GOLD LAYER 5: Diabetes Readmission Data Preparation

This notebook prepares and saves the gold-level diabetes readmission dataset for downstream analytics and modeling. Steps include loading silver data, selecting features, saving the gold table, and basic validation.

Gold Layer Data Engineering: Diabetes Readmission

Transforming Silver-level data into analytics-ready assets for downstream modeling.

Pipeline Architecture

01

Ingestion

Load `silver_diabetes_readmission` from Catalog.

02

Feature Selection

Filter for Demographics, Clinical History, and Medications.

03

Persistence

Write to Delta Lake with overwrite logic.

04

Validation

Verify schema integrity and record count (101,766 rows).

Outcome: A verified `gold_diabetes_ready` table optimized for predictive modeling.

Establishing the Foundation with Silver Data

The 'Silver' layer serves as the source of cleansed but potentially broad data. Our goal is to refine this into a specific use-case asset.

The initial ingestion brings in a wide array of fields that require strategic pruning.

python

```
# STEP 1: Load Silver Data
# Load the silver-level diabetes readmission data from Unity Catalog.
gold_df = spark.table("workspace.diabetes_readmissions.silver_diabetes_readmi
display(gold_df)
```

Result Output

```
> gold_df: pyspark.sql.dataframe.DataFrame = [race: string, gender: string ... 46 more fields]
```

Selecting Core Features: Demographics & Administration

To build a robust model, we filter the dataset to strictly relevant features, discarding noise.

Patient Profile

- `race`
- `gender`
- `age`
- `weight`

Hospital Stay Context

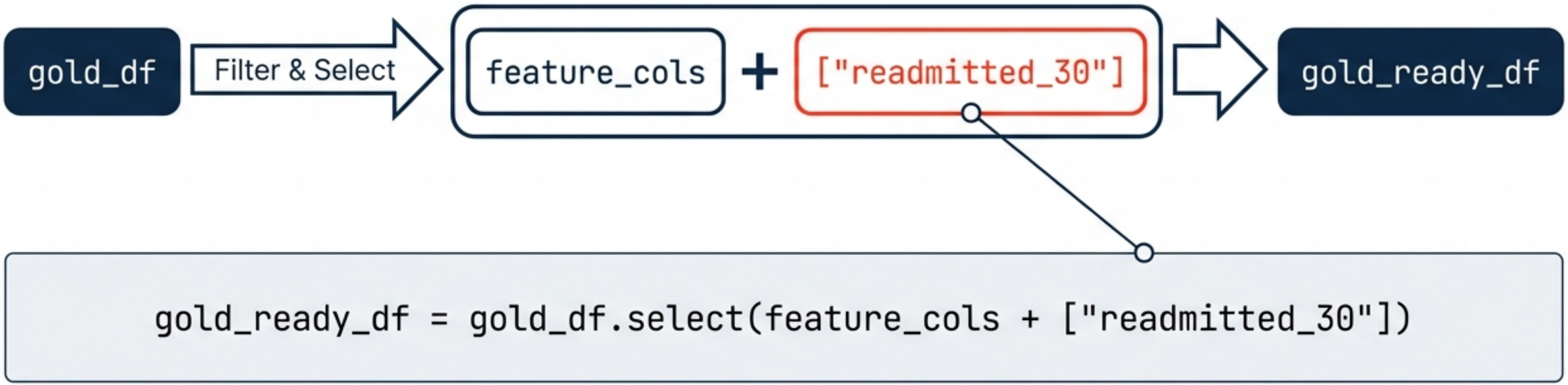
- `admission_type_id`
- `discharge_disposition_id`
- `admission_source_id`
- `time_in_hospital`
- `payer_code`
- `medical_specialty`

Capturing Clinical History and Treatment Patterns

Clinical Metrics	Key Bio-Markers	Pharmacological Array
num_lab_procedures num_procedures num_medications number_outpatient number_emergency number_inpatient	 max_glu_serum  A1Cresult	metformin repaglinide insulin glipizide glyburide pioglitazone rosiglitazone glyburide-metformin glipizide-metformin

Tracking medication changes is a critical indicator for diabetes management.

Finalizing the Gold DataFrame Logic




```
gold_ready_df = gold_df.select(feature_cols + ["readmitted_30"])
```

Target Variable (Label) for Machine Learning.

Persisting the Asset to Delta Lake

Delta Lake Integration

- The transformed data is written to stable storage to allow for ACID transactions and version control.
- We use ‘overwrite’ mode to ensure the Gold table always reflects the latest logic.

```
# STEP 3: Save Gold Table
gold_ready_df.write.format("delta") \
  .mode("overwrite") \
  .saveAsTable("diabetes_readmissions.gold_diabetes_ready")
```

Code for saving the Gold DataFrame as a Delta table.

Visual Validation: Inspecting the Output

race	gender	age	weight	admission_type_id	discharge_disposition_id	time_in_hospital	payer_code
Caucasian	Female	[0-10)	null	6	25	1	null
Caucasian	Female	[10-20)	null	1	1	3	null
AfricanAmerican	Female	[20-30)	null	1	1	2	null
Caucasian	Male	[30-40)	null	1	1	2	null
Caucasian	Male	[40-50)	null	1	1	1	null

Binned Age Intervals
Sparsity / Missing Data

Categorical Data

Quantitative Verification

```
%sql
SELECT COUNT(*) FROM diabetes_readmissions.gold_diabetes_ready;
```

101,766

Total Verified Records

Confirming data integrity and sufficient volume for machine learning training.

Ready for Machine Learning

Input	Process	Output
		
Raw Silver data with high dimensionality.	Strategic feature selection & medication logic.	Validated `gold_diabetes_ready` table. 101,766 Records

The dataset now adheres to the Gold standard: refined, aggregated, and ready for readmission prediction modeling.