Machine Learning for Finance

Neal Parikh

Cornell University

Spring 2018

Supervised learning

Supervised learning

- suppose $x \in \mathbf{R}^n$ and $y \in \mathbf{R}$ believed to be related by some unknown function $f : \mathbf{R}^n \to \mathbf{R}$ such that $y \approx f(x)$

- the function f is unknown, but we have sample/training data

$$\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$$

  - x_i: feature vector, inputs, predictors, ...
  - y_i: outcome, response, output, ...
  - (x_i, y_i): training example, observation, sample, measurement, ...

- use $\mathcal{D}$ to construct (learn, fit, estimate, ...) a model $\hat{f} : \mathbf{R}^n \to \mathbf{R}$ so

$$y \approx \hat{y} = \hat{f}(x)$$

Regression

- regression refers to case when $y \in \mathbf{R}$
- variety of approaches, but the most standard are linear:

$$\hat{f}(x) = w^T x$$

where $w \in \mathbf{R}^n$ are weights or parameters

- generally only care about the model being linear in the *parameters*:

$$\hat{f}(x) = w_1 f_1(x) + \cdots w_K f_K(x),$$

where $f_i : \mathbf{R}^n \to \mathbf{R}$ are **feature mappings** or **basis functions**

- goal is to find $\hat{w} \in \mathbf{R}^n$ for which **residuals** (prediction errors) $r_i = \hat{y}_i - y_i$ are reasonably small

Classification

- classification refers to case when $y \in [K] = \{1, \ldots, K\}$, with $K = 2$ called **binary classification**
- in this case, model $\hat{f}$ also called a **classifier**
- consider input space divided into regions based on classification
  - regions are called **decision regions**
  - boundaries of decision regions are called **decision boundaries**
  - decision boundaries can be rough or smooth
  - if decision boundaries are linear, model is a **linear classifier**
- surprising variety of methods yield linear classifiers
- if dataset can be separated exactly by a linear classifier, it is called **linearly separable**

Approaches to classification

- **probabilistic model**: estimate the conditional probability distribution $p(y \mid x)$, then use this distribution to classify new points
  - **generative model**: model the joint distribution $p(x, y)$, usually by modeling $p(x \mid y)$ and $p(y)$, and derive $p(y \mid x)$ via Bayes' rule
  - **discriminative model**: directly model the conditional distribution $p(y = k \mid x)$ only
- **non-probabilistic model**: construct a function to directly assign each x to a class, *e.g.*, by directly placing a decision boundary somewhere in the space according to some criterion

Linear regression

Linear regression

- consider training set

$$\mathcal{D} = \{(x_1, y_1), \ldots, (x_N, y_N)\}, \quad x_i \in \mathbf{R}^n, y_i \in \mathbf{R}$$

- model: assume y is a linear function of x

$$\hat{f}(x) = w^T x = w_0 + w_1 x_1 + \cdots + w_n x_n$$

  or linear combination of basis functions f_i of x

- either include a constant 1 in x or use separate term w_0

- now need to choose w according to some criterion

Least squares

- optimal weights $\hat{w} \in \mathbf{R}^n$ are the solution to

$$\text{minimize} \quad \|Xw - y\|_2^2$$

where $X \in \mathbf{R}^{N \times n}$, $y \in \mathbf{R}^N$; row i of **feature matrix** X given by x_i

- objective is equivalent to the **residual sum of squares**

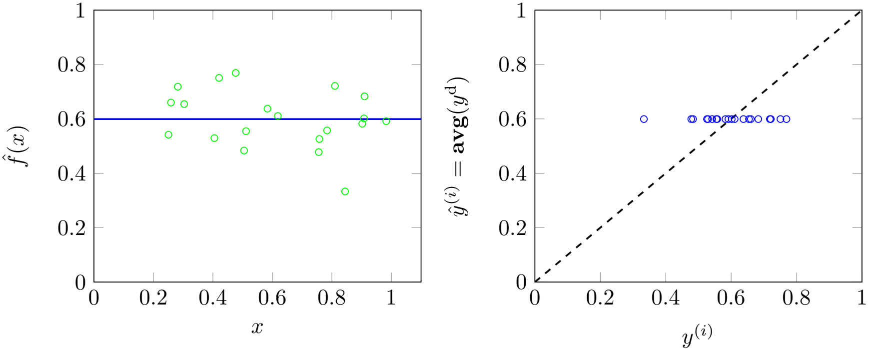
$$\|Xw - y\|_2^2 = \sum_{i=1}^{N} (w^T x_i - y_i)^2,$$

- an unconstrained convex QP with the closed form solution

$$w^\star = (X^T X)^{-1} X^T y$$

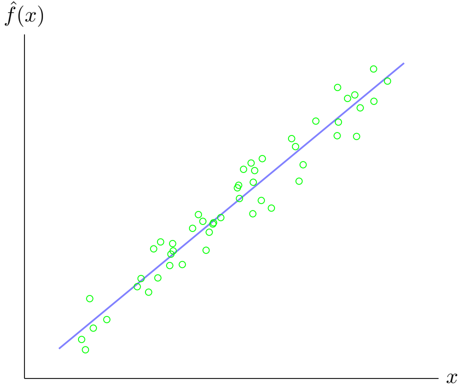
assuming the columns of X are linearly independent

Constant fit



The constant fit $\hat{f}(x) = \mathbf{avg}(y^{\mathrm{d}})$ to $N = 20$ data points and a scatter plot of $\hat{y}^{(i)}$ versus $y^{(i)}$.

Example



Straight-line fit to 50 points $(x^{(i)}, y^{(i)})$ in a plane.

Probabilistic interpretation

- consider the probabilistic model

$$y_i = w^T x_i + \epsilon_i$$

where ϵ_i is an error term capturing unmodeled effects or noise

- assume that the ϵ_i are i.i.d. normal:

$$\epsilon_i \sim \mathrm{N}(0, \sigma^2), \quad p(\epsilon_i) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{\epsilon_i^2}{2\sigma^2}\right)$$

- this implies that $y_i \,|\, x_i \sim \mathrm{N}(w^T x_i, \sigma^2)$ with parameter w, *i.e.*,

$$p(y_i \,|\, x_i; w) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(y_i - w^T x_i)^2}{2\sigma^2}\right)$$

Maximum likelihood estimation

- how to estimate parameters w of a probabilistic model (choose in a parameterized family of probability distributions)?

- several approaches, but the most classical is the method of maximum likelihood

- **likelihood function** is the probability of the data, viewed as a function of the (unknown) weights w

$$L(w) = p(y \mid x_1, \ldots, x_N; w)$$

- **maximum likelihood**: choose w to maximize L

- *i.e.*, choose w that makes the observed data $\mathcal{D}$ the most likely to have been generated under the model assumptions

Maximum likelihood estimation

- since error terms are assumed independent, the likelihood decomposes as

$$L(w) = \prod_{i=1}^{N} p(y_i \,|\, x_i; w)$$

- typically maximize the **log-likelihood** instead

$$\ell(w) = \log L(w) = \sum_{i=1}^{N} \log p(y_i \,|\, x_i; w)$$

- if ℓ is concave, then this yields a convex problem (though not relevant, usually has no closed form solution)

Maximum likelihood estimation for linear regression

- note that

$$\log p(y_i \mid x_i; w) = \log \frac{1}{\sqrt{2\pi\sigma}} - \frac{1}{2\sigma^2}(w^T x_i - y_i)^2$$

so maximizing ℓ reduces to minimizing

$$\sum_{i=1}^{N}(w^T x_i - y_i)^2$$

after removing irrelevant constants; *i.e.*, least squares objective

- under the previous assumptions, the least squares estimator is also the maximum likelihood estimator for w

Capital asset pricing model

- observe market returns $x = (r_1^m, \ldots, r_T^m)$ and individual asset returns $y = (r_1^i, \ldots, r_T^i)$ over some period of length T
- regress individual returns onto market returns

$$\hat{f}(x) = (r^{\mathrm{rf}} + \alpha) + \beta(x - \mu^{\mathrm{mkt}})$$

  - r^{rf} is the risk-free interest rate over the period
  - $\mu^{\mathrm{mkt}} = \mathbf{avg}(x)$ is the average market return

- a linear regression model $\hat{f}(x) = w_1 + w_2 x$ with

$$w_1 = r^{\mathrm{rf}} + \alpha - \beta\mu^{\mathrm{mkt}}, \qquad w_2 = \beta$$

- prediction of asset return has two components:
  - constant $r^{\mathrm{rf}} + \alpha$, where α is average asset return over risk-free rate
  - a proportion β of de-meaned market performance $x - \mu^{\mathrm{mkt}}$

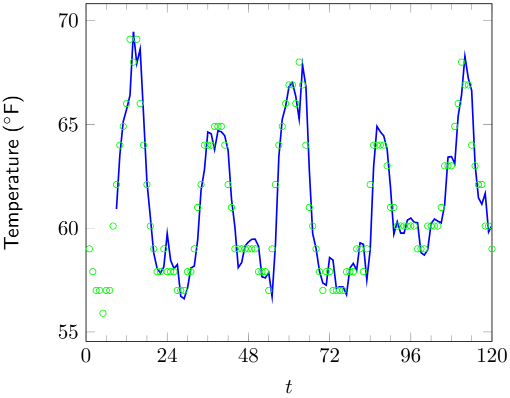
Time series

- suppose data is a series of samples of quantity y at time $x_i = i$
- **trend line** is linear fit to the time series data

$$\hat{y}_i = w_1 + w_2 i$$

- slope w_2 is interpreted as the trend in the quantity over time
- subtrating the trend line from original time series gives de-trended time series
- can extend further to handle seasonal components

Autoregressive time series



Hourly temperature at Los Angeles International Airport between 12:53AM on May 1, 2016, and 11:53PM on May 5, 2016, shown as circles. The solid line is the prediction of an auto-regressive model with eight coefficients. From Boyd & Vandenberghe.

Polynomial regression

- consider basis functions

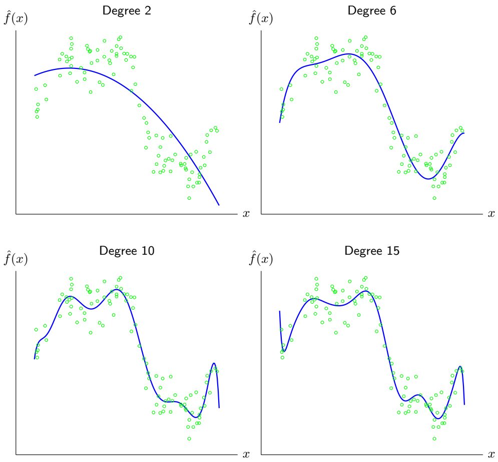
$$f_i(x) = x^{i-1}, \quad i = 1, \ldots, p,$$

  so $\hat{f}$ is a polynomial of degree at most $p-1$:

$$\hat{f}(x) = w_1 + w_2 x + \cdots + w_p x^{p-1}$$

- smallest residuals given by the highest degree polynomial, but generally don't want to choose this (will overfit the data)

Polynomial regression



Least squares polynomial fits of degree 2, 6, 10, and 15 to 100 points. From Boyd & Vandenberghe.

