

```

In[542]:= ABMInputs800 = Import[
    "/Users/thorsilver/Downloads/ABM outputs1/LPtau800runs_GEMSA_inputs.csv"];

ABMOutputs800 =
    Import["/Users/thorsilver/Downloads/ABM outputs1/LPtau800runs GEMSA
    outputs only.csv"];

In[544]:= ABMOutputs800 = Function[x, x/1000] /@ ABMOutputs800;
ABMAssoc800 = AssociationThread[ABMInputs800 → Flatten[ABMOutputs800]];
ABMnewData800 = Dataset[ABMAssoc800];
ABMNormal800 = Normal[ABMAssoc800];
ABMNormalRandom = RandomSample[ABMNormal800];
ABMtrain800 = TakeDrop[ABMNormal800, 640];
ABMtest800 = ABMtrain800[[2]];
ABMtraining800 = ABMtrain800[[1]];
trainDevSplit800 = TakeDrop[ABMtraining800, 512];
finalTrain800 = trainDevSplit800[[1]];
finalDev800 = trainDevSplit800[[2]];
finaltest800 = ABMtest800;

In[556]:= Length[finalDev800]
Length[finalTrain800]
Length[finaltest800]

Out[556]= 128

Out[557]= 512

Out[558]= 160

```

Predictor Comparisons

```

In[559]:= pRFv800 = Predict[finalTrain800, Method → "RandomForest"]

```

```

Out[559]= PredictorFunction[

```





Input type: Mixed (number: 10)
 Method: RandomForest

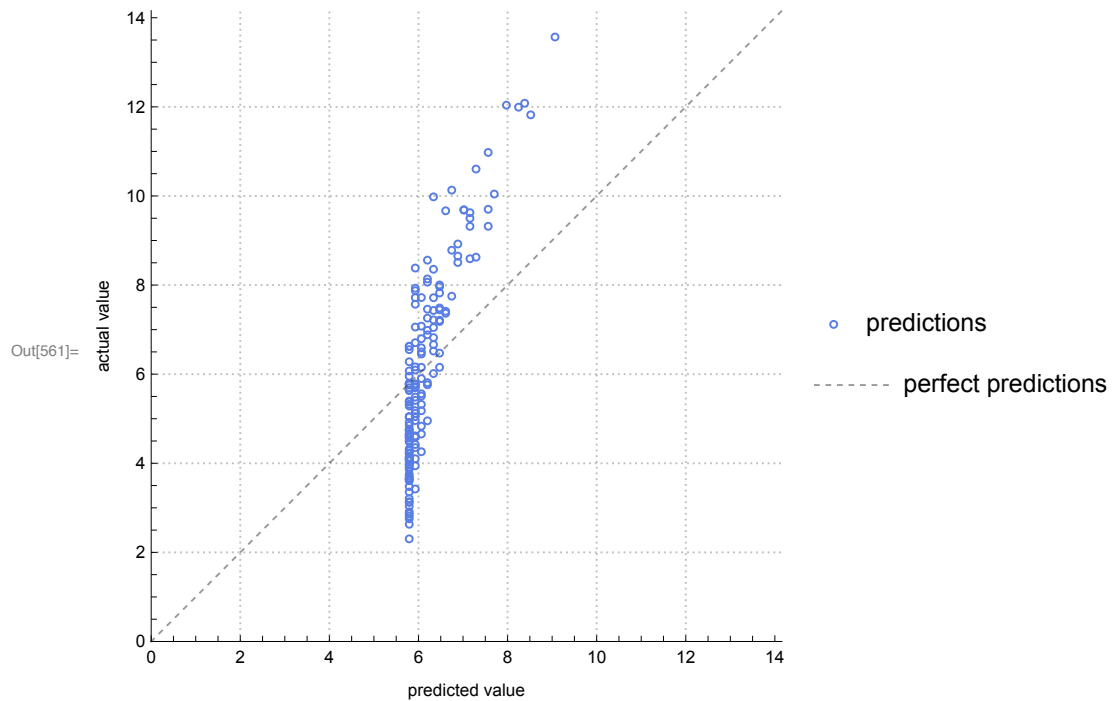
```

]
```


```
In[560]:= pmRFv800 = PredictorMeasurements[pRFv800, finaltest800]
pmRFv800["ComparisonPlot"]
```

```
Out[560]= PredictorMeasurementsObject[
```

