Unsupervised Learning and Clustering

Recap: Supervised Learning

- Supervised learning (think regressors and classifiers)
    - trains a learner to predict a dependent variable, given independent variables
- The training set contains the results for the dependent variable
- There is a definitive “answer” for training data

Unsupervised Learning

Learning method that focuses on **latent variables** (variables not observed but may be inferred)

Examples of latent variables

- Genre groups in movies
- Social/demographic groups within a population
- Particular users or equipment that drive trends

Use of “unlabeled” data

One common case of **unlabeled data**: data with missing values.

A column could have a lot of missing values that need to be approximated.

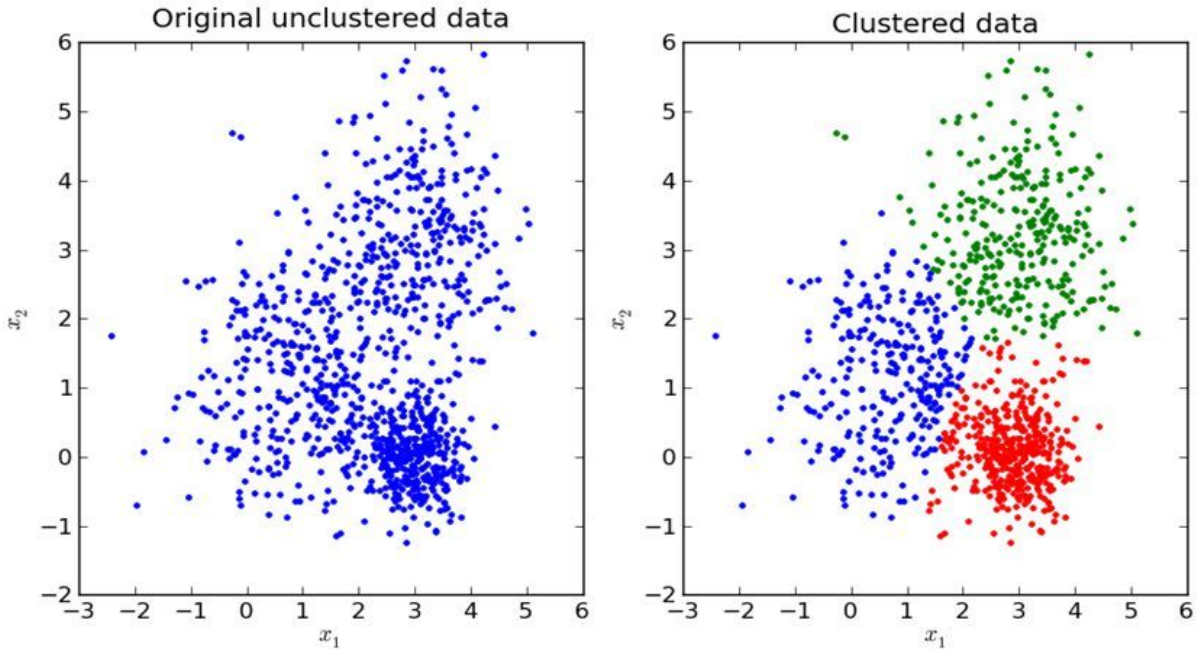
Knowing which “group” the missing points are in can help estimate the true values (imputation from lecture 2).

Cluster Analysis

Clusters are latent variables (can’t find them in data!)

Understanding cluster behavior in data can:

- Yield better understanding of underlying trends in data
- Yield useful parameters for predictive analysis
- Challenge the boundaries of predefined classes in variables



http://images.slideplayer.biz.tr/7/1917098/slides/slide_2.jpg

Recommendation Systems

Recommendations are the heart of many businesses

- Content Providers like Google, Youtube, Spotify...
- Friend Finders in Facebook, Linkedin
- Targeted Advertisements
- Netflix Challenge: 1 million dollar prize for a 10% increase in accuracy!

