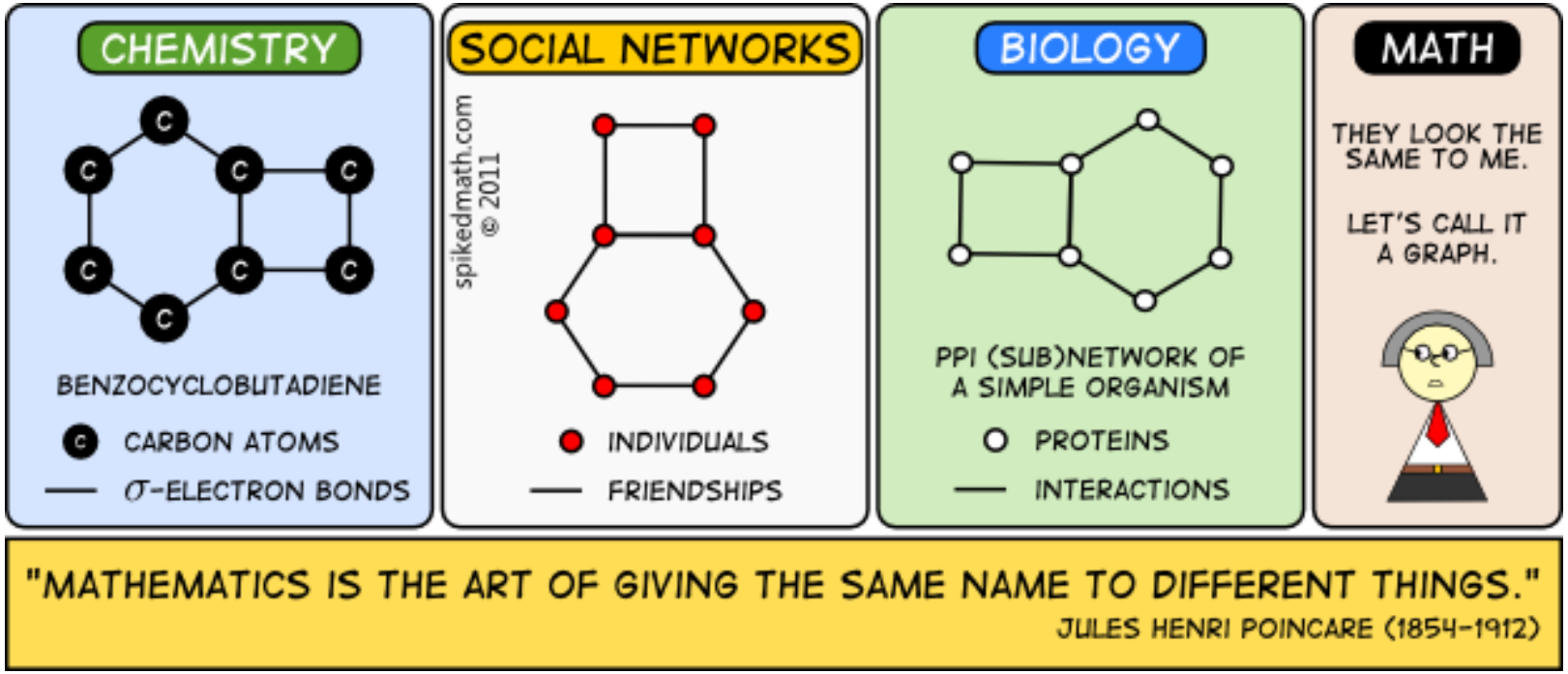
UNIVERSITY of WASHINGTON

Data Science UW Methods for Data Analysis

More on Regression and Graph Algorithms
Lecture 5
Nick McClure


CHEMISTRY
C
BENZOCYCLOBUTADIENE
CARBON ATOMS
σ-ELECTRON BONDS
SOCIAL NETWORKS
spikedmath.com
© 2011
INDIVIDUALS
FRIENDSHIPS
BIOLOGY
PPI (SUB)NETWORK OF
A SIMPLE ORGANISM
PROTEINS
INTERACTIONS
MATH
THEY LOOK THE
SAME TO ME.
LET'S CALL IT
A GRAPH.
"MATHEMATICS IS THE ART OF GIVING THE SAME NAME TO DIFFERENT THINGS."
JULES HENRI POINCARE (1854-1912)

Topics

- Review
- Linear Regression
- Introduction to Graph Theory
- Common Graph Algorithms
- Genetic Algorithms (time permits)

Review

- Normality Tests
  - KS Test
  - Shapiro-Wilk Test
  - QQ-plot: qqnorm()
- Group Testing
  - ANOVA
  - Multiple Hypotheses Testing:
    - Bonferonni Correction (slightly strict)
- Central Limit Theorem
  - Confidence Intervals
- Linear Regression
  - MSE, R^2, least squares fitting

Linear Regression Assumptions

- Linear relationship between dependent variable and independent variables.
- Measurement error is random.
- Residuals are homoscedastic. I.e, the errors are the same across all groups of independent variables.

Homoscedasticity

- Our assumed model:

$$y_i = mx_i + b + \epsilon_i$$

$$\epsilon_i \sim N(0, \sigma)$$

  - Know that m,b, σ are all fixed parameters in that model.

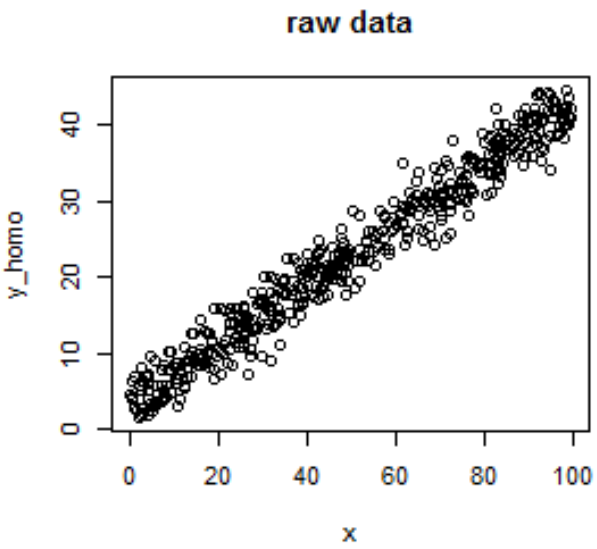
- Compare with:

$$y_i = mx_i + b + \epsilon_i$$

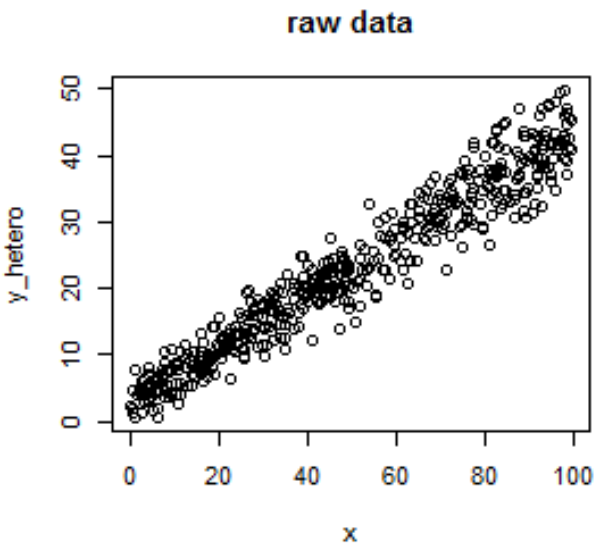
$$\epsilon_i \sim N(0, e^{x_i})$$

Homoscedasticity

homoscedastic



heteroscedastic



Interpreting R's Output

```
x = 1:20
y = x + rnorm(20)
best_fit = lm(y~x)
summary(best_fit)
```

```
Call: lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.04153 -0.43442 -0.01455 0.73806 1.93583

Coefficients:
            Estimate      Std.    Error t value Pr(>|t|)
(Intercept) -0.23738   0.53628   -0.443   0.663
x            1.02521   0.04477   22.901   9.19e-15 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 1.154 on 18 degrees of freedom
Multiple R-squared: 0.9668,      Adjusted R-squared: 0.965
F-statistic: 524.4 on 1 and 18 DF,   p-value: 9.186e-15
```

Distribution of residuals:
Residuals = (actual – pred)

Interpreting R's Output

```
x = 1:20
y = x + rnorm(20)
best_fit = lm(y~x)
summary(best_fit)
```

```
Call: lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.04153 -0.43442 -0.01455 0.73806 1.93583

Coefficients:
            Estimate   Std. Error  t value   Pr(>|t|)
(Intercept) -0.23738      0.53628   -0.443      0.663
x            1.02521      0.04477   22.901   9.19e-15 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 1.154 on 18 degrees of freedom
Multiple R-squared: 0.9668,     Adjusted R-squared: 0.965
F-statistic: 524.4 on 1 and 18 DF,   p-value: 9.186e-15
```

Least Squares estimates of coefficients

Standard Error of coefficients

Hypothesis statistic for test that coefficient is NOT equal to zero. (Two tailed). The Null is equal to zero.

