

# How to use Intel® DAAL Logistic Regression via SageMaker web interface

## Description of algorithm:

Logistic regression is a method for modeling the relationships between one or more explanatory variables and a categorical variable by expressing the posterior statistical distribution of the categorical variable via linear functions on observed data. If the categorical variable is binary, that is it takes only two values, "0" and "1", the logistic regression is simple, otherwise, it is multinomial.

[Intel® DAAL developer guide](#)

[Intel® DAAL documentation for Logistic Regression](#)

## Instruction:

1. Visit page on SageMaker Marketplace and click "Continue to Subscribe"

The screenshot shows the AWS Marketplace page for Intel® DAAL Logistic Regression. The page header includes the AWS Marketplace logo, a search bar, and navigation links. The product title is 'Intel® DAAL Logistic Regression.' with a 'Continue to Subscribe' button highlighted by a red arrow. Below the title is a 'Product Overview' section with a description of logistic regression, a 'Key Data' table, and a 'Highlights' box.

**Key Data**

Version	0.1 <a href="#">Usage Instructions</a> <a href="#">Release Notes</a>
By	Intel® AI
Categories	<a href="#">Regression</a> <a href="#">ML Solutions</a> <a href="#">High Performance Computing</a>
Type	Algorithm
Fulfillment Methods	Amazon SageMaker <a href="#">?</a>

**Highlights**

- Intel® DAAL Machine Learning algorithms are lightning fast and built for performance !
- Intel® DAAL uses optimized algorithms from the Intel® Math Kernel Library and Intel® Integrated Performance Primitives.

2. Click "Accept Offer" if you agree with EULA at end of page. If you already subscribed on algorithm on Marketplace this step will be skipped.

The screenshot shows the AWS Marketplace interface for the Intel DAAL Logistic Regression software. The top navigation bar includes the AWS Marketplace logo, a search bar, and links for Categories, Delivery Methods, Solutions, Migration Mapping Assistant, Your Saved List, Partners, Sell in AWS Marketplace, Amazon Web Services Home, and Help. The Intel logo is prominently displayed on the left, with the product name 'Intel® DAAL Logistic Regression.' and a 'Continue to configuration' button on the right.

Below the navigation bar, the page title is 'Subscribe to this software'. A paragraph of text explains the subscription process: 'Review the pricing terms and End User License Agreement (EULA) below. By clicking Accept Offer, you will be subscribed to this software and agree that you use of this software is subject to the pricing terms and the seller's End User License Agreement (EULA) and your use of AWS services is subject to the AWS Customer Agreement'. A red arrow points to the 'Accept Offer' button, which is located in the top right corner of the main content area.

The 'Accept Offer' button is orange and contains the text 'Accept Offer'. Below it, the 'Pricing Terms' section is visible, featuring a table titled 'Algorithm usage' with columns for 'Instance type' and 'Algorithm/hr'. The table lists various instance types and their corresponding pricing, all of which are \$0.00.

Instance type	Algorithm/hr
mL.m4.xlarge	\$0.00
mL.m4.2xlarge	\$0.00
mL.m4.4xlarge	\$0.00
mL.m4.10xlarge	\$0.00
mL.m4.16xlarge	\$0.00
mL.m5.large	\$0.00
mL.m5.xlarge	\$0.00

3. Click "Continue to configuration"

The screenshot shows the AWS Marketplace interface for the Intel DAAL Logistic Regression software. At the top, the AWS Marketplace header includes navigation links like Categories, Delivery Methods, Solutions, Migration Mapping Assistant, Your Saved List, Partners, Sell in AWS Marketplace, Amazon Web Services Home, and Help. The Intel logo and product name 'Intel® DAAL Logistic Regression.' are displayed. A red arrow points to the 'Continue to configuration' button. Below this, there's a 'Subscribe to this software' section with a brief review of pricing terms and EULA. A green notification bar states 'Thank you for subscribing. You can now view your product.' Below this, the 'Intel® AI Offer' section shows a table of pricing terms for algorithm usage across various instance types, all priced at \$0.00.

Product Detail > Subscribe

## Subscribe to this software

Review the pricing terms and End User License Agreement (EULA) below. By clicking Accept Offer, you will be subscribed to this software and agree that you use of this software is subject to the pring terms and the seller's End User License Agreement (EULA) and your use of AWS services is subject to the [AWS Customer Agreement](#)

