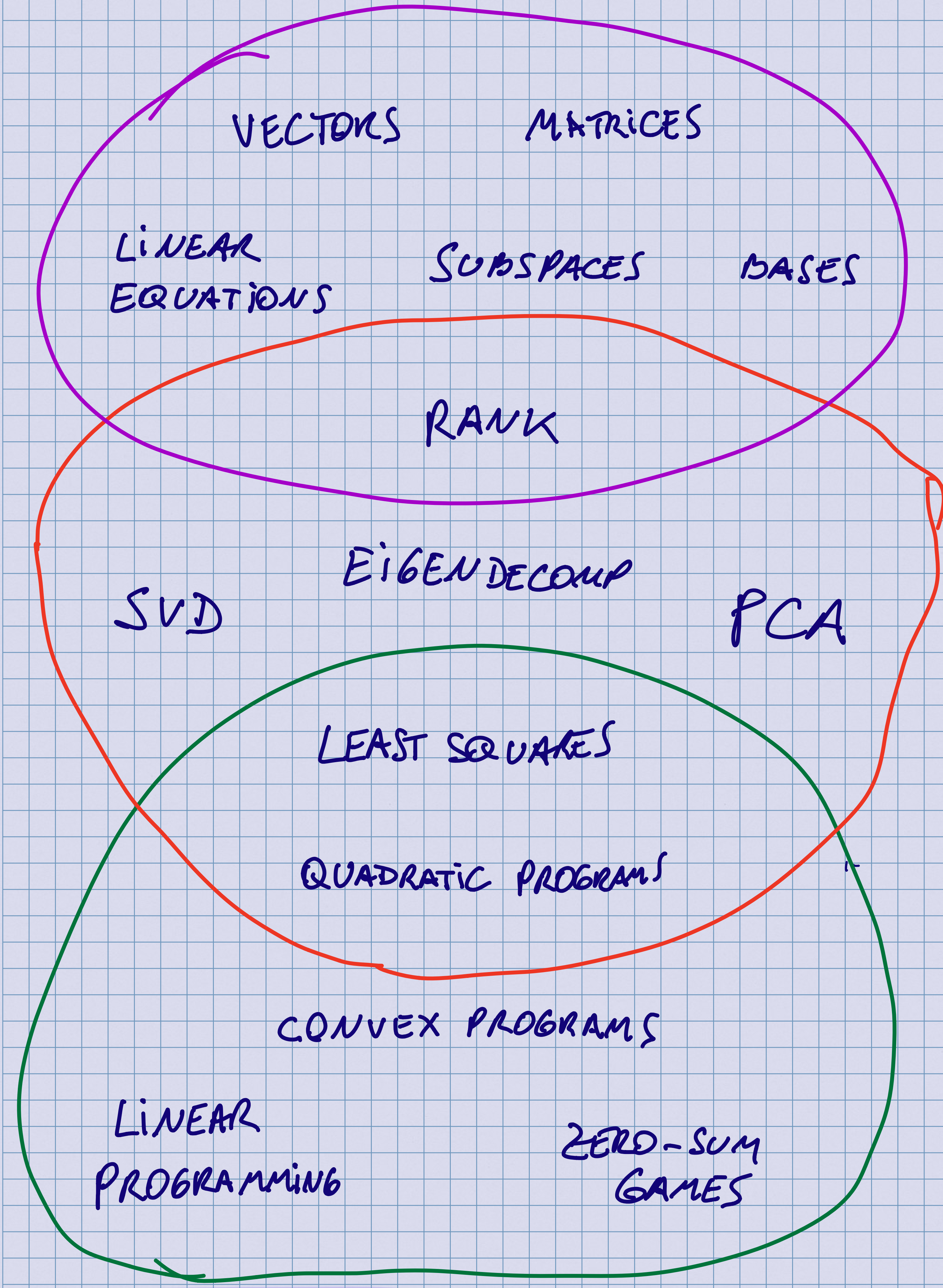
Lecture 36

FINAL REVIEW (II)