Rec. Technique 1: Collaborative Filtering

Collaborative filtering uses other data points

- Example: Using other users' ratings to suggest content
- Advantages:
  - If cluster behavior is clear, can yield good insights
- Disadvantages:
  - Computationally expensive
  - Can lead to dominance of certain groups in predictions

Rec. Technique 2: Content Filtering

Content filtering uses similar content

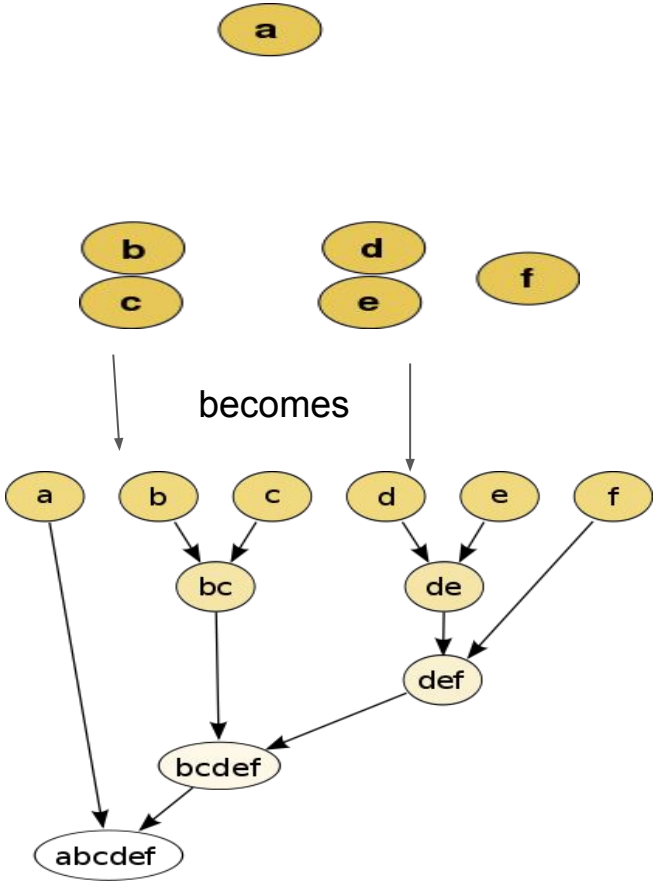
- Example: Using other movies watched by user to recommend unwatched movie
- Advantages:
  - Recommendations made by learner are intuitive
  - Scalable
- Disadvantages:
  - Limited in Scope and applicability

Popular Clustering Algorithms

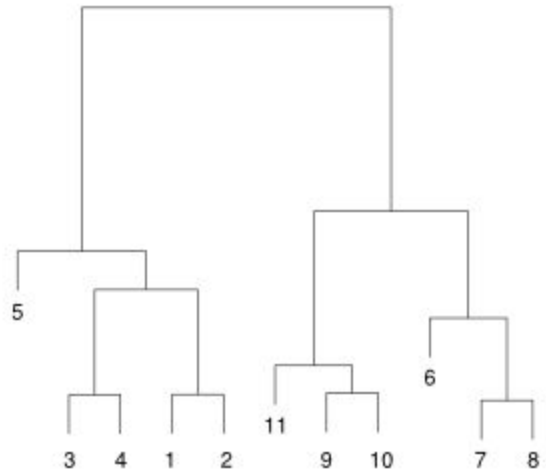
- Hierarchical Clustering
- K-means Clustering
- Gaussian Mixture Model

