P. Sieminski Machine Learning May 2023 Individual Reflection

First Team Project

Concluding reflection

Upon the end of Unit 6, our first collaborative project was due for submission. The task presented was intellectually stimulating. The dataset, publicly provided by Airbnb, contained listing details, which we sought to utilise in answering a pertinent question that could potentially assist Airbnb's executive team. We have asked ourselves: "*Which feature most strongly affects Airbnb's listing price?*"

Collaborating with Rory (my teammate) offered an enriching learning experience. The divergence in our approaches to the task was particularly advantageous, as it allowed us to learn from our unique perspectives.

EDA (Exploratory Data Analysis)

My initial approach involved a comprehensive EDA to unravel the intricacies of the dataset. The resources provided by the University of Essex, including notebooks, proved invaluable in uncovering diverse techniques and significant aspects for the EDA.

The *missingno* Python library was a new discovery for me. It has proven particularly useful when visualising null values in the dataset.

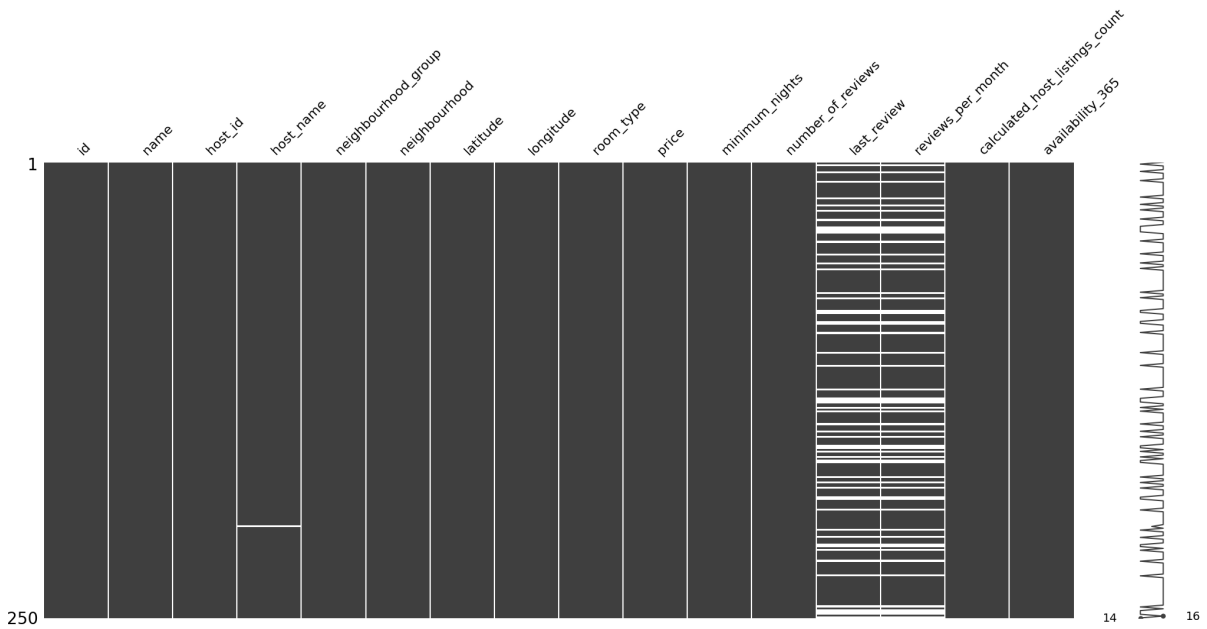

Figure 1: *Missigno*-based chart, visualising null values in the dataset.

ML (Machine Learning) model

Subsequent to the EDA, we embarked on data preprocessing, an area within my professional expertise.

Rory's incremental strategy for selecting the final ML model was mind changing. His progression from simpler models to the ultimately selected XGBoost Regression model was inspirational, particularly in discovering the enhancements realised with the increasing complexity of the algorithms. I, on the other hand, devoted significant time to researching the most suitable model for the task first, eventually selecting XGBoost, with the remaining development time allocated to feature engineering and hyperparameter tuning.

The introduction to SHapley Additive exPlanations (SHAP) analysis and visualisation, courtesy of Rory, was a major learning outcome. It offered the most comprehensive depiction of our model's accomplishments, revealing room type and location as the two key determinants of Airbnb's listing price.

