Problem 1.

Consider the following set of 6 data points:

$$\vec{x}^{(1)} = (2, 4, 4)^T \tag{1}$$
$$\vec{x}^{(2)} = (-1, 2, 1)^T \tag{2}$$
$$\vec{x}^{(3)} = (3, -3, 2)^T \tag{3}$$
$$\vec{x}^{(4)} = (0, -3, -3)^T \tag{4}$$
$$\tag{5}$$

In the below parts, your answers should be given as numbers. You may leave your answer as an unsimplified fraction or a decimal, if you prefer.

a) What is the (1,3) entry of the sample covariance matrix?

9/4

b) What is the (1,2) entry of the sample covariance matrix?

-3/4

Problem 2.

Recall that the Gaussian Radial Basis Function (RBF) kernel is defined as:

$$\kappa(\vec{x}, \vec{x}') = \exp\left(-\gamma \|\vec{x} - \vec{x}'\|^2\right).$$

Suppose a kernel SVM is trained on a data set with the Gaussian RBF kernel, and a training accuracy of 80% is achieved. If you wish to increase the training accuracy, what should be done to the value of γ?

- ● Increase the value of γ.
- ○ Decrease the value of γ.
- ○ Keep the value of γ the same, as it will not have an effect on the training accuracy.

Solution: First, recall that γ controls the width of the Gaussian: the larger γ is, the *narrower* the Gaussian becomes.

Making the Gaussians narrower will mean that the prediction function H is more "spiky"; that is, we have greater control over the local value of H near each training point. Therefore, we can better fit (even overfit) the training data, which will increase the training accuracy. This is like what we saw in the last slides of Lecture 10.

To put it another way, increasing γ will make the decision boundary more complex, which can help the model better fit the training data.

Problem 3.

You and your friend are both training regularized least squares models on the same training data set of 50 points. You'd like to check if your models are the same by comparing the weights you've learned, but you're using different machine learning libraries that do things slightly differently.

The documentation of the library that your friend is using says that the regularization parameter is λ, and that the library minimizes the regularized risk:

$$\frac{1}{n}\sum_{i=1}^{n}(\vec{w}\cdot\vec{x}^{(i)}-y_i)^2+\lambda\|\vec{w}\|^2.$$

On the other hand, your library uses a regularization parameter of C and minimizes the regularized risk:

$$C\sum_{i=1}^{n}(\vec{w}\cdot\vec{x}^{(i)}-y_i)^2+\|\vec{w}\|^2.$$

If your friend is using a regularization parameter of $\lambda = 3$, what value of C should you use to ensure that your models are equivalent (that is, to get the same weights)?

- ◯ $C = 3$
- ◯ $C = 1/3$
- ◯ $C = 150$
- ● $C = 1/150$
- ◯ It is not possible to guarantee that the models will be equivalent with any value of C.

Solution: There are two ways of arriving at the correct answer.

First, remember that if α is a positive constant, then the minimizer of $f(\vec{w})$ and the minimizer of $\alpha f(\vec{w})$ are the same. In other words, if we scale the regularized risk by a positive number α and solve, we get the same model.

We can think of our library's regularized risk as a scaled version of our friend's library's regularized risk. Their library uses a factor of 3 on the regularization term, while our library uses a factor of 1, meaning that our risk is 1/3 times theirs. So, if their library uses a factor of $1/n$ on the risk, ours should use $(1/3)(1/n) = 1/(3 \times 50) = 1/150$.

Here's a second approach. In the library that our friend is using, λ controls the tradeoff between the regularization term and the risk term. The factor controlling the importance of the risk is $1/n$, and the factor controlling the importance of the regularization term is λ. But what is important is the ratio of these two factors; in your friend's case, the ratio is:

$$\frac{\lambda}{1/n}=\frac{3}{1/50}=150.$$

In our library, the factor controlling the importance of the risk is C, and the factor controlling the importance of the regularization term is simply 1. But we can get the same model as our friend by ensuring that the ratio of these factors is the same. That is:

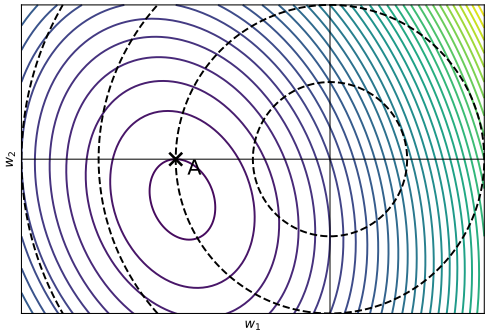
$$\frac{1}{C}=150 \implies C=\frac{1}{150}.$$

Problem 4.

The 2-norm of a vector $\vec{w} \in \mathbb{R}^d$ is defined as:

$$\|\vec{w}\|_2 = \sqrt{\sum_{i=1}^{d} w_i^2}.$$

Let $R(\vec{w})$ be the *un*regularized empirical risk with respect to a data set. The solid curves below are the contours of $R(\vec{w})$. The dashed lines show where $\|\vec{w}\|_2$ is equal to 1, 2, 3, and so on.



Let $\tilde{R}(\vec{w}) = R(\vec{w}) + \lambda\|\vec{w}\|_2^2$ be the regularized empirical risk, with $\lambda > 0$.

True or False: there is a choice of λ so that the point marked A is a minimizer of the regularized empirical risk.

◯ True

● False

Solution: The regularized risk has two components: the risk and the regularization penalty. All points on the same dashed circle as A have the same regularization penalty, but different risks.

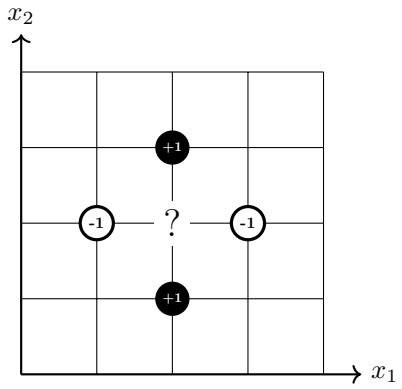
If you put your finger on point A and move it *down* along the dashed circle, you'll *decrease* the risk, while keeping the regularization penalty constant; in other words, you'll be decreasing the regularized risk. This means that A could not be a minimizer of the regularized risk: there are apparently other points on the dashed circle that are better.

Problem 5.

Consider the following data set of four points whose feature vectors are in $\mathbb{R}^2$ and whose labels are in $\{-1, 1\}$:

i	1	2	3	4
$\vec{x}^{(i)}$	(2,1)	(2,3)	(1,2)	(3,2)
y_i	1	1	-1	-1

For convenience, we've plotted the data below.



Suppose an unnamed kernel classifier $H(\vec{x}) = \sum_{i=1}^{n} \alpha_i \kappa(\vec{x}^{(i)}, \vec{x})$ has been trained on this data using a (spherical) Gaussian kernel and kernel width parameter $\gamma = 1$. Suppose the solution to the dual problem is found to be $\vec{\alpha} = (0, 4, -1, -2)^T$.

What class will the classifier predict for the point $(2, 2)$? For convenience, we've plotted this point on the graph above as a question mark.

● +1

◯ -1

Solution: The prediction at ? is influenced by the four training points shown. This is captured by the formula:

$$\begin{aligned} H(\vec{x}) &= \sum_{i=1}^{4} \alpha_i \kappa(\vec{x}^{(i)}, \vec{x}) \\ &= \alpha_1 \kappa(\vec{x}^{(1)}, \vec{x}) + \alpha_2 \kappa(\vec{x}^{(2)}, \vec{x}) + \alpha_3 \kappa(\vec{x}^{(3)}, \vec{x}) + \alpha_4 \kappa(\vec{x}^{(4)}, \vec{x}) \\ &= 0 \cdot \kappa(\vec{x}^{(1)}, \vec{x}) + 4 \cdot \kappa(\vec{x}^{(2)}, \vec{x}) - 1 \cdot \kappa(\vec{x}^{(3)}, \vec{x}) - 2 \cdot \kappa(\vec{x}^{(4)}, \vec{x}) \\ &= 0 + 4 \cdot \kappa(\vec{x}^{(2)}, \vec{x}) - 1 \cdot \kappa(\vec{x}^{(3)}, \vec{x}) - 2 \cdot \kappa(\vec{x}^{(4)}, \vec{x}) \end{aligned}$$

Now, κ is the Gaussian kernel, and it's a pain to evaluate it by hand (or even with a calculator). But we don't need to evaluate it: we just need to remember that the Gaussian kernel measures the similarity between two points based on only the distance between them. Crucially, all four points are the same distance away from the new point, so the kernel values will be the same for all of them. Let's call this value c; it will be a positive number. Then we have:

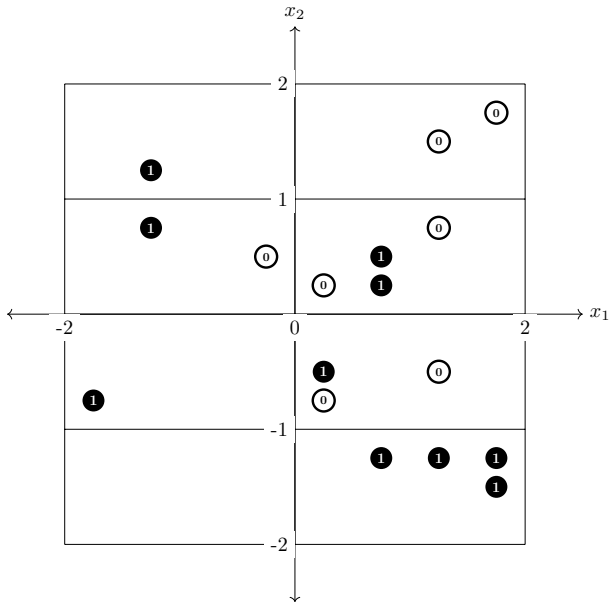
$$\begin{aligned} &= 4c - c - 2c \\ &= c. \end{aligned}$$

Since c is positive, H at the new point will be positive. Therefore, the classifier will predict +1.

To put it more informally: each of the four training points influences the value of H at the new point. Because each of the four points is the same distance from the new point, their influence is totally determined by their corresponding α_i. And since the α_i's add up to a positive number, the classifier will predict +1.

Problem 6. (2 points)

For this problem, consider the binary classification training set shown below. The filled points have label 1 and the empty points have label 0. Suppose a histogram density estimator is used to estimate the class-conditional densities. The grid shown in the plot depicts the histogram bins; each bin has a width of 2 along the x_1 axis and height 1 along the x_2 axis.



a) What is the estimated density $p(x_1, x_2)$ at the point $(1.5, 1.5)$? You may leave your answer as a fraction or convert it to a decimal.

1/17

Solution: The estimated density has the formula:

$$\frac{\#\text{ of points in bin}}{\text{total }\#\text{ of points} \times \text{bin area}}$$

In this case, there are 2 points in the bin containing $(1.5, 1.5)$, and there are 17 points in total. The area of the bin is $2 \times 1 = 2$. So the density estimate is:

$$\frac{2}{17 \times 2} = \frac{1}{17}$$

b) What is the estimated density $p_1(x_1, x_2 \,|\, Y = 1)$ at the point $(0.5, 0.5)$?

1/10

Solution: Now that we're conditioning on $Y = 1$, we only consider the black points with label 1. Otherwise, we use the same formula as in the previous part. There are 2 black points in the bin containing $(0.5, 0.5)$, and there are 10 black points in total. The area of the bin is $2 \times 1 = 2$.

So the density estimate is:

$$\frac{2}{10 \times 2} = \frac{1}{10}$$

c) Using this histogram density estimator, what is the estimate of $\mathbb{P}(Y = 1 \,|\, x_1 = 0.5, x_2 = 0.5)$?

1/2

Solution: Because we're using a histogram estimator, there is a "shortcut" to computing this: we find the bin containing $(0.5, 0.5)$ and use the ratio of black points to total points in that bin. In this bin, there are 2 black points and 4 points total. So the estimate is $2/4 = 1/2$.

There's a second (harder) way to do this: via Bayes' rule. We have:

$$\mathbb{P}(Y = 1 \,|\, x_1 = 0.5, x_2 = 0.5) = \frac{\mathbb{P}(x_1 = 0.5, x_2 = 0.5 \,|\, Y = 1)\mathbb{P}(Y = 1)}{\mathbb{P}(x_1 = 0.5, x_2 = 0.5)}$$

$\mathbb{P}(Y = 1)$ is easy to estimate: it is the number of black points divided by the total number of points, or $10/17$.

$p_1(0.5, 0.5 \,|\, Y = 1)$ is the density estimate we computed in part (b), and is $1/10$.

