Code Output: Using Machine Learning to predict Financial Crises -

An Evaluation of different Learning Algorithms for Early Warning Models.

Author: Chris Reimann.

Date: 28.02.2023.

In [1]:
```
# Import Experiment Code
from prepareData import Data
from doExperiment import Experiment
```

In [2]:
```
# Define Indicators and Models
iv_macro = ["rconsbarro", "iy", "money", "xrusd", "gdp", "cpi", "ca"]
iv_credit = ["tloans", "debtServ", "yieldCurve", "ltd", "debtgdp", "globaltloans", "globalyieldCurve"]
iv_ca = iv_credit + ["hpnom"]
iv_all = iv_macro + iv_ca

models = ["Logit", "KNeighbors", "RandomForest", "ExtraTrees", "SVM", "NeuralNet"]
```

In [3]:
```
# Construct Datasets
df_macro = Data(iv_macro).getReady("Macro")
df_credit = Data(iv_credit).getReady("Credit")
df_ca = Data(iv_ca).getReady("Credit & Asset")
df_all = Data(iv_all).getReady("All")
```

```
Macro: The final dataset contains 1591 observations with 63 distinct crisis events.
Credit: The final dataset contains 1373 observations with 60 distinct crisis events.
Credit & Asset: The final dataset contains 1159 observations with 46 distinct crisis events.
All: The final dataset contains 1101 observations with 41 distinct crisis events.
```

In [4]:
```
# Define Experiments
ex_macroIS = Experiment(df_macro, models, "InSample")
ex_creditIS = Experiment(df_credit, models, "InSample")
ex_caIS = Experiment(df_ca, models, "InSample")
ex_allIS = Experiment(df_all, models, "InSample")

ex_macro = Experiment(df_macro, models, "CrossVal")
ex_credit = Experiment(df_credit, models, "CrossVal")
ex_ca = Experiment(df_ca, models, "CrossVal")
ex_all = Experiment(df_all, models, "CrossVal")
```

Run Experiments: In-Sample

In [5]:
```
# Run Experiments: Macro Indicators InSample
ex_macroIS.run()
ex_macroIS.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[5]:

	Model	AUC
0	KNeighbors	1.000000
1	RandomForest	0.981277
2	ExtraTrees	0.833469
3	SVM	0.759257
4	NeuralNet	0.713687
5	Logit	0.707710
6	Random Assignment	0.500000

In [6]:
```
# Run Experiments: Credit Indicators InSample
ex_creditIS.run()
ex_creditIS.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[6]:

	Model	AUC
0	KNeighbors	1.000000
1	RandomForest	0.983538
2	ExtraTrees	0.934343
3	SVM	0.918643
4	NeuralNet	0.851912
5	Logit	0.790217
6	Random Assignment	0.500000

In [7]:
```
# Run Experiments: Credit & Asset Indicators InSample
ex_caIS.run()
ex_caIS.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[7]:

	Model	AUC
0	KNeighbors	1.000000
1	RandomForest	0.992726
2	ExtraTrees	0.955384
3	SVM	0.943323
4	NeuralNet	0.864574
5	Logit	0.808380
6	Random Assignment	0.500000

In [8]:
```
# Run Experiments: All Indicators InSample + Logit Coefficients
ex_allIS.run()
display(ex_allIS.auc)
ex_allIS.logitCoef()
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

	Model	AUC
0	KNeighbors	1.000000
1	RandomForest	0.994486
2	ExtraTrees	0.963970
3	SVM	0.951174
4	NeuralNet	0.932865
5	Logit	0.858774
6	Random Assignment	0.500000

```
Optimization terminated successfully.
         Current function value: 0.228989
         Iterations 8
```

Out[8]:

	Coef.	Std.Err.	z	P>|z|	[0.025	0.975]
rconsbarro	-9.339301	2.363592	-3.951316	7.772255e-05	-13.971857	-4.706745
iy	1.889071	4.128478	0.457571	6.472609e-01	-6.202598	9.980740
money	-1.333617	1.874857	-0.711316	4.768882e-01	-5.008269	2.341036
xrusd	-2.041967	0.624737	-3.268525	1.081097e-03	-3.266428	-0.817506
gdp	-8.000322	1.324385	-6.040784	1.533676e-09	-10.596068	-5.404575
cpi	8.991885	1.436438	6.259851	3.853465e-10	6.176519	11.807251
ca	-7.857055	3.789597	-2.073322	3.814230e-02	-15.284528	-0.429582
tloans	1.684385	1.383464	1.217512	2.234093e-01	-1.027155	4.395925
debtServ	-10.678297	8.783358	-1.215742	2.240832e-01	-27.893362	6.536769
yieldCurve	-0.481616	0.096841	-4.973246	6.584103e-07	-0.671421	-0.291810
ltd	1.717015	0.885701	1.938595	5.255069e-02	-0.018927	3.452957
debtgdp	-3.377012	0.811291	-4.162518	3.147576e-05	-4.967112	-1.786911
globaltloans	0.952151	2.056721	0.462946	6.434029e-01	-3.078948	4.983251
globalyieldCurve	-0.934300	0.166426	-5.613895	1.978227e-08	-1.260490	-0.608111
hpnom	0.749860	0.234076	3.203489	1.357731e-03	0.291079	1.208640

