Microsoft

Microsoft Azure AI Fundamentals

Exam Ref AI-900

Julian Sharp

Exam Ref AI-900 Microsoft Azure AI Fundamentals

Julian Sharp

Exam Ref AI-900 Microsoft Azure AI Fundamentals

Published with the authorization of Microsoft Corporation by:
Pearson Education, Inc.

ISBN-13: 978-0-13-735803-8
ISBN-10: 0-13-735803-2

Library of Congress Control Number: 2021947615

ScoutAutomatedPrintCode

EDITOR-IN-CHIEF
Brett Bartow

EXECUTIVE EDITOR
Loretta Yates

SPONSORING EDITOR
Charvi Arora

DEVELOPMENT EDITOR
Songlin Qiu

MANAGING EDITOR
Sandra Schroeder

SENIOR PROJECT EDITOR
Tracey Croom

COPY EDITOR
Sarah Kearns

INDEXER
Timothy Wright

PROOFREADER
Donna Mulder

TECHNICAL EDITOR
Francesco Esposito

EDITORIAL ASSISTANT
Cindy Teeters

COVER DESIGNER
Twist Creative, Seattle

COMPOSITOR
codeMantra

GRAPHICS
codeMantra

Pearson's Commitment to Diversity, Equity, and Inclusion

Pearson is dedicated to creating bias-free content that reflects the diversity of all learners. We embrace the many dimensions of diversity, including but not limited to race, ethnicity, gender, socioeconomic status, ability, age, sexual orientation, and religious or political beliefs.

Education is a powerful force for equity and change in our world. It has the potential to deliver opportunities that improve lives and enable economic mobility. As we work with authors to create content for every product and service, we acknowledge our responsibility to demonstrate inclusivity and incorporate diverse scholarship so that everyone can achieve their potential through learning. As the world's leading learning company, we have a duty to help drive change and live up to our purpose to help more people create a better life for themselves and to create a better world.

Our ambition is to purposefully contribute to a world where:

- Everyone has an equitable and lifelong opportunity to succeed through learning.
- Our educational products and services are inclusive and represent the rich diversity of learners.
- Our educational content accurately reflects the histories and experiences of the learners we serve.
- Our educational content prompts deeper discussions with learners and motivates them to expand their own learning (and worldview).

While we work hard to present unbiased content, we want to hear from you about any concerns or needs with this Pearson product so that we can investigate and address them.

- Please contact us with concerns about any potential bias at https://www.pearson.com/report-bias.html.

This page intentionally left blank

I would like to dedicate this book to my wife, Clare Sharp, for her constant support and demonstrating that learning never stops.

—Julian Sharp

This page intentionally left blank

Contents at a glance

This page intentionally left blank

Contents

Acknowledgments

I'd like to thank the following people without whom this book would not have been possible.

Thank you to Loretta and Charvi for your patience and encouragement with this project. To the various editors for your correcting my errors. To Francesco for the quality of the review and your helpful suggestions. To Andrew Bettany for originally recommending me for this book and also for involving me in your Cloud Ready Graduate program, where I learned how to refine the content and the messaging around AI Fundamentals for a wider audience, making this a much better book as a result.

About the author

JULIAN SHARP is a solutions architect, trainer, and Microsoft Business Applications MVP with over 30 years of experience in IT. He completed his MA in Mathematics at the University of Cambridge. Julian has spoken at Microsoft Ignite and many other community events. For the past 15 years, he has been a Microsoft Certified Trainer delivering certification training around Dynamics 365, Azure, and the Power Platform. He has taught thousands of students with a high pass rate. Julian has a passion for Artificial Intelligence to enhance user experience and customer data in the solutions that he designs.

Introduction

Artificial Intelligence (AI) impacts almost everything we do today with devices and computers. The purpose of the AI-900 exam is to test your understanding of the fundamental concepts of AI and the AI services Microsoft provides in Azure. The exam includes high-level concepts for AI and machine learning (ML), as well as the capabilities of particular Azure AI services.

Having a high-level appreciation of AI is important to everyone involved in building and using solutions that make use of AI. There are no prerequisite skills or experience required.

Like the exam, this book takes a high-level approach and is geared toward giving you a broad understanding of the use cases for AI, as well as the common AI services in Azure. Both the exam and the book are at such a high level that there is no coding involved.

This book covers every major topic area found on the exam, but it does not cover every exam question. Only the Microsoft exam team has access to the exam questions, and Microsoft regularly adds new questions to the exam, making it impossible to cover specific questions. You should consider this book to be a supplement to your relevant real-world experience and other study materials. If you encounter a topic in this book that you do not feel completely comfortable with, use the "Need more review?" links you'll find in the text to find more information and take the time to research and study the topic. Great information is available on Microsoft Docs, and in blogs and forums.

Organization of this book

This book is organized by the "Skills measured" list published for the exam. The "Skills measured" list is available for each exam on the Microsoft Learn website: *http://aka.ms/examslist*. Each chapter in this book corresponds to a major topic area in the list, and the technical tasks in each topic area determine a chapter's organization. If an exam covers six major topic areas, for example, the book will contain six chapters.

