The Netherlands Press

Journal of Emerging Strategies in New Economics

Article

PREDICTING THE INSURANCE CLAIM BY EACH USER USING MACHINE LEARNING ALGORITHMS

Seshu Kumar Vandrangi

Doctoral Research Fellow,
Department of Mechanical Engineering,
Universiti Teknologi PETRONAS,
Persiaran UTP, 32610 Seri Iskandar, Perak, Malaysia

Orchid Id: https://orcid.org/0000-0002-5806-6913

E-mail: Seshu1353@gmail.com

Abstract.

Today, data will play a critical role and become a significant wealth creator in the insurance sector. The insurance sector is very significant in today's travel industry. Insurance companies now have more information than ever before. There have been three key eras in the insurance sector over the last 700 years. The manual era lasted from the 15th century to the 1960s, the system age from the 1960s to the 2000s, and now the digital age. H. 2001-20X0. In all three periods, the ultimate corporate objective has been pushed by the core insurance industry's conviction in data analytics to accept evolving technologies in order to enhance its route and concentrate capital. That's all. Inadequate analytical models and algorithms to serve insurance businesses is a big concern in advanced analytics. Only machines are capable of overcoming this obstacle. In this paper, insurance data based on some features are trained and tested over Artificial Neural Networks, Random Forest Regressors, Logistic Regression and predict the charges based on features for predicting the insurance claim.

Keywords: Data pre-processing, Artificial Neural Networks, Capital, Random Forest Regressors, Logistic Regression.

Journal of Emerging Strategies in New Economics Volume 1 Issue 1

Received Date: 08 July 2022
Accepted Date: 12 August 2022
Published Date: 07 September 2022

1. Introduction

The capacity to simplify datasets is a fundamental advantage of employing machine learning (ML) in the insurance sector. Machine learning (ML) is effective with organised, semi-structured, and unstructured datasets. Use machine learning across the value chain to accurately forecast risks, claims, and consumer behaviour. Machine learning has several applications in insurance, ranging from perceived risk-taking and premium loss to expenditure management, regression, processes, and fraud detection. Machine learning is not a novel concept. This technology has been around for a long time. Learning is classified into three types: supervised learning, unsupervised learning, and reinforcement learning. Following decades of supervised learning, the majority of insurers estimate risk and accomplish desired results by combining known parameters. If variables change, the technique detects them and seeks to alter them in accordance with the purpose. For example, depending on traffic circumstances, GPS will dynamically recommend alternative routes[1]. Learning is also being implemented into pay-as-you-go insurance in the insurance sector. Reinforcement learning is mostly based on ANN (Artificial Neuron Network), which may adjust the target/goal dynamically depending on the purpose. His IOT application will employ reinforcement learning.

Insurance Consulting: From scheduling advance contact to assessing and responding to consumer needs, machines play an important part in customer service. According to recent study, more clients are pleased to get computer-generated insurance suggestions. Customers desire tailored solutions generated by machine learning algorithms that assess profiles and offer tailored items. PROPERTY/CASUALTY Innovation and Emerging Technologies, SMA Research, 2016. We can see from the above chart that the n=84 insurance business is machine learning and may be exploited[2]. Throughout the insurance value chain, machine learning is frequently applied. Allstate, for example, collaborated with (EIS) Early Information Science to create the Allstate Business Insurance Expert, a virtual assistant (ABIE).

Claims Processing: Machine learning is being used by insurers to improve operational excellence from claim registration through claim payment. Many operators have already started automating their billing operations, which improves the user experience while decreasing billing time. Insurers can also benefit from machine learning and prediction models to better understand claim expenses[3]. Through proactive management, faster turnaround, focused investigations, and improved case management, these insights help carriers save millions of dollars in claims expenses. Insurers can be more confident in the amount of money they set up for claims reserves. Tokyo Marine offers an AI-assisted damage document detection system that uses a cloud-based AI optical character recognition (OCR) service to aid handle handwritten damage reports. Reduce document input burden by half while adhering to data protection laws. With "packet-like" data exchange, AI understands difficult and ambiguous kanji while protecting client privacy. As a consequence, 90%+ of claims are detected, input time is reduced by half, human error is reduced by 80%, and claims are paid faster and with less difficulty.