P-value for coefficient hypothesis test.

Interpreting R's Output

```
x = 1:20
y = x + rnorm(20)
best_fit = lm(y~x)
summary(best_fit)
```

```
Call: lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.04153 -0.43442 -0.01455 0.73806 1.93583

Coefficients:
            Estimate      Std.    Error t value Pr(>|t|)
(Intercept) -0.23738   0.53628   -0.443   0.663
x            1.02521   0.04477   22.901   9.19e-15 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 1.154 on 18 degrees of freedom
Multiple R-squared: 0.9668,      Adjusted R-squared: 0.965
F-statistic: 524.4 on 1 and 18 DF,   p-value: 9.186e-15
```

Least squares estimate of our standard deviation of error (σ):

$$y_i = mx_i + b + N(0, \sigma)$$

Interpreting R's Output

```
x = 1:20
y = x + rnorm(20)
best_fit = lm(y~x)
summary(best_fit)
```

```
Call: lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.04153 -0.43442 -0.01455 0.73806 1.93583

Coefficients:
            Estimate      Std.    Error t value Pr(>|t|)
(Intercept) -0.23738   0.53628   -0.443   0.663
x            1.02521   0.04477   22.901   9.19e-15 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 1.154 on 18 degrees of freedom
Multiple R-squared: 0.9668,      Adjusted R-squared: 0.965
F-statistic: 524.4 on 1 and 18 DF,   p-value: 9.186e-15
```

R-squared of the model.

Adjusted R-squared, accounts for complexity of formula:

$$R^2 - adj = 1 - \frac{(1-R^2)(n-1)}{(n-1)-p} \text{ for p<n-1}$$

Interpreting R's Output

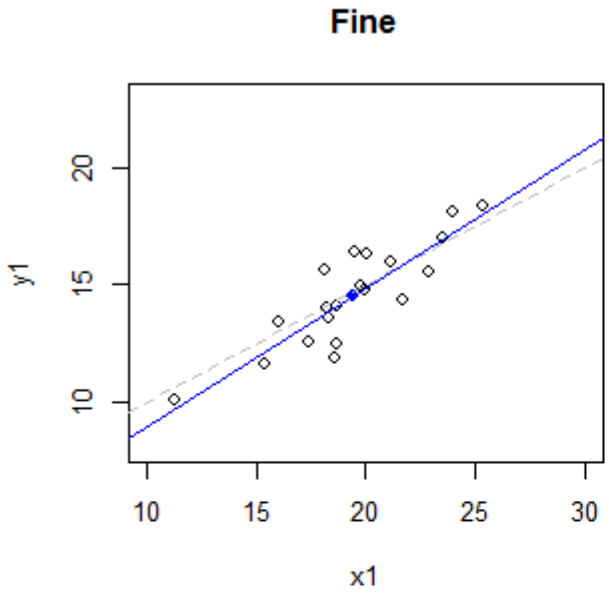
```
x = 1:20
y = x + rnorm(20)
best_fit = lm(y~x)
summary(best_fit)
```

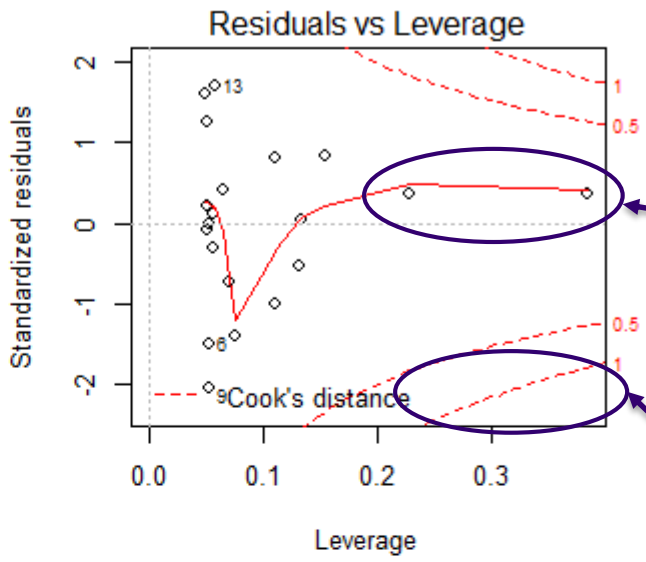
```
Call: lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max
-2.04153 -0.43442 -0.01455 0.73806 1.93583

Coefficients:
            Estimate      Std.    Error t value Pr(>|t|)
(Intercept) -0.23738   0.53628   -0.443   0.663
x            1.02521   0.04477   22.901   9.19e-15 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 1.154 on 18 degrees of freedom
Multiple R-squared: 0.9668,      Adjusted R-squared: 0.965
F-statistic: 524.4 on 1 and 18 DF,   p-value: 9.186e-15
```

- The F-statistic is a statistic for the hypothesis test that the linear model is a better fit than the mean of y. ($H_0: R^2 = 0$)

Leverage and Cook's Distance

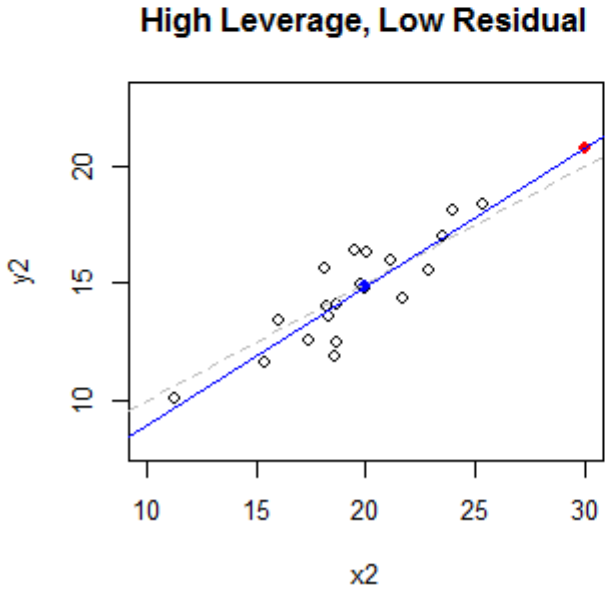
- Linear regression fits a line based on the means of the y and x values. It fits a line that goes through the means of both values.
- The line pivots around that point relative to the pull of each point. Points that are further away from the mean pull harder on the slope.
- Another way to quantify the 'pull' of each point, is to fit the line to the data without each point and see how the parameters move. This is called Cook's Distance.

