OLX Used Car Price Regression Using Neural Network

By: Irfani Sakinah
Bimantara Hanumpraja
R. Achmad Dadang Nur Hidayanto

Github repository: *https://github.com/bangkitjkt/bangkit-1*

Metadata

- Dataset : olxmobilbekas (https://www.kaggle.com/fadhrigabestari/olxmobilbekas)
- Dataset owner : Fadhriga Bestari
- Date created : 2019-11-03
- Columns : Harga, Lokasi, Penjual, Merek, Model, Varian, Tahun, Jarak tempuh, Tipe bahan bakar, Warna, Transmisi, Tipe bodi, Kapasitas mesin, Tipe Penjual, Sistem Penggerak, Nama Bursa Mobil
- Dimension : 14657 rows, 16 columns

Framing

- We want the ML model to predict the suitable price for OLX used cars
- Ideal outcome: the OLX seller will continue to sell their cars in the platform
- Metrics: the number of seller in OLX does not decrease in three months
- Type of output: continuous values (regression)
- Output: predict the suitable price for OLX used cars
- Using the output: we will predict the price of used cars after any seller inputs the description of the car they want to sell. The outcome will give the seller a reference to set the price.

Hypothesis

The price will arise corresponding to high “tahun” value, lower “jarak_tempuh” value, and larger “kapasitas_mesin”.

Preparation

1. Feature selection
   We select several features that we think will deeply influence the price:
   `merek`, `model`, `tahun`, `jarak tempuh`, `tipe bahan bakar`, `warna`, `transmisi`, `kapasitas mesin`, `tipe penjual`, `sistem penggerak`, and `kota`
2. Data Preparation
   - Remove outliers
   - Replace missing values with modus of corresponding model
   - Remove the remaining rows with any missing values
   - Turn ‘jarak tempuh’ and ‘kapasitas mesin’ to numerical data

Techniques

1. Split dataset into train and test examples
2. Scale training and test label by a factor of 1000000
3. Scale the numeric features (jarak_tempuh and kapasitas_mesin) on training and test datasets with minmax_scale function from sklearn.preprocessing
4. Set 1986 as the minimum value at Tahun features and scale it with minmax_scale function
5. Create a list that hold all the features column names
6. Convert the previous created list into layer for the model
7. Model:
    - Define deep neural model, using a linear regression model with tf.keras. models.Sequential()
    - Define a training function
    - Define plotting function

Results

1. Experiment 1 with learning rate 0.001, epoch 100, and batch size 10 generates MSE 5911.7139 against the test dataset using two hidden layers with 20 nodes and 12 nodes respectively.
2. Experiment 2 with learning rate 0.01, epoch 10000, and batch size 1024 give MSE for about 8461.6201 using two hidden layers with 120 nodes and 100 nodes respectively.
3. Experiment 3 with learning rate 0.01, epoch 10000, and batch size 2048 without ‘sistem_penggerak’ features generates MSE 8499 against the test dataset using two hidden layers with 120 nodes and 100 nodes respectively.

Results - cont.

1. Experiment 4 with learning rate 0.1, epoch 2000, and batch size 1024 without ‘sistem_penggerak’ features generates MSE 7685 against the test dataset using two hidden layers with 120 nodes and 100 nodes respectively.
2. Experiment 5 with learning rate 0.01, epoch 200, and batch size 1024 give MSE for about 7404.8062 using two hidden layers with 120 nodes and 100 nodes respectively.
3. Experiment 6 with learning rate 0.1, epoch 2000, and batch size 500 (features consist of: merek, model, tahun, jarak_tempuh, kapasitas_mesin, kota) generates MSE 7061 against the test dataset using two hidden layers with 20 nodes and 12 nodes respectively.

Conclusions

1. Several features in the dataset are not included in the model training because they less significantly affect the car prices.
2. The tuning of hyperparameters are very essential in finding the ideal model, but we should be aware of the overfitting phenomena.
3. There should be more exploration on Neural Net creation, such as number of layers, number of nodes, and regularization.
4. Despite of we use all features to predict car prices, we can conclude that none of them can give insight to the model. It means that the features are probably weak.
5. We have to do features selection to find best features can be used by the model. If the best features is found but still gives bad result, probably the model used is wrong. We need to explore further about another ML algorithm which are best for this problem.

Conclusions

6. We should consider to do cross feature between features that are likely interconnected to see whether they can produce a more meaningful value.
7. There are some car variants which have very few data, therefore their prices are more likely to predict incorrectly. There sholud be more collection of data for each type of car. Another alternative, we also can consider to drop certain types of car with very few data.