

BACHELOR APPLIED COMPUTER SCIENCES

SEMESTER 6 ACADEMIC YEAR 2021 - 2022

LECTURERS: H. TERRYN, T. CLAUWAERT

---

# UDCP: AI-model opstellen met AWS SageMaker

Stav Heuvinck  
13/02/2022

---

# Inhoudstafel

Inhoudstafel	1
1. Inleiding	2
1.1. Algemeen	2
1.2. Probleemstelling	2
1.3. Onderzoeksvraag	2
1.4. Structuur van deze paper	3
2. Artificiële intelligentie	3
2.1. Definitie	3
2.2. Wat?	3
2.3. Waarom?	4
2.4. Vormen	4
3. AWS	4
3.1. Wat?	4
3.2. Sagemaker	5
4. Tutorial	6
4.1. Opzetten SageMaker	6
4.2. Lambda-functie	11
4.3. API-call	16
5. Conclusie	19
6. Bijlagen	20
7. Bronnenlijst	21

# 1. Inleiding

## 1.1. Algemeen

Artificiële intelligentie wordt steeds populairder alsook de complexiteit. Het trainen van simpele modellen gebeurt meestal in luttele seconden maar dit is niet altijd het geval. Wanneer men complexe modellen wil trainen kan dit uren tot dagen duren op je computer (tijd hangt af van sterkte pc). Tijdens het trainen kan je de computer nauwelijks gebruiken omdat hij al zijn resources gebruikt voor het trainen. Ook ben je niet zeker dat je na het trainen een gewenst resultaat bekomt en is al deze tijd verloren. Dit kost dus veel tijd en geld.

De laatste jaren hebben verschillende Cloud platformen zoals AWS, Azure, Google Cloud, Netlify, etc. een extra AI-functionaliteit toegevoegd aan hun assortiment.

Met deze functionaliteit kunnen ontwikkelaars machine learning modellen in de cloud maken, trainen en implementeren. Door deze toevoeging kan de ontwikkelaar meerdere modellen tegelijkertijd laten runnen en kan hij zich sneller aanpassen in functie van wat hij nodig heeft.

## 1.2. Probleemstelling

Het trainen van machine learning modellen lokaal op je pc wordt steeds ingewikkelder. Doordat de complexiteit en dus de rekenkracht die nodig is om een model te trainen steeds groter wordt is het bijna niet meer werkbaar. Daarom ben ik op zoek naar alternatieven om dit probleem op te lossen. In deze paper zal ik het specifiek hebben over het AWS Cloud platform alternatief en hier de voor- en nadelen van bespreken met een korte tutorial om je op weg te zetten.

## 1.3. Onderzoeksvraag

Uit de bovenstaande problemen volgt volgende onderzoeksvraag: Wat zijn de voordelen van het gebruik van een Cloud platform voor AI-development? We willen dus een duidelijk antwoord op het verschil tussen trainen in de cloud of lokaal .

Deze onderzoeksvraag kan opgedeeld worden in 2 deelvragen:

1. Hoe flexibel is een cloud platform?
2. Voor welke use cases zijn ze vooral geschikt?

***If someone asks me what cloud computing is, I try not to get bogged down with definitions. I tell them that, simply put, cloud computing is a better way to run your business.”***

***- Marc Benioff, Founder, CEO and Chairman, Salesforce***

## 1.4. Structuur van deze paper

In deze paper zal ik eerst een kleine toelichting doen over wat artificiële intelligentie is en wat men hiermee kan doen. Ook zal ik in het kort AWS, het grootste cloud platform, verder toelichten zodat u zich een duidelijk beeld kunt vormen over de verdere flow van deze paper.

Voor deze paper heb ik zelf ook een volledig uitgewerkte flow op AWS gemaakt zodat u elke stap kan volgen. Dit gaat van het aanmaken van het notebook tot het aanmaken van een API call voor het aanroepen van je model. Alvorens te starten wil ik duidelijk maken dat hiervoor meerdere mogelijke oplossingen zijn maar ik verkies deze werkwijze.

## 2. Artificiële intelligentie

### 2.1. Definitie

AI is de mogelijkheid van een machine om mensachtige vaardigheden te vertonen - zoals redeneren, leren, plannen en creativiteit.

Het is de intelligentie waarmee machines, software en apparaten zelfstandig problemen oplossen. Zij imiteren hierbij het denkvermogen van de mens. De apparaten zijn zich echter niet bewust van de taken die ze uitvoeren. Ze volgen algoritmes en herkennen patronen.

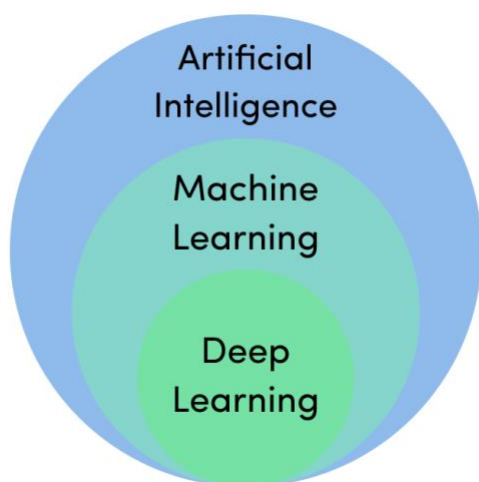
Iedereen heeft zijn eigen definitie van artificiële intelligentie. Ze kunnen allemaal juist zijn maar ik verkies deze van het Europees parlement (Zie bronvermelding).

### 2.2. Wat?

AI is eigenlijk de brede familie van technologieën die proberen menselijke intelligentie te vangen en na te bootsen. Dit bestaat uit 2 groepen:

Machine learning: geeft computers de mogelijkheid te leren zonder expliciet geprogrammeerd te worden.

Deep Learning: algoritmen leren met een hersenachtige logische structuur van algoritmen, genaamd kunstmatige neurale netwerken.



*Figuur 1 Artificiële intelligentie*

## 2.3. Waarom?

Artificiële intelligentie kan grote voordelen hebben voor de mensheid. De herkenning van spraak, beeld en patronen, de zelfsturende systemen, vertaalmachines, lopende robots en vraag-antwoordsystemen kunnen ons helpen op het gebied van gemak, gebruikerservaring en efficiëntie.

Ook de gezondheidszorg verwacht grootse dingen van AI. Een intelligente computer zou kleine veranderingen en afwijkingen veel nauwkeuriger kunnen waarnemen en herkennen dan een arts van vlees en bloed. Ook kan AI onvermoeibaar zeer complexe operaties uitvoeren.

## 2.4. Vormen

Er zijn verschillende vormen van AI waarmee we dagdagelijks in contact komen:

- softwarematige AI zoals virtuele assistentie, software voor beeldanalyse, zoekmachines, systemen voor spraak- en gezichtsherkenning;
- "belichaamde" AI zoals robots, zelfrijdende wagens, drones, IoT.

# 3. AWS

## 3.1. Wat?

AWS is een cloudplatform van de Amerikaanse e-commercegigant Amazon. Ze bieden een breed assortiment aan van software en services op het AWS-platform. Ze zijn hiervoor al sinds 2010 jaarlijks verkozen tot marktleider. AWS is zowel geschikt voor start-ups als voor multinationals omdat het een zeer snelle, flexibele IT-omgeving aanbiedt tegen zeer beperkte kosten.

Dit platform biedt aan softwareontwikkelaars enorm veel mogelijkheden. Ze kunnen direct gebruik maken van servers, opslag, netwerken, remote computing, mobiele ontwikkeling en beveiliging zonder dat zij het zware werk hoeven te doen.

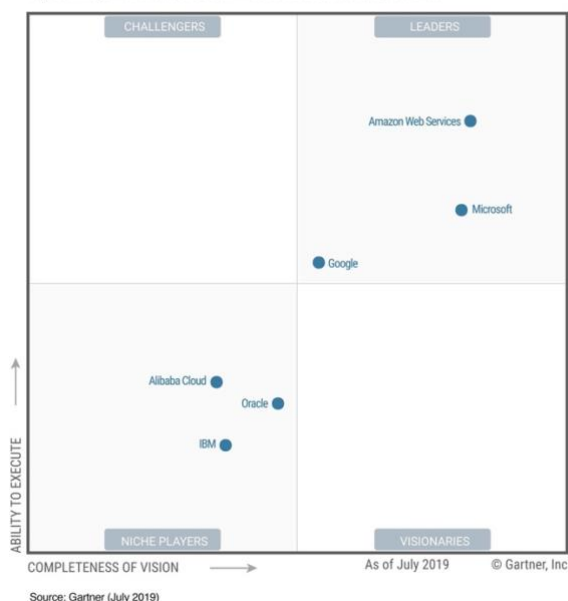
Met andere woorden is het een pay-as-you-go cloud computing-model waarbij je enkel hoeft te betalen voor wat je gebruikt. Je hoeft dus niet te betalen wanneer je de server niet nodig hebt of voor opslag die je niet gebruikt. Dit is wel het geval als je dit allemaal zelf opzet in je bedrijf.

Een ander groot voordeel van AWS is dat je zeer makkelijk kan schalen en bv minder of extra opslag kan vragen zonder moeilijkheden.

Enkele software/services die het meest gebruikt worden:

- AWS S3 (dataopslag)
- Amazon Relational Database Service voor bv SQL Server, PostgreSQL, MySQL
- Amazon connect (klantenservice)
- Amazon Sagemaker (machine learning)
- AWS Identity en Access Management (Authenticatie & Autorisatie)

Figure 1. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide

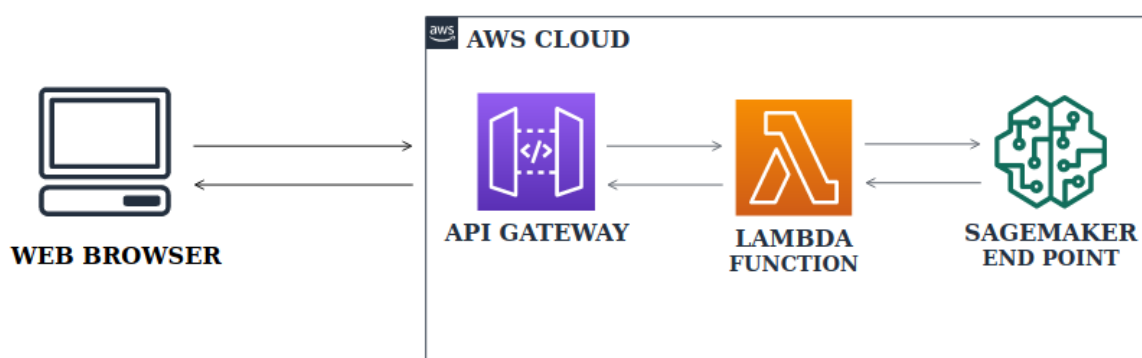


Figuur 2 Marktleider Cloud Platform

### 3.2. Sagemaker

Sagemaker is een service van AWS om machine learning modellen in de cloud te maken, trainen en implementeren. Je maakt een notebook instance aan met je favoriete framework en je laat het notebook runnen. Na het trainen en testen kan je je model deployen op een endpoint dat aangeroepen kan worden om een voorspelling te doen.

Vervolgens kan men aan deze endpoint een lambda functie hangen die aangeroepen kan worden uit andere services van AWS. In ons geval wordt deze gebruikt door de AWS API zodat wanneer wij een call doen, de body wordt doorgestuurd naar de lambda functie om op die gegevens een voorspelling te doen. Vervolgens wordt een resultaat teruggestuurd via de API-response.



Figuur 3 Flow Endpoint-Lambda-API

Ik hoop dat je nu een iets duidelijker beeld hebt van AI en het AWS-platform.

Om nu echt de kracht van een Cloud platform te zien gaan we stap voor stap bovenstaande afbeelding in de praktijk uitproberen.

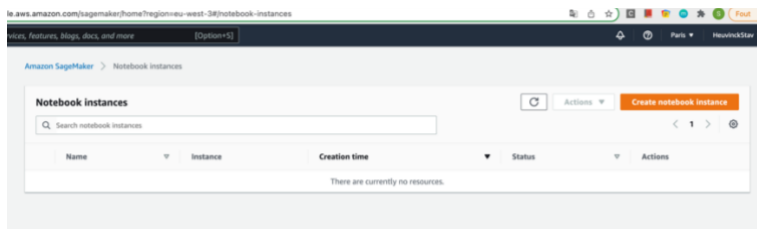
## 4. Tutorial

Deze tutorial bevat:

1. Notebook instance aanmaken in SageMaker
2. Data voorbereiden
3. Het model trainen op basis van een csv-bestand
4. Het model deployen op een endpoint
5. Het ML model evalueren
6. De lambda functie aan een endpoint hangen
7. De API-call creëren met trigger naar de lambda functie
8. Testen met postman/webapplicatie

### 4.1. Opzetten SageMaker

We starten met het aanmaken van een notebook instance, waar we het jupyter notebook en data kunnen in plaatsen.



*Figuur 4 Screenshot Notebook instance*

Amazon SageMaker > Notebook instances > Create notebook instance

## Create notebook instance

Amazon SageMaker provides pre-built fully managed notebook instances that run Jupyter notebooks. The notebook instances include example code for common model training and hosting exercises. [Learn more](#)

### Notebook instance settings

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

Notebook instance type  
ml.t2.medium

**Amazon SageMaker Notebook Instance is ending its standard support on Amazon Linux AMI (AL1). [Learn more](#)**

Platform identifier [Learn more](#)  
notebook-al1-v1

► Additional configuration

### Permissions and encryption

**IAM role**  
Notebook instances require permissions to call other services including SageMaker and S3. Choose a role or let us create a role with the [AmazonSageMakerFullAccess](#) IAM policy attached.  
Choose an IAM role

Don't see a role - not found

Figuur 5 Screenshot Notebook instance aanmaken

Geef het notebook een naam, in dit geval SageMaker-Tutorial.

Platform identifier [Learn more](#)  
notebook-al1-v1

► Additional configuration

### Permissions and encryption

**IAM role**  
Notebook instances require permissions to call other services including SageMaker and S3. Choose a role or let us create a role with the [AmazonSageMakerFullAccess](#) IAM policy attached.  
Choose an IAM role  
Create a new role  
Enter a custom role  
Use existing role

**Encryption key - optional**  
Encrypt your notebook data. Choose an existing KMS key or enter a key's ARN.  
No Custom Encryption

► Network - optional

► Git repositories - optional

► Tags - optional

Cancel Create notebook instance

Figuur 6 Notebook instance create role

Om de nodige rechten te hebben over het notebook en S3 (storage voor de data) moeten we een nieuwe rol creëren met de AmazonSageMakerFullAccess.

Klik op 'Create a new role' en er opent een nieuw venster.



**Create an IAM role**

Passing an IAM role gives Amazon SageMaker permission to perform actions in other AWS services on your behalf. Creating a role here will grant permissions described by the [AmazonSageMakerFullAccess](#) IAM policy to the role you create.

The IAM role you create will provide access to:

- ☒ **S3 buckets you specify - optional**
  - ☒ **Any S3 bucket**  
Allow users that have access to your notebook instance access to any bucket and its contents in your account.
  - ☐ **Specific S3 buckets**  
Example: bucket-name-1, bucket-name-2  
Comma delimited. ARNs, "\*" and "?" are not supported.
  - ☐ **None**
- ☒ Any S3 bucket with "sagemaker" in the name
- ☒ Any S3 object with "sagemaker" in the name
- ☒ Any S3 object with the tag "sagemaker" and value "true" [See Object tagging](#)
- ☒ S3 bucket with a Bucket Policy allowing access to SageMaker [See S3 bucket policies](#)

Figuur 7 Notebook instance create bucket

Maak een S3 bucket aan voor het opslaan van alle data. Je kan ook kiezen voor een specifieke bucket als je reeds een bucket hebt aangemaakt.

Klik op 'Create role'.

**Permissions and encryption**

**IAM role**  
Notebook instances require permissions to call other services including SageMaker and S3. Choose a role or let us create a role with the [AmazonSageMakerFullAccess](#) IAM policy attached.  
AmazonSageMaker-ExecutionRole-20220212T185398

☒ **Success! You created an IAM role.**  
AmazonSageMaker-ExecutionRole-20220212T185398

**Root access - optional**  
☒ **Enable** - Give users root access to the notebook  
☐ **Disable** - Don't give users root access to the notebook  
Lifecycle configurations always have root access

**Encryption key - optional**  
Encrypt your notebook data. Choose an existing KMS key or enter a key's ARN.  
No Custom Encryption

**Network - optional**

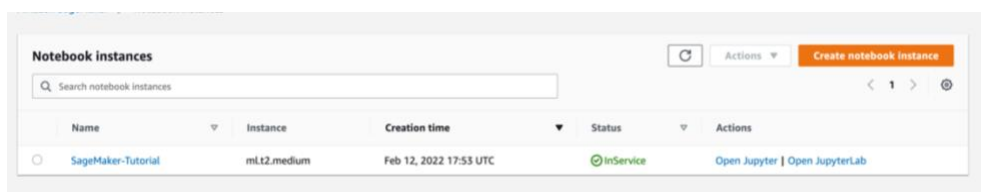
**Git repositories - optional**

**Tags - optional**

Figuur 8 Create notebook instance

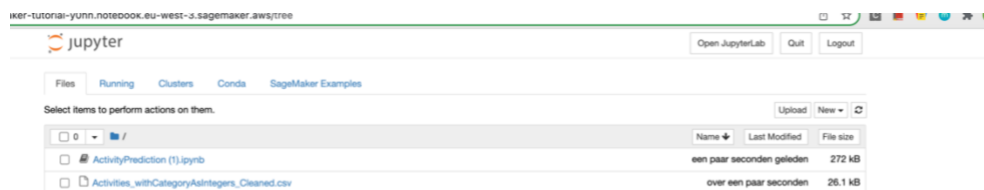
Wanneer dit gelukt is krijg je volgend scherm en is de bucket succesvol aangemaakt.

Klik op 'Create notebook instance' om verder te gaan.



Figuur 9 Status notebook instance

Het aanmaken van het notebook kan enkele minuten duren. Van zodra de status van 'Pending' naar 'InService' veranderd kan je het notebook gebruiken.



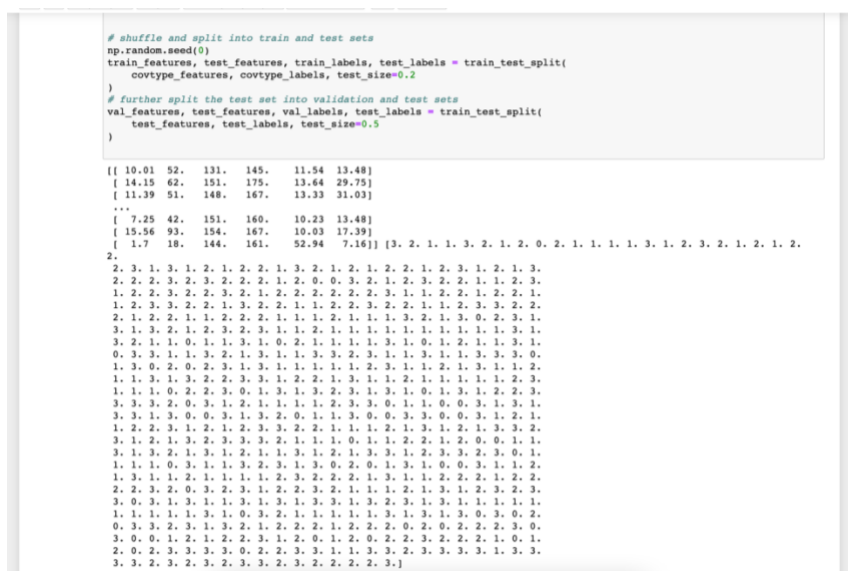
Figuur 10 Jupyter notebook & csv-bestand

Klik op het notebook en je wordt doorgestuurd naar bovenstaande pagina. Hier kan je een jupyter notebook aanmaken voor de data cleaning en het trainen van je model.

Je kan starten met mijn voorgemaakt notebook en csv-bestand om deze tutorial te volgen.

Link github:

<https://github.com/stavheuinck/ActivityPrediction>



Figuur 11 Info data

In bovenstaand afbeelding kan je zien dat een record bestaat uit de afstand, aantal minuten, gem. hartslag, max. hartslag, gem. snelheid en max. snelheid. Ook staat er bij elk record een classificatielabel op basis van de verschillende soorten intensiteit zodat de men tijdens het trainen de correlaties voor elk label kan bepalen en op nieuwe data kan voorspellingen doen.

```
In [10]: import sagemaker
from sagemaker.amazon_estimator import RecordSet
import boto3

# Instantiate the LinearLearner estimator object
multiclass_estimator = sagemaker.LinearLearner(
    role=sagemaker.get_execution_role(),
    train_instance_count=1,
    train_instance_type="ml.m5.large",
    predictor_type="multiclass_classifier",
    num_classes=7,
)

train_instance_count has been renamed in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
train_instance_type has been renamed in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.

In [11]: # wrap data in RecordSet objects
train_records = multiclass_estimator.record_set(train_features, train_labels, channel="train")
val_records = multiclass_estimator.record_set(val_features, val_labels, channel="validation")
test_records = multiclass_estimator.record_set(test_features, test_labels, channel="test")

In [12]: # start a training job
multiclass_estimator.fit([train_records, val_records, test_records])

Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: 1.
Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: 1.

2022-01-19 10:28:35 Starting - Starting the training job...
2022-01-19 10:28:57 Starting - Launching requested ML instancesProfilerReport-1642580115: InProgress
...
2022-01-19 10:29:28 Starting - Preparing the instances for training.....
2022-01-19 10:31:29 Downloading - Downloading input data
2022-01-19 10:31:29 Training - Downloading the training image..Docker entrypoint called with argument(s): train
Running default environment configuration script
[01/19/2022 10:31:49 INFO 139671436584768] Reading default configuration from /opt/amazon/lib/python3.7/site-package
s/algorithms/resources/default-input.json: {'mini_batch_size': '1000', 'epochs': '15', 'feature_dim': 'auto', 'use_bia
s': 'true', 'binary_classifier_model_selection_criteria': 'accuracy', 'f_beta': '1.0', 'target_recall': '0.8', 'targ
t_precision': '0.8', 'num_models': 'auto', 'num_calibration_samples': '10000000', 'init_method': 'uniform', 'init_sca
le': '0.07', 'init_sigma': '0.01', 'init_bias': '0.0', 'optimizer': 'auto', 'loss': 'auto', 'margin': '1.0', 'quantil
e': '0.5', 'loss_insensitivity': '0.01', 'huber_delta': '1.0', 'num_classes': '1', 'accuracy_top_k': '3', 'wd': 'aut
o', 'l1': 'auto', 'momentum': 'auto', 'learning_rate': 'auto', 'beta_1': 'auto', 'beta_2': 'auto', 'bias_lr_mult': 'a
uto', 'bias_wd_mult': 'auto', 'use_lr_scheduler': 'true', 'lr_scheduler_step': 'auto', 'lr_scheduler_factor': 'auto',
```

