REPORT ON

Problem Based Learning

Carried out on

PREDICTING MEDICAL INSURANCE CHARGES USING RBF REGRESSOR

Submitted to

NMAM INSTITUTE OF TECHNOLOGY, NITTE

(Deemed to Be University)

In partial fulfilment of the requirements for the award of the

Degree of Bachelor of Technology in

Robotics & Artificial Intelligence Engineering

By

Adhvaith P Kateel	**NNM22RI005**
Neha A Shetty	**NNM22RI032**
Neharika R	**NNM22RI033**
Nidheesh	**NNM22RI034**
Nihal E Praveen	**NNM22RI035**
P Kaushik	**NNM22RI036**

Under the guidance of

Dr. Rashmi P. Shetty

Department of Robotics & Artificial Intelligence Engineering

NITTE EDUCATION TRUST | NMAM INSTITUTE OF TECHNOLOGY

ABSTRACT

This project explores the use of a Radial Basis Function (RBF) network to predict medical insurance charges based on demographic and health-related factors such as age, BMI, smoking status, and region. The dataset was pre-processed using encoding and normalization techniques to ensure consistency. The model was trained and evaluated using performance metrics to assess its accuracy and generalization ability. Results indicate that the model effectively captures complex relationships within the data, demonstrating the potential of RBF networks in insurance cost estimation, risk assessment, and policy planning.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Predicting insurance costs is an important challenge in the healthcare and insurance industries. Our model explores how machine learning can be used to estimate insurance charges based on key factors like age, BMI, smoking status, and region.

To build an accurate model, the dataset is carefully processed, ensuring that categorical variables are encoded properly and numerical features are normalized for consistency. The **Radial Basis Function (RBF) model**, a powerful machine learning approach, is used to train and predict insurance costs by capturing complex patterns in the data.

To evaluate the model's effectiveness, we use performance metrics like Mean Squared Error (MSE) and R^2 score, which help us understand how well the model predicts real-world insurance costs. Now let us walk through the entire process from preparing the data to training the model and analysing the results, providing insights into how machine learning can improve insurance cost estimation.

1.1 : **ABOUT THE DATA**

The dataset was taken from Kaggle website. It is used in this analysis containing information related to individuals and their respective insurance charges. The data includes both categorical and numerical features that influence the cost of insurance premiums.

Features in the Dataset:

1. age – Age of the individual.
2. sex – Gender of the individual (encoded as male = 1, female = 0).
3. bmi – Body Mass Index (BMI), a measure of body fat based on height and weight.
4. children – Number of dependent children covered by the insurance policy.
5. smoker – Smoking status (encoded as smoker = 1, non-smoker = 0).
6. region – Geographic region where the individual resides (encoded into numerical values).
7. charges – The insurance cost (target variable).

Preprocessing Steps:

a. Categorical variables such as sex, smoker, and region were encoded using Label Encoding:

- If not encoded into categorical features, then the model won't work or will learn incorrectly. Encoding them as 0 and 1 (for binary variables) or using One-Hot Encoding (for multi-category variables) ensures correct learning and better performance.
- **CODE:**

```
label_encoders = {}
for col in df.select_dtypes(include=['object']).columns:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    label_encoders[col] = le
```

b. Numerical variables including age, BMI, children, and charges were normalized using StandardScaler to standardizes features by removing the mean and scaling to unit variance.

- It transforms the data to have a mean of 0 and a standard deviation of 1 and is useful for machine learning algorithms (especially those using distance-based methods like Kernel Ridge Regression and K-Means) to ensure that all features contribute equally and prevent dominance by larger-scale features.
- **CODE:**

```
scaler = StandardScaler()
data_scaled = scaler.fit_transform(df)
```

c. The dataset was then split into training and testing sets to evaluate the model performance effectively. 80% training and 20% testing split (random_state=42 for reproducibility).

