

CAR SALES DATA ANALYSIS PROJECT

Using Azure Data Factory and Databricks
Presented by: Aditya Kangude

OBJECTIVE

- The primary objective of this project is to demonstrate the application of various Azure data engineering services to manage, process and analyze car sales data
- The goal is to showcase the efficient handling of large datasets, perform complex transformations and derive meaningful business insights

year	make	model	trim	body	transmissi	vin	state	condition	odometer	color	interior	seller	mmr	sellin	gprice	saledate				
2015	Kia	Sorento	LX	SUV	automatic	5xyktca69	ca	5	16639	white	black	kia motors	20500	21500	Tue Dec 16 2014 12:30:00 GMT-0800 (PST)					
2015	Kia	Sorento	LX	SUV	automatic	5xyktca69	ca	5	9393	white	beige	kia motors	20800	21500	Tue Dec 16 2014 12:30:00 GMT-0800 (PST)					
2014	BMW	3 Series	328i SULEV	Sedan	automatic	wba3c1c5	ca	45	1331	gray	black	financial s	31900	30000	Thu Jan 15 2015 04:30:00 GMT-0800 (PST)					
2015	Volvo	S60	T5	Sedan	automatic	yv1612tb4	ca	41	14282	white	black	volvo na r	27500	27750	Thu Jan 29 2015 04:30:00 GMT-0800 (PST)					
2014	BMW	6 Series G	650i	Sedan	automatic	wba6b2c5	ca	43	2641	gray	black	financial s	66000	67000	Thu Dec 18 2014 12:30:00 GMT-0800 (PST)					
2015	Nissan	Altima	2.5 S	Sedan	automatic	1n4al3ap1	ca	1	5554	gray	black	enterprise	15350	10900	Tue Dec 30 2014 12:00:00 GMT-0800 (PST)					
2014	BMW	M5	Base	Sedan	automatic	wbsfv9c51	ca	34	14943	black	black	the hertz c	69000	65000	Wed Dec 17 2014 12:30:00 GMT-0800 (PST)					
2014	Chevrolet	Cruze	1LT	Sedan	automatic	1g1pc5sb2	ca	2	28617	black	black	enterprise	11900	9800	Tue Dec 16 2014 13:00:00 GMT-0800 (PST)					
2014	Audi	A4	2.0T Prem	Sedan	automatic	wauffafl3e	ca	42	9557	white	black	audi missi	32100	32250	Thu Dec 18 2014 12:00:00 GMT-0800 (PST)					
2014	Chevrolet	Camaro	LT	Convertible	automatic	2g1fb3d37	ca	3	4809	red	black	d/m auto	26300	17500	Tue Jan 20 2015 04:00:00 GMT-0800 (PST)					
2014	Audi	A6	3.0T Prest	Sedan	automatic	wauhgafc0	ca	48	14414	black	black	desert aut	47300	49750	Tue Dec 16 2014 12:30:00 GMT-0800 (PST)					
2015	Kia	Optima	EX	Sedan	automatic	5xxgm4a7	ca	48	2034	red	tan	kia motors	15150	17700	Tue Dec 16 2014 12:00:00 GMT-0800 (PST)					

PRIMARY DATASET

About the Dataset

- The dataset comprises of 3,00,000+ rows and 16 columns
- Columns include-
 - 1. year
 - 2. make
 - 3. model
 - 4. trim
 - 5. body
 - 6. transmission
 - 7. vin
 - 8. state
 - 9. condition
 - 10. odometer
 - 11. color
 - 12. interior
 - 13. seller
 - 14. mmr
 - 15. selling price
 - 16. saledate

Key Points to be noted

- This dataset is not clean so data cleaning has to be done
- The saledate column is not in the ideal format so it has to be transformed to a certain format for further data analysis

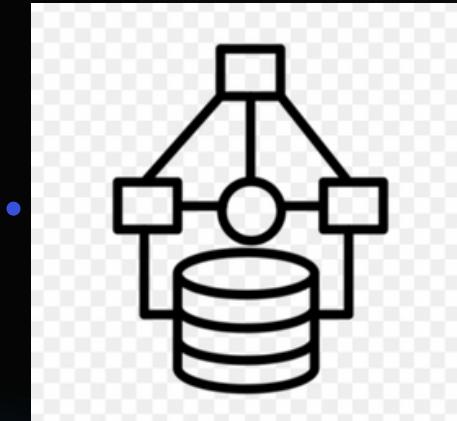
OVERVIEW



01

Data Cleaning and Transformation:

Extract, transform, and load (ETL) car sales data into a structured format using Databricks and Azure Data Factory.



02

Data Modelling

Create a robust data model by designing and implementing fact and dimension tables to facilitate efficient data querying and reporting.



03

Data Storage and Accessibility

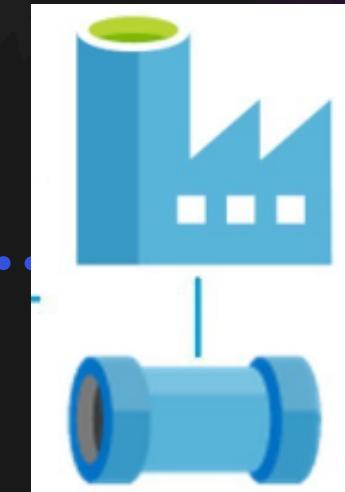
Convert and store the processed data in Delta tables and Parquet files for optimized storage and accessibility.



04

Data Loading

Load the processed data into an Azure SQL Database to enable advanced data analysis and integration with other business intelligence tools.



05

Data Handling

Implement dynamic data paths and pipeline variables in Azure Data Factory to handle data flexibly and efficiently.



06

Data Insights

Generate actionable insights from the transformed data to support business decisions and strategy formulation.

