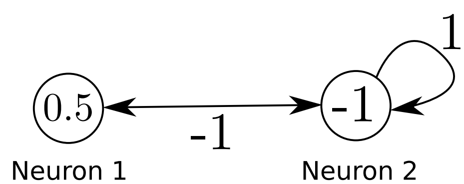
ECE/CS 559 - Fall 2016 - Final Exam.

Full Name: **ID Number:**

- **Q0 (5 pts):** Attach the e-mail/webpage confirming that you completed the instructor/TA evaluations.

- **Q1 (31 pts):** Consider the Hopfield network below. The activation function is $\phi(x) = 1$ if $x \geq 0$, and $\phi(x) = -1$ if $x < 0$.



  (a) **(9 pts):** Draw the state transition diagram together with state energy levels for the asynchronous update rule. Indicate the steady state(s) of the network.

  (b) **(7 pts):** Does the network always converge to a steady state under the asynchronous update rule? Justify your answer.

  (c) **(15 pts):** Repeat (a) and (b) for the synchronous update rule.

(a) Let $W = \begin{bmatrix} 0 & -1 \\ -1 & 1 \end{bmatrix}$, $b = \begin{bmatrix} 0.5 \\ -1 \end{bmatrix}$, and $E(x) \triangleq -x^T W x - 2b^T x$. We calculate

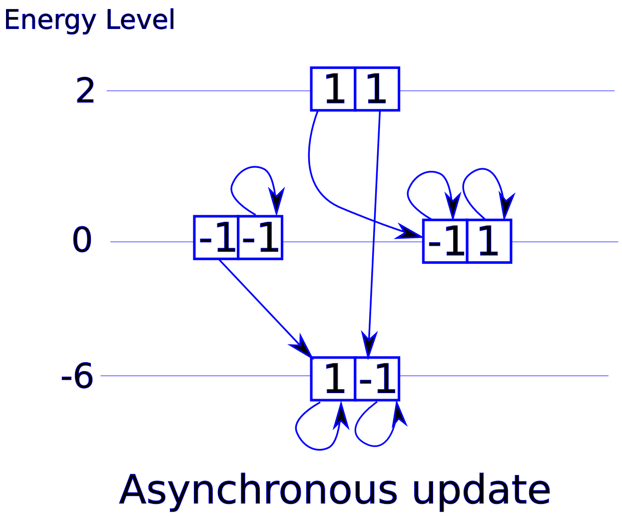
$$\phi\left(W \begin{bmatrix} -1 \\ -1 \end{bmatrix} + b\right) = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \text{ with } E\left(\begin{bmatrix} -1 \\ -1 \end{bmatrix}\right) = 0.$$

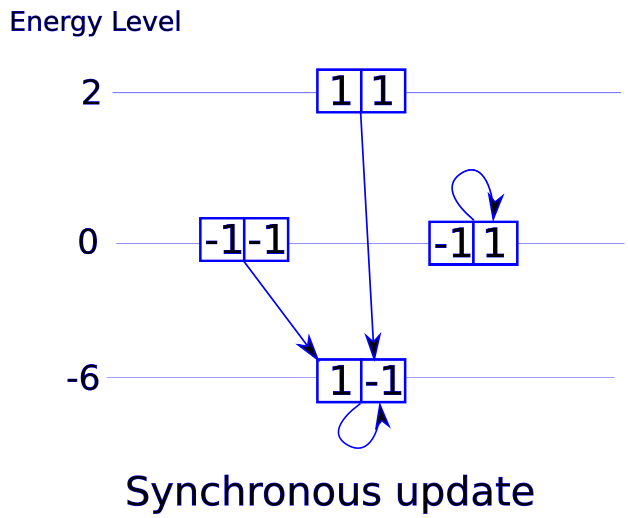
$$\phi\left(W \begin{bmatrix} -1 \\ 1 \end{bmatrix} + b\right) = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \text{ with } E\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}\right) = 0.$$

$$\phi\left(W \begin{bmatrix} 1 \\ -1 \end{bmatrix} + b\right) = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \text{ with } E\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right) = -6.$$

$$\phi\left(W \begin{bmatrix} 1 \\ 1 \end{bmatrix} + b\right) = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \text{ with } E\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}\right) = 2.$$

