

# FINAL PROJECT

**Student Name: Kritika Sejwal****UID: 24MCI10023****Section/Group: 24MAM-1/A****Branch: UIC****Date of Performance: 02-10-2024****Subject: Statistical Techniques using R****Subject Code: 24CAP-614****Semester: 1<sup>st</sup>**

## ○ Aim:

The aim of this practical is to perform regression analysis on the Pima Indians Diabetes dataset using R. We will create a linear regression model to predict glucose levels based on several independent variables. The model's performance will be evaluated through various metrics, and results will be interpreted.

## ○ Task to be done:

- Load the Pima Indians Diabetes dataset.
- Perform linear regression analysis.
- Evaluate the model's performance using RMSE, R-squared, and MAE.
- Visualize the residuals to check model assumptions.

## ○ Algorithm:

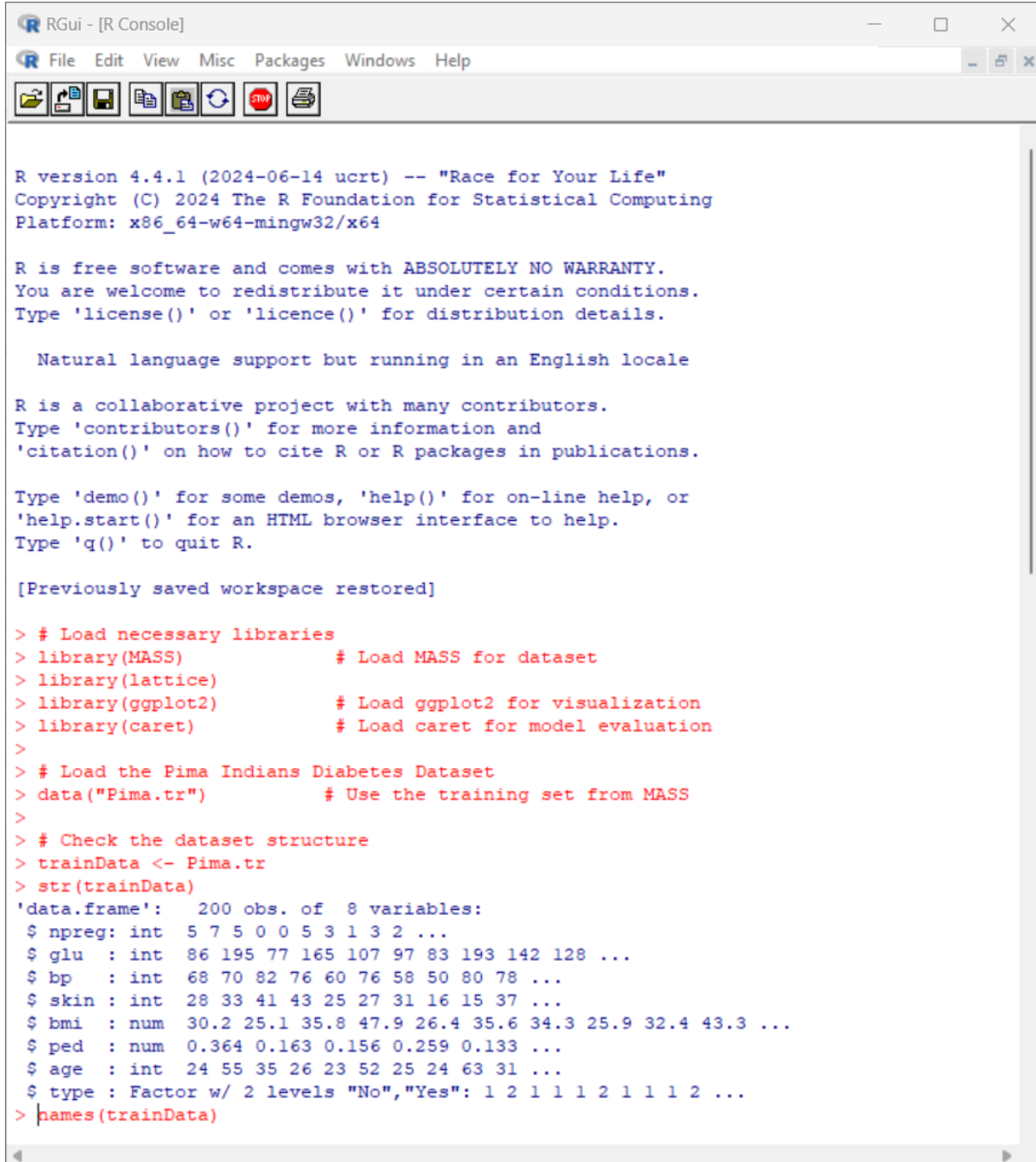
1. Load necessary libraries and the dataset.
2. Inspect the dataset structure and variables.
3. Define the dependent variable and independent variables.
4. Fit a linear regression model.
5. Summarize the model to interpret the results.
6. Calculate fitted values and residuals.
7. Create a residuals plot.
8. Evaluate model performance using RMSE, R-squared, and MAE.

