WHAT IS OPTIMIZATION?

A: Minimizing things

In college you learned:
set derivative to zero

sounds easy.

convex!
BUT THEN...

What if there’s no closed-form solution?
What if the problem has 1 BILLION dimensions?
What if the problem is non-convex?
What if the function has no derivative?
What if there are constraints?
What if the objective function has a BILLION terms?

Does this ever really happen?
MODEL FITTING PROBLEMS
BASIC OPTIMIZATION PROBLEMS: MODEL FITTING

- Training data / inputs
- Label data / outputs
- Model
- Parameters
- Loss function

Example: Linear model

$$f(d_i, w) = y_i$$

$$d_i^T w = b_i$$

Least-squares

$$\min \sum_i \ell(d_i, w, y_i)$$

$$\min \| Dw - b \|^2$$

$$\ell(d_i, w, b_i) = (d_i^T w - b_i)^2$$
BASIC OPTIMIZATION PROBLEMS: MODEL FITTING

\[
\min \sum_i \ell(d_i, w, y_i) \quad \text{loss function}
\]

\[
\min \sum_i l(d_i, w, y_i) \quad \text{"prior"}
\]

\[
\min J(w) + \sum_i l(d_i, w, y_i)
\]

\[
\min \|Dw - b\|^2
\]

\[
\ell(d_i, w, b_i) = (d_i^T w - b_i)^2
\]

\[
\min w_2^2 + \|Dw - b\|^2
\]

penalized regressions

ridge penalty
EIGENVALUE DECOMPOSITION

• Spectral theorem: symmetric matrices have a complete, orthogonal set of eigenvalues

\[ A = \]

Change of basis
EIGENVALUE DECOMPOSITION

- Action of matrix is described by eigenvalues

\[
A = \begin{pmatrix}
\end{pmatrix}
\]

\[
x = \beta_1 e_1 + \beta_3 e_2 + \beta_4 e_3
\]

\[
Ax = A(\beta_1 e_1 + \beta_3 e_2 + \beta_4 e_3)
\]

\[
= \beta_1 Ae_1 + \beta_3 Ae_2 + \beta_4 Ae_3
\]

\[
= \beta_1 \lambda_1 e_1 + \beta_3 \lambda_2 e_2 + \beta_4 \lambda_3 e_3
\]
**MATRIX INVERSE**

- Action of matrix is described by eigenvalues

\[
Ax = A(\beta_1 e_1 + \beta_3 e_2 + \beta_4 e_3)
\]

\[
= \beta_1 Ae_1 + \beta_3 Ae_2 + \beta_4 Ae_3
\]

\[
= \beta_1 \lambda_1 e_1 + \beta_3 \lambda_2 e_2 + \beta_4 \lambda_3 e_3
\]

\[
A^{-1}x = \beta_1 \lambda_1^{-1} e_1 + \beta_2 \lambda_2^{-1} e_2 + \beta_3 \lambda_3^{-1} e_3
\]
ESTIMATION PROBLEM

Suppose \( \lambda_1 = 1, \lambda_2 = 0.1, \lambda_2 = 0.01 \)

\[
Ax = b = \beta_1 e_1 + 0.1 \beta_2 e_2 + 0.01 \beta_3 e_3
\]

I tell you \( \hat{b} = b + \eta \)

Can you recover \( x \)?

(Do on board)

Does this ever really happen?
**CONDITION NUMBER**

- Ratio of largest to smallest singular value

\[
\kappa = \frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}
\]

\[
\kappa = \left| \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \right|
\]

\[
\kappa = \| A \| \| A^{-1} \|
\]

-  \[ A \mathbf{x} = \mathbf{b} \]
  - vs
  -  \[ A \mathbf{x} = \hat{\mathbf{b}} \]

\[
\frac{\| \hat{\mathbf{x}} - \mathbf{x} \|}{\| \mathbf{x} \|} \leq \kappa \frac{\| \mathbf{b} - \hat{\mathbf{b}} \|}{\| \mathbf{b} \|}
\]

Why are these definitions the same for symmetric matrices? What is the condition number of our problem?
DOES THIS EVER HAPPEN IN REAL LIFE?