Price optimization: Price optimization approaches are difficult to grasp. As a result, it employs diverse combinations of various methodologies and algorithms [4]. The use of this technique for insurers is still contentious, although the practise is being adopted by an increasing number of insurers. This procedure presupposes that data unrelated to projected cost and risk features are coupled with data unrelated to expected losses and costs and analysed further. That is, changes from the previous year and policy are considered. As a result, price optimization is inextricably linked to consumer price sensitivity. In other words, prior costs, expenses, damages, risks, and advantages are carried forward [5].

A sophisticated algorithm enables insurance firms to alter premiums on the fly. Price optimization helps boost consumer loyalty in the long run, which goes hand in hand with increasing profits.

Customer segmentation: Modern technology has elevated product and service advertising to a whole new level. Customers typically have different expectations of their insurance company. Insurance marketing employs a variety of tactics to increase the number of clients and assure focused marketing strategies. As you can see, client segmentation is a crucial strategy in this case. Algorithms separate customers based on financial experience, age, region, and other factors. All clients are therefore classified into groups based on similarities in attitudes, interests, behaviours, or personal information[6]. This grouping enables us to design attitudes and solutions that are unique to each customer. As a consequence, you may create targeted cross-selling rules and provide tailored and personalised service to certain customer categories.

Fraud Prevention: False claims cost insurers an estimated $30 billion each year. Machine learning assists in more rapidly and correctly identifying probable fraudulent claims and flagging them for inquiry. Machine learning algorithms outperform traditional predictive models in this application because they can identify possible fraud using unstructured and semi-structured data, such as billing records and documents, as well as structured data. Chola MS, one of India's fastest growing insurance firms, has implemented mobile technology in their claims processing. The company's car measuring programme captures and stores vehicle measurement data in a database using the voice, camera, and data networking capabilities of Samsung Galaxy tablets. Before estimating damage, surveyors had to manually match survey notes with emails and images maintained in other systems. This programme has aided in the acceleration of the claims process, increased adjuster productivity, and enhanced fraud protection.

Management of Risk Machine learning is used by insurers to anticipate premiums and victims for their policies. Detecting risks early in the process allows insurers to make better use of underwriters' time, giving them a significant competitive advantage. Progressive Insurance is clearly utilising machine learning algorithms for predictive analytics using data obtained from customer drivers[7]. The auto insurance claims that its Snapshot telematics smartphone app has acquired 14 billion miles of driving data. Progressive gives most drivers an auto insurance savings of US$130 after six months of use to promote the adoption of Snapshot.

2. IMPLEMENTATION OF PREDICTING RISK:

This dataset forecasts insurance claims for each user, does regression analysis with machine learning methods, and creates data visualisations to aid in the study. A BMI that recognises excessive or low weight relative to height, preferably an objective measure of weight (kg/m2) utilising a height-to-weight ratio of 18.5 to 25.

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

2.1. EXPLORATORY DATA ANALYSIS:

The price column is the dependent variable of the data or labels, and when plotted as a lmplot and examined against the age column, regplot() always displays a single connection, but lmplot() combines regplot() with a Facet Grid. When used together, they provide a straightforward interface for viewing linear regression in a "faceted" graphic that allows you to investigate interactions with up to three additional categorical variables. Because of this parcel, the fee field is age dependent.

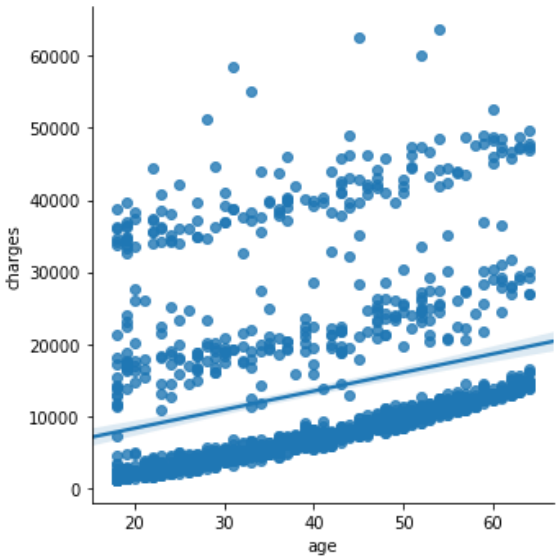

Figure 1: age vs charges plot

The price rises with age, depending on your health state at each age. As a result, if you're younger, you could pay less. In addition, elderly adults will have greater health issues. Regarding the male-female ratio, the statistics for males and females are the same, therefore training on this data may be acceptable from a gender standpoint.

