CS 170A Notes

Setting up Jupiter
>> jupyter notebook in /CS-170A

W 1 M Lec 9-26-16
Stott Parker: stott@cs.ucla.edu
Office: 3532-H Boelter Hall
Office Hours:

- After every class (6 PM onwards)
- Tuesday (1:30-3:30 PM)
- By appointment

Course about mathematical linear algebra with a computing flavor to it

- Created because students in computer science didn't like EE 103.
- More focused on algorithms and this course is more about modeling and the higher-level view of what is going on.

MATLAB is a wonderful tool and has served EE and CS admirably for a number of years

- Mathworks had a seminar and we had a bunch of people coming in from industry and engineering will become a sequence of steps using a series of tools.
- Working with a toolchain to do engineering development and product development

- Methods for numeric and symbolic computation
- Matrix algebra
- Statistics
- Floating point
- Optimization
- Spectral analysis

Quiz - October 12 (covering Matrix Algebra)
Midterm (covering Least Squares, PCA)

Final
Option #1: Final Exam 11:30 AM - 2:30 PM
Option #2: Project Report Due 11:55 PM

We get you up to speed with linear algebra tools and take the quiz.

- More advanced material and then the final.
- The final is at 11:30 AM
- If you want to do a project instead, we need the proposal by November 2nd

- Projects like sports data analysis (Novak Djokovic's serve and returning stats)

Lab & Homework assignments	25%
Quiz	5%
Midterm exam	30%
Project or Final exam	40%

Assignments will require programming using Jupyter and a numerical computing environment

Acceptable computing environments:
- Matlab (no need for any Matlab toolboxes)
- Octave (= 'GNU Matlab')
- 'PythonLab' (= Python with Matlab-like extensions)
- These environments are lifelong learning environments

Matlab: The popular matrix algebra computing environment, from MathWorks.com
- In Matlab, put semicolons at the end to tell it NOT to send anything.
- You get used to putting semicolons at the end of statements

W 1 W Lec 9-28-16
- Review matrix algebra for the quiz
- Kind of a long cheatsheet and hopefully it is fun
- Overview of the course from a matrix algebra perspective
- Octave has the same basic capabilities of Matlab
- Be able to put in any sort of matrix algebra expression and know what is going to happen.

Octave Examples
- Indices are start from 1
- Matrix(row, column)

```
>> rand(3)
ans =

   0.238347   0.565565   0.512604
   0.692750   0.048431   0.729124
   0.872565   0.935593   0.994081

>> A = rand(3)
A =

   0.86454   0.53379   0.89363
   0.26302   0.75227   0.89724
   0.59912   0.40901   0.94188
```

```
>> A'
ans =

   0.86454   0.26302   0.59912
   0.53379   0.75227   0.40901
   0.89363   0.89724   0.94188

>> 'abc'
ans = abc

>> A(2,3)
ans =  0.89724

>> inv(A)
ans =

   2.38232  -0.95733  -1.34832
   2.02140   1.94520  -3.77085
  -2.39317  -0.23575   3.55684
```

What is the Matrix?

- a Datatype -> A hub of constructs from different fields

The Matrix is an Array of Numbers

- If we look at the dictionary, we will discover there is a lot of definitions for the matrix and matrices can become more general than they were in the past.
- Distinguish between a matrix (2D array) vs a tensor (ND array)

The Matrix is a Datatype (Class)

- Stolen from a game engine and all the matrices here are 3D because they are 3D transformations
- All C++ code so it is fast when executed
- We can think of matrix algebra as a class with a set of methods and algebraic operators
- Constructors to build matrices and it is a better way of thinking about what a matrix is
- Makes it clear what the interface is and so forth

Matlab - Numeric Matrix datatype

- That is the usual syntax for creating a small array in Matlab.
- 2x2 matrix with row concatenation operators
- Semicolons are vertical concatenation operators
- All of these things are very standard operators or constructors
- A = eye(4) % a 4x4 identity matrix (terrible pun)

```
A =
```

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Matlab is a data science language

If you are used to Python, it is similar to that

- The slices really are powerful and you just put colon to represent all of the values
- Do things like all the elements in the array that are larger than 3
- Turns array into a boolean pattern of trues and false
- Use Matlab notation and this does that for numerical analysis
- Mathematicians write everything in this notation
- Not a Matlab class!

The Matrix is … an Algebra

- A set of objects and operators that is just defined with two operators.
- If you look at the definition, it is just defined in those terms of those things.

Algebra

- Follows a set of axioms that I have heard of and these axioms, so we can break down identities among matrix expressions using formulas for individual operators and the axioms that we see at the top.

Complex Conjugate

- 1 + 3i (its complex conjugate is 1 - 3i)

```
>> (1+3i)'
ans =  1 - 3i
>> z = (1+3i)
z =  1 + 3i
>> sqrt(z' * z)
ans =  3.1623
```

Find all the most interesting identities and these are all listed here

- It goes through a lot of identities and at the end, if we go to the quiz, this is a dry run.
- Go through lots of identities and check your understanding of things with the quiz at the end

```
>> intrand = @(m,n) round (100 * rand(m,n))
intrand =

@(m, n) round (100 * rand (m, n))
```

```
>> intrand(2,3)
ans =

  17   3  26
  79  38  87

>> n = 1024
n =  1024
>> A = intrand(n,n); //Need the semicolon to avoid a lot of output
```

Block Matrix Algebra

- Used for scalable algorithms and stuff

Strassen's Algorithm

- Instead of using 8 products, you could break it down into 7 by adding, subtracting, and multiplying in a clever way
- Much faster than O(n^3): O(n log_2 (7))

Asymptotically, the complexity is O(n^(log_2(7)) = O(n^2.81)

```
t0 = time; C = A * Bl;time - t0
ans = 27.403
```

Block LU Decomposition

- Apply them on a block scale and break this maria into a LU decomposition

Sylvester's Determinant Identity (one of many)
Theorem if A, B are matrices of size p x n and n x p, then
det(I_n + BA) = det(I_p + AB)
where I_n is the n x n identity and I_p is the p x p identity

