AI Workshop: Predict Bike Demand

In this AI workshop, you are going to build a model to predict the bike demand for a specific hour of a day for the city of Washington. The data is available as sample data in the Azure ML Studio (classic) and is based on the data that has been collected in 2011 and 2012 in Washington.

The dataset contains whether information and the number of bikes that have been rented. More information can be found on the UCI Machine Learning repository site: https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset

We highly recommend you visit that site and investigate what kind of data you have available.

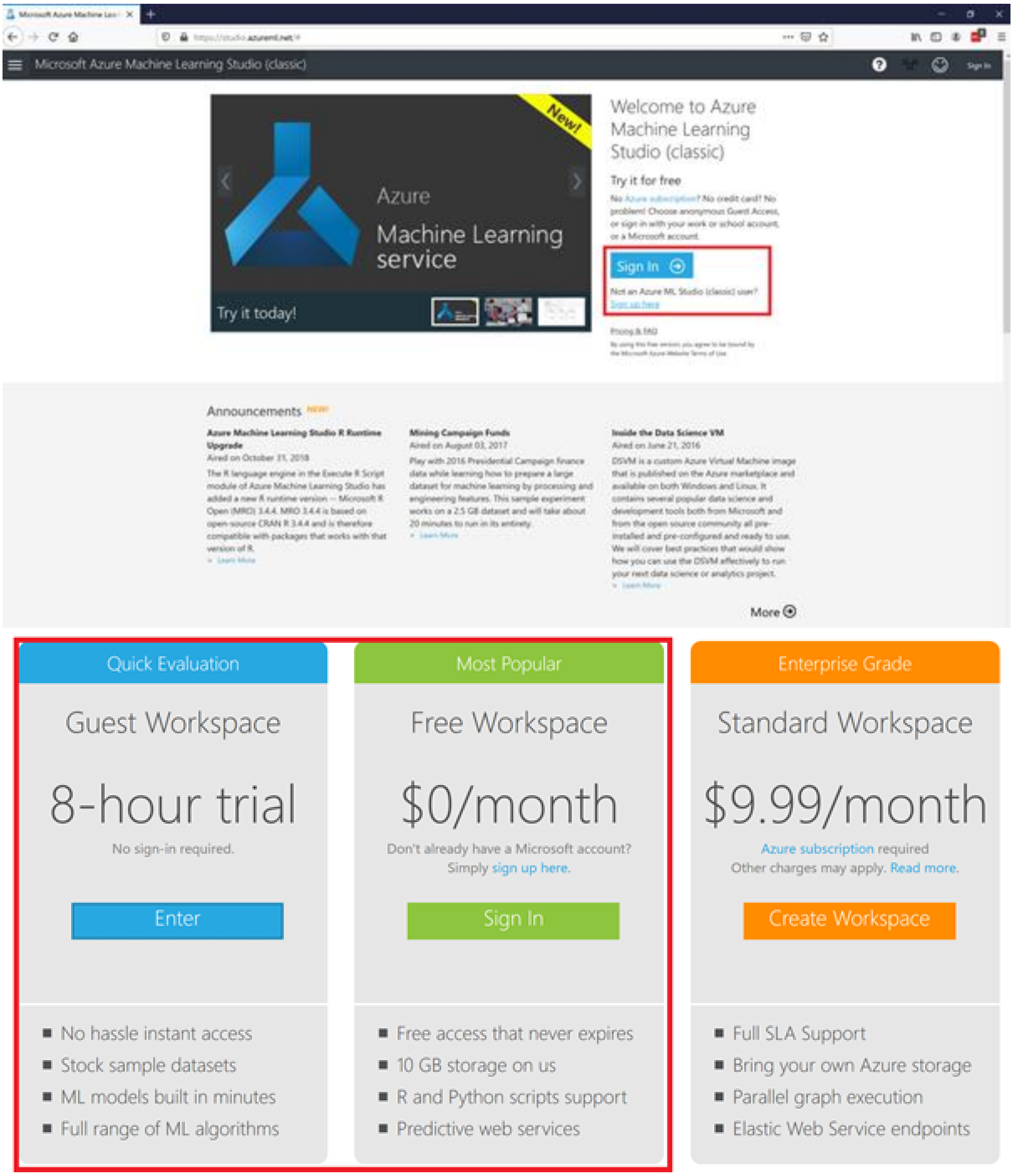
Note: this workshop is to get in touch with machine learning. We won't pretend to build an excellent model in 1 hour. In the real world you would have to do a lot more, but this workshop does give you an idea about the steps, with the free option of the Microsoft Azure Machine Learning Studio (classic).

Steps to build the model

You will first open the online environment where you will build your model. Then you will select the dataset and inspect the data. Next, you will select the required columns and transform them if needed. Then you will split the dataset into 2 parts: 1 part to train the model with, and 1 to test the model with. You will train the model and use this model to score the test dataset. Finally, you will evaluate the model.

Step 1: Get access to the environment

Go to https://studio.azureml.net/ and select **Sign up here** for Azure ML Studio. For this workshop, you can select the 8-hour trial.



Microsoft Azure Machine Lea
https://studio.azureml.net/
Microsoft Azure Machine Learning Studio (classic)
Sign In
New!
Azure
Machine Learning service
Try it today!

Welcome to Azure Machine Learning Studio (classic)

Try it for free

No Azure subscription? No credit card? No problem! Choose anonymous Guest Access, or sign in with your work or school account, or a Microsoft account.

Sign In

Not an Azure ML Studio (classic) user? Sign up here

Pricing & FAQ

By using this free version you agree to be bound by the Microsoft Azure Website Terms of Use

Announcements NEW!

Azure Machine Learning Studio R Runtime Upgrade
Aired on October 31, 2018
The R language engine in the Execute R Script module of Azure Machine Learning Studio has added a new R runtime version -- Microsoft R Open (MRO) 3.4.4. MRO 3.4.4 is based on open-source CRAN R 3.4.4 and is therefore compatible with packages that works with that version of R.
» Learn More

Mining Campaign Funds
Aired on August 03, 2017
Play with 2016 Presidential Campaign finance data while learning how to prepare a large dataset for machine learning by processing and engineering features. This sample experiment works on a 2.5 GB dataset and will take about 20 minutes to run in its entirety.
» Learn More

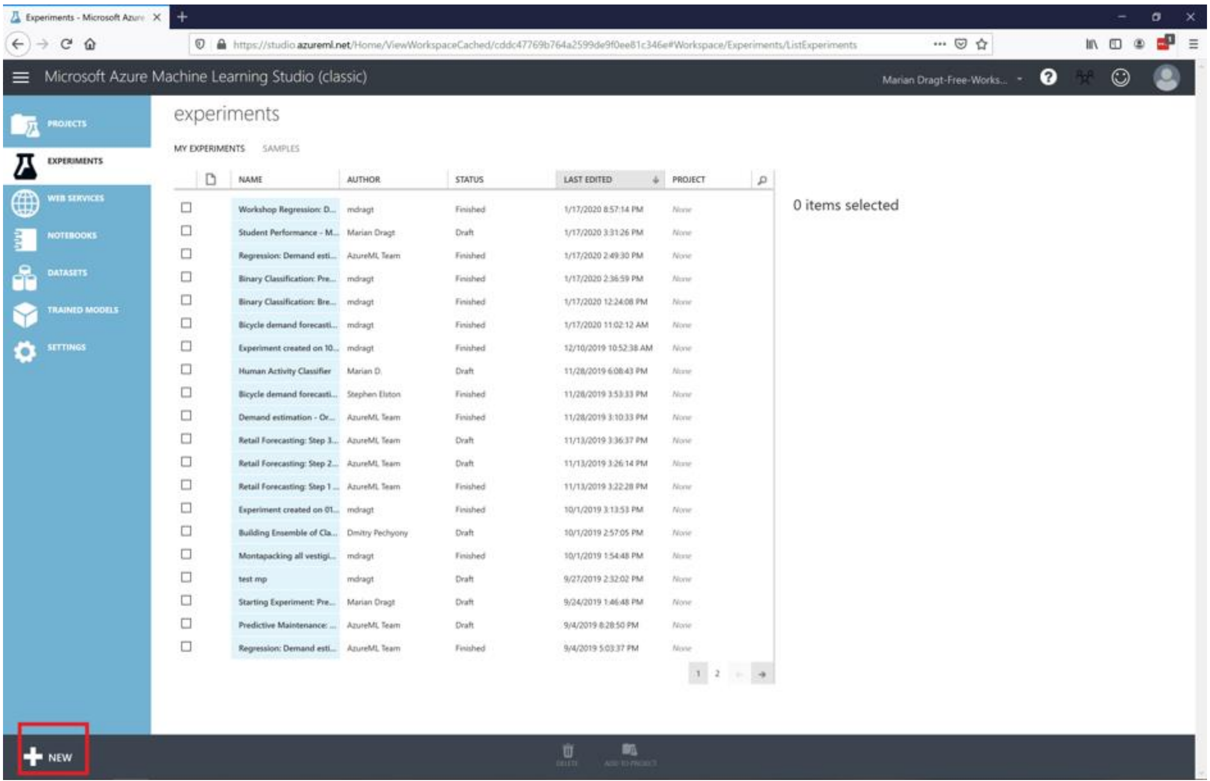
Inside the Data Science VM
Aired on June 21, 2016
DSVM is a custom Azure Virtual Machine image that is published on the Azure marketplace and available on both Windows and Linux. It contains several popular data science and development tools both from Microsoft and from the open source community all pre-installed and pre-configured and ready to use. We will cover best practices that would show how you can use the DSVM effectively to run your next data science or analytics project.
» Learn More

More

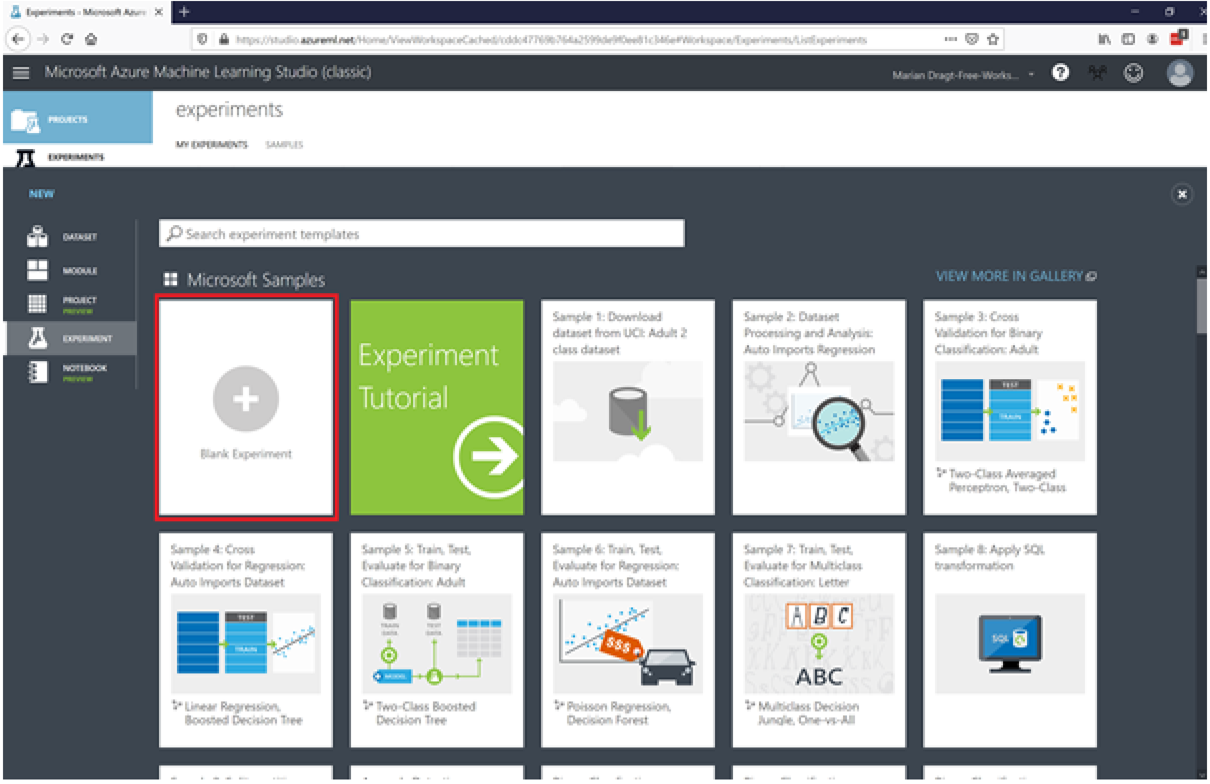
Quick Evaluation	Most Popular	Enterprise Grade
Guest Workspace	Free Workspace	Standard Workspace
8-hour trial	$0/month	$9.99/month
No sign-in required.	Don't already have a Microsoft account? Simply sign up here.	Azure subscription required Other charges may apply. Read more.
Enter	Sign In	Create Workspace
■ No hassle instant access ■ Stock sample datasets ■ ML models built in minutes ■ Full range of ML algorithms	■ Free access that never expires ■ 10 GB storage on us ■ R and Python scripts support ■ Predictive web services	■ Full SLA Support ■ Bring your own Azure storage ■ Parallel graph execution ■ Elastic Web Service endpoints

Step 2: Create a new blank Experiment and give it a name

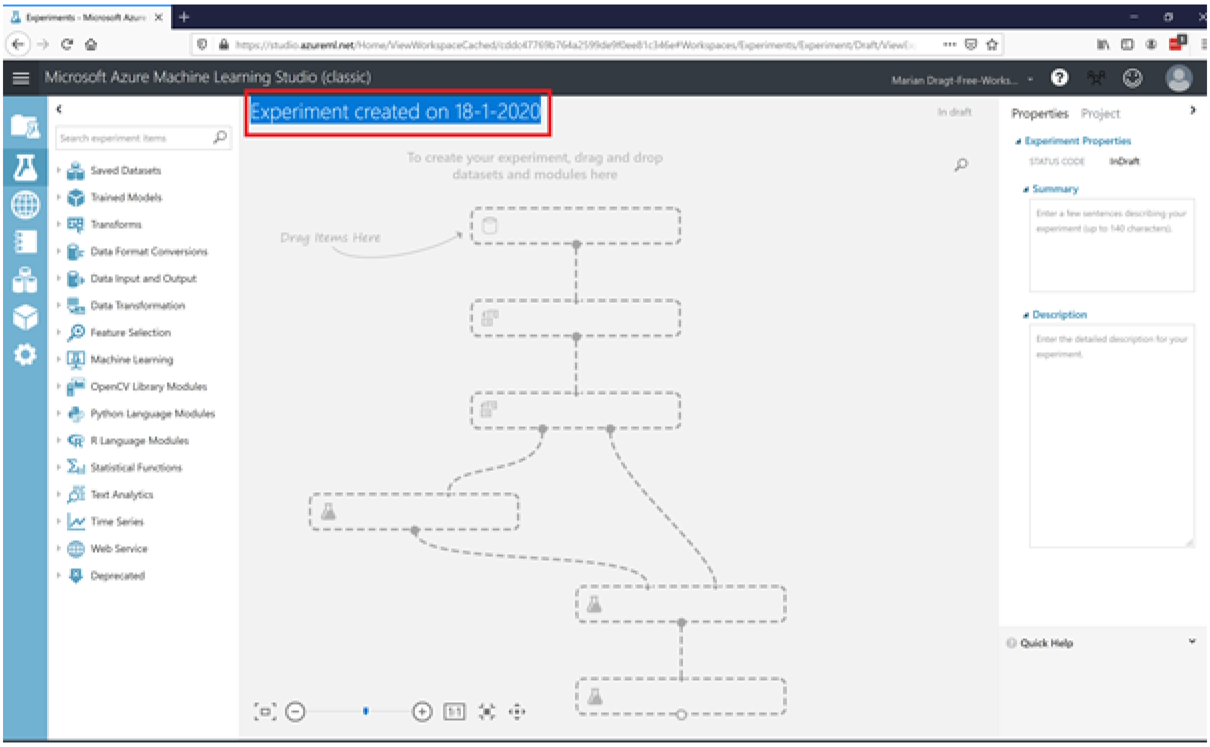
To build your model, you first have to create a new experiment. An experiment is like an instance of your model. It will open a canvas where you can drag your modules on to build your model and run it. Create a new blank experiment by clicking on the **+ NEW** button at the left bottom corner of the screen. Now select the **Blank Experiment**.


Experiments - Microsoft Azure
https://studio.azureml.net/Home/ViewWorkspaceCached/cddc47769b764a2599de9f0ee81c346e#Workspace/Experiments/ListExperiments
Microsoft Azure Machine Learning Studio (classic)
Marian Dragt-Free-Works...
PROJECTS
EXPERIMENTS
WEB SERVICES
NOTEBOOKS
DATASETS
TRAINED MODELS
SETTINGS
NEW
experiments
MY EXPERIMENTS
SAMPLES
0 items selected
1
2
DELETE
ADD TO PROJECT

	NAME	AUTHOR	STATUS	LAST EDITED	PROJECT
□	Workshop Regression: D...	mdragt	Finished	1/17/2020 8:57:14 PM	None
□	Student Performance - M...	Marian Dragt	Draft	1/17/2020 3:31:26 PM	None
□	Regression: Demand esti...	AzureML Team	Finished	1/17/2020 2:49:30 PM	None
□	Binary Classification: Pre...	mdragt	Finished	1/17/2020 2:36:59 PM	None
□	Binary Classification: Bre...	mdragt	Finished	1/17/2020 12:24:08 PM	None
□	Bicycle demand forecasti...	mdragt	Finished	1/17/2020 11:02:12 AM	None
□	Experiment created on 10...	mdragt	Finished	12/10/2019 10:52:38 AM	None
□	Human Activity Classifier	Marian D.	Draft	11/28/2019 6:08:43 PM	None
□	Bicycle demand forecasti...	Stephen Elston	Finished	11/28/2019 3:53:33 PM	None
□	Demand estimation - Or...	AzureML Team	Finished	11/28/2019 3:10:33 PM	None
□	Retail Forecasting: Step 3...	AzureML Team	Draft	11/13/2019 3:36:37 PM	None
□	Retail Forecasting: Step 2...	AzureML Team	Draft	11/13/2019 3:26:14 PM	None
□	Retail Forecasting: Step 1 ...	AzureML Team	Finished	11/13/2019 3:22:28 PM	None
□	Experiment created on 01...	mdragt	Finished	10/1/2019 3:13:53 PM	None
□	Building Ensemble of Cla...	Dmitry Pechyony	Draft	10/1/2019 2:57:05 PM	None
□	Montapacking all vestigi...	mdragt	Finished	10/1/2019 1:54:48 PM	None
□	test mp	mdragt	Draft	9/27/2019 2:32:02 PM	None
□	Starting Experiment: Pre...	Marian Dragt	Draft	9/24/2019 1:46:48 PM	None
□	Predictive Maintenance: ...	AzureML Team	Draft	9/4/2019 8:28:50 PM	None
□	Regression: Demand esti...	AzureML Team	Finished	9/4/2019 5:03:37 PM	None


Experiments - Microsoft Azure
https://studio.azureml.net/Home/ViewWorkspaceCached/cddc47769b764a2599de9f0ee81c346e#Workspace/Experiments/ListExperiments
Microsoft Azure Machine Learning Studio (classic)
Marian Dragt-Free-Works...
PROJECTS
EXPERIMENTS
experiments
MY EXPERIMENTS
SAMPLES
NEW
DATASET
MODULE
PROJECT
PREVIEW
EXPERIMENT
NOTEBOOK
PREVIEW
Search experiment templates
Microsoft Samples
VIEW MORE IN GALLERY
Blank Experiment
Experiment Tutorial
Sample 1: Download dataset from UCI: Adult 2 class dataset
Sample 2: Dataset Processing and Analysis: Auto Imports Regression
Sample 3: Cross Validation for Binary Classification: Adult
Two-Class Averaged Perceptron, Two-Class
Sample 4: Cross Validation for Regression: Auto Imports Dataset
Linear Regression, Boosted Decision Tree
Sample 5: Train, Test, Evaluate for Binary Classification: Adult
Two-Class Boosted Decision Tree
Sample 6: Train, Test, Evaluate for Regression: Auto Imports Dataset
Poisson Regression, Decision Forest
Sample 7: Train, Test, Evaluate for Multiclass Classification: Letter
Multiclass Decision Jungle, One-vs-All
Sample 8: Apply SQL transformation

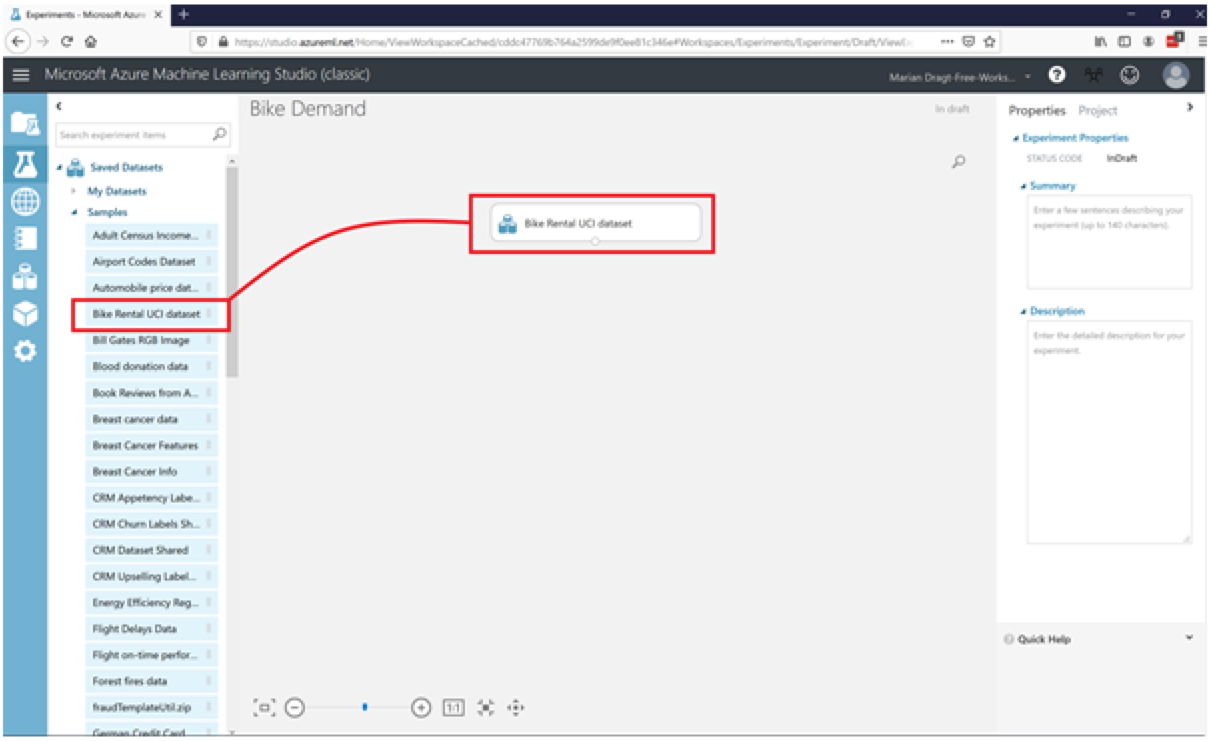
This will open a canvas where you can build your model. First give your model a name. You can select the title and change it.



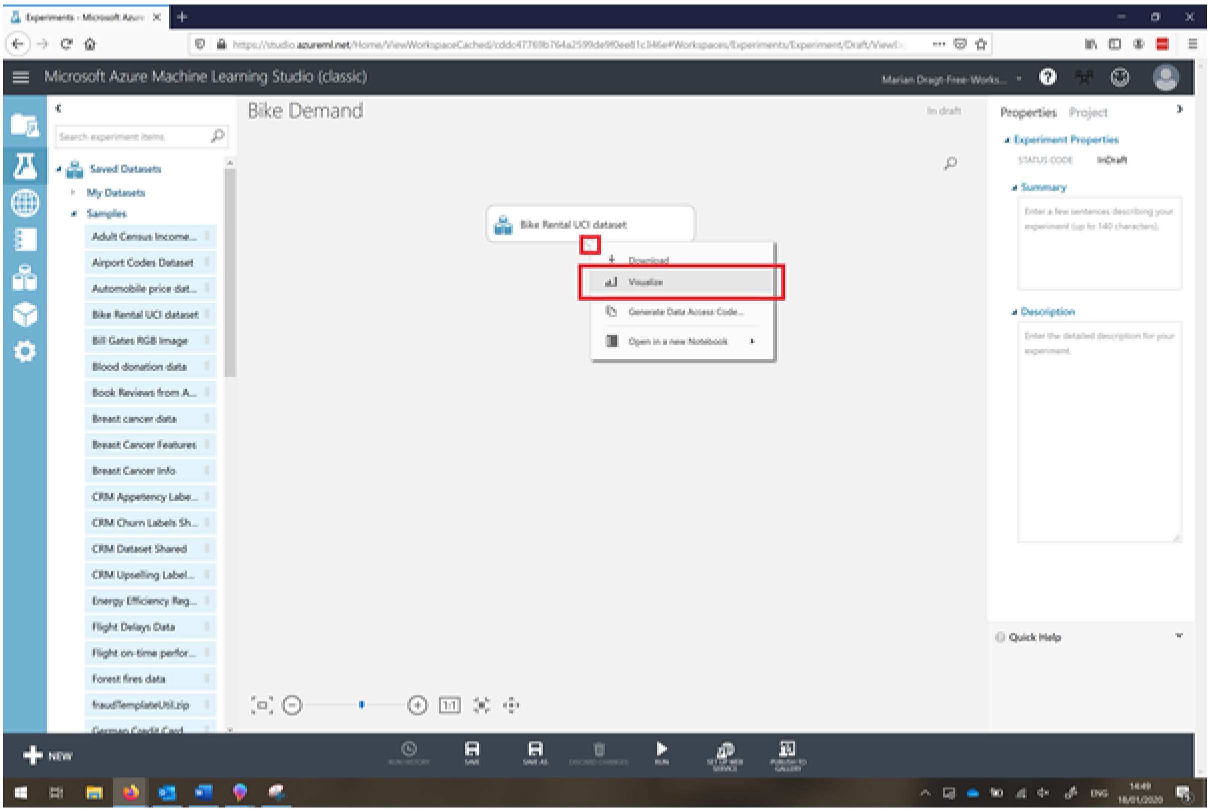
At the left, you have a menu will all kind of modules to build your model with.

Step 3: Get and visualize the data

We start with the data. You can open the **Saved Datasets** item, select the Bike Rental UCI dataset, and drag it on the canvas.



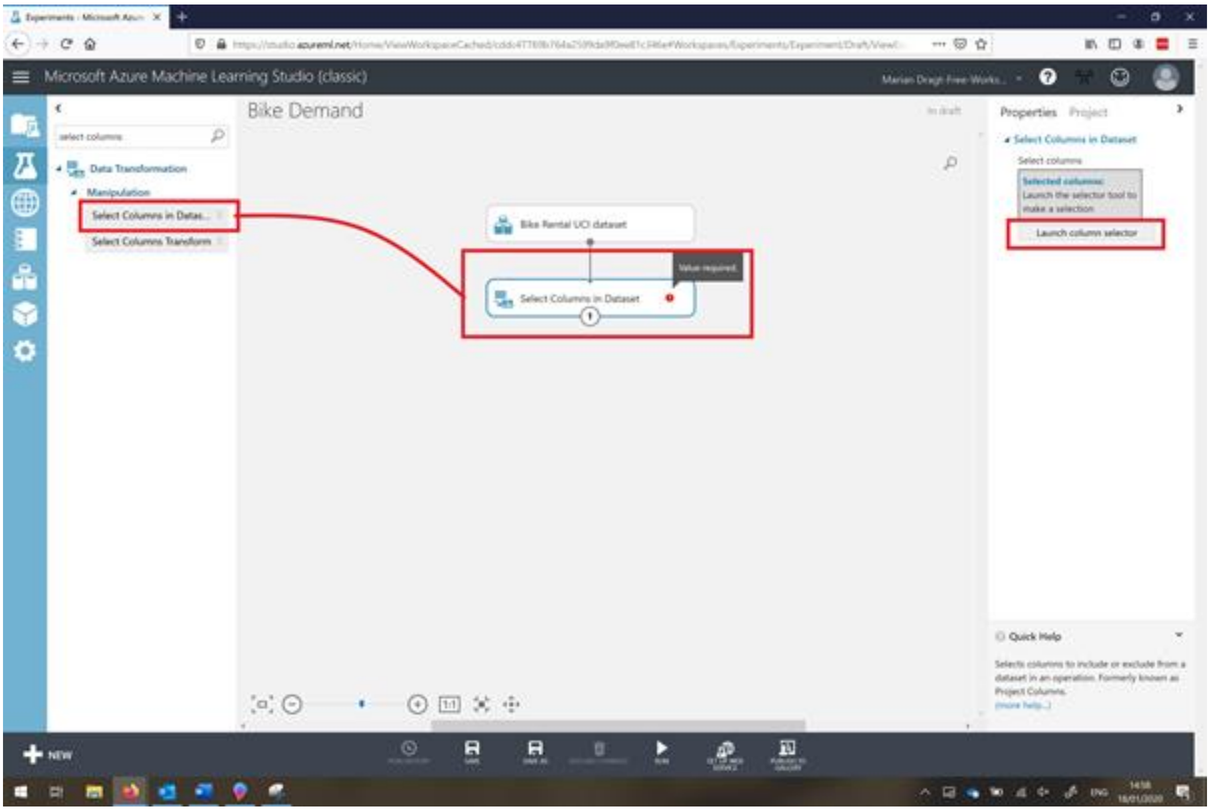
To get a quick overview of the data, you can right-click the output port and select the option **Visualize**. This will show you some quick insights regarding the data, like the amount of observations and variables, and the shape of the data.



As you could read at the UCI Machine Learning repository page, this dataset contains the count of usual users (column ‘casual’), registered users (column ‘registered’), and the total amount of rented bikes (column ‘cnt’). For this model we are only going to use the latter. Furthermore, we don’t need the instant variable.

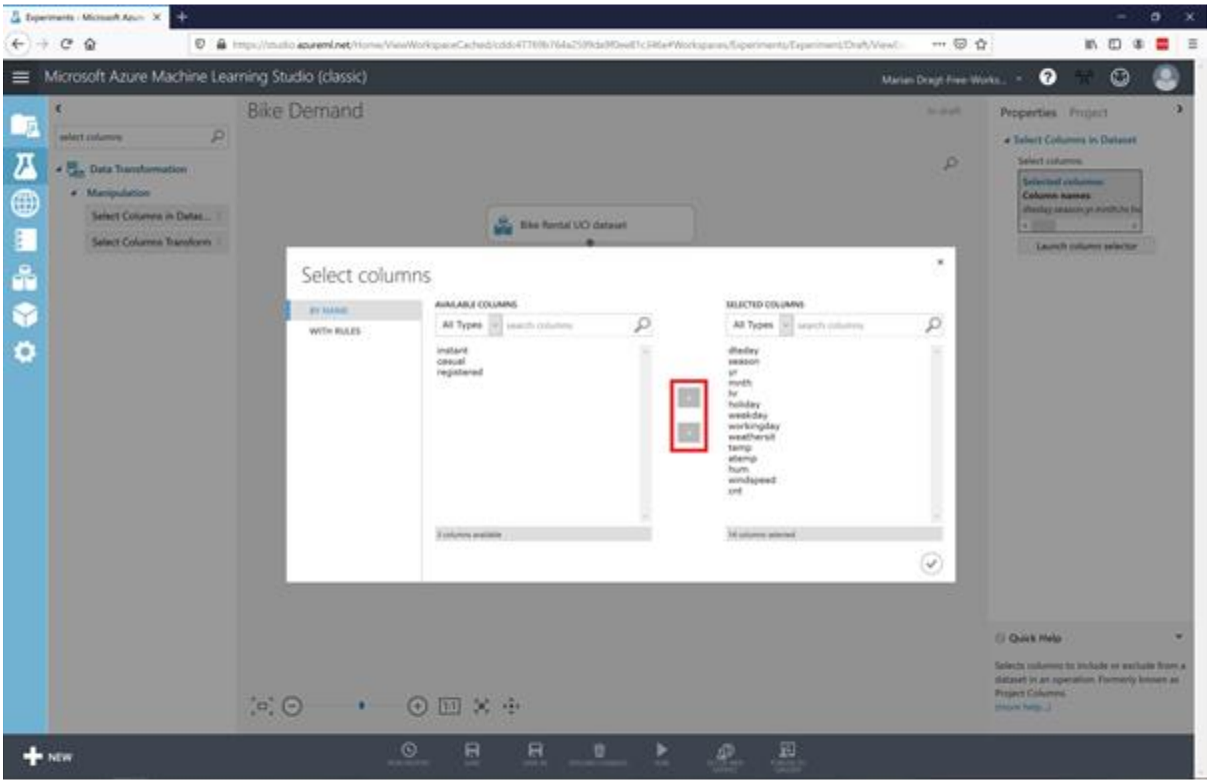
Step 4: Select the required data

To select the required columns, you can use the **Select Columns in Dataset** module, which you can find under Manipulation in the left menu. You can connect the output port of the Bike Rental UCI dataset module with the input port of the Select Columns in Dataset module (use your mouse to draw a line between the modules). You will see a red exclamation mark, because we haven’t informed the module which variables to use. Therefore, you can open the column selector at the right side of the screen.



Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.

You can now select your desired variables by using the arrows. Make sure all variables except instant, casual and registered are in the 'selected column' (see picture below). Click on the ok sign right below when you are ready.



Available columns:

instant
casual
registered

3 columns available

Selected columns:

dteday
season
yr
mnth
hr
holiday
weekday
workingday
weathersit
temp
atemp
hum
windspeed
cnt

14 columns selected

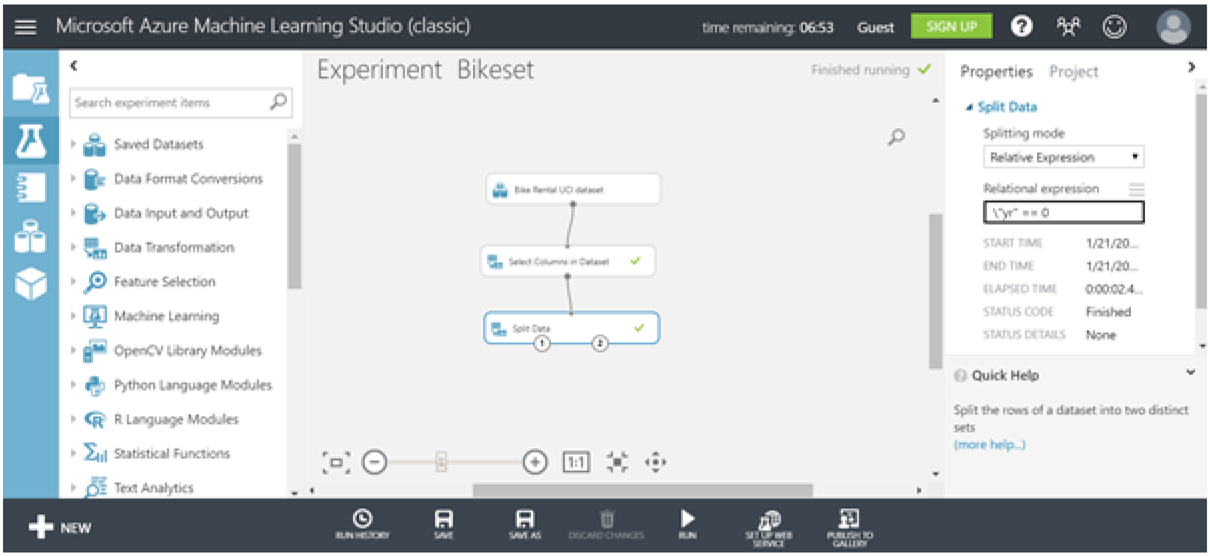
In order to see the results, you have to **SAVE** and **RUN** the model. You can find these options at the bottom menu of the page.

Step 5: Splitting the data

Now you are ready to split the dataset into a training dataset and a test dataset. You will train the model with 2011 data and test the model with 2012 data. Select the **Split Data** module and drag it on the canvas. Connect the output port of the Select Columns in Dataset module to the input port of the Split Data module. At the right, you can configure this module. In this case, we can use an expression to select 2011:

```
\"yr" == 0
```

Make sure you also set the correct Splitting mode. This will give you the 2011 data in the left (1) output port to train your model with, and the right (2) output port will contain the 2012 data to test your model with later.



Experiment Bikeset

Finished running ✓

Properties Project

Split Data

Splitting mode: Relative Expression

Relational expression: \"yr" == 0

START TIME	1/21/20...
END TIME	1/21/20...
ELAPSED TIME	0:00:02.4...
STATUS CODE	Finished
STATUS DETAILS	None

Quick Help

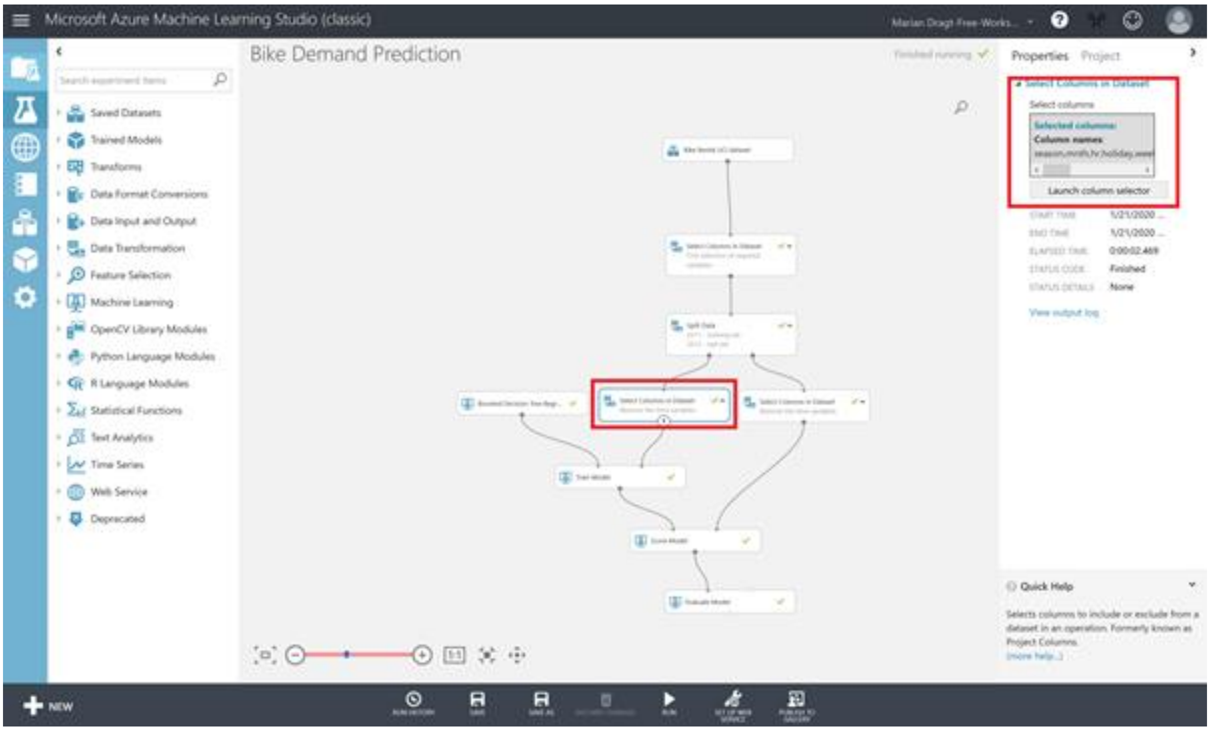
Split the rows of a dataset into two distinct sets

(more help...)

RUN the model and check your output data.

Step 6: Make sure you have all the required variables

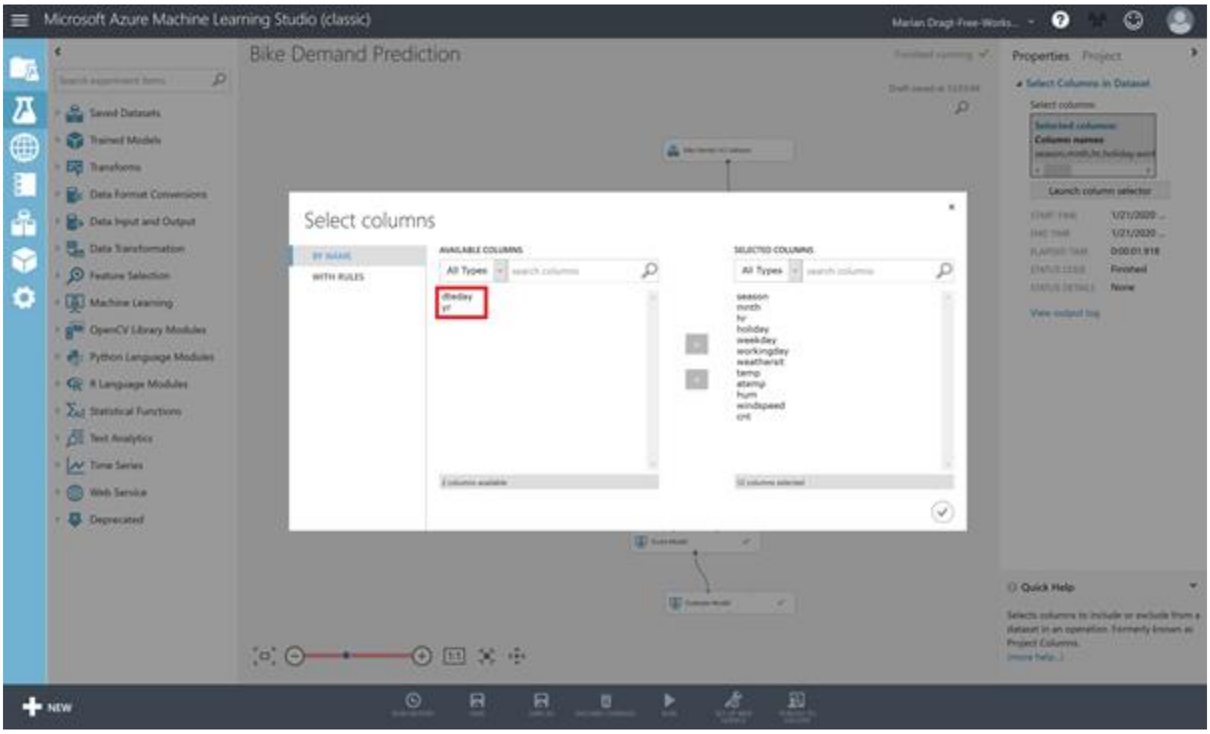
Now we can remove the last variables that we don't need anymore. We remove the year variable and the date variable. You can use the same procedure as before, by adding the **Select Columns in Dataset** module and connecting the Split Data module to it.



Select columns
Selected columns:
Column names: season,mnth,hr,holiday,weel

Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.

You can now select the variables that you need and leave those that you don’t.



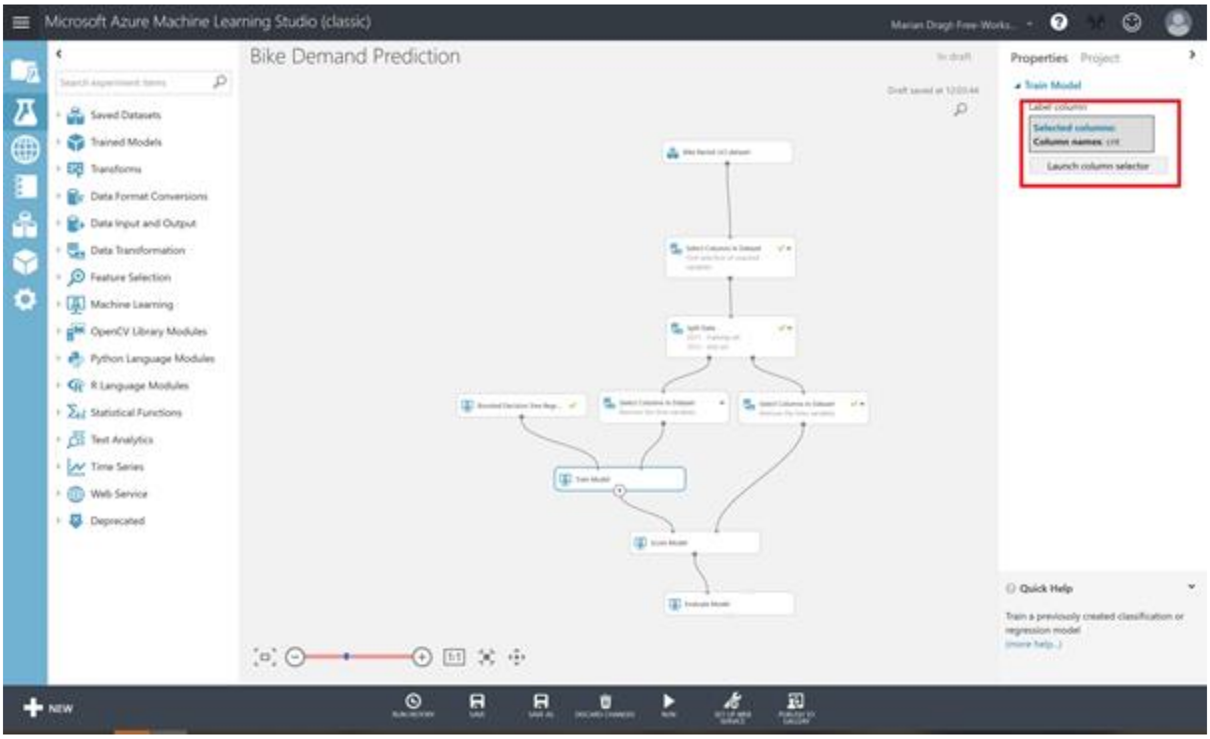
AVAILABLE COLUMNS
dteday
yr

SELECTED COLUMNS
season
mnth
hr
holiday
weekday
workingday
weathersit
temp
atemp
hum
windspeed
cnt

Make sure all variables except dteday and yr are in selected columns. Repeat this for both outputs of the Split Data module. You can copy-paste a module (right-click on a module) to lower your effort.

Step 7: Train the model

You are now ready to train the model. You need the **Train Model** module, an algorithm module, and the training dataset.



Train Model

Label column

Selected columns:
Column names: cnt

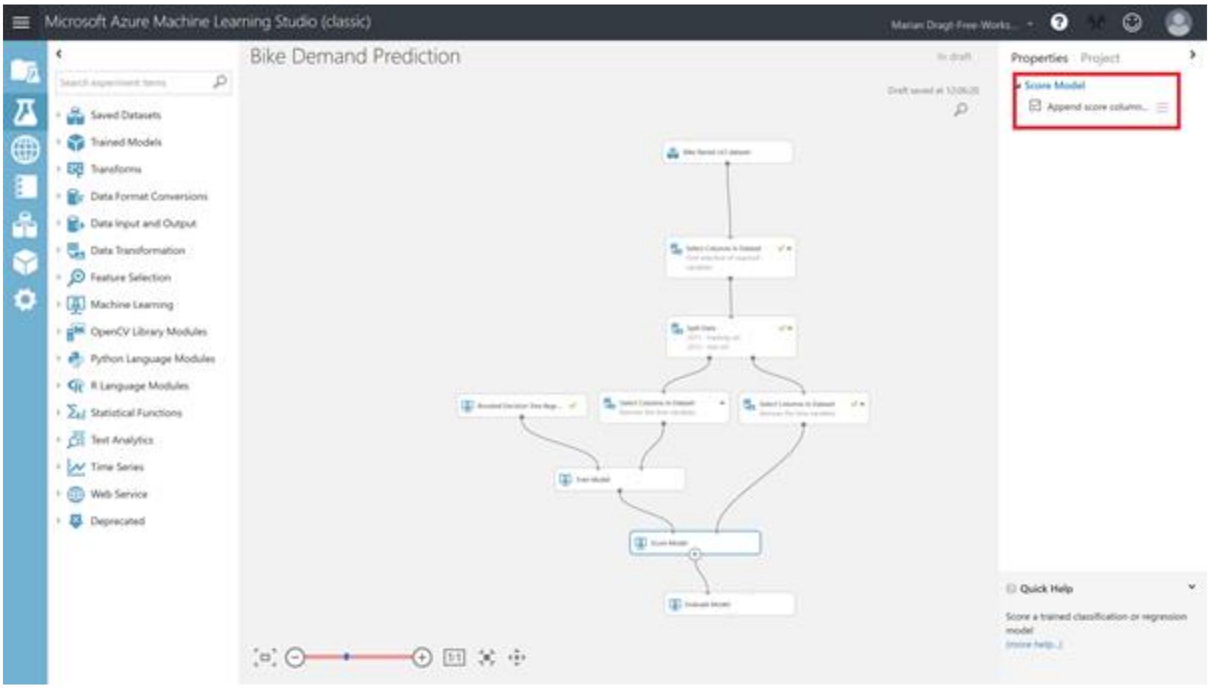
Launch column selector

Train a previously created classification or regression model
(more help...)

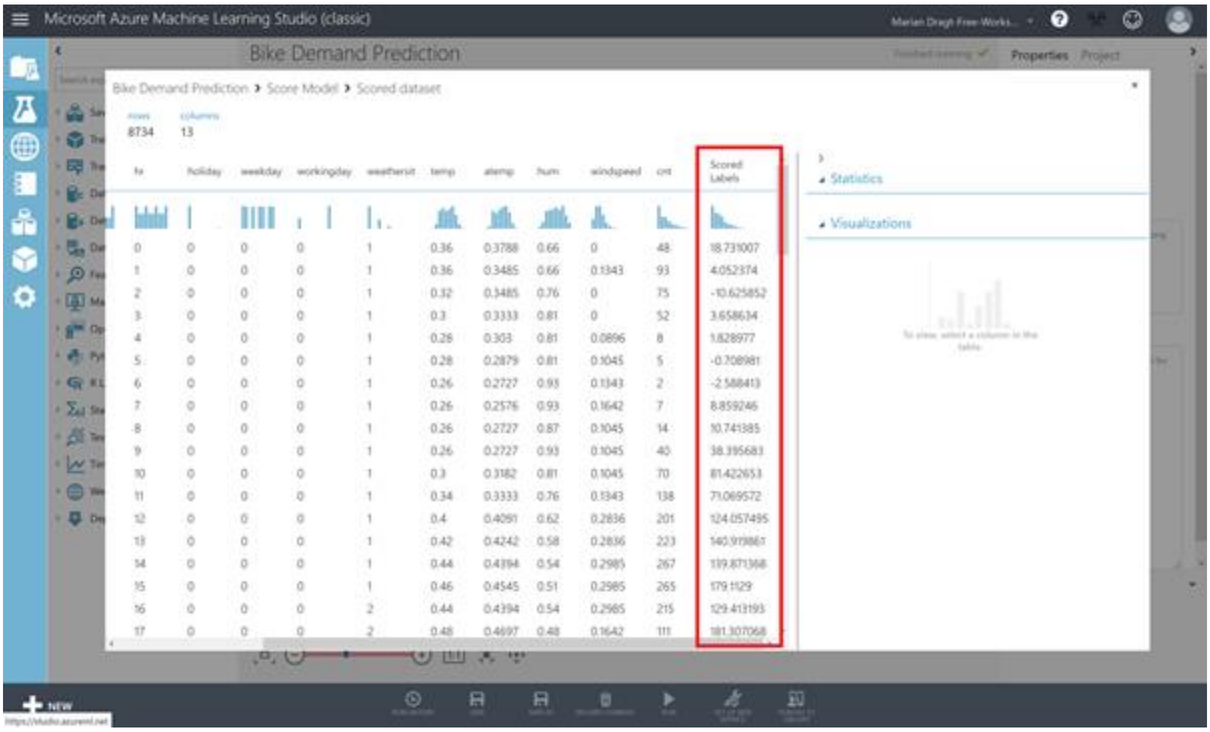
Drag the Train Model module on the canvas and connect the training dataset to it. Besides, drag the **Boosted Decision Tree Regression** algorithm on the canvas and connect it to the Train Model module. You can leave the pre-set hyperparameters as they are. In the Train Model module, make sure you select the dependent variable “cnt” to train the model on. Save and run your model.

Step 8: Test the model

Now you have trained the model, and it’s time to test is. You can use your model to score the test dataset by dragging the **Score Model** module on the canvas and connecting it to the test dataset. By default, the results will be appended to the dataset. Save and run your model.



If you inspect the output of the Score Model module, by righ-clicking on the output port and selecting Visualize, you will see that there is an extra column in your dataset, named Scored Labels. This column contains the predicted number of rented bikes.



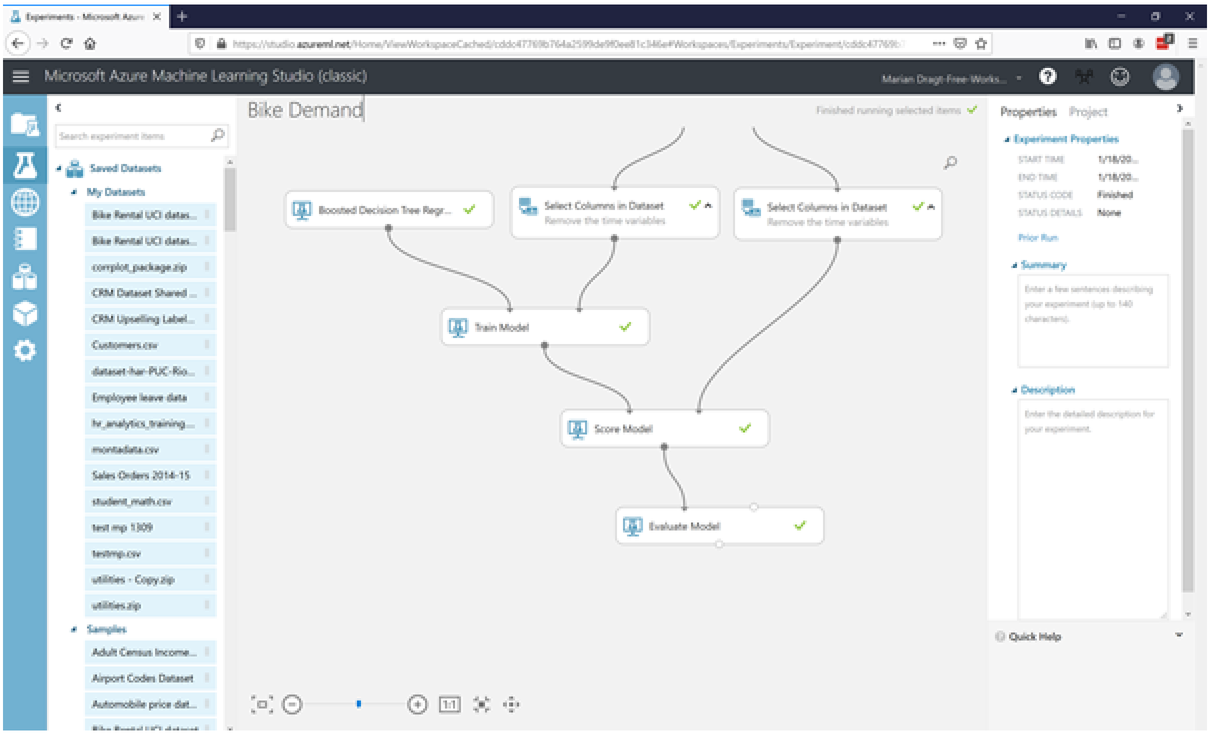
Bike Demand Prediction › Score Model › Scored dataset