- FINAL EXAM ON FRI 12/18
- PRACTICE EXAM WILL BE POSTED SOON


VECTORS
MATRICES
LINEAR EQUATIONS
SUBSPACES
BASES
RANK
EIGENDECOMP
SVD
PCA
LEAST SQUARES
QUADRATIC PROGRAMS
CONVEX PROGRAMS
LINEAR PROGRAMMING
ZERO-SUM GAMES

LINEAR SYSTEMS TAKE MANY FORMS,

not just $$Ax = b$$

- LINEAR DIFFERENTIAL EQUATIONS

$$\frac{dx(t)}{dt} = \underset{n \times n}{A}\ \underset{\in \mathbb{R}^n}{x(t)} + \underset{n \times m}{B}\ \underset{\in \mathbb{R}^m}{u(t)}$$

- LINEAR PARTIAL DIFF EQS. (PDE)

e.g., heat equation $$\boxed{\Delta f = \frac{\partial f}{\partial t}}$$

where $f = f(x_1 \dots x_n, t)$

$$\Delta f = \frac{\partial^2 f}{\partial x_1^2} + \dots + \frac{\partial^2 f}{\partial x_n^2}$$ (LAPLACIAN)

COMPUTATIONAL / ALGORITHMIC TASKS

- SOLVE LINEAR SYSTEMS

$$Ax = b$$

- COMPUTE EIGENVALUES / SINGULAR VALUES

$$A = TDT^{-1}$$

or

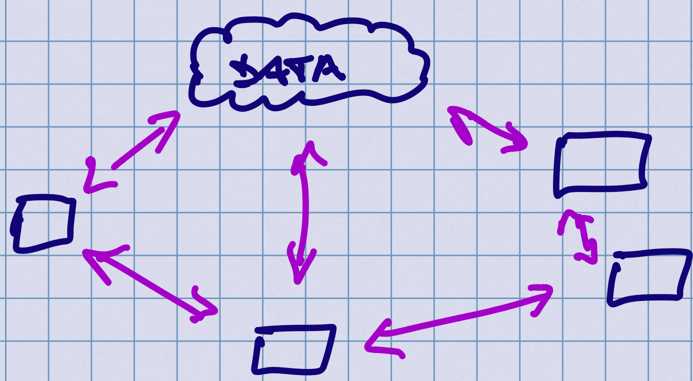
$$A = U \Sigma V^T$$

- LOW-RANK APPROXIMATIONS / PCA

$$A \approx \sum_{i=1}^{k} u_i v_i^T$$

DIFFERENT FLAVORS / APPROACHES

- DISTRIBUTED / PARALLEL COMPUTATION



- RANDOMIZED ALGORITHMS

"Randomized Linear Algebra".

E.g., random projection

$$A \rightsquigarrow A' = PAQ \rightsquigarrow \mathrm{SVD}(A')$$

(big matrix)

P, Q "random"

small matrix

svd of small matrix gives approximate SVD of A.

OPTIMIZATION

- ML / SGD / DEEP LEARNING

$$\min_{\theta} \sum_i \ell(\theta, x_i)$$

parameters

data points.

- better versions of gradient descent / SGD?
- no "hyperparameter tuning"?
- better <u>generalization</u> properties?

- ROBUSTNESS

Say I want to solve

$$Ax = b$$

but A, b are only approximately known.

- What to do?
- What are the implications (e.g., in x).

Same issues in optimization:

$$\min_x \; f(x)$$

"Uncertain" function

- probabilistic
- worst-case (adversarial)
- semi-random

MIN-MAX AND GAMES

We've talked about

$$\boxed{\max_x \; f(x)}$$

(OPTIMIZATION)

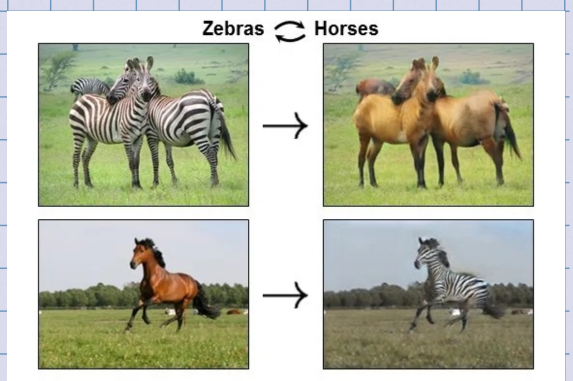
Sometimes, we're interested in

$$\boxed{\min_x \; \max_y \; f(x,y)}$$

- How to solve this? (gradient descent?)

