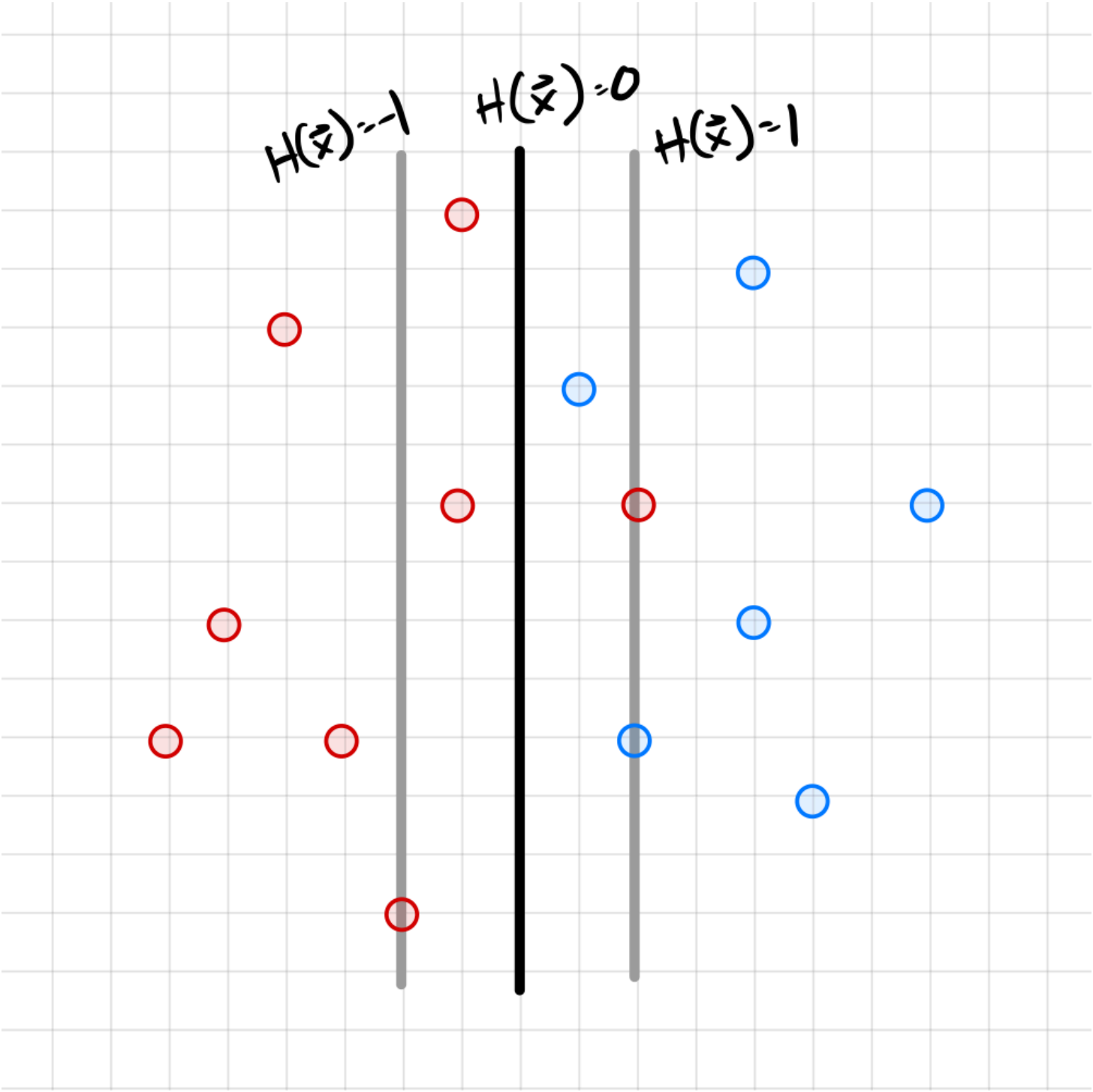
Consider the linear classifier depicted below. Shown is its decision boundary, as well as the lines where the output of the linear prediction function is 1 and -1. You can assume that the data and lines are placed precisely on the grid; this can be used to calculate distances. You may assume that the red points have correct label of -1, while the blue points have a correct label of +1.



What is the empirical risk of this classifier with respect to the hinge loss? Report your answer as a decimal number with two decimal places of precision.

The hinge loss will only penalize points that are 1) misclassified or 2) too close to the decision boundary (|H(x)| < 1). In this case this is only four points: the two red points immediately to the left of the decision boundary (they are too close to the boundary), the blue point closest to the decision boundary (also too close), and the red point that has been misclassified.

First, what is the loss of the two red points to the left of the boundary? On these two points, $H = -1/2$. Therefore, the loss is $1 - (-1)(-1/2) = 1/2$.

