Neural Networks: A Theory Lab

Topic: Universal features

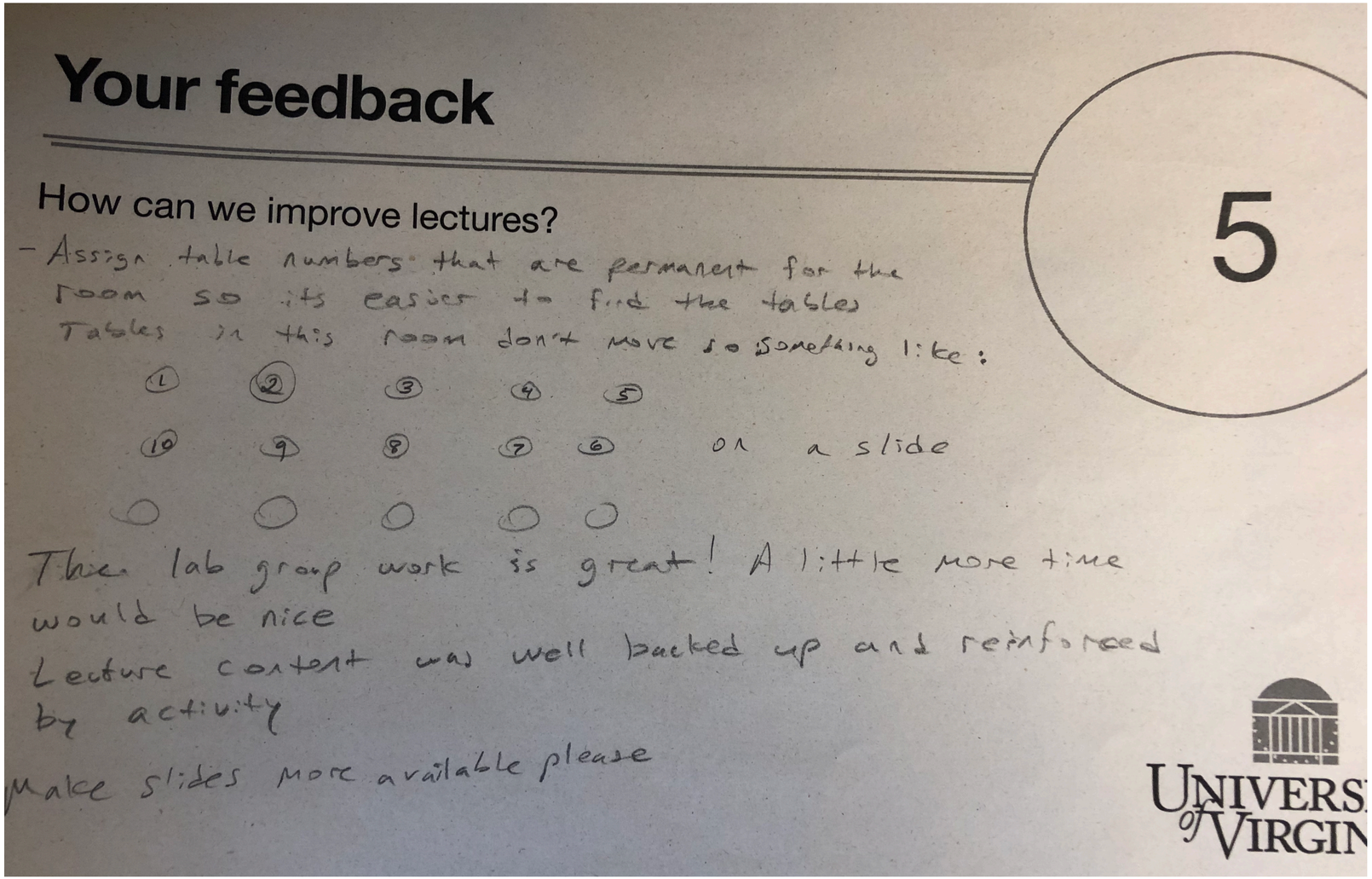
Outline:

- Recap
- Lab
- Theory

Hadi Daneshmand

University of Virginia

Thank you so much for your feedback



Your feedback

5

How can we improve lectures?

- Assign table numbers that are permanent for the room so its easier to find the tables
Tables in this room don't move so something like:

