Estructura del Libro

my_epub

Índice de Contenidos

1 Cover

2 Table of Contents

- 2.1 List of Tables
- 2.2 List of Illustrations
- 2.3 Guide
- 2.4 Pages

3 Official Google Cloud CertifiedProfessional Machine Learning EngineerStudy Guide

4 F02

5 F03

6 Acknowledgments

7 About the Author

8 About the Technical Editors

- 8.1 About the Technical Proofreader
- 8.2 Google Technical Reviewer

9 Introduction

9.1 Google Cloud Professional Machine Learning Engineer Certification

9.1.1 Why Become Professional ML Engineer (PMLE) Certified?

9.1.2 How to Become Certified

9.2 Who Should Buy This Book

9.3 How This Book Is Organized

9.3.1 Chapter Features

9.4 Bonus Digital Contents

9.4.1 Interactive Online Learning Environment and Test Bank

9.5 Conventions Used in This Book

9.6 Google Cloud Professional ML Engineer Objective Map

9.7 How to Contact the Publisher

9.8 Assessment Test

9.9 Answers to Assessment Test

10 Chapter 1Framing ML Problems

10.1 Translating Business Use Cases

10.2 Machine Learning Approaches

10.2.1 Supervised, Unsupervised, and Semi■supervised Learning

10.2.2 Classification, Regression, Forecasting, and Clustering

10.3 ML Success Metrics

10.3.1 Area Under the Curve Receiver Operating Characteristic (AUC ROC)

10.3.2 The Area Under the Precision■Recall (AUC PR) Curve

10.3.3 Regression

10.4 Responsible AI Practices

10.5 Summary

10.6 Exam Essentials

10.7 Review Questions

11 Chapter 2Exploring Data and Building Data Pipelines

11.1 Visualization

11.1.1 Box Plot

11.1.2 Line Plot

11.1.3 Bar Plot

11.1.4 Scatterplot

11.2 Statistics Fundamentals

11.2.1 Mean

11.2.2 Median

11.2.3 Mode

11.2.4 Outlier Detection

11.2.5 Standard Deviation

11.2.6 Correlation

11.3 Data Quality and Reliability

11.3.1 Data Skew

11.3.2 Data Cleaning

11.3.3 Scaling

11.3.4 Log Scaling

11.3.5 Z■score

11.3.6 Clipping

11.3.7 Handling Outliers

11.4 Establishing Data Constraints

11.4.1 Exploration and Validation at Big■Data Scale

11.5 Running TFDV on Google Cloud Platform

11.6 Organizing and Optimizing Training Datasets

11.6.1 Imbalanced Data

11.6.2 Data Splitting

11.6.3 Data Splitting Strategy for Online Systems

11.7 Handling Missing Data

11.8 Data Leakage

11.9 Summary

11.10 Exam Essentials

11.11 Review Questions

12 Chapter 3Feature Engineering

12.1 Consistent Data Preprocessing

12.2 Encoding Structured Data Types

12.2.1 Why Transform Categorical Data?

12.2.2 Mapping Numeric Values

12.2.2.1 Normalizing

12.2.2.2 Bucketing

12.2.3 Mapping Categorical Values

12.2.3.1 Label Encoding or Integer Encoding

12.2.3.2 One■Hot Encoding

12.2.3.3 Out of Vocab (OOV)

12.2.3.4 Feature Hashing

12.2.3.5 Hybrid of Hashing and Vocabulary

12.2.3.6 Embedding

12.2.4 Feature Selection

12.3 Class Imbalance

12.3.1 Classification Threshold with Precision and Recall

12.3.2 Area under the Curve (AUC)

12.3.2.1 AUC ROC

12.3.2.2 AUC PR

12.4 Feature Crosses

12.5 TensorFlow Transform

12.5.1 TensorFlow Data API (tf.data)

12.5.2 TensorFlow Transform

12.6 GCP Data and ETL Tools

12.7 Summary

12.8 Exam Essentials

12.9 Review Questions

13 Chapter 4Choosing the Right ML Infrastructure

13.1 Pretrained vs. AutoML vs. Custom Models

13.2 Pretrained Models

13.2.1 Vision AI

13.2.2 Video AI

13.2.3 Natural Language AI

13.2.4 Translation AI

13.2.5 Speech■to■Text

13.2.6 Text■to■Speech

13.3 AutoML

13.3.1 AutoML for Tables or Structured Data

13.3.2 AutoML for Images and Video

13.3.3 AutoML for Text

13.3.4 Recommendations AI/Retail AI

13.3.5 Document AI

13.3.6 Dialogflow and Contact Center AI

13.3.6.1 Virtual Agent (Dialogflow)

13.3.6.2 Agent Assist

13.3.6.3 Insights

13.3.6.4 CCAI

13.4 Custom Training

13.4.1 How a CPU Works

13.4.2 GPU

13.4.3 TPU

13.4.3.1 How to Use TPUs

13.4.3.2 Advantages of TPUs

13.4.3.3 When to Use CPUs, GPUs, and TPUs

13.4.3.4 Cloud TPU Programming Model

13.5 Provisioning for Predictions

13.5.1 Scaling Behavior

13.5.2 Finding the Ideal Machine Type

13.5.3 Edge TPU

13.5.4 Deploy to Android or iOS Device

13.6 Summary

13.7 Exam Essentials

13.8 Review Questions

14 Chapter 5Architecting ML Solutions

14.1 Designing Reliable, Scalable, and Highly Available ML Solutions

14.2 Choosing an Appropriate ML Service

14.3 Data Collection and Data Management

14.3.1 Google Cloud Storage (GCS)

14.3.2 BigQuery

14.3.3 Vertex AI Managed Datasets

14.3.4 Vertex AI Feature Store

14.3.5 NoSQL Data Store

14.4 Automation and Orchestration

14.4.1 Use Vertex AI Pipelines to Orchestrate the ML Workflow

14.4.2 Use Kubeflow Pipelines for Flexible Pipeline Construction

14.4.3 Use TensorFlow Extended SDK to Leverage Pre■built Components for Common Steps

14.4.4 When to Use Which Pipeline

14.5 Serving

14.5.1 Offline or Batch Prediction

14.5.2 Online Prediction

14.6 Summary

14.7 Exam Essentials

14.8 Review Questions

15 Chapter 6Building Secure ML Pipelines

15.1 Building Secure ML Systems

15.1.1 Encryption at Rest

15.1.2 Encryption in Transit

15.1.3 Encryption in Use

15.2 Identity and Access Management

15.2.1 IAM Permissions for Vertex AI Workbench

15.2.2 Securing a Network with Vertex AI

15.2.2.1 Securing Vertex AI Workbench

15.2.2.2 Securing Vertex AI Endpoints

15.2.2.3 Securing Vertex AI Training Jobs

15.2.2.4 Federated Learning

15.2.2.5 Differential Privacy

15.2.2.6 Format■Preserving Encryption and Tokenization

15.3 Privacy Implications of Data Usage and Collection

15.3.1 Google Cloud Data Loss Prevention

15.3.2 Google Cloud Healthcare API for PHI Identification

15.3.3 Best Practices for Removing Sensitive Data

15.4 Summary

15.5 Exam Essentials

15.6 Review Questions

16 Chapter 7Model Building

16.1 Choice of Framework and Model Parallelism

16.1.1 Data Parallelism

16.1.1.1 Synchronous Training

16.1.1.2 Asynchronous Training

16.1.2 Model Parallelism

16.2 Modeling Techniques

16.2.1 Artificial Neural Network

16.2.2 Deep Neural Network (DNN)

16.2.3 Convolutional Neural Network

16.2.4 Recurrent Neural Network

16.2.5 What Loss Function to Use

16.2.6 Gradient Descent

16.2.7 Learning Rate

16.2.8 Batch

16.2.9 Batch Size

16.2.10 Epoch

16.2.11 Hyperparameters

16.2.11.1 Tuning Batch Size

16.2.11.2 Tuning Learning Rate

16.3 Transfer Learning

16.4 Semi■supervised Learning

16.4.1 When You Need Semi■supervised Learning

16.4.2 Limitations of SSL

16.5 Data Augmentation

16.5.1 Offline Augmentation

16.5.2 Online Augmentation

16.6 Model Generalization and Strategies to Handle Overfitting and Underfitting

16.6.1 Bias Variance Trade■Off

16.6.2 Underfitting

16.6.3 Overfitting

16.6.4 Regularization

16.7 Summary

16.8 Exam Essentials

16.9 Review Questions

17 Chapter 8Model Training and Hyperparameter Tuning

17.1 Ingestion of Various File Types into Training

17.1.1 Collect

17.1.2 Process

17.1.2.1 Cloud Dataflow

17.1.2.2 Cloud Data Fusion

