In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn import preprocessing
import dataprep
```

In [2]:
```
df=pd.read_csv("intensity 3k.csv")
```

In [3]:
```
df
```

Out[3]:

	Magnitude	Depth	destruction	Victim's reaction	Intensity
0	7.1	40.0	6	7	8
1	7.7	23.5	6	7	8
2	7.7	196.0	4	5	6
3	6.9	15.0	3	4	5
4	6.9	51.6	4	5	6
...	...	...	...	...	...
9995	6.7	15.0	7	8	9
9996	6.5	16.7	5	6	7
9997	7.8	60.0	6	7	8
9998	5.0	28.0	0	1	2
9999	6.5	2.7	6	7	8

10000 rows × 5 columns

In [4]:
```
df.head()
```

Out[4]:

	Magnitude	Depth	destruction	Victim's reaction	Intensity
0	7.1	40.0	6	7	8
1	7.7	23.5	6	7	8
2	7.7	196.0	4	5	6
3	6.9	15.0	3	4	5
4	6.9	51.6	4	5	6

In [5]:
```
df.tail()
```

Out[5]:

	Magnitude	Depth	destruction	Victim's reaction	Intensity
9995	6.7	15.0	7	8	9
9996	6.5	16.7	5	6	7
9997	7.8	60.0	6	7	8
9998	5.0	28.0	0	1	2
9999	6.5	2.7	6	7	8

In [6]:
```
df.shape
```

Out[6]:
```
(10000, 5)
```

In [7]:
```
df.describe()
```

Out[7]:

	Magnitude	Depth	destruction	Victim's reaction	Intensity
count	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	6.805333	53.459081	4.840900	5.840200	6.840300
std	0.799281	102.372100	1.825081	1.825047	1.826836
min	3.200000	0.000000	0.000000	0.000000	1.000000
25%	6.500000	12.590000	4.000000	5.000000	6.000000
50%	6.800000	24.000000	5.000000	6.000000	7.000000
75%	7.300000	40.000000	6.000000	7.000000	8.000000
max	9.500000	670.810000	10.000000	11.000000	12.000000

In [8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 5 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Magnitude          10000 non-null  float64
 1   Depth              10000 non-null  float64
 2   destruction        10000 non-null  int64
 3   Victim's reaction  10000 non-null  int64
 4   Intensity          10000 non-null  int64
dtypes: float64(2), int64(3)
memory usage: 390.8 KB
```

In [9]:
```
df.isnull().sum()
```

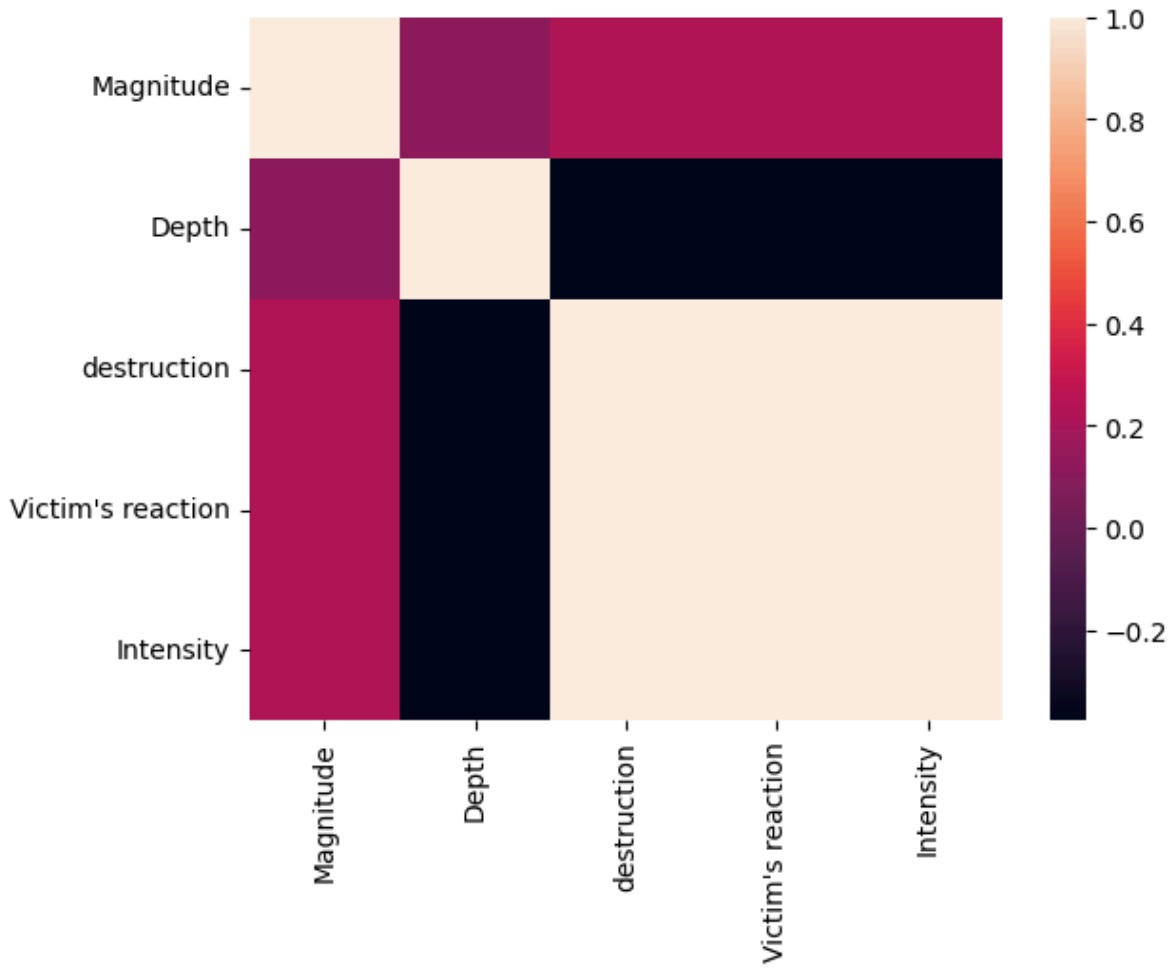
Out[9]:
```
Magnitude           0
Depth               0
destruction         0
Victim's reaction   0
Intensity           0
dtype: int64
```

