Fundamentals of AI & ML

Monsoon Semester V 2021-22

Lab - 1

Date: 24 September 2021

Topic: Variables & CSV

AIM

Write a python program to open Comma Separated Value (CSV) and perform given statistical operations.

THEORY

CSV

A CSV (comma-separated values) file is a text file that has a specific format which allows data to be saved in a table structured format.

In statistics, there are broadly 2 types of variables:

Numerical variables: Numbers which should be treated as they usually are in mathematics. For example, age and weight would be considered numerical variables, while phone number and ZIP code would not be considered numerical variables. There are 2 types of numerical variables:

- **Continuous variable:** A numerical variable that can take values on a continuous scale (e.g. age, weight).
- **Discrete variable:** A numerical variable that only takes on whole numbers (e.g. number of visits).

Categorical variables: Variables which should not be treated like numbers (as in mathematics), and whose values come from a list of possibilities.

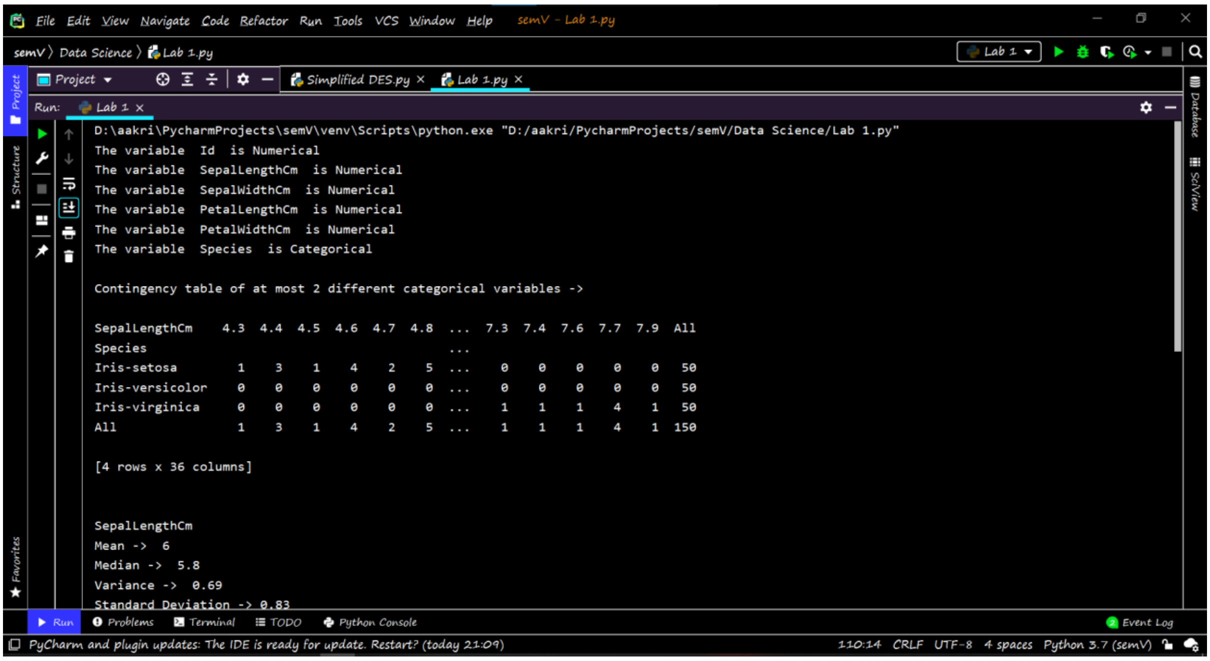
- **Nominal variable:** A categorical variable where the categories do not have a natural ordering (e.g. gender, ethnicity, country).
- **Ordinal variable:** A categorical variable where the categories have a natural ordering (e.g. age group, income level, educational status).

