10 July 2022

Capstone Project – HealthCare

PGP-DSBA June-Batch

SOOREJ KUNNOOL BALAN

2022

Table of Contents

[1. Introduction 3](#_Toc108310431)

[2. EDA & Business Implication 4](#_Toc108310432)

[Univariate Analysis 6](#_Toc108310433)

[Correlation Heatmap 14](#_Toc108310434)

[Multivariate Analysis 15](#_Toc108310435)

[Business Insights 19](#_Toc108310436)

[3. Data Cleaning and Pre-processing 19](#_Toc108310437)

[Column Deletion 19](#_Toc108310438)

[Column Renaming 19](#_Toc108310439)

[Duplicate Rows 20](#_Toc108310440)

[Null value Treatment 20](#_Toc108310441)

[Outlier Treatment 20](#_Toc108310442)

[Data grouping via Clusters 20](#_Toc108310443)

[Variable transformation 21](#_Toc108310444)

[Standardization 21](#_Toc108310445)

[Dummy encoding 21](#_Toc108310446)

[4. Model building 21](#_Toc108310447)

[Train-test split 21](#_Toc108310448)

[Strategy 22](#_Toc108310449)

[Linear Models 23](#_Toc108310450)

[Non-Linear Models 25](#_Toc108310451)

[Ensemble Models 27](#_Toc108310452)

[5. Model Validation 29](#_Toc108310453)

[Linear Models 29](#_Toc108310454)

[Non-Linear Models 30](#_Toc108310455)

[Ensemble Models 32](#_Toc108310456)

[Model Performance Comparison 33](#_Toc108310457)

[6. Final Interpretation/recommendation 34](#_Toc108310458)

[END 34](#_Toc108310459)

List of Tables List of Figures

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Table-2.1 Data- Concise Summary | 4 |  | Figure-2.1 Histogram/Boxplot: relationship\_years | 6 |
| Table-2.2 Central Measures of Tendency(continuous variables) | 6 |  | Figure-2.2 Histogram/Boxplot: checkup\_count | 7 |
| Table-2.3 Descriptive Stats – Discrete variable | 11 |  | Figure-2.3 Histogram/Boxplot: adventure\_sports | 7 |
| Table-3.1 K Means cluster means | 20 |  | Figure-2.4 Histogram/Boxplot: consultation\_count | 7 |
| Table-4.1 Train and test target cluster variables distribution | 21 |  | Figure-2.5 Histogram/Boxplot: avg\_steps | 8 |
| Table-4.2 OLS Regression Summary | 23 |  | Figure-2.6 Histogram/Boxplot: age | 8 |
| Table-4.3 OLS Regression Summary-Final | 24 |  | Figure-2.7 Histogram/Boxplot: heart\_incident | 8 |
| Table-4.4 Lasso: Predictors with Coefficient >0 | 24 |  | Figure-2.8 Histogram/Boxplot: dcs\_incident | 9 |
| Table-4.5 Ridge: Predictor coefficient | 25 |  | Figure-2.9 Histogram/Boxplot: glucose\_level | 9 |
| Table-4.6 DecisionTree: Feature importance | 26 |  | Figure-2.10 Histogram/Boxplot: bmi | 9 |
| Table-4.7 ANN: Feature importance | 27 |  | Figure-2.11 Histogram/Boxplot: last\_admitted | 10 |
| Table-5.1 Linear Regression: Performance Metrics | 29 |  | Figure-2.12 Histogram/Boxplot: weight | 10 |
| Table-5.2 Lasso Regression: Performance Metrics | 30 |  | Figure-2.13 Histogram/Boxplot: weight\_change | 10 |
| Table-5.3 Ridge Regression: Performance Metrics | 30 |  | Figure-2.14 Histogram/Boxplot: fat\_percentage | 11 |
| Table-5.4 DecisionTree Regression: Performance Metrics | 30 |  | Figure-2.15 Histogram/Boxplot: insurance\_cost | 11 |
| Table-5.5 ANN Regression: Performance Metrics | 31 |  | Figure-2.16 Countplot/Pie chart: Occupation | 12 |
| Table-5.6 KNN Regression: Performance Metrics | 31 |  | Figure-2.17 Countplot/Pie chart: cholesterol\_level | 12 |
| Table-5.7 Random Forest – Performance metrics | 32 |  | Figure-2.18 Countplot/Pie chart: Gender | 12 |
| Table-5.8 Bagging – Performance metrics | 32 |  | Figure-2.19 Countplot/Pie chart: smoking\_status | 13 |
| Table-5.9 AdaBoost – Performance metrics | 32 |  | Figure-2.20 Countplot/Pie chart: Location | 13 |
| Table-5.10 Gradient Boost – Performance metrics | 33 |  | Figure-2.21 Countplot/Pie chart: other\_insurers | 13 |
| Table-5.11 Gradient Boost – Performance metrics | 33 |  | Figure-2.22 Countplot/Pie chart: Alcohol | 14 |
| Table-5.12 Gradient Boost – Performance metrics | 33 |  | Figure-2.23 Countplot/Pie chart: exercise | 14 |
|  |  |  | Figure-2.24 Correlation Heatmap | 15 |
|  |  |  | Figure-2.25 Barplot: insurance\_cost vs Alcohol | 15 |
|  |  |  | Figure-2.26 Barplot: insurance\_cost vs Occupation | 16 |
|  |  |  | Figure-2.27 Barplot: insurance\_cost vs cholesterol | 16 |
|  |  |  | Figure-2.28 Barplot: insurance\_cost vs Gender | 16 |
|  |  |  | Figure-2.29 Barplot: insurance\_cost vs smoking\_status | 17 |
|  |  |  | Figure-2.30 Barplot: insurance\_cost vs exercise\_status | 17 |
|  |  |  | Figure-2.31 Barplot: insurance\_cost vs Location | 17 |
|  |  |  | Figure-2.32 Barplot: insurance\_cost vs other\_insurers | 18 |
|  |  |  | Figure-2.33 Scatterplot: insurance\_cost vs weight | 18 |
|  |  |  | Figure-2.34 Scatterplot: insurance\_cost vs last\_admitted | 18 |
|  |  |  | Figure-2.35 Scatterplot: insurance\_cost vs bmi | 19 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Introduction

