University College Dublin
Ireland's Global University

Amazon Web Services
Amazon SageMaker - An Exploratory Approach

Nguyen, Hang, BA. and Lalhlimpuii, Rosy, BE.

A Capstone submitted to University College Dublin in part fulfilment of the requirements of the degree of M.Sc. in Business Analytics

Michael Smurfit Graduate School of Business, University College Dublin

September 2020

Supervisors: Dr. Michael MacDonnell

Head of School: Professor Anthony Brabazon

Dedication

This work is dedicated to our family, teachers and friends without whom we would not be here and also to help and support us throughout this challenging and difficult year.

Contents

List of Figures ix

List of Tables x

Preface xi

Acknowledgements xii

Executive Summary xiii

Abstract xiv

List of Important Abbreviations xv

1 Introduction 1

1.1 Amazon Web Services 1

1.1.1 Overview 1

1.1.2 Global Infrastructure 2

1.1.3 Access AWS Services 2

1.1.4 Recognition 2

1.1.5 Why AWS is dominating the Marketplace? 5

1.2 AWS Academy 7

1.2.1 Overview 7

1.2.2 Difference between AWS Academy and AWS Educate 7

2 Literature Reviews 7

2.1 Characteristics of Cloud Computing 7

2.2 Delivery Models for Cloud Computing 11

2.3 Service Deployment Models 12

2.4 Benefits and Drawbacks of Cloud Computing 12

3 AWS Academy 13

3.1 Curriculum 14

3.2 Accounts, Registration and Canvas Classroom 15

3.2.1 Accounts and Registration 15

3.2.2 Canvas Classroom 15

4 AWS Services and Amazon SageMaker 19

4.1 Categories of Services 19

4.2 Amazon SageMaker 22

4.2.1 Machine Learning workflow with Amazon SageMaker 22

4.2.2 Benefits and features of Amazon SageMaker 25

4.2.3 Amazon SageMaker Vs Other Competitors 26

4.2.4 Getting started with Amazon SageMaker 27

5 Implementation 28

5.1 Customer Churn Prediction using Amazon SageMaker Autopilot . . . 29

5.1.1 Overview of Amazon SageMaker Autopilot, Amazon SageMaker Python SDK library and AWS built-in algorithms for SageMaker Autopilot 29

5.1.2 Overview of problem and dataset 31

5.1.3 Approach 1: Amazon SageMaker Studio Experiment 31

5.1.4 Approach 2: Notebook Instance 35

5.1.5 Comparison between Autopilot on Studio Experiment and Autopilot on Notebook Instance 36

5.2 Stock Price Prediction using GluonTS library and DeepAR algorithm 37

5.2.1 Overview of GluonTS library and DeepAR algorithm 37

5.2.2 Overview of problem and dataset 38

5.2.3 Approach 1: Amazon SageMaker Console 39

5.2.4 Approach 2: Notebook Instance 41

5.2.5 Comparison between AWS Console and Notebook Instance approaches 46

6 Limitations and Recommendations 47

6.1 Limitations of SageMaker 47

6.2 Limitations of our use cases 48

6.3 Recommendations for SageMaker practitioners 48

7 Intergration **48**

7.1 AWS Products - Module Integration 48

7.2 Learning Outcomes 49

8 Adoption Practice **51**

8.1 Cloud Computing Adoption Framework 51

8.2 Cloud Computing - Security Risks Assessment 53

9 Conclusion **55**

A Appendix A: Capstone Project Plan and Timeline **56**

B Appendix B: 1st Use Case: SageMaker Autopilot - Customer Churn Prediction **57**

C Appendix C: 2nd Use Case: SageMaker SDK - Stock Price Prediction **70**

List of Figures

1 Magic Quadrant for Cloud AI Developer Services, adopted from Baker et al., 2020 . . . 3
2 Magic Quadrant for Operational Database Management Systems, adopted from Adrian et al., 2019 . . . 4
3 Magic Quadrant for Cloud Infrastructure as a Service, Worldwide, adopted from Bala et al., 2019 . . . 5
4 Major Public Cloud Usage, adopted from Weins, 2020 . . . 6
5 Available AWS Certifications, adopted from Pearson., 2020 . . . 13
6 AWS Academy Course material, adopted from Academy, 2020 . . . 16
7 AWS Academy Course Announcements, adopted from Academy, 2020 16
8 AWS Academy Course Discussions, adopted from Academy, 2020 . . 17
9 Creating new event in AWS Academy Calendar, adopted from Academy, 2020 . . . 17
10 Sending an email in AWS Academy Inbox, adopted from Academy, 2020 . . . 18
11 On-demand lecture videos with English subtitles, adopted from Academy, 2020 . . . 18
12 Student guides (Module Summary), adopted from Academy, 2020 . . 19
13 Knowledge check (Exercises) after each module, adopted from Academy, 2020 . . . 19
14 Workflow for creating a Machine Learning model, adopted from AWS., 2020c . . . 23
15 Training and Deploying a model with Amazon SageMaker, adopted from AWS., 2020b . . . 25
16 Available Machine learning services provided by Amazon SageMaker, adopted from AWS., 2020a . . . 25
17 Description of 2 use cases . . . 29
18 A high-level view of component jobs that Autopilot executes while processing a customer request, adopted from Piali Das, 2020 . . . 30
19 Automated Processes in Realtime . . . 33
20 Autopilot automatically chooses the best model after Training jobs . . 34

21 Trial Component List 34
22 Stock Price Prediction - Approach 1: Training Logs 40
23 Stock Price Prediction - Approach 1: Suggested Machine learning workflow for production, adopted from Correa, 2019 41
24 Clustering Result 44
25 Stock Price Prediction - Approach 2: Test metrics result 45
26 Stock Price Prediction - Approach 2: Visualisation of predicted and actual data 46
27 Cloud Adoption Framework, adopted from Paredes-Gualtor et al., 2017 51
28 AWS Academy Capstone Project Plan and Timeline 56
29 Approach 1: Customer Churn Prediction - Data Exploration notebook - Dataset Sample 58
30 Approach 1: Customer Churn Prediction - Data Exploration notebook - Column Analysis 58
31 Approach 1: Customer Churn Prediction - Data Exploration notebook - Count Statistics 59
32 Approach 1: Customer Churn Prediction - Data Exploration notebook - Descriptive Statistics 60
33 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot Job and Amazon S3 Configuration 60
34 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Configuration Results 61
35 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate List 61
36 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 1 61
37 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 2 62
38 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 3 62
39 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 4 63

40 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 5 63
41 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 6 64
42 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 7 64
43 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 8 65
44 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 9 65
45 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 10 66
46 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot recommendation of hyperparameters, objectives and accuracy metrics for the algorithm and problem type 66
47 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot recommendation of tunable hyperparameters search ranges 66
48 Approach 1: Customer Churn Prediction - Candidate Generation notebook - A multi-algorithm model tuning job across the different possible algorithms and feature processing strategies. 67
49 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Hyperparameter tuning result 67
50 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Build the best model, by linking the best training job with feature engineering pipelines 68
51 Approach 1: Customer Churn Prediction - Candidate Generation notebook - Deploy the best model, create an endpoint for production . . . 68
52 Create an endpoint for production/testing 69
53 Approach 2: Customer Churn Prediction - Autopilot Jobs Setup Logs 69
54 Approach 2: Customer Churn Prediction - The best model. ObjectiveMetric = validation:accuracy 69

55 Stock Price Prediction - Approach 1: Create a new container repository from ECR 70
56 Stock Price Prediction - Approach 1: Create new processing job . . . 71
57 Stock Price Prediction - Approach 1: Provide local and S3 paths to input data and processing artifact 72
58 Stock Price Prediction - Approach 1: Choose an AWS built-in algorithm 73
59 Stock Price Prediction - Approach 1: Choose the appropriate hyperparameters 74
60 Stock Price Prediction - Approach 1: Create two channels for training set and validation set 75
61 Stock Price Prediction - Approach 1: Create folder for checkpoint and output data in s3 bucket 76
62 Stock Price Prediction - Approach 1: Create a new model with the completed training job 76
63 Stock Price Prediction - Approach 2: Custom life-cycle for new Notebook Instance 77
64 Stock Price Prediction - Approach 2: Create a new Notebook Instance 78
65 Stock Price Prediction - Approach 2: Create new IAM role 79
66 Stock Price Prediction - Approach 2: Custom policy for IAM role . . 79
67 Stock Price Prediction - Approach 2: Unpreprocessed Data 80
68 Stock Price Prediction - Approach 2: Data Exploration before Feature Engineering 80
69 Stock Price Prediction - Approach 2: Transformed Data for DeepAR algorithm 81
70 Stock Price Prediction - Approach 2: Training loss per epoch 81

List of Tables

1 ML Solution Basic Pricing 27

2 Features Descriptions of SageMaker Console 39

3 Business Analytics Module Integration 49

4 Features Descriptions of the Customer Churn Dataset 57

5 Features Descriptions of the Deutsche Börse Public XETRA Dataset . 70

Preface

This Capstone thesis is written and completed as a part of M.Sc. in Business Analytics program, Michael Smurfit Graduate Business School, University College Dublin under the supervision of Dr Michael MacDonnell, in collaboration with Amazon Web Services. The main objective of this project is to come up with an extensive comparison among Amazon SageMaker tools: SageMaker Studio, SageMaker Notebook Instance and SageMaker Console. Furthermore, we had the opportunity to delve deeply into the latest feature of Amazon SageMaker - Autopilot and the AWS built-in algorithm DeepAR. Finally, the thesis covers cloud computing concepts and different Amazon Web Services products extensively.

Dublin Hang Nguyen

September 2020 Rosy Lalhlimpuii

Acknowledgements

We want to thank and express our deepest appreciation to our Supervisor, Professor Dr Michael MacDonnell, for his guidance and continuous support to accomplish this capstone. We would also like to sincerely thank all the Professors who were part of the course. The concept learnt in the course greatly contributed to complete the project. Last but not least, we are grateful to our classmates and group members who gave feedback and constant support in making this project complete.

Executive Summary

Cloud computing is a model used for delivering on-demand computing services over the internet on a pay-for-use basis. It enables convenient access to a shared pool of configurable computing resources such as networks, servers, storage, applications, and services. All these resources can be quickly provisioned and released with minimal management effort or service provider interaction. It is composed of five essential characteristics, three service models, and four deployment models. Amazon Web Services (AWS) is a comprehensive, easy to use cloud service offered by Amazon, which provides services in the form of building blocks to create and deploy any type of application in the cloud. It offers a wide range of different business purpose global cloud-based products. AWS Academy was launched by Amazon with its free, ready-to-teach cloud computing curriculum that helps students and educators to stay at the forefront of AWS Cloud innovation.

This thesis explores the AWS Academy platform and performs two use cases using the AWS Machine learning product. The first use case is based on customer churn prediction using Amazon SageMaker Autopilot and the second use case is based on Stock Price Prediction using using GluonTS library and DeepAR algorithm. This thesis also covers the suggestion of possible integration of relevant AWS resources with MSc Business Analytics coursework.

Abstract

Nowadays, Machine Learning (ML) and Artificial Intelligence (AI) platforms are one of the fastest growing services of the cloud computing providers. However, unlike other cloud-based services, ML and AI platforms are available through diverse offerings, including Cognitive Services, ML Platform as a Service - ML PaaS (e.g. automated machine learning (AutoML), ML model serving, ML model management, etc.) and ML Infrastructure Services (e.g.GPU-based computing). Among those, ML PaaS provides pre-configured and ready-to-use data science environments in which data scientists and engineers could process, train, tune, and deploy the model in a single platform. As a result, ML PaaS tackles the complexity caused by running the training jobs on a cluster of computers. Some examples of ML PaaS in the cloud are Amazon SageMaker, Microsoft Azure ML Services, Google Cloud ML Engine, IBM Watson Studio (Janakiram, 2019). Within the scope of this thesis, we will explore Amazon SageMaker by comparing among 3 approaches (Studio, Notebook Instance, Console) in 2 use cases (Autopilot, Built-in algorithm). Besides, we will discuss how AWS Academy could help students gain knowledge about AWS Services and recommend an Integration of AWS Products - Modules in MSc in Business Analytics.

List of Important Abbreviations

Abbreviation	Full Form
API	Application Program Interface
A.S.E	Amazon SageMaker Examples
A.S.S.S.C.U.G	Amazon Simple Storage Service Console User Guide
A.S.T	Amazon SageMaker Tutorial
AWS	Amazon Web Services
CRM	Customer Relationship Management
DBG	Deutsche Börse
EC2	Elastic Compute Cloud
ECR	Elastic Container Registry
EFT	Exchange Traded Funds
ERP	Enterprise Resource Planning
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
IAM	Identity and Access Management
IoT	Internet of Things
IT	Information Technology
MOOC	Massive Open Online Course
NIST	National Institute of Standards and Technology
OPDBMS	Operational Database Management Systems
OS	Operating System
QoS	Quality of service
S3	Simple Storage Service
SDK	Software Development Kits
VM	Virtual Machine
VPC	Virtual Private Cloud
VUE	Virtual University Enterprises

1 Introduction

The main contributions of this thesis are: (i) to provide an experiment of Amazon SageMaker Autopilot on both SageMaker Studio and SageMaker Notebook Instance; (ii) to apply an Amazon built-in algorithm on both SageMaker Notebook Instance and SageMaker Console; and (iii) to recommend an Integration of AWS Products - Modules in MSc in Business Analytics.

Section 1 presents an introduction on AWS, AWS SageMaker and AWS Academy.

Section 2 presents an in-depth literature review on Cloud Computing and Cloud-based Machine Learning services.

Section 3 introduces the AWS Academy and how to use this learning platform.

Section 4 briefly discusses AWS current services and Amazon SageMaker.

Section 5 shows 2 use cases of Amazon SageMaker Autopilot and Amazon built-in algorithms to address 2 well-known business problems: Customer Churn Prediction and Stock Price Prediction.

Section 6 discusses the limitations of Amazon SageMaker and our use cases, and the recommendations for SageMaker practitioners.

Section 7 recommends an Integration of AWS Products - Modules in MSc in Business Analytics.

Section 8 shows the detailed steps to help organisations with successful cloud adoption.

Section 9 present the conclusion

1.1 Amazon Web Services

1.1.1 Overview

First launched in 2006, Amazon Web Services (AWS) is the comprehensive and secure cloud platform, offering over 175 fully-featured services, including data warehousing, deployment tools, directories, content delivery, etc. with pay-as-you-go pricing from data centres globally. AWS offers startups, enterprises, and government agencies with flexibility and on-demand access to compute, storage, network, database, and other IT resouces and management tools, in order to respond quickly to ever-changing business requirements.

1.1.2 Global Infrastructure

As on Jan 2020, AWS is serving over a million active customers in more than 190 countries from over 60 Availability Zones within over 20 geographic Regions around the world. An AWS Region is a isolated physical location where AWS place multiple Availability Zones, which consist of one or more discrete data centers and locate. AWS global infrastructure is steadily growing to help customers achieve lower latency and higher throughput, and to ensure that their data resides only in the AWS Region they specify (Mathew, 2020).

1.1.3 Access AWS Services

To access AWS services, customers could use one of the three following options:

- *AWS Management Console:* This is an easy-to-use graphical interface, which has both web-based application and mobile application.
- *Command Line Interface:* Customers could access AWS services from the command line and automate them through scripts.
- *Software Development Kits (SDKs):* By simply installing the package, customers could interact with AWS services directly from their API with the programming language of choice.

AWS offers a wide range of cloud services that could be used separately or in combinations tailored to the customers' needs. AWS services fall under different categories and each category contains one or more service. There are a total of twenty-five AWS categories, covering from the web and mobile applications, game development, data processing and warehousing, storage to the quantum technologies, a recently launched service.

1.1.4 Recognition

The leader in Magic Quadrant for Cloud AI Developer Services

Fig. 1. Magic Quadrant for Cloud AI Developer Services, adopted from Baker et al., 2020

This Gartner Magic Quadrant report (Baker et al., 2020) compares among cloud-hosted artificial intelligence (AI) services in three key areas: language, vision and automated machine learning (AutoML). Vendors in the shortlist are evaluated based on two criteria. The first is "Ability to Execute", which combines a broad set of services, the promising overall viability, the effectiveness of sales and marketing execution, customer satisfaction and operational efficiency. As shown in Fig.1, AWS is the vendor that has the highest score in both two criteria. Additionally, AWS, along with Microsoft Azure, Google Cloud and IBM Cloud is assessed as a Leader, which offers API-accessible services in all three key service areas and does not require developers to have data science expertise.

A leader in Magic Quadrant for Operational Database Management Systems

Fig. 2. Magic Quadrant for Operational Database Management Systems, adopted from Adrian et al., 2019

(Adrian et al., 2019) Assuming that 75% of all databases will be on a cloud platform by 2023, Data and Analytics leaders are relentlessly growing their set of OPDBMS services. The purpose of these OPDBMS services is to support enterprise-level applications, which combines both purchased business applications, such as ERP and CRM applications, and custom-made transactional systems. Although AWS is evaluated as a Leader in the field, it is still lagging behind Microsoft Azure and Oracle. According to the report, the reasons are the lack of features, such as multi-region writes and auto sharding, and on-premises deployments.

The leader in Magic Quadrant for Cloud Infrastructure as a Service, Worldwide

Fig. 3. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide, adopted from Bala et al., 2019

(Bala et al., 2019) Gartner placed AWS as a Leader of global Cloud Infrastructure as a Service. In the report, they indicated that AWS is the most experienced, enterprise-ready provider, with the most robust track record of customer success and the most valuable partner ecosystem. As shown in Fig.3, AWS is the provider that has the highest score in both two criteria.

1.1.5 Why AWS is dominating the Marketplace?

AWS remains a leader in cloud computing and Infrastructure-As-A-Service (IaaS) mainly due to its combination of technological advances, unique service offerings, and years of cloud computing excellence as mentioned by Underwood (2019). There are other major tech names including Microsoft, IBM, and Google all have their own cloud offerings, but yet somehow AWS manages to keep its lead. Below are the three reasons why AWS is dominating the cloud computing marketplace:

- *It Has Been Around the Longest:* It was introduced to the world by Amazon in 2006. It has had a decade to refine and perfect its product as well as build a solid customer base, apart from one of the first major cloud computing platforms widely available. When compare it with Microsoft Azure, Microsoft's cloud

platform was introduced in 2010, four years after AWS and Google App Engine (a predecessor to Google Cloud) was released in 2008, two years after AWS.

- *It is available to test it out for free:* Before committing to a full cloud migration, the users get hands-on experience with the AWS cloud platform with AWS Free Tier and at no cost to them. Users have access to AWS Free Tier services such as Amazon EC2 (cloud computing capacity), Amazon S3 (cloud storage), and Amazon RDS (Relational Database Service) for 1 year. It is very easy to move a subscription from the AWS Free Tier to the full-service AWS and continue with business as usual.

- *Much bigger than its competitors:* According to Forbes discussed a Gartner Report (Kepes, 2015), it estimated that AWS has more than ten times the computing capacity of the next 14 largest infrastructure vendors combined. There are multiple data centers located across the globe and it now spans 77 Availability Zones within 24 geographic regions around the world. It has also announced plans for nine more Availability Zones and three more AWS Regions in Indonesia, Japan, and Spain.

The below chart from Flexera 2020 State of the Cloud Report Fig.4 shows the major public cloud provider use shifting among enterprises. It shows that AWS has the highest current usage of 76% followed by Azure with 69%. Azure has nearly closed the gap with AWS in the breadth of adoption, while Google Cloud comes third followed by Oracle which has moved into the number four spot.

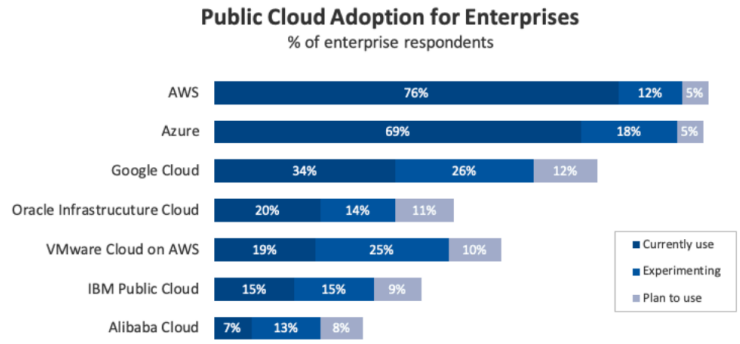

Fig. 4. Major Public Cloud Usage, adopted from Weins, 2020

1.2 AWS Academy

1.2.1 Overview

According to (Adrian et al., 2019), 75% of all databases will be on a cloud platform by 2023. Clearly, cloud computing would likely to be an essential hard skill in the future and employees with cloud computing skills are in high demand.
Noticing this opportunity, AWS launched AWS Academy, a program that offers higher education institutions a cloud computing curriculum in order to prepare students to pursue industry-recognized certifications, keep up with the cutting edge of AWS Cloud innovation and needed skills to get hired in the fastest-growing industries. Some of the most prestigious institutions applied AWS Academy into their curriculum are Harvard University, The University of Auckland, Singapore Management University, ESADE, London School of Economics and Political Science.

1.2.2 Difference between AWS Academy and AWS Educate

AWS Educate is a membership-based program which offers curated learning content, collaboration tools, and access to selected AWS Training and the Job Board. Whereas, AWS Academy is a higher education program designed to be delivered over a semester by AWS Academy Accredited Instructors. This program consists of end-to-end curriculums, instead of selected training materials as AWS Educate.

2 Literature Reviews

2.1 Characteristics of Cloud Computing

Over the past few years, the evolution of cloud computing is potentially one of the significant advances in the history of computing (Marston et al., 2011). Cloud computing is now a global trend, and it is not only the business world that concentrates on cloud computing but also the academic world pays great attention to this hot topic during the past decade (Bayramusta & Nasir, 2016). Cloud computing due to its abilities to offer flexible, dynamic IT infrastructures, QoS guaranteed computing environments and configurable software services is currently emerging as a hot topic (Wang et al., 2010). Although the Cloud computing practice has attracted much attention, there are still no

widely accepted definitions for Cloud computing. There are several reasons that lead to this situation. One reason is that Cloud computing involves researchers and engineers who work on it from different viewpoints from various backgrounds, e.g., Grid computing, software engineering and database. The other reason is that technologies are still evolving and progressing, which enable Cloud computing, for example, Web 2.0 and Service-Oriented Computing. Another vital reason is that large scale deployment and usage are still lacked by existing computing Clouds, which would finally justify the concept of Cloud computing. According to the AWS website, cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. The technology services such as computing power, storage, and databases can be accessed on an as-needed basis instead of buying, owning, and maintaining physical data centres and servers.

It may seem to be that cloud computing was introduced over the past decade as it started to develop into the giant, omnipresent and all-powerful behemoth we know today. But the truth is that concept on cloud started way back in the 1950s when mainframe computers came into existence. Back in those days, the central computer was accessed by several users via dummy terminals. The mainframe computers were huge machines, and too expensive to buy and maintain for every single employee, and the device did not make them economically feasible for organisations to buy them (Namboori, 2014). It was then the idea of provision of shared access to a single computer occurred to the companies to save costs. According to Namboori, many further advancements in cloud technology came into being over the decades that followed. In the 1970s, IBM released an operating system (OS) called the Virtual Machine (VM). Simultaneous operation of more than one OS was made possible by the virtual machine. The guest operating systems with their own memory and other infrastructure could be run on every virtual machine, making it possible to share these resources.

Amazon Web Services (AWS) emerged in the early 2000s and in 2006 Amazon launched Elastic Compute Cloud (EC2) allowing companies and individuals to rent virtual computers through which they could use their own programs and applications (Foote, 2017). Google launched its Google Docs services in that same year, allowing users to save, edit and transfer documents in the cloud. In 2017, IBM and Google, along with several universities, joined forces to develop a server farm for research projects.

As mentioned by Foote, during the same year, with the help of cloud Netflix launched its video streaming service to stream movies and other video content into the homes and onto the computers of millions of subscribers worldwide.

In cloud computing, when large-scale computing infrastructures are adopted, Machine learning as-a-Service can be considered as one of the most demanded services (Garcıa et al., 2020). The machine learning pipelines for large-scale learning tasks can be exploited to provide an additional level of challenge with the increasing availability of the amount of data. There are few cloud-based initiatives and solutions which aimed to lower the entry barrier for users that want to leverage the usage of machine learning models. When it comes to machine learning as a service, Google, Microsoft, and Amazon have been releasing their own proprietary platforms (Ribeiro et al., 2015). In 2014, Google released its prediction API, and in the same year, Microsoft launched Azure Machine Learning. Then AWS Machine Learning was launched by Amazon in 2015. Flexible alternatives for enterprises of all sizes to build working models are offered by machine learning cloud services. Their increasing sales shows that the demand exists, and many small and medium businesses are adopting machine learning as a service at a faster rate.

