EECS 16A Designing Information Devices and Systems I
Fall 2016 Babak Ayazifar, Vladimir Stojanovic **Final Exam**

Exam location: 1 Pimental, Last Name: A-Ketsamanian

PRINT your student ID: ____

PRINT AND SIGN your name: ________________, ________________ ____________________
(last) (first) (signature)

PRINT your Unix account login: ee16a-_____________

PRINT your discussion section and GSI(s) (the one(s) you attend): ____

Name and SID of the person to your left: ____

Name and SID of the person to your right: ____

Name and SID of the person in front of you: ____

Name and SID of the person behind you: ____

Section 0: Pre-exam questions (2 points)

1. **What was your favorite thing about EE16A? (1 pt)**

2. **What are your plans for winter break? (1 pt)**

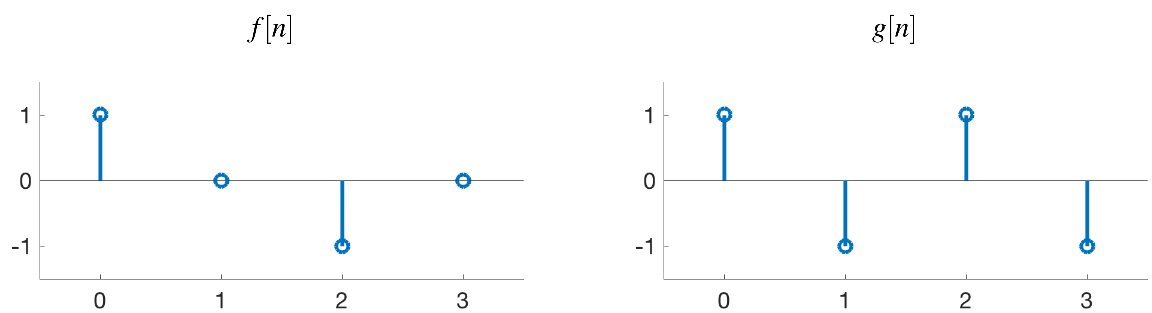
PRINT your name and student ID: ____

Section 1 (48 points)

3. Mechanical Correlation (8 points)

All cross-correlations and auto-correlations in this particular problem are circular.

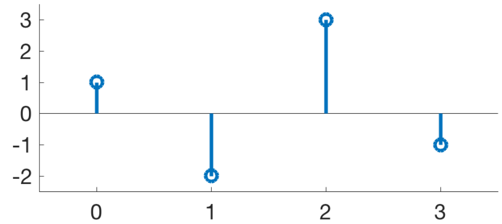
(a) (**4 points**) For the following calculation, please see the figures below for functions f and g. You may assume that both functions are periodic with a period of 4.



Find $corr(f,g)$. You may leave your answer as either a graph/plot or a vector. However, it must be clear what the precise values are.

PRINT your name and student ID: ______________________________

(b) (**4 points**) The figure below is $corr(x,y)$ for two signals x,y. Sketch $corr(y,x)$. Make sure to clearly label your axes.



4. Mechanical Gram-Schmidt (15 points)

(a) (**5 points**) Use Gram-Schmidt to find an orthonormal basis for the following three vectors.

$$\vec{v}_1 = \begin{bmatrix} \sqrt{2} \\ -\sqrt{2} \\ 0 \end{bmatrix},\ \vec{v}_2 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},\ \vec{v}_3 = \begin{bmatrix} \sqrt{2} \\ 0 \\ -\sqrt{2} \end{bmatrix}$$

(b) (**5 points**) Express $\vec{v}_1, \vec{v}_2,$ and $\vec{v}_3$ as vectors in the basis you found in part a.

(c) (**5 points**) Decompose the following matrix **A** as an orthonormal matrix **Q** and an upper-triangular matrix **R**:

$$A = \begin{bmatrix} \sqrt{2} & 0 & \sqrt{2} \\ -\sqrt{2} & 0 & 0 \\ 0 & 1 & -\sqrt{2} \end{bmatrix}$$

PRINT your name and student ID: ____

5. Eigenvalues, Eigenvectors, and Determinants (15 points)

(a) (**5 points**) Find the eigenvalues of $\mathbf{B} = \begin{bmatrix} 3 & 7 & 4 \\ 0 & 4 & 2 \\ 0 & 1 & 5 \end{bmatrix}$.

