SEATTLE PACIFIC
UNIVERSITY

ISM 6362 BIG DATA AND CLOUD TOOLS

Topic: Cricket

- Done By

Sheetal Murali

Abhi Munjal

Sunku Ganga Mahesh

CONTENTS

OVERVIEW

This project focuses on a data analytics startup that specializes in providing in-depth analysis of ICC World Cup cricket matches. The business leverages its expertise in data analytics to offer insights into player performances and statistics. Additionally, the business extends its analytical capabilities to understand customer behavior related to product sales, offering valuable insights into customer buying patterns, preferences, and demographics.

The startup's client base includes fans, sports enthusiasts, and organizations looking to optimize their operations based on data-driven decisions. By analyzing customer sales data, through this project we aim to enhance customer engagement, improve sales strategies, and maximize revenue during the ICC World Cup events.

OBJECTIVES:

- Develop a scalable and efficient data pipeline architecture using AWS services, Databricks and Azure ML
- Analyze match data to provide detailed player performance metrics
- Utilize customer data to Estimate the total sales generated by a customer based on their demographic information (e.g., age, gender, income level).

SCOPE:

The scope of this project is extensive, covering data collection, processing, analysis, and visualization for both match and customer data. The project aims to deliver actionable insights that will drive decision-making in player performance analysis and customer engagement strategies.

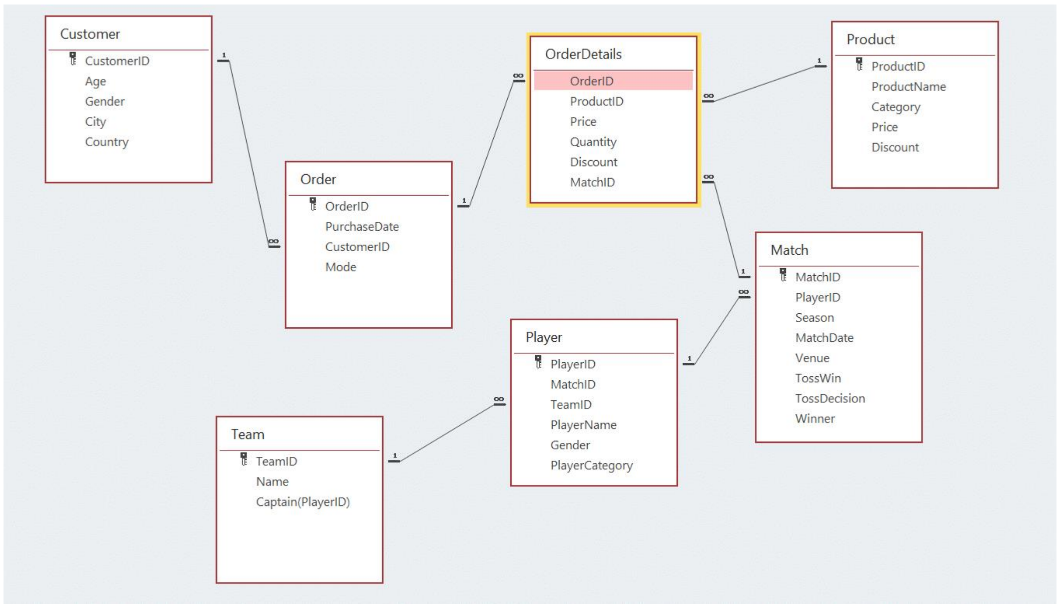
The data pipeline is designed to be scalable and adaptable, ensuring that it can meet evolving business needs and accommodate future growth.

DATA SOURCE

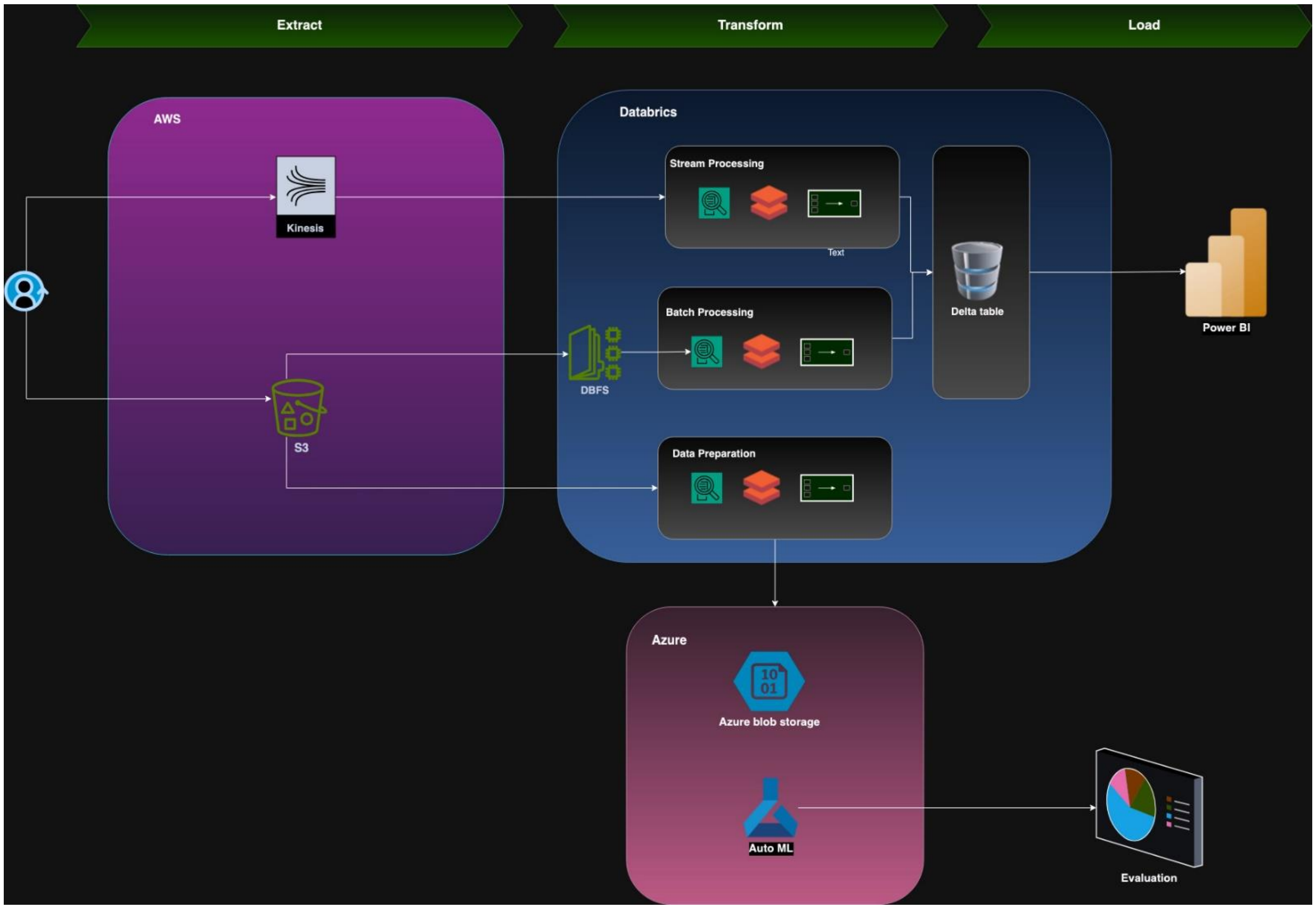
For the performance metrics analysis, we have utilized the cricket ball-by-ball dataset, which offers a comprehensive record of every ball bowled during a series of cricket matches, including detailed information about each delivery and its context. Given the extensive size of this dataset, our analysis will focus specifically on the 2023 ICC Men's Cricket World Cup season. The data has been sourced from the following website:

