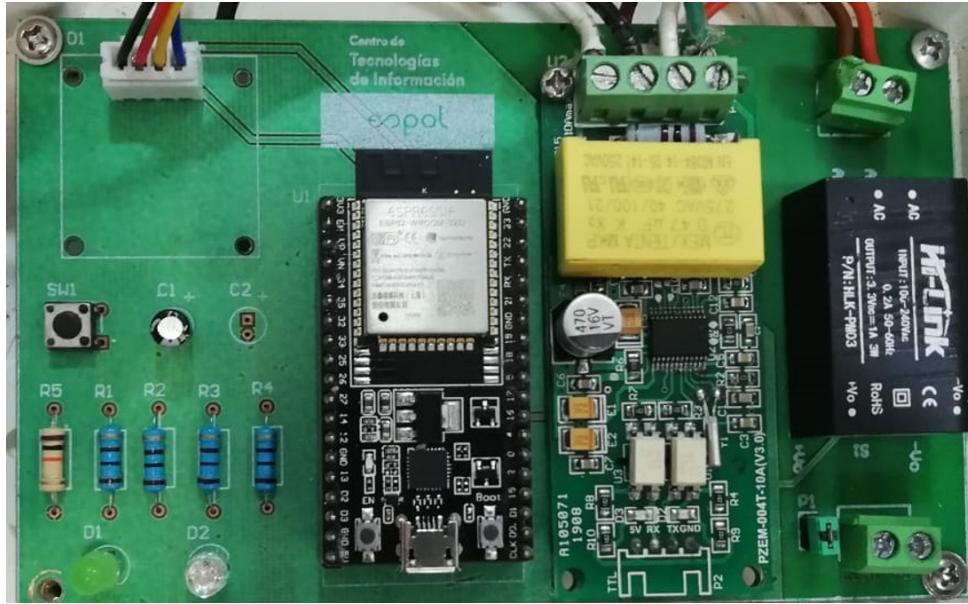


# ENERGY CONSUMPTION PREDICTION

Rebeca Estrada, Víctor Asanza, Danny Torres, Adrian Bazuerto, Irving Valeriano

- Main Code: <https://github.com/vasanza/EnergyConsumptionPrediction>
- DataPort: <https://dx.doi.org/10.21227/x6jw-m015>
- Reference: 10.1109/SAS51076.2021.9530151
- Sampling Rate=1Hz
- Date = '2021-05-06 00:00:00' AND '2021-11-22 22:04:21'



## 1- Load Raw Data

```
clear;clc;%clear all
%Prepare the raw dataset
addpath(genpath('./src'))%functions folders
datapath = fullfile('./data/');%data folder
filenames = FindCSV(datapath);%List All CSV files
allData=[];
for i=1:length(filenames)% Through all files
    data=readtable(fullfile(datapath,filenames(i).name));%Select i CSV file
    %The first 3 columns are deleted: topic, server date, esp32 date, and esp32 date.
    data=[data(:,2) data(:,5:end)];
    dataNew=table2array(data);% Array Double
    dataNew(isnan(dataNew)) = 0;%Remove NAN numbers
    allData=[allData;dataNew];
end
clear data dataNew;
```

## 2- Every 60 samples the value of 1 minute is averaged

```
win=60;% 1 minute
%It takes time to iterate the number of samples / time window.
```

```

allDataMean = fData_MeanWin(allData,win);
%save(fullfile(datapath,'allData.mat'),'allData');%No es posible para datos
%mayores a 2GB
save(fullfile(datapath,'allDataMean.mat'),'allDataMean');
clear allData win

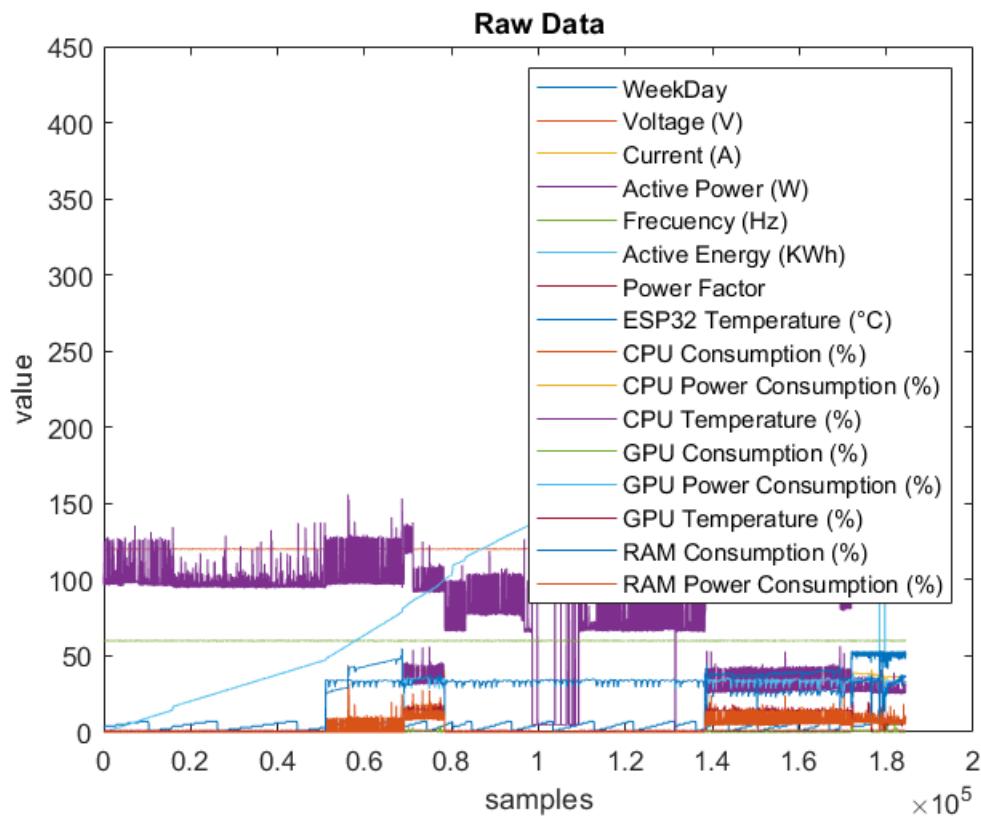
```

### 3- Plot Raw Data (Dataset)

```

figure
plot(allDataMean);title('Raw Data');xlabel('samples');ylabel('value');
legend('WeekDay','Voltage (V)', 'Current (A)', 'Active Power (W)', 'Frecuency (Hz)', 'Active Energy
    'ESP32 Temperature (°C)', 'CPU Consumption (%)', 'CPU Power Consumption (%)', 'CPU Temperature
    , 'GPU Power Consumption (%)', 'GPU Temperature (%)', 'RAM Consumption (%)', 'RAM Power Consumption

```



### 4- Preprocessing: Data Normalization and Plot Normalization Dataset

```

allDataNorm = fNormalization(allDataMean);%Normalization
%'Active Energy (KWh)'
[AEds] = datastats(allDataMean(:,6))

```

```

AEds = struct with fields:
  num: 184426
  max: 432.5034
  min: 0.0062

```

```

mean: 122.3349
median: 127.9700
range: 432.4971
std: 82.8325

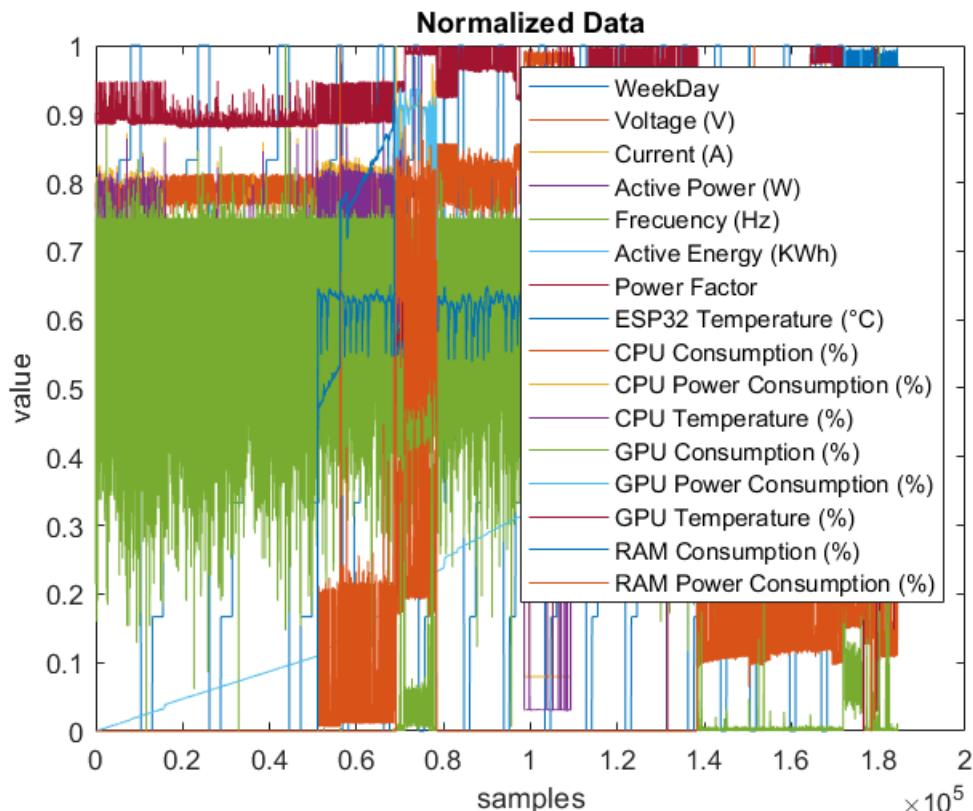
```

```
figure
```

```

plot(allDataNorm);title('Normalized Data');xlabel('samples');ylabel('value');
legend('WeekDay','Voltage (V)', 'Current (A)', 'Active Power (W)', 'Frecuency (Hz)', 'Active Energy (KWh)', 'Power Factor', 'ESP32 Temperature (°C)', 'CPU Consumption (%)', 'CPU Power Consumption (%)', 'GPU Consumption (%)', 'GPU Power Consumption (%)', 'GPU Temperature (%)', 'RAM Consumption (%)', 'RAM Power Consumption (%)')