Data Modelling

SALES FACT TABLE

PK:fact_sales_id

FK:car_id

- vin
- yearMonth VARCHAR(6)
- total_sellingprice
DECIMAL(18, 2)
- mmr DECIMAL(18, 2)

CAR DIMENSION TABLE

PK:car_id

- vin
- make VARCHAR(50)
- model VARCHAR(50)
- color VARCHAR(50)
- body VARCHAR(50)

Azure Resources Used

Resources	
Recent	Favorite
Name	Type
 AdiDataBricksWS	Azure Databricks Service
 SQLDatabase (adiserver1234/SQLDatabase)	SQL database
 storagessample	Storage account
 adiserver1234	SQL server
 Demo1.1	Resource group
 AdiDF2	Data factory (V2)

Linked Services Used

Linked services

Linked service defines the connection information to a data store or compute. [Learn more](#)

+ New

 Filter by name

Annotations : **Any**

Showing 1 - 3 of 3 items

Name ↑↓	Type ↑↓	Related ↑↓	Annotations ↑↓
 AzureBlobStorageLS	Azure Blob Storage	1	
 AzureDatabricks1	Azure Databricks	2	
 AzureDataLakeStorage1	Azure Data Lake Storage Gen2	3	

Containers Used

storagessample | Containers ↗ ★ ...

Storage account

Search Container Change access level Restore containers Refresh Delete Give feedback

Overview Activity log Tags Diagnose and solve problems Access Control (IAM) Data migration Events Storage browser Storage Mover

Search containers by prefix Show deleted containers

Name	Last modified	Anonymous access level	Lease state
\$logs	5/8/2024, 12:33:08 PM	Private	Available
adftutorial	5/8/2024, 1:17:03 PM	Private	Available
finaldata	6/2/2024, 12:21:06 PM	Private	Available
intermediatedata	6/2/2024, 9:25:25 PM	Private	Available
rawcarprices	6/2/2024, 10:41:57 AM	Private	Available
web	5/10/2024, 9:21:02 AM	Private	Available

Data storage

Containers

ENTIRE PIPELINE



Datasets Used

▲ Datasets

4

✚ DeltaConvtoParquet

✚ FinalParquet

✚ IntermediateParquet

✚ SourceCSV

SourceCSV Dataset



DelimitedText
SourceCSV

Connection

Schema

Parameters

Linked service *

AzureBlobStorageLS

Test connection

Edit

New

Learn more

File path *

rawcarprices

/ Directory

/ car_prices.csv

Browse

Preview

Compression type

Select...

Column delimiter ⓘ

Comma (,)