✔ Thank you for subscribing. You can now view your product. ✕

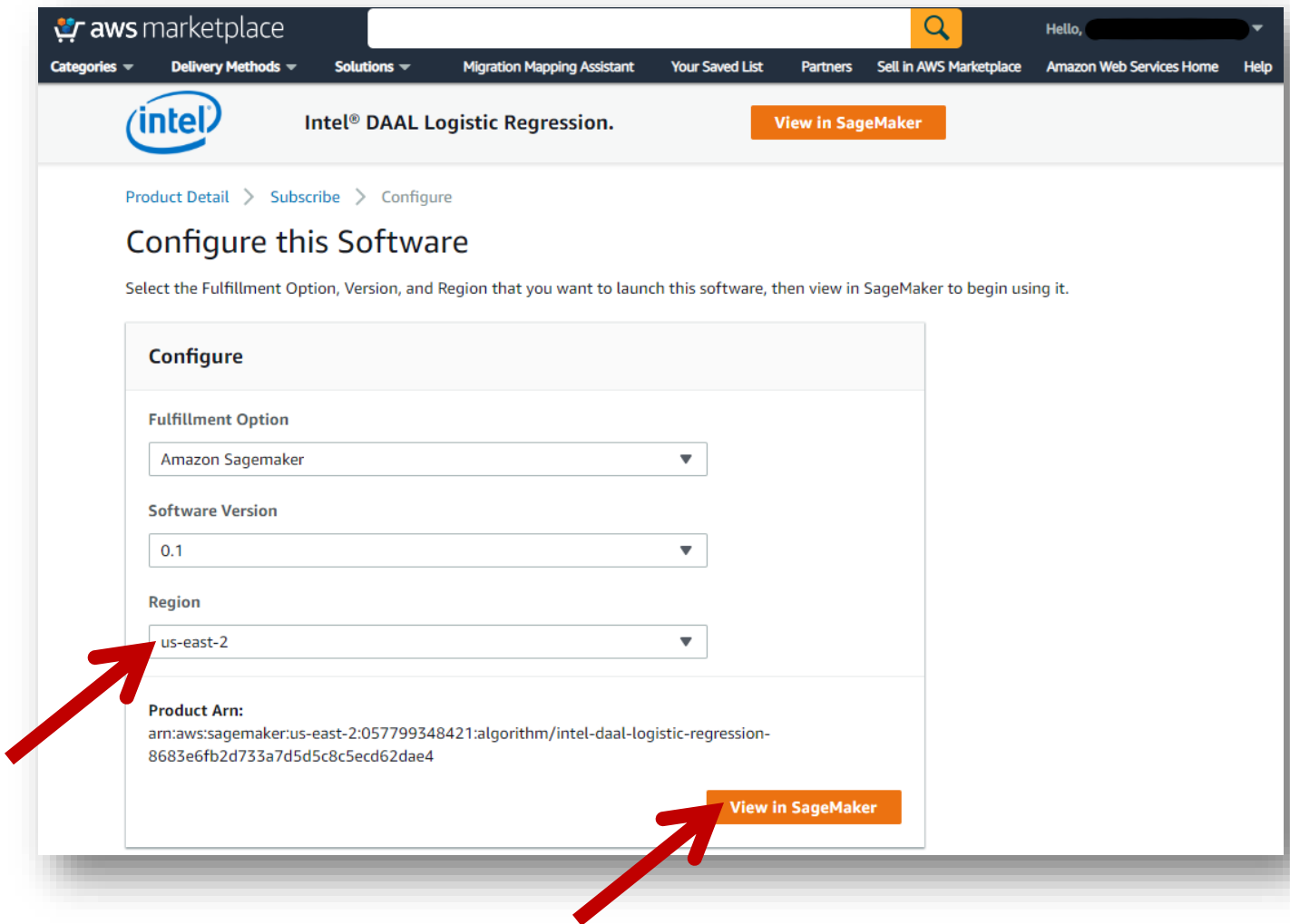
**Intel® AI Offer** Already Subscribed

By: Intel® AI

**Pricing Terms**

Algorithm usage	
Instance type	Algorithm/hr
mL.m4.xlarge	\$0.00
mL.m4.2xlarge	\$0.00
mL.m4.4xlarge	\$0.00
mL.m4.10xlarge	\$0.00
mL.m4.16xlarge	\$0.00
mL.m5.large	\$0.00
mL.m5.xlarge	\$0.00

4. Choose the Region and click “View in SageMaker”



Logistic regression algorithm on AWS SageMaker is divided into two stages: training job and getting inference from endpoint.

Training job is computing model values from provided training data.

After that, you should create model with computed values and endpoint based on it.

Sending data to endpoint gives you predictions in response.

5. Select needed algorithm version and click “Create training job”

The screenshot shows the Amazon SageMaker console interface. On the left is a navigation sidebar with options like Dashboard, Search, Ground Truth, Notebook, Training, and Inference. The main content area is titled 'Intel® DAAL Logistic Regression.' and includes two buttons: 'Create hyperparameter tuning job' and 'Create training job'. A red arrow points to the 'Create training job' button. Below the buttons is a table titled 'Algorithm versions' with columns for Title, Version, and Algorithm ARN. The table contains one entry: 'Intel® DAAL Logistic Regression.' with version '0.1' and a long ARN.

Title	Version	Algorithm ARN
Intel® DAAL Logistic Regression.	0.1	arn:aws:sagemaker:us-east-2:057799348421:algorithm/intel-daal-logistic-regression-8683e6fb2d733a7d5d5c8c5ecd62dae4

6. Type job name, select IAM role and instance type

The screenshot shows the 'Job settings' page in the Amazon SageMaker console. It contains several sections: 'Job name' with a text input field containing 'daal-log-reg-test'; 'IAM role' with a dropdown menu showing 'AmazonSageMaker-ExecutionRole-'; 'Algorithm options' with a section for 'Algorithm source' and 'Choose an algorithm subscription'; and 'Resource configuration' with fields for 'Instance type' (set to 'ml.m4.xlarge'), 'Instance count' (set to '1'), and 'Additional storage volume per instance (GB)' (set to '1'). Three red arrows point to the 'Job name' field, the 'IAM role' dropdown, and the 'Instance type' dropdown respectively.

## 7. Choose hyperparameters

### Hyperparameters

You can use hyperparameters to finely control training. We've set default hyperparameters for the algorithm you've chosen.

