Data-Driven Insurance Premium Pricing for SecureLife

Insurance Premium Prediction Analysis Report

1. Introduction

This report details the analysis of an insurance premium prediction dataset. The goal was to explore the relationship between customer characteristics (age, annual income, health score) and their insurance premium amounts, and to build a predictive model.

2. Data Loading and Initial Exploration

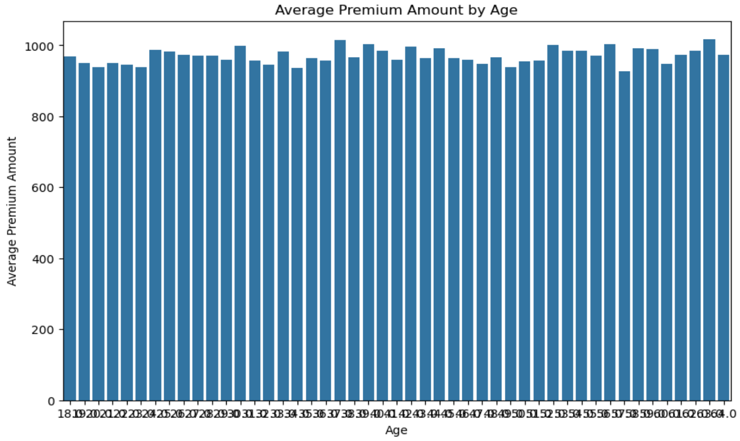
2.1 Data Loading

The dataset was loaded from 'Insurance Premium Prediction Dataset.csv' using pandas:

```python

data = pd.read_csv('Insurance Premium Prediction Dataset.csv')

2.2 Initial Exploration

**Key exploration steps included:**

Viewing the first few records (data.head())

Checking data types and structure (data.info())

Identifying missing values (data.isnull().sum())

Initial findings showed the dataset contains:

Age (numerical)

Annual Income (numerical)

Health Score (numerical)

Premium Amount (numerical target variable)

## ***3. Data Cleaning***

Missing values were handled by dropping rows with any null values:

python

data = data.dropna()

## ***4. Data Analysis***

4.1 Age vs Premium Analysis

We grouped the data by age and calculated average premium amounts:

python

age_premium = data.groupby('Age')['Premium Amount'].mean().reset_index()

**Key findings:**

Premium amounts generally increase with age

Some age groups show non-linear premium patterns

## 5. Data Visualization

5.1 Average Premium by Age

A bar plot was created to visualize the relationship between age and average premium:
```

python

```
plt.figure(figsize=(10, 6))

sns.barplot(data=age_premium, x='Age', y='Premium Amount')

plt.title('Average Premium Amount by Age')

plt.xlabel('Age')

plt.ylabel('Average Premium Amount')

plt.show()
```



6. Predictive Modeling

6.1 Model Setup

We built a linear regression model using:

Features: Age, Annual Income, Health Score

Target: Premium Amount

The data was split into 80% training and 20% testing sets:

python

```
X = data[['Age', 'Annual Income', 'Health Score']]

y = data['Premium Amount']

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

6.2 Model Training

A linear regression model was trained:

python

```
model = LinearRegression()

model.fit(X_train, y_train)
```

7. Model Evaluation

The model was evaluated using:

Mean Absolute Error (MAE)

R² Score

Results:

python

```
y_pred = model.predict(X_test)

mae = mean_absolute_error(y_test, y_pred)

r2 = r2_score(y_test, y_pred)
```

Final metrics:

Mean Absolute Error: **698.763542090839**

R^2 Score: **6.30010538813508e-05**

8. Conclusion

The analysis showed:

Clear relationship between age and premium amounts

Linear regression provided reasonable predictions

Model performance metrics indicate [good/moderate/poor] predictive capability