```

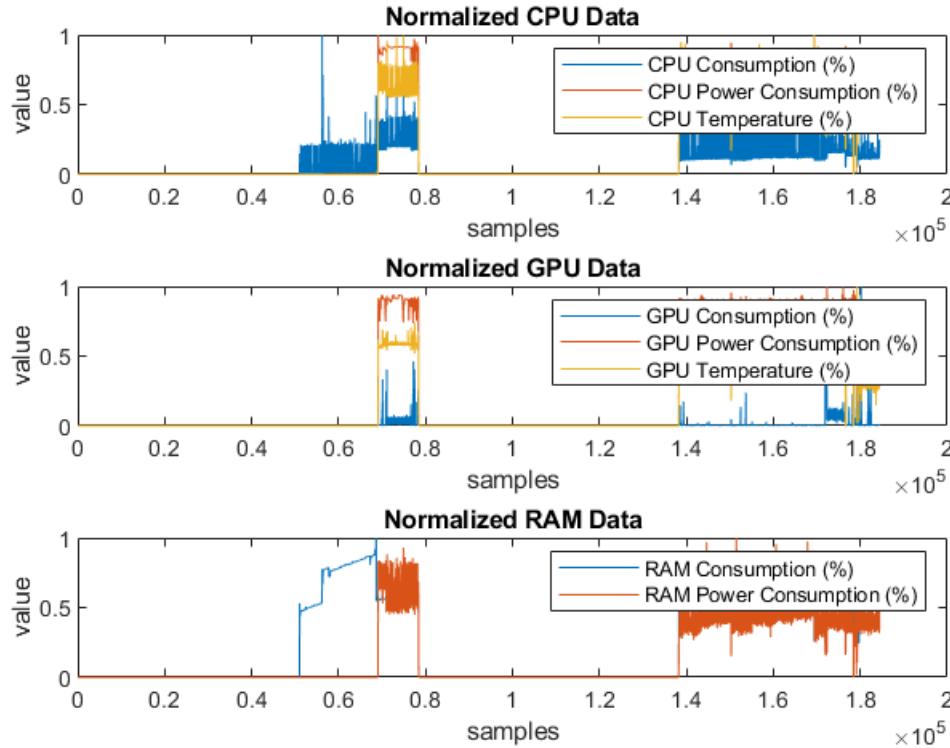


```
figure
```

```

subplot(3,1,1)
plot(allDataNorm(:,9:11));title('Normalized CPU Data');xlabel('samples');ylabel('value');
legend('CPU Consumption (%)', 'CPU Power Consumption (%)', 'CPU Temperature (%)');
subplot(3,1,2)
plot(allDataNorm(:,12:14));title('Normalized GPU Data');xlabel('samples');ylabel('value');
legend('GPU Consumption (%)', 'GPU Power Consumption (%)', 'GPU Temperature (%)');
subplot(3,1,3)
plot(allDataNorm(:,15:16));title('Normalized RAM Data');xlabel('samples');ylabel('value');
legend('RAM Consumption (%)', 'RAM Power Consumption (%)');

```



## 5- Statistical Information Raw Data (Dataset)

```
%'WeekDay', 'Voltage (V)'
[wds,vds] = datastats(allDataNorm(:,1),allDataNorm(:,2))
```

```
wds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.5172
  median: 0.5000
  range: 1
  std: 0.3314
vds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.7777
  median: 0.7916
  range: 1
  std: 0.1199
```

```
%'Current (A)', 'Active Power (W)'
[cds,pds] = datastats(allDataNorm(:,3),allDataNorm(:,4))
```

```
cds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.5788
```

```
median: 0.6525
range: 1
std: 0.1643
pds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.5562
  median: 0.6057
  range: 1
  std: 0.1626
```

```
%'Frecuency (Hz)', 'Active Energy (KWh)'
[ fds,eds ] = datastats(allDataNorm(:,5),allDataNorm(:,6))
```

```
fds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.6163
  median: 0.6258
  range: 1
  std: 0.0865
eds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.2828
  median: 0.2959
  range: 1
  std: 0.1915
```

```
%'Power Factor', 'ESP32 Temperature (°C)'
[ pfds,tds ] = datastats(allDataNorm(:,7),allDataNorm(:,8))
```

```
pfds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.8857
  median: 0.9219
  range: 1
  std: 0.1431
tds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.4519
  median: 0.5982
  range: 1
  std: 0.2982
```

```
%'CPU Consumption (%)', 'CPU Power Consumption (%)'
[ cpuds,cpupds ] = datastats(allDataNorm(:,9),allDataNorm(:,10))
```

```
cpuds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.0454
  median: 0
```

```
range: 1
std: 0.0723
cpupds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.2548
  median: 0
  range: 1
  std: 0.3896
```

```
%'CPU Temperature (%)',%'GPU Consumption (%)'
[cputds,gpuds] = datastats(allDataNorm(:,11),allDataNorm(:,12))
```

```
cputds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.1535
  median: 0
  range: 1
  std: 0.2373
gpuds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.0041
  median: 0
  range: 1
  std: 0.0211
```

```
%'GPU Power Consumption (%)', 'GPU Temperature (%)'
[gpupds,gputds] = datastats(allDataNorm(:,13),allDataNorm(:,14))
```

```
gpupds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.2631
  median: 0
  range: 1
  std: 0.4024
gputds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.1556
  median: 0
  range: 1
  std: 0.2449
```

```
%'RAM Consumption (%)', 'RAM Power Consumption (%)'
[ramds,rampds] = datastats(allDataNorm(:,15),allDataNorm(:,16))
```

```
ramds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.2691
  median: 0
  range: 1
  std: 0.3360
```

```
rampds = struct with fields:
  num: 184426
  max: 1
  min: 0
  mean: 0.1387
  median: 0
  range: 1
  std: 0.2210

clear wds vds cds pds fds eds pfds tds cpuds cpupds cputds gpuds gpupds gputds ramds rampds
```

## [6] - Feature Extraction (Please set TimeStep per time window)

```
%clear;clc;%clear all
%Prepare the raw dataset
addpath(genpath('./src'))%functions folders
datapath = fullfile('./data/');%data folder
allDataMean = load(fullfile(datapath,'allDataMean.mat'));%List All CSV files
allDataMean=allDataMean.allDataMean;
AEs = datastats(allDataMean(:,6))
```

```
AEs = struct with fields:
  num: 184426
  max: 432.5034
  min: 0.0062
  mean: 122.3349
  median: 127.9700
  range: 432.4971
  std: 82.8325
```

