Predictive Model on Arrival Time of Flights

Content

1. **Project Overview**: Describe the Project with summary of analytical processes and project outcomes (Explain the Project in your own words in 15 – 20 lines)

For this project, R and Azure Machine Learning studio will be used to train, evaluate, and publish a Linear Regression Model. We are using Linear Regression Model to predict the arrival time of Flights. The objective is to construct a predicting Machine Learning Model at the end of the workflow, to predict flight arrival delays (in minutes).

The flight dataset is explored to identify features that might be predictive of how many minutes late or early a flight will be. The dataset contains information about flights that have landed such as flights were on-time, early, or late.

According to a 2013 report made by the US Federal Aviation Administration, the economic price of domestic flight delays entails a yearly cost of 32.9 billion dollars to passengers, airlines, and other parts of the economy. More than half of that amount comes from the wallets of passengers lost time waiting for their planes to leave due to delays, but they also miss connecting flights, spend money on food and have to sleep on hotel rooms while they're stranded.

This project aims at getting an insight to identify features that might be predictive of how many minutes late or early a flight will be.

A major factor in increased airspace efficiency and capacity is accurate prediction of Estimated Time of Arrival (ETA) for commercial flights, which can be a challenging task due to a non-deterministic nature of environmental factors, and air traffic. Inaccurate prediction of ETA can cause potential safety risks such as having misarranged arrival schedule and may cause potential collision of aircrafts. In addition, inaccurate prediction can also lead to loss of resources for Air Navigation Service Providers (ANSP), airlines and passengers.

2. Project Technical Environment:

This Project utilizes the Microsoft Azure Machine Learning (AML) platform to perform data processing and analyses.

1. An account with Microsoft Azure ML portal and
2. Installed R and R Studio on my computer

A free account on Microsoft Machine Learning Studio (classic) is registered and used. It provides a range of cloud services, including those for compute, analytics, storage, and networking. Users can pick and choose from these services to develop and scale new applications or run existing applications in the public cloud.

Microsoft Azure Machine Learning is a collection of services and tools intended to help developers train and deploy machine learning models. Microsoft provides these tools and services through its Azure public cloud.

It is a visual, drag and drop tool designed to help users build and deploy predictive analysis models with no coding required. We can create a model in Azure Machine Learning or use a model built from an open-source platform, such as Python, Tensor Flow, Scikit Learn, or R. Machine Learning Studio is where data science, predictive analytics, cloud resources and your data meet. Microsoft Azure is widely considered both a platform as a Service (PaaS) and an Infrastructure as a Service (IaaS) offering.

3. **Analytical Technique & Tools used:**

1. R Visualizations for plotting Histograms/Scatter Plots used

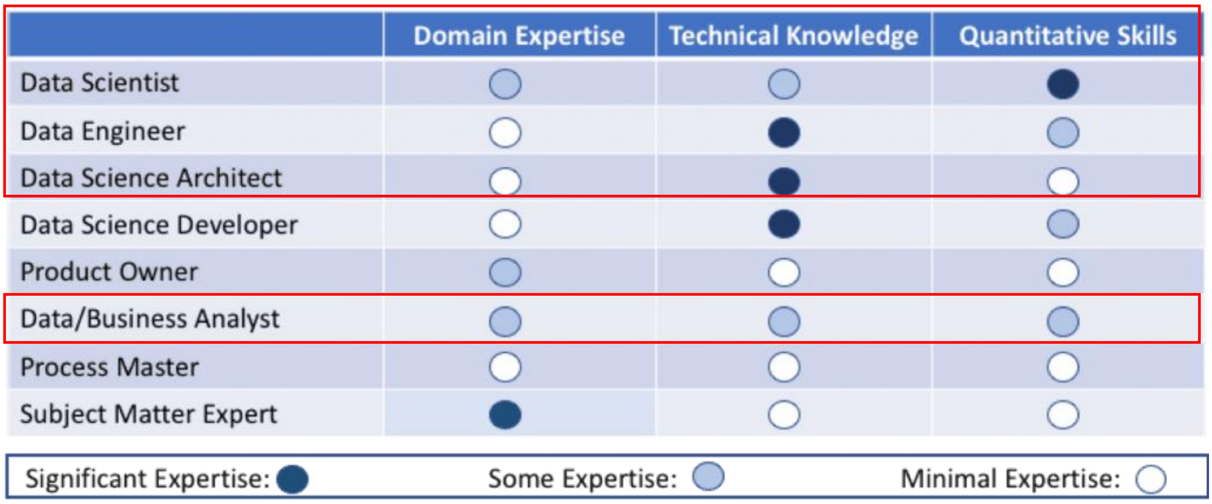
2. R -is used to produce statistical information such as mean, standard deviation, to provide insights and study relationship between variables.

3. for Machine Learning (for prediction), Regression Model, Boosted Decision Tree Regression is used. The technique used for this project is Boosted Decision Tree Regression. Regression is a Supervised Learning Technique. Regression is used in this project as we wish to predict the number of minutes (a numeric value) of the Flight Delay (or Early)

4. Once we have created and refined a model in Azure ML, we publish it as a web service so that client applications can use it to retrieve predicted labels based on feature inputs

5. We use Excel Online to access the URL of the web service we created in this project

4. Data Science Project Team – Roles and Responsibilities Table



	Domain Expertise	Technical Knowledge	Quantitative Skills
Data Scientist	◍	◍	●
Data Engineer	○	●	◍
Data Science Architect	○	●	○
Data Science Developer	○	●	◍
Product Owner	◍	○	○
Data/Business Analyst	◍	◍	◍
Process Master	○	○	○
Subject Matter Expert	●	○	○

Significant Expertise: ● Some Expertise: ◍ Minimal Expertise: ○

The Data Science Team consist of several key roles that work hand in hand to deliver an insightful and informative outcome for their user needs. They consist of Data Scientist, Engineer, Architect, Developers, Analysts. In these projects, various disciplines and skills are needed and there is often a confluence between software and data engineering as well as data analysis.

Data Scientist

Data Scientists find and interpret rich data sources, merge data sources, create visualizations, and use machine learning to build models that aid in creating actionable insight from the data. They know the end-to-end process of data exploration and can present and communicate data insights and findings to a range of team members. In short, they apply the scientific discovery process, including hypothesis testing, to obtain actionable knowledge related to a scientific or business problem.

If you lead a data science team, you need to understand that data scientists might get frustrated if they are managed like software engineers. It's key to understand the difference between data scientists and software engineers and to manage the data scientists in ways that don't alienate them into a different role.

Data Engineer

Data engineers make the appropriate data accessible and available for data science efforts. They design, develop, and code data-focused applications that capture data, as well as clean the data. This role also helps to ensure consistency of datasets (e.g., meaning of attributes across datasets).

Data Science Architect

Data science architects design and maintain the architecture of data science applications and facilities. In other words, this role creates and manages relevant data models, data storage systems and processes workflows. In conjunction with the Data Engineer, they manage and merge large amounts of data and their related sources.

Data Science Developer

Data Science Developers design, develop, and code large data (science) analytics applications to support scientific or enterprise/business processes. This role enables models to be deployed (i.e., use a model in production) and requires some expertise in data science, as well as knowledge of how to effectively develop software applications. Sometimes this role is known as a machine learning engineer. Regardless, they help bridge the worlds of data science and software development.

Data/Business Analyst

Data/Business Analysts analyse a large variety of data to extract information about system, service, or organization performance and present them in usable/actionable form. They better shape a problem for the data scientist to explore. Note the difference between a data analyst and data scientist.

Activity 1: Activities Summary	
Activity 2	Flights Delay Dataset Overview. This sample is available in AML Sample Data. A summary of K-means Clustering is provided running with sample experiments in AML Database.
Activity 3	Login to MS Azure. Load Flights Delay Data and Airport Codes Dataset from AML Sample Data. Flights Delay dataset had 2719418 rows and 14 columns. The mean arrival delay was 6.6377. Some flights Min = -94 -> 94 / 60 = 1.56 Hours Early Arrival Max = 1845 -> 1845/60 = 30.75 Hours (More than a day) of Delay in arriving.
Activity 4	To see the details of airports, join Flights Delay Data and Airport Codes Dataset using inner join with airport code. Has 2719418 rows and 22 columns. Found that Hartsfield-Jackson Atlanta International is the most frequently occurring destination airport.
Activity 5	Remove Duplicate rows and Clean the Missing Values. Rows are considered duplicates in this dataset if they have matching values for all the following fields: **Year, Month, DayofMonth, Carrier , Origin Airport ID, Dest Airport ID, CRS Dep Time, CRS Arr Time** Remove Duplicate module -> 2719418 - 2719397 = **21 duplicate rows are removed** Results dataset from Clean Missing Data module shows missing values for DepDelay and ArrDelay are replaced with 0.

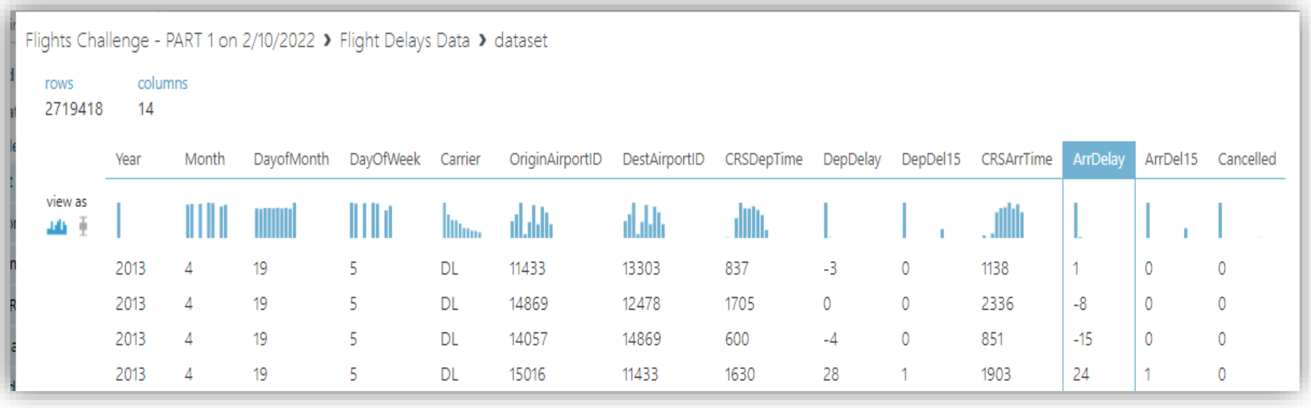
Activity 6	We use the R summary and sd functions to display summary statistics for all columns in the flights data.
	Summary values of ArrDelay are :
	Min = -94, Max = 1845, Mean = 6.5669,
	SD = 38.4481

Activity 7	Range and Distribution of ArrDelay are computed in this activity to understand this value better. Range = 1939, Distribution = 21 .

	We plot Boxplot and Histogram for ArrDelay to understand its distribution.
Activity 8	We plot histograms conditioned by the ArrDel15 column, which is a binary column indicating whether a flight arrived 15 or more minutes late and how it is related to these columns -DepDelay - CRSArrTime - CRSDepTime - DayofMonth - DayOfWeek - Month
Activity 9	We plot conditioned scatter plots for the following columns, conditioned by the ArrDel15 column for values of 0 and 1: - DepDelay - CRSArrTime - CRSDepTime - DayofMonth - DayOfWeek - Month
Activity 10	We create a Machine Learning Model (Regression Model) train the model to predict the arrival delay of flights (delay or early). The columns such as **Month, DayofMonth, DayOfWeek, Carrier, OriginAirportID, DestAirportID, CRSDepTime, DepDelay, CRSArrTime, and ArrDelay are selected.** **The OriginAirportID, DestAirportID and Carrier** features are set to Categorical. Used Normalization to standardize the **CRSDepTime, CRSArrTime and DepDelay using ZScore** transformation method. Split the Data into 70% for Training and 30% for Testing Use Boosted Decision Tree Regression module and Train Model module to train with 70% data to predict ArrDelay column. Use Score Model module to score the trained model using 30% of Test Data. Use Evaluate Model module to evaluate the results from the Score Model.
Activity 11	We run the experiment to test and evaluate the model. The RMSE = 12.78 minutes. COD = 0.8898
Activity 12	Run the experiment and deploy as a web service - Predictive Experiment outputs the Scored Labels column. We use Excel Online to test the web service and generate the predicted values for the given sample data.

Activity 2

Task 1: Login to MS Azure and identify the Flight Data Set which must be used in this Project



Flights Challenge - PART 1 on 2/10/2022 › Flight Delays Data › dataset

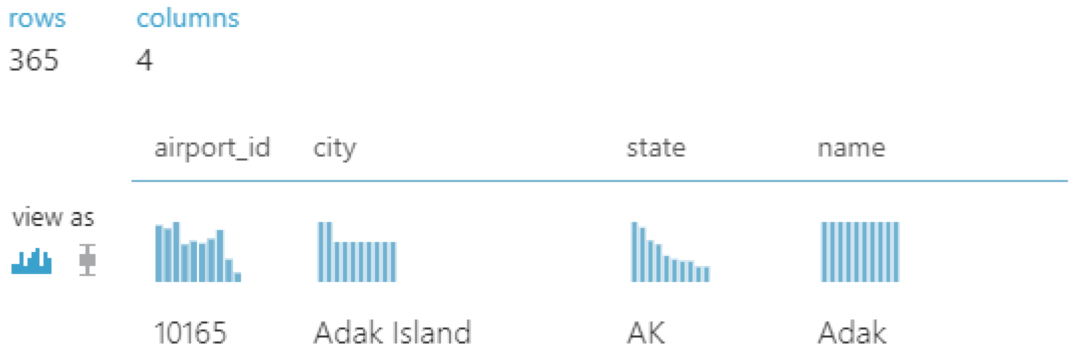
rows	columns
2719418	14

Year	Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled
2013	4	19	5	DL	11433	13303	837	-3	0	1138	1	0	0
2013	4	19	5	DL	14869	12478	1705	0	0	2336	-8	0	0
2013	4	19	5	DL	14057	14869	600	-4	0	851	-15	0	0
2013	4	19	5	DL	15016	11433	1630	28	1	1903	24	1	0

The flight Data set is a dataset inside the MS AML platform.

Data set is then retrieved into the workspace and visualize for Data Exploration. After observation, 2,719,418 rows and 14 attributes are found. Each row consists of a flight performed which shows the air carrier, where it came from and where is the destination, along with timings and delay if there is any.

The Origination and Destination Airport both carried Airport Code. In additional, the data also capture if an airplane arrived late or departs late using the ArrDel15 and DepDel15 to see if it exceeds the 15 minutes threshold.



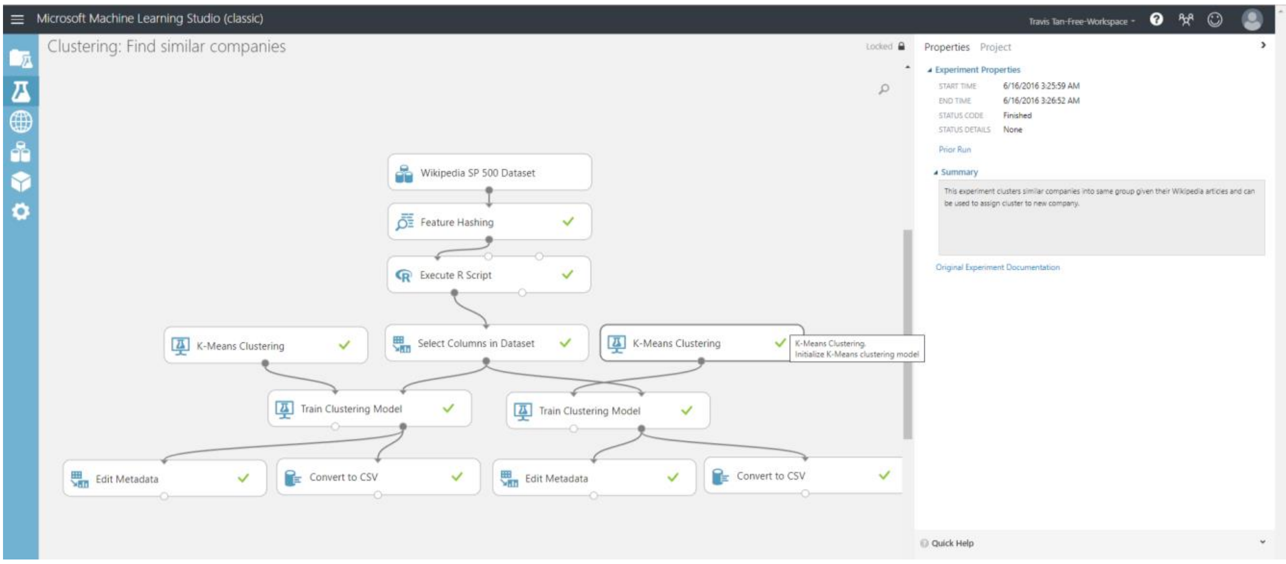
rows	columns
365	4

airport_id	city	state	name
10165	Adak Island	AK	Adak

For the airport code dataset, there are a total of 365 airports and 4 columns of information such as Name of the Airport, the City and State and the unique identification airport code.

Check whether there are any experiments with K-means Clustering or any other Clustering models

K-Means Clustering sample experiment called Find similar companies was read through to understand how K-Means Clustering worked.



Properties Project

Experiment Properties

START TIME 6/16/2016 3:25:59 AM

END TIME 6/16/2016 3:26:52 AM

STATUS CODE Finished

STATUS DETAILS None

Prior Run

Summary

This experiment clusters similar companies into same group given their Wikipedia articles and can be used to assign cluster to new company.

Original Experiment Documentation

Function Prcomp was used in the R Script to perform a principal components analysis on the given data matrix.

First 10 PCA was selected and used to train the clustering model. Two K-Means Clustering with 2 and 3 Centriods was used to create similar clusters of company together.

Introduction to K-Means Algorithm

The K-means clustering algorithm computes centroids and repeats until the optimal centroid is found. It is presumptively known how many clusters there are. It is also known as the flat clustering algorithm. The number of clusters found from data by the method is denoted by the letter ‘K’ in K-means.

K-means algorithm explores for a pre-planned number of clusters in an unlabelled multidimensional dataset, it concludes this via an easy interpretation of how an optimized cluster can be expressed.

Primarily the concept would be in two steps;

Firstly, the cluster centre is the arithmetic mean (AM) of all the data points associated with the cluster.

Secondly, each point is adjoint to its cluster centre in comparison to other cluster centres. These two interpretations are the foundation of the k-means clustering model.

You can take the centre as a data point that outlines the means of the cluster, also it might not possibly be a member of the dataset.

In simple terms, k-means clustering enables us to cluster the data into several groups by detecting the distinct categories of groups in the unlabelled datasets by itself, even without the necessity of training of data.

This is the centroid-based algorithm such that each cluster is connected to a centroid while following the objective to minimize the sum of distances between the data points and their corresponding clusters.

As an input, the algorithm consumes an unlabelled dataset, splits the complete dataset into k-number of clusters, and iterates the process to meet the right clusters, and the value of k should be predetermined.

Specifically performing two tasks, the k-means algorithm

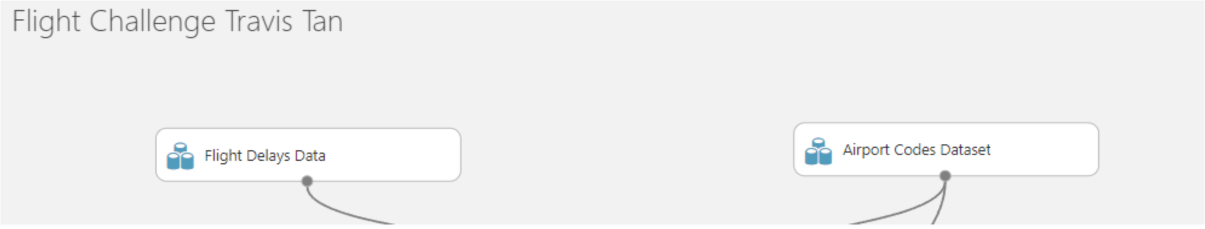
Calculates the correct value of K-centre points or centroids by an iterative method

Assigns every data point to its nearest k-centre, and the data points, closer to a particular k-centre, make a cluster. Therefore, data points, in each cluster, have some similarities and far apart from other clusters.

In this method, data points are assigned to clusters in such a way that the sum of the squared distances between the data points and the centroid is as small as possible. It is essential to note that reduced diversity within clusters leads to more identical data points within the same cluster.

Activity 3

In this Activity, the first task is to create a new Experiment which had been named Flight Challenge. The second task is to load the FlightDelayData.csv and AirportcodeData.csv datafile into the new experiment, to visualize the output for data exploration.



Flight Delays Data has following features:

Year : Year of departure

Month : Month of departure

DayofMonth: Date of departure (1ST to 31st)

DayOfWeek : Day of week of departure (1-5 being Monday to Friday, and 6-7 being Saturday and Sunday respectively)

Carrier : Abbreviation for air carrier

OriginAirportID : Origin airport code

DestAirportID : Destination airport code

CRSDepTime : Scheduled departure time (in local time, format in Hours Hours: Mins Mins) - shown in Computerized Reservations Systems (CRS)

DepDelay : Departure delay in minutes

DepDel15 : Departure delay indicator 0 means departure delayed less than 15 minutes A flight is counted as departure "on time" if it departed less than 15 minutes later the scheduled time shown in the carriers 'Computerized Reservations Systems' (CRS). 1 means departure delayed more than 15 minutes A flight is counted as departure "delayed" if it departed more than 15 minutes later the scheduled time shown in the carriers 'Computerized Reservations Systems' (CRS).