Currently, a notable competitor among automated solutions for ML is the Amazon machine learning for predictive analytics (Jain, 2020). Amazon ML is ideal for operations limited to particular deadlines. The facility of loading data from multiple sources with automatic and flexible data preprocessing operations improve the credibility of Amazon ML. A reliable alternative for data scientists known as the AWS SageMaker was launched in 2017 to get a machine learning environment with tools for faster model creation and deployment. It includes many pre-built algorithms optimised for massive datasets and computing in large, distributed systems.

The three core technologies that will enable the evolution of cloud computing to unfold fully are virtualisation, multitenancy and web services and currently are taking rapid shape (Marston et al., 2011).

Virtualisation is the technology that presents an abstract, emulated computing platform and hides the physical characteristics of a computing platform from the users (A Vouk, 2008). Unlike a physical system, the emulated computing platform for all prac-

tical purposes behaves like an independent system and can be configured on-demand, and maintained and replicated very easily. There are lower upfront and operational costs due to better utilisation of the computing infrastructure. Within an emulated system, computing power and networking resources have caught up to deliver the level of seamless performance that users have grown accustomed to on personal computers. Multitenancy is a concept by which a single instance of an application software serves multiple clients. System resources are better utilised with the help of it in terms of memory and processing overhead. According to W3C (2004), a Web Service is defined as "a software system designed to support interoperable machine-to-machine interaction over a network". The term commonly refers to clients and servers that communicate over the HTTP protocol used on the Web. Web services make it easier for a software client like a web browser to access server applications over a network by standardising the interfaces between applications.

In Cloud computing, distributed computing can provide more effectiveness for sharing resources and collaborations in a group, making it user-centric and task-centric (El-Gazzar, 2014). Cloud computing service models share five common essential characteristics that distinguish cloud computing from other computing technologies (Mell, Grance, et al., 2011).

- *On-demand self-service*: It is where the consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service provider.

- *Broad network access*: It is where the capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).

- *Resource pooling*: It is where the provider's computing resources are pooled to serve multiple consumers using a multitenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.

- *Rapid elasticity*: It is where capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward com-

mensurate with demand.

- *Measured service*: It is where cloud systems automatically control and optimise resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts).

2.2 Delivery Models for Cloud Computing

Under the cloud computing umbrella, there are a number of service offerings and implementation models. The three delivery models defined by the NIST definition of cloud computing are as follows:

- *Software-as-a-Service (SaaS):* A finished application is offered by SaaS, which the end-users can access through a thin client. The finished applications are run and managed by the service provider, so the end-user does not have any control over the application design, server networking and storage infrastructure which makes the users free from thinking about the services and underlying infrastructure maintenance. Notable examples of SaaS include Gmail, Google Docs, and Salesforce.com.

- *Platform-as-a-Service (PaaS):* An operating system, as well as suites of programming languages and software development tools, are offered by PaaS. It allows the customers to develop their own applications without the need to manage underlying infrastructure resulting from being more efficient. Well-known examples include Microsoft Windows Azure and Google App Engine.

- *Infrastructure-as-a-Service (IaaS):* IaaS contains the basic building blocks for cloud IT. It offers end users direct access to processing, networking features, storage and other computing resources. IaaS provides the highest level of flexibility and management control by allowing the users to configure those resources and by running the operating systems and software on them accordingly. Examples of IaaS include Amazon Elastic Compute Cloud (EC2), Rackspace, and IBM Computing on Demand.

2.3 Service Deployment Models

The four deployment models defined by the National Institute of standards and technology (NIST) are as follows:

- *Private Cloud:* This cloud infrastructure is exclusively used by a single organisation with multiple business units. It may be managed, operated and owned by the organisation or a third party or some combination of them.

- *Community Cloud:* This cloud infrastructure is shared by several organisations. It supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organisations in the community, a third party, or some combination of them.

- *Public Cloud:* This cloud infrastructure is available to the general public or a large industry group, and it may be owned, managed, and operated by a business, academic, or government organisation, or some combination of them.

- *Hybrid cloud:* It is the combination of two or more clouds (public, private or community). It remains a unique entity but is bound together.

2.4 Benefits and Drawbacks of Cloud Computing

The benefits of cloud computing mainly include low-cost, availability on a network, innovation power, high expandability, friendly utilisation, and environmental protection (Linthicum, 2009). Cloud computing is a cost-effective solution especially for data-intensive applications (Deelman et al., 2008). The virtual machine, which is one of the basic characteristics of cloud computing, benefits both clouds providers and users. The virtual machine (VM) makes an illusion of running directly on the physical machine and can be beneficial for isolation and resource sharing and multiple favourite operating systems (Sugerman et al., 2001).

Along with the benefits, there are several risks caused by using cloud computing. The major potential risks include security problems, low controllability, financial burden, weak openness, lack of auditing feature, and service agreement issues (Linthicum,

2009). The two conventional security problems involve poorly implemented protocols and authentication process (Mansfield-Devine, 2008). Devine also mentioned that most security issues can only reply to vendors or consumers experiences or some other conventional security concepts as cloud computing do not have specific security standards yet. Challenges like changes of software/hardware architecture, the obstacle of data transfer, and interoperability concern might also be faced by enterprises while immigrating to cloud computing (Dikaiakos et al., 2009).

3 AWS Academy

AWS Academy offers university student with a free and on-demand cloud computing courses that prepares them to pursue certifications and high-demand cloud jobs. As on Aug 2020, the five courses within the AWS Academy platform assists student to achieve the Foundation, Associate level and Machine Learning, Data Analytics Specialty AWS Certifications. As on Pearson VUE website, the authorised test centre for AWS certification exams, there is a total of 12 available AWS Certifications which are shown in Fig.5 as follows.

Fig. 5. Available AWS Certifications, adopted from Pearson., 2020

3.1 Curriculum

As of Aug 2020, AWS Academy offers five self-paced courses, including two introductory course and three intermediate-level courses, along with lab exercises. The content of each course and lab exercises is described as follows:

- *AWS Academy Cloud Foundations:* This introductory course is intended for students who seek a fundamental understanding of AWS cloud computing technology and/or pursue an AWS Certified Cloud Practitioner certification. This course provides a detailed overview of cloud concepts, pricing, AWS global infrastructure and cloud security, cloud services, architecture, scaling, and monitoring.

- *AWS Academy Cloud Architecting:* This intermediate-level course is intended for students who seek the fundamentals of building a robust IT infrastructure on AWS and/or pursue the AWS Certified Solutions Architect – Associate certification. In this course, students will learn how to optimise the AWS Cloud and how these services fit into cloud-based solutions.

- *AWS Academy Cloud Developing:* This intermediate - level course will help students obtain technical expertise in development using cloud technologies, and prepare them with the AWS Certified Developer − Associate certification exam. AWS Cloud Developing teaches students how to configure AWS Identity and Access Management for programmatic access, configure storage with Amazon S3 programmatically, develop with DynamoDB, etc.

- *AWS Academy Cloud Operations:* This intermediate - level course is designed to prepare students not only to pursue entry-level DevOps, support, and cloud operations roles but also to take the AWS SysOps Administrator – Associate exam. During this course, students will learn how to troubleshoot various scenarios and automate deployments of networks and systems on AWS.

- *AWS Academy Machine Learning Foundations:* This introductory course teaches the core concepts of artificial intelligence and machine learning, particularly how to label, build, train, and deploy a custom machine learning model, and how to select and apply machine learning services to solve business problems.

- *AWS Academy Data Analytics:* These lab exercises are intended for students interested in pursuing careers in data analysis. Through these exercises, students will learn how to conduct Big Data analysis with practical, real-world examples and to create visual representations from analysed data. These labs and learning resources support students as a supplement to an institution's existing Big Data and data analytics courses.

3.2 Accounts, Registration and Canvas Classroom

3.2.1 Accounts and Registration

There would be no charge for these courses. To join one of the courses, students must be given the authorisation by the institution they are attending. Once registered for a course, students would receive a 50% discount voucher on this AWS Certification exams and have unlimited access to the online curriculum, including lecture videos and hands-on lab exercises, within a limited time. Upon completion of this course, students would receive a Certification of Completion and continue to be able to view the course materials in the next 2 months.

To set up accounts, your institution needs to provide AWS Academy the students' email. Around one to two days before your course's starting day, AWS Academy would send the 'Course Invitation' email to students' registered email with the course information and a register link to Canvas classroom.

3.2.2 Canvas Classroom

Dashboard

Upon the registration, the students could log in AWS Academy via the link https://awsacademy.instructure.com/ and access the course content in the Courses section, Module tab as shown in Fig.6 below.

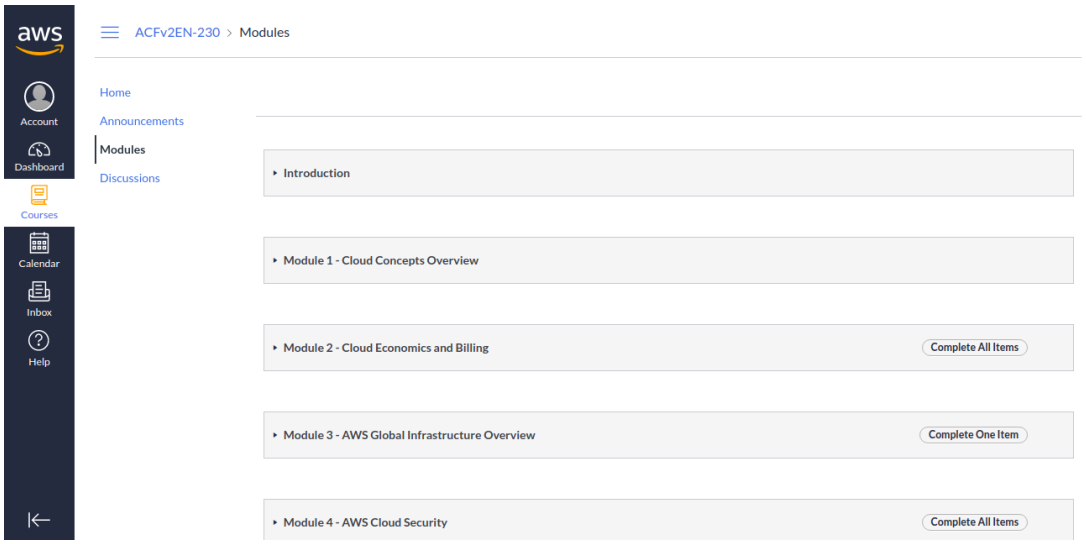

Fig. 6. AWS Academy Course material, adopted from Academy, 2020

Under the Courses section, student could access to the Announcements tab (Fig.7) and participate in Class Discussions in Discussions tab (Fig.8) as shown below.

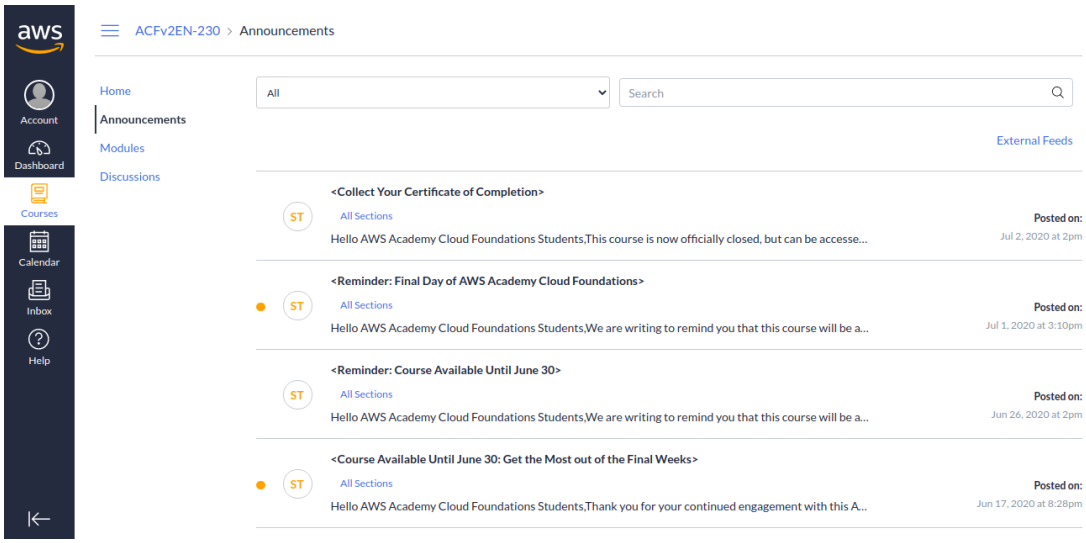

Fig. 7. AWS Academy Course Announcements, adopted from Academy, 2020

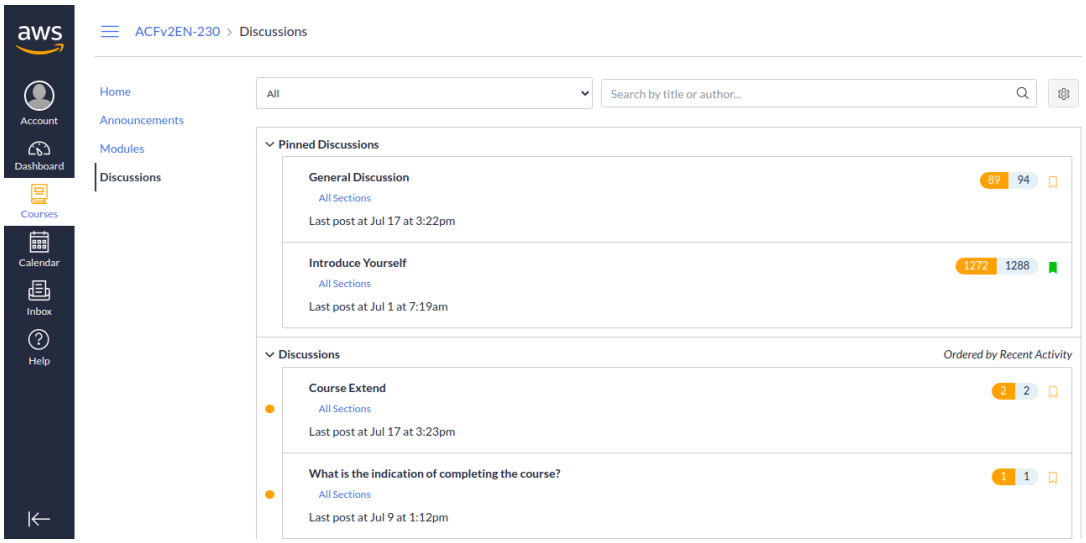

Fig. 8. AWS Academy Course Discussions, adopted from Academy, 2020

To support the study, students could create new event and task in the Calendar section (Fig.9) and send email to instructors and other student in the Inbox section (Fig.10).

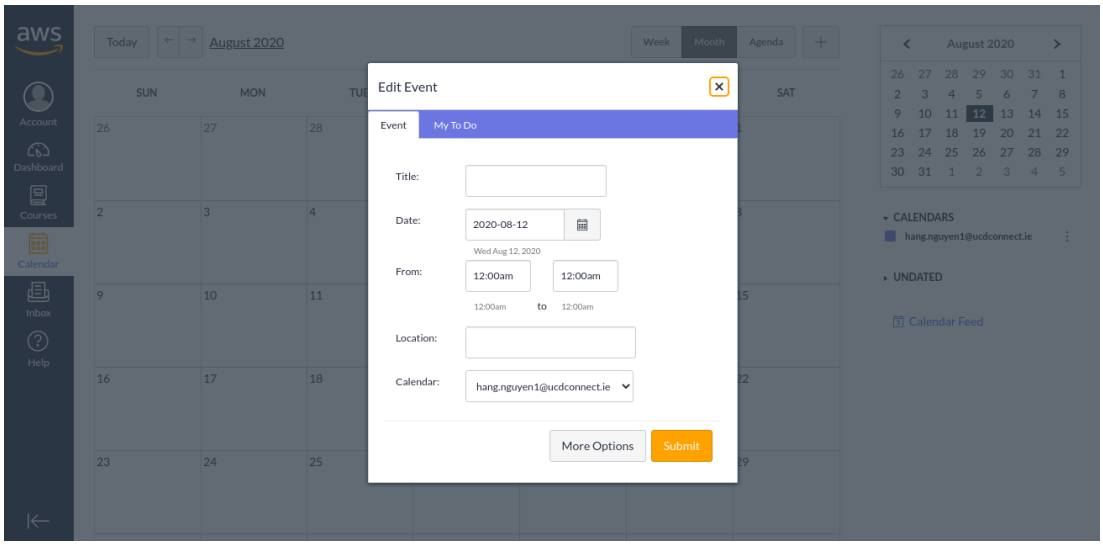

Fig. 9. Creating new event in AWS Academy Calendar, adopted from Academy, 2020

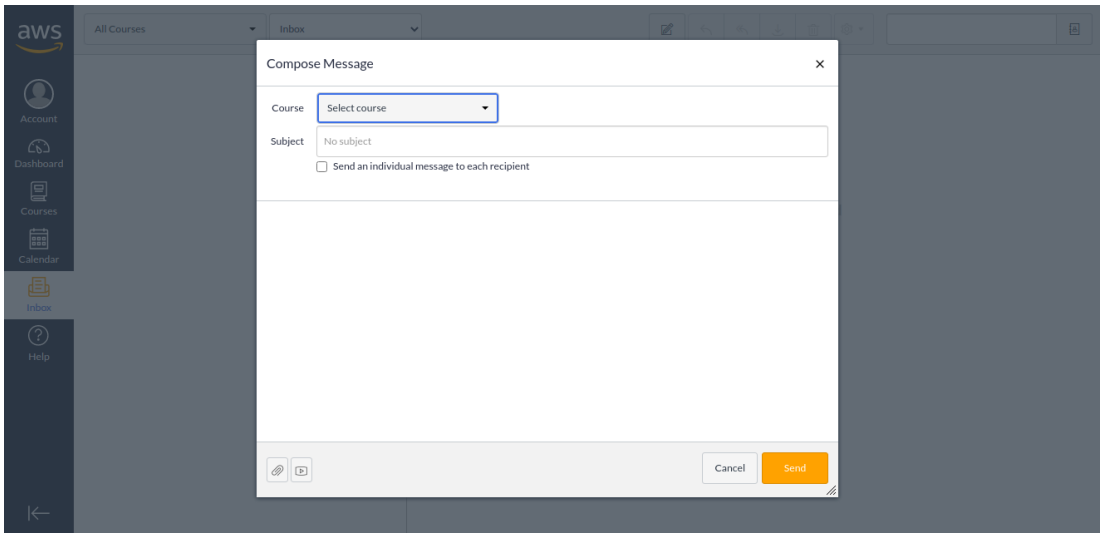

Fig. 10. Sending an email in AWS Academy Inbox, adopted from Academy, 2020

Learning Materials

Like many other MOOC (Massive Open Online Course) programs, the course content is delivered in the form of on-demand lecture videos with English subtitles (Fig.11), module summaries (Fig.12) and MCQ knowledge-check exercises (Fig.13). However, compared with Coursera, one of the most popular MOOC platform, the Canvas platform lacks some of the useful features, such as note taking, speed adjustment, goal setting, etc. (Goli, 2020).

Fig. 11. On-demand lecture videos with English subtitles, adopted from Academy, 2020

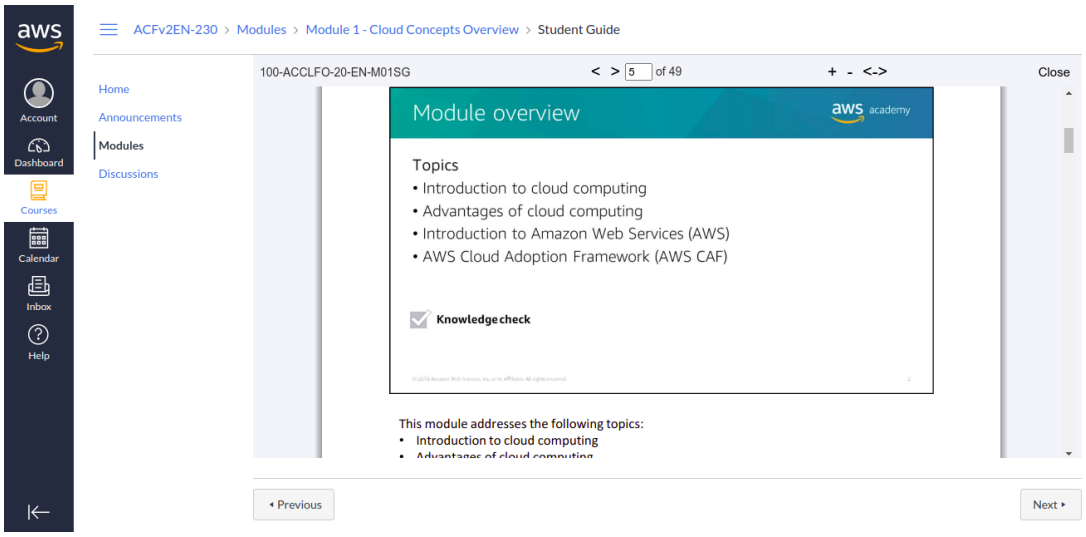

Fig. 12. Student guides (Module Summary), adopted from Academy, 2020

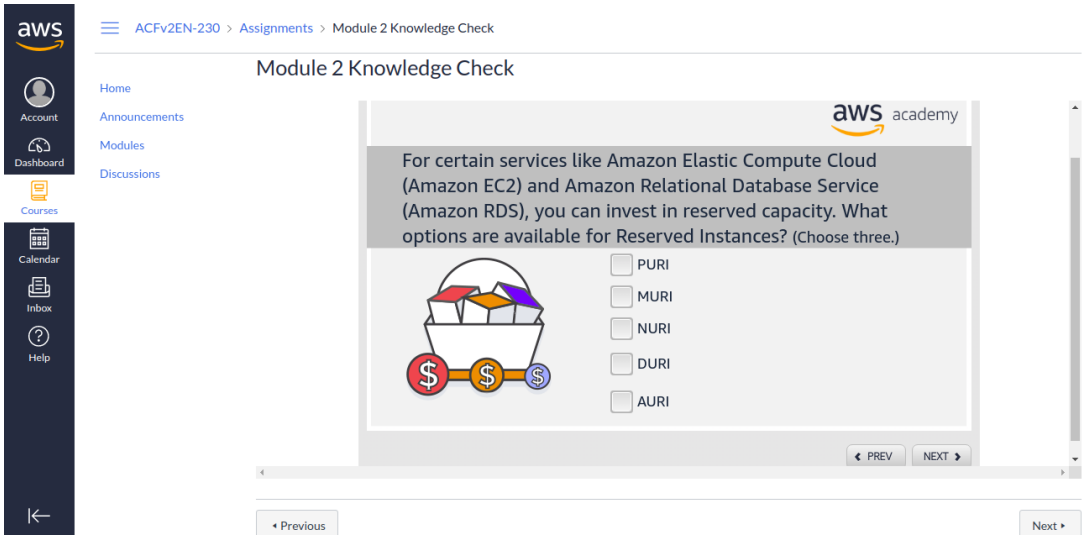

Fig. 13. Knowledge check (Exercises) after each module, adopted from Academy, 2020

4 AWS Services and Amazon SageMaker

4.1 Categories of Services

A broad set of cloud-based services are offered by AWS. There are 23 different product or service categories, and each category consists of one or more services. The most widely used services and which offers the best introduction to the AWS Cloud includes compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security, enterprise applications and machine learning. The benefits

of these services are that it helps organisations move faster, lower IT costs, and scale. Many large enterprises and start-ups trust AWS to power a wide variety of workloads.

- *Compute Services:* Many compute services are offered by AWS. One of the most popular services is the Amazon EC2 which provides secure, resizable compute capacity in the cloud, and the making of web-scale cloud computing gets much easier with the help of it for developers. AWS Lambda is a zero-administration compute platform, and it enables us to run code without provisioning or managing servers. It supports cloud-native architectures, which enable massive scalability at a lower cost. Other container-based services include Amazon Elastic Container Service, Amazon Elastic Kubernetes Service, AWS Fargate, and Amazon Elastic Container Registry. All these enables us to run multiple workloads on a single operating system (OS).

- *Storage Services:* Compare to the traditional on-premises storage systems cloud storage is typically more reliable, scalable, and secure. As cloud storage holds the information that the applications use, it acts as a critical component of cloud computing. Systems like the Internet of Things (IoT), big data analytics, data warehouses, databases, backup and archive applications all depend on data storage architecture. A complete range of services to store, access, govern, and analyse data are offered by AWS to reduce costs, increase agility, and accelerate innovation. Some of the common ones are Amazon Elastic Block Store (Amazon EBS), Amazon Simple Storage Service (Amazon S3), Amazon Elastic File System (Amazon EFS) and Amazon Simple Storage Service Glacier.

