# **SPSS using NPS – Deeper Dive**

## Lab overview

In this lab you will use SPSS Modeler (premium cartridge) to build a churn model using training data stored in Netezza Performance Server (NPS). You will also use a Jupyter notebook to build a churn model using training data stored in NPS.

These two methods for building models are distinctly different: SPSS uses a clickable paradigm, where Jupyter notebooks use coding. After building the models you will deploy them in a Cloud Pak for Data deployment space.

You will then create two types of deployments: batch and online. Finally, you will execute the deployments to complete an end-to-end example of training, deploying and using predictive models via the Cloud Pak for Data platform with NPS data.

## Persona represented in this lab

The Data Scientist persona is the likely role to perform all the tasks shown in this lab.

|  |  |  |  |
| --- | --- | --- | --- |
| **Persona (Role)** | **Capabilities** | | |
| Data Scientist | | Data Scientists bring expertise in statistics and the process of building ML/AI models to make predictions and answer key business questions. |

## Logging into the CPD web client (if you have not already done so)

1. If you are starting this lab stand-alone (without going through previous labs) do the following:
2. Click the desktop icon: Cloud Pak for Data Web Client.

A picture containing object, clock

Description automatically generated

1. The CPD web client GUI displays as shown. Use cpduser and cpduser for the *Username* and *Password* and click Sign In.

A screenshot of a cell phone

Description automatically generated

## Working with NPS Data in SPSS

1. Click Navigation Menu 🢧 Projects 🢧 NPS Analytics Project.

A screenshot of a cell phone

Description automatically generated

1. Click Assets.

A screenshot of a cell phone

Description automatically generated

1. Scroll down to the Modeler flows Section and click Churn Prediction Flow.

A close up of a logo

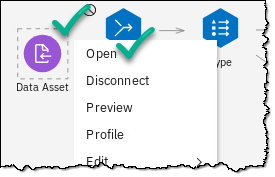
Description automatically generated

|  |  |
| --- | --- |
| Data Scientist | Note: This SPSS Modeler flow performs data preparation, model building, and model evaluation, all without requiring coding, by using a graphical user interface. |

### **Importing data**

1. You’ll need to connect to three tables stored in NPS. To connect to the first table, do the following:

Mouse over the Data Asset Import Node 🢧 Right click (or click the ellipsis) 🢧 Open.



1. On the right side of the screen a pop-up will appear.

Select Change data asset.

A screenshot of a cell phone

Description automatically generated

1. Click Connections 🢧 NPS\_Database 🢧 ADMIN 🢧 CUSTOMER\_ACTIVTY 🢧 OK.

A screenshot of a cell phone

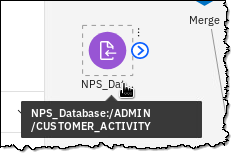
Description automatically generated

1. Click Save in the lower right to finish setting up the Data Asset Import node to the first table in NPS.

A screenshot of a cell phone

Description automatically generated

1. Hover over the node on the canvas and notice it has changed its name to NPS\_Database:/ADMIN/CUSTOMER\_ACTIVITY.



1. Mouse over the Data Asset Import node you are working with until an arrow in a circle appears.

Click and hold the arrow while dragging to the Merge node located just to the right. Release the mouse button to connect the two nodes.

A screenshot of a cell phone

Description automatically generated

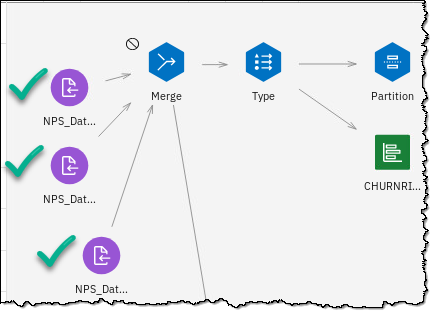
1. Now create and connect to the remaining two tables by clicking the dropdown menu Import, then click and hold Data Asset and drag and drop it onto the canvas to the right. Follow the Importing Data steps above to create connection for CUSTOMER\_CHURN and CUSTOMER\_\_DEMOGRAPHICS.

A screenshot of a cell phone

Description automatically generated

When done, you will have three Data Asset Import nodes connected to these tables in NPS CUSTOMER\_ACTIVTY, CUSTOMER\_CHURN, CUSTOMER\_DEMOGRAPHICS.

