AWS re:Invent

AIM215-R1

Introducing Amazon SageMaker Autopilot

Kumar Venkateswar
Principal Product Manager
Amazon Web Services

Rahul Subramaniam
CEO
DevFactory FZ LLC

AWS re:Invent

aws

Agenda

Scaling machine learning

How Amazon SageMaker Autopilot helps you scale ML

How DevFactory uses ML to scale

Demo of Amazon SageMaker Autopilot

Conclusion

We are facing a new, fundamental challenge

1,000% growth in data every two years

~2.2% growth in world population in the past two years

Most of the data is untapped… and our customers can't hire fast enough to keep up with data growth

Why ML?

Machine learning is the technology that enables computers to <u>learn</u> human expertise in a <u>scalable</u> way

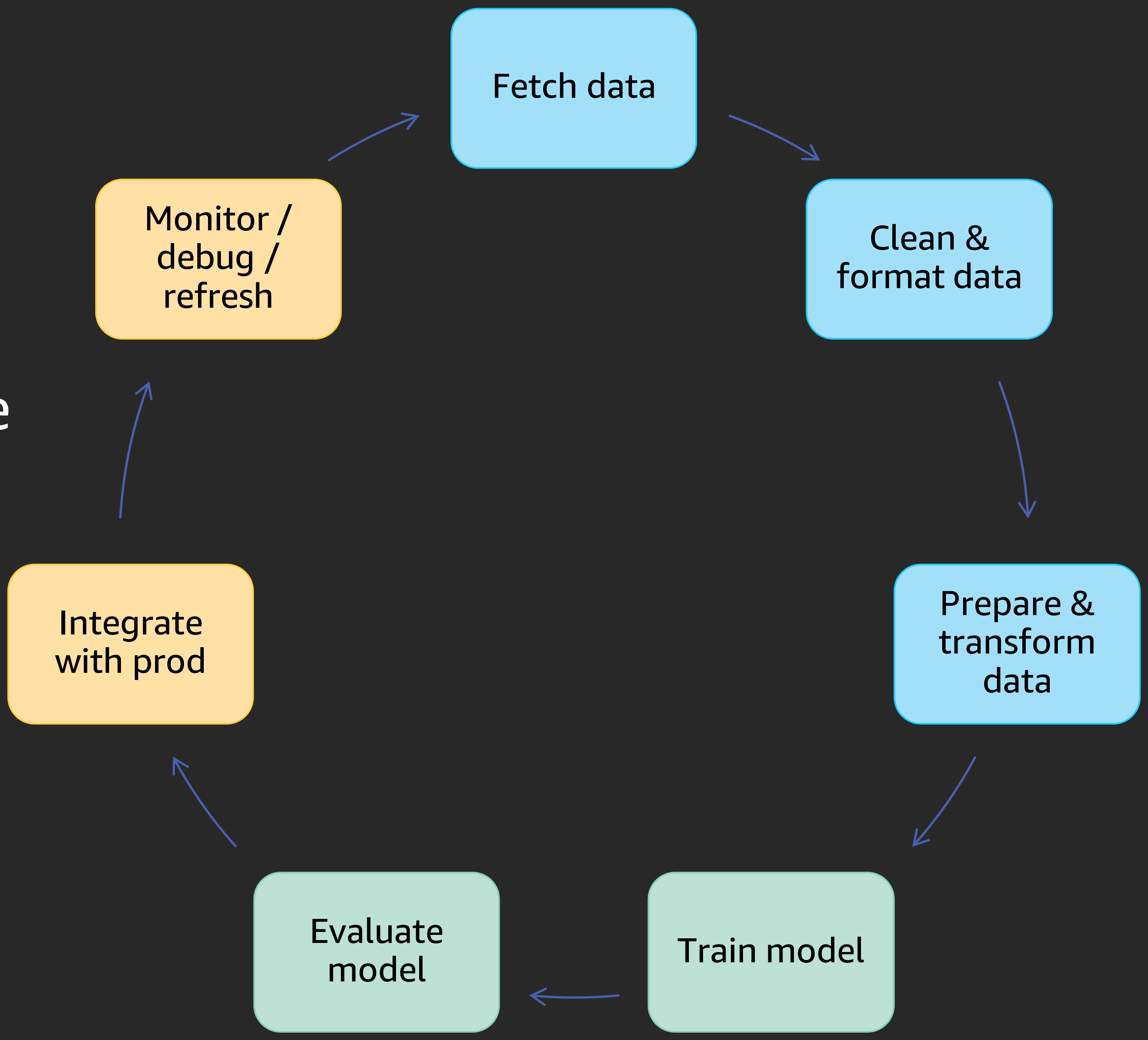
Put another way, it <u>enables humans to scale their expertise</u> to keep up and take advantage of data growth

