Audio file

Your Recording 19.wav

Transcript

00:00:02 Speaker 2

So let's jump in to our course introduction lesson.

00:00:05 Speaker 2

In this lesson, we're going to look at what the challenges and realizations you will go through as you're learning Sagemaker, just as I did. Then we'll cover what topics are included in this course. And lastly, I'll give you some key takeaways that help you prepare for the main lessons. So let's start with challenges and realizations when you're looking to learn AWS. Maybe you've already logged into the AWS management console and clicked around some of the services in a way, a number of the services you can kind of figure out on your own, For example, EC2, the virtual machine service. If you click into that in the management console, you can kind of quite quickly realize this is something to do with things called instances, our name for virtual machines. And you can figure out, OK, here's how to launch an instance or.

00:00:35 Speaker 2

Sorry, start an instance. So it can be quite intuitive to figure out what a particular service does and what some of its features are. And in the past, that's very much how I learned a lot of the AWS services. I poked around, I looked in the management console. I kind of figured it out by myself. So when it came to learning Sagemaker, I thought it would be similar and I was wrong. Because with Sagemaker, when you click into the management console and then click into Sagemaker, there's a lot of options around. And if you click around them, you'll find things like models, which will be empty. You will find things like processing jobs, which will be empty, training jobs, which will be empty. And you'll kind of have a hard job figuring out what it is you should be doing. Even if you click on the buttons that say things like create, training job, the inputs.

00:01:05 Speaker 2

Asking for don't really make that much sense. And this is because in reality, Sagemaker is a collection of tools that are going to help you develop, train, and release a model into production. OK, but generally you're going to be doing that as a code first project. Most of the time, your approach to machine learning project will be code first. You will be working either in a code editor like Intelli J or Visual Studio Code, or you'll be working on something like Jupiter Lab and you're generally writing Python code to do that. But as you go through the steps of a machine learning pipeline, when I say pipeline, think of it as the sequence of activities I need to do to take and develop a machine learning model and take it to production. Sagemaker is going to give me the.

00:01:35 Speaker 2

That will help me achieve that job, but I need to think code first and then I'll see how Sagemaker fits in around it to help me achieve that. So it's very much focused on getting you to the end goal of how can I actually develop a model, host it, and then gain predictions from it. Now, I was lucky enough to work with a large financial institution who were onboarding their data scientists onto the Sagemaker platform. So I had a great opportunity to work with a large customer where we had data scientists who were using a variety of different methods to develop their models. And in fact, that was the problem that many data scientists were doing things their own way. Some of them were using Intelli J on their laptops, others were using corporate desktop and were using Visual Studio.

00:02:06 Speaker 2

Some of them are using scikit learn, some of them using something else, so they're doing things in their own way. So every project was very different with no lessons learned or ability to reuse concepts from one project into another. But also there was a very confused path from once they developed a model that could generate predictions on their laptop, how could we take that model forward and host it into a production environment? That kind of last mile step just wasn't clear to them. And this is when I realized that what we needed was a platform. And that's why this enterprise financial institution had moved to Sagemaker. Because they realised that having all those data scientists do those things in their own way, in a non standard way, without guardrails, without standardised operating model, then that resulted in wasted effort.

00:02:36 Speaker 2

So many scientists were generating models never actually delivered value to the business. So at this point, when my discussions with data scientists, data engineers, that realization that, OK, in a lot of ways when we look at Sagemaker, we find this mysterious, intimidating product. It's not really Sagemaker that is intimidating, mysterious in a way. It's the overall data science job or the role of how we get raw data into some kind of form where the data scientists can work with it and ultimately develop a model that ultimately will generate us predictions. And that's what the business want. The business want to get predictions from a model, you know, thereby achieving something that was not previously either able to be achieved or at a reasonable price point. So.

00:03:06 Speaker 2

In the organization I work with, they had local training on all their devices, but they're all things different. Way it was, it was a massive amount of wasted time because training models took a huge amount of compute resource could take a long time. And they decided waiting for those processes to finish and then realizing that no, that really tried again. And we do say Bombay, which production, but a lot of enterprise organizations find it's an almost impossible road to production because we not only have to adhere to our enterprise security infrastructure rules, but we also have new rules and governance rules to take care of to make sure that our models that we develop are not biased or don't leak data in some way. So a lot of models that might work well in development never actually get that roadway to production.

