Arondeep Khakh

Insurance Predict

# Introduction –

I used the random forest classifier in order to predict whether a person will make a health insurance claim or not based upon varying different factors. There are multiple different ways to do this, but I chose the random forest classifier as it is the most flexible and easy to use algorithm.

# Aims and objectives –

I recently passed my driving test and realised that the price of car insurance was outrageous. This being the case I did wish to find a dataset that provided me with the factors that determine car insurance itself but could not find a good dataset. After wasting too much time looking for the dataset my keyword search brought me to a dataset on health insurance claims, I decided to settle here as the dataset was very simple and easy for me to understand as I have struggled with this project. I am also an ex-smoker, and this was another factor within the dataset that piqued my interest.

When looking at the dataset it was also clear what the information would be useful for, there was really only one option from the dataset and that was to make the ‘insuranceclaim’ factor the target variable.

I also had to produce some questions for my model to answer, I will list them below.

Question 1) Which pieces of data have the most influence upon whether a person makes an insurance claim or not.

Question 2) Determine which factors from my dataset have a low impact upon the target variable.

Question 3) – Does smoking have an effect upon whether a person would make an insurance claim?

Question 4) Does BMI have a strong effect upon whether a person would make an insurance claim?

Question 5) Is there a correlation between age and BMI?

# Dataset analysis

The dataset I have chosen has multiple different features within. I will explain each feature here.

Age – Represents the age of the person in years.

Sex – Represents the gender or ‘sex’ of the person, 1 being male and 0 being female.

Bmi – Represents the body mass index of the individual, higher the body mass index the more unhealthy a weight the person has, unless ofcourse the body mass index is extremely low, this can also represent an underweight and unhealthy person.

Children – Represents the number of children the person in question has.

Smoker – Represents whether the person in question is a smoker, represented by 1 meaning the person is a smoker and 0 meaning the person is not a smoker.

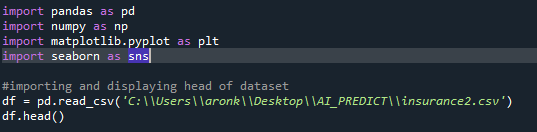
Region – Represents the region of the person in question, this dataset is taken from the usa. 0 Represents a person residing in the northeast, 1 represents the northwest, 2 represents the southeast and 3 represents the south-west.

Charges – These are the individual medical costs that are billed by the health insurance companies.

Insuranceclaim – represents whether that person has made an insurance claim, 1 representing that they have and 0 representing that they have not.

# Machine Learning Model

I started by importing the relevant libraries. These libraries would allow me to manipulate the dataset in the manner I wish along with plotting the graphs I wished to plot. Of course, I also had to import the relevant data set from my device. I will provide example code below.



## Data pre-processing

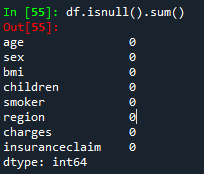
When you prepare data, you have the opportunity to ensure that your model is more reliable and accurate, an example that I have used is checking for null values within the dataset. Of course, Null values would make the code more unreliable. However, this is not always required as if you are to used a pre cleaned dataset the implementation of checking null values would be irrelevant and inefficient as the null values will have already been removed, this is down to yourself to determine whether is the case or not, you would refer back to the data analysis part of the assignment.

I will provide the code used within my work to check for null values below.

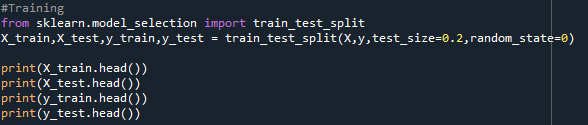


Another thing to consider when pre-processing your data is whether you want two factors from your data set that correlate strongly to be a part of your model. Using two factors that correlate strongly can cause overfitting within my model and effect the accuracy of my prediction. Luckily, within my data set this was not an issue as there are no factors that correlate strongly.

