PI World 2019 Lab

Exploring AF Analytics for Advanced Analysis and Prediction



OSIsoft, LLC 1600 Alvarado Street San Leandro, CA 94577

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of OSIsoft, LLC.

OSIsoft, the OSIsoft logo and logotype, Managed PI, OSIsoft Advanced Services, OSIsoft Cloud Services, OSIsoft Connected Services, OSIsoft EDS, PI ACE, PI Advanced Computing Engine, PI AF SDK, PI API, PI Asset Framework, PI Audit Viewer, PI Builder, PI Cloud Connect, PI Connectors, PI Data Archive, PI DataLink, PI DataLink Server, PI Developers Club, PI Integrator for Business Analytics, PI Interfaces, PI JDBC Driver, PI Manual Logger, PI Notifications, PI ODBC Driver, PI OLEDB Enterprise, PI OLEDB Provider, PI OPC DA Server, PI OPC HDA Server, PI ProcessBook, PI SDK, PI Server, PI Square, PI System, PI System Access, PI Vision, PI Visualization Suite, PI Web API, PI WebParts, PI Web Services, RLINK and RtReports are all trademarks of OSIsoft, LLC.

All other trademarks or trade names used herein are the property of their respective owners.

U.S. GOVERNMENT RIGHTS

Use, duplication or disclosure by the US Government is subject to restrictions set forth in the OSIsoft, LLC license agreement and/or as provided in DFARS 227.7202, DFARS 252.227-7013, FAR 12-212, FAR 52.227-19, or their successors, as applicable.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording or otherwise, without the written permission of OSIsoft, LLC.

Published: March 22, 2019

Table of Contents

Contents

Table of Contents
Course Learning Objectives and Toolkit5
Problem Statement5
Learning Objectives5
Supporting PI Infrastructure
Software Toolkit6
Exercise 1 - Develop Predictive Model for a Single Transformer8
Learning Objectives
Create an Asset View – PI Integrator for BA8
Visual Assessment of Transformer Data 15
Python Development Environment
Jupyter Notebooks
Develop the Model in Python
Test Model - AF Analytics Results Preview 22
Evaluate Model – Power BI
Exercise 2 - Develop Predictive Models for Multiple Transformers27
Learning Objectives
Create a PI View – PI SQL Client
Develop the Model in Python
Test Model - AF Analytics Backfill
Evaluate Model – PI Vision Collection 41
Example 3 - Operationalize Forecast Models for Multiple Transformers
Learning Objectives
Create a Forecast from the Predictive Model 48
Appendix A - Python Access to PI using PI OLEDB Enterprise
Create a PI View – PI OLEDB Enterprise
Develop the Model in Python
Appendix B - Python Access to PI via PI Web API65
Develop the Model in Python

Appendix C – Learning Resources	72
Save the Date!	75

Course Learning Objectives and Toolkit

Problem Statement

Secondary transformers deliver electric power to homes and businesses. You have probably seen one on a pole or pad in your own neighborhood. Utilities monitor energy loads on these transformers because exceeding designed capacity can cause them to fail, resulting in loss of service or liability from explosion. Although secondary transformers are not instrumented, their energy loads can be computed using AF analytics to sum up readings taken from meters placed at each service delivery point they supply. This summing up, or rollup, analysis has been completed for the sixty transformers we will be analyzing in this lab.

Since transformer load monitoring is important, our objective will be to develop a simple statistical model for predicting the power load on each of the sixty transformers operating on one circuit of our power distribution network. The predictive model will be operationalized to forecast network loading to anticipate loading problems before they occur. The model will be developed from utility metering and weather forecast data stored in the PI archive and contextualized in PI Asset Framework (AF).

Learning Objectives

AF Analytics offers distinctive advantages when preparing time-series data for advanced analytics. It can play an essential role in testing, evaluating and operationalizing developed models. In addition, the openness of the PI System supports several different data access methods, meeting the needs of data engineers or scientists.

In this lab, we'll complete three exercises demonstrating how AF Analytics can serve as an advanced analytics workbench. For comparison, we will employ different data access methods, PI Integrator for Business Analytics and PI SQL Client. (Exercises are replicated in the Appendix using PI OLEDB Enterprise and the PI Web API.) Predictive models using PI Data will be developed in Python, a readily available open source application that is widely used for statistical modelling and analysis.

The lab consists of three exercises. The same statistical model is developed in each exercise, however different features of the PI Infrastructure are used in each case.

- Exercise 1 Develop Predictive Model for a Single Transformer
 - Data preparation PI Integrator for Business Analytics (BA).
 - Model testing AF Analytics "Preview Results" No PI Tags required
 - Model evaluation Microsoft Power BI
- Exercise 2 Develop Predictive Model for Multiple Transformers
 - o Data preparation PI SQL Client
 - Model testing AF Analytics Backfill PI Tags
 - required o Model evaluation PI Vision Collections

- **Exercise 3** Operationalize Forecast Model for Multiple Transformers
 - Operationalization AF Analytics, PI Future Data and PI Vision

Additional data access methods and learning resources are described in the appendix.

- Appendix A Example 2 using PI OLEDB Enterprise
- Appendix B Example 2 using PI Web API
- Appendix C Learning Resources

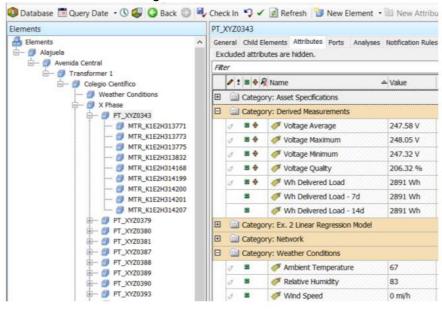
Supporting PI Infrastructure

The distribution network for the **Avenada Central** substation in the city of Alajuela, Costa Rica is represented in the PI AF hierarchy below. The AF model provides required context, data preparation, and operationalization support for statistical modelling projects. We will be focusing on the three-phase (X, Y, Z) **Colegio Cientrifico** circuit which has sixty pole transformers.

Our modelling effort requires data from the pole transformer (PT_*) level of the AF hierarchy, e.g. **PY_XYZ20343**. Meter data (MTR_*) has been loaded into the PI Server for the three-month period, 6/1/17 to 8/31/17. An AF Rollup analytic was used to compute pole transformer power loads by summing the attribute **Wh Delivered Load**, for the meters connected to it.

Additional AF attributes have been configured as relevant modelling features. These provide historical values for the Wh Delivered Load at exactly 7 and 14 days prior to the current time (*Wh Delivered Load – 7d* and *Wh Delivered Load – 14d*, respectively).

Weather conditions in Alajuela are also available in the AF Model and will be used in our analysis. For this lab, we will be assuming today is 8/31/17. The PI System contains historical weather data for ambient temperature, relative humidity and wind speed from 6/1/17 to 8/31/17, as well as, hourly weather forecast for the upcoming month of September 2017. These values have been stored in PI Future data tags.



Software Toolkit

- PI Server 2018 (with the optional RTQP Engine installed to support PI SQL Client)
- PI Integrator for Business Analytics
- PI SQL Client 2018
- PI OLEDB Enterprise 2017 R2
- PI Web API
- PI Vision
- Python and required package within Jupyter Notebook
- Microsoft Power BI

Exercise 1 - Develop Predictive Model for a Single Transformer

Learning Objectives

- Export transformer load and weather data into a text file using the PI Integrator for Business Analytics
- Examine the transformer dataset to gain a better understanding of loading characteristics
- Develop load prediction model for a single transformer using Python, in Jupyter Notebooks
- Test the model using AF Analytics "Results Preview" No PI tags required.
- Assess model results in MS Power BI

Create an Asset View – PI Integrator for BA

From the Windows taskbar, open AF Explorer and review the "Distribution Network" database. We will be using the PI Integrator for BA to create an asset view publication based on this asset model. The publication target will be a text file posted in the subdirectory, **C:\Users\student01\Python Lab** (this is the "Exploring AF for Advanced Analytics" folder shortcut on the desktop). This text file will provide data for creating the predictive model in Python.

- a. Open the PI Integrator for Business Analytics from the desktop shortcut.
- b. Click **Create Asset View** on the top menu, name your view *My Pole Transformer Loads* and click **Create View**.



c. On the Select Data page, select the **Distribution Network** database. Expand the AF Hierarchy to the circuit level and select element **Colegio Científico**. Drag and drop this element into the Search Shape area in the middle of the screen under Asset Shape.

Select Data >	Modify View >	Publish	
O Source Assets			
Server	PISRV01		~
Database	Distribution Network		S 💌
Assets			
🔺 😭 Alajuela			
🔺 💮 Avenida Ce	ntral		
A 💮 Transform	ner 1		
► 😭 Coleg	io Científico		

d. Expand the hierarchy to expose the phase level and select element **X Phase**. Drag and drop this element into the Search Shape area under the **Colegio Científico** element.

Source Ass	ets	
Server	PISRV01	~
Database	Distribution Network	S 🗸
Avenid:		
S	Weather Conditions	
¢	Weather Conditions	

e. Expand the hierarchy to expose the transformer level, select element **PT_XYZ0343.** Drag and drop the element into the Search Shape area under the **X Phase** element.

Select Data	> Modify View > Publish	
💮 Source Ass	ets	
Server	PISRV01	~
Database	Distribution Network	S 💌
© • ©	sformer 1 olegio Científico I Weather Conditions I X Phase	
•	PT_XYZ0343	
•	PT_XYZ0379	
•	PT_XYZ0380	

f. Use the sort button next to the Attributes filter to group the transformer attributes list by category.



g. Click the **Asset Specifications** category and select the attributes shown below. Drag and drop the group into the Search Shape area under the element **PT_XYZ0343**.

Attributes Filter	x [17]
Asset Specifications	
Rated KVA	0
Maximum KVA	θ
Loading	θ
Asset ID	θ

h. Click the **Derived Measurements** category and select the attributes shown below. Drag and drop the group into the Search Shape area under the element **PT_XYZ0343**.

Derived Measurements	
Voltage Quality	θ
💣 Voltage Minimum	0
🔗 Voltage Maximum	0
Voltage Average	0
🔗 Wh Delivered Load - 7d	θ
🔗 Wh Delivered Load - 14d	0
🔗 Wh Delivered Load	0

i. Click the **Weather Conditions** category and select the attributes shown below. Drag and drop the group into the Search Shape area under the element **PT_XYZ0343**.

Weather Conditions	
Wind Speed	0
Relative Humidity	0
Ambient Temperature	0

j. The steps above define an Asset Shape for a Single Phase Transformer element along with its selected attributes.

At this point, only one element is returned in the Match area because only one transformer element, PT_XYZ0343, has been specified. We need to leverage our AF Model to obtain data for all the assets we want to include in our analysis.

🖥 Search Shape	
🖥 Asset Shape	
Colegio Científico	/ ×
X Phase	/ ×
PT_XYZ0343	/ ×
Ambient Temperature	/ x
E Loading	/ x
Maximum KVA	/ ×
Rated KVA	/ ×
Relative Humidity	/ ×
Wh Delivered Load	/ x
Wh Delivered Load - 14d	/ ×
Wh Delivered Load - 7d	/ ×
Wind Speed	/ ×

Tip

k. Click the edit pencil next to element PT_XYZ0343 and modify the shape filter. Uncheck the Asset Name and check Asset Template. Select the Single Phase Transformer template to include all transformers under the X Phase of the circuit. The Matches area should show a list of 16 elements.

 Colegio Científico 	, A
X Phase	
▲	1
Edit Filters	×
□ Asset Name	
PT_XYZ0343	
🖌 Asset Template	
Single Phase Transformer	v
Asset Category	
Single Phase	
Add Filter	
	Cancel Save

I. Click the edit pencil next to the element **X Phase** and modify the shape filter to include all phases derived from the **Phase** template. (Remember to uncheck the **Asset Name**.)

Edit Filters	3
Asset Name X Phase	
Asset Template	V
Asset Category	V
⊕ Add Filter	
	Cancel Save

m. With the shape selected and filters set, you should see all sixty transformers listed int he Matches area.

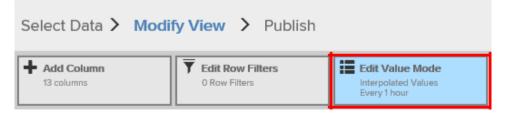


n. Click the **Next** button in the upper right corner to proceed to the Modify View page.

o. The Modify View page shows a preview of the first 100 rows from the first 10 matches of the asset shape. Set the time range, 6/15/17 to 8/31/17, to match the available PI data for the lab. (Since the Wh Delivered -14d attribute returns values from two weeks ago, we must start the request on 6/15/17 instead of 6/1/17.) Click Apply.

Start Time	End Time	
6/15/17	8/31/17	

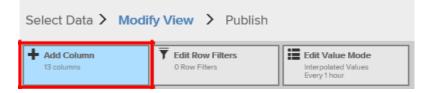
p. Click Edit Value Mode at the left side of the top menu.



Leave the Sampled Values setting defaulted to return interpolated values. However, use the drop-down menu to select hours instead of minutes samples for each row. Click **Save Changes**.

Edit Value Mode			×
Sampled Values			
Sample values every 1	- minutes -		
Interpolate	seconds		
O Exact 🕄	minutes		
O Use Key Column Ambie	nt T hours		
	days		
		Cancel	Save Changes

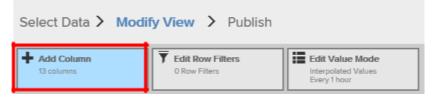
q. Click Add Column to add several time columns to the asset view.



Select the **Time Column** tab and add the columns shown below (the TimeStamp column is always included by default). Click **Display 4 time columns**.

Add Column	×
Data Column Time Column	
Select Time Column Options for Local 🗸	
Year (2018)	TimeStamp (Local)
Month (9)	Month Name (Local)
Week of the Year (38)	Day of the Week (Local)
Day (19)	Hour (Local)
Minute (57)	I I I I I I I I I I I I I I I I I I I
Second (13)	7
Milliseconds (850)	
UTC Seconds (1537379833.85)	
UTC Milliseconds (1537379833850)	
Ticks (636729766338500000)	
Time Zone Offset (420)	
	Cancel Display 4 time columns

r. Click Add Column again, this time to illustrate a point.



Select the **Data Column** tab. Select the **Wh Delivered Load** attribute and change the Column Name to **Wh Delivered Load Offset**. Click **Add Column**.

Add Column	x
Data Column Time Column	
Select Column Data Source	
🔺 🛱 Colegio Científico	
🔺 🕎 Phase	Wh Delivered Load Offset ×
 Single Phase Transformer 	Use Default Name
Ambient Temperature	Column Data Content 🚱
🔳 Loading	Value Value
🔳 Maximum KVA	Value
Rated KVA	Unit of Measure
Relative Humidity	watt hour 💙
I Wh Delivered Load	Data Type
Wh Delivered Load - 14d	Double
Wh Delivered Load - 7d	
Wind Speed	
	Cancel Add Column

s. Select the newly added **Wh Delivered Load Offset** column in the table, set its **Column Offset** to be minus 2 hours and click **Apply Changes**.

My Pole Transformer Loads	1		Olumn Details
Start Time		End Time	Name Wh Delivered Load Offset
6/15/17	[8/31/17	Reset Name to Default
Wh Delivered Load	Wh Delivered Load Offset	Wh Delivered Load - 14d	Data Content 3
20,953		13,643	Value 🗸
19,727		12,872	
19,281	20,953	12,481	Column Offset 📀
17,885	19,727	13,016	- -2 +
17,857	19,281	13,419	
17,343	17,885	11,338	Unit of Measure
17,839	17,857	11,488	watt hour
19,708	17,343	13,586	
21,678	17,839	15,010	Data Type
20,293	19,708	17,612	Double 🗸
24,100	21,678	14,573	
22,529	20,293	14,905	
23,287	24,100	14,232	Remove Column
26,313	22,529	15,433	
25,659	23,287	14,246	Apply Changes

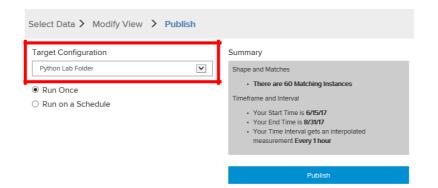
The PI Integrator for BA will return null values for requested values falling outside the shape's time range. Fortunately, we have configured AF attributes to return Wh Delivered Load values for 7 and 14 days ago.

- t. Select the **Wh Delivered Load Offset** column in the table again. Click **Remove Column**, in its Column Details, we're not going to need this.
- u. Click **Next** to proceed to the Publish page.

			Back	Next	
Start Time		End Time			
6/15/17	[000. 	8/31/17	(000) (****	Apply	
ek	Hour	Phase	Single Phase Transfor	mer	•=
0		X Phase	PT_XYZ0343	68	~
1		X Phase	PT_XYZ0343	68	

Tip

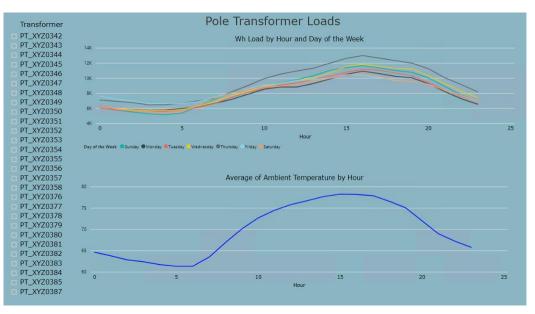
v. Select **Python Lab Folder** as the Target Configuration and click **Publish**, then **Confirm** to publish the data as a text file.



The file name will be the name of the asset view, My Pole Transformer Loads, and it will be in the subdirectory, C:\Users\student01\Python Lab (this is the "Exploring AF for Advanced Analytics" folder shortcut on the desktop). This target was configured using the Administration pages of the PI Integrator for BA.

Visual Assessment of Transformer Data

From the desktop, open the Power BI report called "Example 1 Data Analysis". The Pole Transformer Loads text file has been imported into this report to allow visual inspection of all sixty transformer load profiles. The profiles are presented for the 24 hours of the day and broken down by the days of the week. Look at transformers, 342, 358, and 416, to observe some of the differences than can exist. Obviously, we will eventually need to build separate models for each transformer.



Python Development Environment

There are many open-source packages that support Python programming. In this lab, we'll use the analysis environment contained in the Anaconda distribution of Python. Also included in this distribution is Jupyter Notebooks. We will be using Jupyter Notebooks as our Integrated Development Environment (IDE) for the Python work. It is a browser-based environment which makes it easy to manage program development. This lab contains the following key components:

- Anaconda 5.1 distribution of Python 3.6
 - The Anaconda distribution of Python contains <u>hundreds of commonly used</u> <u>packages</u> to support python application development and data science
 - o <u>Jupyter</u>: An interactive notebook useful for annotating python code with graphics, documentation, or tutorial instructions
- Python packages utilized in this lab included in Anaconda 5.1
 - o Numpy: Highly optimized numerical library useful for linear algebra
 - o Pandas: Data management library for DataFrames, similar to R DataFrames
 - o Matplotlib: Visualization library that feels similar to Matlab based plotting
 - o Seaborn: Visualization library with a focus on statistics
 - o Scikit-Learn: Machine Learning library
 - o json, requests, urllib.parse, requests_kerberos: Support access to PI Web API
- Python packages utilized in this lab not included in anaconda 5.1
 - <u>adodbapi</u>: toolkit for interacting with MS SQL and PI via the PI SQL Client and PI OLEDB

Jupyter Notebooks

Launch Jupyter Notebook by clicking once on the <u>see</u> icon in the Windows taskbar. You will see a command window appear - <u>keep this open (minimized) throughout the lab</u>. Eventually, Internet Explorer will open exposing a list of files in Jupyter's root directory. Click the link to the **Python Lab** folder.

💭 jupyter	Logout
Files Running Clusters	
Select items to perform actions on them.	Upload New - 2
	Name 🔶 Last Modified
Contacts	a month ago
Desktop	3 days ago
Documents	23 days ago
Downloads	15 days ago
Favorites	a month ago
Library	8 days ago
Links	a month ago
Ch Music	a month ago
C Pictures	a month ago
C Python Lab	24 minutes ago
Cames	a month ago
Searches	a month ago
C Videos	a month ago
Dryer.png	6 months ago

The Python Lab folder contains the Python scripts files used the lab examples. We will be returning to it several times. Keeping the browser open will save time in accessing them during the session.