From these calculations, we obtain the diagrams below:





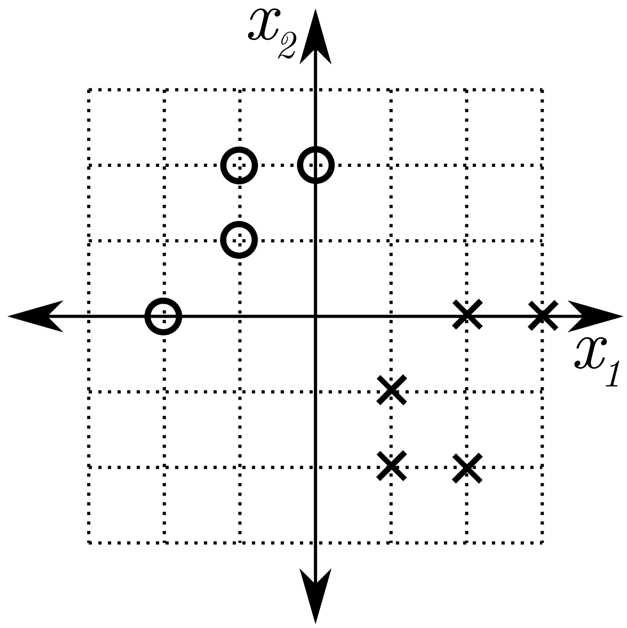
The steady states for the synchronous and asynchronous update rules are the same and are $\left[\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right]$ and $\left[\begin{smallmatrix} -1 \\ 1 \end{smallmatrix}\right]$.

(b) From the diagrams, we observe that the network converges under both synchronous and asynchronous update rules.

(c) Already solved.

- **Q2 (30 pts):** This problem will be on SVMs. **Note that neither of the two parts (a) and (b) of this question require solving complex optimization problems**.

  (a) **(15 pts):** Consider the figure below. Each small dotted square is 1×1. Members of C^+ are represented by crosses while members of C^- are represented by hollow disks. According to this description, we thus have $C^+ = \{\left[\begin{smallmatrix}2\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}3\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\-2\end{smallmatrix}\right], \left[\begin{smallmatrix}2\\-2\end{smallmatrix}\right]\}$, and $C^- = \{\left[\begin{smallmatrix}-2\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\2\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\2\end{smallmatrix}\right]\}$.

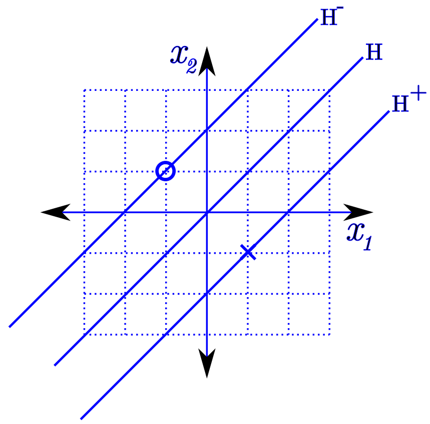


  Design a **linear SVM** that separates the classes C^+ and C^-.
  Recall that the result of the SVM will be a discriminant function $g(\mathbf{x}) = g(\left[\begin{smallmatrix}x_1\\x_2\end{smallmatrix}\right])$ with the property that $g(\mathbf{x}) \geq 1$ for every $\mathbf{x} \in C^+$, and $g(\mathbf{x}) \leq -1$ for every $\mathbf{x} \in \mathcal{C}^-$. A pattern $\mathbf{x} \in C^+$ with $g(\mathbf{x}) = 1$ is called a support vector for class C^+, while a pattern $\mathbf{x} \in C^-$ and $g(\mathbf{x}) = -1$ is called a support vector for class C^-. In your solution, you should clearly indicate