- *Database Services:* AWS databases are built for business-critical, enterprise workloads, offering high availability, reliability, and security. They are chosen from 15 purpose-built database engines including relational, key-value, document, in-memory, graph, time series, and ledger databases. It supports diverse data models and allows to build use case driven, highly scalable, distributed applications. The AWS database is fully managed, and it continuously monitors to keep the workloads up and running with self-healing storage and automated scaling. The database's computing and storage resources can be scaled easily, often with no downtime. Some of the common ones are Amazon Relational

Database Service (or Amazon RDS), Amazon DynamoDB, Amazon Redshift, and Amazon Aurora.

- *Security, Identity, and Compliance Services:* AWS provides services helps to protect data, accounts, and workloads from unauthorised access. Encryption and key management and threat detection are provided by AWS data protection services that continuously monitors and protects accounts and workloads. The identity services in AWS helps workforce and customer-facing applications to get started quickly and manage access to workloads and applications. A comprehensive view of the compliance status is given by AWS, and the environment is continuously monitored using automated compliance checks based on the AWS best practices. Some of the common ones are AWS Identity & Access Management (IAM), AWS Cognito, AWS Shield, AWS Artifact and AWS Key Management Service (KMS).

- *Networking and Content Delivery Services:* On the AWS global network, essential security features are offered by the set of AWS networking services by isolating resources, encrypting data, and connecting privately. The best support for the broadest set of applications is delivered by AWS global network, even for those with the highest throughput and lowest latency requirements. AWS provides the highest network availability with fewer downtime hours from networking issues than the next largest cloud provider. The AWS global network has broadest global coverage that delivers customer applications and content anywhere in the world over a private network. Some of the common ones are Amazon VPC, Amazon Route 53, Amazon CloudFront and Elastic Cloud Balancing.

- *Management and Governance Services:* For both business agility and governance control, customers can enable, provision, and operate their environment with AWS. Highly dynamic cloud resources at massive scale are managed by the AWS Management and Governance services. To manage and govern resources in AWS, on-premises, and other clouds, AWS offers a single control plane for customers, which helps in reducing complexity. With the help of this service, customers can also assess their resource utilisation and identify ways to reduce

costs. Some of the common ones are AWS Trusted Advisor, AWS CloudWatch, AWS CloudTrail, AWS Management Console, AWS Command Line Interface, etc.

- *Machine Learning Services:* The broadest and deepest set of machine learning services and supporting cloud infrastructure is offered by AWS. It also imposes machine learning in the hands of every developer, data scientist and expert practitioner. AWS helps accelerate tens of thousands of customers in their machine learning journey. Some of the common ones are Amazon SageMaker, Amazon Augmented AI, Amazon CodeGuru, Amazon Forecast, Amazon Fraud Detector, Amazon Kendra, Amazon Lex, etc.

4.2 Amazon SageMaker

As mentioned in the AWS website, Amazon SageMaker is the Machine Learning platform on AWS that provides infrastructure to build, train, and deploy machine learning (ML) models quickly. Other storage and analytics services are integrated with AWS Amazon SageMaker to make the essential data management tasks for a successful Machine Learning project secure, scalable and streamlined. The heavy lifting from each step of the machine learning process are removed by SageMaker to make it easier to develop high quality models.
In traditional machine learning development, there are no integrated tools for the entire machine learning workflow, so the tools and workflows are stitch together which is time-consuming and error-prone. This also lead to a complex, expensive, iterative process made even harder in the traditional machine learning development. The AWS SageMaker resolve this challenge by providing all of the components used for machine learning in a single toolset so models get to production faster with much less effort and at lower cost.

4.2.1 Machine Learning workflow with Amazon SageMaker

A typical machine learning workflow and how the task can be accomplished with Amazon SageMaker is shown in Fig.14.

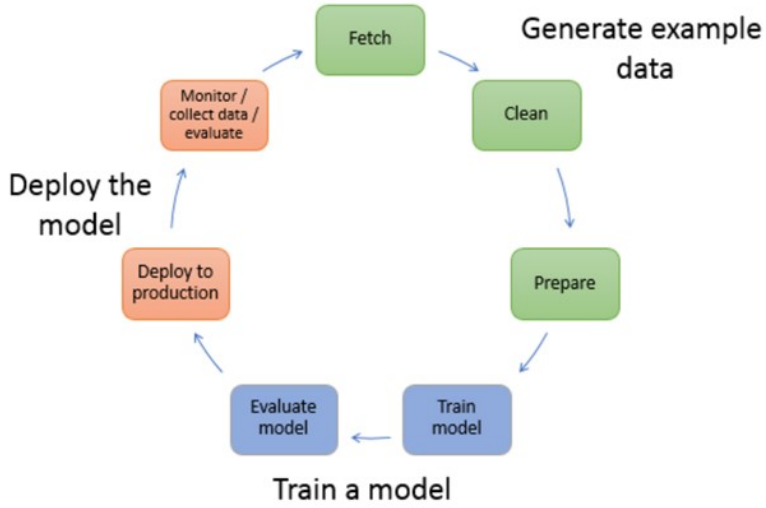

Fig. 14. Workflow for creating a Machine Learning model, adopted from AWS., 2020c

- *Generate example data:* An example of data is needed to train any model, and the data type depends on the business problem. A lot of time is spent exploring and preprocessing example data before using it for model training by Data scientists. To preprocess data, the initial task is to fetch the data from publicly available datasets or from in-house example data repositories. Then the next task is to clean and inspect the data to improve model training. This is followed by preparing and transforming the data to improve performance. In Amazon SageMaker, Preprocessing of example data takes place in a Jupyter notebook on the notebook instance. Notebook is used to fetch the dataset, explore it, and prepare it for model training.

- *Train a model:* This step involves two tasks which are training and evaluating the model. A training job is created, which includes information like the URL of the Amazon S3 bucket where the training data, as well as the output of the job, are stored. The training job also includes the Amazon Elastic Container Registry path, where the training code is stored. An algorithm is needed to train a model and the algorithm chosen depends on a number of factors. There are a list of algorithms provided by Amazon SageMaker and can be used for a quick solution. Compute resources are also needed for training, resources ranging from a single general-purpose instance to a distributed cluster of GPU instances can be used depending on the size of the training dataset and how quickly the results are needed.

After creating the training job, ML compute instances are launch by the Amazon SageMaker and uses the training code and the training dataset to train the model. The resulting model artifacts and other output are saved in the S3 bucket, which was specified for that purpose. The next task is to evaluate the model; the evaluation is done to determine whether the accuracy of the inferences is acceptable. The AWS SDK for Python (Boto) or the high-level Python library is used in Amazon SageMaker to send requests to the model for inferences. Fig.15 shows how a model is trained and deployed with Amazon SageMaker.

- *Deploy the model:* Model can be deployed independently, decoupling it from the application code with Amazon SageMaker hosting services. An HTTPS endpoint is provided by Amazon SageMaker where the machine learning model is available to provide inferences. Using Amazon SageMaker hosting services, a model can be deployed in a three-step process:
  - *Create a model in Amazon SageMaker:* A model is created which tells the Amazon SageMaker where it can find the model components. This includes the S3 path where the model artifacts are stored and the Docker registry path for the image that contains the inference code
  - *Create an endpoint configuration for an HTTPS endpoint:* The name of one or more models in production variants and the ML compute instances are specified so that Amazon SageMaker can launch to host each production variant. The endpoint can be configured to elastically scale the deployed ML compute instances when hosting models in production.
  - *Create an HTTPS endpoint:* The endpoint configuration is provided to Amazon SageMaker. The ML compute instances are launched by the service and deploy the model or models as specified in the configuration. Then the client applications send requests to the Amazon SageMaker Runtime HTTPS endpoint to get inferences from the model. The user's input data and ground truth are saved as part of the training data to increase model's accuracy. The model can then retrain periodically with a larger, improved training dataset.

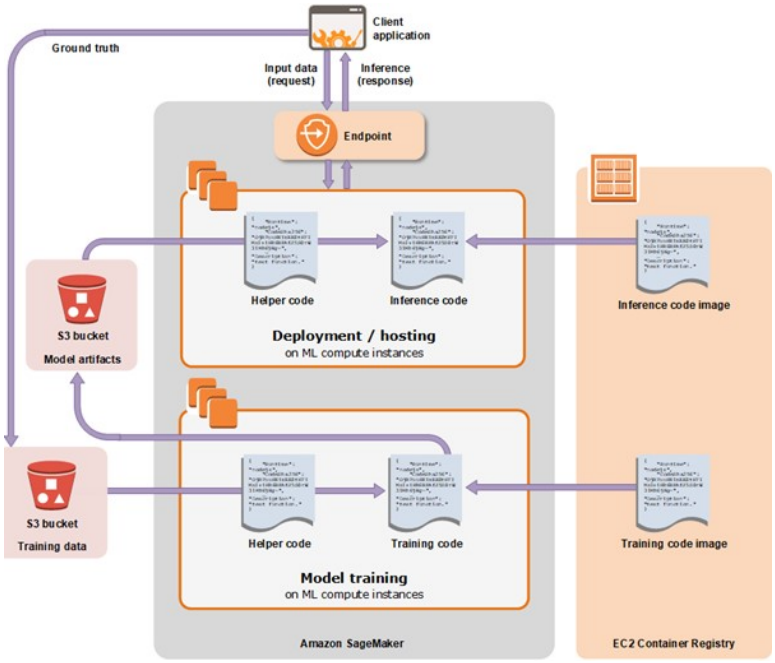

Fig. 15. Training and Deploying a model with Amazon SageMaker, adopted from AWS., 2020b

4.2.2 Benefits and features of Amazon SageMaker

As a single toolset, SageMaker offers a wide range of services, covering all of the main steps used for machine learning. The Fig.16 below shows the available Machine Learning services that Amazon SageMaker has been offering.

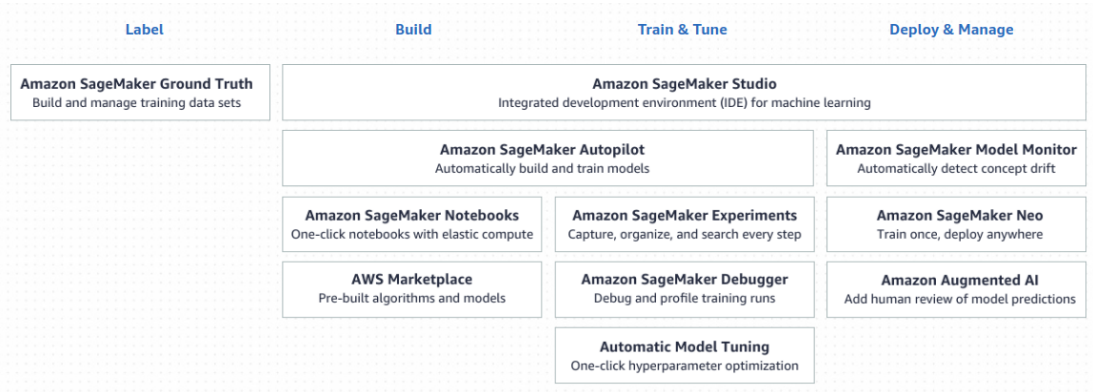

Fig. 16. Available Machine learning services provided by Amazon SageMaker, adopted from AWS., 2020a

- *Labelling raw data with active learning:* The training data are automatically labelled by Amazon SageMaker Ground Truth that uses a machine learning model. Data that cannot be confidently labelled by the model is automatically sent to

human labellers. In order to continuously learn and improve, the data which is labelled by a human is provided back to the model.

- *Fully-managed notebook instances:* Amazon SageMaker provides fully managed instances running Jupyter notebooks for training data exploration and preprocessing that includes example code for common model training and hosting exercises.

- *Highly accurate training datasets:* A very high level of consistency and accuracy for training datasets are provided by Amazon SageMaker Ground Truth. This avoids the burden of audits and asking the same data to be labelled multiple times to remove outliers.

- *Highly-optimised machine learning algorithms:* High-performance and scalable machine learning algorithms are installed by Amazon SageMaker, which are optimised for speed, scale, and accuracy, to run on extremely large training datasets.

- *One-click training:* Training can be initiated with just a single click by simply indicating the type and quantity of instances in Amazon SageMaker.

- *Deployment without engineering effort:* SageMaker provides model artifacts and scoring images after training for deployment to Amazon EC2 or anywhere else.

4.2.3 Amazon SageMaker Vs Other Competitors

Like AWS SageMaker, Microsoft and Google also provide a machine learning platform known as Azure ML Service and Google ML Engine, respectively. Azure ML Service is a machine learning SDK made for python. It ships as an SDK which makes it lightweight and a viable option for custom models but the necessary tools for monitoring and controlling the processes within a running pipeline is still not provided and is not a cloud provider agnostic. Cognitive Services is also provided by Microsoft, which is a collection of SDKs/APIs/complementing tools that make ML applications more intelligent as mentioned in Microsoft Azure website.

Similar to AWS SageMaker, Google ML Engine only supports one storage type, which is Google Cloud Storage equivalent to AWS S3. It doesn't support custom models and is also not a cloud provider agnostic as mentioned in Google Cloud Website. It is commonly paired with TensorFlow, but it does not mean that it is limited to using deep neural networks with this tool.

All three providers have the same ML services for managing particular tasks and it also comes down to project requirements, features, designs, budget, and even personal preferences as stated by Illia (2020). They are all committed to inventing new ML tools, services, and APIs to attract more clients with more advanced solutions. There are no significant gap in all market segments currently. However, Amazon being the first of the trio to invest heavily in the cloud it is trying to lead the way in the ML sphere offering a lot of interesting tools. Pricing is another major factor, choosing the right ML tools for right businesses is very crucial. The basic pricing of Machine Learning solution for top three cloud service providers is mentioned below in Table 1:

Solution	**Pricing**
AWS SageMaker	Building from 0.05 USD/hour. Training for the memory optimized instance from 0.15 USD/hour. Deployment from 0.07 USD/hour
Azure ML Service	Pay as you go price is 0.0176 USD/hour.
Google ML Engine	Pay as you go price is 0.6915 USD/hour

Table 1: ML Solution Basic Pricing

4.2.4 Getting started with Amazon SageMaker

Who is Amazon SageMaker for?

The purpose of Amazon SageMaker is to build a Machine Learning platform in which everybody could easily build, train and deploy models in the cloud, regardless of their machine learning expertise. In particular:

- *Machine Learning non-experts* could perform most of the necessary modelling tasks with Amazon SageMaker Autopilot Experiment, and AWS provided algorithms (e.g. Linear Learner, XGBoost, etc.) in order to develop and deploy simple machine learning models (regression and classification), thereby allowing data scientists to focus on more complex problems instead of models.

- *Machine Learning experts* (e.g. data engineers, data scientists, software engineer, etc.) could build, train and deploy AWS high-quality provided algorithms (e.g. DeepAR forecasting, BlazingText, etc.) or custom algorithms at scale and in real-time.

How to use Amazon SageMaker?

There are three ways to use Amazon SageMaker.

- Develop a model using AWS-provided algorithm
  - on SageMaker Studio
  - on Notebook Instance
  - on SageMaker Console
- Clone a model to train or use a trained model artifact from AWS Marketplace.
- Develop a model using user's custom algorithm
  - on SageMaker Studio
  - on Notebook Instance
  - with script mode
  - with docker container
  - with Notebook trained locally and SageMaker to run it in production

Within the scope of this thesis, the authors will build, train and test 2 models using AWS-provided algorithms on SageMaker Studio, Notebook Instance and SageMaker Console.

5 Implementation

In this thesis, the authors explore two use cases using AWS-provided algorithm. Briefs description of these use cases are presented below (Fig.17):

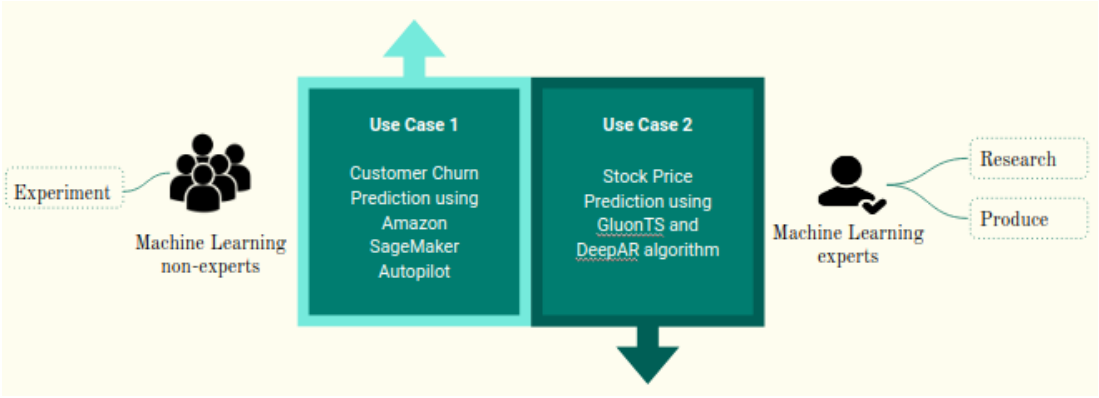

Fig. 17. Description of 2 use cases

5.1 Customer Churn Prediction using Amazon SageMaker Autopilot

5.1.1 Overview of Amazon SageMaker Autopilot, Amazon SageMaker Python SDK library and AWS built-in algorithms for SageMaker Autopilot

As introduced by Piali Das (2020), Amazon SageMaker Autopilot is fully managed system which offers an automatic machine learning solution (Jouganous et al., 2018). This system is suitable for users who do not have a machine learning expertise, therefore do not have a particular choice of algorithm to solve their business problem. SageMaker Autopilot could only be accessed via (1) Studio Experiment and (2) Jupyter Notebook. Given a tabular data set, the target column name, Autopilot help users analyze the data, engineer features, produce a diverse set of complete machine learning pipelines, tune hyperparameters, and generate a leaderboard of candidate models, as shown in Fig.18 below.

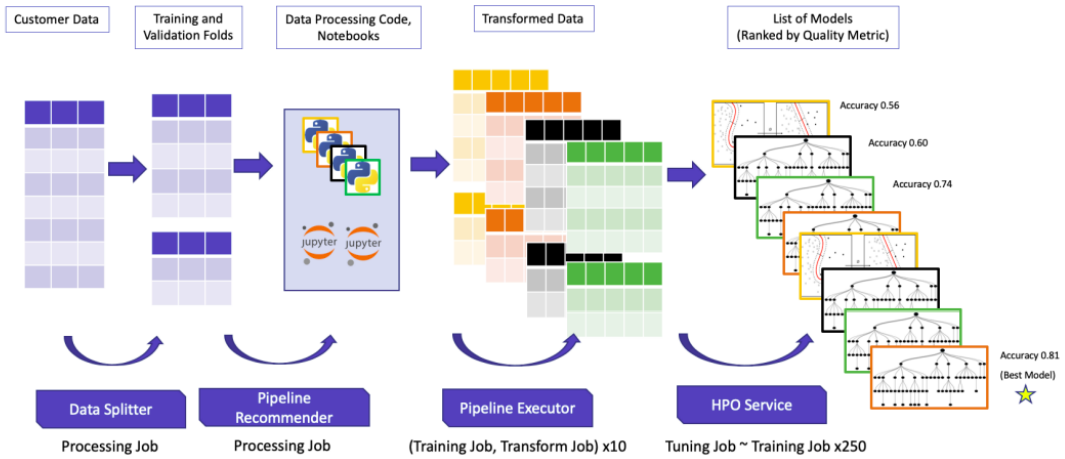

Fig. 18. A high-level view of component jobs that Autopilot executes while processing a customer request, adopted from Piali Das, 2020

The Autopilot automated process (Step 3 in each approach) has two main phases - *candidate generation* and *candidate exploration*.

- In the *candidate generation* phase, Autopilot automatically: (1) splits the input data into training and validation set; (2) infers the problem type by analyzing the values in the target column (if not provided by user); (3) generates a custom python module containing code to transform the input data in 10 different ways; (4) generates 10 tunable ML pipeline in *Candidate Generation notebook*, each pipeline set to use the respective transformed data; (5) generates a notebook with insights about the data - *Data Exploration notebook*.

- In the *candidate exploration* phase, Autopilot automatically performs advanced hyperparameter tuning techniques for up to 250 training jobs to find the pipeline that yields best prediction accuracy. Once finished, Autopilot generates inference pipeline definitions for each of the candidates which are ready to be deployed. Autopilot allows user to choose to execute only the *candidate generation* phase.

Amazon SageMaker Python SDK is an open source library used to train and deploy machine-learned models on Amazon SageMaker. API documentation and instruction for SageMaker Python SDK could be found at AWS, 2020d

For this classification problem, Amazon SageMaker Autopilot offers 2 built-in algorithms: (1) Linear Learner and (2) XGBoost. The instruction on how to use these algorithms is available at Guide, 2020.

5.1.2 Overview of problem and dataset

This use case is a binary classification problem. For this use case, the authors are going to predict the customer churn in a mobile phone operator by following the instruction in the notebook A. S. E. AWS, 2020. The dataset is obtained from the link http://dataminingconsultant.com/DKD2e_data_sets.zip, folder **Data sets**, file **churn.txt**. Collected by Daniel T. Larose from University of California Irvine and mentioned in Discovering Knowledge in Data book, this dataset consists of customer's profiles of an unknown US mobile operator. The dataset contains 32,169 instances, 21 features, in which the last feature ('Churn?') is the target. The feature descriptions are shown in Table 4 in Appendix B.

The authors hereby introduce two approaches to SageMaker Autopilot: (1) Studio Experiment and (2) Jupyter Notebook.

5.1.3 Approach 1: Amazon SageMaker Studio Experiment

Overview of Studio and Experiment

The Amazon SageMaker Studio is the first fully integrated development environment (IDE) for machine learning which helps to improve productivity. A single, web-based visual interface is provided by Amazon SageMaker Studio where all ML development steps can be performed. Complete access, control, and visibility into each step required to build, train, and deploy models is given by the SageMaker Studio. Data can be quickly uploaded, new notebooks can be created, models can be trained and tuned, the experiments can be adjusted by moving back and forth in between the steps, the results can be compared and deployed the models to production all in one place. All these features results in much more productivity. Within the unified SageMaker Studio visual interface, all ML development activities including notebooks, experiment management, automatic model creation, debugging, and model drift detection can be performed.

In machine learning thousands of different models are trained which involves many iterations to get maximum accuracy. So AWS introduces Amazon SageMaker Experiments to help organize, evaluate, compare and track iterations to machine learning models. The input parameters, configurations, and results are automatically captured and stored as 'experiments' by the SageMaker Experiments which helps in managing

the iterations.

For more details about how to use Amazon SageMaker Autopilot with this approach, the readers could consult the written tutorial at A. S. T. AWS, 2020 and watch video tutorial at Simon, 2019. Note: For Step 1 and Step 5, the authors create a Studio notebook inside the SageMaker Studio.

Step 1: Data Preparation

This step is conducted in a Studio Notebook. For the comparison purpose, the authors use the same training and test datasets for two approaches, so the authors apply the same code for both approaches.

Firstly, we need to split the dataset into the training and test datasets, then change the data sets format to **.csv** files, finally upload the two datasets to one S3 bucket by following the guideline at A. S. S. S. C. U. G. AWS, 2019. After uploading the two datasets, the users need to copy the S3 path to the training dataset and the S3 path to the output artifacts. These information would be used in the next step.

Step 2: Setup

This step is conducted in SageMaker Studio. An Experiment is created by following the guideline at A. S. T. AWS, 2020.

At this step, the user need to specify Experiment Name, S3 paths to the training dataset and the output artifacts, Target attribute name, Problem type (in this case, it is binary classification), and whether he or she want to run a complete experiment (Yes: execute both *candidate generation* phase and *candidate exploration* phase, No: execute only *candidate generation* phase). When choosing Objective metric, there are only 2 options: F1 and Accuracy. In this use case, the authors choose Accuracy, as the dataset is not very imbalanced with the churn rate in the raw dataset is 14.49%.

Step 3: Pre-processing and Training

This step is conducted automatically in SageMaker Studio. After creating a Autopilot experiment, the system would automatically run through 3 stages: (1) Analysing Data, (2) Feature Engineering and (3) Model Tuning. These stages are running in real-time, as in Fig.19 as follows.