① ② ③ ④ ⑤

⑩ ⑨ ⑧ ⑦ ⑥ or a slide

○ ○ ○ ○ ○

This lab group work is great! A little more time would be nice

Lecture content was well backed up and reinforced by activity

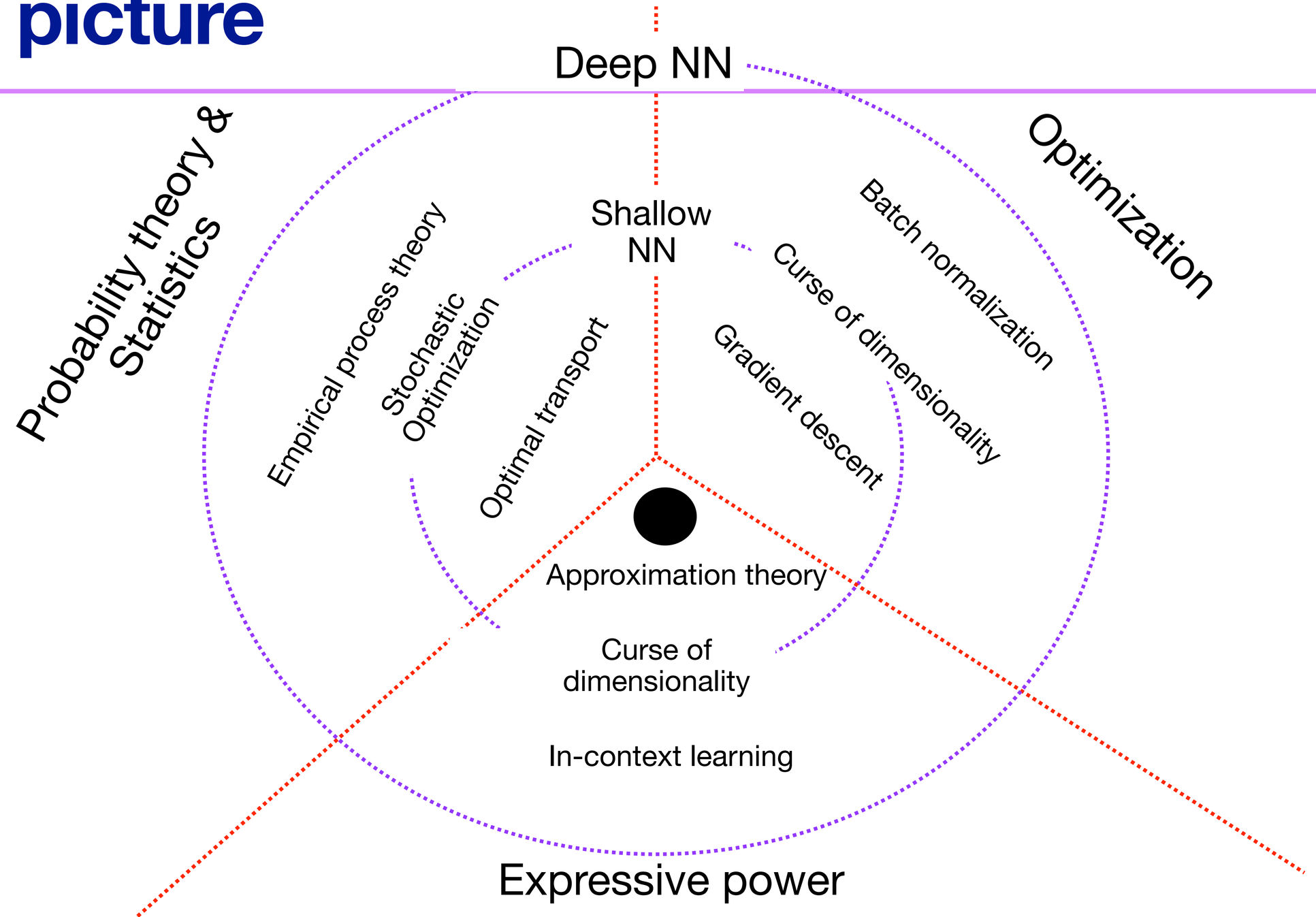
Make slides more available please

UNIVERSITY of VIRGINIA

Before starting the lecture

- Recorded video is available on Canvas
  - Unfortunately, the quality of recorded video is not good
- I posted the lecture note on Canvas and course website: https://hackmd.io/@hadidanesh/SJJYw3Lvke
  - You can post your comments on lecture notes online.

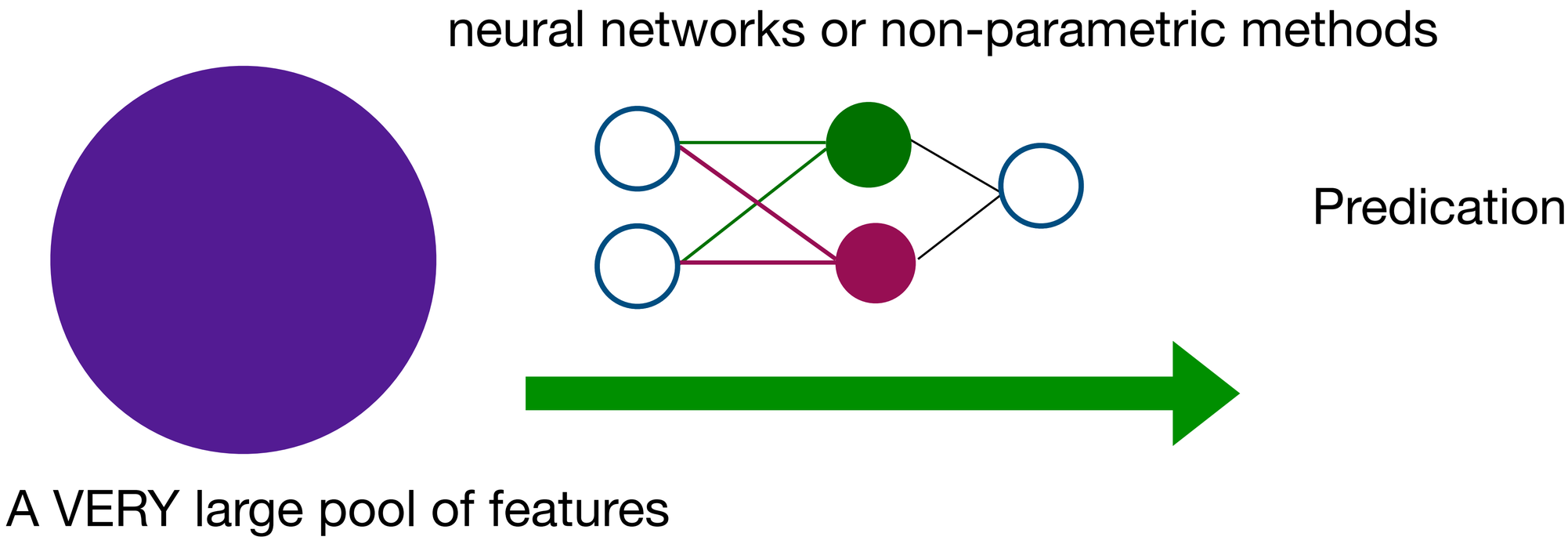
Big picture


Deep NN
Probability theory & Statistics
Optimization
Shallow NN
Empirical process theory
Stochastic Optimization
Optimal transport
Gradient descent
Curse of dimensionality
Batch normalization
Approximation theory
Curse of dimensionality
In-context learning
Expressive power

Recap

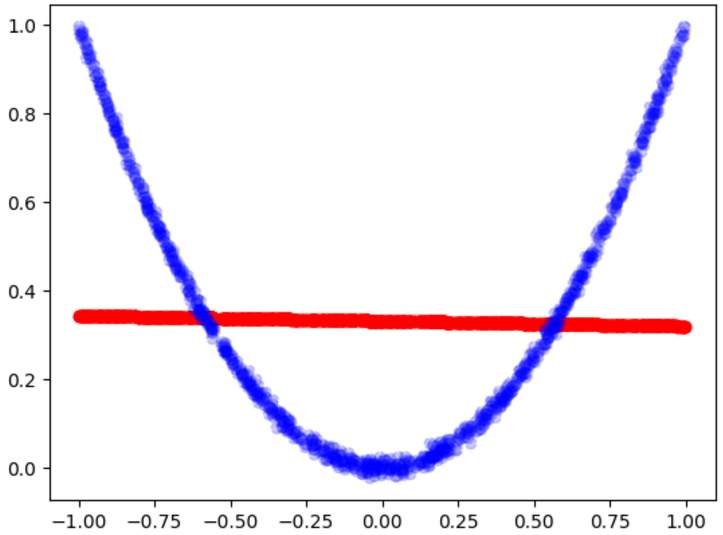
Deep Learning Goal

- Deep learning automates feature extraction (representation learning)