💭 jupyter	Logout
Files Running Clusters	
elect items to perform actions on them.	Upload New - 2
🖸 0 👻 🖿 / Python Lab	Name 🕹 Last Modified
D	seconds ago
🗌 🛢 Load Prediction - Example 1.ipynb	19 hours ago
Load Prediction - Example 2 (PI OLEDB Enterprise).ipynb	21 hours ago
🗌 🛢 Load Prediction - Example 2 (PI Web API).ipynb	28 minutes ago
Load Prediction - Example 2.(PI SQL Client).ipynb	21 hours ago

For this lab, you will simply need to know how to step through the Python scripts in a cell by cell fashion. In Jupyter, this is accomplished by hitting **Shift+Enter** while focused on a cell. This will execute the current cell and advance to the next cell. As you execute each cell in the

notebook, wait for the Python's busy light to switch from busy, Python 3 • to ready Python 3 •. This indication is in the upper right-hand corner of the Jupyter Notebook. If this is your first time using Jupyter Notebooks or simply want to learn more, the Jupyter learning resources listed in Appendix C may be helpful.

Develop the Model in Python

Tip

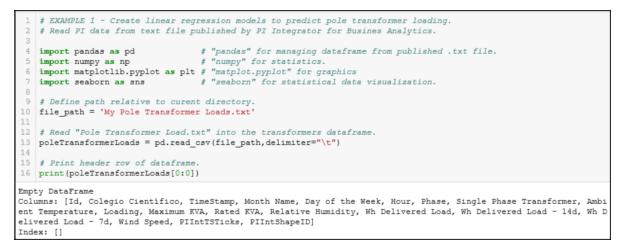
The following code creates a multivariable linear regression model for one transformer. We will import the transformer data from the text file published by the PI Integrator for B A. Through a simple cut and paste operation, we will place the model's equation into AF Analytics and leverage the AF Analysis Service to test the model. Test results will be exported from AF Analytics and evaluated using Power BI.

To run the code with your PI Asset View, you'll want to replace "Pole Transformer Loads" with "My Pole Transformer Loads" in the first cell of this Jupyter Notebook

- b. Edit the file_path assignment statement in the first cell to be 'My Pole Transformer Loads.txt'

a. From Jupyter, open the Python script, Load Prediction - Example 1.

c. Import transformer data from the text file, "My Pole Transformer Loads.txt" published by the PI Integrator for Business Analytics. Select the first cell and type Shift+Enter. It may take a few seconds for the file to be loaded.



The column headers returned should match the ones configured in the PI Asset View. The columns "Id", "PIIntTSTicks" and "PIIntShapeID" have been added by the PI Integrator for BA and are for internal use.

d. Rename columns with shorter names. Create a smaller dataframe, modellingData, containing only the values need for predicting transformer loads. Select the next cell and type Shift+Enter.

2 poleTran 3 poleTran 4 poleTran 5 poleTran 6 poleTran 7 poleTran 9 10 <i># Define</i> 11 modellin 12 13 14 <i># Peek a</i>	sformerLoads.ren sformerLoads.ren sformerLoads.ren sformerLoads.ren sformerLoads.ren sformerLoads.ren sscond datafram gData = poleTran	ame (c ame (c ame (c ame (c ame (c ame (c ame (c e wit sform d', 'W	<pre>olumns = { olumns = { olumns = { olumns = { olumns = { olumns = { olumns = { olumns = { olumns = { rLoads[[' h Load', '' } } } }</pre>	'Single 'Ambient 'Relativ 'Wh Deli 'Wh Deli 'Wh Deli 'Wind Sp a needed Transfor Wh Load-	Phase Temp ve Hur verec verec verec verec verec i i i i i i i i i i i i i i i i i i i	e Transf perature midity': d Load': d Load - d Load - :'Wind'} <i>our mod</i> , 'TimeS 'Wh Loa	ormer':'Tr ':'Tempera 'Humidity' 'Wh Load') 14d':'Wh I 7d':'Wh I , inplace 'elling. tamp', 'Ho d-14d']]	<pre>ansformer' ture'}, in }, inplace , inplace Load-14d'} Goad-7d'}, = True)</pre>	= True) = True) , inplace = True)
Transformer	TimeStamp	Hour	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d	
0 PT_XYZ0343	6/15/2017 12:00:00 AM	0	68	93.0	7.0	6529.0	3278.0	2464.0	
1 PT_XYZ0343	6/15/2017 1:00:00 AM	1	68	93.0	6.0	6413.0	3122.0	2798.0	
2 PT_XYZ0343	6/15/2017 2:00:00 AM	2	68	93.0	6.0	6781.0	2658.0	2490.0	
3 PT_XYZ0343	6/15/2017 3:00:00 AM	3	68	90.0	0.0	5348.0	2873.0	2722.0	
4 PT_XYZ0343	6/15/2017 4:00:00 AM	4	67	93.0	0.0	6916.0	3306.0	2698.0	

The new dataframe, **modellingData**, contains only data needed for building the predictive model.

e. Examine the last five rows of the dataframe using the **tails()** method. Select this cell and hit **Shift+Enter**.

	<pre># Peek at the last five rows to make sure ve got them all. modellingData.tail()</pre>											
	Transformer	Time Stamp	Hour	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d			
110935	PT_XYZ0414	8/30/2017 8:00:00 PM	20	68	73.0	7.0	69.0	68.0	44.0			
110936	PT_XYZ0414	8/30/2017 9:00:00 PM	21	64	81.0	3.0	68.0	68.0	45.0			
110937	PT_XYZ0414	8/30/2017 10:00:00 PM	22	66	75.0	3.0	68.0	69.0	45.0			
110938	PT_XYZ0414	8/30/2017 11:00:00 PM	23	65	78.0	3.0	67.0	68.0	46.0			
110939	PT_XYZ0414	8/31/2017 12:00:00 AM	0	62	86.0	3.0	68.0	69.0	44.0			

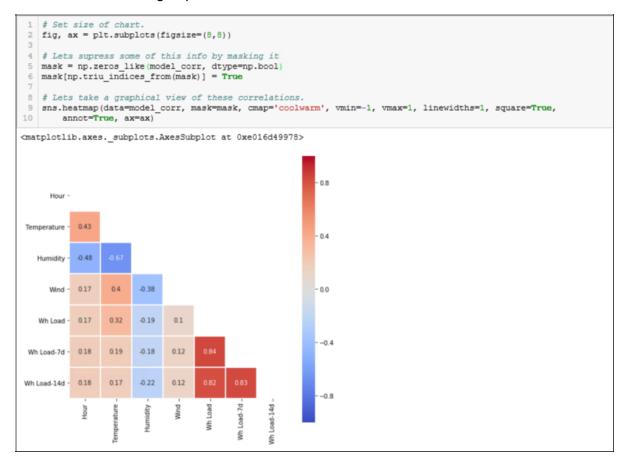
f. Gain a simple statistical perspective of the values in our dataframe. Select this cell and hit **Shift+Enter**.

		y <i>statisti</i> describe()	<i>cal perspe</i>).T	ctiv	e of t	he val	ues in	our dat
	count	mean	std	min	25%	50%	75%	max
Hour	110940.0	11.493780	6.925509	0.0	5.0	11.0	17.00	23.0
Temperature	110940.0	69.853434	8.511516	43.0	64.0	70.0	76.00	89.0
Humidity	110940.0	69.361259	18.196455	29.0	54.0	71.0	87.00	100.0
Wind	110940.0	6.581908	4.486848	0.0	3.0	6.0	9.00	24.0
Wh Load	110940.0	8475.092098	7452.368773	0.0	1970.0	7372.0	12713.00	57814.0
Wh Load-7d	110940.0	8859.001180	7763.364070	0.0	2063.0	7732.0	13279.25	57814.0
Wh Load-14d	110940.0	8667.456263	7665.040191	0.0	1973.0	7519.0	12951.00	57814.0

g. Generate a correlation matrix of the variables. This will provide statistics on all sixty transformers as a group. Select this cell and hit **Shift+Enter**.

```
# Generate a correlation matrix to check for the exstance of good relatinoships for our model.
   model_corr = modellingData.corr(method ='pearson')
  model corr
             Hour Temperature Humidity Wind Wh Load Wh Load-7d Wh Load-14d
     Hour 1.000000 0.431996 -0.478854 0.165933 0.172268 0.177432 0.178858
Temperature 0.431996
                   1.000000 -0.667318 0.403062 0.318962 0.185169
                                                                    0.169464
  Humidity -0.478854 -0.667318 1.000000 -0.382336 -0.194920 -0.184463 -0.220623
     Wind 0.165933 0.403062 -0.382336 1.000000 0.099989
                                                         0.120011
                                                                    0.121844
  Wh Load 0.172268
                    0.318962 -0.194920 0.099989 1.000000
                                                        0.843289
                                                                    0.820274
Wh Load-7d 0.177432 0.185169 -0.184463 0.120011 0.843289 1.000000
                                                                    0.833011
Wh Load-14d 0.178858 0.169464 -0.220623 0.121844 0.820274 0.833011 1.000000
```

h. Generate a correlation chart of the variables. This will provide statistics on all sixty transformers as a group. Select this cell and hit **Shift+Enter**.



i. Change the index of **modellingData** from the "id" column to the "Transformer" column. Select this cell and hit **Shift+Enter**.

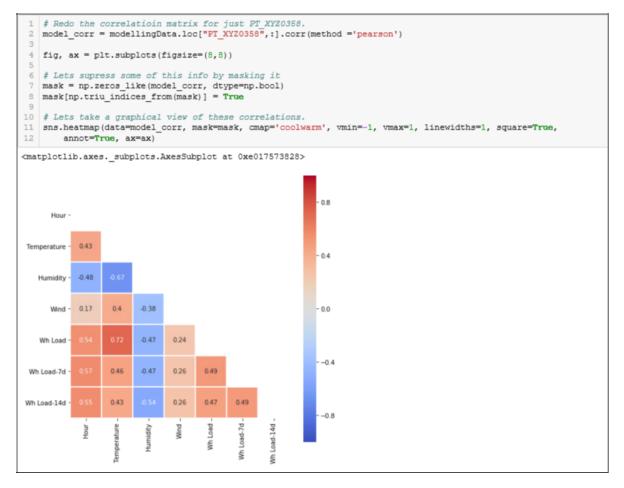
2 modell 3 4 <i># Take</i>	ingData =	modellingData.se	t_ind	-				atframe	's index to	> the "Transformer" colu
Transformer	Transformer	Time Stamp	Hour	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d	
PT_XYZ0343	PT_XYZ0343	6/15/2017 12:00:00 AM	0	68	93.0	7.0	6529.0	3278.0	2464.0	
PT_XYZ0343	PT_XYZ0343	6/15/2017 1:00:00 AM	1	68	93.0	6.0	6413.0	3122.0	2798.0	
PT_XYZ0343	PT_XYZ0343	6/15/2017 2:00:00 AM	2	68	93.0	6.0	6781.0	2658.0	2490.0	
PT_XYZ0343	PT_XYZ0343	6/15/2017 3:00:00 AM	3	68	90.0	0.0	5348.0	2873.0	2722.0	
PT_XYZ0343	PT_XYZ0343	6/15/2017 4:00:00 AM	4	67	93.0	0.0	6916.0	3306.0	2698.0	

With rows indexed by transformer name, we will be able to build a regression models for one transformer.

j. Based on the Power BI report, we know each transformer's load profile is unique. Let's focus on just one, PT_XYZ0358. Select this cell and hit **Shift+Enter**.

	<pre># Here's how we can focus on one transformer, say PT_XYZ0358. modellingData.loc["PT_XYZ0358",:].head()</pre>												
	Transformer	TimeStamp	Hour	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d				
Transformer													
PT_XYZ0358	PT_XYZ0358	6/15/2017 12:00:00 AM	0	68	93.0	7.0	8439.0	2519.0	3071.0				
PT_XYZ0358	PT_XYZ0358	6/15/2017 1:00:00 AM	1	68	93.0	6.0	7596.0	2454.0	2468.0				
PT_XYZ0358	PT_XYZ0358	6/15/2017 2:00:00 AM	2	68	93.0	6.0	7497.0	2253.0	2285.0				
PT_XYZ0358	PT_XYZ0358	6/15/2017 3:00:00 AM	3	68	90.0	0.0	6089.0	2238.0	2193.0				
PT_XYZ0358	PT_XYZ0358	6/15/2017 4:00:00 AM	4	67	93.0	0.0	6128.0	2530.0	2497.0				

k. Generate a correlation chart for PT_XYZ_0358. Select this cell and hit Shift+Enter.



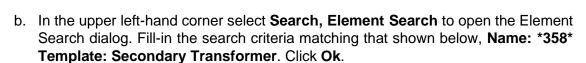
 Generate linear regression model coefficients for PT_XYZ0358 and print the equation. Select this cell and hit Shift+Enter. Keep the browser open, we will be coming back to get this equation.



Test Model - AF Analytics Results Preview

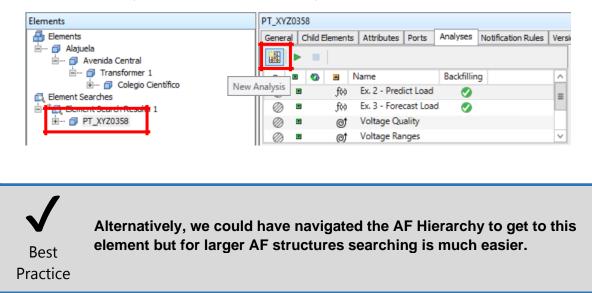
Our first Python script has produced a modelling equation for predicting the load on transformer, PT_XYZ0358. We can use AF Analytics to test the results without using PI tags.

a. Open PI System Explorer from the Windows taskbar,



		Elem	ent Search				X
Template: "Secondary Transformer" Na	me:*358*					× •	Search
		C	riteria				۲
Name: *358*	×.	Element Search Root:			×		
All Descendants: True	¥ ×	Template:	Secondary Trans	former	✓ ×		
Category: <all></all>	v ×						
ନ Add <u>C</u> riteria 🔻	5						
	105	Ri	esults				۲
						Group by: 🗌 Ca	tegory 🗌 Template
🗉 🖻 Name	△ Description	Category	Туре	Template			٢
		Single Phase	None	Single Phase Transformer			
The search found 1 Element(s) matching	g the search criteria.						
					(OK Cancel	Reset

c. Under **Element Search Results** in the AF hierarchy, select element **PT_XYZ0358**. Select the Analysis tab and add a new analysis.



d. Select "Analysis 1" and change the name to "Test Prediction".

PT_XY	Z03	58										
Gener	al	Child	l Ele	ment	s Attributes	Ports	Analyses	Notification Rules	Version			
	▶										Name:	Test Prediction
0	T	6) }	A	Name			Backfilling		^	Description:	
\otimes	T	8		Ø	Voltage Qu	ality					Categories:	
\otimes		1		Ø	Voltage Ra	nges					Analysis Type:	 Expression
\otimes	T	1		Ø	Wh Load					=	11	
				*€	Test Predict	tion						
										\sim		

e. Return to the Python script in Jupyter Notebook and select the load prediction equation for transformer PT_XYZ0358 **Copy** it into the clipboard.



f. Go back to PI System Explorer and **Paste** the equation into the "Test Prediction" analysis.

Add a new va	ariable					Evalua	ite
Name	Expression					Output Attribute	
Variable1	0.16826318479599217 * 'Wh Deliv	vered Load - 7d'	+ 0.1781988453046811	* 'Wh Delivered Load - 14d'	+ 330.09369774241975	* 'Ambient Te <u>Map</u>	
	318479599217 * 'Wh Delivered L 301245552 * 'Relative Humidity			Wh Delivered Load - 14d' +	330.09369774241975	* 'Ambient Temperature' +	



g. Change the variable name from **Variable 1** to *Prediction*. Click **Add a new variable** to add another one.

Г	Add a new variable			t Evaluate	
L	Name	Exp	ssion	Output Attribute	
	Prediction	0.1	826318479599217 * 'Wh Delivered Load - 7d' + 0.1781988453046811 * 'Wh Delivered Load - 14d' + 330.09369774241975 * 'Ambie	Map	۲
1	Variable1			Map	\otimes
	1				

h. Change the new Variable1 to Actual and type in the transformer load attribute
 'Wh Delivered Load'. Once you type the first single quote, AF Analytics should provide a selection list.

Add a new variable]		t Evaluate	
Name	Expression		Output Attribute	
Prediction	0.16826318479599217 *	Wh Delivered Load - 7d' + 0.1781988453046811 * 'Wh Delivered Load - 14d' + 330.09369774241975 * 'Ambie	Map	۲
Actual	'Wh Delivered Load'		Map	⊗

i. Right-click on the "Test Prediction" analysis and click **Preview Results**.

PT_XYZ0	358								
General	Child E	Elements	s Attributes	Ports	Analyses	Notification Ru	ules V	ersion	
0 .	۵	A	Name			Backfilling			^
\otimes		Ø	Voltage Qu	ality					
\otimes		©†	Voltage Rar	nges					
0	T	đ	Wh Load						=
0		.f60	Test Predict		1				
					New				\sim
				×	Doloto				-
Add a	new va	viable			Preview	Results		1×	
Add a	new ve	nable		at	0.1000				

j. In the preview dialog, set the Start Time to 6/15/17 and the End Time to 8/31/17. Click Generate Results.

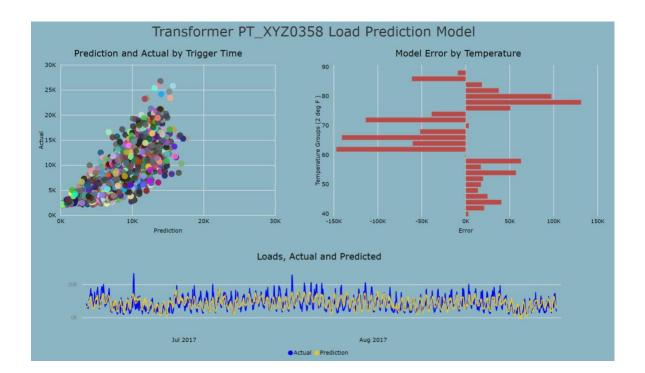
0		Previev	v results for Test Pr	ediction	-	• ×
Start Time: 6/15/17			End Time: 8/31/17	7	Gene	rate Results
Trigger Time I 6/15/2017 12:00:00 AM 6/15/2017 1:00:00 AM 6/15/2017 1:00:00 AM 6/15/2017 2:00:00 AM 6/15/2017 3:00:00 AM 6/15/2017 3:00:00 AM 6/15/2017 4:00:00 AM 6/15/2017 4:00:00 AM 6/15/2017 7:00:00 AM 6/15/2017 9:00:00 AM 6/15/2017 9:00:0	6805.7 7 6739.3 7 6607.4 6 6493.6 6 6715.7 6 6821 7 7291.3 2 8077.5 8	Actual W 8439 7596 7497 6089 6128 6213 6213 7140 5713 8761 8551	/h Delivered Load - 7d 2519 2454 2253 2238 2530 3991 2591 3197 4077 4362	Wh Delivered Load - 14d 3071 2468 2285 2193 2497 2364 2424 3273 4418 6282		100% Progress 1.1ms Min (78.5ms Max
• 5752.93 • 4052 • 3455 30000 20000 20000	ш	2 - 86			re = ⁴ re	2.5ms Avg (2954.2s) Avg Trigger (5.8s) Total
-5000 6/15/2017 12:00:00 AM Value Scale: Single		7 days	8/31/2017 12;	00:00 AM	Expor	2253 Evaluations t Results

k. Once you see a completed trend of data, the PI Analysis Service has finished. Click **Export Results** and save the .csv file to the **Exploring AF for Advanced Analytics** desktop folder.

e) - + 🖡	This PC Desktop		v c	Search Desktop	p
Organize • New	folder)# •	
Recent places Chis PC Desktop Des		-	Name Old Old Python Scripts Python Scripts Fr.2: Bod Display Soc.2: Bod Display Soc.2: Prediction Evaluation Soc.2: Prediction Evaluation Soc.2: Prediction Evaluation Soc.2: Prediction Evaluation Comparison of the Advanced Analytic Comparison of the Advanced Analytic P Integrator for BA	Date modified 4/21/2016 10.19 PM 1/17/2016 10-90 AM 6/6/2018 7/19 PM 1/2/17/2018 10-12 1/2/17/2018 10-12 1/2/19/2018 10-02 5 9/17/2018 10-	Fi In In In
File name:	Test Prediction_15-Jun-	17-0	0-00-00_31-Aug-17-00-00-00.csv		-
Save as type:	CSV (*.csv)				

Evaluate Model – Power BI

Reopen the Power BI report, "Example 1 Data Analysis". The "Test Prediction_15-Jun-17-00-00-00_31-Aug-17-00-00-csv" file has been imported into this report. Click the bottom tab **PT_XYZ0358 Model**, to see some example analysis of the model test. Looks like there may be room for improvement, but this will be good enough for this lab's curriculum.



Exercise 2 - Develop Predictive Models for Multiple Transformers

Learning Objectives

- Create a view using PI SQL Client to extract transformer load and weather data directly from the PI System.
- Develop load prediction model for multiple transformers using Python, in Jupyter Notebooks.
- Test and assess this model using AF analytics by backfilling the prediction into PI.

Create a PI View – PI SQL Client

- a. Open PI SQL Commander Lite from the Windows taskbar,
- b. Under the PI SQL Client branch of the Object Explorer hierarchy, select the PISRV01\Distribution Network catalog and click Connect. Once the Data Link Properties dialog appears, click Ok.

💼 🛛 Data I	Link Properties
Connection Advanced All	
Data Source	
AF Server:	PISRV01
AF Database:	Distribution Network 🗸 🗸
Authentication	
✓ Trusted connection	
User Name:	
Password:	
Test Connection]
C)K Cancel Help

c. Expand the Object Explorer hierarchy through the **Master \ Element \ Table Valued-Functions** path.

Right-click on the **Single Phase Transformer_GetSampleValues** function and select **Execute Predefined Query...** The query's results will be returned to the grid.

This function was created based on the Single Phase Transformer template defined in the Distribution Network AF model. Next, we will create our own view to access the data needed for our Python modelling.

0	PI SQL Commander Lite	_ _ X
File Edit View Query Tools Help		
📜 New Query 🚦 Execute 💷 🚳 🏹 🖏 🚱 Query Compendium		
Object Explorer ×	Query1.sql - PISRV01\Distribution Network X	-
	SELECT e.Name, s." FROM	
⊿ j OLEDB Data Sources	(
PI OLEDB Enterprise	SELECT TOP 100 ID, Name, Template	
I PISRV01	FROM [Master].[Element]	
A CONTRACT OF CONTRACT.) e CROSS APPLY [Master].[Element].[Single Phase Transformer_GetSampledValues]	
PISRV01	(
	e.ID,Element ID	
PISRV01\Distribution Network	N'y',Start Time N't',End Time	
⊿ 🚞 Catalogs	N'th'Time Step	
Master) s	
Category	WHERE e.Template = N'Single Phase Transformer'	
⊿ 📴 Element		
> 🗀 Tables	Results hessages	
Views	Name TimeStamp Loading Maximum KVA Rated KVA Wh Delivered Load Wh Delivered Load - 7d Wh D	Delivered Load - 14d Ambient Temperature Relative Humidity
⊿ Dable-Valued Functions	1 PT XYZ0379 2018-09-23 00:00:00 00 31.28 30.5 25 7820 7820 7820 7820	
FindElements	2 PT_XYZ0379 2018-09-23 01:00:00.000 31.28 30.5 25 7820 7820 7820 7820	
j /k GetPlotValues	3 PT_XYZ0379 2018-09-23 02:00:00 000 31.28 30.5 25 7820 7820 7820 7820	
⊳ 6 GetSampledValue	4 PT_XYZ0379 2018-09-23 03:00:00:000 31:28 30.5 25 7820 7820 7820 7820	
GetSampledValues	5 PT_XY20379 2018-09-23 04:00:00 000 31.28 30.5 25 7820 7820 7820 7820 7820 7820 7820 7820	
K GetSummaries	6 PT_XY20379 2018-09-23 05:00:00:000 31.28 30.5 25 7820 7820 7820 7820 7820 7820 7820 7820	
p (windex.contributes	8 PT_XYZ0379 2018-09-23 07:00:00 00 31.28 30.5 25 7820 7820 7820 7820	
Sinole Phase Transformer. GetSamiledValues	9 PT_XYZ0379 2018-09-23 08:00:00.000 31:28 30.5 25 7820 7820 7820 7820	
	10 PT_XYZ0379 2018-09-23 09:00:00:000 31:28 30:5 25 7820 7820 7820 7820	
	11 PT_XYZ0379 2018-09-23 10:00:00 00 31.28 30.5 25 7820 7820 7820 7820	
Drop	12 PT_XYZ0379 2018-09-23 11:00:00:000 31:28 30.5 25 7820 7820 7820 7820	
EventFrat Create Function Table	13 PT_XYZ0379 2018-09-23 12:00:00:000 31.28 30.5 25 7820 7820 7820 7820 7820 7820 7820 7820	
Carl UnitOfMe Scripts	14 PT_X1203/3 2018-05-23 1500/00.000 31.28 30.5 25 7820 7820 7820 7820 7820 7820	
Scalar Functions		
ODBC Data Sources	Q Query executed successfully PISRV01\Distribu	ution Network PI SQL Client OLEDB 00:00:05.268 175 rows

d. Make a new Table-Valued Function by right-clicking on the Element folder and selecting Create Template-specific Data Model... This will open the dialog shown below. Select the Single Phase Transformer template and click Next.

	Template-Specific Data Model x
🥢 OSIs	oft.
Template	Template Type
Data Model Objects	Element
Summary	O Event Frame
Execution	Template
	Circuit
	Headquarters
	HMM Single Phase Hourly
	Meter
	Phase
	Secondary Transformer
	Single Phase Transformer
	Single Transformer Substation
	Substation Transformer Bank
	Weather

	Next > Cancel

e. Next, click Add GetSampledValues in the Data Model dialog.

	Template-Specific Data Model	
i os	soft.	
Template	Template-Specific Data Model Objects	
Data Model Objects		Add Element View
Summary		Add GetSampledValue
Execution		Add GetSampledValues
		M.
		Add GetSummaries
		Modify
		Remove
	< Back Next >	Cancel

f. At the top of the Column Definition dialog, uncheck Units of Measure and Error. Change the Table-valued function name from the default, which has already been created, to "<u>My</u> Single Phase Transformer_GetSampledValues". Drag and drop the attributes shown below to specify the columns we need for analysis. Click OK.

Table-valued function name:					
My Single Phase Transformer_	GetSa	mpledValues			
Drag and drop attributes:	_				
Kambient Temperature	^	Attribute	Value	Unit of Measure	Error
Asset ID		ILoading	Loading		
Coefficient_0 Coefficient_1		IMaximum KVA	Maximum KVA		
Goefficient_1	≡	Rated KVA	Rated KVA		
Coefficient_3		Wh Delivered Load	Wh Delivered Load		
Intercept			Wh Delivered Load - 7d		
📑 Loading			Wh Delivered Load - 14d		
of MATLAB Forecasted Load		[] Ambient Temperature	Ambient Temperature		
KATLAB Predicted Load					
🔄 Maximum KVA		Relative Humidity	Relative Humidity		
Phase		Wind Speed	Wind Speed		
🐔 Python Predicted Load	~				
Show hidden	¥				

g. Returning to the Data Model dialog, with the new function configured, click **Next**.

	Template-Specific Data Model	x
i osi	soft.	
Template	Template-Specific Data Model Objects	
Data Model Objects	My Single Phase Transformer_GetSampledValues	Add Element View
Summary		Add GetSampledValue
Execution		Add GetSampledValues
		Add GetSummary
		Add GetSummaries
		Modify
		Remove
	< Back Next >	Cancel

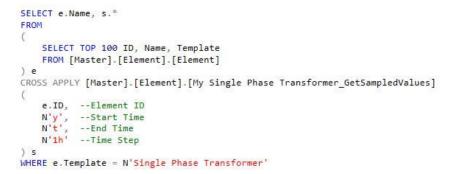
h. The CREATE FUNCTION has been configured for you. Click **Execute** to create the function.

	Template-Specific Data Model	
CO. OSISC	Summary	
Data Model Objects Summary	CREATE FUNCTION [Master].[Element].[My Single Phase Transformer_GetSampledValues]	^
Execution	©ElementID Guid ©StarTime DateTime @EndTime DateTime ©TimeStep String	=
	/ AS SELECT * FROM [Master] [Element] [GetSampledValues] <	
	N'Single Phase Transformer',Template { N'[Loading', AttributeTemplatePath N'Loading', ValueColumnName	
	NULL UnitOfMessureColumnName NULL ErrorColumnName),	
	N'IMaximum KVA', N'Maximum KVA', NULL,	~
	< Back Execute	Cancel

i. A green Status check means the function is configured. Click **Done**.

Template-Specific Data Model									
CO. OSI	oft.								
Data Model Objects	- Togical	Execution completed succesfully							
Summary Execution	Details:								
	Data Model Object Type	Name	Status						
	GetSampledValues Function My Single Phase Transformer_GetSampledValues								

j. Return to the Table-Valued Functions folder in the Object Explorer and find your new function, **My Single Phase Transformer_GetSampledValues**. Right-click on it and select **Execute Predefined Query...** The following query should appear about the query viewer.

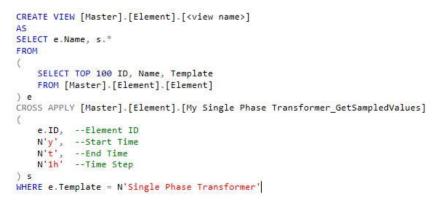


k. The default function is limited to 100 rows and its time range is set to yesterday at onehour intervals. We will modify this query and save it as a view, so we can get all the data needed for our statistical model. To start, copy this entire query into the clipboard.

Select the **Views** folder under Element. Right-click on **Views**, select **Scripts**, then **CREATE VIEW**, then click **New Query Editor Window**. The following query will appear. Replace <view definition> with the query saved in the clipboard.

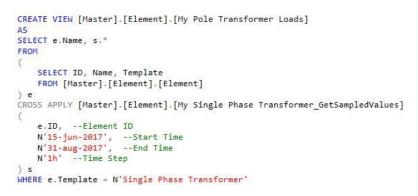
```
CREATE VIEW [Master].[Element].[<view name>]
AS
<view definition>
```

I. The following query should now appear in the query window.



- m. Modify this query by;
 - a. Replacing, within the brackets, "<view name>" with "My Pole Transformer Loads".
 - b. Deleting "TOP 100" from the second SELECT statement.
 - c. Change the Start Time from 'y' to '15-jun-2017'.
 - d. Change the End Time from 't' to '31-aug-2017'.

n. When finished with the edits, the query should look like the one below.



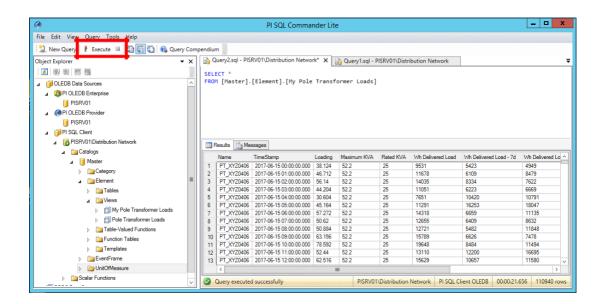
 Either right-click on the query and select Execute or click on the Execute button on the top menu bar to run this query and to create the view. You should see the following message in the Results grid and a green light in the bottom message bar of PI SQL Commander Lite.

Results	Messages	
Affected	RowCount	0
1 [-1		Query exe

p. Return to the Views folder in the Object Explorer hierarchy. You may need to right-click on the view folder and **Refresh** it to see your new view. Right-click on the view, **My Pole Transformer Loads**, and **Execute Predefined Query...** The query will return the first 100 rows of the view.

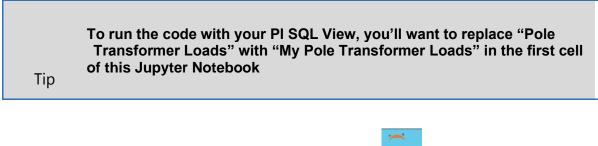
PI SQL Commander Lite											
File Edit View Query Tools Help											
😫 New Query 🚦 Execute 💷 🦥 🏹 🍋 Query Con	npendiur	m									
Object Explorer 🗸 🙀 Query2.sql - PISRV01\Distribution Network 🗙 🎧 Query1.sql - PISRV01\Distribution Network 🗴 🖉											
Image: Select TOP 100 * I											
CLEDB Data Sources	11				-						
PISRV01											
	11										
GPI OLEDB Provider											
	I PISRV01										
⊿ <i>i</i> PI SQL Client											
PISRV01\Distribution Network	E F	Results 📑 Me	isages								
⊿ 🚞 Catalogs		Name	TimeStamp	Loading	Maximum KVA	Rated KVA	Wh Delivered Load	Wh Delivered Load - 7d	Wh Delivered Lo	Ē	
⊿ 🔰 Master	1	PT XYZ0342	2017-06-15 00:00:00.000		0.1	10	365	370	369	il.	
Category	2	PT_XYZ0342	2017-06-15 01:00:00.000	3.65	0.1	10	365	370	370	4	
⊿ 📄 Element	3	PT XYZ0342	2017-06-15 02:00:00.000	3.64	0.1	10	364	371	370		
Tables	4	PT_XYZ0342	2017-06-15 03:00:00.000	3.64	0.1	10	364	346	371		
Views	5		2017-06-15 04:00:00.000		0.1	10	0	0	7		
My Pole Transformer Loads	6		2017-06-15 05:00:00.000		0.1	10	0	0	0		
Pole Transformer Loads	7		2017-06-15 06:00:00.000	-	0.1	10	0	0	0		
Table-Valued Functions	8		2017-06-15 07:00:00.000 2017-06-15 08:00:00.000	0	0.1	10	0	0	0		
	10		2017-06-15 09:00:00.000		0.1	10	0	0	0		
Function Tables	11	PT_XYZ0342	2017-06-15 10:00:00.000		0.1	10	0	0	0		
Templates	12	PT_XYZ0342	2017-06-15 11:00:00.000	0	0.1	10	0	0	0		
EventFrame	13	PT_XYZ0342	2017-06-15 12:00:00.000	0	0.1	10	0	0	0	7	
DitOfMeasure		<			1				>		
Calar Functions	0	Query executed	l successfully	1000	PISR	V01\Distribut	ion Network PI SC	L Client OLEDB 00:00	:02.749 100 row:	5	

q. Edit the query by deleting the "TOP 100" specification in the SELECT statement. Either right-click on the query and select **Execute** or click on the **Execute** button on the top menu bar to rerun the query. This time you should get all 110,940 rows.



Develop the Model in Python

The following code creates a multivariable linear regression model for all 60 transformers. We will use the "adodbapi" package for Python to connect directly to PI through the view previously created using the PI SQL Client. The code will publish coefficients for each modelling equation to a table in MS SQL. This table will be reference by AF allowing us to easily reproduce these models through the Single Phase Transformer template.



- a. Open Jupyter Notebooks from the windows Taskbar, _____. Open the Python script, "Load Prediction - Example 2 (PI SQL Client)".
- b. In the first cell, change the select statement to 'SELECT * FROM [<u>My</u> Pole Transformer Loads]'). This will cause Python to access your PI SQL Client view and create the dataframe, poleTransformerLoads. Select this cell and hit Shift+Enter. Wait until you see the table preview before moving to the next cell.

PI View column	PI View column Names:											
['Name', 'TimeStamp', 'Loading', 'Maximum KVA', 'Rated KVA', 'Wh Delivered Load', 'Wh Delivered Load - 14d', 'Wh Del ivered Load - 7d', 'Ambient Temperature', 'Relative Humidity', 'Wind Speed']												
Name	TimeStamp	Loading	Maximum KVA	Rated KVA	Wh Delivered Load	Wh Delivered Load - 14d	Wh Delivered Load - 7d	Ambient Temperature	Relative Humidity	Wind Speed		
0 PT_XYZ0342	2017-06-15 00:00:00	3.65	0.1	10.0	365.0	369.0	370.0	68.0	93.0	7.0		
1 PT_XYZ0342	2017-06-15 01:00:00	3.65	0.1	10.0	365.0	370.0	370.0	68.0	93.0	6.0		
2 PT_XYZ0342	2017-06-15 02:00:00	3.64	0.1	10.0	364.0	370.0	371.0	68.0	93.0	6.0		
3 PT_XYZ0342	2017-06-15 03:00:00	3.64	0.1	10.0	364.0	371.0	346.0	68.0	90.0	0.0		
4 PT_XYZ0342	2017-06-15 04:00:00	0.00	0.1	10.0	0.0	7.0	0.0	67.0	93.0	0.0		

c. Rename columns with shorter names. Create a smaller dataframe, **modellingData**, containing only the values need for predicting transformer loads. Select this cell and hit **Shift+Enter**.

3 4 5 6 7 8 9 10 11 12 13	<pre>1 # Rename some columns vith shorter names to make them easier to vork vith. 2 poleTransformerLoads.rename(columns = {'Name' : 'Transformer'}, inplace = True) 3 poleTransformerLoads.rename(columns = {'Wh Delivered Load': Wh Load'}, inplace = True) 4 poleTransformerLoads.rename(columns = {'Wh Delivered Load - 14d': Wh Load-14d'), inplace = True) 5 poleTransformerLoads.rename(columns = {'Wh Delivered Load - 7d': 'Wh Load-14d', inplace = True) 6 poleTransformerLoads.rename(columns = {'Rubient Temperature': 'Temperature', inplace = True) 7 poleTransformerLoads.rename(columns = {'Relative Humidity': Humidity', inplace = True) 8 poleTransformerLoads.rename(columns = {'Wind Speed': 'Wind', inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Speed': 'Wind', inplace = True) 9 # Define second dataframe vith just data needed for our modelling. 1 modellingData = poleTransformerLoads[['Transformer', 'TimeStamp', 'Temperature', 'Humidity', 2 'Wind', 'Wh Load', 'Wh Load-7d', 'Wh Load-14d']] 4 # Peek at the first five rows to make sure things look right. 5 modellingData.head()</pre>										
	Transformer	TimeStamp	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d			
0	PT_XYZ0342	2017-06-15 00:00:00	68.0	93.0	7.0	365.0	370.0	369.0			
1	PT_XYZ0342	2017-06-15 01:00:00	68.0	93.0	6.0	365.0	370.0	370.0			
2	PT_XYZ0342	2017-06-15 02:00:00	68.0	93.0	6.0	364.0	371.0	370.0			
3	PT_XYZ0342	2017-06-15 03:00:00	68.0	90.0	0.0	364.0	346.0	371.0			
4	PT_XYZ0342	2017-06-15 04:00:00	67.0	93.0	0.0	0.0	0.0	7.0			

d. Change the index of **modellingData** to be the transformer name. Select this cell and hit **Shift+Enter**.

