Introduction

This assignment report consists of three sections which are following:

First section is about hypothesis testing,

The second section is about correlation analysis.

The third section is about the Linear Regression Model .

The fourth and last section is about Report writing in which all questions are answered properly.

First section (30 marks)

This section contains hypothesis testing.

Hypothesis testing:

Hypothesis testing is a statistical method that is used in making statistical decisions using experimental data. Hypothesis Testing is basically an assumption that we make about the population parameter.

Hypothesis testing is an essential procedure in statistics. A hypothesis test evaluates two mutually exclusive statements about a population to determine which statement is best supported by the sample data. When we say that a finding is statistically significant, it's thanks to a hypothesis test

Techniques used in hypothesis testing:

- T Test (Student T test)
- Z Test
- ANOVA Test
- Chi-Square Test

Which is important parameter of hypothesis testing?

Null hypothesis:-

In inferential statistics, the null hypothesis is a general statement or default position that there is no relationship between two measured phenomena, or no association among groups

In other words it is a basic assumption or made based on domain or problem knowledge.

Example:

A company production is = 50 unit/per day etc.

Alternative hypothesis:-

The alternative hypothesis is the hypothesis used in hypothesis

Testing that is contrary to the null hypothesis. It is usually taken to be that the observations are the result of a real effect (with some amount of chance variation superposed)

Example: a company production is !=50 unit/per day etc.

Analysis:

Step 1:

Importing python libraries and dataset.

```
import seaborn as sns
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()
pd.options.mode.chained_assignment = None  # default='warn'
```

```
import scipy.stats as stats
```

```
df=pd.read_csv('dataset 10k.csv')
df
```

	month	state	permit	permit_recheck	handgun	long_gun	other	multiple	admin	prepawn_handgun	...	rentals_long_gun	private_sale_handgun
0	2022-02	Alabama	25401	499	21822.0	14541.0	1351.0	1260.0	0.0	13.0	...	0.0	28.0
1	2022-02	Alaska	301	0	2644.0	2178.0	348.0	202.0	0.0	0.0	...	0.0	2.0
2	2022-02	Arizona	2560	473	20150.0	9935.0	1690.0	1153.0	0.0	11.0	...	0.0	15.0
3	2022-02	Arkansas	1842	309	7780.0	5756.0	429.0	515.0	4.0	15.0	...	0.0	5.0
4	2022-02	California	15815	10550	36362.0	23017.0	4941.0	1.0	0.0	1.0	...	0.0	7638.0

Here above following python libraries are imported in program:

1)**pandas**:
pandas is used for data crunching.
2)**numpy** :
It is used for scientific computing it provides some mathematics and statistics like functions.
3)**Matplotlib:**
It is used for data Visualization For example to make Charts and graphs.
4)**Seaborn**:
It is also used for data Visualization For example to make Charts and graphs.

5)Scipy:

it is used for statistical operations.

Now here I am loading data. As data is in excel csv file so I am importing csv file with .csv file format.

Here df is a data frame which stores excel csv file .

Step 2 :

```
df[['handgun','long_gun']]=df[['handgun','long_gun']].fillna(0)
df[['handgun','long_gun']].isnull().sum()
```

```
handgun     0
long_gun    0
dtype: int64
```

As these two features are used in all process so null values of that features are filled with zero.

Sample selection:

```
sample_size=10
sample=np.random.choice(df['handgun'],sample_size)
sample
```

```
array([32684., 15736.,     0., 12032., 20325., 11775.,  1873.,  3651.,
           0.,     0.])
```

```
sample.mean()
```

```
9807.6
```

Here a sample containing 10 values are chosen for hypothesis testing.

The mean of sample is 9807.6.

T test:

A t-test is a type of inferential statistic which is used to determine if there is a significant difference between the means of two groups which may be related in certain features. It is mostly used when the data sets, like the set of data recorded as out come from flipping a coin a 100 times, would follow a normal distribution and may have unknown

Variances

T test is used as a Hypothesis testing tool, which allows testing of an assumption applicable to a population.

T-test has 2 types:

1. One sampled t-test

2. Two-sampled t-test.

Note: here one sample t test is used.

One sample t-test

It determines whether the sample mean is statistically different from a known or hypothesized population mean. The One Sample

T Test is a parametric test.

Example: - you have 10 ages and you are checking whether average age is 30 or not.

Test, whether the population mean, is less than 7000.

Hypothesis

H0: There is no significant mean difference i.e., μ = 7000
H1: The population mean is less than 7000. i.e., μ < 7000

Here H0 is null hypothesis and H1 is alternative hypothesis.