- Ball-by-ball data for ICC Men's Cricket World Cup 2023: CricSheet

For marketing analytics for building predictive models, we have employed simulated data that aligns with our conceptual framework, based on the following ER (Entity-Relationship) Diagram.



ARCHITECTURE DIAGRAM



The data pipeline architecture for our project is designed to manage the complete lifecycle of data processing, specifically tailored for the analysis of player performance in International Cricket as well as the evaluation of marketing strategies. This architecture is comprehensive, incorporating both batch and stream processing capabilities, leveraging the power of AWS services such as Kinesis. These components work in harmony to ensure the seamless ingestion, processing, and transmission of data across the pipeline.

At the core of the architecture lies a robust ETL (Extract, Transform, Load) process, which is capable of handling diverse data formats and volumes. For batch processing, data is periodically ingested from Amazon S3 to Databricks file system which is cleansed, transformed, and loaded into a cloud-based data storage solution which is the Delta table in Databricks. This storage serves as a central repository, enabling efficient data retrieval from Power BI for reporting.

For stream processing, the architecture utilizes AWS Kinesis to handle real-time data flows, capturing and processing data as it arrives. Once processed, we save it in the delta table for further analysis.

We used the raw customer data from S3 bucket and conducted data pre-processing in Databricks and then stored it in the delta table which was later extracted to conduct machine learning analysis in Microsoft azure.

On the visualization front, Power BI is integrated into the architecture to deliver interactive and dynamic dashboards. These dashboards enable the analysts to gain actionable insights by visualizing data trends and performance metrics.

DATA FLOW

EXTRACT: DATA INGESTION

1. **Data Sources**

Analyst (User Interaction):

Data originates from user action. In this case we have uploaded (Files of all matches 48 matches for Season 2023) CSV files to a folder in S3 bucket. Additionally, we have also run a Python script that processes the ICC Men’s Worldcup final match CSV file to AWS Kinesis. The script reads the CSV file using pandas, converting each row (ball-by-ball data) into a dictionary format (record) to Kinesis Data stream.

2. Data Storage:

AWS S3 (Simple Storage Service):

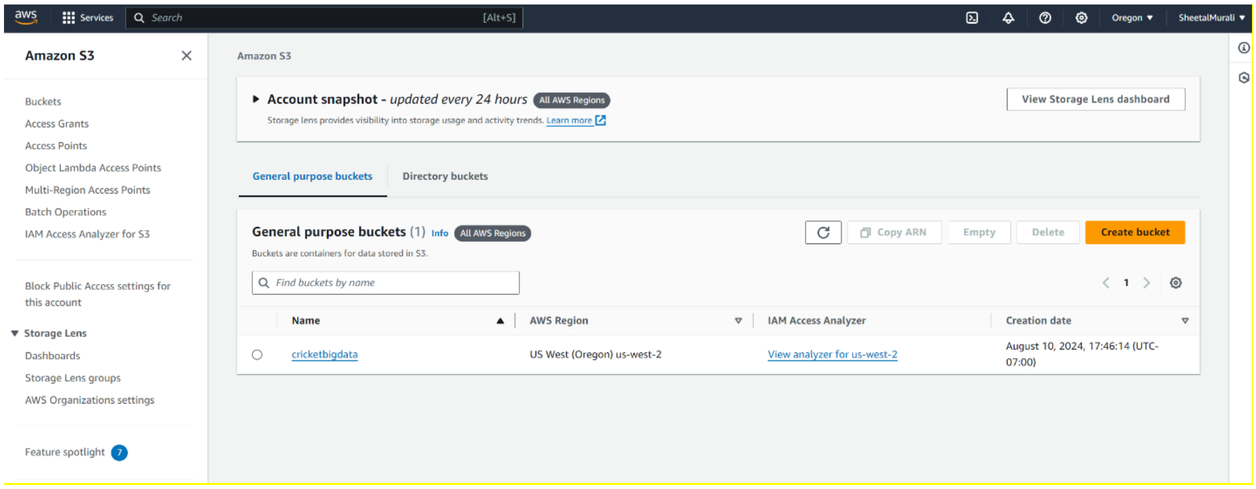
All Matches Data is initially stored in an S3 bucket. S3 acts as a staging area for raw data. Storing the CSV in S3 allows the data to be available for multiple processes. For our project, we have used this file to retrieve a batch processing job in Databricks, which processes large amounts of data.