In [10]:
```
df.dtypes
```

Out[10]:
```
Magnitude           float64
Depth               float64
destruction           int64
Victim's reaction     int64
Intensity             int64
dtype: object
```

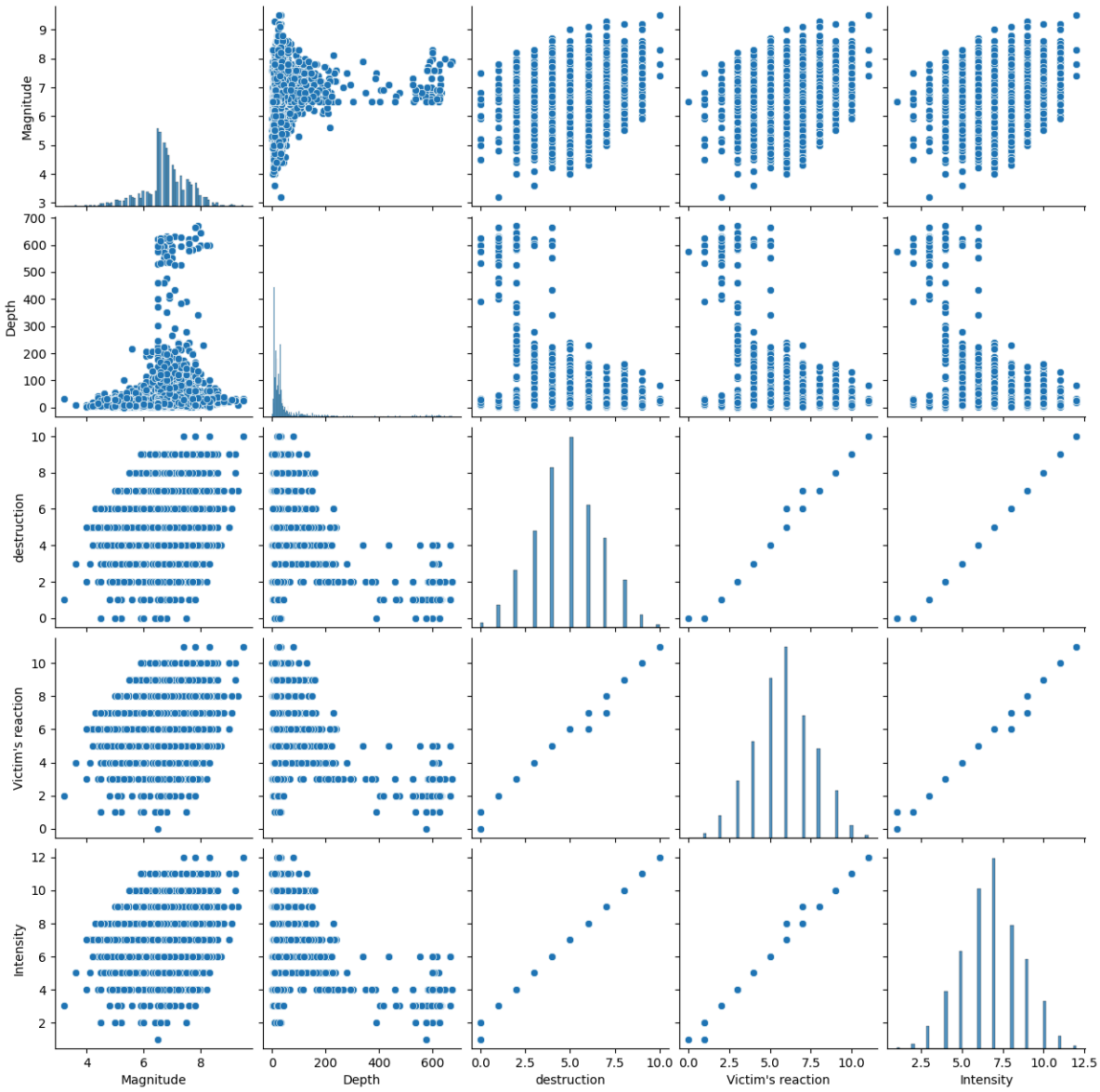
In [11]:
```
sns.heatmap(df.corr())
```

Out[11]:
```
<Axes: >
```



In [12]:
```
sns.pairplot(df)
```

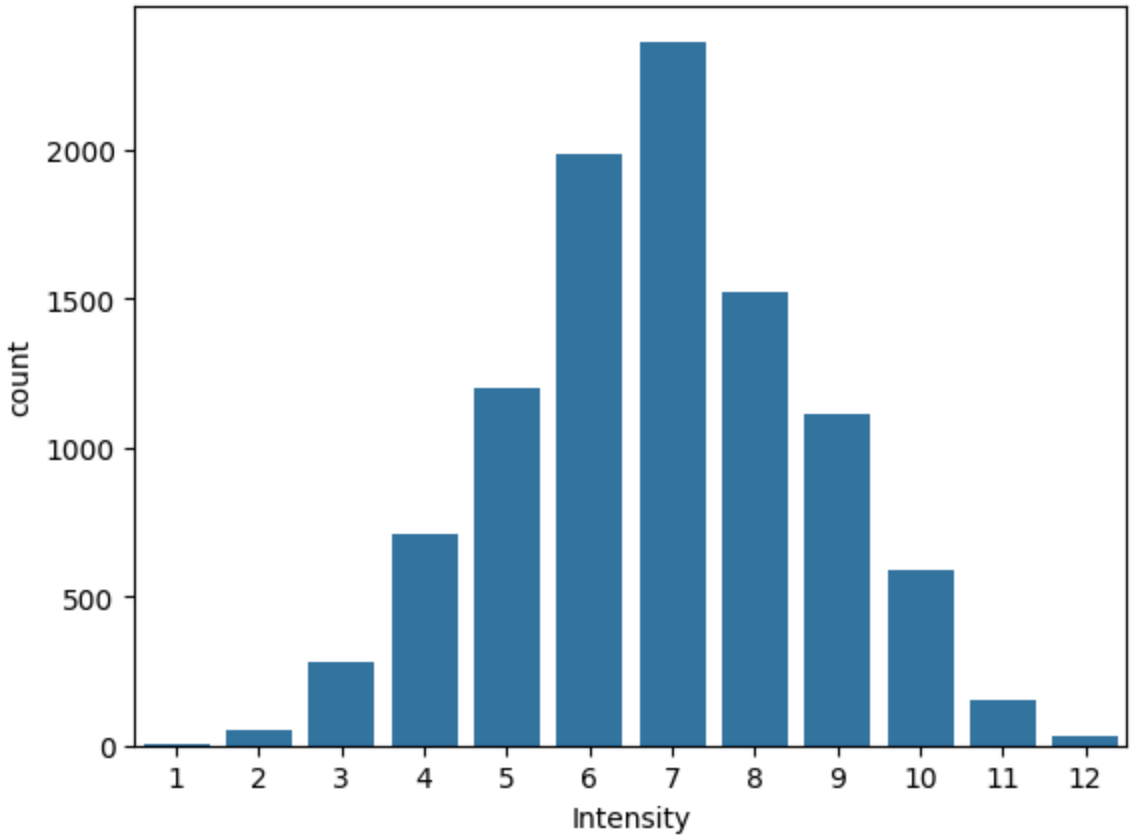
Out[12]:
```
<seaborn.axisgrid.PairGrid at 0x2970b947190>
```