2 model] 3 4 <i># Take</i>	4 # Take a look, see the difference?												
Transformer	Transformer	Time Stamp	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d					
PT_XYZ0342	PT_XYZ0342	2017-06-15 00:00:00	68.0	93.0	7.0	365.0	370.0	369.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 01:00:00	68.0	93.0	6.0	365.0	370.0	370.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 02:00:00	68.0	93.0	6.0	364.0	371.0	370.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 03:00:00	68.0	90.0	0.0	364.0	346.0	371.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 04:00:00	67.0	93.0	0.0	0.0	0.0	7.0					

e. Leverage the connection to PI SQL Client and **Pole Transformer Loads** view to get the list of transformer names. Select this cell and hit **Shift+Enter**.



f. Generate linear regression model coefficients for each of the 60 transformers and write them to SQL. Select this cell and hit **Shift+Enter**.

```
# Using the "adodbapi" package, to connect to the "Predictive Equations" MS SQL.
    # Set connection parameters.
 4 con string = 'DRIVER={SQL Server};SERVER=PISRV01;DATABASE=PIWorld;Trusted Connection=Yes;'
    # Connect to "Distribution Network Lab" model in AF through the PI SQL Client.
 6
 7 SQL connection = ado.connect(con string)
    # Create a cursor object to access the data serer for the "Distribution Network Lab" database in AF.
 9
10 SQL_cursor = SQL_connection.cursor()
12
    # Create linear regression object from the "sklearn" package we imported earlier.
13 LinReg = LinearRegression()
15 # Looping through the trasformer list, Perform linear regression on each transformer.
16 for transformer in transformerNames:
         # Create dataframe for one transformer.
18
19
        transformerData = modellingData.loc[transformer.:]
20
21
         # Perform linear regression fit
22
        LinReg.fit(transformerData[["Wh Load-7d", "Wh Load-14d", "Temperature", "Humidity"]], transformerData["Wh Load"])
23
        # Update asset ID value with the name of this transformer.
asset_id = "!"+transformer+"!"
24
25
26
27
         # Print equation.
        # Fint equation:
print etamsformer, "Eq:\n", LinReg.coef_[0], "*","'Wh Delivered Load - 7d' + ",LinReg.coef_[1],
    "*","'Wh Delivered Load - 14d' + ", LinReg.coef_[2], "*", "'Ambient Temperature' +",
    LinReg.coef_[3], "*","'Relative Humidity' +(",LinReg.intercept_, ")" )
28
29
30
31
32
        ## INSERT and UPDATE queries to load table for the first time or update an existing one.
         # Construct query to add this transfomer's model coefficients.
33
34
35
        insert_query = f'INSERT [Predictive Equations] ([Asset ID], Coefficient_0, Coefficient_1, Coefficient_2, Coeffic
36
        update_query = f'UPDATE [Predictive Equations] SET Coefficient_0={LinReg.coef_[0]}, Coefficient_1={LinReg.coef_[
37
38
         # Insert this record into the "Predictive Equations" table.
39
        SQL_cursor.execute(update_query)
40
41 # Commit the queries to write the data into SQL.
42 SQL_connection.commit()
43
44
    # Close database connections.
45 SQL_connection.close()
46 PI connection.close()
```

PT_XYZ0378 Eq:

```
N___NC2008013229636 * 'Wh Delivered Load - 7d' + 0.17917457192857675 * 'Wh Delivered Load - 14d' + 110.23388419266524 * 'A
mbient Temperature' + -0.2658384322606984 * 'Relative Humidity' +( -3314.933455710232 )
PT__XY20415 Eq:
-0.001999771117489486 * 'Wh Delivered Load - 7d' + 0.0035564537270706955 * 'Wh Delivered Load - 14d' + 0.05148928700184387
4 * 'Ambient Temperature' + -0.0032873513352864587 * 'Relative Humidity' +( -1.0928457626402812 )
PT__XY20407 Eq:
0.22133353596401245 * 'Wh Delivered Load - 7d' + 0.18940221947528418 * 'Wh Delivered Load - 14d' + 303.2482562107521 * 'Am
bient Temperature' + 7.845159067779704 * 'Relative Humidity' +( -14852.0518428231 )
PT__XY20385 Eq:
0.17033499389302167 * 'Wh Delivered Load - 7d' + -0.13442528680210902 * 'Wh Delivered Load - 14d' + -0.001006904293367613
* 'Ambient Temperature' + 0.002808240581481683 * 'Relative Humidity' +( 10.552971591136941 )
PT__XY20345 Eq:
0.1429519100661018 * 'Wh Delivered Load - 7d' + 0.06105023128834538 * 'Wh Delivered Load - 14d' + 466.02179316494465 * 'A
mbient Temperature' + 33.496691778996954 * 'Relative Humidity' +( -26399.777514147783 )
PT__XY20488 Eq:
0.22918154817171893 * 'Wh Delivered Load - 7d' + 0.1723254292743076 * 'Wh Delivered Load - 14d' + 207.3195523807009 * 'Amb
ient Temperature' + 5.783909412754433 * 'Relative Humidity' +( -8273.493697771371 )
```

Test Model - AF Analytics Backfill

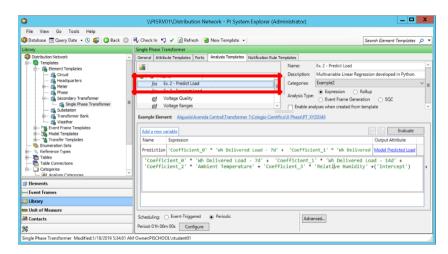
a. Open PI System Explorer and go to the **Library** view. Expand the **Tables** branch and click on the **Predictive Equations** table. Select the **Table** tab to view the model coefficients generated by the Python script. Although the script wrote the coefficients to a MS SQL table, we have imported them into the AF model for use in AF Analytics.

🕽 Database 🛗 Query Date 👻 🕔 🥥	3 Back 💿 🗟 Check li	n 🍤 🖌 🛃 Refrest	n 📲 New Table 🔻			Search	Tables	P
brary	Predictive Equations							
Distribution Network 	General Table Define	Table Version						
B→ B Element Templates B→ B Event Frame Templates B→ B Model Templates B→ B Transfer Templates B→ B Transfer Templates B→ B Elementation Sets B→ B R Efference Types	Predictive Equations							
	Filter							P
	Asset ID	Coefficient_0	Coefficient_1	Coefficient_2	Coefficient_3	Coefficient_4	Intercept	
	PT_XYZ0382	0.07263672	0.002437648	473.7613	67.40076		-24762.56	
	PT_XYZ0342	0.5932921	0.4004194	0.2417774	0.1231081		-21.13118	
Tables Predictive Equations	PT_XYZ0343	0.04395182	0.09916914	351.8439	20.93276		-18311.69	
III Transformer Ratings	PT_XYZ0344	0.07994196	0.0909123	602.0067	75.46542		-35513.53	
	PT_XYZ0345	0.05813053	0.09053257	245.7917	25.87701		-14823.33	
Categories Categories	PT_XYZ0346	0.1423417	0.06116662	466.0096	33.47753		-26399.08	
Attribute Categories	PT_XYZ0347	0.1096358	0.05748888	646.9308	55.40506		-34972.06	
····· @ Element Categories	PT_XYZ0348	0.1038648	0.133573	283.0549	41.00509		-17791.91	
····· 🔳 Notification Rule Categories	PT_XYZ0349	0.06923891	0.1093802	693.1084	60.73282		-38121.64	
Table Categories	PT_XYZ0350	0.1187499	0.09494022	295.3516	30.45709		-17761.45	
	PT_XYZ0351	0.1488309	0.1010185	333.9626	30.93592		-19181.05	
3 Elements	PT_XYZ0352	0.2833661	0.2792848	211.6137	61.29921		-15494.79	
Event Eramor	PT_XYZ0353	0.1959453	0.1267081	237.1805	10.61177		-12522.72	
🔛 Library	PT_XYZ0354	0.05494547	0.04467159	223.9388	19.21531		-12764.99	
	PT_XYZ0355	0.1528177	0.1889331	136.4436	20.26082		-8309.751	_
M Contacts	PT_XYZ0356	0.1931092	0.07253212	271.1282	13.41234		-14217.23	_
*	PT XYZ0357	0.1408201	0.08628307	227.5313	11.93556		-11773.3	-

b. Return to the Elements view in PI System Explorer. Select any Single Phase Transformer element, PT_XYZ... Select the element's Attributes tab. Find the attributes under the "Ex. 2 Load Prediction". Select the Coefficient_1 attribute and note the used of the Table Lookup Data Reference to make each transformer's coefficients available within the element.

🙆 Database 📅 Query Date 🔹 🕔 🛃 🔇 Back 🌘		neck in 🔊	✓ 🕼 Refresh 📑 New Element →	New Attribute			Search Elements	Q		
Elements	-	(YZ0349		-				,		
🖶 – 🗇 X Phase	^ Gen	eral Child E	ements Attributes Ports Analyses	Notification Rules Versio	n					
E- J Y Phase		- ²					Group by: 🗹 Category	Templat		
B→ 🗇 PT_XYZ0344 B→ 🗊 PT_XYZ0345	Filte	er		• م] [] !	Name:	Coefficient_0			
🖻 🗇 PT_XYZ0346			▲ Value @ ^	1	escription:	Coeff. for Wh - 7d Term				
		₫ 🖬 🔶	🛷 Wh Delivered Load	13163 Wh	1	Properties:	<none></none>			
I → ☐ PT_XYZ0349		8	🍼 Wh Delivered Load - 7d	13163 Wh		Categories: Default UOM:	Ex. 2 Load Prediction	6		
		8	🎺 Wh Delivered Load - 14d	13163 Wh	1		<none></none>			
₽ 🗗 PT_XYZ0352	9	Category: Ex. 2 Load Prediction				Value Type:	Single			
Image: Image	T	8	Coefficient_0 Coefficient_1	0.069239	T I	Value:	0.069239			
	-			0.10938	•	Data Reference:	Table Lookup	2		
iii ∰ PT_XY20356 iii ∰ PT_XY20357	=		Coefficient_2	693.11	1	Display Digits:	-5			
			Coefficient 3	60.733	[Settings			
🗑 👩 PT_XYZ0385			Intercept	-38122		SELECT [Coefficie	nt_0] FROM [Predictive Equations] WHERE [Ass	set ID] =		
Image: Image		5	Model Predicted Load	12444 Wh		"%Element%' ORI	DER BY Coefficient_0			
Interpretation Interpretatio Interpretatio Interpretation Interpretation Inte	E	Catego	ry: Ex. 3 Load Forecasting							
- 🗇 PT_XYZ0401		0 84	Model Forecast Load	1760.7 Wh =						
B	. 0	Catego	ry: Network							
1 Elements			E Phase	Y Phase						
Event Frames	-		Substation	Alajuela						
Library	-	Catego	ry: Weather Conditions							
Durit of Measure		0 8	Ambient Temperature	45 deg F						
Contacts		J 18	Relative Humidity	83 %						
🔆 Management			Ø Wind Sneed	0 mi/h ~		Limits Forecasts				

c. Select the element's Analyses tab. Select the Ex. 2 – Predict Load. Here the linear model developed in Python has been expressed in terms of AF Attributes. This expression was configured as part of the template used for all Pole (Single Phase) Transformers.



d. Next, we will use the backfilling feature of AF Analytics to populate he result of the predictive model into PI points configured for all 60 transformers.

Go to the **Management** view in PI System Explorer (icon in lower left-hand corner). Select **Example 2 Prediction** under Analysis Searches. This allows us to manage just the AF Analytic configured for Example 2 of the lab.

			\\PISRV01\Distribution Network - PI S	iystem Explorer (/	Administrator)				
File View Go Tools	Help								
Database 📑 Query Date	- 🕚 💋 🔇	Back	🗊 📕 Check In 🍤 🖌 🛃 Refresh						
lanagement	Analyses								
Choose a type	0 total analys	es sele	cted (0 on this page)	2 0	Operations				
 Analyses 	C Status	Status 🚯 📕 Element Name				Backfilling	_ ~	able Disable selected analyses	
 Notification Rules 			 Alajuela/Avenida Central/Transformer 1\Colegio Científico\Z Phase\PT XYZ0382 		Template Ex. 2 - Predict Load	backning			
Analysis Searches		J.			Ex. 2 - Predict Load			able Disable automatic recalculation for selected alyses	
+ ×		ß			Ex. 2 - Predict Load				
					Ex. 2 - Predict Load Ex. 2 - Predict Load		<u>Br</u>	ckfill/Recalculate selected analyses	
All 👁		ß			Ex. 2 - Predict Load Ex. 2 - Predict Load			Start 6/15/2017 12:00:00 AM	
Enabled 🔷 👁		ft							
Dicabled		ß			Ex. 2 - Predict Load			End 8/31/2017 12:00:00 AM	
Example 2 Prediction		ft			Ex. 2 - Predict Load			What should we do with existing data?	
Example 2 Prediction		ß			Ex. 2 - Predict Load		=	 Leave existing data and fill in gaps 	
Example 3 Forecast 🥜		ft			Ex. 2 - Predict Load			 Permanently delete existing data and recalcula 	
		ft			Ex. 2 - Predict Load			Recalculate dependent analyses	
		ft			Ex. 2 - Predict Load			-	
		f	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0345		Ex. 2 - Predict Load			Queue	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0344		Ex. 2 - Predict Load			Recalculation will permanently delete all the	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0379	Ex. 2 - Predict Load	Ex. 2 - Predict Load		1 C	Recalculation will permanently delete all the	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XY20420	Ex. 2 - Predict Load	Ex. 2 - Predict Load			ending Operations	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0402	Ex. 2 - Predict Load	Ex. 2 - Predict Load			o pending operations	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0388	Ex. 2 - Predict Load	Ex. 2 - Predict Load		N	o pending operations	
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XY20400	Ex. 2 - Predict Load	Ex. 2 - Predict Load				
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0399	Ex. 2 - Predict Load	Ex. 2 - Predict Load				
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0353	Ex. 2 - Predict Load	Ex. 2 - Predict Load				
		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0398	Ex. 2 - Predict Load	Ex. 2 - Predict Load		11		
		ft) Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0343	Ex. 2 - Predict Load	Ex. 2 - Predict Load				
	00	ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0390	Ex. 2 - Predict Load	Ex. 2 - Predict Load		1		
Elements		ft	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0409	Ex. 2 - Predict Load	Ex. 2 - Predict Load		11		
	00	f	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0394	Ex. 2 - Predict Load	Ex. 2 - Predict Load		11		
- Event Frames	- 6	Ŕ	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0414	Ex. 2 - Predict Load	Ex. 2 - Predict Load				
🏐 Library	L ő	ß			Ex. 2 - Predict Load				
Durit of Measure		ft			Ex. 2 - Predict Load				
Contacts			Alatuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT XYZ0408		Ex. 2 - Predict Load				
	- 6 6		Alaiuela/Avenida Central/Transformer 1/Colegio Cientifico/Y Phase/PT XYZ0391		Ex. 2 - Predict Load				
*	Hø		Algiuela/Avenida Central/Transformer 1/Collegio Cientifico/Y Phase/PT_XY20385		Ex. 2 - Predict Load		V		

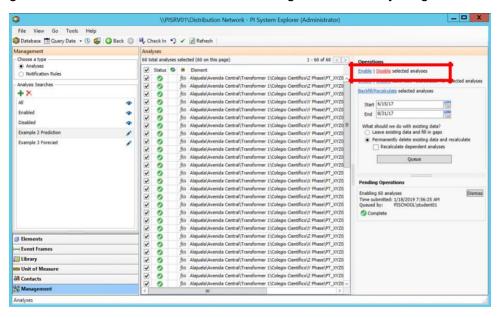
e. Check the **Status** box at the top of this column to select all transformers and click **Enable** to start this group of 60 analytics.

		12.2										
File View Go T												
Database III Query D	ate -	() 🚱 🔾 Ba	ck 🗊 🎽	Check In 🧐 🖌 🛃 Refresh								
Management		Analyses										
Choose a type		1 - 60 of 60 < >						Onesting				
 Analyses 		Status 🚳	Eler	nent	Name	Template	Backfilling	grable Disable selected analyses				
O Notification Rules				ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720382 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
Analysis Searches				ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT X		Ex. 2 - Predict Load		analyses				
+×			foo Alan	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT X	720342 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Backfill/Recalculate selected analyses				
All	-		foo Alate	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0358 Ex. 2 - Predict Load	Ex. 2 - Predict Load		backrightecalculate selected analyses				
				ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT X		Ex. 2 - Predict Load		Start 6/15/2017 12:00:00 AM				
Enabled	*		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Científico\Y Phase\PT_X	720351 Ex. 2 - Predict Load	Ex. 2 - Predict Load		End 8/31/2017 12:00:00 AM				
Disabled	*		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0350 Ex. 2 - Predict Load	Ex. 2 - Predict Load		End Contraction in the				
Example 2 Prediction	1		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0349 Ex. 2 - Predict Load	Ex. 2 - Predict Load		What should we do with existing data?				
Example 3 Forecast			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0348 Ex. 2 - Predict Load	Ex. 2 - Predict Load	=	 Leave existing data and fill in gaps 				
Example 5 Forecase	1			ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT X		Ex. 2 - Predict Load		 Permanently delete existing data and recalculate 				
			foo Alati	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	720346 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Recalculate dependent analyses				
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0345 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Queue				
							foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0344 Ex. 2 - Predict Load	Ex. 2 - Predict Load		
				foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_X	YZ0379 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Recalculation will permanently delete all the			
						foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0420 Ex. 2 - Predict Load	Ex. 2 - Predict Load			
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720402 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Pending Operations				
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_X	720388 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Enabling 60 analyses Dism				
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0400 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Time submitted: 12/20/2018 11:57:59 AM				
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_X	720399 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Queued by: PISCHOOL\student01				
		V 0	foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0353 Ex. 2 - Predict Load	Ex. 2 - Predict Load		Complete				
		V 00	foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	7Z0398 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
		V 0	foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_X	YZ0343 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_X	720390 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
Elements	-		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720409 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
- Event Frames	-		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0394 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
	_		foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720414 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
📓 Library			foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720384 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
Duit of Measure		V 0	foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0404 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
Contacts		V 0	foo Alaji	ela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_X	720408 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
	-	V 📀	foo Alaji	iela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	YZ0391 Ex. 2 - Predict Load	Ex. 2 - Predict Load						
*			foo Alati	ela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_X	720385 Ex. 2 - Predict Load	Ex. 2 - Predict Load	~					

f. Set the Backfill Start and End times to 6/15/17 and 8/31/17, respectively. Check **Permanently delete existing data and recalculate**. The click **Queue** button to start the Backfill operation.

9					\\PISRV01\Distribution Network - PI	System Explorer (Administrator)			
File View Go	Tools	Help								
🕽 Database 🛅 Query	Date 🔸	0	🖳 🔇 E	lack (🕽 💐 Check In 🍤 🖌 🛃 Refresh					
Management		Anal	yses							
Choose a type		60 to	tal analys	es selei	ted (60 on this page)		1 -	60 of 60 < >	Operations	
 Analyses 			Status	0 8	Element	Name	Template	Backfilling	Enable Disable selected analyses	
 Notification Rules 			0		Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0382	Ex. 2 - Predict Load	Ex. 2 - Predict Load		and the second s	
Analysis Searches			ö		Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT XYZ0376		Ex. 2 - Predict Load		Enable Disable automatic recalculation for selected	
+ X			ŏ	fea			Ex. 2 - Predict Load		Backfill/Recalculate selected analyses	
All			õ	fee			Ex. 2 - Predict Load		Backfill/Recalculate selected analyses	
			Ö	fee			Ex. 2 - Predict Load		Start 6/15/2017 12:00:00 AM	
Enabled	•		ŏ	fea			Ex. 2 - Predict Load		End 8/31/2017 12:00:00 AM	
Disabled	•		ŏ	fea			Ex. 2 - Predict Load		End citration of the	
Example 2 Prediction	1		õ	fee			Ex. 2 - Predict Load		What should we do with existing data?	
Example 3 Forecast	-		õ	fee			Ex. 2 - Predict Load		 Leave existing data and fill in gaps 	
	1		ŏ	fea			Ex. 2 - Predict Load		Permanently delete existing data and recalcula	
			ŏ	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT XYZ0346	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Recalculate dependent analyses	
			Ö	f60	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT XYZ0345	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Queues.	
			Ö	fee	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0344	Ex. 2 - Predict Load	Ex. 2 - Predict Load		high states and states	
		2	•	•	ö	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0379	Ex. 2 - Predict Load	Ex. 2 - Predict Load	
							Ö	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0420	Ex. 2 - Predict Load
			0	f(+)	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0402	Ex. 2 - Predict Load	Ex. 2 - Predict Load	_	Pending Operations	
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0388	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Enabling 60 analyses D	
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Y Phase\PT_XYZ0400	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Time submitted: 12/20/2018 11:57:59 AM	
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0399	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Queued by: PISCHOOL\student01	
			0	f(+)	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Y Phase\PT_XYZ0353	Ex. 2 - Predict Load	Ex. 2 - Predict Load		Complete	
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0398	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0343	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
			Ö	fte	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0390	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
Ilements	_		0	f(s)	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0409	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
- Event Frames			0	fee	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0394	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
			0	fw	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0414	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
🗊 Library			0	f()	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0384	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
Unit of Measure			0	f(+)	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0404	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
Contacts			0	f(s)	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0408	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
	_		9	f⇔	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0391	Ex. 2 - Predict Load	Ex. 2 - Predict Load			
*			0	fea	Alaiuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT XYZ0385	Ex. 2 - Predict Load	Ex. 2 - Predict Load	~		

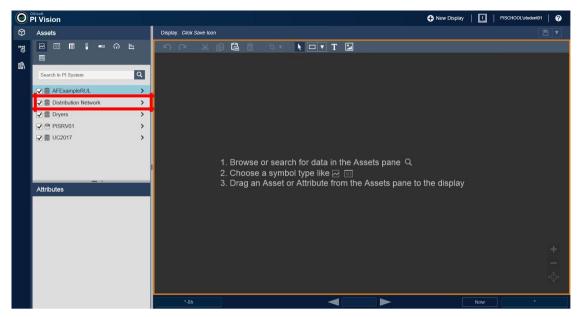
g. Once all 60 analyses have been backfilled, click **Disable** to stop them. We are only testing our new model over a controlled time range. It's not ready for general use yet.



