Olisa Unegbu 22020843

ANALYZING LOCAL RISK CONCERNS IN LONDON: DISCOVERING TRENDS AND PATTERNS

UNEGBU, OLISA UDOCHUKWU
22020843

Abstract

With the recent UK-wide test of emergence alert system which took place at 15:00 on Sunday 23rd April 2023 from the Cabinet Office of National Situation Centre which was received by every 4G and 5G across United Kingdom, it has become pertinent to alert residents on time of any possible or likely risk that may occur within its environment. This report mainly discusses the concerns raised by residents in different wards and boroughs in London with the focus on the overall local risk concerns raised. Therefore, an analysis of the local risk concern is performed in this paper to provide relevant recommendations for how different preventive control measures can be targeted effectively.

The Neural Network and Principal Component Analysis (PCA) models are used to determine the factors highly responsible for the local risk concerns. The Neural Network model achieving an r^2 score of 0.97 explains a high percentage of variability in the target variable (Concerns) while the PCA model explains a total of 54.9% of the variance in the original data. The result of the analysis shows that population, smokers, and disabled individuals dominated areas are significant risk factors for local risk concerns raised in London.

Key words: local risk concerns; analysis; Neural Network; Principal Component Analysis (PCA), r^2 score.

Introduction

Urban areas face numerous types of risks such as higher cost of living, increased violence, environmental hazards, crime, social unrest, social inequality, and economic instability. London, being one of the largest and most diverse cities in the world with a recent population of over nine million people, faces severe unique risks. According to Macrotrends, London had a population of eight million, nine hundred and sixteen thousand (8,916,000) people in 2017 when this data was collected and has experienced a growth rate of 8.2% increasing its population to nine million, six hundred and forty-eight thousand (9,648,000) people presently. A comprehensive set of possible preventive measures need to consider all options and, even though information on effectiveness for specific risk groups are vital, information on effectiveness of different measures for all groups are needed for broader interventions (Marcus R., Nils J., & Patrick V. H. 2017).

To assist members of the public in gaining a better comprehension of how risk has undergone changes over time and how distinct factors interconnect to shape it, the Assessment of Local Dangers (AoLR) report was produced. The AoLR 2017 provides a concise overview of potential threats that may arise from fires and other disasters in London, its primary objective being public education. Reading its executive summary may prove beneficial when attempting to grasp key principles surrounding fire protection practices as well as LFB's efforts towards ensuring citizen well-being. This research aims to examine local risks in London through identifying patterns and trends in the

data. To achieve this goal, the information from the Assessment of Local Risks (AoLR) that accompanied LSP2017 was utilized. In this study, both a K-Means model and a neural network with an impressive R-squared score of 0.96 were implemented.

As it enables organizations and government to understand the risks posed by various factors and develop strategies to mitigate them, the analysis of local risk is a crucial area of research. Many of the studies that have been done in this field have centered on the application of data analytics to the study of local risk. To find earlier research on the subject and to better understand the data being used, a literature review was first conducted. During the literature review, it was discovered that prior research had utilized a range of techniques, such as regression analysis, geographical analysis, and clustering techniques, to evaluate risk in London. However, only a small number of studies had used machine learning models to analyze the data, such as Neural Networks or Principal Component Analysis (PCA).

For instance, one study by Jansson et al. (2018) examined the dangers posed by several forms of crime in a city using a combination of statistical and machine learning techniques. According to the study, machine learning techniques can be used to spot patterns and trends that are not readily apparent using conventional statistical techniques.

Another study by Zijiang Zhu & Yu Zhang (2022) discusses flood disaster risk assessment based on random forest algorithm which was used as the weight of each parameter of the flood disaster index model. The study results show that the combination of random forest algorithm and GIS technology is convenient for analyzing the spatial pattern and internal laws of flood risk and has good applicability.

Overall, these studies highlight the value of systems thinking, data analytics and spatial patterns in the study of local risk. Utilizing these techniques, organizations and government can improve their understanding of the risks presented by different factors and create practical mitigation strategies.
The neural network and PCA techniques were combined to analyze the AoLR data to fill this gap in the literature. During the analysis, population, smokers, and disabilities were shown to be the most crucial factors in determining local risk, with a strong association to the target variable according to our key findings.

Methodology

The Assessment of Local Risk in London data is gotten from the London Datastore (Assessment of Local Risks supporting the London Safety Plan 2017 - London Datastore). The data collected was for 2017. The dataset 24,375 datapoints which includes 625 observations and 39 attributes. Table 1a and 1b below shows the attributes names and types.