(b) (**5 points**) Assuming one of the eigenvalues is $\lambda = 3$, find its corresponding eigenvector.

(c) (**5 points**) Is the matrix $\begin{bmatrix} 3 & 7 & 4 \\ 0 & 4 & 2 \\ 0 & 1 & 5 \end{bmatrix}$ diagonalizable? Provide a succinct, but clear and convincing explanation.

6. Block Determinants (5 points)

The following properties may or may not be useful for this problem, and you do not need to prove them if you choose to use them.

$$\det \begin{bmatrix} \mathbf{A} & 0 \\ 0 & \mathbf{I} \end{bmatrix} = \det(\mathbf{A})$$

$$\det \begin{bmatrix} \mathbf{I} & \mathbf{B} \\ 0 & \mathbf{C} \end{bmatrix} = \det(\mathbf{C})$$

$$\det(\mathbf{AB}) = \det(\mathbf{A})\det(\mathbf{B})$$

where $\mathbf{I}$ represents the identity matrix, and $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are arbitrary matrices of appropriate dimension.

Determine the determinant of:

$$\mathbf{R} = \begin{bmatrix} \mathbf{P} & \mathbf{Q} \\ 0 & \mathbf{S} \end{bmatrix}$$

as a function of the determinants of sub-matrices $\mathbf{P}, \mathbf{Q}$, and $\mathbf{S}$, where $\mathbf{P}$ is invertible.

7. QR (5 points)

Recall that the solution to a linear least-squares problem is a minimization of $||\vec{b} - A\vec{x}||^2$. Show that the approximation of $\vec{x}, \vec{\hat{x}}$, in this linear least squares formula has an equivalent representation using the **A** matrix's QR decomposition ($\mathbf{A} = \mathbf{QR}$). In other words, express $\vec{\hat{x}}$ in terms of **R**, **Q**, and $\vec{b}$, and show your derivation. You may assume the matrix **A** has full column rank. Your final expression must be in the simplest form possible.

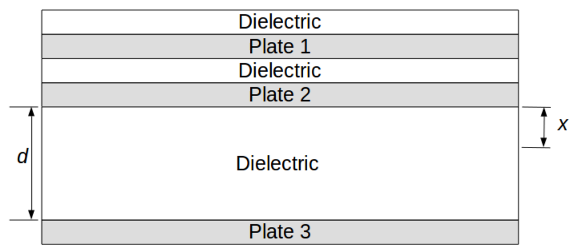
(Hint: remember that **Q** is orthonormal)

Section 2 (130 points)

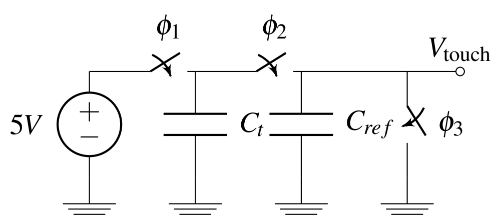
8. Force-Touch (30 points)

In this problem we will explore how to add force measurements to the basic touchscreen seen in lecture. Shown below is a cross section of this new touch screen. The distance between Plate 1 and Plate 2 is fixed. The distance between Plate 2 and Plate 3 is variable, and depends on the force applied on the screen. **Throughout this problem, we will call C_t the capacitance between plate 1 and 2, and C_f the capacitance between plates 2 and 3.**

