Q3 of HW2

Reference:

“Random Features for Large-Scale Kernel Machines”

Test of Time Award, NeurlPS (NIPS) 2017

High level idea

- Motivation:

Classification easier

- Issue:

Expensive

- Solution:

Linear

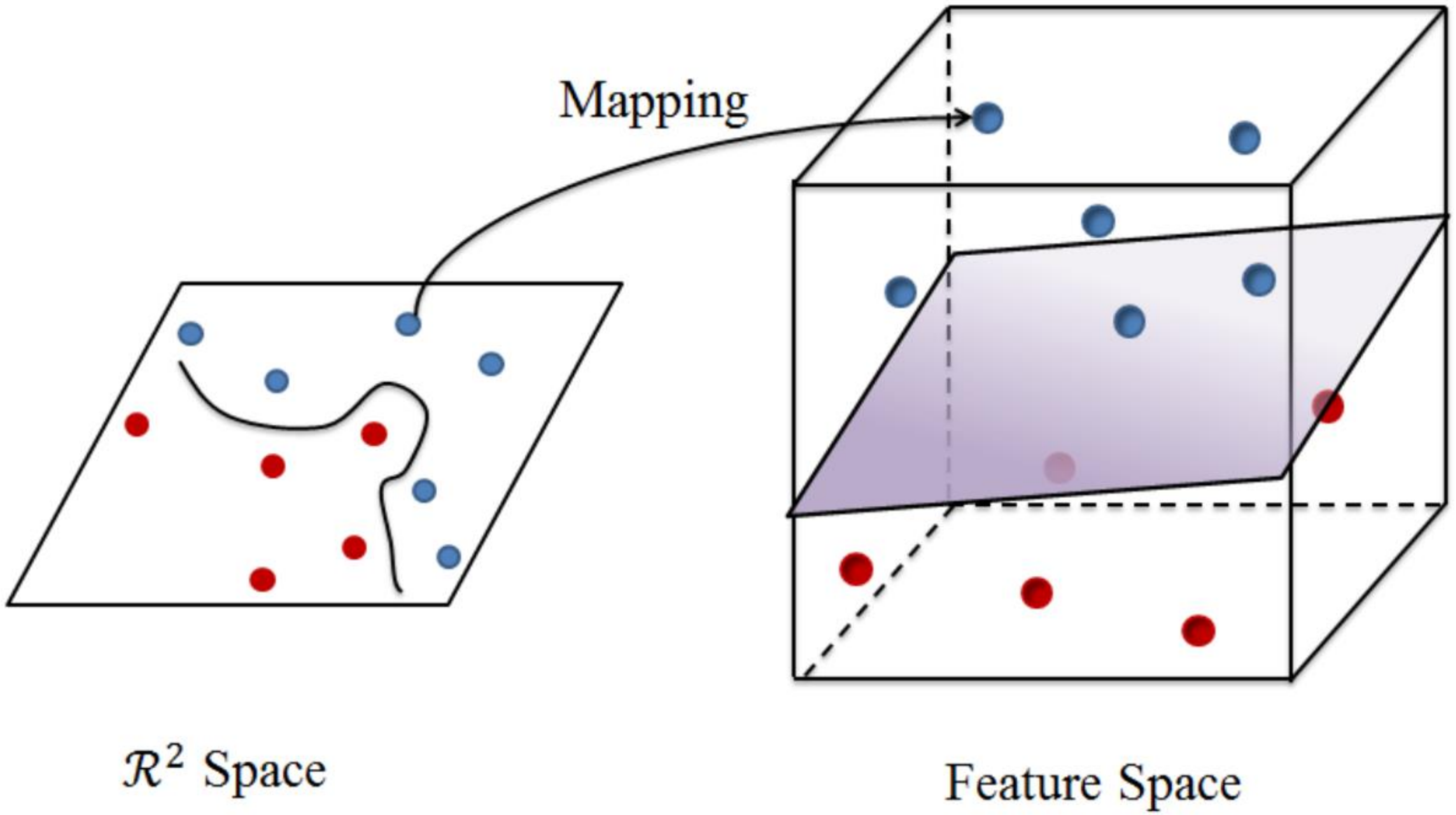


Figure reference: http://songcy.net/posts/story-of-basis-and-kernel-part-2/

Detail

Kernels:

, $k(\boldsymbol{x}_1 - \boldsymbol{x}_2) = e^{-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2}{2\sigma^2}}$ 。

:, $k(\boldsymbol{x}_1 - \boldsymbol{x}_2) = e^{-\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_1}$ 。

$k(\boldsymbol{x}_1 - \boldsymbol{x}_2) = \prod_d \frac{2}{1+\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_d^2}$ 。

$$\begin{aligned} k(\boldsymbol{x}_1 - \boldsymbol{x}_2) &= \int_{\mathbb{R}^d} p(\boldsymbol{w}) e^{-j\boldsymbol{w}^\top(\boldsymbol{x}_1 - \boldsymbol{x}_2)} \mathrm{d}\boldsymbol{w} \\ &= \int_{\mathbb{R}^d} p(\boldsymbol{w}) \cos(\boldsymbol{w}^\top(\boldsymbol{x}_1 - \boldsymbol{x}_2)) \mathrm{d}\boldsymbol{w} \\ &= \mathbb{E}_{\boldsymbol{w}\sim p(\boldsymbol{w})}[\cos(\boldsymbol{w}^\top(\boldsymbol{x}_1 - \boldsymbol{x}_2))] \\ &= \mathbb{E}_{\boldsymbol{w}\sim p(\boldsymbol{w})}[\cos(\boldsymbol{w}^\top\boldsymbol{x}_1)\cos(\boldsymbol{w}^\top\boldsymbol{x}_2) + \sin(\boldsymbol{w}^\top\boldsymbol{x}_1)\sin(\boldsymbol{w}^\top\boldsymbol{x}_2)] \end{aligned}$$

