House price prediction – ML Project Assignment

Name: Joseph Chuang-Chieh Lin
Date: 24 September 2019

Assignment: House Price Prediction

In this assignment, you are asked to build a regression model to predict house prices. The dataset contains house sale prices for Taipei City. The data is from http://lvr.land.moi.gov.tw/ and simplified by removing some columns and irrelevant samples.

Metric

Your model will be evaluated based on the Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the actual sale prices. (Taking logs means that errors in predicting expensive houses and cheap houses will affect the result equally.)

About data

Definition of categories are in the 'category_map.txt' file. The dates corresponding to 1900-01-01 are because the original dataset had some entries missing the date.

Development Environment

Operating system and hardware

- **OS: Fedora 29 (Linux kernel: 5.2.7)**
- **Device: Asus X301A Notebook (bought in July 2013)**
- **Memory: 4GB**
- **CPU: Intel Pentium B980 2.4GHz (2 Cores)**

Packages and Environment Construction

- **Install Anaconda 3**

- **conda install numpy, pandas, scikit-learn, joblib, matplotlib.**
  - **Scikit-learn (sklearn): library of machine learning models.**
  - **joblib: save and load trained models**

- **Create an environment by "conda create -n ML_TASKS python=3.5"**

Raw Data

Raw data fields:

主要建材_cat	主要用途_cat	交易年月日_date	備註	全移轉_bin	單價每平方公尺
土地_num	土地移轉總面積平方公尺	建物_num	建物型態_cat	建物現況格局-廳	建物現況格局-房
建物現況格局-衛	建物現況格局-隔間_bin	建物移轉總面積平方公尺	建築完成年月_date	有無管理組織_bin	移轉層次_num
總價元	總樓層數_num	車位_num	車位移轉總面積平方公尺	車位總價元	車位類別_cat
都市土地使用分區_cat	鄉鎮市區_cat				

Data Preprocessing (1/3)

Clean the raw data

- **removing entries with ' 單價每平方公尺 '== 0**

Creating more features as below:

- **time-diff:** 交易年月日 **_date** – 建築完成年月 **_date (integer; in days)**
- **pre-sold:** 備註 **contains** 「預售」 **(binary)**
- **relatives:** 備註 **contains** 「特殊關係 | 二親等 | 二等親 | 母女 | 姊弟」 **(binary)**
- **additional_construction:** 備註 **contains** 「增建」 **(binary)**
- **transfer_floors_count: the number of floors were sold (integer)**
- **IS_UNLUCKY: the sold floors containing 4th or 10th floor or not (binary)**
- **IS_LOWER: the sold floors are lower than 4th or not (binary)**
- **mean_space:** 建物移轉總面積平方公尺 /(建物現況格局 - 廳 + 建物現況格局 - 房 + 建物現況格局 - 衛)

Data Preprocessing (2/3)

Numerical features:

' 全移轉 _bin', 'time_diff', ' 土地 _num', ' 土地移轉總面積平方公尺 ', ' 建物 _num', ' 建物現況格局 - 廳 ', ' 建物現況格局 - 房 ', ' 建物現況格局 - 衛 ', ' 建物現況格局 - 隔間 _bin', ' 建物移轉總面積平方公尺 ', ' 有無管理組織 _bin', 'transfer_floors_count', ' 車位 _num', ' 車位移轉總面積平方公尺 ', 'IS_UNLUCKY', 'IS_LOWER', 'mean_space', 'pre-sold', 'relatives', 'additional_construction'

Categorical features:

