CSE464: Advanced Database Systems
[Fall 2021]

Project Report
Group No. – 04 (Section 1)

Submitted by:

Student ID	Student Name	Contribution Percentage	Signature
2017-1-60-091	Sk. Amir Hamza	25	Hamza
2017-1-60-008	Md. Maruf Hassan	20	Hassan
2018-1-60-089	Sanjida Akter	23	Sanjida
2018-1-60-152	Ibrahim Khalil Mehedi	17	Mehedi
2018-1-60-085	Md. Abdullah Al Maruf	15	Maruf

Introduction

In our project we estimated the number of deaths per day based on how many people have given both doses.To do this we are using the Linear regression model.
The linear regression shows us if the data has a positive trend or a negative trend, improving or worsening, increase or decrease. By using this method, the government can make a decision from the trend data they have. If we look at the data well, we can see that the number of deaths decreases with the vaccination.If we want to see how many people can die with a single day of vaccination, then we have to deal with the linear model. Because in this model having an independent and dependent variable where the dependent variable is our target value which is dependent on the independent variable .So we have this data where independent variable is the number of first and second dose and dependent variable is number of death cases.
Because of this model, we are able to find out from the dataset how many people died after being vaccinated. As a result, the death toll is increasing or decreasing as a result of the person's vaccine, comparing the actual number of deaths with the predicted number of deaths. So this model gives a good performance for what we are going to find out.

Data Preprocessing

We had three separate tables which are covid_dataset, covid_first_dose, covid_second_dose.
At first we merged these three tables into one table. Which includes no_first,no_second, death.

no_first: In this attribute the first dose count for each day contains the sum of each previous day count including the current day.
no_second: In this attribute the second dose count for each day contains the sum of each previous day count including the current day.
death: This attribute contains the number of deaths per day.

We created three tables by using SQL from three csv files. Then we merged them into one table using SQL. We applied a nested query for fetching total first dose and total second dose and applied **cumulative sum** . As the first dose and second were not given simultaneously that's why we have to put some null values where second doses were not given. We replaced those null values with '0' by using a SQL function "COALESCE". Then we dropped three attributes which are Day, Lab Test, Confirmed Case using SQL SELECT. Then we exported the newly created view table in csv.

	NO_FIRST	NO_SECOND	DEATH
0	26	0	17
1	567	0	15
2	567	0	7
3	567	0	17
4	567	0	16
...	...	...	...
323	68001893	44743382	3
324	68001893	44743382	2
325	68365432	45670638	4
326	68680984	45759687	1
327	68680984	45759687	2

328 rows × 3 columns

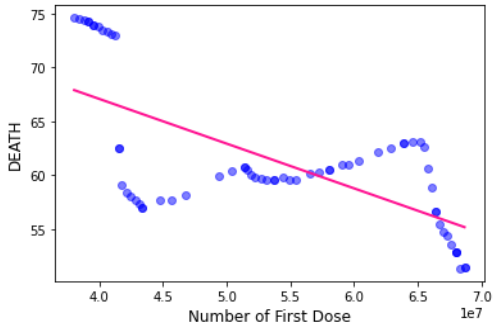
Machine Learning Models

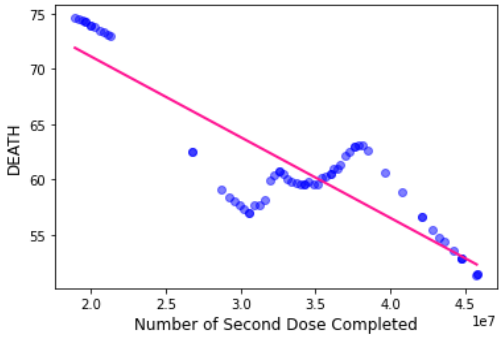
We used a linear regression model in our data set. The main reason for using this model is, the data set we are working on is a continuous data set. Linear regression is mainly used when we try to determine any probability or prediction. For analysis purposes, we can use linear regression to show the trend line of data. The linear regression shows us if the data has a positive trend or a negative trend, improving or worsening. The result of linear regression helps us to make a decision over a situation. For example, in this project our goal is to determine the number how many people could die in covid? For that we are observing the data regarding the first dose and second dose of covid vaccine. The regression will show the prediction and based on that we could decide whether we should boost up the vaccination process to reduce the death or we should proceed as it is.

Performance Evaluation

We have divided our data for training and testing. After training the model we applied testing over the data. The graphical representation of below figures indicates the regression line of death count after completion of first vaccine dose and second vaccine dose. From the figure shown below, we can see that when the second dose of vaccination started the point which indicates the number of deaths started to lean towards the regression line, that means the number of deaths started to decrease.

Coefficients: 9.61442817e-07 , -1.95973812e-06
Mean squared error: 3264.11
Coefficient of determination: -410.32





Discussion

It is used to help model and understand real data ,which is easy to use and to understand perceptively. It allows prediction of future outputs from the case we are modeling.we can also present a number of possible pitfalls when using linear regression.

The dataset that we have is the continuous dataset.By using the linear model on the continuous dataset,we get the number of deaths per day. But if we tried to use another model on this continuous dataset but we do not get the output we want.This is because the output of this model does not match our hypothesis.But according to our hypothesis,how many people will die if given the vaccine. But this prediction number which we will not get valid results (mostly negative value) if we use other models.It will give us wrong output which can not predict whether the tendency to give vaccines in the future will increase or decrease.