### ○ Data Set:

**Dataset Used:** Pima Indians Diabetes dataset available in the `MASS` package. This dataset contains information on several health measurements.

- **Dependent Variable:** `glu` (Glucose level)
- **Independent Variables:**
  - `npreg`: Number of pregnancies
  - `bp`: Blood pressure
  - `skin`: Skin thickness
  - `bmi`: Body Mass Index
  - `ped`: Diabetes pedigree function
  - `age`: Age in years

### o Code for the experiment:



```
RGui - [R Console]

File Edit View Misc Packages Windows Help

R version 4.4.1 (2024-06-14 ucrt) -- "Race for Your Life"
Copyright (C) 2024 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> # Load necessary libraries
> library(MASS)           # Load MASS for dataset
> library(lattice)
> library(ggplot2)        # Load ggplot2 for visualization
> library(caret)          # Load caret for model evaluation
>
> # Load the Pima Indians Diabetes Dataset
> data("Pima.tr")         # Use the training set from MASS
>
> # Check the dataset structure
> trainData <- Pima.tr
> str(trainData)
'data.frame':   200 obs. of  8 variables:
 $ npreg: int   5  7  5  0  0  5  3  1  3  2 ...
 $ glu  : int  86 195 77 165 107 97 83 193 142 128 ...
 $ bp   : int  68 70 82 76 60 76 58 50 80 78 ...
 $ skin : int  28 33 41 43 25 27 31 16 15 37 ...
 $ bmi  : num  30.2 25.1 35.8 47.9 26.4 35.6 34.3 25.9 32.4 43.3 ...
 $ ped  : num  0.364 0.163 0.156 0.259 0.133 ...
 $ age  : int  24 55 35 26 23 52 25 24 63 31 ...
 $ type : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 2 1 1 1 2 ...
> names(trainData)
```

```

RGui - [R Console]
File Edit View Misc Packages Windows Help

> names(trainData)
[1] "npreg" "glu" "bp" "skin" "bmi" "ped" "age" "type"
>
> # Perform Linear Regression Model
> # Here we use 'glu' as the dependent variable
> linear_model <- lm(glu ~ npreg + bp + skin + bmi + ped + age, data = trainData)
>
> # Display model summary to interpret results
> summary(linear_model)

Call:
lm(formula = glu ~ npreg + bp + skin + bmi + ped + age, data = trainData)

Residuals:
    Min       1Q   Median       3Q      Max
-79.043 -16.394  -2.335   18.910   86.392

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  47.21980    16.31856   2.894 0.004246 **
npreg       -0.43140     0.77707  -0.555 0.579424
bp           0.35828     0.20231   1.771 0.078151 .
skin         0.07643     0.24304   0.314 0.753484
bmi          0.61516     0.46193   1.332 0.184522
ped          5.92841     6.98115   0.849 0.396821
age          0.86948     0.25351   3.430 0.000739 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 29.36 on 193 degrees of freedom
Multiple R-squared:  0.1661,    Adjusted R-squared:  0.1402
F-statistic: 6.408 on 6 and 193 DF,  p-value: 3.577e-06

> # Calculate fitted values and residuals
> fitted_values <- linear_model$fitted.values
> residuals <- linear_model$residuals
>
> # Create a data frame for ggplot
> residuals_data <- data.frame(Fitted = fitted_values, Residuals = residuals)
>
> # Plotting the residuals to check assumptions
> ggplot(data = residuals_data, aes(x = Fitted, y = Residuals)) +
+   geom_point() +

