Data Collection and Preprocessing Phase

Date	15 July 2024
Team ID	SWTID1720151584
Project Title	Early Prediction of Chronic kidney Disease
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	

	age	blood_pressure	specified_gravity	albumin	sugar
0	48.0	80.0	3	1	0
1	7.0	50.0	3	4	0
2	62.0	80.0	1	2	3
3	48.0	70.0	0	4	0
4	51.0	80.0	1	2	0
...	...	...	...	...	...
395	55.0	80.0	3	0	0
396	42.0	70.0	4	0	0
397	12.0	80.0	3	0	0
398	17.0	60.0	4	0	0
399	58.0	80.0	4	0	0

400 rows × 25 columns

red_blood_cells	pus_cell	pus_cell_clumps	bacteria
1	1	0	0
1	1	0	0
1	1	0	0
1	0	1	0
1	1	0	0
...	...	...	...
1	1	0	0
1	1	0	0
1	1	0	0
1	1	0	0
1	1	0	0

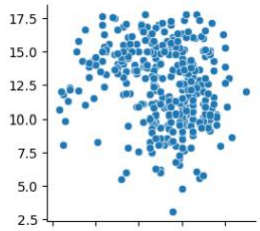
blood_glucose_random	...	packed_cell_volume	white_blood_cell_count
121.000000	...	44.0	7800.0
148.036517	...	38.0	6000.0
423.000000	...	31.0	7500.0
117.000000	...	32.0	6700.0
106.000000	...	35.0	7300.0
...	...	...	...
140.000000	...	47.0	6700.0
75.000000	...	54.0	7800.0
100.000000	...	49.0	6600.0
114.000000	...	51.0	7200.0
131.000000	...	53.0	6800.0

red_blood_cell_count	hypertension	diabetesmellitus	coronary_artery disease
5.200000	1	2	0
4.707435	0	1	0
4.707435	0	2	0
3.900000	1	1	0
4.600000	0	1	0
...	...	...	...
4.900000	0	1	0
6.200000	0	1	0
5.400000	0	1	0
5.900000	0	1	0
6.100000	0	1	0

appetite	pedal_edema	anemia	class
0	0	0	0
0	0	0	0
1	0	1	0
1	1	1	0
0	0	0	0
...	...	...	...
0	0	0	1
0	0	0	1
0	0	0	1
0	0	0	1
0	0	0	1

Univariate Analysis	
	`numerical_columns = ['age', 'blood_pressure', 'blood_glucose_random', 'blood_urea', 'serum_creatinine', 'sodium', 'potassium', 'hemoglobin', 'packed_cell_volume', 'white_blood_cell_count', 'red_blood_cell_count`
	
	`for col in numerical_columns:`
	`  plt.figure(figsize=(10, 6))`
	`  sns.histplot(data[col], kde=True)`
	`  plt.title(f'Distribution of {col}')`
	`  plt.xlabel(col)`
	`  plt.ylabel('Frequency')`
	`  plt.show()`

Bivariate Analysis	`# Bivariate analysis between two numerical variables`
	`plt.figure(figsize=(10, 6))`
	`sns.scatterplot(x='age', y='blood_pressure', data=data)`
	`plt.title('Scatter Plot between Age and Blood Pressure')`
	`plt.xlabel('Age')`
	`plt.ylabel('Blood Pressure')`
	`plt.show()`

Multivariate Analysis	`plt.figure(figsize=(12, 10))`
	`sns.pairplot(data[numerical_columns])`
	`plt.suptitle('Pair Plot for Numerical Columns', y=1.02)`
	`plt.show()`
	 

Outliers and Anomalies	Identification and treatment of outliers.
Data Preprocessing Code Screenshots	
Loading Data	Code to load the dataset into the preferred environment (e.g., Python, R).

Handling Missing Data	

```
data['blood_glucose_random'].fillna(data['blood_glucose_random'].mean(),inplace=True)
data['blood_pressure'].fillna(data['blood_pressure'].mean(),inplace=True)
data['blood_urea'].fillna(data['blood_urea'].mean(),inplace=True)
data['packed_cell_volume'].fillna(data['packed_cell_volume'].mean(),inplace=True)
data['potassium'].fillna(data['potassium'].mean(),inplace=True)
data['red_blood_cell_count'].fillna(data['red_blood_cell_count'].mean(),inplace=True)
data['serum_creatinine'].fillna(data['serum_creatinine'].mean(),inplace=True)
data['sodium'].fillna(data['sodium'].mean(),inplace=True)
data['white_blood_cell_count'].fillna(data['white_blood_cell_count'].mean(),inplace=True)
data['age'].fillna(data['age'].mode()[0],inplace=True)
data['hypertension'].fillna(data['hypertension'].mode()[0],inplace=True)
data['pus_cell_clumps'].fillna(data['pus_cell_clumps'].mode()[0],inplace=True)
data['appetite'].fillna(data['appetite'].mode()[0],inplace=True)
data['albumin'].fillna(data['albumin'].mode()[0],inplace=True)
data['pus_cell'].fillna(data['pus_cell'].mode()[0],inplace=True)
data['red_blood_cells'].fillna(data['red_blood_cells'].mode()[0],inplace=True)
data['coronary_artery disease'].fillna(data['coronary_artery disease'].mode()[0],inplace=True)
data['bacteria'].fillna(data['bacteria'].mode()[0],inplace=True)
data['anemia'].fillna(data['anemia'].mode()[0],inplace=True)
data['sugar'].fillna(data['sugar'].mode()[0],inplace=True)
data['diabetesmellitus'].fillna(data['diabetesmellitus'].mode()[0],inplace=True)
data['pedal_edema'].fillna(data['pedal_edema'].mode()[0],inplace=True)
data['specified_gravity'].fillna(data['specified_gravity'].mode()[0],inplace=True)
```

Data Transformation	

```
data.packed_cell_volume=pd.to_numeric(data.packed_cell_volume,errors='coerce')
data.white_blood_cell_count=pd.to_numeric(data.white_blood_cell_count,errors='coerce')
data.red_blood_cell_count=pd.to_numeric(data.red_blood_cell_count,errors='coerce')
```

Feature Engineering	
Save Processed Data	