• No. The situation is never this good.
• Common in optimization: regularizations and IPM’s
• Example: Convolution
CONVOLUTION MATRIX

Condition number: 3,500,000
CONVOLUTION: 2D

The figure illustrates the concept of 2D convolution, where a kernel (K) is applied to an input image to produce an output image. The kernel slides over the input, performing element-wise multiplication and summing the results to produce each output element.
DEBLURRING

Relative difference 0.0092

$K^{-1}$

$K^{-1}$
UNDER-DETERMINED SYSTEMS

• Another problem: What if matrix isn’t even full-rank?

\[ A \in \mathbb{R}^{M \times N} \quad M < N \]

\[ b = Ax + \eta \]

• If the error is bounded ( \( \|\eta\| \leq \varepsilon \) ) solve

\[
\text{minimize} \quad \|x\| \quad \text{subject to} \quad \|Ax - b\| \leq \varepsilon
\]
GEOMETRIC INTERPRETATION

All points on the red line satisfy
\[ y = Ax \]

Point with the smallest \( \ell_2 \) norm
RIDGE REGRESSION

• If the error is bounded ( $\|\eta\| \leq \epsilon$ ) solve

$$ b = Ax + \eta $$

minimize $\|x\|$ subject to $\|Ax - b\| \leq \epsilon$

• This is equivalent to

minimize $\lambda \|x\|^2 + \|b - Ax\|^2$

for some value of $\lambda$

Why? Show on board.
RIDGE REGRESSION

minimize $\lambda \|x\|^2 + \|b - Ax\|^2$

Closed form solution!

$$(A^T A + \lambda I)^{-1} A^T b$$

What does this do to condition number?
RIDGE REGRESSION

$$\text{minimize} \quad \lambda \|x\|^2 + \|b - Ax\|^2$$

Closed form solution!

$$(A^T A + \lambda I)^{-1} A^T b$$

New condition number

$$\frac{\sigma_{\text{max}}^2 + \lambda}{\sigma_{\text{min}}^2 + \lambda}$$
TIKHONOV REGULARIZATION

$$\text{minimize} \quad \lambda \|x\|^2 + \|b - Ax\|^2$$

- Has many names (ridge regression in stats)
- Advantage: Easier to solve new problem
- Improved condition number (less noise sensitivity)
- Parameter $\lambda$ can be set:
  - Empirically (e.g. cross-validation)
  - Use noise bounds + theory (BIC, etc...)
BAYESIAN INTERPRETATION

• Model data formation: write distribution of data given parameters

• Observe data from random process

• Use Bayes rule: write distribution of parameters given data

• Find “most likely” parameters given data

• Optional: uncertainty quantification / confidence
BAYESIAN INTERPRETATION

minimize $\lambda\|x\|^2 + \|b - Ax\|^2$

• Assumptions:
  
  • We know expected signal power: $\mathbb{E}\{x_i^2\} = E$
  
  • Linear measurement model $b = Ax + \eta$
  
  • Noise is i.i.d. Gaussian: $\eta = N(0, \sigma)$

• MAP (maximum a-posteriori) estimate:
  
  $\underset{x}{\text{maximize}} \ p(x|b)$
Bayes’ Rule

\[ p(X|Y) \propto p(Y|X)p(X) \]

Bayes’ Rule

\[ p(X|b) \propto p(b|x)p(x) \]

Bayesian Interpretation

\[ b \sim N(Ax, \sigma) \]

\[ x \sim N(0, \sqrt{E}) \]
Bayesian Interpretation

\[
\max_x \quad p(x|b) = p(b|x)p(x)
\]

\[
p(b|x) = \prod_i \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2} (b_i - (Ax)_i)^2}
\]

\[
= (2\pi)^{-m/2} e^{-\frac{1}{2\sigma^2} \|b - Ax\|^2}
\]

\[
p(x) = (2\pi)^{-n/2} e^{-\frac{1}{2E} \|x\|^2}
\]

\[
\max \quad \exp \left( -\frac{\|b - Ax\|^2}{\sigma^2} \right) \exp \left( -\frac{\|x\|^2}{E} \right)
\]
NEGATIVE LOG-LIKELIHOOD

\[
\text{maximize } \quad p(x|b) = p(b|x)p(x)
\]

\[
\text{maximize } \quad \exp \left( -\frac{\|b - Ax\|^2}{\sigma^2} \right) \exp \left( -\frac{\|x\|^2}{E} \right)
\]

\[
\text{maximize } \quad -\frac{\|b - Ax\|^2}{\sigma^2} - \frac{\|x\|^2}{E}
\]

\text{NLL}

\[
\text{minimize } \quad \frac{\sigma^2}{E} \|x\|^2 + \|b - Ax\|^2
\]
SPARSE PRIORS

• Priors add information to the problem
• Ridge/Tikhonov priors require a lot of assumptions
• Prior only good when assumptions true!
• A very general prior: sparsity
WHAT IS SPARSITY?