Tip: Hover over each node to make sure you have three different tables represented by the three nodes you just created and subsequently merged.

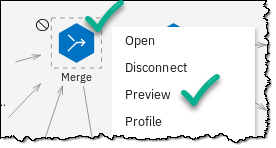
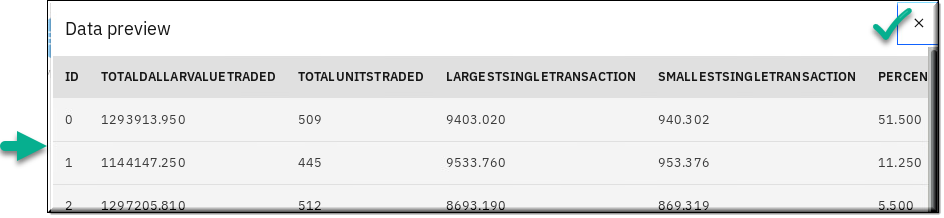


### **Testing the merged data**

1. To test your work, do a preview of the merged data.

On the Merge node click the ellipsis 🢧 Preview.

At this point you have merged all the training data needed to train and test the churn model.

Click the X in the upper right of the Data preview to close the window.

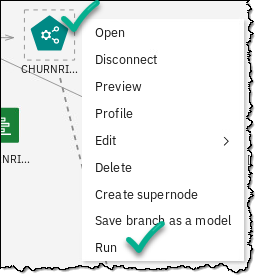
## Building, evaluating, and saving the SPSS model

In this section you will retrain the SPSS model using the training data from NPS. Then, you will evaluate the goodness of the model. Finally, you will deploy the model and save a batch of data for testing the deployed model.

### **Rebuild the model**

1. Rebuild the model by first finding the green CHURNRISK model building node.

Click green CHURNRISK (node) 🢧 ellipsis 🢧 Run.

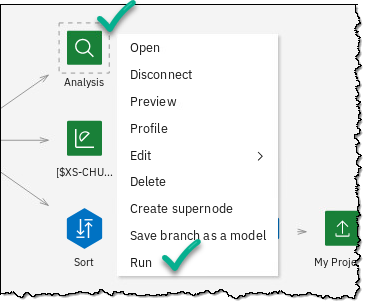


1. You will see the flow running...

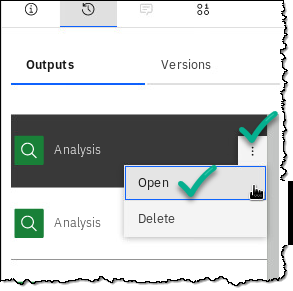


1. Evaluate the goodness of the model by hovering over the ellipsis on the Analysis node.

Click Analysis (node) 🢧 Ellipsis 🢧 Run.

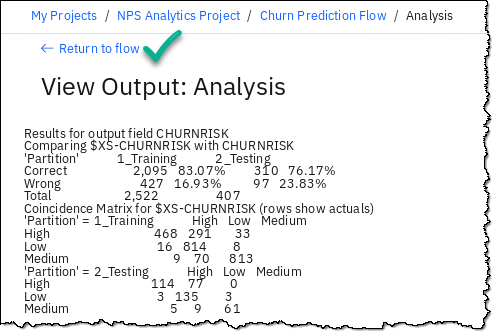


1. In the pop up box Outputs section, click Analysis 🢧 Open.



1. Review the information in View Output: Analysis.

When finished, click Return to Flow.

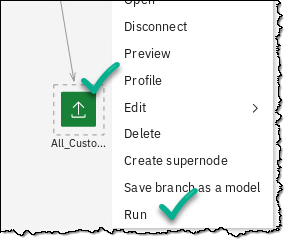


### **Prepare test data**

The portion of an SPSS flow that is to be deployed to Cloud Pak for Data will receive its input data from a CSV file. You will prepare for this by creating a small sample file in CSV format.

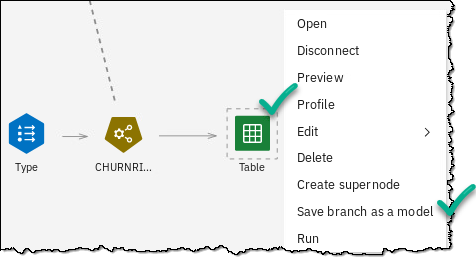
You will save this file for use by the portion of the flow to be deployed as a model.

