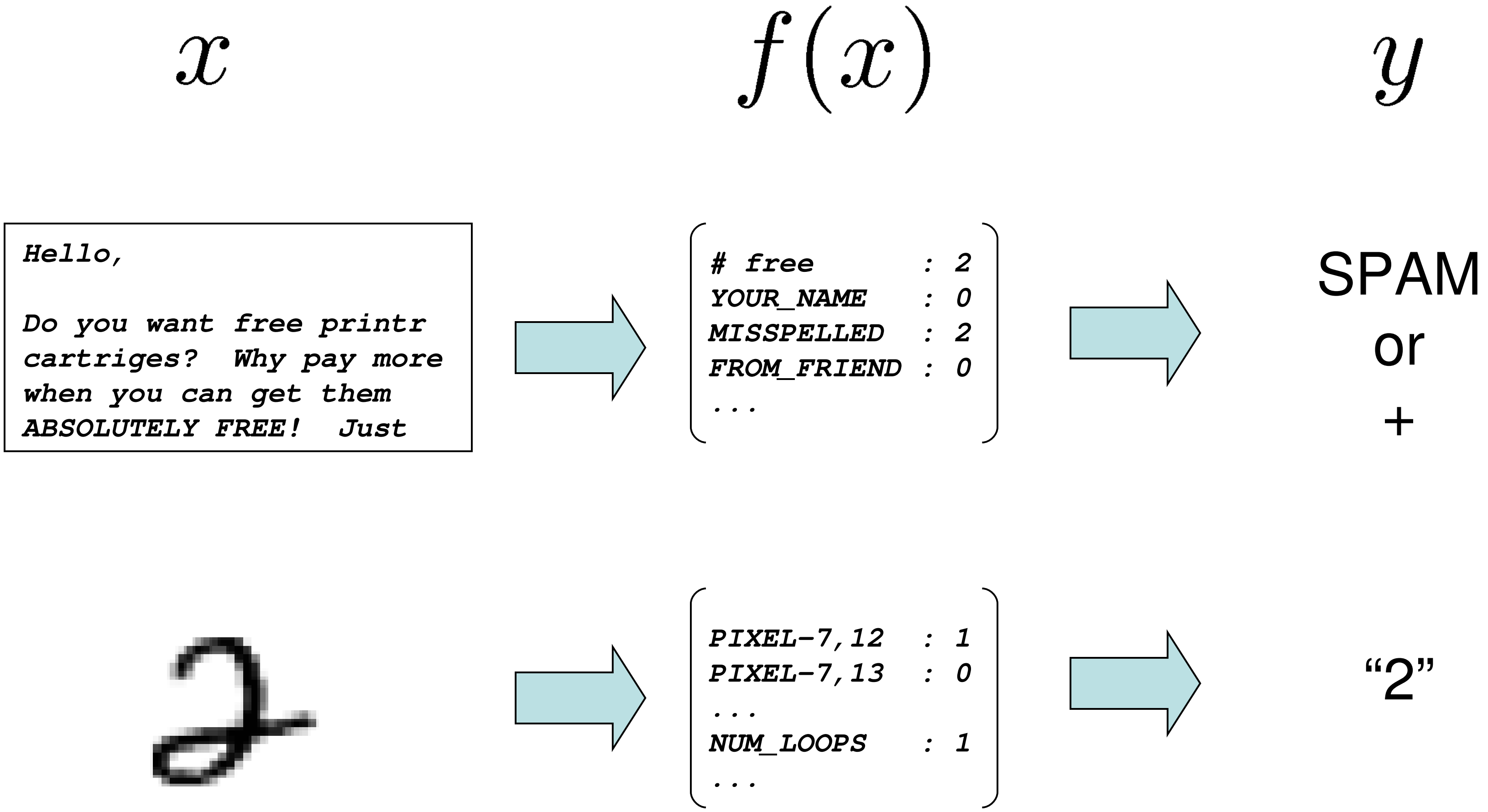
Classification: Feature Vectors



x $f(x)$ y

```
Hello,

Do you want free printr
cartriges?  Why pay more
when you can get them
ABSOLUTELY FREE!  Just
```

```
# free       : 2
YOUR_NAME    : 0
MISSPELLED   : 2
FROM_FRIEND  : 0
...
```

SPAM
or
+

```
PIXEL-7,12   : 1
PIXEL-7,13   : 0
...
NUM_LOOPS    : 1
...
```

“2”

This slide deck courtesy of Dan Klein at UC Berkeley

Properties of Perceptrons

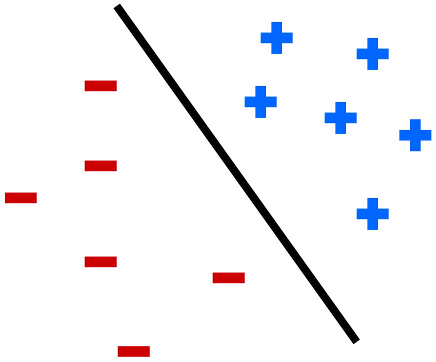
- Separability: some parameters get the training set perfectly correct

- Convergence: if the training is separable, perceptron will eventually converge (binary case)

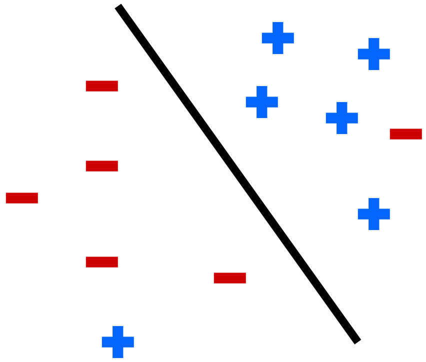
- Mistake Bound: the maximum number of mistakes (binary case) related to the *margin* or degree of separability

$$\text{mistakes} < \frac{k}{\delta^2}$$

Separable

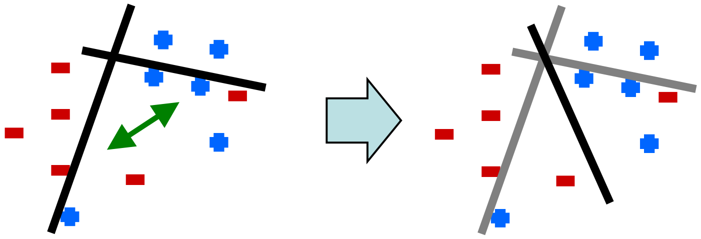


Non-Separable

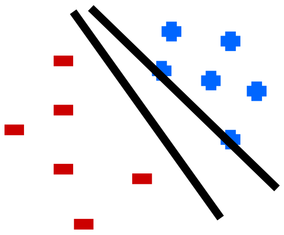


Problems with the Perceptron

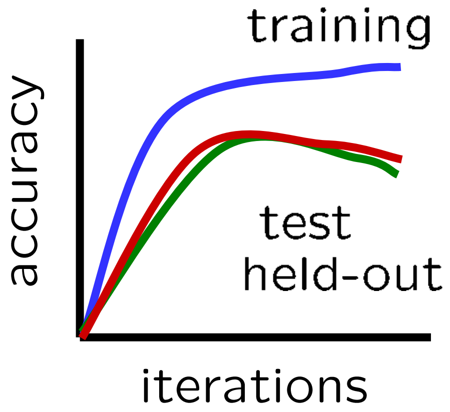
- Noise: if the data isn't separable, weights might thrash
  - Averaging weight vectors over time can help (averaged perceptron)



- Mediocre generalization: finds a "barely" separating solution



- Overtraining: test / held-out accuracy usually rises, then falls
  - Overtraining is a kind of overfitting



Fixing the Perceptron

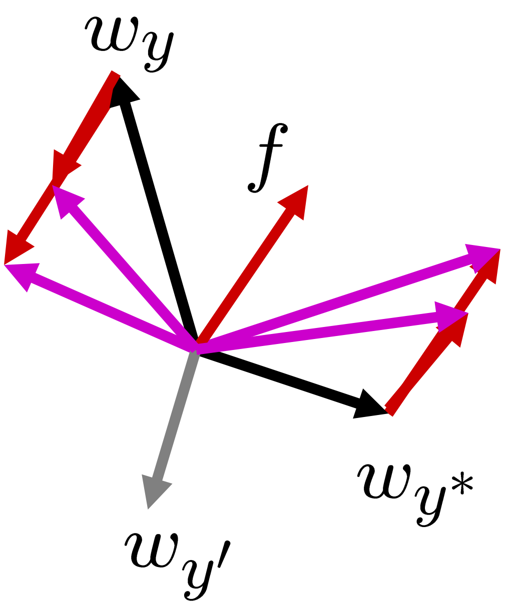
- Idea: adjust the weight update to mitigate these effects