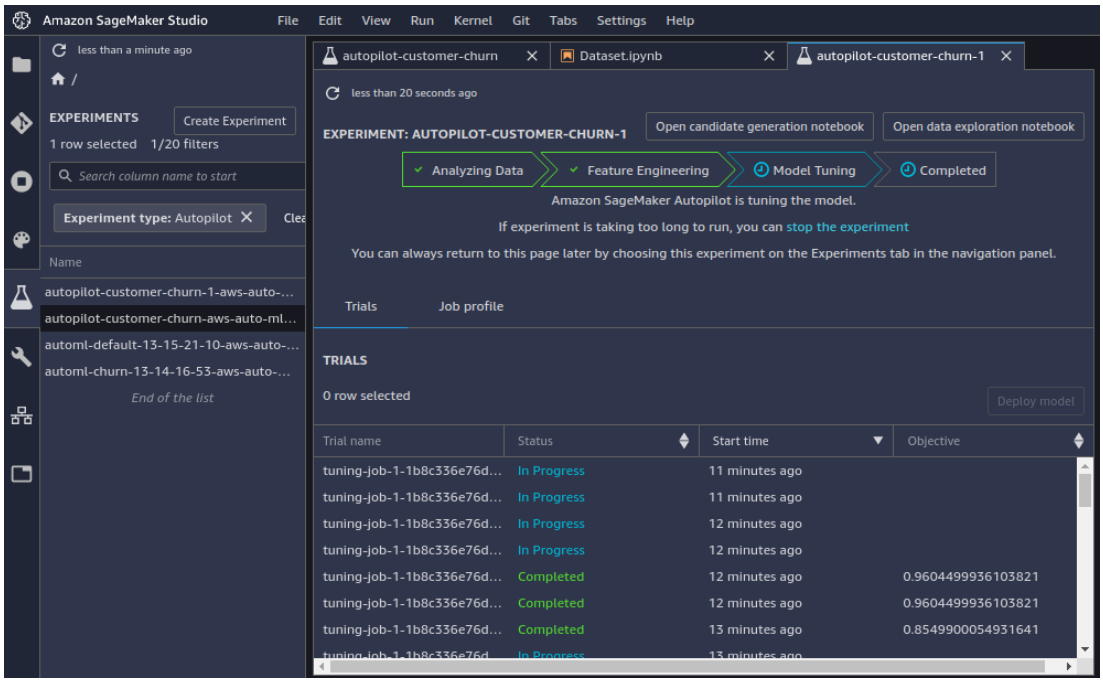

Fig. 19. Automated Processes in Realtime

However, if the user choose to execute only *candidate generation* phase, the experiment would stop once the Analysing Data stage is finished and two notebooks are generated: one is Data Exploration, and the other is Candidate Generation. The user continue to run the experiment by themselves by using the Candidate Generation notebook. Part of the Python code for this notebook is shown from Fig.33 to Fig.51.

- *Data Exploration notebook* presents the first 10 rows of the training dataset (Fig.29), percentage of missing values (Fig.30), number of unique instances in each feature (Fig.31) and descriptive statistics of numerical features (Fig.32).

- *Candidate Generation notebook* shows 10 candidate models and the code needed to configure Autopilot jobs and S3 Bucket, transform data, tune the hyperparameter, deploy the best model and clean up the endpoint. Part of the codes in this notebook are presented in Appendix B, including the configuration of Autopilot jobs and S3 Bucket (Fig.33 and Fig.34), generated candidates and algorithms list (from Fig.35 to Fig.45), Multi Algorithm Hyperparameter Tuning (Fig.46 to 48), Tuning Result (Fig.49), and Model Deployment (Fig.50 and 51).

As soon as 3 stages ((1) Analysing Data, (2) Feature Engineering and (3) Model Tuning) are completed, Autopilot automatically chooses the best trial model, based on the

Objective metric as illustrated in Fig.20 below. However, the users could still explore the results of all the trial models and choose the trial model of their choice, based on other metrics, in Trial Component List as shown in Fig.21 below.

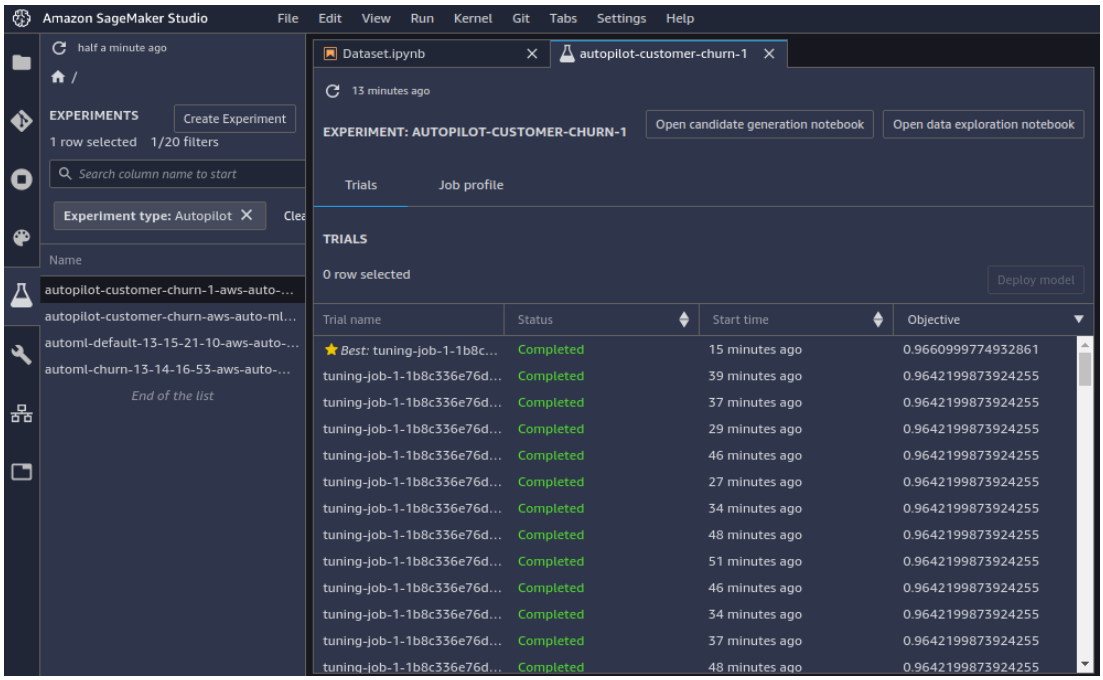

Fig. 20. Autopilot automatically chooses the best model after Training jobs

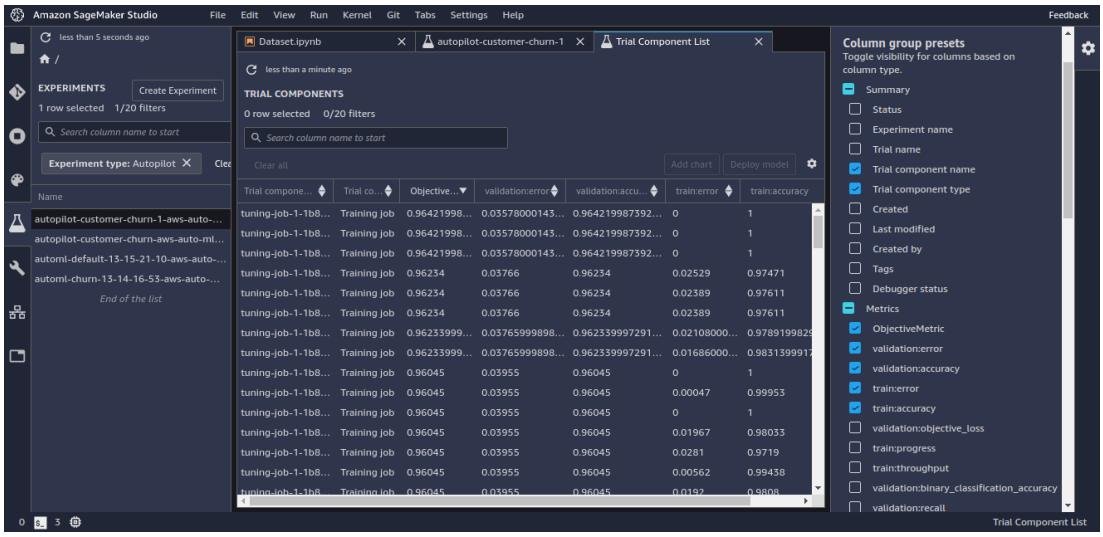

Fig. 21. Trial Component List

Step 4: Model Deployment

This step is conducted in SageMaker Studio. The user could choose to deploy the best model (as shown in Fig.20) or the model by their choice (from Trial Component List as shown in Fig.21) to a hosted endpoint, by following the guideline at A. S. T. AWS,

2020. At this step, the user need to specify the end point name and the S3 path to your end point, as illustration in Fig.52. The endpoint would be stored in this s3 bucket to use for Step 5 - Testing.

Step 5: Test and Cleanup

This step is conducted in a Studio Notebook.

Firstly, the users need to upload the test dataset to the SageMaker Studio. SageMaker Notebook then run the model from the endpoint on the testing dataset. The accuracy on the testing dataset is 0.9625.

To avoid any incurring unnecessary charges in the future, as soon as the test phase is finished, the users should clean the endpoint, the endpoint configuration, the model and other artifacts generated during the training process.

5.1.4 Approach 2: Notebook Instance

Overview of Notebook Instance

AWS introduced an Amazon SageMaker Studio Notebooks which are one-click Jupyter notebooks that can be spun up quickly. The available resources can be easily dialled up or down as the underlying compute resources are fully elastic and without any interruption to the work the changes take place automatically in the background. The notebooks can also be easily shared with others and can be found exactly in the same environment as the original notebook. This helps in reproducing the result easily and collaborate while building models and exploring data.

For more details about how to use Amazon SageMaker Autopilot with this approach, the readers could consult the notebook tutorial at A. S. E. AWS, 2020 and Simon, 2020.

Step 1: Notebook Setup and Data Preparation

The setup starts by specifying the S3 path to input/output data, and the IAM role. Then, the authors import the Python libraries for machine learning, including Numpy, Pandas and Matplotlib, and SageMaker Python SDK library. For the comparison purpose of this thesis, the authors use the training and testing data sets like Approach 1's.

Step 2: Autopilot Setup

The illustration of this step is shown in Fig.53. Firstly, the users need to specify the S3 path to input/output data, and the target feature ('Churn?'). Secondly, the users create an Autopilot job by using `create_auto_ml_job` function from SageMaker SDK library and specifying the problem type (binary classification) and the objective metric (Accuracy).

Step 3: Pre-processing and Training

By calling the function `describe_auto_ml_job`, the status of the current stage ((1) Analysing Data, (2) Feature Engineering or (3) Hyperparameter Tuning) is shown in real time in the notebook. Once all three stages are finished, the best model that Autopilot picked based on the final validation accuracy is shown in Fig.54. Also, the user could explore other candidates sorted by the validation accuracy by using the function `list_candidates_ for_auto_ml_job`.

Step 4: Model Deployment

In this step, the users start by define the best model name, the model arn (the S3 path to model artifacts), the endpoint name, endpoint S3 path and finally using the function `create_endpoint`.

Step 5: Test and Cleanup

The author use the same code as in the Approach 1. The accuracy on the testing dataset is 0.9565.

To avoid any incurring unnecessary charges in the future, after testing the best model and store the endpoint in local computer, the users should clean the endpoint, the endpoint configuration, the model and other artifacts generated during the training process.

5.1.5 Comparison between Autopilot on Studio Experiment and Autopilot on Notebook Instance

In this section, the authors compare between the two approaches shown above, in terms of (1) Accuracy, (2) Workflow, (3) Interpretability and (4) Transparency.

Criteria 1: Accuracy

The Test Accuracy for the best model in both approaches are roughly the same, with 96.25% for Approach 1 and 95.65% for Approach 2. This observation shows that both the Amazon SageMaker Python SDK for Autopilot and Autopilot Studio Experiment are stable.

Criteria 2: Workflow

The authors observed that Studio notebook makes the Data Preparation and Testing a lot easier because starting a Studio notebook is typically faster than launching an notebook instance.

Criteria 3: Interpretability

- Data Exploration notebook from Studio helps users explore the descriptive statistics with no code needed.
- Trial Component list allows users to compare the metrics among trial models via CloudWatch.

Criteria 4: Transparency

- Candidate Generation notebook from Studio helps users examine candidates' hyperparameters.
- Trial Component list: all of model's parameters, configurations and results.

With the comparison above, the authors recommend using SageMaker Autopilot in SageMaker Studio, which is the Approach 1.

5.2 Stock Price Prediction using GluonTS library and DeepAR algorithm

5.2.1 Overview of GluonTS library and DeepAR algorithm

For this use case, the authors use DeepAR forecasting - a supervised learning algorithm for forecasting scalar (one-dimensional) time series using recurrent neural networks

(RNN). The readers could refer to DeepAR documentation at Amazon SageMaker Developer guide, 2020 and Salinas et al., 2017. Since Amazon SageMaker Python SDK for DeepAR forecasting algorithm is in progress, the authors employ GluonTS - an open source library for time series forecasting problem. GluonTS tutorials are available at Jasper, 2020 and GLUON, 2020.

5.2.2 Overview of problem and dataset

This use case is a time series regression problem. For this use case, the authors are going to predict the stock price of Bayerische Motoren Werke AG company (BMW) by following the instruction in the notebook of Binoy, 2018. The authors have used Deutsche Börse (DBG) Public Dataset consisting of trading data from 1 Jul 2017, aggregated to daily intervals from the Xetra trading systems, which is reference market for exchange trading in German shares and exchange-traded funds (ETFs)). The data is uploaded into two Amazon S3 Buckets in the EU Central (Frankfurt) region at the s3 location `s3://deutsche-boerse-xetra-pds`. The dataset contains 3,333 instances and 14 features, in which the target feature is `EndPrice` and the covariate features are `StartPrice, MinPrice and MaxPrice`. The feature descriptions are shown in Table 5 in Appendix C.

The authors choose not to use hyperparameter tuning and cross-validation for this DeepAR application, because of the reasons as follows.

- Hyperparameter tuning is expensive for DeepAR algorithm, and it adds up not much benefit for the model with default hyperparameters, which could yield good enough prediction with low RMSE (gluon-ts GitHub, 2020).
- Hyperparameter tuning method is not open sourced in GluonTS.
- In order to perform k-fold cross validation, users need to create, deploy and make prediction on k models, leading to a high amount of charge.

Although SageMaker Studio is good for Autopilot job, but it is still not matured enough for complicated tasks (research and production workflows), the authors choose to exclude SageMaker Studio from this 2nd use case. The authors hereby introduce two approaches to AWS built-in algorithms: (1) SageMaker Console and (2) Notebook Instance.

5.2.3 Approach 1: Amazon SageMaker Console

Overview about SageMaker Console

Except for Notebook Instance and Studio, there are 6 main features of SageMaker Console, which are (1) Ground Truth, (2) Processing, (3) Training, (4) Inference, (5) Augmented AI and (6) AWS Marketplace. Their purpose are briefly described as in Table 2 as follows.

Features	**Description**
Ground Truth	Custom or built-in data labeling workflows
Processing	Pre- or post-processing and model evaluation workloads.
Training	Train and tune models at any scale.
Inference	Create models from training jobs or import external models for hosting to run inferences on new data.
Augmented AI	Add human review of model predictions.
AWS Marketplace	Pre-built algorithms and models.

Table 2: Features Descriptions of SageMaker Console

Hereby we will introduce 5 steps of using SageMaker Console: (1) Pre-processing, (2) Train and Test, (3) Deploy, (4) Inference and (5) Cleanup. For Pre-processing step, the authors introduce in details how to use docker container to pre-process data once users have pre-processing image of trained model. However, as we experiment with two approaches in parallel, we do not have access to a pre-trained DeepAR model in docker container. Therefore, we will use the processed dataset from Notebook Instance approach.

Step 1: Pre-processing

Firstly, users have to create a container folder, which is a registry path where the processing and training images is stored in Amazon ECR (Fig.55).

Next, the users copy the ECR container location (URL) and provide the local path to processing artifact folders and S3 path to the input/output folder as in Fig.56 and Fig.57.

After completing the processing job, the training and test datasets are generated as JSON files and stored in provided S3 location.

Step 3: Train and Test

The Training step in Console is pretty simple, provided AWS built-in algorithm. The authors choose an algorithm from dropdown list (Fig.58), then specify the hyperparameters (Fig.59), finally provide S3 path to training and test datasets (Fig.60), and S3 path to output folder (Fig.61).

The training and testing logs could be accessed on CloudWatch (Fig.22) below.

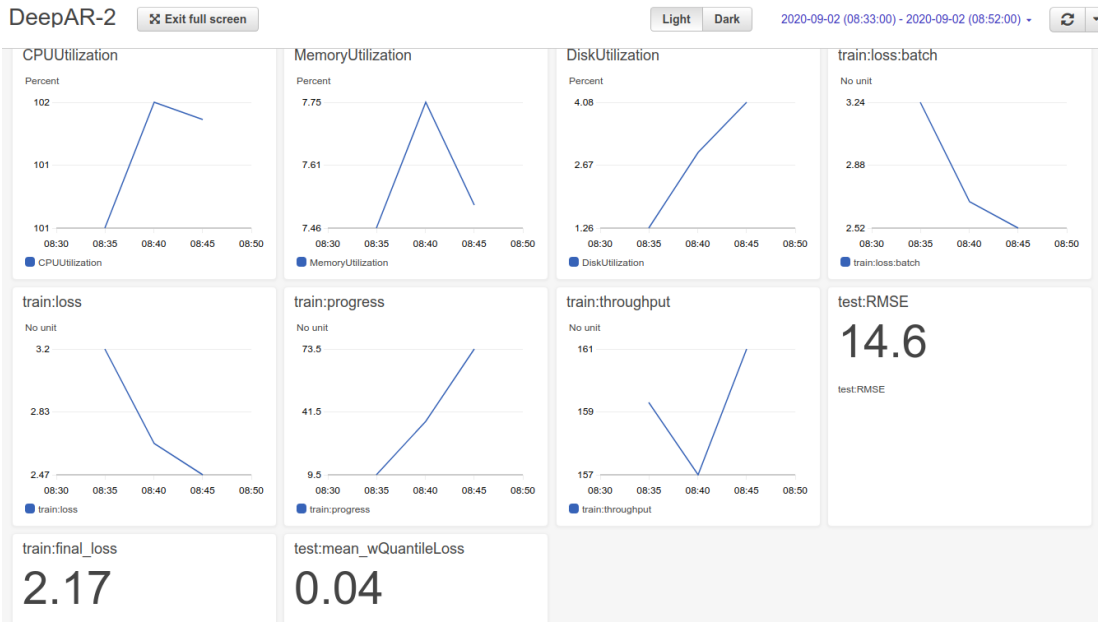

Fig. 22. Stock Price Prediction - Approach 1: Training Logs

Step 3: Deploy

Once the users decide to deploy the tested model, the users could create a model by click on the training job and choose 'Create model' from the top-right dropdown list. The created model is shown in Fig.62.

Step 4: Inference

In the Inference step, the users have 2 options:

- Prediction on an entire dataset: use Batch Transforms without create an endpoint. All users need to do is to specify the model name, input/output S3 location and Batch strategy (SingleRecord or BatchStrategy). After being created, Batch Transforms would read a trained model and the input dataset (located on a S3 bucket), then save the inferences in an specified S3 bucket.

- Real-time prediction on large scale datasets: create an endpoint configuration and an HTTPS endpoint. To perform real-time prediction, users need to set up an eco-system among AWS services, e.g. create triggering rules from CloudWatch. An example of this eco-system (Correa, 2019) is shown in Fig.23 below.

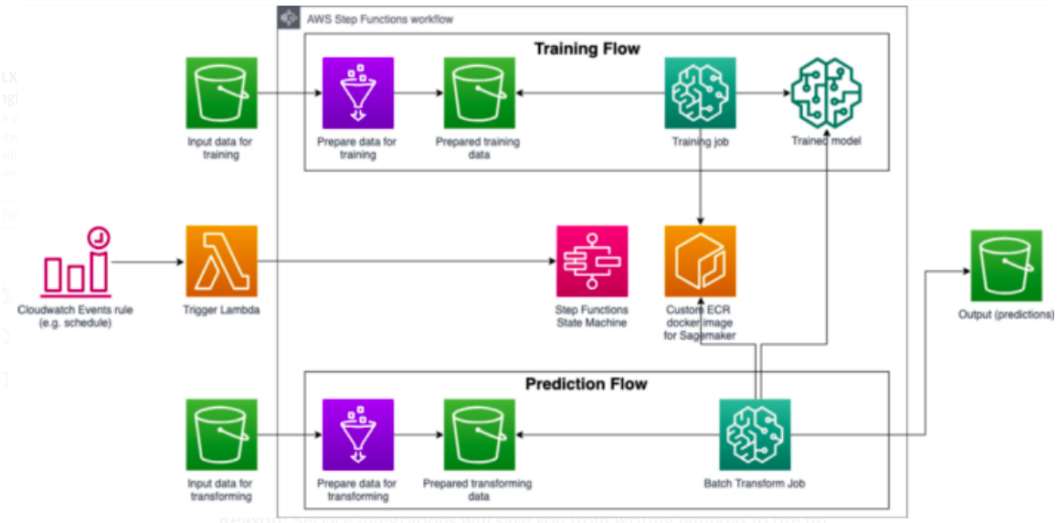

Fig. 23. Stock Price Prediction - Approach 1: Suggested Machine learning workflow for production, adopted from Correa, 2019

Step 5: Cleanup

As usual, the last step is cleanup the models, artifacts and endpoints that is not necessary anymore. Cleanup could be performed by manually deleting the models, endpoints on Amazon Console and deleting artifacts on S3 bucket.

5.2.4 Approach 2: Notebook Instance

Step 1: Notebook Set up

After login to the AWS Console, we start by creating a Lifecycle configuration (Fig.63) which are small scripts used to automate certain tasks when a notebook instance is being created and/or being started. Once the Lifecycle configuration is created, it is used to create a notebook instance.

Next, a small instance class ml.t2.medium is selected as the Notebook Instance type (Fig.64). The instance could be changed for each Jupyter Notebook, especially when we need a faster instance for training and testing.

Finally, an IAM role is created (Fig.65) which needs to be authorized to create an Amazon ECR repository and upload an container image to the repository. An ordinary IAM role on SageMaker only has permission to work on SageMaker. If we want to interact with other AWS services, such as Lambda or ECR, we would have to add a new policy/ attach an existing policy that give our IAM role the permissions on those platforms. In this use case, we need to create an container on Amazon ECR to upload the images from our training job. Therefore, we added five permissions to a new custom policy in our IAM role, namely : CreateRepository, InitiateLayerUpload, UploadLayerPart, CompleteLayerUpload and PutImage. The new policy is illustrated as in Fig.66.

Step 2: Data Preparation and Pre-processing

Data Preparation

The Xetra DBG dataset is stored in a public S3 separate folders by date, so we have to generate a series of scripts to download the data files. We download 16 months of data from 01 Jul 2017 to 31 Oct 2018. All the downloaded data files are stored in a local folder within, prior to pre-processing the data. Then the processed dataset is stored in another folder which will be followed by uploading it to the SageMaker S3 bucket. Then the download script is generated in the script folder and executed to have the files downloaded.

Data Pre-processing

All the csv files are loaded into a single data frame, as described in Fig.67. Then next step is to filter the records pertaining to common stack with trading volumes greater than zero, and within the regular trading hours. A clean data frame is build containing minute by minute transaction records with data points namely: `Mnemonic`, `Minimum Price`, `Maximum Price`, `Start Price`, `End Price`, `Trading Volume` and `Number of Trades`. The processed data are finally saved and uploaded to SageMaker S3 Bucket. In order to do more effective hourly, daily or weekly predictions, the data is resampled at various interval levels. The file is then uploaded to an appropriate S3 location directly after resampling of the intervals (Month, Week, Day and Hour). After the pre-processed data are saved to an S3 bucket, the original data files are deleted locally to conserve storage space on notebook instance.

Step 3: Feature Selection by Data Exploration and Data Clustering

To build high-quality model, it is necessary to select the features that are most important to predict the target feature. To meet that end, we will explore the time series dataset and apply a clustering algorithm to find the related tickers that have high influence on BMW stock movement.

Data Exploration

The result of Data Exploration is shown in Fig.68 in Appendix C. There are 4 observations here.

- The `EndPrice` for 5 automobile companies - BMW, Daimler, Porsche, Volkswagen and Continental (`BMW, DAI, PAH3, VOW3, CON`) - are in the same pattern.
- The `EndPrice` is closer to the `MinPrice` when there is a downward trend and to `MaxPrice` when there is an upward trend.
- The `StartPrice` is closer to the `MaxPrice` when there is a downward trend and to `MinPrice` when there is an upward trend.
- `EndPrice` is above `StartPrice` when the trend is upwards and below `StartPrice` when the trend is downwards.

