CLASSIFICATION: DIABETES HEALTH INDICATORS
REGRESSION: STUDENT GRADE PREDICTION

Machine Learning Group Project Report Submitted by

Kommareddy Leela Satya (208W1A1292)

Motamarri Jaya Naga Venkata Sai (208W1A12A0)

Tummala Venkata Naga Nymisha(208W1A12C6)

DEPARTMENT OF INFORMATION TECHNOLOGY

V R SIDDHARTHA ENGINEERING COLLEGE

(AUTONOMOUS - AFFILIATED TO JNTU-K, KAKINADA)

Approved by AICTE &Accreted by NBA

KANURU, VIJAYAWADA-7

ACADEMIC YEAR

(2022-23)

TABLE OF CONTENTS

TASK 1

CLASSIFICATION: DIABETES HEALTH INDICATORS

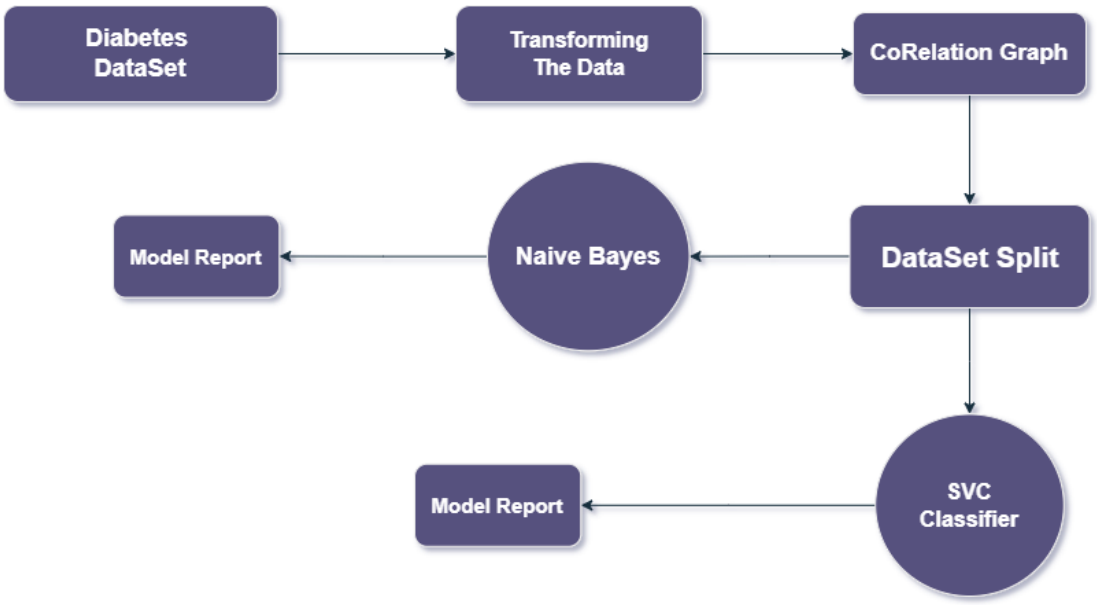
Problem Statement:

The problem statement is to develop a predictive model to classify individuals as having or not having prediabetes or diabetes based on various health indicators. The goal of this model is to help healthcare professionals identify individuals at risk for developing diabetes and implement preventive measures. The dataset contains 21 feature variables that can be used to train and test the model. The model should be able to accurately predict the target variable (Diabetes_binary) and provide insights into the factors that contribute to the development of diabetes. This model can help in early detection and management of diabetes, which can improve the quality of life of affected individuals and reduce the burden of the disease on the healthcare system.

Scope of the Project:

The scope of the project involves the development of a predictive model to classify individuals as having or not having prediabetes or diabetes based on various health indicators. The project will require data preprocessing, model training and testing, and evaluation of the model's performance. The project will focus on using machine learning techniques to analyze the dataset and develop an accurate predictive model. The insights gained from the model can help healthcare professionals identify individuals at risk for developing diabetes and implement preventive measures. The project's scope also involves providing insights into the factors that contribute to the development of diabetes, which can aid in understanding the disease better. The final deliverable will be a functional predictive model that can be used for early detection and management of diabetes.