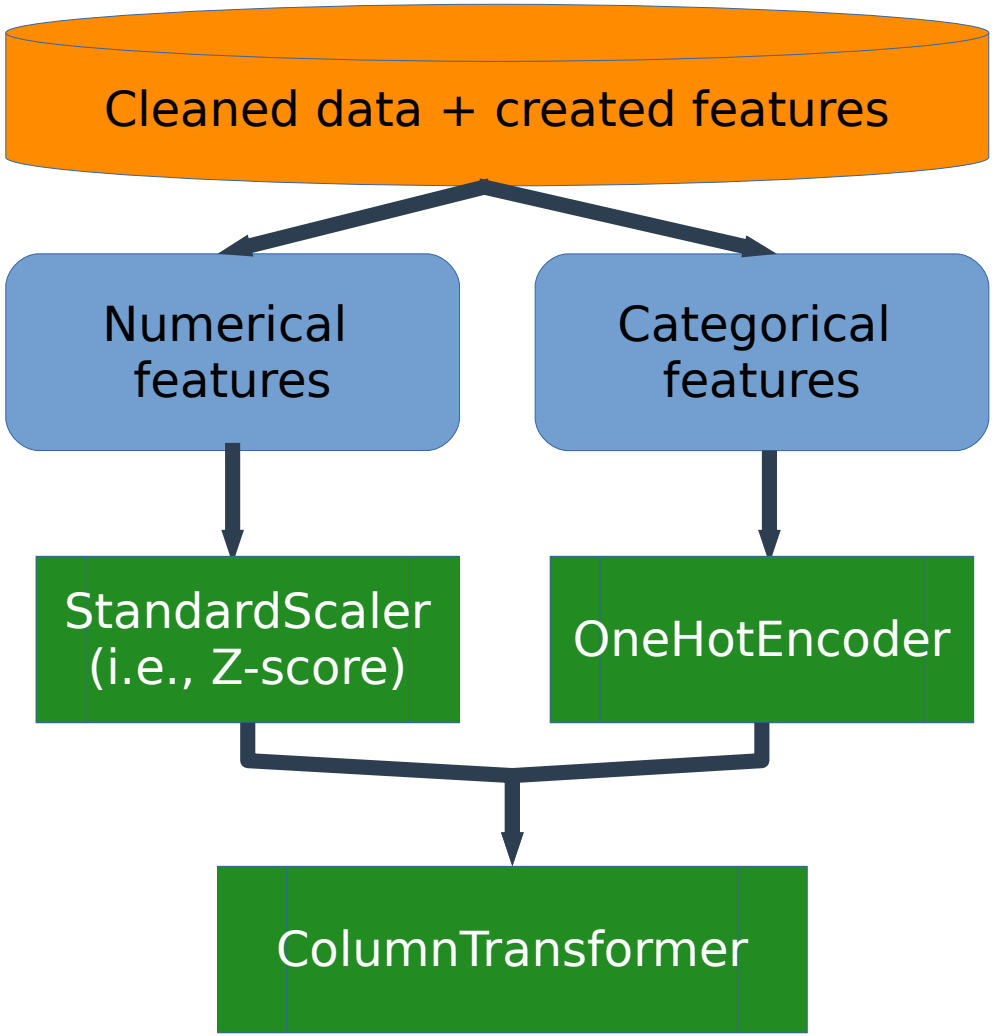
' 主要建材 _cat', ' 主要用途 _cat', ' 建物型態 _cat', ' 車位類別 _cat', ' 都市土地使用分區 _cat', ' 鄉鎮市區 _cat'

Prices:

有車位 → **log(' 總價元 ' + ' 車位總價元 ')**

無車位 → **log(' 總價元 ')**

Data Preprocessing (3/3)

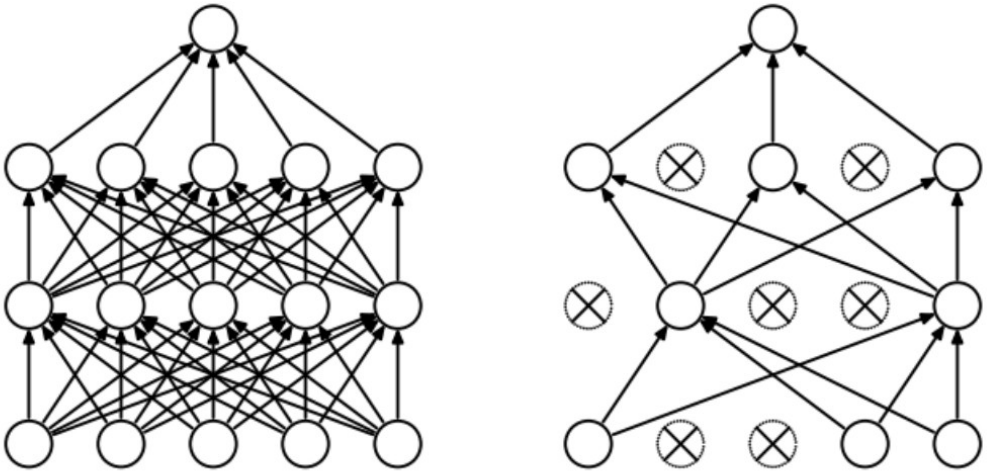


Applied ML Model: multilayer Perceptron

MLPRegressor: A basic deep neural network regression

Use a revised version of it on **https://tinyurl.com/y2udxyvo**
in which **dropout** is added.