Here level of significance (alpha) is 0.05 or 5%.

```
from scipy.stats import ttest_1samp
```

```
ttest,p_value=ttest_1samp(sample,7000)
```

```
print(p_value)
```

```
0.435956749785101
```

```
if p_value < 0.05:    # alpha value is 0.05 or 5%
    print(" we are rejecting null hypothesis")
else:
    print("we are accepting null hypothesis")
```

```
we are accepting null hypothesis
```

As p value is greater than level of significance so null hypothesis is accepted .

So there is no significant difference between population mean and hypothesized mean.

Second section (30 marks)

This section contains correlation analysis.

Correlation analysis:

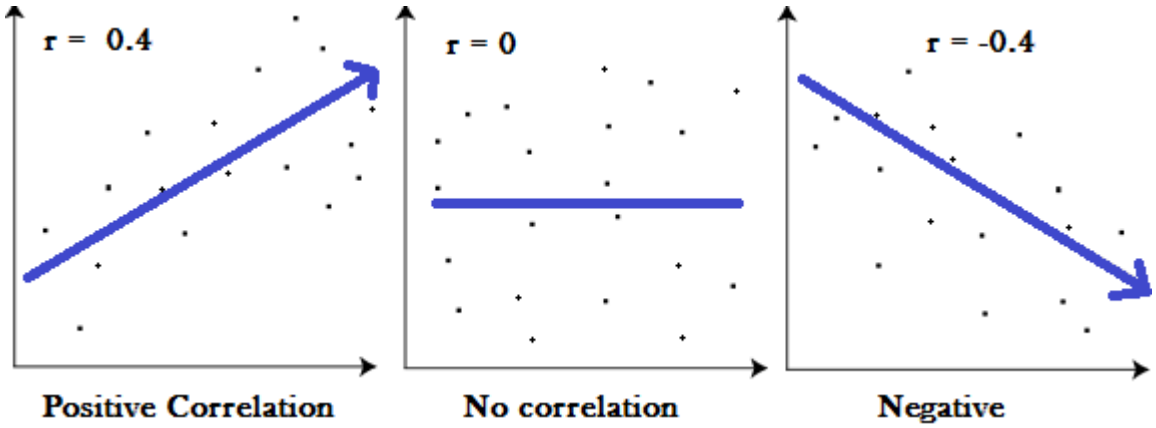
Correlation Analysis is statistical method that is used to discover if there is a relationship between two variables/datasets, and how strong that relationship.

It is basically measure of degree of relatedness of variables.

On the basis of strength it is divided as 5 types.

- Strong negative correlation
- Moderate negative correlation
- Strong positive correlation
- Moderate positive correlation
- No correaltion

A correlation coefficient is a way to put a value to the relationship. Correlation coefficients have a value of between -1 and 1. A “0” means there is no relationship between the variables at all, while -1 or 1 means that there is a perfect negative or positive correlation



The most common correlation coefficient is the Pearson Correlation Coefficient.

Note: in this section the Pearson Correlation Coefficient is used.

Correlation Coefficient using numpy

Here two variables which are hand gun and long_gun are used for correlation analysis so I have to find what type of relationship hand gun and long gun feature have?

```
np.corrcoef(df['handgun'],df['long_gun'])
```

```
array([[1.        , 0.64202673],
       [0.64202673, 1.        ]])
```

The Correlation Coefficient of handgun and long gun is 0.64 which show a strong positive relationship between handgun and long_gun feature.

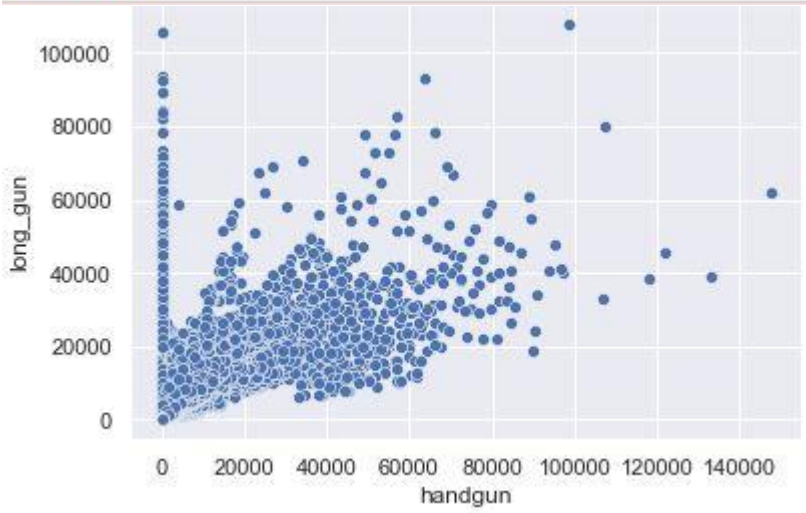
So this positive relationship states that if value of hand gun increased than it tends to increase in the value of long gun.

