AI / ML Workshop

Lab 3 - Introduction to Amazon Machine Learning

June 2018

Table of Contents

Overview

Amazon Machine Learning is an easy to use and robust platform for developers, analysts, and business domain experts of all skill levels. This service allows users to build and train predictive models, using user-friendly visualization tools and wizards, without having to learn complex machine learning algorithms and technology. Users can focus on experimenting and improving models from convert ideas to facts, without worrying about operating and maintaining a highly scalable and reliable infrastructure. Amazon Machine Learning offers a pay-as-you-go model where no upfront hardware or software investment is required, and it has the capability to scale with demand to billions of predictions.

In this lab, we will walk you through creating, training, improving and finally using the machine learning models to perform real-time predictions with Amazon Machine Learning. For the purpose of the tutorial, we will work with the bike sharing dataset from UCI Machine Learning Repository. The sample dataset contains data generated by bike sharing systems, where bikes can be rented via a network of kiosk locations. The goal for this lab is to forecast bike rental demand based on historical usage pattern.

The steps we need to follow to build a machine learning model include:

1. Download the bike sharing dataset and upload it into an Amazon S3 bucket
2. Create a new datasource in the Amazon Machine Learning console using the dataset in the S3 bucket
3. Define a schema on the datasource and build a predictive model
4. Make real-time predictions using the model trained

Download and Review the Bike Sharing Dataset

Before we start training the data, let's review it first by downloading the biking sharing dataset from here. Open the **bike_share_data.csv** file with a text editor such as Notepad. When working with machine learning, majority of the time is spent in gathering, preparing, cleaning and analyzing the data before training it with machine learning. For example, you should look for features or information that have higher predictive power. You should also consider how to deal with missing values and outliers in data. It is also recommended to randomly shuffle the data, especially for online machine learning algorithm to get a better performing model. It is an iterative process to develop a successful predictive model. The dataset from UCI is already cleaned and formatted, we also have downloaded and shuffled the dataset in the interest of time and keeping the focus on the Amazon Machine Learning service.

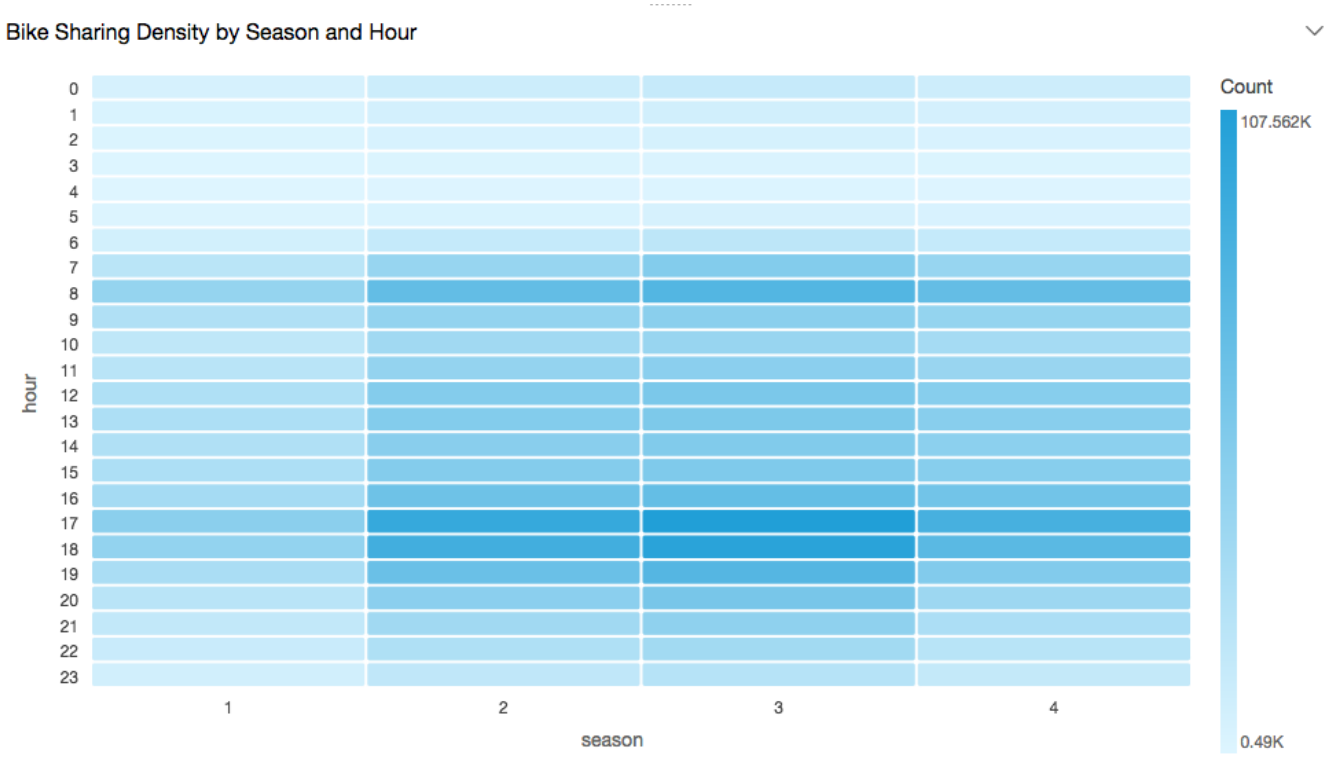
The first line in the dataset is column headers and they represent the following –

Column Name	**Description**
season	1-Spring, 2-Summer, 3-Fall, 4-Winter
holiday	1-Yes, 0-No
weekday	0-Sunday, 1-Monday, 2-Tuesday, 3-Wednesday, 4-Thursday, 5-Friday, 6-Saturday
workingday	1-Yes, 0-No
weather	1-Clear, Few clouds, Partly cloudy, Partly cloudy 2-Misty and Cloudy, Misty with Broken clouds, Misty with Few clouds, Misty 3-Light Snow, Light Rain and Thunderstorm, Light Rain and Scatter clouds 4-Heavy Rain and Ice Pallets, Thunderstorm, Snow and Fog
temp	Normalized temperature in Celsius
atemp	Normalized feeling temperature in Celsius
humidity	Normalized humidity
count	Count of total rental bikes aggregated in one hour
datetime	The hour and date

Difference between BI and Machine Learning

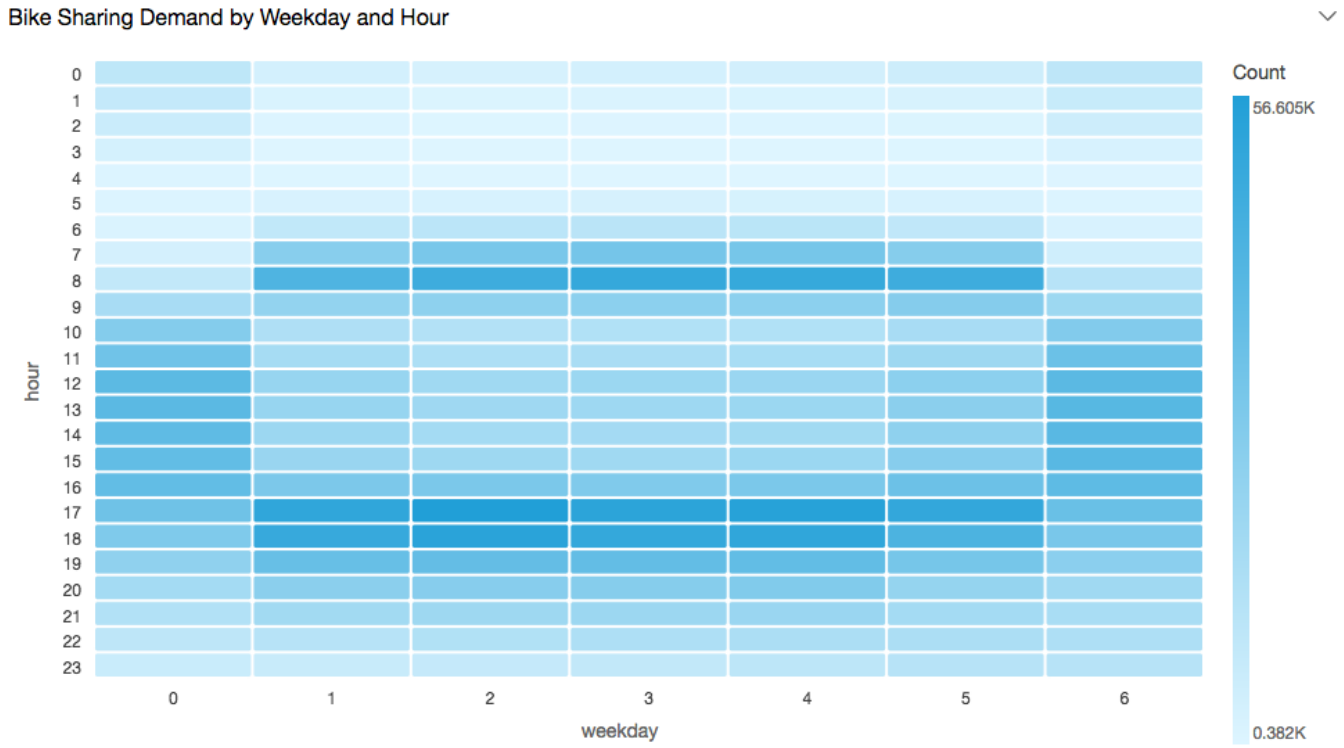
Traditionally, business typically analyze and discover trends based on Business Intelligence (BI) Tools with many different reports and charts. This is still useful to analyze and identifying trends. For this lab, we leveraged Amazon QuickSight to visualize and discover usage patterns in the dataset. Amazon QuickSight is a fast business analytics service you can use to build visualizations perform ad-hoc analysis, and quickly gain business insights from your data.

In the screenshot below, we can see that demands are higher during summer (season=2) and fall (season=3) seasons. We can also see that peak time rentals are around 7-8 a.m. and 4-6 p.m.