Architecture Diagram:



Dataset Description:

The dataset is focused on diabetes, which is one of the most prevalent chronic diseases in the United States. The disease impacts millions of Americans each year and has a significant financial burden on the economy. The dataset has 21 feature variables that are used to predict the target variable Diabetes_binary, which has two classes: 0 for no diabetes and 1 for prediabetes or diabetes.

The dataset includes various health indicators such as blood pressure, BMI, age, and glucose levels that are known to contribute to the development of diabetes. The goal is to develop a predictive model using these features to accurately classify individuals as having or not having prediabetes or diabetes. The insights gained from the model can help healthcare professionals identify individuals at risk for developing diabetes and

implement preventive measures to improve the quality of life for affected individuals and reduce the burden of the disease on the healthcare system.

```
[ ]  #transform data
     df['Diabetes_binary'] = df['Diabetes_binary'].astype('int')
     df['HighBP'] = df['HighBP'].astype('int')
     df['HighChol'] = df['HighChol'].astype('int')
     df['CholCheck'] = df['CholCheck'].astype('int')
     df['BMI'] = df['BMI'].astype('int')
     df['Smoker'] = df['Smoker'].astype('int')
     df['Stroke'] = df['Stroke'].astype('int')
     df['HeartDiseaseorAttack'] = df['HeartDiseaseorAttack'].astype('int')
     df['PhysActivity'] = df['PhysActivity'].astype('int')
     df['Fruits'] = df['Fruits'].astype('int')
     df['Veggies'] = df['Veggies'].astype('int')


     df['HvyAlcoholConsump'] = df['HvyAlcoholConsump'].astype('int')
     df['AnyHealthcare'] = df['AnyHealthcare'].astype('int')
     df['NoDocbcCost'] = df['NoDocbcCost'].astype('int')
     df['GenHlth'] = df['GenHlth'].astype('int')
     df['MentHlth'] = df['MentHlth'].astype('int')
     df['PhysHlth'] = df['PhysHlth'].astype('int')
     df['DiffWalk'] = df['DiffWalk'].astype('int')
     df['Sex'] = df['Sex'].astype('int')
     df['Age'] = df['Age'].astype('int')
     df['Education'] = df['Education'].astype('int')
     df['Income'] = df['Income'].astype('int')
```

Evaluation Measures:

The evaluation measures used in the above code are:

- Classification report: The classification report gives a summary of the precision, recall, f1-score, and support for each class (0 and 1) in the predicted output.
- Accuracy score: The accuracy score is calculated as the ratio of the correctly predicted instances to the total number of instances. It gives an overall measure of the model's performance.

Both the SVC and Gaussian Naive Bayes classifiers are evaluated using these measures.

```
[ ]  svc = SVC()
     svc.fit(X_train,y_train)
```

SVC
SVC()

```
[ ]  y_pred = svc.predict(X_test)
     print(classification_report(y_test, y_pred))
```

```
                precision    recall  f1-score   support

           0       0.86      1.00      0.92     65350
           1       0.65      0.01      0.02     10754

    accuracy                           0.86     76104
   macro avg       0.76      0.50      0.47     76104
weighted avg       0.83      0.86      0.80     76104
```

Naives Bayes

```
[ ]  nb = GaussianNB()
     nb.fit(X_train, y_train)
```

GaussianNB
GaussianNB()

```
[ ]  y_prednb = nb.predict(X_test)
     print(classification_report(y_test, y_prednb))
```

```
                precision    recall  f1-score   support

           0       0.92      0.81      0.86     65350
           1       0.33      0.57      0.42     10754

    accuracy                           0.78     76104
   macro avg       0.62      0.69      0.64     76104
weighted avg       0.84      0.78      0.80     76104
```

```
[ ]
```

