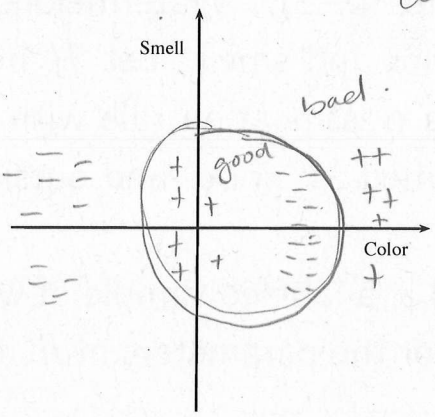
Additional assumption: The sample size was large, and h^* had training error of ~ 0.4

low training ERROR

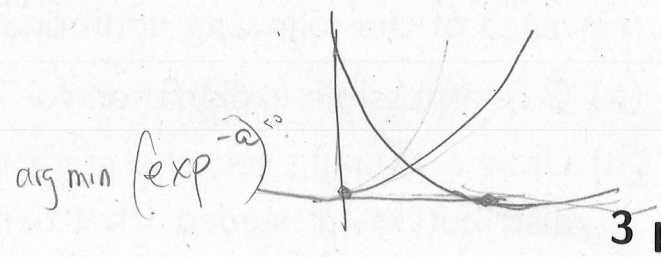


circle would never capture the structure of bimodial data

10 sets of N fruit and 4/10 will be miss-classified **8 pts.**

2. Suggest a way to measure the empirical variance of the classifier h^*, given that we are out of budget for obtaining more watermelon samples. you want to penalize the size of the circle.

$\Omega(w_1, w_2, r) = r^2$



3 pts.

Figure 2 depicts the dataset we had, and h^* that we got using equation (2). To improve h^* we decided to add a regularizing term, the new criterion will be

$$\underset{w_1,w_2,r}{\operatorname{argmin}} \sum_{i=1}^{n} \exp(-y_i[r^2 - ((x_{i1} - w_1)^2 + (x_{i2} - w_2)^2)]) + \lambda\Omega(w_1, w_2, r) \tag{3}$$

Where $\Omega(w_1, w_2, r)$ is the regularizer. We ask you to suggest a suitable regularizer.

4. (a) Draw on Figure 2 the regularized solution you envision.

   (b) Write down the mathematical term of the regularizer. Explain your answer.

$\Omega(w_1, w_2, r) =$ R^2 in L2-norm.

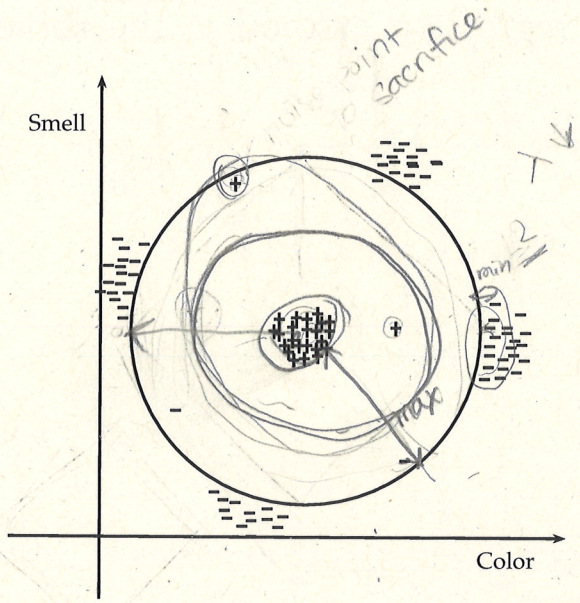


Figure 2: Watermelons dataset and h^*

5 pts.

5. Assume that the true distribution of watermelons consists of high density regions visible in Figure 2, plus sparse outliers. Explain what happens to the variance of h^* as we increase λ compared to some starting value $\lambda_0 > 0$.