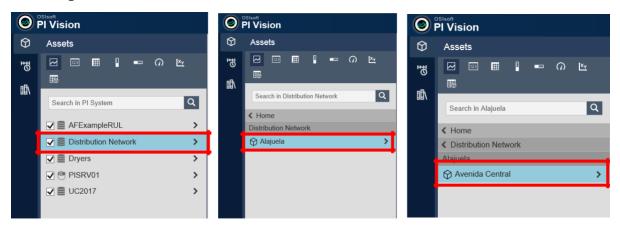
Evaluate Model – PI Vision Collection

PI Vision provides a powerful feature called a "Collection" which makes it easy to group visualization objects and apply them to a set of AF Elements derived from the same AF Template. In this step, we will be creating a PI vision collection to show the backfilled results for all 60 transformers in our network.

a. Open PI Vision using the "Ex. 2 Build Display" shortcut on the desktop. This will open a new display.



b. In the Asset pane on the left, select the Distribution Network AF model and navigate the model's hierarchy by selecting the arrow icon, Drilldown the tree using the following path, Distribution Network \ Alajuela \ Avenida Central \ Transformer 1 \ Colegio Cientifico \ X Phase.



0	PI Vision	۲	PI Vision	PI Vision				
Θ	Assets	Θ	Assets	Θ	Assets			
₩		₽		ち				
881	Search in Avenida Central	881	Search in Transformer 1	887	Search in Colegio Científico			
					< Home +			
	< Home		< Home +		< Avenida Central			
	< Distribution Network		< Alajuela		< Transformer 1			
	Alajuela		< Avenida Central		Colegio Científico			
			Transformer 1		M Weather Conditions			
	Ovenida Central				😚 X Phase 📏			
- 4					Y Phase			
					😚 Z Phase >			

c. Select the transformer element **PT_XYZ_0343**. From the attributes list below, select and drag the **Wh Delivered Load** attribute on to the display canvas. This will add a trend of the transformer loading on the display. You won't see data until we reset the display time range to match our dataset, 6/15/17 to 8/31/17.

۲	osisoft PI Vision		+ New Display 🚺 PISCHOOL\student01 ?
Θ	Assets	Display: Click Save Icon* Asset: PT_XYZ0343 ▼	💭 Ad Hoc Display
墙			
_	85	C Wh Delivered Load	
081	Search in X Phase Q	- 2,891.8 2,891 Wh - 2,891.6	
		-2,891.4	
	Home + Transformer 1	- 2,891 2	
	Colegio Científico	0 ^{2,091}	
	X Phase	- 2,890.6	
		- 2,890.4 - 2,890.2	
	♥ P1_XYZ0379	2,890	
		C/1//2019 10 57:55 FM 8h 0 1/18/2019 6:57:55 AM 0	
	Attributes		
	Voltage Average		
	📄 Voltage Maximum		
	🔲 Voltage Minimum		
	Voltage Quality		
	Wh Delivered Load		
	🔲 Wh Delivered Load - 14d		
	Wh Delivered Load - 7d		
	Ex. 2 Load Prediction		
	Coefficient_0		
	Coefficient_1	1/17/2019 10:57:55 РМ 🛛 🖌 🕨	Now 1/18/2019 6:57:55 AM

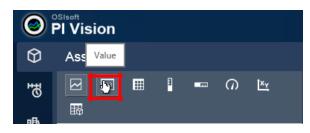
d. Edit the start and end times at the bottom of the PI Vision display to be **6/15/17** and **8/31/17**.

	Pl Vision		🕂 New Display 📗 📗	PISCHOOL\student01 0
0	Assets	Display: Click Save Icon* Asset. PT_XYZ0343 ▼	🐺 Ad Hoc Di	splay 🚺 🖭 🔻
₩		∽○₭▣₲₫ ◣□▾Тਯ		
0ØA	- III	25,000 Wh Delivered Load		
ARI/	Search in X Phase	20.000		
	K Home +			
	C Transformer 1			
	< Colegio Científico			
	X Phase			
		-2,500		
		6/15/2017 12:00:00 AM 77d 8/31/2017 12:00:00 AM		
		1		
		[™]		
	Attributes			
	Voltage Average			
	Voltage Maximum			
	Voltage Minimum			
	Voltage Quality			
	Wh Delivered Load			
	Wh Delivered Load - 14d			
	Wh Delivered Load - 7d			
	Ex. 2 Load Prediction			
	Coefficient_0			
	Coefficient_1	✓ 6/15/2017 12:00:00 AM 🖉 77d ►	Now	8/31/2017 12:00:00 AM

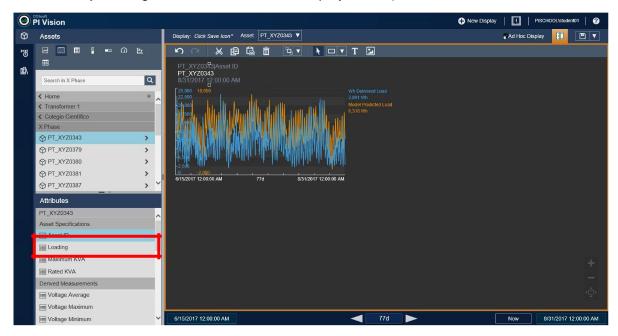
e. Scroll down the attributes list and find the attribute **Model Predicted Load**. Select and drag this attribute onto the display canvas and drop it on the trend object. You should see both traces on the trend.

0	Piston		🕂 New Display	🛄 PISC	HOOL\studen	101 0
Θ	Assets	Display: Click Save Icon* Asset. PT_XYZ0343 ▼		Ad Hoc Display		
178		∽○╳皥╚亩 ◣□▾┰ы				
nfis	15	925.000 18,000 Wh Delivered Load				
081	Search in X Phase	22,000 Vice 2,001 Vice				
	< Home +					
	< Transformer 1					
	Colegio Científico X Phase					
	A Phase	Pette all a de contra con la calina de la cali				
	© PT_XYZ0379 >					
		6/15/2017 12:00:00 AM 77d 8/31/2017 12:00:00 AM 🛛				
	Attributes					
	Ex. 2 Load Prediction					
	Coefficient_0					
	Coefficient_1					
	Coefficient_2					
	Coefficient_3					
	🔲 Intercept					
	Model Predicted Load					
	EX. 3 Load Forecasting					
	Model Forecast Load		r.			
	Network	6/15/2017 12:00:00 AM 77d		Now	/31/2017 12	2:00:00 AM

f. At the top of the **Asset** pane, change the default display object type from a Trend to a Value.



g. Scroll up the attributes list and find the attribute **Asset ID**. Select and drag this attribute onto the display canvas above the trend. The transformer's asset name should appear along with the attribute name and timestamp. (You may need to rearrange the trend and value objects to get them to look like the display below.)



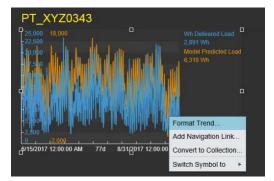
h. Select he value object, right-click on it and choose Format Value.

PT_XYZ03		
PT_XYZ0343 8/31/2017 12	Format Value	
□ [25,000 18,000	Add Multi-State	Wh Delivered Load
-22,500	Add Navigation Link	2,891 Wh
-20,000	Convert to Collection	Model Predicted Load 6,318 Wh
- Mobel . I I	Switch Symbol to	
- Friendly I I P	an a Jing Ping Dar Aliyaka ya	
	VI I MANANA I MILANA INA MINANA MILANA DI KAN	
-2.500 'INP	an de san statue de anders de	
02.000		
6/15/2017 12:00:0	0 AM 77d 8/31/2017 12:00:00 AM	

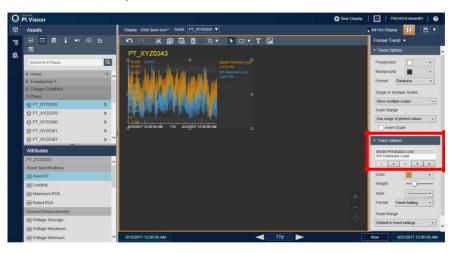
i. In the **Format Value** pane at right, expand the **Visibility** section and uncheck the Label, Units and Timestamp items. In the **Style** section, change the **Font Size** to **18** and the **Value** color to yellow.

0	PI Vision		New Display	🚺 РІЗСНОО	L\student01	0
•	Assets	Display. Click Save Icon* Asset. PT_XYZ0343 ▼		Ad Hoc Display	1	
***				Format Value		
	10	РТ_ХҮД20343		 Style 		
081	Search in X Phase			Fil		*
	K Home +	25,000 18,000 22,500 1, 2,091 Wh		Text		
	Transformer 1 Colegio Científico X Phase	Month Treatment Last	Î	Font Size	18	×
	PT_XYZ0343 >			Value		*
	© PT_XYZ0379 >		•	Format	Default	~
	PT_XYZ0380	Foll all an at a second to addition of all all all all all all all all all al				
		0 2 000 6/15/2017 12:00:00 AM 774 8/31/2017 12:00:00 AM		Text Alignment	Left	×
		9192211 123030 PM 719 93312017 123030 PM	1	 Visibility 		
	Attributes			Label		
	PT_XYZ0343			Units		
	Asset Specifications			Timestamp		
	Asset ID			✓ Value		
	E.oading			V value		
	Maximum KVA		+			
	Rated KVA		÷			
	Derived Measurements					
	Voltage Average		- Či			
	Woltage Maximum					
	Voltage Minimum	6/15/2017 12:00:00 AM 🗧 77d 🕨		Now 8/31/	2017 12:00:0	D.AM

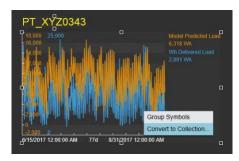
j. Select he value object, right-click on it and choose Format Trend.



k. In the **Format Trend** pane at right, in the **Trace Options** section, move the **Model Predicted Load** attribute up to the first trace in the trend.



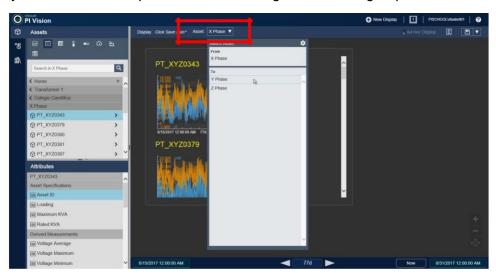
I. Holding the Ctrl key, multiselect the Value and Trend objects. Right-click and choose **Convert to Collection**.



m. Once the display updates, click on the PI Vision display mode icon in the upper right-hand corner of the bowser to switch to the **Monitor Operations** mode.



n. Using the display scrollbar, you will be able to see backfilled results and actual values for each transformer. These will be grouped by the phase, or parent, level of the Distribution Network model's hierarchy. Using the **Asset** dropdown list at the top of the display, select a different phase to different to get a different group.



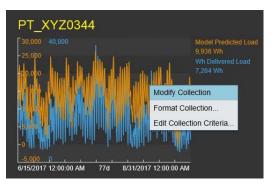
- o. Adjustments can be made to make the display easier to use. Here are some suggestions;
 - 1. Close the Asset pane by clicking on its icon. This will give you more display space.



2. Change the display back to **Modify Display** mode, select the Collection and stretch it out to show more assets.



3. Right-click on the collection and choose **Modify Collection**. This will return you to the original group containing the original Value and Trend objects. Here you can resize them and change their formatting.



Click the Exit Modify Mode icon in the upper right-hand corner of the object



boundary to return to the full display.

Example 3 - Operationalize Forecast Models for Multiple Transformers

Learning Objectives

- Modify the predictive equation to calculate forecasted values for transformer loads
- Employ Advanced scheduling feature of AF Analytics and PI Future data tags.
- Operationalize forecasting model (assume today is 8/31/17)
- Asses the forecast for all 60 transformers in PI Vision Collection

Create a Forecast from the Predictive Model

 a. Go to the Library view in PI System Explorer (lower left-hand corner). Select the Single Phase Transformer template. Click the Analysis Templates tab. Choose the "Ex. 3 Forecast Load" analytic.

0	\\PISRV01\Distribution Network - PI System Explorer (Administrator)	_ _ X
File View Go Tools Help 3 Database 🛅 Query Date 🔹 🕔 🥥 Back	🗊 😼 Check in 🧐 🖌 🗃 Refresh 🛛 New Template 📼	Search Element Templates 🔎
Library	Single Phase Transformer	
Construction Network Categories Construction Network Categories Categor	General Attribute Templates Ports Analysis Templates Notification Rule Templates Image: State S	on () Rollup ame Generation () SQC
	Add a new variable	Evaluate
	Name Expression // Check that input values are all good before making forecast. IF (MOT Badval(TagVal('Mb Delivered Load', "*-7d+lh'))) AND (NOT AND (NOT BadVal(TagVal('Ambient Temperature', "*+1h'))) AND (NOT Bitter (Coefficient_2' * TagVal('Mb Delivered Load', "*-7d+lh')) Forecast + 'Coefficient_2' * TagVal('Ambient Temperature', '*+1h') + 'Coe' + ('Intercept') // One more bad values, do nothing. Else MoOutput()	adVal(TagVal(' pefficient_1'
🗊 Elements		
Event Frames		
🔛 Library		
🚥 Unit of Measure		
38 Contacts	Scheduling: Event-Triggered Periodic Advanced	
*	Period: 01h 00m 00s Configure Output time stamp override: *+1h	

b. Take a closer look at the analytic equation below. First, we have added a conditional check to ensure all forecast-based measurements are good. This reduces the chance of posting an errant forecast value which can erode user confidence.

Secondly, we have shifted the time references forward one hour. This change requires the terms of the modelling equation to be based on the PI Point attribute, Wh Delivered Load using the TagVal() function in AF Analytics.

Name	Expression
Forecast	<pre>// Check that input values are all good before making forecast. IF (NOT BadVal(TagVal('Wh Delivered Load', '*-7d+1h'))) AND (NOT BadVal(TagVal('Wh Delivered Load', '*-14d+1h'))) AND (NOT BadVal(TagVal('Ambient Temperature', '*+1h'))) AND (NOT BadVal(TagVal('Relative Humidity', '*+1h'))) THEN 'Coefficient_0' * TagVal('Wh Delivered Load', '*-7d+1h') + 'Coefficient_1' * TagVal('Wh Delivered Load', '*-14d+1h') + 'Coefficient_2' * TagVal('Ambient Temperature', '*+1h') + 'Coefficient_3' * TagVal('Relative Humidity', '*+1h') + ('Intercept') // One or more bad values, do nothing. Else NoOutput()</pre>

c. By default, AF Analytics posts results at the current time. This analytic is scheduled to run hourly, however, as a forecast, we need to be able to post the results, one hour into the future. Click on **Advanced** to access the dialog to change the result posting rime.

Event Frame Templates Model Templates	Add a new v	Add a new variable				
🗄 – 🍓 Transfer Templates	Name	Expression	Output Attribute			
Commeration Sets Serveration Sets Serverations Tables Categories Analysis Categories Analysis Categories Analysis Categories Analysis Categories Analysis Categories Analysis Categories Table Categories Table Categories Table Categories Table Categories Table Categories	Forecast	<pre>// Check that input values are all good before making forecast. IF (NOT BadVal(TagVal('Mb Delivered Load', '*-7d+1h')) AND (NOT BadVal(TagVal) AND (NOT BadVal(TagVal('Ambient Temperature', '*+1h')) AND (NOT BadVal(TagVal) THEN 'Coefficient_0' * TagVal('Mh Delivered Load', '*-7d+1h') + 'Coefficient_1' + 'Coefficient_2' * TagVal('Ambient Temperature', '*+1h') + 'Coefficient_3' * +('Intercept') // One or more bad values, do nothing. Else NoOutput()</pre>				
🗊 Elements						
🕌 Library						
🚥 Unit of Measure						
A Contacts	Scheduling:	Event-Triggered Periodic Advanced				
*	Period: 01h 00	m 00s Configur Advanced configuration options inc samp override: *+1h				

d. We have specified that the output of this analytics be posted in a future data tag one hour past the time of its calculation.

Ę	Advanced options							
	Output Time Stamp							
	 Trigger Time Execution Time 							
	Relative to Trigger Time: *+1h							
	○ Variable:							
	Mapped output attributes require a future PI point to save output history. For newly mapped output attributes that save output history, PI Analysis Service will create future PI points.							
	- Automatic Recalculation							
	Recalculate analysis for out-of-order input events							
	OK Cancel							

e. We are now ready to test the forecasting analyses. Go to the **Management** view in PI System Explorer (icon in lower left-hand corner). Select **Ex. 3 Load Forecast** under Analysis Searches. Click the checkbox at the top next to Status to select all 60 forecasting analytics. Start them by clicking **Enable**.

File View Go Tools Help										
🗿 Database 🛗 Query Date 🔹 🕔 🤹	G Back 🔘	1 , C	neck In 🍤	1	Refresh					
Management		Analyses								
Choose a type		60 total analyses selected (60 on this page) 1 - 60 of 60 < > Operations								
 Analyses 			Status 🔕	в	Element	Enable Disable selected analyses				
O Notification Rules			0	f60	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0 -					
Analysis Searches			Ø	f60	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	Enable Disable automatic recalculation for selected analysis				
+×			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	Backfill/Recalculate selected analyses				
All	*		Ø	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0					
Enabled			ø	100	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0					
			Ø	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
Disabled	*		0	500	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0 =					
Example 2 Prediction	1		0	100	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0					
Example 3 Forecast	1		0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
	-		0	500	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
			0	500	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XY20					
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XY20					
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0					
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XY20	Pending Operations				
			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	No pending operations				
			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
		\checkmark	0	ftə	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0					
			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0					
J Elements			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
- Event Frames			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
			0	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0					
Library			0	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0					
Duit of Measure			0	f00	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0					
Contacts			0	fto						
Management			0	560	Alajuela/Avenida Central/Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0 ~					

f. Once you get green icons indicating all the analytics have been started, Backfill, or in this case "Forewardfill", the analysis from 8/31/17 to 9/10/17. (Remember, today is 8/31/17 and we have weather forecast data through the end of September 2017.)

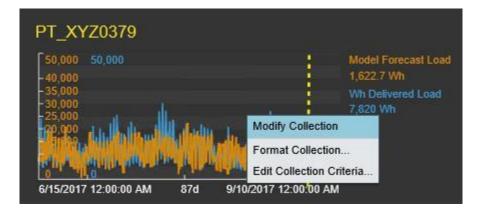
File View Go Tools Help	G Back 🔘 💐	Check	In 1	1	2 Refresh	
Management	-	nalyses				
Choose a type	6) total a	nalyses	selec	ted (60 on this page) 1 - 60 of 60 < >	Operations
Analyses	T.	Sta	tus 🖸		Element	Enable Disable selected analyses
Notification Rules		1 0		fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0 ~	Enable Disable automatic recalculation for selected analys
Analysis Searches		1 0)	f60	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	
+ ×		1 0)	fee	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	Backfill/Recalculate selected analyses
All	•	1 0		foo	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	Start 8/31/17
Enabled		10)	f00	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0	10002
	~	1 0		feo	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	End 9/10/17
Disabled	•	10		_f60	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0	What should we do with existing data?
Example 2 Prediction	1	10		f00	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	 Leave existing data and fill in gaps
Example 3 Forecast	1	10)	f00	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	Permanently delete existing data and recalculate
		1 0)	f00	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0	Recalculate dependent analyses
		10)	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	Queue
		10		foo	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	quar
	1	1 0		foo	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0	
		1 0)	feo	Alajuela\Avenida Central\Transformer 1\Colegio Científico\Z Phase\PT_XYZ0	Pending Operations
	1	10)	ft0	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	
		1)	fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	Enabling 60 analyses Dism Time submitted: 1/18/2019 8:41:15 AM
	1	1 0		fee	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	Queued by: PISCHOOL\student01
	1	1 0		fee	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	Complete
	1	1 0		f00	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0	
		1 0		fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	Recalculating 60 analyses Dism
Elements		10		fto	Alajuela\Avenida Central\Transformer 1\Colegio Científico\X Phase\PT_XYZ0	Time submitted: 1/18/2019 8:41:57 AM Oueued by: PISCHOOL\student01
		2 0		ft)	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	43.3
- Event Frames		10		foo	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\X Phase\PT_XYZ0	
🛿 Library	[1		fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Y Phase\PT_XYZ0	
Dunit of Measure	[1 0		fto	Alajuela\Avenida Central\Transformer 1\Colegio Cientifico\Z Phase\PT_XYZ0	
Contacts		2 0		fto		
10	[[1		fto	Alajuela \Avenida Central \Transformer 1 \Colegio Científico \Z Phase \PT_XYZ0 \lor	
& Management		1			III >	

g. Once the backfill is complete, **Disable** the analyses. Return to the desktop and open PI Vision using the desktop shortcut named "**Ex. 3 Forecast Evaluation**".

