Adult Data Set Analysis

Kelompok 4

- Nuzha Musyafira	(05111640000014)
- Ferdinand Jason Gondowijoyo	(05111640000033)
- Nurlita Dhuha Fatmawati	(05111640000092)
- Jonathan Rehuel Lewerissa	(05111640000105)

Supervisor
Dr. Chastine Fatichah, M.Kom.

DATA MINING

DEPARTEMEN INFORMATIKA

INSTITUT TEKNOLOGI SEPULUH NOPEMBER

2019

Contents

Chapter 1

Dataset Analysis

This dataset analysis task is carried out by Kelompok 4

1.1 Introduction

On this dataset analysis task, we will analyze the Adult Data Set. The Adult Data Set (also known as the Census Dataset) is a dataset that aims to predict whether a person's income exceeds $50000 per year based on their census data.

This data set can be downloaded from https://archive.ics.uci.edu/ml/datasets/adult.

1.2 Preparation

Let's first `import` some libraries that we are going to need for our analysis.

```
In [1]: import math

        import pandas as pd
        import numpy as np
        import seaborn as sns
        import matplotlib.pyplot as plt

        import sklearn.preprocessing as preprocessing

        from sklearn.impute import SimpleImputer

        %matplotlib inline
```

Then, we need to read the adult dataset from `data/adult.csv` which contains comma separated columns and mark the values `?` as missing data points

```
In [2]: original_data = pd.read_csv(
            "data/adult.csv",
            names=["Age", "Workclass", "fnlwgt", "Education", "Education-Num", "Martial Status",
                "Occupation", "Relationship", "Race", "Sex", "Capital Gain", "Capital Loss",
                "Hours per week", "Country", "Target"],
            sep=r'\s*,\s*',
            engine='python',
            na_values="?")

        original_data.head()
```

```
Out[2]:    Age         Workclass  fnlwgt  Education  Education-Num  \
        0   39         State-gov   77516  Bachelors             13
        1   50  Self-emp-not-inc   83311  Bachelors             13
        2   38           Private  215646    HS-grad              9
        3   53           Private  234721       11th              7
        4   28           Private  338409  Bachelors             13

               Martial Status         Occupation   Relationship   Race     Sex  \
        0       Never-married       Adm-clerical  Not-in-family  White    Male
        1  Married-civ-spouse    Exec-managerial        Husband  White    Male
        2            Divorced  Handlers-cleaners  Not-in-family  White    Male
        3  Married-civ-spouse  Handlers-cleaners        Husband  Black    Male
        4  Married-civ-spouse     Prof-specialty           Wife  Black  Female

           Capital Gain  Capital Loss  Hours per week        Country Target
        0          2174             0              40  United-States  <=50K
        1             0             0              13  United-States  <=50K
        2             0             0              40  United-States  <=50K
        3             0             0              40  United-States  <=50K
        4             0             0              40           Cuba  <=50K
```

