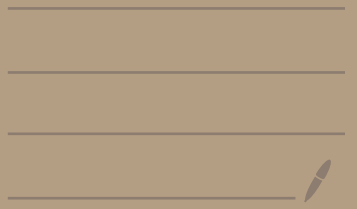
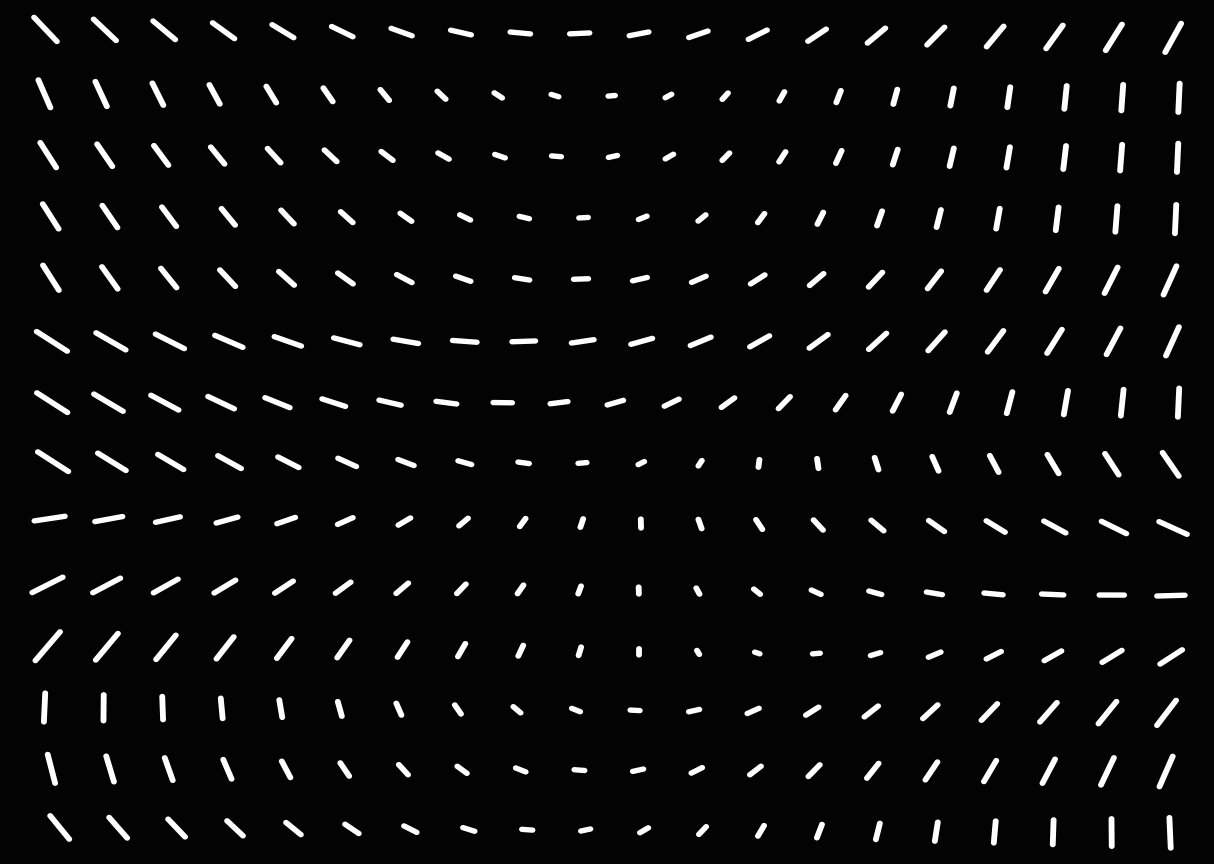
CISC 371



All Lectures



Class #0 : Elementary Differential Calculus

Functions overview

$f : \mathbb{R} \to \mathbb{R}$

A function f takes real number as input and gives real number as output

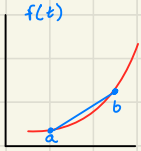
Limit

$\lim_{x \to 0} f(x) = C$

As x approaches 0, the value of the function $f(x)$ approaches 0.

Continuity

- A function is continous at a if $\lim_{t \to a} f(t) = f(a)$
- NO breaks, jump discontinuity
- Chord is a line segment for a function

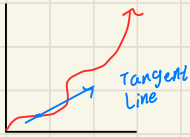


Derivatives

Power rule: $t^x = x t^{x-1}$ $\quad x^5 = 5x^4$

trig basics: $\sin(x) = \cos(x)$, $\cos(t) = -\sin(t)$

$e^x = e^x$, $\ln(x) = \frac{1}{x}$



constant rule: $C \cdot f(x) = C\left(\frac{d}{dt} f(x)\right)$

Power rule: $f(x)' \cdot g(x) + g(x)' \cdot f(x)$

Composition is $f(g(x))$ or $(f \circ g)(x)$

Composite function implies plugging one into another

EX 1

$F(x) = x^2$

$g(x) = t + 1$

$f(g(t)) = f(t+1) = (t+1)^2 = t^2 + 2t + 2$

Chain Rule:

$(f \circ g)' = (f' \circ g) \cdot g'$

Differentiate the outside, keep the inside, multiply by inner deriv

EX $f(g(t)) = (t+1)^2 = 2(t+1)(1) = 2t + 2$

Reciprocal Rule

$$h(t) = \frac{1}{g(t)}$$

$$h'(t) = \frac{g'(t)}{(g(t))^2}$$

Ex $h(t) = \frac{1}{t}$, $h'(t) = -\frac{1}{t^2}$

Quotient Rule

$$h(t) = \frac{f(t)}{g(t)}$$

$$h'(t) = \frac{g(t)\,F'(t) - f(t)\,g'(t)}{(g(t))^2}$$

Ex $h(t) = \frac{t^2}{t+1} \rightarrow h'(t) = \frac{(t+1)(2t) - (t^2)(1)}{(t+1)^2}$

Taylor Series

- Taylor series is a way to approximate a complicated function using polynomials
- Expands around a point to

Ex/ $f(t) = f(t_0) + f'(t_0)(t-t_0) + \frac{f''(t_0)}{2!}(t-t_0)^2 \ldots$

- Analytic means a function is infinitely differentiable
- Zooming in on curve

Linear Approximation

$$f(t) \simeq f(t_0) + f'(t_0)(t-t_0)$$

Quadratic Approximation

$$f(t) = f(t_0) + f'(t_0)(t-t_0) + \frac{f''(t_0)}{2}(t-t_0)^2$$

Lagrange Remainder

- No approximation is perfect, Lagrange remainder tells us how much error there is after we stop after k terms.
- gives worst case error

Class #01: Introduction to Optimization - PDF Notes

- Optimization is choosing the best element from a set according to a rule

Ex $f(t) = \frac{-t}{t^2+1}$

This has max at $t=-1$, min at $t=1$

- Minimum is the lowest value the function takes
- Minimizer is the input that gets you the minimum

Fermat's Problem

- Find the point in the plane that minimizes the sum of distances to all pts

$$f(w) = \sum_{j=1}^{n} \|w - a_j\|$$

- For 3 anchors:
  - if all angles are $< 120^\circ$, minimizer lies inside triangle
  - if an angle is $\geq 120^\circ$, minimizer is that vertex
- For 5 anchors, theres no closed form solution

3 Types of Optimization Problems

1) Unconstrained, Scalar (easy, one var)
2) Unconstrained, Vector (harder, need direction + step size)
3) Constrained, Vector (hardest, reduce to type 2)

Important Definitions

- Open Set, for every vector $\vec{v}$ in an open space if a vector is near it, only interior points
- Interior Point, a point bounded by something, with some wiggle room.
- Boundary Point, its in the subset, but its not an interior point. Touching the edge

More Definitions

- Global minimizer: lowest overall, allows equality, $\vec{w}^*$ of $f(\vec{w})$
- Strict Global minimizer: absolute lowest point, $f(t^*) < f(w)$, $\forall w \neq t^*$
- Global minimum: L is lowest level of some function f
- Local minimizer: lowest in neighborhood but allows equality

Class #02: Minimizing By Approximation - PDF Notes

Approximation

- Some objective functions are too complicated to minimize
  - ↳ Idea: Replace a hard function $f(t)$ with a simpler polynomial model
- Build quadratic approximation
- Solve minizer
- Get better guess

amplitude = size of oscillations

3 Quadratic Interpolation Methods

1) 3 points (NO Derivatives)
   - take 3 nearby points $(t_1, f_1), (t_2, f_2), (t_3, f_3)$

     $p(t) = a_1 t^2 + a_2 t + a_3$
   - Solve system Vandermode matrix to find coefficients
   - Minimizer: $\hat{t} = \frac{a_2}{2a_1}$

2) 2 points + First Derivative
   - if you have derivative at one point
   - Constraints $p(t_1) = f_1$, $p(t_2) = f_2$, $p(t_1) = f_1'$
   - Solve for quadratic coefficients

3) 1 Point + First and Second Derivative
   - $f(t_1)$
   - $f'(t_1)$ enough to know quadratic
   - $f''(t_1)$

Always check if new guess is improvement from old guess.

First Derivative → descent

Second Derivative → positive

Higher order Approximations

- Use quadratics (sweet spot)
  - ↳ Could use cubic models

Class 2: In person / Lecture video Notes

Orders of Approximation

$P_0(t) = c$ Flat line

$P_1(t) = mt + b$ Linear

$P_2(t) = a_1 t^2 + a_2 t + a_3$ Quadratic, can actually have minimum

Vandermode Matrix

- Quadratic Interpolation using vandermode matrices
- approximate using a quadratic model

$$P(t) = a_1 t^2 + a_2 t + a_3$$

To determine a_1, a_2, a_3 (coefficients) we need constraints (data points / derivative)

Case 1) 3 Points

Plug in $(t_1, f_1), (t_2, f_2), (t_3, f_3)$

$$\begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ t_3^2 & t_3 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

Minimizer: (need coeffs)

$$\hat{t} = -\frac{a_2}{2a_1}$$

Think like ↓ This is Vandermode Matrix

$$\begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

Case 2: 2 points + 1 Derivative

Points: $(t_1, f_1), (t_2, f_2)$ but also

Slope at one point: $f'(t_1)$

$$\begin{bmatrix} t_1^2 & t_1 & 1 \\ t_2^2 & t_2 & 1 \\ 2t_1 & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_1' \end{bmatrix}$$

Case 3: 1 point, First and Second Deriv

$f(t_1)$
$f'(t_1)$
$f''(t_1)$

$$\begin{bmatrix} t_1^2 & t_1 & 1 \\ 2t_1 & 1 & 0 \\ 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_1' \\ f_1'' \end{bmatrix}$$

Once coefficients are found, can analytically solve minimizer $p'(t) = 0$

Class #03: Stationarity and Convexity - PDF Notes

Stationary Points

t^* could be a min/max or saddle point

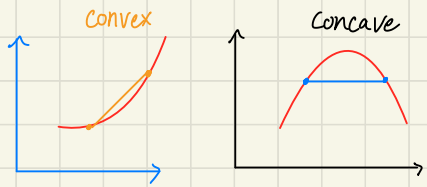
- Stationary point is when $f'(t^*) = 0$
  - ↳ function is flat

If t^* is local minimizer, it must be Stationary point

First order necessary condition?

Stationaly ≠ Minimizer always

Could be local min
local max
Saddle points



Convexity

- Convex means it lies below the chord at any 2 points
  - ↳ Strictly convex:
    - local min = global min
    - guarantees uniqueness

Every local minimum is a global minimum

Convex - function lies below or on chord

Strictly convex - Strictly below chord between any 2 points
↳ guarantees unique minimizer

Gradient Inequality

- From Taylor Expansion with remainder

$$f(t_0) + f'(t_0)(t - t_0) + \frac{1}{2} f''(\varepsilon)(t - t_0)^2$$

Meaning: when you approximate a function around point t_0. There is a point ε between t and t_0, where second deriv is evaluated

If f is convex $f''(\varepsilon) \geq 0$ then:

$$f(t) \geq f(t_0) + f'(t_0)(t - t_0)$$

This says the tangent line is below the curve

For convex functions, the tangent line is always global underestimator

- Convex helps finds mins

4. Key Concepts to Distinguish

- **Stationary point**: derivative = 0 (necessary, not sufficient for min).
- **Saddle point**: stationary but neither max nor min.
- **Convex function**: guarantees well-behaved minima.
- **Gradient inequality**: shows why convexity makes optimization stable.

$\frac{1}{2} f''(\varepsilon)(t - t_0)^2$
Error Term

Class 04: Scalar Minimization - PDF Notes

- Vector valued problems are fine to have good approximations
- Inexact methods: aim to reduce computational effort while making progress

Line Search

- moving along given direction d from current estimate t_k
  - ↳ objective function: $f(t)$

Current guess t_k, function value f_k, derivative f'_k
direction $d = \pm 1$ (scalar)

Fixed Stepsize Search

Simplest approach: $t_{k+1} = t_k - s_0 f'(t_k)$
s_0 - constant stepsize

Too Big → overshoot, divergence
Too Small → slow convergence

Inexact Backtracking Search

- Start with user's stepsize s_0, shrink if it is too large. Keep multiplying by β.
  - Pick largest s improves $f(t)$
- Want smaller stepsizes

Armijo Backtracking Condition

- Based on gradient inequality Reduce s by a factor β

Step s such that:

$$f(t_k + sd) \leq f(t_k) + \alpha_k s d$$

where $\alpha_k = f'(t_k)/2$

New point must be strictly lower than predicted by tangent line.

Key Takeaways

1. **Fixed stepsize** methods are simple but unstable.
2. **Backtracking** systematically reduces stepsize until descent is guaranteed.
3. **Armijo condition** formalizes the test: ensures sufficient decrease, not just any decrease.
4. This makes optimization efficient, robust, and less sensitive to poor stepsize choices.

Class 4 - Lecture Video Notes

Finding Stationary Point for Approximation

$f'(t_{k+1}) = 0$

Differentiate:

$$f'(t) \approx f'(t_k) + f''(t_k)(t - t_k)$$

$$0 = f'(t_k) + f''(t_k)(t_{k+1} - t_k)$$

$$t_{k+1} = t_k - \frac{f'(t_k)}{f''(t_k)}$$

Direction $d_k = -f'(t_k)$

Step size $s_k = \frac{1}{f''(t_k)}$

$\Rightarrow$ Iteration $t_{k+1} = t_k + s_k d_k$

or

$t_{k+1} = t_k - \alpha f'(t_k)$

- Smaller step size is always better

$f(t) = t^2$

$f'(t) = 2t$

$d_k = -2t_k$

Backtracking/inexact line search is for backtracking

$t_{k+1} = t_k - 2s t_k = (1-2s) t_k$

Start with $t = 10$, $s = 0.25$

$t_1 = (1-0.5) \cdot 10 = 5$

$t_2 = (1-0.5) \cdot 5 = 2.5$

$t_3 = 1.25$

Converges to 0.

Backtracking

- Start with intial guess s
  - ↳ Factor of β: s, βs, $\beta^2 s$, $\beta^3 s$
    - ↳ Searching for a good step size

Armijo condition: New point should be below a certain line to ensure progress

Exponential Backoff: s, βs, $\beta^2 s$

Class 05: Functions with a Vector Argument - PDF Notes

From Derivatives to Partial Derivatives

- multivariable calculus uses partial derivatives $\frac{\partial f}{\partial w_1}, \frac{\partial f}{\partial w_2}$

How function changes as one coordinate changes

Directional Derivatives

- move in any direction $\vec{v}$
- Directional derivative: $D_{\vec{v}} f(\vec{w}) = \lim_{h \to 0} \frac{f(\vec{w} + h\vec{v}) - f(\vec{w})}{h}$
- Dot product between gradient and the direction vector.

Gradient (∇f)

- gradient is a row vector (1-form) for all partial derivatives

$$\nabla f(\vec{w}) = \left[\frac{\partial f}{\partial w_1}, \frac{\partial f}{\partial w_2} \ldots \frac{\partial f}{\partial w_n} \right]$$

- gradient points in the direction of steepest ascent.
- negative direction is steepest descent.

Jacobian Matrix

- For vector valued function $\vec{F}(w) = [f_1(\vec{w}), \ldots f_m(\vec{w})]^T$
- Jacobian is matrix of gradients

$$J_f(\vec{w}) = \begin{bmatrix} \nabla f_1 \\ \nabla f_2 \\ \nabla f_3 \end{bmatrix}$$

Level Curves

- level curve of $f(\vec{w})$ is the set of all points with same function value

$$SC(f, 1) = \{\vec{v} : f(\vec{v}) = 1\}$$

- gradient is always perpendicular to level curves.

Class 6 - Video Lecture Notes

Connecting multivariable Calc with Lin Alg

- Functions with multiple variables: $f(w_1, w_2 \dots w_n)$
  - ↳ $f(w_1, w_2) = w_1^2 + 3w_2$

- gradients is vector of partial derivatives: $\nabla f = \left[\frac{\partial f}{\partial w_1}, \frac{\partial f}{\partial w_2} \dots\right]$

Example: $f(w_1, w_2) = w_1^2 + 3w_2$

$\nabla f = [2w_1, 3]$, at point 1,2, $\nabla f = [2,3]$

Directional Derivative:

rate of change of f in some direction v

$D_v f = \nabla f \cdot v$

At a point with certain gradient

$D_v f = [2,3] \cdot [1,0] = 2$

↳ f increases in w_1 direction by 2 steps per

1 Form

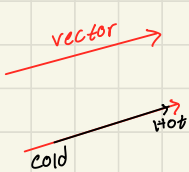
- linear functional that takes a vector and produces a number

vectors are directions

1-forms are row vectors that act on vectors

gradient is naturally a row vector (1 Form)

↳ multiplies a column vector, gets directional derivatives



one Form tells change along vector

If a is vector $[\vec{a}]$, transpose is $[\vec{a}]^T = a$

$[a]^T = \vec{a}$

Jacobian Matrix

$f: \mathbb{R}^n \to \mathbb{R}^m$

$f(x,y) = (x^2+y, xy)$

$J = \begin{bmatrix} 2x & 1 \\ y & x \end{bmatrix}$

Theorem: $a^T w$ (linear function)

gradient is a

Class 07: First Order Optimization - Steepest Descent - PDF Note

Optimization Setup

want: $w^* = \arg\min_w f(w)$

- Start at point w_0

At every step, we need a direction and a stepsize

- Updated rule $w_{K+1} = w_K + sd$

w_{K+1}: next step; w_K: starting step; s: stepsize; d: direction

Descent Directions

- d is descent direction if the directional derivative is negative
- Steepest descent is fastest decreasing f.

$$d = -\nabla f(w_0)$$

Stepsize Selection

- Fixed step size, too large → overshoot, oscillate
  , too small → crawl slowly
- Armijo Backtracking
  - Start with large s_0, shrink backwards

$$f(w_K + sd_K) \leq f(w_K) + \alpha s \nabla F(w_K) \cdot d_K \quad ?$$

Algorithms

- Input w_0, s, K_{max}, g_{mag}

1) Find gradient: $g = \nabla f(w)$
2) Set $d = -g^T$
3) Update $w = w + sd$
4) Stop when $\|g\| < g_{mag}$ or max iterations

Examples

- Quadratic Function

$f_1(w) = w^T K w$, $K = \begin{bmatrix} 1/4 & 0 \\ 0 & 1 \end{bmatrix}$

start at $w_0 = (-3.0, 3.2)$

Backtracking: ($s_0 = 1.5$, $\beta = 0.75$)

$$f_1(w) = \begin{bmatrix} -3.0 \\ 3.2 \end{bmatrix} \begin{bmatrix} 1/4 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -3.0 & 3.2 \end{bmatrix}$$

Class 09: Newton's Method - PDF Notes

Point of Newton's Method

- Steepest descent is cool, but often zigzags in deep valleys → slow convergence
- Use curvature (Hessian, 2nd Derivative) to adjust step direction/length
- Use more info than just slope

Scaled Descent

- Some functions are sensitive to one variable more than others

$f_1(w) = 100 w_1^2 - 10 w_1 w_2 + w_2^2 + 100$

↳ Changes in w_1 are 10x more impactful than w_2

Fix: Scale variables with matrix B

$w = B M_v$, B reduces effect of w_1

TLDR: Use inverse Hessian as B, for smoother and faster convergence

Newton's Method

- Use quadratic model

Taylor Approx: $f(w) = f(w_0) + \nabla f(w_0)(w - w_0) + \frac{1}{2}(w - w_0)^T H (w - w_0)$

$$d = -H^{-1} \nabla f(w_0)$$

$$w_{K+1} = w_K + d$$

Newton Direction:

Converges in 1 step if model is exact?

Damped Newton's Method

- If steps are too big, Newton can overshoot
- Damping → scales back step size with armijo backtracking

??

Pitfalls

- Newtons method only works if the Hessian is positive definite
- Hessian is indefinite, method can divergent

Class 10: Nonlinear least squares and L-M Algorithm

Idea: Solve optimization problems, where the model being fitted are nonlinear. (similiar to GPS)

Nonlinear Least Squares

- Trying to minimize: $f(\vec{w}) = \frac{1}{2} \|\vec{r}(\vec{w})\|^2$

  $\vec{w}$ = parameters you're optimizing

  $r(\vec{w})$ = vector of residuals (error between predicted/observed)

Solving Gauss-Newton

- approximates as a locally linear problem

Iteration: $\vec{w}_{k+1} = \vec{w}_k + (J^T J)^{-1} J^T \vec{r}$

Jacobian Matrix (derivatives of residuals w,r,t parameters)

- Scaled steepest descent method.

Levenberg-Marquardt Algorithm (LM)

- Smarter Gauss newton avoids problems when:
  - your initial guess is bad
  - Jacobian is near singular

Regularized the problem

hyperparam: $\pi \geq 0$

- Add Damping term: $\vec{w}_{k+1} = \vec{w}_k + (J^T J + \pi I)^{-1} J^T \vec{r}$
  - π is large, gradient descent
  - π is small, acts like gauss newton

Non Linear Least Squares Solution

- Finding parameters x that fits observed data by minimizing sum of squared error

- Linear least squares: $f(x) = Ax - b$
  - ↳ minimize $\|Ax - b\|^2$

- GPS has location (x, y, z) and clock bias t
  recieved measures pseudorange: $d = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} + c \cdot t$

To solve: Gauss newton method
: Levenberg - Marquardt

Class 12: Convex Functions, Convex Sets and Level Sets

Convex vs. Monotonic

- Motonic: function decreases steadily
- Convex: function lies between chord connecting two points

Define

- Convex Function: $f((1-\theta)u + \theta v) \leq (1-\theta)f(u) + \theta f(v), \quad 0 \leq \theta \leq 1$

- **Meaning in words:**
  The value of the function at a point in between u and v is less than or equal to the average of the function values at u and v.
- **Geometric meaning:**
  The graph of the function lies below or on the straight line (the "chord") joining the two points $(u, f(u))$ and $(v, f(v))$.
- **Why important:**
  - Ensures **no local minima except the global one** → optimization is easier and more reliable.
  - Makes constrained optimization problems solvable with efficient algorithms (why convexity is a foundation for ML, optimization, and economics).