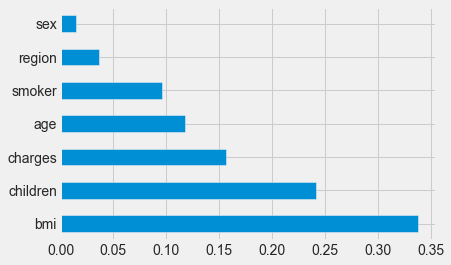
As you can see from the screenshot I will provide below, the output confirms that the data is clean and there is no null variables within the dataset.



I split my data into both a training set and a test set, training is used in order for the model to learn more information about the data while the test set is used to analyse the accuracy of our model.

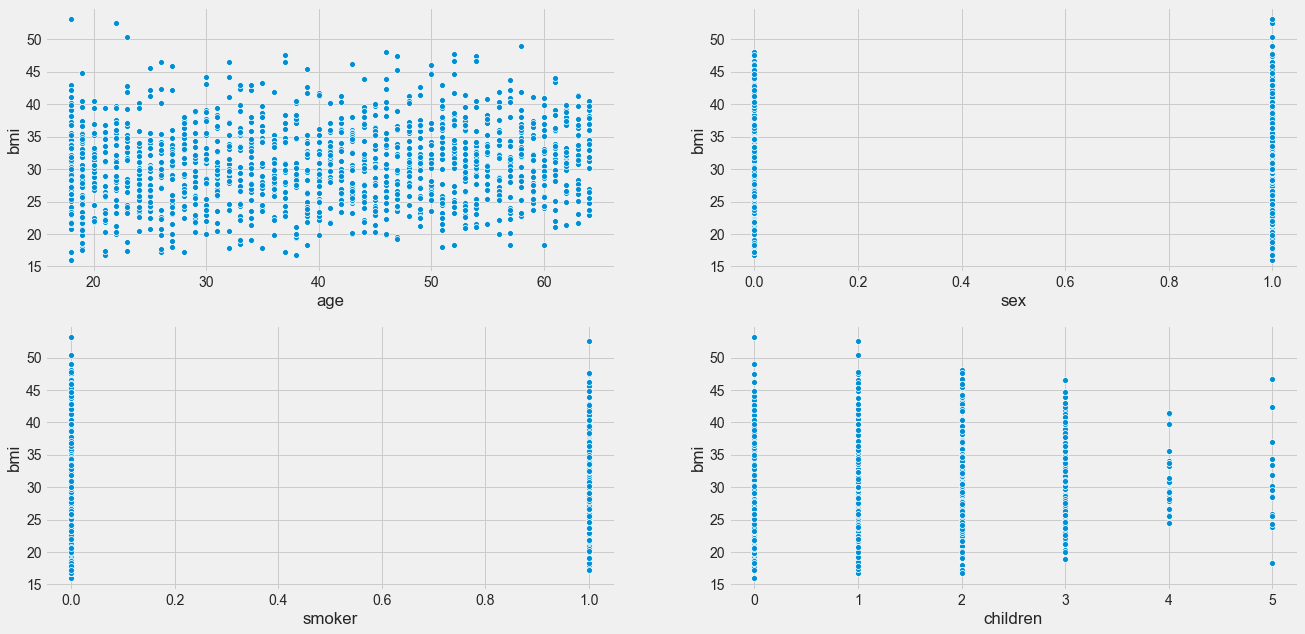


I have also implemented feature selection within my project, feature selection is used in order to determine which factors from your data have the greatest impact upon the target variable, in this case the target variable being whether an insurance claim is made or not. From our example we find that the least relevant factor for determining whether an insurance claim was made or not was sex, this data could be removed in future as it had such a small bearing upon the target variable. On the other hand, BMI was the most important and relevant factor upon our target variable, a graph was plotted to show this. This relates back to my question 1 and 2. Question 3 also asks whether somebody being a smoker will have any bearing on whether they would make an insurance claim, we can see the answer is yes.