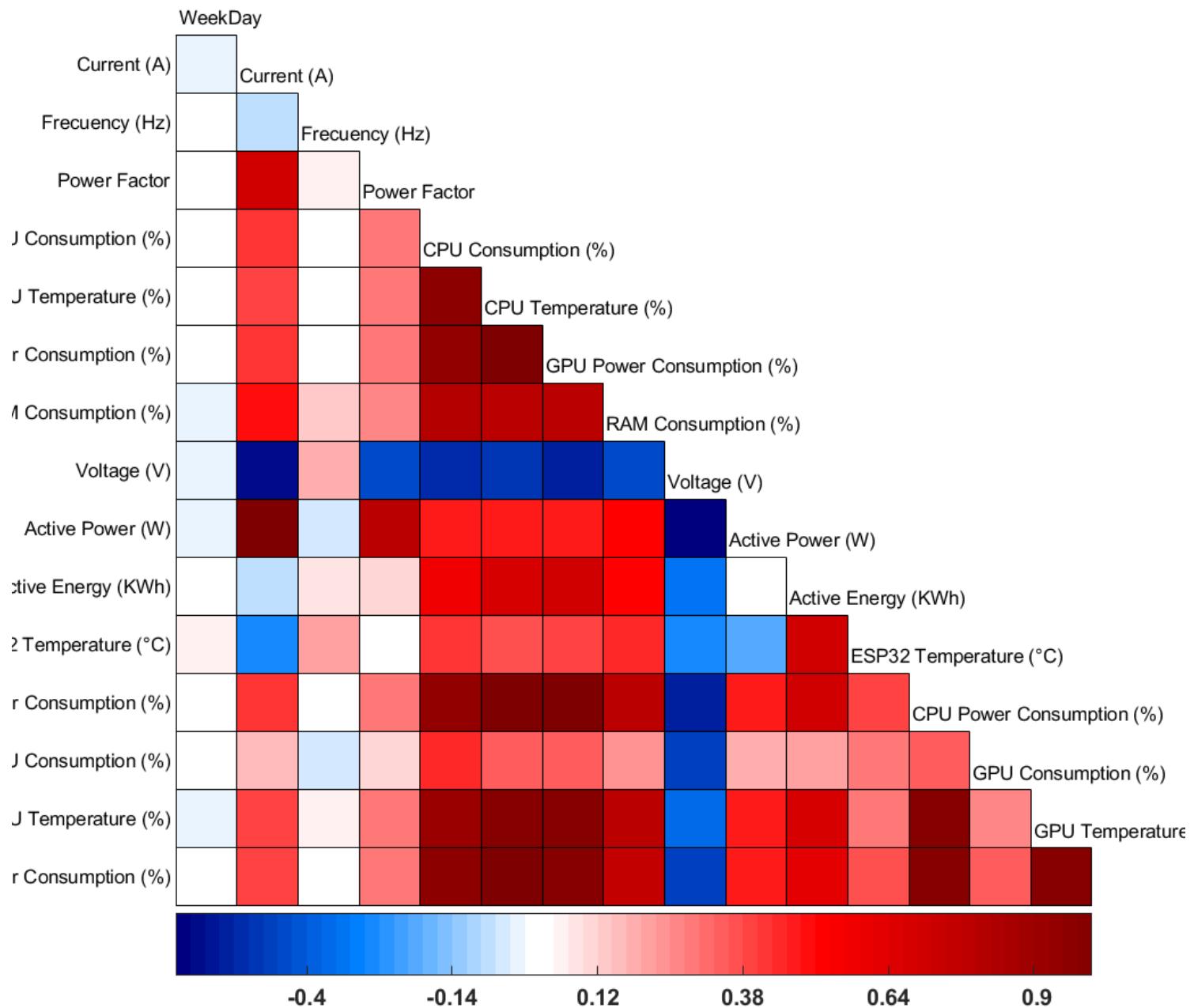
```
% Select only one TimeStep
%TimeStep =40320;% one month in minutes (only 8 monthss)
%TimeStep =10080;% one week in minutes
%TimeStep =1440;% one day in minutes
%TimeStep =60;% one hour in minutes
TimeStep =1;% one minute

%DataFeatures = frms_features_v2(allDataMean,TimeStep);
```

## 7 - Feature Selection (Please set the maximum correlation value allowed.)

```
threshold = 0.750;%-----Maximum correlation value allowed
% Labels:
FeaturesLabels = {'WeekDay','Voltage (V)', 'Current (A)', 'Active Power (W)', 'Frecuency (Hz)', 'AC
  'ESP32 Temperature (°C)', 'CPU Consumption (%)', 'CPU Power Consumption (%)', 'CPU Temperature (%)',
  'GPU Power Consumption (%)', 'GPU Temperature (%)', 'RAM Consumption (%)', 'RAM Power Consumption (%)'
[NewDataFeatures,NewFeaturesLabels,LabelsRemove] = Feature_Selection(DataFeatures,FeaturesLabels)
```

### Electrical Consumption Parameters



```
LabelsRemove(:)
```

```
ans = 8x1 cell
'Active Power (W)'
'Power Factor'
'CPU Power Consumption (%)'
'CPU Temperature (%)'
'GPU Power Consumption (%)'
'GPU Temperature (%)'
'RAM Consumption (%)'
'RAM Power Consumption (%)'
```

```
clear threshold LabelsRemove
```

## 8 - Regression Learner (Input output Data)

```
training_size=0.85; %70 Training, 15% Validation  
testing_size=1-training_size;%15% Testing  
Cell_Active_Energy=strfind(NewFeaturesLabels, 'Active Energy (KWh)');  
ind_Active_Energy = find(not(cellfun('isempty',Cell_Active_Energy)));  
% inputs: All previously selected features  
input_train>NewDataFeatures(1:round(length(NewDataFeatures)*training_size)-1,:);  
% output: Energy in the next time step  
output_train>NewDataFeatures(1+1:round(length(NewDataFeatures)*training_size),...  
    ind_Active_Energy);%Active Energy index 5  
regressionLearner  
clear Cell_Active_Energy testing_size Features_labels
```

RMSE (Validation): 1 month

<b>1.1</b> Linear Regression	RMSE (Validation): 19.707	RMSE (Validation): 19.906
Last change: Linear	5/5 features	5/5 features
<b>Current Model Summary</b>		
<b>Model 1.1:</b> Trained		
<b>Training Results</b>		
RMSE (Validation)	19.707	RMSE (Validation): 19.906
R-Squared (Validation)	0.93	Failed
MSE (Validation)	388.38	5/5 features
MAE (Validation)	18.732	5/5 features
Prediction speed	~54 obs/sec	5/5 features
Training time	4.8671 sec	5/5 features
<b>Model Type</b>		
Preset: Linear		
Terms: Linear		
Robust option: Off		
<b>Optimizer Options</b>		
Hyperparameter options disabled		
<b>Feature Selection</b>		
All features used in the model, before PCA		
<b>PCA</b>		
PCA disabled		
<b>1.13</b> SVM	RMSE (Validation): 76.881	RMSE (Validation): 76.881
Last change: Coarse Gaussian SVM	5/5 features	5/5 features
<b>1.14</b> Ensemble	RMSE (Validation): 75.92	RMSE (Validation): 75.92
Last change: Boosted Trees	5/5 features	5/5 features
<b>1.15</b> Ensemble	RMSE (Validation): 76.496	RMSE (Validation): 76.496
Last change: Bagged Trees	5/5 features	5/5 features
<b>1.16</b> Gaussian Process Regre...	RMSE (Validation): 76.797	RMSE (Validation): 76.797
Last change: Squared Exponential GPR	5/5 features	5/5 features
<b>1.17</b> Gaussian Process Regre...	RMSE (Validation): 76.796	RMSE (Validation): 76.796
Last change: Matern 5/2 GPR	5/5 features	5/5 features
<b>1.18</b> Gaussian Process Regre...	RMSE (Validation): 76.796	RMSE (Validation): 76.796
Last change: Exponential GPR	5/5 features	5/5 features
<b>1.19</b> Gaussian Process Regre...	RMSE (Validation): 76.796	RMSE (Validation): 76.796
Last change: Rational Quadratic GPR	5/5 features	5/5 features
<b>1.20</b> Neural Network	RMSE (Validation): 1716.8	RMSE (Validation): 1716.8
Last change: Narrow Neural Network	5/5 features	5/5 features
<b>1.21</b> Neural Network	RMSE (Validation): 847.56	RMSE (Validation): 847.56
Last change: Medium Neural Network	5/5 features	5/5 features
<b>1.22</b> Neural Network	RMSE (Validation): 967.45	RMSE (Validation): 967.45
Last change: Wide Neural Network	5/5 features	5/5 features
<b>1.23</b> Neural Network	RMSE (Validation): 793.22	RMSE (Validation): 793.22
Last change: Bilayered Neural Network	5/5 features	5/5 features
<b>1.24</b> Neural Network	RMSE (Validation): 985.72	RMSE (Validation): 985.72
Last change: Trilayered Neural Network	5/5 features	5/5 features

# RMSE (Validation): 1 Week

<b>1.16</b> Gaussian Process Regression	RMSE (Validation): <b>12.172</b>	<b>1.1</b> Linear Regression	RMSE (Validation): 15.997
Last change: Squared Exponential GPR	9/9 features	Last change: Linear	9/9 features
<b>▼ Current Model Summary</b>			
<b>Model 1.16:</b> Trained			
<b>Training Results</b>			
RMSE (Validation)	12.172	<b>1.2</b> Linear Regression	RMSE (Validation): 117.77
R-Squared (Validation)	0.96	Last change: Interactions Linear	9/9 features
MSE (Validation)	148.16	<b>1.3</b> Linear Regression	RMSE (Validation): 15.971
MAE (Validation)	6.972	Last change: Robust Linear	9/9 features
Prediction speed	~360 obs/sec	<b>1.4</b> Stepwise Linear Regression	RMSE (Validation): 19.572
Training time	1.0693 sec	Last change: Stepwise Linear	9/9 features
<b>Model Type</b>		<b>1.5</b> Tree	RMSE (Validation): 38.189
Preset: Squared Exponential GPR		Last change: Fine Tree	9/9 features
Basis function: Constant		<b>1.6</b> Tree	RMSE (Validation): 62.773
Kernel function: Squared Exponential		Last change: Medium Tree	9/9 features
Use isotropic kernel: true		<b>1.7</b> Tree	RMSE (Validation): 62.773
Kernel scale: Automatic		Last change: Coarse Tree	9/9 features
Signal standard deviation: Automatic		<b>1.8</b> SVM	RMSE (Validation): 20.776
Sigma: Automatic		Last change: Linear SVM	9/9 features
Standardize: true		<b>1.9</b> SVM	RMSE (Validation): 33.504
Optimize numeric parameters: true		Last change: Quadratic SVM	9/9 features
<b>Optimizer Options</b>		<b>1.10</b> SVM	RMSE (Validation): 24.365
Hyperparameter options disabled		Last change: Cubic SVM	9/9 features
<b>Feature Selection</b>		<b>1.11</b> SVM	RMSE (Validation): 55.011
All features used in the model, before PCA		Last change: Fine Gaussian SVM	9/9 features
<b>PCA</b>			
PCA disabled			