The objective of this problem is to build a model, using provided data, to arrive at predicting optimal insurance cost for an individual. This is important for the insurance provider to be competitive in the market among other insurance providers by providing insurance to customers at competitive premium costs as well as ensure that they do not take undue risk by reducing the cost too much that the resulting insurance claims leads to monetary loss. Data provided is given below:



If we look at the Indian insurance landscape, the health insurance premium collected from customers increased from Rs.1910 crores in 2006–2007 to Rs. 33011 crores in 2018–2019. But claims incurred together with commission and management expenses have grown from Rs. 3349 crores to Rs. 40076 crores during the same period. So, the claims and management expenses incurred together is more than the health insurance premium earned in all the years of our study thereby leading to underwriting loss. According to the Economic Times, inflation in healthcare is growing at a rate of 12 to 18%, this includes overall costs such as cost of medicines, hospital admission charges, cost of various treatments, medical advancements and so on. So overall we can see that with every unit of increase in premium income the claims incurred together with commission and management expenses paid increased more than a unit. So instead of earning profit due to better business through higher premium income, it has incurred losses.

Health care is a critical domain for every individual as it directly links to wellbeing of an individual and by extension his/her family. Money plays a key role in this domain because certain treatments are expensive, and insurance helps to cover the risk. If any individual is not covered under the insurance, then it will become a tough financial situation for that individual. A health insurance risk pool is a group of individuals whose medical costs are combined to calculate premiums. Pooling risks together allows the higher costs of the less healthy to be offset by the lower costs of the healthy. A risk pool with healthy individuals can result in lower-than-average insurance costs, a large pool with a large share of unhealthy individuals can have higher-than-average costs. Hence the ability of the insurance company to assess the health/risk of its customer and correlate it to an optimal insurance cost is paramount.

**Objectives:**

Build a predictive regression model, which can predict the optimum insurance cost for an individual with highest accuracy and in a cost-effective manner.

Consider all the features provided, such as health and habit related parameters, towards arriving at the estimated cost of insurance.

Gain insights on the features using exploratory data analysis.

Treat the data in case of missing values, duplicates, outliers etc.

Provide recommendations to business based on the provided data.

**Scope:**

The scope of this business problem is limited to the data provided by the business.

No other external factors will be considered in the data analysis or model building phase.

The model building exercise will consider linear, non-linear and ensemble models to arrive at the best model based on performance factors such as Adjusted R-square and RMSE.

The exercise will evaluate the data over 10 regression models.

**Constraints:**

The main constraints faced by insurance companies are economic uncertainty, regulatory changes, risk of contagion etc.

Any of these could impact the insurance company’s ability to hedge the treatment cost of unhealthy individuals over perceived healthy individuals.

The model will consider existing data patterns provided and arrive at a prediction model. If the circumstances change and if new factors/features of importance emerge, the model’s efficacy will be severely impacted.

# EDA & Business Implication

Let us look at the concise summary info.

A picture containing table

Description automatically generated

1. Table-2.1 Data- Concise Summary

The data has 25000 rows and 24 columns.

The 24 columns comprise of 8 object columns, 2 float and 14 integer columns.

We can see 2 columns have missing values – ‘bmi’ and ‘Year\_last\_admitted’.

Let us look at what each column signifies:

* applicant\_id (integer data type) – This is the company’s unique id provided to each customer. This is a nominal variable and will be from the model building exercise.
* years\_of\_insurance\_with\_us (integer data type) – Since how many years customer is associated with the company. This field can be renamed to ‘relationship\_years’.
* regular\_checkup\_lasy\_year (integer data type) – Number of times customer has done regular health checkup in last one year. Since we do not have data of other checkup history, the field can be renamed to checkup\_count
* adventure\_sports(integer data type) – If customer is involved with adventure sports like climbing, diving etc. column will have value as ‘1’ else ‘0’. This needs to be treated as a discrete column.
* Occupation(object data type) – Occupation of the customer(Values are ‘Business’, ‘Salried’ & ‘Student’.
* visited\_doctor\_last\_1\_year(integer data type) – Number of times customer has visited doctor in last one year. Since we do not have data of doctor consultations for other years, we can rename the field to ‘consultation\_count’.
* cholesterol\_level(object data type) – Cholesterol level of the customers while applying for insurance(Values are ‘125 to 150’, ‘150 to 175’, ‘175 to 200’, ‘200 to 225’, ‘225 to 250’).
* daily\_avg\_steps(integer data type) – Average daily steps walked by customers. Since only daily information is available, lets rename this to avg\_steps.
* Age(integer data) – Age of the customer
* heart\_decs\_history(integer data type) – Any past heart disease in the past. This can be renamed heart\_incident. This looks to be a discrete column with ‘0’ showing no heart incidents and ‘1’ showing presence of past heart incidents.
* other\_major\_decs\_history(integer data type) – Any past major diseases, apart from heart, like any operation. This can be renamed to decs\_incident. This looks to be a discrete column with ‘0’ showing no major disease incidents and ‘1’ showing presence of major disease incidents.
* Gender(object data type) – Gender of the customer.
* avg\_glucose\_level(integer data type) - Average glucose level of the customer while applying the insurance. Field can be renamed to glucose\_level.
* bmi(float data type) - BMI of the customer while applying the insurance
* smoking\_status(object data type) – Smoking status of the customer
* Year\_last\_admitted(float data type) - When customer have been admitted in the hospital last time. Can be renamed to ‘last\_admitted’.
* Location(object data type) – Location of the hospital
* weight(integer data type) – Weight of the customer
* covered\_by\_any\_other\_company(object data type) – Whether customer is covered by any other insurance company(Values are ‘Y’, ‘N’). Field can be renamed to ‘other\_insurers’.
* Alcohol(object data type) – Alcohol consumption status of the customer (Values are ‘Daily’, ‘No’, ‘Rare’)
* exercise(object data type) – Regular exercise status of the customer.
* weight\_change\_in\_last\_one\_year(integer data type) - How much variation has been seen in the weight of the customer in last year. Field can be renamed to weight\_change.
* fat\_percentage(integer data type) - Fat percentage of the customer while applying the insurance
* insurance\_cost(integer data type) – Total Insurance Cost