00:03:35 Speaker 2

If we're not collaborating between our different data scientists, then we're bringing an awful lot of rework, Something we do for project one. We have to start again and do a project 2IN actual fact, we should be leveraging the work that previous data scientist has done and seeing if we can incorporate that work and just build upon it and get a faster claim to value. Now, if we develop a model and we want to place it in production, ultimately we need to host it somewhere. We need some kind of compute platform. Do you run it on a container platform? Do you run it on Kubernetes? Do you run it on a physical server in your data center? And again, this poses challenges and blockers for getting your model actually producing predictive values for your business. Compliance is always going to be a key requirement for us, but think again about this new requirement, this ability for our model.

00:04:05 Speaker 2

Explainable to make sure it has no bias, make sure that our model doesn't leak data, make sure our model can't be manipulated. So there are new requirements beyond just our standard infrastructure ones. So it's not safe to make this intimidating. In a way, it's data science and your approach to taking a problem that you think can be solved by machine learning and taking that forward to develop into a model that can be hosted on a platform. But an awful lot of times when we start thinking about data scientists again, a number of things appear to get in the way. We need to know about things like statistics, probability, linear algebra, calculus, linear regression, convolutional neural networks, Sometimes when we're starting to learn machine learning and on the stage maker platform.

00:04:35 Speaker 2

Seems like a mountain to climb is too big that do I need to learn all of linear algebra and calculus and probability math and statistics before I open up the maker console or before I open up my Python editor? No, and in a way, this course is absolutely for you If you've looked at that mountain of learning and thought this is too much, this course is absolutely for you because we're going to give you just enough of each of these to be dangerous, give you just enough that you will be able to use the Sagemaker platform to develop a simple model and you'll know just enough around these topics so that you can actually deliver a model that will generate predictions. Now of course you could go into a master's degree in machine learning and it will delve way deeper into the math that we're going to cover but we'll give you just enough to get you productive now I was thinking example is a great.

00:05:06 Speaker 2

To really understand what's happening with any complex topic now let's imagine we buy a car now if you're going to buy a car assuming we're buying a used car here that I might do a little bit of research I might look up online and the car adverts and maybe I picked a particular model of car and I've got an idea of how much I want to spend but when you thinking about well what are the attributes of that car I want to buy because those attributes ultimately determine the price so the car was built in 2025 I'm going to expect that to be more expensive than a car that was built in 2020 or even earlier 2010. Older cars are generally going to be cheaper because they're an asset that depreciates OK this helps me so from a very basic perspective I.

00:05:36 Speaker 2

Of the correlation feature of car age with the target value, critical price, in other words, or price I would expect to pay for a car of that age. Now, if we understand the concept, you basically have the understanding of what we call linear regression and linear regression. Well, that's exactly what we did there. We plot known values. So maybe I look at a whole list of different cars for sale. When I plot how much is a 2016 going for how much 2020 car 24 carbon, I put some data points and then draw a line of best fit through those data points as linear regression. And the reason we do that with a line of best fit is that that means that I could say, well what about if I want to know how much a car that was created in 2018 was? And maybe I don't have a data point for that.

00:06:06 Speaker 2

But how my language best fit so I could just look along the X axis to the data I want look up to the line of best fit. I can figure out what price I would get for that based off of the value had derived from the line of best fit.

00:06:15 Speaker 2

Is ultimately what we're doing there is creating a prediction, isn't it? We don't have a data point for that particular page, but yet we're able to generate prediction for it. So linear regression could be extremely useful for a large number of use cases when you have popular data. And it's where most people start out when we're learning machine learning because we can understand it, can apply the real world and see how that makes a difference. So we're using the best fit to figure out what would be the price or predictive price for a car of that age based on our drawline best fit.

00:06:36 Speaker 2

Let's make our example a little bit more complex. Let's add in the number of miles that car is driven before. Let's add in colour, OK, many condition when we start to add in additional features that that car would have, not just age, things get a little more complex. We now have a number of input features for which we can use the combination of those features to figure out what an appropriate price would be. So we knew that newer cars knew that they got a higher price, but what if the mileage was excessive, like 200,000 miles? That would negatively affect the price. So it started to get more complex. Now if we just had one additional feature, we could start to work in a 3D space or additional space and you can start think, well, here's a plot throwing 3 dimensional space.

00:07:06 Speaker 2

And we go, OK, so we can figure out the cost would be with an additional feature and we can see that instead of drawing a line of best, but you've really got what we call a hyperplane. I think it's like a surface, a three-dimensional surface. You may drop to travel the rug from holding it four corners and drop it towards the ground. It would create a complex shape. So you could kind of query that 3D space and table. At what point on the surface does that generate a particular price given the combination features? So adding in an additional feature and thinking about 3 dimensional space again, we in our heads can probably understand that, get mentally accept that.