Figuur 12 trainen model

Daarna start het trainen. Ik gebruik in dit voorbeeld een simpele Linear Learner om het niet te ingewikkeld te maken en niet te veel tijd te verliezen met trainen.

```
In [14]: from sagemaker.serializers import CSVSerializer
multiclass_predictor = multiclass_estimator.deploy(
    initial_instance_count=1,
    instance_type="ml.m5.large",
    serializer=CSVSerializer()
)

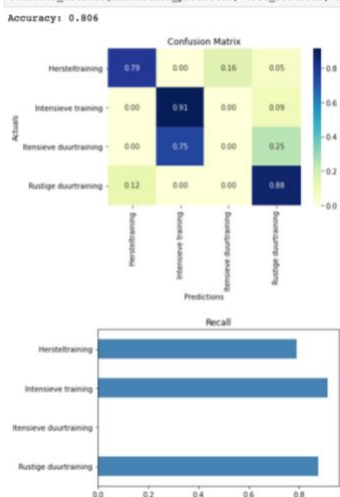
Defaulting to the only supported framework/algorithm version: 1. Ignoring framework/algorithm version: 1.
-----!

In [15]: multiclass_predictor.endpoint_name
Out[15]: 'linear-learner-2022-01-19-10-33-21-264'
```

Figuur 13 endpoint aanmaken

Na het trainen plaatsen we het model op een endpoint zodat het aangeroepen kan worden en op inkomende data via deze endpoint voorspellingen kan maken.

```
In [17]: # evaluate metrics of the model trained with default hyperparameters
evaluate_metrics(multiclass_predictor, test_features, test_labels)
```



Figuur 14 evaluatie model

Hier zie je dat ons model na 5 seconden toch al in gemiddeld 80% van de gevallen het juiste label voorspelt.

```
In [14]: print(test_features[5], test_labels[5])

from sagemaker.predictor import csv_serializer, json_deserializer
multiclass_predictor.serializer = csv_serializer
multiclass_predictor.deserializer = json_deserializer

result = multiclass_predictor.predict(test_features[5])
print(result)

The csv_serializer has been renamed in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.
The json_deserializer has been renamed in sagemaker>=2.
See: https://sagemaker.readthedocs.io/en/stable/v2.html for details.

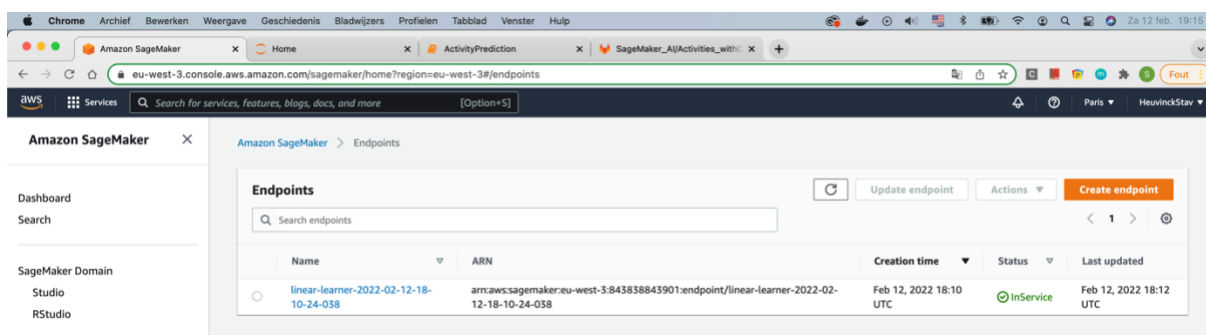
[ 5.01  30.  145.  178.   9.94  11.11] 3.0
{'predictions': [{'score': [0.14249373972415924, 0.09826767444610596, 0.20381565392017365, 0.5551199913024902, 0.00013507893891073763, 7.877222378738225e-05, 8.905459981178865e-05], 'predicted_label': 3}]}
```

```
In [31]: multiclass_predictor.delete_endpoint()

In [ ]: sagemaker_session = sagemaker.Session()
role = sagemaker.get_execution_role()
```

Figuur 15 Test model

Tot slot zie je dat het model 3 voorspelt en het effectieve label ook 3 is.

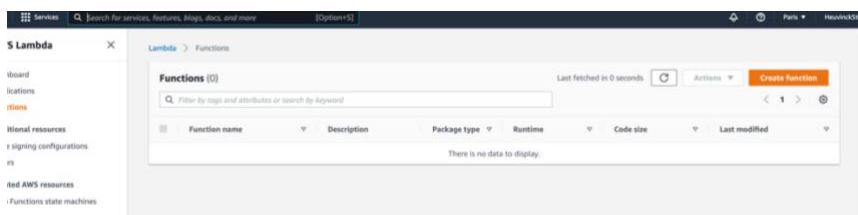


Figuur 16 Check endpoint

Om te controleren of de endpoint effectief is aangemaakt kan je in het linkse menu bij 'Inference' alle endpoints bekijken.

Op dit moment hebben we een perfect werkend model dat voorspellingen kan doen en dat kan aangeroepen worden via een endpoint. Het enige dat we nu nog hoeven te doen is een lambda functie koppelen aan de endpoint en het laten triggeren door een API-call.

## 4.2. Lambda-functie



Figuur 17 Functie pagina

Zoek in de zoekbalk naar AWS lambda en maak bij functies een nieuwe functie aan.

Services Search for services, features, blogs, docs, and more [Optional]

## Create function info

Choose one of the following options to create your function.

**Author from scratch** info  
 Start with a simple Hello World example.

Use a blueprint  
 Build a Lambda application from sample code and configuration presets for common use cases.

Container image  
 Select a container image to deploy for your function.

Browse serverless app repository  
 Deploy a sample Lambda application from the AWS Serverless Application Repository.

### Basic information

**Function name**  
 Enter a name that describes the purpose of your function.  
  
Use only letters, numbers, hyphens, or underscores with no spaces.

**Runtime** info  
 Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

**Permissions** info  
 By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

**▼ Change default execution role**

**Execution role**  
 Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#).

☒ Create a new role with basic Lambda permissions  
☐ Use an existing role  
☐ Create a new role from AWS policy templates

Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Figuur 18 Functie aanmaken

Selecteer 'Author from scratch' en geef de functie een naam. Zorg dat deze naam betekenis heeft want deze zal nog een paar keer gebruikt worden in het verdere verloop. Ik gebruik hier 'GetActivityPrediction'.

Vervolgens kan je kiezen voor je favoriete programmeertaal. Ik kies hier voor python maar het kan evengoed met JavaScript.

Laat de rest op default staan en klik op 'Create function'.

Function: [arn:aws:lambda:eu-west-3:43838843901:function:GetActivityPrediction](#)

Code Test Monitor **Configuration** Aliases Versions

**General configuration**

Triggers

Permissions

Destinations

**Environment variables**

Tags

VPC

Monitoring and operations tools

**Environment variables** (0) [Edit](#)

Key	Value
No environment variables	
No environment variables associated with this function.	