What are the barriers to using machine learning?

Machine learning process is hard

Deploy

- Set up and manage inference clusters
- Manage and auto scale inference APIs
- Testing, versioning, and monitoring



Build

- Set up and manage notebook environments
- Get data to notebooks securely

Train

- Set up and manage clusters
- Scale/distribute ML algorithms

ML is a unique operational challenge

- ML needs iteration because success is statistical and can drift over time
- ML needs critical data and non-standard toolsets, which poses compliance challenges
- ML is computationally expensive, which means cost and scaling challenges
- Plus, there’s a desire to <u>automate</u> these processes to scale…

Scaling ML isn’t the same as training single models

Our goal is to help you scale

Build, Train, Deploy Machine Learning Models Quickly at Scale

Amazon SageMaker

SageMaker Studio

Ground Truth | Marketplace for ML | Algorithms & Frameworks | Collaborative Notebooks | Distributed Training | Tuning & Optimization | Experiments | Debugger | Deployment, Hosting, & Monitoring | Reinforcement Learning

Amazon SageMaker Autopilot

Successful ML requires complex, hard to discover combinations

of algorithms, data, parameters

Largely explorative & iterative

+

Requires broad and complete knowledge of ML domain

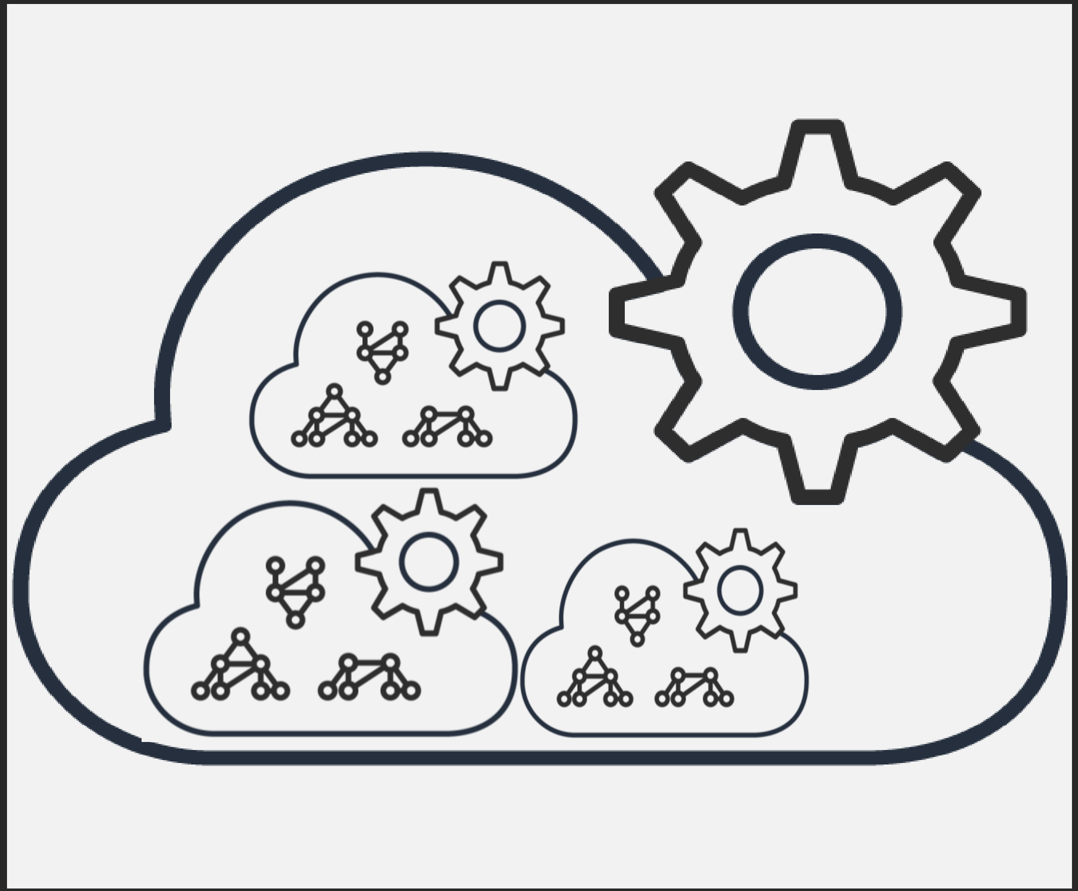
+

Combinatorial

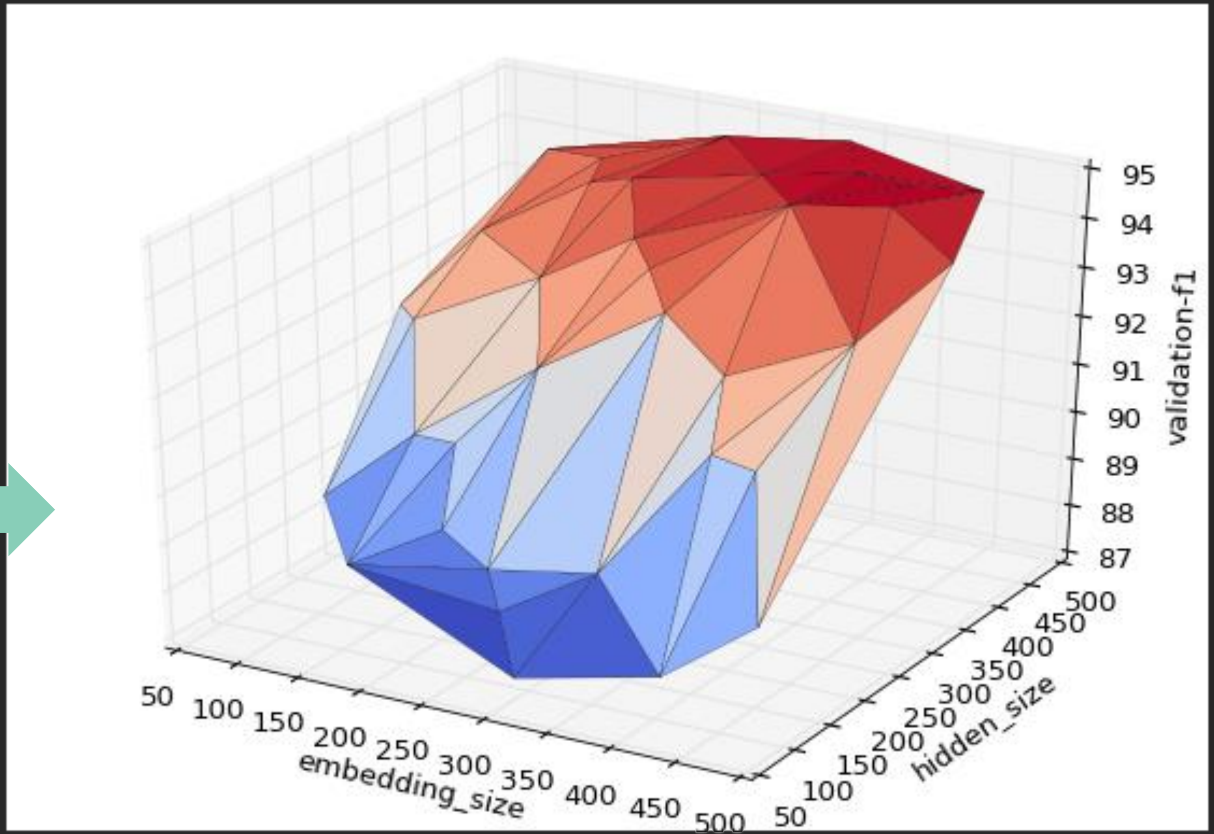
=

Time-consuming, error-prone process even for ML experts

We started with automatic model tuning



Run a large set of training jobs with varying hyperparameters…



…and search the hyperparameter space for improved accuracy

And used Bayesian Optimization for high-quality tuning

- Build a surrogate model from completed trials

$$f(\vec{x}) \to y$$

x - hyperparameter setting

y - model quality estimation (with mean/SD)

