AZURE ML: FLIGHT DELAY MODEL

Jess Chang 張傳忠
jechang@microsoft.com
Architect
Microsoft Technology Center
2014.12

Overview

- Data Insights
  - Inline visualizer
  - Descriptive statistics
- Data Scrubbing
  - Set missing values
  - Scrub missing values
  - Project subset of columns
  - Edit metadata
- Data Transformation (Part one)
  - Apply math operations
  - Remove duplicate rows
  - Join two datasets
  - Split data
- Data Transformation (Part two)
  - Transform data using R script

Windows Azure
Machine Learning

Sample Application – Predicting Flight Delay

- Problem
  - Approx. 20% of flights are delayed or cancelled every year.
  - Many factors affect delays: weather, mechanical issues, air traffic control, etc.
- Goal
  - Leverage past flight and weather data to predict future flight delays.
  - Reference: *Predicting Flight Delays*, Dieterich Lawson and William Castillo, Stanford University
- Data sources
  1. Airline On-time Performance dataset
     - Source: Bureau of Transportation Statistics (BTS)
  2. Weather Observations (ISD-Lite) dataset
     - Source: National Oceanic and Atmospheric Administration (NOAA)
     - Dataset: Quality Controlled Local Climatological Data (FTP)

A sample from July to October 2013 will be used in this exercise.

Windows Azure Machine Learning

Flight On-Time Performance Dataset

Year	Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled
2013	7	16	2	DL	13487	14747	2155	-1	0	2336	5	0	0
2013	7	16	2	DL	12889	13487	1555	-6	0	2057	-7	0	0
2013	7	16	[illegible]	DL	11278	10397	1600	-5	[illegible]	1752	-19	0	0
2013	[illegible]	[illegible]	[illegible]	DL	13851	[illegible]	600	-3	[illegible]	904	4	0	0
2013	[illegible]	[illegible]	[illegible]	DL	14747	[illegible]	2330	49	[illegible]	736	40	[illegible]	0
2013	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1735	-4	[illegible]	2108	-41	[illegible]	0
2013	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1656	33	[illegible]	1831	17	[illegible]	0
2013	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	659	-7	[illegible]	837	-28	[illegible]	0
2013	7	16	2	[illegible]	[illegible]	[illegible]	805	-2	[illegible]	859	-25	[illegible]	0
2013	7	16	2	[illegible]	[illegible]	[illegible]	1005	-6	[illegible]	1650	-8	[illegible]	0
2013	7	16	2	[illegible]	12953	10397	700	112	[illegible]	930	108	[illegible]	0
2013	7	16	2	DL	11433	12953	1725	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2013	7	[illegible]	[illegible]	[illegible]	13495	12892	720	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2013	7	[illegible]	[illegible]	[illegible]	12889	13487	720	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2013	7	[illegible]	[illegible]	[illegible]	13487	12889	1750	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2013	7	[illegible]	[illegible]	[illegible]	12889	12892	620	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2013	7	[illegible]	[illegible]	[illegible]	12889	10397	715	-7	0	1415	-1	0	0
2013	7	[illegible]	[illegible]	[illegible]	10397	12892	940	-2	0	1110	-11	0	0
2013	7	[illegible]	[illegible]	[illegible]	15304	10397	1445	-5	0	1615	-5	0	0
2013	7	16	2	DL	14869	14747	2155	-5	0	2300	31	1	0
2013	7	16	2	DL	15304	12892	1930	-8	0	2125	-17	0	0
2013	7	16	2	DL	13487	12892	1135	2	0	1321	-17	0	0
2013	7	16	2	DL	12892	12173	1442	4	0	1723	2	0	0
2013	7	16	2	DL	11433	10529	720	-4	0	900	-9	0	0
2013	7	16	2	DL	10529	11433	941	-4	0	1129	-12	0	0
2013	7	16	2	DL	10397	14100	1117	-4	0	1320	-5	0	0
2013	7	16	2	DL	14100	10397	1415	-5	0	1616	-12	0	0
2013	7	16	2	DL	14869	13487	2016	6	0	2346	0	0	0

