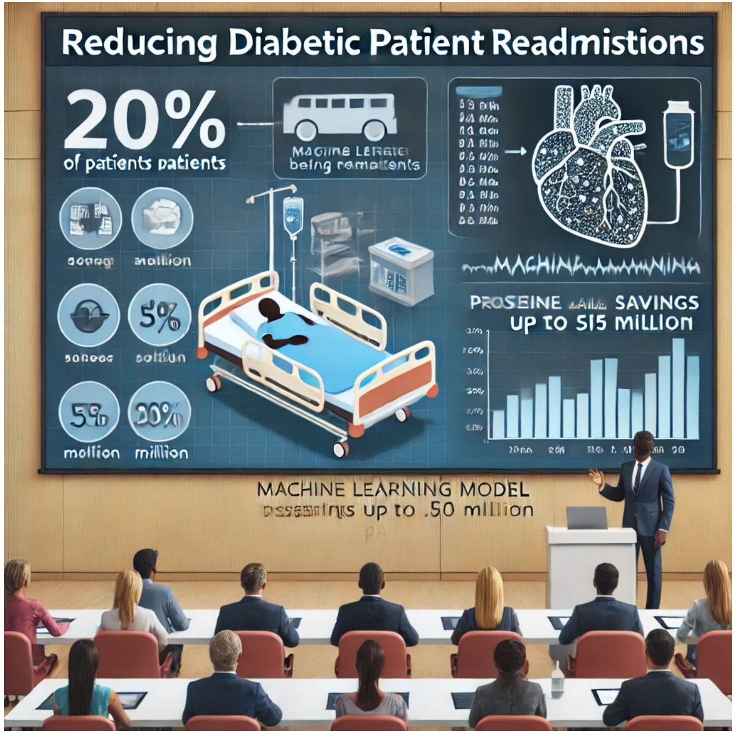
Diabetic Readmission Risk Prediction

Jonathan Agustin　　Zach Robertson　　Lisa Vo

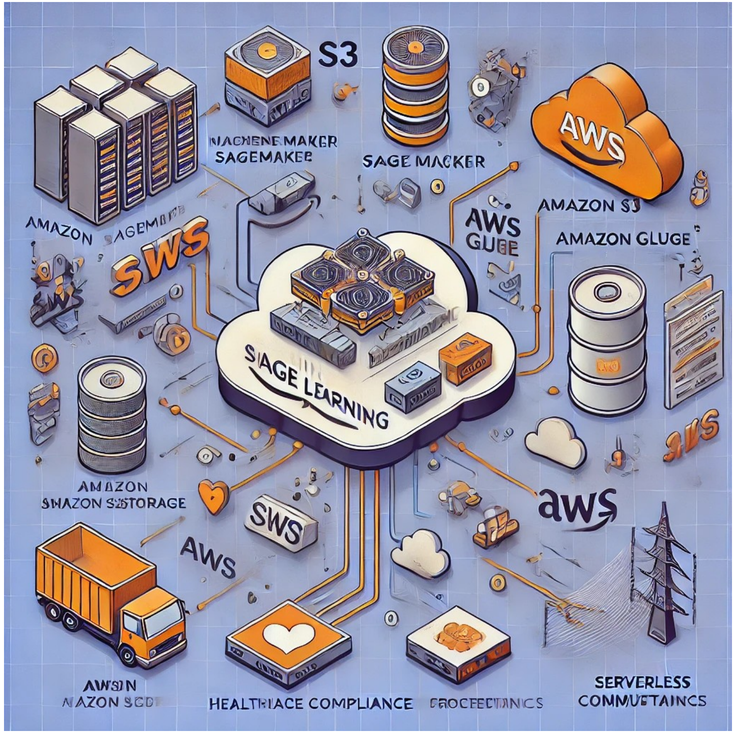
Slide 2

Hospitals face high costs due to diabetic patient readmissions, with 20% of patients returning within 30 days of discharge. Our solution uses machine learning to predict high-risk patients, enabling interventions that could reduce readmission rates by 10%, saving mid-sized hospitals up to $5 million annually and improving patient health outcomes.



