IES

Integrated Energy Systems

Synthetic Time-Series
Leveraging Uncertainty in Market Behavior

FORCE Overview and Training

April 4-6, 2023

Dylan McDowell

Modeling & Simulation Engineer

Idaho National Laboratory

INL/MIS-22-65661

Idaho National Laboratory

Overview

- **What is a Synthetic Time-Series?**
- **Why do I need a Synthetic Time-Series?**
- **How do I produce a Synthetic Time-Series?**
  - The anatomy of an input file
  - Knobs & settings
  - Diagnosing model fit
  - Future work
- **Examples**
  - 2020 EPRI IL Study
  - 2021 EPRI NY Study
- **Q&A**

IES
Integrated Energy Systems

What is a Synthetic Time-Series?

- **Concept vs. Concrete**
  - A stochastic time-dependent signal (not necessarily ”historic”)
  - A serialized python object that RAVEN understands and can stochastically sample

- **For example:**
  - An ARMA model trained, using RAVEN, on hourly demand data 2010-2020
  - An ARMA model trained, using RAVEN, on hourly **forecasted demand projections** 2025-2050

- Many models might be categorized as conceptual synthetic time-series
- Only models trained using RAVEN are proper Synthetic History objects

IES
Integrated Energy Systems

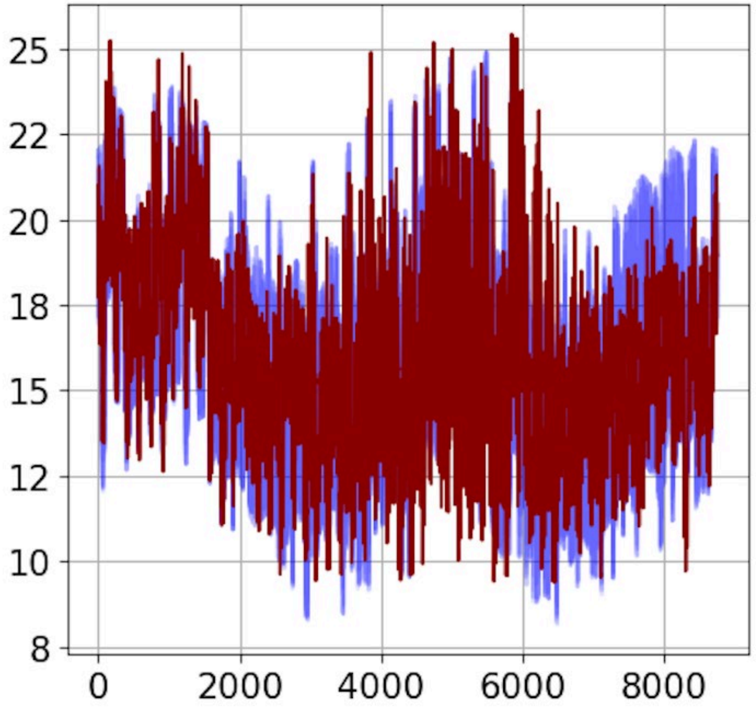
ARMA Trained Synthetic Time-Series

Fourier Detrending

$$x_t = y_t - \sum_m \{a_m \sin(2\pi f_m t) + b_m \cos(2\pi f_m t)\}$$

ARMA Model

$$x_t = c + \sum_{i=1}^{p} \phi x_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t$$



IES
Integrated Energy Systems

A Synthetic Time-Series is not…

- Model trained using python, R, excel, etc.
- Symbolic Expression
- Data from a model trained using above methods

Static CSVs are a drop-in replacement for Synthetic Time-Series

Why do I need a Synthetic History?

- "Golden Year Problem"
  - Common practice: solve optimal portfolio for single history
  - Fails to capture range of possible outcomes
  - Driving economics is in outlier scenarios
    - High demand/low VRE
    - Low demand/high VRE
    - Sudden ramping demand
    - Stressed storage usage
  - Historic or Forecasted single scenarios can't reliably capture outliers

IES
Integrated Energy Systems

How do I train a Synthetic History?

1. Preprocess time-series input data
2. Create a working directory containing:
   - CSV pointer file
   - CSV input data
3. Create a RAVEN XML input file
4. Run RAVEN via the command line
5. Diagnose output
6. Save PK files for use in HERON

Pointer CSV

	A	B
1	scaling	filename
2	1	Data0.csv
3		

Input Data

B	C
Time	Signal
0	0
0.1	0.06279052
0.2	0.12533323
0.3	0.18738132
0.4	0.24868989
0.5	0.30901699
0.6	0.36812455
0.7	0.42577929
0.8	0.48175367
0.9	0.5358268
1	0.58778525
1.1	0.63742399
1.2	0.68454711
1.3	0.72896863
1.4	0.77051324
1.5	0.80901699
1.6	0.84432793
1.7	0.87630668
1.8	0.90482705
1.9	0.92977649
2	0.95105652

IES Integrated Energy Systems

How do I train a Synthetic History?