The blue point to the right of the boundary incurs a loss of 1/2 as well, since $H = 1/2$ on this point, and the hinge loss is therefore $1 - (1)(1/2) = 1/2$.

The misclassified red point incurs a loss of $1 - (-1)(1) = 2$, since $H(x) = 1$ for that point.

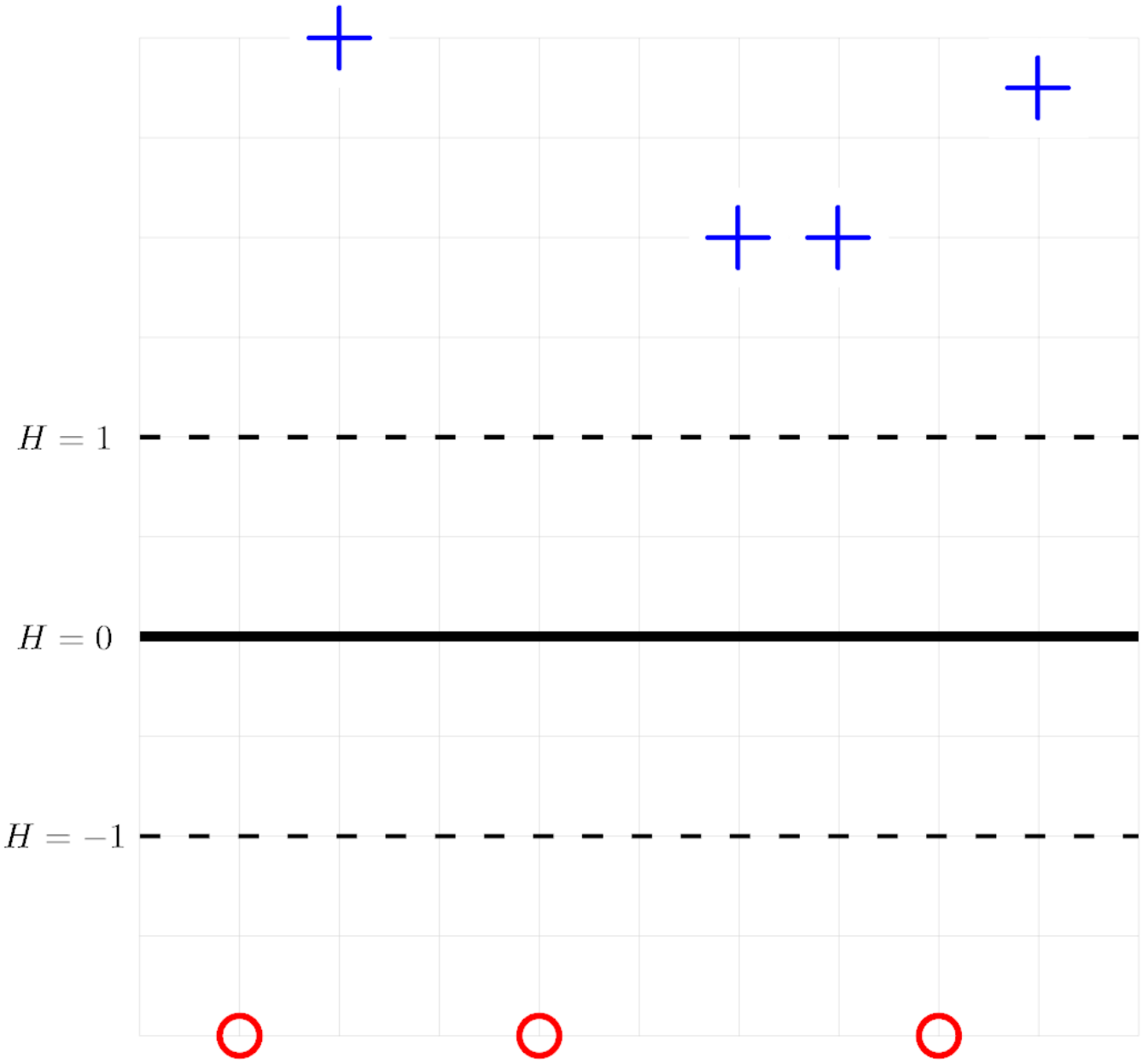
The total loss is therefore $2(1/2) + (1/2) + 2 = 3.5$. Since there are 14 points in total, the risk is 3.5 / 14 = 0.25

True or False: a Soft-SVM trained on a linearly-separable training data set must achieve 100% training accuracy (it will correctly predict the label of all training points).

( ) True
(X) False

The key thing to note here is that the statement says “must achieve 100% training accuracy”. This is in fact not the case; e.g. in Lecture 07 Slides 33 and 34, we see an example of a linearly-separable data set that does not achieve 100% training accuracy with a smaller C because we allow for more slack compared to a larger C.

