INTRODUCTION TO MACHINE LEARNING COMPSCI 4ML3 LECTURE 6

HASSAN ASHTIANI



 $\vec{p} : \mathbb{R}^{d_1} - \mathbb{R}^{d_2}$

D: dzzoz

(2, operations

- TRAINING: CALCULATE W^{RLS} = $(\overline{\phi}^T \phi + \lambda I)^{-1} \overline{\phi}^T Y$
- BOTTLENECK: MATRIX INVERSION
 - How MANY OPERATIONS?

• PREDICTION: $\hat{y} = \langle \phi(x), w^{RLS} \rangle$ $(x); w^{RLS} \rightarrow$

• How many operations?

 REGULARIZATION ALLOWS US TO GO INTO HIGH-DIMENSIONAL SPACE WITHOUT OVERFITTING, BUT IT DOES NOT SOLVE THE COMPUTATIONAL PROBLEM

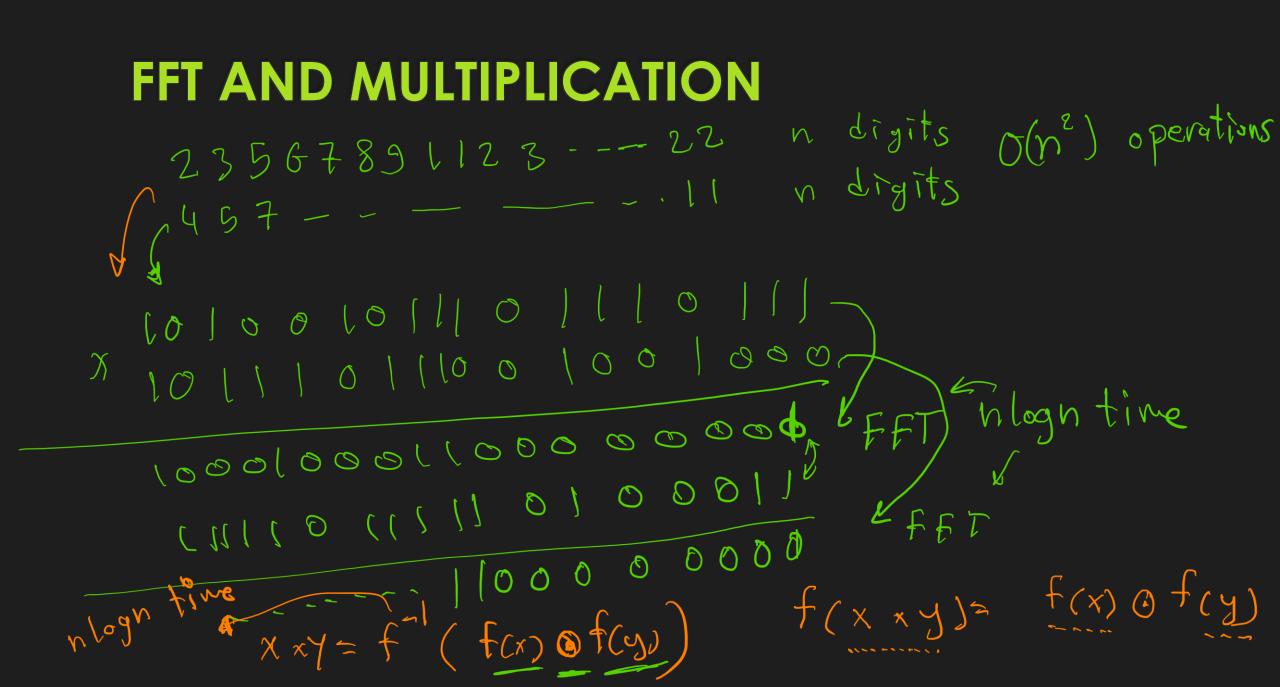
COMPUTATIONAL COMPLEXITY



- MATRIX MULTIPLICATION (N-BY-N MATRICES)
 - NATIVE METHOD: $O(N^3)^{\swarrow}$
 - STRASSEN'S ALGORITHM: $O(N^{2.8074})$
 - COPPERSMITH-WINOGRAD-LIKE ALGORITHMS [CURRENT BEST $O(N^{2.3728639})$]
- MATRIX INVERSION
 - Gaussian Elimination: $O(N^3)$
- POSSIBLE TO REDUCE IT TO MULTIPLICATION

THE COMPUTATIONAL PROBLEM

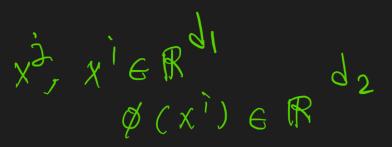
- CAN WE SOLVE THE REGULARIZED LEAST SQUARES IN \mathbb{R}^{d_2} without explicitly mapping the data into \mathbb{R}^{d_2} ?
 - $W^* = \min_{W \in \mathbb{R}^{d_2}} \|\Phi W Y\|_2^2 + \lambda \|W\|_2^2$
- Something like multiplication using FFT
- IF SO, WE COULD EVEN MAP THE DATA TO AN INFINITE DIMENSIONAL SPACE!!



THE COMPUTATIONAL PROBLEM

- CAN WE SOLVE THE REGULARIZED LEAST SQUARES IN R^{d_2} without explicitly mapping the data into R^{d_2} ? • $\min_{W} ||\Phi W - Y||_2^2 + \lambda ||W||_2^2$
- Something like multiplication using FFT
- IF SO, WE COULD EVEN MAP THE DATA TO AN INFINITE DIMENSIONAL SPACE!!