Preparing for the exam

Microsoft certification exams are a great way to build your resume and let the world know about your level of expertise. Certification exams validate your on-the-job experience and product knowledge. Although there is no substitute for on-the-job experience, preparation through study and hands-on practice can help you prepare for the exam. This book is *not* designed to teach you new skills.

We recommend that you augment your exam preparation plan by using a combination of available study materials and courses. For example, you might use the Exam Ref and another study guide for your "at home" preparation and take a Microsoft Official Curriculum course for the classroom experience. Choose the combination that you think works best for you. Learn more about available classroom training and find free online courses and live events at *http://microsoft.com/learn*. Microsoft Official Practice Tests are available for many exams at *http://aka.ms/practicetests*.

Note that this Exam Ref is based on publicly available information about the exam and the author's experience. To safeguard the integrity of the exam, authors do not have access to the live exam.

Microsoft certifications

Microsoft certifications distinguish you by proving your command of a broad set of skills and experience with current Microsoft products and technologies. The exams and corresponding certifications are developed to validate your mastery of critical competencies as you design and develop, or implement and support, solutions with Microsoft products and technologies both on-premises and in the cloud. Certification brings a variety of benefits to the individual and to employers and organizations.

NEED MORE REVIEW? **ALL MICROSOFT CERTIFICATIONS**

For information about Microsoft certifications, including a full list of available certifications, go to *http://www.microsoft.com/learn*.

Check back often to see what is new!

Quick access to online references

Throughout this book are addresses to webpages that the author has recommended you visit for more information. Some of these links can be very long and painstaking to type, so we've shortened them for you to make them easier to visit. We've also compiled them into a single list that readers of the print edition can refer to while they read.

Download the list at *MicrosoftPressStore.com/ExamRefAI900/downloads*.

The URLs are organized by chapter and heading. Every time you come across a URL in the book, find the hyperlink in the list to go directly to the webpage.

Errata, updates, & book support

We've made every effort to ensure the accuracy of this book and its companion content. You can access updates to this book—in the form of a list of submitted errata and their related corrections—at:

MicrosoftPressStore.com/ExamRefAI900/errata

If you discover an error that is not already listed, please submit it to us at the same page.

For additional book support and information, please visit

MicrosoftPressStore.com/Support

Please note that product support for Microsoft software and hardware is not offered through the previous addresses. For help with Microsoft software or hardware, go to *http://support.microsoft.com*.

Stay in touch

Let's keep the conversation going! We're on Twitter: *http://twitter.com/MicrosoftPress*.

This page intentionally left blank

CHAPTER 2

Describe fundamental principles of machine learning on Azure

Machine learning (ML) is the current focus of AI in computer science. Machine learning focuses on identifying and making sense of the patterns and structures in data and using those patterns in software for reasoning and decision making. ML uses past experiences to make future predictions.

ML allows computers to consistently perform repetitive and well-defined tasks that are difficult to accomplish for humans. Over the past few years, machine learning algorithms have proved that computers can learn tasks that are tremendously complicated for machines and have demonstrated that ML can be employed in a wide range of scenarios and industries.

This chapter explains machine learning algorithms such as clustering, classification, and regression. The chapter then explains how machine learning works in terms of organizing datasets and applying algorithms to train a machine learning model. The chapter then looks at the process of building a machine learning model and the tools available in Azure.

Skills covered in this chapter:

- Skill 2.1: Identify common machine learning types
- Skill 2.2: Describe core machine learning concepts
- Skill 2.3: Identify core tasks in creating a machine learning solution
- Skill 2.4: Describe capabilities of no-code machine learning with Azure Machine Learning

Skill 2.1: Identify common machine learning types

Machine learning requires lots of data to build and train models to make predictions and inferences based on the relationships in data. We can use machine learning to predict a new value based on historical values and trends, to categorize a new piece of data based on data the model has already seen, and to find similarities by discovering the patterns in the data.

As humans, we can often see patterns in small datasets with a few parameters. For example, take this small set of data for students studying for this exam, as shown in Figure 2-1.

Student	Background	Hours Studied	Completed Labs	Score	Pass
Student A	Computer Science	5	No	500	No
Student B	Computer Science	6	Yes	600	No
Student C	Mathematics	8	Yes	650	No
Student D	History	10	No	623	No
Student E	Computer Science	12	Yes	733	Yes
Student F	Mathematics	15	No	697	No
Student G	Mathematics	18	No	727	Yes
Student H	Computer Science	20	Yes	850	Yes
Student I	History	20	Yes	715	Yes
Student J	English	25	No	780	Yes
Student K	English	30	No	767	Yes

FIGURE 2-1 Sample data

You can probably see that there is a pattern that shows studying more hours leads to a higher exam score and passing the exam. However, can you see a pattern between the students' academic backgrounds and whether they pass or fail, and can you answer the question of how much does completing the labs affect their score? What if you were to have more information about the student, and what if there were many more records of data? This is where machine learning can help.