In [13]:
```
sns.countplot(x='Intensity',data=df)
```

```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

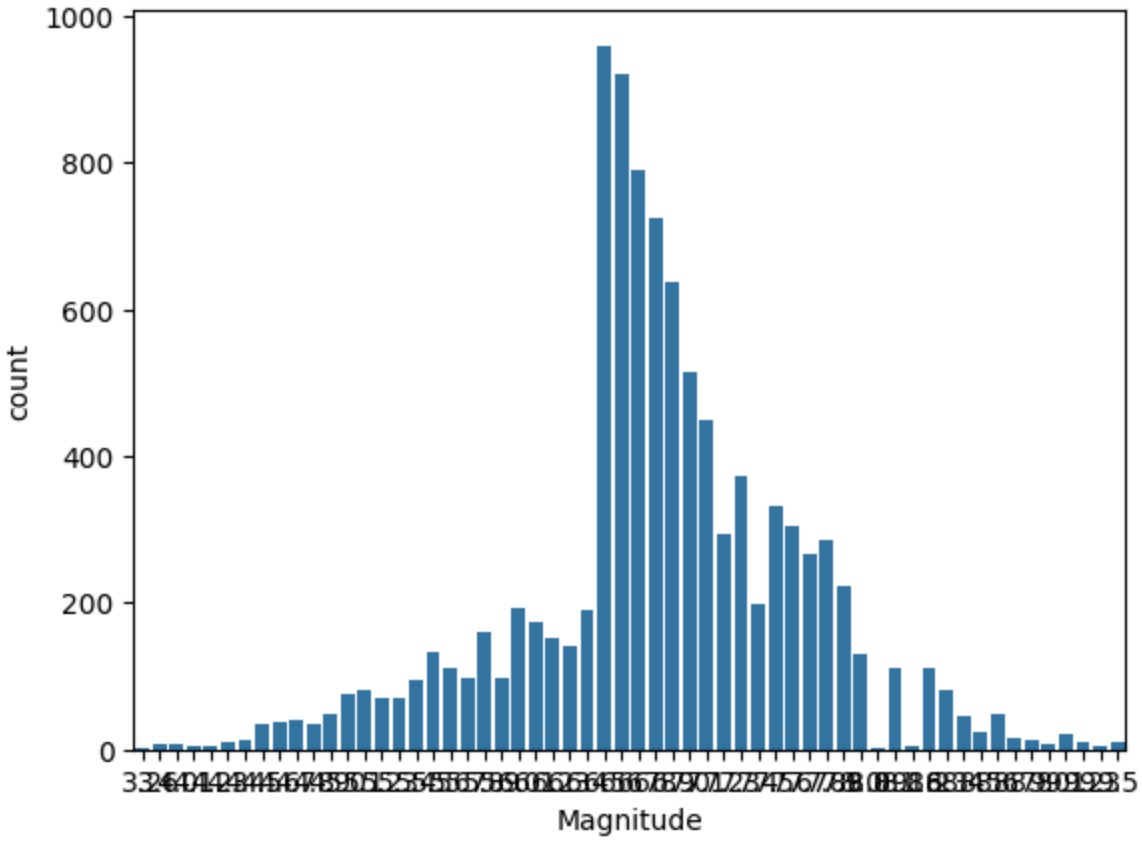
Out[13]:
```
<Axes: xlabel='Intensity', ylabel='count'>
```



In [14]:
```
sns.countplot(x='Magnitude',data=df)
```

```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

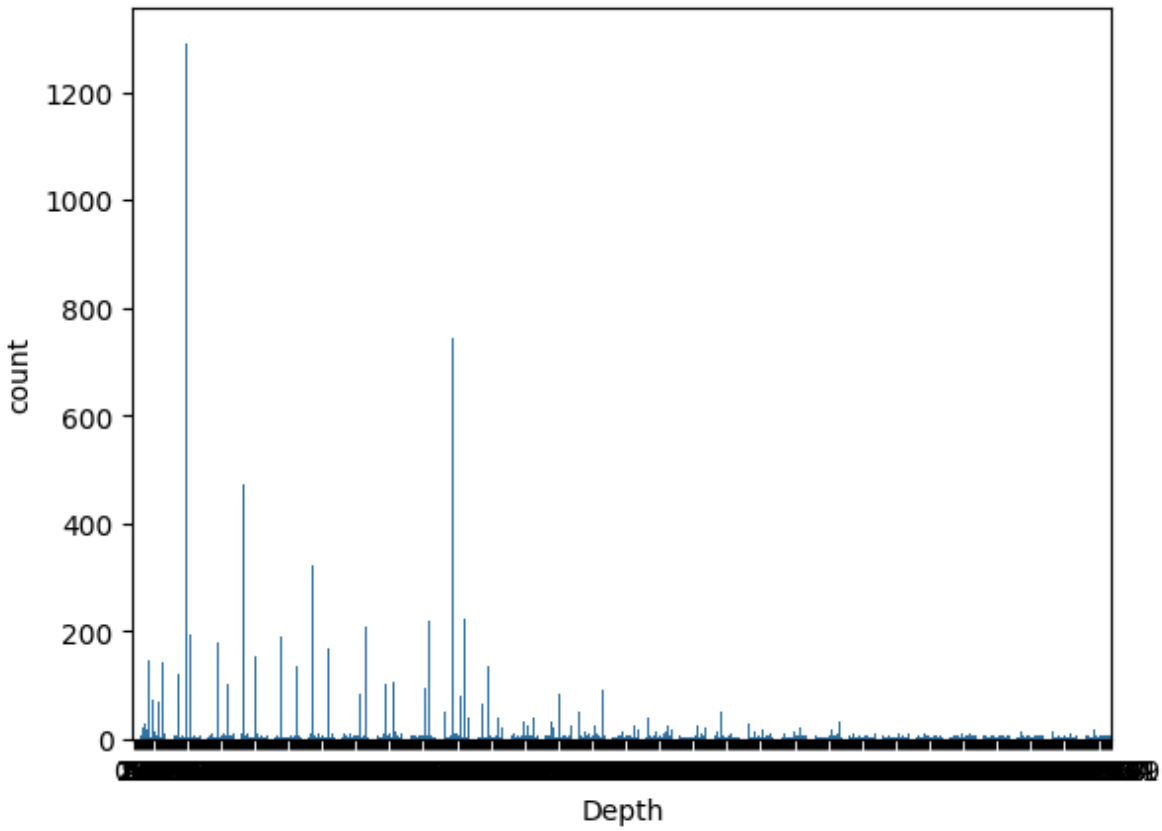
Out[14]: <Axes: xlabel='Magnitude', ylabel='count'>