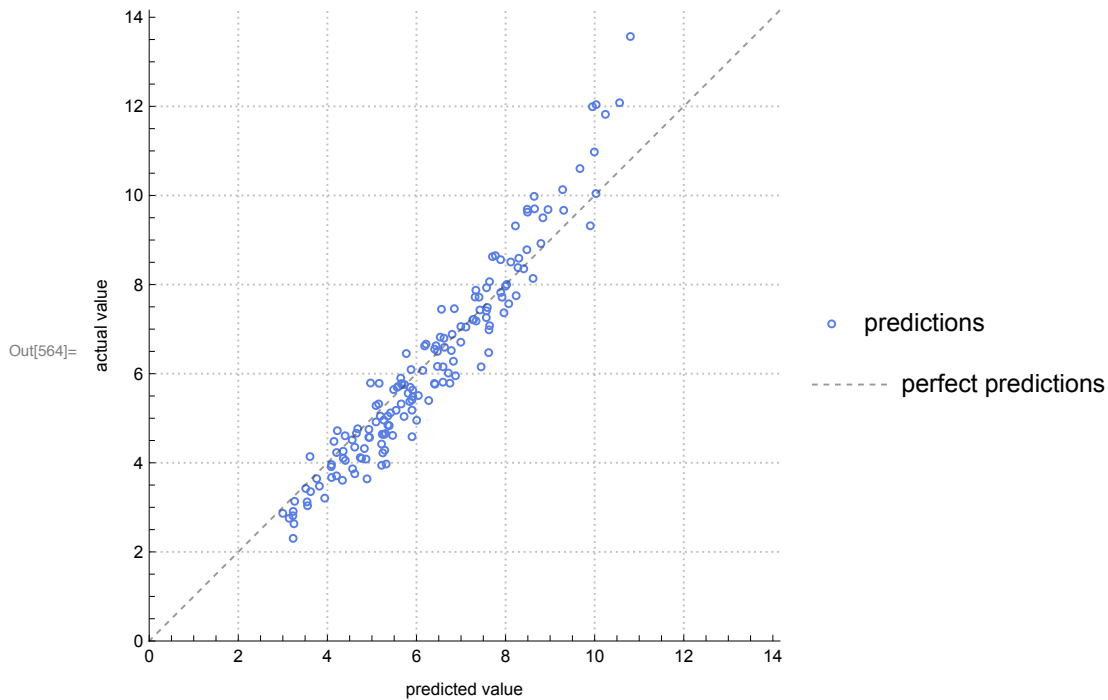
 Predictor: RandomForest
Number of test examples: 160




```
In[562]:= pXGTV800 = Predict[finalTrain800, Method -> "GradientBoostedTrees"]
pmXGTV800 = PredictorMeasurements[pXGTV800, finaltest800]
pmXGTV800["ComparisonPlot"]
```

Out[562]= PredictorFunction [ Input type: Mixed (number: 10)
Method: GradientBoostedTrees]

Out[563]= PredictorMeasurementsObject [ Predictor: GradientBoostedTrees
Number of test examples: 160]

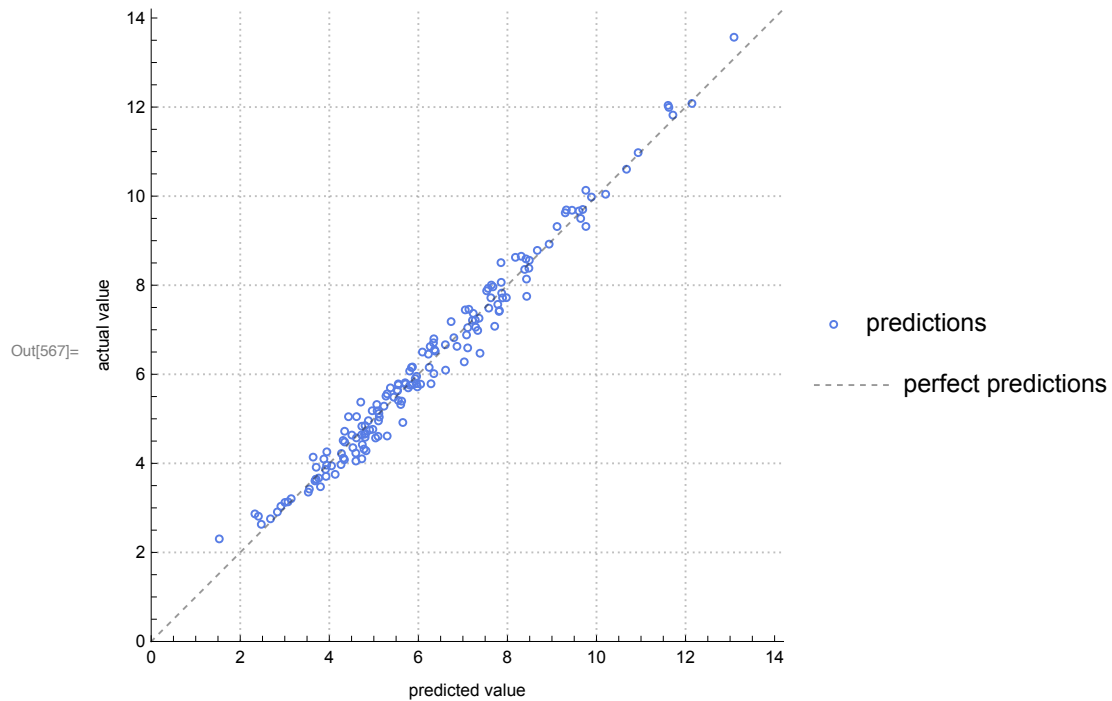


```
In[565]:= pNNv800 = Predict[finalTrain800, Method ->
{"NeuralNetwork", "NetworkDepth" -> 3, "NetworkType" -> "FullyConnected",
"L2Regularization" -> 0.05, MaxTrainingRounds -> 16 000}]
```

