Convex Optimization and Dual Problems

Sunghee Yun

Gauss Labs

What will be covered today

- Convex Optimization
  - why convex optimization?
  - optimization problems
  - definition of convex optimization
  - convex optimizations in ML
- Dual Problems
  - Lagrangian and dual function
  - dual problem examples
  - KKT condition
  - optimality condition for support vector machine (SVM) formulation

Why convex optimization?

- many machine learning algorithms (inherently) depend on convex optimization
- quite a few optimization problems can (actually) be solved
- many engineering and scientific problems can be cast into convex optimization problems
- many more can be approximated to convex optimization
- convex optimization sheds lights on understanding intrinsic property and structure of all optimization problems

Mathematical optimization

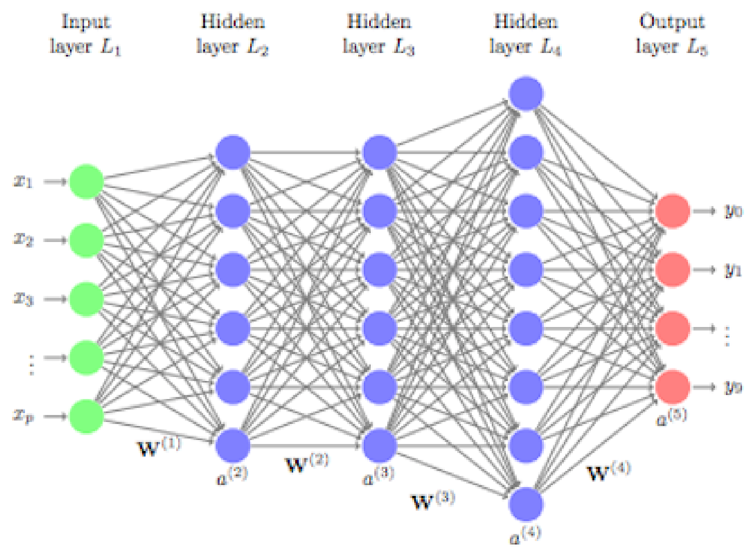
- mathematical optimization problem:

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

  - $x = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T \in \mathbf{R}^n$ is (vector) *optimization variable*
  - $f_0 : \mathbf{R}^n \to \mathbf{R}$ is *objective function*
  - $f_i : \mathbf{R}^n \to \mathbf{R}$ are *inequality constraint functions*
  - $h_i : \mathbf{R}^n \to \mathbf{R}$ are *equality constraint functions*

Optimization problem example

- machine learning
  - optimization variables: model parameters ($e.g.$, neural net weights)
  - objective: loss function / error function
  - constraints: network architecture



Solution methods

- for general optimization problems
  - extremely difficult to solve (practically impossible to solve), $e.g.$, TSP
  - most methods try to find (good) suboptimal solutions, $e.g.$, using heuristics
- some exceptions
  - least-squares (LS)
  - liner programming (LP)
  - semidefinite programming (SDP)

Least-squares (LS)

- least-squares (LS) problem:

$$\text{minimize} \quad \|Ax - b\|_2^2 = \sum_{i=1}^m (a_i^T x - b_i)^2$$

  - analytic solution: any solution satisfying $(A^T A)x^* = A^T b$
  - extremely reliable and efficient algorithms
  - has been there at least since Gauss

- applications
  - LS problems are easy to recognize
  - has huge number of applications, $e.g.$, line fitting

Linear programming (LP)

- linear program (LP):

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

  - no analytic solution
  - reliable and efficient algorithms exist, $e.g.$, simplex method, interiorpoint method
  - has been there at least since Fourier
  - systematical algorithm existed since World War II
- applications
  - less obvious to recognize (than LS)
  - lots of problems can be cast into LP, $e.g.$, network flow problem

Semidefinite programming (SDP)

- semidefinite program (SDP):

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & F_0 + x_1 F_1 + \cdots + x_n F_n \succeq 0
\end{array}
$$

  - no analytic solution
  - but, reliable and efficient algorithms exist, $e.g.$, interior-point method
  - recent technology
- applications
  - never easy to recognize
  - lots of problems, $e.g.$, optimal control theory, can be cast into SDP
  - extremely non-obvious, but convex, hence global optimality easily achieved!

