ReDI School of Digital Integration

Airport Capacity Prediction

Using Machine Learning

Pavithra Sathya Kumar
Contact: pavithrask836@gmil.com

We use tech to connect human potential and opportunity with dignity & humility

Objective

Objective

Goal:

Predict **daily airport capacity** (total flights per day) using machine learning.

Why this is required?

- Airports experience strong variations (seasonal peaks, weekends, holidays)
- Helps with **staffing, resource allocation, runway usage & operational planning**
- Supports decision-making after disruptions (ex: COVID recovery)

Target variable:

- **FLT_TOT_1** = total departures + arrivals for each airport per day

Methodology

Data Overview

- Data was obtained from Kaggal
- Data was collected for European flight data between 01-01-2016 to 31-05.2022 ~ 6.5 years data coverage
- Overall, it has 688099 rows and 14 columns
- Based on the Dataset observation:
  - 332 ICAO code
  - 333 unique airport names
  - 42 countries
- IFR data is missing for most airports
- This is moveover a flight traffic dataset rather than flight delay

Methodology

EDA Analysis

Seasonality:
Traffic peaks in summer (Jun–Aug)
Lowest in winter (Jan–Feb)

Country trends:
UK, Germany, Netherlands are busiest

Airport trends:
Frankfurt, Amsterdam, Heathrow highest traffic

IFR/VFR:
Major airports = IFR-heavy (stable)
Small airports = VFR-heavy (irregular)

Features Used for Modeling:

Time-based features:

YEAR, MONTH, DAYOFWEEK, IS_WEEKEND
Airport characteristics:

APT_ICAO (encoded)
IFR_ratio (long-term instrument dependency)
Lag features (for forecasting):

lag_1 (yesterday)
lag_7 (last week)
lag_30 (last month)

Target: FLT_TOT_1 = total flights per day

Methodology

Modelling Approach

- **Linear Regression**
- **Random Forest**
- **XGBoost**

1. **Linear Regression**
   Simple baseline
   Cannot model non-linear aviation patterns

2. **Random Forest**
   Learns non-linear patterns
   Still unstable with highly seasonal data

3. **XGBoost**
   Best at capturing seasonality + airport interactions
   Most accurate

Results

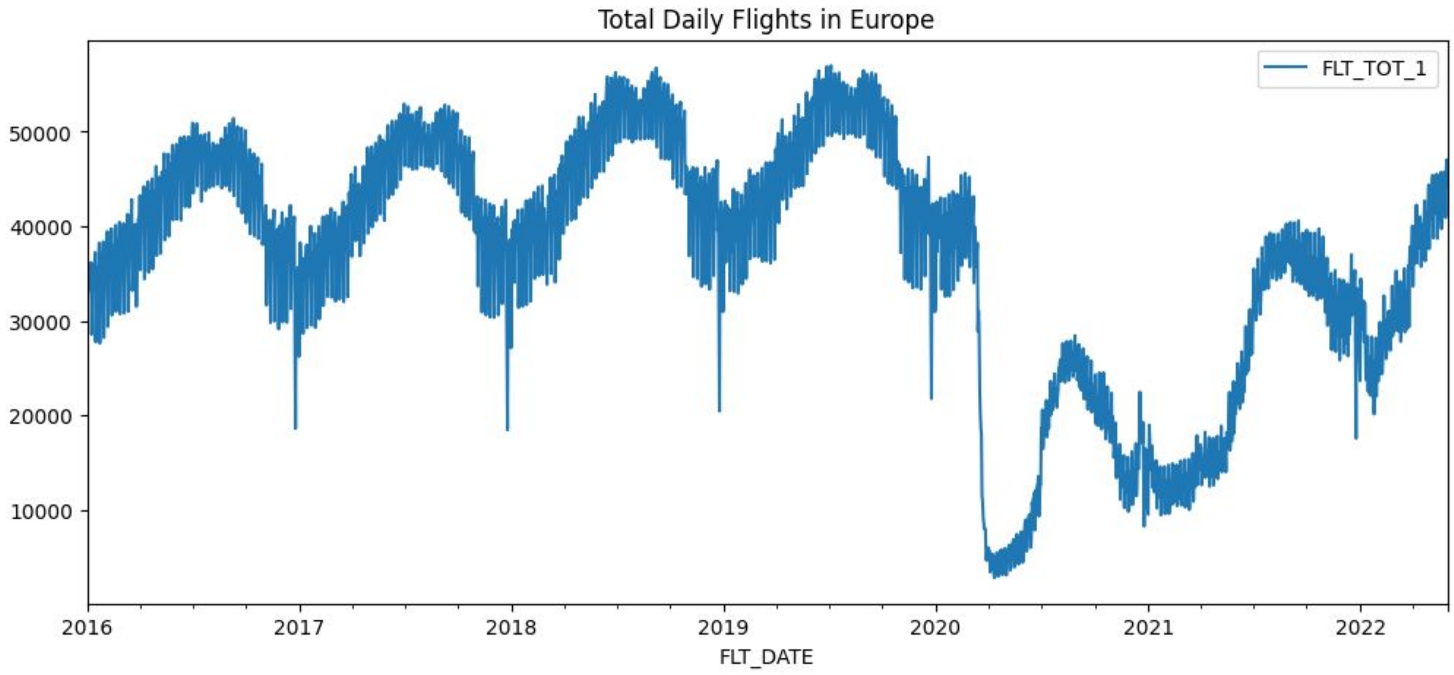
Model Comparison without fixing the lag / data leakage vs After Fixing Leakage

