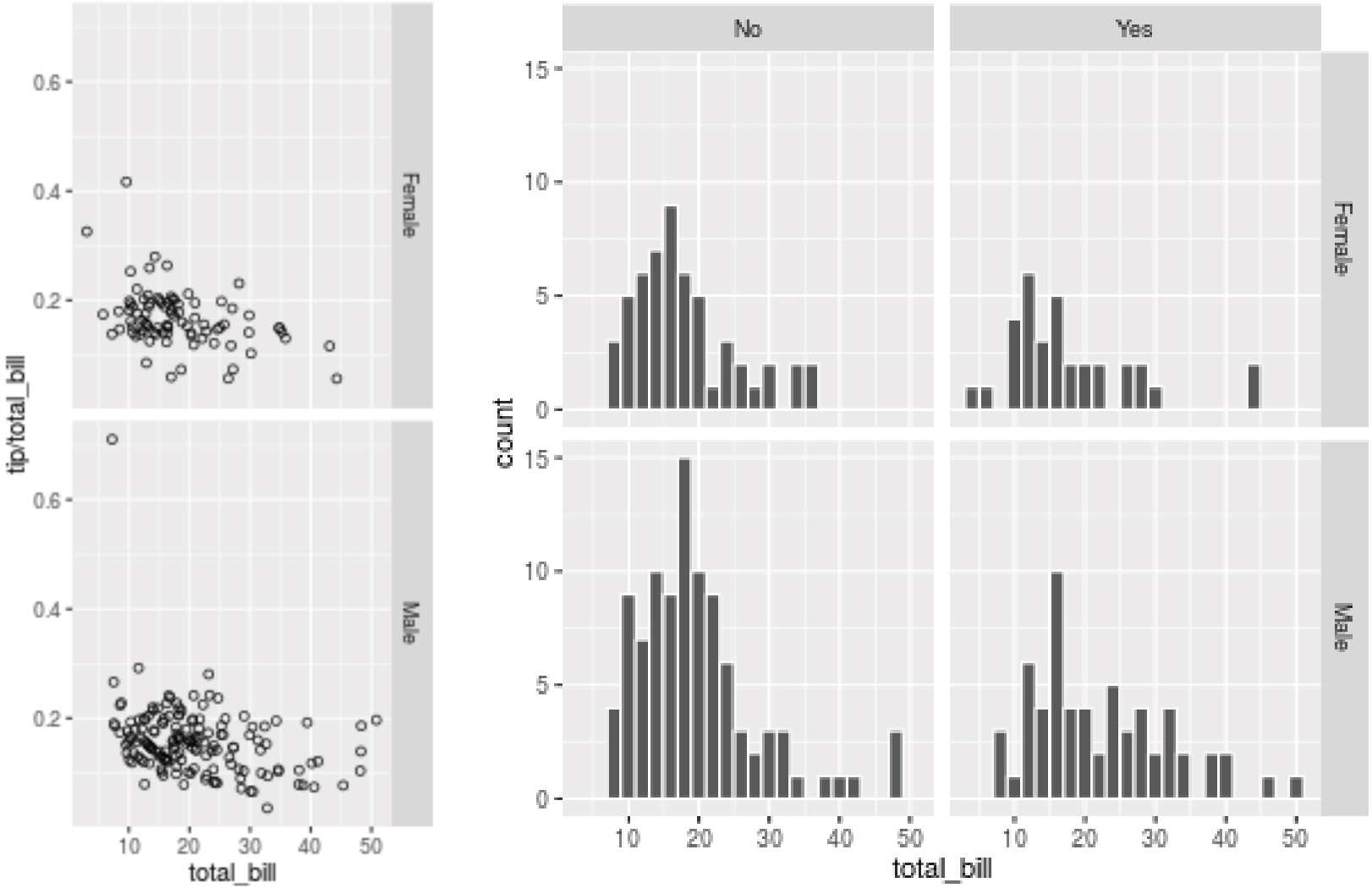


Visualizing correlations



Scatterplot matrix: showing correlations between variables

Visualizing multivariate datasets

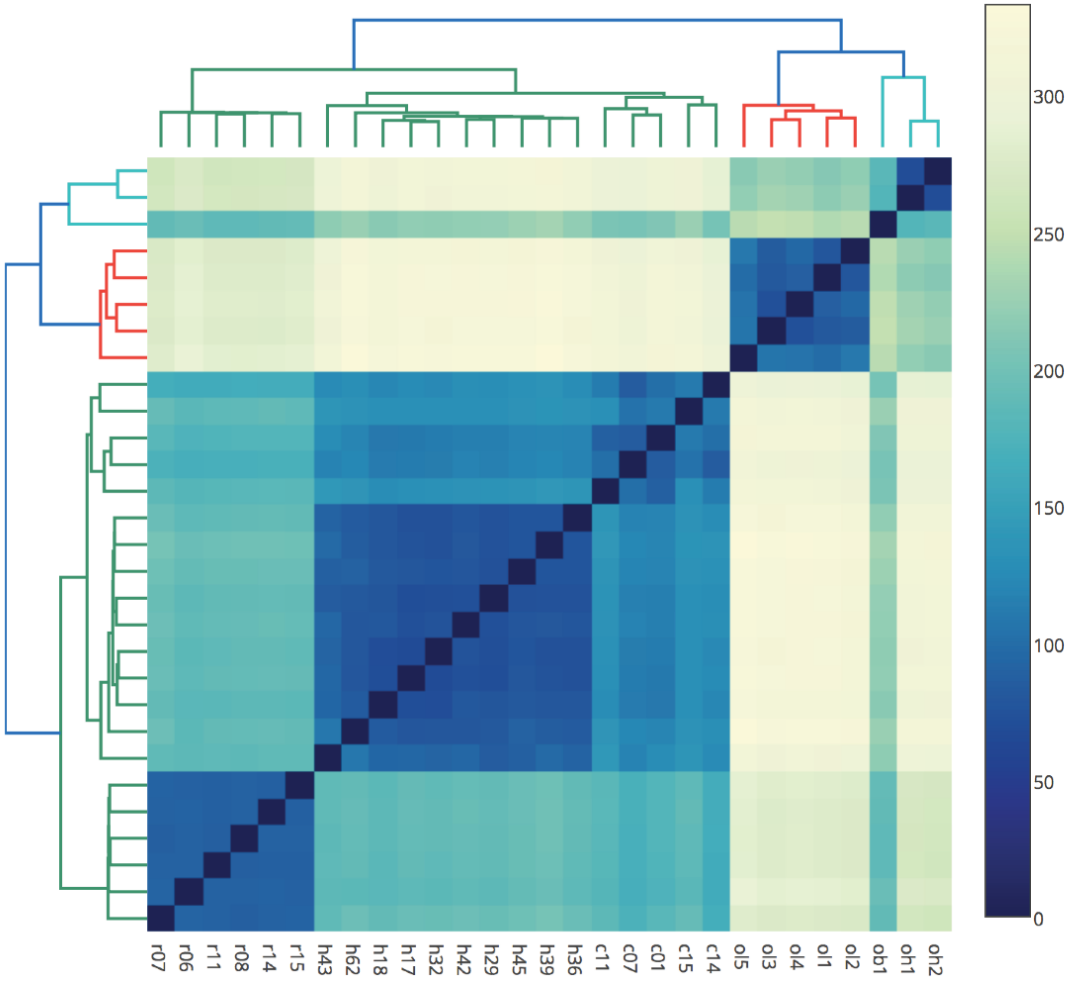


Trellis plot

Faceting/conditioning/latticing/trellising is a technique to break up the data into subsets and display as **small multiples.**

X

Visualizing multivariate datasets



Heatmap and dendrogram

Correlations and hierarchical information across variables

X


Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions
X

Visualization tools [Heer]

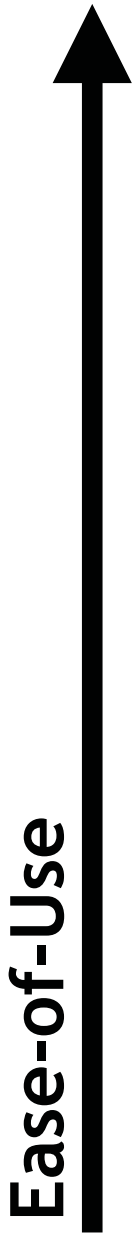


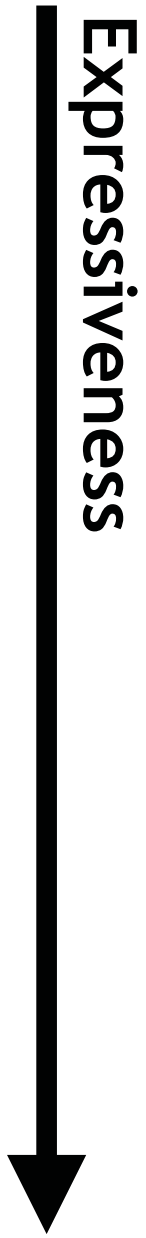
Chart Typologies
Excel,Google Charts, Plotly, Matplotlib