00:07:29 Speaker 2

But when we had more than just one additional feature, if we've got ten additional features, we're not going to talk about mileage and color and condition. We're going to talk about whether there's a sunroof and whether there's alloy wheels and whether there is a Bluetooth car phone in it. More features are going to create a more complex multi dimension space and it's harder for us to visualize. But from mathematics it's easy. We can have 100 dimensional space and we can still work out correlations and relationships between those features and how they relate to the target price. So just like we drew a line of best fit in our 2D example, here in our 3D example, we've got a surface, a best fit, but in A10 dimensional space, it's harder for us to visualize. The mathematics absolutely can and it can work out the.

00:07:59 Speaker 2

Surface of the best fit for the data and when we start to drain our model, ultimately that's what we are doing now looking beyond Sagemaker's features, what are we talking about here? I've mentioned a couple of times about the pipeline. Now when I say pipeline, I want you to think about the sequence of activities I'm trying to do. I'm going to collect some data I'm going to use for my sample set of data like in my car price example will be what's the list of known car prices that I found in adverts online. I might need to shape that data and get it ready into a form where I can actually train my machine learning model to use that data and work out what that not just line the best fit, but that multi dimensional surface.

00:08:26 Speaker 2

Or best fit would look like and so the model training process will do that for me. I'll then need to evaluate my model. So evaluate my model is going to be asking my model to generate prediction. But I want to generate prediction for which I know of the valid answer for. And that way I can validate that my model is producing predictions that meet pretty closely what I would expect because I would have the answer. Then I can deploy my model, in other words, host it on a server or a container somewhere and then send some prediction request data to it so it can generate me some predictions.

00:08:47 Speaker 2

And ultimately, through that process, I want to be monitoring the health of my model.

00:08:50 Speaker 2

During this course, we're going to see how Sagemaker can provide us with assistance at each of those steps along the way. So you cannot have to have a use case in mind in order to use Sagemaker because it's only going to be helpful to you when you're developing and hosting a model. You're not doing that. City Maker is of no use to you. Now, part of what we're going to look at over the coming lessons of this course will be who does what job. Now, generally, we're going to fall into two to three main categories. We have in the earlier stages, we have the data scientist and optionally the data engineer. Now, again, depending on the size of your organization, that will depend on whether you have separate data engineers from data scientists. But if you do engineers, they tend to be focused on very much the front end, the harvesting of the data from maybe multiple data sources.

00:09:20 Speaker 2

Extraction from different relation data sources for example, and maybe preprocessing that data. The data scientist is going to be focused on getting that data ready for actual training. So they'll have the knowledge about what algorithms are going to try and use and what shape the data should be into best be used by that algorithm. The data scientist will be responsible for training model and evaluating whether the model can produce good predictions. And then we'll introduce the Mlops engineer. The Mlops engineer persona will be the person who's going to really take that model that was produced by the data scientists and get it deployed out to either pre production or production environments. So we can actually generate some inference, which are inferences, just another name for prediction, and generate some predictions from our model.

00:09:49 Speaker 2

Again, Sagemaker's gonna help us every stage along the way. So we're gonna review some basic statistical methods, probably go too deep about things like mean standard deviation, things like that. We're gonna talk a little bit linear algebra. Think about the function. We describe a line of best fit, and we'll talk about some techniques around ensuring data quality. We're gonna look at use cases that we can perform. Again, I'm gonna steer you during this training course around the different persons, the data engineer, the data scientist, and the Mlops engineer. Now, we during this course are really acting as a solo practitioner who's putting on the hat of each one of those personas. But think about it at scale. It's unlikely that you will be doing all of the jobs involved in an ML pipeline. We'll talk about how we can foster team collaboration.

00:10:19 Speaker 2

We're going to look at techniques that will allow us to share our work with other data scientists that allow us to use version controlling of our assets. So we are going to talk about things like Git compatible code repositories and we're going to talk about model repositories in the form of the model registry.

00:10:30 Speaker 2

And we're going to end up with a model hosted in a production grade environment. Now at Step 4 here, that's we're actually going to use Sagemaker as our compute platform to actually host our model. Now in production, you might not be doing that, you might be hosting your model somewhere else, but we're going to show you end to end how you could use Sagemaker for everything. Not everyone will, but once we through these steps, that will unlock about 70% of Sagemaker functionality. So what are we covering? We're going to be covering some machine learning use cases. We'll talk a little around the use cases of image processing, keep learning large language models, tabular data processing, and tabular data training. But the primary focus for us will be on tabular data processing and training models based on tabular data from.

