Insurance Claim Prediction- Linear Regression

Context

A key challenge for the insurance industry is to charge each customer an appropriate premium for the risk they represent. The ability to predict a correct claim amount has a significant impact on insurer's management decisions and financial statements. Predicting the cost of claims in an insurance company is a real-life problem that needs to be solved in a more accurate and automated way. Several factors determine the cost of claims based on health factors like BMI, age, smoker, health conditions and others. Insurance companies apply numerous techniques for analyzing and predicting health insurance costs.

Attribute information:

age: Age of the policyholder
sex: Gender of policyholder
bmi: Body mass index, providing an understanding of body, weights that are relatively high or low relative to height, objective index of body weight (kg / m ^ 2) using the ratio of height to weight
children: Number of children of the policyholder
smoker: Indicates policy holder is a smoker or a non-smoker (non-smoker=0; smoker=1)
region: The region where the policy holder belongs to (northeast, northwest, southeast, southwest)
claim: Claim amount
bloodpressure: Blood pressure reading of policyholder
diabetes: Suffers from diabetes or not (non-diabetic=0; diabetic=1)
regular_ex: Regularly exercise or not (no-exercise=0; exercise=1)

Approach:

- We have data on policy holders across attributes like age, sex, bmi, number of children, weather or not the person smokes etc.
- There will be a few categorical variables that we will have to encode, we will also look at transforming variables that are heavily skewed.
- Using this data, we will attempt to train a Regression model that can make predictions on unseen data.
- We will see different ways of evaluating a regression model.

Steps to be taken:

1. Import the data and perform the following checks and **write down your insights** at every step
    a. Shape of the data
    b. Data types of attributes
    c. 5-point summary of the relevant attributes
    d. Missing values

e. Correlation among the attributes
f. Outliers (display a boxplot)
g. Remove outliers (using IQR)
h. Distribution of the target column("claim")

2. Transform the column "claim" using log transformation (hint: use np.log('column') and append the transformed column to the dataframe under the column name "log_claim" - optionally you can check the effect of the transformation by plotting histogram of "claim" before and after transformation

3. Encode the categorical variables. In case a column has more than 2 categories, use one-hot encoding
4. Separate out the dependant variable("claim") from the independent variables(exclude claim and log_claim from the rest of the variables)
5. Split the data into testing and training sets (X_train, y_train, X_test, y_test)
6. Train a linear regression model using the training data and print the r_squared value of the prediction on the test data.
7. Plot a scatter plot between the actual values and the predicted values for the test set (because plain numbers might not give the entire picture)
8. Comment on the performance of the model
9. Repeat steps 4, 5, 6,7 and 8 except, this time use "log_claim" as your dependant variable (note: "claim" cannot be among the predictors)
10. Compare the performance of the models trained using the skewed dependant variable as it is and log transformed variable - write your comments and conclude the project

Additional Remarks:

Apart from the above steps, additionally you can try improving the performance of the model by using label encoding instead of one hot encoding or dropping few variables that do not contribute much to the model, using a different or more sophisticated model altogether etc
In the future courses we will come across all different regression approaches that might help us make better predictions out of our data.

Learning Outcome:

Linear Regression
Exploratory Data Analysis
Descriptive Statistics
Log Transformations
Dealing with categorical data