This skill covers how to:

- Understand machine learning model types
- Describe regression models
- Describe classification models
- Describe clustering models

Understand machine learning model types

The amount of data created by businesses, people, their devices, and applications in ordinary daily life has grown exponentially and will grow even more as sensors are embedded into machinery in factories and in our devices and our homes. This volume of data is such that we can leverage it to improve the way we make decisions and how we operate.

Microsoft runs a competition for students called the Imagine Cup (https://imaginecup.microsoft.com). In the latest competition, students were asked to design solutions using AI to tackle global problems. As a judge, I evaluated the submissions, and the breadth and creativity of the proposals were astounding. We have not yet understood all the ways that machine learning can make a difference in our lives.

When you decide that you want to use machine learning, one of the first things is to decide what type of learning you will use in your model. The type of learning determines how your model will use data to determine its outcome:

- Supervised
- Unsupervised
- Reinforcement

Supervised learning

In *supervised learning*, the existing data contains the desired outcome. In machine learning, we say that the data contains a label. The labeled value is the output we want our model to determine for new data. A label can either be a value or a distinct category.

The other data that is supplied and that is used as inputs to the model are called features. A supervised learning model uses the features and label to train the model to fit the label to the features. After the model is trained, supplying the model with the features for new data will predict the value, or category, for the label.

You use supervised learning where you already have existing data that contains both the features and the label.

Unsupervised learning

In *unsupervised learning*, we do not have the outcome or label in the data. We use machine learning to determine the structure of the data and to look for commonalities or similarities in the data. Unsupervised learning separates the data based on the features.

You use unsupervised learning where you are trying to discover something about your data that you do not already know.

Reinforcement learning

Reinforcement learning uses feedback to improve the outcomes from the machine learning model. Reinforcement learning does not have labeled data.

Reinforcement learning uses a computer program, an agent, to determine if the outcome is optimal or not and feeds that back into the model so it can learn from itself.

Reinforcement learning is used, for example, in building a model to play chess and is commonly used in robotics.

Describe regression models

You will have probably used regression in school to draw a best fit line through a series of data points on a graph. Using the data from Figure 2-1, the hours studied are plotted against the exam scores, as shown in Figure 2-2.

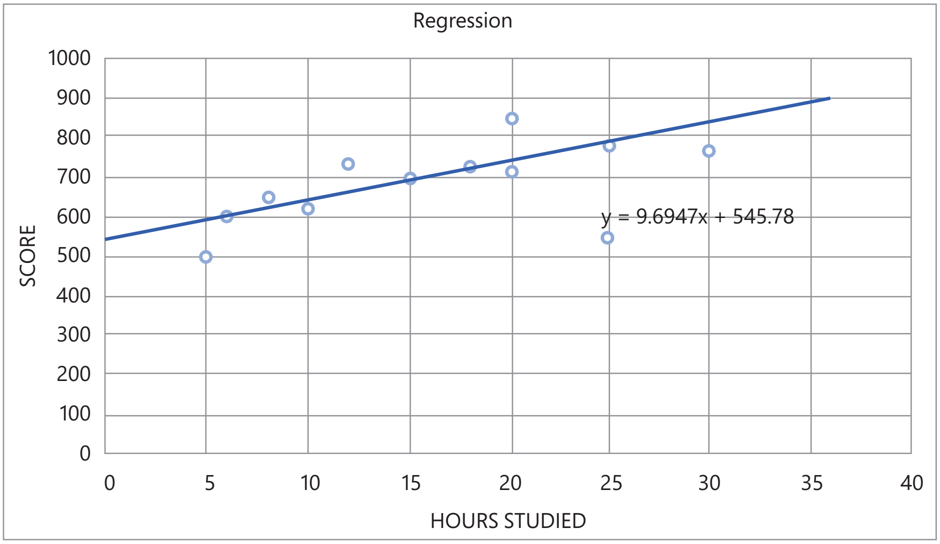

FIGURE 2-2 Regression graph

Regression is an example of supervised machine learning where the features (in this case, the hours studied) and the label (the exam score) are known and are both used to make the model fit the features to the label.

This graph is a very simple example of linear regression with just one feature. Regression models in machine learning can have many features. There are regression algorithms other than a simple linear regression that can be used.

EXAM TIP

For this exam, you do not need to know about the different algorithms, but you must be able to differentiate between the different learning models, regression, classification, and clustering.

Regression is used to predict a numeric value using a formula that is derived from historic data. Regression predicts continuous values, not distinct categories. In Figure 2-2, the formula that has been generated is $y = 9.6947x + 545.78$, which implies that every hour of studying increases the exam score by almost 10 points. We can use the model and ask the question how many hours a student should study to pass the exam (Microsoft exams require a score of 700 to pass). For 16 hours of studying, our model predicts a score of 700, a pass.

However, this is where we need to start considering how our data can affect the model. If we have another result where a student has studied for 30 hours and scored 650, the regression formula changes to $y = 6.7243x + 579.49$, as shown in Figure 2-3.

