(cont. from last time) Composition rules that preserve the convexity of functions

1. Affine composition

If f is convex, then $g(x) = f(Ax + b)$ is convex

2. General composition

Suppose $f = h \circ g$, *i.e.* $f(x) = h(g(x))$, $g: \mathfrak{R}^n \to \mathfrak{R}$, $h: \mathfrak{R} \to \mathfrak{R}$, $f: \mathfrak{R}^n \to \mathfrak{R}$.
Then

a. f is convex if h is convex and nondecreasing, g is convex

b. f is convex if h is convex and nonincreasing, g is concave

How to remember this? for n=1, find f''(x)

$f(x) = h(g(x))$

$When\ n = 1$

$f'(x) = h'(g(x))\, g'(x)$

$f''(x) = h''(g(x))(g'(x))^2 + h'(g(x))g''(x)$

$If\ g\ is\ convex,\ g''(x)\ is\ positive$

$if\ h\ is\ convex,\ h''(g(x))\ is\ positive$

3. Vector composition

something similar...

Example: log-sum-exp function

Musings from later

log(exp(a)) + log(exp(b)) = log(exp(a).exp(b)) = log(exp(a + b)) = log(sum{i=1 to 1}exp(x)) for x = a+b

sum of log exponentials = log sum of exponentials, where number of summed items = 1

Log-sum-exp function $g(x) = \log\left(\sum_{i=1}^{k} \exp\left\{a_i^T x_i + b_i\right\}\right)$. This is also known as softmax because it smoothly approximates $max_{i=1,..,k}\left(a_i^T + b_i\right)$.

Fact 1: $g(x)$ is a convex function.

Proof: First, note that is suffices to prove convexity of $f(x) = \log\left(\sum_{i=1}^{n} \exp\{x_i\}\right)$. Then $g(x)$ is an affine composition of $f(x)$.

Now use second-order characterization.

$$\nabla_i f(x) = \frac{e^{x_i}}{\sum e^{x_l}}$$

$$\left(\textit{Remember } dy/dx = \frac{v\,du - u\,dv}{v^2}. \textit{ And the hessian is the diff of } \nabla f(x_i) \textit{ wrt other indices of the vecto} \right.$$

then Hessian $\nabla^2_{ij} f(x) = \frac{e^{x_i}}{\sum e^{x_l}} 1\{i = j\} - \frac{e^{x_i} e^{x_j}}{\left(\sum e^{x_l}\right)^2}$

A note on diagonal dominance: A square matrix is diagonally dominain if

$$|a_{ii}| \geq \sum_{j \neq i} |a_{ij}| \; for\; all\; i$$

A Hermitian matrix is a complex square matrix such that the matrix is its own complex conjugate.
Every Hermitian diagonally dominant matrix with real non-negative diagonal entries is positive semidefinite.

Then, we can say that $\nabla^2_{ij} f(x) = diag(z) - zz^T$, where $z = \frac{e^{x_i}}{\sum e^{x_l}}$, is diagonally dominant, and is hence psd.
Using the second order condition, we know that a function that has always has psd second derivative and convex domain is convex - so $f(x)$ is convex, and there $g(x)$ is convex as an affine composition.

Recap of what we have seen so far

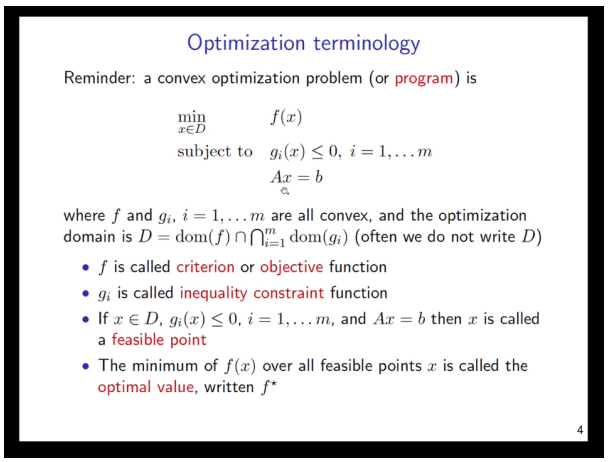
'Convex calculus' makes it easy to check convexity. Tools:
1. Definiition on convex sets and functions
2. Key properties (e.g. first and second-order characterizations for functions)
3. Operations that preserve convexity (e.g. affine composition)

