**Problem 1 (General Machine Learning)** **3+3+3+3+3+3 = 18 Points**

a) What is the main advantage of having a convex loss function?

☐ They are relatively easy to optimize.

☐ They avoid overfitting.

☐ Once optimized, they tend to perform better than non-convex algorithms.

b) The k-means algorithm is an example of which of the following?

☐ Unsupervised learning

☐ Regression

☐ Classification

c) The gradient of a function $f$ points in the direction of ______ of $f$.

☐ steepest ascent

☐ steepest descent

☐ the global minimum

d) What is the aim of supervised machine learning?

e) Describe the phenomena of “overfitting” and give an example of a technique one can employ to avoid it.

f) Describe how the random forest algorithm works. You may assume that it is known how decision trees work.

**Problem 2 (Support Vector Machines)** **5 + 5 + 5 = 15 Points**

Let $(x_i, y_i) \in \mathbb{R}^d \times \{-1, 1\}$, $i = 1, \dots, n$, be classification training data. Consider the following variation on the soft margin support vector machine:

$$\begin{aligned} \min_{w,\xi} \quad & \frac{1}{2} \|w\|^2 + \frac{C}{2n} \sum_{i=1}^{n} \xi_i^2 \\ \text{s.t.} \quad & y_i \left( \langle w, x_i \rangle \right) = 1 - \xi_i \text{ for } i = 1, 2, \dots, n. \end{aligned}$$

Note the equality in the constraint and that $\xi_i$ can be non-negative.

a) Construct the Lagrangian. Note that $\alpha = (\alpha_1, \dots, \alpha_n)$ can be negative since we have equality constraints.

b) Take the derivative of the Lagrangian and express the optimal $w, \xi$ in terms of equations.

c) Plug the equations for the optimal $w, \xi$ back into the Lagrangian and write down the resulting dual problem, which will now depend only on the variables $\alpha_1, \dots, \alpha_n$.

---

**Problem 3 (Kernels)** **3+3+3+4+4 = 17 Points**

a) Describe the "kernel trick" and explain why it is useful in machine learning.

b) For an input dimension $d$, of roughly what order is the feature map of the polynomial kernel of order 2?

- ☐ $\log(d)$
- ☐ $d^2$
- ☐ $d^3$
- ☐ $e^d$

c) Write down the precise definition (equation) of a kernel mentioned in lecture.

d) *Note: this question was considered too hard for ML1 students.*

Let $k(\cdot,\cdot)$ be a kernel. Prove that $k'(x, y) = k(x, y) + \delta(x, y)$ is also a kernel, where $\delta(x, y)$ is 1 if $x = y$ and 0 if $x \neq y$. You may utilize standard results regarding matrix properties without proof.

e) While kernels in lecture were always used to transform euclidean space, kernels can also be used to transform other spaces as well. Let $k(\cdot,\cdot)$ be a kernel on $\{0, 1\}$ with $k(0, 0) = k(1, 1) = 1$ and $k(0, 1) = k(1, 0) = \frac{1}{2}$. Find an explicit feature mapping for this kernel to euclidean space **or** geometrically describe the relationship between two vectors from such a feature mapping.

---

**Problem 4 (Regression)** **4 + 3 + 4 + 5 = 16 Points**

a) One can construct a very crude regressor by adapting the linear least squares algorithm to have no linear term, i.e. just a constant term:

$$\min_b \sum_{i=1}^{n} \left( y_i - \left( \cancel{x_i^T w} + b \right) \right)^2 = \min_b \sum_{i=1}^{n} (y_i - b)^2 \,.$$

What is the solution to this regressor?

b) This regressor is quite basic; perhaps it would be good to make it a bit more powerful. Let $k(q, r) = \exp\left( - \|q - r\|^2 \right)$. Consider the following regressor which returns $\hat{y}_0$ for a test point $x_0$:

$$\hat{y}_0 = \arg\min_b \sum_{i=1}^{n} k(x_0, x_i) \, (y_i - b)^2 \,.$$

This regressor is an example of what is called a *Nadaraya–Watson kernel regressor*. In your own words, describe how this regressor changes the behavior of the regressor in part a and explain why this change could be beneficial.

c) The loss function for linear regression is

$$\sum_{i=1}^{n} \left( y_i - \left( x_i^T w + b \right) \right)^2 .$$

This loss function can be adapted to yield a new regressor which also has a reasonable loss function,

$$\sum_{i=1}^{n} \left| y_i - \left( x_i^T w + b \right) \right| .$$

What is a possible advantage and disadvantage to this formulation?

d) Ridge regression minimizes the following loss function:

$$w^\star = \arg\min_w \lambda \|w\|^2 + \sum_{i=1}^{n} \left( y_i - x_i^T w \right)^2 .$$

Derive a closed form solution for the minimizer $w^\star$.

---

**Problem 5 (Principal Component Analysis)** **5 + 5 + 5 = 15 Points**

Suppose you are given data $x_1, \ldots, x_n \in \mathbb{R}^d$.

a) Write pseudocode that calculates the top $d' < d$ principal components of the data.

b) Let $v_1, \ldots, v_{d'} \in \mathbb{R}^d$ be the principal components found in part a. Write pseudocode that reduces the dimensionality of $x_1, \ldots, x_n$ into new samples $x'_1, \ldots, x'_n \in \mathbb{R}^{d'}$.

c) Suppose that the reduced dimensionality, $d'$, is 2. Write **code in Python or Matlab** which displays the reduced dimension data as points on the euclidean plane, i.e. a scatter plot. You may assume that a variable X_red is already defined in a reasonable data structure. If you do use a library, be sure to include all code necessary to utilize it, e.g. do not assume it has already been imported.

---

**Problem 6 (k-Means Algorithm)** **5 + 3 = 8 Points**

a) The following pseudocode implementing k-means contains an error. Explain how running this algorithm will differ from running a correctly implemented k-means algorithm.

**Algorithm** Given data: $x_1, \dots, x_n$, Find centers: $c_1, \dots, c_m$

indices $\leftarrow \{1, 2, \dots, n\}$<br>
**for** $i \in 1, \dots, m$ **do**<br>
    $j \leftarrow$ randomEntry(indices)<br>
    $c_i \leftarrow x_j$<br>
    indices.remove($j$)<br>
**end for**<br>
**repeat**<br>
    **for** $i \in 1, \dots, n$ **do**<br>
        clusterAssignment[$i$]$\leftarrow \arg\min_j \|x_i - c_j\|$<br>
    **end for**<br>
    **for** $i \in 1, \dots, m$ **do**<br>
        $c_i \leftarrow$ average($\{x_j \mid$ clusterAssignment[$j$] $= i\}$)<br>
    **end for**<br>
    **for** $i \in 1, \dots, m$ **do**<br>
        $c'_i \leftarrow c_i$<br>
    **end for**<br>
**until** $c_i = c'_i$ for all $i$<br>
**return** $c_1, \dots, c_m$

b) Precisely describe how to fix the algorithm to correctly implement k-means.

**Problem 7 (Neural Networks)** **3 + 4 + 4 = 11 Points**

a) *Note: This question was not phrased properly and is thus ambiguous.*

Once properly trained, an artificial neural network can be represented by a function $f : \mathbb{R}^d \to \mathbb{R}$ that can compute for each input $x \in \mathbb{R}^d$ a function value $f(x)$.

☐ True

☐ False

b) Describe what the "learning rate" is in the context of neural network training. How should the learning rate change during training and why?

c) Describe in two or three sentences how backpropagation works.

---