The sun and cloud symbols above are fun but also point to a set of examples people have used in textbooks about machine learning (this also means learning from data). Here's a version of this classic example:

Soccer exercise: You are trying to figure out if a soccer game will happen today. Last year, you know the following happened:

Temperature	Rain	Humidity	Play
90	no	high	no
73	no	low	yes
81	yes	high	no
67	no	high	yes
72	yes	high	yes
77	no	low	yes
96	no	low	no
81	yes	high	no
58	yes	high	yes
72	no	medium	yes

- On Saturday, the forecast is for 90 degrees, high humidity, and no rain. Using your intuition and the data above, do you think the game will happen? Draw a start next to any data you used.

  no

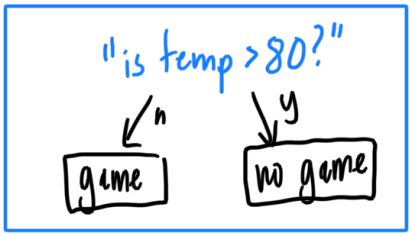
- On Sunday, the forecast is for 70 degrees, low humidity, and rain. Using your intuition and the data above, do you think the game will happen? Draw a smilie face next to any data you used.

  yes

- Based on the data above, make a decision tree that tells you if a game will happen, based on the weather. Start by boxing all 'play = yes' records. What question could you ask that would separate these records from the unboxed records?

EDA

- unboxed mainly high humidity
- no clear division on rain
- boxed all have temps under 80* 0



Compare your decision tree with your neighbor. How can you tell if one decision tree is better or worse than another?

correctly classified records in each box

- Looking at the soccer game data, what do you think is the most important factor in determining whether or not a game occurs?

  temperature (then maybe humidity & lastly rain)

- How does your answer to the previous question affect how you could construct a decision tree?

  gives evidence to support tree

- Can you think of a way to use the data to show mathematically that your answer to which factor is the most important is correct?

  some sort of measure of how good boxes classify records within them and any penalty for getting them wrong.

Deciding on the Decision Tree

We've run into an important question: how can we be sure that we're constructing the best possible decision tree? We've seen that we want the "most important question" to be at the top of our tree, but how can we determine what this is? The "most important question" will be the one that reduces uncertainty the most. In other words which question, when asked, would result in the most informative split (i.e. better odds than an even split). We will be using the soccer game example to illustrate.

- Looking at the days with high humidity, how many games were played? How many games weren't played? Based on the data, if the humidity is high, what is the probability that the game is played?

  6 days high humidity : 3 yes & 3 no → P(game played & high humidity) = 3/6 = 1/2

- Looking at the days with low humidity, how many games were played? How many games weren't played? Based on the data, if the humidity is low, what is the probability that the game is played?

  3 days low humidity : 2 yes & 1 no → P(game played & low humidity) = 2/3

- Based on your answers above, which situation reduces uncertainty more, if the humidity is low or if it's high? Why?

  low humidity since more info 2/3 > 1/2.

Now, we'll look at a couple of ways to mathematically measure uncertainty: Gini impurity and Entropy. Either of these can be used to choose an optimal decision tree, and they may result in different models.

Gini Impurity: To find this, we need to answer the question: if we guess whether or not a game was played, what is the probability that we're wrong? Let's look at when the temperature is above 75°.

Count the number of times when this occurs: 5. If the temperature is above 75°,the number of times a game played is 1 which means the probability that a game was played when the temperature is above 75° is 1/5. If the temperature is above 75°, the probability that a game was not played is 4/5.

Let p be the probability that a game is played in temperatures over 75°.
Then the Gini impurity given by $p(1-p)+(1-p)p$ is $\frac{1}{5}\left(\frac{4}{5}\right)\times 2 = \frac{8}{25}$

↳ or $p(1-p)\cdot 2$

- If you have a set of five games which are all played, what is the Gini impurity? Why does this make sense?

  p=1 → 1(0) + 0(1) = 0 ← impurity ← no uncertainty bc ALL games played

- If the humidity is high, what is the probability that the game is played? What is the Gini impurity?

  p: game played & high humidity → p = 1/2 → 1/2(1/2)·2 = 2/4 = 1/2 impurity

- If the humidity is low, what is the probability that the game is played? What is the Gini impurity?

  p: game played & low humidity → p = 2/3 → 2/3(1/3)·2 = 4/9 impurity

- In which situation is there more uncertainty: when the humidity is high, or when it is low?