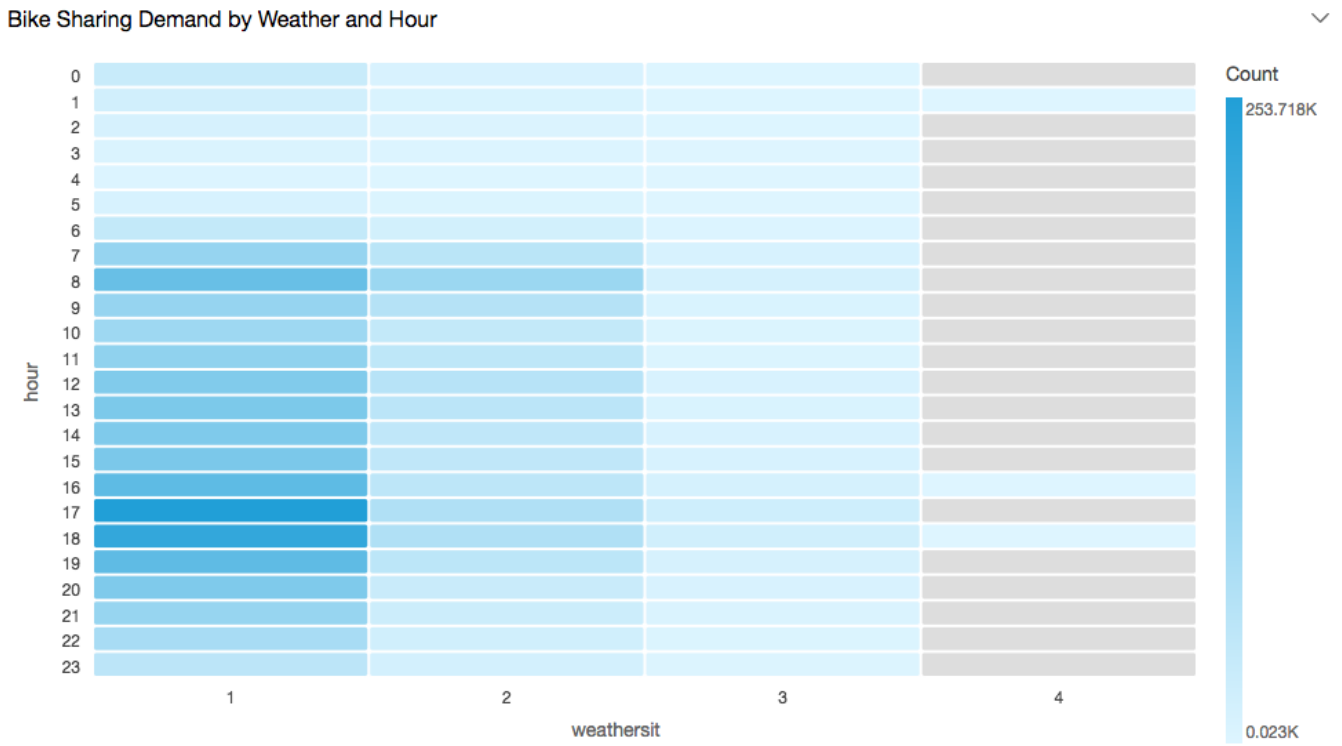


Let’s take a look at a different chart. In the screenshot below, we can see that during the work days (weekday=1-5), peak times are 7-8 a.m. and 5-6 p.m. And during the weekends

(weekday=0 and 6), peak times are between 10 a.m. – 5 p.m.



Let’s look at another chart. In the screenshot below, we can see that when weather is sunny and clear (weathersit=1), we have a healthy usage. But when there is heavy rain, thunderstorm or snow (weathersit=4), we see there is hardly any usage.

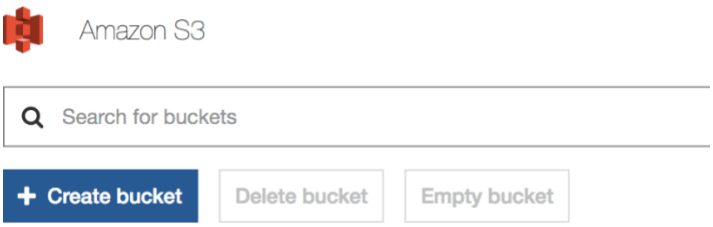


Using a BI tool such as Amazon QuickSight provides the ability to quickly analyze and discover trends in the dataset. This is very helpful; however, it is hard to provide a specific answer given a set of parameters such as specific date and time, weather forecast, day of the week, season and etc in an automated fashion. Machine learning is able to find a pattern in the dataset and predict an answer given the same set of parameters. In the following sections, we will leverage Amazon Machine Learning service to find answers.

Download the Dataset and Upload to Amazon S3

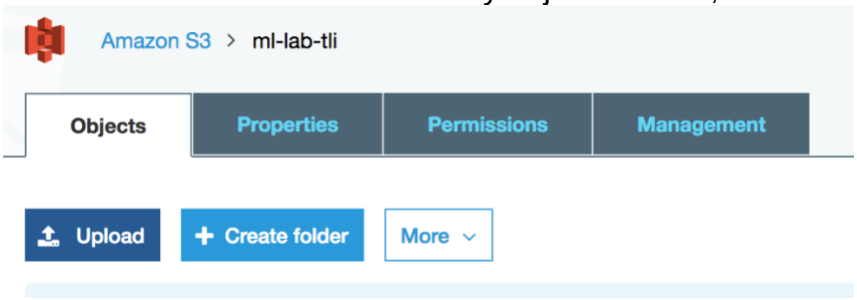
In this section, we will download the bike sharing datasets and load them into an Amazon S3 bucket. Amazon Machine Learning currently supports S3, Redshift and Relational Database Service (RDS) as the datasource.

1. If you have not downloaded the bike sharing dataset yet, be sure to download it from here.
2. Sign into the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3
3. In the upper-right corner of the AWS Management Console, confirm you are in the desired AWS region (e.g., Oregon).
4. Now, we will need to create a bucket. In the S3 console, click the **Create Bucket** button.

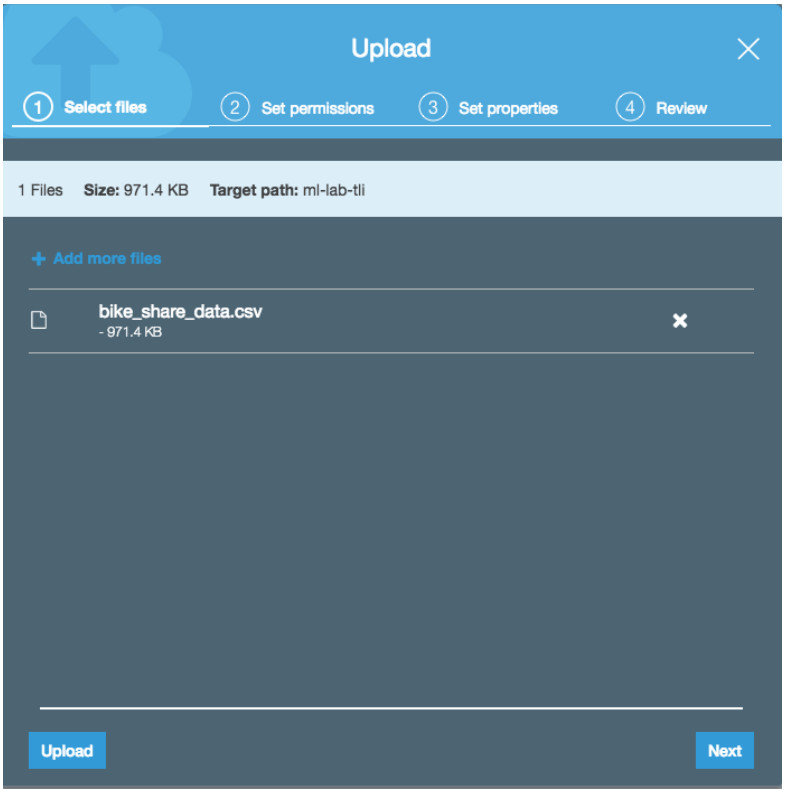


5. For the **Bucket Name**, type “ml-lab-**<your-initials>**” in the text box and click **Next** (take note of the bucket name, it will be needed later for creating the datasource). Leave everything default in the next 2 pages and click **Create Bucket** in **Review** page.

6. Click the link for the bucket name you just created, then click **Upload.**



7. Click **Add Files**, find and select the **bike_share_data.csv** file and click **Upload**.



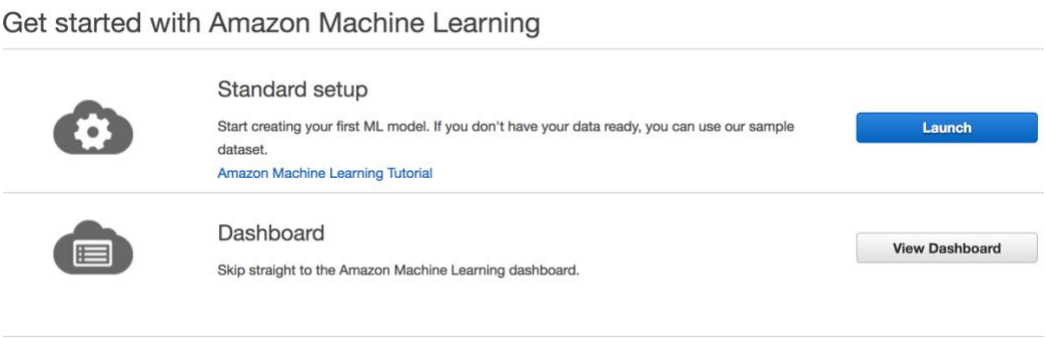
Create a Datasource and a Machine Learning Model

In this section, we will create the datasource needed to train the model with Amazon ML.

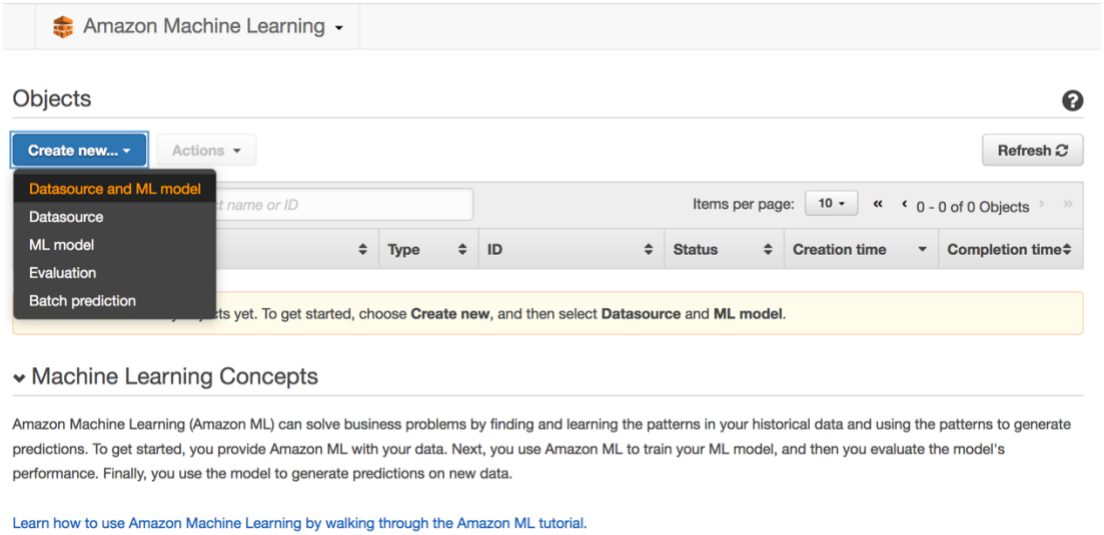
1. Open the Amazon Machine Learning console at https://console.aws.amazon.com/machinelearning/home