[Edit](#)

Figuur 19 Environmental variable

**Edit environment variables**

**Environment variables**

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#)

Key	Value	
ENDPOINT_NAME	https://learn-2022-02-12-18-10-24-0	<a href="#">Remove</a>

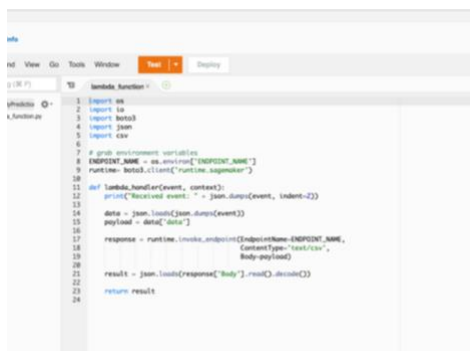
[Add environment variable](#)

**Encryption configuration**

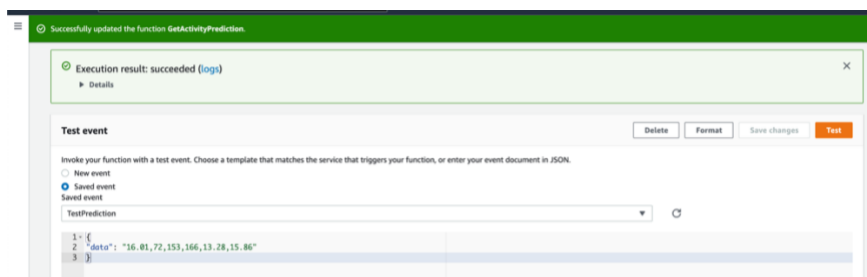
[Cancel](#) [Save](#)

Figuur 20 Environmental variable aanmaken

Om makkelijk en veilig de endpoint aan te spreken maken we een omgevingsvariabele aan met als value de link naar de endpoint. Wanneer we deze link dus nodig hebben gebruiken we altijd deze variabele.



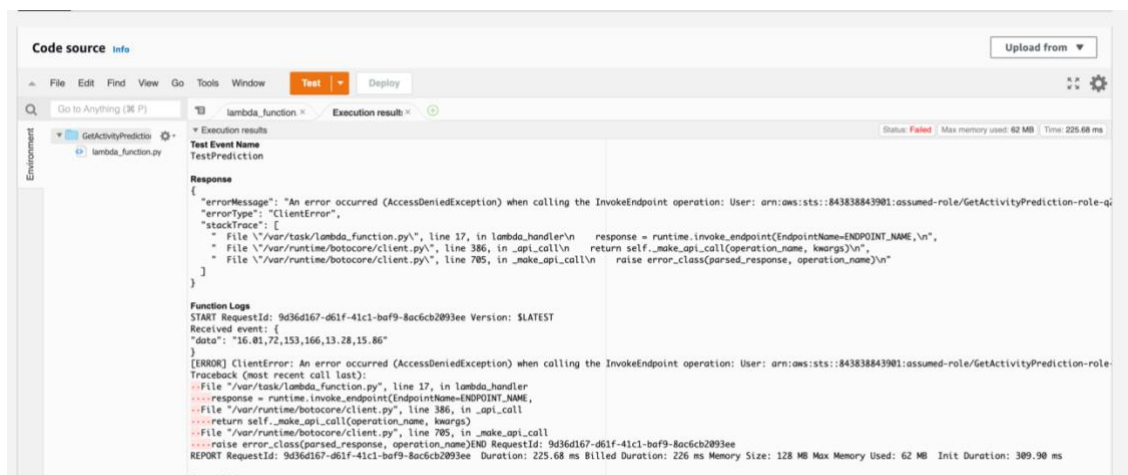
Figuur 21 Code lambda functie



Figuur 22 Test lambda functie

Ik heb een klein scriptje geschreven dat je kan kopiëren vanuit github. Het laadt de json data in en zet het om naar het juiste formaat om in het model te steken. Vervolgens zet het de response terug om in json om naar de client te sturen.

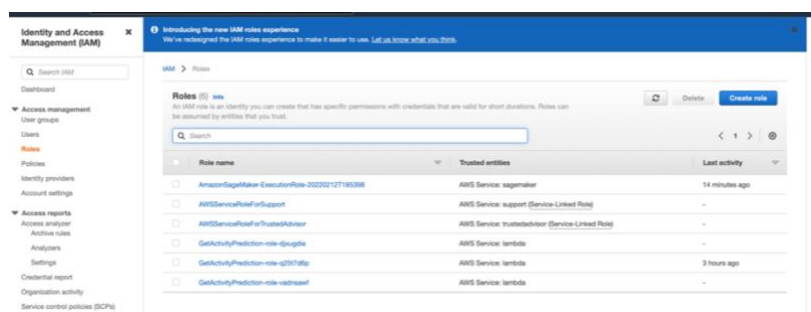
Daarna schrijven we een korte test om de lambda functie te testen met een meegegeven body zoals hij die zal ontvangen via de API.



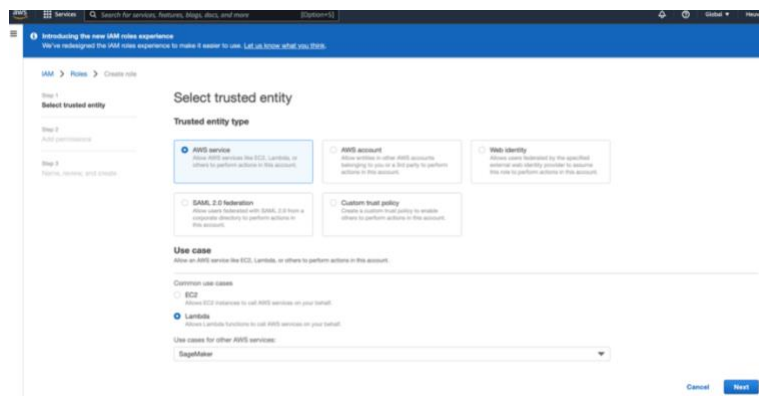
Figuur 23 Error lambda aanroepen lambda

Klik vervolgens op Deploy en Test. Je zal merken dat er een fout optreedt omdat je geen rechten hebt om de endpoint van SageMaker aan te roepen en te gebruiken.

Om dit op te lossen moeten we een nieuwe rol aanmaken met de AmazonSageMakerFullAccess rechten zodat lambda dit kan gebruiken. Dit wordt gedaan in AWS IAM service.



Figuur 24 Nieuwe rol aanmaken

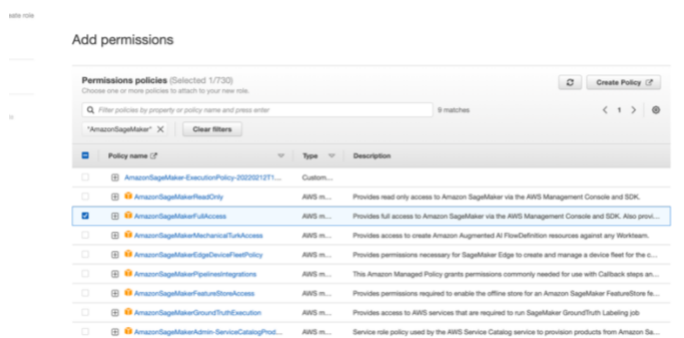


Figuur 25 Nieuwe rol met full access

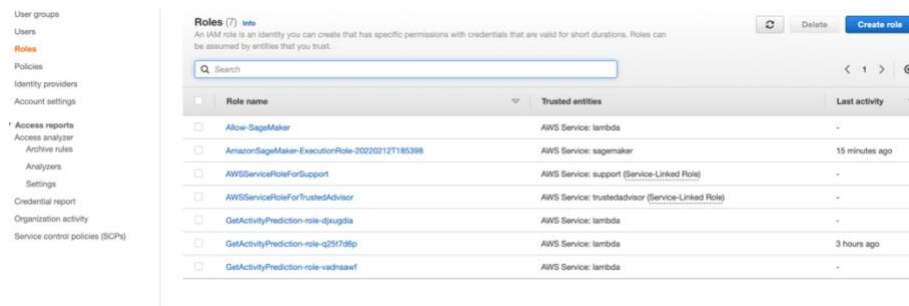
Creëer een nieuwe rol met AWS service en selecteer lambda.

Als service neem je SageMaker.

Selecteer nu AmazonSageMakerFullAccess als permissions en maak een rol aan.