  1/2 > 4/9 → more uncertainty when high humidity since larger impurity

- If the humidity is *not* high (so it's either low or medium), what is the probability that the game is played? What is the Gini impurity? # low/medium humidity days: 4 # games played: 3

  P(game played & not high humidity) = 3/4 → 3/4(1/4)·2 = 6/16 = 0.375 impurity

- If the humidity is *not* low (so it's either medium or high), what is the probability that the game is played? What is the Gini impurity? # medium/high days: 7 # games played: 4

  P(game played & not low humidity) = 4/7 → 4/7(3/7)·2 = 24/49 ≈ 0.48 impurity

- In which situation is there more uncertainty: when the humidity is not high, or when it is not low?

  not low since impurity 24/49 > 6/16

Now, let's look at how we can use Gini impurity to measure how "good" a split is. This will require a few steps.

Find the Gini impurity of the entire set (before the split). We would like to construct our decision tree in the way that reduces uncertainty the most. First, we need to consider Gini impurity of the entire system. Ignoring the weather, 6 games were played and 4 games were not. So the probability of a game being played is $\frac{6}{10}$. The Gini impurity is then

$$\frac{6}{10} \times \frac{4}{10} + \frac{4}{10} \times \frac{6}{10} = \frac{48}{100} = 0.48.$$

Impurity Total = P(play)·P(no play)·2

That's a lot of uncertainty! After a split in the tree, we'll measure the uncertainty by computing the Gini impurity of each branch, and averaging the impurities.

Find the Gini impurities of the branches (after the split).
Let's start by splitting on whether or not the humidity is high. If the humidity is high, 3 games were played and 3 games were not. This means that if the humidity is high, then the probability that a game is played is $\frac{1}{2}$, and the Gini impurity is

$$\frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times \frac{1}{2} = \frac{1}{2} = 0.5.$$

Impurity feature=yes = P(play & feature=yes)·P(no play & feature=yes)·

Next we need to find the Gini impurity of the other branch, which is when the humidity is not high. If the humidity was low or medium, 3 games were played, and 1 was not. So, if the humidity is not high, then the probability that a game is played is $\frac{3}{4}$, and the Gini impurity is

$$\frac{1}{4} \times \frac{3}{4} + \frac{3}{4} \times \frac{1}{4} = \frac{6}{16} = 0.375.$$

Impurity feature=no = P(play & feature=no)·P(no play & feature=no)·

Find the weighted average of the Gini impurities of the branches.
Now, we need to average our two impurities. However, we need to take into account that it's more likely that humidity is high than that it's not. Because of this, we compute the weighted average, using the probability that the humidity is high, $\frac{6}{10}$, and the probability that it's not, $\frac{4}{10}$. This weighted average is

$$\frac{6}{10} \times 0.5 + \frac{4}{10} \times 0.375 = 0.45.$$

Impurity feature = P(feature=yes)·Impurity feature=yes + P(feature=no)·Impurity feature=no

Finally, we need to compare this result with the Gini impurity of the entire set.

Take the difference between impurity of set before and after the split.
From this weighted average, we can see that splitting on high humidity decreases Gini impurity by

$$0.48 - 0.45 = 0.03.$$

So we've decreased uncertainty by a bit, but hopefully we can do better! Let's try some different splits to try to find the best one.

- What is the decrease in Gini impurity if you split on whether or not the humidity is low?

feature split

3 days feature = yes (low humid) → Impurity feature = yes (low humidity): P(feature yes & target yes) · P(feature yes target no) · 2 → (2/3)(1/3)·2 = 4/9
(target yes & feature yes (low humid) play = 2)

(1)

7 days feature = no (not low) → Impurity feature = no (not low): P(feature no & target yes) · P(feature no target no) · 2 → (4/7)(3/7)·2 = 24/49
(target yes & feature no (not low) play = 4)

(2)

(3) Impurity feature ("is humidity low?"): P(feature = yes (low)) · Impurity feature = yes (low) + P(feature = yes (not low)) · Impurity feature = yes (not low)
→ 3/10 (4/9) + 7/10 (24/49) → 0.47619 impurity

(4) Impurity TOTAL − Impurity FEAT
= 0.48 − 0.47619
= 0.0038
decrease in total impurity

- What is the decrease in Gini impurity if you split on whether or not it rains?