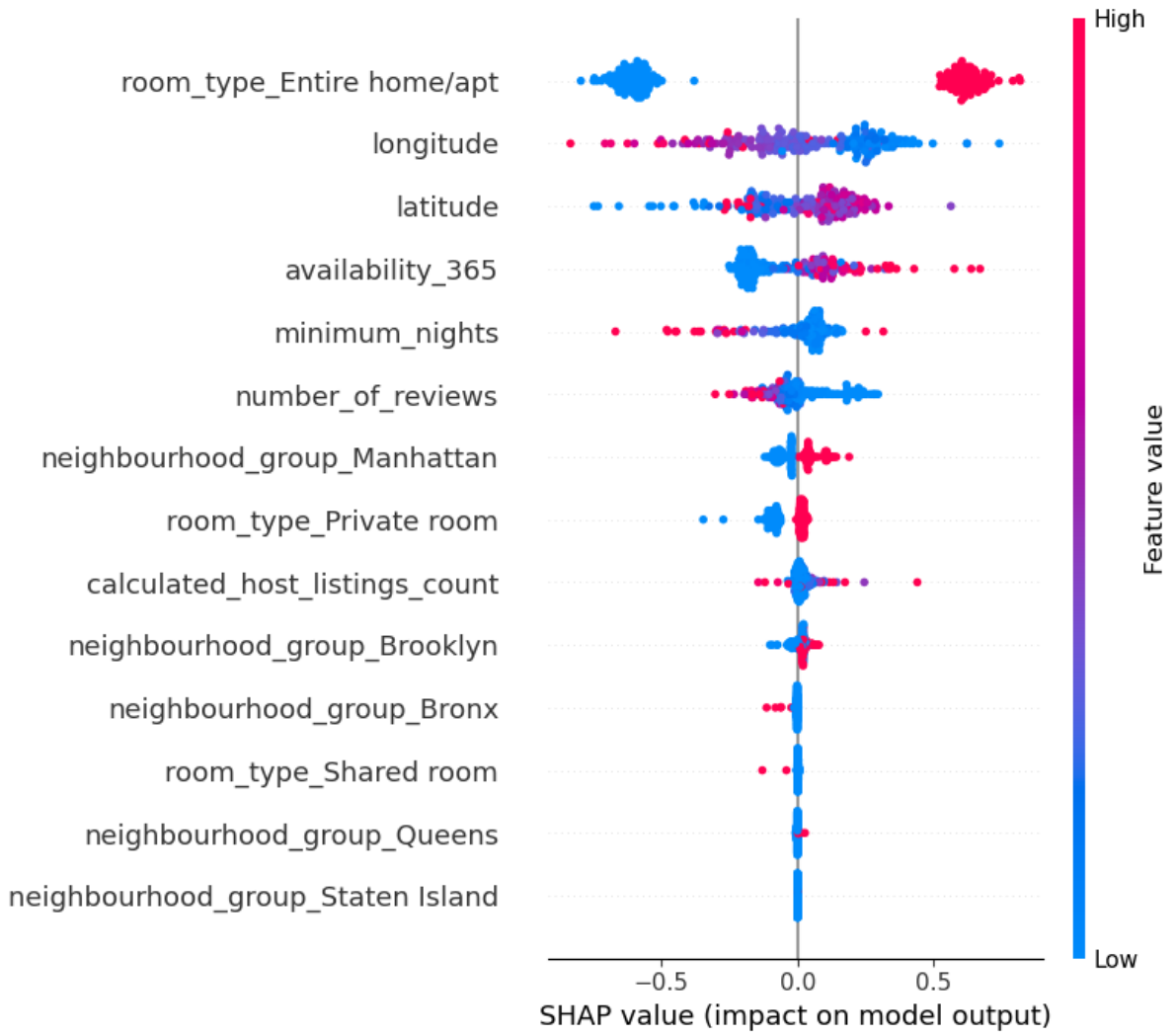

Figure 2: SHAP feature importance analysis of the XGBoost regression model.

This project was a holistic learning experience, enriching my understanding of EDA, visualisation, model selection, feature engineering, and hyperparameter tuning. The acquired transferable skills proved essential in confidently completing this project, which made me feel eager to embark on future ones.

Second Team Project

Concluding reflection

The second project built upon the learnings from the first, significantly shaping my approach to problem-solving.

The project entailed constructing a Neural Network algorithm for the CIFAR-10 dataset (Krizhevsky et al., 2009). An initial attempt with the Artificial Neural Network (ANN) algorithm proved inefficient for this dataset. While research suggested that ANN could achieve 99% accuracy (Real et al.,2019), its advanced nature precluded its application in our case, prompting the switch to a Convolutional Neural Network (CNN) algorithm.