SourceCSV Dataset

Connection **Schema** Parameters

Import schema

Clear

Column name

Type

year

String

make

String

model

String

trim

String

body

String

transmission

String

IntermediateParquet Dataset



Parquet

IntermediateParquet

Connection

Schema

Parameters

Linked service *

AzureDataLakeStorage1

Test connection

Edit

New

Learn more

File path

intermediatedata

/ Directory

/ File name

Browse

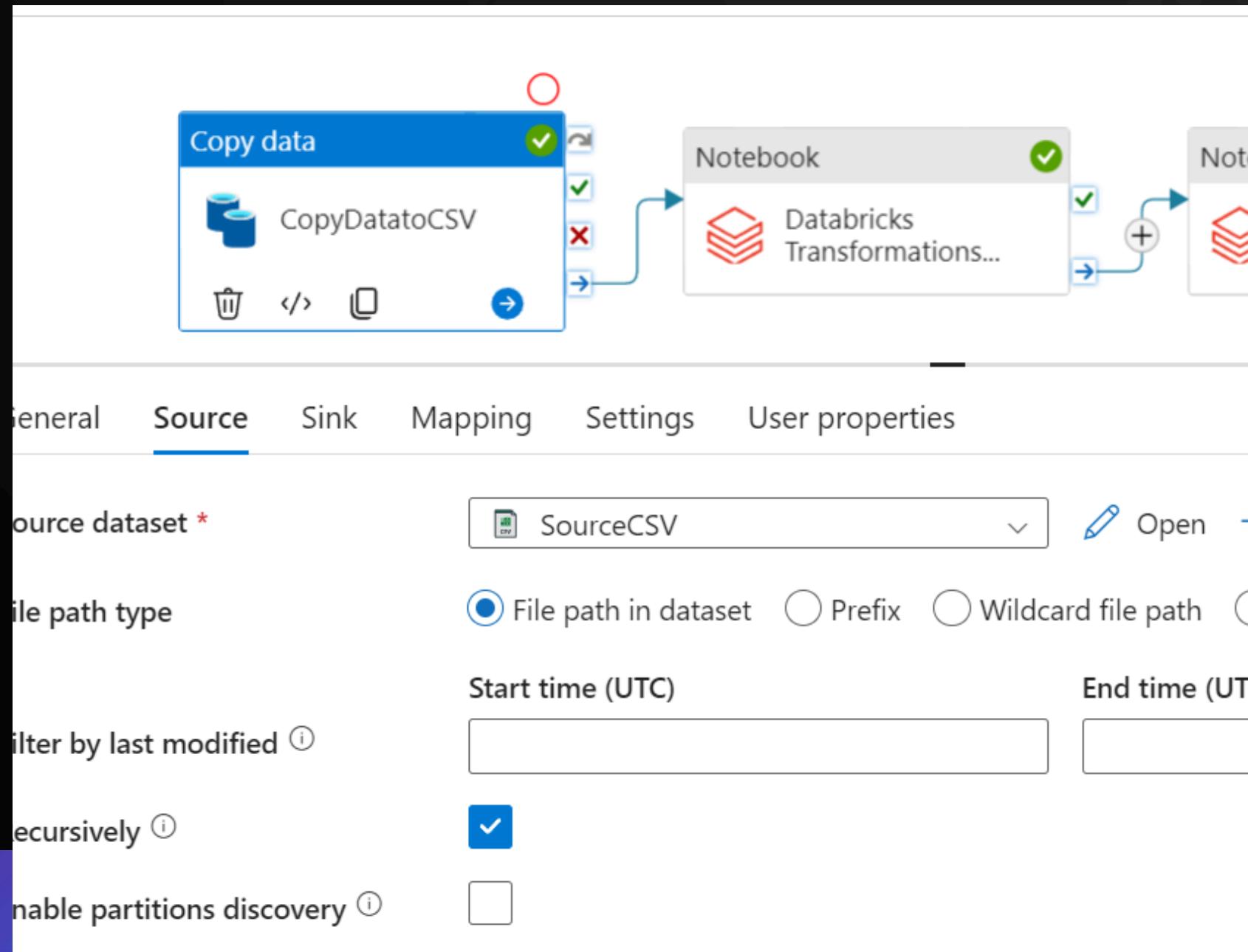
▾

Preview

Compression type

snappy

Copy Activity



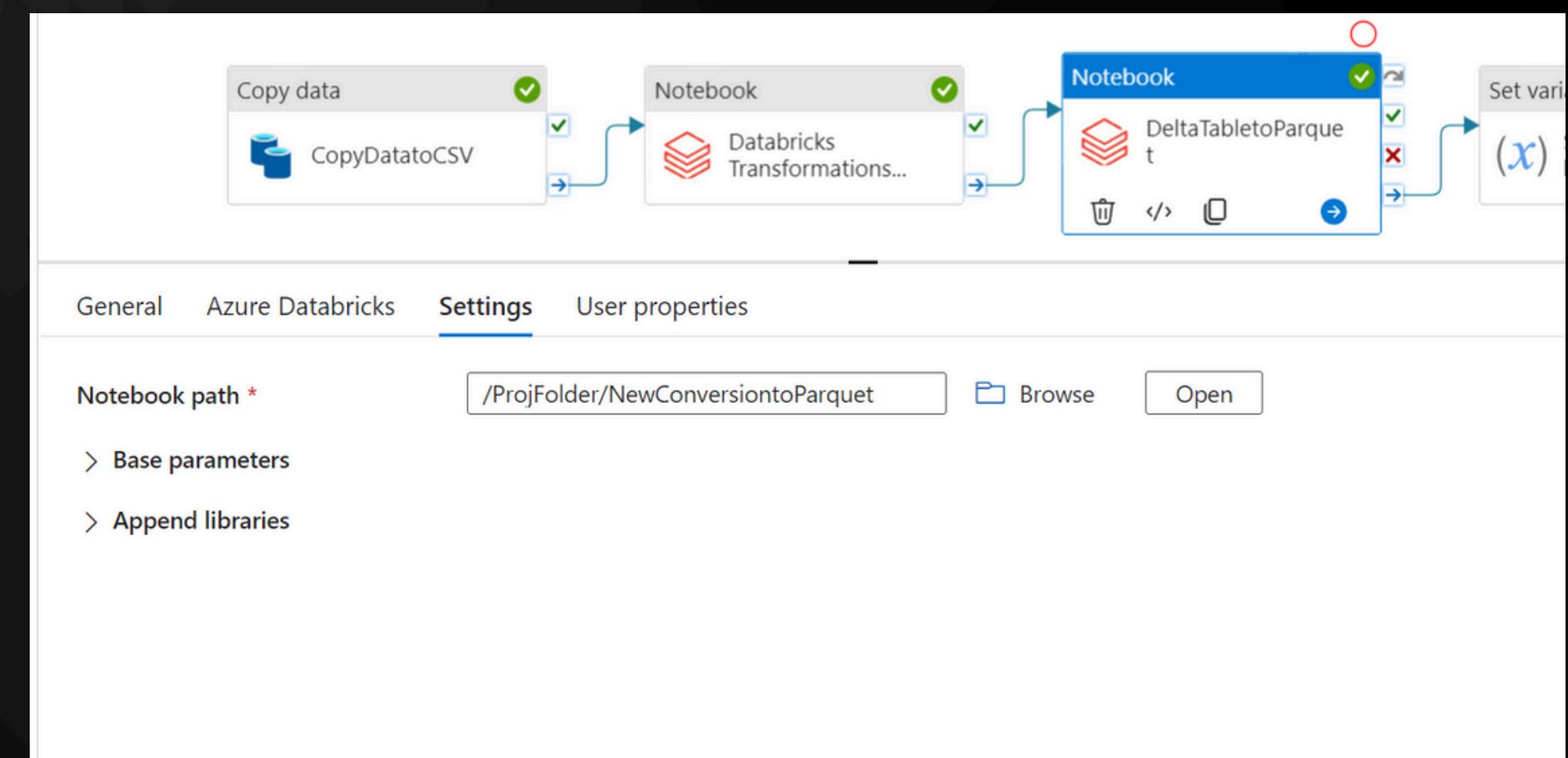
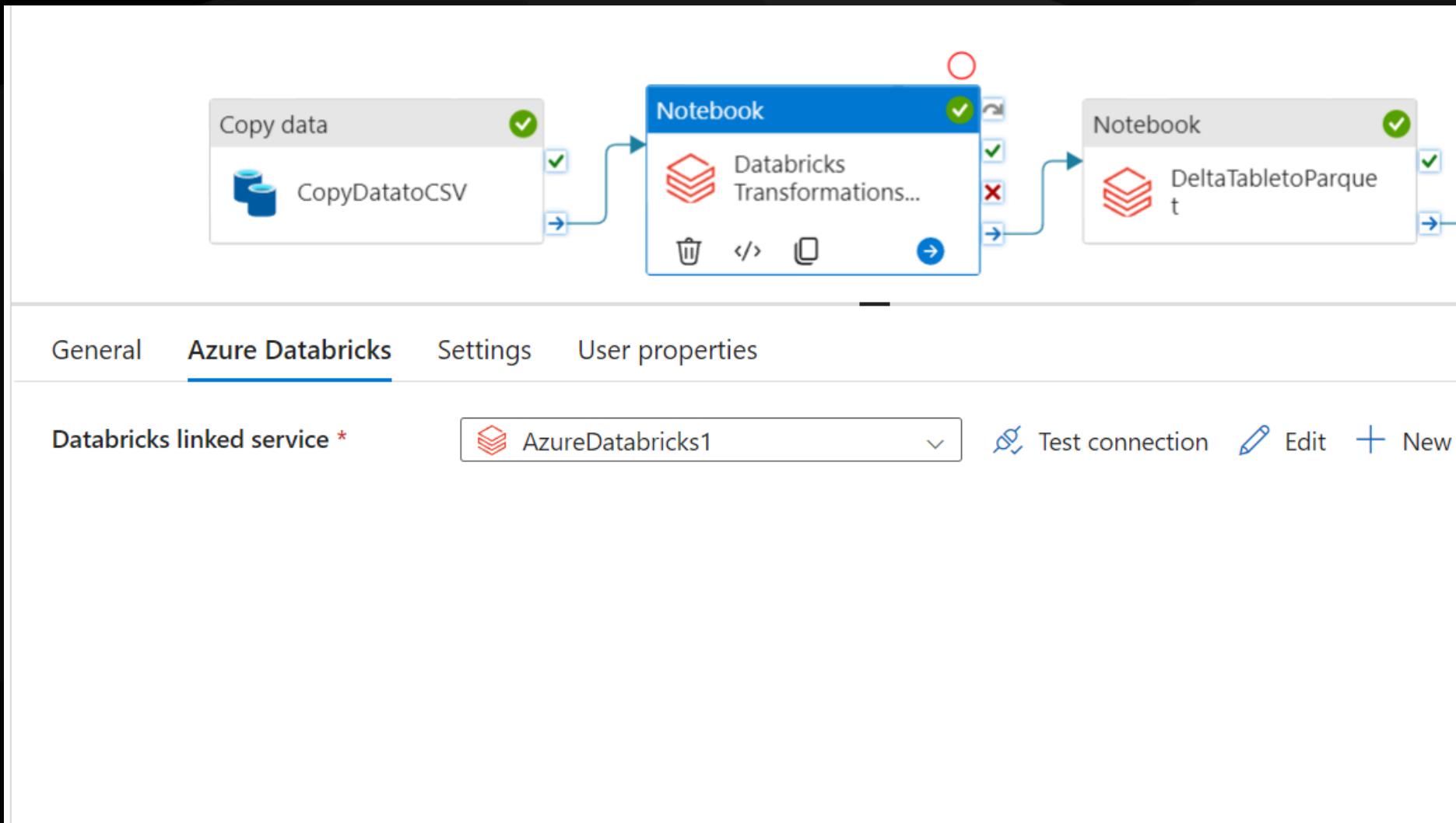
The screenshot shows the 'Sink' tab configuration for the 'Copy data' activity. The sink dataset is set to 'IntermediateParquet'. Other settings include 'Copy behavior' (set to 'Select...') and 'Max concurrent connections'.