• Signal has very few non-zeros: small $\ell_0$ norm
OTHER NOTIONS OF SPARSITY

• “Low density” signals - rapid decay of coefficients

• Fast decay: $= \text{Small Weak } \ell_p \text{ norm}$
DENSE SIGNAL / SPARSE REPRESENTATION: AUDIO

- Sounds produced by vibrating objects
- Energy is concentrated at resonance frequencies of object
- Defined by eigenvalues of the Laplacian of vibrating surface
Echolocation chirp: brown bat

- Bat hears convolution of signal with environment
- Chirps: generate well-conditioned convolution matrices

Gabor transform (STFT)

source: Christoph Studer
DENSE SIGNAL / SPARSE REPRESENTATION: IMAGES

- Approximately Piecewise constant
- High correlations between adjacent pixels within objects
- High variation across edges

Wavelet transform of natural image

Credit: Mark Davenport
NEURAL EVENTS

- Neural potentials: convolution of spike train with kernel

Real recording

Spiking events

Kernel
COMPRESSED SENSING

• Assume $\mathbf{x}$ has only $k \ll N$ non-zero entries

$$\mathbf{y} = \text{nice MxN matrix with } M < N \ast \mathbf{x}$$

• If vector $\mathbf{x}$ is sufficiently sparse, then one can uniquely recover $\mathbf{x}$ from $\mathbf{y}$
**MACHINE LEARNING: OVERFITTING**

\[ A x = b \]

- Features/Data → Model Parameters → Labels

- Happens when you can design the measurement matrix
- More features = better fit
EXAMPLE: POLYNOMIAL FITTING

Noisy data drawn from polynomial what degree is best?

$n=20$
EXAMPLE: POLYNOMIAL FITTING

n=20

model fitting error

degree
EXAMPLE: POLYNOMIAL FITTING

degree = 17
EXAMPLE: POLYNOMIAL FITTING

don’t fit the noise!

bad “out of sample” error
WHY DID THIS HAPPEN?

n=20
WHY DID THIS HAPPEN?

We know we’re overfitting just from looking at this curve. Why?
SPARSE RECOVERY PROBLEMS

used to control over-fitting

\[
\begin{align*}
\text{minimize} & \quad \|x\|_0 \quad \text{subject to} \quad Ax = b \\
\end{align*}
\]

Sparse solution

“Fat” matrix (underdetermined)

Dense measurements

COMPLEXITY ALERT: NP-Complete

(Reductions to subset cover and SAT)

Nonetheless: can solve by greedy methods

• Orthogonal Matching Pursuit (OMP)
• Stagewise methods: StOMP, CoSAMP, etc…
L0 IS NOT CONVEX

minimize $\|x\|_0$ subject to $Ax = b$

minimize $|x|$ subject to $Ax = b$
WHY USE L1?

• L1 is the “tightest” convex relaxation of L0
CONVEX RELAXATION

minimize $|x|$ subject to $Ax = b$

$\ell_1$ ball touches the affine plane
# Sparse Optimization Problems

<table>
<thead>
<tr>
<th>Method</th>
<th>Objective Function</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basis Pursuit</td>
<td>( \minimize ,</td>
<td>x</td>
</tr>
<tr>
<td>Basis Pursuit</td>
<td>( \minimize , \lambda</td>
<td>x</td>
</tr>
<tr>
<td>Denoising</td>
<td>( \minimize , \frac{1}{2} |Ax - b|^2 ) subject to (</td>
<td>x</td>
</tr>
</tbody>
</table>
BAYESIAN LAND!

Basis Pursuit Denoising

\[
\begin{align*}
\text{minimize} & \quad \lambda |x| + \frac{1}{2} \|Ax - b\|^2 \\
\text{What prior is this?}
\end{align*}
\]

Laplace distribution

\[p(x) = e^{-|x|}\]

“robust to outliers”
HOW TO SET LAMBDA?

minimize out-of-sample error

\[ \text{minimize } \lambda |x| + \frac{1}{2} \| Ax - b \|^2 \]