Max-det problem (extension of SDP)

- max-det program:

$$\begin{array}{ll} \text{minimize} & c^T x + \log\det(F_0 + x_1 F_1 + \cdots + x_n F_n) \\ \text{subject to} & G_0 + x_1 G_1 + \cdots + x_n G_n \succeq 0 \\ & F_0 + x_1 F_1 + \cdots + x_n F_n \succ 0 \end{array}$$

  - no analytic solution
  - but, reliable and efficient algorithms exist, $e.g.$, interior-point method
  - recent technology
- applications
  - never easy to recognize
  - lots of stochastic optimization problems, $e.g.$, every covariance matrix is positive semidefinite
  - again convex, hence global optimality (relatively) easily achieved!

Common features in these exceptions?

- they are convex optimization problems!
- convex optimization:

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

where

- $f_0(\lambda x + (1-\lambda)y) \leq \lambda f_0(x) + (1-\lambda) f_0(y)$ for all $x, y \in \mathbf{R}^n$ and $0 \leq \lambda \leq 1$
- $f_i : \mathbf{R}^n \to \mathbf{R}^{k_i}$ are K_i-convex w.r.t. proper cone $K_i \subseteq \mathbf{R}^{k_i}$
- all equality constraints are linear

Convex optimization

- algorithms
  - classical algorithms like simplex method still work well for many LPs
  - many state-of-the-art algorithms develoled for (even) large-scale convex optimization problems
    * barrier methods
    * primal-dual interior-point methods
- applications
  - huge number of engineering and scientific problems are (or can be cast into) convex optimization problems
  - many others can be (approximately) solved using convex relaxation

What's the fuss about convex optimization? Here's why!

- which one of these problems are easier to solve?
  - (generalized) geometric program with $n = 3,000$ variables and $m = 1,000$ constraints

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{p_0} \alpha_{0,i} x_1^{\beta_{0,i,1}} \cdots x_n^{\beta_{0,i,n}} \\ \text{subject to} & \sum_{i=1}^{p_j} \alpha_{j,i} x_1^{\beta_{j,i,1}} \cdots x_n^{\beta_{j,i,n}} \leq 1, \; j = 1, \ldots, m \end{array}$$

    with $\alpha_{j,i} \geq 0$ and $\beta_{j,i,k} \in \mathbf{R}$
    $\Rightarrow$ the *global* optimum can be found within 1 minute using your laptop!
  - minimization of 10th order polynomial of $n = 20$ variables with no constraint

$$\text{minimize} \quad \sum_{i_1=1}^{10} \cdots \sum_{i_n=1}^{10} c_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$$

    with $c_{i_1,\ldots,i_n} \in \mathbf{R}$
    $\Rightarrow$ you *cannot* solve it!

What's the fuss about convex optimization? Here's why!

- which one of these problems are easier to solve?
  - (generalized) geometric program with $n = 3,000$ variables and $m = 1,000$ constraints

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{p_0} \alpha_{0,i} x_1^{\beta_{0,i,1}} \cdots x_n^{\beta_{0,i,n}} \\ \text{subject to} & \sum_{i=1}^{p_j} \alpha_{j,i} x_1^{\beta_{j,i,1}} \cdots x_n^{\beta_{j,i,n}} \leq 1, \ j = 1, \ldots, m \end{array}$$

    with $\alpha_{j,i} \geq 0$ and $\beta_{j,i,k} \in \mathbf{R}$
    $\Rightarrow$ the *global* optimum can be found within 1 minute using your laptop!
  - minimization of 10th order polynomial of $n = 20$ variables with no constraint

$$\text{minimize} \quad \sum_{i_1=1}^{10} \cdots \sum_{i_n=1}^{10} c_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$$

    with $c_{i_1,\ldots,i_n} \in \mathbf{R}$
    $\Rightarrow$ you *cannot* solve it!

What's the fuss about convex optimization? Here's why!

- which one of these problems are easier to solve?
  - (generalized) geometric program with $n = 3,000$ variables and $m = 1,000$ constraints

$$\begin{array}{ll} \mbox{minimize} & \sum_{i=1}^{p_0} \alpha_{0,i} x_1^{\beta_{0,i,1}} \cdots x_n^{\beta_{0,i,n}} \\ \mbox{subject to} & \sum_{i=1}^{p_j} \alpha_{j,i} x_1^{\beta_{j,i,1}} \cdots x_n^{\beta_{j,i,n}} \leq 1, \; j = 1, \ldots, m \end{array}$$

    with $\alpha_{j,i} \geq 0$ and $\beta_{j,i,k} \in \mathbf{R}$
    $\Rightarrow$ the *global* optimum can be found within 1 minute using your laptop!
  - minimization of 10th order polynomial of $n = 20$ variables with no constraint

$$\mbox{minimize} \quad \sum_{i_1=1}^{10} \cdots \sum_{i_n=1}^{10} c_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$$

    with $c_{i_1,\ldots,i_n} \in \mathbf{R}$
    $\Rightarrow$ you *cannot* solve it!