EXPERIMENT

Program CODE

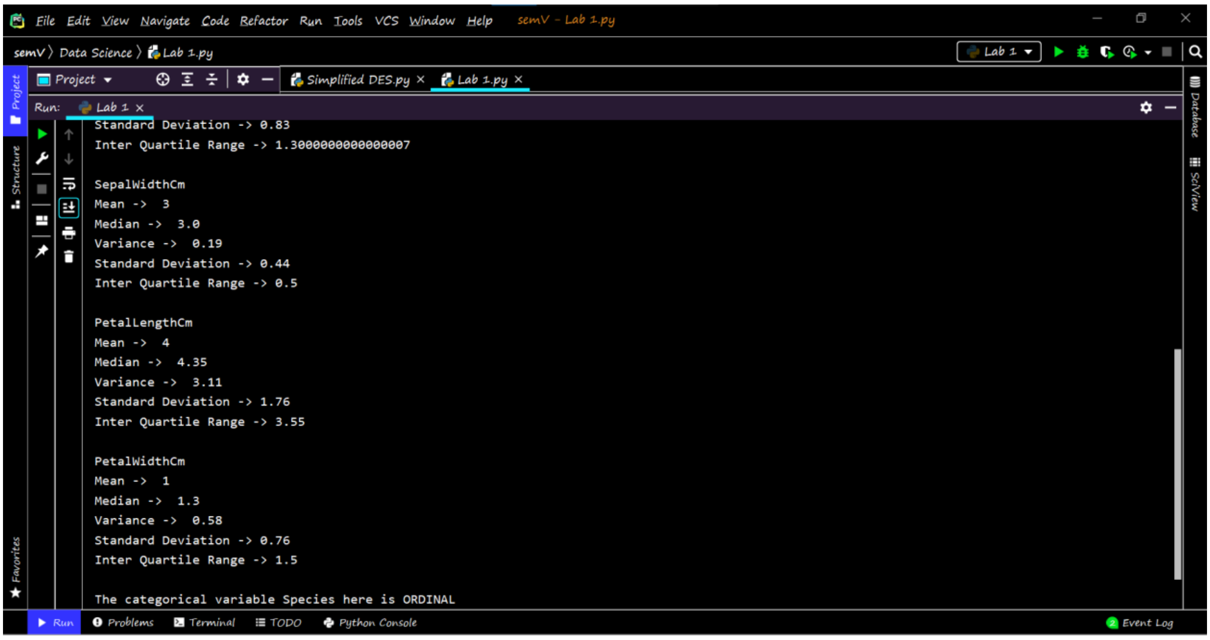
```
import csv
import pandas as pd

data_set = dict()
with open(".\\Data Set.csv", 'r') as csvfile:
    reader = list(csv.DictReader(csvfile))
    for key in reader[0].keys():
        data_set[key] = []
    for row in reader:
        for key, value in row.items():
            if '.' in value:
                value = float(value)
            elif value.isdigit():
                value = int(value)
            else:
                pass
            data_set[key].append(value)
df = {k: tuple(v) for k, v in data_set.items()}
data = pd.read_csv("Data Set.csv")


def cat_or_num(var):
    return 'Numerical' if isinstance(var, (int, float)) else 'Categorical'


for i in df:
    print("The variable ", i, " is", cat_or_num(df[i][0]))

print("\nContingency table of at most 2 different categorical variables ->\n")
print(pd.crosstab(data.Species, data.SepalLengthCm, margins=True))
print("\n")

def mean(arr):
    return round(sum(arr)/150)


def median(arr):
    arr = sorted(arr, reverse=False)
    if len(arr) % 2 != 0:
        return arr[int(len(arr) / 2)]
    else:
        first_i = arr[int(len(arr) / 2) - 1]
        second_i = arr[int(len(arr) / 2)]
        return sum((first_i, second_i)) / 2


def variance(arr):
    mean = sum(arr) / len(arr)
    summation = sum([(a - mean) ** 2 for a in arr])
    n = len(arr) - 1
    return round(summation / n, 2)


def standard_dev(arr):
    return round(variance(arr) ** (1 / 2), 2)


def inter_quartile_range(arr):
    arr = sorted(arr, reverse=False)
    if len(arr) % 2 != 0:
        mid_i = int(len(arr) / 2)
        arr_1 = arr[:mid_i]
        arr_2 = arr[mid_i + 1:]
        arr_1_mid = arr_1[int(len(arr_1) / 2)]
        arr_2_mid = arr_2[int(len(arr_2) / 2)]
        return arr_2_mid - arr_1_mid
    else:
        first_i = int(len(arr) / 2)
        second_i = int(len(arr) / 2)
        arr_1 = arr[:first_i]
        arr_1_mid_1 = arr_1[int(len(arr_1) / 2) - 1]
        arr_1_mid_2 = arr_1[int(len(arr_1) / 2)]
        arr_1_mid = sum([arr_1_mid_1, arr_1_mid_2]) / 2
        arr_2 = arr[second_i:]
        arr_2_mid_1 = arr_2[int(len(arr_2) / 2) - 1]
        arr_2_mid_2 = arr_2[int(len(arr_2) / 2)]
        arr_2_mid = sum([arr_2_mid_1, arr_2_mid_2]) / 2
        return arr_2_mid - arr_1_mid


c = 0
for i in df:
    if 5 > c > 0:
        print(i)
        print("Mean -> ", mean(df[i]))
        print("Median -> ", median(list(df[i])))
        print("Variance -> ", variance(list(df[i])))
        print("Standard Deviation ->", standard_dev(list(df[i])))
        print("Inter Quartile Range ->", inter_quartile_range(list(df[i])))
        print()
    c+=1


def binary_nominal_ordinal(arr):
    arr = list(arr)
    n = len(arr)
    unique_n = len(set(arr))
    # check if binaries
    unique_01_check = set("".join(arr))
    if unique_01_check == {'0', '1'} or unique_01_check == {'0'} or
unique_01_check == {'1'}:
        return "BINARY"
    # check if nominal or ordinal
    uniq_vars = dict(zip(tuple(set(arr)), range(1, unique_n + 1)))
    encoded_arr = (uniq_vars[x] for x in arr)
    ordinal_checker = [next(encoded_arr)]
    for x in encoded_arr:
        prev_element_check = ordinal_checker[len(ordinal_checker) - 1]
        if x == prev_element_check:
            ordinal_checker.append(x)
        else:
            if x < prev_element_check:
                order = 'descending'
            else:
                order = 'ascending'
            if order == 'descending':
                if x <= prev_element_check:
                    ordinal_checker.append(x)
                else:
                    return "NOMINAL"
            else:
                if x >= prev_element_check:
                    ordinal_checker.append(x)
                else:
                    return "NOMINAL"
    return "ORDINAL"

    # print(ordinal_checker)


print("The categorical variable Species here is",
binary_nominal_ordinal(df['Species']))
```