30% test set
CROSS VALIDATION

minimize out-of-sample error

choose lambda to minimize test error

\[
\minimize \lambda |x| + \frac{1}{2} \|Ax - b\|^2
\]
CROSS VALIDATION

minimize **out-of-sample** error

training data

[Graph showing error vs. model complexity with two curves: training error and true error]

test data

idealistic curves: no sampling noise
CROSS VALIDATION

- K-fold CV
- leave-one-out CV
- random sampling CV
AFTER MODEL SELECTION

minimize \( \lambda |x| + \frac{1}{2} \| A_{tr}x - b_{tr} \|^2 \)

de-biasing

minimize \( \frac{1}{2} \| A_{all}x - b_{all} \|^2 \)

subject to \( x \in C \)

ensemble learning

minimize \( \lambda |x| + \frac{1}{2} \| A_1x - b_1 \|^2 \)

minimize \( \lambda |x| + \frac{1}{2} \| A_2x - b_2 \|^2 \)

\[ \vdots \]

minimize \( \lambda |x| + \frac{1}{2} \| A_Kx - b_K \|^2 \)

These methods work best on small problems!
CO-SPARSITY

\[ \text{minimize} \quad \lambda |\phi x| + \frac{1}{2} \| Ax - b \|^2 \]

- Sometimes signal is sparse under a transform

\[ \text{minimize} \quad \lambda |v| + \frac{1}{2} \| A\phi^{-1}v - b \|^2 \]

- When transform is invertible, can use synthesis

- Otherwise, use the analysis formulation

- The thing in the L1 norm is sparse!
EXAMPLE:
IMAGE PROCESSING
IMAGE GRADIENT

Neumann
\[
\begin{pmatrix}
u_1 - u_0 \\
u_2 - u_1 \\
u_3 - u_2 \\
u_4 - u_3
\end{pmatrix}
\]

Circulant
\[
\begin{pmatrix}
u_1 - u_0 \\
u_2 - u_1 \\
u_3 - u_2 \\
u_4 - u_3 \\
u_1 - u_4
\end{pmatrix}
\]
TOTAL VARIATION

\[ TV(x) = \sum |x_{i+1} - x_i| \]

- Discrete gradient operator
  \[ \nabla x = (x_2 - x_1, x_3 - x_2, x_4 - x_3) \]
- Linear filter with “Stencil” \((-1, 1)\)

\[ TV(x) = |\nabla x| \]

Is this a norm?
TV IN 2D

$$(\nabla x)_{ij} = (x_{i+1,j} - x_{i,j}, x_{i,j+1} - x_{i,j})$$

Anisotropic  $| (\nabla x)_{ij} | = |x_{i+1,j} - x_{i,j}| + |x_{i,j+1} - x_{i,j}|$

Isotropic  $\| (\nabla x)_{ij} \| = \sqrt{(x_{i+1,j} - x_{ij})^2 + (x_{i,j+1} - x_{ij})^2}$
TOTAL VARIATION: 2D

\((\nabla x)_{ij} = (x_{i+1,j} - x_{ij}, x_{i,j+1} - x_{ij})\)

\[TV_{iso}(x) = |\nabla x| = \sum_{ij} \sqrt{(x_{i+1,j} - x_{ij})^2 + (x_{i,j+1} - x_{ij})^2}\]

\[TV_{an}(x) = |\nabla x| = \sum_{ij} |x_{i+1,j} - x_{ij}| + |x_{i,j+1} - x_{ij}|\]
IMAGE RESTORATION

Original

Noisy
TOTAL VARIATION DENOISING

\[ \lambda \| \nabla x \|^2 + \frac{1}{2} \| u - f \|^2 \]

\[ \lambda | \nabla x | + \frac{1}{2} \| u - f \|^2 \]
\[
\lambda \| \nabla x \|^2 + \frac{1}{2} \| u - f \|^2
\]

\[
\lambda | \nabla x | + \frac{1}{2} \| u - f \|^2
\]
TV IMAGING PROBLEMS

Denoising (ROF)  \[ \text{minimize} \quad \lambda |\nabla x| + \frac{1}{2} \| u - f \|^2 \]

Deblurring  \[ \text{minimize} \quad \lambda |\nabla x| + \frac{1}{2} \| Ku - f \|^2 \]

TVL1  \[ \text{minimize} \quad \lambda |\nabla x| + |u - f| \]
MACHINE LEARNING TOPICS
LINEAR CLASSIFIERS

training data

feature vectors
vectors containing descriptions of objects

labels
+1/-1 labels indicating which “class” each vector lies in
LINEAR CLASSIFIERS

goal
learn a line that separates two object classes

what line is best?

\[ d^T w = 0 \]
SUPPORT VECTOR MACHINE

**SVM**
choose line with maximum "margin"

$$\text{margin width} = \frac{1}{\|w\|}$$

$$d^T w = 1$$
$$a^T w = 0$$
$$d^T w = -1$$
SUPPORT VECTOR MACHINE

margin width = \frac{1}{\|w\|}

hinge loss

penalize points that lie within the margin

\sum_i h(d_i^T w)

\begin{align*}
d^T w &= 1 \\
\frac{1}{\|w\|} &\quad +1 \\
-1 &\quad \text{blue points}
\end{align*}
SUPPORT VECTOR MACHINE

margin width = \( \frac{1}{\|w\|} \)