In [15]:
```
sns.countplot(x='Depth',data=df)
```

```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

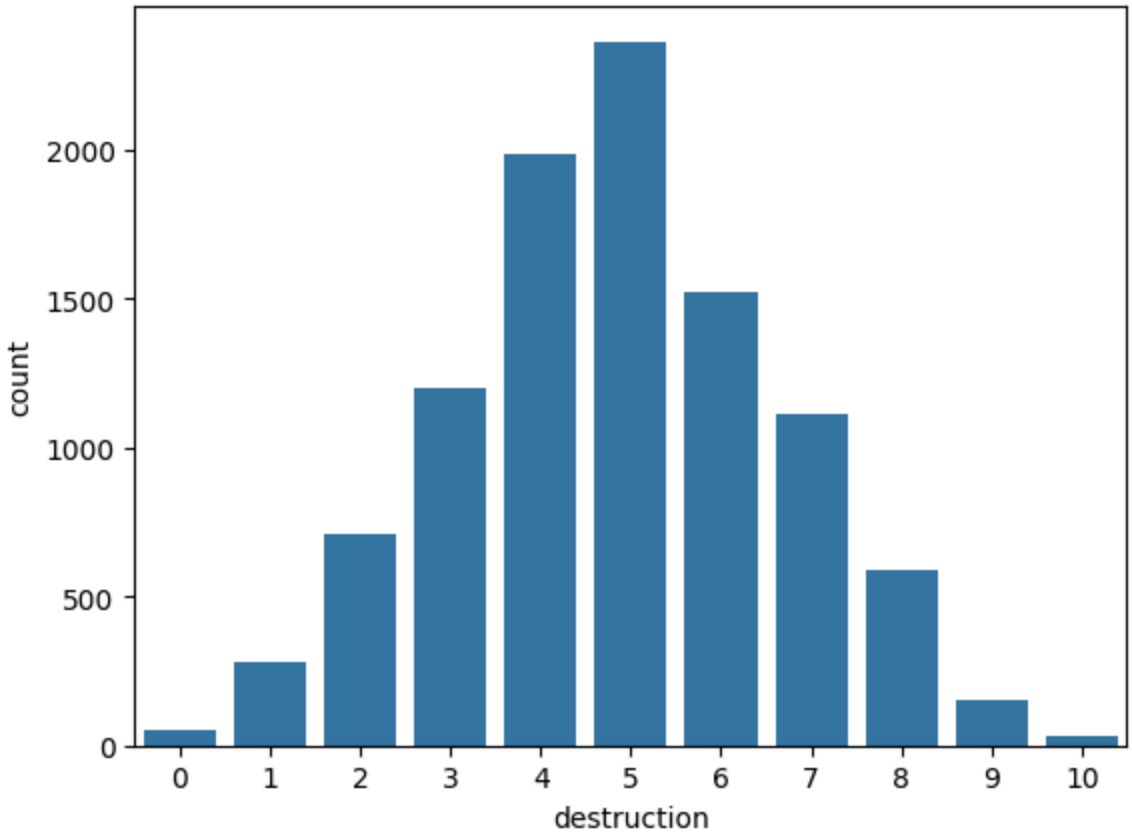
Out[15]:
```
<Axes: xlabel='Depth', ylabel='count'>
```



In [16]:
```
sns.countplot(x='destruction',data=df)
```

```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

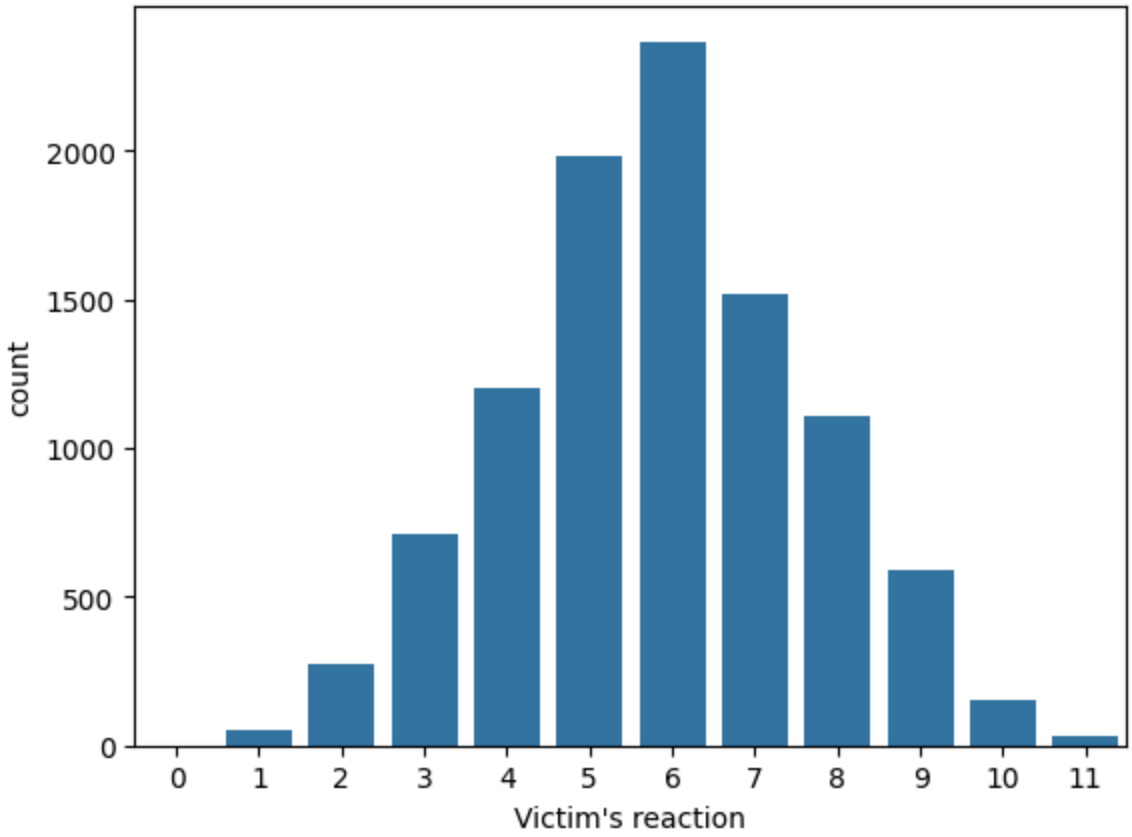
Out[16]:
```
<Axes: xlabel='destruction', ylabel='count'>
```