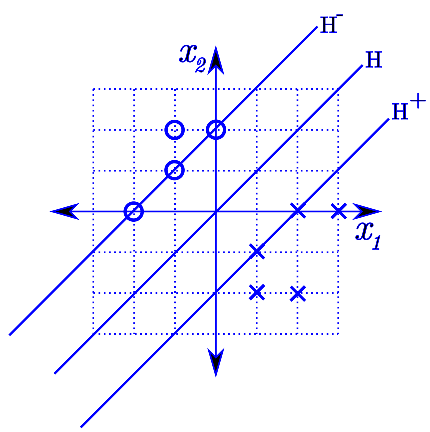
  - The steps and justifications of your solution to the corresponding SVM design problem.
  - The resulting discriminant function $g(\mathbf{x})$.
  - The support vectors for class C^+, and the support vectors for class C^-.
  - A sketch of the decision boundaries

  $$\mathcal{H} = \{\mathbf{x} : g(\mathbf{x}) = 0\},\ \mathcal{H}^+ = \{\mathbf{x} : g(\mathbf{x}) = 1\},\ \text{and } \mathcal{H}^- = \{\mathbf{x} : g(\mathbf{x}) = -1\}.$$

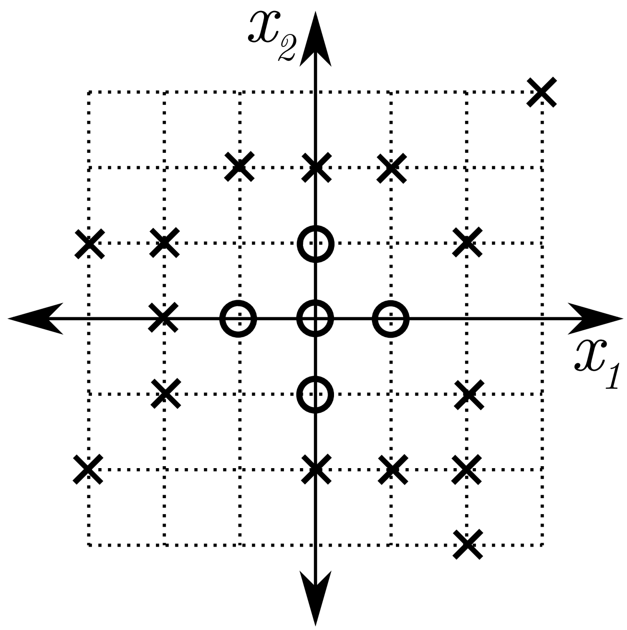
As illustrated in the figure below, consider an alternate SVM design problem where we have one member from each class as $C^+ = \{\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]\}$, and $C^- = \{\left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right]\}$. In this case, as derived in class, the best separating hyperplane (the SVM) passes between the points $\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]$ and $\left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right]$ and is perpendicular to the line that connects $\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]$ and $\left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right]$. The support vector for class C^+ is $\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]$ and the support vector for class C^- is $\left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right]$. Since $g(\mathbf{x})$ should vanish whenever $x_1 = x_2$, we have $g(\mathbf{x}) = c(x_1 - x_2)$. By $g(\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]) = 1$ (note that $\left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]$ is a support vector for C^+), we obtain $c = \frac{1}{2}$.



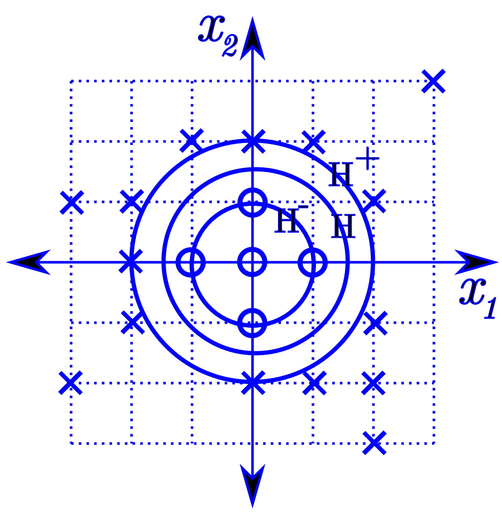
Now, in our simple problem, each pattern is at least $\sqrt{2}$-far to the separating hyperplane $\mathcal{H}$. We observe that if we use the same hyperplane in our original problem, then, again, each pattern is at least $\sqrt{2}$-far to the separating hyperplane, and each pattern will be classified correctly. There cannot be a better hyperplane for the original problem, as that would imply a better hyperplane for the simple problem as well! Hence, the discriminant function for the original problem will be $g(\mathbf{x}) = \frac{1}{2}(x_1 - x_2)$. The sets $\mathcal{H}^+$, $\mathcal{H}$, and $\mathcal{H}^-$ will be the same as in the figure above. The support vectors for class C^+ are $\{\left[\begin{smallmatrix}2\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]\}$. Support vectors for class C^- are $\{\left[\begin{smallmatrix}2\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\-1\end{smallmatrix}\right]\}$. Support vectors for class C^- are $\{\left[\begin{smallmatrix}-2\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\2\end{smallmatrix}\right]\}$.