Out[565]= PredictorFunction [ Input type: Mixed (number: 10)
Method: NeuralNetwork]

```
In[566]:= pmNNv800 = PredictorMeasurements[pNNv800, finaltest800]
pmNNv800["ComparisonPlot"]
```

```
Out[566]= PredictorMeasurementsObject [ +  Predictor: NeuralNetwork  
Number of test examples: 160 ]
```



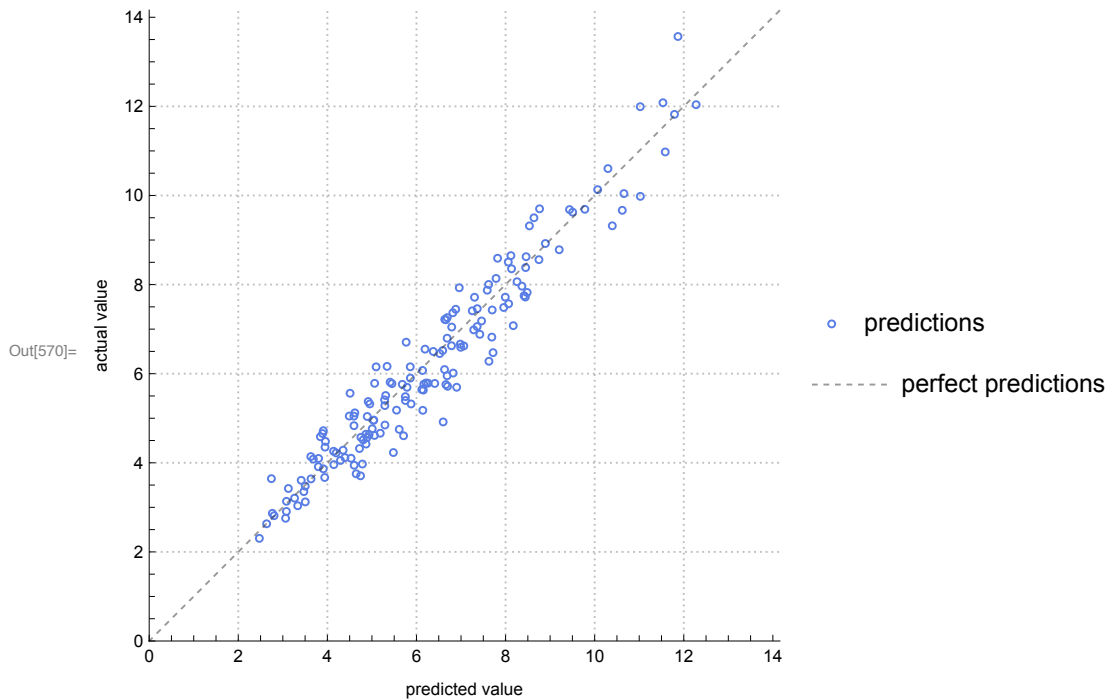
```

In[568]:= pGPEv800 = Predict[finalTrain800, Method → "GaussianProcess"]
pmGPEv800 = PredictorMeasurements[pGPEv800, finaltest800]
pmGPEv800["ComparisonPlot"]

```

Out[568]= PredictorFunction [ Input type: Mixed (number: 10)
Method: GaussianProcess]

Out[569]= PredictorMeasurementsObject [ Predictor: GaussianProcess
Number of test examples: 160]