With these observations, the authors develop an intuition that behaviour of the various metrices present in the DBG data set might have a correlation with future movement of the time series, and the time series movements of companies in the same industry are correlated. We then will apply `HDBSCAN` clustering algorithm to check this intuition and select 4 stocks along to predict the time series movement of BMW stock.

Data Clustering

As we observed during data analysis phase that there are certain stocks that are clustered relatively more, based on their past price trends. Naturally, using these similarly moving stock would increase overall model performance, in absence of any other available external time series.

Using HDBSCAN clustering algorithm, similar stocks are found out. First the data re-sampled at daily interval are loaded from the S3 bucket location. Then a time frame is selected within which to analyze the stocks, followed by selection of an interval within which to aggregate the prices. A function of the price is selected such as percent change. A similarity function between the timeseries is selected such as dot product, cosine or correlation coefficient. Then a clustering algorithm is selected and the results are visualized.

From the result of clustering we see that somewhat similar stocks clustered together. For an instance, we can see from in Fig.24 that all the automobile companies such as BMW (BMW), Volkswagen (VOW3), Continental (CON), Daimler (DAI), Porshe (PAH3) and Covestro (1COV) are grouped together.

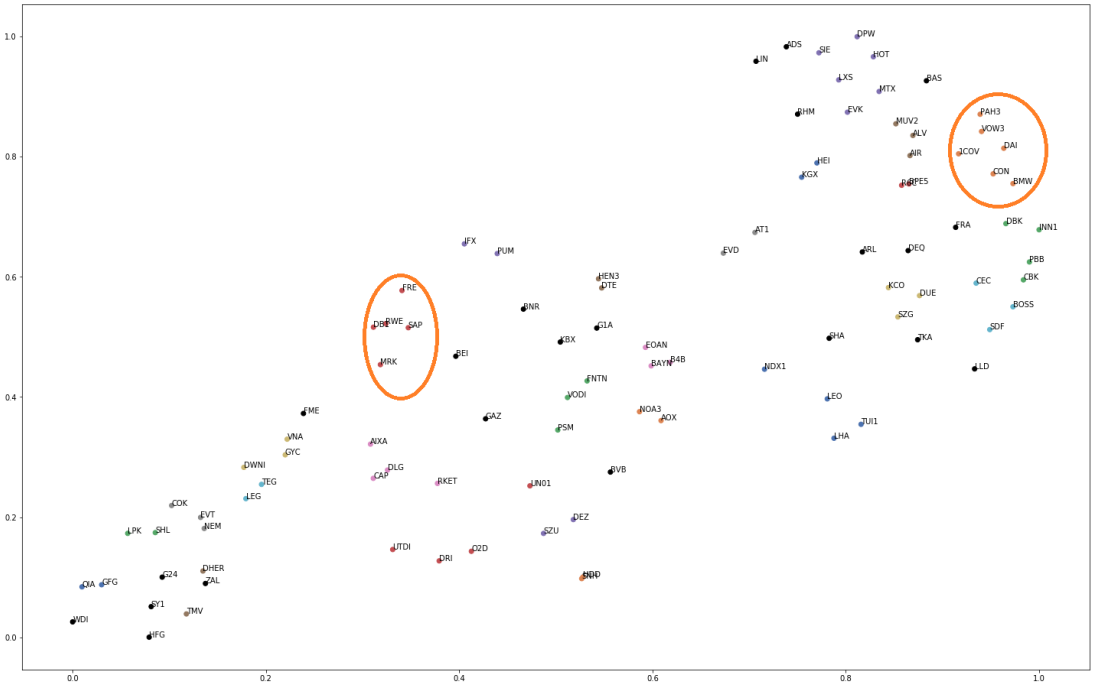

Fig. 24. Clustering Result

Step 4: Train and Test

To get started, the authors define the time interval (`'D'`), select the covariate ticker (`'CON','DAI','PAH3','BMW','VOW3'`), the target feature (`'EndPrice'`), covariate features (`'StartPrice', 'MinPrice', 'MaxPrice'`). Before splitting dataset, the authors need to do some pre-processing to have the data formatted following DeepAR Input/Output interface. The function named `deeparize` does this transformation. The

transformed dataset is illustrated as in Fig.69 in Appendix C.

Next, the authors split the transformed dataset into 2 parts (training (80%) and test (20%)) and transform them to JSON files. Finally, the authors use `DeepAREstimator` class from GluonTS library and call the method `train` to submit a training job, provided DeepAR hyperparameters. The training loss per epoch is shown in Fig.70. Once the estimator is fully trained, we generate predictions from it for the test values. The test metrics is presented in Fig.25 below. Final Test RMSE = 12.01.


```
"MSE": 144.13641629732572,
"abs_error": 2191.0641326904297,
"abs_target_sum": 31252.248779296875,
"abs_target_mean": 96.16076547475961,
"seasonal_error": 0.9662580428584929,
"MASE": 5.213259141800666,
"MAPE": 0.05764099816775735,
"sMAPE": 0.054051056718994225,
"OWA": NaN,
"MSIS": 208.91195481967847,
"QuantileLoss[0.5]": 2191.0640754699707,
"Coverage[0.5]": 0.9353846153846155,
"QuantileLoss[0.67]": 1542.811789398193,
"Coverage[0.67]": 0.9815384615384616,
"QuantileLoss[0.95]": 278.0602039337161,
"Coverage[0.95]": 0.9938461538461538,
"QuantileLoss[0.99]": 58.623118515014696,
"Coverage[0.99]": 0.9969230769230769,
"RMSE": 12.005682666859295,
"NRMSE": 0.12485011540398522,
"ND": 0.07010900713621303,
"wQuantileLoss[0.5]": 0.0701090053052901,
"wQuantileLoss[0.67]": 0.049366424806532076,
"wQuantileLoss[0.95]": 0.008897286268817804,
"wQuantileLoss[0.99]": 0.0018758048078079334,
"mean_wQuantileLoss": 0.03256213029711198,
"MAE_Coverage": 0.19942307692307695,
"MRMSSE": 12.845471048355103
```


Fig. 25. Stock Price Prediction - Approach 2: Test metrics result

The author also specify the quantiles to get results at various confidence levels.

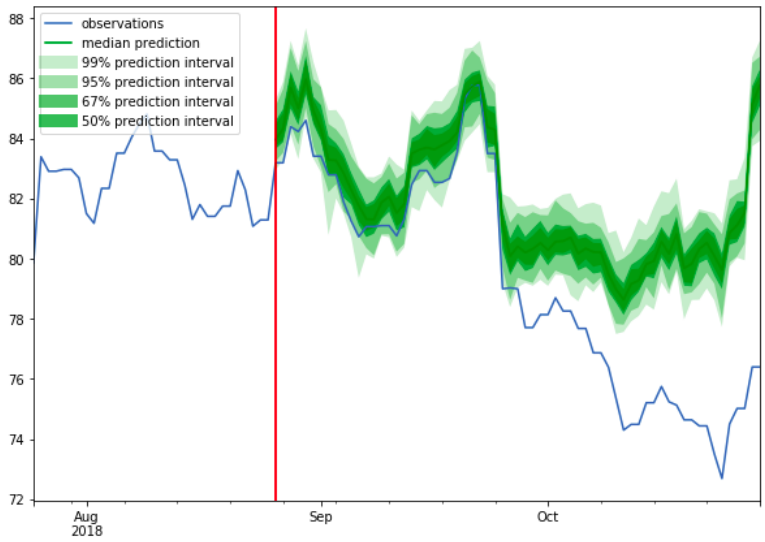

Fig. 26. Stock Price Prediction - Approach 2: Visualisation of predicted and actual data

Step 5: Deploy, Inference and Cleanup

Firstly, the authors use the trained model (`DeepAREstimator`) to create a hosted model definition, configuration and endpoint, and then deploy the model to the endpoint. For inference purpose, users could apply the 2 Inference options as described in Approach 1. Finally, the users should delete the the endpoint after finish making inference to avoid unexpected cost. In order to contain cost overrun, you should always delete the unused endpoints. However, the endpoint can be re-provisioned at anytime by using the corresponding training job name. Finally, users should stop the unnecessary Notebook Instances once they finished.

5.2.5 Comparison between AWS Console and Notebook Instance approaches

In this section, the authors compare between the two approaches shown above, in terms of (1) RMSE, (2) Needed coding skills, (3) Application in Research workflow and (4) Application in Production workflow.

Criteria 1: RMSE

The test RMSEs for the DeepAR model in both approaches are considerably different, with test RMSE = 14.6 for Approach 1 and test RMSE = 12.01 for Approach 2. This observation shows that the DeepAR model artifact in Amazon SageMaker Console is less effective than that of Amazon's GluonTS library.

Criteria 2: Needed coding skills

The Notebook Instance provides users with conda-installed environment and other data science packages. However, it requires users should have a decent knowledge of Amazon SageMaker Python SDK. On the other hand, with SageMaker Console, users only need to perform data processing, training and deployment by navigating through the interface, without any coding skill except for AWS services.

Criteria 3: Application in Research workflow

The main advantage of Jupyter Notebook is that it assists data scientists with data exploration. Also, data scientists could easily execute and share Jupyter Notebooks like 'technical' papers. However, to access Jupyter Notebook on cloud-based platform, data scientists still need to have a decent knowledge of cloud security and networking, lifecycle configuration. That's when data scientists need to use the Console.

Criteria 4: Application in Production workflow

Once data scientists finished the research workflow in Jupyter Notebook, it is difficult to keep these notebooks for inference in production when the organisation does not have an advanced infrastructure (Rodriguez, 2018). Usually at this stage, software engineers would take over the work from data scientists to make real-time inference. Software engineers use script mode, docker container to process, retrain on new data and deploy model to use at scale. This work could not be done without SageMaker Console.

6 Limitations and Recommendations

6.1 Limitations of SageMaker

After exploring SageMaker with 2 use cases, we assemble its limitations as follows.

- SageMaker has complex connections with other AWS services and requires a decent engineering knowledge.
- Local mode is only available in Notebook Instance, not in Studio Notebook.

- Data pre-processing and Cross-validation are not well-supported.
- Detailed permissions system is needed when starting notebook instance and when onboarding to Studio.
- Python SDK is incomplete and some algorithms are not open-sourced.

6.2 Limitations of our use cases

Due to our limited time and knowledge, we could not cover all AWS Machine Learning services (Augmented AI, Forecast, Fraud Detector, Personalize, Rekognition, etc.) and approaches (script mode, docker container, AWS Marketplace and local model training). Also, we have not fully explore Inference step, including Batch Transform and Incremented prediction in real-time, in Use Case 2.

6.3 Recommendations for SageMaker practitioners

Given all limitations above, we recommend SageMaker practitioners should have an in-depth research on other AWS services (S3, ECR, Step Functions, CloudWatch, Glue, Lambda) and keep up-to-date with AWS SageMaker newly-launched features and libraries.

7 Intergration

7.1 AWS Products - Module Integration

Business Analytics is something that every organisation is trying its best to utilise it for their decision-making purposes. It basically includes quantitative and statistical analysis, predictive modelling, data mining, and multivariate testing. Business Analytics helps to explore management issues and solve managerial problems by answering questions about a business using information or data gathered about the business. The questions can be answered by using many techniques, querying the data and then analyse the results of the queries.

The Master's in Business Analytics course at UCD Smurfit currently offers 6 core modules and 6 option modules (or 4 option modules + Capstone project) divided across three trimesters. All the modules are extensively covered in such a way that it enhances our technological skills and business knowledge, which are key requirements in modern knowledge-based organisations. There are many AWS products that can be integrated with the modules and are highlighted below:

Module	**AWS Product**
MIS41150: Introduction to Business Analytics	AWS Cloud Security
MIS41110: Programming for Analytics	AWS Developer Tools
MIS41130: Statistics & Simulation Methods	AWS Analytics
MIS41270: Data Management & Mining	AWS Machine Learning
MIS41040: Business Intelligence & Visual Analytics	Amazon QuickSight
MIS41050: Consulting Change & Project Management	AWS IoT, AWS Management and Governance
MIS41120: Statistical Learning	AWS Machine Learning

Table 3: Business Analytics Module Integration

7.2 Learning Outcomes

- *MIS41150 Introduction to Business Analytics:*

  The module is designed to explain the role of business analytics in the organisation. Understand the different sub-topics of business analytics and their relation to the organisation's goals. It also covers different issues of security, privacy and ethics related to business analytics. The product Security, Identity and Compliance under AWS Cloud Security could be integrated into this module. This product provides services that help to protect data, manages identity and access, also gives comprehensive compliance status and data privacy.

- *MIS41110 Programming for Analytics:*

  The module teaches the basic of programming in sufficient detail to survive in analytics. The program helps to convert, clean, manage and interpret data. The product AWS Developer Tools could be integrated into this module as it helps

to host code, build, test and deploy applications. It implements continuous integration and delivery to accelerate software development and release cycle.

- *MIS41130 Statistics & Simulation Methods:*

  The module mainly focused on learning probability and statistics along with the fundamentals of simulation. The ultimate scope is to learn how to perform data analysis and solve real-world business questions. The product Data Lakes and Analytics under AWS Analytics could be integrated into this module. It helps in collecting, processing, and analysing real-time, streaming data to get timely insights and react quickly to new information.

- *MIS41040 Business Intelligence & Visual Analytics:*

  The module mainly covers topics on Decision Support Systems, business intelligence and visualisation. It provides us with the knowledge to access the importance of the current developments to decision-making systems. The Amazon QuickSight could be integrated into this module. It is a fast, cloud-powered business intelligence service that easily helps to create and publish interactive dashboards that include ML Insights.

- *MIS41050 Consulting Change & Project Management:*

  The module covers the consulting and change management with different approaches to project management. It includes describing the consulting environment, including technology planning. The AWS Management and Governance services, which provides business agility and governance control could be integrated into this module. This service can be used to assess resource utilisation and identify ways to reduce costs. Also, the AWS IoT provides services for industrial, consumer and commercial solutions. AWS IoT brings together data management and rich analytics specially designed for noisy IoT data.

- *MIS41090 Advanced Operations Research, MIS41270 Data Management & Mining & MIS41120 Statistical Learning:*

  These modules cover different applications of Network models in analytics, helps to understand the strength and weaknesses of data mining as part of a decision support environment and also covers machine learning concepts. The AWS Machine Learning service could be integrated into all these modules as it

offers the broadest and deepest set of machine learning services and supporting cloud infrastructure. The Amazon SageMaker Autopilot has an automated machine learning capability. It helps in exploring data, engineering features, trying different algorithms, and selecting the best model. This service is very much advisable for Data Management & Mining and Statistical Learning modules where the fundamentals of Machine Learning is extensively covered.

8 Adoption Practice

8.1 Cloud Computing Adoption Framework

Cloud Adoption Framework is created to help organisations design and travel an accelerated path to successful cloud adoption. As mentioned in the AWS website (CAF, 2020), the framework may vary depending on the business, but the general purpose of it is to provide guidance and best practices to build a comprehensive approach to cloud computing across the organisation and throughout IT lifecycle. A generic, widely-applicable and implementable cloud computing adoption framework (Paredes-Gualtor et al., 2017) which consist of six phases are shown and discussed below in Fig.27:

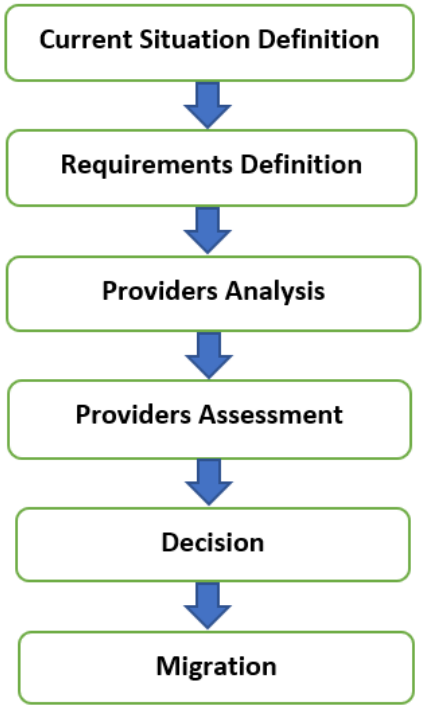

Fig. 27. Cloud Adoption Framework, adopted from Paredes-Gualtor et al., 2017

- *Current Situation Definition:* The first step is to define the company-specific-scenario before adopting cloud computing services. This process involves the study of organisational, which is an internal characteristic and environmental, which is an external characteristic. The organisational context describes different attributes of the company such as the size and age of the company, product/service, infrastructure and customers.

  To define the company's size, the most widely accepted concept is based on the number of employees and annual revenue. The number of years the company has been on the market shows the company's age and how it has handled its growth as well as market changes. The main differentiators for the cloud service each company may need are the products or services the company offers. Infrastructure is another attribute which refers to network technologies and enterprise systems installed. It is a platform on which cloud computing applications can be built. Then comes the customers of the company that can lead the company in a certain direction based on customer distribution and role.

- *Requirements Definition:* The cloud computing quality-criteria such as performance, reliability, capacity, availability, security, regulations, and operativity needs to be defined before setting the business's objectives. Each of these criteria must be studied for each service the company needs, such as email or file storage.

- *Providers Analysis:* Once the company's real situation and its needs are established, the next step is to choose the best provider and determine the available alternatives. The leading three cloud computing providers, according to Gartner (Leong et al., 2017) are Amazon Web Services (AWS), Windows Azure and Google Cloud Platform.

- *Assessment:* The business goals are validated with the company's current situation and the provider's alternatives. Technical feasibility, Economic feasibility and Legal feasibility all these aspects are evaluated.

- *Decision:* There are several decision-making models that influence an organisation's decision to adopt the cloud and its return on investment (ROI). The model helps to support decision-makers in evaluating the benefits, risks and costs of

using cloud computing. The specific requirement of the business will help in making the decision to select the most suitable cloud service.

- *Migration:* Once the decision is made to switch to cloud computing provider, migration takes place which is a 7-step process. The first step is to get familiar with the cloud computing environment that the company is going to use. Then support is build among members of the organisation. This is followed by testing existing software in the cloud and migrating the company data. Next, the company software applications are migrated, After this, security, scalability, and maintenance options are configured. The process is completed by the deployment phase.

8.2 Cloud Computing - Security Risks Assessment

IT risks are in common with any externally provided service in cloud-computing environments, as stated by (Heiser & Nicolett, 2008). It is very important for any organisation who are considering to use cloud-based services to understand the associated risks, defining acceptable use cases and necessary compensating controls. Below are important points to evaluate before using any external services:

- *Privileged User Access:* Organisational managers are not much aware of the nature and level of risk, and sometimes they don't have direct ability to control these risks when sensitive data is processed outside the enterprise or by non-employees. External providers who do not have a long-term commitment to the organisation can make mistakes and commit fraud which brings the organisation at great risk. Hence, to know how much control they have over their access, it is always best practice to ask external service providers to furnish specific information on the hiring and oversight of privileged administrators.

- *Compliance:* Cloud service providers must show their responsibility for the security and integrity of their corporate and customer data. Customers must be provided with information on the specific controls that were evaluated along with security certifications and submit to external audits. When a cloud-computing provider is unwilling or unable to provide all the mentioned measures, then it is a signal that they can be used only for trivial functions by customers.

- *Data Segregation:* Most cloud offerings store data in a shared environment, and it becomes very crucial to find out what is done to segregate data at rest. Most software-as-a-service (SaaS) vendors are offering the use of encryption for the stored data, but there are encryption accidents caused by implementation mistake that results in unexpected and exploitable failures. Evidence should be asked that the encryption implementation was designed and tested by experienced specialists. It is also important to find out if the data will be stored and backed up in the encrypted form.

- *Availability:* Critical business processes require service-level commitments, but many cloud-based offerings do not provide them. For any nontrivial IT workload, organisations should define service-level requirements and service-level agreements from the provider should be demanded. When service-level agreements are not met, it must ensure that the contract contains penalty clauses.

- *Data Location:* It is important to know where the data is hosted and in which country it is stored. Not knowing in which country the data is stored could be a concern to meet national privacy regulations. It is a good practice to know if the services provider will commit to store and process data in specific jurisdictions.

- *Recovery:* When there is an unexpected disaster, there must be a recovery process, and organisations need to know how cloud offerings will recover. The offerings are vulnerable be a failure when it does not replicate the data and application infrastructure across multiple sites. If one of the sites face any disaster, the service provider should be able to tell what would happen to the data and service. The organisation have the right to know how long it will take for a complete restoration.

- *Investigative Support:* It is very difficult to have investigation support and discovery requests for cloud services because logging and data for multiple customers may be spread across an ever-changing set of hosts and data centres. It would be considered a best practice to get a contractual commitment to support specific forms of investigation and also show evidence that the vendor has already successfully supported such activities.

9 Conclusion

Amazon offers a whole universe of products and services for application development and analytics, also to build or expand the cloud environment. A complete set of highly available services are offered by Amazon Web Services which are designed to work together to create sophisticated, scalable applications. With AWS, the specific solution needed for problems can be selected, resulting in lower capital expenditure and faster time which further aid in application performance and user experience. AWS Academy is something that is bridging the gap between industry and academia. It offers cloud computing curriculum where students or learners are equipped with skills which are needed in current fast-growing industries.

The first use case to predict the customer churn rate using Amazon Sagemaker Autopilot and the second use case to predict stock price using GluonTS library and DeepAR algorithm with two approaches are implemented. The test results show how efficient and versatile platform the Amazon Sagemaker is, for model building, training or deployment. A relevant suggestion for integrating AWS services with the modules of MSc Business Analytics are also highlighted in this report to complement the course learnings.

A Appendix A: Capstone Project Plan and Timeline

Task Mode	Task Name	Duration	Start	Finish	Predecessors
	AWS Academy Capstone Project Plan	61 days	Tue 02/06/20	Tue 25/08/20	
	Phase 1: AWS Academy Review	21 days?	Tue 02/06/20	Tue 30/06/20	
	AWS Academy Account Setup	1 day	Tue 02/06/20	Tue 02/06/20	
	AWS Academy Cloud Fundamentals Training	20 days	Wed 03/06/20	Tue 30/06/20	3
	Phase 2: Problem Statement	6 days	Wed 01/07/20	Wed 08/07/20	4
	Problem Description	2 days	Wed 01/07/20	Thu 02/07/20	
	Scope	2 days	Fri 03/07/20	Mon 06/07/20	6
	Outcome Expectations	2 days	Tue 07/07/20	Wed 08/07/20	7
	Phase 3: Project Initialisation	5 days	Thu 09/07/20	Wed 15/07/20	
	Requirement Gathering and Analysis	2 days	Thu 09/07/20	Fri 10/07/20	8
	Technology Identification	3 days	Mon 13/07/20	Wed 15/07/20	10
	Phase 4: Project Execution	9 days	Thu 16/07/20	Tue 28/07/20	
	Development Phase	7 days	Thu 16/07/20	Fri 24/07/20	
	System Setup	1 day	Thu 16/07/20	Thu 16/07/20	
	Systems Database Creation	1 day	Fri 17/07/20	Fri 17/07/20	14
	Data Scraping	2 days	Mon 20/07/20	Tue 21/07/20	15
	Final Build Development	2 days	Wed 22/07/20	Thu 23/07/20	16
	Unit Testing	1 day	Fri 24/07/20	Fri 24/07/20	17
	Changes Post Unit Testing	1 day	Tue 28/07/20	Tue 28/07/20	18
	Technical Specification Documents	5 days	Mon 20/07/20	Fri 24/07/20	19
	Phase 5: Testing	10 days	Mon 27/07/20	Fri 07/08/20	
	System Integration	3 days	Mon 27/07/20	Wed 29/07/20	
	Test Cases Preparation	2 days	Thu 30/07/20	Fri 31/07/20	22
	System Testing	2 days	Mon 03/08/20	Tue 04/08/20	23
	Regression Testing	1 day	Thu 06/08/20	Thu 06/08/20	24
	Quality Assurance Checks	1 day	Fri 07/08/20	Fri 07/08/20	25
	Phase 6: Product Roll-Out	5 days	Mon 10/08/20	Fri 14/08/20	
	Conduct Final Quality Review	1 day	Mon 10/08/20	Mon 10/08/20	
	Product Release Information Document	3 days	Wed 12/08/20	Fri 14/08/20	28
	Phase 7: AWS and MSc BA Module Integration	12 days	Mon 17/08/20	Tue 01/09/20	
	Modules Based Mappings	4 days	Fri 14/08/20	Wed 19/08/20	
	Relevant Suites and Class Setup	4 days	Thu 20/08/20	Tue 25/08/20	31
	Suggestions and Improvements	4 days	Thu 27/08/20	Tue 01/09/20	32

Fig. 28. AWS Academy Capstone Project Plan and Timeline

B Appendix B: 1st Use Case: SageMaker Autopilot - Customer Churn Prediction

Features	Description
State	the US state in which the customer resides, indicated by a two-letter abbreviation; for example, OH or NJ
Account Length	the number of days that this account has been active
Area Code	the three-digit area code of the corresponding customer's phone number
Phone	the remaining seven-digit phone number
Int'l Plan	whether the customer has an international calling plan: yes/no
VMail Plan	whether the customer has a voice mail feature: yes/no
VMail Message	presumably the average number of voice mail messages per month
Day Mins	the total number of calling minutes used during the day
Day Calls	the total number of calls placed during the day
Day Charge	the billed cost of daytime calls
Eve Mins	the billed cost for calls placed during the evening
Eve Calls	the total number of calls placed during the evening
Eve Charge	the billed cost of evening calls
Night Mins	the billed cost for calls placed during nighttime
Night Calls	the total number of calls placed during the nighttime
Night Charge	the billed cost of nighttime calls
Intl Mins	the billed cost for international calls
Intl Calls	the total number of international calls
Intl Charge	the billed cost of international calls
CustServ Calls	the number of calls placed to Customer Service
Churn?	whether the customer left the service: true/false

Table 4: Features Descriptions of the Customer Churn Dataset

Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	...	Eve Calls	Eve Charge	Night Mins	Night Calls	Night Charge	Intl Mins	Intl Calls	Intl Charge	CustServ Calls	Churn?
no	no	0	90.700000	90	15.420000	...	109	17.640000	169.400000	96	7.620000	5.600000	5	1.510000	2	False.
no	no	0	155.300000	93	26.400000	...	95	22.580000	145.700000	67	6.560000	12.400000	4	3.350000	0	False.
no	no	0	189.700000	100	32.250000	...	133	9.850000	220.600000	115	9.930000	7.400000	4	2.000000	0	False.
no	no	0	179.300000	93	30.480000	...	98	15.180000	225.200000	131	10.130000	11.500000	6	3.110000	3	False.
no	no	0	147.000000	79	24.990000	...	103	13.800000	162.900000	80	7.330000	10.500000	4	2.840000	1	False.
no	no	0	179.400000	70	30.500000	...	93	12.160000	116.300000	113	5.230000	11.200000	5	3.020000	1	False.
yes	yes	33	200.300000	75	34.050000	...	67	19.260000	198.800000	91	8.950000	12.900000	3	3.480000	2	False.
no	no	0	178.600000	83	30.360000	...	103	18.110000	198.000000	119	8.910000	10.900000	5	2.940000	1	False.
no	yes	38	213.700000	61	36.330000	...	104	21.510000	207.700000	73	9.350000	10.700000	5	2.890000	2	False.
no	no	0	208.700000	97	35.480000	...	83	23.420000	182.500000	122	8.210000	8.000000	3	2.160000	2	False.