In [17]:
```
sns.countplot(x="Victim's reaction",data=df)
```

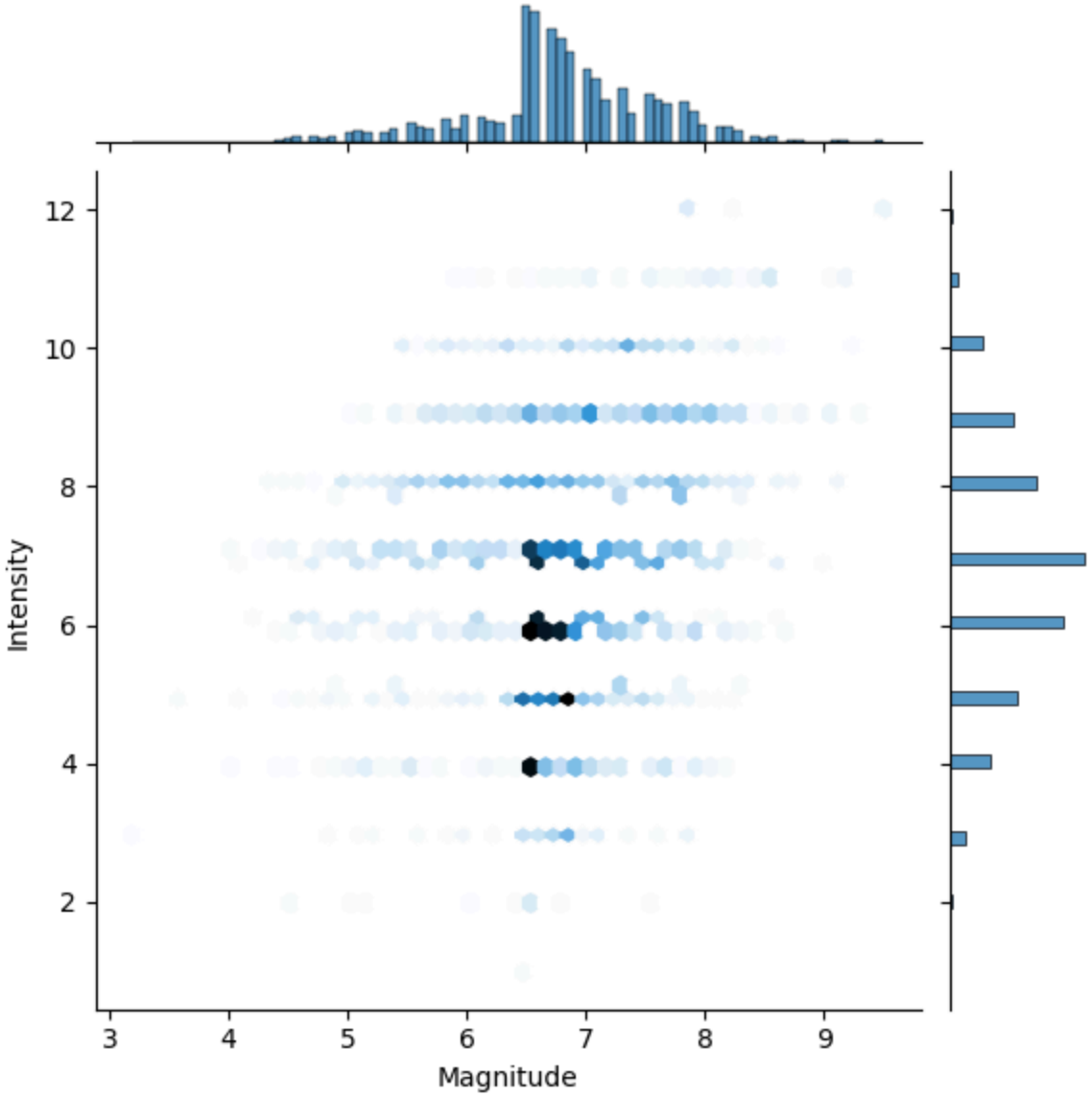
```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

Out[17]:
```
<Axes: xlabel="Victim's reaction", ylabel='count'>
```



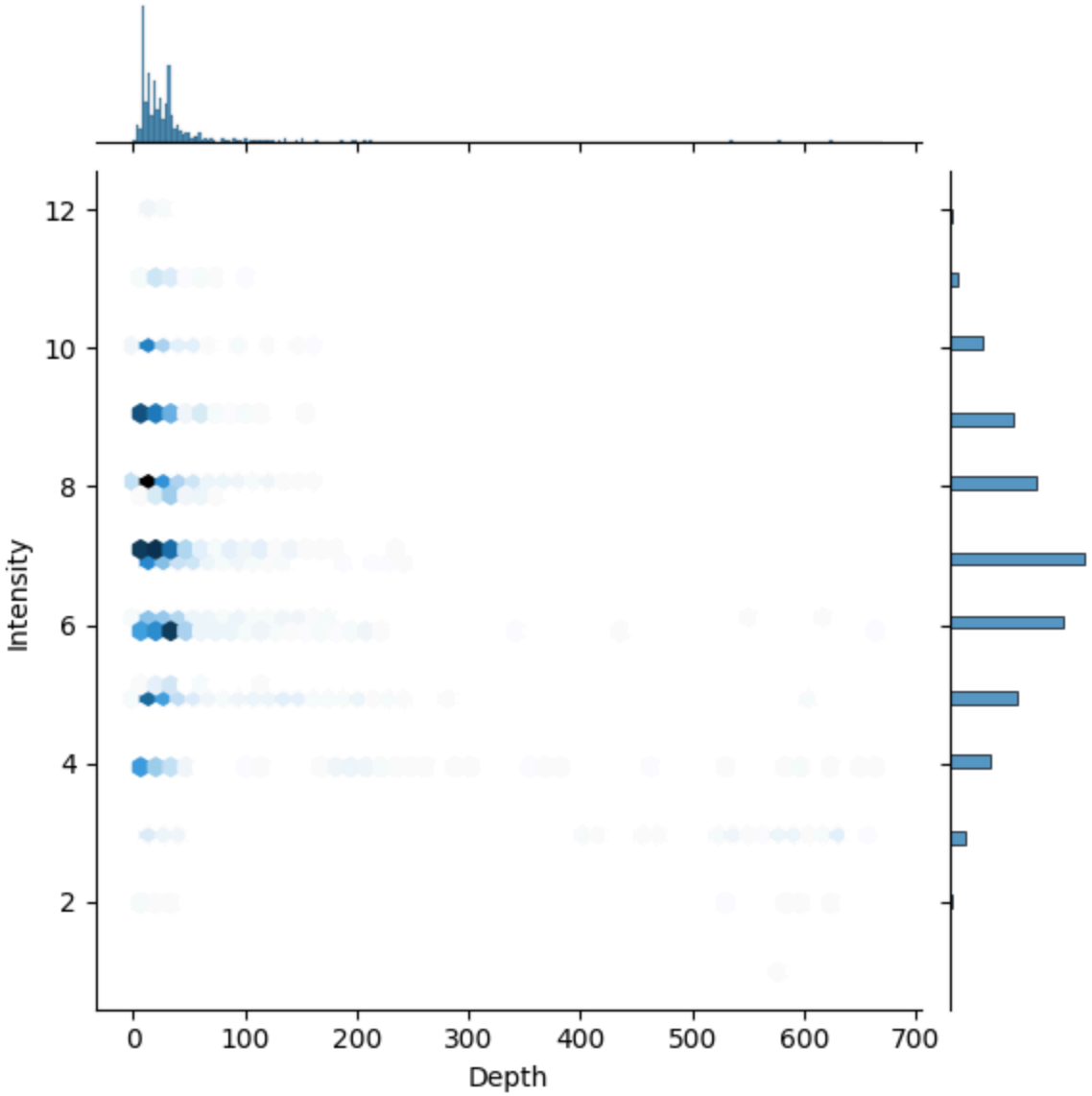
In [18]:
```
sns.jointplot(x='Magnitude', y='Intensity', data=df, kind='hex')
```

