SPRINGBOARD DATA SCIENCE CAREER TRACK

FINAL CAPSTONE PROJECT

Predicting Home Prices from Zillow Images:

by Sangeeta Jayakar, PhD.

Introduction

The real estate market has seen a great deal of activity in the recent year, especially with places like Austin, TX getting an influx of new residents. Real estate website Zillow saw 9.6 billion visits to its website in 2020, up from the previous year by 1.5 billion. Zillow listings include all pricing and sales history of its properties as well as pictures of the home for sale. For this project I wanted to use the images from the listings to predict price.

Problem Statement

Can machine modeling with input from images taken from Zillow listings be used to accurately predict home prices?

Outline

In this presentation we will discuss:

- Data Wrangling
- EDA
- Pre-Processing
- Neural Network Modeling
- Results
- Conclusions

Data Wrangling

Finding and preparing the data

- Image processing
- Features data cleanup

Dataset was found on Kaggle.com

- **FOLDER OF IMAGES:** INCLUDES THE 1ST IMAGE FOR EACH ZILLOW LISTING IN JPG FORMAT OF DIFFERENT SIZES

- **CSV FILE:** INCLUDES 15,171 ZILLOW LISTINGS WITH 45 FEATURES, ONE KEY COLUMN (ZILLOWID), AND ONE IMAGE NAME WHICH REFERENCES AN IMAGE IN THE IMAGES FOLDER.

IMAGE PROCESSING

Resized

All images were resized to be a uniform size (250, 300, 3)

Grayscale

Resized images were converted to grayscale, (250, 300, 1)

```
[[0.7372549 ]
 [0.70588235]
 [0.76470588]
 ...
 [0.83529412]
 [0.83921569]
 [0.84705882]]

[[0.68235294]
 [0.61568627]
 [0.69411765]
 ...
 [0.83529412]
 [0.83921569]
 [0.84705882]]]
```

Numpy array

A function to convert each image to a normalized Numpy array

HOME FEATURES DATA CLEANUP

Dropped rows

- rows with 'latestPrice' less than 70,000
- rows with 20+ bedrooms/bathrooms
- row with extremely large square feet
- rows with 0 bedrooms or bathrooms

Features scaled

- LivingAreaSqFt scaled dividing by standard deviation
- latestPrice scaled divided by standard deviation

Feature Columns Kept

- livingAreaSqFt
- numOfBedrooms
- numOfBathrooms
- zipcode
- avgSchoolRating
- numOfPrimarySchools
- numOfHighSchools

