IntroML: Final Review Competition

Kyle Swanson

Massachusetts Institute of Technology

Rules

- Teams
  - 3 people per team
- Open notes - you can use any material from the class
- 20 Questions
  - First team to answer correctly gets the points
  - Some questions have bonus questions worth additional points
  - Questions vary in difficulty - harder questions are worth more points
  - Questions fall into three categories
    - Theory
    - Computation - have a piece of paper ready
    - Coding - have your laptops ready
- Winning
  - Team with most points wins
  - Prizes for the top 2 teams

Question 1 (10 points)

Question 1 (10 points)

Question: Define machine learning.

Question 1 (10 points)

Question: Define machine learning.

Answer: “Field of study that gives computers the ability to learn without being explicitly programmed.” (Arthur Samuel, 1959)

Question 2 (10 points)

Question 2 (10 points)

Question: Give an example of each of the following types of problems:

- Classification
- Regression
- Generation

Question 2 (10 points)

Question: Give an example of each of the following types of problems:

- Classification
- Regression
- Generation

Answer:

- Sentiment analysis for Amazon reviews (Lab 2)
- Movie rating prediction (Lab 6)
- Digit image generation (Lab 10)

Question 3 (10 points)

Question 3 (10 points)

Question: Give an algorithm for each of the following types of learning:

- Supervised
- Unsupervised
- Reinforcement

Question 3 (10 points)

Question: Give an algorithm for each of the following types of learning:

- Supervised
- Unsupervised
- Reinforcement

Answer:

- Perceptron, SVM, random forests, neural networks
- PCA, k-means, autoencoders
- Q-Learning, DQN

Question 4 (20+20 points)

Question 4 (20+20 points)

Question: Explain the tradeoff between overfitting and generalization.

Question 4 (20+20 points)

Question: Explain the tradeoff between overfitting and generalization.

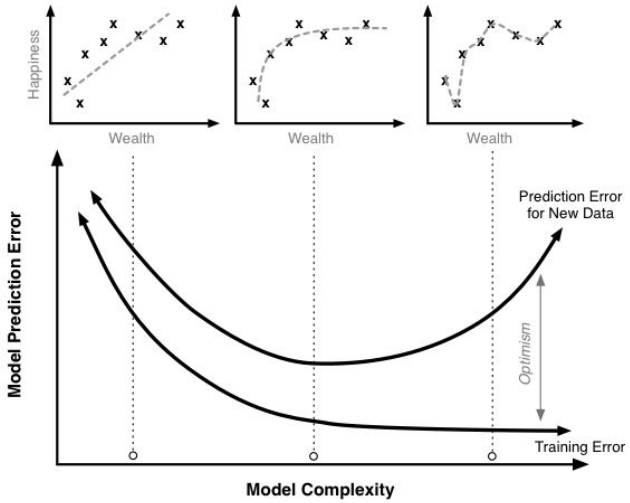
Bonus question: Draw a diagram illustrating the training and testing error as a function of model complexity.

Question 4 (20+20 points)

Question: Explain the tradeoff between overfitting and generalization.

Bonus question: Draw a diagram illustrating the training and testing error as a function of model complexity.

Answer:



Question 5 (20 points)

Question 5 (20 points)

Question: Draw the theta vector and the decision boundary for

$$\theta = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

Question 5 (20 points)

Question: Draw the theta vector and the decision boundary for

$$\theta = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

Answer: See whiteboard

Question 6 (20 points)

Question 6 (20 points)

Question: Given theta and theta_0 below, how will x be classified?

$$\theta = \begin{pmatrix} -3 \\ 2 \\ 5 \\ -1 \end{pmatrix}, \ \theta_0 = 4, \ x = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Question 6 (20 points)

Question: Given theta and theta_0 below, how will x be classified?

$$\theta = \begin{pmatrix} -3 \\ 2 \\ 5 \\ -1 \end{pmatrix}, \ \theta_0 = 4, \ x = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Answer:

$$\text{sign}(\theta \cdot x + \theta_0) = \text{sign}(-3 * 1 + 2 * 2 + 3 * 5 + -1 * 4 + 4) = \text{sign}(16) = +1$$

Question 7 (20 points)

Question 7 (20 points)

Question: Draw two datasets, one which can be separated by the perceptron algorithm and one which cannot.

