



Amazon Machine Learning

An Introduction

Dan Moore

Amazon Machine Learning - An Introduction

Dan Moore

This book is for sale at <http://leanpub.com/amazonmachinelearning-anintroduction>

This version was published on 2017-07-29



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2017 Dan Moore

Contents

Introduction	1
What is Amazon Machine Learning	2
Key Elements	2
Interface	3
Regional Availability And Data Sources	3
Support	4
Types of Data Sources	5
S3	5
Redshift	7
RDS	7
External data	7
Conclusion	8

Introduction

Amazon Machine Learning, or AML, lets you leverage Amazon's existing infrastructure to perform supervised learning with no infrastructure management. Instead of worrying about servers and versions, you can focus on preparing and cleaning your data and creating multiple models based upon that data. The data can be drawn from RDS, Redshift or any source that can create CSV files in S3,

If you are just learning about AML, a wizard will help you through the steps of creating a machine learning model solving your binary classification, multiclass classification or regression problems using the stochastic gradient descent algorithm.

If you are an machine learning expert, you can also tweak the algorithms by setting various hyperparameters, including the learning rate and the number of passes. You can also control all of the model creation from the API, allowing you to script the entire process.

The model, when ready, is available for both batch and real time predictions. Batch predictions can process millions of records at a time, and give you back both a firm answer and a probability XXX. Real-time predictions typicall give responses within 100 milliseconds. And the [pricing for AML](#)¹ is simple to understand and you only pay for what you use.

With AML you can predict user behavior, pricing of houses, and answer other questions that can help your business or application.

¹<https://aws.amazon.com/machine-learning/pricing/>

What is Amazon Machine Learning

Amazon Machine Learning, or AML, provides you access to widely applicable machine learning algorithms without having to run any servers. This type of learning is useful for making predictions based on a set of data for which answers are known. AML supports supervised learning with the stochastic gradient descent algorithm. The end goal of AML is to create a model, which is what will allow you to make further predictions based on past data.

AML supports three different kinds of predictions. For binary outcomes, where observations lead to a yes/no result, AML supports binary classification. An example would be whether or not a prospect is likely to sign up for a new account, given their past interactions with your company. For multi valued results, where observations lead to one of N results, AML supports multi class classification. A good example of this would be which product to show a customer, given what they've looked at and bought in the past. And, for numeric values, AML supports regression. An example of that would be predicting house prices based on sales data and house attributes.

If you are not trying to use existing data and create predictions out of it using supervised learning, but are trying to instead recognize images or tease out patterns in text, you may want to consider [alternatives to AML](#).

Key Elements

While AML won't help you procure or prepare your data, AML provides three key elements to building a machine learning system.

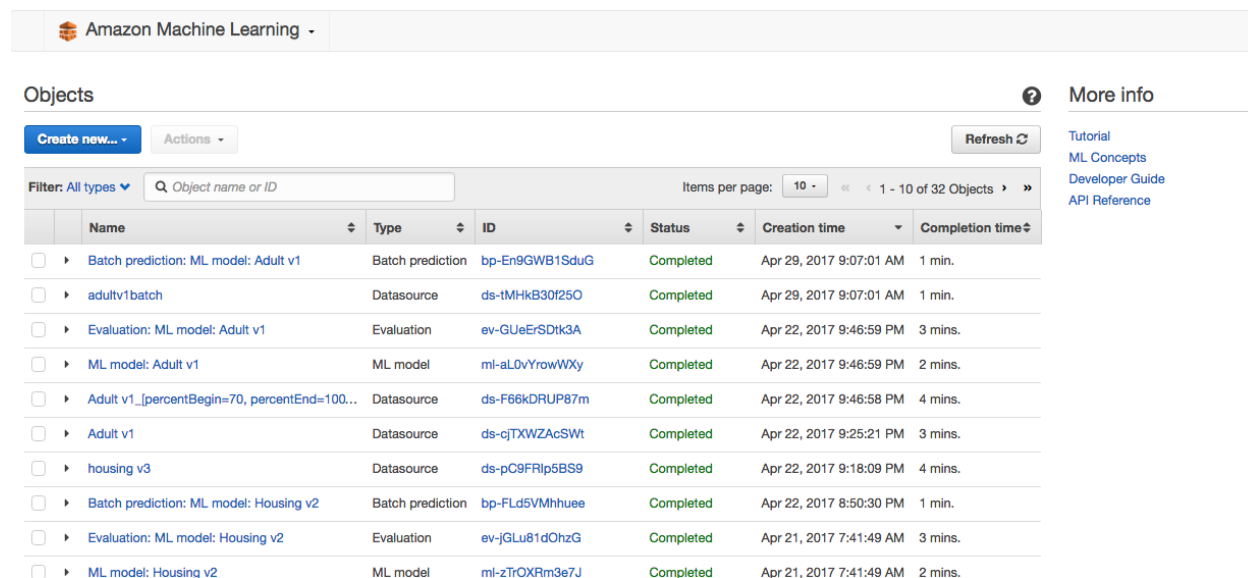
First, it reads data from and writes results to S3, an object store with unlimited capacity. S3 stores your data redundantly, inexpensively and securely. Having your data in S3 allows you to build multiple models off of it.