Key	Value
nClasses	100
penaltyL1	0.1
penaltyL2	0.1
interceptFlag	False
solverName	sgd
solverMethod	momentum
solverMaxIterations	1000
solverLearningRate	0.001
solverAccuracyThreshold	0.0001
solverBatchSize	150

Parameter	Type	Default value	Description
nClasses	int	None (required)	Number of classes in training dataset
penaltyL1	float	0	Penalty coefficient for L1 regularization
penaltyL2	float	0	Penalty coefficient for L2 regularization
solverName	str	'sgd'	Name of solver that will be used for training stage available values: 'lbfgs', 'adagrad', 'sgd'
solverMethod	str	'defaultDense'	Method of the solver. Available values for 'sgd': 'momentum', 'minibatch', 'defaultDense' available values for others solver: 'defaultDense'
solverMaxIterations	int	100	Max number of iterations for training stage
solverAccuracyThreshold	float	1.0-e4	Accuracy of the algorithm. The algorithm terminates when this accuracy is achieved.
solverBatchSize	int	number of rows in training dataset	Number of batch indices to compute the stochastic gradient.
solverLearningRate	float	1.0-e3	learning rate for optimization problem applicable for 'sgd','adagrad', 'saga' only
solverStepLength	float	1.0-e3	step size for optimization problem applicable for 'lbfgs' only
solverCorrectionPairBatchSize	int	number of rows in training dataset	Number of batch indices to compute Hessian approximation. applicable for 'lbfgs' only
solverL	int	1	The number of iterations between calculations of the curvature estimates applicable for 'lbfgs' only

## 8. Specify S3 location of input data for training

### Input data configuration

Create up to 8 channels of input sources. If the algorithm you chose supports multiple input channels, you can specify those here. See [Algorithms Provided by Amazon SageMaker: Common Parameters](#)

#### Channels

##### ▼ training

Remove

Channel name

training

Input mode - *optional*

File

Content type - *optional*

text/csv

Choose one of the formats below

- text/csv

Compression type

None

Record wrapper

None

S3 data type

S3Prefix

S3 data distribution type

FullyReplicated

S3 location

s3://daal-log-reg-test/input/data/training/training\_data.csv

Add channel





9. Specify S3 output path (model will be stored here) and click “Create training job”

### Output data configuration

S3 output path

s3://bucket/path-to-your-data/

Encryption key - *optional*

If you want Amazon SageMaker to encrypt the output of your training job using your own AWS KMS encryption key instead of the default S3 service key, provide its ID or ARN.

### ▼ Tags - *optional*

Key

Value

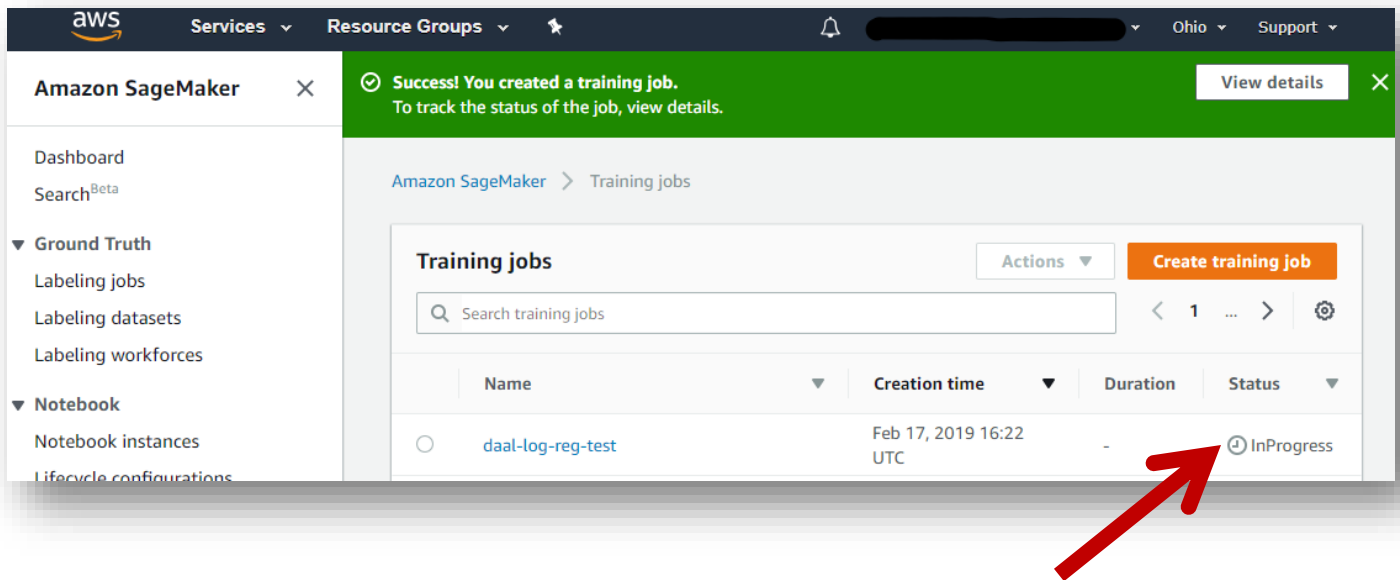
Remove

[Add tag](#)

