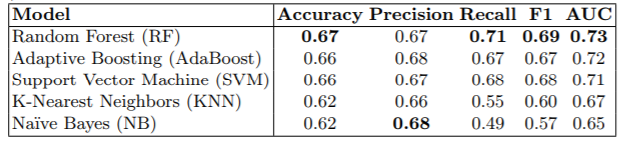
**Regression problem on**

**Online news popularity**

**Data introduction**

This dataset was extracted from UCI machine learning repository, originally acquired by Mashable in the period of 2 years, from January 7, 2013 to January 7 2015. The initial goal of this dataset collected was used in the predication of online news popularity for a new system, Intelligent Decision Support System (IDSS). In the original paper, the dataset was predicated as a classification question with the best evaluation in bold as below.



However, in this project, we predict the total shares as a regression problem with this dataset.

In this dataset, there are total of 39,797 records with 61 attributes. These attributes contain 58 predictive attributes, 2 non-predictive attributes and 1 goal field (shares: Number of shares (target)). Dataset does not contain any missing values. Supervise learning technique is used to approach this problem, since target was given in this dataset.

Further details of each attributes reference can be found in attached [*OnlineNewsPopularity.names*](OnlineNewsPopularity.names) document.

**Data cleaning**

On preliminary inspection, retraining only predictive attributes, dataset will still give a relatively large attributes space. Using a linear regression as a baseline, prediction produced undesired result which is caused by substantial number of attributes in dataset. To improve model ability to generalize, there is a need to reduce the number of attributes in the dataset. The following attributes are removed as they do not contribute relevantly to the training of the neural network model.

Reference material on why these attributes are removed can be found in[*online news popularity.html*](ref%20materials/online%20news%20popularity.html)file.

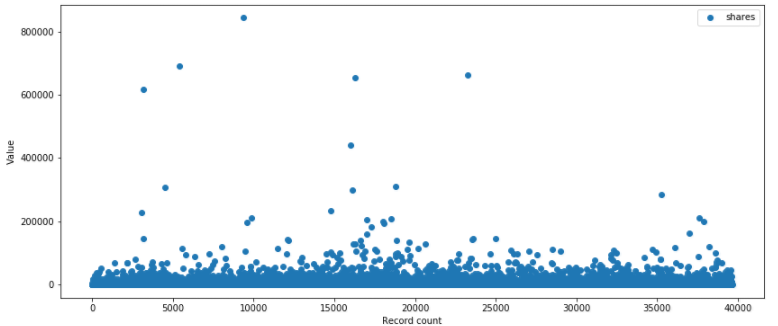
Irrelevant variables are omitted as follows:

|  |  |
| --- | --- |
| **Attributes** | **Remarks** |
| **url** | URL of the article. (non-predictive) |
| **timedelta** | Days between the article publication and the dataset acquisition. (non-predictive) |
| **LDA\_01**  **LDA\_02**  **LDA\_03**  **LDA\_04**  **LDA\_05** | Latent Dirichlet allocation variables, derive field from URL on topics. |
| **is\_weekend** | Since it seems to be duplicating days of week. |
| **kw\_min\_min**  **kw\_avg\_min**  **kw\_min\_avg** | Contain mainly of zero or negative values, incomplete dataset. |
| **Weekday\_is\_monday**  **weekday\_is\_tuesday**  **weekday\_is\_wednesday**  **weekday\_is\_thursday**  **weekday\_is\_friday**  **weekday\_is\_saturday**  **weekday\_is\_sunday** | Binary fields will be converted to polytomous variable to reduce the numbers of attributes in dataset.  A polytomous variable is more prefer than multiple binary field in a regression problem as it reduce the model complexity as less initial variable is included. |
| **data\_channel\_is\_lifestyle**  **data\_channel\_is\_entertainment**  **data\_channel\_is\_bus**  **data\_channel\_is\_socmed**  **data\_channel\_is\_tech**  **data\_channel\_is\_world** | Binary fields, will be converted to categorical variable to reduce the numbers of attributes in dataset. |