CNN model

CNN's complex resource and time intensity necessitated the use of distributed resources, for which I purchased the Google Colab licence, which enabled the execution of such a memory and computing-intensive model. This sparked contemplation with concluding thought: "*If neural networks aim to replicate the way the human brain is learning, it is astonishing that our heads do not explode when processing such vast amounts of information as we do on a daily basis!*".

The project centred on understanding neural networks (Chollet, 2021) and the strategic application of different layers for optimal learning rate and performance (Garg, 2022).

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 32, 32, 32)        896

 batch_normalization (BatchN  (None, 32, 32, 32)       128
 ormalization)

 conv2d_1 (Conv2D)           (None, 32, 32, 32)        9248

 batch_normalization_1 (Batc  (None, 32, 32, 32)       128
 hNormalization)

 max_pooling2d (MaxPooling2D  (None, 16, 16, 32)       0
 )

 dropout (Dropout)           (None, 16, 16, 32)        0

 conv2d_2 (Conv2D)           (None, 16, 16, 64)        18496
```

Figure 3: Structure of the final CNN model.

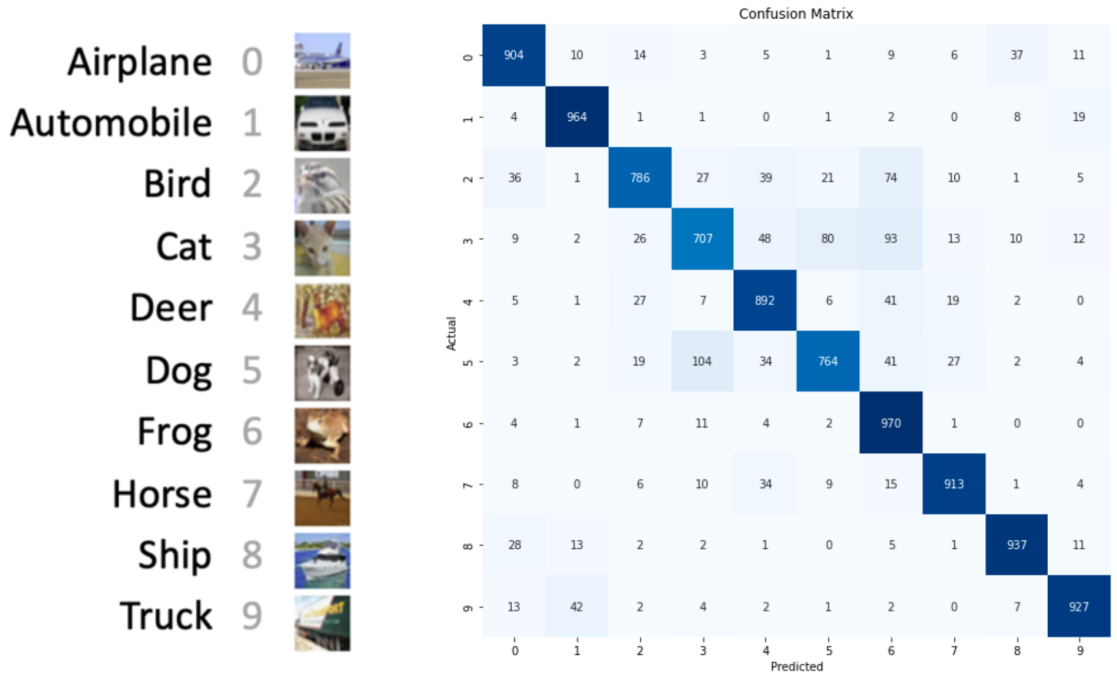

Figure 4: Confusion matrix with referenced classes.

The confusion matrix provided insights into the CNN model's errors, such as misidentifying dogs and cats - a distinction easily made by humans, yet pixels proved to be similar enough to confuse the machine.