Optimization basics

Today:

- terminonology
- first-order optimality
- equivalent translations

Optimization terminology



Q: Why can you only have linear *equality* constraints in a convex optimization problem? Why not any generic convex function and set it to zero? e.g. why not a quadratic constraint = 0?
A. All the constraints in the problem need to be convex. An equality constraint of the form $h(x) = c$ is equivalent to saying $h(x) \leq c$ AND $-h(x) \leq -c$. But the only form of the function $h(x)$ such that both $h(x)$ and $-h(x)$ are convex, or in other words that $h(x)$ is both convex and concave (since that implies that $-h(x)$ is convex) is an affine function!

Optimization problems do not need to have solutions!
Example: $f(x) = e^{-x}$.
$min\ f(x)$ has no solution

- these problems only have solutions under certain conditions - see Rockefeller book for more

WE will assume here that the problems we are dealing with have solutions.

If x is feasible and $f(x) = f^*$, then x is called optimal, also called a solution
If x is feasible and $f(x) \leq f^* + \epsilon$, then x is called $\epsilon - suboptimal$
if x is feasible and $g_i(x) = 0$, then we say that g_i is active at x (strictly equal to zero, not less than)
Convex minimization ≡ concave maximization.

A. Properties of convex optimization

1. Solution set is convex
Let X_{opt} be the set of all solutions of a convex problem, written $X_{opt} = argmin\ f(x)$
Then X_{opt} is a convex set!
Proof

- idea: you need to show that for any solutions x, y in the set X_{opt}, $tx + (1-t)y$ is also in X_{opt}
- So work backwards from here!
- Assume $z = tx + (1-t)y$
- Then
  - $g_i(z) < 0$
    - $g_i(z) = g_i(tx + (1-t)y) \leq tg_i(x) + (1-t)g_i(y)$ because g_i is a convex function
    - $tg_i(x) + (1-t)g_i(y) < 0$ because $g_i(x) < 0$ and $g_i(y) < 0$, as x, y are solutions to the problem!
    - so $g_i(z) < 0$!
  - $A(z) = b$
    - work this one out, same way
  - $f(z) = f^*$
    - same way
- Therefore z is also a solution and in the set!

2. If f is strictly convex, then the solution is unique!

- Proof: by contradiction. Assume, as above, two solutions x, y. Then $z = tx + (1-t)y$ must also be a solution. But $f(z) = f(tx + (1-t)y) < tf(x) + (1-t)f(y) < f^*$,(strictly less than, not equal) which contradicts the statement that these are all solutions
- So there must only be one solution
- Same idea as example in lecture 1
  - assume two points - means that all points on line segment must satisfy criterion - but this leads to some contradcition - meaning that two points could not exist in the first place.

Example: Lasso

Constrained form of the lasso

- this is equavalent to the penalized form
- the KKT conditions will show this

$min_\beta \, ||y - XB||_2^2 \; subject \; to \; ||\beta||_1 \leq s$

- gives a sparse estimate of regression coefficients

Questions

1. Is the criterion function convex?

A: Yes. Least squares loss is convex - it is quadratic and the form it takes yields Q that is psd
Another way to answer this: This is a sum of squares, which is convex, and this is an affine transformation of β.

- Note: I want to think through this properly

2. Is this a convex optimization problem?
A: Yes.
- The criteration function is convex (as shown above).
  - No equality constraint
  - One inequality constraint: $g(\beta) \leq s$

3. What is the feasible set?
4. Is the solution unique?
Case 1: If $n > p$ and X has full column rank? (i.e. X has linearly independent columns)

- The only condition we know for uniqueness is strict convexity
- $f(\beta) = ||y - XB||_2^2 = \beta^T X^T XB - 2y^t XB + y^T y$
- We need to check if this is strictly convex!
- This means, from the second-order condition, that the Hessian is strictly positive definite

Aside: I am trying to understand matrix diffentiation rules

$$f_1(B) = B^T X^T XB = \left[\sum_{k=1}^{p} \sum_{i=1}^{n} \sum_{j=1}^{p} B_k X_{ik} X_{ij} B_j \right]$$