- MIRA*: choose an update size that fixes the current mistake…
- … but, minimizes the change to w

$$\min_{w} \ \frac{1}{2}\sum_{y} ||w_y - w'_y||^2$$

$$w_{y^*} \cdot f(x) \geq w_y \cdot f(x) + 1$$

- The +1 helps to generalize



Guessed y instead of y^* on example x with features $f(x)$

$$w_y = w'_y - \tau f(x)$$
$$w_{y^*} = w'_{y^*} + \tau f(x)$$

[*] Margin Infused Relaxed Algorithm

Minimum Correcting Update

$$\min_{w} \frac{1}{2}\sum_{y} ||w_y - w'_y||^2$$

$$w_{y^*} \cdot f \geq w_y \cdot f + 1$$

$$\boxed{\begin{aligned} w_y &= w'_y - \tau f(x) \\ w_{y^*} &= w'_{y^*} + \tau f(x) \end{aligned}}$$

$$\Downarrow$$

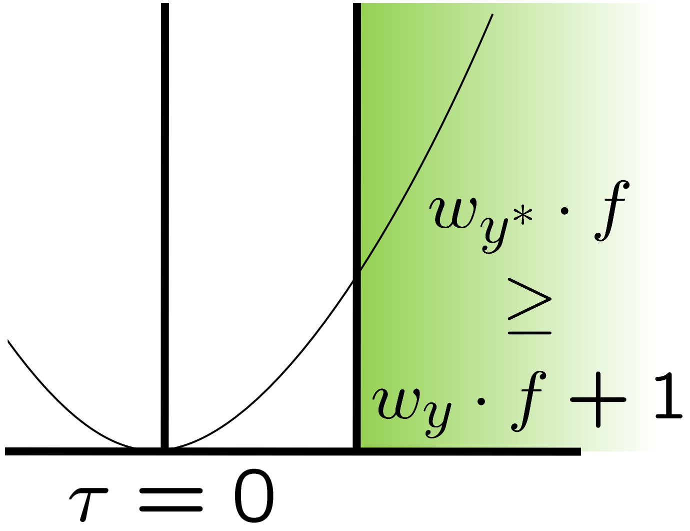
$$\min_{\tau} \; ||\tau f||^2$$

$$w_{y^*} \cdot f \geq w_y \cdot f + 1$$

$$\Downarrow$$

$$(w'_{y^*} + \tau f) \cdot f = (w'_y - \tau f) \cdot f + 1$$

$$\tau = \frac{(w'_y - w'_{y^*}) \cdot f + 1}{2 f \cdot f}$$



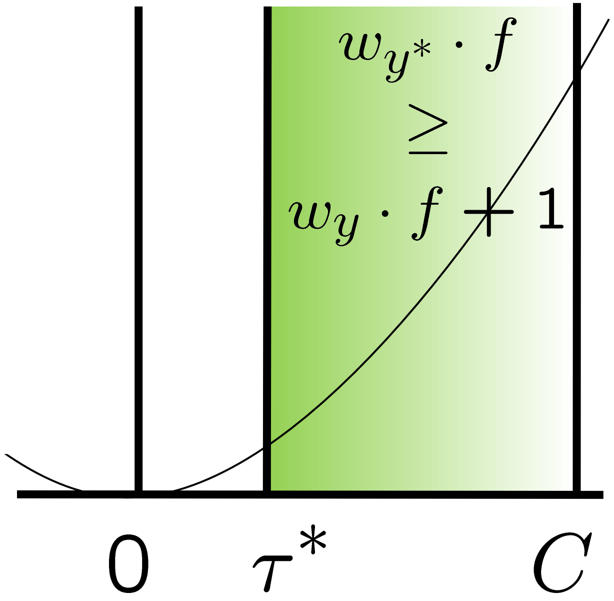
min not τ =0, or would not have made an error, so min will be where equality holds

Maximum Step Size

- In practice, it's also bad to make updates that are too large
  - Example may be labeled incorrectly
  - You may not have enough features
  - Solution: cap the maximum possible value of τ with some constant C

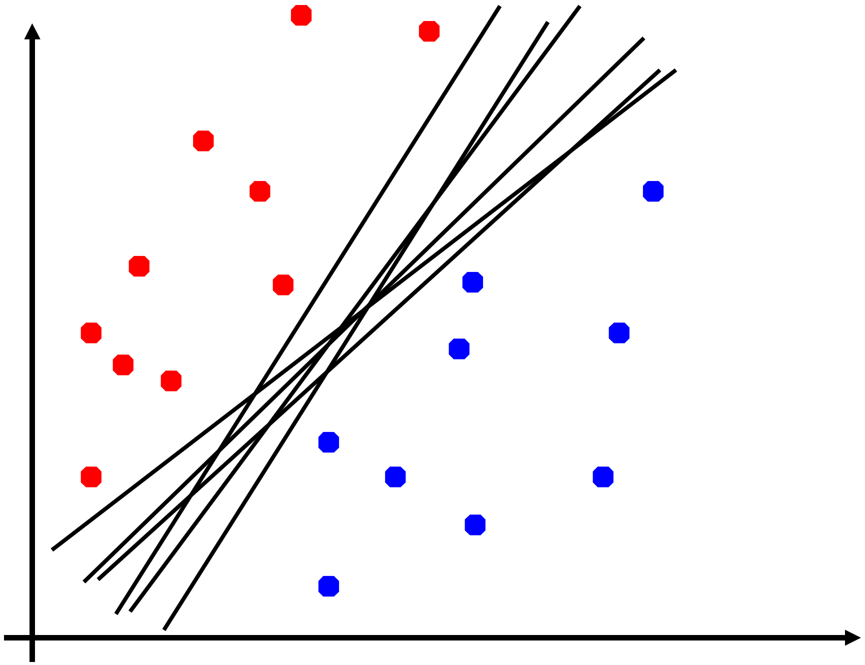
$$\tau^* = \min\left(\frac{(w'_y - w'_{y^*}) \cdot f + 1}{2f \cdot f}, C\right)$$

  - Corresponds to an optimization that assumes non-separable data
  - Usually converges faster than perceptron
  - Usually better, especially on noisy data



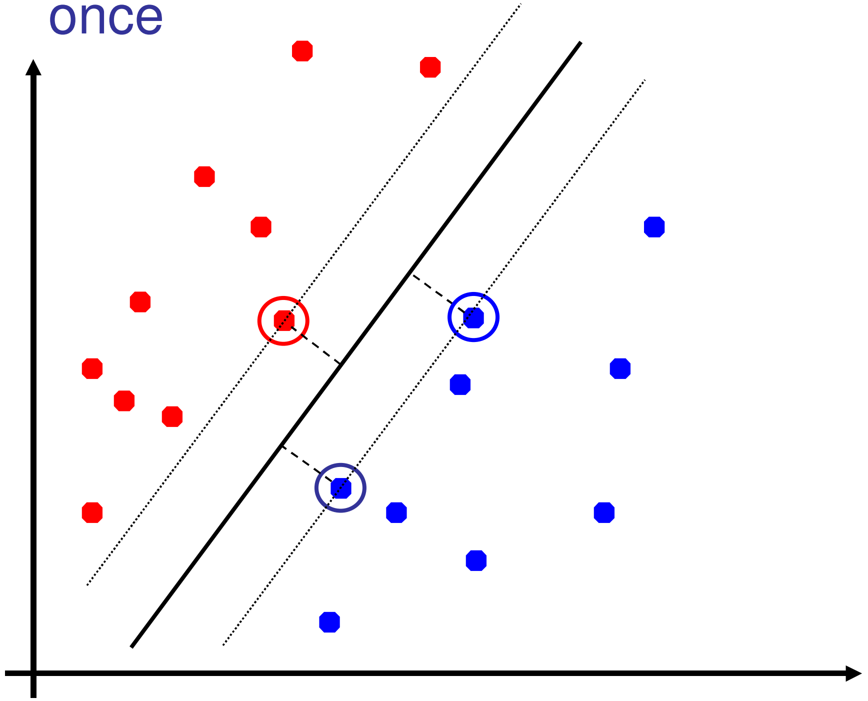
Linear Separators

- Which of these linear separators is optimal?



Support Vector Machines

- Maximizing the margin: good according to intuition, theory, practice
- Only support vectors matter; other training examples are ignorable
- Support vector machines (SVMs) find the separator with max margin
- Basically, SVMs are MIRA where you optimize over all examples at once



MIRA

$$\min_{w} \frac{1}{2}||w - w'||^2$$

$$w_{y^*} \cdot f(x_i) \geq w_y \cdot f(x_i) + 1$$

SVM

$$\min_{w} \frac{1}{2}||w||^2$$

$$\forall i, y \; w_{y^*} \cdot f(x_i) \geq w_y \cdot f(x_i) + 1$$

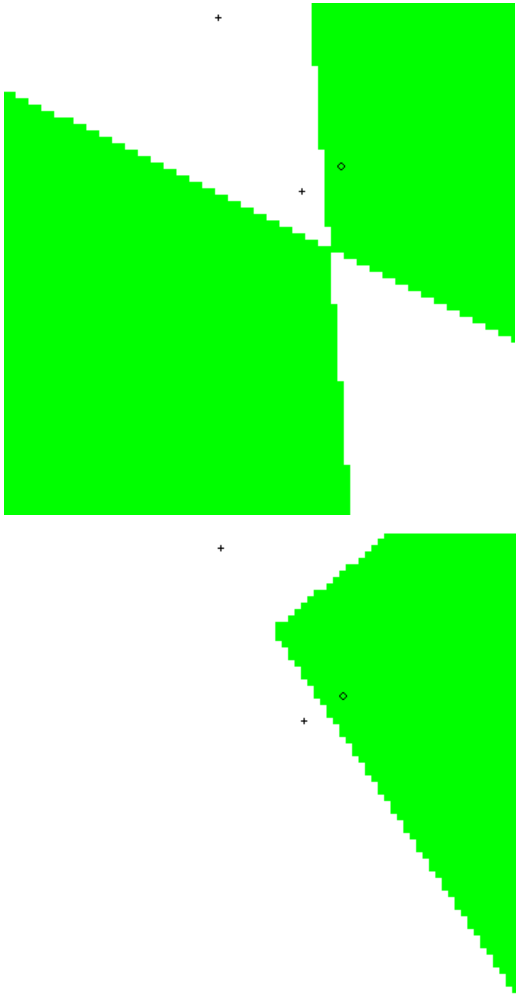
Classification: Comparison

- Naïve Bayes
  - Builds a model training data
  - Gives prediction probabilities
  - Strong assumptions about feature independence
  - One pass through data (counting)

- Perceptrons / MIRA:
  - Makes less assumptions about data
  - Mistake-driven learning
  - Multiple passes through data (prediction)
  - Often more accurate

Case-Based Reasoning

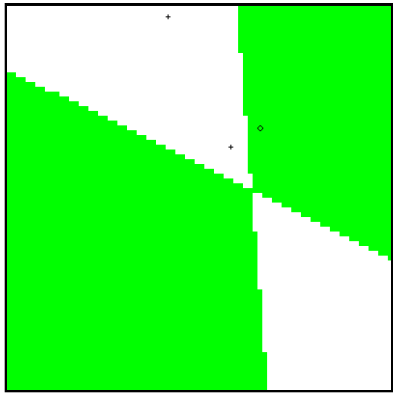
- Similarity for classification
  - Case-based reasoning
  - Predict an instance's label using similar instances
- Nearest-neighbor classification
  - 1-NN: copy the label of the most similar data point
  - K-NN: let the k nearest neighbors vote (have to devise a weighting scheme)
  - Key issue: how to define similarity
  - Trade-off:
    - Small k gives relevant neighbors
    - Large k gives smoother functions
    - Sound familiar?