The size the circle will decrease. $\lambda \downarrow$

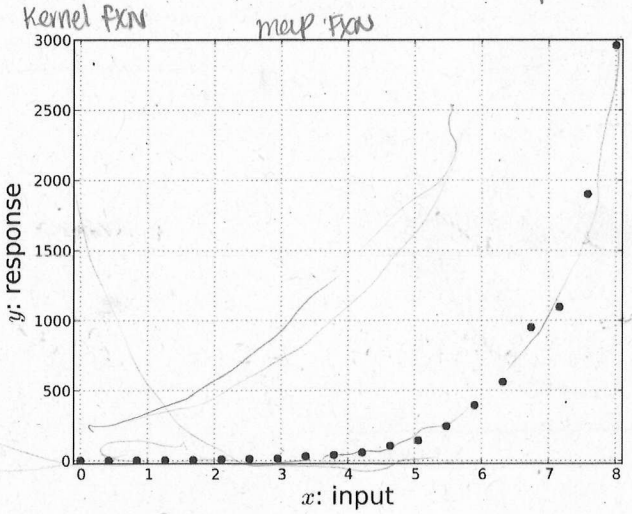
variance ↓ 150

4 pts.

Question 4: Kernelized Ridge Regression (20 pts.)

Recall the regression setting: Given input vectors $\mathbf{x}_i$, and output (response) variables y_i, the goal is to find a functional relation between them, often expressed with a weight vector $\mathbf{w}$ and bias b.

1. Below is a dataset with one dimensional input variables x, and response variable y. Your task is to find a kernel function $K(x_i, x_j)$, such that you can use a linear regression method in the kernel space.



$K(x_i, x_j) = \ldots$ **3 pts.**

2. You will now derive a kernelized version of ridge regression by introducing a feature transform $\mathbf{\Phi}(\mathbf{x}_i)$. This should allow a non-linear regression solution, for datasets such as the one depicted above.

   Recall the formulation of ridge regression as an optimization problem:

$$\min_{\mathbf{w},b} \sum_i (y_i - \mathbf{w}^\top \mathbf{x}_i - b)^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2 \qquad (4)$$

   (a) We replace the inputs $\mathbf{x}_i$ in Equation (4) with the vectors of the features in the kernel space and rewrite the problem as a

constrained optimization problem by introducing the new variables ξ_i. Write down the equality constraint in Equation (6).

primal

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \quad \sum_i \xi_i^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2 \tag{5}$$

$$\text{s.t.} \qquad \xi_i = \ldots \tag{6}$$

2 pts.

(b) Write down the Lagrangian of this new optimization problem using α as the dual variable.

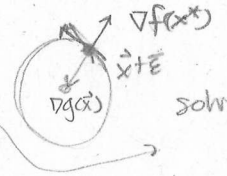
objective of lagrangian is to minimize the problem.

max $f(\vec{x})$

s.t. $g(\vec{x})=0$ $\quad \mathcal{L}(\vec{x},\lambda) = f(\vec{x}) + \lambda g(\vec{x})$

$\vec{x} \in \mathbb{R}^D$

$\nabla g(\vec{x}) \perp \quad g(\vec{x}+\vec{\varepsilon}) \simeq g(\vec{x}) + \vec{\varepsilon}^T \nabla g(x)^T \quad \|\vec{\varepsilon}\| \to 0$



solve the larangian

$\frac{\partial L}{\partial x} = 0$

$\frac{\partial L}{\partial \lambda} = g(x) = 0$

(c) Derive the dual optimization problem.

objective fxn

$L(\vec{\alpha}, \vec{w}, b, \vec{\xi}) = \sum_i \xi_i^2 + \frac{\lambda}{2}\|\vec{w}\|^2 + \sum_i \alpha_i (y_i - \vec{w}^T \vec{\phi}_i - b - \xi_i)$

$\vec{w} = \begin{pmatrix} b \\ \vec{w} \end{pmatrix} \quad \vec{x} = \begin{pmatrix} 1 \\ \vec{x} \end{pmatrix}$

$\frac{\partial L}{\partial \vec{w}} = -\sum_i \alpha_i \vec{\phi}_i + \lambda \vec{w} = 0 \Rightarrow \boxed{\vec{w} = \frac{\sum_i \alpha_i \vec{\phi}_i}{\lambda}}$ this is why you derive on α.