## Univariate Analysis

Let’s look at the central measure of tendency, inter quartile values, histogram and boxplot of continuous variables.

Table

Description automatically generated with medium confidence

Table-2.2 Central Measures of Tendency(continuous variables)

The above table shows the mean, median(column name 50%), min, max, Range, Inter Quartile value (IQR) and Quartile min & Quartile max.

Let us look at boxplot/histogram of the continuous variables.

* **relationship\_years:**

Chart, histogram

Description automatically generatedFigure-2.1 depicts the histogram and boxplot of “relationship\_years” which shows negative skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are no outliers present in the field.

Figure-2.1 Histogram/Boxplot: relationship\_years

* **checkup\_count:**Chart, box and whisker chart

  Description automatically generated

Figure-2.2 depicts the histogram and boxplot of “checkup\_count” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are outliers present in the field.

Figure-2.2 Histogram/Boxplot: checkup\_count

* **adventure\_sports:**

Graphical user interface, application, Teams

Description automatically generatedFigure-2.3 depicts the histogram and boxplot of “adventure\_sports” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are outliers present in the field. Although the value in this field is numeric, this needs to be treated as a discrete variable. Figure-2.3 Histogram/Boxplot: adventure\_sports

We will be transforming this variable in section 3f.

Graphical user interface, chart

Description automatically generated

* **consultation\_count:**

Figure-2.4 depicts the histogram and boxplot of “consultation\_count” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are outliers present in the field. Figure-2.4 Histogram/Boxplot: consultation\_count

* Chart, histogram

  Description automatically generated**avg\_steps:**

Figure-2.5 depicts the histogram and boxplot of “avg\_steps” which shows positive skewness in the data. The histogram shows a normal distribution.

From the boxplot we can see that there are outliers present in the field.

Figure-2.5 Histogram/Boxplot: avg\_steps

* Chart, box and whisker chart

  Description automatically generated**age:**

Figure-2.6 depicts the histogram and boxplot of “age” which shows slight positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are no outliers present in the field.

Figure-2.6 Histogram/Boxplot: age

* Graphical user interface, application, Teams

  Description automatically generated**heart\_incident:**

Figure-2.7 depicts the histogram and boxplot of “heart\_incident” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are outliers present in the field. Although the value in this field is numeric, this needs to be treated as a Figure-2.7 Histogram/Boxplot: heart\_incident

discrete variable.

* Graphical user interface, application

  Description automatically generated**decs\_incident:**

Figure-2.8 depicts the histogram and boxplot of “decs\_incident” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are outliers present in the field. Although the value in Figure-2.8 Histogram/Boxplot: decs\_incident

this field is numeric, this needs to be treated as a discrete variable.

* Chart, box and whisker chart

  Description automatically generated**glucose\_level:**

Figure-2.9 depicts the histogram and boxplot of “glucose\_level” which shows slight positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram.

From the boxplot we can see that there are no outliers present in the field. Figure-2.9 Histogram/Boxplot: glucose\_level

* Chart

  Description automatically generated**bmi:**

Figure-2.10 depicts the histogram and boxplot of “bmi” which shows positive skewness in the data. The histogram shows a normal distribution.

From the boxplot we can see that there are outliers present in the field.

Figure-2.10 Histogram/Boxplot: bmi

* Chart, histogram

  Description automatically generated**last\_admitted:**

Figure-2.11 depicts the histogram and boxplot of “last\_admitted” which shows slight positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are no outliers present in the field. Figure-2.11 Histogram/Boxplot: last\_admitted

* **Chart, histogram

  Description automatically generatedweight:**

Figure-2.12 depicts the histogram and boxplot of “weight” which shows slight positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are no outliers present in the field. Figure-2.12 Histogram/Boxplot: weight

* Chart, box and whisker chart

  Description automatically generated**weight\_change:**

Figure-2.13 depicts the histogram and boxplot of “weight\_change” which shows slight positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are no outliers present in the field. Figure-2.13 Histogram/Boxplot: weight\_change

* Chart, histogram

  Description automatically generated**fat\_percentage:**

Figure-2.14 depicts the histogram and boxplot of “fat\_percentage” which shows negative skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are no outliers present in the field. Figure-2.14 Histogram/Boxplot: fat\_percentage

* Chart, histogram

  Description automatically generated**insurance\_cost:**

Figure-2.15 depicts the histogram and boxplot of “insurance\_cost” which shows positive skewness in the data. The data depicts the presence of multi-modes as we can see multiple peaks in the histogram. From the boxplot we can see that there are no outliers present in the field. Figure-2.15 Histogram/Boxplot: insurance\_cost

Let us look at the descriptive statistics of object variables.