**hinge loss**
penalize points that lie within the margin
\[
\sum_i h(d_i^T w)
\]

short-hand
\[
h(Dw)
\]

combined objective
\[
\text{minimize} \quad \frac{1}{2} \|w\|^2 + h(Dw)
\]
LOGISTIC REGRESSION

\[ f(x) = \frac{e^x}{1 + e^x} \]

\[ P[y = 1] = \frac{e^{d^T w}}{1 + e^{d^T w}} \]

\[ P[y = 0] = 1 - \frac{e^{d^T w}}{1 + e^{d^T w}} = \frac{1}{1 + e^{d^T w}} \]
LIKELIHOOD FUNCTIONS

probability of observing label given data

case: observed $y=1$

$$\mathbb{P}[y = 1] = \frac{e^{d^Tw}}{1 + e^{d^Tw}}$$

$$NLL(d) = \log(1 + e^{d^Tw}) - d^Tw$$

case: observed $y=0$

$$\mathbb{P}[y = 0] = 1 - \frac{e^{d^Tw}}{1 + e^{d^Tw}} = \frac{1}{1 + e^{d^Tw}}$$

$$NLL(d) = \log(1 + e^{d^Tw})$$
LIKELIHOOD FUNCTIONS

probability of observing label given data

case: observed \( y = 1 \)

\[
P[y = 1] = \frac{e^{d^T w}}{1 + e^{d^T w}}
\]

\[
NLL(d) = \log(1 + e^{d^T w}) - d^T w
\]

case: observed \( y = 0 \)

\[
P[y = 0] = 1 - \frac{e^{d^T w}}{1 + e^{d^T w}} = \frac{1}{1 + e^{d^T w}}
\]

\[
NLL(d) = \log(1 + e^{d^T w})
\]
**LOGISTIC OBJECTIVE**

probability of observing label given data

\[ f(w) = \sum_{y_i=1} \ell(d^T x, 1) + \sum_{y_i=0} \ell(d^T x, 0) \]

shorthand

\[ f(w) = L(Dw, y) \]

shorthand

\[ L(z, y) = \sum_i \ell(z_i, y_i) \]
NEURAL NETWORKS

\[ W_1 \quad W_2 \quad W_3 \]

\[ d_i \quad Yi \]

input layer

hidden layer 1

hidden layer 2

output layer
ARTIFICIAL NEURON

\[ f \left( \sum_{i} w_{i}z_{i} \right) \]
NEURAL NETWORKS

\[ f(f(f(d_i W_1)W_2)W_3) = y_i \]
TRAINING

\[ f(f(f(d_i W_1) W_2) W_3) = y_i \]

least-squares loss

\[ \ell(z, y) = (z - y)^2 \]

“cross entropy”/log-likelihood loss

\[
\begin{align*}
\min_{W_1, W_2, W_3} \sum_{i} \| f(f(f(d_i W_1) W_2) W_3) - y_i \|^2 \\
\min_{W_1, W_2, W_3} \sum_{i} \ell(f(f(f(d_i W_1) W_2) W_3) - y_i)
\end{align*}
\]
MULTI-CLASS NETWORK

d_1^i \rightarrow \text{target label} 0

d_2^i \rightarrow \text{representation} \text{"ones hot"} 0

d_3^i \rightarrow Y_i 1

\text{interpretation?}
MULTI-CLASS NETWORK

```
| d_{i}^{1} | .75 | .09 |
| d_{i}^{2} | -.1 | .01 |
| d_{i}^{3} | 1.5 | .9  |
```

target label
```
0
0
1
```

“ones hot” representation

 probability interpretation
SOFTMAX

- Make non-negative: $e^{z_1}$
- Compute normalization: $\sum_i e^{z_i}$
- Probabilities:
  - $\frac{e^{z_1}}{\sum_i e^{z_i}} = 0.09$
  - $\frac{e^{z_2}}{\sum_i e^{z_i}} = 0.01$
  - $\frac{e^{z_3}}{\sum_i e^{z_i}} = 0.9$
CROSS-ENTROPY LOSS

target label

\[ \ell(x, y) = - \sum_k y_i^k \log(x_i^k) \]

“cross entropy loss” is negative log likelihood

\[ \text{NLL} = - \log(0.9) \]