Out[18]:
```
<seaborn.axisgrid.JointGrid at 0x2971202c3a0>
```



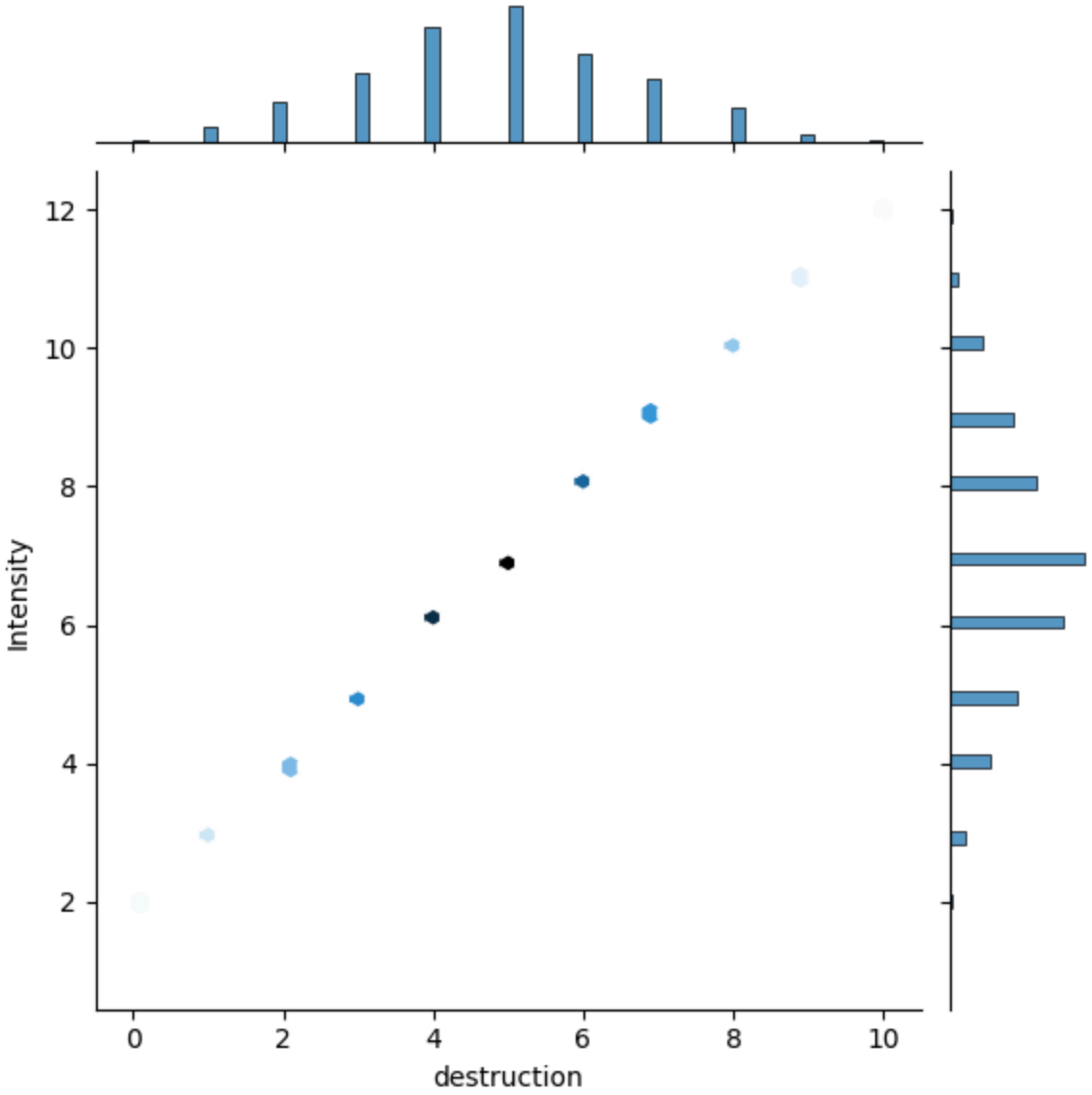
In [19]:
```
sns.jointplot(x='Depth', y='Intensity', data=df, kind='hex')
```

Out[19]:
```
<seaborn.axisgrid.JointGrid at 0x297104a0be0>
```



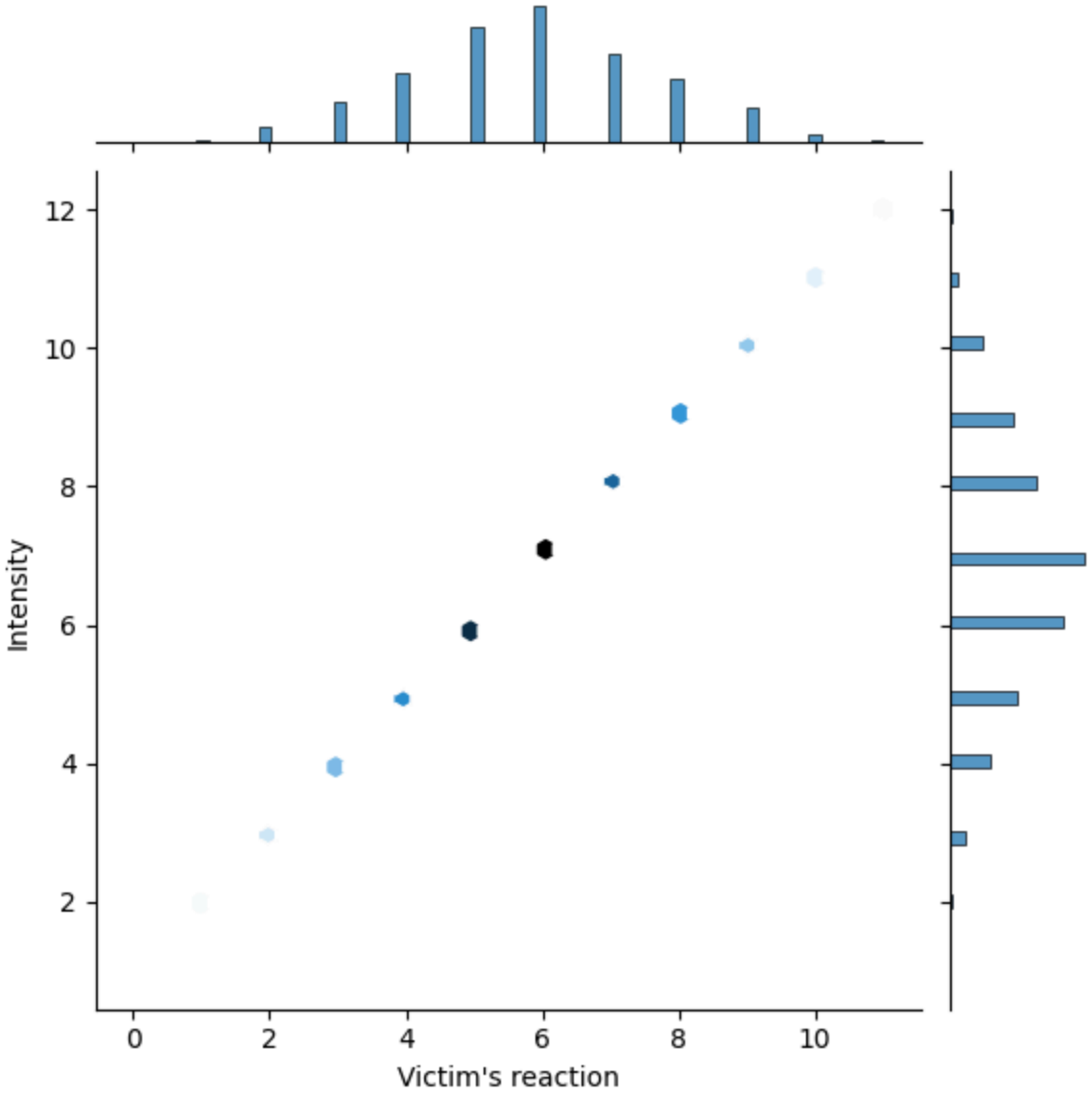
In [20]:
```
sns.jointplot(x='destruction', y='Intensity', data=df, kind='hex')