Table

Description automatically generated with medium confidence

Table-2.3 Descriptive Stats – Discrete variable

Let us look at the distribution of categorical variable in detail:

* **Chart, bar chart, pie chart

  Description automatically generatedOccupation:**

Unique : 3

Top : Student

Freq : 10169

Let us look at the distribution of the field:

Student 10169

Business 10020

Salried 4811 Figure-2.16 Countplot/Pie chart: Occupation

‘Student’ occupation is the highest at 40.68% and ‘Salried’ is lowest at 19.24%.

* Chart, pie chart

  Description automatically generated**cholesterol\_level:**

Unique : 5

Top : 150 to 175

Freq : 8763

Let us look at the distribution of the field:

150 to 175 8763

125 to 150 8339

200 to 225 2963

175 to 200 2881 Figure-2.17 Countplot/Pie chart: cholesterol\_level

225 to 250 2054

‘150 to 175’ cholesterol\_level is the highest at 35.05% and ‘225 to 250’ is lowest at 8.22%.

* Chart, bar chart, treemap chart

  Description automatically generated**Gender:**

Unique : 2

Top : Male

Freq : 16422

Let us look at the distribution of the field:

Male 16422

Female 8578

Figure-2.18 Countplot/Pie chart: Gender

‘Male’ gender is the highest at 65.69% and ‘Female’ is lowest at 34.31%.

* Chart, bar chart, pie chart

  Description automatically generated**smoking\_status:**

Unique : 4

Top : never smoked

Freq : 9249

Let us look at the distribution of the field:

never smoked 9249

Unknown 7555

formerly smoked 4329

smokes 3867 Figure-2.19 Countplot/Pie chart: smoking\_status

‘never smoked’ smoking\_status is the highest at 37% and ‘smokes’ is lowest at 15.47%.

* Chart

  Description automatically generated**Location:**

Unique : 15

Top : Bangalore

Freq : 1742

Let us look at the distribution of the field:

Bangalore 1742

Jaipur 1706

Bhubaneswar 1704

Mangalore 1697

Delhi 1680 Figure-2.20 Countplot/Pie chart: Location

Ahmedabad 1677

Guwahati 1672

Chennai 1669

Kanpur 1664

‘Bangalore’ location is the highest at 6.97% and ‘Surat’ is lowest at 6.36%.

Nagpur 1663

Mumbai 1658

Lucknow 1637

Pune 1622

Kolkata 1620

**Chart, treemap chart

Description automatically generated** Surat 1589

* **other\_insurers:**

Unique : 2

Top : N

Freq : 17418

Let us look at the distribution of the field:

N 17418

Y 7582 Figure-2.21 Countplot/Pie chart: other\_insurers

‘N’ other\_insurers is the highest at 69.67% and ‘Y’ is lowest at 30.33%.

Chart, bar chart, pie chart

Description automatically generated

* **Alcohol:**

Unique : 3

Top : rare

Freq : 13752

Let us look at the distribution of the field:

Rare 13752

No 8541

Daily 2707

Figure-2.22 Countplot/Pie chart: Alcohol

‘Rare’ alcohol status is the highest at 55.01% and ‘Daily’ is lowest at 10.83%.

* Chart, pie chart

  Description automatically generated**exercise:**

Unique : 3

Top : rare

Freq : 13752

Let us look at the distribution of the field:

Moderate 14638

Extreme 5248

No 5114

Figure-2.23 Countplot/Pie chart: exercise

‘Moderate’ exercise status is the highest at 58.55% and ‘No’ is lowest at 20.46%.

## Correlation Heatmap

From Figure 2.24 we can see that most of the variables show less to almost no correlation with each other.

* Highest correlation is between weight and insurance\_cost, which is a positive correlation.
* We can see that weight\_change and last\_admitted has medium strong positive correlation.
* Weight and last\_admitted has very strong negative correlation.
* Weight and weight\_change has negative medium correlation.
* Weight\_change has medium negative correlation with insurance\_cost.

Chart, waterfall chart

Description automatically generated

Figure-2.24 Correlation Heatmap

## Multivariate Analysis

Alcohol vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the alcohol status. Surprisingly, alcohol status seems to not have impacted insurance cost.

Chart, bar chart, treemap chart

Description automatically generated

Figure-2.25 Barplot: insurance\_cost vs Alcohol

Occupation vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the occupation status.

Chart, bar chart, treemap chart

Description automatically generated

Figure-2.26 Barplot: insurance\_cost vs Occupation

cholesterol\_level vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the various cholesterol levels.

Chart, bar chart

Description automatically generated

Figure-2.27 Barplot: insurance\_cost vs cholesterol

Gender vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the Genders.

Chart

Description automatically generated

Figure-2.28 Barplot: insurance\_cost vs Gender

smoking\_status vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the different smoking status.

Chart, bar chart

Description automatically generated

Figure-2.29 Barplot: insurance\_cost vs smoking\_status

exercise vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the different exercise status.

Chart, bar chart, treemap chart

Description automatically generated

Figure-2.30 Barplot: insurance\_cost vs exercise\_status

Location vs insurance\_cost

From the below barplot we can see that the average insurance cost is almost similar across the various Locations.

