Machine Learning examples

Autonomous helicopter: an autonomous helicopter that uses learning to incorporate information about the environment (like wind) to plan its route.

Autonomous vehicle: hands-free car that needs to follow some dessert terrain route.

Many other fields: computer vision, NLP, games etc.

Why learn?

Some tasks are easy for humans, but hard to express in simple mathematical rules, such as face recognition.

Supervised learning:

- The systems receives a training data, for both "true" and "false" classes. For instance: faces of the desirable face to be recognized, and faces of others. Labeled data.
- Recognition of unseen instances.

Unsupervised learning:

- Only inputs are given, with no labels.
- Find a structure in the given set.
- For instance: clustering, anomaly detection.

Reinforcement learning:

- Train a system by giving it "rewards" when it does something good – kind of like human learning…
- For testing instances, given a data – if it is classified correctly, you are rewarded. The goal of the system will be to maximize the rewards. **Will not be covered**.

Examples of supervised learning:

Regression (curve fitting): given x, y points as data, fitting a curve to those points is regression. If we have an idea of how the data looks like, we can fit a function to the data and predicts other instances.

Classification: trying to infer the label of given data instances; for instance, face recognition. The classifier will be trained on "true" and "false" instances (i.e. labeled). It is a partition of the domain such that future unseen data is mapped to either part of the partition.

Examples of unsupervised learning:

Clustering: Will be used for high-dimensional universes. For instance, finding centroids for clustering.

Embedding: finding low-dimensional manifold near which the data live: learning what is the manifold, without having the mathematical representation of the manifold in advance.

Compression/Quantization: find a function that compresses the data such that each input can be reconstructed from it.

Learning $\neq$ Memorization:

Memorization can be useful when we have a large amount of data storage capacity, and want to make decisions fast. But a distance measurement will be hard (for instance, measuring differences of a given face image to the memorized data); also, a lot of data is required; lastly, accuracy is not guaranteed.

Paradigms:

Frequentist vs. Bayesian:

Later today.

Generative vs. Discriminative:

- Generative:?
- Discriminative: trying to find a classifier that separates the data into different classes.

Probability Theory

The Monty Hall problem:

We are given 3 doors in a game-show, 1 door with a car and 2 with goats.

We pick 1 door, and the host of th game-show opens another door, showing it has a goat.

What should we do? Change our decision? Or keep the original door we picked? **We should switch!**

Say we choose door 1, and a goat is revealed behind door 2.

Case 1: loose after the switch, i.e. the car was behind door #1. Probability: $\frac{1}{3}$ – we pick 1 out of 3 doors.

Case 2: win after switch, i.e. a goat was behind door #1 (since the host took care of the other goat). Probability: $\frac{2}{3}$ – as we have 2 goats behind 3 doors.

Therefore we should switch.

Variation: there are 100 doors, 99 goats, 98 of them are revealed – we should still switch.

Variation 2: 97/100 goats are revealed.

Variation 3: 1/100 goat is revealed.

General:

We have n doors, 1 car, $n-1$ goats and the host opens p doors. Should we switch?

My calculations:

$$\Pr[loose\ after\ switch] = \frac{1}{n}$$

$$\Pr[win\ after\ switch] = \Pr[choose\ goat] \cdot \Pr[switch\ to\ car \mid p\ goats\ are\ revealed] = \frac{n-1}{n} \cdot \frac{1}{n-(p+1)}$$

$$\frac{n-1}{n} \cdot \frac{1}{n-(p+1)} > \frac{1}{n} \Leftrightarrow n-1 > n-p-1 \Leftrightarrow p > 0 \Rightarrow$$

It is sufficient that the host opens 1 door to increase the odds of winning.