General	Source	Sink	Mapping	Settings	User properties
Sink dataset *					
<input checked="" type="checkbox"/> IntermediateParquet					
Open New Learn more					
Copy behavior					
<input type="button" value="Select..."/>					
Max concurrent connections					

Databricks

Notebook

Activities



Getting Primary and Foreign Keys

SQLDatabase (MyBoiAdi)



i Showing limited object explorer here. For full capability please click here to open Azure Data Studio.

Tables

> dbo.DimCar

...

> dbo.FactSales

...

> Views

> Stored Procedures

Query 1 X **Query 2 X** Query 5 X Query 6 X Query 7 X

▶ Run Cancel query ⏴ Save query ⏴ Export data as Show only Editor

```
1
2   SELECT CONSTRAINT_NAME
3   FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS
4   WHERE TABLE_NAME = 'DimCar'
```

Results Messages

Search to filter items...

CONSTRAINT_NAME

PK_DimCar_4C9A0DB3813BDEBE

Getting Primary and Foreign Keys

The screenshot shows the Azure Data Studio interface. On the left, there's a sidebar with various tabs like 'Login', 'New Query', 'Open query', 'Feedback', and 'Getting started'. Below that is a 'SQLDatabase (MyBoiAdi)' section with a note about limited object explorer and a link to Azure Data Studio. The main area has tabs for 'Query 1' through 'Query 7', with 'Query 5' selected. The query editor contains the following T-SQL code:

```
1 SELECT CONSTRAINT_NAME  
2 FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS  
3 WHERE TABLE_NAME = 'FactSales'
```

The results pane shows two rows of data under the 'CONSTRAINT_NAME' column:

CONSTRAINT_NAME
PK_FactSale_17A06D843390D48D
FK_FactSales_car_i_5EBF139D

At the bottom, a yellow bar indicates 'Query succeeded | 1s'.