$Say\ p = 2,\ k = 1,\ n = 2.\ Then$

$$f_1(B_1) = B_1 X_{11} X_{11} B_1 + B_1 X_{11} X_{12} B_2 + B_1 X_{21} X_{21} B_1 + B_1 X_{21} X_{22} B_2$$
$$+ B_2 X_{12} X_{11} B_1 + B_2 X_{12} X_{12} B_2 + B_2 X_{22} X_{21} B_1 + B_2 X_{22} X_{22} B_2$$
$$f_1'(B_1) = 2X_{11} X_{11} B_1 + X_{11} X_{12} B_2 + 2X_{21} X_{21} B_1 + X_{21} X_{22} B_2 + B_2 X_{12} X_{11}$$
$$+ B_2 X_{22} X_{21}$$
$$= 2X_{11} X_{11} B_1 + 2X_{11} X_{12} B_2 + 2X_{21} X_{21} B_1 + 2X_{21} X_{22} B_2$$
$$= 2X_k^T XB$$
$$Then,\ \nabla f(B) = 2X^T XB$$
$$and\ \nabla^2 f(B) = 2X^T X$$

This is strictly positive definite when $X^T X$ is invertible, which is true when X has linearly independent columns, which is true!
So yes, the solution is unique in case 1.

Case 2: if $p > n$ (high dimensional case)?
In this case, X cannot have more than n linearly indepenent columns, so it cannot have full rank. This means X will have a null space, ie there is some non-zero vector a st $Xa = 0$. Which means one of the eigenvalues fo X is 0, which means you cannot invert the problem.

So no, the solution is not unique.
BUT, the KKT conditions suggest that in most forms of this problem, the solution will indeed be unique with certainty approaching 1?

Q: How do the answers change if you changed the criterion from squared error to Huber loss?

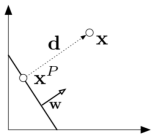
Huber loss = $\sum \rho\left(y_i - x_i^T B\right),\ \rho(z) = \left\{ z^2/z\ if\ |z| \le \delta,\ \delta|z| - \delta^2/2\ else \right\}$
This loss function is quadratic in the input near zero but smooths into a linear function in Z outside the interval δ

- Try thinking through this!

Example 2: SVM

- SVMs do classification in a potentially non linearly separable setting
- y = +1 or -1
- When the two classes are not separated by any linear decision boundary in feature space, the SVM is an optimization that defines a decision boundary
  - defines a hyperplane through the feature space that attempts to maximise the margin between the two classes subject to allowing some overlap
  - the hyperplane you fit is $x^T B + B_0 = 0$ - this is the affine space
  - the ξ_i variables are slack vars
    * non zero when a point is on the wrong side of the boundary
- https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote09.html
  - Typically, if a data set is linearly separable, there are infinitely many separating hyperplanes. A natural question to ask is: Question: What is the best separating hyperplane?
  - SVM Answer: The one that maximizes the distance to the closest data points from both classes. We say it is the hyperplane with maximum margin.
  - We already saw the definition of a margin in the context of the Perceptron. A hyperplane is defined through w,b as a set of points such that H={x|wTx+b=0}. Let the margin γ be defined as the distance from the hyperplane to the closest point across both classes.
  -

What is the distance of a point $\mathbf{x}$ to the hyperplane $\mathcal{H}$?
Consider some point $\mathbf{x}$. Let $\mathbf{d}$ be the vector from $\mathcal{H}$ to $\mathbf{x}$ of minimum length. Let $\mathbf{x}^P$ be the projection of $\mathbf{x}$ onto $\mathcal{H}$. It follows then that:



- $\mathbf{x}^P = \mathbf{x} - \mathbf{d}$.
- $\mathbf{d}$ is parallel to $\mathbf{w}$, so $\mathbf{d} = \alpha\mathbf{w}$ for some $\alpha \in \mathbb{R}$.
- $\mathbf{x}^P \in \mathcal{H}$ which implies $\mathbf{w}^T\mathbf{x}^P + b = 0$
- therefore $\mathbf{w}^T\mathbf{x}^P + b = \mathbf{w}^T(\mathbf{x} - \mathbf{d}) + b = \mathbf{w}^T(\mathbf{x} - \alpha\mathbf{w}) + b = 0$
- which implies $\alpha = \frac{\mathbf{w}^T\mathbf{x}+b}{\mathbf{w}^T\mathbf{w}}$