Class solution:

$$\Pr[loose\ after\ switch] = \Pr[choose\ car\ first] + \Pr[hit\ goat\ first] \cdot \Pr[hit\ goat\ again] = \frac{1}{n} + \frac{n-1}{n} \cdot \frac{n-p-2}{n-p-1}$$

The $n-p-2$ is $n-p-1$ left goats, minus the additional 1 that we hit first.

$$\Pr[win\ afer\ switch] = \Pr[hit\ goat\ first] \cdot \Pr[hit\ car\ next] = \frac{n-1}{n} \cdot \frac{1}{n-p-1}$$

Sanity check: the probabilities must sum up to 1:

$\frac{1}{n}+\frac{n-1}{n}\cdot\frac{n-p-2}{n-p-1}+\frac{n-1}{n}\cdot\frac{1}{n-p-1}=\frac{1}{n}+\frac{(n-1)(n-p-2+1)}{n-p-1}=1 \Rightarrow$ that's good.

Now we want $\Pr[win] > \Pr[loose]$, and that happens when:

$$\frac{n-1}{n}\cdot\frac{1}{n-p-1}>\frac{1}{n}+\frac{n-1}{n}\cdot\frac{n-p-2}{n-p-1} \Leftrightarrow n-1>n-p-1+(n-1)(n-p-2) \Leftrightarrow$$

$$p>(n-1)(n-p-2) \Leftrightarrow p>n^2-np-2n-n+p+2 \Leftrightarrow n^2-n(p+3)+2<0 \Leftrightarrow$$

$$n^2+2<n(p+3) \Leftrightarrow n+\frac{2}{n}<p+3 \Leftrightarrow p>n+\frac{2}{n}-3$$

If $n=100$, p has to be $p>100+\frac{2}{100}-3=97.02 \Rightarrow p \geq 98$.

That is the case where the host opens all doors except the one I chose and some other door.

Another way to look at the problem:

Case 1: loose **because** I switched.

Case 2: win **because** I switched.

$$\Pr[loose\ because\ I\ switch]=\Pr[originally\ chose\ the\ car]=\frac{1}{n}$$

$$\Pr[won\ because\ I\ switch]=\Pr[loose\ first]\cdot\Pr[then\ hit]=\frac{n-1}{n}\cdot\frac{1}{n-p-1} \Rightarrow$$

$$\Pr[win]>\Pr[loose] \Leftrightarrow \frac{n-1}{n}\cdot\frac{1}{n-p-1}>\frac{1}{n} \Leftrightarrow p>0$$

And this says that you should always switch, but these **don't sum up to 1** (we disregarded the cases in which we lost both choices).

We can look at the ratio $\frac{\Pr[win]}{\Pr[lose]}=\frac{n-1}{n-p-1}>1$ but when $n\to\infty$, this ratio $\to 1$, so it won't matter.

Another approach:

We can intuitively think that as goats are revealed, the probability the car is behind the door I initially chose immediately increases from $\frac{1}{n}$ to $\frac{1}{n-p}$.

A simple example:

Given 2 boxes (red, blue) with oranges and apples, denoted:

$B = \{r, b\}, F = \{a, o\}$

B and F are **random variables**.

Assume we pick red 40% of the time, then:

$p(B = r) = \frac{4}{10}, p(B = b) = \frac{6}{10}$ and they sum up to 1.

Consider two random variables $X \in \{x_i \mid i = 1, \dots, M\}, Y \in \{y_i \mid i = 1, \dots, L\}$ and we conduct N trials.

Sum rule:

$$p(X) = \sum_Y p(X, Y)$$

$$p(Y) \sum_X p(X, Y)$$

Marginalize the joint probability: marginal probability.

Product rule:

$p(X, Y) = p(Y|X) \cdot p(X) = p(X|Y) \cdot p(Y)$ – seeing X and then seeing Y given we saw X, or vice versa.

Bayes theorem:

$$p(X, Y) = p(Y|X) \cdot p(X) = p(X|Y) \cdot p(Y) \Rightarrow$$

$$p(Y|X) = \frac{p(X|Y) \cdot p(Y)}{p(X)} = \frac{p(X|Y) \cdot p(Y)}{\sum_Y p(X|Y) \cdot p(Y)}$$

When X, Y are independent:

$p(Y|X) = p(Y)$ and vice versa $\Rightarrow p(X, Y) = p(X) \cdot p(Y)$

<u>Back to B, F</u>:

Say $p(B = r) = \frac{4}{10}, p(B = b) = \frac{6}{10}$.

