**CSE/STAT 416** **Name:** _____
**Summer 2019**
**Final Exam**
**08/21/2019** **Student Number:** _____
**Time Limit: 2 hours 10 minutes**

---

**Do not open the exam before the exam begins and close the booklet when time is called. Starting early or working after time is called will lead to a -10 deduction.** You may write your name and Net ID on the front of the exam before the exam starts.

This exam contains 20 pages (including this cover page) and 11 questions.

Some questions have sub-parts. True/false, multiple choice, and multiple answer questions will have bubbles for you to fill in. Fill them in clearly to indicate an answer and erase/cross out an answer if you wish for us to not use that choice. **Circles represent problems where you should select one option, while squares represent questions where you should select all that apply.**

You are allowed to have one sheet of paper (both sides) with you as your cheat sheet. All other materials besides writing utensils should be put away before the exam starts. This includes all electronic devices like phones, calculators, and smart watches.

If you need more room to work out your answer to a question, you may ask for scratch paper. If you want to submit work on scratch paper to be graded, you should indicate so on the page of the question in the exam, indicate on the scratch paper which part is the answer, and staple your scratch paper to the **end** of your exam. Failure to do any of these steps may result in your scratch paper not being graded.

Work efficiently. Some questions are easier, some more difficult. Be sure to give yourself time to answer all of the easy ones, and avoid getting bogged down in the more difficult ones before you have answered the easier ones.

**Initials:** _____

Initial above to indicate you have read and agreed to these rules. Failure to initial may result in your exam not being accepted for credit.

Good luck!

| Question | Topic | Max. score | Score |
|---|---|---|---|
| 1 | Regularized Linear Regression | 14 | |
| 2 | Assessing Performance | 7 | |
| 3 | Logistic Regression | 4 | |
| 4 | Decision Trees | 15 | |
| 5 | Boosting | 7 | |
| 6 | Miscellaneous | 11 | |
| 7 | Precision Recall | 9 | |
| 8 | Clustering | 10 | |
| 9 | Principal Component Analysis | 6 | |
| 10 | Recommender Systems | 10 | |
| 11 | Deep Learning | 9 | |
| | Total | 102 | |

# 1 Regularized Linear Regression [14 points]

1. [9 points] Five different linear regression models are trained and evaluated. Assume the training set, validation set, and test set are the same for each model.

   - **M1**: Unregularized linear regression
   - **M2**: M1 + LASSO penalty with $\lambda = 0.1$
   - **M3**: M1 + LASSO penalty with $\lambda = 10$
   - **M4**: M1 + Ridge penalty with $\lambda = 0.1$
   - **M5**: M1 + Ridge penalty with $\lambda = 10$

   (a) [5 pts] Fill in the "Model" column in the table below by writing in the model number (M1, M2, etc.) that best explains each row.

| **Model** | **Coefficients** | **Train Error** | **Validation Error** | **Test Error** |
|---|---|---|---|---|
| M2 | $600x + 0x^2 + 500x^3$ | 0.15 | 0.23 | 0.20 |
| M5 | $3x + 4x^2 + 10x^3$ | 0.17 | 0.07 | 0.15 |
| M1 | $1000x + 2000x^2 + 1500x^3$ | 0.05 | 0.34 | 0.38 |
| M4 | $250x + 600x^2 + 500x^3$ | 0.12 | 0.13 | 0.10 |
| M3 | $10x + 0x^2 + 0x^3$ | 0.21 | 0.16 | 0.17 |

   (b) [2 pts] Which model would you choose? **(Select one)**

   ○ **M1** ○ **M2** ○ **M3** ○ **M4** ● **M5**

   (c) [1 pt] What would you estimate the generalization error on future data for the model you chose in the last problem?