4 days rain & played → impurity rains = (2/4)(2/4)·2 = 1/2

6 days no rain & played → impurity no rain = (4/6)(2/6)·2 = 24/36 = 4/9

impurity feature = 4/10 (1/2) + 6/10 (4/9) = 0.46̄
("will it rain?")

decrease: 0.48 − 0.46̄ = 0.01̄3̄

- Now beyond if it's raining, or if humidity is high, or if humidity is low, what is the question that, when asked, will give us the most ordered feature space? Hint: look at the very first decision tree we drew on the first page of this handout. Calculate the decrease in Gini impurity on that split.

4 days games → temp > 80 : (0/4)(4/4)(2) = ∅

6 days 6 games → temp ≤ 80 : (6/6)(0/6)(2) = ∅

impurity feature = 4/10 (0) + 6/10 (0) = ∅
("is temp > 80?")

decrease: 0.48 − 0 = 0.48
huge! decrease

Once we find the split that reduces uncertainty the most, we can repeat this process for each of our branches, further reducing the uncertainty. Fortunately, we don't need to do this by hand! We'll be able to use Python to quickly construct decision trees.

Another way to mathematically measure uncertainty is with *entropy*. We'll repeat the work that we did with Gini impurity, but this time use entropy instead.

Entropy: in mathematics, uncertainty is measured using something called *entropy*. You might have heard of entropy in physics, or some other context. Entry is used to describe uncertainty in many different contexts, including the disorder in the universe!. If the probability of an event happening is p, then the probability of the same event *not* happening is $1-p$, and the *entropy* of this split is

$$-p\log_2(p)-(1-p)\log_2(1-p).$$

For example, let's look at when the temperature is above 75°. In this case, one game was played and four were not, so the probability of a game being played is $\frac{1}{5}$. So, the entropy is

$$-\frac{1}{5}\log_2\left(\frac{1}{5}\right)-\left(1-\frac{1}{5}\right)\log_2\left(1-\frac{1}{5}\right)\approx 0.7219.$$

Entropy is always between 0 and 1, and larger values correspond to more uncertainty.

- If the humidity is high, what is the probability that the game is played? What is the entropy?

  $p=1/2$ $\quad -1/2\log_2(1/2)-(1-1/2)\log_2(1-1/2)=1$

- If the humidity is low, what is the probability that the game is played? What is the entropy?

  $p=2/3$ $\quad -2/3\log_2(2/3)-(1/3)\log_2(1/3)\approx 0.9183$

- In which situation is there more uncertainty: when the humidity is high, or when it is low?

  larger entropy

- If the humidity is *not* high (so it's either low or medium), what is the probability that the game is played? What is the entropy?

  $p=3/4$ $\quad -3/4\log_2(3/4)-1/4\log_2(1/4)\approx 0.8193$

- If the humidity is *not* low (so it's either medium or high), what is the probability that the game is played? What is the entropy?

  $p=4/7$ $\quad -4/7\log_2(4/7)-3/7\log_2(3/7)\approx 0.98152$

- In which situation is there more uncertainty: when the humidity is not high, or when it is not low?

  larger entropy value.

We would like to construct our decision tree in the way that reduces uncertainty the most. First, we need to consider entropy of the entire system. Ignoring the weather, 6 games were played and 4 games were not. So the probability of a game being played is $\frac{6}{10}$. The entropy is then

$$-\frac{6}{10}\log_2\left(\frac{6}{10}\right)-\left(1-\frac{6}{10}\right)\log_2\left(1-\frac{6}{10}\right)\approx 0.9710.$$

That's a lot of uncertainty!

We want to figure out what question will split the data in the way that reduces uncertainty the most. We'll measure the uncertainty by computing the entropy of each branch, and averaging the entropies.