00:11:00 Speaker 2

This is in excess of 85% of what they do. They work with tabular data. So that's where our focus is going to be. Now the product we're going to be focused on is called Sagemaker AI. And you be very clear here that the name of the Sagemaker product has changed over the last 12 months or so. There used to just be a product called Maker. There is now 2 distinct products, one called Sagemaker AI and one called Sagemaker Platform. But Sagemaker Platform is really just a superset of Sagemaker AI. So everything we do here in Sagemaker AI, you can take forward and use in Sagemaker Platform. It's just Sagemaker Platform can do so much more. OK? But if you're looking at management console, you see two products. It'll be confusing actually. But let's start with Maker AI. The focus of this course, this is will allow.

00:11:30 Speaker 2

To train and develop machine learning models. So huge proportion of what we do is going to be using Sagemaker to develop models using tabular data. And in our course we are going to use an example throughout this course of house price prediction. We're going to reference a data set that's hosted on Kaggle, a well known data source site where you can get numerous data sources across different industry verticals. But we're going to use house price data set. So it's got all kinds of good data in there, like suburban area, post code, is it in number of bedrooms, square footage And then we've got price data from that. Once we've got that and we train a model in there, we will be able to then say, well what about a house in this post code, this number of bedrooms in this square footage, What would I expect to pay for that? So we'll be using.

00:12:00 Speaker 2

Example again and again because it's a good example of using linear regression remember that line of best fit or surface of best fit in multidimensional space. We're covering using Sagemaker to host what we call notebooks. These are we call Jupiter notebooks. We'll define exactly what they are how to use them we'll be covering Sagemaker domains so that we can use what we call Sagemaker Studio. We'll be covering the Sagemaker software developer kit for Python, something that basically extends Python capabilities so that it can automate things like training jobs and hosting within Sagemaker platform directly from Python code we'll be covering the Sagemaker studio and within the Sagemaker Studio we'll be covering the original Sagemaker classic. It's now being deprecated how we get a Jupyter Lab interface and how we can get a Visual Studio Code interface as well so.

00:12:30 Speaker 2

By the end of this course, you will have seen how to cleanse a tabular data set. You will have trained a model using Sagemaker, and you'll have done that training by using Python code with the Sagemaker SDK. You will then store your model in the Sagemaker model registry, which gives us version control of our model asset. We'll then use that model asset to deploy it into what we call a Sagemaker endpoint, which is essentially A hosting environment. So you're going to host the model within Sagemaker itself rather than spinning up a new server or container to do that Sagemaker. So what are our key takeaways? You're going to be able to create to train and deploy models based on tabular data set and specifically using house price data set. It's freely downloadable from Kaggle. You're going to see how to use the.

00:13:01 Speaker 2

Notebooks and you're going to use the understanding of how to get the most out of an exploratory data analysis of a data set by using Jupyter and lastly, we're going to understand the NL pipeline and again when I say NL pipeline don't think something at the moment that is an actual entity or resource in your account. I want you simply to think about the sequence of activities I need to go through to go from ideation the idea that I'm going to use linear regression to solve my machine learning problem all the way through to the point where somebody is sending a prediction to your model saying, OK, how much would house be with these features and you get a response all the steps activities that take you from A to B, that's your machine learning pipeline. I don't want you to think about who will be performing each activity along that pipeline because those different person.

00:13:31 Speaker 2

We'll do different things. We'll be using different parts of what Amazon Sagemaker AI can do for you.

00:13:35 Speaker 2

So that wraps up our introduction and hopefully frames exactly what we see and the remaining license. So let's live in and get started.