- **CODE:**

```
X_train, X_test, y_train, y_test = train_test_split(data_scaled[:, :-1], data_scaled[:, -1],
test_size=0.2, random_state=42)
```

CHAPTER 2

LITERATURE REVIEW

[1] Marc Claesen, Johan A.K. Suykens

Claesen and Suykens used Support Vector Machines (SVMs) with RBF kernels and introduced a second-order Maclaurin series approximation to speed up predictions. Instead of relying on a large number of support vectors, their method focuses on input dimensions, reducing computation time significantly. For the model they tested to approach on datasets like a9a, MNIST, ijcnn1, sensit, and epsilon, showing high accuracy with minimal label differences. For example, a9a achieved 84.8%–85.0% accuracy with only 0.2%–3.5% differences, while MNIST reached 99.3% with just 0.08% differences. Other datasets also maintained over 86% accuracy, with minimal variations.

This method makes SVM models with RBF kernels much faster while keeping accuracy nearly the same. It is especially useful for large datasets and real-time applications like object detection. Additionally, it reduces model size, making it more efficient for practical use. This approach strikes a strong balance between speed and accuracy, making SVMs more effective for complex tasks**.**

[2] Ivan Izonin, Roman Tkachenko, Natalia Kryvinska

The study introduced a Committee of SGTM Neural-Like Structures with RBF kernels to enhance accuracy in insurance cost prediction. This approach improves the Successive Geometric Transformations Model (SGTM) by integrating RBF kernels, which enhance generalization and approximation capabilities. By dividing data into clusters and processing them through multiple SGTM structures, the proposed method achieves more accurate and efficient predictions. The model was evaluated using a real-world health insurance dataset containing 1,338 records with features such as age, gender, BMI, number of children, smoking status, and region. The dataset was split into 1,070 training samples and 268 test samples for evaluation. Experimental results demonstrated superior accuracy compared to conventional methods, with a Mean Absolute Percentage Error (MAPE) of 33.56%, outperforming the Stochastic Gradient Descent (SGD) Regressor (58.98%), Adaptive Boosting (58.61%), and Multi-Layer Perceptron (53.34%).

This methodology substantially improves both prediction accuracy and computational efficiency in insurance cost estimation. By leveraging SGTM Neural-Like Structures with RBF kernels, the approach achieves faster training times and more precise results compared to traditional models.

[3] Mohanarangan Veerappermal Devarajan, Rajya Lakshmi Gudivaka, Basava Ramanjaneyulu Gudivaka, Raj Kumar Gudivaka

The study applied Radial Basis Function Networks (RBFNs) combined with deep learning to analyse medical images such as X-rays, MRI, and CT scans for anomaly detection. The proposed model utilizes encoder-decoder layers to enhance efficiency while minimizing computational costs.

Evaluated on real-world healthcare data, the approach demonstrated strong performance across multiple metrics, achieving 80% operational reliability, 79% network speed, 82% accuracy, 84% energy efficiency, 83% classifier analysis, and 90% patient health performance. These results indicate that RBFNs outperform conventional diagnostic methods by offering improved accuracy, efficiency, and reduced energy consumption.

This methodology enhances early disease detection, pattern recognition, and real-time decision-making, making it a valuable tool for AI-driven healthcare applications. Its ability to process medical imaging data with high precision and efficiency highlights its potential for advancing modern diagnostic systems.

CHAPTER 3

METHODOLOGY

The predictive modeling in this study utilizes a Radial Basis Function (RBF) network, a type of artificial neural network, to estimate insurance charges based on input features. RBF networks use radial basis functions as activation functions to model complex nonlinear relationships. By transforming inputs into a higher-dimensional space, they enhance separability, improving learning, accuracy, and generalization, feature engineering with KMeans clustering improves model performance.

Techniques used in the model development are:

1. Data Pre-processing
2. Machine Learning Model - RBF Regressor
3. Model Training and Evaluation

[1] Data Preprocessing:

- Handled missing values by filling numerical features with the median and categorical features with the mode to maintain data integrity.
- Encoded categorical variables (e.g., sex, smoker status, region) into numerical values for compatibility with the model.
- Scaled numerical features using StandardScaler to normalize data and improve model performance. It maintains a normal distribution and handles outliers better compared to MinMaxScaler

[2] Machine Learning Model - RBF Regressor:

- Used the Radial Basis Function (RBF) model, which applies kernel transformations to capture complex, non-linear relationships between input features and insurance costs.
- Integrated Kernel Ridge Regression (KRR) to enhance stability and improve generalization.
- Optimized the gamma parameter in the RBF kernel to control how far influence extends across the feature space.
- KRR supports kernelized transformations, allowing us to apply the RBF kernel directly.
- RBF transforms the data → KRR learns the regression model in this transformed space.