Question: How to solve the non-linearity challenge

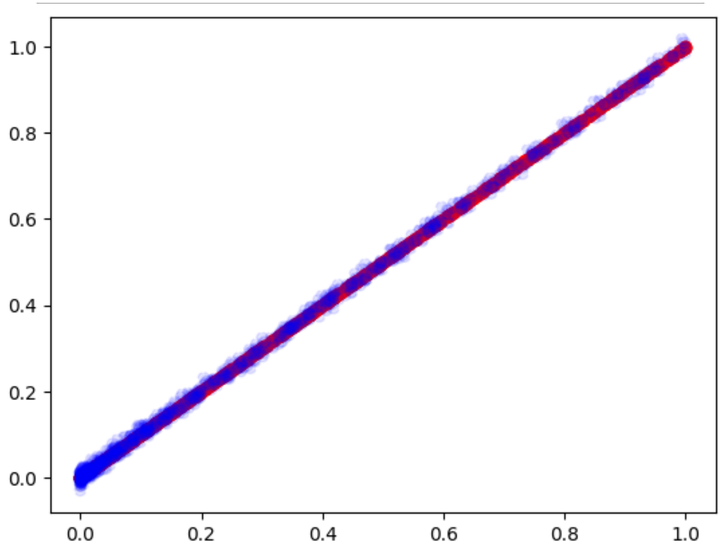
y



x

$\phi(x) = x^2$

y

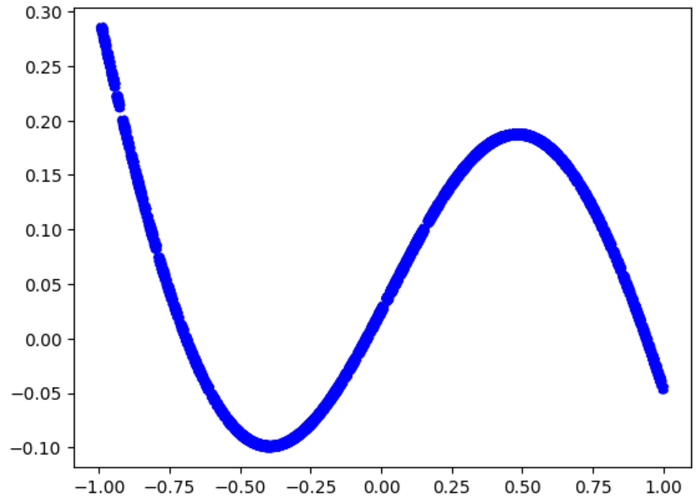


x^2

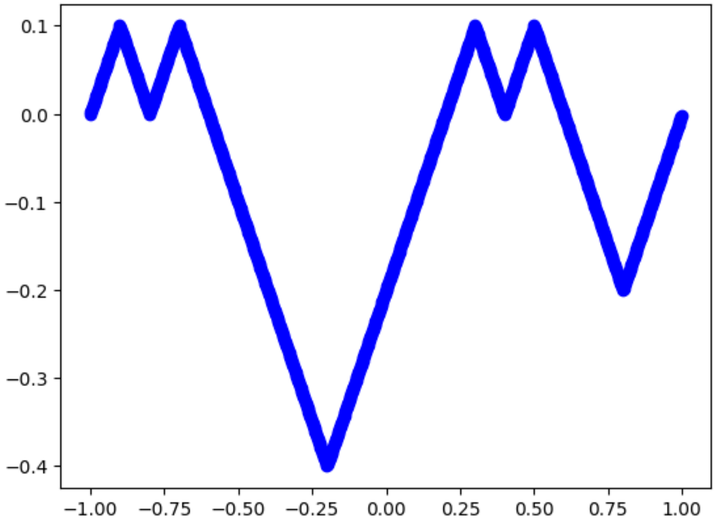
Group activity

- Goal: We want to design **universal** non-linear features
- **Universal** features can approximate various y s

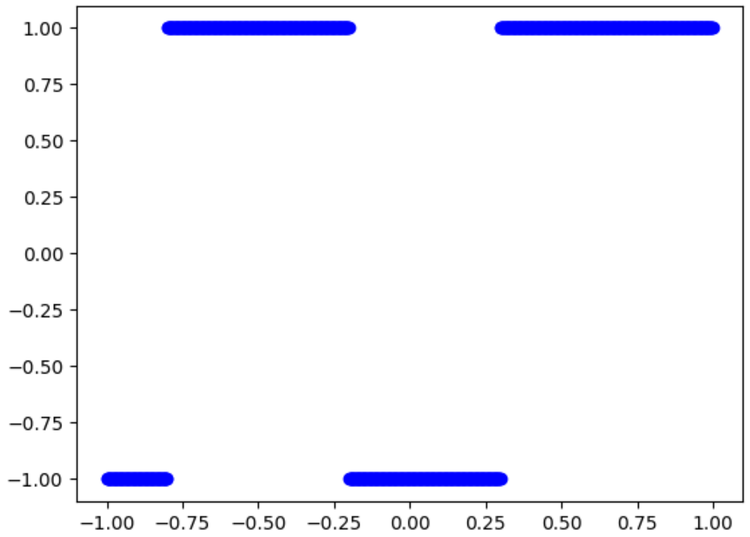
y



x



x



x

Kernel regression and representer theorem

- Kernel regression: $f^* = \arg\min_f \sum_i (f(x_i) - \underbrace{y_i}_{p(x_i)})^2 + \lambda \|f\|^2$

  subject to f obeys the reproducing property $f(x) = \int_{-\infty}^{\infty} k(y, x) f(y) dy$

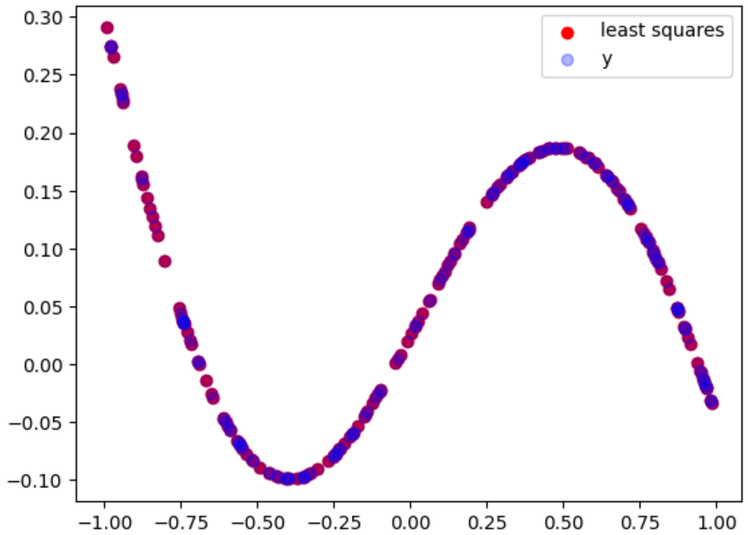
- **Representer theorem [Schölkopf et al. (2001)]:** $f^* = \sum_i \alpha_i k(x, x_i)$ for $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$