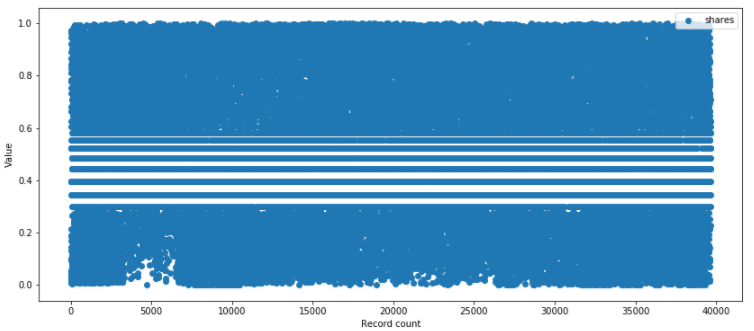
After omission of irrelevant attributes, there are 38 predictive attributes and 1 goal field remaining in the dataset. The remainder attributes used for training are as follows:

|  |  |
| --- | --- |
| **Retain attributes** | |
| n\_tokens\_title  n\_tokens\_content  n\_unique\_tokens  n\_non\_stop\_words  n\_non\_stop\_unique\_tokens  num\_hrefs  num\_self\_hrefs  num\_imgs  average\_token\_length  num\_keywords  kw\_max\_min  kw\_min\_max  kw\_max\_max  kw\_avg\_max  kw\_max\_avg  kw\_avg\_avg  self\_reference\_min\_shares  self\_reference\_max\_shares  self\_reference\_avg\_sharess | global\_subjectivity  global\_sentiment\_polarity  global\_rate\_positive\_words  global\_rate\_negative\_words  rate\_positive\_words  rate\_negative\_words  avg\_positive\_polarity  min\_positive\_polarity  max\_positive\_polarity  avg\_negative\_polarity  min\_negative\_polarity  max\_negative\_polarity  title\_subjectivity  title\_sentiment\_polarity  abs\_title\_subjectivity  abs\_title\_sentiment\_polarity  weekday  data\_channel  Shares (Target) |

The objective of this project is to estimate the exact shares of a given article. When visualizing shares, there are many outliers in this dataset. These outliers are removed by using Quantile Transformer. Upon transformation, the resulting shares will fall within 0 and 1.



**Before normalization**



**After normalization**

Training dataset is split into 80% training set 20% test set and seeded shuttle is applied when generating these datasets. Summary of training and testing dataset looked as follows.

Dataset shape: (39644, 38), Labels: (39644, 1)

x\_train: (31715, 38), y\_train: (31715, 1)

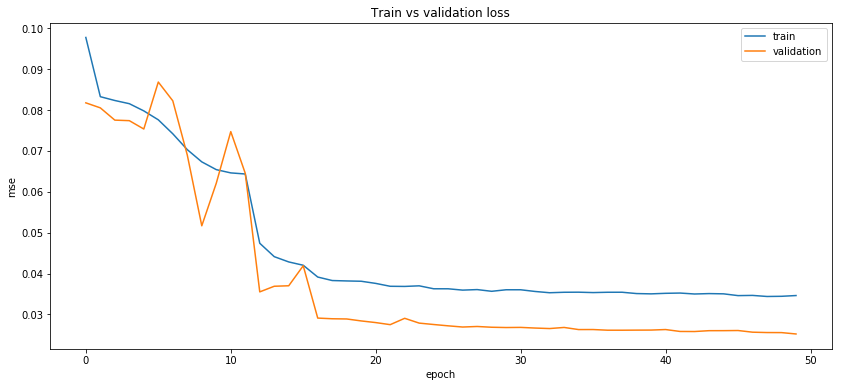
x\_test: (7929, 38), y\_test: (7929, 1)

[**Multilayer Perceptron**](regression_mlp%20-%20shares.ipynb)

Layer (type) Output Shape Param #   
=================================================================  
dense\_11 (Dense) (None, 512) 20480   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_9 (Dropout) (None, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_12 (Dense) (None, 256) 131328   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_10 (Dropout) (None, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_13 (Dense) (None, 256) 65792   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_11 (Dropout) (None, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_14 (Dense) (None, 128) 32896   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_12 (Dropout) (None, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_15 (Dense) (None, 1) 129   
=================================================================  
Total params: 250,625  
Trainable params: 250,625  
Non-trainable params: 0  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