17.1.2.3 Cloud Dataproc

17.1.2.4 Cloud Composer

17.1.2.5 Cloud Dataprep

17.1.2.6 Summary of Processing Tools

17.1.3 Store and Analyze

17.2 Developing Models in Vertex AI Workbench by Using Common Frameworks

17.2.1 Creating a Managed Notebook

17.2.2 Exploring Managed JupyterLab Features

17.2.3 Data Integration

17.2.4 BigQuery Integration

17.2.5 Ability to Scale the Compute Up or Down

17.2.6 Git Integration for Team Collaboration

17.2.7 Schedule or Execute a Notebook Code

17.2.8 Creating a User■Managed Notebook

17.3 Training a Model as a Job in Different Environments

17.3.1 Training Workflow with Vertex AI

17.3.2 Training Dataset Options in Vertex AI

17.3.3 Pre■built Containers

17.3.4 Custom Containers

17.3.5 Distributed Training

17.4 Hyperparameter Tuning

17.4.1 Why Hyperparameters Are Important

17.4.2 Techniques to Speed Up Hyperparameter Optimization

17.4.3 How Vertex AI Hyperparameter Tuning Works

17.4.4 Vertex AI Vizier

17.4.4.1 How Vertex AI Vizier Differs from Custom Training

17.5 Tracking Metrics During Training

17.5.1 Interactive Shell

17.5.2 TensorFlow Profiler

17.5.3 What■If Tool

17.6 Retraining/Redeployment Evaluation

17.6.1 Data Drift

17.6.2 Concept Drift

17.6.3 When Should a Model Be Retrained?

17.7 Unit Testing for Model Training and Serving

17.7.1 Testing for Updates in API Calls

17.7.2 Testing for Algorithmic Correctness

17.8 Summary

17.9 Exam Essentials

17.10 Review Questions

18 Chapter 9Model Explainability on Vertex AI

18.1 Model Explainability on Vertex AI

18.1.1 Explainable AI

18.1.2 Interpretability and Explainability

18.1.3 Feature Importance

18.1.4 Vertex Explainable AI

18.1.4.1 Feature Attribution

18.1.4.2 Vertex AI Example–Based Explanations

18.1.5 Data Bias and Fairness

18.1.6 ML Solution Readiness

18.1.7 How to Set Up Explanations in the Vertex AI

18.2 Summary

18.3 Exam Essentials

18.4 Review Questions

19 Chapter 10Scaling Models in Production

19.1 Scaling Prediction Service

19.1.1 TensorFlow Serving

19.1.1.1 Serving a Saved Model with TensorFlow Serving

19.2 Serving (Online, Batch, and Caching)

19.2.1 Real■Time Static and Dynamic Reference Features

19.2.2 Pre■computing and Caching Prediction

19.3 Google Cloud Serving Options

19.3.1 Online Predictions

19.3.1.1 Deploying the Model

19.3.1.1.1 Deploying a Model Using an API

19.3.1.1.2 Deploying a Model Using the Google Cloud Console

19.3.1.2 Make Predictions

19.3.1.3 A/B Testing of Different Versions of a Model

19.3.1.4 Undeploy Endpoints

19.3.1.5 Send an Online Explanation Request

19.3.2 Batch Predictions

19.4 Hosting Third■Party Pipelines (MLflow) on Google Cloud

19.5 Testing for Target Performance

19.6 Configuring Triggers and Pipeline Schedules

19.7 Summary

19.8 Exam Essentials

19.9 Review Questions

20 Chapter 11Designing ML Training Pipelines

20.1 Orchestration Frameworks

20.1.1 Kubeflow Pipelines

20.1.2 Vertex AI Pipelines

20.1.3 Apache Airflow

20.1.4 Cloud Composer

20.1.5 Comparison of Tools

20.2 Identification of Components, Parameters, Triggers, and Compute Needs

20.2.1 Schedule the Workflows with Kubeflow Pipelines

20.2.2 Schedule Vertex AI Pipelines

20.3 System Design with Kubeflow/TFX

20.3.1 System Design with Kubeflow DSL

20.3.1.1 Kubeflow Pipelines Components

20.3.2 System Design with TFX

20.4 Hybrid or Multicloud Strategies

20.5 Summary

20.6 Exam Essentials

20.7 Review Questions