Here is an example folder structure of a Synthetic History

- At least 3 files required to run RAVEN
- Several files are created during the training
- "*arma.pk*" is what will be used in HERON
- "*synth.csv*" is the evaluated output of the model
- It's typically easiest to start with a previous input file

IES
Integrated Energy Systems

The anatomy of an input file

- `Simulation` – contains all sub-nodes in the xml file
- `RunInfo` – contains working directory and steps
- `Files` – defines files that will be used as input during execution
- `DataObjects` – defines objects created during execution
- `Steps` – defines the steps used by RAVEN
- `Models` – contains all model parameters and knobs
- `OutStreams` – defines files that will be output after execution
- `Samplers` – contains sampler settings

IES
Integrated Energy Systems

The anatomy of an input file (Files & RunInfo)

- Working directory can be named anything
- 5 typical steps
- Make sure to specify pointer CSV
- Variable names are arbitrary, just be consistent

```xml
<?xml version="1.0" ?>
<Simulation verbosity="debug">

  <RunInfo>
    <WorkingDir>WorkingDir</WorkingDir>
    <Sequence>load, train, meta, serialize, sample</Sequence>
  </RunInfo>

  <Files>
    <Input name="input">Data.csv</Input>
    <Input name="pk">arma.pk</Input>
  </Files>
```

IES
Integrated Energy Systems

The anatomy of an input file (DataObjects)

- Similar to `Steps` & `Files`
- These objects will be used during execution
- Can be referenced anywhere in the input file by its name

```xml
<DataObjects>
  <PointSet name="placeholder">
    <Input>scaling</Input>
    <Output>OutputPlaceHolder</Output>
  </PointSet>

  <HistorySet name="input">
    <Input>scaling</Input>
    <Output>Signal, Time</Output>
    <options>
      <pivotParameter>Time</pivotParameter>
    </options>
  </HistorySet>

  <DataSet name="synthetic">
    <Input>scaling</Input>
    <Output>Signal</Output>
    <Index var="Time">Signal</Index>
    <Index var="Year">Signal</Index>
  </DataSet>
  <DataSet name="meta"/>
</DataObjects>
```

IES
Integrated Energy Systems

The anatomy of an input file (Steps)

- This section rarely changes across different scenarios and trainings
- Load, train, and serialize are required steps.
- Meta & sample are used in model diagnostics
- Advanced cases might require more steps

```xml
<Steps>
  <IOStep name="load">
    <Input class="Files" type="">input</Input>
    <Output class="DataObjects" type="HistorySet">input</Output>
  </IOStep>

  <RomTrainer name="train">
    <Input class="DataObjects" type="HistorySet">input</Input>
    <Output class="Models" type="ROM">arma</Output>
  </RomTrainer>

  <IOStep name="meta">
    <Input class="Models" type="ROM">arma</Input>
    <Output class="DataObjects" type="DataSet">meta</Output>
    <Output class="OutStreams" type="Print">romMeta</Output>
  </IOStep>

  <IOStep name="serialize">
    <Input class="Models" type="ROM">arma</Input>
    <Output class="Files" type="">pk</Output>
  </IOStep>

  <MultiRun name="sample">
    <Input class="DataObjects" type="PointSet">placeholder</Input>
    <Model class="Models" type="ROM">arma</Model>
    <Sampler class="Samplers" type="MonteCarlo">mc</Sampler>
    <Output class="DataObjects" type="DataSet">synthetic</Output>
    <Output class="OutStreams" type="Print">synthetic</Output>
  </MultiRun>
</Steps>
```

IES
Integrated Energy Systems

The anatomy of an input file (Models)

- Specify target and pivot parameters
- Specify P, Q, and Fourier
- Segmentation strategy can change formulation
- Specify clustering algo and number of clusters
- This node will change the most across different input files

```xml
<Models>

  <ROM name="arma" subType="ARMA">
    <Features>scaling</Features>
    <Target>Signal, Time</Target>
    <pivotParameter>Time</pivotParameter>
    <P>3</P>
    <Q>2</Q>
    <Fourier>10, 20, 25</Fourier>
    &lt;Segment grouping="interpolate">
      <macroParameter>Year</macroParameter>
      <Classifier class="Models" type="PostProcessor">classifier</Classifier>
      <subspace divisions="1">Time</subspace>
    &lt;/Segment&gt;
    <reseedCopies>False</reseedCopies>
    <seed>42</seed>
  </ROM>

  <PostProcessor name="classifier" subType="DataMining">
    <KDD labelFeature="labels" lib="SciKitLearn">
      <SKLtype>cluster|KMeans</SKLtype>
      <Features>Signal</Features>
      <n_clusters>10</n_clusters>
    </KDD>
  </PostProcessor>
</Models>
```

IES
Integrated Energy Systems

The anatomy of an input file (OutStreams)

- These are instructions that take in **`DataObjects`** and output them to the file system.