Properties of convex optimization

- convex optimization problems can be solved extremely reliably (and fast)
- a local minimum is a global minimum, which is implied by

$$f(y) \geq f(x) + \nabla f(x)^T (y - x)$$

- nice theoretical property, $e.g.$, self-concordance implies complexity bound (for Newton's method)

$$\frac{f(x_0) - p^*}{\gamma} + \log_2 \log_2(1/\epsilon)$$

- even better pratical performance!
- more on this in future seminars (hopefully)

Convex optimization example in ML: linear regression

- formulation

$$\text{minimize} \quad f(\theta) = \frac{1}{m}\sum_{i=1}^m \left(\theta^T \begin{bmatrix} 1 \\ x^{(i)} \end{bmatrix} - y^{(i)}\right)^2$$

- linear regression is nothing but LS since

$$\begin{aligned} mf(\theta) &= \sum_{i=1}^m \left(\theta^T \begin{bmatrix} 1 \\ x^{(i)} \end{bmatrix} - y^{(i)}\right)^2 = \left\| \begin{bmatrix} 1 & {x^{(1)}}^T \\ \vdots & \vdots \\ 1 & {x^{(m)}}^T \end{bmatrix} \theta - \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(m)} \end{bmatrix} \right\|_2^2 \\ &= \|X\theta - y\|_2^2 \end{aligned}$$

- convex in θ, hence obtains its global optimality when the gradient vanishes, *i.e.*,

$$m\nabla f(\theta) = 2X^T(X\theta - y) = 2((X^TX)\theta - X^Ty) = 0$$

Convex optimization example in ML: ridge regression

- Ridge regression solves the following problem: (for some $\lambda > 0$)

$$\text{minimize} \quad f_0(x) = \|Ax - y\|_2^2 + \lambda \|x\|_2^2$$

  - regularization, *e.g.*, to preventing overfitting
- can be extended to (without sacraficing solvability!)

$$\begin{array}{ll} \text{minimize} & f_0(x) = \|Ax - y\|_2^2 + \lambda \|x\|_2^2 = \left\| \begin{bmatrix} A \\ \sqrt{\lambda} I \end{bmatrix} x - \begin{bmatrix} y \\ 0 \end{bmatrix} \right\|_2^2 \\ \text{subject to} & f_i(x) \leq 0, \ i = 1, \ldots, m \\ & h_i(x) = 0, \ i = 1, \ldots, p \end{array}$$

- can be incorporated into gradient descent algorithm, *e.g.*,

$$\nabla f(x) = 2A^T(Ax - y) + 2\lambda x$$

Convex optimization example in ML: lasso

- (lasso stands for least absolute shrinkage & selection operator)
- lasso solves (a problem equivalent to) the following problem:

$$\text{minimize} \quad f_0(x) = \|Ax - y\|^2 + \lambda \|x\|_1$$

  - 1-norm penalty term for parameter selection

- objective funtion *not* smooth.
- however, simple trick would solve this problem (with additional convex inequality constraints and affine equality constraints)