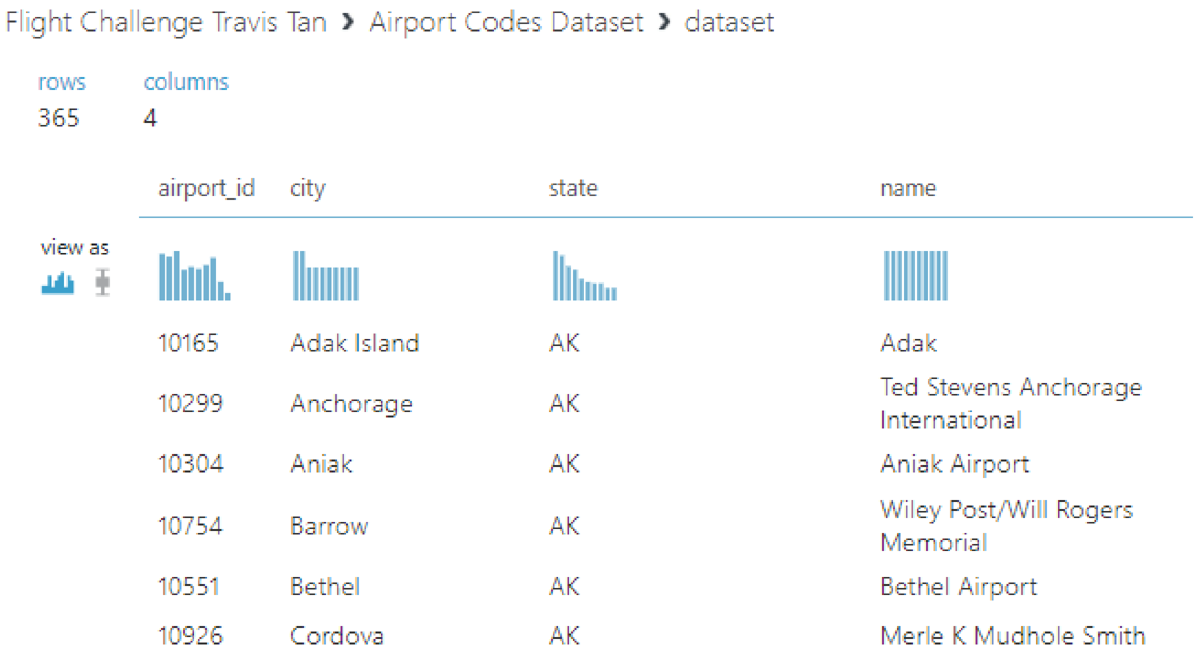
CRSArrTime : Scheduled arrival time (in local time, format in Hours Hours: Mins Mins) - shown in Computerized Reservations Systems (CRS)

ArrDelay : Arrival delays, in minutes

ArrDel15 : Arrival delay indicator 0 means arrival delayed less than 15 minutes. A flight is counted as arrival "on time" if it operated less than 15 minutes later the scheduled time shown in the carriers 'Computerized Reservations Systems' (CRS).

1 means arrival delayed more than 15 minutes. A flight is counted as arrival "delayed" if it operated more than 15 minutes later the scheduled time shown in the carriers 'Computerized Reservations Systems' (CRS).

Cancelled : If the flight is cancelled it is indicated by 1 = Yes ELSE 0 = No (Not cancelled)



Flight Challenge Travis Tan › Airport Codes Dataset › dataset

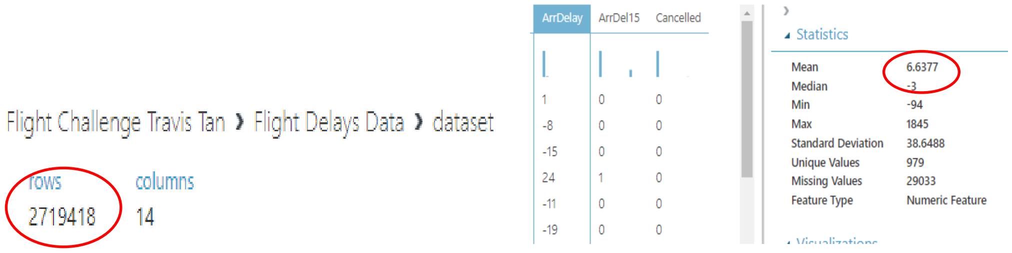
rows	columns
365	4

airport_id	city	state	name
10165	Adak Island	AK	Adak
10299	Anchorage	AK	Ted Stevens Anchorage International
10304	Aniak	AK	Aniak Airport
10754	Barrow	AK	Wiley Post/Will Rogers Memorial
10551	Bethel	AK	Bethel Airport
10926	Cordova	AK	Merle K Mudhole Smith

Airport Code Data has following features:
Airport Codes Data dataset contains details about all the airports. It has four columns and 365 rows.

airport_id: Unique Identifier for each airport
city: City in which airport is located.
state: State in which airport is located
name: Name of the airport.

Task 4 - Answer the following questions
How many rows are in the dataset?
2719418
What is the mean value of the ArrDelay column?
6.6377



Flight Challenge Travis Tan › Flight Delays Data › dataset

rows	columns
2719418	14

ArrDelay	ArrDel15	Cancelled
1	0	0
-8	0	0
-15	0	0
24	1	0
-11	0	0
-19	0	0

Statistics

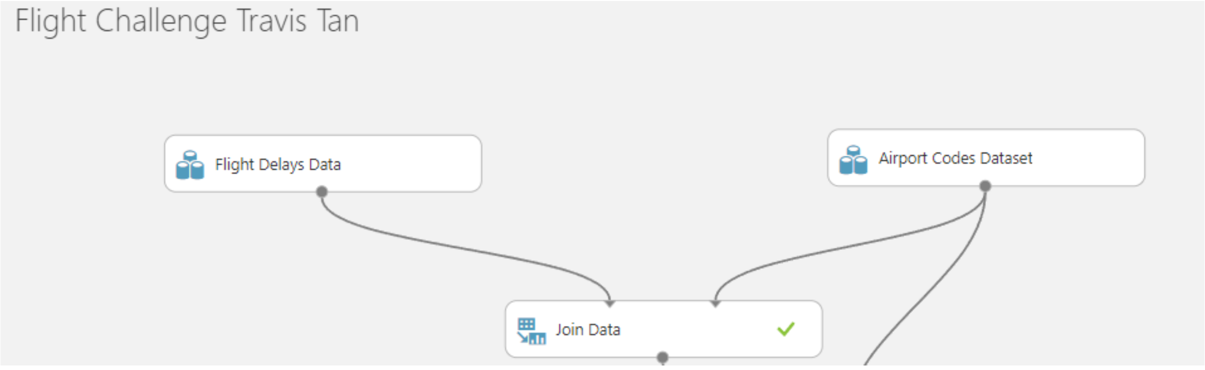
Mean	6.6377
Median	-3
Min	-94
Max	1845
Standard Deviation	38.6488
Unique Values	979
Missing Values	29033
Feature Type	Numeric Feature

Activity 4: Join the Airport Codes Dataset

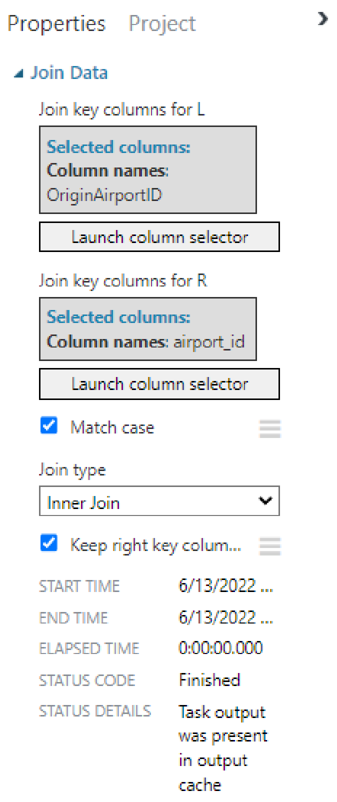
Create Origin Airport Dataset

The Flight Delays Data dataset includes the columns: OriginAirportID, DestAirportID. These are numeric identifiers for the origin airport and destination airport for each flight.

To see the details of the airports, add the Airport Codes Dataset sample dataset to the experiment, and join it to the Flight Delays Data dataset.



For the first join, the columns selected in Flight Delays Data is OriginAirportID and in Airport Codes Dataset is airport_id



Join Data

Join key columns for L

Selected columns:
Column names: OriginAirportID

Join key columns for R

Selected columns:
Column names: airport_id

☑ Match case

Join type

Inner Join

☑ Keep right key colum...

START TIME 6/13/2022 ...

END TIME 6/13/2022 ...

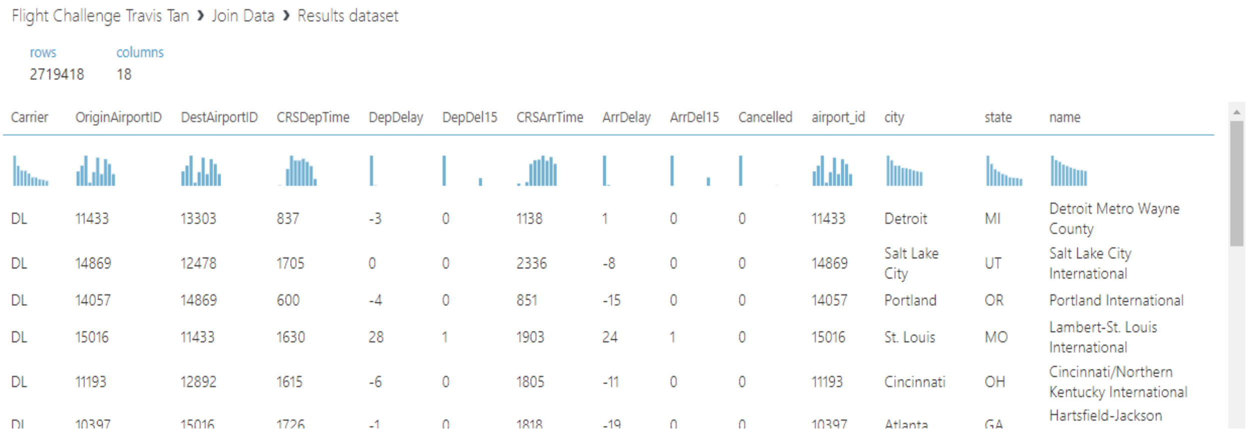
ELAPSED TIME 0:00:00.000

STATUS CODE Finished

STATUS DETAILS Task output was present in output cache

This will result an Origin Airport Dataset that contains the Airport Code information for the respective origin airport locations.

Join1 – Origin Airport Dataset



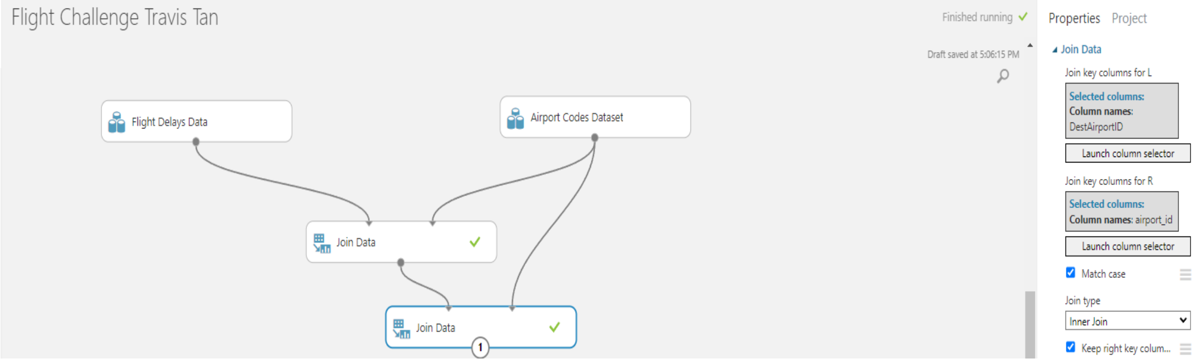
Flight Challenge Travis Tan › Join Data › Results dataset

rows	columns
2719418	18

Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled	airport_id	city	state	name
DL	11433	13303	837	-3	0	1138	1	0	0	11433	Detroit	MI	Detroit Metro Wayne County
DL	14869	12478	1705	0	0	2336	-8	0	0	14869	Salt Lake City	UT	Salt Lake City International
DL	14057	14869	600	-4	0	851	-15	0	0	14057	Portland	OR	Portland International
DL	15016	11433	1630	28	1	1903	24	1	0	15016	St. Louis	MO	Lambert-St. Louis International
DL	11193	12892	1615	-6	0	1805	-11	0	0	11193	Cincinnati	OH	Cincinnati/Northern Kentucky International
DL	10397	15016	1726	-1	0	1818	-19	0	0	10397	Atlanta	GA	Hartsfield-Jackson

Observation: Dataset results dataset contains 2719418 rows and 18 columns(14+4 = 18 columns). This shows all 4 columns related to Origin Airport are fetched the second table and gets added to the results.

Create Destination Airport Dataset



Join Data

Join key columns for L

Selected columns:
Column names: DestAirportID

Launch column selector

Join key columns for R

Selected columns:
Column names: airport_id

Launch column selector

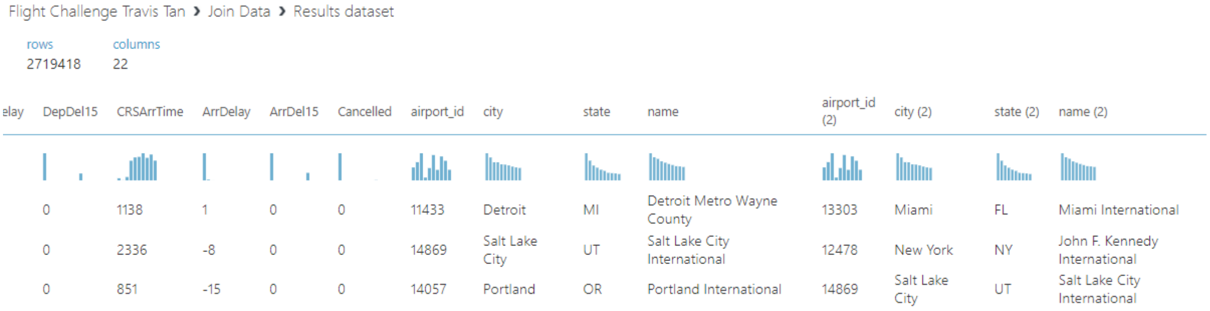
☑ Match case

Join type: Inner Join

☑ Keep right key colum...

For the second join, the columns selected in Flight Delays Data is OriginAirportID and in Airport Codes Dataset is airport_id

Join2 – Destination Airport Dataset



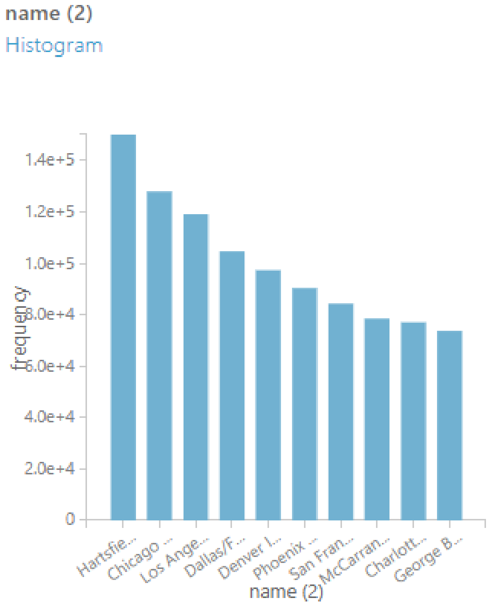
Flight Challenge Travis Tan › Join Data › Results dataset

rows	columns
2719418	22

elay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled	airport_id	city	state	name	airport_id (2)	city (2)	state (2)	name (2)
	0	1138	1	0	0	11433	Detroit	MI	Detroit Metro Wayne County	13303	Miami	FL	Miami International
	0	2336	-8	0	0	14869	Salt Lake City	UT	Salt Lake City International	12478	New York	NY	John F. Kennedy International
	0	851	-15	0	0	14057	Portland	OR	Portland International	14869	Salt Lake City	UT	Salt Lake City International

Dataset contains 2719418 rows and 22 columns (18+4=22 columns). This shows all 4 columns related to Destination Airport and Origin Airport.

Find the most frequently occurring destination airport in the dataset.



Hartsfield-Jackson Atlanta International tops the list in Histogram.

Activity 5: Remove Duplicates

Before exploring data, the data is cleaned by removing duplicate rows and replacing missing values.

*Remove duplicate rows (retaining the first instance of each row). To set the criteria for whether a row is duplicate or not, specify a single column or a set of columns to use as **keys**.*

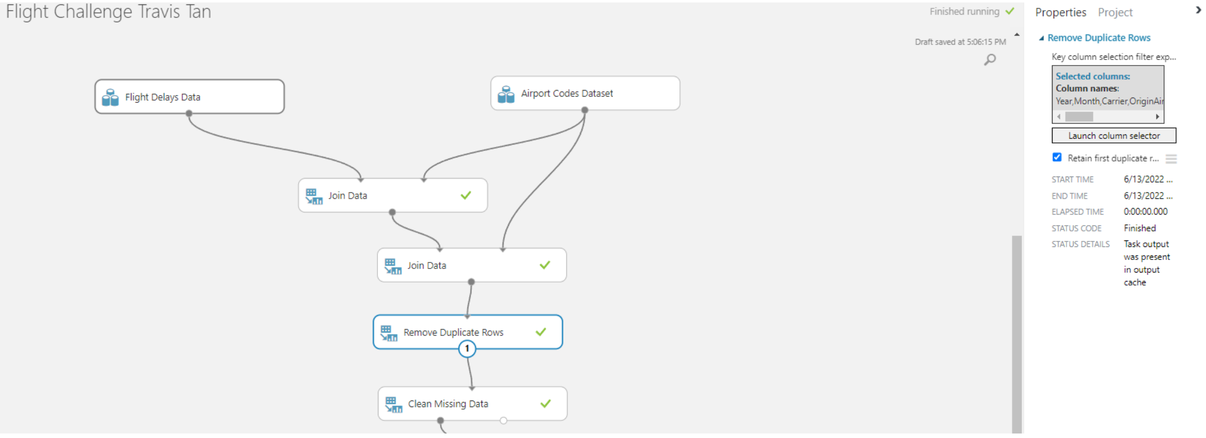
Use the Retain first duplicate row checkbox to indicate which row to return when duplicates are found: If selected, the first row is returned, and others discarded. If you uncheck this option, the last duplicate row is kept in the results, and others are discarded.

Rows are considered duplicates in this dataset if they have matching values for all the following fields:

Year, Month, DayOfMonth, Carrier, Origin Airport ID, Dest Airport ID, CRS Dep Time, CRS Arr Time

The Remove Duplicate Rows module is used to accomplish this. Two rows are considered duplicates only when the values in all key columns are equal. If any row has missing value for keys, they will not be considered duplicate rows.

Module is pulled into the experiment and joined to the output of the 2nd Join Data module. Retain first duplicate is checked.



Remove Duplicate Rows

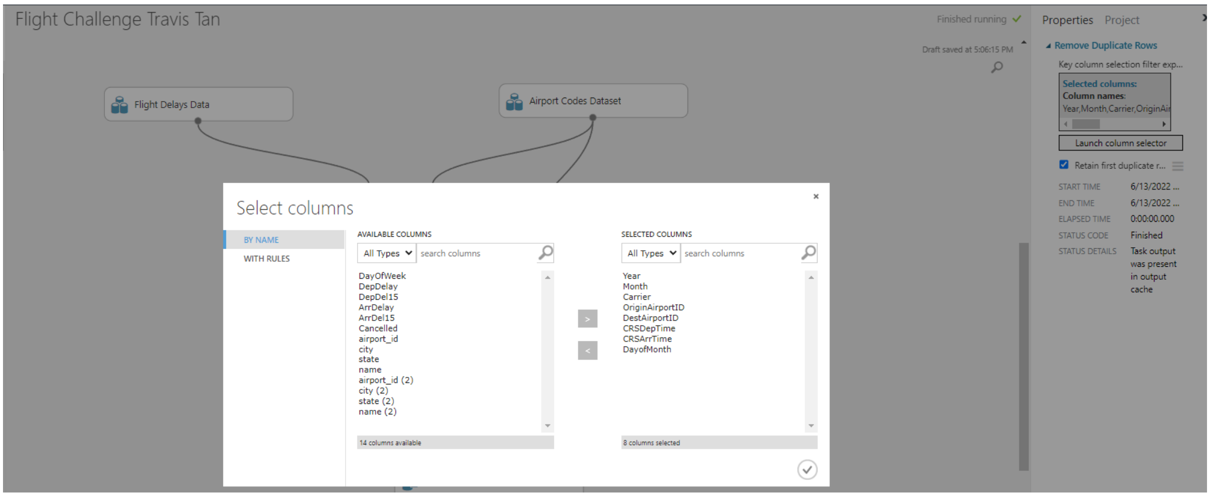
Key column selection filter exp...

Selected columns:
Column names:
Year,Month,Carrier,OriginAi

☑ Retain first duplicate r...

START TIME 6/13/2022 ...
END TIME 6/13/2022 ...
ELAPSED TIME 0:00:00.000
STATUS CODE Finished
STATUS DETAILS Task output was present in output cache

Column selector is launched and the mentioned columns are selected.



Remove Duplicate Rows

Key column selection filter exp...

Selected columns:
Column names:
Year,Month,Carrier,OriginAi

☑ Retain first duplicate r...