Carrier is a categorical field

Hour and minutes concatenated in one field

OriginAirportID and DestAirportID are categorical fields with numeric values

Multiple target leaks: DepDelay, DelDel15, ArrDelay, Cancelled

Assume ArrDel15 is the target to be predicted

Sample flight data

Windows Azure Machine Learning

NOAA Weather Observations Dataset

Year	Month	Day	AirportID	Time	TimeZone	SkyCondition	Visibility	WeatherType	DryBulbFarenheit	DryBulbCelsius	WetBulbFarenheit	WetBulbCelsius	DewPointFarenheit	DewPointCelsius	RelativeHumidity
2013	7	1	14843	56	-4	FEW020 SCT035	10		78	25.6	75	23.6	73	22.8	85
2013	7	1	14843	156	-4	FEW035	10		78	25.6	74	23.2	72	22.2	82
2013	7	1	[illegible]	[illegible]	-4	FEW050	10		78	25.6	75	23.6	73	22.8	85
2013	7	1	[illegible]	[illegible]	-4	[illegible]	10		78	25.6	75	23.6	73	22.8	85
2013	7	1	[illegible]	[illegible]	-4	[illegible]	10		77	25	74	23.4	73	22.8	88
2013	7	1	[illegible]	[illegible]	-4	[illegible]	10		78	25.6	75	23.6	73	22.8	85
[illegible]	[illegible]	[illegible]	[illegible]	656	-4	[illegible]	10		78	25.6	75	24	74	23.3	88
[illegible]	[illegible]	[illegible]	[illegible]	756	-4	[illegible]	10		81	27.2	77	24.8	75	23.9	82
[illegible]	[illegible]	[illegible]	[illegible]	856	-4	[illegible]	9		85	29.4	77	25.1	74	23.3	70
[illegible]	[illegible]	[illegible]	[illegible]	956	-4	[illegible]	9		86	30	78	25.6	75	23.9	70
2013	7	1	[illegible]	1056	-4	[illegible]	9		87	[illegible]	78	25.8	75	23.9	68
2013	7	1	[illegible]	1156	-4	[illegible]	[illegible]	TS	84	[illegible]	77	25	74	23.3	72
2013	7	1	[illegible]	1215	-4	[illegible]	[illegible]		84	29	76	24.6	73	23	70
2013	7	1	[illegible]	1256	-4	[illegible]	[illegible]		81	27.2	[illegible]	24.8	75	23.9	82
2013	7	1	[illegible]	1356	-4	[illegible]	[illegible]		83	28.3	[illegible]	[illegible]	73	22.8	72
2013	7	1	[illegible]	1456	-4	[illegible]	[illegible]		84	28.9	[illegible]	[illegible]	74	23.3	72
2013	7	1	[illegible]	1556	-4	[illegible]	[illegible]		83	28.3	[illegible]	[illegible]	73	22.8	72
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-4	FEW042 BKN100	9		83	28.3	[illegible]	[illegible]	74	23.3	74
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-4	FEW055 BKN100	9		83	28.3	[illegible]	[illegible]	74	23.3	74
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-4	V038 SCT075 BKN	9		82	27.8	76	24.3	73	22.8	74
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-4	FEW055	10		81	27.2	75	24.1	73	22.8	77
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-4	FEW045 SCT065	10		81	27.2	75	24.1	73	22.8	77
2013	7	1	14843	2156	-4	FEW025 SCT055	10		80	26.7	75	23.9	73	22.8	79
2013	7	1	14843	2256	-4	FEW025 SCT055	10		80	26.7	76	24.3	74	23.3	82
2013	7	1	14843	2356	-4	FEW025 SCT060	10		79	26.1	76	24.1	74	23.3	85
2013	7	2	14843	56	-4	FEW022 BKN070	10		79	26.1	76	24.1	74	23.3	85
2013	7	2	14843	156	-4	FEW040	10		79	26.1	75	23.8	73	22.8	82
2013	7	2	14843	256	-4	FEW040	10	=-RA	78	25.6	75	23.6	73	22.8	85
2013	7	2	14843	356	-4	FEW022 SCT040	10		78	25.6	75	24	74	23.3	88