3. Ingestion Tools:

The ingestion tool in this pipeline AWS Kinesis and AWS SDK for python (Boto3). For Stream processing, AWS Kinesis supports real-time data ingestion efficiently into Databricks as new data is being added to the data stream. For Machine Learning model, Databricks reads the data directly from S3 using 'spark.read' API. The data is loaded into Spark DataFrames for processing.

TRANFORM: DATA PROCESSING

4. **Batch Processing:**



Storage lens provides visibility into storage usage and activity trends. Learn more

General purpose buckets (1) Info All AWS Regions

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
cricketbigdata	US West (Oregon) us-west-2	View analyzer for us-west-2	August 10, 2024, 17:46:14 (UTC-07:00)

The raw data in S3 is ingested into Databricks, where it is stored temporarily in DBFS (Databricks File System). Data in DBFS can be easily accessed and processed by Databricks notebooks, jobs, and workflows, making it highly efficient for transformation tasks. Databricks then processes the data, performing necessary transformations.

Tools Used: Apache Spark within Databricks.

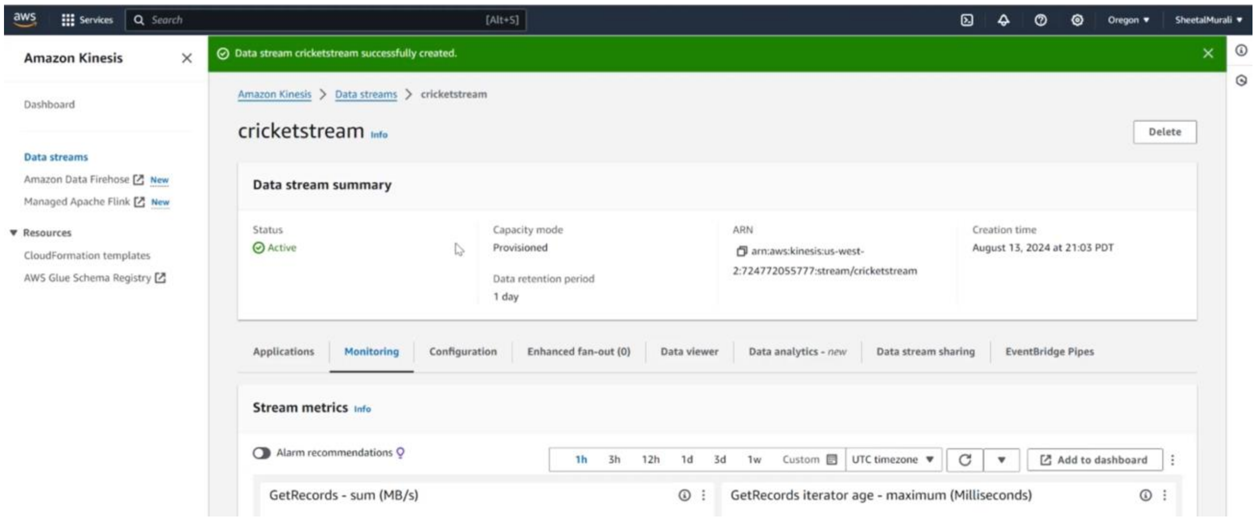
Purpose: To aggregate, clean, and transform data in batches, making it ready for Visualization and reporting.

5. **Stream Processing:**

Stream processing is performed in real-time, processing data as it arrives. The process begins with the user running a Python script that processes the CSV file as mentioned above. The script initializes a boto3 client to interact with AWS services. The script reads the CSV file using pandas, converting each row into a comprehensive record of every ball bowled in a series of cricket matches, providing detailed information about each delivery and the context in which it occurred.

AWS Kinesis:

First step is to create a data stream, and we created “cricketstream” as shown in the image below and then ran the script to ingest data.



Data stream cricketstream successfully created.