Figure 1: Cross section

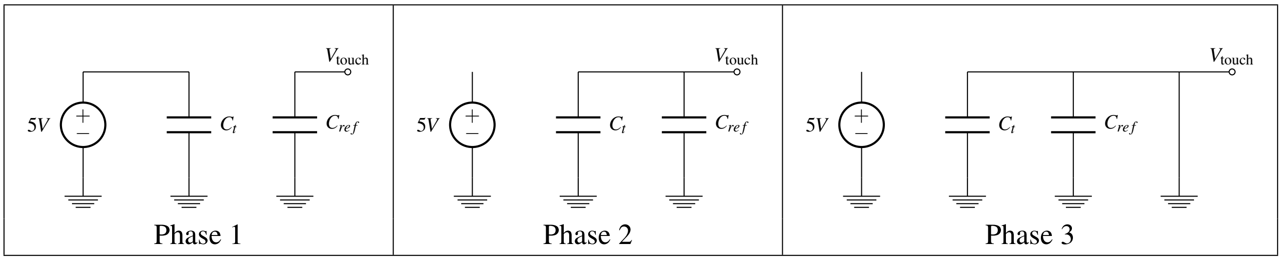


The circuit below is designed to measure a touch. Here Plate 2 is connected to ground, and C_t is the *total* capacitance, including the effect of a finger when present, between Plates 1 and 2.



PRINT your name and student ID: ____

The circuit above cycles through three phases, depicted in the diagram below. For this problem, you may assume switching is controlled through a microcontroller and all capacitors reach steady state during each phase.

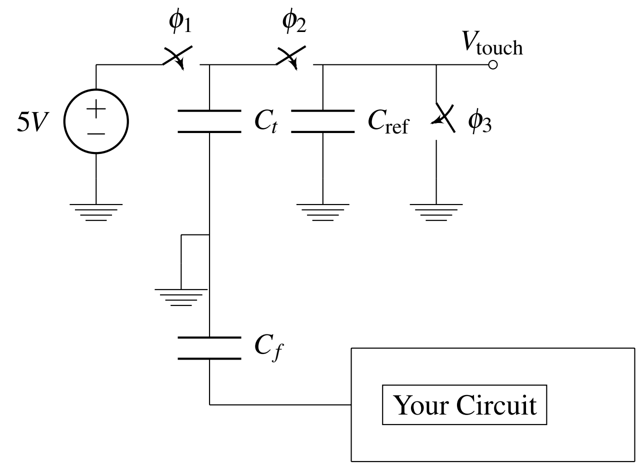


(a) (**5 points**) Write an expression for C_{ref} such that $V_{\text{touch}} = 2.5V$ in Phase 2 when nothing is touching the sensor. Assume C_t is some value $C_{\text{t,nom}}$ when there is no touch. Your answer may include $C_{\text{t,nom}}$. Justify your answer.

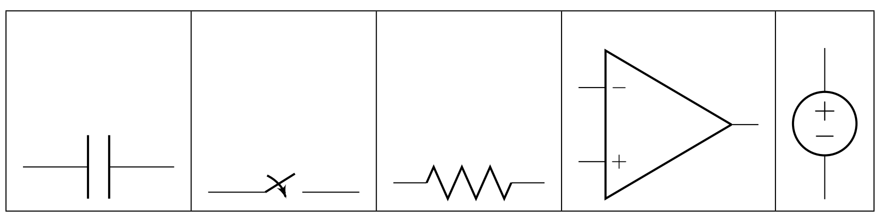
(b) (**5 points**) When a finger is touching the screen, C_t increases to twice it's nominal value, or $2C_{\text{t,nom}}$. Write an expression for V_{touch} in phase 2 when someone is touching the sensor. Use the value for C_{ref} you found in part a).

(c) (**5 points**) Let us now consider the C_f. Assume the plates have an area A and are nominally a distance d apart. When a force is applied on the screen, Plate 2 moves closer to Plate 3 by a distance x. We will ignore the capacitance between Plate 3 and Plate 1. **Find an expression for the capacitance C_f as a function of x, A, ε, and d.**

(d) (**10 points**) You will now design a circuit to measure the displacement of the screen. Your circuit should connect to the touch screen as shown below:

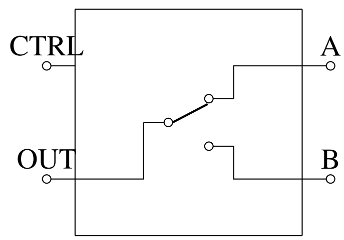


Design a circuit using the components in the box below to output V_{force}, a voltage which is inversely proportional to $(d-x)$. Solutions where V_{force} is some other function of x will receive partial credit.