Time is in UTC, not local time

Time zone adjustment, relative to UTC

Fields with many missing values

Hour and minutes concatenated in one field

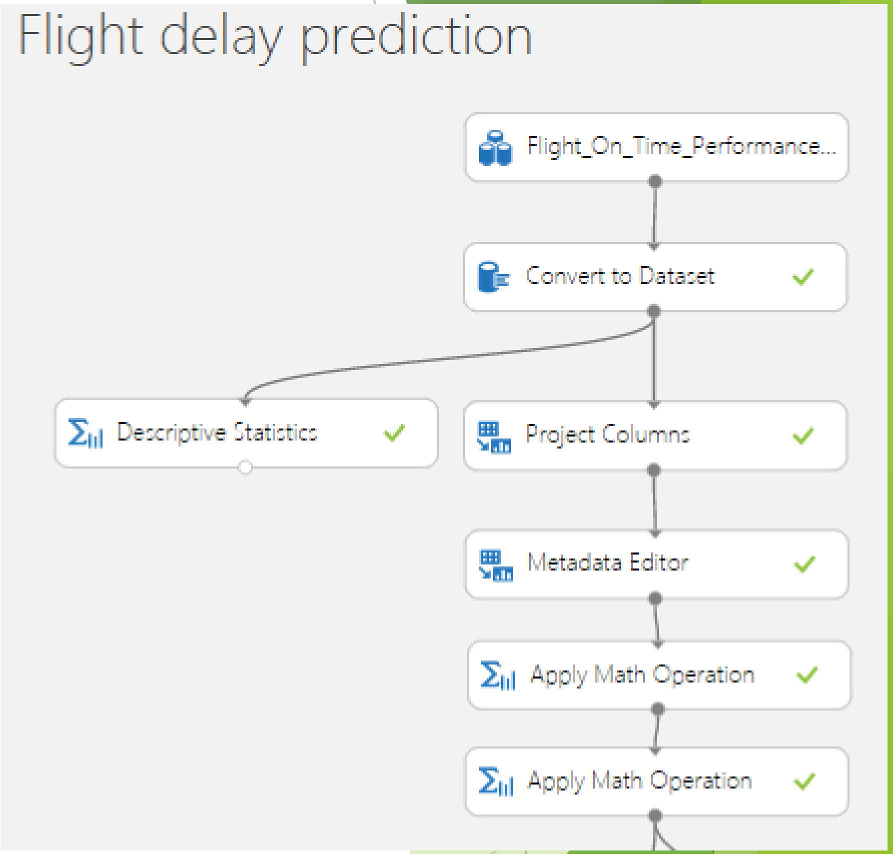
Sample weather data

Prepare Data Transformation Experiment

- Download the two sample datasets for July-October 2013 locally
  - Flight On-Time Performance
  - NOAA Weather Observations
- Upload the datasets to Azure ML
  1. Go to https://studio.azureml.net
  2. Click on **+New** to create a new Dataset
  3. Click on **Dataset** → **From Local File**
  4. Upload the Flight On-Time Performance local file.
  5. Repeat to upload the NOAA Weather Observations local file.
- Start new experiment
  1. Go to https://studio.azureml.net → Click on **Experiments**
  2. Click on **+New** →**Experiment** to create a new experiment
  3. Double click on "**Untitled**" on top of the screen and type in "**Flight delay prediction**"
  4. Expand **Saved Datasets** in the left panel.
  5. Drag the dataset **Flight_On_Time_Performance_July_October_2013.csv** to the experiment.
  6. Drag the dataset **NOAA_Weather_July_October_2013.csv** to the experiment.