Datatypes changed

- 'Zipcode' converted from integer to string, treated as categorical

Exploratory Data Analysis

Exploring the target feature

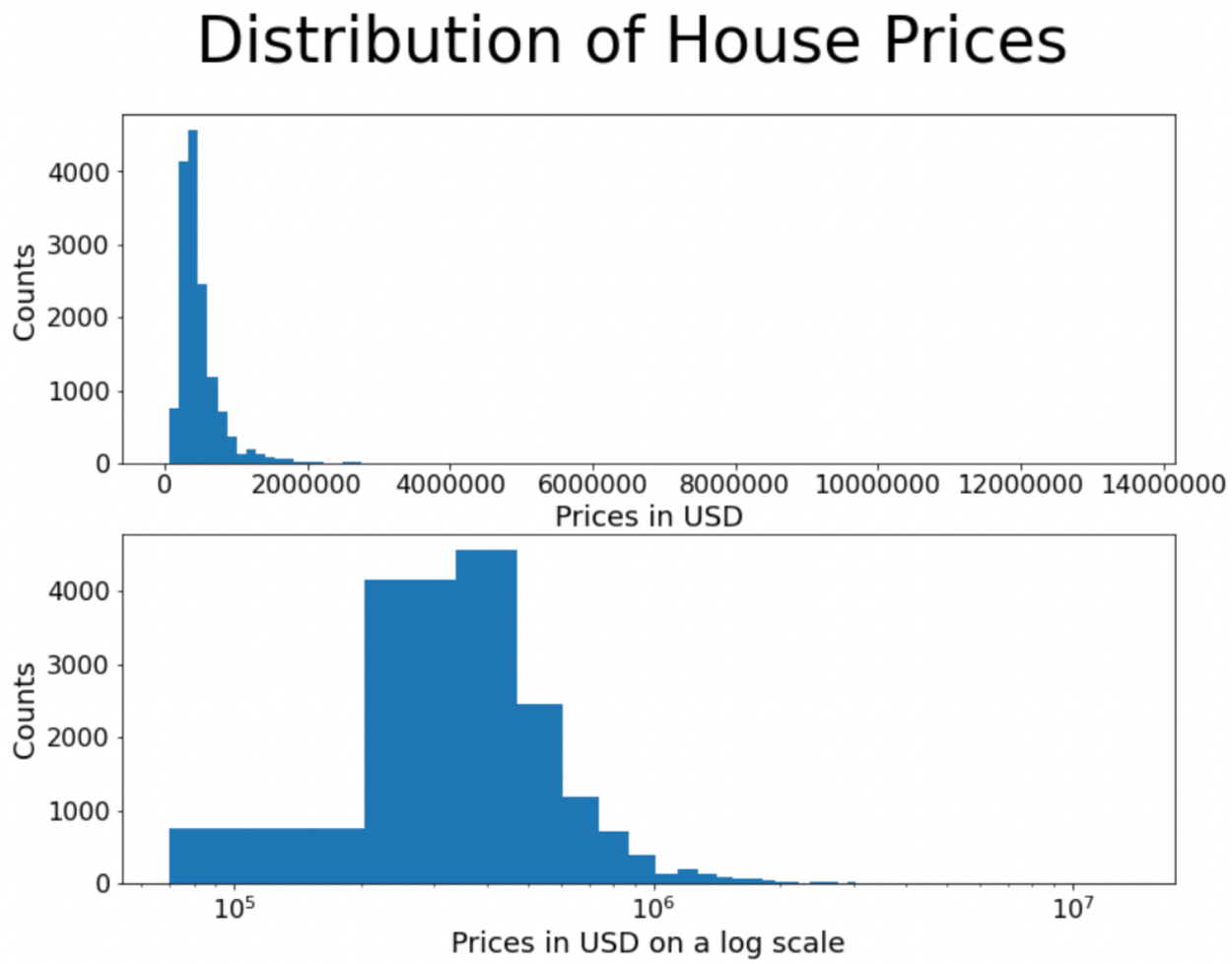
& FEATURES THAT WERE CORRELATED WITH THE TARGET FEATURE:

- living area sq ft
- number of bathrooms
- number of bedrooms
- average school rating
- number of primary schools
- number of high schools

Target Feature

'latestPrice'

