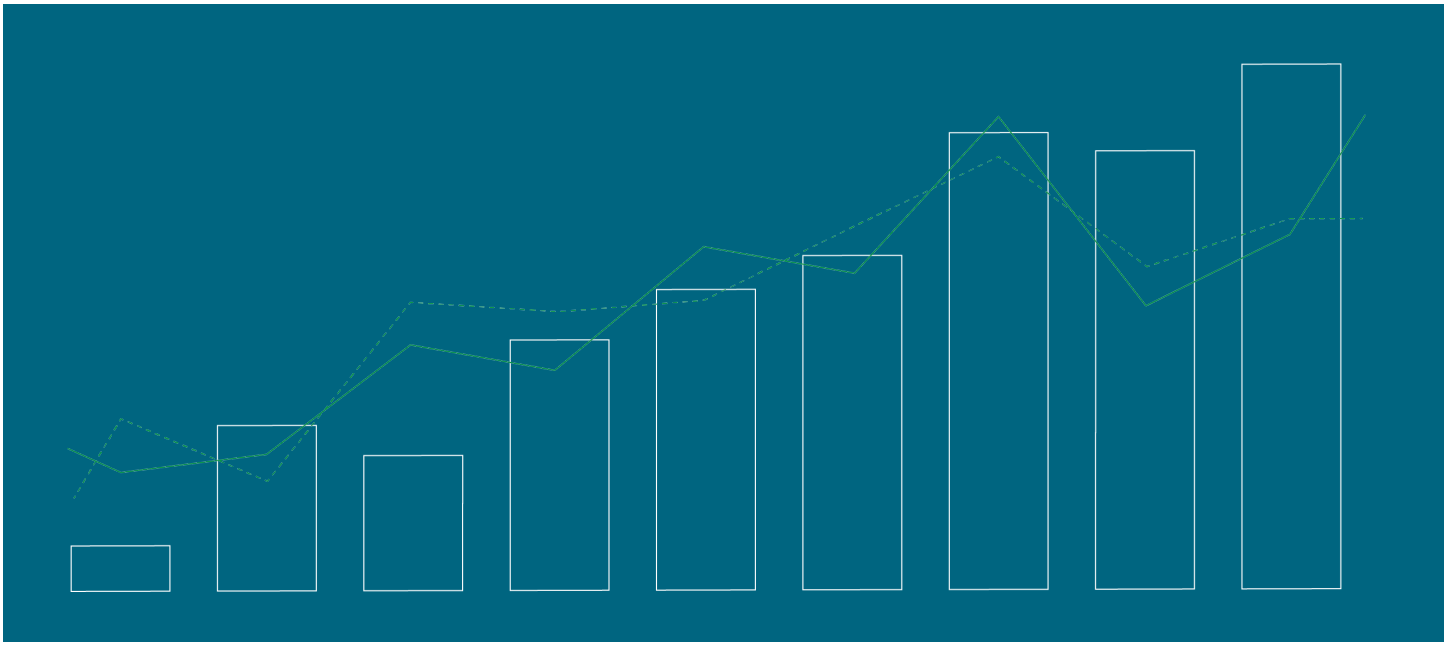
Qlik2DataRobot Installation Guide

Integrate Qlik with DataRobot



DataRobot

TABLE OF CONTENTS

What's required to get started? 2

Download the extensions 2

Qlik2DataRobot (Server Side Extension) 3

Installing Qlik2DataRobot on Windows 3

Configuring Qlik Sense Desktop 6

Configuring Qlik Sense Enterprise on Windows 7

Qlik2DataRobot Log Files 9

Qlik2DataRobot (Client Extension) 10

Installing on Qlik Sense Desktop 10

Installing on Qlik Sense Enterprise on Windows 10

Using Qlik2DataRobot 11

Sending data to DataRobot from Qlik Sense 11

Consuming a published DataRobot Deployment 15

Getting more advanced Prediction information 18

Appendix 1: Connector Specification 22

Supported Request Types 22

Document Information

This document is maintained with each version of the software, for the latest version visit:

https://github.com/AnalyticsEarth/Qlik2DataRobot/tree/master/docs

Author:

Steven Pressland, Qlik

Version Date:

14 March 2020

What's required to get started?

Before you can start integrating Qlik Sense and DataRobot you need to have the following components in place:

- Qlik Sense; this can either be the free Desktop edition which can be downloaded with a QlikID from qlik.com (a great way to get started and test the integration), or using Qlik Sense Enterprise.
- DataRobot; users will need an API key, which can be found on the account profile page within DataRobot.
- Connectivity from the Qlik2DataRobot application to both Qlik Sense and DataRobot. DataRobot typically runs against an internet facing URL and therefore the appropriate network access must be available to ensure server to server communication.

Download the extensions

Download the two installation packages required for installation on Windows and pick a location for the server side extension to be run, if not already downloaded to that machine, transfer Qlik2DataRobot Windows Installer.msi to a convenient location on that machine.

- If you are running Qlik Sense Desktop the installation will likely be on the same machine as desktop is running, but does not have to be.
- For Qlik Sense Enterprise you can install on an existing Qlik Sense server, however you are free to install on a separate machine if it is accessible by all Qlik Sense nodes running an engine service.

The Qlik2DataRobot installation files can be found on github (download all components from the releases page):

https://github.com/AnalyticsEarth/Qlik2DataRobot/releases

https://github.com/AnalyticsEarth/Qlik2DataRobot-Ext/releases

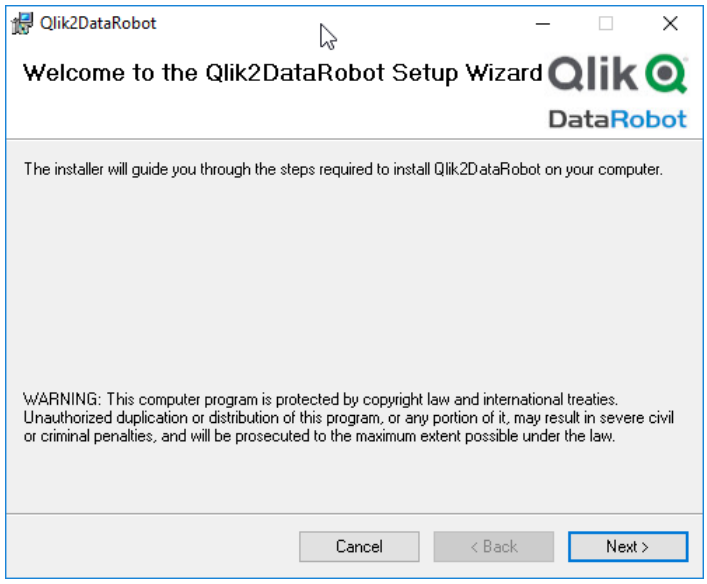
Qlik2DataRobot (Server Side Extension)

The first component to consider is the Server Side Extension, which will be configured as an Analytic Connector within Qlik Sense. This provides the link between the Qlik associative engine and DataRobot.

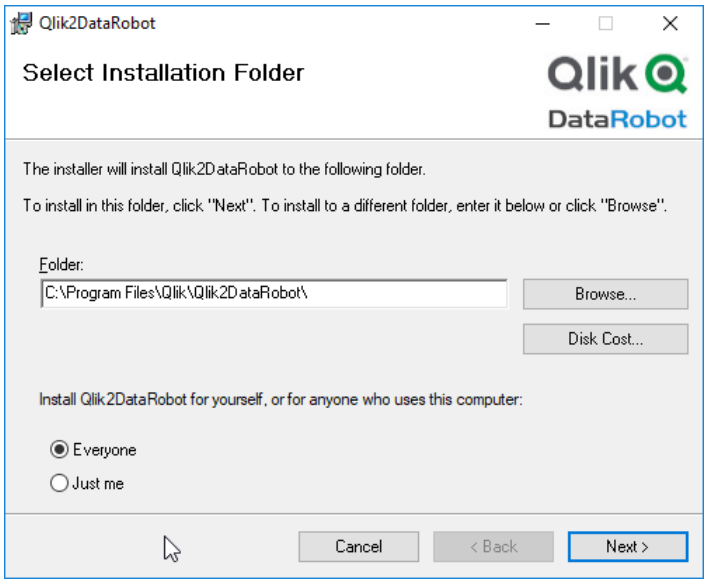
Installing Qlik2DataRobot on Windows

Installing on Windows is straightforward and achieved through an installation wizard package which will install and configure the application.

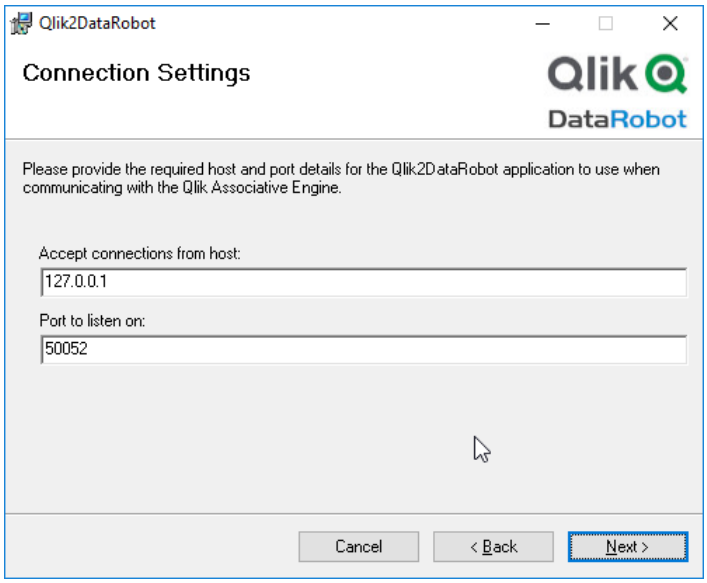
1. Run Qlik2DataRobot Windows Installer.msi



2. Click Next



3. Confirm the location you wish to install at click **Next**



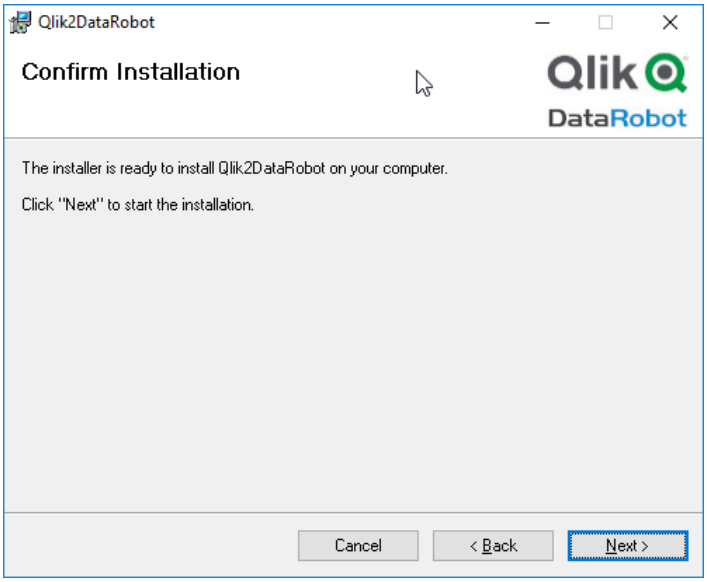
4. Update the connection settings required:

    a. The default settings will only allow connections coming from the local machine, if you are running on a different machine to the Qlik Associative Engine, the host setting will need to change. To allow connections from multiples sites or engines, use **0.0.0.0**.

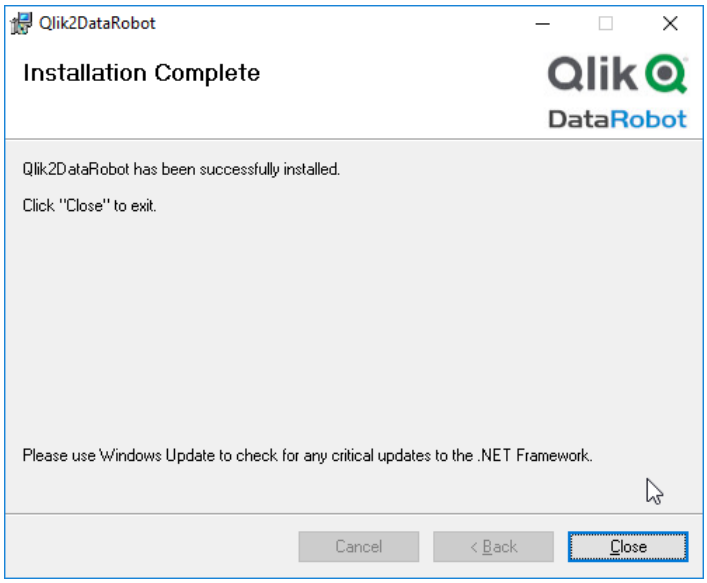
    b. The port is arbitrary, however if you already have a connector listening on this port, you will need to set a new one.

    c. These settings can be changed post installation by editing a configuration file.

    d. Click **Next**



5. Click **Next**

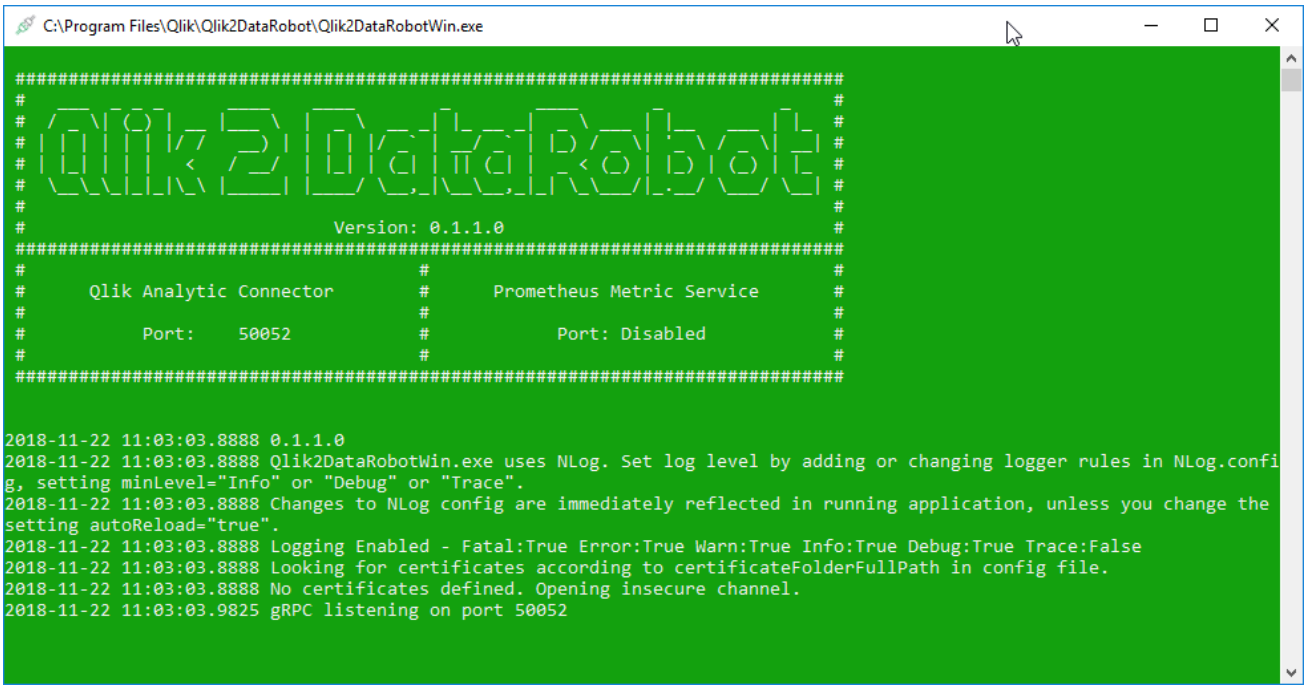


6. Once installation has completed, click **Close**

   You may get an installation error, this usually means that your machine does not have a compatible version of .NET framework. To get the latest version, visit https://dotnet.microsoft.com/download/dotnet-framework

7. Version 2+ of Qlik2DataRobot installs as a service and this will automatically start. However, you may want to manually run the connector, in which case you need to stop the windows service (and change its startup behavior if you want this to not start on boot) and run the Qlik2DataRobot application manually.

8. A new shortcut is provided on the windows application list. Launch the application (if not already running the windows service).

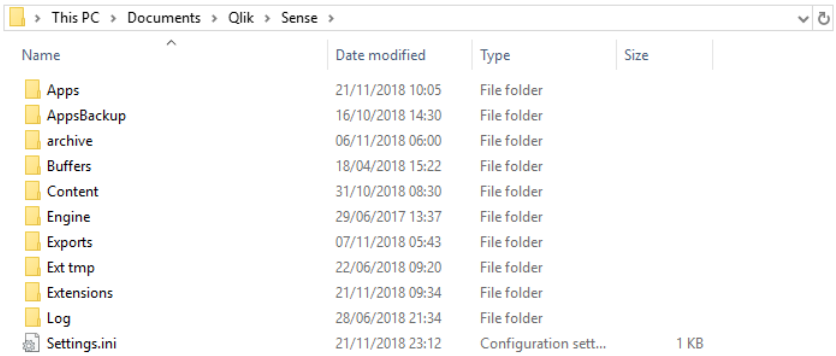


9. Confirm that Qlik2DataRobot is running. The application needs to be left open for the connectivity to work (you can minimize the window), however you may prefer to use the windows service if you require a log running background service.

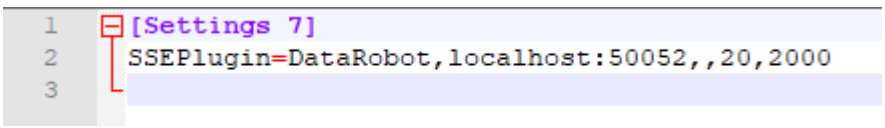
10. Always launch Qlik2DataRobot before starting Qlik Sense as the connection is made on start up.

Configuring Qlik Sense Desktop

Configuration of Qlik Sense Desktop is done using the Settings.ini file located at:



1. Open this file (or create one if it does not exist).



2. Within the [Settings 7] section add a SSEPlugin line:

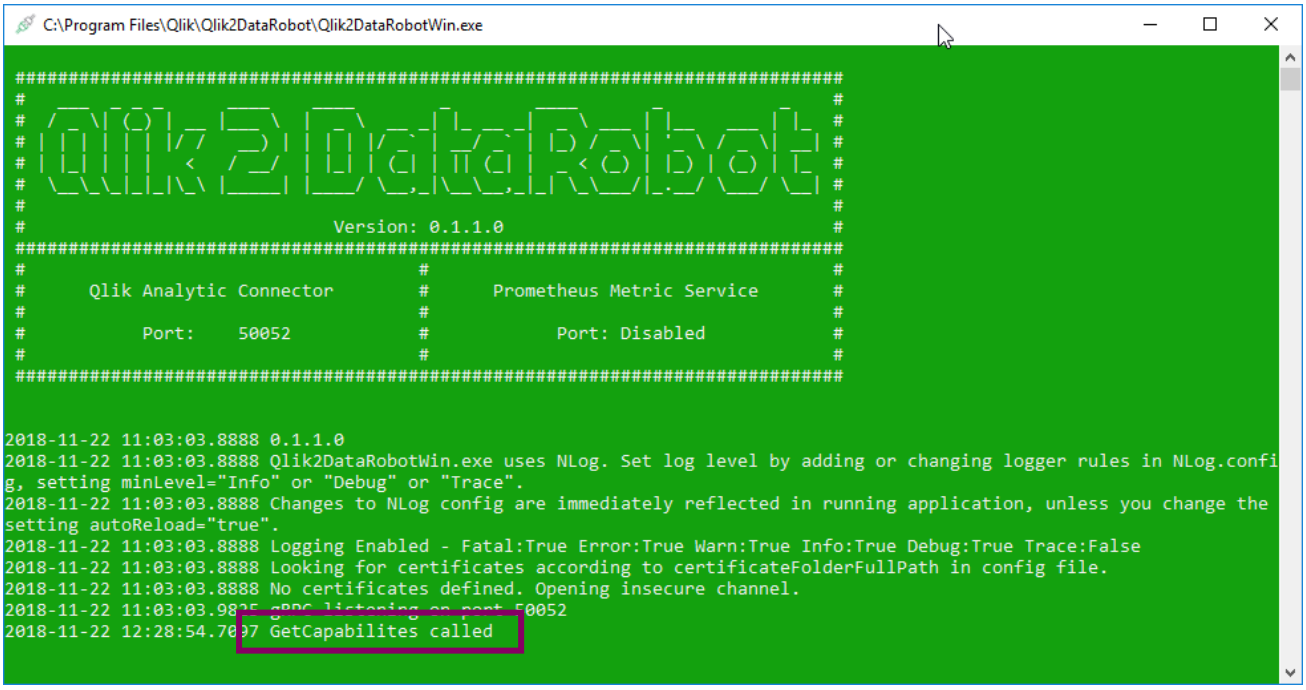
```
SSEPlugin=DataRobot,localhost:50052,,20,2000
```

a. If other plugins have been installed, the connection should be appended to the end of the line, delimited with a semi-colon:

```
SSEPlugin=R,localhost:50051,,2000,2000;DataRobot,localhost:50052,,20,2000
```

3. Ensure there is a blank line at the end of the text file.

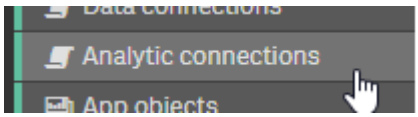
4. Start Qlik Sense Desktop, if everything is configured correctly a new log entry should appear in the Qlik2DataRobot window.



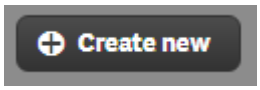
Configuring Qlik Sense Enterprise on Windows

Configuration of Qlik Sense Enterprise is performed through the QMC:

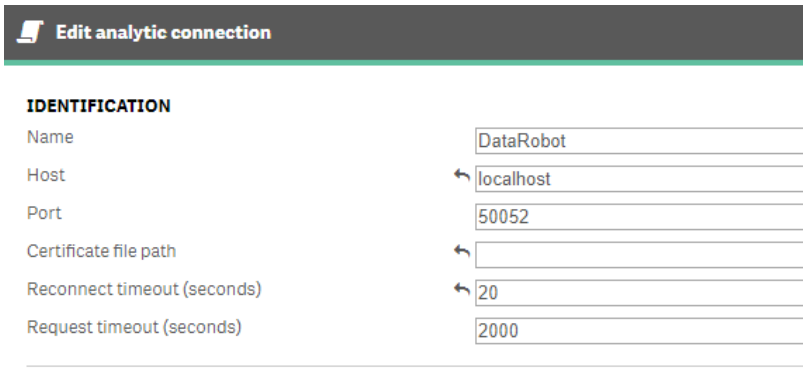
1. Navigate to the QMC for the Qlik Sense site



2. Select Analytic connections

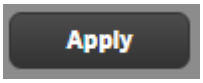


3. Click Create new



4. Enter the correct details to refer to the host and port you set when configuring the application above.

   The name by default should be called ***DataRobot****. If you choose to change this, you will need to make additional changes when configuring the client extension and script code snippets.*



5. Click Apply

6. All Qlik engines will attempt to connect to Qlik2DataRobot and you should see a *GetCapabilites* log item for each request.

Qlik2DataRobot Log Files

Qlik2DataRobot uses a logging technology called NLog. This will record the output from the connector to either a file or the console. In the event of running as a windows service, it is recommended to use the file logging. The default settings for the connector will save log files to c:\logs\ however you can change this path my changing the NLog.config file located in the installation folder you selected in the install wizard.

Changing the Log location

Edit the config file and replace the path with the folder of your choice:

```
<target xsi:type="File" name="file" fileName="c:/logs/${shortdate}.log"
            layout="${longdate} ${uppercase:${level}} ${message}" />
```

Change the Log Level

By default, the log level is set to Info, for more detailed logs you can specify Debug or Trace.

```
<logger name="*" minlevel="Info" writeTo="file" />
```

Qlik2DataRobot (Client Extension)

Qlik Sense client extensions are distributed as .zip files. The client extension for Qlik2DataRobot is called:

- qlik2datarobotext.zip

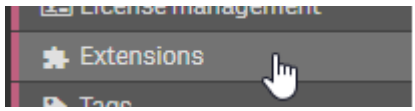
Installing on Qlik Sense Desktop

Unzip the contents of the file and copy to the extension folder, by standard located at:

- Documents\Qlik\Sense\Extensions
- The files should be placed in a folder called qlik2datarobotext

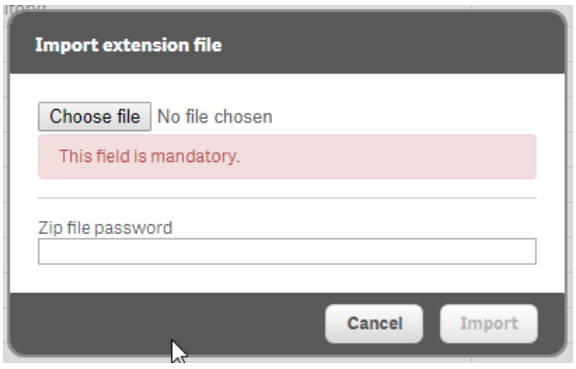
Installing on Qlik Sense Enterprise on Windows

Import the zip file using the QMC

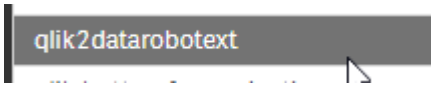


1. Click Extensions

2. Click Import



3. Select the zip file and click Import



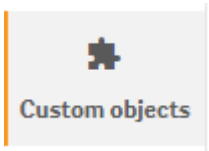
4. Confirm that the extension is listed in the extension list

Using Qlik2DataRobot

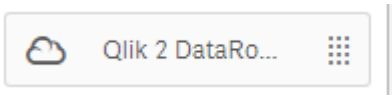
Sending data to DataRobot from Qlik Sense

Sending data from a Qlik application happens after the app has been loaded. The transfer is configured and triggered by the client extension. The data sent will be limited to that included in selections made by the user. All data transfers happen via the server side extension and no data is sent to or via the client browser.

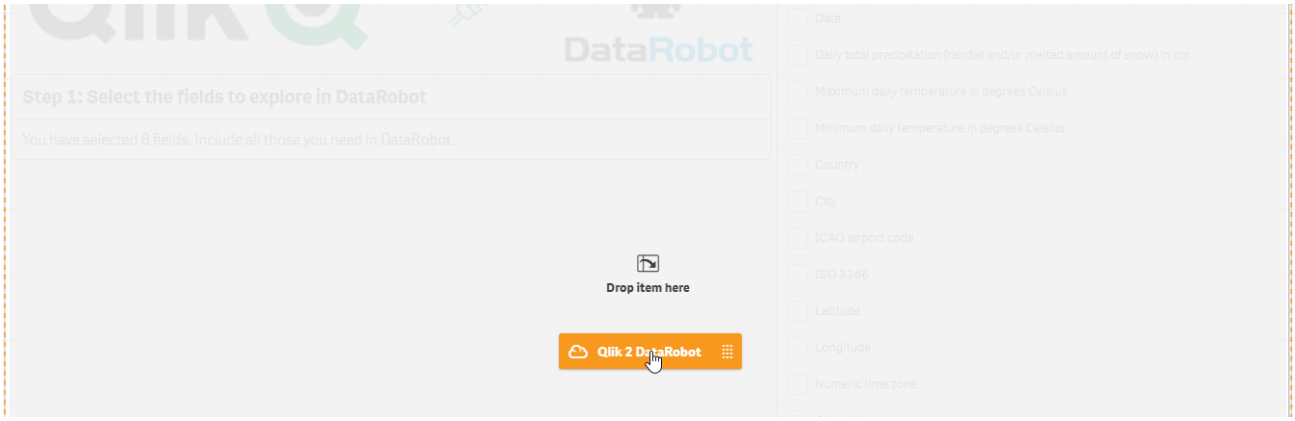
Add the *Qlik 2 DataRobot* extension from the Custom object menu:



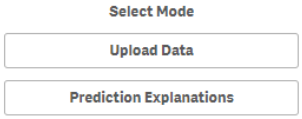
1. Select Custom objects



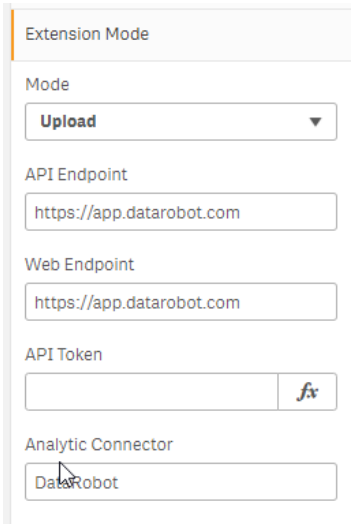
2. Drag Qlik 2 DataRobot onto the sheet



3. Select the extension mode (new for version 2) as “Upload Data”



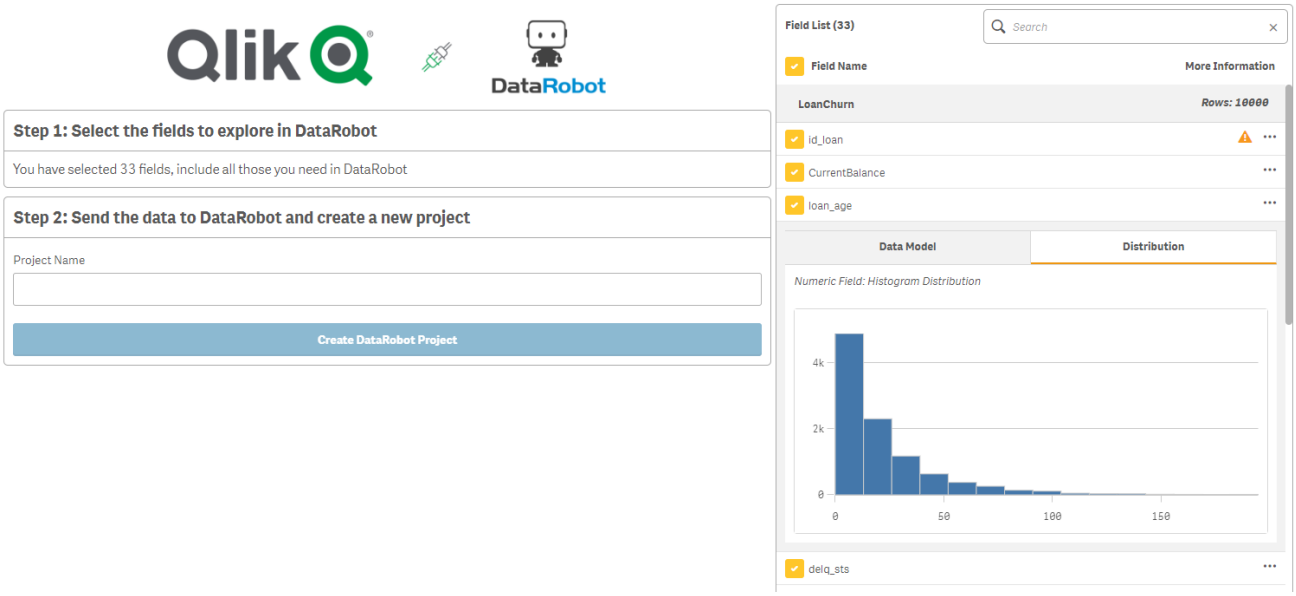
4. Configure the extension:



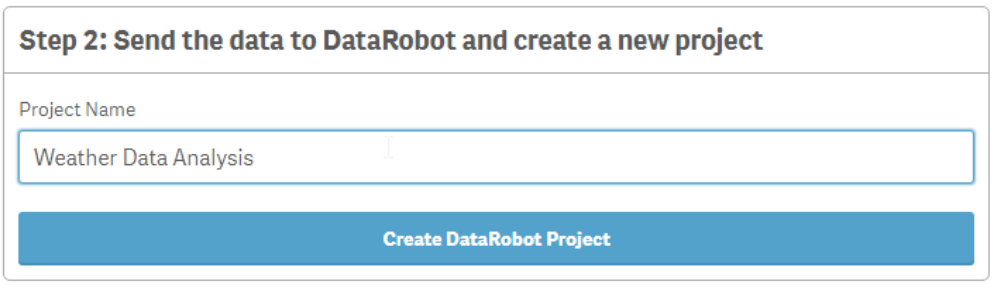
5. Enter your API Token, the other endpoints should not need changing, however if you have a different configuration it can be entered here.

   (If you do not enter the API token, you will be prompted to enter it each time when creating a project. This is helpful if the extension is part of a shared application)

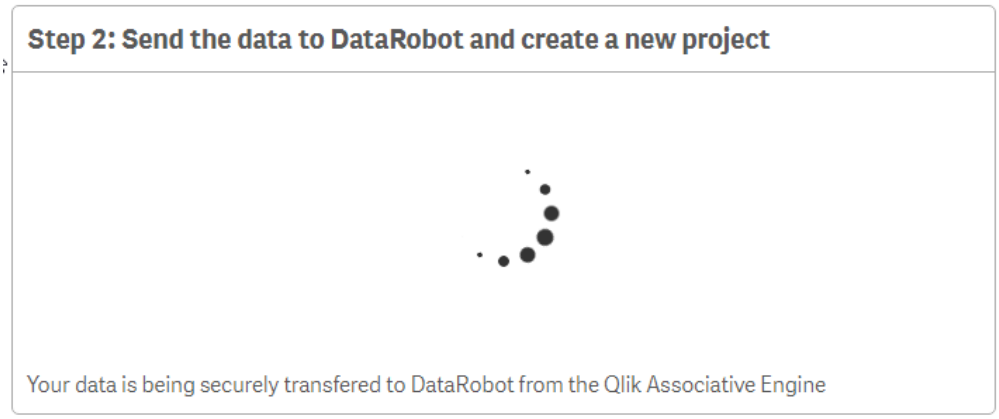
6. Click Done



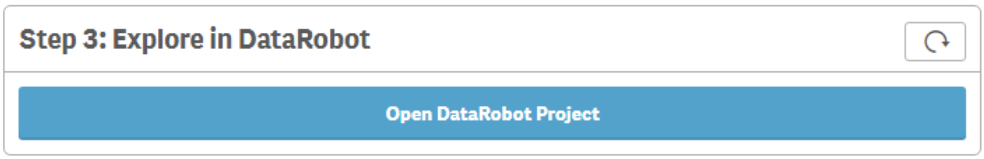
7. Select the fields you wish to send to DataRobot



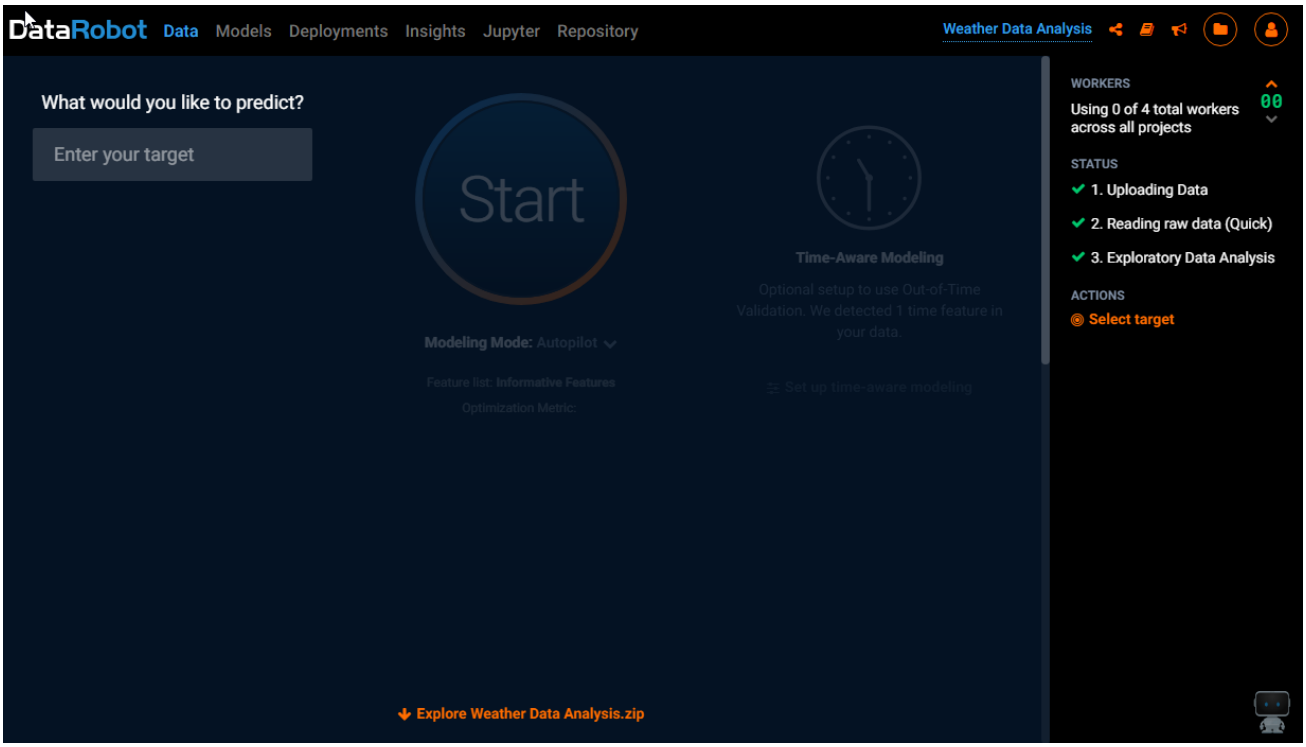
8. Enter the Project Name and click Create DataRobot Project



9. Wait for the data to transfer



10. Once the data transfer is complete, click Open DataRobot Project to launch in a new browser window.

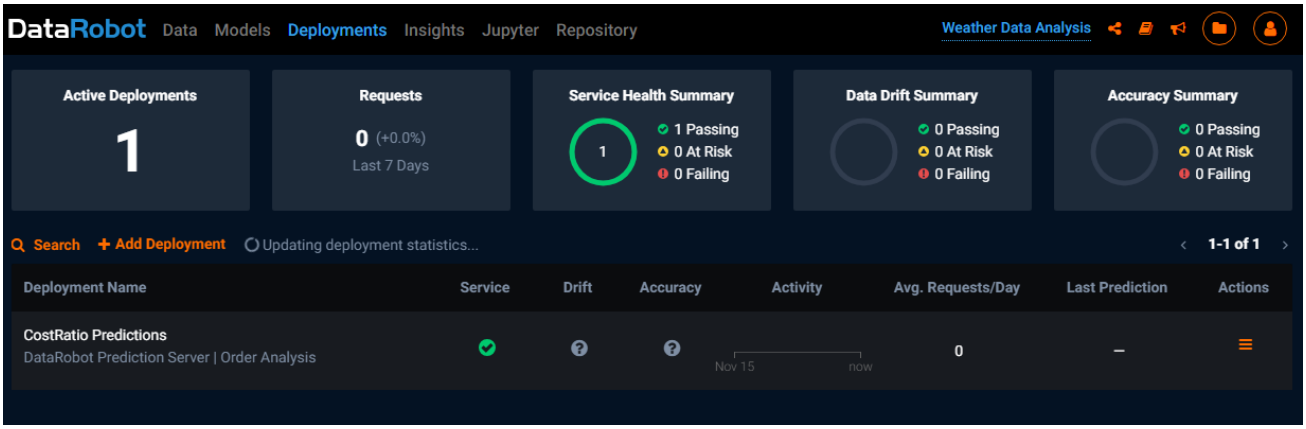

DataRobot
Data
Models
Deployments
Insights
Jupyter
Repository
Weather Data Analysis
What would you like to predict?
Enter your target
Start
Modeling Mode: Autopilot
Feature list: Informative Features
Optimization Metric:
Time-Aware Modeling
Optional setup to use Out-of-Time Validation. We detected 1 time feature in your data.
Set up time-aware modeling
Explore Weather Data Analysis.zip
WORKERS
Using 0 of 4 total workers across all projects
00
STATUS
1. Uploading Data
2. Reading raw data (Quick)
3. Exploratory Data Analysis
ACTIONS
Select target

Consuming a published DataRobot Deployment

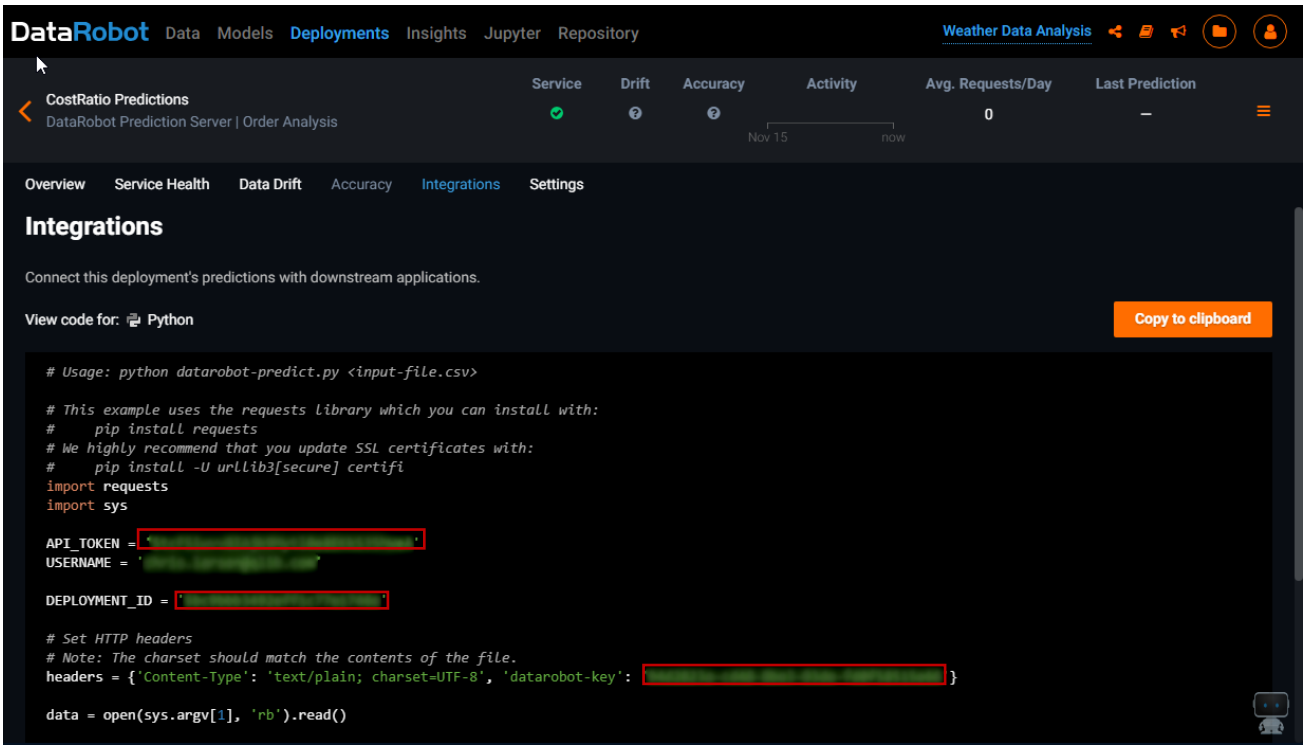
Once a deployment has been created within DataRobot this can be used to request a prediction from a Qlik application. The functionality used for this capability does not reply on a Qlik Sense specific extension, therefore is compatible with Qlik Sense, QlikView and Qlik Core.

Getting the required information from DataRobot

1. Select your deployment



2. Select Integrations



Using the sample Python code, make a note of the following parameters:

- API_TOKEN
- DEPLOYMENT_ID
- datarobot-key

Creating a Qlik application load script

Ensure all the data needed in DataRobot is in one Qlik table within your script. This may involve transforming your tables using joins for scoring. This table does not have to be used in the finished data model, so can be a temporary asset as required. This document does not cover how to produce this table.

The current version of Qlik2DataRobot requires the requests be made for a block of data no larger than the limit allowed by DataRobot deployments service (currently 50MB).

Using a Qlik script loop, the requests can be made in succession and combined back into a single results table. The result table will provide two columns of data, the field supplied in the request marked as the “keyfield” and the prediction value from the DataRobot API.

Example Script

The example script presumes the data is stored in a table called LoadStats

```
LET vRows = NoOfRows('LoanStats');
SET vBatchSize = 80000; //Set the batch size required
LET vGroups = $(vRows) / $(vBatchSize);

FOR i = 0 TO $(vGroups)
TRACE BATCH $(i);

[BatchData]:
NoConcatenate
LOAD
      [RecID],
      [loan_amnt],
      [term],
      [int_rate],
      [installment],
      [grade],
      [sub_grade],
      [emp_title],
      [emp_length],
      [home_ownership],
      [annual_inc],
      [verification_status],
      [issue_d],
      [loan_status],
      [pymnt_plan],
      [purpose],
      [title],
      [addr_state],
      [dti]
RESIDENT [LoanStats]
WHERE [RecID] > ($(i) * $(vBatchSize)) AND [RecID] <= (($(i) + 1) *
$(vBatchSize));

GetFromDataRobot:
LOAD
[RecID],
[Prediction]
EXTENSION DataRobot.ScriptEvalStr('{
    "request_type":"predictapi",
    "auth_config":{
        "api_token":"SdcF4iwyv8ILjk9Nytl3e8EHkS45hUeB",
        "endpoint":"https://customer.orm.datarobot.com",
        "datarobot_key":"84e2322a-cd38-8bf3-04da-fd6f57516e21"
    },
    "deployment_id":"5cf9abc3432eff2c75e1644f",
    "keyfield":"RecID"
}',BatchData);

Drop Table [BatchData];
Next
```

Getting more advanced Prediction information

Prediction Details

For classification models, you can receive the probabilities associated with each possible output label. To include this information, set **"inc_details":true** in the request.

Prediction Explanations

Prediction explanations are available and will be returned as either set of columns or a JSON string from the connector, which can be parsed in the client (for example when using the client extension in Prediction Explanation mode).

Prediction explanations can be loaded in the load script; however, they will slow down the results and may need to be requested in smaller batches.

Include the following in the request JSON:

```
"explain":{
      "max_codes":3,
      "threshold_high":0.5,
      "threshold_low":0.15,
      "return_raw":true
}
```

The **"return_raw"** is important as indicates whether the results be returned as a JSON object which will be stored in the field or not. You can parse that into Qlik data model fields or use the "Prediction Explanation" mode in the client extension.

When **return_raw** is false, a set of 5 fields will be returned for each number counting up to the value of **max_codes**. The fields are prefixed with the value **PE_n** where **n** is a number.

The fields returned are (using the first explanation as an example):

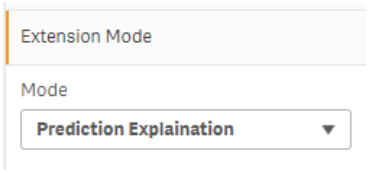
- **PE_1_label**
- **PE_1_feature**
- **PE_1_featureValue**
- **PE_1_strength**
- **PE_1_qualitativeStrength**

These fields can be converted into an associated table, an example script for a similar use case is shown here as reference:

```
ChurnPredictions:
LOAD
id_loan,
Prediction,
[Prediction value for label: Yes],
[Prediction value for label: No],
WillChurn
RESIDENT
ChurnPrediction;

For pe = 1 to 3
PE:
LOAD
id_loan,
[PE_$(pe)_label] as pe_label,
[PE_$(pe)_feature] as pe_feature,
[PE_$(pe)_featureValue] as pe_featureValue,
[PE_$(pe)_strength] as pe_strength,
[PE_$(pe)_qualitativeStrength] as pe_qualitativeStrength
RESIDENT ChurnPrediction
WHERE [PE_$(pe)_label] <> '';

Next pe;

DROP TABLE ChurnPrediction;
```

Using the Prediction Explanation client extension mode

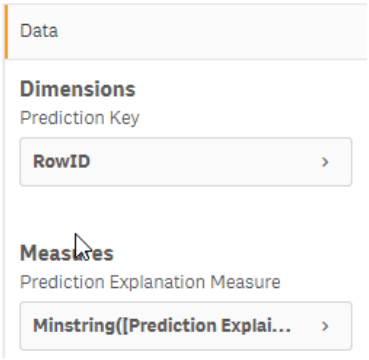
Select “Prediction Explanation” from the Extension Mode.



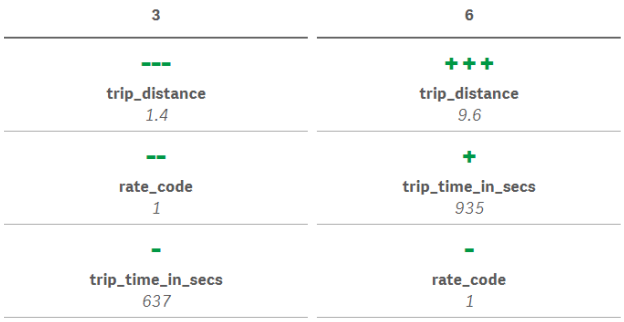
In the data section, select the “Prediction Key”, this should be the primary key that uniquely identifies a single row and prevents aggregation of the explanations. For the measure select

```
“Minstring([Prediction Explanations])”
```

if you have loaded the default column name.



Typically, the extension is used to view a small number of explanations and this should be limited by selection and the calculation condition on the extension.



The predictions can also be made using the chart expression syntax. The below example is an abridged version from the supplied sample app.

```
DataRobot.ScriptEvalStr(
'{
    "request_type":"predictapi",
    "auth_config":{
        "api_token":"$(vApiToken)",
        "endpoint":"$(vEndpoint)",
        "datarobot_key":"$(vDataRobotKey)"
    },
    "deployment_id":"$(vDeploymentId)",
    "should_cache":true,
    "explain":{
      "max_codes":3,
        "threshold_high":0.5,
        "threshold_low":0.15,
        "return_raw":true
    }
}',
(int_rt + $(vIntRate)) as int_rt,
orig_upb,
Margin,
CurrentBalance,
)
```

Appendix 1: Connector Specification

The Qlik2DataRobot connector uses the SSE protocol for communication. More specifically it uses the script evaluation functions to provide a more flexible framework for integration.

In the context of this connector, the "Script" expected in the SSE protocol should be a JSON message.

The JSON message specification differs based upon the value of the mandatory **"request_type"** object.

Base JSON message:

```
{
      "request_type":"<REQUEST TYPE>",
      "auth_config":{
            "api_token":"<API TOKEN>",
            "endpoint":"<API ENDPOINT>"
      }
}
```

Supported Request Types

createproject

```
{
    "request_type":"createproject",
    "auth_config":{
        "api_token":"<API TOKEN>",
        "endpoint":"<API ENDPOINT>"
    },
      "project_name":"<PROJECT NAME>"
}
```

Example Values:

Parameter	Type	Required	Example
API TOKEN	String	Yes	SdcF4iwyv8ILjk9Nytl3e8EHkS45hUeB
API ENDPOINT	String (URL)	Yes	https://app.datarobot.com/api/v2
PROJECT NAME	String	Yes	Data Churn Analysis

predictapi

```
{
	"request_type":"predictapi",
	"auth_config":{
		"api_token":"<API TOKEN>",
		"endpoint":"<ENDPOINT>",
		"datarobot_key":"<DATAROBOT KEY>"
	},
	"deployment_id":"<DEPLOYMENT ID>",
	"keyfield":"<KEYFIELD>",
	"should_cache":<SHOULD CACHE>,
	"inc_details":<INC DETAILS>,
	"explain": {
		"max_codes":<MAX CODES>,
		"threshold_high":<THRESHOLD HIGH>,
		"threshold_low":<THRESHOLD LOW>,
		"return_raw":<RETURN RAW>
	}

}
```

Example Values:

Parameter	Type	Required	Example
API TOKEN	String	Yes	SdcF4iwyv8ILjk9Nytl3e8EHkS45hUeB
ENDPOINT	String (URL)	Yes	https://customer.orm.datarobot.com
DATAROBOT KEY	String	Yes	84e2322a-cd38-8bf3-04da-fd6f57516e21
DEPLOYMENT ID	String	Yes	5cf9abc3432eff2c75e1644f
KEYFIELD	String	No	RowID (NOT for use in chart expressions)
SHOULD CACHE	Boolean	No	true (defaults to false, only for use in chart expressions)
INC DETAILS	Boolean	No	True (defaults to false)
MAX CODES	Integer	Yes (when explain included)	3
THRESHOLD HIGH	Decimal		0.5
THRESHOLD LOW	Decimal		0.15
RETURN RAW	Boolean		true

timeseries

```
{
      "request_type":"timeseries",
      "auth_config":{
            "api_token":"<API TOKEN>",
            "endpoint":"<ENDPOINT>",
            "datarobot_key":"<DATAROBOT KEY>"
      },
      "deployment_id":"<DEPLOYMENT ID>",
      "should_cache":<SHOULD CACHE>,
      "inc_details":<INC DETAILS>,
      "forecast_point":<FORECAST POINT>,
      "timestamp_field":<TIMESTAMP FIELD>,
      "timestamp_format":<TIMESTAMP FORMAT>
}
```

Example Values:

Parameter	Type	Required	Example
API TOKEN	String	Yes	SdcF4iwyv8ILjk9Nytl3e8EHkS45hUeB
ENDPOINT	String (URL)	Yes	https://customer.orm.datarobot.com
DATAROBOT KEY	String	Yes	84e2322a-cd38-8bf3-04da-fd6f57516e21
DEPLOYMENT ID	String	Yes	5cf9abc3432eff2c75e1644f
SHOULD CACHE	Boolean	No	true (defaults to false, only for use in chart expressions)
INC DETAILS	Boolean	No	True (defaults to false)
FORECAST POINT	String	Yes	2017-01-20 00:00:00
TIMESTAMP FIELD	String	Yes	timestamp
TIMESTAMP FORMAT	String	Yes	yyyy-MM-dd HH:mm:ss

Note regarding timeseries format

The **`timestamp_format`** needs to specify the original timestamp field format as it was upload to DataRobot when the original model was trained. If the format is not correct, you will be presented with an error and the intended format. The format however must be formatted using dotnet standards.

https://docs.microsoft.com/en-us/dotnet/standard/base-types/custom-date-and-time-format-strings

About Qlik

Qlik is on a mission to create a data-literate world, where everyone can use data to solve their most challenging problems. Only Qlik's end-to-end data management and analytics platform brings together all of an organization's data from any source, enabling people at any skill level to use their curiosity to uncover new insights. Companies use Qlik to see more deeply into customer behavior, reinvent business processes, discover new revenue streams, and balance risk and reward. Qlik does business in more than 100 countries and serves over 48,000 customers around the world.

qlik.com

© 2018 QlikTech International AB. All rights reserved. Qlik®, Qlik Sense®, QlikView®, QlikTech®, Qlik Cloud®, Qlik DataMarket®, Qlik Analytics Platform®, Qlik NPrinting®, Qlik Connectors®, Qlik GeoAnalytics®, Qlik Core®, Associative Difference®, Lead with Data™, Qlik Data Catalyst™, Qlik Associative Big Data Index™ and the QlikTech logos are trademarks of QlikTech International AB that have been registered in one or more countries. Other marks and logos mentioned herein are trademarks or registered trademarks of their respective owners.
CODE NEEDED FOR PUBLISHING