Visual Analysis Grammars
VizQL (Tableau), ggplot2

Visualization Grammars
Protovis, D3.js

Component Architectures
Prefuse, Flare, Improvise, VTK

Graphics APIs
Processing (P5.js), WebGL, OpenGL

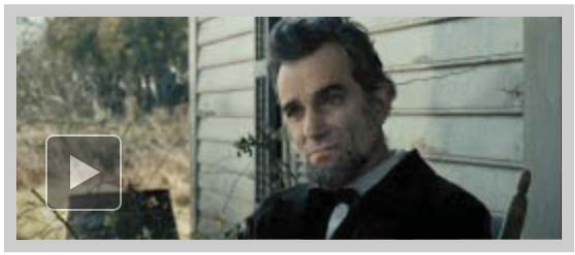



Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions

Interactive WEB based infographics

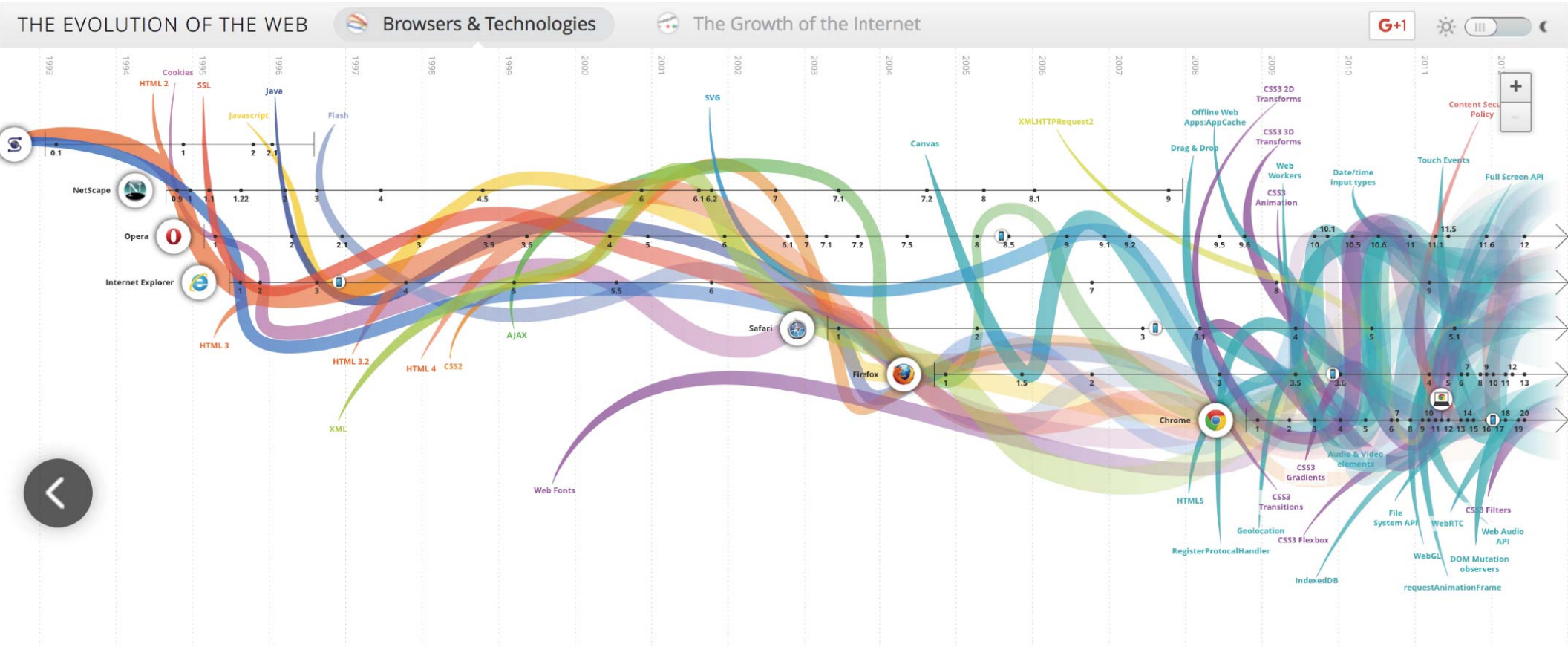
Lincoln

The “Lincoln” trailer is more like the typical teaser than a trailer, according to Stephen Garrett, who owns Jump Cut, a trailer house that specializes in foreign, independent and documentary films. While trailers often focus on plot or character descriptions, teasers establish the mood and tone of a film. Teasers “don’t have to be chronological,” Mr. Garrett said.



http://www.nytimes.com/interactive/2013/02/19/movies/awardsseason/oscar-trailers.html