   **Generalization error =** **0.15**

   (d) [1 pt] Which model is most likely to be overfit? **(Select one)**

   ● **M1** ○ **M2** ○ **M3** ○ **M4** ○ **M5**

2. [2 pts] Which of the following can possibly happen when adding a regularization term to the ordinary least quares regression quality metric? **(Select all that apply)**

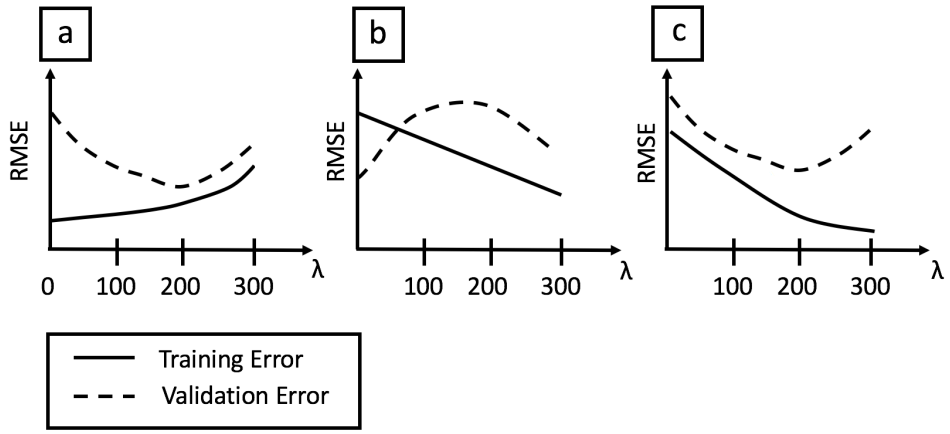
   - ■ **Increase training error**
   - ■ **Increase validation error**
   - ■ **Increase bias**
   - □ Increase variance

3. [3 pts] Suppose we perform $L_2$**-regularized linear regression** (ridge regression).

   (a) [2 pts] Which of the plots in Figure 1 describes a possible trend for training and validation error as a function of the ridge regression penalty parameter $\lambda$? **(Select one)**

   - ● **Figure 1(a)**
   - ◯ Figure 1(b)
   - ◯ Figure 1(c)

Figure 1: Training and validation error versus increasing $L_2$ penalty strength $\lambda$.




   (b) [1 pt] Based on your answer above, which of the four values of $\lambda \in \{0, 100, 200, 300\}$ would you choose to make your final predictions? **(Select one)**

   - ◯ $\lambda = 0$
   - ◯ $\lambda = 100$
   - ● $\lambda = 200$
   - ◯ $\lambda = 300$

# 2 Assessing Performance [7 points]

1. [2 pts] For the bias-variance tradeoff, which of the following substantially increases the test error more than the training error. **(Select one)**

   ◯ Bias ● **Variance**

2. [2 pts] A model is considered overfit if it achieves lower training error than another model. **(Select one)**

   ◯ True ● **False**

3. [3 pts] The two plots below show 3 points that are drawn randomly from the **same** underlying data generating mechanism:

   $$y_i = f(x_i) + \epsilon,$$

   Suppose we fit a **degree-2 polynomial** to two different datasets drawn from the same process.

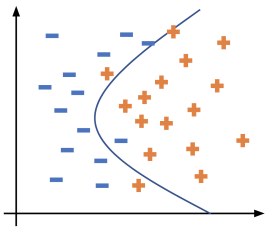
   (a) [2 pts] For each plot, draw the parabola that best fits the data.

   (b) [1 pt] Does our chosen model have **low** or **high** variance? Explain in a single sentence.

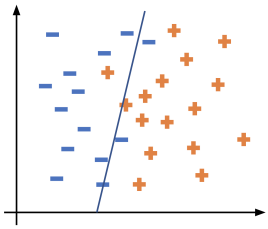
> **Solution:** This model, when using datasets of this size, has **high variance** because we get very different fits for slightly different datasets.

# 3 Logistic Regression [4 points]

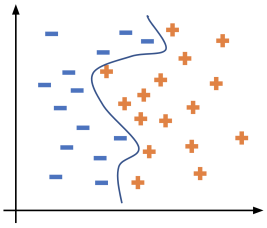
1. [5 points] Suppose we are using logistic regression to predict whether or not someone has a disease. We want to investigate the effect of using higher order polynomial features. Below, we have plotted the decision boundaries using logistic regression with different polynomial degree: one using linear features, one using quadratic features, and one using up to 10 degree polynomial features.