The length of $\mathbf{d}$:

$$\|\mathbf{d}\|_2 = \sqrt{\mathbf{d}^T\mathbf{d}} = \sqrt{\alpha^2\mathbf{w}^T\mathbf{w}} = \frac{|\mathbf{w}^T\mathbf{x}+b|}{\sqrt{\mathbf{w}^T\mathbf{w}}} = \frac{|\mathbf{w}^T\mathbf{x}+b|}{\|\mathbf{w}\|_2}$$

Margin of $\mathcal{H}$ with respect to D: $\gamma(\mathbf{w}, b) = \min_{\mathbf{x}\in D} \frac{|\mathbf{w}^T\mathbf{x}+b|}{\|\mathbf{w}\|_2}$

By definition, the margin and hyperplane are scale invariant: $\gamma(\beta\mathbf{w}, \beta b) = \gamma(\mathbf{w}, b), \forall\beta \neq 0$

Note that if the hyperplane is such that γ is maximized, it must lie right in the middle of the two classes. In other words, γ must be the distance to the closest point within both classes. (If not, you could move the hyperplane towards data points of the class that is further away and increase γ, which contradicts that γ is maximized.)

-

Max Margin Classifier

We can formulate our search for the maximum margin separating hyperplane as a constrained optimization problem. The objective is to maximize the margin under the constraints that all data points must lie on the correct side of the hyperplane:

$$\underbrace{\max_{\mathbf{w},b} \gamma(\mathbf{w}, b)}_{\text{maximize margin}} \text{ such that } \underbrace{\forall i\ y_i(\mathbf{w}^T x_i + b) \geq 0}_{\text{separating hyperplane}}$$

If we plug in the definition of γ we obtain:

$$\underbrace{\max_{\mathbf{w},b} \underbrace{\frac{1}{\|\mathbf{w}\|_2} \min_{\mathbf{x}_i \in D} |\mathbf{w}^T\mathbf{x}_i + b|}_{\gamma(\mathbf{w},b)}}_{\text{maximize margin}} \quad s.t. \quad \underbrace{\forall i\ y_i(\mathbf{w}^T x_i + b) \geq 0}_{\text{separating hyperplane}}$$

Because the hyperplane is scale invariant, we can fix the scale of $\mathbf{w}, b$ anyway we want. Let's be clever about it, and choose it such that

$$\min_{\mathbf{x}\in D} |\mathbf{w}^T\mathbf{x} + b| = 1.$$

We can add this re-scaling as an equality constraint. Then our objective becomes:

$$\max_{\mathbf{w},b} \frac{1}{\|\mathbf{w}\|_2} \cdot 1 = \min_{\mathbf{w},b} \|\mathbf{w}\|_2 = \min_{\mathbf{w},b} \mathbf{w}^\top\mathbf{w}$$

(Where we made use of the fact $f(z) = z^2$ is a monotonically increasing function for $z \geq 0$ and $\|\mathbf{w}\| \geq 0$; i.e. the $\mathbf{w}$ that maximizes $\|\mathbf{w}\|_2$ also maximizes $\mathbf{w}^\top\mathbf{w}$.)

The new optimization problem becomes:

$$\begin{aligned} &\min_{\mathbf{w},b} \mathbf{w}^\top\mathbf{w} \\ \text{s.t.} \quad &\begin{aligned} \forall i,\ y_i(\mathbf{w}^T\mathbf{x}_i + b) &\geq 0 \\ \min_i |\mathbf{w}^T\mathbf{x}_i + b| &= 1 \end{aligned} \end{aligned}$$

These constraints are still hard to deal with, however luckily we can show that (for the optimal solution) they are equivalent to a much simpler formulation. (Makes sure you know how to prove that the two sets of constraints are equivalent.)

$$\begin{aligned} &\min_{\mathbf{w},b} \mathbf{w}^T\mathbf{w} \\ \text{s.t.} \quad &\forall i\ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \end{aligned}$$

The problem is

$$min_{B,\,B_0,\,\xi_i} \frac{1}{2}||B||_2^2 \;+\; C\sum_{i=1}^{n} \xi_i,\; subject\ to$$

$$\xi_i \;\geqslant\; 0,\; i \;=\; 1,\; \ldots,\; n$$

$$y_i\left(x_i^T B \;+\; B_0\right) \;\geqslant\; 1 \;-\; \xi_i,\; i \;=\; 1,\; \ldots,\; n$$

The criterion tries to minimize the margin between the two classes (? not maximise)? + penalize the overlap

Is the criterion convex?