- Typical **`OutStreams`** can be:
  - CSV
  - XML
  - Plots (PNG)

```xml
<OutStreams>

  <Print name="romMeta">
    <type>csv</type>
    <source>meta</source>
  </Print>

  <Print name="synthetic">
    <type>csv</type>
    <source>synthetic</source>
  </Print>

</OutStreams>
```

IES
Integrated Energy Systems

The anatomy of an input file (Samplers)

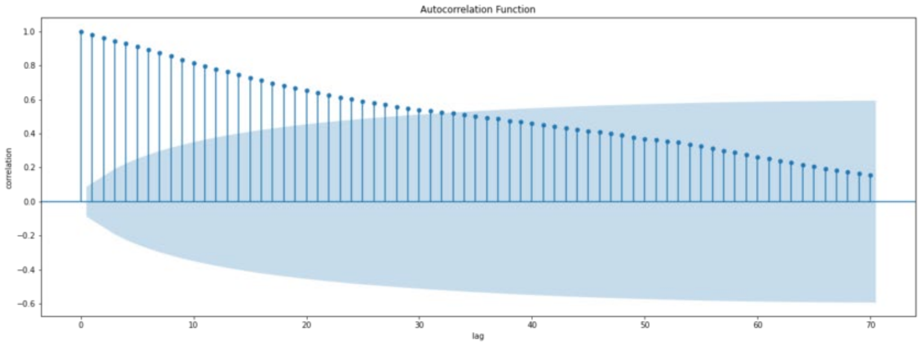
- Larger specified limit will provide better diagnostic information
- Will also increase generation and run-times

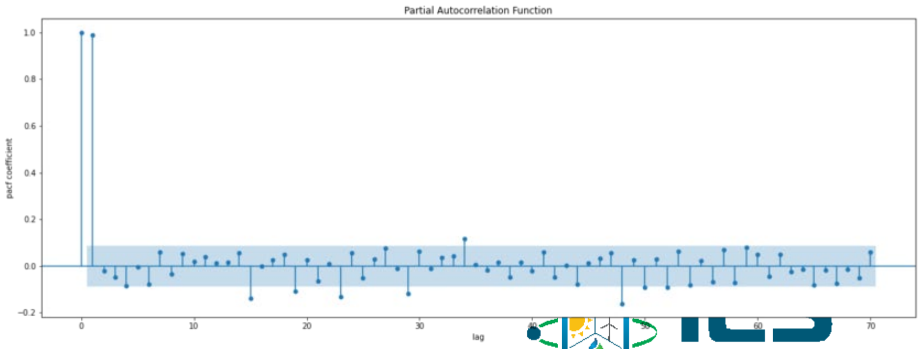
```
<Samplers>

  <MonteCarlo name="mc">
    <samplerInit>
      <limit>1</limit>
      <initialSeed>42</initialSeed>
    </samplerInit>
    <constant name="scaling">1.0</constant>
  </MonteCarlo>

</Samplers>
```

Knobs & Settings - Choosing P & Q

- Choose model order (P, Q) for ARMA generator
- Autocorrelation Function
- Partial-Autocorrelation Plots
- Choosing non-convergent P or Q can result in RAVEN errors
- Future work aims to improve making choices regarding P and Q

```
<ROM name="arma" subType="ARMA">
  <Features>scaling</Features>
  <Target>Signal, Time</Target>
  <pivotParameter>Time</pivotParameter>
  <P>3</P>
  <Q>2</Q>
```