O PI	Vision	- 1			Đ N	ew Display 🛛 🚺 🕴 Pt	SCHOOL/student	01 🛛 🕜
Ø	Ex. 3 Forecasting Model Evaluation Asset: X Phase ▼					CAd Hoc Displ		
뻅								
081	PT_XYZ0343		PT_XYZ0379		PT_XYZ0380			^
	6152017 122000AM 876 9102017 122000AM	Model Forecast Load -338,57 Wh Whi Delivered Load -5,000 Wh	50.000 -40.000 -40.000 -50.000 -25.0000 -25.0000 -25.0000 -25.0000 -25.0000 -25.0000 -25.000 -25.000	Model Forecast Load 1,622.7 Wh Wh Delivered Load 7,820 Wh	C	Model Forecast Load -1,582.2 Wh Wh Delivered Load 5,788 Wh		
	PT_XYZ0381		PT_XYZ0387		PT_XYZ0388			
	50,000 50,000 40,000 50,000	Model Forecast Load -200.23 Wh Wh Delivered Load 3,138 Wh	50,000 40,000	Model Forecast Load 5,281.8 Wh Wh Delivered Load 10,539 Wh	50,000 40,000	Model Forecast Load 3,607.3 Wh Wh Delivered Load 8,971 Wh		
	PT_XYZ0389		PT_XYZ0390		PT_XYZ0393			
	50,000 50,000	Model Forecast Load 9,543 Wh Wh Delivorid Load 15,999 Wh	50,000 50,000 +0,000 -	Model Forecast Load 2,192.4 Wh Wh Delivered Load 4,076 Wh	50,000 50,000 -40,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -30,000 -10,000 -30,000 0 -30,000 6/15/2017 12:00:300 AM 57d 9/10/2017 12:00:300 AM 57d	Model Forecast Load 823-46 Wh Wh Delivered Load 661 Wh		⊕ 1 +
	6/15/2017 12:00:00 AM		8 7d			Now	9/10/2017 12	00:00 AM

h. The dashed yellow lines have been drawn over the trends to show "today", 8/31/17. The forecasted values we have just generated are shown to the right of this line as future data. Let's see how the Model Forecast Loads are affected by the cooler ambient temperatures forecasted for September.

Right-click on any trend in the PI Vision Collection and select Modify Collection.



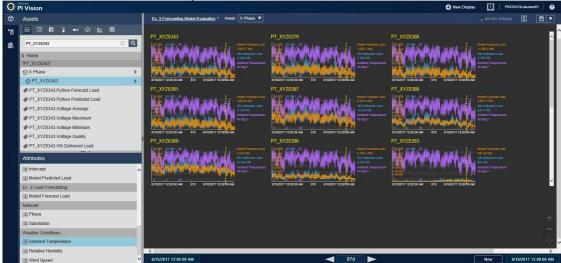
- Θ
- i. Expand the "Assets" pane by clicking in the upper left-hand corner of the PI Vision portal. In the Search dialog type the name of base element used to configure the PI Vision collection, **PT_XYZ0343**. Scroll down the Attributes list and find **Ambient Temperature**.

0	Sisoft PI Vision		New Display PISCHOOL/student01
Ø	Assets	Ex. 3 Forecasting Model Evaluation Asset X Phase V	🗸 Ad Hoc Display 👔 🔳 💌
178			
۵Ø۸	PT_XYZ0343 💿 🔍	PT XY20343	^
	K Home "PT_XYZ0343"	15.00 Hosts Ferent Last 15.00 -23.57 VA 15.00 -W Defined Last	₽
		-3,000 -25,000 47,000 ML, N, N, S, K	
	PT_XYZ0343		
	PT_XYZ0343.Python Forecast Load	4/15/2017 12:00:00 AM 870 9/10/2017 12:00:00 AM	
	PT_XYZ0343.Python Predicted Load		
	PT_XYZ0343.Voltage Average		
	PT_XYZ0343.Voltage Maximum		
	PT_XYZ0343.Voltage Minimum		
	PT_XYZ0343.Voltage Quality		
	PT_XYZ0343.Wh Delivered Load		
	Attributes		
	Intercept		
	Model Predicted Load		
	Ex. 3 Load Forecasting		
	Model Forecast Load		
	Network		
	E Phase		+
	Substation		· · · ·
	vicauler Conditions		
	E Ambient Temperature		<u>بَ</u>
	E Relative Humidity		>
	Wind Speed	6/15/2017 12:00:00 AM 🗧 87d	Now 9/10/2017 12:00:00 AM

j. Select and drag the **Ambient Temperature** attribute onto the trend object in the collection. This will add the temperature forecast data, both historical and future, to the trend.

lision		New Display PISCHOOLteb.
Assets	Ex 3 Forecasting Model Evaluation * Asset: X Phase V	Ad Hoc Display
PT_XYZ0343	O Q PT_XYZ0343	
< Home "PT_XY20343"	N. AWANA ANA ALA SALAN	
	> The Add and Add Add Add Add Add Add Add Add Add A	
@ PT_XYZ0343		
PT_XYZ0343.Python Forecast Load	d-15/2017 12:00:00 AM 678 9102017 12:00 D0 AM 0	
PT_XYZ0343 Python Predicted Load		
PT_XYZ0343.Voltage Average		
PT_XYZ0343.Voltage Maximum		
PT_XYZ0343.Voltage Minimum		
PT_XYZ0343.Voltage Quality		
PT_XYZ0343.Wh Delivered Load		
Attributes		
Im Intercept		
Model Predicted Load		
Ex. 3 Load Forecasting		
Model Forecast Load		
Network		
Phase		
E Substation		
Weather Conditions		
Ambient Temperature		
III Relative Humidity		
I Wind Speed	G/15/2017 12:00:00 AM	87d > 9/10/201

k. Return PI Vision to "Motoring Operations" mode by clicking in the upper fighthand corner of the PI Vision portal.



I. Collapse the "Assets" pane to give you a better look. You can double-click on any individual trend for a more detailed examination of how things look.

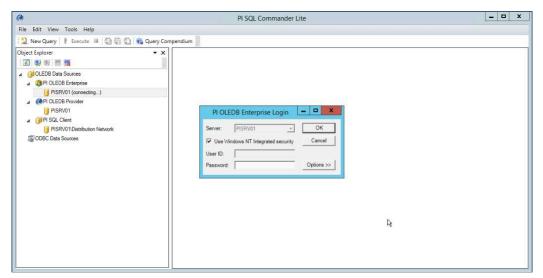
Appendix A - Python Access to PI using PI OLEDB Enterprise

Create a PI View – PI OLEDB Enterprise

PI Data can be imported directly into Python through a view configured in PI OLEDB Enterprise. Views are configured in PI SQL Commander Lite. Here are the steps to create a view containing the data used in the lab example.

- a. Open PI SQL Commander Lite from the Windows taskbar,
- b. Under the **PI OLEDB Enterprise** branch of the Object Explorer hierarchy, select the **PISRV01**, right-click and **Connect.** Click **Ok**.

1



c. Expand the **Distribution Network** catalog in Object Explorer hierarchy. Expand the **Data T** folder. Then right-click on the Data T folder and slide through the menu to select **Dynamic Transpose Function (Asset)**.

PI SQL Commander Lite	 x
File Edit View Tools Help	
🕄 New Query 👔 Execute 💷 🚳 🖓 🖏 🔇 ee Query Compendium	
Object Explorer	
New Transpose Function Dynamic Transpose Function (Asset) Export Custom Database Objects Dynamic Transpose Function (Event Frame) Create a transpose function with all the attributes of an element template. The transpose function with all the attributes are added, removed, or renamed. b Custom Transpose Function (Event Frame)	
▷ U Clysis ▷ System ▷ U C2017 ▲ @ PI OLEDB Provider □ PISRV01 ▲ @ PI SQL Client	

d. In the Transpose Function Wizard dialog choose **Transpose Interpolate Range**. Click **Next**.

	Transpose Function Wizard
CO. OSIS	
Element Template Definition	Select transpose function: Transpose Snapshot Transpose Interpolate Discrete
Options	Transpose Interpolate Range
Summary	Transpose Archive
Execution	Versioned Transpose Snapshot
	Versioned Transpose Interpolate Discrete
	Versioned Transpose Interpolate Range
	Versioned Transpose Archive
	Next > Cancel

e. Choose Single Phase Transformer. Click Next.

	Transpose Function Wizard
osi	soft.
Transpose Function	Select Element Template:
Element Template	Circuit
Definition	Headquarters
Options	HMM Single Phase Hourly
Summary	Meter
Execution	Phase
	Secondary Transformer
	Single Phase Transformer
	Single Transformer Substation
	Transformer Bank
	Weather
	The other
	< Back Next > Cancel

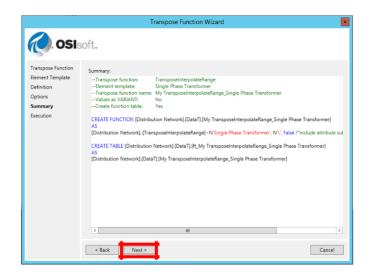
f. Change the function name to "My TransposeInterpolateRange_Single Phase Transformer". Click Next.

	Transpose Function Wizard	×
🕢 . OSI	soft.	
Transpose Function Element Template Definition Options Summary Execution	Transpose function name: My/[TransposeInterpolateRange_Single Phase Transformer Select Attribute Path: \	
	Cancel Cancel	

g. Click Next.

	Transpose Function Wizard	×
🕢 osi	soft.	
Transpose Function Element Template Definition Options Summary Execution	Options: ☐ Values as VARIANT ☑ Create function table	
	< Back Next > Conce	4

h. Click Next.



i. Click **Execute**.

	Transpose Function Wizard	I	
🥢 OSIs	oft.		
Transpose Function Element Template	Progress:		
Definition	Details:		
Options	Element Template	Function	Table
Summary Execution	TransposeInterpolateRange		
Execution	Single Phase Transformer		
	Execute		Cancel

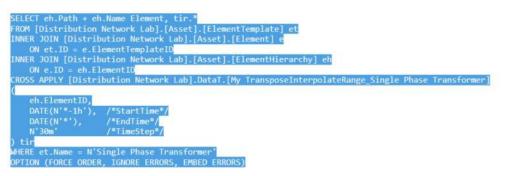
j. Click Done.

	Transpose Function	Wizard	×
🥢 OSIs	oft.		
Transpose Function Element Template Definition	Execution completed succesfully		
Options	Details:		
Summary	Element Template	Function	Table
Execution	TransposeInterpolateRange		
	Single Phase Transformer	0	•
			Done

k. Expand the **Table-Valued Functions** folder to see you newly created function. Rightclick on **My TransposeInterpolateRange_Single Phase Transformer** and **Execute Predefined Query...**

9	PI SQL Commander Lite		-	• ×
File Edit View Query Tools Help				
2 New Query Execute # 🔄 🌄 🔄 😜 Query Compendium				
Object Explorer • x	Guery1.sql - PISRV01 ×			
2 部方市西	SELECT eh.Path + eh.Name Element, tir.*			1
(CLOB Devisions (Process) (Process) (Clob Devisions (Clob Devisio	<pre>FION [Distribution Network].[Ainset].[Linear/Implicit] et mices 300% [Distribution Network].[Ainset].[Linear/Implicit] et mices 300% [Distribution Network].[Ainset].[Linear/Implicit 00 et al. = et Linear/Implicit.[Network].[Linear/Implicit 01 for the set Linear/Implicit.[Network].[Linear/Implicit 0100 (Network].//Stattlan// Network]. Distribution.[Network].[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network]. Distribution.[Network].[Network]. Distribution.[Network].[Network]. Distribution.[Network].[Network].[Network]. Distribution.[Network].[Network].[Network]. Distribution.[Network].[Net</pre>	and the second second second second	le Phase Transforme	e] ,
) 🔄 Vevs	Florent	Time	Intercept Coefficient 3	Coeffici n
Cash Point Function Cash Point Point Cash Point Point	Visperkalvenis Cenet Tenterer (Colgo Centrol 7 Neurf) (2013) Visperkalvenis Cenet Tenterer (Colgo Centrol 7 Neurf) (2013)	2011-09-25 (301 22,000 2018-09-25 (331 22,000 2018-09-25 (331 22,000 2018-09-25 (331 22,000 2018-09-25 (331 22,000 2018-09-25 (231 22,000) 2018-09-25 (231 22,000)	211318 0.1221081 211318 0.1231081 211318 0.1231081 183118 0.52526 185115 55454 395135 754654 395135 754654 495135 754654 495135 754654 -442233 258701 -46233 2587701 -561990 314753	0.2417 0.2417 0.2417 0.2417 0.2417 351.84 351.84 351.84 602.00 602.00 602.00 602.00 602.00 245.79 245.79 245.79 245.79 245.79 245.79 245.00
VISHV01Datributen Network				

I. Select the entire predefined query shown in the query viewer and copy it into the clipboard.



m. Under the Asset folder, find the Views folder. Right-click on Views select Create View.

Ø	PI SQL Commander Lite	- • ×
File Edit View Query Tools Help		
😫 New Query 🕴 Execute 💷 🦓 🏹 🍋 🤬 Query Compendium 📗		
Object Explorer 🔹 🗧	c 👔 Query1.sql - PISRV01 🗙	,
2 単型 目標	SELECT eh.Path + eh.Name Element, tir.*	
CLEDB Data Sources CLEDB Crange Sources Control CLEDB Crange Source Cranops Crano	<pre>StiC: dh.Path.s.dh.Made Lineart.trillamentTemplate1 et THMET SOUN [Distribution Network] [Aises1].[Element] = 000 et.lD = e.llementTemplateD DHET_SOUN [Distribution Network].Classe1.[ElementUrarchy] eh ON e.lD = eh.ElementD COSS APPLY_DIstribution Network].DataT.[Py TransposeInterpolateRange_Single Phase Trans (</pre>	former]
D = Pol Create View. Export Custom Database Objects. B = Funds Refresh B = Data	Pesude Time Intercept Coefficient 1 Wagesbluends Central Transformer 1Colego Cientificol 2 Phase PTYY20342 2018-05-312.312.0000 -2113118 0.123 2 Wagesbluends Central Transformer 1Colego Cientificol 2 Phase PTYY20342 2018-05-312.312.0000 -2113118 0.123 2 Wagesbluends Central Transformer 1Colego Cientificol 2 Phase PTYY20342 2018-05-312.312.000 -2113118 0.123	
TotaT Tables Totexs	3 Ulapado-levendo Central/Transformer TColegio Cientifico? Phase/PT_VYZ0042 2016/09-25 13:312.000 -21 13118 0.12 4 Ulapado-levendo Central/Transformer TColegio Científico? Phase/PT_VYZ0043 2018/09-25 12:312.2000 -183116 2023 5 Ulapado-levendo Central/Transformer TColegio Centifico? Phase/PT_VYZ0043 2018/09-25 12:312.2000 -183116 2033 5 Ulapado-levendo Central/Transformer TColegio Centifico? Nane/PT_VYZ0043 2018/09-25 13:012:000 -183116 2033	276 351.84 276 351.84
Table-Valued Functions EMy TransposeInterpolateRange_Single Phase Transformer ETransposeInterpolateRange_Single Phase Transformer	6 Waquebi/venida central Transformer TCodego Centitiox/P NeurIPT_X723Ha 2018-05-313312.2000 1831169 203 7 Waquebi/venida central Transformer TCodego Centitiox/P NeurIPT_X723Ha 2018-05-31312.12.000 351133 74 8 Waquebi/venida central Transformer TCodego Centitiox/P NeurIPT_X723H4 2018-05-31312.12.000 3551133 74 9 Waquebi/venida central Transformer TCodego Centitiox/P NeurIPT_X723H4 2018-05-31312.2000 3551133 74 9 Waquebi/venida Central Transformer TCodego Centitiox/P NeurIPT_X723H4 2018-05-31312.12.000 3551133 74	542 602.00 542 602.00
Car EventFrame Dryers	10 Wagekh/wends central Transformer TCodego Centificior / Pasae/FT_XX2034 2019.09.3512.312.200 +1422.33 26.7 11 Wagekh/wends Central Transformer TCodego Centificior / Pasae/FT_XX2034 2019.09.3512.012.200 +1422.33 26.7 12 Wagekh/wends Central Transformer TCodego Centificior / Pasae/FT_XX2034 2019.09.3513.012.200 +1422.33 26.7 13 Wagekh/wends Central Transformer TCodego Centificior / Pasae/FT_XX2034 2019.09.3513.122.200 >1422.33 26.7 13 Wagekh/wends Central Transformer TCodego Centificior / Pasae/FT_XX2034 2019.09.3513.122.200 >149.233 26.7	701 245.79 701 245.79
System UC2017	13 Valapide Juenda Central Transformer T Collegio Gentifico IP Inseet PT/V20346 2016/06/25 13 01 20300 26399 08 33 47 14 Valapide Juenda Central Transformer T Collegio Gentifico IP Inseet PT/V20346 2016/06/25 13 01 20300 26399 08 33 47 15 Valapide Juenda Central Transformer T Collegio Gentifico IP Inseet PT_V20346 2016/06/25 13 01 20300 26399 08 33 47	753 466.00
PISRV01	Query executed successfully PISRV01 PI OLEDB Enterprise 00.003	05.437 180 row

n. The following query should now appear in the query window.

```
CREATE VIEW [Distribution Network].[Asset].[<view name>]
AS
<view definition>
```

o. Replace <view definition> with the query saved in the clipboard

```
CREATE VIEW [Distribution Network].[Asset].[<view name>]
AS
SELECT eh.Path + eh.Name Element, tir.*
FROM [Distribution Network].[Asset].[ElementTemplate] et
INNER JOIN [Distribution Network].[Asset].[Element] e
   ON et.ID = e.ElementTemplateID
INNER JOIN [Distribution Network]. [Asset]. [ElementHierarchy] eh
    ON e.ID = eh.ElementID
CROSS APPLY [Distribution Network].DataT.[My TransposeInterpolateRange Single Phase Transformer]
0
    eh.ElementID,
   DATE(N'*-1h'), /*StartTime*/
   DATE(N'*'), /*EndTime*/
N'30m' /*TimeStep*/
) tir
WHERE et.Name = N'Single Phase Transformer'
OPTION (FORCE ORDER, IGNORE ERRORS, EMBED ERRORS)
```

- p. Modify this query by;
 - a. Replacing, within the brackets, "<view name>" with "My Pole Transformer Loads".
 - b. Replace the SELECT statement by cut-and-pasting the following text:

eh Name, tir. Time Timestamp, tir. Loading, tir. [Maximum KVA], tir. [Rated KVA], tir. [Wh Delivered Load], tir. [Wh Delivered Load - 7d], tir. [Wh Delivered Load -14d], tir. [Ambient Temperature], tir. [Relative Humidity], tir. [Wind Speed]

- c. Change the Start Time from 'y' to '15-jun-2017'.
- d. Change the End Time from 't' to '31-aug-2017'.
- e. Change the Time Step from '30m' to '1h.
- q. When finished with the edits, the query should look like the one below.

```
CREATE VIEW [Distribution Network].[Asset].[My Pole Transformer Loads]
AS
SELECT eh.Name, tir.Time Timestamp, tir.Loading, tir.[Maximum KVA], tir.[Rated KVA],
tir.[Wh Delivered Load], tir.[Wh Delivered Load - 7d], tir.[Wh Delivered Load - 114d],
tir.[Ambient Temperature], tir.[Relative Humidity], tir.[Wind Speed]
FROM [Distribution Network].[Asset].[ElementTemplate] et
INNER JOIN [Distribution Network].[Asset].[Element] e
    ON et.ID = e.ElementTemplateID
INNER JOIN [Distribution Network].[Asset].[ElementHierarchy] eh
    ON e.ID = eh.ElementID
CROSS APPLY [Distribution Network].DataT.[My TransposeInterpolateRange_Single Phase Transformer]
(
    eh.ElementID.
    DATE(N'06/15/17'), /*StartTime*/
    DATE(N'08/31/17'),
                          /*EndTime*/
                  /*TimeStep*/
    N'1h'
) tir
WHERE et.Name = N'Single Phase Transformer'
OPTION (FORCE ORDER, IGNORE ERRORS, EMBED ERRORS)
```

r. Either right-click on the query and select **Execute** or click on the **Execute** button on the top menu bar to run this query and to create the view. You should wee the following message in the Results grid and a green light in the bottom message bar of PI SQL Commander Lite.