Figuur 26 Full SageMaker access

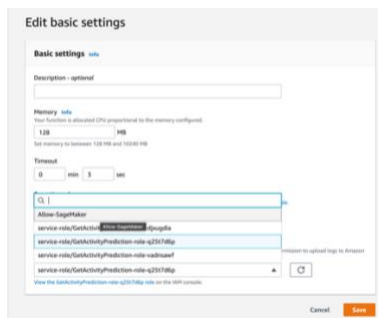


Wanneer alles gelukt is heb je nu een nieuwe rol die ik Allow-SageMaker heb genoemd.

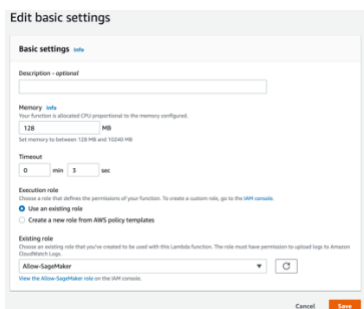
Ga nu terug naar AWS lambda en verander in 'configuration permissions' de execution role naar Allow-Sagemaker.



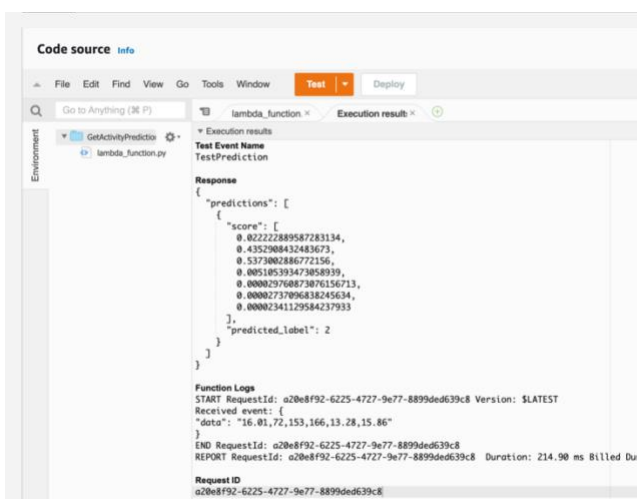
Figuur 27 Verander execution role



Figuur 28 Verander execution role



Figuur 29 Verander execution role



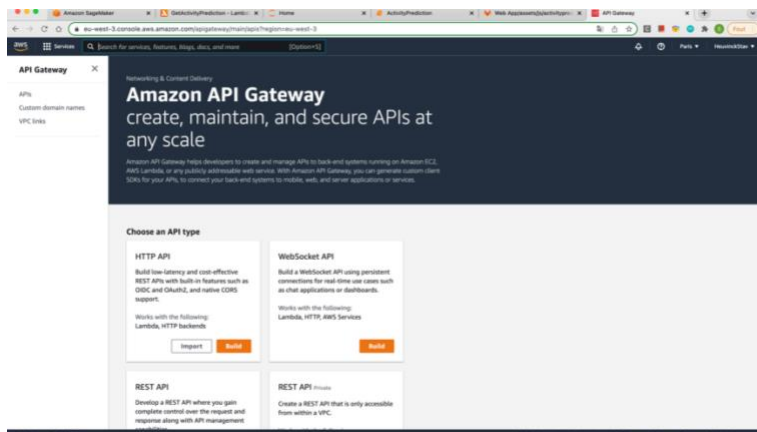
Figuur 30 Test endpoint

Wanneer we nu de test opnieuw uitvoeren werkt alles perfect en krijgen we een response.



### 4.3. API-call

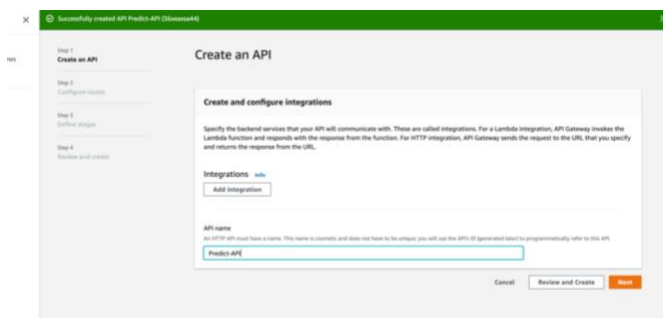
Tot slot moeten we enkel nog een API-call koppelen aan de lambda functie.



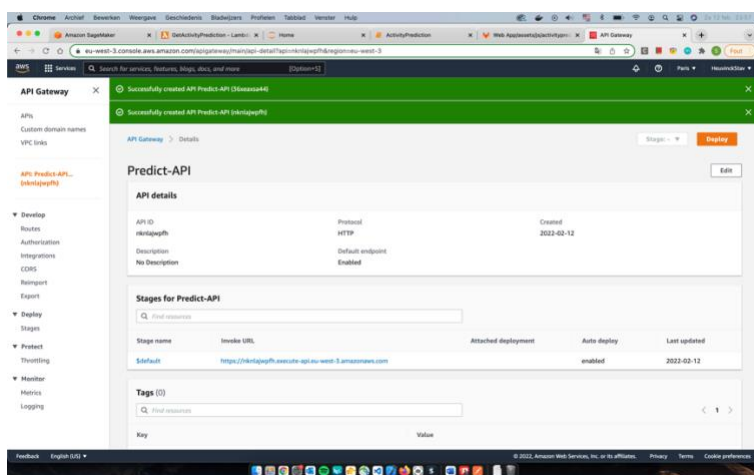
Figuur 31 API aanmaken

Maak een simpele HTTP API aan. Geef het de naam Predict-API en laat alle instellingen op default staan.

Op de laatste afbeelding zie je de URL staan waaraan we de API-call gaan koppelen.

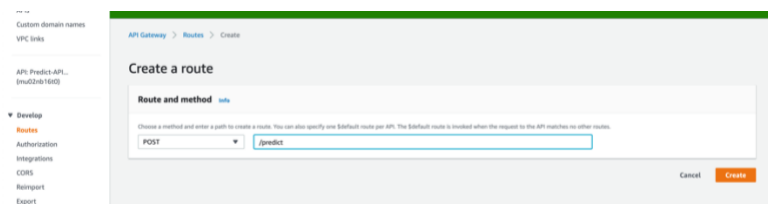


Figuur 32 API aanmaken



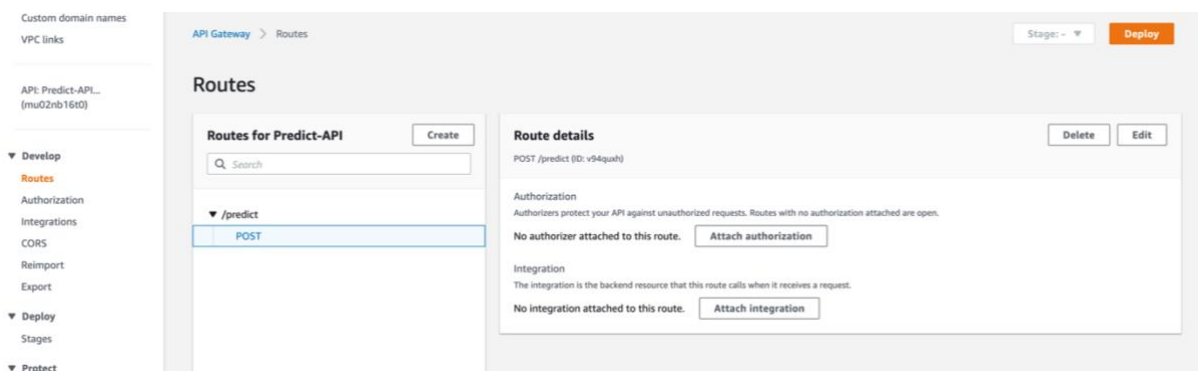
Figuur 33 API Aanmaken

Bij routes maak je een nieuwe POST route aan met pad /predict



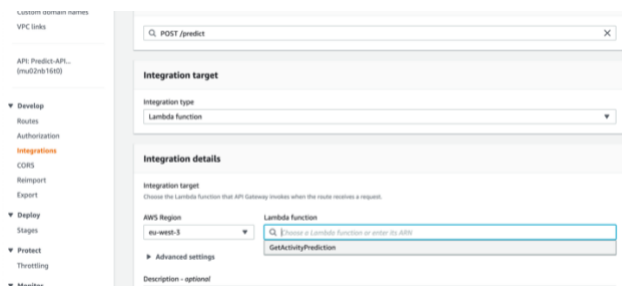
Figuur 34 Aanmaken route

Wanneer deze is aangemaakt klikken we op 'Attach integration' voor de koppeling naar de lambda functie.