START TIME 6/13/2022 ...
END TIME 6/13/2022 ...
ELAPSED TIME 0:00:00.000
STATUS CODE Finished
STATUS DETAILS Task output was present in output cache

AVAILABLE COLUMNS

DayOfWeek
DepDelay
DepDel15
ArrDelay
ArrDel15
Cancelled
airport_id
city
state
name
airport_id (2)
city (2)
state (2)
name (2)

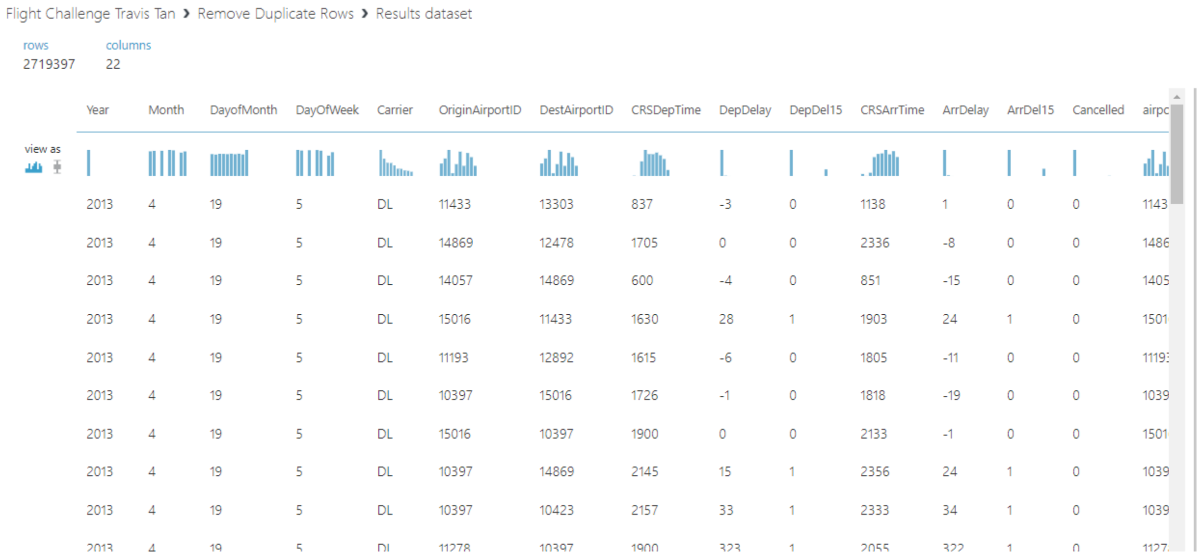
14 columns available

SELECTED COLUMNS

Year
Month
Carrier
OriginAirportID
DestAirportID
CRSDepTime
CRSArrTime
DayofMonth

8 columns selected

Visualize the output



Flight Challenge Travis Tan › Remove Duplicate Rows › Results dataset

rows 2719397

columns 22

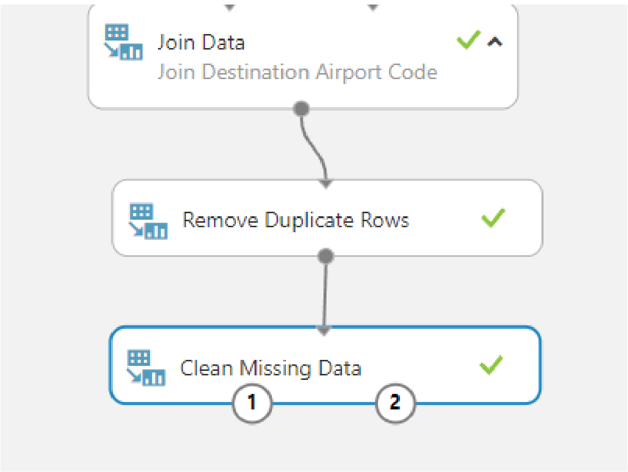
Year	Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled	airpc
2013	4	19	5	DL	11433	13303	837	-3	0	1138	1	0	0	1143
2013	4	19	5	DL	14869	12478	1705	0	0	2336	-8	0	0	1486
2013	4	19	5	DL	14057	14869	600	-4	0	851	-15	0	0	1405
2013	4	19	5	DL	15016	11433	1630	28	1	1903	24	1	0	1501
2013	4	19	5	DL	11193	12892	1615	-6	0	1805	-11	0	0	1119
2013	4	19	5	DL	10397	15016	1726	-1	0	1818	-19	0	0	1039
2013	4	19	5	DL	15016	10397	1900	0	0	2133	-1	0	0	1501
2013	4	19	5	DL	10397	14869	2145	15	1	2356	24	1	0	1039
2013	4	19	5	DL	10397	10423	2157	33	1	2333	34	1	0	1039
2013	4	19	5	DL	11278	10397	1900	323	1	2055	322	1	0	1127

Observation: After removing duplicate rows there are 2719397 rows available. So, 2719418 - 2719397 = 21 duplicate rows are removed.

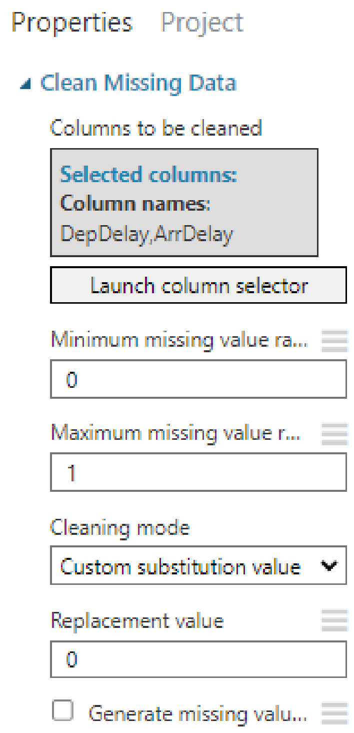
Activity 5: Replace Missing Values

After removing the duplicate rows, replace missing values in the DepDelay and ArrDelay columns with the value 0 (zero). The built-in Azure Machine Learning Clean Missing Data module is utilized to achieve the desired result.

The Clean Missing Data module is added. It is then connected to the output results from Remove Duplicate Rows with input port of Clean Missing Data module.



In the Properties pane, click Launch column selector to choose columns with missing data.
Selected columns are DepDelay and ArrDelay



Clean Missing Data

Columns to be cleaned

Selected columns:
Column names:
DepDelay,ArrDelay

Minimum missing value ra...
0

Maximum missing value r...
1

Cleaning mode
Custom substitution value

Replacement value
0

☐ Generate missing valu...

For Minimum missing value ratio, specify the minimum number of missing values required for the operation to be performed. The number you enter represents the ratio of missing values to all values in the column. By default, the Minimum missing value ratio property is set to 0. This means that missing values are cleaned even if there is only one missing value.

For Maximum missing value ratio, specify the maximum number of missing values that can be present for the operation to be performed. For example, we might want to perform missing value substitution only if 30% or fewer of the rows contain missing values but leave the values as-is if more than 30% of rows have missing values. We define the number as the ratio of missing values to all values in the column. By default, the Maximum missing value ratio is set to 1. This means that missing values are cleaned even if 100% of the values in the column are missing.

For Cleaning Mode, select one of the following options for replacing or removing missing values:
Custom substitution value: Use this option to specify a placeholder value (such as a 0 or NA) that applies to all missing values. The value that you specify as a replacement must be compatible with the data type of the column.

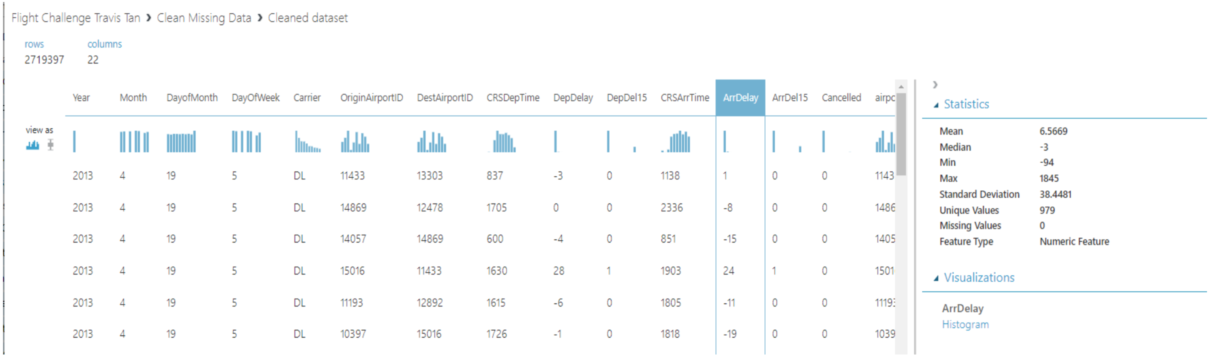
The option Replacement value is available if you have selected the option, Custom substitution value. Type a new value to use as the replacement value for all missing values in the column.

Analysis of Cleaned Dataset

Before Cleaning :
Dataset from Join Data (After 2nd Join) had missing values for DepDelay and ArrDelay.
ArrDelay feature had 29033 missing values and DepDelay feature had 27444 missing values.

After Cleaning:
Dataset from Clean Missing Data module shows missing values for DepDelay and ArrDelay are replaced with 0. ArrDelay had 0 missing values and DepDelay had 0 missing values. Which shows no missing values in both columns.

Visualize the Cleaned Dataset Port 1 of Clean Missing Data module



Flight Challenge Travis Tan › Clean Missing Data › Cleaned dataset

rows 2719397
columns 22

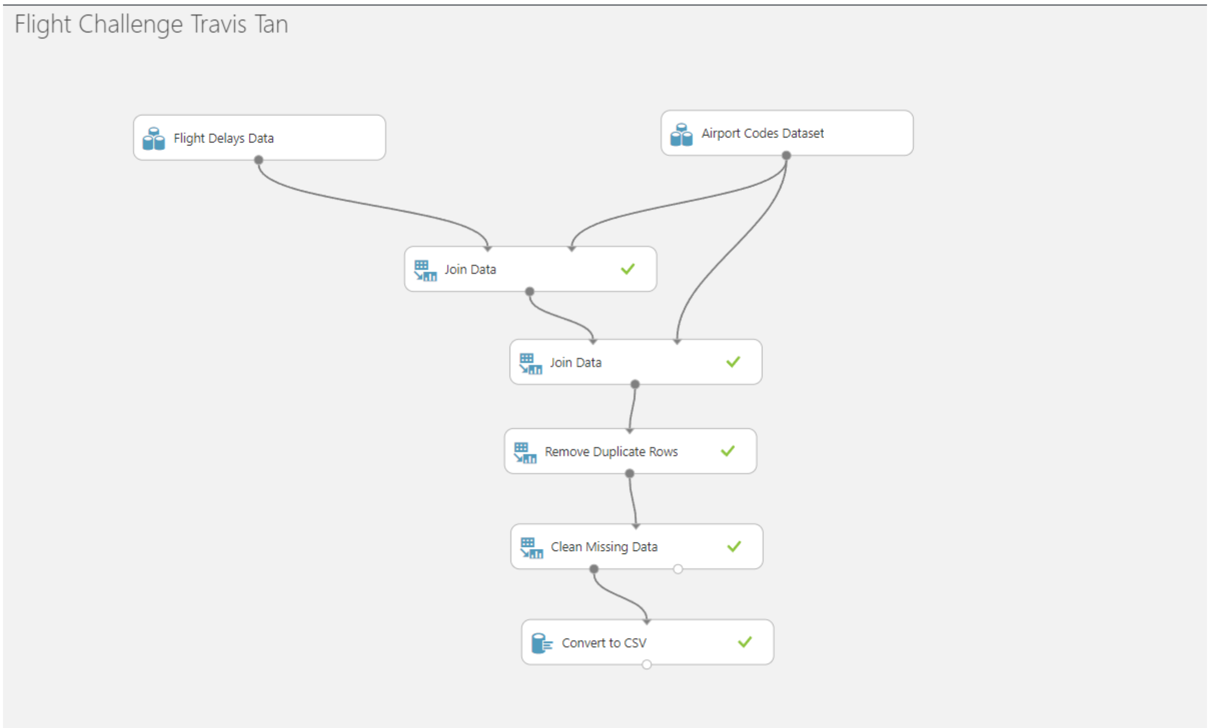
Year	Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled	airpc
2013	4	19	5	DL	11433	13303	837	-3	0	1138	1	0	0	1143
2013	4	19	5	DL	14869	12478	1705	0	0	2336	-8	0	0	1486
2013	4	19	5	DL	14057	14869	600	-4	0	851	-15	0	0	1405
2013	4	19	5	DL	15016	11433	1630	28	1	1903	24	1	0	1501
2013	4	19	5	DL	11193	12892	1615	-6	0	1805	-11	0	0	1119
2013	4	19	5	DL	10397	15016	1726	-1	0	1818	-19	0	0	1039

Statistics

Mean	6.5669
Median	-3
Min	-94
Max	1845
Standard Deviation	38.4481
Unique Values	979
Missing Values	0
Feature Type	Numeric Feature

How many rows remain in the dataset?
2719397 rows are remained.

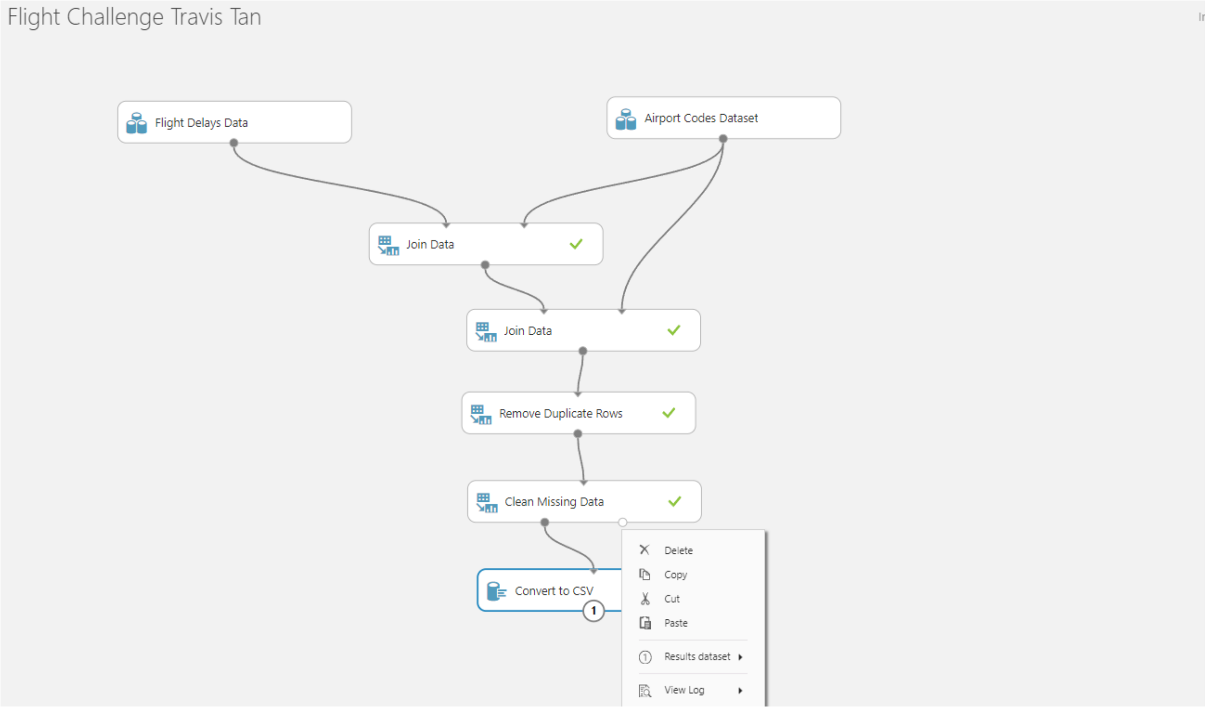
What is the mean value of the ArrDelay column?
Mean of ArrDelay = 6.5669



Overall workflow of the project (Data Cleansing segment)

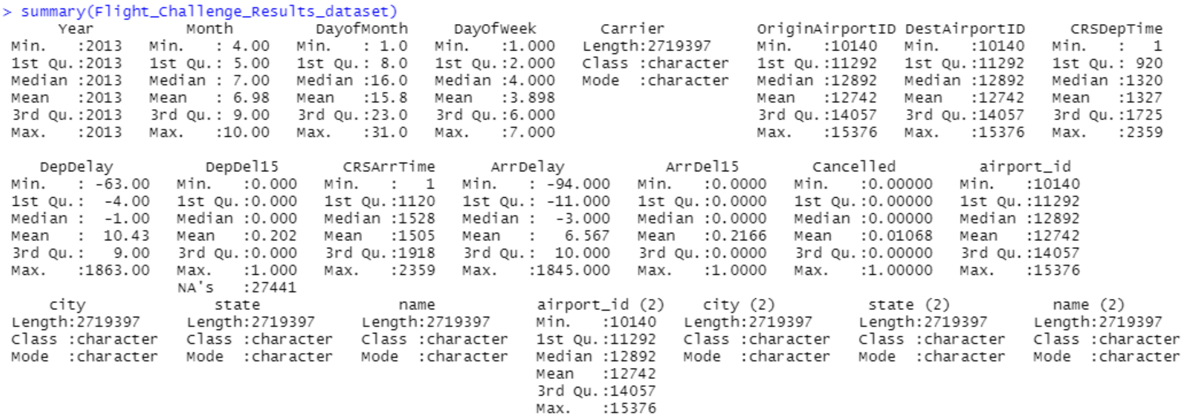
Activity 6: View Summary Statistics

The convert to CSV module was added and the experiment was ran to retrieve the CSV file for Flight destination datasheet.



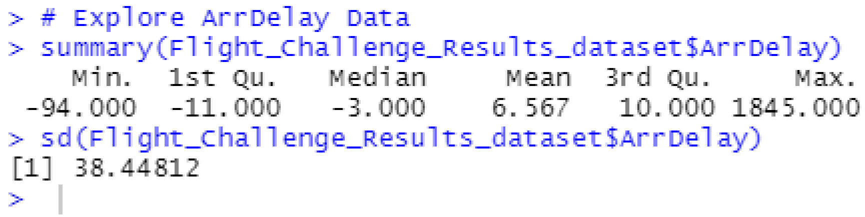
**Right click on the Convert to CSV module and select on Results Datasheet, after that click on download*

Rstudio was utilized for plotting of the necessary visualizations. First we import the CSV file into the script and we then view its Summary.



```
> summary(Flight_Challenge_Results_dataset)
      Year          Month          DayofMonth      DayOfWeek       Carrier          OriginAirportID DestAirportID     CRSDepTime
 Min.   :2013   Min.   : 4.00   Min.   : 1.0   Min.   :1.000   Length:2719397     Min.   :10140   Min.   :10140   Min.   :   1
 1st Qu.:2013   1st Qu.: 5.00   1st Qu.: 8.0   1st Qu.:2.000   Class :character   1st Qu.:11292   1st Qu.:11292   1st Qu.: 920
 Median :2013   Median : 7.00   Median :16.0   Median :4.000   Mode  :character   Median :12892   Median :12892   Median :1320
 Mean   :2013   Mean   : 6.98   Mean   :15.8   Mean   :3.898                      Mean   :12742   Mean   :12742   Mean   :1327
 3rd Qu.:2013   3rd Qu.: 9.00   3rd Qu.:23.0   3rd Qu.:6.000                      3rd Qu.:14057   3rd Qu.:14057   3rd Qu.:1725
 Max.   :2013   Max.   :10.00   Max.   :31.0   Max.   :7.000                      Max.   :15376   Max.   :15376   Max.   :2359
                                                                                                                                  
    DepDelay          DepDel15        CRSArrTime      ArrDelay            ArrDel15        Cancelled         airport_id   
 Min.   : -63.00   Min.   :0.000   Min.   :   1   Min.   : -94.000   Min.   :0.0000   Min.   :0.00000   Min.   :10140  
 1st Qu.:  -4.00   1st Qu.:0.000   1st Qu.:1120   1st Qu.: -11.000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:11292  
 Median :  -1.00   Median :0.000   Median :1528   Median :  -3.000   Median :0.0000   Median :0.00000   Median :12892  
 Mean   :  10.43   Mean   :0.202   Mean   :1505   Mean   :   6.567   Mean   :0.2166   Mean   :0.01068   Mean   :12742  
 3rd Qu.:   9.00   3rd Qu.:0.000   3rd Qu.:1918   3rd Qu.:  10.000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:14057  
 Max.   :1863.00   Max.   :1.000   Max.   :2359   Max.   :1845.000   Max.   :1.0000   Max.   :1.00000   Max.   :15376  
                   NA's   :27441                                                                                         
     city              state               name            airport_id (2)    city (2)           state (2)            name (2)        
 Length:2719397     Length:2719397     Length:2719397     Min.   :10140   Length:2719397     Length:2719397     Length:2719397    
 Class :character   Class :character   Class :character   1st Qu.:11292   Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character   Median :12892   Mode  :character   Mode  :character   Mode  :character  
                                                          Mean   :12742                                                           
                                                          3rd Qu.:14057                                                           
                                                          Max.   :15376                                                           
```

ArrDelay column was then analysed by using summary() and sd() functions to retrieve more statistical information.