Question 7 (20 points)

Question: Draw two datasets, one which can be separated by the perceptron algorithm and one which cannot.

Answer: See whiteboard

Question 8 (50 points)

Question 8 (50 points)

Question: Can a linear classifier separate the following dataset?

```
from sklearn.datasets import make_blobs
X, y = make_blobs(centers=2, random_state=42)
```

Question 8 (50 points)

Question: Can a linear classifier separate the following dataset?

```
from sklearn.datasets import make_blobs
X, y = make_blobs(centers=2, random_state=42)
```

Hint: Train a linear classifier on X and y. Then use the classifier to predict the labels for X. Compare these predicted labels to y with `sklearn.metrics.accuracy_score`
What does the accuracy tell you about whether the linear classifier can separate the dataset?

Question 8 (50 points)

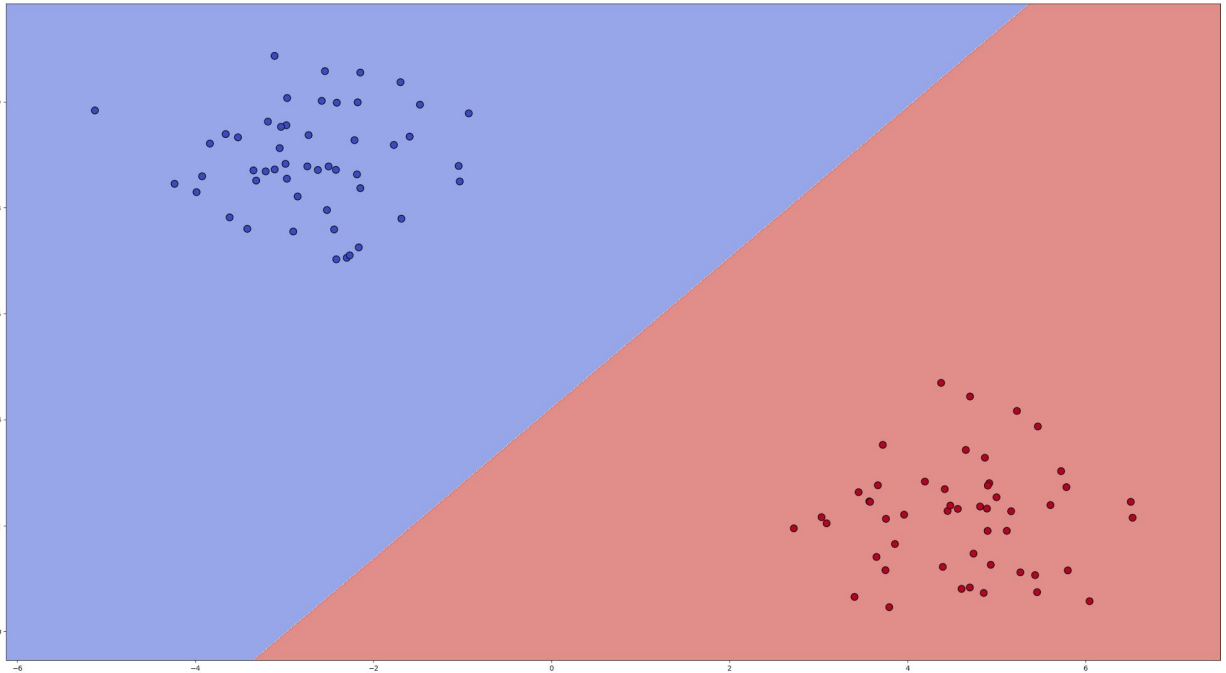
Question

```
from sk
X, y = r
```

Hint: Trair edict the labels for

`sklearn`

What does an separate the datase



Answer: Accuracy = 100%, meaning the dataset is linearly separable

Question 9 (10+10 points)

Question 9 (10+10 points)

Question: Why is non-linearity important when building classifiers?