Conclusion

It was an interesting project for us, we used some amount of real data of covid 19, if we could use more data then we would find more accuracy on our figure. This project will help us to find near future situations of death affected by covid 19. In this project, our challenging part was data maintaining because all input data are continuous data. We learned data mining by using PL-SQL. For this project, we have an opportunity which is we could use this system for other countries covid death situations also.

References

1. https://medium.com/swlh/linear-regression-in-sql-is-it-possible-b9cc787d622f
2. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html
3. https://docs.oracle.com/en/database/oracle/machine-learning/oml4sql/21/dmcon/regression.html#GUID-51A08CFC-1487-4887-AB47-794C50D67358
4. https://www.researchgate.net/publication/324944461_Linear_regression_analysis_study

Appendix

covid_death_predict.sql

—----------

```
SET VERIFY OFF;
SET SERVEROUTPUT ON;


– merge three table for make dataset

CREATE OR REPLACE VIEW DEATH_PREDICT AS (SELECT * from (SELECT
(select sum(first_dose.no_vaccine) from first_dose where first_dose.day <= covid.day)
As NO_FIRST,
 COALESCE((select sum(second_dose.no_vaccine) from second_dose
where second_dose.day <= covid.day),0)
AS NO_second, death FROM covid) T
where T.no_first is not null );
/

SELECT * FROM DEATH_PREDICT;
```

```
CREATE OR REPLACE VIEW TRAIN_DATA_COVID AS SELECT * FROM DEATH_PREDICT SAMPLE (70)
SEED (1);
CREATE OR REPLACE VIEW TEST_DATA_COVID AS SELECT * FROM DEATH_PREDICT MINUS
SELECT * FROM TRAIN_DATA_COVID;
/
SELECT *FROM TRAIN_DATA_COVID;
SELECT *FROM TEST_DATA_COVID;

– check if 'LINEAR_REGRESSION' model exists then drop this model
BEGIN DBMS_DATA_MINING.DROP_MODEL('LINEAR_REGRESSION');
EXCEPTION
   WHEN OTHERS THEN
      NULL;
END;
/

– Make 'LINEAR_REGRESSION' for prediction
DECLARE
   setting  DBMS_DATA_MINING.SETTING_LIST;

BEGIN
   setting('PREP_AUTO') := 'ON';
   setting('ALGO_NAME') := 'ALGO_GENERALIZED_LINEAR_MODEL';
   setting('GLMS_DIAGNOSTICS_TABLE_NAME') := 'GLMR_SH_SAMPLE_DIAG_LINEAR';
   setting('GLMS_RIDGE_REGRESSION') := 'GLMS_RIDGE_REG_ENABLE';


   DBMS_DATA_MINING.CREATE_MODEL2(
      MODEL_NAME          => 'LINEAR_REGRESSION',
      MINING_FUNCTION     => 'REGRESSION',
      DATA_QUERY          => 'SELECT * FROM TRAIN_DATA_COVID',
      SET_LIST            => setting,
      CASE_ID_COLUMN_NAME => 'NO_FIRST',
      TARGET_COLUMN_NAME  => 'DEATH'
   );
END;
/

-- create DEATH_prediction_linear model
CREATE OR REPLACE VIEW DEATH_prediction_linear AS
   SELECT NO_FIRST,NO_SECOND, round(PREDICTION(LINEAR_REGRESSION USING *))
PREDICTION_DEATH, DEATH ACTUAL_DEATH
   FROM TEST_DATA_COVID;
/

--  call DEATH_prediction_linear model in test dataset
DECLARE
CURSOR c1 IS
      SELECT *
      FROM DEATH_prediction_linear;


BEGIN
 FOR record IN c1
   LOOP
      dbms_output.put_line(record.NO_FIRST || ':' || record.NO_SECOND || ' ' || record.PREDICTION_DEATH
```

```
      ||'---' || record.ACTUAL_DEATH);

   END LOOP;

END;
/
```

covid_death_predict.py

```
—----------------------
import numpy as np
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt

dataset = pd.read_csv('covid_combine_dataset.csv')

X = dataset
y = dataset['DEATH']
X = X.drop('DEATH', axis=1)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state = 42,shuffle=False)

reg = LinearRegression().fit(X_train, y_train)
reg.score(X, y)
pri = reg.predict(X_test)

# Make predictions using the testing set
predictData = reg.predict(X_test)

# The coefficients
print("Coefficients: \n", reg.coef_)
# The mean squared error
print("Mean squared error: %.2f" % mean_squared_error(y_test, predictData))
# The coefficient of determination: 1 is perfect prediction
print("Coefficient of determination: %.2f" % r2_score(y_test, predictData))

x = np.array(X_test['NO_FIRST'])
y = np.array(predictData)
plt.plot(x, y, 'o', color='blue', alpha=0.5)
m, b = np.polyfit(x, y, 1)
plt.plot(x, m*x + b, color='deeppink', linewidth=2)
plt.xlabel('Number of First Dose', size=12)
plt.ylabel('DEATH', size=12);
plt.show()

x = np.array(X_test['NO_SECOND'])
```

```
y = np.array(predictData)
plt.plot(x, y, 'o', color='blue', alpha=0.5)
m, b = np.polyfit(x, y, 1)
plt.plot(x, m*x + b, color='deeppink', linewidth=2)
plt.xlabel('Number of Second Dose Completed', size=12)
plt.ylabel('DEATH', size=12);
plt.show()
```