- Strictly convex: inequality is strict when $u \neq v$.
- Convex set: contains all linear interpolations between its points.
  - Taking weighted average between two points

Convex, if any two points inside it, entire line segment connecting is also in the set.

Important Convex Functions

- Affine functions ($f(w) = Mw + C$): always convex
- Quadtratics form ($f(w) = w^T K w$): convex if K is positive definite

Operations Preserving Convexity

- Positive Scaling lf is convex if $l \geq 0$
- sum of convex functions is convex
- Affine transformation: if f is convex, $g(v) = f(Mv + c)$ is convex

Tangent Plane Property

- Convex functions are always above tangent plane
  - $f(w) \geq f(w_0) + \nabla f(w_0)(w - w_0)$
  - gradient & inequality ensures convexity

Level Sets

- $SL(f, l) = \{w \mid f(w) \leq l\}$
  - level sets of convex functions are convex

Class 12: In Person Notes

convexity, find point inside/outside set.

for any point, line is entirely in the set

monotonally increasing → $f(t+\theta)$ on an interval means θ

$l(\theta) = tk + l$

$= (1-\theta)$

$F(1) =$

linear interpolant above line

Convex on $\mathbb{V} \subseteq \mathbb{R}^1$

Recall scalar

Convex Sets:

$\mathbb{V} \subseteq \mathbb{R}^n$ is convex set

$v \in \mathbb{W}$ or $w = v$

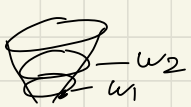
Convex function is bowl like

Find points and planes, line is always underneath.

Example Plane:

$f(\vec{w}) = m\vec{w} + b$

b: offset constant



$= [w_1 \; w_2] \begin{bmatrix} & \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$

$= \vec{v}^T K \vec{v}$

Class 14: Neural Networks (Single neuron)

Covering

- Activation function
- Optimization with steepest descent
- vector representation of data
- Back propagation

How do we handle multiple observations?

In practice, $\vec{d} \sum_{k=1}^{m} \vec{d}_k$

Structure of Single Neuron

- Artificial neuron: $u = \vec{x} \cdot \vec{w} + b$ and then $z = \phi(u)$
  - x: input vector
  - $\vec{w}$: weight vector
  - b: bias (Augment $\vec{x}$ with 1)
  - $\phi(u)$: activation function

Common Activation Functions

- Logistic / Sigmoid $\phi(u) = \frac{1}{1+e^{-u}}$ Outputs between 0-1 Deriv: $\phi(u)(1-\phi(u))$
- ReLU: $\phi(u) = \max(0, u)$ Outputs between $0 - \infty$ Deriv 1, if $u > 0$ else 0
  - ↳ Hidden layers for neural nets
- Heaviside: 0 if $u < 0$, 1 if $u \geq 0$ outputs between $\{0, 1\}$ Output Classification

Training via Steepest Descent

- minimized Squared error loss: $g(r) = \frac{1}{2} r^2$, where $r = y - \phi(u)$
- Gradient Descent update:

$$w_{k+1} = w_k + \eta \cdot \vec{d}, \text{ where } \vec{d} = -\nabla f(\vec{w})$$

- Backpropagation Terms:
  - Residual error: $r = y - \phi(u)$
  - Descent Direction: $\vec{d} = -\vec{x} \cdot r$

After training, neuron learns a decision boundary

Output Layer Differentiation

- weight vector split into $\vec{w} = [w_1, w_2, \ldots, w_n, b]$
- Linear term becomes: $u = \vec{w}_{1..n} \cdot \vec{x} + b$
- Backpropagation over layers

Class 15: Neural Networks: Multiple Layers

Goal: Extend neural net training with grad descent to multi-layer ANN's. Calculating gradients (derivatives) using back propogation when multiple layers

Adding more layers and nonlinear activations allows neural nets to learn more complex decision boundaries.

Conceptually

- layer is a lin transformation followed by a nonlinear activation
- Linear: matrix vector product: $Wx + b$
- Activation: $\phi(v)$

- Forward pass: compute activations layer by layer
- Backward pass: (Back propagation): compute gradients layer by layer to update weights

Structure of 3 layer Neural Network

- Layer 1: (Input) Raw input features
- Layer 2: (Hidden) Applies weights and activation functions
- Layer 3: (Output) Computes final prediction from Layer 2's output

Forward Passing

Layer 2: $v^2 = [x, 1] \cdot w^2$

Layer 2 Activation: $\phi^2 = \phi(v^{(2)})$

Layer 3 Linear output: $v^{(3)} = [\phi^{(2)}, 1] \cdot w^3$

Layer 3 Activation $\phi^{(3)} = \phi(v^{(3)})$

$\psi = \frac{d\phi}{dv}$: derivative of activation function

Gradients use Hadamard Product (element wise): $\psi \odot b$

Backpropogation

- gradients are computed layer by layer
- Output layers use errors between prediction and true label (residual)
- Hidden layers use chain rule to propogate error backward using jacobian

Each layers gradient is computed based on:

- activation derivative, error propagated from next layer, input data

Class 17: Back-Propogating Scale Factors of Gradient Components

- How propogating scale facts backword through neural networks instead of computing jacobian matrices
- Scale factors tells how much error flows backward through each neuron

Forward and Backward Computation

- Forward Evaluation: Compute the output $z(w)$ of the network given an input x and then compute the objective function (squared error)

- Backward: Compute gradients layer by layer using chain rule.
  - Uses scale factors (partial derivatives) to propagate error backward

Optimization Formulation

- Neural networks are treated as an unconstrained optimization problem:

$$w_{k+1} = w_k - \eta \nabla f(\vec{w}_k)$$

where η is the learning rate and f is the squared error function

- 2-layer network:
  - Layer 1 (input): 2 features
  - Layer 2 (Hidden): 2 neurons
  - Layer 3 (Output): 1 neuron

- Each layer has weights and activations, denoted $\phi(t)$ and its derivative $\psi(t)$

Backpropagation Steps

- residual: $r = y - z(\vec{w})$
- output layer scale factor: $b_3 = -r$
- compute layer 3 gradient: $\vec{d_3} = -\vec{x_3} \cdot \psi_3 \cdot b_3$
- Back-propogate to L2: $\vec{b_2} = \vec{w_3} \cdot w_3 \cdot b_3$
  $\vec{d_2} = -\vec{x_2} \cdot \psi_2 \cdot b_2$
- Final descent vector: $\vec{d_3}$ and $\vec{d_2}$
- Equations: Activation: $\phi(t) = \frac{1}{1+e^{-t}}$, $\psi(t) = \phi(t)(1-\phi(t))$

Forward Evaluation Example

- Given: weights and observation vector
  - ↳ Activation function: logistic

- Forward computations:

  Layer 2 outputs: $\vec{v}_2, \vec{\phi}_2, \vec{\psi}_2$

  Layer 3:

  $v_3 = \vec{x}_3 \cdot \vec{w}_3$,

  $\phi_3(v_3) = z(\vec{w})$

Observation:

- derivatives (ψ) are numerically small → training progresses at different rates across layers

Class 18: Constrained Optimization Problems

Purpose: introduce constrained optimization. Point is to minimize objective function subject to constraint be equality or inequality-based.

- Feasible Sets: Allowed Solutions
- Solve both Quadratics / Linear Objectives
- 3 Classic optimization types:

Quadratic Programming (QP), Convex Optimization, Linear Programming

Constrained Optimization

- Minimizing a function (squared distance) subject to constraints.
- 2 constraint types: Equality Constraint $Mw = c$
  : Inequality Constraint $Mw \leq c$

Linear Equality Example

- $-w_1 + 2w_2 = 4$
  (two variables)
- Closest point given constraints
  ↳ subject to $[-1\ 2]w = 4$

Linear Inequality

- $w_1 - w_2 \leq 4$, feasible region in half space. Optimization form:
  $\min \|w - w_0\|^2$. Subject to $[1\ -1]w + 4 \leq 0$.

Convex Optimization Basics

- Property function $P_i(w)$ is convex → defines constraint
- Feasible point: satisfies all $P_i(w) \leq 0$
- Feasible set F: is all such w values that satisfy the property function/constraint
- If $Mw - c \leq 0$ defines constraint and M is full rank, feasible region is convex

Standard form for Convex Optimization

- min $f(w)$, subject to $p_i(w) \leq 0$, $\mu w = c$
  $f(w)$ is convex, $p_i(w)$ are convex
- Quadratic Programming: min $w^T K w + a^T w$
  $Aw - b \leq 0$, $\mu w = c$
  where K is symmetric, positive semidef

Examples:

Class 20: Lagrage Multipliers for 2D Functional Convex problems

Purpose: lagrange multipliers help solve equality constraints. Shows how minimizer lies on level curve of objective function and constraint function.

The key takeaway is: At the optimal point, the gradients of the objective and constraint are parallel and scaled versions of each other, giving rise to the Lagrange multiplier condition.

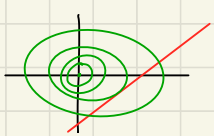
Example problems

- Level curve and constraint curve.
- Looking for point where lowest possible level curve just touches the constraint. Geometrically this corresponds to gradients being parallel.

$$\nabla f(w^*) = \nu \nabla p(w^*)$$

Ex 1

- Squared Length + Linear Equality
- Objective: Minimize $f_1(w) = \|w\|^2 = w^T w$
- Constraint: $P_1(w) = [-1 \ \ 1]w + 3 = 0$



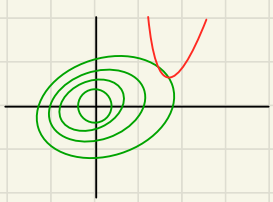
Increasing values of l, (radius)

Optimal Point $w_1^* = [1.5, -1.5]$

1) Compute gradients
$\nabla f(w) = [2w_1, 2w_2]$
$\nabla p(w) = [-1, 1]$
$[2w_1, 2w_2] = \nu[-1, 1]$
2) $2w_1 = -\nu$, $2w_2 = \nu$
$w_1 = -\frac{\nu}{2}$ $w_2 = \frac{\nu}{2}$
3) Apply constraint
$-w_1 + w_2 + 3 = 0$
$\nu = -3$
$w_1 = -\frac{-3}{2} = 1.5$, $w_2 = \frac{-3}{2}$
$w^* = [1.5, -1.5]$, $\nu = -3$

Ex 2

- Weighted Norm and Quadratic Constraint
- Objective: $F_2(w) = \frac{1}{2} w^t K w$
- Constraint: $P_2(w) = (w_1 - 3)^2 + 1 - w_2 = 0$ → parabola



Increasing l, will first touch

$w_2^* = [2.155, 1.714]$

Satisfies lagrange condition

$\nabla f_2(w^*) = \nu \nabla p_2(w^*)$

Ex3

Affine Objective + Quadratic Constraint

- Objective $f_3(w) = [1, 1]\vec{w} = w_1 + w_2$

$p_3(\vec{w}) = (w_1 - 1)^2 + 1 - w_2 = 0$

Using lagrange multipliers:

$\nabla f_3(w^*) = \nu_3 \nabla p_3(w^*)$

$[1, 1] = \nu_3 [2w_1 - 2, -1]$

$1 = \nu_3 (2w - 2)$

$1 = \nu_3 (-1)$

$1 = -\nu_3$

$1 = -1(2w_1 - 2)$

$1 = -2w_1 + 2$

$\frac{-1}{-2} = \frac{-2w_1}{-2}$

$w_1 = 0.5$

Plug $(w_1 - 1)^2 + 1 - w_2 = 0$

$(w_1 - 1)^2 + 1 - 0.5 = 0$

$(w_1 - 1)^2 = -0.5$

$w_1 = 1.25$

Insert lecture photo of meshes

Core Idea: $\nabla f(w^*) = -\nu \Delta p(w^*)$

ν is the lagrange multiplier

gradients are parallell and opposites

- Linear Constraint, flat vertical plane
- Quadratic Constraint, curved surface

Class 21: Constrained Optimization using Lagrange Multipliers

- Introduces lagrange function, lagrange equation.
- Finding minimizer and multiplier using matrix algebra
- Solves using KKT system (Jacobian + gradient)

Lagrange function and Equation

$L(w, \nu) = f(w) + \nu \cdot p(w)$

$f(w)$: Objective; ν: lagrange multiplier

$p(w) = 0$ is constraint

$\frac{\partial L}{\partial w} = 0, \quad \frac{\partial L}{\partial \nu} = 0$. Solve for both w^* and ν^*

Ex 1: Squared Length + Linear Equality

- Objective $f_1(w) = w^t I w$
- Constraint $p_1(w) = [-1 \ 1]w + 3 = 0$

Lagrangian: $L_1(w, \nu) = w^t w + \nu([-1, 1]w + 3)$

$w^* = [1.5; -1.5], \quad \nu = 3$

Solve:
$$\begin{bmatrix} 2I & [-1;1] \\ [-1\ 1] & 0 \end{bmatrix} \begin{bmatrix} w^* \\ \nu^* \end{bmatrix} = \begin{bmatrix} 0 \\ -3 \end{bmatrix}$$

Lagrange function is $L(w, \nu) = f(w) + \nu p(\vec{w})$

Objective + Scalar multiple

- minimizer of $L(w, \nu)$ must be a stationary point.

Example 2: Mechanical system with 2 springs

- Objective: minimize potential energy of 2 springs: $f_2(w) = \frac{1}{2} w^t K w$
- Constraint: $p_2(w) = mw - Ce = 0$

where $K = \text{diag}(K_1, K_2), \quad m = [-K_1, K_2]$

- Lagrangian System: $L_2(w, \nu) = \frac{1}{2} w^t K w + \nu(mw - Ce)$
- Solve:
$$\begin{bmatrix} K & m^t \\ m & 0 \end{bmatrix} \begin{bmatrix} w^* \\ \nu^* \end{bmatrix} = \begin{bmatrix} 0 \\ Ce \end{bmatrix}$$

Sample: $K_1 = 1, K_2 = 2, Ce = 6$

$w^* = [-2; 2], \quad \nu = -2$

General Case: Quadratic objective + Linear Equality Constraints

- Problem Form: $\min f(w) = \frac{1}{2} w^t K w + q^t w$ subject to $Mw = c$
- KKT System: $\begin{bmatrix} K & M^T \\ M & 0 \end{bmatrix} \begin{bmatrix} w \\ \mu \end{bmatrix} = \begin{bmatrix} -q \\ c \end{bmatrix}$
  - used to solve minimization with constraints
  - K must be symmetric positive definite, M full rank
  - has both positive and negative eigenvalues due to form

Class 23: Dual Formulation of Lagrange Equations

Focus: Newton's method extension

- Taylor Approximation, gradient + Hessian

- Minimizing Functions:

  1D Newtons Method: $x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$

- From 1D to Multivariate:

  gradient ∇f replaces slope,

  Hessian matrix replaces second deriv

  : $x_{k+1} = x_k - H^{-1} \nabla F(x_k)$

Feasible set:

$F = \{ \vec{v} : m\vec{v} - c = 0 \}$

Taylor Expansion view

$f(x + \Delta x) \simeq f(x) + \nabla f(x)^T \Delta x + \frac{1}{2} x^T H \Delta x$

Computing Newton Step

- $H \Delta x = -\nabla f(x)$
- Linear system:

Ex $F(x) = \begin{bmatrix} 6 \\ 4 \end{bmatrix}$, $H = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$

$\Delta x = -H^{-1} \nabla F(x) = \begin{bmatrix} -3 \\ -1 \end{bmatrix}$

Newton Decrement

- tells us how close we are to optimum

$\lambda(x) = \sqrt{\nabla f(x)^T H^{-1} \nabla f(x)}$

Stop when $\frac{1}{2} \lambda(x)^2 \leq \epsilon$

---

3 Phases of Convergence

1) Damped, take small steps unsure of curve
2) Quadratic, Extremely fast convergence
3) Overshoot, may take large steps

Newton Direction is always a descent direction as long as Hessian H is positive definite.

Lagrangian Function: $b(w, \nu) = f(\vec{w}) + \nu p(\vec{w})$

At a stationary point w^* $\nabla F(\vec{w}^*) = -\nu \nabla p(w^*)$

By Solving Dual, you get same as solving Primal problem

Dual Function is always concave, even if original function isn't convex

Class 24: Inequality Constraints and KKT Conditions

- Linear inequality constraints define convex sets
- KKT conditions help determine optimal solutions
- Feasible Point: Solution that satisfies all inequality constraints

Problems have objective function and constraints:

$$f(w) = \frac{1}{2} w^T K w + q^T w \qquad A\vec{w} \leq \vec{b}$$

$$L(w, \pi) = f(w) + \pi^T (A\vec{w} - b)$$

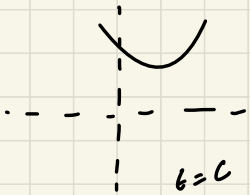
KKT conditions

1.) Primal Feasibility: $A\vec{w} \leq \vec{b}$
2.) Dual Feasibility: $\pi \geq 0$
3.) Stationarity: $K\vec{w} + \vec{q} + A^T \vec{\pi} = 0$
4.) Complementary Slackness: $\pi_i (a_i \cdot \vec{w} - b_i) = 0$ for each constraint

If all 4 hold, the point is a KKT point which is optimal if the objective is convex.

watch videos

Class #24: KKT Conditions for Constrained Optimization



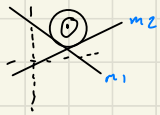
$b = c$

$r = c - 0$

$f(t) = 0$

Solution at $b = c$

$F(w) = \|w - g\|$



$m_1 : w - c_1 = 0$

Minimize Objective

Intersection

Dual formulation (Big → Small)

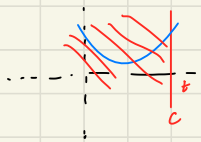
Consider: 1D quadratic objective

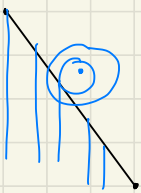
$f(b) = a_1 b^2 + a_2 b + a_3$

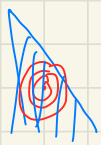
Constrained by $b^* \leq c$

F: Feasible Set

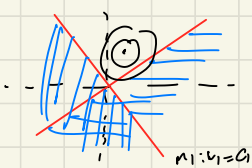
Functions are locally convex







If the unconstrained minimizer is the only its in the feasible set



Intersection of 2 half spaces

Property P_1 is active

Lagrange Eq

$L(w, \pi) = f(\vec{v}) = \pi_1 [w - c_1] + \pi_2 [c_2 - w]$

KKT Conditions:

For quadratic: $f(\vec{v}) = \frac{1}{2} v^T K v + \vec{f}^T \vec{v}$

For linear inequalities $A\vec{v} = \vec{b} = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} \vec{v}$

A Point v^* is a constrained minimizer

Primal feasability: $Av^* = \vec{b}$

Complementary slackness

Class 25: Geometry AT KKT Points

Purpose: KKT conditions describe how constraints interact with objective functions:
- which constraints are active
- which are inactive

Interpreting Lagrange Multipliers

- $\lambda = 0$: constraint is inactive at optimum (solution must be in feasability region)
- $\lambda > 0$: constraint is active (solution lies on boundary defined by constraint)
- $\lambda < 0$: occurs only with equality constraints (Equivalent to 2 constraints)
  - ↳ $\lambda_1 = 0$, $\lambda_2 > 0$

Main Geometric Takeaways

Quadratic objective + linear inequality constraints
- minimizing quadratic that produces level curves
- Constraints produce half-spaces
- Optimal point found when: 1) objective is minimized
  2) Constraints are satisfied

Inactive: $\lambda = 0$
active: $\lambda > 0$

KKT Conditions

1.) Primal feasibility: $A\hat{w} \leq b$

2.) Dual feasibility: $\lambda \geq 0$

3.) Stationarity: gradient of lagrangian equals zero: $Kw + q + A^T\lambda = 0$

4.) complementary slackness:
↳ $\lambda(a_i^T w - b_i) = 0$ for each constraint

Example 1: (3 constrained)
- only 3rd constrained is active → $\lambda(0,0)$

Example 2:
- When each constrained is used alone, λ values are larger

Equality constraint interpretation
- Equality $mw - c = 0$, is treated as two inequalities
- $mw - c \leq 0$
- $-mw + c \leq 0$
- only one inequality is active → explains negative λ in equality-case solutions

Class 26: Constrained Least Squares and Tikhonov Regularization

- Extension of ordinary least squares by adding constraints/penalties to stabalize solutions

Ordinary Least Squares

Goal: find weights w that make predictions $Xw \approx y$

OLS objective: $\min \|Xw - y\|^2$

Solution:

take gradient and set to 0, gives Normal Equation

$$X^T X w = X^T y$$

Assuming X has full rank: $w^* = (X^T X)^{-1} X^T y$

Issues:
- Sensitive to outliers
- test error high (overfitting)

Constrained Least Squares

- Force solution vector to have limited magnitude
  - $\|w\|^2 \leq \theta$

CLS Problem: $\min \|Xw - y\|^2$, s.t. $\|w\|^2 \leq \theta$

Lagrange Multiplier: $L(w, \pi) = \|Xw - y\|^2 + \pi(\|w\|^2 - \theta)$

2 Cases:

1) OLS solution already satisfies constraint
2) OLS violates constraint (need $\pi > 0$)
   - $(X^T X + \pi I) w = X^T y$

CLS pulls solution closer to smaller norm.