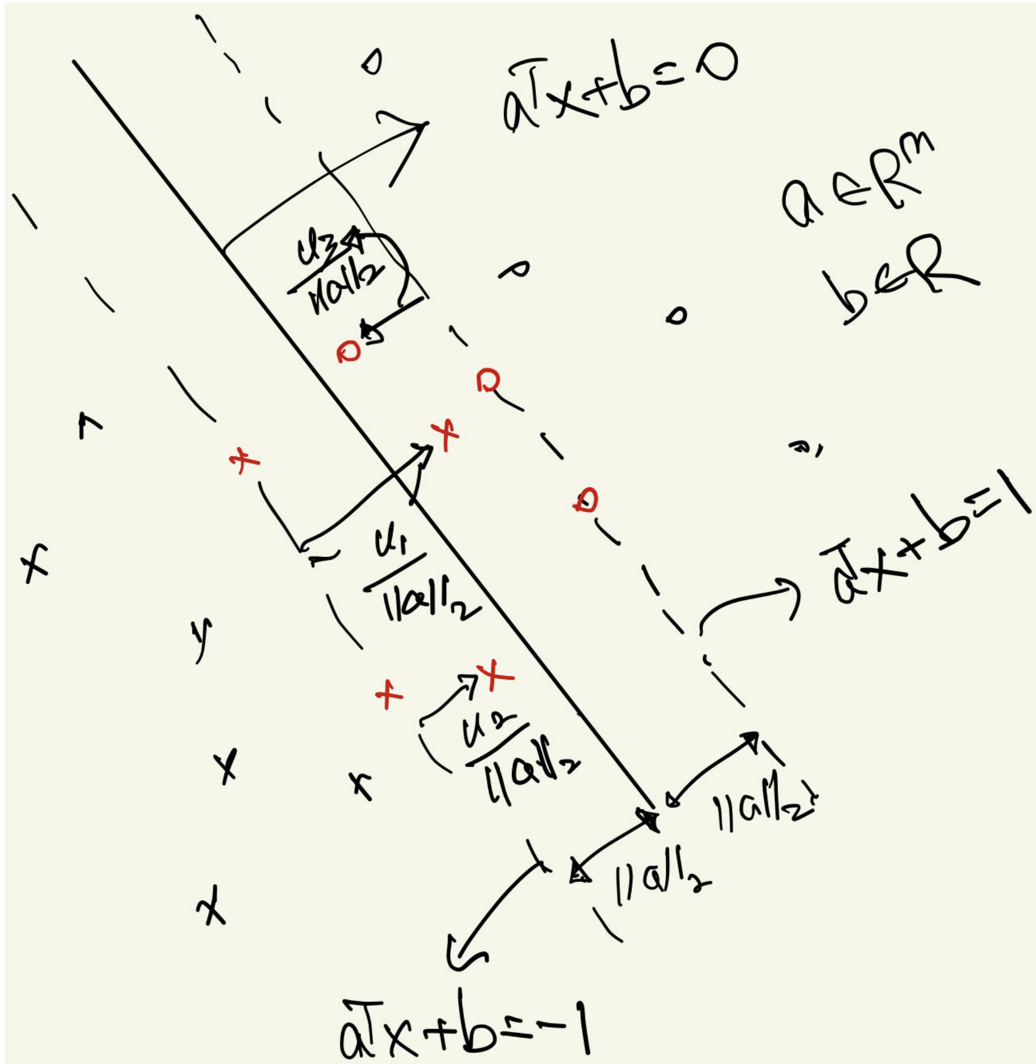
$$\begin{array}{ll} \text{minimize} & f_0(x) = \|Ax - y\|^2 + \lambda \sum_{i=1}^n z_i \\ \text{subject to} & -z_i \leq x_i \leq z_i, \ i = 1, \ldots, n \end{array}$$

Convex optimization example in ML: SVM

- problem definition:
  - given $x^{(i)} \in \mathbf{R}^p$: input data, and $y^{(i)} \in \{-1, 1\}$: output labels
  - find hyperplane which separates two different classes as distinctively as possible (in some measure)
- (typical) formulation:

$$
\begin{array}{ll}
\mbox{minimize} & \|a\|_2^2 + \gamma \sum_{i=1}^m u_i \\
\mbox{subject to} & y^{(i)}(a^T x^{(i)} + b) \geq 1 - u_i, \ i = 1, \ldots, m \\
& u \geq 0
\end{array}
$$

  - convex optimization problem with optimization variables, $a \in \mathbf{R}^p$, $b \in \mathbf{R}$, and $u \in \mathbf{R}^m$
  - hence stable and efficient algorithms exist even for very large problems
  - has worked extremely well in practice

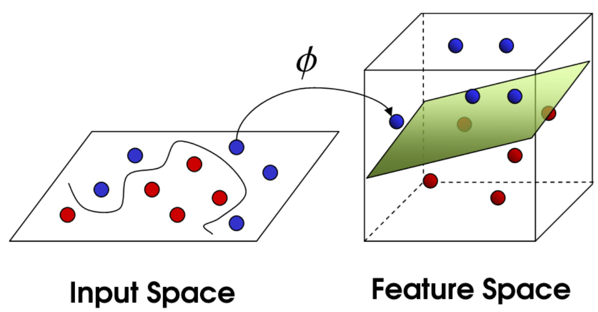

aTx+b=0
a∈R^n
b∈R
aTx+b=1
aTx+b=-1

Support vector machine with kernels

- use feature transformation $\phi : \mathbf{R}^p \to \mathbf{R}^q$ (with $q > p$)
- formulation:

$$\begin{array}{ll} \text{minimize} & \|\tilde{a}\|_2^2 + \gamma \sum_{i=1}^m \tilde{u}_i \\ \text{subject to} & y^{(i)}(\tilde{a}^T \phi(x^{(i)}) + \tilde{b}) \geq 1 - \tilde{u}_i, \ i = 1, \ldots, m \\ & \tilde{u} \geq 0 \end{array}$$

- still convex optimization problem with optimization variables, $\tilde{a} \in \mathbf{R}^q$, $\tilde{b} \in \mathbf{R}$, and $\tilde{u} \in \mathbf{R}^m$



Duality

- every (constrained) optimization problem has a *dual problem* (whether or not it is a convex optimization problem)
- every dual problem is a *convex optimization problem* (whether or not the primal problem is a convex optimization problem)
- duality provides *optimality certificate*, hence plays *central role* for modern optimization and machine learning algorithm implementation
- (usually) solving one readily solves the other!