- Applications:

  GANs

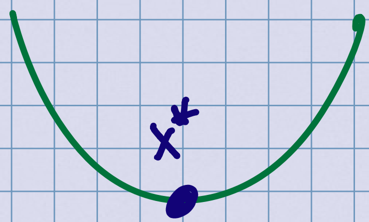


(generative adversarial networks)

- NON-CONVEX OPTIMIZATION ?

how to produce guarantees for non convex problems?

Recall:



convex case:

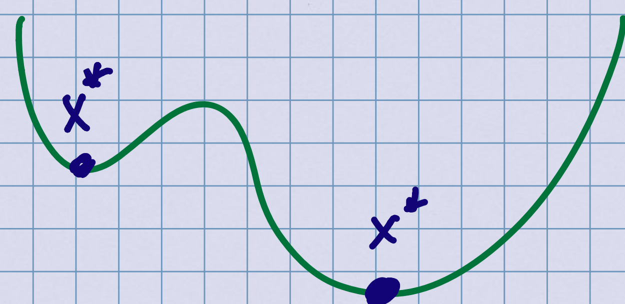
$\nabla f(x^*) = 0$

$\Updownarrow$

x^* global optimum

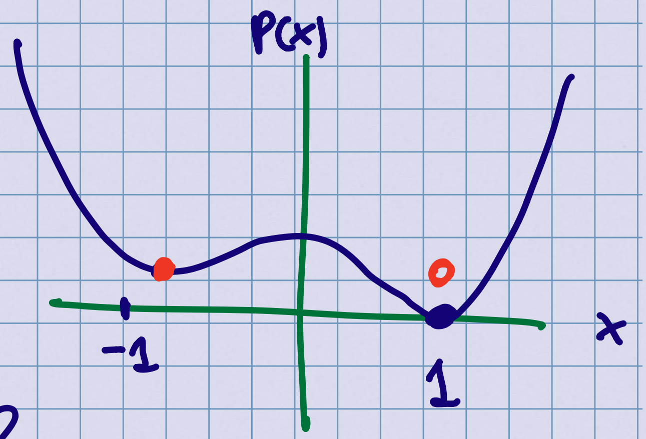
$$f(y) \geq f(x) + \nabla f(x) \cdot (y - x)$$

But, what to do here?



In some cases, we can do interesting things, by exploiting algebraic properties of $f(x)$ (e.g., if f is a polynomial).

Example :



$$P(x) = 4(x^2-1)^2 + (x-1)^2$$

Two local minima : $x = 1$, $x \approx -0.853$

Can "easily" show that $x=1$ is the global minimum

How?

- $P(1) =$

- $P(x)$

What to take next? (partial selection!)

18.065 MATRIX METHODS IN DATA ANALYSIS, SIGNAL PROCESSING AND ML

18.085 COMPUTATIONAL SCIENCE AND ENGINEERING I

18.335 INTRODUCTION TO NUMERICAL METHODS

18.337 PARALLEL COMPUTING AND SCIENTIFIC MACHINE LEARNING

OPTIMIZATION @ MIT

6.251	LINEAR OPTIMIZATION
6.252	NONLINEAR PROGRAMMING
6.253	CONVEX ANALYSIS AND OPT.
6.255	OPTIMIZATION METHODS
6.881	OPTIMIZATION FOR MACHINE LEARNING
15.053	OPTIMIZATION METHODS IN BUSINESS ANALYTICS
15.094	SYSTEMS OPTIMIZATION: MODELS AND COMPUTATION