1.3 Data Insight

First, we need to see the general statistical information of the dataset.

```
In [3]: def summarize_data(df):
            print('Continuous Data : ')
            print(df.describe())
            print('\n\n')
            print('Categorical Data : ')
            for column in df.columns:
                if df.dtypes[column] == np.object : # Categorical Data
                    print(column)
                    print(df[column].value_counts())
                print()

        summarize_data(original_data)
```

```
Continuous Data :
                Age        fnlwgt  Education-Num  Capital Gain  Capital Loss  \
count  32561.000000  3.256100e+04   32561.000000  32561.000000  32561.000000
mean      38.581647  1.897784e+05      10.080679   1077.648844     87.303830
std       13.640433  1.055500e+05       2.572720   7385.292085    402.960219
min       17.000000  1.228500e+04       1.000000      0.000000      0.000000
25%       28.000000  1.178270e+05       9.000000      0.000000      0.000000
50%       37.000000  1.783560e+05      10.000000      0.000000      0.000000
75%       48.000000  2.370510e+05      12.000000      0.000000      0.000000
max       90.000000  1.484705e+06      16.000000  99999.000000   4356.000000

       Hours per week
count    32561.000000
mean        40.437456
std         12.347429
min          1.000000
```

```
25%          40.000000
50%          40.000000
75%          45.000000
max          99.000000
```

```
Categorical Data :

Workclass
Private             22696
Self-emp-not-inc     2541
Local-gov            2093
State-gov            1298
Self-emp-inc         1116
Federal-gov           960
Without-pay            14
Never-worked            7
Name: Workclass, dtype: int64


Education
HS-grad         10501
Some-college     7291
Bachelors        5355
Masters          1723
Assoc-voc        1382
11th             1175
Assoc-acdm       1067
10th              933
7th-8th           646
Prof-school       576
9th               514
12th              433
Doctorate         413
5th-6th           333
1st-4th           168
Preschool          51
Name: Education, dtype: int64


Martial Status
Married-civ-spouse       14976
Never-married            10683
Divorced                  4443
Separated                 1025
Widowed                    993
Married-spouse-absent      418
Married-AF-spouse           23
Name: Martial Status, dtype: int64

Occupation
Prof-specialty       4140
Craft-repair         4099
Exec-managerial      4066
Adm-clerical         3770
Sales                3650
Other-service        3295
Machine-op-inspct    2002
Transport-moving     1597
Handlers-cleaners    1370
Farming-fishing       994
Tech-support          928
Protective-serv       649
Priv-house-serv       149
Armed-Forces            9
Name: Occupation, dtype: int64

Relationship
Husband           13193
Not-in-family      8305
Own-child          5068
Unmarried          3446
Wife               1568
Other-relative      981
Name: Relationship, dtype: int64

Race
White                 27816
Black                  3124
Asian-Pac-Islander     1039
Amer-Indian-Eskimo      311
Other                   271
Name: Race, dtype: int64

Sex
Male      21790
Female    10771
Name: Sex, dtype: int64


Country
United-States                 29170
Mexico                          643
Philippines                     198
Germany                         137
Canada                          121
Puerto-Rico                     114
El-Salvador                     106
```

```
India                      100
Cuba                        95
England                     90
Jamaica                     81
South                       80
China                       75
Italy                       73
Dominican-Republic          70
Vietnam                     67
Guatemala                   64
Japan                       62
Poland                      60
Columbia                    59
Taiwan                      51
Haiti                       44
Iran                        43
Portugal                    37
Nicaragua                   34
Peru                        31
France                      29
Greece                      29
Ecuador                     28
Ireland                     24
Hong                        20
Trinadad&Tobago             19
Cambodia                    19
Thailand                    18
Laos                        18
Yugoslavia                  16
Outlying-US(Guam-USVI-etc)  14
Honduras                    13
Hungary                     13
Scotland                    12
Holand-Netherlands           1
Name: Country, dtype: int64

Target
<=50K    24720
>50K      7841
Name: Target, dtype: int64
```

1.3.1 Data Dictionary

1. Categorial Attributes

- workclass: (categorical) Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
  - Individual work category
- education: (categorical) Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- Individual's highest education degree
- marital-status: (categorical) Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
  - Individual marital status
- occupation: (categorical) Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
  - Individual's occupation
- relationship: (categorical) Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
  - Individual's relation in a family
- race: (categorical) White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
  - Race of Individual
- sex: (categorical) Female, Male.
- native-country: (categorical) United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.
  - Individual's native country

2. Continuous Attributes
    - age: continuous.
        - Age of an individual
    - education-num: number of education year, continuous.
        - Individual's year of receiving education
    - fnlwgt: final weight, continuous.
        - The weights on the CPS files are controlled to independent estimates of the civilian noninstitutional population of the US. These are prepared monthly for us by Population Division here at the Census Bureau.
    - capital-gain: continuous.
    - capital-loss: continuous.
    - hours-per-week: continuous.
        - Individual's working hour per week

Check if there are any NaNs in the dataframe and count every columns

```
In [4]: original_data.isnull().sum()
```

```
Out[4]: Age                  0
        Workclass         1836
        fnlwgt               0
        Education            0
        Education-Num        0
        Martial Status       0
        Occupation        1843
        Relationship         0
        Race                 0
        Sex                  0
        Capital Gain         0
        Capital Loss         0
        Hours per week       0
        Country            583
        Target               0
        dtype: int64
```

1.3.2 Histogram Analysis

A histogram is an accurate representation of the distribution of numerical data. It is an estimate of the probability distribution of a continuous variable (quantitative variable) and was first introduced by Karl Pearson. It differs from a bar graph, in the sense that a bar graph relates two variables, but a histogram relates only one. To construct a histogram, the first step is to "bin" (or "bucket") the range of values—that is, divide the entire range of values into a series of intervals—and then count how many values fall into each interval. The bins are usually specified as consecutive, non-overlapping intervals of a variable. The bins (intervals) must be adjacent, and are often (but are not required to be) of equal size.