cricketstream

Data stream summary

Status: Active

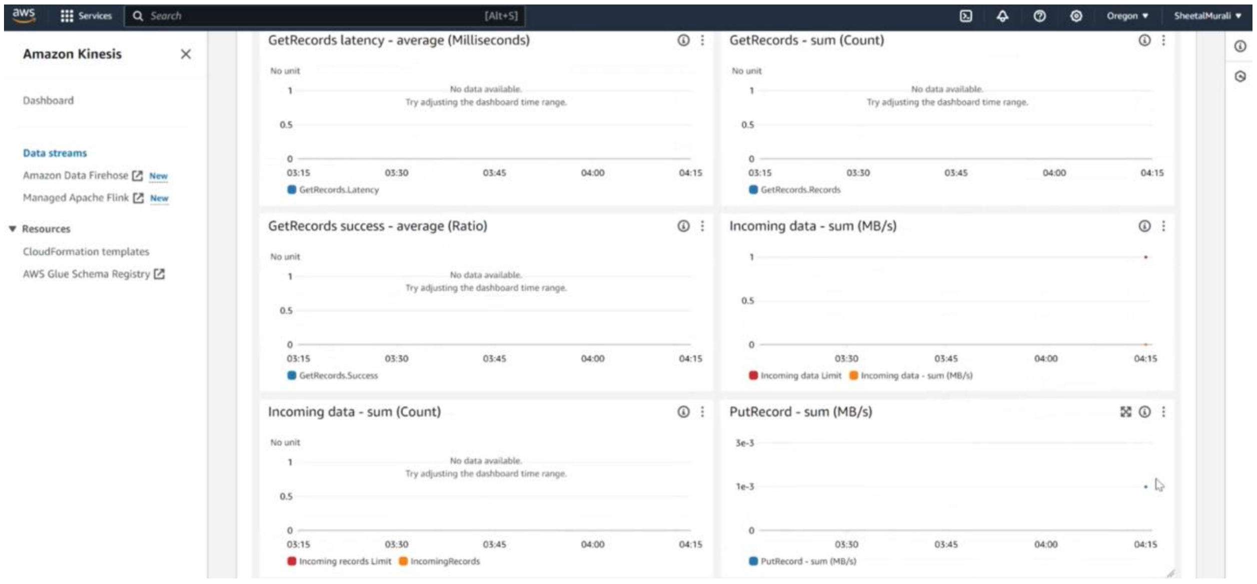
Capacity mode: Provisioned

Data retention period: 1 day

ARN: arn:aws:kinesis:us-west-2:724772055777:stream/cricketstream

Creation time: August 13, 2024 at 21:03 PDT

The script iterates over each row in the CSV, converting it into a JSON message. Each row (as a JSON object) is sent to AWS Kinesis, enabling real-time data streaming. We can see the incoming data in the image below. Kinesis streams the data real-time, allowing us to capture and process data as it is generated.



Databricks, with its seamless integration with Apache Spark, provides a robust platform for real-time data analytics. With the AWS credentials and stream settings configured, we can establish a connection to the Kinesis stream. The connection is set up using Spark Structured Streaming, which allows Databricks to read data from the Kinesis stream continuously. Once the data is being read from the Kinesis stream, the next step is to write this streaming data to a Delta table in Databricks.

Data Storage in Delta Table:

The processed data from both batch and stream processing is stored in Delta tables, which are optimized for performance and reliability.

LOAD: DATA VISUALIZATION AND REPORTING

Power BI:

Power BI is used for data visualization and reporting. It connects to the Delta Tables to create interactive dashboards and reports for end-users, allowing them to derive insights from the processed data. We ensured that the JDBC driver for Databricks was available, allowing us to access the Delta tables via a JDBC connection.

We launched Power BI Desktop on our local machines. In the Get Data Menu, We input the connection details, including the server URL, HTTP path, and credentials, and clicked "Connect" to establish a connection to Databricks.

Once connected, we selected the Delta tables we wanted to import into Power BI and loaded the data.

ANALYSIS

PLAYER PERFORMANCE:

The objective of this analysis was to aggregate and analyze raw player performance data using Databricks. We aimed to derive meaningful insights by cleaning and aggregating the data, ultimately visualizing key performance metrics related to cricket players.

1. Data Aggregation and Cleaning:

- **Data Cleaning:** Removed the inaccuracies, duplicate information/ irrelevant information to ensure the data quality. For example, there was an extra column which had same string value throughout.
- **Data Aggregation:** Combining and summarizing data from multiple sources to prepare it for meaningful analysis.

2. Performance Metrics Visualization:

After cleaning and aggregating the data from batch processing, we created tables in Databricks to help visualize the following performance metrics:

1. **Top 5 Scorers:**
    - **Description:** Identified the top 5 strikers based on their highest scores.
2. **Top 5 Bowlers:**
    - **Description:** Determined the top 5 bowlers with the highest number of wickets.
3. **Batsman Team Association:**

    - **Description:** For any given batsman, identified the team they belong to.

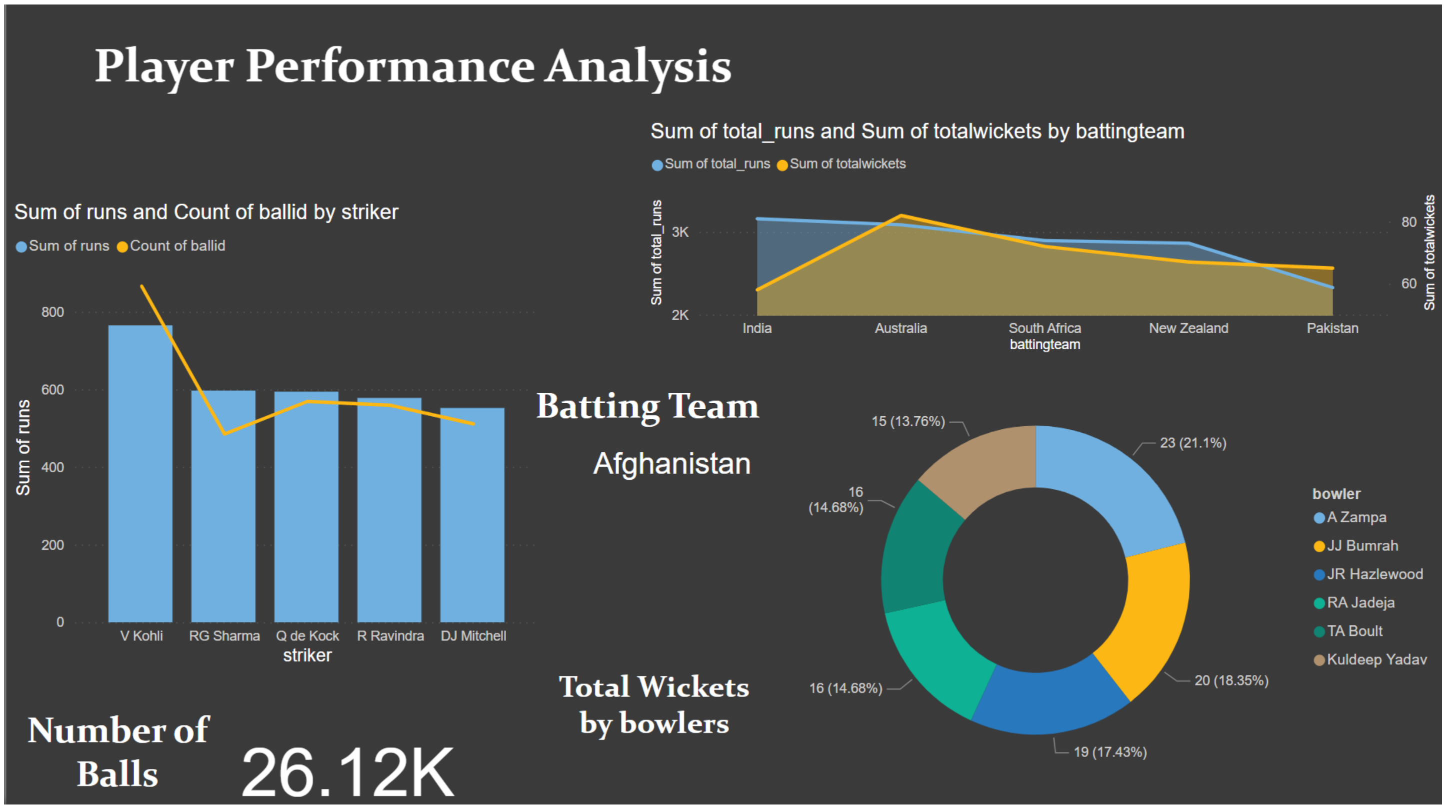
4. **Balls Bowled for Total Score:**
    - **Description:** Calculated the number of balls a given batsman required to achieve their total score.

5. **Sum of Total Runs and Total Wickets for a Batting Team:**
    - **Description:** Summarized the total runs and wickets for a given batting team.

After cleaning and aggregating the data from Stream processing, we created tables in Databricks to help visualize the following performance metrics:

1. **Total runs by each over**
    - **Description**: Total runs scored by the batting teams up to the current ball.
2. **Economy Rate**
    - **Description**: Runs conceded by the bowler per over up to the current ball.
3. **Total wide balls given by bowlers**
    - **Description**: Sum of wide balls delivered in the match up to the current ball
4. **Strike Rate:**
    - **Description**: Percentage of the total runs divided by the total balls faced upto the current ball.

DATA VISUALIZATION



INTERPRETATION:

The interactive dashboard provides insights into player performance and team dynamics through various visualizations. Each chart is designed to highlight key metrics and facilitate in-depth analysis of player contributions.

1. **Line and Clustered Column Chart:**

Top Scorers: This chart displays the top 5 batsmen based on their individual scores. The key performers are: Virat Kohli: 765 runs from 866 balls, Rohit Sharma: 597 runs from 485 balls followed by Quinton de Kock, Ravindra Jadeja and Mitchell Starc

The column bars represent the total runs scored by each player, while the line graph illustrates the number of balls faced to achieve these scores. This visualization helps identify the top-performing batsmen and the efficiency with which they scored their runs.

2. Donut Chart:

Top Bowlers: This chart highlights the top 6 bowlers with the highest number of wickets. Notable performances include Adam Zampa: 23 wickets (highest percentage) on the first place and, Jasprit Bumrah with 20 wickets on the second followed by Hazlewood, and Jadeja and Boult with a tie. The donut chart provides a clear view of each bowler's contribution to the team's wicket tally, emphasizing their impact on the game.

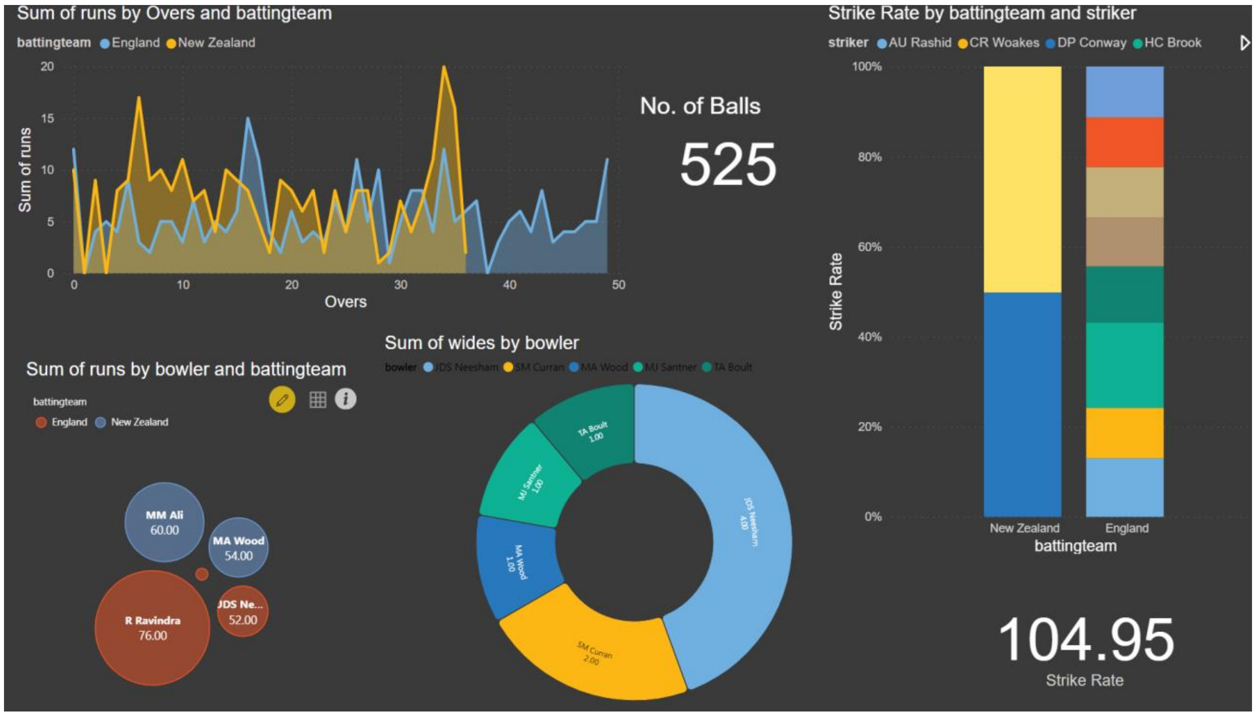
3. Area Chart:

Total Runs and Wickets per Team: This chart displays the total runs and wickets accumulated by each team. It provides a visual representation of team performance across matches.

Interactive Features:

The dashboard enables interactive exploration of player statistics. By clicking on any player, users can view detailed information about the team they belong to and the number of balls faced by that player. This feature enhances the understanding of individual contributions and team dynamics.

The interactive dashboard effectively visualizes player performance metrics and team statistics, offering valuable insights for performance analysis and strategic decision-making.



INTERPRETATION:

1. **Area Chart: Total runs by each over:**

   This chart displays the sum of runs scored by each team (England in blue, New Zealand in yellow) by the overs in the match. The Area for **New Zealand** shows peaks, indicating better performance, since they were able to win the match within 36 overs while England's performance shows a shower trend losing the match.

2. **Bar Chart: Strike Rate by Batting Team and Striker:**

   This stacked bar chart visualizes the strike rate of different players (strikers) for both New Zealand and England. The y-axis represents the strike rate, with 100% at the top. The different colored segments in each bar correspond to individual players, from this chart we can see that Ravindra has the highest strike rate of 125.5 and JE Root having the lowest strike rate of 89.53.

3. **Bubble Chart: Sum of Runs by Bowler and Batting Team**:

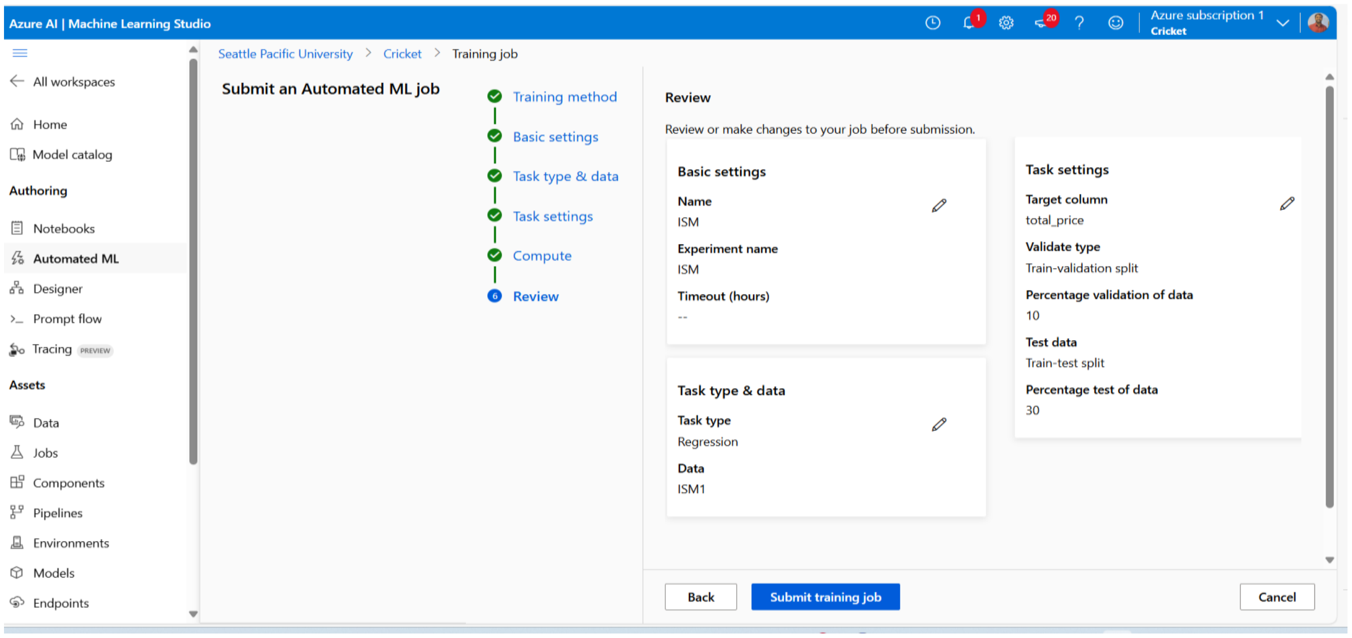
   This bubble chart shows the sum of runs conceded by each bowler, with different bubble sizes representing the total runs given. The chart is split by batting team, with England and New Zealand represented by different colors. The size of the bubbles (e.g., R Ravindra for New Zealand and MM Ali for England) reflects the amount of runs conceded.

4. **Donut Chart: Sum of Wides by Bowler:**

   This donut chart visualizes the number of wides bowled by different bowlers. The chart is segmented by individual bowlers with each segment's size representing the number of wides they have bowled. The largest segment represents Neesham with the most wides.

