Machine Learning

SYSPRO 8

Reference Guide

Published: October 2024

SYSPRO

CONTENTS

AI Machine Learning

AI Machine Learning

Machine learning forms part of the ***SYSPRO Artificial Intelligence*** module. It uses specific algorithms and statistics to examine historical data. The program then uses the data patterns to reveal trends and predict future outcomes, benefiting management by presenting the big business picture.

Although these predictions require minimal human intervention, they rely heavily on the data quality and the attributes of the SQL statement. Reliable predictions will support business processes and improve decision making by learning from past experiences.

Exploring

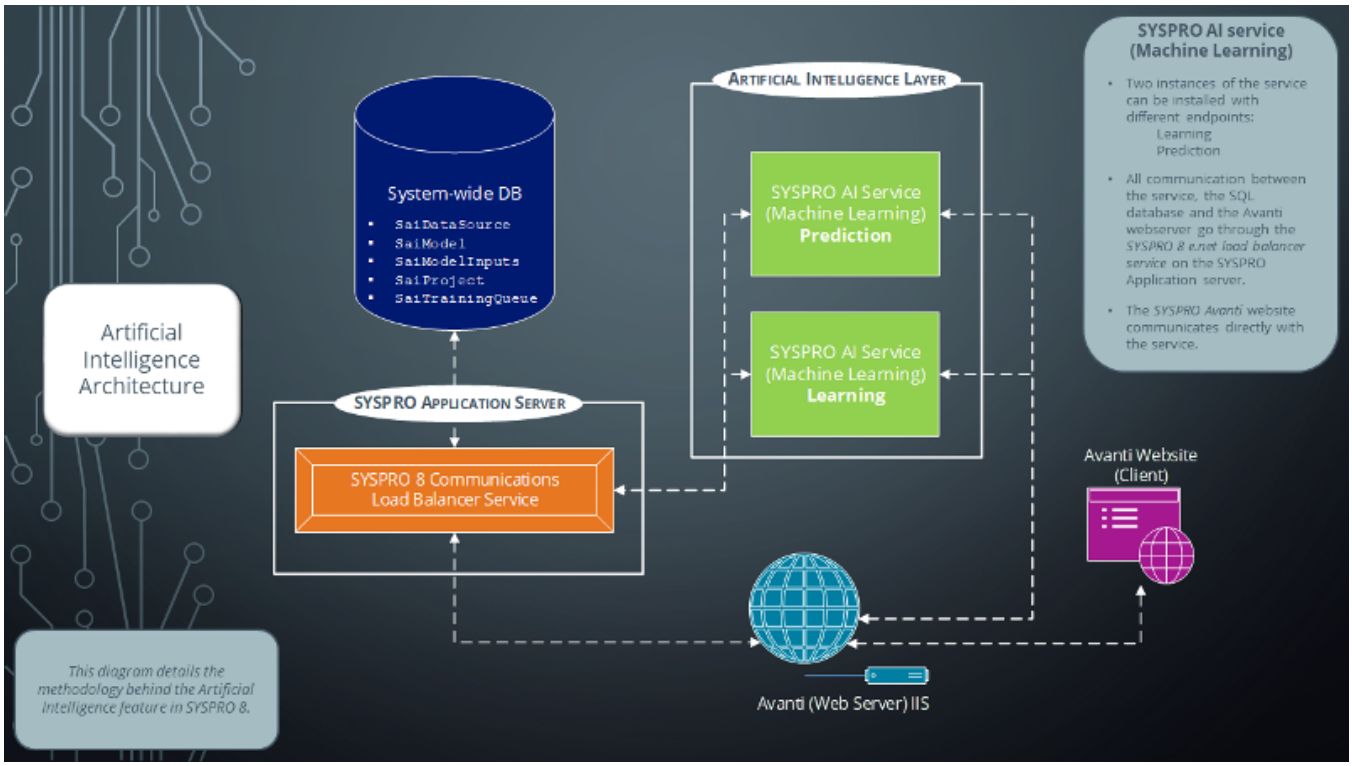
Machine Learning, the engine behind artificial intelligence, uses specific algorithms and statistics to examine historical data. This gives computers the ability to learn from large amounts of business and industry data sets in order to provide analysis and insights, which is increasingly required as part of the decision making process.

Business insights can be used in conjunction with machine learning projects by creating tiles with defined KPI thresholds. The KPIs are defined using the **Insight Tile Definition** program in SYSPRO, whereas the actual tiles are created in ***SYSPRO Web UI (Avanti)***. The tiles display a visual representation of the prediction results and indicate the proximity of the threshold, which can facilitate informed business decisions.

You access the ***SYSPRO Artificial Intelligence*** module from the SYSPRO Avanti website (client). Accessing and processing of information is then done via the **IIS** and **Avanti Web Server**. The **Web Server** communicates with the **SYSPRO 8 Machine Learning** service (AI Layer) and the **SYSPRO 8 e.net Communications Load Balancer** service (SYSPRO App Server).

The **SYSPRO 8 Machine Learning** service can have 2 instances installed, with different endpoints for Training and Prediction. It sends all communications through to the **SYSPRO 8 e.net Communications Load Balancer** service, which then calls business objects via e.Net to read and write data to the system-wide database. Data is then added to (or retrieved) from the database and the communication is sent back to ***SYSPRO Web UI (Avanti)*** and displayed on the user interface (graphically depicted by the bi-directional arrows).

AI Architecture Diagram



The following sample projects are shipped with the product to help kick-start the machine learning process. They are located in the `Harmony_SaiProject` file in the `\Base\Harmony\Standard` folder:

Regression

- LCT days late

  This predicts how late a shipment may be, measured in days.
- PO days late

  This predicts the number of days a purchase order line may be late.
- Chance order will be late

  This displays a percentage that predicts the chance a purchase order line may be late.
- Customer invoice pay days

  This predicts the number of days a customer may take to pay an invoice.

Classification

- Job status

  This predicts whether a job may be completed early or late.
- Lost sales reason

- This prediction determines whether sales may be lost and what the reason for losses would be.
- Problems with sales order

  This predicts the most likely reason why a sales order is returned.
- Quote success

  This predicts how many quotes should be accepted by the customer and result in sales orders.
- Stock code profitability

  This predicts how profitable a stock code should be by location.
- Supplier performance

  This predicts how well a supplier should perform in terms of deliveries, and indicates the likelihood of the delivery being on time and in full.
- AP invoice payment prediction

  This predicts by when a supplier should pay an invoice.
- Customer classification

  This predicts how profitable a customer would be.

Anomaly Detection

- Purchase order line anomaly

  This detects anomalies on purchase order lines based on the selected columns.

- Sales order line anomaly

  This detects anomalies on sales order lines based on the selected columns.

Navigation

- ***SYSPRO Web UI (Avanti)***

  SYSPRO Avanti > SYSPRO Artificial Intelligence

Terminology

Project (ML & AI)

A machine learning project consists of a data-source, a SQL statement, and machine learning training options.

The project tells the SYSPRO Artificial Intelligence module what data must be used for training, and provides various options to control the training process. The SYSPRO Artificial Intelligence module uses the SQL statement to query the data-source. It then learns the patterns in the data to produce a model.

Data source

A data-source is a reusable link to an SQL Server instance hosting the database and consists of the address of the server, login credentials, and other options specific to the server in question.

The actual dataset that used in the machine learning project, is specified in the SQL query within a project.

The data-source can be setup once, and then reused by any user to quickly train a model based on data on that specific SQL Server.

Training (ML & AI)

Training a model involves examining patterns in the data using various machine learning algorithms.

The data can be thought of as containing 1 or more input columns (`x`), and 1 output column (`y`). Training is the process of learning how `x` maps to `y`.

Symbolically, if we say `f(x) = y`, then f is some function that can map `x` to `y`. In mathematics, we are usually given `f` and `x`, and asked to calculate `y`. In supervised machine learning, we give the computer `x` and `y`, and ask it to learn `f`.

Model (ML & AI)

A machine learning model is a reusable block of code that can make predictions on new, or unseen data. The model is a binary object that cannot be directly inspected.

The ***SYSPRO Artificial Intelligence*** module takes care of loading models. To make a prediction, you have to provide the same number & type of columns on which the model was trained. The model will then produce a prediction.

For example: You may have trained the project to predict a value `y1`, based on the input values of `a1, b1, c1`. If you provide new values of `a2, b2, c2`, the model will output a new value of `y2`, based on those new input values.

Prediction

A machine learning prediction requires a trained model and an active project.

When performing a prediction, the ***SYSPRO Artificial Intelligence*** module uses a new set of input data to forecast or foresee a possible scenario or outcome. This can be considered a more advanced form of a what-if analysis.

Rather than relying on simple linear relationships, like a pro-rata calculation, a machine learning prediction can predict non-linear, or multi-dimensional relationships.

Prediction end point

This is the rest endpoint address of the ***SYSPRO Artificial Intelligence*** service (e.g. `http://localhost:30238/SYSPROMLE/rest`) that is used by the **SYSPRO Machine Learning** module to perform predictions.

Training end point

This is the training endpoint address to the ***SYSPRO Artificial Intelligence*** service (e.g. `http://localhost:30238/SYSPROMLE/rest`) that is used to train projects to generate a model that can be used to perform predictions.

Starting

Prerequisites

Technology

The following technology prerequisites are applicable to using this feature:

- Microsoft .NET Framework 4.6
- SYSPRO 8 e.net Communications Load Balancer

  A valid endpoint must be configured in the **Setup Options** program of ***SYSPRO 8***.
- SYSPRO 8 Machine Learning

  This service can be installed on any server as long as the **SYSPRO 8 e.net Communications Load Balancer** endpoint is configured correctly in the service's configuration file.
- SYSPRO Avanti Web Service
- SYSPRO 8 Avanti Initialization Service

You can use the **SYSPRO Installer Application** to install these requirements.

Configuring

Once you have installed the **SYSPRO 8 Machine Learning** service, the following setup options must be configured to use this feature:

Artificial Intelligence System Setup

Setup Options > System Setup > Artificial Intelligence

- Prediction endpoint
- Training endpoint

Installation considerations

- If you have two installations of the **SYSPRO 8 Machine Learning** service installed on different servers, then you can configure SYSPRO to use a **TRAINING END POINT** and a **PREDICTION END POINT**.

  Separate endpoints for training and predicting ensures better responsiveness, especially as the training endpoint may seem unresponsive when training projects.

Security

You can secure this feature by implementing a range of controls against the affected programs. Although not all these controls are applicable to each feature, they include the following:

- You restrict operator access to *activities* within a program using the **Operator Maintenance** program.
- You can restrict operator access to the *fields* within a program (configured using the **Operator Maintenance** program).
- You can restrict operator access to *functions* within a program using passwords (configured using the **Password Definition** program). When defined, the password must be entered before you can access the function.
- You can restrict access to the eSignature *transactions* within a program at operator, group, role or company level (configured using the **Electronic Signature Configuration Setup** program). Electronic Signatures provide security access, transaction logging and event triggering that gives you greater control over your system changes.
- You can restrict operator access to *programs* by assigning them to groups and applying access control against the group (configured using the **Operator Groups** program).
- You can restrict operator access to *programs* by assigning them to roles and applying access control against the role (configured using the **Role Management** program).

Restrictions and Limits

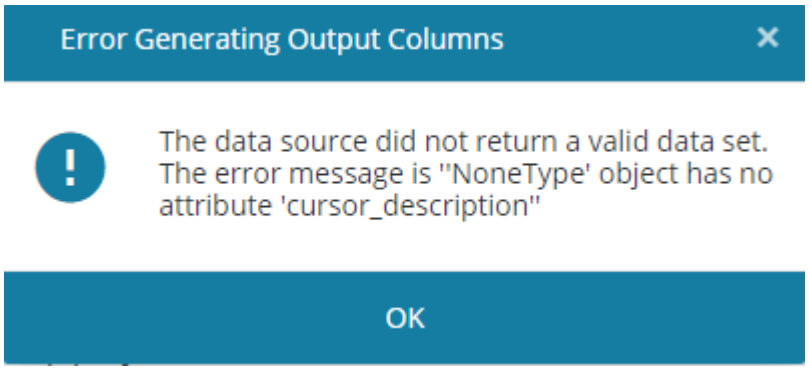
- A machine learning project can only be created in ***SYSPRO Web UI (Avanti)***.

Solving

System messages

Error messages

Error generating output columns



Cause

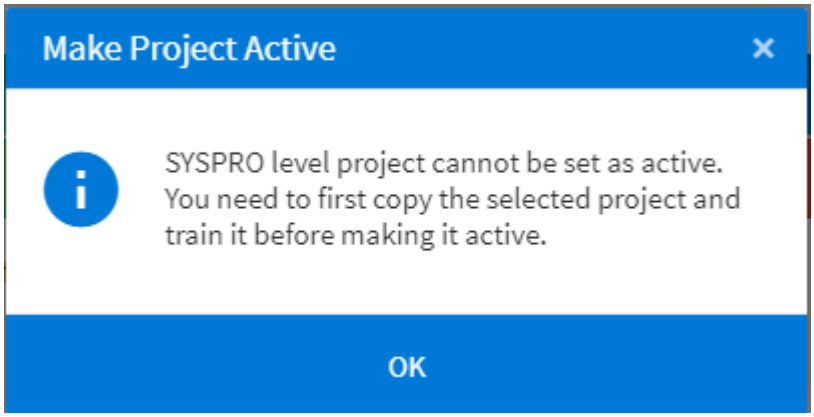
This message is displayed when you save an AI project in the **SYSPRO Machine Learning** program if the SQL select statement returns only one or zero rows of data.

Solution

Ensure that the SQL statement used to create the project returns more than one row of data, as projects require at least 2 rows of data to be trained.

Informational messages

Make Project Active



Synopsis

This message is displayed when you attempt to activate a project that hasn't been trained yet.

Solution

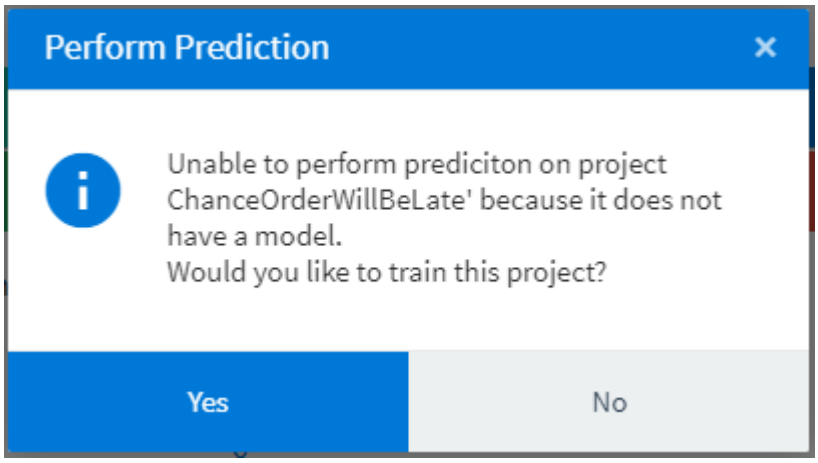
Before activating the machine learning project, train it so that it gets to know the data.

1. Open ***SYSPRO Web UI (Avanti)***.
2. From the menu, select **SYSPRO Artificial Intelligence** and select **AI Administrator**.

   The SYSPRO Artificial Intelligence application is displayed.
3. Select the project you want to train from the **AI Projects** list.
4. Select **Train Project** to add the project to the training queue.

   The **Training Queue window** is displayed.
5. Once the training is done, the status **Training completed** is displayed and a model is created that can be used for predictions.

Perform Prediction



Synopsis

This message is displayed when you attempt to perform a prediction on a project that hasn't been trained yet.

Solution

Before using the machine learning project for predictions, it must be trained it so that it gets to know the data.

1. Open ***SYSPRO Web UI (Avanti)***.
2. From the menu, select **SYSPRO Artificial Intelligence** and select **AI Administrator**.

   The SYSPRO Artificial Intelligence application is displayed.
3. Select the project you want to train from the **AI Projects** list.
4. Select **Train Project** to add the project to the training queue.

   The **Training Queue window** is displayed.
5. Once the training is done, the status **Training completed** is displayed and a model is created that can be used for predictions.

FAQs

Working with projects and data sources

Is it necessary to create more data sources?

You don't need to create your own data sources, as they are automatically created for every SYSPRO company.

You can create additional data sources for data that resides outside of the SYSPRO database using the **Data source List** window (*SYSPRO Avanti* > *SYSPRO Artificial Intelligence* > *Setup* > *Data Source List*).

Can I train more than one project at a time?

No, due to the intensity of the training process, only one project can be trained at a time.

You can, however, queue projects for training, which will automatically start the next project as soon as the previous project is completed.

Can I edit and delete the SYSPRO projects?

No, the shipped projects can't be edited or deleted.

You can, however, create a copy of a shipped project. The copied project can then be trained and the model used for predictions.

What is the difference between saving the project on system, company or role level?

The level on which the project is saved, determines which models are available for users and how the predictions are done.

Models that are saved on role level, are only available for operators who are grouped into that role. Models that are saved on company level, are available for all operators in that company. Models that are saved on system level are available for all users across companies.

When predicting an outcome on operator level, the system uses the model that is saved for the current operator role. If there is no model for that operator role, the system first determines if a model exists for the current company and then whether a model exists on system-wide level.

Why is my machine learning project not displayed in the Insight Tile KPI Definition program?

This may be due to one of the following reasons:

- The ML project must be active.
- The ML project must be trained.
- The ML project must be a **regression** type project.

Why can't I delete the projects in the SYSPRO folder?

The projects listed in the SYSPRO folder are sample projects that are shipped with the product.

You can only copy sample projects to create a new version, but you can't delete or edit them. Only the copied versions can be edited and deleted.

Sample projects are located in the `Harmony_SaiProject` file in the `\Base\Harmony\Standard` folder.

Why is there no data in the Feature importance graph?

The model may need to be retrained in order to display the feature importance percentage.

Working with models

What do the records in the Model information section represent?

The **Model information** pane is used to display basic statistics about the trained project. To compare models and identify whether one model is better than another, some objective metric is required.

The ML Engine supports various metrics that are defined in the project using the **Scoring method**. The data set is split into a training set and a testing set. The ML Engine learns from the training set and is not allowed to see the testing set. Once a model is created, its performance on unseen data can be measured by passing the testing set into the model and comparing predicted values in the training set. This process of comparing predicted values to actual values can be used to objectively score the model.

The scores presented to the user are the result of this comparison.

- The **Scoring method** function is used to evaluate the quality of the given pipeline for the classification problem.
- The **Total set score** indicates how the model scored over the entire data set i.e. both the training and testing data sets combined.
- The **Possible error** percentage indicates how the model performed on the testing data set and is, therefore, a better indication of the prediction performance on unseen data.
- The **Test Observations** figure is the total number of records used for the selected prediction results.
- The **Training Observations** figure is the total number of records returned from the SQL statement.

Project scoring metrics

Which scoring metrics can be used for regression type projects?

The following scoring metrics can be used in regression projects:

- neg_mean_squared_error
- neg_mean_absolute_error
- neg_mean_absolute_error
- r2

Which scoring metrics can be used for classification type projects?

The following scoring metrics can be used in classification projects:

- accuracy
- adjusted_rand_score
- average_precision
- balanced_accuracy
- f1
- f1_macro
- f1_micro
- f1_samples
- f1_weighted
- neg_log_loss
- precision
- precision_macro
- precision_micro
- precision_samples
- precision_weighted
- recall
- recall_macro
- recall_micro
- recall_samples
- recall_weighted
- roc_auc

Which scoring metrics can be used for anomaly type projects?

There are no scoring methods for anomaly type projects. Anomaly detectors can't be scored as they are unsupervised machine learning algorithms and don't use analogous scoring metrics.

Service operators

What is the difference between a service operator and a normal operator?

Service operators are signed in by a *SYSPRO service* and are used for any functionality that is required by the service to obtain information about SYSPRO.

Normal operators sign in to SYSPRO with a username and password. They access certain programs and functions to perform specific tasks.

> **FOR EXAMPLE:**
>
> the __SRS service operator is used by the **SYSPRO 8 Reporting Host Service** to obtain information about the companies in an environment, but it isn't used by the business object to retrieve the data for a report, as that would be the specific normal operator that submits the request for the report.

What are service operators and what are they used for?

Service operators are created by SYSPRO and are used by SYSPRO services to obtain information about SYSPRO.

The service operator code starts with a double underscore to differentiate them from other operators. A default company code must be assigned to each service operator within the **Operator Maintenance** program, as we use the company code to log in the service operator via e.net .

The following is a list of service operators and their function within SYSPRO:

- The __ADSYNC service operator is used by the **SYSPRO 8 Active Directory Sync Service** to push **Microsoft Active Directory (AD)** information into SYSPRO for Active Directory managed operators.
- The __DFM service operator is used by the **SYSPRO 8 Document Flow Manager Folder Poller** and **SYSPRO 8 Document Flow Manager Queue Poller** to monitor folders, send files to the queue and process files.
- The __ESP service operator is used by the following services:
  - **SYSPRO 8 Espresso Service**,
  - **SYSPRO 8 Espresso Notification Service**,
  - **SYSPRO Espresso Development Plugin** and the
  - **SYSPRO Avanti Web Service** to obtain information for the password reset and forgot password functionality.

- The __POS service operator is used by the **SYSPRO 8 Point of Sale Services** to determine and validate the setup options and required credentials at start up, update the required databases and post to SYSPRO (if the Point of Sale operator doesn't have access to SYSPRO).
- The __RUL service operator is used by the **SYSPRO 8 Rules Engine Service** and the **SYSPRO 8 Rules Data Service**.
- The __SA service operator is used by the **SYSPRO 8 Analytics** service to make business object calls.
- The __SAI service operator is used by the **SYSPRO 8 Machine Learning** service.
- The __SRS service operator is used by the **SYSPRO 8 Reporting Host Service** and the **SYSPRO 8 Cognitive Service** to manage client-side report printing.

- Only specific services use service operators to log in via e.net.
- SYSPRO creates service operators by copying the ADMIN operator. If the ADMIN operator record doesn't exist (i.e. it may have been deleted), then the current operator is used when saving system details from the **Setup Options** program.

Using

Process

How to configure machine learning

The configuration of **SYSPRO Machine Learning** has to be done within the core SYSPRO product.

1. Open the **System Setup** program and navigate to the **Artificial Intelligence** tab.
2. At the **MACHINE LEARNING** section, enter the **PREDICTION END POINT** and **TRAINING END POINT** REST addresses (e.g. `http://localhost:30238/SYSPROMLE/REST` where `localhost` indicates the server name and `30238` indicates the port number).

   Separate endpoints for training and predicting ensures better responsiveness, as especially the training endpoint may seem unresponsive when training projects.
3. Save your changes.

How to create a machine learning project

SYSPRO Web UI (Avanti)

1. Open ***SYSPRO Web UI (Avanti)***.
2. From the menu, select **SYSPRO Artificial Intelligence** and **AI Administrator**.

   This loads the **SYSPRO Machine Learning** program.
3. Create a new learning project.
4. Create a new data source (if you don't have one).
5. Create a new project.
6. Add the machine learning project to the training queue.

   Once the training is done, the training queue displays a **Training completed** status and a model is created that can be used for predictions.
7. Specify input values for the model to run a prediction.

How to create or edit a data source

1. Open ***SYSPRO Web UI (Avanti)***.
2. From the menu, select **SYSPRO Artificial Intelligence** and **Setup**, then select **Data Source List**.
3. Select **New Data Source** to create your own data source or select an existing one to edit it.
4. Enter the data source details and test the connection.

> SYSPRO only supports the ODBC data connection.

How to link a prediction to tile with KPIs

1. Access the **AI Administrator** program from within ***SYSPRO Web UI (Avanti)***.

   This loads the **SYSPRO Machine Learning** program, where you can train and activate projects.
2. Access the **Insight Tile Definition** program from within SYSPRO 8 to set operator permissions and define KPIs for the tile.
3. Access the program or location in ***SYSPRO Web UI (Avanti)*** where you want to place the tile.
4. Select **Customize** from the toolbar.
5. Select **Design Web View**.

   The Visual Designer is displayed.
6. Select a layout section and then select to add a tile.

   The **Add tile** window is displayed.
7. Select **Artificial Intelligence** and the tile you want to add.
8. Configure the tile by adding titles, additional parameters and styling.
9. Select **Add tile**.

   You will return to the program where the tile is displayed.

> You can only link an active, trained machine learning project to a tile with KPIs.
>
> If you haven't activated the project in the **SYSPRO Machine Learning** program, you won't be able to add KPIs against the project.

Available training algorithms per project type

This lists the training algorithms that can be used per project type.

Both Regression & Classification models use the TPOT (Tree-based Pipeline Optimization Tool) library. This tool acts as a data science assistant, as it automatically tries multiple algorithms and chooses the model with the best score, based on a user-selected scoring function. You can read more about the algorithms and techniques, such as *K-fold cross-validation*, that are used by TPOT to create a good model.

Classification projects

The majority of the functionality within TPOT is derived from the features available in *Sklearn*, which is the open sourced Python machine learning library.

The following scoring functions are supported in classification projects:

- accuracy
- adjusted_rand_score
- average_precision
- balanced_accuracy
- f1
- f1_macro
- f1_micro
- f1_samples
- f1_weighted
- neg_log_loss
- precision
- precision_macro
- precision_micro
- precision_samples
- precision_weighted
- recall
- recall_macro
- recall_micro
- recall_samples
- recall_weighted
- roc_auc

Regression projects

The majority of the functionality within TPOT is derived from the features available in *Sklearn*, which is the open sourced Python machine learning library.

The following scoring functions are supported in regression projects:

- neg_mean_squared_error
- neg_mean_absolute_error
- neg_mean_absolute_error
- r2

Anomaly projects

Anomaly detection projects use the *PyOD* library.

The following estimator algorithms are supported in anomaly projects:

- abod
- auto_encoder
- cblof
- combination
- feature_bagging
- hbos
- iforest
- knn
- lof
- loci
- lcsp
- mcd
- mo_gaal
- ocsvm
- pca
- so_gaal
- sos
- xbod

Clustering projects

Clustering detection projects use the *PyClustering* library.

The following estimator algorithms are supported in clustering projects:

- auto
- agglomerative
- bang
- birch
- bsas
- clarans
- clique
- cure
- dbscan
- elbow
- ema
- fcm
- kmeans
- hsyncnet
- kmedians
- kmedoids
- mbsas
- optics
- rock
- somsc
- syncsom
- xmeans

Available scoring methods per project type

This lists the scoring methods that can be used per project type.

There are no scoring methods for anomaly type projects. Anomaly detectors can't be scored as they are unsupervised machine learning algorithms and don't use analogous scoring metrics.

Classification projects

The following scoring metrics can be used in classification projects:

- accuracy
- adjusted_rand_score
- average_precision
- balanced_accuracy
- f1
- f1_macro
- f1_micro
- f1_samples
- f1_weighted
- neg_log_loss
- precision
- precision_macro
- precision_micro
- precision_samples
- precision_weighted
- recall
- recall_macro
- recall_micro
- recall_samples
- recall_weighted
- roc_auc

Regression projects

The following scoring metrics can be used in regression projects:

- neg_mean_squared_error
- neg_mean_absolute_error
- neg_mean_absolute_error
- r2

Status Codes

AI and Machine Learning Statuses

The status of a machine learning project indicates the current stage of the project and how it can be used.

Status	Description
`0 - IN DEVELOPMENT`	The machine learning project is new or currently in development and has not been trained yet.
`5 - READY FOR USE`	The machine learning project was trained successfully and a model was produced. The project hasn't been activated, so the project model can't be used for predictions yet.
`1 - TRAINING`	The machine learning project is currently being trained by the AI engine.
`10 - ACTIVE`	The machine learning project is active and the model for this project can be used for predictions.
`99 - ERROR`	The AI engine ran into an error while training the project. The detailed error can be viewed from the training queue.

Hints and Tips

- The KPI definitions for machine learning projects must be created in SYSPRO 8 using the **Insight Tile Definition** program.
- Only **Regression** type machine learning projects can be linked to a tile with KPIs, as values are compared to thresholds to determine the outcome.

Referencing

Menu and Toolbar

Field	Description
AI projects in training queue	This opens the **Training Queue** pane and lists all projects that are currently queued for training.
Models available for predictions	This opens the **List of Models** pane where all trained models are listed that can be used for predictions.
Import project	This lets you import a project that was previously downloaded. The **Import Project** window is displayed.
New machine learning project	This lets you add a new machine learning project.
Save project	This lets you save the new, or changes made to the existing machine learning project.
Delete project	This lets you delete the existing machine learning project.

Project information

Field	Description
Model	This indicates the project model. *Classification* This model predicts a condition (e.g. good or bad) and the **Output column** is usually a text field. *Regression* This model predicts the quantity or an amount (e.g. days late). *Anomaly* This model predicts a data irregularity or inconsistency.
Name	This displays the project name.
Description	This displays the project description.
Data source	This displays the data source.

Field	Description
Select statement	This displays the select statement (usually a SQL query) that extracts the required data from the database.
Output column	This is populated using the **Generate output columns** button on the **Select statement** field. Once this field is populated, you can select the column that contains the primary data you want to query and use for prediction.  For classification models, this is usually a text field. For regression models, this is always a numeric field.
Train Project	This lets you train the project, i.e. the project examines and learns the data.  Predictions can only be made on trained projects.
Copy to new version	This lets you create a new version of an existing project on **system**, **company** or **role** level.  You typically copy a shipped project listed in the **SYSPRO** category to use that project as a basis to work with. You can also copy a project located in the **System** or **Company** category to create a new version of an existing project.
Advanced options	This displays the **Advanced option** pane, where you can select additional training and toolkit options.
Activate project	This lets you activate a project.  SYSPRO projects that are shipped with Avanti must be copied to a new version, before they can be activated. Only active projects can be used for predictions.
Deactivate project	This lets you deactivate a project that is no longer used for predictions.

Field	Description
Download project	This lets you download the selected project, so that you can later import and train it. Once downloaded, you can find the `.proj` file in your Windows **Downloads** folder.

Model information

This displays the basic statistics about the selected trained project.

Some objective metric is required to compare and identify the better model. The ML Engine supports various metrics that are defined in the project using the **Scoring method**.

The data set is split into a training set and a testing set. The ML Engine learns from the training set and isn't allowed to see the testing set. Once a model is created, its performance on unseen data can be measured by passing the testing set into the model and comparing predicted values in the training set. This process of comparing predicted values to actual values can be used to objectively score the model. The displayed scores are the result of this comparison.

Field	Description
Scoring method	This indicates the method used to evaluate the quality of the given pipeline for the classification problem.
Total set score	This indicates how the model scored over the entire data set i.e. both the training and testing data sets combined.
Possible error	This indicates how the model performed on the testing data set only, and is therefore a good indication of how well it will predict the right outcome on unseen data.
Test observations	This indicates the total number of records used for the selected prediction results.
Training observations	This indicates the total number of records returned from the SQL statement.
Perform prediction	This lets you perform a prediction on the trained model.  You can only perform predictions on trained models in active projects.

Feature importance

This displays the **Feature importance** graph containing a score for each SQL column in the AI project - indicating which data most affects the predicted value.

The score percentage against each column indicates the frequency according to which the data was used for predictions and therefore the significance of the data in the project. This information allows you to evaluate the data used in the project and delete rarely-used columns.

The **Feature importance** graph and a link to the corresponding data is also displayed when selecting an AI tile in ***SYSPRO Web UI (Avanti)***.

You must retrain existing projects to recreate the compact model and include the feature importance.

New Project

This window is displayed when you select the **New machine learning project** button on the toolbar.

Field	Description
Save project	This lets you save the project.
Project	Select the type of project model you want to create. *Classification* Select this if you want the prediction result to be a condition (e.g. good or bad). The **Output column** is usually a text field. Accuracy (the closeness of a measured value to a standard or known value) is used to determine the condition. *Regression* Select this if you want the prediction result to be a quantity or an amount (e.g. days late). The **Output column** is usually a numeric field. Mean squared error (average squared difference between the estimated values and what is estimated) is used to determine the regression. *Anomaly* Select this model if you want to predict an irregularity or inconsistency within the data source. There are no scoring methods for anomaly type projects. Anomaly detectors can't be scored as they are unsupervised machine learning algorithms and don't use analogous scoring metrics.
Name	Indicate the project name.
Description	Enter a description for the project.
Data source	Select the data source from the list. Data sources are automatically created for every SYSPRO company in your database. You can create your own data sources for data that lies outside of SYSPRO.

Field	Description
Status	This indicates the status of the selected project.
Select statement	Enter the select statement (usually a SQL query) that will extract the required data from the database. Select **Generate output columns** to populate the column names in the **Output column** field.
Output column	Once this field is populated, you can select the column that contains the primary data you want to query and use for prediction. For classification models, this is usually a text field. For regression models, this is always a numeric field.

Advanced Options

The advanced machine learning options are intended for data scientists and allow an increased training time for projects with a lot of data.

Field	Description
Training options	This applies to the entire machine learning project.
Random state	This ensures that TPOT (Tree-based Pipeline Optimization Tool) gives you the same results each time you run it against the same data set using that seed. The data type is an integer and can be set to null.
Scoring method	This function is used to evaluate the quality of the given pipeline for the classification problem. The available regression and classification functions are listed in the Using section.
Test ratio	This indicates the percentage of the data set that is used to test the trained model that is generated. *For example:* 0.2 means that 20% of the data that is returned from the select statement will be used for testing.

Field	Description
Toolkit setup	
Generations	This indicates the number of iterations required to run the pipe line optimization process. This defaults to 100 and must be a positive number. The more generations, the better TPOT works, as it evaluates the population size + generation * offspring_size pipelines in total.
Population size	This indicates the number of individuals to retain in the generic programming population of every generation. This defaults to 100, but the greater the number, the better the prediction outcome.
Offspring	This indicates the number of offspring to produce in each genetic programming generation. This has a default value of 100, and must be a positive number.
Mutation rate	The mutation rate for the genetic programming algorithm indicates how many pipelines apply random changes to every generation.
Cross over rate	This indicates how many pipelines to breed for every generation. The range is [0.0, 1.0], with 0.1 being the default value. The mutation_rate + crossover_rate must not exceed 1.0.
Number of folds to evaluate	This indicates the folds that evaluate each pipeline in k-fold cross validation during the optimization process. This is the cv number and must be an integer. The default value is 5.
Sub sample	This indicates the fraction of training samples that are used during the optimization process. It must be in the range [0.0, 1.0], with 1.0 being the default value.
Max time	This indicates the time the machine learning engine spends on optimizing the pipeline. This must be a positive integer.

Field	Description
Max evaluation time	This indicates the time the machine learning engine spends on evaluating a single pipeline. This must be a positive integer. Setting this to a higher number allows the machine learning engine to consider pipelines that are more complex, and increases the time it takes to train the project.
Configuration dictionary	This lets you customize the operators and parameters that TPOT searches in the optimization process.
Early stop	This indicates the number of generations TPOT uses to check if there are improvements in the optimization process. If no improvements are found, the optimization process ends.

Save For

This pane is displayed when you select a shipped project (in the SYSPRO folder) and then select the **Copy to a new version** button.

This lets you duplicate the project to create various versions, while keeping the original project intact. You can then edit the project versions by changing the SQL statement and/or data source.

Field	Description
Save for	This lets you specify the level of the project, which can be: ▪ System ▪ Company ▪ Role
Company	This lets you select the company for which the project will be saved.
Role	This lets you select the role for which the project will be saved.
Data source	This lets you select the data source that will be used for the project.

Import project

The window is displayed when you select the **Import project** button.

Field	Description
Import path	This lets you navigate to the path where the project was downloaded. Once you have selected the `.proj` file, the remaining fields are auto-populated.
Save for	This lets you select the level of the project.
Company	This indicates the company that was specified when the project was created, if the project was created at company level.
Role	This indicates the role that was specified when the project was created, if the project was created at role level.
Data source	This indicates the data source that is used for the project.
ProjectVersion	This indicates the version of the project.
Train project	Select this to add the project to the **AI Training Queue** as soon as it is imported.
Model	This indicates the type of project model for this project.
Name	This indicates the name of the project and can be edited.
Description	This indicates the description of the project and can be edited.
Select statement	Enter the select statement (usually a SQL query) that will extract the required data from the database. Select **Generate output columns** to populate the column names in the **Output column** field.
Output column	Once this field is populated, you can select the column that contains the primary data you want to query and use for prediction.  For classification models, this is usually a text field. For regression models, this is always a numeric field.

New data source

This pane is displayed when you select **New Data source** from the **List of Data Sources** window.

Data sources you have created, are saved in the SaiDataSource table in the system-wide database.

Field	Description
Name	Indicate a name for the data source.
Description	Indicate a description for the data source.
Type	This defaults to ODBC.
Connection string	Enter the connection string. This must be in the following format: `DRIVER={SQL Server};SERVER={SERVERNAME};DATABASE={Database Name};UID={user};PWD={Password}` Select **Test connection** to confirm that the connection is working.

Sample project details

Chance order will be late

Field	Description
Description	This displays a percentage that predicts the chance a purchase order line may be late. This is based on all orders placed in the past as well as current information about the supplier.
Model type	Regression
Output column	`IsLate`
SQL query	

```
select PTL.IsLate ,
PMH.OrderStatus,
PMH.NextDetailLine,
PMH.ExchangeRate,
PMH.TermsCode,
APS.CurrentBalance,
APS.SupplierClass,
APS.PyrEndBalance,
APS.YtdAmount1099,
APS.NumMonthsZero,
APS.LastChequeVal
from
(Select top (1000) PMH.PurchaseOrder,
case
when sum(datediff(day, MLatestDueDate,
MLastReceiptDat)) > 0
then 1 else 0 End
as IsLate
from PorMasterDetail as PMD inner join PorMasterHdr as
PMH
on(PMD.PurchaseOrder=PMH.PurchaseOrder)
where (PMD.MLatestDueDate is not
null)and(PMD.MLastReceiptDat is not null)
group by PMH.PurchaseOrder)
PTL
join PorMasterHdr PMH on PMH.PurchaseOrder =
PTL.PurchaseOrder
join ApSupplier APS on APS.Supplier = PMH.Supplier
```

LCT days late

Field	Description
Description	This predicts how late a shipment may be, measured in days.
Model type	Regression
Output column	`DaysLate`

Field	Description

SQL query	<pre>select datediff(day,LH.EstArrivalDate, LH.ActArrivalDate)
	as DaysLate,
	case
	when LTRIM(RTRIM(LH.PlaceOfShipment)) = '' then
	null
	else lower(LH.PlaceOfShipment) end as
	PlaceOfShipment,
	case
	when LTRIM(RTRIM(LH.PlaceOfDestination)) = ''
	then null
	else lower(LH.PlaceOfDestination) end as
	PlaceOfDestination,
	LH.Route,
	LR.DaysBeforeEta,
	LR.DemurrageDays,
	LR.DaysAfterEta
	from LctShipmentHdr LH
	left join LctRoute LR on LH.Route = LR.Route
	where LH.ActArrivalDate is not null</pre>

PO days late

Field	Description
Description	This predicts the number of days a purchase order line may be late.
Model type	Regression
Output column	`DaysLate`

SQL query	<pre>select PMH.Supplier,
	PMD.MStockCode,
	PMD.MWarehouse,
	PMD.MOrderQty,
	PMD.MReceivedQty,
	PMD.MOrigDueDate,
	PMD.MLatestDueDate,
	PMD.MPrice,
	PMD.MProductClass,
	datediff(day, MLatestDueDate, MLastReceiptDat) as DaysLate
	from PorMasterDetail as PMD inner
	join PorMasterHdr
	as PMH on(PMD.PurchaseOrder=PMH.PurchaseOrder)
	where
	(PMD.MLatestDueDate is not null)and(PMD.MLastReceiptDat
	is not null)</pre>

Customer invoice pay date

Field	Description
Description	This predicts the number of days a customer may take to pay an invoice.
Model type	Regression
Output column	`DaysToPayEstimate`

Field	Description
SQL query	``` ;With TheInvoicePayments as (select IP.Customer, IP.Invoice, max(IP.JournalDate)as MaxJournalDate from ArInvoicePay IP with (nolock) group by IP.Invoice, IP.Customer) select I.Customer, I.Invoice, I.InvoiceDate, I.OrigDiscValue, I.InvoiceBal1, I.InvoiceBal2, I.InvoiceBal3, I.CurrencyValue, C.Area, C.ApplyLineDisc, C.ApplyOrdDisc, C.BalanceType, C.Contact, C.CreditLimit, C.CreditStatus, C.Currency, C.DateLastPay, C.DateLastSale, C.HighestBalance, C.HighInvDays, C.Name, C.MinimumOrderValue, C.PaymentsAllowed, C.PriceCode, C.ShippingInstrs, C.Salesperson, C.Salesperson1, C.Salesperson2, C.Salesperson3, C.CustomerOnHold, case when IP.MaxJournalDate is null then Datediff(day, I.InvoiceDate, Dateadd(day, 90, I.InvoiceDate)) when I.InvoiceBal1 > 0 OR I.InvoiceBal2 > 0 OR I.InvoiceBal3 > 0 then Datediff(day, I.InvoiceDate, Dateadd(day, 60, I.InvoiceDate)) else Datediff(day, I.InvoiceDate, IP.MaxJournalDate) end as DaysToPayEstimate From ArInvoice I with (nolock) left join TheInvoicePayments IP with (nolock) on (I.Customer = IP.Customer and I.Invoice = IP.Invoice) left join ArCustomer C with (nolock) on (I.Customer = C.Customer) where I.DocumentType = 'I' ```

Job status

Field	Description
Description	This predicts whether a job may be completed early or late. The outcome of the prediction displays **On time** or **Late**.
Model type	Classification

Field	Description
Output column	`JobStatus`

SQL query	`select top 1000`
	`JobDescription,`
	`JobClassification,`
	`StockCode,`
	`Warehouse,`
	`JobTenderDate,`
	`JobDeliveryDate,`
	`JobStartDate,`
	`OrigDueDate,`
	`case when (ActCompleteDate>OrigDueDate) then 'Late' else`
	`'On time'`
	`end as JobStatus`
	`from [WipMaster]`
	`where (OrigDueDate is not null)`

Lost sales reason

Field	Description
Description	This prediction determines whether sales may be lost and what the reason for losses would be.
Model type	Classification
Output column	`ReasonLost`

SQL query	`select CostValue,`
	`QuantityLost,`
	`QuantityRequested,`
	`QuantityUom,`
	`Price,`
	`PriceUom,`
	`RC.Description as ReasonLost,`
	`IW.QtyAllocated,`
	`IW.QtyOnHand,`
	`IW.QtyOnOrder,`
	`IW.QtyAllocatedWip,`
	`IW.QtyDispatched,`
	`IW.MinimumQty,`
	`IW.MaximumQty,`
	`IW.SafetyStockQty,`
	`IW.ReOrderQty,`
	`CM.CreditLimit,`
	`CM.HighestBalance,`
	`CM.OutstOrdVal,`
	`CM.NumOutstOrd,`
	`CM.ApplyOrdDisc,`
	`CM.BackOrdReqd`
	`from [dbo].[SorLostSales] LS`
	`join [dbo].TblSoReason RC on RC.ReasonCode =`
	`LS.ReasonCode`
	`join [dbo].InvWarehouse IW on IW.StockCode =`
	`LS.StockCode and IW.Warehouse = LS.Warehouse`
	`join [dbo].ArCustomer CM on CM.Customer = LS.Customer`

Problem with sales order

Field	Description
Description	This predicts the most likely reason why a sales order is returned. This is based on the reasons past sales order items were returned.
Model type	Classification
Output column	`ProblemDesc`
SQL query	

```
select
RP.ProblemDesc,
SD.MOrderQty,
SD.MWarehouse,
SD.MBin,
SD.MPrice,
IM.Supplier,
M.Mass,
IM.Volume
from [dbo].[RmaDetail] RD
join SorDetail SD on SD.SalesOrder = RD.SalesOrder and
SD.SalesOrderLine = SD.SalesOrderLine
join RmaProblemCodes RP on RP.ProblemCode = RD.ProblemCode
join InvMaster IM on SD.MStockCode = IM.StockCode
```

Quote success

Field	Description
Description	This predicts how many quotes should be accepted by the customer and result in sales orders. The model returns either `SALE` or `NOSALE` based on historical quotations and the subsequent conversion into sales orders.
Model type	Classification
Output column	`QuoteStatus`

Field	Description
SQL query	

```
declare @startDate datetime = dateadd(year, -2,
current_timestamp);
declare @endDate datetime = current_timestamp;
with _quoteMaster as
(
select
QM.Quote,
QM.DateEnquiry,
QM.DateDeliveryReq
from QotMaster as QM
where (QM.DateEnquiry between @startDate and @endDate)
)
, _quoteOffer as
(
select
Quote,
QuoteVersion,
Line,
MQuantity,
MPrice,
MDiscountValue,
MNetValue,
MCost,
MLeadTime
from QotOffer
)
, _quoteDetail as(
select
Quote,
QuoteVersion,
Line,
ProductClass,
MStockCode
from QotDetail),
 _quoteSalesLines as(
select
QO.Quote,
QO.QuoteVersion,
QO.Line,
QO.MQuantity,
QO.MPrice,
QO.MDiscountValue,
QO.MNetValue,
QO.MCost,
QO.MLeadTime,
QD.ProductClass,
QD.MStockCode
from _quoteOffer as QO
join _quoteDetail as QD
on((QO.Quote=QD.Quote)and(QO.QuoteVersion=QD.QuoteVersion)and(QO.Line=QD.Line))),
 _quoteSalesLinesSummary as(
select
QSL.Quote,
QSL.QuoteVersion,
count(QSL.Line) as TotalLines,
sum(QSL.MQuantity) as MQuantity,
sum(QSL.MPrice) as MPrice,
sum(QSL.MDiscountValue) as MDiscountValue,
sum(QSL.MNetValue) as MNetValue,
sum(QSL.MCost) as MCost,
max(QSL.MLeadTime) as MLeadTime,
count(distinct QSL.ProductClass) as UniqueProductClasses,
count(distinct QSL.MStockCode) as UniqueStockCodes
from _quoteSalesLines as QSL
group by QSL.Quote, QSL.QuoteVersion),
_quoteSales as(
select
SalesOrder,
Quote,
QuoteVersion,
Customer,
Salesperson,
ReqShipDate,
Area,
```

Field	Description

	``
	CashCredit,
	case when ltrim(CompanyTaxNo)='' then 'NON-VENDOR' else'VENDOR'
	end as TaxStatus
	from QotSoMasterHdrwhere (ReqShipDate between @startDate and @endDate))
	select top (50000)
	QS.SalesOrder,
	QS.Quote,
	QS.QuoteVersion,
	QS.Customer,
	QS.Salesperson,
	QS.ReqShipDate,
	QS.Area,
	QS.CashCredit,
	QS.TaxStatus,
	QSLS.TotalLines,
	QSLS.MQuantity,
	QSLS.MPrice,
	QSLS.MDiscountValue,
	QSLS.MNetValue,
	QSLS.MCost,
	QSLS.MLeadTime,
	case when (QSLS.Quote is null) then 'NOSALE' else 'SALE'
	end as QuoteStatus
	from _quoteSales as QS
	left join _quoteSalesLinesSummary as QSLS
	on((QS.Quote=QSLS.Quote)and(QS.QuoteVersion=QSLS.QuoteVersion));
	``

Stock code profitability

Field	Description
Description	This predicts how profitable a stock code should be by location. The stock code will be grouped in a bad, medium, good or great category, which is based on the profitability and returns of similar stock codes. The categories are hard coded as follows: ▪ Below 0. ▪ Between 0 and 10000. ▪ Between 10000 and 100000. ▪ Above 100000.
Model type	Classification
Output column	`ProfitCategory`

Field	Description
SQL query	

```
with _profitByStockCode as
(
select
IM.StockCode,
sum(SD.MOrderQty*(SD.MPrice-SD.MUnitCost))
as TotalProfit
from SorDetail SD
join InvMaster IM on IM.StockCode = SD.MStockCode
where SD.MStockCode is not null
group by IM.StockCode
)
,
profitStats as
(
select
avg(TotalProfit) as Avg_TotalProfit,
min(TotalProfit) as Min_TotalProfit,
max(TotalProfit) as Max_TotalProfit
from _profitByStockCode as SCP
)
,
detailedData as
(
select
SCP.StockCode,
IM.AlternateUom,
IM.OtherUom,
IM.ConvMulDiv,
IM.ConvFactAltUom,
IM.Mass,
IM.Volume,
IM.Supplier,
IM.ProductClass,
IM.Buyer,
IM.LeadTime,
cast(SCP.TotalProfit as decimal(18,2)) as
TotalProfit
from _profitByStockCode as SCP
join InvMaster as IM
on IM.StockCode = SCP.StockCode
)
select
StockCode,
AlternateUom,
OtherUom,
ConvMulDiv,
ConvFactAltUom,
Mass,
Volume,
Supplier,
ProductClass,
Buyer,
LeadTime,
--TotalProfit,
case
when TotalProfit <0 then 'BAD'
when TotalProfit between 0 and 10000 then 'MEDIUM'
when TotalProfit between 10000 and 100000 then
'GOOD'
else 'GREAT'
end as ProfitCategory
from detailedData as DD
```

Supplier performance

Field	Description
Description	This predicts how well a supplier should perform in terms of deliveries, and indicates the likelihood of the delivery being on time and in full. The possible prediction outcomes are as follows: ▪ Delivers late, in full. ▪ Delivers late, low stock. ▪ Delivers late, with extra. ▪ Delivers on time, in full. ▪ Delivers on time, low stock. ▪ Delivers on time, with extra stock. ▪ Delivers early, in full. ▪ Delivers early, low stock. ▪ Delivers early, with extra.
Model type	Classification
Output column	`SupplierRanking`

Field	Description
SQL query	select PMH.Supplier, PMD.MStockCode as StockCode, PMD.MStockDes as [Description], PMD.MWarehouse as Warehouse, PMD.MOrderQty as OrderQty, PMD.MReceivedQty as ReceivedQty, PMD.MLatestDueDate as LatestDueDate, PMD.MLastReceiptDat as LastReceiptDat, PMD.MProductClass as ProductClass, PMD.MCompleteFlag as CompleteFlag, PMD.MOrigDueDate as OrigDueDate, PMD.MReschedDueDate as ReschedDueDate, PMH.ShippingLocation, PMH.[State] as [State], PMH.ShippingInstrs, case when MLastReceiptDat > MLatestDueDate and MReceivedQty = MOrderQty then 'Delivers late, in full' when MLastReceiptDat > MLatestDueDate and MReceivedQty < MOrderQty then 'Delivers late, low stock' when MLastReceiptDat > MLatestDueDate and MReceivedQty > MOrderQty then 'Delivers late, with extra' when MLastReceiptDat = MLatestDueDate and MReceivedQty = MOrderQty then 'Delivers on time, In full' when MLastReceiptDat = MLatestDueDate and MReceivedQty < MOrderQty then 'Delivers on time, low stock' when MLastReceiptDat = MLatestDueDate and MReceivedQty > MOrderQty then 'Delivers on time, With extra stock' when MLastReceiptDat < MLatestDueDate and MReceivedQty = MOrderQty then 'Delivers early, in full' when MLastReceiptDat < MLatestDueDate and MReceivedQty < MOrderQty then 'Delivers early, low stock' when MLastReceiptDat < MLatestDueDate and MReceivedQty > MOrderQty then 'Delivers early, with extra' end as SupplierRanking From PorMasterDetail PMD join PorMasterHdr PMH on PMD.PurchaseOrder = PMH.PurchaseOrder where MStockCode <> ' '

AP invoice payment

Field	Description
Description	This predicts by when a supplier should pay an invoice. The outcome of the prediction displays **On time** or **Late**.
Model type	Classification
Output column	`Status`

Field	Description

SQL query	<pre>with _invoiceMaster as
	(
	select
	Supplier,
	Invoice,
	InvoiceDate,
	DueDate,
	OrigInvValue,
	Currency,
	CurrencyValue,
	from ApInvoice
	)
	,
	_invoicePaid as
	(
	select
	Supplier,
	Invoice,
	EntryNumber,
	TrnType,
	JournalDate,
	TrnValue,
	TrnYear,
	TrnMonth,
	PostValue,
	PostCurrency
	from ApInvoicePay
	where TrnType='P'
	)
	select top(500)
	IM.Supplier,
	IM.InvoiceDate,
	IM.DueDate,
	IM.OrigInvValue,
	IM.Currency
	,case when (IP.JournalDate>IM.DueDate)
	then 'Late'
	else
	'On time' end as Status
	from _invoiceMaster as IM
	join _invoicePaid as IP
	on(
	(IM.Supplier = IP.Supplier)
	and(IM.Invoice = IP.Invoice)
	)</pre>

Customer classification

Field	Description
Description	This predicts how profitable a customer would be. The customer will be grouped in a bad, medium, good or great category, which is based on the profitability and returns of similar customers. The categories are calculated using the mean profitability of all customers and the maximum profitability.
Model type	Classification
Output column	`ProfitCategory`

Field	Description

SQL query	<pre>with _profitByCustomer as
	(select
	SM.Customer,
	sum(SD.MOrderQty*(SD.MPrice-SD.MUnitCost))as TotalProfit
	from SorDetail SD
	join SorMaster SM on SM.SalesOrder = SD.SalesOrder
	where SD.MStockCode is not null
	group by SM.Customer),
	profitStats as(
	select
	avg(TotalProfit)*0.2 as Avg_TotalProfit,
	min(TotalProfit) as Min_TotalProfit,
	max(TotalProfit) as Max_TotalProfitfrom _profitByCustomer as SCP),
	detailedData as(
	select
	AC.Customer,
	AC.Name,
	AC.CreditLimit,
	AC.Salesperson,
	AC.Area,
	AC.Branch,
	AC.TermsCode,
	AC.OutstOrdVal,
	AC.NumOutstOrd,
	AC.HighestBalance,
	AC.SoldPostalCode,
	AC.ShipPostalCode,
	AC.RouteCode,
	AC.StoreNumber,
	AC.MasterAccount,
	cast(SCP.TotalProfit as decimal(18,2)) asTotalProfitfrom _profitByCustomer as SCP
	join ArCustomer
	as ACon AC.Customer = SCP.Customer
	)
	select
	CreditLimit,
	Salesperson,
	Area,
	Branch,
	TermsCode,
	SoldPostalCode,
	ShipPostalCode,
	RouteCode,
	StoreNumber,
	MasterAccount,
	TotalProfit,
	case
	when TotalProfit <0 then 'bad'
	when TotalProfit between 0 and Avg_TotalProfit then 'medium'
	when TotalProfit between Avg_TotalProfit andMax_TotalProfit-Avg_TotalProfit then 'good'
	else 'great'
	end as ProfitCategory
	from detailedData as DD
	join profitStats on 1=1
	order by TotalProfit</pre>

Sales order line anomaly

Field	Description
Description	This detects anomalies on sales order lines based on the selected columns.
Model type	Anomaly
Output column	The **Output column** doesn't apply to anomaly projects.

Field	Description

SQL query	<pre>select
	MStockCode,
	MWarehouse,
	MOrderQty,
	MOrderUom,
	MPrice,
	MPriceUom,
	MCustRequestDat
	from SorDetail
	where
	(MStockCode is not null)
	and (MWarehouse is not null)
	and (MOrderQty is not null)
	and (MOrderUom is not null)
	and (MPrice is not null)
	and (MPriceUom is not null)
	and (MCustRequestDat is not null)</pre>

Purchase order line anomaly

Field	Description
Description	This detects anomalies on purchase order lines based on the selected columns.
Model type	Anomaly
Output column	The **Output column** doesn't apply to anomaly projects.

SQL query	<pre>select
	MStockCode,
	MWarehouse,
	MOrderQty,
	MOrderUom,
	MPrice,
	MPriceUom,
	MLatestDueDate
	from PorMasterDetail
	where
	(MStockCode is not null)
	and (MWarehouse is not null)
	and (MOrderQty is not null)
	and (MOrderUom is not null)
	and (MPrice is not null)
	and (MPriceUom is not null)
	and (MLatestDueDate is not null)</pre>

SYSPRO™

www.syspro.com