Histogram can be summarized roughly as an inventory of what "kinds of items" you have and "how many of each kind" you have. In computer vision, histogram appears a lot and many times helps to introduce some sort of robustness to your method. For example a bunch of techniques called local features/descriptors make use of the histogram of the image gradient in an image region. This summary representation helps you compare different images without being affected too much by variations in pixel values, shifts and tilts, etc. that change the individual pixel values significantly. So, histogram has the benefit of a summary data structure that is robust to certain changes that you want to ignore in the raw data.

```
In [5]: def make_histogram(df):
            fig = plt.figure(figsize=(20,35))
            COL = 3
            ROW = math.ceil(float(df.shape[1])/COL)

            for i , column in enumerate(df.columns):
                ax = fig.add_subplot(ROW, COL, i+1)
                ax.set_title(column)
                if df.dtypes[column] == np.object:
                    df[column].value_counts().plot(kind="bar", axes = ax)
                else:
                    df[column].hist(axes = ax)
                    plt.xticks(rotation="vertical")

            plt.subplots_adjust(hspace=0.7, wspace=0.2)

        make_histogram(original_data)
```

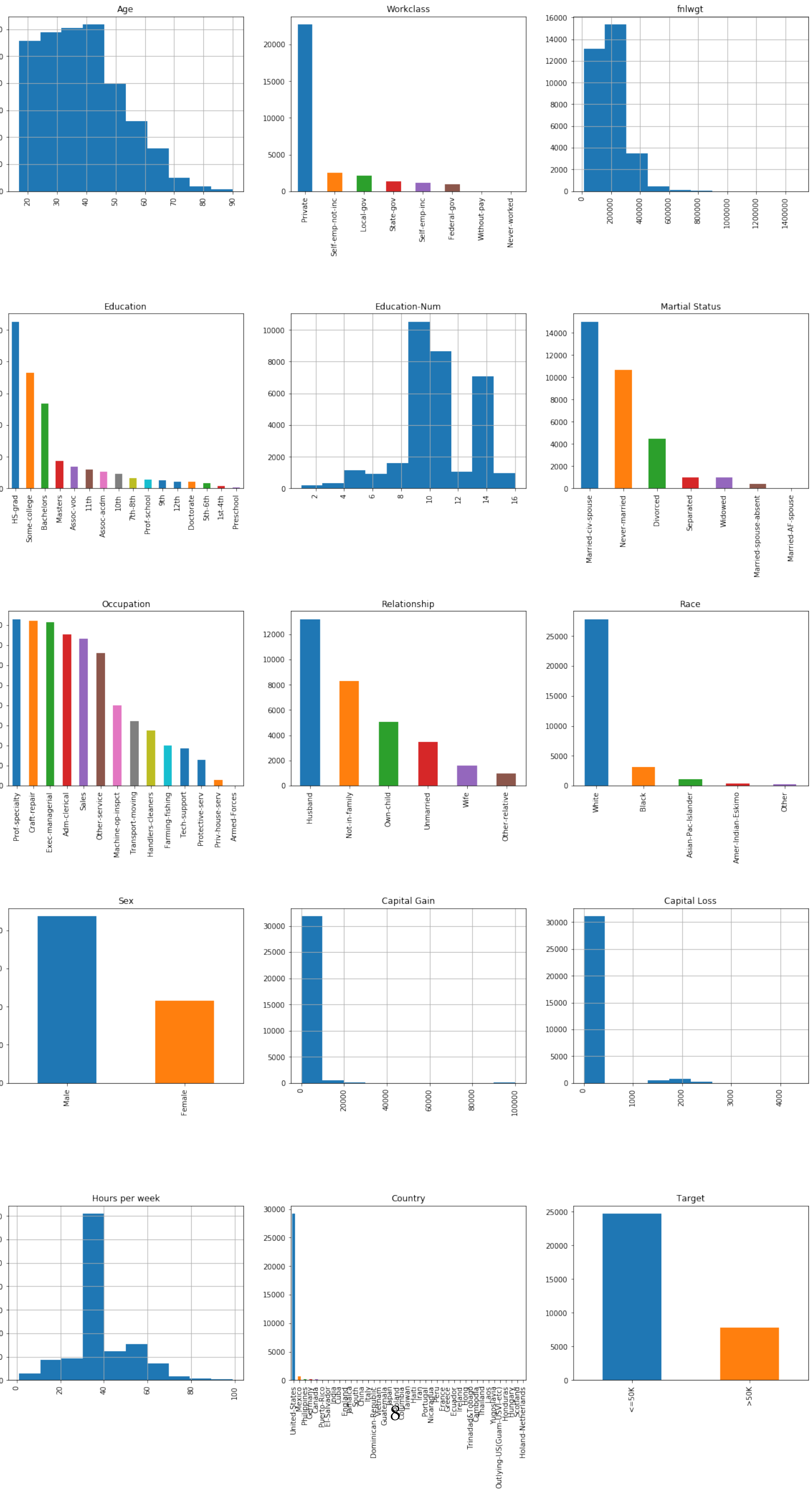
The histograms below shows that all of the data do not have a normal distribution, therefore requiring special methods to deal with the missing value.