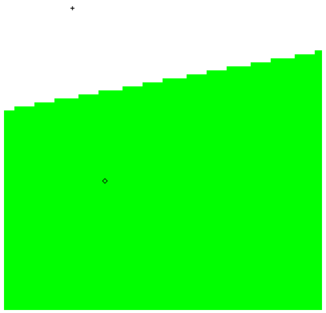
http://www.cs.cmu.edu/~zhuxj/courseproject/knndemo/KNN.htm

Parametric / Non-parametric

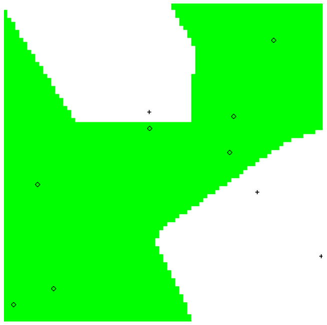
- Parametric models:
  - Fixed set of parameters
  - More data means better settings
- Non-parametric models:
  - Complexity of the classifier increases with data
  - Better in the limit, often worse in the non-limit
- (K)NN is non-parametric



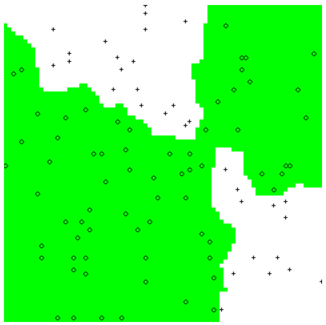
Truth



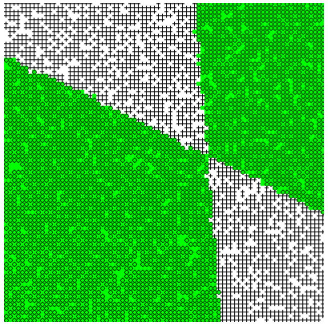
2 Examples



10 Examples



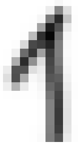
100 Examples

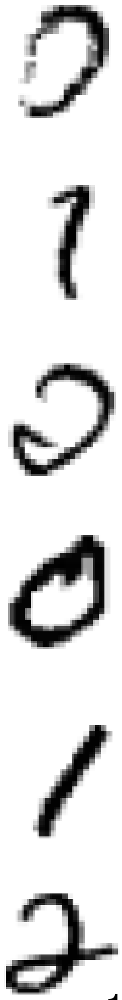


10000 Examples

Nearest-Neighbor Classification

- Nearest neighbor for digits:
  - Take new image
  - Compare to all training images
  - Assign based on closest example





- Encoding: image is vector of intensities:

$= \langle 0.0 \;\; 0.0 \;\; 0.3 \;\; 0.8 \;\; 0.7 \;\; 0.1 \ldots 0.0 \rangle$

- What's the similarity function?
  - Dot product of two images vectors?

$$\text{sim}(x, x') = x \cdot x' = \sum_i x_i x'_i$$

  - Usually normalize vectors so ||x|| = 1
  - min = 0 (when?), max = 1 (when?)

Basic Similarity

- Many similarities based on feature dot products:

$$\mathsf{sim}(x, x') = f(x) \cdot f(x') = \sum_i f_i(x) f_i(x')$$

- If features are just the pixels:

$$\mathsf{sim}(x, x') = x \cdot x' = \sum_i x_i x'_i$$

- Note: not all similarities are of this form

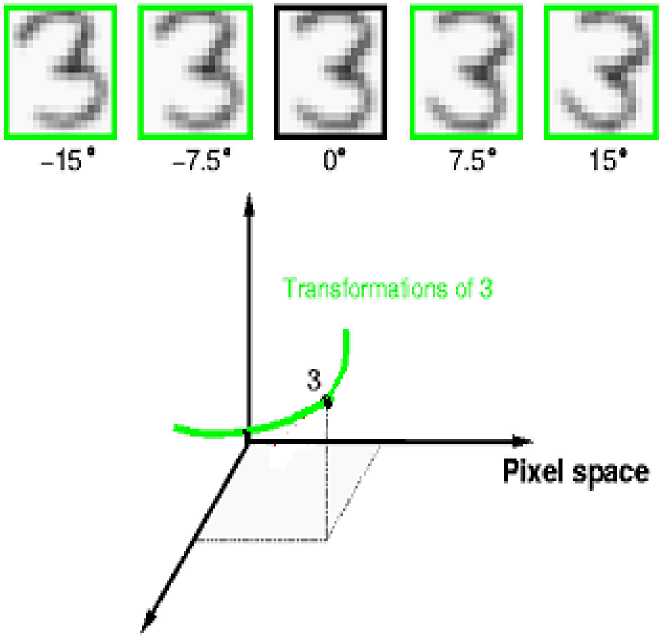
Invariant Metrics

- Better distances use knowledge about vision
- Invariant metrics:
  - Similarities are invariant under certain transformations
  - Rotation, scaling, translation, stroke-thickness…
  - E.g:

    - 16 x 16 = 256 pixels; a point in 256-dim space
    - Small similarity in R^{256} (why?)
  - How to incorporate invariance into similarities?

This and next few slides adapted from Xiao Hu, UIUC

Rotation Invariant Metrics

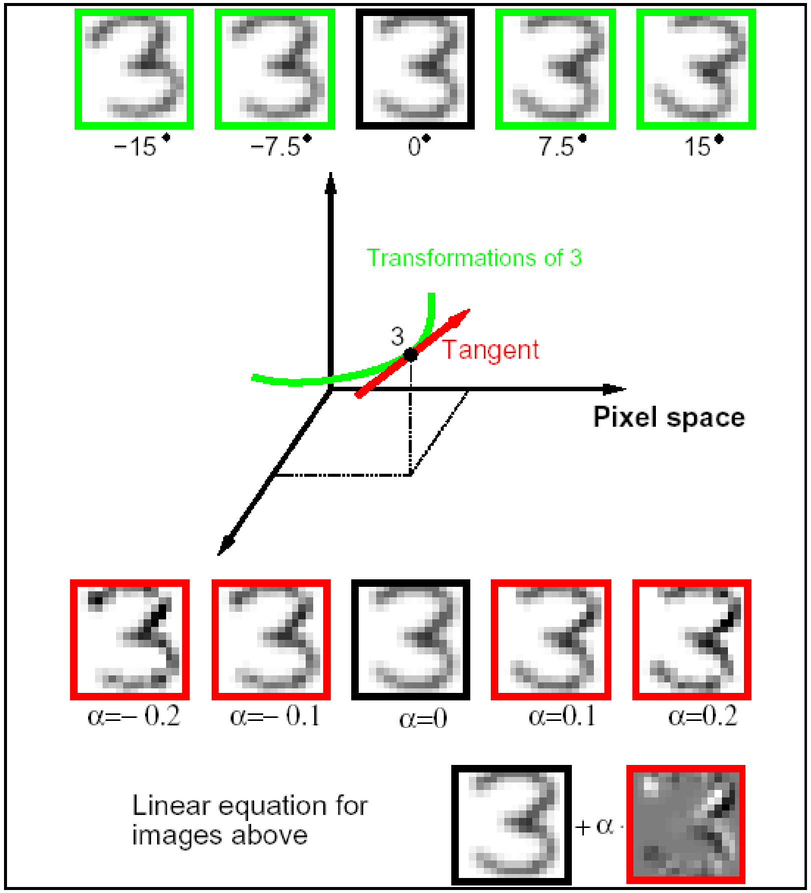


- Each example is now a curve in R^{256}
- Rotation invariant similarity:

s'=max s(r( ), r( ))

- E.g. highest similarity between images' rotation lines

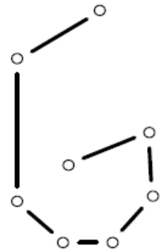
Tangent Families

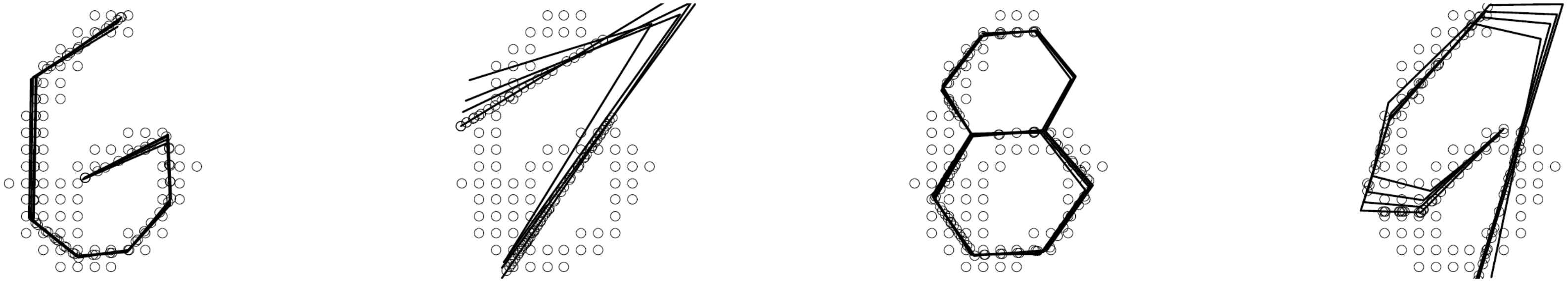


- Problems with s’:
  - Hard to compute
  - Allows large transformations (6 → 9)
- Tangent distance:
  - 1st order approximation at original points.
    - Easy to compute
    - Models small rotations

Template Deformation

- Deformable templates:
  - An “ideal” version of each category
  - Best-fit to image using min variance
  - Cost for high distortion of template
  - Cost for image points being far from distorted template
- Used in many commercial digit recognizers





Examples from [Hastie 94]

A Tale of Two Approaches…

- Nearest neighbor-like approaches
  - Can use fancy similarity functions
  - Don’t actually get to do explicit learning
- Perceptron-like approaches
  - Explicit training to reduce empirical error
  - Can’t use fancy similarity, only linear
  - Or can they? Let’s find out!