1. Find the All\_Customer\_Data node 🢧 Click the ellipsis 🢧 Run.



1. The node to the immediate right of the Data Asset Export node that you just ran reads the CSV file you created and provides it as input to a copy of the Churn model you built earlier.

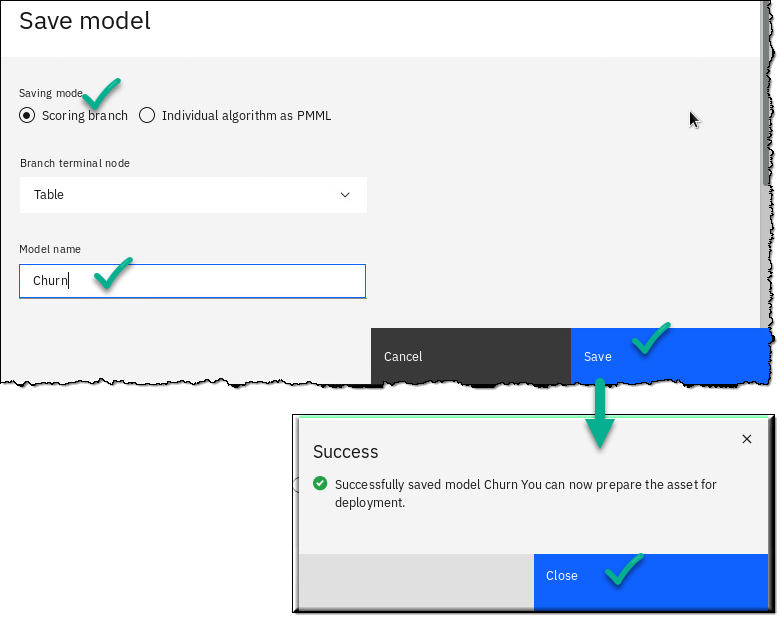
Hover over the Table node 🢧 Click on the ellipsis 🢧 Save branch as a model



1. Leave the Saving mode as Scoring branch.

Name the model Churn.

Save 🢧 Close.



## Creating and testing an online model deployment

Now you will create and test an online deployment using the model you just saved.

### **Create the model deployment**

1. Click on NPS Analytics Project.

A screenshot of a cell phone

Description automatically generated

1. Scroll down to the Models Section and click on the ellipsis for the Churn model 🢧 Promote 🢧 Promote to space.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

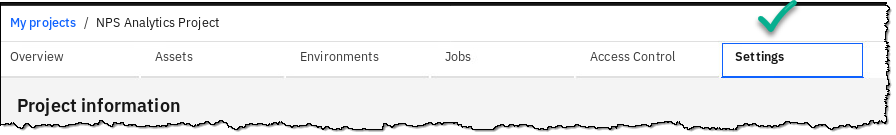
Description automatically generated

You will see a message near the top of your screen indicating success.

A screenshot of a cell phone

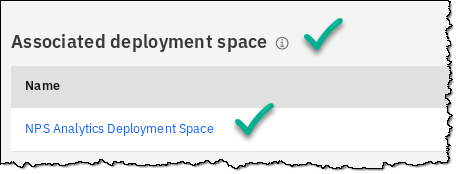
Description automatically generated

1. Scroll to the top of the screen and click Settings.



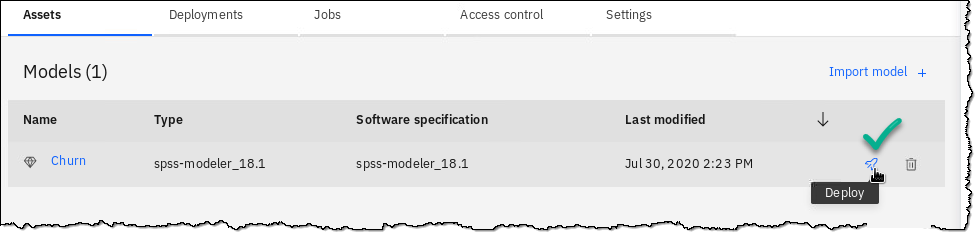
1. Scroll down to find section Associated deployment space.