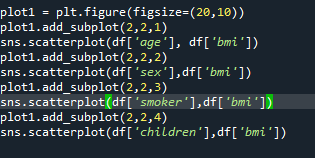


## Data Visualization

Plotting charts is helpful as it allows you to identify any outliers within the dataset, again this is not required but it is always useful to understand the data you are using in a more detailed way. I plotted a graph in order to understand the relationship between certain factors and a person’s BMI. An interesting conclusion that I had made from this is that as a person gets to the upper limit of age their average BMI is also more stable within what would be considered the ‘healthy region’.



I will also provide the code required for plotting this graph below.

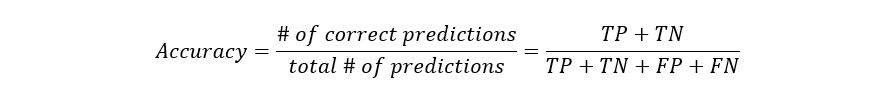


## Evaluating my model –

In order to evaluate my model I decided to implement a confusion matrix, I decided to use a confusion matrix as it is able to assess key factors which are relevant to my model, for example, precision, accuracy etc. I will go on to explain all of the different metrics further below.

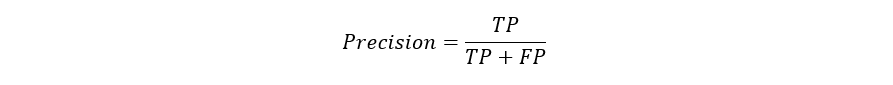
Accuracy –

Equal to the proportion of predictions that the model correctly classified.



Precision –

Answers ‘what proportion of positive identifications were correct’?



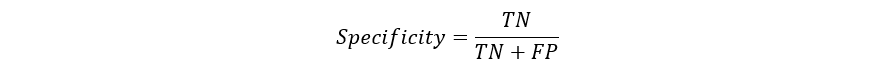
Recall –

Also called ‘hit rate’, ‘sensitivity’ or ‘true positive rate’, it is the proportion of the total amount of relevant instances that were retrieved. Answering the question ‘Of the positives, what proportion of them were identified correctly?

## Image for post

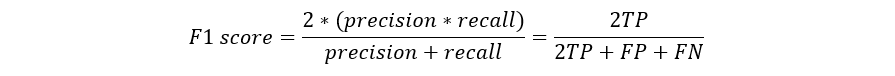
Specificity –

Measures the proportion of actual negative that were correctly identified, in other words it is the opposite of recall.

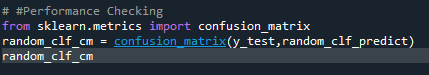


F1 Score –

The mean of precision and recall, maximum score of 1 and a minimum of 0. It is a measure of the preciseness of your model.



I will also provide the code for my confusion matrix below.

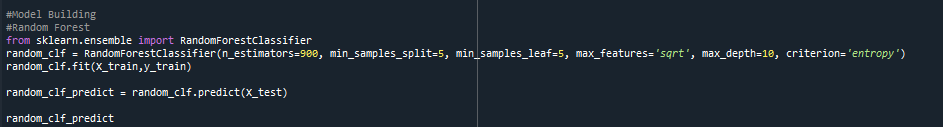


## RandomForestClassifier Algorithm –

RandomForestClassifier can be used both for classification and regression. It is a flexible and easy to use algorithm comprised of trees, the more trees it has, the more robust the model. Random forests create decision trees upon randomly selected data (Factors from dataset) gathers a prediction from each tree and the selects the best solution by means of voting. It is a strong indicator of the feature importance, which was relevant to the questions I defined for my assignment. The randomforest also reduces the problem of overfitting (although not a huge problem for my dataset).

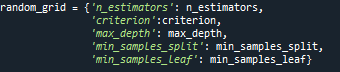
It does this by taking the average of all the predictions. It does have it’s downsides though, it is slow to execute due to the amount of multiple decision trees. It is also difficult to interpret.

I will provide the code for my randomforestclassifier implementation below.



## RandomizedSearchCV

I also attempted to implement the randomizedsearchCV function, the first step was to write the parameters that I wished to consider and from those parameters select the best ones. I will provide example from my code below.



I then create my RandomizedSearchCV object and fit the data. I then find the best parameters and scores. I will provide example from my code below.