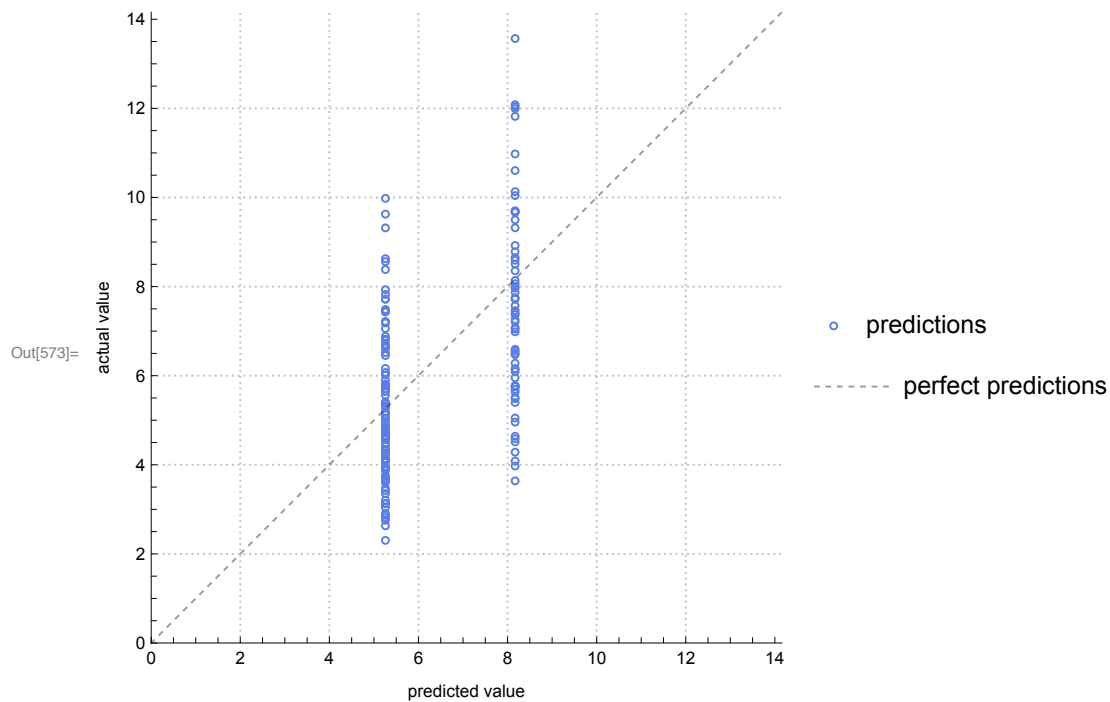
```

In[571]:= pDTv800 = Predict[finalTrain800, Method → "DecisionTree"]
          pmDTv800 = PredictorMeasurements[pDTv800, finaltest800]
          pmDTv800["ComparisonPlot"]

```

Out[571]= PredictorFunction [ Input type: Mixed (number: 10)
Method: DecisionTree]

Out[572]= PredictorMeasurementsObject [ Predictor: DecisionTree
Number of test examples: 160]




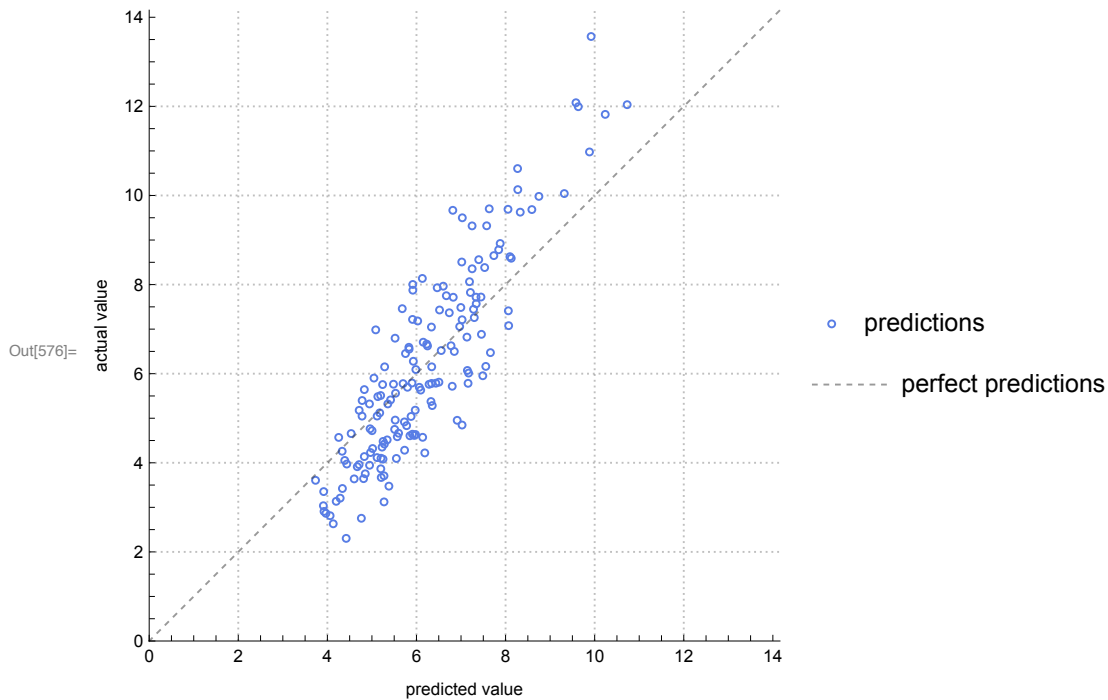
```

In[574]:= pNearestv800 = Predict[finalTrain800, Method → "NearestNeighbors"]
pmNearestv800 = PredictorMeasurements[pNearestv800, finaltest800]
pmNearestv800["ComparisonPlot"]

```


Out[574]= PredictorFunction [ Input type: Mixed (number: 10)
Method: NearestNeighbors]

