Convex Optimization

Statistics & Machine Learning

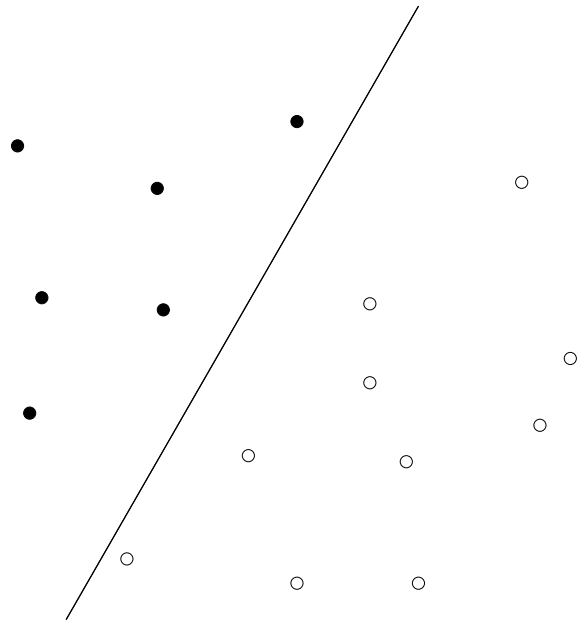
Today

- Classification
- Maximum likelihood estimation
- Optimal detector design
- Experiment design
- Collaborative prediction
- Transportation problems

Linear discrimination

Separate two sets of points $\{x_1, \ldots, x_N\}$, $\{y_1, \ldots, y_M\}$ by a hyperplane:

$$a^T x_i + b_i > 0, \quad i = 1, \ldots, N, \qquad a^T y_i + b_i < 0, \quad i = 1, \ldots, M$$



homogeneous in a, b, hence equivalent to

$$a^T x_i + b_i \geq 1, \quad i = 1, \ldots, N, \qquad a^T y_i + b_i \leq -1, \quad i = 1, \ldots, M$$

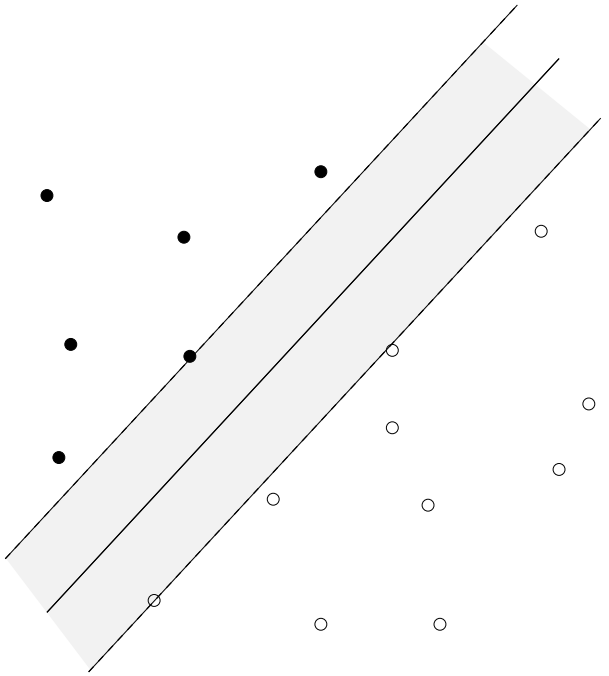
a set of linear inequalities in a, b

Robust linear discrimination

(Euclidean) distance between hyperplanes

$$\begin{aligned} \mathcal{H}_1 &= \{z \mid a^T z + b = 1\} \\ \mathcal{H}_2 &= \{z \mid a^T z + b = -1\} \end{aligned}$$

is $\mathbf{dist}(\mathcal{H}_1, \mathcal{H}_2) = 2/\|a\|_2$



to separate two sets of points by maximum margin,

$$\begin{array}{ll} \text{minimize} & (1/2)\|a\|_2 \\ \text{subject to} & a^T x_i + b \geq 1, \quad i = 1, \ldots, N \\ & a^T y_i + b \leq -1, \quad i = 1, \ldots, M \end{array} \tag{1}$$

(after squaring objective) a QP in a, b

Lagrange dual of maximum margin separation problem

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T\lambda + \mathbf{1}^T\mu \\ \text{subject to} & 2\left\|\sum_{i=1}^N \lambda_i x_i - \sum_{i=1}^M \mu_i y_i\right\|_2 \leq 1 \\ & \mathbf{1}^T\lambda = \mathbf{1}^T\mu, \quad \lambda \succeq 0, \quad \mu \succeq 0 \end{array} \tag{2}$$

from duality, optimal value is inverse of maximum margin of separation

interpretation

- change variables to $\theta_i = \lambda_i/\mathbf{1}^T\lambda$, $\gamma_i = \mu_i/\mathbf{1}^T\mu$, $t = 1/(\mathbf{1}^T\lambda + \mathbf{1}^T\mu)$
- invert objective to minimize $1/(\mathbf{1}^T\lambda + \mathbf{1}^T\mu) = t$

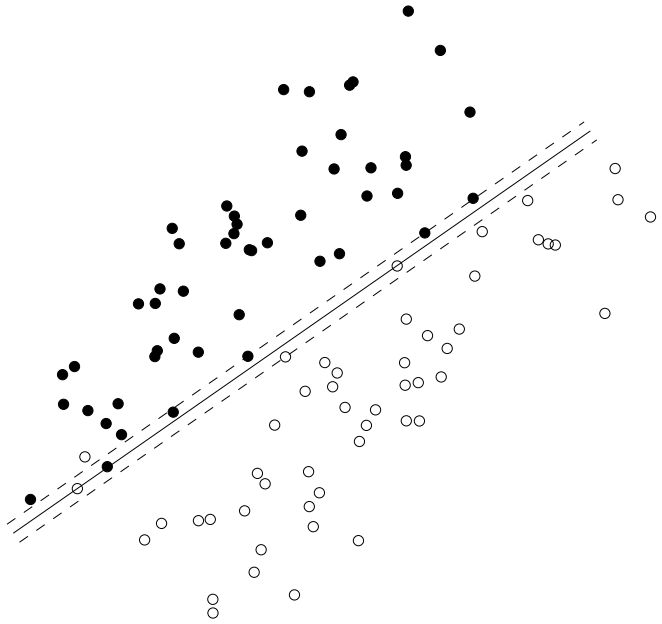
$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \left\|\sum_{i=1}^N \theta_i x_i - \sum_{i=1}^M \gamma_i y_i\right\|_2 \leq t \\ & \theta \succeq 0, \quad \mathbf{1}^T\theta = 1, \quad \gamma \succeq 0, \quad \mathbf{1}^T\gamma = 1 \end{array}$$

optimal value is distance between convex hulls

Approximate linear separation of non-separable sets

$$
\begin{array}{ll}
\text{minimize} & \mathbf{1}^T u + \mathbf{1}^T v \\
\text{subject to} & a^T x_i + b \geq 1 - u_i, \quad i = 1, \ldots, N \\
& a^T y_i + b \leq -1 + v_i, \quad i = 1, \ldots, M \\
& u \succeq 0, \quad v \succeq 0
\end{array}
$$

- an LP in a, b, u, v
- at optimum, $u_i = \max\{0, 1 - a^T x_i - b\}$, $v_i = \max\{0, 1 + a^T y_i + b\}$
- can be interpreted as a heuristic for minimizing #misclassified points

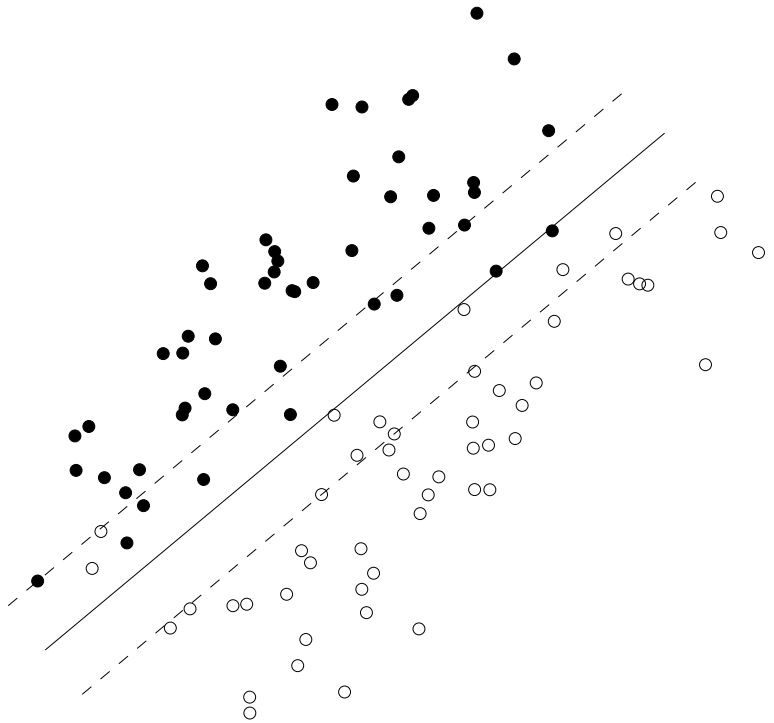


Support vector classifier

$$
\begin{array}{ll}
\text{minimize} & \|a\|_2 + C(\mathbf{1}^T u + \mathbf{1}^T v) \\
\text{subject to} & a^T x_i + b \geq 1 - u_i, \quad i = 1, \ldots, N \\
& a^T y_i + b \leq -1 + v_i, \quad i = 1, \ldots, M \\
& u \succeq 0, \quad v \succeq 0
\end{array}
$$

produces point on trade-off curve between inverse of margin $2/\|a\|_2$ and classification error, controlled by $C > 0$, measured by total slack $\mathbf{1}^T u + \mathbf{1}^T v$

same example as previous page, with $C = 0.1$:



Nonlinear discrimination

Separate two sets of points by a nonlinear function:

$$f(x_i) > 0, \quad i = 1, \ldots, N, \qquad f(y_i) < 0, \quad i = 1, \ldots, M$$

- choose a linearly parametrized family of functions

$$f(z) = \theta^T F(z)$$

  $F = (F_1, \ldots, F_k) : \mathbf{R}^n \to \mathbf{R}^k$ are basis functions

- solve a set of linear inequalities in θ:

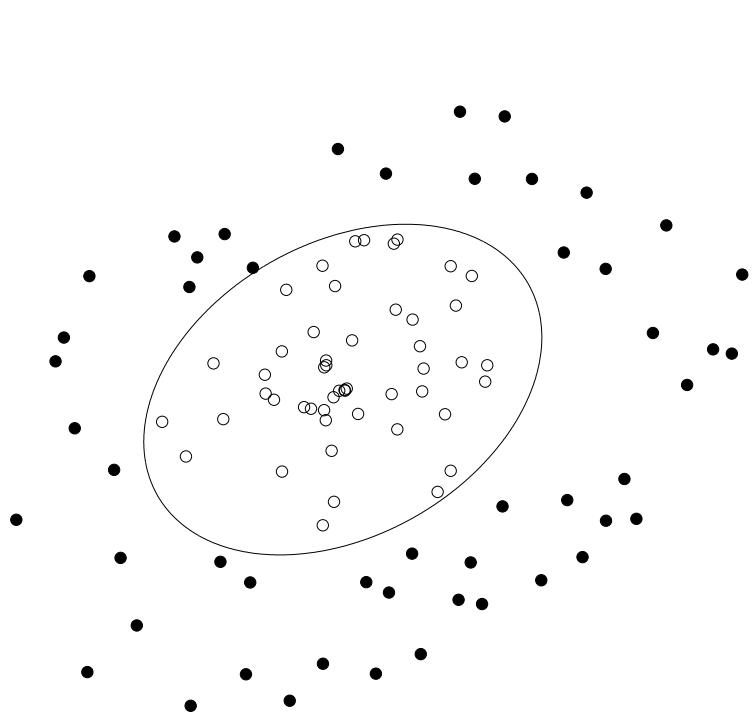
$$\theta^T F(x_i) \geq 1, \quad i = 1, \ldots, N, \qquad \theta^T F(y_i) \leq -1, \quad i = 1, \ldots, M$$

quadratic discrimination: $f(z) = z^T P z + q^T z + r$

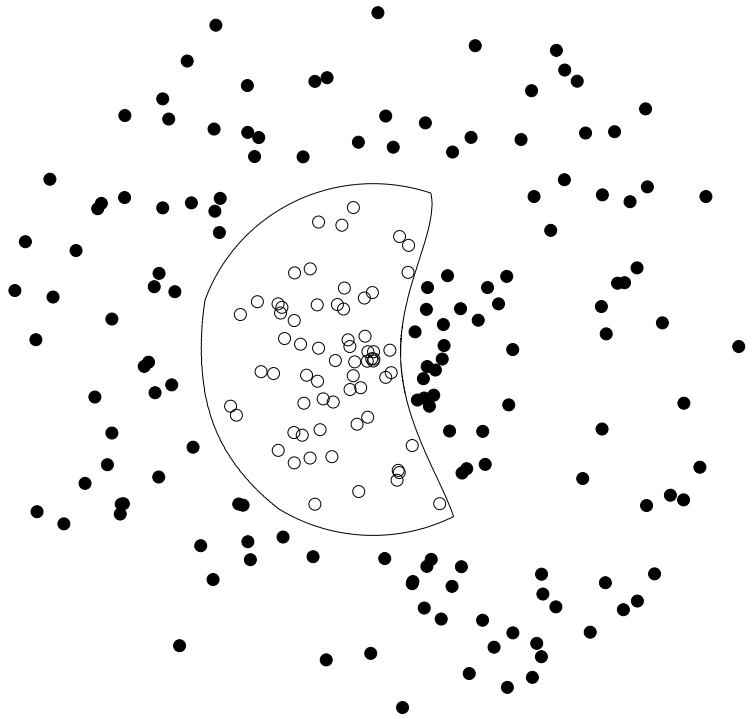
$$x_i^T P x_i + q^T x_i + r \geq 1, \qquad y_i^T P y_i + q^T y_i + r \leq -1$$

can add additional constraints ($e.g.$, $P \preceq -I$ to separate by an ellipsoid)

polynomial discrimination: $F(z)$ are all monomials up to a given degree



separation by ellipsoid



separation by 4th degree polynomial

Support Vector Machines: Duality

Given m data points $x_i \in \mathbf{R}^n$ with labels $y_i \in \{-1, 1\}$.

- The maximum margin classification problem can be written

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|w\|_2^2 + C\mathbf{1}^T z \\ \text{subject to} & y_i(w^T x_i) \geq 1 - z_i, \quad i = 1, \ldots, m \\ & z \geq 0 \end{array}$$

  in the variables w, $z \in \mathbf{R}^n$, with parameter $C > 0$.

- We can set $w = (w, \mathbf{1})$ and increase the problem dimension by 1. So we can assume w.l.o.g. $b = 0$ in the classifier $w^T x_i + b$.
- The Lagrangian is written

$$L(w, z, \alpha) = \frac{1}{2}\|w\|_2^2 + C\mathbf{1}^T z + \sum_{i=1}^{m} \alpha_i (1 - z_i - y_i w^T x_i)$$

  with dual variable $\alpha \in \mathbf{R}_+^m$.

Support Vector Machines: Duality

- The Lagrangian can be rewritten

$$L(w,z,\alpha) = \frac{1}{2}\left(\left\|w - \sum_{i=1}^m \alpha_i y_i x_i\right\|_2^2 - \left\|\sum_{i=1}^m \alpha_i y_i x_i\right\|_2^2\right) + (C\mathbf{1}-\alpha)^T z + \mathbf{1}^T\alpha$$

  with dual variable $\alpha \in \mathbf{R}^n_+$.

- Minimizing in (w,z) we form the dual problem

$$\begin{array}{ll} \text{maximize} & -\frac{1}{2}\left\|\sum_{i=1}^m \alpha_i y_i x_i\right\|_2^2 + \mathbf{1}^T\alpha \\ \text{subject to} & 0 \leq \alpha \leq C \end{array}$$

- At the optimum, we must have

$$w = \sum_{i=1}^m \alpha_i y_i x_i \quad \text{and} \quad \alpha_i = C \text{ if } z_i > 0$$

  (this is the representer theorem).

Support Vector Machines: the kernel trick

- If we write X the data matrix with columns x_i, the dual can be rewritten

$$\begin{array}{ll} \text{maximize} & -\frac{1}{2}\alpha^T \mathbf{diag}(y) X^T X \mathbf{diag}(y)\alpha + \mathbf{1}^T\alpha \\ \text{subject to} & 0 \leq \alpha \leq C \end{array}$$

- This means that the data only appears in the dual through the gram matrix

$$K = X^T X$$

  which is called the **kernel** matrix.

- In particular, the original dimension n **does not appear in the dual.** SVM complexity only grows with the number of samples.
- In particular, the x_i are allowed to be infinite dimensional.
- The only requirement on K is that $K \succeq 0$.

Suppose we want to classify a problem with x of size 30,000 and we have 170 sample points: we only solve a problem of size 170. . .

Kernels

The matrix of scalar products $K_{ij} = (x_i^T x_j)$ has the following properties:

- If two points $x_i,\ x_j$ are **similar**, then their scalar product $x_i^T x_j$ is large.
- As a scalar product matrix, the matrix K is **positive semidefinite** (i.e. $v^T K v \geq 0$, for all vectors v).

We can generalize this. What defines a **kernel matrix**?

- It should be **positive semidefinite**.
- It should also represent **similarity**: high coefficients K_{ij} for similar data points.
- It should be easy to compute.

Kernels

There a lot more kernels. . .