Perceptron Weights

- What is the final value of a weight w_y of a perceptron?
  - Can it be any real vector?
  - No! It's built by adding up inputs.

$$w_y = \mathbf{0} + f(x_1) - f(x_5) + \dots$$

$$w_y = \sum_i \alpha_{i,y} f(x_i)$$

- Can reconstruct weight vectors (the primal representation) from update counts (the dual representation)

$$\alpha_y = \langle \alpha_{1,y} \;\; \alpha_{2,y} \;\; \dots \;\; \alpha_{n,y} \rangle$$

Dual Perceptron

- How to classify a new example x?

$$
\begin{aligned}
\text{score}(y,x) &= w_y \cdot f(x) \\
&= \left(\sum_i \alpha_{i,y}\, f(x_i)\right) \cdot f(x) \\
&= \sum_i \alpha_{i,y}\, (f(x_i) \cdot f(x)) \\
&= \sum_i \alpha_{i,y}\; K(x_i, x)
\end{aligned}
$$

- If someone tells us the value of K for each pair of examples, never need to build the weight vectors!

Dual Perceptron

- Start with zero counts (alpha)
- Pick up training instances one by one
- Try to classify x_n,

$$y = \arg\max_y \sum_i \alpha_{i,y} K(x_i, x)$$

- If correct, no change!
- If wrong: lower count of wrong class (for this instance), raise score of right class (for this instance)

$$\alpha_{y,n} = \alpha_{y,n} - 1 \qquad w_y = w_y - f(x)$$

$$\alpha_{y^*,n} = \alpha_{y^*,n} + 1 \qquad w_{y^*} = w_{y^*} + f(x)$$

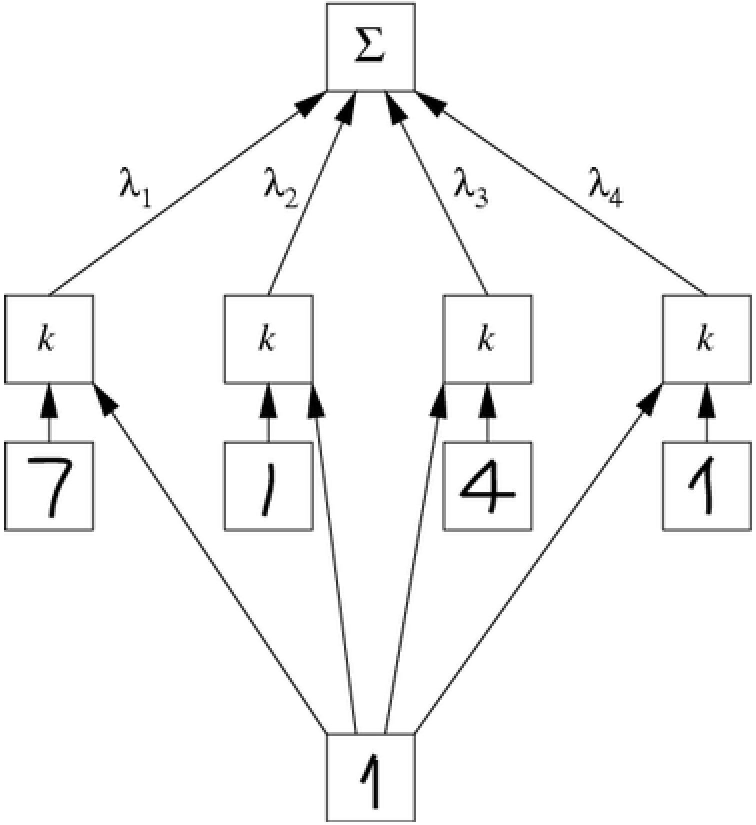
Kernelized Perceptron

- If we had a black box (kernel) which told us the dot product of two examples x and y:
  - Could work entirely with the dual representation
  - No need to ever take dot products ("kernel trick")

$$\begin{aligned} \text{score}(y, x) &= w_y \cdot f(x) \\ &= \sum_i \alpha_{i,y} \; K(x_i, x) \end{aligned}$$

- Like nearest neighbor – work with black-box similarities
- Downside: slow if many examples get nonzero alpha

Kernelized Perceptron Structure



$$\sum = \text{score}(c, x)$$

$$\lambda_i = \alpha_{c,i}$$

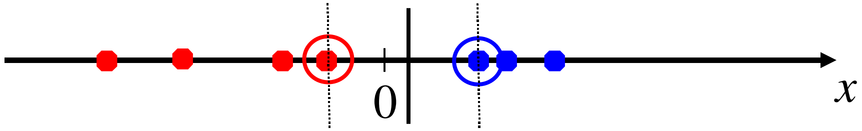
Kernels: Who Cares?

- So far: a very strange way of doing a very simple calculation
- “Kernel trick”: we can substitute any* similarity function in place of the dot product
- Lets us learn new kinds of hypothesis

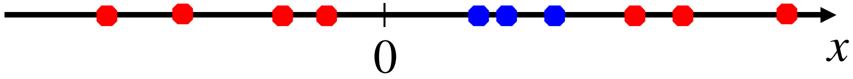
* Fine print: if your kernel doesn’t satisfy certain technical requirements, lots of proofs break. E.g. convergence, mistake bounds. In practice, illegal kernels *sometimes* work (but not always).

Non-Linear Separators

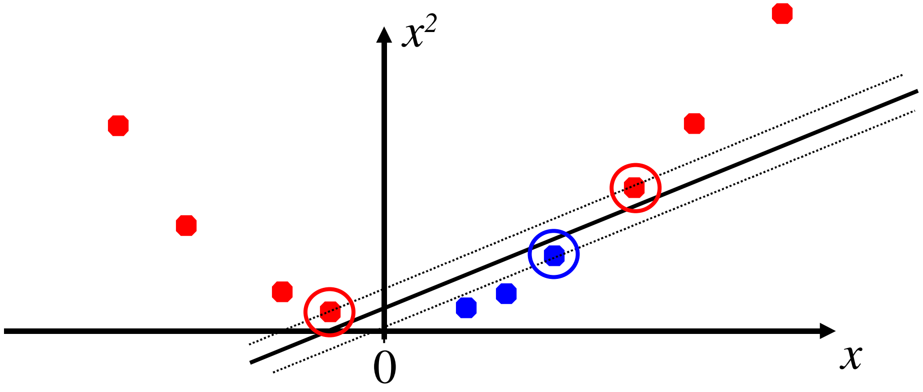
- Data that is linearly separable (with some noise) works out great:



- But what are we going to do if the dataset is just too hard?



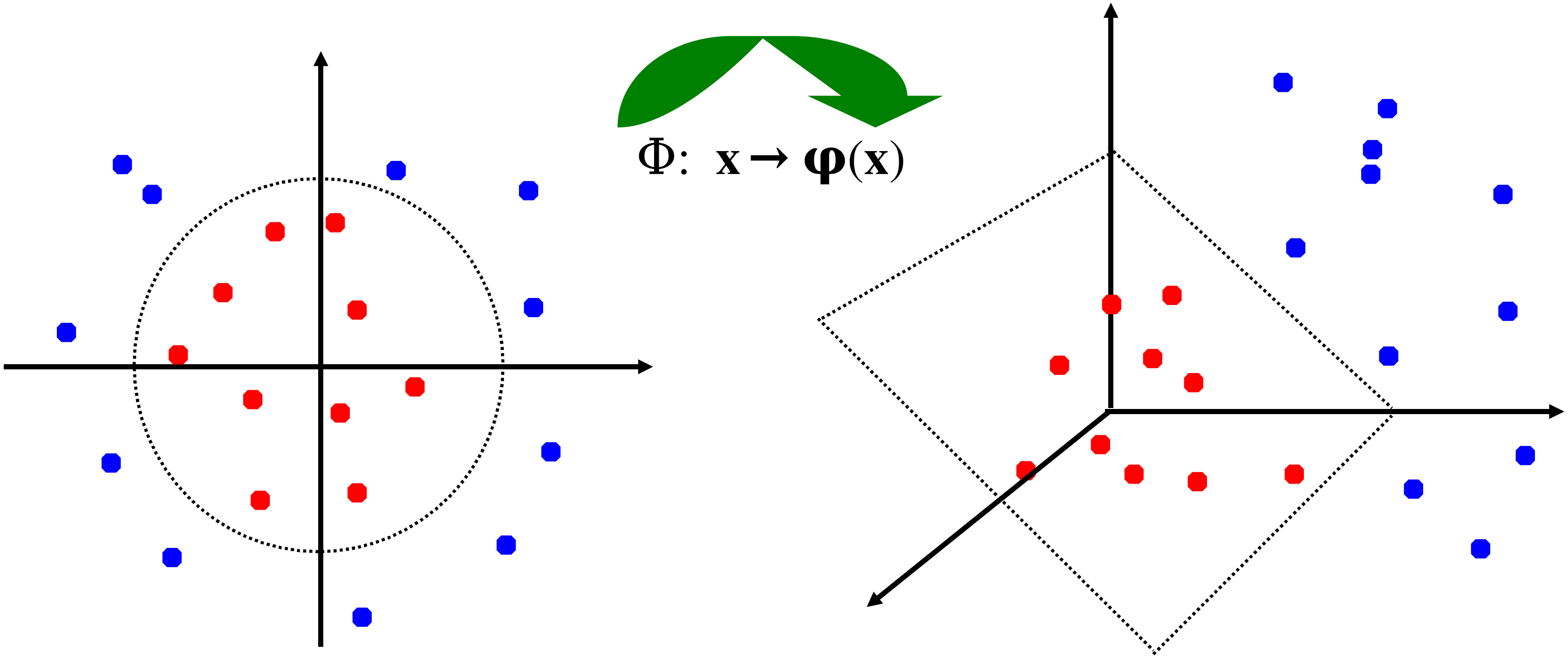
- How about… mapping data to a higher-dimensional space:



This and next few slides adapted from Ray Mooney, UT

Non-Linear Separators

- General idea: the original feature space can always be mapped to some higher-dimensional feature space where the training set is separable:



$\Phi:\ \mathbf{x} \rightarrow \boldsymbol{\varphi}(\mathbf{x})$

Some Kernels

- Kernels implicitly map original vectors to higher dimensional spaces, take the dot product there, and hand the result back

- Linear kernel: $K(x, x') = x' \cdot x' = \sum_i x_i \, x'_i$

- Quadratic kernel: $K(x, x') = (x \cdot x' + 1)^2$

$$= \sum_{i,j} x_i x_j \, x'_i x'_j + 2 \sum_i x_i \, x'_i + 1$$

- RBF: infinite dimensional representation

$$K(x, x') = \exp(-||x - x'||^2)$$

- Discrete kernels: e.g. string kernels

Why Kernels?