Consider the data set shown below. The points marked with "+" have label 1, while the "o" points have label -1. Shown are the places where a linear prediction function H is equal to zero, 1, and -1. Each cell of the grid is 1 unit by 1 unit. The origin is not plotted (it isn't necessary to know where it is to solve this problem).



Suppose that the weight vector $\vec{w}$ of the linear prediction function $w_0 + w_1 x_1 + w_2 x_2$ shown above is $(-2, 0, \frac{1}{2})^T$. This is not a solution to the Hard-SVM problem, but which of the below vectors *is*?

() $(-2, 0, 1)^T$
(x) $(-1, 0, \frac{1}{4})^T$
() $(0, 1, 0)^T$
() $(-2, 0, \frac{1}{2})^T$

Although this decision boundary is in the right place, it can't be the solution to the Hard-SVM problem because its margin isn't maximized. Remember that the surface of H is a plane, and in this case the plane is too steep; we need to decrease its slope. We do this by scaling the weight vector by

a constant factor; in this case, we want to double the margin, so we need to halve the slope. Therefore, the right answer is $\frac{1}{2}(-2, 0, \frac{1}{2})^T = (-1, 0, \frac{1}{4})^T$.

In a binary classification problem, the data are said to be **linearly separable** if there exists a linear prediction function which classifies every point correctly. Geometrically, this means that there exists a linear decision boundary (i.e., a line, plane, or hyperplane, depending on the dimensionality of the data) which neatly separates all of the points -- all from class +1 are on one side of the boundary, while all points from class -1 are on the other side.

Suppose a data set is linearly separable. True or False: there must be a linear prediction function $H(\vec{x}) = w_0 + w_1 x_2 + \ldots w_d x_d$ which incurs a **mean square loss** of zero on this data set.

( ) True
(x) False

A prediction function can only achieve a mean square loss of zero if $H(x) = 1$ for every point from class 1 and $H(x) = -1$ for every point from class -1. But our linear prediction rule outputs real numbers, like 0.8 or -1.3, which in general will be slightly different from -1 and 1. If this is the case for even a single point, the mean square loss cannot be zero.

Consider a dataset where input points $\vec{x} \in \mathbb{R}^3$ are represented as (x_1, x_2, x_3). Given four basis functions: $\varphi_1, \varphi_2, \varphi_3, \varphi_4$ defined as:

- $\varphi_1(\vec{x}) = x_1 x_2$
- $\varphi_2(\vec{x}) = x_2^2$
- $\varphi_3(\vec{x}) = x_3^2 x_1$
- $\varphi_4(\vec{x}) = x_1 x_2 x_3$

The basis functions map the original data point $\vec{x}$ from $\mathbb{R}^3 \rightarrow \mathbb{R}^4$. What will be the representation of a data point $\vec{x} = (3, 2, -1)$ in the new space after applying the basis functions.

Report the **sum of features** as your final answer. If the new representation is given as (a,b,c,d), you should report a + b + c + d as the final answer.

$\varphi_1(\vec{x}) = 6$, $\varphi_2(\vec{x}) = 4$, $\varphi_3(\vec{x}) = 3$, $\varphi_4(\vec{x}) = -6$. Hence, the sum of these features is 6 + 4 + 3 - 6 = 7

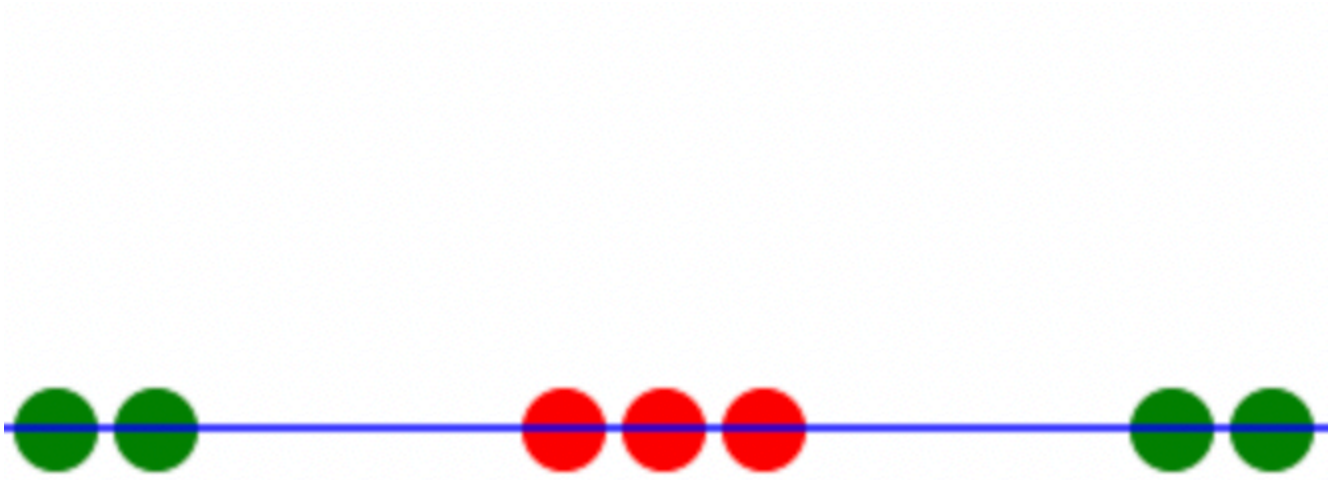
Consider the same description as above, we learn a linear classifier in the feature space ($\mathbb{R}^4$) based on the training data, and the learned weights of the hyperplane is $\vec{w} = (0.4, 0.3, -0.6, 1.3, 0.7)$. What is the value of the prediction function (H) for the input point $\vec{x} = (3, 2, -1)$? Note that the input point is given in the original $\mathbb{R}^3$ space. Your answer should be a number.