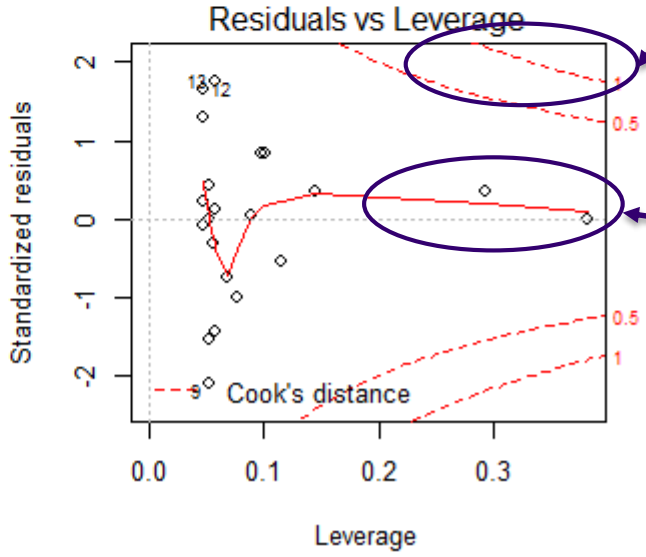




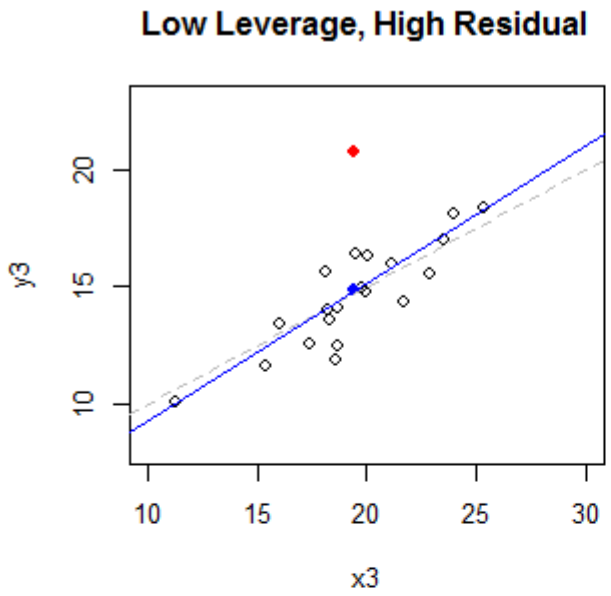
Be aware: High Leverage, Low Residual

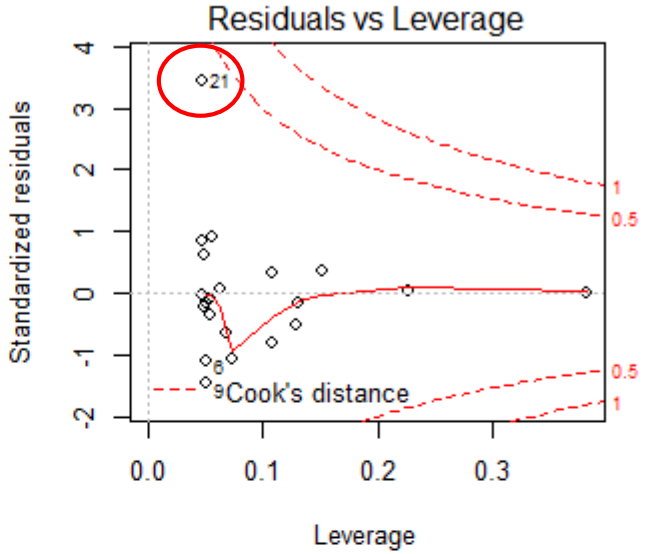
Problem areas: High Leverage, High Residual

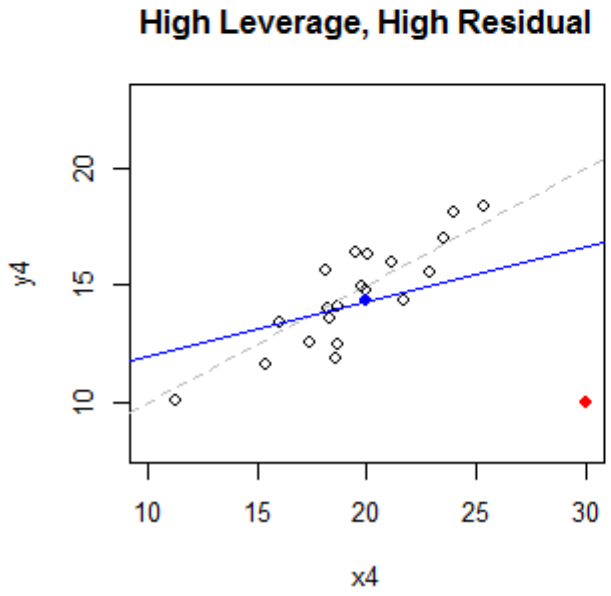


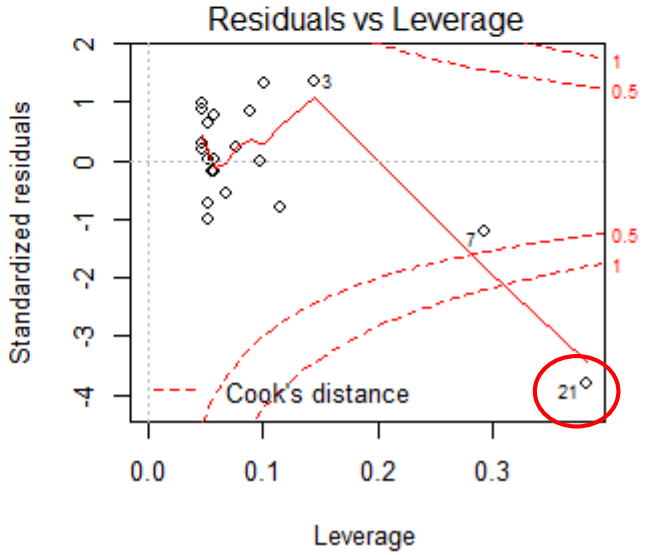


Be aware: High Leverage, Low Residual


Low Leverage, High Residual
y3
x3
10
15
20
25
30

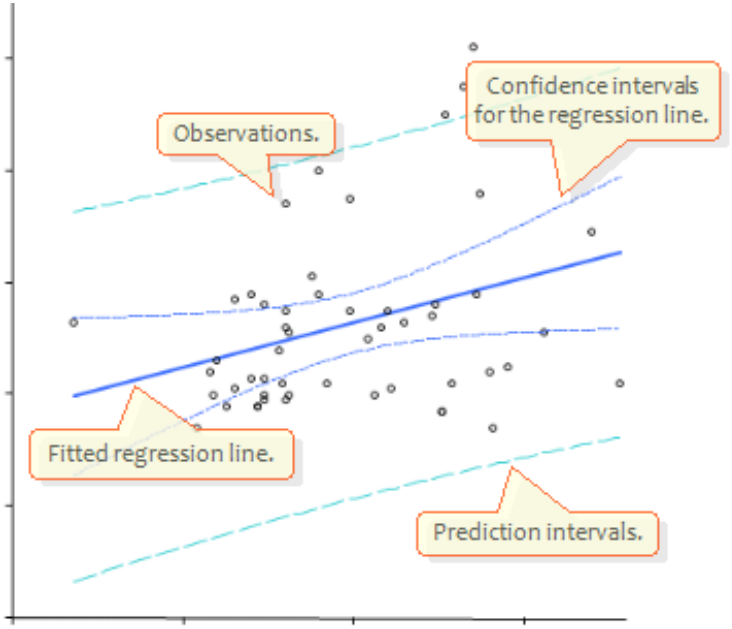

Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
21
6
9
1
0.5
0.0
0.1
0.2
0.3


High Leverage, High Residual
y4
x4
10
15
20
25
30


Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
3
7
21
1
0.5
0.0
0.1
0.2
0.3

Prediction Vs. Confidence in Linear Regression

- Confidence error is the error we assign to the parameters. (m,b,…)
- Prediction error is the error in observing another point.



- R demo

Linear Models

- Models

  $y = ax + bz + c + \epsilon$

  $y = ax + bz + cx^2 + dxz + \epsilon$

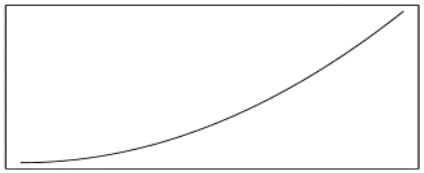
  $y = a\ln(x) + b\sin(z^2) + c + \epsilon$

  …

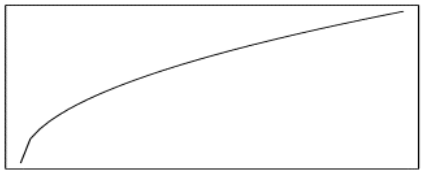
- All of these are linear *in the coefficients*
- Think of these as transformations on the independent variables.
- Methods and interpretation are largely the same as in the simple case

Non-Normal Data

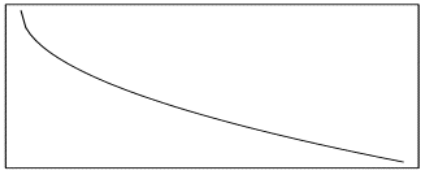
- Transform data to normality:
  - Examples:



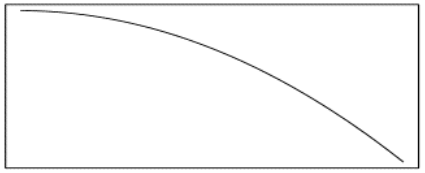
Exp(), power>1



Power>1



Log(), power<1



Power<1, sqrt()

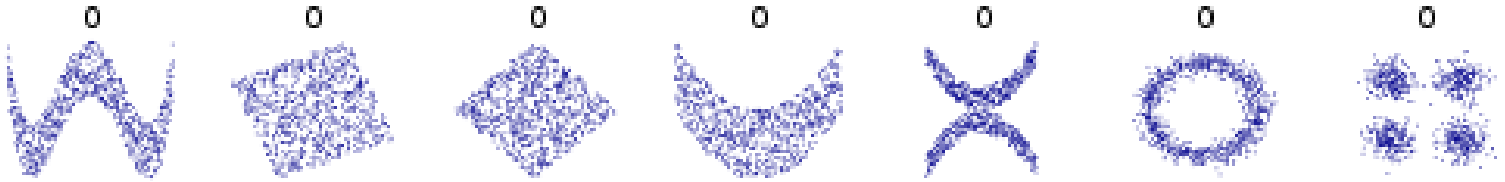
- R-Demo
  - data(attenu) = peak acceleration data from the 1979 CA earthquake measured at 182 different sites.
    - Event: event #
    - Mag: magnitude
    - Station: Station factor
    - Dist: Distance from earthquake origin
    - Accel: Max peak acceleration

Non-Normal Data

- Transform data to normality:
  - Note you can only use **monotonic** functions for transformations for interpretability.
  - These are functions that preserve:
    - If a < b then f(a) < f(b).
  - Always either non-increasing or non-decreasing.