THE KERNEL TRICK



 COMPUTE THE HIGH-DIMENSIONAL INNER PRODUCT EFFICIENTLY
d₂

$$K(x^i, x^j) = \langle \phi(x^i), \phi(x^j) \rangle =$$

- Use this as a building-block for performing other operations
- REWRITE THE LEAST SQUARES SOLUTION SO THAT IT ONLY USES THE INNER PRODUCT OF THE FEATURE MAPS?!

 $(\Phi^T \Phi + \lambda I)^{-1} \Phi^T Y$

THE KERNEL FUNCTION

- Kernel function for a mapping ϕ :
- EXAMPLE:

$$\phi(x)^{T} = \begin{bmatrix} 1, \sqrt{2}x_{1}, \sqrt{2}x_{2}, \dots, \sqrt{2}x_{d}, & 0 \\ (x_{1})^{2}, x_{1}x_{2}, \dots, \sqrt{2}x_{d}, & 2 \end{bmatrix}$$

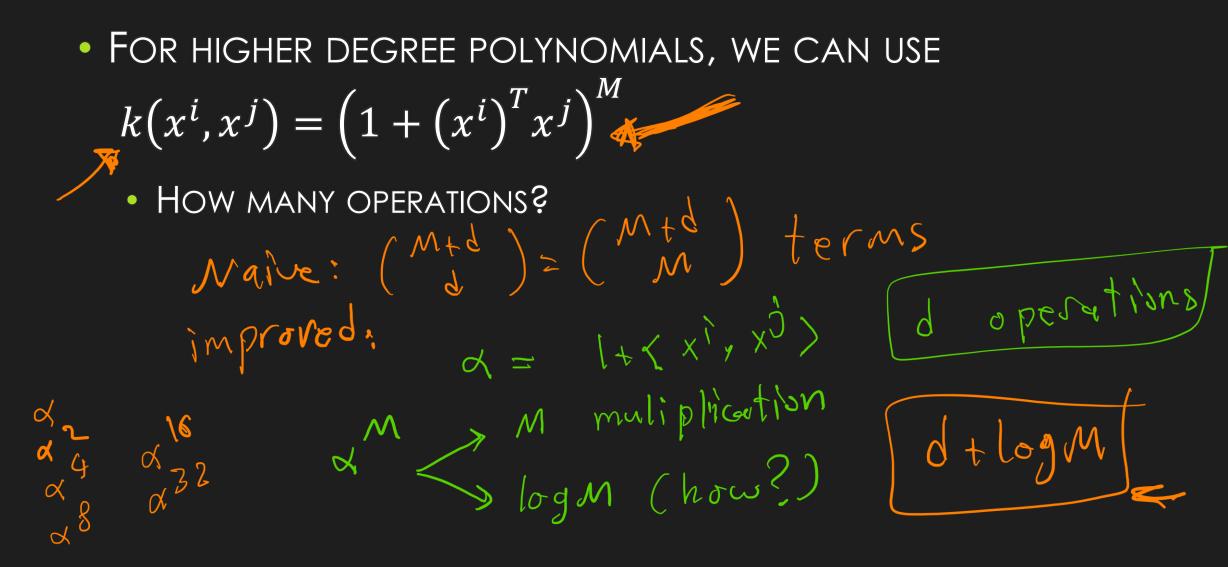
- So polynomial basis functions with M = 2
- COMPUTING $K(u, v) = \langle \phi(u), \phi(v) \rangle$ $d_2 = (d_1)^2$ operations
 - COMPLEXITY OF NAÏVE CALCULATION? •
 - BETTER APPROACH?

 $\begin{aligned} d = d_{1} \\ (u_{3}V) = \langle \phi(u), \phi(v) \rangle = \langle (1, \sqrt{2}u_{3}, \dots, \sqrt{2}u_{d}), \frac{u_{1}^{2}, u_{1}u_{3} \dots, u_{1}^{2}}{(1, \sqrt{2}u_{1}, \dots, \sqrt{2}v_{d})} \\ & (1, \sqrt{2}v_{1}), \sqrt{2}v_{1}, \frac{u_{1}^{2}, u_{1}u_{3} \dots, u_{1}^{2}}{(1, \sqrt{2}v_{1}, \dots, \sqrt{2}v_{d})} \\ & (1, \sqrt{2}v_{1}), \sqrt{2}v_{1}, \frac{u_{1}^{2}, u_{1}u_{3} \dots, u_{1}^{2}}{(1, \sqrt{2}v_{1}, \dots, \sqrt{2}v_{d})} \end{aligned}$ $= 1 + 2 \sum_{i=1}^{d} u_i v_i + \sum_{i=1}^{d} \sum_{j=1}^{d} u_i u_j^2 v_i v_j^2 \qquad (o(d^2) \text{ terms})$ $= \left(\left| + \sum u_i v_i \right|^2 = \left(\left| + \langle u_i v \rangle \right|^2 \right)^2$ O(d) operations is enoug

•
$$k(x^{i}, x^{j}) = (1 + (x^{i})^{T} x^{j})^{2} = (1 + \langle x^{i}, x^{j} \rangle)^{2}$$

• NUMBER OF OPERATIONS?

DEGREE M POLYNOMIALS



THE KERNEL TRICK

- Compute the high-dimensional inner product EFFICIENTLY
 - $K(x^i, x^j) = \langle \phi(x^i), \phi(x^j) \rangle$
- Use this as a building-block for performing other operations
- REWRITE THE LEAST SQUARES SOLUTION SO THAT IT ONLY USES THE INNER PRODUCT OF THE FEATURE MAPS?!
- $(\Phi^T \Phi + \lambda I)^{-1} \Phi^T Y$

ROADMAP

- ASSUME d_2 is very large, even $d_2 >> n$
- Instead of finding W, try to introduce new parameter a whose size is n rather than d_2
- Now we have n parameters
- FIND OPTIMAL a as a function of K