Process the Flight On time Performance Dataset

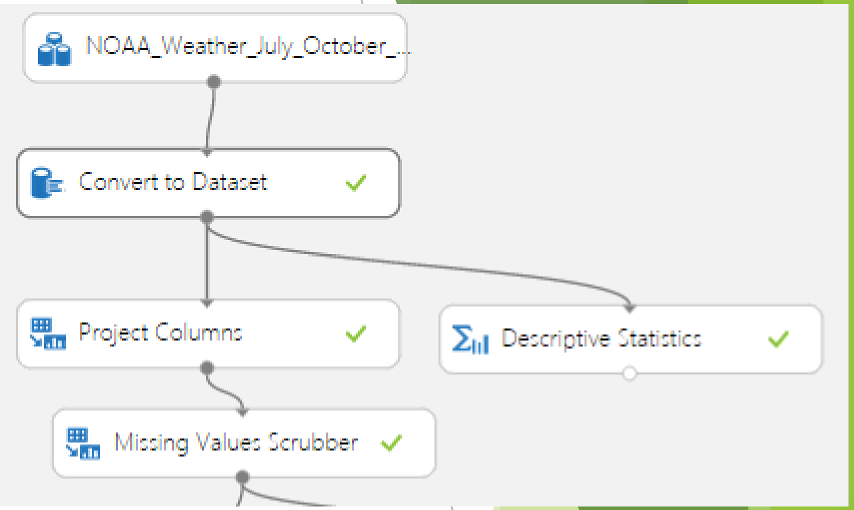
1. Ingest and set missing values to '?'
   - Module: **Convert to Dataset**
   - **SetMissingValues** → ?
2. Explore the data
   - Click output node → **Visualize**
   - Module: **Descriptive Statistics**
3. Remove target leak fields from dataset
   - Module: **Project Columns**
   - **Begin With All Columns**
   - **Exclude columns** [DepDelay, DepDel15, ArrDelay, Cancelled]
4. Edit metadata to indicate **categorical** fields
   - Module: **Metadata Editor**
   - **Begin With NO Columns**
   - **Include** columns [Carrier, OriginAirportID, DestAirportID] to **Categorical**
5. Extract *hour* from time fields [CRSDepTime and CRSArrTime]
   - Module: **Apply Math Operation**
   - Step 1: **Operations** → **Divide** *time* field by 100 → **Begin With NO Columns** → **Include** columns [CRSDepTime and CRSArrTime]
   - Module: **Apply Math Operation**
   - Step 2: **Rounding** → **Floor** operation → **Begin With NO Columns** → **Include** columns [CRSDepTime and CRSArrTime]
   - Output mode: **Inplace**



Windows Azure Machine Learning

Process the NOAA Weather Observations Dataset

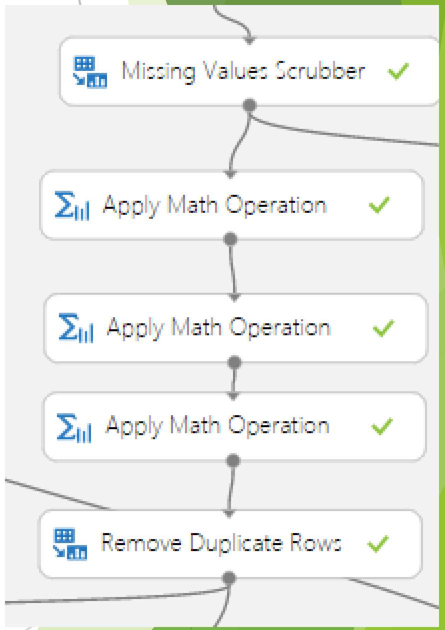
1. Ingest and set missing values to ‘M’
   - Module: **Convert to Dataset**
   - **SetMissingValues** → M
2. Explore the data
   - Click output node → **Visualize**
   - Module: **Descriptive Statistics**
3. Remove unnecessary fields from dataset
   - Module: **Project Columns**
   - **Begin With All Columns**
   - **Exclude columns names** [WetBulbFarenheit, WetBulbCelsius, ValueForWindCharacter , StationPressure, PressureTendency, PressureChange, SeaLevelPressure]
   - **Exclude columns types** [*String*]
4. Remove rows with missing values
   - Module: **Missing Values Scrubber**
   - For missing values → **Remove entire row**
   - Cols with all MV → **KeepColumns**
   - MV indicator column → **DoNotGenerate**



Windows Azure Machine Learning

Process the NOAA Weather Observations Dataset (ctd.)

