In[∘]:=
```
file = File["\\\\vmware-host\\Shared Folders\\C\\Users\\peter\\Documents\\semesters-at-marshall\\2024-fall-cs-515-101\\2024-W38-2.xlsx"]
```

Out[∘]=

File[\\vmware-host\Shared Folders\C\Users\peter\Documents\semesters-at-marshall\2024-fall-cs-515-101\2024-W38-2.xlsx]

In[∘]:=
```
data = Import[file, {"Dataset", 1}, "HeaderLines" → {1, 1}]
```

Out[∘]=

	male?	car_type	shirt_size	class
1.0	True	family	small	0.0
2.0	True	sports	medium	0.0
3.0	True	sports	medium	0.0
4.0	True	sports	large	0.0
5.0	True	sports	extra large	0.0
6.0	True	sports	extra large	0.0
7.0	False	sports	small	0.0
8.0	False	sports	small	0.0
9.0	False	sports	medium	0.0
10.0	False	luxury	large	0.0
11.0	True	family	large	1.0
12.0	True	family	extra large	1.0
13.0	True	family	medium	1.0
14.0	True	luxury	extra large	1.0
15.0	False	luxury	small	1.0
16.0	False	luxury	small	1.0
17.0	False	luxury	medium	1.0
18.0	False	luxury	medium	1.0
19.0	False	luxury	medium	1.0
20.0	False	luxury	large	1.0

In[∘]:=

$$\text{RelativeFrequency[list\_]} := \frac{\text{Counts[list]}}{\text{Total[Counts[list]]}}$$

For example,

In[•]:=
```
data[Counts, "class"]
```
Out[•]=

0.0	10
1.0	10

In[•]:=
```
Normal[Values[data[Counts, "class"]]]
```
Out[•]=
```
{10, 10}
```

In[•]:=
```
RelativeFrequency[Normal[Values[data[All, "class"]]]]
```
Out[•]=

$\langle\left| 0. \to \frac{1}{2},\ 1. \to \frac{1}{2} \right|\rangle$

In[•]:=
```
Table[term^2, {term, RelativeFrequency[Normal[Values[data[All, "class"]]]]}]
```
Out[•]=

$\left\{\frac{1}{4}, \frac{1}{4}\right\}$

In[•]:=
```
Sum[term^2, {term, RelativeFrequency[Normal[Values[data[All, "class"]]]]}]
```
Out[•]=

$\frac{1}{2}$

In[•]:=
```
1 - Sum[term^2, {term, RelativeFrequency[Normal[Values[data[All, "class"]]]]}]
```
Out[•]=

$\frac{1}{2}$

In[•]:=
```
N[1 - Sum[term^2, {term, RelativeFrequency[Normal[Values[data[All, "class"]]]]}]]
```
Out[•]=
```
0.5
```

The initial Gini index is 0.5.

In[•]:=
```
FromRelativeFrequencyToGiniIndex[list_] := 1 - Sum[term^2, {term, list}]
```

In[•]:=
```
GiniIndex[<|"Data" → data_, "Attribute" → attribute_, "outcome" → outcome_|>] :=
  (Values[RelativeFrequency[Normal[Values[data[All, attribute]]]]]).
   (FromRelativeFrequencyToGiniIndex /@ Table[Values[RelativeFrequency[
         Normal[Values[data[Select[Slot[attribute] == value &], outcome]]]]],
       {value, Normal[Values[data[DeleteDuplicates, attribute]]]}])
```

Let's compute the exact and the numeric values for the different types.

In[•]:=
```
{#, N[#]} &@GiniIndex[<|"Data" → data, "Attribute" → "male?", "outcome" → "class"|>]
```
Out[•]=

$\left\{\frac{12}{25}, 0.48\right\}$

In[•]:=
```
{#, N[#]} &@GiniIndex[<|"Data" → data, "Attribute" → "car_type", "outcome" → "class"|>]
```

Out[•]=

$\left\{\frac{13}{80}, 0.1625\right\}$

In[•]:=
```
{#, N[#]} &@GiniIndex[<|"Data" → data, "Attribute" → "shirt_size", "outcome" → "class"|>]
```

Out[•]=

$\left\{\frac{86}{175}, 0.491429\right\}$

The following function will calculate which one we want.