Fig. 29. Approach 1: Customer Churn Prediction - Data Exploration notebook - Dataset Sample

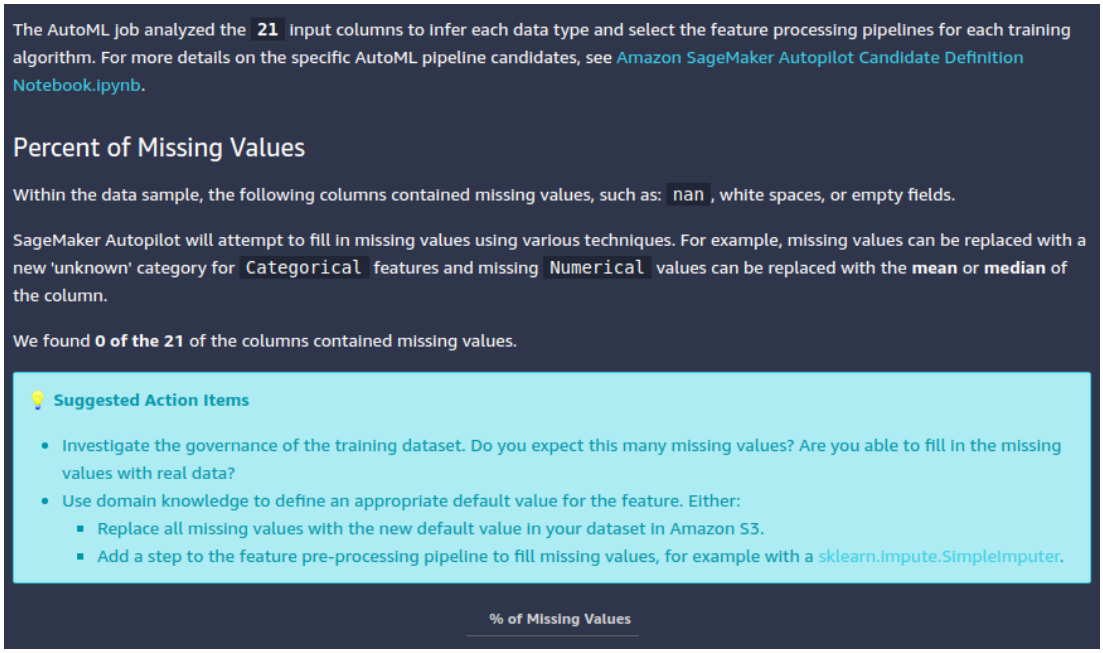

Fig. 30. Approach 1: Customer Churn Prediction - Data Exploration notebook - Column Analysis

	Number of Unique Entries	Number of Unique Words (if Text)
Churn?	2	n/a
Int'l Plan	2	n/a
VMail Plan	2	n/a
Area Code	3	n/a
CustServ Calls	10	n/a
Intl Calls	21	n/a
VMail Message	46	n/a
State	51	51
Day Calls	119	n/a
Night Calls	120	n/a
Eve Calls	123	n/a
Intl Mins	162	n/a
Intl Charge	162	n/a
Account Length	211	n/a
Night Charge	935	n/a
Eve Charge	1443	n/a
Night Mins	1594	n/a
Eve Mins	1606	n/a
Day Mins	1664	n/a
Day Charge	1668	n/a
Phone	3331	3431

Fig. 31. Approach 1: Customer Churn Prediction - Data Exploration notebook - Count Statistics

	% of Numerical Values	Mean	Median	Min	Max
Account Length	100.0%	101.065	101.0	1.0	243.0
Area Code	100.0%	437.182	415.0	408.0	510.0
VMail Message	100.0%	8.09901	0.0	0.0	51.0
Day Mins	100.0%	179.775	179.4	0.0	350.8
Day Calls	100.0%	100.436	101.0	0.0	165.0
Day Charge	100.0%	30.5623	30.5	0.0	59.64
Eve Mins	100.0%	200.98	201.4	0.0	363.7
Eve Calls	100.0%	100.114	100.0	0.0	170.0
Eve Charge	100.0%	17.0835	17.12	0.0	30.91
Night Mins	100.0%	200.872	201.2	23.2	395.0
Night Calls	100.0%	100.108	100.0	33.0	175.0
Night Charge	100.0%	9.03932	9.05	1.04	17.77
Intl Mins	100.0%	10.2373	10.3	0.0	20.0
Intl Calls	100.0%	4.47945	4.0	0.0	20.0
Intl Charge	100.0%	2.76458	2.78	0.0	5.4
CustServ Calls	100.0%	1.56286	1.0	0.0	9.0

Fig. 32. Approach 1: Customer Churn Prediction - Data Exploration notebook - Descriptive Statistics

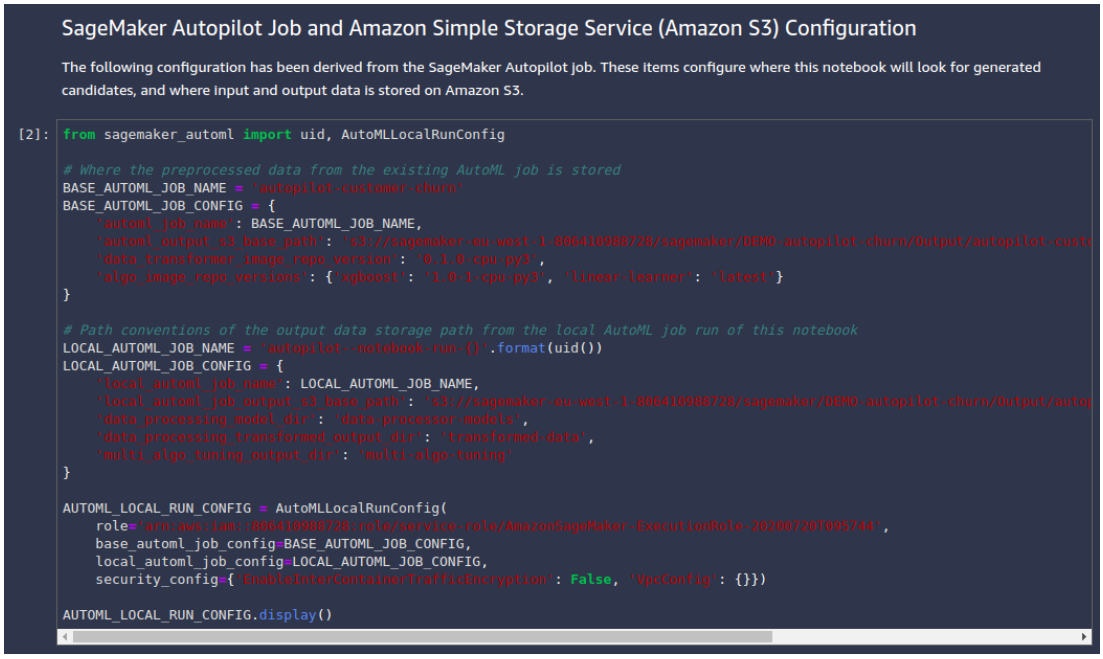

Fig. 33. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot Job and Amazon S3 Configuration

	Name	Value
General	Role	arn:aws:iam::806410988728:role/service-role/AmazonSageMaker-ExecutionRole-20200720T095744
Base AutoML Job	Job Name	autopilot-customer-churn
	Base Output S3 Path	s3://sagemaker-eu-west-1-806410988728/sagemaker/DEMO-autopilot-churn/Output/autopilot-customer-churn
Interactive Job	Job Name	autopilot--notebook-run-14-14-22-46
	Base Output S3 Path	s3://sagemaker-eu-west-1-806410988728/sagemaker/DEMO-autopilot-churn/Output/autopilot-customer-churn/autopilot--notebook-run-14-14-22-46
	Data Processing Trained Model Directory	s3://sagemaker-eu-west-1-806410988728/sagemaker/DEMO-autopilot-churn/Output/autopilot-customer-churn/autopilot--notebook-run-14-14-22-46/data-processor-models
	Data Processing Transformed Output	s3://sagemaker-eu-west-1-806410988728/sagemaker/DEMO-autopilot-churn/Output/autopilot-customer-churn/autopilot--notebook-run-14-14-22-46/transformed-data
	Algo Tuning Model Output Directory	s3://sagemaker-eu-west-1-806410988728/sagemaker/DEMO-autopilot-churn/Output/autopilot-customer-churn/autopilot--notebook-run-14-14-22-46/multi-algo-tuning

Fig. 34. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Configuration Results

Selected Candidates

You have selected the following candidates (please run the cell below and click on the feature transformer links for details):

```
[14]: automl_interactive_runner.display_candidates()
```

Candidate Name	Algorithm	Feature Transformer
dpp0-xgboost	xgboost	dpp0.py
dpp1-xgboost	xgboost	dpp1.py
dpp2-linear-learner	linear-learner	dpp2.py
dpp3-xgboost	xgboost	dpp3.py
dpp4-xgboost	xgboost	dpp4.py
dpp5-xgboost	xgboost	dpp5.py
dpp6-xgboost	xgboost	dpp6.py
dpp7-xgboost	xgboost	dpp7.py
dpp8-xgboost	xgboost	dpp8.py
dpp9-xgboost	xgboost	dpp9.py

Fig. 35. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate List

dpp0-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer (converts missing values to nan), 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

```
[4]: automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp0",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": True
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 36. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 1

dpp1-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer, 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustPCA followed by RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

[5]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp1",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "text/csv",
        "sparse_encoding": False
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 37. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 2

dpp2-linear-learner: This data transformation strategy first transforms 'numeric' features using combined RobustImputer and RobustMissingIndicator followed by QuantileExtremeValuesTransformer, 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustPCA followed by RobustStandardScaler. The transformed data will be used to tune a *linear-learner* model. Here is the definition:

[6]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp2",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": False
    },
    "algorithm": {
        "name": "linear-learner",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

```
2020-08-14 14:23:08,588 WARNING sagemaker.amazon.amazon_estimator: 'get_image_uri' method will be deprecated in favor of 'Ima
geURIProvider' class in SageMaker Python SDK v2.
```

Fig. 38. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 3

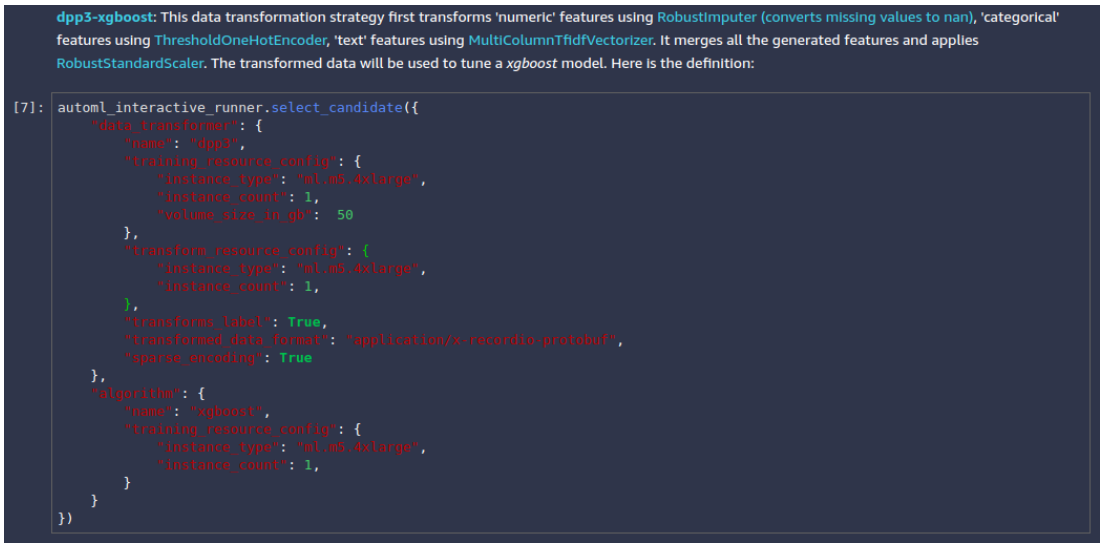

Fig. 39. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 4

Fig. 40. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 5

dpp5-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer (converts missing values to nan), 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

[9]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp5",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": True
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 41. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 6

dpp6-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer (converts missing values to nan), 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

[10]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp6",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": True
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 42. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 7

dpp7-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer (converts missing values to nan), 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

[11]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp7",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": True
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m6.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 43. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 8

dpp8-xgboost: This data transformation strategy first transforms 'numeric' features using RobustImputer (converts missing values to nan), 'categorical' features using ThresholdOneHotEncoder, 'text' features using MultiColumnTfidfVectorizer. It merges all the generated features and applies RobustStandardScaler. The transformed data will be used to tune a *xgboost* model. Here is the definition:

[12]:
```
automl_interactive_runner.select_candidate({
    "data_transformer": {
        "name": "dpp8",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
            "volume_size_in_gb":  50
        },
        "transform_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        },
        "transforms_label": True,
        "transformed_data_format": "application/x-recordio-protobuf",
        "sparse_encoding": True
    },
    "algorithm": {
        "name": "xgboost",
        "training_resource_config": {
            "instance_type": "ml.m5.4xlarge",
            "instance_count": 1,
        }
    }
})
```

Fig. 44. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 9

Fig. 45. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Candidate 10

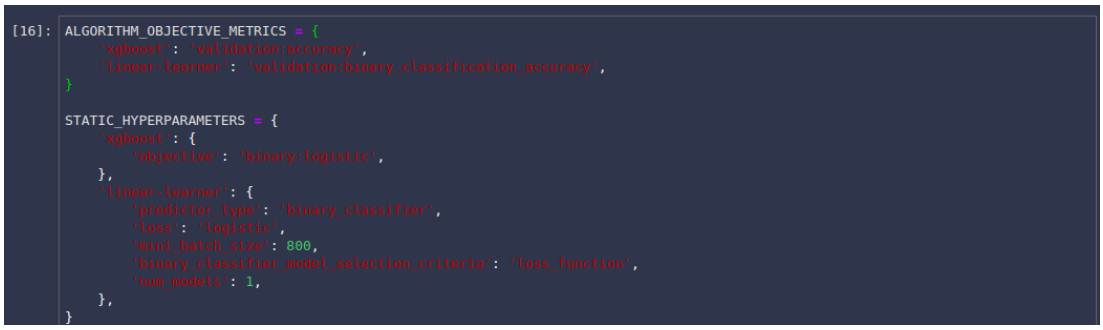

Fig. 46. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot recommendation of hyperparameters, objectives and accuracy metrics for the algorithm and problem type

Fig. 47. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Autopilot recommendation of tunable hyperparameters search ranges

```
[20]: from sagemaker.tuner import HyperparameterTuner

      base_tuning_job_name = "{}-tuning".format(AUTOML_LOCAL_RUN_CONFIG.local_automl_job_name)

      tuner = HyperparameterTuner.create(
          base_tuning_job_name=base_tuning_job_name,
          strategy='Bayesian',
          objective_type='Maximize',
          max_parallel_jobs=2,
          max_jobs=250,
          **multi_algo_tuning_parameters,
      )
```

Fig. 48. Approach 1: Customer Churn Prediction - Candidate Generation notebook - A multi-algorithm model tuning job across the different possible algorithms and feature processing strategies.

```
[ ]: from pprint import pprint
     from sagemaker.analytics import HyperparameterTuningJobAnalytics

     SAGEMAKER_SESSION = AUTOML_LOCAL_RUN_CONFIG.sagemaker_session
     SAGEMAKER_ROLE = AUTOML_LOCAL_RUN_CONFIG.role

     tuner_analytics = HyperparameterTuningJobAnalytics(
         tuner.latest_tuning_job.name, sagemaker_session=SAGEMAKER_SESSION)

     df_tuning_job_analytics = tuner_analytics.dataframe()

     # Sort the tuning job analytics by the final metrics value
     df_tuning_job_analytics.sort_values(
         by=['FinalObjectiveValue'],
         inplace=True,
         ascending=False if tuner.objective_type == "Maximize" else True)

     # Show detailed analytics for the top 20 models
     df_tuning_job_analytics.head(20)

[ ]: attached_tuner = HyperparameterTuner.attach(tuner.latest_tuning_job.name, sagemaker_session=SAGEMAKER_SESSION)
     best_training_job = attached_tuner.best_training_job()

     print("Best Multi Algorithm HPO training job name is {}".format(best_training_job))
```

Fig. 49. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Hyperparameter tuning result

```
[ ]: from sagemaker.estimator import Estimator
     from sagemaker import PipelineModel
     from sagemaker_automl import select_inference_output

     # Get a data transformation model from chosen candidate
     best_candidate = automl_interactive_runner.choose_candidate(df_tuning_job_analytics, best_training_job)
     best_data_transformer_model = best_candidate.get_data_transformer_model(role=SAGEMAKER_ROLE, sagemaker_session=SAGEMAKER_

     # Our first data transformation container will always return recordio-protobuf format
     best_data_transformer_model.env["SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT"] = 'application/x-recordio-protobuf'
     # Add environment variable for sparse encoding
     if best_candidate.data_transformer_step.sparse_encoding:
         best_data_transformer_model.env["AUTOML_SPARSE_ENCODE_RECORDIO_PROTOBUF"] = '1'

     # Get a algo model from chosen training job of the candidate
     algo_estimator = Estimator.attach(best_training_job)
     best_algo_model = algo_estimator.create_model(env={'SAGEMAKER_DEFAULT_INVOCATIONS_ACCEPT':"text/csv"})

     # Final pipeline model is composed of data transformation models and algo model and an
     # inverse label transform model if we need to transform the intermediates back to non-numerical value
     model_containers = [best_data_transformer_model, best_algo_model]
     if best_candidate.transforms_label:
         model_containers.append(best_candidate.get_data_transformer_model(
             transform_mode="inverse-label-transform",
             role=SAGEMAKER_ROLE,
             sagemaker_session=SAGEMAKER_SESSION))

     # This model can emit response ['predicted_label', 'probability', 'labels', 'probabilities']. To enable the model to emit
     # of the response content, pass the keys to `output_key` keyword argument in the select_inference_output method.

     model_containers = select_inference_output("BinaryClassification", model_containers, output_keys=['predicted_label'])


     pipeline_model = PipelineModel(
         name="AutoML-{}".format(AUTOML_LOCAL_RUN_CONFIG.local_automl_job_name),
         role=SAGEMAKER_ROLE,
         models=model_containers,
         vpc_config=AUTOML_LOCAL_RUN_CONFIG.vpc_config)
```

Fig. 50. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Build the best model, by linking the best training job with feature engineering pipelines

```
[ ]: pipeline_model.deploy(initial_instance_count=1,
                          instance_type='ml.m5.2xlarge',
                          endpoint_name=pipeline_model.name,
                          wait=True)
```

Fig. 51. Approach 1: Customer Churn Prediction - Candidate Generation notebook - Deploy the best model, create an endpoint for production

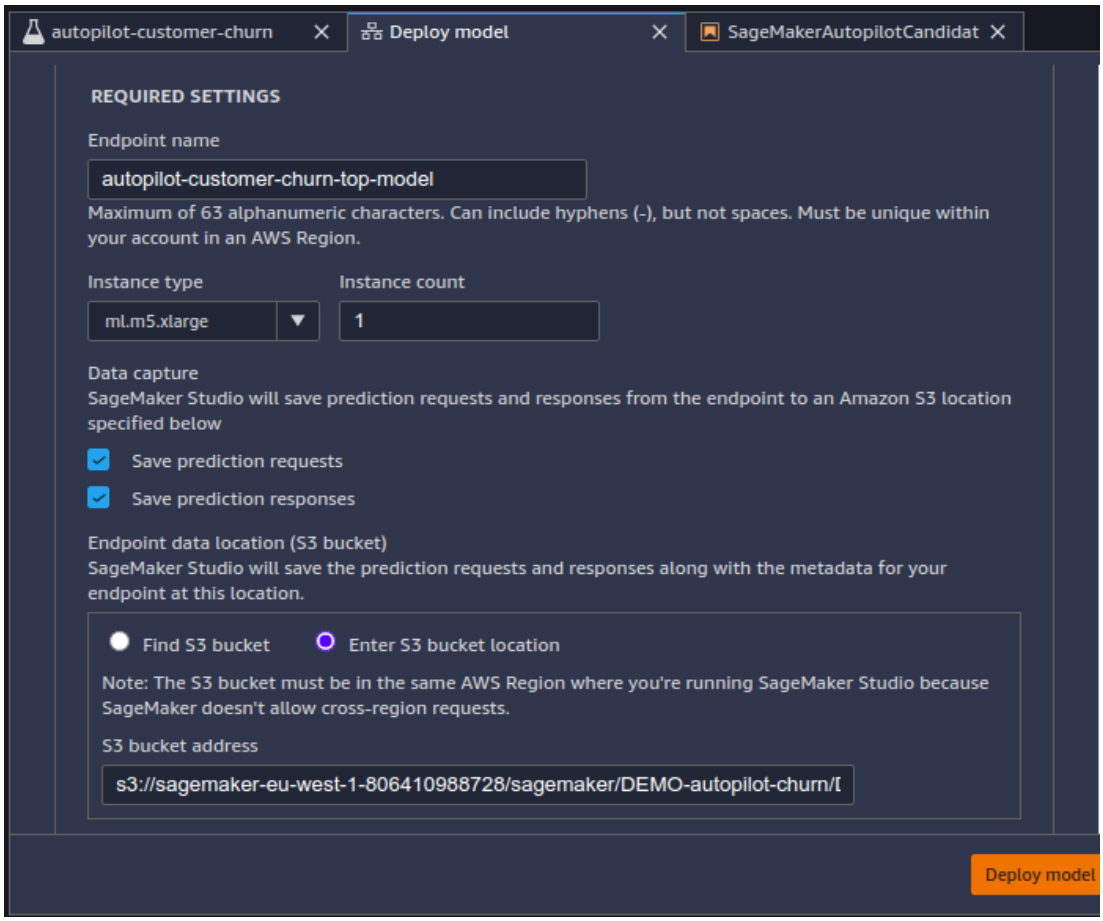

Fig. 52. Create an endpoint for production/testing