Non-Linear Relationships & Measurement Error

- Other than transformations, we can't do much here.
- Solutions:
  - Fit a non-linear regression.
  - Use a non-regression machine learning model. (Touch on these at the end of the semester, and it is a large subject of the third class)



- Cannot change measurement error, but you can control for it.

Example: Temperature measurements may have more variance at the high and low ends of an instruments limits. We can add a variable that takes into account how far from the limits the measurement is. More to come on this later.

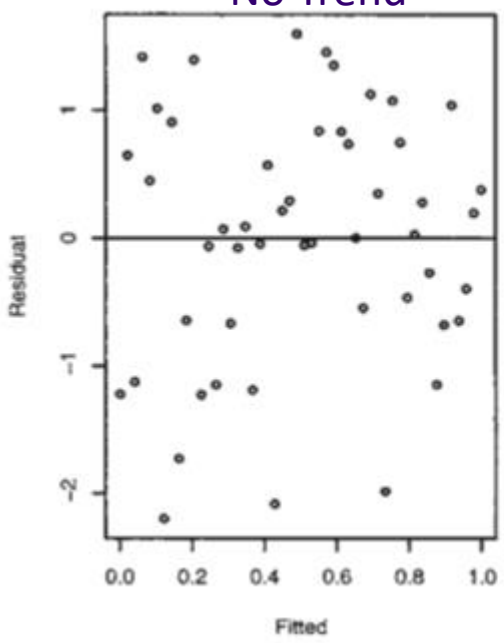
Checking the Residuals

- Always plot the residuals!
- You should always check for trends in the residuals.
- Residuals vs. fitted values:
  - We should never see a trend here. If there is a trend, our linear fitting failed.
- Residuals vs. y:
  - Always positively correlated. The higher the correlation, the worse the fit. This is an effect of the influence of points on the end of the line. A point near the end has much more influence than a point in the middle. Because of this, higher values of y (near the end) tend to have higher residuals (and vice versa for lower values of y).
- Testing the normality of the residuals is also important
  - shapiro.test() in R.

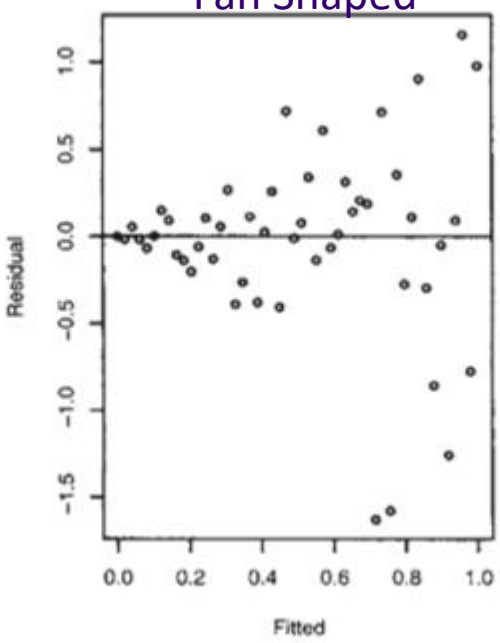
Trends in Residuals vs. Fitted

- Types of outcomes:

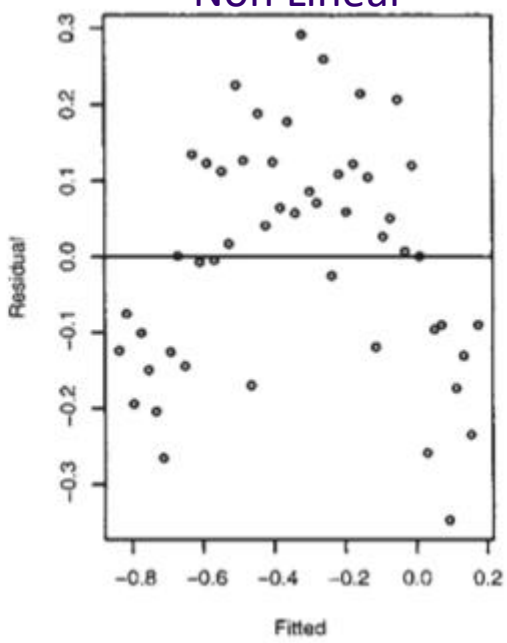
No Trend



Fan Shaped



Non Linear



Trends in Residuals vs. Fitted

- Fan-shaped residuals (Heteroskedastic Residuals):
  - This does not affect our parameter estimates, but only our standard errors.
    - i.e. our population parameter confidence intervals are wider
  - In order to correct for this, we may try transforming our variables
    - Log transforms, sqrt transforms…
- Non-linear residuals
  - Best solution is to use non-linear regression, we will touch on this later.
- R demo
- More in-depth statistics information:
- https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf

Multiple Linear Regression

- Again, by linear, we mean linear in the parameters.

$y = \beta_0 + \beta_1 x_1 + \beta_2 {x_1}^2 + \epsilon$ Univariate Quadratic Model

$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \epsilon$ First Order Model

$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \epsilon$ 2nd-order Interaction Model

$y = \beta_0 + \beta_1 e^{x_1} + \beta_2 {x_2}^{0.5} + \epsilon$ First Order Model (with transformations)

β_0 is still the intercept (what y is equal to when all x's = 0).

$\beta_1, \beta_2, \ldots$ Are known as the partial slopes. Each one still represents the change in y per unit increase in x.

Multiple Linear Regression

- Start with a first order model.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \epsilon$$

- How do we deal with factor/categorical variables?

Gender
F
M
F
F
M
…

Gender
1
0
1
1
0
…

Eye Color
Brown
Brown
Blue
Green
Green
Blue
Brown
…

Brown	Blue
1	0
1	0
0	1
0	0
0	0
0	1
1	0
…	…

“One hot encoding”

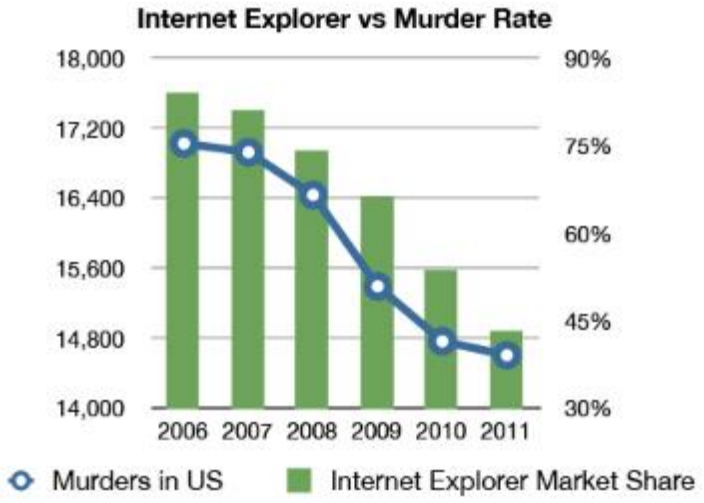
DayOfWeek
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
…

DayOfWeek
1
2
3
4
5
6
7
…

“Factor encoding”

Multiple Linear Regression

- Throwing in all possible variables to help explain our response is sometimes *not* a good thing
  - Variables can be dependent on each other.
  - Variables might not be important to explain the response.
  - Note that the SSE is always larger for reduced models!



How do we choose which combinations of independent variables to use?