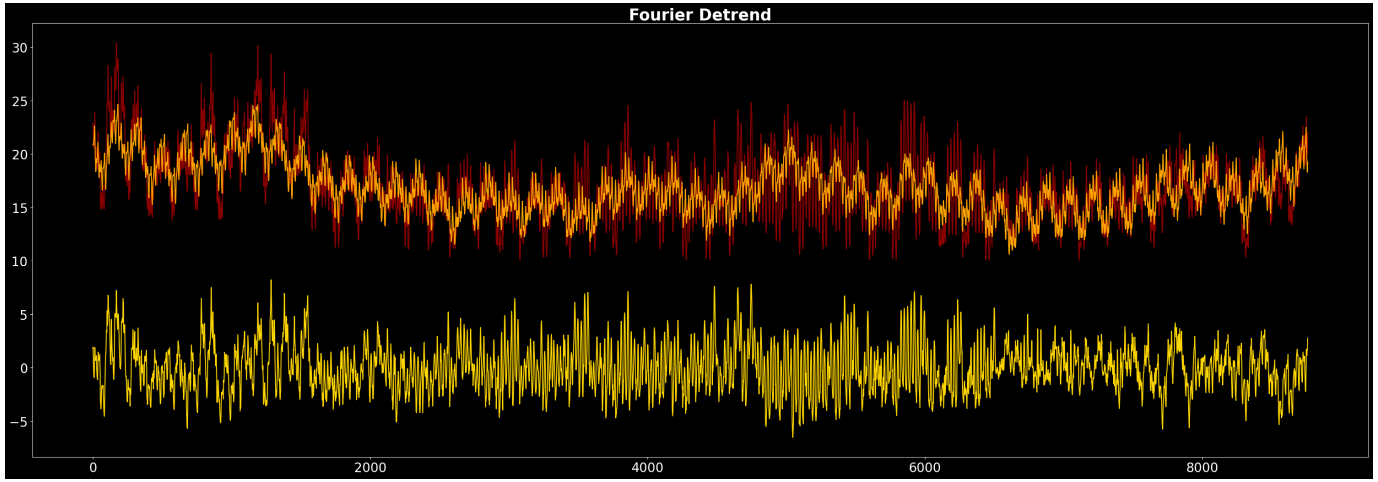


Knobs & Settings – Choosing Fourier Basis

- These numbers typically have a real-world analog
  - Hourly demand data might be: [8760, 4380, 168, 24, 12]
  - Hourly solar data might be: [24, 12]

- Applying a Fast Fourier Transform on your input data could be useful in determining signal components with the highest normalized amplitude.

```
<Fourier>10, 20, 25</Fourier>
&lt;Segment grouping="interpolate"&gt;
  <macroParameter>Year</macroParameter>
  <Classifier class="Models" type="PostProcessor">classifier</Classifier>
  <subspace divisions="1">Time</subspace>
&lt;/Segment&gt;
```



IES Integrated Energy Systems

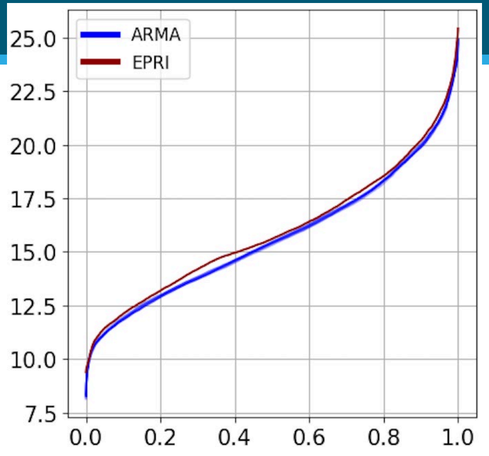
Knobs & Settings - Other

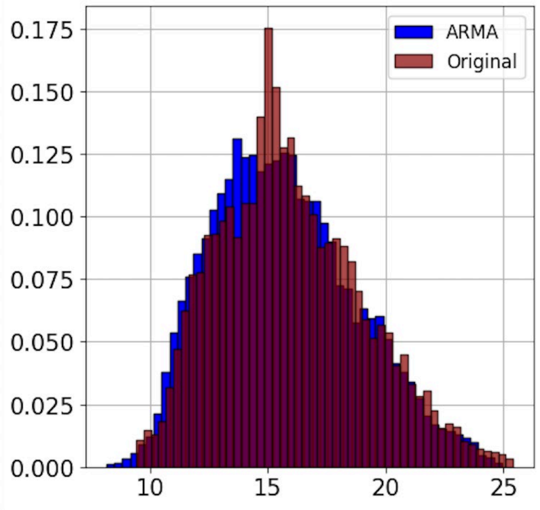
- Choosing larger/smaller number of clusters could have an impact on model fit.
- `<preserveInputCDF>` could sometimes be a useful option.
- Please refer to the raven user manual/guide for more information on Synthetic History settings.