1. Extract *hour* from time field [Time]
   - Module: **Apply Math Operation**
   - Step 1: **Operations** → **Divide** *time* field by 100 → **Begin With NO Columns → Include** columns [Time]
   - Module: **Apply Math Operation**
   - Step 2: **Rounding** → **Ceiling** operation → **Begin With NO Columns → Include** columns [Time]
   - Output mode: **Inplace**
2. Adjust time from UTC to local time using TimeZone field
   - Module: **Apply Math Operation**
   - **Operations** → **Subtract** → **ColumnSet**
   - Operation argument → **Begin With NO Columns → Select column** [TimeZone]
   - Column set → **Begin With NO Columns → Select column** [Time]
   - Output mode: **Append**



- **Save Experiment and Run**

1. Remove duplicate rows to keep one weather observation per hour
   - Module: **Remove Duplicate Rows**
   - Key column selection filter expression → **Begin With NO Columns → Select columns** [Year, Month, Day, AirportID, Subtract(Time_TimeZone)]

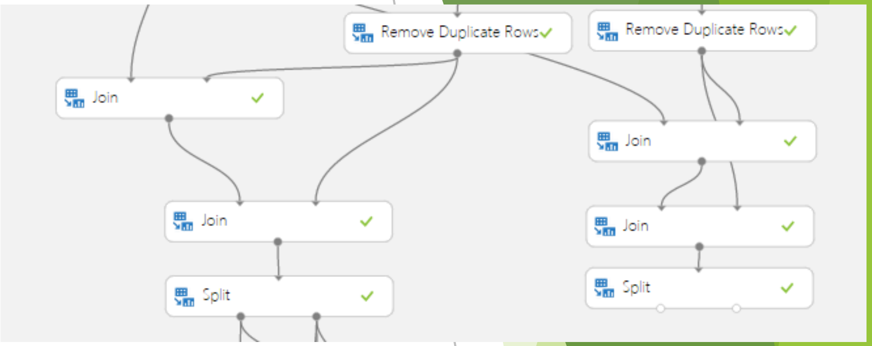
Windows Azure Machine Learning

Join Flight and Weather Datasets + Split

1. Join the two datasets to add weather information at departure airport

   Begin With NO Columns

   - Module: **Join**
   - Join key columns for L → **Select columns** [Year, Month, DayOfMonth, **OriginAirportID**, CRSDepTime]
   - Join key columns for R → **Select columns** [Year, Month, Day, AirportID, Subtract(Time_TimeZone)]
   - Join type → **Inner Join**
   - **Uncheck** "Keep right key columns in joined table"



1. Join the two datasets to add weather information at arrival airport
   - Repeat (1) with the following left key columns
   - Join key columns for L → **Select columns** [Year, Month, DayOfMonth, **DestAirportID**, CRSDepTime]
2. Split the final data to separate October 2013 data
   - Module: **Split**
   - Splitting mode → **Relative Expression**
   - Relational expression → Enter the value **"Month" < 10**

Windows Azure Machine Learning

Transform Data Using an R Script

- Motivation
  - Perform multiple transformations in one step
  - Complement Azure ML Studio with additional operations, functions, packages, etc.
- Exercise: Replace NOAA Weather Observations data processing with an R script
  - Extract *hour* from the Time field
  - Adjust weather time from UTC to local time using DateTime operations
  - Append the adjusted Month/Day/Hour fields to input dataset
  - Project required output columns and re-order them
  - R script code in next slide
- Run the R script code in Azure ML Studio
  - Module: **Execute R Script**
  - Copy the R script from your favorite editor (e.g., Rstudio)
  - Paste the code in the designated *Script* area in the module Properties panel
  - Random Seed: 42

