# An irrigation machine

You're going to automate the watering of parcels by making an intelligent irrigation machine. Multi-label classification problems differ from multi-class problems in that each observation can be labeled with zero or more classes. So classes are not mutually exclusive.

To account for this behavior what we do is have an output layer with as many neurons as classes but this time, unlike in multi-class problems, each output neuron has a `sigmoid` activation function. This makes the output layer able to output a number between 0 and 1 in any of its neurons.

Keras `Sequential()` model and `Dense()` layers are preloaded. It's time to build an intelligent irrigation machine!
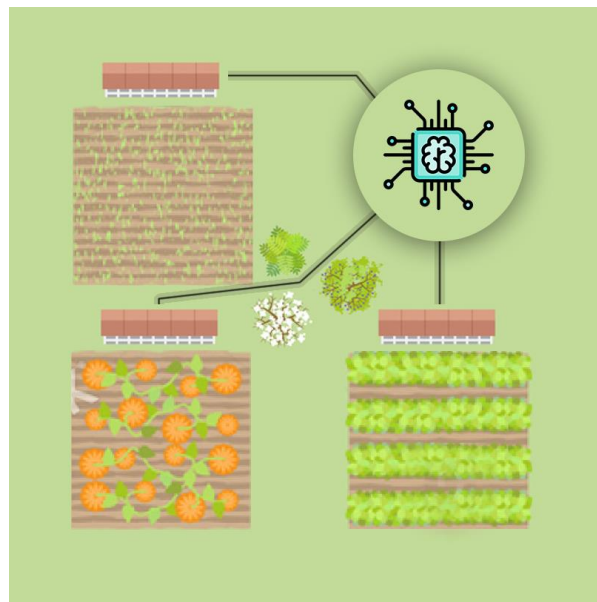
- Instantiate a `Sequential()` model.
- Add a hidden layer of 64 neurons with as many input neurons as there are sensors and `relu` activation.
- Add an output layer with as many neurons as parcels and `sigmoid`activation.
- Compile your model with `adam` and `binary_crossentropy` loss.

```python
# Instantiate a Sequential model
model = Sequential()


# Add a hidden layer of 64 neurons and a 20 neuron's input
model.add(Dense(64, input_shape=(20,), activation='relu'))


# Add an output layer of 3 neurons with sigmoid activation
model.add(Dense(3, activation='sigmoid'))


# Compile your model with adam and binary crossentropy loss
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])


model.summary()
```