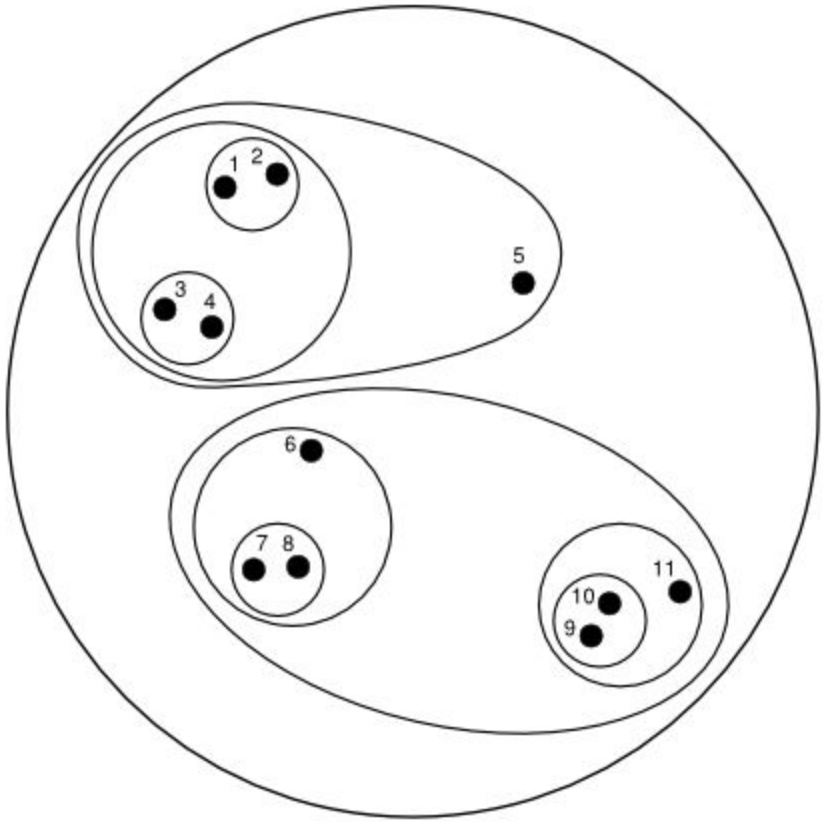
Hierarchical Clustering

- Creates a hierarchy of clusters (clusters of clusters)
- Group each object by distance
- Group until one mega-cluster



https://en.wikipedia.org/wiki/Hierarchical_clustering#/media/File:Hierarchical_clustering_simple_diagram.svg



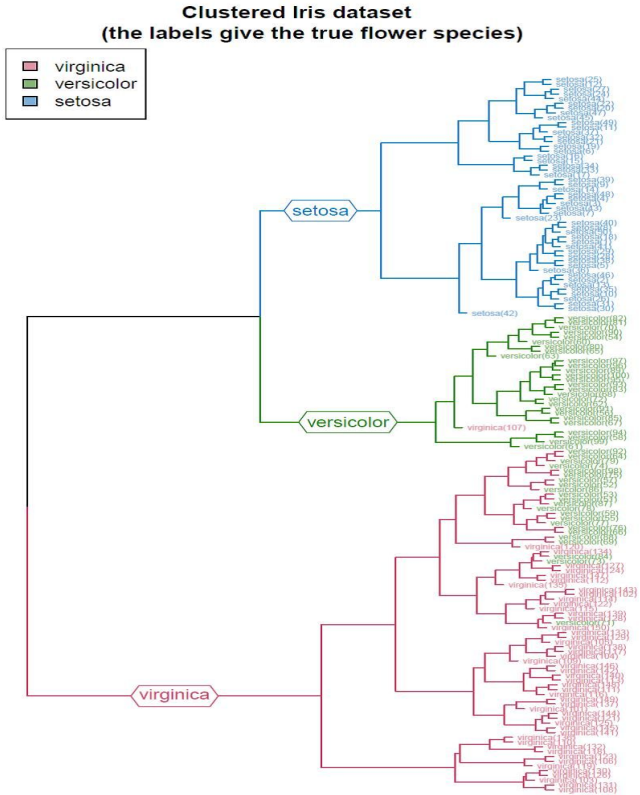


http://cs.jhu.edu/~razvanm/fs-expedition/hclust-example.png

Dendrograms

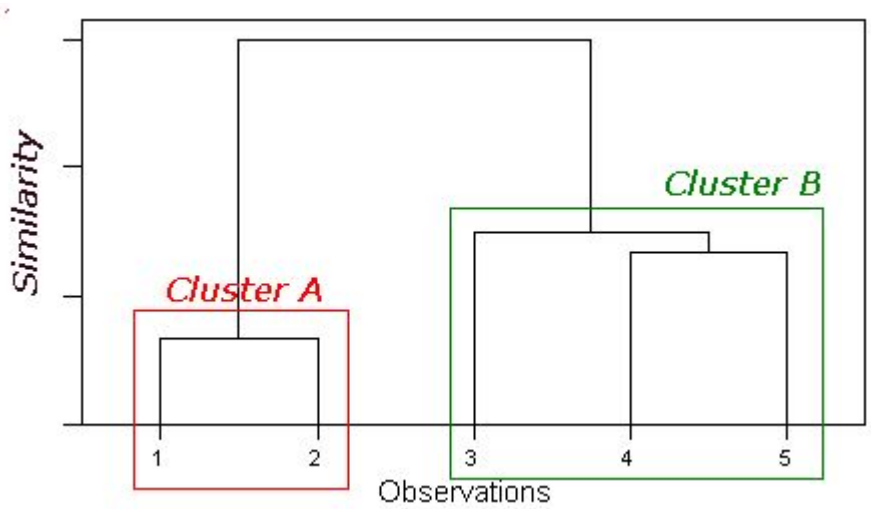
Visualizing hierarchical clustering

- Each width represents distance between clusters before joining
- Useful to estimate # of clusters



https://earlconf.com/2014/Images/Speakers/Tal_Clustered_Iris_dataset.jpg

Demo Time!