MACHINE LEARNING MODEL

The machine learning model used in this project is a regression model designed to predict the total sales price attributed to customers during the ICC Cricket World Cup. Azure's Automated Machine Learning (AutoML) was employed to automatically train and select the best-performing regression model.



Submit an Automated ML job

Review

Review or make changes to your job before submission.

Basic settings

Name
ISM

Experiment name
ISM

Timeout (hours)
--

Task type & data

Task type
Regression

Data
ISM1

Task settings

Target column
total_price

Validate type
Train-validation split

Percentage validation of data
10

Test data
Train-test split

Percentage test of data
30

MODEL TYPE AND CONFIGURATION

Model Type: Regression

The goal is to build a regression model that estimates the total sales generated by a customer based on their demographic attributes. By accurately predicting sales, the company can enhance its targeting strategies, improve customer retention, and ultimately increase revenue. Azure Machine Learning (Azure ML) provides a powerful platform to build, train, and deploy regression models, which can be highly beneficial for solving business problems like estimating total sales from customer demographics

Input Features: Customer demographics (age, gender, income, education) and match-related factors were used as input features.

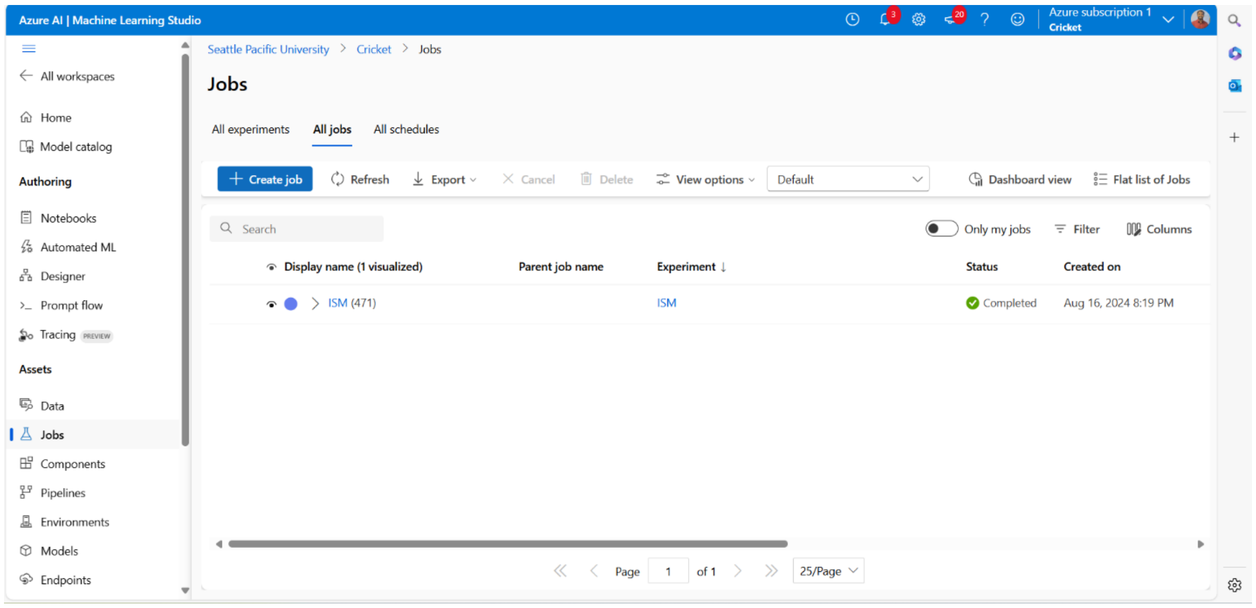
Target Variable: The target variable for the model was the total sales price, representing the monetary value generated from each customer.

Training Data: The dataset was split into training and testing sets using a train-validation split. 10% of the data was allocated for validation to evaluate the model's performance during the training process.

Model Training Process

Step 1: A workspace named "Cricket" was created in Azure Machine Learning Studio to host the project.

Step 2: An Automated ML job was initiated, titled "ISM," with the experiment name "AutoML_Cricket."



Display name (1 visualized)	Parent job name	Experiment ↓	Status	Created on
ISM (471)		ISM	Completed	Aug 16, 2024 8:19 PM

Step 3: The task type was set to Regression, as the goal was to predict a continuous variable (total sales price).

Step 4: The cleaned data from Databricks was uploaded to Azure Blob Storage and selected as the dataset for the model.

Step 5: Task settings included selecting the target column (total price) and specifying the validation type as train-validation split, with 10% of the data used for validation and 30% for testing.

Step 6: A compute instance was created in Azure ML to process the job.

Step 7: After reviewing the configuration, the job was submitted, allowing Azure AutoML to train multiple models and choose the best one.

The machine learning model was developed using Azure Machine Learning Studio, though the process could also be replicated using AWS Sagemaker. The model image represents the best-