Tikhonov Regularization

- Instead of constraint, add penalty to loss

General form: $T(w, \alpha) = \|Xw - y\|^2 + \|\alpha R w\|^2$

Special case:
- $R = I$
- $\pi = \alpha^2$

$$\min \|Xw - y\|^2 + \pi \|w\|^2$$

Ridge Regression

Tikhonov, π is chosen by user

Class 27: Ridge Regression and The Lasso

Goal: Improving Linear regression with messy, high dimensional data

Standardizing Data (Z-Score)

- mean = 0
- variance = 1

$$z = \frac{x - \mu}{\sigma}$$

- Standardized data has lower RMSE

Ridge Regression (L2 Regularization)

- Ordinary least savare creates nonsense coefficients when:
  - variables are correlated
  - data is noisy
  - many features

Ridge adds penalty to large weight:

$$\min_w \|Xw - y\|_2^2 + \lambda \|w\|_2^2$$

Shrinks weights towards 0

Lasso (L1 Regularization)

- imposes constraint on sum of absolute values of weights

$$\|w\|_1 \leq \theta$$

- Key superpower:
  sets some coefficients Exactly to 0
  This means automatic feature selection

$$w_{ridge} \approx (0.254, 0.962)$$
$$w_{lasso} \approx (0, 1)$$

Elastic Net

- combines ridge + lasso and gives best of both worlds

Class 29: Support Vector Machines

Core Idea: SVM finds best seperating hyperplanes between 2 classes (class +1, class -1)

Basic Intuition

- Hyperplane: $w^T x + b = 0$
  - w: weight vector
  - b: bias
- perceptron can find seperating lines, but not the best one. SVM finds line with largest margin between 2 classes
- Support vectors are data points closest to hyperplane that define dges of margin, hyperplane is midway between closest pos/neg points

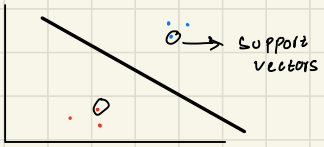
Marginal Max

$y_i(w^T x_i + b) \geq 1$

Smaller weight vector means flatter slope → wider margin

margin = $1/\|w\|$, maximizing margin

minimize $w^T w$



$\vec{n} = \frac{\vec{w}}{\|w\|}$, $a = \frac{b}{\|w\|}$

Margin (r) is distance between HI and closest data points.

$r = \vec{n} \cdot \vec{x}_s + a$

$r = x_s^T \vec{n} + a$

Primal Form:

- minimize $\frac{1}{2}\|w\|^2$
- subject to $y_i(w^T x_i + b) \geq 1$
- impossibly to solve normally, use dual formula

$$L(w, b, \alpha) = \frac{1}{2} w^T w - \sum_i a_i \left[y_i (w^T x_i + b) - 1 \right]$$

1) derive w.r.t w
2) derive w.r.t b

dual: $\max_{\alpha} \left(\sum_i a_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (x_i^T x_j) \right)$

only dot products left

Class 30: Dual Formulation of SVM

Goal: Find best seperating hyperplane: $\min \frac{1}{2} w^T w$ subject to $y_i(w^T x_i + b) \geq 1$

Lagrange Multipliers

- introduce 1 multiplier for every constraint

$$L(w, b, \alpha) = \frac{1}{2} w^T w + \sum_i \alpha_i \left[1 - y_i(w^T x_i + b)\right]$$

Derivatives

- set gradients to 0 to remove w and b
- w is just weighted combination of data points

Plug back into Lagrangian: $\alpha_i \geq 0$, $\sum_i \alpha_i y_i = 0$

Once α are found:

- Compute $w = X^T Y \alpha$
- Compute b using support vector $y_i(w^T x_i + b) = 1$
- Classify new data with $\text{sign}(w^T x + b)$

- no standard way of rep data in ML
- Optimal, Plug into neuron.
- adjust learning rate, bill training data is good.

Labels $y_i = \pm 1$

Concept: Hyperplane Seperation

H is a unit normal, $\|h\| = 1$, bias scaler $= a$

Idea: maximize margin from hyperplane to data

Define: Support vector is point closest to hyperplane

Consider: 1 Support vector

Define $r \geq 0$ as the distance from Support vector to hyper plane

$\vec{h} \cdot x_i + a = r$

Sneaky:

use $y_i = \pm 1$

For $y = +1$ have $w(h \cdot x_i + b)$

For $y = -1$ have $y_i(h \cdot x_i + a) = r$

What if its not a support vector,

- has to be atleast r away
- For $y_i = +1$ have $h \cdot x_i + a \geq r$

$(h \cdot x_i + a) \geq 1$

For any vector x_i have $y_i(h \cdot x_i + a) \geq r$

Consider an unrestricted vector $\vec{w} \neq 0$

Define $\frac{1}{r} = \|\vec{w}\|$ s.t. $\vec{h} = \frac{\vec{w}}{\|\vec{w}\|}$

or

$\vec{w} = \frac{\vec{h}}{r}$

$w \cdot x_i + b = 1$

Rewrite: $h \cdot x_i + a = r$

$= \frac{1}{r}(\vec{h} \cdot x_i + a)$

Maximize: r

minimize: $\frac{1}{r}$

or minimize: $\frac{1}{\|w\|}$

or minimize: $w^T w$

Class 32: SVM: Soft Margins

Motivation: Data are not always linearly separable

→ when data are not linearly separable, misclassified points create false positives and false negatives

Slack Variables (ξ) Allowing Constraint Violations

- a slack variable turns optimization problem to turn an inequality constraint → equality constraint

→ hard margin constraint: $y_i(w^T x_j + b) \geq 1$

→ soft margin constraint: $y_j(w^T x_j + b) + \xi_j \geq 1$, $\xi_j \geq 0$

- allow some points inside the margin
- allow some points to be misclassified.

Incorperating Slack Variables

- Instead of ξ as free variables, we regularize them.

Objective: $\min \frac{1}{2}\|w\|^2 + C\sum_j \xi_j$

- L1 Penalty: $\xi_i \xi_i$, just sum of violations
- C controls the trade off
  - Large C → fewer violations (harder margin)
  - Small C → more flexibility (softer margin)

Case 0: $\xi = 0$, correctly classified outside margin

Case 1: $0 < \xi < 1$, inside margin correct side

Case 2: Misclassifred $\xi_j > 1$ (full mistake)

Constraints in matrix Form

1. Modified classification constraint:

$$1 - YXw - by - \xi \leq 0$$

2. Non negativity of slacks

$$-\xi \leq 0$$

Lagrangian:

$$L(w, b, \xi, \alpha, \beta) = \frac{1}{2}w^T w + \alpha^T(1 - YXw - by - \xi) + C\sum \xi_j - \beta^T \xi$$

Dual Formulation:

- Differentiating w.r.t the primal variables using KKT conditions shows: $\alpha + \beta = C$
- Final dual constraint $0 \leq \alpha_j \leq C$

Class 33: The Kernel Trick

Why Kernels: because datasets are not linearly seperable

We embed data into higher dimensional space where you can seperate it.

$(u_1, u_2) \to (u_1, u_2, u_1^2 + u_2^2)$

Issue: In explicit embedding computing $\psi(x)$ in high dimension is expensive

↳ solution: SVM only requires dot products between embedded points: $\phi(x_i)^T \phi(x_j)$

Kernel Trick

- Kernel K computes dot product without explicitly forming $\psi(x)$

  $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$

  - replace $x_i^T x_j$ in the dual SVM with $K_{ij} = K(x_i, x_j)$

- This bypasses the embedding and allows nonlinear SVM's

Gram Matrix

$K_{ij} = K(x_i, x_j)$

- Symmetric
- PSD
- Plugged directly into Dual

$\frac{1}{2} \alpha^T Y K Y \alpha - 1^T \alpha$

Common Kernals

- Linear $K(u, v) = u^T v$
- Polynomial $(u^T v + c)^p$
- gaussian $\exp\left(\frac{-\|u - v\|^2}{2\sigma^2}\right)$

Class 34: SVM Kernel Classification

Goal: How SVM's classify new points when using kernels

- After training SVM with kernel, must decide how to score a new vector and assign a class label.

SVM: Classification Score

- The score of the data vector is $z_j = x_j^T w + b$
  - ↳ classify using $a_j = \text{sign}(z_j)$

Using Lagrange Multipliers

- Primal lagrange equation, with equality

$$\vec{w} = X^T Y \vec{\alpha}$$

↳ sub into score:

$$z(x_j) = \sum_{i=1}^{m} \alpha_i y_i (x_i \cdot x_j) + b$$

Kernel Version (Key Formula)

- replace dot product with kernel: $(x_i \cdot x_j) \rightarrow K(x_i, x_j)$

↳ Final classification score: $z(x) = \sum_{i=1}^{m} \alpha_i y_i K(x_i, x) + b$

- only support vectors ($\alpha > 0$)
- classification $a(x) = \text{sign}(z(x))$

Gaussian (RBF) Kernel Behaviour

- gaussian kernel: $K(u,v) = e^{-\frac{\|u-v\|^2}{2\sigma^2}}$
- Close Points → Kernel ≈ 1
- Far Points → Kernel ≈ 0
- Classification becomes non linear curves
- SVM seperating ± 1 uses curved boundaries
- Adding one new point changes data boundary significantly

Class 33: Kernel Trick - In person notes.

Concept: instead of doing embedding use kernel function in a lower dimension

How can we use a kernel in SVM:

Ex

Kernel function of $u \cdot v^T + (u^T v)^2$

Embedding costs you, twice (space, time)

Recall Lagrange dual Function

$$\mathcal{L}(\vec{\alpha}, \vec{\beta}) = \frac{1}{2}\alpha^T Y x x^T Y \alpha + 1^T \alpha$$

Rewrite individual terms

$$Y x x^T \begin{bmatrix} y_1 & & \\ & y_2 & 0 \\ 0 & & y_n \end{bmatrix} \begin{bmatrix} & \\ & \end{bmatrix}$$

In vector terms, $\cdot Vx$

$$\mathcal{L}(\vec{\alpha}, b) = \frac{1}{2} Y K + 1^T \alpha$$

replacing LSDM

common Functions

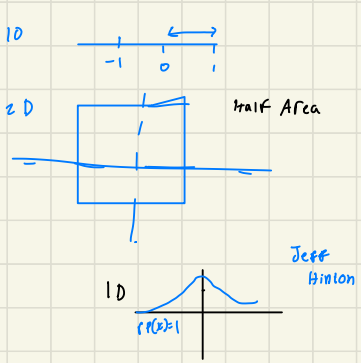
Linear: $K(\vec{v}, \vec{v}) = u^T \vec{v}$

Quadratic: $\psi(u, v) = (u^T v + c)^2$

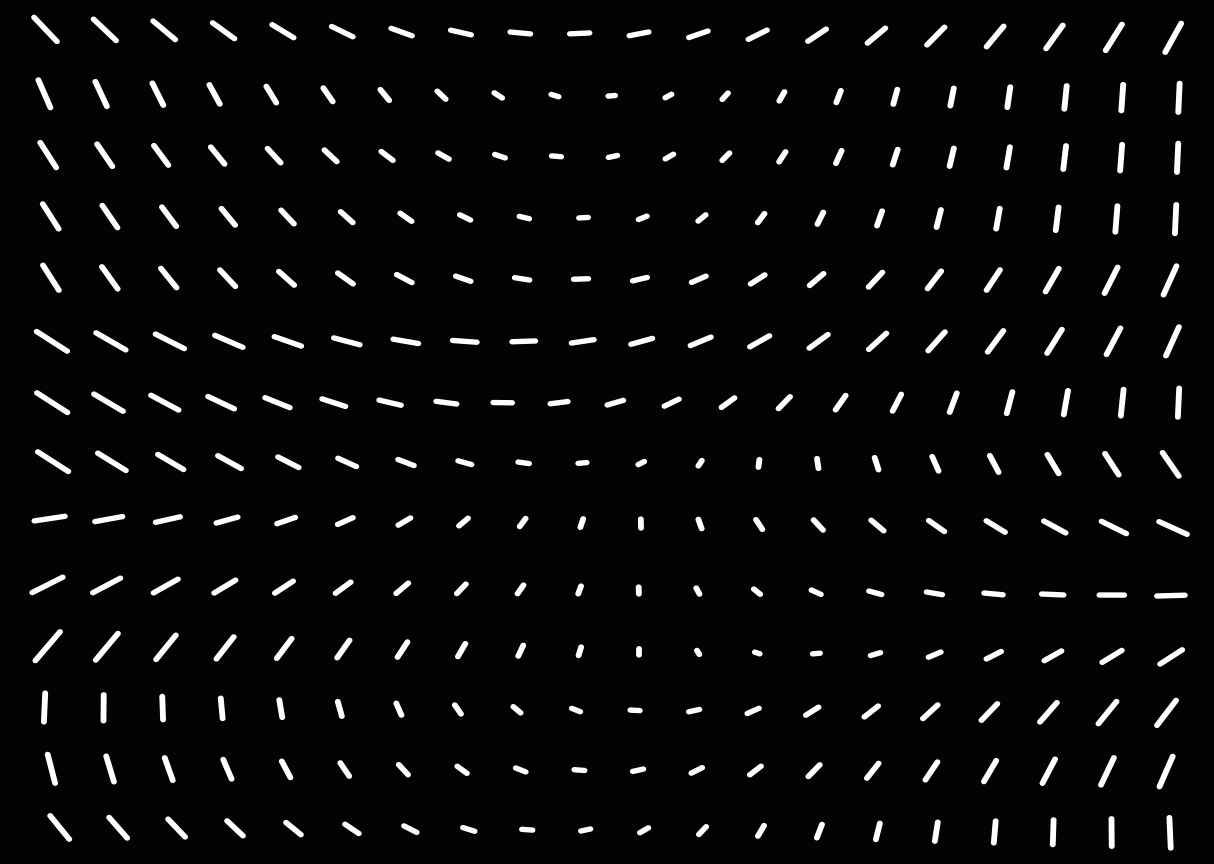
where $c \geq 0$ (works well up to 5 dimensions)

gaussian: $(u, v) = e^{-\delta \|\vec{u} - \vec{v}\|^2}$

Confirm data



Quiz 1 Prep



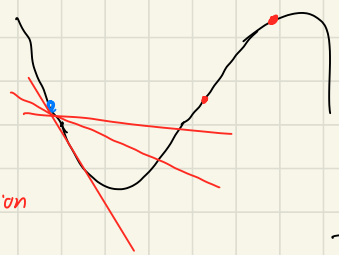
Quiz 1 Prep: Main concepts

- Scalar Calculus
- Derivatives, first and second
- Stationary points, convexity, second derivative

- Scalar Optimization
  - Taylor series approximations
  - Iterative methods:
    - ↳ Step size iteration $t_{k+1} = t_k - s f'(t_k)$
- Backtracking Armijo Method
  - ↳ α, β parameters

Armijo

Tangent line keeps coming flat, distance halfs backwards until condition is met.



to find step size

Fixed

- avoid oscillation in 3D
- Backtracking is a better way to find fixed stepsize

Taylor Series / Quadratic

input Point, first deriv, second deriv

↓ gives function that's easier to work with

Set that to 0, find min.

Hopefully that min is close to og function min

Partial Derivatives

rate of Chang of f, with respect to variables for direction $\vec{v}$

key relation to partials

$$D_v f = \nabla f \cdot v$$

∇f always points in direction of steepest ascent

Directional derivative in which direction you ask for

Practise 1: Prep for Online Quiz 1

Content: classes 1-5
(no hessian)

Main Themes

- Find first or second derivatives in matlab
- Compare 3 1-D optimization methods
- Determine the kind of stationary point in 2D (using Hessian)

Question 1: Scalar optimization

- given 3 functions, start at right endpoint
- 3 methods to implement:
  - ↳ Quadratic Taylor
  - ↳ Fixed step size GD:
  - ↳ Armijo Backtracking

Stopping rule: $|f'(t_k)| \leq 10^{-6}$

Deliver: Table with estimated minimizer and iteration count for each method x function combo.

1) plug in FD, SD, function itself

f = function
g = first derivative
h = second derivative

2) unify functions allow us to handle

fgl, given scalar t, returns two outputs

Fixed stepsize, Armijo functions expect obj + gradient

3) Compute steepest descent / backtracking estimates

w=b-a, a lot of setup

4) steepfixed function

- moves downhill along negative gradient with constant step

5) steepline function
- direction of +1 / -1, move opposite to slope
- user armijo condition (be under a certain line, otherwise keep on shrinking it)

Question 2: Stationary Points

- Finding and classifying stationary points using gradient and Hessian
  - ↳ Define 3 functions
    compute gradient and its square
    solves $\nabla f(w) = 0$ for real stationary points
    evaluate Hessian at each point, take eigen values
    - ↳ print table with point, eigen values and classification

Week 1: Homework 1

- Differentiate simple scalar functions
- Find stationary points (when first derivative = 0)
- Use second deriv test to classify points

5 Functions

$f_1(t) = 2t^3 - 9t^2 + 5$

$f_2(t) = t^3 - 12t + 8$

$f_3(t) = 3t^4 + 4t^3 - 72t^2 + 1$

$f_4(t) = t^4$

$f_5(t) = \frac{-t}{t^2+1}$

Bounds $t \in [-5, 5]$

1) $f_1(t) = 2t^3 - 9t^2 + 5$

$f_1'(t) = 6t^2 - 18t$

$0 = 6t^2 - 18t$

$0 = 6t(t-3)$

$t = 0 \quad t = 3$

Stationary points @ $(0, 0)$ and $(3, 0)$

$f_1''(t) = 12t - 18$

$f_1''(0) = -18 \therefore$ local max

$f_1''(3) = 18 \therefore$ local min

2) $f_2(t) = t^3 - 12t + 8$

$f_2'(t) = 3t^2 - 12$

$0 = 3(t^2 - 4)$

$t = \pm 2$

$(2, 0), (-2, 0)$

$f_2''(t) = 6t$

$f_2''(2) = 12 \therefore$ local min

$f_2''(-2) = -12 \therefore$ local max

$f_3(t) = 3t^4 + 4t^3 - 72t^2 + 1$

$f_3'(t) = 12t^3 + 12t^2 - 144t$

$0 = 12t(t^2 + t - 12)$

$t = 0 \quad t = 4 \quad t = -3$

$f_3''(t) = 36t^2 + 24t - 144$

$f_3''(0) = -144 \therefore$ local max

$f_3''(-3) = 108 \therefore$ local min

$f_3''(4) = 528 \therefore$ local min

$f_5(t) = \frac{-t}{t^2+1}$

$f_5'(t) = \frac{(-1)(t^2+1) - (2t)(-t)}{(t^2+1)^2}$

$0 = \frac{t^2 - 1}{(t^2+1)^2}$

$t = \pm 1$

$f_5''(t) = \frac{(2t)(t^2+1)^2 - (t^2-1)(2)(t^2+1)(2t)}{((t^2+1)^2)^2}$

$= \frac{2t(-t^2+3)}{(t^2+1)^3}$

$f_5''(-1) = -0.5$ local max

$f_5''(+1) = 0.5$ local min

Week 2: Homework 2

workflow:

1) gradient ∇f
2) Directional Derivative at Point w_0 in Vector V
3) One Step Steepest descent
   ↳ one step from w_0 in the direction $-\nabla f(w_0)$ with step size $s_0 = 1$
4) Solve $\nabla f(w) = 0$ for (w_1, w_2)

1) $f_1(w) = w_1^2 + \cosh(w_2)$

gradient is partial derivatives

$\nabla f(w) = (2w_1, \sinh(w_2))$

2) Directional Derivative in direction v

$D_v f(w) = \nabla f(w) \cdot v$

compute gradient at $\vec{w_0} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$

↳ $(-2, 0)$

$D_v = (-2, 0)(1, 1)$

$= -2$

Slope is in direction $v = (1, 1)$ is -2

3) one Steepest descent update

$w^{(1)} = w^0 - \alpha \nabla f(w^{(0)})$

$w^1 = (-1, 0) - 0.1(-2, 0) = (-0.8, 0)$

4) Stationary Point

$\nabla f(w) = (2w_1, \sinh(w_2)) = (0, 0)$

$2w_1 = 0$

$\sinh(w_2) = 0$

$w^* = (0, 0)$

Function $f_3(w) = (w_2 - w_1)^2 + w_1^3$

$w_0 = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$, $V = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

1) gradient

$\nabla f(w) = (2w_1 - 2w_2 + 3w_1^2, 2(w_2 - w_1))$

2) DD at w_0 $\nabla f(-1, 0) = (1, 2)$

along V

$\nabla f(-1, 0) \cdot V$

$= (1, 2) \cdot (1, 1)$

$= 3$

3) Steepest desc Step

$w^{(1)} = w^{(0)} - \alpha \nabla f(w^{(0)})$

$w^{(1)} = (-1, 0) - 0.1(1, 2)$

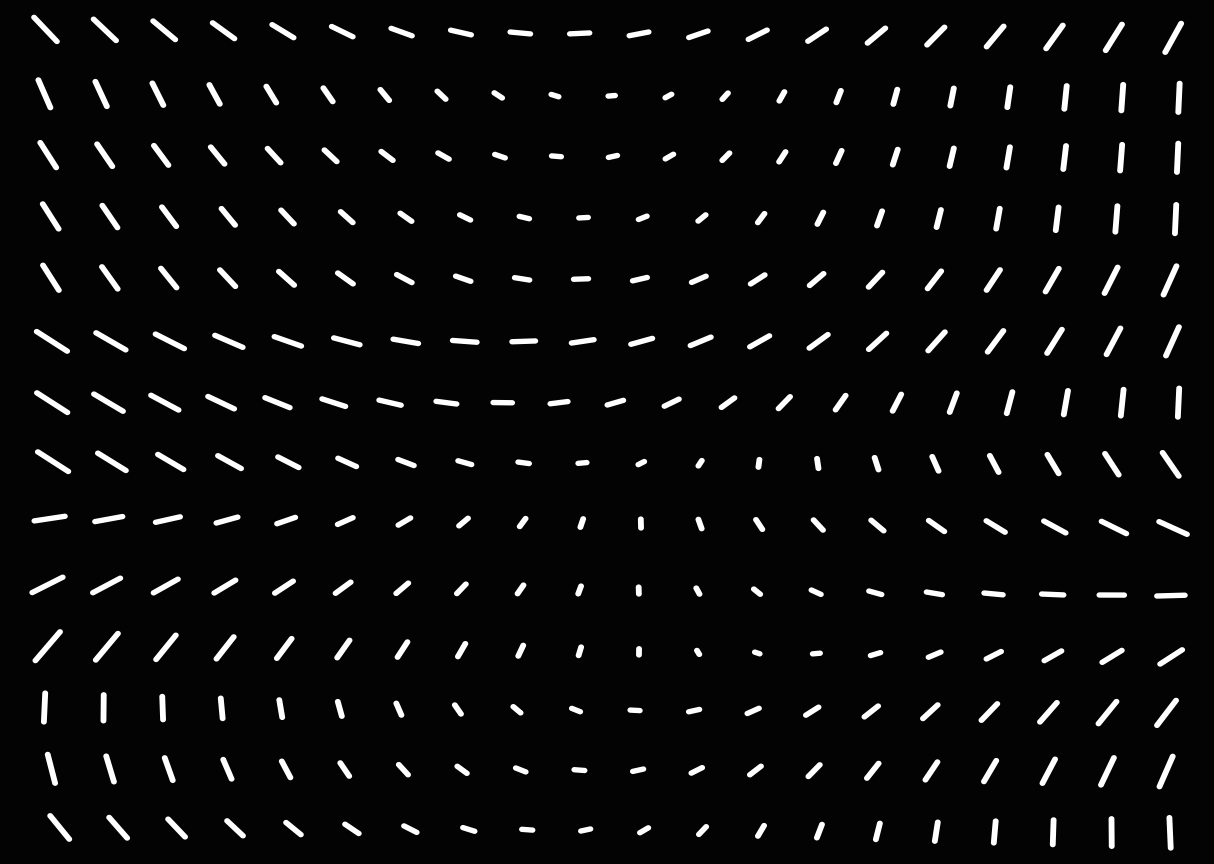
$= (-1.1, -0.2)$

4) Stationary Points

$w^* = (0, 0)$

Leave Hessian Stuff

Test 1 Prep



Class 11: Test 1 Prep

Basic Scalar Calculus

- Derivatives for basic functions
- Chain rule, product rule, quotient rule
- Find Stationary Points, Classify Point
- Convexity (second deriv test)

Scalar Optimization

- Iterations with fixed Stepsize (potential problems): $t_{k+1} = t_k - s f'(t_k)$
  - t_k: Initial guess
  - s: fixed Stepsize
  - $f'(t_k)$: deriv of function @ initial guess
- moves opposite to gradient
  - Too Big → Divergence
  - Too small → slow convergence
- Stepsize selection, backtracking, armijo method
  - ↳ Dynamically choose good Stepsize, armijo condition
  - ↳ Armijo Condition:

new function value after step

$$f(t_k + \beta^\gamma s_0 d_k) \leq f(t_k) + \alpha_k \beta^\gamma s_0 d_k$$

- t_k: initial guess
- β^γ: Shrink factor
- γ: current Shrink factor
- s_0: initial step size
- $d_k = -f'(t_k)$: descent direction
- $f(t_k) + \alpha_k \beta^\gamma s_0 d_k$: predicted using Slope margin

- Taylor Series approx, with/without derivatives → Explain grad desc
  - ↳ This helps Predict function behaviour near a Point

First order: $f(t) \approx f(t_0) + f'(t_0)(t - t_0)$

Second order: $f(t) \approx f(t_0) + f'(t_0)(t - t_0) + \frac{1}{2} f''(t_0)(t - t_0)^2$

$f''(t_0) > 0$: curves upward ∪ (min) (convex)

$f''(t_0) < 0$: curves down ∩ (max) (concave)

Vector Calc

- Partial derivatives, forms gradient
- Directional deriv $D_v F(\vec{w_0}) = \nabla F(\vec{w_0}) \cdot \vec{v}$
  - $\nabla F(\vec{w_0})$: gradient
  - $\vec{v}$: direction vector
- gradient Points in direction of steepest increase

Fall 2024 - Practise Test 1

1) $f(t) = \sin(\cos(t))$

$f'(t) = \cos(\cos(t))(-\sin(t))$

$= -\cos(\cos(t))\sin(t)$

2) $f_2(t) = \cos(t)$, $t_0 = 0$

Find quad approx?