Question 9 (10+10 points)

Question: Why is non-linearity important when building classifiers?

Bonus question: Draw a dataset which cannot be separated by a linear classifier but can be separated by a quadratic classifier.

Question 9 (10+10 points)

Question: Why is non-linearity important when building classifiers?

Bonus question: Draw a dataset which cannot be separated by a linear classifier but can be separated by a quadratic classifier.

Answer: Because non-linearity allows our classifier to classify a much broader range of datasets than just those which are linearly separable. See whiteboard.

Question 10 (10+10 points)

Question 10 (10+10 points)

Question: Describe a method for turning a linear classifier into a non-linear classifier.

Question 10 (10+10 points)

Question: Describe a method for turning a linear classifier into a non-linear classifier.

Bonus question: Describe a second method for turning a linear classifier into a non-linear classifier.

Question 10 (10+10 points)

Question: Describe a method for turning a linear classifier into a non-linear classifier.

Bonus question: Describe a second method for turning a linear classifier into a non-linear classifier.

Answer: 1) Use a non-linear data transformation $x \mapsto \phi(x)$

2) Use a kernel function

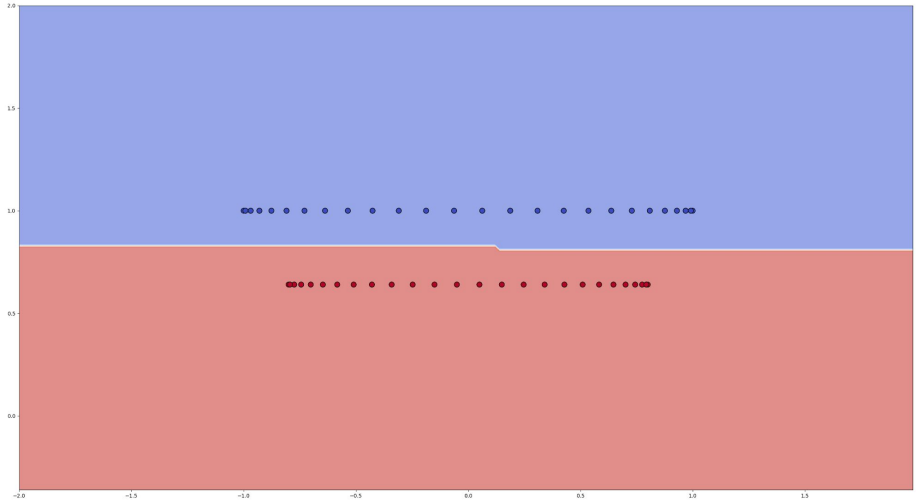
Question 11 (50 points)

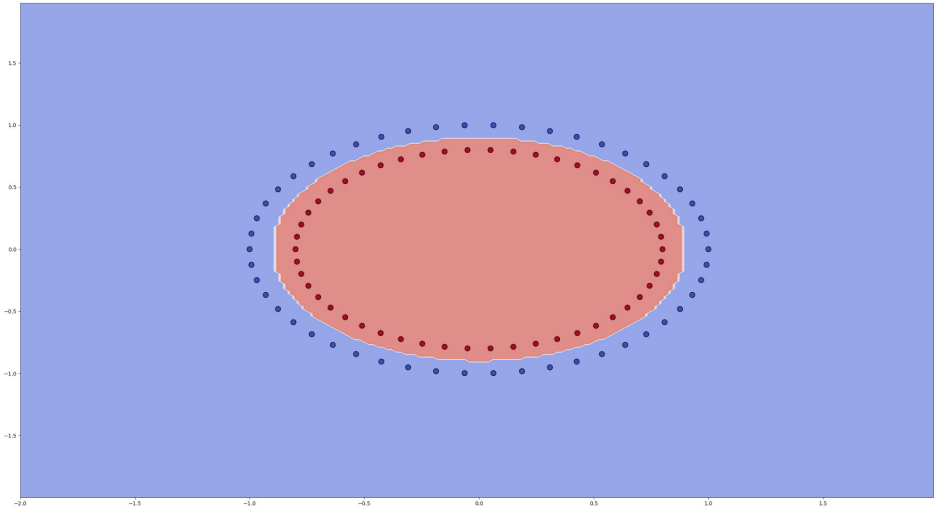
Question 11 (50 points)