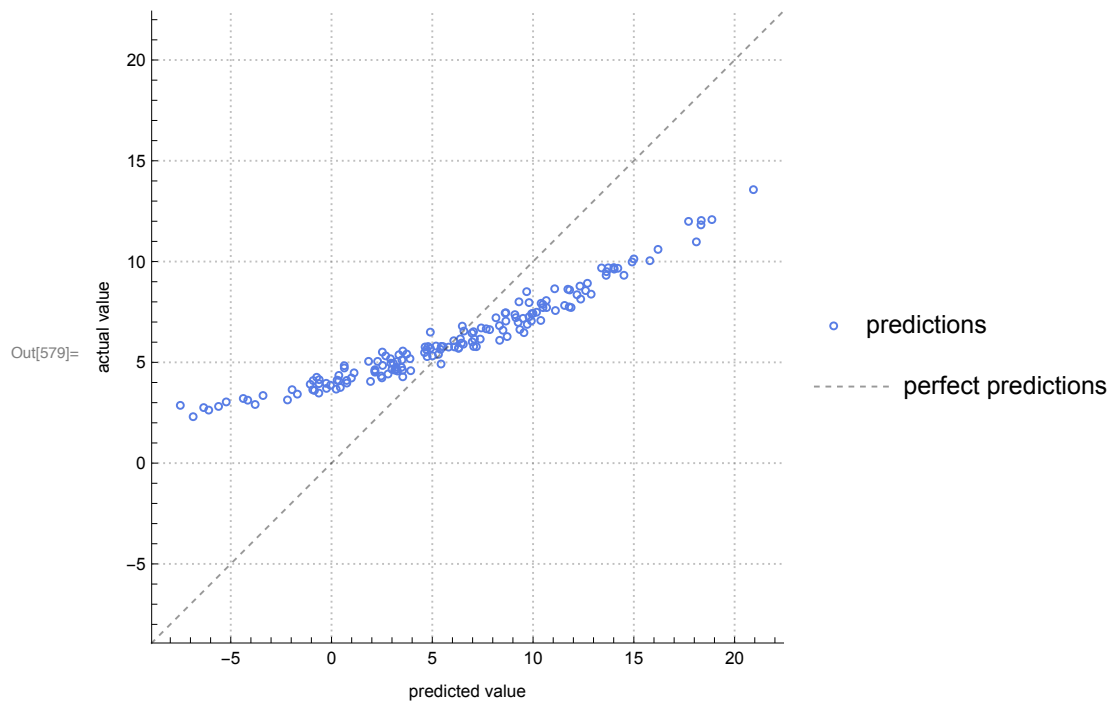
Out[575]= PredictorMeasurementsObject [ Predictor: NearestNeighbors
Number of test examples: 160]



```
In[577]:= pLRv800 = Predict[finalTrain800, Method → "LinearRegression"]
pmLRv800 = PredictorMeasurements[pLRv800, finaltest800]
pmLRv800["ComparisonPlot"]
```

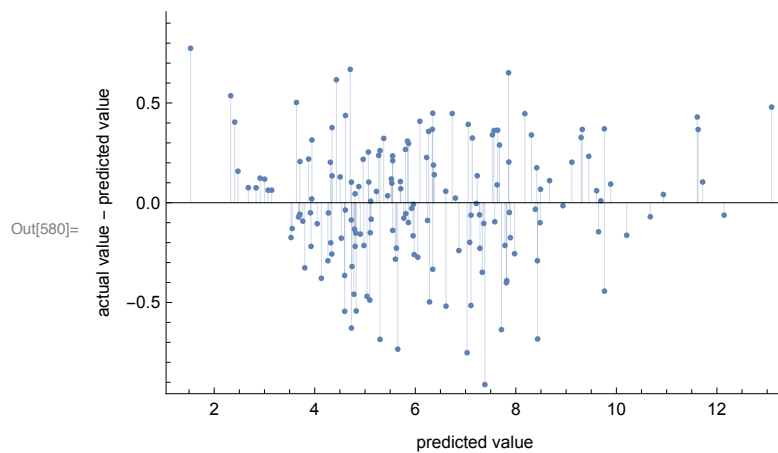
Out[577]= PredictorFunction [ Input type: Mixed (number: 10)
Method: LinearRegression]

Out[578]= PredictorMeasurementsObject [ Predictor: LinearRegression
Number of test examples: 160]

