Building Machine Learning Models for Predictive Maintenance Applications

Yan Zhang, PhD
Data Scientist
zhangya@Microsoft.com

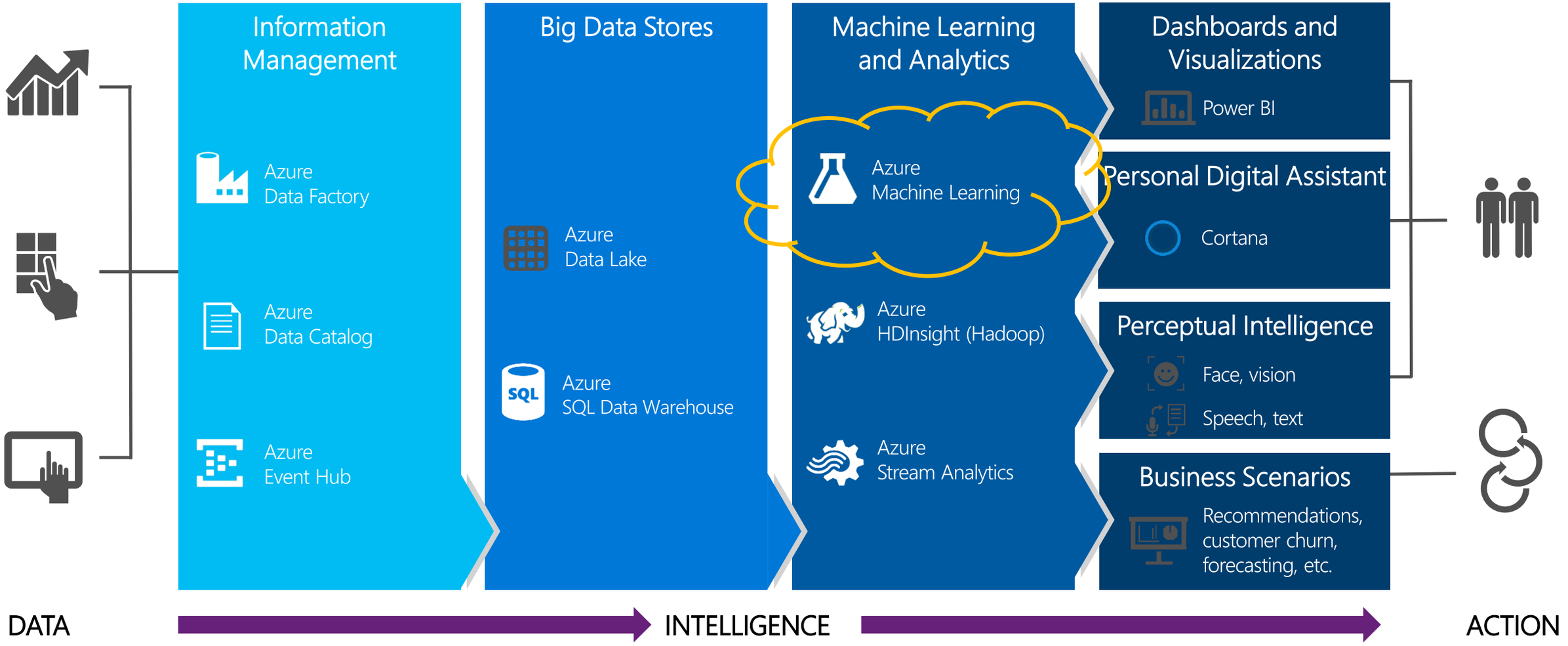
September 2015

©2015 Microsoft Corporation. All rights reserved.

Microsoft

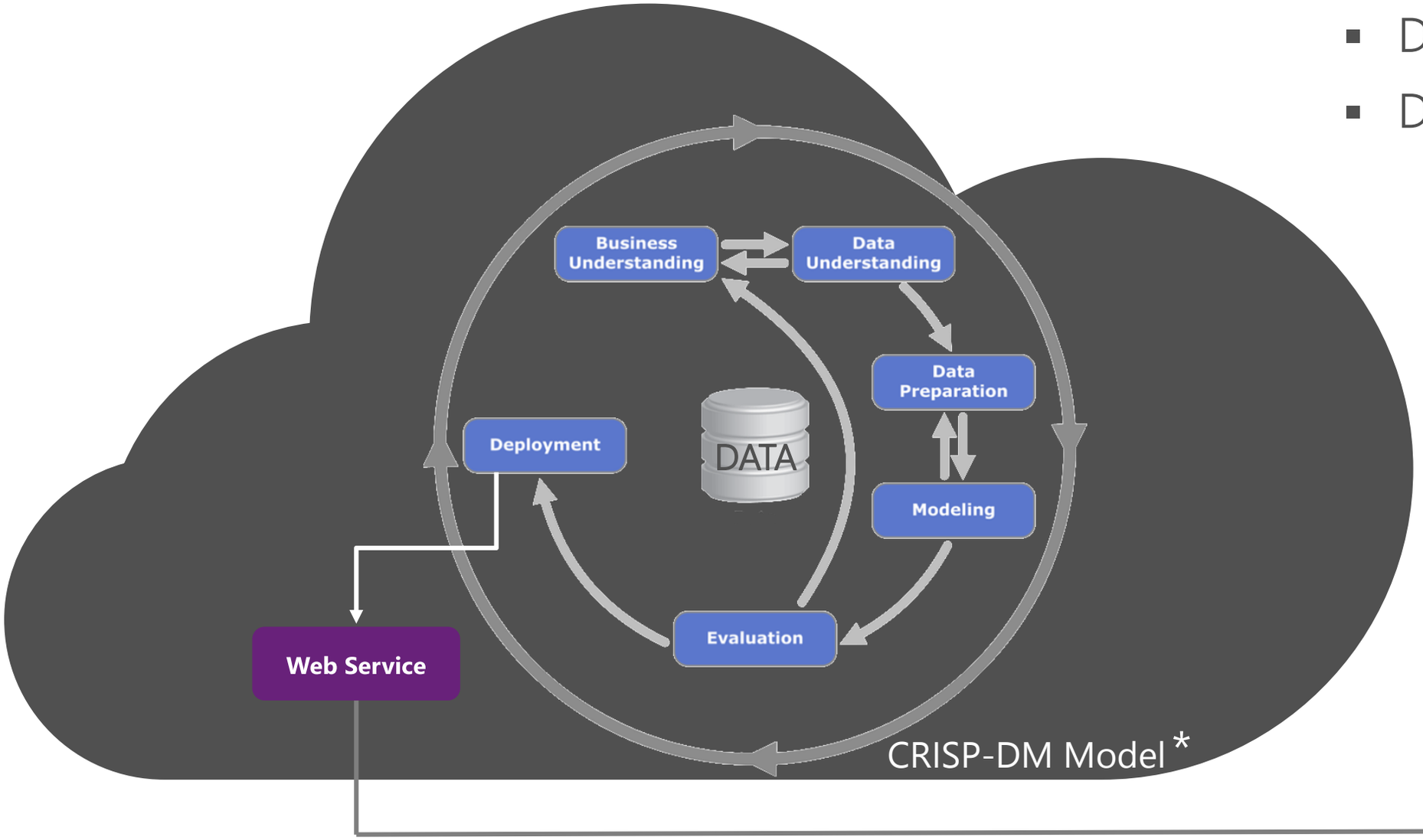
Cortana Analytics Suite

Transform data into intelligent action

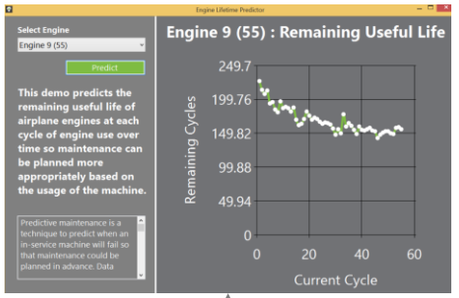


Data Pipeline

- Data science process
- Data pipeline



Desktop app



* Graph from wikipedia.org: Cross Industry Standard Process for Data Mining

Outline

Predictive Maintenance Use Case

- Predictive maintenance in IoT applications vs. traditional predictive maintenance concepts
- Predictive problem: “When an in-service machine will fail?”
- Machine learning approach
- Problem formulation

Data

- Input data – publicly available aircraft engine run-to-failure data
- Data labeling and feature engineering

Tools to build end-to-end solution with Azure ML

- Azure ML
- Predictive Maintenance Template in Azure ML

Demo: desktop app to predict machine’s remaining useful life

Objectives

Understand data science process in general

Get an idea about how to predict future machine failure using machine learning approach

Know how to approach the problem “When an in-service machine will fail?” using Azure ML

Understand how to build an end-to-end solution: from data to web service

Predictive Maintenance Concepts

Important task in Internet of Things applications

Goal: improve production/maintenance efficiency

	Predictive Maintenance in IoT	**Traditional Predicative Maintenance**
Goal	Improve production and/or maintenance efficiency	Ensure the reliability of machine operation
Data	Data stream (time varying features), Multiple data sources	Very limited time varying features
Scope	Component level, System level	Parts level
Approach	Data driven	Model driven
Tasks	Failure prediction, fault/failure detection & diagnosis, maintenance actions recommendation, etc. Essentially any task that improves production/maintenance efficiency	Failure prediction (prognosis), fault/failure detection & diagnosis (diagnosis)

Predictive Maintenance Use Cases

Aerospace	Utilities	Manufacturing	Transportation & Logistics
What is the likelihood of delay due to mechanical issues?	When is my solar panel or wind turbine going to fail next?	Will the component pass the next stage of testing on factory floor or do I need to rework?	Should I replace the break disks in my car or can I wait for another month?
When is this aircraft component likely to fail next?	Which circuit breakers in my system are likely to fail in the next month?	What is the root cause of the test failure?	What maintenance task should I perform on my elevator?
	Is the ATM going to dispense the next 5 notes without failing?		

ML Modelling Techniques

BINARY CLASSIFICATION

Predict failures within a future period of time

REGRESSION

Predict remaining useful life, the amount of time before the next failure

MULTICLASS CLASSIFICATION

Predict failures with their causes within a future time period.

Predict remaining useful life within ranges of future periods

ANOMALY DETECTION

Identify change in normal trends to find anomalies

Input Data

http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/ [1]

6.Turbofan Engine Degradation Simulation Data Set

- Training data (“train_FD001.txt”): aircraft engine run-to-failure data
- Testing data (“test_FD001.txt”): aircraft engine operating data without failure events recorded
- Ground truth data (“RUL_FD001.txt”): the information of true remaining cycles for each engine in the testing data

Sample training data

~20k rows,
100 unique engine id

id	cycle	setting1	setting2	setting3	s1	s2	s3	...	s19	s20	s21
1	1	-0.0007	-0.0004	100	518.67	641.82	1589.7		100	39.06	23.419
1	2	0.0019	-0.0003	100	518.67	642.15	1591.82		100	39	23.4236
1	3	-0.0043	0.0003	100	518.67	642.35	1587.99		100	38.95	23.3442
...	...										
1	191	0	-0.0004	100	518.67	643.34	1602.36		100	38.45	23.1295
1	192	0.0009	0	100	518.67	643.54	1601.41		100	38.48	22.9649
2	1	-0.0018	0.0006	100	518.67	641.89	1583.84		100	38.94	23.4585
2	2	0.0043	-0.0003	100	518.67	641.82	1587.05		100	39.06	23.4085
2	3	0.0018	0.0003	100	518.67	641.55	1588.32		100	39.11	23.425
...	...										
2	286	-0.001	-0.0003	100	518.67	643.44	1603.63		100	38.33	23.0169
2	287	-0.0005	0.0006	100	518.67	643.85	1608.5		100	38.43	23.0848

Sample testing data

~13k rows,
100 unique engine id

id	cycle	setting1	setting2	setting3	s1	s2	s3	...	s19	s20	s21
1	1	0.0023	0.0003	100	518.67	643.02	1585.29		100	38.86	23.3735
1	2	-0.0027	-0.0003	100	518.67	641.71	1588.45		100	39.02	23.3916
1	3	0.0003	0.0001	100	518.67	642.46	1586.94		100	39.08	23.4166
...	...										
1	30	-0.0025	0.0004	100	518.67	642.79	1585.72		100	39.09	23.4069
1	31	-0.0006	0.0004	100	518.67	642.58	1581.22		100	38.81	23.3552
2	1	-0.0009	0.0004	100	518.67	642.66	1589.3		100	39	23.3923
2	2	-0.0011	0.0002	100	518.67	642.51	1588.43		100	38.84	23.2902
2	3	0.0002	0.0003	100	518.67	642.58	1595.6		100	39.02	23.4064
...	...										
2	48	0.0011	-0.0001	100	518.67	642.64	1587.71		100	38.99	23.2918
2	49	0.0018	-0.0001	100	518.67	642.55	1586.59		100	38.81	23.2618
3	1	-0.0001	0.0001	100	518.67	642.03	1589.92		100	38.99	23.296
3	2	0.0039	-0.0003	100	518.67	642.23	1597.31		100	38.84	23.3191
3	3	0.0006	0.0003	100	518.67	642.98	1586.77		100	38.69	23.3774
...	...										
3	125	0.0014	0.0002	100	518.67	643.24	1588.64		100	38.56	23.227
3	126	-0.0016	0.0004	100	518.67	642.88	1589.75		100	38.93	23.274

Sample ground truth data

100 rows

RUL
112
98
69
82
91

Please refer to following link of doc for Data description section

Data Labeling

Regression

Binary classification

Multi-class classification

id	cycle	...	RUL	label1	label2
1	1		191	0	0
1	2		190	0	0
1	3		189	0	0
1	4		188	0	0
...			...	...	...
1	160		32	0	0
1	161		31	0	0
1	162		30	1	1
1	163		29	1	1
1	164		28	1	1
1	165		27	1	1
1	166		26	1	1
1	167		25	1	1
1	168		24	1	1
1	169		23	1	1
1	170		22	1	1
1	171		21	1	1
1	172		20	1	1
1	173		19	1	1
1	174		18	1	1
1	175		17	1	1
1	176		16	1	1
1	177		15	1	2
1	178		14	1	2
1	179		13	1	2
1	180		12	1	2
1	181		11	1	2
1	182		10	1	2
1	183		9	1	2
1	184		8	1	2
1	185		7	1	2
1	186		6	1	2
1	187		5	1	2
1	188		4	1	2
1	189		3	1	2
1	190		2	1	2
1	191		1	1	2
1	192		0	1	2

w1

w0

Predefined window size for classification models

w1 = 30
w0 = 15

Feature Engineering

The process of creating features that provide better or additional predictive power to the learning algorithm.

Aggregate Features

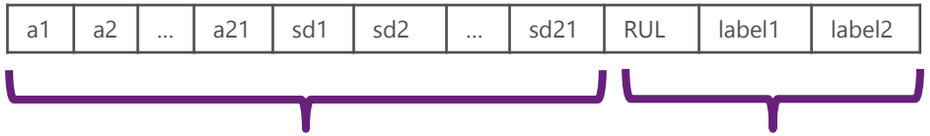
a1-a21: the moving average of sensor values in the most w recent cycles
sd1-sd21: the standard deviation of sensor values in the most w recent cycles

The time window parameter *w* is predefined.

Prepared Training Data

After data labeling and feature engineering

id	cycle	setting1	setting2	setting3	s1	s2	s3	...	s19	s20	s21
1	1	-0.0007	-0.0004	100	518.67	641.82	1589.7		100	39.06	23.419
1	2	0.0019	-0.0003	100	518.67	642.15	1591.82		100	39	23.4236
1	3	-0.0043	0.0003	100	518.67	642.35	1587.99		100	38.95	23.3442
...	...										
1	191	0	-0.0004	100	518.67	643.34	1602.36		100	38.45	23.1295
1	192	0.0009	0	100	518.67	643.54	1601.41		100	38.48	22.9649
2	1	-0.0018	0.0006	100	518.67	641.89	1583.84		100	38.94	23.4585
2	2	0.0043	-0.0003	100	518.67	641.82	1587.05		100	39.06	23.4085
2	3	0.0018	0.0003	100	518.67	641.55	1588.32		100	39.11	23.425
...	...										
2	286	-0.001	-0.0003	100	518.67	643.44	1603.63		100	38.33	23.0169
2	287	-0.0005	0.0006	100	518.67	643.85	1608.5		100	38.43	23.0848



Other potential features: change from initial value, velocity of change, frequency count over a predefined threshold

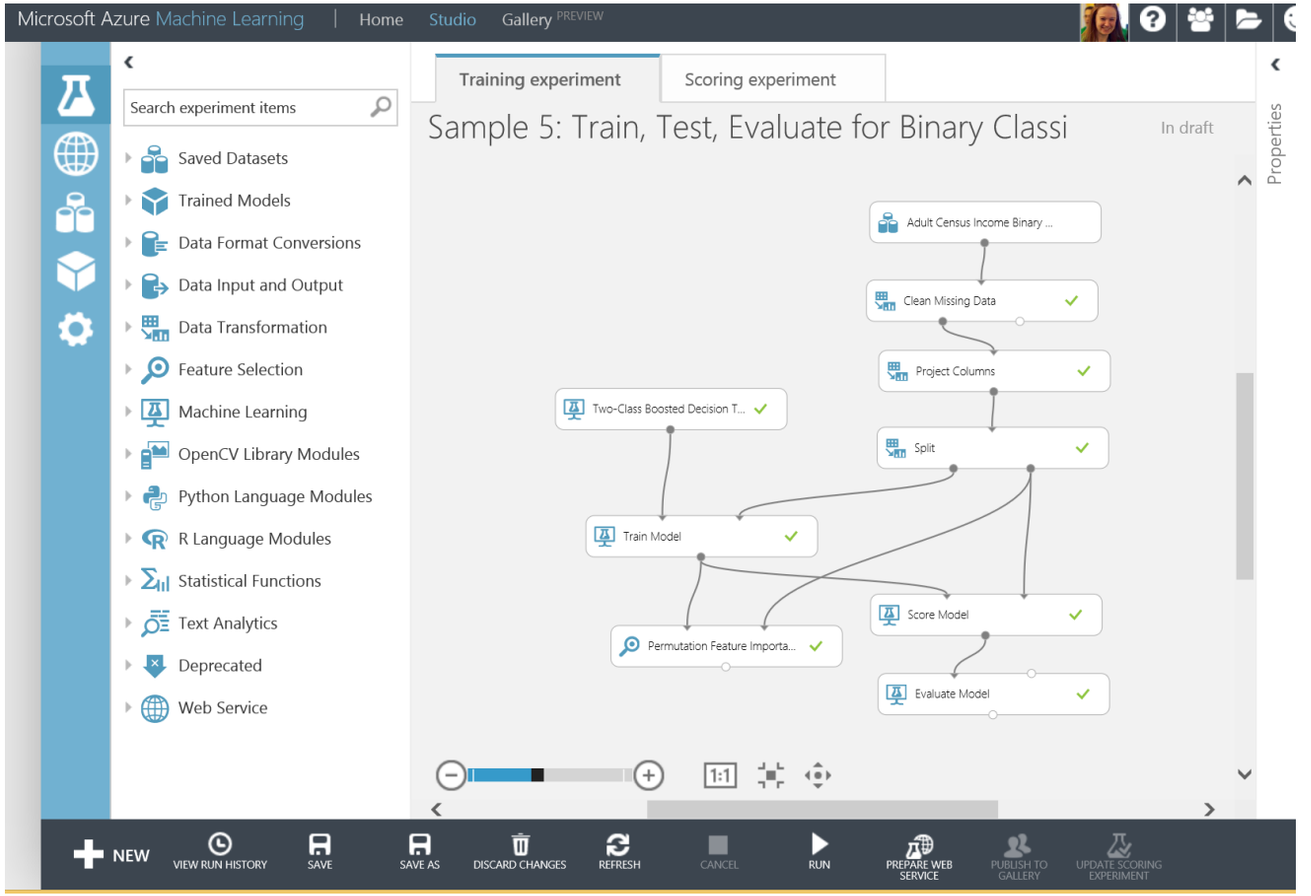
Predictive Maintenance Template in Azure ML

http://gallery.azureml.net (search “predictive maintenance”)

Microsoft Azure Machine Learning

http://azure.com/ml
free tier & standard tier

- Accessible through a web browser, no software to install
- Best ML algorithms
- Extensible, support for R & Python
- Collaborative work with anyone, anywhere via Azure workspace
- Visual composition with end2end support for data science workflow

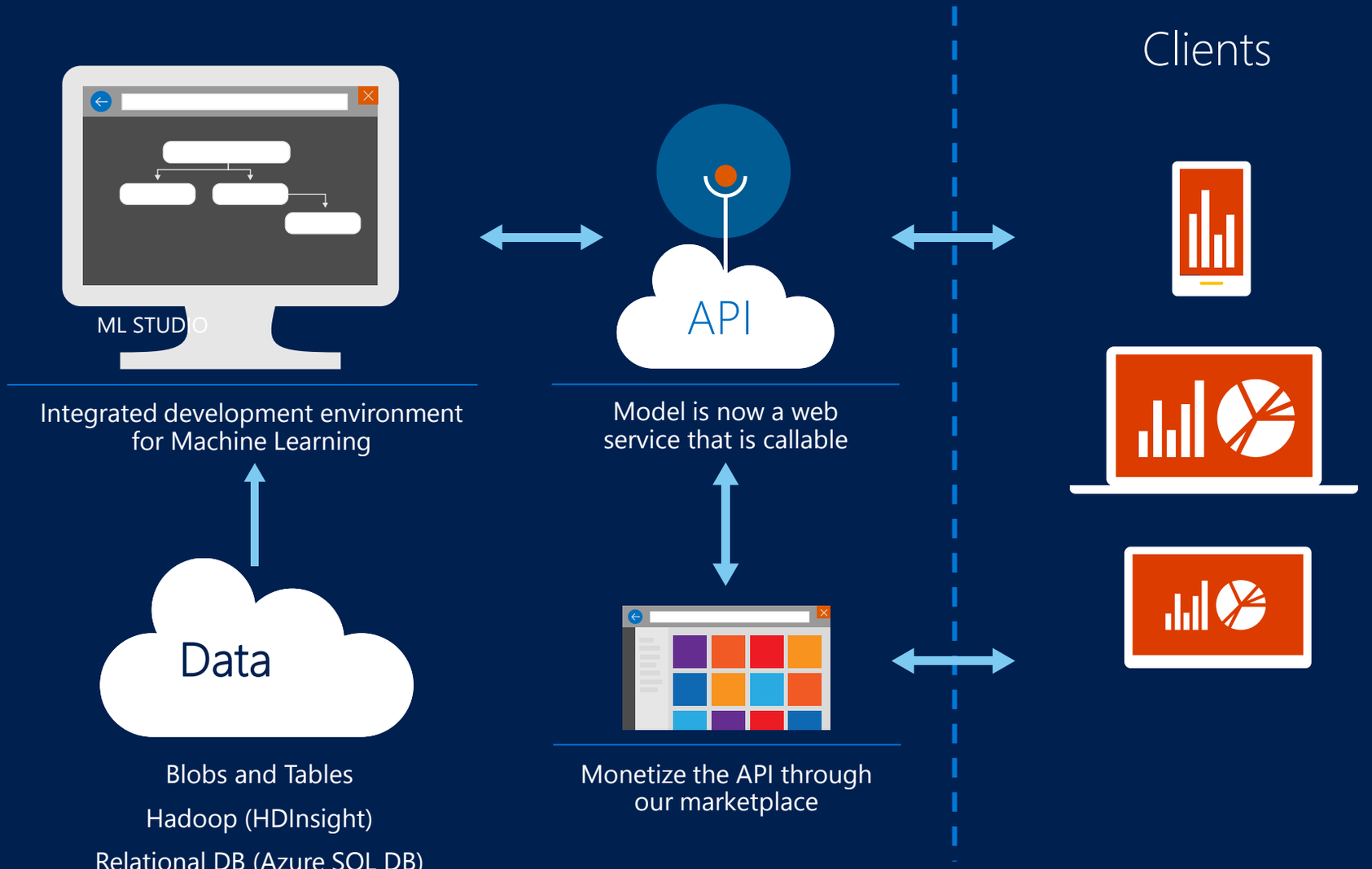


Key Differentiator of Azure ML

Data -> Predictive model -> Operational web API in minutes

ML model can be easily turned to

- Operationalized solution
- Monetized API



Predictive Maintenance Template Experiments Flow Diagram

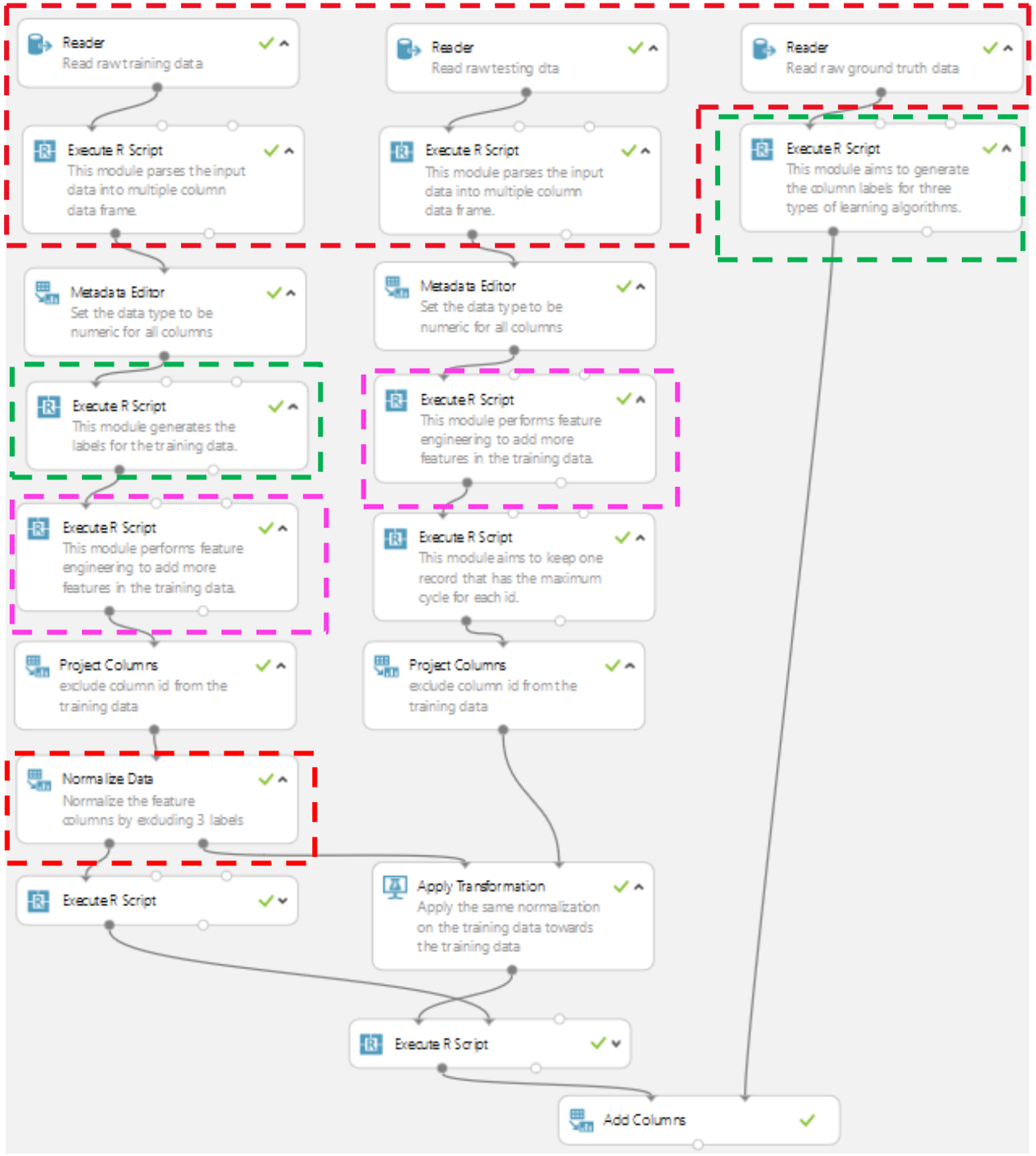
Step 1 experiment

Data input

Data labeling

Feature engineering

Data normalization



Step 2: Train and Evaluate Model

Regression models

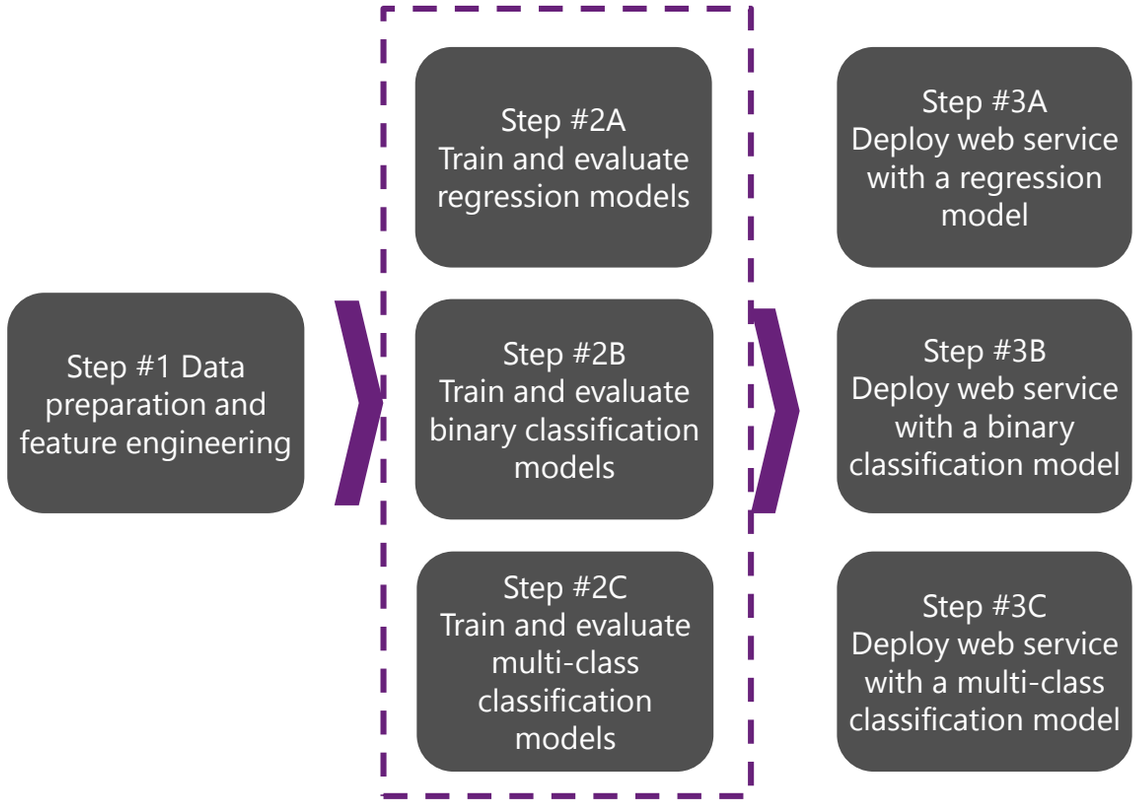
- Decision Forest Regression
- Boosted Decision Tree Regression
- Poisson Regression
- Neural Network Regression

Binary classification models

- Logistic Regression
- Boosted Decision Tree
- Decision Forest
- Neural Network

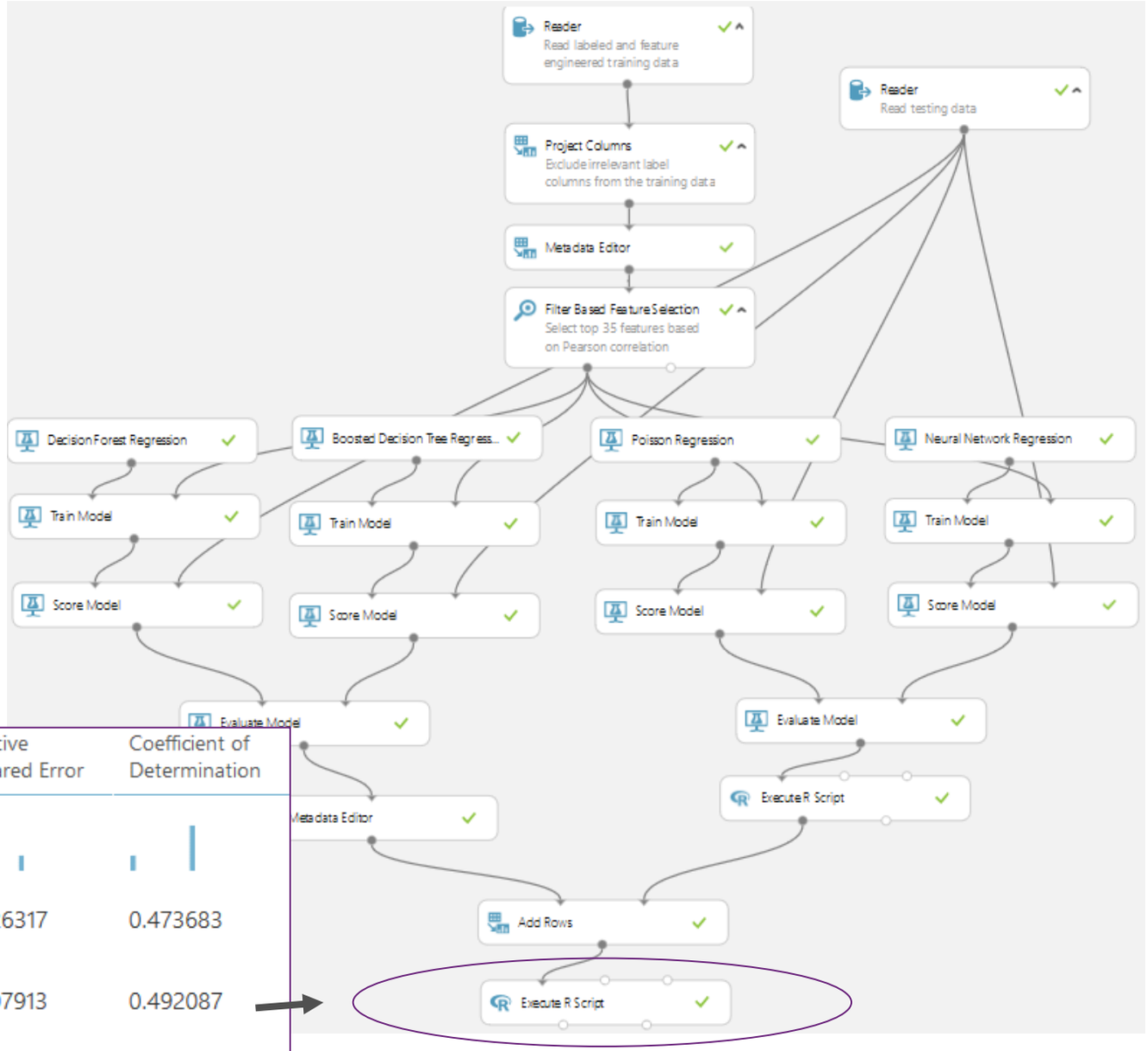
Multi-class classification models

- Multiclass Logistic Regression
- Multiclass Neural Network
- Ordinal regression on two-Class Logistic Regression
- Ordinal regression on two-Class Neural Network



Regression Models

- Decision Forest Regression
- Boosted Decision Tree Regression
- Poisson Regression
- Neural Network Regression



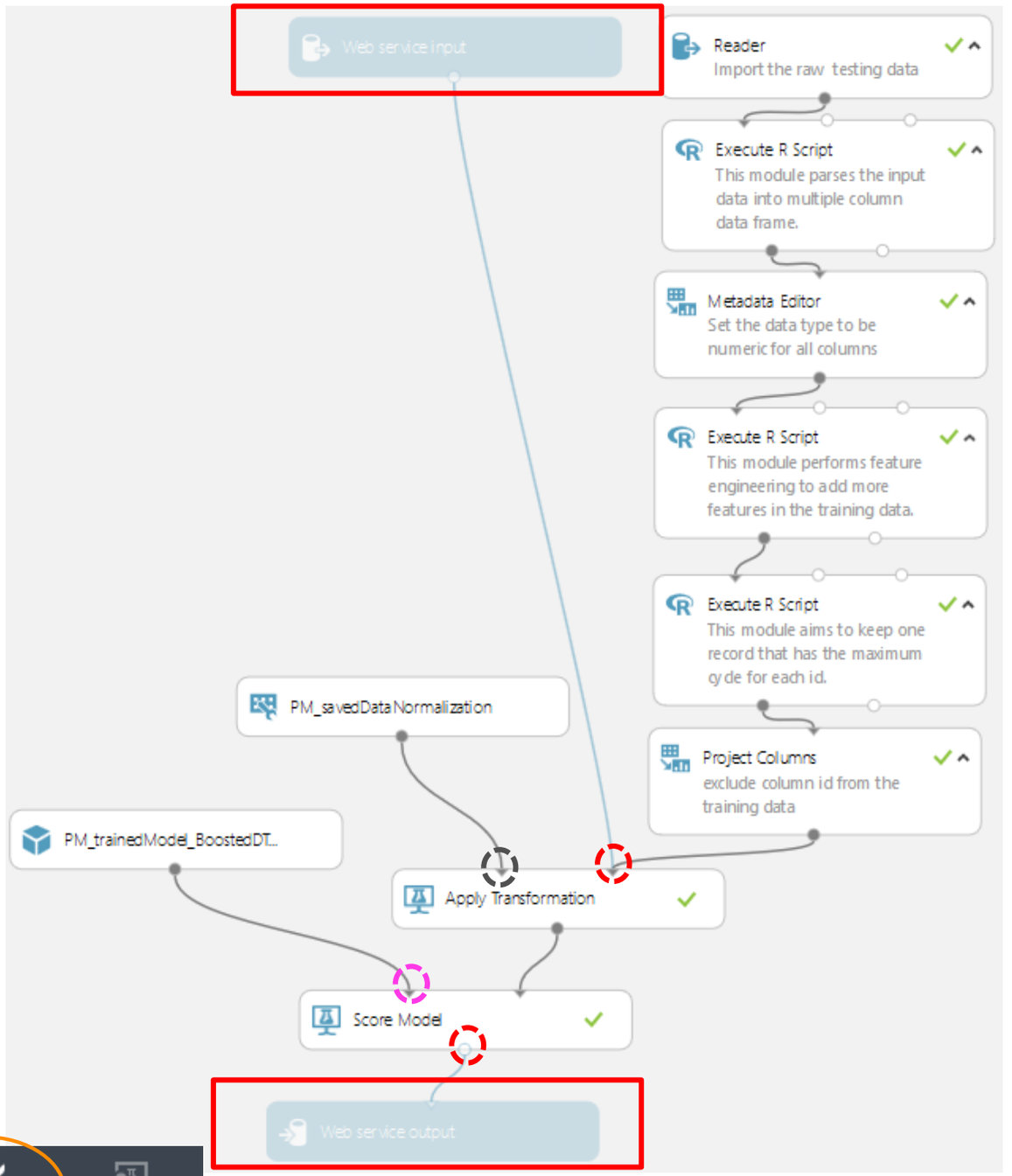
Algorithms	Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Decision Forest Regression	462.724302192819	21.141062	30.147662	0.574998	0.526317	0.473683
Boosted Decision Tree Regression	Infinity	21.283579	29.615866	0.578874	0.507913	0.492087
Poisson Regression	NA	23.243022	29.973356	0.632167	0.520249	0.479751
Neural Network Regression	NA	75.505023	86.185055	2.053597	4.301346	-3.301346

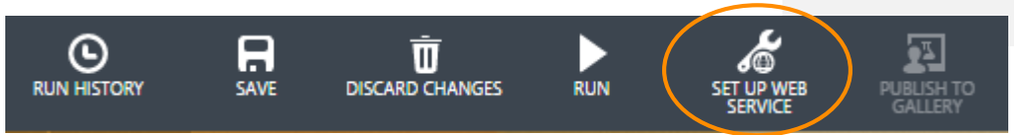
Scoring Experiment (Deploy as web service)

Web service input/output

Trained model

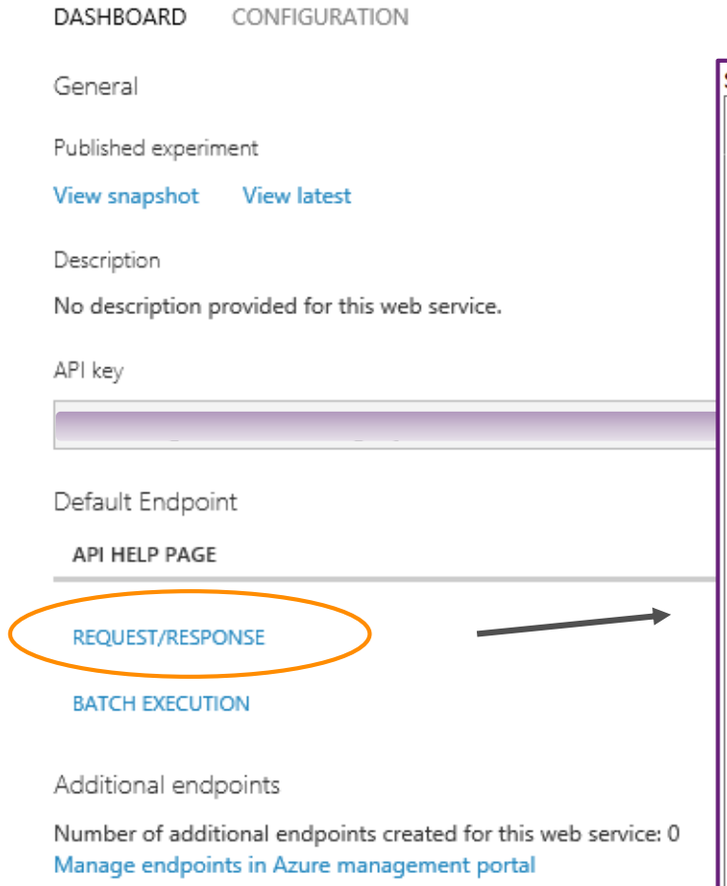
Saved Transform





Scoring Experiment (Deploy as web service)

predictive maintenance: step 3a of 3, deploy web service with a regression model



Sample Code

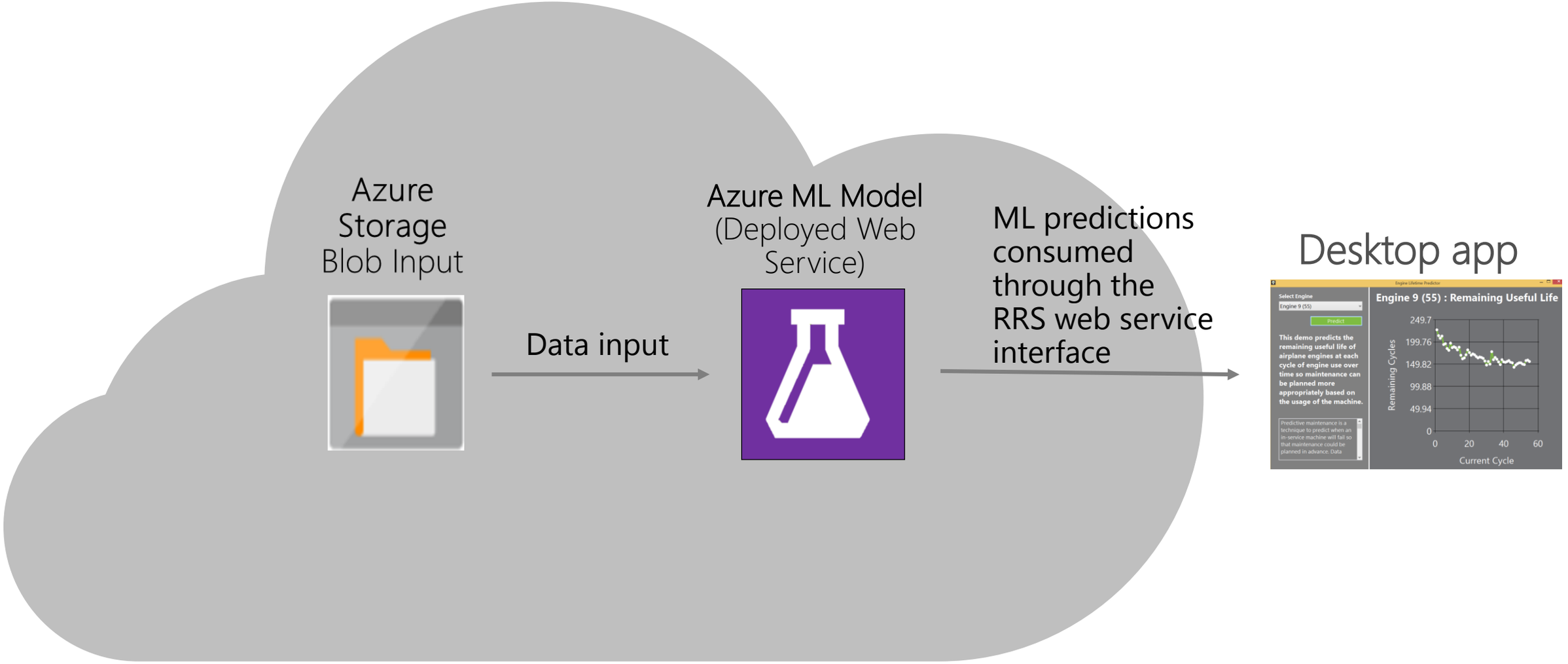
C# | Python | R

```
{
        var scoreRequest = new
        {

            Inputs = new Dictionary<string, StringTable> () {
                {
                    "data",
                    new StringTable()
                    {
                        ColumnNames = new string[] {"p", "size", "prob", "side
                        Values = new string[,] {  { "0.4", "10", "0.5", "L" },
                    }
                },
                            },
                        GlobalParameters = new Dictionary<string, string>(
}
        };
        const string apiKey = "abc123"; // Replace this with the API key for t
        client.DefaultRequestHeaders.Authorization = new AuthenticationHeaderV

        client.BaseAddress = new Uri("https://ussouthcentral.services.azureml.

        // WARNING: The 'await' statement below can result in a deadlock if yo
        // One way to address this would be to call ConfigureAwait(false) so t
        // For instance, replace code such as:
        //      result = await DoSomeTask()
        // with the following:
        //      result = await DoSomeTask().ConfigureAwait(false)
```

Demo: When an in-service engine will fail?

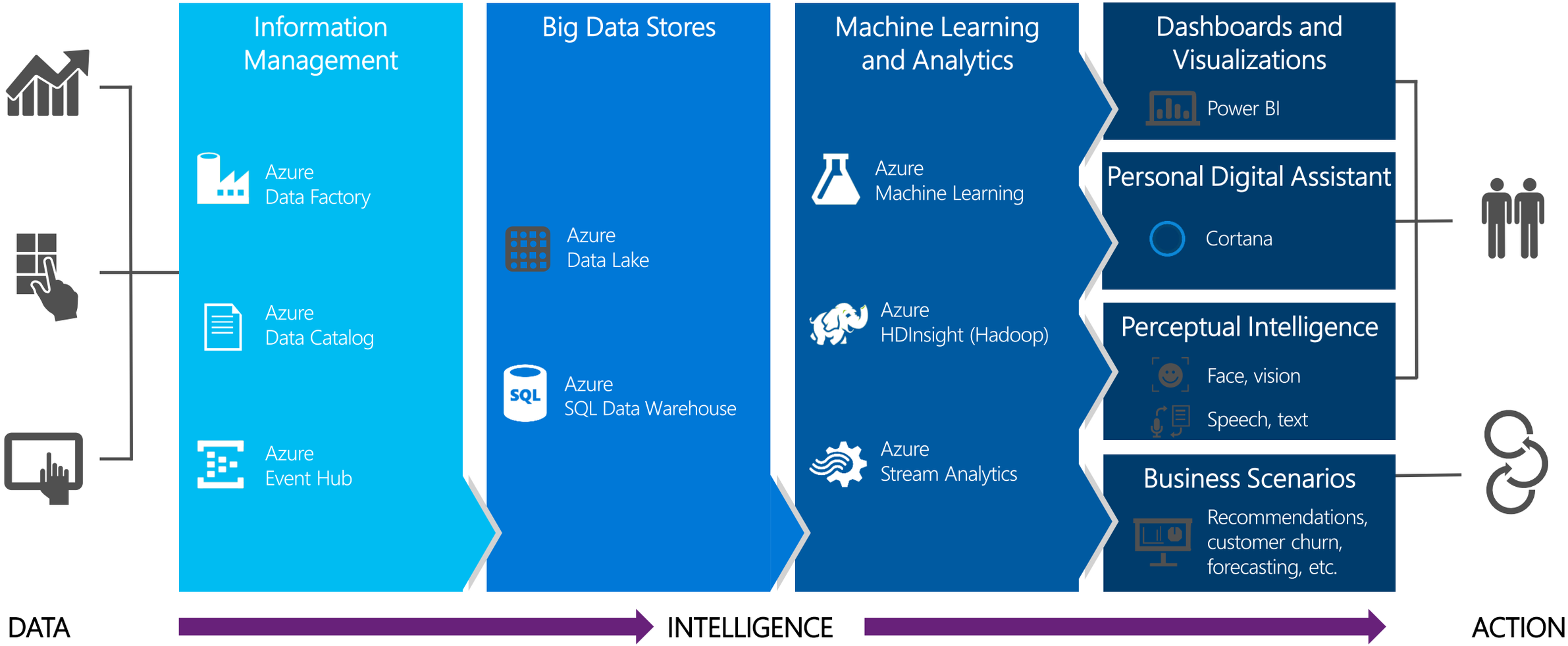
Invoke the web service deployed from Step 3A (regression model) by a desktop app showing how to predict RUL

Demo - Data Pipeline Architecture



Cortana Analytics Suite - Revisit

Transform data into intelligent action



Go Dos

- Microsoft Azure Machine Learning http://azure.com/ml
- http://gallery.azureml.net (search “predictive maintenance”)
- Key words “Cortana Analytics Suite”

© 2015 Microsoft Corporation. All rights reserved.