Chart, bar chart

Description automatically generated

Figure-2.31 Barplot: insurance\_cost vs Location

other\_insurers vs insurance\_cost

From the below barplot we can see that insurers who are insured with other companies pay higher premium that the ones which do not. This would indicate that customers have insurance with other companies as they are expecting higher medical costs.

Chart

Description automatically generated

Figure-2.32 Barplot: insurance\_cost vs other\_insurers

weight vs insurance\_cost

From the below scatterplot we can see that as the weight increases the cost of insurance increases linearly.

Chart

Description automatically generated

Figure-2.33 Scatterplot: insurance\_cost vs weight

last\_admitted vs insurance\_cost

From the below scatterplot we can see that the insurance\_cost decreases as the last\_admitted to hospital is recent.

Graphical user interface

Description automatically generated with medium confidence

Figure-2.34 Scatterplot: insurance\_cost vs last\_admitted

bmi vs insurance\_cost

From the below scatterplot we can see that the insurance\_cost does not seem to have any certain pattern with the bmi.

Chart, scatter chart

Description automatically generated

Figure-2.35 Scatterplot: insurance\_cost vs bmi

## Business Insights

Weight taken at time of enrollment is the most crucial factor in deciding the insurance cost. With the advent of digital technology, a mechanism to track weight and reduction in weight leading to brownie points could be a strategy for the business.

All other factors mostly thought to influence the insurance cost such as age, bmi, smoking etc. does not seem to significantly impact insurance cost.

We have seen that average insurance cost is higher for those who have insurance from other company, as this shows that the customer is trying to cover an impending health risk.

# Data Cleaning and Pre-processing

## Column Deletion

applicant\_id though numerical columns is nominal in nature and will not contribute to the model building process. Hence we can remove this column.

## Column Renaming

We will rename the below 10 columns, as shown below to make better sense:

'Years\_of\_insurance\_with\_us' : 'relationship\_years'

'regular\_checkup\_lasy\_year' : 'checkup\_count'

'visited\_doctor\_last\_1\_year' : 'consultation\_count'

'daily\_avg\_steps' : 'avg\_steps'

'heart\_decs\_history' : 'heart\_incident'

'other\_major\_decs\_history' : 'decs\_incident'

'avg\_glucose\_level' : 'glucose\_level'

'Year\_last\_admitted' : 'last\_admitted'

'covered\_by\_any\_other\_company' : 'other\_insurers'

'weight\_change\_in\_last\_one\_year' : 'weight\_change'

## Duplicate Rows

The dataset does not have any duplicate columns.

## Null value Treatment

There are 2 features with null values – ‘bmi’ and ‘last\_admitted’.

‘bmi’ feature has 990 rows out of 25,000 rows as nulls, we can remove these 990 observations from the dataset, which amounts to less than 4% of the total observations.

‘last\_admitted’ columns missing values seems to be valid data where the customers were never admitted to hospital, which is a realistic scenario.

## Outlier Treatment

We can see outliers mainly in features such as ‘bmi’, ‘avg\_steps’, ‘checkup\_count’ etc. These are realistic values and the model will need to factor in these outliers when predicting the target variable, which is insurance\_cost.

## Data grouping via Clusters

Let us run K Means clustering on the data and group the data into clusters, where we got 4 clusters with highest silhouette score.

Let us look at the means of variables across clusters and the number of records in each cluster.

Table

Description automatically generated

Table-3.1 K Means cluster means

Let us look at the mean of insurance cost across the cluster labels in ascending order.

Cluster 3 – 11631

Cluster 0 – 20573

Cluster 1 – 33086

Cluster 4 – 36904

We can classify the customers as low, medium, high and very high-risk customers, corresponding to increasing chances of their raising a claim.

Cluster 3 with mean insurance cost of 11631 can be considered as the low-risk customers.

Cluster 0 with mean insurance cost of 20573 can be considered as medium risk customers.

Cluster 1 with mean insurance cost of 33086 can be considered as high-risk customers.

Cluster 4 with mean insurance cost of 36904 can be considered as very high-risk customers.

From table 4.2 we can see that weight of the customer seems to be in direct correlation with the insurance cost values. Surprisingly fields such as age, bmi , fat percentage are more or less same across the clusters.

High risk customers have the maximum representation in the provided dataset, and they have the maximum average association with the company. This tells us that even though average insurance cost is high for these customers, they are vested with the institution.

Medium risk customers are the ones who are health conscious with minimal hospital admissions and maximum regular checkup conducted.

Surprisingly, the weight change in last year is negatively correlated to the insurance cost, which could indicate that positive weight change is weight reduction.

## Variable transformation

3 features, namely heart\_incident, decs\_incident and adventure\_sports are given as integer columns with values 0 and 1. These are in fact categorical variables and the column values are replaced with N and Y, corresponding to 0 and 1.

## Standardization

The numerical columns are scaled using Standard Scaler.

## Dummy encoding

The categorical variables are dummy encoded, in order to run through the various machine learning models.

# Model building

## Train-test split

We will do train and test split with KMeans\_cluster as the target variable and ensure to enable the stratify option(train to test split is kept at 30%). This will help to create the train and test datasets with similar cluster distribution:

Text

Description automatically generated Text

Description automatically generated

Table-4.1 Train and test target cluster variables distribution

So, we can see above that the train and test dataset have similar percentage distribution of the different clusters.

## Strategy

Since our target variable, insurance\_cost is a continuous variable, we will be looking at various regression models to train and test.