2. If you see the Getting Started page, click on **Get started** button. If you do not see it, go to step 4.

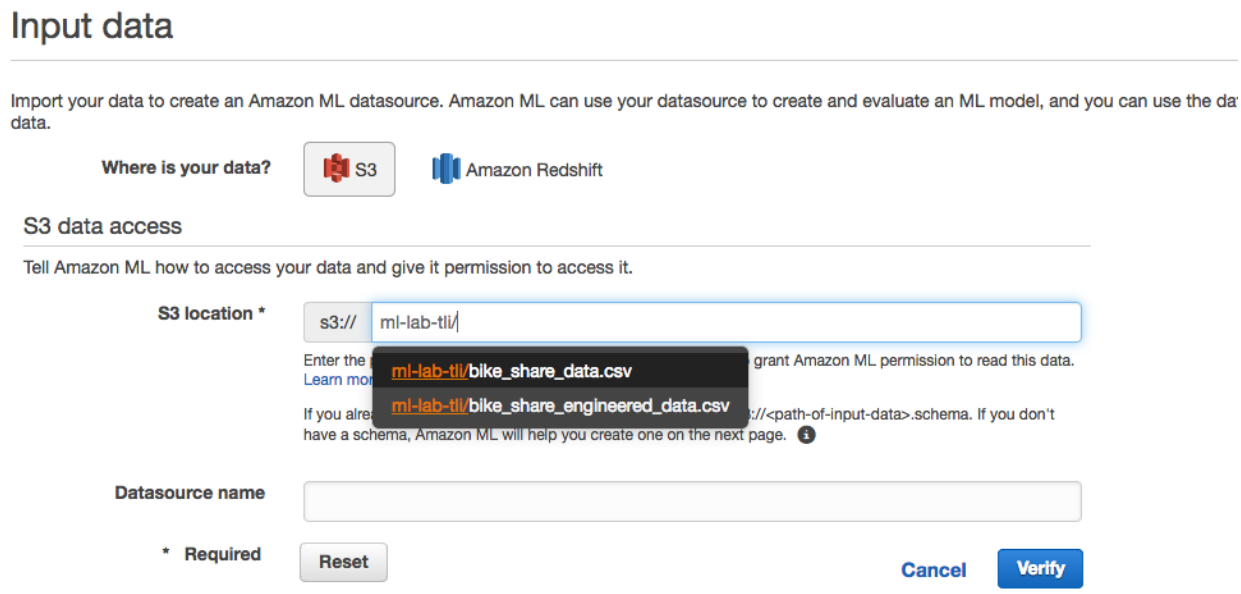
3. Click on **View Dashboard** button



4. Click on **Create New…** button and select "**Datasource and ML Model**"

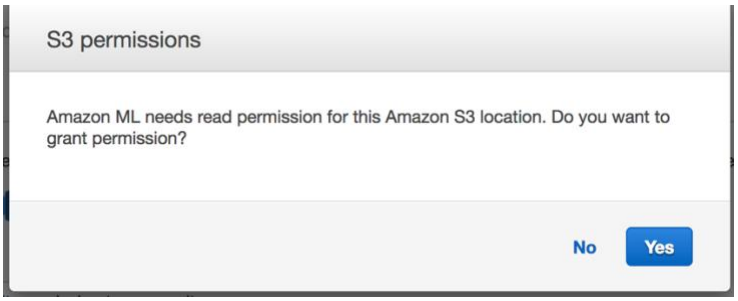


5. Under "**Where is you data?**" section, make sure the S3 icon is selected. In the S3 location text box, input the bucket name "ml-lab-<**your-initials>**". Auto-complete will prompt a list of matches. Select the one that you created. Auto-complete will then prompt you with list of files in the bucket, select **bike_share_data.csv** from the list.

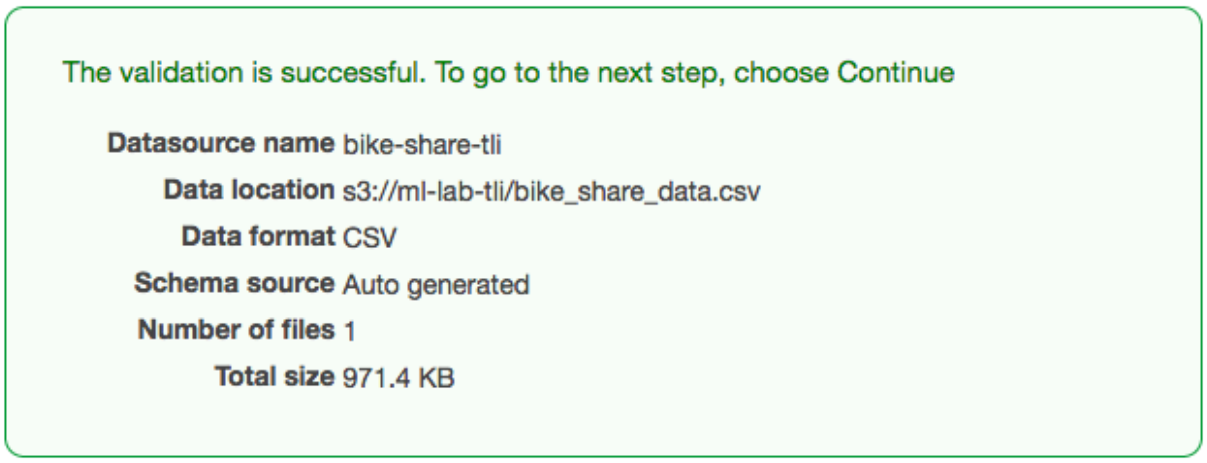


6. In the Datasource name text box, input "bike-share-<**your-initials>**". Write down the Datasource name as you will need it later to create the model.

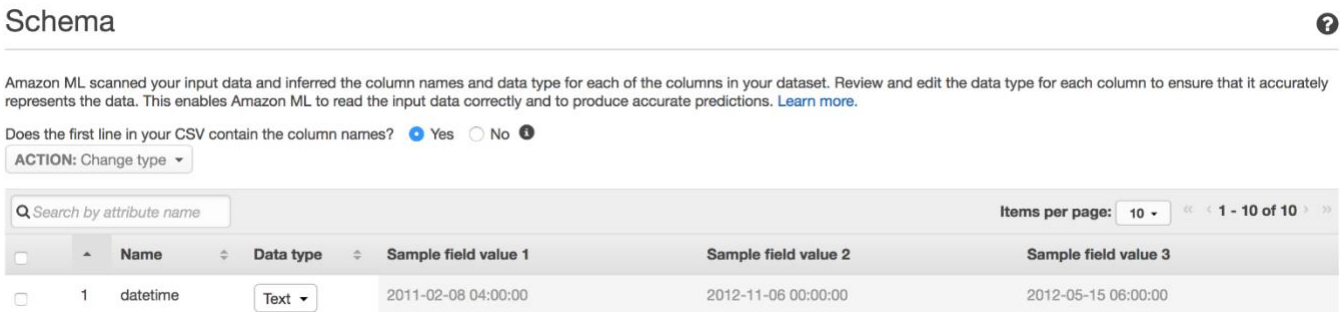
7. Click the **Verify** button. When prompted, give Amazon ML access to your S3 bucket by clicking **Yes**.



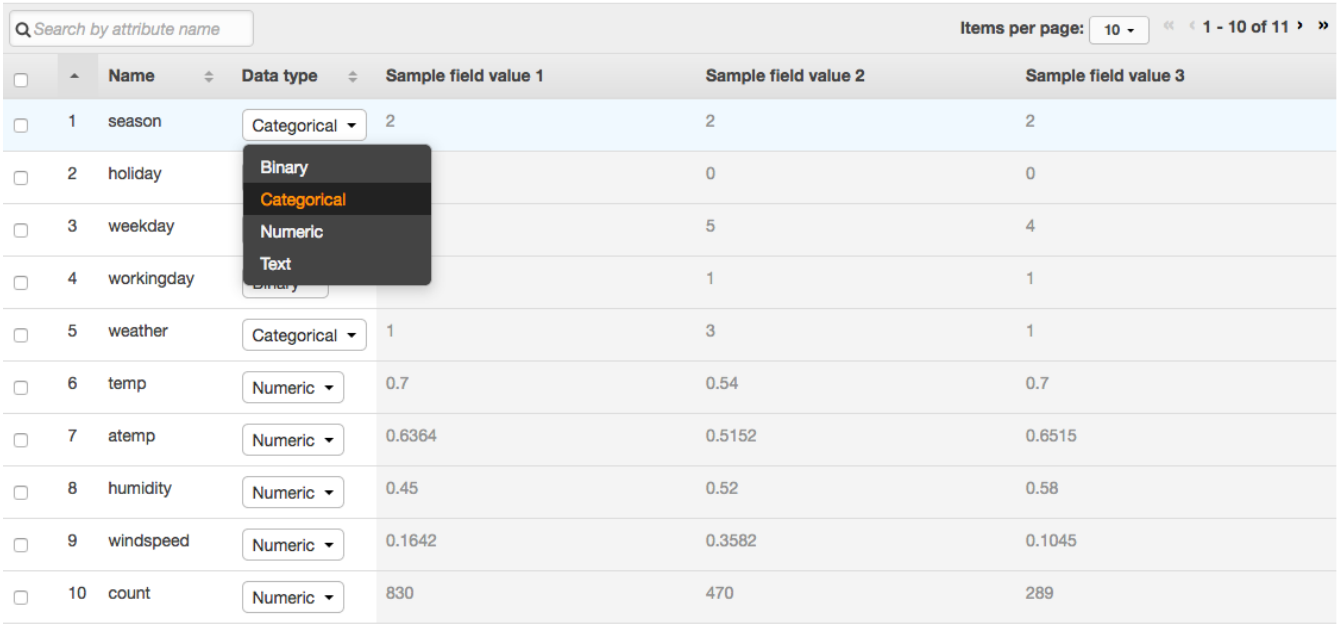
8. You should see a message indicating validation is successful. Click the **Continue** button to move to **Schema** page.



9. Select the **Yes** radio button in the "**Does the first line in your CSV contain the column names?**" section. The page will refresh and display column names.

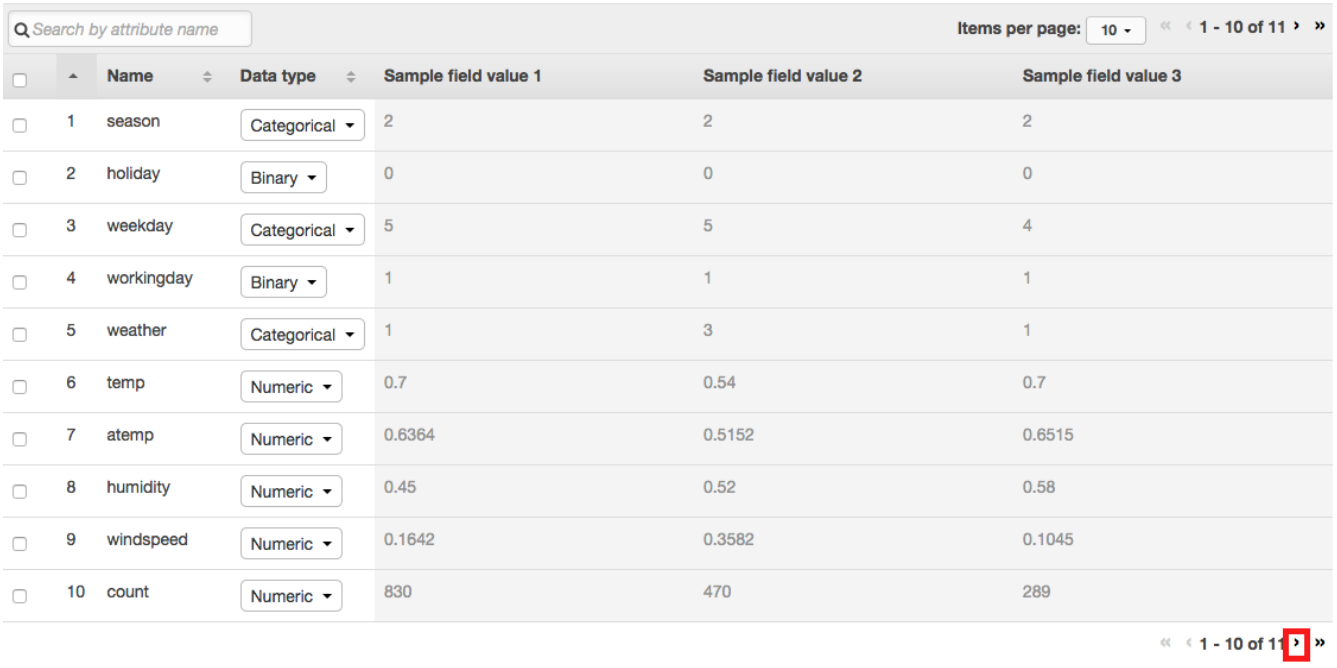


10. Amazon Machine Learning will automatically classify the data it detects into one of four categories - Text, Categorical, Binary, and Numeric. Review the inferred data types and verify they are valid. For example, **season, weekday,** and **weather** are encoded as numbers, hence Amazon Machine Learning will infer as such. While Amazon Machine Learning does a good job on handling the data type even when the scheme is not completely accurate; however, if you know the data type, it is best to define it as such. In the **season, weekday,** and **weather** column, change **Data type** from **Numeric** to **Categorical.** With numeric data type, order and distance have typically have a meaning, where categorical data type does not.