Model Parameters:

- alpha = 0.001 (regularization parameter to prevent overfitting).
- gamma = 0.08 (controls spread of the RBF function).
- kernel = 'rbf' (applies the RBF transformation).

RBF Formula:

$$K(x, x') = \exp\left(-\gamma |x - x'|^2\right)$$

Feature Engineering with KMeans Clustering

- KMeans Clustering is used to identify hidden patterns in data. It Enhances feature transformation for RBF kernel regression and provides localized feature groupings that improve model predictions. For the model, 45 clusters were used to capture complex feature interactions which are determined using the Elbow Method, balancing variance capture and computational efficiency while avoiding over-segmentation.

[3] Model Training and Evaluation:

- Split the dataset into training (80%) and testing (20%) which sets to assess model generalization.
- Trained the model using RBF transformation and kernel regression techniques.
- Evaluated performance using:
    1. Mean Squared Error (MSE) to measure prediction error.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

    2. R^2 Score to assess how well the model explains variance in insurance charges.

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

CODE

```
# Handle missing values
for col in df.select_dtypes(include=['number']).columns:
    df[col].fillna(df[col].median(), inplace=True)
for col in df.select_dtypes(include=['object']).columns:
    df[col].fillna(df[col].mode()[0], inplace=True)
# Train a Kernel Ridge Regression model
num_clusters = 45
kmeans = KMeans(n_clusters=num_clusters, random_state=42, n_init=10)
kmeans.fit(X_train)
gamma_val = 0.08
X_train_rbf = rbf_kernel(X_train, kmeans.cluster_centers_, gamma=gamma_val)
X_test_rbf = rbf_kernel(X_test, kmeans.cluster_centers_, gamma=gamma_val)
# Train Kernel Ridge Regression with RBF kernel
model = KernelRidge(kernel='rbf', gamma=0.08, alpha=0.001)
model.fit(X_train_rbf, y_train)
# Make predictions
train_predictions = model.predict(X_train_rbf)
test_predictions = model.predict(X_test_rbf
# Compute error metrics
train_mse = mean_squared_error(y_train, train_predictions)
test_mse = mean_squared_error(y_test, test_predictions)
train_tse = np.sum((y_train - train_predictions) ** 2)
```

```
test_tse = np.sum((y_test - test_predictions) ** 2)

train_r2 = r2_score(y_train, train_predictions)

test_r2 = r2_score(y_test, test_predictions)

# Calculate errors for each data point

errors = y_test - test_predictions

squared_errors = errors ** 2

# Create a DataFrame to display results (first 10 values)

results_df = pd.DataFrame({

    'Actual': y_test[:40],

    'Predicted': test_predictions[:40],

    'Error': errors[:40],

    'Squared Error': squared_errors[:40]

})

print(results_df.to_string(index=False))

print(f"Training Mean Squared Error: {train_mse}")

print(f"Testing Mean Squared Error: {test_mse}")

print(f"Training R2 Score: {train_r2}")

print(f"Testing R2 Score: {test_r2}")
```