A determinant is the result of taking the diagonal elements

- Since it is a simple object, it is easy to measure the volume
- If this is troublesome, plug in some matrices and see what happens!
- You have Matlab or Octave and you can compute this really easily

```
>> Id = eye(3)
Id =

Diagonal Matrix

  1  0  0
  0  1  0
  0  0  1
```

```
>> det(Id)
ans =  1
>> Z = zeros(3)
Z =

   0   0   0
   0   0   0
   0   0   0

>> det(Z)
ans = 0
>> Foo = diag( [1], [1 2; 3 4])
Foo =

   0   1
   0   0

>> Foo = blkdiag([1], [1 2; 3 4])
Foo =

   1   0   0
   0   1   2
   0   3   4

>> det(Foo)
ans = -2
>> Foo = blkdiag( [3], [1 2; 3 4])
Foo =

   3   0   0
   0   1   2
   0   3   4

>> det(Foo)
ans = -6.0000
>>
```

There are many Determinant Identities

- Lots of properties like these and they all seem reasonable
- Transpose is essentially rotating it and it winds up the same in the end

The Matrix is a Thread in the History of Mathematics

Linear equations

$ax + by = e$

cx + dy = f

- Where did they come up with these equations 300 years ago?

W 1 Dis 9-30-16
Outline:
Basic Keywords
Det + Inv
Inner Product

Schur Complement
Scour-Sylvester Identity
LU Décomposition

Matrix is also an operator
- This is not just a number spectrum, it will be your imagination

Matrix algebra entails if you have addition, subtraction, AND multiplication

- If you multiply two substitution matrices, you get another substitution matrix
- Unitary matrix

W 2 M Lec 10-3-16
- Good books to read
- Erwin Kreyszig, *Advanced Engineering Mathematics*
- Christopher Bishop, *Pattern Recognition and Machine Learning*
- Kevin Murphy, *Machine Learning: A Probabilistic Approach*

Scalar Product
- Find the transpose to be able to do that and then you can define all of these invariants and properties of the scalar product
- The axioms that we talked about were present in all of this and these are pretty much what you would expect
- If <x, y> = 0, we say **x** and **y** are **orthogonal**

Vector Norms
- If we take the dot product with itself, we get a measure of the size of it.
- The Euclidean norm is the transpose times x and we find the Euclidean norm
- Let p get larger and they start to dominate the sum.
- As p goes to infinity, you get the general L^p norm
- The scalar product of x transpose times x has this complex absolute value squared.
- Even if x is a complex value, the product of its conjugate gives us something real (always!)

- Thus, it is important to define the transpose in that way.
- The scalar product is always going to be nonnegative

Vector Rotation

- This is crucial in understanding all sorts of things and understand what a linear transformation does.
- Here you see what happens if you are applying a rotation to the blue vectors.
- If you do that, you get these nice unit vectors and if you apply a rotation with angle theta, you wind up with r_1 and r_2
- They are what you would expect since they are sines and cosines
- If we take that vector **v** and rotate it, we get **w**.
- How did we get that and why do we care?
- We are expressing **v** as a sum as a linear combination of **e**_1 and **e_2.**
- Instead of computing the rotation from scratch, we are going to decompose it into two parts and these are easy to update.

Bases and Coordinate Systems

- Set of vectors using linear transforms and you can represent vectors using coefficients times those things.
- Just for simplicity, select the basis elements to be **orthonormal.**

Think of the Euclidean system for now and you can see they are all orthogonal to one another.

Vector Spaces

- Take a vector v and think about it in terms of being v_1 through v_n
- Vector space is all of the vectors you can construct in that way.

Example: Color Models

- Red is x-axis
- Green is y-axis
- Blue is z-axis
- If you add these things, you get interesting other colors
- Represent all the primary colors in this system very nicely.
- Top right thing is the complete saturation of color (white)

Black is the absence of color, while white is the complete presence of color

- The tradition is to use 0-255 and this gives intensity for each of the three dimensions that we have.

- Blue is very saturated representation of blue color (has 255 in the B component)
- Navy is half as much (the closer you get to 0, the darker it is)

- Characteristics of gray is that you have three identical versions of gray.

- You go from black at the origin to white and any point in between will have R, G, B identical

Example: Discretized Functions
plot(x, diff(fx), ‘r-‘)

- Do all sorts of calculus computations using vectors

Vector Projection

- When people do scalar products, it isn’t intuitive and once you have thought of the scalar product as a projection operator, then this sort of thing is obvious.
- How do we prove this lambda is the right value for this projection?

Vector Coordinates and Vector Projection

- **v** is the sum of constant factors that are the coefficients (scalar product)
- v_i = <e_i, v>

Simultaneous Vector Projection
Question: given *m* vectors v_1, … v_m, how can we find their coordinates relative to the dimensions e_1, …, e_p?

- Write a for loop for each of the subscripts for e’s and v’s to get the resulting matrix.
- You put it in matrices at UCLA!

Strassen:

- Complexity is 2.81, while normal complexity is n^3

Orthogonal Matrices

- A coordinate system is a matrix with all the basis elements and insist that those basis elements are *orthonormal*.
- If we do that, we wind up with a very nice matrix called an orthogonal matrix
- Orthogonal matrix is an orthonormal basis
- No distinction between the two

Orthogonal matrix will have nice inverses

- Q’Q = Identity matrix where Q’ is the orthogonal matrix and Q is the original matrix
- You will wind up getting a square matrix and the only ones that survive this are the diagonal ones
- Q transpose must be the inverse of Q
- You get a lot of power putting coordinate systems into a matrix like this

Rotations: an essential type of Orthogonal Matrix

- If we take a rotation by theta as a linear transform, it is of a certain format.