Lagrangian

- standard form problem:

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

  where $x \in \mathbf{R}^n$ is optimization variable, $\mathcal{D}$ is domain, p^* is optimal value

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

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

  - λ_i: Lagrange multiplier associated with $f_i(x) \leq 0$
  - ν_i: Lagrange multiplier associated with $h_i(x) = 0$

Lagrange dual function

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

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

  - g is *always* concave
  - $g(\lambda, \nu)$ can be $-\infty$
- lower bound property: if $\lambda \geq 0$, then $g(\lambda, \nu) \leq p^*$

  Proof: If $\tilde{x}$ is feasible and $\lambda \geq 0$, then $f_0(\tilde{x}) \geq L(\tilde{x}, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) = g(\lambda, \nu)$. Thus,

$$p^* = \inf_{x \in \mathcal{F}} f_0(x) \geq g(\lambda, \nu)$$

  where $\mathcal{F} = \{x \mid f_i(x) \leq 0 \text{ for } 1 \leq i \leq m,\ h_j(x) = 0 \text{ for } 1 \leq j \leq p\}$.

Dual problem

- Lagrange dual problem:

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

  - is a convex optimization problem
  - provides a lower bound on p^*

- let d^* denote the optimal value for the dual problem
  - weak duality: $d^* \leq p^*$
  - strong duality: $d^* = p^*$

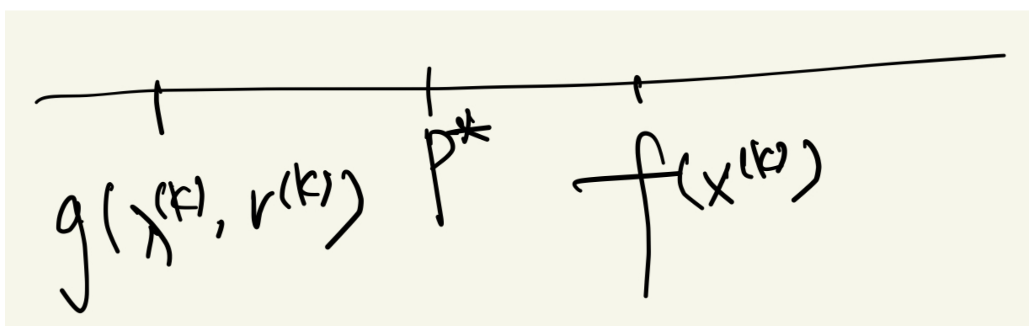
Dual problem provides optimality certificate

- many algorithms solves the dual problem simultaneously
  - Lagrangian dual variables obtained with no additional cost
- if iterative algorithm generates feasible solution sequence,

$$(x^{(1)}, \lambda^{(1)}, \nu^{(1)}) \to (x^{(2)}, \lambda^{(2)}, \nu^{(2)}) \to (x^{(3)}, \lambda^{(3)}, \nu^{(3)}) \to \cdots$$

then, we have an *optimality certificate*:

$$f(x^{(k)}) - p^* \leq f(x^{(k)}) - g(\lambda^{(k)}, \nu^{(k)})$$



Weak duality

- weak duality implies $d^* \leq p^*$
  - always true (by construction of dual problem)
  - provides *nontrivial* lower bounds, especially, for difficult problems, $e.g.$, solving the following SDP:

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0 \end{array}$$

gives a lower bound for (NP-hard) max-cut problem (maximizing total weight of edges between a subset of vertices and its complement)

$$\begin{array}{ll} \text{minimize} & x^T W x \\ \text{subject to} & x_i^2 = 1, \ i = 1, \ldots, n \end{array}$$

Derivation of dual problem of max-cut problem

- Lagrangian

$$L(x,\nu) = x^T W x + \sum_{i=1}^n \nu_i (x_i^2 - 1) = x^T (W + \mathbf{diag}(\nu))x - \mathbf{1}^T \nu$$