PIE CHART

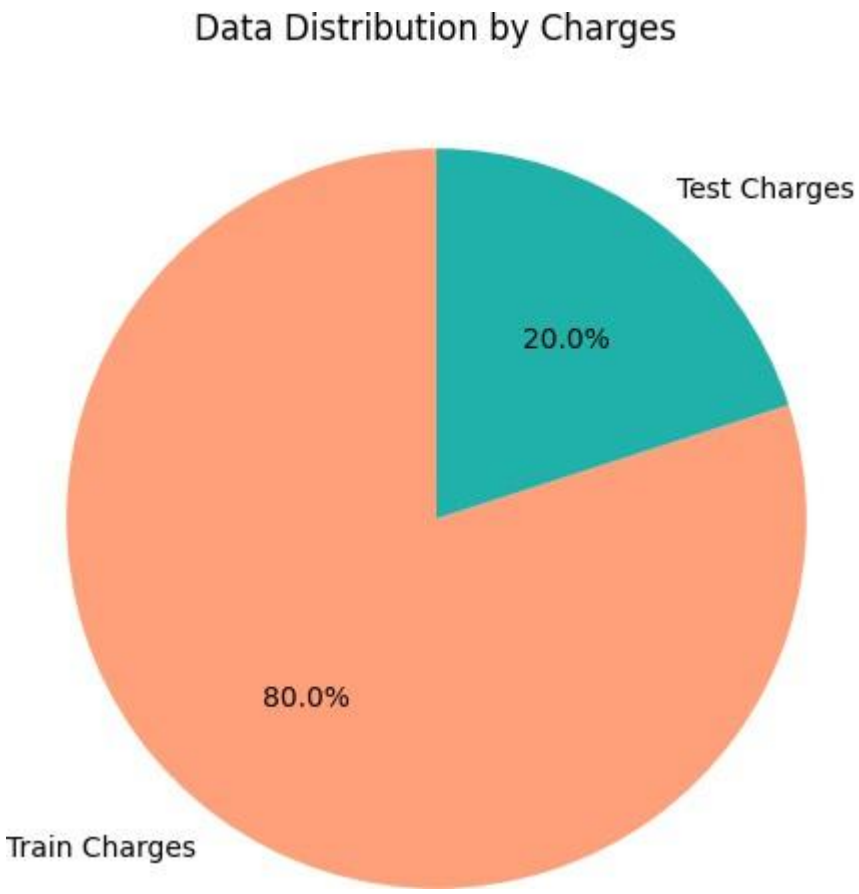

Fig 1.0

- **Observation:** In this pie chart, 80% of data is used for training and 20% for testing.
- **Key Insight:** Ensures model generalization without overfitting**.**