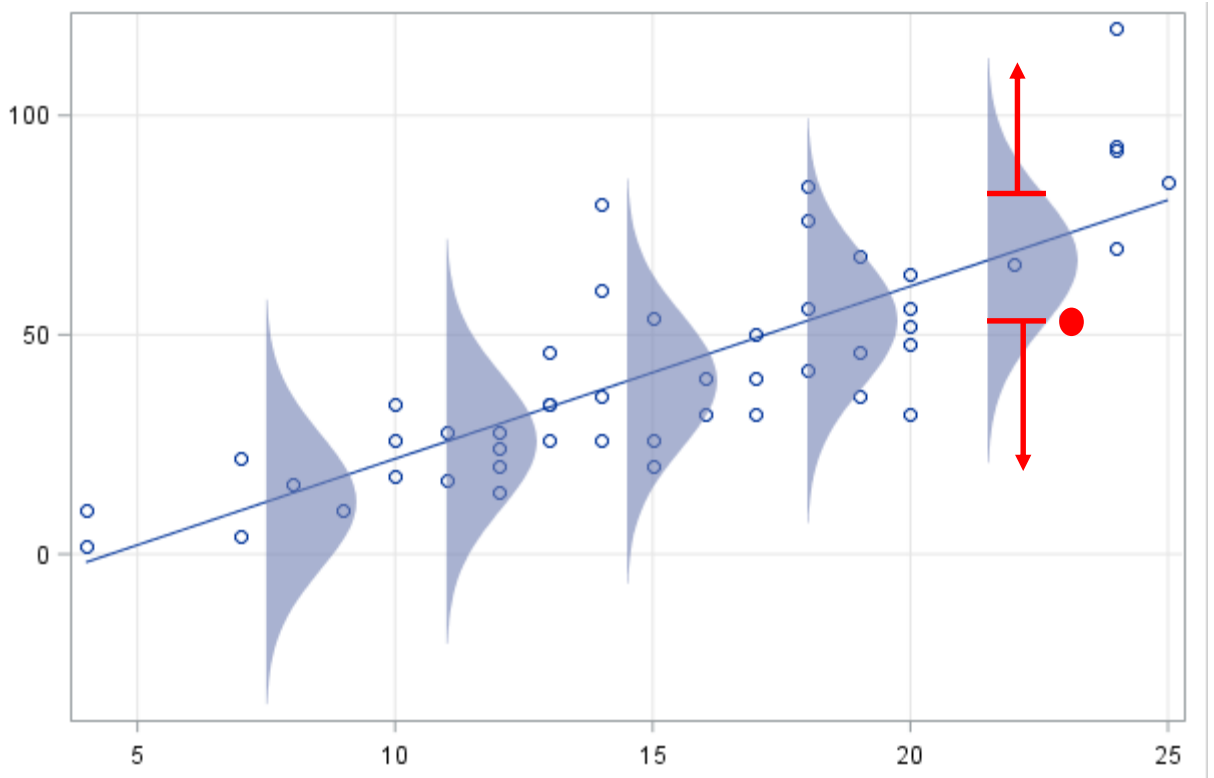
We might consider looking at the difference in SSE between models and the number of explanatory variables.

Multiple Linear Regression

- Given a linear model with known constants:

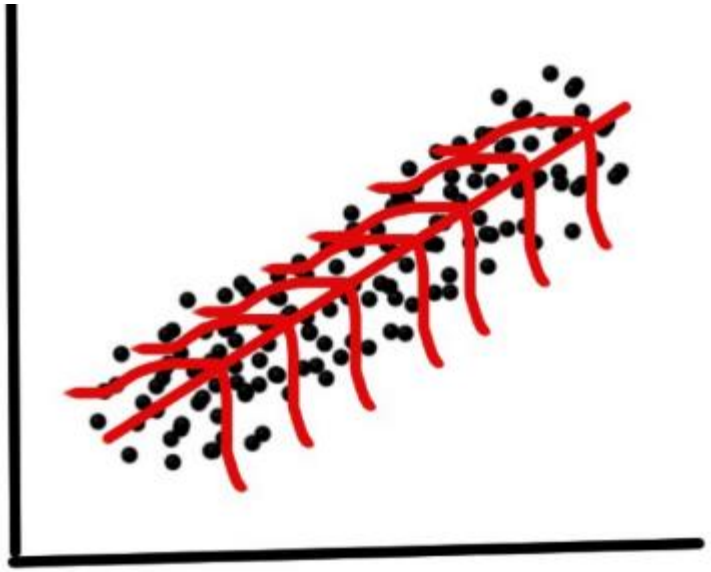
$$y_i = mx_i + b + N(0, \sigma)$$

- Given a point that comes from that line, we can come up with a probability of observing that point.



Multiple Linear Regression

- Linear Model Likelihood
  - We assume errors are normally distributed. From the resulting set of errors, we can come up with a distribution. We then use each residual point and come up with a total error and calculate the probability of that model given our data. This is the likelihood.



To make the calculations easier, we usually take the logarithm of the model (remember we can do this because it is monotonic). This is called the 'log-likelihood'. We will talk more about this when we get to Bayesian Statistics.

Multiple Linear Regression

- Akaike Information Criterion (AIC)
  - Given a model with k-parameters, and a likelihood of L,

$$AIC = 2k - 2\ln(L)$$

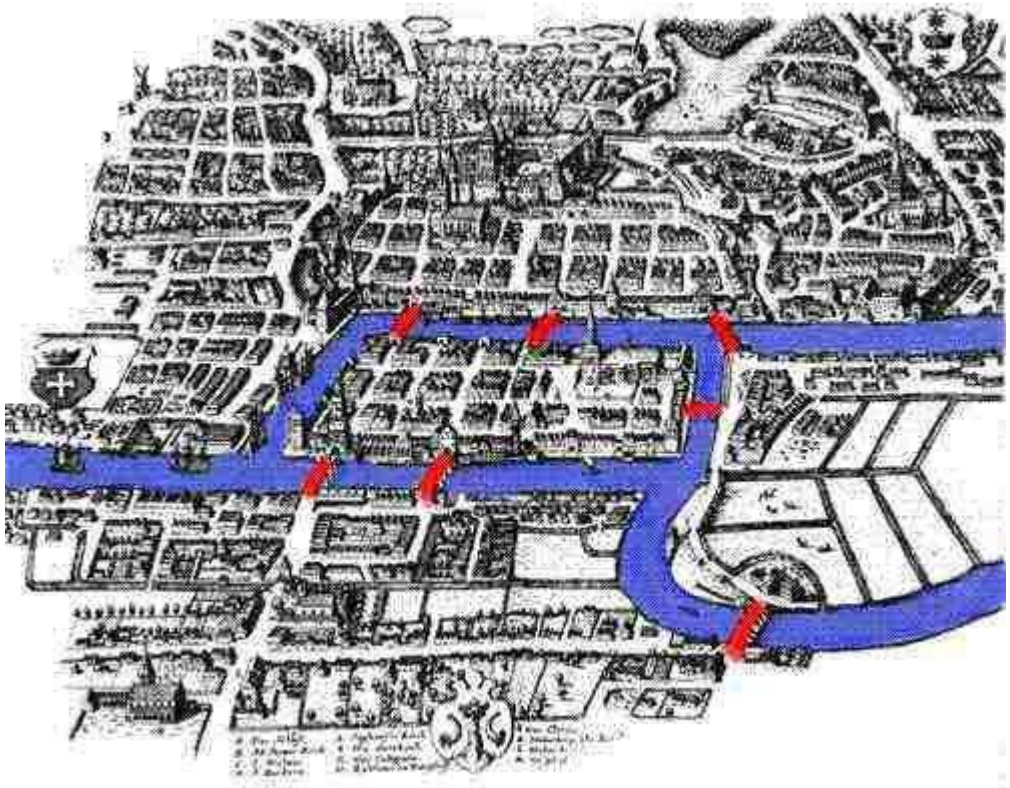
  - Note that the more parameters, the higher the AIC.
  - The higher the likelihood, the lower the AIC.
  - Better models have lower AIC values.