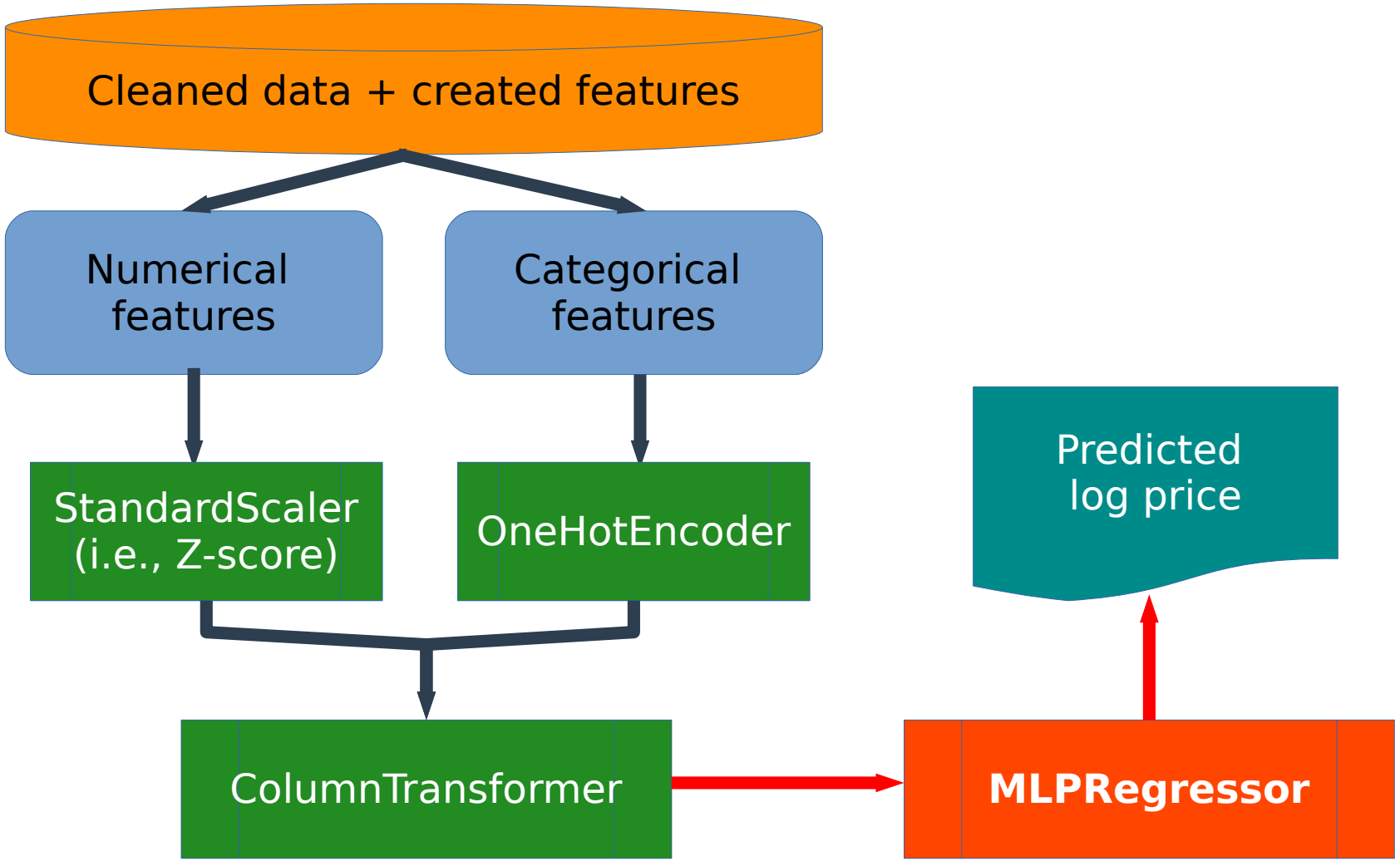
→ replace the one in
<environment_path>/sklearn/neural_network/multilayer_perceptron.py



(a) Standard Neural Net

(b) After applying dropout.