Question: Build a non-linear classifier which can separate the following dataset with 100% accuracy.

```
from sklearn.datasets import make_circles
X, y = make_circles(random_state=42)
```

Question 11 (50 points)





Answer: 1) Use non-linear transformation $\phi(x) = \phi\left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}\right) = \begin{pmatrix} x_1 \\ x_1^2 + x_2^2 \end{pmatrix}$

2) Use the radial basis kernel

Question 12 (20+5 points)

Question 12 (20+5 points)

Question: What is an ensemble and why do we use ensembles?

Question 12 (20+5 points)

Question: What is an ensemble and why do we use ensembles?

Bonus question: Name an ensemble algorithm.

Question 12 (20+5 points)

Question: What is an ensemble and why do we use ensembles?

Bonus question: Name an ensemble algorithm.

Answer: An ensemble is a collection of classifiers. Ensembles are used because they counteract the bias of any single classifier, thereby producing better predictions. The Random Forest Algorithm is an example of an ensemble algorithm.

Question 13 (15 points)

Question 13 (15 points)

Question: When building a decision tree, how do we decide what feature to use at each point in the tree?

Question 13 (15 points)

Question: When building a decision tree, how do we decide what feature to use at each point in the tree?

Answer: We compute the gini impurity coefficient, which measures how well each feature splits the remaining data. The feature with the smallest gini coefficient (most pure split) is used.

Question 14 (10 points)

Question 14 (10 points)

Question: Name a method for recommending content.

Question 14 (10 points)

Question: Name a method for recommending content.

Answer: Content-based recommendation (using linear regression), collaborative filtering (using nearest-neighbors prediction or low-rank matrix factorization)

Question 15 (15 points)

Question 15 (15 points)

Question: A single neuron with a linear activation is equivalent to what other type of machine learning algorithm?

Question 15 (15 points)

Question: A single neuron with a linear activation is equivalent to what other type of machine learning algorithm?

Answer: Perceptron

Question 16 (10 points)

Question 16 (10 points)

Question: How do neural networks incorporate non-linearity?

Question 16 (10 points)

Question: How do neural networks incorporate non-linearity?

Answer: By using non-linear activation functions (like tanh, sigmoid, relu).

Question 17 (10 points)

Question 17 (10 points)

Question: What does the Universal Approximation Theorem say? (in English, not in math)

Question 17 (10 points)

Question: What does the Universal Approximation Theorem say? (in English, not in math)

Answer: That a neural network with a single hidden layer with a finite number of neurons can approximate any reasonable function.

Question 18 (10+10+10 points)

Question 18 (10+10+10 points)

Question: Name the algorithm which is used to learn the weights of a neural network.

Question 18 (10+10+10 points)

Question: Name the algorithm which is used to learn the weights of a neural network.

Bonus question: Name the method for updating each of the weights.

Question 18 (10+10+10 points)

Question: Name the algorithm which is used to learn the weights of a neural network.

Bonus question: Name the method for updating each of the weights.

Bonus bonus question: Write the gradient descent update step.

Question 18 (10+10+10 points)

Question: Name the algorithm which is used to learn the weights of a neural network.

Bonus question: Name the method for updating each of the weights.

Bonus bonus question: Write the gradient descent update step.

Answer: Backpropagation. Gradient descent.

$$W = W - \eta \nabla_W J(\theta)$$

Question 19 (20 points)

Question 19 (20 points)

Question: What are convolutional neural networks (CNNs) and recurrent neural networks (RNNs) typically used for?