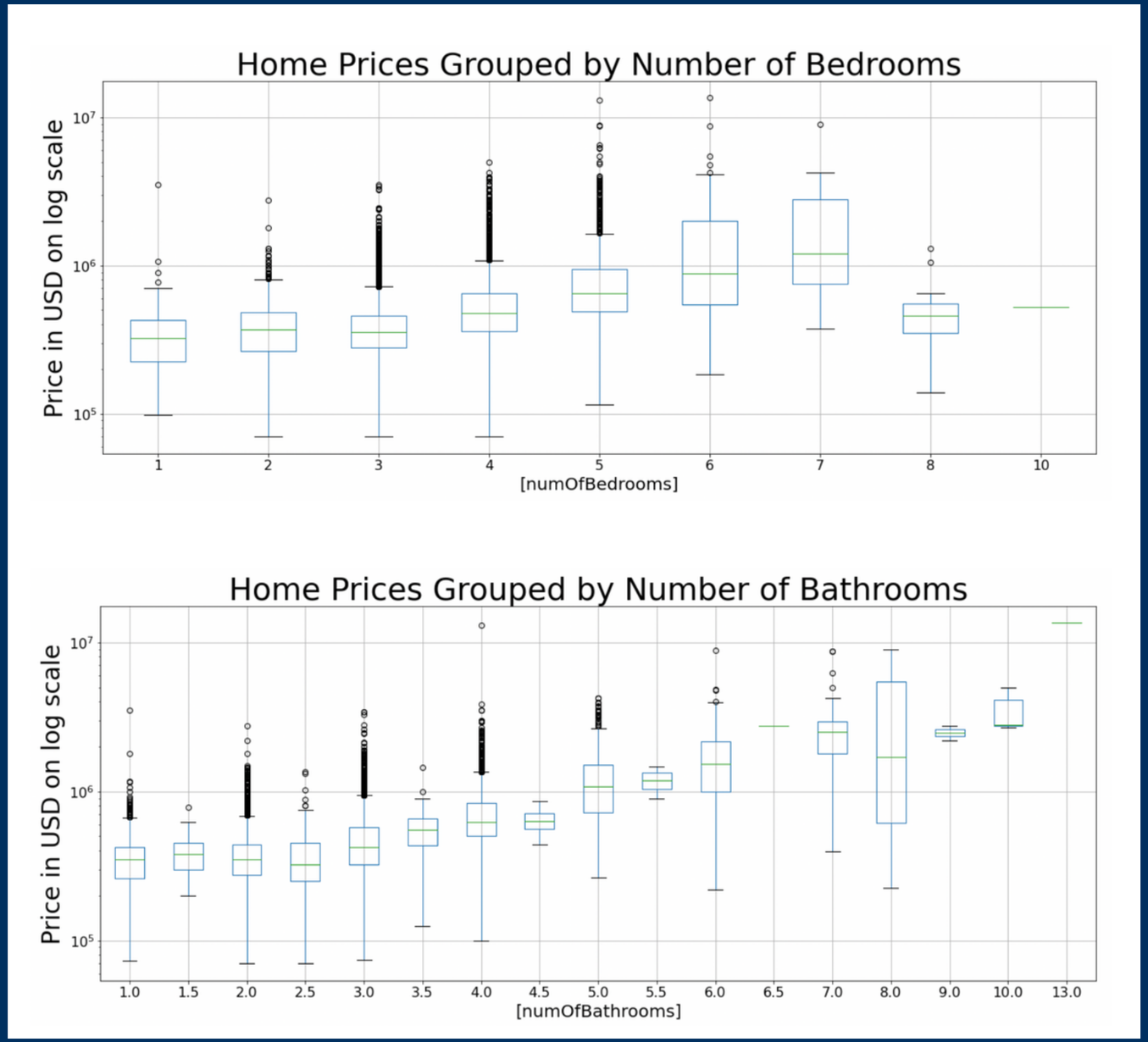
The distribution of the home prices revealed that the mean was about $512K, with the majority of listings under $1M.
The lowest value was $70K
The highest value was $13.5M



Number of Rooms

'numOfBedrooms' & 'numOfBathrooms'

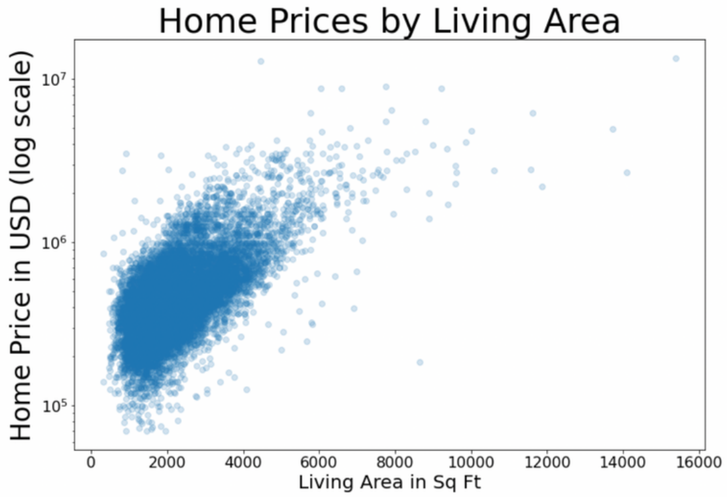
Home prices generally tend to increase as both of these features increase in number. It is more visible in number of bathrooms.