Cancel

Create training job

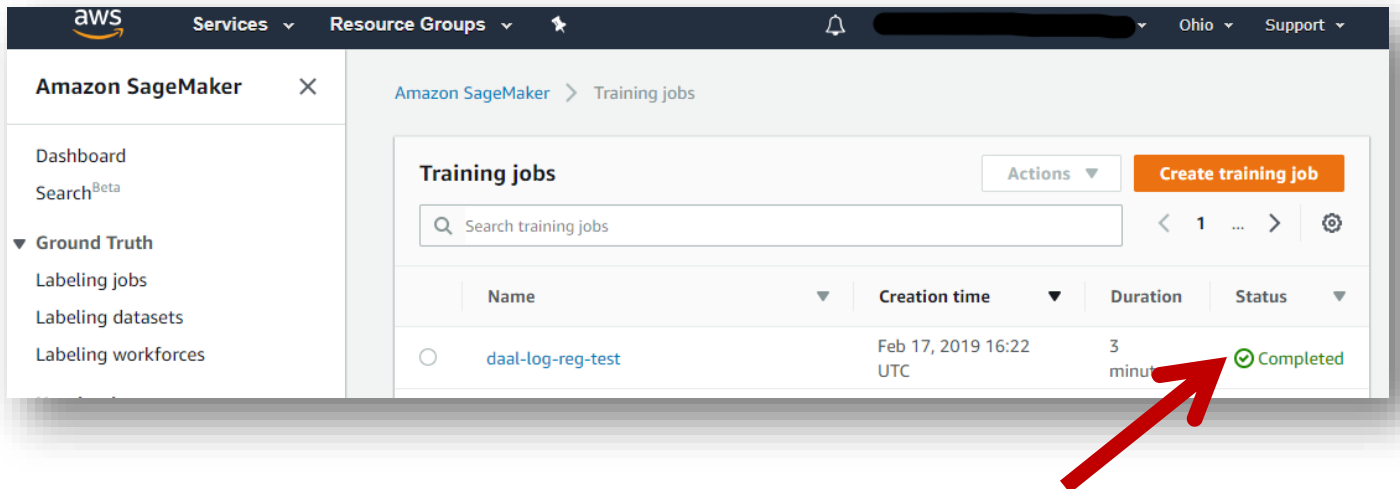
## 10. Wait until finish of training job



The screenshot shows the AWS SageMaker console. A green banner at the top states: "Success! You created a training job. To track the status of the job, view details." The left sidebar shows the navigation menu with "Amazon SageMaker" selected. The main content area is titled "Training jobs" and contains a table with the following data:

Name	Creation time	Duration	Status
daal-log-reg-test	Feb 17, 2019 16:22 UTC	-	InProgress

A red arrow points to the "InProgress" status of the training job.

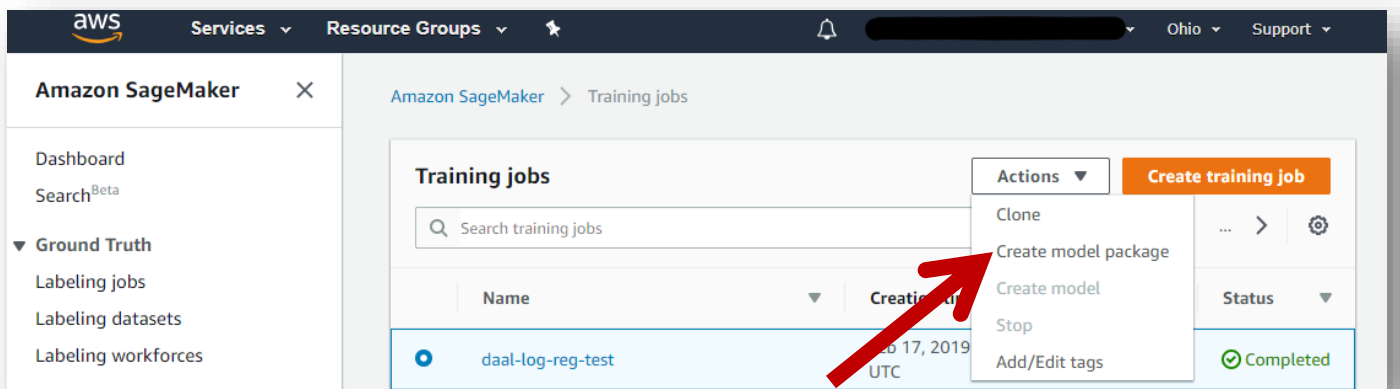


The screenshot shows the AWS SageMaker console. The training job "daal-log-reg-test" is now completed. The table data is as follows:

Name	Creation time	Duration	Status
daal-log-reg-test	Feb 17, 2019 16:22 UTC	3 minutes	Completed

A red arrow points to the "Completed" status of the training job.

## 11. Select training job and take action "Create model package"



The screenshot shows the AWS SageMaker console. The training job "daal-log-reg-test" is selected. The "Actions" dropdown menu is open, showing the following options: Clone, Create model package, Create model, Stop, and Add/Edit tags. A red arrow points to the "Create model package" option.

Name	Creation time	Duration	Status
daal-log-reg-test	Feb 17, 2019 16:22 UTC	3 minutes	Completed

12.Type model package name and click “Next”

## Create model package

### Inference specifications

#### Model package name and description

Model package name

daal-log-reg-test

The model package name must be unique in your account and in the AWS Region and can have up to 63 characters. Valid characters: a-z, A-Z, 0-9, and - (hyphen)

Description - *optional*

The description can be up to 1024 characters.

#### Inference specification options

- ☐ Provide the location of the inference image and model artifacts  
Choose this option if your model was trained using an algorithm stored in ECR.
- ☒ Provide the algorithm used for training and its model artifacts  
Choose this option if you are using a model trained by an algorithm resource or subscription algorithm from AWS Marketplace.