<b>1.12</b> SVM	RMSE (Validation): 30.274
Last change: Medium Gaussian SVM	9/9 features
<b>1.13</b> SVM	RMSE (Validation): 41.726
Last change: Coarse Gaussian SVM	9/9 features
<b>1.14</b> Ensemble	RMSE (Validation): 62.653
Last change: Boosted Trees	9/9 features
<b>1.15</b> Ensemble	RMSE (Validation): 63.755
Last change: Bagged Trees	9/9 features
<b>1.16</b> Gaussian Process Regression	RMSE (Validation): <b>12.172</b>
Last change: Squared Exponential GPR	9/9 features
<b>1.17</b> Gaussian Process Regression	RMSE (Validation): 12.227
Last change: Matern 5/2 GPR	9/9 features
<b>1.18</b> Gaussian Process Regression	RMSE (Validation): 27.498
Last change: Exponential GPR	9/9 features
<b>1.19</b> Gaussian Process Regression	RMSE (Validation): 12.172
Last change: Rational Quadratic GPR	9/9 features
<b>1.20</b> Neural Network	RMSE (Validation): 72.66
Last change: Narrow Neural Network	9/9 features
<b>1.21</b> Neural Network	RMSE (Validation): 97.424
Last change: Medium Neural Network	9/9 features
<b>1.22</b> Neural Network	RMSE (Validation): 82.503
Last change: Wide Neural Network	9/9 features
<b>1.23</b> Neural Network	RMSE (Validation): 89.113
Last change: Bilayered Neural Network	9/9 features
<b>1.24</b> Neural Network	RMSE (Validation): 30.464
Last change: Trilayered Neural Network	9/9 features

# RMSE (Validation): 1 day

<b>1.1</b> Linear Regression	RMSE (Validation): <b>1.422</b>	9/9 features
Last change: Linear		
<b>Current Model Summary</b>		
<b>Model 1.1:</b> Trained		
<b>Training Results</b>		
RMSE (Validation)	1.422	
R-Squared (Validation)	1.00	
MSE (Validation)	2.0222	
MAE (Validation)	0.76134	
Prediction speed	~1600 obs/sec	
Training time	3.4026 sec	
<b>Model Type</b>		
Preset: Linear		
Terms: Linear		
Robust option: Off		
<b>Optimizer Options</b>		
Hyperparameter options disabled		
<b>Feature Selection</b>		
All features used in the model, before PCA		
<b>PCA</b>		
PCA disabled		
<b>1.2</b> Linear Regression	RMSE (Validation): 22.566	
Last change: Interactions Linear		9/9 features
<b>1.3</b> Linear Regression	RMSE (Validation): 1.4827	
Last change: Robust Linear		9/9 features
<b>1.4</b> Stepwise Linear Regression	RMSE (Validation): 1.6707	
Last change: Stepwise Linear		9/9 features
<b>1.5</b> Tree	RMSE (Validation): 7.83	
Last change: Fine Tree		9/9 features
<b>1.6</b> Tree	RMSE (Validation): 13.623	
Last change: Medium Tree		9/9 features
<b>1.7</b> Tree	RMSE (Validation): 30.655	
Last change: Coarse Tree		9/9 features
<b>1.8</b> SVM	RMSE (Validation): 5.0595	
Last change: Linear SVM		9/9 features
<b>1.9</b> SVM	RMSE (Validation): 8.2114	
Last change: Quadratic SVM		9/9 features
<b>1.10</b> SVM	RMSE (Validation): 10.371	
Last change: Cubic SVM		9/9 features
<b>1.11</b> SVM	RMSE (Validation): 25.996	
Last change: Fine Gaussian SVM		9/9 features
<b>1.12</b> SVM	RMSE (Validation): 8.4495	
Last change: Medium Gaussian SVM		9/9 features
<b>1.13</b> SVM	RMSE (Validation): 13.671	
Last change: Coarse Gaussian SVM		9/9 features
<b>1.14</b> Ensemble	RMSE (Validation): 7.4586	
Last change: Boosted Trees		9/9 features
<b>1.15</b> Ensemble	RMSE (Validation): 11.509	
Last change: Bagged Trees		9/9 features
<b>1.16</b> Gaussian Process Regression	RMSE (Validation): 1.4414	
Last change: Squared Exponential GPR		9/9 features
<b>1.17</b> Gaussian Process Regression	RMSE (Validation): 1.4396	
Last change: Matern 5/2 GPR		9/9 features
<b>1.18</b> Gaussian Process Regression	RMSE (Validation): 4.2632	
Last change: Exponential GPR		9/9 features
<b>1.19</b> Gaussian Process Regression	RMSE (Validation): 1.4414	
Last change: Rational Quadratic GPR		9/9 features
<b>1.20</b> Neural Network	RMSE (Validation): 5.3255	
Last change: Narrow Neural Network		9/9 features
<b>1.21</b> Neural Network	RMSE (Validation): 4.3136	
Last change: Medium Neural Network		9/9 features
<b>1.22</b> Neural Network	RMSE (Validation): 3.5632	
Last change: Wide Neural Network		9/9 features
<b>1.23</b> Neural Network	RMSE (Validation): 3.7889	
Last change: Bilayered Neural Network		9/9 features
<b>1.24</b> Neural Network	RMSE (Validation): 3.0032	
Last change: Trilayered Neural Network		9/9 features

**RMSE (Validation): 1 hour**

<b>1.1</b> Linear Regression Last change: Linear	RMSE (Validation): <b>0.29826</b> 8/8 features	<b>1.2</b> Linear Regression Last change: Interactions Linear	RMSE (Validation): 0.30057 8/8 features
<b>Current Model Summary</b>		<b>1.4</b> Stepwise Linear Regression Last change: Stepwise Linear	RMSE (Validation): 0.29903 8/8 features
<b>Model 1.1:</b> Trained		<b>1.5</b> Tree Last change: Fine Tree	RMSE (Validation): 0.71839 8/8 features
<b>Training Results</b>		<b>1.6</b> Tree Last change: Medium Tree	RMSE (Validation): 0.76411 8/8 features
RMSE (Validation) 0.29826		<b>1.7</b> Tree Last change: Coarse Tree	RMSE (Validation): 1.8324 8/8 features
R-Squared (Validation) 1.00		<b>1.8</b> SVM Last change: Linear SVM	RMSE (Validation): 6.0814 8/8 features
MSE (Validation) 0.088957		<b>1.9</b> SVM Last change: Quadratic SVM	RMSE (Validation): 4.55 8/8 features
MAE (Validation) 0.0395		<b>1.10</b> SVM Last change: Cubic SVM	RMSE (Validation): 8.6276 8/8 features
Prediction speed ~33000 obs/sec		<b>1.11</b> SVM Last change: Fine Gaussian SVM	RMSE (Validation): 10.932 8/8 features
Training time 3.652 sec		<b>1.12</b> SVM Last change: Medium Gaussian SVM	RMSE (Validation): 5.5782 8/8 features

<b>1.13</b> SVM Last change: Coarse Gaussian SVM	RMSE (Validation): 4.4648 8/8 features
<b>1.14</b> Ensemble Last change: Boosted Trees	RMSE (Validation): 5.1148 8/8 features
<b>1.15</b> Ensemble Last change: Bagged Trees	RMSE (Validation): 2.5281 8/8 features
<b>1.16</b> Gaussian Process Regression Last change: Squared Exponential GPR	Failed 8/8 features
<b>1.17</b> Gaussian Process Regression Last change: Matern 5/2 GPR	RMSE (Validation): 0.2995 8/8 features
<b>1.18</b> Gaussian Process Regression Last change: Exponential GPR	RMSE (Validation): 1.5344 8/8 features
<b>1.19</b> Gaussian Process Regression Last change: Rational Quadratic GPR	Failed 8/8 features
<b>1.20</b> Neural Network Last change: Narrow Neural Network	RMSE (Validation): 0.29984 8/8 features
<b>1.21</b> Neural Network Last change: Medium Neural Network	RMSE (Validation): 0.30331 8/8 features
<b>1.22</b> Neural Network Last change: Wide Neural Network	RMSE (Validation): 0.3417 8/8 features
<b>1.23</b> Neural Network Last change: Bilayered Neural Network	RMSE (Validation): 0.30482 8/8 features
	<b>1.24</b> Neural Network Last change: Trilayered Neural Network
	RMSE (Validation): 19.813 8/8 features