(e) (**5 points**) We now have circuits to output two voltages: V_{touch} a voltage which changes value depending on whether there is a touch or not and V_{force} a voltage which is some function of the displacement x and thus the force applied on the screen. We want to output V_{force}, but only when there is a touch. For example, this could be useful to prevent your phone from activating apps while in your pocket. **Using V_{touch} the output of the touch-sensor and V_{force} the output of the force sensor as inputs to your circuit, design a circuit to output V_{out} such that $V_{\text{out}} = V_{\text{force}}$ if there is a touch, and 0V otherwise.**

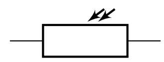
In addition to the box of components from part d), you have access to a voltage controlled switch shown below. When CTRL is connected to 5V, OUT is conneced to B, otherwise when CTRL is 0V out is connected to A. You cannot supply a voltage other than 0V or 5V to the CTRL input of the voltage controlled switch.



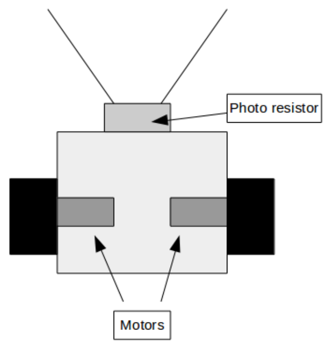
PRINT your name and student ID: ____

9. PetBot Design (30 points)

In this problem, you will design circuits to control PetBot, a simple robot designed to follow light. PetBot measures light using photoresistors. **A photo resistor is a light-sensitive resistor. As it is exposed to more light, its resistance decreases.** Given below is the circuit symbol for a photoresistor.



Below is the basic layout of the PetBot. It has one motor on each wheel. We will model each motor as a 1Ω resistor. When motors have positive voltage across them, they drive forward, when they have negative voltage across them, they drive backward. At zero voltage across the motors, the PetBot stops. The speed of the motor is directly proportional to the magnitude of the motor voltage. The light sensor is mounted to the front of the robot.