```
      AutoMLJobName: autopilot-churn-notebook
[7]: {'AutoMLJobArn': 'arn:aws:sagemaker:eu-west-1:806410988728:automl-job/autopilot-churn-notebook',
      'ResponseMetadata': {'RequestId': '4dbfe720-b3c1-45ba-983f-c74b2a2bc86e',
       'HTTPStatusCode': 200,
       'HTTPHeaders': {'x-amzn-requestid': '4dbfe720-b3c1-45ba-983f-c74b2a2bc86e',
        'content-type': 'application/x-amz-json-1.1',
        'content-length': '95',
        'date': 'Sat, 15 Aug 2020 10:31:29 GMT'},
       'RetryAttempts': 0}}
```

Fig. 53. Approach 2: Customer Churn Prediction - Autopilot Jobs Setup Logs

```
------------------------------

CandidateName: tuning-job-1-894566e499d642a091-004-e3bde875

------------------------------

FinalAutoMLJobObjectiveMetricName: validation:accuracy

------------------------------

FinalAutoMLJobObjectiveMetricValue: 0.9585700035095215
```

Fig. 54. Approach 2: Customer Churn Prediction - The best model. ObjectiveMetric = validation:accuracy

C Appendix C: 2nd Use Case: SageMaker SDK - Stock Price Prediction

Features	Description
ISIN	ISIN of the security
Mnemonic	Stock exchange ticker symbol
SecurityDesc	Description of the security
SecurityType	Type of security
Currency	Currency in which the product is traded
SecurityID	Unique identifier for each contract
Date	Date of trading period
Time	Minute of trading to which this entry relates
StartPrice	Trading price at the start of period
MaxPrice	Maximum price over the period
MinPrice	Minimum price over the period
EndPrice	Trading price at the end of the period
TradedVolume	Total value traded
NumberOfTrades	Number of distinct trades during the period

Table 5: Features Descriptions of the Deutsche Börse Public XETRA Dataset

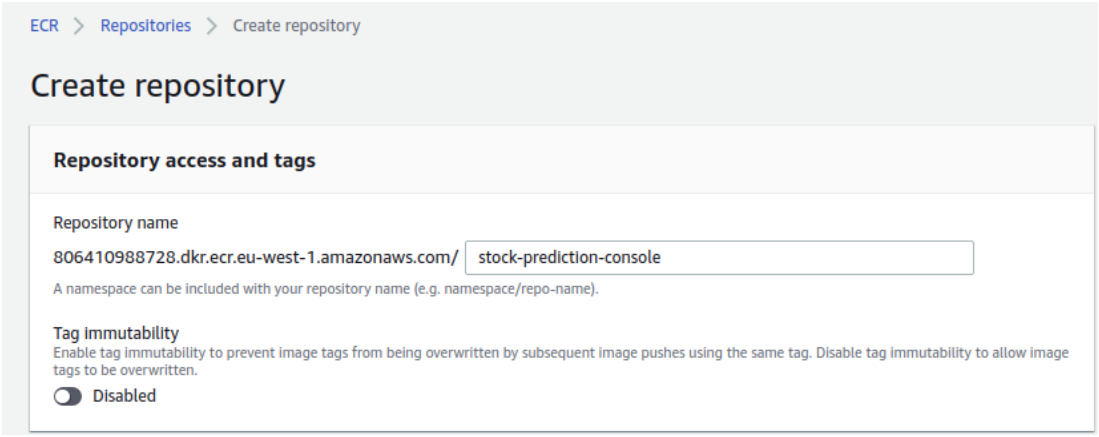

Fig. 55. Stock Price Prediction - Approach 1: Create a new container repository from ECR

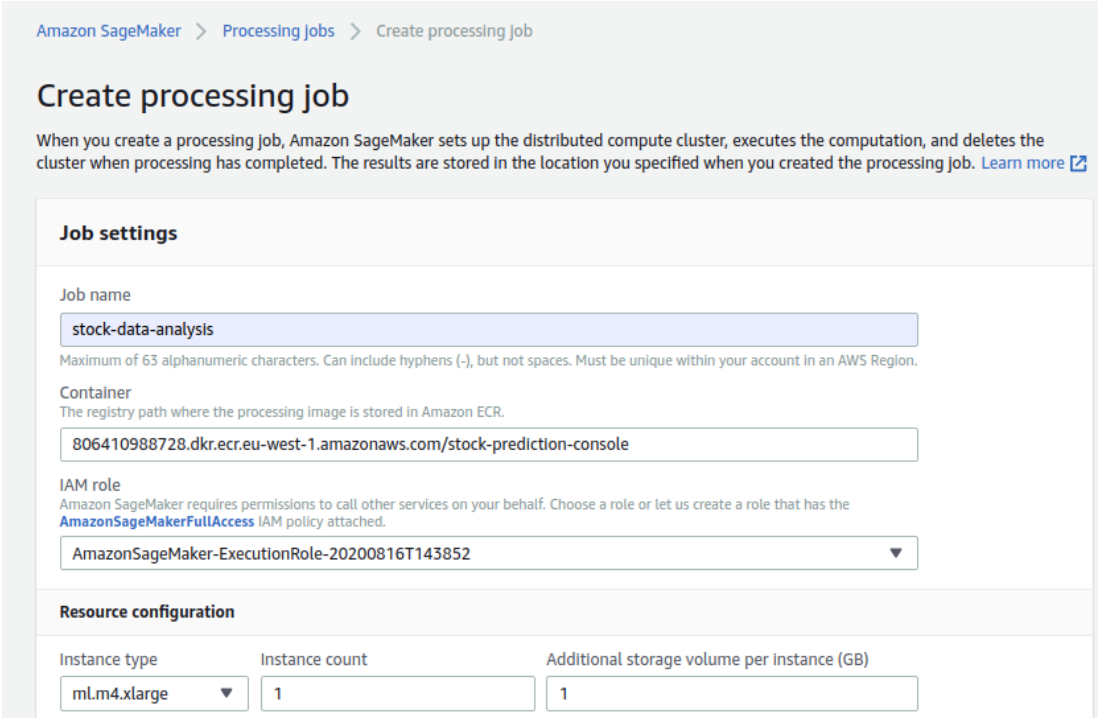

Fig. 56. Stock Price Prediction - Approach 1: Create new processing job

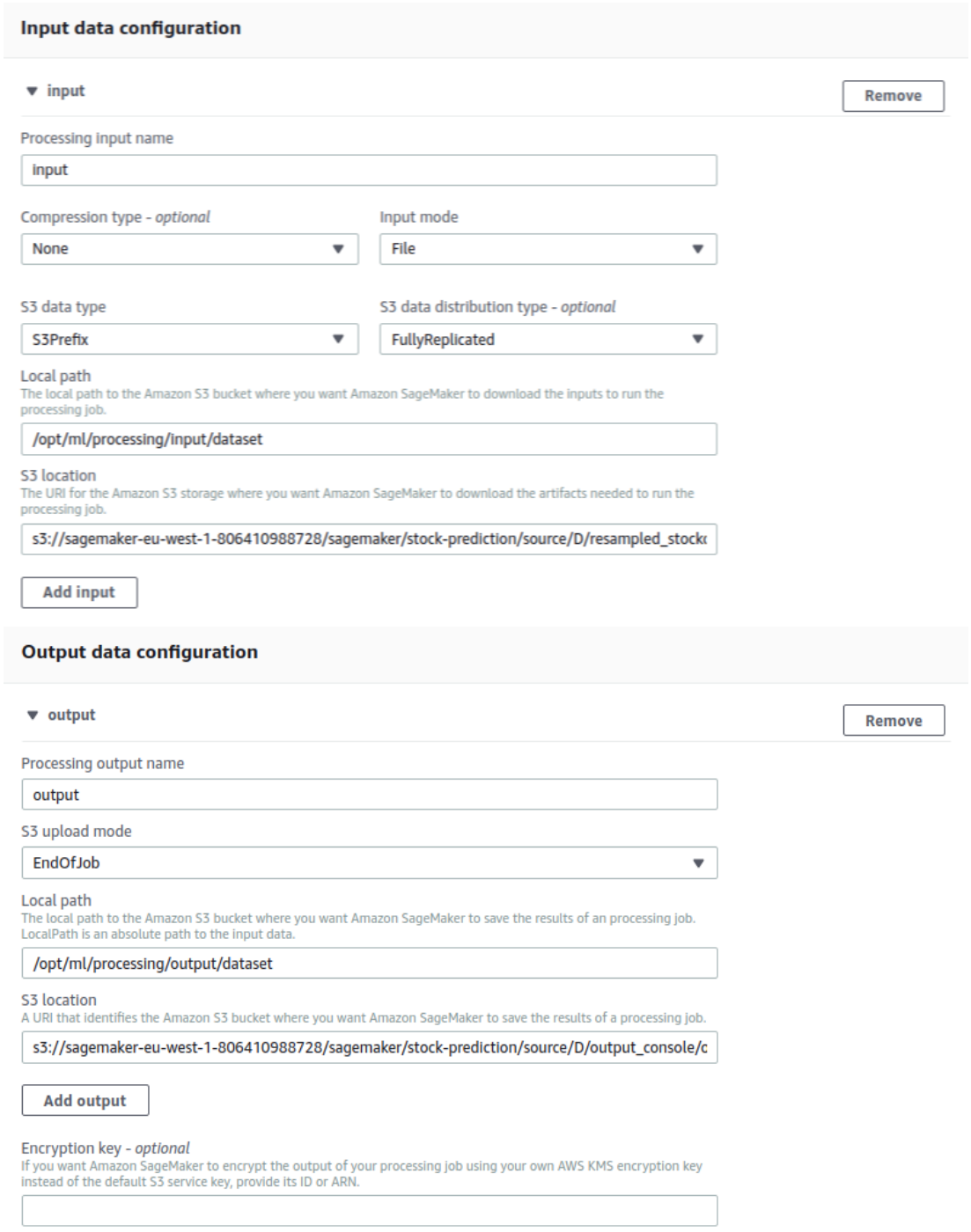

Fig. 57. Stock Price Prediction - Approach 1: Provide local and S3 paths to input data and processing artifact

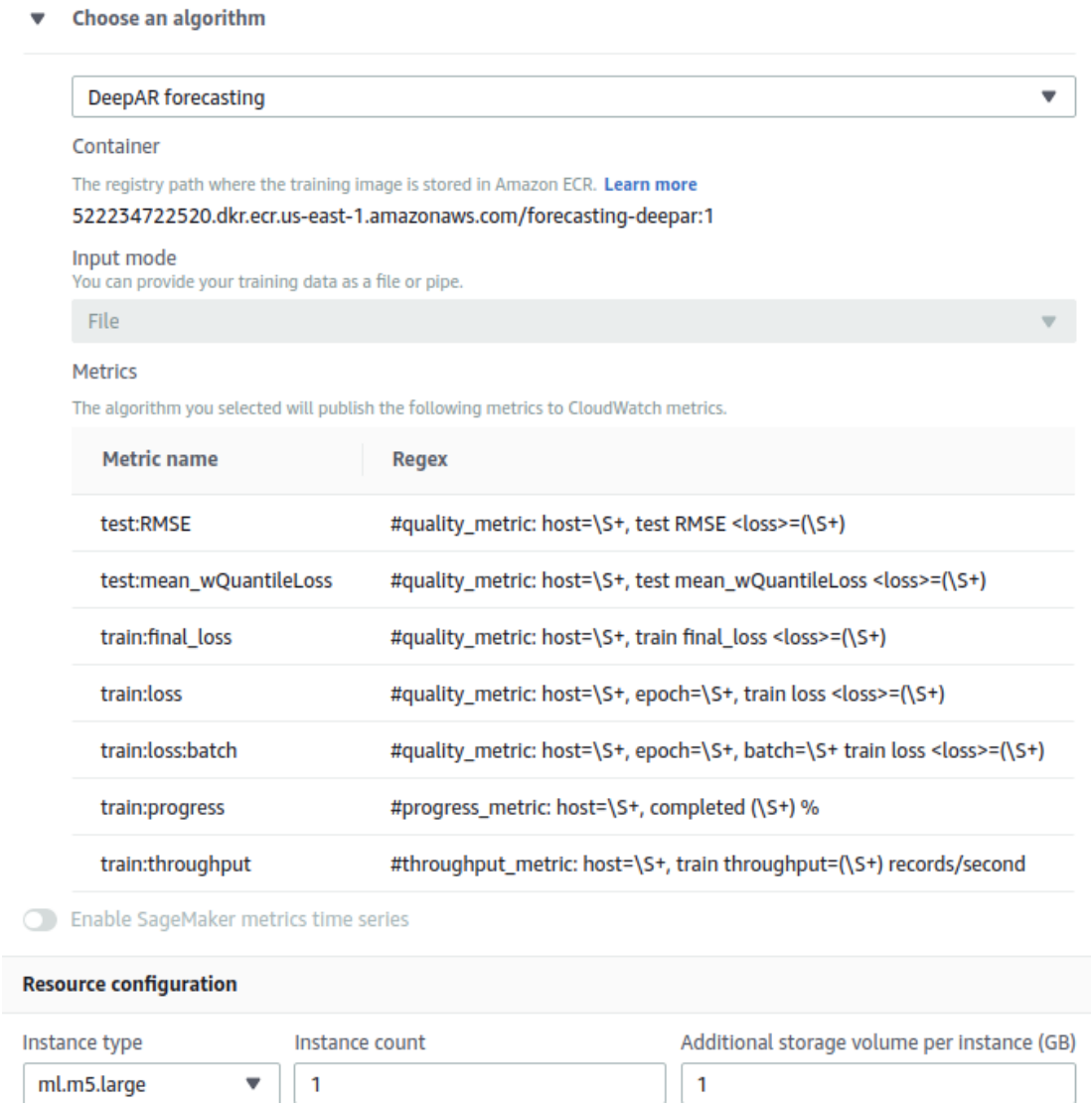

Fig. 58. Stock Price Prediction - Approach 1: Choose an AWS built-in algorithm

Fig. 59. Stock Price Prediction - Approach 1: Choose the appropriate hyperparameters

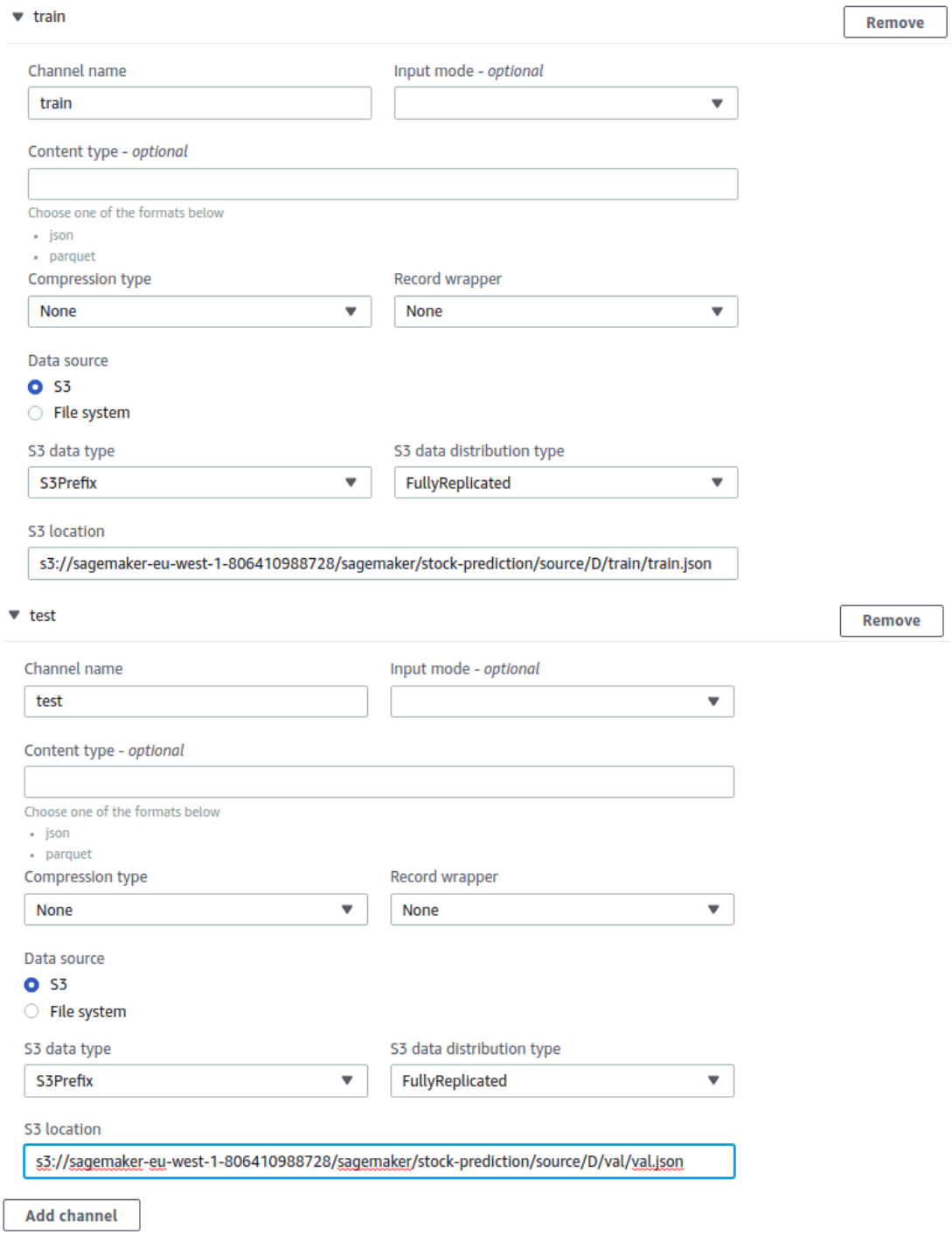

Fig. 60. Stock Price Prediction - Approach 1: Create two channels for training set and validation set

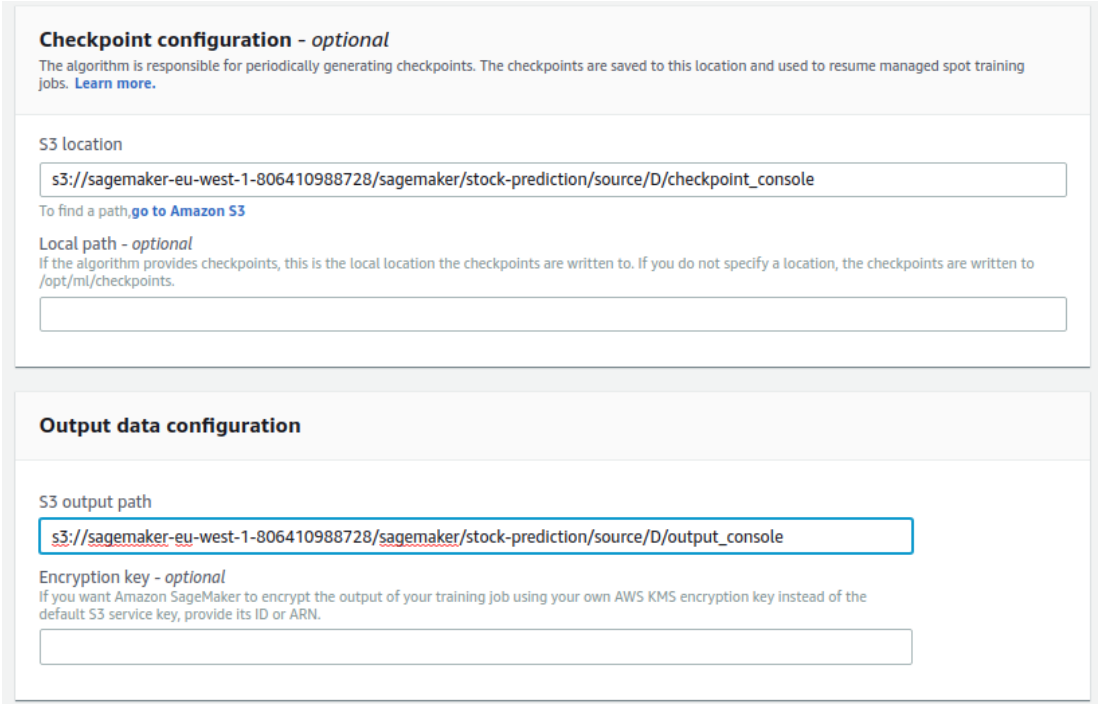

Fig. 61. Stock Price Prediction - Approach 1: Create folder for checkpoint and output data in s3 bucket

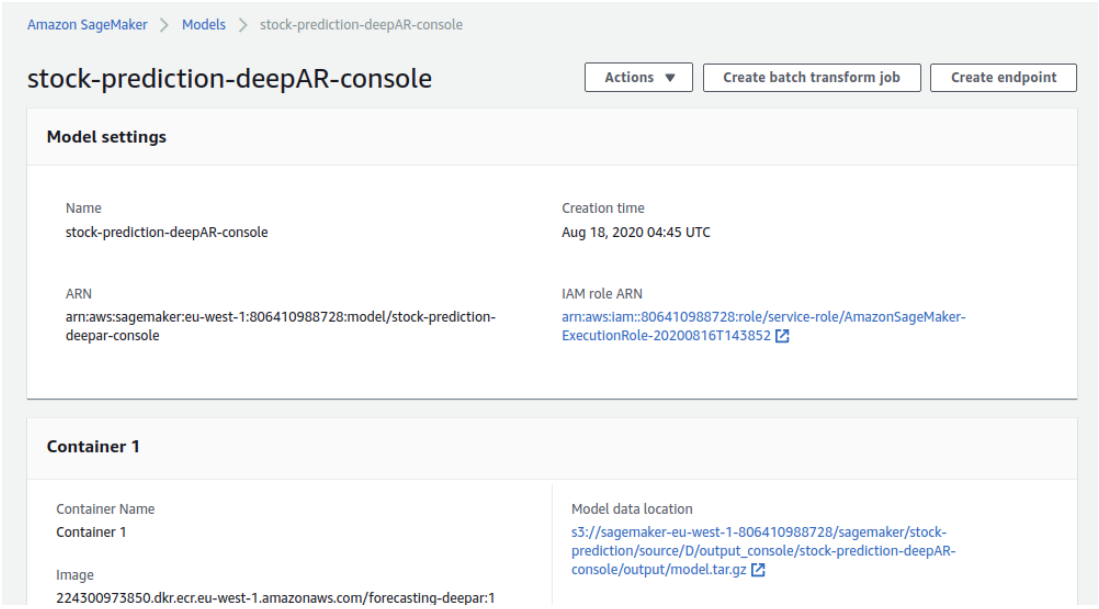

Fig. 62. Stock Price Prediction - Approach 1: Create a new model with the completed training job

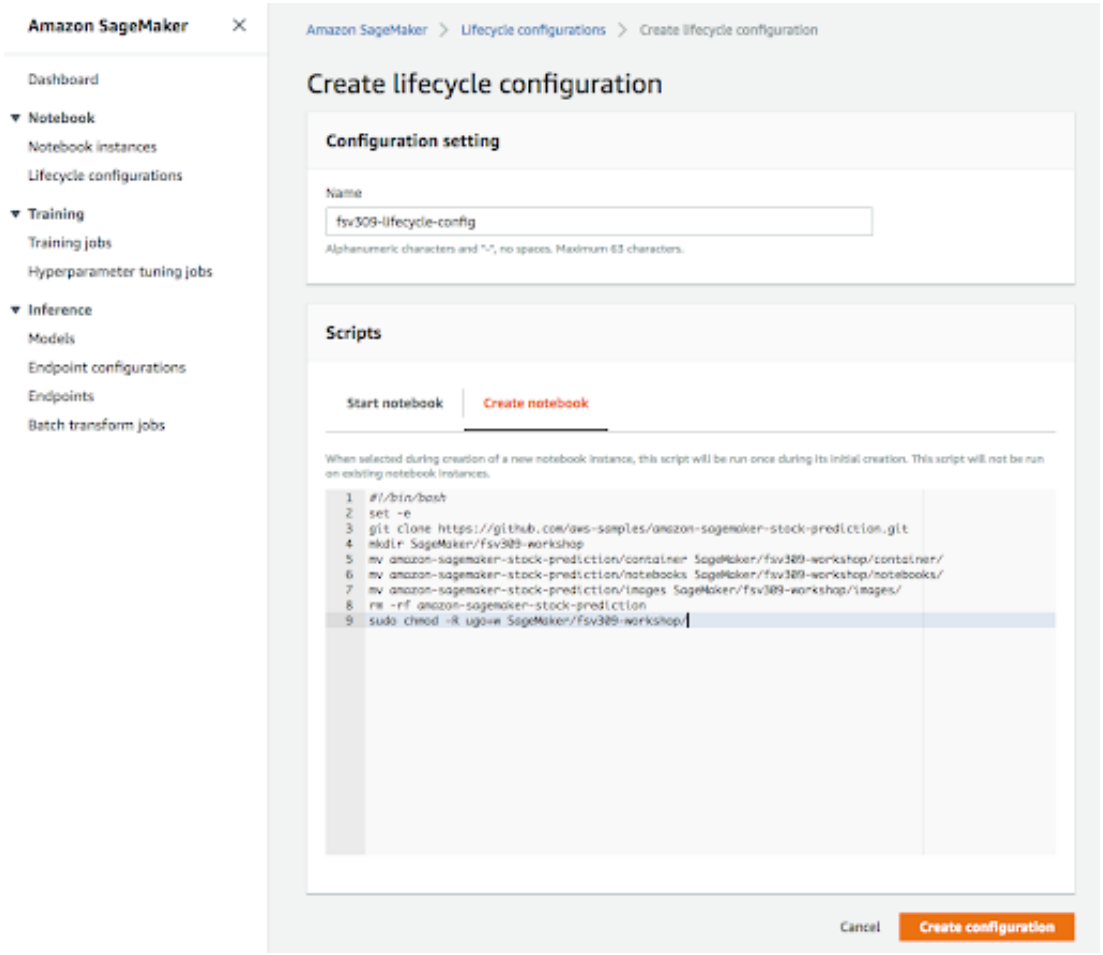

Fig. 63. Stock Price Prediction - Approach 2: Custom life-cycle for new Notebook Instance

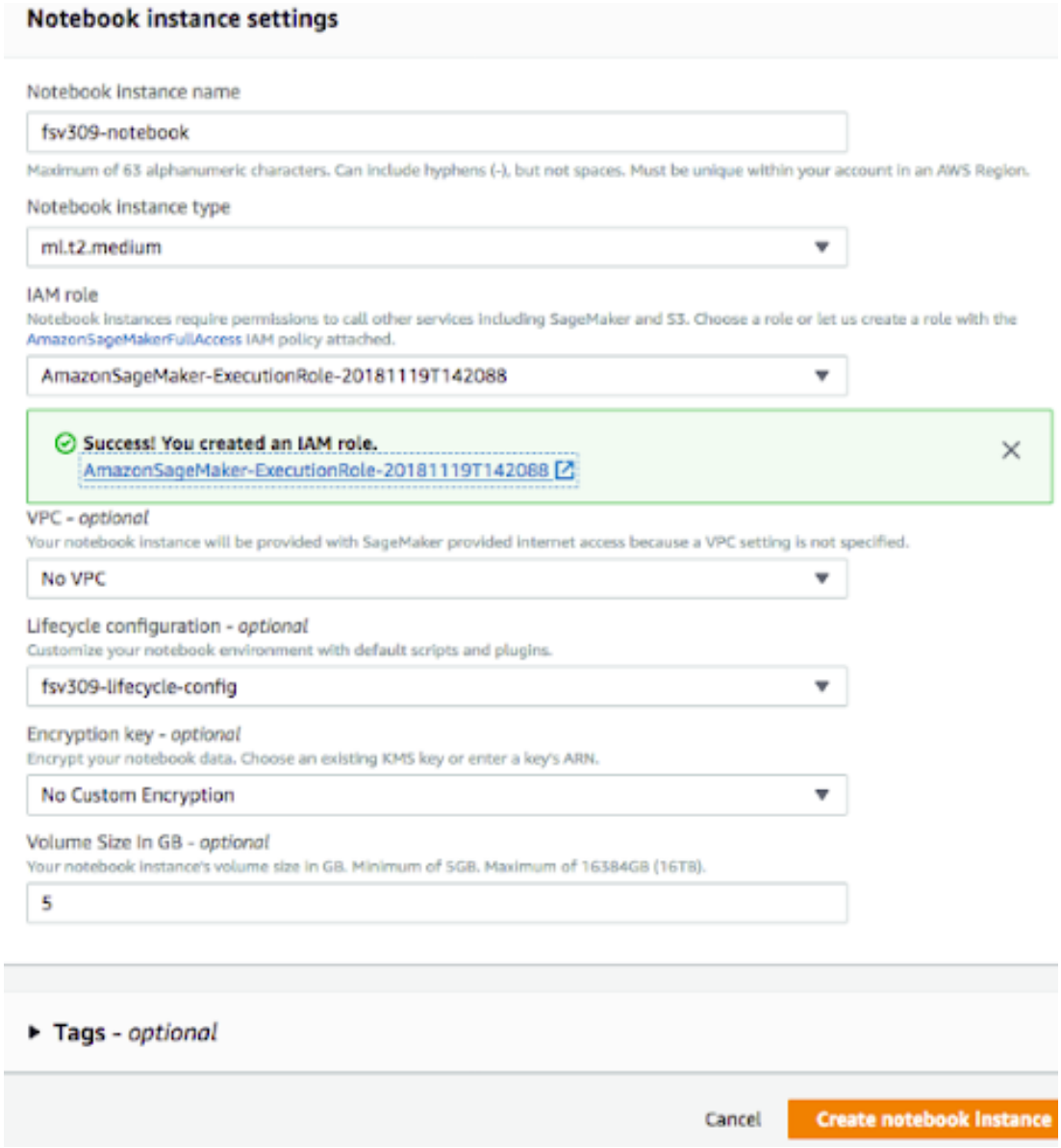

Fig. 64. Stock Price Prediction - Approach 2: Create a new Notebook Instance

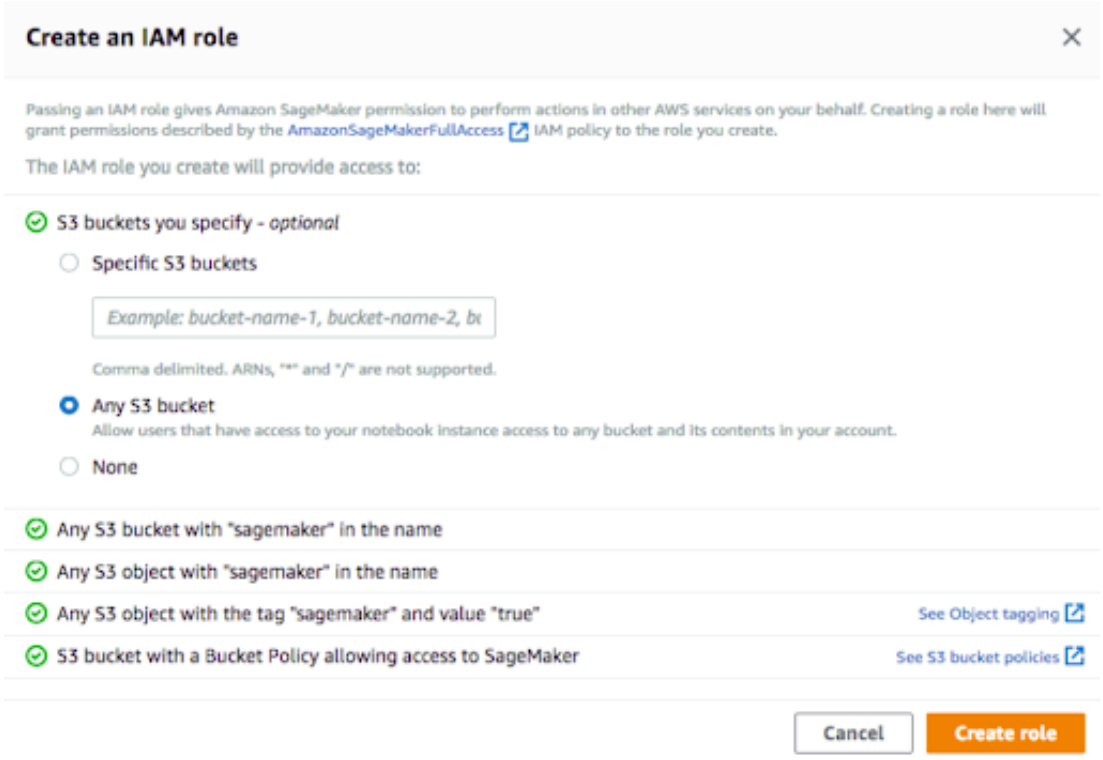

Fig. 65. Stock Price Prediction - Approach 2: Create new IAM role

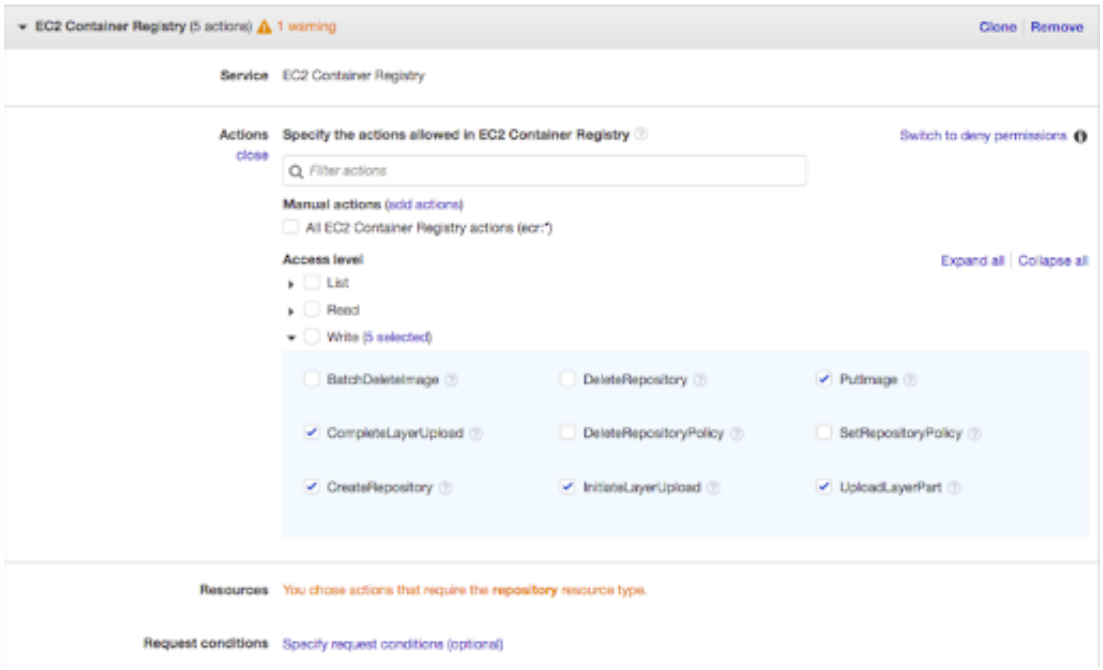

Fig. 66. Stock Price Prediction - Approach 2: Custom policy for IAM role

ISIN	Mnemonic	SecurityDesc	SecurityType	Currency	SecurityID	Date	Time	StartPrice	MaxPrice	MinPrice	EndPrice	TradedVolume	NumberOfTrades	CalcTime	CalcDateTime
IG001	BWO	BUWOG AG	Common stock	EUR	2504156	2017-07-01	08:29	25.090	25.090	25.090	25.090	0	1	1900-01-01 08:29:00	2017-07-01 08:29:00
SCH9	BZ6	KAPSCH TRAFFICCOM AG	Common stock	EUR	2504157	2017-07-01	08:29	46.715	46.715	46.715	46.715	0	1	1900-01-01 08:29:00	2017-07-01 08:29:00
OST4	O3P	OESTERREICH. POST AG	Common stock	EUR	2504158	2017-07-01	08:29	38.095	38.095	38.095	38.095	0	1	1900-01-01 08:29:00	2017-07-01 08:29:00
E9W5	SANT	S+T AG (Z.REG.MK.Z.)O.N.	Common stock	EUR	2504159	2017-07-01	08:29	13.175	13.175	13.175	13.175	0	1	1900-01-01 08:29:00	2017-07-01 08:29:00
I0XX9	P4N	POLYTEC HLDG AG INH. EO 1	Common stock	EUR	2504160	2017-07-01	08:29	16.590	16.590	16.590	16.590	0	1	1900-01-01 08:29:00	2017-07-01 08:29:00

Fig. 67. Stock Price Prediction - Approach 2: Unpreprocessed Data

Fig. 68. Stock Price Prediction - Approach 2: Data Exploration before Feature Engineering

CalcDateTime	BMW-EndPrice	BMW-MaxPrice	BMW-MinPrice	BMW-StartPrice	CON-EndPrice	CON-MaxPrice	CON-MinPrice	CON-StartPrice	DAI-EndPrice	DAI-MaxPrice	DAI-MinPrice	DAI-StartPrice	PAH3-EndPrice	PAH3-MaxPrice	PAH3-MinPrice	PAH3-StartPrice	Er
2017-07-03	83.12	83.52	81.86	81.96	190.30	191.55	189.55	189.80	64.14	64.48	63.60	63.68	50.66	50.73	49.62	49.69	
2017-07-04	82.53	83.47	82.34	83.41	190.30	190.65	188.70	190.65	64.19	64.46	63.89	64.45	50.78	51.00	50.42	51.00	
2017-07-05	81.48	81.84	81.19	81.76	189.20	189.95	188.65	189.75	63.54	63.89	63.25	63.88	50.37	50.77	50.10	50.71	
2017-07-06	81.34	81.63	80.50	81.26	189.85	190.10	187.50	189.45	63.36	64.04	63.09	63.80	51.31	51.45	50.66	50.96	
2017-07-07	81.26	81.35	80.83	81.19	188.90	189.15	187.20	188.70	63.44	63.57	63.13	63.21	51.17	51.26	50.80	51.09	

Fig. 69. Stock Price Prediction - Approach 2: Transformed Data for DeepAR algorithm

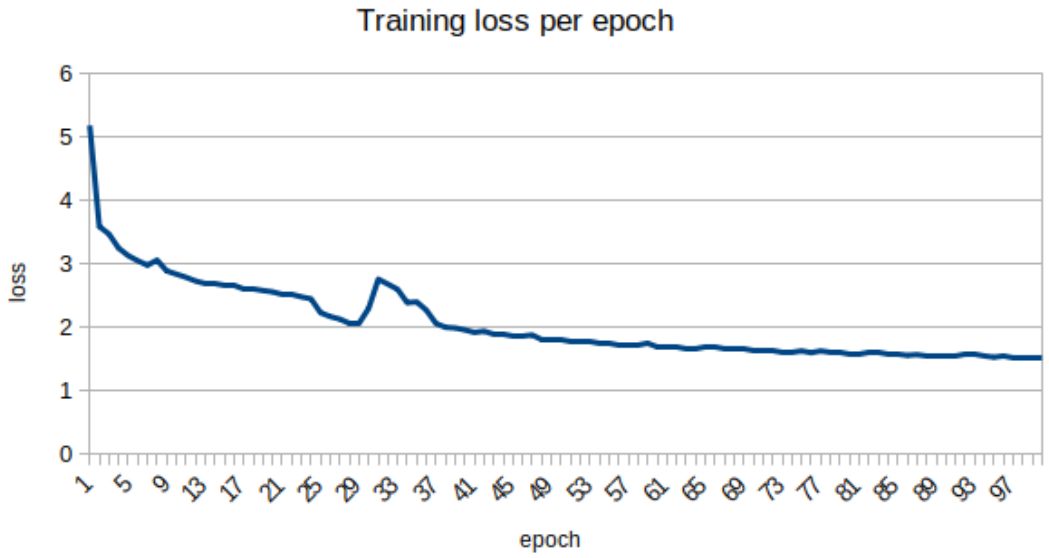

Fig. 70. Stock Price Prediction - Approach 2: Training loss per epoch

References

A Vouk, M. (2008). 'Cloud computing–issues, research and implementations'. *Journal of computing and information technology*, *16*(4), pp. 235–246.

Academy, A. (2020). *'AWS Academy Cloud Foundations', Available at:* https://awsacademy.instructure.com/courses/230

Adrian, M., Feinberg, D., & Cook, H. (2019). 'Magic quadrant for operational database management systems'. *Gartner Magic Quadrant*.

Amazon SageMaker Developer guide, . (2020). *'DeepAR Forecasting Algorithm', Available at:* https://docs.aws.amazon.com/sagemaker/latest/dg/deepar.html

AWS. (2020a). *'Amazon SageMaker', Available at:* https://aws.amazon.com/sagemaker/

AWS. (2020b). *'Deploy a Model in Amazon SageMaker', Available at:* https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-deployment.html

AWS. (2020c). *'Machine Learning with Amazon SageMaker', Available at:* https://docs.aws.amazon.com/sagemaker/latest/dg/how-it-works-mlconcepts.html

AWS. (2020d). *'SageMaker Python SDK', Available at:* https://github.com/aws/sagemaker-python-sdk

AWS, A. S. E. (2020). *'Customer Churn Prediction with Amazon SageMaker Autopilot', Available at:* https://github.com/awslabs/amazon-sagemaker-examples/blob/master/autopilot/autopilot_customer_churn.ipynb

AWS, A. S. T. (2020). *'Create a machine learning model automatically with Amazon SageMaker Autopilot', Available at:* https://aws.amazon.com/getting-started/hands-on/create-machine-learning-model-automatically-sagemaker-autopilot/

AWS, A. S. S. S. C. U. G. (2019). *'How do I create an S3 Bucket?', Available at:* https://docs.aws.amazon.com/AmazonS3/latest/user-guide/create-bucket.html

