



QUESTION 1 OF 2

You are trying to create a classifier that labels an animal a **cat** or **dog** based on their height, weight, and fur length. Here is an example of what the data looks like:

height	weight	fur_length	animal
40	50	2	dog
10	5	5	cat
30	25	10	dog
15	10	3	cat

Given the scenario above, why may you *not* want to use XGBoost as your first model of choice?

- You would solve the problem easily and have nothing else to do.
- Why use XGBoost when you can go straight into using deep learning.

While powerful, XGBoost has many hyperparameters which may need to be tuned extensively before finding suitable performance.

SUBMIT

QUESTION 2 OF 2

Have a look at the python code below:

```
dtrain = xgb.DMatrix(df_train_features, label=df_train_target)
param = {"max_depth": 10, "eta": 1, "objective": "multi:softmax", "num_class": 2}
num round = 100
```