Feature engineering

- an important topic we will not emphasize in this course
- transforming features
  - standardizing / whitening
  - Winsorizing
  - log transform
  - P/E ratio
  - TFIDF
- adding new features
  - one-hot encoding of categorical features
  - product and interaction terms
  - nonlinear transforms
  - stratified models

Logistic regression

Binary classification

- consider training set

$$\mathcal{D} = \{(x_1, y_1), \ldots, (x_N, y_N)\}, \quad x_i \in \mathbf{R}^n, y_i \in \{0, 1\}$$

- idea: instead of assuming $y \approx w^T x$, transform $w^T x$ to lie in the interval $[0, 1]$

$$y \approx s(w^T x), \quad s(z) = \frac{1}{1 + \exp(-z)}$$

  where s is the **logistic function** or **sigmoid function**

- will see that approach of using a nonlinear transformation of a linear function will recur repeatedly

- for now, choice of s is fairly arbitrary, but variety of motivations

Sigmoid and logit functions

- sigmoid/logistic function takes the form

$$s(x) = \frac{1}{1 + \exp(-x)}$$

- its inverse is the **logit function**

$$s^{-1}(p) = \log \frac{p}{1-p}, \qquad p \in (0,1)$$

  also known as the **log odds ratio**

- these functions will appear repeatedly

Probabilistic formulation

- logistic regression model assumes

$$p(y = 1 \mid x; w) = s(w^T x)$$

here, $s(w^T x)$ is interpreted as a probability that $y = 1$

- likelihood function can be written as

$$L(w) = \prod_{i=1}^{N} p(y_i \mid x_i; w) = \prod_{i=1}^{N} s(w^T x_i)^{y_i} (1 - s(w^T x_i))^{1-y_i}$$

so the log-likelihood is

$$\ell(w) = \sum_{i=1}^{N} y_i \log s(w^T x_i) + (1 - y_i) \log(1 - s(w^T x_i))$$

- maximizing ℓ is a convex problem

Log odds formulation

- alternatively, assuming that the log odds is a linear function

$$\log \frac{p(y = 1 \mid x)}{p(y = 0 \mid x)} = w^T x$$

implies that

$$p(y = 1 \mid x) = \frac{1}{1 + \exp(-w^T x)} = s(w^T x)$$

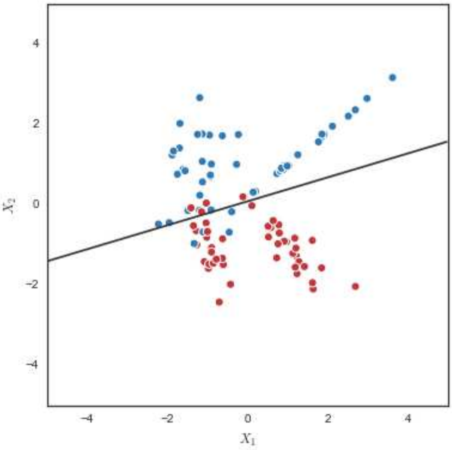
Logistic regression as linear classifier

- if $p(y = 1 \,|\, x) > p(y = 0 \,|\, x)$, classify point as $y = 1$
- *i.e.*, decision boundary is set of points for which log odds are zero

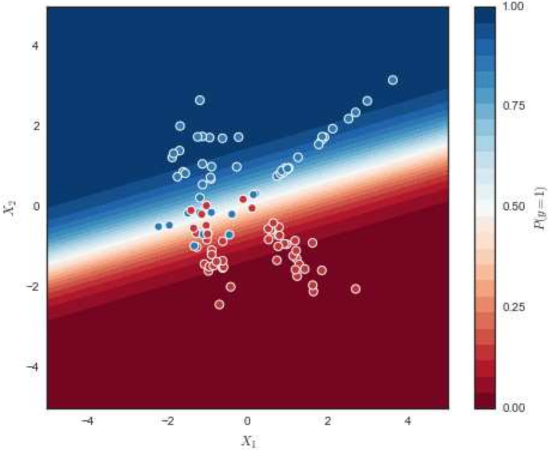
  $$\{x \mid s^{-1}(p(y = 1 \,|\, x)) = 0\} = \{x \mid w^T x = 0\}$$

  a hyperplane giving a linear decision boundary
- if any monotone transformation (here, logit) of $p(y = k \,|\, x)$ is linear, then classifier has linear decision boundaries
- corresponds to probability of either class being $1/2$, but can adjust to other thresholds if there's asymmetric cost in different classification errors
- can also use the output $p(y = 1 \,|\, x)$ directly, if goal is to predict a probability rather than making a decision

Example



Example



Convex approximation to 0-1 loss

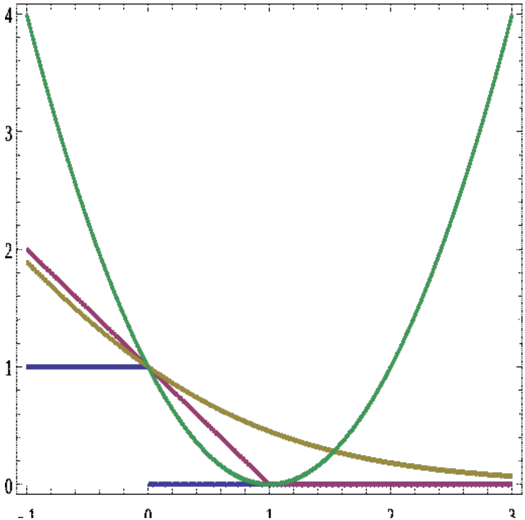
- suppose $y \in \{-1, 1\}$; want to choose f so $\mathbf{sign}\, f(x)$ matches y
- consider choosing f to minimize

$$\frac{1}{N}\sum_{i=1}^{N}[y_i f(x_i) \le 0]$$

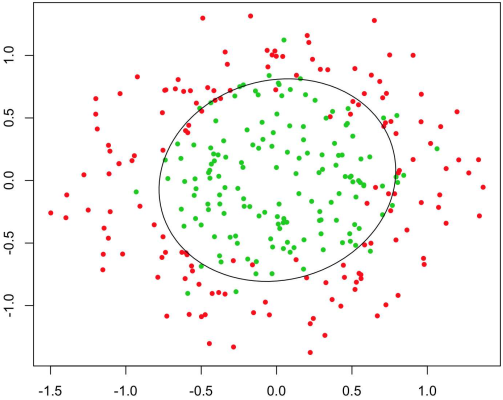
  - $[u \le 0]$ is 0-1 loss
  - $u_i = y_i f(x_i)$ is the **margin**; errors correspond to $u_i < 0$
  - amounts to minimizing (empirical) probability that $y \neq \mathbf{sign}\, f(x)$

- problem: 0-1 loss is nonconvex and so not easy to optimize
- idea: use a convex upper bound as an approximation

Convex approximation to 0-1 loss



Logistic regression with quadratic basis functions



Ad click-through rate prediction

- digital ad revenue: \$200B+/year (Google: $\sim$\$70B, 95%+ of total)
- key task: **click-through rate (CTR) prediction**
- given user search query, initial set of candidate ads is matched based on advertiser-chosen keywords
- use auctions to determine
  - whether these ads are chosen to the user
  - what order they're shown in
  - what prices advertisers pay if their ad is clicked
- inputs for auction mechanism
  - advertiser bids
  - estimate of CTR $p(c = 1 \mid q, a)$ for click $c \in \{0, 1\}$, query q, ad a
- billions of features and examples, predict/update billions times/day

Exponential Families and Generalized Linear Models

Generalizing linear and logistic regression

- so far, considered two models:

$$\begin{array}{lcl} \text{linear regression } (y \in \mathbf{R})\text{:} & y \,|\, x & \sim & \mathrm{N}(\mu, \sigma^2) \\ \text{logistic regression } (y \text{ binary})\text{:} & y \,|\, x & \sim & \mathrm{Bernoulli}(\phi) \end{array}$$

- want to generalize these models to work for other kinds of distributions and types of response variables

- observe the following properties of the models above:

  1. model $y \,|\, x \sim F(\theta)$, where F is some distribution
  2. prediction rule is $\hat{f}(x) = \mathrm{E}[y \,|\, x]$
  3. $\mathrm{E}[y \,|\, x]$ given by the model parameters μ and ϕ above
  4. these 'mean' parameters are modeled as $g(w^T x)$, for some g

Generalized linear models

- generalized linear models follow essentially the same structure and include linear and logistic regression as special cases
- based on letting F be any member of the **exponential family**, a very large class of distributions with many convenient properties
- include most of the distributions one uses, *e.g.*, Gaussian, exponential, gamma, beta, Bernoulli, Dirichlet, categorical, Poisson, multinomial (with fixed number of trials), . . .
- have various definitions of increasing generality, so will start with simpler special cases and build from there

Exponential families

class of distributions is in the **exponential family** if

$$\begin{aligned} p(y;\theta) &\propto \exp(\theta y) \\ &= \frac{1}{Z(\theta)}\exp(\theta y) \end{aligned}$$

- $\theta \in \mathbf{R}$ is the **natural parameter**
- $Z(\theta)$ is the normalization constant or **partition function**

often written as

$$p(y;\theta) = \exp(\theta y - A(\theta))$$

where $A(\theta) = \log Z(\theta)$ is the **log partition function**

Exponential families

exponential families have many useful properties, *e.g.*:

- log partition function is convex in θ

$$A(\theta) = \log \int \exp(\theta y)\ dy$$

  so maximizing the log likelihood

$$\log p(y; \theta) = \theta y - A(\theta)$$

  is a convex optimization problem

- mean of the distribution is given by

$$\mathrm{E}[y] = \frac{d}{d\theta} A(\theta)$$

Bernoulli distribution

- recall that if $z \sim \mathrm{Bernoulli}(\phi)$, then

$$\begin{aligned} p(z=1;\phi) &= \phi \\ p(z=0;\phi) &= 1-\phi \end{aligned}$$

  a distribution over $\{0,1\}$ parameterized by $\phi \in [0,1]$

- often use the fact that

$$\exp(\log(x)) = x$$

  e.g., by applying $\exp \cdot \log$ to the 'usual' parametrization of the density function and rearranging

Bernoulli distribution

- rewrite Bernoulli density

$$\begin{aligned} p(z;\phi) &= \phi^z(1-\phi)^{1-z} \\ &= \exp\log(\phi^z(1-\phi)^{1-z}) \\ &= \exp\left(\left(\log\frac{\phi}{1-\phi}\right)z + \log(1-\phi)\right) \end{aligned}$$

- this is an exponential family distribution with

$$\theta = \log\frac{\phi}{1-\phi}, \quad A(\theta) = \log(1+e^\theta)$$

- note that θ is a logit function of ϕ

Bernoulli distribution

- since we know that $\mathrm{E}[z] = \phi$, gives that

$$\mathrm{E}[z] = \phi = \frac{1}{1 + \exp(-\theta)}$$

  since the logit function is an inverse sigmoid function

- could also derive the mapping between $\mathrm{E}[z]$ and θ via

$$\begin{aligned} \frac{d}{d\theta} A(\theta) &= \frac{e^\theta}{1 + e^\theta} \\ &= \frac{1}{1 + \exp(-\theta)} \end{aligned}$$

Generalized linear models

assumptions

1. $y \mid x \sim \mathcal{E}(\theta)$, where $\mathcal{E}$ is an exponential family distribution
2. given x, goal is to predict $\hat{f}(x) = \mathrm{E}[y \mid x]$
3. $\theta = w^T x$

Canonical response function

- to obtain prediction $\hat{f}(x)$ from input x, go through the chain

$$\begin{aligned} \hat{f}(x) &= \mathrm{E}[y \,|\, x] && \text{(assumption 2)} \\ &= g(\theta) && \text{(for some } g\text{)} \\ &= g(w^T x) && \text{(assumption 3)} \end{aligned}$$

- the mapping $g : \theta \mapsto \mathrm{E}[y \,|\, x]$ is known as the **canonical response function** and is given by

$$g(\theta) = \frac{d}{d\theta} A(\theta)$$

- inverse of g is known as the **canonical link function**

- often $\mathrm{E}[y \,|\, x]$ is simply the usual parameter of the distribution (*e.g.*, ϕ for $\mathrm{Bernoulli}(\phi)$), so no need to differentiate A

Logistic regression as a GLM

- choose exponential family distribution $\mathcal{E}(\theta)$ as $\mathrm{Bernoulli}(\phi)$, so

$$\theta = \log \frac{\phi}{1-\phi}, \quad A(\theta) = \log(1+e^{\theta})$$

- prediction rule given by

$$\begin{array}{rcll} \hat{f}(x) & = & \mathbf{E}[y \,|\, x; w] & \text{(assumption 2)} \\ & = & \phi & \text{(expected value of } \mathrm{Bernoulli}(\phi)\text{)} \\ & = & 1/(1+\exp(-\theta)) & \text{(assumption 1 \& } \theta \text{ from above)} \\ & = & g(\theta) & \text{(definition of sigmoid)} \\ & = & g(w^T x) & \text{(assumption 3)} \end{array}$$