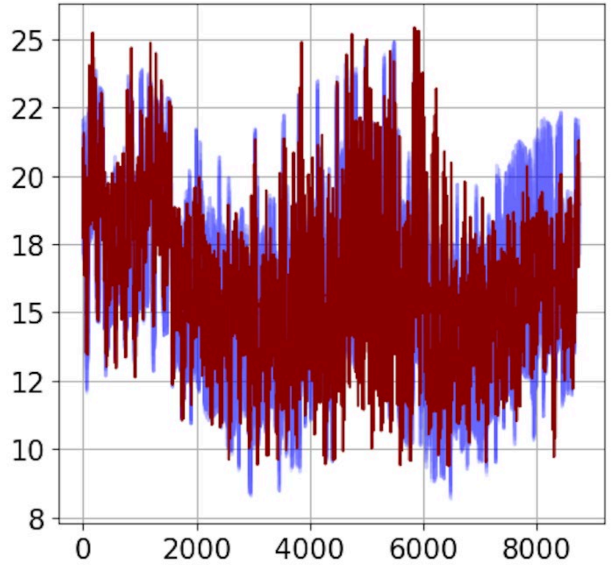
```
<PostProcessor name="classifier" subType="DataMining">
  <KDD labelFeature="labels" lib="SciKitLearn">
    <SKLtype>cluster|KMeans</SKLtype>
    <Features>Signal</Features>
    <n_clusters>10</n_clusters>
  </KDD>
</PostProcessor>
```

Diagnosing Model Fit

- More art than science
- Future work aims to provide better automated metrics
- Histograms, Load Duration Curve, Time Series Plots
- Cluster information can be found in romMeta.xml





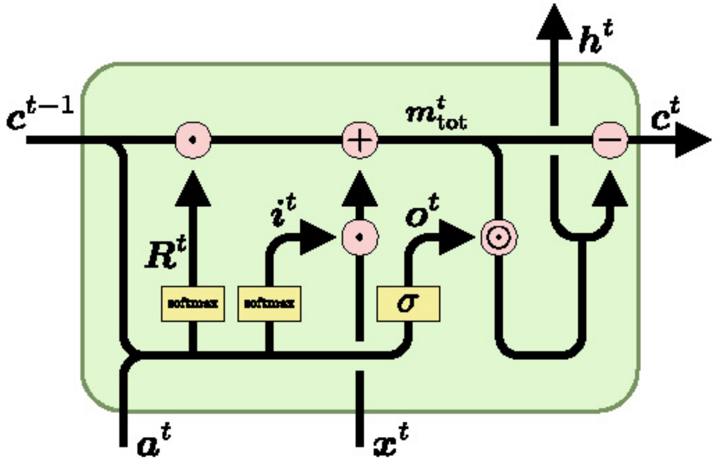


IES Integrated Energy Systems

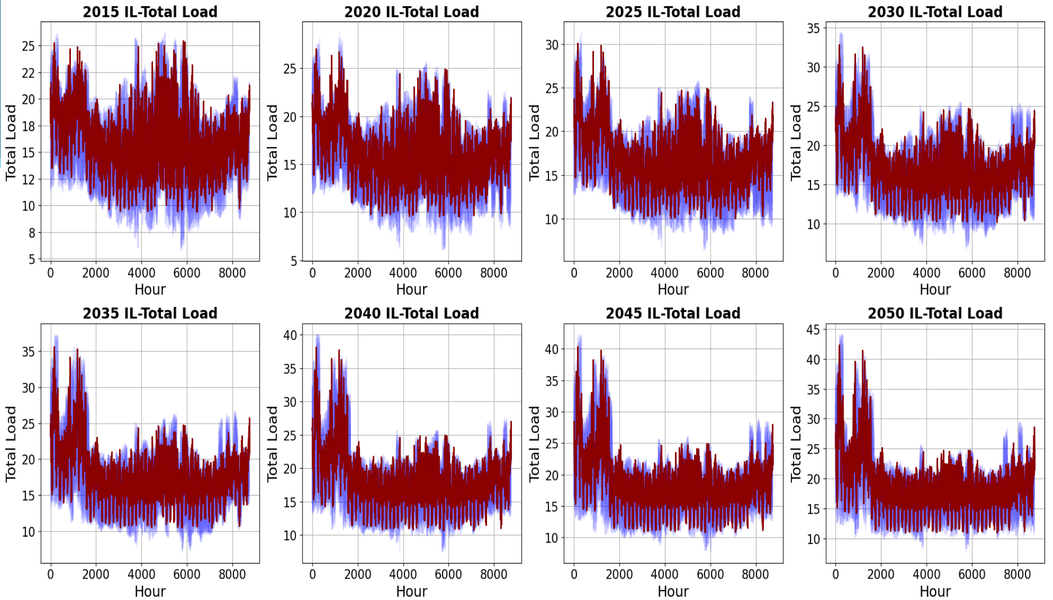
Future Work

We are now working on the following features:

- A variety of Synthetic History models
  - LSTM
  - GARCH
  - Etc.
- Interactive Synthetic History training via Jupyter Notebooks
- Improved model diagnostics