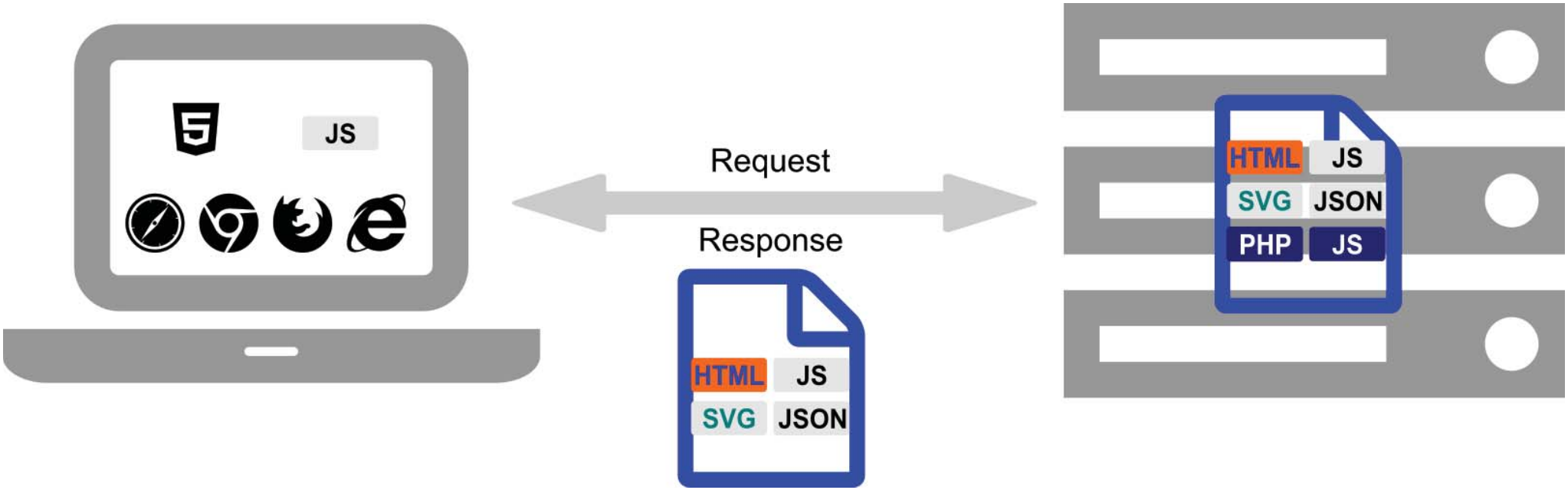
X



http://www.evolutionoftheweb.com

X

The Web




SVG

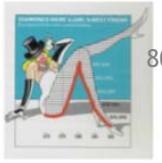







BIG
DATA
ANALYTICS
TECHNOLOGIES
STORAGE
INFORMATION
INTERNET
SOFTWARE
SHARED
COMPLEX
SETS

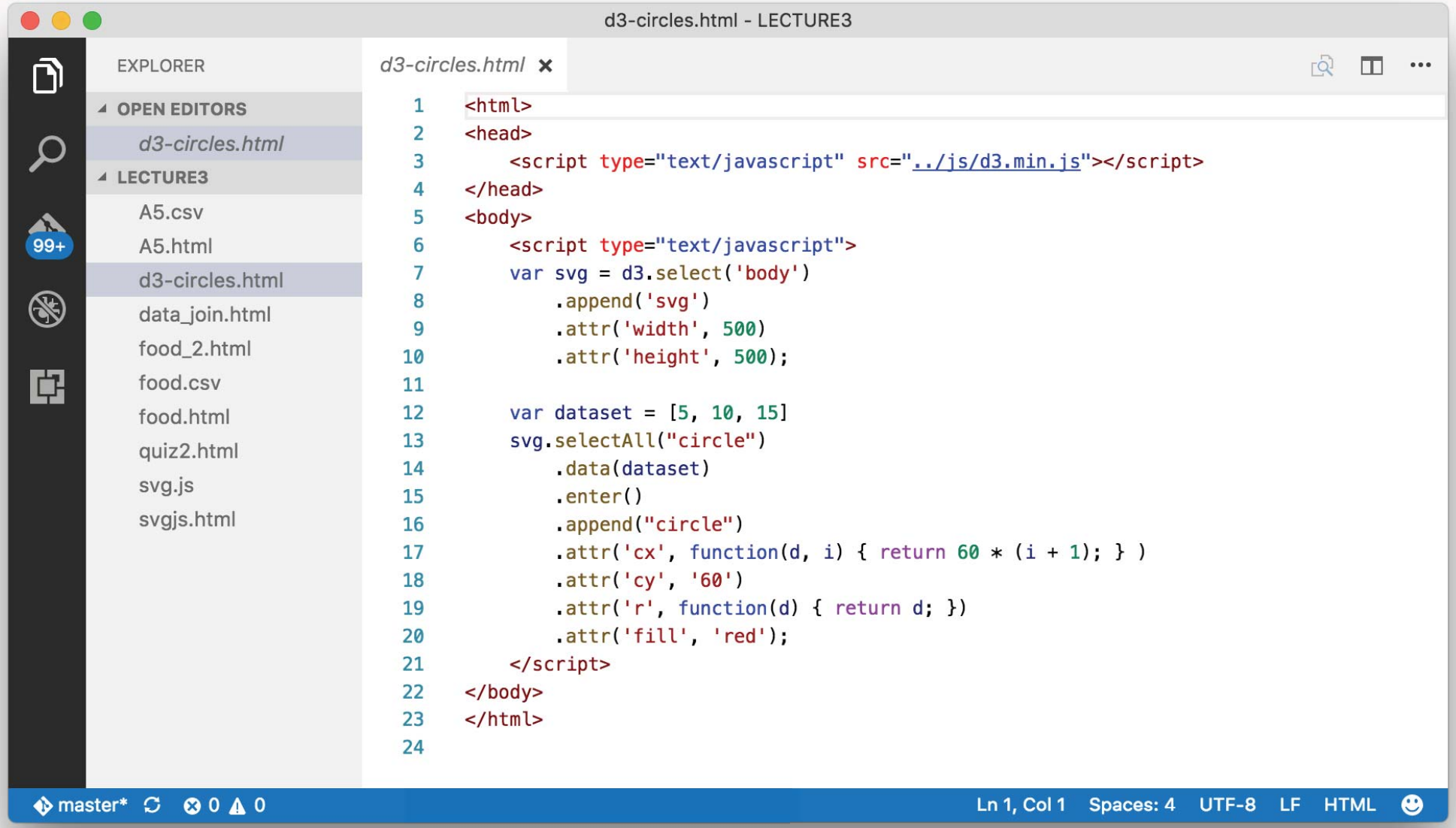




X

```
// https://d3js.org Version 4.2.5. Copyright 2016 Mike Bostock.
(function(t,n){"object"==typeof exports&&"undefined"!=typeof module?n(exports):"function"==typeof define&&define.amd?define(
["exports"],n):n(t.d3=t.d3||{})})(this,function(t){"use strict";function n(t){return function(n,e){return ms(t(n),e)}}function e(t,n,
e){var r=Math.abs(n-t)/Math.max(0,e),i=Math.pow(10,Math.floor(Math.log(r)/Math.LN10)),o=r/i;return o>=Rs?i*=10:o>=Us?i*=5:o>=Ds&&
(i*=2),n<t?-i:i}function r(t){return t.length}function i(){}function o(t,n){var e=new i;if(t instanceof i)t.each(function(t,n){e.set
(n,t)});else if(Array.isArray(t)){var r,o=-1,u=t.length;if(null==n)for(;++o<u;)e.set(o,t[o]);else for(;++o<u;)e.set(n(r=t[o],o,t),r)}
else if(t)for(var a in t)e.set(a,t[a]);return e}function u(){return{}}function a(t,n,e){t[n]=e}function c(){return o()}function s(t,
n,e){t.set(n,e)}function f(){}function l(t,n){var e=new f;if(t instanceof f)t.each(function(t){e.add(t)});else if(t){var r=-1,
i=t.length;if(null==n)for(;++r<i;)e.add(t[r]);else for(;++r<i;)e.add(n(t[r],r,t))}return e}function h(t){return+t}function p(t)
{return t*t}function d(t){return t*(2-t)}function v(t){return((t*=2)<=1?t*t:--t*(2-t)+1)/2}function _(t){return t*t*t}function y(t)
{return--t*t*t+1}function g(t){return((t*=2)<=1?t*t*t:(t-=2)*t*t+2)/2}function m(t){return 1-Math.cos(t*xf)}function x(t){return
Math.sin(t*xf)}function b(t){return(1-Math.cos(mf*t))/2}function w(t){return Math.pow(2,10*t-10)}function M(t){return 1-Math.pow(2,
-10*t)}function T(t){return((t*=2)<=1?Math.pow(2,10*t-10):2-Math.pow(2,10-10*t))/2}function N(t){return 1-Math.sqrt(1-t*t)}function k
(t){return Math.sqrt(1- --t*t)}function S(t){return((t*=2)<=1?1-Math.sqrt(1-t*t):Math.sqrt(1-(t-=2)*t)+1)/2}function A(t){return 1-E
(1-t)}function E(t){return(t=+t)<bf?Cf*t*t:t<Mf?Cf*(t-=wf)*t+Tf:t<kf?Cf*(t-=Nf)*t+Sf:Cf*(t-=Af)*t+Ef}function C(t){return((t*=2)<=1?
1-E(1-t):E(t-1)+1)/2}function z(t,n){return t[0]-n[0]||t[1]-n[1]}function P(t){for(var n=t.length,e=[0,1],r=2,i=2;i<n;++i){for(;r>1&&
jf(t[e[r-2]],t[e[r-1]],t[i])<=0;)--r;e[r++]=i}return e.slice(0,r)}function q(){this._x0=this._y0=this._x1=this._y1=null,this._=[]}
function L(){return new q}function R(t,n,e,r){if(isNaN(n)||isNaN(e))return t;var i,o,u,a,c,s,f,l,h,p=t._root,d={data:r},v=t._x0,
_=t._y0,y=t._x1,g=t._y1;if(!p)return t._root=d,t;for(;p.length;)if((s=n>=(o=(v+y)/2))?v=o:y=o,(f=e>=(u=(_+g)/2))?_=u:g=u,i=p,!(p=p
[l=f<<1|s]))return i[l]=d,t;if(a=+t._x.call(null,p.data),c=+t._y.call(null,p.data),n===a&&e===c)return d.next=p,i?i[l]=d:t._root=d,t;
do i=i?i[l]=new Array(4):t._root=new Array(4),(s=n>=(o=(v+y)/2))?v=o:y=o,(f=e>=(u=(_+g)/2))?_=u:g=u;while((l=f<<1|s)===(h=(c>=u)<<1|
a>=o));return i[h]=p,i[l]=d,t}function U(t){var n,e,r,i,o=t.length,u=new Array(o),a=new Array(o),c=1/0,s=1/0,f=-(1/0),l=-(1/0);for
(e=0;e<o;++e)isNaN(r=+this._x.call(null,n=t[e]))||isNaN(i=+this._y.call(null,n))||(u[e]=r,a[e]=i,r<c&&(c=r),r>f&&(f=r),i<s&&(s=i),
i>l&&(l=i));for(f<c&&(c=this._x0,f=this._x1),l<s&&(s=this._y0,l=this._y1),this.cover(c,s).cover(f,l),e=0;e<o;++e)R(this,u[e],a[e],t
[e]);return this}function D(t){for(var n=0,e=t.length;n<e;++n)this.remove(t[n]);return this}function O(t){return t[0]}function F(t)
{return t[1]}function I(t,n,e){var r=new Y(null==n?O:n,null==e?F:e,NaN,NaN,NaN,NaN);return null==t?r:r.addAll(t)}function Y(t,n,e,r,
i,o){this._x=t,this._y=n,this._x0=e,this._y0=r,this._x1=i,this._y1=o,this._root=void 0}function B(t){for(var n={data:t.data},e=n;
t=t.next;)e=e.next={data:t.data};return n}function j(t){if(!(t>=1))throw new Error;this._size=t,this._call=this._error=null,
this._tasks=[],this._data=[],this._waiting=this._active=this._ended=this._start=0}function H(t){if(!t._start)try{X(t)}catch(n){if
(t._tasks[t._ended+t._active-1])W(t,n);else if(!t._data)throw n}}function X(t){for(;t._start=t._waiting&&t._active<t._size;){var
n=t._ended+t._active,e=t._tasks[n],r=e.length-1,i=e[r];e[r]=V(t,n),--t._waiting,++t._active,e=i.apply(null,e),t._tasks[n]&&(t._tasks
[n]=e||hl)}}function V(t,n){return function(e,r){t._tasks[n]&&(--t._active,++t._ended,t._tasks[n]=null,null==t._error&&(null!=e?W(t,
e):(t._data[n]=r,t._waiting?H(t):$(t))))}}function W(t,n){var e,r=t._tasks.length;for(t._error=n,t._data=void 0,t._waiting=NaN;
--r>=0;)if((e=t._tasks[r])&&(t._tasks[r]=null,e.abort))try{e.abort()}catch(t){}t._active=NaN,$(t)}function $(t){if(!t._active&&
t._call){var n=t._data;t._data=void 0,t._call(t._error,n)}}function Z(t){return new j(arguments.length?+t:1/0)}function G(t){return
t.innerRadius}function J(t){return t.outerRadius}function Q(t){return t.startAngle}function K(t){return t.endAngle}function tt(t)
{return t&&t.padAngle}function nt(t){return t>=1?_l:t<=-1?-_l:Math.asin(t)}function et(t,n,e,r,i,o,u,a){var c=e-t,s=r-n,f=u-i,l=a-o,
h=(f*(n-o)-l*(t-i))/(l*c-f*s);return[t+h*c,n+h*s]}function rt(t,n,e,r,i,o,u){var a=t-e,c=n-r,s=(u?o:-o)/Math.sqrt(a*a+c*c),f=s*c,
l=-s*a,h=t+f,p=n+l,d=e+f,v=r+l,_=(h+d)/2,y=(p+v)/2,g=d-h,m=v-p,x=g*g+m*m,b=i-o,w=h*v-d*p,M=(m<0?-1:1)*Math.sqrt(Math.max(0,b*b*x-w*w)
),T=(w*m-g*M)/x,N=(-w*g-m*M)/x,k=(w*m+g*M)/x,S=(-w*g+m*M)/x,A=T-_,E=N-y,C=k-_,z=S-y;return A*A+E*E>C*C+z*z&&(T=k,N=S),{cx:T,cy:N,
x01:-f,y01:-l,x11:T*(i/b-1),y11:N*(i/b-1)}}function it(t){this._context=t}function ot(t){return t[0]}function ut(t){return t[1]}
```

```
// https://d3js.org Version 4.2.5. Copyright 2016 Mike Bostock.
(function (global, factory) {
  typeof exports === 'object' && typeof module !== 'undefined' ? factory(exports) :
  typeof define === 'function' && define.amd ? define(['exports'], factory) :
  (factory((global.d3 = global.d3 || {})));
}(this, (function (exports) { 'use strict';

var version = "4.2.5";

var ascending = function(a, b) {
  return a < b ? -1 : a > b ? 1 : a >= b ? 0 : NaN;
}

var bisector = function(compare) {
  if (compare.length === 1) compare = ascendingComparator(compare);
  return {
    left: function(a, x, lo, hi) {
      if (lo == null) lo = 0;
      if (hi == null) hi = a.length;
      while (lo < hi) {
        var mid = lo + hi >>> 1;
        if (compare(a[mid], x) < 0) lo = mid + 1;
        else hi = mid;
      }
      return lo;
    },
    right: function(a, x, lo, hi) {
      if (lo == null) lo = 0;
      if (hi == null) hi = a.length;
      while (lo < hi) {
        var mid = lo + hi >>> 1;
        if (compare(a[mid], x) > 0) hi = mid;
        else lo = mid + 1;
      }
      return lo;
    }
  };
}

function ascendingComparator(f) {
  return function(d, x) {
```