```

Out[20]:
```
<seaborn.axisgrid.JointGrid at 0x29712961b50>
```



In [21]:
```
sns.jointplot(x="Victim's reaction", y='Intensity', data=df, kind='hex')
```

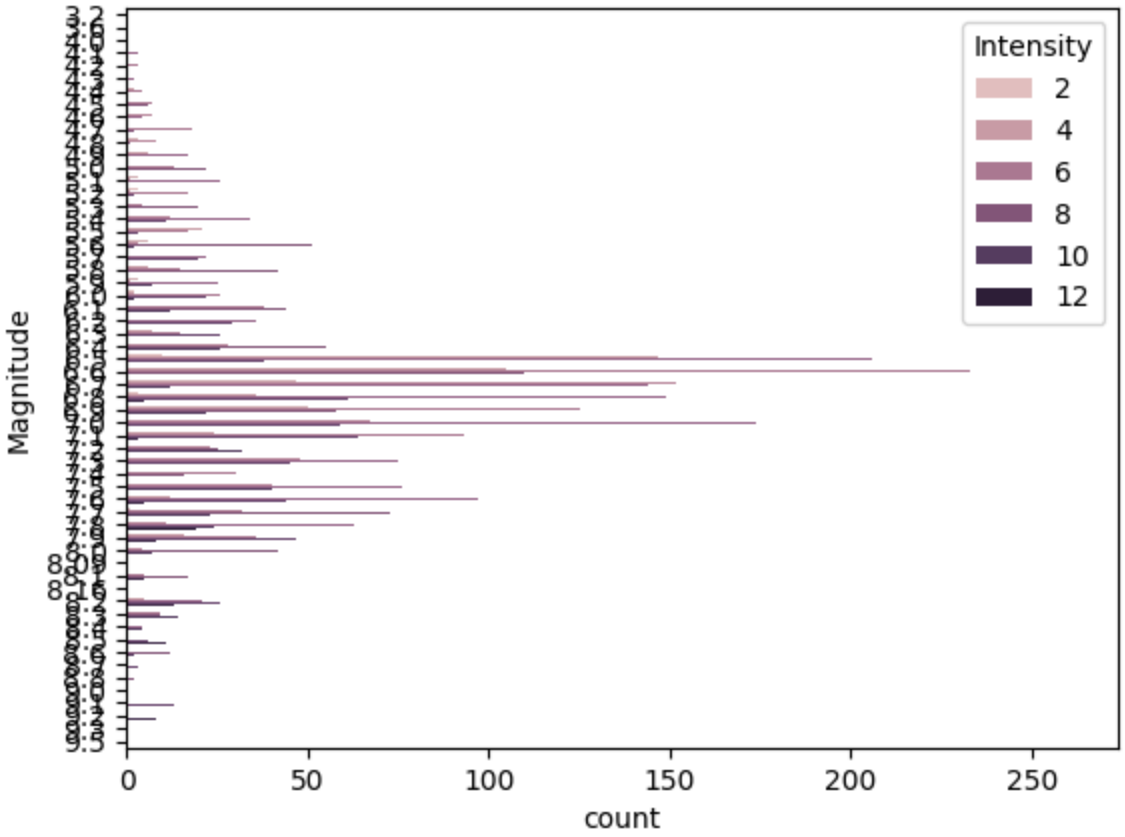
Out[21]:
```
<seaborn.axisgrid.JointGrid at 0x29712405d30>
```



In [22]:
```
sns.countplot(y='Magnitude',hue='Intensity',data=df)
```