- Very common choice, the **Gaussian** kernel:

$$K(x, y) = \exp\left(-\frac{\|x - y\|_2^2}{2\sigma^2}\right)$$

- Another popular choice for text classification:

$$K(x, y) = \exp\left(-\frac{\|x - y\|_1}{\gamma}\right)$$

- Optimize σ and C simultaneously. . .

The best choice of kernel varies with the application.

Text Classification

How can we do this on **text**? Suppose our data is composed of news articles:

- Start from a (large) dictionary.
- For each article: form a vector counting the frequency of dictionary words inside the article.

Example from Reuters.

Ryanair Q3 ***profit up*** *30%,* ***stronger*** *than expected.*

DUBLIN, Feb 5 (Reuters) - Ryanair (RYA.I: Quote, Profile , Research) posted a 30 pct ***jump*** *in third-quarter net* ***profit*** *on Monday, confounding analyst* ***expectations*** *for a* ***fall****, and* ***ramped up*** *its full-year* ***profit*** *goal while predicting big fuel-cost* ***savings*** *for the following year (. . .).*

This becomes:

profit	loss	up	down	jump	fall	below	expectations	ramped up
3	0	2	0	1	1	0	1	1

Kernels

What if we want to use a mix of data types:

time series + text

e.g. a big spike 5 minutes before an article would mean that the news is already out.

- We can pick the best possible kernel: e.g. $\mathbf{K_1}$ for the time series and $\mathbf{K_2}$ for text.
- Any convex combination of kernels is again a kernel

$$\mathbf{K} = \lambda \mathbf{K_1} + (1 - \lambda)\mathbf{K_2}$$

- Optimize in λ to find the optimal mix.

Cross-validation

How do we get C (and the kernel parameter σ)?

- First, split the sample data between **training** and **test** set.
- Perform **cross-validation** (CV) on the **training set**:
  - Randomly split the training set into several equally large subsets.
  - Train the classifier on all but one of these subsets.
  - Measure its performance (precision & recall) on the remaining subset.
  - Repeat for each subset of the training set.
- Compute performance of best classifier from CV on the original test set data.

A simple example

Simple illustrative example.

- Picked 16 active stocks from 3 sectors: Healthcare, Energy and Technology.
- Every day, collected the top **Reuters** articles containing the stock ticker. (using Google news.)
- Created a dictionary of 168 common financial terms.
- All the articles are reduced to vectors (with the frequency of each word).
- One month of data: Dec. 12, 2005 until Jan. 21, 2006.
- Collected 1134 articles.

Stocks tracked

Company	Ticker	Sector	Industry
Amgen, Inc.	AMGN	Healthcare	Biotechnology & Drugs
Apple Computers, Inc.	AAPL	Technology	Computer Hardware
AstraZeneca plc	AZN	Healthcare	Major Drugs
BP plc	BP	Energy	Oil & Gas - Integrated
ConocoPhillips	COP	Energy	Oil & Gas - Integrated
Exxon Mobil Corp.	XOM	Energy	Oil & Gas - Integrated
Genentech, Inc.	DNA	Healthcare	Biotechnology & Drugs
Google Inc.	GOOG	Technology	Computer Services
Guidant Corp.	GDT	Healthcare	Medical Equipment and Supplies
Pfizer	PFE	Healthcare	Major Drugs
Royal Dutch Shell plc.	RDS	Energy	Oil & Gas - Integrated
Samsung Electronics Co. Ltd.	N/A	Consumer Cyclical	Audio and Visual Equipment
Sanofi-Aventis	SNY	Healthcare	Biotechnology & Drugs
Sony Corp.	SNE	Consumer Cyclical	Audio and Visual Equipment
Teva Pharmaceutical Industries Ltd.	TEVA	Healthcare	Biotechnology & Drugs
Wyeth	WYE	Healthcare	Major Drugs

Sample dictionary words

Theme	Dictionary Words
Announcement of new product line, innovation	announce, introduce, product, innovate, develop, technology, begin, prospects, excite, design, hot, new
Companys earnings, sales information	earning, sales, met expectations, below expectations, above expectations, growth, data, gross margin, quarterly, profit, forecast
Competition	competition, competing, threat, shift, replace
Deal, merger, acquisition	talks, sell, sold, deal, buy, acquire, bid, bought, agree, acquisition, hopes
Negative sentiment	murky, worry, end, loss, dampen, down, drop, fall, severe, thin, weak, offset, slow, hurt, negative, suffer, sink, below, fell short, lackluster, flaw, failure
Positive sentiment	grow, increase, rise, surge, bullish, strength, fast, positive, lead, prominent, success, gain, high, jump

A simple example

Record answers for the following **questions**:

- Does the article relate good news or bad news for the company? (Good, Bad)
- Does the article relate new news or updated news regarding the company? (New, Updated)
- Is the company in the Healthcare sector? (Yes, No)
- Is the company in the Energy sector? (Yes, No)
- Does the article refer to a corporate acquisition, merger, deal regarding the company or within the industry of the company? (Yes, No)
- Is the article firm-specific? (Yes, No)
- Is the article industry-specific? (Yes, No)
- Does the article refer to the release of earnings/sales/profits figures of the company? (Yes, No)

For each article, manually classify it to create a **training set**. . .

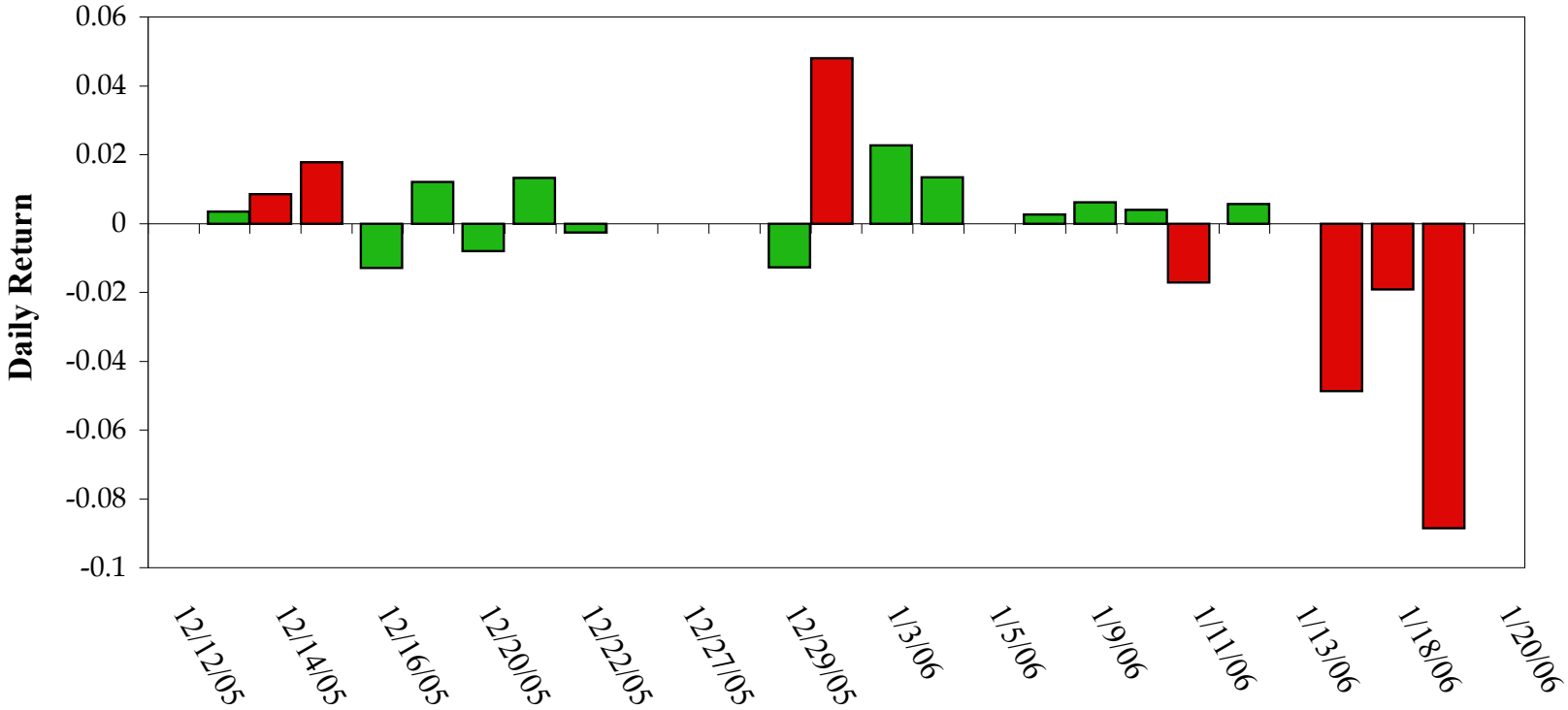
A simple example

Results (Recall), leave-one-out cross-validation:

$$Recall = \frac{TP}{FN + TP}$$

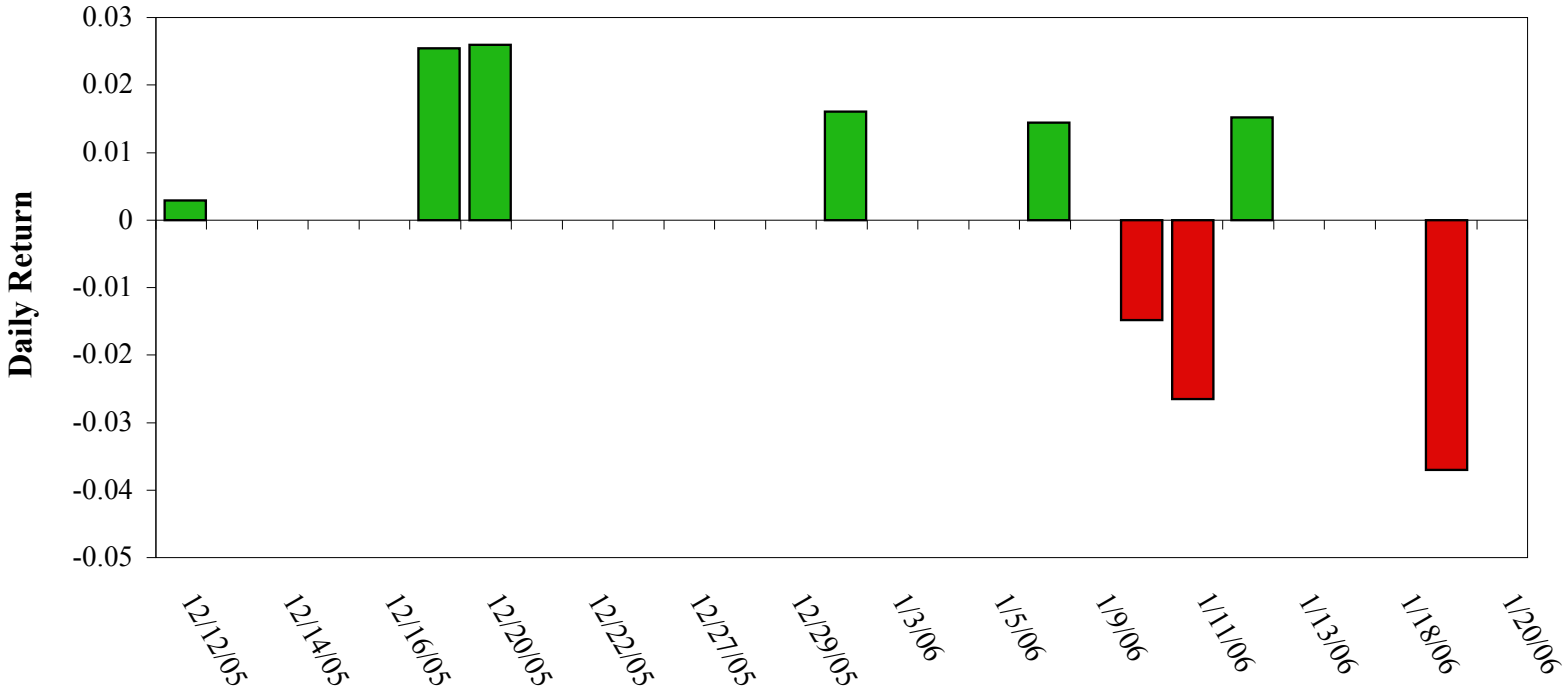
Question	**Recall**
Good news or bad news?	65%
New news?	65%
Healthcare sector?	94%
Energy sector?	100%
Corporate acquisition, merger, deal?	89%
Firm-specific?	66%
Industry-specific?	89%
Release of earnings/sales/profits figures of the company?	91%

A simple example



Stock returns (bars) and news (**good** in green, **bad** in red) for Google.

A simple example



Stock returns (bars) and news (**good** in green, **bad** in red) for Genentech.

Another simple example

Another classic example on a larger data set:

- Question: Is this article about **corporate acquisitions**?
- Original data: 21578 Reuters articles.
- Train the classifier on a set of 2000 articles: 1000 positive, 1000 negative.
- Test the classifier on another set of 600 articles.

Results:

- Accuracy on test set: 97.33%

$$Accuracy = \frac{TP}{FP + TP}$$

- Recall on test set: 95.81%

$$Recall = \frac{TP}{FN + TP}$$

Maximum Likelihood

Parametric distribution estimation

- distribution estimation problem: estimate probability density $p(y)$ of a random variable from observed values
- parametric distribution estimation: choose from a family of densities $p_x(y)$, indexed by a parameter x

maximum likelihood estimation

$$\text{maximize (over } x) \quad \log p_x(y)$$

- y is observed value
- $l(x) = \log p_x(y)$ is called log-likelihood function
- can add constraints $x \in C$ explicitly, or define $p_x(y) = 0$ for $x \notin C$
- a convex optimization problem if $\log p_x(y)$ is concave in x for fixed y

Linear measurements with IID noise

linear measurement model

$$y_i = a_i^T x + v_i, \quad i = 1, \ldots, m$$

- $x \in \mathbf{R}^n$ is vector of unknown parameters
- v_i is IID measurement noise, with density $p(z)$
- y_i is measurement: $y \in \mathbf{R}^m$ has density $p_x(y) = \prod_{i=1}^m p(y_i - a_i^T x)$

maximum likelihood estimate: any solution x of

$$\text{maximize} \quad l(x) = \sum_{i=1}^m \log p(y_i - a_i^T x)$$

(y is observed value)

examples

- Gaussian noise $\mathcal{N}(0, \sigma^2)$: $p(z) = (2\pi\sigma^2)^{-1/2} e^{-z^2/(2\sigma^2)}$,

$$l(x) = -\frac{m}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{m}(a_i^T x - y_i)^2$$

  ML estimate is LS solution

- Laplacian noise: $p(z) = (1/(2a))e^{-|z|/a}$,

$$l(x) = -m\log(2a) - \frac{1}{a}\sum_{i=1}^{m}|a_i^T x - y_i|$$

  ML estimate is ℓ_1-norm solution

- uniform noise on $[-a, a]$:

$$l(x) = \begin{cases} -m\log(2a) & |a_i^T x - y_i| \leq a, \quad i = 1, \ldots, m \\ -\infty & \text{otherwise} \end{cases}$$

  ML estimate is any x with $|a_i^T x - y_i| \leq a$

Logistic regression

random variable $y \in \{0, 1\}$ with distribution

$$p = \mathbf{Prob}(y = 1) = \frac{\exp(a^T u + b)}{1 + \exp(a^T u + b)}$$

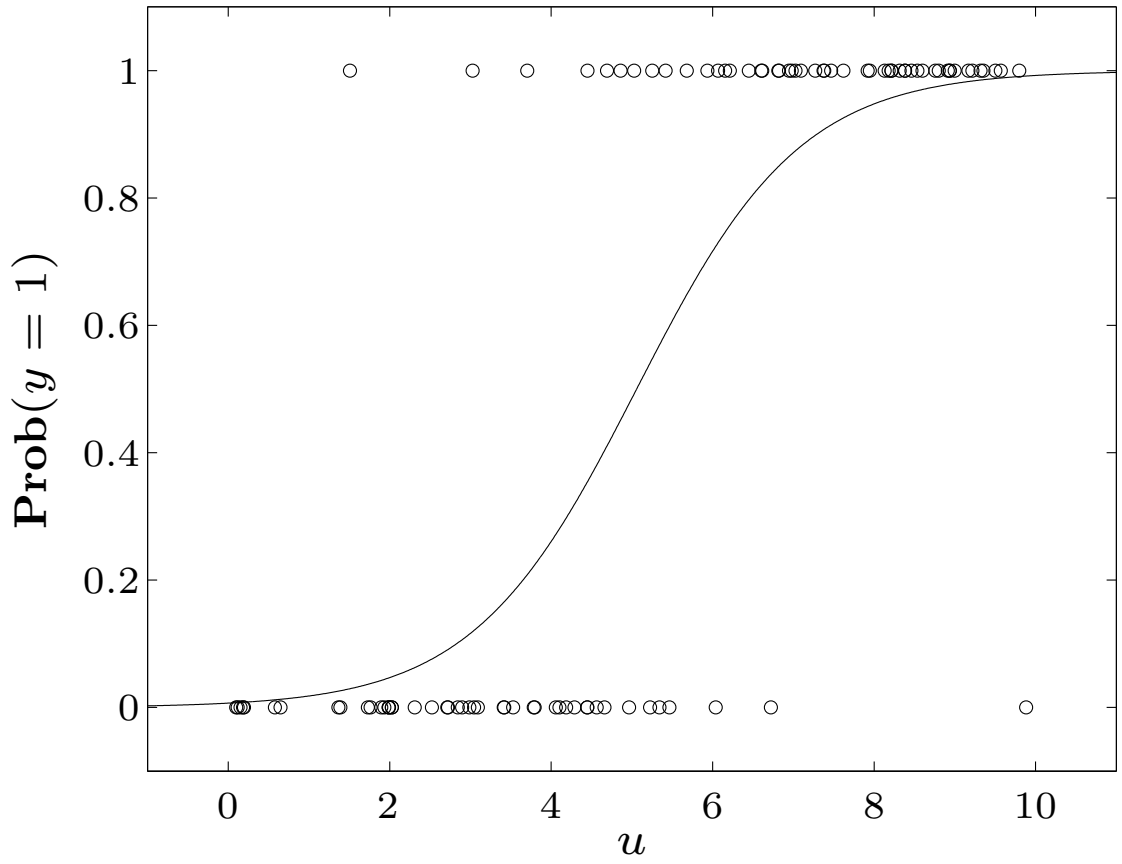
- a, b are parameters; $u \in \mathbf{R}^n$ are (observable) explanatory variables
- estimation problem: estimate a, b from m observations (u_i, y_i)

log-likelihood function (for $y_1 = \cdots = y_k = 1$, $y_{k+1} = \cdots = y_m = 0$):

$$\begin{aligned} l(a, b) &= \log \left(\prod_{i=1}^{k} \frac{\exp(a^T u_i + b)}{1 + \exp(a^T u_i + b)} \prod_{i=k+1}^{m} \frac{1}{1 + \exp(a^T u_i + b)} \right) \\ &= \sum_{i=1}^{k} (a^T u_i + b) - \sum_{i=1}^{m} \log(1 + \exp(a^T u_i + b)) \end{aligned}$$

concave in a, b

example ($n = 1$, $m = 50$ measurements)



- circles show 50 points (u_i, y_i)
- solid curve is ML estimate of $p = \exp(au + b)/(1 + \exp(au + b))$

Hypothesis Testing

(Binary) hypothesis testing

detection (hypothesis testing) problem

given observation of a random variable $X \in \{1, \ldots, n\}$, choose between:

- hypothesis 1: X was generated by distribution $p = (p_1, \ldots, p_n)$
- hypothesis 2: X was generated by distribution $q = (q_1, \ldots, q_n)$

randomized detector

- a nonnegative matrix $T \in \mathbf{R}^{2 \times n}$, with $\mathbf{1}^T T = \mathbf{1}$
- if we observe $X = k$, we choose hypothesis 1 with probability t_{1k}, hypothesis 2 with probability t_{2k}
- if all elements of T are 0 or 1, it is called a deterministic detector

detection probability matrix:

$$D = \begin{bmatrix} Tp & Tq \end{bmatrix} = \begin{bmatrix} 1 - P_{\mathrm{fp}} & P_{\mathrm{fn}} \\ P_{\mathrm{fp}} & 1 - P_{\mathrm{fn}} \end{bmatrix}$$

- P_{fp} is probability of selecting hypothesis 2 if X is generated by distribution 1 (false positive)
- P_{fn} is probability of selecting hypothesis 1 if X is generated by distribution 2 (false negative)

multicriterion formulation of detector design

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{R}^2_+) & (P_{\mathrm{fp}}, P_{\mathrm{fn}}) = ((Tp)_2, (Tq)_1) \\ \text{subject to} & t_{1k} + t_{2k} = 1, \quad k = 1, \ldots, n \\ & t_{ik} \geq 0, \quad i = 1, 2, \quad k = 1, \ldots, n \end{array}$$

variable $T \in \mathbf{R}^{2 \times n}$

scalarization (with weight $\lambda > 0$)

$$\begin{array}{ll} \text{minimize} & (Tp)_2 + \lambda(Tq)_1 \\ \text{subject to} & t_{1k} + t_{2k} = 1, \quad t_{ik} \geq 0, \quad i = 1, 2, \quad k = 1, \ldots, n \end{array}$$

an LP with a simple analytical solution

$$(t_{1k}, t_{2k}) = \begin{cases} (1, 0) & p_k \geq \lambda q_k \\ (0, 1) & p_k < \lambda q_k \end{cases}$$

- a deterministic detector, given by a likelihood ratio test
- if $p_k = \lambda q_k$ for some k, any value $0 \leq t_{1k} \leq 1$, $t_{1k} = 1 - t_{2k}$ is optimal (*i.e.*, Pareto-optimal detectors include non-deterministic detectors)

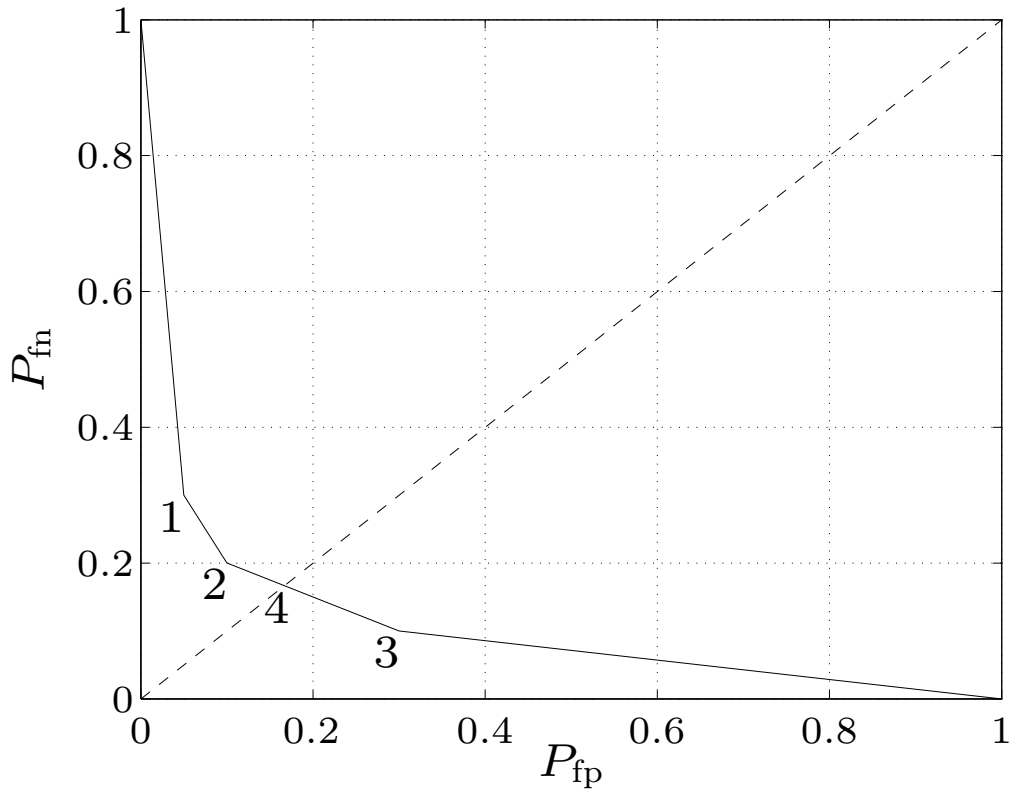
minimax detector

$$\begin{array}{ll} \text{minimize} & \max\{P_{\text{fp}}, P_{\text{fn}}\} = \max\{(Tp)_2, (Tq)_1\} \\ \text{subject to} & t_{1k} + t_{2k} = 1, \quad t_{ik} \geq 0, \quad i = 1, 2, \quad k = 1, \ldots, n \end{array}$$

an LP; solution is usually not deterministic

example

$$P = \begin{bmatrix} 0.70 & 0.10 \\ 0.20 & 0.10 \\ 0.05 & 0.70 \\ 0.05 & 0.10 \end{bmatrix}$$



solutions 1, 2, 3 (and endpoints) are deterministic; 4 is minimax detector

Experiment Design

Experiment design

m linear measurements $y_i = a_i^T x + w_i$, $i = 1, \ldots, m$ of unknown $x \in \mathbf{R}^n$

- measurement errors w_i are IID $\mathcal{N}(0, 1)$
- ML (least-squares) estimate is

$$\hat{x} = \left(\sum_{i=1}^m a_i a_i^T \right)^{-1} \sum_{i=1}^m y_i a_i$$

- error $e = \hat{x} - x$ has zero mean and covariance

$$E = \mathbf{E} e e^T = \left(\sum_{i=1}^m a_i a_i^T \right)^{-1}$$

  confidence ellipsoids are given by $\{x \mid (x - \hat{x})^T E^{-1} (x - \hat{x}) \leq \beta\}$

experiment design: choose $a_i \in \{v_1, \ldots, v_p\}$ (a set of possible test vectors) to make E 'small'

vector optimization formulation

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{S}_+^n) & E = \left(\sum_{k=1}^p m_k v_k v_k^T\right)^{-1} \\ \text{subject to} & m_k \geq 0, \quad m_1 + \cdots + m_p = m \\ & m_k \in \mathbf{Z} \end{array}$$

- variables are m_k (# vectors a_i equal to v_k)
- difficult in general, due to integer constraint

relaxed experiment design

assume $m \gg p$, use $\lambda_k = m_k/m$ as (continuous) real variable

$$\begin{array}{ll} \text{minimize (w.r.t. } \mathbf{S}_+^n) & E = (1/m)\left(\sum_{k=1}^p \lambda_k v_k v_k^T\right)^{-1} \\ \text{subject to} & \lambda \succeq 0, \quad \mathbf{1}^T\lambda = 1 \end{array}$$

- common scalarizations: minimize $\log\det E$, $\mathbf{Tr}\, E$, $\lambda_{\max}(E)$, . . .
- can add other convex constraints, $e.g.$, bound experiment cost $c^T\lambda \leq B$

D-optimal design

$$\begin{array}{ll} \text{minimize} & \log\det\left(\sum_{k=1}^p \lambda_k v_k v_k^T\right)^{-1} \\ \text{subject to} & \lambda \succeq 0, \quad \mathbf{1}^T\lambda = 1 \end{array}$$

interpretation: minimizes volume of confidence ellipsoids

dual problem

$$\begin{array}{ll} \text{maximize} & \log\det W + n\log n \\ \text{subject to} & v_k^T W v_k \leq 1, \quad k = 1, \ldots, p \end{array}$$

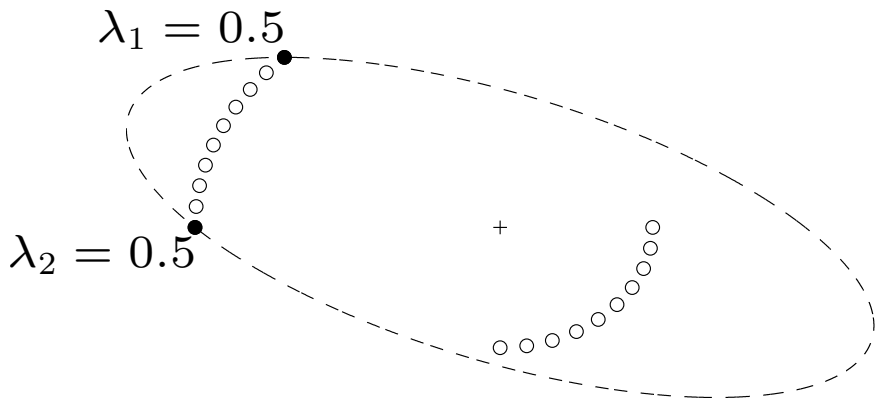
interpretation: $\{x \mid x^T W x \leq 1\}$ is minimum volume ellipsoid centered at origin, that includes all test vectors v_k

complementary slackness: for λ, W primal and dual optimal

$$\lambda_k(1 - v_k^T W v_k) = 0, \quad k = 1, \ldots, p$$

optimal experiment uses vectors v_k on boundary of ellipsoid defined by W

example $(p = 20)$



design uses two vectors, on boundary of ellipse defined by optimal W

derivation of dual

first reformulate primal problem with new variable X:

$$\begin{array}{ll} \text{minimize} & \log\det X^{-1} \\ \text{subject to} & X = \sum_{k=1}^p \lambda_k v_k v_k^T, \quad \lambda \succeq 0, \quad \mathbf{1}^T\lambda = 1 \end{array}$$

$$L(X,\lambda,Z,z,\nu) = \log\det X^{-1} + \mathbf{Tr}\left(Z\left(X - \sum_{k=1}^p \lambda_k v_k v_k^T\right)\right) - z^T\lambda + \nu(\mathbf{1}^T\lambda - 1)$$

- minimize over X by setting gradient to zero: $-X^{-1} + Z = 0$
- minimum over λ_k is $-\infty$ unless $-v_k^T Z v_k - z_k + \nu = 0$

dual problem

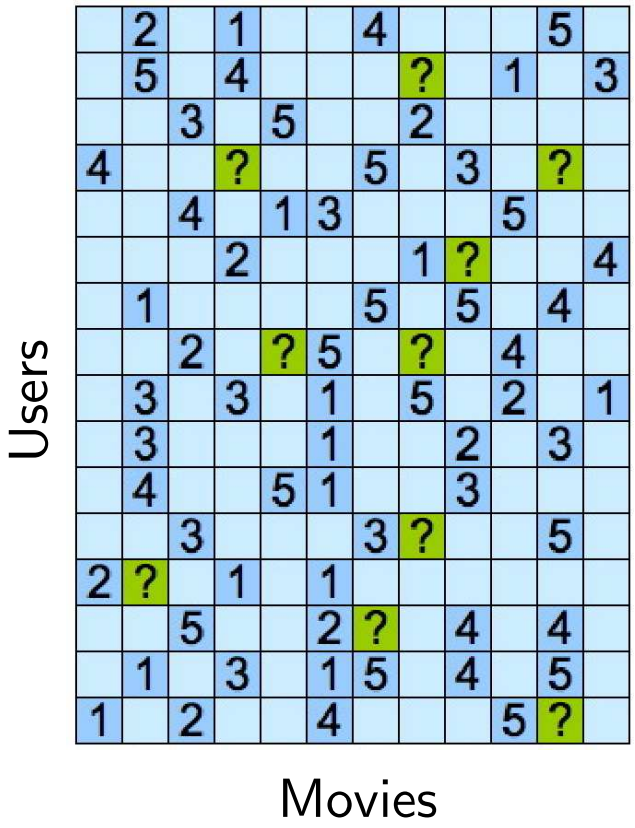
$$\begin{array}{ll} \text{maximize} & n + \log\det Z - \nu \\ \text{subject to} & v_k^T Z v_k \leq \nu, \quad k = 1,\ldots,p \end{array}$$

change variable $W = Z/\nu$, and optimize over ν to get dual of page 40

Collaborative prediction

Collaborative prediction

- Users assign **ratings** to a certain number of movies:



- Objective: make recommendations for other movies. . .

Collaborative prediction

- Infer **user preferences** and **movie features** from user ratings.
- We use a linear prediction model:

$$rating_{ij} = u_i^T v_j$$

  where u_i represents user characteristics and v_j movie features.
- This makes collaborative prediction a **matrix factorization** problem
- Overcomplete representation. . .

Collaborative prediction

- **Inputs**: a matrix of ratings $M_{ij} = \{-1, +1\}$ for $(i, j) \in S$, where S is a subset of all possible user/movies combinations.
- We look for a linear model by factorizing $M \in \mathbf{R}^{n \times m}$ as:

$$M = U^T V$$

  where $U \in \mathbf{R}^{n \times k}$ represents user characteristics and $V \in \mathbf{R}^{k \times m}$ movie features.
- **Parsimony**. . . We want k to be as small as possible.
- **Output**: a matrix $X \in \mathbf{R}^{n \times m}$ which is a low-rank approximation of the ratings matrix M.

Least-Squares

- Choose Means Squared Error as measure of discrepancy.
- Suppose S is the full set, our problem becomes:

$$\min_{\{X:\, \mathbf{Rank}(X)=k\}} \|X - M\|^2$$

- This is just a **singular value decomposition** (SVD). . .

Problem: Not true when S is not the full set (partial observations). Also, MSE not a good measure of prediction performance. . .

Relaxation

Partial observations for pairs $(i,j) \in S$.

$$\text{minimize} \quad \mathbf{Rank}(X) + c \sum_{(i,j)\in S} \max(0, 1 - X_{ij}M_{ij})$$

non-convex and numerically hard. . .

- Relaxation result in Fazel et al. [2001]: replace $\mathbf{Rank}(X)$ by its convex envelope on the spectahedron to solve:

$$\text{minimize} \quad \|X\|_* + c \sum_{(i,j)\in S} \max(0, 1 - X_{ij}M_{ij})$$

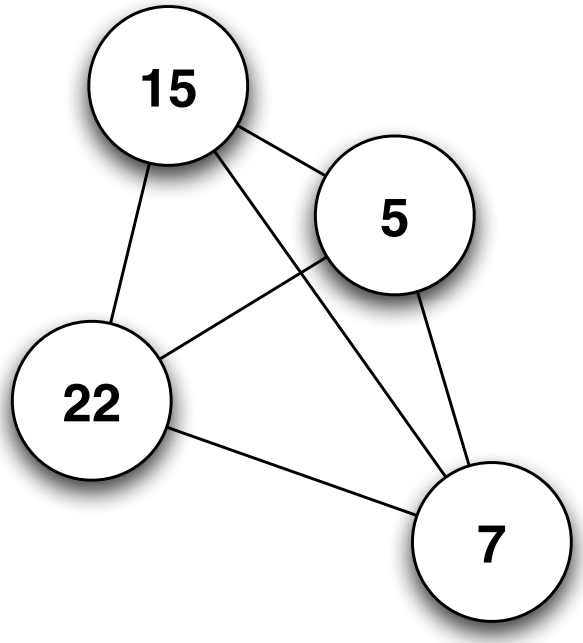
  where $\|X\|_*$ is the **nuclear norm**, $i.e.$ sum of the singular values of X.
- Srebro [2004]: This relaxation also corresponds to multiple large margin SVM classifications.

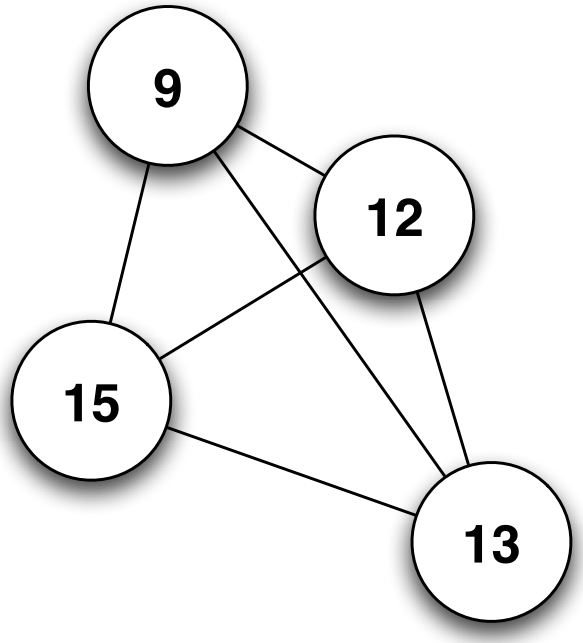
Transportation Problems

Transportation problem

- A company stocks a certain product at various locations across the country.
- When demand shifts geographically, this stock needs to be readjusted.
- **Input**: An initial stock configuration, the transportation cost between two locations.
- **Objective**: Find the optimal transportation plan for moving the stock from one configuration to another.
- **Output**: A transportation plan.

Transportation problem





The total stock is the same in both configurations: 49. What's the best transportation plan?

Transportation problem

We can formulate this as a **linear program**.

- Let a_i be the stock at factory i in configuration one, with $i = 1, \ldots, n$.
- Let b_i be the stock at factory i in configuration two, with $i = 1, \ldots, n$.
- Let C_{ij} the cost of moving one unit of stock from factory i to factory j.
- Our **variable** is $X \in \mathbf{R}^{n \times n}$, where X_{ij} represents the stock moving from factory i to factory j.

The problem can be written:

$$\begin{array}{ll} \text{minimize} & \sum_{i,j=1}^{n} C_{ij} X_{ij} \\ \text{subject to} & \sum_{i=1}^{n} X_{ij} = b_j, \quad j = 1, \ldots, n \\ & \sum_{j=1}^{n} X_{ij} = a_i, \quad i = 1, \ldots, n \\ & X_{ij} \geq 0 \end{array}$$

Transportation problem

- The total cost of the transportation plan X is given by

$$\sum_{i,j=1}^{n} C_{ij} X_{ij}$$

- The constraint $\sum_{i=1}^{n} X_{ij} = b_j$ ensures that the total stock going to factory j is equal to b_j.
- The constraint $\sum_{j=1}^{n} X_{ij} = a_i$ ensures that the total stock extracted from factory i is equal to a_i

This can also be written as:

$$\begin{array}{ll} \text{minimize} & \mathbf{Tr}(C^T X) \\ \text{subject to} & \mathbf{1}^T X = b^T \\ & X\mathbf{1} = a \\ & X \geq 0 \end{array}$$

Transportation problem

- If there are n factories, this is a linear program with $O(n^2)$ variables.
- This means that the complexity of solving the transportation problem in this format grows as $O(n^7)$.

Simple example:

$$C = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix} \quad a = \begin{bmatrix} 1 \\ 2 \\ 4 \end{bmatrix} \quad b = \begin{bmatrix} 2 \\ 3 \\ 2 \end{bmatrix}$$

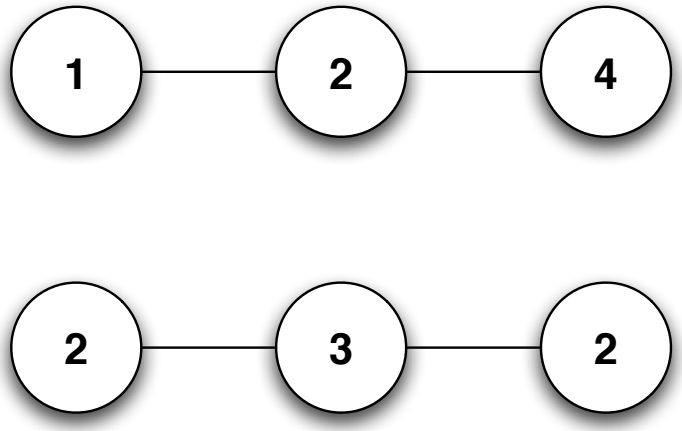
The factories are on a line. We solve:

$$\begin{array}{lll} d(a,b) = & \text{minimize} & \mathbf{Tr}(C^T X) \\ & \text{subject to} & \mathbf{1}^T X = b^T \\ & & X\mathbf{1} = a \\ & & X \geq 0 \end{array}$$

which has 9 variables.

Transportation problem

Graphically, this is:



The optimal cost is 3 and the optimal transport plan is here:

$$X = \begin{bmatrix} 1.0000 & 0.00000 & 0.00000 \\ 0.82877 & 1.1712 & 0.00000 \\ 0.17123 & 1.8288 & 2.0000 \end{bmatrix}$$

Interpretation?

Transportation problem: generalization

For our choice of C, the function $d(a, b)$ satisfies:

- $d(a, b) = d(b, a) \geq 0$ and $d(a, b) = 0$ iff $a = b$.
- $d(a, c) \leq d(a, b) + d(b, c)$.

This means that it defines a **distance** (called the Earth Mover's Distance).

- We can always normalize the vectors a and b so that they sum to 1.
- If a and b are two positive vectors that sum to one, they can be **probability** vectors.

$$\sum_{i=1}^{n} a_i = 1, \ a \geq 0$$

- We can use this to compute distances between probability distributions.
- This has tons of applications in imaging, probability theory, etc.

Image Classification

Let's look at an application to image classification:

- Start from a collection of images
- Try to categorize them according to some key features
- Simple enough to process very large databases
- Should be independent of resolution, size, orientation, etc.

Image Classification

Looking for ocean pictures on Google images. . .

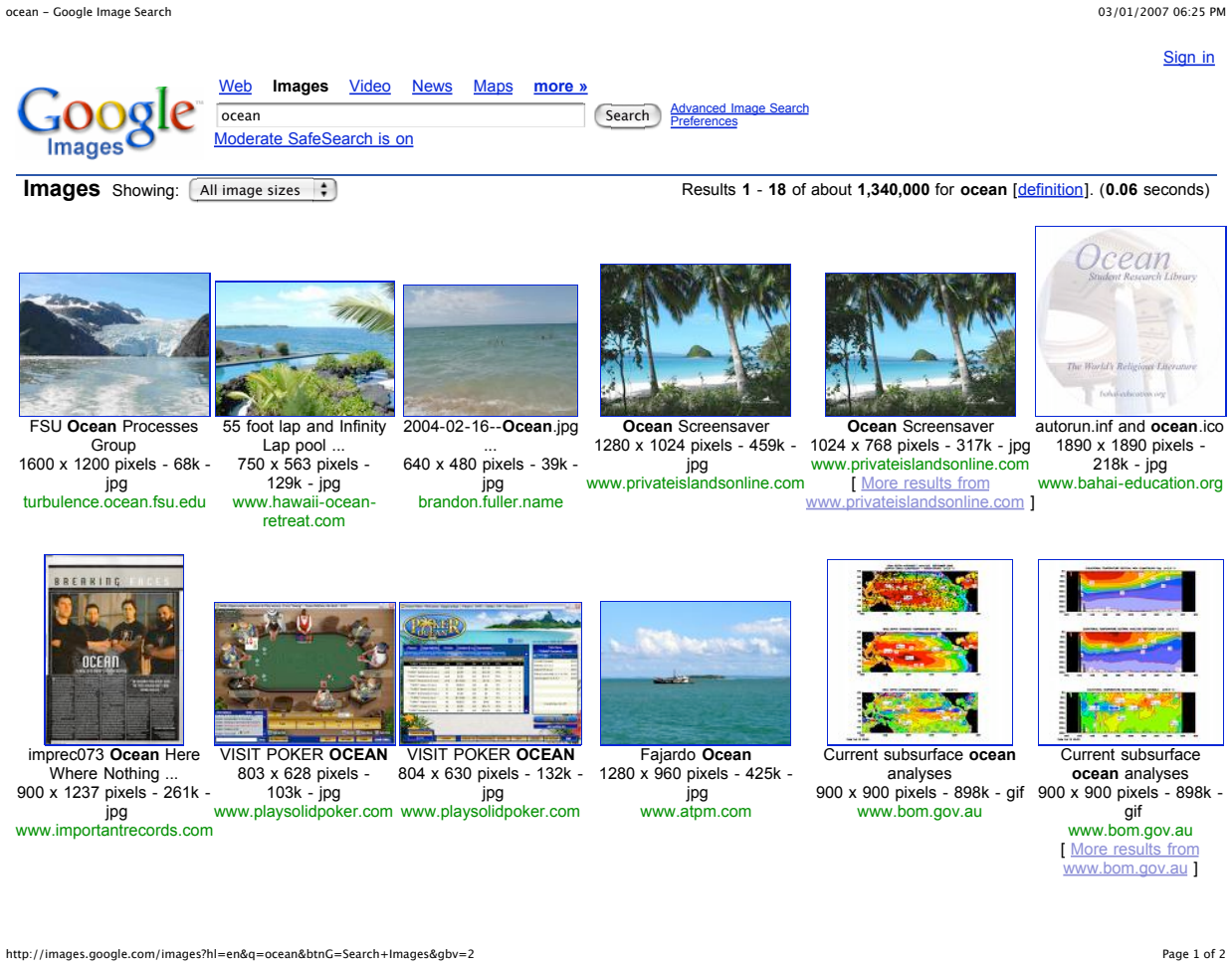


Image Classification

Our fundamental feature here: the **spectrum**.

- Each color corresponds to a particular light wavelength:
  - **Red** is 825 nm.
  - **Green** is 550 nm.
  - **Blue** is 412 nm.
- For each image in the collection, we can extract its spectrum, i.e. the distribution of colors in the image.
- Alternative: use hue from `rgb2hsv` in MATLAB.
- Good invariance properties. The spectrum of an image is not affected by:
  - Scaling
  - Rotation
  - Resolution
  - Brightness

Image Classification

Example of image spectrum.

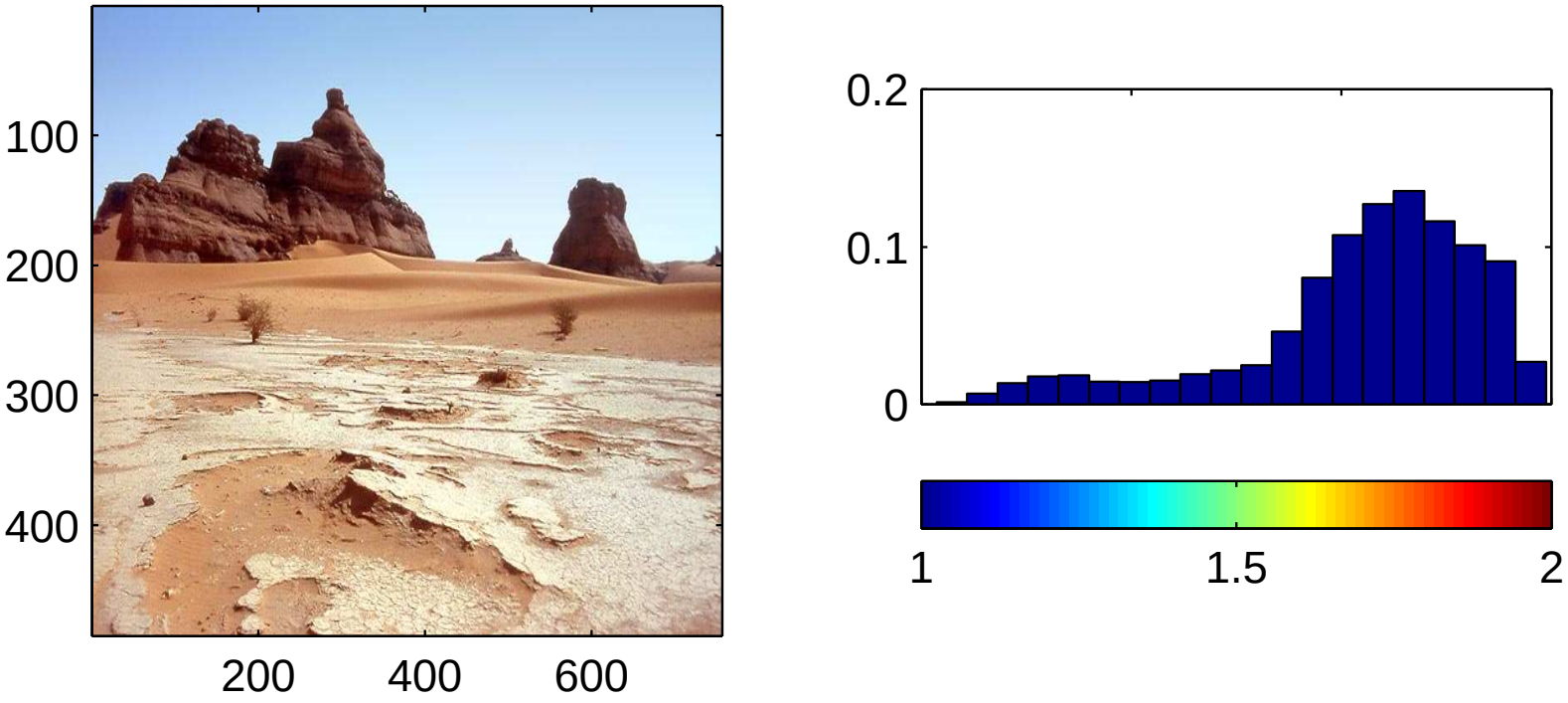
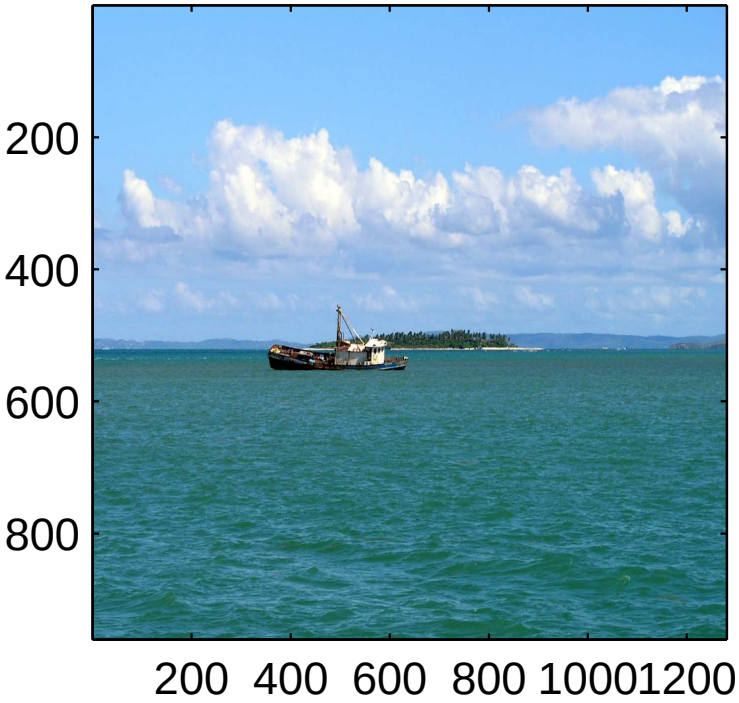


Image Classification

Another example of image spectrum.



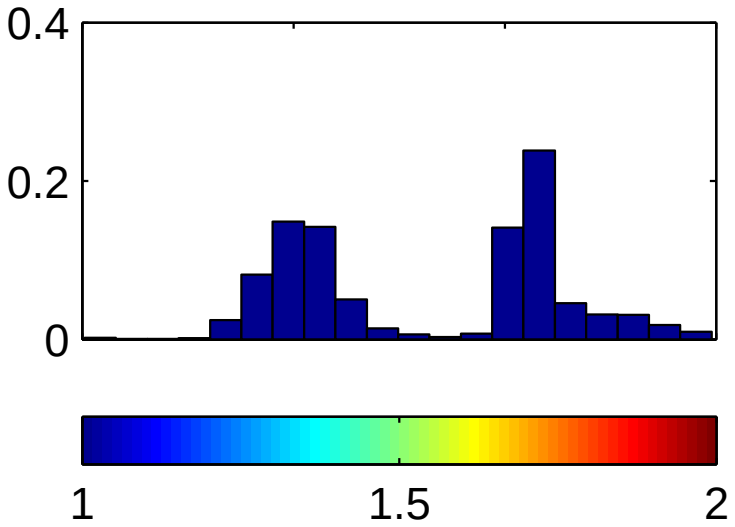


Image Classification

Compare two images:

- Compute the spectrum of each image. Create two positive vectors a and b with sum 1.
- Compute the Earth Mover's Distance between a and b by solving:

$$d(a,b) = \begin{array}{ll} \text{minimize} & \mathbf{Tr}(C^T X) \\ \text{subject to} & \mathbf{1}^T X = a \\ & X\mathbf{1} = b \\ & X \geq 0 \end{array}$$

- Group images where $d(a,b)$ is small (note: not an easy visualization problem).

Image Classification

Example, image classification using EMD (Rubner et al. 1997):

*

References

M. Fazel, H. Hindi, and S. Boyd. A rank minimization heuristic with application to minimum order system approximation. *Proceedings American Control Conference*, 6:4734–4739, 2001.

N. Srebro. *Learning with Matrix Factorization*. PhD thesis, Massachusetts Institute of Technology, 2004.