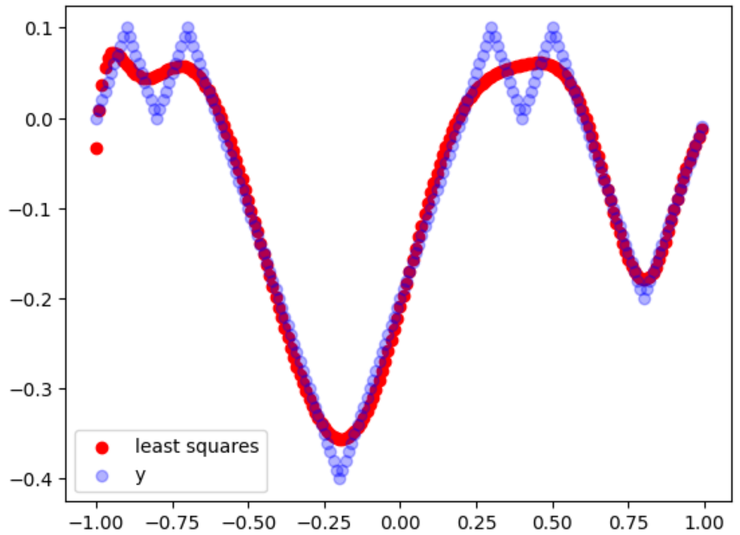
- As long as: k is positive semi-definite, namely $\sum_{i,j} \alpha_i \alpha_j k(x_i, x_j) \geq 0$ holds for all $x_i, x_j \in \mathbb{R}^d$, $\alpha_i \in \mathbb{R}$

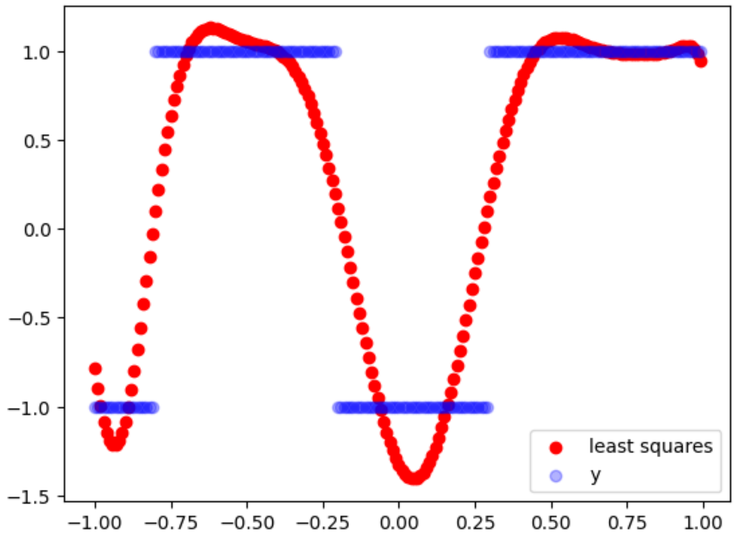
Lessons from representer theorem

- A linear combination of the same (universal) features can construct many many functions at the same time.

- Blue: y and red is the solution of regression on random features







Thank you very much for your help

- I thank Jiuqi Wang and Evan Conway for pointing out an issue with the code
- The bug is fixed and solution is included in colab: https://colab.research.google.com/drive/1M-nRBdhg1XJiV8sy4wMkUsfPJLKKSCB0#scrollTo=diCPqiTj5wB3

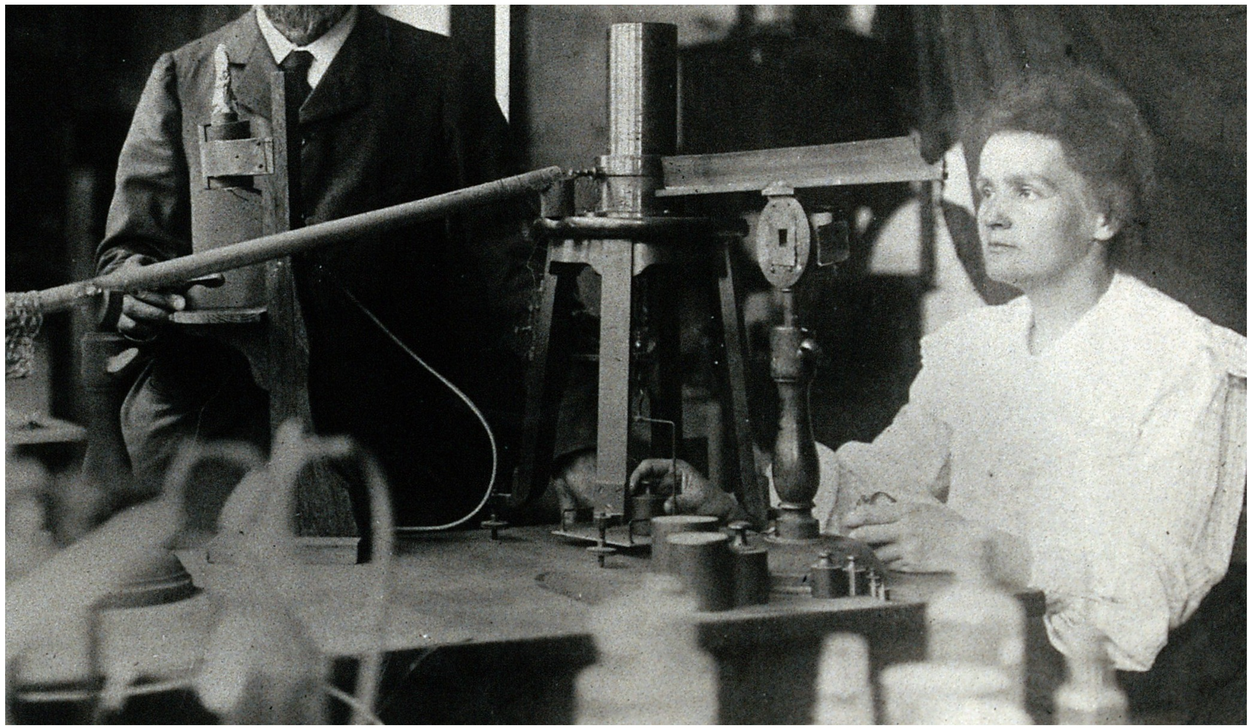
A computational challenge with kernels

- For dataset of size n, we need n^2 features
- Hence, we **can not** even evaluate the performance of kernel methods on standard ML benchmarks such as ImageNet.