	Data Type
Borough code	object
Borough name	object
Ward Code	object
Ward name	object
Concerns	int64
Consequence	int64
Control	int64
Population	int64
Density	int64
65+ age	int64
Deprivation -2015	int64
HR buildings	int64
Heritage	int64
local concerns	int64
Student	int64
crime	int64
Over crowding	int64
No central heat	int64
older alone	int64
Disability	int64
Smokers	int64
1 pump fires	int64
2+ pump fires	int64

Table 1a: metadata table.

1 pump special services	int64
2+ pump special services	int64
Fire casualties	int64
Special services casualties	int64
entry / exit	int64
flooding	int64
making safe	int64
medical	int64
rtc	int64
Rescue / Release	int64
1st pump attendance	object
2nd pump attendance	object
3rd pump attendance	object
HFSV	int64
FSR Inspection	int64
Visual audit	int64

Table 1b: continuation of the metadata table.

During the Exploratory Data Analysis (EDA), the info() function was used to get summary of the data which range index of the data with 625 entries and 39 columns while the describe function was used to get the mean, std, min, 25%, 50%, 75%, and max of all the numeric variables. The isnull().sum() function was used to check for missing values while isin(['error_value']).count function was used to check for error values. Both functions returned zeros (0) for all attributes except for '3rd pump attendance' that returned 23 missing values.

	Data Type
Concerns	int64
Population	int64
Density	int64
65+ age	int64
Deprivation -2015	int64
HR buildings	int64
Heritage	int64
local concerns	int64
Student	int64
crime	int64
Over crowding	int64
No central heat	int64
older alone	int64
Disability	int64
Smokers	int64

Table 2: cleaned metadata table.

Since the focus of this study is on variables where more concerns for risk were raised, the drop() function was used to remove variables with little or no effect to the goal variable. The cleaned metadata table is shown in table 2 above.

Fig 1 below shows the scatterplot and histogram chart of cleaned data.

Olisa Unegbu 22020843

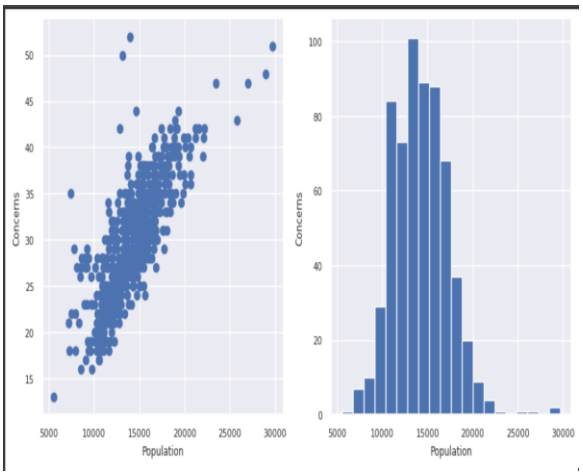

fig. 1: showing the scatterplot and histogram chart of the cleaned data.

Fig 1 above shows the relationship between the dependent variable (Concerns) and the independent variable (Population). The plots show a positive relationship between dependent variable and the independent variable.

After concluding the Exploratory Data Analysis (EDA), the correlation analysis technique was used to determine the relationship between the independent variables and the dependent variable. Correlation analysis according to B. Lantz (2015), is a statistical technique used to determine the relationship between two or more variables. It explains covariance between two or more variables. Covariance is a measure of how changes in one variable are associated with changes in a second variable. The corr() function was used to calculate the correlation of each variable with the target variable and the result shows that Population has the highest relationship with the target variable with a correlation of 0.79. The sns.heatmap() function from seaborn python library was used to plot the correlation heatmap as shown in fig. 2 below:

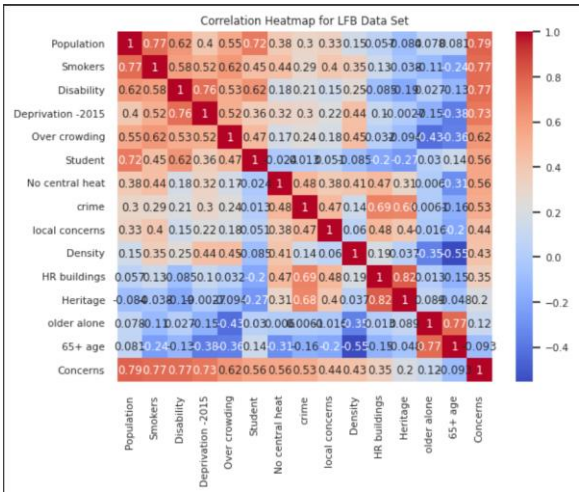

fig. 2: showing the correlations between the variables.