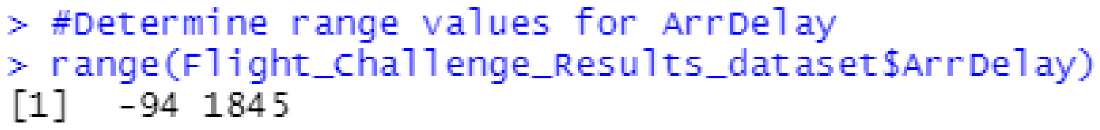
```
> # Explore ArrDelay Data
> summary(Flight_Challenge_Results_dataset$ArrDelay)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
 -94.000  -11.000   -3.000    6.567   10.000 1845.000 
> sd(Flight_Challenge_Results_dataset$ArrDelay)
[1] 38.44812
> 
```

We found out the following about ArrDelay
Min = -94
Mean = 6.567
SD = 38.44812
Max = 1845

Activity 7: View ArrDelay Distribution

Determine the range and distribution of values for ArrDelay

range() function was called to determine the range

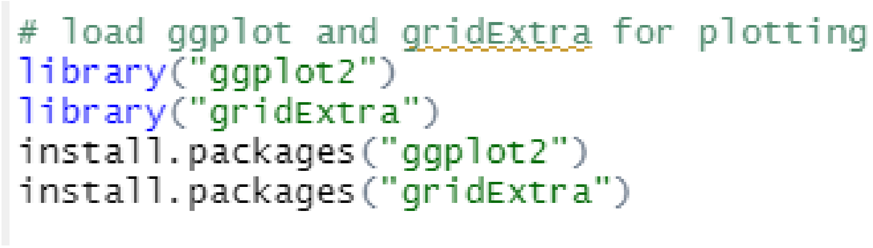


```
> #Determine range values for ArrDelay
> range(Flight_Challenge_Results_dataset$ArrDelay)
[1]  -94 1845
```

We can observe that it ranges from a negative value of 94 to 1845. In total the absolute range will be 94 + 1845 = 1939

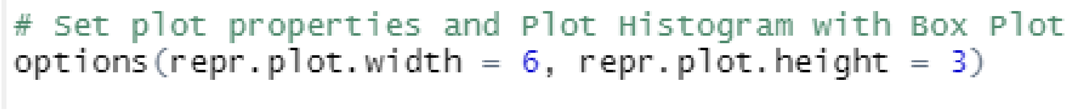
Explore the range and distribution of values in the ArrDelay column using Histogram and Boxplot

ggplot2 and gridExtra was loaded and installed for the plotting of Histogram and Boxplot.



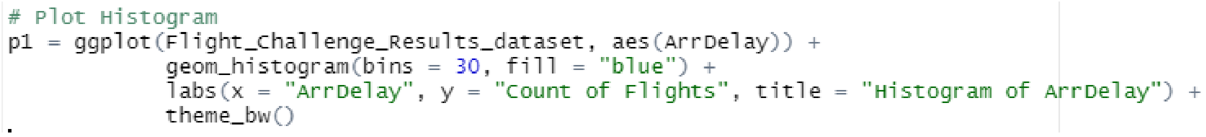
```
# load ggplot and gridExtra for plotting
library("ggplot2")
library("gridExtra")
install.packages("ggplot2")
install.packages("gridExtra")
```

Properties for the plot are set at width = 6 and height = 3

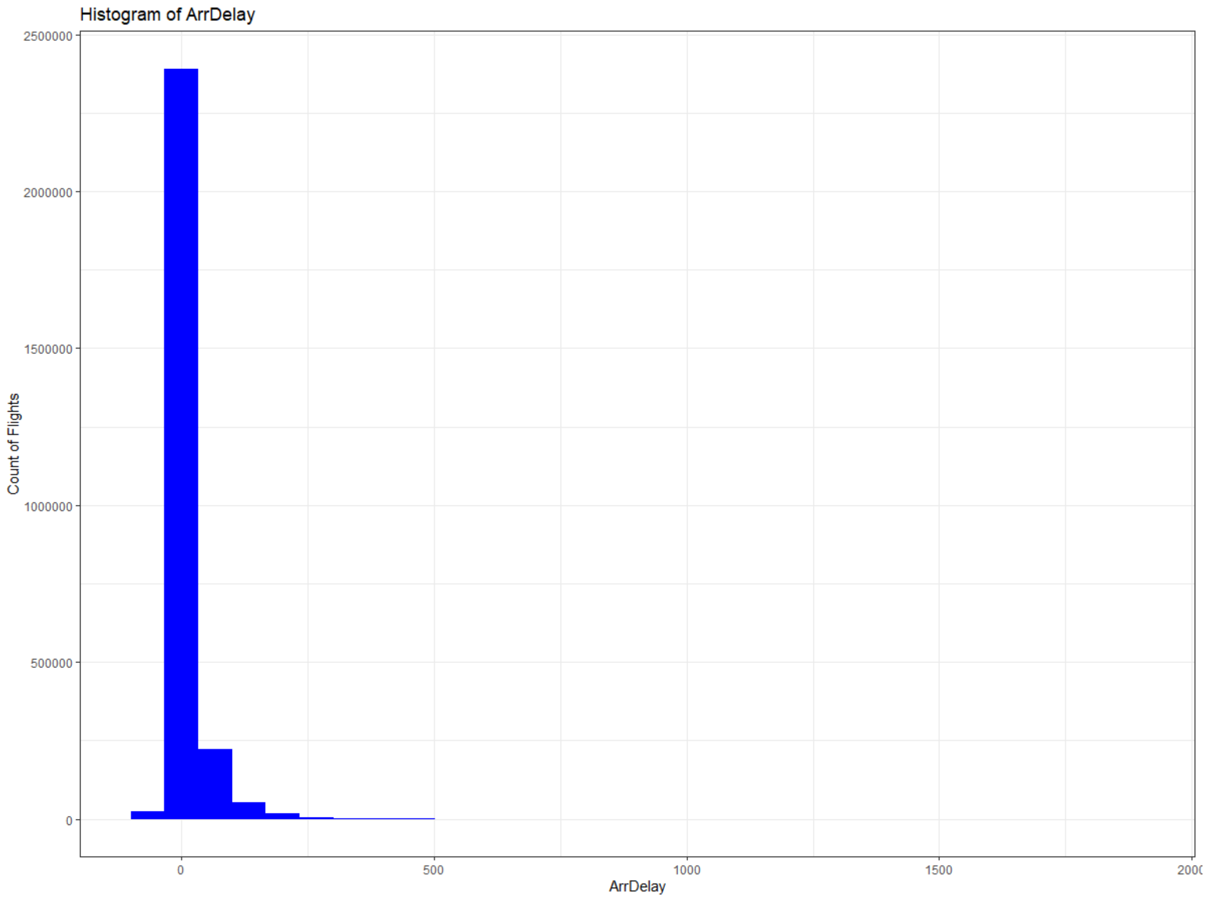


```
# Set plot properties and Plot Histogram with Box Plot
options(repr.plot.width = 6, repr.plot.height = 3)
```

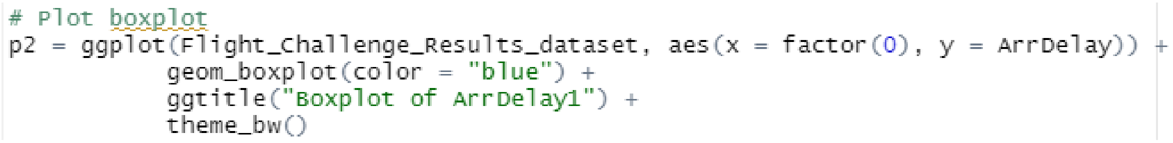
Histogram was plot with ArrDelay as the x Axis, bins set to 30 and color of the visualization to be blue.



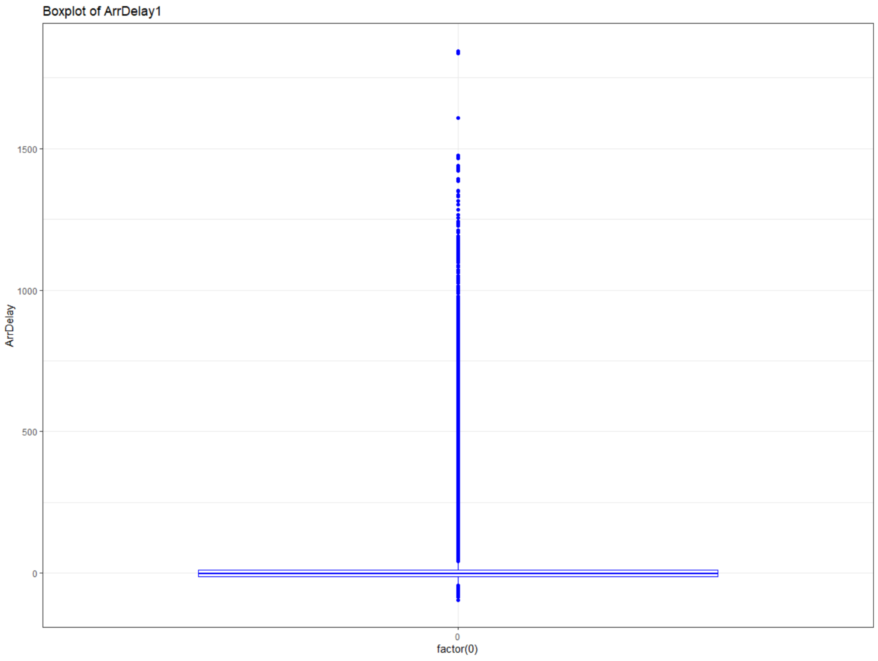
```
# Plot Histogram
p1 = ggplot(Flight_Challenge_Results_dataset, aes(ArrDelay)) +
            geom_histogram(bins = 30, fill = "blue") +
            labs(x = "ArrDelay", y = "Count of Flights", title = "Histogram of ArrDelay") +
            theme_bw()
```



Boxplot was plot with a 0 factor for the x Axis and ArrDelay as the y Axis, with color of the visualization to be blue.

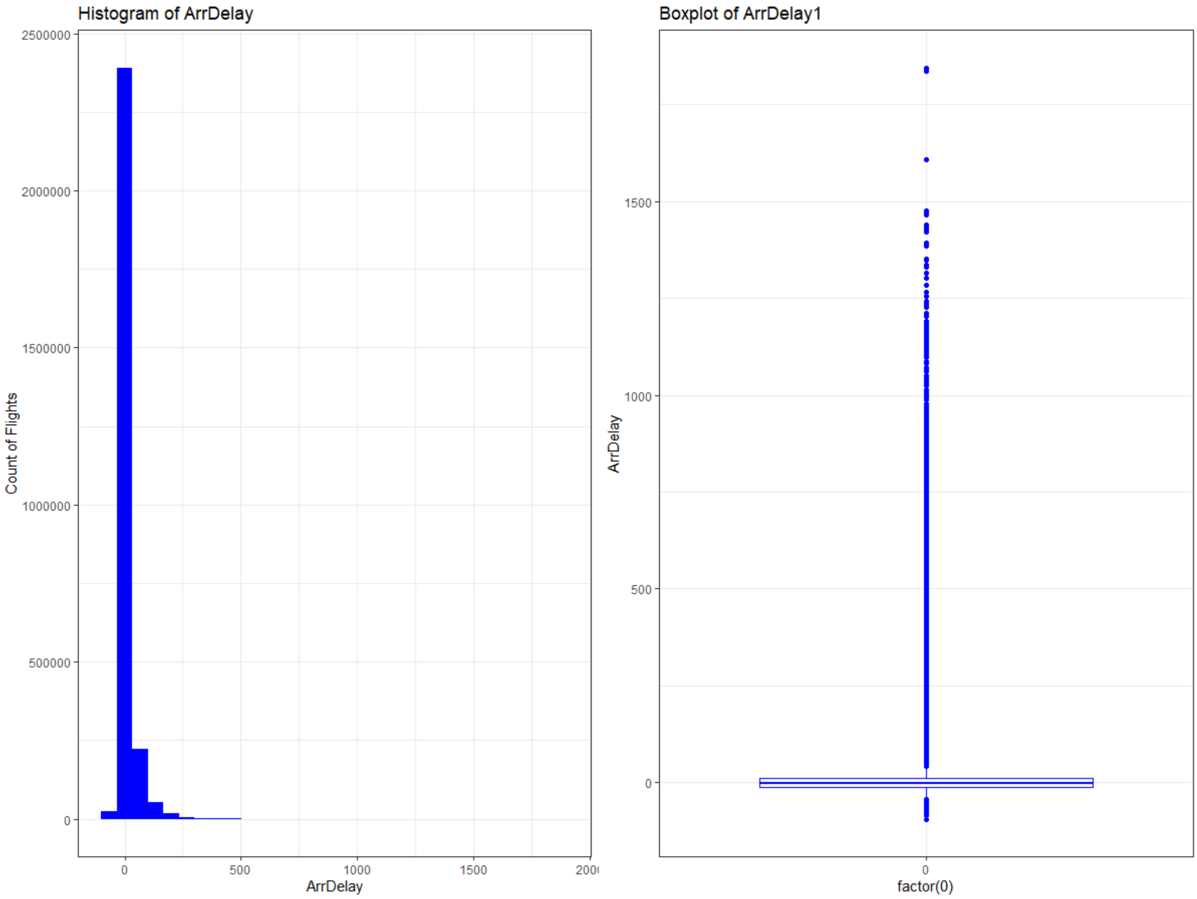


```
# Plot boxplot
p2 = ggplot(Flight_Challenge_Results_dataset, aes(x = factor(0), y = ArrDelay)) +
          geom_boxplot(color = "blue") +
          ggtitle("Boxplot of ArrDelay1") +
          theme_bw()
```



Histogram and Box plot are then put together so that we can visualize them together as a whole

```
# Arrange the Histogram and Box plot on same row
grid.arrange(p1,p2,nrow=1)
```



Histogram Observations:

1) Majority of flights Arrive with a Delay less than 15 mins from the scheduled arrival time.

2) Peak of the distribution occurs when the time is less than 0. It means most of the flights arrive early than the scheduled arrival time.

3) Median -3 mins indicates majority of flights arrive early (i.e.) close to 3 mins earlier.

4) Distribution is right skewed.

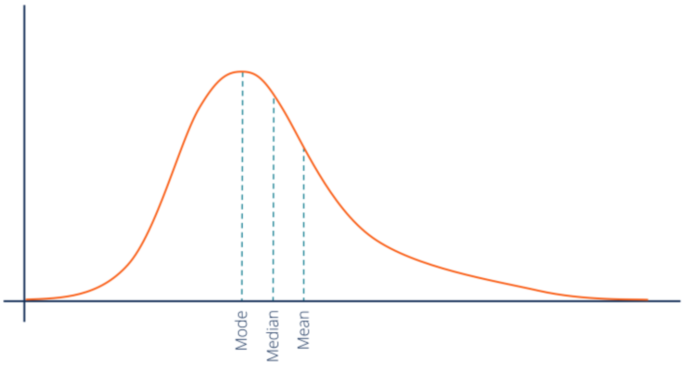
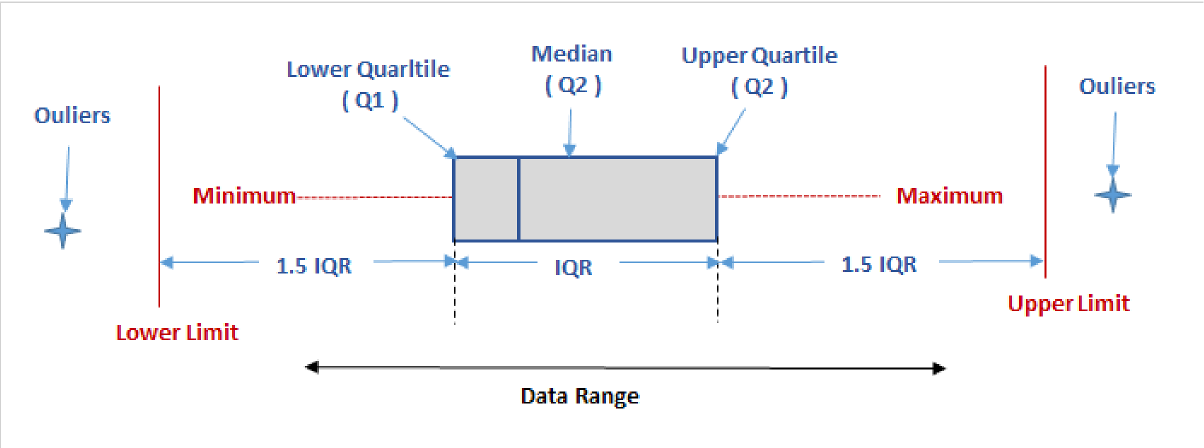


Image of a right skewed distribution

Boxplot Observations:
Using the Summary and SD function that we have called earlier, we can affirm the statements with the boxplots
1. Minimum: -94.000
2. 1st Quartile: -11.000 (The value that cuts off first 25% of data when sorted in ascending order or the inner border of the box)
3. Median: -3.000 (The second quartile, or median, is the value that cuts off the first 50% or the centre line within the box)
4. 3rd Quartile: 10.000 (is the value that cuts off the first 75% or the outer border of the box)
5. Maximum: 1845.000 (The last data point on the graph)

1) Here Q1 = -11 So 25% of Arrival Delay of flights is below -11 mins. Means 25% of Flights reach less than 11 minutes before the scheduled Arrival Time. These flights arrive early from 11 minutes to 94 minutes.
2) Q3 = 10. Here 75 % of Arrival Delay of flights is between -11mins and 1845 minutes. Means 75% of Flights reach between early by 11 minutes to late by 1845 mins from the scheduled Arrival Time.
3) Box is centred within -11 to 10. There appears to contain many outliers. With many points outside of the IQR.



Based on the data visualization, we can infer the following statements accurately reflect the distribution of ArrDelay values:

1. The median, first quartile, and third quartile are all fairly close to 0, indicating that most flights arrive close to their scheduled time.
2. The range of arrival times ranges extensively, with some flights arriving as much as 1500 minutes late.
3. The distribution is right-skewed, so there is a higher range of values for late flights than for early flights.

Activity 8: Use Histograms to Compare Numeric Columns

The flights dataset includes several numeric features (for example DepDelay, which indicates the number of minutes late a flight departed) or pseudo-numeric features (for example CRSDepTime, which indicates the scheduled departure time as a whole number in 24-hour clock format).

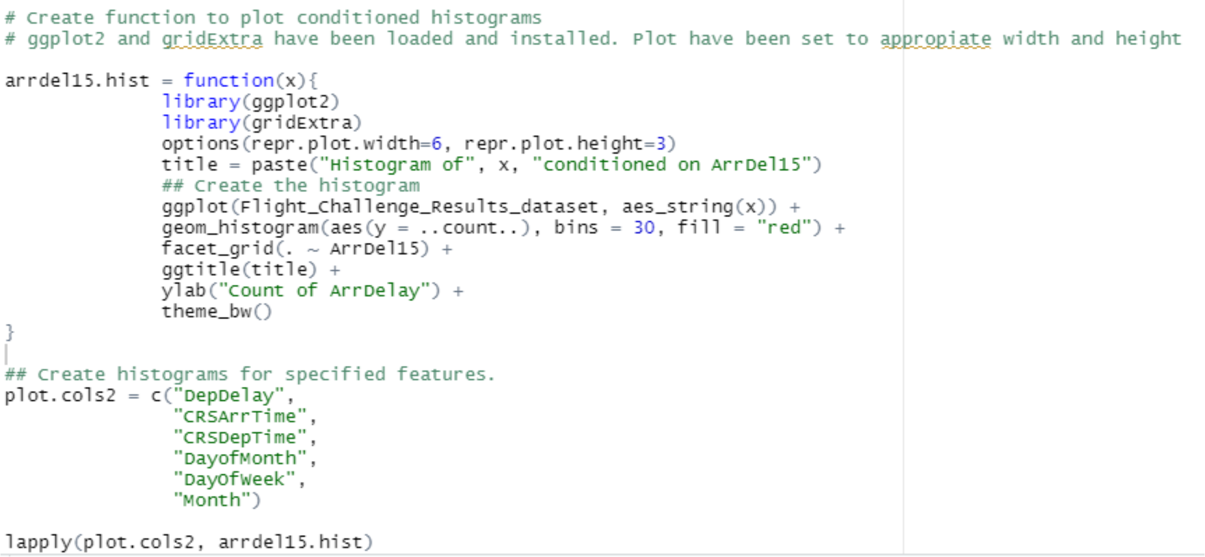
Explore how these values might be related to arrival delay, plot histograms conditioned by the ArrDel15 column, which is a binary column indicating whether a flight arrived 15 or more minutes late.

Write code to generate conditioned histograms for the following columns, conditioned by the ArrDel15 column:

- DepDelay
- CRSArrTime
- CRSDepTime
- DayofMonth
- DayOfWeek
- Month

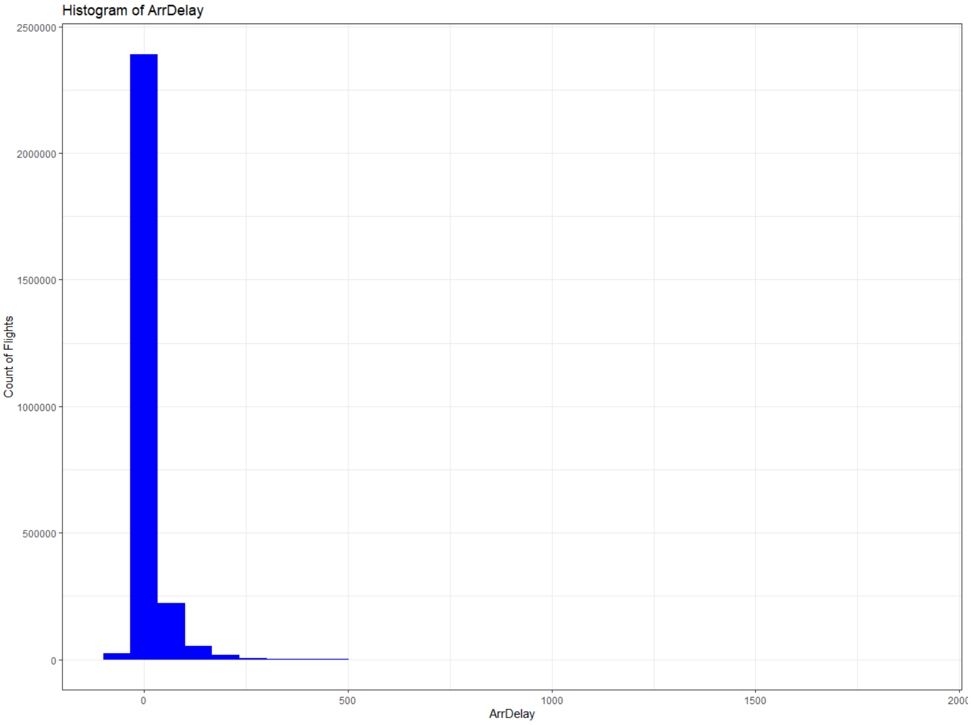
A function is created along with a vector of conditioned columns that we are plotting ArrDel15 against with.

It is then parsed through lapply() to create multiple plots.

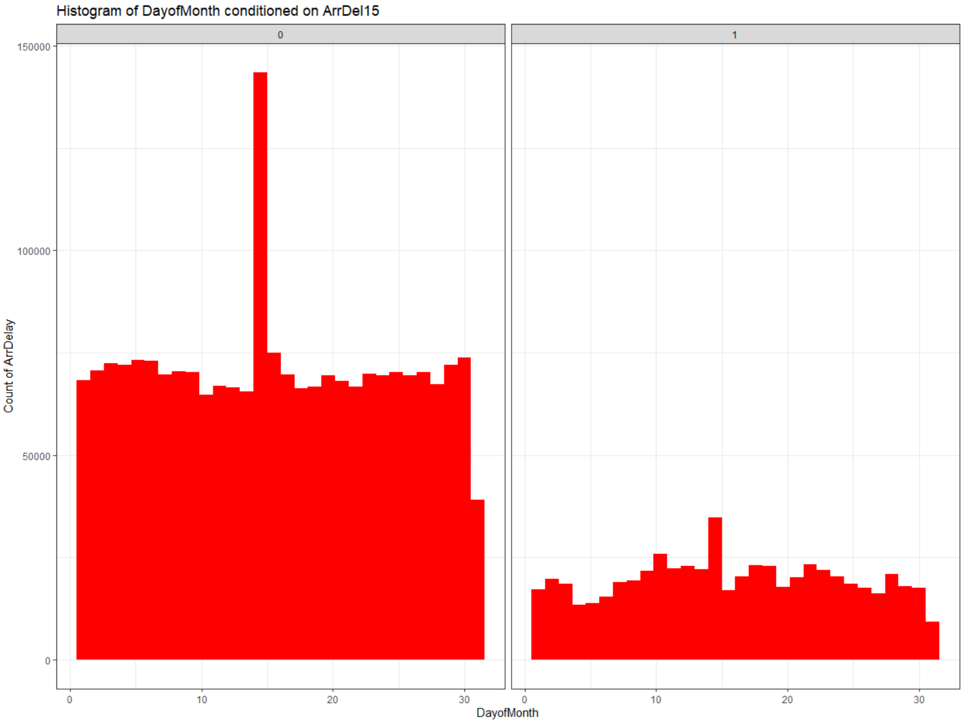


```r
# Create function to plot conditioned histograms
# ggplot2 and gridExtra have been loaded and installed. Plot have been set to appropiate width and height

arrdel15.hist = function(x){
              library(ggplot2)
              library(gridExtra)
              options(repr.plot.width=6, repr.plot.height=3)
              title = paste("Histogram of", x, "conditioned on ArrDel15")
              ## Create the histogram
              ggplot(Flight_Challenge_Results_dataset, aes_string(x)) +
              geom_histogram(aes(y = ..count..), bins = 30, fill = "red") +
              facet_grid(. ~ ArrDel15) +
              ggtitle(title) +
              ylab("Count of ArrDelay") +
              theme_bw()
}

## Create histograms for specified features.
plot.cols2 = c("DepDelay",
               "CRSArrTime",
               "CRSDepTime",
               "DayofMonth",
               "DayOfWeek",
               "Month")

lapply(plot.cols2, arrdel15.hist)
```

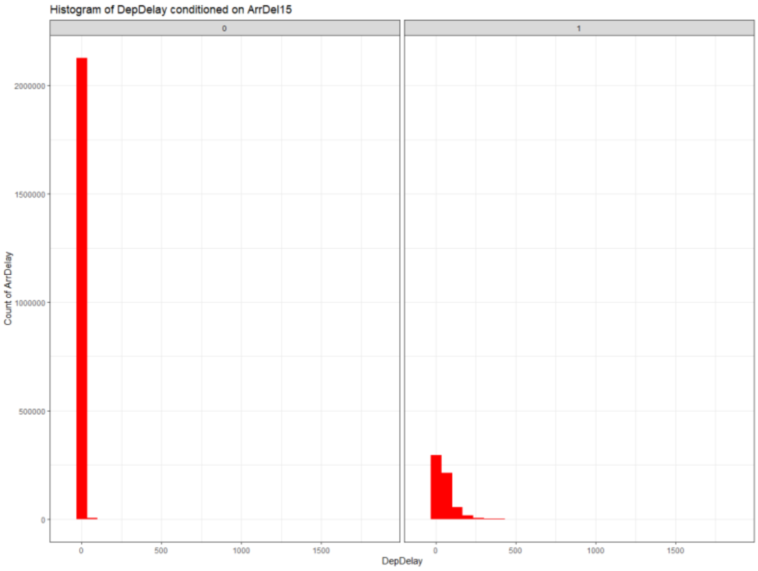
Based on the conditioned histograms, we can derive the following statements:



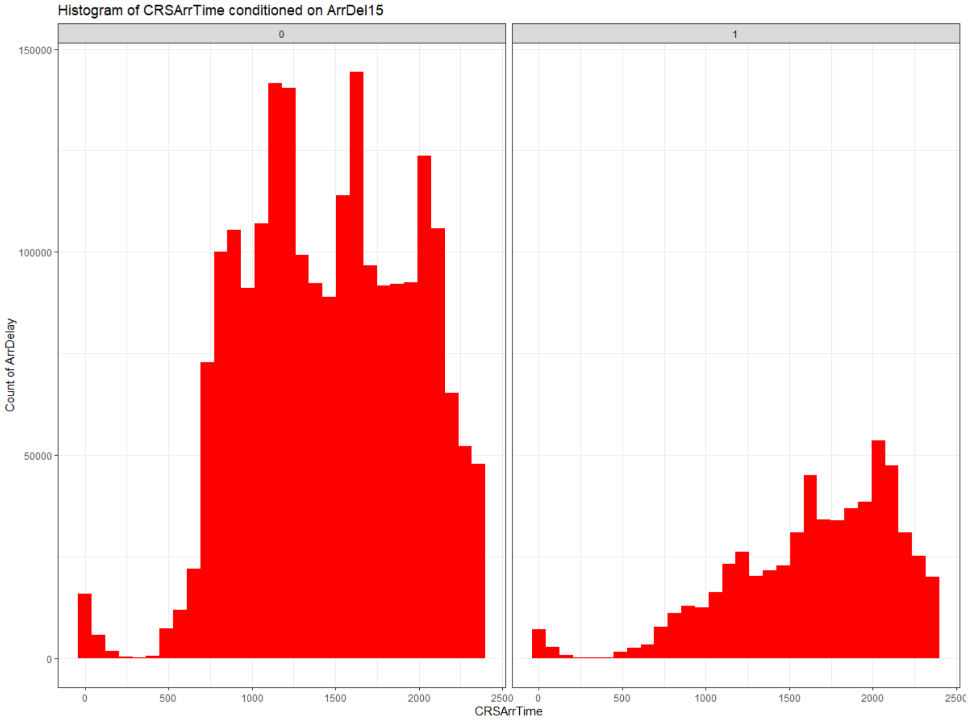
There are significantly more flights that are less than 15 minutes late than there are flights that are 15 minutes late or more.



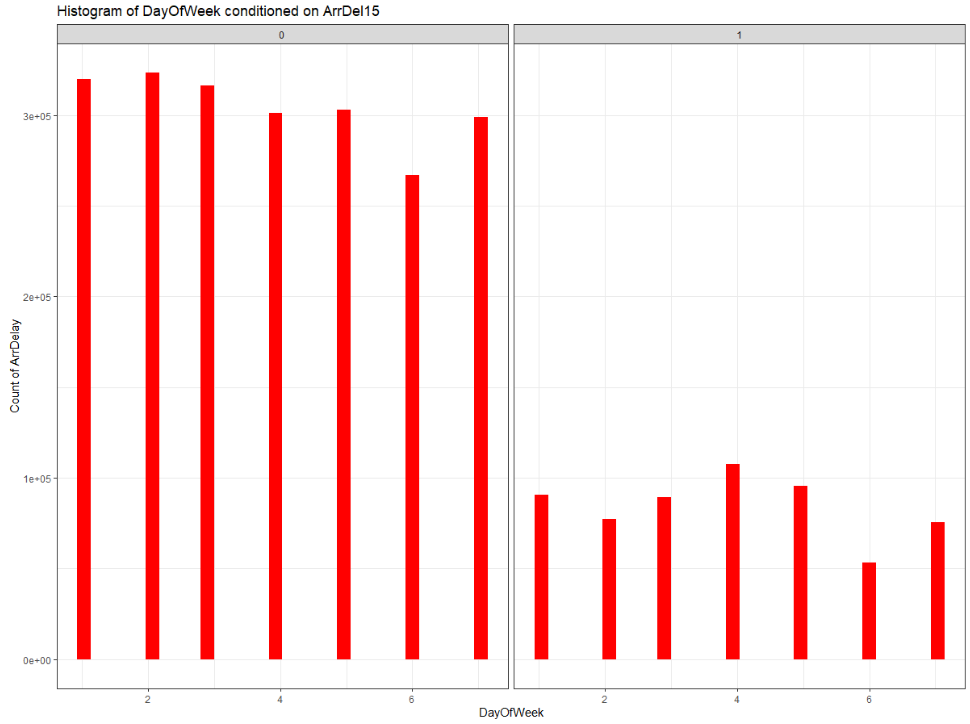
Late flights tend to occur more frequently at the middle of the month.



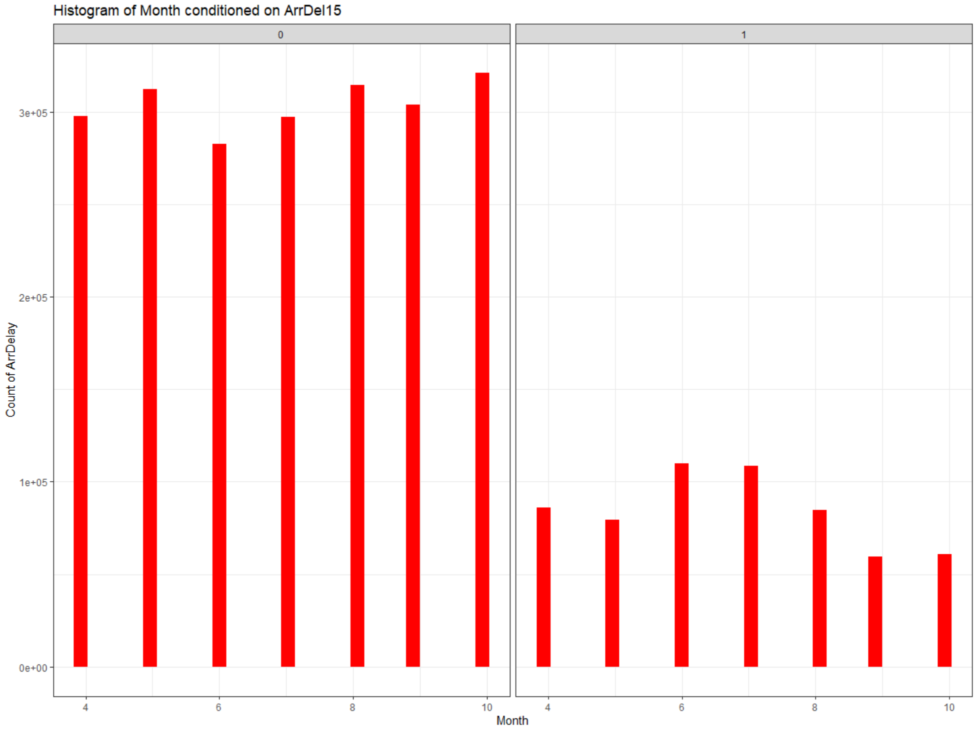
Flights that are 15 minutes or more late tend to have a higher DepDelay value than flights that are on-time.



Late flights tend to occur more frequently for flights with a CRSArrTime that is later in the day, the highest volume of delayed flights scheduled to arrive between 3pm (1500 hours) and 8pm (2000 hours)



The relative distribution of late flights varies significantly from that of on-time flights based on the day of the week.



Activity 9: Use Scatter Plots to Compare Numeric Columns

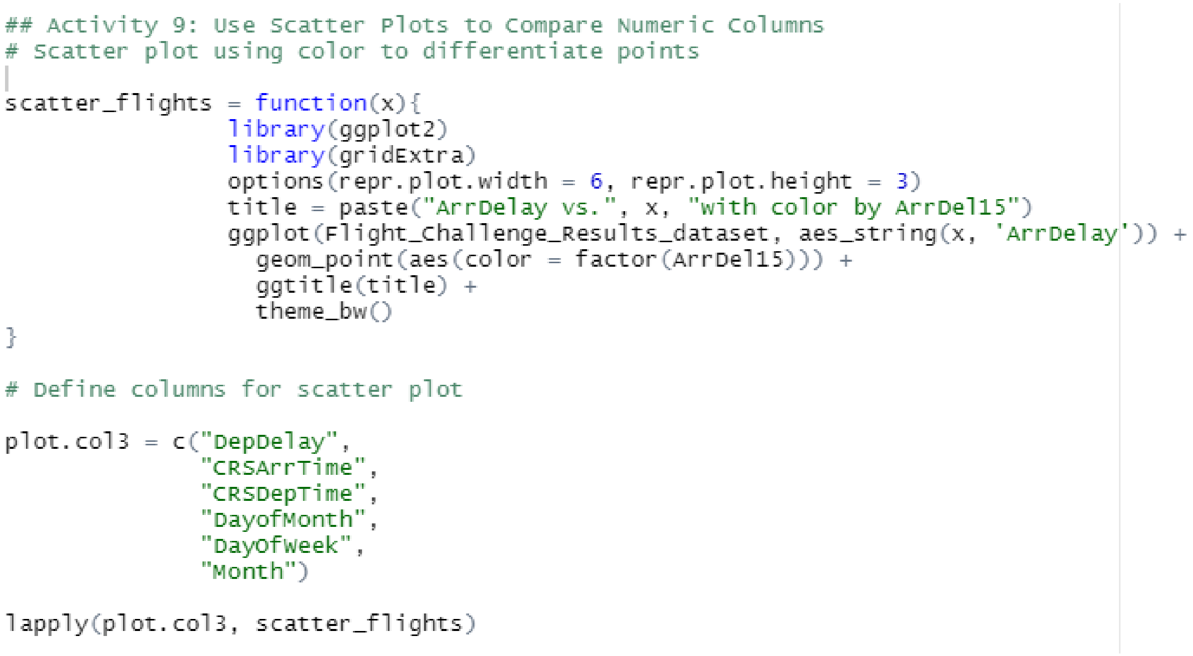
TScatter plots can be used to compare two numeric values and can be conditioned on one or more variables using colors and shapes.

generate conditioned scatter plots for the following columns, conditioned by the ArrDel15 column using different colors for values of 0 and 1:

- DepDelay
- CRSArrTime
- CRSDepTime
- DayofMonth
- DayOfWeek
- Month

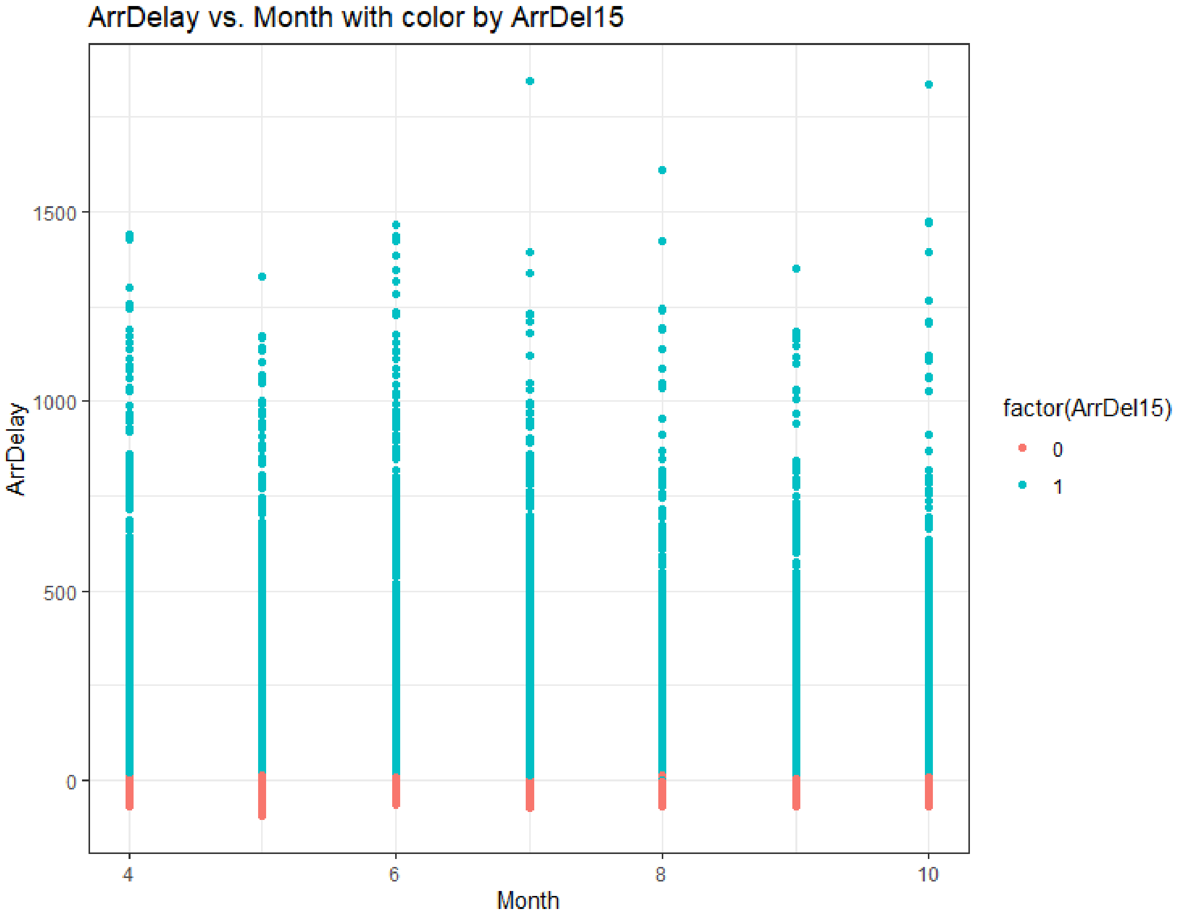
A function is created along with a vector of conditioned columns that we are plotting ArrDel15 against with.

It is then parsed through lapply() to create multiple plots.

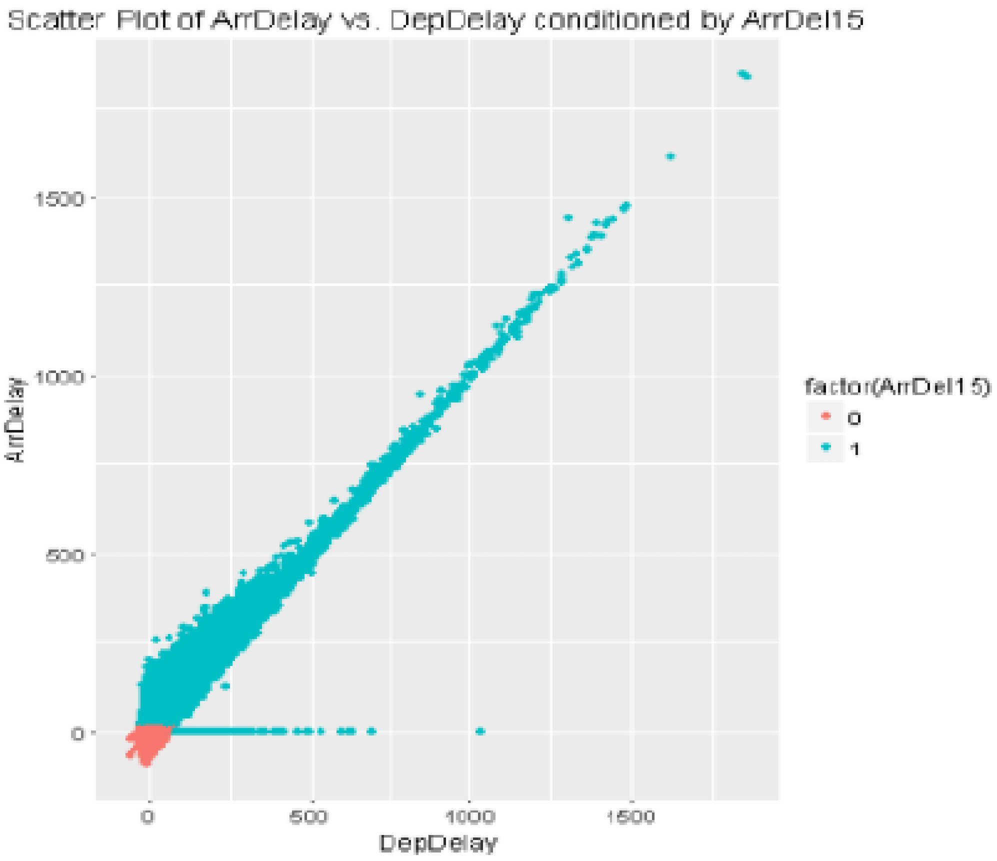


```
## Activity 9: Use Scatter Plots to Compare Numeric Columns
# Scatter plot using color to differentiate points

scatter_flights = function(x){
                library(ggplot2)
                library(gridExtra)
                options(repr.plot.width = 6, repr.plot.height = 3)
                title = paste("ArrDelay vs.", x, "with color by ArrDel15")
                ggplot(Flight_Challenge_Results_dataset, aes_string(x, 'ArrDelay')) +
                  geom_point(aes(color = factor(ArrDel15))) +
                  ggtitle(title) +
                  theme_bw()
}

# Define columns for scatter plot

plot.col3 = c("DepDelay",
              "CRSArrTime",
              "CRSDepTime",
              "DayofMonth",
              "DayOfWeek",
              "Month")

lapply(plot.col3, scatter_flights)
```

Based on the conditioned scatter plots, we can derive the following statements:

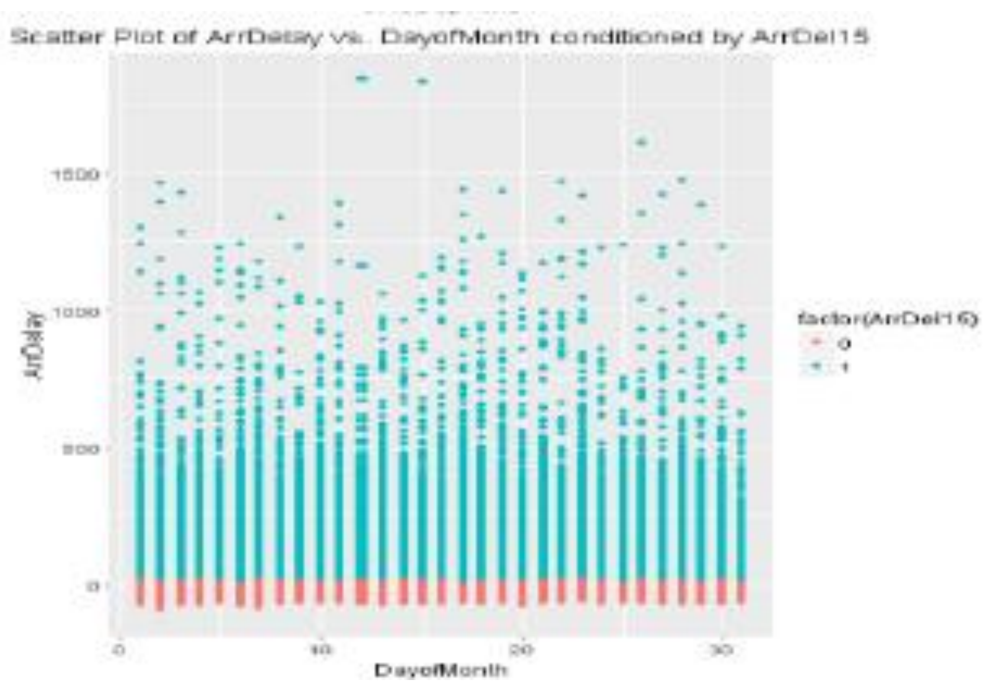


There is no clear relationship between ArrDelay and month. Later months of the year show markedly longer delays.

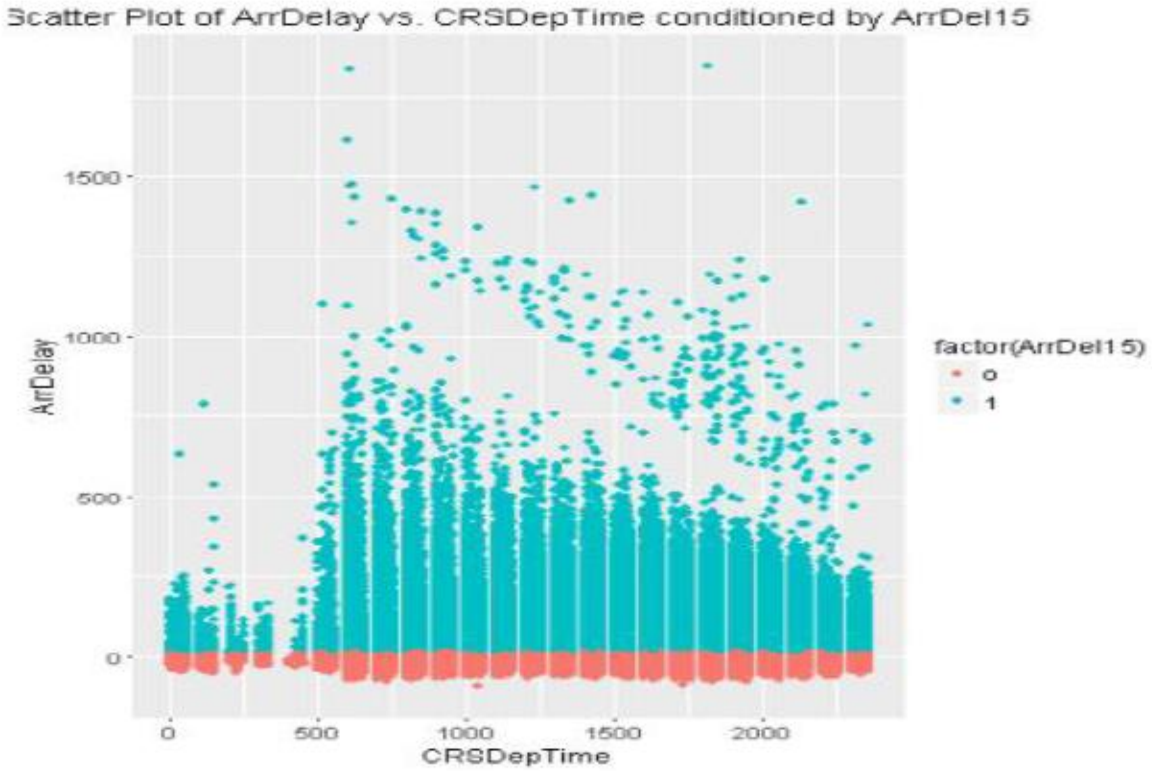


There is a near-linear relationship between DepDelay and ArrDelay for late flights. As departure delay increases, so does arrival delay.

ArrDelay is positively correlated to DepDelay. Whenever Flight departs with a delay then there is an equal amount of delay in Arrival Time.



There is no clear correlation between ArrDelay and DayofMonth. Earlier days of the month show markedly longer delays.
ArrDelay is evenly distributed throughout the month.



There is an apparent relationship between ArrDelay and CRSDepTime. Flights that depart early in the morning are typically less delayed than flights that are scheduled to depart after around 5am (0500 hours), at which time delays tend to get significantly longer. Delays then gradually get shorter as the day progresses.

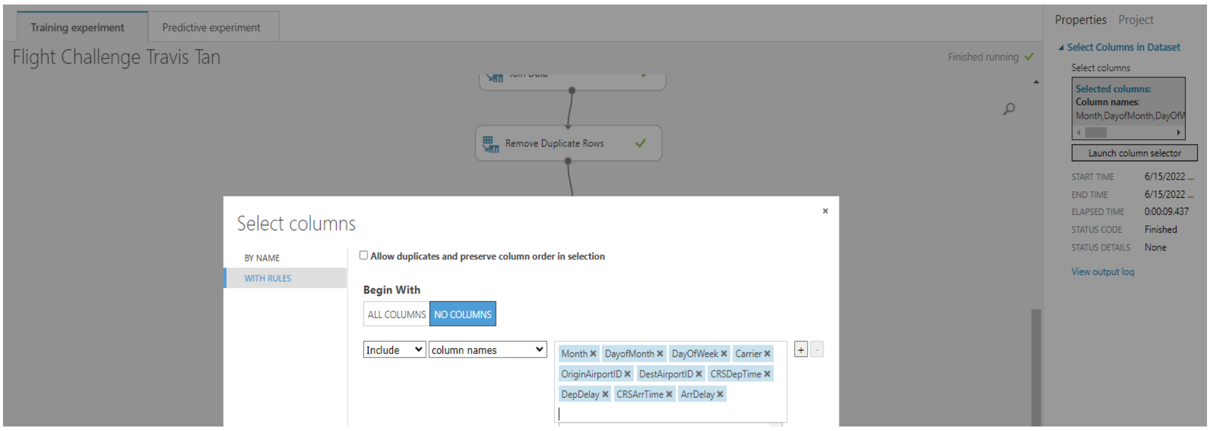
Flights that depart early in the morning are typically less delayed than flights that are scheduled to depart after around 5am (0500 hours), at which time delays tend to get significantly.

Activity 10: Train a regression model

Using Azure Machine Learning Studio, we continue with our experiment and delete the Convert to CSV module.

Select columns in Dataset module is dragged into the experiment. In the properties pane click on launch column selector and select the following columns:

- Month
- DayofMonth,
- DayOfWeek
- Carrier
- OriginAirportID
- DestAirportID
- CRSDepTime
- DepDelay,
- CRSArrTime
- ArrDelay



Select Columns in Dataset

Select columns

Selected columns:
Column names:
Month,DayofMonth,DayOfW

START TIME	6/15/2022 ...
END TIME	6/15/2022 ...
ELAPSED TIME	0:00:09.437
STATUS CODE	Finished
STATUS DETAILS	None

Select columns

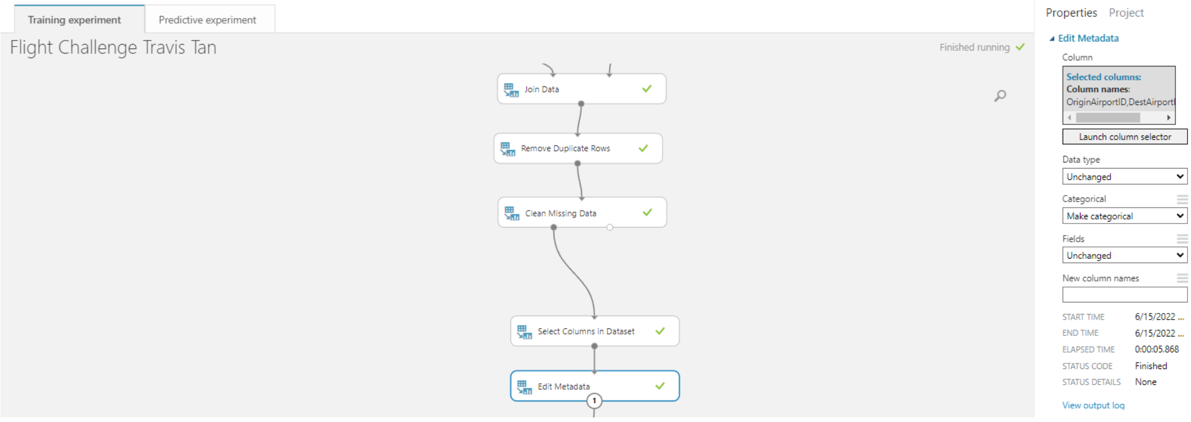
□ Allow duplicates and preserve column order in selection

Begin With: NO COLUMNS

Include | column names

Month ✖ DayofMonth ✖ DayOfWeek ✖ Carrier ✖ OriginAirportID ✖ DestAirportID ✖ CRSDepTime ✖ DepDelay ✖ CRSArrTime ✖ ArrDelay ✖

Edit Metadata module is used to convert OriginAirportID, DestAirportID, and Carrier columns Categorical.



Edit Metadata

Column

Selected columns:
Column names:
OriginAirportID,DestAirportI

Launch column selector

Data type: Unchanged

Categorical: Make categorical

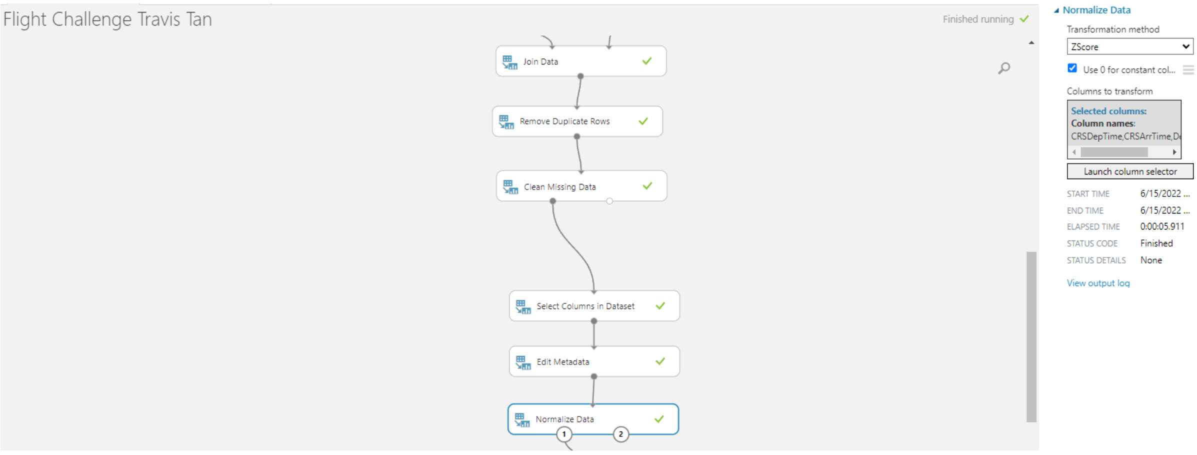
Fields: Unchanged

New column names:

START TIME	6/15/2022 ...
END TIME	6/15/2022 ...
ELAPSED TIME	0:00:05.868
STATUS CODE	Finished
STATUS DETAILS	None

View output log

Normalized Data module is then used to standardize CRSDepTime, CRSArrTime, and DepDelay columns using the ZScore transformation method.



Normalize Data

Transformation method: ZScore

☑ Use 0 for constant col...

Columns to transform

Selected columns:
Column names:
CRSDepTime,CRSArrTime,De

Launch column selector

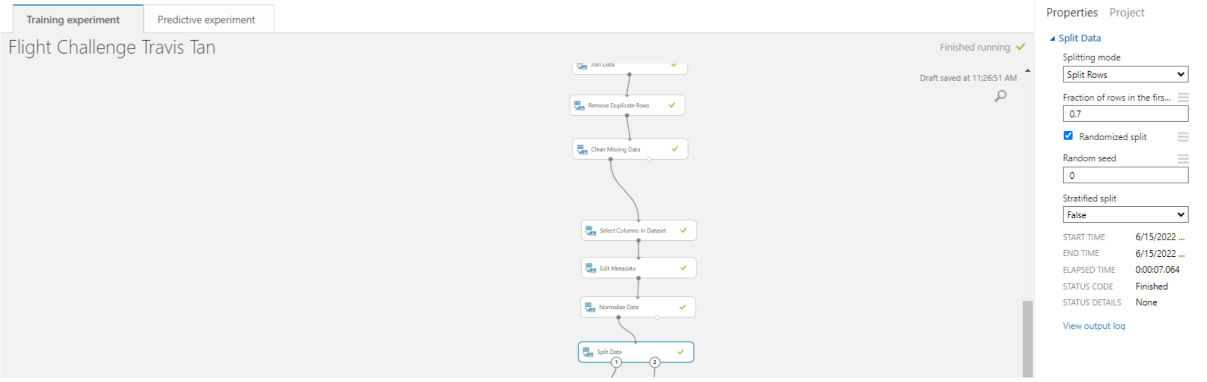
START TIME	6/15/2022 ...
END TIME	6/15/2022 ...
ELAPSED TIME	0:00:05.911
STATUS CODE	Finished
STATUS DETAILS	None

View output log

The goal of normalization is to make every datapoint have the same scale, so each feature is equally important. Variables that are measured at different scales do not contribute equally to the model fitting. CRSDepTime, CRSArrTime and DepDelay columns are scaled to a proper proportion.

Split Data module is used to split the rows into 70% / 30% subsets. Random seed value of 0 is used.

Here 70% of data is taken for Training and 30 % data for Testing purpose.



◢ Split Data

Splitting mode: Split Rows

Fraction of rows in the firs...: 0.7

☑ Randomized split

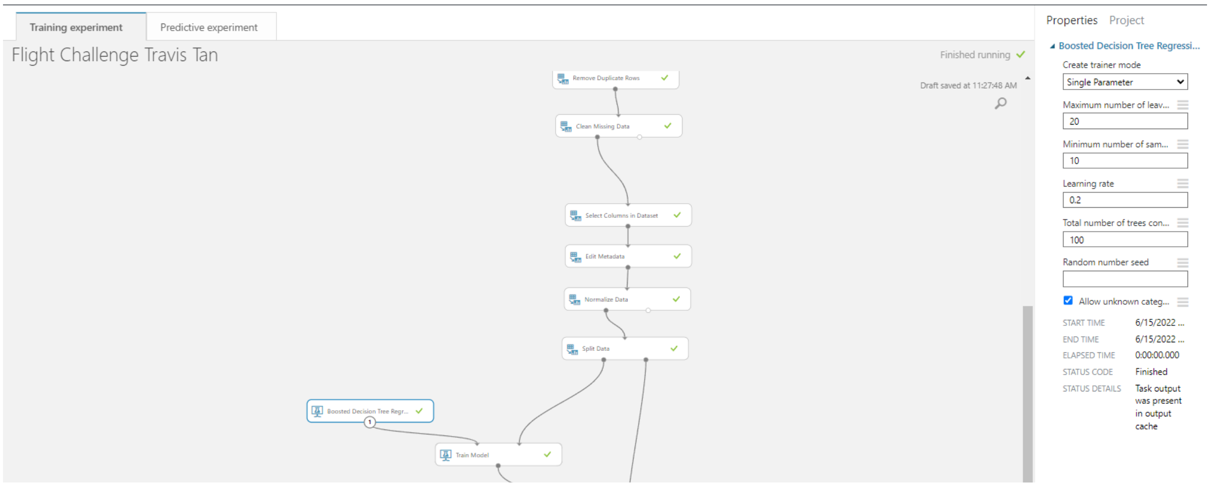
Random seed: 0

Stratified split: False

START TIME 6/15/2022 ...
END TIME 6/15/2022 ...
ELAPSED TIME 0:00:07.064
STATUS CODE Finished
STATUS DETAILS None

View output log

Add a Boosted Decision Tree Regression module and a Train Model module. Then use the default settings to train the model with the 70% data split to predict the ArrDelay label column.



◢ Boosted Decision Tree Regressi...

Create trainer mode: Single Parameter

Maximum number of leav...: 20

Minimum number of sam...: 10

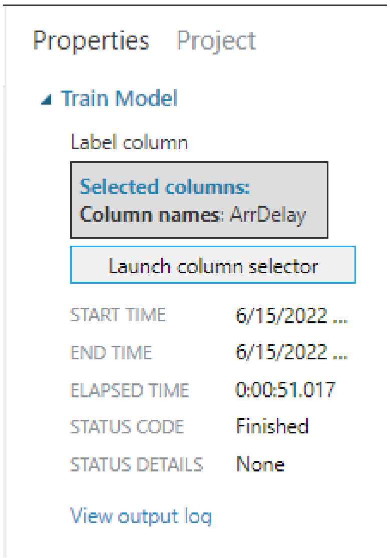
Learning rate: 0.2

Total number of trees con...: 100

Random number seed:

☑ Allow unknown categ...

START TIME 6/15/2022 ...
END TIME 6/15/2022 ...
ELAPSED TIME 0:00:00.000
STATUS CODE Finished
STATUS DETAILS Task output was present in output cache



◢ Train Model

Label column

Selected columns:
Column names: ArrDelay

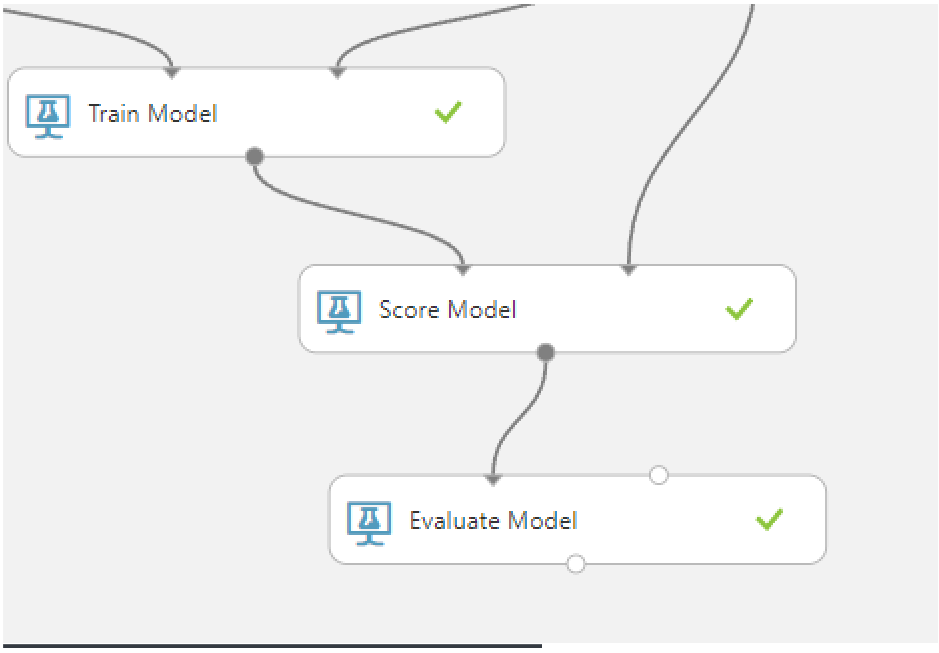
START TIME 6/15/2022 ...
END TIME 6/15/2022 ...
ELAPSED TIME 0:00:51.017
STATUS CODE Finished
STATUS DETAILS None

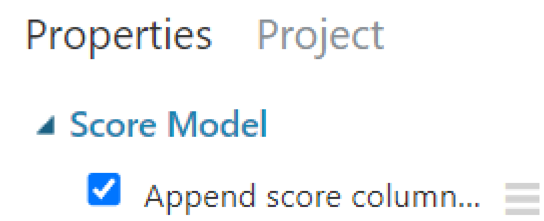
View output log

In the Train model module,

Select ArrDelay column as we will be predicting ArrDelay using AML

Add a Score Model module and use it to score the trained model using the 30% split of data. Use this component to generate predictions using a trained regression model.