- Pick the hyperparameter setting that has highest chance of creating the best model based on the surrogate model

What about instance type, preprocessor, and algorithm selection?

Customers faced a false choice

DIY model training

- Manual effort by experts
- Fully controlled and auditable
- Experts make tradeoff decisions
- Gets better over time with experience

Automated ML

- Accessible to experts and non-experts alike
- No visibility into the training process
- Can’t make tradeoffs between accuracy and other characteristics

Customers now have a better choice

DIY model training

- Manual effort by experts
- Fully controlled and auditable
- Experts make tradeoff decisions
- Gets better over time with experience

Automated ML

- Accessible to experts and non-experts alike
- No visibility into the training process
- Can’t make tradeoffs between accuracy and other characteristics

Amazon SageMaker Autopilot

- Fully automatic model training for experts and non-experts alike
- Candidate generation notebook for control and auditing
- Easy tradeoffs by editing source code
- Learn from your experience
- Visibility into alternative candidate models

Automate machine learning with visibility and control: Amazon SageMaker Autopilot

Specify prediction target

Regression & classification

Automated feature engineering

Automated algorithm selection & HPO

Commented notebook describing actions

Integrated with Studio

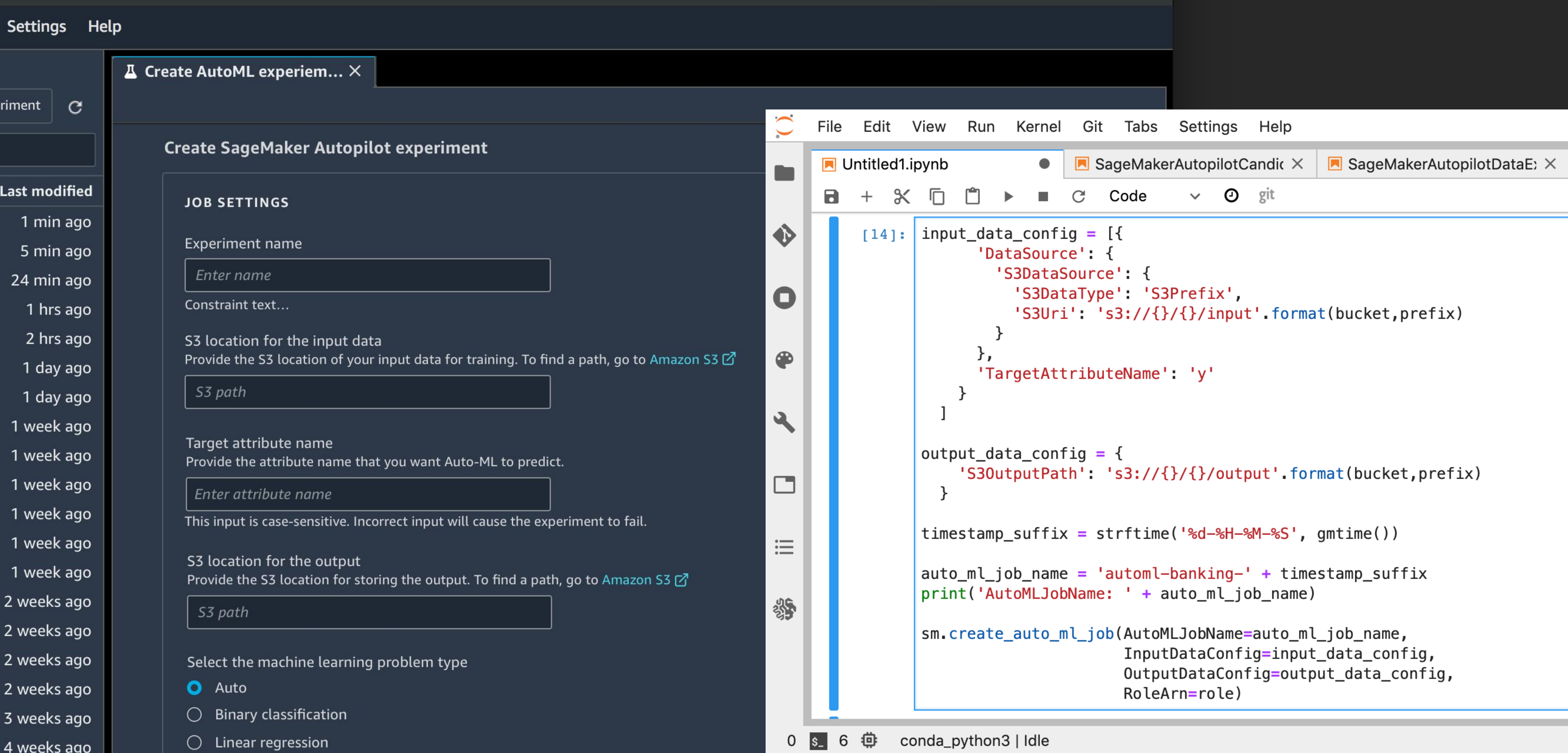
How it works

1. Candidate generation: Flintstone analyzes your dataset, calculates statistics, and determines the combination of instance type, preprocessing, and algorithm to evaluate

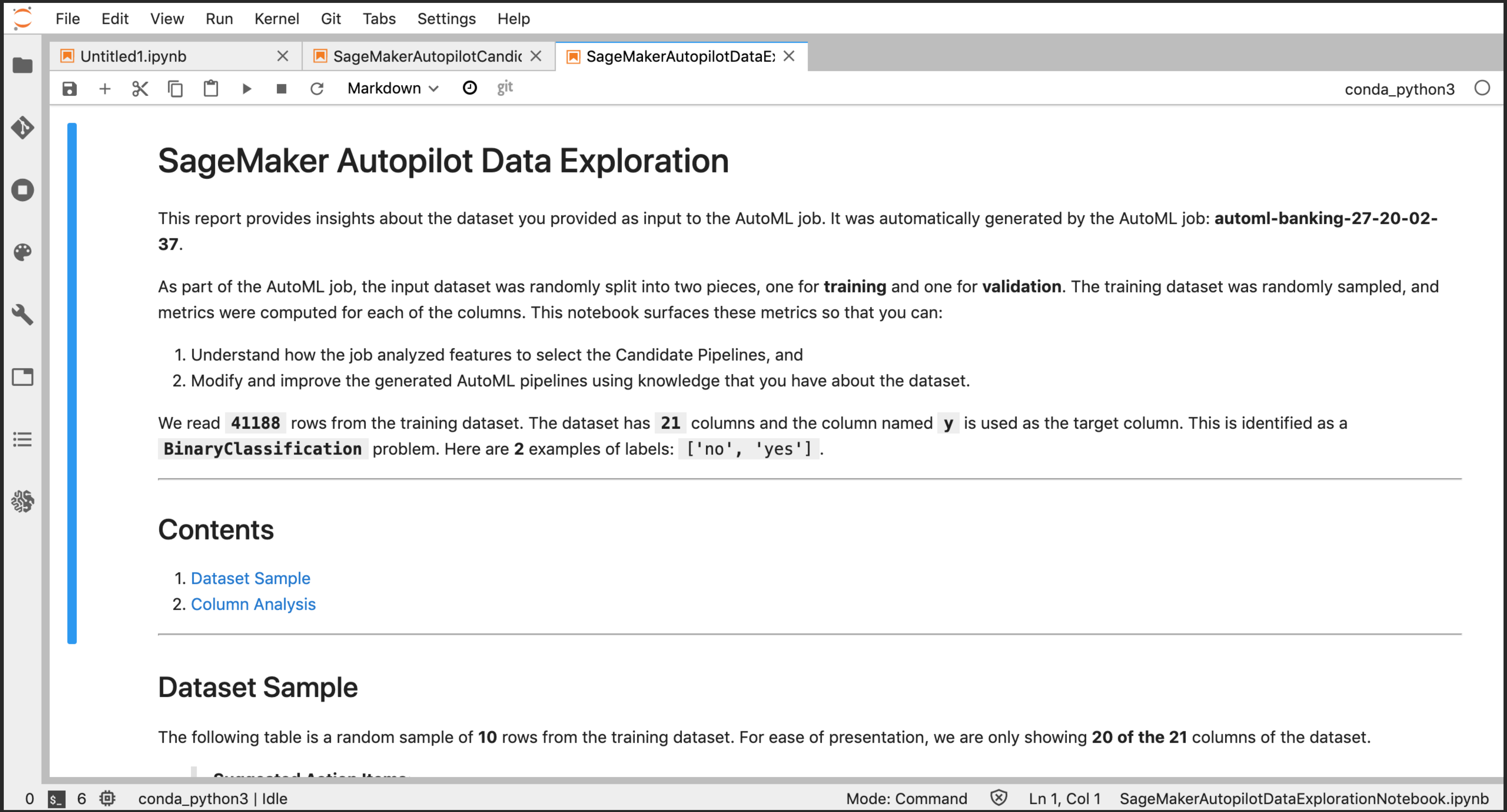
2. (Optional) candidate notebook: Flintstone provides you with a commented notebook describing the choices, enabling data scientists to provide their input to the process