We will evaluate the model performance of linear, non-linear and ensemble regression models and choose the best model based on parameters such as Adjusted R-square, RMSE(Root Mean square) etc.

We will consider creating the following linear models:

* Linear Regression
* Lasso Regression
* Ridge Regression

We will choose the following non-linear models.

* Decision Tree Regression
* Artificial Neural Network (MLP Regression)
* KNN (K Nearest Neighbor) Regression

We will try some ensemble models and check the performance.

The ensemble models we are going to check are:

* Random Forest
* Bagging
* AdaBoost
* Gradient Boosting

We will be looking at the below performance metrics for train and test dataset:

* R-squared – This metrics explains the degree to which the predictor variables explain the variation of target variable. Higher the R-squared better the model.
* Adjusted R-squared – The adjusted R-squared is a variation of R-squared that accounts for predictors that are not significant in the model. We will be using this metric primarily to compare performance. Higher the metric better the model.
* MSE – Mean squared error – Mean of the square of the difference between actual and forecasted values. Lower the MSE better is the forecast.
* RMSE – Root Mean squared error – This indicates the standard deviation of the residuals, i.e., prediction error. Lower the RMSE, better is the forecast.
* MAE – Mean Absolute Error – Mean of the absolute residual values. Lower the MAE better the model.
* MAPE – Mean Absolute Percentage Error – This is a measure of prediction accuracy. Lower the value better the model.

## Linear Models

* Linear Regression

First we will try to do linear regression using ordinary least square method of stats models.

Create formula with target variable as insurance\_cost and predictors as all the columns in train dataset and fit the ols model on train dataset. Let us look at the summary of the model

We can see the R-squared and adjusted R-squared value at 0.944

All the variables where P-value is greater than 0.05 can be ignored as their contribution to the variance in target variable is insignificant.

Hence from table 4.2 the significant predictors are:

relationship\_years, checkup\_count, age, last\_admitted, weight, weight\_change

Table

Description automatically generated

Table-4.2 OLS Regression Summary

In the second run we set the formula as target variable, insurance\_cost, is a function of the significant variables: relationship\_years, checkup\_count, age, last\_admitted, weight, weight\_change.

We fit the new ols model against the train dataset and check the model summary as given below:

Table

Description automatically generated

Table-4.3 OLS Regression Summary-Final

We can see all the predictors are statistically significant and R-squared and Adjusted R-Squared is at 0.944 which indicates reliable performance.

* Lasso Regression

Lasso stands for Least Absolute Shrinkage & Selection Operator. Lasso regression is also known as regularized linear regression where regularization parameter multiplied by the summation of the absolute value of weights gets added to the loss function (ordinary least squares) of linear regression. The optimization of the Lasso loss function results in some of the weights becoming zero and hence can be seen as a method of selection of the features.

We will run lasso model through gridsearchcv with alpha values [0.005, 0.02, 0.03, 0.05, 0.06,0.1,0.2,0.3].

The gridsearch return Lasso(alpha=0.005) as the best model.

The features lasso model has considered as statistically important, with coefficients <> 0 are:

Table

Description automatically generated

Table-4.4 Lasso: Predictors with Coefficient >0

* Ridge Regression

Ridge Regression is a regularized linear regression that includes an L2 penalty. This has the effect of shrinking the coefficients for those input variables that do not contribute much to the variance of target variable.

We will run ridge model through gridsearchcv with alpha values ranging from 1 to 100 and then re-run based on the values around the best alpha of the model before until finally we reach to the best model.

The gridsearch return Ridge(alpha=32) as the best model.

The coefficient for features in Ridge model is given below:

Table

Description automatically generated

Table-4.5 Ridge: Predictor coefficient

Table 4.5 refers to the top 15 predictors in descending order of absolute value of coefficients. We can see that upto feature Location\_Kolkata the coefficient is in the order of 10-2 , after which the order reduces to 10-3 and so on. Weight is the feature with highest coefficient and significantly important than others.

## Non-Linear Models

* DecisionTree

DecisionTree Regression breaks down a dataset into smaller subsets while an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes.

We will run decision tree through gridsearchcv towards hyper parameter tuning.

The parameters through which we will start gridsearchcv are as follows:

'criterion' : ['mse’, ‘friedman\_mse', 'mae'],

'max\_depth' : [40,50,60],

'min\_samples\_leaf' : [50,100,150],

'min\_samples\_split' : [150,300,450]

After first run we can see that the model considers ‘mse’ as the best criterion, the numeric parameters we run multiple times to ensure that the final best models pick the parameters within the lower and upper bounds of the values specified.

The best model thus derived for DecisionTree regression is DecisionTreeRegressor(max\_depth=10, min\_samples\_leaf=35, min\_samples\_split=250).

If we look at the feature importance derived by decision tree in descending order, the first 15 are given below, where coefficient refers to the percent of variance of target variable explained by feature:

Graphical user interface, application

Description automatically generated

Table-4.6 DecisionTree: Feature importance

The 3 most prominent features are weight,last\_admitted and checkup\_count. From the 15th predictor Occupation\_Salried on all features are not considered.

* Artificial Neural Network

ANN regression is implemented via Multi-Layer Perceptron regression where the model devises hidden layers with non-linear learning capability.

We will run MLP regressor through gridsearchcv towards hyper parameter tuning.

The parameters through which we will start gridsearchcv are as follows:

"hidden\_layer\_sizes": [(1,),(2,),(3,),(4,),(5,)],

"activation": ["identity", "logistic", "tanh", "relu"],

"solver": ["sgd", "adam"],

"alpha": [0.00005,0.0005]

After first run we can see that the model considers ‘relu’ as the best activation and ‘adam’ as solver, the numeric parameters we run multiple times to ensure that the final best models pick the parameters within the lower and upper bounds of the values specified.