Visual Studio Code

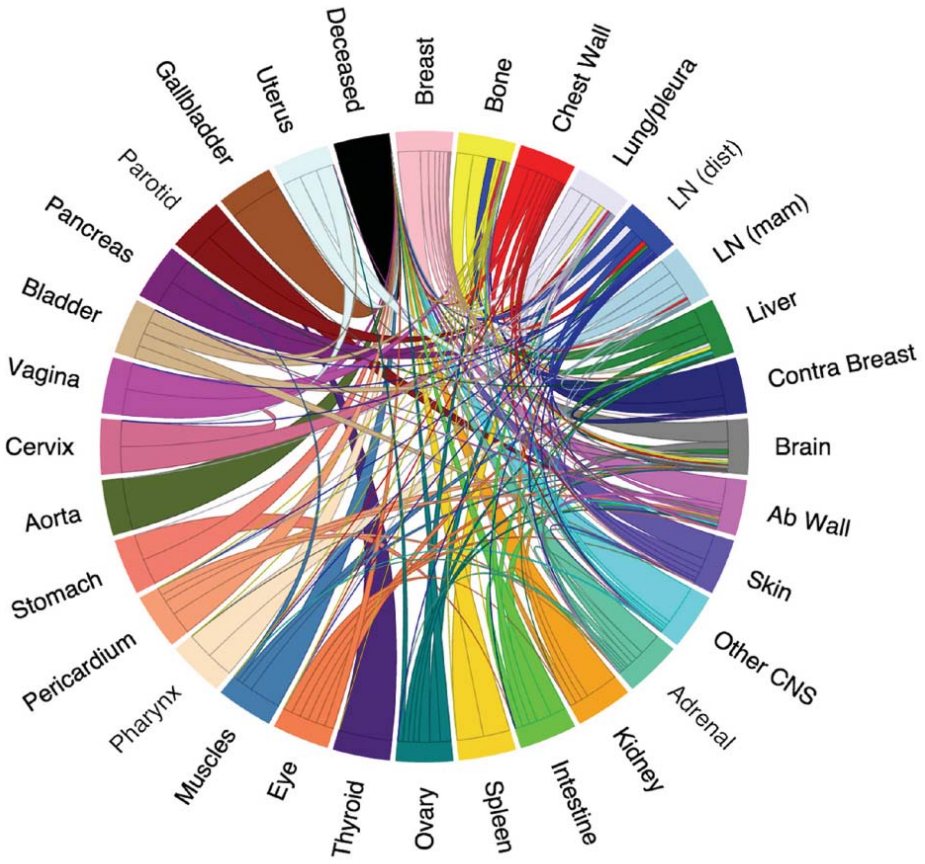
https://code.visualstudio.com/

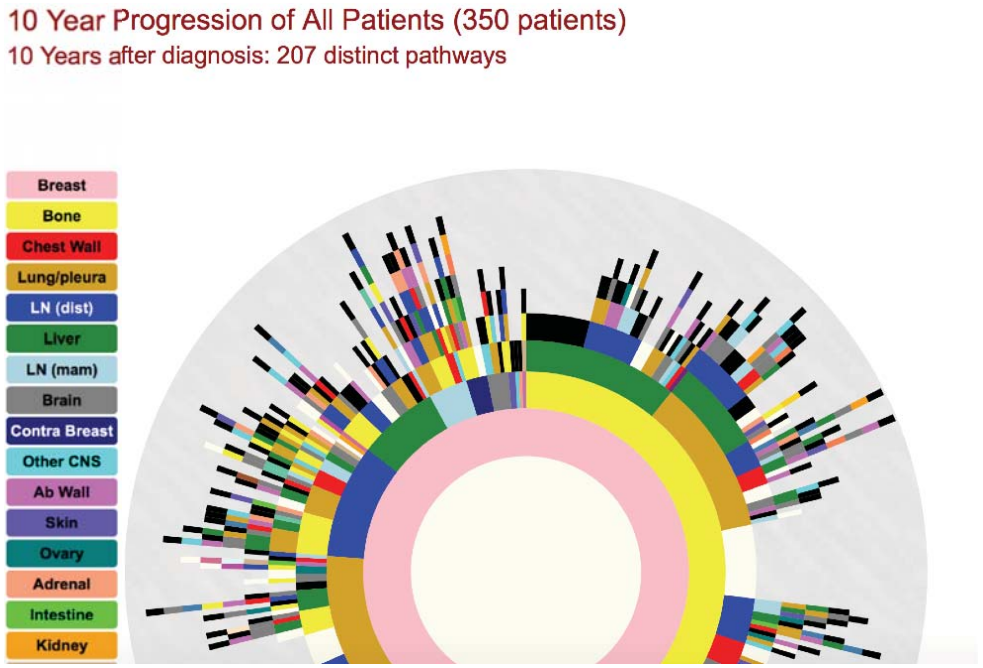
https://d3js.org

Declarative programming (you say what not how)
Data Join central mechanism

```
<html>
<head>
    <script type="text/javascript" src="http://d3js.org/d3.v4.min.js"></script>
</head>
<body>
    <script type="text/javascript">
    var svg = d3.select('body')
        .append('svg')
        .attr('width', 500)
        .attr('height', 500);

    var dataset = [5, 10, 15]
    svg.selectAll("circle")
        .data(dataset)
        .enter()
        .append("circle")
        .attr('cx', function(d, i) { return 60 * (i + 1); } )
        .attr('cy', '60')
        .attr('r', function(d) { return d; })
        .attr('fill', 'red')·
    </script>
</body>
</html>
```

https://d3js.org





http://kuhn.usc.edu/breast_cancer

Chord diagram layout

X


Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions

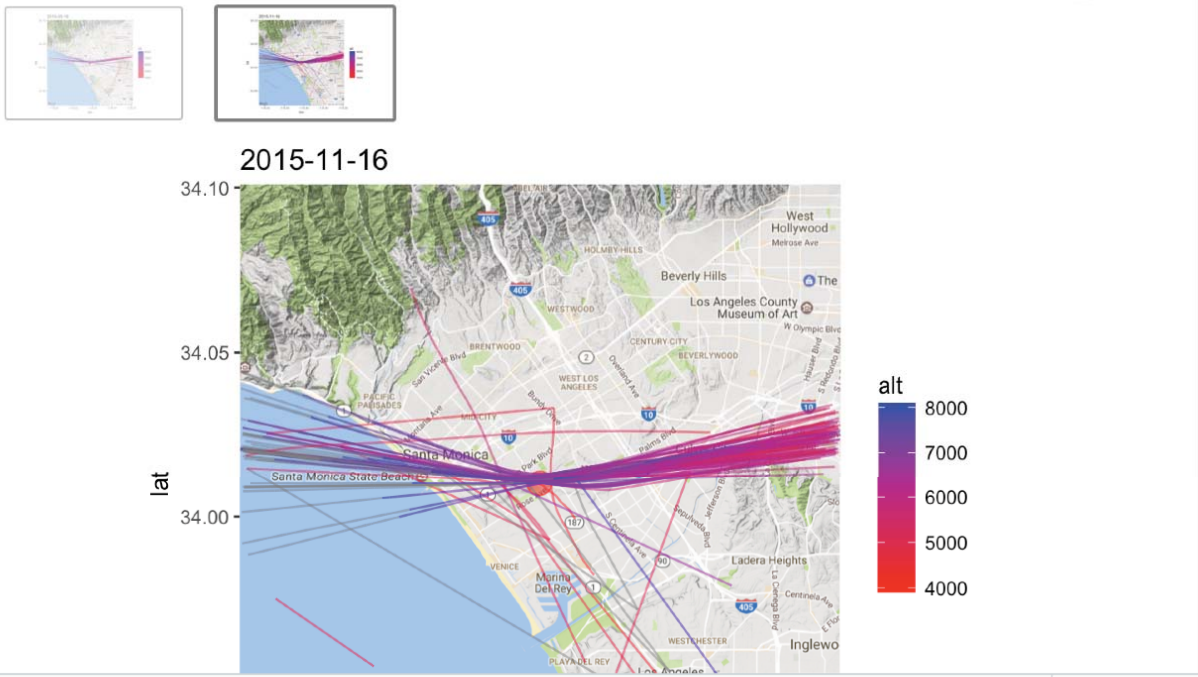
- Open source
- Statistical computing language
- Graphics programming language
- Software environment