$f(t) = \cos(t)$

$f(0) = 1$

$f'(t) = -\sin(t)$, $f'(0) = 0$

$f''(t) = -\cos(t)$, $f''(0) = -1$

$f(0)=1$, $f'(0)=0$, $f''(0)=-1$

Plug in

$h_2(t) = 1 + 0 + \frac{1}{2}(-1)t^2$

$= 1 - \frac{t^2}{2}$

$\cos(t) = 1$

$-\sin(0) = 0$

$-\cos(0) = -1$

$= 0 - \frac{0}{-1}$

$= 0$

3) For $f(t) = t^3 + 3t^2$ describe stationary point $t^* = 0$

$f'(t) = 3t^2 + 6t$

$= 3t(t+2)$

$t=0$ $t=-2$

$f''(t) = 6t + 6$

$f''(0) = 0 + 6$ ∴ Pos ∴ local min

4) $f(t) = t^3 - 12t + 11$, $t_1^* = -2$, $t_2^* = 2$

$f'(t) = 3t^2 - 12$

$f''(t) = 6t$

t_1^*: local max → Concave

t_2^*: local min → convex

5) $f(t) = t^2 - t$, $t_1 = 1$, $S = 1$

Find new estimate of minimizer

$t_{K+1} = t_K - S f'(t_K)$ $f'(t) = 2t - 1$

$t_{K+1} = 1 - 1(1)$

$t_2 = 0$

Point
Stepsize
# of Steps

Review

6) $f(t) = t^2 - t$, $\beta = \frac{1}{2}$, $t_1 = 1$, $S = 1$

Find stepsize $S = \beta^{s_0}$ after one step of armijo backtracking

$f(t) = t^2 - t$, $f'(t) = 2t - 1$

$f'(1) = 1$

$d_K = -f'(1) = -1$

$\alpha_K = \frac{1}{2}$

$t_{trial} = t_1 + S \cdot d_K$

$= 0$

RHS: $\alpha_K \cdot s \cdot d_K = -0.5$

$0 \leq -0.5$

False

Shrink to 0.5 $-0.25 \leq -0.25$

RHS: $t_{trial} = 1 + 0.5(-1) = 0.5$

LHS: $0 + \frac{1}{2} \cdot 0.5 \cdot (-1) = -0.25$

∴ $s_0 = 0.5$

Fall 2024 - Test 2A Practise

1) $f_1(\vec{w}) = \sin(w_1) + \cos(w_2)$, find $\nabla f(\vec{w})$

$\nabla f(\vec{w}) = \left[\frac{\partial f}{\partial w_1}, \frac{\partial f}{\partial w_2}\right]$

$\frac{\partial}{\partial w_1}\sin(w_1) \Rightarrow \cos(w_1)$

$\frac{\partial}{\partial w_2}\cos(w_2) = -\sin(w_2)$

$\nabla f(\vec{w}) = ?$

2) $f(\vec{w}) = w_1^2 + (w_2+1)^2$ $\quad \vec{w_0} = \begin{bmatrix}2\\1\end{bmatrix}$, $\vec{v} = \begin{bmatrix}-2\\-1\end{bmatrix}$

Find directional derivative $D_{\vec{v}} f(\vec{w_0})$

$\nabla f \cdot v$

$\nabla f = 2w_1 + 2(w_2+1)$ @ $\begin{bmatrix}2\\1\end{bmatrix}$

$\nabla f = [4, 4]$

$D_{\vec{v}} f(\vec{w_0}) = \begin{bmatrix}4\\4\end{bmatrix}\begin{bmatrix}-2\\-1\end{bmatrix}$

$= -8 - 4$

$= -12$

3) $f_3(\vec{w}) = (w_1+1)^2 + w_2^2$ has a stationary point at $w^* = \begin{bmatrix}-1\\0\end{bmatrix}$

Hessian is matrix of second order derivatives

? Clarity

			Second Deriv
at w^*	$= 2(w_1+1)$	$= 2w_2$	
	$= 2w_1 + 2$	$= 0$	
	$= 2(-1) + 2$		
	$= 2$		

$\nabla f = \begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}$

Eigenvalues: 2, 2 → Imply locally convex, min

All pos	min	convex
All neg	max	concave
mixed	Saddle	

∴ Positive Definite,

4) $f(\vec{w}) = w_1^2 + 2w_2^2$, $w_0 = \begin{bmatrix}1\\1\end{bmatrix}$, $s_0 = 1$

Find $\vec{w_1}$, that's in direction of steepest descent with stepsize $s_0 = 1$

$\vec{w_1} = w_0 - s_0 \cdot \nabla f(w_0)$ $\quad \vec{w_1} = \begin{bmatrix}-1\\-3\end{bmatrix}$

$\vec{w_1} = \begin{bmatrix}1\\1\end{bmatrix} - 1 \cdot \begin{bmatrix}2\\4\end{bmatrix}$

$\vec{w_1} = \begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}2\\4\end{bmatrix}$

5) $f(\vec{w}) = w_1^2 - w_2^2 + w_2^4$, $\vec{w_0} = \begin{bmatrix}-1\\-1\end{bmatrix}$

find $\vec{d}$ descent vector using newtons method

$d = -H^{-1}\nabla f(w_0)$

$d = -\left[\nabla^2 f(w_0)\right]^{-1}\nabla f(\vec{w_0})$

$\nabla f = \left[2w_1, -2w_2, +4w_2^3\right]$

$\nabla f = \left[-2, -2(-1) + 4(-1)^3\right]$

$\nabla f = [-2, -2]$

$\nabla f^2 = \begin{bmatrix}2 & 0\\0 & -2+12w_2^2\end{bmatrix}$

at $w_2 = -1$ $= \begin{bmatrix}2 & 0\\0 & 10\end{bmatrix}$

Inverse $\left[\nabla^2 f\right]^{-1} = \begin{bmatrix}1/2 & 0\\0 & 1/10\end{bmatrix}$

$= -\begin{bmatrix}1/2 & 0\\0 & 1/10\end{bmatrix}\begin{bmatrix}-2\\-2\end{bmatrix} = \begin{bmatrix}1\\0.2\end{bmatrix}$

6.) $f(w) = (w_1+1)^2 + w_2^2$

Both terms are quadratic

Both are convex ∴ this general function is convex

Use Hessian

Fall 2024 - Test B - Practise

1) $f(w) = \cos(w_1^2) + \sin(w_2^2)$

$\nabla f(w) = -2\sin(w_2) + \cos(w_1)$

2) $f(\vec{w}) = (w_2+1)^3 + w_1^3$, $\vec{w_0} = \begin{bmatrix}1\\1\end{bmatrix}$, $\vec{v} = \begin{bmatrix}-1\\-1\end{bmatrix}$

$\nabla f \cdot v$

$3(w_2+1)^2 + 3w_1^2$

$3(2)^2 + 3$

$12, 3$ $\quad \begin{bmatrix}12\\3\end{bmatrix} \begin{bmatrix}-1\\-1\end{bmatrix}$

$= -12 - 3$

$= -15$

3) $f(\vec{w}) = w_1^2 - (w_2+1)^2$, Stationary point at $w^* = \begin{bmatrix}0\\-1\end{bmatrix}$

Find eigenvalues, describe convexity at point w^*.

$f'(w) = 2w_1 - 2(w_2+1)$

$f''(w) = 2 - 2$

$H = \begin{bmatrix}2 & 0\\0 & -2\end{bmatrix}$ $\therefore$ Saddle Point

Curves upward then downward

Convex, then concave

If variables are left in Hessian plug in your point.

4) $f(w) = 2w_1^2 + w_2^2$, $\vec{w_0} = \begin{bmatrix}1\\1\end{bmatrix}$, $s_0 = 1$

$t_{k+1} = t_k - d_k s_k$ gradient

$t_{k+1} = \begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}4\\2\end{bmatrix}\begin{bmatrix}1\\1\end{bmatrix}$

$d = \nabla f = \begin{bmatrix}1\\1\end{bmatrix} - \begin{bmatrix}4\\2\end{bmatrix}$

$= 4w_1, 2w_2 = \begin{bmatrix}-3\\-1\end{bmatrix}$

$= 4, 2$

5) $f(w) = w_1^2 + 3w_2^2 - w_2^4$, $\vec{w_0} = \begin{bmatrix}-1\\-1\end{bmatrix}$

Find desc vector using N.M

using Hessian, which requires invertible / positive definite

$\nabla f = 2w_1 + 6w_2 - 4w_2^3$

$\nabla^2 f = 2 + 6 - 12w_2^2$

$H = \begin{bmatrix}2 & 0\\0 & 6-12w_2^2\end{bmatrix}$ Plug in

$= \begin{bmatrix}2 & 0\\0 & -6\end{bmatrix}$ Newtons Method

Assume Positive Definite

6) $f(w) = w_1^2 - (w_2+1)^2$

$= 2w_1 - 2(w_2+1)$

$= 2 - 2$

$\begin{bmatrix}2 & 0\\0 & -2\end{bmatrix}$ $\therefore$ Saddle Point

The function is convex then concave

2019 Test 1A Practise

1) $f(t) = te^t$

Stationary Point at $t = -1$

$f'(t) = e^t \cdot (t)e^t$

$e^t[t+1]$

$t=0 \quad t=-1$

$f''(t) = e^t + [e^t + (t)e^t]$

$f''(-1) = 0.367 \quad \therefore$ Pos

$\therefore$ Min, Convex

Answer: A

2) $f(t) = \cosh(t) = \frac{e^t + e^{-t}}{2} \qquad t_1 = 1$

Using Quadratic approx to find a t_2

$t_2 = t_1 - \frac{f'(t_1)}{f''(t_2)}$

$f'(t) = \sinh(t)$

$f''(t) = \cosh(t)$

$t_2 = 1 - \frac{1.175}{1.543}$

$t_2 = 0.2385$

Answer A

$t_{k+1} = t_0 - dk\,sk$

3) trial point

$f(t_1 + \beta^r sd) \leq f(t_1) + \frac{\alpha}{2}\beta^r s |f'(t_1)|^2 d$

To find number of r backtracking steps:

1) Find direction and slope

$f(t) = t^2 - 1$

$f'(t) = 2t$

$f'(1) = 2, \quad d = -2$

Trial Point: $t = 1 + 1 \cdot 2 \cdot -2$

$t = -3$

$f(-3) = 8$

$8 \leq -4$ ✗

2nd Trial

$f(1 + 0.5 \cdot 2 \cdot -2)$

$f(-1) = 0$

RHS

$0 + \frac{0.5}{2} \cdot 0.5 \cdot 2 \cdot 4 \cdot -2$

$= -2$

$0 \leq -2$ ✗

Third Trial

$f(1 + 0.5^2 \cdot 2 \cdot -2)$

$f(0) = -1$

RHS

$0 + \frac{0.5}{2}(\frac{1}{2})^2 \cdot 2 \cdot (4) \cdot -2$

$= -1$

$\therefore \quad r = 2$

4) $f(\vec{w}) = \frac{1}{3}w_1^3 + w_1 w_2^2 + w_1 w_2 + 5$

$H = \begin{bmatrix} 2w_1 & 2w_2 + 1 \\ 2w_2 + 1 & 2w_1 \end{bmatrix}$

$H = \begin{bmatrix} 2(0.5) & 2(-0.5) + 1 \\ 2[-0.5] + 1 & 2(0.5) \end{bmatrix}$

$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \therefore$ local minimum

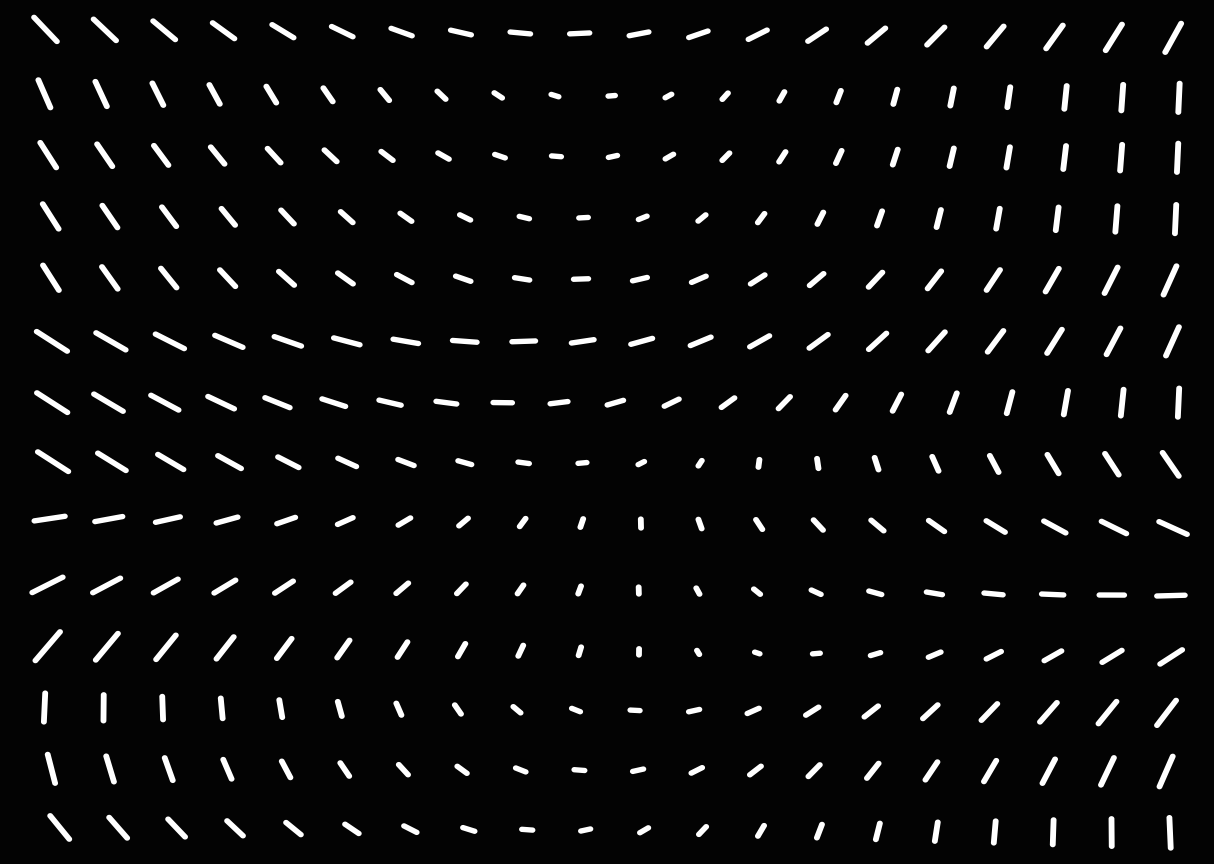
5) gradient of $\tanh(\vec{w} \cdot \vec{x})$

Chain rule: $\frac{1}{\cosh^2(w \cdot x)} \cdot \vec{x}$

6) $f(\vec{w}) = \frac{1}{3}w_1^3 + w_1 w_2^2 + w_1 w_2 + 5$

$\vec{w_1} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Quiz 2 Prep



Key concepts - Practise 2

- Find stationary points for 2 variables
- Hessian to classify
- Iterative solvers; second order
- Compare convergence behaviour
- Formulate and solve nonlinear squares problem

Part 1: Stationary points

- 3 Functions
  - ↳ Compute Hessians, gradients
    Solve $\nabla f = 0$ to find stationary points
    Classify, visualize w mesh / contour

1) Set gradient to 0 to find stationary points

$$\nabla f = 0$$

Ex

$$f(w) = 2w_2^3 - 6w_2^2 + 3w_1^2 w_2$$

$$\frac{\partial f}{\partial w_1} = 6w_1 w_2, \quad \frac{\partial f}{\partial w_2} = 6w_2^2 - 12w_2 + 3w_1^2$$

$$0 = 6w_1 w_2$$
$$0 = 6w_2^2 - 12w_2 + 3w_1^2$$

2) Hessian

$$H = \begin{bmatrix} 6w_2 & 6w_1 \\ 6w_1 & 12w_2 - 12 \end{bmatrix}$$

$$\text{At } (0,0) = \begin{bmatrix} 0 & 0 \\ 0 & -12 \end{bmatrix}$$

$$\text{At } (0,2) = \begin{bmatrix} 12 & 0 \\ 0 & 12 \end{bmatrix}$$

Both eigenvalues > 0 $\therefore$ min
Both eigenvalues < 0 $\therefore$ max
one pos, one neg $\therefore$ saddle
any 0 → inconclusive

Part 2: Vector Optimization

- numerically minimize each function

Fixed stepsize steepest Descent

- update rule: $w_{k+1} = w_k - s \nabla f(w_k)$

Backtracking line search steepest descent

update rule: $w_{k+1} = w_k - s_k \nabla f(w_k)$

↳ start with $s_0 = 0.1$, shrink by beta $= 0.5$

↳ converges faster

Damped Newtons Method

- updated Rule: $w_{k+1} = w_k - s_k H^{-1}(w_k) \nabla f(w_k)$
- uses Hessian curvature info

For each function; must report the minimizer w^*, number of iterations

Part 3: Non Linear Squares

- estimation location of gps reciever
  ↳ given: 3D coordinates of 6 satellites
  measured distance between each satellite and reciever

$$diw = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}$$

Content overview

Vector calc

- Partials, gradient, Directional deriv
  $\frac{\partial f}{\partial w}$ ∇f $D_{\vec{v}} f = \nabla f \cdot \vec{v}$

- Jacobian matrix, vector valued function
  $J = m \times n$
  $J_{ij} = \frac{\partial f_i}{\partial w_j}$
  Jacobian gives matrix, best linear approximation to a vector valued function.
  ↳ $f(w) = \begin{bmatrix} w_1^2 + w_2 \\ \sin(w_1 w_2) \end{bmatrix}$ Jacobian $= \begin{bmatrix} 2w_1 & 1 \\ w_2 \cos(w_1 w_2) & w_1 \cos(w_1 w_2) \end{bmatrix}$
  → can do mappings and plug ins.

- Hessian $\nabla^2 f$, second partials, eigenvalues = convexity

Steepest Descent

- Dese direction, steepest descent $= -[\nabla f(w)]^T$
- Fixed stepsize $w_{k+1} = \vec{w_k} + s\vec{d_k}$
- Back tracking

Newtons Method

- Helps find minimum by using second order information (∇ grad, ∇^2 Hessian)
  - must be pos definite
- Damped newtons method, adds scaling factor, (newton steps too agressive)
- Iterate until convergence

Non linear Least Squares

- useful when model is nonlinear in params
- minimizes sum of squared residuals
- reduce $\frac{1}{2}\|\vec{r}(\vec{w})\|^2$

1) residual vector

$$\vec{r}(w)=\begin{bmatrix} r_1(\vec{w}) \\ r_2(\vec{w}) \\ \vdots \\ r_m(\vec{w}) \end{bmatrix}$$

model output − actual value

2) linearized residual, helps linearize

$$a_i(\vec{w}) = r_i(\vec{w}_0) + \nabla r_i(\vec{w}_0)^T(\vec{w}-\vec{w}_0)$$

3) linearized objective function:

$$f(\vec{w}) = \frac{1}{2}\|\vec{a}(\vec{w})\|^2$$

iteratively update.