Exponential families

- to express some other distributions, like Gaussians, as exponential family distributions, need slightly more general definition

$$p(y;\theta) = h(y) \exp(\theta y - A(\theta))$$

- all the main properties remain

Gaussian distribution with fixed variance

- choose $\sigma^2 = 1$ (for linear regression, σ^2 doesn't matter)
- then follows that

$$\begin{aligned} p(z;\mu) &= (1/\sqrt{2\pi})\exp(-(z-\mu)^2/2) \\ &= (1/\sqrt{2\pi})\exp(-z^2/2)\cdot\exp(\mu z - \mu^2/2) \end{aligned}$$

- this is an exponential family distribution with

$$h(z) = (1/\sqrt{2\pi})\exp(-z^2/2), \quad \theta = \mu, \quad A(\theta) = \theta^2/2$$

Linear regression as a GLM

- let $y \mid x \sim \mathrm{N}(\mu, 1)$, so

$$h(z) = (1/\sqrt{2\pi}) \exp(-z^2/2), \quad \theta = \mu, \quad A(\theta) = \theta^2/2$$

- prediction rule given by

$$\begin{array}{rcll} \hat{f}(x) & = & \mathbf{E}[y \mid x; w] & \text{(assumption 2)} \\ & = & \mu & \text{(expected value of Gaussian)} \\ & = & \theta & \text{(assumption 1 \& } \theta \text{ from above)} \\ & = & w^T x & \text{(assumption 3)} \end{array}$$

Exponential families

- the most general definition we will use is

$$p(y;\theta) = h(y)\exp(\theta^T\varphi(y) - A(\theta))$$

  - $\theta = (\theta_1, \ldots, \theta_K)$ is now a vector of natural parameters
  - $\varphi(y) = (\varphi_1(y), \ldots, \varphi_K(y))$ is a vector of **sufficient statistics**

- previous properties carry over, with adjustments, *e.g.*,

$$\nabla A(\theta) = \mathrm{E}[\varphi(y)]$$

- GLMs are as before, but $\hat{f}(x) = \mathrm{E}[\varphi(y) \mid x]$

Sufficient statistics

- a **statistic** is a function of a random variable
- informally, sufficiency characterizes what is essential in a dataset: if $X \sim F(\theta)$, then the statistic T is **sufficient for** θ if there is no information in X about θ beyond what is in $T(X)$
- given density $p(x; \theta)$, the statistic T is sufficient for θ if and only if there are functions $f, g \geq 0$ such that

  $$p(x; \theta) = f(x) g(T(x), \theta)$$

  (known as Neyman-Fisher factorization theorem)
- maximum likelihood estimate of θ only depends on $T(X)$
- application: large-scale streaming data

Sufficient statistics and exponential families

sufficiency is a more general concept than the exponential family, but is also closely connected

(a) can obtain sufficient statistics by inspection (φ is sufficient for θ)

(b) only* distributions having sufficient statistics with dimension bounded as sample size increases (Pitman-Koopman-Darmois thm.)

given i.i.d. random variables $X = (X_1, \ldots, X_N)$ with the same exponential family density, joint density given by

$$p(x;\theta) = \left(\prod_{i=1}^{N} h(x_i)\right) \exp\left(\theta^T \sum_{i=1}^{N} \varphi(x_i) - NA(\theta)\right)$$

so X is also exponential with statistic $\sum_{i=1}^{N} \varphi(x_i)$

Gaussian with unknown variance

- (univariate) Gaussian distribution

$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

  can be written in exponential family form, with

$$h(x) = \frac{1}{\sqrt{2\pi}}, \quad \theta = \begin{bmatrix} \mu/\sigma^2 \\ -1/2\sigma^2 \end{bmatrix}, \quad \varphi(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix}$$

$$A(\theta) = \frac{\mu}{2\sigma^2} + \log\sigma = -\frac{\theta_1^2}{4\theta_2} - \frac{1}{2}\log(-2\theta_2)$$

- similar result for multivariate case with

$$\varphi(x) = \left(\sum_{i=1}^N x_i, \sum_{i=1}^N x_i x_i^T\right)$$

Maximum entropy and sufficient statistics

- another motivation for exponential family form
- the **entropy** of a discrete random variable

$$H(X) = -\sum_x p(x) \log p(x)$$

  is a measure of the average information content of X

- can be viewed as 'expected surprisal' $\mathrm{E}[-\log p(X)]$

Maximum entropy and sufficient statistics

- suppose there are certain features of interest of the data
- consider finding distribution p consistent with some constraints on these features f_i, but want to be agnostic about p otherwise
- the solution to

$$\begin{array}{ll} \text{maximize} & H(X) \\ \text{subject to} & \mathrm{E}_p[f_i(X)] = \alpha_i, \quad i = 1, \ldots, m \end{array}$$

  with variable p is a distribution in the (exponential family) form

$$p(x;\theta) = \frac{1}{Z(\theta)} h(x) \exp\left(\sum_{i=1}^{m} \theta_i f_i(x)\right)$$

- **method of moments**: let α_i be empirical expectations of f_i

Terminology

- exponential family models
- log-linear models
- maximum entropy models

- Gibbs distribution
- Boltzmann distribution
- energy-based model
- conditional random field

Multinomial distribution

- want to build classifier that handles more than two outcomes
- use the multinomial distribution, which models the probability of rolling a k-sided die n times
- mass function given by

$$p(x_1, \ldots, x_k) = \frac{n!}{\prod_{i=1}^{k} x_i!} \prod_{i=1}^{k} \phi_i^{x_i}$$

  where $x_i \in \{1, \ldots, n\}$

- when $k = 2$, reduces to binomial distribution

Categorical distribution

- when $n = 1$, called a **categorical distribution**, a generalization of the Bernoulli distribution with mass

$$p(x) = \prod_{i=1}^{k} \phi_i^{[x=i]}$$

  so $p(x = i) = \phi_i$

- often represent outcomes of categorical distributions as 'one-hot' vectors $e_1, \ldots, e_k \in \mathbf{R}^k$

- in machine learning areas, 'multinomial' is often used to refer to the categorical distribution

- often OK, but sometimes causes confusion and have to be careful: *e.g.*, consider n different categorical variables vs one multinomial variable with n trials

Categorical distribution

- can parametrize categorical (or multinomial) distribution either with $\phi_1, \dots, \phi_k$, or $\phi_1, \dots, \phi_{k-1}$, to account for $\phi_k = 1 - \sum_i [i \neq k]\phi_i$; here, use the latter

- member of the exponential family with

$$\varphi_i(x) = [x = i], \quad \theta = \begin{bmatrix} \log(\phi_1/\phi_k) \\ \vdots \\ \log(\phi_{k-1}/\phi_k) \end{bmatrix}, \quad A(\theta) = -\log(\phi_k)$$

so $\varphi(x) \in \mathbf{R}^{k-1}$

Softmax regression

- consider GLM with categorical response
- prediction rule given by

$$\hat{f}(x) = \mathrm{E}[\varphi(y) \mid x] = \phi = g(\theta) = g(w^T x)$$

where canonical response function

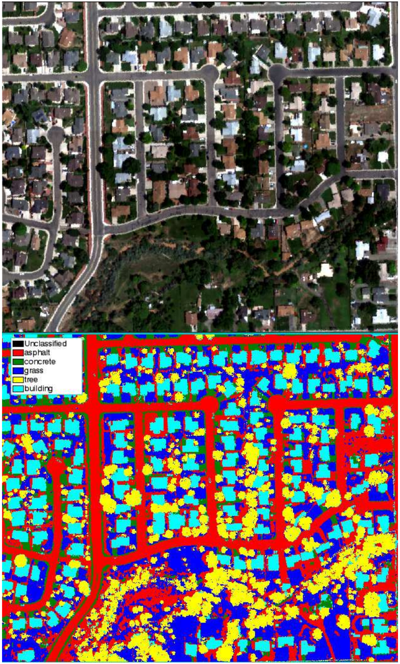
$$g(\theta)_i = \frac{\exp(\theta_i)}{\sum_j \exp(\theta_j)}, \quad g : \mathbf{R}^{k-1} \to [0, 1]^{k-1}$$

is the **softmax function**

Geospatial imaging

(Wolfe et al., Harris Corporation)

- classify components of (multispectral or hyperspectral) images
- classification (via softmax regression) of urban environment into 5 classes: asphalt, concrete, grass, tree, building
- data provided by National Ecological Observatory Network (NEON) on urban test site in Fruita, Colorado
- images from imaging spectrometer; use RGB + near-infrared bands
- combine with height data by using LIDAR on NEON point clouds, along with reflectance, elevation, texture, shape
- ‘examples’ are attributes of a single pixel


Unclassified
asphalt
concrete
grass
tree
building

Insurance claim modeling

(Goldburd, Khare, Tevet)

- GLMs are pervasive in insurance modeling: *e.g.*, predict severity of auto claims using driver age and marital status
- model: claim severity is gamma distributed; use log link function (captures premiums being positive, multiplicative behavior like violations increasing premium by x%)
- if $w = (5.8, 0.1, -0.15)$, then claim severity for 25 year old married driver is \$3,463.38 via

$$\log \mathrm{E}[y \mid x] = 5.8 + 0.1 \times 25 + (-0.15) \times 1 = 8.15$$

- also useful to interpret as

$$\begin{aligned} \mu &= e^{5.8} \times e^{0.1(25)} \times e^{-0.15(1)} \\ &= \$330.30 \times 12.18 \times 0.86 \end{aligned}$$

i.e., 'base average severity' of \$330.30 with additional factors applied

Generative classifiers

Generative models

- discriminative models estimate $p(y \mid x)$ (*e.g.*, logistic regression) or directly learn a mapping from the input to output space (*e.g.*, SVM)

- alternatively, can model the full joint distribution $p(x, y)$; these are called **generative** because they can generate (x_i, y_i)

- usually, model the joint by modeling $p(x \mid y)$ and $p(y)$, and positing the following recipe for how the data was generated:

  1. sample y_i from $p(y)$
  2. sample x_i from $p(x \mid y_i)$

  useful to 'read' generative models using this data generation story

- distributions typically chosen to be in the exponential family

Generative classifiers

- then posterior distribution $p(y \mid x)$ can be derived by reversing the generative process via Bayes' rule

$$p(y \mid x) = \frac{p(x, y)}{p(x)} = \frac{p(x, y)}{\int_y p(x, y)} = \frac{p(x \mid y)p(y)}{\int_y p(x \mid y)p(y)}$$

- denominator (normalization constant) can be expressed directly using class priors $p(y)$ and class-conditional densities $p(x \mid y)$

- normalization constant not needed strictly to make predictions

$$\begin{aligned} \operatorname*{argmax}_y p(y \mid x) &= \operatorname*{argmax}_y \frac{p(x \mid y)p(y)}{p(x)} \\ &= \operatorname*{argmax}_y p(x \mid y)p(y) \end{aligned}$$

- to suppress importance of normalization constant, can write

$$p(y \mid x) \propto p(x \mid y)p(y)$$

Outline

Gaussian discriminant analysis

Naive Bayes classifier

Multivariate Gaussian distribution

- if $X \sim \mathrm{N}(\mu, \Sigma)$, with $\mu \in \mathbf{R}^n$, $\Sigma \succ 0$, density given by

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2}(\det \Sigma)^{1/2}} \left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu) \right)$$

- also have

$$\begin{aligned}
\mathrm{E}[X] &= \int_x x p(x)\ dx = \mu \\
\mathbf{var}[X] &= \mathrm{E}[(X - \mathrm{E}[X])(X - \mathrm{E}[X])^T] \\
&= \mathrm{E}[XX^T] - \mathrm{E}[X]\mathrm{E}[X]^T \\
&= \Sigma
\end{aligned}$$