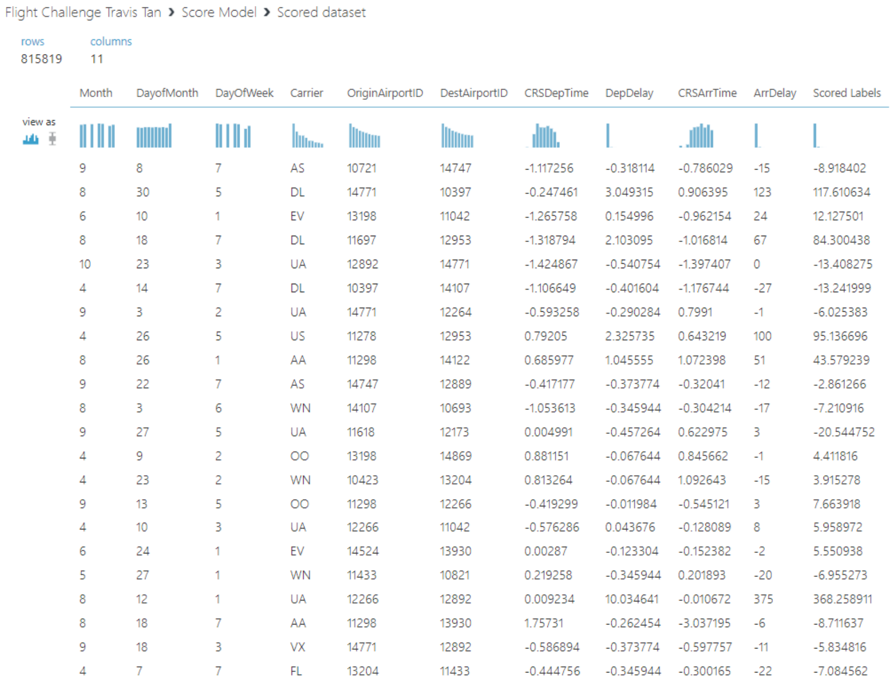




Score Model

☑ Append score column...

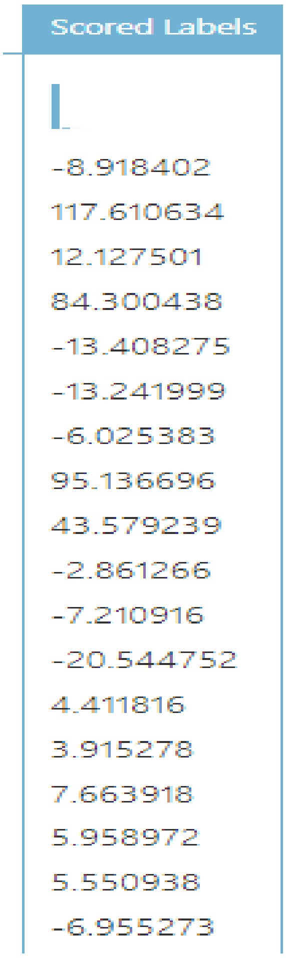
Visualize the results dataset of Score Model



Flight Challenge Travis Tan › Score Model › Scored dataset

rows 815819 columns 11

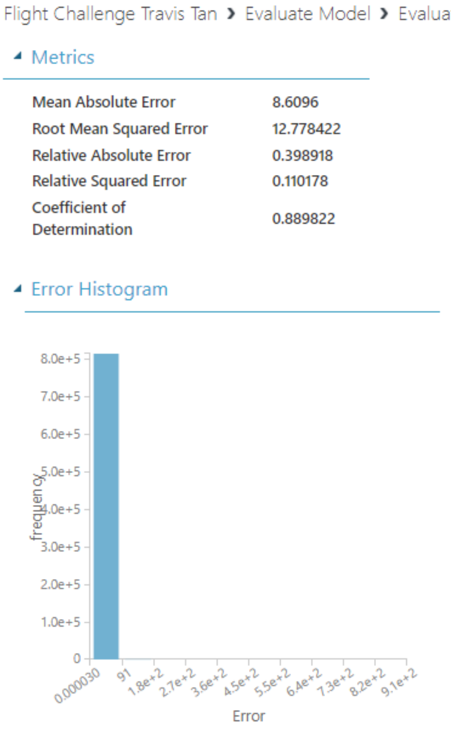
Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	CRSArrTime	ArrDelay	Scored Labels
9	8	7	AS	10721	14747	-1.117256	-0.318114	-0.786029	-15	-8.918402
8	30	5	DL	14771	10397	-0.247461	3.049315	0.906395	123	117.610634
6	10	1	EV	13198	11042	-1.265758	0.154996	-0.962154	24	12.127501
8	18	7	DL	11697	12953	-1.318794	2.103095	-1.016814	67	84.300438
10	23	3	UA	12892	14771	-1.424867	-0.540754	-1.397407	0	-13.408275
4	14	7	DL	10397	14107	-1.106649	-0.401604	-1.176744	-27	-13.241999
9	3	2	UA	14771	12264	-0.593258	-0.290284	0.7991	-1	-6.025383
4	26	5	US	11278	12953	0.79205	2.325735	0.643219	100	95.136696
8	26	1	AA	11298	14122	0.685977	1.045555	1.072398	51	43.579239
9	22	7	AS	14747	12889	-0.417177	-0.373774	-0.32041	-12	-2.861266
8	3	6	WN	14107	10693	-1.053613	-0.345944	-0.304214	-17	-7.210916
9	27	5	UA	11618	12173	0.004991	-0.457264	0.622975	3	-20.544752
4	9	2	OO	13198	14869	0.881151	-0.067644	0.845662	-1	4.411816
4	23	2	WN	10423	13204	0.813264	-0.067644	1.092643	-15	3.915278
9	13	5	OO	11298	12266	-0.419299	-0.011984	-0.545121	3	7.663918
4	10	3	UA	12266	11042	-0.576286	0.043676	-0.128089	8	5.958972
6	24	1	EV	14524	13930	0.00287	-0.123304	-0.152382	-2	5.550938
5	27	1	WN	11433	10821	0.219258	-0.345944	0.201893	-20	-6.955273
8	12	1	UA	12266	12892	0.009234	10.034641	-0.010672	375	368.258911
8	18	7	AA	11298	13930	1.75731	-0.262454	-3.037195	-6	-8.711637
9	18	3	VX	14771	12892	-0.586894	-0.373774	-0.597757	-11	-5.834816
4	7	7	FL	13204	11433	-0.444756	-0.345944	-0.300165	-22	-7.084562



Scored Labels
-8.918402
117.610634
12.127501
84.300438
-13.408275
-13.241999
-6.025383
95.136696
43.579239
-2.861266
-7.210916
-20.544752
4.411816
3.915278
7.663918
5.958972
5.550938
-6.955273

Score Model outputs a predicted value for the flight Arrival Delay (Scored Labels) in minutes. For few cases actual values are less than Predicted values and vice versa and are close to the predicted values.

Add an Evaluate Model module and use it to evaluate the results from the Score Model module. Include screen shots of each task of Activity 10 in the Project Report.



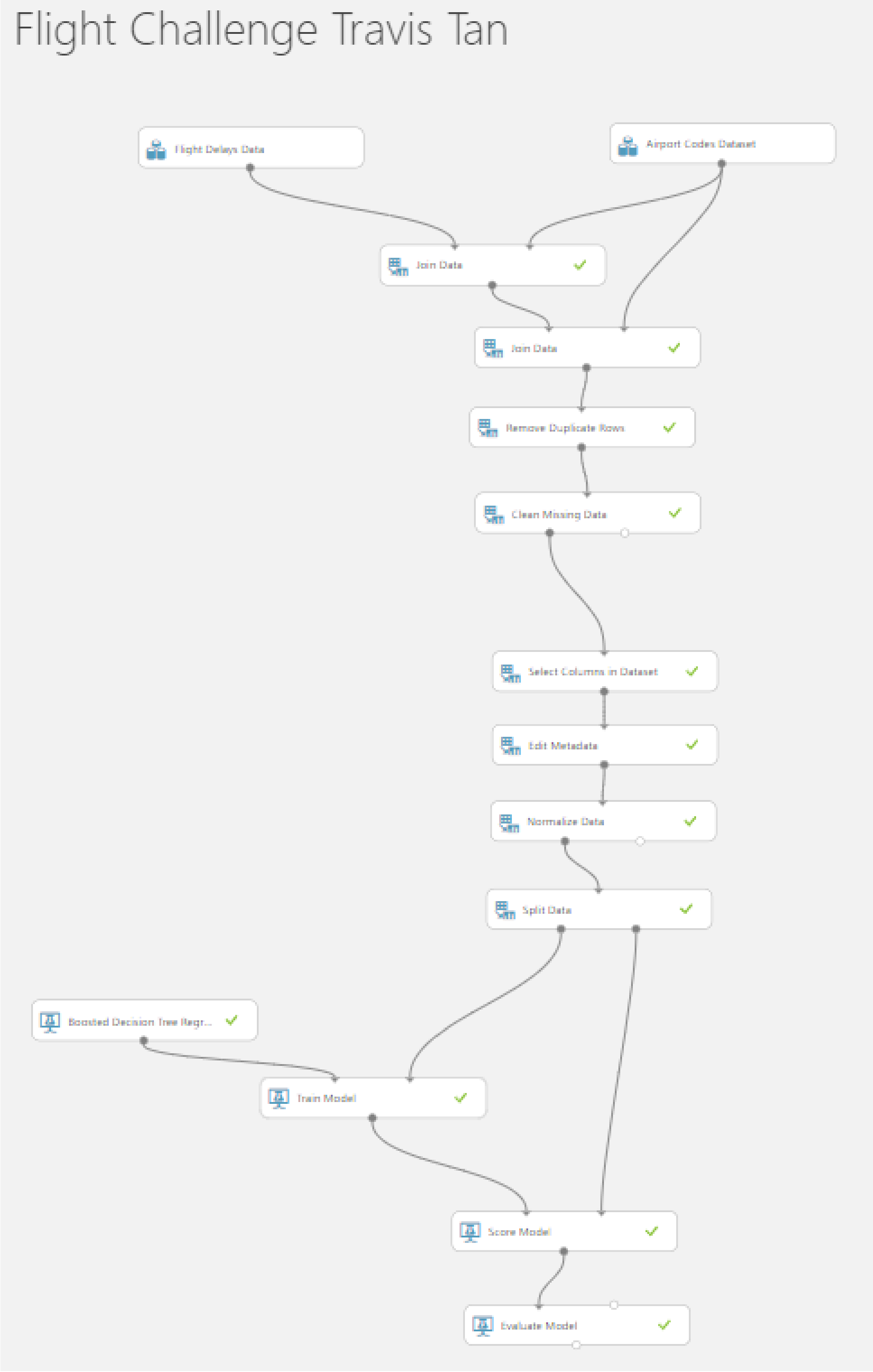
Flight Challenge Travis Tan › Evaluate Model › Evaluation results

Metrics

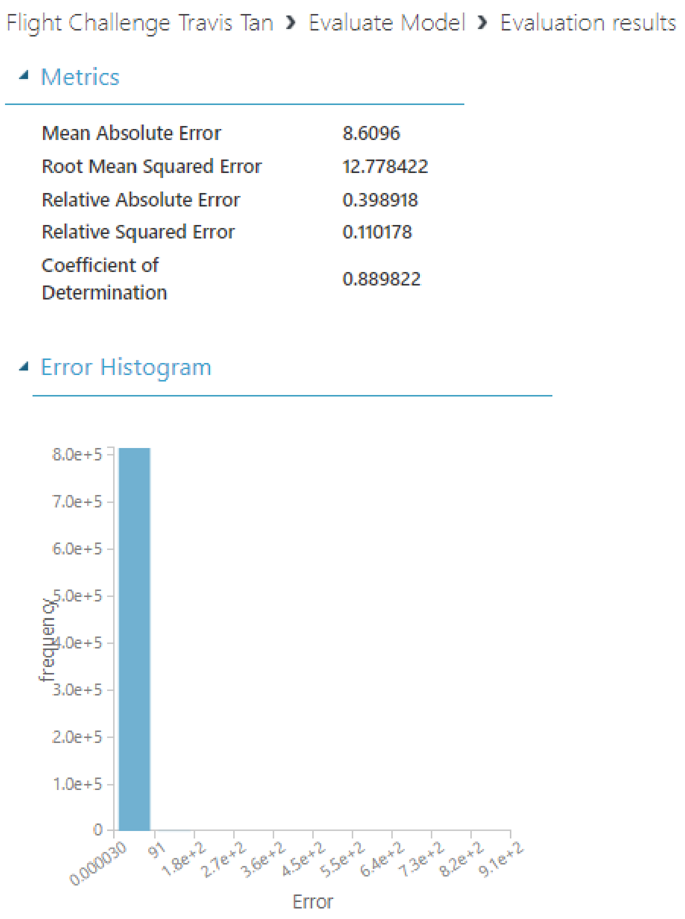
Mean Absolute Error	8.6096
Root Mean Squared Error	12.778422
Relative Absolute Error	0.398918
Relative Squared Error	0.110178
Coefficient of Determination	0.889822

Error Histogram

Overall workspace in Experiment



Activity 11: Test and Evaluate the Model



Flight Challenge Travis Tan › Evaluate Model › Evaluation results

Metrics

Mean Absolute Error	8.6096
Root Mean Squared Error	12.778422
Relative Absolute Error	0.398918
Relative Squared Error	0.110178
Coefficient of Determination	0.889822

Error Histogram

R2 (coefficient of determination) is a statistical measure of how well the regression predictions approximate the real data points. An R2 of 1 indicates that the regression predictions perfectly fit the data. Here COD = 0.8898 which is close to 1 and hence the Model can be regarded as best fit.
Task 3: Determine the Root Mean Squared Error for the model

The Root Mean Squared Error (RMSE), which indicates the mean variance between predicted and actual label values, in this case, the number of minutes on average by which predicted flight delays vary from actual flight delays. Here RMSE = 12.78 minutes which is below 15 minutes is a good measure for this model.

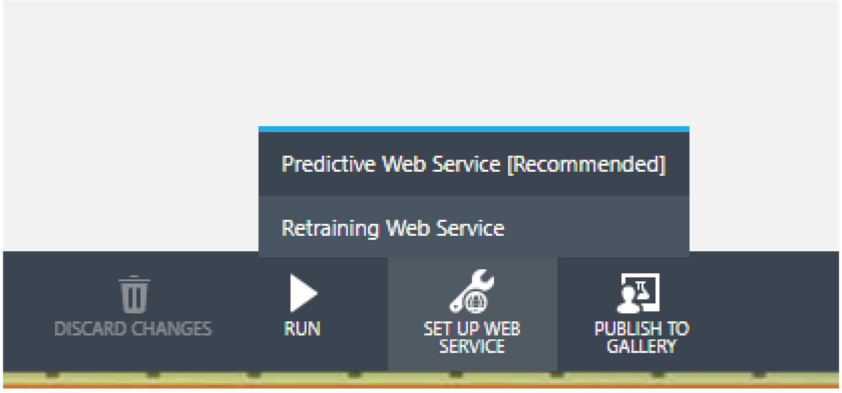
Activity 12: Publish and Use the Model

Performing the following tasks to publish the model as a web service and use the model to predict flight delays. In most cases, the primary objective in creating a machine learning model is to drive actions based on the predictions that it generates. When we have created and refined a model in Azure ML, you can publish it as a web service so that client applications can use it to retrieve predicted labels based on feature inputs. This experiment explores the steps we need to follow to publish and use an Azure ML web service from our model.

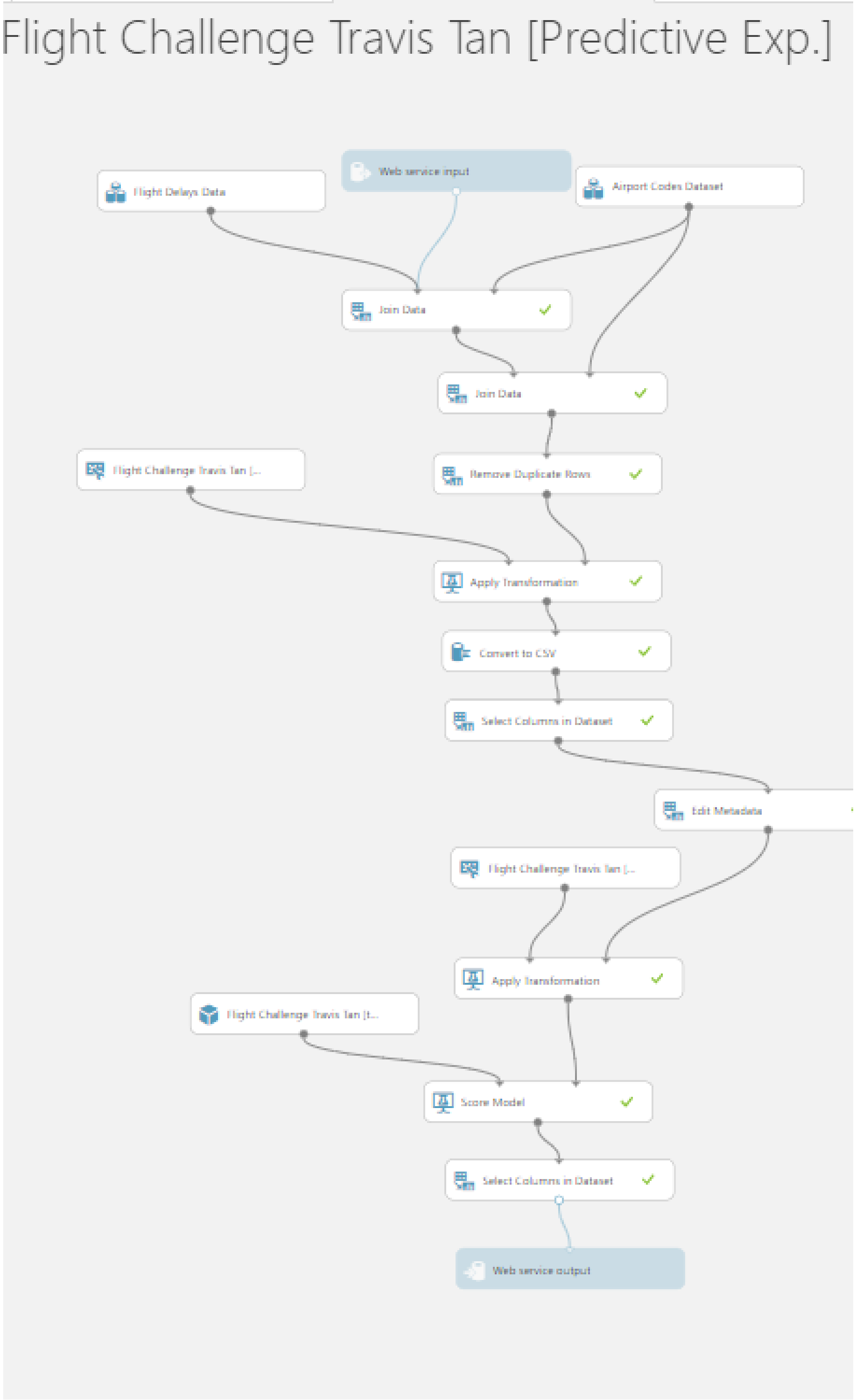
Set up the experiment as a web service, creating a predictive experiment (if the option to do this is not available, save and re-run the experiment).

With the Flight Challenge experiment open, clicked the SET UP WEB SERVICE icon at the bottom of the Azure ML Studio page and clicked Predictive Web Service [Recommended].

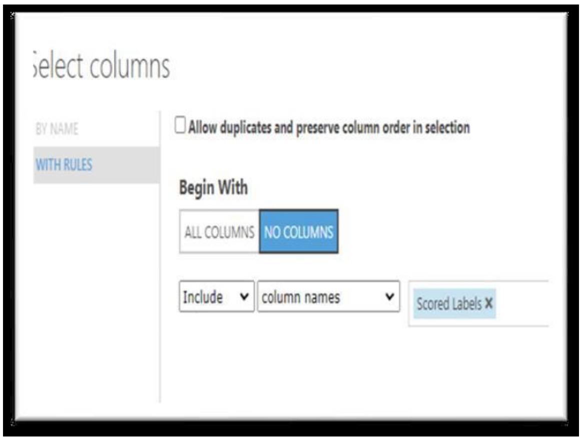
A new Predictive Experiment tab gets automatically created.



A predictive experiment will be created and required modules will be added. We then organize and arrange the experiment such that it looks neat.


Flight Challenge Travis Tan [Predictive Exp.]
Flight Delays Data
Web service input
Airport Codes Dataset
Join Data
Join Data
Flight Challenge Travis Tan [...
Remove Duplicate Rows
Apply Transformation
Convert to CSV
Select Columns in Dataset
Edit Metadata
Flight Challenge Travis Tan [...
Apply Transformation
Flight Challenge Travis Tan [t...
Score Model
Select Columns in Dataset
Web service output

In the predictive experiment, connection between the Score Model module and the Web service output module is deleted.