Table Successfully Loaded into SQL Database

Run Cancel query Save query Export data as Show only Editor

```
1 SELECT TOP (1000) * FROM [dbo].[DimCar]
```

Results Messages

Search to filter items...

vin	make	model	color	body
1ftkr4ee4bpa55734	Ford	Ranger	black	SuperCab
1f1431441070	Ford	Mustang	white	Coupe

Table Successfully Loaded into SQL Database

A screenshot of a SQL query editor interface. At the top, there are several buttons: Run, Cancel query, Save query, Export data as, and Show only Editor. Below these is a code editor window containing the following SQL query:

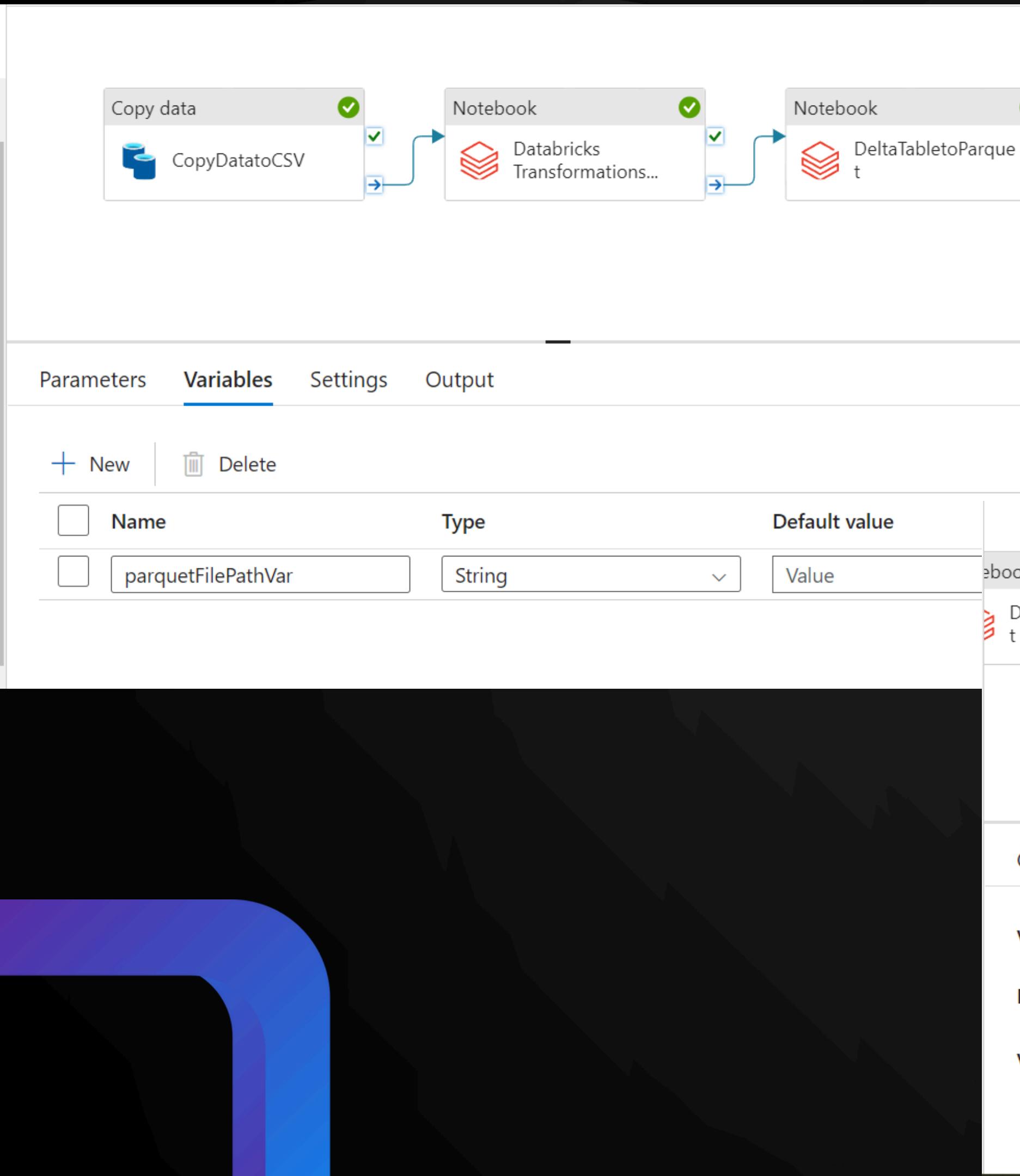
```
1 SELECT TOP (1000) * FROM [dbo].[FactSales]
```

The main area below the code editor is labeled "Results". It features a search bar with the placeholder "Search to filter items...". A table is displayed with the following columns: vin, YearMonth, total_selling_price, and mmr. One row of data is shown:

vin	YearMonth	total_selling_price	mmr
5npdh4ae7dh327127	2014-12	8700	11900

At the bottom of the interface, a yellow bar displays the message "Query succeeded | 1s".

Set Variable Activity



The screenshot shows the 'Settings' tab for the 'Set variable' activity. The 'Variable type' is set to 'Pipeline variable'. The 'Name' is 'parquetFilePathVar' and the 'Value' is '@activity('DeltaTabletoParquet').outp...'. There are also 'General' and 'User properties' tabs.

General **Settings** User properties

Variable type ⓘ

Pipeline variable Pipeline return value

Name *

parquetFilePathVar

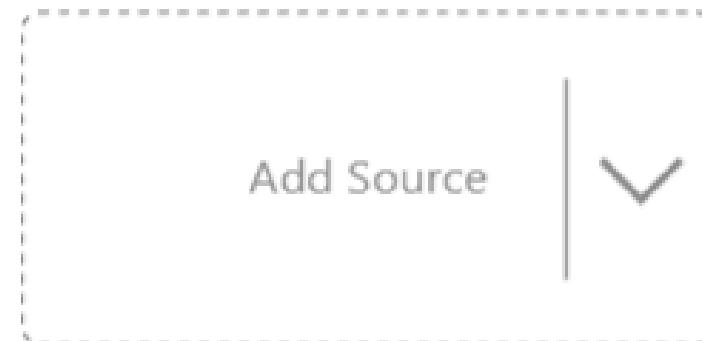
+ New

Value

@activity('DeltaTabletoParquet').outp...