Today topics

- Reviewing an effective method to reduce the cost for kernel method
- A bridge to neural networks

- Recap
- Lab
- Theory



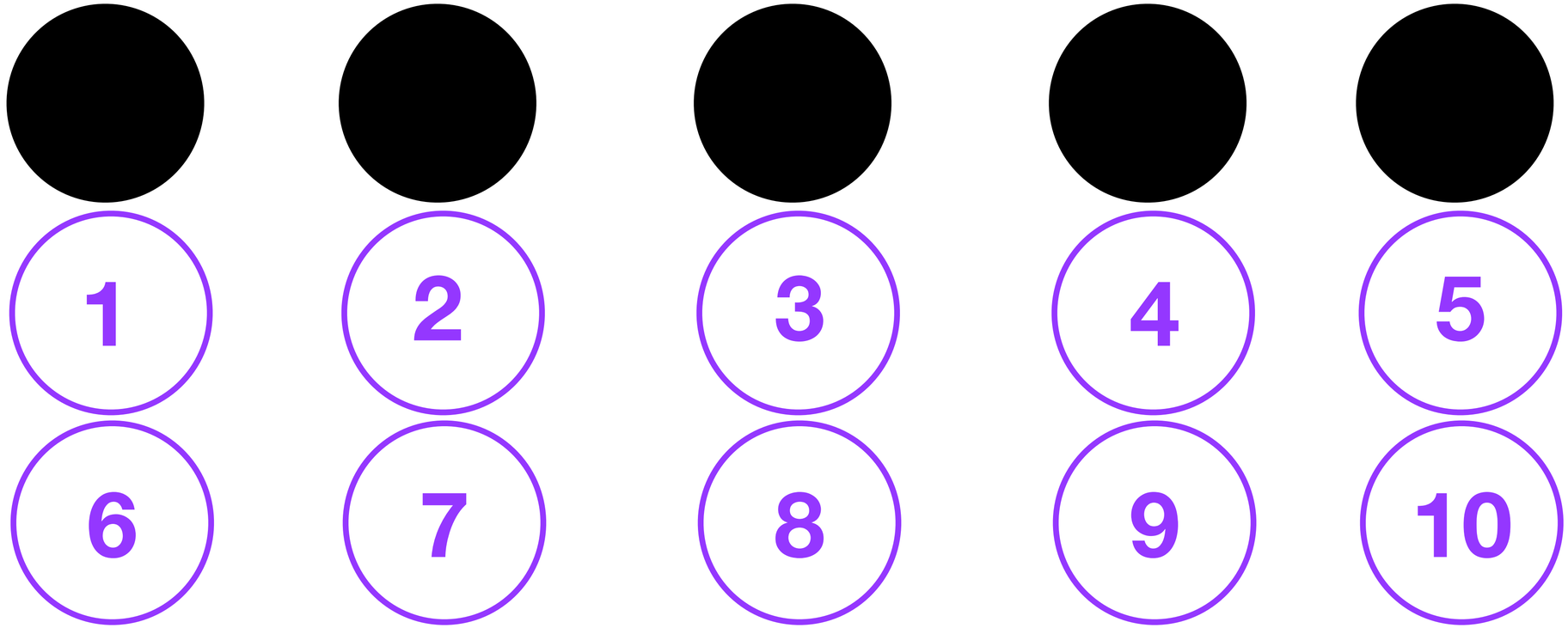
wikipedia: Marie Curie

Lab

Get ready for hands-on group activity

Group assignment

- Feel free to choose your own group if you want
- Thank you for your thoughtful approach to numbering the tables

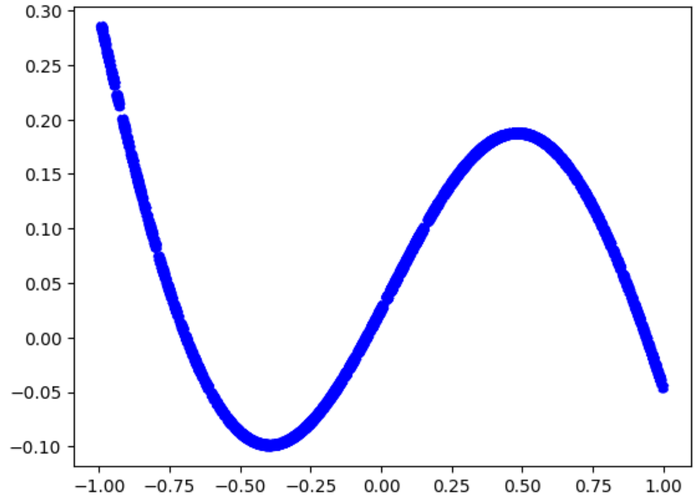


- I am here

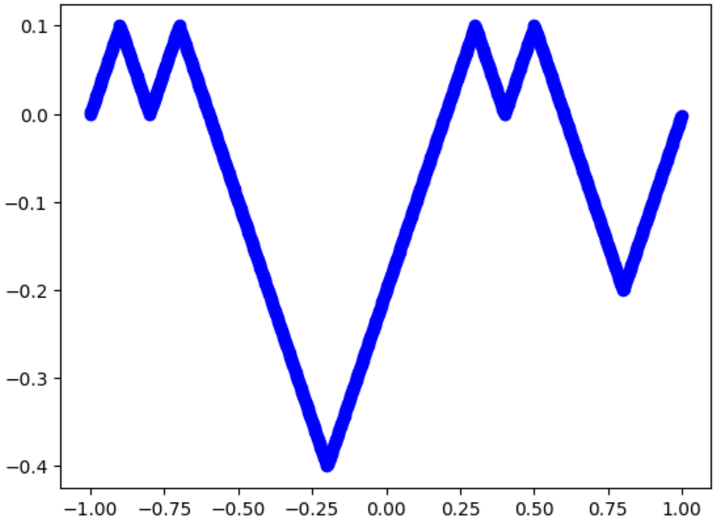
Group activity

- Goal: We want to design **universal** non-linear features
- **Universal** features can approximate various y s