#### Algorithm and model artifacts

Algorithm ARN

Enter the Amazon Resource Name (ARN) used to create the training job and model artifacts.

arn:aws:sagemaker:us-east-2:057799348421:algorithm/intel-daal-logistic-regression-i

Location of model artifacts - *optional*

If you want buyers to use the model artifacts from a specific model, enter the path to the S3 bucket where they are stored.

s3://daal-log-reg-test/output/daal-log-reg-test/output/model.tar.gz

To find a path, [go to Amazon S3](#)

Cancel

Next

### 13. Click "Create model package"

Step 1  
Inference specifications

Step 2  
Validation specifications

## Create model package

### Validation specifications

To list your model package on AWS Marketplace, you must have it validated by Amazon SageMaker. Provide the information that Amazon SageMaker needs to run transform jobs to validate your product.

**Validation and scanning**

Publish this model package on AWS Marketplace  
AWS SageMaker requires successful validation before you can publish to AWS Marketplace.

☐ Yes ☒ No

Validate this resource  
AWS SageMaker will create a training job and/or transform job based on your validation profiles below.

☐ Yes ☒ No

Cancel Previous **Create model package**

### 14. Wait until package is created

Amazon SageMaker

Dashboard  
Search<sup>Beta</sup>

Ground Truth  
Labeling jobs  
Labeling datasets  
Labeling workforces

Notebook  
Notebook instances  
Lifecycle configurations  
Git repositories

Training  
Algorithms

**New model package created successfully.**

Amazon SageMaker > My model packages

My model packages | AWS Marketplace subscriptions

### Model packages

Actions Find model packages Create model package

Search model packages

Name	Status	ARN	Creation time
daal-log-reg-test	Completed	arn:aws:sagemaker:us-east-2:██████████:model-package/daal-log-reg-test	Feb 17, 2019 16:31 UTC

## 15. Select package and take action "Create endpoint"

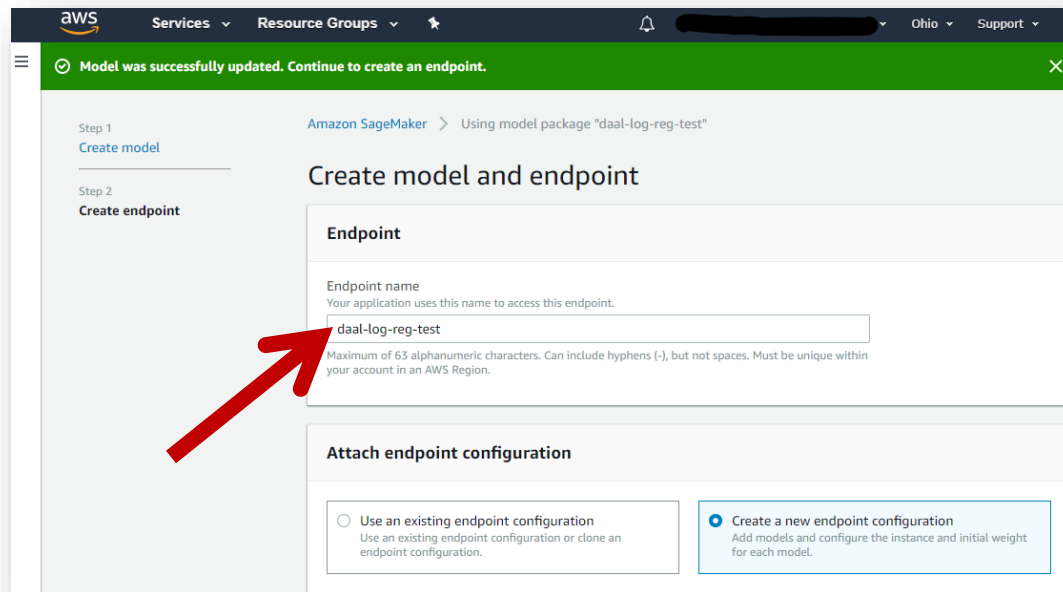
The screenshot shows the Amazon SageMaker console. At the top, a green banner indicates "New model package created successfully." The left sidebar shows the navigation menu with categories like Ground Truth, Notebook, and Training. The main content area is titled "My model packages" and shows a table of model packages. The package "daal-log-reg-test" is selected, and the "Actions" dropdown menu is open, showing options like "Clone model package", "Create model", "Create endpoint", "Publish new ML Marketplace listing", "Delete", and "Add/edit tags". A red arrow points to the "Create endpoint" option.

Name	Creation time
daal-log-reg-test	Feb 17, 2019 16:31 UTC

## 16. Type model name and click "Next"

The screenshot shows the "Create model and endpoint" page in the Amazon SageMaker console. The page has a sidebar with "Step 1 Create model" and "Step 2 Create endpoint". The main content area is titled "Create model and endpoint" and shows the "Model settings" section. The "Model name" input field is highlighted with a red arrow and contains the text "daal-log-reg-test". Below the input field, there is a note: "Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region." The "IAM role" section shows a dropdown menu with "AmazonSageMaker-ExecutionRole" selected.

17. Type endpoint name, edit and create endpoint configuration and click "Submit"



Model was successfully updated. Continue to create an endpoint.

Step 1  
Create model

Step 2  
Create endpoint

### Create model and endpoint

**Endpoint**

Endpoint name  
Your application uses this name to access this endpoint.