(a) Fig. 1



(b) Fig. 2



(c) Fig. 3

(a) [2 points] Identify which feature set was used to train the model shown in each figure. Write the figure number (e.g. 2) in the blank for the appropriate degree polynomial.

Linear Features = **2**

Quadratic Features = **1**

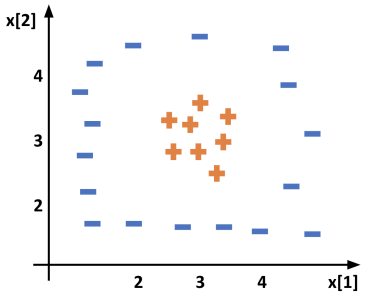
Degree 10 Features = **3**

(b) [2 points] For each model above, consider the "uncertainty region" of values $x$ such that $0.4 \leq P(y = +1|x) \leq 0.6$. Which set of features, when used with the logistic regression model, would you expect to have the **widest** uncertainty region? **(Select one)**

- ● **Linear Features**
- ○ Quadratic Features
- ○ Degree 10 Features
- ○ All have same width

# 4 Decision Trees [15 points]

1. [5 pts] Suppose we had the following dataset shown below, where the true rule the classifies the points is: $(2 \leq x[1] \leq 4)$ $AND$ $(2 \leq x[2] \leq 4)$. Our goal is to learn a decision tree (using binary splits) on this data that achieves 0 training error.

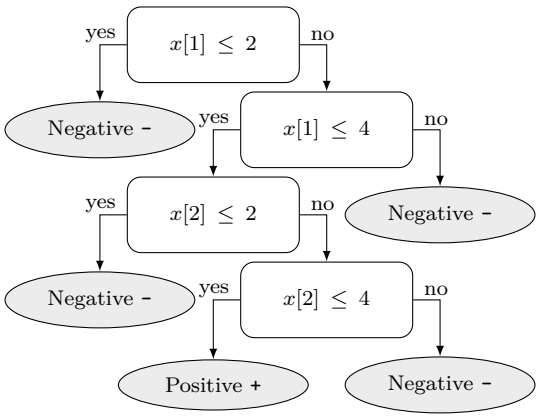



   (a) [2 pt] First, we want to know if it is possible for a decision tree to be learned on this dataset such that it achieves 0 training error. If it is possible, write the height of the **minimum** height tree that can achieve 0 training error. If it is not possible, write NA in the space below.

   **Solution:** We accepted any number that reasonably matched the tree drawn below since we did not precisely define height of the tree.

   (b) [3 pts] If you answered a number for the previous problem, draw a decision tree of that height that perfectly classifies the data. Make sure you clearly label the nodes of your tree so we can use it to classify the dataset. If you answered "NA" to the previous problem, explain in at most two sentences why it is not possible.

**Solution:**



2. [10 points] Your colleague is working on a machine learning project where they are using a **decision tree** to predict whether or not emails are spam based on some input features. Your colleague unfortunately did not take CSE/STAT 416 and keeps asking you to help them solve issues they are running into with their model. Here are the things you can suggest they try:

   a. Find a larger training set
   b. Increase the depth of the tree
   c. Decrease the depth of the tree
   d. Add an L1 regularization penalty
   e. Add a regularizer for the number of leaf nodes

   Considering each of the following issues separately, which of the above approaches has potential to help when applied by itself? **For each problem, if there are two or more strategies that will help, list <u>two</u> that you think are most likely to help. If there is only one strategy that will help, list it. If there are none, write "none".** To answer, write the letter associated to each strategy in the answer space for that problem.

   (a) [2 pts] They are seeing that their model has high training error and high test error.

   **Suggestion(s) =** **b**

   *This situation describes a model that is probably underfit. Increase complexity can help it fit better (b). Getting more data is likely to not be helpful since the model is not complex enough to fit the data already available.*

   (b) [2 pts] They are seeing that their model has low training error and high test error.