(b) **(15 pts):** Repeat (a) for classes C^+ and C^- illustrated in the figure below. This time, instead of a linear SVM, you have to design **a non-linear SVM** by picking an appropriate feature mapping/kernel. **Your feature mapping $\phi(\mathbf{x})$ should depend only on the norm $\|\mathbf{x}\|$ of x.** For example, $\phi(\mathbf{x}) = \begin{bmatrix} \|\mathbf{x}\|+1 \\ \|\mathbf{x}\|^2 \end{bmatrix}$ would be a valid feature mapping for this question.



Choose, e.g. $\phi(\mathbf{x}) = \|\mathbf{x}\|$. Samples in the feature space are then the scalars $\phi(C^+) = \{2, \sqrt{5}, \sqrt{8}, \sqrt{10}, \sqrt{13}, \sqrt{18}\}$ and $\phi(C^-) = \{0, 1\}$. Using the same arguments as in (a), the discriminant function in the feature space is $g'(x) = 2x - 3$. The discriminant function in the original space is then $g(\mathbf{x}) = 2\|\mathbf{x}\| - 3$. The support vectors and the separating hyperplanes are as shown below.



- **Q4 (34 pts):** Let $\mathbf{x}_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}$, $\mathbf{x}_2 = \begin{bmatrix} 2 & 1 \end{bmatrix}$, $\mathbf{x}_3 = \begin{bmatrix} -1 & -1 \end{bmatrix}$. We wish to design a one-dimensional SOM with 2 neurons. Let $\mathbf{w}_{1,0} = \begin{bmatrix} -1 & 0 \end{bmatrix}$ and $\mathbf{w}_{2,0} = \begin{bmatrix} 0 & 1 \end{bmatrix}$ be the initial weights of the first neuron and the second neuron, respectively.

  We recall the standard online learning phase. The patterns are shown sequentially as

  $$\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots$$

  resulting in the sequence of weights

  $$\mathbf{w}_{1,0} = [-1 \quad 0], \mathbf{w}_{1,1}, \mathbf{w}_{1,2}, \mathbf{w}_{1,3}, \mathbf{w}_{1,4}, \mathbf{w}_{1,5}, \ldots$$

  for the first neuron, and the sequence of weights

  $$\mathbf{w}_{2,0} = [0 \quad 1], \mathbf{w}_{2,1}, \mathbf{w}_{2,2}, \mathbf{w}_{2,3}, \mathbf{w}_{2,4}, \mathbf{w}_{2,5}, \ldots$$

  for the second neuron. Thus, for $n \in \{1, 2, \ldots\}$, the vectors $\mathbf{w}_{1,n}$ and $\mathbf{w}_{2,n}$ denote the updated weights after n patterns are shown to the network. For notational convenience, we let $\mathbf{x}_n$ denote the nth pattern shown to the network. For example, $\mathbf{x}_n = \mathbf{x}_3$ whenever n is a multiple of 3.

  When the nth pattern $\mathbf{x}_n$ is shown, we define the winning neuron

  $$i_n = \arg\min_{i \in \{1,2\}} \|\mathbf{x}_n - \mathbf{w}_{i,n-1}\|, \; n \in \{1, 2, \ldots\}$$

  as the neuron whose weight vector is closest to $\mathbf{x}_n$ in terms of the Euclidean distance.