Messages		0
Command completed successfully	$\mathbf{\nabla}$	Query executed successfully

s. Return to the Views folder in the Object Explorer hierarchy. You may need to right-click on the view folder and **Refresh** it to see your new view. Right-click on the view, **My Pole Transformer Loads**, and **Execute Predefined Query...**

0			PI SQL Comm	iander Lite					-		x
File Edit View Query Tools Help											
😫 New Query 🕴 Execute 💷 🦓 🏹 🖏 🎕 Query Compendium 📗											
Object Explorer	×	n c	Query3.sql - PISRV01	1 📄 Query1.sq	I - PISRVO	1 X					Ŧ
	Шľ	SELL	ECT TOP 100 *								
OLEDB Data Sources	ā.			n Network].[Ass	et].[My	Pole Transf	ormer Load	s]			<u> </u>
PI OLEDB Enterprise											
PISRV01											
A Catalogs											
AFExampleRUL											
J Distribution Network											
A Asset											
Tables											
⊿ 🐚 Views											
My Pole Transformer Loads											-
Pole Transformer Loads		1999.0									
Table-Valued Functions	=		Results 📑 Message			1			1		_
Function Tables				estamp	Loading	Maximum KVA	Rated KVA	Wh Delivered Load	Wh Delivered Load - 7d	Wh Delin	
Data		1		17-06-15 00:00:00.000 17-06-15 01:00:00.000		0.1	10	365	370	369 370	Ξ
a 🗀 DataT		2		17-06-15 02:00:00.000		0.1	10	364	370	370	
Tables		4		17-06-15 03:00:00.000		0.1	10	364	346	371	1 11
Ca Views		5		17-06-15 04:00:00.000		0.1	10	0	0	7	
Table-Valued Functions		6		17-06-15 05:00:00.000		0.1	10	0	0	0	
My TransposeInterpolateRange_Single Phase Transformer		7		17-06-15 06:00:00.000 17-06-15 07:00:00.000		0.1	10	0	0	0	- 11
TransposeInterpolateRange_Single Phase Transformer		9		17-06-15 08:00:00.000		0.1	10	0	0	0	1 11
Garage Function Tables		10		17-06-15 09:00:00.000		0.1	10	0	0	0	1 1
		11		17-06-15 10:00:00.000		0.1	10	0	0	0	1
Dryers		12		17-06-15 11:00:00.000		0.1	10	0	0	0	
b) is statem		13		17-06-15 12:00:00.000		0.1	10	0	0	0	$\mathbf{\nabla}$
b UC2017			<		ш					>	
OLEDB Provider	τl		Query executed suc	cessfully			PI	SRV01 PI OLEDB E	interprise 00:00:30.489	100 m	ows
	Ľ.	-		,					001001001		

t. Edit the query by deleting the "TOP 100" specification in the SELECT statement. Either right-click on the query and select **Execute** or click on the **Execute** button on the top menu bar to rerun the query. This time you should get all 110,940 rows.

		PI SQL Cor	mmander Lite					-	
ie Edit V <mark>iew Query Tools Hel</mark> p									
🔔 New Que y 🚦 Execute 💷 👌 🎆 🖏 🚳 Query Compendium 📗									
	× 📦	Query3.sql - PISI	RV01 📄 Query1.sc	ql - PISRVO	1" X				
· · · · · · · · · · · · · · · · · · ·	SEL	LECT *							
1 OLEDB Data Sources	A FRO	M [Distribu	tion Network].[Ass	et].[My	Pole Transf	ormer Load	s]		
OLEDB Enterprise									
PISRV01									
⊿ 🛅 Catalogs									
AFExampleRUL									
Configuration									
Distribution Network									
A 🛄 Asset									
> 📴 Tables									
4 📴 Views									
My Pole Transformer Loads									
Pole Transformer Loads		a							
Pole Transformer Loads Table-Valued Functions	-	Results 🚺 Mer	Isages						
	•	Name	Timestamp	Loading	Maximum KVA	Rated KVA	Wh Delivered Load	Wh Delivered Load - 7d	
b Table-Valued Functions	1	Name PT_XYZ0342	Timestamp 2017-06-15 00:00:00 000	3.65	0.1	10	365	370	369
Compared by the second se	II	Name PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00:00:00 000 2017-06-15 01:00:00 000	3.65	0.1	10 10	365 365	370 370	369 370
Com Table-Valued Functions Com Tables	II II 1 2 3 4	Name PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00:00:00 000	3.65 3.65 3.64	0.1	10	365	370	369
) ⊡ Table Valued Functions) □ En Function Tables) □ Data □ Data □ Table T	II II 1 2 3 4 5	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00:00:00 000 2017-06-15 01:00:00 000 2017-06-15 02:00:00 000	3.65 3.65 3.64 3.64	0.1 0.1 0.1	10 10 10	365 365 364	370 370 371	369 370 370
) in Table Yakud Functions) in Fruction Tables) in Data 2 in Tables in Tables in Tables	II 1 2 3 4 5 6	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00:00:000 2017-06-15 01:00:00:00 2017-06-15 02:00:00 2017-06-15 02:00:00 2017-06-15 03:00:00 000 2017-06-15 04:00:00 000 2017-06-15 05:00:00 000	3.65 3.65 3.64 3.64 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10	365 365 364 364 0 0	370 370 371 346 0 0	369 370 370 371 7 0
) i i Tabi-Musi Fundions) ii fundion Tables) iii fundion Johan iii fabian iii fabian iii fabian iii fabian iii fabian fundions	II II 1 2 3 4 5 6 7	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00:00:00 000 2017-06-15 01:00:00 000 2017-06-15 02:00:00 000 2017-06-15 03:00:00 000 2017-06-15 04:00:00 000 2017-06-15 06:00:00 000	3.65 3.65 3.64 3.64 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10	365 365 364 364 0 0 0	370 370 371 346 0 0 0	369 370 370 371 7 0 0
) Table Valued Functions) Tables) Tables) Tables Tables Tables (Table Valued Functions) Tables (Table Valued Functions) Tables (Table Valued Functions))	II II 1 2 3 4 5 6 7 8	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timeetlamp 2017-06-15 00:00:00 000 2017-06-15 01:00:00 000 2017-06-15 01:00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000 2017-06-15 00:00 000	3.65 3.65 3.64 3.64 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10	365 365 364 364 0 0 0 0	370 370 371 346 0 0 0 0	369 370 370 371 7 0 0 0
)] Table Mark Functions) [and function Tables] [and function] Tables] [and functions] [able Value Functions] [able Value Functions] [able Value Functions] [able TransconstrumptionEflarge_Single Place Transformer] [able TransconstrumptionEflarge_Single Place Transformer]	1 2 3 4 5 6 7 8 9	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timeetlamp 2017-06-15 00:00:00 000 2017-06-15 01:00:00 000 2017-06-15 02:00 000 2017-06-15 02:00 000 2017-06-15 04:00:00 000 2017-06-15 06:00:00 000 2017-06-15 06:00:00 000 2017-06-15 08:00:00 000	3.65 3.65 3.64 3.64 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10	365 365 364 364 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0	369 370 370 371 7 0 0 0 0 0
))	1 2 3 4 5 6 7 8 9 10	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-06-15 00 00 00 000 2017-06-15 01 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000 2017-06-15 00 00 000	3.65 3.65 3.64 3.64 0 0 0 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10 10	365 365 364 0 0 0 0 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0 0 0 0	369 370 370 371 7 0 0 0 0 0 0 0
)] Table Mark Functions) [Fundon Tables] [Endon] Transform Tables] [Endon] [En	1 2 3 4 5 6 7 8 9 10 11	Name PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342 PT_XY20342	Timestamp 2017-66-15:00:00:00:00 2017-66-15:00:00:00:00 2017-66-15:02:00:00:00 2017-66-15:02:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00	3.65 3.65 3.64 3.64 0 0 0 0 0 0 0 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10 10 10 10	365 365 364 0 0 0 0 0 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0 0 0 0 0	369 370 370 371 7 0 0 0 0 0 0 0 0 0 0
))	1 2 3 4 5 6 7 8 9 10 11 12	Name PT_XY20342 PT_XY20342 PT_XY2042 PT_XY204	Timestamp 2017-66-15:00:00:00:00 2017-66-15:01:00:00:00 2017-66-15:02:00:00:00 2017-66-15:02:00:00:00 2017-66-15:02:00:00:00 2017-66-15:06:00:00:00 2017-66-15:06:00:00:00 2017-66-15:06:00:00:00 2017-66-15:06:00:00:00 2017-66-15:08:00:00:00 2017-66-15:08:00:00:00 2017-66-15:08:00:00:00 2017-66-15:10:00:00:00 2017-66-15:10:00:00:00	3.65 3.65 3.64 3.64 0 0 0 0 0 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10 10	365 365 364 0 0 0 0 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0 0 0 0	369 370 370 371 7 0 0 0 0 0 0 0
)] Table Mark Fractions) [Frinden Tables] Date] Tables] Tables] Tables] [Finden Tables] [Suffer Tables]	1 2 3 4 5 6 7 8 9 10 10 11 12 13	Name PT_XY20342 PT_XY2042 PT_XY2042 PT_XY2042 PT_XY2042 PT_XY2042	Timestamp 2017-66-15:00:00:00:00 2017-66-15:00:00:00:00 2017-66-15:02:00:00:00 2017-66-15:02:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00 2017-66-15:00:00:00	365 365 364 364 0 0 0 0 0 0 0 0 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10 10 10 10 10	365 365 364 0 0 0 0 0 0 0 0 0 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0 0 0 0 0 0 0 0 0	369 370 370 371 7 0 0 0 0 0 0 0 0 0 0 0
	1 2 3 4 5 6 7 8 9 10 10 11 12 13	Name PT_XY20342 PT_XY2042 PT_XY2042 PT_XY2042 PT_XY2042 PT_XY2042	Timestamp 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000 2017-46-15 10:00:00:000	3.65 3.65 3.64 3.64 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	10 10 10 10 10 10 10 10 10 10 10 10 10 1	365 365 364 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	370 370 371 346 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	370 370 371 7 0 0 0 0 0 0 0 0 0 0 0 0

Develop the Model in Python

Tip

The following code provides the same analysis as Example 2 except it accesses PI data using a PI OLEDB view. The Python "adodbapi" package is used to manage the connection.

To run the code with your PI OLEDB View, you'll want to replace "Pole Transformer Loads" with "My Pole Transformer Loads" in the first cell of this Jupyter Notebook.

- a. Open Jupyter Notebooks from the windows Taskbar, _____. Open the Python script, Load Prediction Example 2 (PI OLEDB Enterprise).
- b. In the first cell, change the select statement to 'SELECT * FROM [Distribution Network].[Asset].[My Pole Transformer Loads]'. This will cause Python to access your PI OLEDB Client view and create the dataframe, poleTransformerLoads. Select this cell and hit Shift+Enter. Wait until you see the table preview before moving to the next cell.

<pre>1 # EXAMPLE 2 - Create linear regression models for all 60 pole transformers. 2 # Read PI data via PI OLEDE Enterprise using the adodbapi package to establish an OLEDE connection. 3 4 # Import Fython packages. 5 import adodbapi as ado</pre>	'I data.
3 4 # Import Python packages. 5 import adodbapi as ado # OLEDB support for accessing PI SQL Client - import AF/H	PI data.
4 # Import Python packages. 5 import adodbapi as ado # OLEDB support for accessing PI SQL Client - import AF/H	₽I data.
5 import adodbapi as ado # OLEDB support for accessing PI SQL Client - import AF/B	?I data.
	?I data.
C demonstransformer and A Deterformer support	
6 import pandas as pd # Dataframe support.	
7 from sklearn.linear_model import LinearRegression # Linear regression model from the scikit-learn packa	ige.
8	
9 # Connect to "Distribution Network Lab" model in AF through the PI OLEDE Enterprise.	
10 PI_connection = ado.connect("Provider=PIOLEDBEnt; Integrated Security=SSPI; Data Source=PISRV01;Command	i Timeout=0;")
11	
12 # Create a cursor object to access the data serer for the "Distribution Network Lab" database in AF.	
13 PI_cursor = PI_connection.cursor()	
14	
15 # Select the entire table using the view configured in PI SQL Commander.	
16 [PI_cursor.execute('SELECT * FROM [Distribution Network].[Asset].[My Pole Transformer Loads]')	
17	
18 # Extract first row, index=0, to get column names for use as dataframe headers	
19 columnNames = [x[0] for x in PI_cursor.description]	
<pre>20 print("PI View column Names:\n\n", columnNames)</pre>	
21	
22 # Unpack the cursor rows into a datafraem.	
<pre>23 poleTransformerLoads = pd.DataFrame([dict(zip(columnNames, row)) for row in PI_cursor], columns=columnN</pre>	(ames)
24	
25 # Take a look to see if everythking worked.	
26 poleTransformerLoads.head()	

PI View column Names:

['Name', 'Timestamp', 'Loading', 'Maximum KVA', 'Rated KVA', 'Wh Delivered Load', 'Wh Delivered Load - 7d', 'Wh Deli vered Load - 14d', 'Ambient Temperature', 'Relative Humidity', 'Wind Speed']

Name Timestamp 0 PT_XYZ0342 2017-06-15 00:00:00	 Maximum KVA	Rated KVA	Wh Delivered Load	Wh Delivered Load - 7d	Wh Delivered Load - 14d	Ambient	Relative	Wind
					Luau - 14u	Temperature	Humidity	Speed
- 00.00.00	0.1	10.0	365.0	370.0	369.0	68.0	93.0	7.0
1 PT_XYZ0342 2017-06-15 01:00:00	0.1	10.0	365.0	370.0	370.0	68.0	93.0	6.0
2 PT_XYZ0342 2017-06-15 02:00:00	0.1	10.0	364.0	371.0	370.0	68.0	93.0	6.0
3 PT_XYZ0342 2017-06-15 03:00:00	0.1	10.0	364.0	346.0	371.0	68.0	90.0	0.0
4 PT_XYZ0342 2017-06-15 04:00:00	0.1	10.0	0.0	0.0	7.0	67.0	93.0	0.0

c. Rename columns with shorter names. Create a smaller dataframe, **modellingData**, containing only the values need for predicting transformer loads. Select this cell and hit **Shift+Enter**.

	<pre>1 # Rename some columns with shorter names to make them easier to vork with. 2 poleTransformerLoads.rename(columns = {'Name' :'Transformer'}, inplace = True) 3 poleTransformerLoads.rename(columns = {'Wh Delivered Load ':'Wh Load'}, inplace = True) 4 poleTransformerLoads.rename(columns = {'Wh Delivered Load - 14d':'Wh Load-14d'}, inplace = True) 5 poleTransformerLoads.rename(columns = {'Wh Delivered Load - 7d':'Wh Load-7d'}, inplace = True) 6 poleTransformerLoads.rename(columns = {'Ambient Temperature':'Temperature', inplace = True) 7 poleTransformerLoads.rename(columns = {'Relative Humidity':'Humidity', inplace = True) 8 poleTransformerLoads.rename(columns = {'Relative Humidity', inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened':'Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inplace = True) 9 poleTransformerLoads.rename(columns = {'Wind Sened': Wind'L inp</pre>													
	<pre>8 poleTran 9</pre>	poleTransformerLoads.rename(columns = {'Wind Speed':'Wind'}, inplace = True)												
1	-	second datafra	ume with ju	ist data	need	ed for a	our modell	ing.						
1		2 1	ansformerLo						ature', 'Humidity',					
1			ind , whi be	au, m	n boa	u-7u ,	wn boad-1	40]]						
		t the first fiv	ve rows to	make su	re th	ings loo	ok right.							
-	5 modellin	gData.head()												
	Transformer	Timestamp	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d						
0	PT_XYZ0342	2017-06-15 00:00:00	68.0	93.0	7.0	365.0	370.0	369.0						
1	PT_XYZ0342	2017-06-15 01:00:00	68.0	93.0	6.0	365.0	370.0	370.0						
2	PT_XYZ0342	2017-06-15 02:00:00	68.0	93.0	6.0	364.0	371.0	370.0						
3	PT_XYZ0342 2017-06-15 03:00:00 68.0 90.0 0.0 364.0 346.0 371.0													
						001.0	340.0	571.0						

d. Change the index of **modellingData** to be the transformer name. Select this cell and hit **Shift+Enter**.

2 modell 3 4 <i># Take</i>	<pre>1 # In order to analyze transformers individually, we need to set the datframe's index to the "Transformer" column. 2 modellingData = modellingData.set_index("Transformer", drop=False) 3 4 # Take a look, see the difference? 5 modellingData.head()</pre>												
Transformer	Transformer	Timestamp	Temperature	Humidity	Wind	Wh Load	Wh Load-7d	Wh Load-14d					
PT_XYZ0342	PT_XYZ0342	2017-06-15 00:00:00	68.0	93.0	7.0	365.0	370.0	369.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 01:00:00	68.0	93.0	6.0	365.0	370.0	370.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 02:00:00	68.0	93.0	6.0	364.0	371.0	370.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 03:00:00	68.0	90.0	0.0	364.0	346.0	371.0					
PT_XYZ0342	PT_XYZ0342	2017-06-15 04:00:00	67.0	93.0	0.0	0.0	0.0	7.0					

e. Leverage the connection to PI OLEDB Enterprise and **My Pole Transformer Loads** view to get the list of transformer names. Select this cell and hit **Shift+Enter**.



f. Generate linear regression model coefficients for each of the 60 transformers and write them to SQL. Select this cell and hit **Shift+Enter**.

```
1 # Using the "adodbapi" package, to connect to the "Predictive Equations" MS SQL.
   # Set connection parameters.
4 con_string = 'DRIVER={SQL_Server}; SERVER=PISRV01; DATABASE=PIWorld; Trusted_Connection=Yes; '
б
   # Connect to "Distribution Network Lab" model in AF through the PI SQL Client.
7 SQL_connection = ado.connect(con_string)
   # Create a cursor object to access the data serer for the "Distribution Network Lab" database in AF.
10 SQL_cursor = SQL_connection.cursor()
12
   # Create linear regression object from the "sklearn" package ve imported earlier.
13 LinReg = LinearRegression()
  # Looping through the trasformer list, Perform linear regression on each transformer.
16 for transformer in transformerNames:
       # Create dataframe for one transformer.
19
       transformerData = modellingData.loc[transformer,:]
       # Perform linear regression fit
      LinReg.fit(transformerData[["Wh Load-7d", "Wh Load-14d", "Temperature", "Humidity"]], transformerData["Wh Load"])
       # Update asset ID value with the name of this transformer.
asset_id = "'"+transformer+"'"
24
25
       # Print equation.
      28
29
             LinReg.coef_[3], "*", "'Relative Humidity' +(", LinReg.intercept_, ")" )
      ## INSERT and UPDATE queries to load table for the first time or update an existing one.
# Construct query to add this transfomer's model coefficients.
34
      insert_query = f'INSERT [Predictive Equations] ([Asset ID], Coefficient_0, Coefficient_1, Coefficient_2, Coeffic
       update guery = f'UPDATE [Predictive Equations] SET Coefficient 0={LinReg.coef [0]}, Coefficient 1={LinReg.coef [
        # Insert this record into the "Predictive Equations" table.
       SQL_cursor.execute(update_query)
40
   # Commit the queries to write the data into SOL.
41
42 SQL connection.commit()
44
   # Close database connections.
45 SQL_connection.close()
46 PI connection.close()
```

PT_XY20407 Eq: 0.22133353596401245 * 'Wh Delivered Load - 7d' + 0.18940221947528418 * 'Wh Delivered Load - 14d' + 303.2482562107521 * 'Am bient Temperature' + 7.845159067779704 * 'Relative Humidity' +(-14852.0518428231) PT_XY20408 Eq: 0.22918154817171893 * 'Wh Delivered Load - 7d' + 0.1723254292743076 * 'Wh Delivered Load - 14d' + 207.3195523807009 * 'Amb ient Temperature' + 5.783909412754433 * 'Relative Humidity' +(-8273.493697771371) PT_XY20409 Eq: 0.270655850527234 * 'Wh Delivered Load - 7d' + 0.24808404285129917 * 'Wh Delivered Load - 14d' + 257.19175040719324 * 'Am bient Temperature' + 37.67773251209764 * 'Relative Humidity' +(-15786.699575218732) PT_XY20410 Eq: 0.28804834752338854 * 'Wh Delivered Load - 7d' + 0.23893391851491846 * 'Wh Delivered Load - 14d' + 472.11670533964144 * 'A mbient Temperature' + 27.87650630785805 * 'Relative Humidity' +(-24316.385741388473) PT_XY20411 Eq: 0.435869138307828 * 'Wh Delivered Load - 7d' + 0.41339607398963046 * 'Wh Delivered Load - 14d' + 0.3590444851840645 * 'Amb ient Temperature' + 0.3196969465241239 * 'Relative Humidity' +(-34.91262860815391) PT_XY20412 Eq: 0.2075887778891295 * 'Wh Delivered Load - 7d' + 0.20641741419115514 * 'Wh Delivered Load - 14d' + 321.2758186735902 * 'Amb ient Temperature' + 36.830405127594084 * 'Relative Humidity' +(-17855.7284495021)

Appendix B - Python Access to PI via PI Web API

Develop the Model in Python

The PI Web API is installed with the PI System. It provides programmatic access to the PI Archive and PI AF. The following script imports the same dataset used in Example 2 using the PI Web API.

