In [ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

Adult data set cleaning:

In [ ]:
```
import numpy as np
import pandas as pd
%config InlineBackend.figure_format = 'retina'
pd.options.mode.chained_assignment = None
```

In [ ]:
```
adult_data_UCI = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/data/UCI-adult.csv')
```

In [ ]:
```
def binarize(income):
    if (income == '<=50K'): return 0
    else: return 1
```

In [ ]:
```
adult_data_UCI['income'] = adult_data_UCI['income'].apply(binarize)
adult_data_UCI.rename(columns={'income':'y'}, inplace=True)
```

In [ ]:
```
adult_data_UCI.columns
```

In [ ]:
```
# del adult_data_UCI['fnlwgt']
adult_data_UCI.shape
```

In [ ]:
```
adult_data_UCI = adult_data_UCI.replace('?', np.nan).dropna(axis = 0, how = 'any')
adult_data_UCI.shape
```

In [ ]:
```
adult_data_UCI = pd.get_dummies(adult_data_UCI)

y = adult_data_UCI.pop('y')
adult_data_UCI['y'] = y
adult_data_UCI = adult_data_UCI.reset_index().drop('index', axis = 1)
# adult_data_UCI.to_csv('/content/drive/MyDrive/Colab Notebooks/data/cleaned/adult.csv')
```

In [ ]:
```
adult_data_UCI.shape
```

In [ ]:
```
print(adult_data_UCI.y.value_counts())
```

In [ ]:
```
adult_data_UCI.describe()
```

Letter data set cleaning:

In [ ]:
```
letter_p1_data = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/data/UCI-Letter.csv', header=None)
letter_p2_data = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/data/UCI-Letter.csv', header=None)
letter_p1_data.drop([0], inplace=True)
letter_p2_data.drop([0], inplace=True)
```

In [ ]:
```
def option1(letter):
    if (letter <= 'M'): return 1
    else: return 0
def option2(letter):
    if (letter == 'O'): return 1
    else: return 0
letter_p1_data[0] = letter_p2_data[0].apply(option1)
letter_p2_data[0] = letter_p2_data[0].apply(option2)
```

In [ ]:
```
letter_p1_data.rename(columns={0:'y'}, inplace=True)
letter_p2_data.rename(columns={0:'y'}, inplace=True)
y1 = letter_p1_data.pop('y')
letter_p1_data['y'] = y1
y2 = letter_p2_data.pop('y')
letter_p2_data['y'] = y2
```

In [ ]:
```
letter_p1_data.to_csv('/content/drive/MyDrive/Colab Notebooks/data/cleaned/letter_p1.csv')
letter_p2_data.to_csv('/content/drive/MyDrive/Colab Notebooks/data/cleaned/letter_p2.csv')
```

Covtype data set cleaning:

In [ ]:
```
covtype = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/data/covtype/covtype.data', header = None)
covtype.shape
```

In [ ]:
```
covtype[54].value_counts()
```

In [ ]:
```
def binarize_cov(cov_class):
    if (cov_class == 2): return 1
    else: return 0
```

In [ ]:
```
# only once
covtype[54] = covtype[54].apply(binarize_cov)
```

In [ ]:
```
covtype[54].value_counts()
```

In [ ]:
```
# COV TYPE has been converted to a binary problem by treating the largest class as the positive and the rest as negative.

covtype = covtype.reset_index().drop('index', axis = 1)
covtype[54].value_counts()
```

In [ ]:
```
covtype.rename(columns={54:'y'}, inplace=True)
covtype['y'].value_counts()
```

In [ ]:
```
covtype.shape
```

In [ ]:
```
covtype5 = covtype.sample(frac=0.05, random_state=1)
covtype5.shape
```

In [ ]:
```
covtype5.y.value_counts()
```

In [ ]:
```
covtype.to_csv('/content/drive/MyDrive/Colab Notebooks/data/cleaned/covtype.csv')
covtype5.to_csv('/content/drive/MyDrive/Colab Notebooks/data/cleaned/covtype5.csv')
```

In [ ]:
```

```