Click NPS Analytics Deployment Space.



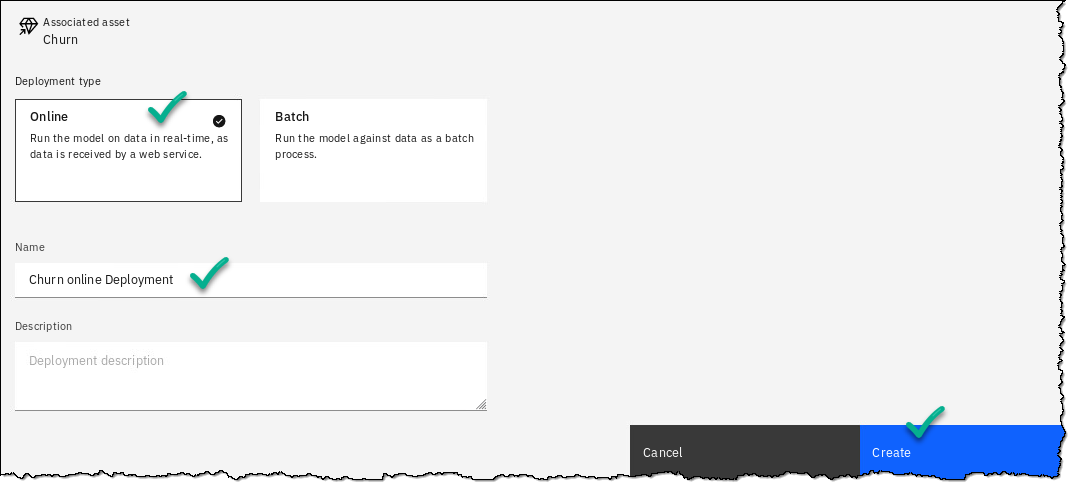
1. Locate the model Churn and hover at the end of the line.

Click Deploy.



1. Create an online deployment.

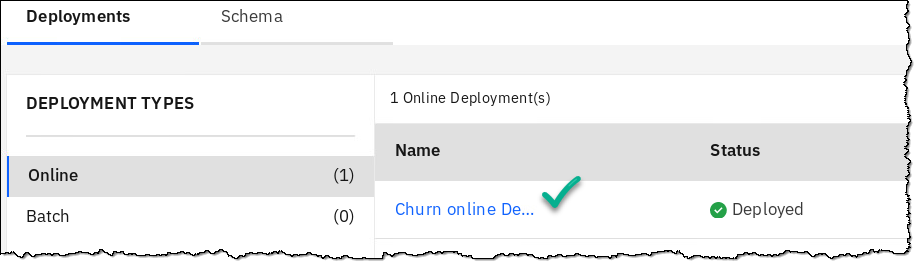
Select Online 🢧 Name it Churn online Deployment 🢧 Create.



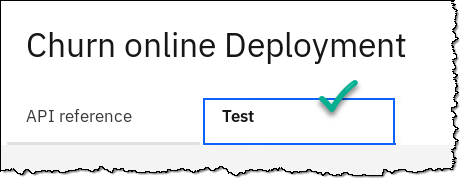
1. Wait for the Status to change to Deployed.

### **Test the online model deployment**

1. Click Churn online Deployment.

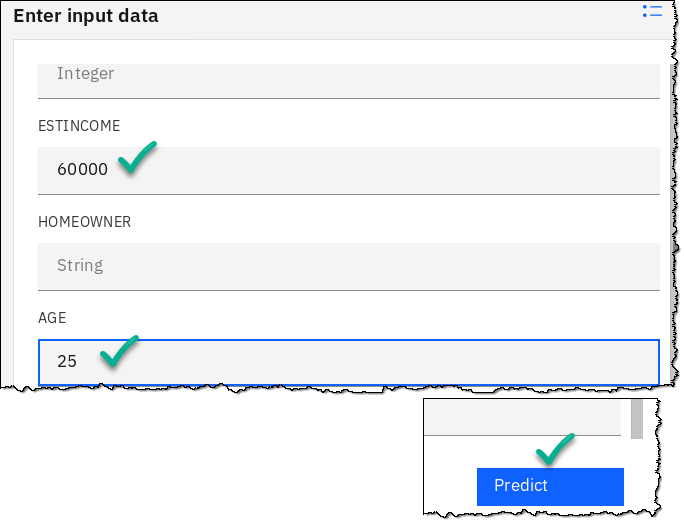


1. Select tab Test.

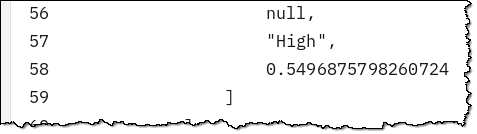


1. Enter input data by first scrolling down until you can see the ESTINCOME and AGE fields, enter 60000 for ESTINCOME and 25 for AGE.