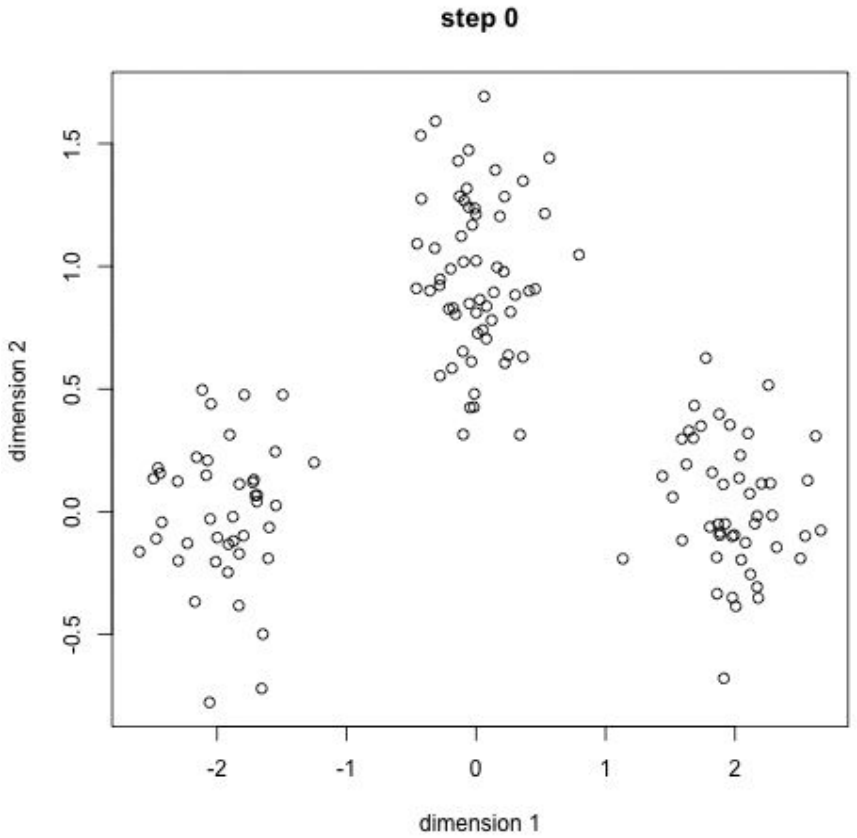


http://www.alanfielding.co.uk/multivar/images/dend5.gif

K-means Algorithm

Simplest clustering algorithm. Input parameter: *k*

1. Starts with *k* random centroids
2. Cluster points using “centroids”
3. Take average of clustered points
4. Use as new centroids
5. Repeat until convergence