- RStudio: https://www.rstudio.com
- Manage R projects
- Explore and analyze data
- Generate plots
- Document & publish

NorthWindArrivals.R | NorthWindArrivalsBook.Rmd*

Preview | Insert | Run

```
72
73  p2 <- ggmap(map.google) +
74    geom_point(data = SMO, aes(x = lon, y = lat), color="red", size=5, alpha=.5) +
75    geom_path(data = df2_smo, aes(x = lon, y = lat, color=alt), alpha=.5) +
76  #  geom_point(data = dfb_smo, aes(x = lon, y = lat, color=alt), size=.8, alpha=.3) +
77    scale_colour_gradient(limits=c(4000, 8000), low="red", high="blue") +
78    ggtitle(paste(day2))
79  p1
80  p2
81  ggplotly()
82  ```
```



45:1 | Chunk 2 | R Markdown

Console ~/Documents/XLAX/rtlogs.git/

```
> ?ggplot
>
```

Environment | History | Git

Import Dataset | Grid

Global Environment

Name	Type	Length	Size	Value
dist_max	numeric	1	48 B	2000
filterBBox	function	1	9.7 KB	function (df, p1,…
filterDist	function	1	26.1 …	function (dfin, p…

Files | Plots | Packages | Help | Viewer

R: Create a new ggplot | Find in Topic

Create a new ggplot

Description

`ggplot()` initializes a ggplot object. It can be used to declare the input data frame for a graphic and to specify the set of plot aesthetics intended to be common throughout all subsequent layers unless specifically overridden.

Usage

```
ggplot(data = NULL, mapping = aes(), ..., environment = p
```

Arguments

`data`	Default dataset to use for plot. If not already a data.frame, will be converted to one by `fortify`. If not specified, must be suppled in each layer added to the plot.
`mapping`	Default list of aesthetic mappings to use for plot. If not specified, must be suppled in each layer added to the plot.
`...`	Other arguments passed on to methods. Not currently used.
`environment`	If an variable defined in the aesthetic mapping is not found in the data, ggplot will look for it in this environment. It defaults to using the environment in which `ggplot()` is called.

Dataframe

Similar to a list of vectors ordered inhomogeneous with same number of rows

Variables, Fields or Dimensions

Categories or (Factor) Levels

Observations, Measurements or Cases

	mpg	cyl	disp	hp	drat	wt	qse
Mazda RX4	21.0	6	160.0	110	3.90	2.620	1
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	1
Datsun 710	22.8	4	108.0	93	3.85	2.320	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	1
Valiant	18.1	6	225.0	105	2.76	3.460	2
Duster 360	14.3	8	360.0	245	3.21	3.570	1
Merc 240D	24.4	4	146.7	62	3.69	3.190	2
Merc 230	22.8	4	140.8	95	3.92	3.150	2
Merc 280	19.2	6	167.6	123	3.92	3.440	1
Merc 280C	17.8	6	167.6	123	3.92	3.440	1
Merc 450SE	16.4	8	275.8	180	3.07	4.070	1
Merc 450SL	17.3	8	275.8	180	3.07	3.730	1
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	1
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	1
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	1
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	1
Fiat 128	32.4	4	78.7	66	4.08	2.200	1
Honda Civic	30.4	4	75.7	52	4.93	1.615	1

X

Getting your dataframe right

How many variables? What are they?

	Granite	Limestone	Sandstone
Trad	36	0	52
Sport	76	8	41
Bouldering	102	0	13

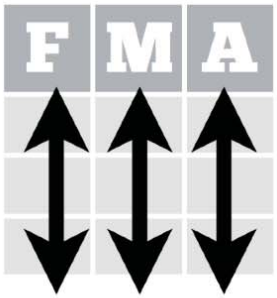
Long format

rock	type	count
Granite	Trad	36
Granite	Sport	76
Granite	Bouldering	102
Limestone	Trad	0
Limestone	Sport	8
Limestone	Bouldering	0
Sandstone	Trad	52
Sandstone	Sport	41
Sandstone	Bouldering	13

X

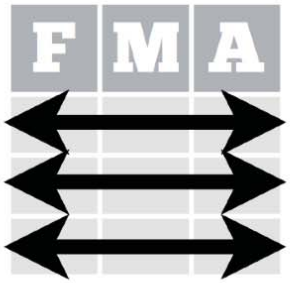
Data Wrangling

In a tidy data set:



Each **variable** is saved in its own **column**

&



Each **observation** is saved in its own **row**

Use R tools (e.g., subset, plyr, dplyr)

- Reshaping the data
- Creating/deleting variables
- Subsetting
- Summarizing
- Grouping
- Combining data sets

See: https://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf

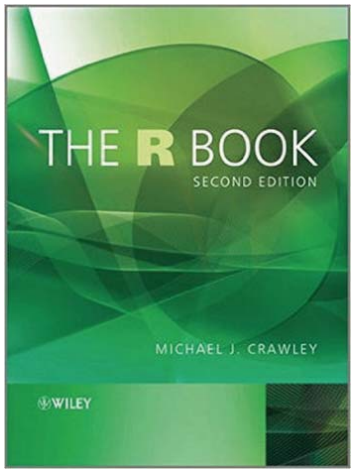
Learn more

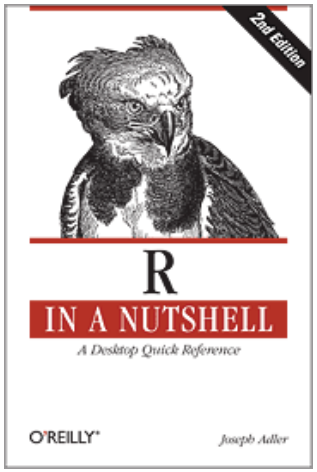
(very) short introduction to R - CRAN
https://cran.r-project.org/doc/contrib/Torfs+Brauer-Short-R-Intro.pdf
An Introduction to R (also in EPub format)
https://cran.r-project.org/manuals.html

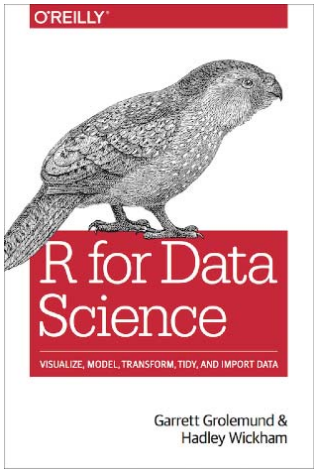
R Studio website resources: https://www.rstudio.com/online-learning/

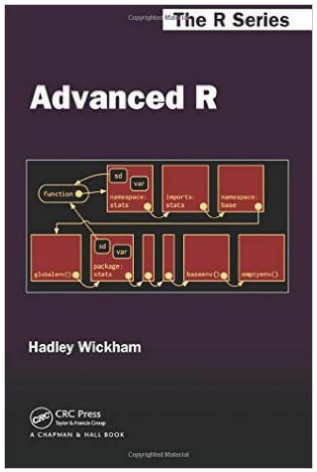
Cheatsheets and reference cards:
https://www.rstudio.com/wp-content/uploads/2016/09/r-cheat-sheet-1.pdf
https://cran.r-project.org/doc/contrib/Short-refcard.pdf
https://cran.r-project.org/doc/contrib/Baggott-refcard-v2.pdf

Spark R documentation: https://spark.apache.org/docs/latest/sparkr.html









ggplot2 [Wickham 2009]

Declarative programming (you say what not how)
Minimum plot: data, aesthetic mapping and geom

ggplot2 grammar components

Defaults
- **Data**
- **Mapping**

Layer
- Data
- **Mapping**
- **Geom**
- Stat
- Position

Scale

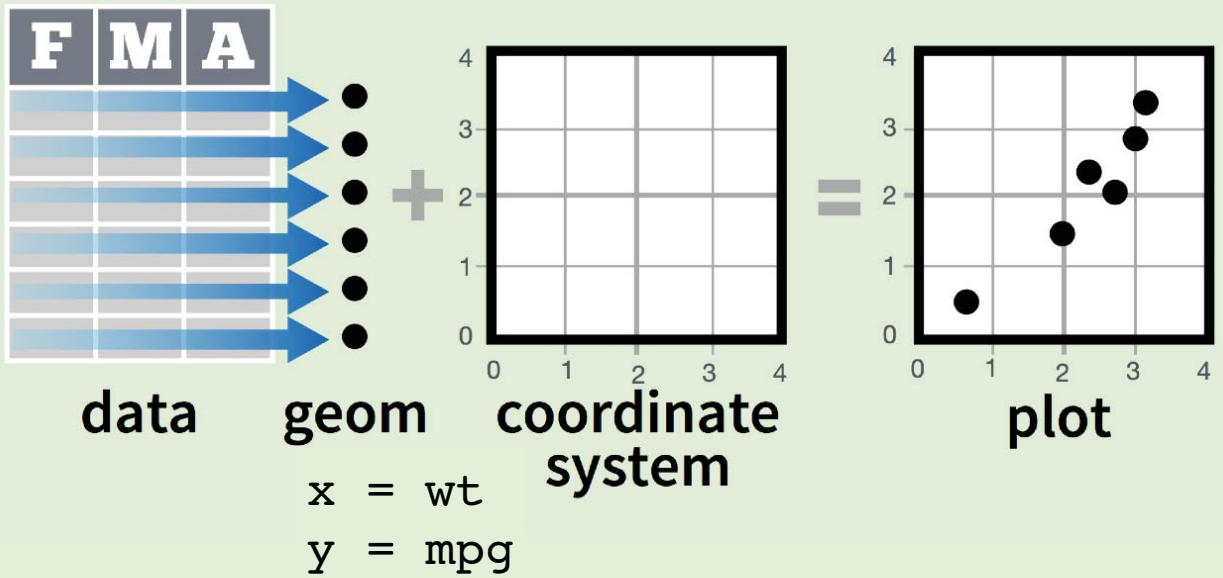
Coord
Facet

```
ggplot(mtcars, aes(x = wt, y = mpg)) +
  geom_point()
```