By default, there is a constraint placed on calls made to the PI System when using the PI Web API. Each call can return no more than 150,000 items. The type of items returned depends on the nature of the call. For example, your maximum items could consist of elements, attributes, or time series data values depending on how your call is defined.

This restriction can be adjusted by adding or resetting the "MaxReturnedItemsPerCall" attribute value under the PI Web API branch of the Configuration database on your AF Server. More information can be found on page 3 of the PI Web API 2018 Users Guide.

a. Import Python packages for supporting PI Web API session and managing message requests and unpacking json responses. Select this cell and hit **Shift+Enter**.

```
# EXAMPLE 2 - Create linear regression models for all 60 pole transformers.
   # Read PI data via PI Web API.
   # Import Python packages.
  import adodbapi as ado
                                                 # Support for accessing MS SQL to store model coefficients.
5
   import pandas as pd
6
                                                # Dataframe support.
  from sklearn.linear model import LinearRegression # Linear regression model from the scikit-learn package.
  # Import packages
11 import json, requests, urllib.parse, requests_kerberos # Support access to PI Web API.
13 # Setup Kerberos cnnection.
14 from requests_kerberos import HTTPKerberosAuth, OPTIONAL
15 clientAuth = HTTPKerberosAuth(force_preemptive=True, mutual_authentication=OFTIONAL, delegate=True)
16
  # Disable warning messages.
18
  requests.packages.urllib3.disable_warnings()
19
20 # Open session.
21 session = requests.session()
```

b. Make a client connection to PI via the PI Web API. Select this cell and hit Shift+Enter.

```
# Connect to PI System at highest level.
print('\nSending top level PI Web API Request\n')
     # Base URL to PI System.
 5 baseurl = "https://PISRV01/piwebapi/"
 6
 7 # Post URL to PI System and retrieve response.
 8 response = session.get(baseurl, auth=clientAuth, verify=False, timeout=30)
10 # Response status - did it work, i.e. 200 means success?
11 print('Response Received: {0} ' .format(response.status_code),'\n')
    # Unpack items of interest using the "json" package. Extracts dictionary of links.
14 Links = json.loads(response.text)['Links']
16 # Extracts base level URL's for PI Web API from the Links dictionary using the "json" package.
17 print('Self:', Links['Self'])
18 print('Asset Servers:', Links['AssetServers'])
19 print('Data Servers:', Links['DataServers'])
20 print('PI System:', Links['System'], '\n')
Sending top level PI Web API Request
Response Received: 200
Self: https://pisrv01/piwebapi/
Asset Servers: https://pisrv01/piwebapi/assetservers
Data Servers: https://pisrv01/piwebapi/dataservers
PI System: https://pisrv01/piwebapi/system
```

c. PI Web API Request for "WebId" of Colegio Científico circuit (parent) element.

```
1 # Get WebId for circuit element, "Colegio Clientifico".
 3 # Root path to "circuit" level in AF heirarchy.
 4 circuitPath = '\\\pisrv01\\Distribution Network\\Alajuela\\Avenida Central\\Transformer 1\\Colegio Científico'
 6 # Construct URL to the Get Elements controller of the PI Web API.
 7 baseurl elements = baseurl + '/elements'
 8
 9 # Set parameters list by specifying the AF path to the circuit's element.
10 parameters = { 'path': circuitPath, }
11
12 # Format parameter list for URL.
13 url_parameters = urllib.parse.urlencode(parameters, quote_via=urllib.parse.quote)
14
15 # Show HTTPS request string.
16 print('PI Web API Request for WebId of Colegio Cientifico. Click it to see the response from PI.\n\n', \
17 baseurl_elements + '?' + url_parameters, '\n')
18
19 # Post URL with path and paramters to PI System
20 response = session.get(baseurl_elements, auth=clientAuth, params=url_parameters, verify=False, timeout=30)
22 # Response status - did it work?
23 print('Response Received: {0} ' .format(response.status_code),'\n')
24
25 # Show element's Web Id.
26 circuitWebId = json.loads(response.text)['WebId']
27 print("Colegio Clientifico element's WebId:\n", circuitWebId)
PI Web API Request for WebId of Colegio Científico. Click it to see the response from PI.
https://PISRV01/piwebapi//elements?path=%5C%5Cpisrv01%5CDistribution%20Network%5CAlajuela%5CAvenida%20Central%5CTran
sformer%201%5CColegio%20Cient%C3%ADfico
Response Received: 200
Colegio Clientifico element's WebId:
Flem BJ6yx0WfEulXN8dxGowlg3a Sqp0s6BGBFQAN0jYKWwUElTUlYwMVxESVNUUklCVVRJT04gTkVUV09SS1xBTEFKVUVMQVxBVkV0SURBIENFTLRS
QUxcVFJBT1NGT1JNRVIgMVxDT0xFR01PIENJRU5Uw41GSUNP
```

For this example, we launched Jupyter Notebooks in Google Chrome. In Chrome, hyperlinks printed from Python are active and when clicked will open an adjacent browser window allowing you to see messages returned from the PI Server. This is a very handy feature for debugging your script.

d. Clicking the hyperlink in the Python output shows the response message from the PI Web API.



e. PI Web API Request for "WebId"'s of X, Y and Z Phase (child) elements.

Tip



f. Define an empty dataframe named **modellingData** and list name **allTransformers** to write the transformer data and names into. Select this cell and hit **Shift+Enter**.

```
1 # Create dataframe of interpolated data for each transformer.
2
3 # Create empty pandas dataframe for transformer data.
4 columns =['Transformer', 'TimeStamp', 'Temperature', 'Humidity','Wind','Wh Load', 'Wh Load-14d', 'Wh Load-7d']
5 modellingData = pd.DataFrame(columns=columns)
6
7 # As we loop through heirarchy to get data, establish a list of all transformers.
8 allTransformers = []
```

g. We will step through each Phase, X, Y and Z. For each phase, we will loop through each transformer, appending its attribute values to the dataset. Select this cell and hit Shift+Enter.

```
# Fill dataframe by stepping through the phase level and appending each transformer's attribute values.
   # Print URL as example on the first time through the loop.
 4 printURL = True
   # Step through phases.
6
   for phaseName, phaseWebId in phaseElements.items():
        # Constructed URL to reference each phase element.
       baseurl_referencedelements = baseurl + '/elements' + '/' + phaseWebId + '/referencedelements'
       # Request for all child elements derived from the "Single Phase Transfrmer" template.
      parameters = {'templateName': 'Single Phase Transformer','selectedFields':'Items.Name;Items.WebId'}
14
       # Format parameter list for URL.
16
       url_parameters = urllib.parse.urlencode(parameters, quote_via=urllib.parse.quote)
18
       # Show HTTPS request string
19
       if printURL:
         print("Sample PI Web API request for Name and WebId's of transformers under",\
20
21
               phaseName, ' Click it to see the response from PI.\n\n', \
               baseurl_referencedelements + '?' + url_parameters, '\n')
22
24
       # Post URL with path and paramters to PI System
25
      response = session.get(baseurl_referencedelements, auth=clientAuth, params=url_parameters, verify=False, timeou
26
27
       # Response status - did it work?
28
       print('Response Received: {0}' .format(response.status_code), "Transformer Name and WebID's for", phaseName, '\n
29
30
       # Define\empty dictionary to hold transformer names and WebId's.
31
       transformerElements = {}
32
33
34
35
       # Loop through list and put names and WebId's into dictinary.
       for item in json.loads(response.text)['Items']:
36
            # Store the strings.
           transformerElements.update({item['Name']:item['WebId']})
```

```
# Loop through phases.
40
        for transformerName, transformerWebId in transformerElements.items():
41
42
           # Add this transformer to the list.
           allTransformers.append(transformerName)
43
44
45
           # Construct URL for getting selected attributes.
46
47
           baseurl_multipleattributes = baseurl + '/attributes/multiple'
48
           # Request WebId's for specified transformer attributes.
           49
50
51
52
                         ('Path', circuitPath + '\\' + phaseName + '\\' + transformerName + '|Wh Delivered Load - 14d'
                         ('Path', circuitPath + '\\' + phaseName + '\\' + transformerName + '|Wh Delivered Load - 7d')
54
55
56
                         ('Path', circuitPath + '\\' + phaseName + '\\' + transformerName + '|Wind Speed'), \
                         ('SelectedFields', 'Items.Object.Name; Items.Object.WebId')]
           # Format parameter list for URL.
58
           url parameters = urllib.parse.urlencode (parameters, quote via=urllib.parse.quote)
59
60
61
           # Show HTTPS request string
           if printURL:
62
               print("Sample PI Web API request for Name an WebId's of", transformerName, "'s selected attributes.", \
63
                    Click it to see the response from PI.\n\n', baseurl_multipleattributes + '?' + url_parameters,
64
65
           # Post URL with path and paramters to PI System.
66
           response = session.get(baseurl_multipleattributes, auth=clientAuth, params=url_parameters, verify=False, t:
67
68
           # Response status - did it work?
                 (' Response Received: {0}' .format(response.status_code),"Attribute Name and WebId's for", \ transformerName, '\n')
69
70
           print('
```

```
# Define list for attribute Webid's.
              attributeWebId = []
74
75
              # Loop through unnamed list and extract WebId and Name into lists.
              for attributes in json.loads(response.text)['Items']:
78
                  attribute = attributes['Object']
79
                  attributeWebId.append(attribute['WebId'])
80
81
             # Construct URL for getting interpoated data for attributes.
baseurl streamsets = baseurl + '/streamsets/interpolated'
82
83
84
              # Request for start and end times, and interval.
85
              parameters = [('WebId', attributeWebId[0]),('WebId', attributeWebId[1]),\
                              ('WebId', attributeWebId[2]),('WebId', attributeWebId[3]),\
('WebId', attributeWebId[2]),('WebId', attributeWebId[3]),\
('startTime','15-jun-2017'),('endTime','31-aug-2017'),('interval','lh'),\
87
88
89
                              ('selectedFields', 'Items.Name; Items.Items.Timestamp; Items.Items.Value')]
90
91
              # Format parameter list for URL.
             url parameters = urllib.parse.urlencode(parameters, quote via=urllib.parse.quote)
92
93
94
              # Show HTTPS request string
95
             if printURL:
96
                 print("Sample PI Web API request for Timestamp and Values for transformer's selected attributes.", \
97
                       'Click it to see the response from PI.\n\n', baseurl streamsets + '?' + url parameters, '\n')
98
99
              # Post URL with path and paramters to PI System
100
              response = session.get(baseurl_streamsets, auth=clientAuth, params=url_parameters, verify=False, timeout=30
              # Response status - did it work?
             print('
                           Response Received: {0}' .format(response.status_code), "Attribute Timestamp and Values for", \
                     transformerName, ' n')
```

106	# Initialize lists to store unpacked data.
107	<pre>transformer = []; timestamp = []; temperature = []; humidity= []; wind = []</pre>
108	load = []; load 1dd = []; load 7d = []
109	
110	# Loop through response and create pandas dataframe for this transformer.
111	# not and response and reade panals decarrance for only craniformer. # For each attribute.
112	for attribute in json.loads(response.text)['Items']:
113	for dooring of the joon reader (response reck) (reader j.
114	<pre># For each teimstamp\value.</pre>
115	for value in attribute('Items'):
116	
117	# Store values for "Ambient Temperature" attribute into its list.
118	if attribute['Name'] == 'Ambient Temperature':
119	temperature.append(value['Value'])
120	
121	# Fill timestamp and transformer name lists here, only need to do this once.
122	<pre>timestamp.append(value['Timestamp'])</pre>
123	transformer.append(transformerName)
124	
125	# Store values for "Relative Humidity" attribute into its list.
126	if attribute['Name'] == 'Relative Humidity':
127	<pre>humidity.append(value['Value'])</pre>
128	
129	# Store values for "Wind Speed" attribute into its list.
130	if attribute['Name'] == 'Wind Speed':
131	wind.append(value['Value'])
132	
133	# Store values for "Wh Delivered Load" attribute into its list.
134	<pre>if attribute['Name'] == 'Wh Delivered Load':</pre>
135	<pre>load.append(value['Value'])</pre>
136	
137	# Store values for "Wh Delivered Load - 14d" attribute into its list.
138	if attribute['Name'] == 'Wh Delivered Load - 14d':
139	<pre>load_14d.append(value['Value'])</pre>
140	
141	# Store values for "Wh Delivered Load - 7d" attribute into its list.
142	if attribute['Name'] == 'Wh Delivered Load - 7d':
143	<pre>load_7d.append(value['Value'])</pre>
144	
145	# Put lists of data for this transformer into a temporary dataframe
146	thisTransformerData = pd.DataFrame({ 'Transformer': transformer, 'TimeStamp': timestamp, 'Temperature' :temperat
147	'Humidity':humidity, 'Wind':wind, 'Wh Load':load, 'Wh Load-14d':load_14d
148	'Wh Load-7d':load_7d})

150 # Add this transformer's data to the others. 151 modellingData = pd.concat([modellingData, thisTransformerData],ignore_index=True) 152 # Turn off URL printing. 153 # Turn off URL printing. 154 printURL = False 155 156 # Print end of dataframe to see if we got all 110,940 records(row index of 110,939). 157 modellingData.tail(5)

h. Scroll down the results window to see the phase steps and transformer loops used to build the dtataframe. At the very bottom inspect the last five records to confirm all 110, 939 have been extracted.

1								
	Humidity	Temperature	TimeStamp	Transformer	Wh Load	Wh Load-14d	Wh Load-7d	Wind
110935	73.0	68.0	2017-08-31T03:00:00Z	PT_XYZ0414	69.0	44.0	68.0	7.0
110936	81.0	64.0	2017-08-31T04:00:00Z	PT_XYZ0414	68.0	45.0	68.0	3.0
110937	75.0	66.0	2017-08-31T05:00:00Z	PT_XYZ0414	68.0	45.0	69.0	3.0
110938	78.0	65.0	2017-08-31T06:00:00Z	PT_XYZ0414	67.0	46.0	68.0	3.0
110939	86.0	62.0	2017-08-31T07:00:00Z	PT_XYZ0414	68.0	44.0	69.0	3.0

i. Generate linear regression model coefficients for each transformer and write them to SQL. Select this cell and hit **Shift+Enter**.

```
3 # In order to analyze transformers individually, we need to set the datframe's index to the "Transformer" column.
   modellingData = modellingData.set index("Transformer", drop=False)
    # Set connection parameters.
 6
   con_string = 'DRIVER={SQL Server}; SERVER=PISRV01; DATABASE=PIWORLD; Trusted_Connection=Yes; '
8
9 # Connect to "Distribution Network Lab" model in AF through the PI SOL Client.
10 SQL_connection = ado.connect(con_string)
12 # Create a cursor object to access the data serer for the "Distribution Network Lab" database in AF.
13 SQL_cursor = SQL_connection.cursor()
14
15 # Create linear regression object from the "sklearn" package we imported earlier.
16 LinReg = LinearRegression()
18 # Looping through the trasformer list, Perform linear regression on each transformer.
19 for transformer in allTransformers:
         # Create dataframe for one transformer.
        transformerData = modellingData.loc[transformer,:]
23
24
       # Perform linear regression fit
25
       LinReg.fit(transformerData[["Wh Load-7d", "Wh Load-14d", "Temperature", "Humidity"]], transformerData["Wh Load"])
26
27
        # Update asset ID value with the name of this transformer.
       asset_id = "'"+transformer+"'"
28
29
        # Print equation.
        # Fint equation:
print(transformer, "Eq:\n", LinReg.coef_[0], "*","'Wh Delivered Load - 7d' + ",LinReg.coef_[1], "*","'Wh Deliver
LinReg.coef_[2], "*", "'Ambient Temperature' +", LinReg.coef_[3], "*","'Relative Humidity' +(",LinReg.intercept_
31
33
34
35
        ## INSERT and UPDATE queries to load table for the first time or update an existing one.
       # Construct query to add this transfomer's model coefficients.
insert_query = f'INSERT [Predictive Equations] ([Asset ID], Coefficient_0, Coefficient_1, Coefficient_2, Coeffic
36
37
38
        update_query = f'UPDATE [Predictive Equations] SET Coefficient_0={LinReg.coef_[0]}, Coefficient_1={LinReg.coef_[
39
40
        # Insert this record into the "Predictive Equations" table.
41
        SQL cursor.execute (update query)
42
43 # Commit the queries to write the data into SQL.
44 SQL_connection.commit()
45
46 # Close database connections.
47 SQL_connection.close()
```

Appendix C – Learning Resources

PI Resources

PI Integrator for Business Analytics (YouTube Playlist)

Exposing PI Data with the PI SQL Framework (Online Course)

Programming in PI Web API (Online Course)

Jupyter Resources

Jupyter Notebook for Beginners: A Tutorial

Jupyter Notebook Tutorial: The Definitive Guide

Jupyter Notebook Cheat Sheet

Jupyter Notebook Keyboard Shortcuts

28 Jupyter Notebook tips, tricks, and shortcuts

Machine Learning Resources

The table below contains a collection of learning resources I have experience with **Course | Topic:** Abbreviated title and hyperlink of learning resource **Type:**

Course or Book

Language: Programming language used in the course

Platform: Course provider

Complexity: (1 introductory) $\xrightarrow{}$ (5 bleeding-edge) **Curve:** Learning-curve or approachability of the material (1 requires strong background

knowledge) \rightarrow (5 easy to follow given the complexity)

Theoretical: Theoretical coverage of algorithm details (1 few details) \rightarrow (5 in-depth explanations)

Practical: (1 difficult to apply to project) \rightarrow (5 readily applicable to real-world scenarios)

Course Topic	Туре	Language	Platform	Complexity	Curve	Theoretical	Practical
<u>Data Science</u> <u>Bootcamp</u>	Course	Python	Udemy	2	5	2	4
The Analytics Edge	Course	R	edX	3	5	4	4
Statistical Learning in R	Book	R	Book	3	5	5	5
Machine Learning	Course	Matlab	Coursera	3	4	5	4
TensorFlow Deep Learning	Course	Python	Udemy	3	4	2	3
Neural Networks for Machine Learning	Course	Python	Coursera	3	3	5	3

Hands-On Machine Learning	Book	Python	Book	4	5	4	5	
Deep Learning Specialization	Course	Python	Coursera	4	4	5	4	
Machine Learning for Coders	Course	Python	Fast.Al	4	4	4	4	
Practical Deep Learning For Coders	Course	Python	Fast.Al	4	4	3	5	
Deep Learning with Python	Book	Python	Book	4	4	4	5	
Deep Learning	Book	Python	Book	4	2	5	1	
Cutting Edge Deep Learning For Coders	Course	Python	Fast.Al	5	4	3	5	
Time Series Analysis	Course	R	edX	Unknown				
Big Data Analytics Using Spark	Course	Python	edX	Unknown				

Join <u>Kaggle!</u> A vibrant machine learning community with over 13,000 datasets to practice on and learn from others. Kaggle has many community-made tutorials for beginners, competitions with rewards (such as the \$1.5 million Homeland Security Passenger Screening Challenge), and many examples of bleeding-edge machine learning algorithms.





Save the Date!

OSIsoft PI World Users Conference in Gothenburg, Sweden. September 16-19, 2019.

Register your interest now to receive updates and notification early bird registration opening.

<u>https://pages.osisoft.com/UC-EMEA-Q3-19-PIWorldGBG-</u> <u>RegisterYourInterest_RegisterYourInterest-LP.html?_ga=2.20661553.86037572.1539782043-</u> <u>591736536.1533567354</u>