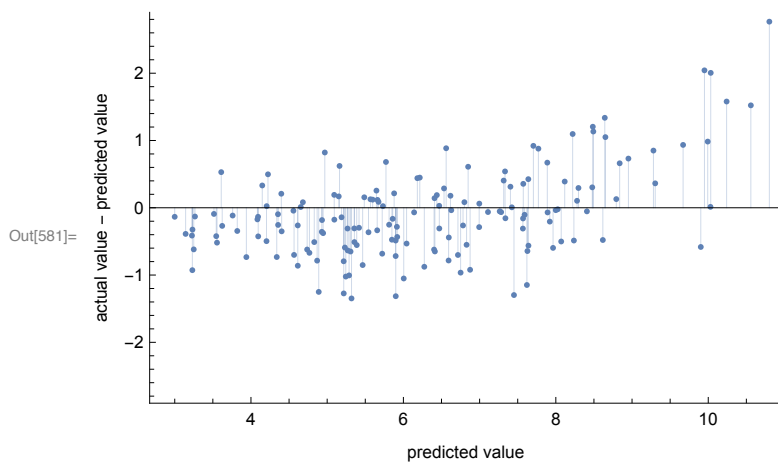


Residual Plots

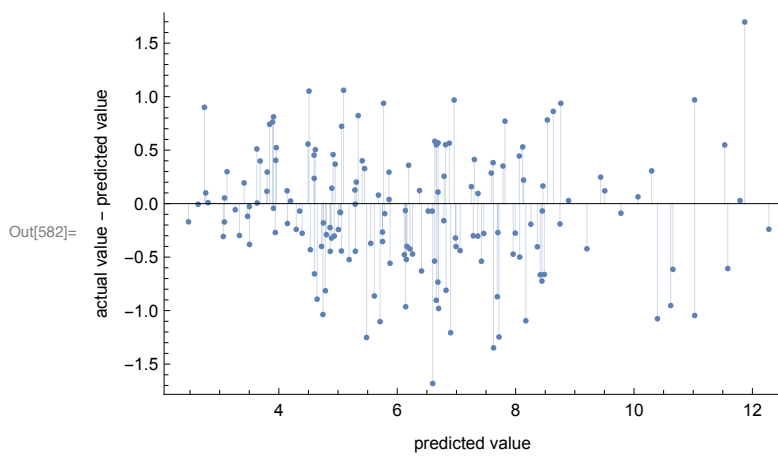
```
In[580]:= pmNNv800["ResidualPlot"]
```



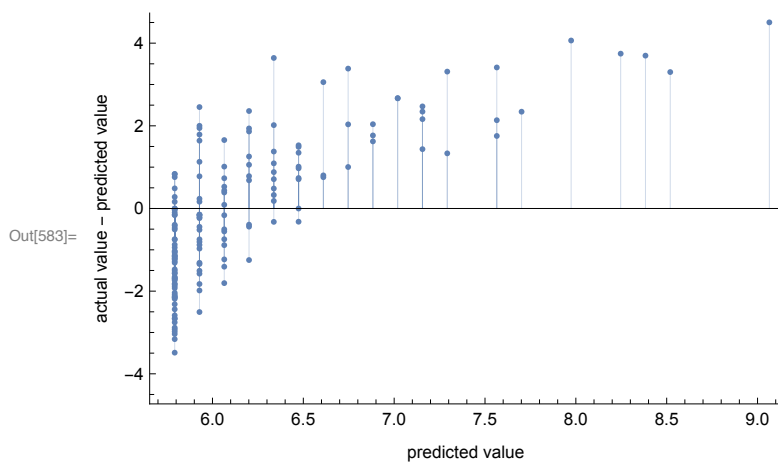

```
In[581]:= pmXGTV800["ResidualPlot"]
```



```
In[582]:= pmGPEv800["ResidualPlot"]
```



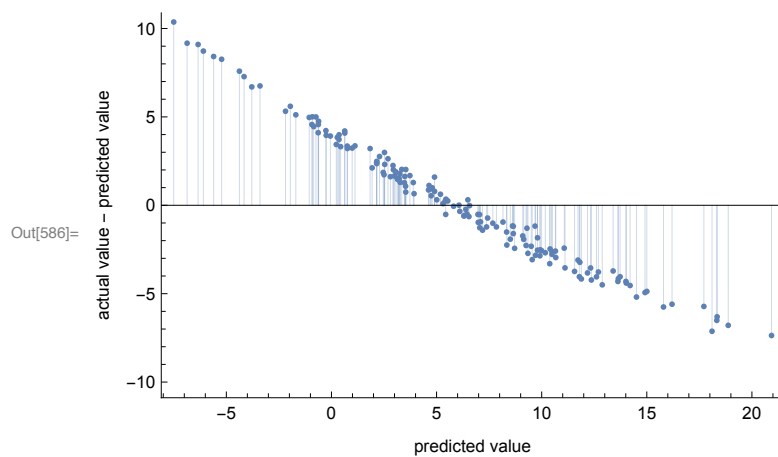
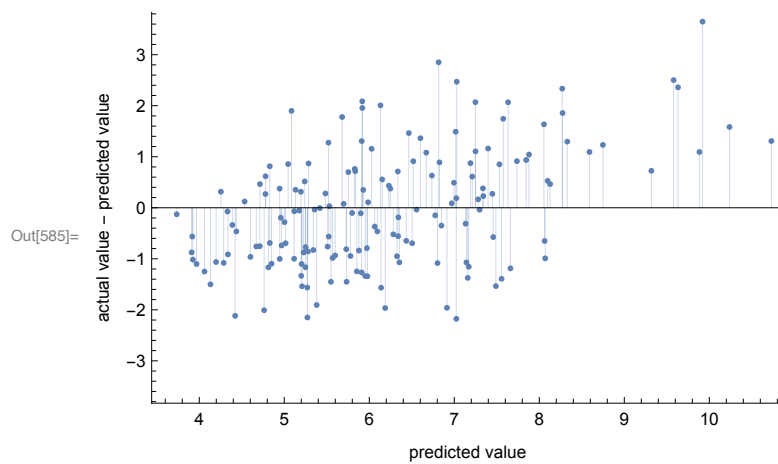
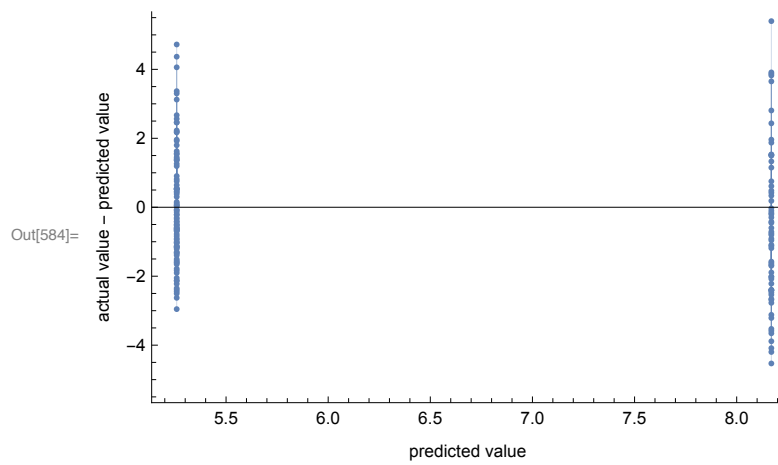
```
In[583]:= pmRFv800["ResidualPlot"]
```



```

In[584]:= pmDTv800["ResidualPlot"]
pmNearestv800["ResidualPlot"]
pmLRv800["ResidualPlot"]