Click Predict 🢧



1. Scroll down in the Results window and you’ll see the churn prediction along with the confidence in the prediction.



## Working with NPS Data in Jupyter Notebooks

In this section you will use a Jupyter notebook to train and deploy a predictive model. This is a coding approach and this notebook is written in Python.

The notebook is complete except for the connections to the three tables containing training data stored in NPS. You need to add the code to accomplish this. However, you’ll find that the code will be automatically generated for you.

### **Launch the notebook**

1. Click Navigation Menu 🢧 Projects 🢧 NPS Analytics Project.

A screenshot of a cell phone

Description automatically generated

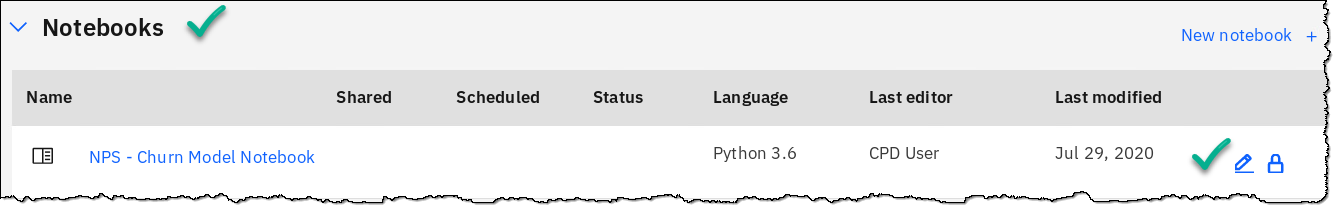
1. Click Assets.

A screenshot of a cell phone

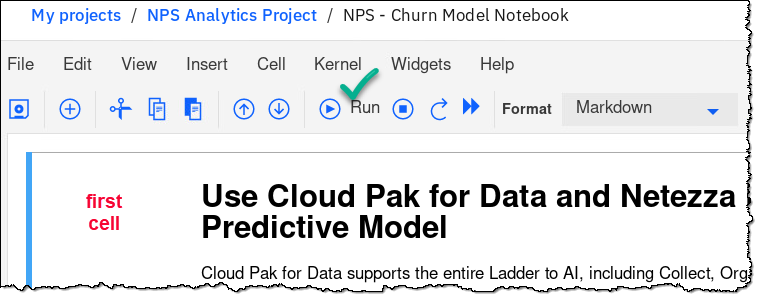
Description automatically generated

1. Scroll down to the Notebooks Section.

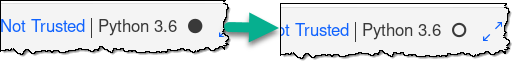
Click the Pencil icon next to the Notebook named NPS - Churn Model Notebook.



1. You will start by selecting the first cell by clicking on it 🢧 Read the contents of the cell then click Run.



You can tell when the notebook is running a cell by observing a solid black circle in the upper right-hand corner of the screen. It turns to an open circle when the notebook has completed its run. After any individual cell has run, any output from the code is displayed immediately after that cell, and the next cell is automatically selected for you.



|  |  |
| --- | --- |
| A picture containing clothing, hat, white, light  Description automatically generated  Data Scientist | If you receive a 403: Forbidden error restart the notebook by finding the information icon at the top right of the screen and clicking as shown below. |

### **Load data into the notebook**

1. Continue reading and running each cell until you reach the three cells containing the text.

# Load data from NPS Performance Server



1. Position and Click your cursor on the last blank line inside the first of the three data cells under the notebook Section named “Load and explore the data.” Then, Click Data (the 0100 Icon) 🢧 Connections 🢧 Insert to code 🢧 Insert pandas DataFrame.