The `Country` feature analysis is described below.

```
In [6]: (original_data["Country"].value_counts() / original_data.shape[0]).head()

Out[6]: United-States    0.895857
        Mexico           0.019748
        Philippines      0.006081
        Germany          0.004207
        Canada           0.003716
        Name: Country, dtype: float64
```

Indeed! 89% of the samples are for people from the US. Mexico comes next with less than 2%.


Age
Workclass
Private
Self-emp-not-inc
Local-gov
State-gov
Self-emp-inc
Federal-gov
Without-pay
Never-worked
fnlwgt
Education
HS-grad
Some-college
Bachelors
Masters
Assoc-voc
11th
Assoc-acdm
10th
7th-8th
Prof-school
9th
12th
Doctorate
5th-6th
1st-4th
Preschool
Education-Num
Martial Status
Married-civ-spouse
Never-married
Divorced
Separated
Widowed
Married-spouse-absent
Married-AF-spouse
Occupation
Prof-specialty
Craft-repair
Exec-managerial
Adm-clerical
Sales
Other-service
Machine-op-inspct
Transport-moving
Handlers-cleaners
Farming-fishing
Tech-support
Protective-serv
Priv-house-serv
Armed-Forces
Relationship
Husband
Not-in-family
Own-child
Unmarried
Wife
Other-relative
Race
White
Black
Asian-Pac-Islander
Amer-Indian-Eskimo
Other
Sex
Male
Female
Capital Gain
Capital Loss
Hours per week
Country
Target
<=50K
>50K