Agenda

01 SageMaker – Challenges and realizations

02 Topics to be covered in this course

03 Key takeaways for the learners

SageMaker – Challenges and Realizations

Traditional AWS Learning

- Console-based
- Feature-focused

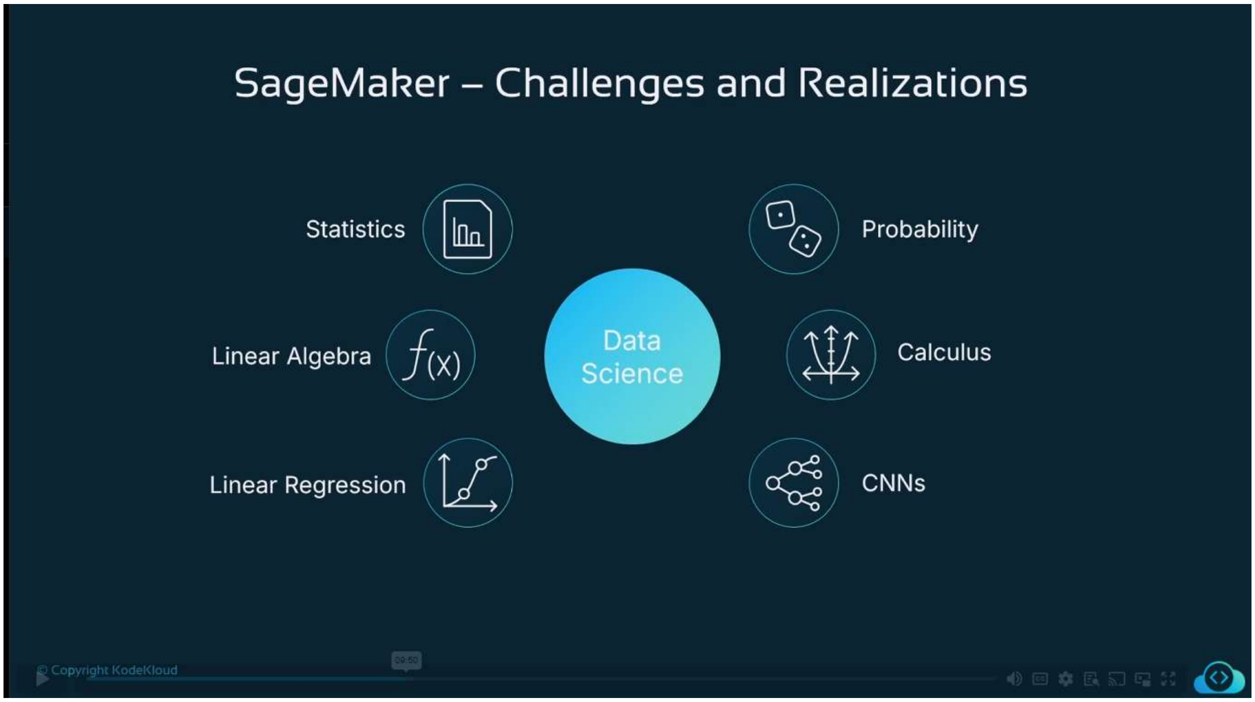
SageMaker Learning

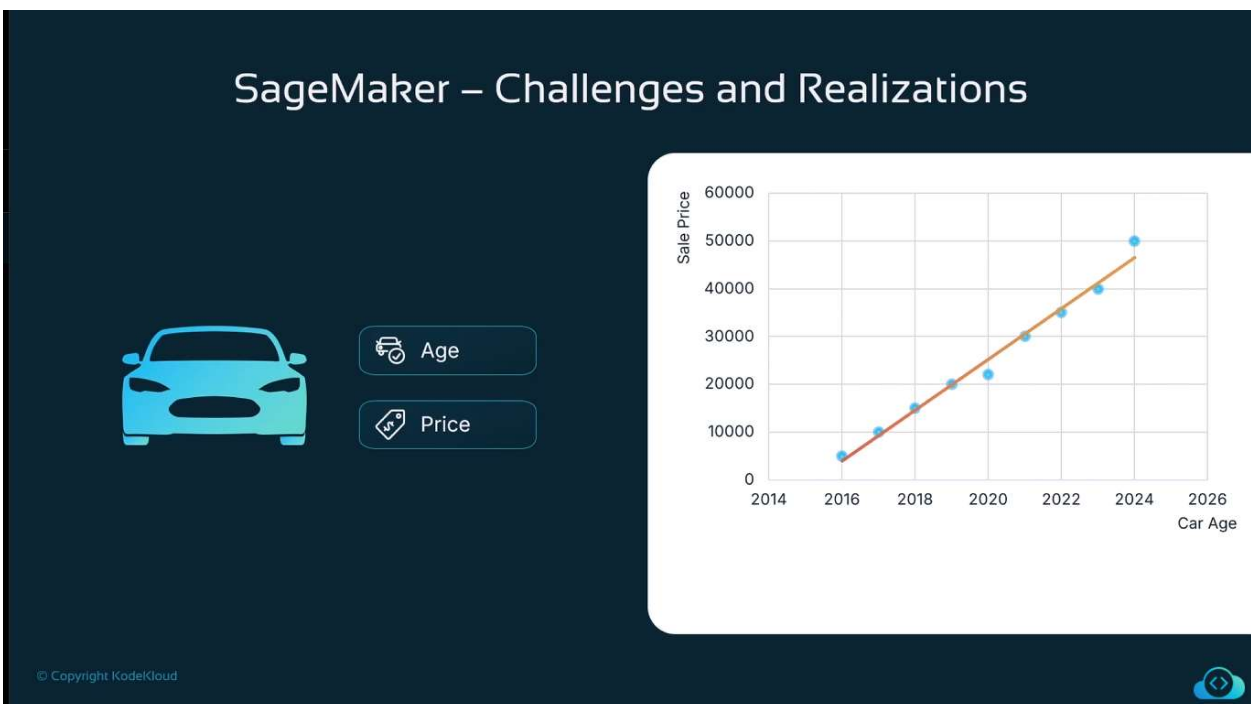
- Code-first

SageMaker – Challenges and Realizations

SageMaker – Challenges and Realizations

- Local training models lacked standardization
- Time-consuming process
- Bumpy production route
- High rework due to limited collaboration
- Many models not deployed due to complex infra requirements
- Challenging to meet compliance requirements

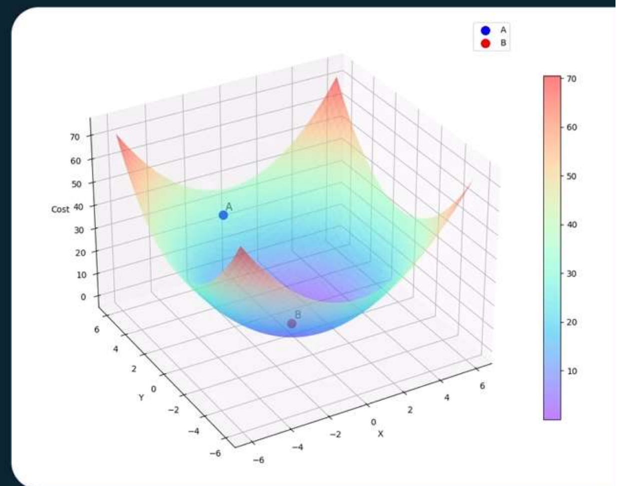

SageMaker – Challenges and Realizations
Statistics
Probability
Linear Algebra
Data Science
Calculus
Linear Regression
CNNs
© Copyright KodeKloud


SageMaker – Challenges and Realizations
Age
Price
Sale Price
60000
50000
40000
30000
20000
10000
0
2014
2016
2018
2020
2022
2024
2026
Car Age
© Copyright KodeKloud

SageMaker – Challenges and Realizations

- Mileage
- Color
- Condition

ML moves from a line to a **3D space** to predict prices.



Looking Beyond SageMaker Features

- Data Engineer
  - Data Collection
  - Data Preprocessing
- Data Scientist
  - Model Training
  - Model Evaluation
- MLOps Engineer
  - Model Deployment
  - Model Monitoring

Understand SageMaker's role at each stage

Looking Beyond SageMaker Features