A screenshot of a cell phone

Description automatically generated

1. In the pop-up to Select schema and table, Click ADMIN 🢧 CUSTOMER\_CHURN 🢧 Select. This will insert the code needed to load the data table from NPS into the notebook.

A screenshot of a cell phone

Description automatically generated

1. Near the end of the inserted code there are two lines of code that start with data\_df\_#, where the # is a number. If the number is not equal to 1, then change it to 1 in both lines of code.

data\_df\_4 🢧 data\_df\_1

A screenshot of a cell phone

Description automatically generated

1. Click Run to load the table into the notebook and display the first few rows.

A screenshot of a cell phone

Description automatically generated

1. Now you should be on the cell that loads the CUSTOMER\_DEMOGRAPHICS table from NPS. To do so, repeat steps from the top of this sub-section Loading data into the Notebook.

Be sure to select the CUSTOMER\_DEMOGRAPICS table and use data\_df\_2 as the DataFrame variable name. Note that the code to load this table is shorter than the code to load the first table. This is because the code for the first table included loading the database credentials, which is only required once.

A screenshot of a cell phone

Description automatically generated

|  |  |
| --- | --- |
| A picture containing clothing, hat, white, light  Description automatically generated  Data Scientist | Note: The code to load this table is shorter than the code to load the first table. This is because the code for the first table included loading the database credentials, which is only required once. |

1. Now you should be on the cell that loads the third and final table, CUSTOMER\_ACTIVITY, from NPS. To do so repeat the steps again from sub-section Loading data into the notebook.

The only changes are to make sure to select the CUSTOMER\_ACTIVTY table and use data\_df\_3 as the DataFrame variable name.

A screenshot of a cell phone

Description automatically generated

1. Continue running the notebook one cell at a time until you reach the end. Near the end of the notebook a new deployment space will be created and the model that you’ve trained will be deployed as an online deployment. You’ll work with that deployed model in the next Section.

A screenshot of a cell phone

Description automatically generated

## Working with Batch Deployments

In this section we’ll work with the model you saved from the Jupyter notebook.

1. Click Navigation Menu 🢧 Analyze 🢧 Analytics deployments.

A screenshot of a cell phone

Description automatically generated

1. Click Deployment-Space-Created-From-Notebook 🢧 Churn-Model 🢧 Create deployment 🢧 Batch 🢧 name it Churn Batch Deployment 🢧 select One Standard CPU 🢧 Click CreateA screenshot of a cell phone

   Description automatically generated

A screenshot of a cell phone

Description automatically generated

1. Before creating the batch job, we need to get some input data.

Click Navigation Menu 🢧 Click Projects 🢧 Click NPS Analytics Project 🢧 Under Data assets click Payload to test deployment.json 🢧 Carefully copy the entire contents of the json file (making sure to highlight everything).

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

1. Click Navigation Menu 🢧 Analyze 🢧 Analytics deployments 🢧 Deployment-Space-Created-From-Notebook 🢧 Deployments 🢧 Churn Batch Deployment.A screenshot of a cell phone

   Description automatically generated

A screenshot of a cell phone

Description automatically generated

1. Click Create Job 🢧 name the job Churn Batch Job 🢧 Specify input data 🢧 Click Inline input data 🢧 Paste the JSON data you copied in the previous step 🢧 Confirm 🢧 Create and Run.

A close up of a logo

Description automatically generatedA screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

1. Let the job complete (takes about 20 seconds).

Click the start time for the job run.

Click Show More 🢧 scroll to the bottom of the log to view the prediction and probability.

A close up of a logo

Description automatically generated A close up of a logo

Description automatically generated

A screenshot of text

Description automatically generated

## Lab Conclusion

In this lab, you used tables stored in NPS as a source of training data for building churn models.

Using SPSS Modeler, part of the SPSS Modeler Premium Cartridge in Cloud Pak for Data, you built a churn model using the clickable paradigm for modeling. You deployed the model for online use and tested the deployment.

Using a Jupyter notebook written in Python you built another churn model using the coding paradigm for modeling. You deployed this model programmatically from the notebook and manually from the deployment space. You then tested the batch deployment using a JSON payload.

**\*\* End of Lab 15 – SPSS using NPS – Deeper Dive**

Lab by Tom Konchan, Edited by Burt Vialpando, John Lucas and Kent Rubin