NetID, last name, first name

General Instructions

- Put your name on ALL sheets, including extra sheets
- Show all work for credit
- Electronic devices, books, lecture notes etc. are *not* allowed
- Work efficiently: (1) work on the easy and short problems first to quickly collect points and (2) note that subproblems typically can be solved independently from the other subproblems
- Use the back of the page if you need more space and clearly mark on the front of the page if we are to look at what's on the back
- Good luck!

P1	P2	P3	$\sum$
/12	/14	/14	/40

1 Problem (12 points)

Explain in 1-2 sentences.

1. [2 points] True or False: The training error of a nearest neighbor classifier with 1 neighbor is 0. *Explicitly* state True or False, additionally to your one-sentence reasoning.

2. [2 points] Why does the kernel trick allow us to solve SVMs with high dimensional feature spaces, without significantly increasing the running time?

3. [2 points] Let $k_1(x, z) = \phi_1(x)^T \phi_1(z)$ and $k_2(x, z) = \phi_2(x)^T \phi_2(z)$ be kernels corresponding to feature map ϕ_1 and ϕ_2, respectively. Show that $k(x, z) = k_1(x, z) + k_2(x, z)$ is a kernel. (*Hint:* Show that there exists a feature map ϕ so that $k(x, z) = \phi(x)^T \phi(z)$.)

4. [3 points] Suppose you have regression data generated by a polynomial of degree 3. Characterize the bias-variance of the estimates of the following models on the data with respect to the true model by circling the appropriate entry.

	Bias	Variance
Linear regression	low/high	low/high
Polynomial regression with degree 3	low/high	low/high
Polynomial regression with degree 10	low/high	low/high

5. [3 points] True or False (no explanation needed)?

   (a) In SVMs, the values of α_i (dual formulation) for non-support vectors are 0?

   (b) Cross validation will guarantee that our model does not overfit?

   (c) SVMs directly give us the probabilities that a data point is in class -1 or 1, i.e., $p(y = 1|x)$ or $p(y = -1|x)$?

2 Problem (14 points)

Consider the training data set $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)})\}$ with $N = 2$ points $(x^{(1)}, y^{(1)}) = (1, 1)$ and $(x^{(2)}, y^{(2)}) = (2, 0)$. Note that the class labels are either 0 or 1. For a parameter θ, the hypothesis in logistic regression is given by

$$h_\theta(x) = \frac{1}{1 + \exp(-\theta^T x)} \,.$$

Recall that $h_\theta(x) \in [0, 1]$ and that $h_\theta(x)$ gives the probability that x is in class $y = 1$ and $1 - h_\theta(x)$ gives the probability that x is in class $y = 0$:

$$\begin{aligned} p(y = 1|x) &= h_\theta(x) \\ p(y = 0|x) &= 1 - h_\theta(x) \,. \end{aligned}$$

The classification threshold is given by 0.5 in this example so that

$$y = \begin{cases} 1 \,, & h_\theta(x) \geq 0.5 \\ 0 \,, & \text{otherwise} \,. \end{cases}$$

1. [6 points] Consider logistic regression *without* an intercept term, which means in this example that $\theta \in \mathbb{R}$ is a scalar. Proof that no $\theta \in \mathbb{R}$ exists such that h_θ classifies the training data set $\mathcal{D}$ without classification error (i.e., there is no $\theta \in \mathbb{R}$ such that the two points $x^{(1)}$ and $x^{(2)}$ in $\mathcal{D}$ get assigned the correct label).

2. [4 points] Consider now logistic regression with an intercept term, i.e., the parameter is now 2-dimensional $\boldsymbol{\theta} = [\theta_1, \theta_2]^T \in \mathbb{R}^2$. As in class, we use the convention that the first component of the data points is constant 1, which means in our example that $\boldsymbol{x}^{(1)} = [1, x^{(1)}]^T = [1, 1]^T$ and $\boldsymbol{x}^{(2)} = [1, x^{(2)}]^T = [1, 2]^T$. Fit the parameter $\boldsymbol{\theta}$ via the principle of maximum likelihood and gradient descent. The gradient descent update step is given by

$$\begin{bmatrix} \theta_1^{(k+1)} \\ \theta_2^{(k+1)} \end{bmatrix} = \begin{bmatrix} \theta_1^{(k)} \\ \theta_2^{(k)} \end{bmatrix} + \alpha \begin{bmatrix} \sum_{i=1}^N (y^{(i)} - h_{\boldsymbol{\theta}^{(k)}}(\boldsymbol{x}^{(i)})) x_1^{(i)} \\ \sum_{i=1}^N (y^{(i)} - h_{\boldsymbol{\theta}^{(k)}}(\boldsymbol{x}^{(i)})) x_2^{(i)} \end{bmatrix}, \tag{1}$$