   **Suggestion(s) =** **Two of a, c, or e**

   *This situation describes a model that is probably overfit. Acceptable answers are to use a less complex model (c or e) or to get more data to prevent it from overfitting (a). L1 regularization is not appropriate for tree models since there are no coefficients to regularize.*

   (c) [2 pts] They suspect their model is underfitting

   **Suggestion(s) =** **b**

   *This is another way of describing the first problem in this set (a)*

(d) [2 pts] They suspect their model is overfitting

**Suggestion(s) = Two of a, c, or e**

*This is another way of describing the second problem in this set (b)*

(e) [2 pts] They are having difficulty understanding which inputs are the most important for prediction

**Suggestion(s) = none but we also accept c/e**

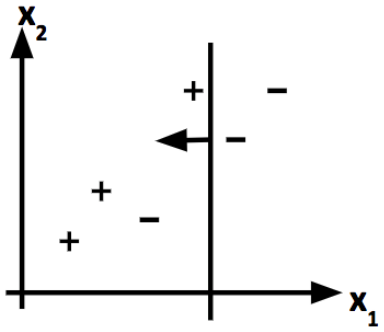
*Decision trees can already help with feature importance since you can look at features split early in the tree. We accepted none because making the tree shorter doesn't necessarily help make the feature importances early in the tree. We also accepted suggestions to make the tree simpler (c or e) even though you can still easily do feature importance with tall trees. L1 regularization (d) is not applicable to trees since they do not have coefficients.*

# 5 Boosting [7 points]

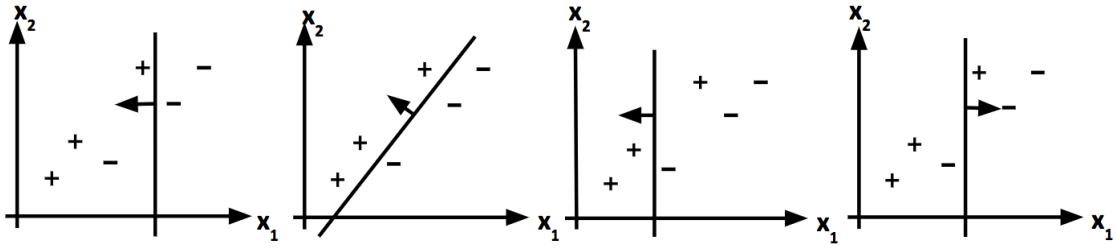
1. [5 points] Consider the algorithm for training the AdaBoost model. Recall that the algorithm trains a series of decision stumps on the data and changes the weights of points it misclassifies.

   The figure below displays a 2-dimensional training dataset, as well as the first stump chosen. The little arrow in the figure is the normal to the stump decision boundary indicating the positive side where the stump predicts +1. All points start with uniform weights.

   (a) [2 pts] **Circle all points** in the figure below whose weight will **increase** as a result of incorporating the first stump.




   (b) [2 pts] Which of the figures below shows the best stump that we could select at the next boosting iteration? **(Select one)**




   ○ Fig a ○ Fig b ● **Fig c** ○ Fig d

   (c) [1 pt] If you keep running AdaBoost on this dataset until the training error does not decrease further, the training error will be: **(Select one)**

   ● **0**
   ○ $^1/_6$
   ○ Cannot be determined

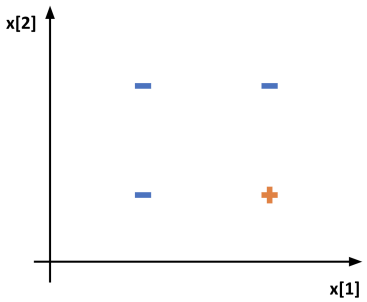
2. [2 points] Suppose your friend just started studying ensemble models. **In at most three sentences**, explain the difference between AdaBoost and Random Forests to your friend. Assume your friend does not know what "Bagging" or "Boosting" mean in general, so you should explicitly compare details of the AdaBoost and Random Forest models. Write your answer in the space below.

**Solution:** One example answer:

AdaBoost uses very simple models (weak learners) that are weighted differently based on how accurate they were on their training data. Additionally, the models are trained sequentially on weighted datasets such that data points that were harder to classify for previous models receive more weight for later models. Random Forests train many decision trees that are intended to overfit on random samples (with replacement) of the original dataset.