Data Flow Activity

Validate Debug Settings



Data Flow Source

Source settings Source options Projection Optimize Inspect Data preview ●

Output stream name * [Learn more](#)

Description [Reset](#)

Source type * Dataset Inline

✓ Connection successful [Dataset](#) [Test connection](#) [Open](#) [New](#)

Options Allow schema drift ⓘ Infer drifted column types ⓘ Validate schema ⓘ

Sampling * ⓘ Enable Disable

Data Flow

Source Preview

Validate Data flow debug   Debug Settings

Source settings Source options Projection Optimize Inspect Data preview 

Number of rows  INSERT 100  UPDATE 0  DELETE 0  UPSERT 0  LOOKUP 0  ERROR 0 TOTAL

Refresh | Typecast  Modify  Map drifted Statistics Remove Export to CSV |

↑↓	make	abc ↑↓	Year...	abc ↑↓	color	abc ↑↓	body	abc ↑↓	condi...	abc ↑↓	total_...	1.2 ↑↓	mmr	1.2
+	ford		2014...		white		na		38		51250.0		11268...	
+	Nissan		2014...		gray		SUV		36		86850.0		14633...	
+	Toyota		2014...		blue		SUV		38		49800.0		17533...	
+	Chevr...		2015...		white		Sedan		28		30510...		7341....	
+	Jaquar		2014...		gray		Sedan		38		31500.0		30700.0	
+	Mazda		2014...		gray		Sedan		19		62851.0		6595....	
+	Honda		2014...		white		SUV		47		21000.0		17300.0	
+	MINI		2014...		gray		Hatch...		2		4600.0		5825.0	



✓ Validate Data flow debug Debug Settings

The data flow diagram consists of three main components: a Source component (Import data from DeltaConvtoParquet) which feeds into a FilterAccSellingPrice component (Columns: 7 total). This is followed by an aggregate1 component (Aggregating data by 'make, YearMonth' producing columns 'total_sellingprice'). Below the diagram are tabs for Filter settings, Optimize, Inspect, and Data preview.

Filter settings

Output stream name *

FilterAccSellingPrice

Description

Filtering rows using expressions on columns 'total_sellingprice'

Incoming stream *

Source

Filter on *

total_sellingprice >= 200000

Number of rows + INSERT 40 * UPDATE 0 × DELETE 0 *+ UPSERT 0 🔎 LOOKUP 0 ✘ ERROR 0 TOTAL 40

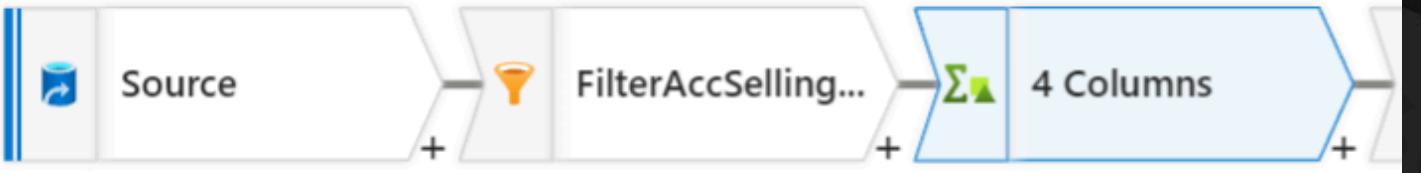
Refresh | Typecast | Modify | Map drifted | Statistics | Remove | Export to CSV |

↑↓	make	abc ↑↓	Year...	abc ↑↓	color	abc ↑↓	body	abc ↑↓	condi...	abc ↑↓	total_...	1.2 ↑↓	mmr	1.2 ↑↓
+	Chevr...		2015...		white		Sedan		28		30510...		7341...	
+	Toyota		2015...		silver		Sedan		19		59620...		6300....	
+	Jeep		2015...		red		SUV		4		37310...		21125.0	
+	Kia		2015...		white		SUV		5		89685...		19663...	
+	Chrysl...		2015...		blue		Sedan		4		30340...		14589...	
+	Ford		2015...		blue		SUV		49		27780...		27207...	

Data Insights -

Filter

✓ Validate Data flow debug   Debug Settings



Aggregate settings Optimize Inspect Data preview 

Output stream name *

Description 
Aggregating data by 'make, YearMonth'
producing columns 'total_sellingprice_sum,
avg_sellingprice'

Incoming stream *

Group by Aggregates

Columns	Name as
abc make	make
abc YearMonth	YearMonth

Data Insights - Aggregate

Aggregate settings Optimize Inspect Data preview 

Number of rows  **INSERT** 28  **UPDATE** 0  **DELETE** 0  **UPSERT** 0  **LOOKUP** 0  **ERROR** 0

 Refresh | Typecast  Modify  Map drifted  Statistics  Remove  Export to CSV |

↑↓	make	abc ↑↓	YearMonth	abc ↑↓	total_sellingprice_s...	1.2 ↑↓	avg_sellingprice
+	Chevrolet		2015-01		1105701.0		552850.5
+	Toyota		2015-01		596200.0		596200.0
+	Jeep		2015-01		777500.0		388750.0
+	Kia		2015-02		1401300.0		467100.0
+	Chrysler		2015-01		303400.0		303400.0
+	Ford		2015-01		974300.0		324766.6666666667

Data Insights- To Sink



Validate Data flow debug Debug Settings

Sink Settings Errors Mapping Optimize Inspect Data preview

Output stream name * Learn more [↗](#)

Description [Reset](#)