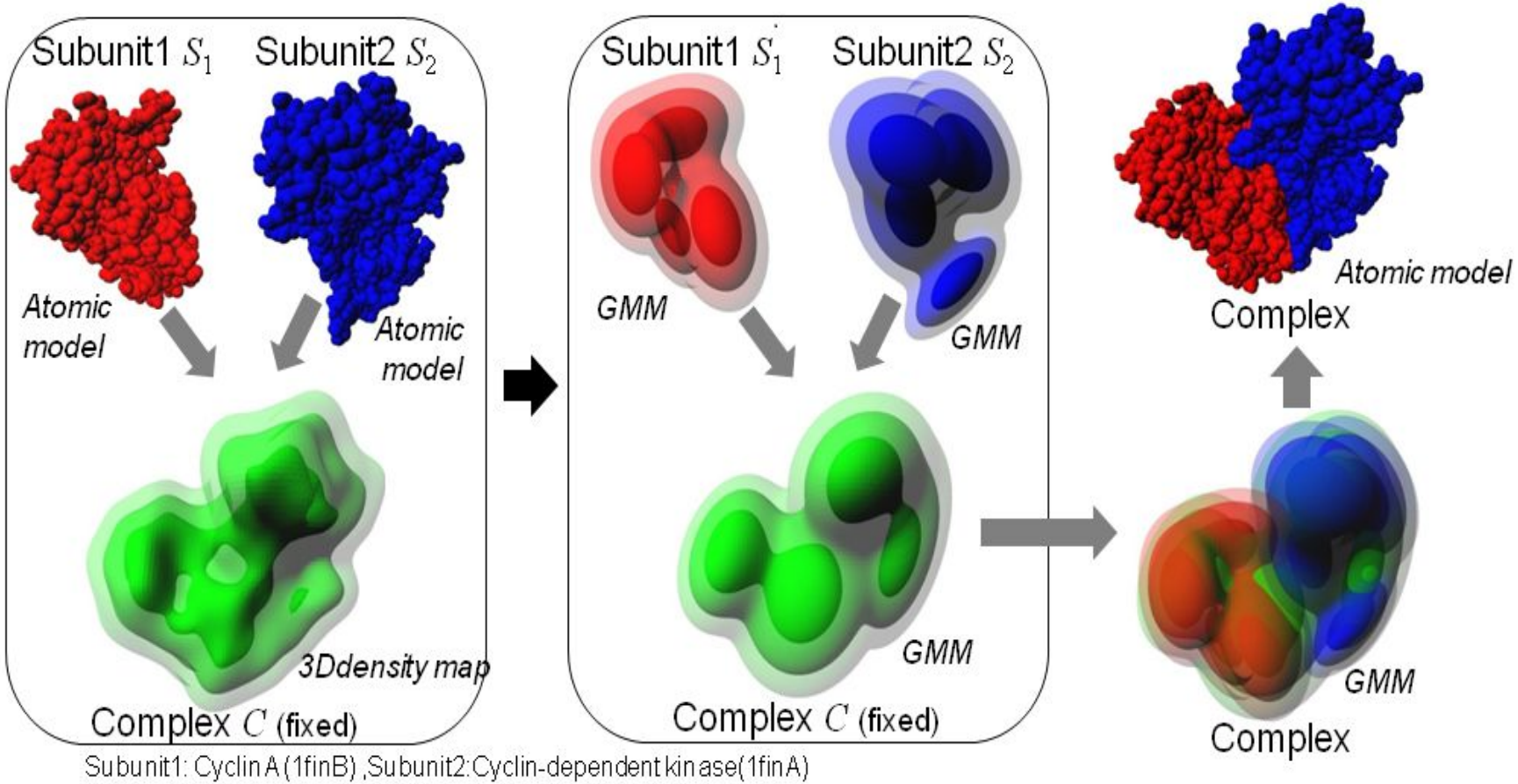
http://www.turingfinance.com/wp-content/uploads/2015/02/K-Means-Clustering-Gif.gif

Demo Time!

Gaussian Mixture Model

Assumptions that the data is a mix of clusters:

- Clusters may overlap and “mix”
- Gaussian mixture models assume that each cluster is **normally distributed**
- May more accurately describe reality since boundaries are usually not clear cut



Subunit1: Cyclin A (1finB) ,Subunit2:Cyclin-dependent kinase(1finA)

Maximum Likelihood Estimator

Given observations, how likely is a certain set of parameters?

- Assumptions must be made on the probability distribution
- Obtain a function of maximum likelihood
- Obtain local maxima, minima (calculus)

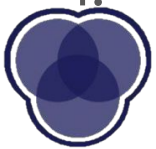
$$L(\mu,\sigma^2;x_1,\ldots,x_n) = \prod_{j=1}^{n} f_X(x_j;\mu,\sigma^2)$$
$$= \prod_{j=1}^{n} (2\pi\sigma^2)^{-1/2} \exp\left(-\frac{1}{2}\frac{(x_j-\mu)^2}{\sigma^2}\right)$$

https://www.statlect.com/images/normal-distribution-maximum-likelihood__9.png

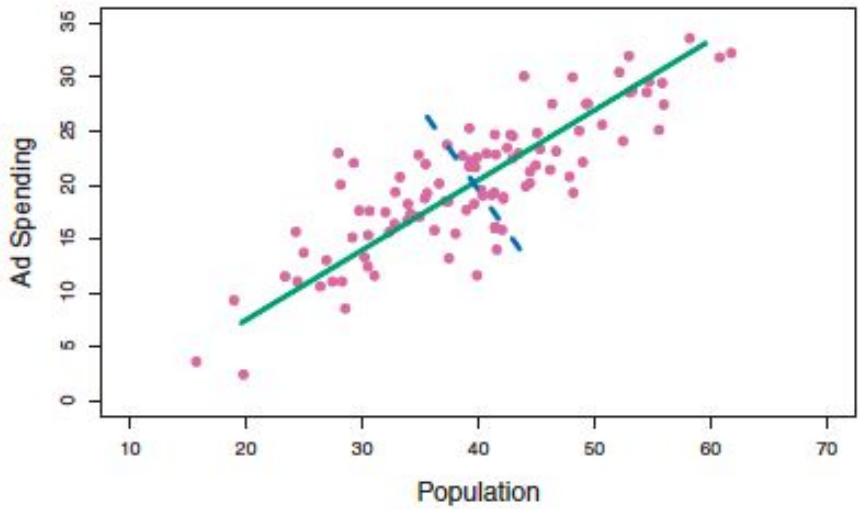
Expectation-Maximization Algorithm

A general unsupervised learning method for MLEs

1. Pick random values for parameters.
2. Make predictions based on the parameters.
3. Take these predictions as true, solve for most likely parameters. Repeat step 2 with these parameters.
4. Repeat until convergence.



Principal Component Analysis (PCA)



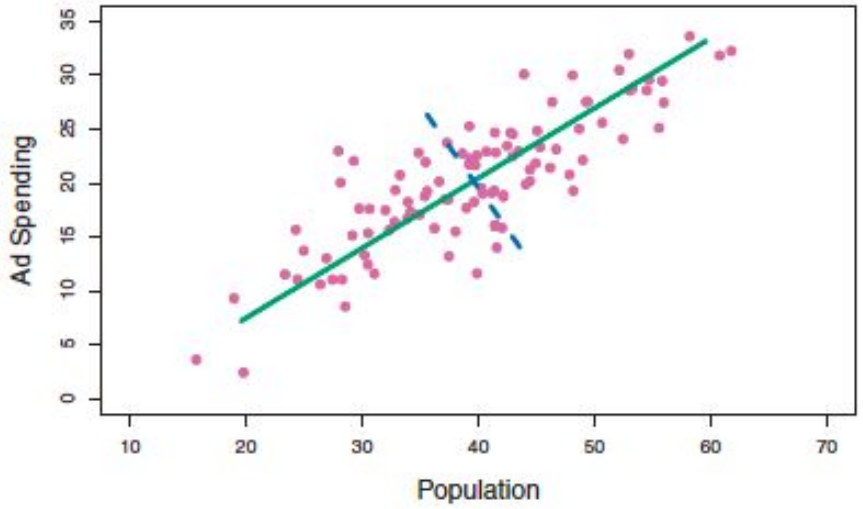
Want to understand the "direction" that our data goes in without storing whole data set.

1. Find the direction along which the data has the largest variance (projections of all data points are the largest).

   Called the **first principal component** (in green at left).

Hastie, Trevor, et al. "An Introduction to Statistical Learning.

Principal Components



2. Find the direction which is orthogonal to the first principal component and has the largest variance (projections of points are largest).

This is the **second principal component**.

Garath, James, et al. “An Introduction to Statistical Learning in R.”

Principal Components

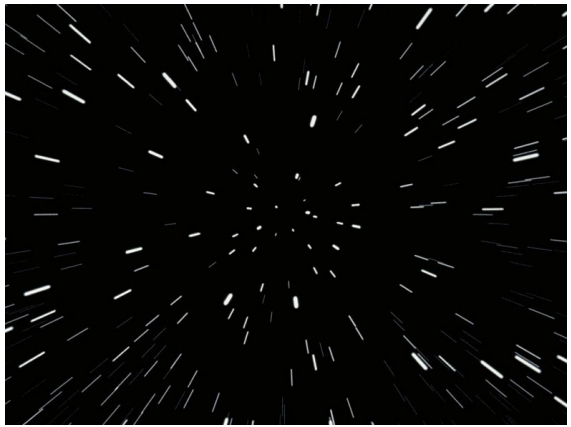
Generally, *n* dimensional data can have *n* principal components.

Principal component analysis - process of constructing these components (orthogonal directions of largest variance)

Why?

PCA is used for two things:

- Exploratory data analysis for unsupervised learning (what are the general trends?)
- Obtaining a low-dimensional approximation for high dimensional data (thousands of features)



http://vignette3.wikia.nocookie.net/starwars/images/7/72/Hyperspace-SWPuzzle.jpg/revision/latest?cb=20101018151244

PCA Demo Time

We'll compute the two principal components of a two-dimensional data set using the `prcomp` function.

We'll then visualize the components using the `plot` function.

Coming Up

Your problem set: None

Next week: Leveling up as a data scientist.

See you then!