Question 19 (20 points)

Question: What are convolutional neural networks (CNNs) and recurrent neural networks (RNNs) typically used for?

Answer: CNNs are typically used for image data while RNNs are typically used for sequence data (most often text).

Question 20 (50 points)

Question 20 (50 points)

Question: Use dimensionality reduction to estimate the number of classes contained in the iris dataset.

```
from sklearn.datasets import load_iris
X = load_iris().data
```

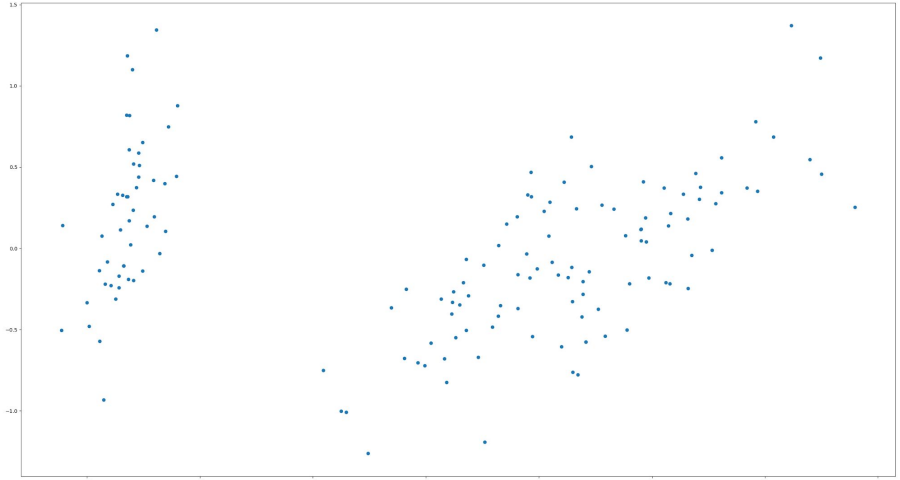
Question 20 (50 points)

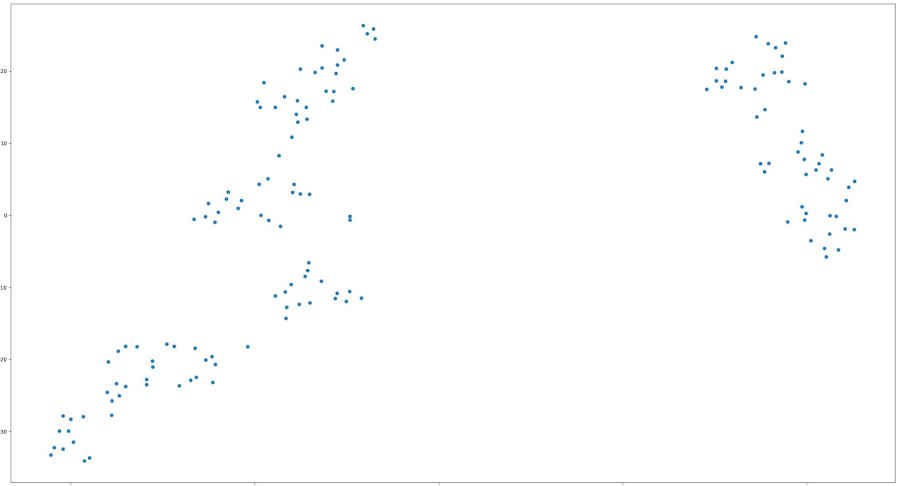
Question: Use dimensionality reduction to estimate the number of classes contained in the iris dataset.

```
from sklearn.datasets import load_iris
X = load_iris().data
```

Hint: Reduce the data to two dimensions and plot with `matplotlib.pyplot.scatter`

Question 20 (50 points)





Answer: PCA shows 2 distinct clusters, so a good guess would be 2 classes. TSNE with default parameters also shows 2 clusters. TSNE with `perplexity=8` and `random_state=42` better shows the true value, 3 classes.