Run Experiments: Out-of-Sample

In [9]:
```
# Run Experiments: Macro Indicators
ex_macro.run(n = 100)
ex_macro.auc
```

```
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[9]:

	Model	AUC
0	RandomForest	0.710936
1	Logit	0.674463
2	NeuralNet	0.663469
3	ExtraTrees	0.655023
4	KNeighbors	0.606674
5	SVM	0.529537
6	Random Assignment	0.500000

In [10]:
```
# Run Experiments: Credit Indicators
ex_credit.run(n = 100)
ex_credit.auc
```

```
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[10]:

	Model	AUC
0	ExtraTrees	0.825846
1	RandomForest	0.822628
2	KNeighbors	0.795530
3	NeuralNet	0.768945
4	Logit	0.757818
5	SVM	0.751454
6	Random Assignment	0.500000

In [11]:
```
# Run Experiments: Credit & Asset Indicators
ex_ca.run(n = 100)
ex_ca.auc
```

```
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[11]:

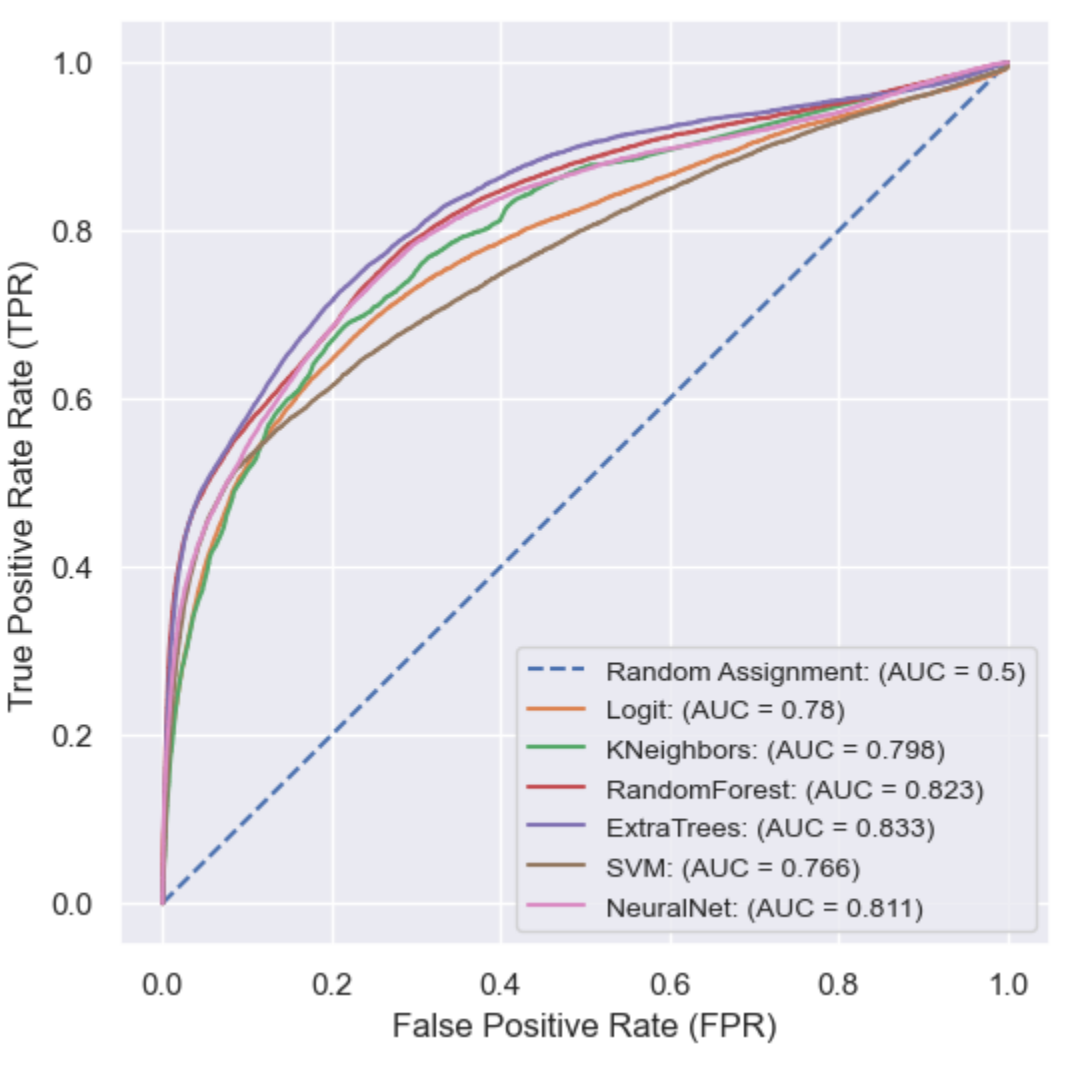
	Model	AUC
0	ExtraTrees	0.840335
1	RandomForest	0.832646
2	KNeighbors	0.803151
3	NeuralNet	0.773489
4	Logit	0.761357
5	SVM	0.756435
6	Random Assignment	0.500000

In [12]:
```
# Run Experiments: All Indicators
ex_all.run(n = 100)
ex_all.rocGraph()
ex_all.auc
```