OUTPUT



```
D:\aakri\PycharmProjects\semV\venv\Scripts\python.exe "D:/aakri/PycharmProjects/semV/Data Science/Lab 1.py"
The variable  Id  is Numerical
The variable  SepalLengthCm  is Numerical
The variable  SepalWidthCm  is Numerical
The variable  PetalLengthCm  is Numerical
The variable  PetalWidthCm  is Numerical
The variable  Species  is Categorical

Contingency table of at most 2 different categorical variables ->

SepalLengthCm    4.3  4.4  4.5  4.6  4.7  4.8  ...  7.3  7.4  7.6  7.7  7.9  All
Species                                        ...
Iris-setosa        1    3    1    4    2    5  ...    0    0    0    0    0   50
Iris-versicolor    0    0    0    0    0    0  ...    0    0    0    0    0   50
Iris-virginica     0    0    0    0    0    0  ...    1    1    1    4    1   50
All                1    3    1    4    2    5  ...    1    1    1    4    1  150

[4 rows x 36 columns]


SepalLengthCm
Mean ->  6
Median ->  5.8
Variance ->  0.69
Standard Deviation -> 0.83
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help semV - Lab 1.py
semV 〉 Data Science 〉 Lab 1.py
Lab 1
Project
Simplified DES.py ×
Lab 1.py ×
Run: Lab 1 ×
Run Problems Terminal TODO Python Console
Event Log
PyCharm and plugin updates: The IDE is ready for update. Restart? (today 21:09)
110:14 CRLF UTF-8 4 spaces Python 3.7 (semV)



```
Standard Deviation -> 0.83
Inter Quartile Range -> 1.3000000000000007

SepalWidthCm
Mean ->  3
Median ->  3.0
Variance ->  0.19
Standard Deviation -> 0.44
Inter Quartile Range -> 0.5

PetalLengthCm
Mean ->  4
Median ->  4.35
Variance ->  3.11
Standard Deviation -> 1.76
Inter Quartile Range -> 3.55

PetalWidthCm
Mean ->  1
Median ->  1.3
Variance ->  0.58
Standard Deviation -> 0.76
Inter Quartile Range -> 1.5

The categorical variable Species here is ORDINAL
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help semV - Lab 1.py
semV 〉 Data Science 〉 Lab 1.py
Lab 1
Project
Simplified DES.py ×
Lab 1.py ×
Run: Lab 1 ×
Run Problems Terminal TODO Python Console
Event Log

CONCLUSION

The experiment on Command Separated value was performed successfully and given operations were done on it.