Yes. Norm squared, and linear function

Are the constraints convex?

The constraint $\xi_i \geqslant 0$ is equivalent to saying $-\xi_i \leqslant 0$, so this is convex in ξ and takes the form we need for the problem i.e. $g_i(x) \leqslant 0$

Linear functions can be written as something convex leq zero - which makes them convex in form?

- As they are both convex and concave, they can be written to fit this form

Aside:

f(x) = x is convex iff

f(tx + (1-t)y) = tx + (1-t)y $\leqslant$ tf(x) + (1-t)f(y) = tx + (1-t)y, which is true

f(x) = -x is convex iff

f(tx + (1-t)y) = -(tx + (1-t)y) = -tx - (1-t)y $\leqslant$ -tx - (1-t)y, which is true

f(x) = x^2 is convex iff

t2x2 + (1-t)2y2 + 2t(1-t)xy $\leqslant$ tx2 + (1-t)y2

-t(1-t)x2 - t(1-t)y2 + 2t(1-t)xy $\leqslant$ 0

-t(1-t)[y2 + x2 - 2xy] $\leqslant$ 0

(t2 - t)(x-y)^2 $\leqslant$ 0, which is true for t $\in$ [0, 1]

f(x) = -x^2 is convex iff

-(t2x2 + (1-t)2y2 + 2t(1-t)xy) $\leqslant$ -tx2 -(1-t)y2

-t2x2 + tx2 - (1-t)2y2 + (1-t)y2 - 2t(1-t)xy $\leqslant$ 0

t(1-t)x2 + t(1-t)y2 - 2t(1-t)xy $\leqslant$ 0

(t-t2)(x2 + y2 -2xy) $\leqslant$ 0

(t-t2)(x-y)2 $\leqslant$ 0

but (x-y)2 > 0 for all x neq y

and (t-t2) > 0 for all 0 $\leqslant$ t $\leqslant$ 1

so this is not true

therefore f(x) is not convex

Is the solution unique?

Is the criterion strictly convex?

- no because the function is linear in one of the variables, so it cannot be
- Because the objective function is linear in one of the variables, the optimization cannot have a unique solution

But what about just the B?

- BUT the β alone does have a unique solution, just by using strict convexity
  - the function $||B||_2^2$ is strictly convex in B alone

- say you have two solutions: $(B_1,\ B_{01},\ \xi_1)\ and\ (B_2,\ B_{02},\ \xi_2)$
  - can it be the case that $B_1\ \neq B_2$?
  - No, because if this was the case, that contradicts strict convexity of the function
- Q: What if you rotate your data points around some symmetry? is this a counterexample

B. Equivalent translations of optimization problems

B1. Rewriting constraints

The proble, $min_x\ f(x)\ subject\ to\ g_i(x)\ \leqslant\ 0,\ Ax = b$ can be rewitten as $min_x\ f(x)\ subject\ to\ x\ \in C$, where $C\ =\ \{x\colon\ g_i(x)\ \leqslant\ 0,\ Ax =\ b\}$

This latter formulation is <u>completely general</u>

- I think this means it does not imply convexity or need convexity to be true
- With I_C the indicator of C, we can write this in an unconstrained form: $min_x\ f(x)\ +\ I_C(x)$
  - $I_c(x)\ =\ 0\ if\ x\ \in C,\ inf\ otherwise$

B2. Using the first-order optimality condition to create an equivalent problem

For a **c**onvex problem $min_x\ f(x)\ subject\ to\ x\ \in convex\ set\ C$ and differentiable f, a feasible point x is optimal iff $\nabla f(x)^T(y-x)\ \geqslant\ 0$ for all $y\ \in C$.

In other words: all feasible direactions from x are aligned with gradient $\nabla f(x)$.

I**mportant special case:** If $C\ =\ \Re^n$(unconstrained optimization), then the optimality condition reduces to familiar $\nabla f(x)\ =\ 0$.