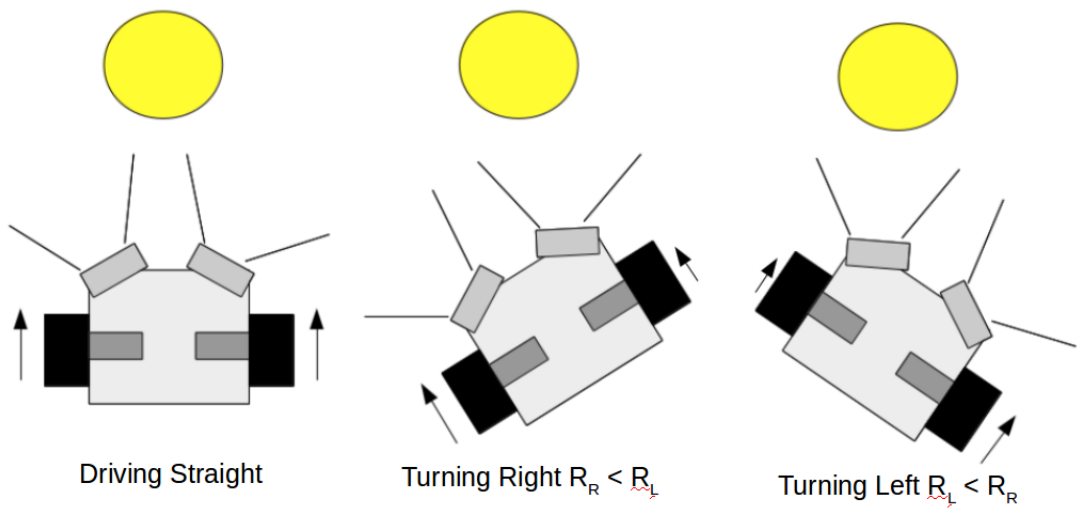


(a) (**5 points**) **Speed control -** Let us begin by first having PetBot decrease its speed as it drives toward the flashlight. **Design a motor driver circuit that outputs a decreasing positive motor voltage as the PetBot drives toward the flashlight. The motor voltage should be at least 5V far away from the flashlight. When far away from the flashlight, the photoresistor value will be $10K\Omega$ and dropping toward 100Ω as it gets close to the flashlight.** In your design, you may use any number of resistors and Op-Amps. You also have access to voltage sources of +10V and -10V. Based on your circuit, derive an expression for the motor voltage as a function of the circuit components that you used.

(b) (**15 points**) **Distance control -** Let us now have PetBot drive up to a flashlight (or away from the flashlight) and stop at distance of 1ft away from the light. At the distance of 1ft from the flashlight, the photoresistor has a value $1K\Omega$.

Design a circuit to output a motor voltage that is positive when the PetBot is at a distance greater than 1ft from the flashlight (making the PetBot move toward it), zero at 1ft from the flashlight (making the PetBot stop), and negative at a distance of less than 1ft from the flashlight (making the PetBot back-away from the flashlight. In your design, you may use any number of resisitors and Op-Amps. You also have access to voltage sources of value +10V and -10V. Based on your circuit, derive an expression for the motor voltage as a function of the values of circuit components that you used.

(c) (**10 points**) **Turning control -** We now want the PetBot to turn toward the flashlight that is not directly in front of it, while moving toward it and stopping at 1ft distance like before. To do so we will use two photoresistors angled slightly away from each other. In order to control the motion of the PetBot, we will use a simple control scheme described below.
Assume there are two photoresistors, R_L and R_R. If the **left** photoresistor is pointed towards the light, and thus has less resistance, we drive the **left** motor **slower** than the right motor to turn the PetBot towards the light. A similar process occurs if the right photoresistor is pointed towards the light.



Design the control circuit for the PetBot. Your circuit should use two photoresistors and drive two motors. Both motors should only stop at distance of 1ft from the flashlight. Note that since photoresistors are angled, their value at 1ft away from the flashlight has changed to $2K\Omega$. **Clearly label which side of the robot each photoresistor and motor belongs on.** You have access to op-amps, resistors, and voltage sources of value +10V and -10V. For this problem, you may assume the PetBot will initially always be far away from the flashlight. You may continue your work for this problem on the next page.

10. A Tale of Technocrats and Three Dream Cities (30 points)

This problem is a tale of Technocrats and three Dream Cities. The three cities are (I) San Francisco, (II) Chicago, and (III) Boston. The Technocrats don't die. They don't reproduce. In other words, their total population size is a constant from the initial time $n = 0$ onward indefinitely (as $n \to \infty$).

At the strike of every daily tick of a Universal Clock, each Technocrat chooses to either remain at the city he or she is already in, or move to another of the three Dream Cities instantaneously, but can only move in a single-hop fashion (i.e., traverse only one branch on its state-transition diagram, whether that branch is a self-loop or a branch to another node).

Let the state vector for this system be $s[n] = \begin{bmatrix} s_1[n] \\ s_2[n] \\ s_3[n] \end{bmatrix}$, where $s_\ell[n]$ denotes the *fraction* of the Technocrats in City ℓ at time n; for example, $s_3[n]$ denotes the fraction of the Technocrats who are in Boston on Day n. Accordingly, note that

$$1^\mathsf{T} s[n] = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} s_1[n] \\ s_2[n] \\ s_3[n] \end{bmatrix} = s_1[n] + s_2[n] + s_3[n] = 1, \qquad \forall n \in \{0, 1, 2, \ldots\}.$$

The state-evolution equation for this network is $s[n+1] = \mathbf{A}\, s[n]$, where the state-transition matrix is

$$\mathbf{A} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & \frac{1}{4} \end{bmatrix}.$$

In one or more parts below, you may or may not find it useful to know that eigenvectors corresponding to distinct eigenvalues are linearly independent.

(a) (**5 points**) Provide a *well-labeled* state-transition diagram that models the migration pattern of the Technocrats, as described above.

(b) (**5 points**) In this part, we're interested in a backward inference of the state vector $s[n]$ from a future state vector, say $s[n+1]$. Is it possible to determine the state of the network at time n (i.e., the state vector $s[n]$) from the state of the network at time $n+1$ (i.e., the state vector $s[n+1]$)? Provide a succinct, but clear and convincing explanation for your answer.

(c) (**5 points**) Without writing a single equation to derive the result, determine the limiting state vector:

$$\lim_{n\to\infty} s[n].$$

Explain your reasoning in succinct, but clear and convincing English (no mathematical derivation!).

(d) (**5 points**) By referring only to one or more specific aspects of the structure of the state-transition matrix **A**, and without any complicated mathematical derivations, determine the largest eigenvalue λ_1. Once you've inferred the largest eigenvalue λ_1, determine its corresponding eigenvector v_1 using whatever method suits your taste—though you should be able to determine v_1, too, with little mathematical exertion.

If you cannot see how to infer λ_1 from the structure of the matrix **A**, you may still receive partial credit if you determine the eigenpair (λ_1, v_1) as part of your derivations in the next part. If you choose to do so, simply write in the space below "*See my work in the next part.*," and otherwise leave the space for this part blank. If you write anything more for this part, we will not grade any of your work for (λ_1, v_1) in the next part, and we will grade only what you've written for this part.

(e) (**5 points**) Determine the remaining eigenpairs (λ_2, v_2) and (λ_3, v_3), where $\lambda_2 \geq \lambda_3$.

(f) (**5 points**) Suppose the initial population of Technocrats is distributed equally among the three cities. That is,

$$s[0] = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}$$

Express the state vector as

$$s[n] = \alpha_1[n]\, v_1 + \alpha_2[n]\, v_2 + \alpha_3[n]\, v_3, \tag{1}$$

where $\alpha_1[n]$, $\alpha_2[n]$, and $\alpha_3[n]$ are appropriate time-dependent scalar functions that *you must determine in as explicit and numerically-specific a form as possible*. If you're unsure of your numerical answers to the previous parts, you may express $\alpha_1[n]$, $\alpha_2[n]$, and $\alpha_3[n]$ symbolically, but still in the simplest form possible.

Explain how your expression for this part is consistent with the limiting state vector

$$\lim_{n\to\infty} s[n]$$

that you obtained previously. In particular, explain which eigenvalues of $\mathbf{A}$ influence the limiting state, and which are irrelevant.

11. Inverse Power Iteration (20 points)

In homework, we introduced the method of Power Iteration to find the dominant eigenvector of a given matrix. In this problem, we'll explore a similar method, Inverse Power Iteration, and use it to estimate eigenvectors and eigenvalues of a given matrix.

In Inverse Power Iteration, we let $\mathbf{A}$ be an $n \times n$ matrix and $\mathbf{B} = (\mathbf{A} - \mu\mathbf{I})^{-1}$ where μ is some scalar (we get to pick this scalar) and $\mathbf{I}$ is the $n \times n$ identity matrix. $\mathbf{B}$ must be constructed such that it is invertible for this method to work. We begin by making a guess μ for one of $\mathbf{A}$'s eigenvalue and a guess $b_0 \neq \vec{0}$ for the corresponding eigenvector. (Assume we never pick b_0 as an eigenvector of $\mathbf{B}$.) At each iteration we perform the update:

$$\vec{b}_{k+1} = \frac{\mathbf{B}\vec{b}_k}{\left\|\mathbf{B}\vec{b}_k\right\|}$$

As k becomes large, $\vec{b}_k$ converges to the eigenvector of $\mathbf{B}$ that corresponds to the eigenvalue with the largest magnitude.

(a) (**10 points**) Show that if $\vec{x}$ is an eigenvector of $\mathbf{A}$ with the corresponding eigenvalue λ, then $\vec{x}$ is also an eigenvector of $\mathbf{B}$ with eigenvalue $\frac{1}{\lambda-\mu}$.

(b) (**5 points**) Let **A** have unique eigenvalues $\lambda_1, \lambda_2, ...\lambda_i$. For a given choice of μ, where $\mu \neq \lambda_i$ show that with large k, $\vec{b}_k$ converges to the eigenvector of **A** whose eigenvalue λ_i is closest to μ.

(c) (**5 points**) Let **A** be a PageRank transition matrix. To find its steady state behavior we're looking for **A**'s eigenvector that corresponds to the eigenvalue $\lambda = 1$. Argue that we should **not** use $\mu = 1$ to perform inverse power iteration to estimate **A**'s eigenvector.

12. Patient Classification (15 points)

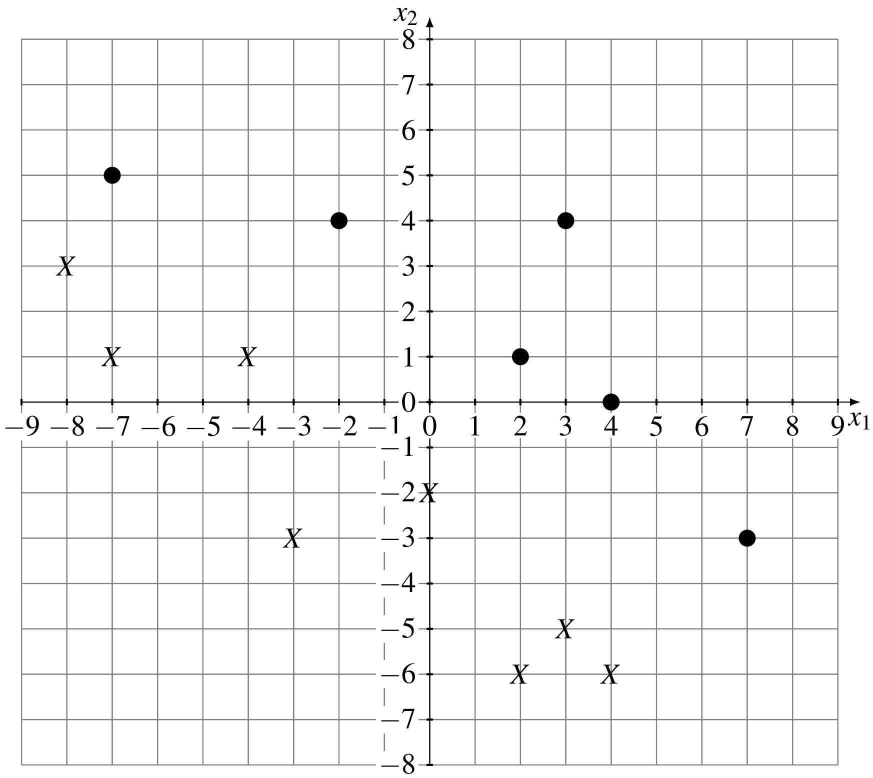
Consider a set of patients. Patient i can be represented by an attribute vector $\vec{x}^{(i)} = \begin{bmatrix} x_1^{(i)} \\ \\ x_2^{(i)} \end{bmatrix}$ as well as a known value $y^{(i)} \in \{+1, -1\}$ indicating whether they have the disease Dragon Pox. We want to design a simple classifier that can use the information from the data we have in order to predict whether a patient with attributes $x_1^{(i)}$ and $x_2^{(i)}$ and unknown diagnosis status has Dragon Pox. To do this, we will use our knowledge of least squares linear regression. We would like to design a linear function

$$f(\vec{x}^{(i)}) = \vec{w}^T \vec{x}^{(i)}$$

that takes in a vector $\vec{x}^{(i)}$ for a patient i and computes $y^{(i)} = sign(f(\vec{x}^{(i)}))$ to predict whether the patient has Dragon Pox.