Gaussian discriminant analysis

- consider binary classification problem
- assume data comes from generative model

$$\begin{aligned} y &\sim \mathrm{Bernoulli}(\phi) \\ x \,|\, y = 0 &\sim \mathrm{N}(\mu_0, \Sigma) \\ x \,|\, y = 1 &\sim \mathrm{N}(\mu_1, \Sigma) \end{aligned}$$

  i.e., data comes from one of two Gaussians chosen with a ϕ-coin flip

- when class-conditional densities $x \,|\, y$ share the same covariance matrix Σ, model called **linear discriminant analysis**
- to obtain other models, use other forms for $x \,|\, y$

Maximum likelihood estimation

- estimate $w = (\phi, \mu_k, \Sigma)$ by maximizing $p(\mathcal{D} \mid w)$

$$
\begin{aligned}
\ell(w) &= \log L(w) \\
&= \log \prod_{i=1}^{N} p(x_i, y_i; w) \\
&= \log \prod_{i=1}^{N} p(x_i \mid y_i; w) p(y_i; w) \\
&= \sum_{i=1}^{N} \log p(x_i \mid y_i; w) + \sum_{i=1}^{N} \log p(y_i; w) \\
&= \sum_{i=1}^{N} \log p(x_i \mid y_i; \mu_0, \mu_1, \Sigma) + \sum_{i=1}^{N} \log p(y_i; \phi)
\end{aligned}
$$

Maximum likelihood estimation

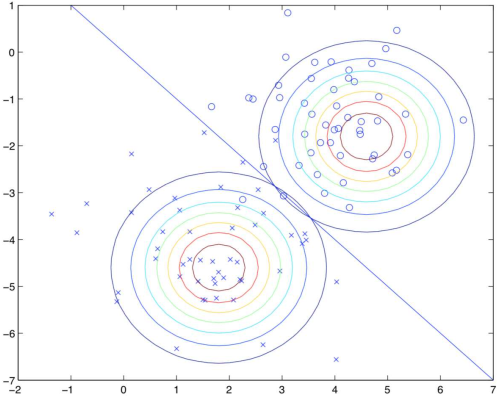
maximum likelihood estimates of parameters given by

$$\begin{aligned}
\hat{\phi} &= \frac{1}{N}\sum_{i=1}^{N}[y_i = 1] \\
\hat{\mu}_k &= \frac{\sum_{i=1}^{N}[y_i = k]x_i}{\sum_{i=1}^{N}[y_i = k]} \\
\hat{\Sigma} &= \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_{y_i})(x_i - \mu_{y_i})^T
\end{aligned}$$

very natural interpretations:

- $\hat{\phi}$ is empirical proportion of positive label in $\mathcal{D}$
- $\hat{\mu}_k$ is empirical average of x_i with label k
- $\hat{\Sigma}$ is empirical covariance, with variance measured to relevant mean

GDA as a linear classifier



GDA and logistic regression

- consider posterior of positive label as function of x

$$p(y = 1 \,|\, x; w) = \frac{1}{1 + \exp(-\theta^T x)}$$

  where θ is a function of $w = (\phi, \mu_0, \mu_1, \Sigma)$

- *i.e.*, has the same functional form as logistic regression, but logistic regression makes no Gaussian assumption about $x \,|\, y$

- GDA makes stronger assumptions and is more data efficient ('asymptotically efficient') if the model is accurate

- logistic regression is more robust to model misspecification (*e.g.*, $p(y \,|\, x)$ also logistic if $x \,|\, y$ in certain class of exponential families)

Multiclass Gaussian discriminant analysis

- more generally, consider modeling $p(y = k \mid x)$ for $k \in [K]$ as Gaussians with equal covariance

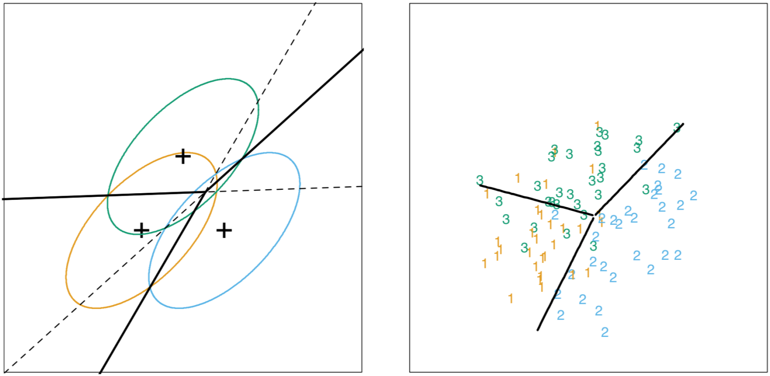
- find that log odds ratio between two classes

$$\log \frac{p(y = k \mid x)}{p(y = k' \mid x)} = w^T x$$

  for some w, *i.e.*, linear in x

- get linear decision boundaries, and obtain MLEs of the parameters along the same lines as before

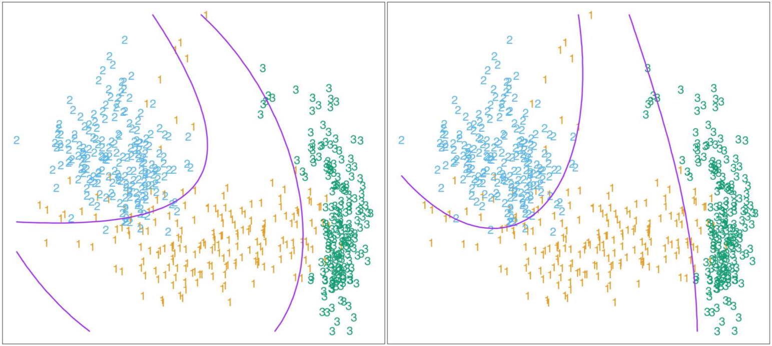
Multiclass Gaussian discriminant analysis



Quadratic discriminant analysis

- consider discriminant analysis where covariances *not* equal
- then decision boundaries described by quadratic equations
- similar, but not identical to, linear GDA in enlarged quadratic space

Quadratic discriminant analysis



Outline

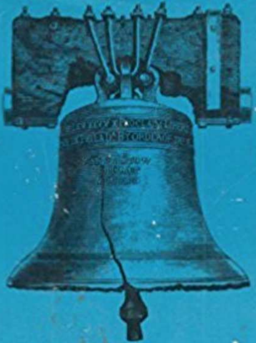
Gaussian discriminant analysis

Naive Bayes classifier

Classifier for discrete inputs

- binary classification problem where inputs x_i are discrete
- example: spam classification
- assume $x \in \{0, 1\}^{|V|}$, with $x^j = 1$ indicating that feature j is true, *e.g.*, example contains some word
- vocabulary V is set of all words being considered
- often take V to be all words observed in training data, minus very common 'stopwords' like 'the', 'and', *etc.*

Frederick Mosteller / David L. Wallace



Inference & Disputed Authorship: The Federalist

Vocabulary and feature vector representation

$$
x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}
\qquad
\begin{array}{c} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \text{buy} \\ \vdots \\ \text{zymurgy} \end{array}
$$

Classifier for discrete inputs

- consider generative model with

$$\begin{array}{rcl} y & \sim & \text{Bernoulli}(\phi) \\ x \mid y = 0 & \sim & \text{Categorical}(\theta_0) \\ x \mid y = 1 & \sim & \text{Categorical}(\theta_1) \end{array}$$

- **note**: same as GDA model with categorical distributions

- **problem**: because here $x \in \{0,1\}^{|V|}$, if, *e.g.*, 50K words in vocabulary, then $x \mid y = k$ has $2^{50000} - 1 \approx 3 \times 10^{15051}$ parameters

Naive Bayes assumption

- idea: to simplify, assume that all the features x^j are conditionally independent of each other given y, implying that

$$p(x \mid y) = \prod_{j=1}^{n} p(x^j \mid y)$$

- gives the model

$$\begin{aligned} y &\sim \text{Bernoulli}(\phi) \\ x^j \mid y = 0 &\sim \text{Bernoulli}(\theta_0^j) \\ x^j \mid y = 1 &\sim \text{Bernoulli}(\theta_1^j) \end{aligned}$$

  which has only $2|V| + 1$ parameters

- $x^j \mid y$ could also be categorical, discretized continuous, *etc.*

Bag of words and exchangeability

- especially for text data, naive Bayes (conditional independence) assumption called a **bag of words model**
- equivalent to assuming that *order* of words doesn't matter
- in statistics, called **exchangeability**, and exchangeable sequences of random variables have various useful properties

Maximum likelihood estimation

maximum likelihood estimation as in GDA, giving

$$\begin{aligned}\hat{\phi} &= \frac{1}{N}\sum_{i=1}^{N}[y_i = 1] \\ \hat{\theta}_k^j &= \frac{\sum_{i=1}^{N}[x_i^j = 1, y_i = k]}{\sum_{i=1}^{N}[y_i = k]}\end{aligned}$$

very natural interpretations:

- $\hat{\phi}$ is empirical proportion of positive label in $\mathcal{D}$
- $\hat{\theta}_k^j$ is empirical proportion of documents containing j in label k

e.g., $\hat{\theta}_{\text{spam}}^{\text{viagra}} = 0.4$ means 'viagra' appears in 40% of the emails labeled as spam in the training set

Labeling new points

to classify new example x, compute

$$
\begin{aligned}
p(y = 1 \mid x) &= \frac{p(x \mid y = 1) p(y = 1)}{p(x)} \\
&= \frac{p(x \mid y = 1) p(y = 1)}{p(x \mid y = 0) p(y = 0) + p(x \mid y = 1) p(y = 1)} \\
&= \frac{p(y = 1) \prod_{j=1}^{|V|} p(x^j \mid y = 1)}{p(y = 0) \prod_{j=1}^{|V|} p(x^j \mid y = 0) + p(y = 1) \prod_{j=1}^{|V|} p(x \mid y = 1)}
\end{aligned}
$$

Smoothed estimators

- problem: $p(x^j \mid y) = 0$ if x^j is not in the training set, so $p(y = k \mid x) = 0/0$

- a general problem with maximum likelihood estimators

- prompted NLP researchers to come up with a range of heuristic 'smoothed' estimates

- in general, if estimating parameters of a multinomial with N trials from d observations $z_1, \ldots, z_d$, could instead use estimator

$$\hat{\theta}_i = \frac{z_i + \alpha}{N + d\alpha}$$

  where $\alpha > 0$ is a *pseudocount*

- called **Laplace smoothing** or **additive smoothing**

Multinomial event model

- previous model known as (multivariate) Bernoulli event model:

  1. flip a ϕ-coin to decide whether document is spam/not
  2. for each $j \in V$, flip θ_k^j-coin to include word or not

- could also consider **multinomial event model**, in which each x^j is categorical over the vocabulary

- still a bag of words model, but very different interpretation

  - multinomial accounts for multiple occurrences of words
  - Bernoulli may overweight single occurrences in long documents
  - Bernoulli accounts for non-occurrence of words

- multinomial models generation of words while Bernoulli models generation of documents

Maximum likelihood estimation

maximum likelihood estimation very similar to before, except

$$\begin{aligned}\hat{\phi} &= \frac{1}{N}\sum_{i=1}^{N}[y_i = 1] \\ \hat{\theta}_k^l &= \frac{\sum_{i=1}^{N}\sum_{j=1}^{N_i}[x_i^j = l, y_i = k]}{\sum_{i=1}^{N}[y_i = k]N_i}\end{aligned}$$

where N_i is number of words in document i

- $\hat{\phi}$ is empirical proportion of positive label in $\mathcal{D}$ (as before)
- $\hat{\theta}_k^l$ is empirical proportion of word l in label k

e.g., $\hat{\theta}_{\text{spam}}^{\text{viagra}} = 0.4$ means 'viagra' is 40% of the words across all spam emails in the training set