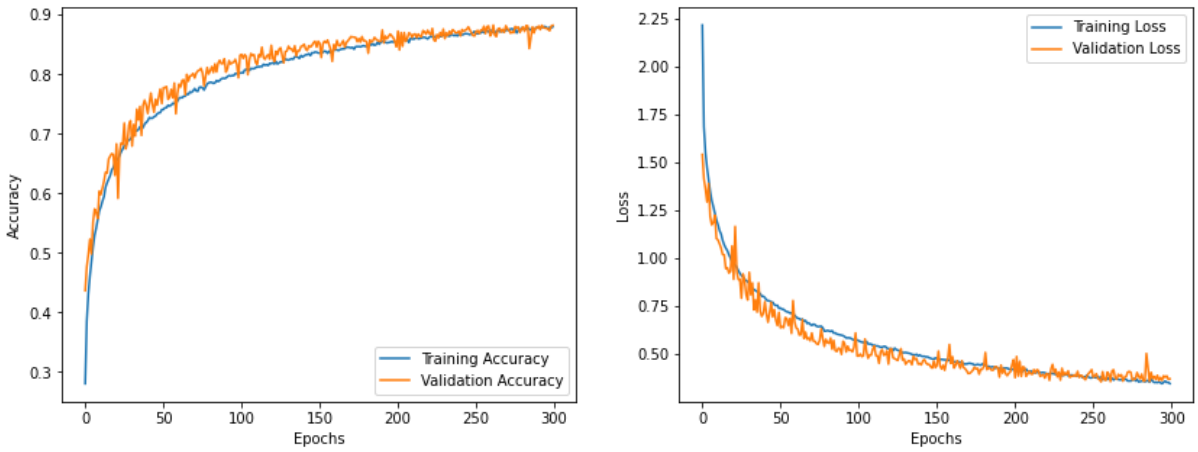

Figure 5: Training and Validation Loss & Accuracy plotted across model's epochs.

Rory and I collectively achieved impressive results, each model improvement reinforcing our enthusiasm to continuously enhance the model.

The adoption of Rory's gradual learning approach from the first project proved beneficial, providing a systematic structure to my work.

Additional Learnings and Adjustments

The ML module encompassed not only the two main projects, but also explored legal, ethical, and security aspects associated with the 4th Industrial Revolution's enhanced use of AI and ML. Other topics included regression, correlation, and associated metrics and techniques, which, while not new to me, are worth mentioning. This reflection, however, focuses on the parts where I gained the most learning and experience.

KMeans clustering

Unit 6's KMeans clustering tasks offered an enjoyable experience. Despite familiarity from professional practice, the tasks' structure enabled exploration of diverse approaches and visualisation techniques. This module provided me with time for a deeper understanding of the concept, a luxury often absent in the professional setting due to project deadlines.

Task A

I have been given a task to perform K-means clustering on a provided set of data (which you can find here)

You can find below the source code for solving this task, as well as two visualisations that help understand the output.

PYTHON