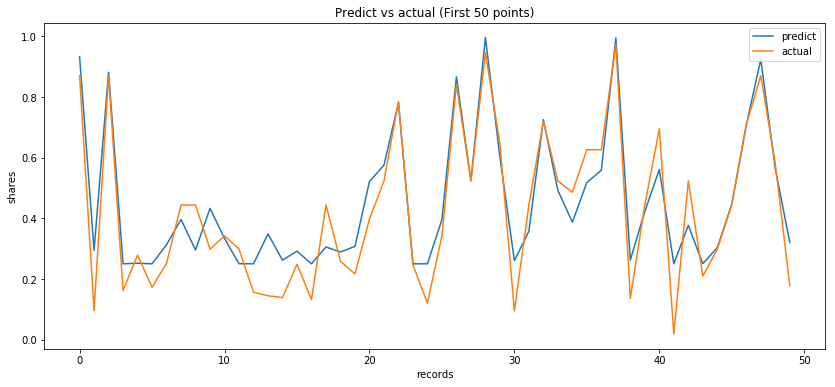
MLP model consists of 5 layers, with each layer being a fully connected layer. To prevent vanishing gradients problem, hidden layers use ReLu for as the activation model. In addition, to avoid overfitting during training, L2 regularization and dropout are used. This model uses mean square error to calculate losses while using RMSprop as optimizer. In summary, model will accept the 39 attributes (after pre-data preparation) as input and output a single number between 0 and 1. All nodes in model are trainable.

**Evaluating training results**

****

At a glance, validation loss is lower than training loss which is very unusual. This is cause by keras framework where dropout layers is applied during training phase and not during validation phase. Therefore training loss is higher than validation loss.

Train and validate loss before 5th epoch shows signs of under fitting. After 10 epoch, loss started to plateau and shows sign of converge for this model. After sign of convergence further training does not show any of improvement, gaps in between indicates model does not over fit.

****

MLP model shows predicted value fluctuating closely to actual value. But this model has an issue as it fails to predict value lower than 0.2.

[**Radial Basis Function Network**](regression_rbf%20-%20shares.ipynb)

RBF network is an architecture that has fully connected input layer to a single hidden layer. This hidden layer is then fully connected to the output layer, with each hidden node providing a radial basis function of input variables.

In order to improve RBF network performance, a simple MLP network is used to further extract important attributes from the dataset. This extraction network will output 20 nodes that will act as input for the RBF network.

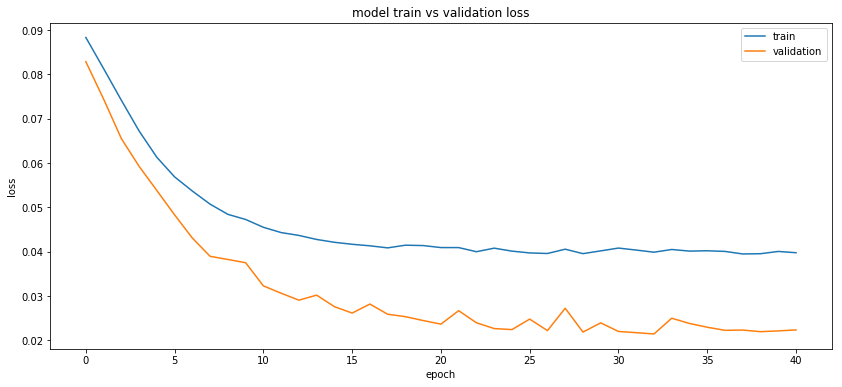
Extraction MLP network before slicing

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Layer (type) Output Shape Param #   
=================================================================  
dense\_17 (Dense) (None, 128) 5120   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_9 (Dropout) (None, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_18 (Dense) (None, 20) 2580   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_10 (Dropout) (None, 20) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_19 (Dense) (None, 1) 21   
=================================================================  
Total params: 7,721  
Trainable params: 7,721  
Non-trainable params: 0  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