The denominator is the total density estimate at $(0.5, 0.5)$, which hasn't been computed yet. It is:

$$\frac{\#\text{ of points in bin}}{\text{total }\#\text{ of points} \times \text{bin area}} = \frac{4}{17 \times 2} = \frac{2}{17}$$

Putting all of these pieces together into Bayes' rule, we get:

$$\begin{aligned}
\mathbb{P}(Y = 1 \,|\, x_1 = 0.5, x_2 = 0.5) &= \frac{\mathbb{P}(x_1 = 0.5, x_2 = 0.5 \,|\, Y = 1)\mathbb{P}(Y = 1)}{\mathbb{P}(x_1 = 0.5, x_2 = 0.5)} \\
&= \frac{1/10 \times 10/17}{2/17} \\
&= \frac{\frac{1}{17}}{\frac{2}{17}} \\
&= \frac{1}{2}
\end{aligned}$$

d) Suppose the Bayes classification rule is used with the estimated class-conditional densities in place of the true densities to predict the label of the point $(1, -.5)$. What is the predicted label?

- ○ 1
- ● 0

Solution: We estimate $\mathbb{P}(Y = 1 \,|\, x_1 = 1, x_2 = -0.5)$. Since we're using histograms, we estimate that $\mathbb{P}(Y = 1 \,|\, x_1 = 1, x_2 = -0.5) = 1/3$, since one out of three points in the bin containing $(1, -0.5)$ is black. Since this probability is less than $1/2$, we should predict for the other class (Class 0).

Problem 7.

Consider the below data set collecting information on a set of 10 customers at a gym.

Personal Trainer?	Workout Type	Payment Method	Retained?
Yes	Yoga	Monthly	No
No	Cardio	Monthly	No
Yes	Cardio	Yearly	No
Yes	Strength	Monthly	No
No	Strength	Yearly	No
No	Cardio	Monthly	No
No	Yoga	Yearly	No
No	Yoga	Monthly	No
No	Strength	Yearly	Yes
Yes	Cardio	Yearly	Yes

Suppose we wish to predict whether a new customer will be retained. What does a Naïve Bayes classifier predict if the customer:

- Does not have a personal trainer,
- Does cardio workouts,
- Pays monthly?

◯ Retained

● Not Retained

Solution: Naïve Bayes wants us to compute two estimates:

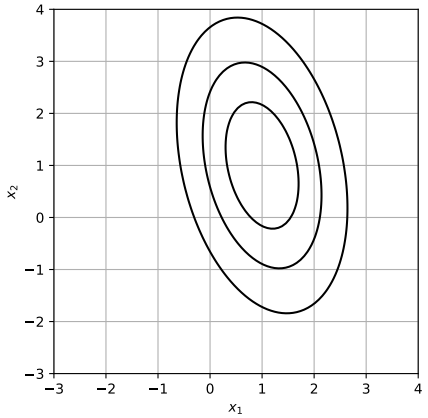
$$\overbrace{\mathbb{P}(\text{No PT} \mid \text{Retained})}^{1/2}\overbrace{\mathbb{P}(\text{Cardio} \mid \text{Retained})}^{1/2}\overbrace{\mathbb{P}(\text{Monthly} \mid \text{Retained})}^{0}\overbrace{\mathbb{P}(\text{Retained})}^{2/10} = 0$$

$$\underbrace{\mathbb{P}(\text{No PT} \mid \text{Not Retained})}_{5/8}\underbrace{\mathbb{P}(\text{Cardio} \mid \text{Not Retained})}_{3/8}\underbrace{\mathbb{P}(\text{Monthly} \mid \text{Not Retained})}_{5/8}\underbrace{\mathbb{P}(\text{Not Retained})}_{8/10} > 0$$

Since the second estimate is greater than the first, the Naïve Bayes classifier predicts that the customer will not be retained.

Problem 8.

The plot below shows the contours of a multivariate Gaussian density:



Which of the following could be the Gaussian's covariance matrix?

- ● $\begin{pmatrix} 1 & -\frac{1}{2} \\ -\frac{1}{2} & 3 \end{pmatrix}$
- ○ $\begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}$
- ○ $\begin{pmatrix} 3 & \frac{1}{2} \\ \frac{1}{2} & 1 \end{pmatrix}$
- ○ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Solution: This is not a spherical or axis-aligned Gaussian, so it's covariance matrix should be "full", i.e., have non-zero off-diagonal elements. The first and third options fit this description. There are two ways to decide between them.

First, note that the Gaussian is taller than it is wide, so the covariance matrix should have a larger entry corresponding to x_2-axis than the x_1-axis. This rules out the third option, leaving the first option as the only possibility.