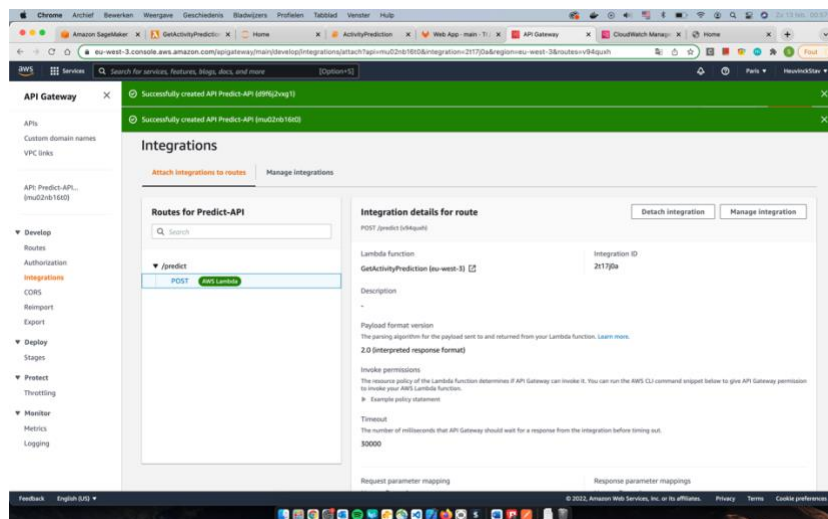


Figuur 35 Lambda functie aan route hangen

Selecteer bovenaan 'lambda function' en in het dropdown menu kies je voor onze zelf aangemaakte 'GetActivityPrediction'.



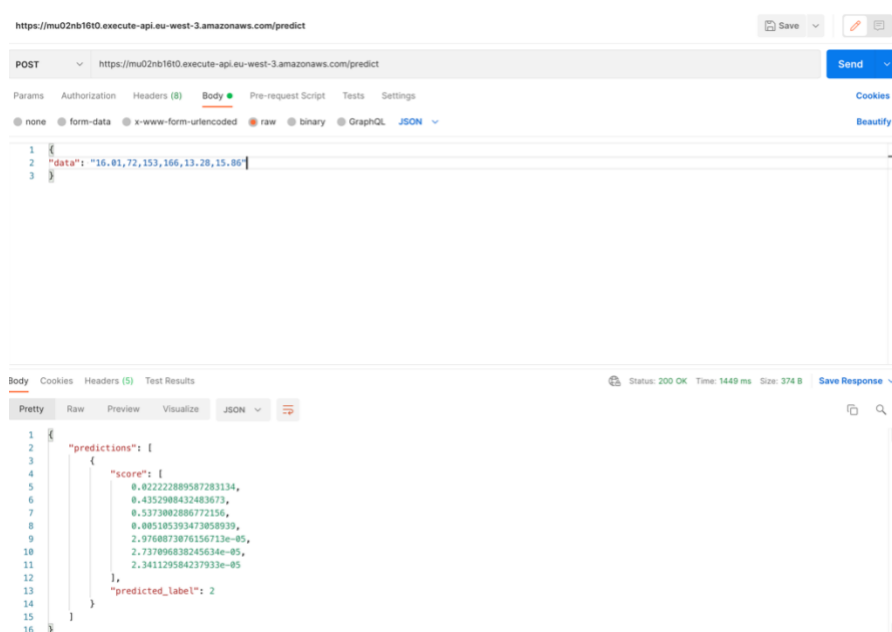
Figuur 36 Lambda functie aan route hangen



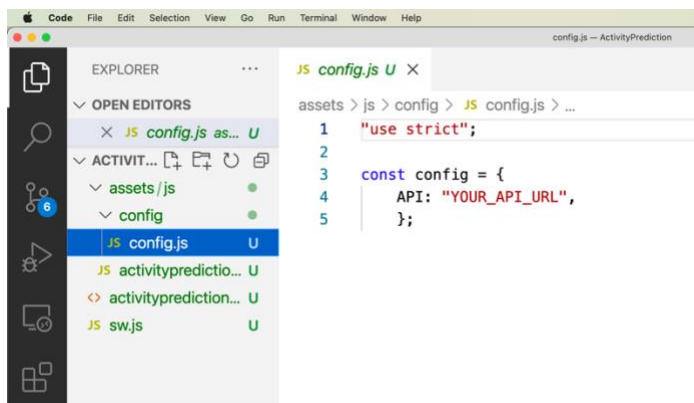
Figuur 37 Lambda functie aan route hangen

Wanneer dit gelukt is hebben we een perfect werkende flow tussen 3 AWS-services. Je kan dit gerust testen in postman. Ook heb ik een kleine applicatie geschreven waar we de API-call gebruiken en je zelf de parameters kan ingeven.

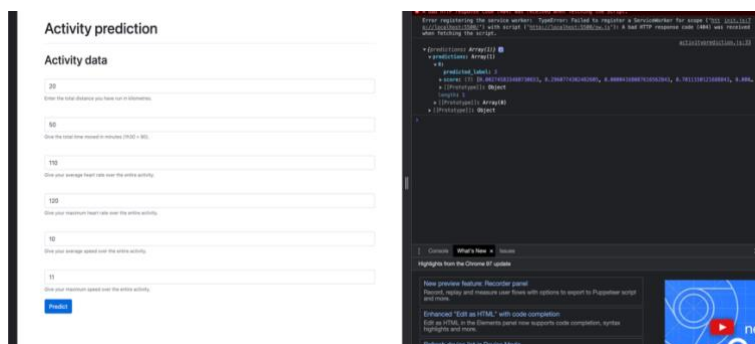
Via Postman kan je de http-call testen met de volledige URL en de vereiste body.



Figuur 38 API test postman



Figuur 39 API test webapp



Figuur 40 API test webapp

Als je mijn naamgeving gebruikt hebt moet je enkel in de config file je http url aanpassen en kan je de webapplicatie te testen.

De code die ik geschreven heb kan je terugvinden op mijn github repo:

<https://github.com/stavheuinck/ActivityPrediction>

## 5. Conclusie

We kunnen besluiten dat het maken, trainen en implementeren van een AI-model in de cloud vele voordelen heeft tegenover lokaal. Zo kan men meerdere modellen met verschillende parameters tegelijkertijd laten lopen en vervolgens een betere evaluatie doen van het getrainde resultaat. Wanneer men dit lokaal zou willen doen verlies je gemakkelijk een paar dagen tijd evenals het gebruik van je computer. In de cloud kan je ook veel beter schalen naargelang de grootte van de data, wat lokaal op je computer voor problemen kan zorgen.

Een ander groot voordeel van het werken met AWS is de mogelijkheid om SageMaker makkelijk te combineren met andere AWS services zoals in onze tutorial lambda en HTTP API. Wanneer je dit lokaal zou willen doen moet je zelf een server opzetten en hier kruipt ook wat tijd in.

We kunnen dus stellen dat het werken met een cloud platform voor een AI-ontwikkelaar veel tijd kan besparen en dit tegen een beperkte kost. Echter als men alleen kleine projectjes maakt zal dit wel wat duurder uitkomen. Ook is de flexibiliteit van het platform, waar het in vele use cases (veel verschillende algoritmen) kan gebruikt en geschaald worden en dit voor elke grootte van data, een echt pluspunt zonder dat het plaats of tijd van je computer afneemt.