Inverse of a rotation by theta should be a rotation of minus theta

Reflections = Rotations with a ‘Flip’ (Mirror)

- What is the determinant of a rotation?
- 1 = (cos^2(theta) + sin^2(theta))
- What is the determinant of the reflection?
- -1 = (-cos^2(theta) + sin^2(theta))

n-dimensional Rotations

- Theorem:
- Any n-dimensional rotation can be expressed as a product of 2D rotations
- n(n-1)/2 pairwise (2-dimensional) rotations

3D Rotation with Euler Angles

- If you are going to try to write into these, you can use the blkdiag
- As bubble sort, can you change the order of pair-wise comparisons?
- Yes, but if you do the same pair twice, it isn’t going to work.
- Can we produce a more satisfying sequence of pairs?
- Yes!

Pitch, Roll, and Yaw

- Pitch is where you rotate vertically
- Roll is rotating around the axis of motion
- Yaw is where you rotate horizontally

Compose those into a large rotation!

Orthogonal Matrices: Rotations and Reflections

- The matrix will either be a rotation or reflection, and we want something similar to cover that

Theorem: Any n-dimensional orthogonal matrix can be decomposed into a sequence of n(n-1)/2 pairwise rotations or reflections

If you thought orthogonal matrices were abstract, it’s not. You need it instantly for eigen-decomposition

W 2 W Lec 10-5-16

- Gene Golub, Charles van Loan, Matrix Computations, 2013

Course Project

- Following techniques that we are going through in class like eigenvalues analysis, etc.
- A few actual titles from things and we have edited some of them and people wanted to get League of Legends data (LMFAO!)

Things like image brightening

- Since it was a warmup, there is something on global warming

simplify

- It trims it down to the identity matrix
- Algebraic systems know all about trigonometry and calculus
- They use all sorts of facts about those things
- You can calculate stuff this way

Is there global warming given all this data input?

- Quiz is next Wednesday so study your ass off!

2nd page is going to be more like slices and array options

- Be able to read slice code and things like that.
- On Friday, they will go through an actual sample quiz and it will be very much like what you see a week from today.
- The review is a pretty good model of what is going to happen.

People generally don't worry about the matrix where you get the eigenvalues for.

- The result is pretty, beautiful, and works.
- The minute you start to deal with non-symmetric matrices will work.
- Once you have the eigen-decomposition with all the eigenvectors, you are set and you can do all kinds of stuff

Pencil-shaped ellipsoid, so if we plotted this, we would see something that looks like a very stretched out ellipsoid.

Linear Transforms

- Red dots are the result of applying matrix A to the blue dots.
- You could see all these magenta arrows and we put the letter A there and it wasn't really necessary.
- A transform transforms it into another vector (transforms a point into this red vector)
- This is what is really happening
- For any of these points on a circle, they get mapped to a corresponding point on the ellipse.

Q. Could A be decompressed into both a squishing down and a rotation?
A. Yes! That is stuck at the origin.

- Generates the same thing without all the arrows and draws in all the eigenvectors and values.
- The original thing if it is e_1 or e_2

Huge arrows on the vectors and you can take this code and print out eigenvectors and values.

Dilation

- If the matrix only has a diagonal part, then it has to do something like this.
- This particular matrix maps x into 3x, and it leaves the y-axis alone.
- This tells us where the dilation comes from and it tells us the stretching factors.

Matrix Norms and Eigenvalues

- They look just like the ones we did for vectors
- These are cleverly different in that they are talking about a norm of a matrix
- The norm of a matrix is a bit more confusing
- Matrix A would have an L2 norm of 2.7 and these other things like row sums or col sums
- We want to measure how much A does to vectors.
- If we apply A to a vector, how much larger is the vector afterwards.
- If we think about this, the most that I can get out of the vector by applying A to it is the long axis after it is an eigenvector.
- The length of the longest axis is lambda max.
- The matrix norm is about worrying about the effect of a transform

- Here we are using the symbolic stuff and people are looking for the two eigenvalues of a 2 x 2 matrix.
- In principle, we can do this for all these other operations but it is in fact easier than that.

2 x 2 Real Symmetric Matrix Decomposition

- This is what you get with a 2 x 2 real matrix
- Solve the equations by the eigenvalues and that is what you get
- Not really the best way to do things anymore.

Find a value y in the form of this rotation and this is going to be our transformation T in 2D

- Pick c and s be the result of 0
- It will change the x values into something else and we want to repeatedly apply transformations to eliminate all the off-diagonal elements.
- Do the same thing iteratively
- Eliminate everything except the tri-diagonal
- From there, repeatedly make the off diagonal things smaller

Simple Jacobi method for Real Symmetric Matrices

- Uses the ‘fro’ norm (Frobaneous norm): Sums up the squares of the off diagonal elements
- Compare that with some limiting accuracy (eps)
- Once the matrix has a nice format, you can build an algorithm to work on it well

Unitary is almost the same as Orthogonal

- Orthogonal matrices are real though.
- For our purposes, we need to insist that orthogonal matrices are real

Positive Definite: Symmetric by definition
- Real matrix that is symmetric and has A real and positive eigenvalues

Hermitian can be complex and for a lot of people, these two things are never different.

Blue matrices are the sequence of eigenvalues
- Like complex arithmetic but in a matrix scale.
- Unit complex things and purely real things with polar coordinates.

W 2 Dis 10-7-16
- Image Processing in Matlab

load mandrill
Mandrill = ind2rgb(X, map);

- Changing color to grey
- Unsigned integer of 8 bits
- You can also do it in char

Gray is an image matrix and that is all it does.

- Anything different from 0 would be all white
- Finally, it becomes all white except at certain points
- As you lower it, you will see it change brightness

0. Color Models
- Why can't CbCr exceed 255 or go below 0
- RGB would need to start as a negative value, which isn't true.
- The sum of a row cannot be negative.
- Same idea

0. ghcn_script
- You get this data and these 4 lines here are important.

CS 170A Quiz Preview
- A.' is a regular transpose
- It cannot be unitary.

