Electricity Load Forecasting Using Neural Networks and Bagged Decision Trees

[View all machine learning examples](https://www.mathworks.com/solutions/machine-learning/examples.html)

This example demonstrates how to build an electricity load forecasting regression model using neural networks and bagged regression trees (data courtesy of ISO New England). Training data includes observations from the years 2004 to 2008 and test data includes observations after 2008.

Get Data

See the Dataset and References Section.

Load Data

load trainSet

load testSet

Regression Tree Model

mdlTree = RegressionTree.fit(trainX,trainY,'PredictorNames',labels);

% Store Forecasts in a stucture

forecast(1).Y = predict(mdlTree,testX);

Neural Network Model

net = fitnet(20);

net = train(net, trainX', trainY');

forecast(2).Y = net(testX')';

Bagged Decision Trees Model

mdlTreeBag = TreeBagger(100, trainX, trainY, 'method', 'regression', ...

'oobpred', 'on', 'minleaf', 30);

forecast(3).Y = predict(mdlTreeBag, testX);

Visualize Prediction Performance

Pick the one month visualization period between June and July. We will compare the predicted outputs on the test data along with the prediction error. This will aid in identifying which algorithm is more suitable for this application. All the custom visualization code can be automatically generated using [Plottools](https://www.mathworks.com/help/matlab/ref/plottools.html).

idx = testDates > datenum('Jun-01-2008') & testDates < datenum('Jul-01-2008');

Dates = testDates(idx);

figure('Units','Normalized','Position',[0.05,0.4,0.4,0.5]), subplot(2,1,1)

hPlot1 = plot(Dates, [testY(idx),forecast(1).Y(idx),...

forecast(2).Y(idx),forecast(3).Y(idx)],'LineWidth',2);

set(hPlot1(1),'LineWidth',5,'Color',[1 1 0],'DisplayName','Actual');

set(hPlot1(2),'Color',[0 1 0],'DisplayName','Regression Tree');

set(hPlot1(3),'DisplayName','Neural Network');

set(hPlot1(4),'Color',[0.5 0.5 0.5],'DisplayName','Bagged Regression Trees');

legend('show'),

datetick('x','mmm-dd','keepticks'), xlabel('Time'),

ylabel('Load'),

title('Load Prediction','FontSize',12,'FontWeight','Bold')

subplot(2,1,2)

hPlot2 = plot(Dates,[testY(idx)-forecast(1).Y(idx)...

testY(idx)-forecast(2).Y(idx),testY(idx)-forecast(3).Y(idx)]);

set(hPlot2(1),'Color',[0 1 0],'DisplayName','Regression Tree Error');

set(hPlot2(2),'Color','r','DisplayName','Neural Network Error');

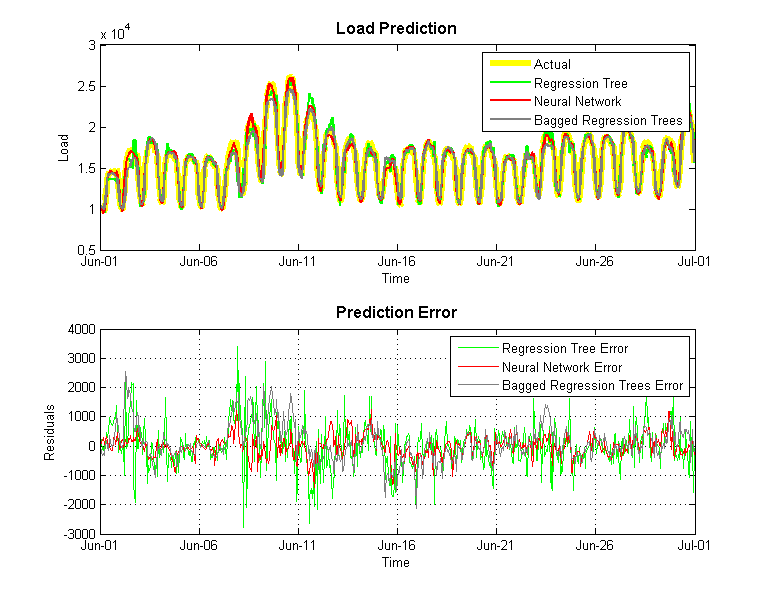
set(hPlot2(3),'Color',[0.5 0.5 0.5],'DisplayName','Bagged Regression Trees Error');

datetick('x','mmm-dd','keepticks'), xlabel('Time'),

title('Prediction Error','FontSize',12,'FontWeight','Bold')

ylabel('Residuals'), grid on

legend('show')



The plot chart reveals that this is a nonlinear regression problem and is not easily solved using parametric methods such as linear regression. Nonlinear models can be used, but only if you already know a model or equation that can be used to represent the load changes.

By looking at the residuals or the errors, decisions can be made about the machine learning algorithms that are most appropriate to use. In this example, it is clear that a single regression tree has high levels of residuals or errors as compared to a neural network or bagged decision trees.

Datasets and References

This demonstration uses data obtained from [ISO New England](http://www.iso-ne.com/) . The MATLAB MAT-file version of the data and the complete demonstration are available in the [Electricity Load and Price Forecasting Webinar Case Study](https://www.mathworks.com/matlabcentral/fileexchange/28684-electricity-load-and-price-forecasting-webinar-case-study/). To obtain the training and test datasets, click above link and extract the compressed file. Then follow the path included below:

*/Electricity Load & Price Forecasting/Load/Data/testSet.mat*

*/Electricity Load & Price Forecasting/Load/Data/trainSet.mat*