In[•]:=
```
CalculateAttributeToSplitOnByGiniIndex[<|"Data" → data_, "outcome" → outcome_|>] :=
 First[Keys[Sort[
    AssociationMap[GiniIndex[<|"Data" → data, "Attribute" → #, "outcome" → "class"|>] &,
     DeleteCases[First[Normal[data[DeleteDuplicates, Keys]]], "class"]]]]]
```

In[•]:=
```
CalculateAttributeToSplitOnByGiniIndex[<|"Data" → data, "outcome" → "Class"|>]
```

Out[•]=

```
car_type
```

We can see from above that car type has the smallest impurity at 0.1625 so that's what we want to split on.

In[•]:=
```
FromRelativeFrequencyToEntropy[list_] :=
 Sum[-term * Log[2, term], {term, RelativeFrequency[list]}]
```

In[•]:=
```
FromRelativeFrequencyToEntropy[{4, 6}]
```

Out[•]=

```
1
```

In[•]:=
```
Table[-term * Log[2, term], {term, RelativeFrequency[{4, 6}]}]
```

Out[•]=

$\left\{\frac{1}{2}, \frac{1}{2}\right\}$

In[•]:=
```
Table[-term * Log[term], {term, RelativeFrequency[{4, 6}]}]
```

Out[•]=

$\left\{\frac{\text{Log}[2]}{2}, \frac{\text{Log}[2]}{2}\right\}$

In[•]:=
```
Total[Table[-term * Log[term], {term, RelativeFrequency[{4, 6}]}]]
```

Out[•]=

```
Log[2]
```

My function produces the same output as the built in entropy function.

In[•]:=
```
Entropy[2, Normal@Values@data[All, "class"]]
```

Out[•]=

$-\frac{\text{Log}[10]}{\text{Log}[2]} + \frac{\text{Log}[20]}{\text{Log}[2]}$

In[ ]:=
```
FullSimplify[Entropy[2, Normal@Values@data[All, "class"]]]
```
Out[ ]=
```
1
```

In[ ]:= $N\left[-\frac{\text{Log}[10]}{\text{Log}[2]}+\frac{\text{Log}[20]}{\text{Log}[2]}\right]$

Out[ ]=
```
1.
```

In[ ]:=
```
FromRelativeFrequencyToEntropy[Normal@Values@data[All, "class"]]
```
Out[ ]=
```
1
```

In[ ]:=
```
Information[Entropy]
```
Out[ ]=

Symbol

Entropy[*list*] gives the base *e* information entropy of the values in *list*.

Entropy[*k*, *list*] gives the base *k* information entropy.

Documentation Local » | Web »

Options SameTest → Automatic

Attributes {Protected, ReadProtected}

Full Name System`Entropy

In[ ]:=
```
calculateEntropy[<|"Data" → data_, "Attribute" → attribute_, "outcome" → outcome_|>] :=
  (Values@RelativeFrequency[Normal[Values[data[All, attribute]]]]).
   (FromRelativeFrequencyToEntropy /@ Table[Values@RelativeFrequency[
        Normal[Values[data[Select[Slot[attribute] == value &], outcome]]]],
      {value, Normal[Values[data[DeleteDuplicates, attribute]]]}])
```

In[ ]:=
```
{#, N[#]} &@calculateEntropy[<|"Data" → data, "Attribute" → "male?", "outcome" → "class"|>]
```
Out[ ]=
```
{1, 1.}
```

In[ ]:=
```
{#, N[#]} &@
 calculateEntropy[<|"Data" → data, "Attribute" → "car_type", "outcome" → "class"|>]
```
Out[ ]=

$\left\{\frac{3}{5}, 0.6\right\}$

In[ ]:=
```
{#, N[#]} &@
 calculateEntropy[<|"Data" → data, "Attribute" → "shirt_size", "outcome" → "class"|>]
```
Out[ ]=

$\left\{\frac{3}{5}, 0.6\right\}$

```
In[ ]:= CalculateAttributeToSplitOnByEntropy[<|"Data" → data_, "outcome" → outcome_|>] :=
         First[Keys[Sort[AssociationMap[
             calculateEntropy[<|"Data" → data, "Attribute" → #, "outcome" → "class"|>] &,
             DeleteCases[First[Normal[data[DeleteDuplicates, Keys]]], "class"]]]]]

In[ ]:= CalculateAttributeToSplitOnByEntropy[<|"Data" → data, "outcome" → "Class"|>]
Out[ ]=
        car_type
```

Both gini index and entropy say split on car type.

The entropy at the beginning is.

```
In[ ]:= RelativeFrequency[Normal[Values[data[Counts, "class"]]]]
Out[ ]=
```

$$\left\{\frac{1}{2}, \frac{1}{2}\right\}$$