

- Information for entire dataset: 1.53
- Information for meat: 0.98, information gain: 0.55
- Information for crust: 1.42, information gain: 0.11
- Information for veggies: 1.29, information gain: 0.24

```
import numpy as np
```

```
# Find information and gain for vegetables
```

```
# Do yes attribute first
```

```
prob_yes_veg = 4/9
```

```
prob_bad_yes_veg = 0/4
```

```
prob_good_yes_veg = 2/4
```

```
prob_great_yes_veg = 2/4
```

```
info_yes_veg = -prob_yes_veg*(np.nan_to_num(prob_bad_yes_veg*np.log2(prob_
info_yes_veg
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:10: RuntimeWarni
```

```
# Remove the CWD from sys.path while we load stuff.
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:10: RuntimeWarni
```

```
# Remove the CWD from sys.path while we load stuff.
```

```
0.4444444444444444
```

```
# Now do no attribute
```

```
prob_no_veg = 5/9
```

```
prob_bad_no_veg = 2/5
```

```
prob_good_no_veg = 2/5
```

```
prob_great_no_veg = 1/5
```

```
info_no_veg = -prob_no_veg*(prob_bad_no_veg*np.log2(prob_bad_no_veg) + pro
info_no_veg
```

```
0.8455156082707568
```

```
info_veg = info_yes_veg + info_no_veg
```

```
info_gain_veg = 1.53 - info_veg
```

```
info_gain_veg
```

```
0.24003994728479872
```

The meat feature has the highest information gain and should therefore be what we split on first. Now we will find the best attribute for the yes node branching off of meat. The information for yes_meat is 0.54.

```
# Do the crust feature first
prob_thin_crust_no_meat = 2/3
prob_bad_thin_crust_no_meat = 1/2
prob_good_thin_no_meat = 1/2

info_thin_crust = -prob_thin_crust_no_meat*(prob_bad_thin_crust_no_meat*np
info_thin_crust

0.6666666666666666

prob_deep_no_meat = 1/4
prob_bad_deep_no_meat = 1/1

info_deep_crust = -prob_deep_no_meat*(prob_bad_deep_no_meat*np.log2(prob_b
info_deep_crust

-0.0

prob_stuffed_no_meat = 1/2
prob_good_stuffed_no_meat = 1/1

info_stuffed_crust = -prob_stuffed_no_meat*(prob_good_stuffed_no_meat*np.l
info_stuffed_crust

-0.0

info_crust_no_meat = info_thin_crust + info_deep_crust + info_stuffed_crus
info_crust_no_meat

0.6666666666666666
```

```
# Do the veggie feature now
prob_no_veg_no_meat = 2/4
prob_bad_no_veg_no_meat = 2/2
```

```
info_no_veg_no_meat = -prob_no_veg_no_meat*(prob_bad_no_veg_no_meat*np.log
info_no_veg_no_meat
-0.0
```

```
prob_yes_veg_no_meat = 2/4
prob_good_no_veg_no_meat = 2/2
```

```
info_yes_veg_no_meat = -prob_yes_veg_no_meat*(prob_good_no_veg_no_meat*np.
info_yes_veg_no_meat
-0.0
```

```
info_veg_no_meat = 0.0
```

The veggie feature has the least entropy for the no meat branch, so that branch should lead to the veggie feature.