During the data mining model building process for the unsupervised learning method, several unsupervised methods like Hierarchical clustering, K-means, and PCA were tried. PCA, being a powerful unsupervised learning method for dimensionality reduction extracts information of high dimensional data and models it in low dimensions, Solis (2023).

The Principal Component Analysis (PCA) unsupervised learning method was chosen because it extracts information of the AoLR dataset which happens to be a high dimensional data and models it in low dimensions. First, the StandardScaler() function from sklearn.preprocessing python library was used to standardize the data thereby ensuring that all variables have zero mean and unit variance. Next, an instance of PCA() function from sklearn.decomposition python library with n_components = 2 was created. This extracts two principal components. Then, the PCA model was fitted to the standardized data using the fit() function, while the transform() method was used to transform the standardized data. Note: the transformed data contains the principal components that represent the AoLR data in a low dimension. The AgglomerativeClustering() function from sklearn.cluster python library was used to cluster the data points according to their similarity. The

pca.explained_variance_ratio_ method was used to produce the explained variance ratio of the principal components, which represents the proportion of variance in the AoLR data that is explained by each principal component while the pca.components_ and pca.explained_variance_ methods were used to extract the eigenvectors and eigenvalues of the principal components from the PCA objects respectively.

Unsupervised learning can aid in the implementation of supervised learning by providing insights and knowledge about the patterns and structure present in the dataset. It can be used for attribute extraction, exploratory data analysis (EDA), and in this study, for dimensionality reduction which can be used as input features for supervised learning algorithm. Generally, unsupervised learning can be used as a preparation step to extract insights and knowledge from the data which can then be used to improve the performance of the supervised learning algorithms.

During the data mining model building process for supervised learning methods, several methods like Decision Tree, Logistic Regression, Neural Network, and Support Vector Machine were all tested. The artificial neural network model was inspired by the biological neural network of the human brain, comprising billions of interconnected neurons, each receiving and sending signals from and to other neurons, Solis (2023).

The neural network supervised learning method was used because it gave a high r^2 score. The machine learning model was built using a neural network to predict the target variable based on the other attributes represented by the X-variable. The get_dummies() function from the pandas python library was used convert the categorical attributes into dummy variables. The train_test_split() function from the sklearn.model_selection python library was used to split the data into training and testing sets of 80% and 20% respectively while 42 was chosen as the random state. A multilayer perceptron regressor model was created using the MLPRegressor() class function from the sklearn.neural_network python library. A multilayer perceptron (MLP) is a neural network that has distinct layers of neurons. The inputs of the neural network comprise the input layer, and the final outputs comprise the output layer. A 2-hidden layer also known as the intermediate layer comprising 180 neurons for the first hidden layer and 150 neurons for the second hidden layer. An identity activation function was used and the 'lbfgs' solver was to optimize the weights (w_{ij}) of the network. The fit() function was used to fit the training data while the predict() function was used to obtain the predicted values of the target variable (Concerns) for the testing data. The r2_score() function from the sklearn.metrics python library was used to measure the proportion of variance in the target variable (Concerns) explained by the model.

Results and discussions

The results of the Principal Component Analysis (PCA) for the unsupervised learning method shows that the proportion of the variance explained in the AoLR data of the first principal component is 33.46%, while the second principal component explains 21.39% of the variance which gives the total variance explained by the two principal components as 54.85%. The eigenvectors also known as coefficients or loadings of the principal components indicates that each variable represents a principal component, and the numbers in each variable represent the weights (w_{ij}) that each original attribute has in that component. The eigenvectors of the first principal component shows that it is positively

influenced by all the original variables in the AoLR dataset especially by Population, Smokers, Disability, and the Deprivation -2015 features, while the second principal component was shown to be negatively influenced by Student, No central heat, and 65+ age, and positively influenced by HR buildings, Heritage, and Density features. For the eigenvalues, the first principal component has an eigenvalue of 4.69, while the second principal component has an eigenvalue of 2.99. The results shows the proportional amount of the variance in the AoLR data that was explained by each principal component.

