Daniel Adalaid

Mayumi Luque

Let's address the elephant in the room and say it aloud, "We're not as bright as we think we are." There, done. That's the biggest disadvantage of writing a program on your own instead of using one that was created by a group of people with more experience, time, and whose job is *literally* to create those programs. We, simple students, don't have the time to invest to write our own programs for everything (expect for this class, I guess…) However, there's also the glaring issue that the team of people that are developing the tools we use aren't as bright as they think, (We're looking at Android Studio.) There might be bugs, errors, mistakes, however you want to call it. There's also the probability that a function you *really need* is not included in the tool. So, the biggest advantage you have when writing your own programs is that it's *yours*. You say what goes in it and what doesn't. You want it to read your inputs in a certain way? Go ahead. You want it to insult you each time it processes information? The sky and your programming abilities are the limit, you masochistic person.

The dataset that we chose was on Molecular Biology. It depicted the splice-junction of Gene Sequences. We chose this because it was on the smaller side and it made it easier to translate into our program's input. We also used the weather dataset that already comes with Weka for simplicity's sake. Apart from that, we also used the tiny dataset from the examples provided in Alphagrader.

The trees created by both our programs and Weka can be observed in figures 1 and 2. The trees are noticeably different. Though, Weka's tree is more detailed, giving you all the options, instead of just one. Besides that, both trees reach the same conclusions. So, the algorithms used might be similar, but the presentation is different.

```
            outlook
            temperature
            humidity
            windy
            play
Test mode:    evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree
------------------

outlook = sunny
|   humidity = high: no (3.0)
|   humidity = normal: yes (2.0)
outlook = overcast: yes (4.0)
outlook = rainy
|   windy = TRUE: no (2.0)
|   windy = FALSE: yes (3.0)

Number of Leaves  :     5

Size of the tree :      8
```

Figure 1. Weka Tree

```
mayumil:~/workspace/decision_trees (mast
temperature: hot
  outlook: sunny
    ANSWER: no
  outlook: overcast
    ANSWER: yes
  outlook: rainy
```

Figure 2. Our Tree

Decision trees can be used in various fields, business, analytics, videogames, and many others. As avid gamers, the first thing we thought about was videogames. Of course decision trees would not be the perfect fit for enemies in action games, nor platformers. But if would be a perfect fit for visual novels and horror games. For visual novels, so you can experience a better game where the options aren't hard programmed, but

you can actually have a conversation with the characters (We’re sad people, okay?) And in horror games because can you imagine an AI that can learn how to scare you? Actually, don’t. There’s an episode of Black Mirror that’s *terrifying*.