# 6 Miscellaneous [11 points]

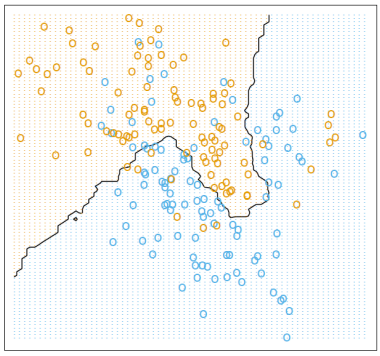
1. [3 points] Below is a training set with 4 examples where each is labelled + or -. The inputs $x[1]$ and $x[2]$ are real-valued numbers and the labels $y_i \in \{-1, +1\}$. The points form a square so we know that adjacent points will be vertically or horizontally aligned.




Which of the following models can achieve 100% training accuracy using only the data inputs as features? **(Select all that apply)**

- ■ **Logistic Regression**
- ■ **1-Nearest Neighbor**
- □ 2-Nearest Neighbor
- □ Decision Tree with depth 1
- ■ **Decision Tree with depth 2**
- ■ **Neural Network using logistic activation with no hidden layers**

2. [2 pts] Consider the following decision boundary learned by a k-Nearest Neighbor algorithm.



Which value of $k$ most likely learned this decision boundary? **(Select one)**

- ◯ $k = 1$
- ◯ $k = 2$
- ● $k = 10$
- ◯ $k = 100$

3. [6 pts] You have learned many ML models and techniques this quarter. In practice, one of the hardest things is just knowing which tool you should invest time in that is most likely to solve a task well. Each sub-part of this problem will present a situation and you should identify one model or technique from the list below that can best solve the problem. **There may be more than one correct answer, but you should pick one that you think is the most likely to succeed. State which model you'd use and a one sentence justificaiton of why.** A model may be used more than once.

- Linear Regression
- Ridge Regression
- LASSO Regression
- Logistic Regression
- Decision Tree
- Random Forest
- AdaBoost
- k-Nearest Neighbors
- k-means
- Locality Sensitive Hashing
- Principal Component Analysis
- Fully Connected Neural Network
- Convolutional Neural Network

(a) [1 pt] You have a weather dataset that has very few rows and many data inputs (columns). You want to predict the day's temperature (in Celsius) given other weather conditions.

**Solution:** We accepted any model that can do regression that was not a neural network since they tend to require much more training data.

(b) [1 pt] You want to visualize (in 2D) the relationship between images in a dataset of images.

**Solution:** PCA allows you to to find 2-dimensions that can best represent the dataset that minimizes the reconstruction error.

(c) [1 pt] You want to predict the probability a student passes a class given how many hours they studied for an exam. Based on your domain expertise, you suspect there is a linear relationship between study time and likelihood of passing.

**Solution:** Logistic Regression is the model that we learned in class that estimates probabilities and assumes the likelihood is controlled by a linear function.

(d) [1 pt] You have an unlabeled image dataset containing pictures of dogs and cats, and are interested in grouping all the dog and cat images into separate groups.

**Solution:** k-means is the unsupervised algorithm that we learned to help cluster examples in a dataset based on similarity. Many students answered PCA which might help get better results, but this does not solve the problem described since you would still need to do something like k-means afterwards.

(e) [1 pt] You are faced with a learning task of identifying whether or not to give someone a loan based on their credit history. You have tried some simple models but they aren't performing well. You are interested in using an ensemble model that you can train the sub-models in parallel on a large cluster of computers.

**Solution:** Random Forest is an ensemble model where the models could be trained independently.

(f) [1 pt] You want to participate in the CIFAR-100 challenge which is a image classification task where you are given a large image dataset, where each image is labelled with one of 100 possible labels. Training time is not a concern.

**Solution:** CNN is the model of choice for image recognition since they tend to have really good generalization error on image datasets.

# 7 Precision/Recall [9 points]

1. [7 points] Consider the following confusion matrix for some model on some datset

| | Predicted + | Predicted − |
|---|---|---|
| True + | 200 | 10 |
| True − | 20 | 5 |

For each of the sub-problems, leave your answer as a fraction. You should not leave your answer in terms of a summation, but you do not need to reduce/simplify your fractions.