```

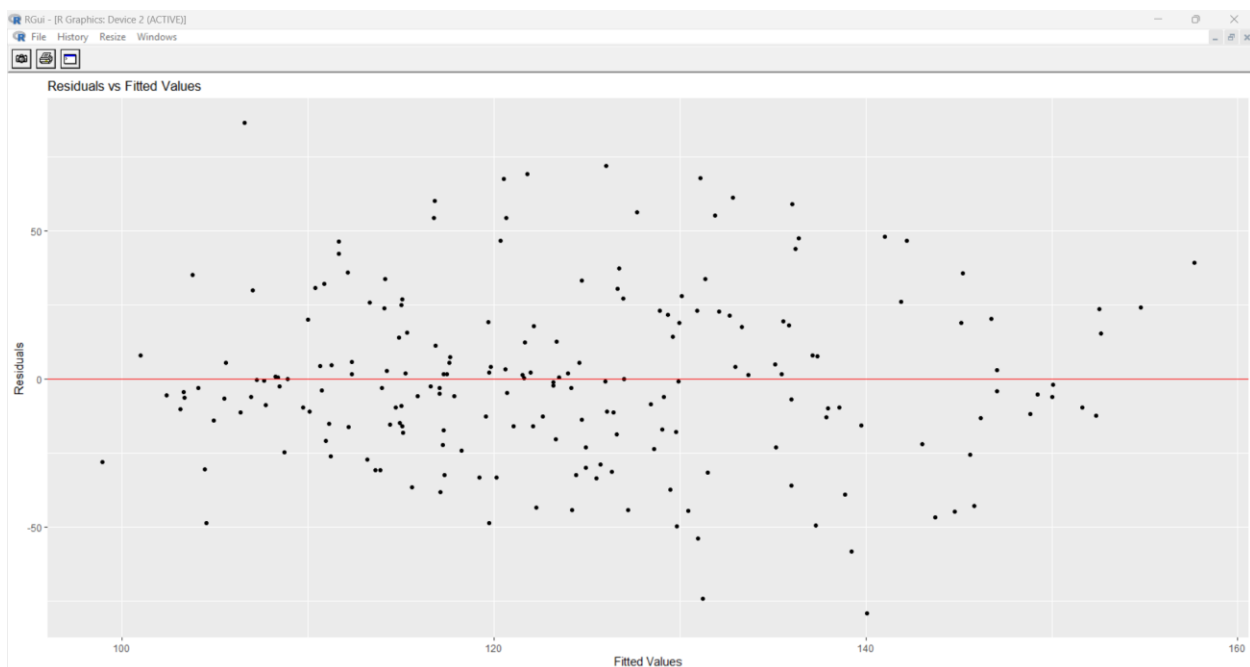
```

RGui - [R Console]
File Edit View Misc Packages Windows Help

> # Plotting the residuals to check assumptions
> ggplot(data = residuals_data, aes(x = Fitted, y = Residuals)) +
+   geom_point() +
+   geom_hline(yintercept = 0, color = "red") +
+   labs(title = "Residuals vs Fitted Values", x = "Fitted Values", y = "Residuals")
>
> # Evaluate model performance
> predictions <- predict(linear_model, newdata = trainData)
> model_performance <- postResample(predictions, trainData$glu)
>
> # Print model performance metrics
> print(model_performance)
      RMSE Rsquared      MAE
28.845220  0.166118 22.041044
> save.image("D:\\MCA(AI&ML)\\R Programming\\final_project_r")
>

```

## ○ Output:



### ○ Time and Space Complexities:

- **Time Complexity:** The linear regression fitting process generally runs in  $O(n \cdot p^2)$ , where  $n$  is the number of observations and  $p$  is the number of predictors.
- **Space Complexity:** The space complexity is primarily influenced by the storage of data points and coefficients, which is  $O(n+p)$ .

### ○ Model Performance Evaluation:

The output of the model performance evaluation provides three key metrics: Root Mean Squared Error (RMSE), R-squared (Rsquared), and Mean Absolute Error (MAE).