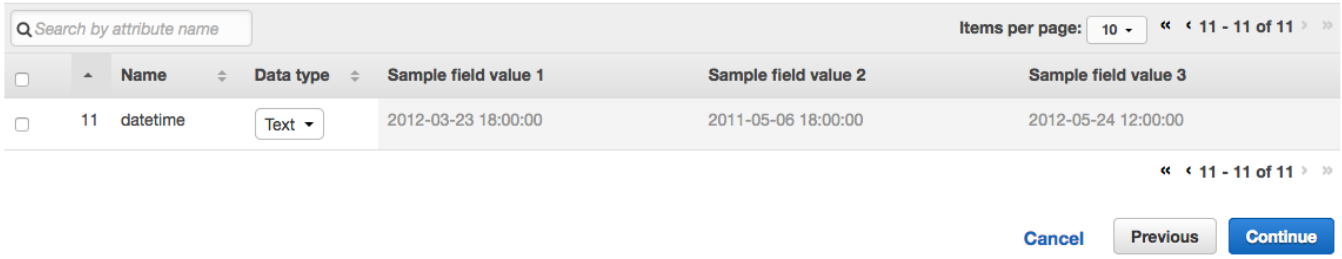


11.

12. Make sure the schemas definition matches the following screenshot then click on the right arrow at the bottom of the table.

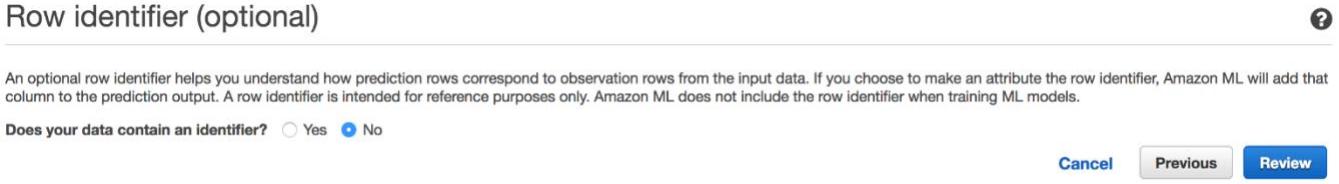


13. Make sure the schemas definition matches the following screenshot and note that **datetime** is set to **Text** as the data type. Click **Continue** move to the **Target** page**.**

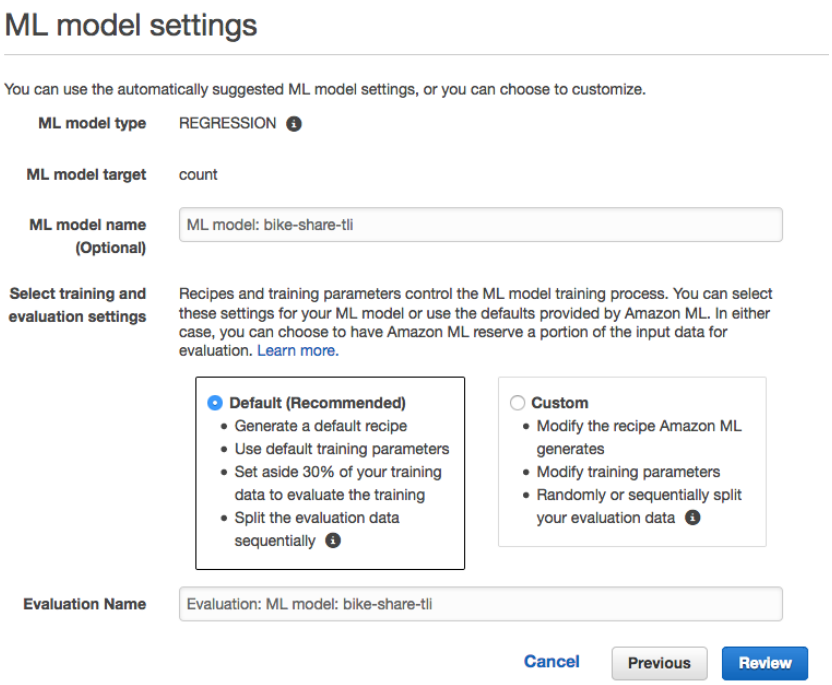


14. Select the **count** column as the target. This will forecast the bike rentals based on other variables such as weather, season, temperature and etc. Click **Continue** to move to **Row ID** page.

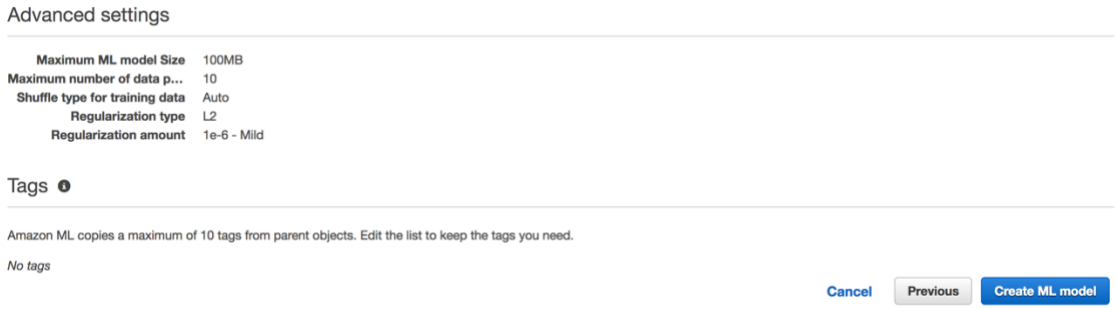
15. Select **No** in the "**Does your data contain an identifier?**" section. Click **Review** button to move the **Review** page.



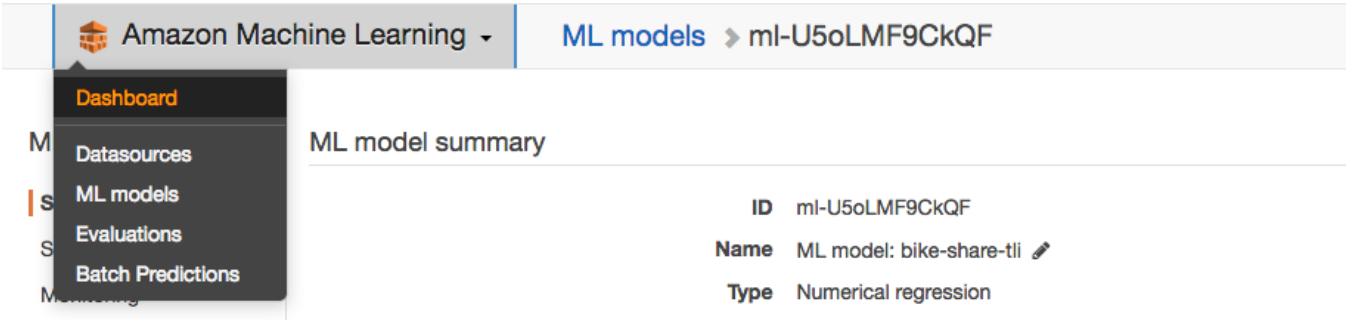
16. Click **Continue** to move to the **ML Model Settings** page. Leave everything as default on the page and click **Review**. Note the ML model type is **REGRESSION**, which is industry-standard learning algorithm known as Linear Regression.



17. Review the configuration parameters then scroll down and click the **Create ML Model** button.



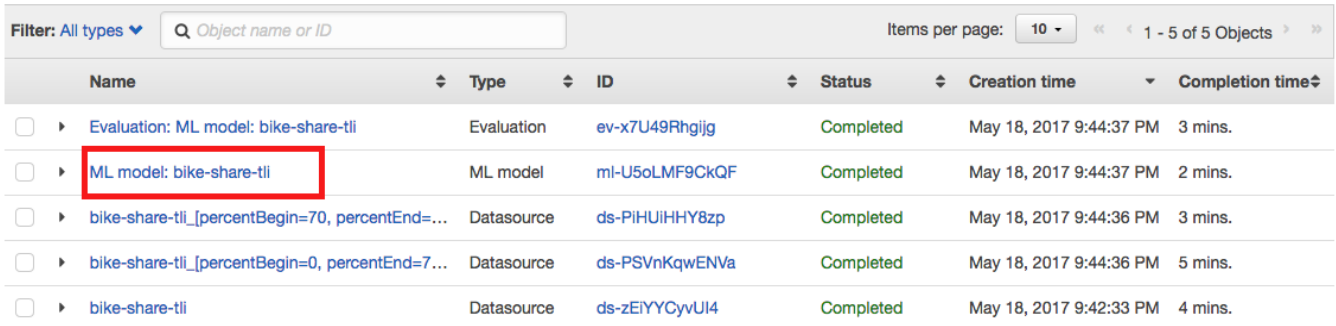
18. After a short while, the datasource is imported and the wizard will take you to the ML model summary page. Click on the **Amazon Machine Learning** drop down and select **Dashboard**.



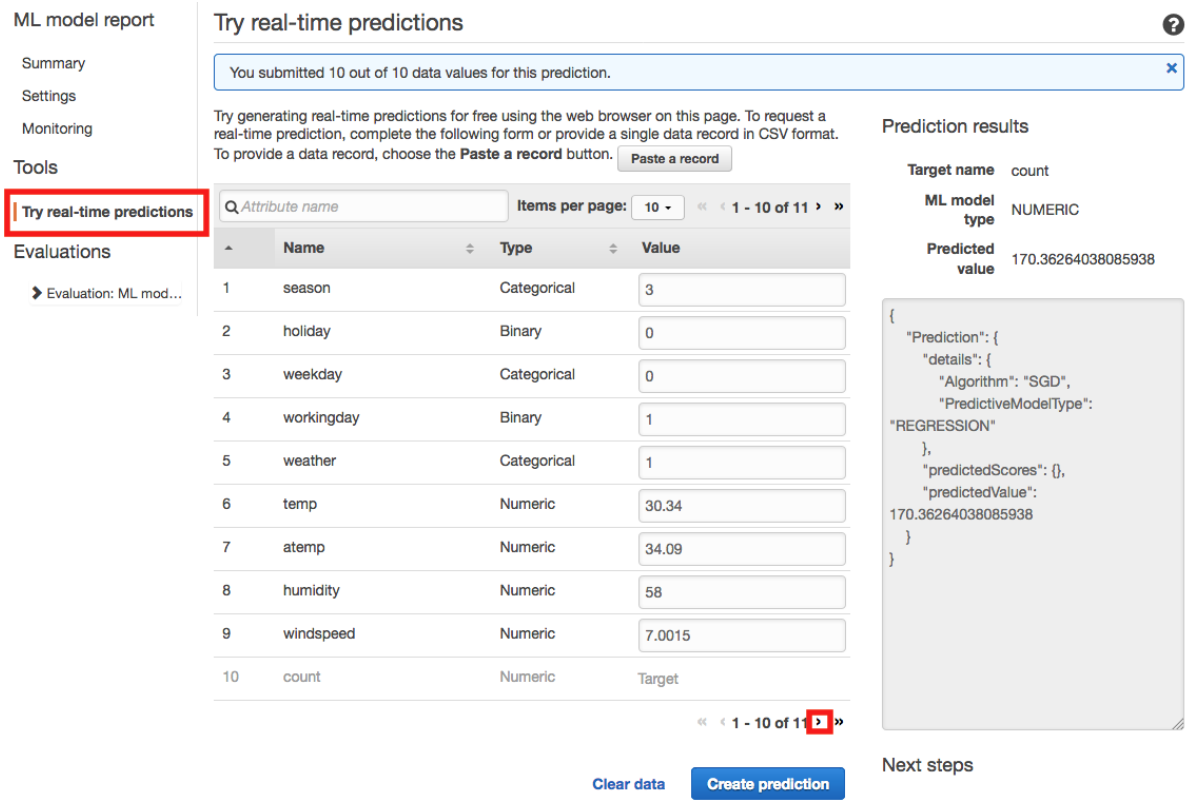
19. Please wait for the status changes from **Pending, In progress** to **Completed** on all items before continue to the next section. You can click on **Refresh** (on upper right hand corner of the table) to refresh the statuses. The process typically takes about 3 – 5 minutes.