2 Main NLS Algorithms

Gauss Newton

$$w_{k+1} = w_k + \left[J_k^T J_k\right]^{-1} J_k^T \vec{r}(w_k)$$

J_k is the jacobian matrix at point w_k

$$\text{Hessian} = J_k^T J_k$$

Levenberg marquat method

$$w_{k+1} = w_k + \left[J_k^T J_k + \pi I\right]^{-1} J_k^T \vec{r}(w_k)$$

Adds damping factor $\pi \geq 0$ to stabalize updates to prevent 0 eigenval problem

Example

$r_1(w) = w^2 - 2$, $r_2(w) = \sin(w) - 0.5$

objective function, $f(w) = \frac{1}{2}(r_1(w)^2 + r_2(w)^2)$

residual vector $\vec{r}(w) = \begin{bmatrix} w^2-2 \\ \sin(w)-0.5 \end{bmatrix}$

$$\text{Jacobian} = \begin{bmatrix} 2w \\ \cos w \end{bmatrix}$$

$$J(w_0) = \begin{bmatrix} 2.0 \\ 0.5403 \end{bmatrix}$$

$$r(1.0) = \begin{bmatrix} 1.0^2 - 2 \\ \sin(1.0) - 0.5 \end{bmatrix} = \begin{bmatrix} -1.0 \\ 0.3415 \end{bmatrix}$$

$$w_1 = w_0 + (J^T J)^{-1} J^T \vec{r}$$

$$w_1 = w_0 + \left(\frac{-1.8154}{4.2929}\right)$$

$$w_1 = 0.5771$$

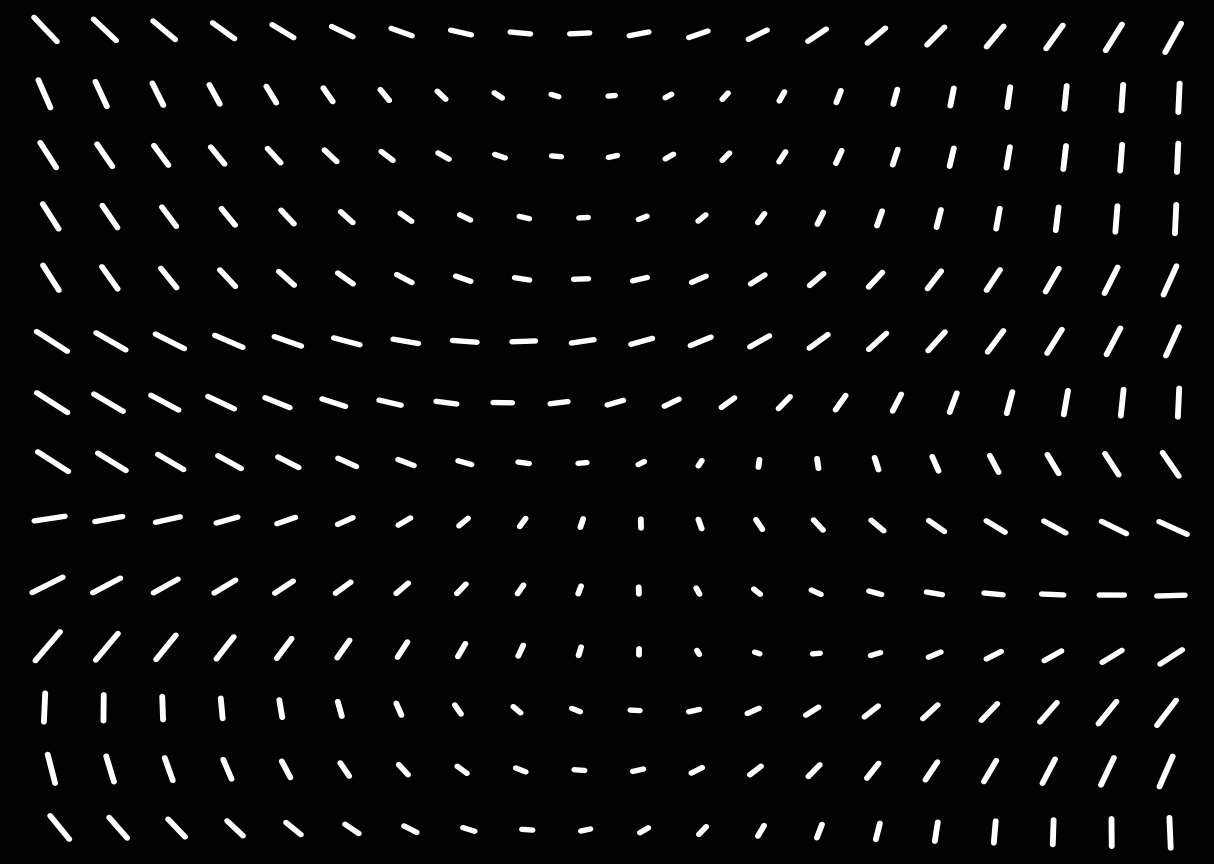
Level curve:

- multiple curves, contour plot

  $z = f(x,y)$

- concentric circles

Test 2 Prep



Test 2 Prep

Vector Calc

- $f(w1, w2) = 3w_1^2 + 2w_1 w_2 + w_2^3$

$\frac{df}{dw_1} = 6w_1 + 2w_2$

$\frac{df}{dw_2} = 2w_1 + 3w_2^2$

→ Plug in $(1, 2)$

$= 10, 14$

Directional Deriv

$DD = \nabla f \cdot v$

$f_2(\vec{w}) = 3(w_2 + 1)^2 + 2w_1$

Plug in $(1, 1)$

$= (12, 2) \cdot (-1, -1)$

$= -14$

Gradient 1 form

$f(w_1, w_2) = w_1^2 + 5w_1 w_2 + 4w_2^2$

$\frac{\partial f}{\partial w_1} = 2w_1 + 5w_2$

$\frac{\partial f}{\partial w_2} = 5w_1 + 8w_2$

$-\nabla f$ → steepest Descent

Jacobian

$\begin{bmatrix} \partial f_1/\partial w_1 & \partial f_1/\partial w_2 \\ \partial f_2/\partial w_1 & \partial f_2/\partial w_2 \end{bmatrix}$

Jacobian contains derivatives of vector valued function

Each row is gradient of residual

$f_1(w) = w_1^2 + w_2$

$f_2(w) = \sin(w_1 w_2)$

$\partial f_1/\partial w_1 = 2w_1$

$\partial f_1/\partial w_2 = 1$

$\partial f_2/\partial w_1 = \cos(w_1 w_2) \cdot w_2$

$\partial f_1/\partial w_2 = \cos(w_1 w_2) \cdot w_1$

At $w = (1, \pi)$ $= \begin{bmatrix} 2 & 1 \\ -\pi & -1 \end{bmatrix}$

$g_i(w) = \|a_i - w\|$

For each row $J_i = \frac{a_i - w^T}{\|a_i - w^T\|}$

Each residual: $\|a_i - w\|_2$

direction vector / length of vector

$A = [3, 12, 1]$

$b_0 = 2$

$J(b) = \begin{bmatrix} \frac{1}{1} & \frac{10}{10} & \frac{-1}{1} \\ [1, & 1, & -1] \end{bmatrix}$

$J_i = \frac{a_i - w}{\|a_i - w\|}$

2D Example

$= a_1 - \vec{w}$

$= [4-1, 3-1] = [3, 2]$

$J_1 = \frac{[3, 2]}{\sqrt{13}}$

$J_i = \frac{a_i - b_0}{|a_i - b_0|}$ One D

Gauss-Newton

residual $r(w) = [w-2, w-3]$

$J(w) = [1, 1]$

From $w_0 = 0$

$w_{k+1} = w_k + [J^T J]^{-1} J^T r(w_k)$

$r = [-2, -3]$

$0 + [2]^{-1} (-2) + (-3)$

$= 0 + \frac{-5}{2}, = -2.5$

3 by 2 · 2 by 5

valid

Resultant

Hessian

$f(\vec{w}) = w_1^2 + w_2^4$

@ $[1, 0]$

$H = 2w_1 + 4w_2^3$

$= 2, 12w_2^2$

$\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$

Hessian is not positive definite ∴ newtons method can't be applied

Hessian

$f(w_1, w_2) = w_1^2 + 4w_1 w_2 + 4w_2^2$

$H = \nabla F$ $\frac{\partial f_1}{\partial w} = 2w_1 + 4w_2$

$\frac{\partial f_2}{\partial w_2} = 4w_1 + 8w_2$

$\begin{bmatrix} 2 & 4 \\ 4 & 8 \end{bmatrix}$ Min Strictly U ∴ convex

$\nabla F = 0$

$0 = \begin{matrix} 6w_1 + 2w_2 \\ 2w_2 + 2w_2 \end{matrix}$

$(0, 0)$

$f(w_1, w_2) = \log(1 + w_1^2) + e^{w_2}$

Convexity

$\log(1 + w_1^2)$ is convex from second deriv test

Optimization For Steepest Descent

- minimizing function by moving in direction of negative gradient. $\vec{d} = -\nabla f(w)$

1) $\vec{w}, x, b =$

Steepest desc vector: $\vec{d} = -b \cdot \vec{x} \cdot \vec{w}$

$\vec{d} = -b \cdot \text{avg } x$

Levenburg Example

5) $A = \begin{bmatrix} 1 \\ 4 \\ 8 \end{bmatrix}$, $t_0 = 2$, $\pi = 1$

$$b_1 = b_0 + (J^T J + \pi I)^{-1} J^T r$$

1) Compute r: $\begin{bmatrix} 1 & -2 \\ 4 & -2 \\ 8 & -2 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ 6 \end{bmatrix}$

2) Jacobian: $\frac{a_i - t}{|a_i - t|} = \left[\frac{-1}{1}, \frac{1}{1}, \frac{1}{1}\right]$

$J = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$

$J^T J = 3$

$J^T r = \begin{bmatrix} -1 \\ 2 \\ 6 \end{bmatrix} [-1 \ 1 \ 1] = 9$

3)

$$= t_0 + \frac{J^T r}{J^T J + \pi}$$

$$= 2 + \frac{9}{3+1}$$

$$= 3.75$$

$A = \begin{bmatrix} 0 \\ 5 \\ 6 \end{bmatrix}$ $t_0 = 3$, $\pi = 2$

$w_{k+1} = 3 + [J^T J + 2]^{-1} J^T r$

$r = \begin{bmatrix} 0 & -3 \\ 5 & -3 \\ 6 & -3 \end{bmatrix}$ $r = \begin{bmatrix} -3 \\ 2 \\ 3 \end{bmatrix}$

$J = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$ $\begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} [-1 \ 1 \ 1]$

$$= 3 + [3 + 2]^{-1} [-1 \ 1 \ 1] \begin{bmatrix} -3 \\ 2 \\ 3 \end{bmatrix}$$

$$= 3 + \frac{1}{5} \cdot 8 = 4.6$$

$A = \begin{bmatrix} 2 \\ 7 \\ 10 \end{bmatrix}$, $t_0 = 5$, $\pi = 3$

$r = \begin{bmatrix} 2 & -5 \\ 7 & -5 \\ 10 & -5 \end{bmatrix} = \begin{bmatrix} -3 \\ 2 \\ 5 \end{bmatrix}$, $\pi = 3$

$J = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$ $= 5 + [3 + 3]^{-1}$

$5 + \frac{1}{6} \cdot 10$

$= 6.\overline{6}$

NLS

$r(w) = [r_1(w), r_2(w), \ldots r_m(w)]$

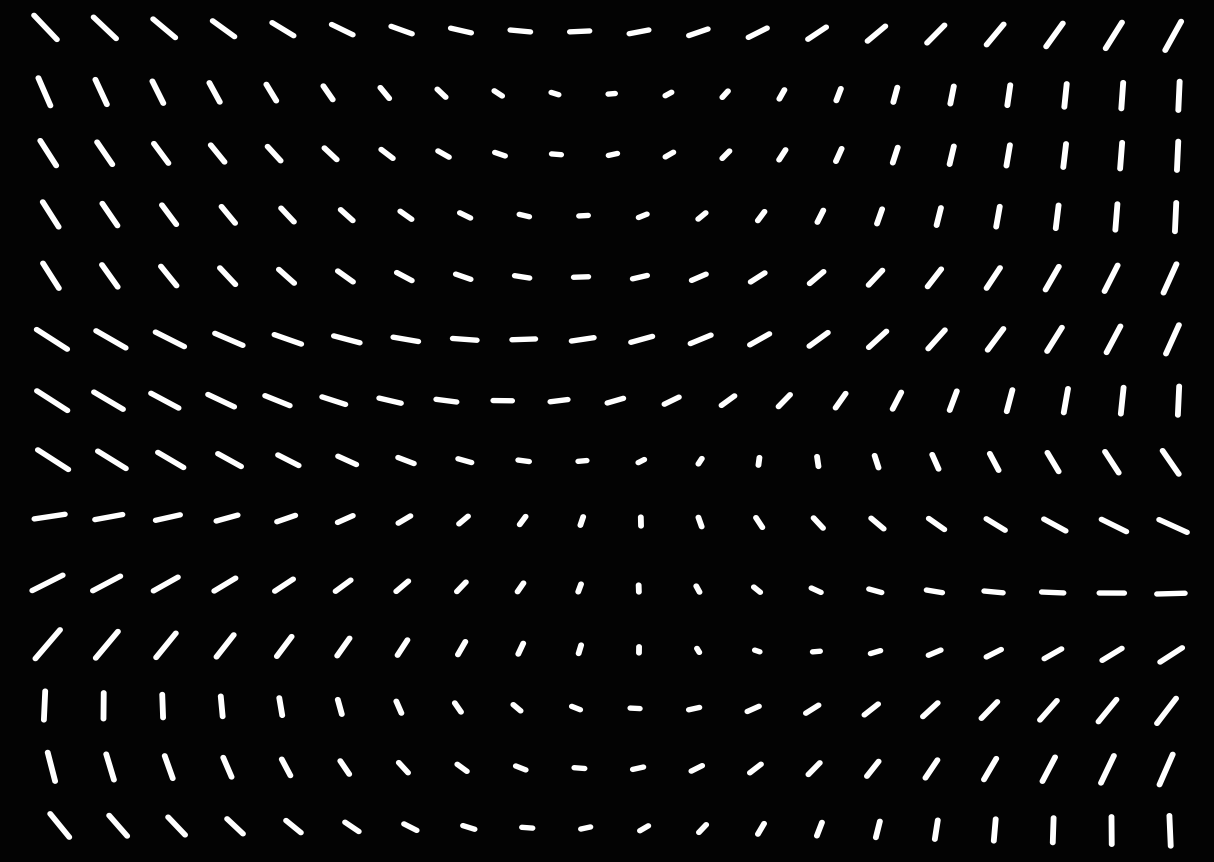
$F(w) = \frac{1}{2} \|r(w)\|^2$

Descent Direction: $d = -\nabla f(w_0)$

$w_{k+1} = w_k + \alpha_k d_k$

Newtons Method: $d_k = -H^{-1} \nabla f(w_k)$

Quiz 3/ Test Prep



Unit 3 - Important Concepts

Single Artificial Neuron

- Smallest building block of Neural net
- Takes input vector → applies linear function → passes through activation function → Produces output score
- data vector is transpose of observation
- Neurons include bias term. $[x\ 1]$
- for classification, labels are either 0 or 1, because of logistic activation
- Linear response is $v(\vec{w}) = [x\ 1]\, w$

Ex

$v = x \cdot w$ given $\vec{w} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$ $A = \begin{bmatrix} 2 & 3 \\ 0 & 2 \end{bmatrix}$

$$\begin{bmatrix} 2 & 3 & 1 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \rightarrow \vec{v} = \begin{bmatrix} 0.00 \\ -1.00 \end{bmatrix}$$

- Activation function is ϕ

$$\phi_L(v) = \frac{1}{1+e^{-v}} \rightarrow \text{Logistic}$$

output is between 0 and 1

- ReLU activation

$$\phi_R(v) = \max(0, v)$$

for hidden layers

Outputs 0 if input is negative

Outputs v if input is Positive

- derivative of activation function

$$\psi(v) = \phi'(v)$$

$$\phi'(v) = \phi_L(v)(1 - \phi_L(v))$$

Ex $\phi(v) = 0.4$, $\psi = 0.4(1-0.4) = 0.240$

- residual error: $r(\vec{w}) = y - \phi(\vec{w})$

(actual - predicted)

- Objective $f(w) = \frac{1}{2} r(w)^2$

Always reduce residuals

- Learning rate η is like weight updater:

$$\vec{w}_{new} = \vec{w}_{old} + \eta \vec{d}$$

Neural Networks with hidden layers

Layer 1: Input

$$x = [x_1, x_2]$$

$$[x\ 1] = [x_1\ x_2\ 1]$$

Layer 2: Hidden Layer (new part)

multiple neurons → weight matrix

2 hidden neurons → 2 weight columns

Layer 3 - output neuron

recieves input from hidden layer, weight vector: $w^{(3)}$

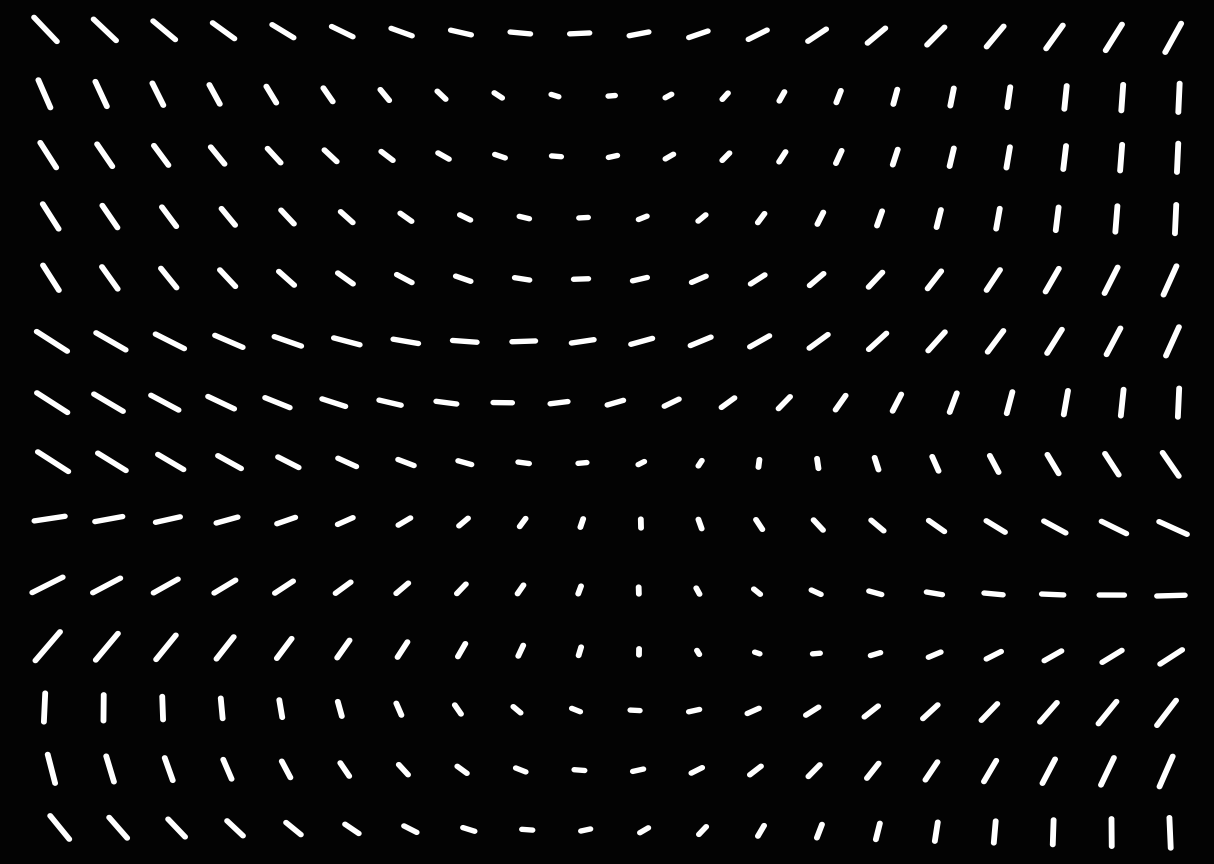
$$x^{(3)} = [z_1^{(2)}\ z_2^{(2)}\ 1]$$

- back prop is the same

  $b = -r$

- just extends single neurons to vectors/matrices

Quiz 4 Prep



Key Topics for Quiz 4

Lagrange Multipliers and Constrained optimization

- goal: Solve constrained Optimization problem
  to minimize function Subject to constraints

- if you have constraints like $g(x) = 0$ or $h(x) \leq 0$
  can't take deriv, set equal to 0 like unconstrained problems

- Combine objective and constraints into 1 expression
  This is the lagrangian and uses lagrange multipliers

Ex Equality constraint

Lagrangian: $L(x, y, \nu) = f(x, y) + \nu \cdot g(x, y)$

↓ lagrange multiplier → how sensitive the optimal solution is to changes in constraint

Minimize $f(x,y)$
Subject to $g(x,y) = 0$

Solve $\nabla_{x,y} L = 0$ and $g(x, y) = 0$

ν: large means constraint heavily affects solution

Ex Inequality Constraint

minimize $f(x)$
subject to $h(x) \leq 0$

Lagrangian: $L(x, \pi) = f(x) + \pi \cdot h(x)$

But: we require $\pi \geq 0$, $\pi \cdot h(x) = 0$

not all inequality constrains are active at solution

$\pi \cdot h(x) = 0$: condition forces 2 cases

Case 1: Constraint is active
its "tight" $x - 5 \leq 0$, $x = 5$
π can be non zero

Case 2: constraint is inactive
Strictly within bounds
$x - 5 \leq 0$, solution is $x = 3$
$h(x) = -2 < 0$
Then $\pi = 0$, because constraint does not affects solution

Example

min $f(x) = x^2$, subject to $x \geq 2$

$h(x) = 2 - x \leq 0$

1) Langrangian:

complementary slackness

$$L(x, \pi) = x^2 + \pi(2-x)$$

primal feasability

KKT conditions

1.) Stationarity: $\frac{dL}{dx} = 2x - \pi = 0$ ①

2.) Primal feasibility: $2 - x \leq 0$

3.) Dual feasibility: $\pi \geq 0$

4.) Complementary slackness $\pi(2-x) = 0$ ②

From ① $\pi = 2x$

Sub into ②

$2x(2-x) = 0$

So, $x = 2$ or $x = 0$

$x = 2$: active constraint

$x = 0$: constraint violation

Check $\pi = 2x = 4 \geq 0$ ✓

Optimal $x = 2$

Stationarity: critical point w.r.t x is zero

Primal feasibility: original constraint checker

Dual feasibility: $\pi \geq 0$

Complementary slackness: can be active/inactive $(\pi_i \cdot h_i(x) = 0)$

If objective is quadratic: $f(x) = \frac{1}{2} x^T Q x + c^T x$

constraints are linear equalities: $Ax = b$

KKT Matrix System: $\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \nu \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$

- gives optimal x and multipliers

Ex

$Q = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$

$c = \begin{bmatrix} -2 \\ -5 \end{bmatrix}$

$A = \begin{bmatrix} 1 & 1 \end{bmatrix}$

$b = 3$

$$\begin{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} & \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ \begin{bmatrix} 1 & 1 \end{bmatrix} & 0 \end{bmatrix} \begin{bmatrix} x \\ \nu \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} -2 \\ -5 \end{bmatrix} \\ 3 \end{bmatrix}$$

1) Minimizing $\frac{1}{2}(2x_1^2 + 2x_2^2) - 2x_1 - 5x_2$

2) Subject to $x_1 + x_2 = 3$

$$x^T Q x = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 4 & 1 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 4x_1^2 + 2x_1 x_2 + 3x_2^2$$