Second, AML builds a model for you. You have two options. You can build your model based on a default set of training parameters found to work across a large set of problem spaces. Or you can customize your model by tweaking the data using recipes and the training algorithm hyper parameters. However, if you are looking for full control over the model, AML might not be the right fit, as it doesn't provide that level of customization.

Finally, AML makes the model available for you to use with either batch or real-time predictions. Batch predictions read from data sources and write results to an arbitrary S3 locations, and real-time predictions are accessed via an API. The real-time prediction API endpoint is managed for you by AWS, and you needn't worry about updating or patching the underlying server infrastructure.

Interface

AML has two main interfaces. The first is the AWS console. The console requires manual interaction. You are creating objects using your mouse and keyboard. When you create a model with the console, the AML system helps you in many ways—setting up correct permissions and in some cases IAM roles, for example. You are also limited in some ways—your data definition file (called a schema) must be in a certain location.



The screenshot shows the Amazon Machine Learning console interface. At the top, there's a header with the Amazon Machine Learning logo and a 'More info' link. Below the header, there's a 'Objects' section with a 'Create new...' button and an 'Actions' dropdown. A search bar is present with the placeholder 'Object name or ID'. To the right of the search bar, it says 'Items per page: 10' and '1 - 10 of 32 Objects'. Below this is a table listing various objects. The table has columns for Name, Type, ID, Status, Creation time, and Completion time. The objects listed include batch predictions, data sources, evaluations, and ML models for both Adult and Housing datasets.

	Name	Type	ID	Status	Creation time	Completion time
<input type="checkbox"/>	Batch prediction: ML model: Adult v1	Batch prediction	bp-En9GWB1SduG	Completed	Apr 29, 2017 9:07:01 AM	1 min.
<input type="checkbox"/>	adultv1batch	Datasource	ds-tMHkB30f25O	Completed	Apr 29, 2017 9:07:01 AM	1 min.
<input type="checkbox"/>	Evaluation: ML model: Adult v1	Evaluation	ev-GUeErSDtk3A	Completed	Apr 22, 2017 9:46:59 PM	3 mins.
<input type="checkbox"/>	ML model: Adult v1	ML model	ml-aL0vYrowWxy	Completed	Apr 22, 2017 9:46:59 PM	2 mins.
<input type="checkbox"/>	Adult v1_percentBegin=70, percentEnd=100...	Datasource	ds-F66kDRUP87m	Completed	Apr 22, 2017 9:46:58 PM	4 mins.
<input type="checkbox"/>	Adult v1	Datasource	ds-cjTXWZAcSWt	Completed	Apr 22, 2017 9:25:21 PM	3 mins.
<input type="checkbox"/>	housing v3	Datasource	ds-pC9FRlp5BS9	Completed	Apr 22, 2017 9:18:09 PM	4 mins.
<input type="checkbox"/>	Batch prediction: ML model: Housing v2	Batch prediction	bp-FLd5VMhhuee	Completed	Apr 22, 2017 8:50:30 PM	1 min.
<input type="checkbox"/>	Evaluation: ML model: Housing v2	Evaluation	ev-jGLu81dOhzG	Completed	Apr 21, 2017 7:41:49 AM	3 mins.
<input type="checkbox"/>	ML model: Housing v2	ML model	ml-zTrOXRm3e7J	Completed	Apr 21, 2017 7:41:49 AM	2 mins.