01

Review basic statistics, linear algebra, data quality

02

Identify use cases as a solo practitioner

03

Foster team collaboration

04

Host a model in a production-grade environment

Unlock **>70%** of SageMaker functionality

Topics to Be Covered

SageMaker AI

Focus on building and deploying ML models

© Copyright KodeKloud

Topics to Be Covered

95%

SageMaker AI and **building an ML model** to predict house prices using tabular data

Topics to Be Covered

01 | SageMaker Notebooks

02| SageMaker Domains and User Profiles

03| SageMaker SDK for Python

04| SageMaker Studio

Studio Classic

JupyterLab

Code Editor

Topics to Be Covered

01 Data Cleansing of a Tabular Dataset

02 Training a Model in SageMaker

03 Storing a Model in SageMaker Model Registry

04 Hosting a Model in SageMaker

Key Takeaways

01

Create, train, and deploy SageMaker models using linear regression to predict house values based on historical data.

02

Leverage SageMaker SDK in Jupyter notebooks for **data prep, feature engineering, and tuning.**

03

Explore SageMaker interfaces to understand and navigate its evolving platform options.

Key Takeaways

03

Explore SageMaker interfaces to understand and navigate its evolving platform options.

04

Master Jupyter notebooks by creating and running them in JupyterLab or SageMaker environments.

05

Understand the ML pipeline, its workflow, personas, and tools to use SageMaker effectively.