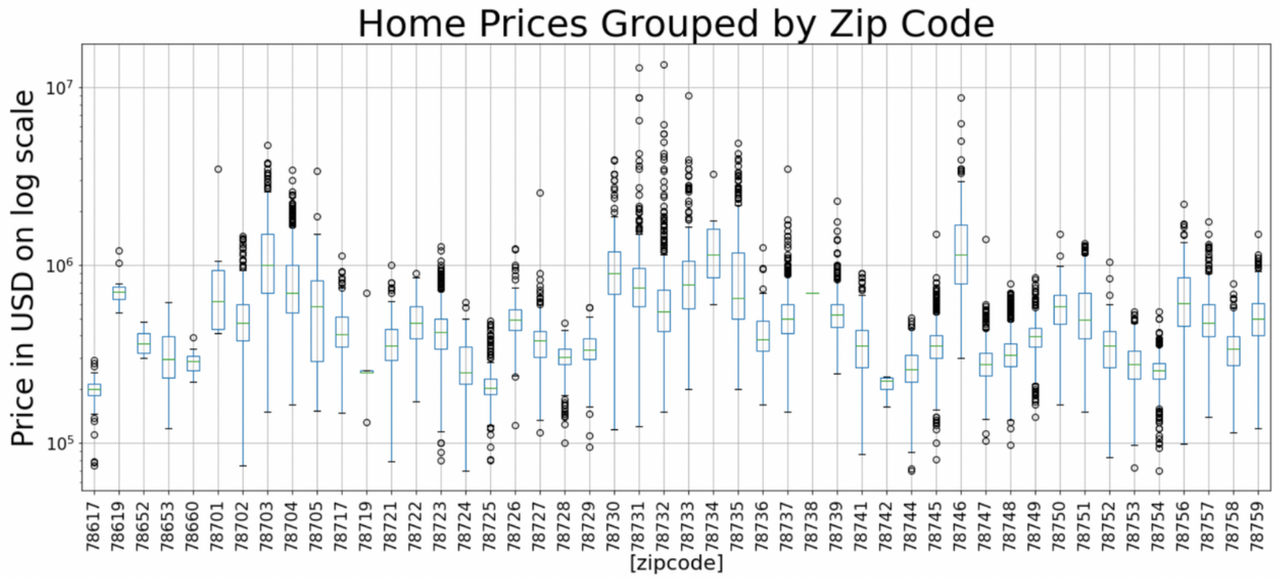


Other Features

Living Area & Zipcode

The overall trend is that as living area increases, so does the price.
Plotting the average home price grouped by each zipcode shows how the price varies across the different areas of Austin.





Pre-Processing & Modeling

Train/Test Split: test size of 0.2

Created the train/test set split such that image data and features data would be split with the same samples in each group.

Data Generator

Created a generator to feed a batch of the data to the model at one time in order to save on RAM required to load the entire set of images

Modeling Strategy

Images only

Tested 1 model:

- Convolutional Neural Network (CNN)

Features only

Tested 2 models:

- Linear regression
- MLP (mixed layer perceptron or Artificial Neural Network)