(a) [2 pts] What is the accuracy of the model?

**Accuracy =** **205/235**

(b) [2 pts] What is the precision of the model?

**Precision =** **200/220**

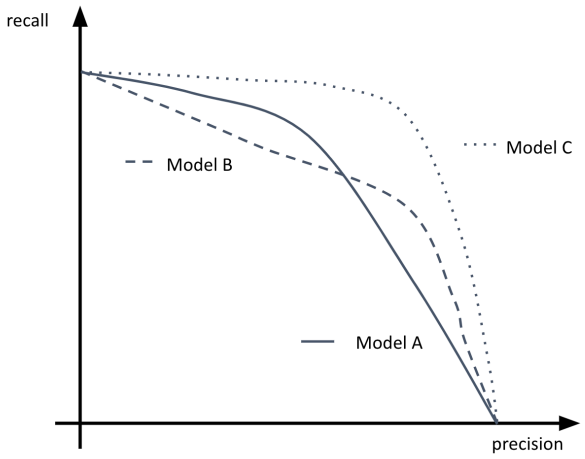
(c) [2 pts] What is the recall of the model?

**Recall =** **200/210**

(d) [1 pt] Is there a class imbalance in this dataset? Select which one is over-represented or none if they are balanced. **(Select one)**

● **+ class** ○ − class ○ No imbalance

2. [2 pts] Consider the following precision recall curves. Which model would you choose if your only considerations were precision and recall? **(Select one)**




○ Model A

○ Model B

● **Model C**

# 8 Clustering [10 points]

## 8.1 K-Means Clustering [6 points]

1. [2 points] Suppose you have an unlabeled dataset:

$$\{x_1, x_2, ..., x_N\}$$

You run k-means with 50 different random initializations (always with the same value of k), and obtain 50 different clusterings of the data. What is the recommended way for choosing which one of the clusterings to use? **(Select one)**

- ◯ Always pick the final (50th) clustering found, since by that time it is more likely to have converged to a good solution.
- ◯ The only way to do so is if we also have labels $y_i$ for our data.
- ● **Choose the clustering with smallest $\frac{1}{N}\sum_{i=1}^{N}\|x_i - \mu_{z_i}\|^2$, where $\mu_{z_i}$ is the mean of the cluster to which $x_i$ is assigned.**
- ◯ The answer is ambiguous, and there is no good way of choosing.

2. [2 pts] If you run k-means multiple times using the same initial cluster centers, the resulting clusters will be the same. **(Select one)**

● **True** ◯ False

3. [2 pts] Clustering with k-means is a **supervised learning** problem. **(Select one)**

◯ True ● **False**

## 8.2 Hierarchical Clustering [4 points]

3. [4 points] Recall that for agglomerative hierarchical clustering, we computed the distances between clusters using a *linkage function*. In class we saw the **single linkage**:

$$dist(C_1, C_2) = \min_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2)$$

This function uses some metric $d$ to measure the distance between points (assumed here to be **standard Euclidean distance**). Instead of single linkage, you decide to try **complete linkage**:

$$dist(C_1, C_2) = \max_{x_1 \in C_1, x_2 \in C_2} d(x_1, x_2)$$

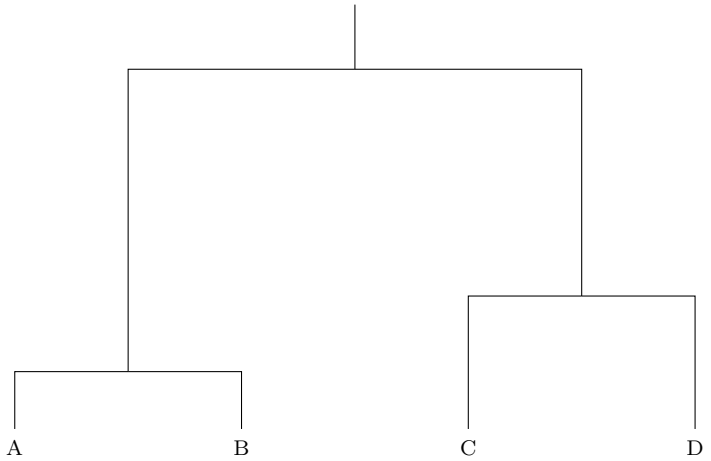
This definition is similar to single linkage, but we are looking at the *maximum* distance between the points in the clusters when deciding which clusters are closest together.