  (a) **(9 pts):** When $\mathbf{x}_n$ is shown, suppose that we update the winning neuron as

  $$\mathbf{w}_{i_n,n} = \mathbf{w}_{i_n,n-1} + \tfrac{1}{2}(\mathbf{x}_n - \mathbf{w}_{i_n,n-1}),$$

  while keeping the loser neuron weights the same (i.e. $\mathbf{w}_{i,n} = \mathbf{w}_{i,n-1}$ if $i \neq i_n$). Find the weights of both neurons after one epoch of training.

  (b) **(8 pts):** Do the limits $\lim_{n\to\infty} \mathbf{w}_{1,n}$ and $\lim_{n\to\infty} \mathbf{w}_{2,n}$ exist? In other words, do the weight vectors converge? Justify your answer.

  (c) **(9 pts):** For any real number x, let $\lfloor x \rfloor$ denote the largest integer that is less than or equal to x. For example, $\lfloor 2.1 \rfloor = 2$, $\lfloor 3 \rfloor = 3$, $\lfloor -1.1 \rfloor = -2$, $\lfloor -5 \rfloor = -5$. For a vector $\mathbf{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}$, we let $\lfloor \mathbf{x} \rfloor = \begin{bmatrix} \lfloor x_1 \rfloor & \lfloor x_2 \rfloor \end{bmatrix}$.

  Now suppose that, when $\mathbf{x}_n$ is shown, we update the winning neuron as

  $$\mathbf{w}_{i_n,n} = \left\lfloor \mathbf{w}_{i_n,n-1} + \tfrac{1}{2}(\mathbf{x}_n - \mathbf{w}_{i_n,n-1}) \right\rfloor,$$

  while keeping the loser neuron weights the same (i.e. $\mathbf{w}_{i,n} = \mathbf{w}_{i,n-1}$ if $i \neq i_n$). Find the weights of both neurons after one epoch of training.

  (d) **(8 pts):** Repeat (b) for the setup in (c).

I will use a slightly different notation.

(a) Let $\mathbf{w}_1 = \begin{bmatrix}-1 & 0\end{bmatrix}$ and $\mathbf{w}_2 = \begin{bmatrix}0 & 1\end{bmatrix}$. Show $\mathbf{x}_1 = \begin{bmatrix}1 & 1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_1\| = \sqrt{5}$ and $\|\mathbf{w}_2 - \mathbf{x}_1\| = 1$. So, the second neuron wins, and its weights are updated as $\mathbf{w}_2 \leftarrow \mathbf{w}_2 + \frac{1}{2}(\mathbf{x}_1 - \mathbf{w}_2) = \frac{1}{2}(\mathbf{x}_1 + \mathbf{w}_2) = \begin{bmatrix}0.5 & 1\end{bmatrix}$. The weight $\mathbf{w}_1$ remains the same.

Now, show $\mathbf{x}_2 = \begin{bmatrix}2 & 1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_2\| = \sqrt{10}$ and $\|\mathbf{w}_2 - \mathbf{x}_2\| = \sqrt{(2-0.5)^2 + (1-1)^2} = 1.5$ (note that we are using the updated weights). So, again, the second neuron wins, and its weights are updated as $\mathbf{w}_2 \leftarrow \frac{1}{2}(\mathbf{x}_2 + \mathbf{w}_2) = \frac{1}{2}(\begin{bmatrix}2 & 1\end{bmatrix} + \begin{bmatrix}0.5 & 1\end{bmatrix}) = \begin{bmatrix}1.25 & 1\end{bmatrix}$.

Show $\mathbf{x}_3 = \begin{bmatrix}-1 & -1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_3\| = 1$ and $\|\mathbf{w}_2 - \mathbf{x}_3\| = \sqrt{(1.25+1)^2 + (1+1)^2} > 1$. So, the first neuron wins, and its weights are updated as $\mathbf{w}_1 \leftarrow \frac{1}{2}(\mathbf{x}_3 + \mathbf{w}_1) = \begin{bmatrix}-1 & -0.5\end{bmatrix}$

Hence, after one epoch of training, the updated weights are $\mathbf{w}_{1,3} = \begin{bmatrix}-1 & -0.5\end{bmatrix}$ and $\mathbf{w}_{2,3} = \begin{bmatrix}1.25 & 1\end{bmatrix}$.