Try Real-time Prediction

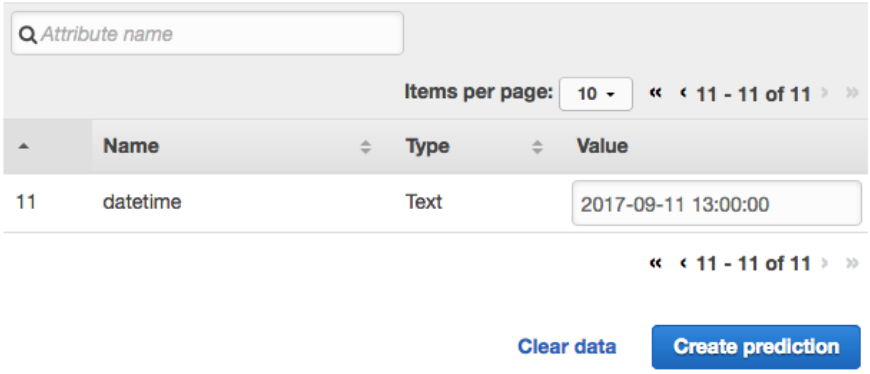
1. Now that the model is trained, we are ready to try some real-time predictions. Click on the machine learning model to go to the summary page.

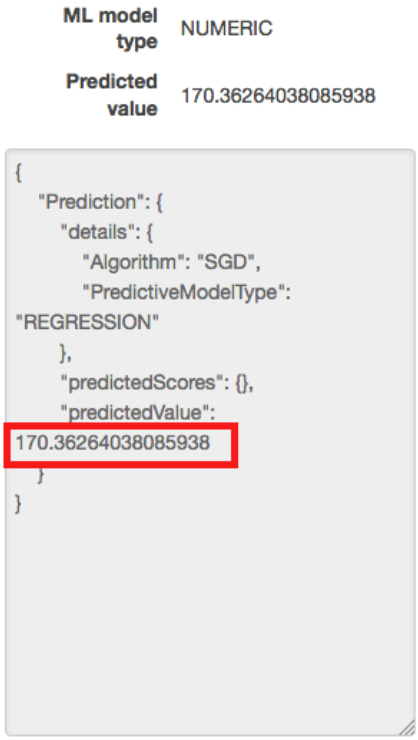


2. Click on **Try real-time predictions** on the left pane, enter the values shown below, then click the right arrow at the bottom right corner.



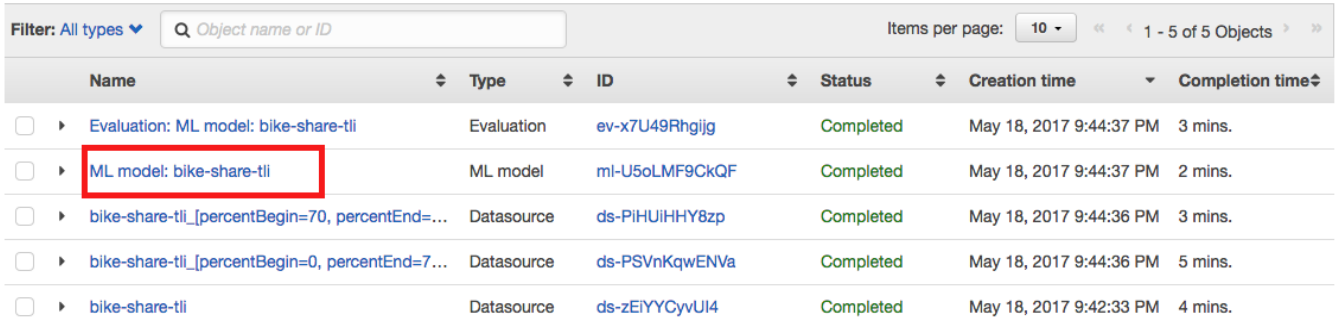
3. Enter a value in the datetime field and click **Create prediction**. Given the parameters entered, the predicted number of bikes will be rented is 170.



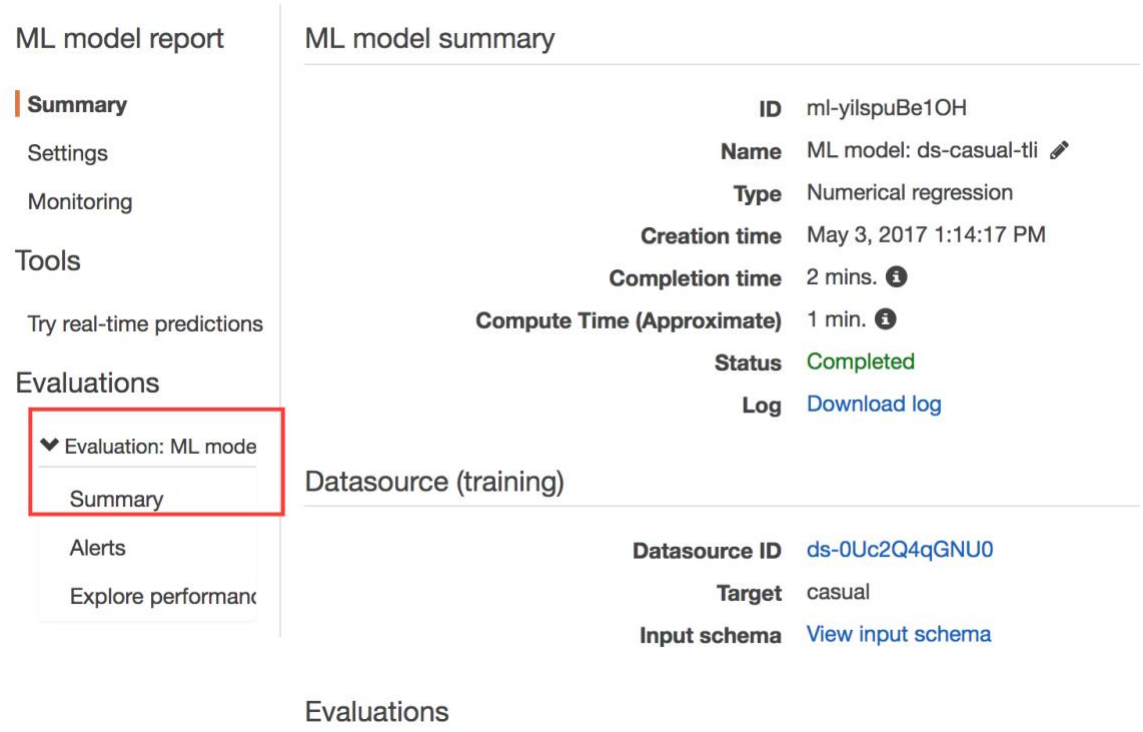


Evaluate the Model

1. Part of the machine learning process is to evaluate the generated model. The Amazon Machine Learning automatically evaluate its model's accuracy based on industry-stand 70/30 data split. Which means by default, it will split 70% of the dataset to train the model but leaves out the 30% of the data for validating the model. During the validation process, Amazon ML evaluates row by row with its prediction against the actual known value. This is a way to prevent the model from learning too much about the training data and not generalized enough to predict on data that has not seen. This phenomenon is called "over-fitting". There are many techniques such as regularization to prevent this from happening, Amazon Machine Learning offers regularization in advanced mode. However, this is beyond the scope of this tutorial.
2. To visualize the model performance, click on **Amazon Machine Learning** drop down and select **Dashboard**. Click on the machine learning model to go to the summary page.

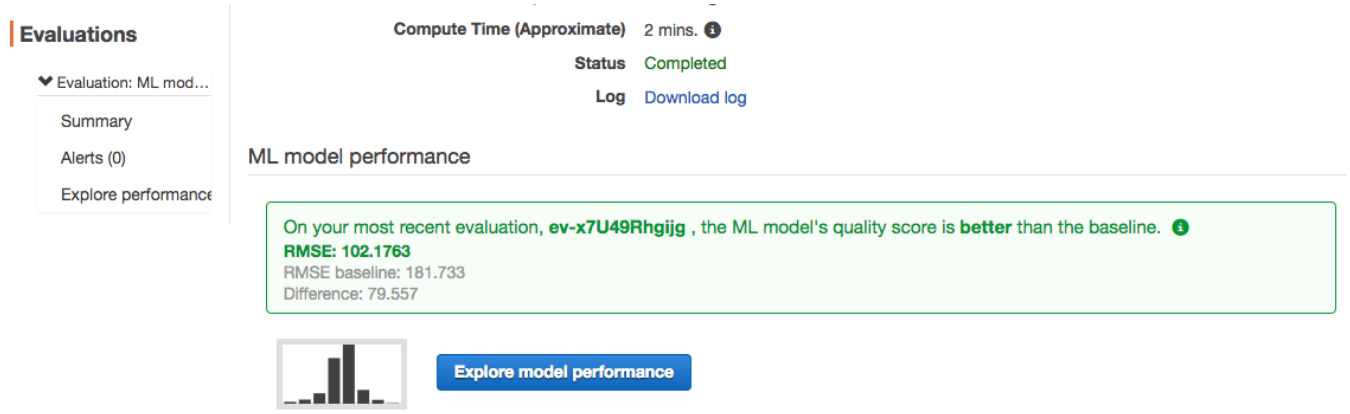


3. Under the **Evaluation** section, click on the down arrow to expand the options and click on **Summary**.

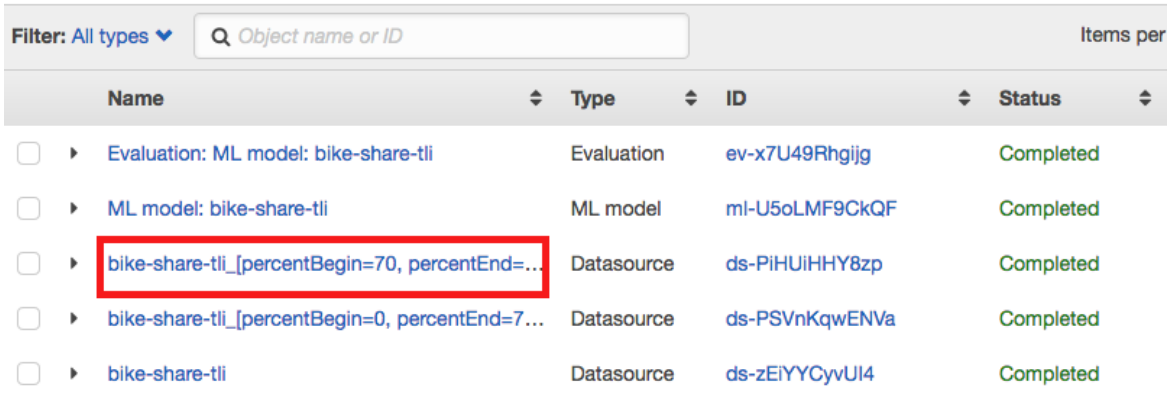


4. The accuracy of an regression model is evaluated by root-mean-square-error (**RMSE**). The lower the **RMSE** is, the better quality the model is. Amazon ML provides this as a baseline metric which takes the average of the target values. In the screen shot below, the model quality score (102) is performing better than the baseline (181) with a

difference of 80.

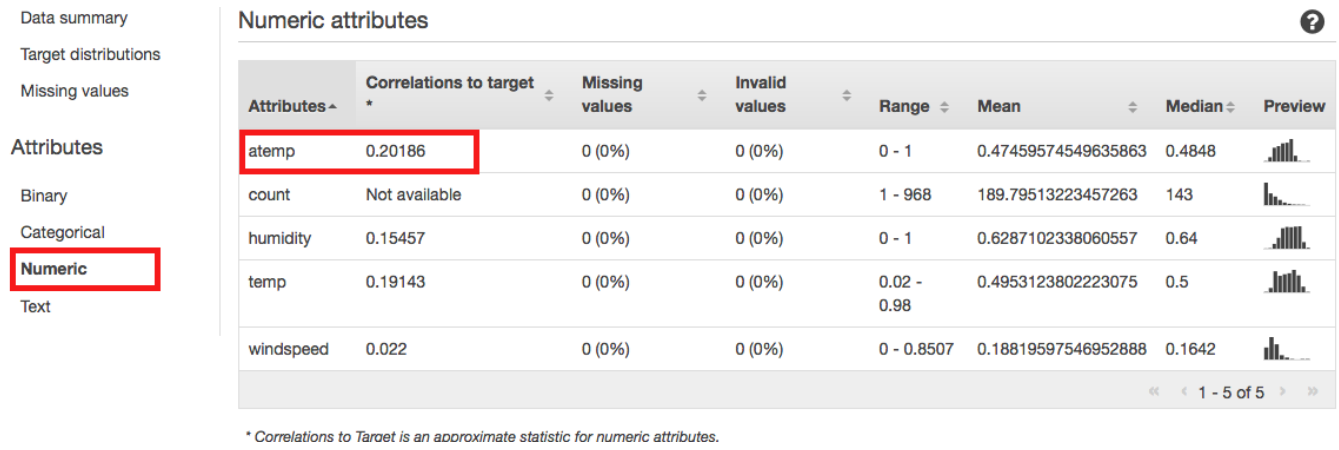


5. It is also possible to explore the datasource and evaluate each feature's (column) correlation to the target value (count). Features with higher value means they are more correlated to the target value, hence higher predictive power. This is very helpful when the datasource has a lot of features, you should select only the features with higher predictive power to reduce noise and improve the model's performance. This process is called **feature selection**.
6. To visualize the data correlations to target, click on **Amazon Machine Learning** drop down and select **Dashboard**. Click on one of the datasource to go to the Data insights page.



7. Under the **Attributes** section, click the **Numeric** attribute. In the screenshot below, we can see the atemp (feels like temperature) feature has a relatively higher correlation to the target. This makes sense as people may not want to rent a bike when it is too cold

or hot.



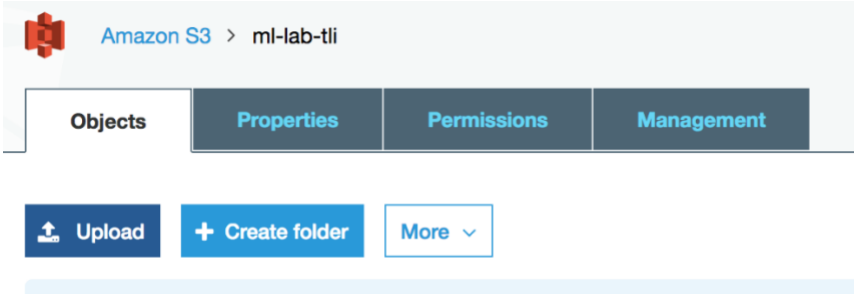
Improve Prediction Accuracy

When the model accuracy is not satisfactory, you can improve it through feature engineering and/or apply expert domain knowledge to the data such as feature selection. Dataset quality is an extremely important part of the model training process, remember the rule of “Gold-In-Gold-Out”. For example, recall that the **datetime** column is text. You could break it apart into separate year, month, date and time columns so that Amazon ML could use the new information to improve its accuracy. Recall that in the Amazon QuickSight charts, usage varies throughout the day. So, hour could be a key feature to improve the model’s accuracy. Another suggestion is to add actual text description to the weather feature instead of just a categorical number. Sometimes text can provide more meaning therefore a higher predictive power.

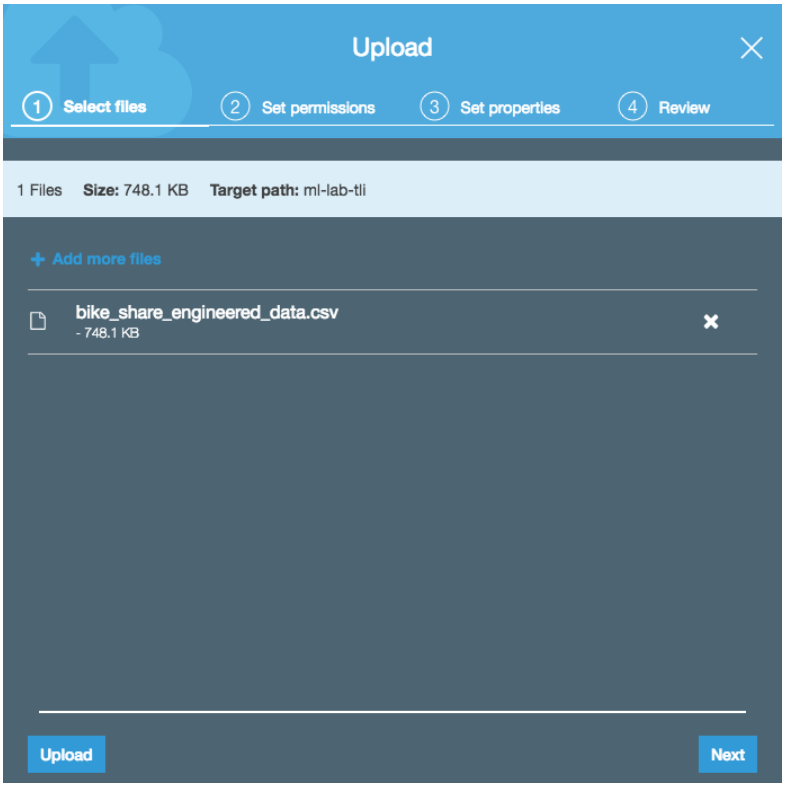
In this section, we will train a new model with a dataset that has datetime broken out to year, month, day, and hour. The step is fairly straight forward, you can use a simple Unix awk command or Excel to extract the data. For example, we used the following awk command to extract the date and time - `awk '{gsub("-", ",");gsub(" ", ",");gsub(":00:00", "");print }' bike_share_data.csv > bike_share_engineered_data.csv`

In the interest of time and keeping the focus on the Amazon Machine Learning service, we have broken out the datetime and made it available for download below.

1. Download the engineered biking sharing dataset from here.
2. Sign into the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3
3. In the upper-right corner of the AWS Management Console, confirm you are in the desired AWS region (e.g., Oregon).
4. Click the bucket name link with “ml-lab-**<your-initials>**”, then click **Upload.**

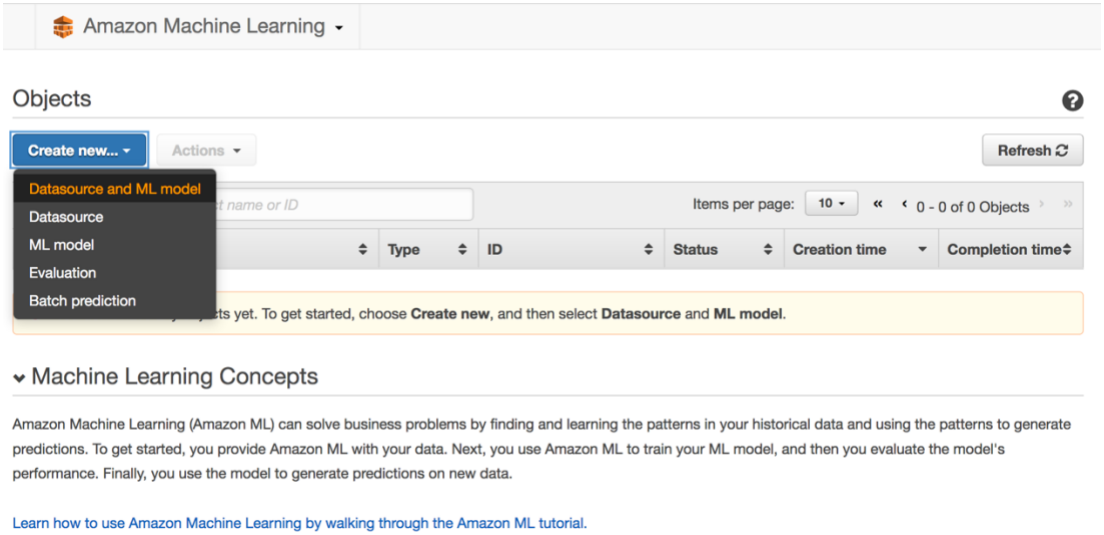


5. Click **Add Files**, find and select the **bike_share_engineered_data.csv** file and click **Upload**.

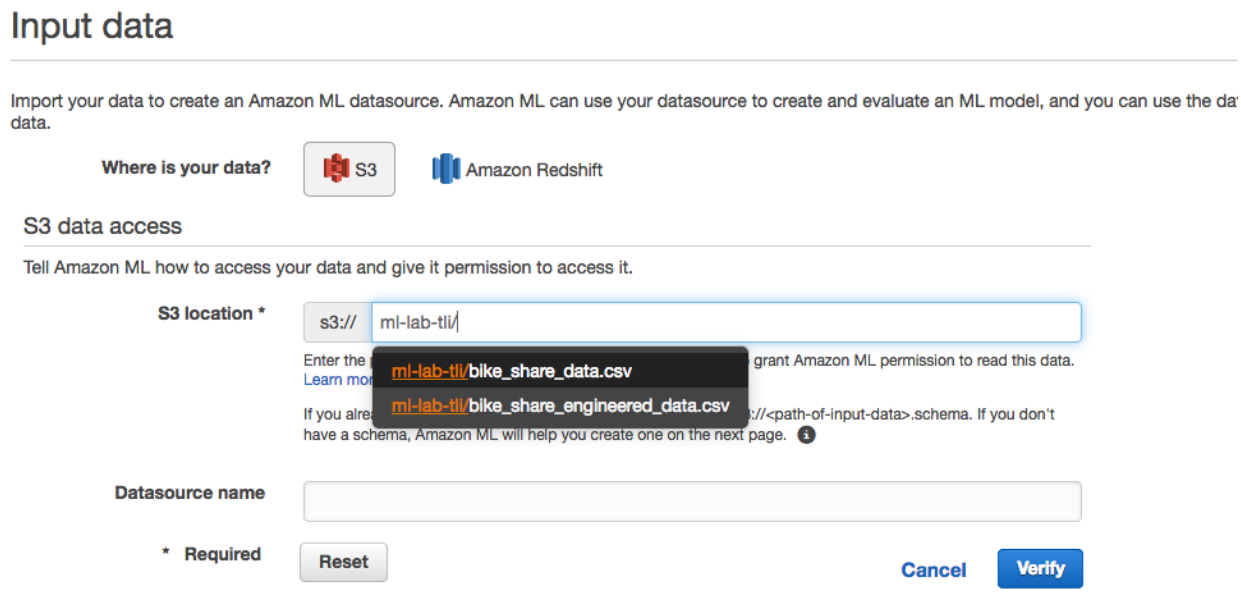


20. Open the Amazon Machine Learning console at https://console.aws.amazon.com/machinelearning/home

21. Click on **Create New…** button and select "**Datasource and ML Model**"

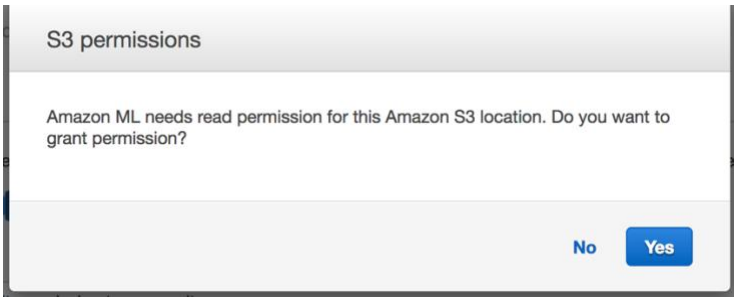


22. Under "**Where is you data?**" section, make sure the S3 icon is selected. In the S3 location text box, enter "ml-lab-<**your-initials>**" for the bucket name. Auto-complete should prompt a list of matches. Select the one that you created. Auto-complete will then prompt you with list of files in the bucket, select **bike_share_engineered_data.csv** from the list.