CORRELATION HEATMAP

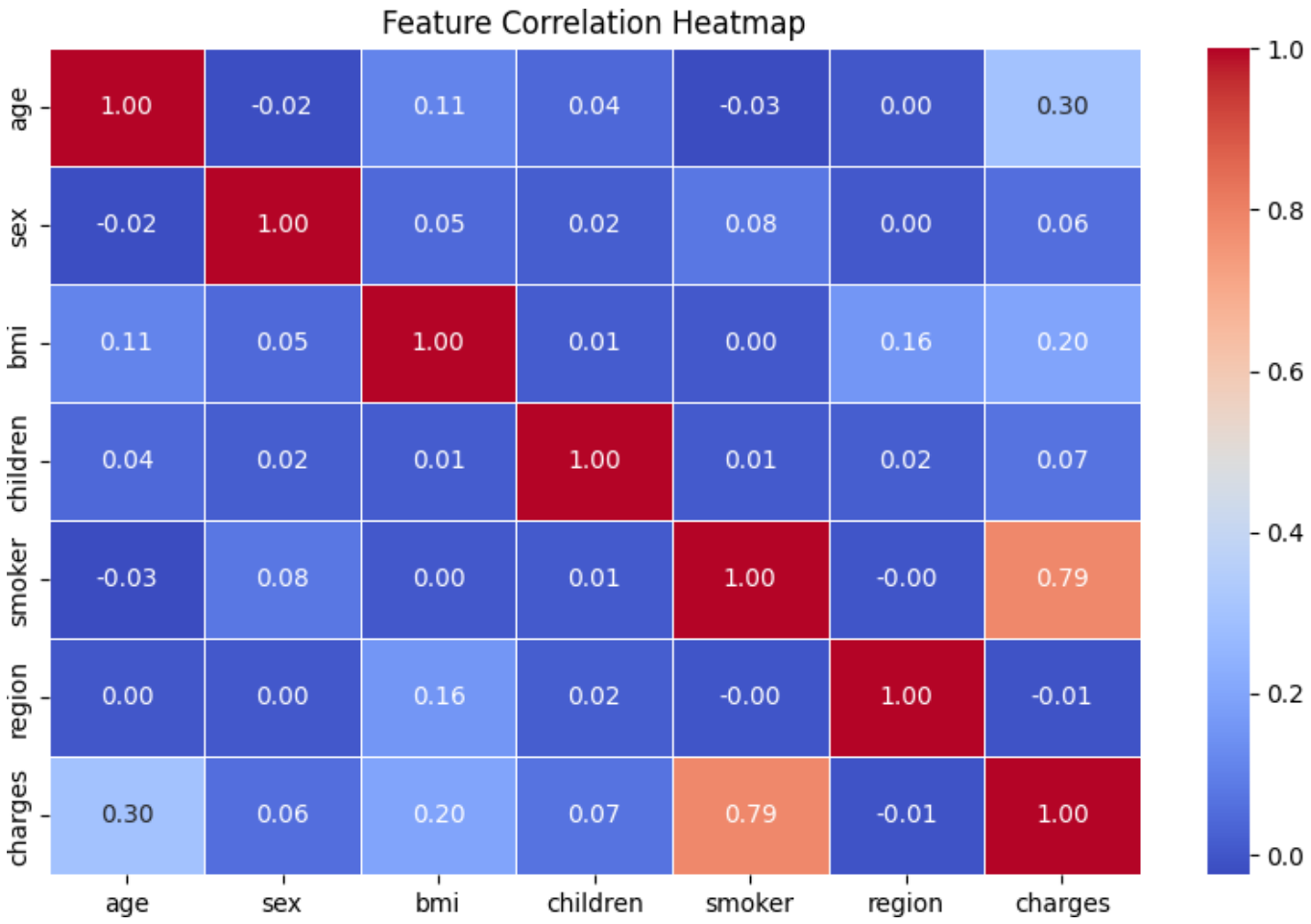

Fig 1.1

- **Observation:**

1. Smoker has the highest correlation with charges (0.79), meaning it is a strong predictor.
2. Age (0.30) and BMI (0.20) contribute but with lower influence.
3. Region has almost no correlation, indicating it is less relevant.

CHAPTER 5

RESULT

After training the Radial Basis Function (RBF) Regressor on the insurance dataset, the model demonstrated strong performance. The Mean Squared Error (MSE) was 0.1301 for training and 0.1506 for testing, while the R^2 Score was 0.8679 and 0.8578, respectively. Our model effectively captures the variance in the data and generalizes well to unseen samples, with minimal overfitting.

Display first 10 values:

Sl.No	Actual value	Predicted value	Error	Squared Error
1	0.481511	0.471904	0.009607	0.000092
2	0.770148	0.755162	0.014986	0.000225
3	0.660713	0.716750	0.056037	0.003140
4	0.286608	0.204671	0.081937	0.006714
5	0.921325	0.840793	0.080532	0.006485
6	0.344914	0.289721	0.055193	0.003046
7	0.992453	0.966737	0.025716	0.000661
8	1.326718	1.411583	0.084865	0.007202
9	2.716182	2.810113	0.093930	0.008823
10	2.924553	3.011421	0.086868	0.007546

Output:

Training Mean Squared Error: 0.13014008922662346

Testing Mean Squared Error: 0.15061467326607114

Training R2 Score: 0.8678680098818563

Testing R2 Score: 0.8578314252049046

SCATTER PLOT (ACTUAL VS PREDICTED CHARGES)