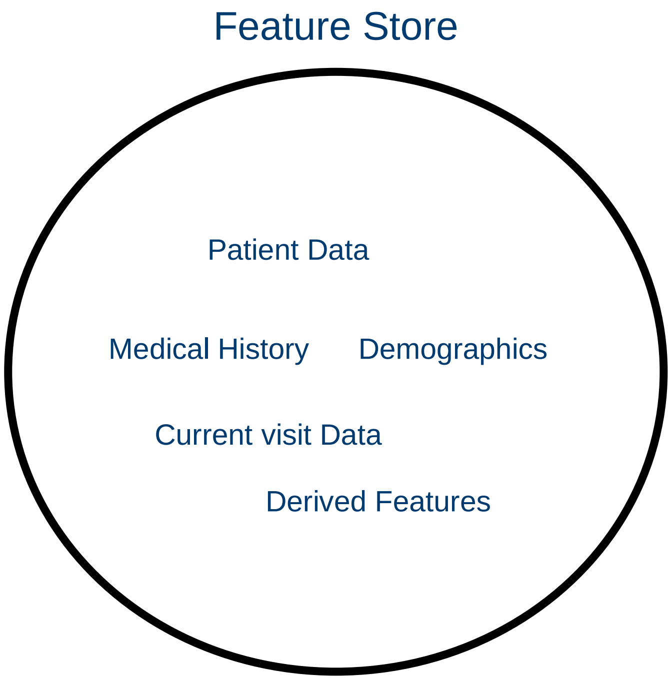
Slide 3

Our machine learning system architecture is built on AWS, using Amazon SageMaker for model training and deployment. Data is stored in Amazon S3, processed through AWS Glue, and computations are handled serverlessly via AWS Lambda. This design ensures scalability, regulatory compliance, and minimal maintenance.



Slide 4

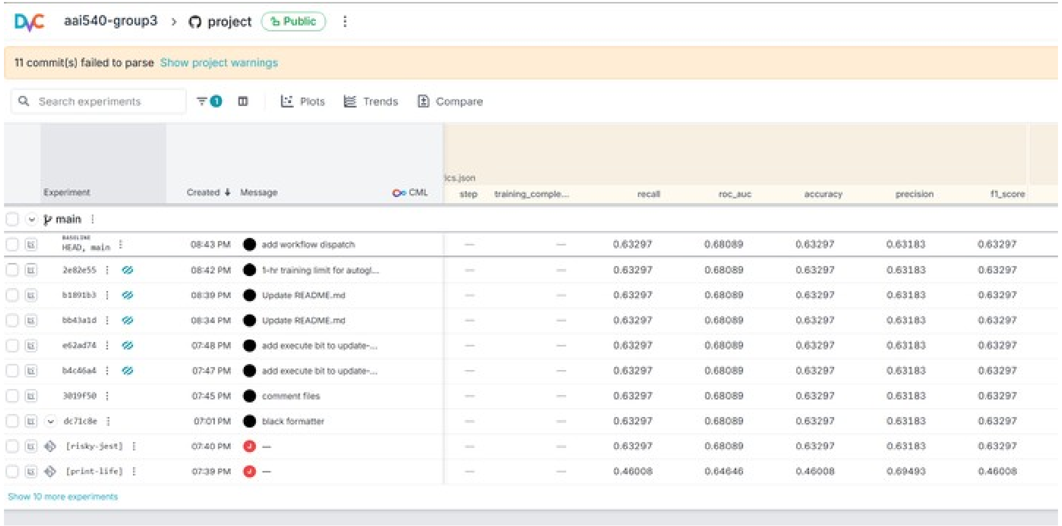
Our system uses the SageMaker Feature Store to organize key patient data into groups such as demographics, medical history, current visit data, and derived features. This organization allows easy feature reuse, version control, and ensures reproducibility.



Slide 5

We manage our data using AWS SageMaker Feature Store, which organizes the data used by our machine learning model into several feature groups:

1. **Patient Demographics** – Includes age, gender, and ethnicity.
2. **Medical History** – Contains previous diagnoses and treatments.
3. **Current Visit Data** – Provides details about the patient's most recent hospital stay.
4. **Derived Features** – Newly calculated data points that enhance predictive accuracy.