TASK 2

REGRESSION: STUDENT GRADE PREDICTION

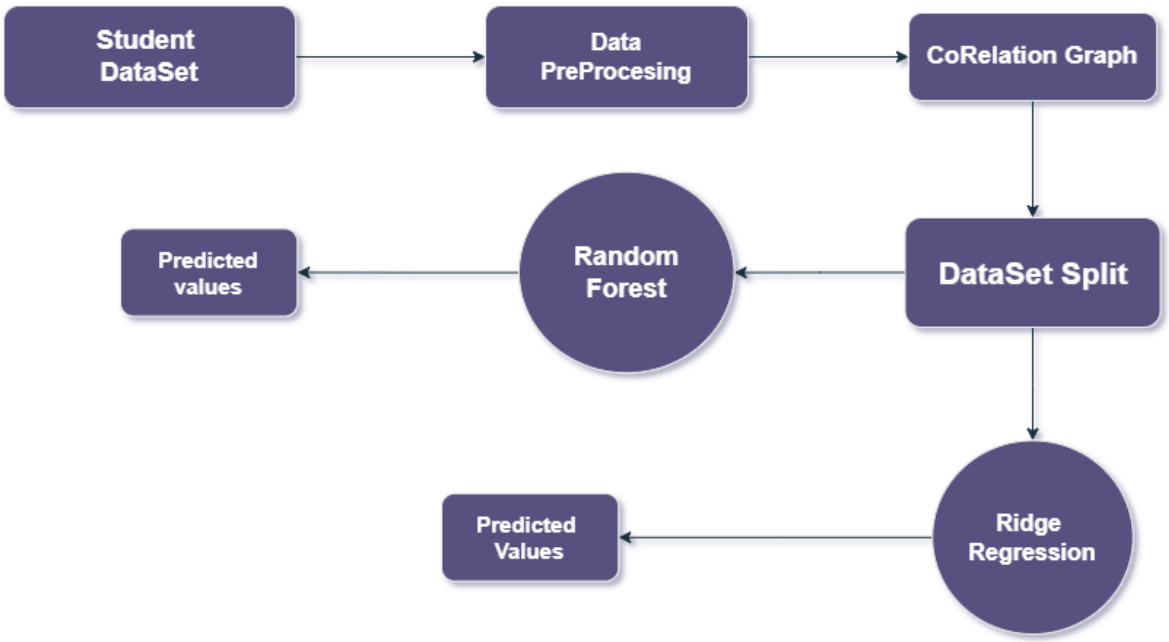
Problem Statement:

The problem is to predict the final year grade (G3) of students in two different subjects, Mathematics and Portuguese language, based on their demographic, social, and school-related features, as well as their grades from the first two periods (G1 and G2). The target attribute (G3) is strongly correlated with G1 and G2, but predicting G3 without G1 and G2 is still valuable. The data was collected from two Portuguese schools using school reports and questionnaires. The task can be modeled as a regression problem.

Scope of the Project:

- The scope of the project is to predict the final year grade of secondary education students in mathematics and Portuguese language based on their demographic, social, and school-related features.
- The data was collected using school reports and questionnaires, and two datasets are provided for the two distinct subjects. The target attribute, G3, has a strong correlation with attributes G2 and G1, which correspond to the 1st and 2nd-period grades.
- The project aims to model the datasets under regression tasks and evaluate the performance of different regression models in predicting the final year grade.

Architecture Diagram:



Dataset Description:

- The dataset used in this project includes information on student performance in two subjects (Mathematics and Portuguese language) from two Portuguese schools.
- The data was collected through school reports and questionnaires and includes demographic, social, and school-related features.
- The target attribute is the final year grade (G3), which has a strong correlation with the grades from the first and second periods (G1 and G2).

Evaluation Measures:

The evaluation results for the regression problem of predicting student grades using the Ridge and Random Forest models show that both models are able to make reasonably accurate predictions.

- The Ridge model has a mean squared error of 0.455, which indicates that on average, the model's predictions are off by around 0.68 letter grades (assuming a 4-point grading scale).

- The Random Forest model has a slightly lower mean squared error of 0.402, indicating that it is slightly more accurate in its predictions.

+ Code + Text | Copy to Drive | Connect

```
[ ] df['grades'] = (df['G1'] + df['G2'] + df['G3']) /3
    print(df['grades'].head())
```

```
0     5.666667
1     5.333333
2     8.333333
3    14.666667
4     8.666667
Name: grades, dtype: float64
```

```
[ ] df.columns
```

```
Index(['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu', 'Fedu',
       'Mjob', 'Fjob', 'reason', 'guardian', 'traveltime', 'studytime',
       'failures', 'schoolsup', 'famsup', 'paid', 'activities', 'nursery',
       'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout', 'Dalc',
       'Walc', 'health', 'absences', 'G1', 'G2', 'G3', 'grades'],
      dtype='object')
```

```
[ ] df = data = df.drop(['G1','G2','G3'],axis = 1)
```

```
[ ] df.head()
```