Proof:

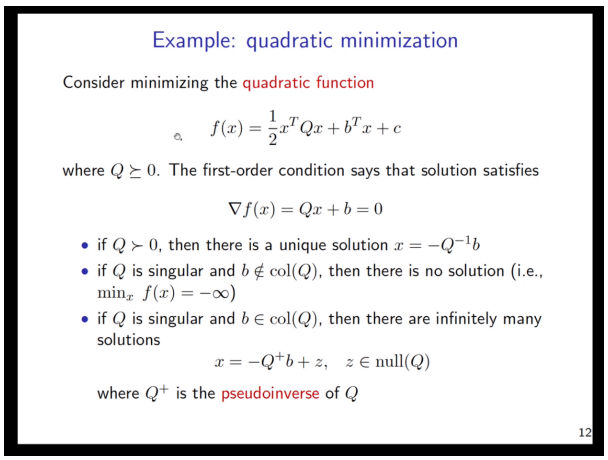
We know $\nabla f(x)^T(y-x)\ \geqslant\ 0$, for all $y\ \in\ \Re^n$

Say $a\ =(y-x)$

So $\nabla f(x)^T a \geqslant\ 0$ for all $a\ \in\ R^n$

but this means $\nabla f(x)^T(-a) \geqslant\ 0$ as well

which means that $\nabla f(x)$ must be 0



Example: Langrange multiplier optimality condition

- Lagrange multipliers solve an equality-constrained optimization problem
- $min_x\ f(x)\ subject\ to\ Ax\ =\ b$
- The Lagrange condition is $\nabla f(x)\ +\ A^T u\ =\ 0$ for some u

Proof for this condition:

Accoriding to the first-order optimality conditions, solution x satisfies $Ax\ =\ b$ and $\nabla f(x)^T(y-x)\ \geqslant\ 0$ for all y such that $Ay\ =\ b$

This is equivalent to $\nabla f(x)^T v\ =\ 0$ for all $v\ \in null(A)$

- THe null space of a matrix A consists of all the vectors B such that AB = 0 and B is not zero
- From the conditions above, A(y-x) = 0!

The null space is orthogonal to the row space of a matrix

- https://ocw.mit.edu/courses/mathematics/18-06sc-linear-algebra-fall-2011/least-squares-determinants-and-eigenvalues/orthogonal-vectors-and-subspaces/MIT18_06SCF11_Ses2.1sum.pdf

'The result follows because of this'

- Idk why?
- My guess is: This statement implies that $\nabla f(x)$ is in the row space of A
- Therefore should be some other vector in the row space that when added to $\nabla f(x)$ will make it zero?

B3. Partial optimization

- $g(x)\ =\ min_{y \in C} f(x,\ y)$ is convex in x provided that f is convex in $(x,\ y)$ and C is a convex set
- So you can always partially optimize a convex problem and retain convexity
- This helps with rewriting the SVM problem statement

Example: Support Vector Machines, again

As a reminder, the problem is:

$$min_{B,\, B_0,\, \xi_i} \frac{1}{2}||B||_2^2 \; + \; C\sum_{i=1}^{n} \xi_i, \; subject \; to$$

$$\xi_i \; \geqslant \; 0, \; i \; = \; 1, \; ..., \; n$$

$$y_i\left(x_i^T B \; + \; B_0\right) \; \geqslant \; 1 \; - \; \xi_i, \; i \; = \; 1, \; ..., \; n \; \rightarrow \xi_i \geqslant 1 \; - \; y_i\left(x_i^T B \; + \; B_0\right),$$

We can rearrange and rewrite the constraints as $\xi_i \; \geqslant \; max\left\{0, \; 1 - y_i\left(x_i^T B \; + \; B_0\right)\right\}$

- combine the two constraints as lower bounds on ξ_i

At a solution, you must have equality - cannot have strict inequality!

- Else you would be making the criterion unnecessarily large
- You could always search more of the space to get a better solution

Since this is true for all ξ in the solution set, you can rewrite the original criterion function as a **partial optimization**

- **that is, you can substitute ξ with a form you know it will take when it is part of the solution set!**
- Analogous to saying $g(x) \; = \; min_{y \in C} f(x, \; y)$, where $\xi \; = \; y$ and C is the solution set
- As we know that f is convex in all the variables, g is also convex

Knowing this, you can write the hinge form of the SVM

$$min_{B,\, B_0} \; \frac{1}{2}||B||_2^2 \; + \; C\sum_{i=1}^{n} \left[1 \; - \; y_i\left(x_i^T B \; + \; B_0\right)\right]_+$$

where $a_+ \; = \; max\{0, \; a\}$ is the hinge function.