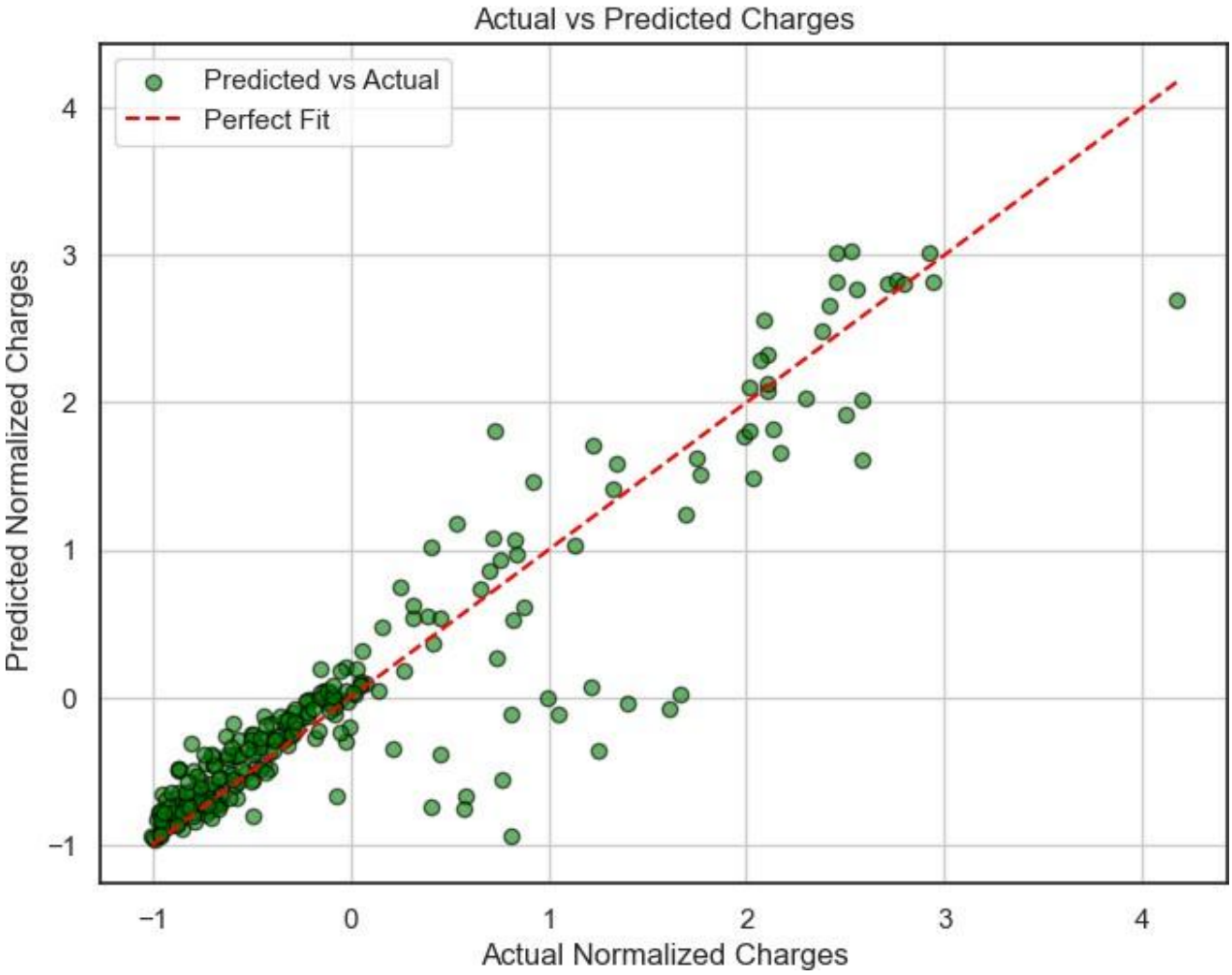

Fig 1.2

- **Observation:** Most points align closely with the red reference line, indicating a good fit.
- **Key Insight:** The model generalizes well, with minimal error in predictions.

LINE PLOT - SUBSET (ACTUAL VS PREDICTED CHARGES)