In the table below we show the pair-wise distances between 4 points. For example, this table tells you that A and B are 0.08 units away.

| | **A** | **B** | **C** | **D** |
|---|---|---|---|---|
| **A** | 0.00 | 0.08 | 0.65 | 0.78 |
| **B** | | 0.00 | 0.39 | 0.56 |
| **C** | | | 0.00 | 0.63 |
| **D** | | | | 0.00 |

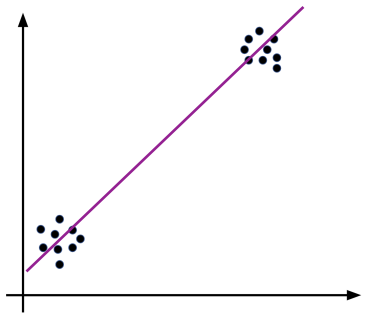
In the space below, draw a dendrogram that shows the order that the clusters are formed on this data using the agglomerative clustering algorithm with the **complete linkage** function as defined above. **We do not care about the exact heights of the dendrograms, just that they are joined in the right order**. Make sure you label the points on the dendrogram.

**Solution:**



# 9 Principal components analysis (PCA) [6 points]

1. [4 pts] Suppose you were to use PCA on the following dataset



Dataset for PCA

   (a) [2 pts] **On the figure above**, draw the line that is in the direction of the first principal component of the data.

   (b) [1 pt] Justify your answer to part (a) in one sentence.

**Solution:** This is the direction that minimizes reconstruction error (alternatively, the direction of maximum variance).

   (c) [1 pt] One the line below, draw what the dataset will look like after projecting onto the first principal component.

   We are not looking for you to make a picture-perfect projection, but your drawing should capture the general idea of how the data will be projected onto the first principal component.

2. [2 pts] This question concerns PCA and LASSO. **(Select all statements that are true)**

   - ■ **PCA and LASSO can both be used for dimensionality reduction.**
   - ■ **LASSO selects a subset (potentially all of) of the original features.**
   - □ PCA and LASSSO both allow you to pre-specify how many features are chosen.
   - ■ **PCA produces features that are linear combinations of the original features.**

# 10 Recommender Systems [10 points]

1. [1 pts] As we increase the number of items that are recommended by our model what would we expect to happen to the precision of the model? **Select one.**

   ● **Precision will Increase**

   ◯ Precision will Decrease

   ◯ Shouldn't see a difference

2. [3 pts] Recall the featurized matrix factorization model that combines the predictions from a matrix factorization model with a classification model trained on a set of features. In at most three sentences, explain **2 advantages** and **1 disadvantage** to using a featurized matrix factorization model rather than just a matrix factorization model.

**Solution:** One example answer:

Featurized matrix factorization can help solve the cold start problem. Additionally, it also can improve recommendations by capturing more context about the users or items based on additional features than just ratings. A disadvantage is that it can be difficult to train a model that actually helps since the right features might be difficult to come by.

3. [6 pts] Let's use matrix factorization for a slightly different context than recommending movies. You have a small dataset consisting of 2 drugs and 2 diseases. The observed drug effectiveness are shown as numbers (higher is better) and '?' indicate missing observations; these are the entries we wish to predict in order to identify which drug might be best for a disease.

| | **Drug 1** | **Drug 2** |
|---|---|---|
| Disease 1 | 3 | 1 |
| Disease 2 | ? | 2 |

You choose to use matrix factorization and run coordinate descent for some number of iterations and arrive at the following disease and drug factors:

| | **Disease Factors** |
|---|---|
| Disease 1 | [0, 1, 2] |
| Disease 2 | [1, 0, 1] |

| | **Drug Factors** |
|---|---|
| Drug 1 | [2, 1, 0] |
| Drug 2 | [0, 0, 1] |

(a) [2 pts] What is the current residual sum of squares loss?