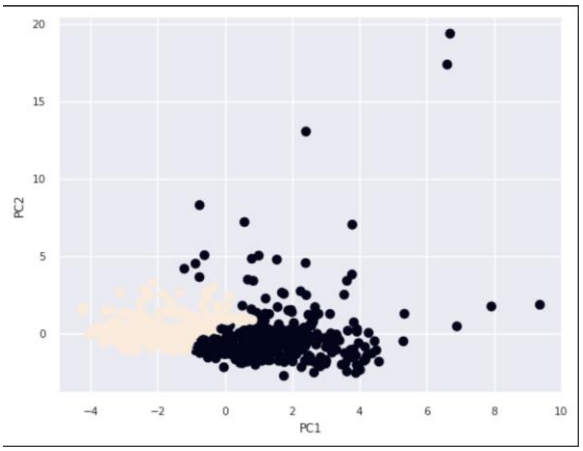

fig. 3: PCA scatter plot of the first 2 principal components.

Fig. 4 above shows a PCA scatter plot that used the agglomerative clustering method to cluster each data point in accordance with their similarity.

The result of the Neural Network model for the supervised learning method which gave an r^2 score of 0.97 indicates that the model fits the data well and explains a high percentage of the variability in the goal variable (Concerns). After attempts of values for the 2-hidden layer architecture, 180 neurons for the first hidden layer and 150 neurons for the second hidden layer gave the highest r^2 score of 0.97. The r^2 score of 0.97 measures the proportion of variance in the target variable (Concerns) explained by the model. This goes to show that the model has a 97% accuracy in predicting the target variable.

For the model assessment of the principal component analysis (PCA) unsupervised learning method which was used to analyze the performance of the PCA on the AoLR data, the results show that the first two principal components explain 33.46% and 21.39% of the total variance in the AoLR data, respectively, gives a total of 54.85% variance of the AoLR data. For further analysis, the first two principal components could be used as they are shown to be the most important variables which can be used to identify patterns and trends in a dataset.

The scatterplot of the eigenvectors shown in fig. 4 below shows the relationship between the different variables in AoLR dataset and points the direction and strength of each principal component.

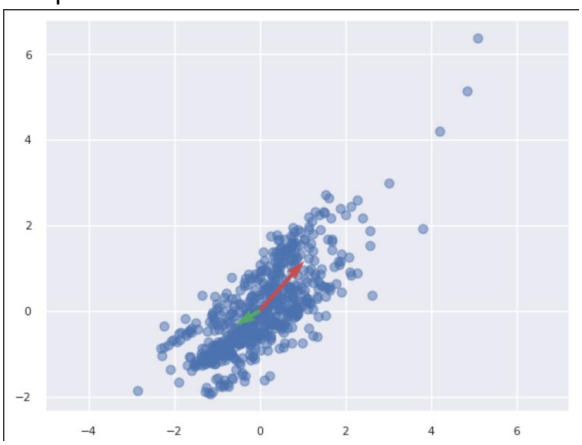

fig. 4: scatterplot of the eigenvectors.

Fig. 5 below shows a bar chart of the eigenvalues with the magnitude of each principal component. It shows that the first principal component explains more of the variance of the data as it has larger values than the second principal component.

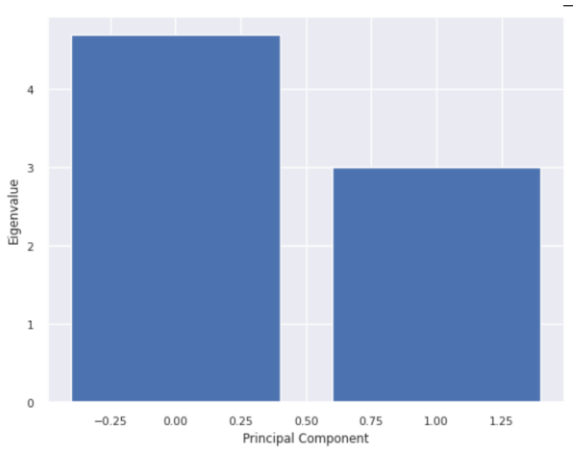

fig. 5: bar chart of the eigenvalues.

The model assessment for the neural network supervised learning model shows to be a good fit for the AoLR data having given an accuracy of 97% to predict the target variable (Concerns). In fig. 6 below, the scatter plot compares the actual and predicted values of the target variable. The red dashed line shows the line of perfect predictions, where the actual and predicted values are same. The overall performance of the model is good even though some values are overpredicting and underpredicting others by the points below and above the red dashed line as seen in the plot.