```python
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix, classification_report

df = pd.read_csv('Unit06 iris.csv')

le = LabelEncoder()
df['species'] = le.fit_transform(df['species'])

X = df.drop('species', axis=1)
y = df['species']

kmeans = KMeans(n_clusters=3, random_state=0).fit(X)

print(classification_report(y, kmeans.labels_))
print(confusion_matrix(y, kmeans.labels_))
```

Figure 6: Task A from Unit 6 description and python code.

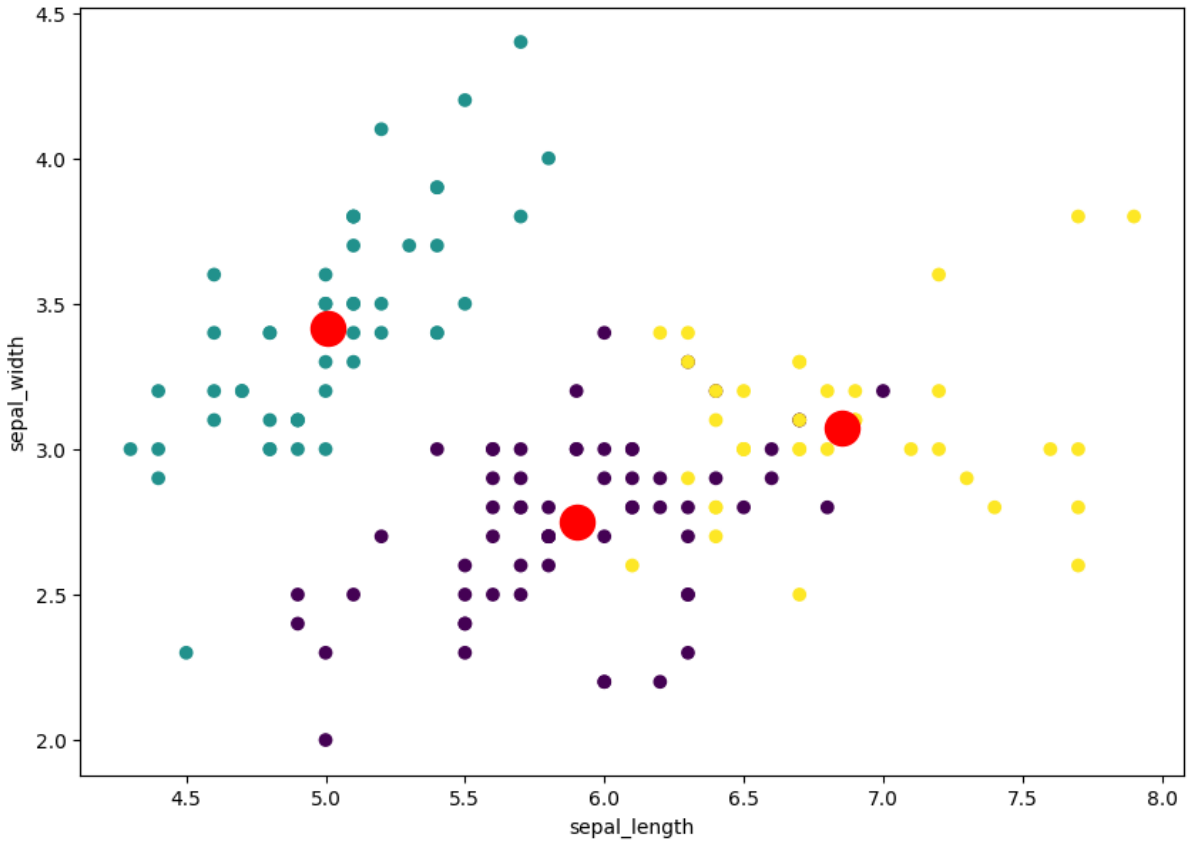

Figure 7: Scatter plot visualising the output of the above algorithm.

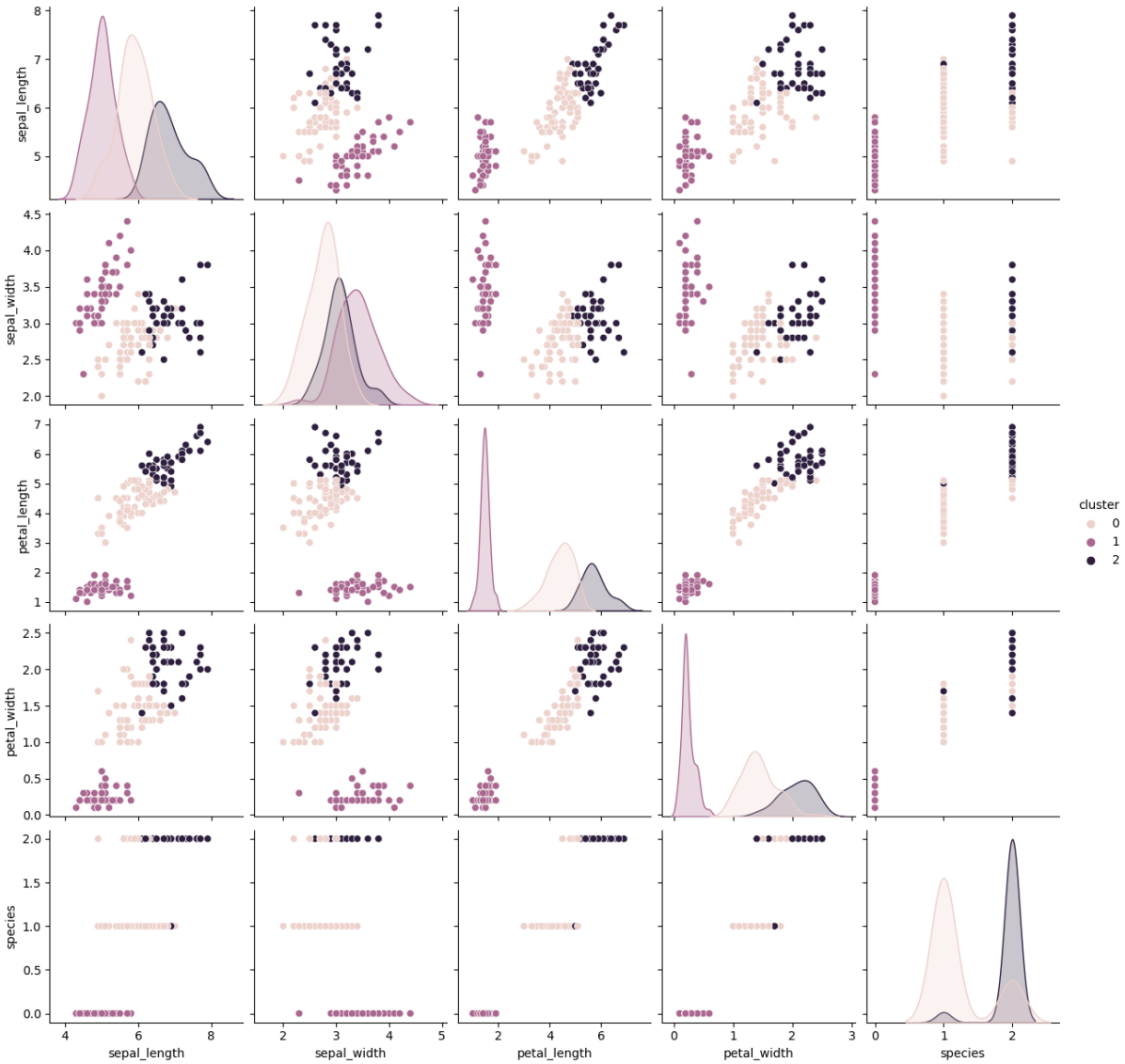

Figure 8: Pair plot visualising clusters across dimensions.

Summary

The Machine Learning Unit offered an opportunity to broaden my knowledge and skills in several areas, some of which were already familiar to me. While I would have liked to enumerate all the acquired learnings, brevity necessitates referring the reader to my comprehensive e-portfolio (referenced below). The collective projects' experiences were invaluable, enriching my knowledge beyond the project requirements and fortifying my professional competencies in the data field, due in large part to the cooperative learning with Rory.

References

1. Sieminski, P. (2023) *Machine Learning*, Available from: https://piotr1204essex.github.io/machine_learning.html (Accessed: 14 July 2023).
2. Inside Airbnb (no date) New York City Airbnb Open Data. Available from: https://www.kaggle.com/datasets/dgomonov/new-york-city-airbnb-open-data (Accessed 8 June 2023).
3. Chen, T. & Guestrin, C. (2016)'XGBoost: A Scalable Tree Boosting System' In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data MiningKDD '16* 785–794. Available from: http://doi.acm.org/10.1145/2939672.2939785.
4. Lundberg, S.M. & Lee, S.-I. (2017) A Unified Approach to Interpreting Model Predictions In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S. and Garnett, R. (eds.) *Advances in Neural Information Processing Systems 30* Curran Associates, Inc.: 4765–4774. Available from: https://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.
5. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, É. (2011) Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (85): 2825–2830.
6. Sieminski, P., Maclean, R. (2023) Which feature most strongly affects the Airbnb listing price? Unpublished manuscript, University Of Essex.
7. Krizhevsky, A., Nair, V., and Hinton, G. (2009). The CIFAR-10 dataset. Canadian Institute for Advanced Research. Available at: https://keras.io/api/datasets/cifar10/ (Accessed: 13 June 2023).
8. Real, E., Aggarwal, A., Huang, Y. and Le, Q.V., (2019). Regularized Evolution for Image Classifier Architecture Search. arXiv preprint arXiv:1802.01548. Available at: https://arxiv.org/abs/1802.01548 (Accessed: 15 June 2023).
9. Madhugiri, D. (2021) Using CNN for image classification on CIFAR-10 Dataset, Medium. Available at: https://devashree-madhugiri.medium.com/using-cnn-for-image-classification-on-cifar-10-dataset-7803d9f3b983 (Accessed: 22 June 2023).
10. Garg, A. (2022) How to classify the images of the CIFAR-10 dataset using CNN?, Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2022/09/how-to-classify-the-images-of-the-cifar-10-dataset-using-cnn/ (Accessed: 22 June 2023).
11. Le, J. (2018) The 4 convolutional neural network models that can classify your fashion images, Medium. Available at: https://towardsdatascience.com/the-4-convolutional-neural-network-models-that-can-classify-your-fashion-images-9fe7f3e5399d (Accessed: 22 June 2023).
12. Kubat, M. (2021) Introduction to machine learning. Cham: Springer.
13. Chollet, F. (2021) 'Getting started with neural networks', in Deep learning with Python. Shelter Island: Manning Publications, pp. 56–92.
14. Implementing a CNN in Tensorflow & Keras (2023) LearnOpenCV. Available at: https://learnopencv.com/implementing-cnn-tensorflow-keras/ (Accessed: 11 July 2023).
15. Chollet, F., 2021. Deep learning with Python. Simon and Schuster.
16. Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.