Baker, V., Elliot, B., Sicular, S., Mullen, A., & Brethenoux, E. (2020). 'Magic quadrant for cloud ai developer services'. *Gartner Magic Quadrant*.

Bala, R., Gill, B., Smith, D., & Wright, D. (2019). 'Magic quadrant for cloud infrastructure as a service, worldwide'. *Gartner Magic Quadrant*.

Bayramusta, M., & Nasir, V. A. (2016). 'A fad or future of it?: A comprehensive literature review on the cloud computing research'. *International Journal of Information Management*, *36*(4), pp. 635–644.

Binoy. (2018). *'Stock Prediction using Neural Network on Amazon SageMaker',Available at:* https://github.com/aws-samples/amazon-sagemaker-stock-prediction

CAF, A. (2020). *'AWS Cloud Adoption Framework', Available at:* https://aws.amazon.com/professional-services/CAF/

Correa, R. F. (2019). *'Building an AWS Serverless ML Pipeline with Step Functions, OLX Group Engineering', Available at:* https://tech.olx.com/building-an-aws-serverless-ml-pipeline-with-step-functions-b39feed12bab

Deelman, E., Singh, G., Livny, M., Berriman, B., & Good, J. (2008). 'The cost of doing science on the cloud: The montage example', In *Sc'08: Proceedings of the 2008 acm/ieee conference on supercomputing*. Ieee.

Dikaiakos, M. D., Katsaros, D., Mehra, P., Pallis, G., & Vakali, A. (2009). 'Cloud computing: Distributed internet computing for it and scientific research'. *IEEE Internet computing*, *13*(5), pp. 10–13.

El-Gazzar, R. F. (2014). 'A literature review on cloud computing adoption issues in enterprises', In *International working conference on transfer and diffusion of it*. Springer.

Foote, K. D. (2017). 'A brief history of cloud computing'. *Data Topics*.

Garcıa, Á. L., De Lucas, J. M., Antonacci, M., Zu Castell, W., David, M., Hardt, M., Iglesias, L. L., Moltó, G., Plociennik, M., Tran, V., Et al. (2020). 'A cloud-based framework for machine learning workloads and applications'. *IEEE access*, *8*, pp. 18681–18692.

GLUON. (2020). *'Extended Forecasting Tutorial', Available at:* https://gluon-ts.mxnet.io/examples/extended_forecasting_tutorial/extended_tutorial.html

gluon-ts GitHub. (2020). *'Bayesian Optimization for parameter tuning', Available at:* https://github.com/awslabs/gluon-ts/issues/637

Goli, S. (2020). 'Introducing new tools and features as demand for online learning grows'.

Guide, A. S. D. (2020). *'Use Amazon SageMaker built-in algorithms', Available at:* https://docs.aws.amazon.com/sagemaker/latest/dg/algos.html

Heiser, J., & Nicolett, M. (2008). 'Assessing the security risks of cloud computing'. *Gartner report*, *27*, pp. 29–52.

Illia, Y. (2020). *'Machine Learning Tools: AWS vs. Azure vs. Google Cloud', Available at:* https://www.ssa-data.com/blog/archive/top-machine-learning-tools-comparison/

Jain, N. (2020). *'Comparing Machine Learning as a Service: Amazon, Microsoft Azure, Google Cloud AI', Available at:* https://www.whizlabs.com/blog/comparing-machine-learning-as-a-service/#:~:text=GCP%5C%20offers%5C%20its%5C%20machine%5C%20learning,ideal%5C%20for%5C%20fluent%5C%20data%5C%20specialists.

Janakiram. (2019). *'An Executive's Guide To Understanding Cloud-based Machine Learning Services', Available at:* https://www.forbes.com/sites/janakirammsv/2019/01/01/an-executives-guide-to-understanding-cloud-based-machine-learning-services/#3b7a40163e3e

Jasper. (2020). *'M5 Forecasting Competition GluonTS Template', Available at:* https://github.com/awslabs/gluon-ts/blob/master/examples/m5_gluonts_template.ipynb

Jouganous, J., Savidan, R., & Bellec, A. (2018). *'From mathematical formalism to a description of available solutions (+ Kaggle benchmark)', Available at:* https://hackernoon.com/a-brief-overview-of-automatic-machine-learning-solutions-automl-2826c7807a2a

Kepes, B. (2015). *'Amazon Web Services Decimates All Comers–Bigger Base, Faster Growth, More Innovation'*, Available at: https://www.forbes.com/sites/benkepes/2015/05/20/amazon-web-services-decimates-all-comers-bigger-base-faster-growth-more-innovation/#12c039d2383c

Leong, L., Petri, G., Gill, B., & Dorosh, M. (2017). 'Magic quadrant for cloud infrastructure as a service, worldwide'. *Gartner Rep. G*, *278620*, pp. 2016.

Linthicum, D. S. (2009). *'Cloud computing and soa convergence in your enterprise: A step-by-step guide'*. Pearson Education.

Mansfield-Devine, S. (2008). 'Danger in the clouds'. *Network security*, *2008*(12), pp. 9–11.

Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). 'Cloud computing—the business perspective'. *Decision support systems*, *51*(1), pp. 176–189.

Mell, P., Grance, T. Et al. (2011). 'The nist definition of cloud computing'.

Namboori, R. (2014). 'The evolution of cloud computing'. *Cloud Zone*.

Paredes-Gualtor, J., Moscoso-Zea, O., Saa, P., Sandoval, F., & Rodas, P. (2017). 'Unified cloud computing adoption framework', In *2017 international conference on information systems and computer science (inciscos)*. IEEE.

Pearson. (2020). *'AWS Certification', Available at:* https://home.pearsonvue.com/Clients/Amazon-Web-Services.aspx

Piali Das, N. I. (2020). 'Amazon sagemaker autopilot: A white box automl solution at scale'. *International Conference on Management of Data*.

Ribeiro, M., Grolinger, K., & Capretz, M. A. (2015). 'Mlaas: Machine learning as a service', In *2015 ieee 14th international conference on machine learning and applications (icmla)*. IEEE.

Rodriguez, J. (2018). *'Inside Netflix's Notebook-Driven Architecture', Available at:* https://blog.goodaudience.com/inside-netflixs-notebook-driven-architecture-aedded32145e

Salinas, D., Flunkert, V., & Gasthaus, J. (2017). 'Deepar: Probabilistic forecasting with autoregressive recurrent networks'.

Simon, J. (2019). *'Amazon SageMaker Studio - AutoML with Amazon SageMaker AutoPilot', Available at:* https://www.youtube.com/watch?v=qMEtqJPhqpA&t=12s

Simon, J. (2020). *'Direct Marketing with Amazon SageMaker AutoPilot', Available at:* https://gitlab.com/juliensimon/awsdevdays2020/-/blob/master/mls2/XGBoost-AutoPilot.ipynb

Sugerman, J., Venkitachalam, G., & Lim, B.-H. (2001). 'Virtualizing i/o devices on vmware workstation's hosted virtual machine monitor.', In *Usenix annual technical conference, general track*.

Underwood, N. (2019). *'3 Reasons Why AWS Is Dominating the Cloud Marketplace'*, Available at: https://www.privoit.com/resources/3-reasons-why-aws-is-dominating-the-cloud-marketplace

W3C. (2004). 'Web services glossary'. *W3C Working Group Note*.

Wang, L., Von Laszewski, G., Younge, A., He, X., Kunze, M., Tao, J., & Fu, C. (2010). 'Cloud computing: A perspective study'. *New Generation Computing*, *28*(2), pp. 137–146.

Weins, K. (2020). *'Cloud Computing Trends: 2020 State of the Cloud Report', Available at:* https://www.flexera.com/blog/industry-trends/trend-of-cloud-computing-2020/