Images and Features

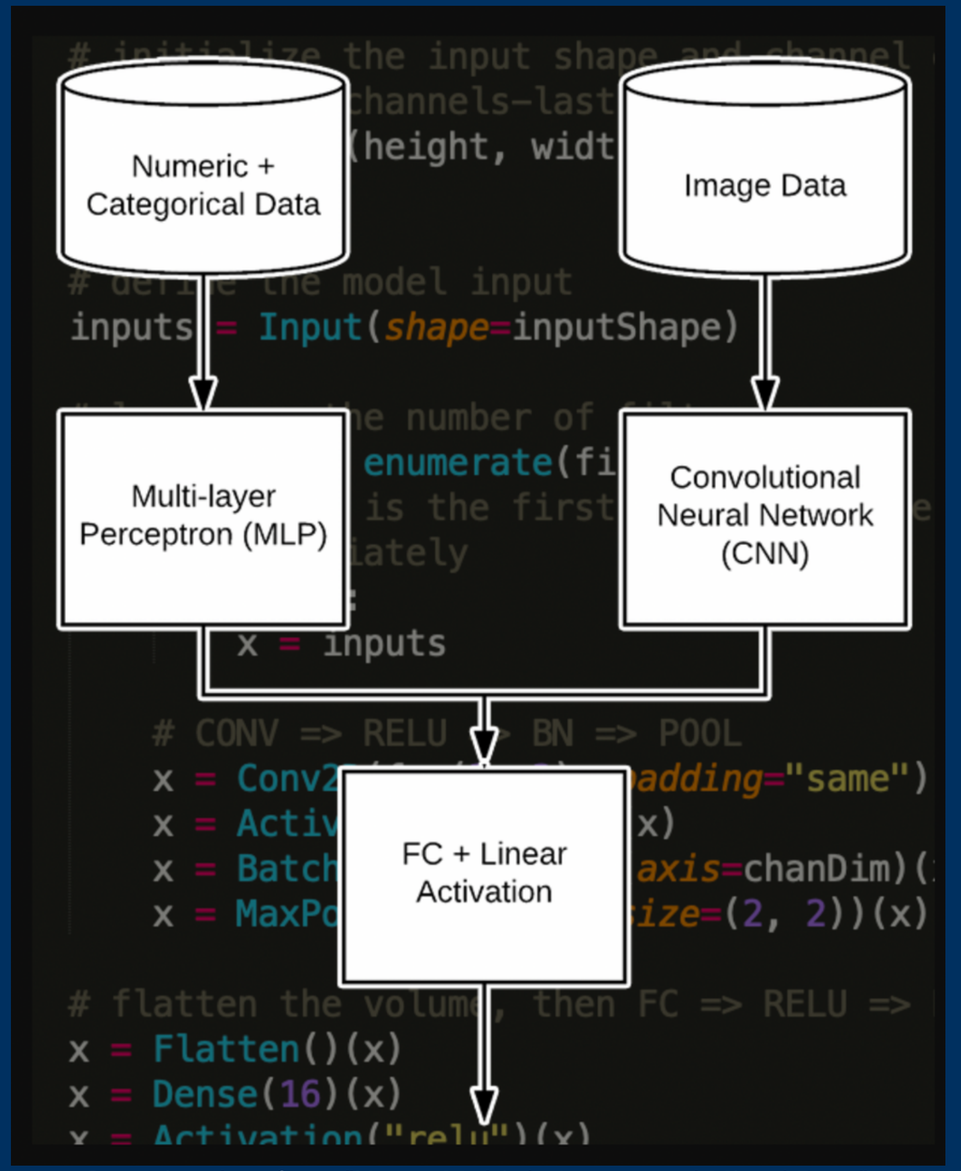
Tested 3 variations of the mixed inputs CNN model:

- normalization of target feature
- more dense layers
- kernel size (3x3) vs (7X7)

Model Architecture

Mixed Inputs Model

This model takes features data into an MLP model, image data into a CNN model, then concatenates the outputs from the two to create the mixed inputs model.