1.3.3 Boxplot Analysis

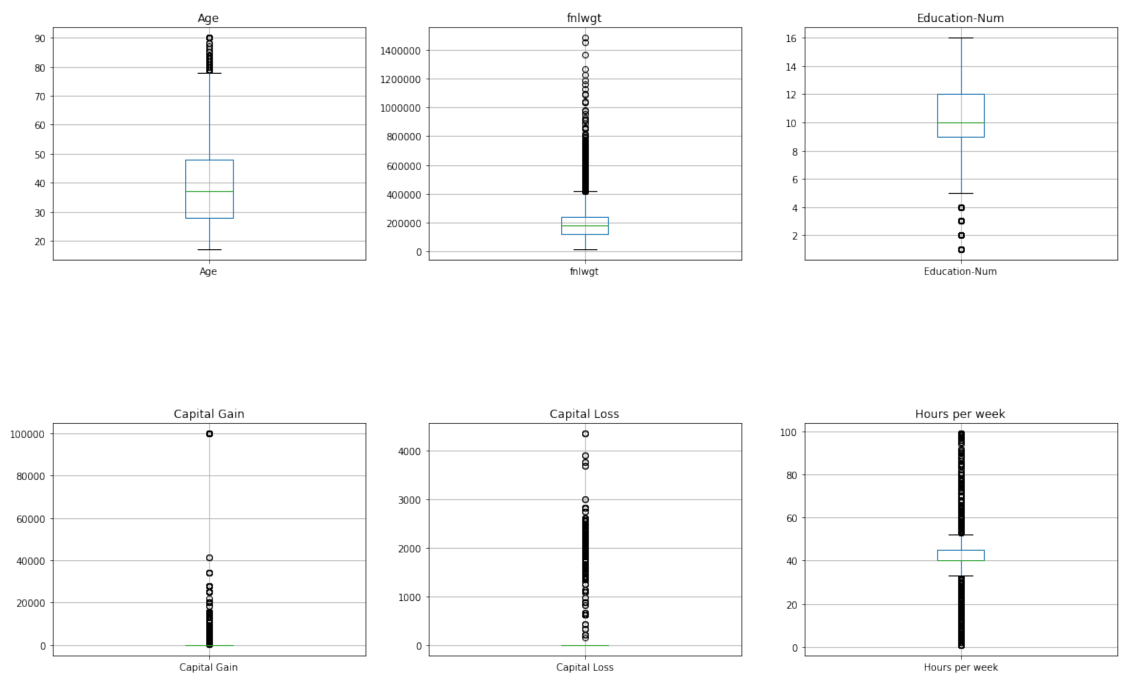
Boxplot is a method for graphically depicting groups of numerical data through their quartiles. Box plots handle large data effortlessly, but they do not retain the exact values and the details of the results of the distribution. These graphs allow a clear summary of large amounts of data.

```
In [7]: def make_boxplot(df):
            fig = plt.figure(figsize=(20,35))
            COL = 3
            ROW = math.ceil(float(df.shape[1])/COL)

            iterator = 1
            for column in df.columns:
                if df.dtypes[column] != np.object:
                    ax = fig.add_subplot(ROW, COL, iterator)
                    ax.set_title(column)
                    pd.DataFrame(df[column], columns=[column]).boxplot()
                    iterator+=1

            plt.subplots_adjust(hspace=0.7, wspace=0.2)
            plt.show()

        make_boxplot(original_data)
```