❖ **Save Experiment and Run**

Windows Azure Machine Learning

Weather Data Transformation R Script

```
# Map input port to variable
dataset <- maml.mapInputPort(1) # class: data.frame

Year = dataset$Year
Month = dataset$Month
Day = dataset$Day
Time = dataset$Time
Hour = ceiling(Time / 100)
Timezone = dataset$TimeZone

# Number of rows to process
n = nrow(dataset)

# Concatenate date fields and apply time zone difference
fulldate = lapply(1:n, function(i) as.POSIXlt(sprintf("%4d-%02d-%02d %02d:00:00", Year[i], Month[i], Day[i], Hour[i]), tz = "UTC") - Timezone[i] * 3600)
adjustdate = do.call(c,fulldate)

# Extract the adjusted month, day, and hour - Should adjust year too for general case
AdjustedMonth = as.POSIXlt(adjustdate)$mon + 1
AdjustedDay   = as.POSIXlt(adjustdate)$mday
AdjustedHour  = as.POSIXlt(adjustdate)$hour

#  Extract other columns
AirportID = dataset$AirportID
Weather = dataset[,7:14]

# Construct output data frame and send to the output Dataset port
data.set = cbind(Year, AdjustedMonth, AdjustedDay, AirportID, AdjustedHour, Weather)
maml.mapOutputPort("data.set");
```

