# INTRO TO AMAZON MACHINE LEARNING WITH LOGISTIC REGRESSION





we look at Amazon's Machine Learning cloud service. In this first article we will look at logistic regression. In future blog posts we will see what other algorithms it offers.

Remember that **logistic regression** is similar to **linear regression**. It looks at a series of **independent variables** and calculates one **dependant variable**. If the probability of that outcome is > 50%, the that is classified as a 1 (true). Otherwise it is false (0). (Amazon lets you change that threshold, which is a little strange, as 50% is the standard value used by statisticians. But you could fiddle around with that nevertheless, such as when, for example, 30% means **true** in your situation.)

Here is related reading if you are just getting started:

- <u>Using Logistic Regression, Scala, and Spark</u>
- Introduction to TensorFlow and Logistic Regression
- <u>SGD Linear Regression Example with Apache Spark</u>
- Machine Learning and AI Frameworks: What's the Difference and How to Choose?
- <u>Amazon Machine Learning and Analytics Tools</u>

### **Explanation of the Process**

The idea behind Amazon ML is that you can run predictive models with without any programming. That is true for logistic regression. But you still need to put your data into a .csv format. Then you upload it to Amazon S3, which is their file storage system.

Here we run logistic regression using the <u>sample banking.csv data set</u> provided by Amazon. The goal is to predict whether a customer is likely to buy the banking service given the attributes shown below:

```
{
```

}

```
"version" : "1.0",
"rowId" : null,
"rowWeight" : null,
"targetAttributeName" : "y",
"dataFormat" : "CSV",
"dataFileContainsHeader" : true,
"attributes" : ,
"excludedAttributeNames" :
```

When you load this data set into ML, Amazon walks you through each field. It looks at each and determines whether they are **numeric** (could be any number), **categorical** (a specific set of numbers or text values), or **binary** (y or n or 1 or 0). The binary answers the question of whether this customer has pushed the banking product. That is the value we want to predict.

To use this, you need to do is to put your data into a spreadsheet format, with the first row as column headers. Unlike writing code yourself, where you have to convert all values to number, the algorithm here lets you use text or numeric values. Amazon will then take a guess as to which is the dependent variable and ask you to confirm that.

Then Amazon does what any ML programmer would do. It splits the input data set into a **training** data set and a **test** data set. It uses a 70/30 split, meaning 70% for one data set and 30% for another. Then it **evaluates** the model, meaning shows how accurately the independent variables predict the

dependant ones.

It could be that there is not much relationship at all between these variables. That would mean your assumption that this data is correlated is wrong. Of course, Amazon picked this banking data because it is correlated.

Having done the model correlation and evaluation, you can now use the trained model to run a **prediction**. In other words you go get some new data and run your prediction on whether this batch of persons might buy your banking product. Here Amazon charges you. They charged me \$2.90 to do this.

# **Getting Started**

Now we show how to use the service.

First you sign into the service by clicking on the <u>Amazon AWS Console</u> and click on **Machine Learning** to add that service to your account. Note that this service is not free. So set up a billing alert so that you do not get changed more than you have budgeted for.



# **Building the Model**

Here are the steps to build and use the model. We do not go in any particular order. Do not worry as Amazon has wizards to guide you through the process.

You can see how accurate the model is by the AUC (area under the curve). Don't worry about the exact definition. Unless you are a mathematician or statistician you will not understand it. Just understand that it is the difference between the observed values and predicted values. If they value was 1 then he model is perfect. 0.936 is a very high level of correlation. Anything below 0.5 is deemed to indicate that the data is not sufficiently correlated. In other words, that would mean your assumption of whether a customer might buy this banking product has nothing to do with those input values.

On your mo	st recent evaluation, ev-o5yt6SYasGi, the ML model's quality score is considered extremely good for most machine learning applications.
0	AUC: 0.936     Baseline AUC: 0.500     Difference: 0.436
Next step: If y	ou want to use this ML model to generate predictions, explore trade-offs to optimize the performance of your ML model first.
	Score threshold: U.5 Explore performance

# The ML Dashboard

Below is my dashboard showing what I have run. It's all the same model, but each time I used different datasets. One is prediction and the others Amazon generated automatically when it did the training and evaluation steps.

Objects								e		
Creat	te n	1ew •	Actions -							Refresh 2
Filter: All types ♥ Q Object name or ID						Items per page: 10 • 《 〈 1 - 5 of 5 Objects 〉				
		Name	\$	Туре	\$	ID	\$	Status 🗘	Creation time •	Completion time\$
	•	Evaluation	: ML model: banking	Evaluation	n	ev-o5yt6SYasGi		Completed	Mar 5, 2018 2:34:07 PM	3 mins.
	•	ML model:	: banking	ML mode	I	ml-BwZ5toyY935		Completed	Mar 5, 2018 2:34:07 PM	3 mins.
	•	banking_[	percentBegin=70, percentEnd=100,	Datasour	ce	ds-TWFB7xlSvvQ		Completed	Mar 5, 2018 2:34:06 PM	4 mins.
	•	banking_[	percentBegin=0, percentEnd=70, st	Datasour	ce	ds-uvkhhOiWDOY		Completed	Mar 5, 2018 2:34:06 PM	4 mins.
	•	banking		Datasour	ce	ds-b9GgsVz4Rt9		Completed	Mar 5, 2018 2:33:43 PM	4 mins.