```
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[12]:

	Model	AUC
0	ExtraTrees	0.833260
1	RandomForest	0.823036
2	NeuralNet	0.810856
3	KNeighbors	0.798073
4	Logit	0.779664
5	SVM	0.765645
6	Random Assignment	0.500000



Robustness Checks: Alternative Crisis Data

In [13]:
```
# Robustness Check: ESRB Crisis Data
df_alt1 = Data(iv_all, crisisData = "ESRB").getReady("ESRB")
ex_alt1 = Experiment(df_alt1, models, "CrossVal")
ex_alt1.run(n = 100)
ex_alt1.auc
```

```
ESRB: The final dataset contains 426 observations with 22 distinct crisis events.
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[13]:

	Model	AUC
0	RandomForest	0.896491
1	ExtraTrees	0.868925
2	KNeighbors	0.804878
3	Logit	0.759008
4	SVM	0.757480
5	NeuralNet	0.740183
6	Random Assignment	0.500000

In [14]:
```
# Robustness Check: Laeven & Valencia Crisis Data
df_alt1 = Data(iv_all, crisisData = "LaevenValencia").getReady("LaevenValencia")
ex_alt1 = Experiment(df_alt1, models, "CrossVal")
ex_alt1.run(n = 100)
ex_alt1.auc
```

```
Yugoslavia, SFR not found in regex
LaevenValencia: The final dataset contains 674 observations with 18 distinct crisis events.
Random Assignment:   0%|          | 0/100 [00:00<?, ?it/s]
Logit:   0%|          | 0/100 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/100 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/100 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/100 [00:00<?, ?it/s]
SVM:   0%|          | 0/100 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/100 [00:00<?, ?it/s]
```

Out[14]:

	Model	AUC
0	NeuralNet	0.886967
1	ExtraTrees	0.867728
2	KNeighbors	0.863598
3	RandomForest	0.849177
4	SVM	0.847194
5	Logit	0.836380
6	Random Assignment	0.500000

In [15]:
```
# Robustness Check: Strict Forecasting
ex_macroF = Experiment(df_macro, models, "Forecast")
ex_creditF = Experiment(df_credit, models, "Forecast")
ex_caF = Experiment(df_ca, models, "Forecast")
ex_allF = Experiment(df_all, models, "Forecast")
```

Robustness Checks: Strict Forecasting

In [16]:
```
# Forecast: Macro
ex_macroF.run()
ex_macroF.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[16]:

	Model	AUC
0	Logit	0.635322
1	NeuralNet	0.626200
2	KNeighbors	0.600879
3	RandomForest	0.534933
4	ExtraTrees	0.520512
5	Random Assignment	0.500000
6	SVM	0.427193

In [17]:
```
# Forecast: Credit
ex_creditF.run()
ex_creditF.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[17]:

	Model	AUC
0	NeuralNet	0.696083
1	KNeighbors	0.670350
2	SVM	0.647153
3	RandomForest	0.642819
4	ExtraTrees	0.619348
5	Logit	0.525632
6	Random Assignment	0.500000

In [18]:
```
# Forecast: Credit & Asset
ex_caF.run()
ex_caF.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[18]:

	Model	AUC
0	RandomForest	0.725951
1	KNeighbors	0.705793
2	ExtraTrees	0.673546
3	NeuralNet	0.660708
4	SVM	0.642525
5	Logit	0.583442
6	Random Assignment	0.500000

In [19]:
```
# Forecast: All Indicators
ex_allF.run()
ex_allF.auc
```

```
Random Assignment:   0%|          | 0/1 [00:00<?, ?it/s]
Logit:   0%|          | 0/1 [00:00<?, ?it/s]
KNeighbors:   0%|          | 0/1 [00:00<?, ?it/s]
RandomForest:   0%|          | 0/1 [00:00<?, ?it/s]
ExtraTrees:   0%|          | 0/1 [00:00<?, ?it/s]
SVM:   0%|          | 0/1 [00:00<?, ?it/s]
NeuralNet:   0%|          | 0/1 [00:00<?, ?it/s]
```

Out[19]:

	Model	AUC
0	NeuralNet	0.757596
1	Logit	0.671451
2	KNeighbors	0.651607
3	RandomForest	0.647569
4	ExtraTrees	0.594835
5	SVM	0.585685
6	Random Assignment	0.500000

In [ ]:

In [ ]: