Topic 2
Applying Machine Learning (ML) using Cloud tools

ST0249 (AIML) AI & MACHINE LEARNING

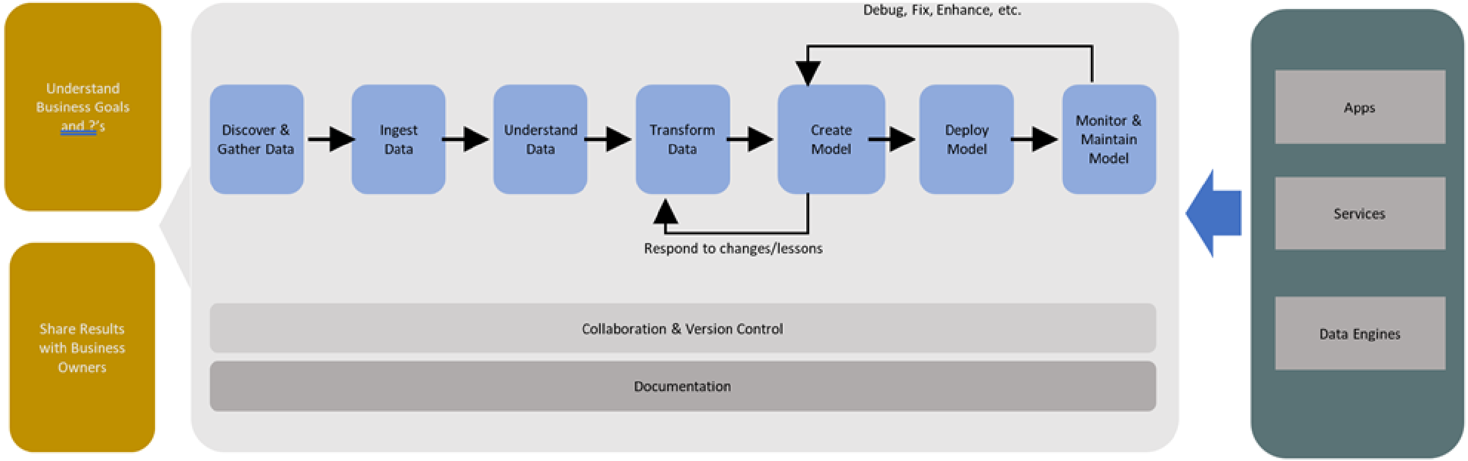
Learning Outcomes

- Use of Machine Learning Tools
  - Describe some online/cloud based machine learning tools
  - Configure and setup Microsoft Azure Machine Learning and Microsoft Azure Notebooks
- Understand principles of data representation and feature engineering
  - Understanding representation of categorical variables
  - Explain binning, discretization
  - Understand feature selection
  - Understand how to utilize expert knowledge

Use of Machine Learning Tools

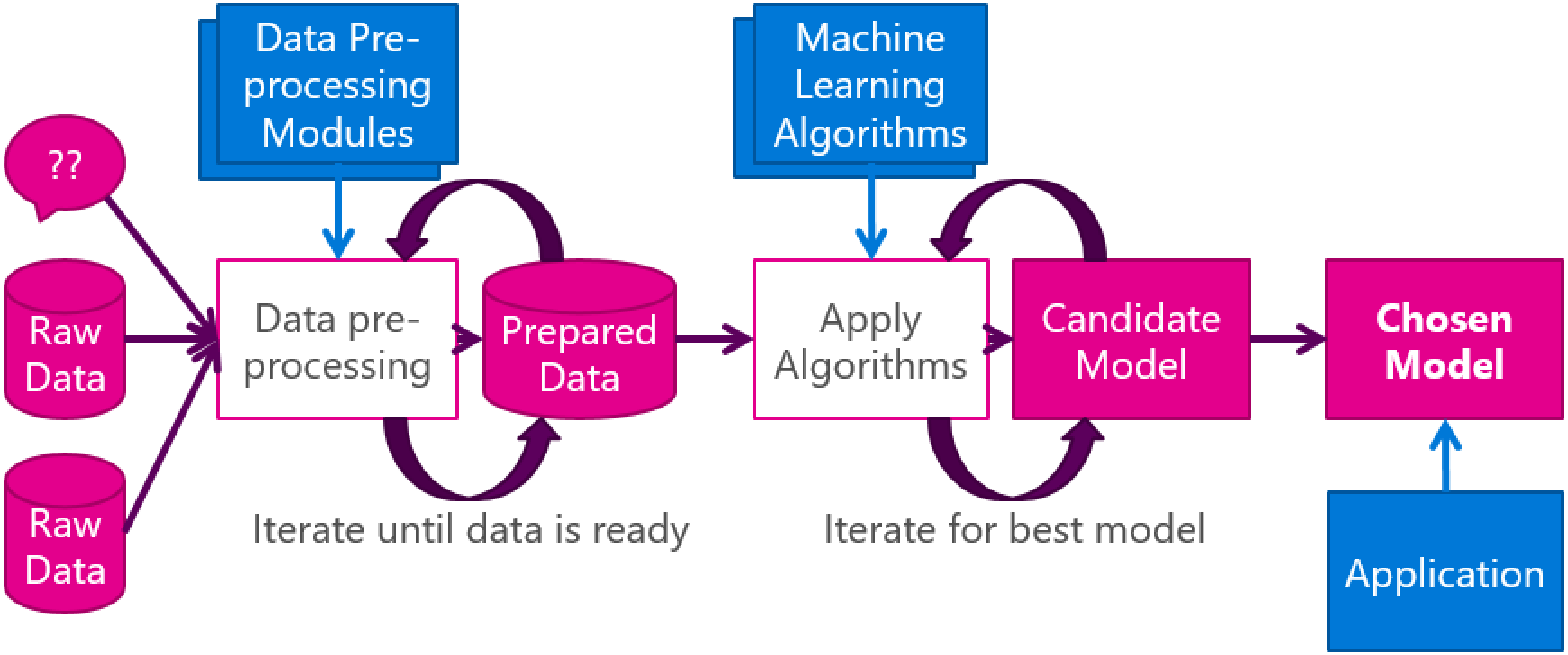
Azure Machine Learning

- ❑ Cloud based tool that covers the entire workflow from data source to deployment of machine learning algorithm as a web service
- ❑ Cloud based tool to perform all the steps needed to prepare the data, explore the data, create models, score and evaluate the models


Understand Business Goals and ?'s
Share Results with Business Owners
Debug, Fix, Enhance, etc.
Discover & Gather Data
Ingest Data
Understand Data
Transform Data
Create Model
Deploy Model
Monitor & Maintain Model
Respond to changes/lessons
Collaboration & Version Control
Documentation
Apps
Services
Data Engines

source https://azure.microsoft.com/en-us/blog/diving-deep-into-what-s-new-with-azure-machine-learning/

Machine Learning Process



source https://blogs.msdn.microsoft.com/martinkearn/2016/03/01/machine-learning-is-for-muggles-too/

Machine Learning Process

- **The primary goal of the process is to identify a 'Model'**. The Model is the main thing that applications can submit requests to in order to gain insight on new data. A person working as the role of a Data Scientist performs the Machine Learning process and will ultimately decide on the right model to use.
- **The process starts with a question**; what are you trying to learn from your Machine Learning experiment? For example, in the case of recommendations, the question might be *"identify most commonly sold products for each product in the inventory"*

Machine Learning Process

- **The next step is to provide 'prepared data'**. Prepared data is one or more data sets that have been pre-processed (formatted, cleaned and sampled) in readiness to apply Machine Learning algorithms to. Preparing the data means that the data is in the best shape to draw scientific conclusions from and is not skewed in any way.
- Once you have your prepared data, you apply one or more Machine Learning algorithms to it with a view to producing a **Model**. This is an iterative process and you may loop around testing various algorithms until you have a Model that sufficiently answers your question.
- Once you have produced your chosen model, it will typically be exposed via some kind of API.

Azure Machine Learning

Azure Machine Learning service is one of the main platforms for doing Machine Learning in a quick, easy, cloud-based way.

The service contains a set of tools and modules that help the data scientist setup and run the Machine Learning process. It is designed for applied machine learning meaning and is designed to be used by real world applications and developers.

Azure Machine Learning

The Azure machine learning service offers 4 main components

- **ML Studio**: A web-based graphical user interface used to design experiments in a simple drag and drop style. Think of it as a web-based IDE for Data Scientists
- **Data pre-processing modules**: Azure offers a set of data pre-processing modules which can help clean, format and sample the raw data in order to get to the 'prepared data' stage of the process
- **Machine Learning Algorithms**: Azure offers a set of well-known and understood Machine Learning algorithms which can simply be imported into your experiment and applied to your prepared data to produce a model
- **REST API**: Once your chosen model is established, Azure can package it up as a published REST API which client applications can easily call in any language or platform

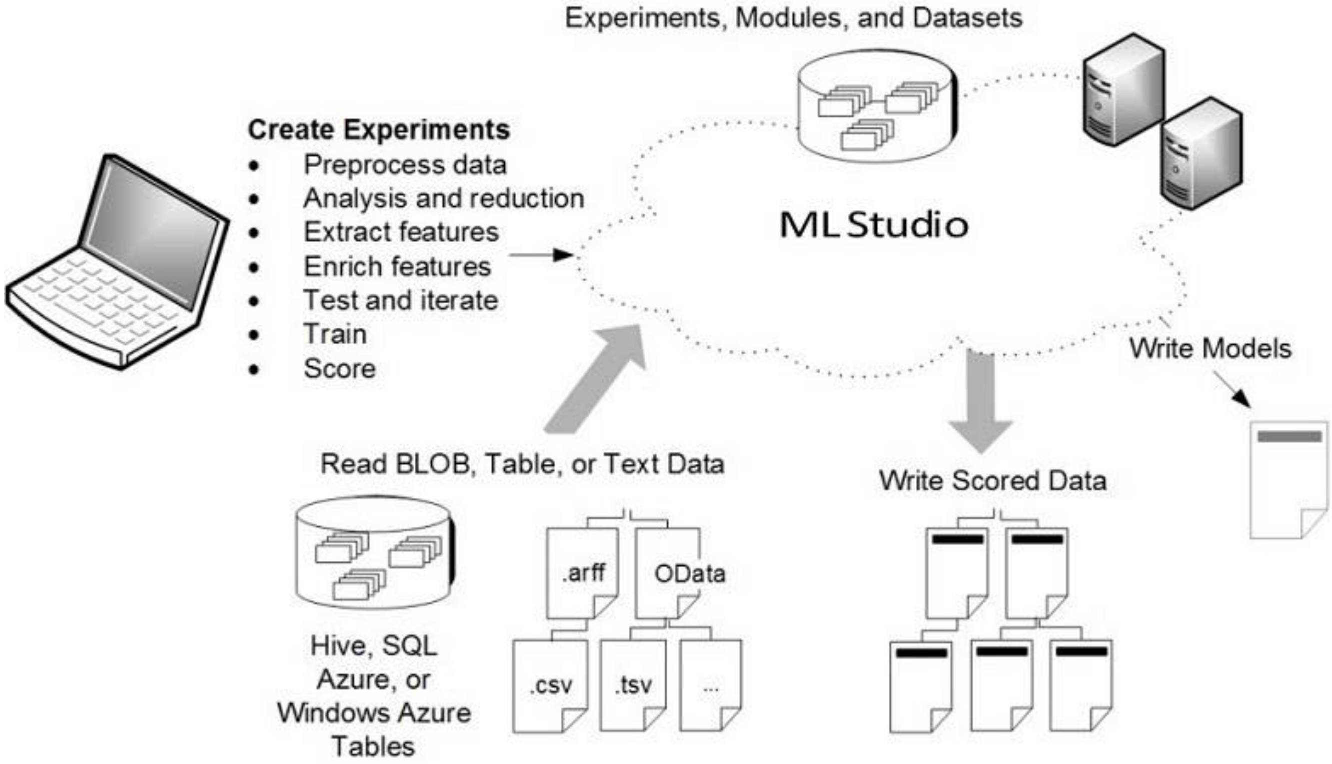
What is Azure Machine Learning (ML) Studio?

To develop a predictive analysis model, you typically use data from one or more sources, transform and analyse that data through various data manipulation and statistical functions, and generate a set of results. Developing a model like this is an iterative process. As you modify the various functions and their parameters, your results converge until you are satisfied that you have a trained, effective model.

Azure Machine Learning Studio gives you an interactive, visual workspace to easily build, test, and iterate on a predictive analysis model. You drag-and-drop ***datasets*** and analysis ***modules*** onto an interactive canvas, connecting them together to form an ***experiment***, which you run in Machine Learning Studio. To iterate on your model design, you edit the experiment, save a copy if desired, and run it again. When you're ready, you can convert your ***training experiment*** to a ***predictive experiment***, and then publish it as a ***web service*** so that your model can be accessed by others.

There is no programming required, just visually connecting datasets and modules to construct your predictive analysis model.

What is Azure ML Studio?



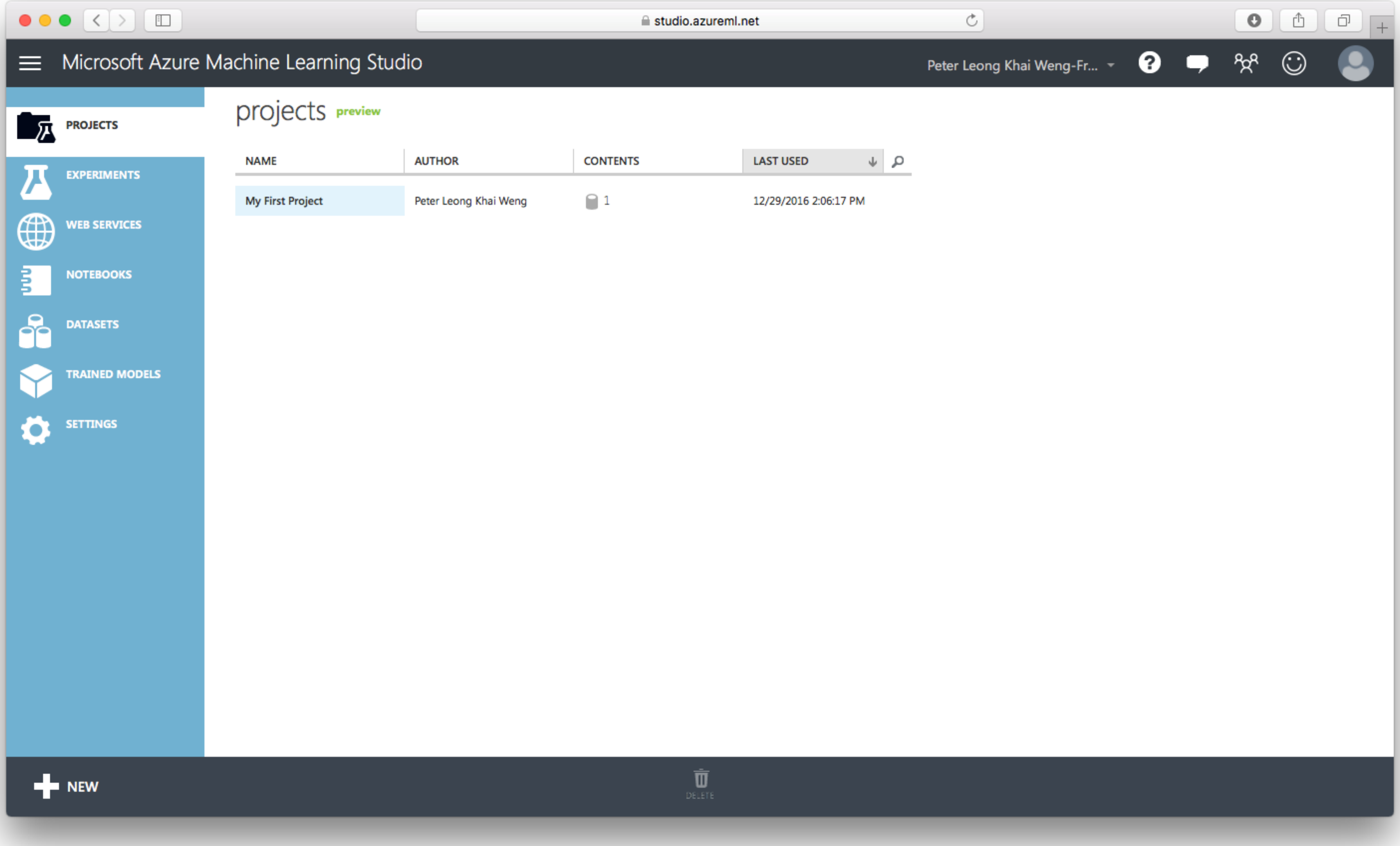
source https://docs.microsoft.com/en-us/azure/machine-learning/studio/what-is-ml-studio

What is Azure ML Studio?

http://studio.azureml.net

First you'll be asked to sign in using your Microsoft account, or your work or school account. Once signed in, you'll see the following tabs on the left:

- **PROJECTS** - Collections of experiments, datasets, notebooks, and other resources representing a single project
- **EXPERIMENTS** - Experiments that you have created and run or saved as drafts
- **WEB SERVICES** - Web services that you have deployed from your experiments
- **NOTEBOOKS** - Jupyter notebooks that you have created
- **DATASETS** - Datasets that you have uploaded into Studio
- **TRAINED MODELS** - Models that you have trained in experiments and saved in Studio
- **SETTINGS** - A collection of settings that you can use to configure your account and resources.



studio.azureml.net

Microsoft Azure Machine Learning Studio

Peter Leong Khai Weng-Fr...

PROJECTS
EXPERIMENTS
WEB SERVICES
NOTEBOOKS
DATASETS
TRAINED MODELS
SETTINGS

projects preview

NAME	AUTHOR	CONTENTS	LAST USED
My First Project	Peter Leong Khai Weng	1	12/29/2016 2:06:17 PM

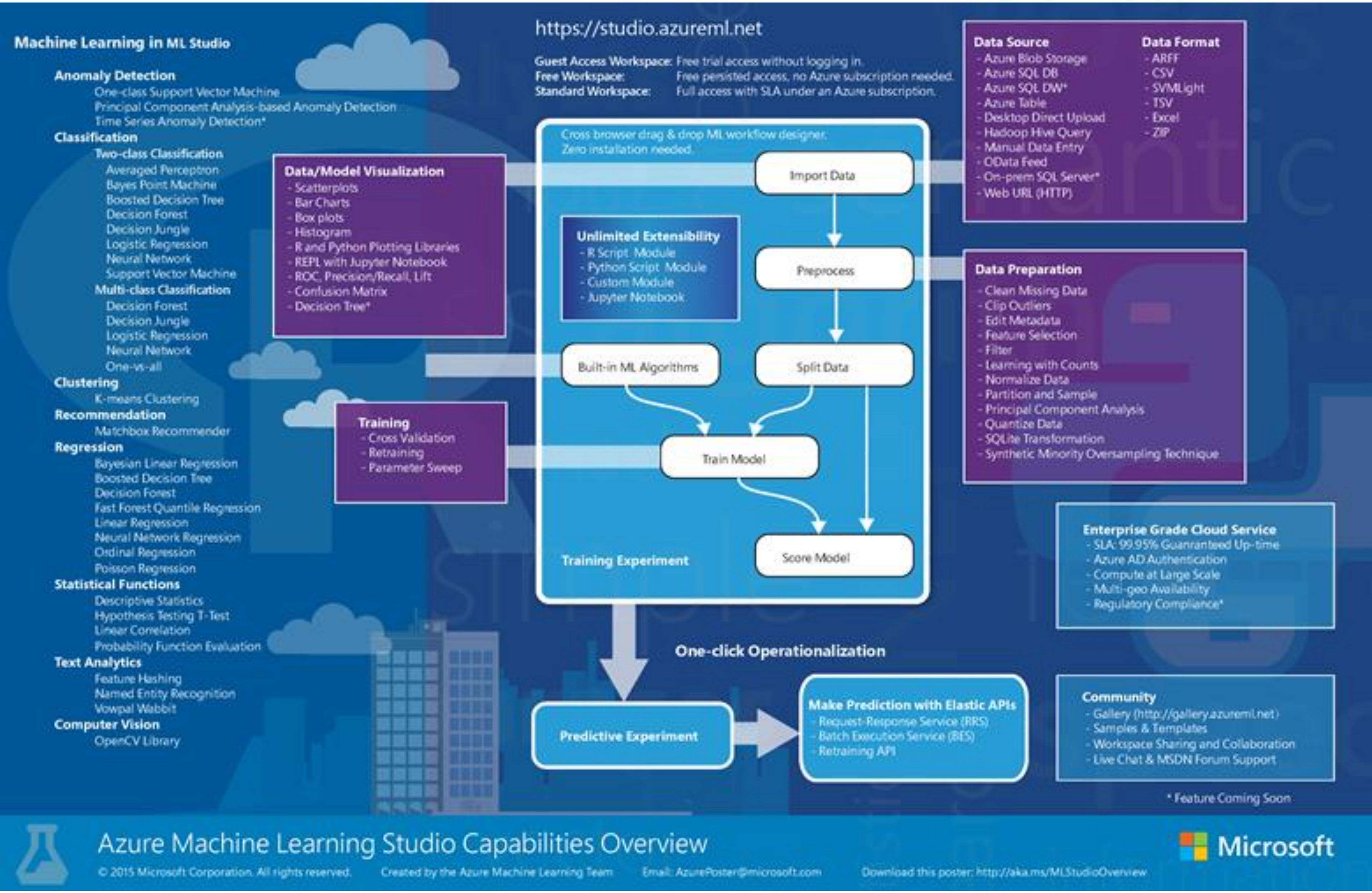
NEW

DELETE

The Azure ML Studio Capabilities Overview

The **Microsoft Azure Machine Learning Studio Capabilities Overview** diagram gives you a high-level overview of how you can use Machine Learning Studio to develop a predictive analytics model and operationalize it in the Azure cloud.

Azure Machine Learning Studio has available a large number of machine learning algorithms, along with modules that help with data input, output, preparation, and visualization. Using these components you can develop a predictive analytics experiment, iterate on it, and use it to train your model. Then with one click you can operationalize your model in the Azure cloud so that it can be used to score new data.



Machine Learning in ML Studio

Anomaly Detection
- One-class Support Vector Machine
- Principal Component Analysis-based Anomaly Detection
- Time Series Anomaly Detection*

Classification
- **Two-class Classification**
  - Averaged Perceptron
  - Bayes Point Machine
  - Boosted Decision Tree
  - Decision Forest
  - Decision Jungle
  - Logistic Regression
  - Neural Network
  - Support Vector Machine
- **Multi-class Classification**
  - Decision Forest
  - Decision Jungle
  - Logistic Regression
  - Neural Network
  - One-vs-all

Clustering
- K-means Clustering

Recommendation
- Matchbox Recommender

Regression
- Bayesian Linear Regression
- Boosted Decision Tree
- Decision Forest
- Fast Forest Quantile Regression
- Linear Regression
- Neural Network Regression
- Ordinal Regression
- Poisson Regression

Statistical Functions
- Descriptive Statistics
- Hypothesis Testing T-Test
- Linear Correlation
- Probability Function Evaluation

Text Analytics
- Feature Hashing
- Named Entity Recognition
- Vowpal Wabbit

Computer Vision
- OpenCV Library

https://studio.azureml.net

Guest Access Workspace: Free trial access without logging in.
Free Workspace: Free persisted access, no Azure subscription needed.
Standard Workspace: Full access with SLA under an Azure subscription.

Data/Model Visualization
- Scatterplots
- Bar Charts
- Box plots
- Histogram
- R and Python Plotting Libraries
- REPL with Jupyter Notebook
- ROC, Precision/Recall, Lift
- Confusion Matrix
- Decision Tree*

Training
- Cross Validation
- Retraining
- Parameter Sweep

Cross browser drag & drop ML workflow designer.
Zero installation needed.

Unlimited Extensibility
- R Script Module
- Python Script Module
- Custom Module
- Jupyter Notebook

Make Prediction with Elastic APIs
- Request-Response Service (RRS)
- Batch Execution Service (BES)
- Retraining API

Data Source	Data Format
- Azure Blob Storage	- ARFF
- Azure SQL DB	- CSV
- Azure SQL DW*	- SVMLight
- Azure Table	- TSV
- Desktop Direct Upload	- Excel
- Hadoop Hive Query	- ZIP
- Manual Data Entry	
- OData Feed	
- On-prem SQL Server*	
- Web URL (HTTP)	

Data Preparation
- Clean Missing Data
- Clip Outliers
- Edit Metadata
- Feature Selection
- Filter
- Learning with Counts
- Normalize Data
- Partition and Sample
- Principal Component Analysis
- Quantize Data
- SQLite Transformation
- Synthetic Minority Oversampling Technique

Enterprise Grade Cloud Service
- SLA: 99.95% Guaranteed Up-time
- Azure AD Authentication
- Compute at Large Scale
- Multi-geo Availability
- Regulatory Compliance*

Community
- Gallery (http://gallery.azureml.net)
- Samples & Templates
- Workspace Sharing and Collaboration
- Live Chat & MSDN Forum Support

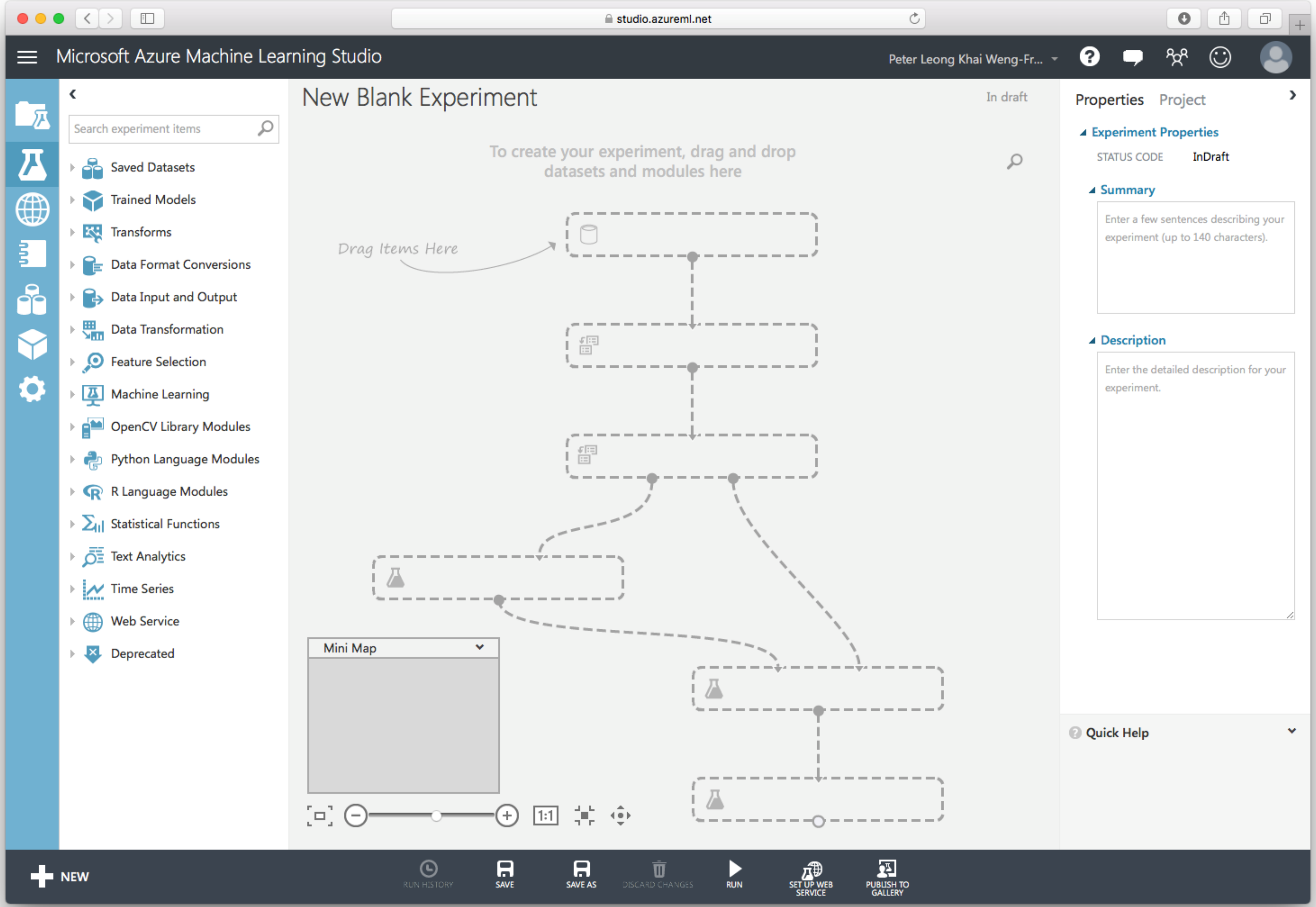
* Feature Coming Soon

Azure Machine Learning Studio Capabilities Overview

Created by the Azure Machine Learning Team Email: AzurePoster@microsoft.com Download this poster: http://aka.ms/MLStudioOverview

Microsoft

source https://docs.microsoft.com/en-us/azure/machine-learning/studio/studio-overview-diagram
Download http://download.microsoft.com/download/C/4/6/C4606116-522F-428A-BE04-B6D3213E9E52/ml_studio_overview_v1.1.pdf



New Blank Experiment

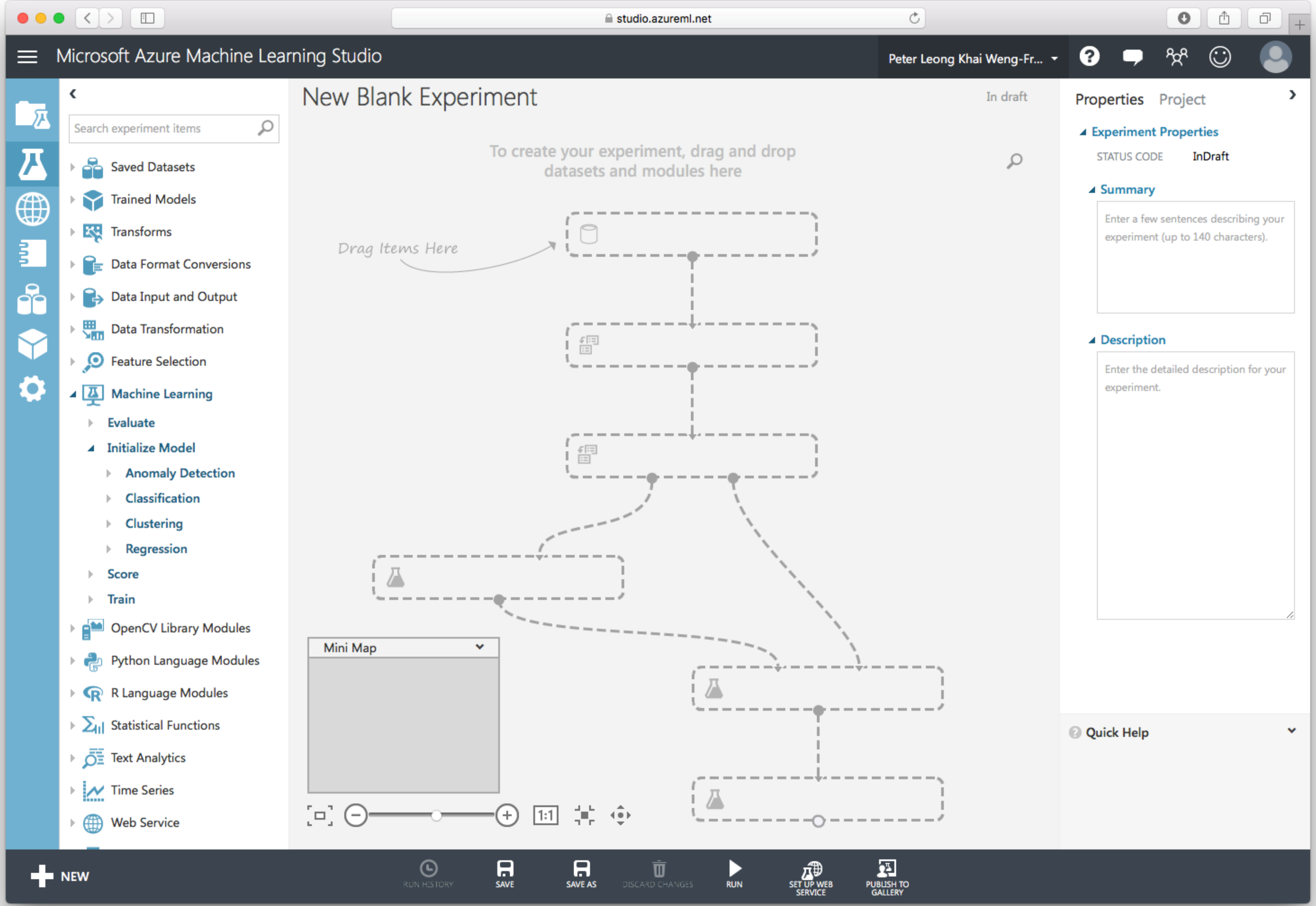
In draft

To create your experiment, drag and drop datasets and modules here

STATUS CODE InDraft

Enter a few sentences describing your experiment (up to 140 characters).

Enter the detailed description for your experiment.



studio.azureml.net
Microsoft Azure Machine Learning Studio
Peter Leong Khai Weng-Fr...
Search experiment items
Saved Datasets
Trained Models
Transforms
Data Format Conversions
Data Input and Output
Data Transformation
Feature Selection
Machine Learning
Evaluate
Initialize Model
Anomaly Detection
Classification
Clustering
Regression
Score
Train
OpenCV Library Modules
Python Language Modules
R Language Modules
Statistical Functions
Text Analytics
Time Series
Web Service
New Blank Experiment
In draft
To create your experiment, drag and drop datasets and modules here
Drag Items Here
Mini Map
1:1
Properties
Project
Experiment Properties
STATUS CODE
InDraft
Summary
Enter a few sentences describing your experiment (up to 140 characters).
Description
Enter the detailed description for your experiment.
Quick Help
NEW
RUN HISTORY
SAVE
SAVE AS
DISCARD CHANGES
RUN
SET UP WEB SERVICE
PUBLISH TO GALLERY

Choosing a ML algorithm

- Every machine learning algorithm has its own style or ***inductive bias***. For a specific problem, several algorithms may be appropriate and one algorithm may be a better fit than others. But it's not always possible to know beforehand which is the best fit. In cases like these, several algorithms are listed together in the cheat sheet. An appropriate strategy would be to try one algorithm, and if the results are not yet satisfactory, try the others.

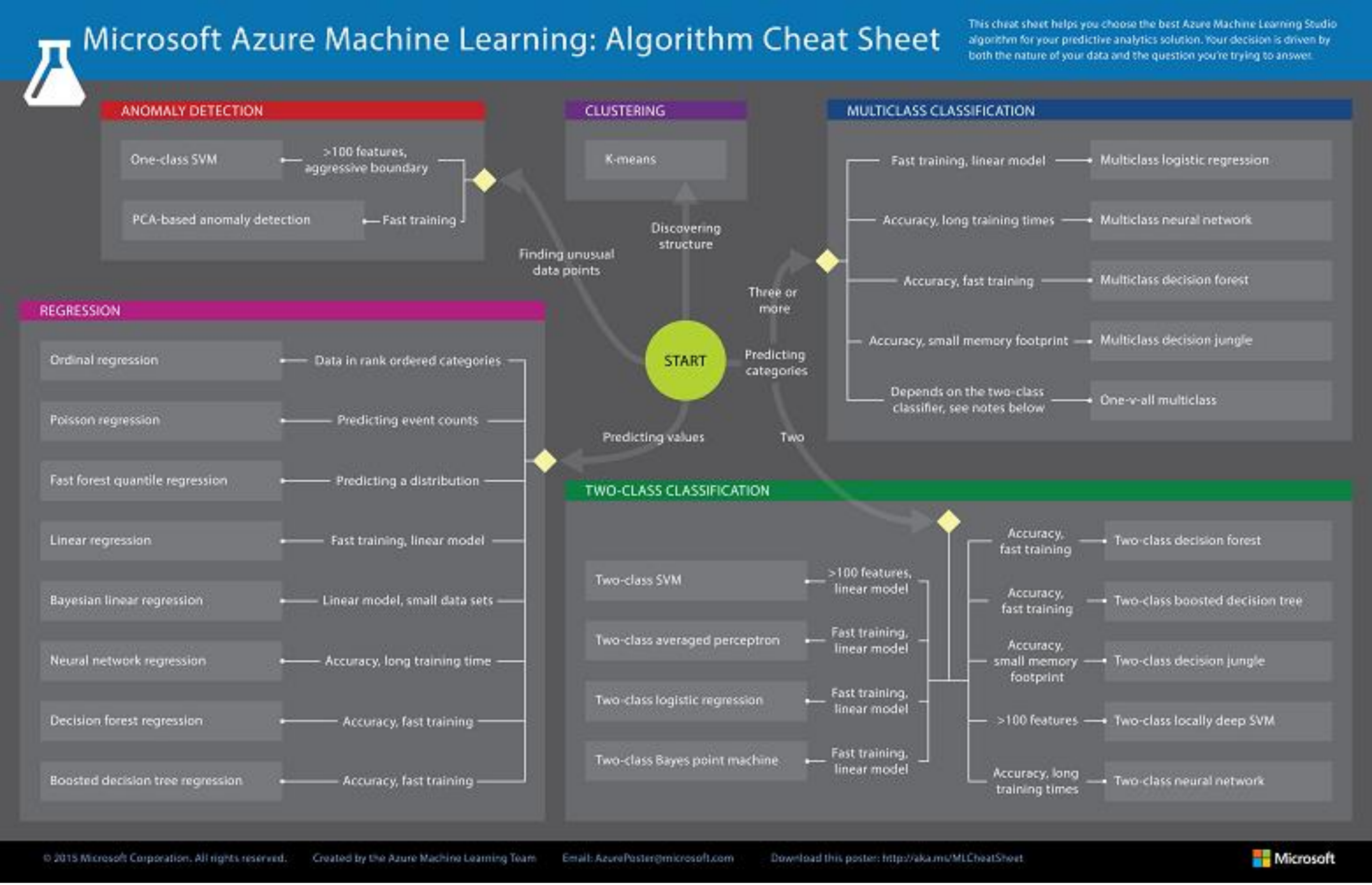
Choosing a ML algorithm

There are three main categories of machine learning: **supervised learning**, **unsupervised learning**, and **reinforcement learning**.

- In **supervised learning**, each data point is **labelled** or associated with a **category** or **value** of interest. An example of a categorical label is assigning an image as either a ‘cat’ or a ‘dog’. An example of a value label is the sale price associated with a used car. The goal of supervised learning is to study many labelled examples like these, and then to be able to make predictions about future data points. For example, identifying new photos with the correct animal or assigning accurate sale prices to other used cars. This is a popular and useful type of machine learning. All of the modules in Azure Machine Learning are supervised learning algorithms except for K-Means Clustering.

Choosing a ML algorithm

- In **unsupervised learning**, data points have no labels associated with them. Instead, the goal of an unsupervised learning algorithm is to organize the data in some way or to describe its structure. This can mean grouping it into clusters, as K-means does, or finding different ways of looking at complex data so that it appears simpler.
- In **reinforcement learning**, the algorithm gets to choose an action in response to each data point. It is a common approach in robotics, where the set of sensor readings at one point in time is a data point, and the algorithm must choose the robot’s next action. It's also a natural fit for Internet of Things applications. The learning algorithm also receives a reward signal a short time later, indicating how good the decision was. Based on this, the algorithm modifies its strategy in order to achieve the highest reward. Currently there are no reinforcement learning algorithm modules in Azure ML.



Microsoft Azure Machine Learning: Algorithm Cheat Sheet

This cheat sheet helps you choose the best Azure Machine Learning Studio algorithm for your predictive analytics solution. Your decision is driven by both the nature of your data and the question you're trying to answer.

START

ANOMALY DETECTION

Finding unusual data points

- >100 features, aggressive boundary → One-class SVM
- Fast training → PCA-based anomaly detection

CLUSTERING

Discovering structure

- K-means

MULTICLASS CLASSIFICATION

Predicting categories → Three or more

- Fast training, linear model → Multiclass logistic regression
- Accuracy, long training times → Multiclass neural network
- Accuracy, fast training → Multiclass decision forest
- Accuracy, small memory footprint → Multiclass decision jungle
- Depends on the two-class classifier, see notes below → One-v-all multiclass

REGRESSION

Predicting values

- Data in rank ordered categories → Ordinal regression
- Predicting event counts → Poisson regression
- Predicting a distribution → Fast forest quantile regression
- Fast training, linear model → Linear regression
- Linear model, small data sets → Bayesian linear regression
- Accuracy, long training time → Neural network regression
- Accuracy, fast training → Decision forest regression
- Accuracy, fast training → Boosted decision tree regression

TWO-CLASS CLASSIFICATION

Predicting categories → Two

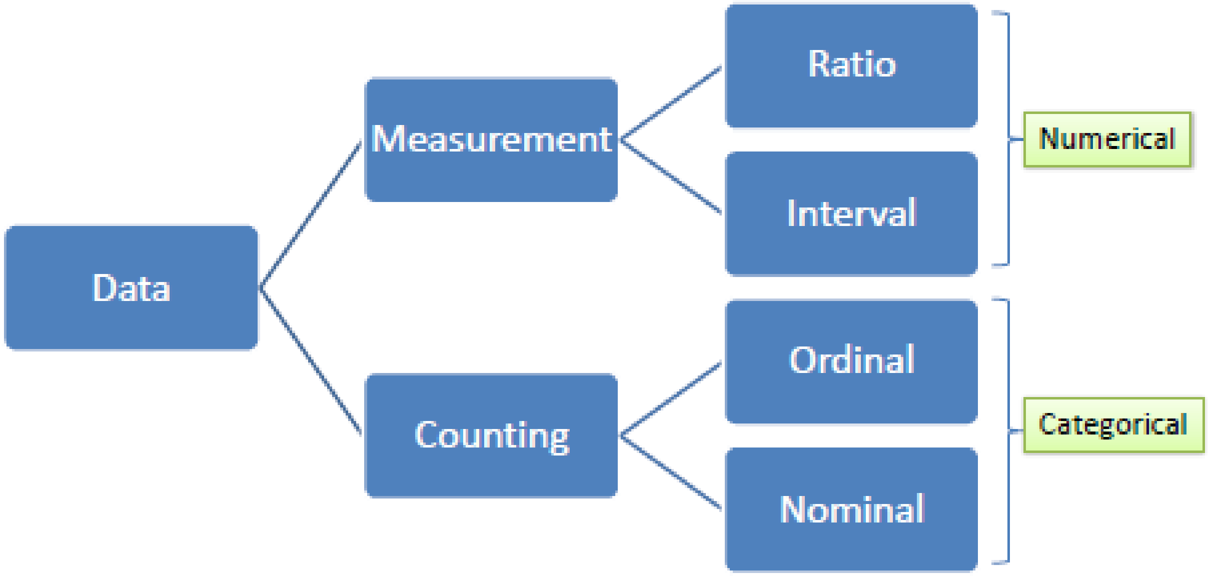
- >100 features, linear model → Two-class SVM
- Fast training, linear model → Two-class averaged perceptron
- Fast training, linear model → Two-class logistic regression
- Fast training, linear model → Two-class Bayes point machine
- Accuracy, fast training → Two-class decision forest
- Accuracy, fast training → Two-class boosted decision tree
- Accuracy, small memory footprint → Two-class decision jungle
- >100 features → Two-class locally deep SVM
- Accuracy, long training times → Two-class neural network

Created by the Azure Machine Learning Team Email: AzurePoster@microsoft.com Download this poster: http://aka.ms/MLCheatSheet

source https://docs.microsoft.com/en-us/azure/machine-learning/studio/algorithm-cheat-sheet
download http://download.microsoft.com/download/A/6/1/A613E11E-8F9C-424A-B99D-65344785C288/microsoft-machine-learning-algorithm-cheat-sheet-v6.pdf

Understand Principles of Data Representation and Feature Engineering

Data Types



source http://www.saedsayad.com/data_preparation.htm

Data Types

Data is information typically the results of measurement (numerical) or counting (categorical). Variables serve as placeholders for data. There are two types of variables, numerical and categorical.

- A **numerical** or **continuous variable** is one that can accept any value within a finite or infinite interval (e.g., height, weight, temperature, blood glucose, ...). There are two types of numerical data, ***interval*** and ***ratio***. Data on an interval scale can be added and subtracted but cannot be meaningfully multiplied or divided because there is no true zero. For example, we cannot say that one day is twice as hot as another day. On the other hand, data on a ratio scale has true zero and can be added, subtracted, multiplied or divided (e.g., weight).
- A **categorical** or **discrete variable** is one that can accept two or more values (categories). There are two types of categorical data, ***nominal*** and ***ordinal***. Nominal data does not have an intrinsic ordering in the categories. For example, "gender" with two categories, male and female. In contrast, ordinal data does have an intrinsic ordering in the categories. For example, "level of energy" with three orderly categories (low, medium and high).

ML algorithms require the inputs variables to be numerical

- No problem for data that is already a real number or integer
- Categorical variables need to be converted or encoded into a number

Encoding categorical variables

- Use of dummy variables encoding for categorical variables

Original values (Direction)	I0	I1	I2
NORTH	1	0	0
EAST	0	1	0
SOUTH	0	0	1
WEST	0	0	0

Number of dummy/indicator variables = number of values – 1
In the case 4 possible values – 1 = 3 (I0,I1,I2 dummy variables)
The original column is replaced by 3 new columns

Encoding categorical variables

- Use of one-hot encoding for categorical variables

Original values (Direction)	I0	I1	I2	I3
NORTH	1	0	0	0
EAST	0	1	0	0
SOUTH	0	0	1	0
WEST	0	0	0	1

One-hot encoding of variables = number of values
In the case 4 possible values = 4 (I0,I1,I2,I3 variables)
The original column is replaced by 4 new columns

Why not just use a value encoding (e.g. LabelEncoder)?

The problem with value encoding or LabelEncoder is that is leads to undesirable side effects if the ML algorithm uses the numerical values. For example, Average of East and West = (1+3)/2 = 2 (South).

This is nonsense as the average of East and West has no meaning.

Direction	Value
NORTH	0
EAST	1
SOUTH	2
WEST	3

Discretization/Binning

Discretization (otherwise known as quantization or binning) provides a way to partition continuous features into discrete values.

Certain datasets with continuous features may benefit from discretization, because discretization can transform the dataset of continuous attributes to one with only nominal attributes.

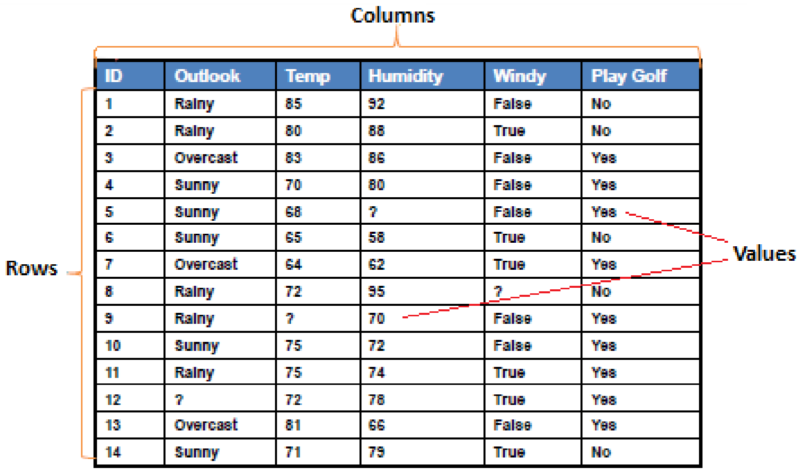
Time	TimeOfDay
00:00	0
07:30	1
08:00	1
12:00	2
13:00	2
14:00	2
17:00	2

We convert the 24H into 4 time period bins
00:00-06:00 is the bin 0 for early morning

Dataset

Dataset is a collection of data, usually presented in a tabular form. Each **column** represents a particular **variable** (**attribute**), and each **row** corresponds to a given **member** (**observation**) of the data.

In predictive modeling, **predictors** or **attributes** are the **input variables** and **target** or **class attribute** is the **output variable** whose value is determined by the values of the predictors and function of the predictive model.


Columns
Rows
Values

ID	Outlook	Temp	Humidity	Windy	Play Golf
1	Rainy	85	92	False	No
2	Rainy	80	88	True	No
3	Overcast	83	86	False	Yes
4	Sunny	70	80	False	Yes
5	Sunny	68	?	False	Yes
6	Sunny	65	58	True	No
7	Overcast	64	62	True	Yes
8	Rainy	72	95	?	No
9	Rainy	?	70	False	Yes
10	Sunny	75	72	False	Yes
11	Rainy	75	74	True	Yes
12	?	72	78	True	Yes
13	Overcast	81	66	False	Yes
14	Sunny	71	79	True	No

Dataset

There are some alternatives terms for columns, rows and values.

- Columns, Fields, Attributes, Variables
- Rows, Records, Objects, Cases, Instances, Examples, Vectors
- Values, Data

Feature Engineering

A **feature** is an **attribute** or **property** shared by all of the independent units on which analysis or prediction is to be done. Any attribute could be a feature, as long as it is useful to the model.

Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work.

- **feature engineering**: This process attempts to create additional relevant features from the existing raw features in the data, and to increase the predictive power of the learning algorithm.
- **feature selection**: This process selects the key subset of original data features in an attempt to reduce the dimensionality of the training problem.

Normally **feature engineering** is applied first to generate additional features, and then the **feature selection** step is performed to eliminate irrelevant, redundant, or highly correlated features.

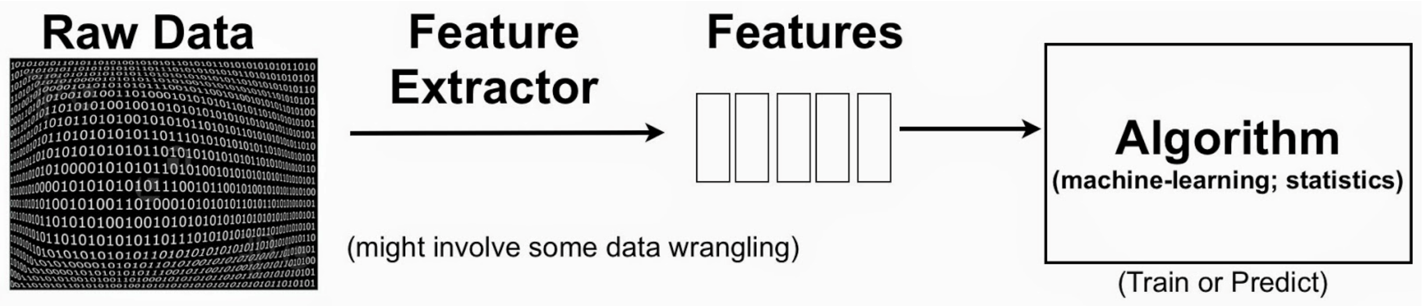
Feature engineering is fundamental to the application of machine learning, and is both difficult and expensive.

source https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/create-features

Feature Engineering

The **training data** used in machine learning can often be enhanced by **extraction of features** from the raw data collected. An example of an engineered feature in the context of learning how to classify the images of handwritten characters is creation of a bit density map constructed from the raw bit distribution data. This map can help locate the edges of the characters more efficiently than simply using the raw distribution directly.

What kind of features should be created to enhance the dataset when training a model? Engineered features that enhance the training provide information that better differentiates the patterns in the data. The new features are expected to provide additional information that is not clearly captured or easily apparent in the original or existing feature set. But **this process is something of an art**. Sound and productive decisions often require some domain expertise.


Raw Data
Feature Extractor
Features
Algorithm
(machine-learning; statistics)
(might involve some data wrangling)
(Train or Predict)

Feature Engineering

- Engineered and selected features increase the efficiency of the training process, which attempts to extract the key information contained in the data.
- They also improve the power of these models to classify the input data accurately and to predict outcomes of interest more robustly.
- Feature engineering and selection can also combine to make the learning more computationally tractable. It does so by enhancing and then reducing the number of features needed to calibrate or train a model. Mathematically speaking, the features selected to train the model are a minimal set of independent variables that explain the patterns in the data and then predict outcomes successfully.

Feature Selection vs Feature Extraction

In **feature selection** we try to find the best subset of the input feature set

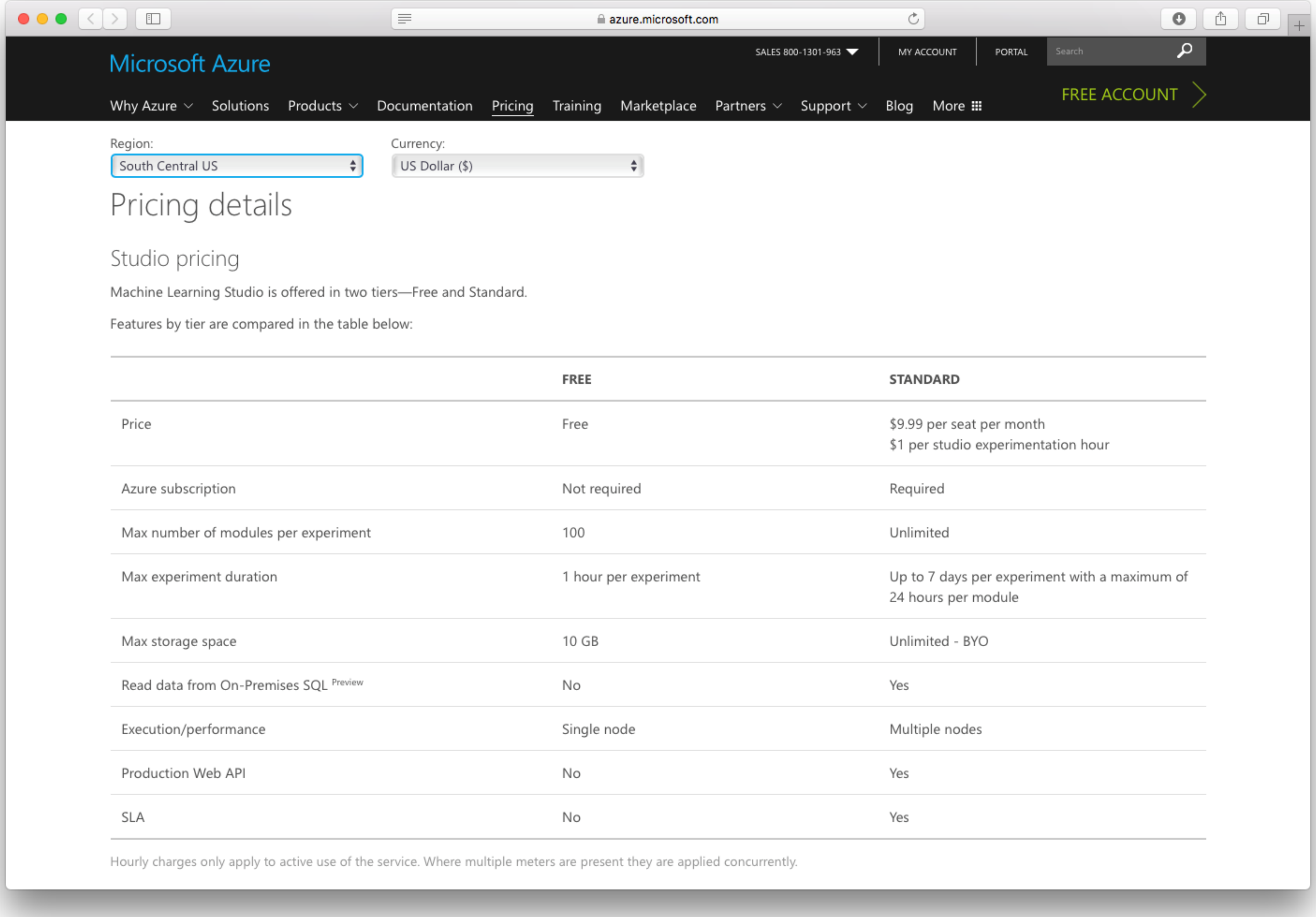
In **feature extraction** we create new features based on the transformation or combination of the original feature set

Summary

We have learnt that:

- Azure Machine Learning Studio is a cloud based tool for using Machine Learning (ML) algorithm without any coding required
- Azure ML Studio allows us to familiarize with the ML workflow and also quickly prototype some possible approaches for our modelling
- Data may need to be transformed/encoded before it can be fed to a machine algorithm
- Categorical variables can be encoded as dummy variables
- Sometimes it is useful to discretize numeric values
- Feature engineering is the part that needs real intelligence

Additional Slides



Region: South Central US

Currency: US Dollar ($)

Pricing details

Studio pricing

Machine Learning Studio is offered in two tiers—Free and Standard.

Features by tier are compared in the table below:

	FREE	STANDARD
Price	Free	$9.99 per seat per month $1 per studio experimentation hour
Azure subscription	Not required	Required
Max number of modules per experiment	100	Unlimited
Max experiment duration	1 hour per experiment	Up to 7 days per experiment with a maximum of 24 hours per module
Max storage space	10 GB	Unlimited - BYO
Read data from On-Premises SQL Preview	No	Yes
Execution/performance	Single node	Multiple nodes
Production Web API	No	Yes
SLA	No	Yes

Hourly charges only apply to active use of the service. Where multiple meters are present they are applied concurrently.