0. Matrix Algebra and Eigenstructure

a. False
Orthogonal matrix is det - 1, there is no way to rotate it
b. False
c. True?
d. True

e. ?
f. False
Diagonal - only non-zero numbers in its diagonal
g. Can a linear transformation have three variables? Yes!
i. True
j. True
k. True
l. False
m. False, it is only true iff you have a Q that can diagonalize both of them. A and B are real symmetric iff they have a common Q
n. False
o.
p.

0. c. Hilbert matrix is square (All special matrices are square)

0. c. True
- The rotation is NOT going to change the length

d. True
- You can figure it out from the actual matrix

0. sqrt(6) and 1
0. 1) False

2)
3)

Key Terms
Symmetric, Hermitian, and their skew versions and normal

Diagonalizability
When do they commute
SVD
Diagonal structure

Unitary, Orthogonal

Diagonalizability
Determinant
Preserving inner product
Rotation, reflection, (odd number, even number)
Eigen structure
2x2 matrix structure

Positive definite, positive semidefinite
Negative Definite, Negative semidefinite

SVD of rectangular

Eigen decomposition
Spectral theorem

det(A-IL) = 0

Invertibility
Non-singular
non-zero determinant

Matlab code

Kronecker Product, Tensor product
Hadamard product, Schur product
Trace, Determinant relations
Tr(ABC) = Tr(CAB) = Tr(BCA)
Direct Sum

LDU, LU

Block matrix (diagonal, trace, rotational block)

W 3 M Lec 10-10-16

- You cannot write everything in multi-line functions in Octave
- Have it running with Jupyter?
- In Jupiter, you can run a function in one cell and you can do it inline.
- For Octave, you have to put an end in the function and Octave doesn't have the same limitation?

[U, S, V] = svd(A)

- You could fit in any matrix as input and you could get out three nice things about this.

A = rand(3, 2); [U,S,V] = svd(A)

- S has the same shape as matrix A that we gave this
- norm(A - U * S * V')

Code

```
>> A = rand(3, 2); [U, S, V] = svd(A)
U =

  -0.798284   0.601109   0.037554
  -0.578373  -0.747711  -0.326210
  -0.168008  -0.282129   0.944551
```

S =

Diagonal Matrix

```
   1.22731         0
         0   0.17153
         0         0
```

V =

```
  -0.75972   0.65025
  -0.65025  -0.75972
```

```
>> norm(A - U * S * V')
ans =    4.9081e-16
```

- SVD is better than a lot of exceptions and it is pretty robust

What are U and V?

- U and V look like rotation matrices while S is a Diagonal Matrix again
- This is similar to the stuff we see for eigen-decomposition
- Values in SVD are that the eigenvalues are descending
- That is why the flip is there
- SVD is a generalization of the eigen-stuff we learned

Unitary and Hermitian matrices:

- symmetric matrices are **real**
- Hermitian**:** A is NOT necessarily real and it isn't necessarily the case that the ordinary transpose is equal to A

Theorem A matrix A is normal iff it has a decomposition
A = U D U'
where D is diagonal and U is unitary.

- Good for polar coordinates because D is like the theta, and U is the radius

A (possibly complex) square matrix U is **unitary** if U'U = UU' = I.

- every orthogonal matrix Q is unitary: QQ' = I.
- a **complex unit value** (a value like z = e^(i * theta)) is unitary: z !z = 1.
- Unitary does come up so it is important to know that it is one step more general.
- Are all Unitary matrices products of rotations and reflections?
- It should be decomposable

Interesting Matrices - the Fourier Matrix?

- Vector * Matrix
- Taking the Fourier transform of an identity matrix will give back the Fourier matrix
- If we did the Fourier transform of the identity matrix, we will get the Fourier Matrix

A is a symmetric matrix, so U and V wind up being the same

- What we had with the eigen stuff before because the S is like the eigenvalues, while U and V are the eigenvectors and they are actually the same because of the fact that the matrix is symmetric

SVD: Definition
Theorem: If A is an n x p matrix, then A = U S V'
where U and V are unitary matrices and S is diagonal

- U is square because it is a unitary matrix, V is square because it is a unitary matrix
- A unitary matrix is square
- Assume S is the same shape as A
- The shapes of the matrices can vary quite a bit

- Use n and p because that is good practice to look at things that way.

The SVD as a Transformation

- V' = inv(V)
- Transform it into U S coordinate system
- Rotate the U coordinate system and what we get is this picture.
- Almost exactly like the eigen-decomposition
- Start out with one coordinate system and rotating into another coordinate system u
- If you look at the diagram, look at the expansion and then dilate it with a singular value and unrotating it to get the U system back.

One of the eigenvalues is 0.69 and you can get those red vectors there, which are the U eigenvectors

- People just call them eigenvalues or eigenvectors

Throw away the last Eigenvector of U because it will get zeroed out.

- Why is it economy?
- You throw away eigenvectors that don't matter.

- The first k of these sigmas will only keep the first k of the u's and the first k of the v's
- k can be 2 or 10, and then you can throw away hundreds of things after that and not lose very much.

Image SVD Demo

- We threw away 3/4ths of the image and we still see a pretty good clown
- Throw away first 60 eigenvalues and we still see a pretty good clown
- SVD allows you to throw away a lot of stuff in the matrix
- This is also called “dimensionality degradation”

A lot of life is redundancy and here we have 500 things down to less than 1/10 of that so 90% of the image is thrown away

Pseudo inverse -> built in Matlab function

W 3 Dis 10-14-16

- Sigma values are real, nonnegative, and array descending order

svds(A, 2)

- Look at U2 and the first three rows

norm(A - U2 * S2 * V2’)

- Approximation errors will increase over time.
- Frobenius norm is the sum of all entries in the difference matrix

Look at the simplest case (Identity Matrix) and then give a counterexample

Frobenius norm: The default norm is the maximum value of this vector

- Spectral norm and the maximum value of the eigenvalue
- Infinite norm = take all the rows and each row
- L_1 norm is the opposite because it is a max col sum rZZxzxzCzCow
- Frobenius norm is easy to play with

A - A^(k) can be brought to this form and if you put any value into k and take the summation, what will happen?