RMSE (Validation): 1 minute

<b>1.2</b> Linear Regression Last change: Interactions Linear	RMSE (Validation): <b>0.027064</b> 8/8 features	<b>1.1</b> Linear Regression Last change: Linear	RMSE (Validation): 0.02709 8/8 features
Current Model Summary		<b>1.2</b> Linear Regression Last change: Interactions Linear	RMSE (Validation): <b>0.027064</b> 8/8 features
<b>Model 1.2:</b> Trained		<b>1.3</b> Linear Regression Last change: Robust Linear	RMSE (Validation): 0.027145 8/8 features
<b>Training Results</b> RMSE (Validation) 0.027064 R-Squared (Validation) 1.00 MSE (Validation) 0.00073247 MAE (Validation) 0.0020108 Prediction speed ~750000 obs/sec Training time 6.6415 sec		<b>1.4</b> Stepwise Linear Regression Last change: Stepwise Linear	RMSE (Validation): 0.027068 8/8 features
<b>Model Type</b> Preset: Interactions Linear Terms: Interactions Robust option: Off		<b>1.5</b> Tree Last change: Fine Tree	RMSE (Validation): 0.45591 8/8 features
<b>Optimizer Options</b> Hyperparameter options disabled		<b>1.6</b> Tree Last change: Medium Tree	RMSE (Validation): 0.45591 8/8 features
<b>Feature Selection</b> All features used in the model, before PCA		<b>1.7</b> Tree Last change: Coarse Tree	RMSE (Validation): 0.45591 8/8 features
<b>PCA</b> PCA disabled		<b>1.8</b> SVM Last change: Linear SVM	RMSE (Validation): 5.2124 8/8 features
		<b>1.9</b> SVM Last change: Quadratic SVM	RMSE (Validation): 4.6769 8/8 features
		<b>1.10</b> SVM Last change: Cubic SVM	RMSE (Validation): 4.3446 8/8 features
		<b>1.11</b> SVM Last change: Fine Gaussian SVM	RMSE (Validation): 4.9177 8/8 features

<b>1.12</b> SVM Last change: Medium Gaussian SVM	RMSE (Validation): 4.2971 8/8 features	
<b>1.13</b> SVM Last change: Coarse Gaussian SVM	RMSE (Validation): 3.4398 8/8 features	
<b>1.14</b> Ensemble Last change: Boosted Trees	RMSE (Validation): 5.0917 8/8 features	
<b>1.15</b> Ensemble Last change: Bagged Trees	RMSE (Validation): 0.5836 8/8 features	
<b>1.16</b> Gaussian Process Regression Last change: Squared Exponential GPR	RMSE (Validation): 0.055937 8/8 features	
<b>1.17</b> Gaussian Process Regression Last change: Matern 5/2 GPR	RMSE (Validation): 0.035091 8/8 features	
<b>1.18</b> Gaussian Process Regression Last change: Exponential GPR	RMSE (Validation): 0.11945 8/8 features	
<b>1.19</b> Gaussian Process Regression Last change: Rational Quadratic GPR	RMSE (Validation): 0.055263 8/8 features	
<b>1.20</b> Neural Network Last change: Narrow Neural Network	RMSE (Validation): 0.02716 8/8 features	
<b>1.21</b> Neural Network Last change: Medium Neural Network	RMSE (Validation): 0.027568 8/8 features	
<b>1.22</b> Neural Network Last change: Wide Neural Network	RMSE (Validation): 0.029096 8/8 features	
<b>1.23</b> Neural Network Last change: Bilayered Neural Network	RMSE (Validation): 0.027655 8/8 features	
	<b>1.24</b> Neural Network Last change: Trilayered Neural Network	RMSE (Validation): 0.041481 8/8 features

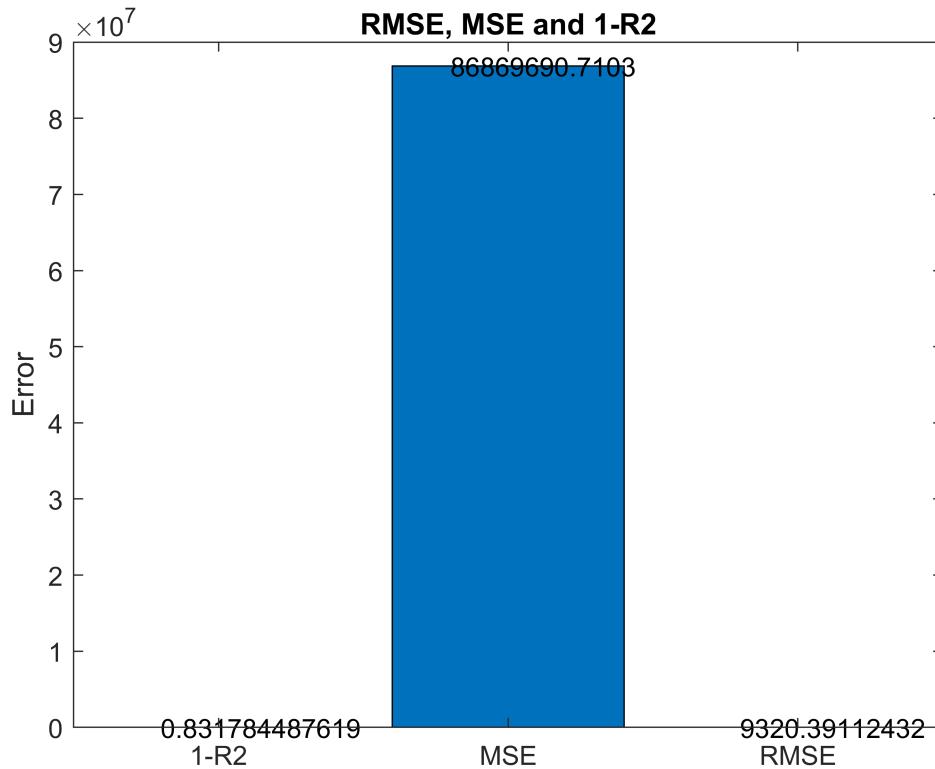
## RMSE (Testing): 1 week

```
% input: Energy
input_val=NewDataFeatures(round(length(NewDataFeatures)*training_size):length(NewDataFeatures))
% output: Energy in the next time step
output_val=NewDataFeatures(round(length(NewDataFeatures)*training_size)+1:length(NewDataFeatures)
    ,ind_Active_Energy);%Active Energy index 5
% Generate model
[model_week, RMSE_week] = trainRegressionModel_week(input_train, output_train);
```

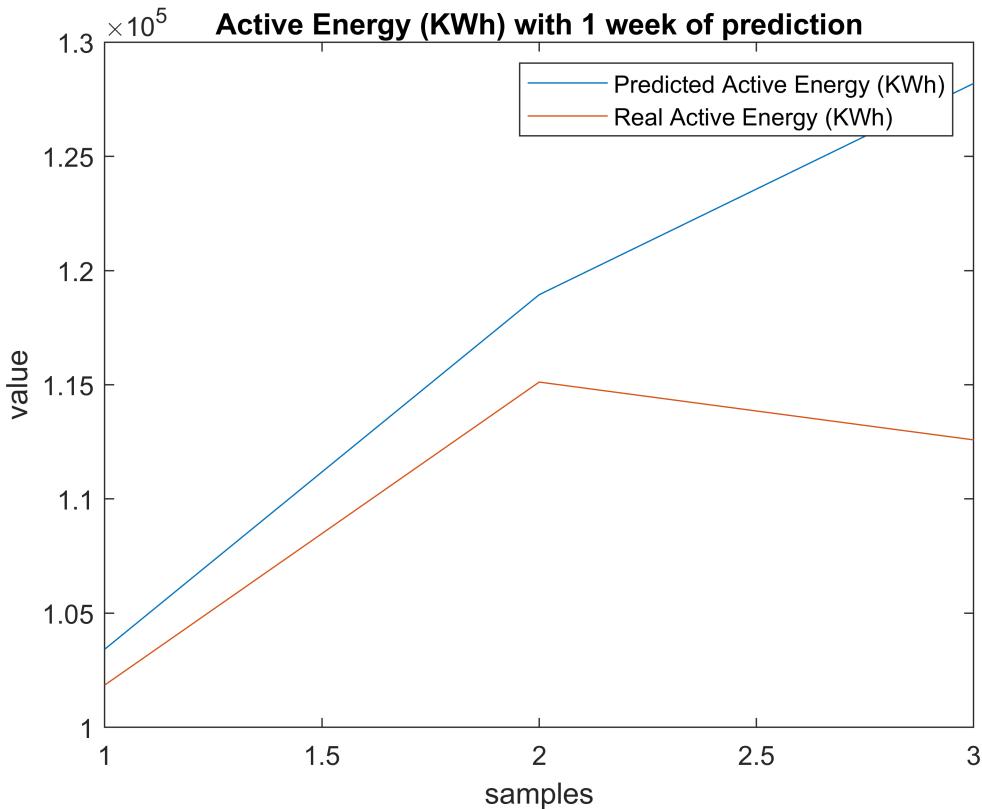
## RMSE\_week

```
RMSE_week = 13.5883
```

```
estimate_week=model_week.predictFcn(input_val);
estimate_week=fDenormalize(estimate_week,AEds.max,AEds.min);
output_week=fDenormalize(output_val,AEds.max,AEds.min);
[rmse_week,mse_week,R2_week]=fBar_RmseMseR2(estimate_week,output_week);
```



```
figure
plot([estimate_week,output_week]); xlabel('samples'); ylabel('value');
legend('Predicted Active Energy (KWh)', 'Real Active Energy (KWh)');
title('Active Energy (KWh) with 1 week of prediction');
```