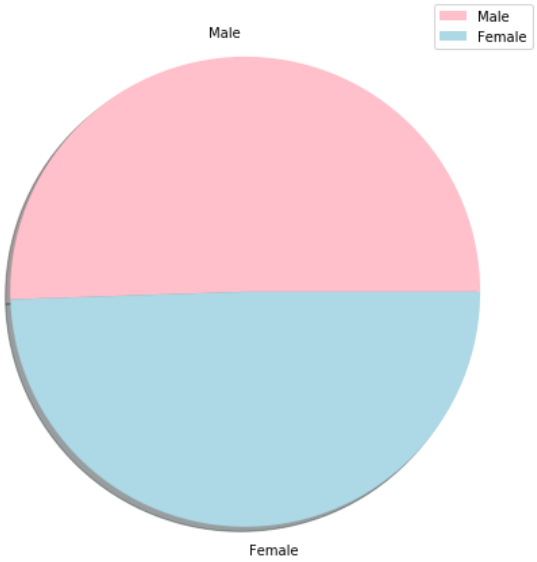

Figure 2: share of men and women plot

The data contains around 20 years more, 40 years is the intermediate number, and 60 years is too high, according to the age histogram plot. This enables us to extract age differences from the data.

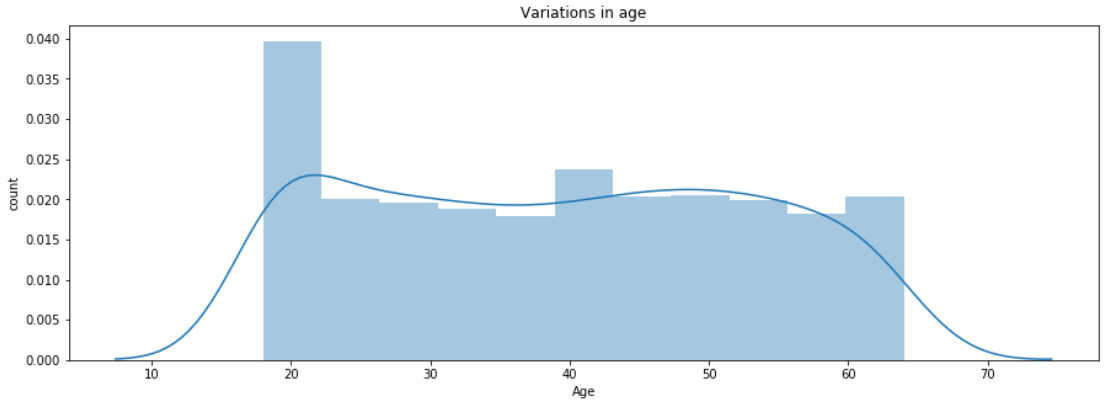

Figure 3: histogram plot for age

The minimum charges are essentially comparable throughout all areas, with some outliers in the data for the Southeast, Northwest, and Northeast regions, according to the boxplots of regions and charges. These outlines can be eliminated or processed using z-score or percentile approaches. Each value's Z-score is the number of standard deviations above and below the mean. A z-score of 2 implies the observation is 2 standard deviations above the mean, whereas a z-score of -2 means it is 2 standard deviations below the mean. Outliers are defined as Z-score values larger than three and less than three. These outliers are eliminated or addressed by substituting the mean or median for them. Averages are useful for dealing with outliers. So you can figure it out by trial and error.