- Can’t you just add these features on your own (e.g. add all pairs of features instead of using the quadratic kernel)?
  - Yes, in principle, just compute them
  - No need to modify any algorithms
  - But, number of features can get large (or infinite)
  - Some kernels not as usefully thought of in their expanded representation, e.g. RBF or data-defined kernels [Henderson and Titov 05]

- Kernels let us compute with these features implicitly
  - Example: implicit dot product in quadratic kernel takes much less space and time per dot product
  - Of course, there’s the cost for using the pure dual algorithms: you need to compute the similarity to every training datum

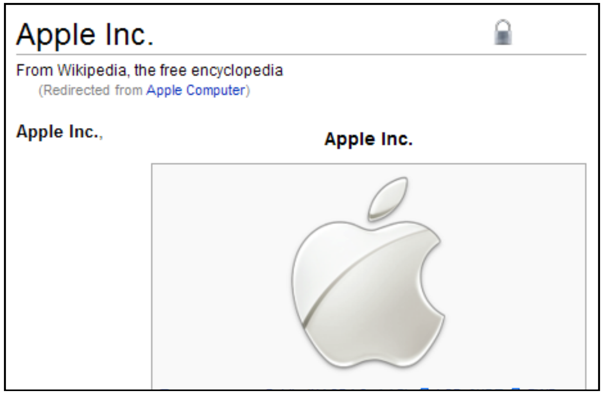
Recap: Classification

- Classification systems:
  - Supervised learning
  - Make a prediction given evidence
  - We've seen several methods for this
  - Useful when you have labeled data

Extension: Web Search

- Information retrieval:
  - Given information needs, produce information
  - Includes, e.g. web search, question answering, and classic IR
- Web search: not exactly classification, but rather ranking

x = “Apple Computers”



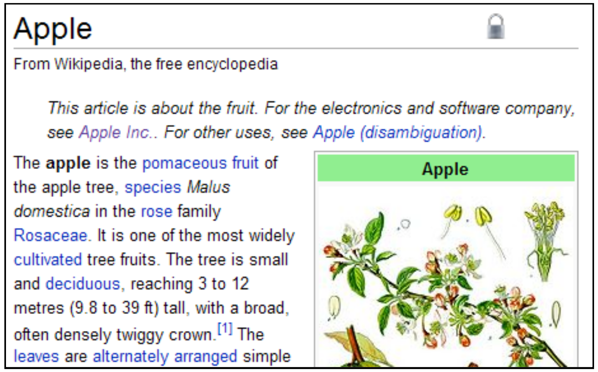
Apple Inc.

From Wikipedia, the free encyclopedia

(Redirected from Apple Computer)

Apple Inc.,

Apple Inc.



Apple

From Wikipedia, the free encyclopedia

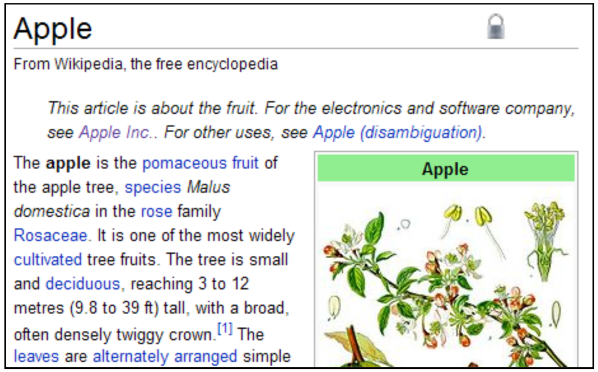
This article is about the fruit. For the electronics and software company, see Apple Inc.. For other uses, see Apple (disambiguation).

The **apple** is the pomaceous fruit of the apple tree, species *Malus domestica* in the rose family Rosaceae. It is one of the most widely cultivated tree fruits. The tree is small and deciduous, reaching 3 to 12 metres (9.8 to 39 ft) tall, with a broad, often densely twiggy crown.[1] The leaves are alternately arranged simple

Apple

Feature-Based Ranking

x = “Apple Computers”

$f(x,$  $) = [0.3\ 5\ 0\ 0\ \ \ldots]$

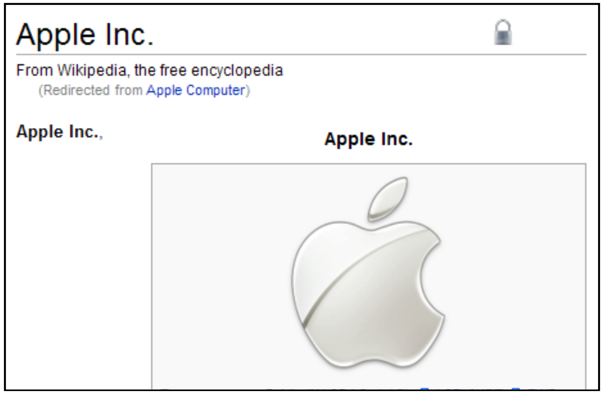
Apple

From Wikipedia, the free encyclopedia

This article is about the fruit. For the electronics and software company, see Apple Inc.. For other uses, see Apple (disambiguation).

The **apple** is the pomaceous fruit of the apple tree, species *Malus domestica* in the rose family Rosaceae. It is one of the most widely cultivated tree fruits. The tree is small and deciduous, reaching 3 to 12 metres (9.8 to 39 ft) tall, with a broad, often densely twiggy crown.[1] The leaves are alternately arranged simple

Apple

$f(x,$  $) = [0.8\ 4\ 2\ 1\ \ \ldots]$

Apple Inc.

From Wikipedia, the free encyclopedia

(Redirected from Apple Computer)

Apple Inc.,

Apple Inc.

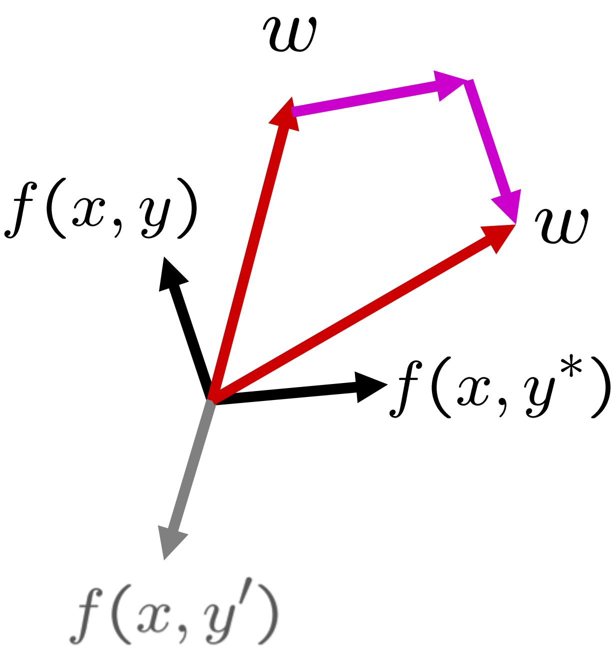
Perceptron for Ranking

- Inputs x
- Candidates y
- Many feature vectors: $f(x, y)$
- One weight vector: w
  - Prediction:

$$y = \arg\max_y \, w \cdot f(x, y)$$

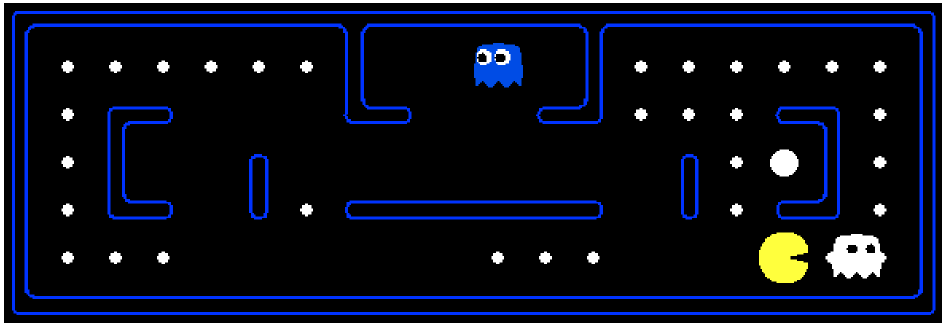
  - Update (if wrong):

$$w = w + f(x, y^*) - f(x, y)$$



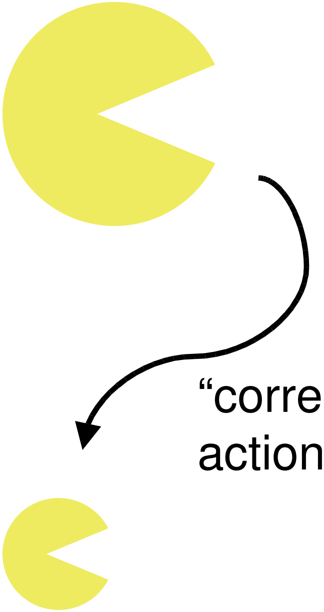
Pacman Apprenticeship!