3. Training and tuning: Flintstone tunes up to 7 model pipelines in parallel to determine the most accurate combination of preprocessor, algorithm, and hyperparameters

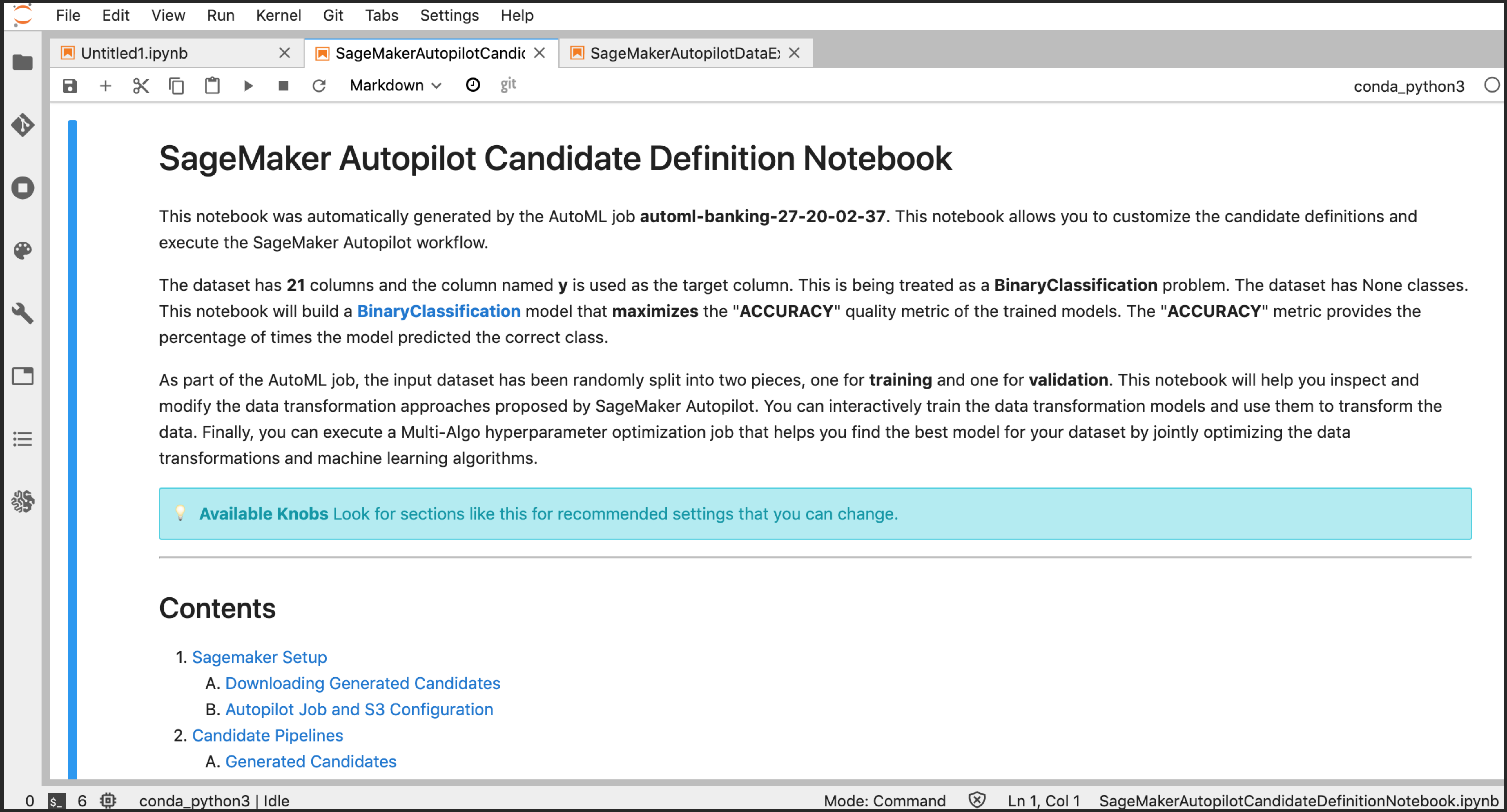
Simple to get started



Amazon SageMaker Data Exploration notebook



Amazon SageMaker Candidate Generation notebook



SageMaker Autopilot Candidate Definition Notebook

This notebook was automatically generated by the AutoML job **automl-banking-27-20-02-37**. This notebook allows you to customize the candidate definitions and execute the SageMaker Autopilot workflow.

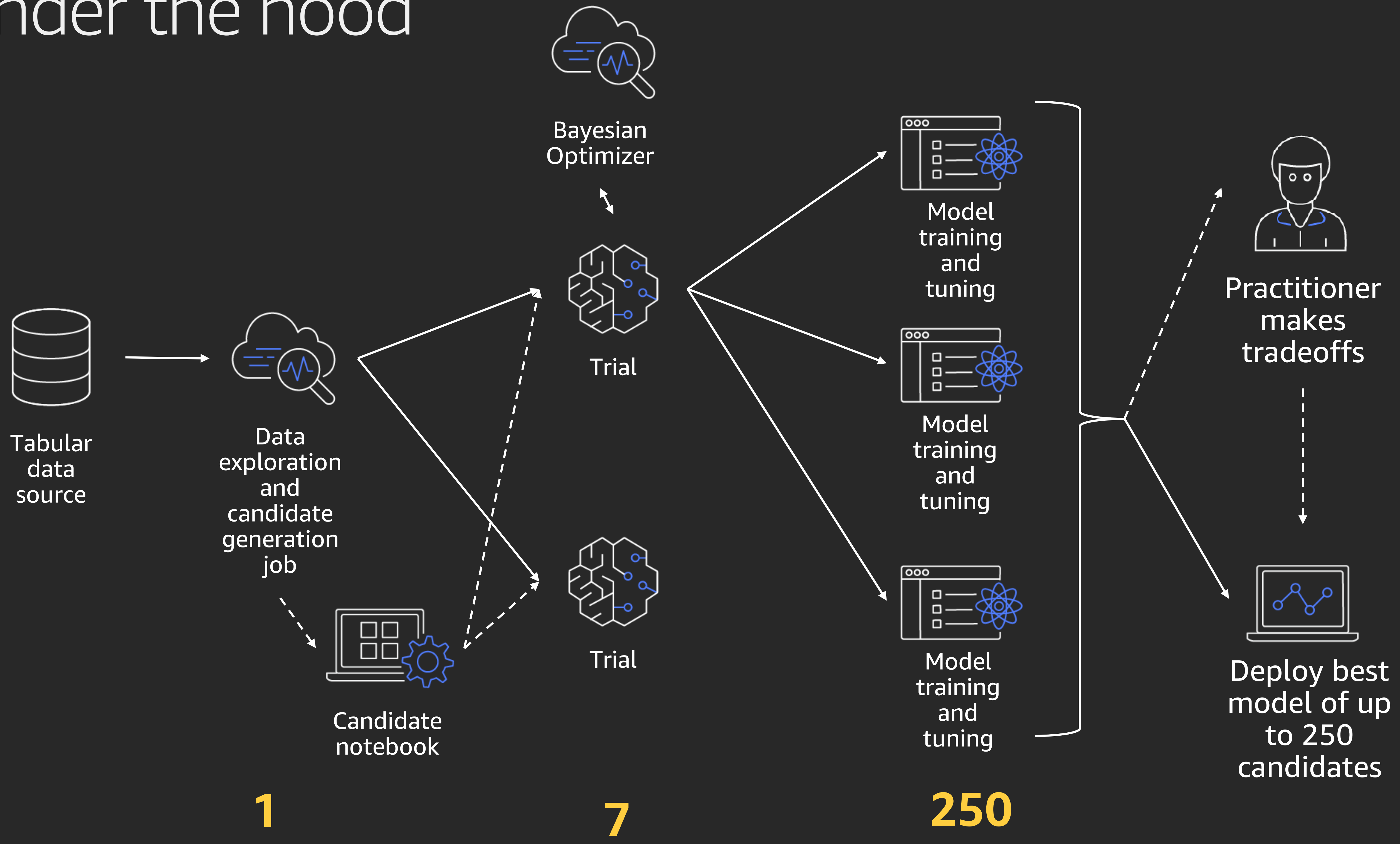
The dataset has **21** columns and the column named **y** is used as the target column. This is being treated as a **BinaryClassification** problem. The dataset has None classes. This notebook will build a **BinaryClassification** model that **maximizes** the "**ACCURACY**" quality metric of the trained models. The "**ACCURACY**" metric provides the percentage of times the model predicted the correct class.

As part of the AutoML job, the input dataset has been randomly split into two pieces, one for **training** and one for **validation**. This notebook will help you inspect and modify the data transformation approaches proposed by SageMaker Autopilot. You can interactively train the data transformation models and use them to transform the data. Finally, you can execute a Multi-Algo hyperparameter optimization job that helps you find the best model for your dataset by jointly optimizing the data transformations and machine learning algorithms.

Available Knobs Look for sections like this for recommended settings that you can change.

Contents

Under the hood



Model training involves tradeoffs

#	Model	Accuracy	Latency	Model Size
1	churn-xgboost-1756-013-33398f0	95%	450 ms	9.1 MB
2	churn-xgboost-1756-014-53facc2	93%	200 ms	4.8 MB
3	churn-xgboost-1756-015-58bc692	92%	200 ms	4.5 MB
4	churn-linear-1756-016-db54598	91%	50 ms	1.3 MB
5	churn-xgboost-1756-017-af8d756	91%	190 ms	4.2 MB

Machine Learning at DevFactory

- DevFactory is the Innovation and Development arm of the Trilogy Group

- Over 100 enterprise software products in the portfolio

Take data to the cloud securely and unlock new value for customers

DEV FACTORY

ZephyrTel CloudForward

Customer Segmentation & Churn Prediction

- Segmentation by RFM (Recency, Frequency, Monetary) as well as Usage
- ~40 Vectors driving the models

- Sensage AP– intrusion Detection and anomaly detection
- Gensym – Predicting system anomalies and breakdowns with digital twin models.

Trilogy SmartLeads lead scoring engine

- Old model built on decades of Automotive expertise – Highly handcrafted with lots of manual data massaging.
- New models rebuilt in a few weeks using Amazon SageMaker with similar or better predictions.
- Sub 100ms latency on scoring a lead.

Amazon SageMaker is a remarkable tool to unlock amazing insights from your data – you only need to know what question to ask.

Demo

AWS re:Invent

Conclusion

Agenda

Scaling machine learning

How Amazon SageMaker Autopilot helps you scale ML

How DevFactory uses ML to scale

Demo of Amazon SageMaker Autopilot

Conclusion

Thank you!

AWS re:Invent

Please complete the session survey in the mobile app.