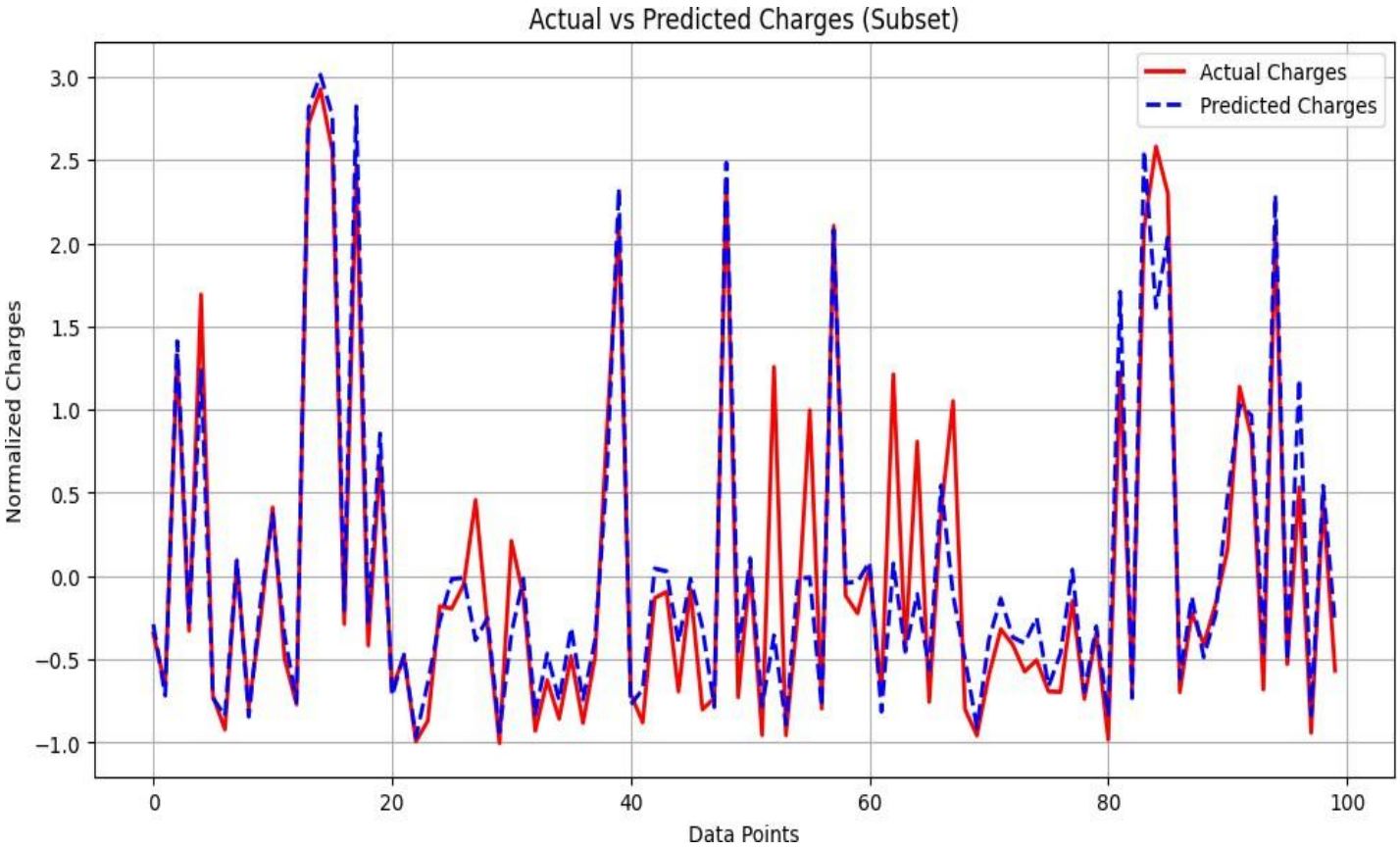

Fig 1.3

- **Observation:** The red line (actual charges) closely follows the blue dashed line (predicted charges).
- **Key Insight:** The model captures trends effectively, with some minor deviations.

CHAPTER 6

CONCLUSION

In this study, we implemented a Radial Basis Function (RBF) Regressor to predict medical insurance charges using key demographic and health-related factors. The model demonstrated strong predictive performance, with a Training R^2 score of 0.8679 and a Testing R^2 score of 0.8578, indicating its ability to generalize well to unseen data.

Our results highlight the effectiveness of RBF networks in capturing complex relationships in the dataset, particularly the strong correlation between smoking status and insurance costs. The low Mean Squared Error (MSE) further confirms the model's precision in estimating charges

Overall, the study underscores the potential of RBF-based models in insurance risk assessment and pricing strategies.

CHAPTER 7

REFERENCES

1. Claesen, M., De Smet, F., Suykens, J.A. and De Moor, B., 2014. Fast prediction with SVM models containing RBF kernels. arXiv preprint arXiv:1403.0736.

2. Izonin, I., Tkachenko, R., Kryvinska, N., Gregus, M., Tkachenko, P. and Vitynskyi, P., 2019, July. Committee of SGTM neural-like structures with RBF kernel for insurance cost prediction task. In 2019 IEEE 2nd Ukraine Conference on Electrical and Computer Engineering (UKRCON) (pp. 1037-1040). IEEE.

3. Veerappermal Devarajan, M., Lakshmi Gudivaka, R., Ramanjaneyulu Gudivaka, B., Kumar Gudivaka, R., Ganesan, T. and Malarvizhi Kumar, P., Computer Vision-Based Radial Basis Function Networks in Healthcare Applications. *Rajya and Ramanjaneyulu Gudivaka, Basava and Kumar Gudivaka, Raj and Ganesan, Thirusubramanian and Malarvizhi Kumar, Priyan, Computer Vision-Based Radial Basis Function Networks in Healthcare Applications*.