Windows Azure Machine Learning

Join Flight and Weather (R executed) Datasets + Split

- Remove duplicate rows to keep one weather observation per hour

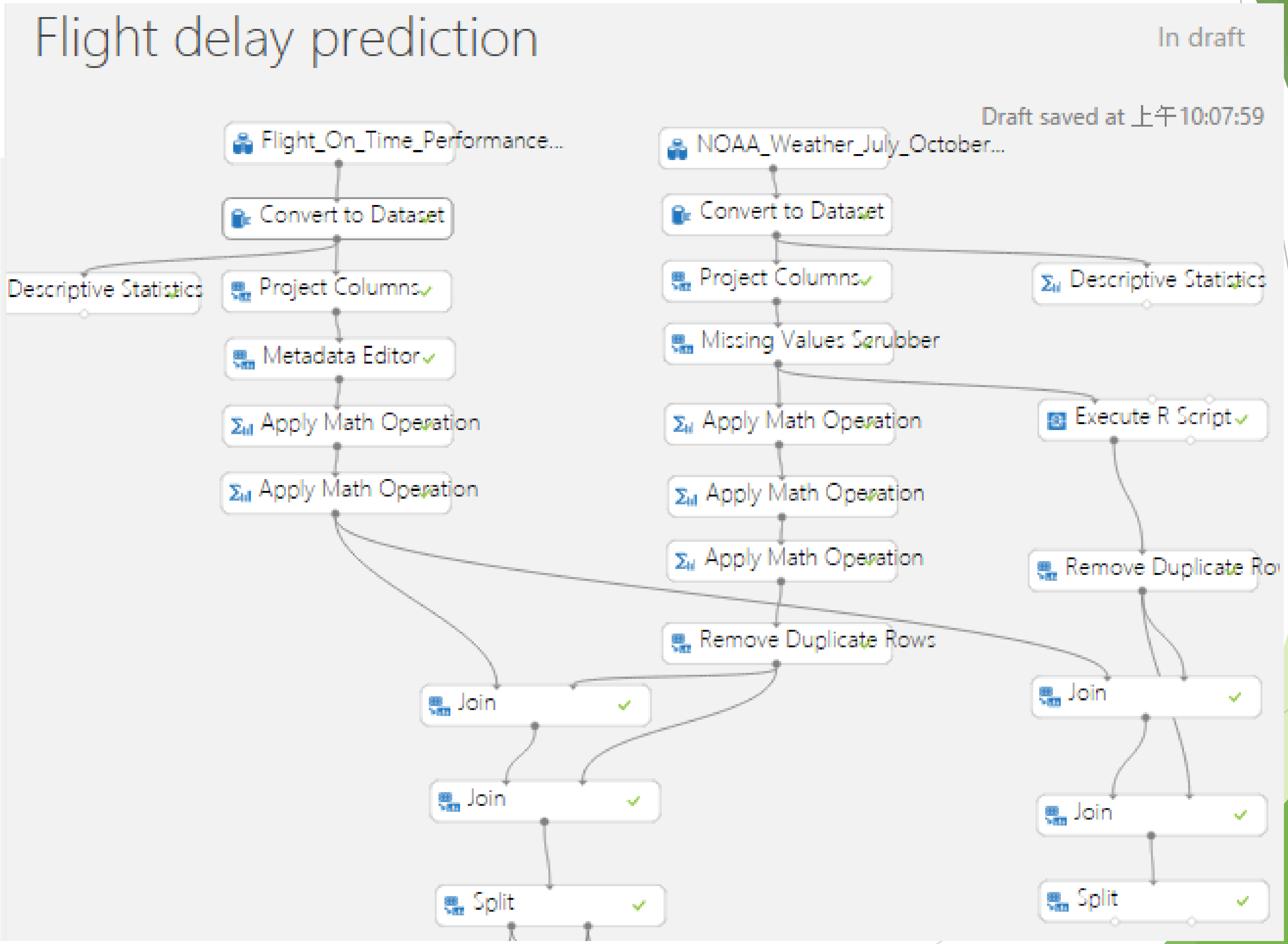
  Begin With NO Columns

  - Module: **Remove Duplicate Rows**
  - Key column selection filter expression → **Select columns** [Year, AdjustedMonth, AdjustedDay, AirportID, AdjustedHour]

1. Join the two datasets to add weather information at departure airport
   - Module: **Join**
   - Join key columns for L → **Select columns** [Year, Month, DayOfMonth, OriginAirportID, CRSDepTime]
   - Join key columns for R → **Select columns** [Year, AdjustedMonth, AdjustedDay, AirportID, AdjustedHour]
   - Join type → **Inner Join**
   - **Uncheck** "Keep right key columns in joined table"
2. Join the two datasets to add weather information at arrival airport
   - Repeat (1) with the following left key columns
   - Join key columns for L → **Select columns** [Year, Month, DayOfMonth, DestAirportID, CRSDepTime]
3. Split the final data to separate October 2013 data
   - Module: **Split**
   - Splitting mode → **Relative Expression**
   - Relational expression → Enter the value **"Month" < 10**

Windows Azure Machine Learning

Data Transformation Experiment



Windows Azure Machine Learning

Two-Class Boosted Decision Tree

1. Adopt Two-Class Boosted Decision Tree Module
   - Maximum number of leaves per tree: **20**
   - Minimum number of samples per leaf node: **10**
   - Learning rate: **0.2**
   - Number of trees constructed: **100**
   - Allow unknown categorical levels
2. Optimize parameter settings: [Random sweep]
   - Module: **Sweep Parameters**
   - Maximum number of runs on random sweep: **10**
   - Label column: Include Column names: **ArrDel15**
   - Metric for measuring performance for classification: **Accuracy**
   - Metric for measuring performance for regression: **Mean absolute error**
   - Input Untrained Model(Left most): **Output of Two-Class Boosted Decision Tree**
   - Input Training Dataset(Central): **Output of Split dataset1**
   - Input Validation Dataset(Right most): **Output of Split dataset2**
3. Score a trained classification regression model
   - Module: **Score Model**
   - Input Trained Model(Left): **Output of Sweep parameters Trained best model(Right)**
   - Input Dataset(Right): **Output of Split dataset2**