where $\boldsymbol{\theta}^{(k)} = [\theta_1^{(k)}, \theta_2^{(k)}]^T$ is the parameter at iteration k and $x_1^{(i)}$ is the first component of $\boldsymbol{x}^{(i)}$ and $x_2^{(i)}$ is the second component of $\boldsymbol{x}^{(i)}$. Set $\boldsymbol{\theta}^1 = [0.6, -1]^T$ and the step size $\alpha = 1$. Perform *one* iteration. Report $\boldsymbol{\theta}^{(2)}$. Are the two training data points classified correctly?

You may use the following simplifications

$$\frac{1}{1+\exp(-0.2)} \approx 0.55\,, \quad \frac{1}{1+\exp(0.4)} \approx 0.4\,, \quad \frac{1}{1+\exp(0.6)} \approx 0.35\,, \quad \frac{1}{1+\exp(1.4)} \approx 0.2\,.$$

NetID, last name, first name

3. [4 points] Write Python (or similar) code that implements the update (1) and iterates until the Euclidean norm of $\boldsymbol{\theta}^{(k)} - \boldsymbol{\theta}^{(k-1)}$ is below the threshold $e = 10^{-12}$. Assume the following numpy arrays are given

$$X = \begin{bmatrix} | & | \\ \boldsymbol{x}^{(1)} & \boldsymbol{x}^{(2)} \\ | & | \end{bmatrix}, \qquad Y = \begin{bmatrix} y^{(1)} & y^{(2)} \end{bmatrix}$$

so that `X.shape` gives $(2, 2)$ and `Y.shape` gives $(1, 2)$. You may use the numpy and the math packages but no other packages.

Note: Make sure that the logic of the program is correct, rather than getting caught up in syntax details.

3 Problem (14 points)

Let $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})\}$ be N training data points with labels $y^{(i)} \in \{-1, 1\}$ for $i = 1, \ldots, N$. The following is the primal formulation of SVMs with slack

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i \\ \text{such that} \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i\,, \quad i = 1, \ldots, N \\ & \xi_i \geq 0\,, \qquad\qquad\qquad\quad i = 1, \ldots, N\,, \end{aligned} \tag{2}$$

which corresponds to minimizing the regularized Hinge loss

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \ell_{\text{hinge}}(w^T x^{(i)} + b, y^{(i)})\,, \tag{3}$$

with $\ell_{\text{hinge}}(\hat{z}, z) = \max(0, 1 - \hat{z}z)$. Consider now the following optimization problem

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \boldsymbol{\xi_i^2} \\ \text{such that} \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i\,, \quad i = 1, \ldots, N \\ & \xi_i \geq 0\,, \qquad\qquad\qquad\quad i = 1, \ldots, N\,, \end{aligned} \tag{4}$$

which is the same as (2) except that the squared ξ_i is used in the objective (highlighted in bold). Similarly as the Hinge loss (3) corresponds to the SVM with slack formulation (2), we now want to derive the loss function $\ell_?$ that corresponds to the formulation (4) with squared slack variables.

1. [3 points] Show that removing the set of constraints $\xi_i \geq 0\,, i = 1, \ldots, N$ does *not* change the optimal solution of (4).

2. [3 points] Derive the Lagrangian of (4). Ignore the constraints $\xi_i \geq 0\,, i = 1, \ldots, N$.

3. [4 points] Derive the optimal values $\xi_i^*, i = 1, \ldots, N$ of (4). Maybe helpful: (a) Use the result from Part 1 of this problem (i.e., you can ignore the constraints $\xi_i \geq 0\,, i = 1, \ldots, N$). (b) Recall that for each training data point $(x^{(i)}, y^{(i)}), i = 1, \ldots, N$ the corresponding slack variable ξ_i^* is either 0 or non-zero.

4. [4 points] Take the optimal values ξ_i^* that you have derived in Part 3 of this problem and plug them into the objective of (4) and write in the form

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \ell_?(\cdot,\cdot)\,.$$

Which loss function $\ell_?$ do you obtain?

NetID, last name, first name

[Extra sheet]

NetID, last name, first name

[Extra sheet]