Support vector machines

Outline

Support vector machines

Duality

Kernelization

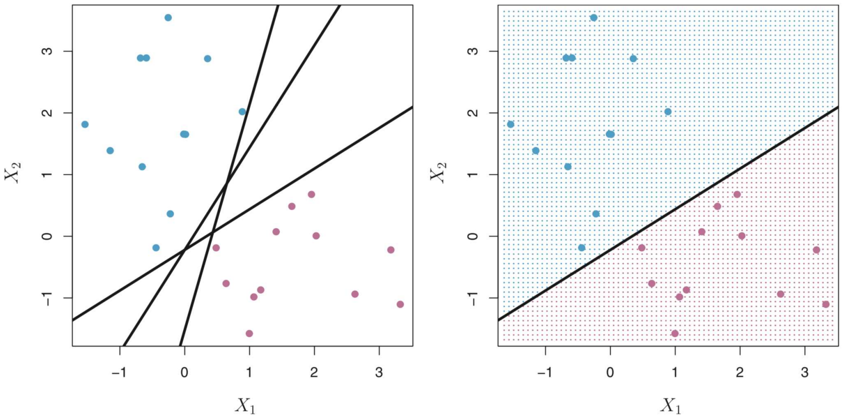
Binary classification

- dataset $\mathcal{D} = \{(x_1, y_1), \dots, (x_N, y_N)\}$
- consider labels $y_i \in \{-1, 1\}$ instead of $\{0, 1\}$
- parameters $w \in \mathbf{R}^n$, $b \in \mathbf{R}$ (intercept)
- consider *directly* fitting w, b to give a linear decision boundary

$$w^T x + b = 0$$

  so $\hat{f}(x) = \mathbf{sign}(w^T x + b)$
- **assume** for now $\mathcal{D}$ is linearly separable

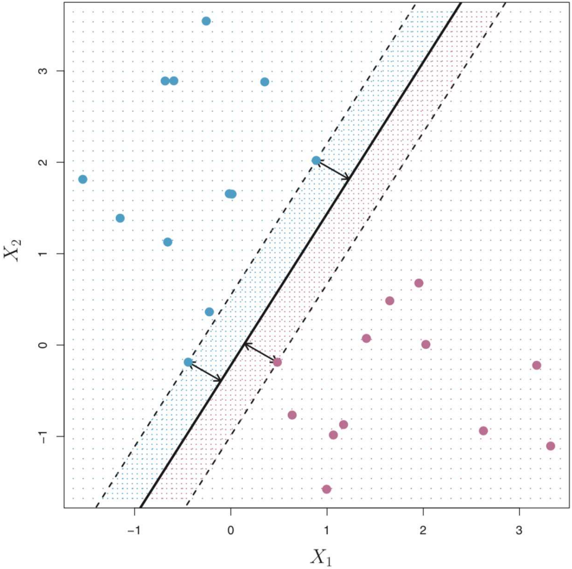
Separating hyperplanes



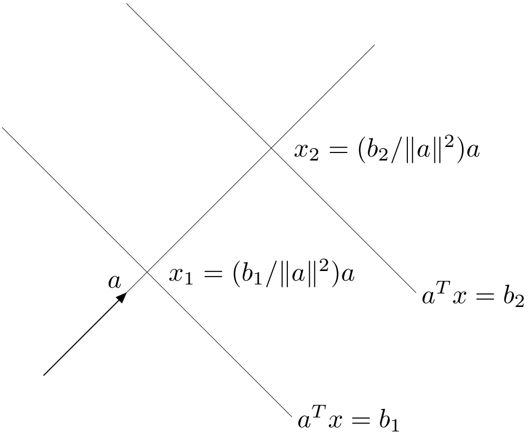
Choosing a separating hyperplane

- since there are multiple separating hyperplanes, need to choose
- there is some distance between the hyperplane and the closest point on either side
- first, observe that parameters (w, b) of hyperplane $w^T x + b = 0$ can be rescaled to $(\alpha w, \alpha b)$, so should choose scaling
- normalize (w, b) so anything with $w^T x + b \geq 1$ is label 1 and points with $w^T x + b \leq -1$ is label -1

Separating hyperplanes



Geometry of parallel hyperplanes



distance between hyperplanes is $\|x_1 - x_2\|_2 = |b_1 - b_2|/\|a\|_2$

Geometry of parallel hyperplanes

- previous diagram shows that distance is given by $2/\|w\|_2$
- could also see this via the following:
  - let x^{neg} be arbitrary negative example on $w^T x + b = -1$
  - let x^{pos} be the projection of x^{neg} onto $w^T x + b = 1$
  - $x^{\text{pos}} = x^{\text{neg}} + \lambda w$ for some λ, and $\lambda \|w\|_2$ is distance between lines
  - solve for λ with three equations above, giving $\lambda = 2/\|w\|_2^2$
- **criterion**: choose w, b to push these lines as far apart as possible
- minimal distance of point to hyperplane is called **margin**, so this criterion typically called *maximum margin classification*

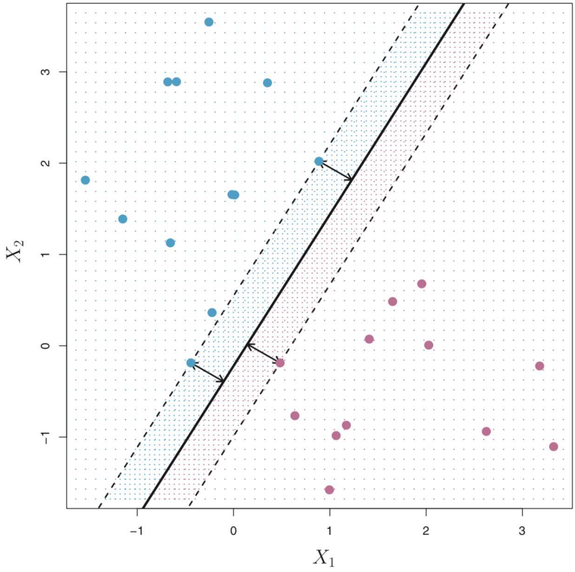
Max-margin classifier

- maximize distance $2/\|w\|_2$ between hyperplanes, subject to constraints that hyperplanes correctly classify points in $\mathcal{D}$
- transform maximization of $2/\|w\|_2$ to minimization of $\|w\|_2^2/2$
- gives the convex QP

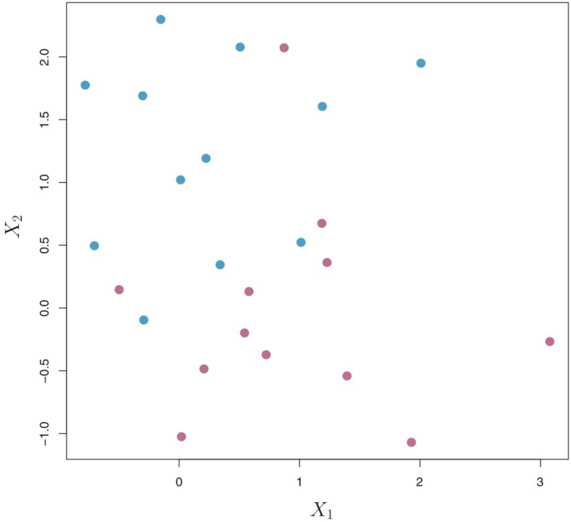
$$\begin{array}{ll} \text{minimize} & (1/2)\|w\|_2^2 \\ \text{subject to} & y_i(w^T x_i + b) \geq 1, \quad i = 1, \ldots, N \end{array}$$

where constraints say margin $u_i = y_i(w^T x_i + b)$ is positive, *i.e.*, example (x_i, y_i) classified correctly

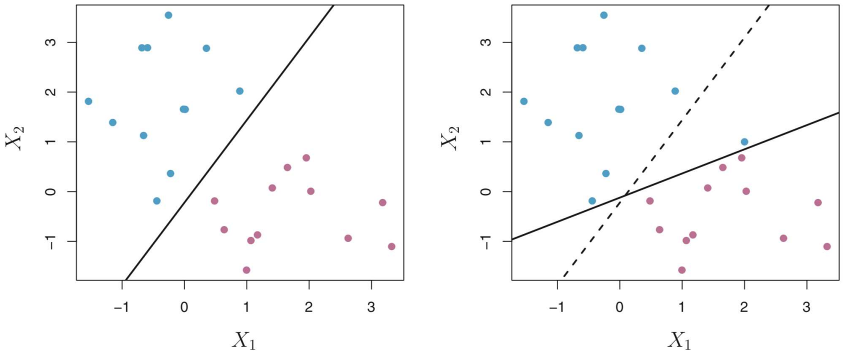
Max-margin classifier



Nonseparable data



Influence of outliers



Soft margin

- for nonseparable data, previous problem is infeasible
- **soft margin**: allow some examples to have negative margin
- roughly, replace $u_i \geq 0$ with $u_i \geq -t_i$, $t_i \geq 0$, and then encourage most t_i to be small or zero
- gives SVM problem

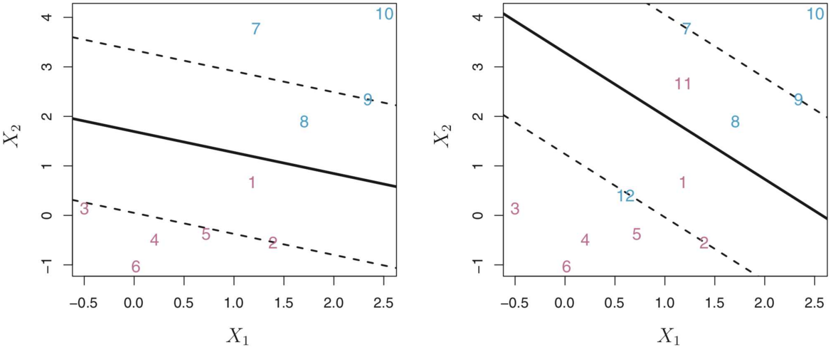
$$\begin{array}{ll} \text{minimize} & (1/2)\|w\|_2^2 + \lambda \mathbf{1}^T t \\ \text{subject to} & y_i(w^T x_i + b) \geq 1 - t_i, \quad i = 1, \ldots, N \\ & t \succeq 0 \end{array}$$

  where $\lambda > 0$ is a trade-off parameter
- can view as scalarization of multicriterion objective $(\|w\|_2^2, \|t\|_1)$

Slack variables

- $t_i = 0$: x_i is on the correct side of the margin
- $t_i > 0$: x_i is on the wrong side of the margin (violated margin)
- $t_i > 1$: x_i is on the wrong side of the hyperplane

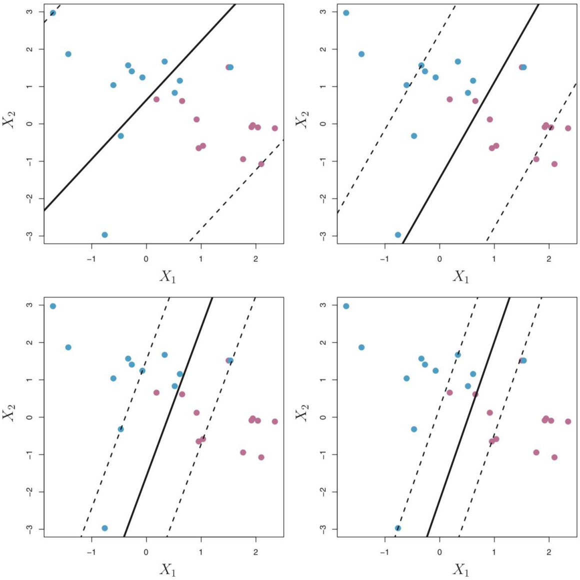
Soft margin and outliers



Observations

- a non-probabilistic method
- max-margin hyperplane only depends on points on the boundary or on wrong side of margin (called **support vectors**)
- the slack variable t will generally be sparse
- model parameter $\lambda > 0$ controls size of margin