data

mapping

geom



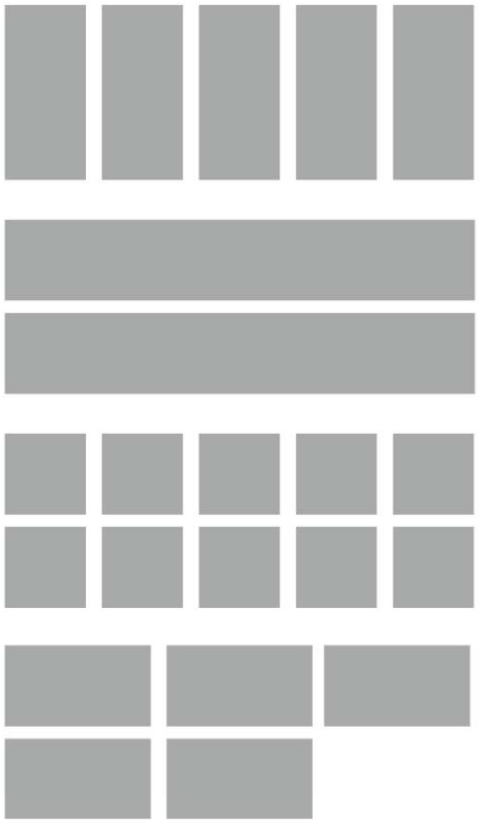
ggplot2 basics geoms (named plots)

Plot	Geom
Area plot	`geom_area()`
Bar chart	`geom_bar(stat = "identity")`
Line plot	geom_line()
Scatterplot	geom_point()
Polygons	geom_polygons()
Rectangles	geom_rect(), geom_tile(), geom_raster()
Text	geom_text()

Faceting

- AKA small multiples, trellis, lattice, grid, panel chart
- Automatic layout of multiple plots on a page
- Alternative to using aesthetics (color, shape, size)

```
t <- ggplot(mpg, aes(cty, hwy)) + geom_point()
```

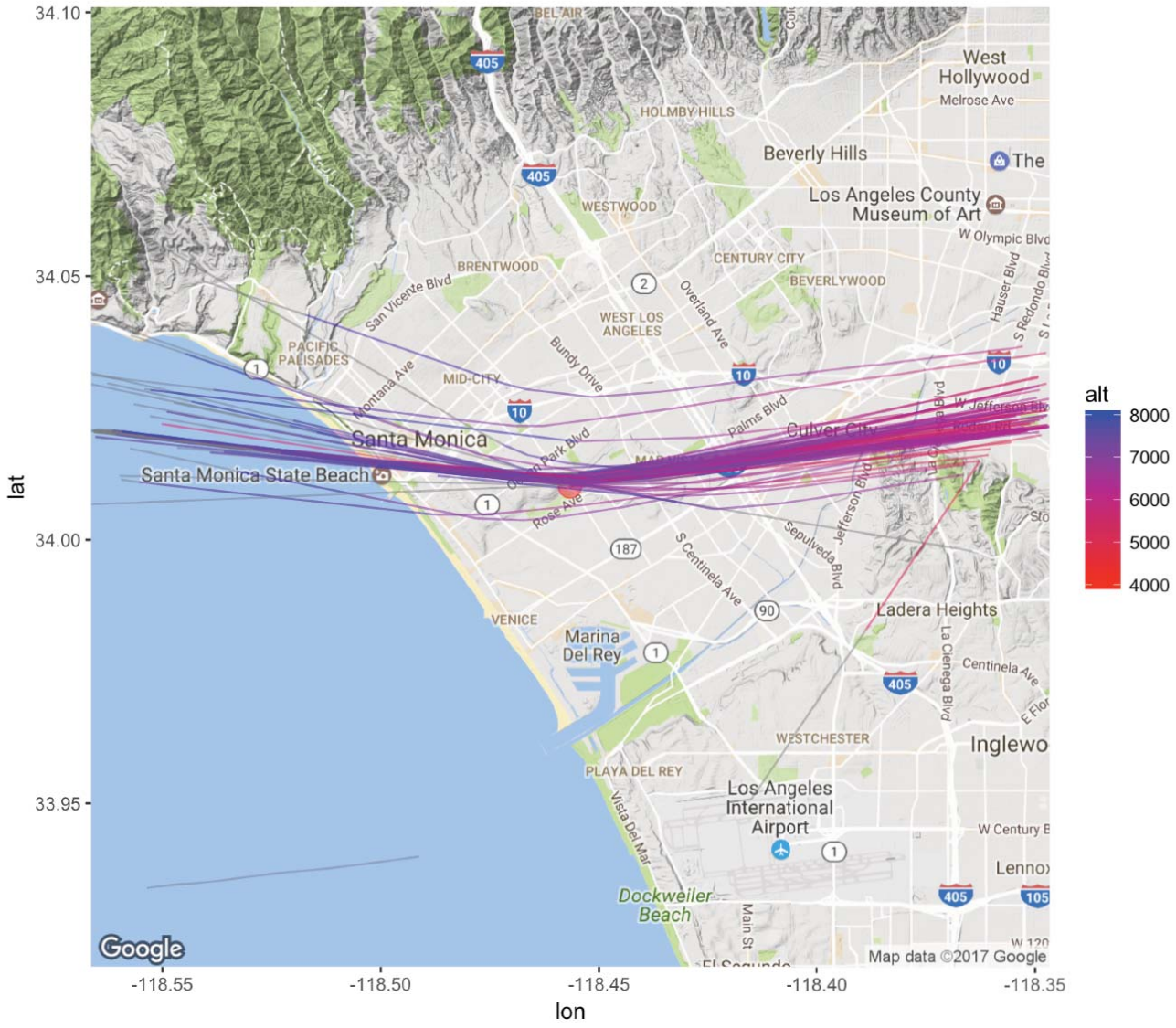


t + facet_grid(. ~ fl)
facet into columns based on fl

t + facet_grid(year ~ .)
facet into rows based on year

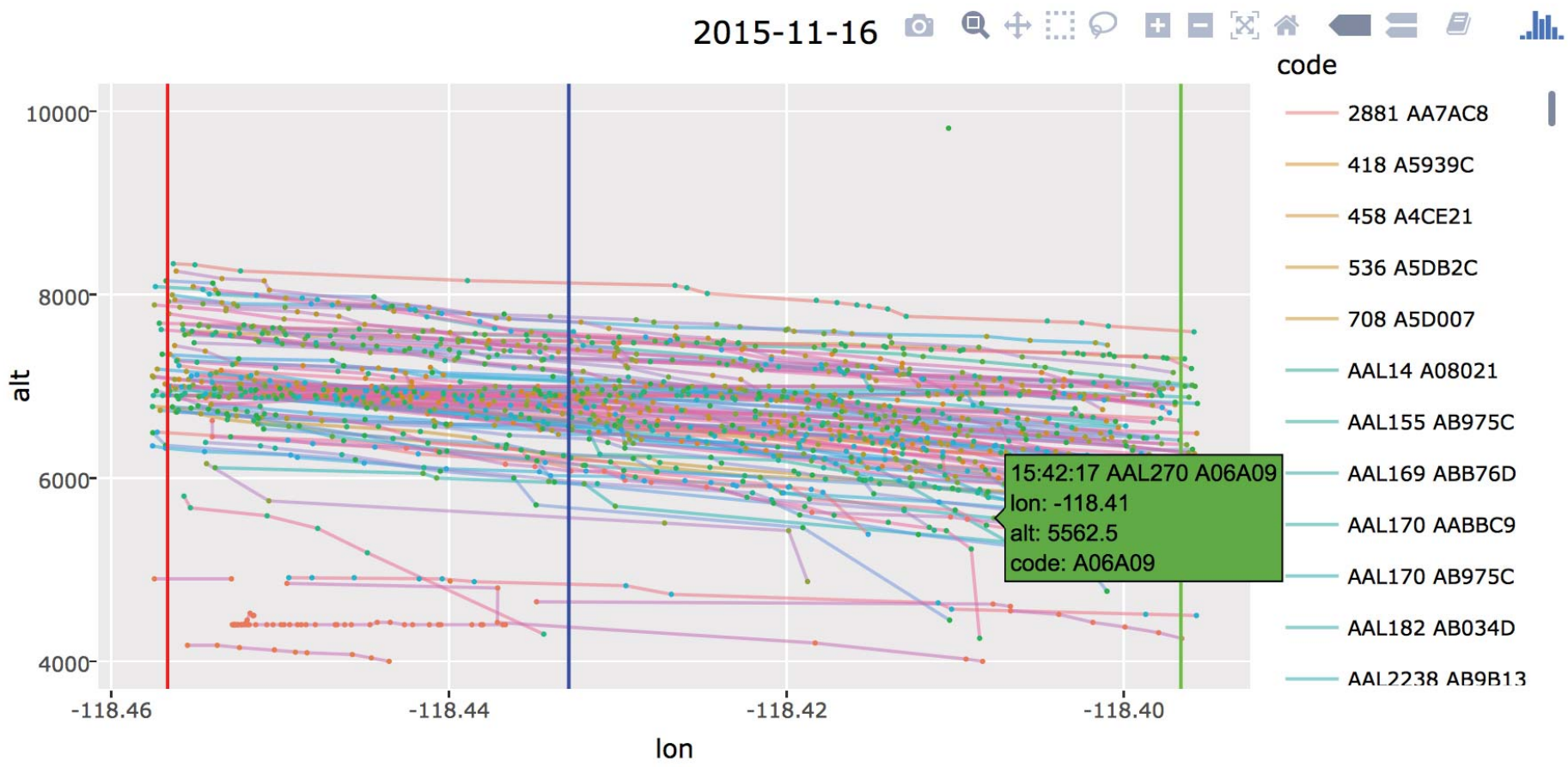
t + facet_grid(year ~ fl)
facet into both rows and columns