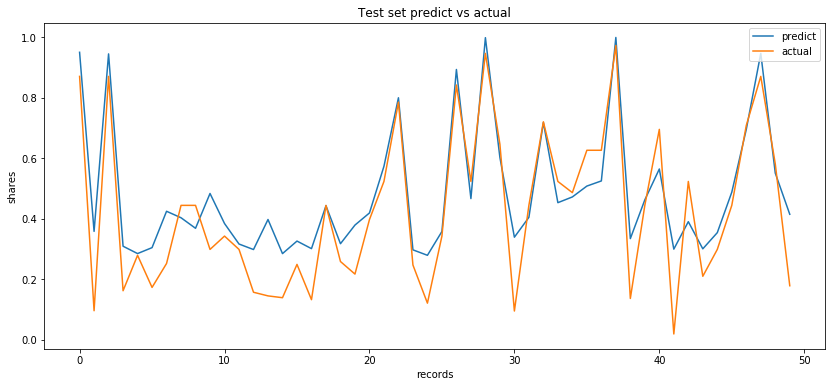
First, the MLP extraction network will be trained to provide the input for the RBF network. Once training is completed, the MLP extraction network will cease training to act as a static input for RBF network. In addition, the last layer of the MLP extraction network will be removed to achieve 20 input nodes for the RBF network. The RBF network will then be attached as the last layer of the MLP extraction network. This newly concocted network will then go through another round of training with the same dataset.

RBF network attached onto freeze extraction MLP network

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Layer (type) Output Shape Param #   
=================================================================  
dense\_17 (Dense) (None, 128) 5120   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_9 (Dropout) (None, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_18 (Dense) (None, 20) 2580   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_10 (Dropout) (None, 20) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
rbf\_layer\_5 (RBFLayer) (None, 20) 420   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_20 (Dense) (None, 1) 21   
=================================================================  
Total params: 8,141  
Trainable params: 441  
Non-trainable params: 7,700  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

****

Training and validation loss before 10th epoch shows signs of under fitting. After 15 epochs, loss started to plateau and showed signs of convergence for this model. Similarly to previous model result is affected by keras framework where training loss is higher than validation loss. This is also cause by dropout layer activated only in training phase. Comparing to the previous MLP training, validation curve is much smoother. This might be the result of features extraction model applied before RBF training.



RBF model shows the predicted value fluctuating closely to actual value. Similar to MLP model, RBF model suffers from the same problem of not being able to predict values below 0.2.

**Results**

|  |  |  |  |
| --- | --- | --- | --- |
|  | [**Linear Regression**](data_visualization.ipynb)  **(Baseline)** | [**MLP**](regression_mlp%20-%20shares.ipynb) | [**RBF**](regression_rbf%20-%20shares.ipynb) |
| **Loss** | - | 0.022189 | 0.022356 |
| **MSE (mean-square error)** | 0.065264 | 0.01104 | 0.014396 |
| **RMSE (root-mean-square error)** | 0.255469 | 0.105073 | 0.119984 |
| **Explained Variance Score** | -4.512672 | 0.817475 | 0.762068 |
| **Mean Absolute Error** | 0.216581 | 0.07962 | 0.09119 |
| **Mean Squared Log Error** | 0.032551 | 0.006801 | 0.009118 |
| **Median Absolute Error** | 0.2093 | 0.056895 | 0.069093 |
| **R2** | -4.675613 | 0.810248 | 0.72928 |

|  |  |  |
| --- | --- | --- |
|  | **MLP** | **RBF** |
| **5 KFold Loss** | 0.020229 | 0.021675 |
| **5 KFold MSE** | 0.009351 | 0.013678 |
| **5 KFold R2** | 0.850991 | 0.758933 |
| **5 KFold RMSE** | 0.096265 | 0.116905 |
| **10 KFold Loss** | 0.020229 | 0.021508 |
| **10 KFold MSE** | 0.009351 | 0.013511 |
| **10 KFold R2** | 0.850991 | 0.765071 |
| **10 KFold RMSE** | 0.096265 | 0.116211 |

|  |  |  |
| --- | --- | --- |
|  | **Average ensemble** | **Weighted ensemble** |
| **MSE (mean-square error)** | 0.010654 | 0.00989 |
| **RMSE (root-mean-square error)** | 0.103218 | 0.09945 |
| **Explained Variance Score** | 0.817691 | 0.836669 |
| **Mean Absolute Error** | 0.080815 | 0.075827 |
| **Mean Squared Log Error** | 0.00657 | 0.006131 |
| **Median Absolute Error** | 0.06289 | 0.055935 |
| **R2** | 0.815295 | 0.832654 |

KFold results show MLP model perform slight better than RBF. This shows that a more complex model will work better in this dataset. When using ensemble on both models, weighted ensemble result in better performance compare to average ensemble. Also, weighted ensemble provides better result compare to both MLP and RBF models.