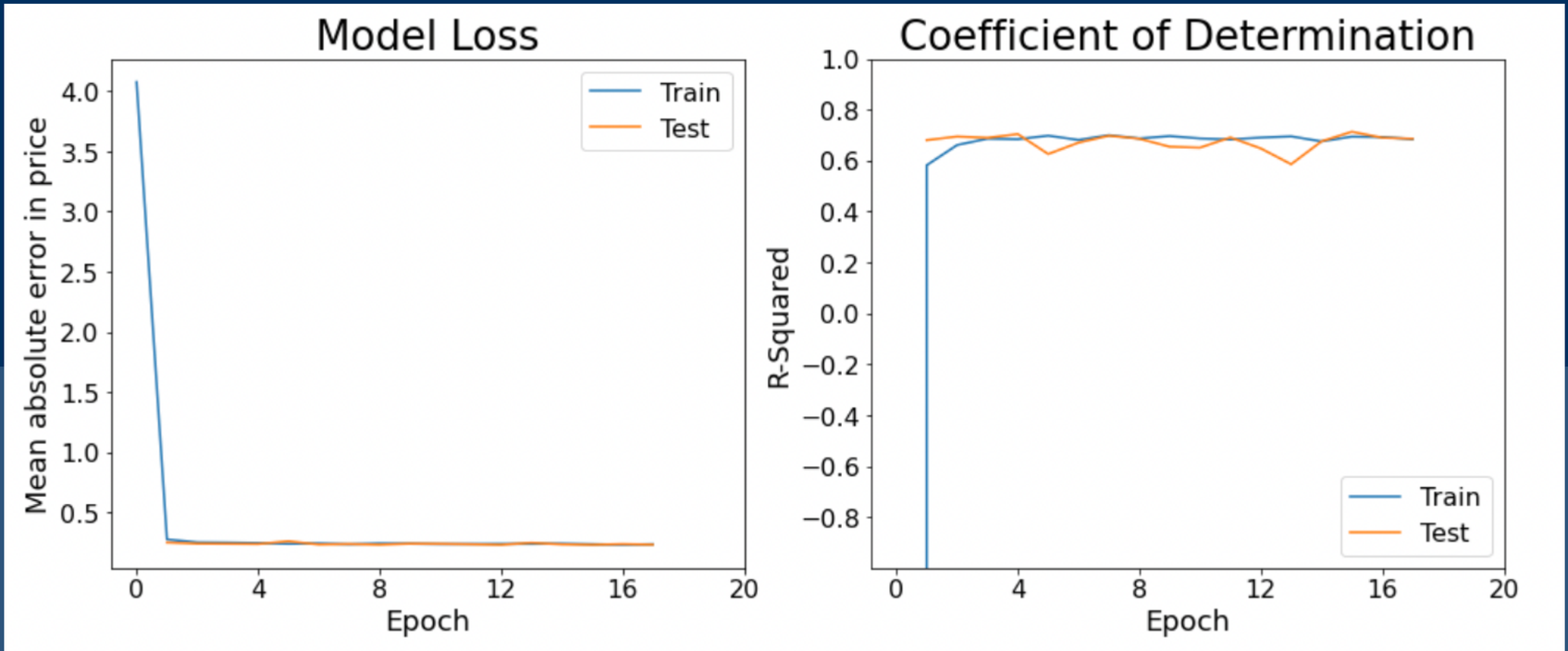
Graphic taken from:"Keras: Multiple Inputs and Mixed Data", https://www.pyimagesearch.com/2019/02/04/keras-multiple-inputs-and-mixed-data/

Results

Highest R-square score achieved for each model tested:

images only:	0.0300
features only:	0.5700
mixed inputs:	0.6644
m.i. + normalized price:	0.7084
m.i. + norm. price + extra dense layer:	**0.7137**
m.i. + norm. price + extra dense layer +increased kernel size:	0.6966
m.i + norm. price + color images:	0.7223

Conclusion



.71

R-Squared Value

- Using the grayscale images and home features into the mixed inputs CNN model provided a better model performance compared to using either method alone.
- Normalizing price and adding an extra dense layer improved the model.
- Color images did not significantly improve the score, and training time was longer

References

Amidi, A. ‘A detalied example of how to use Data Generators with Keras’, <https://stanford.edu/~shervine/blog/keras-how-to-generate-data-on-the-fly>[accessed August 2021].

Brownlee, J. ‘Regression Tutorial with the Keras Deep Learning Library in Python’, MachineLearningMastery.com. < https://machinelearningmastery.com/regression-tutorial-keras-deep-learning-library-python/>[accessed August 2021].

Brownlee, J. ‘How to Visualize Filters and Feature Maps in Convulutional Neural Networks’, MachineLearningMastery.com, <https://machinelearningmastery.com/how-to-visualize-filters-and-feature-maps-in-convolutional-neural-networks/>[accessed August 2021].

‘Implement a Custom Metric Function in Keras’. <https://jmlb.github.io/ml/2017/03/20/CoeffDetermination_CustomMetric4Keras/>[accessed August 2021].

Khandelwal, R. ‘Convultional Neural Network: Feature Map and Filter Visualization,’ Towards Data Science, <https://towardsdatascience.com/convolutional-neural-network-feature-map-and-filter-visualization-f75012a5a49c>[accessed August 2021].

Lewis, L. ‘Building a mixed-data neural network in Keras to predict accident locations’, Heartbeat. <https://heartbeat.fritz.ai/building-a-mixed-data-neural-network-in-keras-to-predict-accident-locations-d51a63b738cf>[accessed August 2021].

Rosebrock, A. ‘Keras: Multiple Inputs and Mixed Data’, pyimagesearch.com <https://www.pyimagesearch.com/2019/02/04/keras-multiple-inputs-and-mixed-data/>accessed August 2021].