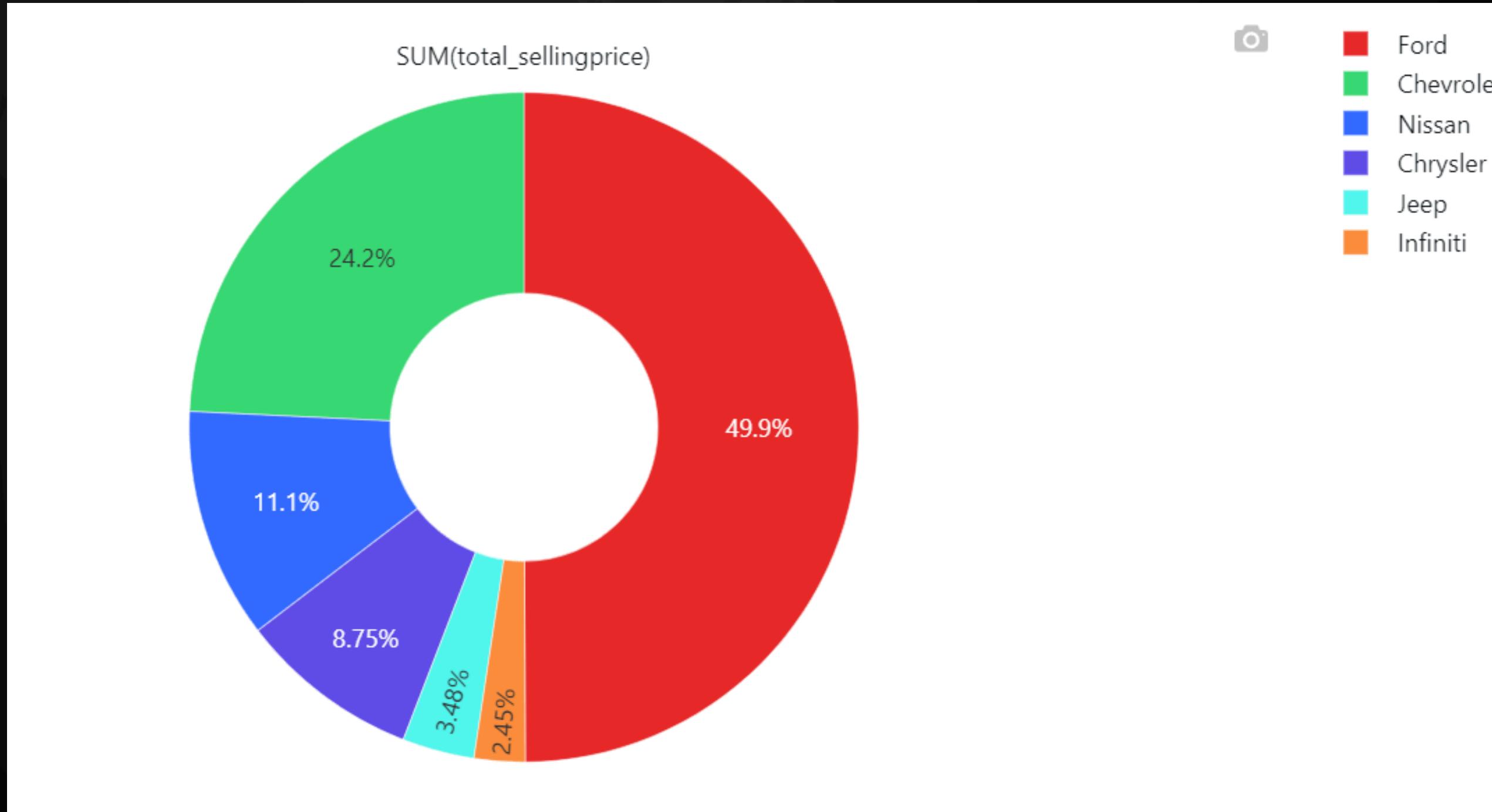
Incoming stream *

Sink type *
 Dataset Inline Cache

Dataset *
 FinalParquet [Test connection](#) [Open](#) [New](#)