- Examples are states s



- Candidates are pairs (s,a)
- “Correct” actions: those taken by expert
- Features defined over (s,a) pairs: f(s,a)
- Score of a q-state (s,a) given by:

$$w \cdot f(s,a)$$



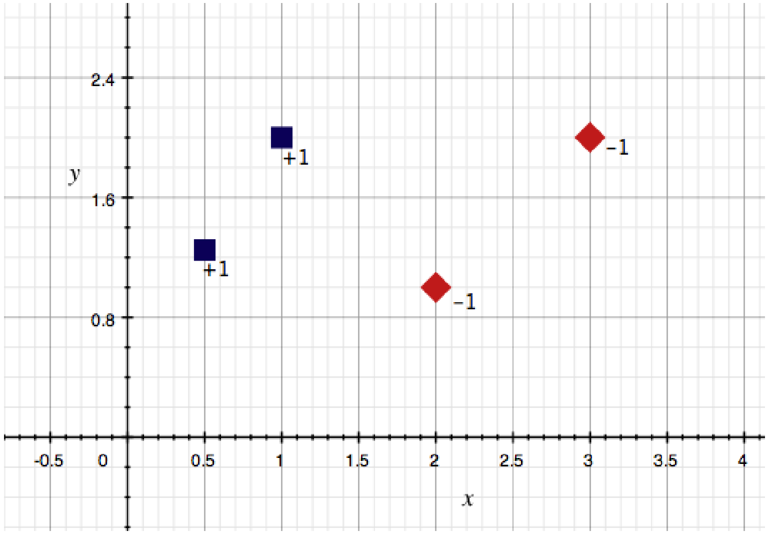
“correct”
action a*

$$\forall a \neq a^*, \\ w \cdot f(a^*) > w \cdot f(a)$$

- How is this VERY different from reinforcement learning?

Perceptron Example

Features	Label
(0.5,1.25)	+1
(1,2)	+1
(2,1)	-1
(3,2)	-1



Obvious problems with the perceptron

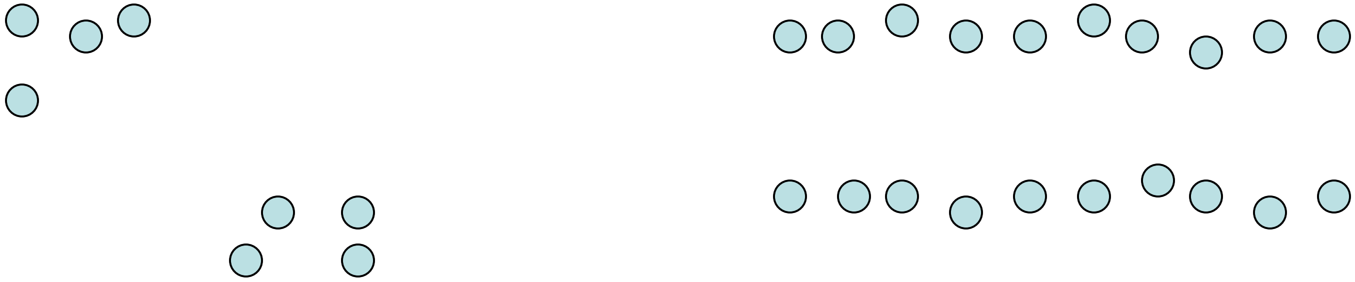
- Sometimes updates too much
  - Good weights can be corrupted by a single outlier datum
- Sometimes updates too little
  - Even after an update, the prediction can still be incorrect
- Assumes separable data
  - Real data is never separable

Clustering

- Clustering systems:
  - Unsupervised learning
  - Detect patterns in unlabeled data
    - E.g. group emails or search results
    - E.g. find categories of customers
    - E.g. detect anomalous program executions
  - Useful when don't know what you're looking for
  - Requires data, but no labels
  - Often get gibberish

Clustering

- Basic idea: group together similar instances
- Example: 2D point patterns

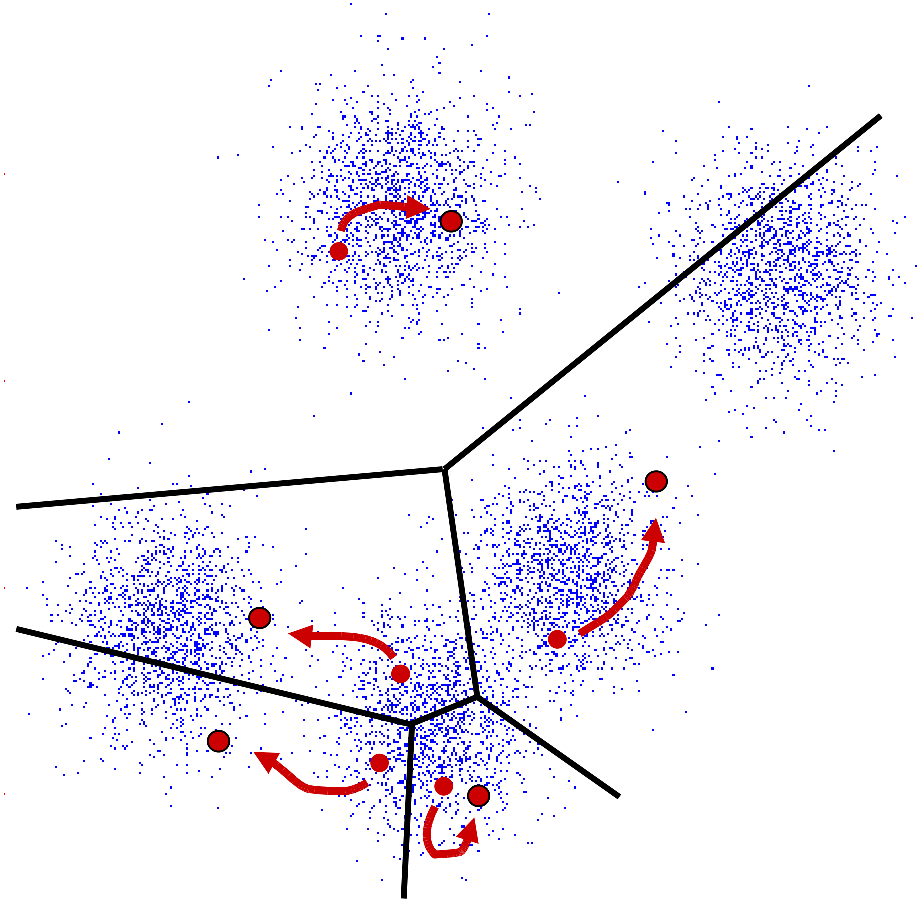


- What could “similar” mean?
  - One option: small (squared) Euclidean distance

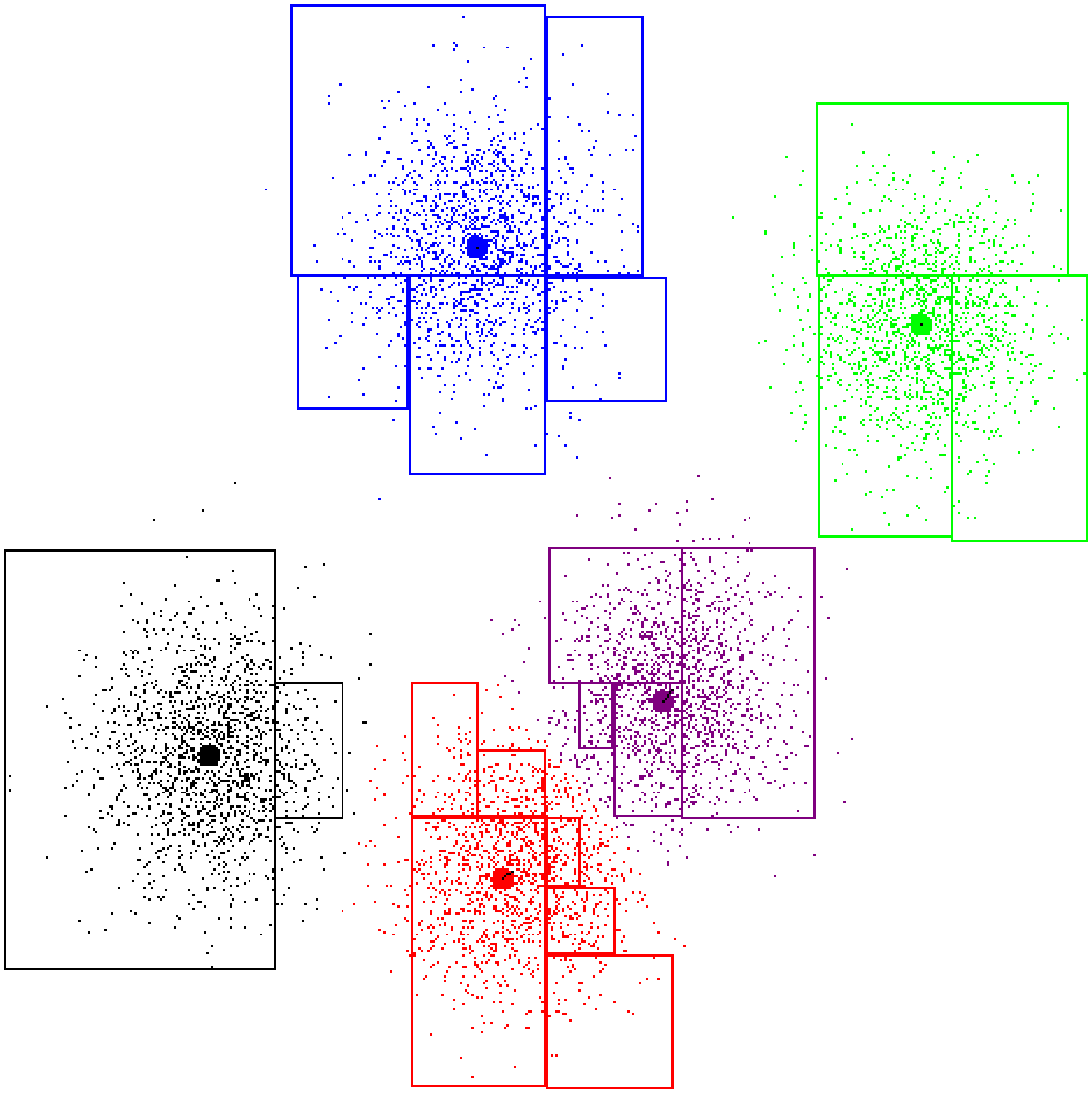
$$\mathrm{dist}(x, y) = (x - y)^{\top}(x - y) = \sum_i (x_i - y_i)^2$$