The best model thus derived for MLP regression is MLPRegressor(alpha=0.01, hidden\_layer\_sizes=(1,)).

If we look at the coefficient assigned to variables by MLP in descending order, the first 15 are given below, where coefficient refers to the percent of variance of target variable explained by feature:

Table

Description automatically generated

Table-4.7 ANN: Feature importance

The 4 most prominent features are weight, checkup\_count, Location\_Chennai and last\_admitted. From the 5th predictor in the table above the coefficient drops to the order of 10-3.

* KNN (K- Nearest Neighbor)

KNN regression approximates the association between independent variables and the continuous outcome by averaging the observations in the same neighborhood. It tries to ensure that observations are closest within its neighborhood and as far as possible from other neighborhoods.

We will run KNN regressor through gridsearchcv towards hyper parameter tuning.

The parameters through which we will start gridsearchcv are as follows:

'n\_neighbors': range(1, 50),

'weights': ['uniform', 'distance']

The best model thus derived for KNN regression is KNeighborsRegressor(n\_neighbors=14, weights='distance').

## Ensemble Models

* Random Forest

Random forests or random decision forests is an ensemble learning method for regression which operates by constructing a multitude of decision trees at training time. For regression, the model returns the mean prediction of the individual tree.

We will run Random Forest with gridsearchcv to hypertune the parameters.

We will start with the below parameter grid:

'criterion' :['mse','mae']

'max\_depth' : [190,200,210],

'min\_samples\_leaf' : [15,20,25],

'min\_samples\_split' : [105,110,115]

The gridsearchcv returns the best model as RandomForestRegressor(max\_depth=200, min\_samples\_leaf=15, min\_samples\_split=110).

* Bagging

Bagging or bootstrap aggregating is an ensemble technique that helps to improve the accuracy of algorithms. In this example we will use Bagging with base model as Random Forest regressor.

We will run bagging Regressor with gridsearchcv to hypertune the parameters.

We will start with the below parameter grid:

'base\_estimator\_\_criterion' :['mse','mae'],

'base\_estimator\_\_min\_samples\_leaf' :[15,20],

'base\_estimator\_\_min\_samples\_split' :[100,110],

'base\_estimator\_\_max\_depth' : [150,190]

We repeat the gridsearch with range of values based o the previous loop and so on to reach the optimal model as per gridsearchcv:

{'base\_estimator\_\_criterion': 'mse',

'base\_estimator\_\_max\_depth': 60,

'base\_estimator\_\_min\_samples\_leaf': 10,

'base\_estimator\_\_min\_samples\_split': 60}

* AdaBoost

AdaBoost can be used to boost the performance of any machine learning algorithm. It is best used with weak learners. These are models that achieve accuracy just above random chance on a classification problem. The most suited and therefore most common algorithm used with AdaBoost are decision trees. We will use gridsearchcv for hyper tuning and start with below parameters:

We will start with the below parameter grid:

"base\_estimator\_\_criterion" : ["mse", "mae"],

"base\_estimator\_\_splitter" : ["best", "random"],

"base\_estimator\_\_max\_depth" : [5,10],

"base\_estimator\_\_min\_samples\_leaf" : [30,35,40],

"base\_estimator\_\_min\_samples\_split" : [225,250,275]

We repeat the gridsearch with range of values based on the previous loop and so on to reach the optimal model as per gridsearchcv:

'base\_estimator\_\_max\_depth': 10,

'base\_estimator\_\_min\_samples\_leaf': 25,

'base\_estimator\_\_min\_samples\_split': 150

* GradientBoosting

GradientBoosting is a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees. When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest. We will use gridsearchcv for hyper tuning and start with below parameters:

We will start with the below parameter grid:

'criterion' :['friedman\_mse', 'mse'],

'max\_depth' : [10,15],

'min\_samples\_leaf' : [30,35],

'min\_samples\_split' : [225,250]

We repeat the gridsearch with range of values based on the previous loop and so on to reach the optimal model as per gridsearchcv:

{'max\_depth': 10, 'min\_samples\_leaf': 25, 'min\_samples\_split': 300}.

# Model Validation

Let us validate all the model’s by checking the performance against both train and test dataset.

## Linear Models

* Linear Regression

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

Table

Description automatically generated

Table-5.1 Linear Regression: Performance Metrics

We can see from table 5.1 above that Adjusted R-squared value is 0.9438 and 0.9437 for train and test datasets respectively, i.e., the model is able to explain 94.37% of the variance in test target variable using the test predictors. This is very good performance. We can see that RMSE value is 0.2369 for train and 0.2383 for test dataset. The model neither is underfit or overfit.

* Lasso Regression

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

A screenshot of a computer

Description automatically generated with medium confidence

* Table-5.2 Lasso Regression: Performance Metrics

We can see from table 5.2 above that Adjusted R-squared value is 0.9436 for train and test datasets respectively, i.e., the model is able to explain 94.36% of the variance in test target variable using the test predictors. This is very good performance. We can see that RMSE value is 0.2373 for train and 0.2385 for test dataset. The model neither is underfit or overfit.

* Ridge Regression

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

Table

Description automatically generated

* Table-5.3 Ridge Regression: Performance Metrics

We can see from table 5.3 above that Adjusted R-squared value is 0.9438 and 0.9437 for train and test datasets respectively, i.e., the model is able to explain 94.37% of the variance in test target variable using the test predictors. This is very good performance. We can see that RMSE value is 0.2369 for train and 0.2383 for test dataset. The model neither is underfit or overfit.