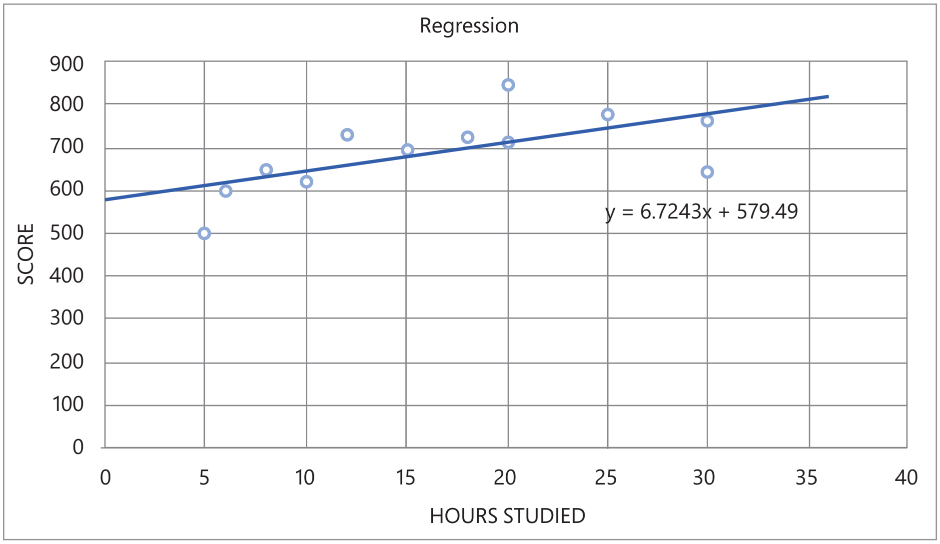

FIGURE 2-3 Regression graph with additional data

With this change to our model, we now need 18 hours of studying to pass the exam. In machine learning, one of the major concerns is how data can bias our model; we will discuss bias later in this chapter.

Describe classification models

Classification machine learning models are used to predict mutually exclusive categories, or classes. Classification involves learning using labels to classify data and is an example of supervised machine learning.

Classification is used to make predictions where we do not require continuous values but need distinct categories, such as Pass or Fail.

Using the same data from Figure 2-3, we could build and train a classification model to use the hours studied to predict whether a student passes the exam or not. Using this data, a simple two-class model will likely predict that studying for less than 18 hours will fail and 18 hours or more will pass.

In a classification model, we can compare the actual labels with the prediction from the model, as shown in the table in Figure 2-4.

Student	Hours Studied	Pass/Fail		Result
		Actual	Model	
Student A	5	No	No	True Negative
Student B	6	No	No	True Negative
Student C	8	No	No	True Negative
Student D	10	No	No	True Negative
Student E	12	Yes	No	False Negative
Student F	15	No	No	True Negative
Student G	18	Yes	Yes	True Positive
Student H	20	Yes	Yes	True Positive
Student I	20	Yes	Yes	True Positive
Student J	25	Yes	Yes	True Positive
Student K	30	Yes	Yes	True Positive
Student L	30	No	Yes	False Positive

FIGURE 2-4 Classification model

We can see that the classification model correctly predicts all but two of the results. If the model predicts a pass and the actual is a pass, this is a true positive. If the model predicts a fail and the actual is a fail, this is a true negative.

In a classification model, we are interested where the model gets it wrong. For student L, the model predicts a pass, but the actual result was a fail—this is a false positive. Student E actually passed, but the model predicted that the student will fail—this is a false negative.

Describe clustering models

Clustering machine models learn by discovering similarities, patterns, and relationships in the data without the data being labeled. *Clustering* is an example of unsupervised learning where the model attempts to discover structure from the data or tell us something about the data that we didn't know.

Clustering analyzes unlabeled data to find similarities in data points and groups them together into clusters. A clustering algorithm could be used, for example, to segment customers into multiple groups based on similarities in the customer's details and history.

A clustering model predicts mutually exclusive categories, or classes. K-means clustering is a common clustering model where K is the number of distinct clusters you want the model to group the data by. The way clustering works is to calculate the distance between the data point and the center of the cluster and then to minimize the distance of each data point to the center of its cluster.

Let's use our sample data but assume no one has taken the exam yet, so we do not have the scores or pass/fail. We have unlabeled data. Let's see if there is a relationship between the background of the students and the hours studied. We can plot these as shown in Figure 2-5.

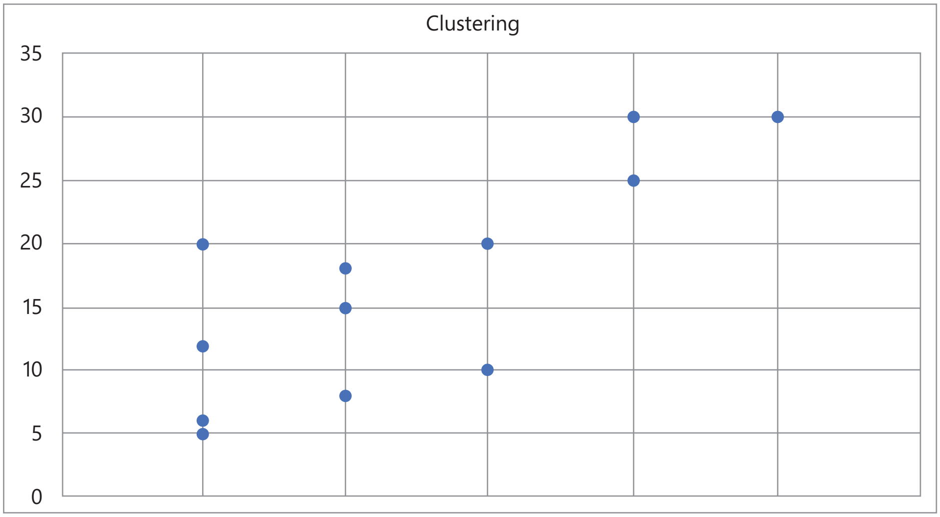

FIGURE 2-5 Clustering data

If we were to create a clustering model with K=3, then it might group the data into the three clusters—A, B, and C—as shown in Figure 2-6.

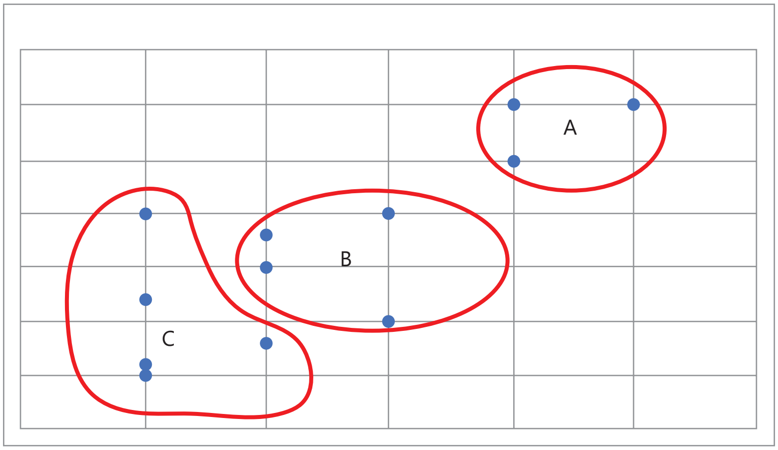

FIGURE 2-6 Clustering model

A common example of a clustering model is the recommender model. This is the model that was shown in Figure 1-1 for a company that wants to provide recommendations to its users by generating personalized targeted recommendations. A recommender model looks for similarities between customers. For example, a video streaming service knows which movies customers watch and can group customers by the types of movies they watch. A customer can then be shown other movies watched by other customers which are similar to them based on their viewing history.

Skill 2.2: Describe core machine learning concepts

Machine learning has several common concepts that apply when building machine learning models. These concepts apply no matter which tools, languages, or frameworks that you use. This section explains the fundamental concepts and processes involved in building machine learning models.

This skill covers how to:

- Understand the machine learning workflow
- Identify the features and labels in a dataset for machine learning
- Describe how training and validation datasets are used in machine learning
- Describe how machine learning algorithms are used for model training
- Select and interpret model evaluation metrics

Understand the machine learning workflow

Building a machine learning model follows the process outlined in Figure 2-7. It is important to note that building a model is an iterative process where the model is evaluated and refined.

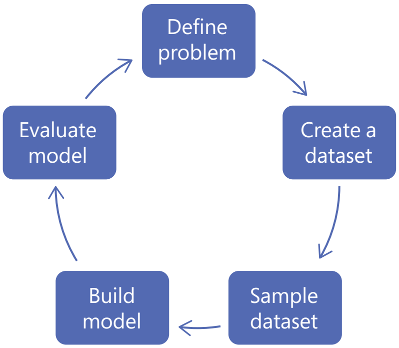

FIGURE 2-7 Machine learning workflow

First, you define the problem. This means translating the business problem into a machine learning problem statement. For example, if you are asked to understand how groups of customers behave, you would transform that as create a clustering model using customer data.

In the example used in this chapter, we want to discover how much activity a student needs to undertake to pass the exam. So far, we have kept this simple just by looking at the hours studied, but we have been asked to look at other factors such as whether they have completed the labs and their choice of degree subject. We could transform these into the following problem statements:

- Create a regression model to predict a student's score for the exam using their degree subject and their exam preparation activities.
- Create a classification model to predict if a student will pass the exam using their degree subject and their exam preparation activities.

Identify the features and labels in a dataset for machine learning

The next step in the machine learning workflow is to create a dataset. Data is the most important asset you have as you use data to train your model. If your data is incomplete or inaccurate, then it will have a major impact on how well your model performs.

You must first collect your data. This can mean extracting data from multiple systems, transforming, cleansing, and importing the data.

EXAM TIP

How to extract and transform data is outside the scope of this exam.

We will start with the same dataset we used earlier in this chapter with some additional exam results, as shown in the table in Figure 2-8.