aai540-group3 > project Public

11 commit(s) failed to parse Show project warnings

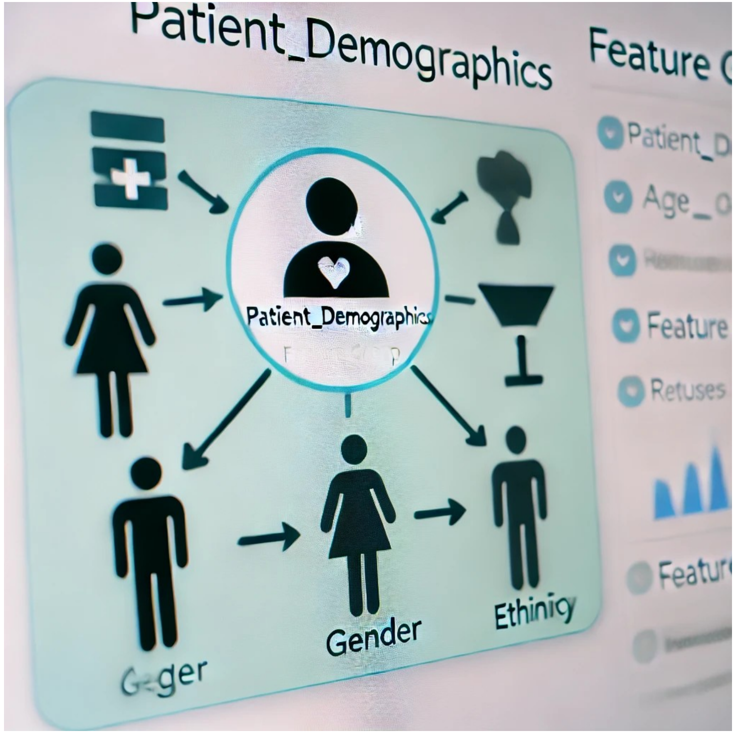
Search experiments Plots Trends Compare

Experiment	Created	Message	step	training_comple...	recall	roc_auc	accuracy	precision	f1_score
main									
BASELINE HEAD, main	08:43 PM	add workflow dispatch	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
2e82e55	08:42 PM	1-hr training limit for autogl...	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
b1891b3	08:39 PM	Update README.md	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
bb43a1d	08:34 PM	Update README.md	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
e62ad74	07:48 PM	add execute bit to update-...	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
b4c46a4	07:47 PM	add execute bit to update-...	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
3019f50	07:45 PM	comment files	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
dc71c8e	07:01 PM	black formatter	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
[risky-jest]	07:40 PM	—	—	—	0.63297	0.68089	0.63297	0.63183	0.63297
[print-life]	07:39 PM	—	—	—	0.46008	0.64646	0.46008	0.69493	0.46008

Show 10 more experiments

Slide 6

In our 'patient_demographics' feature group, data like age, gender, and ethnicity are stored. This approach promotes collaboration across teams and ensures feature consistency and trackability over time.



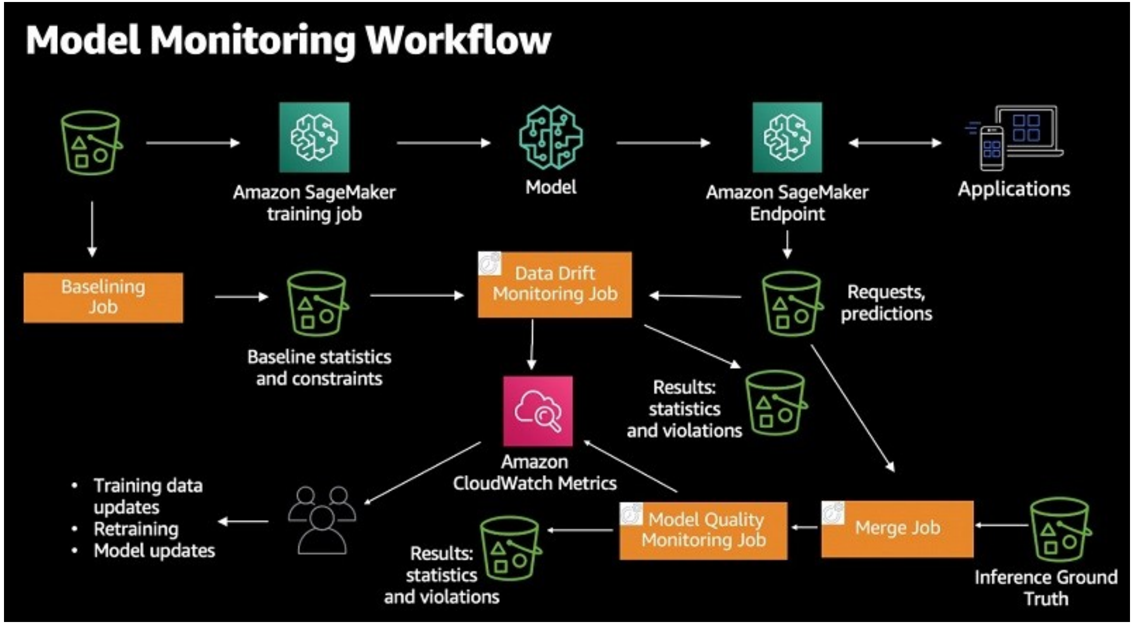
Slide 7

We monitor system performance using AWS CloudWatch, tracking metrics like CPU utilization, memory usage, and API latency. Alerts are set for predefined thresholds to ensure proactive issue resolution.

Amazon CloudWatch

Slide 8

SageMaker Model Monitor tracks prediction distribution, feature importance, and accuracy over time. This allows us to detect performance degradation and shifts in prediction patterns.



Slide 9

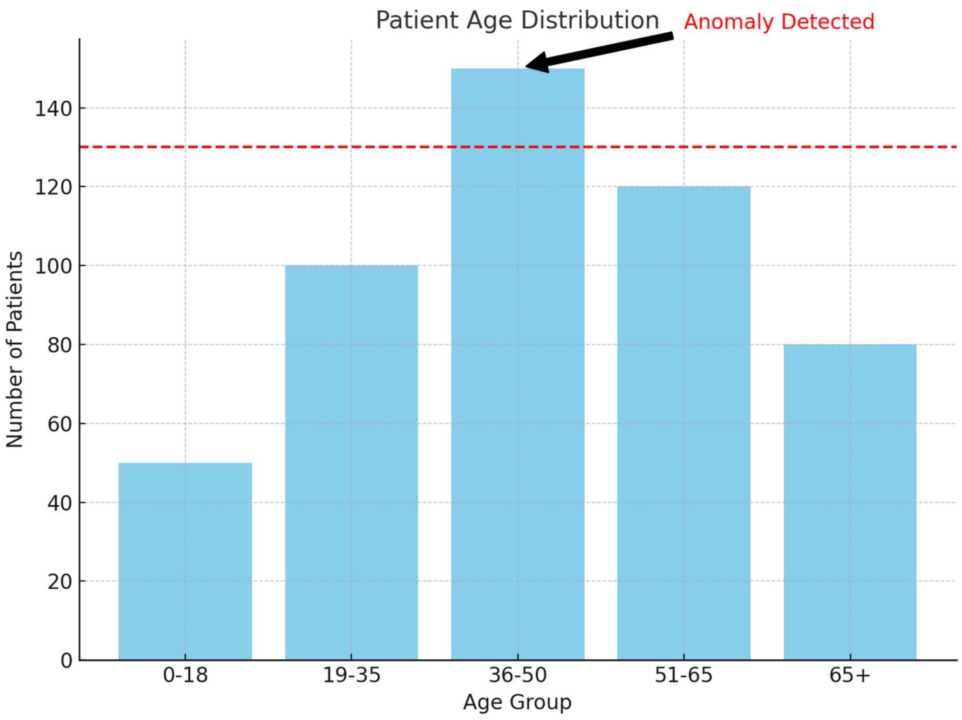
We use Amazon SageMaker Model Monitor to continuously track the performance of our machine learning model. This tool helps us in the following ways:

1. **Prediction Distribution:** Monitors if the model is predicting high risk for more (or fewer) patients than usual.
2. **Feature Importance:** Tracks changes in the importance of features, alerting us if the model starts relying too much on a single feature.
3. **Accuracy Over Time:** Detects any degradation in model performance, allowing us to take corrective action promptly.

Example

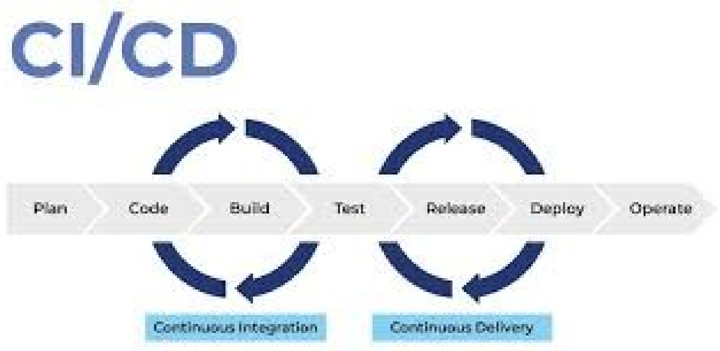
Slide 10

Custom data quality checks, such as monitoring patient age distribution, are in place to detect issues with the data ingestion process, ensuring reliable and accurate input for the model.



Slide 11

Our continuous integration and deployment pipeline, implemented with GitHub Actions, automates updates to the system, ensuring smooth and reliable model deployment.



Slide 12

Slide 13

Slide 14

Slide 15

Slide 16

Slide 18

Slide 19

Slide 20

Slide 21

Slide 22

Slide 23

Slide 24

Slide 25