How to make sense of this

KKT Matrix System

→ used when Quadratic + linear term and linear equality constraints

Ex

$$\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \nu \end{bmatrix} = \begin{bmatrix} -c \\ b \end{bmatrix}$$

$Q = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$

$A^T = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$A = \begin{bmatrix} 1 & 1 \end{bmatrix}$

$c = \begin{bmatrix} -2 \\ -5 \end{bmatrix}$

$b = 3$

$$\begin{bmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \\ 3 \end{bmatrix}$$

Lin alg this out

$2x_1 + \nu = 2$

$2x_2 + \nu = 5$

$x_1 + x_2 = 3$

$\nu = 2 - 2x_1$

$2x_2 + (2 - 2x_1) = 5$

$x_2 = 3 - x_1$

$2(3 - x_1) - 2x_1 = 3$

$x_1 = \frac{3}{4}$

$x_2 = \frac{9}{4}$

$\nu = \frac{1}{2}$

$\nu = 2 - 2x_1$

$2x_2 + (2 - 2x_1) = 5$

$2x_2 - 2x_1 = 3$

$x_1 = 3 - x_2$

$2x_2 - 2(3 - x_2) = 3$

$2x_2 - 6 + 2x_2 = 3$

$\frac{4x_2}{4} = \frac{9}{4}$

$x_2 = \frac{9}{4}$

Primal Problem - the one you start with

Dual Formulation with Linear Constraints

- write in terms of lagrange multipliers ν

1.) $f(w)$, s.t $A\vec{w} = \vec{b}$

$L(w, \nu) = f(w) + \nu^T(Aw - b)$

↓ vector of lagrange multipliers

2. take w.r.t $\vec{w}$, express as function of ν

$g(\vec{\nu}) = \inf L(w, \nu)$

$g(\nu)$ dual function → always concave

maximizing

Ex

$\min \frac{1}{2}\|w - c\|^2$ s.t $a^T w = b$

$c = \begin{bmatrix} \frac{1}{2} \\ \end{bmatrix}$

$a = \begin{bmatrix} \frac{3}{4} \\ \end{bmatrix}$

$b = 10$

① $L(w, \nu) = \frac{1}{2}\|w - c\|^2 + \nu \cdot (a^T w - b)$

$= \frac{1}{2}(w - c)^T(w - c) + \nu \cdot (a^T w - b)$

② w.r.t to w = $w - c + \nu a = 0$

$w = c - \nu a$

3) Plug $w = c - \nu a$ into langrangian:

objective: $\frac{1}{2}\nu^2(25)$

Constraint: $\nu(1 - 25\nu)$

4) Simplify → $g(\nu) = -\frac{25}{2}\nu^2 + \nu$

Then find ν

Then get w

w.r.t gradient

$w^T w = 2w$

$a^T w = a$

$Aw = A^T$

$\frac{1}{2}\|w - c\|^2 = w - c$

2 ways to solve quadratic optimization problem with linear equality constraints:

1) Solve with KKT (primal)
2) Derive, Solve dual problem

B - Dual matrix. Problem becomes how to maximize: $-\frac{1}{2}\lambda^T B\lambda - \lambda^T d$

- Always concave

- KKT conditions must be satisfied to be optimal solution

Basically, given any $f(\vec{w})$ if it has equality or inequality constraints for any optimal points there is $\lambda \geq 0$, ν to satisfy:

1) Stationarity → finding gradient = 0
2) Primal feasibility → must satisfy original constraints $g_i(w^*) \leq 0$, $h_j(\vec{w}) = 0$
3) Dual feasibility → α_i for all i, lagrange multipliers for inequality are non negative.
4) Complementary slackness → if constraints active $\alpha_i \geq 0$
   if constraint inactive $\lambda = 0$

KKT conditions together determine a solution to a specific problem

---

Constraind Least Squares (CLS) and Tikhonov Regularization

- Minimize $\|Aw-b\|^2$

Control size or penalize large weights, you add constraints or regularization.

Both CLS and Tikhonov, you control $\|\vec{w}\| = w_1^2 + w_2^2 + \dots w_n^2$, helps avoid overfitting

- Fits training data, sacrafices some accuracy for better smooth curve
- Tikhonovs Regularization: $\min_w \|A\vec{w} - \vec{b}\|^2 + \lambda\|w\|^2$ (penalty on large weights)

without regularization: $w = \begin{bmatrix} 1 \\ 1000 \end{bmatrix}$, $w = \begin{bmatrix} 0.99 \\ 10 \end{bmatrix}$

- CLS problem is like $\min \|Aw-b\|^2$ subject to $\|w\|^2 \leq \theta$
- θ max allowed size of w
- $\min \|Aw-b\|^2 + \lambda \|R\vec{w}\|^2$
  ↳ Tunable constant
  Penalizing size of w

Tikhonov Type Questions

1) Given: $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$, $b = \begin{bmatrix} 5 \\ 6 \end{bmatrix}$, $\pi = 1$

Find minimizer w^* for $\|Aw - b\|^2 + \pi \|w\|^2$

$w^* = (A^TA + \pi I)^{-1} A^Tb$ Tikhonov: $\|Aw - b\|^2 + \pi \|w\|^2$

$(A^TA + \pi I) = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 10 & 14 \\ 14 & 20 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

$= \begin{bmatrix} 11 & 14 \\ 14 & 21 \end{bmatrix}^{-1} \rightarrow \frac{1}{ad-bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{35} \begin{bmatrix} 21 & -14 \\ -14 & 11 \end{bmatrix}$

Larger π Shrinks w more

$A^Tb = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 23 \\ 34 \end{bmatrix} = \begin{bmatrix} 1/5 \\ 52/35 \end{bmatrix}$

2.) CLS Behaviour

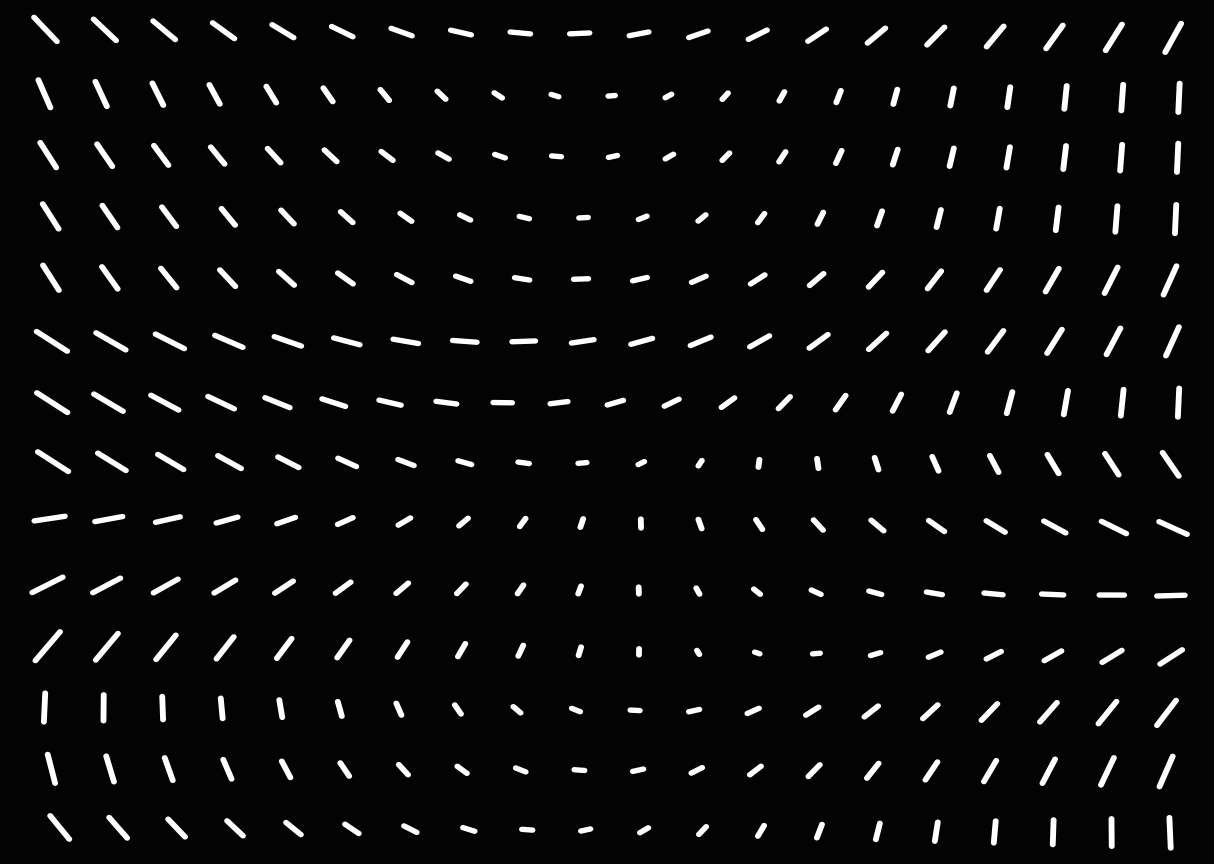
$\min \|Aw - b\|^2$ subject to $\|w^2\| \leq \theta$

$\|w\|^2 = 25$

$\theta = 10$

$\|w^*\|^2 = 25 > \theta = 10$ violates constraint

Test 4 Prep



2024 - Test 4

2) Objective function and non linear equality constraint:

$$f(\vec{w}) = (w_1 - 1)^2 + (w_2 - 1)^2$$
$$p(\vec{w}) = w_1^2 - w_2^2 = 0$$

Find langrange function

$$L(x, y, \nu) = f(x, y) + \nu \cdot g(x, y)$$
$$= (w_1 - 1)^2 - (w_2 - 1)^2 + \nu (w_1^2 - w_2^2)$$

3) Objective function: $f(w) = \frac{1}{2} \vec{w}^T K \vec{w}$ and linear equality constraint $m\vec{w} - c = 0$, where

$K = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ $m = \begin{bmatrix} 1 & 1 \end{bmatrix}$ $c = 2$, find minimizer that satisfies constraint

Note: no linear term in $\frac{1}{2} w^T K \vec{w}$
so, means $c = 0$ in general form

Note: RHS of KKT
Linear → $\frac{1}{2} x^T Q x \longrightarrow \begin{bmatrix} 0 \\ b \end{bmatrix}$
Quadratic → $\frac{1}{2} x^T Q x + c^T x \longrightarrow \begin{bmatrix} -c \\ b \end{bmatrix}$

1) Setup KKT matrix:

$$\begin{bmatrix} K & m^T \\ m & 0 \end{bmatrix} \begin{bmatrix} w \\ \nu \end{bmatrix} = \begin{bmatrix} 0 \\ c \end{bmatrix}$$

$$\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

2) Setup linear systems:

① $2w_1 + w_2 + \nu = 0$
② $w_1 + 2w_2 + \nu = 0$
③ $w_1 + w_2 = 2$

3) Solve:

③ $w_2 = 2 - w_1$

Sub in ①: $2w_1 + (2 - w_1) + \nu = 0$

$w_1 + 2 + \nu = 0$

$\nu = -w_1 - 2$

$w_1 + 2(2 - w_1) + (-w_1 - 2) = 0$

$w_1 + 4 - 2w_1 - w_1 - 2 = 0$

$-2w_1 + 2 = 0$

$w_1 = 1$

$w_2 = 2 - 1$

$w_2 = 1$

$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$\nu = -1 - 2$

$\nu = -3$

$\therefore w^* = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

Q4) Objective $f(\vec{w}) = \frac{1}{2} w^T K \vec{w} + q^T w$ and linear equality constraint $m\vec{w} - c = 0$

$K = \begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix}$ $m = [1 \ 1]$ $q = \begin{bmatrix} -4 \\ -2 \end{bmatrix}$ $c = 2$

Find minimizer w^* and dual matrix B

1) Setup KKT matrix: $\begin{bmatrix} K & m^T \\ m & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} -q \\ c \end{bmatrix}$

$\begin{bmatrix} 8 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \\ 2 \end{bmatrix}$

WATCH OUT FOR $-q$

$A^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

$\frac{1}{16-0}$

2) Setup and solve for ν

$8w_1 + \nu = 4$
$2w_2 + \nu = 2$
$w_1 + w_2 = 2$

$w_1 = 2 - w_2$ $\quad \nu = 2 - 2w_2$

$8(2-w_2) + \nu = 4$
$8(2-w_2) + (2-2w_2) = 4$
$16 - 8w_2 + 2 - 2w_2 = 4$
$-10w_2 = 4 - 18$
$-10w_2 = -14$
$w_2 = \frac{14}{10}$

$w_1 = 2 - \frac{14}{10}$
$w_1 = \frac{3}{5}$

$\nu = 2 - 2w_2$
$\nu = 2 - 2(\frac{14}{10})$
$\nu = -0.8$

$\nu = 2 - 2(\frac{7}{5})$
$\nu =$

Finding Dual Matrix Use $B = mK^{-1}m^T$

$B = (mK^{-1}m^T)^{-1}$

$B = \left([1 \ 1] \begin{bmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right)^{-1}$

$B = \begin{bmatrix} \frac{1}{8} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$= \left[\frac{5}{8}\right]^{-1} \rightarrow \frac{8}{5}$?

$B = 0.625$

5.) Convex objective: $f(t) = t^2 + 4t$, nonlinear inequality constraint $P_1(t) = t^2 - 6t + 8$ with $P_1(t) \leq 0$ find minimizer t^*

1) Lagrangian: $f(t) + \nu \cdot P_1(t)$

$(t^2 + 4t) + \nu(t^2 - 6t + 8)$

1) minimize $f(t)$:

$f(t) = t^2 + 4t$
$f'(t) = 2t + 4$
$= 2(t+2)$
$t = -2$

$(-2)^2 - 6(-2) + 8$
$4 + 12 + 8$ ✗

constraint violation

now what?

For a convex function on closed interval the minimum should occur at

- end points
- or stationary point (all tried here)

2) Optimize over feasible region,

$P_1(t) = t^2 - 6t + 8$

Factor: $(t-4)(t-2)$

$t = 4$ $\quad t = 2$

3) Minimize on this interval $t \in [2, 4]$

$f(2) = 12$
$f(4) = 32$

$\therefore$ min @ $t = 2$

This is inspection method: only use lagrange if you know constraints active

Q6) Convex objective function $f(\vec{w}) = (w_1-3)^2 + (w_2-4)^2$, nonlinear inequality constraint $p_1(\vec{w}) = w_1^2 + w_2^2 - 4$ with $p_1(\vec{w}) \leq 0$, find minimizer w^*.

1) Intuition: The constraint is circle of radius 2, centered at the origin.
Objective: convex quadratic point (3,4)
Trying to minimize distance from (w_1, w_2) to (3,4)

2) Try unconstrained: $f(w_1, w_2) = (w_1-3)^2 + (w_2-4)^2$
$w_1 = 3 \quad w_2 = 4$
Plug into p_1: $p_1(3,4) = 21 \leq 0$ ×

3) Does not work, must be lagrange:
$f(w_1, w_2) + \mu(p(w_1, w_2))$
$(w_1-3)^2 + (w_2-4)^2 + \mu(w_1^2 + w_2^2 - 4)$

4) KKT
w_1: $2(w_1-3) + \mu(2w_1)$
w_2: $2(w_2-4) + \mu(2w_2)$
Stationary: π: $w_1^2 + w_2^2 - 4$
w.r.t w_1, w_2, π

$2w_1 - 6 + 2\pi w_1 = 0$
$2w_1 + 2\pi w_1 = 6$

$(w_1-3) + \pi w_1 = 0$
$\frac{w_1-3}{w_1} = (-\pi)$
$\frac{3-w_1}{w_1} = \pi$

5) Set all equal to 0:
$2(w_1-3) + 2\pi w_1 = 0 \rightarrow \pi = \frac{3-w_1}{w_1}$
$2(w_2-4) + 2\pi w_2 = 0 \rightarrow \pi = \frac{4-w_2}{w_2}$
$w_1^2 + w_2^2 = 4$

6) Solve:
$\frac{3-w_1}{w_1} = \frac{4-w_2}{w_2}$
$w_2(3-w_1) = w_1(4-w_2)$
$3w_2 - w_1w_2 = 4w_1 - w_1w_2$
$3w_2 = 4w_1$
$w_2 = \frac{4}{3}w_1$

$\nabla f = -\mu$ in vector form

$w_1^2 + (\frac{4}{3}w_1)^2 = 4$
$w_1^2 + \frac{16}{9}w_1^2 = 4$
$w_1^2(1 + \frac{16}{9}) = 4$
$w_1^2(\frac{25}{9}) = 4$
$w_1^2 = \frac{4}{\frac{25}{9}}$
$w_1^2 = \frac{36}{25}$
$w_1 = \pm\sqrt{\frac{36}{25}}$
$w_1 = \pm 1.2$

$w_2 = \frac{4}{3}w_1, \quad \pm\frac{4}{3}\cdot 1.2$
$w_2 = \pm 1.6$

7) Possible points:
(1.2, 1.6)
(-1.2, -1.6)

8) Evaluate
$f(1.2, 1.6) = 9$
$f(-1.2, -1.6) = 49$

$\therefore w^* = \begin{bmatrix} 1.2 \\ 1.6 \end{bmatrix}$

Test 5- Breakdown

1) Convex objective function $f(\vec{w}) = [\vec{w}-\vec{g}]^T [\vec{w}-\vec{g}]$, where $\vec{g} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$

linear inequality constraints

$A = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{bmatrix}$, $b = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix}$ Find number of inca constraints

$A\vec{w} \leq \vec{b}$

1) $f(w) = \|w-g\|^2$

minimized at $w = g = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$, if no constraints Sub g as $\vec{w}$

2) Check if $\vec{g} = [1 \;\; 3]^T$ satisfies constraints

$A\vec{w} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \\ 4 \end{bmatrix}$ Compare to B $\begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix}$

$A\vec{g} = b$?

$-1 \leq 0$
$-3 \leq 0$
$4 \leq 4$ ∴ One constraint is active

---

1) $f(w) = (w_1-1)^2 + (w_2-1)^2$

$p(w) = w_1^2 - w_2^2$

$L(w, \mu) = (w_1-1)^2 + (w_2-1)^2 + \mu(w_1^2 - w_2^2)$

2) $f(w) = \frac{1}{2} w^T K w$, lin equality constraint $m\vec{w} - c = 0$

$\begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$

$w_1 = 1$
$w_2 = 1$
$\mu = -3$

∴ Minimizer $w^* = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

3) $\begin{bmatrix} 8 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \\ 2 \end{bmatrix}$ $\frac{1}{16}\begin{bmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}$

$w_1 = \frac{3}{5}$
$w_2 = \frac{7}{5}$
$\mu = \frac{-4}{5}$

$\mu = -0.8$

$B = MK^{-1}M^T$

$B = [1 \;\; 1] \begin{bmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$B = \begin{bmatrix} \frac{1}{8} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$B = \frac{5}{8}$

4) $f(t) = t^2 + 4t$, $P(t) = t^2 - 6t + 8$ with $P(t) \leq 0$

1) check if uncons works

$f'(t) = 2t + 4$
$= 2(t+2)$
$t = -2$

2) check condition: $P(-2) = -2^2 - 6(-2) + 8$
$= 4 + 12 + 8 \leq 0$ X

3) Must lie on boundary, find crits on P

4) $P(t) = t^2 - 6t + 8$
$0 = (t-4)(t-2)$
$t = 4 \quad t = 2$

Check both endpoints
$f(4) =$
$P(2) =$

Check if this works

∴ min @ 2

5) $f(w) = (w_1-3)^2 + (w_2-4)^2$

$P(w) = w_1^2 + w_2^2 - 4$, $P_1(\vec{w}) \leq 0$, Find min

1) Try $w_1 = 3$, $w_2 = 4$

$P(3,4) = 21 \leq 0$ X

2) setup lagrange:

Homework 9 - Practise

Quadratic Objective with Linear Constraints

1. Find primal langrange equation
2. Solve KKT matrix to find w^* and ν^*
3. Formulate Dual
4. dual formulation to find ν^* then w^*

Linear equality constraints:

$M\vec{w} = c$ where

$M = [1\ \ 1]$ and $c = 4$

Task #1

$f(w) = \frac{1}{2} w^T K_1 \vec{w} + q_1^T \vec{w}$, where $K_1 = \begin{bmatrix} 18 & 8 \\ 8 & 10 \end{bmatrix}$, $g_1 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$, $\vec{q_1} = \begin{bmatrix} -106 \\ -60 \end{bmatrix}$

i) $L(\vec{w}, \nu) = \frac{1}{2} w^T K_1 \vec{w} + q_1^T \vec{w} + \nu(M\vec{w} - c)$

$= \frac{1}{2} [w_1, w_2] \begin{bmatrix} 18 & 8 \\ 8 & 10 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + \begin{bmatrix} -106 \\ -60 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + \nu \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} [w_1, w_2] - 4 \right)$

$= \frac{1}{2} \begin{bmatrix} 18w_1 + 8w_2 \\ 8w_1 + 10w_2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$

$= \frac{1}{2} \begin{bmatrix} 18w_1^2 + 8w_2w_1 \\ 8w_1w_2 + 10w_2^2 \end{bmatrix}$

$= \frac{1}{2}(18w_1^2 + 16w_1w_2 + 10w_2^2) - 106w_1 - 60w_2 + \nu(w_1 + w_2 - 4)$

→ Do this way, matrix form, omits constants

ii) $\begin{bmatrix} K & M^T \\ M & 0 \end{bmatrix} \begin{bmatrix} w \\ \nu \end{bmatrix} = \begin{bmatrix} -q \\ c \end{bmatrix}$

$\begin{bmatrix} 18 & 8 & 1 \\ 8 & 10 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 106 \\ 60 \\ 4 \end{bmatrix}$

$18w_1 + 8w_2 + \nu = 106$

$8w_1 + 10w_2 + \nu = 60$

$w_1 + w_2 = 4$

$w_1 = 4 - w_2$

$\nu = 60 - 8w_1 - 10w_2$

$\nu = 60 - 8(4 - w_2) - 10w_2$

$18(4 - w_2) + 8w_2 + 60 - 8(4 - w_2) - 10w_2 = 106$

$72 - 18w_2 + 8w_2 + 60 - 32 + 8w_2 - 10w_2 = 106$

$-12w_2 = 6$

$w_2 = -\frac{1}{2}$

$w_1 + (-\frac{1}{2}) = 4$

$w_1 = 4 + \frac{1}{2}$

$w_1 = \frac{9}{2}$

$8(\frac{9}{2}) + 10(-\frac{1}{2}) + \nu = 60$

$31 + \nu = 60$

$\nu = 29$

$w^* = \begin{bmatrix} 4.5 \\ -0.5 \end{bmatrix}$ $\nu^* = 29$

iii) dual: $g(\nu) = \min_w L(w, \nu)$

$g(\nu) = f(w(\nu)) + \nu(Mw(\nu) - 4)$

$L(w, \nu) = \frac{1}{2}(w - g)^T K (w - g) + \nu(Mw - 4)$

$h(w - g) = -\nu M^T$

$w(\nu) = g - \nu K^{-1} M^T$

$w(\nu) = \begin{bmatrix} 5 \\ 2 \end{bmatrix} - \nu \begin{bmatrix} \frac{1}{58} \\ \frac{5}{58} \end{bmatrix}$

$w_1 + w_2 = 7 - \frac{6\nu}{58} - 4$

$= 3 - \frac{3\nu}{29}$

Dual: $\frac{3\nu(58 - \nu)}{58}$

↳ Derive, set to 0, find ν then w_1, w_2.