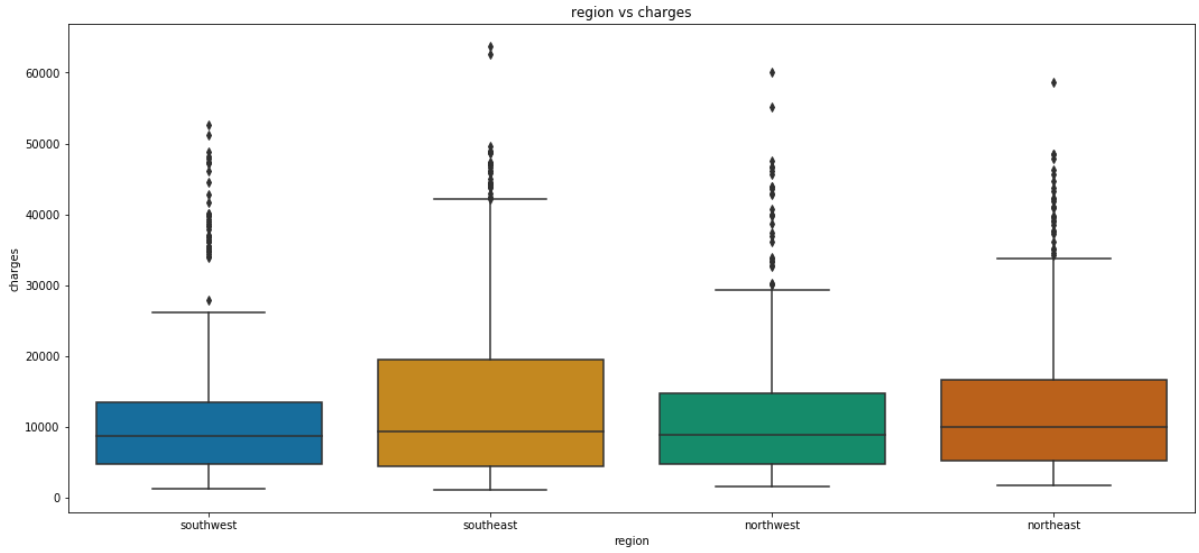

Figure 4: Box plot for region vs charges

According to the data, there are more smokers than non-smokers. Insurance companies care about this illness since smoking shortens your life.

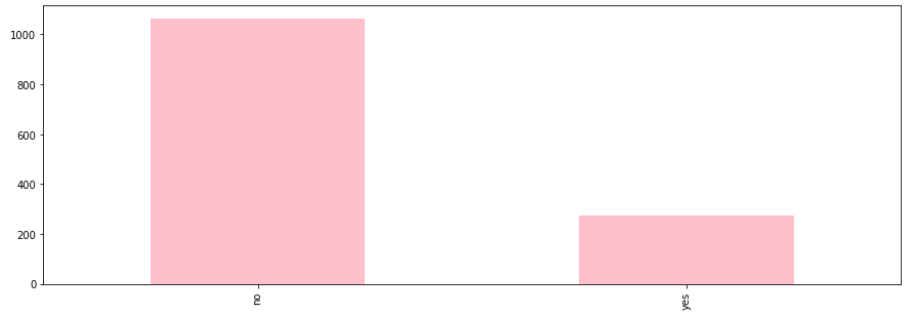

Figure 5: count of smokers and non-smokers

2.2. DATA PREPROCESSING:

To begin pre-processing the data, null values are verified. Because there are no null values, data processing is required. Age, Gender, BMI, Children, Smokers, Region, and Price are among the columns contained in the data. We require the mean, median, and mode to determine the data statistics. Explanatory functions are utilised to do this since categorical functions such as gender and geography overlap with numeric functions. The presence or lack of variance as shown by the explanatory table. Everything appears to be in order. Nothing has changed

	age	bmi	children	charges
count	1338.000000	1338.000000	1338.000000	1338.000000
mean	39.207025	30.663397	1.094918	13270.422265
std	14.049960	6.098187	1.205493	12110.011237
min	18.000000	15.960000	0.000000	1121.873900
25%	27.000000	26.296250	0.000000	4740.287150
50%	39.000000	30.400000	1.000000	9382.033000
75%	51.000000	34.693750	2.000000	16639.912515
max	64.000000	53.130000	5.000000	63770.428010

