Data Science with Intellicus

Version: 18.1

intellicus

Dated: November 2018

Acknowledgements

Intellicus acknowledges using of third-party libraries to extend support to the functionalities that they provide.

For details, visit: http://www.intellicus.com/acknowledgements.htm

Contents

1 Introduction

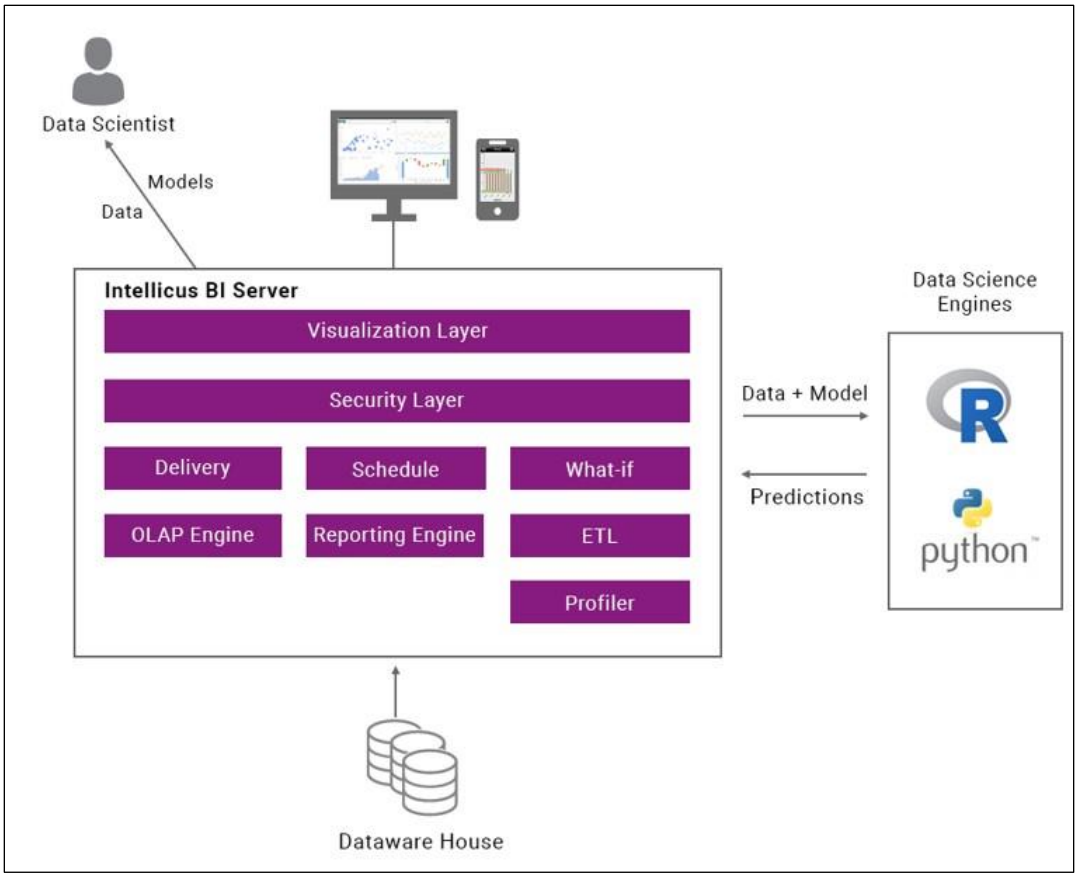
Intellicus BI can integrate with different data science environments to make your data-driven decision-making more powerful with predictions and what-if analysis.

Data Science is the technology that helps in analyzing data in-depth, form co-relations and patterns, bring out intelligent insights and future predictions for businesses. It studies historical data by performing machine learning and brings out insights from all possible combinations that help in better decision making. It has the capability to predict what our data would look like in the future, thereby leaving little to no doubt what decision to make.

Intellicus provides a seamless connection to different data science environments with access to all the available libraries to fulfill your Data Science requirements.

With What-If analysis, business users can analyze the predictions based on various business scenarios. They can play with the quantum of different variables to achieve a desired outcome and make their business strategies accordingly.

Below image shows how Intellicus connects with various Data Science engines and the process it follows to push your data to these engines and bring back the predictions.



Data Science works as an integrated platform with Intellicus. You need to create connections with different data science engines to let Intellicus communicate with them and process your data. You also need to create a file source connection that can work as an Exchange file System to help Intellicus and Data Science Engine to transmit data to each other. Both Intellicus and Data Science Engine should have read-write access to this exchange location. We have explained how to form these connections in Creating Connections section.