Multiple Linear Regression

- How to select the variables in the model?
- Stepwise regression.
  - Forward Selection:
    - Start with no independent variables and add the variables one by one, selecting the variable that improves your criterion the most.
  - Backward Selection
    - Start with all independent variables and remove on at a time. Remove the one that improves the chosen criterion.
- R-demo

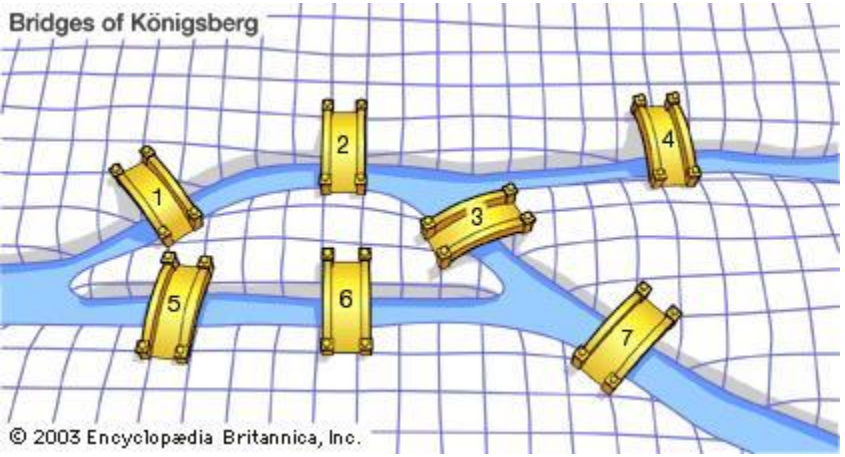
Introduction to Graph Theory

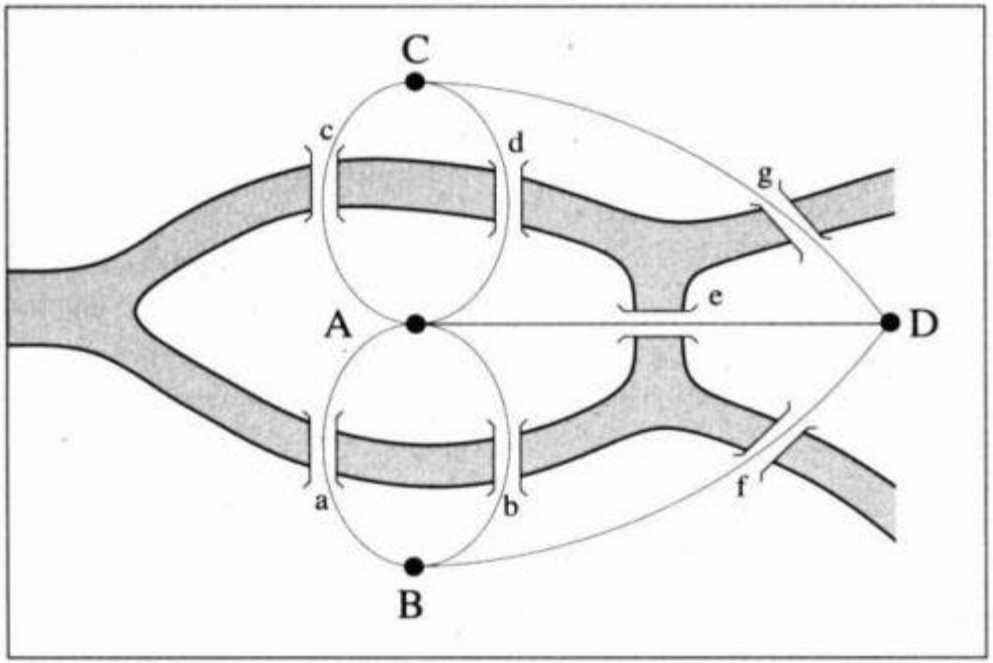
- Leonhard Euler came to Konigsberg, Prussia in 1735 and solved a local problem: Can we go on a walk from home and cross each bridge exactly once and end up at home?



Graph Theory

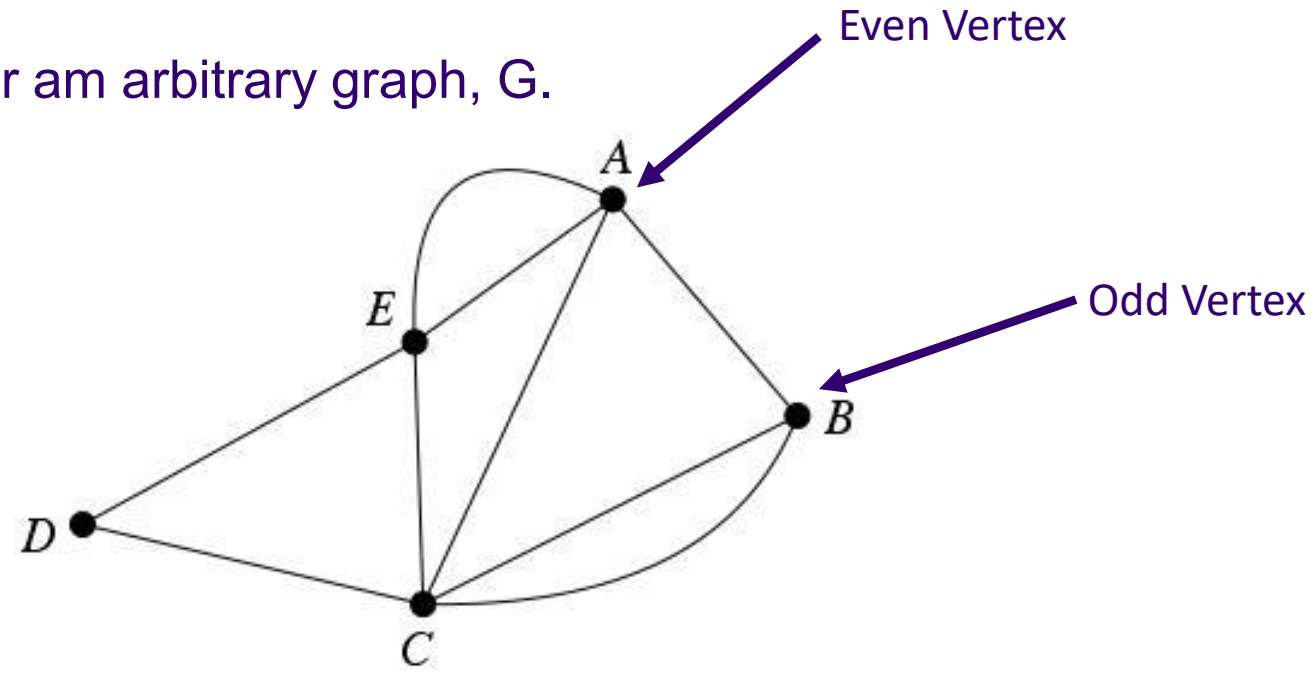
- Euler solved this by creating a graph of the problem.
  - A graph is made up of vertices and edges.
  - All edges have a vertex at both sides.
  - The degree of a vertex is the count of edges that end there.





Graph Theory

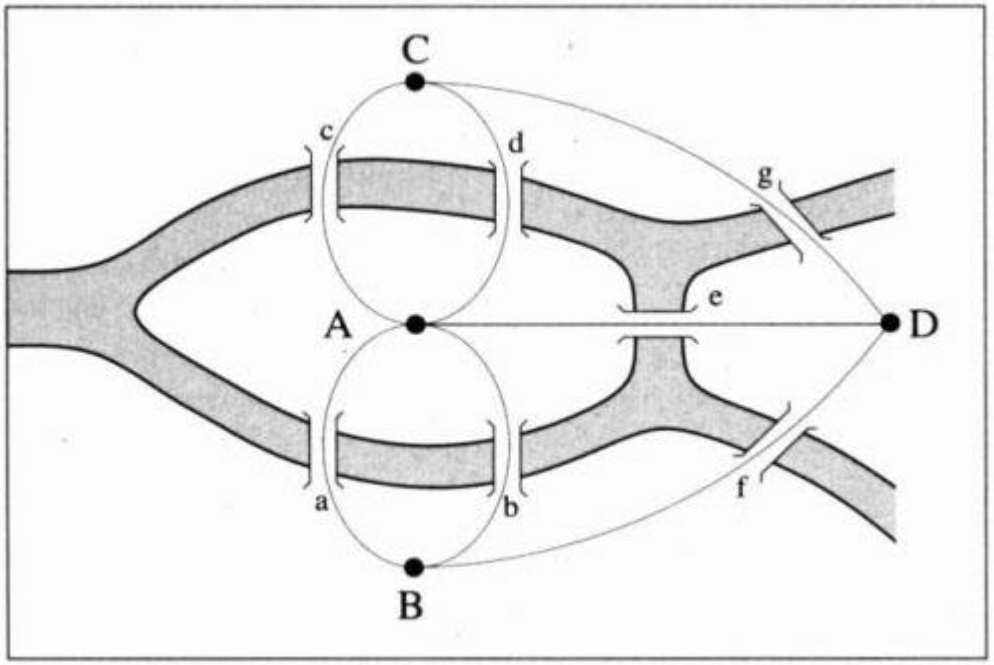
> Let's consider am arbitrary graph, G.



> Properties of any graph:
  - The sum of all the degrees of a graph is even.
  - The sum of all the degrees of a graph is twice the # of edges.
  - There are always an even amount of odd vertices.