Ik hoop dat ik je heb kunnen overtuigen van het gemak van een cloud platform en welke voordelen het jou kan bieden.

## 6. Bijlagen

Figuur 1 Artificiële intelligentie .....	3
Figuur 2 Marktleider Cloud Platform .....	5
Figuur 3 Flow Endpoint-Lambda-API.....	5
Figuur 4 Screenshot Notebook instance .....	6
Figuur 5 Screenshot Notebook instance aanmaken .....	7
Figuur 6 Notebook instance create role.....	7
Figuur 7 Notebook instance create bucket.....	8
Figuur 8 Create notebook instance .....	8
Figuur 9 Status notebook instance.....	9
Figuur 10 Jupyter notebook & csv-bestand .....	9
Figuur 11 Info data .....	9
Figuur 12 trainen model.....	10
Figuur 13 endpoint aanmaken .....	10
Figuur 14 evaluatie model.....	10
Figuur 15 Test model.....	11
Figuur 16 Check endpoint .....	11
Figuur 17 Functie pagina .....	11
Figuur 18 Functie aanmaken .....	12
Figuur 19 Environmental variable .....	12
Figuur 20 Environmental variable aanmaken .....	12
Figuur 21 Code lambda functie .....	13
Figuur 22 Test lambda functie.....	13
Figuur 23 Error lambda aanroepen lambda .....	13
Figuur 24 Nieuwe rol aanmaken .....	14
Figuur 25 Nieuwe rol met full access .....	14
Figuur 26 Full SageMaker access.....	14
Figuur 27 Verander execution role .....	15
Figuur 28 Verander execution role .....	15
Figuur 29 Verander execution role .....	15
Figuur 30 Test endpoint .....	15
Figuur 31 API aanmaken .....	16
Figuur 32 API aanmaken .....	16
Figuur 33 API Aanmaken .....	16
Figuur 34 Aanmaken route.....	17
Figuur 35 Lambda functie aan route hangen.....	17
Figuur 36 Lambda functie aan route hangen.....	17

Figuur 37 Lambda functie aan route hangen .....	18
Figuur 38 API test postman .....	18
Figuur 39 API test webapp .....	19
Figuur 40 API test webapp .....	19

## 7. Bronnenlijst

- Europees parlement. (2021, 29 maart). *Wat is artificiële intelligentie en hoe wordt het gebruikt?* / Nieuws / Europees Parlement. Nieuws Europees parlement.  
Geraadpleegd op 13 februari 2022, van [https://www.europarl.europa.eu/news/nl/headlines/society/20200827STO85804/wat-is-artificiele-intelligentie-en-hoe-wordt-het-gebruikt#:~:text=Definitie%20van%20artifici%C3%ABle%20intelligentie%20\(of,%2C%20leren%2C%20plannen%20en%20creativiteit.](https://www.europarl.europa.eu/news/nl/headlines/society/20200827STO85804/wat-is-artificiele-intelligentie-en-hoe-wordt-het-gebruikt#:~:text=Definitie%20van%20artifici%C3%ABle%20intelligentie%20(of,%2C%20leren%2C%20plannen%20en%20creativiteit.)
- Steeman, G. (2021, 23 november). *Kunstmatige intelligentie*. Mediawijsheid.nl.  
Geraadpleegd op 13 februari 2022, van <https://www.mediawijsheid.nl/kunstmatigeintelligentie/>
- *Call an Amazon SageMaker model endpoint using Amazon API Gateway and AWS Lambda*. (2021, 11 april). Amazon Web Services. Geraadpleegd op 13 februari 2022, van <https://aws.amazon.com/blogs/machine-learning/call-an-amazon-sagemaker-model-endpoint-using-amazon-api-gateway-and-aws-lambda/>
- *How to build, train, and deploy a machine learning model with Amazon SageMaker / AWS*. (2021, 7 juli). Amazon Web Services, Inc. Geraadpleegd op 13 februari 2022, van <https://aws.amazon.com/getting-started/hands-on/build-train-deploy-machine-learning-model-sagemaker/>
- Zimmerman, Y. (2021, 15 december). *Saving 95% on infrastructure costs using AWS Lambda for scikit-learn predictions*. Towards Data Science. Geraadpleegd op 13

februari 2022, van <https://towardsdatascience.com/saving-95-on-infrastructure-costs-using-aws-lambda-for-scikit-learn-predictions-3ff260a6cd9d>

- Wisman, R. (2021, 8 april). *Wat is Amazon Web Services (AWS) + 5 belangrijkste toepassingen*. Dataweb. Geraadpleegd op 13 februari 2022, van <https://www.dataweb.nl/wat-is-amazon-web-services/>
- Arne Wolfewicz. (2021, 9 november). *Deep learning vs. machine learning – What's the difference?* Levity. Geraadpleegd op 13 februari 2022, van <https://levity.ai/blog/difference-machine-learning-deep-learning>
- *Introduction to Amazon SageMaker (4:47)*. (z.d.). Amazon Web Services, Inc. Geraadpleegd op 13 februari 2022, van [https://aws.amazon.com/pm/sagemaker/?trk=ps\\_a134p000007BxclAAC&trkCampaign=acq\\_paid\\_search\\_brand&sc\\_channel=PS&sc\\_campaign=acquisition\\_BEN&sc\\_publisher=Google&sc\\_category=Machine%20Learning&sc\\_country=BEN&sc\\_geo=EMEA&sc\\_outcome=acq&sc\\_detail=aws%20sagemaker&sc\\_content=Sagemaker\\_e&sc\\_matchtype=e&sc\\_segment=532492180735&sc\\_medium=ACQ-P|PS-GO|Brand|Desktop|SU|Machine%20Learning|Sagemaker|BEN|EN|Text&s\\_kwcid=AL!4422!3!532492180735!e!!g!!aws%20sagemaker&ef\\_id=CjwKCAiA9aKQBhBREiwAyGP5leWe7GBEJmNWGhV5M8lz1zXMpjC0BIUQLOapEZC9dx7\\_b71Wt0FbBxoCX9cQAvD\\_BwE:G:s&s\\_kwcid=AL!4422!3!532492180735!e!!g!!aws%20sagemaker](https://aws.amazon.com/pm/sagemaker/?trk=ps_a134p000007BxclAAC&trkCampaign=acq_paid_search_brand&sc_channel=PS&sc_campaign=acquisition_BEN&sc_publisher=Google&sc_category=Machine%20Learning&sc_country=BEN&sc_geo=EMEA&sc_outcome=acq&sc_detail=aws%20sagemaker&sc_content=Sagemaker_e&sc_matchtype=e&sc_segment=532492180735&sc_medium=ACQ-P|PS-GO|Brand|Desktop|SU|Machine%20Learning|Sagemaker|BEN|EN|Text&s_kwcid=AL!4422!3!532492180735!e!!g!!aws%20sagemaker&ef_id=CjwKCAiA9aKQBhBREiwAyGP5leWe7GBEJmNWGhV5M8lz1zXMpjC0BIUQLOapEZC9dx7_b71Wt0FbBxoCX9cQAvD_BwE:G:s&s_kwcid=AL!4422!3!532492180735!e!!g!!aws%20sagemaker)
- *Serverless Computing - AWS Lambda - Amazon Web Services*. (z.d.). Amazon Web Services, Inc. Geraadpleegd op 13 februari 2022, van <https://aws.amazon.com/lambda/>
- *Amazon API Gateway / API Management / Amazon Web Services*. (z.d.). Amazon Web Services, Inc. Geraadpleegd op 13 februari 2022, van <https://aws.amazon.com/api-gateway/>