So let us understand the relationship using scatter plot

Scatterplot to show relationship:

Here scatterplot is used to show relationship between handgun and long gun.

```
plt.figure()
sns.scatterplot(df['handgun'],df['long_gun'])
plt.show()
```



As by looking on scatter plot positive relationship is shown between hand gun and long gun.

Correlation matrix:

This correlation matrix shows that how features are correlated with each other.

```
df.corr()
```

	handgun	long_gun	other	multiple	admin	prepawn_handgun	prepawn_long_gun	prepawn_other	redemption_handgun
handgun	1.000000	0.642027	-0.046369	0.410180	0.005636	0.174942	0.144965	-0.099046	0.574578
long_gun	0.642027	1.000000	0.040290	0.323256	0.033009	0.174003	0.211945	-0.065670	0.454828
other	-0.046369	0.040290	1.000000	0.265127	0.351968	0.169013	0.089596	0.506845	-0.010301
multiple	0.410180	0.323256	0.265127	1.000000	0.006929	0.176335	0.131481	0.096968	0.413633
admin	0.005636	0.033009	0.351968	0.006929	1.000000	-0.001965	-0.010323	0.267444	-0.013863
prepawn_handgun	0.174942	0.174003	0.169013	0.176335	-0.001965	1.000000	0.455603	0.011526	0.427479
prepawn_long_gun	0.144965	0.211945	0.089596	0.131481	-0.010323	0.455603	1.000000	0.167503	0.435746
prepawn_other	-0.099046	-0.065670	0.506845	0.096968	0.267444	0.011526	0.167503	1.000000	-0.014470
redemption_handgun	0.574578	0.454828	-0.010301	0.413633	-0.013863	0.427479	0.435746	-0.014470	1.000000
redemption_long_gun	0.329700	0.394597	0.120110	0.292816	0.018876	0.359803	0.592691	0.180640	0.806806
redemption_other	-0.186216	-0.128019	0.582129	0.014544	0.381092	-0.059524	0.070667	0.726447	-0.105828
returned_handgun	0.546791	0.377354	0.008250	0.251858	-0.011763	0.031398	0.149944	-0.036503	0.208334
returned_long_gun	0.135060	0.128364	0.332762	0.111447	0.058006	0.015953	0.077412	0.252077	-0.028785
returned_other	-0.026279	0.014632	0.317419	0.112512	0.110431	-0.022149	0.014765	0.124948	-0.054876
rentals_handgun	-0.050465	-0.044224	0.022303	-0.016736	0.100532	-0.014275	-0.022399	0.019037	-0.033204

```
df.corr()['handgun']['long_gun']
```

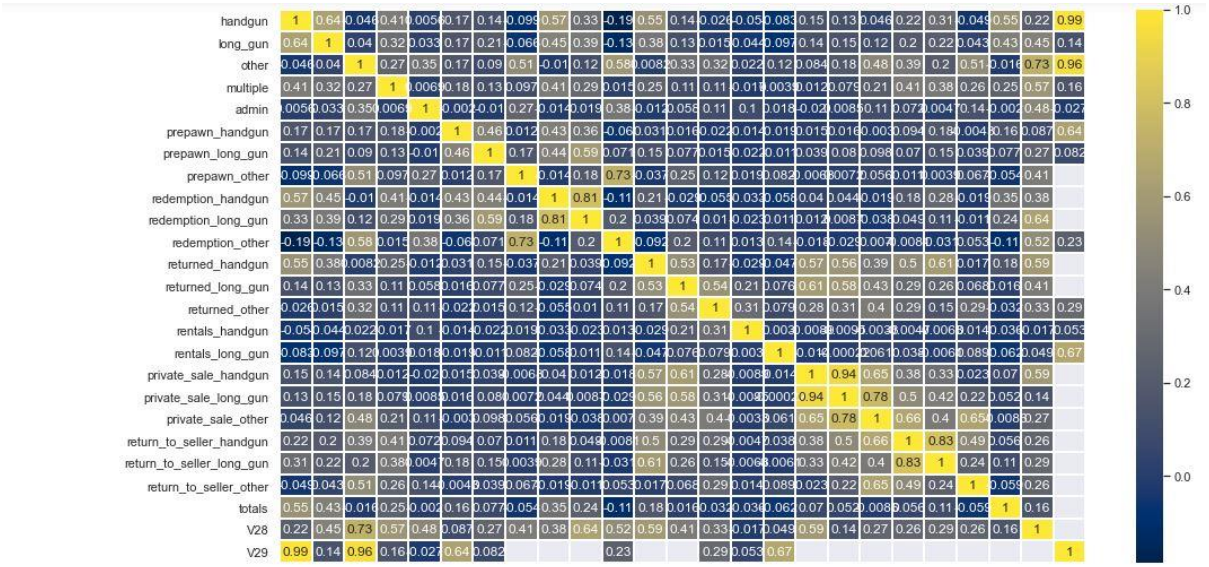
```
0.6420267291566981
```

So this method also shows that the correlation coefficient of hand gun and long gun is 0.64.