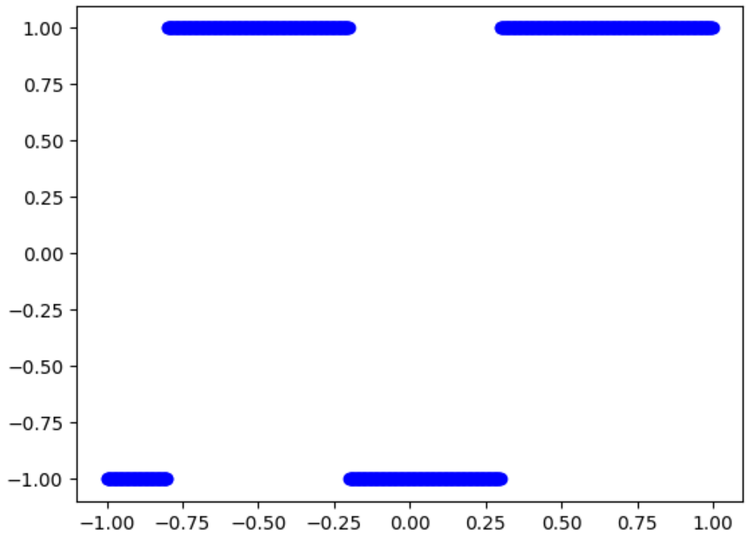
y



x



x



x

Your task (15 mins)

- Given x, design features $\phi_1(x), \ldots, \phi_{30}(x)$ such that $y \approx \sum_{i=1}^{30} w_i \phi_i(x)$

- $\phi_i : \mathbb{R} \to \mathbb{R}, x \in \mathbb{R}$ n Samplesize

Scan to Start

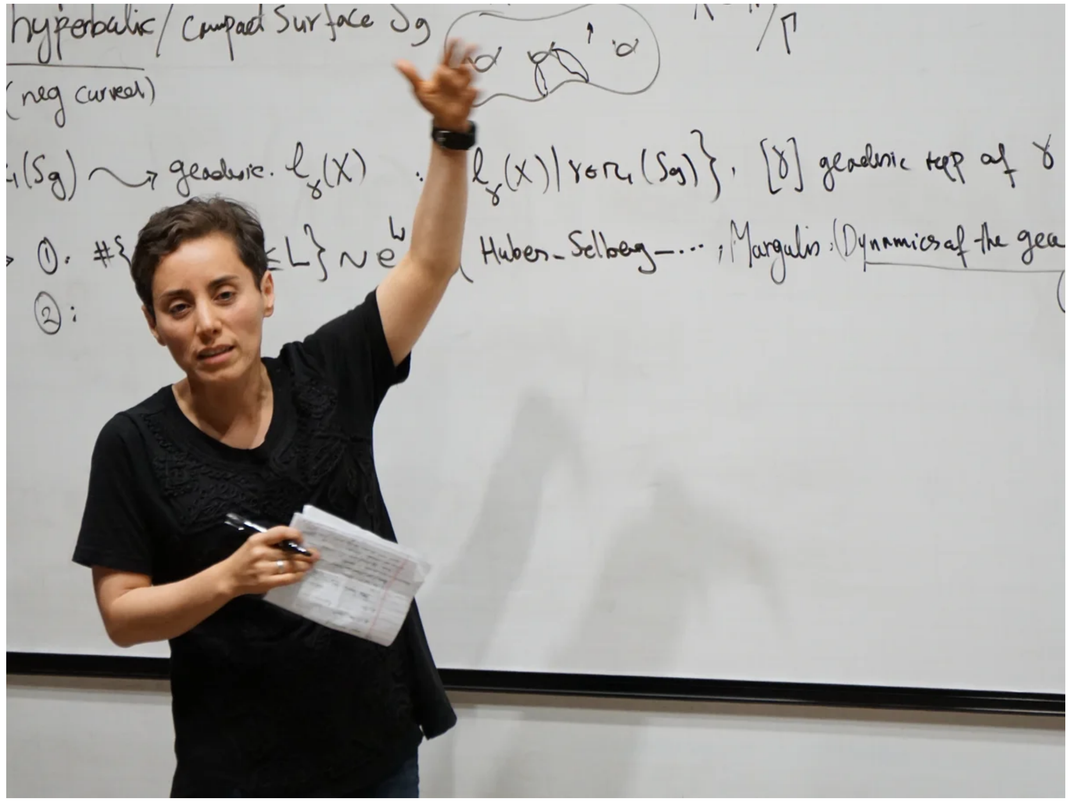
https://shorturl.at/ABb6H

https://colab.research.google.com/drive/1M-nRBdhg1XJiV8sy4wMkUsfPJLKKSCB0?usp=sharing

- You can search or use ChatGPT

Let’s share our ideas

- Intro
- Lab
- Theory



theguardian: Maryam Mirzakhani

Theory

Get ready for math

Semi-positive definite matrices

- Definition: a matrix positive semi-definite (PSD) M is if $\sum_{i,j} M_{ij} x_i x_j \geq 0$
- Theorem: A PSD symmetric square matrix can be written as $M = \sum_i \lambda_i e_i e_i^\top$

  where $\lambda_i \geq 0$ and $e_i \in \mathbb{R}^d$ are orthonormal vectors, namely

  $$\langle e_i, e_j \rangle = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases}.$$

Mercer's theorem

- Kernel k is positive semi-definite, namely $\sum_{i,j} \alpha_i \alpha_j k(x_i, x_j) \geq 0$ holds for all $x_i, x_j \in \mathbb{R}^d, \alpha_i \in \mathbb{R}$
- Theorem: A PSD kernel can be written as $k(x, y) = \sum_{i=1}^{\infty} \lambda_i e_i(x) e_i(y)$ where $e_i(x)$ are eigenfunctions $\int e_i(x) e_j(x) dx = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$.

Consequence of Mercer's theorem

- Mercer's Thm. conclusion: A function $k(x, y)$ is a positive definite kernel if and only if $k(x, y) = \langle \phi(x), \phi(y) \rangle$ where $\phi(x)$ is a feature map of arbitrary size.
- Example 1: dot product kernel $k(x, y) = \langle x, y \rangle$
- Example 2: Gaussian kernel $k(x, y) = \exp(-\frac{1}{2}\|x - y\|_2^2)$

Why a Gaussian kernel is a kernel?

Kernels and non-linear features

$$f^*(x) = \sum_i \alpha_i k(x, x_i) = \langle \phi(x), \underbrace{\sum_i \alpha_i \phi(x_i)}_{w^*} \rangle$$

- Kernel methods are equivalent to regression in high dimensional feature space $\phi(x)$
- What if we could find a finite ϕ that can approximate the kernel?