(b) First observe that in all later epochs, the second neuron will be the winner for patterns $\mathbf{x}_1$ and $\mathbf{x}_2$, and the first neuron will be the winner for pattern $\mathbf{x}_3$. It then easily follows that $\lim_{n\to\infty} \mathbf{w}_{1,n} = \mathbf{x}_3$. Consider now the weights of the second neuron. The y-component will always remain equal to 1. The x-component at the end of the first epoch was equal to 1.25. At the second epoch, when $\mathbf{x}_1$ is shown, the x-component will be updated to $\frac{1+1.25}{2} = 1.125$. Then, when $\mathbf{x}_2$ is shown, the x-component will be updated to $\frac{2+1.125}{2}$. This defines the sequence $1.25, 1.125, 1.5625, \ldots$ where we sequentially apply the functions $x \mapsto \frac{1+x}{2}$ and $x \mapsto \frac{2+x}{2}$ to the initial value 1.25. It is not difficult to show that the sequence converges to the "oscillatory limit" $\frac{5}{3}, \frac{4}{3}, \frac{5}{3}, \frac{4}{3}, \ldots$. Hence, $\lim_{n\to\infty} \mathbf{w}_{2,n}$ does not exist. Instead, $\mathbf{w}_{2,n}$ oscillates between the points $\begin{bmatrix}\frac{4}{3} & 1\end{bmatrix}$ and $\begin{bmatrix}\frac{5}{3} & 1\end{bmatrix}$.

(c) Let $\mathbf{w}_1 = \begin{bmatrix}-1 & 0\end{bmatrix}$ and $\mathbf{w}_2 = \begin{bmatrix}0 & 1\end{bmatrix}$. Show $\mathbf{x}_1 = \begin{bmatrix}1 & 1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_1\| = \sqrt{5}$ and $\|\mathbf{w}_2 - \mathbf{x}_1\| = 1$. So, the second neuron wins, and its weights are updated as $\mathbf{w}_2 \leftarrow \lfloor \mathbf{w}_2 + \frac{1}{2}(\mathbf{x}_1 - \mathbf{w}_2) \rfloor = \begin{bmatrix}0 & 1\end{bmatrix}$. The weight $\mathbf{w}_1$ remains the same.

Now, show $\mathbf{x}_2 = \begin{bmatrix}2 & 1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_2\| = \sqrt{10}$ and $\|\mathbf{w}_2 - \mathbf{x}_2\| = 2$. The second neuron wins, and its weights are updated as $\mathbf{w}_2 \leftarrow \lfloor \frac{1}{2}(\mathbf{x}_2 + \mathbf{w}_2) \rfloor = \lfloor \begin{bmatrix}1 & 1\end{bmatrix} \rfloor = \begin{bmatrix}1 & 1\end{bmatrix}$.

Show $\mathbf{x}_3 = \begin{bmatrix}-1 & -1\end{bmatrix}$ to the network. We have $\|\mathbf{w}_1 - \mathbf{x}_3\| = 1$ and $\|\mathbf{w}_2 - \mathbf{x}_3\| = \sqrt{8}$. So, the first neuron wins, and its weights are updated as $\mathbf{w}_1 \leftarrow \lfloor \frac{1}{2}(\mathbf{x}_3 + \mathbf{w}_1) \rfloor = \lfloor \begin{bmatrix}-1 & -0.5\end{bmatrix} \rfloor = \begin{bmatrix}-1 & -1\end{bmatrix}$.

Hence, after one epoch of training, the updated weights are $\mathbf{w}_{1,3} = \begin{bmatrix}-1 & -1\end{bmatrix}$ and $\mathbf{w}_{2,3} = \begin{bmatrix}1 & 1\end{bmatrix}$.

(d) It is easy to see that there will be no more updates on the weights and we have $\lim_{n\to\infty} \mathbf{w}_{1,n} = \mathbf{x}_3$ and $\lim_{n\to\infty} \mathbf{w}_{2,n} = \mathbf{x}_1$.