Once you create the required connections, you can add data science step while preparing your data at the Query Object level. When you add a data science engine step, Intellicus BI server performs the initial processing and then transmits the data along with a script (that you can build inside Intellicus) for advanced statistical computations to the Data Science engine. The data science engine based on the script processes your data (mostly learning and modelling) and transmits back to Intellicus. The processed data can now be used for further transformation steps and can be visualized on the Intellicus UI.

With Intellicus you can perform Data Science tasks while reporting as well. Intellicus gives you numerous intuitive charts to understand and analyze your current and predicted data.

As a pre-requisite, you need install a few libraries in R in addition to others you may use in your algorithms, to perform Data Science tasks in Intellicus, below are the libraries you need to install –

- lintr
- randomForest
- dplyr
- Rserve

Who can do what

Super administrators and/or users with specific roles and privileges assigned by super administrator can form connections.

Data Scientists or Designers can write R script and perform transformation steps.

End/Business users can use predictive and what-if analysis on the data prepared by designers/data scientists to form reports and visualize data with advanced visualizations

Note: You can create R scripts inside Intellicus' environment.

2 Creating Connections

Creating connection to R Data Science Engine requires a prerequisite file system-based connection. This connection's file location helps as a shared location to exchange data between Intellicus and R.

The first step will be to create a file system connection. The second step will be to create connection to Data Science Engine and providing the file connection location created in step 1.

Creating connection to File System

To create this, please follow the below steps.

1. Login to Intellicus – Navigate – Administration – Configure – Databases Tab
2. Click on Add
3. The page will display the following options

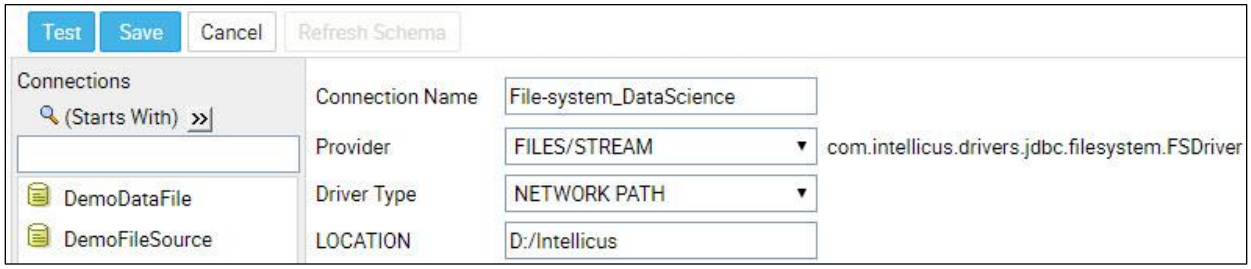


Figure 1: Adding a file system-based connection

Connection Properties

Item	Values	Comments
Connection Name	Provide a name as per your desire	This connection is used as a file-system to exchange data, and this name will be required while connecting to Data Science Engine
Provider	Select from the drop down	Intellicus stores the temporary processing data as a file, hence you need to select Files/Stream
Driver Type	Select from the drop down	You need to select network path to specify the location to save your file

LOCATION	Provide the location where the file will be saved	This location will be used by the Data Science Engine and Intellicus to exchange data while processing the data in Data Science step.

Make sure you uncheck read only in the settings of this connection as it will be accessed by both Intellicus and Data Science Engine to exchange data.

Note: To get more details to create a file system connection, please refer "WorkingwithDatabaseConnections.pdf"

Once you have given the required details, you can test your connection if it has been successfully created, Save it once you get the message 'Connection Test Succeeded.' Cancel if you want to start afresh.

You can Delete a connection once you have saved it or Refresh Schema to let Intellicus refresh the data from the connected database.

Creating connection to Data Science Engine

To create this, please follow the below steps –

1. Login to Intellicus – Navigate – Administration – Configure – Databases tab
2. Click on Add
3. The page will display the following options

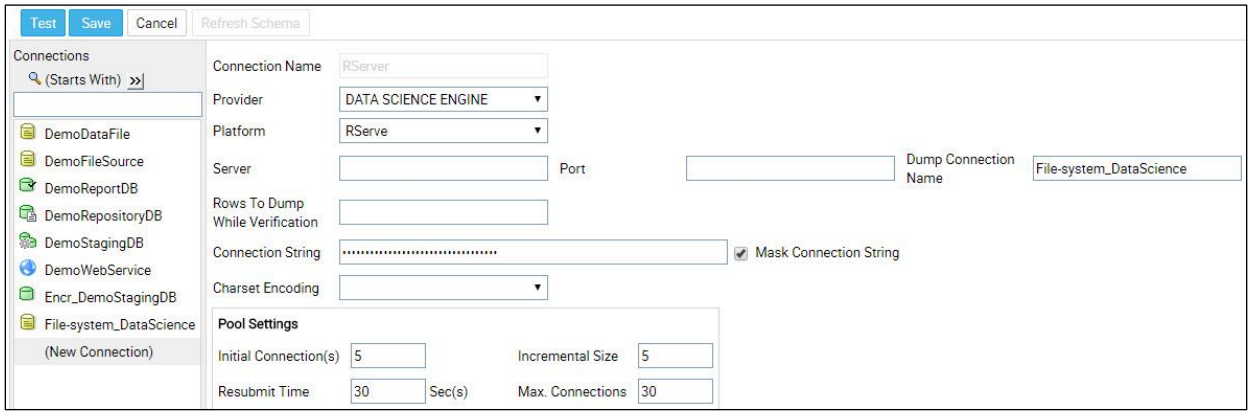


Figure 2: Creating a connection to Data Science Engine

Note: To create a connection to Data Science Engine, you must have a Data Science Engine running parallel on this network.

Data Science Engine Connection Properties

Item	Values	Comments
Connection Name	Rserver	This property will get affected once you choose Data Science Engine provider from the provider drop down. By default, the connection name will remain Rserver as Intellicus is providing connection to R server in this version. The name will be used as a reference at many places hence it is set by default **Note-** you need to select 'Provider' to be able to enter this field
Provider	Select DATA SCIENCE ENGINE from the drop down	Since we need to create a connection to Data Science engine, run to the bottom of the list and select DATA SCIENCE ENGINE
Platform	Select from the drop down	Specify the Data Science engine you need to connect. For now, RServe is available
Server	Type yourself	Provide the server IP address where your Data Science engine is running
Port	Type yourself	Provide the port details on which your Data Science engine is running
Dump Connection Name	Type yourself	Here you need to type the name of the file-based system connection you formed in the first step
Rows To Dump While Verification	Type yourself	Whatever number you enter here, only those many rows from your raw data will be dumped to verify the correctness of the script. By default, the filed remains blank.
Connection String	Autogenerated	Connection string to connect to the Data Science engine
Mask Connection String	Check/Uncheck	If checked connection string is masked

Charset Encoding		Select from list	Leave it blank
Pool Settings	Initial Connections	Type yourself	This feature helps you to define how many initial connections Intellicus will form to the engine so as once you start using it, there are no delays and your process carries out smoothly. Default: 5
	Incremental Size	Type yourself	Once all the available connections are used, increment size helps to increase the number of connections and forms new connections as per the specified number. Default: 5
	Resubmit Time	Type yourself	If the connections were increased and in use, resubmit time checks the current number of connections in use. If the connection value goes down to initial connection value, the incremented connections are released Default: 30 seconds
	Max Connections	Type yourself	The value here specifies how many connections can be formed at max. Say your max connections are 30 and all are in use. If any more connection request is raised, it will go to a queue Default: 30

Once you have given the required details, you can Test your connection if it has been successfully created, save it once you get the message 'Connection Test Succeeded.' Cancel if you want to start afresh.

You can Delete a connection once you have saved it or Refresh Schema to let Intellicus refresh the data from the connected source.

3 Data Science Engine step at Query Object level

Intellicus provides Data Science engine step while transformation of data and Predictive Analytics at Report level. In this section we will be mainly discussing on how you add Data Science Engine step at Query Object and what are its benefits.

You must have connection(s) to the database(s) to extract data for transformation and Data Science step.

To start transforming data, login to Intellicus – Navigate – Design – Query Object

Query object is the step where you extract your data from different databases and transform it to load and/or to use in reporting. You can learn more about working with a Query Object here – "WorkingwithQueryObjects.pdf"

Adding Data Science engine step at Query Object level helps you when predictions on your data are adding new variables and columns in tables. For example, in a market basket analysis, the clusters that would form may require new columns & variables in the table. This can be achieved while data preparation and hence such algorithms need to be defined at Query Object level.

You can also perform Data Cleansing and other Data Science engine related transformation tasks by creating script at Query Object level.

Adding Data

Data Science engines train on your data to bring out predictions. You can input Training as well as Prediction data based on the below conditions.

- If you have separate data to train and predict you need to add data for training as well as prediction.
- If you want training and prediction on the same data, only one data source can be added.
- If you already have a trained model in your script, you need not add training data.

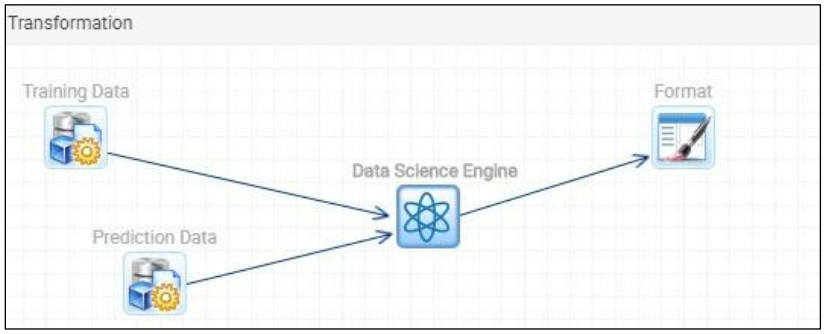


Figure 3: Adding Data

Training and prediction data consist of Independent and dependent variables. Independent variables are the fields in the data source that help in doing predictions i.e. determining the values of dependent variables. For instance, if you want to predict your sales for next year, then sales being a dependent variable will depend on independent variable fields like marketing expenditure, support expenditure, talent acquisition etc. to be predicted. Hence, you need to make sure you provide adequate information in your data.

Adding Data Science Engine step

Like adding steps for Data Source, Join, union etc., you need to drag and drop the Data Science Engine step from the left pane in the transformation area and create necessary links.

The Data Science Engine step takes 2 inputs. This step helps you to transmit your data to Data Science Engines to perform machine learning and modelling. You can add Data Science Engine step before or after adding any other transformation step.

You can add data science step before performing functions like join, union, formula fields etc. so as you can perform different functions on predicted data to further prepare it.

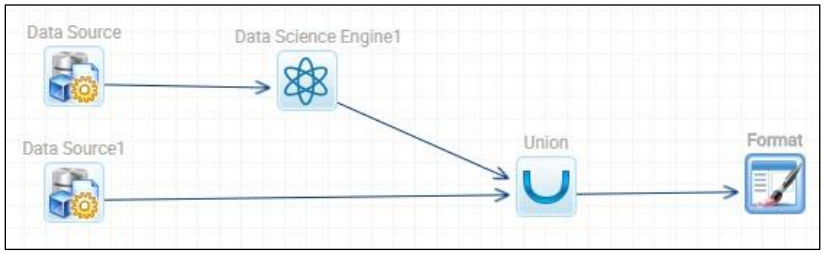


Figure 4: Adding Data Science Engine step before transforming with other steps

Or you can add the step in between or after preparing your data if you want to make predictions on your transformed data.

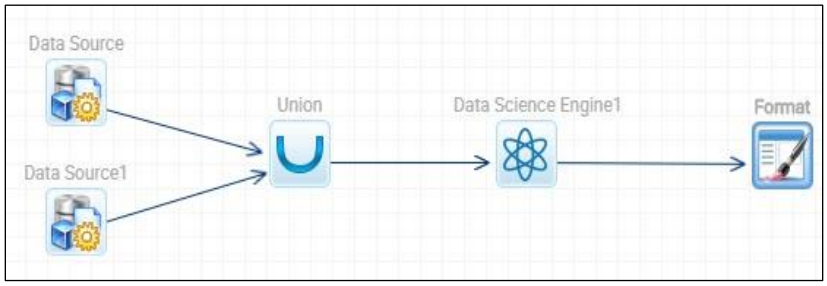


Figure 5: Adding Data Science Engine step after transforming the data with other steps

Adding Data Science script

Select the Data Science Engine step from the Query Object transformation area, you will see the following fields-

Figure 6: Data Science Engine properties

This inbuilt editor gives you the flexibility to write your own Data Science scripts inside Intellicus. This helps you save time and gives you option to verify your script by running it on a small set of data so as you can detect errors and correct them.

Data Science Engine Properties

Property	Values	Comments
Data Source Engine	R Job	Here you need to select the Data Science Engine you want to use
Script	Sample Script	Here you can see the Data Science Engine script you have created
Edit	Type Yourself	Click the Edit button to create Data Science Engine script or edit an already created script.

		 When you click the Edit button, the script editor box will open. Here you can view the fields in your script and write R script for relevant fields. You can also verify your script to check if it is error-free.

Guidelines for writing script at Query Object level

You can write your scripts inside Intellicus' environment for Data Science engines. There are few guidelines that you need to follow while creating scripts.

The guidelines are laid down so as Intellicus can understand and process your script and transmit it to Data Science engine for predictions.

Intellicus suggests creating R script in modular fashion at Query Object level that will help you to get options like Training only, Training and Prediction, and Prediction only at the time of report execution as Machine Learning Operations Toolbar. Training can then be scheduled and will save time if an end-user just wants to view the predictions.

For instance, if you schedule the training let's say in late hours, and next day a user wants to view prediction based on the trained model, he/she just needs to select predict only parameter while creating reports.

There are different sets of guidelines for writing scripts while creating script at Query Object level and while using Predictive Analytics at Report level.

- The script needs to have sections for Training and Prediction. These sections should start with #. These place holders should be surrounded by <%%> for Intellicus to be able to parse and understand the modularization. For e.g., #<% TRAINING.SECTION %>
- The first line of the Training and Prediction script should be for reading the CSV and the last line of Prediction script should be for writing. Argument passed in the reading section should be <% Stepname.data %> For e.g., Read.csv('<% Train.data %>')

- Previous step data should be referred as ‘StepName.data.’ For e.g., in the transformation area if you created the step as Train, the input must be ‘Train.data.’
- The model created is by default saved as ‘myModel.’ This is a mandatory name to the model you create as it is referred to while communicating with Data Science engines.
- The training will only happen if the training script is provided, otherwise it will be assumed that a trained model is used.
- If a trained model is used, it is mandatory for user to provide a prediction script.

Once you have added a script, you can Verify if it is correctly written and click on ok. Save or Save As your query object to use it in reporting.

An example script for your reference –

```
#<%TRAINING.SECTION%>
trainingDataset = read.csv('<%Train.Data%>')
library(randomForest)
myModel = randomForest(x = trainingDataset[1:15], y = trainingDataset$TEMP,ntree = 500)
#<%PREDICTION.SECTION%>
predictionDataset = read.csv('<%Predict.Data%>')
y_pred = predict(myModel,data.frame(predictionDataset[1:15]))
predictionDataset$ExpectedTemp <- y_pred
write.csv(predictionDataset , file='<%Predict.Data%>')
```

4 Running Reports

In Intellicus, Data Science engine step can be added at Query object level and Data Science tasks can also be performed at Report level. If you add a Data Science engine step at Query Object level, you will be able to see predictions once you run the report.

To create Intellicus reports you can refer the below manuals –

“DesigningAdhocReports.pdf”

“WorkingwithSmartView.pdf”

“DesktopStudio-ATour.pdf”

Once you create the necessary steps for a report to be generated with the Data Science engine step at Query Object level, you can run the report and visualize your data with predictions.

You will see the following options once you run the report:

Figure 7: Machine Learning Operations toolbar while running a report in Smart View

Note: The Machine Learning Operations will be visible if you add Data Science engine step with the necessary modular script at Query Object level.

You can choose between Prediction only or Training and Prediction from here. Prediction only will use a last trained model to bring out predictions, whereas, Training and Prediction will perform retraining based on the latest datasets before giving prediction. After selecting your choice, click Apply.

You can save your choice as default option every time you run a report by checking the box for Save Values for Next Run. Once you apply the setting you will be able to view predictions in your reports.

Performing Predictive Analytics

With Intellicus, business users can perform predictive analytics to get predictions on their data. Predictive Analytics helps you to input your script directly at report level and bring out predictions on your data. Adding script at report level is most useful when your predictions are not forming new variables or columns in your data reports.

Turn on the edit mode to view option for Predictive Analytics. You can perform predictive Analytics in Smart View Reports.

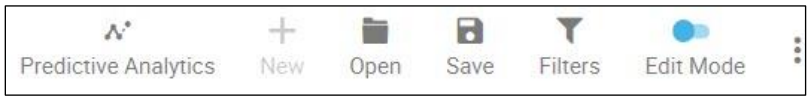


Figure 8: Tabs for Predictive Analytics and What-If Analysis

Predictive Analytics box will give you the options as shown in the image below:

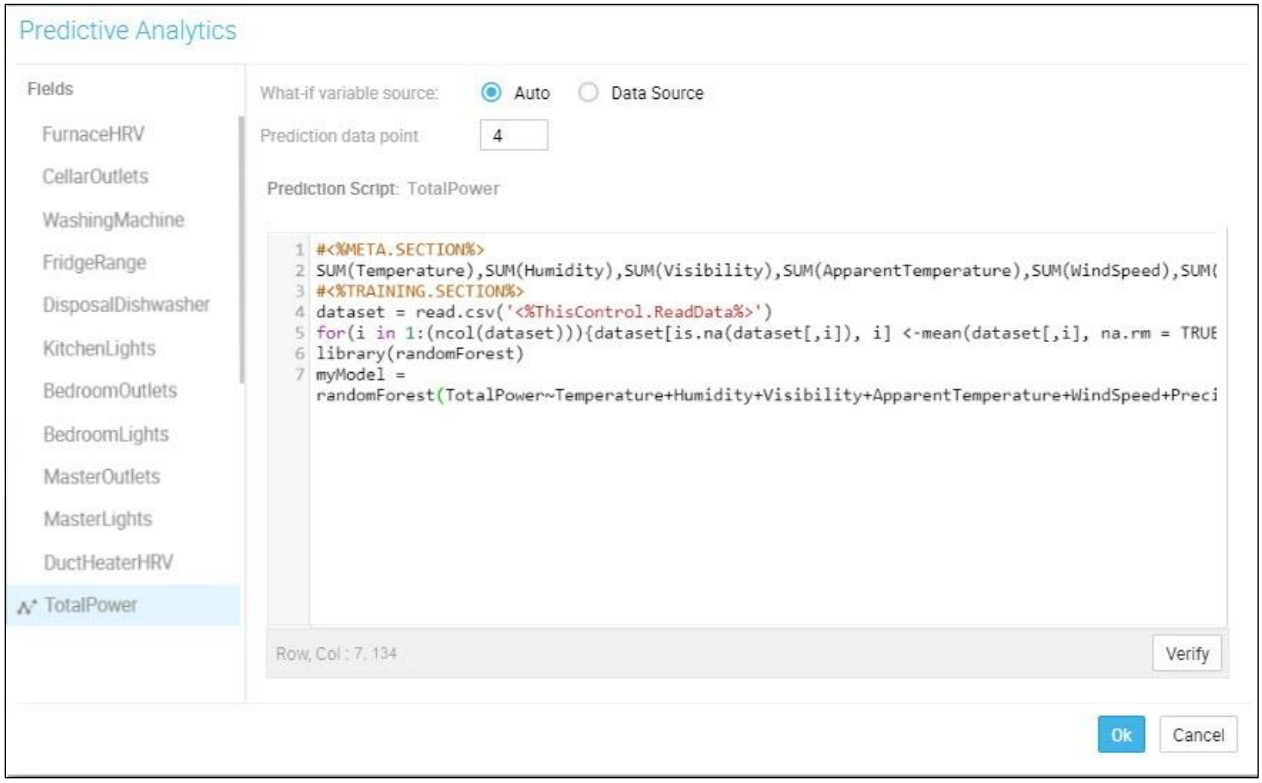


Figure 9: Performing predictive analytics

Fields

This will show the fields present in your report and you can choose on which fields you want predictions. Clicking any field will give you the ability to write Data Science script for that field.

Prediction Script

Here you can write the script for the field(s) you choose.

What-if Variable Source

Here you can select if you want the Data Science engine to analyze the variations in independent variables itself by selecting *Auto* or you can provide the data by selecting *Data Source*. Independent variables help to bring out predictions on dependent variables.
For example, if you want to predict Sales (dependent variable) your company would achieve in the coming years, you will have to provide marketing expenditure (independent variable), investment in infrastructure (independent variable), number of probable hires (independent variable) etc.
You can select *Auto* to let the Data Science engine learn the trend by reading your historic data and predict the values of independent variables. If you have pre-decided values, you can provide it using the Data Source option.

Auto

In Auto, you need to give the **prediction data point** in numeric value, for instance if you keep the value as 4, the predictions will be made for 4 units as per the intervals in your chart.

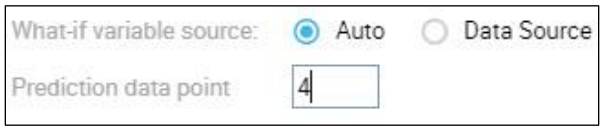


Figure 10: Select Auto in What-if variable source

Data Source

Here you need to specify the query object that has the prediction data (Independent variable values), to get predictions for the fields you want and provide script for the same.

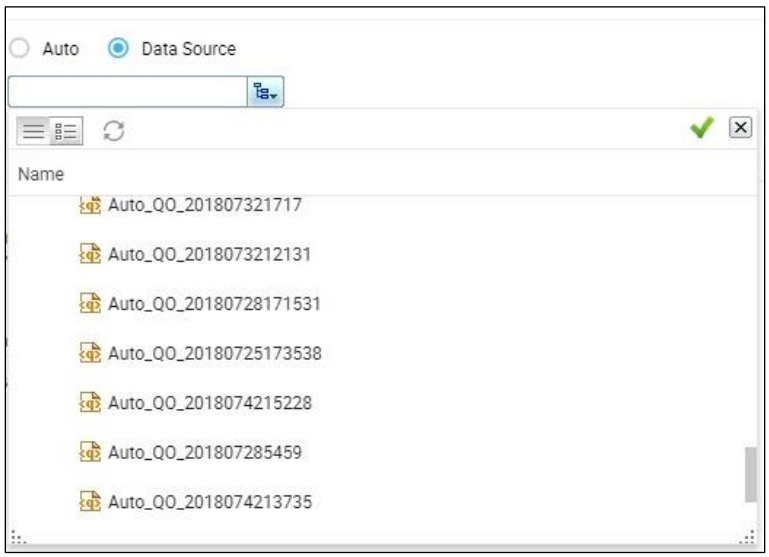


Figure 11: Options if you select Data Source

Upon adding the script, you can verify if the script is error free. Click OK if the verification process succeeds.

Guidelines for writing script at Report level

- The script needs to have Meta section for declaring independent variables, and sections for Training and Prediction. These sections should start with #. These place holders should be surrounded by <%%> for Intellicus to be able to parse and understand the modularization. For e.g., #<% TRAINING.SECTION %>
- Designer/ Data Scientist should specify comma separated independent variables in the comment section at the top of the script (line starting with #) under META.SECTION.
- Appropriate aggregation functions need to be defined while defining the independent variables. E.g. #SUM(Marketing_Spent),SUM(R&D_Spent)
- Independent Variables can either be numeric or categorical data. In case of categorical data, designers/ data scientists should write the script to handle the categorical data (encoding, feature scaling and decoding) in training as well as prediction script.
- If the encoders created in case of categorical data need to persist, the same has to be written by the designer/ data scientist in the script.
- The order of specifying Independent Variables should be considered as the schema of the dataset being used for training and prediction. User should consider this order while writing the script (in case of indexes and '.').
- The first line of the Training and Prediction script should be for reading the CSV. Argument passed in the reading section should be <%ThisControl.ReadData%> Ex. Read.csv('<%ThisControl.ReadData %>')
- The model created is by default saved as 'myModel.' This is a mandatory name to the model you create.
- The training will only happen if the training script is provided, otherwise it will be assumed that a trained model is used.
- If a trained model is used, it is mandatory for user to provide a prediction script.

Once you have added a script, you can Verify if it is appropriately written and click on ok.

An example script for your reference –

```
#<%META.SECTION%>
SUM(RnD Spent)
#<%TRAINING.SECTION%>
dataset = read.csv('<%ThisControl.ReadData%>')
for(i in 1:(ncol(dataset))){dataset[is.na(dataset[,i]), i] <-mean(dataset[,i], na.rm = TRUE)}
library(randomForest)
#myModel = randomForest(x = dataset[1:ncol(dataset)-1], y = dataset$Temperature,ntree = 1)
myModel = lm(Target.Profit~RnD.Spent,data=dataset)
```

What-if Analysis

With Intellicus you can perform What-if analysis to view predictions of different fields based on various business scenarios. For instance, if you want to know how much power will be consumed at a certain temperature, you can adjust the temperature value accordingly to get the prediction. This will help you to make planned decisions of your future actions for your business and help you in taking any other operational decision based on the predictions you derive.

To do What-if analysis, select Filters option and select What-if tab.

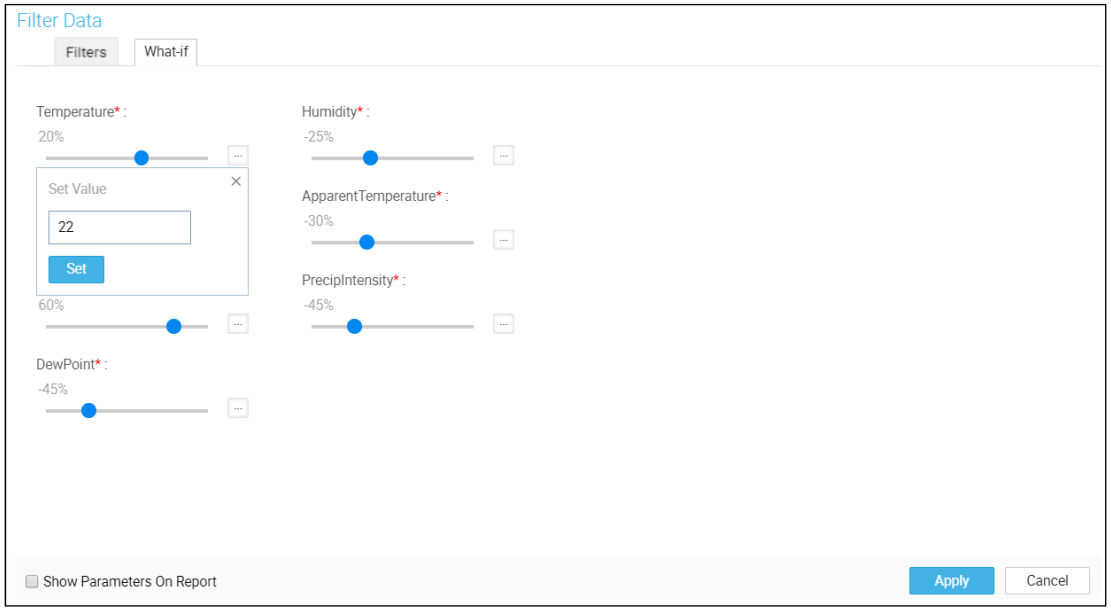


Figure 12: What-if analysis tab

You can use the slider to define the percentage values of different independent variables or manually set them. The values can be positive or negative, which implies the quantity you are increasing or decreasing from the current value. For example, if your current temperature is showing 20 degrees, setting a positive value by 20 percent will mean that the temperature will increase by 20 percent on 20 degrees and similarly decrease by the percentage you set for a negative value.

Click Apply once you have set the desired values and you will be able to view the predictions based on the values you have set.

An example of predictions achieved with the above use-case is shown below:

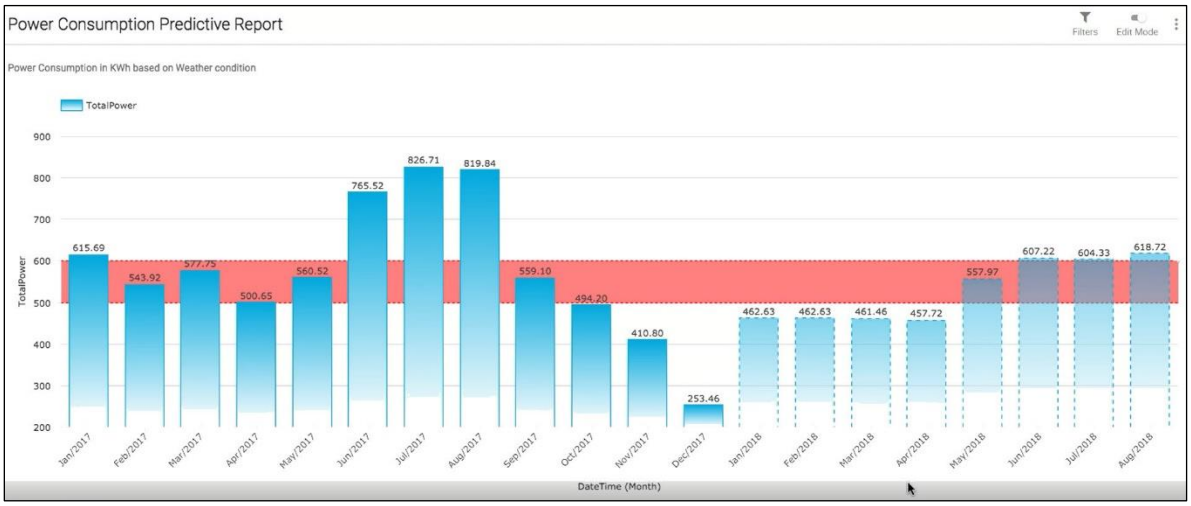


Figure 13: Predictive Report with What-if Analysis

Note: You can see the values of independent variables on the chart tooltip.