#### 1. **Root Mean Squared Error (RMSE)**

- **Value:** 28.845220
- **Interpretation:** RMSE is the square root of the average of the squared differences between the predicted values and the actual values. It provides a measure of how well the model's predictions match the actual data.
- **Lower Values:** A lower RMSE value indicates better predictive accuracy. In this case, an RMSE of approximately **28.85** suggests that, on average, the predicted glucose levels (glu) deviate from the actual glucose levels by about **28.85** units. Whether this is acceptable depends on the context and scale of your data.

#### 2. **R-squared (Rsquared)**

- **Value:** 0.166118
- **Interpretation:** R-squared indicates the proportion of the variance in the dependent variable (glu) that can be explained by the independent variables (npreg, bp, skin, bmi, ped, and age).
- **Range:** R-squared values range from **0 to 1**. A value closer to **1** indicates that a larger proportion of the variance is explained by the model, while a value closer to **0** suggests that the model does not explain much of the variance.
- **Conclusion:** An R-squared of approximately **0.166** means that only **16.6%** of the variance in glucose levels can be explained by the model. This indicates that the model may not fit the data well and that there could be other factors not included in the model that influence glucose levels.

#### 3. **Mean Absolute Error (MAE)**

- **Value:** 22.041044
- **Interpretation:** MAE measures the average magnitude of the errors in a set of predictions, without considering their direction. It is the average of absolute differences between predicted and actual values.

- **Lower Values:** Like RMSE, a lower MAE indicates better model accuracy. In this case, an MAE of approximately **22.04** means that the model's predictions differ from actual glucose levels by about **22.04** units on average.

### ○ Summary:

- The model shows a relatively high RMSE and MAE, suggesting significant prediction errors.
- The R-squared value is quite low, indicating that the independent variables included in the model do not explain much of the variance in glucose levels. This may suggest the need for additional predictors or transformations, or that the relationship between predictors and the response variable might be non-linear or complex.

### ○ Next Steps:

- **Feature Engineering:** Adding or transforming variables to capture more variance.
- **Modeling Techniques:** Trying different modeling approaches, such as polynomial regression, decision trees, or ensemble methods, to improve performance.
- **Data Exploration:** Conducting exploratory data analysis (EDA) to understand the data distribution and relationships better.

### ○ Conclusion:

This project demonstrated the application of linear regression on the Pima Indians Diabetes dataset to predict glucose levels based on various health indicators. The evaluation metrics, including RMSE, R-squared, and MAE, highlighted the model's limitations in accurately predicting glucose levels. The low R-squared value indicates that the model needs improvement, potentially through the inclusion of additional variables or more complex modeling techniques. This analysis emphasizes the importance of exploring various approaches in predictive modeling to enhance accuracy and gain better insights into the underlying data relationships.

### ○ Learning Outcomes:

1. **Understanding Regression Analysis:**
  - Gain a clear understanding of linear regression concepts, including the role of dependent and independent variables in predictive modeling.
2. **Data Preparation and Exploration:**



- Develop skills in loading, exploring, and preparing datasets for analysis in R, including checking data structure and variable types.
- 3. **Model Fitting and Interpretation:**
  - Learn how to fit a linear regression model in R using the `lm()` function and interpret the summary output, including coefficients, significance levels, and overall model fit.
- 4. **Model Evaluation Metrics:**
  - Understand key evaluation metrics such as RMSE, R-squared, and MAE, and how to calculate and interpret these metrics to assess model performance.
- 5. **Residual Analysis:**
  - Explore the importance of residual analysis in regression to check the validity of model assumptions, such as homoscedasticity and independence of errors.
- 6. **Data Visualization:**
  - Gain experience in visualizing data and model diagnostics using the `ggplot2` package, enhancing understanding through graphical representation.
- 7. **Critical Thinking and Problem Solving:**
  - Develop critical thinking skills by analyzing model performance results and identifying potential improvements or alternative modeling techniques.
- 8. **Exploring Advanced Techniques:**
  - Recognize the importance of exploring additional predictors or different modeling techniques to enhance predictive performance and address potential shortcomings of the initial model.
- 9. **Hands-on Experience with R:**
  - Gain practical experience in programming with R, including library management, data manipulation, model fitting, and visualization.
- 10. **Application of Statistical Techniques:**
  - Apply statistical techniques in a real-world context, understanding how data-driven decisions can impact health outcomes and research in diabetes.

○ **GITHUB:**

Link:

Screenshot: