Amazon Redshift ML

Democratize Machine Learning using SQL

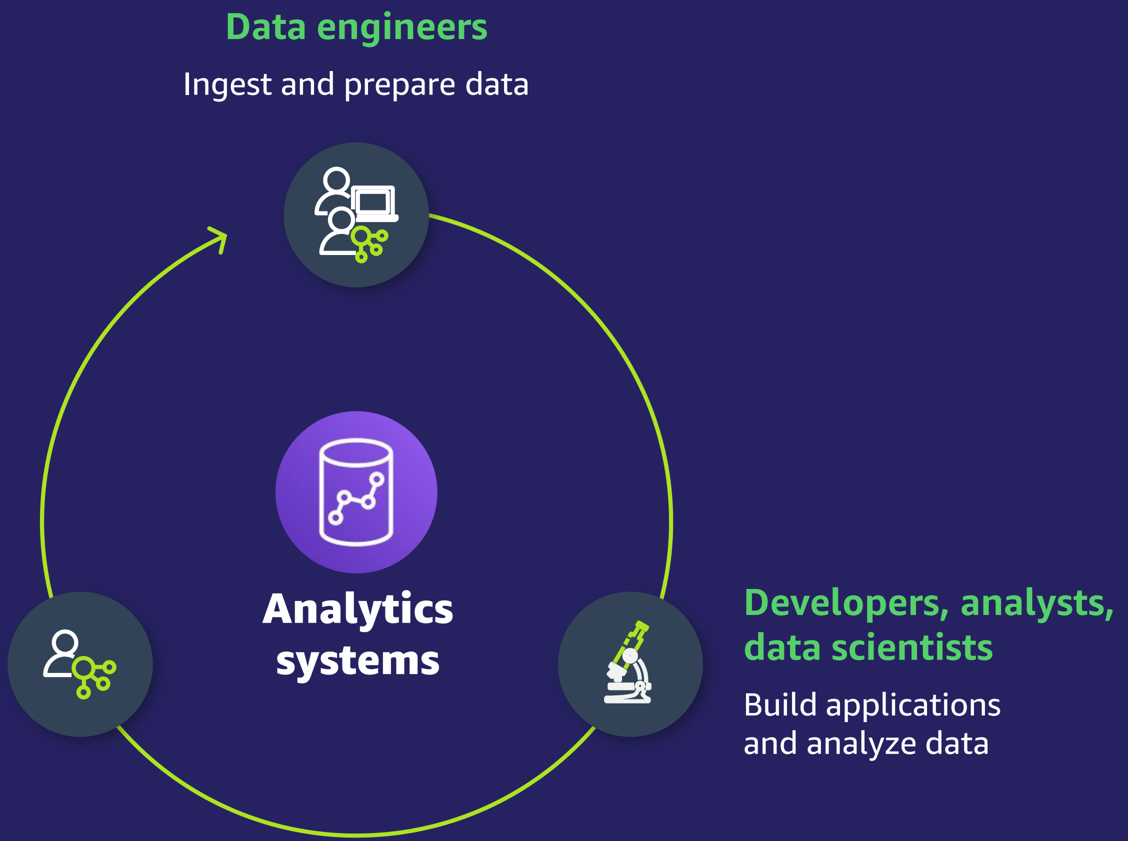
Srikanth Sopirala

Principal Analytics Specialist SA
Amazon Web Services

Agenda

- Amazon Redshift overview
- Benefits of machine learning
- Use cases
- Features deep dive
- Demonstration
- Summary and Additional resources

Data can work more effectively for diverse users through easy-to-deploy, self-serve, and auto-scaling analytics systems

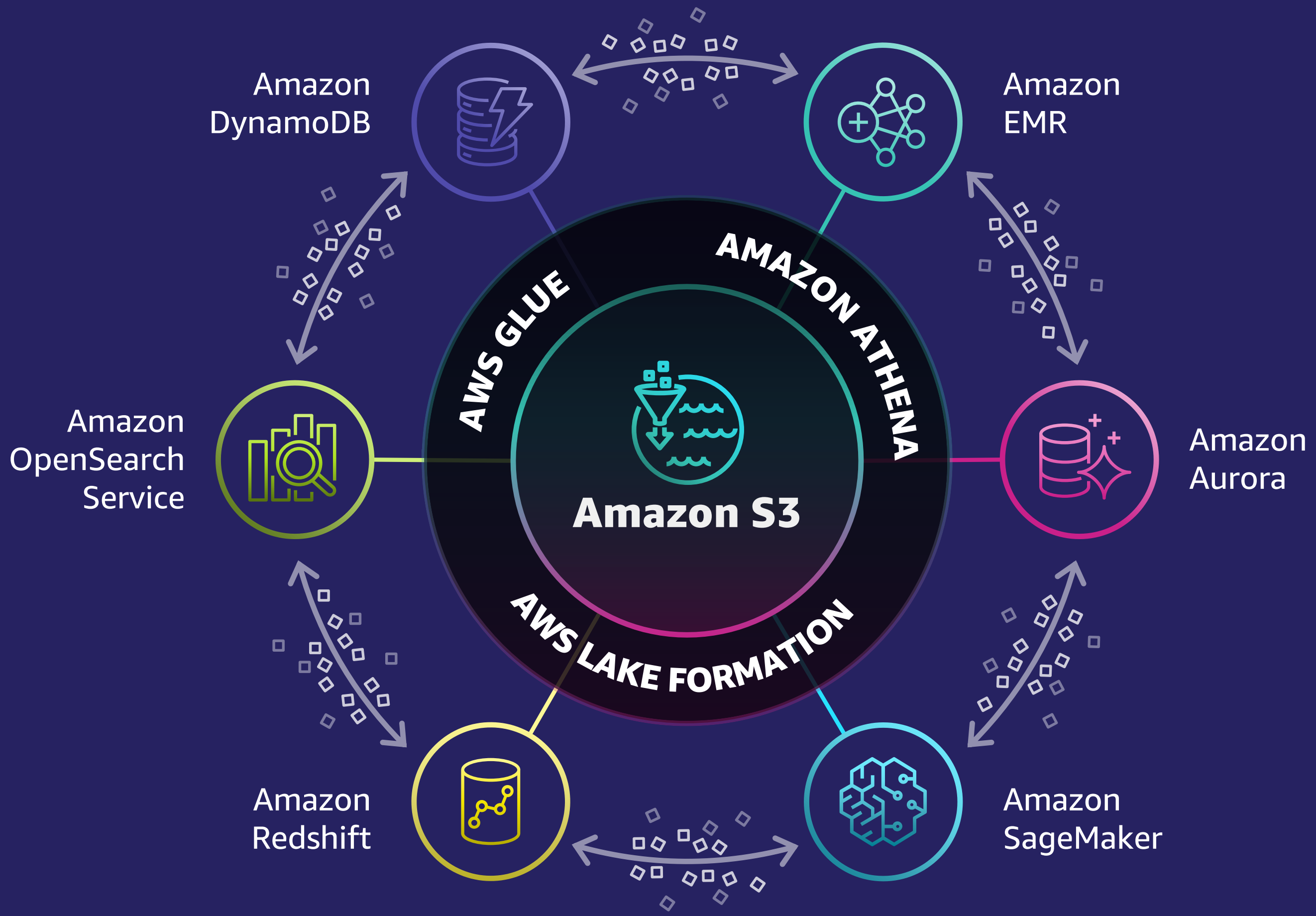

Data engineers
Ingest and prepare data
Analytics systems
Business leaders
Make business decisions and deliver on outcomes
Developers, analysts, data scientists
Build applications and analyze data

Improve operational efficiency; make more informed decisions; accelerate innovation

By making 10% more data accessible, a typical Fortune 1000 company will see a ***$65 million increase in net income***[1]

[1] Dykes, "The Four Key Pillars to Fostering a Data-Driven Culture"

Modern data architecture on AWS



Why Amazon Redshift for your data needs?

FAST, EASY, AND SECURE ANALYTICS AT SCALE

Easy analytics for everyone

Focus on getting from data to insights in seconds without worrying about infrastructure

Analyze all your data

Get insights running real-time and predictive analytics on complex, scaled data across your operational databases, data lake, and data warehouse

Best price performance at any scale

Gain up to 3x better price performance than other cloud data warehouses, and dynamically scale to improve query speed for complex and critical workloads

Tens of thousands of customers process exabytes of data with Amazon Redshift daily

Schneider Electric

Supports tens of thousands of users through Redshift concurrency scaling and RA3 nodes to support a green future

Nasdaq

Accommodated the jump from 30 billion records to 70 billion records a day because of the flexibility and scalability of Amazon S3 and Amazon Redshift

Magellan Rx MANAGEMENT

Redshift ML for predictive analytics to forecast future drug costs and identify trends

zynga

ETL performance improved 2x and scaled to process > 5.3 TB of daily game data

jobcase

Performs model inference on 2–3 billion job search recommendations in 15–20 minutes, down from 2–3 hours, with no additional cost

AstraZeneca AMGEN ancestry coursera DOW JONES EQUINOX GE yelp EA FINANCIAL TIMES docomo

intuit London Stock Exchange Liberty Mutual FOX Pfizer QANTAS SCHOLASTIC Sysco tinder jobcase WB

aws

Personas that use AWS Analytics services

Data Engineers/Database Developers

Data Analysts

Data Scientists

BI Professionals

Administrators

Benefits of ML

Transform customer experience

Improve business operations

Better and faster decision-making

Innovate product or service

Common ML use cases in a data warehouse

Customer churn detection

Predict if a sales lead will close

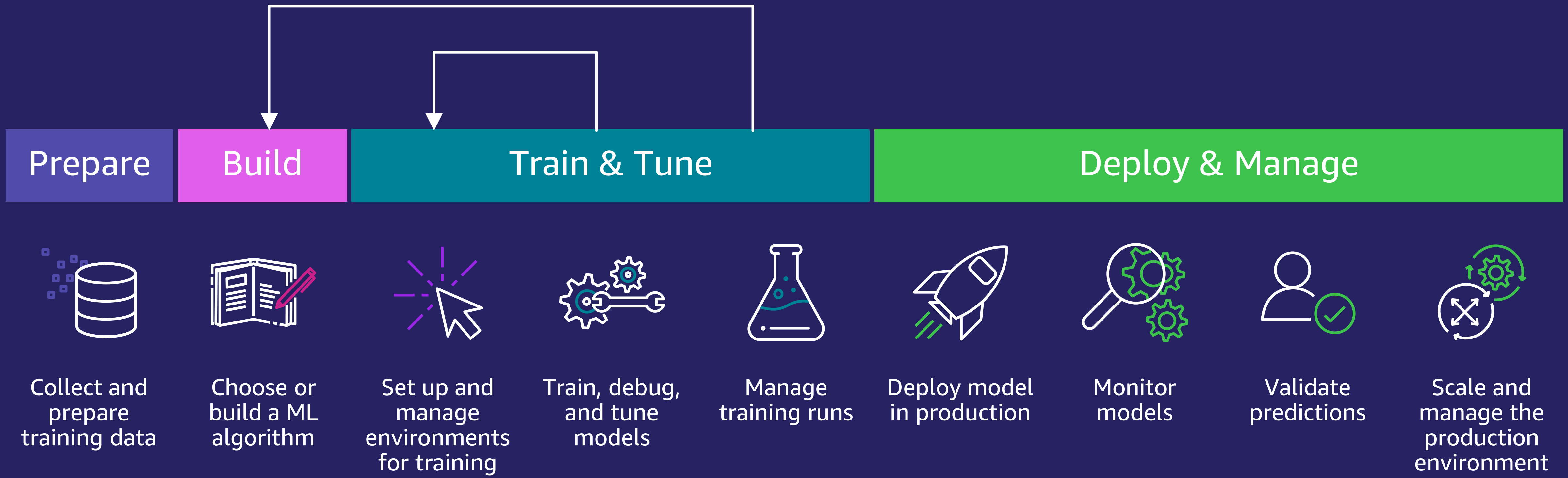
Price/revenue prediction

Product recommendation

Fraud detection

Customer lifetime value prediction

ML workflows can be complex and iterative



Prepare

Build

Train & Tune

Deploy & Manage

Collect and prepare training data

Choose or build a ML algorithm

Set up and manage environments for training

Train, debug, and tune models

Manage training runs

Deploy model in production

Monitor models

Validate predictions

Scale and manage the production environment

ML requirements from data warehouse users

DATA ANALYSTS and DEVELOPERS

Want to train ML models and make ML-based predictions without having to learn complex ML concepts and external ML tools

DATA SCIENTISTS

Want to perform ML training and prediction within the data warehouse

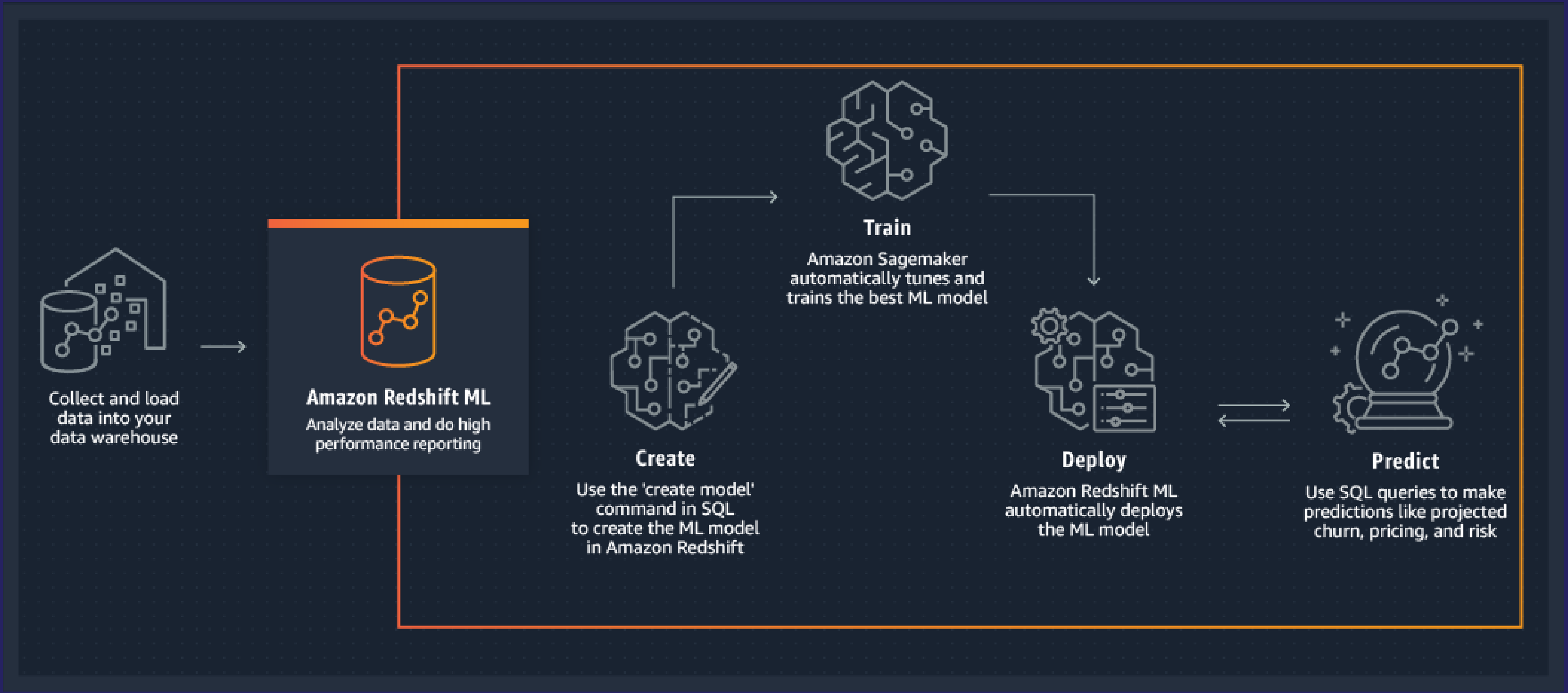
BI PROFESSIONALS

Want to use ML-based prediction with the queries they use in their dashboards and reports

Our mission at AWS

Put the power of ML in the
hands of every data analyst, database
developer, and every data warehouse user

Amazon Redshift ML



Collect and load data into your data warehouse

Amazon Redshift ML

Analyze data and do high performance reporting

Create

Use the 'create model' command in SQL to create the ML model in Amazon Redshift

Train

Amazon Sagemaker automatically tunes and trains the best ML model

Deploy

Amazon Redshift ML automatically deploys the ML model

Predict

Use SQL queries to make predictions like projected churn, pricing, and risk

Amazon Redshift ML : Benefits

EASILY TRAIN AND USE ML IN SQL QUERIES WITH AMAZON SAGEMAKER

Simple

Create your model with SQL and use prediction from SQL

Flexible

Trains or tunes the best ML algorithm for your task and gives you power to select algorithm (e.g., XGBoost)

Automatic

Automatic pre-processing, creation, training, tuning, and deployment of your model

Performant

Models are compiled with SageMaker Neo and deployed in Amazon Redshift; prediction happens locally and efficiently in your data warehouse

Secure

You do not have to worry about managing governance of data; data never leaves your VPC

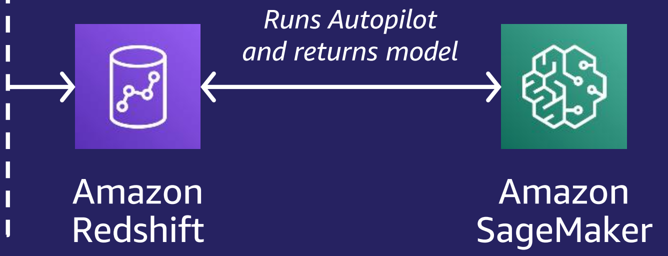
Cost optimized

You only pay for training while prediction comes at no extra cost

How Amazon Redshift ML works

TRAIN

```
CREATE MODEL customer_churn
FROM (SELECT c.age, c.zip,
c.monthly_spend, c.monthly_cases,
c.active AS label
FROM customer_info_table c)
TARGET label
FUNCTION predict_customer_churn
```



Runs Autopilot and returns model

Amazon Redshift

Amazon SageMaker

PREDICT

```
SELECT n.id, n.firstName, n.lastName,
predict_customer_churn(n.age,c.zip,..)
AS activity_prediction
FROM new_customers n
WHERE n.marital_status = ‘single’
```

Amazon Redshift

Uses previously built model to predict in-place (inference executed entirely in Amazon Redshift)

Create, train, and deploy model with a simple SQL command

Auto-selection of model, pre-processing, and training using SageMaker Autopilot

Trained model gets compiled by SageMaker Neo in Amazon Redshift data warehouse so that you can make predictions using SQL

Checking the status of ML Model

Check status of model with SHOW MODEL command

SHOW MODEL ALL shows all your models

Provides status of the models

System table STV_ML_MODEL_INFO provides the model status

```
SHOW MODEL customer_churn
```

```
Key                                            Value
Model Name                                     customer_churn
Schema Name                                    demo_ml
Owner                                          demouser
Creation Time                                  "Tue, 24.11.2020 07:02:51"
Model State                                    READY
validation:                                    f1,0.681240
Estimated Cost                                 0.990443
TRAINING DATA:,
Query          "SELECT STATE, AREA_CODE, TOTAL_CHARGE/ACCOUNT_LENGTH AS
AVERAGE_DAILY_SPEND, CUST_SERV_CALLS/ACCOUNT_LENGTH AS
AVERAGE_DAILY_CASES, CHURN"
FROM DEMO_ML.CUSTOMER_ACTIVITY
WHERE ACCOUNT_LENGTH > 120
Target Column,                                 Active

PARAMETERS:,
Model Type                                     auto
Problem Type                                   BinaryClassification
Objective                                      F1
Function Name                                  predict_customer_churn
Function Parameters,                           "state area_code average_daily_spend
average_daily_cases "
Function Parameter Types  "varchar int4 float8 int4 "
IAM Role                                       arn:aws:iam::9999999999:role/RedshiftML
s3 Bucket                                      redshiftml
Max Runtime                                    1800
```

Persona Examples

Creating and training ML model

DATA ANALYST

Specify training data as a table name or SELECT query

TARGET column specifies the column you are trying to predict

FUNCTION specifies the name of the prediction function that will be generated

```
CREATE MODEL customer_churn

FROM (SELECT c.age as feat_1, c.zip AS feat_2,
c.monthly_spend AS feat_3, c.monthly_cases AS
feat_4, c.active AS label
FROM customer_info_table c)

TARGET label

FUNCTION predict_customer_churn
```

Using ML Model for Prediction

The prediction (inference) function is available as a UDF

You can generate prediction from any SQL construct just as you use UDFs today

You can use WLM to prioritize your compute resources for inference function

Prediction function takes all benefits of Amazon Redshift, including the massively parallel processing capability

```
SELECT customer_id,
predict_customer_churn(age, zip,
monthly_spend, monthly_cases)

FROM customer_info_table;
```

Training with PROBLEM TYPE and Objective

CITIZEN DATA SCIENTIST

PROBLEM_TYPE can be *REGRESSION | BINARY_CLASSIFICATION | MULTICLASS_CLASSIFICATION*

OBJECTIVE Specifies the name of the objective metric used to measure the predictive quality of a machine learning system 'MSE' | 'Accuracy' | 'F1' | 'F1Macro' | 'AUC'

```
CREATE MODEL customer_churn

FROM (SELECT c.age as feat_1, c.zip AS feat_2,
c.monthly_spend AS feat_3, c.monthly_cases AS
feat_4, c.active AS label
FROM customer_info_table c)

TARGET label

FUNCTION predict_customer_churn

PROBLEM_TYPE  BINARY_CLASSIFICATION

OBJECTIVE 'F1'
```

Creating and training ML model

DATA SCIENTIST

Optionally specify:

Model type; e.g., XGBOOST

Objective for training; e.g., mean squared error (MSE)

Preprocessors or hyperparameters

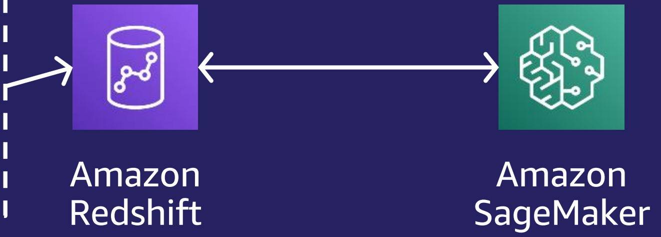
```
CREATE MODEL model_abalone_xgboost_regression
FROM (SELECT shell_weight, …….rings
FROM abalone_xgb_train)
TARGET Rings
FUNCTION func_model_abalone_xgboost_regression
IAM_ROLE
'arn:aws:iam::963462676454:role/Redshift-ML'
AUTO OFF
MODEL_TYPE xgboost
OBJECTIVE 'reg:squarederror'
PREPROCESSORS 'none'
HYPERPARAMETERS DEFAULT EXCEPT (NUM_ROUND
'100')
```

Bring your model to Amazon Redshift ML

INVOKE REMOTE SAGEMAKER ENDPOINT FROM SQL

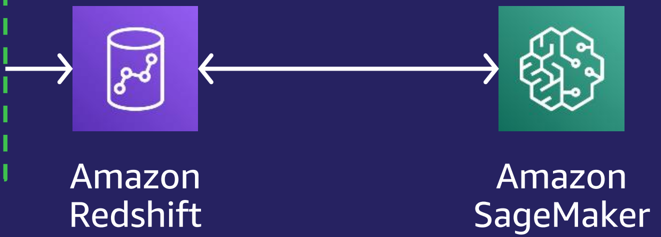
CREATE MODEL

```
CREATE MODEL remote_customer_ltv
FUNCTION  customer_ltv(
integer,integer)
RETURNS float4
SAGEMAKER ‘…’
IAM_ROLE ‘…’;
```



PREDICT

```
SELECT n.id, n.firstName, n.lastName,
customer_ltv(n.age,c.zip)
AS activity_prediction
FROM new_customers n
WHERE n.marital_status = ‘single’
```



Create, train, and deploy model in SageMaker. Make available in Amazon Redshift via SageMaker External Function

Invoke your model from Amazon Redshift

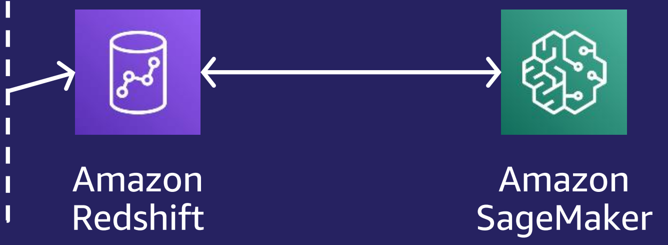
Provides you full flexibility and algorithms of Amazon SageMaker

Bring your model to Amazon Redshift ML

IMPORT SAGEMAKER TRAINED MODELS FOR LOCAL INFERENCE

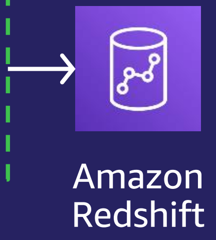
CREATE MODEL

```
CREATE MODEL customer_churn
FROM 'training-job-customer-churn-v4'
FUNCTION customer_ltv
(varchar, int, float, float)
RETURNS int
;
```



PREDICT

```
SELECT n.id, n.firstName, n.lastName,
customer_ltv(n.age,c.zip)
AS activity_prediction
FROM new_customers n
WHERE n.marital_status = ‘single’
```



Create, train, model in SageMaker (XGBoost or MLP)

Import the model into Amazon Redshift

Trained model gets compiled by SageMaker Neo in Amazon Redshift data warehouse so that you can make predictions using SQL

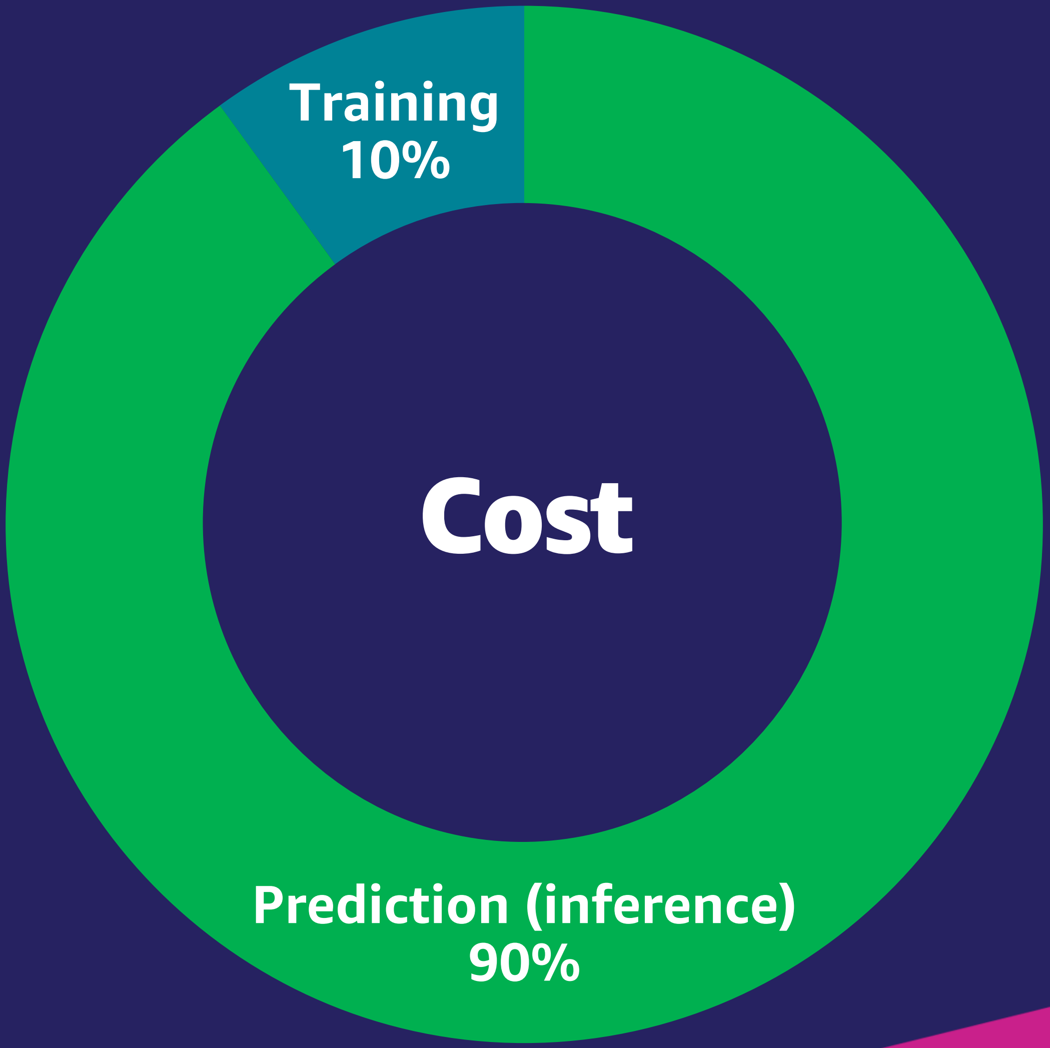
Supported Algorithms

Algorithms	Description
XGBoost	A supervised learning algorithm that attempts to accurately predict a target variable by combining an ensemble of estimates from a set of simpler and weaker models.
MLP	Neural-network based deep learning algorithms for problems with multi-dimensional, multi-class datasets, such as sales forecasting, recommendation systems, call center routing, and advertisement optimization.
KMEANS	Unsupervised learning for use cases such as customer segmentation

Amazon Redshift ML: Optimized for cost

Typically predictions drive cost in production

You only pay for training while **prediction comes at no extra cost** when you use Amazon Redshift ML



Cost controls for training

Optionally specify max_cells (number of rows * number of columns) selected in the training query

If training data produced by *query* exceed max_cells, Amazon Redshift automatically reduces training data without creating bias

Default max_cells (1M cells) keeps cost below <$20 out of the box

You can also set max_runtime to control your cost. Default value is 5400 seconds

```
CREATE MODEL customer_churn
FROM query
...
SETTINGS (
max_cells = 200000)
```

```
CREATE MODEL customer_churn
FROM query
...
SETTINGS (
MAX_RUNTIME 3000)
```

Demo

Machine Learning in Redshift

Additional Resources

- Redshift ML Blog
- Documentation
- GitHub Repository
- Unsupervised training with K-Means
- Regression model
- Multi-class classification
- XGBoost Model
- Bring Your Model for remote inference

Thank you!