Student	Background	Hours Studied	Completed Labs	Score	Pass
Student A	Computer Science	5	No	500	No
Student B	Computer Science	6	Yes	600	No
Student C	Mathematics	8	Yes	650	No
Student D	History	10	No	623	No
Student E	Computer Science	12	Yes	733	Yes
Student F	Mathematics	15	No	697	No
Student G	Mathematics	18	No	727	Yes
Student H	Computer Science	20	Yes	850	Yes
Student I	History	20	Yes	715	Yes
Student J	English	25	No	780	Yes
Student K	English	30	No	767	Yes
Student L	Psychology	30	No	650	No
Student M	Computer Science	20	No	727	Yes
Student N	Mathematics	17	Yes	600	No

FIGURE 2-8 Dataset

Identify labels

If you are using supervised training—for example, a regression or a classification model—then you need to select the label(s) from your dataset. Labels are the columns in the dataset that the model predicts.

For a regression model, the Score column is the label you would choose, as this is a numeric value. Regression models are used to predict a range of values.

For a classification model, the Pass column is the label you would choose as this column has distinct values. Classification models are used to predict from a list of distinct categories.

Feature selection

A *feature* is a column in your dataset. You use features to train the model to predict the outcome. Features are used to train the model to fit the label. After training the model, you can supply new data containing the same features, and the model will predict the value for the column you have selected as the label.

The possible features in our dataset are the following:

- Background
- Hours Studied
- Completed Labs

In the real world, you will have other possible features to choose from.

Feature selection is the process of selecting a subset of relevant features to use when building and training the model. Feature selection restricts the data to the most valuable inputs, reducing noise and improving training performance.

Feature engineering

Feature engineering is the process of creating new features from raw data to increase the predictive power of the machine learning model. Engineered features capture additional information that is not available in the original feature set.

Examples of feature engineering are as follows:

- Aggregating data
- Calculating a moving average
- Calculating the difference over time
- Converting text into a numeric value
- Grouping data

Models train better with numeric data rather than text strings. In some circumstances, data that visually appears to be numeric may be held as text strings, and you need to parse and convert the data type into a numeric value.

In our dataset, the background column, the degree subject names for our students, may not perform well when we evaluate our model. One option might be to classify the degree subjects into humanities and sciences and then to convert to a Boolean value, such as IsScienceSubject, with values of 1 for True and 0 for False.

Bias

Bias in machine learning is the impact of erroneous assumptions that our model makes about our data. Machine learning models depend on the quality, objectivity, and quantity of data used to train it. Faulty, incomplete, or prejudicial data can result in a poorly performing model.

In Chapter 1, we introduced the Fairness principle and how an AI model should be concerned with how data influences the model's prediction to help eliminate bias. You should therefore be conscious of the provenance of the data you are using in your model. You should evaluate the bias that might be introduced by the data you have selected.

A common issue is that the algorithm is unable to learn the true signal from the data, and instead, noise in the data can overly influence the model. An example from computer vision is where the army attempted to build a model that was able to find enemy tanks in photographs of landscapes. The model was built with many different photographs with and without tanks in them. The model performed well in testing and evaluation, but when deployed, the model was unable to find tanks. Eventually, it was realized that all pictures of tanks were taken on cloudy days, and all pictures without tanks were taken on sunny days. They had built a model that identifies whether a photograph was of a sunny or a cloudy day; the noise of the sunlight biased the model. The problem of bias was resolved by adding additional photographs into the dataset with varying degrees of cloud cover.

It can be tempting to select all columns as features for your model. You may then find when you evaluate the model that one column significantly biases the model, with the model effectively ignoring the other columns. You should consider removing that column as a feature if it is irrelevant.

Normalization

A common cause of bias in a model is caused by data in numeric features having different ranges of values. Machine learning algorithms tend to be influenced by the size of values, so if one feature ranges in values between 1 and 10 and another feature between 1 and 100, the latter column will bias the model toward that feature.

You mitigate possible bias by normalizing the numeric features, so they are on the same numeric scale.

After feature selection, feature engineering, and normalization, our dataset might appear as in the table in Figure 2-9.

ID	Feature	Feature	Feature	Label	Label
Student	**ISScienceSubject**	**Hours Studied**	**Completed Labs**	**Score**	**Pass**
Student A	1	0.1667	0	500	0
Student B	1	0.2000	1	600	0
Student C	1	0.2667	1	650	0
Student D	0	0.3333	0	623	0
Student E	1	0.4000	1	733	1
Student F	1	0.5000	0	697	0
Student G	1	0.6000	0	727	1
Student H	1	0.6667	1	850	1
Student I	0	0.6667	1	715	1
Student J	0	0.8333	0	780	1
Student K	0	1.0000	0	767	1
Student L	0	1.0000	0	650	0
Student M	1	0.6667	0	727	1
Student N	1	0.5667	1	600	0

FIGURE 2-9 Normalized dataset

Describe how training and validation datasets are used in machine learning

After you have created your dataset, you need to create sample datasets for use in training and evaluating your model.

Typically, you split your dataset into two datasets when building a machine learning model:

- **Training** The training dataset is the sample of data used to train the model. It is the largest sample of data used when creating a machine learning model.
- **Testing** The testing, or validation, dataset is a second sample of data used to provide a validation of the model to see if the model can correctly predict, or classify, using data not seen before.

A common ratio between training and validation data volumes is 70:30, but you may vary this ratio depending on your needs and size of your data.

You need to be careful when splitting the data. If you simply take the first set of rows, then you may bias your data by date created or however the data is sorted. You should randomize the selected data so that both training and testing datasets are representative of the entire dataset.

For example, we might split our dataset as shown in Figure 2-10.

ID	Feature	Feature	Feature	Label	Label
Student	**ISScienceSubject**	**Hours Studied**	**Completed Labs**	**Score**	**Pass**
Student A	1	0.1667	0	500	0
Student B	1	0.2000	1	600	0
Student C	1	0.2667	1	650	0
Student D	0	0.3333	0	623	0
Student E	1	0.4000	1	733	1
Student F	1	0.5000	0	697	0
Student G	1	0.6000	0	727	1
Student H	1	0.6667	1	850	1
Student I	0	0.6667	1	715	1
Student J	0	0.8333	0	780	1
Student K	0	1.0000	0	767	1
Student L	0	1.0000	0	650	0
Student M	1	0.6667	0	727	1
Student N	1	0.9667	1	600	0

Training

Student	ISScienceSubject	Hours Studied	Completed Labs	Score	Pass
Student A	1	0.1667	0	500	0
Student C	1	0.2667	1	650	0
Student D	0	0.3333	0	623	0
Student F	1	0.5000	0	697	0
Student G	1	0.6000	0	727	1
Student I	0	0.6667	1	715	1
Student J	0	0.8333	0	780	1
Student L	0	1.0000	0	650	0
Student M	1	0.6667	0	727	1

Testing

Student	ISScienceSubject	Hours Studied	Completed Labs	Score	Pass
Student B	1	0.2000	1	600	0
Student E	1	0.4000	1	733	1
Student H	1	0.6667	1	850	1
Student K	0	1.0000	0	767	1
Student N	1	0.9667	1	600	0

FIGURE 2-10 Training and testing datasets

Describe how machine learning algorithms are used for model training

A machine learning model learns the relationships between the features and the label in the training dataset.

It is at this point that you select the algorithm to train the model with.

> ***NOTE*** **ALGORITHMS**
>
> **For this exam, you do not need to know about algorithms or which algorithm should be used for which machine learning problem. If you want to find out more about algorithms, see https://aka.ms/mlcheatsheet.**

The algorithm finds patterns and relationships in the training data that map the input data features to the label that you want to predict. The algorithm outputs a machine learning model that captures these patterns.

Training a model can take a significant amount of time and processing power. The cloud has enabled data scientists to use the scalability of the cloud to build models more quickly and with more data than can be achieved with on-premises hardware.

After training, you use the model to predict the label based on its features. You provide the model with new input containing the features (Hours Studied, Completed Labs) and the model will return the predicted label (Score or Pass) for that student.

Select and interpret model evaluation metrics

After a model has been trained, you need to evaluate how well the model has performed. First, you score the model using the data in the testing dataset that was split earlier from the dataset. This data has not been seen by the model, as it was not used to build the model.

To evaluate the model, you compare the prediction values for the label with the known actual values to obtain a measure of the amount of error. You then create metrics to help gauge the performance of the model and explore the results.

There are different ways to measure and evaluate regression, classification, and clustering models.

Evaluate regression models

When evaluating a regression model, you estimate the amount of error in the predicted values.

To determine the amount of error in a regression model, you measure the difference between the actual values you have for the label and the predicted values for the label. These are known as the residual values. A way to represent the amount of error is to draw a line from each data point perpendicular to the best fit line, as shown in Figure 2-11.

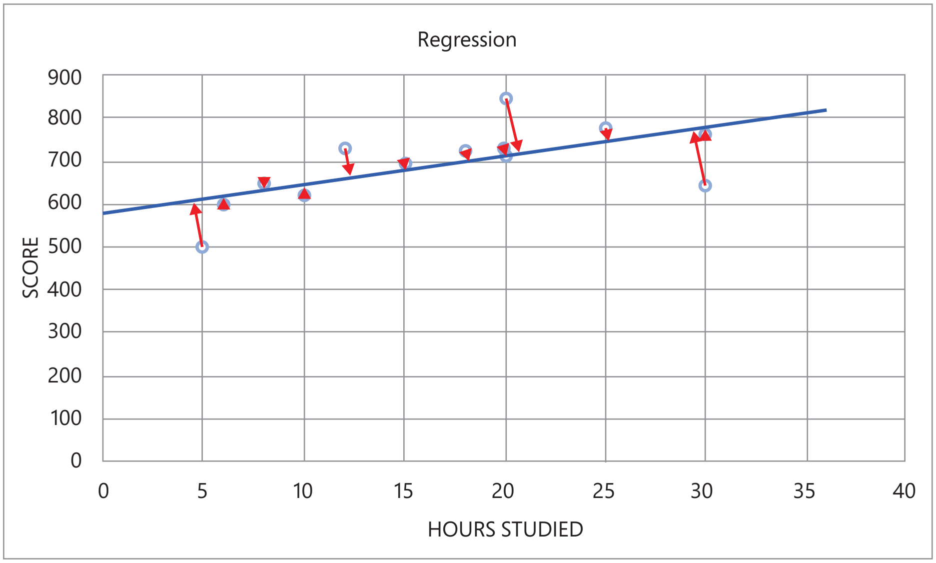

FIGURE 2-11 Regression errors

The length of the lines indicates the size of residual values in the model. A model is considered to fit the data well if the difference between actual and predicted values is small.