- dual function

$$g(\nu) = \inf_{x \in \mathbf{R}^n} L(x,\nu) = \begin{cases} -\mathbf{1}^T\nu & \text{if } W + \mathbf{diag}(\nu) \succeq 0 \\ -\infty & \text{otherwise} \end{cases}$$

because $x^T(W + \mathbf{diag}(\nu))x$ is unbounded below if $W + \mathbf{diag}(\nu) \not\succeq 0$

- hence, the dual problem

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0 \end{array}$$

where the optimization variable is $\nu \in \mathbf{R}^n$

Dual of the dual of max-cut problem

- let the dual of the max-cut problem be our primal problem here

- primal problem

$$\begin{array}{ll} \text{maximize} & -\mathbf{1}^T \nu \\ \text{subject to} & W + \mathbf{diag}(\nu) \succeq 0 \end{array}$$

- Lagrangian

$$L(\nu, X) = -\mathbf{1}^T \nu + \mathbf{Tr}\, X(W + \mathbf{diag}(\nu)) = \sum_{i=1}^{n} \nu_i (X_{ii} - 1) + \mathbf{Tr}\, XW$$

- dual function

$$g(X) = \sup_{\nu \in \mathbf{R}^n} L(\nu, X) = \begin{cases} \mathbf{Tr}\, XW & \text{if } X_{ii} = 1 \text{ for } i = 1, \ldots, n \\ \infty & \text{otherwise} \end{cases}$$

- hence, the dual problem

$$\begin{array}{ll} \text{minimize} & \mathbf{Tr}\, XW \\ \text{subject to} & X_{ii} = 1 \text{ for } i = 1, \ldots, n \end{array}$$

Dual of dual is convex relaxation of the original problem

- now add rank one constraint *i.e.*,

$$
\begin{array}{ll}
\text{minimize} & \mathbf{Tr}\, XW \\
\text{subject to} & X_{ii} = 1 \text{ for } i = 1, \ldots, n \\
& \mathbf{rank}(X) = 1
\end{array}
$$

then this is equivalent to the original max-cut problem because

$$\mathbf{rank}(X) = 1 \Leftrightarrow X = xx^T \text{ for some } x \in \mathbf{R}^n$$

then

$$\mathbf{Tr}\, XW = \mathbf{Tr}\, xx^T W = \mathbf{Tr}\, x^T W x = x^T W x$$

and

$$X_{ii} = 1 \Leftrightarrow x_i^2 = 1$$

- thus it is the convex relaxation of the original problem

- hence, if d^{**} is the optimal value of the dual of the dual, we have

$$d^* = d^{**} \leq p^*$$

because the dual problem is strictly feasible, $i.e.$, satisfies Slater's condition (later)

Strong duality

- strong duality implies $d^* = p^*$
  - not necessarily hold; does not hold in general
  - *usually* holds for convex optimization problems
  - conditions which guarantee strong duality in convex problems called *constraint qualifications*
  - example of constraint qualifications: Slater's condition

Duality example: LP

- primal problem:

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

- Lagrangian:

$$L(x,\lambda) = c^T x + \lambda^T (Ax - b) = (c + A^T\lambda)^T x - b^T\lambda$$

- dual function:

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

- dual problem:

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

  - Slater's condition implies that $p^* = d^*$ if $A\tilde{x} \prec b$ for some $\tilde{x}$

  - truth is, $p^* = d^*$ except when both primal and dual are infeasible

Duality example: QP

- primal problem (assuming $P \in \mathbf{S}_{++}^n$):

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

- Lagrangian:

$$L(x, \lambda) = x^T P x + \lambda^T (Ax - b)$$

- gradient of Lagrangian with respect to x:

$$\nabla_x L(x, \lambda) = 2Px + A^T \lambda$$

- dual function:

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

- dual problem:

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

  - Slater's condition implies that $p^* = d^*$ if $A\tilde{x} \prec b$ for some $\tilde{x}$

  - truth is, $p^* = d^*$ always!

Fun demo for duality

Karush-Kuhn-Tucker (KKT) conditions

- KKT (optimality) conditions consist of
  - primal feasibility: $f_i(x) \leq 0$ for all $1 \leq i \leq m$, $h_i(x) = 0$ for all $1 \leq i \leq p$
  - dual feasibility: $\lambda \succeq 0$
  - complementary slackness: $\lambda_i f_i(x) = 0$
  - zero gradient of Lagrangian: $\nabla f_0(x) + \sum_{i=1}^{m} \lambda_i \nabla f_i(x) + \sum_{i=1}^{p} \nu_i \nabla h_i(x) = 0$
- if strong daulity holds and x^*, λ^*, and ν^* are optimal, they satisfy KKT condtions!