$\nabla f(w)$ is the grad of partial derivatives,

$\nabla f(w^*) = -\nu^* M^T$

or $= K(w^* - g_1)$

HW-9 - Task 2

$m = [1\ 1]$ and $c = 4$

2) i) $f(w) = \frac{1}{2} w^T K w + q^T w$

$= \frac{1}{2} \begin{bmatrix} w_1 & w_2 \end{bmatrix} \begin{bmatrix} 8 & 2 \\ 2 & 12 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + \begin{bmatrix} -34 & 26 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$

$L = \frac{1}{2}(8w_1^2 + 14w_1w_2 + 2w_2^2 - 34w_1 + 26w_2) + \nu(w_1 + w_2 - 4)$

ii) $\begin{bmatrix} K & m^T \\ m & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 34 \\ -26 \\ 4 \end{bmatrix}$

$\begin{bmatrix} 8 & 2 & 1 \\ 2 & 12 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \nu \end{bmatrix} = \begin{bmatrix} 34 \\ -26 \\ 4 \end{bmatrix}$

$8w_1 + 2w_2 + \nu = 34$

$2w_1 + 12w_2 + \nu = -26$

$w_1 + w_2 = 4$

$w_1 = 4 - w_2$

$\nu = 34 - 8(4 - w_2) - 2w_2$

$\nu = 34 - 32 + 8w_2 - 2w_2$

$\nu = 2 + 6w_2$

$2(4 - w_2) + 12w_2 + 2 + 6w_2 = -26$

$8 - 2w_2 + 12w_2 + 2 + 6w_2 = -26$

$10 + 16w_2 = -26$

$\frac{16w_2}{16} = \frac{-36}{16}$

$w_2 = \frac{-9}{4}$

$w_1 = 4 + \frac{9}{4}$

$w_1 = \frac{25}{4}$

$2\left(\frac{25}{4}\right) + 12\left(\frac{-9}{4}\right) + \nu = -26$

$\nu = -11.5$

Could use Calc : Eq → 3 variable

3) $B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} 8 & 2 \\ 2 & 12 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 1 \end{bmatrix}$

$\frac{1}{ad-bc} \begin{bmatrix} d & -c \\ -b & a \end{bmatrix}$

$B = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} \frac{3}{23} & -\frac{1}{46} \\ -\frac{1}{46} & \frac{2}{23} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$\frac{1}{92} \begin{bmatrix} 12 & -2 \\ -2 & 8 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$= \begin{bmatrix} \frac{5}{46} & \frac{3}{46} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$= \frac{4}{23} \rightarrow \mu^* = \frac{23}{4}(-4)$? OFF

$= -23$

Homework 10:

Q1: Vector Argument, Linear Inequality constraints

i) $f(w) = 4w_1^2 + w_2^2$, with $Aw \leq \vec{b}$

where $A = [-1\ 0]$, $\vec{b} = [-2]$ A

$Aw - b = [-1\ 0]\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} - (-2) = -w_1 + 2$

a) yes it is convex

b) $L(w, \mu) = 4w_1^2 + w_2^2 + \mu(-w_1 + 2)$

c) Because: $4w_1^2 = \frac{1}{2}(8w_1^2)$

$w_2^2 = \frac{1}{2}(2w_2^2)$

$K = \begin{bmatrix} 8 & 0 \\ 0 & 2 \end{bmatrix}$, $q = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ (Because no constant term)

d) Unconstrained minimizer:

$\Delta f = 8w_1 = 0$

$2w_2 = 0$

$w_1 \geq 2$

$0 \geq 2$?

Does $(0, 0)$ work? ✗ wrong

The unconstrained minimizer is not feasible.

e) only one active constraint: $w_1 \geq 2$

$A = [-1\ 0]$

F) KKT Matrix for active constraint:

$$\begin{bmatrix} K & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -2 \end{bmatrix}$$

$$\begin{bmatrix} 8 & 0 & -1 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix}$$

$w_1 = 2$

$w_2 = 0$

$\mu = 16$

2) $f(w) = 5w_1^2 - 8w_2 - 10w_1 + 2w_2^2 + 13$

with $Aw \le b$, where $A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ $b = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$

a) Yes convex $\quad \begin{bmatrix} 10 & 0 \\ 0 & 4 \end{bmatrix}$

b) $Aw - b \le 0$

$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$

$-w_1 \le -2$

$w_2 \le -2$

c) K is the hessian

$K = \begin{bmatrix} 10 & 0 \\ 0 & 4 \end{bmatrix}$ $q = \begin{bmatrix} -10 \\ -8 \end{bmatrix}$

d) w.r.t $w_1 = 10w_1 - 10 \rightarrow w_1 = 1$

w.r.t $w_2 = -8 + 4w_2 \rightarrow w_2 = 2$

$=$

$\therefore$ not feasible

Q2: $f(t) = t^4 + t^2$, $P_4(t) \le 0$, $P_4(t) = t^2 - 3t + 2$

a) $f(t) = 4t^3 + 2t$

$f''(t) = 12t^2 + 2 > 0$

for all t $\therefore$ convex

b) $L(w, \lambda) = t^4 + t^2 + \lambda(t^2 - 3t + 2)$

c) $0 = 4t^3 + 2t$ violation

$0 = 2t(2t^2 + 1)$ $P(0) \le 0$ ✗

$t = 0$ $t = -\frac{1}{2}$

d) $P_4(t) = t^2 - 3t + 2 = 0$

$(t-1)(t-2) = 0$

$t = 1$ or $t = 2$

e) t feasible region is between 2 and 3

f) $t = 1$ is the minimizer

P5) $f(t) = (t+1)^2$, $P(t) \le 0$

$P(t) = t^2 + 3t + 2$

a) $f'(t) = 2(t+1)$

$= 2t + 2$

$= 2$ $t > 0$ for all t

$\therefore$ convex

b) $= (t+1)^2 + \lambda(t^2 + 3t + 2)$

$t = -1$

c) $P(-1) = -1^2 + 3(-1) + 2$

$= 1 - 3 + 2$

$= 0$ yes, feasible

7) $f(w; g) = \frac{1}{2}(w - g)^T(w - g)$

squared Euclidean distance from w and g.

unconstrained minimizer is always: $w = \vec{g}$

1) $f(w) = \frac{1}{2}\left(w - \begin{bmatrix} 4 \\ 4 \end{bmatrix}\right)^T \left(w - \begin{bmatrix} 4 \\ 4 \end{bmatrix}\right)$

$P(w) = w_1^2 \le 4$

$|w_1| \le 2$

2) unconstrained optimizer was $w_1 = 4$

which violates constraint

3) That means constraint is active

$w_1^2 = 4 \Rightarrow w_1 = \pm 2$

$L = \frac{1}{2}(w_1 - 4)^2 + (w_2 - 4)^2 + \lambda(w_1^2 - 4)$

7) $f(w; g) = \frac{1}{2}[\vec{w} - g]^T[\vec{w} - \vec{g}]$

hyperparameter: $g = \begin{bmatrix} 4 \\ 4 \end{bmatrix}$

$P(w) = w_1^2$

Find global minimizer of f subject to $P \le 4$.

$w_1^2 \le 4$ or $|w_1| \le 2$

$f(w) = \frac{1}{2}\|w - g\|^2$, squared distance

1) gradient: $\frac{1}{2}((w_1 - 4)^2 + (w_2 - 4)^2)$

w.r.t w_1: $(w_1 - 4) + 2\lambda w_1 = 0$

w.r.t w_2: $(w_2 - 4) = 0 \rightarrow w_2 = 4$

if $w_1^2 = 4 \Rightarrow w_1 = \pm 2$

2 Candidates: $w = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$, $w = \begin{bmatrix} -2 \\ 4 \end{bmatrix}$

$\lambda = 0.5$

w.r.t w_1: $(w_1 - 4) + 2\lambda w_1 = 0$

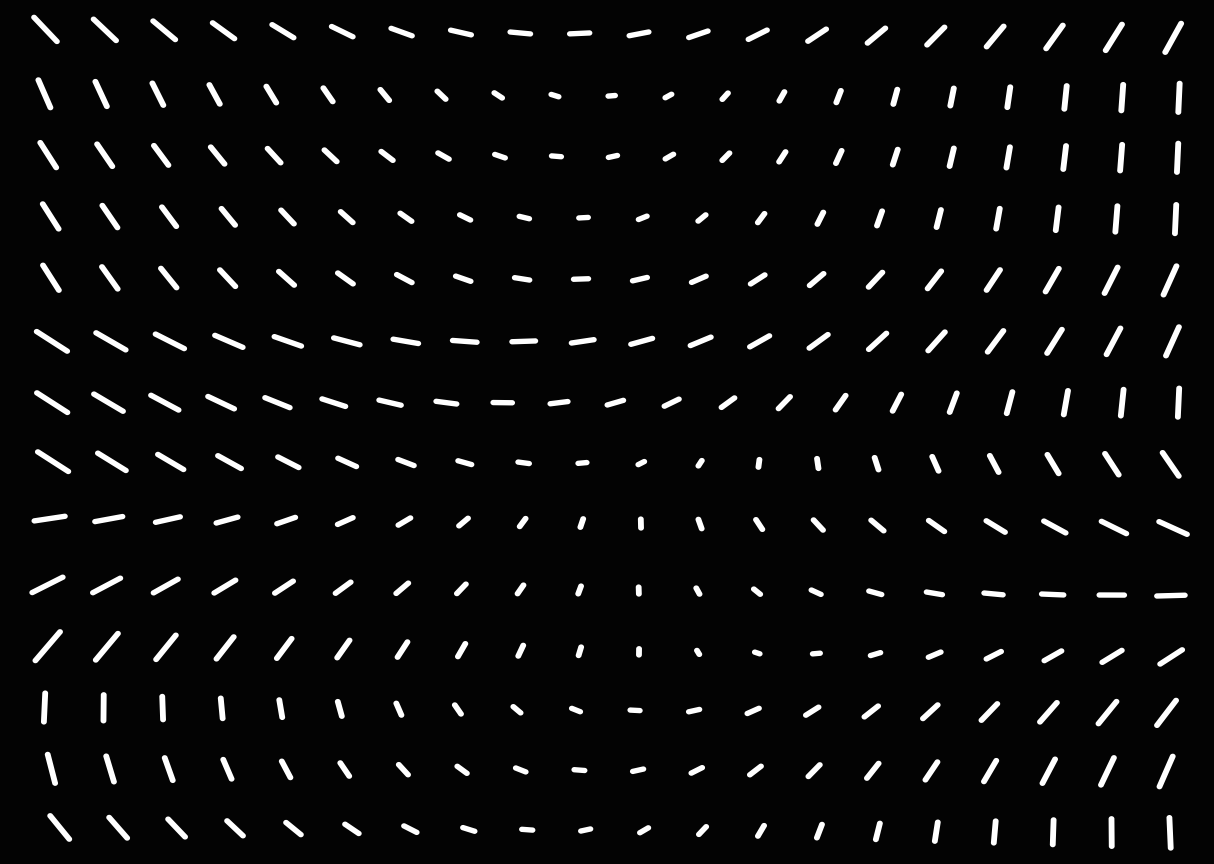
w_2: $w_2^* = 4$

we know $w_1 = \pm 2$

Try $(-2, 4)$

$(2, 4)$

Test 5 Prep



Fall 2024 Test 5:

2) For data vectors and label values

$x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $x_2 = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$, $x_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$, $x_4 = \begin{bmatrix} -3 \\ 0 \end{bmatrix}$, $y^T = [1 \ 1 \ -1 \ -1]$

SVM computation seperates data with weight vector $\vec{w} = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$ and $b = 0$

Find N_S of indexes of support vectors.

1) Find raw score for each point: $z = w^T x_i + b$

$z_1 = 0.5(1) + 0.5(1) = 1$

$z_2 = 0.5(3) + 0.5(0) = 1.5$

$z_3 = 0.5(-1) + 0.5(-1) = -1$

$z_4 = 0.5(-3) + 0.5(0) = -1.5$

2) Check if $y_i \cdot z_i = 1$

$y_1 = +1$, $y_1 z_1 = (1)(1) = 1$ ∴ support vector

$y_2 = +1$, $y_2 z_2 = (1)(1.5) = 1.5$ ∴ not S.V

$y_3 = -1$, $y_3 z_3 = (-1)(-1) = 1$ ∴ yes S.V

$y_4 = -1$, $y_4 z_4 = (-1)(-1.5) = 1.5$ ∴ not S.V

→ Support vectors are $N_S = \{1, 3\}$

3) Consider the information $x_1 = \begin{bmatrix} -1 \\ 3 \end{bmatrix}$, $x_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, $\alpha = \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix}$, $b = -1$

Find score $z(g)$ for $g = \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix}$?

1) Compute dot products $x_i^T g$

$x_1 = \begin{bmatrix} -1 \\ 3 \end{bmatrix}$, $g = \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix}$

$= -2.75$

$x_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $g = \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix}$

$= -0.25$

2) Multiply each term by α_i and y_i

$= -2.75(1)(0.25)$

$= -0.6875$

$= 0.25(-1)(-0.25)$

$= 0.0625$

3) $z(g) = (-0.6875) + (0.0625) + b$

$z(g) = -1.625$

4) For the embedding $\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \hookrightarrow \begin{bmatrix} a_1^2 \\ a_2^2 \\ -\sqrt{2}\, a_1 a_2 \end{bmatrix}$, find the kernel function $K(\vec{u}, \vec{v})$.

$v = (v_1, v_2)$

$v = (v_1, v_2)$

$\phi(v) = \begin{bmatrix} v_1^2 \\ v_2^2 \\ -\sqrt{2}\, v_1 v_2 \end{bmatrix}$

$\phi(v) = \begin{bmatrix} v_1^2 \\ v_2^2 \\ -\sqrt{2}\, v_1 v_2 \end{bmatrix}$

Write it out

$= (u_1^2)(v_1)^2 + (u_2^2)(v_2^2) + (-\sqrt{2}\, u_1 u_2)(-\sqrt{2}\, v_1 v_2)$

$= (u \cdot v)^2$

Shortcut: look at kernel is degree 2 → quadratic, deg = 3 → cubic

: mixed term $a_1 a_2$, $a_1^2 a_2$ → indicates polynomial

: $\|x\|^2$ → RBF

5) For Kernel $K(U,V) = (U \cdot V + 1)^2$, S.V, label values y_i, lagrange multipliers, b scalar

Find Score for $z(g) = \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix}$

1) write scoring equation $z = \alpha_1 y_1 K + \alpha_2 y_2 K_2 + b$

2) compute Kernel function

$(x_1, g) = [-1, 3] \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix} = (-2.75 + 1)^2 = 3.0625$

$(x_2, g) = [0, -1] \begin{bmatrix} 0.5 \\ -0.75 \end{bmatrix} = (0.75 + 1)^2 = 3.0625$

3) Sub in all values, compute: $z = (0.171)(1)(3.0625) + (0.171)(-1)(3.0625) - 1$

$= -1$

To find S.V

$y_i(x^T w) + b$

x_1: $1(1) + 0 = 1$ ∴

x_2: $1(2) + 0 = 2$ ∴ N.S $\{1, 3\}$

x_3: $-1(-1) + 0 = 1$

x_4: $-1(-2) + 0 = 2$

$z(g) = \left(\sum \alpha_i y_i (x_i \cdot \vec{g}) + b\right)$

$= (0.25)(1)(-2.75) + (0.25)(-1)(-0.25) - 1$

$= -1.625$

$z(g) = (0.171)(1)(3.0625) + (0.171)(-1)(3.0625) - 1$

$= -1$

Preparation - Test 5 Notes

CLS (Constrained Least Squares)

- minimizing squared error with constraints
- Normal Least Squares: $\min \|Xw - y\|^2$
  - ↳ with constraint $\|w\|^2 \leq \theta$

θ is a hyperparameter that limits how large w can be

θ helps prevent overfitting

Tikhonov Regularization

- adds penalty to size of weight vector
- minimizes both error and model complexity

$$\min \|Xw - y\|^2 + \lambda \|w\|^2$$

λ is regularization parameter that controls trade off

- Might see regularization matrix R, instead of $\|w\|^2$
- Using $\lambda = 0$, removes the regularization term making it original squared error.

- Both CLS and Tikhonov use controlling w,

$$\|w\|^2 = \sum_i w_i^2$$

  - In CLS: its a constraint
  - In Tikhonov: its a penalty

AS λ increases, weights decrease

Tikhonov Example

$X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $y = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$, $\lambda = 1$

Find w?

$$w = (X^T X + \lambda I)^{-1} X^T y$$

$$w = (I + 1)^{-1} \cdot I \cdot y = \begin{bmatrix} 1.5 \\ 2 \end{bmatrix}$$

- Primal formulation is the original problem that tries to maximize the margin between classes. Assuming the data is linearly seperable.

(Hard Margin SVM)

Core Idea: Finding best seperating hyperplane with the widest margin possible

Hard margin

- given an objective $\frac{1}{2}\|w\|^2$ try to minimize weight vector. which maximizes margin

- Constraint ensures no missclassification:

$$y_i(w^T x_i + b) \geq 1 \quad \text{for all } i$$

- To solve primal SVM Problem, use lagrangian method

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i(w^T x_i + b) - 1 \right]$$

- convert it to solve with derivatives and KKT

- Stationary point, optimal w, b is found by taking partial derivatives of L and setting them to 0:

$$\frac{1}{2}\|w\|^2 - \alpha_i\left[(y_i(w^T x + b) - 1\right]$$

- Taking derivative w.r.t to w1 / w2 results in $w2 - \alpha \quad w2 = \alpha$ $w1 - \alpha$ or $w1 = \alpha$

- α_i is a lagrage multiplier
  if $\alpha_i = 0$, point is not on margin / not support vector
  if $\alpha_i > 0$, point is on margin, point is support vector

Dual Formulation with Hard SVM:

$$\max_{\alpha} \sum_{i=1} \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

Subject to $\alpha_i > 0$

$$\sum \alpha_i y_i = 0$$

Quadratic Problem

Soft Margins as Slack Variables

- In SVM's, margin is gap between classes
- Slack variables allow some points to be inside margin or even be misclassified.

Usual Constraints: $y_i(w^T x_i + b) \geq 1 - \xi_i$

$\xi_i \geq 0$

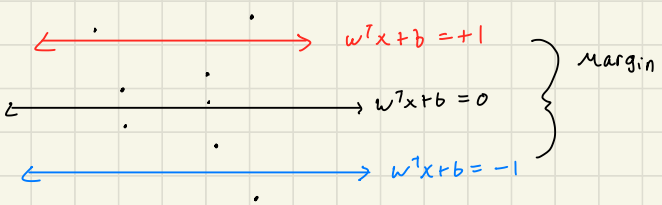
$\xi = 0$, correctly classified, outside margin → Ideal

$0 < \xi < 1$, inside margin but correctly classified → less than ideal

$\xi > 1$, misclassified point → worst

C controls how harshly margin violations are penalized

The decision boundary is $w^T x + b = 0$



Kernel Function to find support vectors and bias value.

- Dual SVM: $f(x) = \sum_i \alpha_i y_i K(x_i, x) + b$

Points with $\alpha_i > 0$ are support vectors

To find b: any support vector with $0 < \alpha_i < C$

$b = y_i - \sum_j \alpha_j y_j K(x_j, x_i)$

Kernel replaces dot products: $K(x, z) = x^T z$

Polynomial: $(x^T z + 1)^d$

Example

i	α_i	y_i	x_i	
1	0.4	+1	[1, 1]	→ S.V
2	0.0	+1	[2, 2]	
3	0.6	-1	[0, 1]	→ S.V

Find bias.

$$b = 1 - \left[\alpha_1 y_1 K(x_1, x_3) + \alpha_3 y_3 K(x_3, x_3)\right]$$

$K(x_1, x_1) = 2$

$K(x_1, x_3) = 1$

$= 1 - (0.4)(1)(2) + (0.6)(-1)(1))$

$b = 0.8$

Scoring a data vector (kernel function)

$$f(x) = \sum_i \alpha_i y_i K(x_i, x) + b$$

Then if $f(x) \geq 0$ class 1, $f(x) < 0$, class 2

Ex

given: $x_1 = [1, 1]$, $\alpha = 0.4$, $y = +1$

$x_3 = [0, 1]$, $\alpha = 0.6$, $y = -1$

$b = 0.8$

Score point $x = [2, 1]$

$K(x_1, x) = 3$

$K(x_3, x) = 1$

$f(x) = (0.4)(1)(3) + (0.6)(-1)(1) + 0.8$

$= 1.4$, positive class $+1$

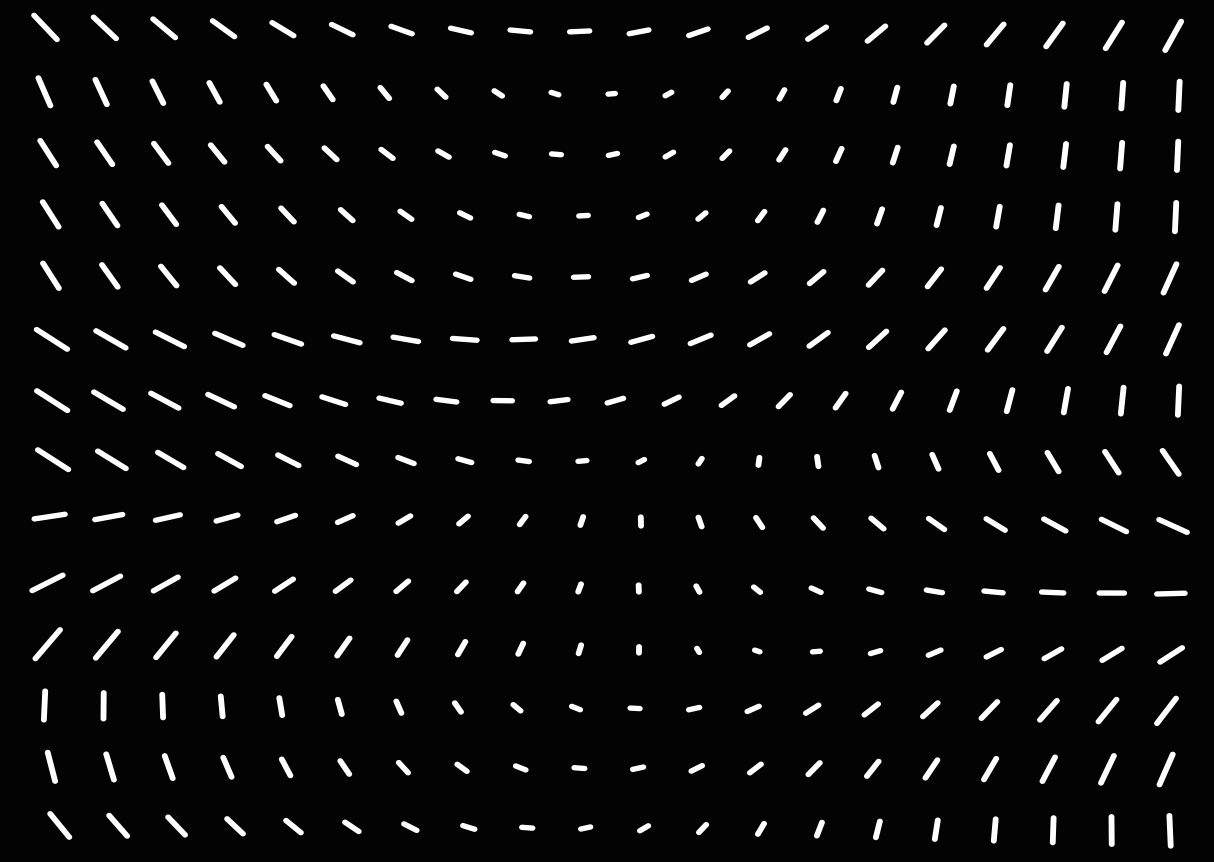
S.V are points closest to decision boundary

support boundary and they lie on or inside margin

Predicting 2 questions

- Tikhonov / CLS
- Slack variables
- Dual constraints for soft margins

Exam Review



Test 1 Takeup

1) For function and point, $f(t) = \ln(1+t)$ $t_0 = 0$
Find quadratic approx using truncated taylor series?