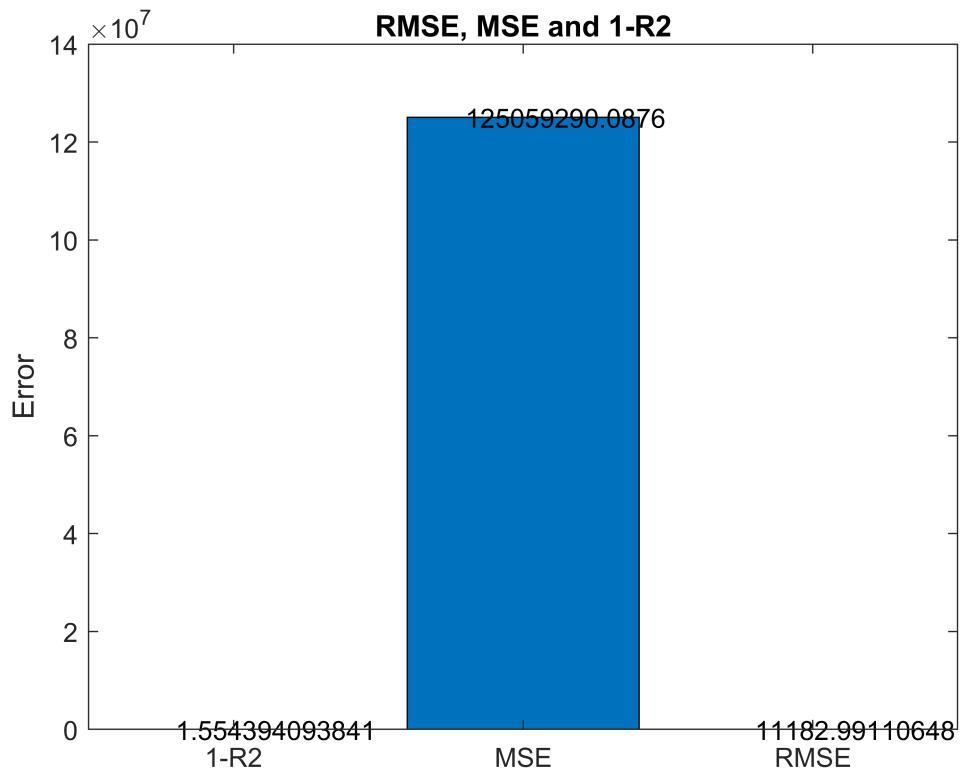
```
clear model_week RMSE_week output_week estimate_week
```

## RMSE (Testing): 1 day

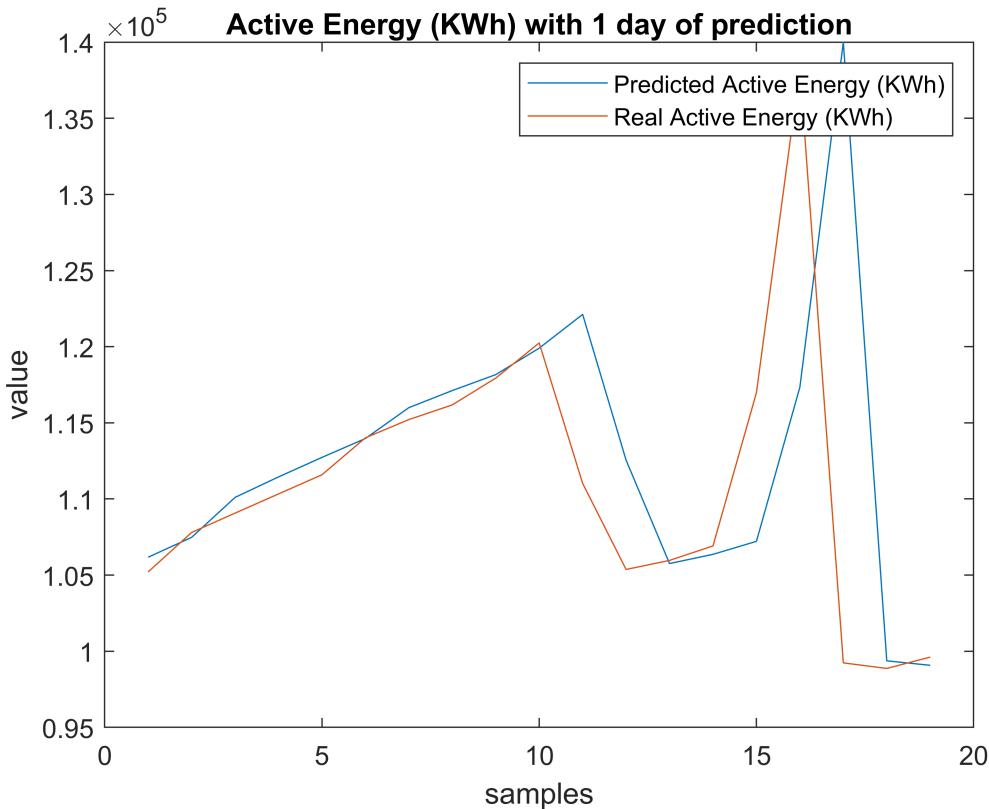
```
% input: Energy
input_val=NewDataFeatures(round(length(NewDataFeatures)*training_size):length(NewDataFeatures));
% output: Energy in the next time step
output_val=NewDataFeatures(round(length(NewDataFeatures)*training_size)+1:length(NewDataFeatures));
    ,ind_Active_Energy);%Active Energy index 5
% Generate model
[model_day, RMSE_day] = trainRegressionModel_day2(input_train, output_train);
RMSE_day
```

```
RMSE_day = 1.4174
```

```
estimate_day=model_day.predictFcn(input_val);
estimate_day=fDenormalize(estimate_day,AEds.max,AEds.min);
output_day=fDenormalize(output_val,AEds.max,AEds.min);
[rmse_day,mse_day,R2_day]=fBar_RmseMseR2(estimate_day,output_day);
```



```
figure
plot([estimate_day,output_day]); xlabel('samples'); ylabel('value');
legend('Predicted Active Energy (KWh)', 'Real Active Energy (KWh)');
title('Active Energy (KWh) with 1 day of prediction');
```



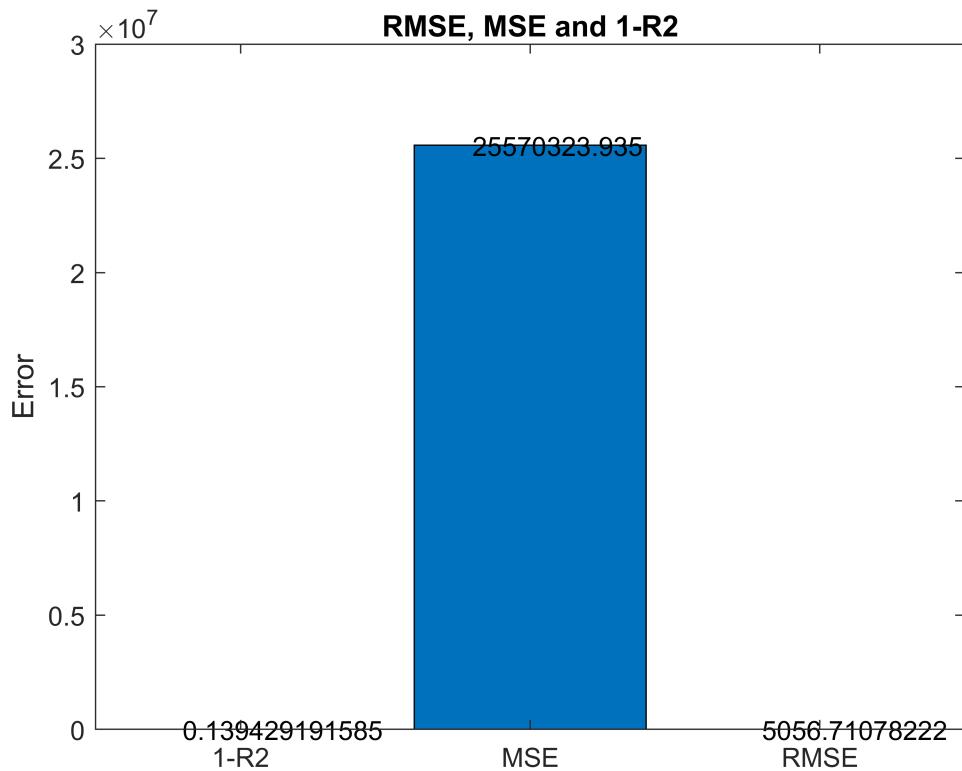
```
clear model_day RMSE_day output_day estimate_day
```