- The difference will be large if the value is small

www.engr.uvic.ca/~seng474/svd.pdf

- The notations are different but the rotations are slightly different

Pseudoinverse

- If it is square and invertible, you can take the inverse
- When can we NOT use this formula for P i.e. P = (A’A)^-1 A’

The pseudo inverse of the pseudo inverse is the original matrix: **(A^+)^+**

Identities

- This satisfies almost all the properties of the inverse

S is skinny, S' is fat

- You can implement Pseudoinverse like this

W 4 M Lec 10-17-16

- Midterm on Oct. 31
- Engineering or data analysis - you will use these tools in different ways and they will be very useful
- Finish up with SVD a little bit and get into Least Squares
- Nice history of Least Squares and go into the a bit
- We will let you use .m files for Matlab or if you cannot get the notebook working, you could submit PDFs
- Don't send emails in PDF formats to Parker!

There is this nice result about the rank-k approximation

- Throw away everything except the first k columns
- Take X and break it down keep the first three columns of each of these matrices
- For U and V, keep the first three eigenvectors
- We used this when looking at the clown, and this kept the columns of the SVD and displayed the results.
- If we kept three columns and multiplied these back together, we get a product that is an approximation of X
- If the singular values are smaller, then this is a pretty good approximation

SVD HW is posted in the zip file

- States this result if you take matrix A and subtract its rank-kth approximation
- Equal to the sum of the squared eigenvalues
- Only keep the first k things and all that remains after we subtract it is that sum!
- Your job in the HW is to prove this and this gives you a lot of the proof right there.
- Full proof.

LSI

- Use the rank-2 approximation of this matrix
- This gives us two eigenvectors U and V along with two little singular values
- Use these two pairs of eigenvectors to do a 2D visualization of the data

Book 17 is around "Algorithms"

- Winds up close to this two-dimensional projection to the terms that you see there.
- Pair of eigenvectors for u turn into the blue dots for Theory
- Project the data that way

- Captures some relationships between terms in the books that you would not expect normally
- **Take the different matrices - baseball and statistics matrix and do the same thing**
- People who are really good at slugging should be near the slugging statistic, for example.

LSI - Latent Semantic Indexing - in Visualization

- If you look at it as a decomposition, the ith row is automatically associated with these two entries.
- Jump between the labeling and the decomposition, rank-k approximation.
- People were really excited because they thought they could turn the World Wide Web and make Google based on this system
- Not one of the great success stories of the web but people were excited about this for a while.

The Pseudoinverse

- By having the SVD, we construct the inverse easily

If a matrix system is a square, you won't have a normal inverse.

- Pseudoinverse is what we need for least squares
- It is the way you solve least squares

The Dwarf Planet Ceres

- Upper blue part of this pie chart - largest planetoid in this solar system up to Uranus
- Significant as an astronomic object.

Baseball

- Random projection: random weightings and plot people where they wind up in.
- Take one random weighting to get your x, take another random weighting to get your y.
- The people who are good at everything are outliers.
- Barry Bonds is way out there and he isn't in the Hall of Fame.

Least Squares

- Take system of equations and find the best solution even if the system isn't squared.
- We cannot just use the inverse of A to solve this.
- Make the system square by simply multiplying both sides by A'
- If we do that, the righthand side is a vector and the lefthand side is a matrix multiplied by coefficient **x**
- Multiply both sides of this by the pseudoinverse of A and you wind up with this solution.

A * coeffs = b

Basic notions of least squares, but for the homework, this should be plenty.

Normal Equations

- Minimize the difference between A**x** and **b** and this will be the squared error
- Just taking a set of vectors where we are taking the ith value of x and the ith value of b and subtracting those to get an individual error.
- Sum up those squares to get the total squared error.
- This will be the sum of squares using the vectors of dot products.
- Not hard to write out exactly what this squared error is.
- We need to minimize that quantity and expand it.
- Do some derivation on what the solution should look like.

We need to somehow minimize each of these things and how can we minimize that?

- Basically a quadratic expression if you want to find the least value of the quadratic expression
- Take this matrix expression and then compute its derivative

For each of the jth positions, we would get **j** entries on the right.

- If you ignore the 2 for a minute, it is the normal equation!

W 4 W Lec 10-19-16

- Pick subset of the data to compute the ratio of y/x, but this requires some subset of the data and we aren't guaranteed of finding the best estimate of the slope
- Least squares picks the slope that minimizes the sum of these distances
- Are there other ways of measuring the best fit?
- Yes! This is the best, classical approach of measuring good fit.

- Take the 1 column here and dot it with the x row, we get the sums

A'A is easy to write down explicitly.

- For inverses, we can pull these out and involve the determinant of this
- The determinant of this will be the sum of x_i squared

- Good example of geometry because it is natural, geometrically speaking
- Encoding all the examples we know about linear algebra and solving that way

Anscobe's Quartet

- Each of these sets of data points will show a different notion of what would be a good fit.

Gives us the error for each position of i and if we take the sum of squares, it is the residual sum of squares (RSS)

- If you divide it by m, it is the MSE

R-Squared:

- How to compute R^2 and to figure if it is near 1

Midterm is on Halloween - 12 days from now

- We will have more time to go back to this stuff later on

- Gigantic sepals
- Sepals are the lower level flowers and these things are quite large, while the petals can be quite tiny.
- Iris setosa has gigantic sepals at the basin
- The petals here*
- The effects of the three different kinds of irises on these variables is that the slope of these things tend to be positive.

Normalization

- Referred to as the z-scores of x
- x bar would be 70, 80, 90
- Standard deviation is generally about 10
- The difference of 10 points is a standard deviation distance.