The AWS console

The second way to interact is with the [machine learning API](#)², typically through an [SDK](#)³ or CLI. With this method of interacting, you have more control over the components of your system. But at the same time you are responsible for all settings and there are no helpful defaults. The API is also the only way a production system can interact with the real-time predictions system.

I recommend starting out using the console to gain familiarity with the AML system. Then, when you have a deeper understanding of the system and want to automate operations like the creation of batch predictions, the API is a better fit.

Regional Availability And Data Sources

At the current time, AML is available in two AWS regions—Northern Virginia and Ireland. For the most up to date availability, review [service availability by region](#)⁴. Your data can be stored anywhere, however cross region transfer costs will apply if you pull data from a different region.

²<http://docs.aws.amazon.com/machine-learning/latest/APIReference/Welcome.html>

³<https://aws.amazon.com/tools/#sdk>

⁴<https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/>

Support

The [AML forum](https://forums.aws.amazon.com/forum.jspa?forumID=194)⁵ is fairly active and monitored by AWS employees.

There is [sample code](https://github.com/aws-labs/machine-learning-samples)⁶ provided by Amazon for a variety of ML tasks.

There are also a couple of tags on AWS official blogs worth reading:

- [AML on the AI Blog](#)⁷
- [AML on the Big Data Blog](#)⁸

⁵<https://forums.aws.amazon.com/forum.jspa?forumID=194>

⁶<https://github.com/aws-labs/machine-learning-samples>

⁷<https://aws.amazon.com/blogs/ai/tag/amazon-machine-learning/>

⁸<https://aws.amazon.com/blogs/big-data/tag/amazon-machine-learning/>

Types of Data Sources

AML can work with four different kinds of data sources. They differ in how you deliver the data, and how easy it is to change the records. They all, however, create a csv file with a variable number of rows and a fixed number of columns.

You create data sources for training your model, for evaluating your model's efficacy, and for batch predictions. In all cases, the data sources need to have the same number of columns and data types within the columns (the same schema), except that any batch prediction data sources can omit the target attribute, the variable you are trying to predict.

Once you have created a data source, if it was for training you should not remove it. If you do delete the data source, you will be unable to try real time predictions in the browser, though you will still be able to use batch predictions and create a real time endpoint. The data source lives separately from the underlying data, but it's recommended that you don't delete the underlying data without removing the data source.

You aren't charged for data sources, just for the underlying storage costs (in the case of S3 data sources) or the costs of the database (RDS and Redshift). You can create multiple data sources pointing at the same data in S3, perhaps with different schemas.

S3

The first type of data source is an S3 location. The size of the data varies depending on whether this is a training datasource or a batch prediction input (training data is limited to 100GB, and batch prediction input is limited to 1TB). The data can be a single object in S3.

Single File With Header Rows

```
1 $ aws s3 ls aml-an-intro/s3-single-file/
2 2017-04-15 06:51:04          0
3 2017-04-15 06:51:47    4882918 banking.csv
```

If you provide a prefix for your data source location, then all the objects with that prefix are combined together. If multiple objects have that prefix, they must all have the same number of columns.

Multiple Files With Header Rows

```

1 $ aws s3 ls aml-an-intro/s3-multiple/
2 2017-04-15 06:52:03      0
3 2017-04-15 06:52:48    4882918 banking1.csv
4 2017-04-15 06:53:40    4882918 banking2.csv

```

CSV files in S3 must be in a plain text character set, in CSV. If an attribute of the record contains a comma, you must enclose that in double quotes. No newlines may be included in the attributes, and every row must end with a newline. If you are using Excel on a Mac, there are some [subtleties of which to be aware](#)⁹.

