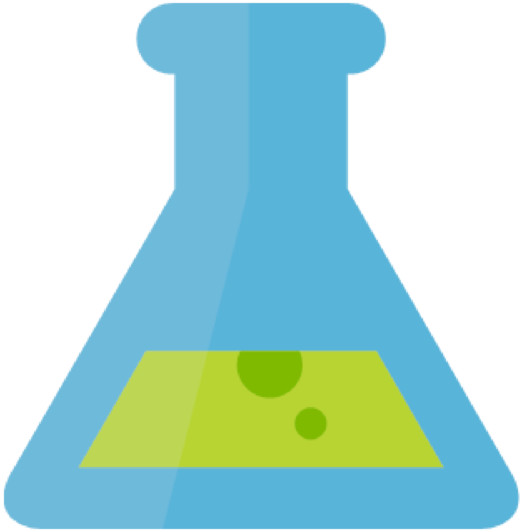
Azure Machine Learning



Hello!

Il cielo è Azure sopra Berlino team

Fabio Rosato - rosato.1565173@studenti.uniroma1.it
Giacomo Lanciano - lanciano.1487019@studenti.uniroma1.it
Francisco Ferreres Garcia - matakukos@gmail.com
Leonardo Martini - martini.1722989@studenti.uniroma1.it
Simone Caldaro - caldaro.1324152@studenti.uniroma1.it
Na Zhu - nana.zhu@hotmail.com

Università degli Studi di Roma “La Sapienza”
MoS in Engineering in Computer Science
Data Mining course
A.Y. 2016/2017

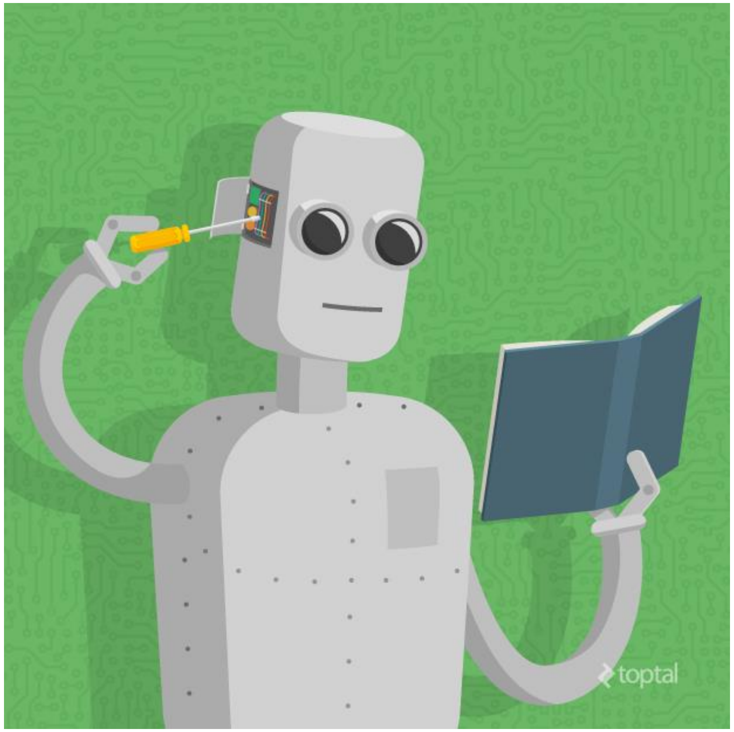
1.

Machine Learning

A brief overview of what (the hell) ML means

Machine Learning

- What is Big Data?
- What is Machine Learning?
- Uses of Machine Learning?
- Why Machine Learning?
- Who uses it?

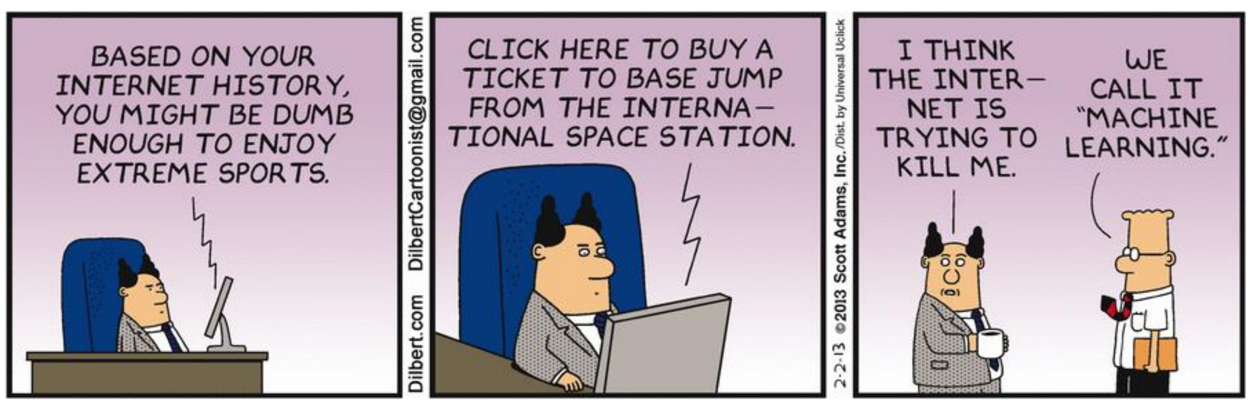


What is big data?

- What is Big Data?
  - Structured
  - Unstructured
- From a variety of sources
  - Commercial transactions
  - Social media
  - Publicly available sources
  - Sensors
  - Business statistics
- How to analyze this data?

What is machine learning?

- Examine LARGE amounts of data
  - Find patterns. Build models.
- Automatic improvement of the algorithms
  - Iterative approach.
  - Multiple passes so the machine *learns*.
- Predictions



Uses of machine learning?

- Classification
  - Supervised.
  - e.g. spam filter
- Regression
  - Supervised.
  - Estimate relationship between continuous variables.
  - e.g. car market price from specs
- Clustering
  - Unsupervised.
  - e.g. identify communities in social networks

Why machine learning?

- Growing volumes and varieties of available data
  - Processing this data manually would be impossible.
- Cheaper computational processing and storage
- Competitive advantage
  - Companies get huge benefits by analyzing data from the markets.

Who uses it?

- Financial institutions
  - e.g. recognize and prevent frauds.
- Governments
  - e.g. increase efficiency and service.
- Medicine and science
  - e.g. dna sequencing, patients wearable sensors.
- Marketing and sales
  - e.g. dna sequencing, patients wearable sensors.
- *You name it!*

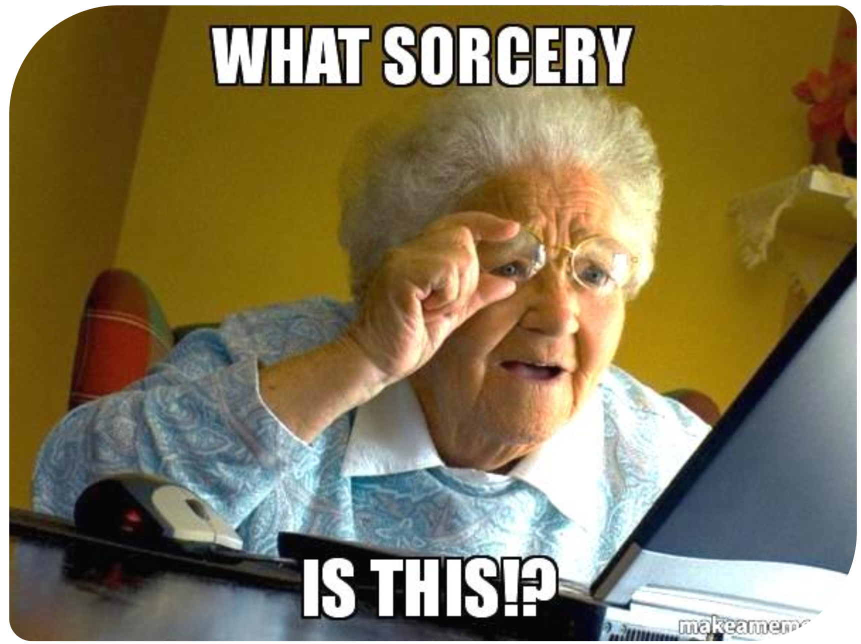
2.

Using ML

A brief overview of the current tools to harness the power of ML

ML is an incredibly powerful set of…

- Algorithms
- Tools
- Techniques
- …
- Magic spells?!



“

Keep it simple,
so you’ll keep doing it.

Back in the ol' days...

To use ML, you'd have to implement the algorithms yourself:

- prototype in some kind of friendlier language (like Matlab/Octave);
- then implement it in a *real language* (like C++) for speed and efficiency.

Back in the ol' days...

In-depth knowledge of ML techniques and algorithms was ***required***.

Huge barrier to adoption.

ML was used only in very big, very serious applications (that could afford and justify the *overhead*).

Tools to the rescue!

Libraries and frameworks lower the initial effort required to get a working prototype.

ML libraries and frameworks

- Exist for practically any widely used programming language.
- Encapsulate most widely used algorithms, abstracting away low-level details.
- Can even offer ad-hoc solutions for greater speed/efficiency/reliability (e.g. *distributed computation*).

ML libraries and frameworks
The celebrities:

scikit learn

TensorFlow™

mlpack

mahout

Spark

MLlib

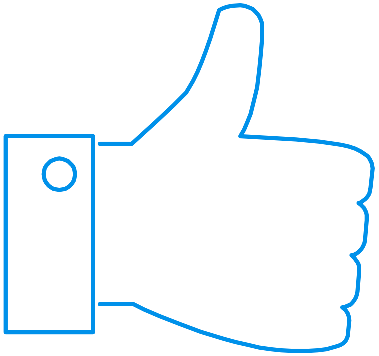
MLaaS

Cloud Computing approach gives us Machine Learning as a Service.

ML as a Service

Outsourcing ML services:

- Incredibly low barrier to adoption.
- Massive scalability.
- *It just works!*



ML as a Service - The celebrities:

- *Google* ***Prediction APIs***
- *Amazon* ***AWS ML***
- *Microsoft* ***Azure ML***
  - Allows users to create and train models,
    then turn them into ready-to-be-consumed APIs.
    All through a beautifully intuitive web interface.

3.

Azure Machine Learning Studio

Azure’s solution to make your own experiments

What is Azure Machine Learning Studio?

- Web-based workspace.
- Drag-and-drop tool.
- Collaborative environment.
- Where data science, cloud resources, and your data meet.

With Azure ML, predictive analytics solutions are...

Easy to build.
Easy to deploy.
Easy to share.

Ease of use!

ML can do amazing things… But they could be even more amazing if **accessible to all!**

Setup

All you need is a **web browser!** Go to Azure ML website and choose:

- **Free** workspace: start using **all the features** of Studio immediately, **no credit card** required!

- **Enterprise** workspace: add extra storage and few additional web services features ($10/month).

Then, start working on your data from **anywhere!**

Build

Creating a predictive model with Azure ML is as easy as ...

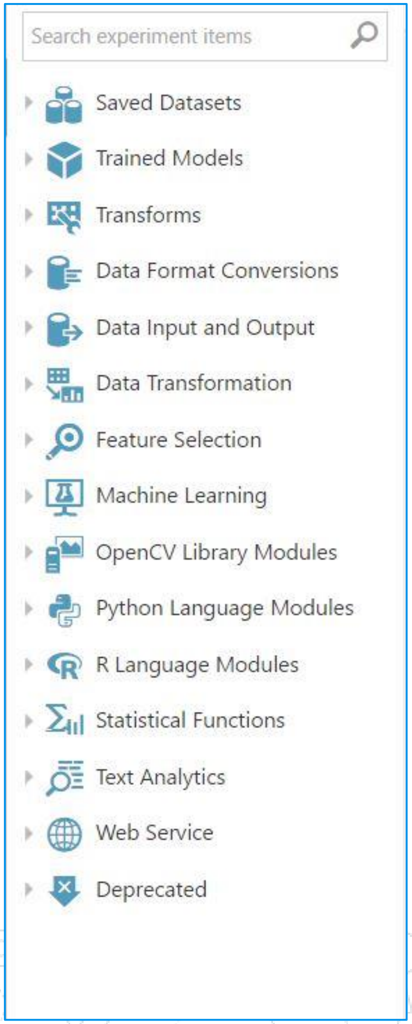
... playing with LEGO®!

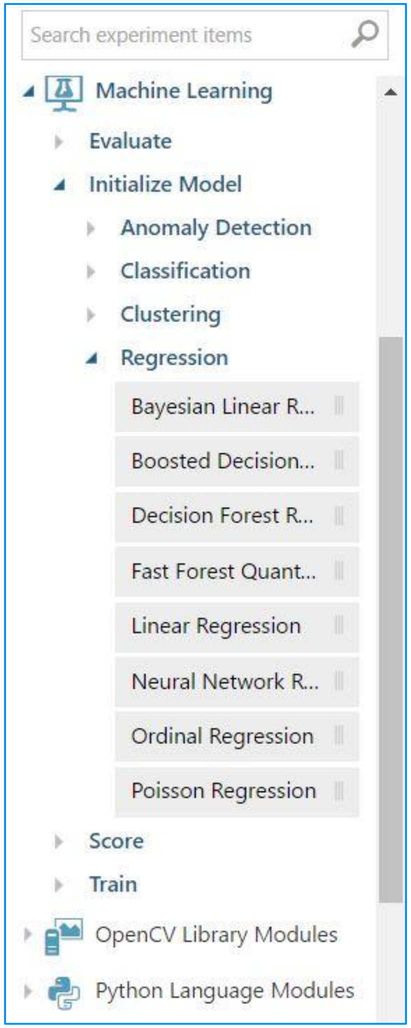
Build - main features

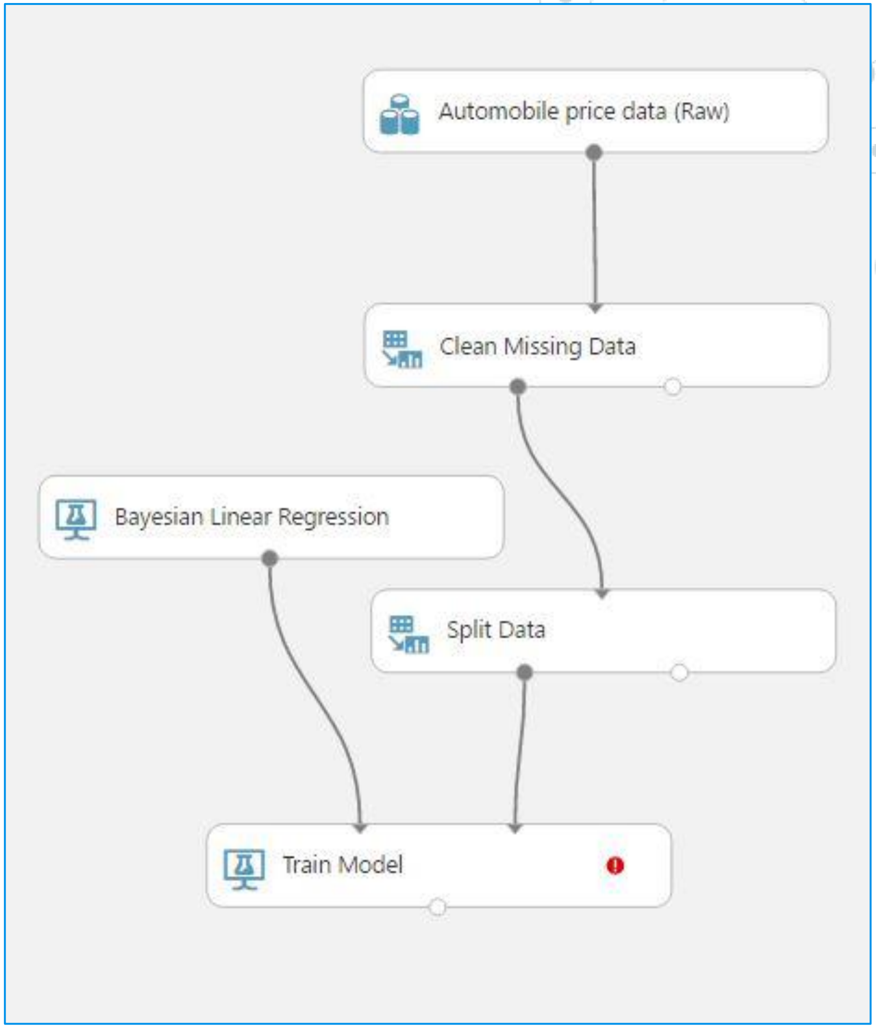
To help you building your **training experiment** (model) from scratch, Studio provides:

- Interactive, intuitive visual workspace.
- Drag-and-drop interaction to connect **modules** each other. For instance:
  - ready-to-use **datasets**.
  - ready-to-use standard **ML algorithms**.
  - your special sauce (cooked in **Python** or **R**).
  - ...
- Huge set of **samples** and **templates**.

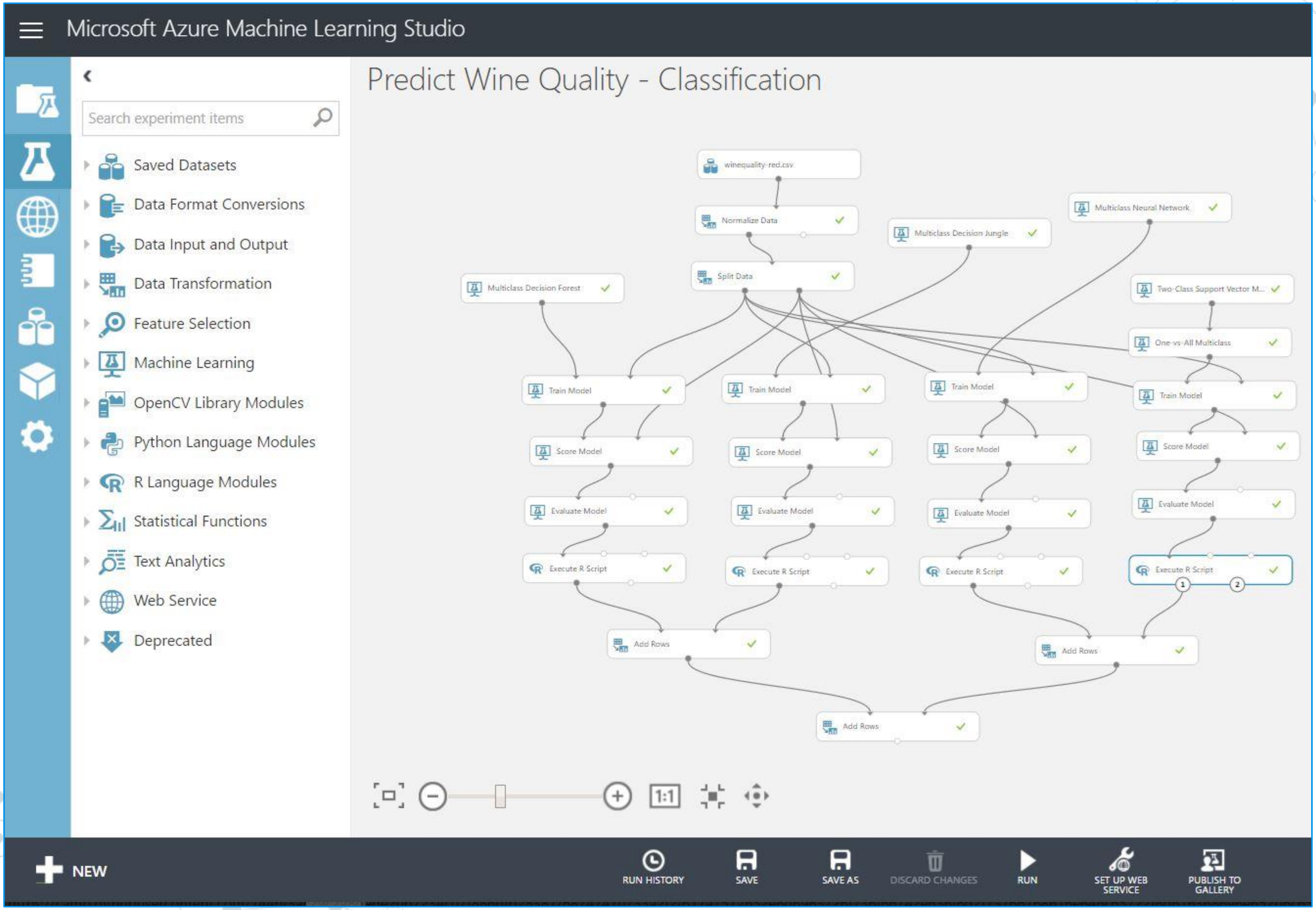
Build - example







Build - advanced example



Build - additional features

Besides creating experiments, Studio allows you to:

- upload your own **datasets**.
- create **web services**. (!!!)
- store and reuse your **trained models**.
- create Jupyter **notebooks**.
- save your account **settings**.
- collect all previous objects into a single **project**.

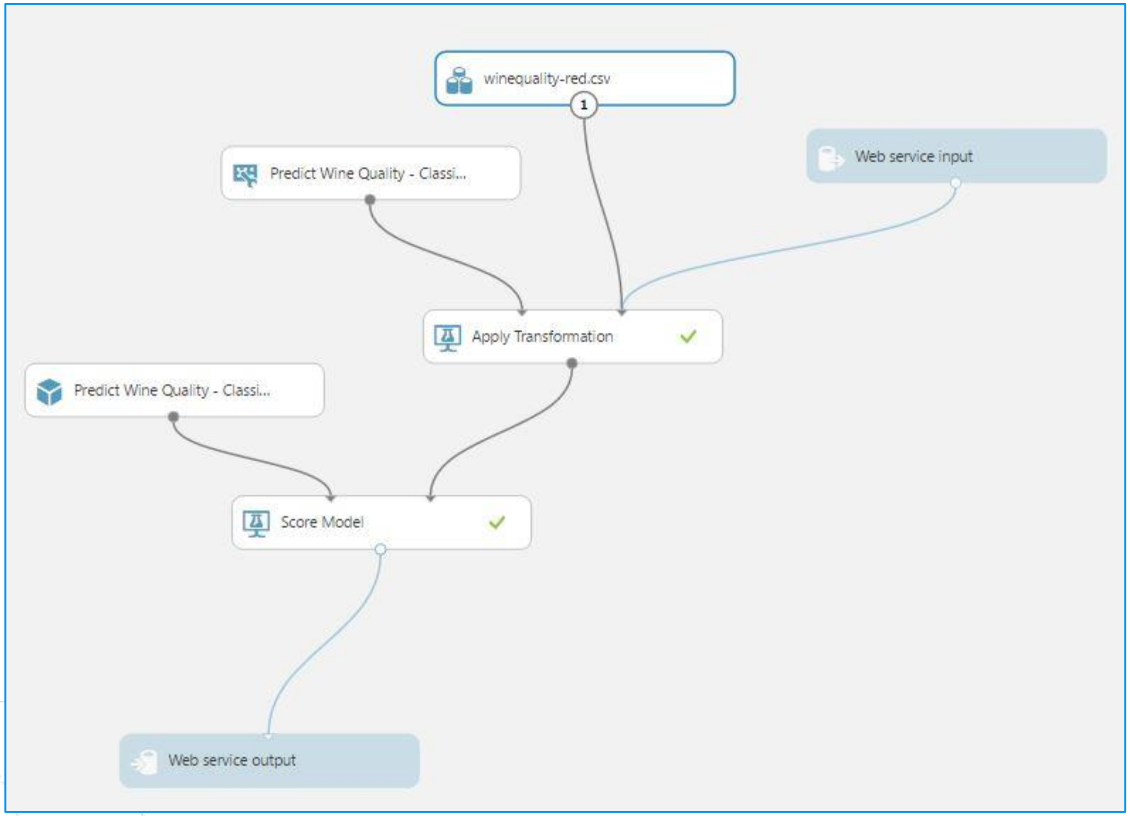
Deploy

Once your model is ready, deploy it as a **web service** in few steps:

- right from Studio, click on “Setup WS”.
- wait for your **predictive experiment** to be created.
- click on “Deploy WS”.
- wait for your **web service** to be deployed.
- **enjoy!**

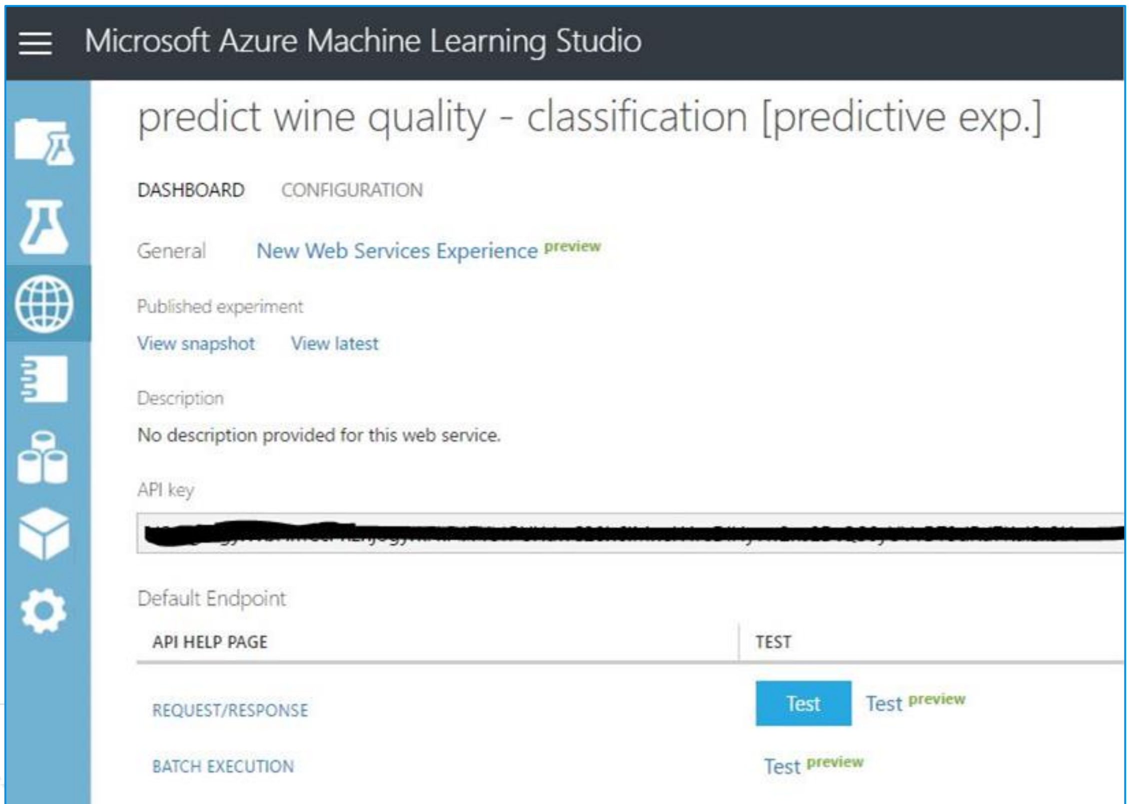
Deploy - predictive experiment

The original experiment is “translated” and the model is used to predict results.



Deploy - web service

To call your new web service, just follow the instructions about building the **POST request**.



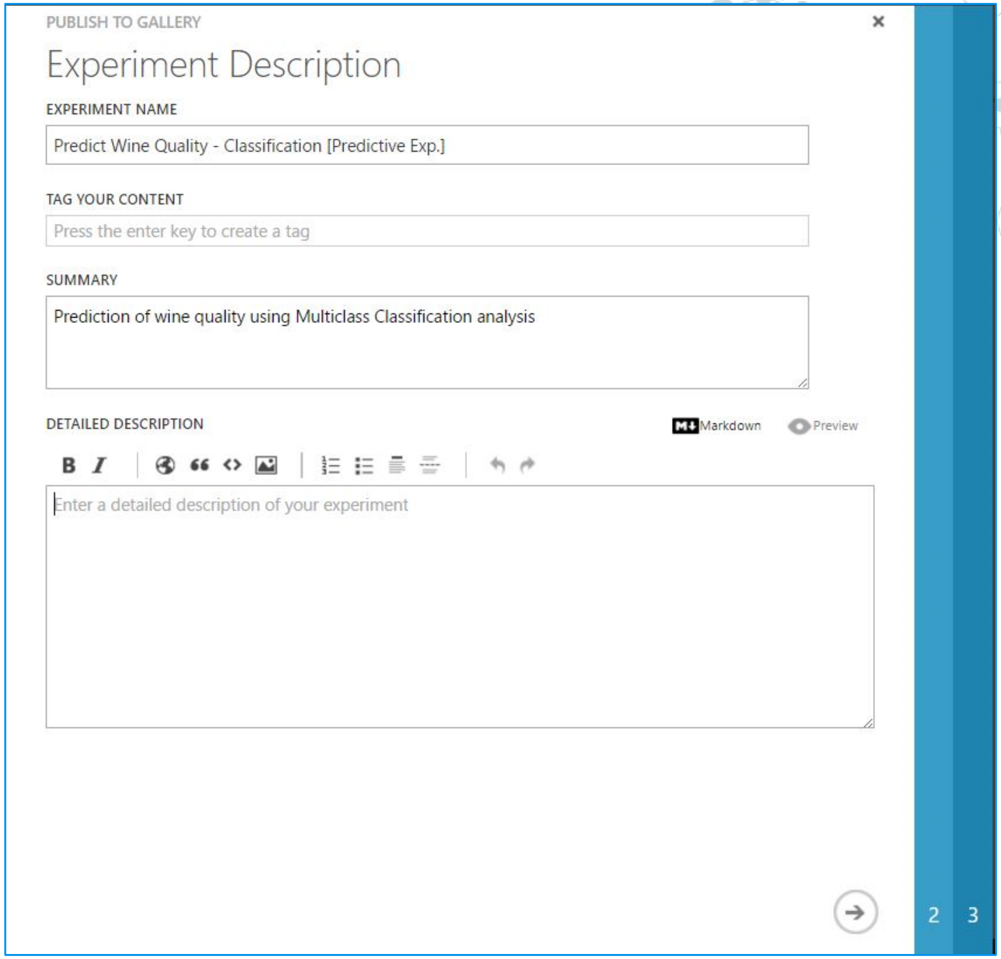
Share

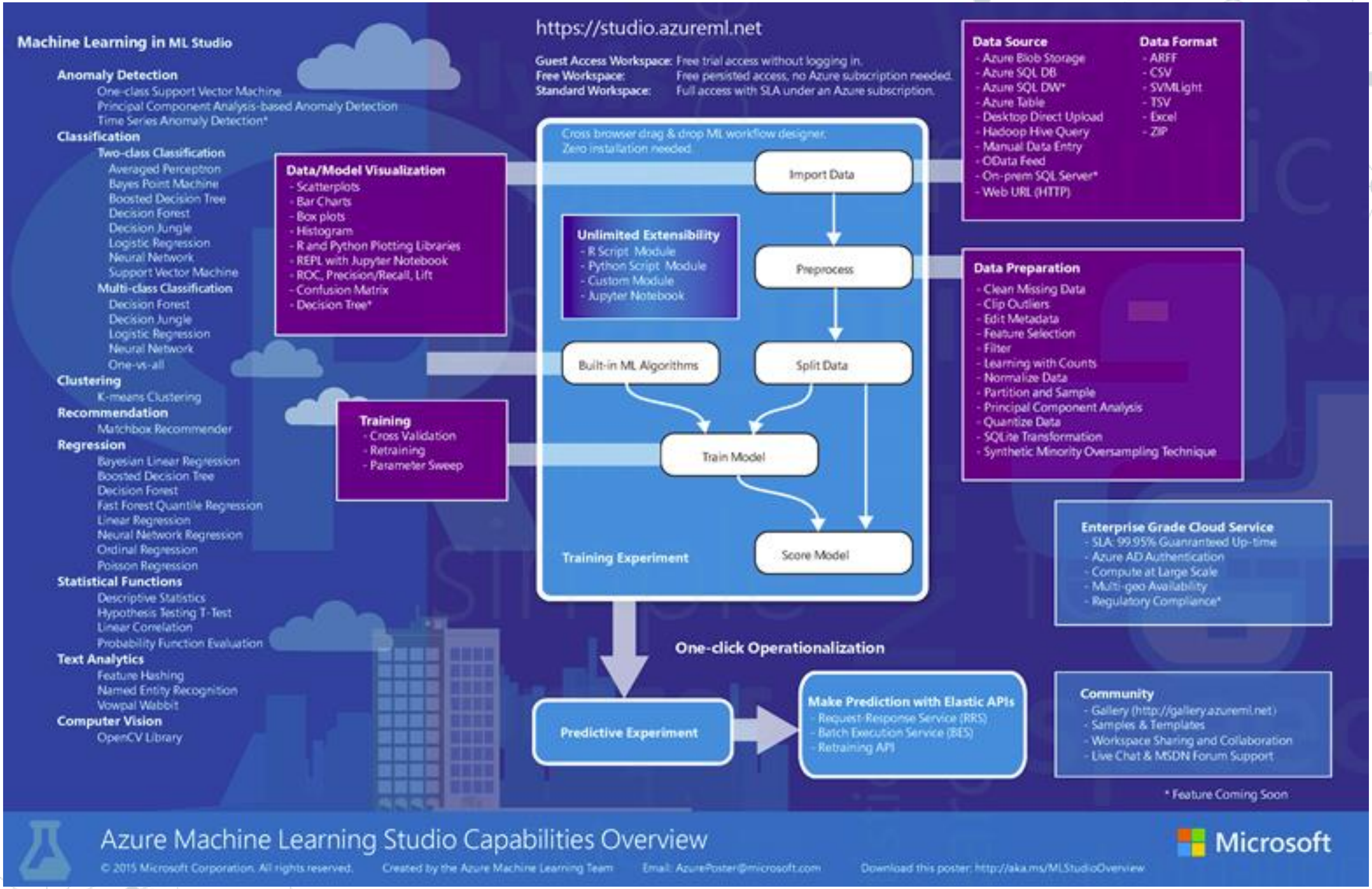
Your brand new experiment is ready to be shared in the community. Remember, **ML accessible for all!**

Upload it on **Cortana Intelligence Gallery**, where data scientists and developers share solutions.

Share - gallery

You can publish your work directly from the Studio. Just follow the instructions and describe what you have done!





https://docs.microsoft.com/en-us/azure/machine-learning/machine-learning-studio-overview-diagram

4.

Hands-on time!

A brief tutorial about creating and deploying an experiment.

Microsoft Azure Machine Learning Studio

- Go to Microsoft Azure Machine Learning Studio.
- In order to use the framework we need a Microsoft account:
  - A. I already have one of them → just "Sign in"
  - B. I do not have any of them → must "Sign Up"

Welcome to Azure Machine Learning

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.

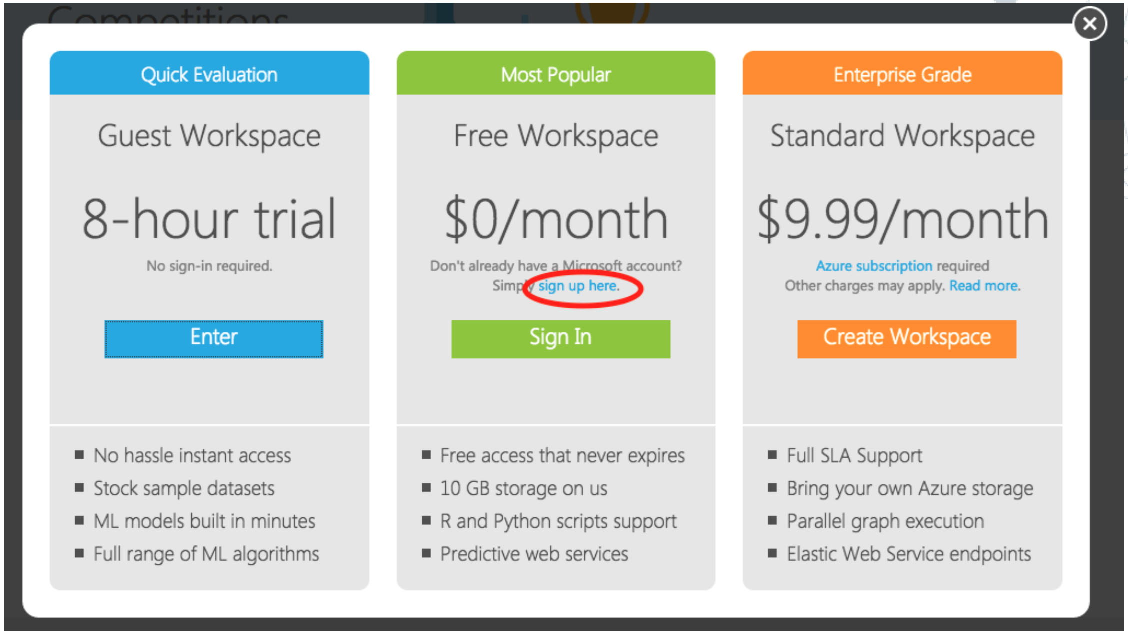
Sign Up

Already an Azure ML User?
Sign in here

Sign up

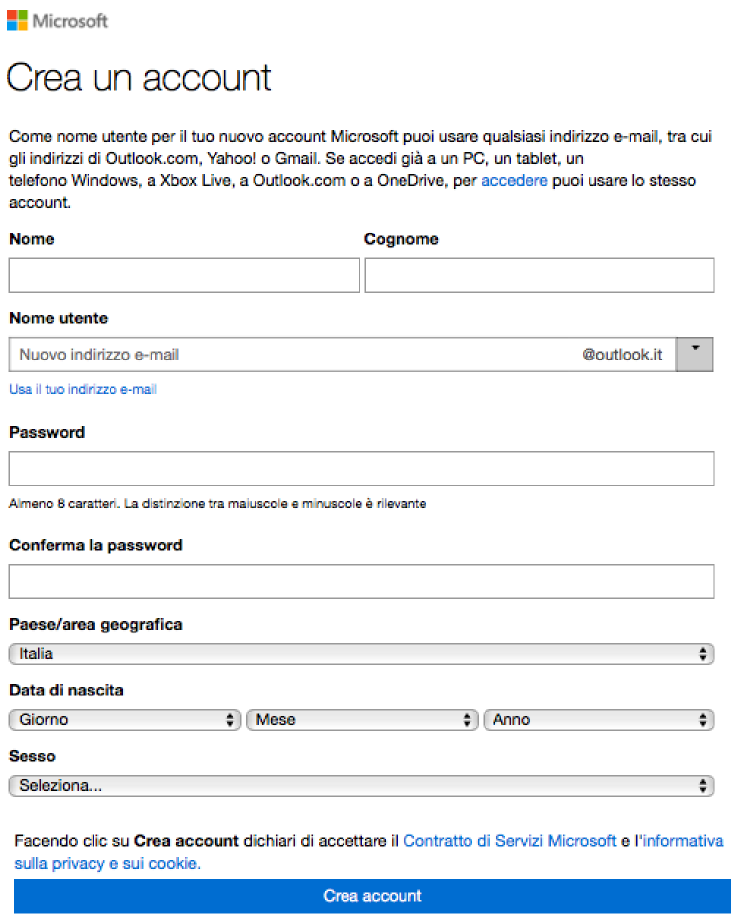
Select “Free Workspace”

- Free access
- 10GB Storage
- R and Python scripts support
- Predictive web services



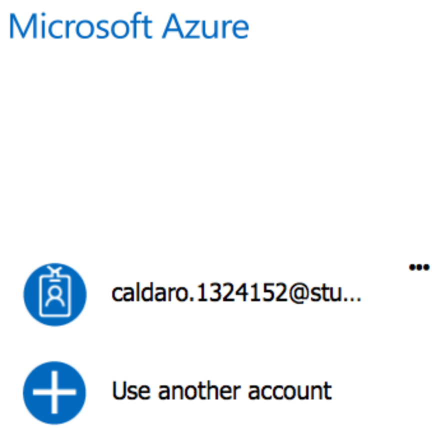
Create an account

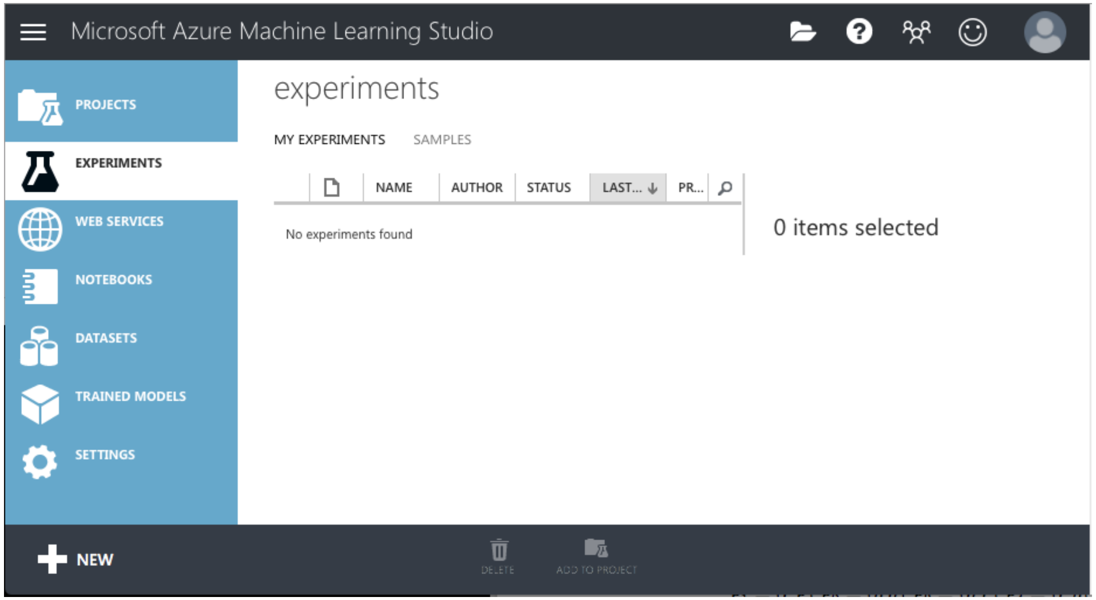
1. Fill the form
2. Click on create an account
3. Verify your email



Sign in

- Type the account you want to use and log in in the free workspace.





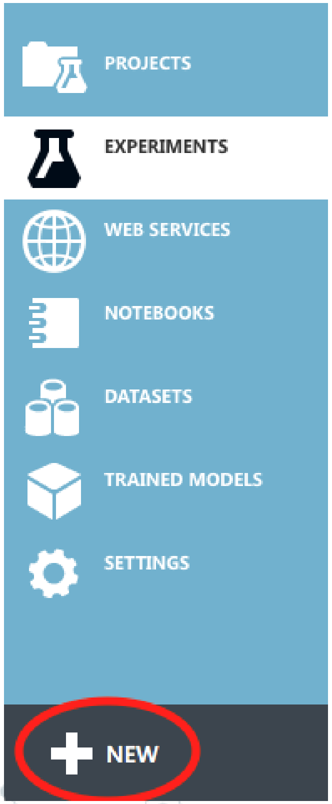
Five steps to create an experiment

- Create a model
  - Get data
  - Prepare the data
  - Define features
- Train the model
  - Choose and apply a learning algorithm
- Score and test the model
  - Predict new automobile prices

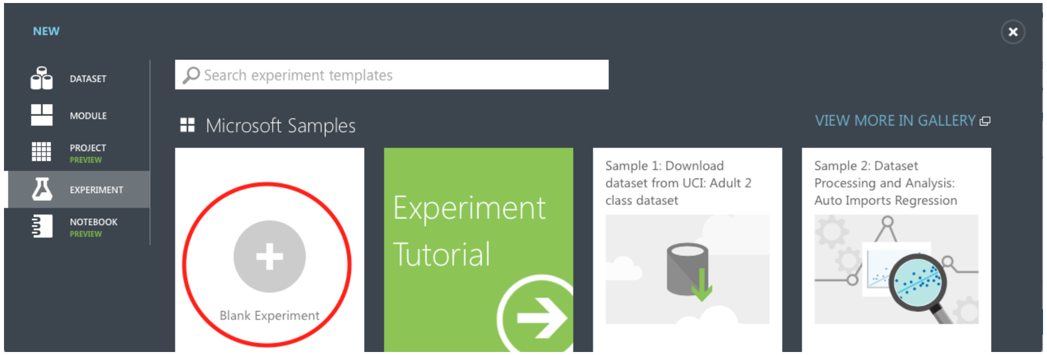
Automobile price prediction

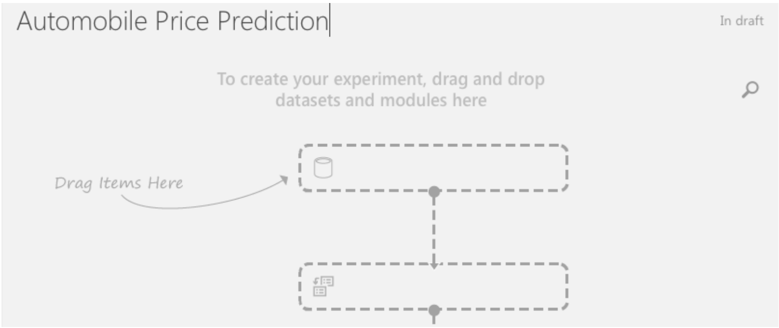
Technical Specifications → Market Price

Create a blank experiment



→





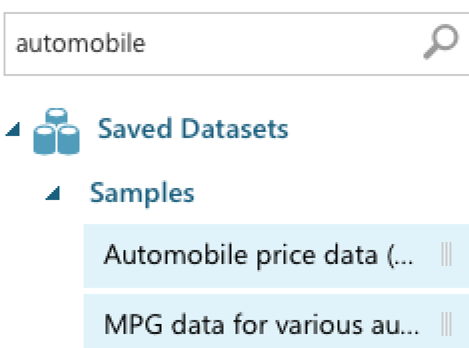
1. Get Data

- Use data in the existing sample datasets
- Create your own dataset by NEW dataset
- Import data: Load data from sources such as the Web, Azure SQL database, Azure table, Hive table, or Windows Azure BLOB storage. Formerly known as Reader

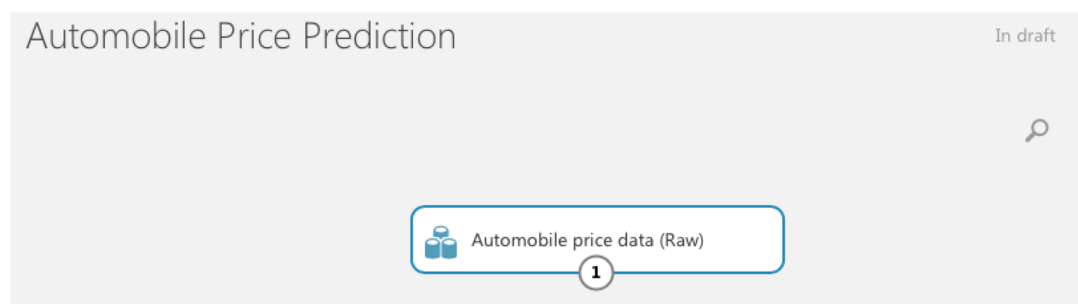
Using Azure saved dataset

- In the search bar, look for automobile

- Drag and drop the dataset in the dashboard

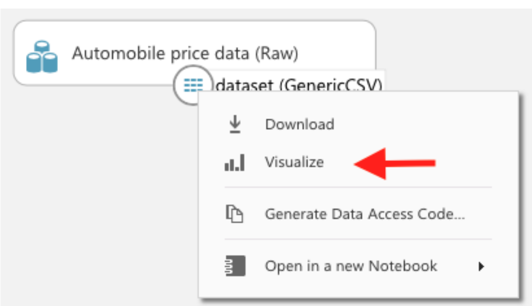


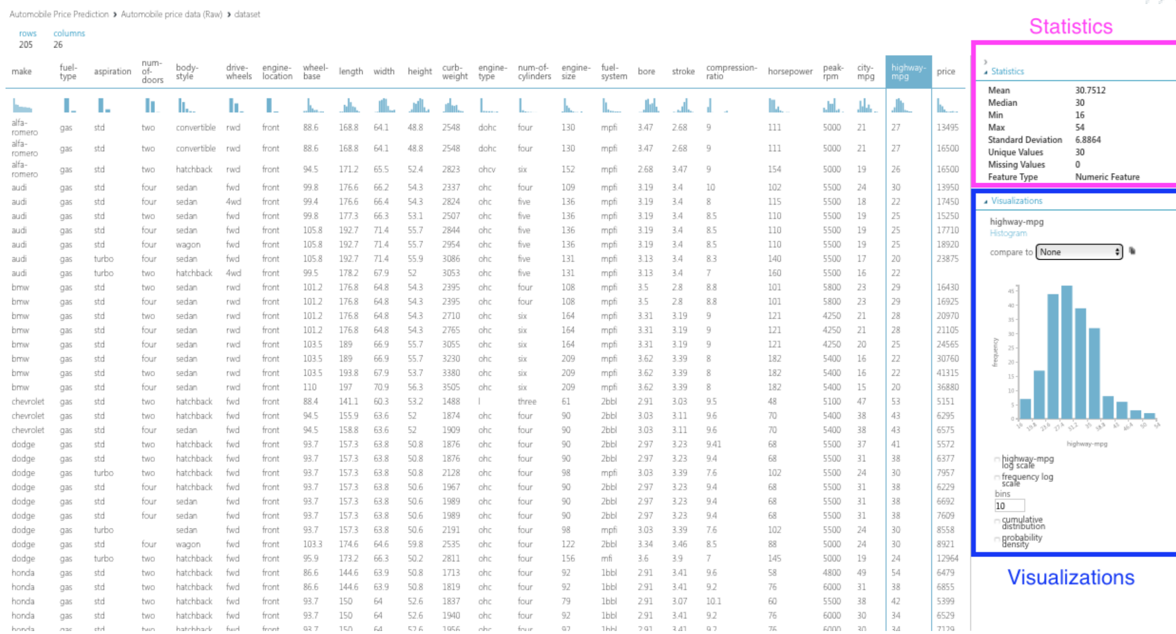
→



Visualize the Data

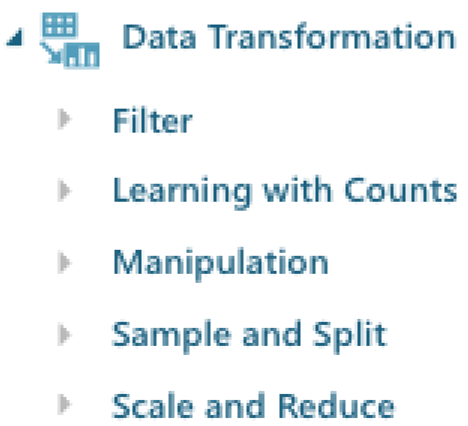
- Selecting one column, some statistics are shown
- Given the variables for a specific automobile, we're going to try to predict the price (last column)





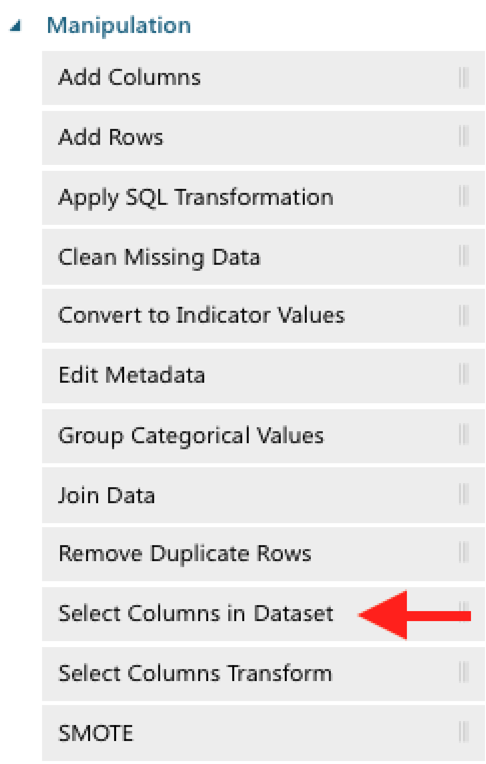
2. Prepare the data

- This menu can be used to transform raw data to the input of the next modules

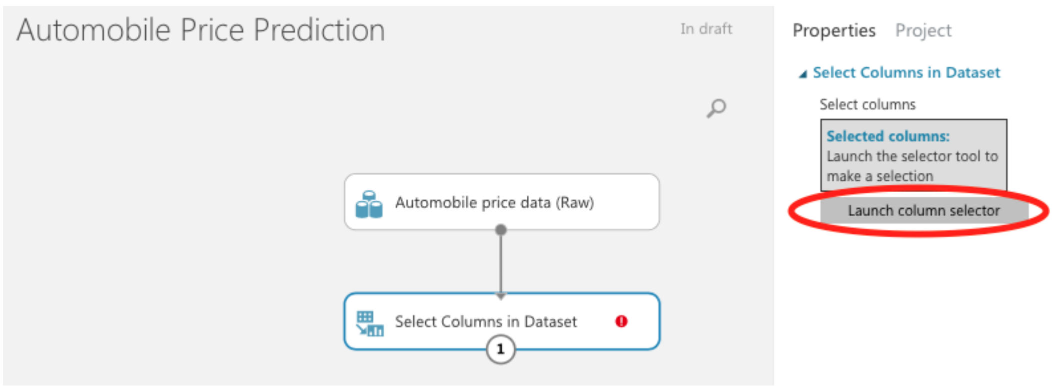


Preprocess automobile dataset

1. Clean missing values present in the columns of various rows so the model can analyze the data correctly.
2. Do not consider some columns.

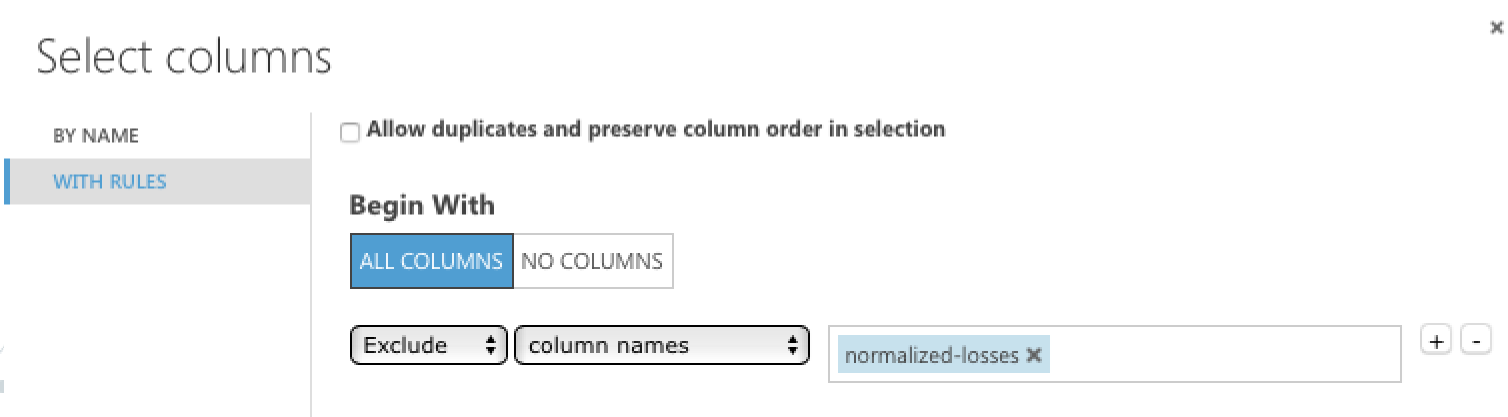


→

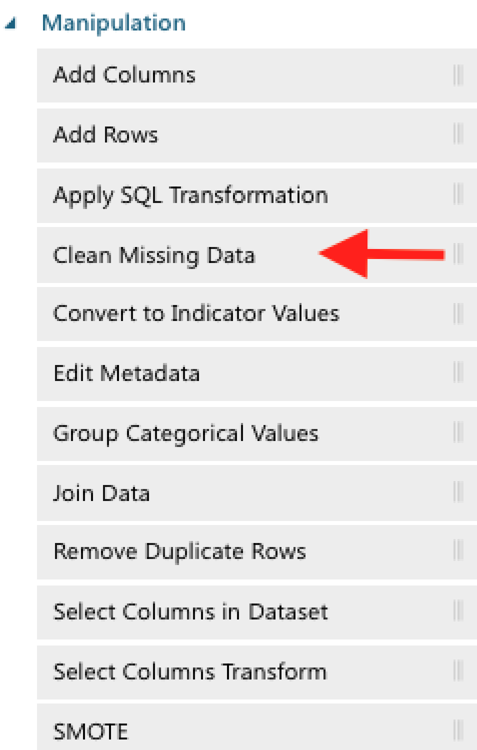


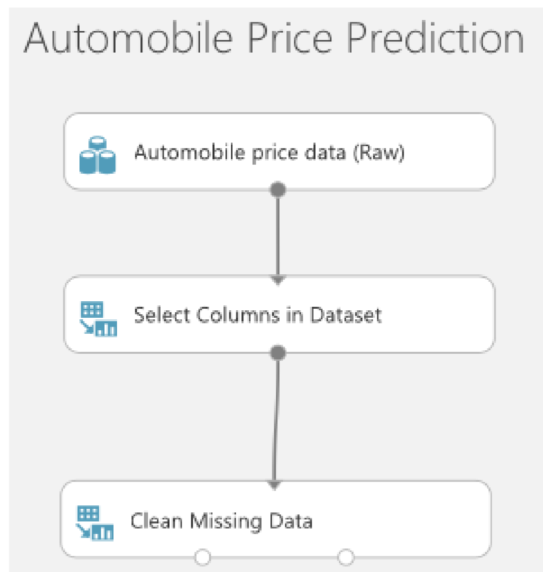
Clean missing data: remove column

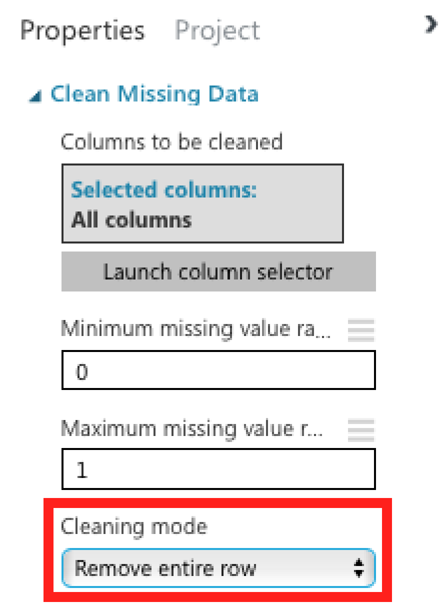
- Click on **Launch column selector**
- On the left, click **With rules**
- Under **Begin With**, click **All columns**.
- Select **Exclude** and **column names**,
- Click inside the text box and select *normalized-losses*



Clean missing data: remove row

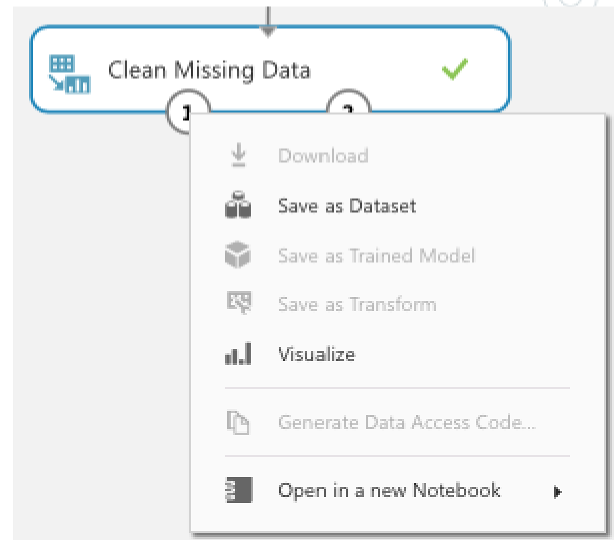






Run the experiment and visualize processed data

- Save the experiment
- Run it
- Visualize data output from Clean Missing Data
- Check differences

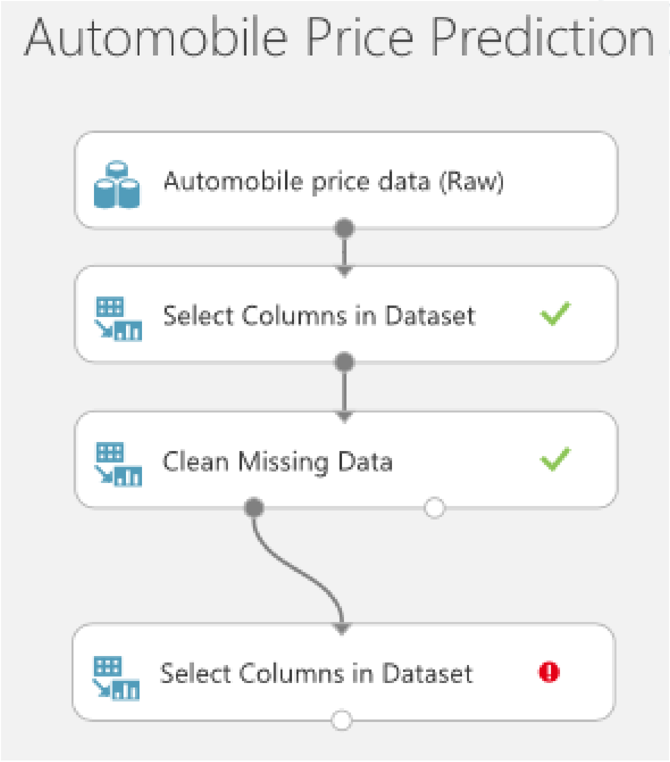


3. Define features

- **Features**: individual measurable properties of something you’re interested in.
- Finding a good set of features for creating a predictive model requires experimentation and knowledge about the problem you want to solve.
- (In our example each row represents one automobile, and each column is a feature of that automobile)

Feature selection

- As before, drag *Select columns in Dataset*
- Connect *Clean Missing Data* to the module just added
- Click on **Launch column selector**
- On the left, click **With rules**
- Under **Begin With**, click **No columns**.
- Select **Include** and **column names**,
- Click inside the text box and select *“make”, “body-style”, “wheel-base”, “engine-size”, “horsepower”, “peak-rpm”, “highway-mpg”, “price”*

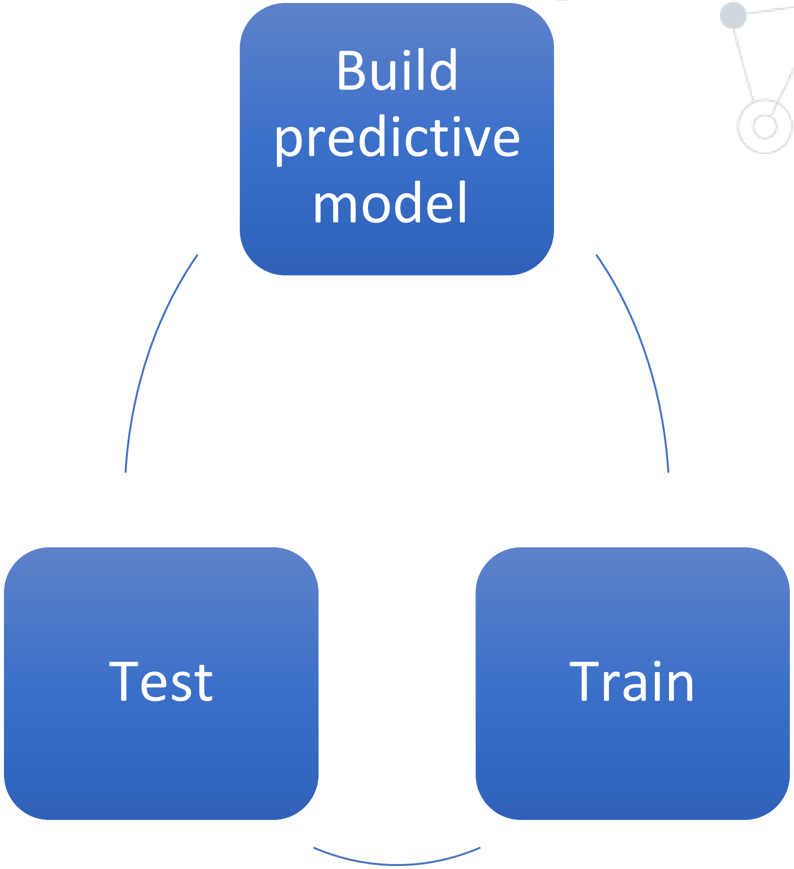


4. Choose and apply a learning algorithm

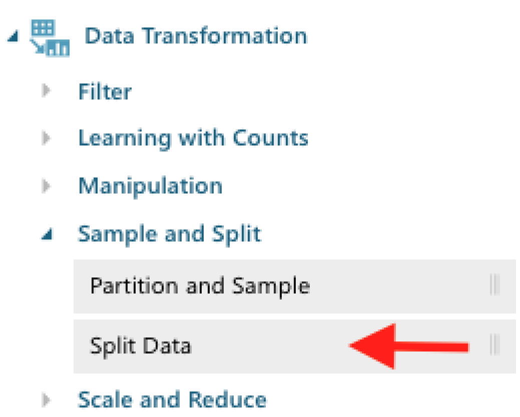
- Classification: predicts an answer from a defined set of categories

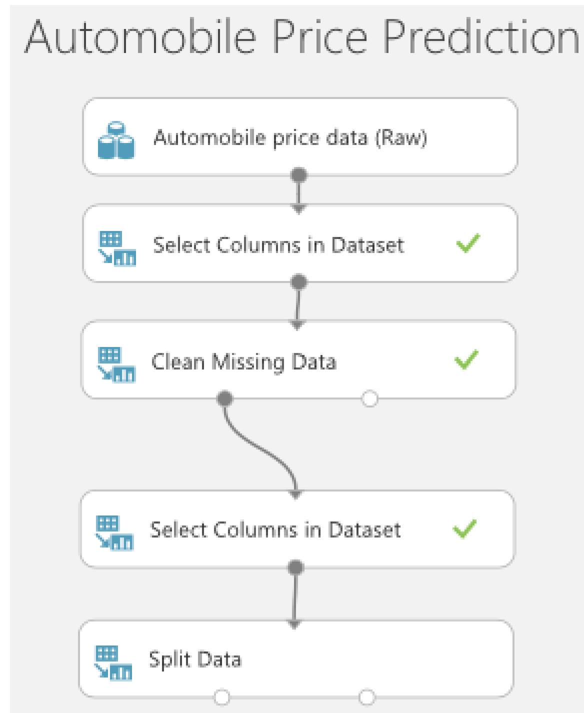
- Regression: predicts a number.

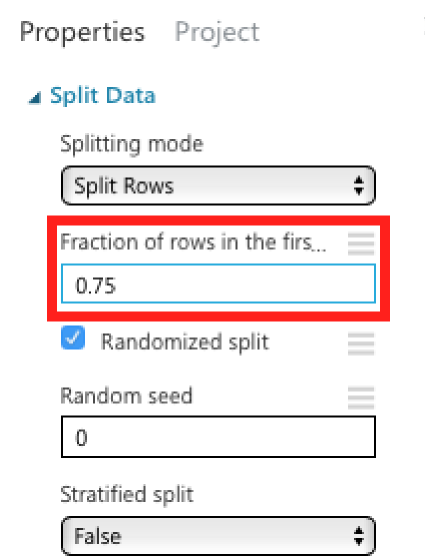
- (Because we want to predict price, which is a number, we'll use a regression algorithm)



Split data into train set and test set

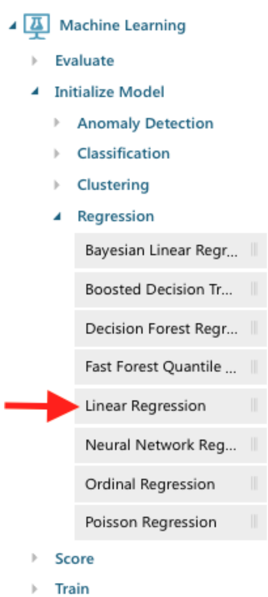




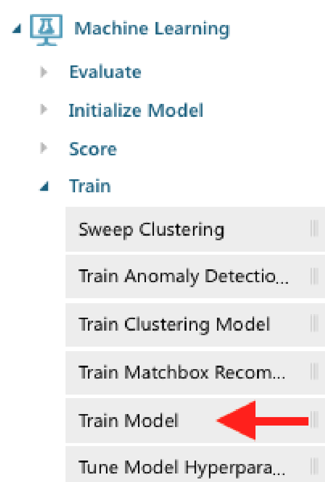


Learning algorithm selection

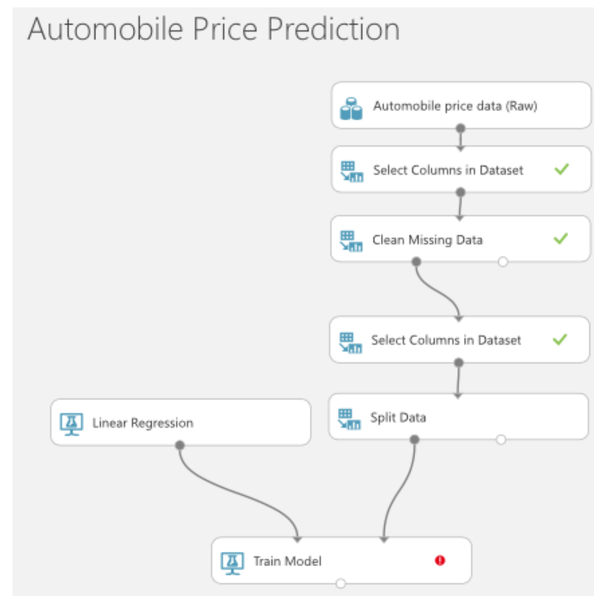
- Connect the *"Train Model"* module to both the *"Linear Regression"* and *"Split Data"* modules



+

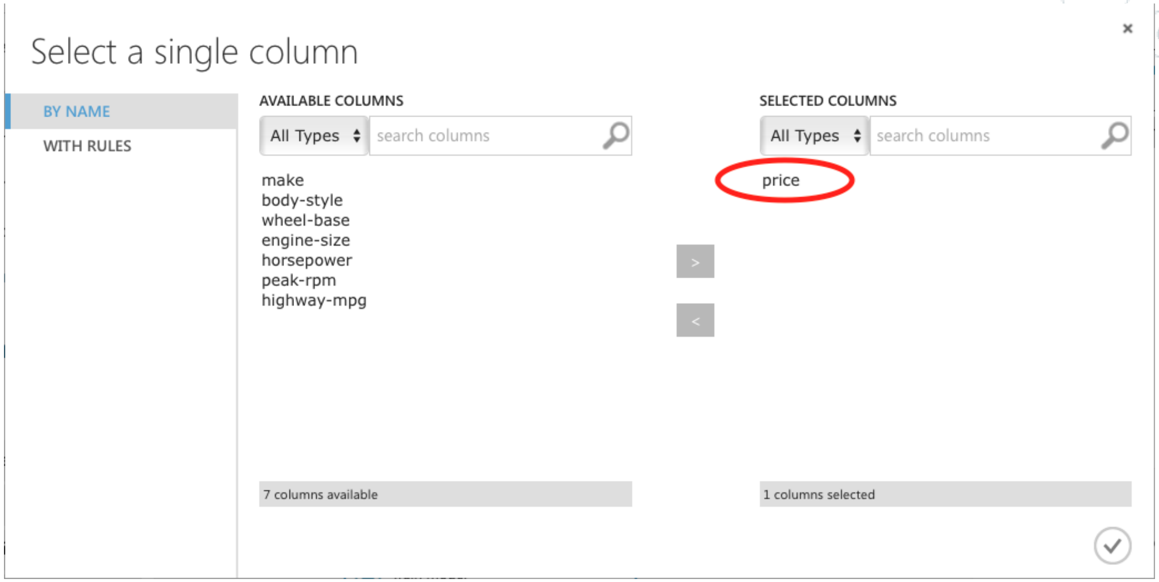


=



Train a specific feature

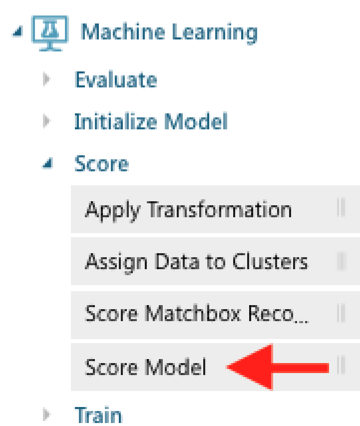
- Click the *Train Model* module
- Click *Launch column* selector in the Properties pane
- Click *By Name*
- Select the **price** column.

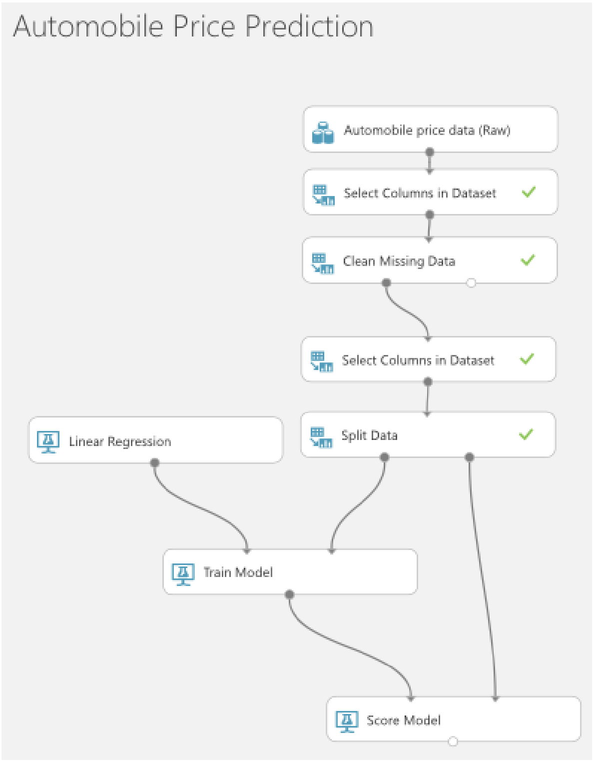


- This is the value that our model is going to predict.

5. Predict new automobile prices

- 75 percent of our data used to train the model using
- 25 percent of the data to score the model functions.





Output of the score module

- Predicted values for price and its

Automobile Price Prediction › Score Model › Scored dataset

rows	columns
48	9

Real Price Predicted Price

make	body-style	wheel-base	engine-size	horsepower	peak-rpm	highway-mpg	price	Scored Labels
subaru	sedan	97	108	111	4800	29	11259	10286.204819
mitsubishi	hatchback	93.7	92	68	5500	38	6669	5446.847864
dodge	hatchback	93.7	90	68	5500	38	6229	6344.800711
honda	hatchback	86.6	92	76	6000	38	6855	5528.302953
alfa-romero	convertible	88.6	130	111	5000	27	16500	13498.476233
volvo	wagon	104.3	141	114	5400	28	16515	16097.608038
isuzu	hatchback	96	119	90	5000	29	11048	8315.257218
dodge	hatchback	93.7	90	68	5500	41	5572	6630.154608
bmw	sedan	101.2	108	101	5800	29	16430	19913.408695
mitsubishi	hatchback	93.7	92	68	5500	41	5389	5732.201761
bmw	sedan	103.5	209	182	5400	22	41315	30548.819502
jaguar	sedan	113	258	176	4750	19	35550	30863.486076
plymouth	hatchback	93.7	90	68	5500	38	6229	5806.676601
toyota	hatchback	102.9	171	161	5200	24	16558	17388.014192
mitsubishi	hatchback	95.9	156	145	5000	24	14489	13094.447938
plymouth	hatchback	93.7	90	68	5500	41	5572	6092.030497
volkswagen	sedan	97.3	97	52	4800	46	7995	8344.693482
dodge	hatchback	93.7	98	102	5500	30	7957	8258.383335
mercedes-benz	sedan	115.6	234	155	4750	18	34184	34960.643871
alfa-romero	hatchback	94.5	152	154	5000	26	16500	14329.816126

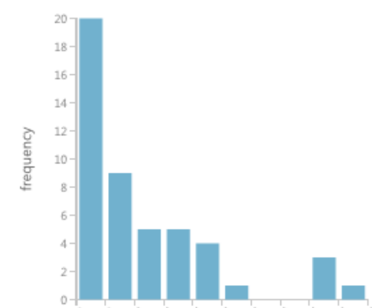
Statistics

Mean	12437.776
Median	10208.7085
Min	5446.8479
Max	34960.6439
Standard Deviation	7323.458
Unique Values	46
Missing Values	0
Feature Type	Numeric Score

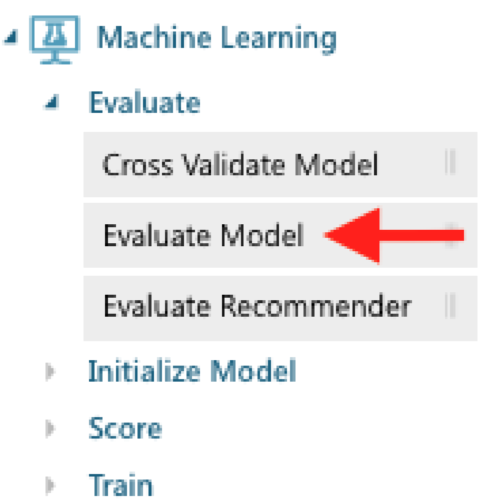
Visualizations

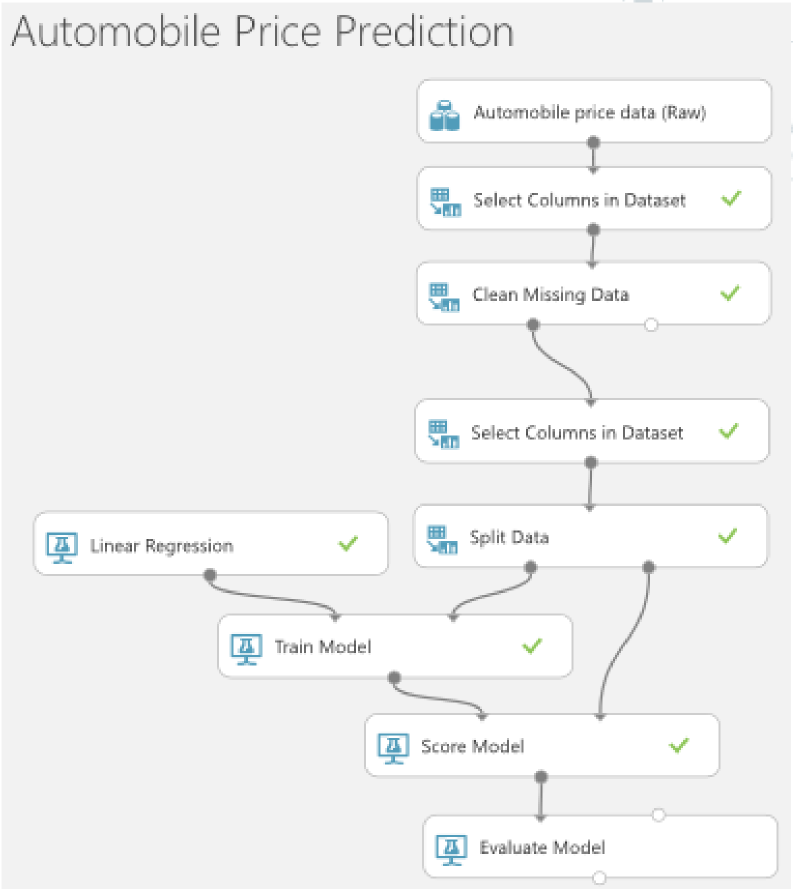
Scored Labels
Histogram

compare to None



Results evaluation





(Final Experiment)

Metrics

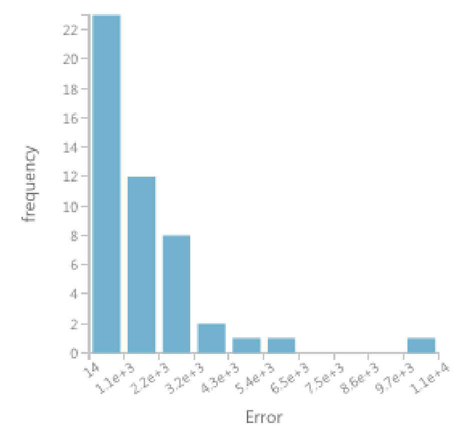
- *Mean Absolute Error* (**MAE**): The average of absolute errors (an error is the difference between the predicted value and the actual value).
- *Root Mean Squared Error* (**RMSE**): The square root of the average of squared errors of predictions made on the test dataset.
- *Relative Absolute Error*: The average of absolute errors relative to the absolute difference between actual values and the average of all actual values.
- *Relative Squared Error*: The average of squared errors relative to the squared difference between the actual values and the average of all actual values.
- *Coefficient of Determination*: Also known as the R squared value, this is a statistical metric indicating how well a model fits the data.

Automobile Price Prediction › Evaluate Model › Evaluation results

Metrics

Mean Absolute Error	1656.147651
Root Mean Squared Error	2456.983209
Relative Absolute Error	0.276606
Relative Squared Error	0.089608
Coefficient of Determination	0.910392

Error Histogram



How a metric should be

- *For each of the error statistics, smaller is better.*
- *A smaller value indicates that the predictions more closely match the actual values.*
- *For Coefficient of Determination, the closer its value is to one (1.0), the better the predictions.*

Iterate to improve the model

- *Change the features you use in your prediction*
- *Modify the properties of the Linear Regression algorithm*
- *Try a different algorithm altogether*
- *Add multiple machine learning algorithms to your experiment at one time*
- *Compare two of them by using the Evaluate Model module*

6. Deploy an Azure Machine Learning web service

- *Satisfied with your model???*
- *You can deploy it as a web service!*
- *Use the WebService to predict automobile prices by using new data…*

Create a training experiment

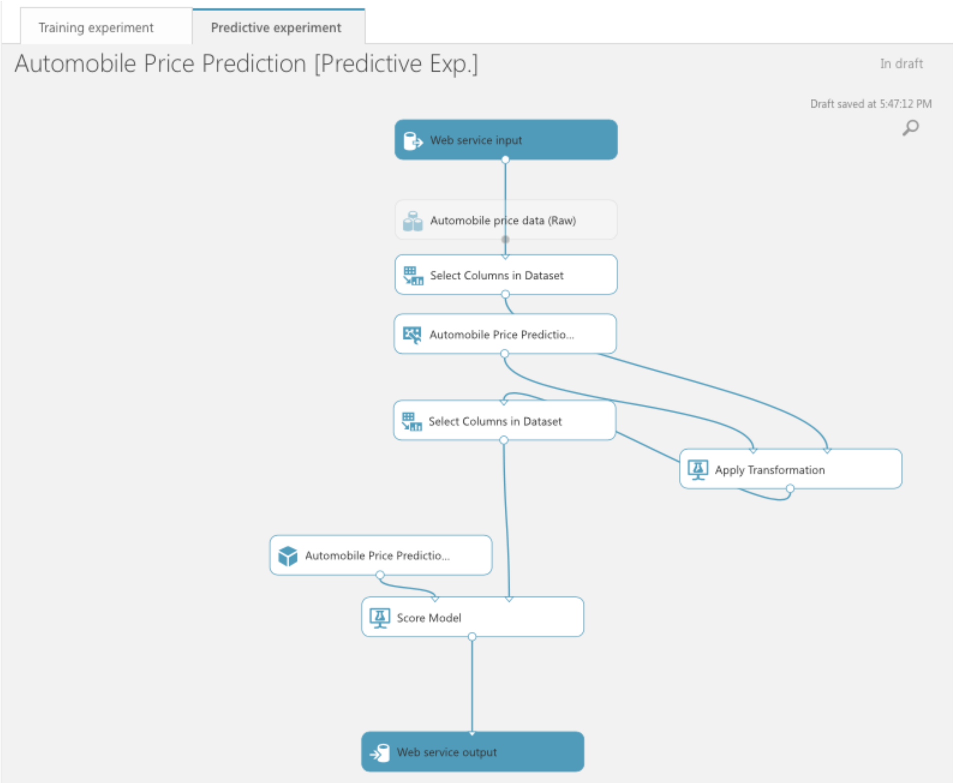
Convert the training experiment to a predictive experiment

Deploy the predictive experiment as a New web service

Convert the training experiment to a predictive experiment

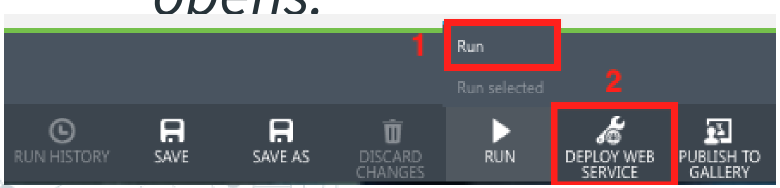
- *By converting to a predictive experiment, you're getting your trained model ready to be deployed as a scoring web service.*
- *Users of the web service can send input data to your model and your model will send back the prediction results.*
- *As you convert to a predictive experiment, keep in mind how you expect your model to be used by others.*

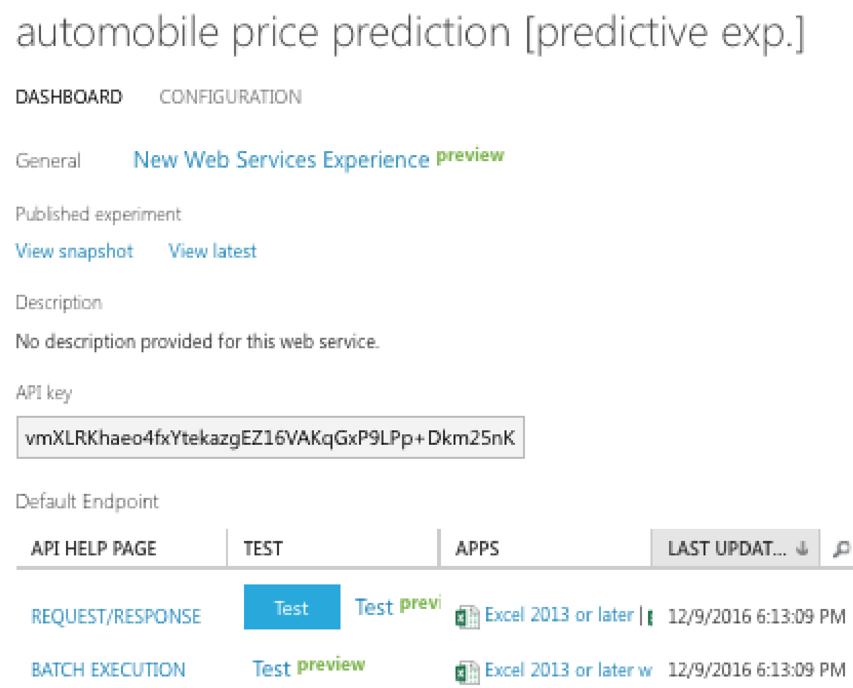
Predictive experiment



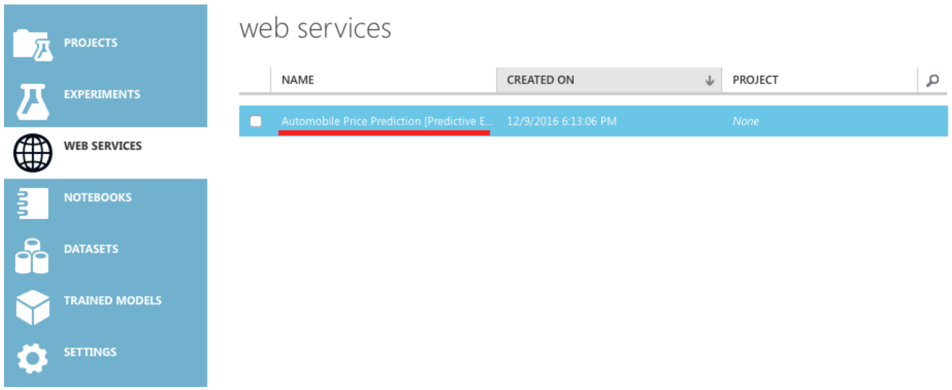
Deploy the predictive experiment as a New web service

- *Click Run*
- *Click Deploy Web Service*
- *Select Deploy Web Service New.*
- *The deployment page of the Machine Learning Web Service portal opens.*

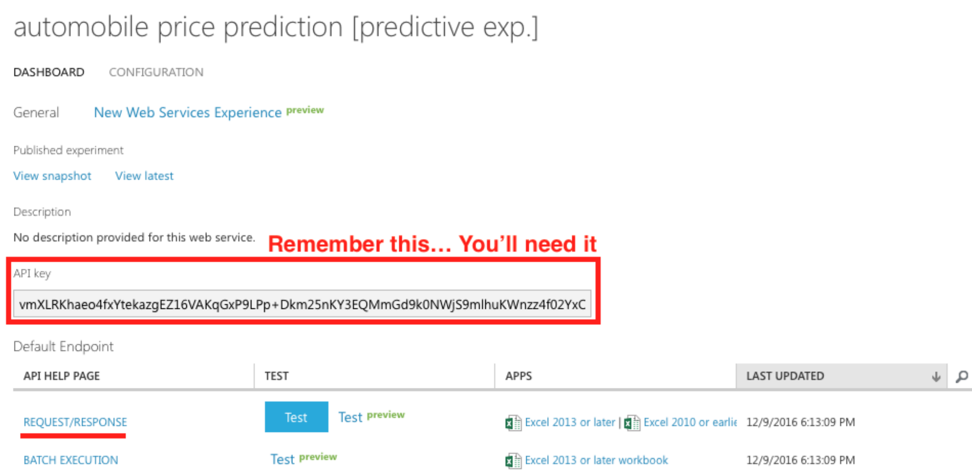




Test your Web Service with a Python Program



- *request/response page contains Request Response API Documentation, with a starter Python program (that must be modified) to call the web service*



Available material

https://github.com/giacomolanciano/Azure-Machine-Learning-tutorial

http://www.slideshare.net/GiacomoLanciano/azure-machine-learning-tutorial

Thanks!

Any questions?