Correlation = normalized Covariance

- correlation is a measure after you have put everything on the same slope scale
- This gives you exactly the right idea of how to compute it

W 4 Dis 10-21-16

- We have observations and we want to estimate lambda 1 and lambda 2
- Least squares error and you want to minimize this with respect to parameter A and with respect to parameter B
- If you take two inside, you can add all the a's and get na
- Put this in matrix form, invert it and get the matrix form.
- More concise form and this is a derivation.
- Differentiate each with respect to unknown.

Infinite number of datasets that give us the same straight line.

For industry:

- Most of the code is there and you have to know the confidence intervals

W 5 M Lec 10-24-16

- Imagine you are in a job interview and a guy gives you a dataset.
- What formula would you use to find the covariance

- X = iris;
- n = size(X, 1); covX = 1/(n-1) * (X-mean(X))' * (X-mean(X));
- Look at all the variables and find the extent at which they co-vary.
- Covariance is sort of a measurement of slope and this can measure a positive slope if you measure the two variables as X and Y.
- A value of 0 means that the two vectors are orthogonal but it also implies that there is no obvious slope in the data.

- XY = rand(100, 2); cov(XY)
- Gives a 2x2 matrix
- The covariance between them will be small but not totally unrelated.

Variance of the 4th variable and sigma X^2 of each of those columns is what you see in the diagonal
- The least correlated is probably the first two variables

[Q L] = eig(CovX)
Q is a n x n matrix, and L is a diagonal matrix

This takes the data and computes the Covariance either with the correlation matrix or with a covariance of the zscore

8 cylinder cars got killed off, so the miles per gallon (MPG) figures show these things are so low.
- People used to be so happy with 8 cylinders and 13 MPG.

W 5 W Lec 10-26-16
- No calculators or phones but open book, open notes
- 4 questions

- Just so you have enough problems, we added extra problems there
- The first page is a lot like the quiz with all the SVD stuff
- Least squares - covariance and correlation

PCA given by example that we want to make a little bit more formal
- Extract first eigenvectors of the covariance matrix, and those are the principle components

Which one is the first principal component?
- Eigenvector corresponding to the largest eigenvalue
- The eigenvalues are the same as the singular values, and the ordering of those things matters.
- The largest singular value or eigenvalue is well-defined and get the largest eigenvalue out of that.
- The eigenvalues are real and this is the covariance matrix, which is another kind of real symmetric matrix

Singular values are always non-negative, so eigenvalues would be non-negative.

- How could you tell if something has negative eigenvalues.
- Covariance is positive definite (nonnegative definite), so it cannot have a nonnegative eigenvalue.
- Covariance matrix has a nice interpretation of the ellipsoid.
- Covariance matrix is an ellipsoid, and we are looking at the shape of the data in terms of ellipsoids!

- First principle component is the x-axis while the second principle component is the y-axis

Petal width vs sepal length -> pairwise plot

- The red ones are the small and low end of the scale
- Find the axis at which the data spreads the most.
- If you stare at the data, this is where the data would stretch out the most.
- If you plot it along the magenta line, the spread is higher and that is what we are after.
- The sum of the squares of the dashed lines is minimum if it is the least squares line.
- In PCA, it is actually different if we are looking for the maximum spread

The eigenvectors are axes and the lengths of the axes are representing the spread

- This is a very nice relationship between what we intuitively see as the spread of the data, and what we already know about ellipsoids.
- If anyone hands you a dataset, it is not that hard to find the axis of maximal spread.
- 2D dataset and if somebody shows you this, you have all pairwise projections of the data and you can see if it is a fairly large spread.
- You can see along pairs of variables and the petal width and length were good.

Q. If Least Squares and PCA differ like this, when would you use Least Squares instead of PCA?
A. This is finding a projection of the data that corresponds to the dimension along which it spreads out.

- Use PCA for plotting the data

PCA lets us take dimensions that spread a lot and turn them into nice plotting dimensions.

- When you are trying to plot the regression line, it looks similar and there are two different ways to look at them.

Least Squares minimizes sum of square errors and PCA maximizes variance of the projection.

- This maximum spread can be expressed in terms of **e**igenvectors, and we know exactly what the maximum effect is on an eigenvector matrix.

We can use k = 2 to get two coefficients for each row and dataset entry and a_1 would be the x component and a_2 would be the y component

- We can add as many terms **k** as we would like and project it onto principal components.
- Approximate principal components on X and this would cleverly go out to the mean and the V are chosen according to the variants of the data.
- If it is a nice dataset, you only need a few terms and typically, the first principal components has the largest values.
- Are you guaranteed to get good results?
- No! You might need hundreds of critical components to get close to the data.
- Sort of a bigger picture of what we are trying to do.
- So much of data analysis was classically about variance.
- PCA is about variance and all of the basic linear modeling is applied here.

Going to be on the test
- How are z-scores related to covariance?
- What you get by subtracting the mean and standardizing the data?
- Gives you scaled data and all of it becomes scaled similarly
- Take every column in the dataset X and replace it with the normalized values.

corr(X) = cov(Z) = corr(Z)

- People are throwing everything into their analysis, so they aren't as careful as some of the other people.

What can you say about the first principal components?
- The largest entries are in the first two positions
- The first principal component is something about petal length.

Notice that the scale goes from -4 to +4 and it goes from -1.5 to 1.5
- Scale is quite a bit different and it is an order of magnitude larger.

- Many things you can do to deal with data and not throw something in.

Midterm Preview
- Why does it have to have positive singular values?
- We wind up with something that has squared values on the diagonal
- These cannot be negative.
- Least Squares Problem will be on the test

- We can resolve this quickly by knowing where to go and do it quickly on a piece of paper.
- We can see what's involved when computing this matrix.
- Requires you to be able to compute things with (A'A)^-1 formula
- Know this in your heart and know this is the pseudo inverse.
- Covariance and Correlation
- What is the covariance of x and y?
- Sine of correlation and covariance is identical

W 5 Dis 10-28-16

Covariance matrix

- Epsilon is positive-semidefinite and symmetric
- cov(**AX + a, B^(T)Y + b**) = **A cov(X, Y) B**

If you don't know the mean, you estimate it.