Graph Theory

- Euler realized that the degrees in the graph are the most important part of solving this problem.
  - |A| = 5
  - |B| = 3
  - |C| = 3
  - |D| = 3



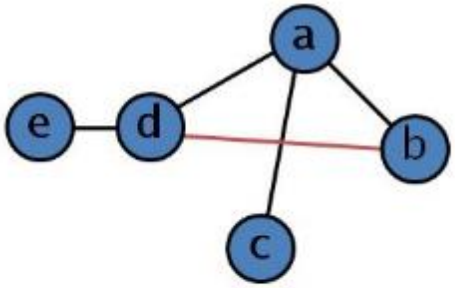
- What's wrong with having an odd vertex?

Why Graphs?

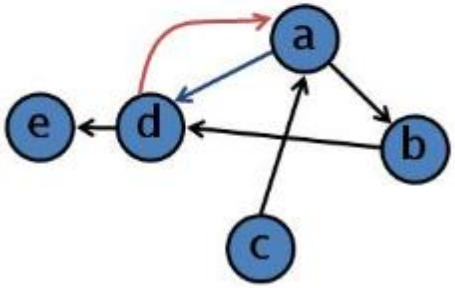
- Graphs can theoretically represent any object.
  - In fact, one of the highest abstractions of mathematics is called category theory, which is represented by graphs.
- For our purposes, that object can be a list, data entries, relationships, joins, etc…
- Databases stored as graphs are a type of NOSQL (not only SQL).
- Why the need? Databases are growing far too big (CAP theorem)
  - It is impossible for computers to provide all of:
    - Consistency (all users have equal access to all data)
    - Availability (every user gets a response)
    - Partition tolerance (the system operates the same under arbitrary partitioning)

Representation of Graphs

- Adjacency matrix



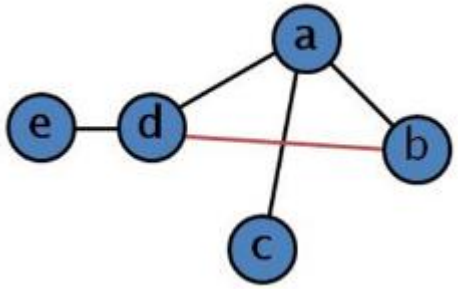
	a	b	c	d	e
a	0	1	1	1	0
b	1	0	0	1	0
c	1	0	0	0	0
d	1	1	0	0	1
e	0	0	0	1	0



	a	b	c	d	e
a	0	1	0	1	0
b	0	0	0	1	0
c	1	0	0	0	0
d	1	0	0	0	1
e	0	0	0	0	0

Representation of Graphs

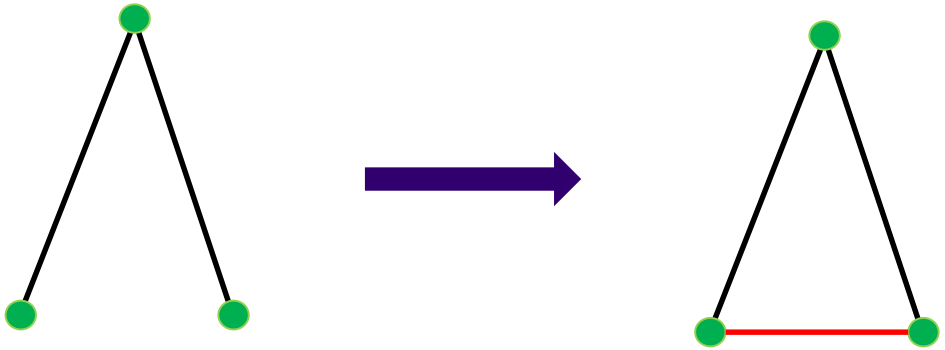
- Vertex-Edge Lists



- Dictionaries (I prefer python for this)
  - E={‘a’:[‘b’,’c’,’d’],’b’:[‘a’],’c’:[‘a’],’d’:[‘a’,’b’,’e’],’e’:[‘d’]}
  - V=E.keys()
- Vectors
  - V=((‘a’,’b’),(‘a’,’c’),(‘a’,’d’),(‘b’,’d’)…)
- R:
  - igraph package
  - Lists:
    - E=list(‘a’=c(‘b’,’c’,’d’),…)
    - V=names(E)
- iPython Demo

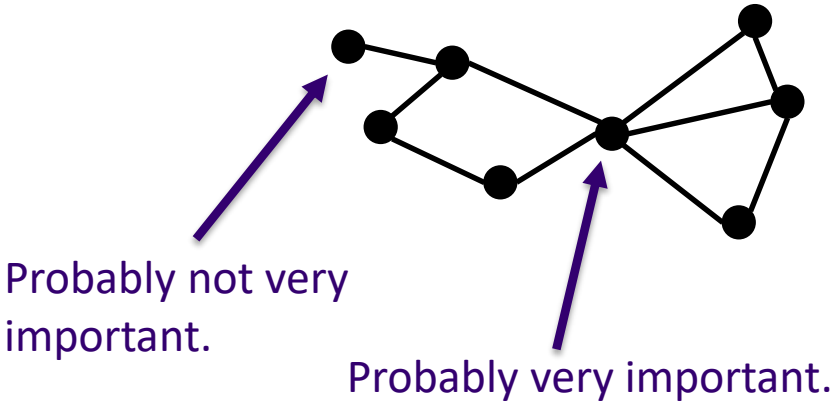
Graph Algorithms

- Recommended Connections:
  - Triangle completions (friends of a friend) (ipython example)



Graph Algorithms

- Measuring Centrality: Finding important nodes/edges.
  - Of all shortest paths, which node/edge are traveled the most?
  - We traverse all combinations of pairs shortest paths and count how many times we crossed a vertex.



Graph Algorithms

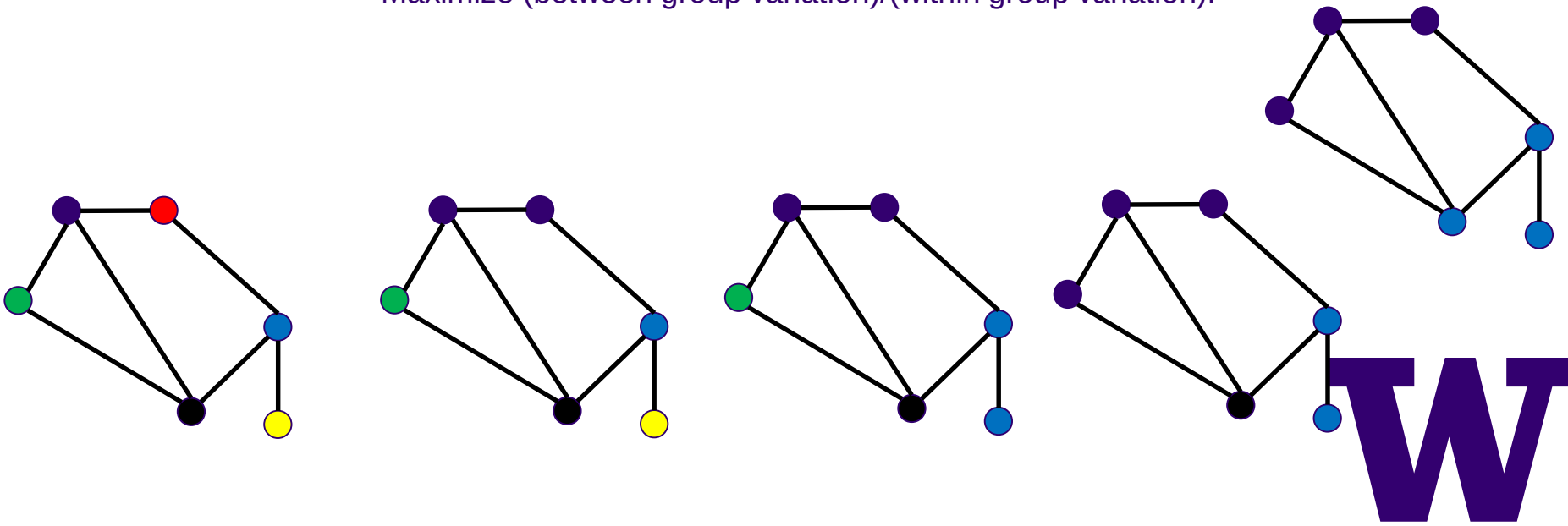
- Graph Labeling/Coloring:
  - Minimum labels/colors needed for each vertex to not be connected to a similar label/color.
  - For a fully connected graph of n-vertices, you need n-colors.
  - For a planar graph (2D, no crossed edges), you need at most 4 colors (4 color theorem).
- Algorithm:
  - Label a starting vertex of your choice with the lowest number (1)
  - For the remaining vertices:
    - Move to a vertex adjacent to your already labeled vertices.
    - Label that vertex the lowest number possible, not including any adjacent labels.
- Graph Coloring is a ‘hard’ problem, i.e., the only way to guarantee the best result is by brute force.