2.3. DATA SCALING:

The mean is removed, and each feature/variable is scaled to unit variance using StandardScaler. This surgery is distinguished by its independence. Outliers can affect Standard Scaler's estimation of the empirical mean and standard deviation of each characteristic (if present in the dataset).

```
# standard scaling

from sklearn.preprocessing import StandardScaler

# creating a standard scaler
sc = StandardScaler()

# feeding independents sets into the standard scaler
x_train = sc.fit_transform(x_train)
x_test = sc.fit_transform(x_test)
```

Using the replace function, replace the category column, such as the sex column, with male as 1 and female as 0. Furthermore, for region column one, hot encoding works nicely. One of the strategies used to make this conversion is one-shot encoding. This strategy is commonly utilised when applying deep learning techniques to sequential classification tasks. Categorical variables are represented as binary vectors in one-hot encoding.

```
# splitting the dataset into training and testing sets

from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 30)
```

Characteristic information is collected as X. This covers age, gender, BMI, kid, smoker, area, and price labelled Y. You may include this into your machine learning model by sharing it. This data is used for training and testing, with a test size of 20% of the data and 30 randomly selected states.

3. TRAINING WITH MACHINE LEARNING MODEL:

For model selection, a random forest regressor from the Scikit learning package is utilised. Regressor with a Random Forest. A random forest is a meta-estimator that employs averaging to increase prediction accuracy and control overfitting by fitting a group of classification decision trees to distinct subsamples of a dataset. This model is loaded from the scikit learning library, and the training data is fitted to it. The model prediction function feeds test data into the model in order for it to predict labels. The mean squared error may be calculated by subtracting the anticipated label from the actual label, applying the mean, then squaring it. Mean squares are used in regression to assess whether or not terms in the model are significant. Divide the sum of squares term by the degrees of freedom to get the mean square term. Divide the residual sum of squares by the degrees of freedom to get the mean squared error (MSE). To calculate the RMSE, use root. The standard deviation of the residuals is defined as the root mean square error (RMSE) (prediction errors). The distance between the data points and the regression line is measured by residuals. The RMSE is a measure of the distribution of these residuals. In other words, it indicates the degree to which the data is crowded around the line of best fit.

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score

# creating the model
model = RandomForestRegressor(n_estimators = 40, max_depth = 4, n_jobs = -1)

# feeding the training data to the model
model.fit(x_train, y_train)

# predicting the test set results
y_pred = model.predict(x_test)

# calculating the mean squared error
mse = np.mean((y_test - y_pred)**2, axis = None)
print("MSE :", mse)

# Calculating the root mean squared error
rmse = np.sqrt(mse)
print("RMSE :", rmse)

# Calculating the r2 score
r2 = r2_score(y_test, y_pred)
print("r2 score :", r2)
```

The r2 score computation evaluates the effectiveness of a linear regression model using the coefficient of determination, often known as the R2 score. It is the degree of variation in an output-dependent attribute that can be anticipated based on the independent variables in the input. Another model in Scikit Learn is Principal Connection Analysis, often known as Principal Component Analysis (PCA). Linear dimensionality reduction is the process of projecting data into a lower-dimensional space by applying singular value decomposition. Prior to applying SVD, the input data is countered but not scaled for each feature. Training is carried out with the use of appropriate training and test data models.

```
# feature extraction

from sklearn.decomposition import PCA

pca = PCA(n_components = None)

x_train = pca.fit_transform(x_train)
x_test = pca.transform(x_test)
```

4. BUILDING AN ARTIFICIAL NEURAL NETWORKS:

Import the initial sequential model from the Keras library and set up the layers to develop an artificial neural network model. Place the first layer to populate the model with the number of neurons equal to the length of the feature column. The number of neurons in the first and buried layers varies, while the output layer is made up of a single neuron that predicts charge. The rectified linear unit activation function and the ReLu output layer are the only components of the hidden layer.

The SGD algorithm is a gradient descent modification that addresses some of the drawbacks of the GD method. One downside of gradient descent is that it needs a considerable amount of memory to load the complete n-point dataset at once to compute the derivative of the loss function. The performance of classification models based on probabilities and mistakes is measured by cross-entropy. The lower the cross-entropy, the more likely (or more probable).