$\frac{\partial L}{\partial \xi_i} = 2\xi_i - \alpha_i = 0 \Rightarrow \boxed{\xi_i = \frac{\alpha_i}{2}}$

$\|\vec{w}\|^2 = \frac{\lambda}{2} \cdot \frac{1}{\lambda^2} \left\| \sum_i \alpha_i \vec{\phi}_i \right\|^2 = \sum_{ij} \alpha_i \alpha_j \vec{\phi}_i^T \vec{\phi}_j$

You want to get the objective fxn on alpha.

$L(\vec{\alpha}) = -\frac{1}{4}\sum_i \alpha_i^2 + \sum_i \alpha_i y_i - \frac{1}{2\lambda} \sum_{ij} \alpha_i \alpha_j \vec{\phi}_i^T \vec{\phi}_j \quad , \quad \min_{\vec{\alpha}} L(\vec{\alpha})$

8 pts.

3. Express the dual problem in terms of the kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$.

$K(\vec{x}_i, \vec{x}_j) = \vec{\phi}_i^T \vec{\phi}_j$

3 pts.

4. Given the optimal solution of the dual problem $\boldsymbol{\alpha}^\star$ and a new point $\mathbf{x}_k$, write down the equation to compute y_k.

$y_k = \vec{w}^T \vec{\phi}(\vec{x}_k)$

↑ plug in $\vec{w} = \sum_i \underline{\alpha_i \vec{\phi}_i}$ (λ)

$= \lambda \sum_i \alpha_i^* \vec{\phi}_i^T \vec{\phi}_k = \frac{1}{\lambda} \sum_i \alpha_i^* K(\vec{x}_i, \vec{x}_j)$

4 pts.

Question 5: Unsupervised Learning (20 pts.)

1. In this section we study non-parametric density estimation of an arbitrary point x. We consider some small region $\mathcal{R}$ containing x. In the class we have seen the following generic formula for density estimation:

$$p(x) = \frac{K}{nV},$$

where K denotes the number of data points falling inside the region $\mathcal{R}$ and V shows the volume of the region. n is the number of data points in the sample set $\mathcal{S} = \{x_1, ..., x_n\}$.

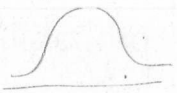
(a) Consider the following Gaussian distribution to be used as a Parzen window function:

$$\phi(x - x_j) = \frac{1}{(2\pi)^{1/2}} \exp\left(-\frac{(x - x_j)^2}{2}\right). \quad (7)$$

What are K and V for this window function?

$K = \sum_{i=1}^{n} \phi(x - x_j)$

$V = 1$ ← because a probability distribution



4 pts.

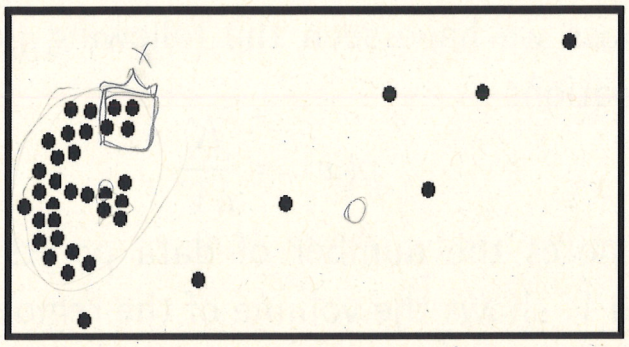
(b) This particular choice of a window function leads to underfitting. Add a parameter to increase the model complexity.

Add variance parameter. $= h$, controls smoothness

$$\phi\left(\frac{(x - x_j)^2}{h}\right) = \frac{1}{(2\pi)^{\frac{1}{2}} \cdot h} \exp\left(-\frac{(x - x_j)^2}{2h^2}\right)$$

2 pts.

(c) Consider the following sample set. Which of the density estimation methods would you choose? Window-based (Eq. (7)) or K-nearest neighbor? Explain your answer.