Heat map of correlation:

This is heat map of correlation which shows correlation coefficient between each other.

```
plt.figure(figsize=(16,9))
sns.heatmap(df.corr(),cmap ='cividis',linewidth = 0.30, annot = True)
plt.show()
```



Third section (30 marks)

This is Machine Learning based section in which I have to build Linear Regression model between chosen features and also I have to evaluate Linear Regression Model that how it performs?

So as in 2nd section hand gun and long gun are selected so in this section are used in Machine Learning

Splitting into dependent and independent feature:

Here below data frame is divided into dependent and Independent feature. Dependent feature is output for training process and Independent features are inputs for machine learning process.

```
X=df[['handgun']]
```

```
y=df['long_gun']
```

Here hand gun feature is independent feature or inputs and long gun is dependent feature or outputs so inputs are stored as X and output is stored as y.

Splitting into train and test:

Here data is further divided into train and test data as train data is used to train model and test data is used to test performance of model because test data is unseen to machine learning model.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=30)
```

Here train and test data is divided in the ratio of 80:20 means to say that 80 % of total data is training data and 20 % of total data is test data.

Building a Linear Regression Model:

Linear regression is one the Supervised Machine Learning technique used for regression. It is used for predicting the continuous dependent variable using a given set of independent variables. Linear regression predicts the output of a continuous dependent variable.

```
from sklearn.linear_model import LinearRegression
LR= LinearRegression()
LR.fit(X_train, y_train)
```

```
LinearRegression()
```

Accuracy score of Linear Regression Model:

Accuracy score or R2 score is a very important metric that is used to evaluate the performance of a regression-based machine learning model. It is pronounced as R squared and is also known as the coefficient of determination. It works by measuring the amount of variance in the predictions explained by the dataset

```
LR.score(X_test, y_test)
```

```
0.40764687089163965
```

So the accuracy score is 40% which is very low.

Mean Squared Error and Root Mean square error:

Root mean squared error (***RMSE***) is the square root of the mean of the square of all of the ***error***. The use of ***RMSE*** is very common

```
from sklearn.metrics import mean_squared_error
import numpy as np
predictions = LR.predict(X_test)
mse = mean_squared_error(y_test,predictions)
rmse = np.sqrt(mse)
print(mse)
print(rmse)
```

```
51794976.09900603
7196.872661024789
```

Here Mean Squared error is 51794976 and Root Mean Squared error is 7196.

Mean Absolute Error:

Mean absolute error refers to **the magnitude of difference between the prediction of an observation and the true value of that observation**. MAE takes the average of absolute errors for a group of predictions and observations as a measurement of the magnitude of errors for the entire group.

```
from sklearn.metrics import mean_absolute_error
mean_absolute_error(y_test,predictions)
```

```
3875.860363266269
```

Mean Absolute Error is 3875.

Conclusion:

T test is used as a Hypothesis testing tool, which allows testing of an assumption applicable to a population.

Here hand gun feature is chosen for hypothesis testing.

One sample T Test is used to find whether the population mean, is less than 7000 or not?

So null hypothesis and alternative hypothesis are:

- H0: There is no significant mean difference i.e., μ = 7000
- H1: The population mean is less than 7000. i.e., μ < 7000

But null hypothesis is accepted here because P value is greater than level of significance so there is no significant difference between hypothesized mean and population mean.

Correlation is the measure of degree of relatedness of variables. Here hand gun and long gun are chosen for correlation analysis.

The correlation coefficient of hand gun and long gun is 0.64 which shows a strong positive relationship between hand gun and long gun.

In 3rd section Linear regression model is applied which have very poor accuracy which is 40%.

The root mean and Mean absolute error are also high.

Reference list

- Hypothesis testing in Machine learning using Python by Yogesh Agrawal Published in Towards Data Science. https://towardsdatascience.com/hypothesis-testing-in-machine-learning-using-python-a0dc89e169ce

- What Do Correlation Coefficients Positive, Negative, and Zero Mean? By Steven Nickolas published on investopedia https://www.investopedia.com/ask/answers/032515/what-does-it-mean-if-correlation-coefficient-positive-negative-or-zero.asp
- Linear Regression for Machine Learning by Jason Brownlee published on Machine Learning Mastery! https://machinelearningmastery.com/linear-regression-for-machine-learning/