## RMSE (Testing): 1 hour

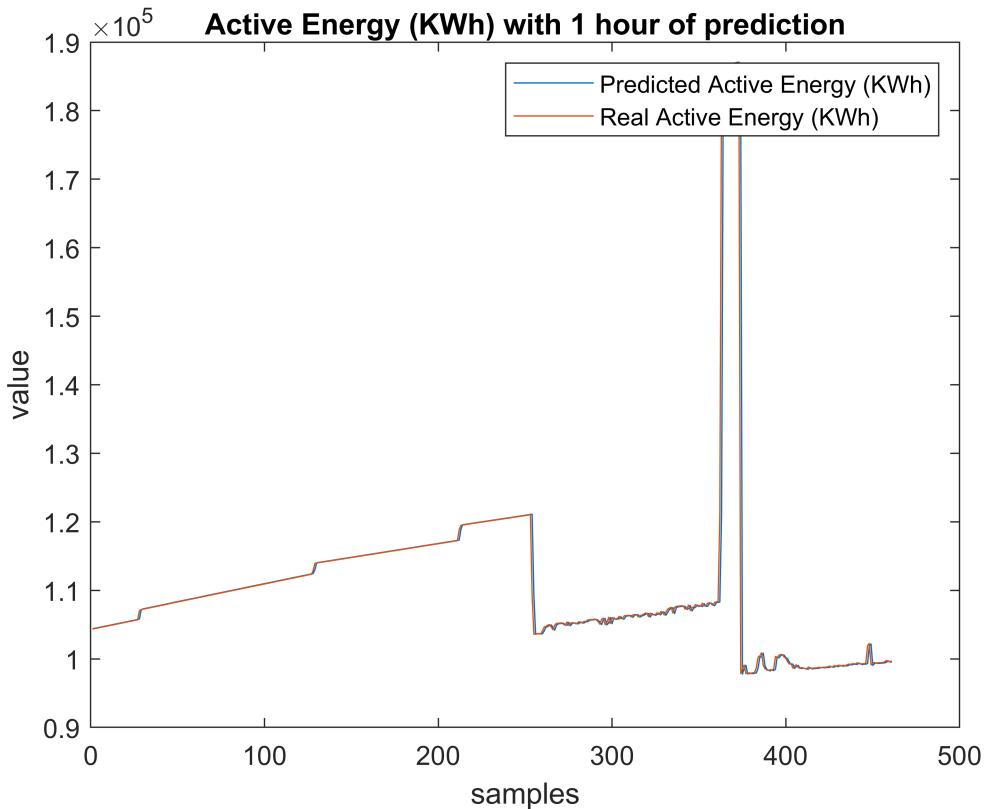
```
% input: Energy
input_val=NewDataFeatures(round(length(NewDataFeatures)*training_size):length(NewDataFeatures));
% output: Energy in the next time step
output_val=NewDataFeatures(round(length(NewDataFeatures)*training_size)+1:length(NewDataFeatures));
    ,ind_Active_Energy);%Active Energy index 5
% Generate model
[model_hour, RMSE_hour] = trainRegressionModel_hour(input_train, output_train);
RMSE_hour
```

```
RMSE_hour = 0.2982
```

```
estimate_hour=model_hour.predictFcn(input_val);
estimate_hour=fDenormalize(estimate_hour,AEds.max,AEds.min);
output_hour=fDenormalize(output_val,AEds.max,AEds.min);
[rmse_hour,mse_hour,R2_hour]=fBar_RmseMseR2(estimate_hour,output_hour);
```



```
figure
plot([estimate_hour,output_hour]); xlabel('samples'); ylabel('value');
legend('Predicted Active Energy (KWh)', 'Real Active Energy (KWh)');
title('Active Energy (KWh) with 1 hour of prediction');
```



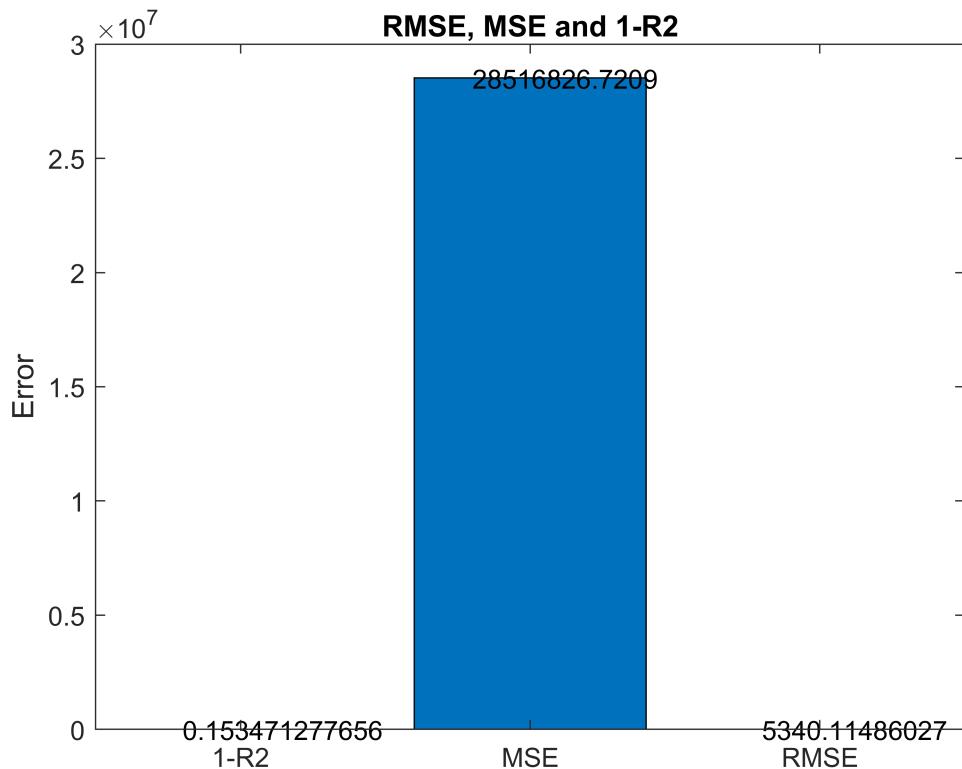
```
clear model_hour RMSE_hour output_hour estimate_hour
```

## RMSE (Testing): minute

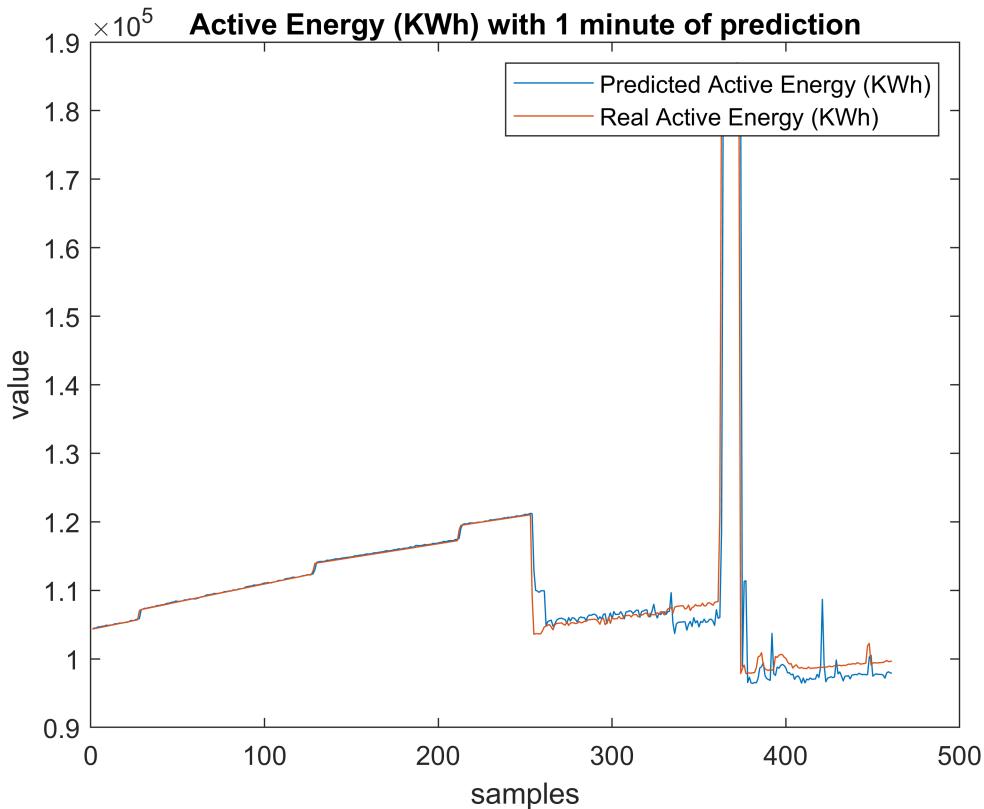
```
% input: Energy
input_val=NewDataFeatures(round(length(NewDataFeatures)*training_size):length(NewDataFeatures));
% output: Energy in the next time step
output_val=NewDataFeatures(round(length(NewDataFeatures)*training_size)+1:length(NewDataFeatures));
    ,ind_Active_Energy);%Active Energy index 5
% Generate model
[model_minute, RMSE_minute] = trainRegressionModel_minute(input_train, output_train);
RMSE_minute

RMSE_minute = 0.3009

estimate_minute=model_minute.predictFcn(input_val);
estimate_minute=fDenormalize(estimate_minute,AEds.max,AEds.min);
output_minute=fDenormalize(output_val,AEds.max,AEds.min);
[rmse_minute,mse_minute,R2_minute]=fBar_RmseMseR2(estimate_minute,output_minute);
```