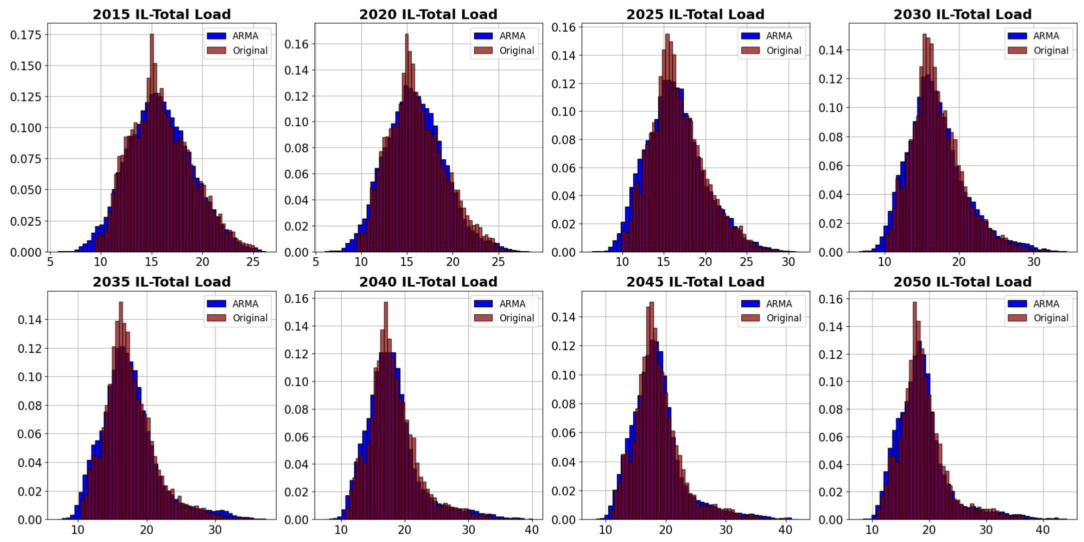
Example - 2020 EPRI IL Study

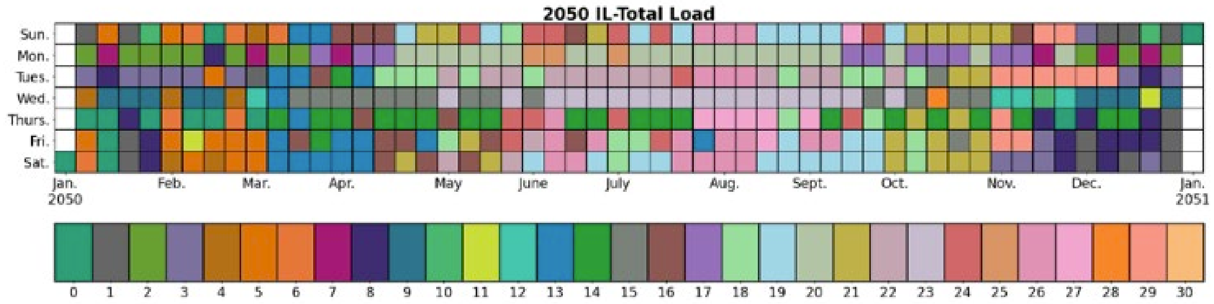


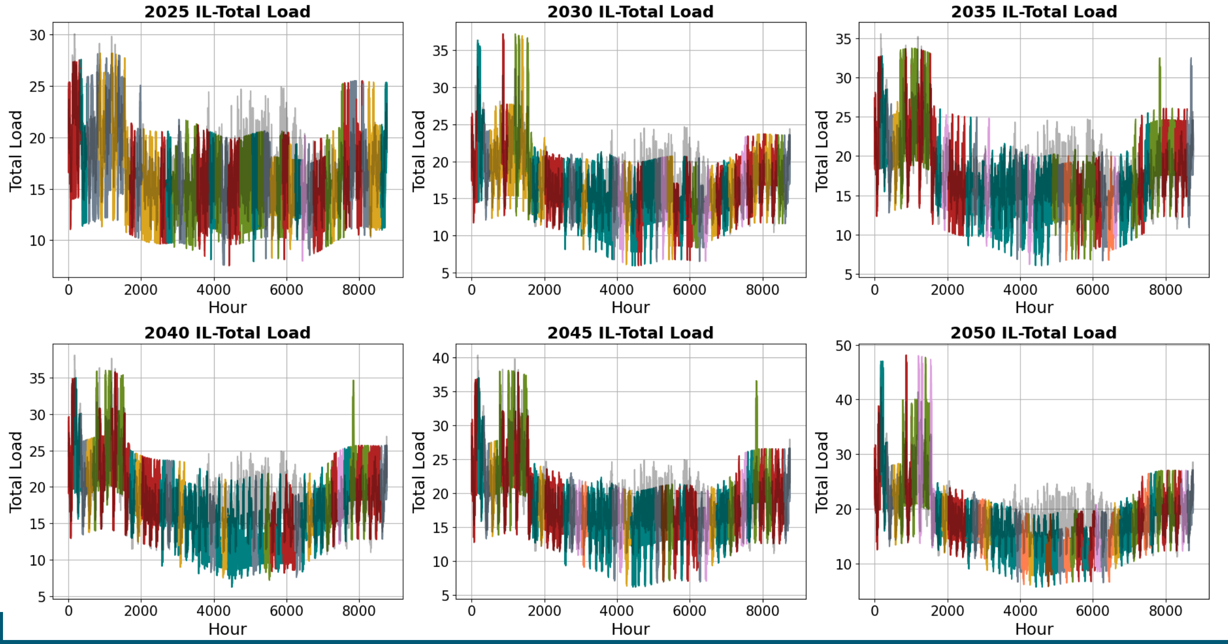
- Hourly demand projections 2015-2050
- Data had projections every 5 years (20’, 25’, 30’, 35’…)
- Required interpolation for years in-between

```xml
<Models>
  <ROM name="arma" subType="ARMA">
    <pivotParameter>HOUR</pivotParameter>
    <Features>scaling</Features>
    <Target>TOTALLOAD, HOUR</Target>
    <P>0</P>
    <Q>1</Q>
    <Fourier>8760, 4380, 2920, 2190, 1460, 1095, 584, 438, 168, 96, 24, 12, 8, 6</Fourier>
    &lt;Segment grouping='interpolate'>
      <macroParameter>YEAR</macroParameter>
      <Classifier class='Models' type='PostProcessor'>classifier</Classifier>
      <evalMode>full</evalMode>
      <subspace pivotLength='24' shift='zero'>HOUR</subspace>
      <evaluationClusterChoice>random</evaluationClusterChoice>
    &lt;/Segment&gt;
    <clusterEvalMode>clustered</clusterEvalMode>
    <reseedCopies>True</reseedCopies>
    <preserveInputCDF>False</preserveInputCDF>
    <seed>42</seed>
  </ROM>
  <PostProcessor name="classifier" subType="DataMining">
    <KDD labelFeature="labels" lib="SciKitLearn">
      <Features>TOTALLOAD</Features>
      <SKLtype>cluster|KMeans</SKLtype>
      <n_clusters>30</n_clusters>
      <tol>1E-12</tol>
      <init>k-means++</init>
      <random_state>3</random_state>
      <precompute_distances>True</precompute_distances>
    </KDD>
  </PostProcessor>
</Models>
```