21 Chapter 12Model Monitoring, Tracking, and Auditing Metadata

21.1 Model Monitoring

21.1.1 Concept Drift

21.1.2 Data Drift

21.2 Model Monitoring on Vertex AI

21.2.1 Drift and Skew Calculation

21.2.1.1 Practical Considerations of Enabling Monitoring

21.2.2 Input Schemas

21.2.2.1 Automatic Schema Parsing

21.2.2.2 Custom Schema

21.3 Logging Strategy

21.3.1 Types of Prediction Logs

21.3.1.1 Container Logging

21.3.1.2 Access Logging

21.3.1.3 Request■Response Logging

21.3.2 Log Settings

21.3.3 Model Monitoring and Logging

21.4 Model and Dataset Lineage

21.4.1 Vertex ML Metadata

21.4.1.1 Manage ML Metadataschemas

21.4.1.2 Vertex AI Pipelines

21.5 Vertex AI Experiments

21.6 Vertex AI Debugging

21.7 Summary

21.8 Exam Essentials

21.9 Review Questions

22 Chapter 13Maintaining ML Solutions

22.1 MLOps Maturity

22.1.1 MLOps Level 0: Manual/Tactical Phase

22.1.1.1 Key Features of Level 0

22.1.1.2 Challenges

22.1.2 MLOps Level 1: Strategic Automation Phase

22.1.2.1 Key Features of MLOps Level 1

22.1.2.2 Challenges

22.1.3 MLOps Level 2: CI/CD Automation, Transformational Phase

22.1.3.1 Key Features of Level 2

22.2 Retraining and Versioning Models

22.2.1 Triggers for Retraining

22.2.2 Versioning Models

22.3 Feature Store

22.3.1 Solution

22.3.2 Data Model

22.3.3 Ingestion and Serving

22.4 Vertex AI Permissions Model

22.4.1 Custom Service Account

22.4.2 Access Transparency in Vertex AI

22.5 Common Training and Serving Errors

22.5.1 Training Time Errors

22.5.2 Serving Time Errors

22.5.3 TensorFlow Data Validation

22.5.4 Vertex AI Debugging Shell

22.6 Summary

22.7 Exam Essentials

22.8 Review Questions

23 Chapter 14BigQuery ML

23.1 BigQuery – Data Access

23.2 BigQuery ML Algorithms

23.2.1 Model Training

23.2.2 Model Evaluation

23.2.3 Prediction

23.3 Explainability in BigQuery ML

23.4 BigQuery ML vs. Vertex AI Tables

23.5 Interoperability with Vertex AI

23.5.1 Access BigQuery Public Dataset

23.5.2 Import BigQuery Data into Vertex AI

23.5.3 Access BigQuery Data from Vertex AI Workbench Notebooks

23.5.4 Analyze Test Prediction Data in BigQuery

23.5.5 Export Vertex AI Batch Prediction Results

23.5.6 Export BigQuery Models into Vertex AI

23.6 BigQuery Design Patterns

23.6.1 Hashed Feature

23.6.2 Transforms

23.7 Summary

23.8 Exam Essentials

23.9 Review Questions

24 AppendixAnswers to Review Questions

24.1 Chapter 1: Framing ML Problems

24.2 Chapter 2: Exploring Data and Building Data Pipelines

24.3 Chapter 3: Feature Engineering

24.4 Chapter 4: Choosing the Right ML Infrastructure

24.5 Chapter 5: Architecting ML Solutions

24.6 Chapter 6: Building Secure ML Pipelines

24.7 Chapter 7: Model Building

24.8 Chapter 8: Model Training and Hyperparameter Tuning

24.9 Chapter 9: Model Explainability on Vertex AI

24.10 Chapter 10: Scaling Models in Production

24.11 Chapter 11: Designing ML Training Pipelines

24.12 Chapter 12: Model Monitoring, Tracking, and Auditing Metadata

24.13 Chapter 13: Maintaining ML Solutions

24.14 Chapter 14: BigQuery ML

25 Index

25.1 A

25.2 B

25.3 C

25.4 D

25.5 E

25.6 F

25.7 G

25.8 H

25.9 I

25.10 J

25.11 K

25.12 L

25.13 M

25.14 N

25.15 O

25.16 P

25.17 Q

25.18 R

25.19 S

25.20 T

25.21 U

25.22 V

25.23 W

25.24 Z

26 Online Test Bank

26.1 Register and Access the Online Test Bank

27 WILEY END USER LICENSE AGREEMENT

Total de capítulos encontrados: 433

Archivo EPUB: my_epub.epub