Proof

- assume strong dualtiy holds, x^* is primal optimal, and (λ^*, ν^*) is dual optimal

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

- *complementary slackness* holds because

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

$$\Rightarrow \quad \sum_{i=1}^m \lambda_i^* f_i(x^*) = 0$$

$$\Rightarrow \quad \lambda_i^* f_i(x^*) = 0 \text{ for all } i = 1, \ldots, m$$

- *complementary slackness* implies

$$\lambda_i^* > 0 \Rightarrow f_i(x^*) = 0, \qquad f_i(x^*) < 0 \Rightarrow \lambda_i^* = 0$$

- we call those inequalities $f_i(x) \leq 0$ with $\lambda_i > 0$ *active constraints*

- zero gradient of Lagrangian because

$$
\begin{aligned}
& \inf_x L(x, \lambda^*, \nu^*) = L(x^*, \lambda^*, \nu^*) \\
\Rightarrow \quad & x^* \text{ minimizes } L(x, \lambda^*, \nu^*) \\
\Rightarrow \quad & \nabla f_0(x) + \sum_{i=1}^m \lambda_i \nabla f_i(x) + \sum_{i=1}^p \nu_i \nabla h_i(x) = 0
\end{aligned}
$$

  - thus, x^* minimizes $L(x, \lambda^*, \nu^*)$
  - hence (if f_i and h_i are differentiable)

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

KKT conditions for convex optimization problem

- if $\tilde{x}$, $\tilde{\lambda}$, and $\tilde{\nu}$ satisfy KKT for convex optimization problem, then they are optimal!
  - complementary slackness implies $f_0(\tilde{x}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$
  - zero gradient of Lagrangian together with convexity implies $g(\tilde{\lambda}, \tilde{\nu}) = L(\tilde{x}, \tilde{\lambda}, \tilde{\nu})$
- for example, if Slater's condition is satisfied for a convex optimization problem,
  - x is optimal if and only if there exist λ, ν that satisfy KKT conditions
- this generalizes optimality condition $\nabla f_0(x) = 0$ for unconstrained problem

Dual problem for SVM problem

- note

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|a\|_2^2 + \gamma \sum_{i=1}^m u_i \\ \text{subject to} & y^{(i)}(a^T x^{(i)} + b) \geq 1 - u_i,\ i = 1, \ldots, m \\ & u \succeq 0 \end{array}$$

- Lagrangian

$$\begin{aligned} L(a, b, u, \lambda, \nu) \\ &= \frac{1}{2}\|a\|_2^2 + \gamma \sum_{i=1}^m u_i + \sum_{i=1}^m \lambda_i (1 - u_i - y^{(i)}(a^T x^{(i)} + b)) + \sum_{i=1}^m \nu_i(-u_i) \\ &= \frac{1}{2}\|a\|_2^2 - \left(\sum_{i=1}^m \lambda_i y^{(i)} x^{(i)}\right)^T a - b \sum_{i=1}^m \lambda_i y^{(i)} + \sum_{i=1}^m u_i(\gamma - \lambda_i - \nu_i) + \sum_{i=1}^m \lambda_i \end{aligned}$$

- dual function

$$g(\lambda, \nu) = \begin{cases} -\frac{1}{2} \left\| \sum_{i=1}^m \lambda_i y^{(i)} x^{(i)} \right\|_2^2 + \sum_{i=1}^m \lambda_i & \text{if } \sum_{i=1}^m \lambda_i y^{(i)} = 0, \lambda_i + \nu_i = \gamma \\ -\infty & \text{otherwise} \end{cases}$$

- dual problem

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^m \lambda_i - \frac{1}{2} \left\| \sum_{i=1}^m \lambda_i y^{(i)} x^{(i)} \right\|_2^2 \\ \text{subject to} & \sum_{i=1}^m \lambda_i y^{(i)} = 0 \\ & \lambda_i + \nu_i = \gamma \text{ for } i = 1, \ldots, m \end{array}$$

- or equivalently,

$$\begin{array}{ll} \text{maximize} & \sum_{i=1}^m \lambda_i - \frac{1}{2} \lambda^T P \lambda \\ \text{subject to} & \sum_{i=1}^m \lambda_i y^{(i)} = 0 \\ & \lambda_i + \nu_i = \gamma \text{ for } i = 1, \ldots, m \end{array}$$