rows 8734 columns 13

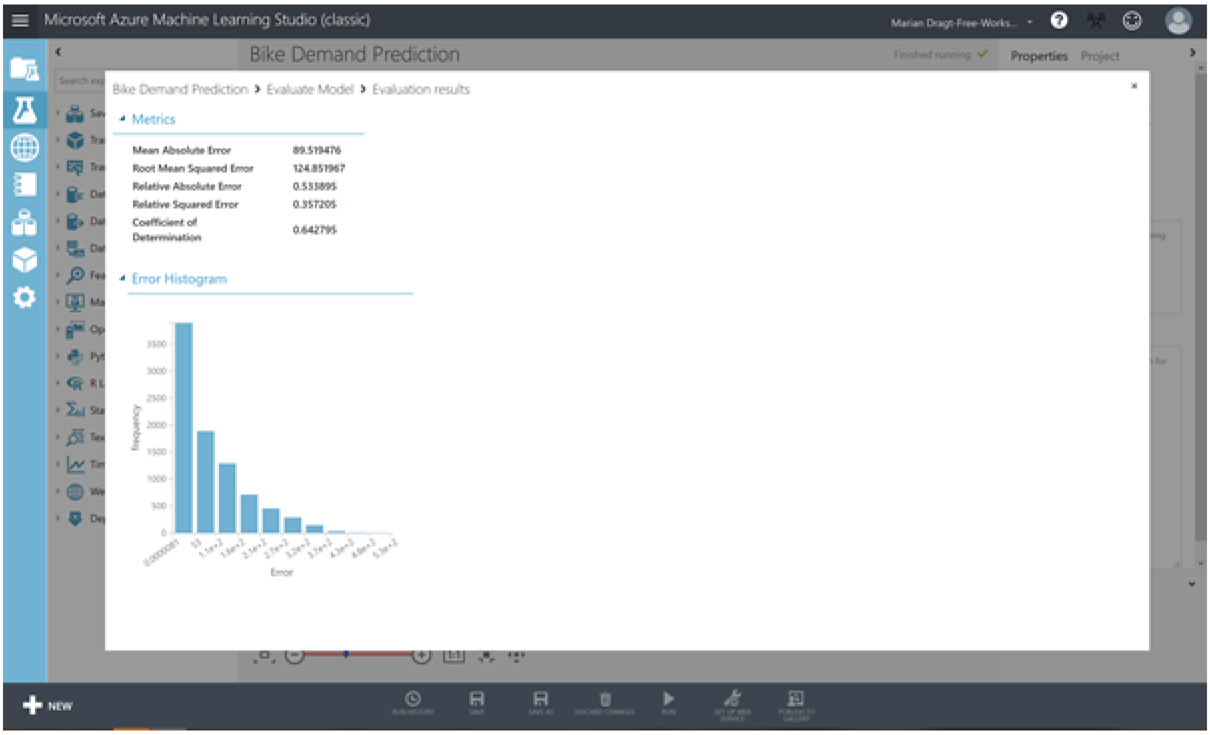
hr	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	cnt	Scored Labels
0	0	0	0	1	0.36	0.3788	0.66	0	48	18.731007
1	0	0	0	1	0.36	0.3485	0.66	0.1343	93	4.052374
2	0	0	0	1	0.32	0.3485	0.76	0	75	-10.625852
3	0	0	0	1	0.3	0.3333	0.81	0	52	3.658634
4	0	0	0	1	0.28	0.303	0.81	0.0896	8	1.828977
5	0	0	0	1	0.28	0.2879	0.81	0.1045	5	-0.708981
6	0	0	0	1	0.26	0.2727	0.93	0.1343	2	-2.588413
7	0	0	0	1	0.26	0.2576	0.93	0.1642	7	8.859246
8	0	0	0	1	0.26	0.2727	0.87	0.1045	14	10.741385
9	0	0	0	1	0.26	0.2727	0.93	0.1045	40	38.395683
10	0	0	0	1	0.3	0.3182	0.81	0.1045	70	81.422653
11	0	0	0	1	0.34	0.3333	0.76	0.1343	138	71.069572
12	0	0	0	1	0.4	0.4091	0.62	0.2836	201	124.057495
13	0	0	0	1	0.42	0.4242	0.58	0.2836	223	140.919861
14	0	0	0	1	0.44	0.4394	0.54	0.2985	267	119.871368
15	0	0	0	1	0.46	0.4545	0.51	0.2985	265	179.1129
16	0	0	0	2	0.44	0.4394	0.54	0.2985	215	129.413193
17	0	0	0	2	0.48	0.4697	0.48	0.1642	111	181.307068

Step 9: Evaluate the model

As we also have the real number of rented bikes, we can make the evaluation. Luckily for us, there is a module that does the trick. Drag the **Evaluate Model** module on the canvas, connect it, and run your model. The Evaluate Model module has 2 input ports so you can compare models with each other. As we have only one model, make sure you connect it to the left input port.



After you have ran the model, you can inspect the results by right-clicking on the output port and choosing the Visualize options.



Bike Demand Prediction › Evaluate Model › Evaluation results

Metrics

Mean Absolute Error	89.519476
Root Mean Squared Error	124.851967
Relative Absolute Error	0.533895
Relative Squared Error	0.357205
Coefficient of Determination	0.642795

Error Histogram

With this model, you can explain 64% of the variance in bike rental amount. Is that good enough? Well, that depends….

Step 10: Be sharp!

And here we come to the part where we look at the limitations and recommendations for future research.

First of all, we have skipped many steps which you would normally take when it comes to building a model. To start with, we would recommend you to understand the business question behind all this. Is predicting the number of rental bikes for a complete city really tell you something? Which business purpose are your serving? If we look at the original data source, you will find out that the original data also had the starting location and the ending location of the rental period. Now that starts to make sense: if we would be able to predict the amount of bikes for a specific location in the city, it could help the business to better stock their locations. Furthermore, we have skipped all the "normal" required steps for doing a regression: checking the distribution of the variables, the relations between them, outliers, etc. Now this was on purpose as this is a short workshop, but please make sure you do this right when you build a real model. Furthermore, based on the variables we have, we could create more features like i.e. amount of bikes rented the prior day, etc. This is up to your creativity.

Although there are quite some critical notes, we hope you have enjoyed this workshop and hopefully it inspired you to build your own models. If you want to take your models into production, then please use another environment: https://ml.azure.com/

Here you can find a similar interface, called Designer, but with this interface, you can also deploy and manage your models.