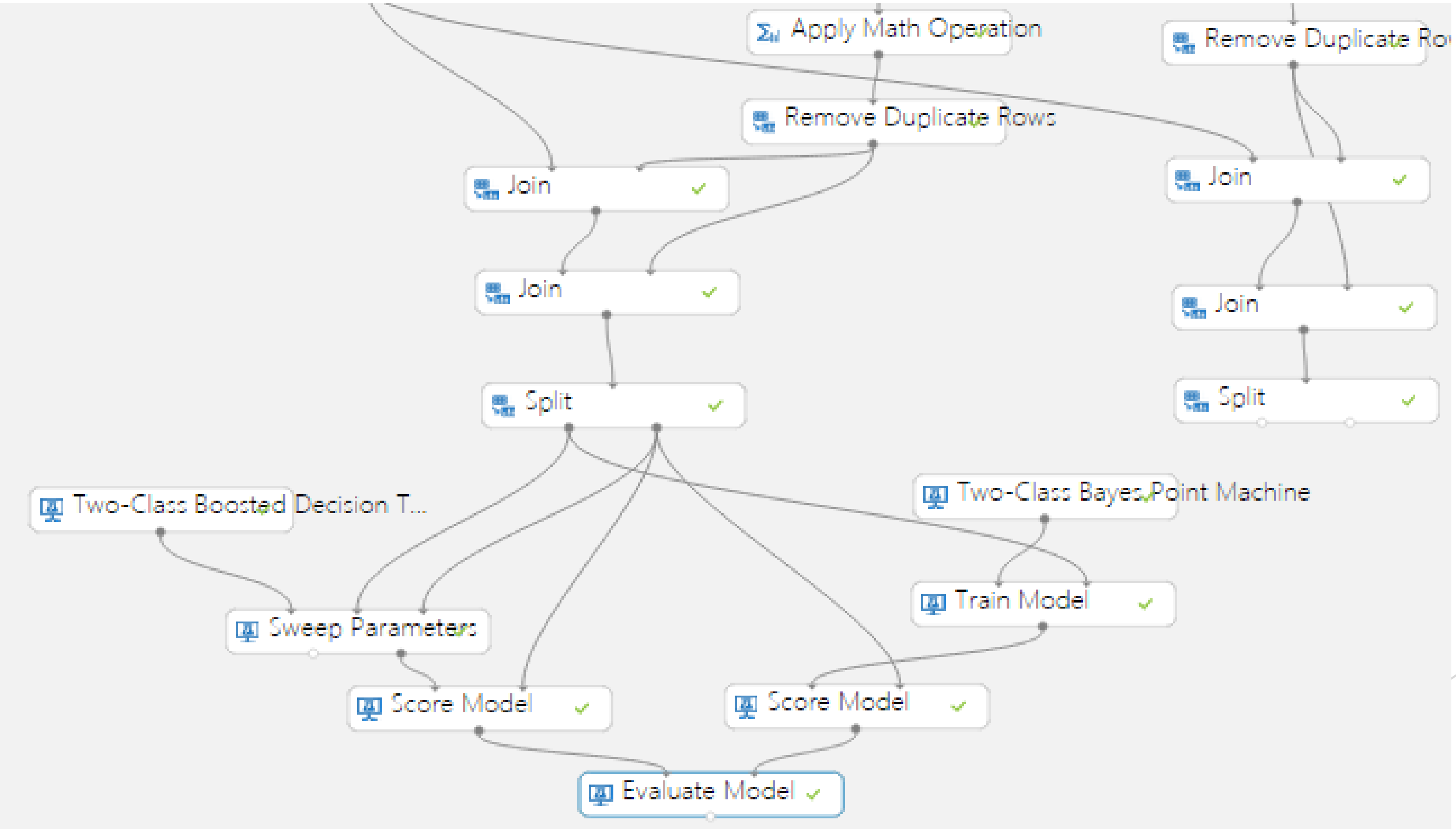
Two-Class Bayes Point Machine

1. Adopt Two-Class Bayes Point Machine Module
   - Number of training iterations: **10**
   - Include bias
   - Allow unknown values in categorical feature
2. Train a previously created classification or regression model
   - Module: **Train Model**
   - Include Label column: **[ArrDel15]**
   - Input Untrained Model(Left): **Output of** Two-Class Bayes Point Machine
   - Input Dataset(Right): **Output of Split dataset1**
3. Score a trained classification regression model
   - Module: **Score Model**
   - Input Trained Model(Left): **Output of Train Model**
   - Input Dataset(Right): **Output of Split dataset2**

Windows Azure
Machine Learning

Evaluate a scored classification regression model

- Module: **Evaluate Model**
- Input scored dataset1: **Score Model of Two-Class Boosted Decision Tree**
- Input scored dataset2 to compare: **Score Model of Two-Class Boosted Decision Tree**



Windows Azure
Machine Learning

AZURE ML:
FLIGHT DELAY MODEL