$f(t) = f(t_0) + f'(t_0)(t-t_0) + \frac{1}{2}f''(c)(t-t_0)^2$

$f(0) = 0$ $(t-t_0) = t$

$f'(t) = \frac{1}{t+1}$ $f'(0) = 1$

$f''(t) = -\frac{1}{(t+1)^2}$ $f''(0) = -1$

$= 1(t) + \frac{1}{2}(-1)(t)^2$

$= \frac{-t^2}{2} + t$

2) For a function, describe Stationary point at $t^* = 0$? $f(t) = t^3 - 3t^2$

$f(t) = t^3 - 3t^2$

$f'(t) = 3t^2 - 6t$

$f''(t) = 6t - 6$

$f''(0) = -6$

Use Second Derivative test
This is Concave, local maximizer

3) For Function $f(t) = t^4 + 4t^3$ has stationary points $t_1^* = -3$, $t_2^* = 0$. Describe the convexity.

$f(t) = t^4 + 4t^3$

$f'(t) = 4t^3 + 12t^2$

$f''(t) = 12t^2 + 24t$

Second Deriv Test:
$f(-3) > 0$ ∴ min, convex
$f(0) = 0$ ∴ indeterminate

4) Backtracking: given $f(t) = t^3 - t^2$, $\beta = \frac{1}{2}$, $t_1 = 1$, $s_0 = 1$

Find Stepsize $s = \beta^r s_0$ for $f(t)$ after single step of backtracking.

Armijo condition: $f(t_k + \beta^r s_0 d_k) \leq f(t_k) + \alpha_k \beta^r s_0 d_k$

$s = \beta^r s_0$

1) Plug in numbers

$f(1) = 0$

$f'(t) = 3t^2 - 2t$ $f'(1) = 1$, $d = -1$

$f''(t) = 6t - 2$

For $r = 0$, $s = 1$
$f(0) = 0 \leq -\frac{1}{2}$

For $r = 1$, $s = 0.5$
$f(1 - \frac{1}{2}) = -\frac{1}{8} \leq -\frac{1}{4}$

For $r = 2$
$f(1 - \frac{1}{4}) = -\frac{9}{64} \leq -\frac{1}{8}$ ∴ $s = \frac{1}{4}$

$s = \frac{1}{2}^2 \cdot 1 = 0.25$

5) For function f and point $\vec{w}_0$ and direction $\vec{v}$

$$f(w) = w_1^2 + (w_2+1)^3 \qquad w_0 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \qquad \vec{v} = \begin{bmatrix} -2 \\ -1 \end{bmatrix}$$

Find Directional deriv?

1) Partials, $f'(w) = 2w_1 + 3(w_2+1)^2$

2) Plug in w_0 $\quad [4, 12]$

$$\nabla f \cdot v = [4, 12] \begin{bmatrix} -2 \\ -1 \end{bmatrix} = -20$$

6) Function $f(w) = (w_1 - 1)^2 + w_2^3$ has stationary point at $w^* = [0]$

From eigenvalues of Hessian, describe convexity?

$f'(w) = 2(w_1 - 1) + 3w_2^2$

$f''(w) = 2, \; 6w_2$ $\qquad \begin{bmatrix} 2 & 0 \\ 0 & 6w_2 \end{bmatrix} \to \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$

$\therefore$ semi definite because one eigenvalue $= 0$

$\vec{w}_1 = w_0 - s_0 \nabla f$

$f(w) = 6w_1 + 4w_2 + 1$ $\qquad \|Rw\|^2$

$f'(w) = -[6, 4] + 1$

$= \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 6 \\ 4 \end{bmatrix}$

$= \begin{bmatrix} -5 \\ -3 \end{bmatrix}$

$w_0 - s\nabla f \to$ steepest descent

$w_0 - s\nabla f$

$w_0 - s\nabla f$

$w_0 - s\nabla f$

$d = -\nabla f(w_0)$

Test 2. Takeup

1) For a function $g_1(\vec{w}) = w_1^4 + w_2^4 + 2w_1^2w_2^2 - 1$, with 3 new functions $f_{1,1}(\vec{w}) = g_1(\vec{w}) + g_1(\vec{w})$, $f_{1,2}(\vec{w}) = (g_1(\vec{w}))^2$ and $f_{1,3}(\vec{w}) = \sqrt{g_1(\vec{w}) + 1}$, State how many are convex?

1) First Step, is recognize its factorable: $g_1(\vec{w}) = w_1^4 + w_2^4 + 2w_1^2w_2^2 - 1$

$g_1(w) = (w_1^2 + w_2^2)^2 - 1$

Convexity Closure Rules:

- Positive Scaling and Sums of Convex functions are convex
- Add/Subtracting constants does not Change Convexity
- norms and Squared norms are convex

1. $f_{1,1}(w) = g_1(w) + g_1(w) = 2g_1(w)$

   Positive scaling Convex functions → Convex

   ∴ 2 functions are convex

2. $f_{1,3}(w) = 2\sqrt{g_1(\vec{w}) + 1}$

   $2\sqrt{(w_1^2 + w_2^2)^2 - 1 + 1}$

   Squared norm is convex. Quadratic function is convex

3. $f_{1,2}(w) = (g_1(\vec{w}))^2 \rightarrow f_{12}(\vec{w}) = \|w\|^8 - 2\|w\|^4 + 1$

   - takes negative values
   - v^2 is decreasing on negative inputs

2.) For a function $f_2(w)$, Starting Point $\vec{w_0}$, Stepsize s_0

$f_2(w) = 3w_1^2 + 2w_2^2$, $w_0 = [1]$, $s_0 = 1$

find Point $\vec{w_1}$ Single Step in direction of Steepest descent?

$w_1 = -\nabla f + s$ $= -[6, 4]$

$= 6w_1 + 4w_2$ $= [6, -4] + [1, 1]$ $w_1 = [-5, -3]$

3.) For a function $f_3(w) = w_1^4 - w_2^2 + w_2^4$ and Starting Point $w_0 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$

Find descent vector $\vec{d}$

$\vec{d} = -|\nabla f(w_0)|^T$

$\nabla f = 4w_1^3 - 2w_2 + 4w_2^3$

$= 4(-1)^3 - 2(-1) + 4(-1)^3$

$= [-4, -2]$

Evaluate at w_0

$\nabla f'' = 12w_1^2 - 2 + 12w_2^2$

$= \begin{bmatrix} 12w_1^2 & 0 \\ 0 & 12w_2^2 - 2 \end{bmatrix} = \begin{bmatrix} 12 & 0 \\ 0 & 10 \end{bmatrix}$

$d = -\begin{bmatrix} 1/12 & 0 \\ 0 & 1/10 \end{bmatrix} \begin{bmatrix} -4 \\ -2 \end{bmatrix}$

$= \begin{bmatrix} 1/3 \\ 1/5 \end{bmatrix}$

$\vec{d} = \begin{bmatrix} 0.33 \\ 0.2 \end{bmatrix}$

4) For an NLS problem where each entry of a function $\vec{r}(w)$ is

$$r_i(\vec{w}) = \|x_i - w^T\|_2 = \left([x_i - w^T][x_i - w^T]^T\right)^{1/2}$$

Describe row J_i of Jacobian matrix $Jr(w)$?

$r_i(w) = \|x_i - w^T\|$ $\quad J_i = \frac{\partial r_i}{\partial w}$

$r_i'(w) = -\left([x_i - w^T][x_i^T - \vec{w}]\right)^{1/2}$

$\nabla r_i(w) = -\frac{x_i - w^T}{\|x_i - w^T\|}$

NLS: residuals are non linear functions of $\vec{w}$

$r_i(\vec{w}) = \|x_i^{\rightarrow} - \vec{w}\|$

need gradient of loss function

Affects distance to each point

5) 1D Fermat problem with X, damping coefficient and initial estimate t_0?

$$X = \begin{bmatrix} 1 \\ 3 \\ 9 \end{bmatrix}, \quad t_0 = 2, \quad \pi = .1$$

$t_1 = t_0 + (J^TJ + \pi I)^{-1} J^T r$

1) Residuals: $r = [x - t_0]$

$r = \begin{bmatrix} 1-2 \\ 3-2 \\ 9-2 \end{bmatrix}$ $\quad r = \begin{bmatrix} -1 \\ 1 \\ 7 \end{bmatrix}$

2) Jacobian: $\frac{x_i - t}{|x_i - t|}$ $\quad = J = \begin{bmatrix} \frac{[1-2]}{|1|} \\ + \\ \frac{9-2}{|1|} \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix}$

$t_1 = 2 + 3 + 1$

3) $J^T r = [-1 \; 1 \; 1] \begin{bmatrix} -1 \\ 1 \\ 7 \end{bmatrix}$

$= 7$

$t_1 = 2 - 4^{-1}(7) = 0.25$ or 3.75

$\pm$

6) Match the contour plot.

$f_1 = (w_1^2 + w_2^2)$

$f_2 = (w_1 + w_2)^2$

$f_3 = (w_1^2 - w_2^2)$

$f_4 = (w_1^3 - 3w_1w_2^2)$

Level curve: $\{w \in \mathbb{R}^2 \mid f(w) = c\}$

Circles: $x^2 + y^2$

ellipses: $ax^2 + by^2$

hyperbolas: $x^2 - y^2$

horizontal opening hyperbola

Can be paraboloid

no constant lines

one constant line

Test 3: Takeup

1.) For artificial neuron with logistic function, weight vector $\vec{w}$, and observations A, find linear response $\vec{v}$.

$$A = \begin{bmatrix} 2 & 3 \\ 0 & 2 \end{bmatrix}, \quad \vec{w} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

2 by 3 3 by 1

Design matrix is A with ones vector

$$X = \begin{bmatrix} 2 & 3 & 1 \\ 0 & 2 & 1 \end{bmatrix} \quad \vec{w} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

$$\vec{v} = X\vec{w} \rightarrow [0, -1]$$

2) 3 Layer ANN, logistic function everywhere, 2 neuron hidden layer with linear response $\vec{v} = [2, 5]$. Find derivative as Jacobian?

1) Logistic function $\sigma(v) = \frac{1}{1+e^{-v}}$

$J_i = \text{diag}(\sigma'(z_1), \sigma'(z_2))$ 2) Activation $\vec{\phi} = [0.8808 \quad 0.9933]$

The chance it's in class 1

3) $\psi = [0.8808(1-0.8808), \quad 0.9933(1-0.0.9933)]$

$\psi = [0.1050, \quad 0.0066]$

$$J\vec{\phi} = \begin{bmatrix} \psi_1 & 0 \\ 0 & \psi_2 \end{bmatrix} = \begin{bmatrix} 0.1050 & 0 \\ 0 & 0.0066 \end{bmatrix}$$

Jacobians always diagonal

3) Given weight vector $\vec{w}$, input vector $\vec{x}$, back-prop factor b Find Steepest descent vector $\vec{d}$?

$$\vec{w} = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} \quad x = [3 \ 2] \quad y = 1 \quad b = -0.5$$

1) Calculate linear response: $v = xw \rightarrow \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} [-1 \ 1 \ 1]$

$v = 0$

2) Activation: $\phi(0) = \frac{1}{1+e^{-0}} = 0.5, \quad \psi = 0.5(1-0.5) = 0.25$

3) Descent Vector: $\vec{d} = -[x \ 1] \psi b$

$\vec{d} = -[3 \ 2 \ 1](0.25)(-0.5)$

$\vec{d} = 0.125[3 \ 2 \ 1] \longrightarrow \vec{d} = [0.375, 0.250, 0.125]$

4) 3 layer ANN with ReLU activation function, with weight matrix ${}_2W$ for hidden layer 2, weight vector ${}_3\vec{w}$ and activation deriv ${}_3\psi$ and input x.

$${}_2W = \begin{bmatrix} 1 & 0 \\ 1 & 2 \\ 1 & 1 \end{bmatrix} \quad {}_3W = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad x = [3 \ -2]$$

1) Linear response: ${}_2U = [x \ \ 1]$

$U = xW$

$U = [3 \ -2 \ 1] \begin{bmatrix} 1 & 0 \\ 1 & 2 \\ 1 & 1 \end{bmatrix}$

${}_2W = [2 \ -3]$

2) Hidden Activation ${}_2\phi = [2 \ 0]$

3) Output Response: $[{}_2\phi \ \ 1] \ {}_3W = 3$

$[2 \ 0 \ 1] \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \ {}_3\vec{w} = 3$

4) Activation: ${}_3\phi = 3$

5) For a 3 layer ANN, with ReLU, weight vector ${}_3\vec{w}$ and back-prop value ${}_3b$ from objective computation, ${}_3\vec{w} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$, ${}_3b = 8$ (error signal from loss function)

Find back propagation values ${}_2\vec{b}$ provided to layer 2?

hidden backprop = (output_weight)(output slope)(deriv)

1) Find weights: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

2) Find $m = {}_3b \cdot {}_3\psi$

$m = 1 \cdot 8$

$m = 8$

3) ${}_2b = [\underset{w \cdot m}{1 \cdot 8}, \ \underset{w \cdot m}{1 \cdot 8}]$

$= [8, 8]$

${}_3\psi$ is usually always 1

If given logistic function

Find $\frac{1}{1+e^{-U}}$, (would be given U)

Then find ${}_3\psi$, then multiply by b.

Test 4: Take up

1) For objective function, nonlinear equality constraint. Find lagrange function:

$f(\vec{w}) = (w_1-1)^2 + (w_2-1)^2$, $p(\vec{w}) = w_1^4 - w_2^2$

$L(w, \mu) = (w_1-1)^2 + (w_2-1)^2 - \mu(w_2^2 - w_1^4)$

Just watch for sign switch. Lagrange is usually $f(w) + \mu(P(w))$

2.) For obj function $F(w) = \frac{1}{2} w^T K w$ and lin equality constraint $mw - c = 0$, where

$K = \begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}$, $m = [1\ 1]$, $c = 2$. Find minimizer

1.) $\begin{bmatrix} K & m^T \\ m & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ b \end{bmatrix}$

2.) $\begin{bmatrix} 2 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$

3) use Lin alg or calculator to solve for w_1, w_2.

4) $w^* = \begin{bmatrix} 1.5 \\ 0.5 \end{bmatrix}$

3.) For obj function $f(\vec{w}) = \frac{1}{2} w^T K \vec{w} + q^T \vec{w}$ and lin equality constraint $mw - c = 0$

where $K = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}$, $m = [1\ 1]$, $q = \begin{bmatrix} -4 \\ -8 \end{bmatrix}$ $c = 1$

Find minimizer; w^* and Dual matrix B.

1) $\begin{bmatrix} 2 & 0 & 1 \\ 0 & 8 & 1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \mu \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \\ 1 \end{bmatrix}$ → Plug into Calc and find minimizer

2) Dual matrix is $B = M^T K^{-1} M$

3) $K^{-1} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{8} \end{bmatrix}$

$B = [1\ 1] \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{8} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$B = [1/2\ \ \frac{1}{8}] \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$B = 0.625$

4) For convex function $f(t) = t^2 + 4t$ and nonlinear inequality constraint $p_1(t) = t^2 - 5t + 6$ with $p_1(t) \leq 0$ Find minimizer t^*

1) Check unconstrained

$f'(t) = 2t + 4 = 0$

$t = -2$

2) check feasible

$p(-2) = 20 \geq 0$ ∴ not possible

3) Find feasible region

$t^2 - 5t + 6 = 0$

$(t-3)(t-2)$

$t = 3$ $t = 2$

4) Check on boundaries

$f(2) = 12$

$f(3) = 21$

5) ∴ minimizer is $t^* = 2$

5) For objective function $f(w) = (w_1-3)^2 + (w_2-4)^2$ and nonlinear inequality constraint $p(w) = w_1^2 + w_2^2 - 9$, find minimizer w^*

$p(w \le 0)$

1) Try unconstrained, given circle eq with roots (3,4), Try $w = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$.

2) Check feasible $3^2 + 4^2 = 25 \ge 9$

Shortcut: $x_{proj} = R \frac{x}{\|x\|}$

R = radius (3)

$x = (3,4)$

$\|x\| = 5$

$= 3 \cdot \frac{1}{5} \begin{bmatrix} 3 \\ 4 \end{bmatrix}$

$= \begin{bmatrix} 1.8 \\ 2.4 \end{bmatrix}$

6) For convex objective $[w-g]^T [w-g]$, where $g = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$ and linear inequality constraints $Aw \le b$ with $A = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{bmatrix}$ $\vec{b} = \begin{bmatrix} 0 \\ 0 \\ 4 \end{bmatrix}$

find linear constraints, exactly satisfied

$Ag = b?$

$\begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 4 \end{bmatrix} = \begin{bmatrix} 0 \\ -4 \\ 4 \end{bmatrix}$ → 0, → 4

$\therefore$ 2 active constraints

Test 5 - Takeup

1) For convex objective: $F(t) = (t-1)^2$, with hyper parameter $\tau = 2$ and $R = 1$ find minimizer of Tikhonov $T(t, \tau)$.

Tikhonv objective: $T(t) = (t-1)^2 + \tau R t^2$

$T(t) = (t-1)^2 + 2t^2$

$= t^2 - 2t + 1 + 2t^2$

$= 3t^2 - 2t + 1$

$0 = 6t - 2$

$= 2(3t - 1)$

$t = \frac{1}{3}$

Just plug into tikhonov and expand, derivative, set to 0.

2) For data vectors and labels

$x_1 = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$, $x_2 = \begin{bmatrix} 3 \\ 0 \\ 2 \end{bmatrix}$, $x_3 = \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix}$, $x_4 = \begin{bmatrix} -3 \\ 0 \\ 0 \end{bmatrix}$, $y = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$

for $\vec{w} = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}$ and bias $b = 1/3$

$\|Rt\|^2$

$2t^2 = 2t \cdot 2t$

$= 4t^2$

x_1: $y_i(xw - b) = 1$?

x_1: $1\left(\frac{4}{3} - 1\right) = 1$

x_2: $1\left(\frac{5}{3} - 1\right) = \frac{4}{3}$

x_3: $-1\left(\frac{-3}{3} - \frac{1}{3}\right) = 1$

x_4: $-1\left(-1 - \frac{1}{3}\right) = \frac{4}{3}$

$\therefore$ Support Vectors $N: \{1, 3\}$

3) For the following, $x_1 = \begin{bmatrix} -1 \\ 3 \\ 2 \end{bmatrix}$, $x_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$, $y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, $\alpha = \begin{bmatrix} 0.1667 \\ 0.1667 \end{bmatrix}$, $b = -1$

for $z(g)$, $\vec{g} = \begin{bmatrix} 0.5 \\ -0.75 \\ 0.0 \end{bmatrix}$

$z(g) = (0.1667)(1)(-2.75) + (0.1667)(-1)(-0.25) - 1$

$= -1.4167$

$= (u \cdot v)^2$

4) For embedding $\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \hookrightarrow \begin{bmatrix} a_1^2 \\ a_2^2 \\ a_3^2 \\ \sqrt{2}\, a_1 a_2 \\ \sqrt{2}\, a_1 a_3 \\ \sqrt{2}\, a_2 a_3 \end{bmatrix}$

For $U = [u_1, u_2, u_3]$

$V = [v_1, v_2, v_3]$

$(-\sqrt{2} u_1 u_2)(-\sqrt{2} v_1 v_2) = 2(u_1 u_2 v_1 v_2)$

$(-\sqrt{2} u_1 u_3)(-\sqrt{2} v_1 v_3) = 2(u_1 u_3 v_1 v_3)$

$(\sqrt{2} u_2 u_3)(\sqrt{2} v_2 v_3) = 2(u_2 u_3 v_2 v_3)$

5) For Kernel $K(u, v) = (u \cdot v + 1)$, S.V, labels, lagrange

$x_1 = \begin{bmatrix} -1 \\ 3 \\ 1 \end{bmatrix}$, $x_2 = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$, $\vec{y} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, $\vec{\alpha} = \begin{bmatrix} 0.0143 \\ 0.0143 \end{bmatrix}$, $b = -1$

$a_{11} = u_1 u_1$

$a_{12} = u_2 u_2$

$z(g)$ for $g = \begin{bmatrix} 0.5 \\ -0.75 \\ 0 \end{bmatrix}$

$K_1 = 3.0625$

$K_2 = 3.0625$

$z(g) = \alpha_i y_i K + \alpha_2 y_2 K_2 + b$

$z(g) = (0.0143)(1)(3.0625) + (0.0143)(-1)(3.0625) - 1$

$= -1$