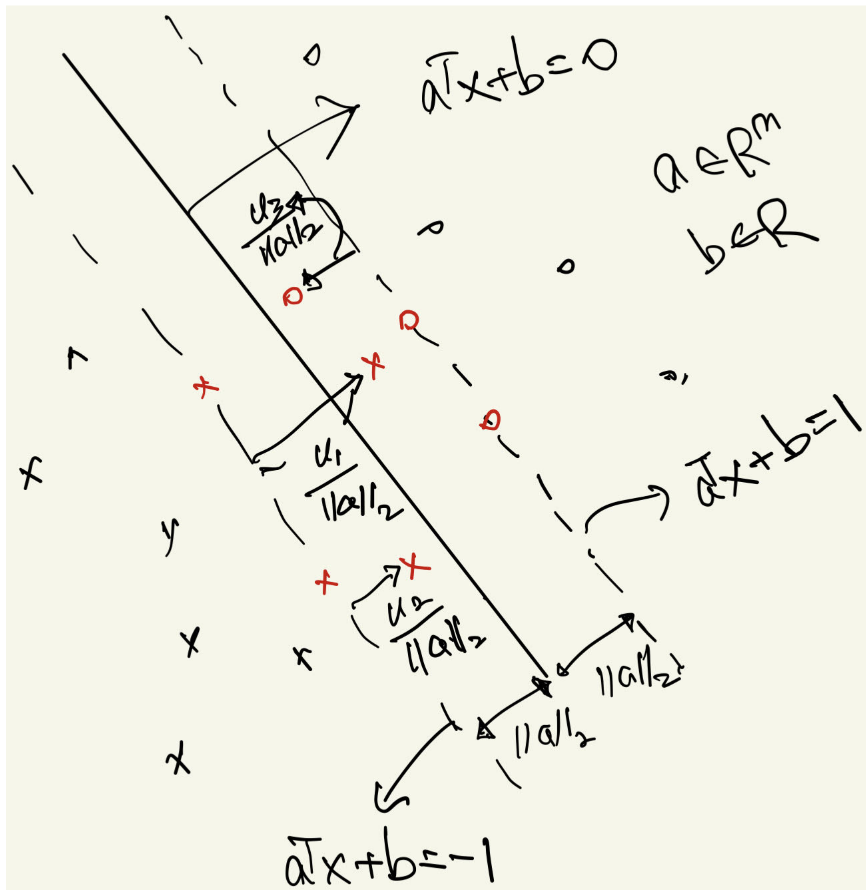
where $P = X^T X \succeq 0$ and $X = \begin{bmatrix} y^{(1)} x^{(1)} & \cdots & y^{(m)} x^{(m)} \end{bmatrix} \in \mathbf{R}^{n \times m}$

- dual problem is *quadratic program*

KKT conditions for SVM problem

- assume that a^*, b^*, u^* are primal optimal and λ^* and ν^* are dual optimal, then KKT conditions imply
  - $y^{(i)}({a^*}^T x^{(i)} + b^*) \geq 1 - u_i^*$ for $i = 1, \ldots, m$
  - $u_i^* \geq 0, \lambda_i^* \geq 0, \nu_i^* \geq 0, \lambda_i^* + \nu_i^* = \gamma$ for $i = 1, \ldots, m$
  - $\nu_i^* u_i^* = 0$ for $i = 1, \ldots, m$
  - $\lambda_i^*(1 - u_i^* - y^{(i)}({a^*}^T x^{(i)} + b^*)) = 0$ for $i = 1, \ldots, m$
  - $\sum_{i=1}^m \lambda_i^* y^{(i)} = 0$
  - $a^* = \sum_{i=1}^m \lambda_i^* y^{(i)} x^{(i)}$
- $x^{(i)}$ with $\lambda_i^* > 0$ are called *support vectors*!
  - those with positive slacks ($u_i^* > 0$), $\lambda_i^* = \gamma$
  - those on the edge ($u_i^* = 0$), $0 < \lambda_i^* \leq \gamma$
- then the boundary can be characterized by $\sum_{i=1}^m \lambda_i^* y^{(i)} {x^{(i)}}^T x + b^*$
  - with kernel, the boundary is $\sum_{i=1}^m \lambda_i^* y^{(i)} K(x, x^{(i)}) + b^*$

SVM figure



Next time

- we can discuss
  - sensitivity analysis using Lagrange dual variables
  - various interpretations for dual problems and dual variables
  - some algorithms for convex optimization, *e.g.*, gradient descent, Newton's method, *etc.*
  - their convergence analysis
  - various applications in approximation, fitting, statistical estimation, geometric problems, *etc.*

Thank you!