## Non-Linear Models

* DecisionTree

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

Table

Description automatically generated

* Table-5.4 DecisionTree Regression: Performance Metrics

We can see from table 5.4 above that Adjusted R-squared value is 0.9558 and 0.9530 for train and test datasets respectively, i.e., the model is able to explain 95.3% of the variance in test target variable using the test predictors. This is very good performance. We can see that RMSE value is 0.21 for train and 0.2177 for test dataset. The model neither is underfit or overfit.

* Artificial Neural Network

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

Table

Description automatically generated

* Table-5.5 ANN Regression: Performance Metrics

We can see from table 5.5 above that Adjusted R-squared value is 0.9503 and 0.9505 for train and test datasets respectively, i.e., the model is able to explain 95.05% of the variance in test target variable using the test predictors. This is very good performance. We can see that RMSE value is 0.2227 for train and 0.2236 for test dataset. The model neither is underfit or overfit.

* KNN (K- Nearest Neighbor)

We predict the target variable against both train and test predictor datasets. Then we find the performance metrics against the prediction in both train and test datasets. We get the below information:

A screenshot of a computer

Description automatically generated with medium confidence

* Table-5.6 KNN Regression: Performance Metrics ‘

We can see from table 5.6 above that Adjusted R-squared value is 1 and 0.8543 for train and test datasets respectively, i.e., the model is able to explain 85.43% of the variance in test target variable using the test predictors. The model is clearly overfit and has below par performance compared to the above 5 models. We can see that RMSE value is 0 for train and 0.3834 for test dataset. The model is not apt for this regression scenario.

## Ensemble Models

* Random Forest

We predict the target variable on train and test dataset via the model and let’s check the performance:

Table

Description automatically generated

Table-5.7 Random Forest – Performance metrics

We can see that the model has reliable performance with Adjusted R-square values of 0.9592 and 0.9544 on the train and test dataset. The RMSE values are also low are 0.2017 and 0.2145. The model neither exhibits overfit or underfit behavior.

* Bagging

We predict the target variable on train and test dataset via the model and let’s check the performance:

Table

Description automatically generated

Table-5.8 Bagging – Performance metrics

We can see that the model has reliable performance with Adjusted R-square values of 0.9599 and 0.9544 on the train and test dataset. The RMSE values are also low at 0.2 and 0.2145. The model neither exhibits overfit or underfit behaviour.

* AdaBoost

We predict the target variable on train and test dataset via the model and let’s check the performance:

Table

Description automatically generated

Table-5.9 AdaBoost – Performance metrics

We can see that the model has reliable performance with Adjusted R-square values of 0.9633 and 0.9520 on the train and test dataset. The RMSE values are also low at 0.1913 and 0.2202. The model neither exhibits overfit or underfit behaviour.

* GradientBoosting

We predict the target variable on train and test dataset via the model and let’s check the performance:

Table

Description automatically generated

Table-5.10 Gradient Boost – Performance metrics

We can see that the model has reliable performance with Adjusted R-square values of 0.9679 and 0.9541 on the train and test dataset. The RMSE values are also low at 0.1789 and 0.2152. The model neither exhibits overfit or underfit behaviour.

## Model Performance Comparison

Comparing the performance of all the models against the train and test datasets.

Graphical user interface, application

Description automatically generated

Table-5.11 Gradient Boost – Performance metrics

Graphical user interface

Description automatically generated with low confidence

Table-5.12 Gradient Boost – Performance metrics

Adjusted R-square value is highest for test datasets of Random Forest and Bagging.

Random Forest train/test datasets values are closer; hence Random Forest has the best performance of all the 10 models built

Linear Regression is at par with Random Forest model (with Adjusted r-squared difference of less than 0.0107).

Linear Regression are less complex & faster to execute resulting in lower execution costs.

Hence, we can consider linear regression as the best model of the lot.

# Final Interpretation/recommendation

* Linear regression is the optimum model to predict insurance costs for the given features, as it has comparable performance with all other models(Random Forest has the highest performance of all models, but RF model is complex and time consuming in real world scenarios with large datasets).
* Linear regression in this scenario can be expressed as a function of predictor and target variables as follows:

insurance\_cost = -1.078e-16 + (0.9692)weight + (0.0290) last\_admitted + (0.0174) weight\_change+

(0.0044) age – (0.0283) checkup\_count – (0.008) relationship\_years

* Weight is the most significant of features in determining the insurance cost.
* Other features of importance are last\_admitted, weight\_change, age, checkup\_count and relationship\_years.
* Considering weight (taken at time of enrollment) to be the most important feature might not be optimal. Height should also be taken into consideration. A better factor to weigh insurance\_cost should be BMI or body fat percentage, which gives a better indication of patient’s health.
* Smoking is also a huge factor in most of the health issues, insurance cost should be heavily correlated to smoking habits, which can be identified via blood tests. This would promote health lifestyle for the customers.
* Age, even though present in the dataset and in the important feature, the correlation seems to be weak. This is another factor that business needs to reconsider.
* Being in the age of digital revolution and widespread use of smart bands/watches, the company should promote healthy lifestyle with brownie points for more steps walked over a period, which can be sourced from the members smart watch information.
* The dataset seems to have missed critical information such as frequency and amount of past claims, which should have a considerable impact on the insurance cost.
* Even though we have information on customers covered by any other company, we should have insights on why the cover with one company is not enough based on the customers' profile.
* Dependents information is also not provided, though these might be individual insurances, this needs to be clarified with business.

# END