**RSS =** $2^2 + 1^2 + 1^2 = 6$

(b) [2 pts] With this current setting of the disease and drug factors, which of the drugs is predicted to be most effective against *Disease 2*. **Select one.**

● **Drug 1** ○ Drug 2 ○ Can't tell with this information

(c) [2 pts] A new drug, Drug 3, comes out. Assume the drug vector is initialized to zeroes and we rerun coordinate descent using the effectiveness data presented at the beginning of the problem. What would you predict the effectiveness of Drug 3 to be for these diseases? **(Select an option and fill in the blanks if necessary).**

● **Disease 1:** 0 **Disease 2:** 0

○ Cannot determine

# 11 Deep Learning [9 points]

1. [3 pts] Which of the following are true about convolutional neural networks (CNNs) for image analysis? **(Select all that apply)**

■ **Earlier layers tend to include edge detectors**

■ **Pooling layers reduce the spatial resolution (i.e. size) of the image**

□ They have more parameters than fully connected networks with the same number of layers and the same number of neurons per layer

□ A CNN can be trained for unsupervised learning, whereas an ordinary neural network cannot

2. [2 pt] Consider one layer of weights (edges) in a CNN for gray scale image classification, where these weights connect one layer of neurons to the next. Which type of layer has the **fewest** parameters to be learned during training. **(Select one)**

   ◯ A convolution layer with ten 3x3 kernels

   ◯ A convolution layer with eight 5x5 kernels

   ● **A max-pooling layer that reduces a 10x10 image to a 5x5 image**

   ◯ A fully connected layer from 20 hidden neurons to 4 output neurons

3. [1 pt] (**True or False**) The XOR operator can be modeled using a neural network with a single hidden layer (i.e. a 3-layer network).

   ● **True**

   ◯ False

4. [3 pts] Using the fully connected neural network below, what is the prediction for the data point $x = (2, 3)$? **Write your answer in the space at the bottom.**

   Assume every neuron (even the last one) uses the following activation function:

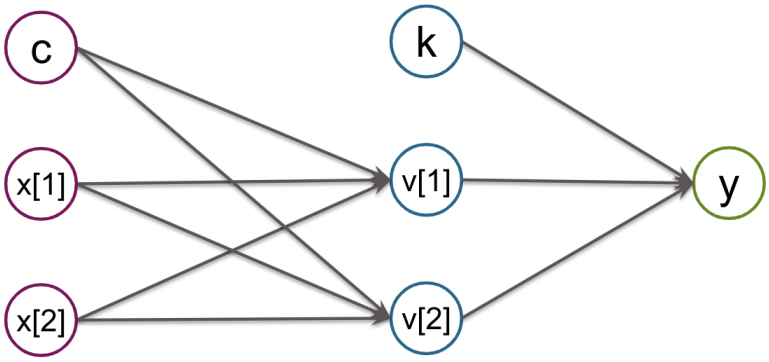
$$g(z) = \begin{cases} -1 & z < 0 \\ 0 & z = 0 \\ 1 & z > 0 \end{cases}$$

   Assume the network is using the weights

| | $v[1]$ | $v[2]$ |
|---|---|---|
| $c$ | 1 | -1 |
| $x[1]$ | 1 | -1 |
| $x[2]$ | -2 | 2 |

| | $y$ |
|---|---|
| $k$ | -1 |
| $v[1]$ | 2 |
| $v[2]$ | 1 |

   What is the prediction for the input $x = (2, 3)$?




**y =** **-1**

*See next page for work*

$$
\begin{aligned}
v[1] &= g(c + 1 \cdot x[1] - 2 \cdot x[2]) \\
&= g(1 + 1 \cdot 2 - 2 \cdot 3) \\
&= g(-3) = -1 \\
v[2] &= g(k - 1 \cdot x[1] + 2 \cdot x[2]) \\
&= g(-1 - 1 \cdot 2 + 2 \cdot 3) \\
&= g(3) = 1 \\
y &= g(k + 2 \cdot v[1] + 1 \cdot v[2]) \\
&= g(-1 + 2 \cdot (-1) + 1 \cdot 1) \\
&= g(-2) = -1
\end{aligned}
$$