Prediction function is given by $H(\vec{x}) = \vec{w} \cdot Aug(\vec{\phi}(\vec{x}))$. From the previous problem, the mapping of $\vec{x}$ in $\mathbb{R}^4$ is (6, 4, 3, -6). Hence, $H(\vec{x}) = 0.4 + 0.3 * 6 - 0.6 * 4 + 1.3 * 3 - 0.7 * 6 = -0.5$

Given data points x on a number line.

- The green points refer to label 1. The points are $x = -6, -5, 5, 6$
- The red points refer to label -1. The points are $x = -1, 0, 1$

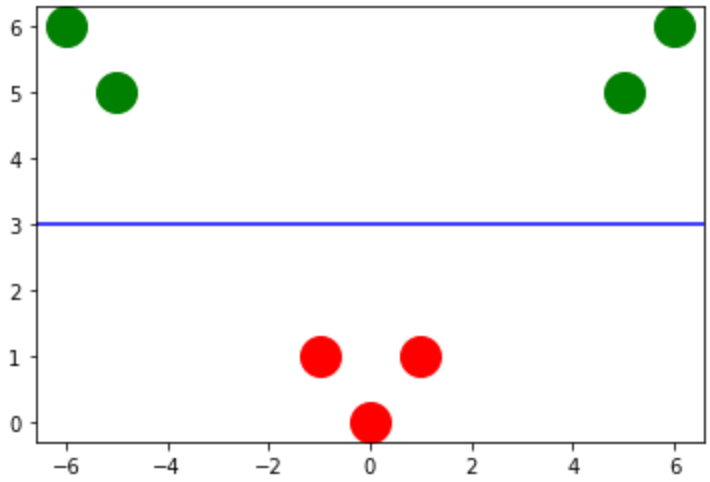
The points might look something like this.



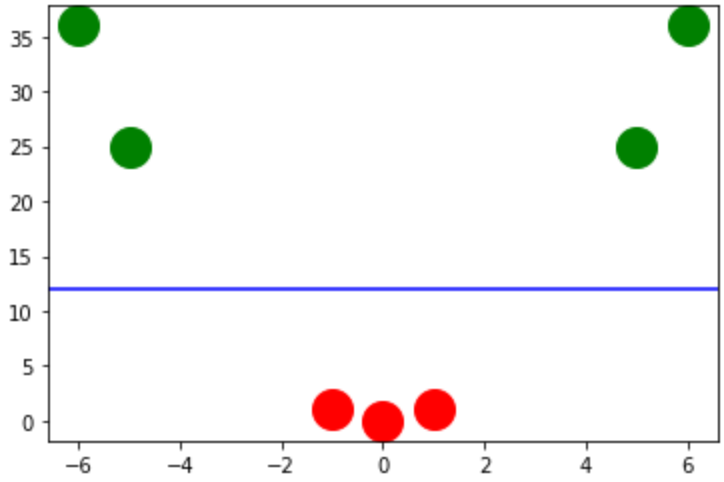
It is evident that we can't learn a linear classifier to separate these points in 1 dimension. The data is not linearly separable. However, it is possible to perform feature mapping and represent these points in $\mathbb{R}^2$ where the data becomes linear separable. Select all those possible transformations which make the data linearly separable in R^2. **More than one option** might be selected.

[ ] $x \rightarrow (x, x^3)$
[x] $x \rightarrow (x, x^2)$
[x] $x \rightarrow (x, |x|)$
[ ] $x \rightarrow (x, x)$

You can plot the points in $\mathbb{R}^2$ after applying the feature transformation.
$x \rightarrow (x, |x|)$ is linearly separable by the blue line.



$x \rightarrow (x, x^2)$ is linearly separable by the blue line.



Other transformations are not able to separate the points as desired. It can be observed by making similar plots.