Example – 2020 EPRI IL Study

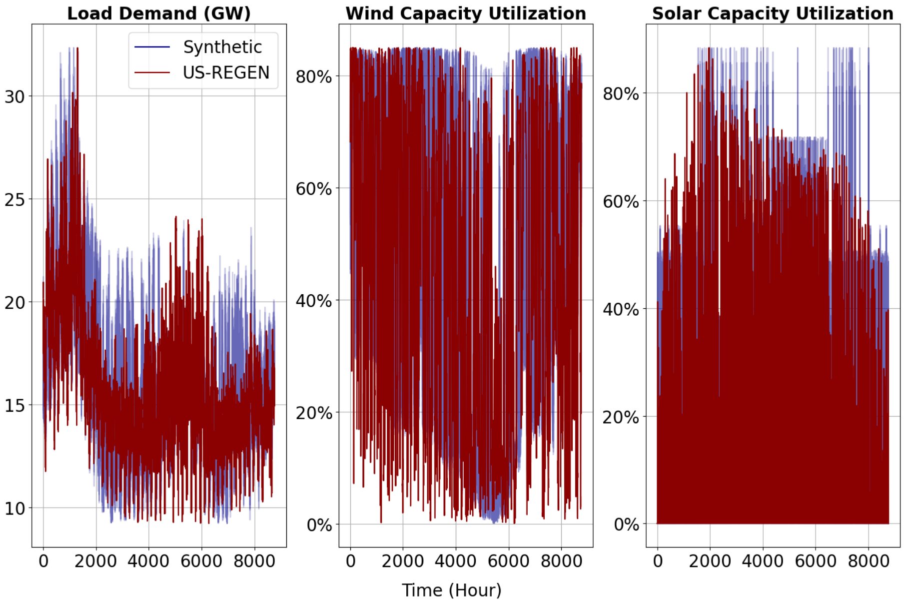






Example – 2021 EPRI NY Study

- Hourly load and wind/solar utilization
- Positive Truncation for solar
- Specific Fourier basis specification




```xml
<Models>
  <ROM name="arma" subType="ARMA">
    <pivotParameter>HOUR</pivotParameter>
    <Features>scaling</Features>
    <Target>TOTALLOAD,WIND,SOLAR,HOUR</Target>
    <P>1</P>
    <Q>0</Q>
    <Fourier>24, 12</Fourier>
    <SpecificFourier variables='TOTALLOAD,WIND'>
      <periods>8760, 4380, 2920, 2190, 438, 168, 24, 12, 6, 3</periods>
    </SpecificFourier>
    <ZeroFilter>SOLAR</ZeroFilter>
    <outTruncation domain='positive'>SOLAR</outTruncation>
    <preserveInputCDF>True</preserveInputCDF>
    <reseedCopies>True</reseedCopies>
    <seed>42</seed>
    &lt;Segment grouping="interpolate">
      <macroParameter>YEAR</macroParameter>
      <Classifier class="Models" type="PostProcessor">classifier</Classifier>
      <evalMode>full</evalMode>
      <subspace pivotLength="24" shift="zero">HOUR</subspace>
      <evaluationClusterChoice>random</evaluationClusterChoice>
    &lt;/Segment&gt;
  </ROM>
  <PostProcessor name="classifier" subType="DataMining">
    <KDD labelFeature="labels" lib="SciKitLearn">
      <Features>TOTALLOAD</Features>
      <SKLtype>cluster|KMeans</SKLtype>
      <n_clusters>8</n_clusters>
      <tol>1E-12</tol>
      <init>k-means++</init>
      <random_state>3</random_state>
      <precompute_distances>True</precompute_distances>
    </KDD>
  </PostProcessor>
</Models>
```

IES
Integrated Energy Systems

References

- McDowell, Dylan James, Talbot, Paul W, Wrobel, Anna Marie, Frick, Konor L, Bryan, Haydn C, Boyer, Chad, Boardman, Richard D, Taber, John, and Hansen, Jason K. *A Technical and Economic Assessment of LWR Flexible Operation for Generation and Demand Balancing to Optimize Plant Revenue*. United States: N. p., 2021. Web. doi:10.2172/1844211.
- Talbot, Paul W., McDowell, Dylan James, Richards, James D., Cogliati, Joshua J., Alfonsi, Andrea, Rabiti, Cristian, Boardman, Richard D., Bernhoft, Sherry, la Chesnaye, Francisco de, Ela, Erik, Hytowitz, Robin, Kerr, Chris, Taber, John, Tuohy, Aiden, and Ziebell, David. *Evaluation of Hybrid FPOG Applications in Regulated and Deregulated Markets Using HERON*. United States: N. p., 2020. Web. doi:10.2172/1755894.
- Talbot, Paul W., Rabiti, Cristian, Alfonsi, Andrea, Krome, Cameron, Kunz, M. Ross, Epiney, Aaron, Wang, Congjian, and Mandelli, Diego. *Correlated synthetic time series generation for energy system simulations using Fourier and ARMA signal processing*. United Kingdom: N. p., 2020. Web. doi:10.1002/er.5115.
- Chen, Jun, and Rabiti, Cristian. *Synthetic wind speed scenarios generation for probabilistic analysis of hybrid energy systems*. United States: N. p., 2016. Web. doi:10.1016/j.energy.2016.11.103.

FFT in RAVEN

```xml
<RunInfo>
  <WorkingDir>IL</WorkingDir>
  <Sequence>read,process</Sequence>
</RunInfo>

<Files>
  <Input name="inFile">FFT.csv</Input>
</Files>

<Models>
  <PostProcessor name="fft" subType="FastFourierTransform">
    <target>TOTALLOAD</target>
  </PostProcessor>
</Models>

<Steps>
  <IOStep name="read">
    <Input class="Files" type="">inFile</Input>
    <Output class="DataObjects" type="HistorySet">inData</Output>
  </IOStep>
  <PostProcess name="process" pauseAtEnd="False">
    <Input class="DataObjects" type="HistorySet">inData</Input>
    <Model class="Models" type="PostProcessor">fft</Model>
    <Output class="DataObjects" type="HistorySet">fft_out</Output>
    <Output class="OutStreams" type="Print">fft_final</Output>
  </PostProcess>
</Steps>
```

```xml
<OutStreams>
  <Print name="fft_final">
    <type>csv</type>
    <source>fft_out</source>
  </Print>
</OutStreams>

<DataObjects>
  <HistorySet name="inData">
    <Input>placeholder</Input>
    <Output>TOTALLOAD, HOUR</Output>
    <options>
      <pivotParameter>HOUR</pivotParameter>
    </options>
  </HistorySet>
  <HistorySet name="fft_out">
    <Output>TOTALLOAD_fft_period,TOTALLOAD_fft_amplitude</Output>
    <options>
      <pivotParameter>TOTALLOAD_fft_frequency</pivotParameter>
    </options>
  </HistorySet>
</DataObjects>
```

IES Integrated Energy Systems