The first row of an input file may contain the header information, with the names of the attributes. If multiple files are provided, they may all contain the header row XXX CONFIRM

If, on the other hand, you don't want to include header lines, you should provide a schema file. In the management console, the schema file should be located in the same bucket and prefix as the data object. If your data is in a single object, the schema object should have a prefix of the data file name, and the suffix of .schema

Single File With Schema

```

1 $ aws s3 ls aml-an-intro/s3-single-file/
2 2017-04-15 06:51:04      0
3 2017-04-15 06:51:47    4882918 banking.csv
4 2017-04-15 06:53:40     488    banking.csv.schema

```

If you are using the AWS console to create your datasource schema files need to be in the same location with a specific format. If your data is in a single object, the schema object should have a prefix of the prefix, and a key of .schema

Multiple Files With Header Rows

```

1 $ aws s3 ls aml-an-intro/s3-multiple-schema/
2 2017-04-15 06:52:03      0
3 2017-04-15 06:52:48    4882918 banking1.csv
4 2017-04-15 06:53:40    4882918 banking2.csv
5 2017-04-15 06:53:40     488    .schema

```

Schema files are key to your data being processed correctly and are discussed in more detail in [Data Nuts and Bolts](#).

⁹<http://docs.aws.amazon.com/machine-learning/latest/dg/understanding-the-data-format-for-amazon-ml.html>

Redshift

Redshift, which is a managed data warehousing solution in the cloud, can provide data to be used as a datasource. This can be done via the API or the console. You need to provide connection information, including username, password and the cluster identifier. You also need to provide the SQL query to run. Note that the Redshift cluster must be publicly accessible in order for AML to retrieve the data, and that the Redshift cluster, the datasource and the target S3 bucket [must all be in the same region](#)¹⁰.

AML then runs the SQL query, optionally generates a schema, and places the data into an S3 location. The query won't be re-run periodically, and the data on S3 is what AML uses. XXX Can you lock down access to Redshift afterwards, then?

RDS

You can also create a datasource that is based on a query from a MySQL RDS database in a VPC. This functionality is unfortunately only available from the API, meaning you need to write scripts in python or another language to have this happen.

The AML API leverages [AWS Data Pipeline](#)¹¹ and helps you set up the correct roles and permissions. You specify a query and an S3 location, and AML takes care of making sure the data is placed there.

This may be an option if you are OK writing code against the API, but I'm not sure it makes sense very often. If the database containing your data runs any database engine other than MySQL, such as PostgreSQL or Oracle, you'll have to roll your own solution. Leveraging the Data Pipeline service may make sense, especially if you are pulling the data over repeatedly (and making batch predictions regularly). The Data Pipeline service can also pull data from other AWS services including S3, SQL databases and DynamoDB. It can also perform transformations against that data.

External data

AML doesn't support creating a data source from any external URL or database. Instead, you must push the data up to S3 via an API. It's easy to use the [S3 API to automate uploads](#)¹², or you can use the [AWS console](#)¹³. Another alternative can be to use a service like Zapier or Segment to get data to S3. You may need to manipulate the data to transform it from a given format into CSV.

You can also leverage the Data Pipeline service, which can read from JDBC databases and many other data sources and output to S3. If the data source is outside of AWS, beware of bandwidth charges.

XXX consider writing thus <<[Convert JSON File to CSV](#)¹⁴

¹⁰<http://docs.aws.amazon.com/machine-learning/latest/dg/using-amazon-redshift-with-amazon-ml.html>

¹¹<https://aws.amazon.com/datapipeline/>

¹²<http://docs.aws.amazon.com/AmazonS3/latest/API/RESTObjectPUT.html>

¹³<http://docs.aws.amazon.com/AmazonS3/latest/user-guide/upload-objects.html>

¹⁴[code/convert-json-to-csv.py](#)

If you are uploading a large object to S3, you should consider using [multi part uploads¹⁵](#), which will allow you to upload parts in parallel to increase the amount of data that can be uploaded in a certain period and also restart failed portions.

Conclusion

It's important to note that new data being fed into a data source previously configured requires a new model to be built against that data. It's not automatic in any way, no matter if you are using RDS, Redshift or S3.

Large amounts of data are key to AML, and there are multiple ways to make such data available to the model building process. Which data source works for you depends on where your data is, but realize that in the end all data ends up in S3 as a CSV file, and the other data sources are just conveniences for pulling data from existing repositories.

¹⁵<http://docs.aws.amazon.com/AmazonS3/latest/dev/uploadobjusingmpu.html>