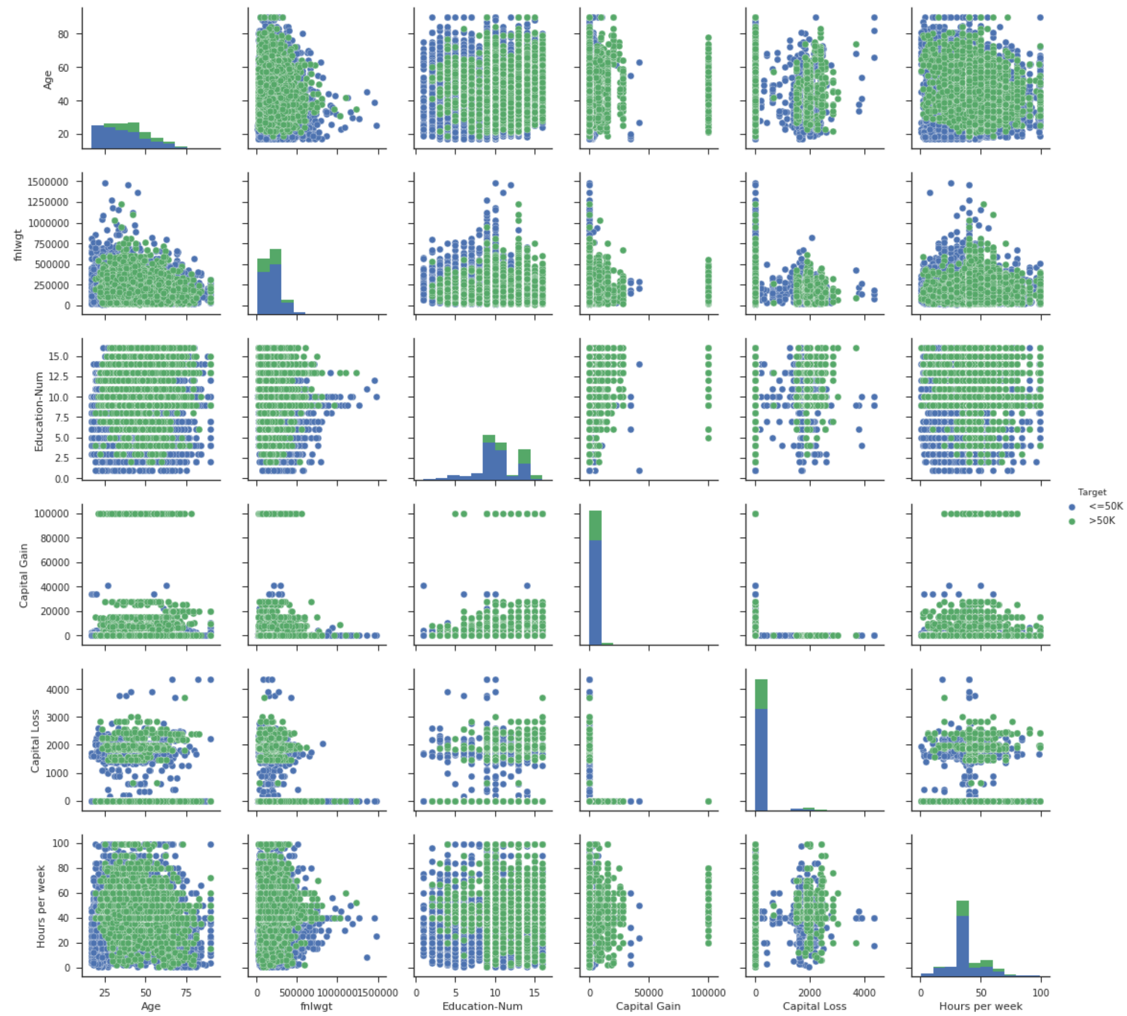


The Boxplot shows that some of the data have many outlier values. This is still acceptable as the main data because these data are consisted of categorical data types.

1.3.4 Correlation Analysis

We also need to do data correlation analysis to figure out the correlation between each feature inside the dataset. Below are the `pairplot` analysis of each features in the dataset.

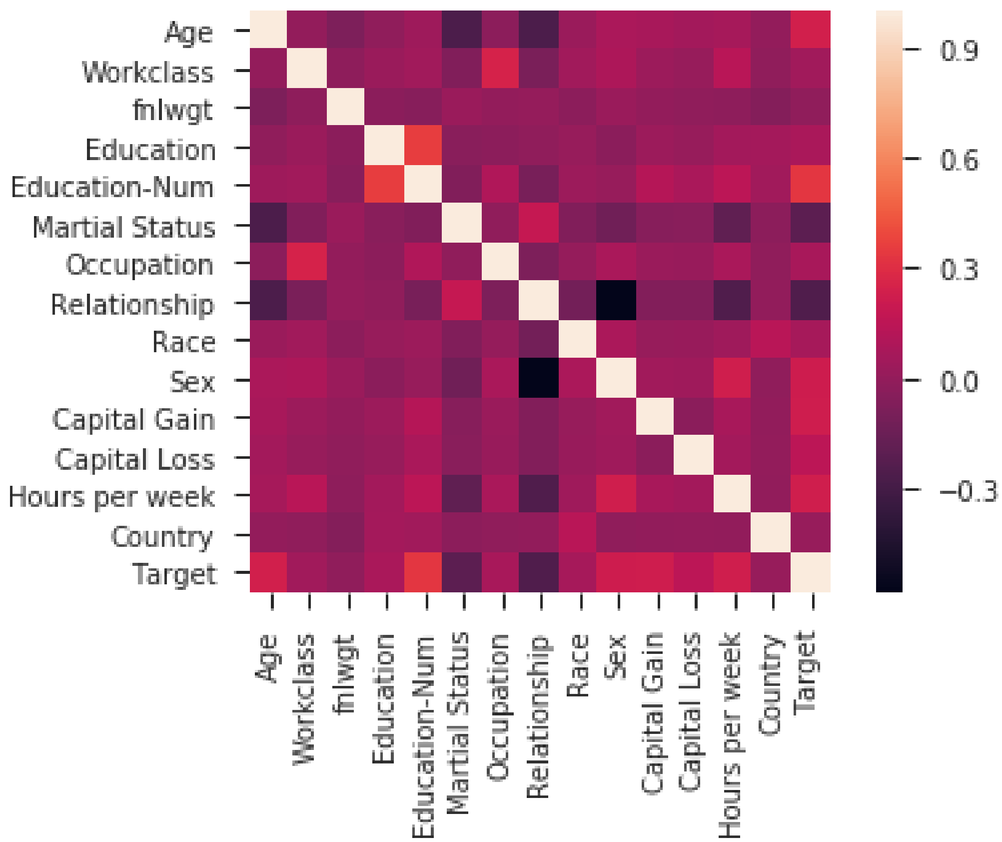
```
In [8]: sns.set(style="ticks")
        sns.pairplot(original_data, hue='Target')
        plt.show()
```