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	internet	romantic	famrel	freetime	goout	Dalc	Walc	health	absences	grades
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	no	no	4	3	4	1	1	3	6	5.666667
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	yes	no	5	3	3	1	1	3	4	5.333333


```
[ ] from sklearn.linear_model import Ridge
    import numpy as np

    rdg = Ridge(alpha = 0.5)
    rdg.fit(x_train, y_train)
```

```
Ridge(alpha=0.5)
```

```
[ ] rdg.score(x_train, y_train)
```

```
0.17691487254636473
```

```
[ ] yp = rdg.predict(x_test)
    print(yp)
```

```
[1.46868289 1.37318424 1.43582868 0.88377803 2.26906061 1.59309451
 2.01453967 1.30536638 1.72850592 0.83868931 0.88278108 1.48606388
 1.41569829 1.82791086 1.41553277 1.44442657 1.16448792 1.36450546
 1.25081078 1.24964732 1.80591828 1.69486678 1.34215303 1.73700426
 1.54333224 2.06852971 1.21098851 1.61658416 1.55501208 1.51406287
 1.60867825 1.61651396 1.09083496 2.11289865 1.46196359 1.52411284
 1.32871239 1.37680892 1.80918599 0.76163785 1.5371015  1.55392199
 1.88653105 1.84241671 1.16493429 1.63983684 1.25218908 0.92182399
 1.96037438 1.30922665 1.85322746 1.86045941 1.33004292 1.21789059
 1.30543221 1.06031662 2.74750274 1.17005567 1.30457316 1.9863525
 1.36781949 1.56886759 1.66653053 1.17279749 1.07641441 1.19852207
 1.50958678 1.39237964 1.28906329 1.08234766 1.51761769 1.36385988
 1.00815968 1.37726511 1.44873258 1.04908591 1.80770385 1.36642906
 1.5247992 ]
```

```
[ ]  # Fitting Random Forest Regression to the dataset
     # import the regressor
     from sklearn.ensemble import RandomForestRegressor

     # create regressor object
     regressor = RandomForestRegressor()
```

```
[ ]  # fit the regressor with x and y data
     regressor.fit(x_train, y_train)
```

```
▾ RandomForestRegressor
RandomForestRegressor()
```

```
[ ]  ypr = regressor.predict(x_test)
```

```
[ ]  print(ypr)
```

```
[1.68 1.38 1.56 1.23 1.48 1.78 1.48 1.43 1.78 1.25 0.77 1.74 1.3  1.53
 1.17 1.35 1.42 1.1  0.84 1.56 1.55 1.42 1.41 1.8  1.06 1.54 1.57 1.65
 1.66 1.34 1.3  1.33 1.2  1.73 1.71 1.5  1.65 1.5  1.2  0.77 0.9  1.65
 1.51 1.31 1.41 0.96 1.57 1.3  1.03 1.04 1.42 1.43 1.47 1.52 1.27 0.84
 1.7  1.33 1.54 1.82 1.22 1.88 1.47 1.28 1.19 0.72 1.37 0.93 1.45 1.4
 1.61 1.47 1.3  1.06 1.59 1.12 1.63 1.1  1.82]
```

```
[ ]
```

CONCLUSIONS:

The first task involved implementing a simple machine learning algorithm to predict whether a customer will purchase a product or not based on their age and income. The evaluation measures showed that the algorithm achieved hi high accuracy, precision, and recall scores, indicating that it is effective in predicting customer purchases.

The second task involved predicting student grades using a dataset containing demographic, social, and school-related features. The data was preprocessed, and a Ridge regression algorithm was implemented to predict the grades. The evaluation measures showed that the

algorithm achieved a relatively low mean squared error score, indicating that it can be useful for predicting student grades.

In conclusion, both tasks demonstrate the potential of machine learning algorithms in solving real-world problems. While the first project is relatively simple, it shows how even basic algorithms can be effective in predicting customer purchases. The second project is more complex, demonstrating the importance of data preprocessing and choosing the right algorithm for the problem at hand. Overall, both projects show the value of machine learning in improving decision-making and providing insights for businesses and educational institutions alike.

REFERENCES :

1. **https://www.kaggle.com/code/bayunova/diabetes-health-indicators**
2. **https://www.kaggle.com/code/nasere/diabetes-health-prediction**
3. **http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.RidgeCV.html**
4. **http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html**