t + facet_wrap(~ fl)
wrap facets into a rectangular layout




```
map.google <- get_map(smo, zoom = 12)
p <- ggmap(map.google)
p + geom_point(SMO, aes(x=lon, y=lat), color="red", size=5, alpha=.5) +
  geom_path(df1_smo, aes(x = lon, y = lat, color=alt), alpha=.5) +
  scale_colour_gradient(limits=c(4000, 8000), low="red", high="blue")
```

https://plot.ly



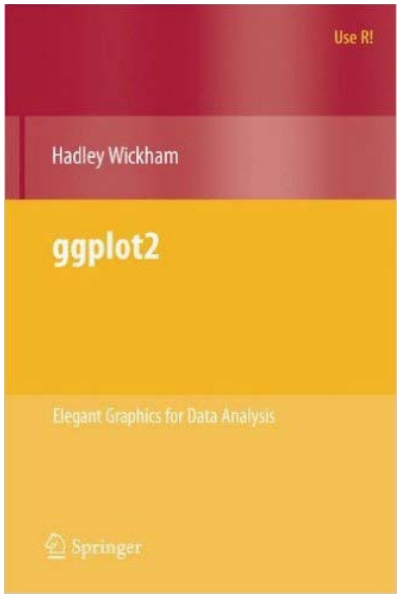

```
p <- ggplot(df2_smo, aes(x = lon, y = alt, color = code)) +
    geom_path(data = df2_smo, aes(x = lon, y = alt, color = name), alpha=.5) +
    geom_point(alpha = 1, size=.3, aes(text = paste(format(datetime,'%H:%M:%S'), name))) +
    geom_vline(xintercept = SMO$lon, color ="red") + annotate("text", x = -118.85, y = 0, label = SMO$label, color ="red") +
    geom_vline(xintercept = MCW$lon, color ="blue") + annotate("text", x = -118.85, y = 1000, label = MCW$label, color ="blue") +
    geom_vline(xintercept = CCD$lon, color ="green") + annotate("text", x = -118.85, y = 2000, label = CCD$label, color ="green") +
    xlim(SMO$lon - 0.001, CCDowntown$lon + 0.001) +
    ylim(4000, 10000) +
    ggtitle(paste(day2))
ggplotly()
```

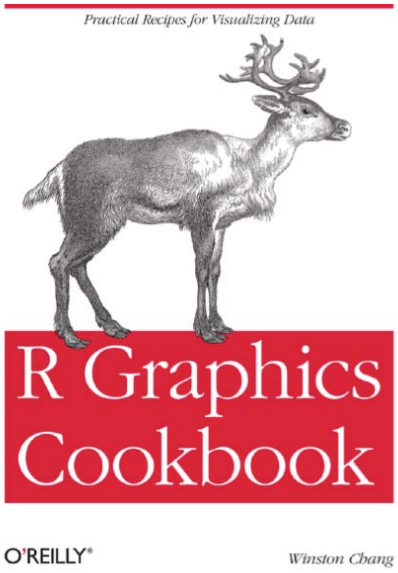
X

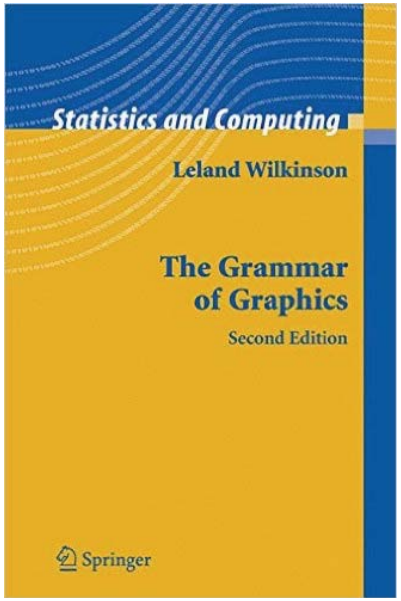
References on ggplot2

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2009.

http://docs.ggplot2.org/current/





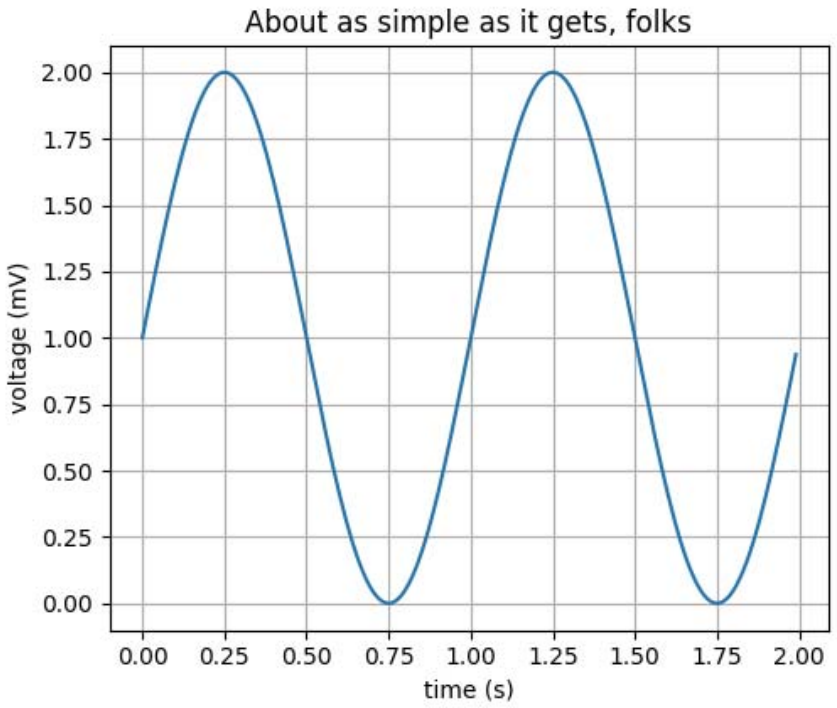



Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions
X

Matplotlib

http://matplotlib.org
Imperative (functional) programming
Emulating the MATLAB® [1] graphics commands
D3 web export with https://plot.ly/matplotlib/
http://matplotlib.org/faq/installing_faq.html#os-x-notes

```
import matplotlib.pyplot as plt
import numpy as np

T = np.arange(0.0, 2.0, 0.01)
S = 1 + np.sin(2*np.pi*t)
plt.plot(T, S)

plt.xlabel('time (s)')
plt.ylabel('voltage (mV)')
plt.title('About as simple as it gets, folks')
plt.grid(True)
plt.savefig("test.png")
plt.show()
```



Visual Studio Code

X

Seaborn Matplotlib Toolkit

http://seaborn.pydata.org

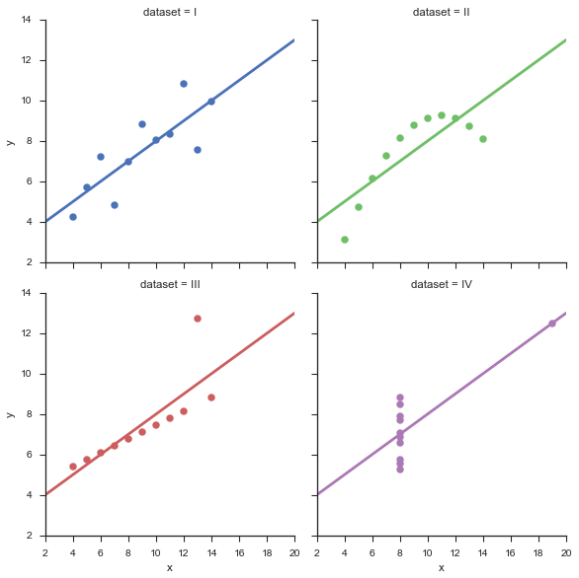
Imperative (functional) programming

Visualization library based on matplotlib

High-level interface for drawing attractive statistical graphics

Support for pandas

```
import seaborn as sns
sns.set(style="ticks")

# Load the example dataset for Anscombe's quartet
df = sns.load_dataset("anscombe")

# Show the results of a linear regression within each dataset
sns.lmplot(x="x", y="y", col="dataset", hue="dataset", data=df,
           col_wrap=2, ci=None, palette="muted", size=4,
           scatter_kws={"s": 50, "alpha": 1})

sns.plt.show()
```