```



MeanSquare

```

In[1120]:= pmNNv800["MeanSquare"]
           pmDTv800["MeanSquare"]
           pmLRv800["MeanSquare"]
           pmRFv800["MeanSquare"]
           pmXGTV800["MeanSquare"]
           pmGPEv800["MeanSquare"]
           pmLRv800["MeanSquare"]
           pmNearestv800["MeanSquare"]

Out[1120]= 0.0965913

Out[1121]= 3.85457

Out[1122]= 13.4386

Out[1123]= 2.9742

Out[1124]= 0.45184

Out[1125]= 0.329983

Out[1126]= 13.4386

Out[1127]= 1.35636

```

Predictor Summaries

```

In[595]:= Information[pNNv800]
          Information[pXGTV800]
          Information[pGPEv800]
          Information[pRFv800]

```

Predictor information

Data type	Mixed (number: 10)
Standard deviation	1.34 ± 0.63
Method	NeuralNetwork
Single evaluation time	2.18 ms/example
Batch evaluation speed	49.2 examples/ms
Loss	1.10 ± 0.16
Model memory	251. kB
Training examples used	512 examples
Training time	3 min 48 s

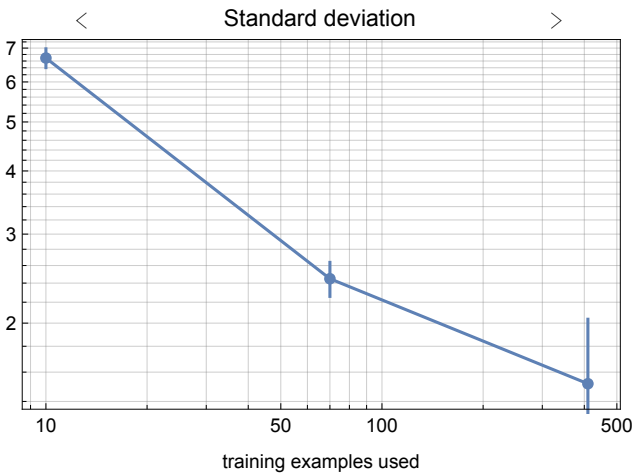
Out[595]=



Predictor information

Data type	Mixed (number: 10)
Standard deviation	1.52 ± 0.52
Method	GradientBoostedTrees
Single evaluation time	3.79 ms/example
Batch evaluation speed	50.5 examples/ms
Loss	1.56 ± 0.15
Model memory	391. kB
Training examples used	512 examples
Training time	1.5 s