Data Preprocessing + flowchart



MLPRegressor Parameters

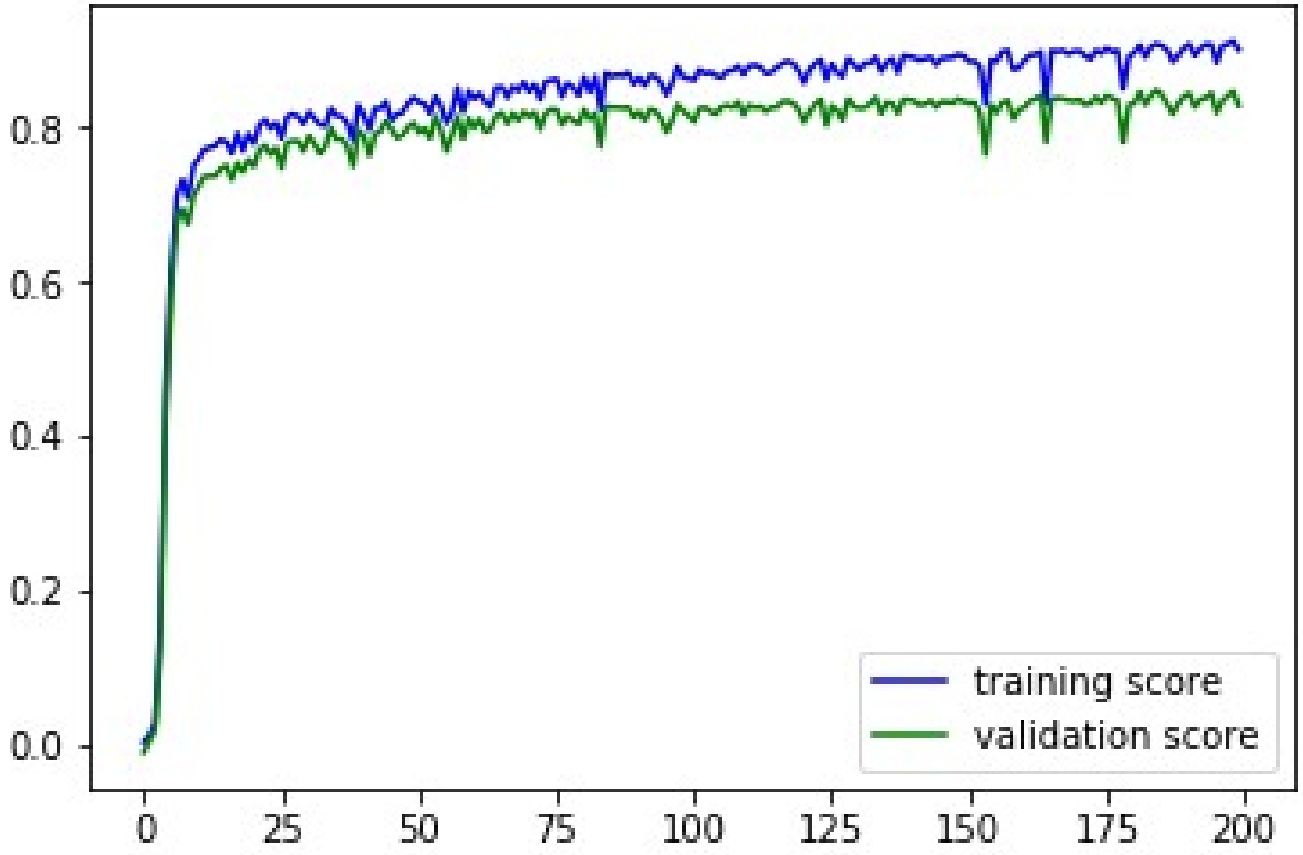
Program: **timeline_mlp_train.py**

Parameter	Value
Hidden layers	4
Activation	tanh
Neurons per layer	128
Epochs to stop if no improving of validation	32
Learning algorithm	Adam
Initial learning rate	0.001
Learning criteria	adaptive