- For the exam, we will always be using the sample even for covariance

Sample Midterm Answers

1 .
If not invertible, not unitary
If not invertible, not positive definite

All three are unitary - true!
All of them are non-negative definite - true!

0. True - proved in the hw
0. False - Singular value decomposition of U and V don't have to be the same. If you want to flip the signs of the eigenvalues, you just flip the sign of the column vectors
0. False - look at notebook for example matrix
0. False - look at notebook
0. False - Matrix does NOT have to be square
0. True
0. False - If A is orthogonal and S is positive definite, we cannot confirm S is positive definite
0. False - Covariance is the variance
0. False
0. True - orthonormal so any column is 1.
0. True
0. True - any invertible matrix has to be positive definite
0. True

2.

0. 3rd answer choice is right
- If something is too simple, then don't look at it.

0. Eliminate 2nd to last answer and 2nd answer because there is no delta
- 1st answer choice is right

0. None of these
0. True -
0. True - formula that says this is true
0. False
0. True
0. False - you get a projection but you won't get an identity (only Identity if it is a square and X is invertible)

Pseudoinverse is multiplying on the right (so it is the left inverse)
- A * A^-1 is multiplying from the left.

3

0. True - centrally symmetric object
0. False - any least squares line you fit in will follow neg. slope
0. True - positive slope
0. False - correlation of x with itself will be positive unless 0 vector
0. False (?)

0. Third answer
0. Third answer

0. Actual #4
0. False - Sneaky! - It is False
0. False - ratio of variance and covariance is not quantified
0. Fakse
0. True
0. ?
0. True

W 6 W Lec 11-2-16
Floating Point Horror Stories
- They usually can be pretty scary and we will see it.
- They all use the same basic flaw and limitation of floating point arithmetic.
- Catastrophic cancellation.
- Floating point can go wrong with subtraction, and you can avoid these horror stories.

Horror Stories
- Convert inverse of the Hilbert matrix and you can get something really bizarre.
- It is multiplying the whole matrix to 10^-9 so this is an easy thing to miss sometimes.
- The scaling factor up there makes them all small

- These are NOT minor values.
- Just this one little computation gives bad results.
- We have to figure out why it is doing this and what to do to avoid it.
- Is there any package for Python that might run a bit slower because it will always give the exact answer.

Algorithms Matter!

- Least clever algorithm and computing the sum, and subtracts xbar from each entry
- We want to know which of these three candidates should be president?

Textbook algorithm is the worst one because it is cancelling
Incremental algorithm and Two-Pass algorithm are close but the Two-Pass algorithm is slightly better

- In this case, all the values are going to be positive and if you add up a bunch of positive values first, there will be no cancellation.

2^52 = how many decimal digits?
log_10 of this?

What is the representation of 0?

- This is cleverly designed so that the representation of the floating point 0 is a word with all 0 bits

You can do all kinds of stuff with Matlab and print out all the digits.

- Format long lets you print out all the digits!

Exploring IEEE 754

- In some system, if you get 1/0, that is division by zero, which is represented by 7ff0000000000….

Richard Stallman is a beast!

- The weakest aspect of floating-point arithmetic is that subtraction an lose all your significant digits.
- Do NOT do subtraction on floating point calculations!

Fast Fourier Transform

- Hadamard Matrix has the bottom right matrix negated and the other three matrices are identical.
- Symmetric matrix
- Result will be symmetric and use proof by induction that this is symmetric
- Orthogonal, so it is its own inverse

W 6 Dis 11-4-16

- The idea is to take the two arrays and stack every column to get a vector.

- Correlation matrix compares most of the information
- The mean is that you are given a lot of images and you want to average them.
- Get some nice information from it here.
- Subtract the mean from every face and you want to center them.
- You get the covariance matrix C

- Looks like a ghost!
- The u_j are the eigenfaces and they come from the dominant eigenvectors.
- You can represent any training face in this basis.
- Phi_i is the eigenface, and we want to look t the best K.
- If you take the mean and add the particular component to it.

Face Recognition using Eigenvalues

- Center it and project it on to the eigenface.
- The idea is that you have w_1 to w_K and you want to see how close is this to a value.

Distance within the face space (difs)

- For us, Euclidean distance is good enough

How do you get e_d, train a lot of false database and get a threshold.

- Simple eigenvalue decomposition to get a face.
- All the images you get have to have the same kind of shading and if you put them in different environments and backgrounds, it will not recognize it.

Projection onto a plane works really well too.

- How does 3D scanning work?
- If you know the distance from your face to the camera, it will probably be easy.
- How to deal with 3D data?

The Fourier Transform

- What are the main properties?
- Unitary?
- Nonsingular
- Symmetric
- Not Hermitian: All the diagonal values can be 0 or negative
- How would you define fourier transform

sin(x) means you can interpret any data you want.

- Given this background, we know that is the way from continuous to discrete.

- The maximum bandwidth is omega, and all the information from D to 1/ omega will be there.
- Get all the information up to a particular time interval.

- Time domain and frequency domain argument.
- Correspondence between analog and digital domain.
- Put integer samples and get a time vector and frequency vector

Look at small frequencies like hertz

- If we have frequency domain of information.
- If I give you one sine wave and if we sample two points, we should get only one of them to be 1 and the other is 0 if we sample them properly.

Sum of sinusoids can be calculated here because it is linear

Succulent matrices

- There is a way to do it but it can be done.

W 7 M Lec 11-7-16
The Hadamard Matrix

- Different matrix and detecting different patterns
- Each of the rows if you plot it as a curve look sinusoidal and the Fourier matrix rows are sinusoidal

The Fourier Matrix

- Easy to memorize and there is no effort required to do this
- Each row of the Fourier Matrix corresponds to cosines and sines in the end.
- The real part of the matrix is the cosines
- The imaginary part of the matrix is sines
- Fourier matrix tightly connected to sines and cosines

The Fourier Matrix for n = 2

- Everything in the first row results in an exponent of 0.