```
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
Using categorical units to plot a list of strings that are all parsable as fl
oats or dates. If these strings should be plotted as numbers, cast to the app
ropriate data type before plotting.
```

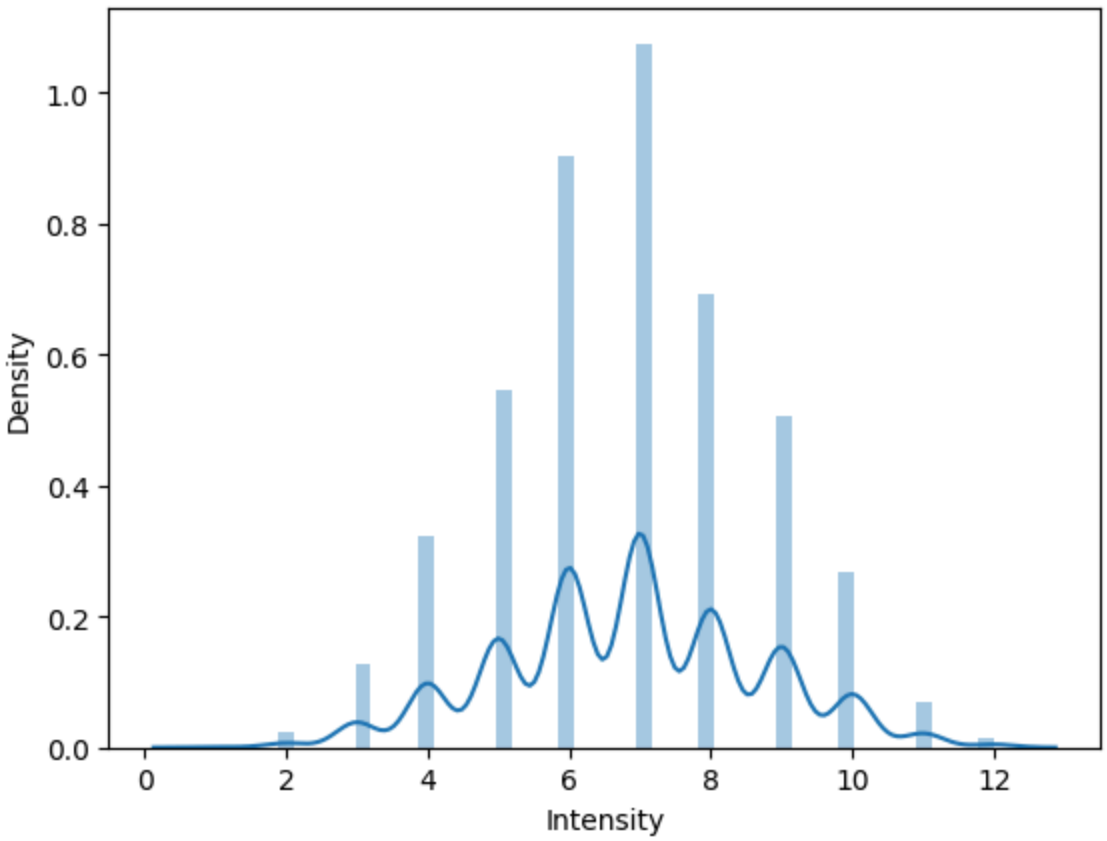
Out[22]: <Axes: xlabel='count', ylabel='Magnitude'>



In [23]:
```
sns.distplot(df['Intensity'] ,kde=True )
```

```
C:\Users\Prern\AppData\Local\Temp\ipykernel_15192\2715816930.py:1: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751 (https://gis
t.github.com/mwaskom/de44147ed2974457ad6372750bbe5751)

  sns.distplot(df['Intensity'] ,kde=True )
```

Out[23]: <Axes: xlabel='Intensity', ylabel='Density'>



In [24]:
```
from dataprep.eda import create_report
```

In [25]:
```
create_report(df).show_browser()
```

```
  0%|          | 0/981 [00:00<?, ?it/s]

C:\Users\Prern\anaconda3\envs\Mine\lib\site-packages\dask\core.py:127: Runtim
eWarning: invalid value encountered in divide
  return func(*(_execute_task(a, cache) for a in args))
```

In [ ]: