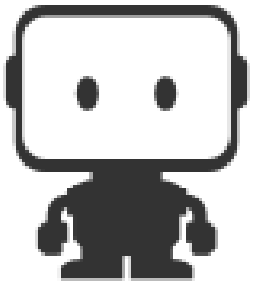


DataRobot

User Guide

Contents

Introduction

Overview

The DataRobot platform enables data scientists to build highly accurate predictive models, and to build them orders of magnitude faster than using traditional methods.

Building an accurate predictive model generally requires evaluating and selecting numerous data transformations, features, algorithms, and tuning parameters. DataRobot simplifies the model development process by performing a parallel heuristic search for the best model or combination of models for the applicable dataset and prediction target.

By cost-effectively evaluating thousands of models in parallel across a large cluster of servers, DataRobot delivers the best predictive model in the shortest amount of time.

About this Guide

Title: DataRobot User Guide

Version: 2.0

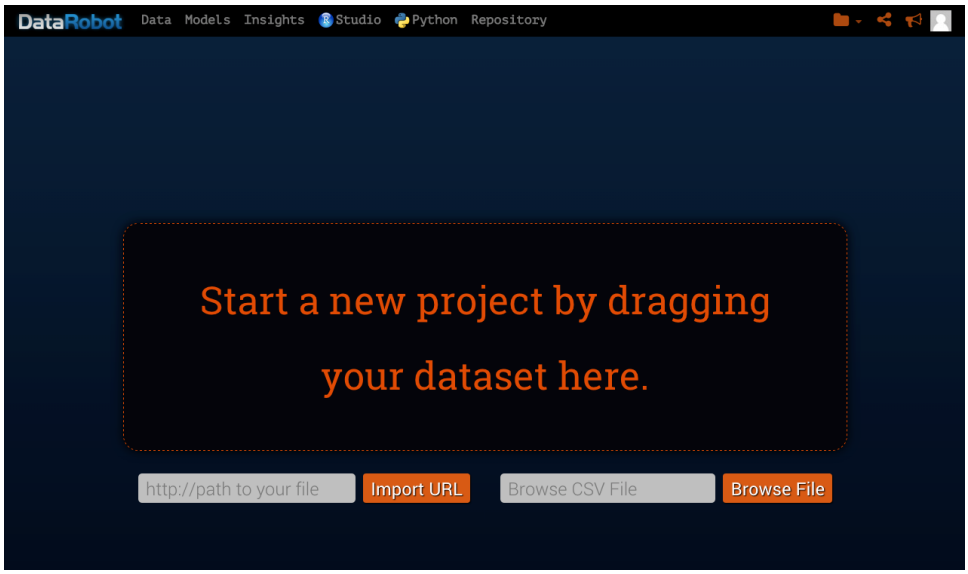
Publication date: 7-27-15

QuickStart

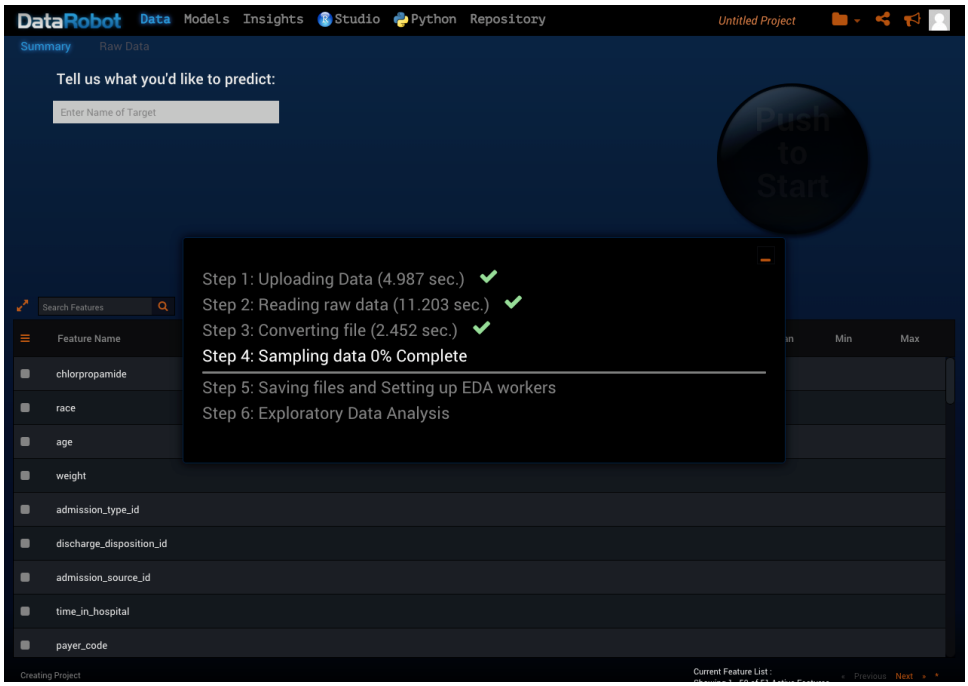
In order to build models in DataRobot, you must create a project with a dataset. After you create a project, you can select a target feature and start the modeling process. A DataRobot project contains all of the models you build for the project dataset.

Use the following steps to begin modeling data with the DataRobot API.

1. You can use any of the following methods to create a new DataRobot project:
   - Drag a .csv data file to the orange box on the DataRobot platform home page.
   - Click **Browse File**, then use the file browser to select a .csv data file.
   - Type a URL in the box labeled "http://path to your file", then click **Import URL**.



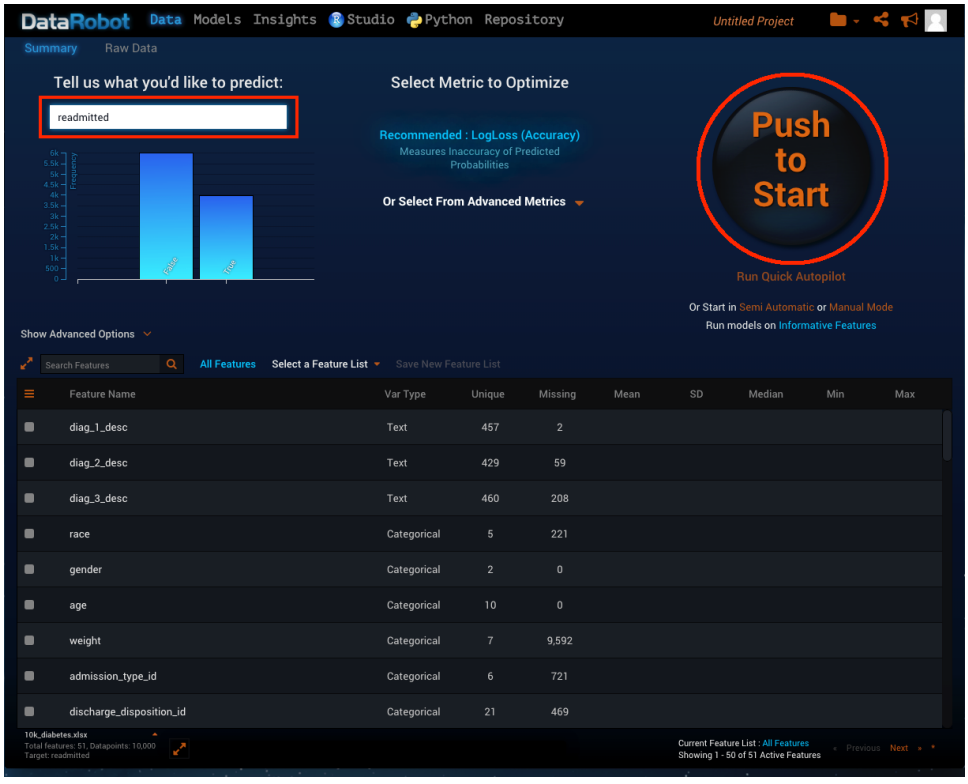
2. A progress indicator is displayed while the file is being processed.



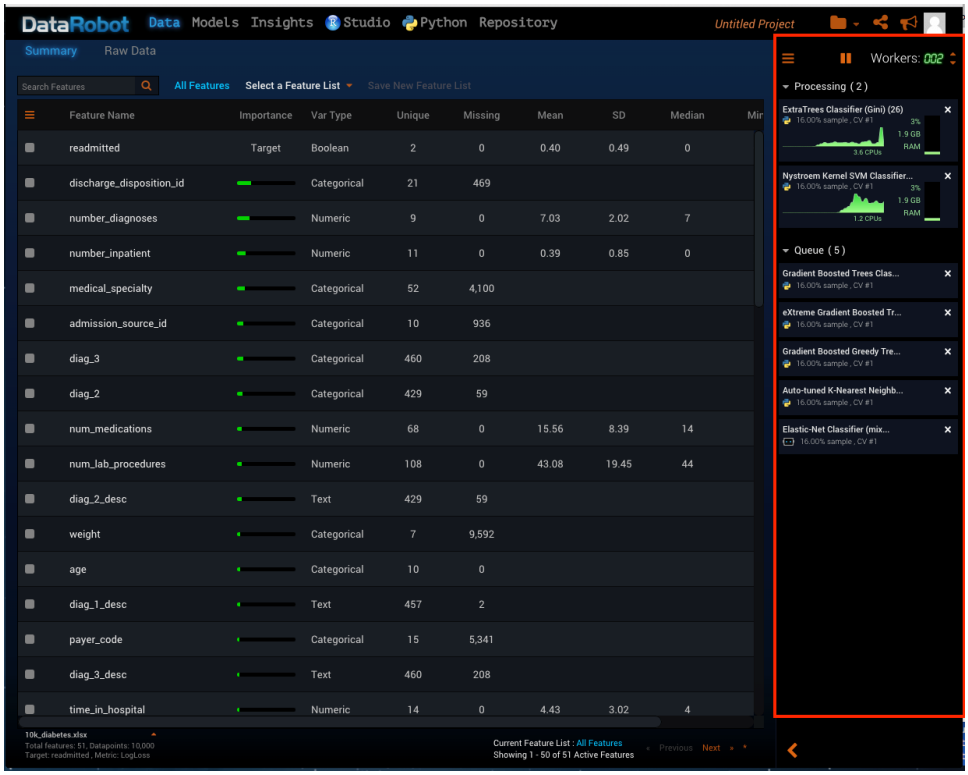
3. To begin modeling, type in the name of the target feature. The target feature is the name of the column in the dataset you would like to predict.

   DataRobot provides a preselected optimization metric based on the modeling task represented by the target feature (regression or classification). To select a different optimization metric, click **Or Select from Advanced Metrics**.

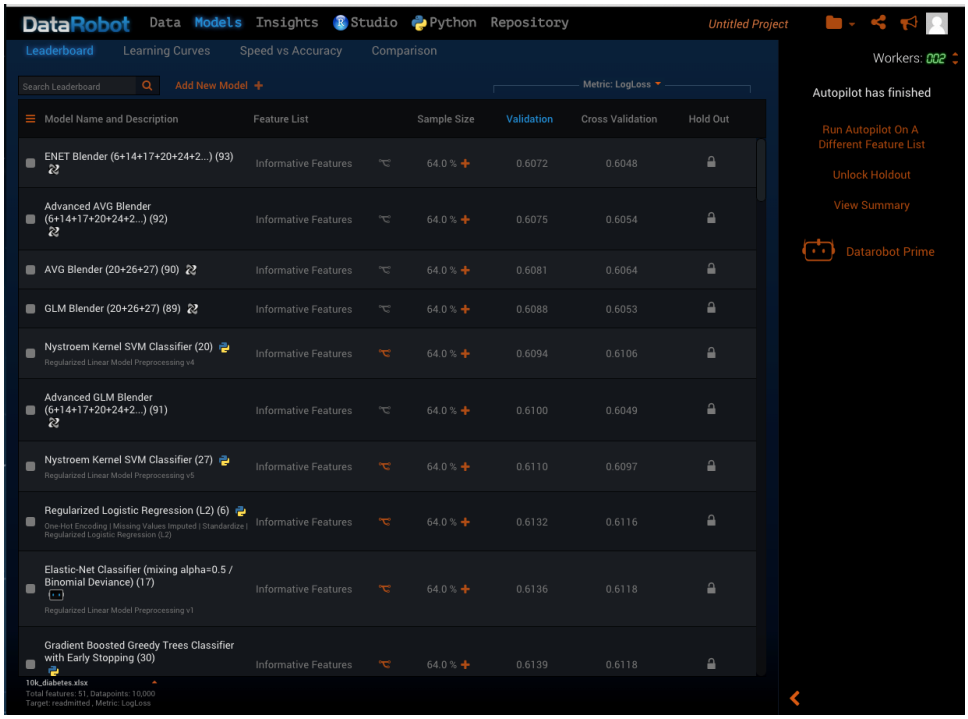
   After specifying the target feature, click **Push to Start** to start modeling. The modeling process will run in Automatic mode by default.



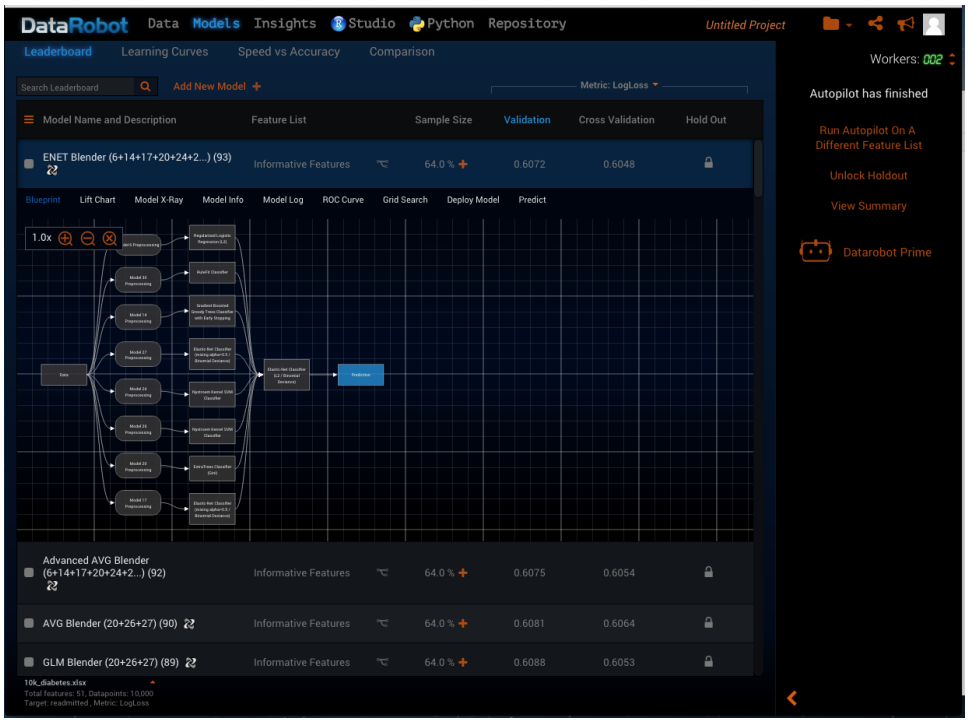
4. The status of running models is displayed in the Worker Queue at the right of the screen. Depending on the size of the dataset, it may take several minutes to complete the modeling process.



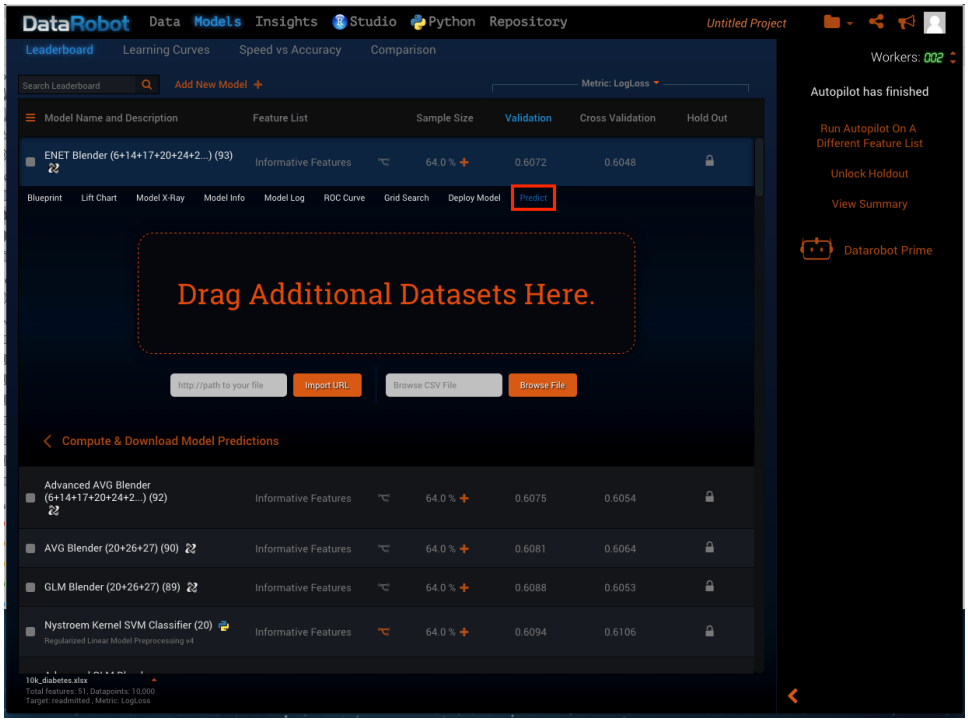
5. The results of the modeling process are displayed in the model Leaderboard, with the best models (based on the chosen performance metric) at the top of the list.



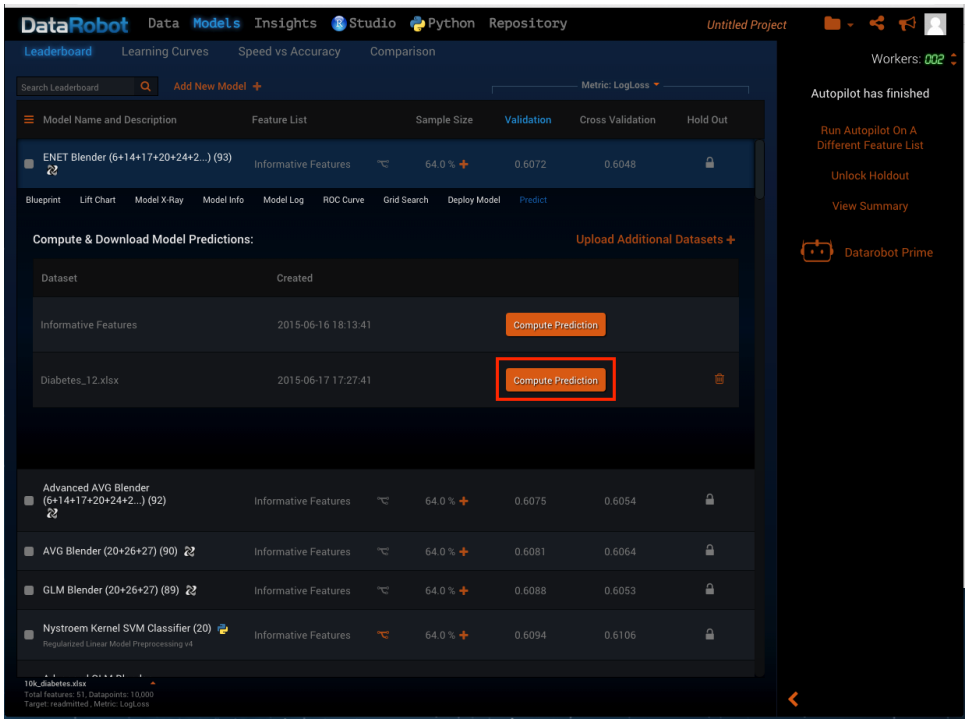
6. Click a model to display the model blueprint, lift chart, and other information.



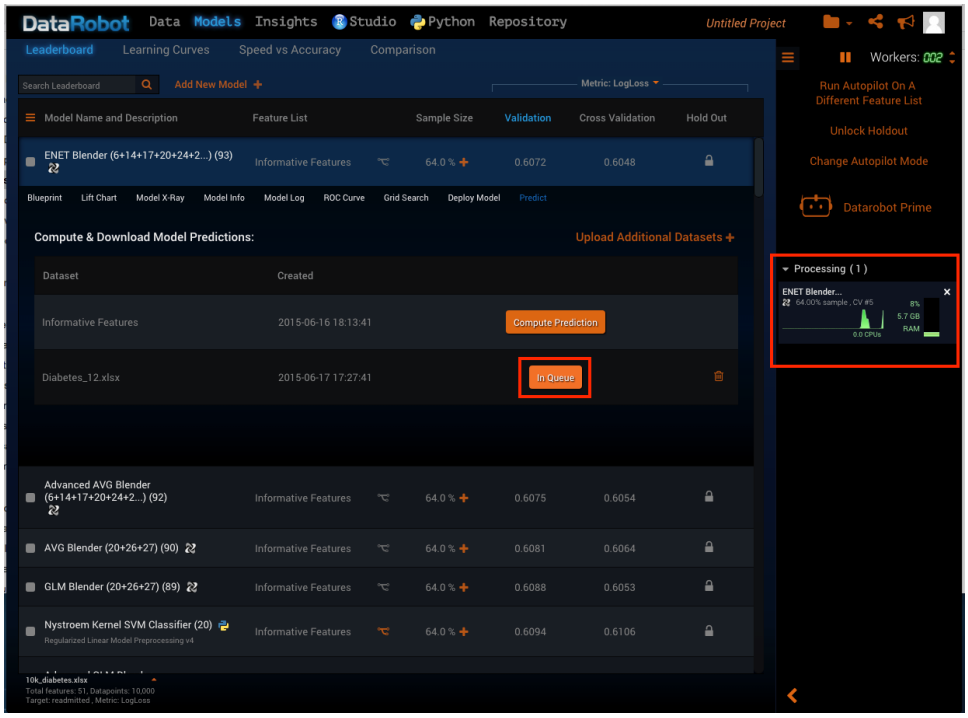
7. To predict against the model, click **Predict**, then select a dataset.



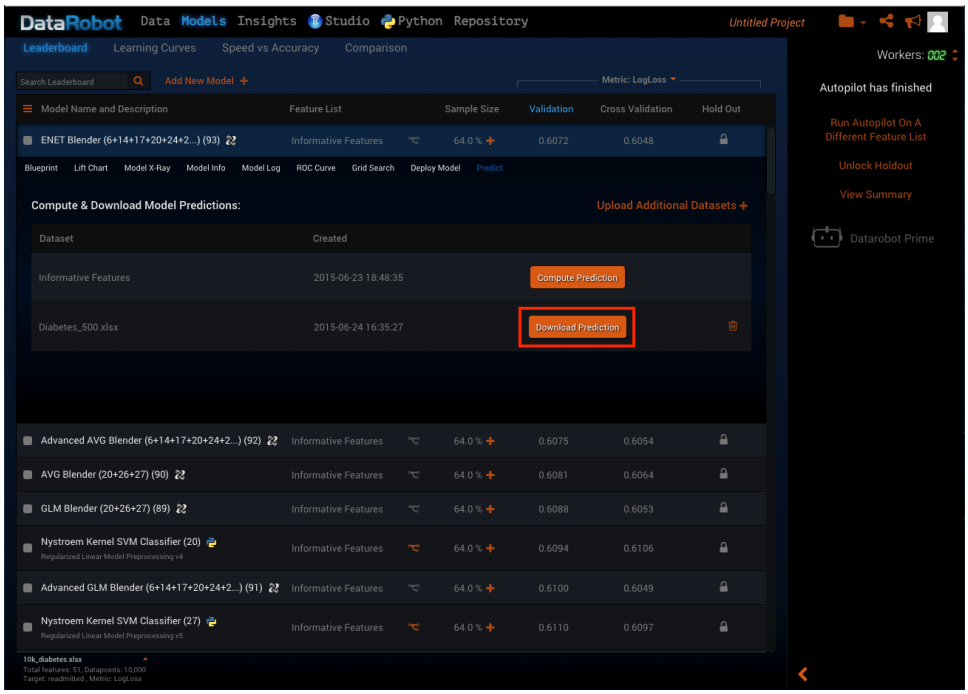
8. After the dataset has been uploaded, click **Compute Prediction** to generate predictions for the dataset.



9. The **Compute Prediction** button changes to **In Queue** for the selected dataset, and the job status appears under Processing in the Worker queue.



10. When the prediction has finished running, click **Download Prediction** to view the results in a .csv file. Prediction data is also retained in the project.


DataRobot
Data
Models
Insights
Studio
Python
Repository
Untitled Project
Leaderboard
Learning Curves
Speed vs Accuracy
Comparison
Workers:
Search Leaderboard
Add New Model
Metric: LogLoss
Autopilot has finished
Model Name and Description
Feature List
Sample Size
Validation
Cross Validation
Hold Out
ENET Blender (6+14+17+20+24+2...) (93)
Informative Features
64.0 %
0.6072
0.6048
Run Autopilot On A Different Feature List
Unlock Holdout
View Summary
Blueprint
Lift Chart
Model X-Ray
Model Info
Model Log
ROC Curve
Grid Search
Deploy Model
Predict
Compute & Download Model Predictions:
Upload Additional Datasets
Datarobot Prime
Dataset
Created
Informative Features
2015-06-23 18:48:35
Compute Prediction
Diabetes_500.xlsx
2015-06-24 16:35:27
Download Prediction
Advanced AVG Blender (6+14+17+20+24+2...) (92)
0.6075
0.6054
AVG Blender (20+26+27) (90)
0.6081
0.6064
GLM Blender (20+26+27) (89)
0.6088
0.6053
Nystroem Kernel SVM Classifier (20)
Regularized Linear Model Preprocessing v4
0.6094
0.6106
Advanced GLM Blender (6+14+17+20+24+2...) (91)
0.6100
0.6049
Nystroem Kernel SVM Classifier (27)
Regularized Linear Model Preprocessing v5
0.6110
0.6097
10k_diabetes.xlsx
Total features: 51, Datapoints: 10,000
Target: readmitted , Metric: LogLoss

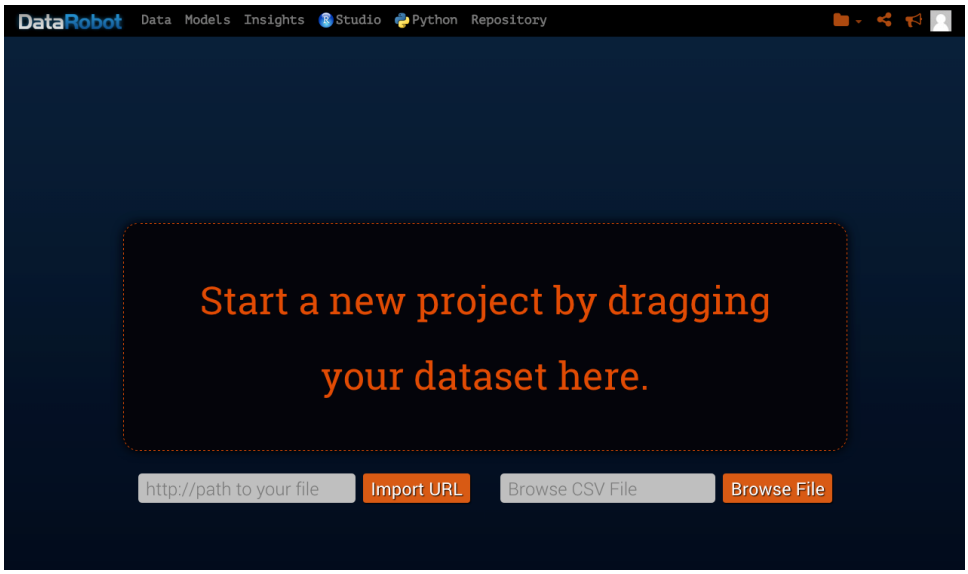
How To...

Create a Project

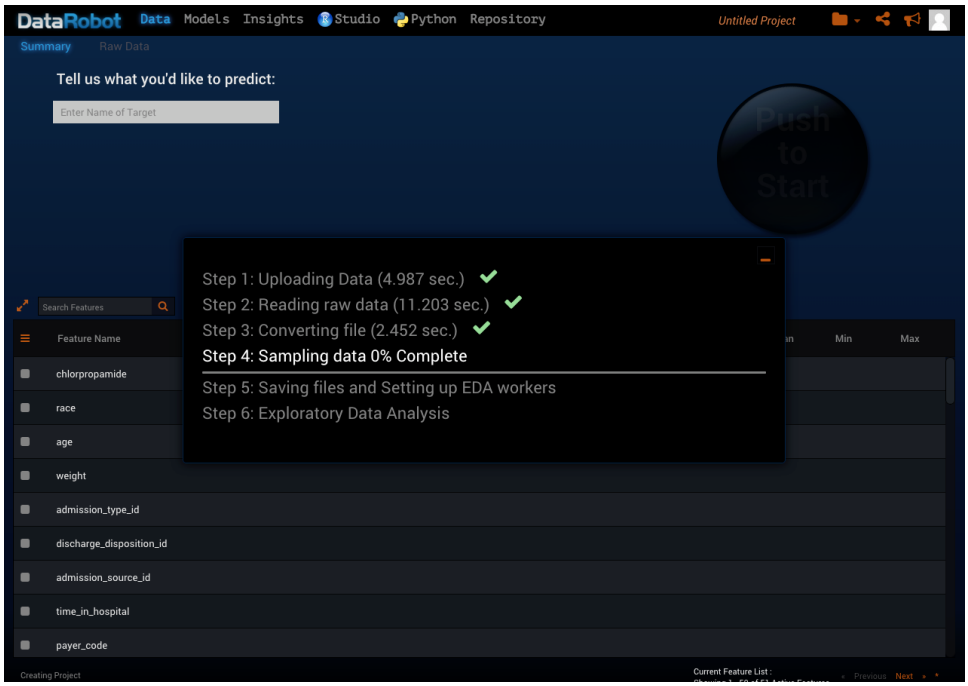
In order to build models in DataRobot, you must create a project with a dataset. After you create a project, you can select a target feature and start the modeling process. A DataRobot project contains all of the models you build for the project dataset.

Use the following steps to begin modeling data with the DataRobot API.

1. You can use any of the following methods to create a new DataRobot project:
   - Drag a .csv data file to the orange box on the DataRobot platform home page.
   - Click **Browse File**, then use the file browser to select a .csv data file.
   - Type a URL in the box labeled "http://path to your file", then click **Import URL**.



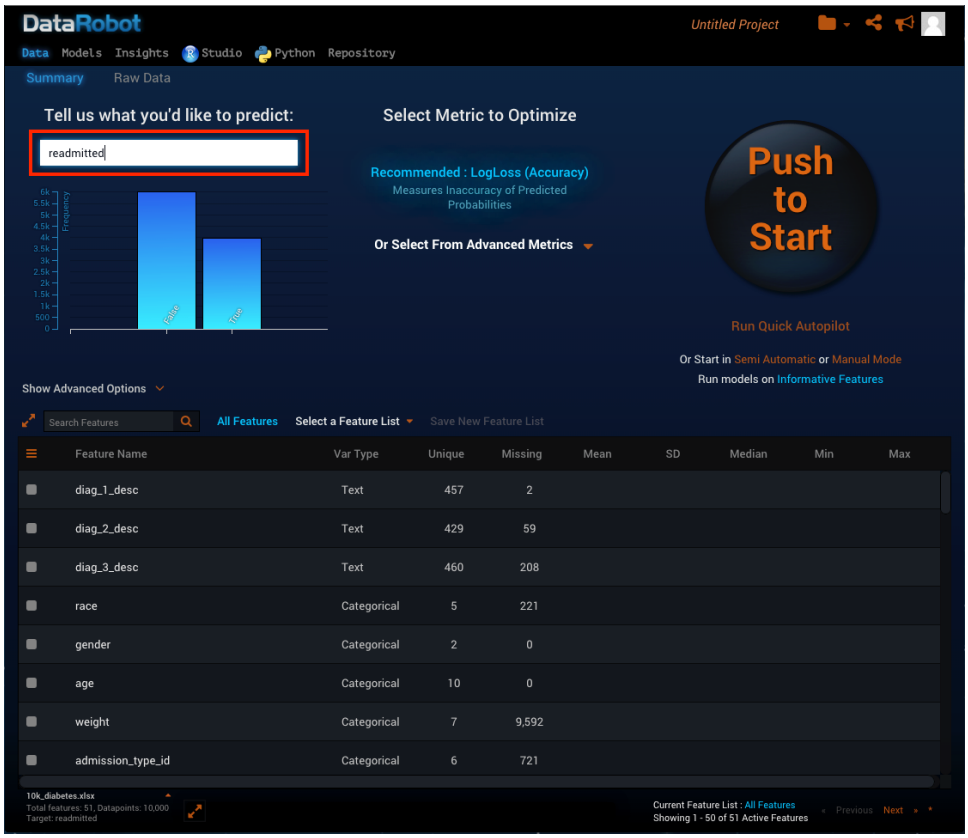
2. A progress indicator is displayed while the file is being processed.



Select a Target Feature

After you create a project, you can select a target feature and an optimization metric and start modeling. The target feature is the name of the column in the dataset you would like to predict. The optimization metric is the metric software used to optimize the models.

On the Data Summary page, type in the name of the target feature you would like to predict.



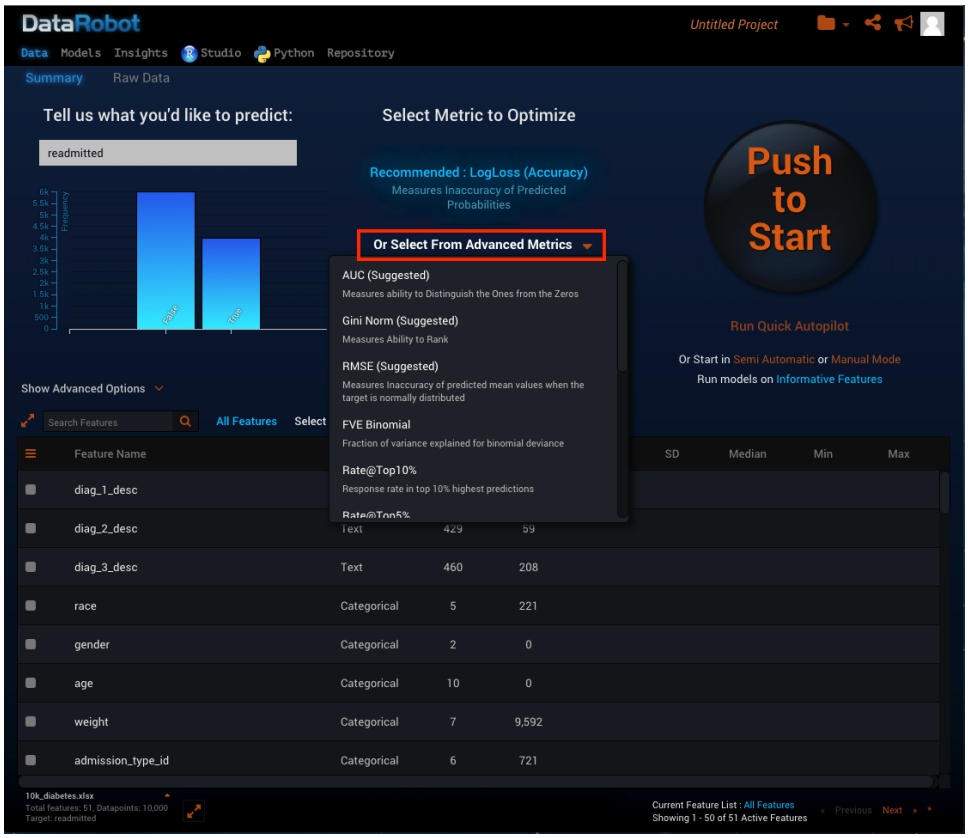
Select an Optimization Metric

After you create a project, you can select a target feature and an optimization metric and start modeling. The target feature is the name of the column in the dataset you would like to predict. The optimization metric is the metric software used to optimize the models.

After you type in a target feature, a recommended optimization metric is automatically selected based on the modeling task.

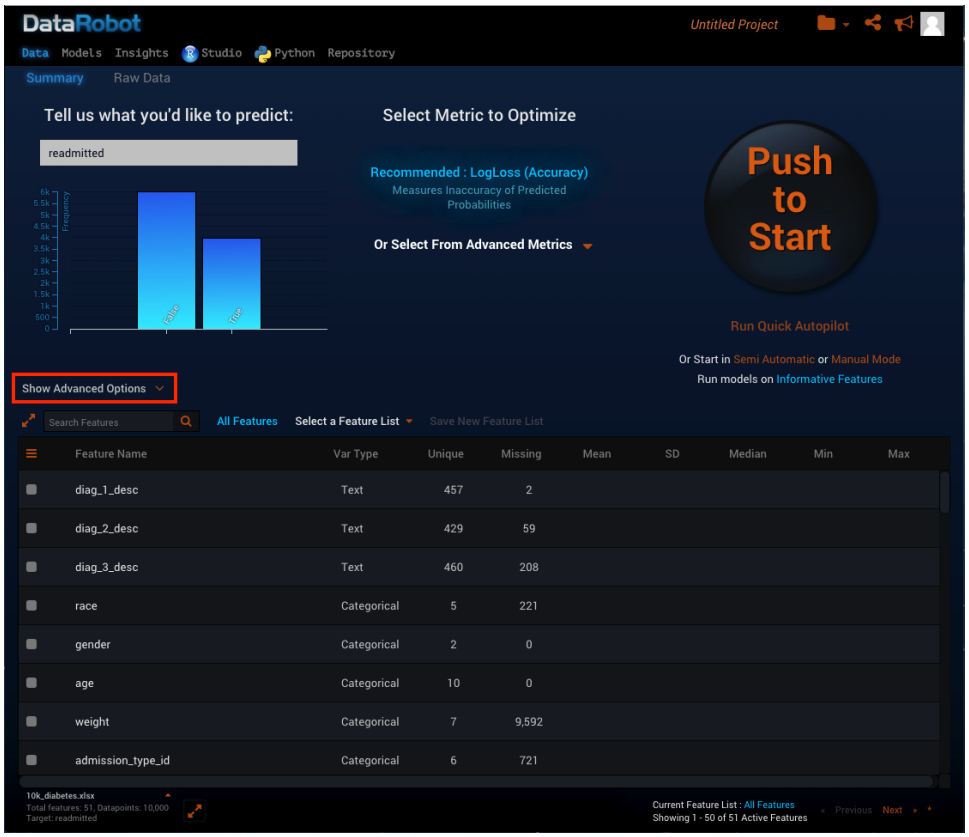
- If the selected target has only two unique values, the assumption is made that the task is a classification, and a classification metric will be recommended. Examples of recommended classification methods include LogLoss (if it is necessary to calculate a probability for each class), and Gini and AUC when it is necessary to sort records in order of ranking.
- Otherwise the assumption is made that the selected target represents a regression task. The most popular metrics for regression are RMSE (Root Mean Square Error) and MAD (Mean Absolute Deviation).

  If you would like to override the recommended optimization metric, click **Select from Advanced Metrics**, then select an alternate metric.

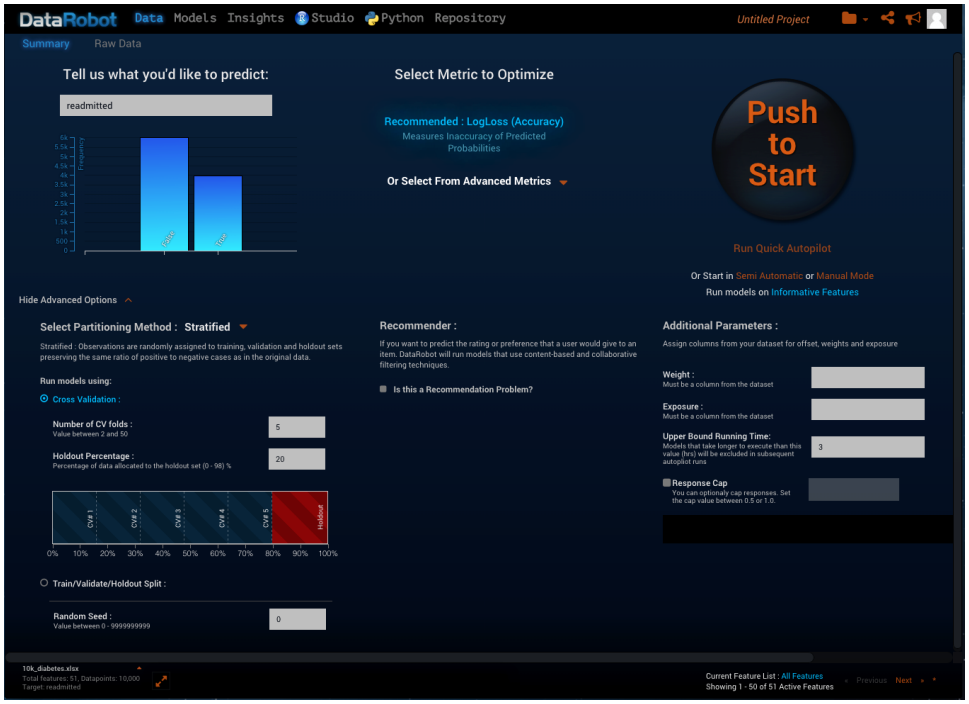


Set Advanced Modeling Parameters

To set advanced modeling parameters, click **Show Advanced Options** on the Data Summary page.



The advanced modeling options appear.



You can specify the following advanced options:

Partitioning Method

Random

Description: Observations are randomly assigned to the training, validation, and holdout sets.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.
- Random Seed -- specify a positive integer value for the random seed.

Partition Column

Description: A column in the data file is used to either specify a train/validate/holdout split or the folds/holdout to be used for cross-validation.

Modeling Options:

- Cross-Validation -- you must specify a value from the selected partition column that will specify the holdout set.
- Training/Validation/Holdout Split -- you must specify training, validation, and holdout set values from the selected partition column.

Group

Description: One or more columns can be selected, and each combination of these values is guaranteed to be in the same training or test set as other matching values.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.

Date

Description: Observations in the holdout set come after observations in the validation set chronologically, and observations in the validation set come after those in the training set. For this method you specify the Date/Time column in the dataset, which is used to create a Train/Validate/Holdout split with all rows in the test set occurring later than all rows in the training set, and all rows in holdout occurring later than the test set.

Modeling Options:

- Validation Percentage -- percentage of data allocated to the validation set.
- Holdout Percentage -- percentage of data allocated to the holdout set.

Stratified

Description: Observations are randomly assigned to training, validation, and holdout sets, preserving the same ratio of positive to negative cases as in the original data.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.

Recommender

Select the **Is this a Recommendation Problem?** check box if you would like to predict the rating or preference that a user would give to an item.

If you select this option, you must specify the two columns that contain the user and item identifiers.

A typical example for recommender systems is a product recommendation on an e-commerce site where you would like to predict how highly a user might rate a product. Predicted ratings are then used for content discovery on the site.

For recommendation problems, DataRobot runs models that use content-based and collaborative filtering techniques.

- Content-based models take into account any available user or item features and create a profile for each user (key words, price range, product types, brands, etc.).
- Collaborative filtering techniques are content-agnostic -- they look solely at the latent structure in the user-item ratings in order to make predictions.

DataRobot also runs hybrid models that combine both content-based and collaborative filtering techniques, as well as combinations of models (blended models).

Additional Parameters

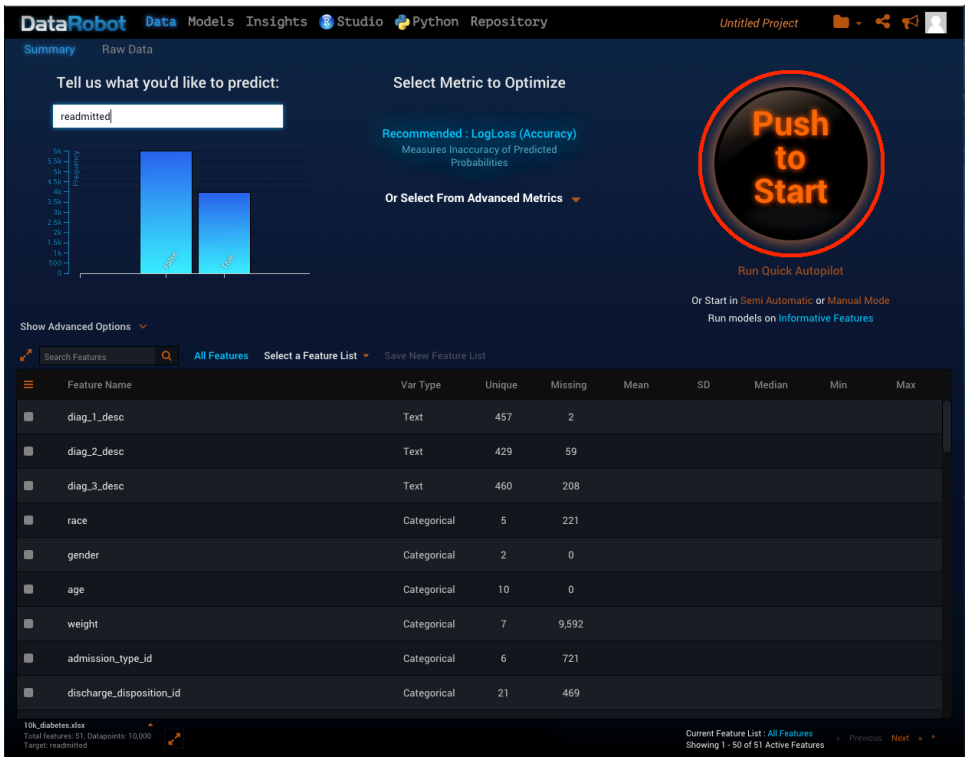
- Weight -- must be the name of a column in the dataset.
- Exposure -- must be the name of a column in the dataset.
- Upper Bound Running Time -- models that take longer to execute than this value (hrs) are excluded in subsequent autopilot runs.
- Response Cap -- The Response Cap limits the maximum value of the response (target) to a percentage of the original values. The value must be between 0.5 and 1.0 (50-100%).

Start Modeling

Start Modeling

After you have selected a target feature and an optimization metric, you can start modeling. The modeling process finds the best predictive models for the target feature.

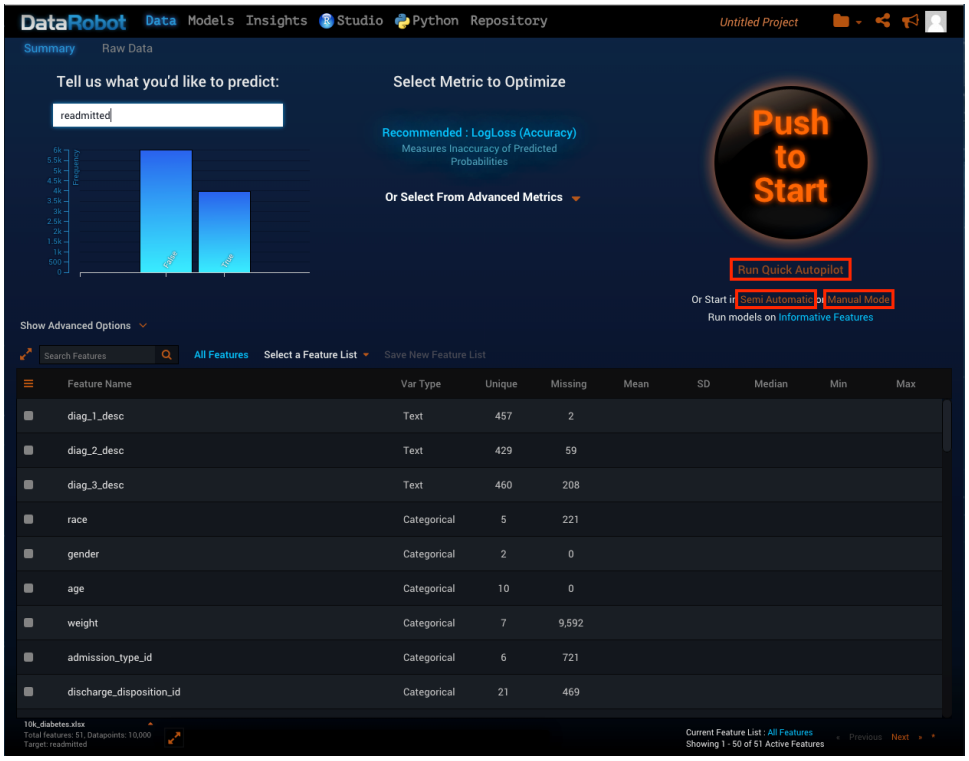
To start modeling, click the **Push to Start** button.



Autopilot Modes

The modeling process runs in Autopilot (fully automatic) mode by default, which means that the DataRobot platform will automatically select the best predictive models for the specified target feature.

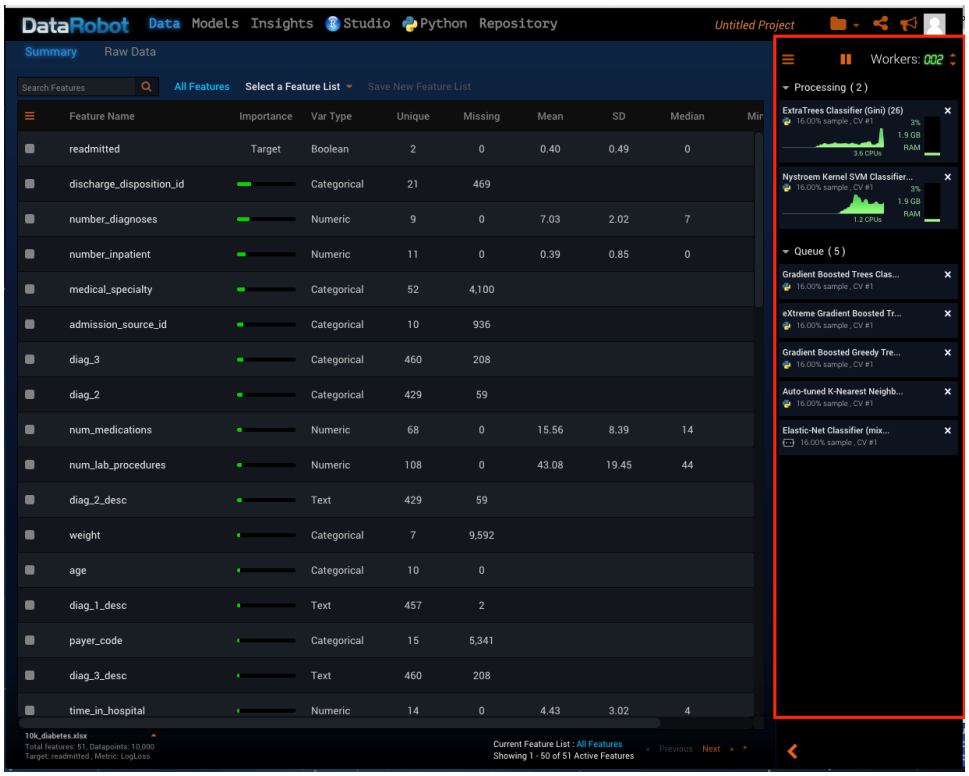
For more control over which models to run, you can select **Run Quick Autopilot**, **Semi Automatic**, or **Manual Mode** under the **Push to Start** button.



- **Run Quick Autopilot** -- The Quick Autopilot runs a small subset of models consisting of the best models based on the specified target feature and performance metric.
- **Semi Automatic** -- In semi-automatic mode, the autopilot will pause after the first cross-validation set before submitting additional models to the job queue. This allows you to inspect the models and choose to either add additional models or delete models.
- **Manual Mode** -- Starting the autopilot in manual mode gives you full control over which models to execute. For example, you can choose a specific model from the Task Repository rather than running the models selected by default.

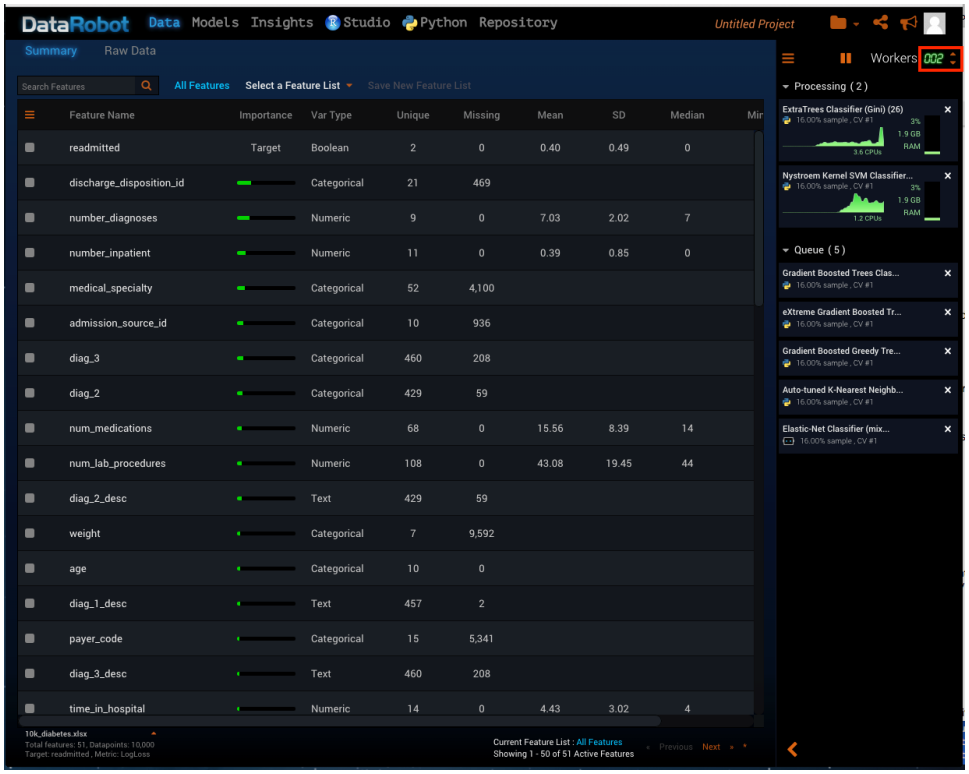
The Worker Queue

After you start the modeling process, running and pending jobs appear in the Worker Queue at the right of the page When a model is running, a graphical display of its CPU and RAM use will be displayed in the Worker Queue. The queue will execute a maximum fixed number of workers per project at any one time.



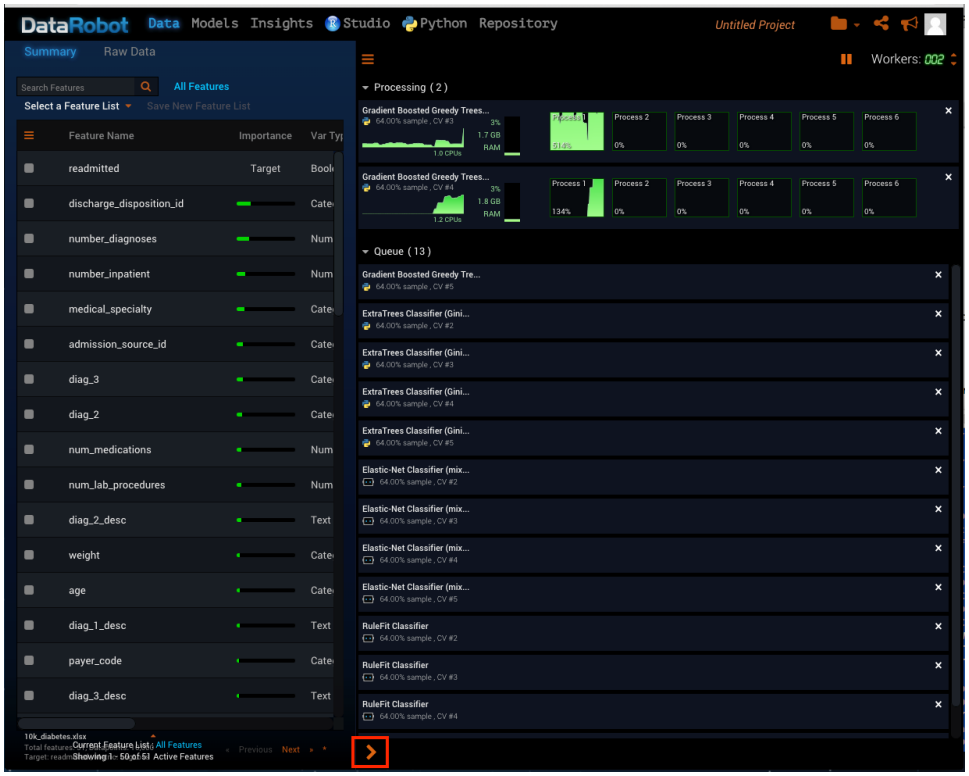
Adjusting the Number of Workers

You can adjust the maximum number of simultaneous workers by clicking the orange arrows at the right of the green numbers in the Workers field at the top of the Worker Queue.



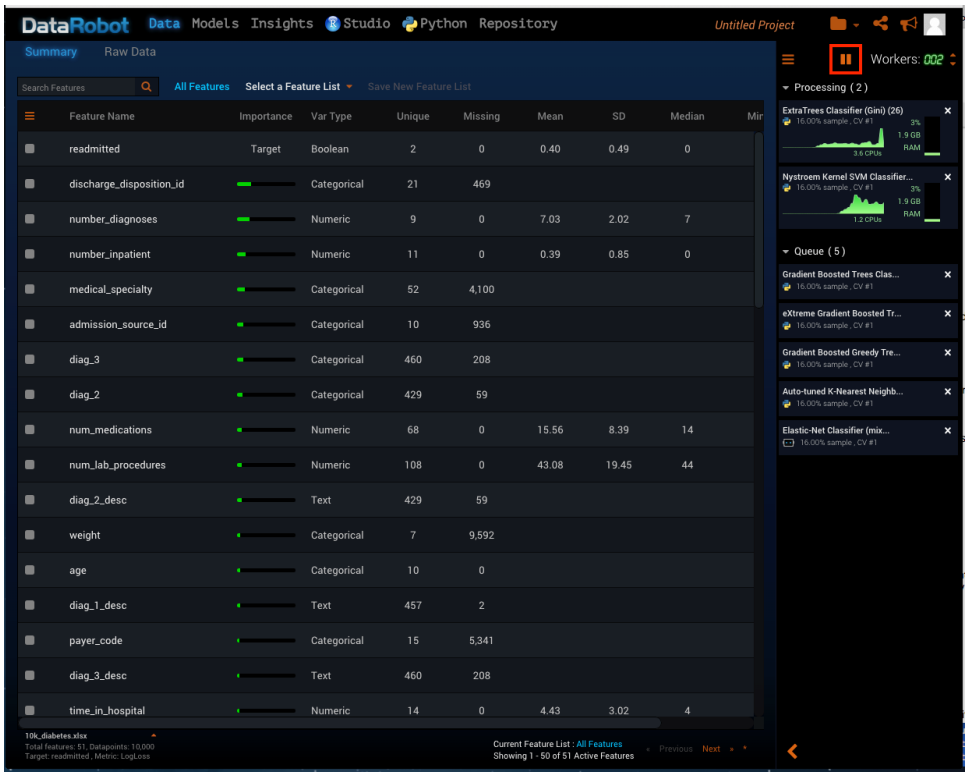
Viewing Worker Queue Details

To view more details about the running jobs, click the orange arrow at the bottom of the Worker Queue. Click the arrow again to hide queue details.



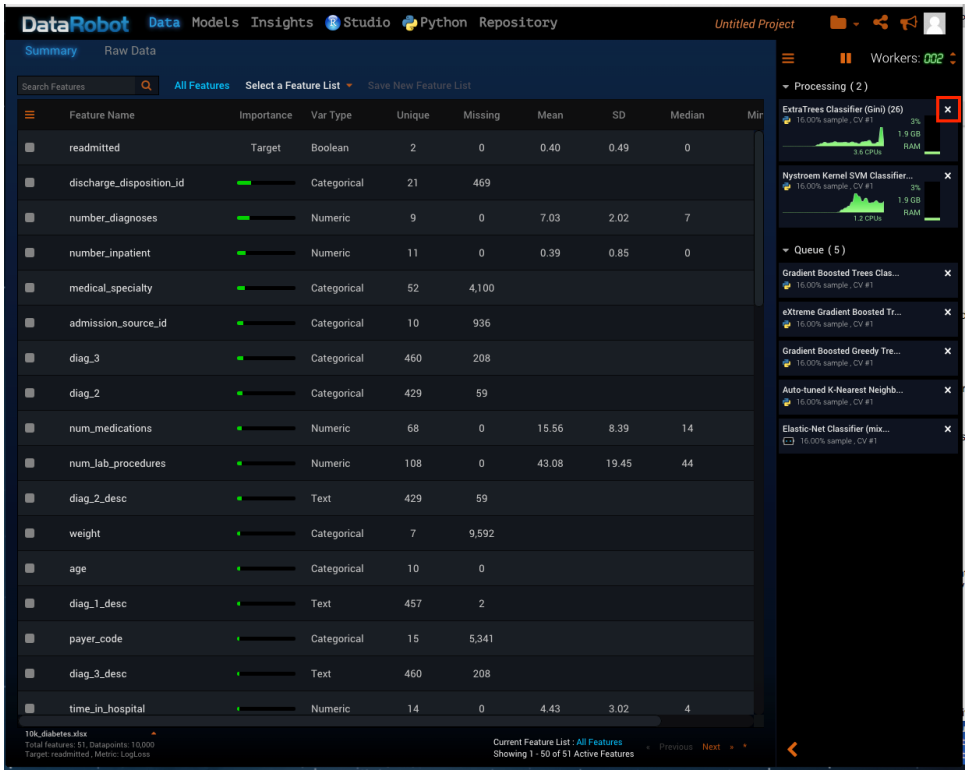
Pausing the Worker Queue

To pause the Worker Queue, click the Pause symbol at the top of the Worker Queue. After you pause the queue, the Pause symbol changes to a Play symbol (arrow). To resume running models, click the Play arrow.



Killing Workers

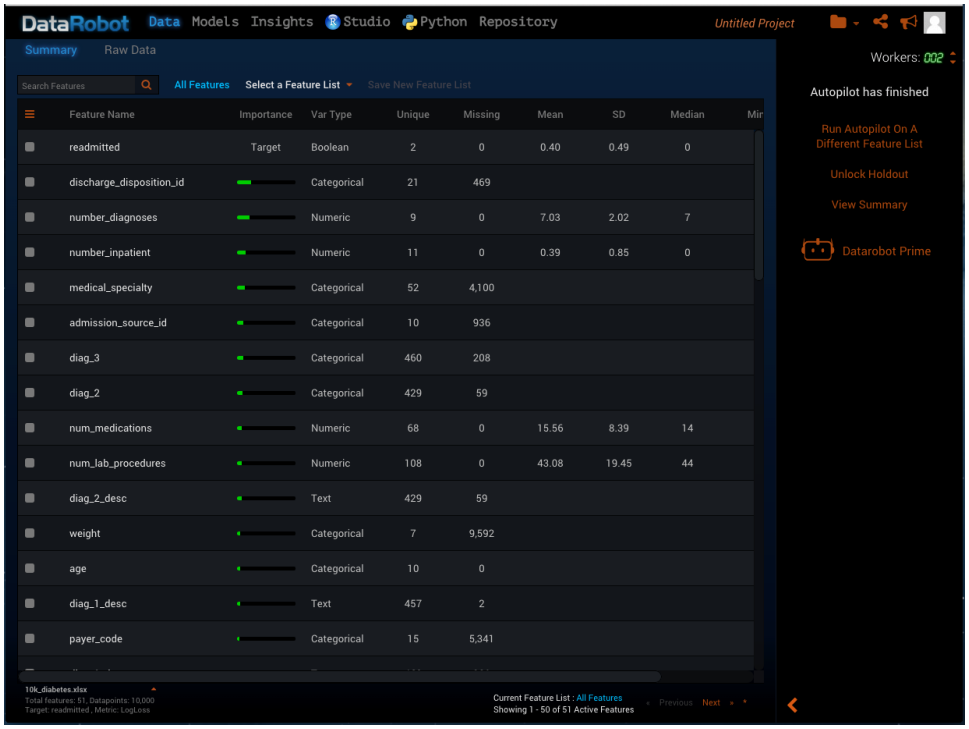
You can kill a worker by clicking the **X** next to the job name in the Worker Queue. If a worker fails for any reason, it will be listed under Errors at the bottom of the Worker Queue.



After modeling has started, you can click **Models** in the top navigation menu to view the Leaderboard. The Leaderboard is a list of models ranked by the chosen performance metric, with the best models at the top

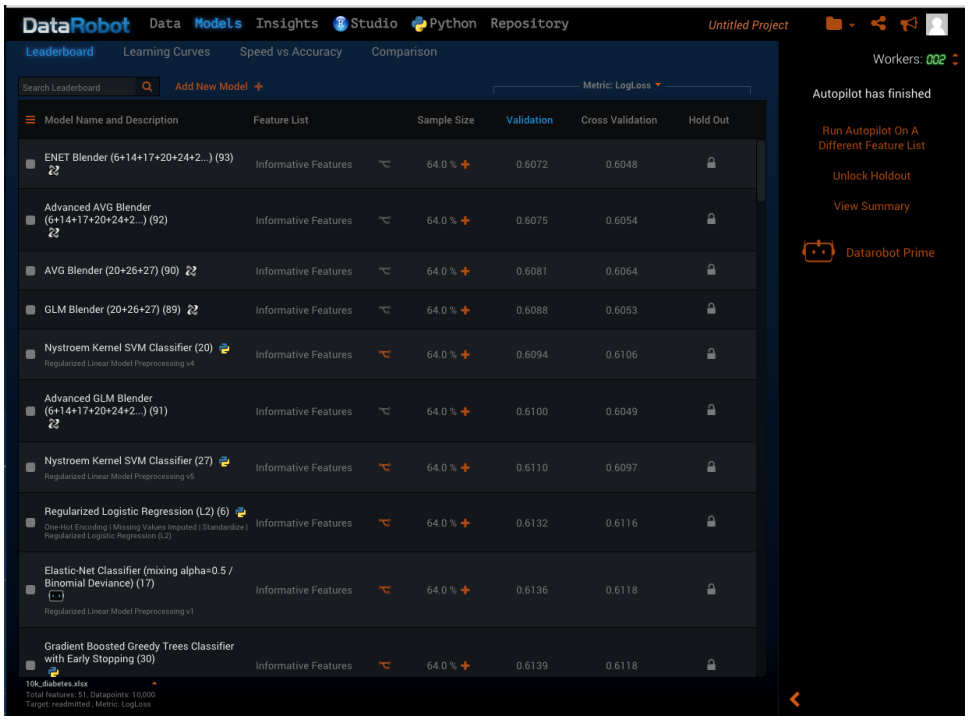
of the list. The rankings keep changing until all models have finished running. You cannot view full details about a model until it finishes running.

When all workers have finished running, the Worker Queue displays a message at the top of the queue and provide options to rerun the models a different feature list, or to unlock the holdout dataset and continue running workers.

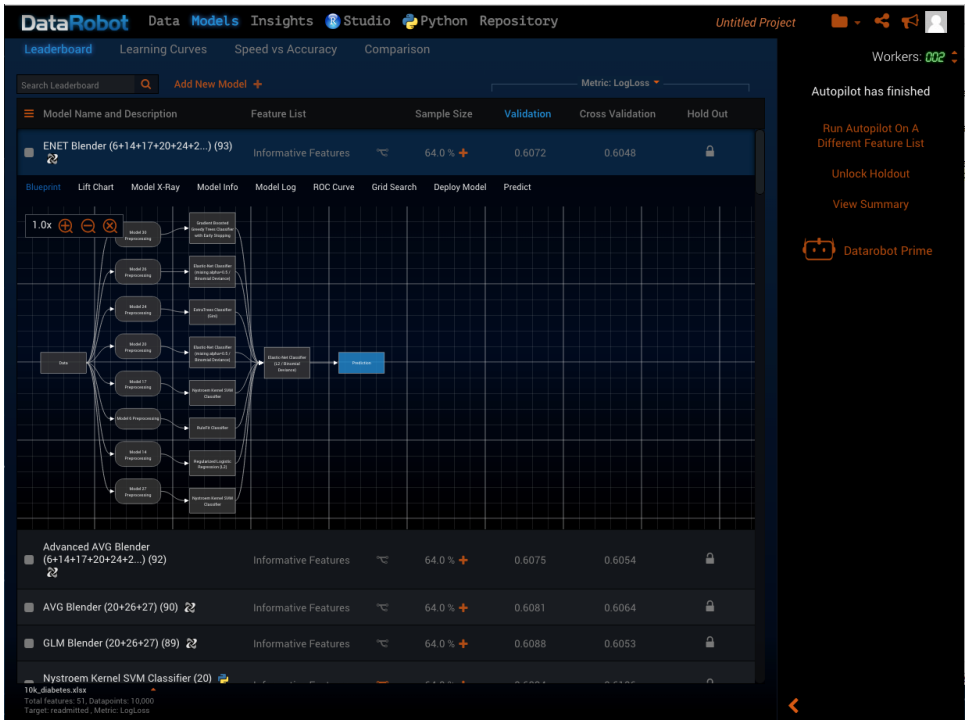


The Model Leaderboard

Click **Models** in the top navigation menu to view the Leaderboard. The Leaderboard is a list of models ranked by the chosen performance metric, with the best models at the top of the list. You cannot view full details about a model until it finishes running.



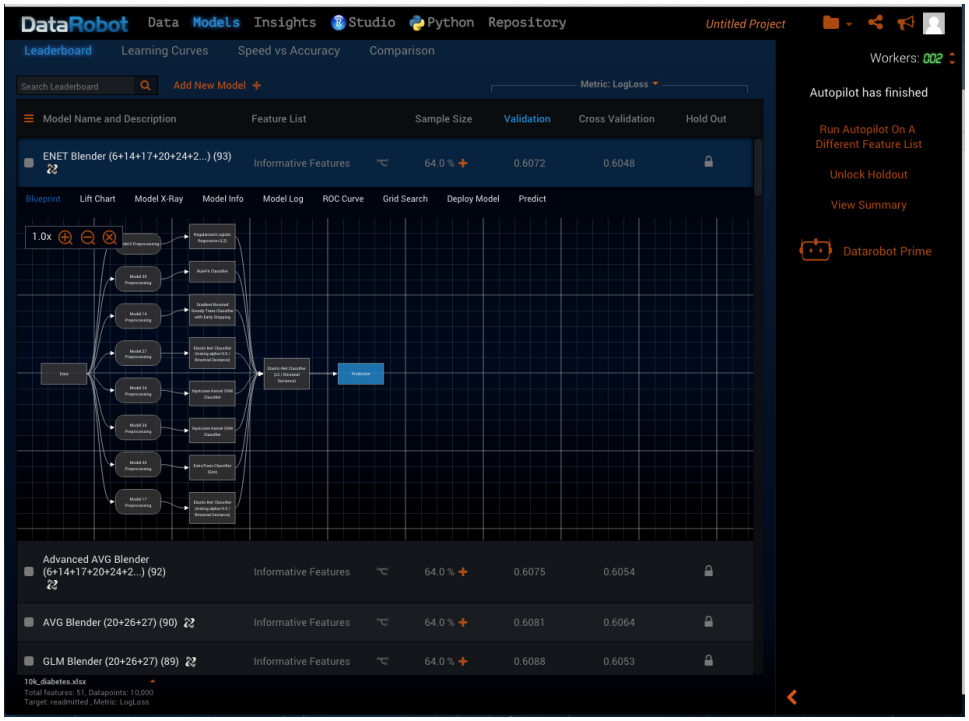
Click a model name to view details about the model. Model details are discussed in the following sections.



Blueprint

Blueprints provide a graphical representation of the many steps involved in transforming input predictors and targets into a model. A blueprint represents the high-level end-to-end procedure for fitting the model, including any pre-processing steps, algorithms, and post-processing. Each box in a blueprint may represent multiple steps.

To view a graphical representation of a blueprint, click a model in the leaderboard. You can use the navigation toolbar to zoom in on the blueprint.

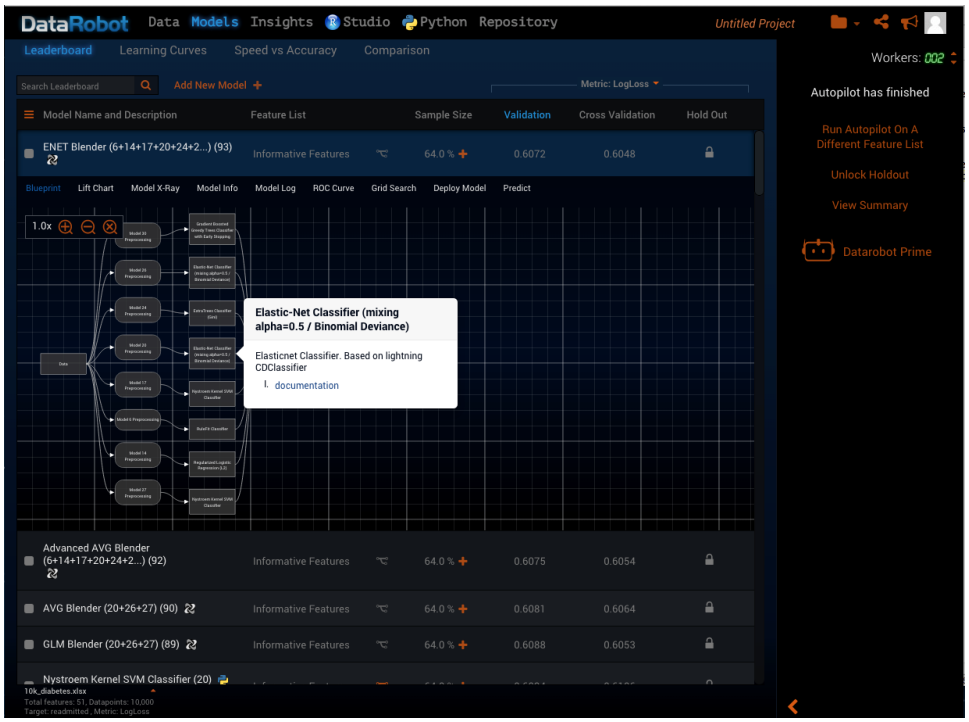


Input Types

Different columns in the dataset require different types of preparation and transformation. For example, some algorithms recommend subtracting the mean and dividing by the standard deviation of the input data -- but this would not make sense for text input data. The first step in the execution of a blueprint is to identify data types that belong together so they can be processed separately.

Transformers and Models

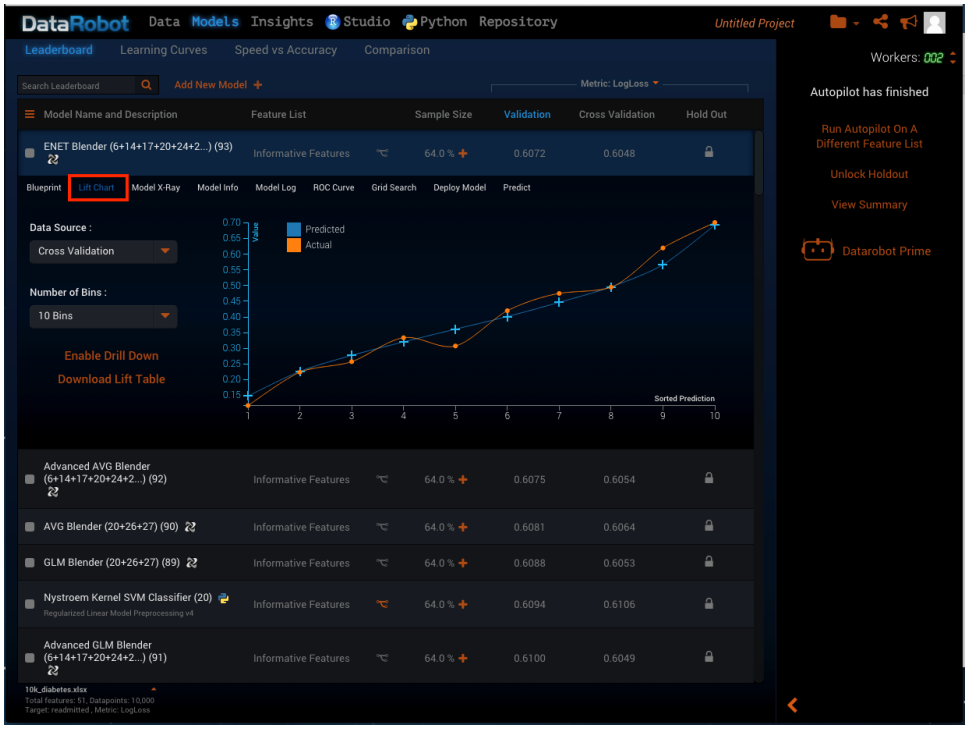
The other nodes in the blueprint are other types of data transformations or models. Click a blueprint node to display additional information. Many of the models use DataRobot proprietary approaches to data pre-processing.



Lift Chart

The lift chart depicts how effective a model is at predicting the target. The chart is sorted by predicted values, so you can see how well the model performs for different ranges of values of the target variable.

To view a lift chart, click a model in the Leaderboard list, then click **Lift Chart**.



Display Options

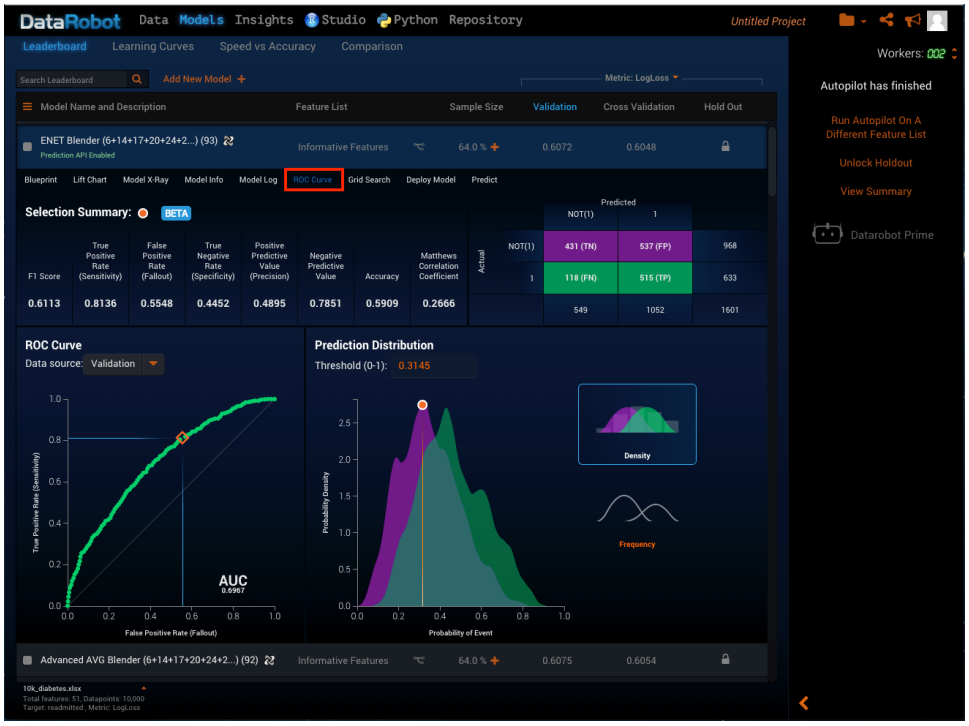
- Data Source -- select validation, cross-validation, or holdout.
- Number of Bins -- use this drop-down to adjust the granularity of the displayed values.
- Enable Drill Down -- click this link to compute and download cross-validation predictions.
- Download Lift Table -- download a .csv file with the lift table data.

ROC Curve

You can use the ROC Curve page to assess model quality. To view the ROC Curve page, click a model in the Leaderboard list, then click **ROC Curve**.

Note: The ROC curve is displayed only for models created for a binary classification target, i.e., a target with two unique values.

The ROC Curve page contains a set of interactive graphical displays that includes the ROC Curve, a selection summary, the prediction distribution, and a confusion matrix.



ROC Curve

The ROC Curve plots the true positive rate against the false positive rate. It has two important characteristics: the area under the curve, and the shape of the curve.

Area Under the ROC Curve

The Area Under the Curve (AUC) is displayed at the lower right of the ROC Curve. The goal is for the AUC to be as large as possible.

- If the area = 0.5, it means that the predictions based on this model are no better than a random model.
- If the area = 1.0, it would mean that predictions based on the model are perfect.

A perfect model is not realistic -- if it does happen, there is likely something wrong with the model (it may contain variables that depend on the response and should be excluded). Therefore the goal is to achieve an area as large as possible between 0.5 and 1.0.

ROC Curve Shape

Another informative feature of the ROC Curve is its shape. It is better when the curve grows quickly for small X-values, and slowly for values of X closer to 1. An ideal ROC curve would hug the top left corner, indicating a high true positive rate and a low false positive rate.

Display Options

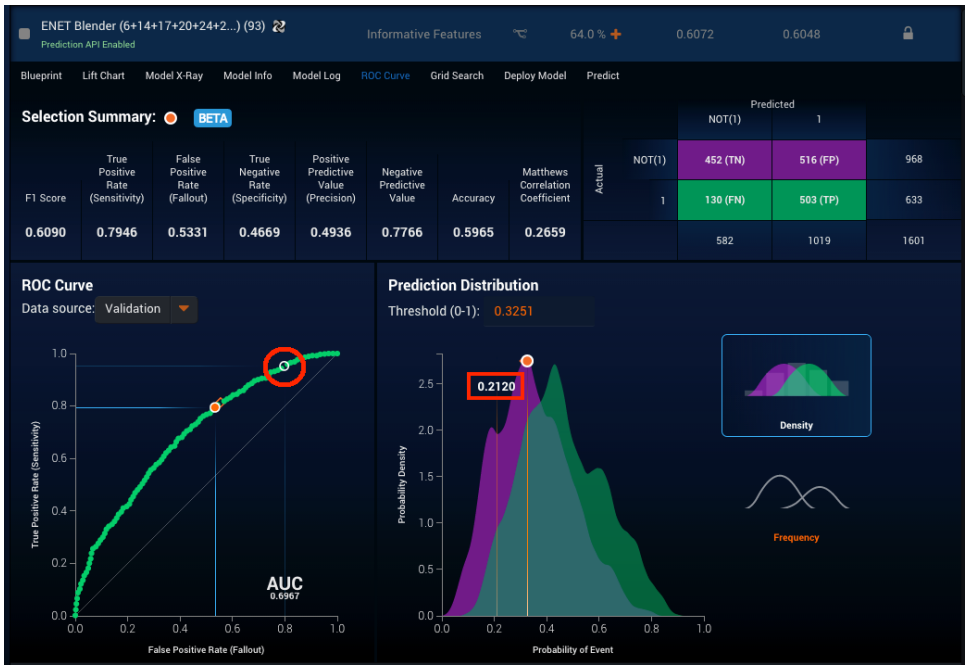
You can use the Data Source drop-down to display validation or cross-validation data.

Prediction Distribution

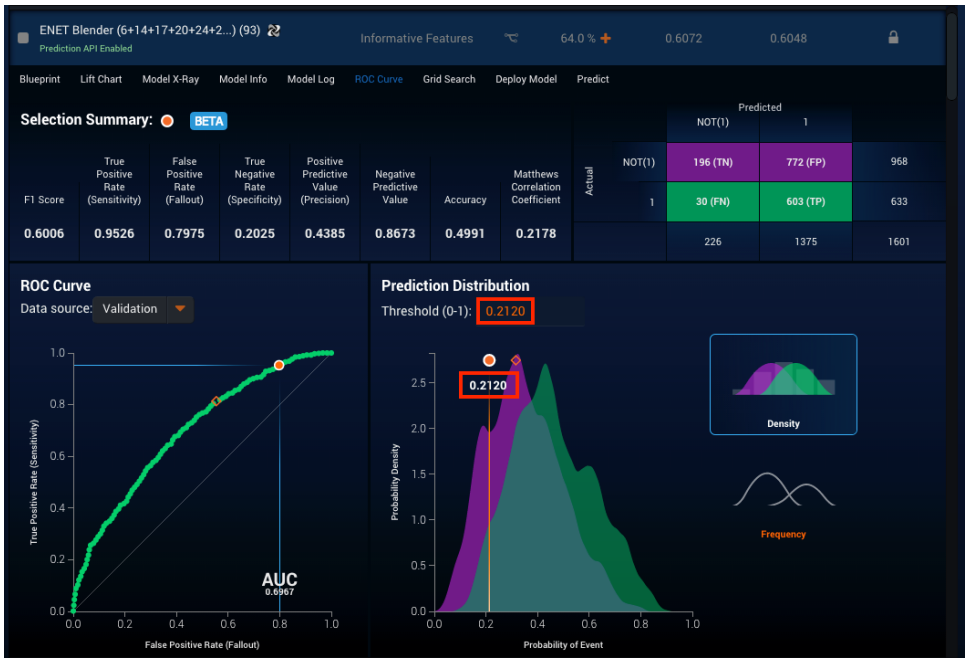
The Prediction Distribution chart expresses the performance of the model on the validation, cross-validation, or holdout dataset. You can use the Prediction Distribution chart to change the threshold value. When you change the threshold value, the ROC Curve, the Selection Summary table, and the Confusion Matrix are also updated to reflect the new value.

Use the following steps to set a new threshold value.

1. Pass the cursor over the Prediction Distribution chart. The threshold value will be displayed in white text as you move the cursor over the chart, and guidelines for the new values will also be dynamically displayed on both the Prediction Distribution chart and the ROC Curve.



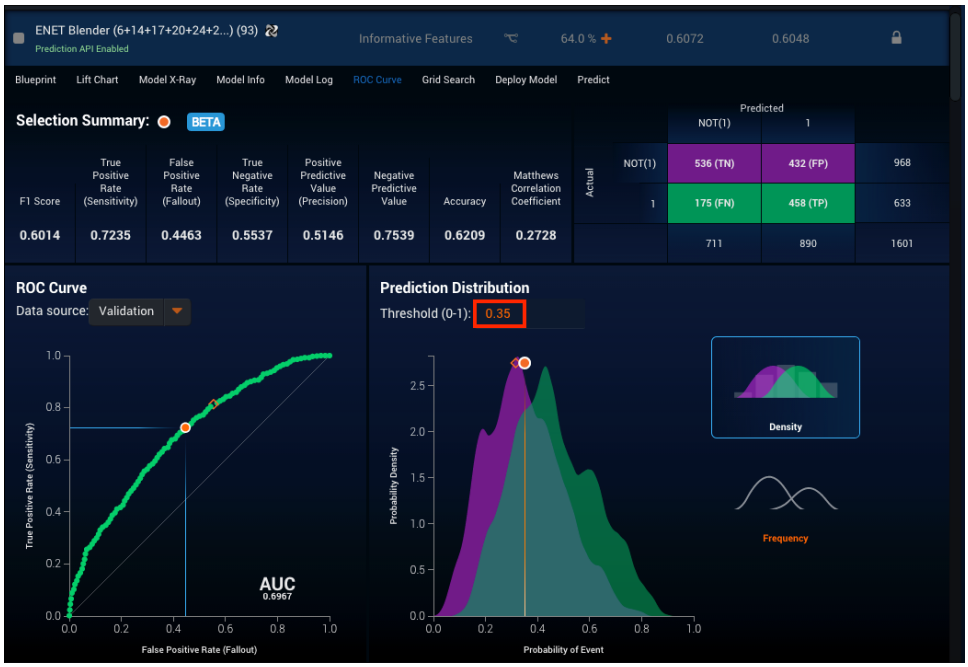
2. Click to select a new threshold value. The new value will appear in orange text in the Threshold field, and the orange dots and guidelines on the Prediction Distribution graph and ROC Curve will move to reflect the new threshold value. The information in the Selection Summary table and the Confusion Matrix will also be updated to reflect the new value.



3. You can also change the threshold setting by typing a new value in the Threshold field. Click the orange Threshold text, then type in a new value.

   Note: "Invalid value" will appear next to the field if the number entered is not between 0 and 1. If a non-numeric character is entered, "NaN" (Not a Number) will appear in the Threshold field.

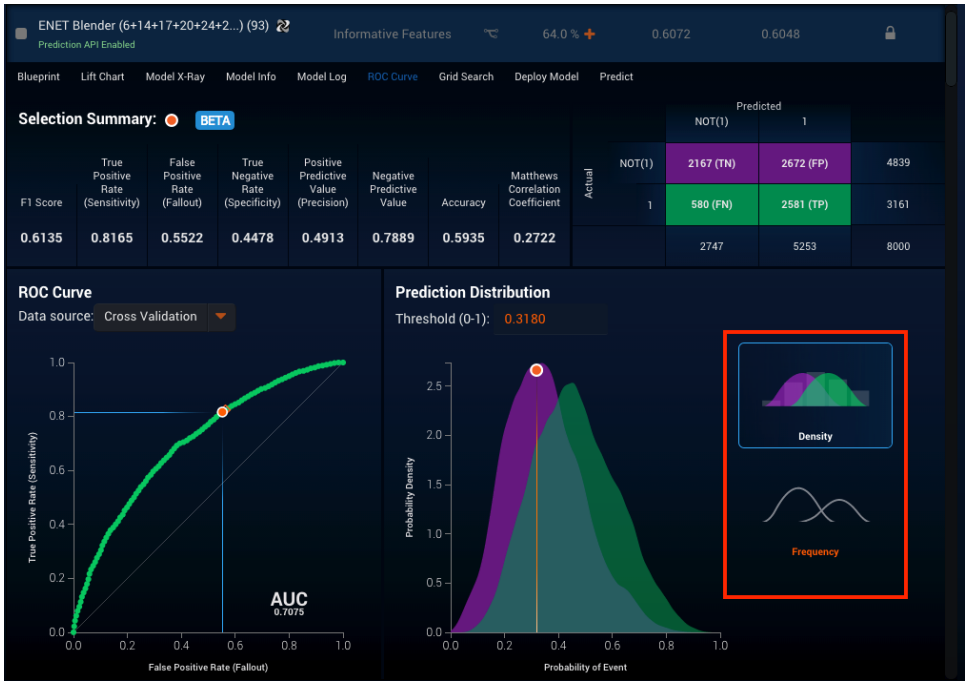
   The orange dots and guidelines on the Prediction Distribution graph and ROC Curve will move to reflect the new threshold value, and the information in the Selection Summary table and the Confusion Matrix will also be updated to reflect the new value.



4. Click the **Density** and **Frequency** symbols at the right of the Prediction Distribution chart to display the Predictions distribution as a density or frequency curve:

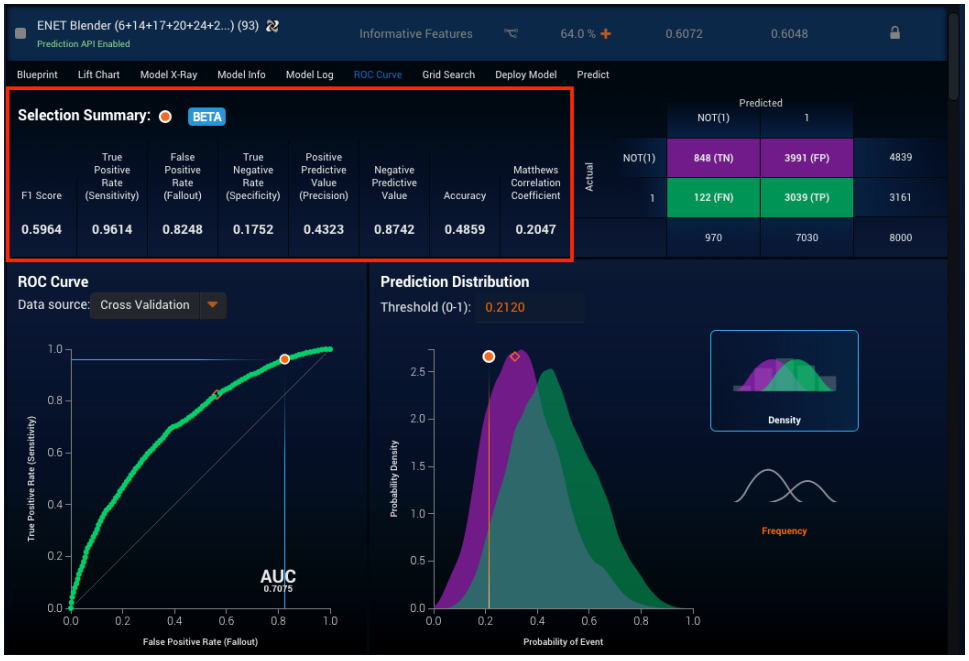
   - Density curve -- There is an equal area underneath both the positive and negative curves.

- Frequency curve -- The area underneath each curve varies, and is determined by the number of observations in each class.



Selection Summary

The Selection Summary table contains a number of statistics describing model performance at the selected threshold, including the F1 Score (the normalized accuracy), sensitivity, fallout, specificity, precision, negative predictive value, accuracy, and the Matthews Correlation Coefficient.

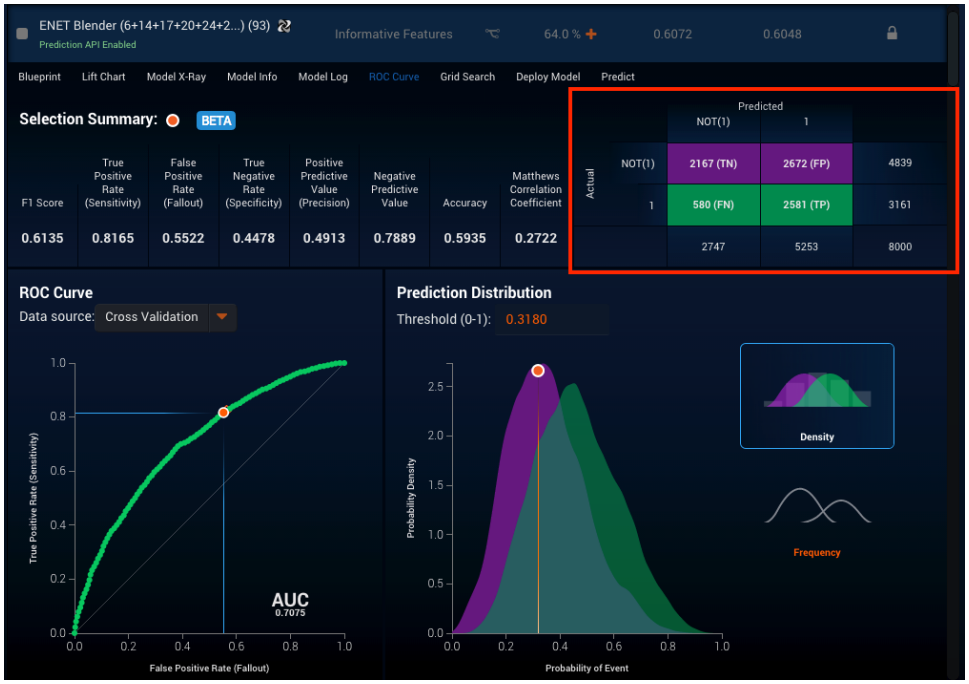


Confusion Matrix

The Confusion Matrix is a table with two rows and two columns that reports the number of false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class.

The name "confusion matrix" refers to the fact that the matrix makes it easy to see if the model is confusing two classes (consistently mislabeling one class as another class).

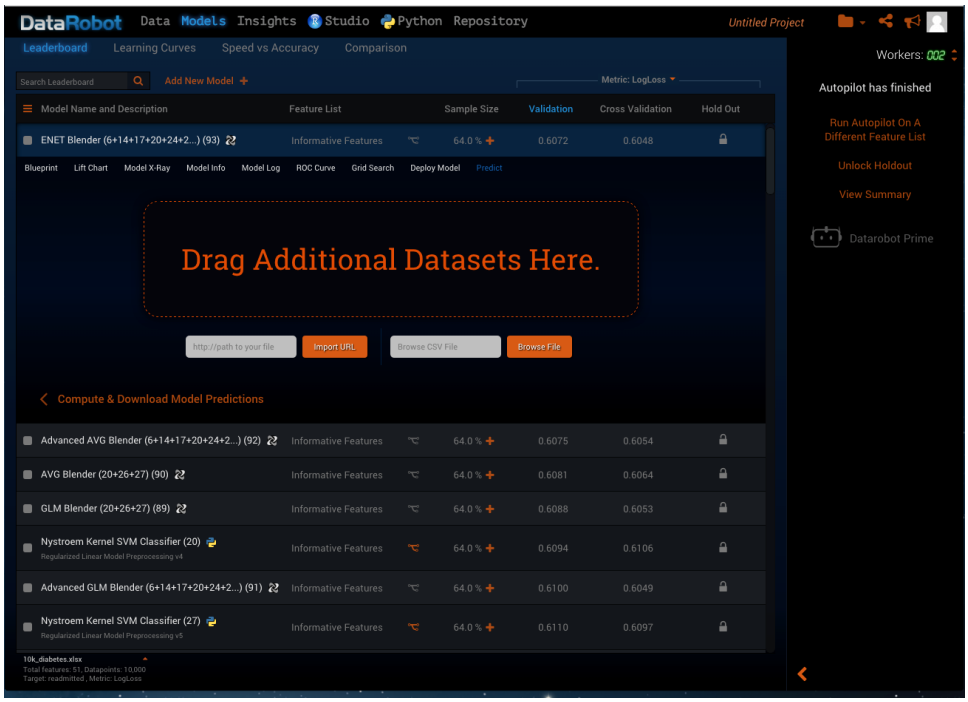
The information in this table facilitates more detailed analysis than relying on accuracy alone. Accuracy is not always a reliable metric for the real performance of a classifier, because it will yield misleading results if the data set is unbalanced (that is, when the number of samples in different classes varies greatly).



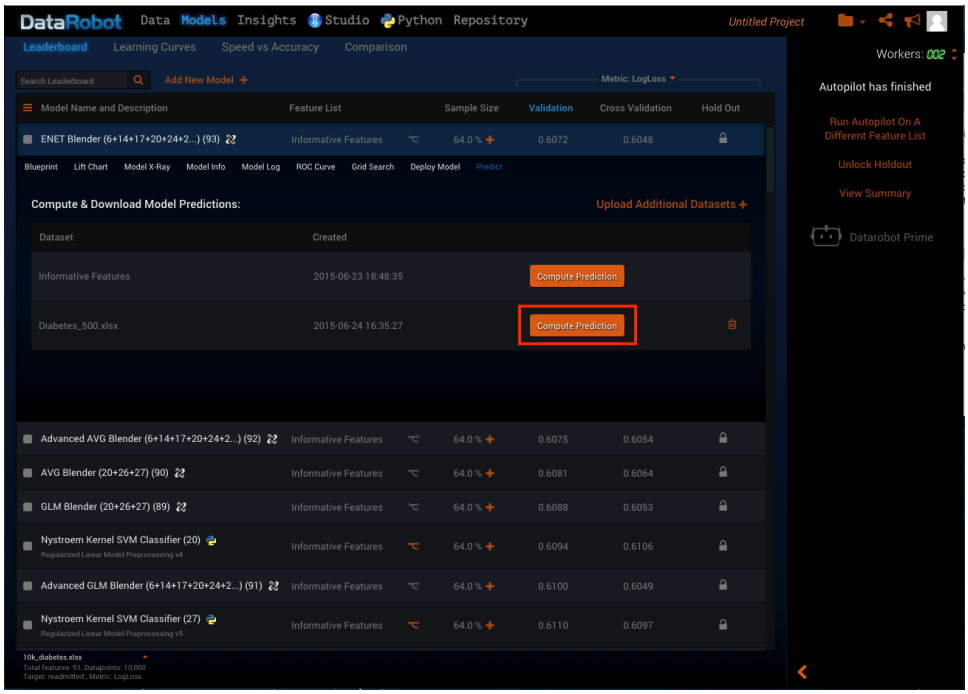
Predict Against a Model

Use the following steps to generate predictions on a new dataset.

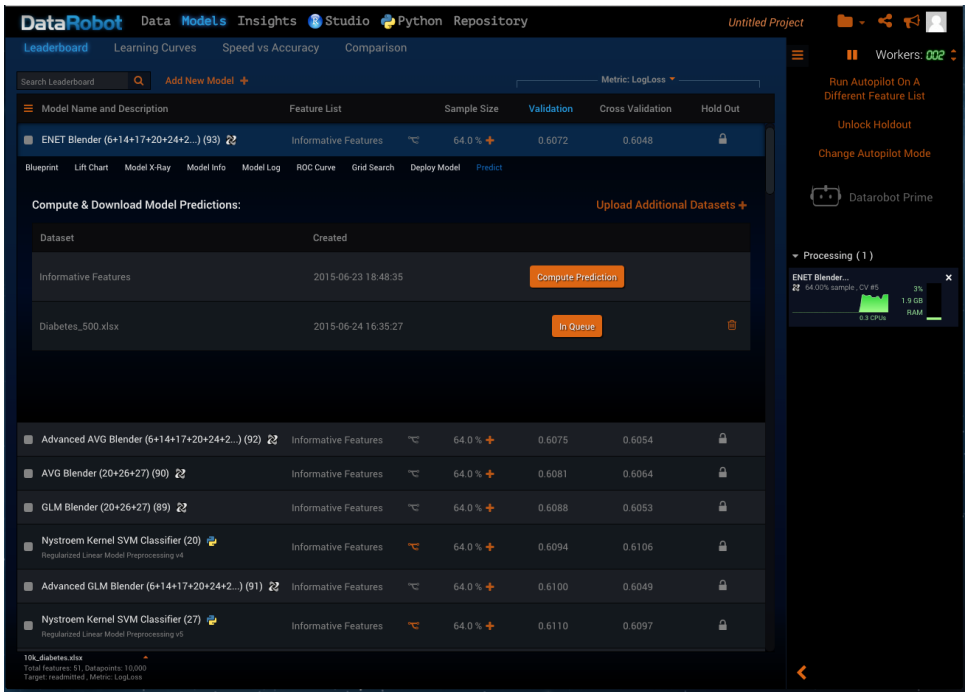
1. Click a model in the Leaderboard list, then click **Predict**. Use one of the following methods to select a dataset to predict with the model.
    - Drag a .csv data file to the orange box on the DataRobot platform home page.
    - Click Browse File, then use the file browser to select a .csv data file.
    - Type a URL in the box labeled "http://path to your file", then click Import URL.



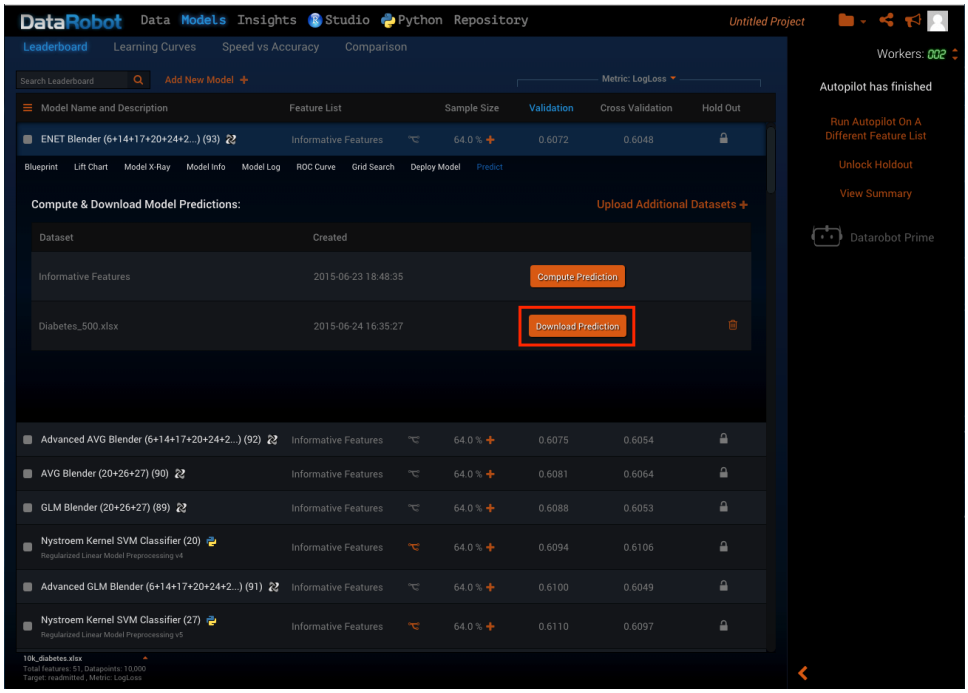
2. After the file is uploaded, click **Compute Prediction** next to the file.



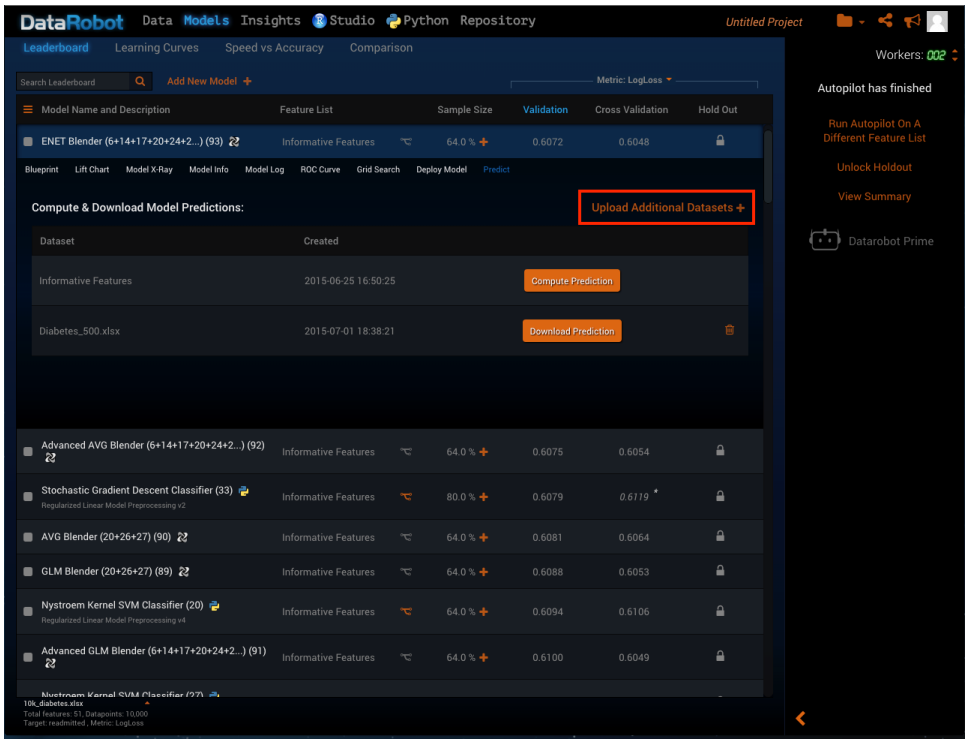
3. The **Compute Prediction** button changes to **In Queue** for the selected dataset, and the job status appears under Processing in the Worker queue.



4. When the prediction has finished running, click **Download Prediction** to view the results in a .csv file.



5. To upload and run predictions on additional datasets, click **Upload Additional Datasets**.


DataRobot
Data
Models
Insights
Studio
Python
Repository
Untitled Project
Leaderboard
Learning Curves
Speed vs Accuracy
Comparison
Workers: 002
Search Leaderboard
Add New Model
Metric: LogLoss
Autopilot has finished
Model Name and Description
Feature List
Sample Size
Validation
Cross Validation
Hold Out
ENET Blender (6+14+17+20+24+2...) (93)
Informative Features
64.0 %
0.6072
0.6048
Run Autopilot On A Different Feature List
Unlock Holdout
View Summary
Blueprint
Lift Chart
Model X-Ray
Model Info
Model Log
ROC Curve
Grid Search
Deploy Model
Predict
Compute & Download Model Predictions:
Upload Additional Datasets +
Datarobot Prime
Dataset
Created
Informative Features
2015-06-25 16:50:25
Compute Prediction
Diabetes_500.xlsx
2015-07-01 18:38:21
Download Prediction
Advanced AVG Blender (6+14+17+20+24+2...) (92)
Informative Features
64.0 %
0.6075
0.6054
Stochastic Gradient Descent Classifier (33)
Regularized Linear Model Preprocessing v2
Informative Features
80.0 %
0.6079
0.6119
AVG Blender (20+26+27) (90)
Informative Features
64.0 %
0.6081
0.6064
GLM Blender (20+26+27) (89)
Informative Features
64.0 %
0.6088
0.6053
Nystroem Kernel SVM Classifier (20)
Regularized Linear Model Preprocessing v4
Informative Features
64.0 %
0.6094
0.6106
Advanced GLM Blender (6+14+17+20+24+2...) (91)
Informative Features
64.0 %
0.6100
0.6049
10k_diabetes.xlsx
Total features: 51, Datapoints: 10,000
Target: readmitted , Metric: LogLoss

Managing Projects

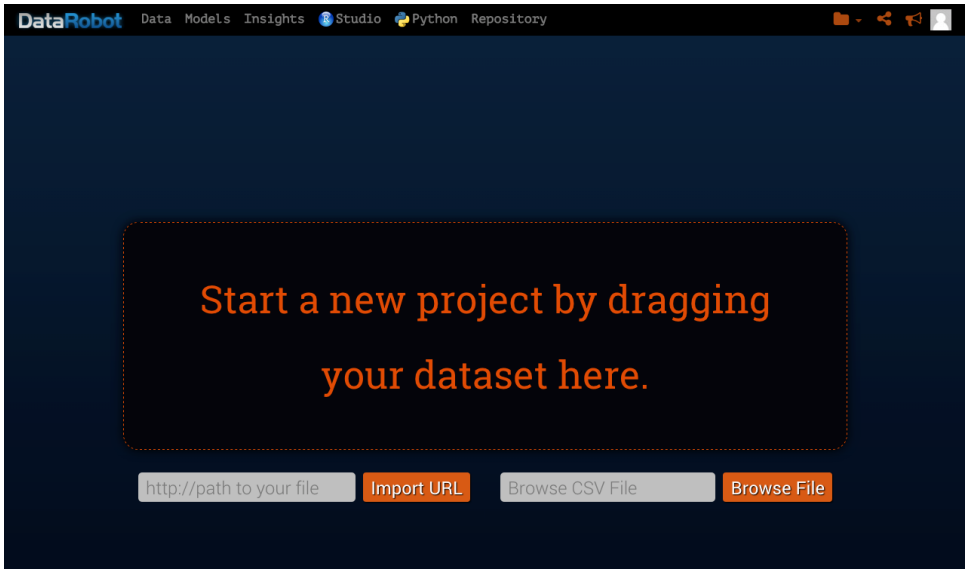
In order to build models in DataRobot, you must create a project. Each project has one dataset that is used as the source from which to train models.

Create a Project

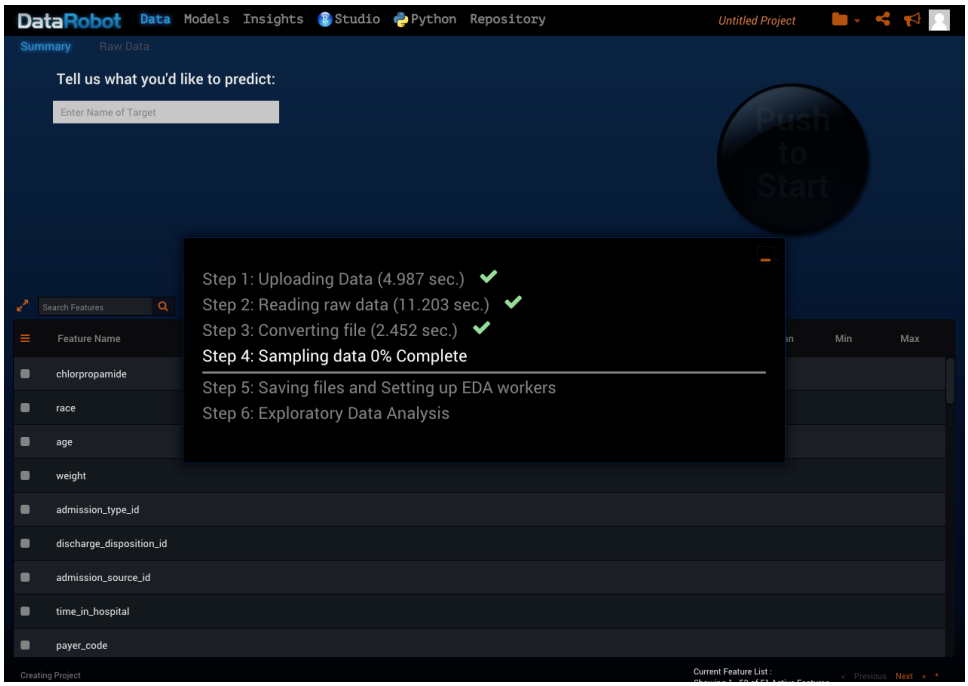
In order to build models in DataRobot, you must create a project with a dataset. After you create a project, you can select a target feature and start the modeling process. A DataRobot project contains all of the models you build for the project dataset.

Use the following steps to begin modeling data with the DataRobot API.

1. You can use any of the following methods to create a new DataRobot project:
   - Drag a .csv data file to the orange box on the DataRobot platform home page.
   - Click **Browse File**, then use the file browser to select a .csv data file.
   - Type a URL in the box labeled "http://path to your file", then click **Import URL**.

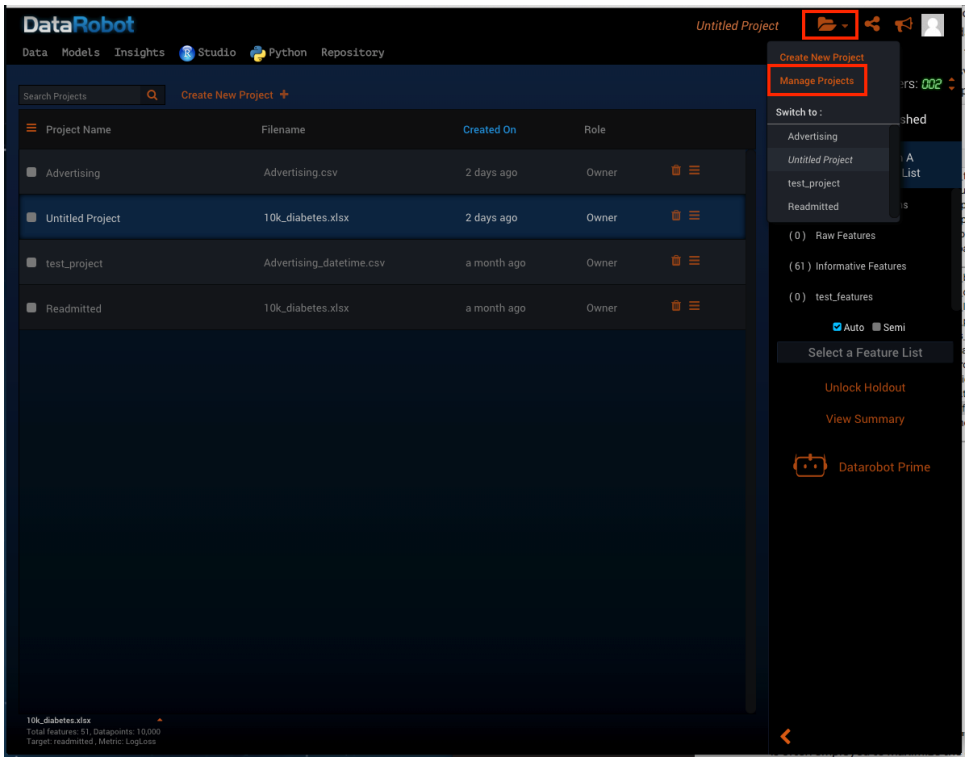


2. A progress indicator is displayed while the file is being processed.

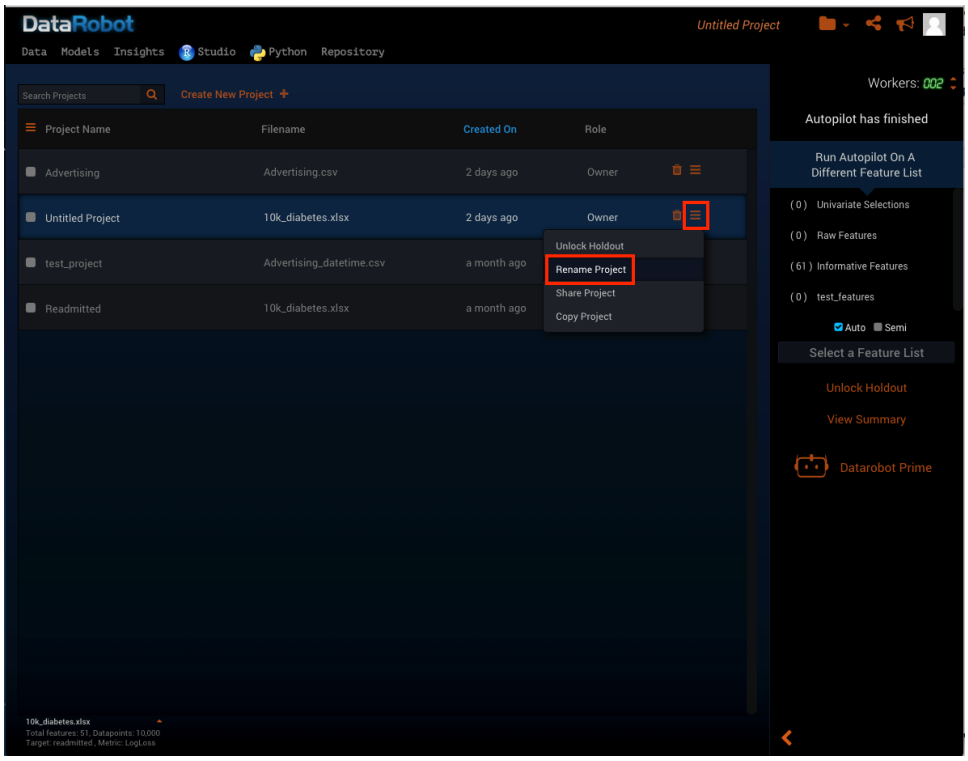


Rename a Project

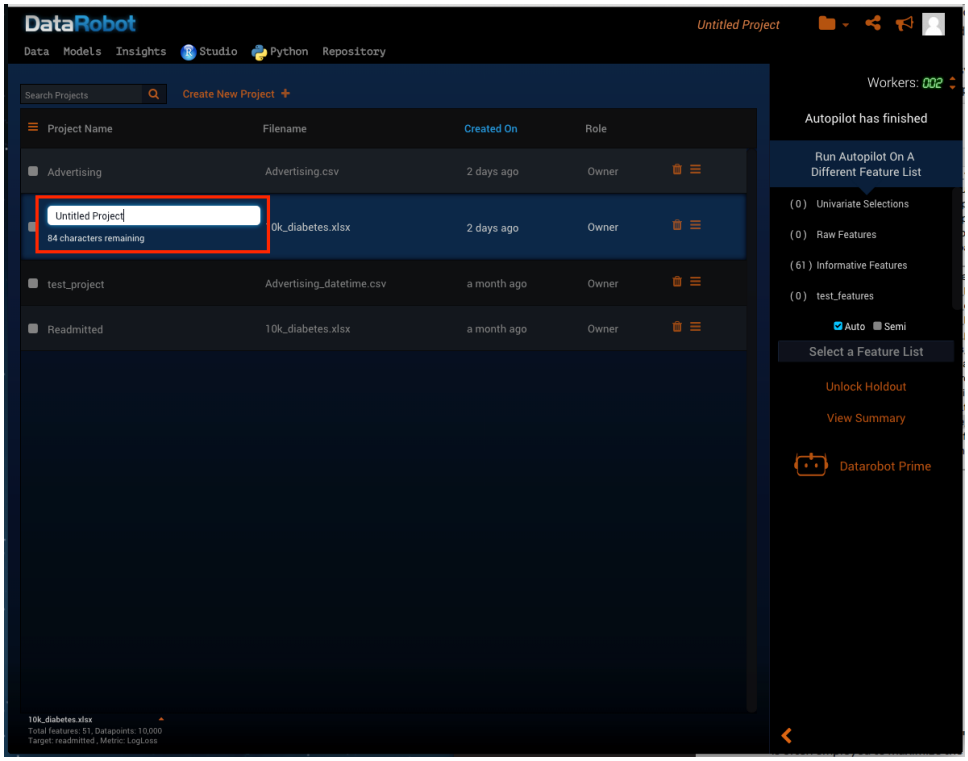
1. Click the folder icon in the menu at the top right of the page, then select **Manage Projects**.



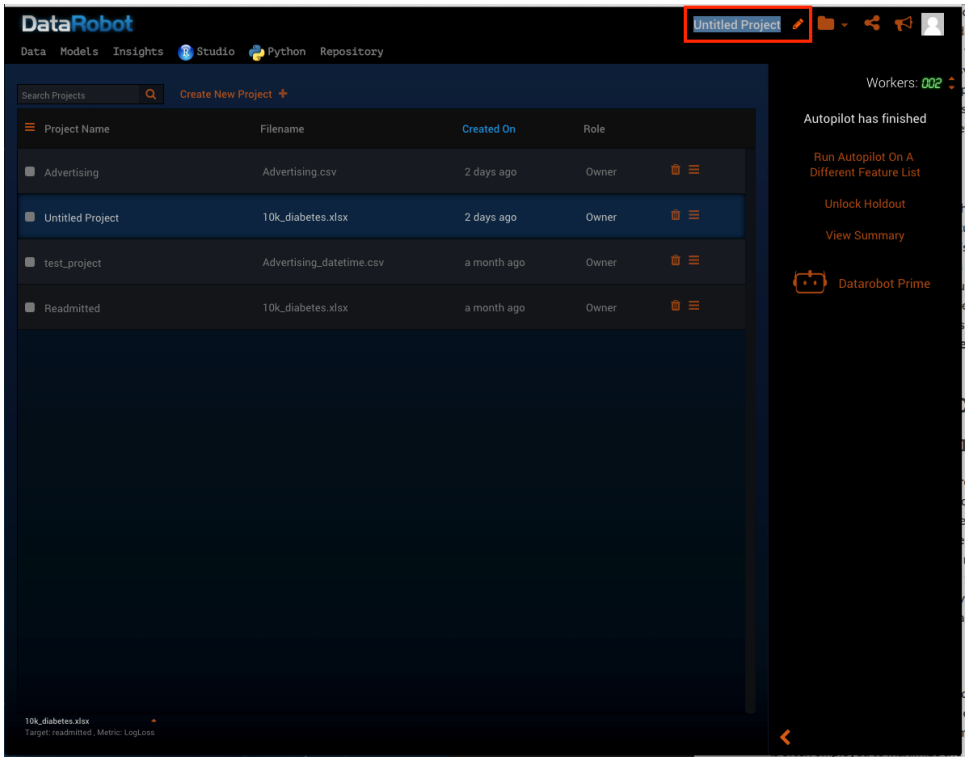
2. Click the project menu icon, then select **Rename Project**.



3. Type in a new project name.

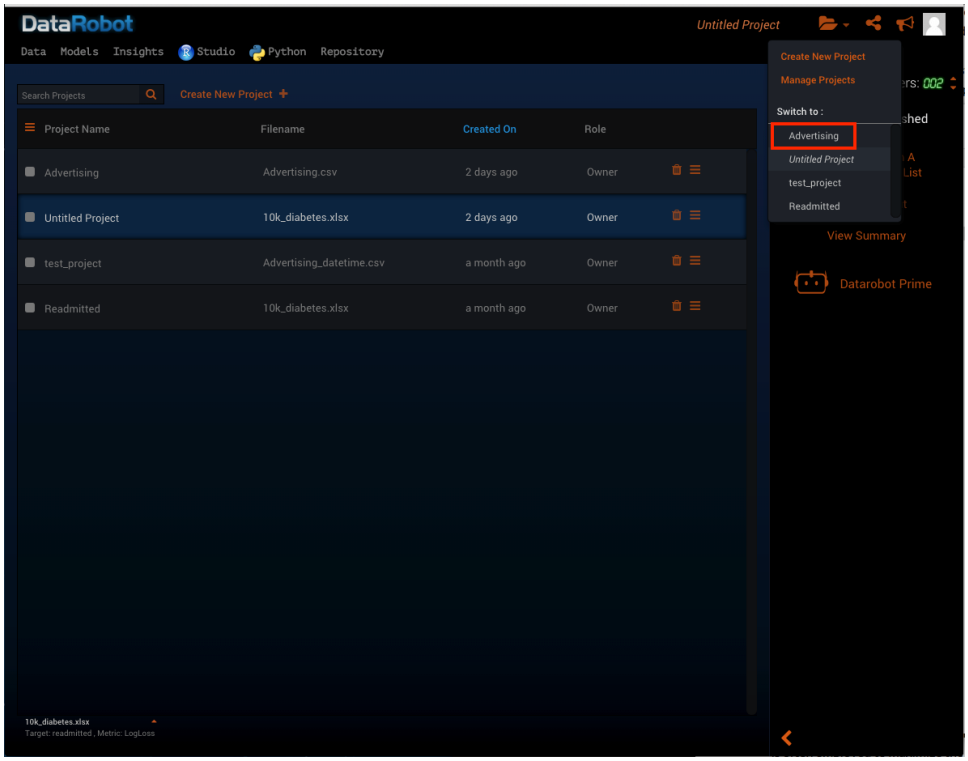


4. You can also rename a project by clicking the edit icon or the project name at the top of the page and entering a new project name.



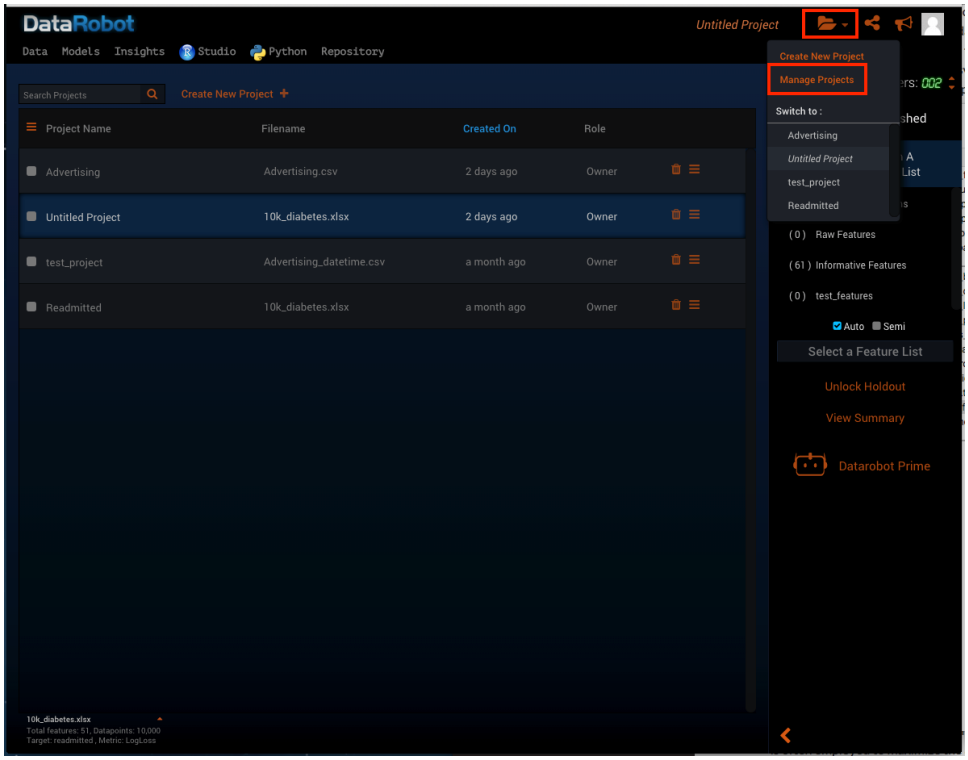
Switch to a Project

To switch to a different project, click the folder icon in the menu at the top right of the page, then select one of the projects listed under **Switch to:**.

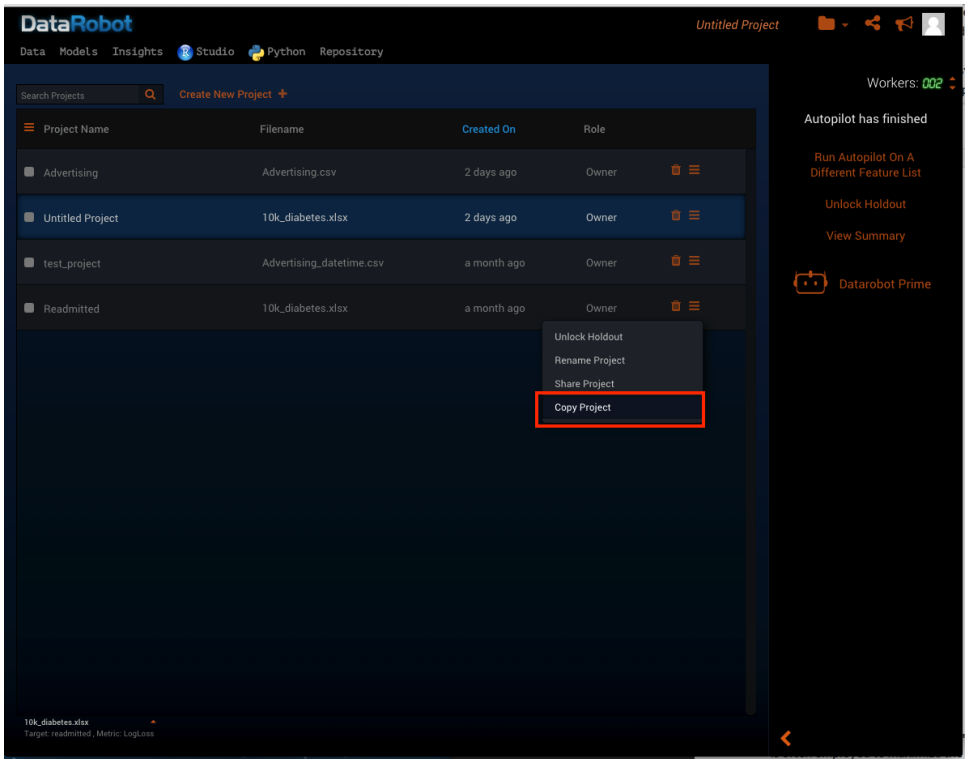


Copy a Project

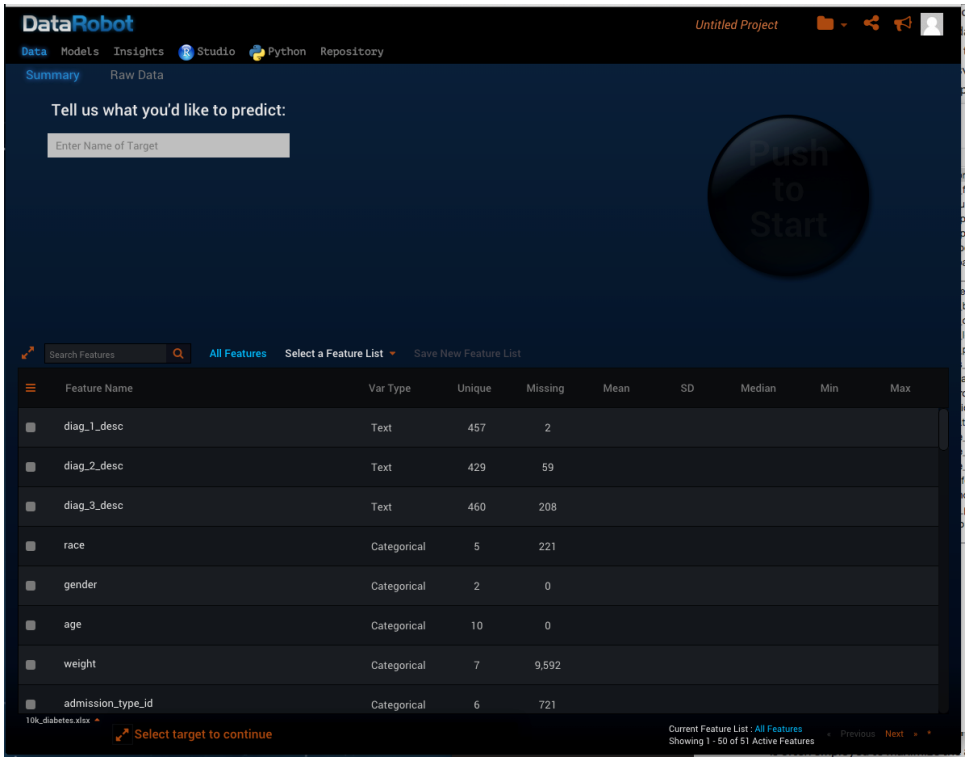
1. Click the folder icon in the menu at the top right of the page, then select **Manage Projects**.



2. Click the project menu icon, then select **Copy Project.**

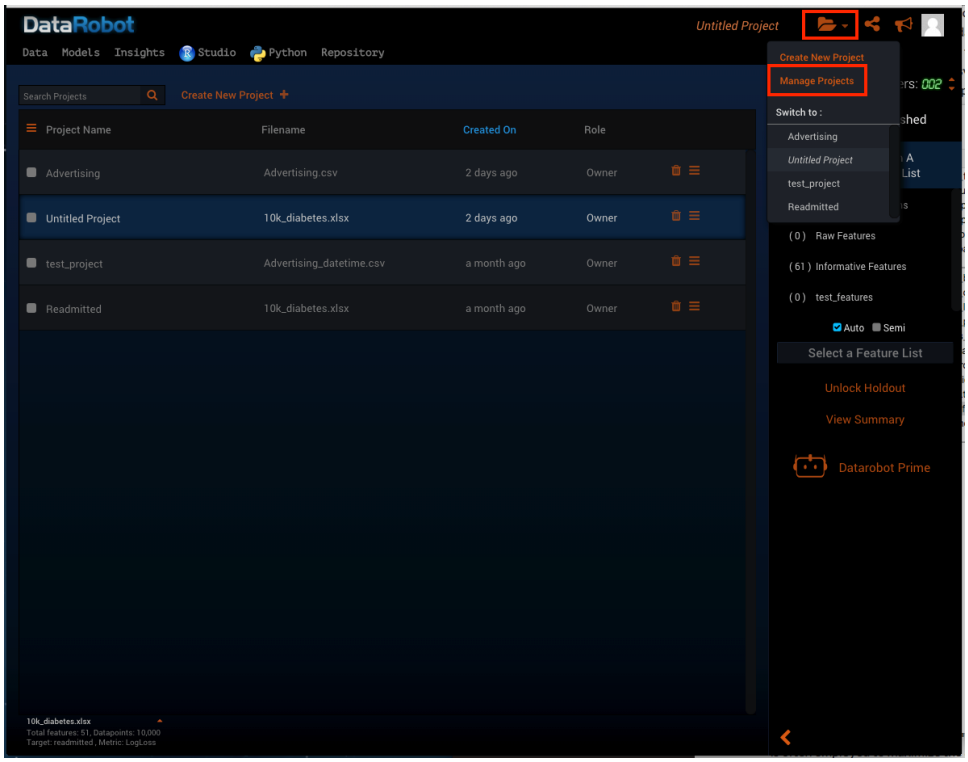


3. A new untitled project will be created using the dataset from the project, and the Data Summary page will be displayed.

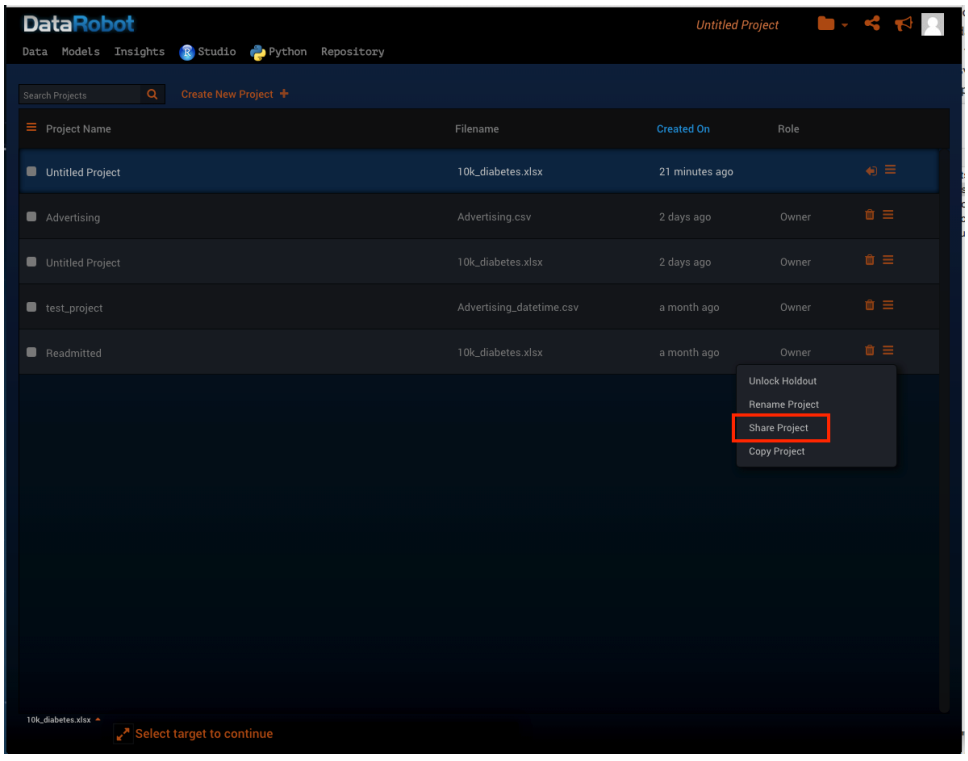


Share a Project

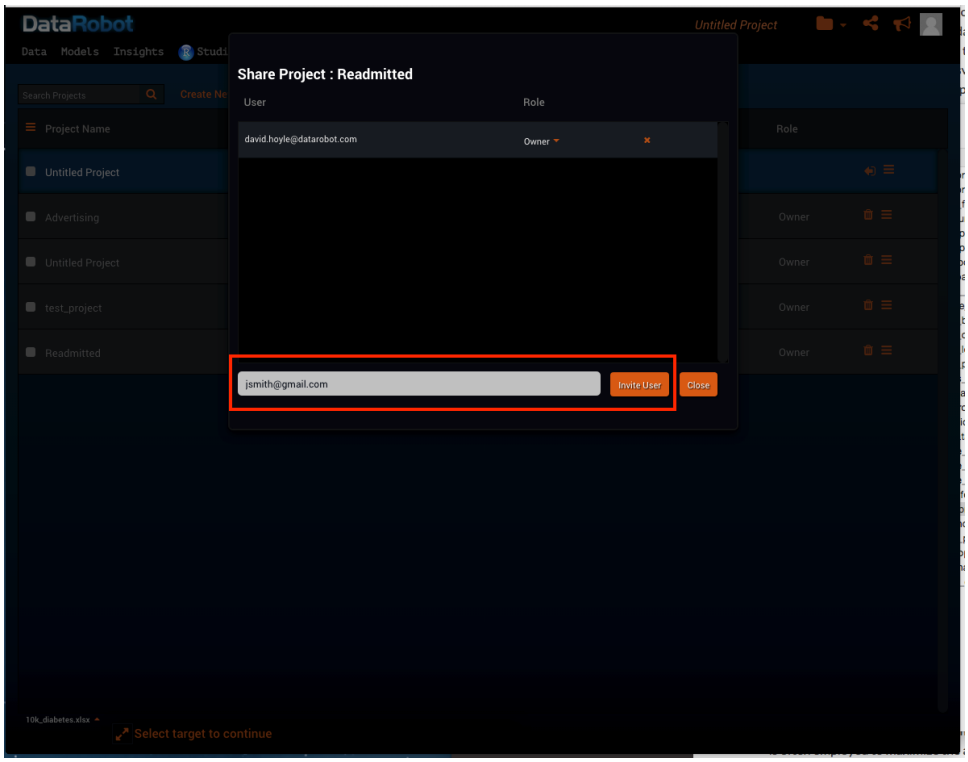
1. Click the folder icon in the menu at the top right of the page, then select **Manage Projects**.



2. Click the project menu icon, then select **Share Project**.



3. On the Share Project pop-up, type in the email address of the person you would like to share the project with, then click Invite User.

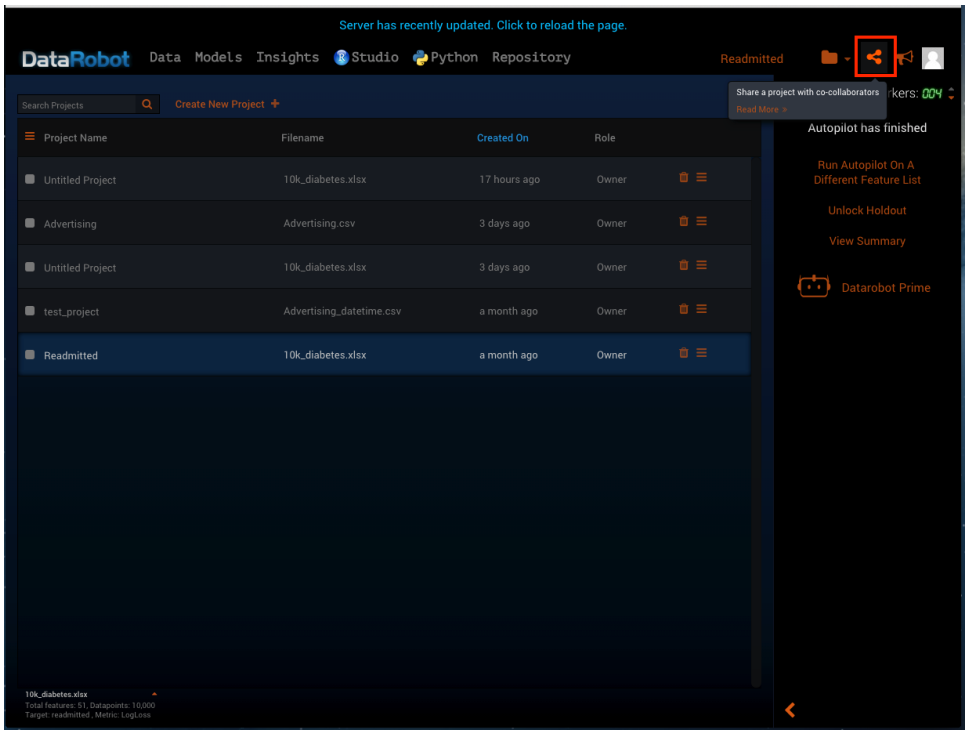


4. After you add the new user, they will receive an email invitation to join the project. You can use the drop-down in the Role column to change their role.

   The available roles are:

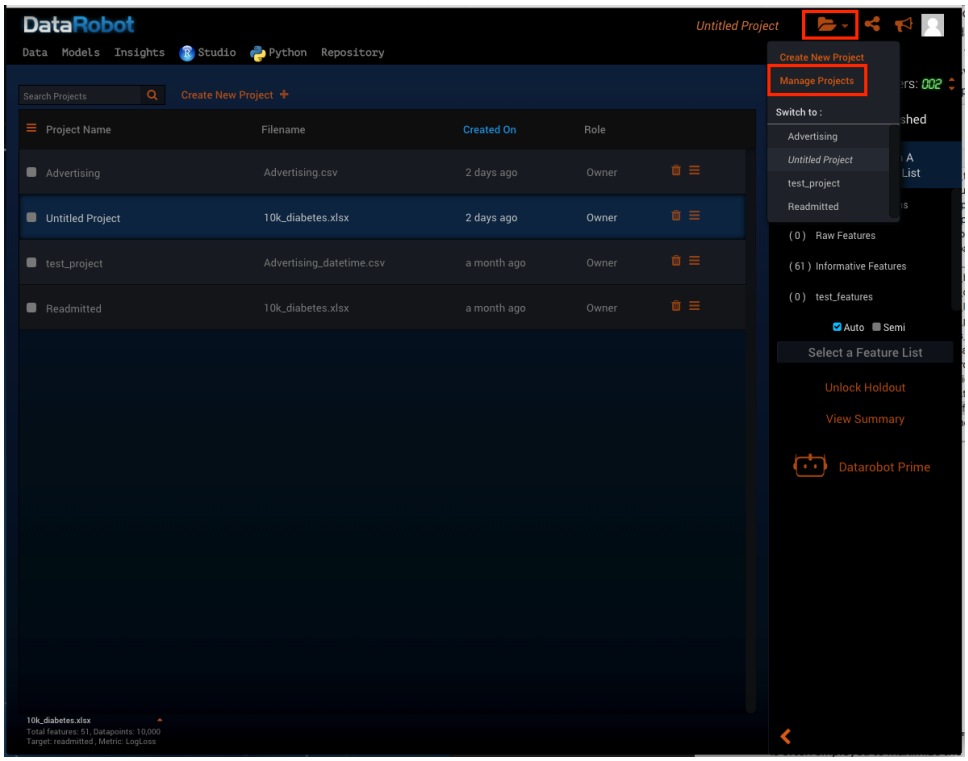
   - Observer -- can view the project, but cannot edit the project.
   - Data Scientist --
   - Admin --
   - Owner --

5. You can also share a project by clicking the Share icon in the top menu.

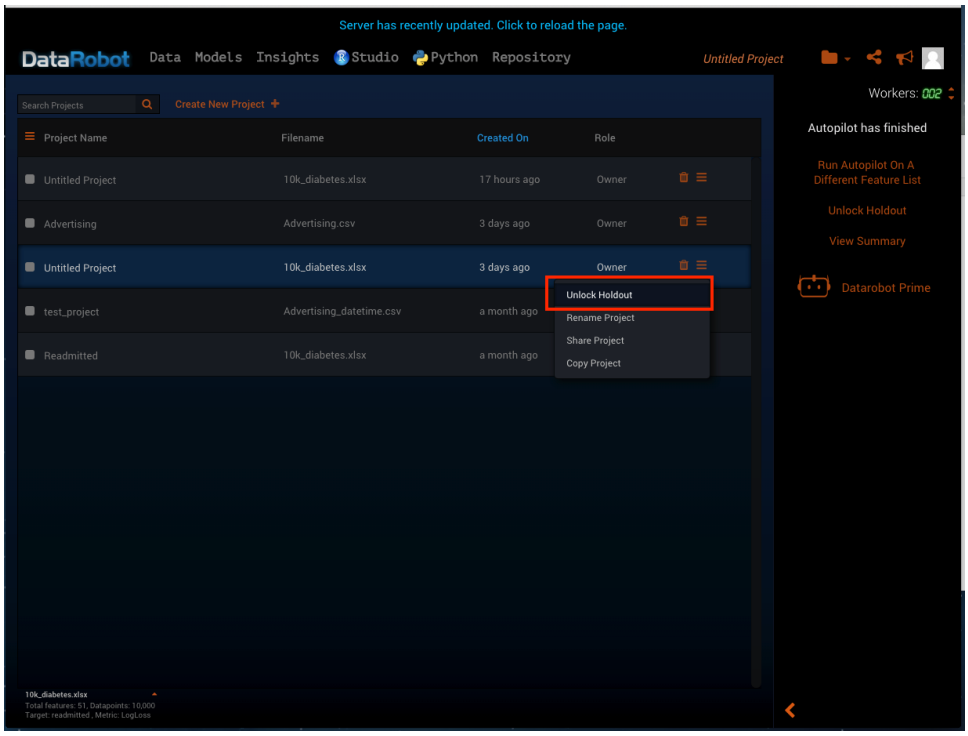


Unlock Holdout

1. Click the folder icon in the menu at the top right of the page, then select **Manage Projects**.

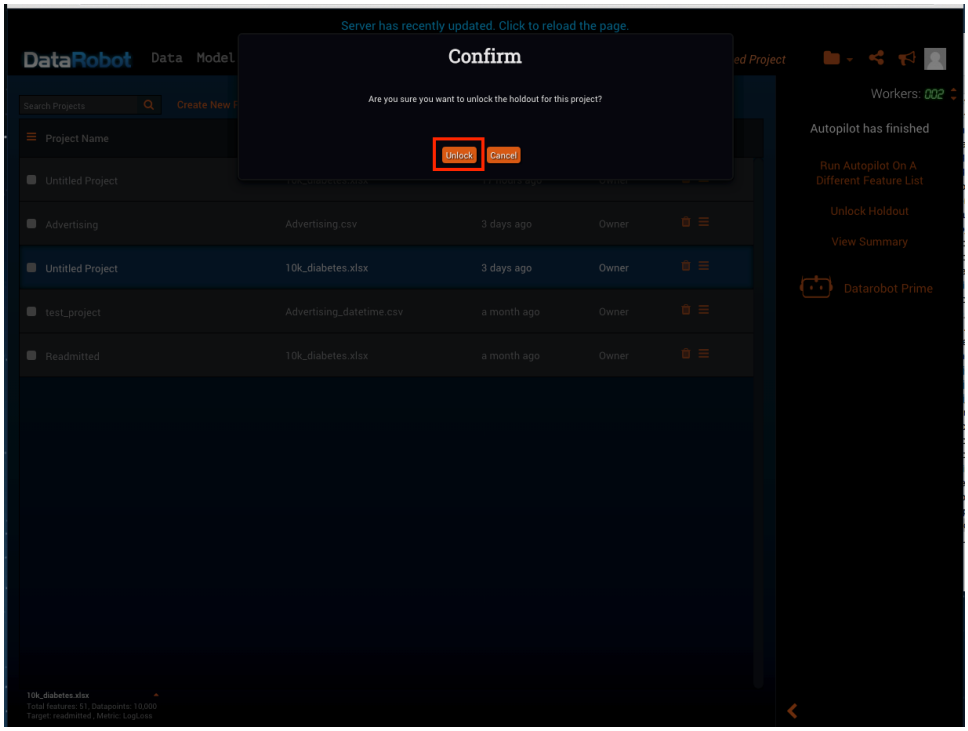


2. Click the project menu icon, then select **Unlock Holdout.**

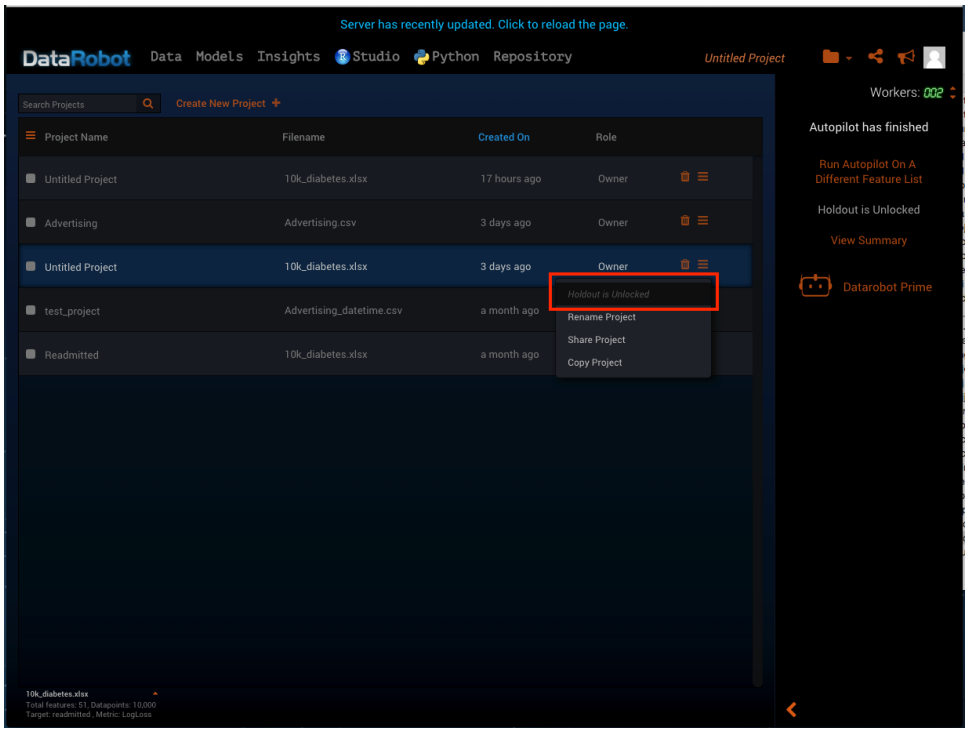


3. On the Confirm pop-up, click **Unlock**.

Note: Once holdout has been unlocked for a project, it cannot be re-locked.



4. Holdout is unlocked and the label on the project menu changes to "Holdout is Unlocked".


Server has recently updated. Click to reload the page.
DataRobot
Data
Models
Insights
Studio
Python
Repository
Untitled Project
Workers: 002
Search Projects
Create New Project
Project Name
Filename
Created On
Role
Untitled Project
10k_diabetes.xlsx
17 hours ago
Owner
Advertising
Advertising.csv
3 days ago
Owner
Untitled Project
10k_diabetes.xlsx
3 days ago
Owner
test_project
Advertising_datetime.csv
a month ago
Readmitted
10k_diabetes.xlsx
a month ago
Holdout is Unlocked
Rename Project
Share Project
Copy Project
Autopilot has finished
Run Autopilot On A Different Feature List
Holdout is Unlocked
View Summary
Datarobot Prime
10k_diabetes.xlsx
Total Features: 51, Datapoints: 10,000
Target: readmitted , Metric: LogLoss

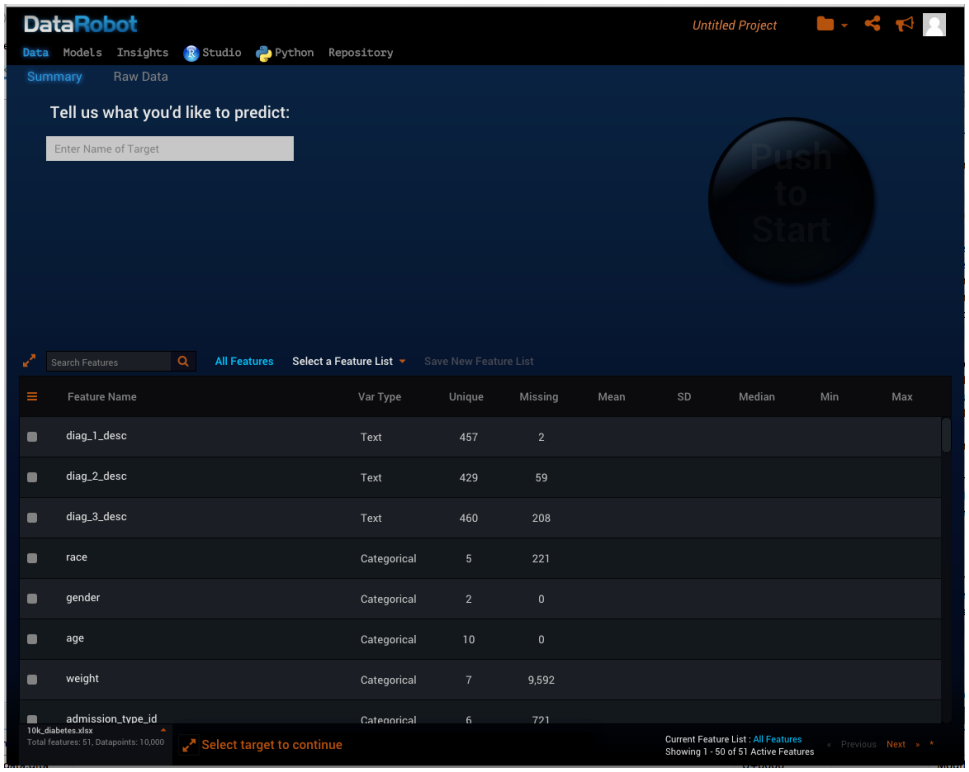
Working with Datasets

Each DataRobot project has one dataset that is used as the source from which to train models. You can use the Data page to view dataset features, create feature lists containing subsets of the dataset, and create derived features.

Viewing Data

You can use the Data page to view dataset features and create feature lists, and then select a target feature and begin modeling.

The Summary page appears after you create a DataRobot project.



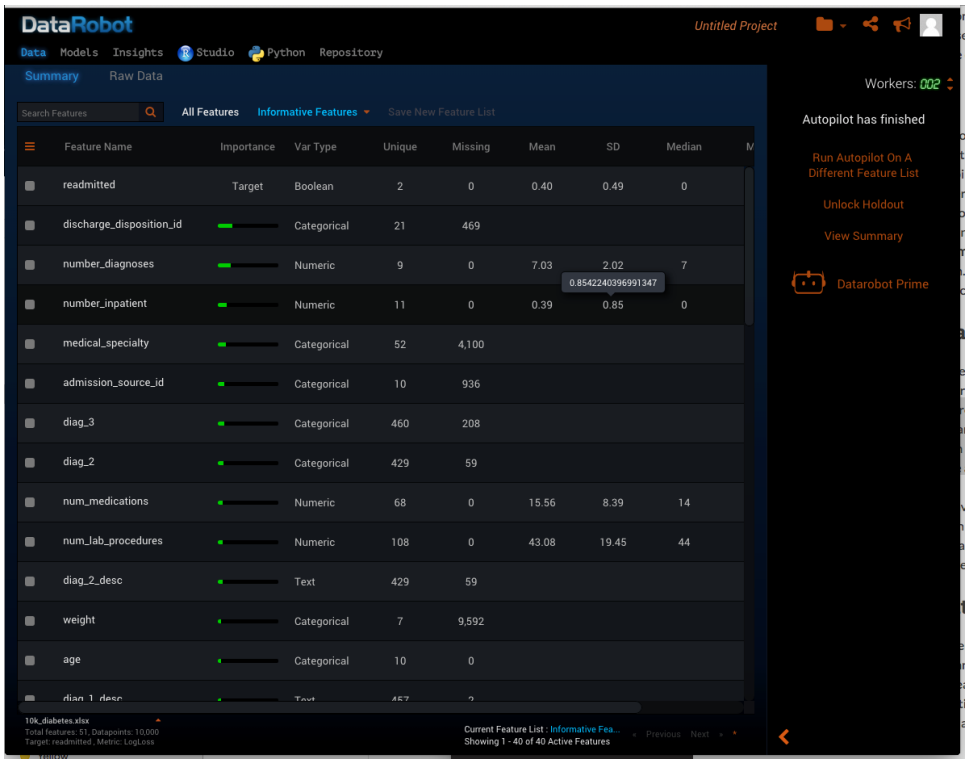
Note:

After you select a target feature and begin modeling, DataRobot analyzes the data and presents this information on the Data Summary page. Until then, the information displayed on the Summary page is somewhat limited. The following information assumes that you have selected a target feature and started the modeling process.

After you select a target feature and begin modeling, DataRobot analyzes the data and sorts the features on the Summary page in order of their importance in predicting the target variable.

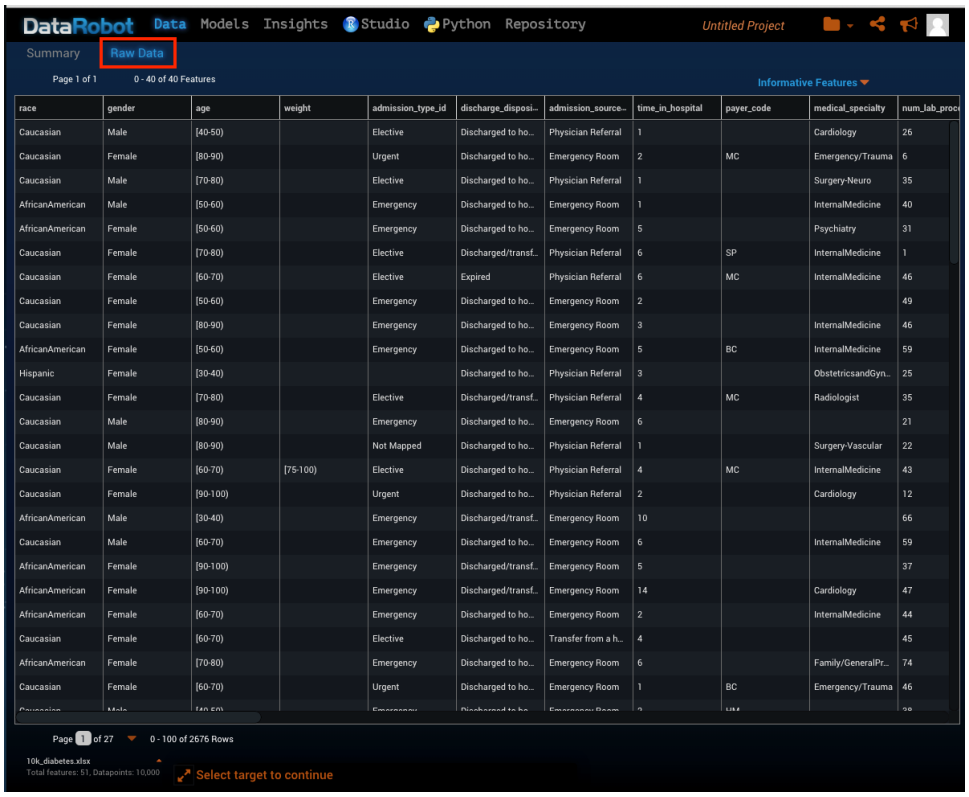
The green bar next to each feature indicates its relative importance. Importance is calculated using an algorithm that measures the information content of the variable, and this calculation is done independently for each feature in the dataset. Importance is scaled such that 1 is equivalent to a perfect model, and 0 is equivalent to a null model, so it can be interpreted in the same manner as an R-squared score.

The following image shows the Data Summary page with features rated by importance. The page also lists unique and missing values, mean, median, standard deviation, and minimum and maximum values.



Click a feature to display a graph of the mean target response plotted against the frequency distribution of the variable. To display a histogram of frequent values, click **Frequent Values**.

Click **Raw Data** to view the raw data table.



Dataset Guidelines

The following data file formats are supported by DataRobot.

- .csv
- .xls
- .xlsx

Those formats can also be compressed or archived with the following formats:

- .bz2
- .zip
- .gz
- .tar
- .tgz
- .tar.gz

The file must be a comma-separated file with a header that matches the number of data columns. Each row must have the same number of fields, some of which may be blank. Ensure that the data is in ASCII, UTF-8, or WINDOWS-1252 CSV file format, and that the data file does not have any extraneous characters or escape sequences from URLs.

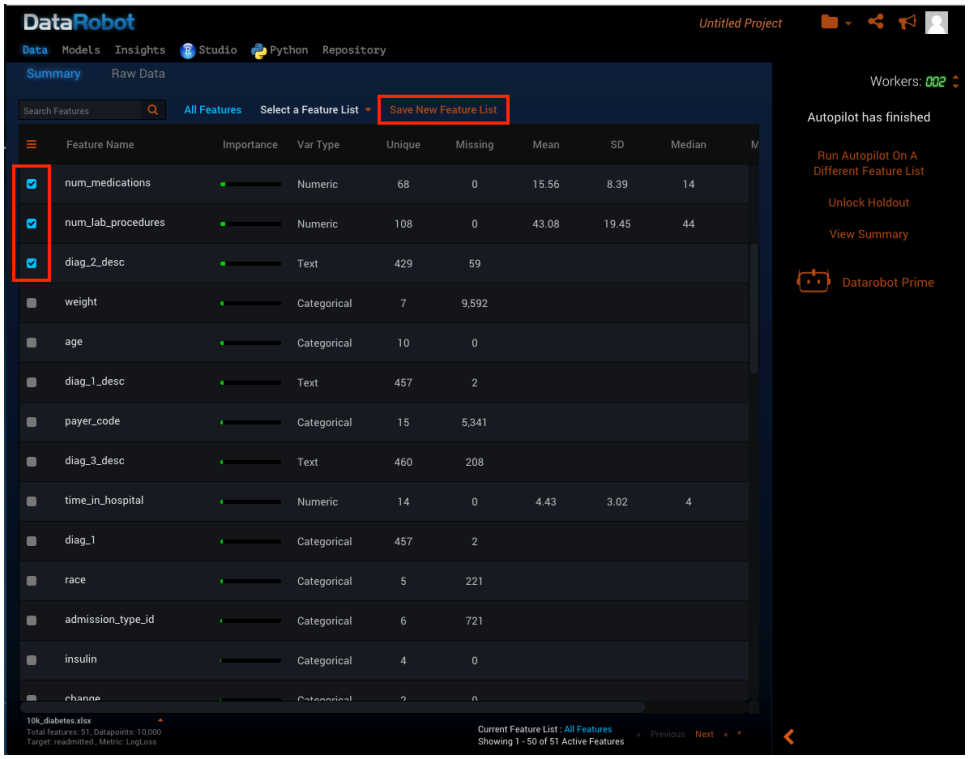
Note: At this time it is safer to upload files smaller than 524 MB, even if they are compressed. If you have a larger dataset, you can submit a subset of the data and contact DataRobot for information on working with large datasets.

Working with Feature Lists

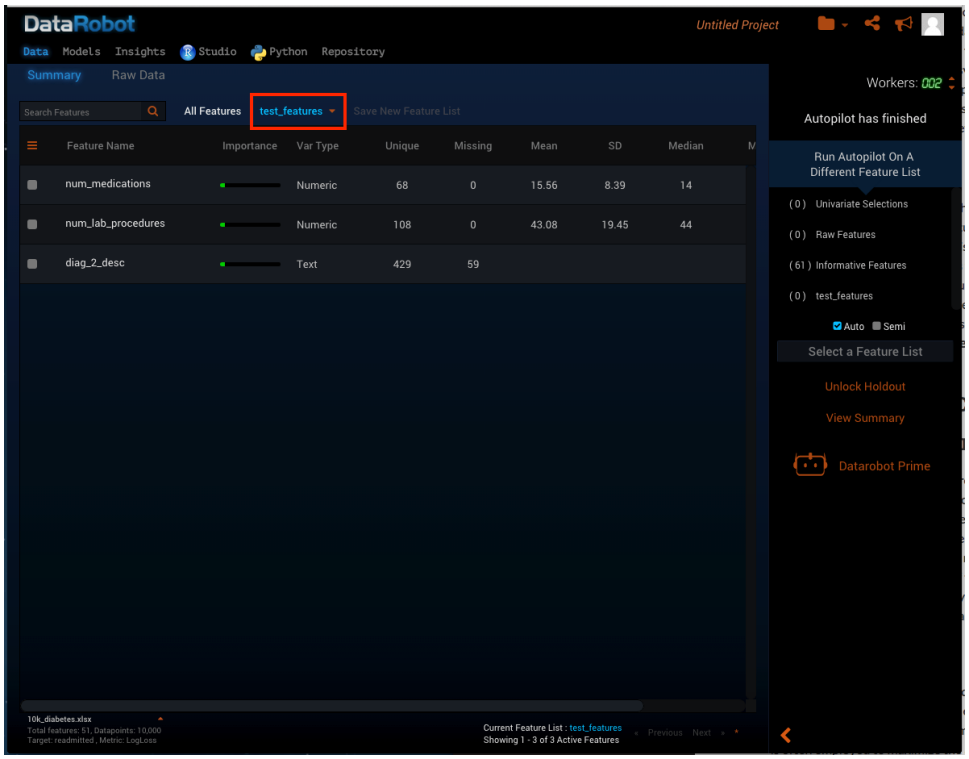
By default DataRobot uses all of the features (also referred to as Raw Features) in the dataset for the prediction task. You can create a feature list that contains a subset of the features, and train the model on this feature set to see if it yields a better model.

Create a Featurelist

To create a new feature list, use the check boxes on the Data Summary page to select a set of features, then click **Save New Feature List**.



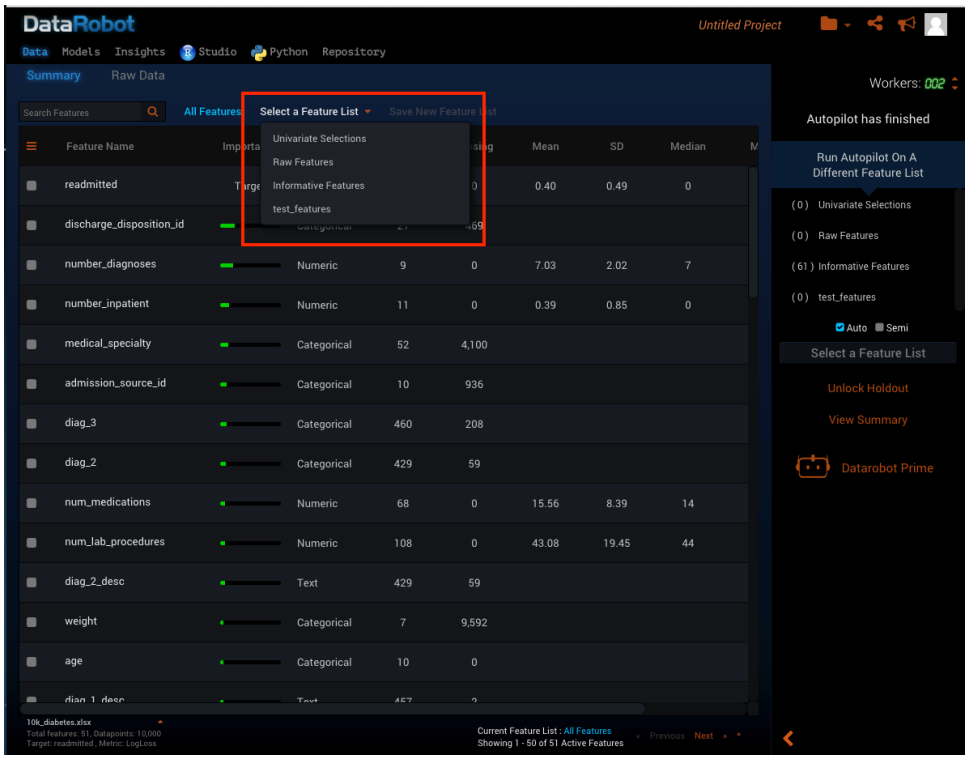
The new feature list appears on the Data Summary page, and its name replaces the Select a Feature List label.



You can use the **Select a Feature List** menu to change the set of features used to train models. DataRobot provides the following feature lists:

- Raw Features -- All of the features (the default selection).
- Univariate Selections -- Features that are determined by the ACE function to have a relationship with the target.
- Informative Features -- Features that pass a reasonableness check that determines whether or not they contain useful information. For example, a column of ones or a duplicate of an existing column would be excluded from the informative features list.

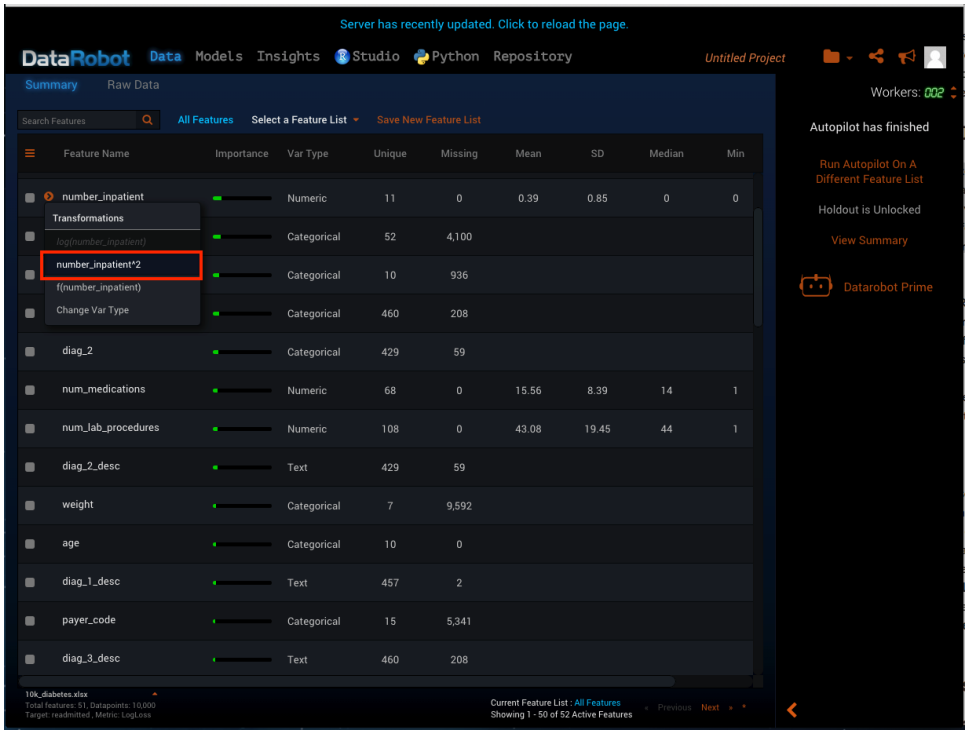
including **Univariate Selections** and **Informative Features**. Informative Features are those features that have been sorted at the top of the Features list.



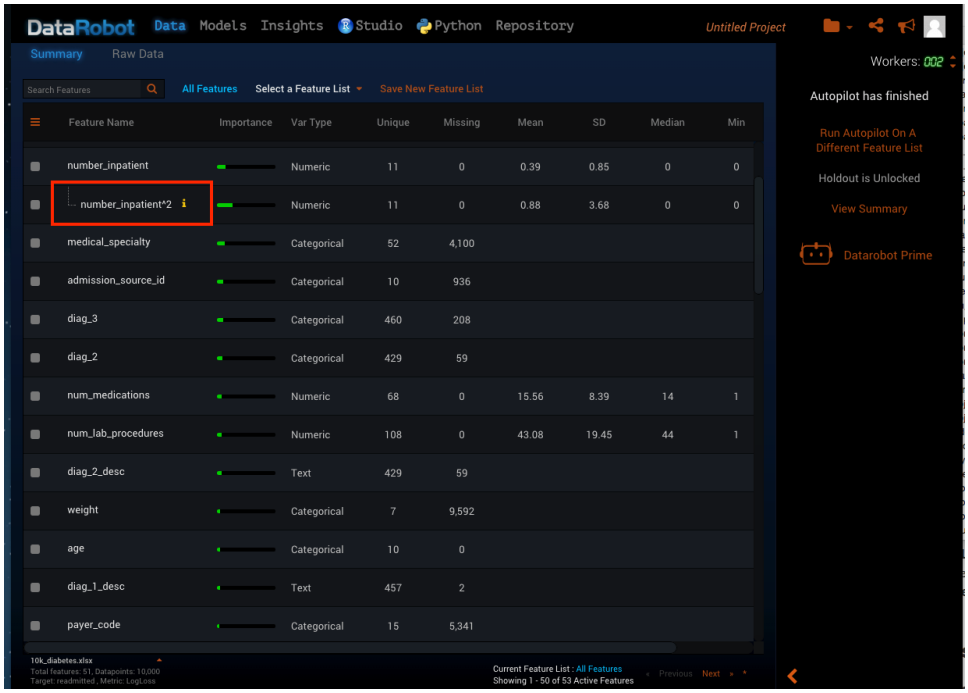
Creating Derived Features

DataRobot provides access to several different transformations that you can apply to your data. Currently, that includes squaring and taking the natural logarithm of numeric data, when appropriate.

1. To create a derived feature, click the orange arrow at the left of the feature name, then select a transformation.



2. The transformed feature appears under the original feature. It can be included in any new feature lists, and can also be used for modeling.



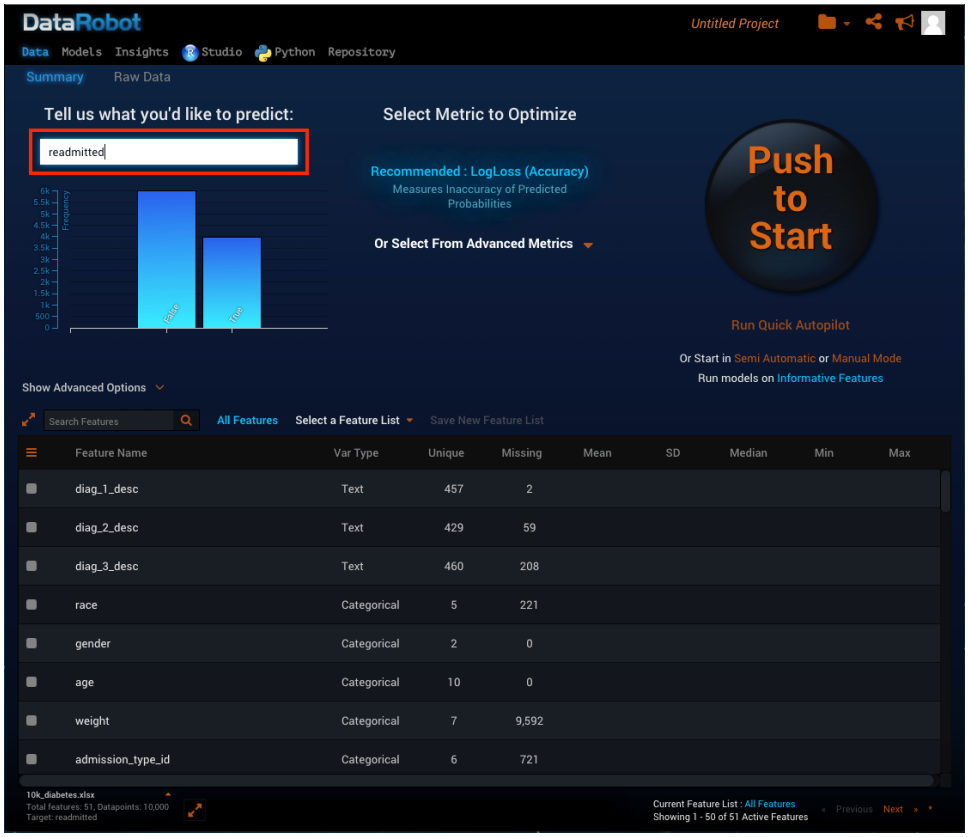
Running Models

After you create a DataRobot project, you can select a target feature and modeling options, and then start the modeling process. You can also add new models, create blended models, and run selected models.

Select a Target Feature

After you create a project, you can select a target feature and an optimization metric and start modeling. The target feature is the name of the column in the dataset you would like to predict. The optimization metric is the metric software used to optimize the models.

On the Data Summary page, type in the name of the target feature you would like to predict.



Select an Optimization Metric

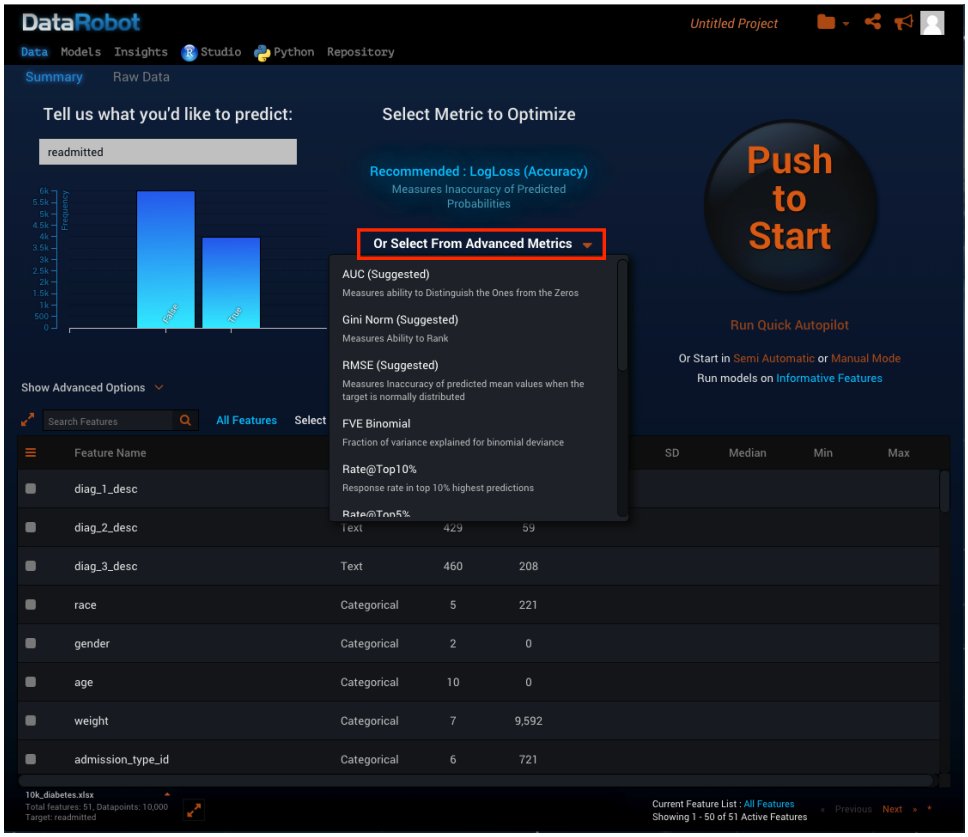
After you create a project, you can select a target feature and an optimization metric and start modeling. The target feature is the name of the column in the dataset you would like to predict. The optimization metric is the metric software used to optimize the models.

After you type in a target feature, a recommended optimization metric is automatically selected based on the modeling task.

- If the selected target has only two unique values, the assumption is made that the task is a classification, and a classification metric will be recommended. Examples of recommended classification methods include LogLoss (if it is necessary to calculate a probability for each class), and Gini and AUC when it is necessary to sort records in order of ranking.

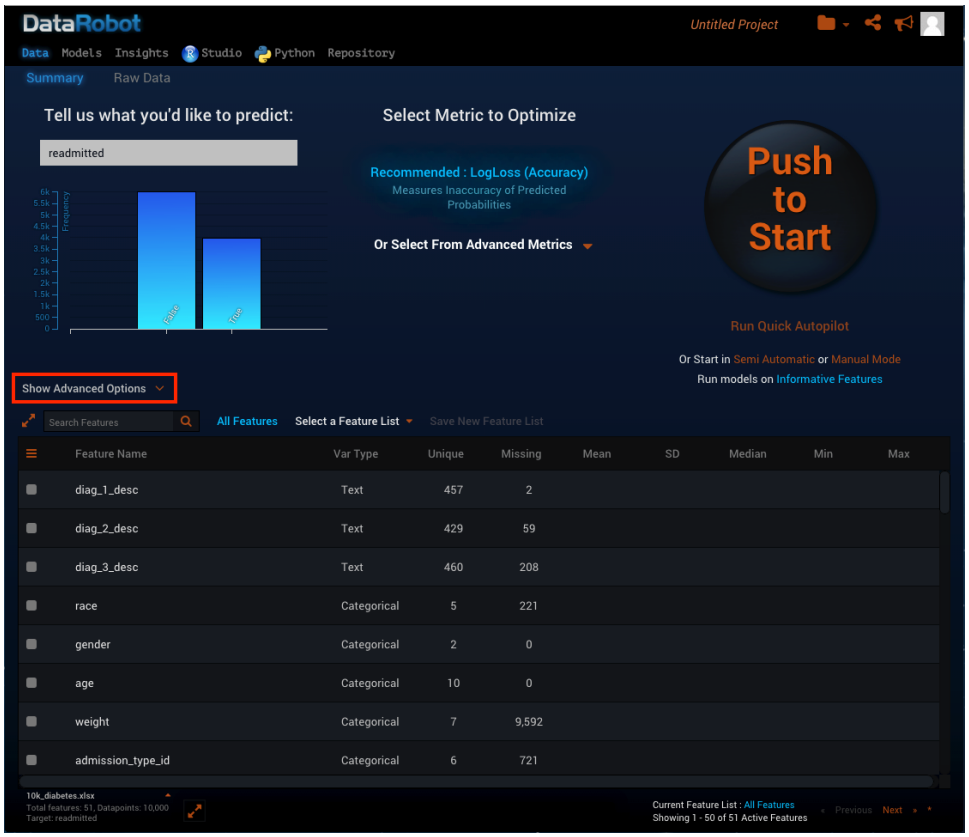
- Otherwise the assumption is made that the selected target represents a regression task. The most popular metrics for regression are RMSE (Root Mean Square Error) and MAD (Mean Absolute Deviation).

  If you would like to override the recommended optimization metric, click **Select from Advanced Metrics**, then select an alternate metric.

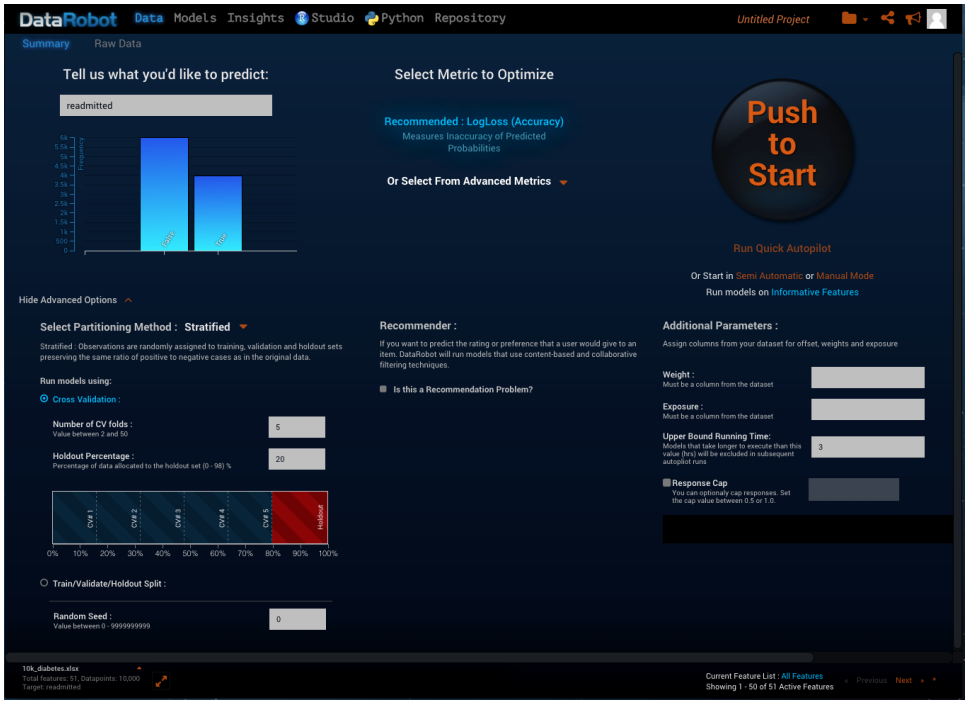


Set Advanced Modeling Parameters

To set advanced modeling parameters, click **Show Advanced Options** on the Data Summary page.



The advanced modeling options appear.



You can specify the following advanced options:

Partitioning Method

Random

Description: Observations are randomly assigned to the training, validation, and holdout sets.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.
- Random Seed -- specify a positive integer value for the random seed.

Partition Column

Description: A column in the data file is used to either specify a train/validate/holdout split or the folds/holdout to be used for cross-validation.

Modeling Options:

- Cross-Validation -- you must specify a value from the selected partition column that will specify the holdout set.
- Training/Validation/Holdout Split -- you must specify training, validation, and holdout set values from the selected partition column.

Group

Description: One or more columns can be selected, and each combination of these values is guaranteed to be in the same training or test set as other matching values.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.

Date

Description: Observations in the holdout set come after observations in the validation set chronologically, and observations in the validation set come after those in the training set. For this method you specify the Date/Time column in the dataset, which is used to create a Train/Validate/Holdout split with all rows in the test set occurring later than all rows in the training set, and all rows in holdout occurring later than the test set.

Modeling Options:

- Validation Percentage -- percentage of data allocated to the validation set.
- Holdout Percentage -- percentage of data allocated to the holdout set.

Stratified

Description: Observations are randomly assigned to training, validation, and holdout sets, preserving the same ratio of positive to negative cases as in the original data.

Modeling Options:

- Cross-Validation -- specify the number of folds and holdout percentage.
- Training/Validation/Holdout Split -- specify the percentages for training, validation, and holdout.

Recommender

Select the **Is this a Recommendation Problem?** check box if you would like to predict the rating or preference that a user would give to an item.

If you select this option, you must specify the two columns that contain the user and item identifiers.

A typical example for recommender systems is a product recommendation on an e-commerce site where you would like to predict how highly a user might rate a product. Predicted ratings are then used for content discovery on the site.

For recommendation problems, DataRobot runs models that use content-based and collaborative filtering techniques.

- Content-based models take into account any available user or item features and create a profile for each user (key words, price range, product types, brands, etc.).
- Collaborative filtering techniques are content-agnostic -- they look solely at the latent structure in the user-item ratings in order to make predictions.

DataRobot also runs hybrid models that combine both content-based and collaborative filtering techniques, as well as combinations of models (blended models).

Additional Parameters

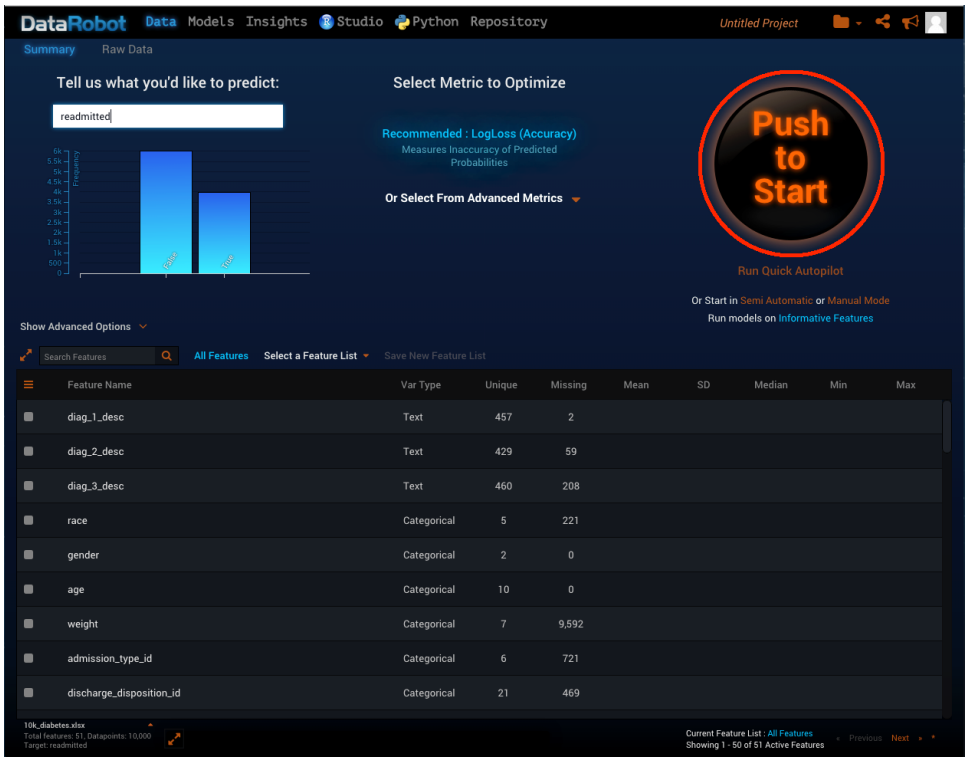
- Weight -- must be the name of a column in the dataset.
- Exposure -- must be the name of a column in the dataset.
- Upper Bound Running Time -- models that take longer to execute than this value (hrs) are excluded in subsequent autopilot runs.
- Response Cap -- The Response Cap limits the maximum value of the response (target) to a percentage of the original values. The value must be between 0.5 and 1.0 (50-100%).

Start Modeling

Start Modeling

After you have selected a target feature and an optimization metric, you can start modeling. The modeling process finds the best predictive models for the target feature.

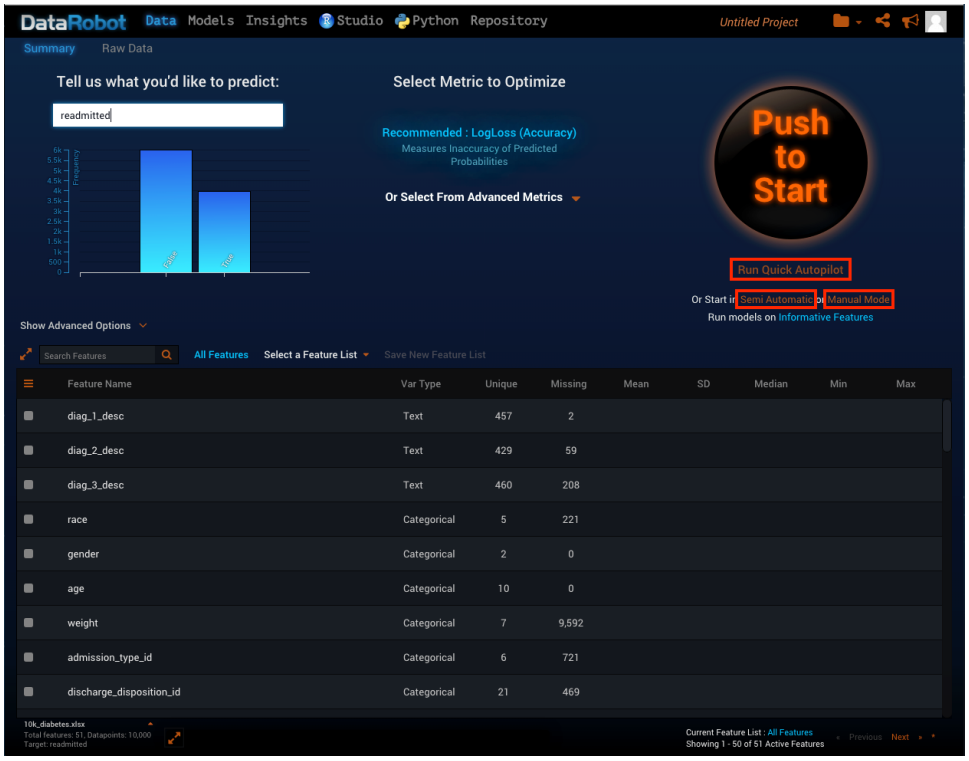
To start modeling, click the **Push to Start** button.



Autopilot Modes

The modeling process runs in Autopilot (fully automatic) mode by default, which means that the DataRobot platform will automatically select the best predictive models for the specified target feature.

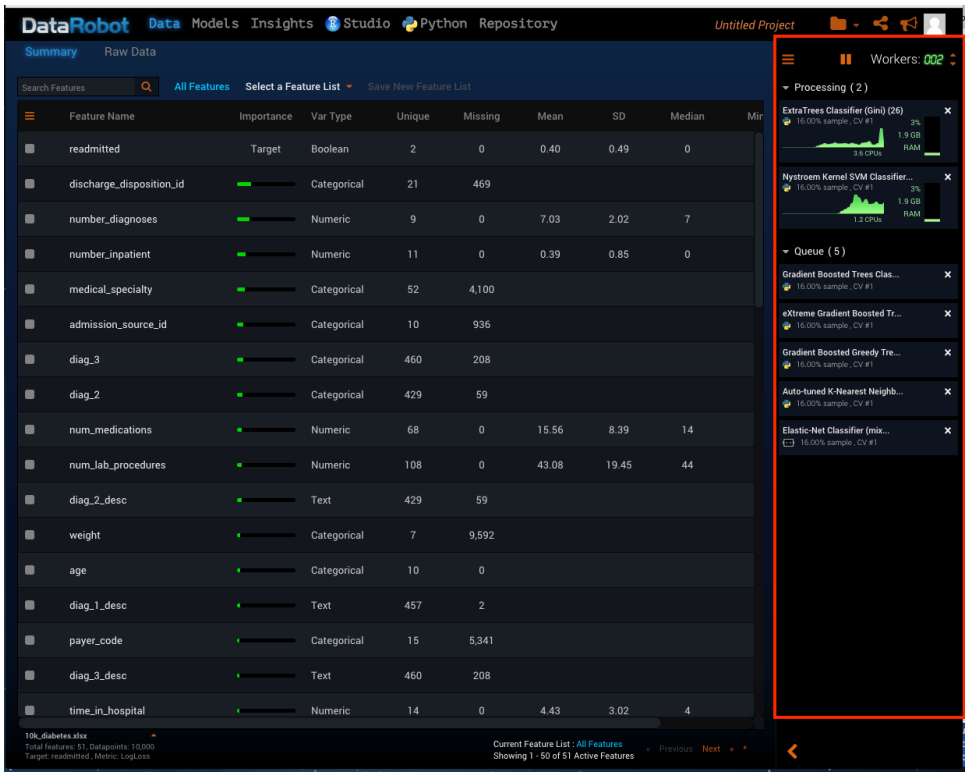
For more control over which models to run, you can select **Run Quick Autopilot**, **Semi Automatic**, or **Manual Mode** under the **Push to Start** button.



- **Run Quick Autopilot** -- The Quick Autopilot runs a small subset of models consisting of the best models based on the specified target feature and performance metric.
- **Semi Automatic** -- In semi-automatic mode, the autopilot will pause after the first cross-validation set before submitting additional models to the job queue. This allows you to inspect the models and choose to either add additional models or delete models.
- **Manual Mode** -- Starting the autopilot in manual mode gives you full control over which models to execute. For example, you can choose a specific model from the Task Repository rather than running the models selected by default.

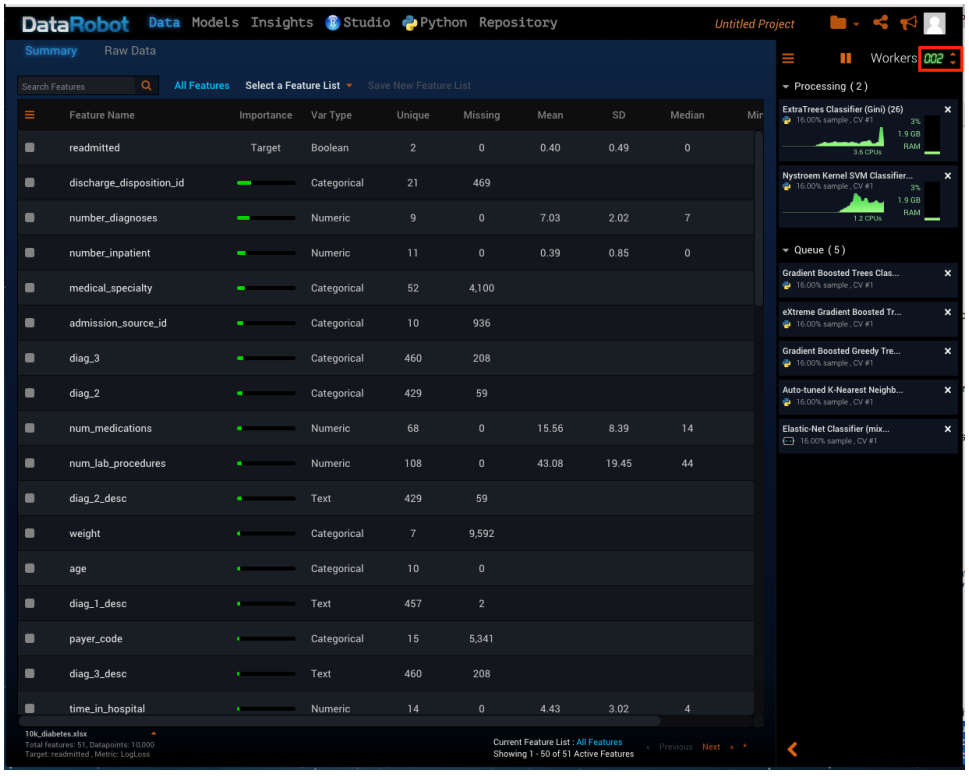
The Worker Queue

After you start the modeling process, running and pending jobs appear in the Worker Queue at the right of the page When a model is running, a graphical display of its CPU and RAM use will be displayed in the Worker Queue. The queue will execute a maximum fixed number of workers per project at any one time.



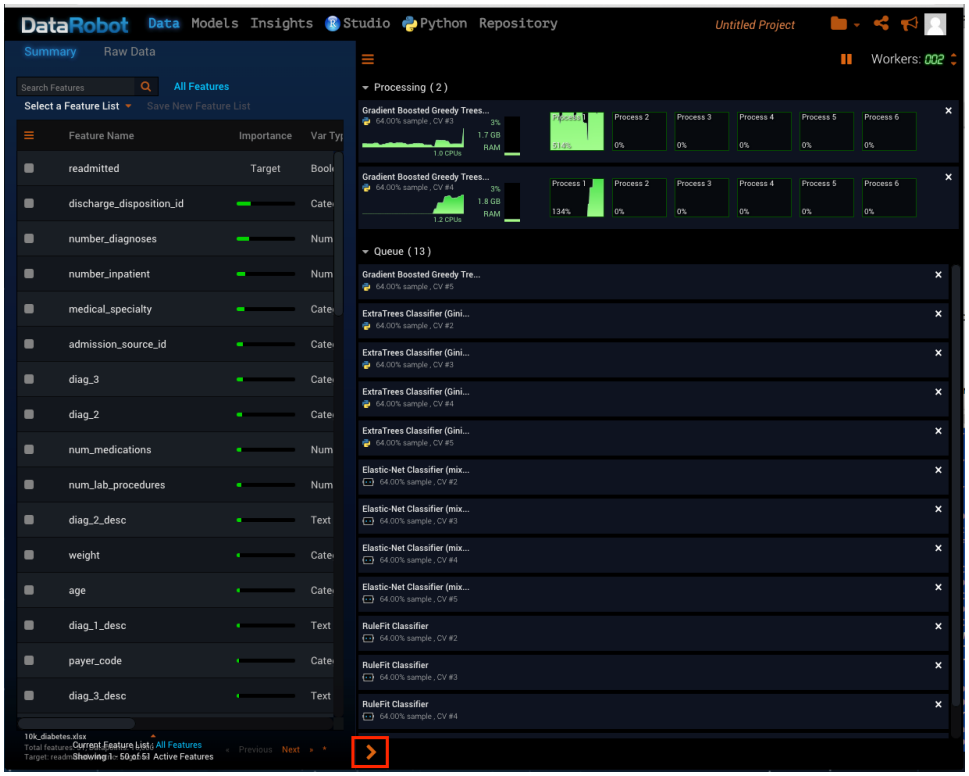
Adjusting the Number of Workers

You can adjust the maximum number of simultaneous workers by clicking the orange arrows at the right of the green numbers in the Workers field at the top of the Worker Queue.



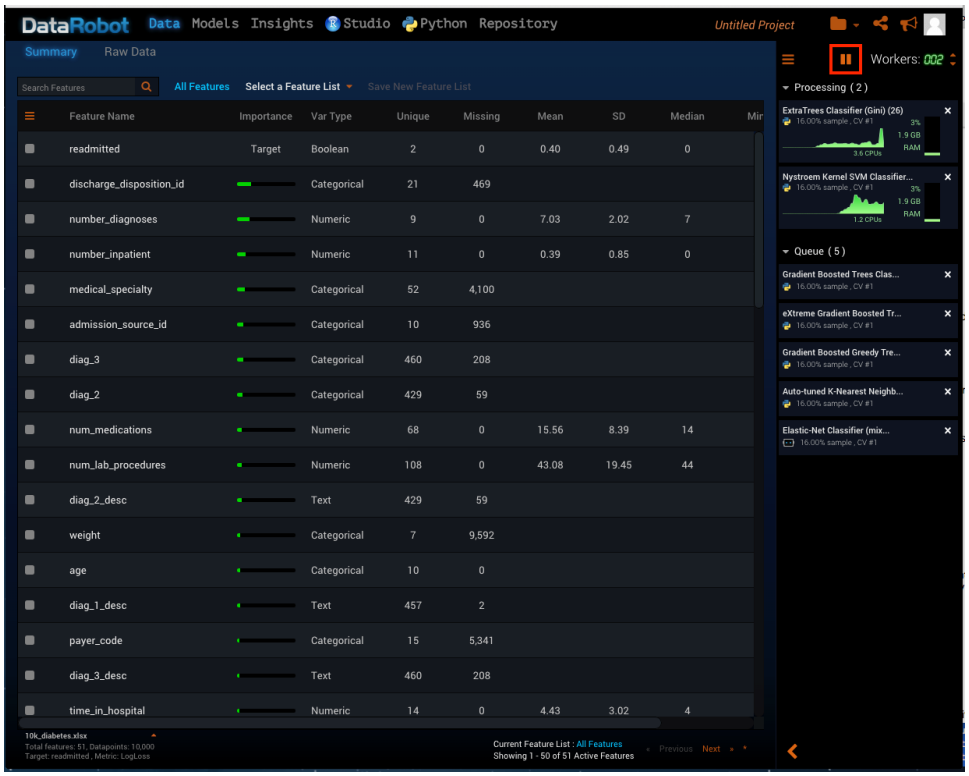
Viewing Worker Queue Details

To view more details about the running jobs, click the orange arrow at the bottom of the Worker Queue. Click the arrow again to hide queue details.



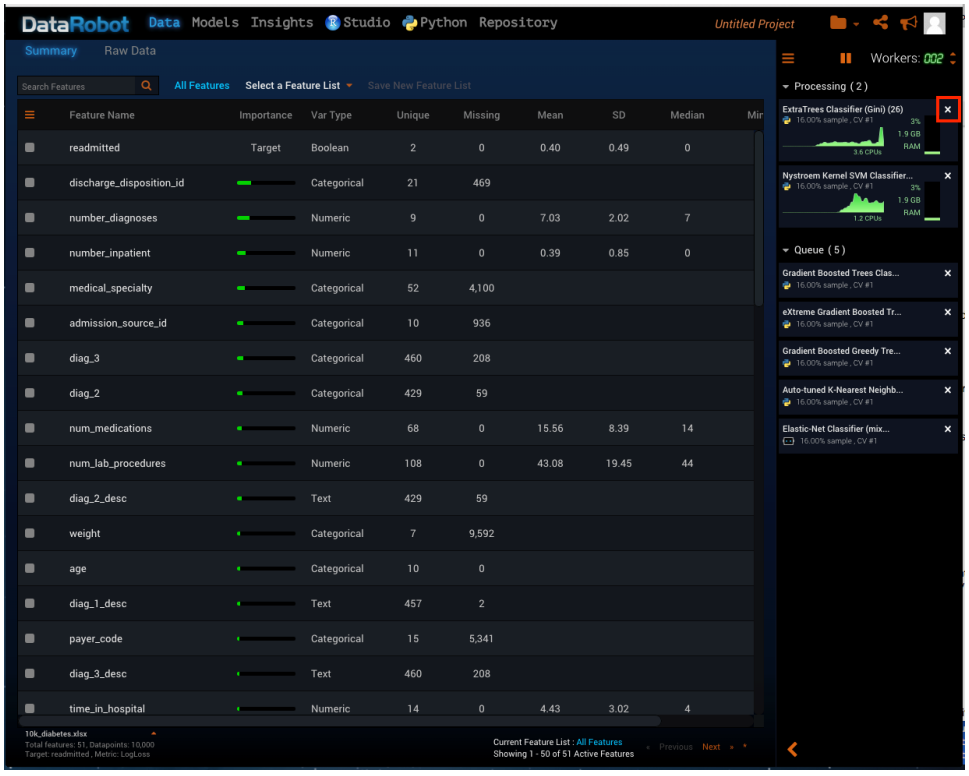
Pausing the Worker Queue

To pause the Worker Queue, click the Pause symbol at the top of the Worker Queue. After you pause the queue, the Pause symbol changes to a Play symbol (arrow). To resume running models, click the Play arrow.



Killing Workers

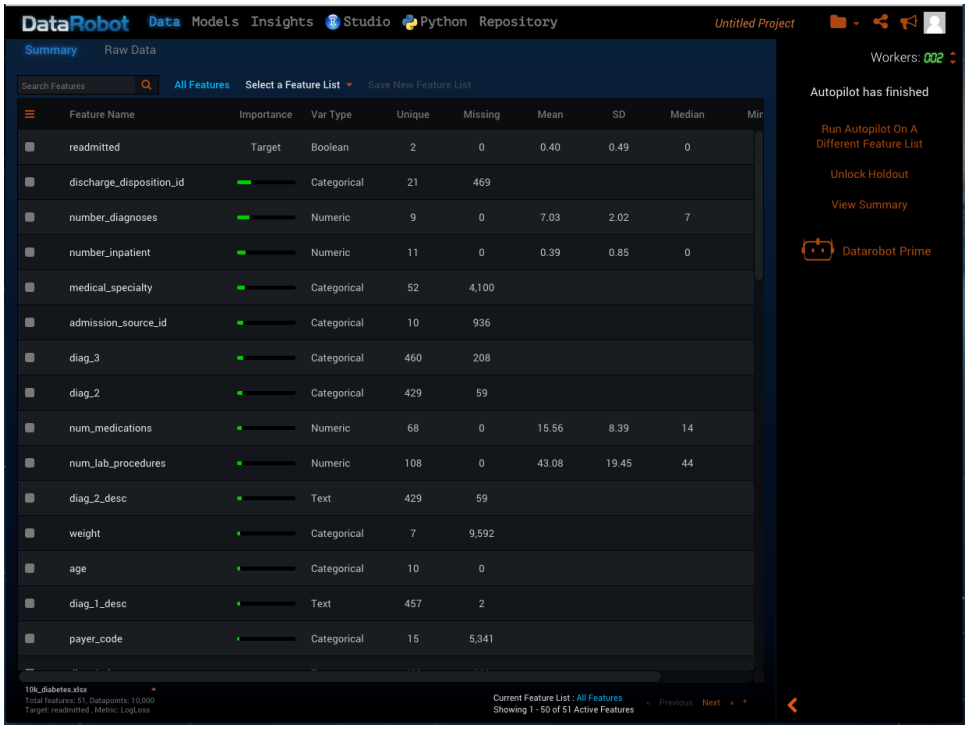
You can kill a worker by clicking the **X** next to the job name in the Worker Queue. If a worker fails for any reason, it will be listed under Errors at the bottom of the Worker Queue.



After modeling has started, you can click **Models** in the top navigation menu to view the Leaderboard. The Leaderboard is a list of models ranked by the chosen performance metric, with the best models at the top

of the list. The rankings keep changing until all models have finished running. You cannot view full details about a model until it finishes running.

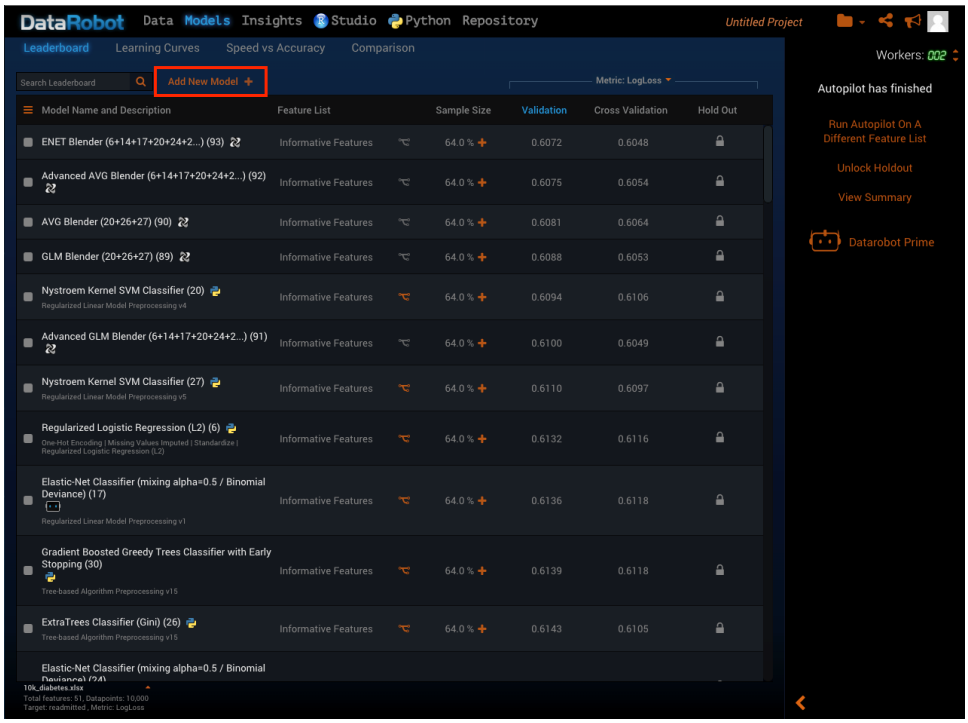
When all workers have finished running, the Worker Queue displays a message at the top of the queue and provide options to rerun the models a different feature list, or to unlock the holdout dataset and continue running workers.



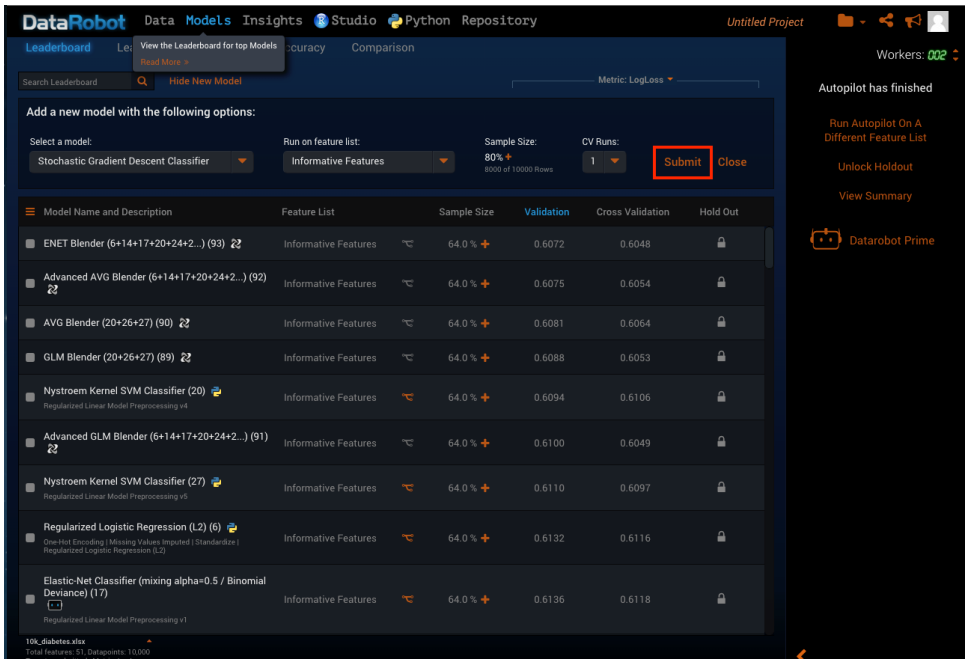
Add a New Model

Use the following steps to create a new model.

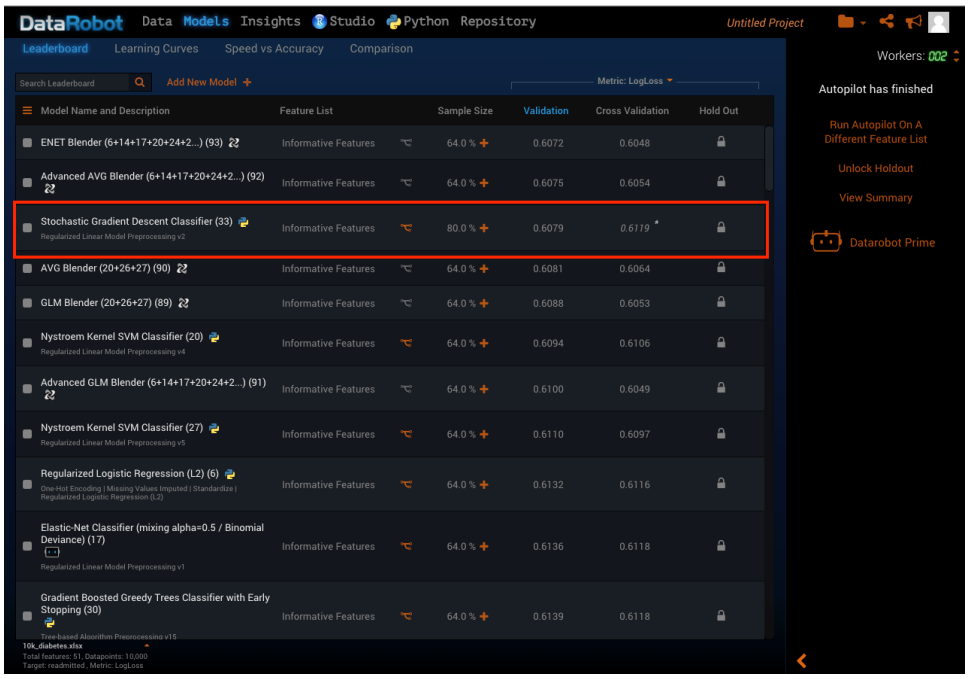
1. Click **Add New Model** at the top of the model Leaderboard.



2. Select a model type and a feature list, specify sample size and the number of cross-validation runs, then click **Submit**.



3. The new model appears in the list on the Leaderboard.

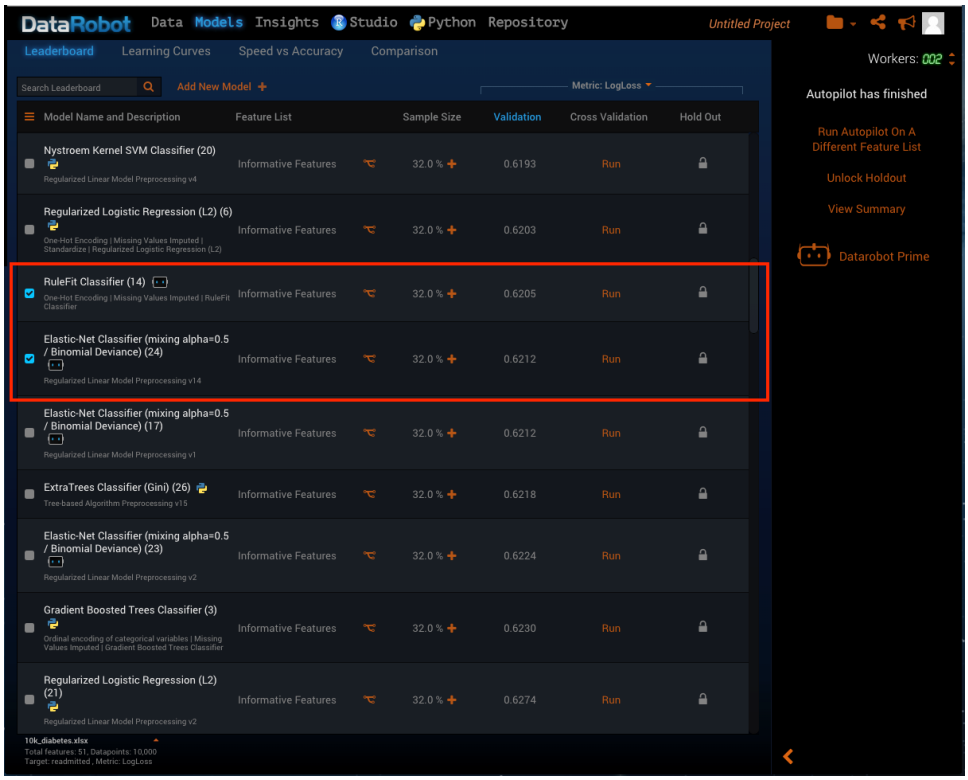


Create a Blended Model

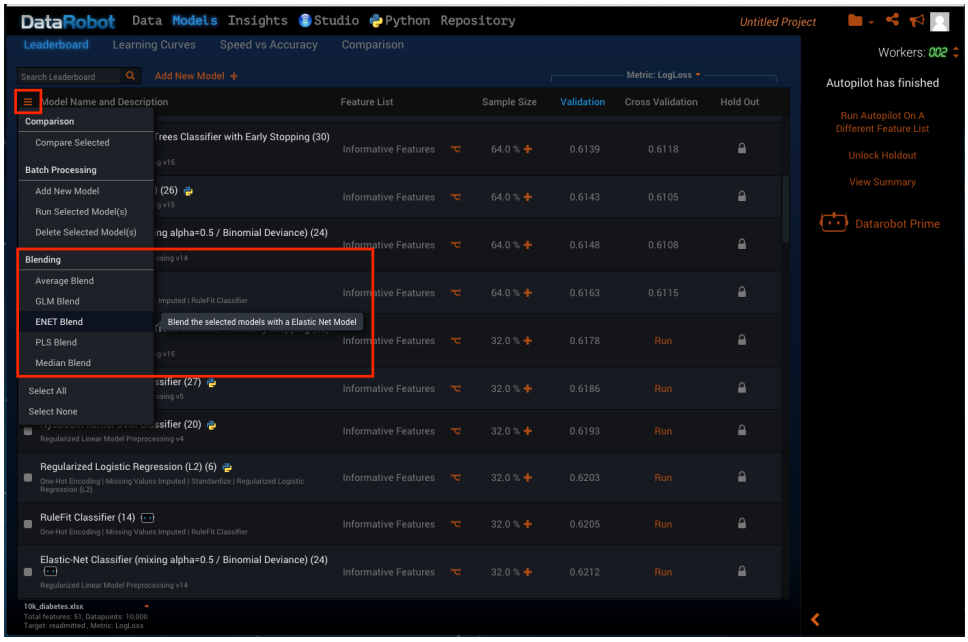
Blending enables you to combine the predictions of two or more models, which often leads to better results than running either of the individual models. Currently DataRobot provides options for Average, GLM, ENET, PLS, and Median blending. For each target point, the Average and Median blenders will calculate the average or median values of the predictions of the selected individual models. GAM (*Generalized Additive Model*) and GLM (Generalized Linear Model) blenders are essentially a second layer of models on the top of the existing models. They use the predictions of the selected models as predictors for GAM and GLM, while keeping the same target as individual models.

Use the following steps to create a blended model.

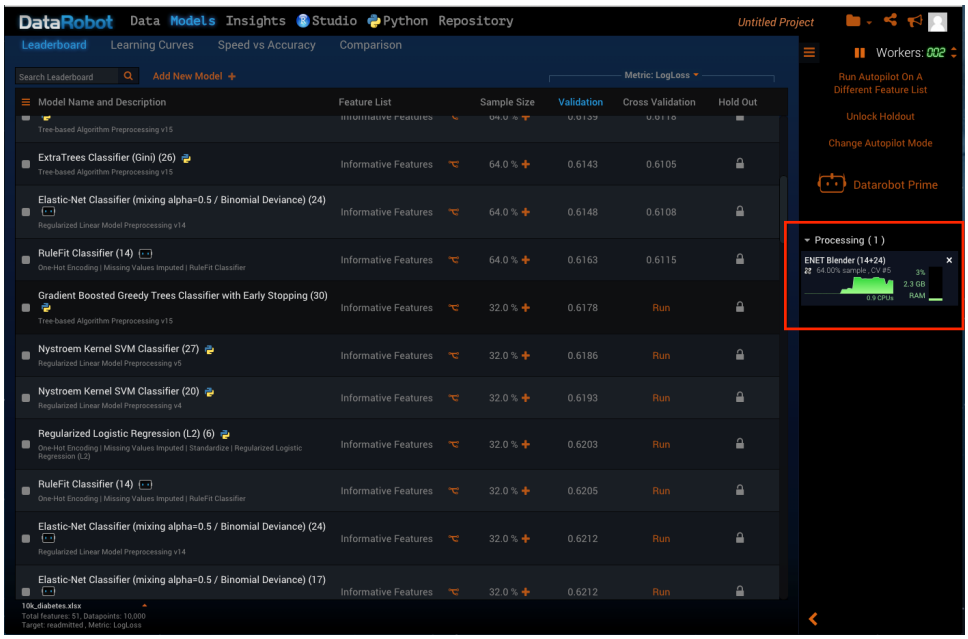
1. Use the check boxes on the left side of the model Leaderboard to select two or more models.



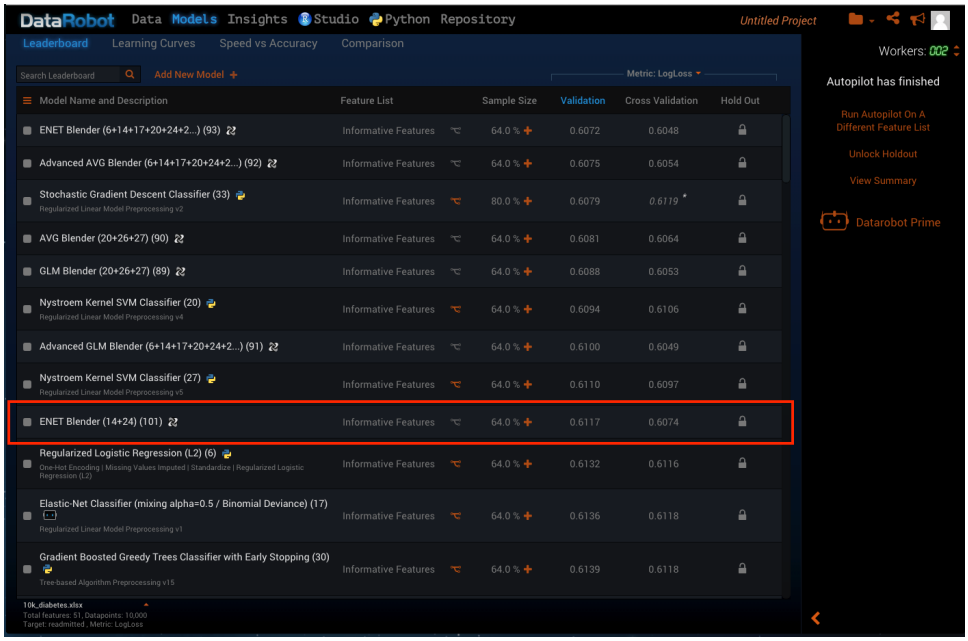
2. Click the model menu icon at the top left of the Leaderboard, then select one of the blending options listed under **Blending**. Hover over each menu item displays a description of the blending option.



3. A new job appears in the Worker queue while the blended model is processed.



4. When processing is complete, the new blended model appears in the list on the Leaderboard.



Run Selected Models

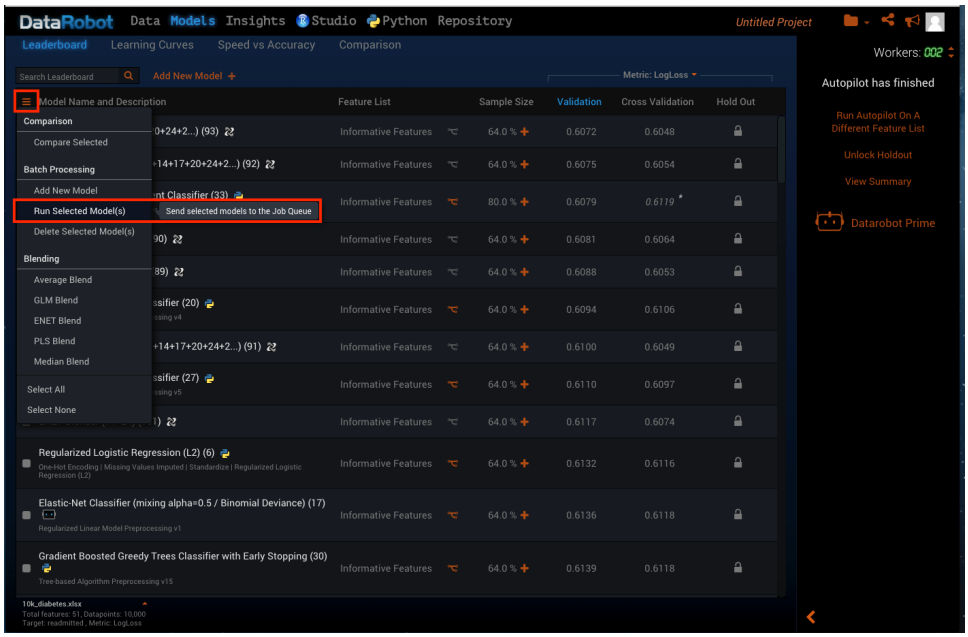
After models are created, you can use **Run Selected Models** to retrain selected models with a different set of parameters.

Note:

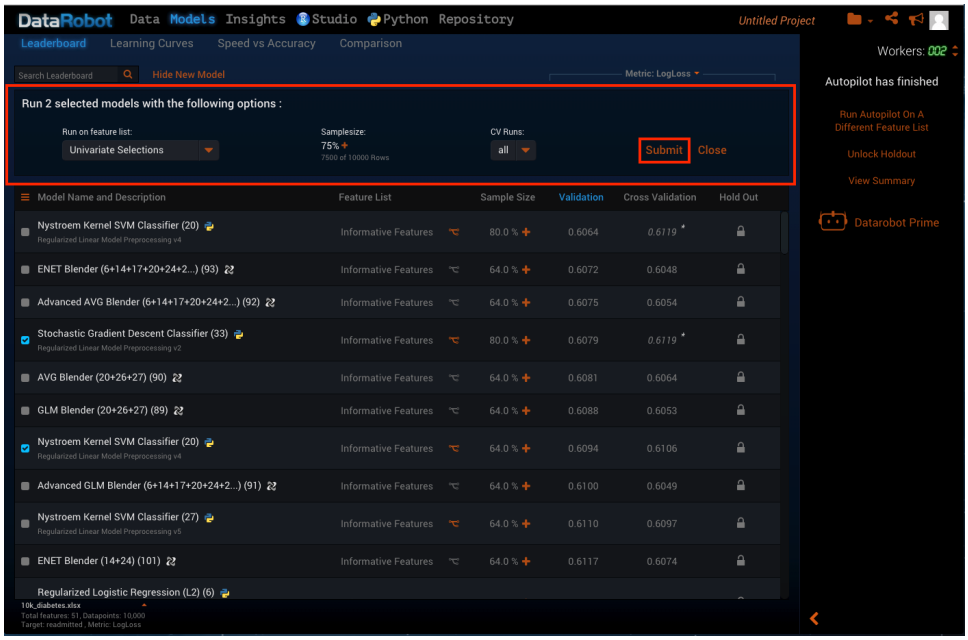
You cannot use **Run Selected Models** on blended models.

Use the following steps to run one or more selected models.

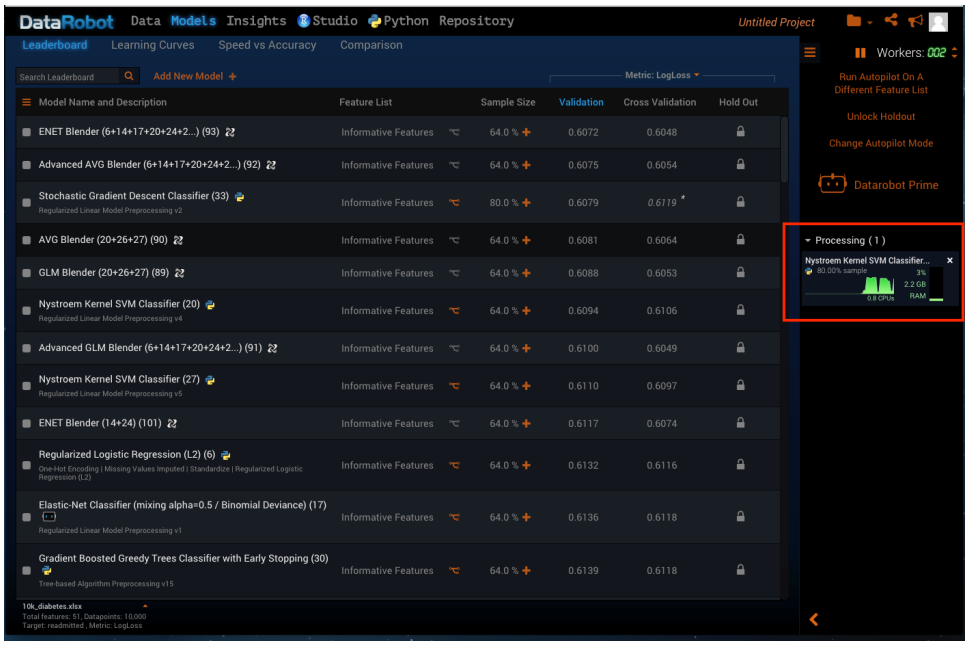
1. Use the check boxes on the left side of the model Leaderboard to select one or more models. Click the model menu icon at the top left of the Leaderboard, then click **Run Selected Model(s)**.



2. Use the box at the top of the Leaderboard to specify a feature list, sample size, and the number of cross-validation (CV) runs, then click **Submit** to retrain the selected models with the specified parameters.



3. New jobs appear in the Worker queue while the models are processed.

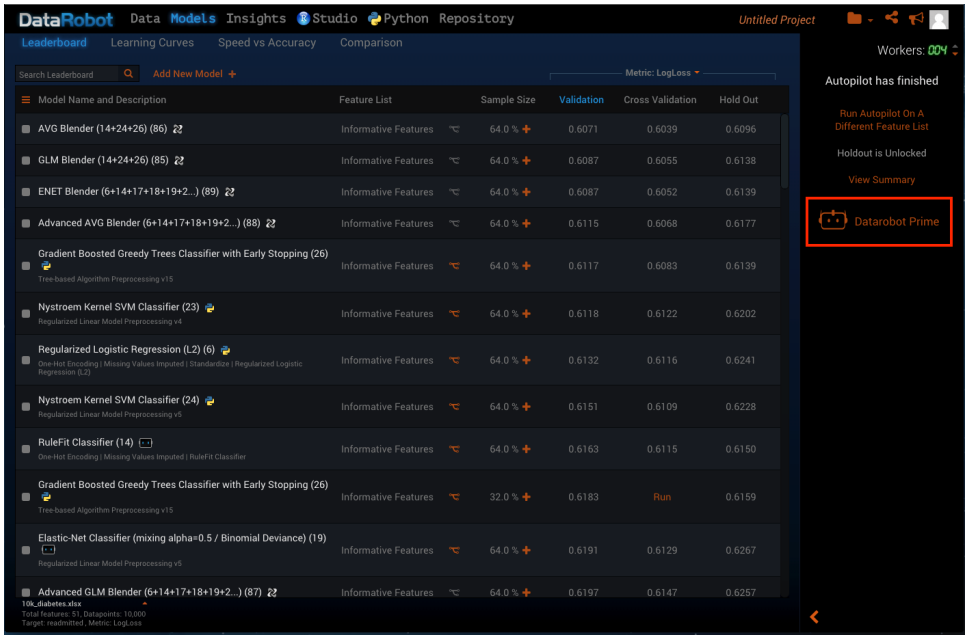


DataRobot Prime

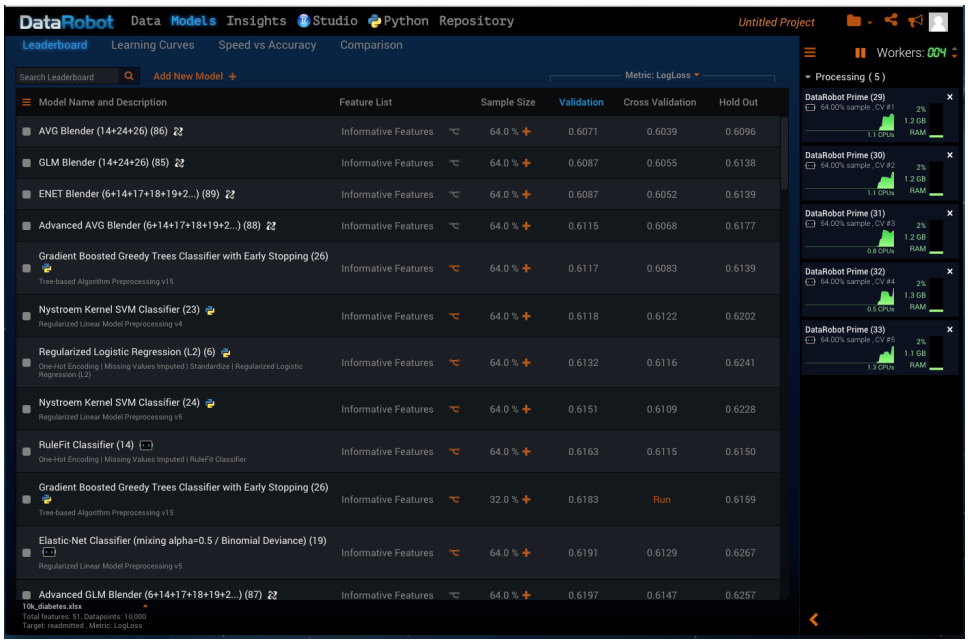
You can use DataRobot Prime to build a flexible model that finds a balance between complexity and accuracy.

Use the following steps to create an optimized model using DataRobot Prime.

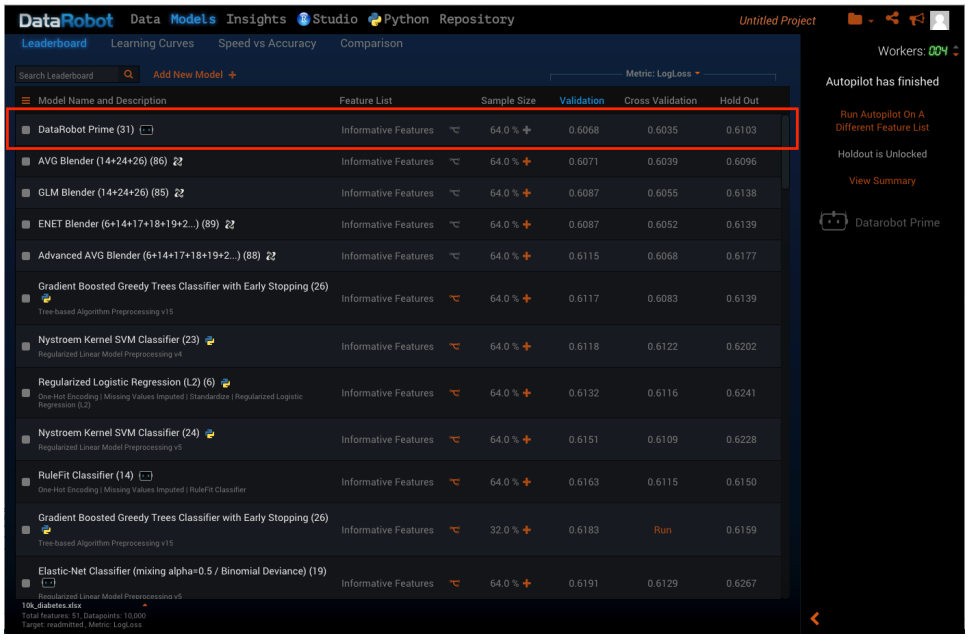
1. In the Worker Queue at right of the model Leaderboard, click **DataRobot Prime**.



2. The DataRobot Prime job progress is displayed in the Worker Queue.



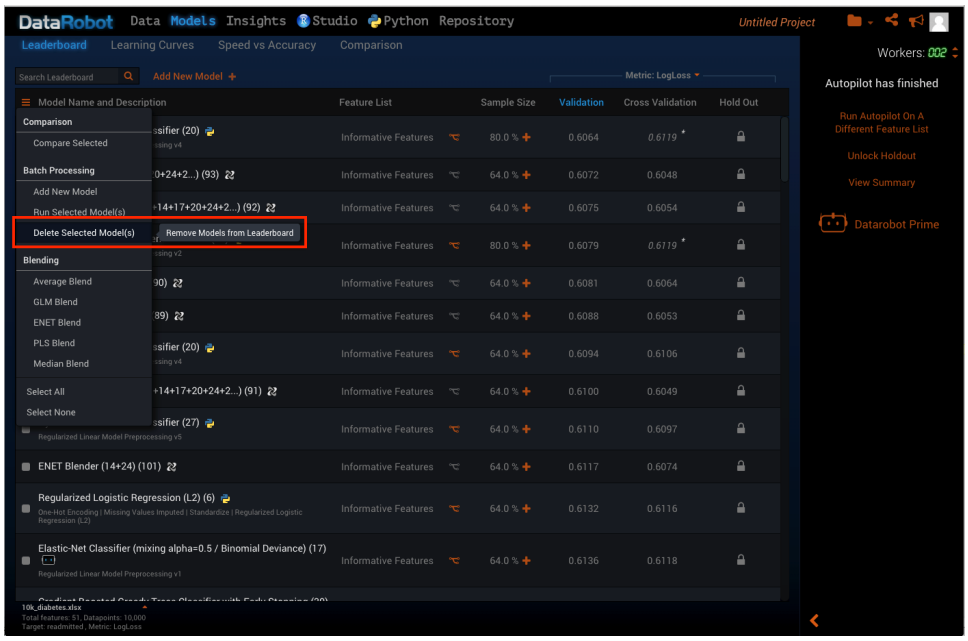
3. When processing is complete, the DataRobot Prime model appears in the list on the model Leaderboard.



Delete Models

Use the following steps to delete models.

Use the check boxes on the left side of the model Leaderboard to select one or more models. Click the model menu icon at the top left of the Leaderboard, then click **Delete Selected Model(s)**.

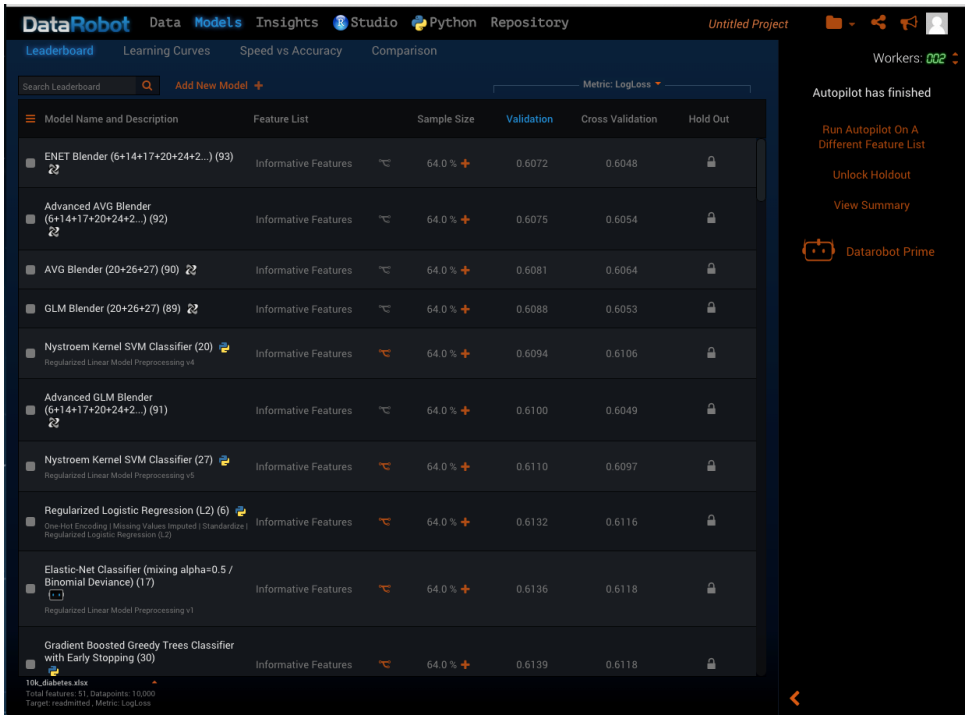

DataRobot
Data
Models
Insights
Studio
Python
Repository
Untitled Project
Leaderboard
Learning Curves
Speed vs Accuracy
Comparison
Search Leaderboard
Add New Model
Metric: LogLoss
Model Name and Description
Feature List
Sample Size
Validation
Cross Validation
Hold Out
Comparison
Compare Selected
Batch Processing
Add New Model
Run Selected Model(s)
Delete Selected Model(s)
Remove Models from Leaderboard
Blending
Average Blend
GLM Blend
ENET Blend
PLS Blend
Median Blend
Select All
Select None
Informative Features
0.6064
0.6119
0.6072
0.6048
0.6075
0.6054
0.6079
0.6119
0.6081
0.6064
0.6088
0.6053
0.6094
0.6106
0.6100
0.6049
0.6110
0.6097
ENET Blender (14+24) (101)
0.6117
0.6074
Regularized Logistic Regression (L2) (6)
0.6132
0.6116
Elastic-Net Classifier (mixing alpha=0.5 / Binomial Deviance) (17)
0.6136
0.6118
Workers:
Autopilot has finished
Run Autopilot On A Different Feature List
Unlock Holdout
View Summary
Datarobot Prime
10k_diabetes.xlsx

Evaluating Models

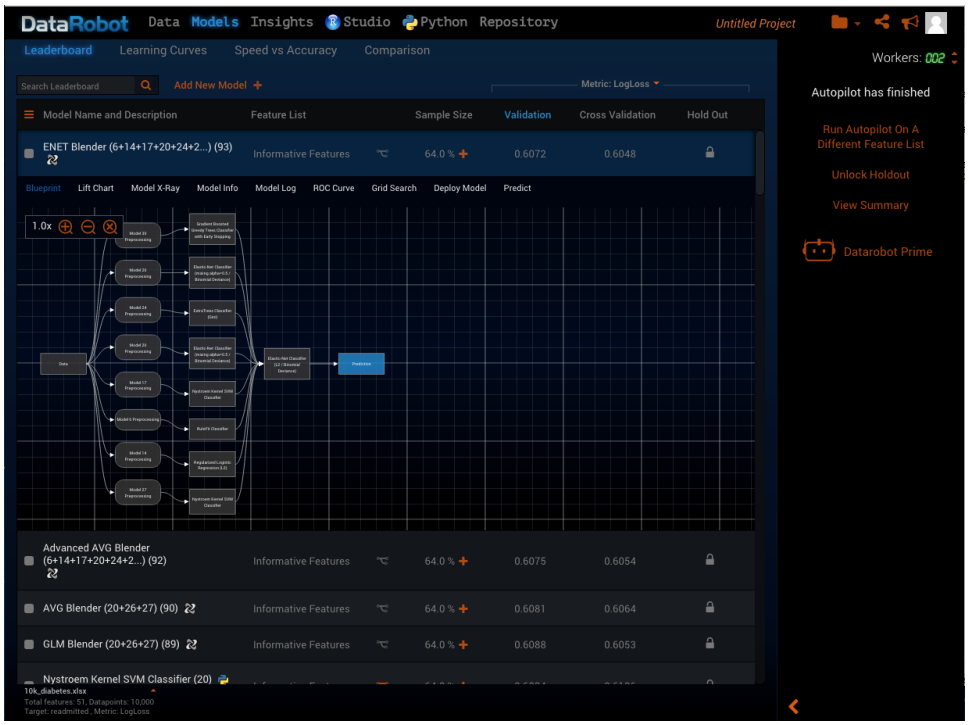
After you create a DataRobot project, you can select a target feature and start the modeling process. You can then evaluate and select the best models to use for prediction.

The Model Leaderboard

Click **Models** in the top navigation menu to view the Leaderboard. The Leaderboard is a list of models ranked by the chosen performance metric, with the best models at the top of the list. You cannot view full details about a model until it finishes running.



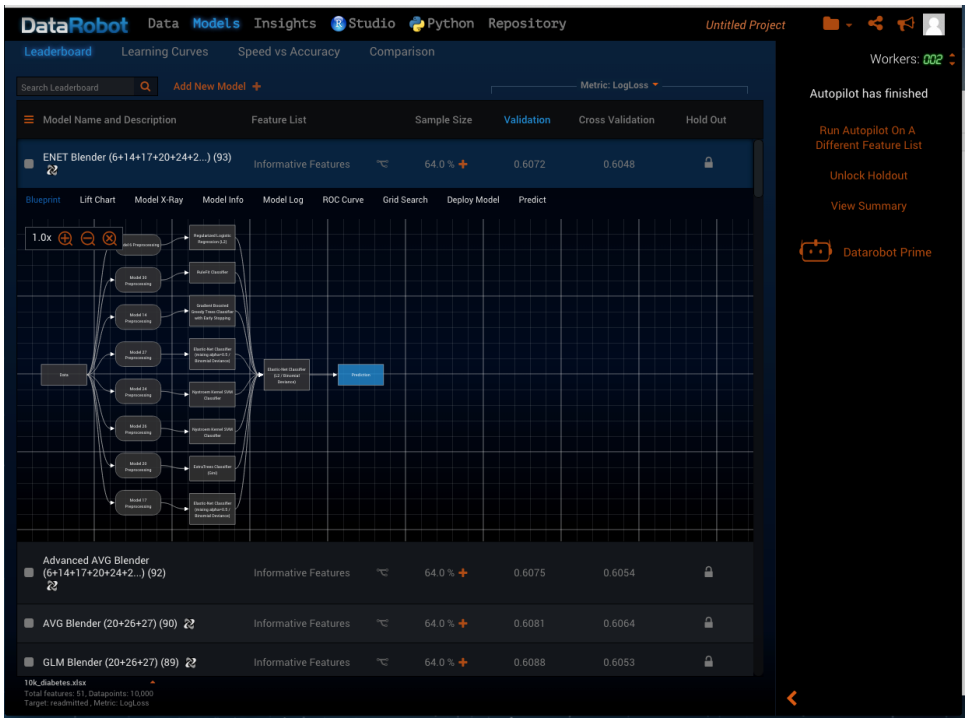
Click a model name to view details about the model. Model details are discussed in the following sections.



Blueprint

Blueprints provide a graphical representation of the many steps involved in transforming input predictors and targets into a model. A blueprint represents the high-level end-to-end procedure for fitting the model, including any pre-processing steps, algorithms, and post-processing. Each box in a blueprint may represent multiple steps.

To view a graphical representation of a blueprint, click a model in the leaderboard. You can use the navigation toolbar to zoom in on the blueprint.

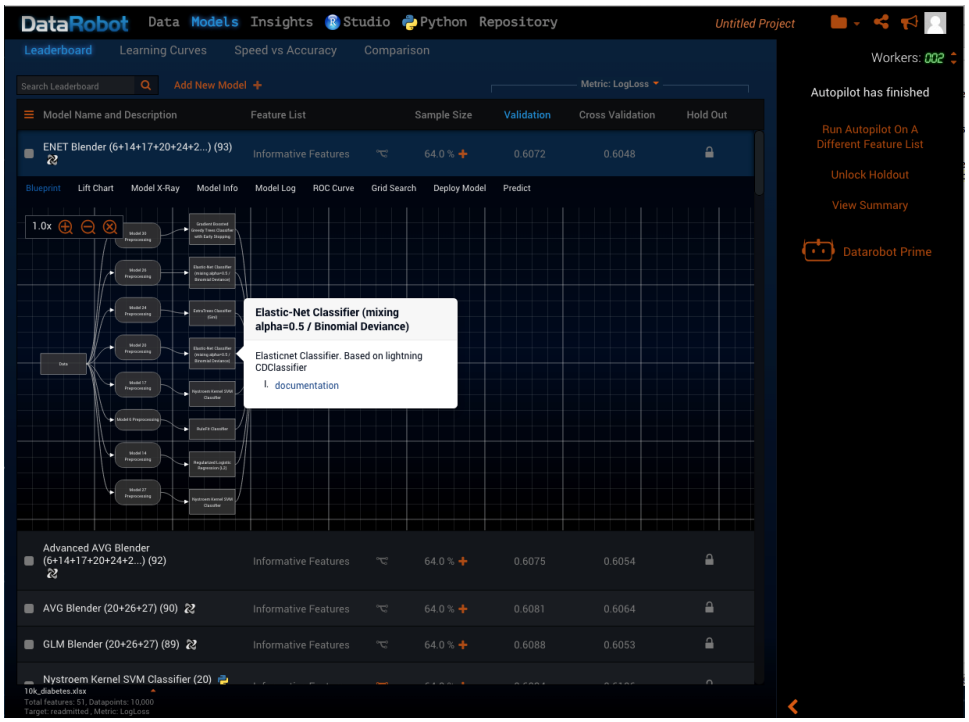


Input Types

Different columns in the dataset require different types of preparation and transformation. For example, some algorithms recommend subtracting the mean and dividing by the standard deviation of the input data -- but this would not make sense for text input data. The first step in the execution of a blueprint is to identify data types that belong together so they can be processed separately.

Transformers and Models

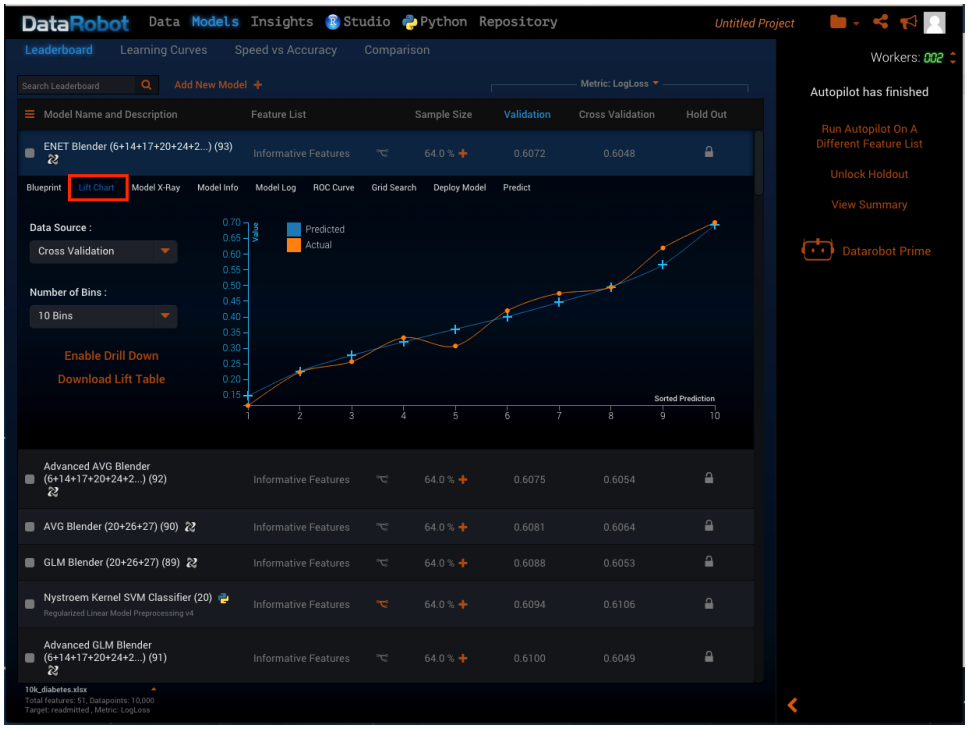
The other nodes in the blueprint are other types of data transformations or models. Click a blueprint node to display additional information. Many of the models use DataRobot proprietary approaches to data pre-processing.



Lift Chart

The lift chart depicts how effective a model is at predicting the target. The chart is sorted by predicted values, so you can see how well the model performs for different ranges of values of the target variable.

To view a lift chart, click a model in the Leaderboard list, then click **Lift Chart**.



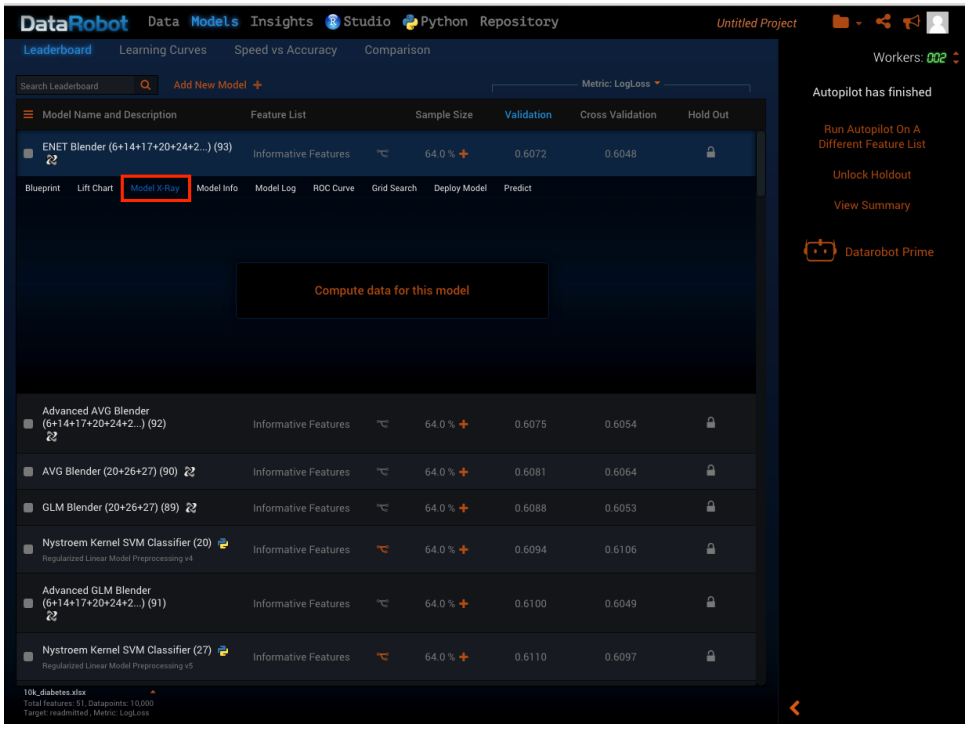
Display Options

- Data Source -- select validation, cross-validation, or holdout.
- Number of Bins -- use this drop-down to adjust the granularity of the displayed values.
- Enable Drill Down -- click this link to compute and download cross-validation predictions.
- Download Lift Table -- download a .csv file with the lift table data.

Model X-Ray

The Model X-Ray enables you to view a chart of model details on a per-feature basis.

To view the Model X-Ray, click a model in the Leaderboard list, select **Model X-Ray**, then click **Compute data for this model**.



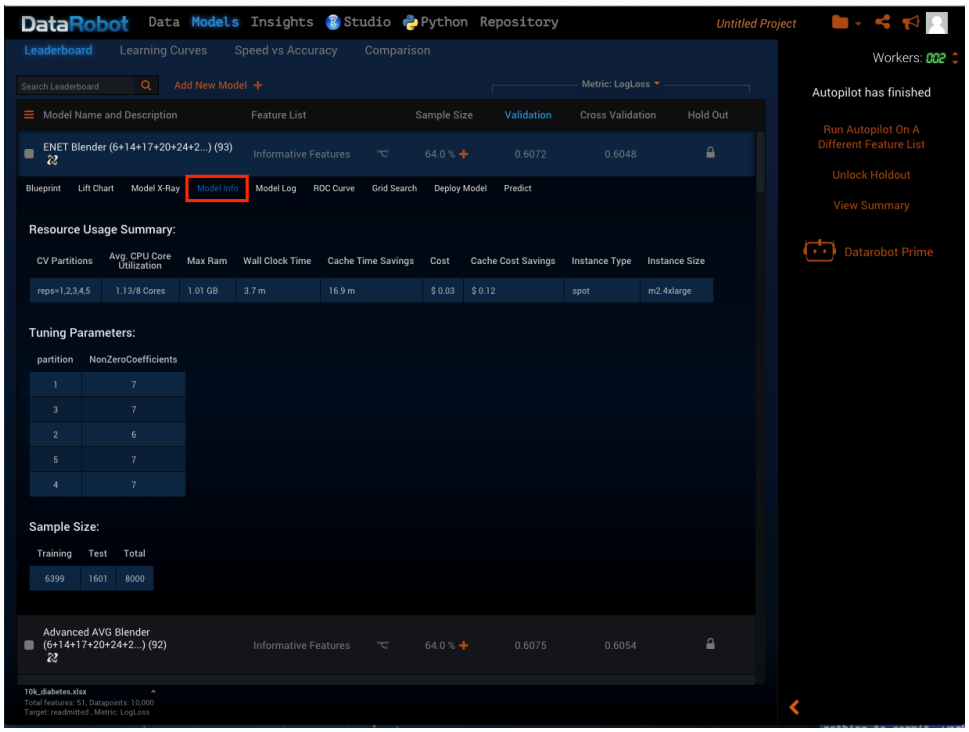
Depending on the model, it may take a few minutes for all of the features to appear in the Feature list.

Display Options

- Data Source -- select a data source.
- Feature -- click a name in the Feature list to view information for that feature, or use the orange arrows to scroll through the list. Hovering over each feature displays a thumbnail image of the chart.
- Sort -- select a sorting option.
- Log X Axis -- display the logs of the x-axis values.

Model Info

To display model information, click a model in the Leaderboard list, then click **Model Info**.



Resource Usage Summary

A summary of resource use that includes elapsed time, computing resources, cost, and the type of machine used to build the model (by cross-validation repetition, if applicable). Savings due to the reuse of previously computed tasks (Cache Time Savings) is also provided.

Tuning Parameters

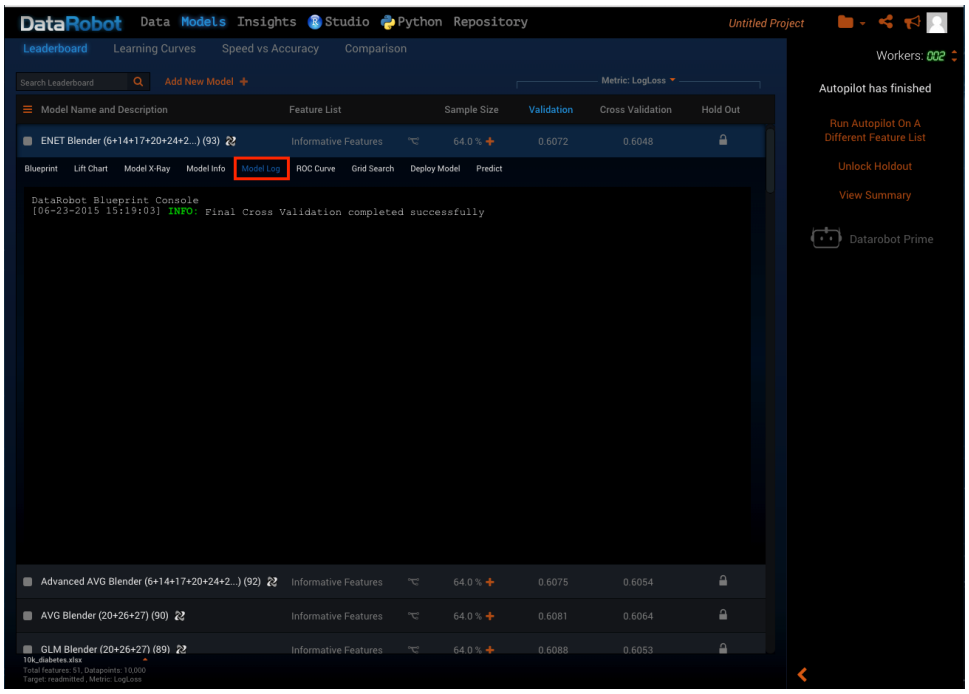
Model optimization parameters.

Sample Size

The number of observations used to train and validate the model (for each cross-validation repetition, if applicable).

Model Log

To display the model log, click a model in the Leaderboard list, then click **Model Log**.

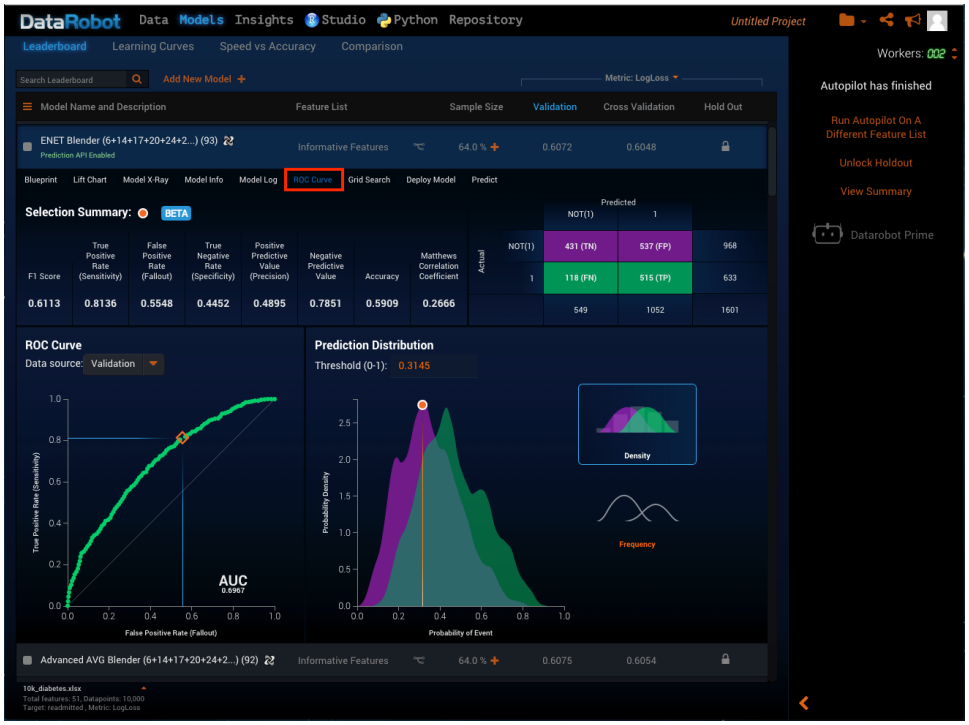


ROC Curve

You can use the ROC Curve page to assess model quality. To view the ROC Curve page, click a model in the Leaderboard list, then click **ROC Curve**.

Note: The ROC curve is displayed only for models created for a binary classification target, i.e., a target with two unique values.

The ROC Curve page contains a set of interactive graphical displays that includes the ROC Curve, a selection summary, the prediction distribution, and a confusion matrix.



ROC Curve

The ROC Curve plots the true positive rate against the false positive rate. It has two important characteristics: the area under the curve, and the shape of the curve.

Area Under the ROC Curve

The Area Under the Curve (AUC) is displayed at the lower right of the ROC Curve. The goal is for the AUC to be as large as possible.

- If the area = 0.5, it means that the predictions based on this model are no better than a random model.
- If the area = 1.0, it would mean that predictions based on the model are perfect.

A perfect model is not realistic -- if it does happen, there is likely something wrong with the model (it may contain variables that depend on the response and should be excluded). Therefore the goal is to achieve an area as large as possible between 0.5 and 1.0.

ROC Curve Shape

Another informative feature of the ROC Curve is its shape. It is better when the curve grows quickly for small X-values, and slowly for values of X closer to 1. An ideal ROC curve would hug the top left corner, indicating a high true positive rate and a low false positive rate.

Display Options

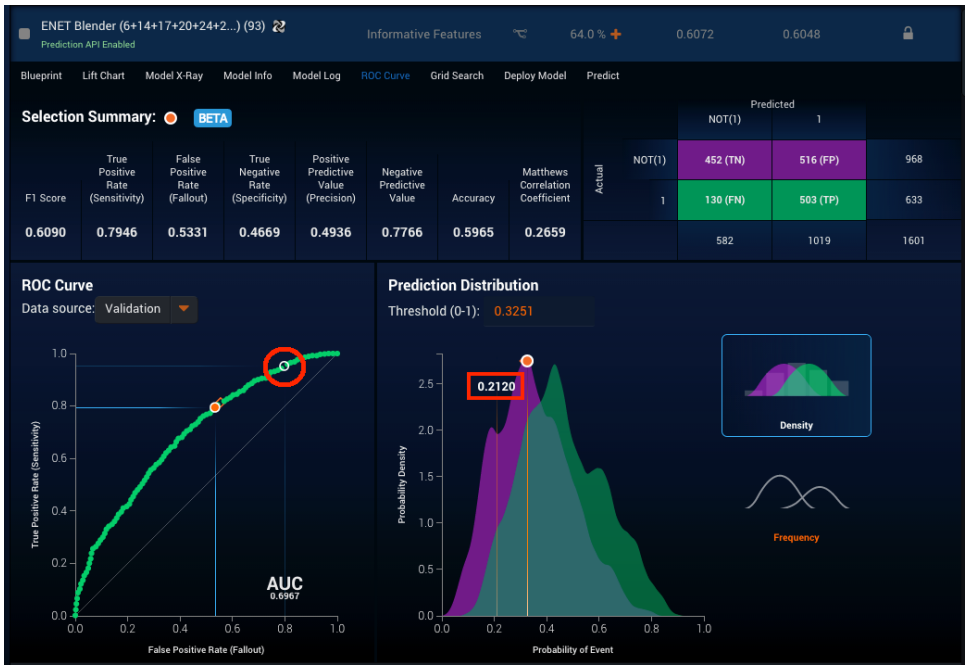
You can use the Data Source drop-down to display validation or cross-validation data.

Prediction Distribution

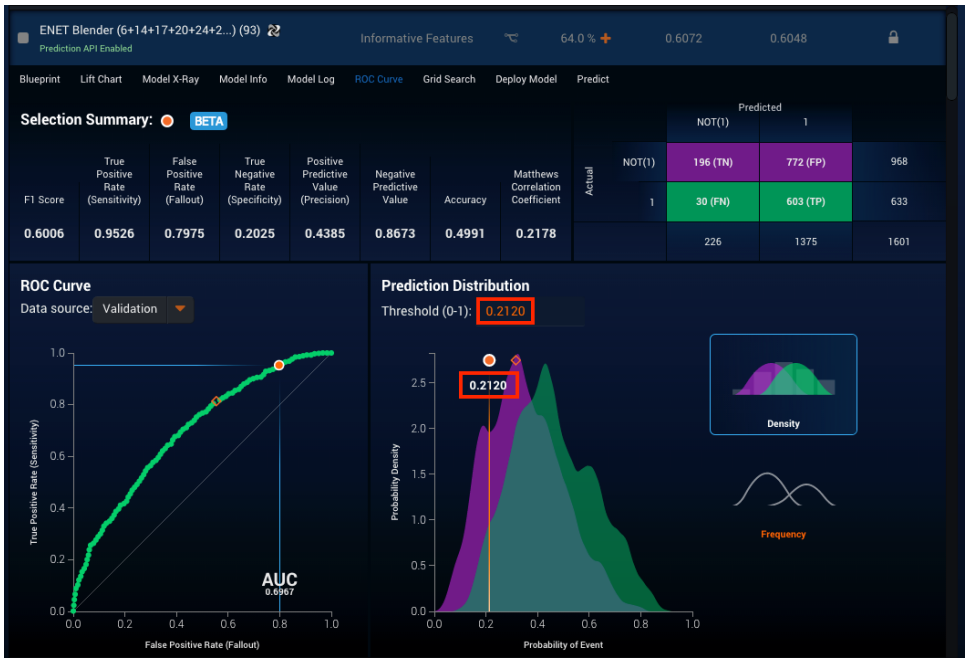
The Prediction Distribution chart expresses the performance of the model on the validation, cross-validation, or holdout dataset. You can use the Prediction Distribution chart to change the threshold value. When you change the threshold value, the ROC Curve, the Selection Summary table, and the Confusion Matrix are also updated to reflect the new value.

Use the following steps to set a new threshold value.

1. Pass the cursor over the Prediction Distribution chart. The threshold value will be displayed in white text as you move the cursor over the chart, and guidelines for the new values will also be dynamically displayed on both the Prediction Distribution chart and the ROC Curve.



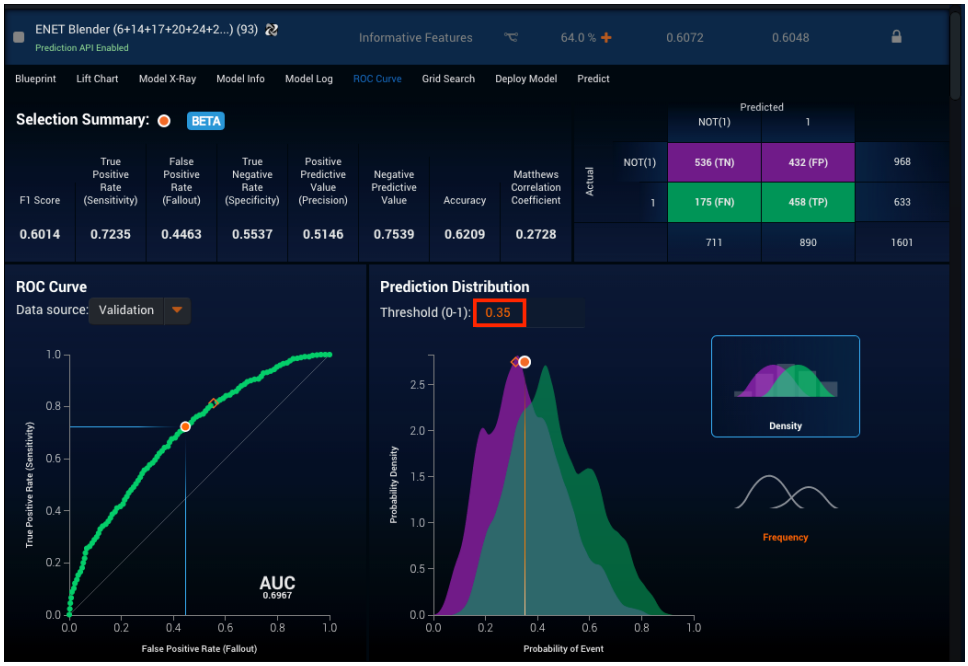
2. Click to select a new threshold value. The new value will appear in orange text in the Threshold field, and the orange dots and guidelines on the Prediction Distribution graph and ROC Curve will move to reflect the new threshold value. The information in the Selection Summary table and the Confusion Matrix will also be updated to reflect the new value.



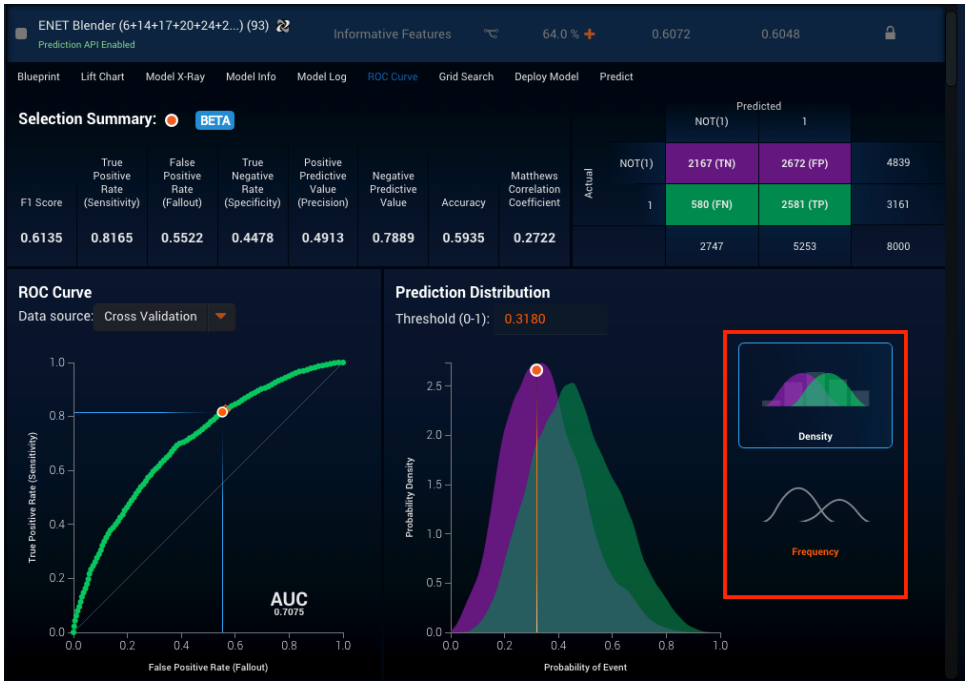
3. You can also change the threshold setting by typing a new value in the Threshold field. Click the orange Threshold text, then type in a new value.

Note: "Invalid value" will appear next to the field if the number entered is not between 0 and 1. If a non-numeric character is entered, "NaN" (Not a Number) will appear in the Threshold field.

The orange dots and guidelines on the Prediction Distribution graph and ROC Curve will move to reflect the new threshold value, and the information in the Selection Summary table and the Confusion Matrix will also be updated to reflect the new value.

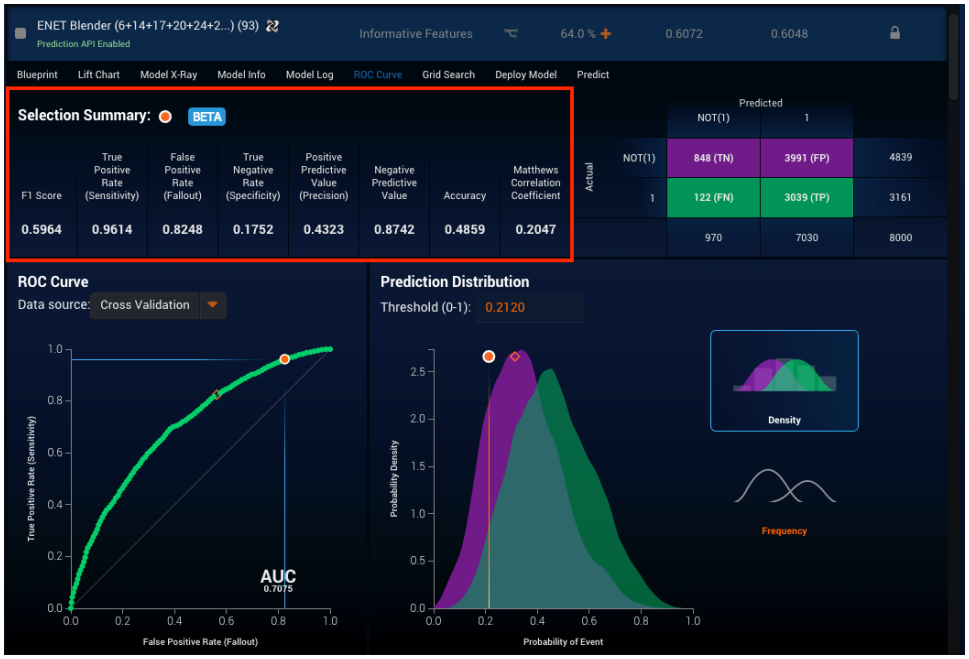


4. Click the **Density** and **Frequency** symbols at the right of the Prediction Distribution chart to display the Predictions distribution as a density or frequency curve:
   - Density curve -- There is an equal area underneath both the positive and negative curves.
   - Frequency curve -- The area underneath each curve varies, and is determined by the number of observations in each class.



Selection Summary

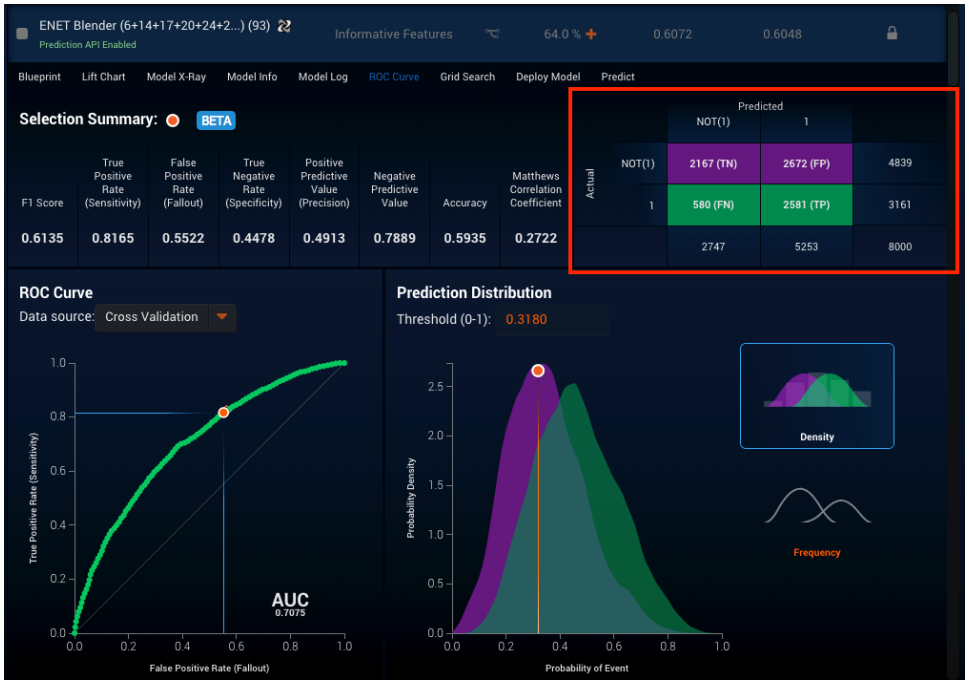
The Selection Summary table contains a number of statistics describing model performance at the selected threshold, including the F1 Score (the normalized accuracy), sensitivity, fallout, specificity, precision, negative predictive value, accuracy, and the Matthews Correlation Coefficient.



Confusion Matrix

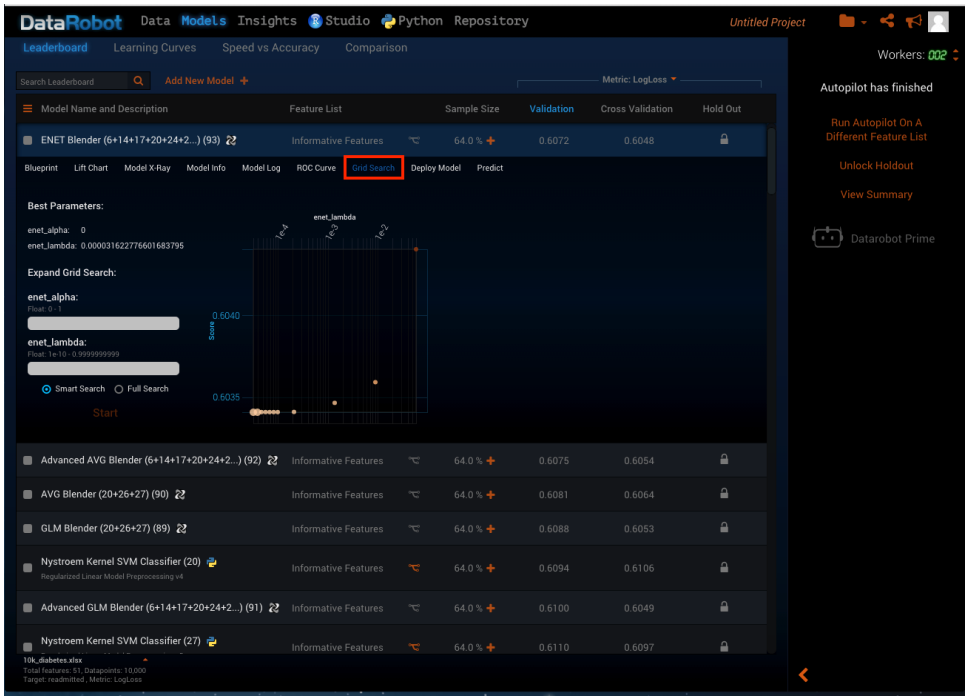
The Confusion Matrix is a table with two rows and two columns that reports the number of false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The name "confusion matrix" refers to the fact that the matrix makes it easy to see if the model is confusing two classes (consistently mislabeling one class as another class).

The information in this table facilitates more detailed analysis than relying on accuracy alone. Accuracy is not always a reliable metric for the real performance of a classifier, because it will yield misleading results if the data set is unbalanced (that is, when the number of samples in different classes varies greatly).



Grid Search

To display the model Grid Search, click a model in the Leaderboard list, then click **Grid Search**.



Model hyperparameters are tuned using internal cross-validation over the training sample, which completely omits the Leaderboard test set. If the hyperparameter space is large, a pattern search algorithm is used to find the optimum more quickly without testing every point on the grid.

You can use Grid Search to start an additional parameter grid search to help fine-tune a model or explore additional tuning parameters. To run a new grid search, type parameters in the **enet_alpha** and **enet_lambda** boxes, select **Smart Search** or **Full Search**, then click **Start**. Parameters can be entered as a comma-separated list of values.

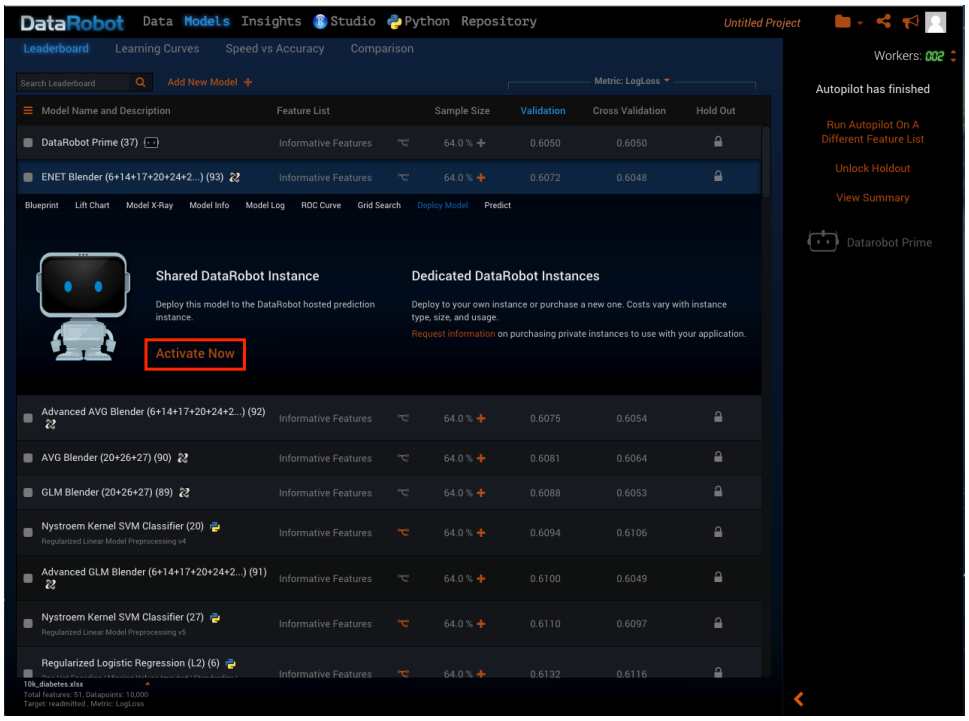
Deploy Model

Deploy Model only applies to DataRobot Enterprise edition. If you are using the hosted service, you can click the **Request Information** link to get more information about purchasing DataRobot Enterprise edition.

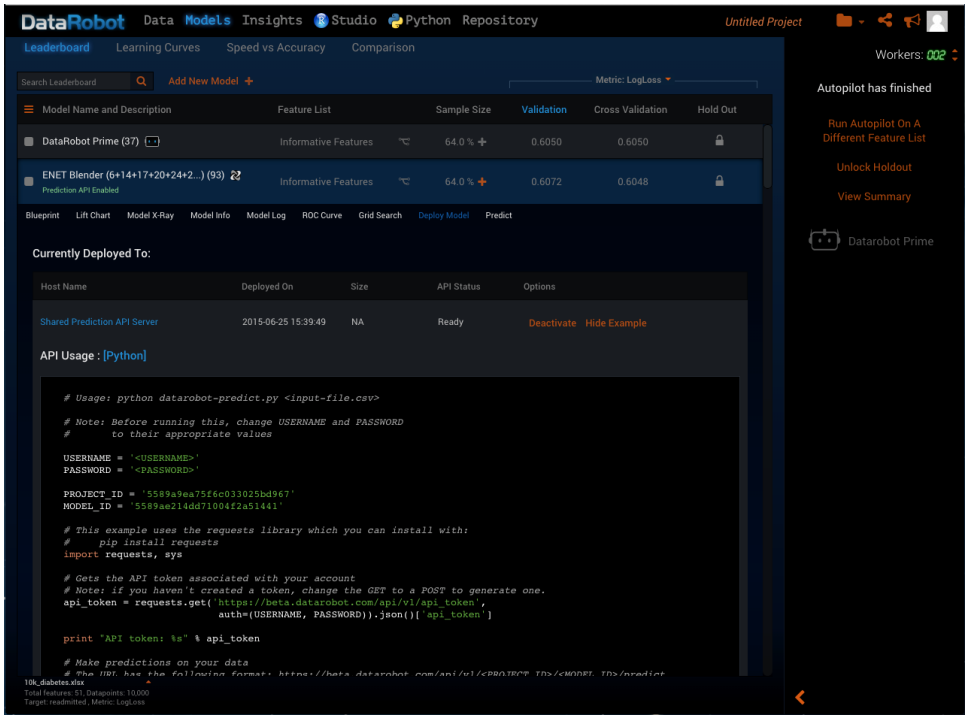
If you are an Enterprise user, you can use Deploy Model to deploy the model to your hosted prediction instance.

To deploy a model, click a model in the Leaderboard list, click **Deploy Model**, then click **Activate Now**. When you deploy and activate a model:

- The model is cached to maximize speed.
- A layer of security is added that restricts access to the prediction server.



After you activate a model, you can click **Show Example** to display sample Python code for running a prediction using the model. The code sample also shows how to create an API token.

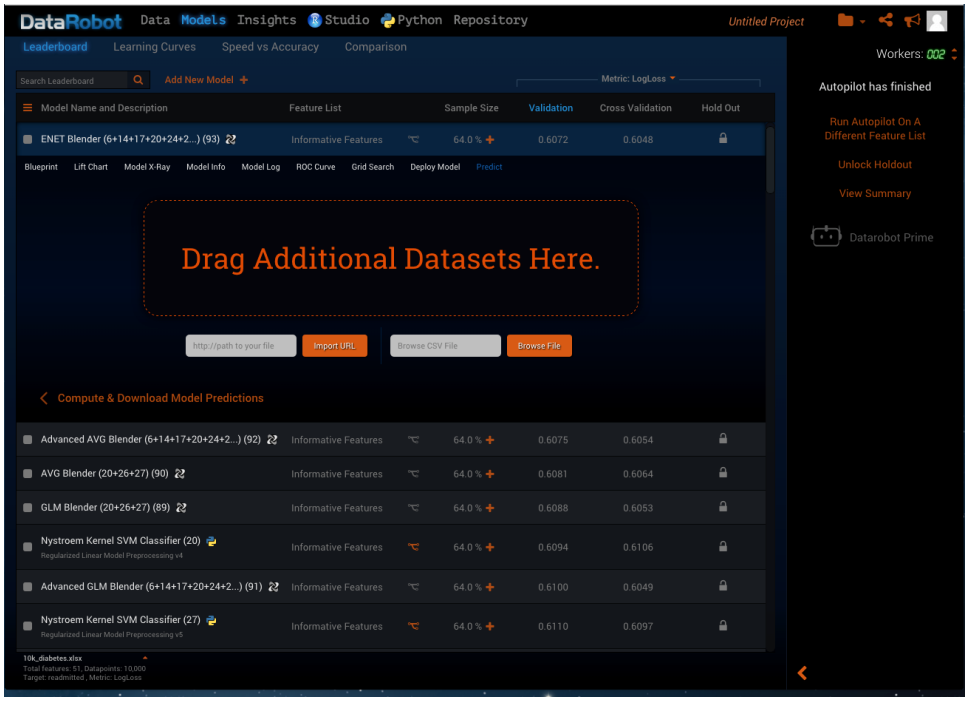


Click **Hide Example** to hide the code sample. Click **Deactivate** to remove the model from the prediction server.

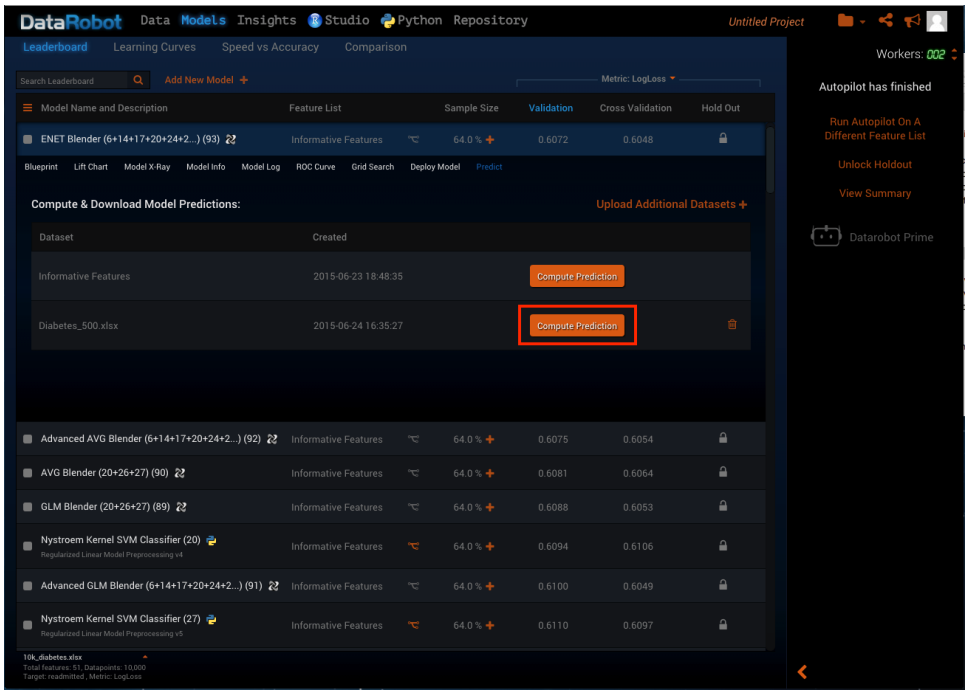
Predict Against a Model

Use the following steps to generate predictions on a new dataset.

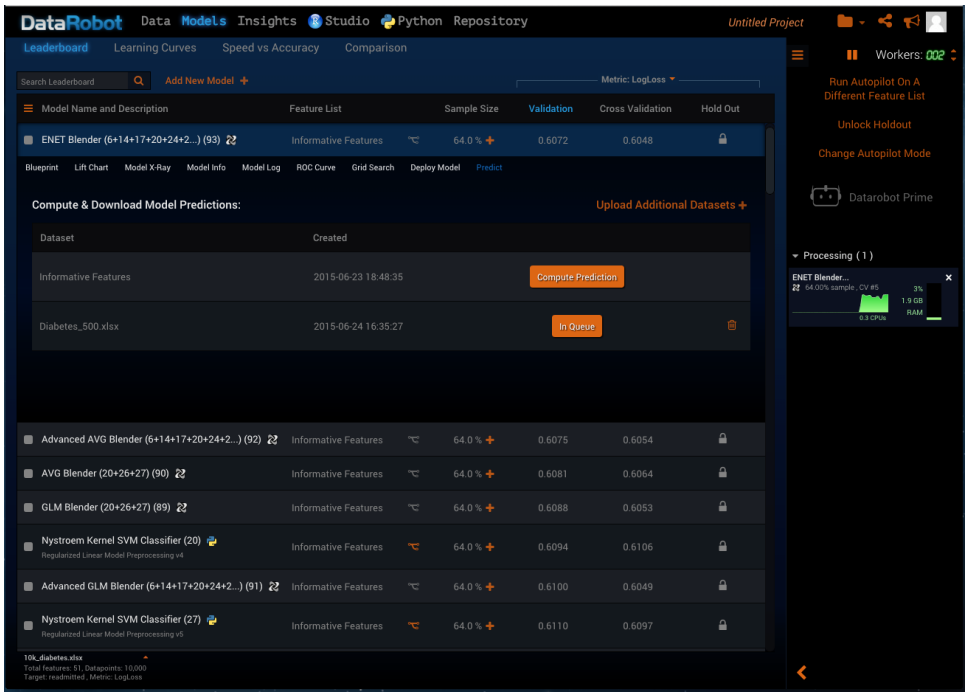
1. Click a model in the Leaderboard list, then click **Predict**. Use one of the following methods to select a dataset to predict with the model.
   - Drag a .csv data file to the orange box on the DataRobot platform home page.
   - Click Browse File, then use the file browser to select a .csv data file.
   - Type a URL in the box labeled "http://path to your file", then click Import URL.



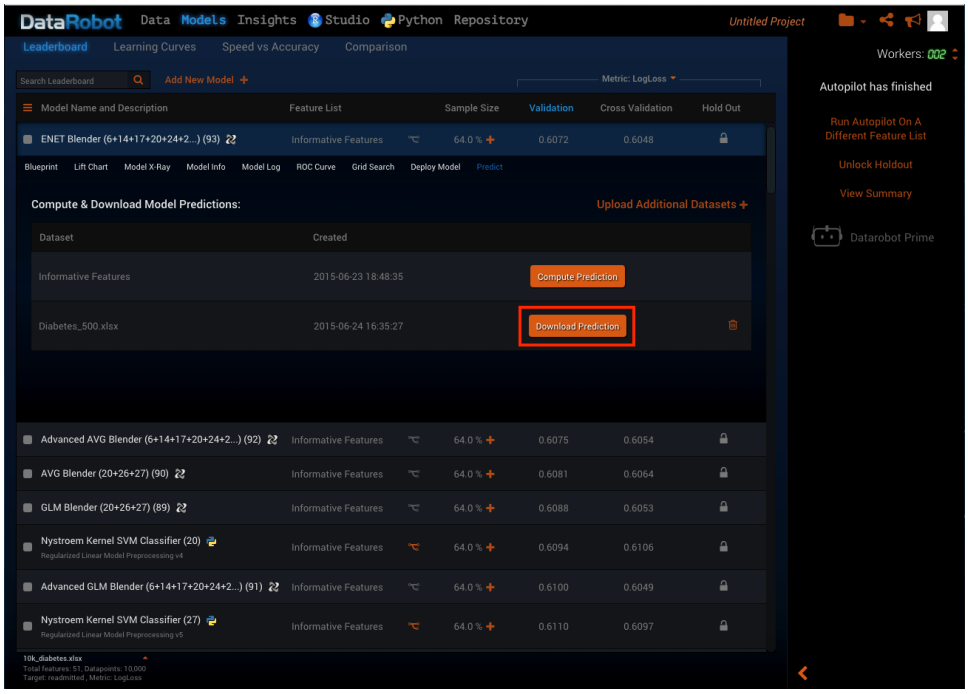
2. After the file is uploaded, click **Compute Prediction** next to the file.



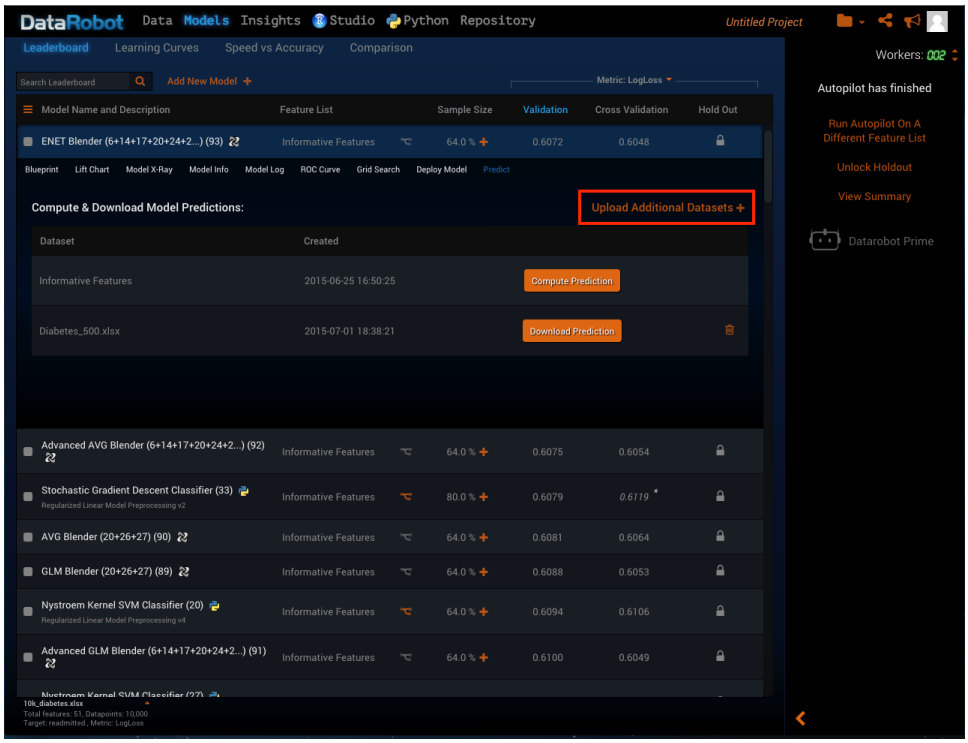
3. The **Compute Prediction** button changes to **In Queue** for the selected dataset, and the job status appears under Processing in the Worker queue.



4. When the prediction has finished running, click **Download Prediction** to view the results in a .csv file.



5. To upload and run predictions on additional datasets, click **Upload Additional Datasets**.



Learning Curves

Learning curves demonstrate how model performance varies as the sample size changes. This is useful in determining whether it is worthwhile to increase the size of the dataset. Getting additional data can be expensive, but may be worthwhile it if it increases performance.

To display the model learning curves, select **Models > Learning Curves**.

Performace versus sample size are displayed for the top-performing models. Hold the cursor over a line to display data for that model only.

Things to observe include whether there are any sharp changes, and whether models are doing worse with increased sample size. If the dataset or the validation set is small, there may be significant variation due to the exact charactersitics of the datasets. It is possible for model performance to decrease with increasing sample size, as the models may become overly sensitive to particular characteristics of the training set. Note that in general, high-bias models (such as linear models) may do better at small sample sizes, while more flexible, high-variance models often perform better at large sample sizes. Preprocessing variations can also increase model flexibility.

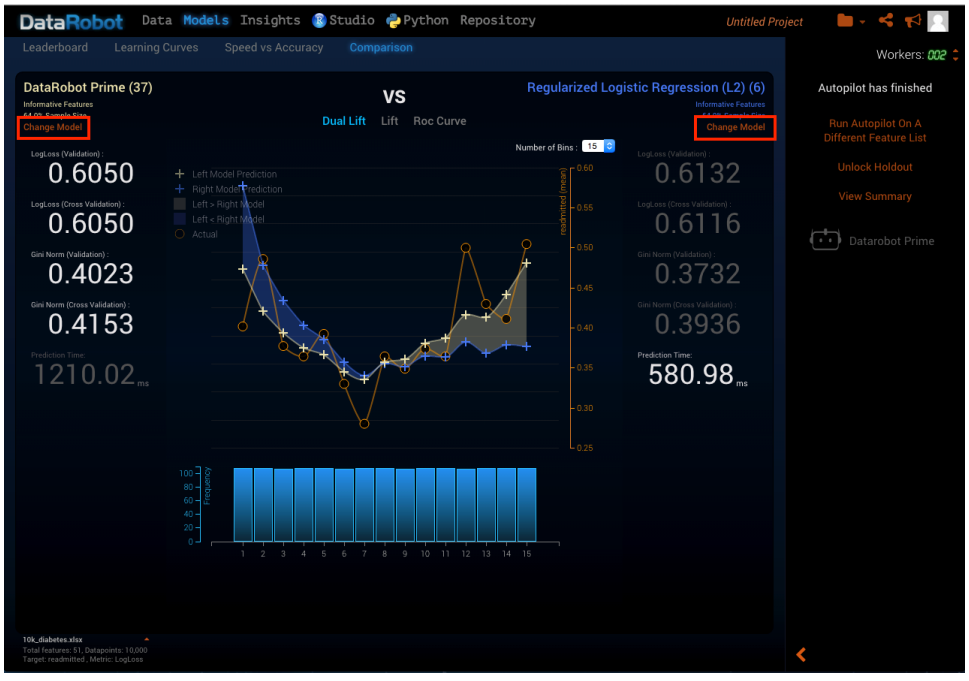
Speed Vs. Accuracy

Predictive accuracy often comes with the price of increased prediction runtime. The Speed vs. Accuracy analysis plot shows the tradeoff between runtime and predictive accuracy, and helps you choose the best model with the lowest overhead.

To display this performance information, select **Models > Speed vs Accuracy**.

Comparison

To compare models, select **Models > Comparison**. You can view a plot of predicted versus actual values for the two models, along with other information, such as LogLoss and prediction runtime. To compare different models, click the left and right **Change Model** links.



Another way to access compare models is to select two models in the model Leaderboard, click the menu icon, and then click **Compare Selected**.

Insights

You can use Insights to view graphical representations of model aspects such as variable importance and variable effects. You can also use a text mining chart and word cloud to assess variable keyword relevancy, and view a hotspot chart of predictive performance.

Variable Importance

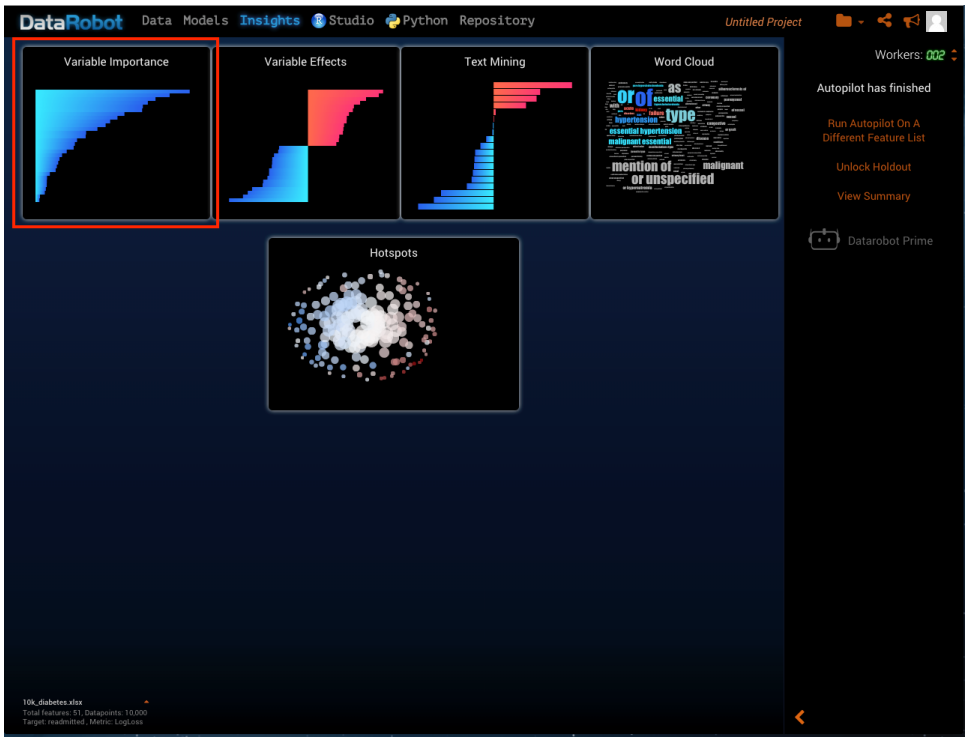
You can use the Variable Importance chart to view the sorted relative importance of all key variables driving each model in the project.

Sometimes this relative importance can be very useful, especially when a particular feature appears to be much more important for the predictions than all of the other features. It is usually worth checking to see if the values of this very important variable do not depend on the response. If this is the case, you may want to exclude this feature when training the model. Not all models have a Coefficients chart, and the Importance graph is the only way to visualize the feature impact to the model.

If a feature is included in only one model out of the dozens generated by DataRobot, it may not be particularly important, and excluding it from the feature set might help optimize model-building and future predictions.

Also, it is useful to compare how feature importance changes for the same model with different feature lists. Sometimes the features recognized as important on a reduced dataset differ substantially from those from the full feature set.

To display the Variable Importance charts, select **Insights > Variable Importance**.

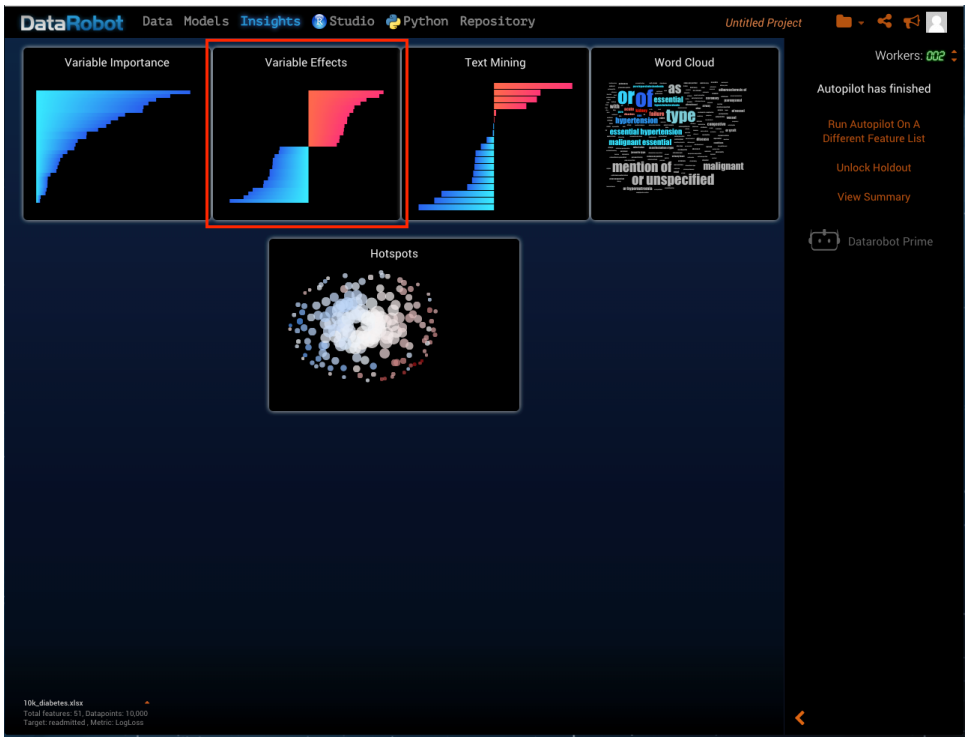


To view the Variable Importance chart for another model, click the arrow next to the model name, then select a new model.

Variable Effects

While Variable Importance shows you the relevancy of different variables for a model, the Variable Effects chart shows the impact of the top 30 variables (by magnitude) in the prediction outcomes. This is helpful when comparing the impact of a feature for different models. It is useful to ensure that the relative rank of feature importance across models does not vary wildly. If in one model a feature is regarded to be very important with positive effect, but has a negative effect in another model, there is probably something worth double-checking in both the dataset and the model.

To display the Variable Effects charts, select **Insights > Variable Effects**.



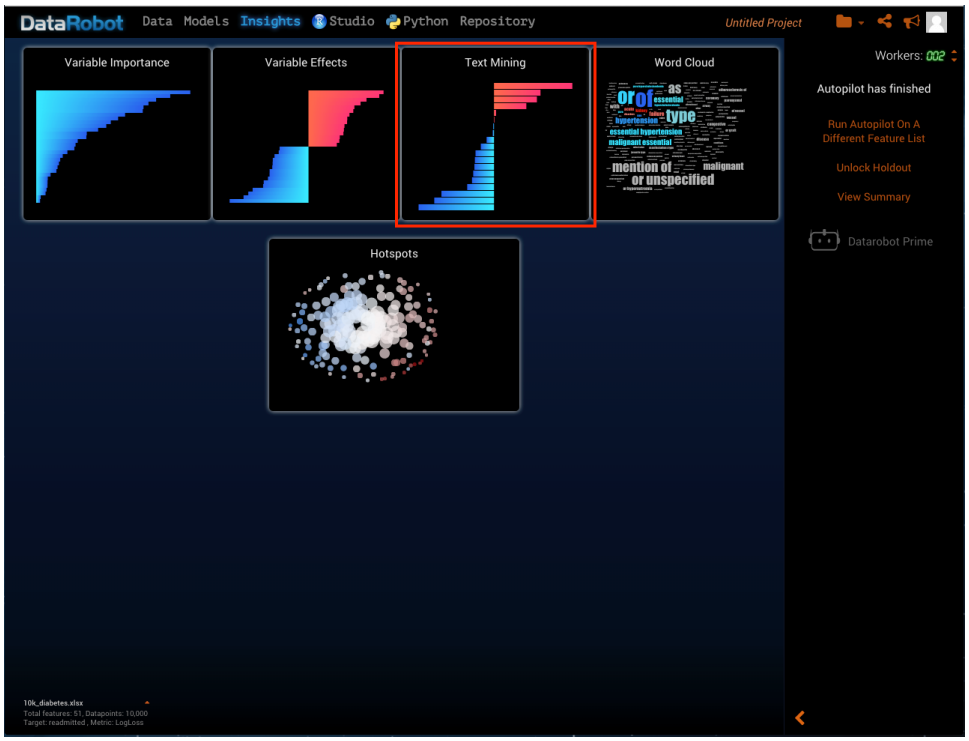
Variables with a positive effect are displayed in red, and those with a negative effect are shown in blue.

Text Mining

The Text Mining chart shows you the most relevant words and short phrases in any variables detected as text. Text variables often contain words that are highly indicative of the response.

The most important of these words and phrases are shown in the text mining chart, along with the coefficients that were built using these text strings in the model. This enables you to compare the relevant strength of the presence of these words and phrases. This side-by-side comparison is useful because individual words can be used in many different -- and sometimes counterinutitive -- ways, with many different implications for the response.

To display the Text Mining charts, select **Insights > Text Mining**.

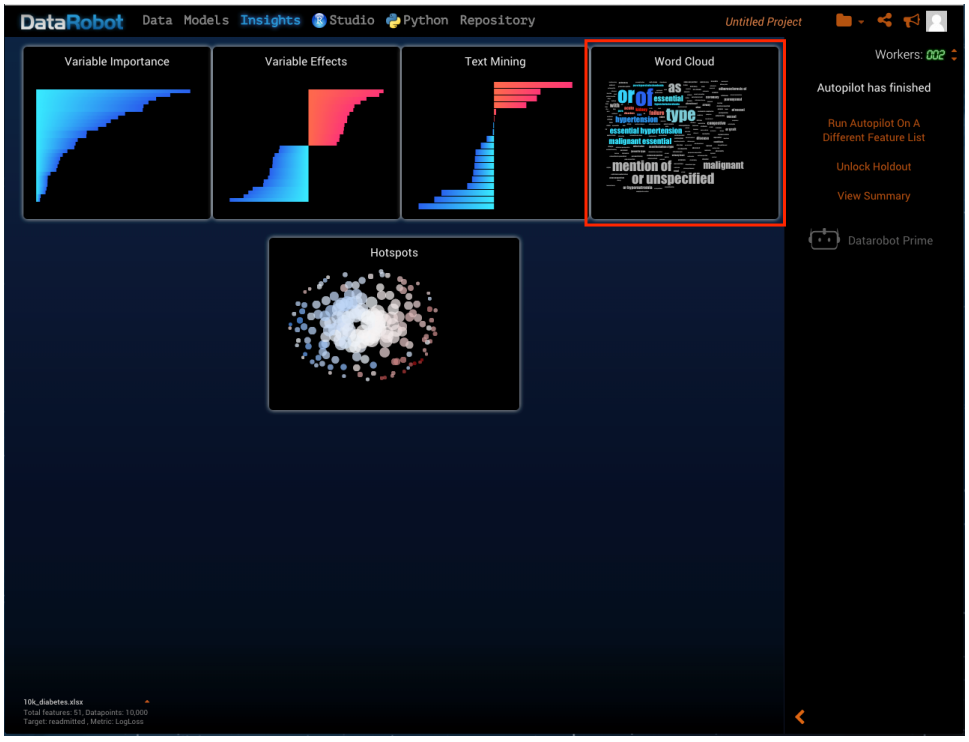


Text strings with a positive effect are displayed in red, and those with a negative effect are shown in blue. To view the Text Mining chart for another model, click the arrow next to the model name, then select a new model.

Word Cloud

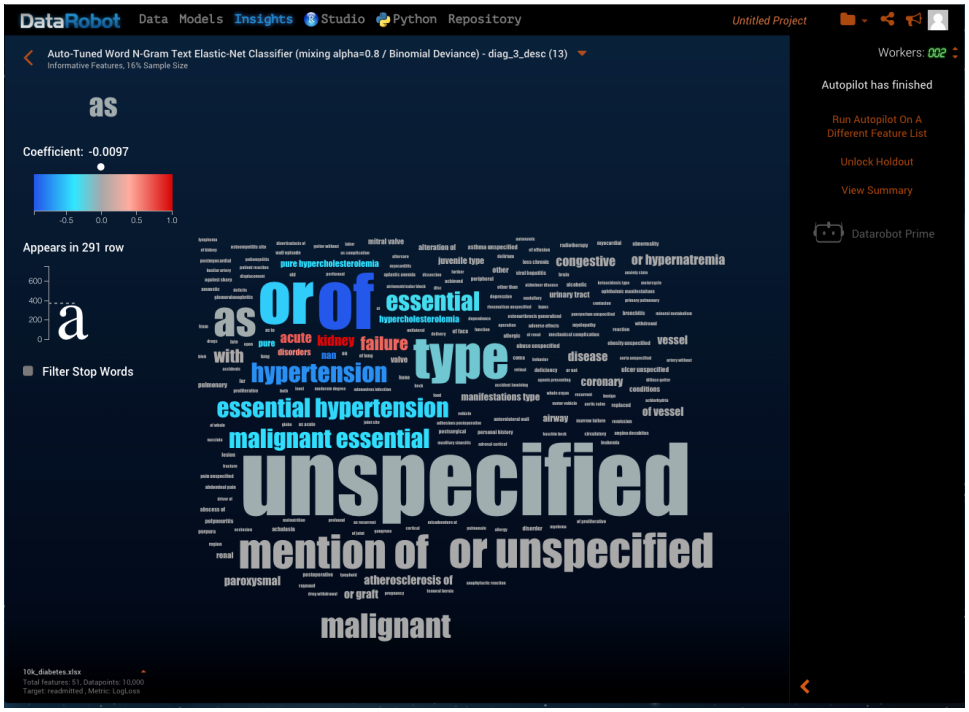
Word Cloud displays the most relevant words and short phrases in Word Cloud format. Text variables often contain words that are highly indicative of the response.

To display the Word Cloud, select **Insights > Word Cloud**.



Text strings are displayed in a color spectrum from blue to red, with blue indicating a negative effect ("cold"), and red indicating a positive effect ("hot"). Text strings that appear more frequently are displayed in a larger font size, and those that appear less frequently are displayed in smaller font sizes.

To view the Word Cloud for another model, click the arrow next to the model name, then select a new model.

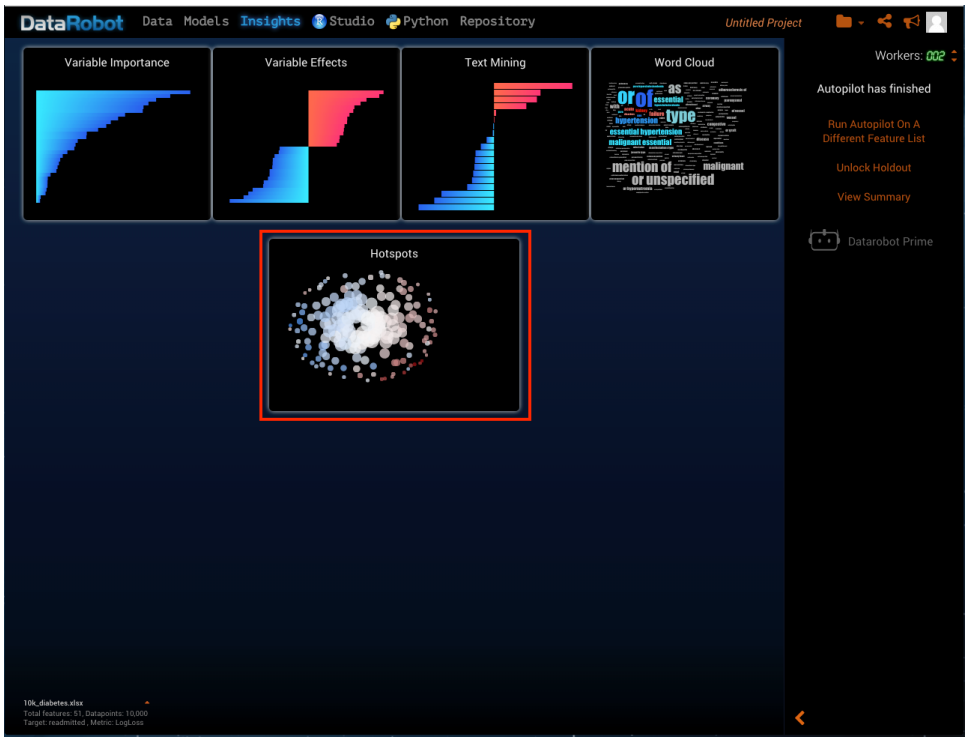


Hotspots

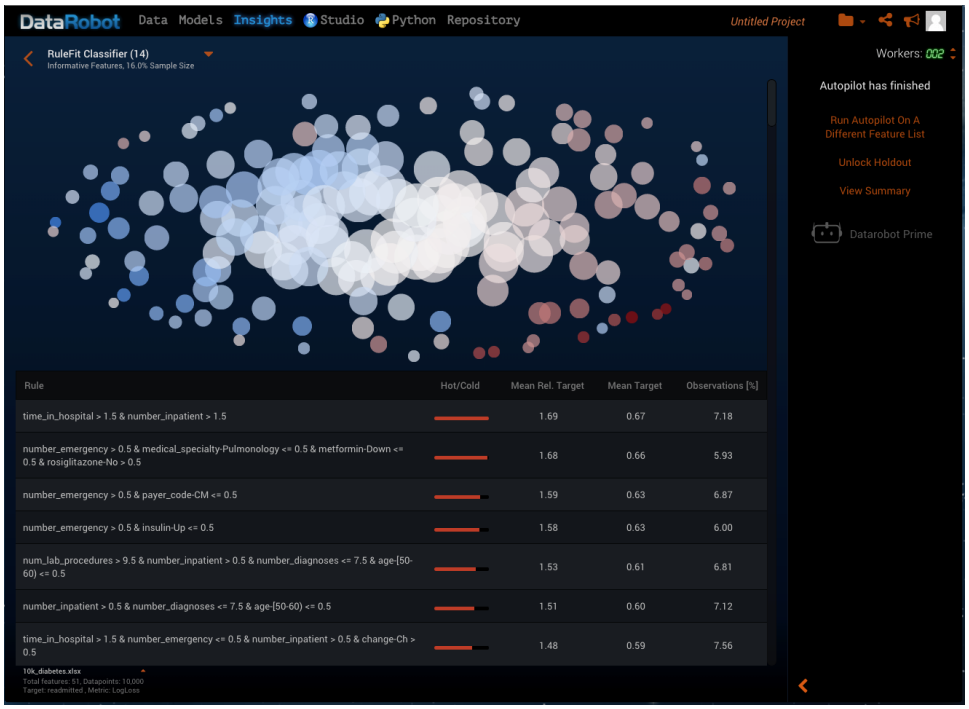
Hot and cold spots represent simple rules with high predictive performance that cover a large fraction of the data. Hotspot rules are good predictors for your data that can easily be translated and implemented as business rules.

The predictive performance of a rule is measured by the Mean Relative Response (MRR). This is the ratio between the mean response of the data points that the rule covers and the mean response of all data points. A MRR of 1 is the worst possible score. A score that deviates from 1 provides a lift with respect to a trivial regressor. Rules with a mean relative response (MRR) significantly higher than 1 are referred to as "hotspots", and rules with a MRR significantly lower than 1 are referred to as "coldspots".

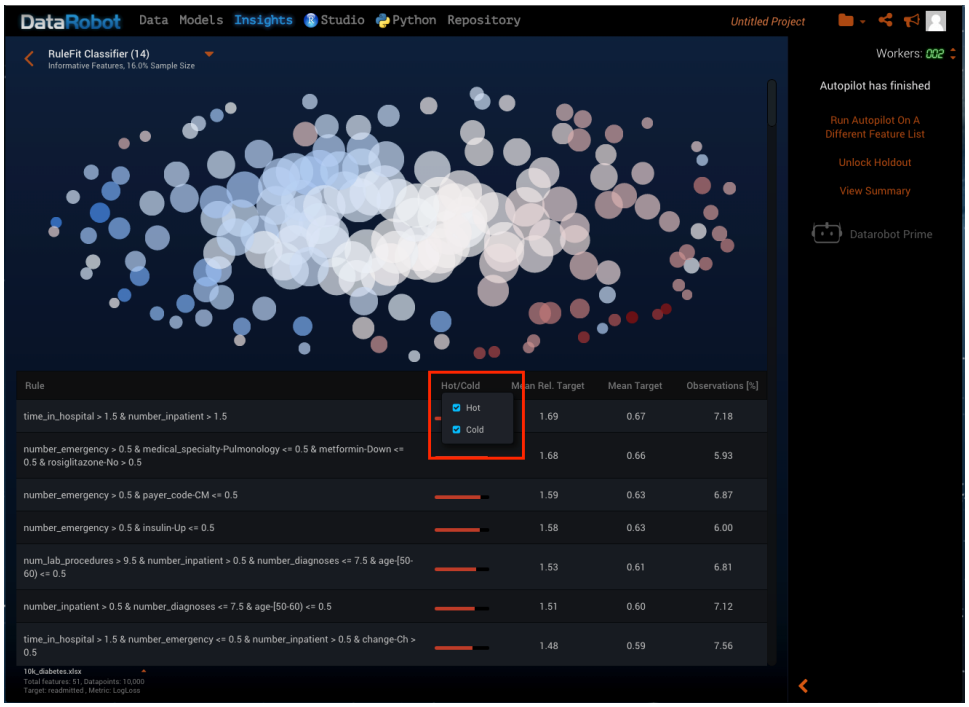
To display Hotspots, select **Insights > Hotspots**.



Hotspot rules are displayed in a color spectrum from blue to red, with blue indicating a negative effect ("cold"), and red indicating a positive effect ("hot"). Rules with a larger positive or negative effect are displayed as larger circles, and those with a smaller magnitude are displayed as smaller circles. Hotspot values are also displayed in a table.



To display only hotspots or coldspots, click **Hot/Cold**, then select or clear the **Hot** and **Cold** check boxes.

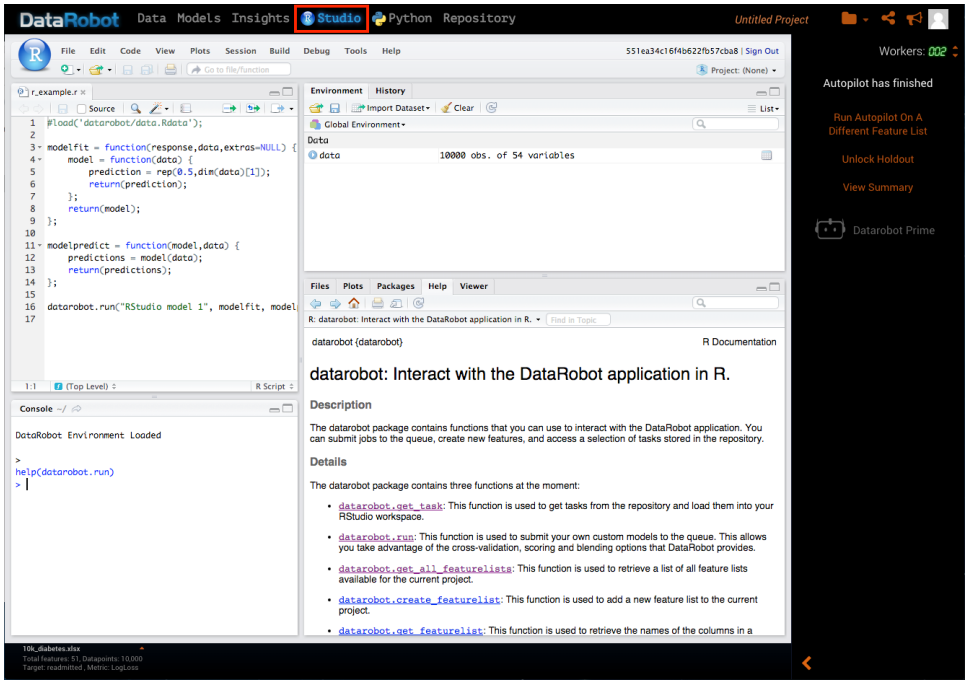


To view the Hotspots chart for another model, click the arrow next to the model name, then select a new model.

RStudio IDE

The RStudio Integrated Development Environment (IDE) enables you to run your own R models using DataRobot. After you run a model in the RStudio IDE, the results are listed on the model Leaderboard. The RStudio IDE includes documentation that describes the functions you can use to interact with DataRobot. You can submit jobs to the queue, create new features, and access models (stored as "Tasks") in the DataRobot repository.

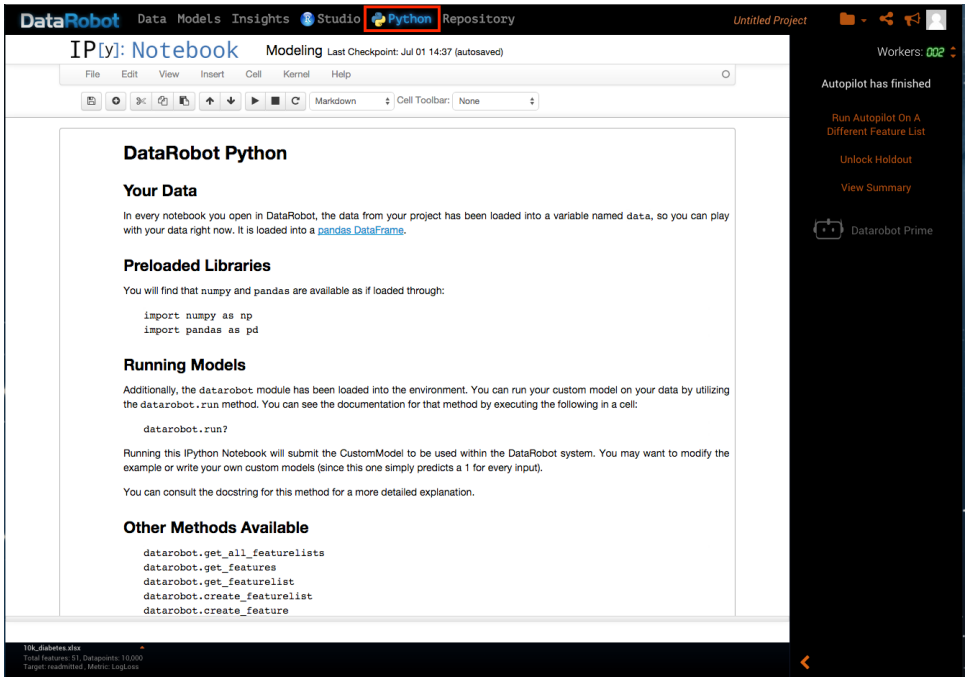
Click **RStudio** in the top menu to access the RStudio IDE.



Python IDE

The Python Integrated Development Environment (IDE) enables you to run models written in Python using DataRobot. After you run a model in the Python IDE, the results are listed on the model Leaderboard. The Python IDE includes documentation that describes the functions you can use to interact with DataRobot. You can submit jobs to the queue, create new features and feature lists, and access models (stored as "Tasks") in the DataRobot repository.

Click **Python** in the top menu to access the RStudio IDE.



Repository

The Python Integrated Development Environment (IDE) enables you to run models written in R and Python using DataRobot. After you run a model in the Python IDE, the results are listed on the model Leaderboard, and also stored in the DataRobot repository as "Tasks". You can use the repository to access your own custom models, as well as models shared with you by other DataRobot users.

To access the repository, click **Repository** in the top menu.