```
import tensorflow as tf
from tensorflow import keras


model = keras.Sequential([
    keras.layers.Dense(len(x.columns), input_shape=(20,), activation='relu'),
    keras.layers.Dense(3, activation='relu'),
    keras.layers.Dense(1, activation='relu')
])

# opt = keras.optimizers.Adam(learning_rate=0.01)

model.compile(optimizer='SGD',
              loss='cross entropyloss',
              metrics='r2_score')

model.fit(X_train, y_train, epochs=50)
```

5. RESULTS AND DISCUSSIONS:

For 50 epochs, the loss is back propagated and optimised throughout the network by fitting the data into the model using a loss and optimization function. After 50 epochs, the accuracy gained was 91.37% and the test accuracy was 91.02%. After training and testing the model, the final output is passed via an inverse transform function and checked using model evaluation performance approaches.

```
Epoch 38/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1974 - accuracy: 0.9136
Epoch 39/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1971 - accuracy: 0.9138
Epoch 40/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1971 - accuracy: 0.9132
Epoch 41/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1971 - accuracy: 0.9140
Epoch 42/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1970 - accuracy: 0.9141
Epoch 43/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1965 - accuracy: 0.9136
Epoch 44/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1968 - accuracy: 0.9132
Epoch 45/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1968 - accuracy: 0.9142
Epoch 46/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1963 - accuracy: 0.9136
Epoch 47/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1960 - accuracy: 0.9144
Epoch 48/50
928/928 [==============================] - 2s 3ms/step - loss: 0.1966 - accuracy: 0.9140
Epoch 49/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1961 - accuracy: 0.9141
Epoch 50/50
928/928 [==============================] - 2s 2ms/step - loss: 0.1964 - accuracy: 0.9137
<keras.callbacks.History at 0x7f2246e08cd0>
```

```
model.evaluate(X_test, y_test)
```

```
232/232 [==============================] - 1s 3ms/step - loss: 0.2070 - accuracy: 0.9102
[0.2070474475622177, 0.9102062582969666]
```

The result achieved by utilising mean squared error is presented below, along with root mean square error and coefficient of determination, which demonstrates how accurate the model was, and r2 score is more than 0.75 and close to 0.80. As a result, this model may be used to forecast insurance claims.

```
MSE : 31195183.386636455
RMSE : 5585.264844806955
r2 score : 0.7977661591808768
```

6. CONCLUSIONS:

Modern technologies are rapidly making their way into numerous industries of business. In these areas, the insurance business does not lag behind the competition. Statistics have long been used in the insurance industry. As a result, the fact that insurance firms are actively employing data science analytics is not surprising. In essence, the goal of using data science analytics in insurance is the same as it is in other industries: to optimise marketing tactics, improve company, increase income, and decrease expenses. In this research, we presented multiple machine learning approaches for effectively analysing insurance claims and comparing their performance using various criteria.

References

[1] Belhadji, E., G. Dionne, and F. Tarkhani, ―A Model for the Detection of Insurance Fraud, Geneva Papers on Risk and Insurance Theory‖, 25: 517-538, May 2012.

[2] Crocker, K. J., and S. Tennyson, ‖ Insurance Fraud and Optimal Claims Settlement Strategies: An Empirical Investigation of Liability Insurance Settlements ‖ The Journal of Law and Economics, 45(2), April 2010.

[3] Kajiamuller, ―The Identification of Insurance Fraud – an Empirical Analysis Working papers on Risk Management and Insurance ‖ no: 137, June 2013.

[4] S. B. Kotsiantis, ―Supervised Machine Learning: A Review of Classification Techniques, ‖ Informaticavol 31, pp 249- 268, May 2011.

[5] Sivarajah U, Kamal M, Irani Z, Weerakkody V (2017) Critical analysis of big data challenges and analytical methods. J Bus Res 70:263–286

[6] Mishr K (2016) Fundamentals of life insurance theories and applications. In: 2nd ed, Delhi: PHI Learning Pvt Ltd

[7] The Kaggle Website. [Online]. https://www.kaggle.com/c/ prudential-life-insurance-assessment/data/