The second way to decide is to note that the covariance between x_1 and x_2 should be negative, since the "trend" is that as x_1 increases, x_2 decreases. This rules out the third option as well.

Problem 9.

Why is it acceptable for AdaBoost to use weak learners, such as decision stumps, instead of strong learners?

- ○ Because each weak learner is expected to classify every training point correctly.
- ● Because weak learners are simple and fast, and AdaBoost combines many of them into a stronger final classifier.
- ○ Because weak learners ignore difficult points, which prevents overfitting.
- ○ Because weak learners always generalize better than strong learners individually.

Solution: AdaBoost's strength comes from combining many simple classifiers, not from making each individual classifier powerful.

Problem 10. (3 points)

Suppose a discrete random variable X takes on values of either 0 or 1 and has the distribution:

$$\mathbb{P}(X = x) = \theta^x (1-\theta)^{1-x}$$

where $\theta \in [0, 1]$ is a parameter.

Given a data set $x_1, \ldots, x_n$, what is the maximum likelihood estimate for the parameter θ? Show your work.

Solution: The maximum likelihood estimate for θ is the mean of all data points: $\theta = \frac{1}{n}\sum_{i=1}^{n} x_i$ We can derive it as follows. First, set up the likelihood function

$$L(\theta) = \prod_{i=1}^{n} \theta^{x_i} (1-\theta)^{1-x_i}$$

Then, to make our lives easier, we can find the log-likelihood before differentiating:

$$\ell(\theta) = \log L(\theta) = \sum_{i=1}^{n} x_i \log\theta + \sum_{i=1}^{n}(1-x_i)\log(1-\theta)$$

$$\frac{\partial \ell(\theta)}{\partial \theta} = \frac{\sum_{i=1}^{n} x_i}{\theta} - \frac{\sum_{i=1}^{n}(1-x_i)}{1-\theta}$$

$$\text{Setting } \frac{\partial \ell(\theta)}{\partial \theta} = 0 \quad \Longrightarrow \quad \frac{\sum_{i=1}^{n} x_i}{\theta} - \frac{\sum_{i=1}^{n}(1-x_i)}{1-\theta} = 0$$

$$\theta \sum_{i=1}^{n}(1-x_i) = (1-\theta)\sum_{i=1}^{n} x_i.$$

$$\text{and hence} \quad \theta = \frac{1}{n}\sum_{i=1}^{n} x_i.$$

Problem 11.

A decision tree is being trained using Gini uncertainty. At the root node, two possible splits are being considered.

Split A produces one perfectly pure child node and one very mixed child node.

Split B produces two moderately pure child nodes.

Which statement is most accurate?

- ◯ Split A must be better because producing a pure child node is always the goal.
- ● Split B may be better because the tree evaluates the weighted uncertainty of both children, not just the purity of the best child.
- ◯ Split A and Split B are equally good according to Gini if they both reduce training error.
- ◯ The tree should choose whichever split uses the feature with the largest numerical values.

Solution: A split is judged by the total weighted uncertainty after splitting. Creating one pure node is not enough if most of the data ends up in a large, highly uncertain node. The decision tree is not greedily optimizing the best child; it is greedily optimizing the combined quality of the children.

Problem 12.

True or False: Using a zero-mean Gaussian prior on the weights in linear regression leads to an L1 regularization penalty.

- ◯ True
- ● False

Solution: It leads to a L2 regularization penalty.

Problem 13.

A classifier achieves 5% test error on a test set of 40 samples.

True or False: The Bayes error for the distribution must be less than or equal to 5

◯ True

● False

Solution: Bayes error is optimal in expectation, i.e. for the population. We cannot say anything about the Bayes error on the distribution given a finite test error, and we cannot say anything about finite test error given a Bayes error on the distribution.

Problem 14.

Suppose data are generated from a distribution with two well-separated clusters, but we fit a single Gaussian distribution using maximum likelihood.

As the number of samples grows, what should we expect?

◯ The fitted Gaussian must recover the true density exactly.

● The fitted Gaussian will converge to the best single-Gaussian approximation

◯ The fitted Gaussian will eventually split into two Gaussians.

◯ The fitted Gaussian will have zero variance.

Solution: Using a Gaussian to model a bimodal distribution will fail to recover the true density exactly, showcasing some of the troubles of parametric estimation. Using MLE, we will converge to the best possible single-Gauss approximation of the true distributions, however.