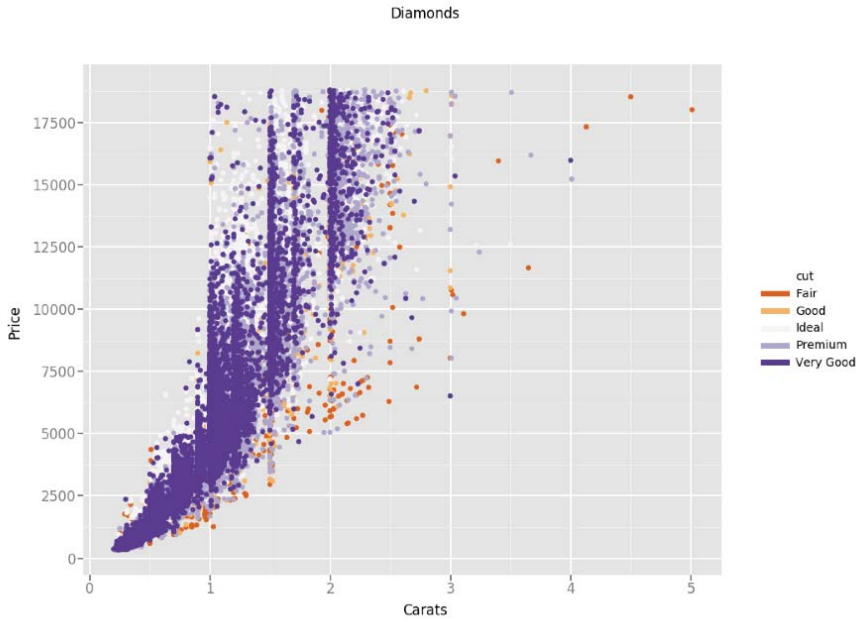
Yhat ggplot Matplotlib Toolkit

http://ggplot.yhathq.com

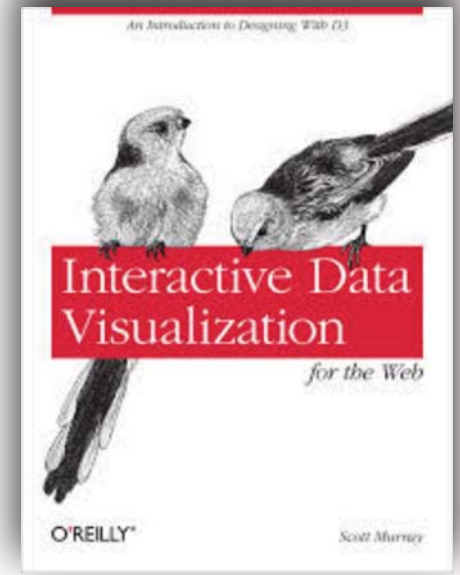
Declarative programming

Visualization library based on ggplot2

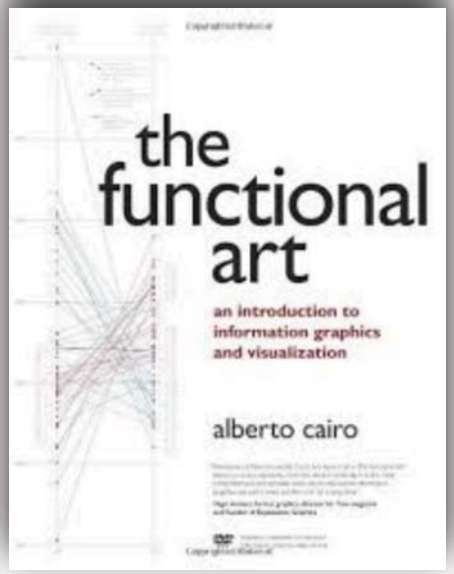
```
from ggplot import *

p = ggplot(diamonds, aes(x='carat', y='price', color='cut')) +\
    geom_point() +\
    scale_color_brewer(type='diverging', palette=4) +\
    xlab("Carats") + ylab("Price") + ggtitle("Diamonds")

p.save('ggplot-plot.png')
```



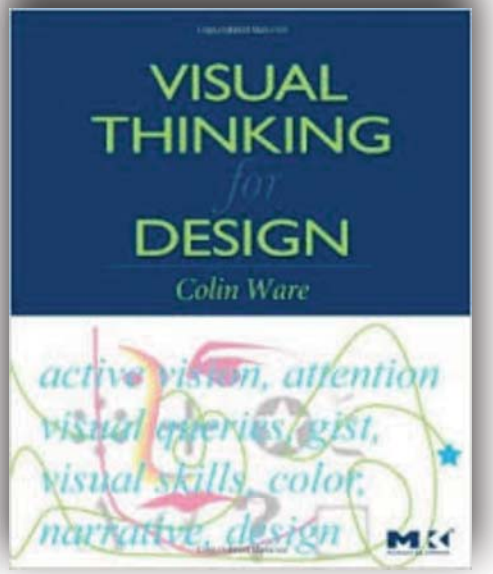
INF 554 textbooks



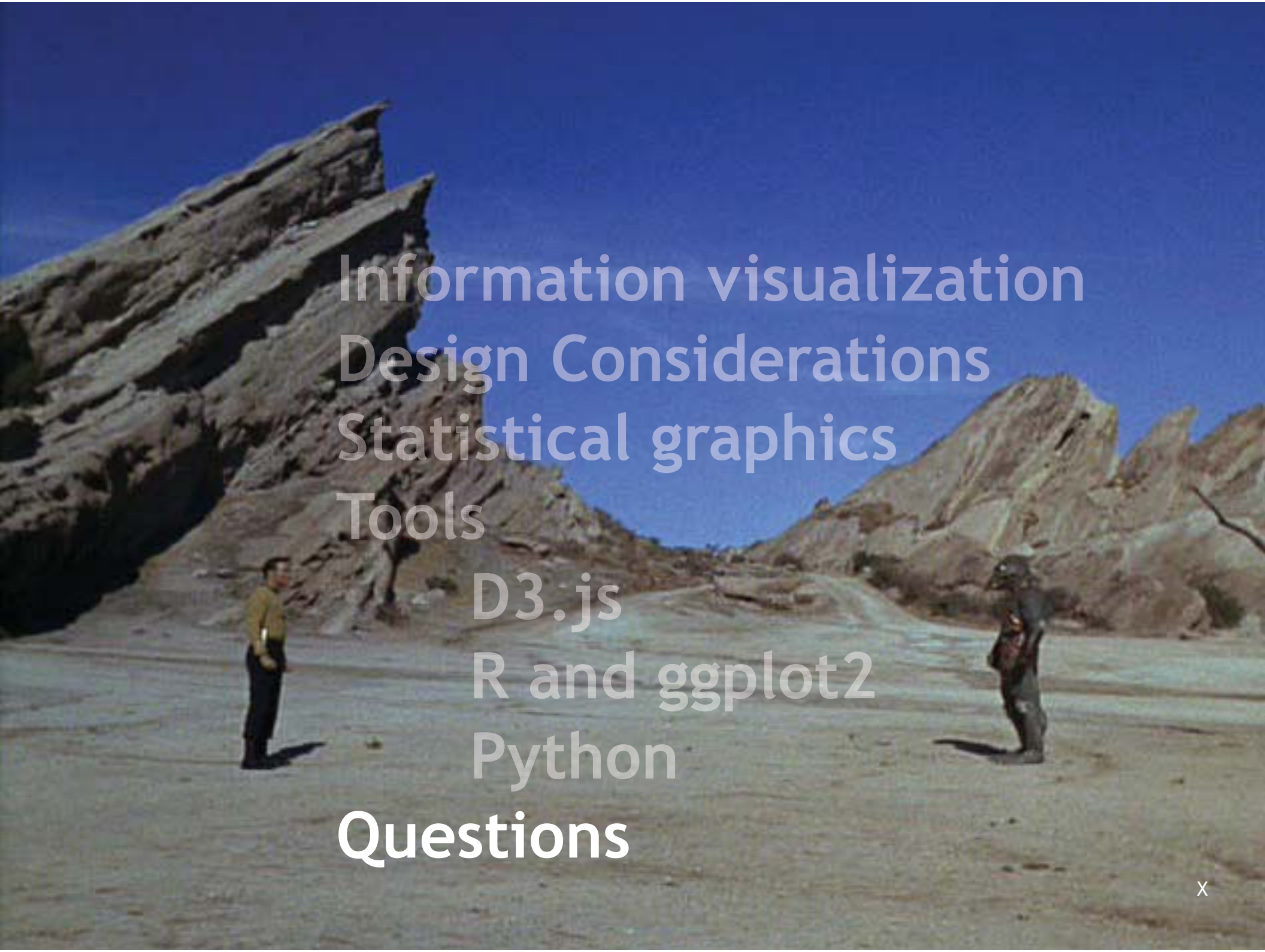
Interactive Data Visualization for the Web, by Scott Murray.
ISBN: 978-1449339739.
Online version available



The Functional Art: An Introduction to Information Graphics and Visualization, by Alberto Cairo.
ISBN: 978-0321834737.



Visual Thinking for Design, by Colin Ware.
ISBN: 978-0123750303.


Information visualization
Design Considerations
Statistical graphics
Tools
D3.js
R and ggplot2
Python
Questions