Creating a Machine Learning Pipeline to Evaluate Employee Attrition

Presented by: Apoorv Anand, Ashmita Mukherjee, Sivaram Mandava

Prepared for: The University of Chicago MS in Applied Data Science, Machine Learning Operations Final Project

Date: August 15, 2024

Agenda

01 EDA + Preprocessing

Dataset overview with exploratory insights + cleaning

02 Pipeline + Modeling

Our Pipeline and Model choice after an initial AutoML run

03 Deployment

How we deployed for inference using a Databricks workflow along with endpointing

04 Monitoring + Dataset Change

How we deployed model monitoring and tracked the synthetic test data drift

We chose an employee attrition dataset where we predict churn based on various features

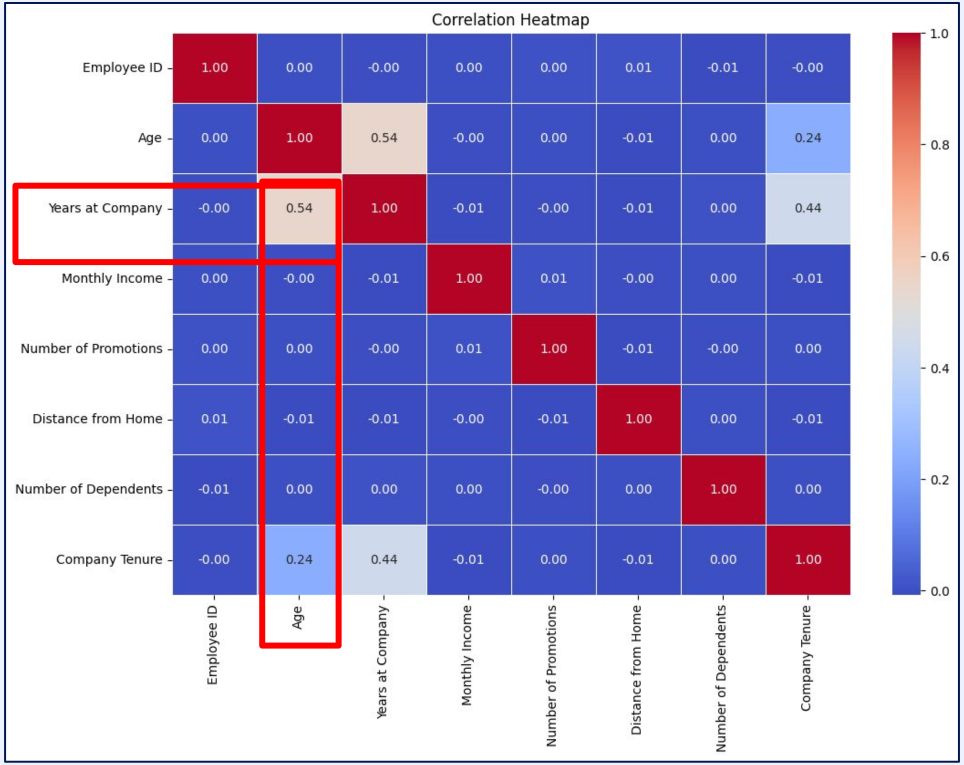
Employee Attrition Classification Dataset (from Kaggle)

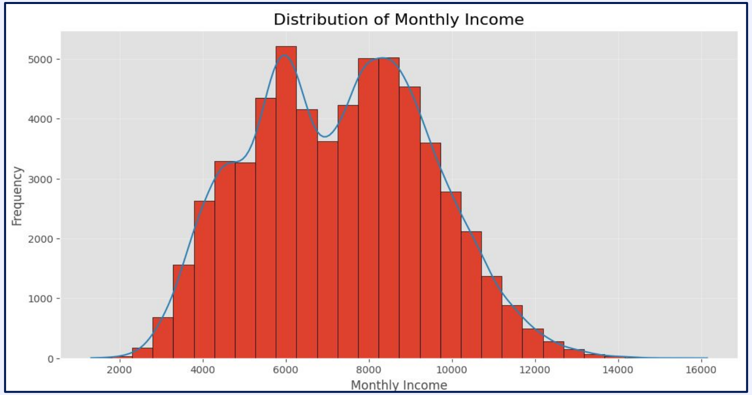
- Simulated dataset designed for the analysis and prediction of employee attrition
- ~75K samples split into train and test sets
- Primary key of employee id
- Useful features such as:
  - **Numeric**: age, years at company, distance from home
  - **Categorical:** gender, job role, job satisfaction, marriage status

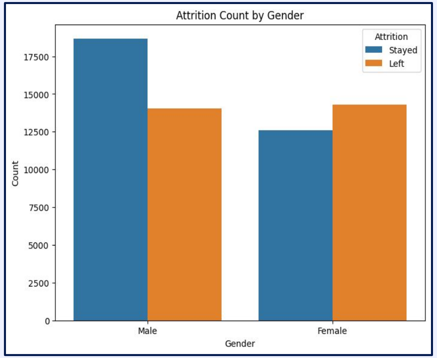
	Train	Test
# Observations	**59,598**	**14,900**
Split-Percentage	80%	20%
Target Variable	**Attrition (Stayed or Left)**	

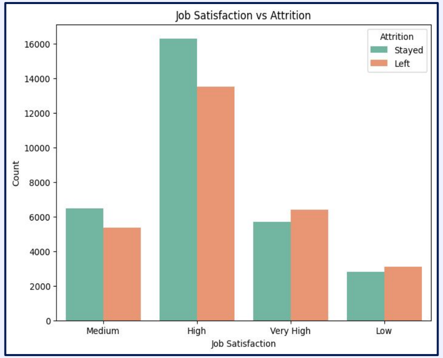



Our exploratory analysis of the data indicates features that are relatively uncorrelated and ripe for pre-processing









We preprocessed our features using techniques like encoding and binning to improve model accuracy

Label Encoding

Education		Education
High School	→	1
Bachelors		2

One Hot Encoding

Gender		Gender_ Male	Gender_ Female
Male	→	1	0
Female		0	1

Dropped Redundant Features

Years at Company | Company Tenure

Years at Company | ~~Company Tenure~~

Binned Continuous Features

Age		Age_ 17-26	Age_ 43-51
16	→	1	0
45		0	1

After preprocessing, we pushed our train features to a feature store within Databricks itself

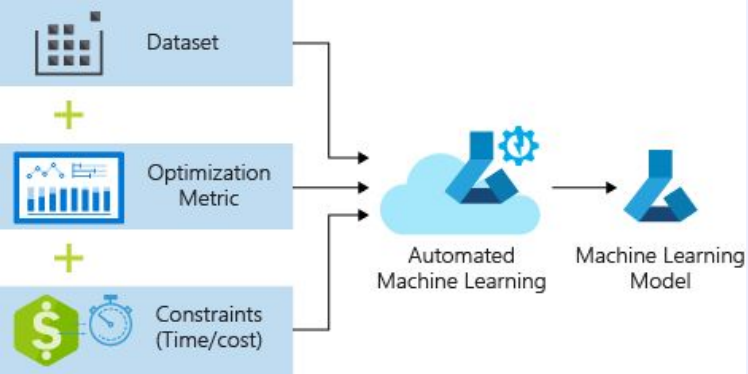
Data downloaded from Kaggle as .csv files (train, test)

Databricks ingestion into separate tables

Conduct preprocessing steps (binning, encoding, etc)

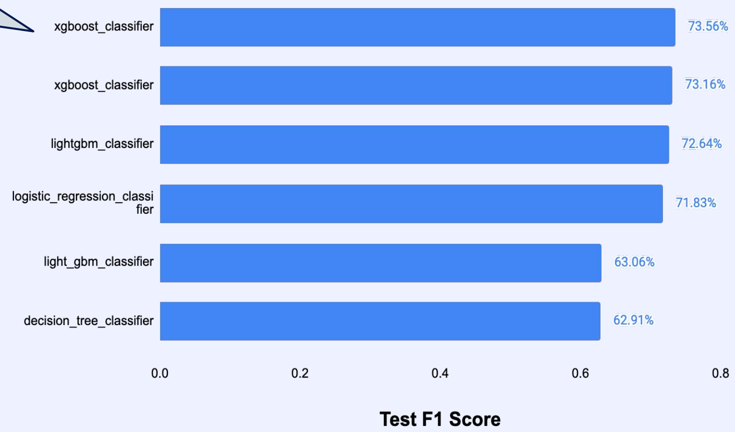
Push cleaned features to Feature Store in Databricks

We deploy AutoML to explore model options and subsequently settled on XGBoost Classifier



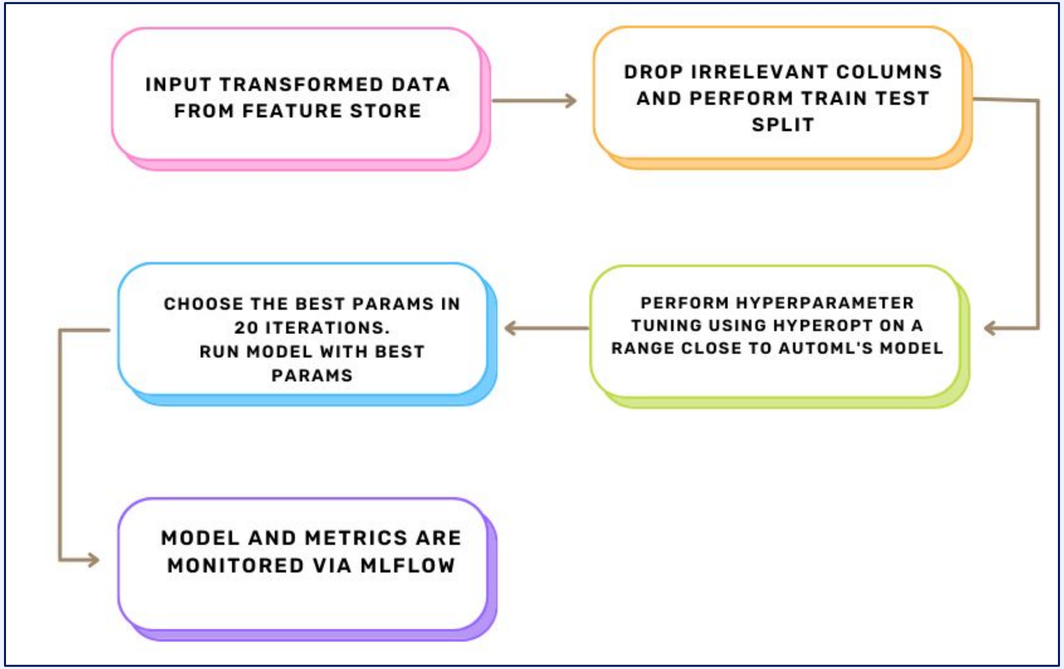
https://softwareengineeringdaily.com/2019/05/15/introduction-to-automated-machine-learning-automl/

AutoML found XGBoost with an F1 score of ~74% as the best model



Run Name	Created	Dataset	Duration	Source	Metrics: test_accuracy_	test_f1_score
smiling-roo-145	1 day ago	dataset (1852359e) Train , da... +2	3.1min	-	0.7583041...	0.7469858...
intrigued-steed-439	1 day ago	dataset (1852359e) Train , da... +2	3.4min	-	0.7576364...	0.7463755...
hilarious-shrew-519	1 day ago	dataset (1852359e) Train , da... +2	5.9min	-	0.7566349...	0.7445690...
vaunted-shrimp-341	1 day ago	dataset (1852359e) Train , da... +2	3.3min	-	0.7560507...	0.7441575...
gentle-fish-177	1 day ago	dataset (1852359e) Train , da... +2	3.6min	-	0.7552996...	0.7440195...
nosy-snail-738	1 day ago	dataset (1852359e) Train , da... +2	1.7min	-	0.7561342...	0.7438639...

We then create our XGBoost model using Hyperopt and MLFlow pipeline to generate our predictions and metric evaluations

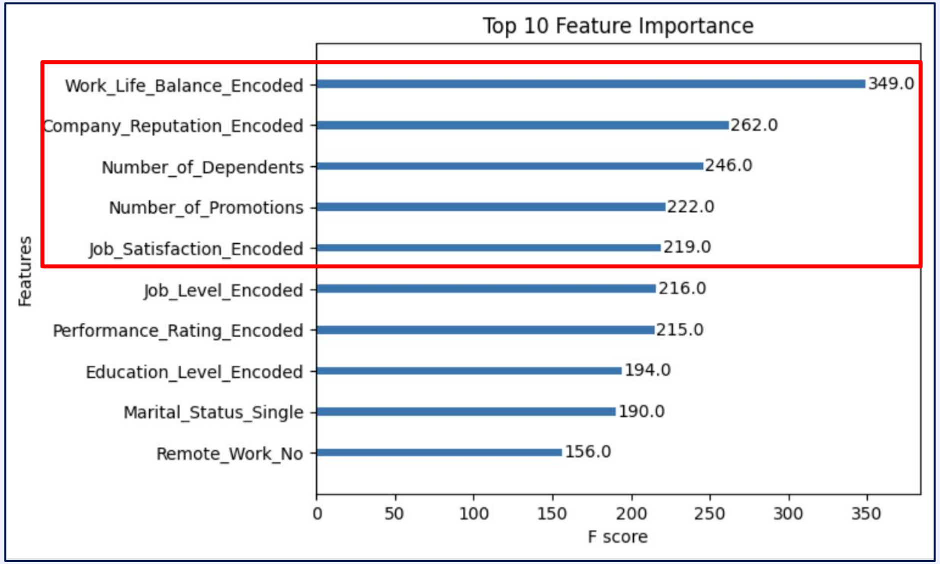


Modelling Process

```
Best Parameters chosen by Hyperopt for XGBoost:
objective: binary:logistic
colsample_bytree: 0.402
enable_categorical: False
learning_rate: 0.072
max_depth: 3
min_child_weight: 1
missing: nan
n_estimators: 584
```

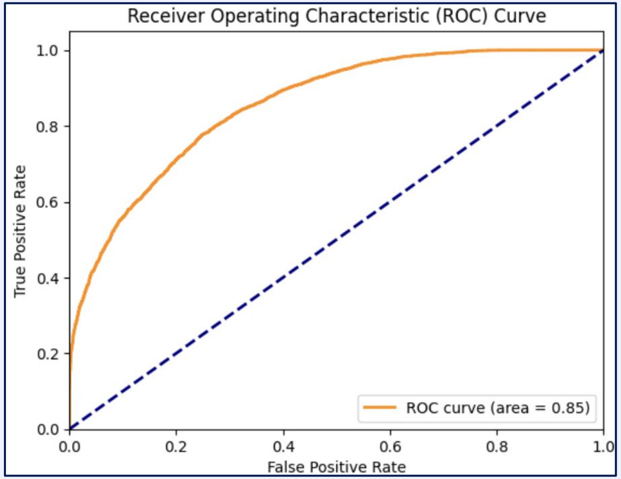
Best Parameters chosen by Hyperopt with values ranges based on baseline AutoML

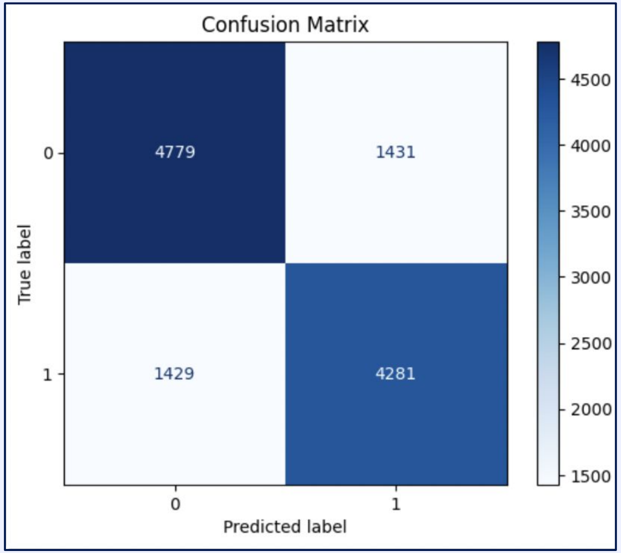
Metrics used for Evaluation and Results



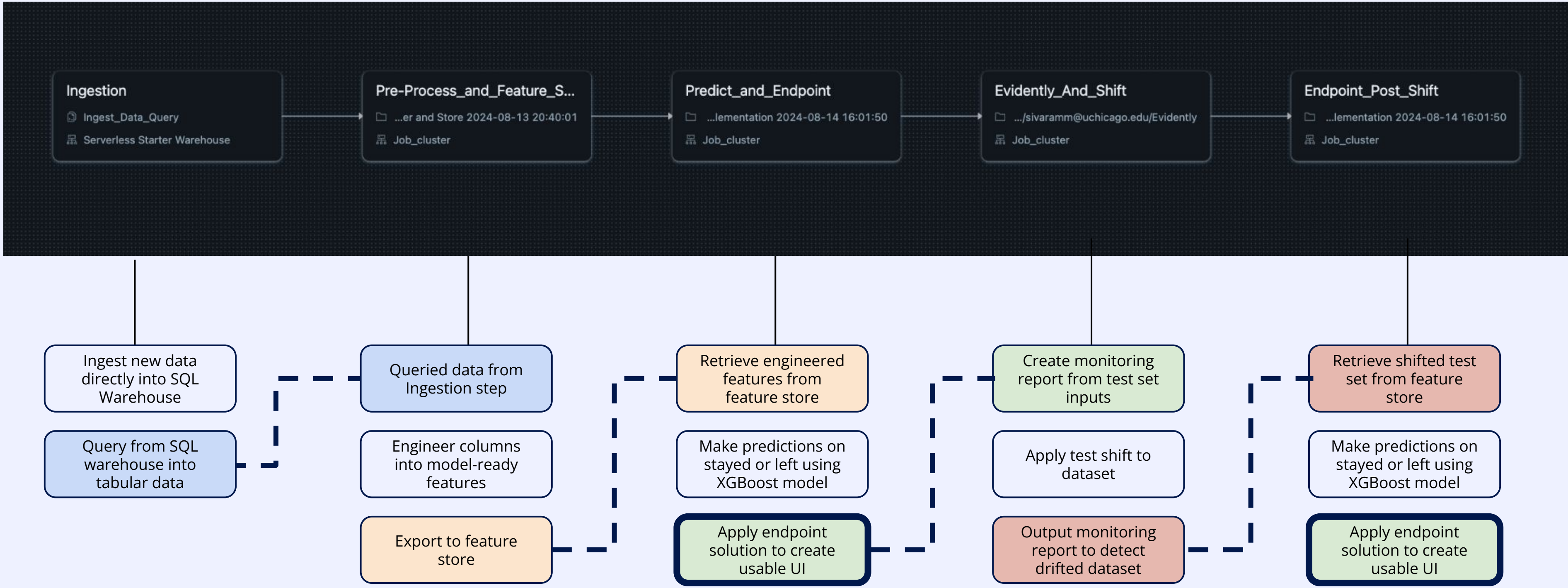
	precision	recall	f1-score	support
0	0.77	0.77	0.77	6210
1	0.75	0.75	0.75	5710
accuracy			0.76	11920
macro avg	0.76	0.76	0.76	11920
weighted avg	0.76	0.76	0.76	11920

F1 score for both classes have similar performance indicating no bias





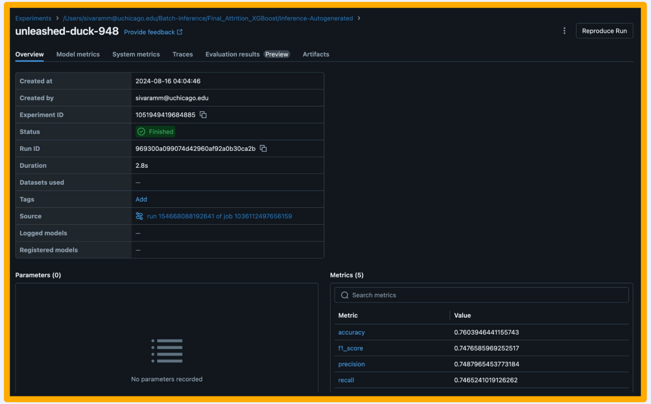
A Databricks workflow allows us to deploy our model in an environment that continuously ingests new data

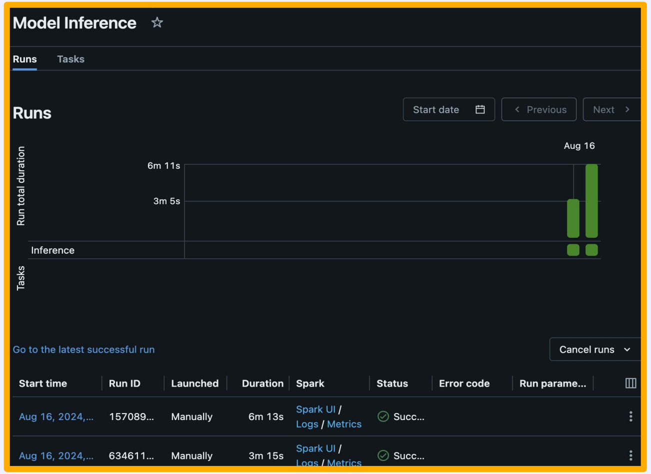


The deployed model is used for inference to predict attrition

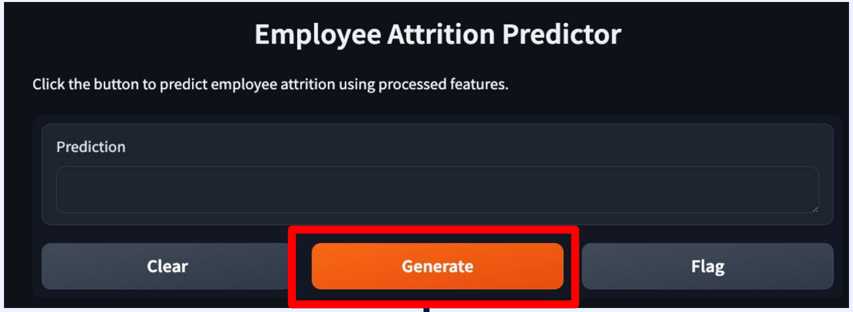
Deployment

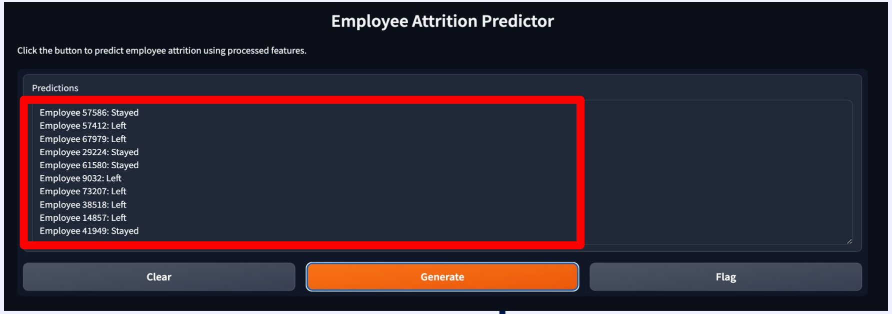
	Employee_ID	prediction
1	38272	1
2	1549	0
3	31466	0
4	30663	0
5	37115	0
6	52460	1
7	7612	1
8	63461	1
9	1	1
10	72300	0
11	37247	1
12	72219	0
13	19533	1
14	50626	0
15	25720	0





Gradio offers a simple endpoint solution for our pipeline, outputting our inference results in an intuitive user interface





*Example of **<u>10 attrition inferences</u>** our model makes from newly ingested data in the Gradio endpoint UI*

Model Monitoring with MLFlow for two separate runs

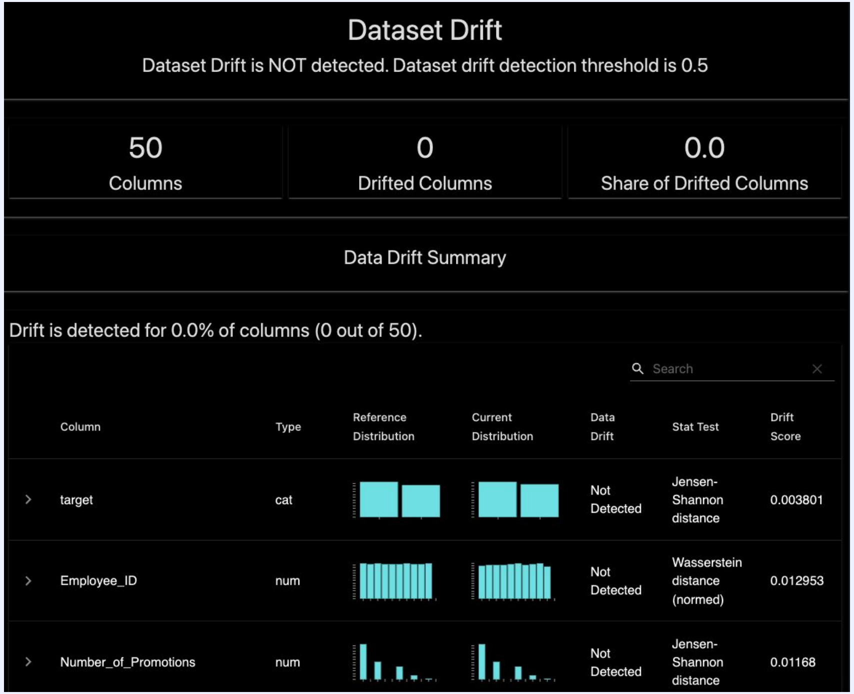
Search metrics

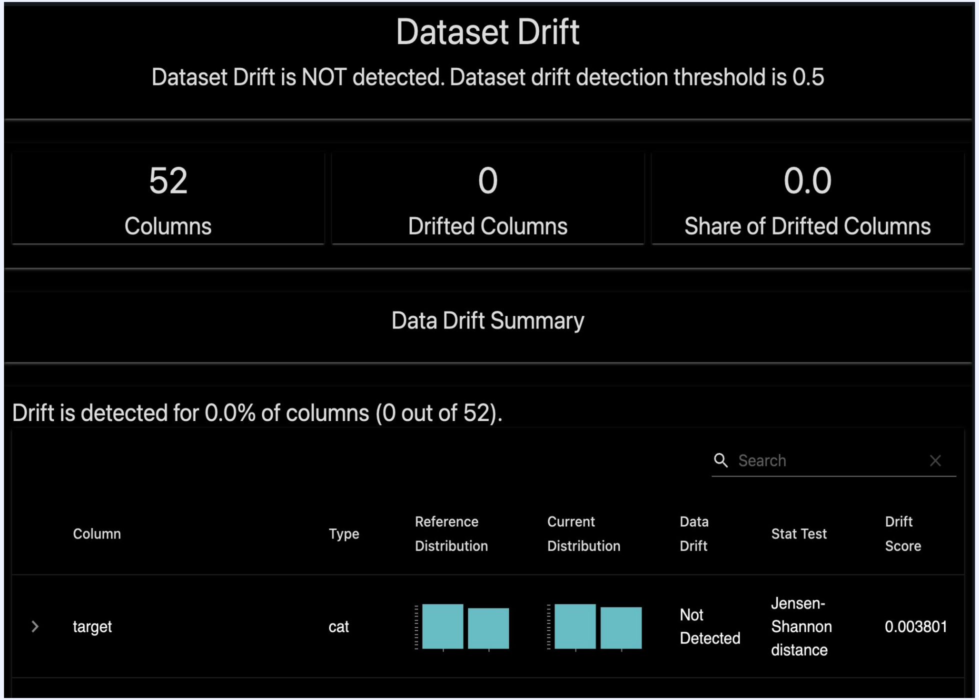
Metric	Value
accuracy	0.7424496644295302
auc_roc	0.8295651144313345
f1_score	0.7285587975243147
precision	0.7270160578789483
recall	0.7301080985291512

Search metrics

Metric	Value
accuracy	0.7408557046979866
auc_roc	0.8294899612190207
f1_score	0.7287257398788091
precision	0.7320042342978123
recall	0.7254764819024305

We spin up Evidently AI to generate monitoring reports to track our model and its performance





To explore "changed" test data, we shift Number of Promotions and Number of Dependents in our dataset

Number of Promotions → (+0-7 shift) → ***Shifted*** Number of Promotions

Number of Dependents → (+0-7 shift) → ***Shifted*** Number of Dependents

Drift in column 'Number_of_Promotions'

Data drift detected. Drift detection method: Wasserstein distance (normed). Drift score: 1.505

Drift in column 'Number_of_Dependents'

Data drift detected. Drift detection method: Wasserstein distance (normed). Drift score: 0.963

2	2	1.0
Columns	Drifted Columns	Share of Drifted Columns

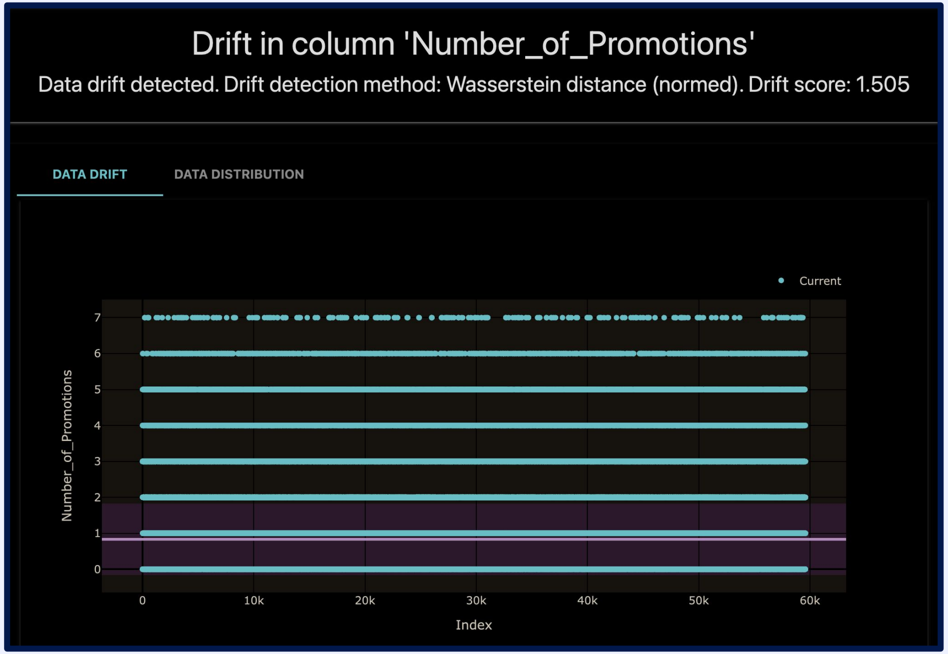
Data Drift Summary

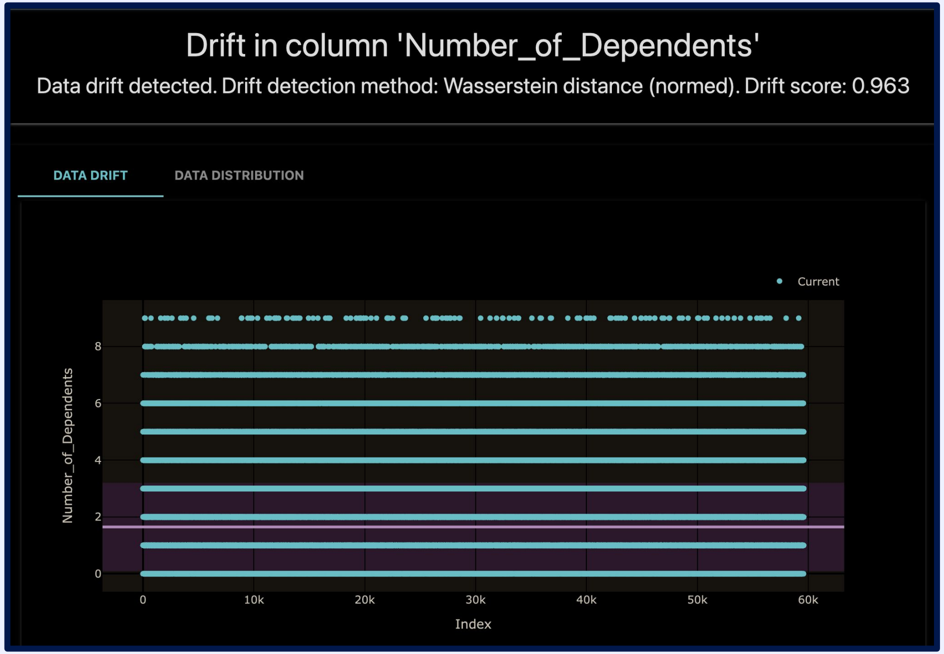
Drift is detected for 100.0% of columns (2 out of 2).

Search

Column	Type	Reference Distribution	Current Distribution	Data Drift	Stat Test	Drift Score
Number_of_Promotions	num			Detected	Wasserstein distance (normed)	1.504748
Number_of_Dependents	num			Detected	Wasserstein distance (normed)	0.963057

Evidently AI automatically displays drift insights for our changed features

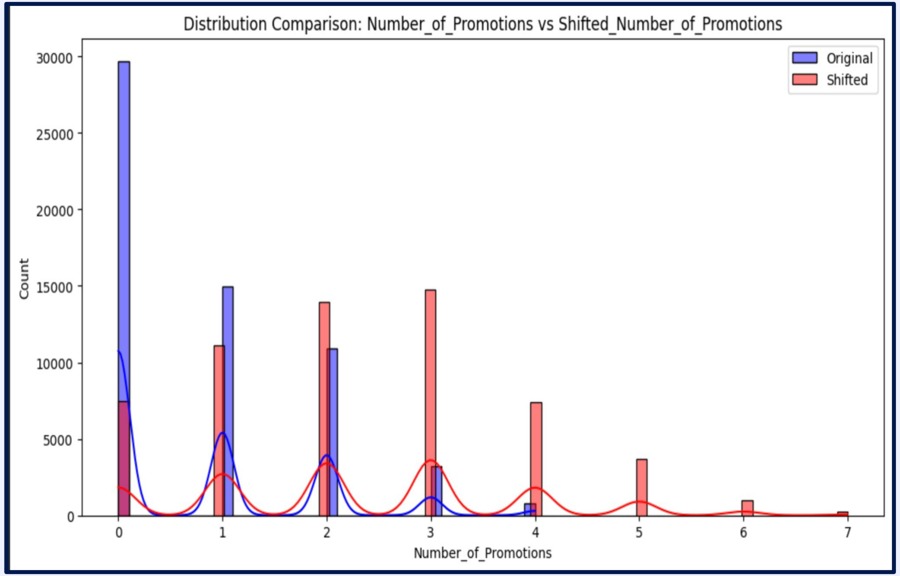


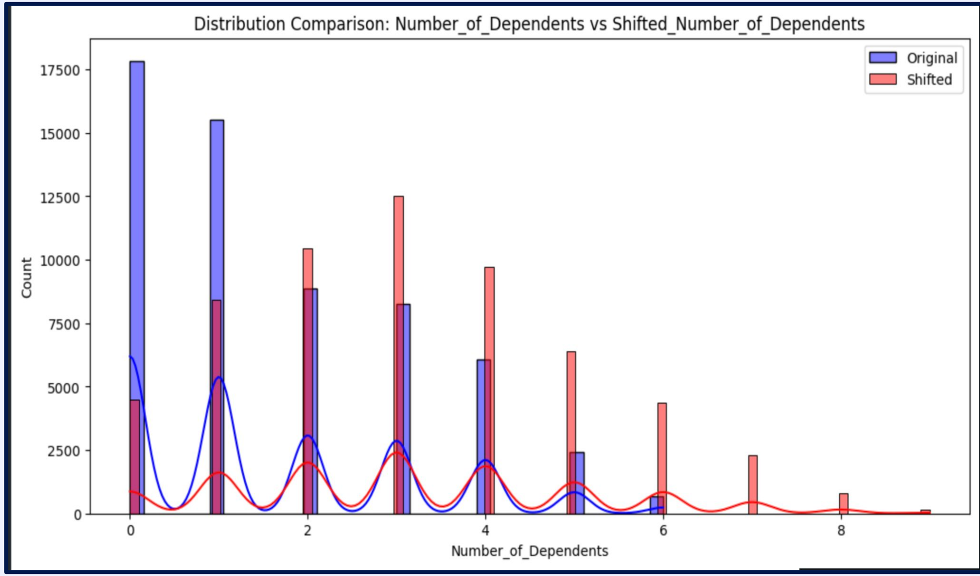


Drift Scores Higher than the Threshold of 0.5, hence Drift is detected

Created Custom Charts to Monitor the Drift

Monitoring + Test Data Drift





Link to Video Demonstration

https://drive.google.com/file/d/1RghT9TW1hIk8lsRimOdat9OhvWh3DfIV/view?usp=share_link

Link to Notebooks

https://uchicago-team2-databricks.cloud.databricks.com/browse/folders/1051949419685084?o=29437060
2992426

Questions?

For any follow-ups, please email:

Apoorv Anand - apanand@uchicago.edu

Sivaram Mandava - sivaramm@uchicago.edu

Ashmita Mukherjee - ashmitam@uchicago.edu