Choosing λ



Outline

Support vector machines

Duality

Kernelization

Duality

- duality in mathematics is a principle or theme, not a theorem
- shows up in many forms, and is pervasive in math and physics
- fundamental idea: two different perspectives on the same object
- *i.e.*, can associate with a given mathematical object a related 'dual' object that helps one understand the properties of the original object

Duality

- if the dual of an object X is denoted X^*, duality often satisfies two key properties:

  (a) involution: $X^{**} = X$

  (b) order-reversing: if $X \leq Y$, then $Y^* \leq X^*$ (for some $\leq$)

- often an additional property as well

  (c) 'regularity': a duality construction for a 'nice' subset $\mathcal{X}^{\mathrm{nice}} \subseteq \mathcal{X}$ has $X^* \in \mathcal{X}^{\mathrm{nice}}$ for $X \in \mathcal{X}$, with X^{**} being the 'closest' nice approximation to X in some sense (often some kind of closure)

Set complement

if $A \subseteq X$, let A^c be the complement of the set A in X

(a) $(A^c)^c = A$

(b) if $A \subseteq B \subseteq X$, then $B^c \subseteq A^c$

can say that intersection and union are 'dual operations' on sets due to de Morgan's laws

$$\begin{aligned}(A \cap B)^c &= A^c \cup B^c \\ (A \cup B)^c &= A^c \cap B^c\end{aligned}$$

Orthogonal complement

if $L \subseteq V$ is a subspace of a (finite dimensional) vector space V, recall that

$$L^{\perp} = \{x \in V \mid x^T z = 0 \text{ for all } z \in L\}$$

(a) $(L^{\perp})^{\perp} = L$

(b) if $\dim L \leq \dim M$ then $\dim M^{\perp} \leq \dim L^{\perp}$

(b) if $L \subseteq M$ then $M^{\perp} \subseteq L^{\perp}$

(c) if $S \subseteq V$ is a set, then $S^{\perp}$ is a subspace, and $(S^{\perp})^{\perp} = \operatorname{span} S$

orthogonal decomposition: for $x \in V$ and subspace L,

$$x = \Pi_L(x) + \Pi_{L^{\perp}}(x)$$

Negation

let $x \in \mathbf{R}$

(a) $-(-x) = x$

(b) if $x \leq y$ then $-y \leq -x$

this is an order-reversing involution, but too dull to be called duality: it doesn't really give two different perspectives on anything

Duality in linear algebra

- **idea**: shift perspective between points (vectors) and linear functions
- more interesting than orthogonal complement example, because the duality involves shifting between two types of objects

Duality in linear algebra

- given vector space V, the **dual space** of V is defined as

$$V^* = \{f : V \to \mathbf{R} \mid f \text{ linear}\}$$

  elements $f \in V^*$ called **linear functionals** on V

- a vector space under the operations

$$\begin{aligned} (f+g)(x) &= f(x) + g(x) \\ (\alpha f)(x) &= \alpha(f(x)) \end{aligned}$$

- each $z \in \mathbf{R}^n$ has an associated $f_z \in (\mathbf{R}^n)^*$ given by $f_z(x) = z^T x$

- every $f \in (\mathbf{R}^n)^*$ has this form (Riesz representation theorem)

- *i.e.*, $(\mathbf{R}^n)^*$ consists of row vectors, interpreted as functions

Duality in linear algebra

- $\mathbf{R}^n$ and $(\mathbf{R}^n)^*$ are isomorphic ($\mathbf{R}^n$ is self-dual), so the duality machinery appears somewhat useless in finite dimensions
- **however**, still have very different interpretations
- in particular, will visualize linear functionals not as points in dual space but as hyperplanes in primal space, and vice versa

$$\text{hyperplanes } H \text{ in } V \quad \Longleftrightarrow \quad \text{linear functionals } f : V \to \mathbf{R} \text{ in } V^*$$

- hyperplanes and transposes are pervasive in dual constructions in optimization for this reason
- gives intuitive interpretations of other dual constructions

Dual norm

- given a general norm $\|\cdot\|$ on $\mathbf{R}^n$, its **dual norm** is

$$\|z\|_* = \sup\{z^T x \mid \|x\| \leq 1\}$$

  – dual of $\|\cdot\|_2$ is $\|\cdot\|_2$ (Euclidean norm is 'self-dual')
  – $\|\cdot\|_1$ and $\|\cdot\|_\infty$ are duals of each other

- interpret $\|\cdot\|_*$ as a norm on $(\mathbf{R}^n)^*$, *i.e.*, a norm on *functions*

- $\|z\|_*$ is the amount the function z^T lengthens vectors x, over vectors x in the unit ball

- *i.e.*, the operator norm of z^T, a standard norm for functions

Dual cones and generalized inequalities

dual cone of a cone K:

$$K^* = \{z \mid z^T x \geq 0 \text{ for all } x \in K\}$$

- $K = \mathbf{R}^n_+$: $K^* = \mathbf{R}^n_+$
- $K = \mathbf{S}^n_+$: $K^* = \mathbf{S}^n_+$
- $K = \{(x,t) \mid \|x\|_2 \leq t\}$: $K^* = \{(x,t) \mid \|x\|_2 \leq t\}$
- $K = \{(x,t) \mid \|x\|_1 \leq t\}$: $K^* = \{(x,t) \mid \|x\|_\infty \leq t\}$

first three examples are **self-dual** cones

dual cones of proper cones are proper, hence define generalized inequalities:

$$z \succeq_{K^*} 0 \quad \Longleftrightarrow \quad z^T x \geq 0 \text{ for all } x \succeq_K 0$$

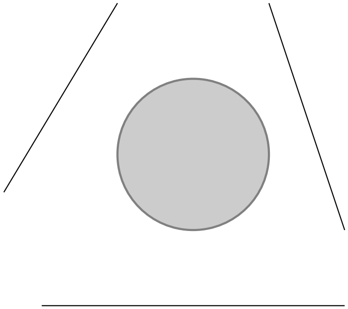
i.e., K^* is linear functionals positive (as functions) on K

Duality in convex optimization

- as in linear algebra, duality in convex analysis also involves shifting perspective between points and hyperplanes (or linear functionals)
- get dual constructions for sets, functions, and optimization problems

Duality for convex sets

a closed convex set C is the intersection of the closed halfspaces containing it



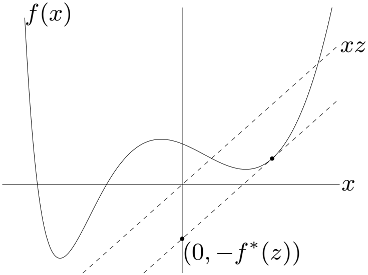
Duality for convex functions

- apply convex duality principle for sets to $\mathbf{epi}\, f$
- a closed proper convex function is the pointwise supremum of its affine underestimators
- translating this geometric idea to the language of functions gives the definition of the conjugate function f^*

The conjugate function

the **conjugate** of a function f is

$$f^*(z) = \sup_{x \in \mathbf{dom}\, f} \left(z^T x - f(x)\right)$$



- $\mathbf{dom}\, f^* \subseteq (\mathbf{R}^n)^*$ is set of slopes z of all possible affine minorizers of f
- $f^*(z)$ is offset from the origin to make that line tangent to f
- $-f^*(0) = \inf f(x)$

The conjugate function

(a) $f^{**} = f$ (if f is closed proper convex)

(b) if $f \leq g$ then $g^* \leq f^*$

(c) if f is not convex, f^* is still closed proper convex, and f^{**} (biconjugate) is the **convex envelope** of f ($\mathbf{epi}\, f^{**} = \mathbf{conv}\, \mathbf{epi}\, f$)

Examples

- negative logarithm $f(x) = -\log x$

$$\begin{array}{rcl} f^*(y) & = & \sup_{x>0}(xy + \log x) \\ & = & \begin{cases} -1 - \log(-y) & y < 0 \\ \infty & \text{otherwise} \end{cases} \end{array}$$

- strictly convex quadratic $f(x) = (1/2)x^T Q x$ with $Q \in \mathbf{S}^n_{++}$

$$\begin{array}{rcl} f^*(y) & = & \sup_{x}(y^T x - (1/2)x^T Q x) \\ & = & \frac{1}{2} y^T Q^{-1} y \end{array}$$

Examples

- often, various notions of duality turn out to be related
- if $L \subseteq \mathbf{R}$ is a vector space, then

$$(I_L)^* = I_{L^\perp}$$

- *i.e.*, the dual of the indicator function of a subspace is the indicator function of the dual of the subspace, for some notion of dual
- here, $f^{**} = f$ corresponds to $(L^\perp)^\perp = L$
- can help extend intuition for, *e.g.*, geometry of orthogonal complement to convex conjugates

Lagrangian

standard form problem (not necessarily convex)

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

variable $x \in \mathbf{R}^n$, domain $\mathcal{D}$, optimal value $p^\star$

Lagrangian: $L : \mathbf{R}^n \times \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$, with $\mathbf{dom}\, L = \mathcal{D} \times \mathbf{R}^m \times \mathbf{R}^p$,

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

- weighted sum of objective and constraint functions
- λ_i is Lagrange multiplier associated with $f_i(x) \leq 0$
- ν_i is Lagrange multiplier associated with $h_i(x) = 0$

Lagrange dual function

Lagrange dual function: $g : \mathbf{R}^m \times \mathbf{R}^p \to \mathbf{R}$,

$$
\begin{aligned}
g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \\
&= \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \right)
\end{aligned}
$$

g is concave, can be $-\infty$ for some λ, ν

lower bound property: if $\lambda \succeq 0$, then $g(\lambda, \nu) \leq p^\star$

proof: if $\tilde{x}$ is feasible and $\lambda \succeq 0$, then

$$
f_0(\tilde{x}) \geq L(\tilde{x}, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = g(\lambda, \nu)
$$

minimizing over all feasible $\tilde{x}$ gives $p^\star \geq g(\lambda, \nu)$

Least norm solution of linear equations

$$\begin{array}{ll} \text{minimize} & x^T x \\ \text{subject to} & Ax = b \end{array}$$

dual function

- Lagrangian is $L(x, \nu) = x^T x + \nu^T(Ax - b)$
- to minimize L over x, set gradient equal to zero:

$$\nabla_x L(x, \nu) = 2x + A^T\nu = 0 \quad \Longrightarrow \quad x = -(1/2)A^T\nu$$

- plug in in L to obtain g:

$$g(\nu) = L((-1/2)A^T\nu, \nu) = -\frac{1}{4}\nu^T A A^T \nu - b^T\nu$$

  a concave function of ν

lower bound property: $p^\star \geq -(1/4)\nu^T A A^T \nu - b^T \nu$ for all ν

Standard form LP

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \quad x \succeq 0 \end{array}$$

dual function

- Lagrangian is

$$\begin{aligned} L(x, \lambda, \nu) &= c^T x + \nu^T (Ax - b) - \lambda^T x \\ &= -b^T \nu + (c + A^T \nu - \lambda)^T x \end{aligned}$$

- L is affine in x, hence

$$g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) = \begin{cases} -b^T \nu & A^T \nu - \lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

g is linear on affine domain $\{(\lambda, \nu) \mid A^T \nu - \lambda + c = 0\}$, hence concave

lower bound property: $p^\star \geq -b^T \nu$ if $A^T \nu + c \succeq 0$

Equality constrained norm minimization

$$\begin{array}{ll} \text{minimize} & \|x\| \\ \text{subject to} & Ax = b \end{array}$$

dual function

$$g(\nu) = \inf_x (\|x\| - \nu^T Ax + b^T\nu) = \begin{cases} b^T\nu & \|A^T\nu\|_* \leq 1 \\ -\infty & \text{otherwise} \end{cases}$$

where $\|v\|_* = \sup_{\|u\|\leq 1} u^T v$ is dual norm of $\|\cdot\|$

proof: follows from $\inf_x(\|x\| - y^T x) = 0$ if $\|y\|_* \leq 1$, $-\infty$ otherwise