Connections PREVIEW

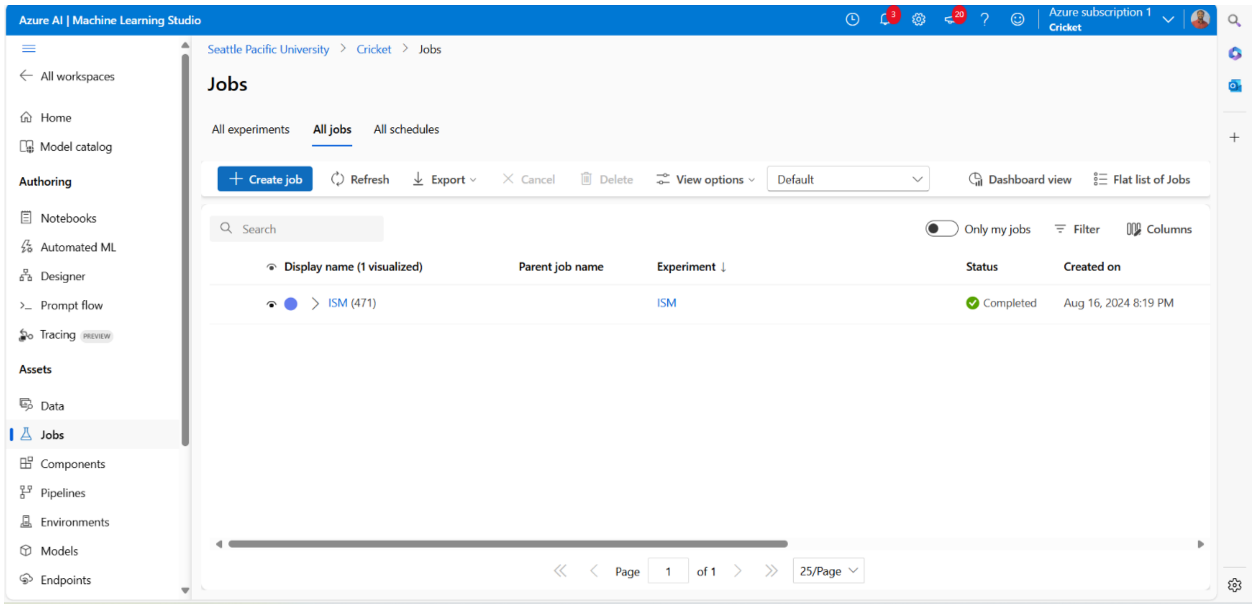
performing regression model selected by Azure AutoML after numerous iterations and optimizations.

Training Process

Test-Train Split: The dataset was divided into training and testing sets to evaluate the model's performance on unseen data.

Model Selection: Azure AutoML experimented with multiple regression models, such as Linear Regression and Decision Tree Regression. The model labeled amusing_kumquat_1 (3) was identified as the best-performing model based on its evaluation metrics.

MODEL EVALUATION


Azure AI | Machine Learning Studio

Azure subscription 1
Cricket

Seattle Pacific University > Cricket > Jobs

Jobs

All experiments · All jobs · All schedules

Default

	Display name (1 visualized)	Parent job name	Experiment ↓	Status	Created on
●	ISM (471)		ISM	Completed	Aug 16, 2024 8:19 PM

Page 1 of 1 · 25/Page

The evaluation of the selected model involved several key metrics:

Accuracy: The accuracy of the regression model was assessed based on how closely the predicted values matched the actual sales prices.

Root Mean Squared Error (RMSE): The RMSE for the best model was 503.5878, indicating the average prediction error.

ROC/AUC Curve: While typically used in classification tasks, similar concepts were applied here using regression-specific metrics such as NRMSE and R^2.

Key Metrics

The performance of the trained model was assessed using various metrics that provide insights into its accuracy and reliability:

Explained Variance: The explained variance for the selected model was 0.1793909, indicating that the model explains about 17.9% of the variance in the target variable. This low value suggests that the model may not capture all the complexities of the data.

Mean Absolute Error (MAE): The MAE for the model was 306.3583, which represents the average absolute difference between the predicted and actual sales prices. This error value indicates significant room for improvement in the model's predictions.

Mean Absolute Percentage Error (MAPE): The MAPE was calculated at 78.61313%, indicating that the model's predictions deviate from the actual values by an average of 78.6% in percentage terms.

Normalized Root Mean Squared Error (NRMSE): With a value of 0.07409647, NRMSE provides a normalized measure of the model's prediction errors relative to the range of the sales prices. The relatively low NRMSE suggests that the model performs reasonably well, though improvement is still possible.

Root Mean Squared Error (RMSE): The RMSE for the model was 503.5878, indicating the average magnitude of the model's prediction errors. The higher RMSE compared to MAE suggests that there are some significant errors in the predictions.

R² Score: The R² score for the model was 0.1793806, which means that the model explains only 17.9% of the variance in the data, suggesting that other factors may influence the total sales price that are not captured by this model.

Spearman Correlation: The Spearman correlation coefficient was 0.3790979, indicating a weak positive relationship between the predicted and actual sales prices. This metric suggests that the model's predictive power is limited.

Visualization of Results

To better understand the model's performance, several visualizations were created:

Predicted vs. True Plot: This plot compares the predicted sales prices against the actual values. Ideally, the points should align along the diagonal line, representing perfect predictions. The deviations observed in the plot indicate that the model's predictions are not entirely accurate.

Residuals Histogram: The residuals histogram displays the distribution of prediction errors (residuals). Most errors are clustered around certain values, but the presence of larger residuals suggests that the model occasionally makes significant prediction errors.

Comparative Analysis

Among the various models evaluated by Azure AutoML, amusing_kumquat_l (3) was selected as the best-performing model. This selection was based on a comprehensive evaluation of the metrics. The model's low NRMSE and better overall metric performance made it the optimal choice for deployment.

CONCLUSION

The regression model developed to predict total sales based on customer demographics offers initial insights but highlights significant opportunities for improvement. The model highlights several key factors that affect sales, though it also suggests that other unaccounted-for variables plays a significant role.

The model’s performance suggests that interactions between demographic variables—such as the combined effect of age and income is crucial in understanding sales patterns. The performance of the regression model suggests that sales is influenced by non-linear relationships between demographic factors and other variables.

The current model's limitations highlight the importance of feature engineering. Creating derived features, such as customer lifetime value (CLV) could improve the model’s ability to predict sales more accurately. This, in turn, will enable the company to optimize its sales strategies, marketing efforts, and operational decisions.