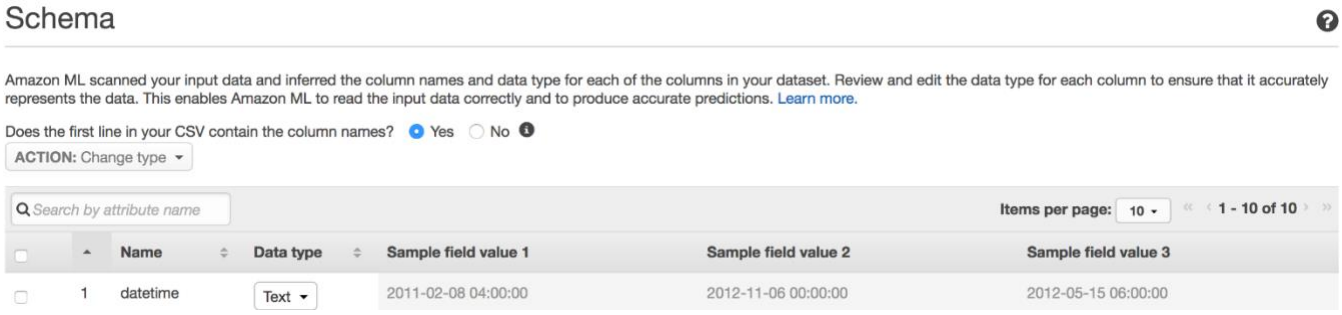
23. In the Datasource name text box, enter "bike-share-engineered-<**your-initials>**". Write down the Datasource name as you will need it later to create the model.

24. Click the **Verify** button. If prompted, give Amazon ML access to your S3 bucket by clicking **Yes**.

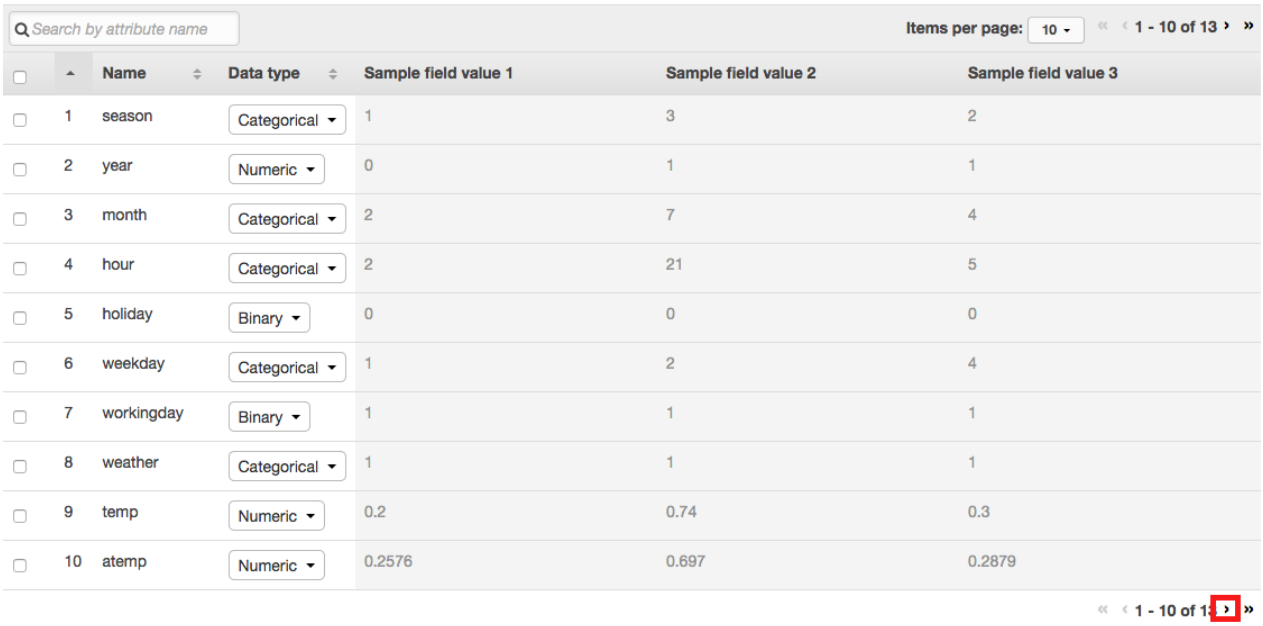


25. You should see a message indicating validation is successful. Click the **Continue** button to move to **Schema** page.

26. Select the **Yes** radio button in the "**Does the first line in your CSV contain the column names?**" section. The page will refresh and display column names.

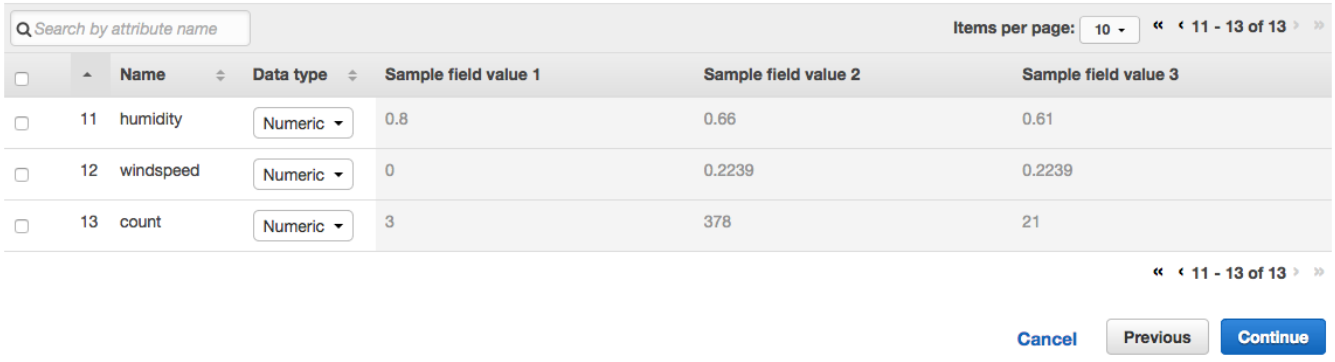


27. In the **season, weekday, weather, month,** and **hour** column, change **Data type** from **Numeric** to **Categorical.** In the **year** column, change **Binary** to **Numeric**. Make sure the schemas definition matches the following screenshot then click on the right arrow at the bottom right of the table.



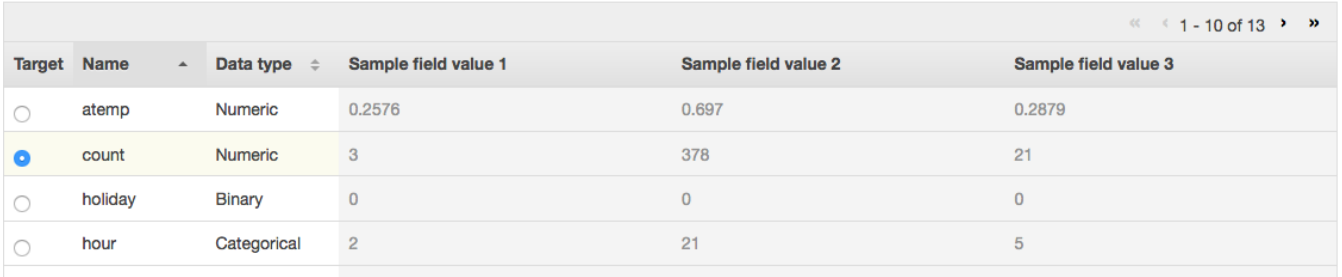
		Name	Data type	Sample field value 1	Sample field value 2	Sample field value 3
☐	1	season	Categorical	1	3	2
☐	2	year	Numeric	0	1	1
☐	3	month	Categorical	2	7	4
☐	4	hour	Categorical	2	21	5
☐	5	holiday	Binary	0	0	0
☐	6	weekday	Categorical	1	2	4
☐	7	workingday	Binary	1	1	1
☐	8	weather	Categorical	1	1	1
☐	9	temp	Numeric	0.2	0.74	0.3
☐	10	atemp	Numeric	0.2576	0.697	0.2879

28. Make sure the schemas definition matches the following screenshot and click **Continue** move to the **Target** page**.**



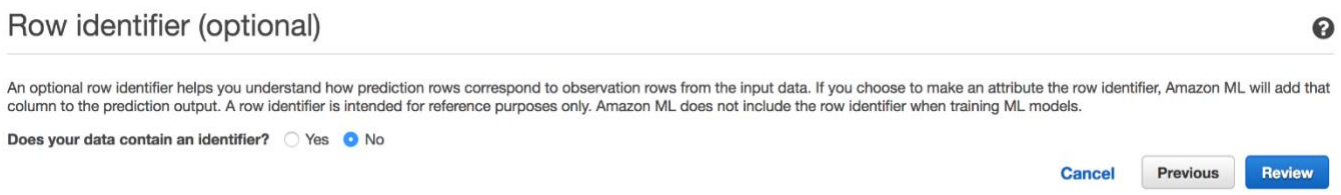
		Name	Data type	Sample field value 1	Sample field value 2	Sample field value 3
☐	11	humidity	Numeric	0.8	0.66	0.61
☐	12	windspeed	Numeric	0	0.2239	0.2239
☐	13	count	Numeric	3	378	21

29. Select the **count** column as the target. This will forecast the bike rentals based on other variables such as weather, season, temperature and etc. Click **Continue** to move to the **Row ID** page.

