Intro

How would you write a program that could detect if an image had a cat?

It's a very hard problem to solve because cats come in all sorts of sizes, shapes, and colors. Pictures are taken at various angles, lighting conditions, and scales.

It's easy, however, to get a bunch of pictures of cats from the internet and a bunch of pictures of "non-cats".

A situation like this (i.e one where it's hard to write a program but easy to find sample data points) is perfect for machine learning.

In machine learning, we take a general purpose statistical model, feed it a lot of sample data (aka training data), and optimize it to solve a particular problem.

Question: Give an example of a problem where it is hard to write a program, but easy to collect a training dataset.

Machine Learning Problems

ML can solve many kinds of problems, but the most common problems are listed below in order of popularity.

Classification

In classification, our model aims to examine a data point and put it into 1 out of K possible classes. When K = 2, we call it binary classification. Here are some examples of classification problems:

- Youtube: Detect if a comment contains hate speech
- Google: From a user's search history, current query, age, income, and more, detect whether a user will click on an ad
- Waymo: Detect if the image from a self driving car's camera shows a stop sign, yield sign, or traffic light
- Rotten Tomatoes: Given a movie review, detect whether the reviewer to like the movie or dislike it

Question: Provide two more examples of classification problems.

Regression

In regression, our model aims to examine a data point and output a real value. Here are some examples of regression:

- Zillow: Given the square footage, number of bedrooms, and number of bathrooms in a home, predict its sale price

- LinkedIn: Given a person's education level, age, home city, and industry, predict their salary

Question: Provide two more examples of regression problems.

Clustering

In clustering, we are given a set of data points and we want to group them together.

- Google Maps: Given a set of geotagged (i.e. includes latitude/longitude) photos, identify which photos were probably taken at the same landmark

Question: Provide two more examples of clustering problems.

Dimensionality Reduction

In dimensionality reduction, we want to shrink a data point down into a smaller representation.

- Google Translate: Represent an arbitrary sentence in the English language with a short vector of numbers
- Reddit: Compress images on a subreddit so they don't take up too much space.

Question: Provide two more examples of dimensionality reduction.

Ranking

In ranking, our model examines a set of data points and puts them into an ordered sequence. Here are some examples of regression:

- Google: Given a set of documents and a search query, what order should the documents be shown?
- Blizzard: Given the stats, inventory, and armor of a set of players attacking a world boss, what order should the boss fight the players

Question: Provide two more examples of ranking problems.

Recommendation

In recommendation, given a set of (user, product, did buy) triples, suggest products for a given user.

- Amazon: Recommend products for a user to buy
- Netflix: Recommend movies that a user might want to see

Question: Provide two more examples of recommendation problems.

Supervised vs. Unsupervised Learning

We can group machine learning problems into two groups – supervised or unsupervised.

In supervised learning, we provided the training examples along with some label indicating the output we want the model to produce for the given training example.

In unsupervised learning, we provide the raw training examples and the model learns to find some sort of structure in the training examples.

Question: For each of the machine learning problems in the previous section (e.g. classification, regression), mark them as unsupervised or supervised.

Other Kinds of Machine Learning

There are many ways to extend machine learning beyond the problems described here.

For example, the situation where a model can interacts with its environment to maximize a reward function is called **reinforcement learning**. Models that play chess or Go use reinforcement learning.

If a model must make a prediction for a given training example before it receives the next training example (but the model does not change the environment producing the training examples), we want to use an **online learning algorithm**.

If a model is given an extremely small training set, we may want to use **one-shot learning algorithms**.