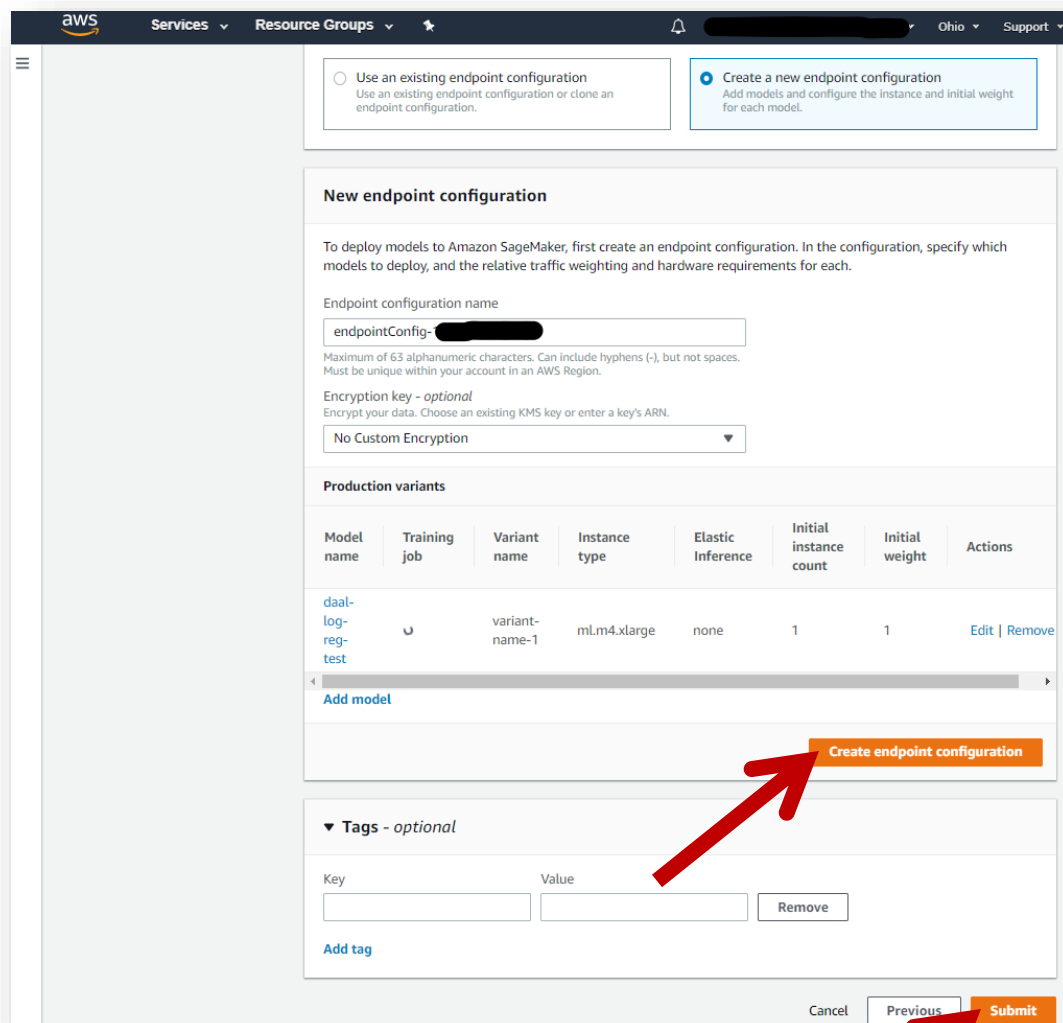
daal-log-reg-test

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

**Attach endpoint configuration**

☐ Use an existing endpoint configuration  
Use an existing endpoint configuration or clone an endpoint configuration.

☒ Create a new endpoint configuration  
Add models and configure the instance and initial weight for each model.



☐ Use an existing endpoint configuration  
Use an existing endpoint configuration or clone an endpoint configuration.

☒ Create a new endpoint configuration  
Add models and configure the instance and initial weight for each model.

### New endpoint configuration

To deploy models to Amazon SageMaker, first create an endpoint configuration. In the configuration, specify which models to deploy, and the relative traffic weighting and hardware requirements for each.

Endpoint configuration name

endpointConfig-[redacted]

Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Encryption key - optional  
Encrypt your data. Choose an existing KMS key or enter a key's ARN.

No Custom Encryption

**Production variants**

Model name	Training job	Variant name	Instance type	Elastic Inference	Initial instance count	Initial weight	Actions
daal-log-reg-test	u	variant-name-1	mLm4.xlarge	none	1	1	Edit   Remove

[Add model](#)

[Create endpoint configuration](#)

**Tags - optional**

Key Value Remove

[Add tag](#)

Cancel Previous [Submit](#)

## 18. Wait until endpoint is ready

Amazon SageMaker

Success! You created an endpoint. To track the status of the endpoint, view details.