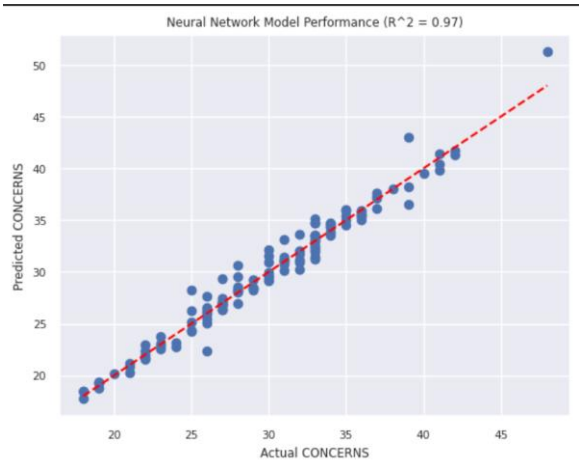

fig. 6: scatter plot of actual vs predicted values.

The PCA unsupervised learning method and the Neural Network supervised learning method were used to identify the patterns and trends in the AoLR 2017 data. The main findings of the PCA model indicates that the first two principal component explained a total variance of 54.85% with the first principal component having a larger value of 33.46% and thereby contributing more than the second principal component with 21.39% while the main findings of the neural network model show a high r^2 score of 0.97, indicating a strong predictive power.

The rationale for the low performance of the PCA model could be the presence of highly correlated variables. The novelty of the analysis is in the combination of PCA and neural network models to gain valuable insights and knowledge into the relationship between the target variable (Concerns) and others and the ability to predict the target variable accurately. The data mining results show that variables like Population, Smokers, Disability, Deprivation - 2015 are significant predictors of the target variable, with Population and Smokers having the highest contributions. The process of data mining involves the dimensionality reduction of the AoLR data using PCA and training a neural network model.

Possible actions and decisions based the findings could include setting up London Fire Brigades offices in wards and boroughs with higher population, promoting healthy lifestyles by discouraging smoking, increasing access to healthcare most especially for people with disability.

Conclusions and perspectives

The study investigated the factors where more concerns were raised which include the population of an area, smokers dominated areas, high disabled individuals' area, students dominated areas, 65+ age individual areas, older people staying alone among others. The study used PCA and neural network models to identify patterns and trends in the data. The

results showed that highly populated areas and high presence of smokers are the highest contributing factors to predict the concerns raised.

The study however has some limitations which include a small sample size of just the 2017 AoLR data which may have affected the accuracy of the results. Further analysis could use a larger sample size for more years.

The previous data used for proposal was changed due to the feedback my tutor, Ayisha Dubi stating that the variables are few and my affect the report. Finding appropriate data from the two websites provided by the module leader, Dr Ingrid Solis, data.gov.uk and data.london.gov.uk was quite challenging for me having searched for weeks to get the AoLR 2017 data. The study cost me two weeks off work and countless sleepless nights and calls with fellow course mates discussing the whole process. Regardless of all the challenges faced, the study provides valuable insights and knowledge into factors influencing where more concerns were raised.

Based on the findings, the study recommends setting up more fire brigades offices in highly populated areas, embarking on health campaigns targeting smokers and addressing the concerns of disabled individuals.

Link to Google Colab project

https://colab.research.google.com/drive/11xWwzEYXy42T06aS06wS-Za8_q1l2R4E?usp=sharing

References

https://www.macrotrends.net/cities/22860/london/population

M. Runefors, N. Johansson, P.V. Hees, “The effectiveness of specific fire prevention measures for different population groups”, Division of Fire Safety Engineering, Lund University, P.O. Box 118, SE-22100, Lund, Sweden (2017)

Assessment of Local Risks supporting the London Safety Plan 2017 - London Datastore

J. Jansson, P. Stenning, “Framing Prosecutor – Police Relations in Europe – A Concept Paper”, The Evolving Role of the Public Prosecutor, pg. 13 (2018)

Z. Zhu & Y. Zhang, “Flood disaster risk assessment based on random forest algorithm”, Neural Computing and Applications 34, 3443-3455 (2022)