K-Means

- An iterative clustering algorithm
  - Pick K random points as cluster centers (means)
  - Alternate:
    - Assign data instances to closest mean
    - Assign each mean to the average of its assigned points
  - Stop when no points' assignments change



K-Means Example



Example: K-Means

- [web demo]
  - http://www.cs.washington.edu/research/image

K-Means as Optimization

- Consider the total distance to the means:

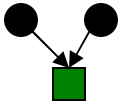
$$\phi(\{x_i\},\{a_i\},\{c_k\}) = \sum_i \text{dist}(x_i, c_{a_i})$$

points

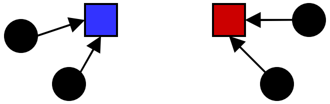
assignments

means

- Each iteration reduces phi



- Two stages each iteration:
  - Update assignments: fix means c,
    change assignments a
  - Update means: fix assignments a,
    change means c



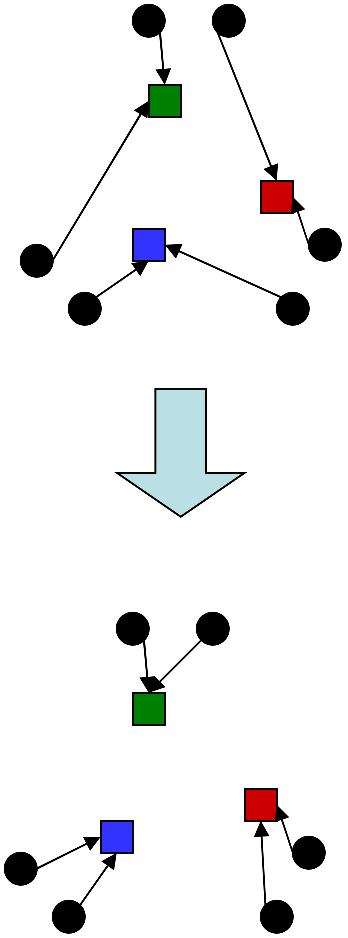
Phase I: Update Assignments

- For each point, re-assign to closest mean:

$$a_i = \underset{k}{\operatorname{argmin}} \operatorname{dist}(x_i, c_k)$$

- Can only decrease total distance phi!

$$\phi(\{x_i\}, \{a_i\}, \{c_k\}) = \sum_i \operatorname{dist}(x_i, c_{a_i})$$



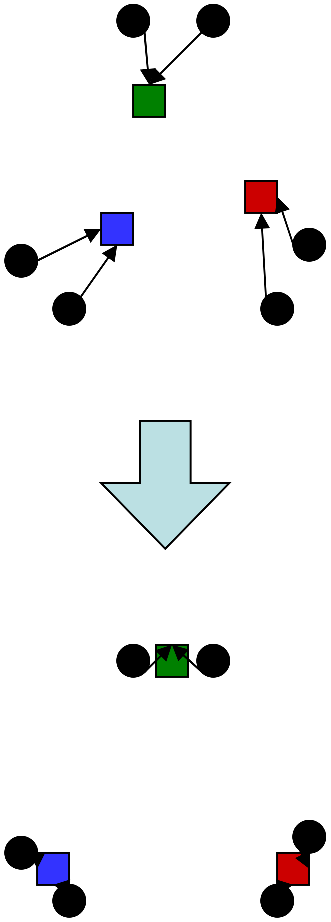
Phase II: Update Means

- Move each mean to the average of its assigned points:

$$c_k = \frac{1}{|\{i : a_i = k\}|} \sum_{i:a_i=k} x_i$$

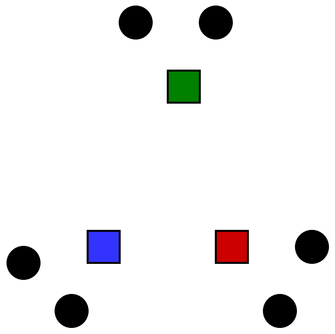
- Also can only decrease total distance… (Why?)

- Fun fact: the point y with minimum squared Euclidean distance to a set of points {x} is their mean

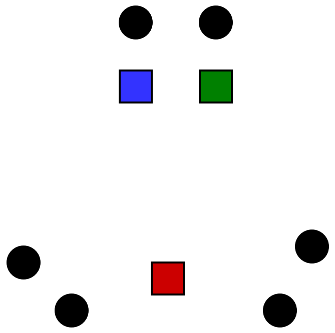


Initialization

- K-means is non-deterministic
  - Requires initial means
  - It does matter what you pick!

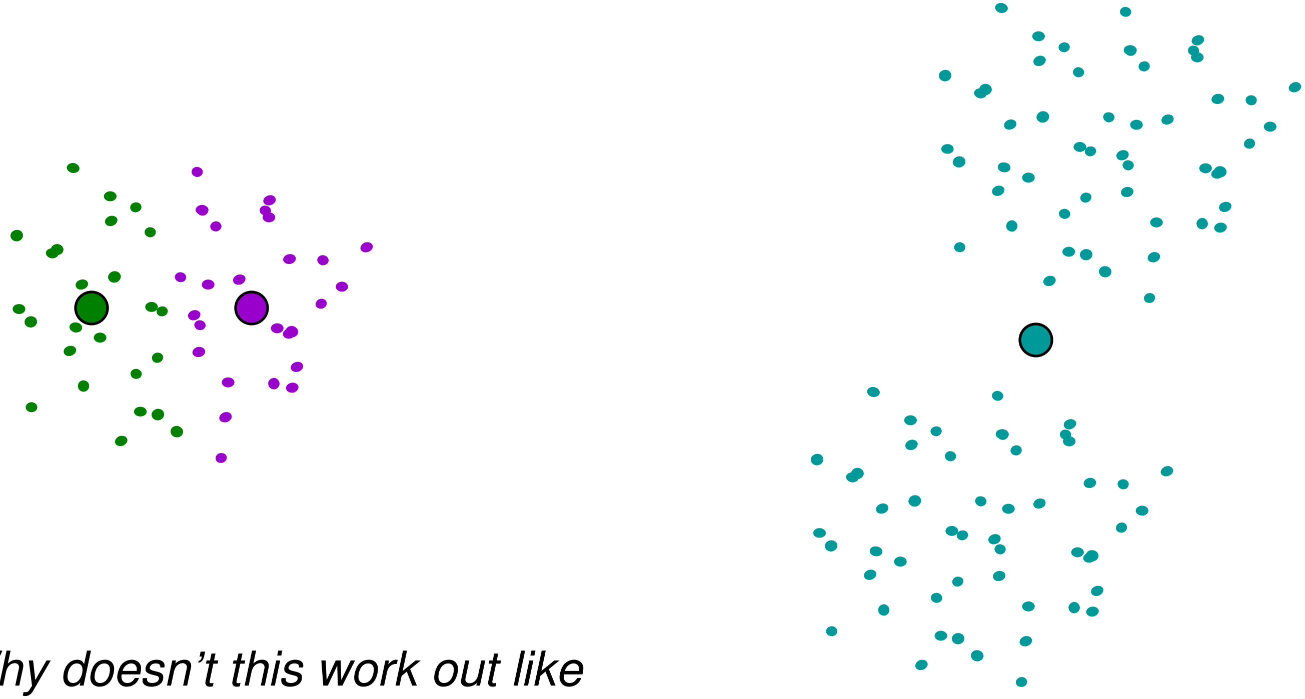


  - What can go wrong?
  - Various schemes for preventing this kind of thing: variance-based split / merge, initialization heuristics



K-Means Getting Stuck

- A local optimum:



Why doesn’t this work out like the earlier example, with the purple taking over half the blue?

K-Means Questions

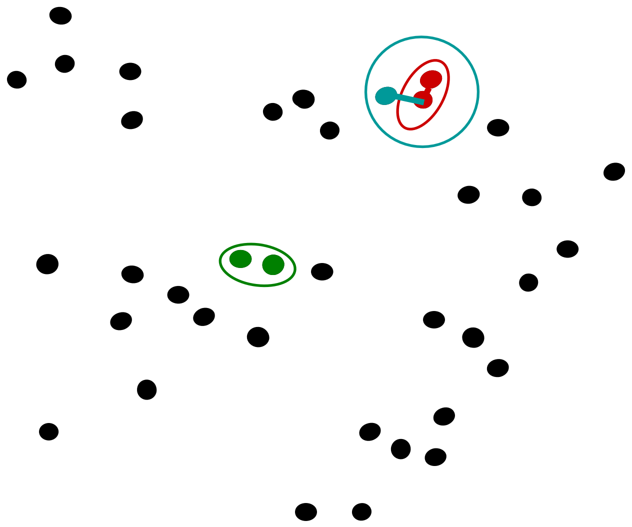
- Will K-means converge?
  - To a global optimum?
- Will it always find the true patterns in the data?
  - If the patterns are very very clear?
- Will it find something interesting?
- Do people ever use it?
- How many clusters to pick?