Let's start by splitting on whether or not the humidity is high. If the humidity is high, 3 games were played and 3 games were not. This means that if the humidity is high, then the probability that a game is played is $\frac{1}{2}$, and the entropy is

$$-\frac{1}{2}\log_2\left(\frac{1}{2}\right)-\left(1-\frac{1}{2}\right)\log_2\left(1-\frac{1}{2}\right)=1.$$

Next we need to find the entropy of the other branch, which is when the humidity is not high. If the humidity was low or medium, 3 games were played, and 1 was not. So, if the humidity is not high, then the probability that a game is played is $\frac{3}{4}$, and the entropy is

$$-\frac{3}{4}\log_2\left(\frac{3}{4}\right)-\left(1-\frac{3}{4}\right)\log_2\left(1-\frac{3}{4}\right)\approx 0.8113.$$

Now, we need to average our two entropies. However, we need to take into account that it's more likely that humidity is high than that it's not. Because of this, we compute the weighted average, using the probability that the humidity is high, $\frac{6}{10}$, and the probability that it's not, $\frac{4}{10}$. This weighted average is

$$\frac{6}{10}\cdot 1+\frac{4}{10}\cdot 0.8113=0.925.$$

From this weighted average, we can see that splitting on high humidity decreases entropy by

$$0.9710-0.925=\boxed{0.046.}$$

So we've decreased uncertainty by a bit, but hopefully we can do better! Let's try some different splits to try to find the best one.

- What is the decrease in entropy if you split on whether or not the humidity is low?

3 days, 2 play → humidity low, p = 2/3 → entropy ≈ 0.9183

7 days, 4 play → humidity not low, p = 4/7 → entropy ≈ 0.9852

entropy$_{\text{feature}}$ "is humidity low" $= \frac{3}{10}(0.9183)+\frac{7}{10}(0.9852) = \boxed{0.9651}$

decrease: $0.9710-0.9651=\boxed{0.0059}$

- What is the decrease in entropy if you split on whether or not it rains?

4 days, 2 play → rain, p = 2/4 → entropy → 1

6 days, 4 play → no rain, p = 4/6 → entropy → ≈ 0.9183

entropy$_{\text{feature}}$ "is it raining?" $= \frac{4}{10}(1)+\frac{6}{10}(0.9183) = \boxed{0.95098}$

decrease: $0.9710-0.951=\boxed{0.0202}$

- Can you find the split that decreases entropy the most?

4 days, 0 play → temp > 80, p = 0 → entropy → 0

6 days, 6 play → temp < 80, p = 1 → entropy → 0

entropy$_{\text{feature}}$ "is temp > 80?" $= \frac{4}{10}\cdot 0+\frac{6}{10}\cdot 0 = \boxed{0}$

decrease: $0.9710-0=\boxed{0.9710}$

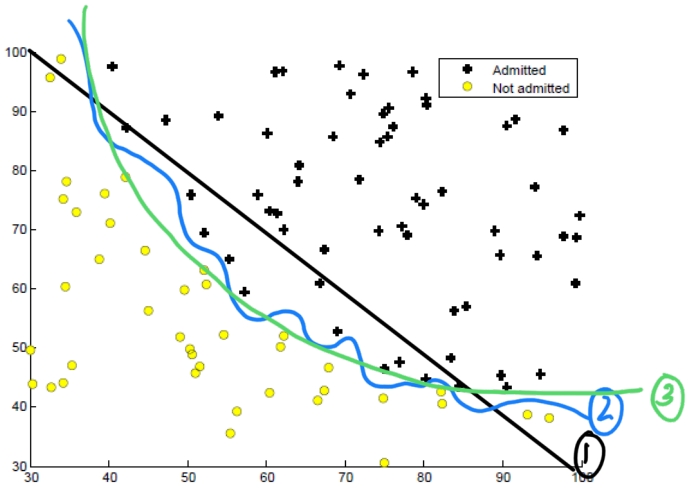
Once we find the split that reduces uncertainty the most, we can repeat this process for each of our branches, further reducing the uncertainty. Fortunately, we don't need to do this by hand! We'll be able to use python to quickly construct decision trees.

Goldilocks and The Three (Tree) Algorithms

We've seen that *overfitting* happens when a model is too specialized to the training data, so that it learns noise or outliers in the data, but has poor performance on new data.

On the other side, *underfitting* happens when a model isn't specialized enough to the training data, so that it doesn't perform as well as it could on either the training data or on new data.

The following plot shows data points with two scores, and the decision of whether or not each student was admitted to a program.



Draw a decision boundary between admitted and not admitted points, which gives an examples of:
(1) underfitting, (2) overfittting, and (3) an appropriate fit.

Let's look at another example. The following plot show accuracy measure for several values of a parameter p. Circle on the plot the values of k where overfitting, underfitting, and an appropriate model occurs. Discuss in a group: did you get the same values?