Here is the screen to kick off the prediction step. Most people would do **Generate Batch Predictions.** That runs the model against data you have loaded into S3. **Real-Time Predictions** lets you type one record into a screen and it will run a prediction against that.

CloudWatch metrics	C View in CloudWatch					
Score threshold	0.5					
	A single dataset Generate one-time predictions for a single dataset. Generate batch predictions					
	Try real-time predictions Generate real-time predictions in your browser. Try real-time predictions					
	Enable real-time predictions To enable real-time predictions now, create a real-time prediction endpoint. Create endpoint					
ere are the prediction results.	As you can see it charged me \$2.90, which is \$0.10 per 1,000	Н				

The Amazon ML fee for batch predictions is \$0.10 per 1,000 predictions, rounded up to the next 1,000. Learn more.

Type the path to the S3 location in which the prediction results will be saved.

S3 destination

Batch prediction name (Optional)

Batch prediction: ML model: banking

s3:// aml-sample-data/predictions.csv

Cancel Previous

T. INCVICT

Review

ID	ds-uvkhhOiWDOY							
Name	banking_[percentBegin=0, percentEnd=70, strategy=sequential] 🖋							
Creation time	Mar 5, 2018 2:34:06 PM							
Completion time	4 mins. 3							
npute Time (Approximate)	15 mins. 🕄							
Status	Completed							
Message	Not available							
Input schema	View input schema							
Log	Download log							
	Use this datasource to 👻							
	Convisettings to create a new datasource							
	Create (train) an MI model							
	Evaluate an ML model							
Target name								
Target type	Generale batch predictions							
Target visualization								
S3 location	s3://aml-sample-data/banking.csv							
Number of files	1							
Data format	CSV							
Total size	3.3 MB							
Data rearrangement	{     "splitting": {         "percentBegin": 0,         "percentEnd": 70     } }							

#### Load Data in S3

0

Amazon's banking data is already at a URL where you can use it. In other to use Amazon's data to run a prediction against it, which in real life you would do by gathering more data about your customers, you need to create a bucket in S3. That is like a folder. Below I create the bucket **walkerbank**.

Create bucket					
1 Name and region	2 Set properties	3 Set permissions	(4) Review		
Name and region					
Bucket name 🕚					
walkerbank					
Region					
EU (Ireland)			~		
Copy settings fror	n an existing bucket				
Select bucket (opti	ional)		2 Buckets 🗸		

# Wait and Wait some More

It will take some time for your model to run as it gets in a queue behind other customers. Below you can see that this one is in a **pending** state.

aws Services ~	Resource Groups	~ <b>*</b>		🗘 🛛 Walker Rowe 🕶	Ireland 👻 S	Support 👻
🌲 Amazon Machine	e Learning 🗸	Batch Predictions > bp-z56	хРVjwpTi			
Batch prediction	Batch prediction s	summary			Delete this	s Batch prediction
Summary		ID	bp-z56xPVjwpTi			
		Name	Batch prediction: ML model: banking 🖋			
		Creation time	Mar 5, 2018 3:25:14 PM			
		Completion time	Not available 🚯			
		Compute Time (Approximate)	Not available 🕄			
		Status	In progress			
		Datasource ID	ds-uvkhhOiWDOY			
		ML model ID	ml-BwZ5toyY935			
		Input S3 URL	s3://aml-sample-data/banking.csv			
		Output S3 URL	s3://walkerbank/predictions.csv/			
		Log	Not available			
Processing information						
		Number of records seen	Not available			
		Records that failed to process	Not available			
	Tags Add or edit tag	JS				
	No tags					
🗨 Feedback 🔇 English (US	5)		© 2008 - 2018, Amazon Web Services, Inc. or it	s affiliates. All rights reserved.	Privacy Policy	Terms of Use

## **Get the Results**

Amazon saves the results in S3. You cannot really browse the results online. Instead you can download the file, unzip it, and then look at it. That is what I have done here.

Here is what Amazon has calculated. Too bad it put the results in a new file instead of appending the prediction as a new column in the input file. Below what we see is the actual value (**trueLabel**) from the input data and the predicted value (**bestAnswer**) based upon the model that Amazon built.

#### trueLabel,bestAnswer,score

```
0,0,1.437033E-2
0,0,1.139906E-2
1,1,8.305257E-1
0,0,8.966137E-2
1,0,4.096018E-1
0,0,3.634616E-3
0,0,2.641097E-2
0,0,3.487612E-2
1,1,5.777377E-1
0,0,4.469287E-2
0,0,2.456573E-3
0,0,4.300581E-1
1,0,8.399929E-2
0,0,1.024602E-2
```

### **Next Steps**

In the next blog post we will see whether Amazon can do k-mean clustering, linear regression, or other types of analysis.