B. Lantz, “Machine Learning with R”, Packt Publishing Ltd, Birmingham, UK (2015)

I. M. Solis, “CC7184 Data Mining and Machine Learning – Other unsupervised learning methods: PCA and association rules”, pg. 4 [PowerPoint slides]. Available: Lecture 5 Other unsupervised learning methods: PCA and association rules (blackboardcdn.com)

Appendix

```
# Data mining for unsupervised learning using PCA
# select feature variables
X = AoLR1[['Population', 'Smokers', 'Disability', 'Deprivation -2015',
           'Over crowding', 'Student', 'No central heat', 'crime',
           'local concerns','Density', 'HR buildings', 'Heritage',
           'older alone', '65+ age']]

# standardize data to have zero mean and unit variance
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_std = scaler.fit_transform(X)

# create a PCA instance with 2 principal components
pca = PCA(n_components=2)

# fit the PCA model to the standardized data
pca.fit(X_std)

# transform the data to the new coordinate system defined by the principal
## components
X_pca = pca.transform(X_std)
```

Figure 1: PCA unsupervised learning method

The code performs a dimensionality reduction using the Principal Component Analysis (PCA). The StandardScaler() function was used to standardize the data to have zero mean and unit variance. Two principal components were created which are the two new dimensions the data will be reduced. The fit() function was used to fit the PCA model to the standardized data and the transform() was used to transform the standardized data.

```
# Apply Agglomerative Clustering to the principal components
agg_clustering = AgglomerativeClustering(n_clusters=2, linkage='ward')
agg_labels = agg_clustering.fit_predict(X_pca)

# create a scatter plot of the first 2 principal components colored by cluster
## label
plt.scatter(X_pca[:, 0], X_pca[:, 1], c=agg_labels)
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.show()

# print the explained variance ratio of the principal components
print('Explained Variance Ratio:', pca.explained_variance_ratio_)

# extract eigenvectors and eigenvalues of the principal components from the PCA
## object
eigenvectors = pca.components_
eigenvalues = pca.explained_variance_
```

Figure 2: PCA unsupervised learning method continuation

The AgglomerativeClustering() function was used to cluster the data points according to their similarity. The pca.explained_variance_ratio_ method was used to produce the explained variance ratio of the principal components, which represents the proportion of variance in the AoLR data that is explained by each principal component while the pca.components_ and pca.explained_variance_ methods were used to extract the eigenvectors and eigenvalues of the principal components from the PCA objects respectively.

```
# Data Mining for supervised learning using Neural Network Model
# select the features and target variable
X = AoLR1[['Population', 'Smokers', 'Disability', 'Deprivation -2015',
           'Over crowding', 'Student', 'No central heat', 'crime',
           'local concerns','Density', 'HR buildings', 'Heritage',
           'older alone', '65+ age']]
y = AoLR1['Concerns']

# convert categorical features into dummy variables
X = pd.get_dummies(X, drop_first=True)

# split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
                                                    random_state=42)

# create a neural network model
nn = MLPRegressor(hidden_layer_sizes=(180,150,), activation='identity',
                  solver='lbfgs', random_state=42)
```

Figure 3: Neural Network supervised learning method

The machine learning model was built using a neural network to predict the target variable based on the other attributes represented by the X-variable. The get_dummies() function from the pandas python library was used convert the categorical attributes into dummy variables. The train_test_split() function was used to split the data into training and testing sets. A multilayer perceptron regressor model was created using the MLPRegressor() class function. A 2-hidden layer also known as the intermediate layer comprises 180 neurons for the first hidden layer and 150 neurons for the second hidden layer.

```
# fit the model to the training data
nn.fit(X_train, y_train)

# predict the CONCERNS for the testing data
y_pred = nn.predict(X_test)

# calculate the R^2 score for the predictions
r2 = r2_score(y_test, y_pred)
print('R^2 Score:', r2)

# plot the actual vs predicted values
plt.scatter(y_test, y_pred)
plt.plot([min(y_test), max(y_test)], [min(y_test), max(y_test)], '--',
         color='red')
plt.xlabel('Actual CONCERNS')
plt.ylabel('Predicted CONCERNS')
plt.title('Neural Network Model Performance (R^2 = {:.2f})'.format(r2))
plt.show()
```

Figure 4: Neural Network supervised learning method continuation

The fit() function was used to fit the training data while the predict() function was used to obtain the predicted values of the target variable (Concerns) for the testing data. The r2_score() function was used to measure the proportion of variance in the target variable (Concerns) explained by the model.