Endpoints

Name	ARN	Creation time	Status	Last updated
daal-log-reg-test	arn:aws:sagemaker:us-east-2:123456789012:endpoint/daal-log-reg-test	Feb 17, 2019 16:43 UTC	Creating	Feb 17, 2019 16:43 UTC

Endpoints

Name	ARN	Creation time	Status	Last updated
daal-log-reg-test	arn:aws:sagemaker:us-east-2:123456789012:endpoint/daal-log-reg-test	Feb 17, 2019 16:43 UTC	InService	Feb 17, 2019 16:49 UTC

## 19. Use AWS CLI to get real-time prediction

Type command:

```
aws sagemaker-runtime invoke-endpoint --endpoint-name <endpoint-name> --body "$(cat <prediction_data_file_name>)" --content-type text/csv --accept text/csv <output_data_file_name>
```

```
(base) ubuntu@ip-172-31-22-46:~$ aws sagemaker-runtime invoke-endpoint --endpoint-name daal-knn-sm-test --body "$(cat probe_data.csv)" --content-type text/csv --accept text/csv output.txt
```

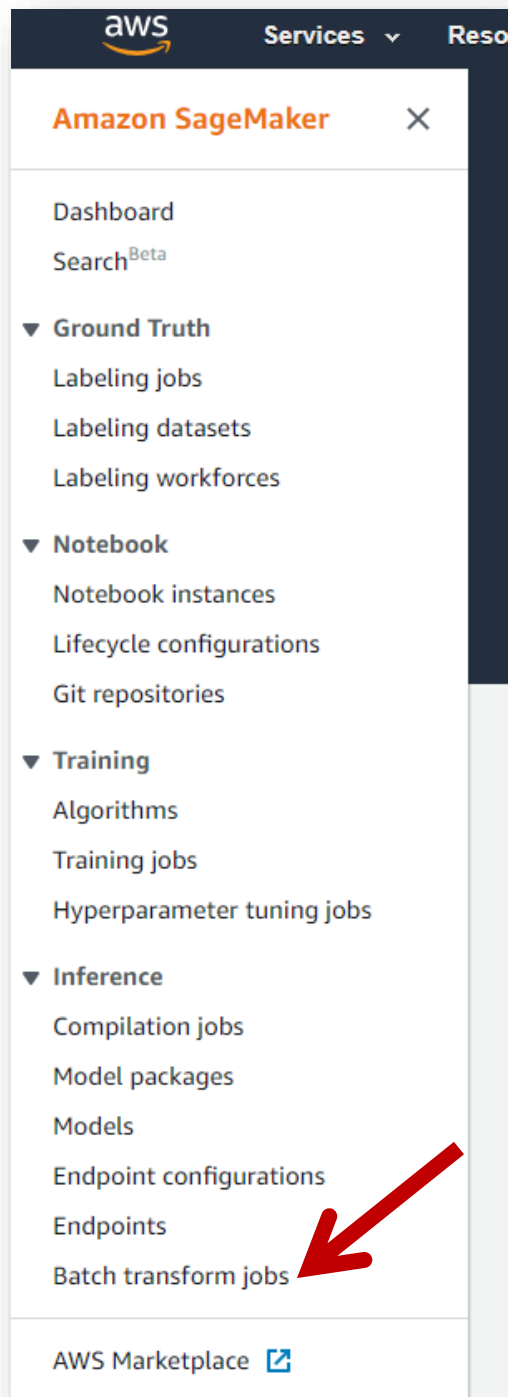
Then, see content of output file:

```
(base) ubuntu@ip-172-31-22-46:~$ cat output.txt
1
4
0
2
3
4
1
1
1
0
2
4
```

## Batch transform job as alternative to endpoint

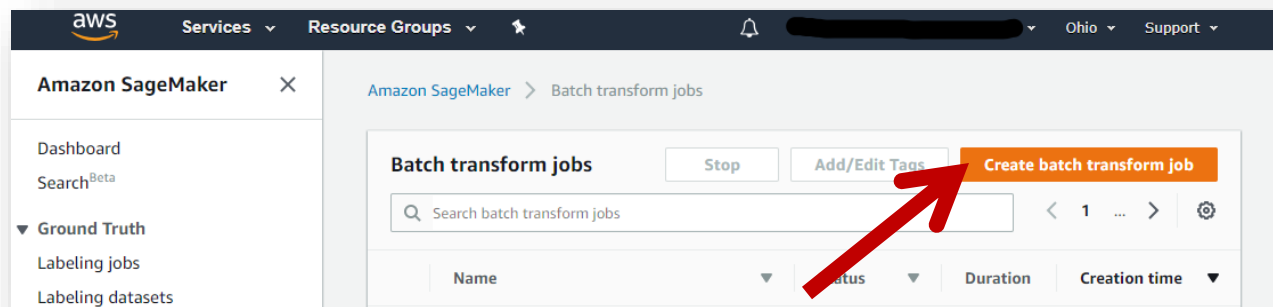
You can use batch transform job if you need compute predictions once.