Select columns

BY NAME

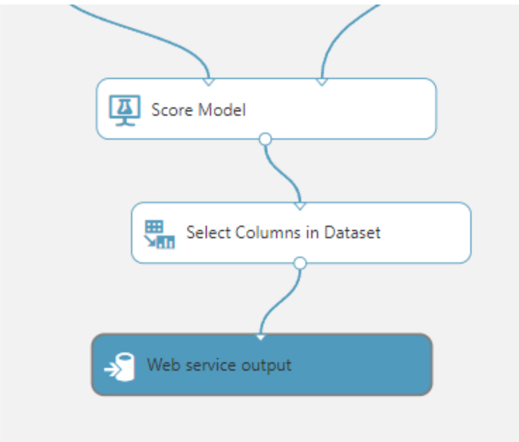
WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS | NO COLUMNS

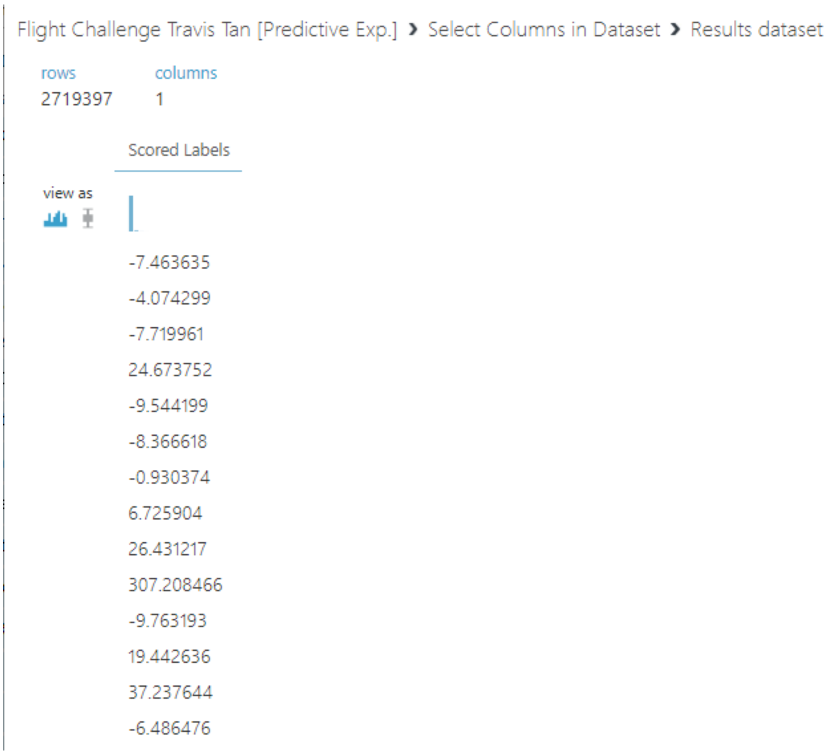
Include | column names | Scored Labels ×



Add a Select Columns in Dataset module and place it between the Score Model and Web service output modules. Use the module to select only the Scored Labels column. This ensures that when the web service is called, only the predicted value (Scored Labels) is returned.

Ensure that the predictive experiment now looks like the following, and then save and run the predictive experiment.

When the experiment has finished running, visualize the output of the last Select Columns in Dataset module and verify that only the Scored Labels column is returned.



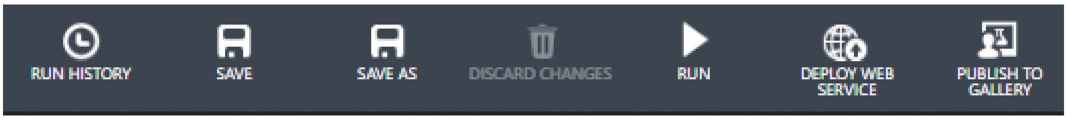
Flight Challenge Travis Tan [Predictive Exp.] › Select Columns in Dataset › Results dataset

rows	columns
2719397	1

view as

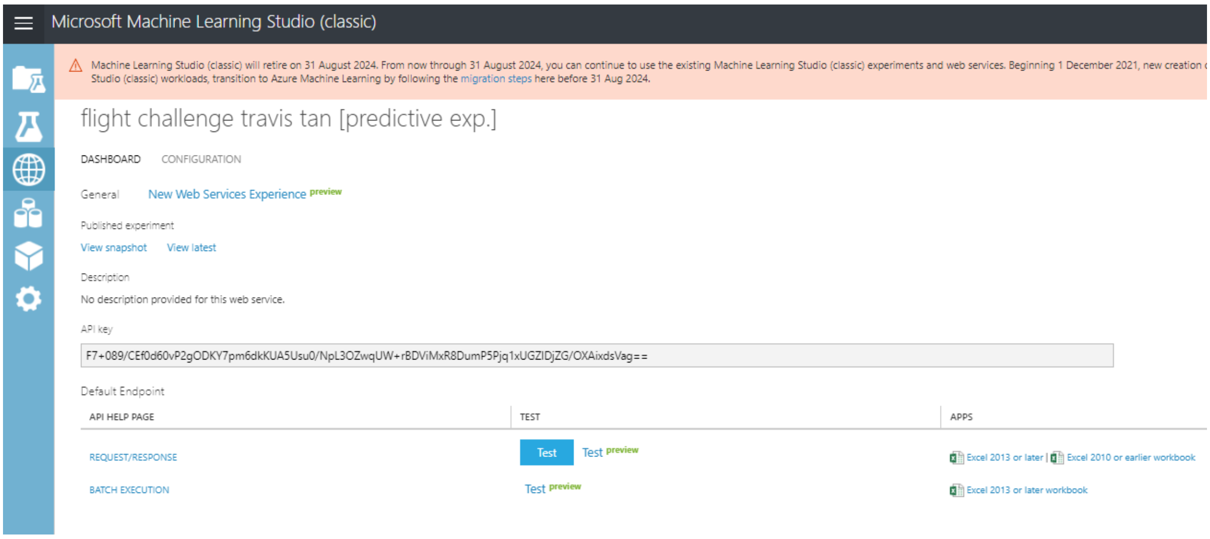
Scored Labels
-7.463635
-4.074299
-7.719961
24.673752
-9.544199
-8.366618
-0.930374
6.725904
26.431217
307.208466
-9.763193
19.442636
37.237644
-6.486476

Deploy the web service.



Machine Learning Studio (classic) deploys the experiment as a web service and takes you to the dashboard for that web service.

Wait for few seconds for the dashboard page to appear and note the API key and Request/Response link. You will use these to connect to the web service from a client application.



⚠ Machine Learning Studio (classic) will retire on 31 August 2024. From now through 31 August 2024, you can continue to use the existing Machine Learning Studio (classic) experiments and web services. Beginning 1 December 2021, new creation
Studio (classic) workloads, transition to Azure Machine Learning by following the migration steps here before 31 Aug 2024.

flight challenge travis tan [predictive exp.]

DASHBOARD CONFIGURATION

General New Web Services Experience preview

Published experiment

View snapshot View latest

Description

No description provided for this web service.

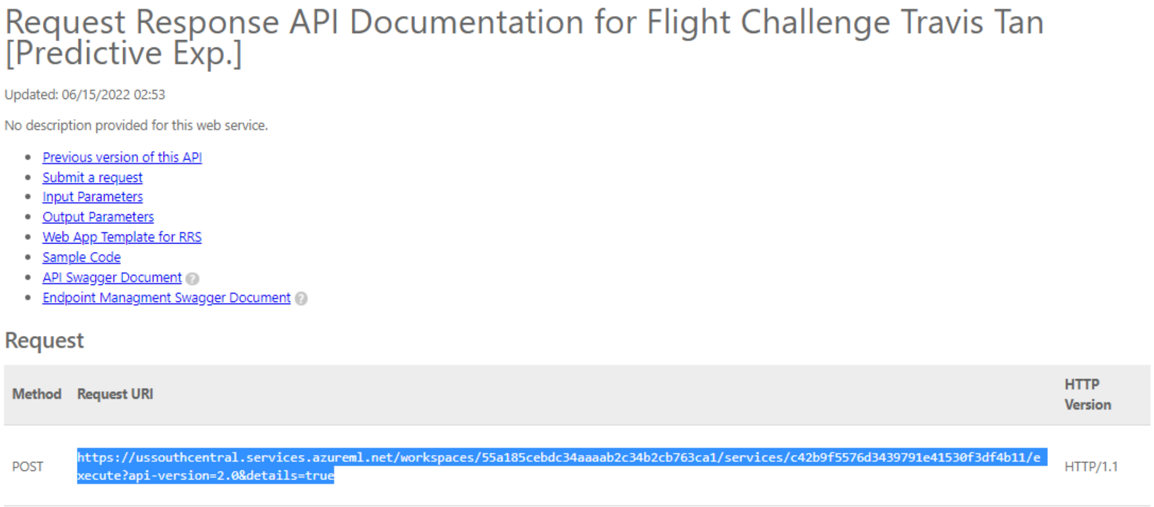
API key

F7+089/CEf0d60vP2gODKY7pm6dkKUA5Usu0/NpL3OZwqUW+rBDViMxR8DumP5Pjq1xUGZIDjZG/OXAixdsVag==

Default Endpoint

API HELP PAGE	TEST	APPS
REQUEST/RESPONSE	Test Test preview	Excel 2013 or later | Excel 2010 or earlier workbook
BATCH EXECUTION	Test preview	Excel 2013 or later workbook

Request Response URL



Request Response API Documentation for Flight Challenge Travis Tan [Predictive Exp.]

Updated: 06/15/2022 02:53

No description provided for this web service.

- Previous version of this API
- Submit a request
- Input Parameters
- Output Parameters
- Web App Template for RRS
- Sample Code
- API Swagger Document
- Endpoint Managment Swagger Document

Request

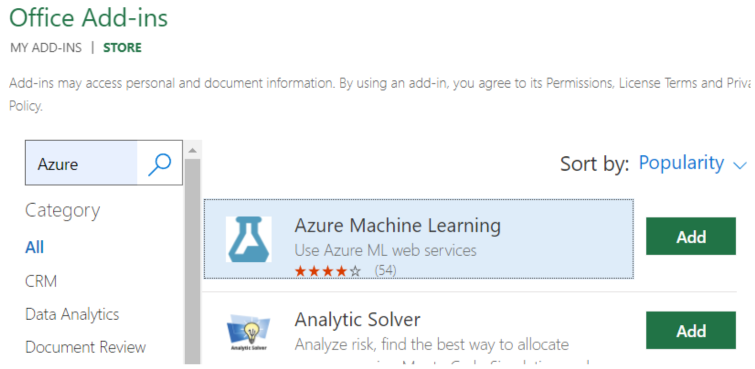
Method	Request URI	HTTP Version
POST	https://ussouthcentral.services.azureml.net/workspaces/55a185cebdc34aaaab2c34b2cb763ca1/services/c42b9f5576d3439791e41530f3df4b11/execute?api-version=2.0&details=true	HTTP/1.1

API Key :
F7+089/CEf0d60vP2gODKY7pm6dkKUA5Usu0/NpL3OZwqUW+rBDViMxR8DumP5Pjq1xUGZlDjZG/OXAixdsVag==

Request/Response link :
https://ussouthcentral.services.azureml.net/workspaces/55a185cebdc34aaaab2c34b2cb763ca1/services/c42b9f5576d3439791e41530f3df4b11/execute?api-version=2.0&details=true

Excel Spreadsheet is opened and on the Insert tab, click Office Add-ins.

Then in the Office Add-ins dialog box, select Store, search for Azure Machine Learning, and add the Azure Machine Learning add-in as shown below



Office Add-ins

MY ADD-INS | STORE

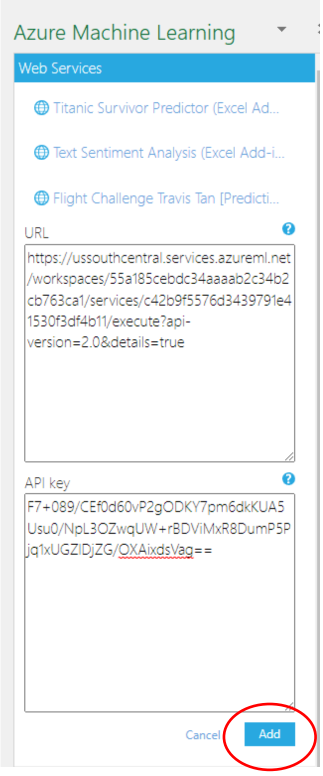
Add-ins may access personal and document information. By using an add-in, you agree to its Permissions, License Terms and Priva
Policy.

After the add-in is installed, in the Azure Machine Learning pane on the right of the Excel workbook, click Add Web Service. Boxes for the URL and API key of the web service will appear.

On the browser tab containing the dashboard page for your Azure ML web service, right-click the Request/Response link you noted earlier and copy the web service URL to the clipboard. Then return to the browser tab containing the Excel Online workbook and paste the URL into the URL box.

On the browser tab containing the dashboard page for your Azure ML webservice, click the Copy button for the API key you noted earlier to copy the key to the clipboard. Then return to the browser tab containing the Excel Online workbook and paste it into the API key box.

Verify that the Azure Machine Learning pane in your workbook now resembles this and click Add.



Azure Machine Learning
Web Services
Titanic Survivor Predictor (Excel Ad...
Text Sentiment Analysis (Excel Add-i...
Flight Challenge Travis Tan [Predicti...

URL

https://ussouthcentral.services.azureml.net/workspaces/55a185cebdc34aaaab2c34b2cb763ca1/services/c42b9f5576d3439791e41530f3df4b11/execute?api-version=2.0&details=true

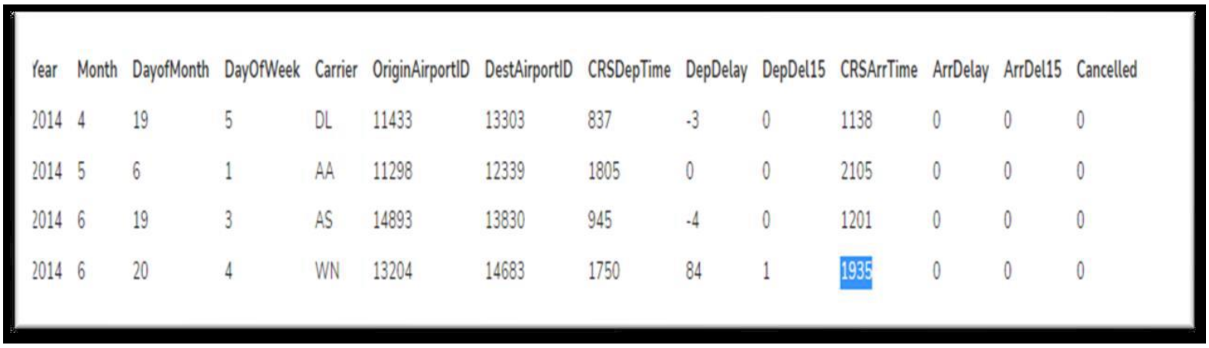
API key

F7+089/CEf0d60vP2gODKY7pm6dkKUA5Usu0/NpL3OZwqUW+rBDViMxR8DumP5Pjq1xUGZIDjZG/OXAixdsVag==

Insert URL and API key followed by clicking the Add button. This will create a predictive model on the excel spreadsheet.

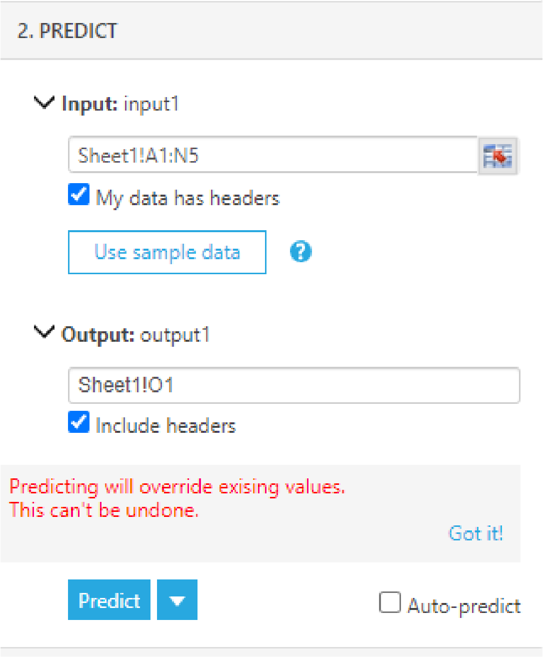
In the Excel worksheet select cell A1.Then in the Azure Machine Learning pane, collapse the 1. View Schema section and in the 2. Predict section, click Use sample data. this enters some sample input values in the worksheet.

Modify the sample data in row 2 as follows:



Year	Month	DayofMonth	DayOfWeek	Carrier	OriginAirportID	DestAirportID	CRSDepTime	DepDelay	DepDel15	CRSArrTime	ArrDelay	ArrDel15	Cancelled
2014	4	19	5	DL	11433	13303	837	-3	0	1138	0	0	0
2014	5	6	1	AA	11298	12339	1805	0	0	2105	0	0	0
2014	6	19	3	AS	14893	13830	945	-4	0	1201	0	0	0
2014	6	20	4	WN	13204	14683	1750	84	1	1935	0	0	0

Select the cells containing the input data (cells A1 to N5), and in the Azure Machine Learning pane, click the button to select the input range and confirm that it is 'Sheet1'!A1:N5.



2. PREDICT

Input: input1

Sheet1!A1:N5

☑ My data has headers

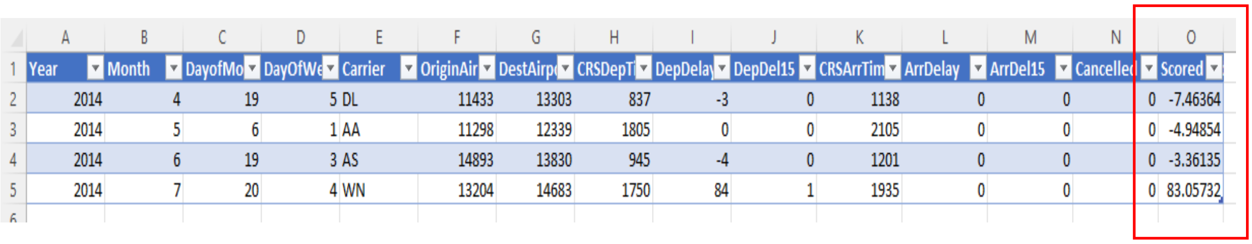
Output: output1

Sheet1!O1

☑ Include headers

Predicting will override exising values.
This can't be undone.

Ensure that the My data has headers box is checked.
In the Output box type O1 and ensure the Include headers box is checked.
Click the Predict button, and after a few seconds, view the predicted label in cell O2.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Year	Month	DayofMo	DayOfWe	Carrier	OriginAir	DestAirp	CRSDepTi	DepDelay	DepDel15	CRSArrTim	ArrDelay	ArrDel15	Cancelled	Scored
2	2014	4	19	5	DL	11433	13303	837	-3	0	1138	0	0	0	-7.46364
3	2014	5	6	1	AA	11298	12339	1805	0	0	2105	0	0	0	-4.94854
4	2014	6	19	3	AS	14893	13830	945	-4	0	1201	0	0	0	-3.36135
5	2014	7	20	4	WN	13204	14683	1750	84	1	1935	0	0	0	83.05732
6															

Scored Label (predicted ArrDelay) is calculated for multiple rows.
For few rows actual is close to predicted and for few actual is different from predicted values.

```r
## Activity 6
#Look at summary of all columns in the dataset
summary(Flight_Challenge_Results_dataset)

# Explore ArrDelay Data
summary(Flight_Challenge_Results_dataset$ArrDelay)
sd(Flight_Challenge_Results_dataset$ArrDelay)

# ArrDelay Summary result
# Min = -94 Mean = 6.567 SD = 38.44812 Max = 1845

## Activity 7
# load ggplot and gridExtra for plotting
library("ggplot2")
library("gridExtra")
install.packages("ggplot2")
install.packages("gridExtra")

# Determine range values for ArrDelay
range(Flight_Challenge_Results_dataset$ArrDelay)

# Set plot properties and Plot Histogram with Box Plot
options(repr.plot.width = 6, repr.plot.height = 3)

# Calculate binwidth for histogram
rg = range(Flight_Challenge_Results_dataset['ArrDelay'])
rg

# Each Bin is seperated by a value of 64.63333
bw = (rg[2] - rg[1])/30
bw

# Plot Histogram
p1 = ggplot(Flight_Challenge_Results_dataset, aes(ArrDelay)) +
        geom_histogram(bins = 30, fill = "blue") +
        labs(x = "ArrDelay", y = "Count of Flights", title = "Histogram of ArrDelay") +
        theme_bw()

p1

# Plot boxplot
p2 = ggplot(Flight_Challenge_Results_dataset, aes(x = factor(0), y = ArrDelay)) +
       geom_boxplot(color = "blue") +
       ggtitle("Boxplot of ArrDelay1") +
       theme_bw()
p2

# Arrange the Histogram and Box plot on same row
grid.arrange(p1,p2,nrow=1)
```

```r
## Activity 8: Histograms to compare Numeric Columns

# Create function to plot conditioned histograms
# ggplot2 and gridExtra have been loaded and installed. Plot have been set to appropiate width and height

arrdel15.hist = function(x){
        library(ggplot2)
        library(gridExtra)
        options(repr.plot.width=6, repr.plot.height=3)
        title = paste("Histogram of", x, "conditioned on ArrDel15")
        ## Create the histogram
        ggplot(Flight_Challenge_Results_dataset, aes_string(x)) +
        geom_histogram(aes(y = ..count..), bins = 30, fill = "red") +
        facet_grid(. ~ ArrDel15) +
        ggtitle(title) +
        ylab("Count of ArrDelay") +
        theme_bw()
}

## Create histograms for specified features.
plot.cols2 = c("DepDelay",
         "CRSArrTime",
         "CRSDepTime",
         "DayofMonth",
         "DayOfWeek",
         "Month")

lapply(plot.cols2, arrdel15.hist)


## Activity 9: Use Scatter Plots to Compare Numeric Columns
# Scatter plot using color to differentiate points

scatter_flights = function(x){
         library(ggplot2)
         library(gridExtra)
         options(repr.plot.width = 6, repr.plot.height = 3)
         title = paste("ArrDelay vs.", x, "with color by ArrDel15")
         ggplot(Flight_Challenge_Results_dataset, aes_string(x, 'ArrDelay')) +
           geom_point(aes(color = factor(ArrDel15))) +
           ggtitle(title) +
           theme_bw()
}
```

```
# Define columns for scatter plot

plot.col3 = c("DepDelay",
            "CRSArrTime",
            "CRSDepTime",
            "DayofMonth",
            "DayOfWeek",
            "Month")

lapply(plot.col3, scatter_flights)
```