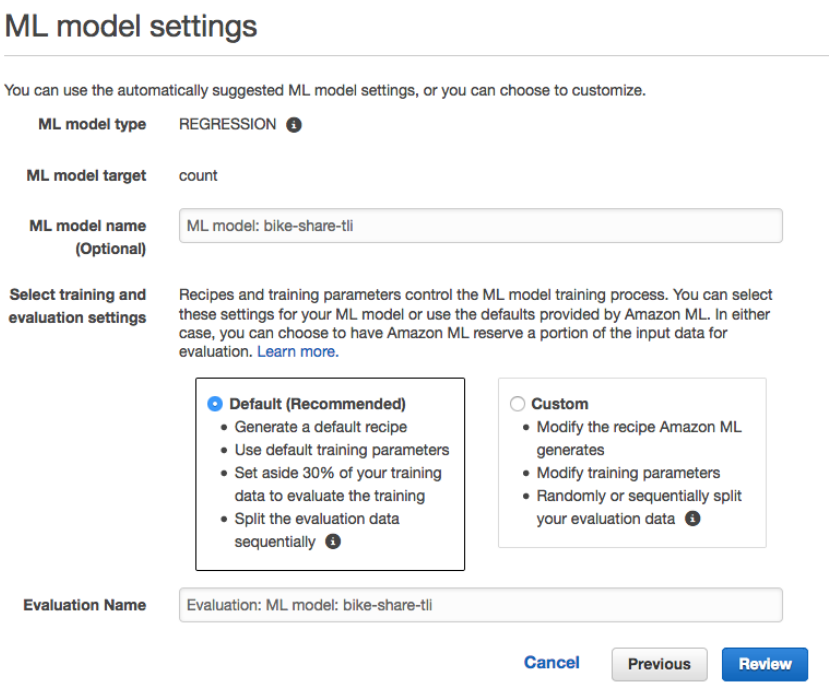


Target	Name	Data type	Sample field value 1	Sample field value 2	Sample field value 3
○	atemp	Numeric	0.2576	0.697	0.2879
◉	count	Numeric	3	378	21
○	holiday	Binary	0	0	0
○	hour	Categorical	2	21	5

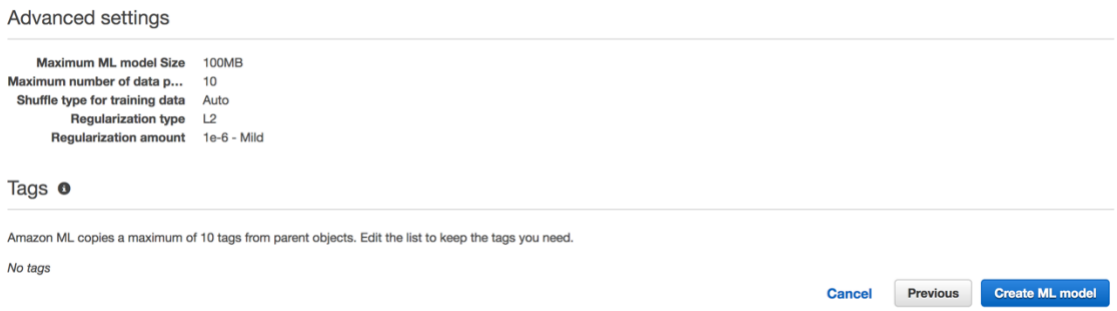
30. Select **No** in the "**Does your data contain an identifier?**" section. Click **Review** button to move the **Review** page.



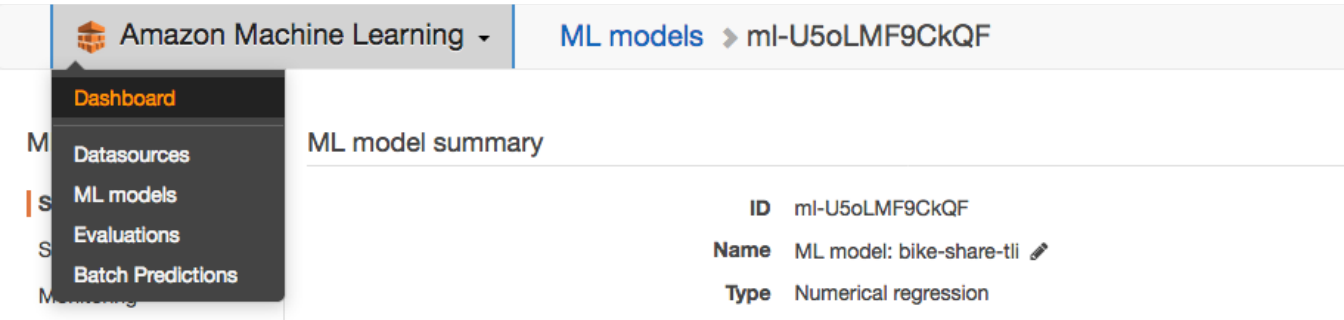
31. Click **Continue** to move to the **ML Model Settings** page. Leave everything as default on the page and click **Review**.



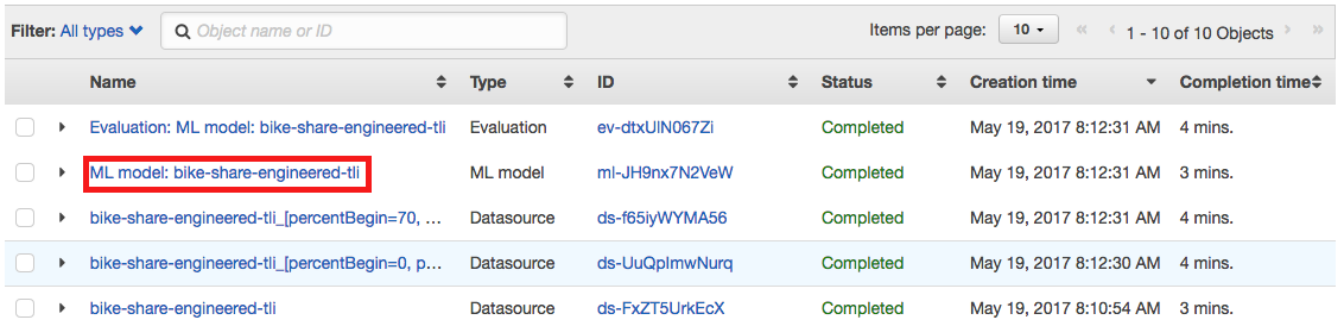
32. Review the configuration parameters then scroll down and click the **Create ML Model** button.



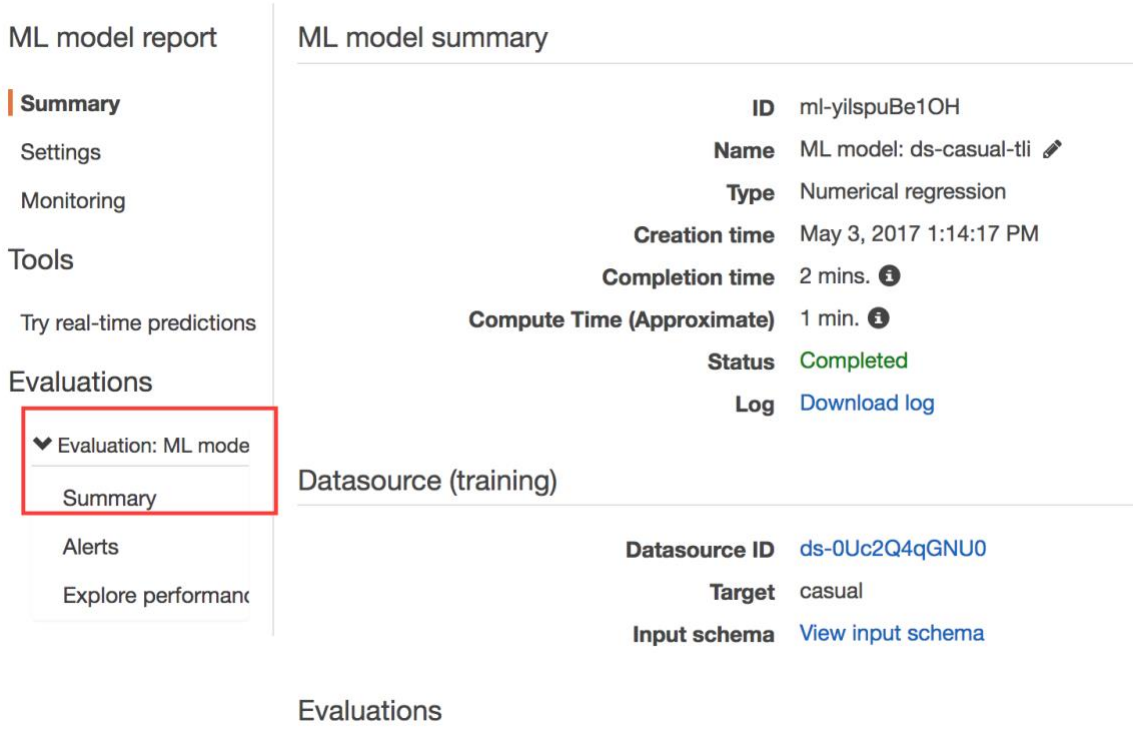
33. After a short while, the datasource is imported and the wizard will take you to the ML model summary page. Click on the **Amazon Machine Learning** drop down and select **Dashboard**.



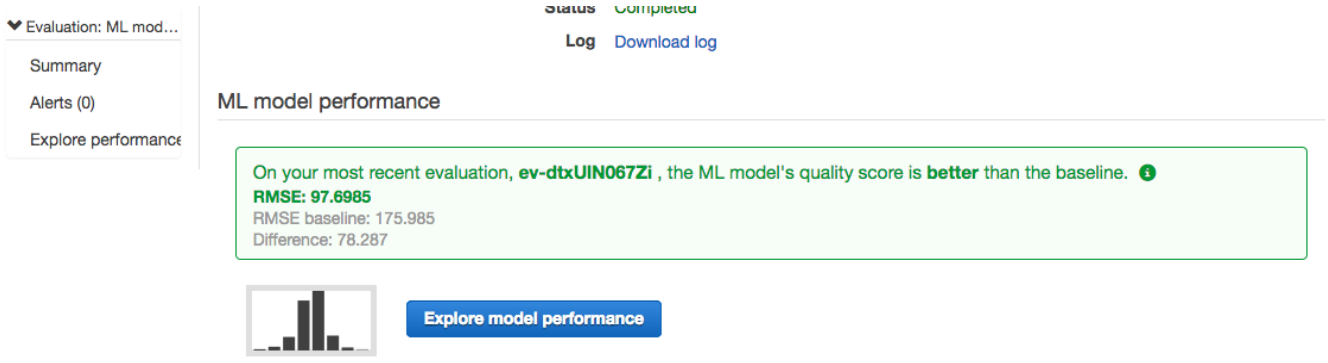
34. Please wait for the status changes from **Pending, In progress** to **Completed** on all items before continue to the next step. You can click on **Refresh** (on upper right hand corner of the table) to refresh the statuses. The process typically takes about 3 – 5 minutes.
35. Let's go ahead and visualize the model performance, click on **Amazon Machine Learning** drop down and select **Dashboard**. Click on the "ML model: bike-share-engineered-**<your-initials>**" model to go to the summary page.



36. Under the **Evaluation** section, click on the down arrow to expand the options and click on **Summary**.



37. In the screen shot below, the model quality score is 97 and baseline is 176 with difference of 78. Recall the first model with datetime not broken out had model quality score of 102 and baseline of 181 with difference of 80. The model with datatime broken out performed better overall than the model without. As demonstrated here, feature engineering is a critical component to improve predictive accuracy of a machine learning model.



Additional Available Models on Amazon Machine Learning Service

Recall the model created in this lab is called Regression, which is typically used to predict a numeric value such as demand for bike rental in this lab. There are two other models that Amazon ML supports, binary classification and multiclass classification. Binary classification is

an industry-standard learning algorithm known as Logistic Regression. This algorithm is great for predicting yes/no type of answers. For example, fraud detection and spam filtering. Multiclass classification is an industry-standard learning algorithm known as multinomial logistic regression. This algorithm is typically used for solving problems such as product categorization.

Clean Up

1. Delete the bucket with your initials in S3
2. Go to Amazon Machine Learning Dashboard and delete all resources with your initials

"Acknowledgement: Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science."