(a) (**10 points**) Given that we are trying to minimize $\left[f(\vec{x}^{(i)}) - y^{(i)}\right]^2$ for each patient i and classification value $y^{(i)}$, what is the overall cost function $J(\vec{w})$ we are trying to minimize if we have n patients? Given a matrix $\mathbf{X}$ with each row vector corresponding to a patient and each column vector corresponding to an attribute and a vector $\vec{y}$ corresponding to the patient's illness status, what is the vector that minimizes this cost function? You may define the cost function in terms of a sum of individual terms, or as a matrix expression.

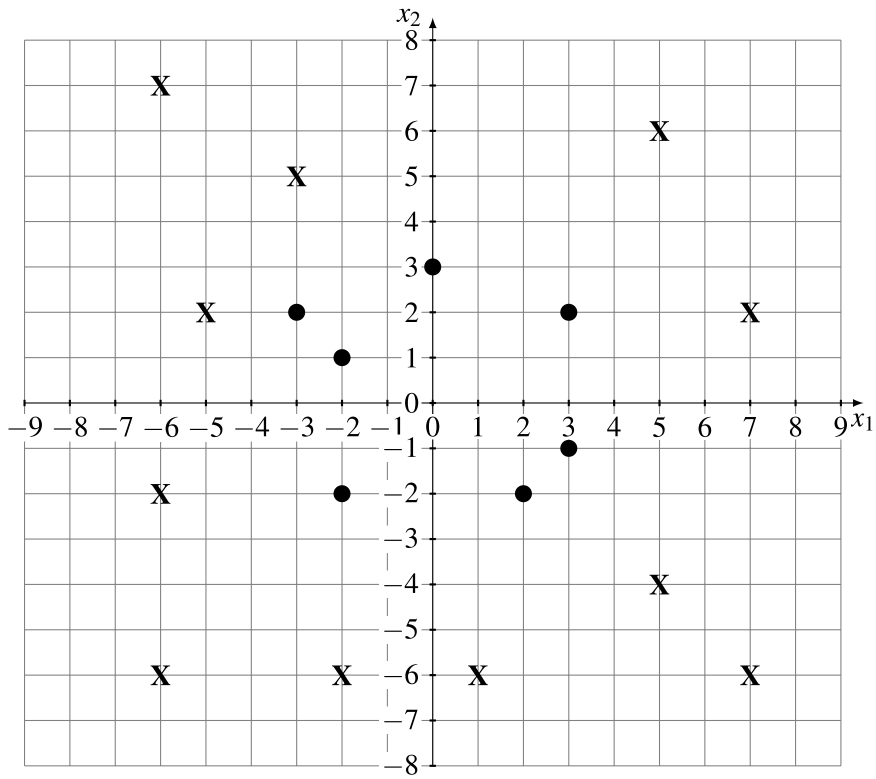
(b) (**5 points**) Suppose we have brand new set of data points as shown on the graph below. Dots represent positive diagnoses ($+1$) and crosses represent negative diagnoses (-1).



Let $w = [2,3]^T$. **Draw the line corresponding to the decision boundary on the graph.**
Given the following information, predict whether each patient has the disease. Fill in the last column of the table with "yes","no", or "inconclusive".

Patient	x_1	x_2	Disease?
1	1	0	
2	-6	1	
3	-5	-5	
4	9	-6	

(c) (**5 points**) Suppose the patient data looked like this:



Given that we still know our original attribute vector for each patient, how can we change it to classify this data correctly, such that their classes can be separated by a circle centered at the origin? Fill in the blanks with the new feature vector below:
Remember that we would like to design a linear function $f(\vec{x}^{(i)}) = \vec{w}^T \vec{x}^{(i)}$ such that $y^{(i)} = sign(f(\vec{x}^{(i)}))$.

$$\vec{x}^{(i)} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\vec{x}_{\text{new}}^{(i)} = \begin{bmatrix} \underline{\qquad\qquad\qquad} \\ \underline{\qquad\qquad\qquad} \end{bmatrix}$$

You may use this page for scratch work but **it will not be graded.**