We choose a fruit and it turns out to be an orange. What is the probability of the box being blue?

Red box: 2 apples, 6 oranges.

Blue box: 1 orange, 3 apples.

Total oranges: 7

Total apples: 5

$$P(B = b|F = o) = \frac{p(F = o|B = b) \cdot p(B = b)}{p(F = o)} = \frac{p(F = o|B = b) \cdot 0.4}{p(F = o)} = \frac{1}{4} \cdot \frac{6}{10} \cdot \frac{20}{9}$$

Probability of choosing an apple:

$$p(F = a) = \sum_B p(F = a, B) = \sum_B p(F = a|B) p(B) = p(F = a|B = r) p(B = r) + \cdots$$

Probability densities:

Continuous random variables:

$$p(x \in (a,b)) = \int_a^b p(x)dx$$

$$p(x) \geq 0, \int_{-\infty}^{\infty} p(x)dx = 1$$

Change of variable $x = g(y)$:

$$p_y(y) = p_x(x)\left|\frac{dx}{dy}\right| = p_x(g(y))|g'(y)|$$

Cumulative distribution function:

$$P(z) = \int_{-\infty}^{z} p(x)dx$$

$$\frac{dP}{dx} = p(x)$$

It means that given a small range (a,b), then $p(x \in (a,b)) = \int_a^b px(dx)$

$P(z)$ is the function of area under the probability, so the probability p is the derivative of P.

For a multiple continuous variable: $p(x) = p(x_1, \dots, x_D), x = \{x_i | i = 1, \dots, D\}$:

$$p(x) \geq 0, \int p(x)dx = 1$$

All previous rules apply: sum, product and Bayes.

$$p(x) = \int p(x,y)dy$$

Etc.

Expectation:

The average value (mean) of a function $f(x)$ under a probability distribution $p(x)$:

Discrete case: $E[f] = \sum_x f(x)p(x)$

Continuous case: $E[f] = \int f(X)p(x)dx$

For N discrete data points: $E[f] \cong \frac{1}{N}\sum_{n=1}^{N} f(x_n)$

Conditional expectation:

Discrete case: $E_x[f|y] = \sum_x f(x)p(x|y)$

And the cont. is similar.

Variance:

The squared difference from the mean:

$$var[f] = E[(f(x) - E[f(x)])^2] = E[f(x)^2] - E[f(x)]^2$$

$$var[x] = E[x^2] - E[x]^2$$

Expectation rules:

- Monotonicity: $X \geq Y \Rightarrow E[X] \geq E[Y]$
- Linearity:
  - $E[X + c] = E[X] + c$
  - $E[X + Y] = E[X] + E[Y]$
  - $E[aX] = aE[X]$

Now:

$var[f(X)] = E[(f(X) - E[f(x)])^2] = E\left[f^2(X) - 2f(X)E[f(X)] + E^2[f(X)]\right] =$

$E[f^2(X)] - 2E[f(x)]E[f(X)] + E[f(X)]^2 = \boxed{E[f^2(X)] - E[f(X)]^2}$

Covariance:

The extent to which two variables vary together:

$var[X,Y] = E[(X - E[X])(Y - E[Y])] = E[XY] - E[X]E[Y]$ – **we should derive this at home**.

Following is the derivation:

$E[(X - E[X])(Y - E[Y])] = E[XY - XE[Y] - YE[X] + E[X]E[Y] =]$

$E[XY] - E[X]E\big[E[Y]\big] - E[Y]E\big[E[X]\big] + E\big[E[X]E[Y]\big] = E[XY] - E[X]E[Y] - E[Y]E[X] + E[X]E[Y] =$

$E[XY] - E[X]E[Y]$