Graph Algorithms

- Use cases:
  - Scheduling: Given a set of jobs that need to be assigned to time slots. Given that some jobs cannot be scheduled together (i.e., the vertices are connected), coloring will provide the minimum time slots required.
  - Meetings for people. Nodes are meetings, edges represent the existence of shared people between meetings. Then a minimum coloring will represent the number of needed time slots for the meetings.

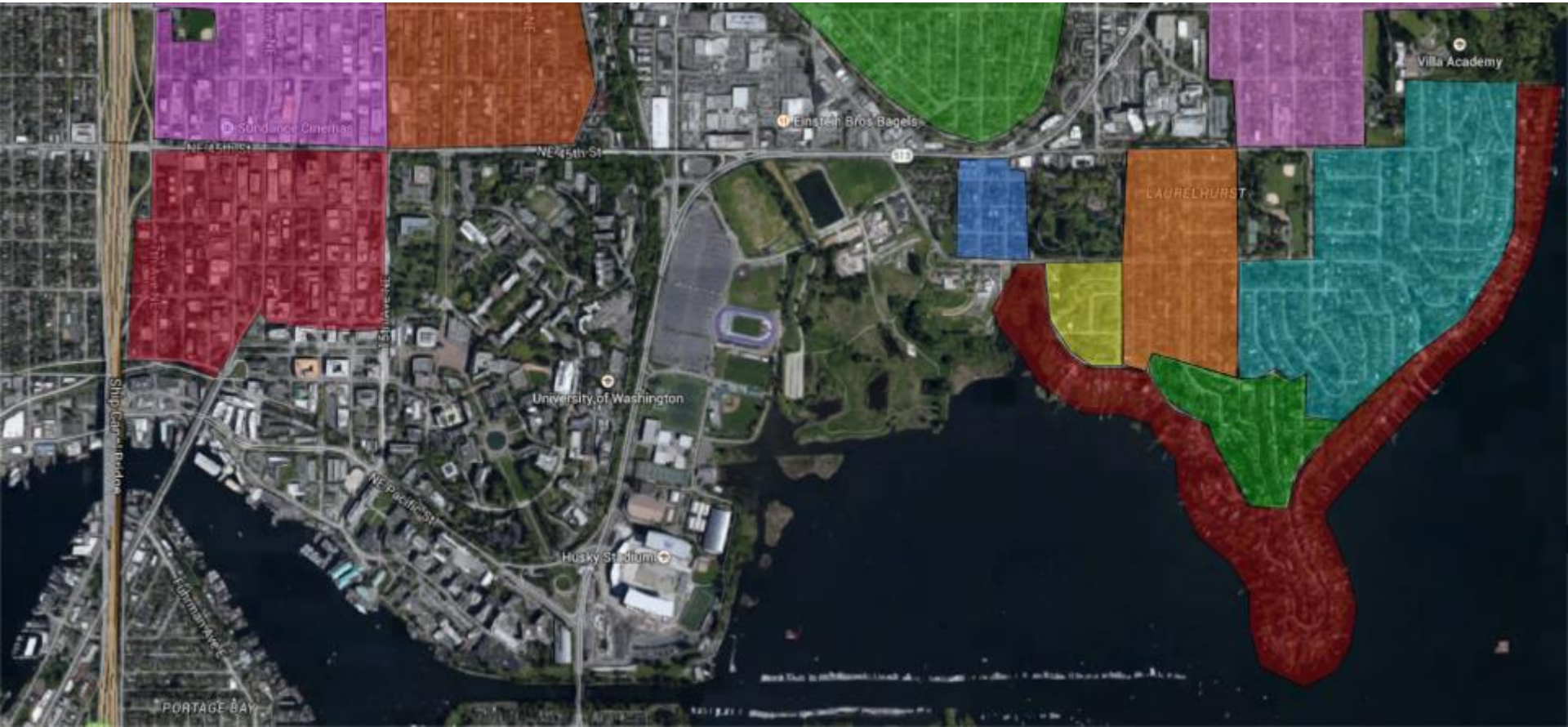
Graph Algorithms

- Clustering Graphs: Finding subgraphs with similar properties
  - Algorithm:
    - Start with every point in it's own group.
    - Find the two closest points, group them together.
    - Repeat this until…
      - Number of specified groups is reached, or
      - No distance is under a specified amount, or
      - Maximize (between group variation)/(within group variation).



Graph Algorithms

- Clustering Graphs: Finding subgraphs with similar properties
  - Zillow uses graph structures to find similar houses for imputation purposes. We can find ‘similar’ neighborhoods using this method.

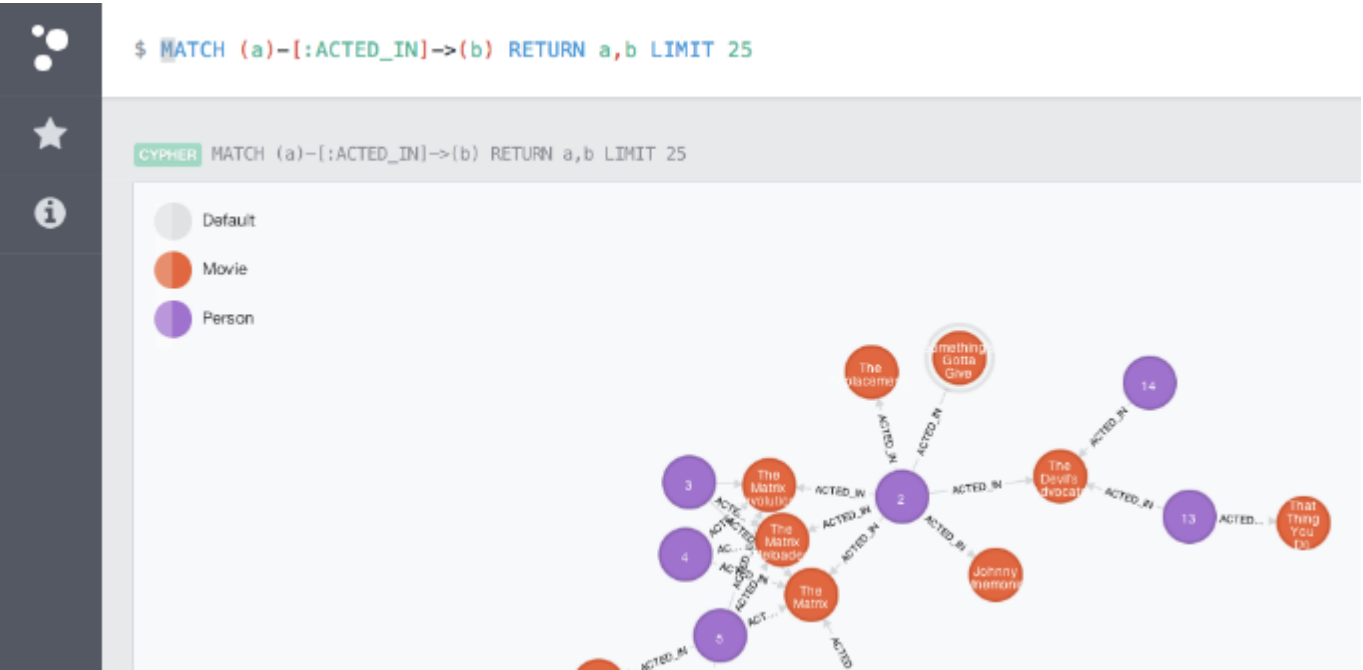


Graph Visualization

- Gephi: Open-source free graph visualization tool.
  - http://Gephi.github.io
  - Still in beta, has a few bugs, but a great tool overall.
- Gephi Example.

A Note on Graph Databases

- Graph databases store data in graph-like structures.
  - Queries are querying graph structures
- Neo4j: Leading developer of Graph Databases.
  - Free online tutorial.
  - http://neo4j.com/graphacademy/online-training/
  - Who uses neo4j? www.stackshare.io



A Note on Graph Databases

- More information available:
  - http://infolab.stanford.edu/~ullman/focs/ch09.pdf

Assignment

- Complete Homework 5:
  - Load Facebook edges file.
    - Write R code that computes the mean degree and shows a plot of the histogram of degrees. Verify this with Gephi.
    - Perform a K-S test for the following: (Use our code from last class!)
      - Test if the distribution of degrees is Poisson. (reuse K-S code)
      - Test if the distribution of degrees is a Power Law. (igraph)
  - You should submit:
    - A script level R-script.
    - A text document summarizing your findings.
    - Read Statistical Thinking for Programmers pages 93-97.