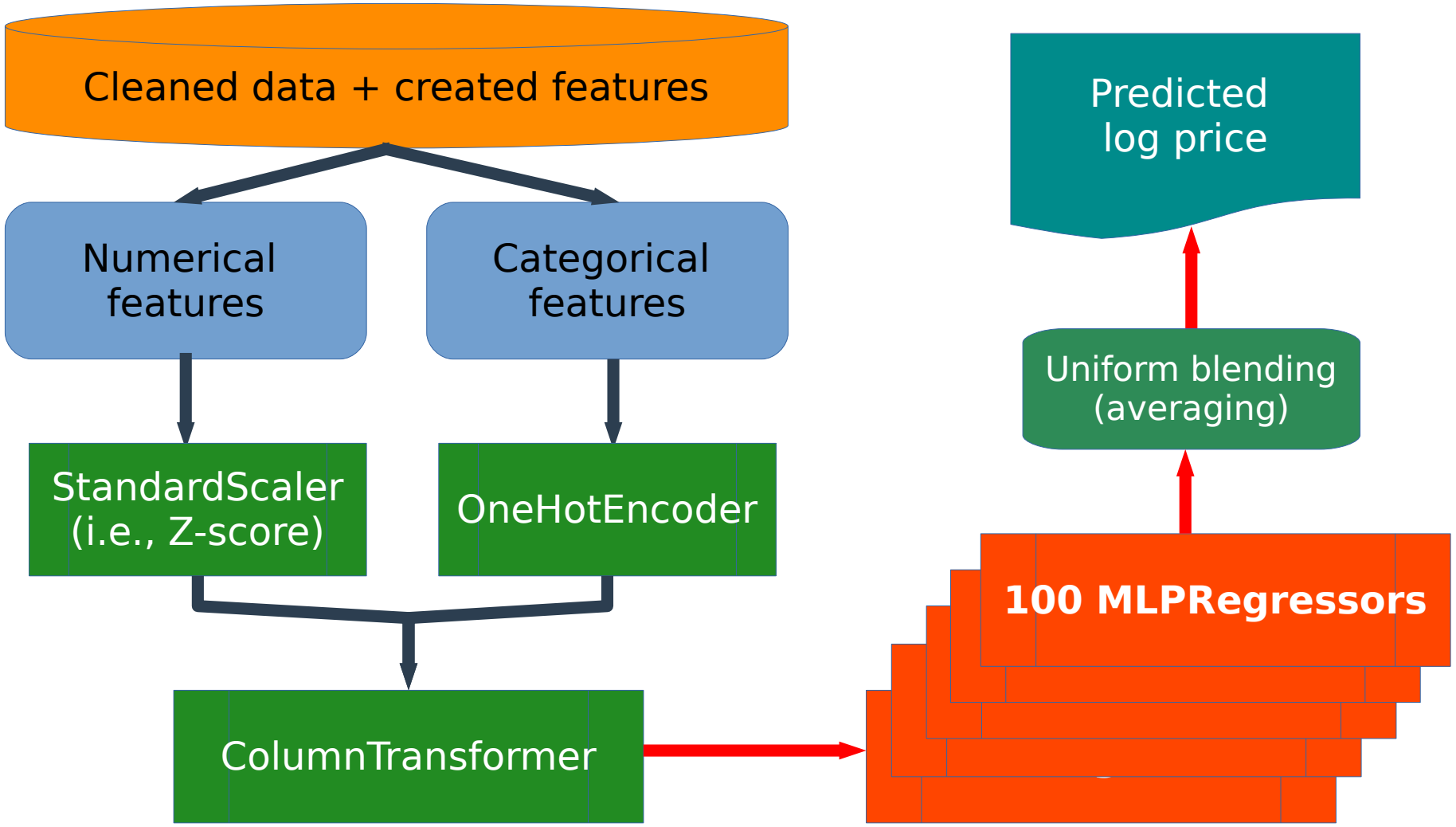
Parameter	Value
Dropout ratio	0.1
Regularization	L2
Weight decay	0.0001
Validation size	0.1
Batch size	128
Max epochs	300
Early stopping	True

Snapshot of Training and Validation

Scores: R^2 score defined as 1 − u/v, where u is the residual sum of squared errors (i.e., predicted log price vs. true log price) and v is the total sum of squared differences between true log price and the mean log price.



Enhancing the regression by uniform blending



Out sample testing result (RMSE)

Program: timeline_mlp_robust_test.py

Out sample data: 0.1 fraction of the preprocessed data

Out sample	RMSE (training+validation set)	RMSE (out sample)
車位_num ≥ 0	0.2645	0.2954
車位_num > 0	0.2439	0.2602

Thanks!