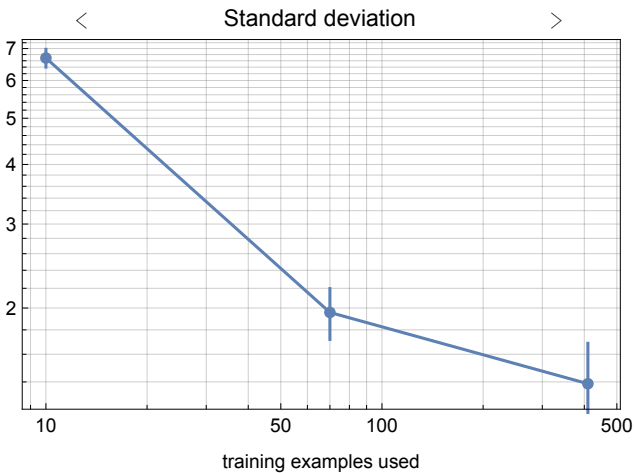
Out[596]=

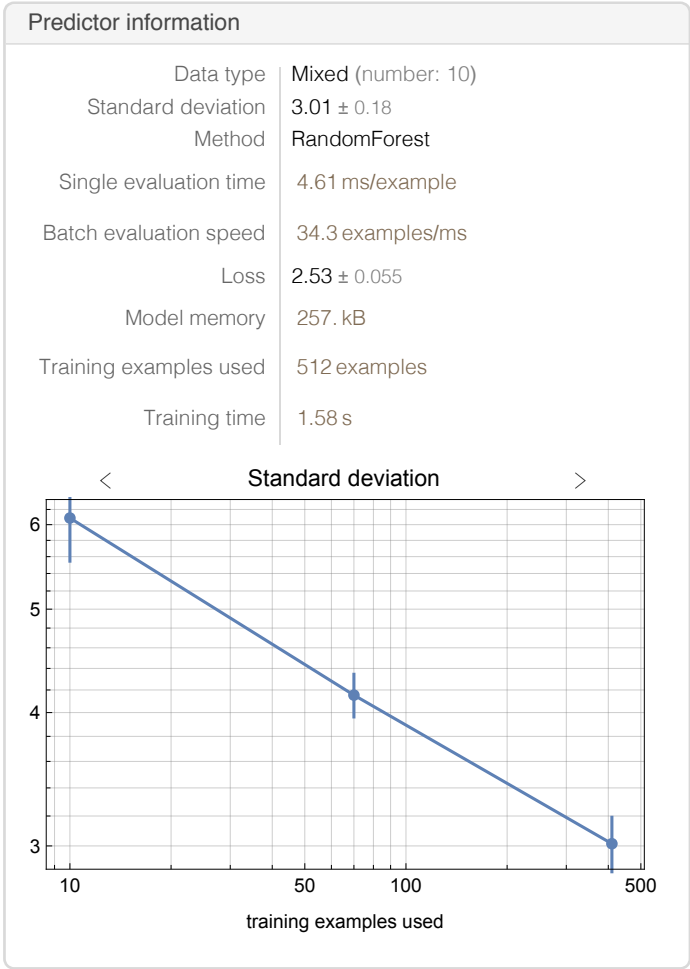


Predictor information

Data type	Mixed (number: 10)
Standard deviation	1.39 ± 0.30
Method	GaussianProcess
Single evaluation time	3.95 ms/example
Batch evaluation speed	22.8 examples/ms
Loss	1.57 ± 0.19
Model memory	1.59 MB
Training examples used	512 examples
Training time	4.12 s

Out[597]=



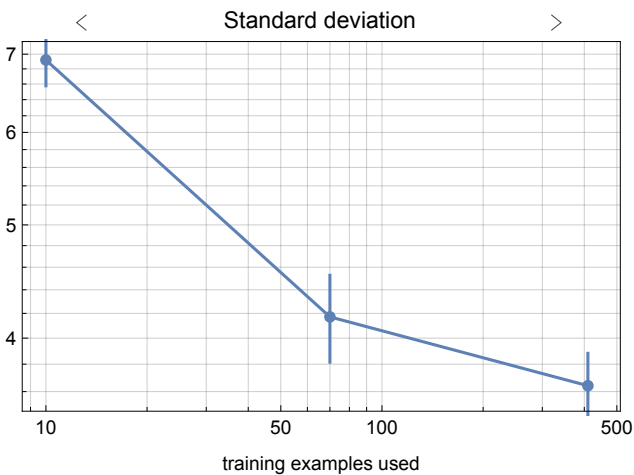


In[599]:= Information[pDTv800]
Information[pNearestv800]
Information[pLRv800]

Predictor information

Data type	Mixed (number: 10)
Standard deviation	3.64 ± 0.24
Method	DecisionTree
Single evaluation time	1.28 ms/example
Batch evaluation speed	418. examples/ms
Loss	2.68 ± 0.042
Model memory	149. kB
Training examples used	512 examples
Training time	532. ms

Out[599]=



Predictor information

Data type	Mixed (number: 10)
Standard deviation	3.15 ± 0.39
Method	NearestNeighbors
Single evaluation time	1.6 ms/example
Batch evaluation speed	84. examples/ms
Loss	2.75 ± 0.19
Model memory	188. kB
Training examples used	512 examples
Training time	370. ms

Out[600]=



Predictor information

Data type	Mixed (number: 10)
Standard deviation	2.54 ± 0.15
Method	LinearRegression
Single evaluation time	1.54 ms/example
Batch evaluation speed	288. examples/ms
Loss	2.34 ± 0.042
Model memory	263. kB
Training examples used	512 examples
Training time	1.42 s

Out[601]=