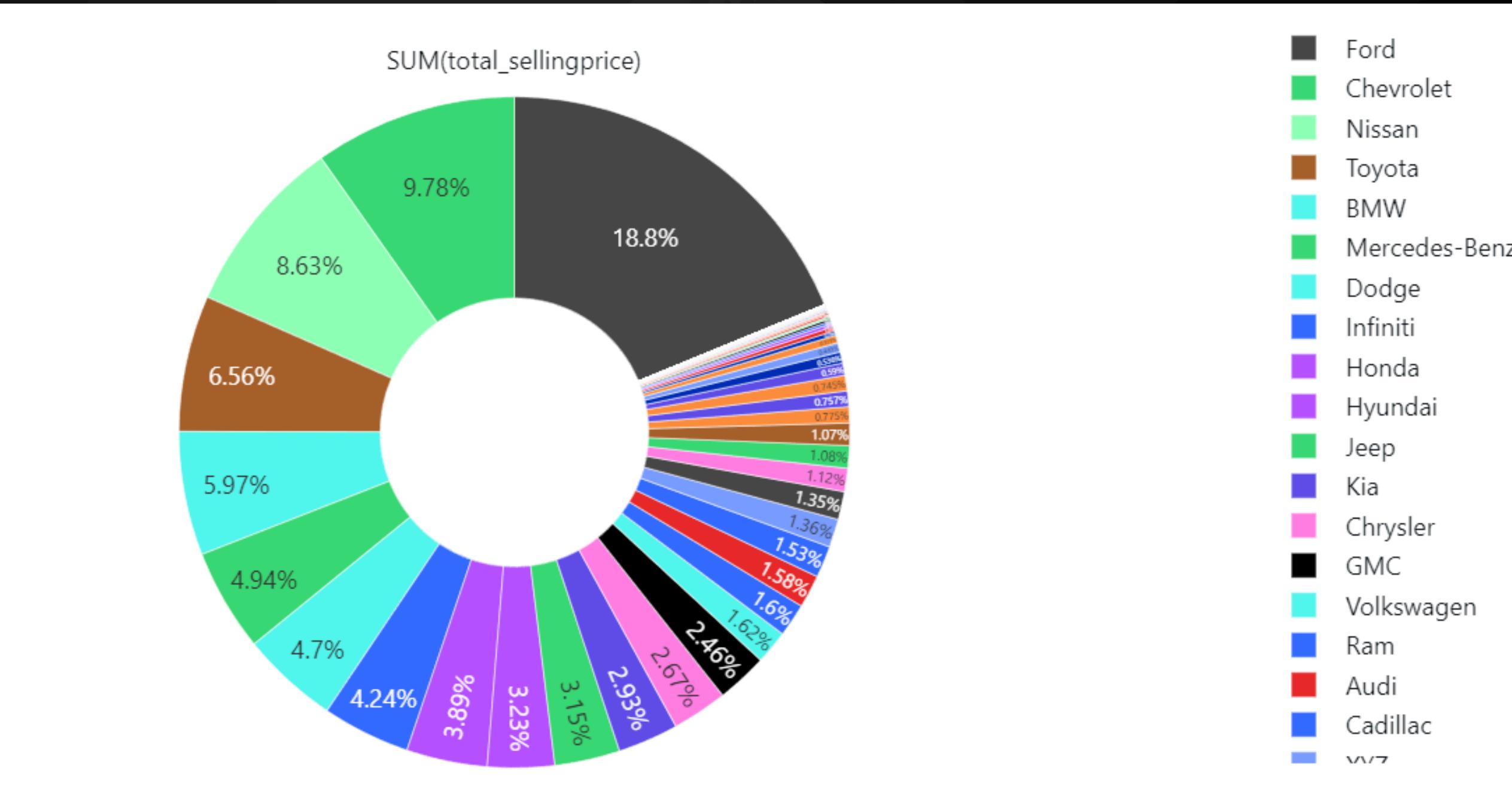
Options Allow schema drift [ⓘ](#)

Data Visualisations



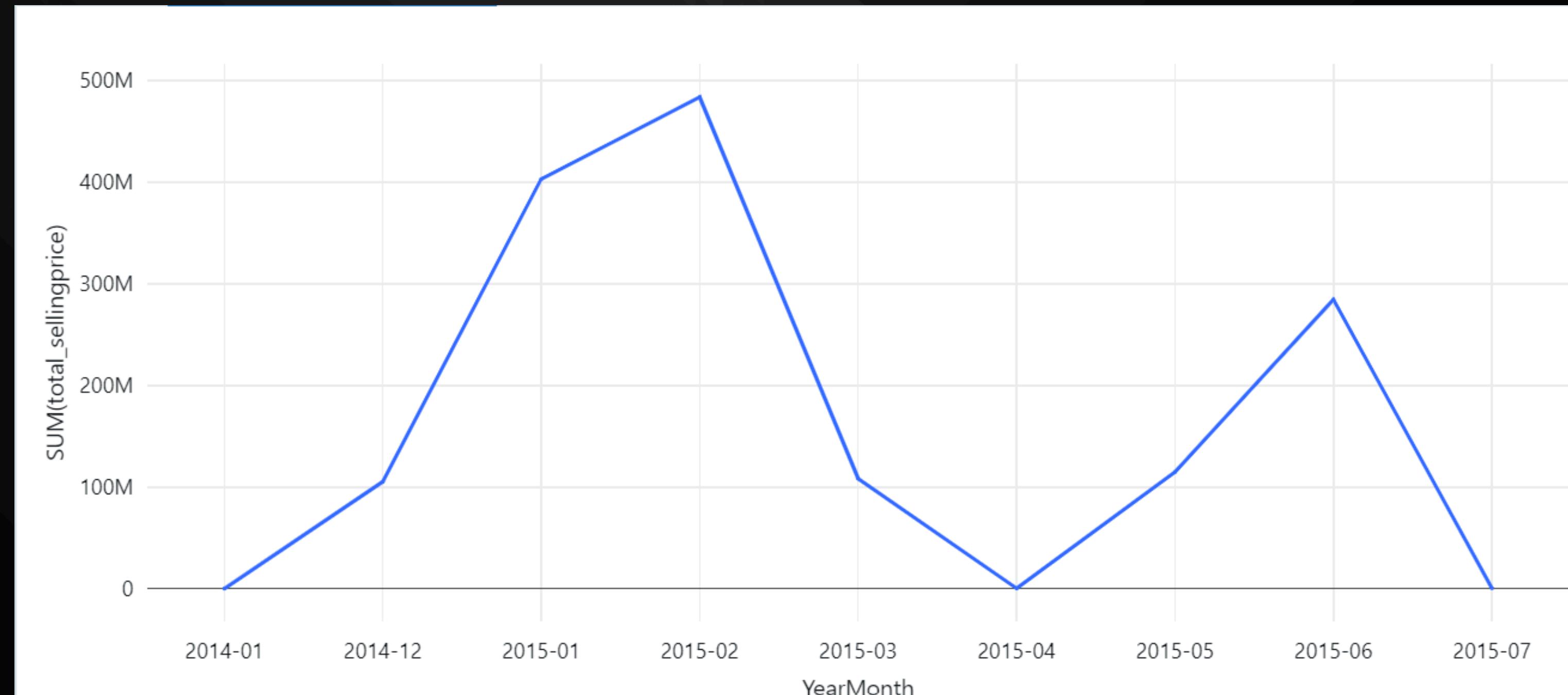
March 2015 Sales by Make

Data Visualisations



2015 Sales by Make

Data Visualisations



Sales by YearMonth

THANK YOU

For watching this presentation



+91 83297 85547



s.kangude@iitg.ac.in