$$\mathbb{E}_{\boldsymbol{w}\sim p(\boldsymbol{w})}[\cos(\boldsymbol{w}^\top\boldsymbol{x}_1)\cos(\boldsymbol{w}^\top\boldsymbol{x}_2) + \sin(\boldsymbol{w}^\top\boldsymbol{x}_1)\sin(\boldsymbol{w}^\top\boldsymbol{x}_2)] \approx \frac{1}{n}\sum_{i=1}^{n} \cos(\boldsymbol{w}_i^\top\boldsymbol{x}_1)\cos(\boldsymbol{w}_i^\top\boldsymbol{x}_2) + \sin(\boldsymbol{w}_i^\top\boldsymbol{x}_1)\sin(\boldsymbol{w}_i^\top\boldsymbol{x}_2)$$

$$z(\boldsymbol{x}_1) = \frac{1}{\sqrt{n}}\left(\cos(\boldsymbol{w}_i^\top\boldsymbol{x}_1), \ldots, \cos(\boldsymbol{w}_D^\top\boldsymbol{x}_1), \sin(\boldsymbol{w}_i^\top\boldsymbol{x}_1), \ldots, \sin(\boldsymbol{w}_n^\top\boldsymbol{x}_1)\right)$$

$$z(\boldsymbol{x}_2) = \frac{1}{\sqrt{n}}\left(\cos(\boldsymbol{w}_i^\top\boldsymbol{x}_2), \ldots, \cos(\boldsymbol{w}_D^\top\boldsymbol{x}_2), \sin(\boldsymbol{w}_i^\top\boldsymbol{x}_2), \ldots, \sin(\boldsymbol{w}_n^\top\boldsymbol{x}_2)\right)$$

$$p(\boldsymbol{w}) = \frac{1}{(2\pi)^n}\int_{\mathbb{R}^d} e^{j\boldsymbol{w}^\top(\boldsymbol{x}_1 - \boldsymbol{x}_2)} k(\boldsymbol{x}_1 - \boldsymbol{x}_2)\mathrm{d}(\boldsymbol{x}_1 - \boldsymbol{x}_2) \Rightarrow p(\boldsymbol{w}) = e^{-\frac{\|\boldsymbol{w}\|_2^2\sigma^2}{2}}\sigma^d(2\pi)^{-\frac{d}{2}} \sim \mathcal{N}(\mathbf{0}, \Sigma)$$

Implementation

- Implement:
  - theta, Omega, B= random_fourier_features(data_X, data_Y, num_fourier_features=10, alpha=0.1, num_iters=500)
  - vis_rff_model(train_X, train_Y, theta, Omega, B)
- Result
  - 12 Figure with 4 dataset, and each dataset with 3 different num_fourier_features(K)

Note: for input data_X of random_fourier_features has one 1 column appended.

In random_fourier_features

Algorithm 1 Random Fourier Features.

Require: A positive definite shift-invariant kernel $k(\mathbf{x}, \mathbf{y}) = k(\mathbf{x} - \mathbf{y})$.

Ensure: A randomized feature map $\mathbf{z}(\mathbf{x}) : \mathcal{R}^d \to \mathcal{R}^D$ so that $\mathbf{z}(\mathbf{x})'\mathbf{z}(\mathbf{y}) \approx k(\mathbf{x} - \mathbf{y})$.

Compute the Fourier transform p of the kernel k: $p(\omega) = \frac{1}{2\pi} \int e^{-j\omega'\delta} k(\delta)\, d\Delta$.

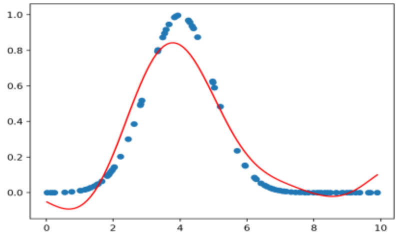
Draw D iid samples $\omega_1, \cdots, \omega_D \in \mathcal{R}^d$ from p and D iid samples $b_1, \ldots, b_D \in \mathcal{R}$ from the uniform distribution on $[0, 2\pi]$.

Let $\mathbf{z}(\mathbf{x}) \equiv \sqrt{\frac{2}{D}} \left[\cos(\omega_1'\mathbf{x}+b_1) \cdots \cos(\omega_D'\mathbf{x}+b_D) \right]'$.

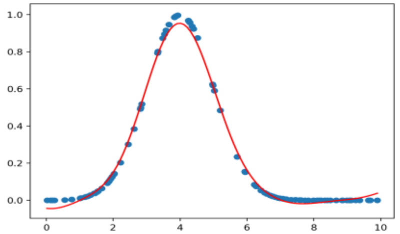
- Only Calculate Omega and B
- What’s the Omega?
  - iid samples ω1,··· ,ωD ∈ Rd from p, You can generate Omega using normal random distribution with mean = 0, sigma = 1
- What’s the B?
  - B = b1,...,bD ∈ R from the uniform distribution on [0, 2π]

Example Result for 1D-exp-samp.txt

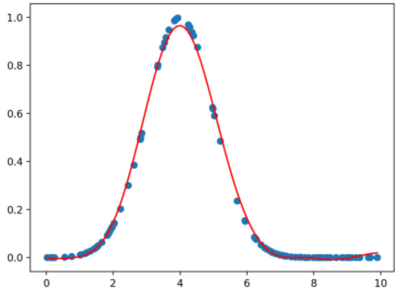
- K = 10



- K = 100



- K = 800



Please also generate the figure for other 3 dataset.

Thanks!

Feel free to email for the additional office hour for more questions!