Below are the data correlation analysis using `heatmap` analysis.

```
In [9]: # Encode the categorical features as numbers
        def number_encode_features(df):
            result = df.copy()
            encoders = {}
            for column in result.columns:
                if result.dtypes[column] == np.object:
                    encoders[column] = preprocessing.LabelEncoder()
                    result[column] = encoders[column].fit_transform(result[column].fillna('0'))
            return result, encoders
```

```
# Calculate the correlation and plot it
encoded_data, _ = number_encode_features(original_data)
sns.heatmap(encoded_data.corr(), square=True)
plt.show()
```



The heatplot above shows that there is a high correlation between `Education` and `Education-Num`.

```
In [10]: original_data[["Education", "Education-Num"]].head(15)
```

```
Out[10]:        Education  Education-Num
         0      Bachelors             13
         1      Bachelors             13
         2        HS-grad              9
         3           11th              7
         4      Bachelors             13
         5        Masters             14
         6            9th              5
         7        HS-grad              9
         8        Masters             14
         9      Bachelors             13
         10  Some-college             10
         11     Bachelors             13
         12     Bachelors             13
```

```
13    Assoc-acdm              12
14     Assoc-voc              11
```

Two columns Education and Education-Num actually represent the same features, but encoded as strings and as numbers. We don't need the string representation, so we can just delete this column. Note that it is a much better option to delete the Education column as the Education-Num has the important property that the values are ordered: the higher the number, the higher the education that person has. This is a vaulable information a machine learning algorithm can use.

1.4 Data Preprocessing

The preprocessing that will be carried out Imputation using `Simpleimputer`. To replace the missing values in the categorical data, we will use the mode or the most frequent value that appeared in each column. On the `SimpleInputer` method, this is carried out using the `strategy='most_frequent'` as the parameter.

In [11]:
```
imputer_modus = SimpleImputer(missing_values=np.nan, strategy='most_frequent')
imputer_modus.fit(original_data)
imputed_data = imputer_modus.transform(original_data)

imputed_dataframe = pd.DataFrame(imputed_data,
    columns=["Age", "Workclass", "fnlwgt", "Education", "Education-Num", "Martial Status",
        "Occupation", "Relationship", "Race", "Sex", "Capital Gain", "Capital Loss",
        "Hours per week", "Country", "Target"])
imputed_dataframe.head()
```

Out[11]:
```
  Age         Workclass  fnlwgt  Education Education-Num      Martial Status  \
0  39         State-gov   77516  Bachelors            13       Never-married
1  50  Self-emp-not-inc   83311  Bachelors            13  Married-civ-spouse
2  38           Private  215646    HS-grad             9            Divorced
3  53           Private  234721       11th             7  Married-civ-spouse
4  28           Private  338409  Bachelors            13  Married-civ-spouse

          Occupation   Relationship   Race     Sex Capital Gain Capital Loss  \
0       Adm-clerical  Not-in-family  White    Male         2174            0
1    Exec-managerial        Husband  White    Male            0            0
2  Handlers-cleaners  Not-in-family  White    Male            0            0
3  Handlers-cleaners        Husband  Black    Male            0            0
4     Prof-specialty           Wife  Black  Female            0            0

  Hours per week        Country Target
0             40  United-States  <=50K
1             13  United-States  <=50K
2             40  United-States  <=50K
3             40  United-States  <=50K
4             40           Cuba  <=50K
```