- if $\|y\|_* \leq 1$, then $\|x\| - y^T x \geq 0$ for all x, with equality if $x = 0$
- if $\|y\|_* > 1$, choose $x = tu$ where $\|u\| \leq 1$, $u^T y = \|y\|_* > 1$:

$$\|x\| - y^T x = t(\|u\| - \|y\|_*) \to -\infty \quad \text{as } t \to \infty$$

lower bound property: $p^\star \geq b^T\nu$ if $\|A^T\nu\|_* \leq 1$

Lagrange dual and conjugate function

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & Ax \preceq b, \quad Cx = d \end{array}$$

dual function

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathbf{dom}\, f_0} \left(f_0(x) + (A^T\lambda + C^T\nu)^T x - b^T\lambda - d^T\nu \right) \\ &= -f_0^*(-A^T\lambda - C^T\nu) - b^T\lambda - d^T\nu \end{aligned}$$

- recall definition of conjugate $f^*(y) = \sup_{x \in \mathbf{dom}\, f} (y^T x - f(x))$
- simplifies derivation of dual if conjugate of f_0 is known

example: entropy maximization

$$f_0(x) = \sum_{i=1}^n x_i \log x_i, \qquad f_0^*(y) = \sum_{i=1}^n e^{y_i - 1}$$

The dual problem

Lagrange dual problem

$$\begin{array}{ll} \text{maximize} & g(\lambda, \nu) \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

- finds best lower bound on $p^\star$, obtained from Lagrange dual function
- a convex optimization problem; optimal value denoted $d^\star$
- λ, ν are dual feasible if $\lambda \succeq 0$, $(\lambda, \nu) \in \mathbf{dom}\, g$
- often simplified by making implicit constraint $(\lambda, \nu) \in \mathbf{dom}\, g$ explicit

example: standard form LP and its dual

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \qquad \begin{array}{ll} \text{maximize} & -b^T \nu \\ \text{subject to} & A^T \nu + c \succeq 0 \end{array}$$

Weak and strong duality

weak duality: $d^\star \leq p^\star$

- always holds (for convex and nonconvex problems)
- can be used to find nontrivial lower bounds for difficult problems

strong duality: $d^\star = p^\star$

- does not hold in general
- (usually) holds for convex problems
- conditions that guarantee strong duality in convex problems are called **constraint qualifications**

Slater's constraint qualification

strong duality holds for a convex problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

if it is strictly feasible, *i.e.*,

$$\exists x \in \mathbf{int}\, \mathcal{D}: \qquad f_i(x) < 0, \quad i = 1, \ldots, m, \qquad Ax = b$$

- also guarantees that the dual optimum is attained (if $p^\star > -\infty$)
- can be sharpened: *e.g.*, can replace $\mathbf{int}\, \mathcal{D}$ with $\mathbf{relint}\, \mathcal{D}$ (interior relative to affine hull); linear inequalities do not need to hold with strict inequality, ...
- there exist many other types of constraint qualifications

Inequality form LP

primal problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \preceq b \end{array}$$

dual function

$$g(\lambda) = \inf_x \left((c + A^T\lambda)^T x - b^T\lambda \right) = \begin{cases} -b^T\lambda & A^T\lambda + c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

dual problem

$$\begin{array}{ll} \text{maximize} & -b^T\lambda \\ \text{subject to} & A^T\lambda + c = 0, \quad \lambda \succeq 0 \end{array}$$

- from Slater's condition: $p^\star = d^\star$ if $A\tilde{x} \prec b$ for some $\tilde{x}$
- in fact, $p^\star = d^\star$ except when primal and dual are infeasible

Quadratic program

primal problem (assume $P \in \mathbf{S}^n_{++}$)

$$\begin{array}{ll} \text{minimize} & x^T P x \\ \text{subject to} & Ax \preceq b \end{array}$$

dual function

$$g(\lambda) = \inf_x \left(x^T P x + \lambda^T (Ax - b)\right) = -\frac{1}{4}\lambda^T A P^{-1} A^T \lambda - b^T \lambda$$

dual problem

$$\begin{array}{ll} \text{maximize} & -(1/4)\lambda^T A P^{-1} A^T \lambda - b^T \lambda \\ \text{subject to} & \lambda \succeq 0 \end{array}$$

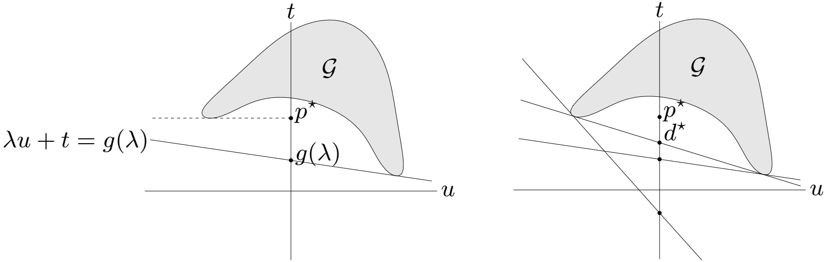
- from Slater's condition: $p^\star = d^\star$ if $A\tilde{x} \prec b$ for some $\tilde{x}$
- in fact, $p^\star = d^\star$ always

Geometric interpretation

for simplicity, consider problem with one constraint $f_1(x) \leq 0$

interpretation of dual function:

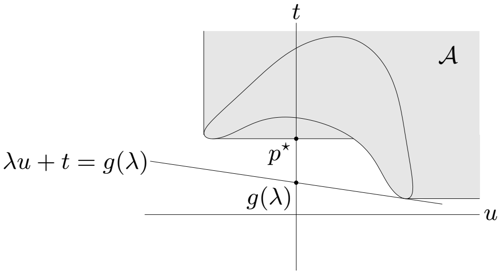
$$g(\lambda) = \inf_{(u,t)\in\mathcal{G}} (t + \lambda u), \qquad \text{where} \quad \mathcal{G} = \{(f_1(x), f_0(x)) \mid x \in \mathcal{D}\}$$



- $\lambda u + t = g(\lambda)$ is (non-vertical) supporting hyperplane to $\mathcal{G}$
- hyperplane intersects t-axis at $t = g(\lambda)$

epigraph variation: same interpretation if $\mathcal{G}$ is replaced with

$$\mathcal{A} = \{(u,t) \mid f_1(x) \le u, f_0(x) \le t \text{ for some } x \in \mathcal{D}\}$$



strong duality

- holds if there is a non-vertical supporting hyperplane to $\mathcal{A}$ at $(0, p^\star)$
- for convex problem, $\mathcal{A}$ is convex, so has supp. hyperplane at $(0, p^\star)$
- Slater's condition: if there exist $(\tilde{u}, \tilde{t}) \in \mathcal{A}$ with $\tilde{u} < 0$, then supporting hyperplanes at $(0, p^\star)$ must be non-vertical

Interpretations

- saddle point interpretation
- game interpretation
- price or tax interpretation

Complementary slackness

assume strong duality holds, $x^\star$ is primal optimal, $(\lambda^\star, \nu^\star)$ is dual optimal

$$\begin{aligned} f_0(x^\star) = g(\lambda^\star, \nu^\star) &= \inf_x \left(f_0(x) + \sum_{i=1}^m \lambda_i^\star f_i(x) + \sum_{i=1}^p \nu_i^\star h_i(x) \right) \\ &\leq f_0(x^\star) + \sum_{i=1}^m \lambda_i^\star f_i(x^\star) + \sum_{i=1}^p \nu_i^\star h_i(x^\star) \\ &\leq f_0(x^\star) \end{aligned}$$

hence, the two inequalities hold with equality

- $x^\star$ minimizes $L(x, \lambda^\star, \nu^\star)$
- $\lambda_i^\star f_i(x^\star) = 0$ for $i = 1, \ldots, m$ (known as complementary slackness):

$$\lambda_i^\star > 0 \Longrightarrow f_i(x^\star) = 0, \qquad f_i(x^\star) < 0 \Longrightarrow \lambda_i^\star = 0$$

Karush-Kuhn-Tucker (KKT) conditions

the following four conditions are called KKT conditions (for a problem with differentiable f_i, h_i):

1. primal constraints: $f_i(x) \leq 0$, $i = 1, \ldots, m$, $h_i(x) = 0$, $i = 1, \ldots, p$
2. dual constraints: $\lambda \succeq 0$
3. complementary slackness: $\lambda_i f_i(x) = 0$, $i = 1, \ldots, m$
4. stationarity: gradient of Lagrangian with respect to x vanishes:

$$\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$$

if strong duality holds and x, λ, ν are optimal, then they must satisfy the KKT conditions

KKT conditions for convex problem

if $\tilde{x}$, $\tilde{\lambda}$, $\tilde{\nu}$ satisfy KKT for a convex problem, then they are optimal:

- from complementary slackness: $f_0(\tilde{x}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$
- from 4th condition (and convexity): $g(\tilde{\lambda}, \tilde{\nu}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$

hence, $f_0(\tilde{x}) = g(\tilde{\lambda}, \tilde{\nu})$

if **Slater's condition** is satisfied: x is optimal if and only if there exist λ, ν that satisfy KKT conditions

- recall that Slater implies strong duality, and dual optimum is attained
- generalizes $\nabla f_0(x) = 0$ condition for unconstrained problem

Duality and problem reformulations

- equivalent formulations of a problem can lead to very different duals
- reformulating the primal problem can be useful when the dual is difficult to derive, or uninteresting

common reformulations

- introduce new variables and equality constraints
- make explicit constraints implicit or vice versa
- transform objective or constraint functions
  e.g., replace $f_0(x)$ by $\phi(f_0(x))$ with ϕ convex, increasing

Introducing new variables and equality constraints

$$\text{minimize} \quad f_0(Ax+b)$$

- dual function is constant: $g = \inf_x L(x) = \inf_x f_0(Ax+b) = p^\star$
- we have strong duality, but dual is quite useless

reformulated problem and its dual

$$\begin{array}{ll} \text{minimize} & f_0(y) \\ \text{subject to} & Ax+b-y=0 \end{array} \qquad \begin{array}{ll} \text{maximize} & b^T\nu - f_0^*(\nu) \\ \text{subject to} & A^T\nu = 0 \end{array}$$

dual function follows from

$$\begin{array}{rcl} g(\nu) & = & \inf_{x,y}(f_0(y) - \nu^T y + \nu^T Ax + b^T\nu) \\ & = & \begin{cases} -f_0^*(\nu) + b^T\nu & A^T\nu = 0 \\ -\infty & \text{otherwise} \end{cases} \end{array}$$

Introducing new variables and equality constraints

norm approximation problem: minimize $\|Ax - b\|$

$$\begin{array}{ll} \text{minimize} & \|y\| \\ \text{subject to} & y = Ax - b \end{array}$$

can look up conjugate of $\|\cdot\|$, or derive dual directly

$$\begin{aligned} g(\nu) &= \inf_{x,y}(\|y\| + \nu^T y - \nu^T A x + b^T \nu) \\ &= \begin{cases} b^T\nu + \inf_y(\|y\| + \nu^T y) & A^T\nu = 0 \\ -\infty & \text{otherwise} \end{cases} \\ &= \begin{cases} b^T\nu & A^T\nu = 0, \quad \|\nu\|_* \le 1 \\ -\infty & \text{otherwise} \end{cases} \end{aligned}$$

dual of norm approximation problem

$$\begin{array}{ll} \text{maximize} & b^T\nu \\ \text{subject to} & A^T\nu = 0, \quad \|\nu\|_* \le 1 \end{array}$$

Implicit constraints

LP with box constraints: primal and dual problem

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & -\mathbf{1} \preceq x \preceq \mathbf{1} \end{array} \qquad \begin{array}{ll} \text{maximize} & -b^T\nu - \mathbf{1}^T\lambda_1 - \mathbf{1}^T\lambda_2 \\ \text{subject to} & c + A^T\nu + \lambda_1 - \lambda_2 = 0 \\ & \lambda_1 \succeq 0, \quad \lambda_2 \succeq 0 \end{array}$$

reformulation with box constraints made implicit

$$\begin{array}{ll} \text{minimize} & f_0(x) = \begin{cases} c^T x & -\mathbf{1} \preceq x \preceq \mathbf{1} \\ \infty & \text{otherwise} \end{cases} \\ \text{subject to} & Ax = b \end{array}$$

dual function

$$\begin{aligned} g(\nu) &= \inf_{-\mathbf{1} \preceq x \preceq \mathbf{1}} (c^T x + \nu^T(Ax - b)) \\ &= -b^T\nu - \|A^T\nu + c\|_1 \end{aligned}$$

dual problem: maximize $-b^T\nu - \|A^T\nu + c\|_1$

Outline

Support vector machines

Duality

Kernelization

Nonlinear decision boundaries

- initial idea to extend SVM to nonlinear case: replace x with $\varphi(x)$
- this is fine, but mathematical structure of SVMs allows for kernelization, a more efficient approach to this
- two main ways to see this
  1. duality
  2. representer theorem
- representer theorem is more general, but uses machinery of reproducing kernel Hilbert spaces

Primal SVM

- recall the SVM problem for linearly separable datasets

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

with variables w, b

- Lagrangian is

$$L(w, b, \alpha) = (1/2)\|w\|_2^2 + \sum_{i=1}^{N} \alpha_i (1 - y_i(w^T x_i + b))$$

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

Dual SVM

- stationarity condition w.r.t. w gives

$$\nabla_w L(w, b, \alpha) = w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0$$

so $w = \sum_{i=1}^{N} \alpha_i y_i x_i$

- plugging into L and simplifying gives

$$L(w, b, \alpha) = \mathbf{1}^T \alpha - \frac{1}{2} \sum_{i,j=1}^{N} y_i y_j \alpha_i \alpha_j x_i^T x_j - b\alpha^T y$$

- stationarity condition w.r.t. b gives

$$\frac{\partial}{\partial b} L(w, b, \alpha) = \sum_{i=1}^{N} \alpha_i y_i = \alpha^T y = 0$$

so last term in L above is zero

Dual SVM

- gives dual problem

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T\alpha - (1/2)\sum_{i,j=1}^{N} y_i y_j \alpha_i \alpha_j x_i^T x_j \\ \text{subject to} & \alpha^T y = 0 \\ & \alpha \succeq 0 \end{array}$$

  with variable $\alpha \in \mathbf{R}^N$

- can reconstruct primal parameters from dual solution $\alpha^\star$ via

$$w^\star = \sum_{i=1}^{N} \alpha_i^\star y_i x_i$$

  (expression for $b^\star$ also available)

Dual form of decision rule

- primal form of decision rule is $w^T x + b$
- dual form given by

$$\begin{aligned} w^T x + b &= \left(\sum_{i=1}^{N} \alpha_i y_i x_i \right)^T x + b \\ &= \sum_{i=1}^{N} \alpha_i y_i x_i^T x + b \end{aligned}$$

  i.e., only requires computing inner products between query point x^{new} and points x_i in the training set

- since α is sparse (nonzero only for support vectors), this is even more efficient to compute

Nonseparable case

- for the nonseparable case, get the dual

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T\alpha - (1/2)\sum_{i,j=1}^N y_i y_j \alpha_i \alpha_j x_i^T x_j \\ \text{subject to} & \alpha^T y = 0 \\ & 0 \preceq \alpha \preceq \lambda \mathbf{1} \end{array}$$

  i.e., only nonnegativity constraint on dual variable changes

- primal parameter w has the form as before

- KKT conditions also imply the following about the margin

$$\begin{array}{rcl} \alpha_i = 0 & \Longrightarrow & u_i \geq 1 \\ \alpha_i = \lambda & \Longrightarrow & u_i \leq 1 \\ \alpha_i \in (0, \lambda) & \Longrightarrow & u_i = 1 \end{array}$$

The kernel trick

- observation: to use a feature map φ, only need to compute inner products $\varphi(x)^T\varphi(z)$

- define the kernel K corresponding to φ as

$$K(x, z) = \varphi(x)^T\varphi(z)$$

- **key idea**: K may be much easier to evaluate than φ, so can *implicitly* learn in high-dimensional feature space implied by φ without computing it directly

- intuitively, kernel functions measure similarity between x and z

Quadratic kernel

- if $x, z \in \mathbf{R}^n$, then $K(x,z) = (x^T z)^2$ is the kernel for

$$\varphi(x) = \begin{bmatrix} x_1x_1 \\ x_1x_2 \\ x_1x_3 \\ x_2x_1 \\ x_2x_2 \\ x_2x_3 \\ x_3x_1 \\ x_3x_2 \\ x_3x_3 \end{bmatrix}$$

  (shown for $n = 3$)

- computing $\varphi(x)$ requires $O(n^2)$ while evaluating K is $O(n)$
- more generally, evaluating $K(x,z) = (x^T z + c)^d$ is $O(n)$ but implicitly works in $O(n^d)$ dimensional space

Mercer's theorem

- what functions of x, z correspond to valid kernels?
- can explicitly construct φ, but this is sometimes awkward
- alternate characterization: the map $K : \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$ is a valid kernel if and only if $\tilde{K} \in \mathbf{S}_+^n$, where the kernel matrix $\tilde{K}$ for a set of points $z_1, \ldots, z_N$ is given by $\tilde{K}_{ij} = K(z_i, z_j)$

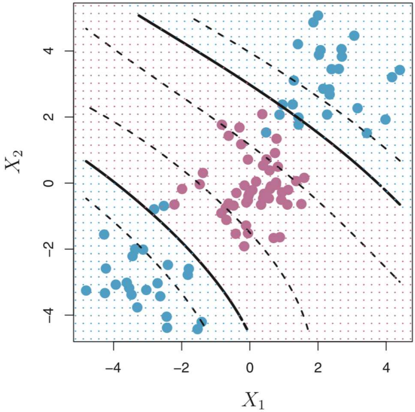
Examples of kernels

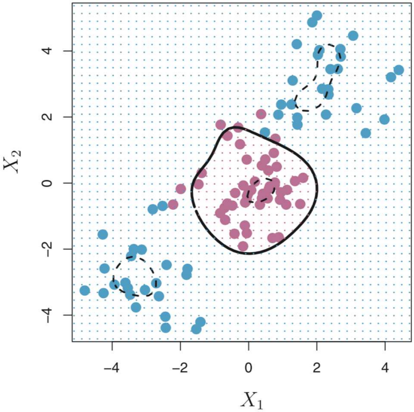
- quadratic kernel: $K(x, z) = (x^T z)^2$
- polynomial kernel: $K(x, z) = (x^T z + c)^d$
- Gaussian kernel (with parameter $\sigma > 0$):

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

- string and sequence kernels
- custom, domain-specific kernels (*e.g.*, bioinformatics)

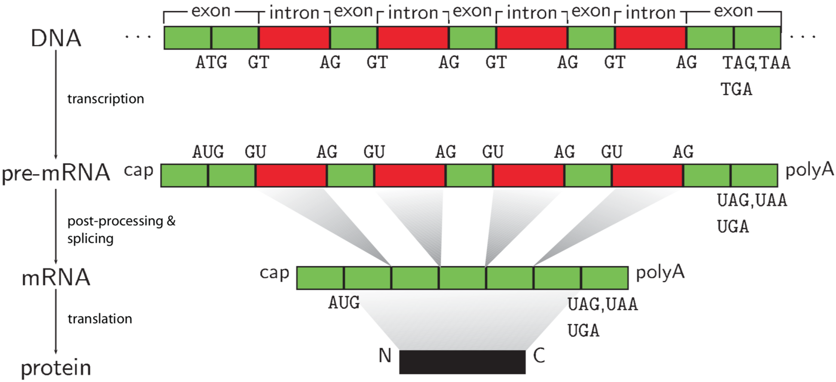
Kernelized support vector machine





Splice site recognition

(Ben-Hur et al., *PLoS Computational Biology*, 2008)



Splice site recognition

(Ben-Hur et al., *PLoS Computational Biology*, 2008)

- a computational gene finding task: find *splice sites* marking boundaries between *exons* and *introns* in eukaryotes
- vast majority of splice sites characterized by presence of specific *dimers* on intronic side of splice site (GT for donor/5' and AG for acceptor/3')
- however, only 0.1%-1% of GT/AG occurrences in genome represent true splice sites
- goal: find acceptor sites in DNA sequences (*C. elegans* dataset)

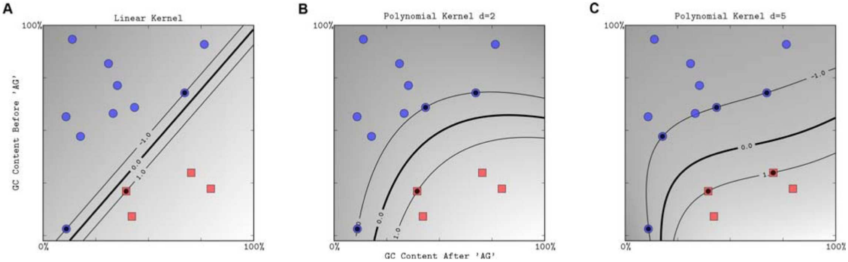
Splice site recognition

(Ben-Hur et al., *PLoS Computational Biology*, 2008)

- first consider just using two (real-valued) features: **GC content** before and after candidate acceptor splice site
- GC content of a DNA sequence is percentage of nucleotides that are G or C (nucleotides are either G, C, A, or T)
- can consider linear, polynomial, and Gaussian kernels

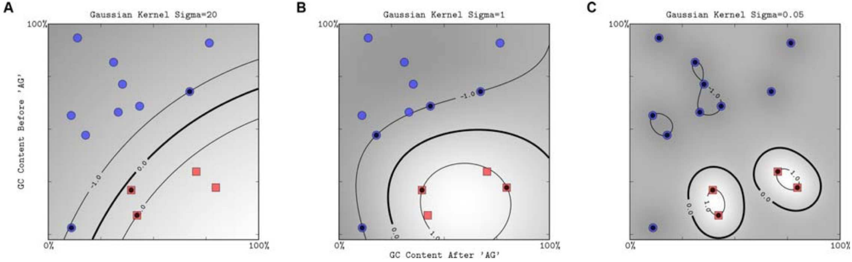
Polynomial kernel (increasing d)

(Ben-Hur et al., *PLoS Computational Biology*, 2008)



Gaussian kernel (decreasing σ)

(Ben-Hur et al., *PLoS Computational Biology*, 2008)



Gaussian kernel

- $\hat{f}$ is sum of Gaussian 'bumps' around each support vector
- to interpret $\hat{f}$, compare relative size of $\|x - z\|_2^2$ and σ^2
- as σ decreases, behavior of kernel becomes more local, leading to greater curvature of decision surface (and potential overfitting)

Spectrum kernel

(Leslie et al., *Biocomputing*, 2001)

- **spectrum kernel**: $\varphi(x)$ is all k-mers (called k-spectrum), so sequences are similar if they contain many of the same k-mers
  - φ maps sequence x over alphabet $\mathcal{A}$ into $|\mathcal{A}|^k$-dimensional space
  - each dimension is $\#$ occurrences of k-mer s in x
- using a suffix tree, can evaluate spectrum kernel in time **linear** in the sequence length rather than exponential $|\mathcal{A}|^k$ time
- can classify a test sequence x^{new} in linear time
  - store hash table mapping k-mers to contributions to w
  - move k-sliding window across x^{new}, look up k-mers in hash, increment classifier value $\hat{f}(x)$ by associated coefficient
- many extensions: weights, add positional/evolutionary information, ...

SVMs and kernel methods

- SVMs are essentially simple linear classifiers, but derive their full power via an elegant extension to the nonlinear setting that implicitly works in high or infinite dimensional feature spaces
- kernels provide an intuitive and flexible modeling toolbox that can be adapted to many different problems, including problems with complex, structured data (strings, sequences, trees, graphs, ...)