The following metrics can be used when evaluating regression models:

- **Mean absolute error (MAE)** Measures how close the predictions are to the actual values; lower is better.
- **Root mean squared error (RMSE)** The square root of the average squared distance between the actual and the predicted values; lower is better.
- **Relative absolute error (RAE)** Relative absolute difference between expected and actual values; lower is better.
- **Relative squared error (RSE)** The total squared error of the predicted values by dividing by the total squared error of the actual values; lower is better.
- **Mean Zero One Error (MZOE)** If the prediction was correct or not with values 0 or 1.
- **Coefficient of determination (R2 or R-squared)** A measure of the variance from the mean in its predictions; the closer to 1, the better the model is performing.

We will see later that Azure Machine Learning calculates these metrics for us.

Evaluate classification models

In a classification model with distinct categories predicted, we are interested where the model gets it right or gets it wrong.

A common way to represent how a classification model is right or wrong is to create a confusion matrix. In a confusion matrix, the numbers of true positives, true negatives, false positives, and false negatives are shown:

- **True Positive** The model predicted true, and the actual is true.
- **True Negative** The model predicted false, and the actual is false.
- **False Positive** The model predicted true, and the actual is false.
- **False Negative** The model predicted negative, and the actual is true.

The total number of true positives is shown in the top-left corner, and the total number of true negatives is shown in the bottom-right corner. The total number of false positives is shown in the top-right corner, and the total number of false negatives is shown in the bottom-left corner, as shown in Figure 2-12.

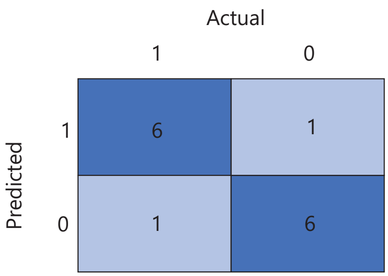

FIGURE 2-12 Confusion matrix

> ***NOTE*** **CONFUSION MATRIX**
>
> **Confusion matrixes are not always shown in the same way by different authors and tools. For instance, the actual and predicted headings may be swapped, and the 1 and 0 can be reversed.**

From the values in the confusion matrix, you can calculate metrics to measure the performance of the model:

- **Accuracy** The number of true positives and true negatives; the total of correct predictions, divided by the total number of predictions.
- **Precision** The number of true positives divided by the sum of the number of true positives and false positives.
- **Recall** The number of true positives divided by the sum of the number of true positives and false negatives.
- **F-score** Combines precision and recall as a weighted mean value.
- **Area Under Curve (AUC)** A measure of true positive rate over true negative rate.

All these metrics are scored between 0 and 1, with closer to 1 being better.

We will see later that Azure Machine Learning generates the confusion matric and calculates these metrics for us.

Evaluate clustering models

Clustering models are created by minimizing the distance of a data point to the center point of its cluster.

The Average Distance to Cluster Center metric is used when evaluating clustering models and is a measure of how focused the clusters are. The lower the value, the better.

Skill 2.3: Identify core tasks in creating a machine learning solution

Azure provides several different tools and services to build and manage machine learning models. While the tools vary, many of the tasks involved are very similar. This section describes how you use Azure Machine Learning to build and deploy machine learning models.

This skill covers how to:

- Understand machine learning on Azure
- Understand Azure Machine Learning studio
- Describe data ingestion and preparation
- Describe feature selection and engineering
- Describe model training and evaluation
- Describe model deployment and management

Understand machine learning on Azure

Azure provides many different ways to create and use Artificial Intelligence models. You can use the prebuilt models in Azure Cognitive Services, or you can build and deploy your own models with Azure Machine Learning services.

Machine learning on Azure

Microsoft provides a number of services created by Microsoft for machine learning and supports a wider set of open source and third-party services for data science and Artificial Intelligence that you can use within Azure for your own AI solutions.

The Azure Marketplace contains services and solutions for machine learning from both Microsoft and its partners, as shown in Figure 2-13.

If you are used to tools such as PyCharm or Jupyter notebooks, you can use these within Azure and leverage other Azure services such as compute and storage.

If you are used to frameworks such as PyTorch, Scikit-Learn, TensorFlow, or ONNX, you can use these frameworks within Azure.

Index

A

B

C

D

E

F

G-H-I

J-K-L

M

N

O

P

Q

R

S

T

U

V

W-X-Y-Z