The Fourier Matrix for n = 4

- Some entries are complex and some entries are real, so you have to be careful when dealing with these matrices.
- Generally don't have to deal with complex numbers in Computer Science, but mathematical modeling is an exception

Fourier Matrix properties

- If you take F_n^4, you get I_n

Computing the Fourier Matrix

- People omitted the 1/sqrt(n) and the Fast Fourier Transform was invented in 1965

W 7 W Lec 11-9-16

- The frequency that is .091 has a period which is the inverse of this (1/f).
- 1/.091 = 11.04

If you Fourier Transform something twice, you get the original value

- The center thing is what we want to get back to soon, and this is known as the convolution theorem.
- If you are convolving two functions, it can actually be computed very efficiently by running a Fourier Transform and taking the products of these.

The conversation function is O(N^2) while the real inverse Fast Fourier Transform is O(N log N)

JPEG - DCT Image Compression Demo

- 8x8 blocks so therefore, if we take the SVD stuff, it will wind up being 64 dimensions and each one is a value

Modular arithmetic was invented in the late 1940s called the mid-square method

W 8 M Lec 11-14-16

- integers are 32 bits, so 4 bytes
- 2^32
- Matlab integers: 2^32, which requires 32 bits to represent, and we only have 31 if we have signed integers.

sum is just going to be Boolean, and we will add them to get zeroes and ones to get the total number of points inside.

- Figure out what the value of pi ought to be

UCLA students are smarter than normal people, so it is a random sampling of people.

- If we have a curve like this, we need the area underneath it to toal to 1
- If we do not have the constant, what does this integral give us?

Get it closer and closer to a normal distribution!

- One of the most basic statistical results that depend on this property that is worth a whole chapter in the course notes.

If you sum up a bunch of random values, it should look like those results are these.

- Stocks in a portfolio or a basket of stocks.

Sigma / sqrt(n)

- n = 256 and the sqrt(n) = 16

Convolution

- People actually use this to compute normal random values

Run the Matlab experiments when we get home

W 8 W Lec 11-16-16

- Dr. Strange has degree 4
- Spiderman may have degree 150

Logistics
HW 2 - due Sunday, Nov. 20
HW 3 - due Sunday, Nov 27
HW 4 - due Sunday, Dec 4

Discussion sections

- none on Friday Nov 25th (No class!)

W 8 Dis 11-18-16

- What if the random process for all the given times is the random processes?
- First moment and second moment do not vary in time.
- When you look through a fan, the fan is doing sampling for you!

Fourier Transform - shift by A and get a term here

- Shift Theorem: Take a signal, shift it, what happens it?
- Similarity Theorem:
- Modulation Theorem: Comes from actual FM modulation
- This property is used in AM/FM radio
- Amplitude modulation and you multiply the signal by the first frequency, and the entire frequency spectrum is modulated.

Convolution: Take two polynomials and multiply them.

Wiener-Khinchin Theorem

- Take the Fourier transform as a duality here.
- Any random variables should give an independent distribution

Cauchy-Distribution

- Given a bunch of random variables - will it convert to normal distribution?
- What if I give you a bunch of Cauchy-random variables?
- Will the Central Limit Theorem apply?
- It wouldn't apply there.
- No!
- Weird distribution with no means and no variance.
- Central Limit Theorem have to have finite points.

- Does NOT work on semi-stable distributions.
- If you have no definite mean or no definite variance, it won't work!

It is good to know Pareto Distribution
- 80-20 rule
- Happens a lot in sales!

W 9 M Lec 11-21-16
- Learn about optimizers!

W 9 W Lec 11-23-16
- Halfway between square root iteration
- Take equations of motion and use symbolic calculation to turn them into code.
- Do things like this and generate source code
- We could have probably done this in a few lines of code using symbolic tools

Think about what Taylor Series looks like and generalize it
- If you have vectors instead of single values x, you suddenly wind up with vector derivatives instead of single derivatives
- F is always just yielding a single real value and in this case, it is NOT.
- If you take the gradient of this, since x is a vector, you wind up with as many derivatives as entries in the vector.
- You can wind up computing the derivative of f, which gives you this nice thing.

Hessian: Taking the Jacobian and then taking the second derivative of it
- Differentiate each entry in the vector for every possible n variable.

3D array and you wind up getting an n x n x n matrix, which is called a tensor!
- You can easily get higher dimensional derivatives if you want.
- If it is a 3D array, you wind up multiplying x on all 3 dimensions.
- Doesn't fit well in matrix algebra anymore and having to use tensor algebra

Banana Function
- Simple little function: 100(x_2 - x_1)^2 + (1 - x_1)^2
- This beautiful plot is a contour plot of that function.
- You could see something about the shape of this function
- Valley in the middle that is shaped like a banana
- Both terms are squared, so it cannot be negative

Rosenbrock's
- Two derivatives that you are computing.
- It is messy but that is what you get

Newton's Method in Higher Dimensions

- Start off in upper left hand corner and this would get you back to the minimum value of the root.
- Amazingly fast.

Newton's Method on the Banana Function

- Shoots back and forth, but this is one of the challenges of vector optimizations.
- You get the idea and you can use the same sort of methods we were doing before.

Jacobians and Hessians in Octave

- Pretty powerful stuff
- Produces elegant looking symbolic output for you.
- Uses the Mac Terminal to show the quality of the symbolic layout

W 10 M Lec 11-28-16

- Hw 4
- Do some ChiSquare stuff and look at distributions of earthquakes by month or minute.

W 10 W Lec 11-30-16

- Fourier Transform works almost exactly the same way and x_0 turns out to the be the even vectors and x_1 turns out to be the odd vectors.
- Almost exactly like the FFT but trivialized because it is a simple matrix.

Suppose we want to simplify the expression at the top of the screen, what would that simplify to if we apply those little identities?