B4. Transformation

Monotone increasing function: If $h: \; R \; \rightarrow \; R$ is a monotone increasing function that $min_x \; f(x) \; = \; min_x \; h(f(x))$

Example: Max likelihood estimation

- A good exmaple is maximising log likelihood instead of likelihood
  - especially when assuming an exponential family form
- $max_\theta \; l(\theta) \; \equiv \; max_\theta \log l(\theta)$
- The former is not concave in its parameters but the latter is!
- So max log likelihood is more tractable!
  - translates to min of a convex funciton

B5. Change of variable

If $\phi: R^n \rightarrow R^m$ is one to one and its image covers feasible set C, you can change variables in an opt problem from x to phi(y)

- can be useful in revealing hidden convex problem when there is not one prima facie

Example: geometric programming

Example 2: Relaxing equality constraints

Pick some x_0 that satisfies $Ax = b$

Let M have columns that span $null(A)$

Then $\{x: Ax = b\} = \{x_0 + My: y\}$

- you can generate any solution by taking a solution and adding something in the null space!

- if the null space only contains 0, there there is just one solution i.e. x_0

Knowing this, $min_x f(x)\ subject\ to\ Ax = b \equiv min_y(x_0 + My)$

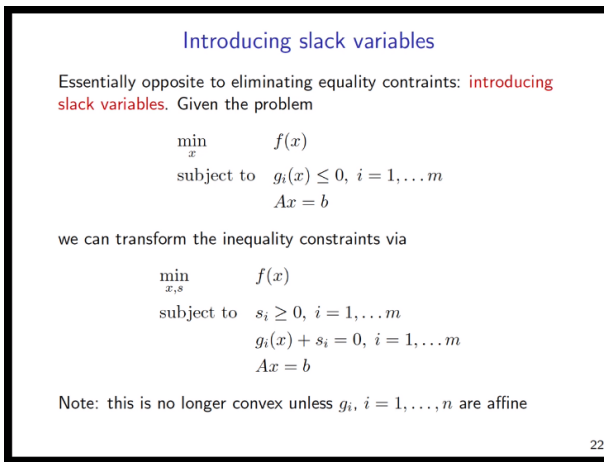
This is fully general. So we don't we always eliminate equality constraints?

1. Forming M (whose columns are the basis for the null space of A) can be hard! A can be large
2. M is going to be generically dense (?) even if A is sparse
    - Typically you would do SVD(A) to get the null space of A
    - You will get a generically dense matrix
    - this destroys the strucute/sparsity in the original matrix A, which can often be useful

B6. Introducing slack variables

Opposite to eliminating equality constraints.

You can get rid of inequality constrainst by transforming them into linear equality constraints



- this is used in Linear Programming

The issue is that introducing these variables can kill the convexity of the problem - eg if you turn a non-affine inequality constraint into an equality constraint

- be careful when you do this!

B7. Relaxing nonaffine equality constraints

In general, you can always take an enlarged constraint set $C^* \supseteq C$ and consider $min_{x \in C^*} f(x)$ instead of $min_{x \in C}\ f(x)$

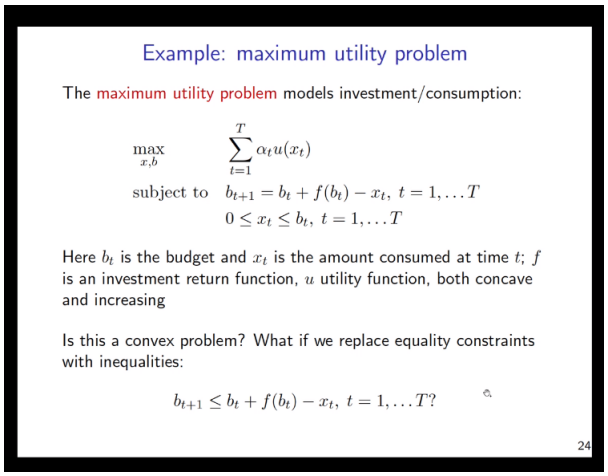
- Its optimal value is always smaller or equal than the original problem
- e.g if C is non convex, you can use its convex hull instead

This is called a relaxation.

This can be used to relax nonaffine equality constraints

e.g. if you have constraints $h_j(x) = 0$ where h_j are convex but not affine, you can replace them with $h_j(x) \leq 0$

Example: maximum utility problem



- non linear equality constraints make this problem a non-convex optimization
- but you can find an equivalent convex optimization by finding convex relaxations for these constraints