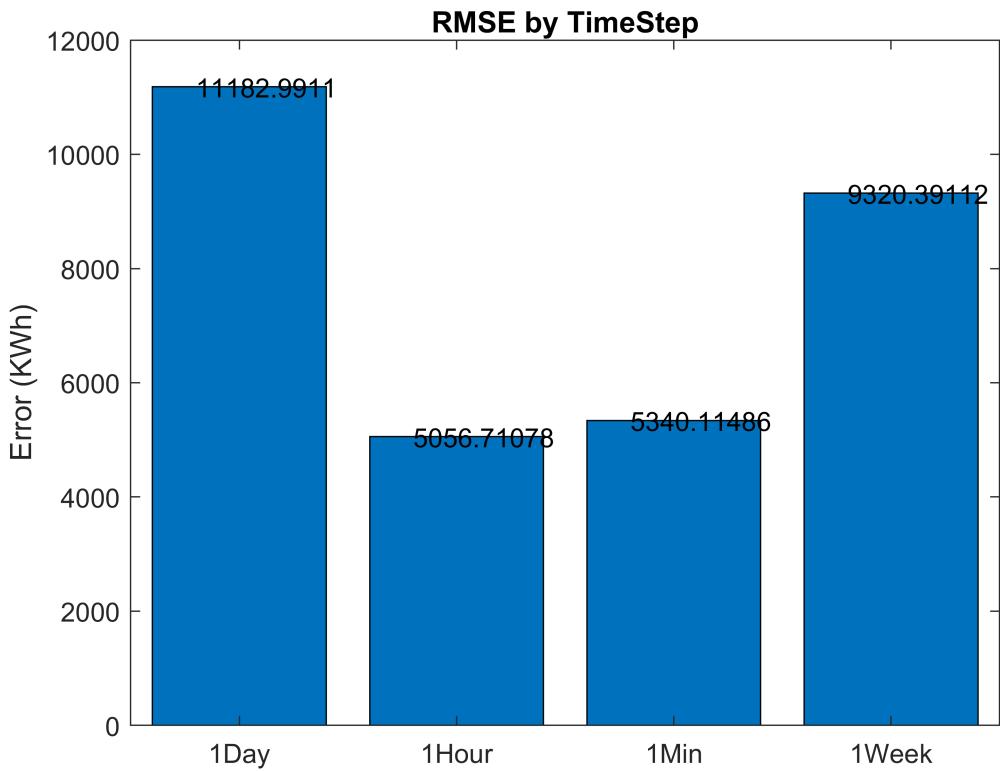
```
figure
plot([estimate_minute,output_minute]); xlabel('samples'); ylabel('value');
legend('Predicted Active Energy (KWh)', 'Real Active Energy (KWh)');
title('Active Energy (KWh) with 1 minute of prediction');
```



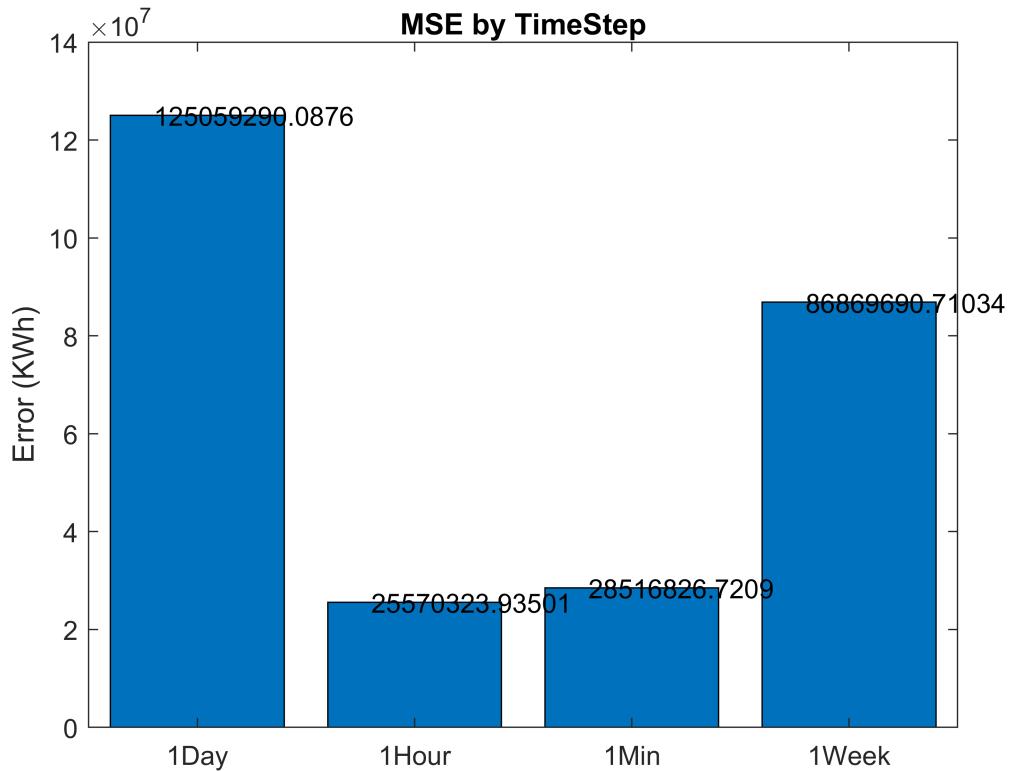
```
clear model_minute RMSE_minute output_minute estimate_minute
```

## Summary RMSE (Testing): Week, Day, Hour and Min

```
c = categorical({'1Week','1Day','1Hour','1Min'});
values = [rmse_week rmse_day rmse_hour rmse_minute];%rmse
figure;
b=bar(c,values);
ylabel('Error (KWh)')
title('RMSE by TimeStep')
xtips1 = b(1).XEndPoints - 0.2;
ytips1 = b(1).YEndPoints + 0.0003;
labels1 = string(b(1).YData);
text(xtips1,ytips1,labels1,'VerticalAlignment','middle')
```



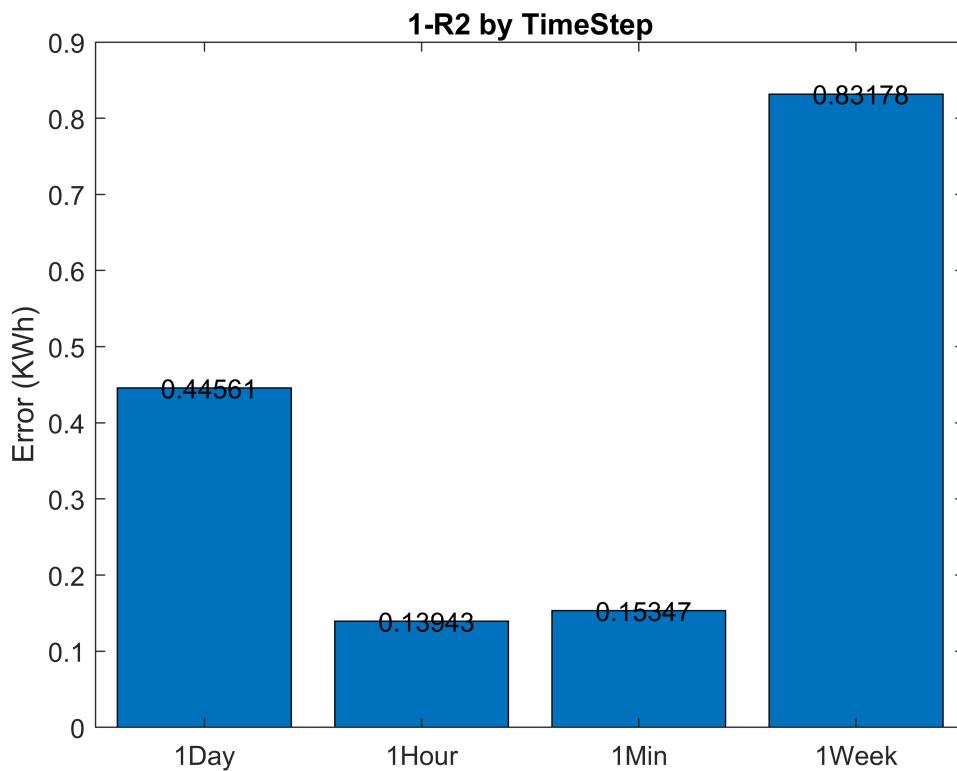
```
values = [mse_week mse_day mse_hour mse_minute];%mse
figure;
b=bar(c,values);
ylabel('Error (KWh)')
title('MSE by TimeStep')
xtips1 = b(1).XEndPoints - 0.2;
ytips1 = b(1).YEndPoints + 0.0003;
labels1 = string(b(1).YData);
text(xtips1,ytips1,labels1,'VerticalAlignment','middle')
```



```

values = [1-R2_week 1+R2_day 1-R2_hour 1-R2_minute];%1-R2
figure;
b=bar(c,values);
ylabel('Error (KWh)')
title('1-R2 by TimeStep')
xtips1 = b(1).XEndPoints - 0.2;
ytips1 = b(1).YEndPoints + 0.0003;
labels1 = string(b(1).YData);
text(xtips1,ytips1,labels1,'VerticalAlignment','middle')

```



## FutureWork

%Continue to record more data on more workstations.

%Create a model for estimating memory and CPU consumption since there are spaces without these  
%Make a real time consumption prediction system.