Group activity

- Mercer's representation: A PSD kernel can be written as $k(x, y) = \sum_{i=1}^{\infty} \lambda_i e_i(x) e_i(y)$ where $e_i(x)$ are eigenfunctions

$$\int e_i(x) e_j(x) dx = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

- Assume $\sum_i \lambda_i = 1$
- Find $\phi : \mathbb{R}^d \to \mathbb{R}$ such that $k(x, y) = \mathbb{E}\phi(x)\phi(y)$

Bochner's Theorem

Theorem. Suppose kernel $k(x, y)$ is shift invariant: $k(x, y) = k(x - y)$, then there is **non-negative measure** $p(w)$ such that $k(\Delta) = \int e^{i\langle w, \Delta \rangle} p(w) dw$, $\Delta := x - y$.

Kernel name	$k(\Delta)$	$p(w)$
Gaussian	$e^{-\frac{1}{2}\|\Delta\|_2^2}$	$(2\pi)^{-d/2} e^{-\frac{1}{2}\|w\|_2^2}$
Cauchy	$e^{-\|\Delta\|_1}$	$\prod_i \frac{1}{\pi(1 + w_i^2)}$
Laplacian	$\prod_i \frac{1}{\pi(1 + \Delta_i^2)}$	$e^{-\|w\|_1}$

[Rahimi & Recht 2007

Random features

- $k(x-y) = \int e^{-i\langle w, x-y\rangle} p(w) dw = \int e^{-i\langle w,x\rangle} e^{i\langle w,y\rangle} p(w) dw$

Lemma [Rahimi & Recht 2007]. Suppose that w is random variable with density p, and $b \sim$ uniform$[0,2\pi]$. Then, $k(x-y) = \mathbb{E}\left[\cos(\langle w, x\rangle + b)\cos(\langle w, y\rangle + b)\right]$ holds.

Question

- $k(x-y) = \int e^{-i\langle w, x-y\rangle} p(w) dw = \int e^{-i\langle w,x\rangle} e^{i\langle w,y\rangle} p(w) dw$

Lemma [Rahimi & Recht 2007]. Suppose that w is random variable with density p, and $b \sim$ uniform$[0,2\pi]$. Then,
$k(x-y) = 2\mathbb{E}\left[\cos(\langle w, x\rangle + b)\cos(\langle w, y\rangle + b)\right]$ holds.

Why a Gaussian kernel is a kernel?

Take home exercise

Theorem. Suppose kernel $k(x, y)$ is shift invariant: $k(x, y) = k(x - y)$, then there is non-negative measure $p(w)$ such that $k(\Delta) = \int e^{i\langle w, \Delta \rangle} p(w) dw$, $\Delta := x - y$.

Exercise: Given theorem, prove the lemma

Lemma [Rahimi & Recht 2007]. Suppose that w is random variable with density p, and $b \sim$ uniform$[0,2\pi]$. Then,
$k(x - y) = 2\mathbb{E}\left[\cos(\langle w, x \rangle + b)\cos(\langle w, y \rangle + b)\right]$ holds when $p(w) = p(-w)$.

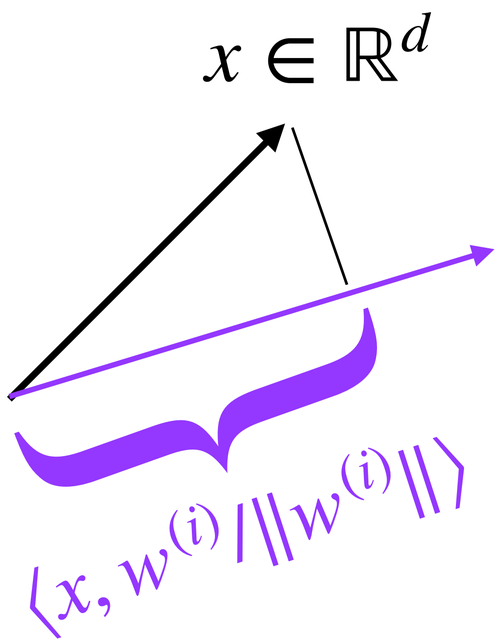
Random features

- $k(x-y) = \mathbb{E}_{w\sim p}\left[\cos(\langle w, x\rangle + b)\cos(\langle w, y\rangle + b)\right]$
- Sampling
  - Draw $w^{(1)}, \ldots, w^{(m)} \sim_{i.i.d} p(w)$ and $b_1, \ldots, b_n \sim$ uniform$[0,2\pi]$
  - $\phi_i(x) = \sqrt{\frac{2}{m}}\cos(\langle w^{(i)}, x\rangle + b_i)$
- Approximation [Rahimi & Recht 2007]
  - $$\sup_{x,y} \left| \frac{1}{m}\sum_{i=1}^{m} \phi_i(x)\phi_i(y) - k(x,y) \right| = O\left(\frac{d}{\sqrt{m}}\right) w\,.\,h\,.\,p$$

Random features

- Given x, design features $\phi_1(x), \ldots, \phi_{30}(x)$ such that $y \approx \sum_{i=1}^{30} w_i \varphi_i(x)$

$$\phi_i(x) = \cos(\langle w^{(i)}, x \rangle + b_i)$$



Recall: $\langle x, w \rangle = \sum_{i=1}^{d} x_i w_i = \|x\| \|w\| \cos(\theta)$

$b_i \sim \text{uniform}[0, 2\pi]$

Revisiting Lab (8mins)

- Given x, design features $\phi_1(x), \ldots, \phi_n(x)$ such that $y \approx \sum_{i=1}^{30} w_i \phi_i(x)$

  $\phi_{30}(x)$

  - Draw $w^{(1)}, \ldots, w^{(m)} \sim_{i.i.d} p(w)$ and $b_1, \ldots, b_n \sim$ uniform$[0, 2\pi]$
  - $\phi_i(x) = \sqrt{2/m} \cos(\langle w^{(i)}, x \rangle + b_i)$

Scan to Start

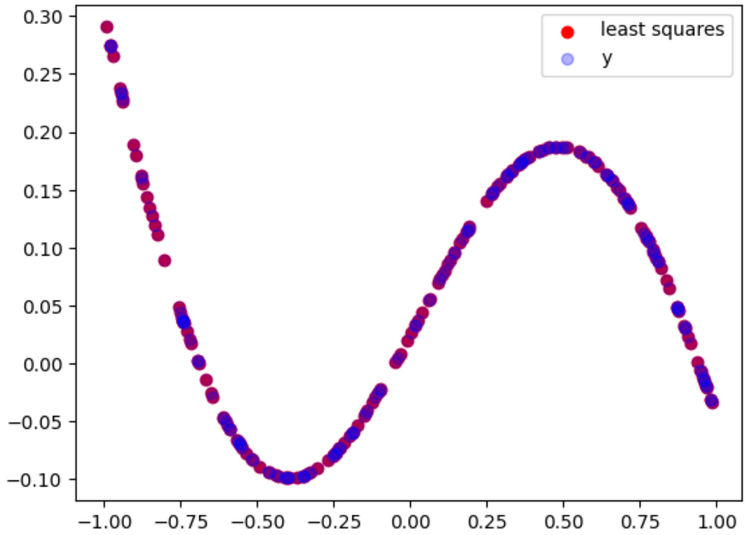
https://shorturl.at/ABb6H

https://colab.research.google.com/drive/1M-nRBdhg1XJiV8sy4wMkUsfPJLKKSCB0?usp=sharing