1. Go to “Batch transform job” page

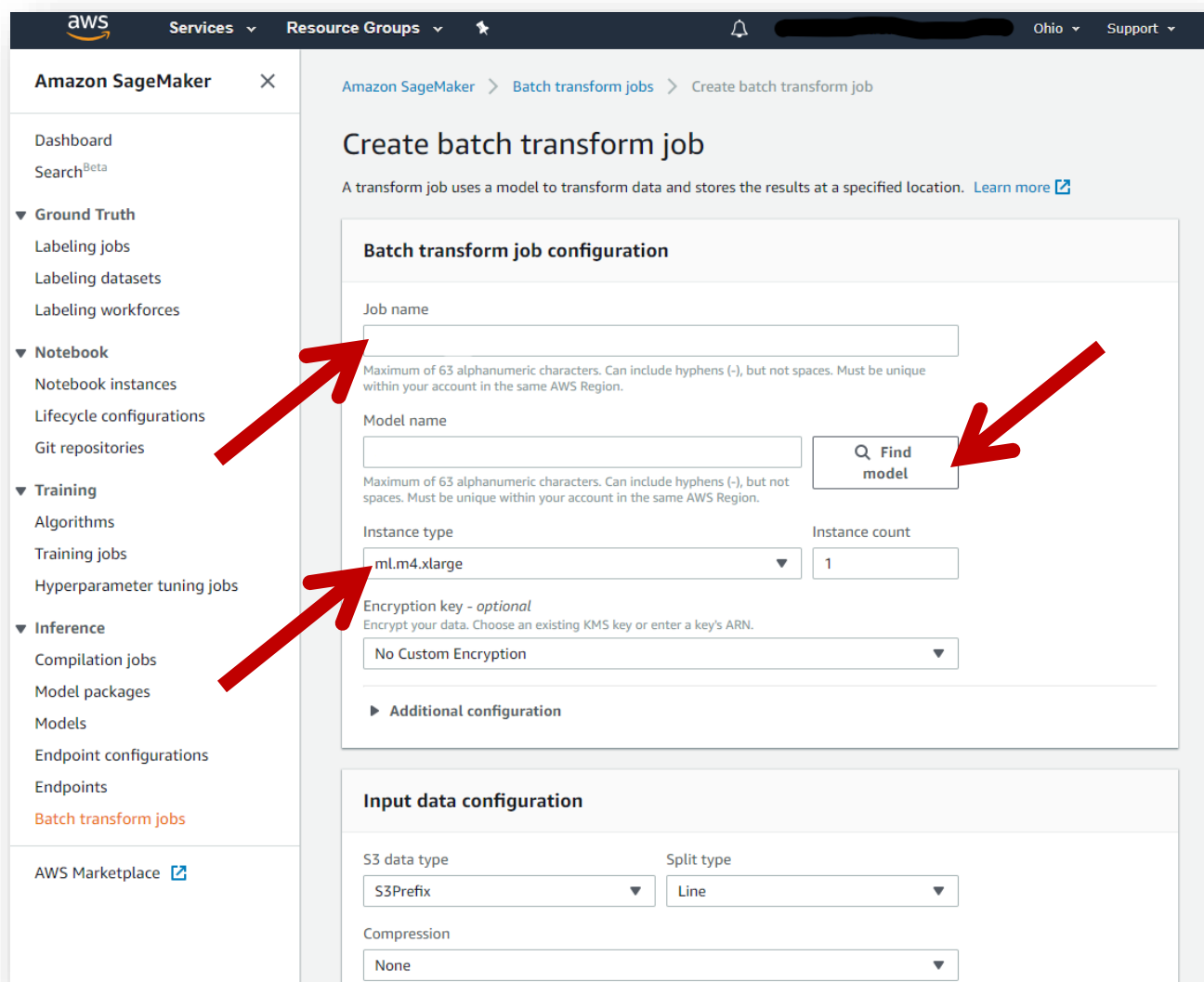




2. Click "Create batch transform job"



3. Enter job name, select previously created model, instance and set instance count to 1



- Specify S3 location of data for prediction, S3 output path (predictions will be stored here) and click "Create job"

**Amazon SageMaker**

Services ▾ Resource Groups ▾ Ohio ▾ Support ▾

Dashboard  
Search Beta

▼ **Ground Truth**  
Labeling jobs  
Labeling datasets  
Labeling workforces

▼ **Notebook**  
Notebook instances  
Lifecycle configurations  
Git repositories

▼ **Training**  
Algorithms  
Training jobs  
Hyperparameter tuning jobs

▼ **Inference**  
Compilation jobs  
Model packages  
Models  
Endpoint configurations  
Endpoints  
**Batch transform jobs**

AWS Marketplace [↗](#)

### Input data configuration

S3 data type:  Split type:

Compression:

Content type - *optional*:   
For content types that are available in built-in algorithms, [view our documentation](#)

S3 location:   
To find a path, [go to Amazon S3](#)

### Output data configuration

S3 output path:   
To find a path, [go to Amazon S3](#)

Encryption key - *optional*:   
An encryption key protects your data. Type the key ID or key ARN that you want to use.

Accept - *optional*:  Assemble with:

► **Tags - optional**

[Cancel](#) [Create job](#)

5. Wait until job is completed and find predictions in previously specified S3 output path

The screenshot shows the Amazon SageMaker console with a green notification banner at the top stating "Batch transform job daal-sm-test was successfully created". The left sidebar contains navigation links for Dashboard, Search<sup>Beta</sup>, Ground Truth, Labeling jobs, Labeling datasets, Labeling workforces, Notebook, and Notebook instances. The main content area displays "Batch transform jobs" with a search bar and a table. The table has columns for Name, Status, Duration, and Creation time. A single job is listed with the name "daal-sm-test" and a status of "InProgress", indicated by a clock icon. A red arrow points from the bottom of the first screenshot to the "InProgress" status in this screenshot.

Name	Status	Duration	Creation time
daal-sm-test	InProgress	a few seconds	Feb 17, 2019 14:32 UTC

This screenshot shows the same Amazon SageMaker console interface, but the batch transform job "daal-sm-test" now has a status of "Completed", indicated by a green checkmark icon. The duration is now "3 minutes" and the creation time remains "Feb 17, 2019 14:32 UTC". A red arrow points from the bottom of the second screenshot to the "Completed" status in this screenshot.

Name	Status	Duration	Creation time
daal-sm-test	Completed	3 minutes	Feb 17, 2019 14:32 UTC