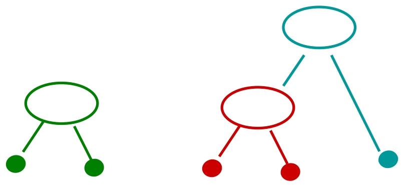
Agglomerative Clustering

- Agglomerative clustering:
  - First merge very similar instances
  - Incrementally build larger clusters out of smaller clusters

- Algorithm:
  - Maintain a set of clusters
  - Initially, each instance in its own cluster
  - Repeat:
    - Pick the two closest clusters
    - Merge them into a new cluster
    - Stop when there's only one cluster left

- Produces not one clustering, but a family of clusterings represented by a dendrogram



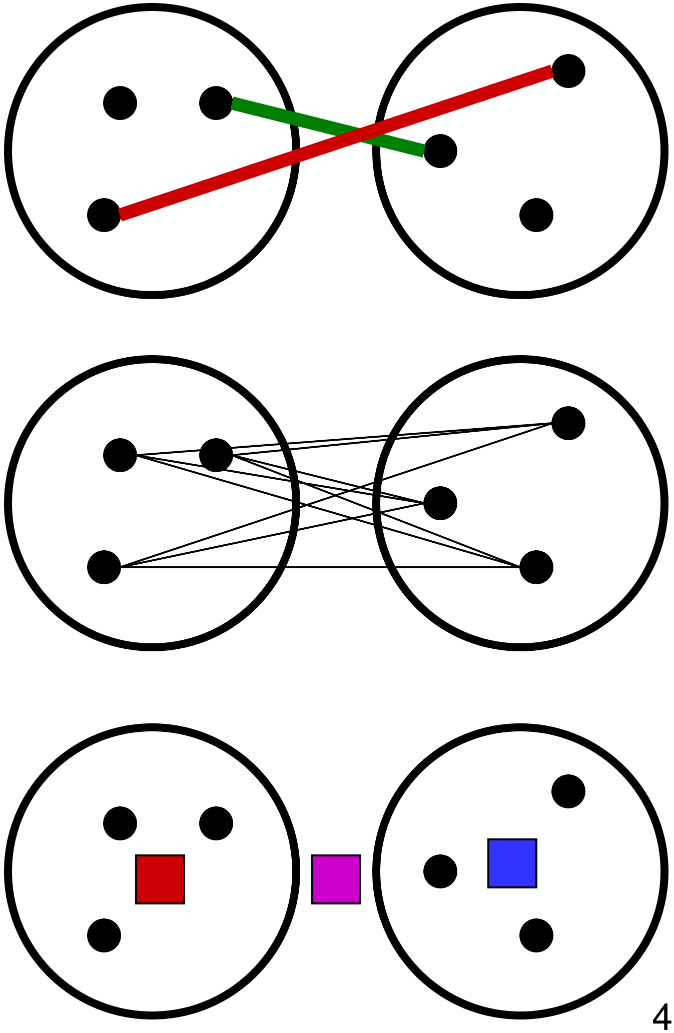


Agglomerative Clustering

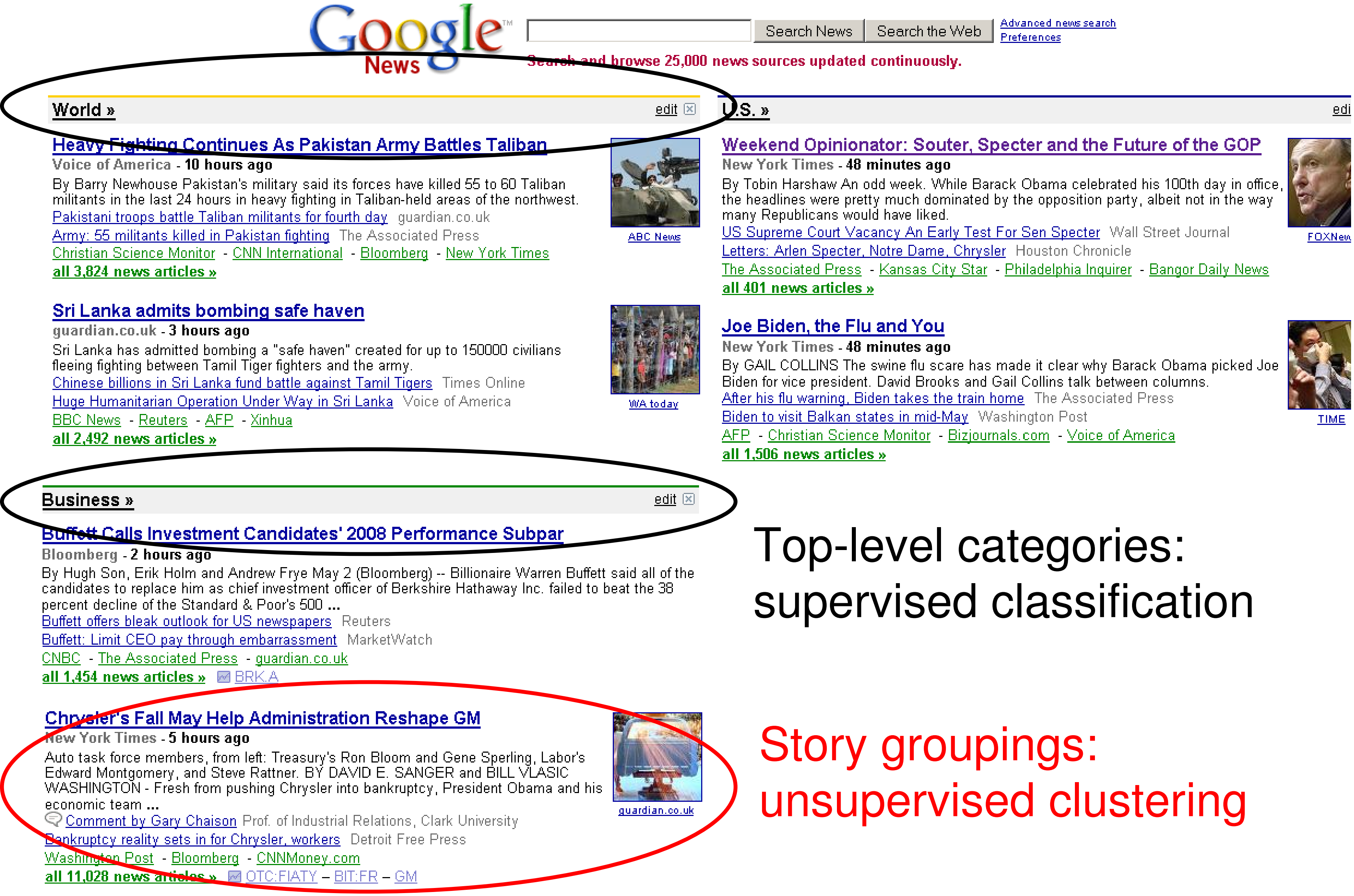
- How should we define "closest" for clusters with multiple elements?

- Many options
  - Closest pair (single-link clustering)
  - Farthest pair (complete-link clustering)
  - Average of all pairs
  - Ward's method (min variance, like k-means)

- Different choices create different clustering behaviors



Clustering Application



Search News | Search the Web | Advanced news search | Preferences

Search and browse 25,000 news sources updated continuously.

World » edit

Heavy Fighting Continues As Pakistan Army Battles Taliban
Voice of America - **10 hours ago**
By Barry Newhouse Pakistan's military said its forces have killed 55 to 60 Taliban militants in the last 24 hours in heavy fighting in Taliban-held areas of the northwest.
Pakistani troops battle Taliban militants for fourth day guardian.co.uk
Army: 55 militants killed in Pakistan fighting The Associated Press
Christian Science Monitor - CNN International - Bloomberg - New York Times
all 3,824 news articles »
ABC News

Sri Lanka admits bombing safe haven
guardian.co.uk - **3 hours ago**
Sri Lanka has admitted bombing a "safe haven" created for up to 150000 civilians fleeing fighting between Tamil Tiger fighters and the army.
Chinese billions in Sri Lanka fund battle against Tamil Tigers Times Online
Huge Humanitarian Operation Under Way in Sri Lanka Voice of America
BBC News - Reuters - AFP - Xinhua
all 2,492 news articles »
WA today

U.S. » edi

Weekend Opinionator: Souter, Specter and the Future of the GOP
New York Times - **48 minutes ago**
By Tobin Harshaw An odd week. While Barack Obama celebrated his 100th day in office, the headlines were pretty much dominated by the opposition party, albeit not in the way many Republicans would have liked.
US Supreme Court Vacancy An Early Test For Sen Specter Wall Street Journal
Letters: Arlen Specter, Notre Dame, Chrysler Houston Chronicle
The Associated Press - Kansas City Star - Philadelphia Inquirer - Bangor Daily News
all 401 news articles »
FOXNew

Joe Biden, the Flu and You
New York Times - **48 minutes ago**
By GAIL COLLINS The swine flu scare has made it clear why Barack Obama picked Joe Biden for vice president. David Brooks and Gail Collins talk between columns.
After his flu warning, Biden takes the train home The Associated Press
Biden to visit Balkan states in mid-May Washington Post
AFP - Christian Science Monitor - Bizjournals.com - Voice of America
all 1,506 news articles »
TIME

Business » edit

Buffett Calls Investment Candidates' 2008 Performance Subpar
Bloomberg - **2 hours ago**
By Hugh Son, Erik Holm and Andrew Frye May 2 (Bloomberg) -- Billionaire Warren Buffett said all of the candidates to replace him as chief investment officer of Berkshire Hathaway Inc. failed to beat the 38 percent decline of the Standard & Poor's 500 **...**
Buffett offers bleak outlook for US newspapers Reuters
Buffett: Limit CEO pay through embarrassment MarketWatch
CNBC - The Associated Press - guardian.co.uk
all 1,454 news articles » BRK.A

Chrysler's Fall May Help Administration Reshape GM
New York Times - **5 hours ago**
Auto task force members, from left: Treasury's Ron Bloom and Gene Sperling, Labor's Edward Montgomery, and Steve Rattner. BY DAVID E. SANGER and BILL VLASIC WASHINGTON - Fresh from pushing Chrysler into bankruptcy, President Obama and his economic team **...**
Comment by Gary Chaison Prof. of Industrial Relations, Clark University
Bankruptcy reality sets in for Chrysler, workers Detroit Free Press
Washington Post - Bloomberg - CNNMoney.com
all 11,028 news articles » OTC:FIATY – BIT:FR – GM
guardian.co.uk

Top-level categories: supervised classification

Story groupings: unsupervised clustering