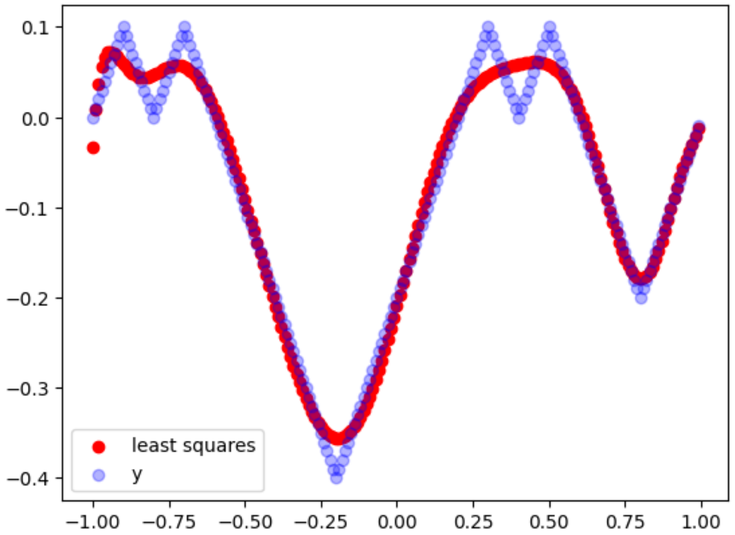
- You can search or use ChatGPT

Are random feature universal?

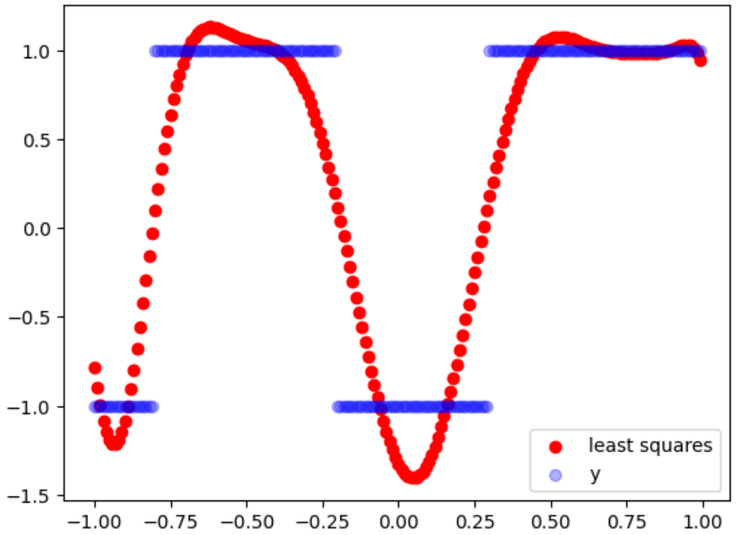
- Blue: y and red is the solution of regression on random features



Good approximation



Reasonable approximation



Poor approximation

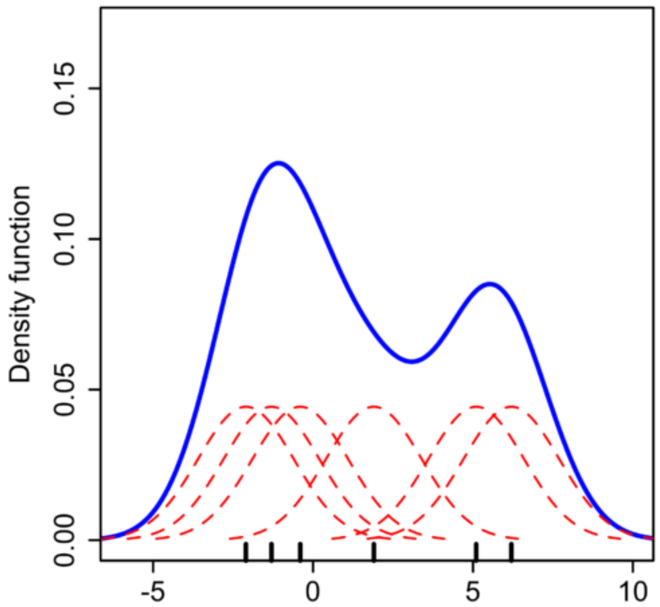
Universality of random features is analyzed in "Weighted Sums of Random Kitchen Sinks" by Ali Rahimi and Benjamin Recht

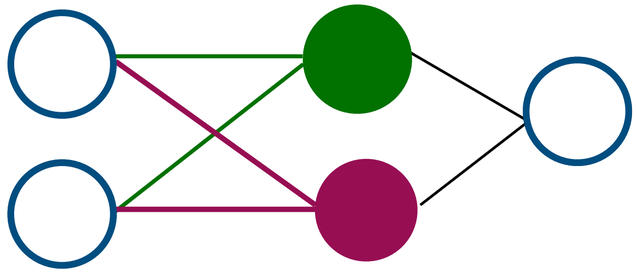
Machine learning universal tools

Kernel methods

Random features

Neural Nets





Kernel methods vs random features

Random Features

$$y \approx \sum_{i=1}^{m} \alpha_i \cos(\langle w^{(i)}, x \rangle + b_i)$$

$O(mn^2)$ computation

Approximate functions obeying reproducing property with an additional condition

Kernel Methods

$$y \approx \sum_{i=1}^{n} \alpha_i k(x, x_i)$$

$O(n^3)$ computation

Approximate functions obeying reproducing property

Neural Networks vs Random features

Random Features

$$y \approx \sum_i \alpha_i \cos(\langle w^{(i)}, x \rangle + b_i)$$

$w^{(i)} \sim p(w)$ depending on $k(x, y)$

Approximation depends on the kernel

Neural Networks

$$y \approx \sum_i \alpha_i \cos(\langle w^{(i)}, x \rangle + b_i)$$

$w^{(i)}$ are optimized since $p(w)$ is unknown

General approximation guarantees

A controversial talk

Random features won the test-of-time award at NIPS 2017 (23min)

The next lecture topic

- A systematic issue with random features