Choose K-nearest neighbor. → local-based

3 pts.

(d) For a general Parzen window function prove that it provides a probability distribution.

MUST satisfy two conditions: ① $\phi(\vec{u}) \geq 0$
close to a probability distribution ② $\int \phi(\vec{u})\, d\vec{u} = 1$

$$p(x) = \frac{k}{nV} = \frac{1}{n} \sum_{j=1}^{n} \int_{-\infty}^{\infty} \phi(x - x_j) \Rightarrow \int p(x)\, dx = 1$$

$\frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}}$ $\frac{1}{n} \sum \frac{1}{h^V} \phi\left(\frac{x - x_j}{h}\right)$

$\frac{1}{\sqrt{2\pi\sigma^2}}$

4 pts.

2. We consider a mixture of K poisson distributions and perform the Expectation-Maximization (EM) algorithm to compute the unknown parameters. The log-likelihood function of n independent objects for mixture of K Poisson distribution is defined as:

$$P(x;\lambda) = \sum_{i=1}^{n} \log \sum_{c=1}^{K} \pi_c f(x_i;\lambda_c)$$

where π_c's are the mixture weights and λ_c's are the parameters of K Poisson distributions. $f(x_i;\lambda_c)$ is defined as:

$$f(x;\lambda_k) = \frac{\lambda_k^x e^{-\lambda_k}}{x!}.$$

(a) Introduce the latent indicator variables necessary for maximizing the log-likelihood function.

$\log P(x|\vec{\lambda},\vec{\pi}) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k f(x_n|\lambda_k)$

$Z \in \mathbb{R}^{N \times K}$, each row is a vector for a point

$z_{nk} = \begin{cases} 1 : \text{component } k \text{ is responsible for generated } x_n \\ 0 : \text{ow} \end{cases}$

2 pts.

(b) Calculate the expectation of the latent variables. Provide a Bayesian interpretation for your answer.

E-step: estimate posterior distribution of data.

$P(Z|X,\theta^{old}) = E_Z(z|\dots)$

M-step: $\theta^{new} = \arg\max_{\theta} Q(\theta,\theta^{old})$

$Q(\theta,\theta^{old}) = \sum_{z} P(z|x,\theta^{old}) \cdot \log P(X,Z|\theta)$; complete data log-likelihood

joint distribution

$E[z_{nk}] = P(z_{nk}=1|\vec{x}_n) * 1 = \frac{P(z_{nk}=1)\,P(\vec{x}_n|z_{nk}=1)}{\sum_{z_1} P(x_n,z_n)} = \frac{\pi_k \cdot P(x_n|\lambda_k)}{\sum_{j=1}^{k} \pi_j \cdot P(x_n|\lambda_j)}$

Responsibility $\lambda(z_{nk})$

$P(\vec{z}_n|\vec{\pi}) = \prod_{k=1}^{K} \pi_k^{z_{nk}}$

$\Rightarrow P(x_n|z_n,\lambda_k) = \prod_{k=1}^{K} P(x_n|\lambda_k)^{z_{nk}}$

5 pts.

conditional expectations

(c) **Bonus question:** Assume the expectations of the latent variables are given. Calculate the unknown parameters λ_c's. Write down the details of your calculations.

objective fxn $Q = \underset{z}{E}\left[\log P(x, z|\vec{\lambda}, \vec{\pi})\right]$

$$= \left[\prod_{n=1}^{N} \prod_{k} \{\pi_k P(\vec{x}_n|\lambda_k)\}\right]^{z_{nk}}$$

$$= \sum_n \sum_k \gamma(z_{nk}) \{\log \pi_k + x_n \log \lambda_k - \lambda_k - \log x\}$$

pg. 445, 430.

5 pts.

Supplementary Sheet

NO 'fixed "right solution", as in real life. → the accuracy is the only goal.

Google solution for insights and do it better → do not reinvent the wheel.

CV scheme is really important

SVM → assume metric space behind data.

Q/A session before exam.

Supplementary Sheet

Supplementary Sheet

Supplementary Sheet