Model	MAE ↓	RMSE ↓	R^2 ↑
Linear Regression	78.49	152.53	0.6
Random Forest	45.66	161.1	0.55
XGBoost	**44.47**	**137.27**	**0.68**

Model	MAE	RMSE	R^2
Linear Regression	185.3	236.89	0.04
XGBoost	213.92	279.78	–0.34
Random Forest	223.65	295.7	–0.49

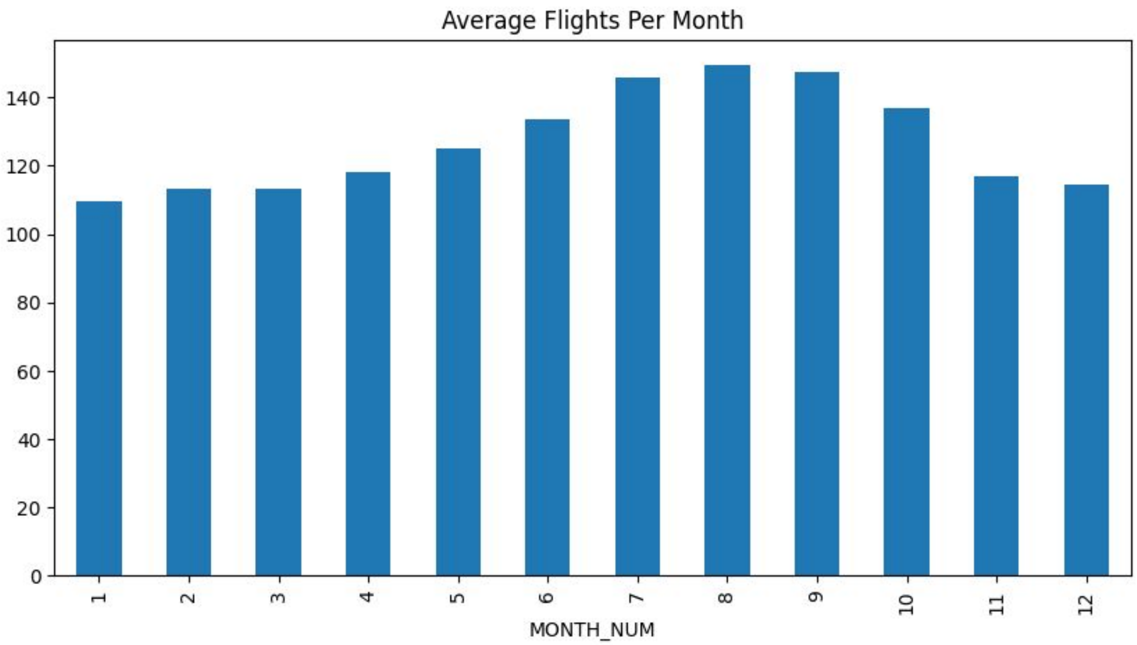
Results

Overall flights trends between 2016 to 2022



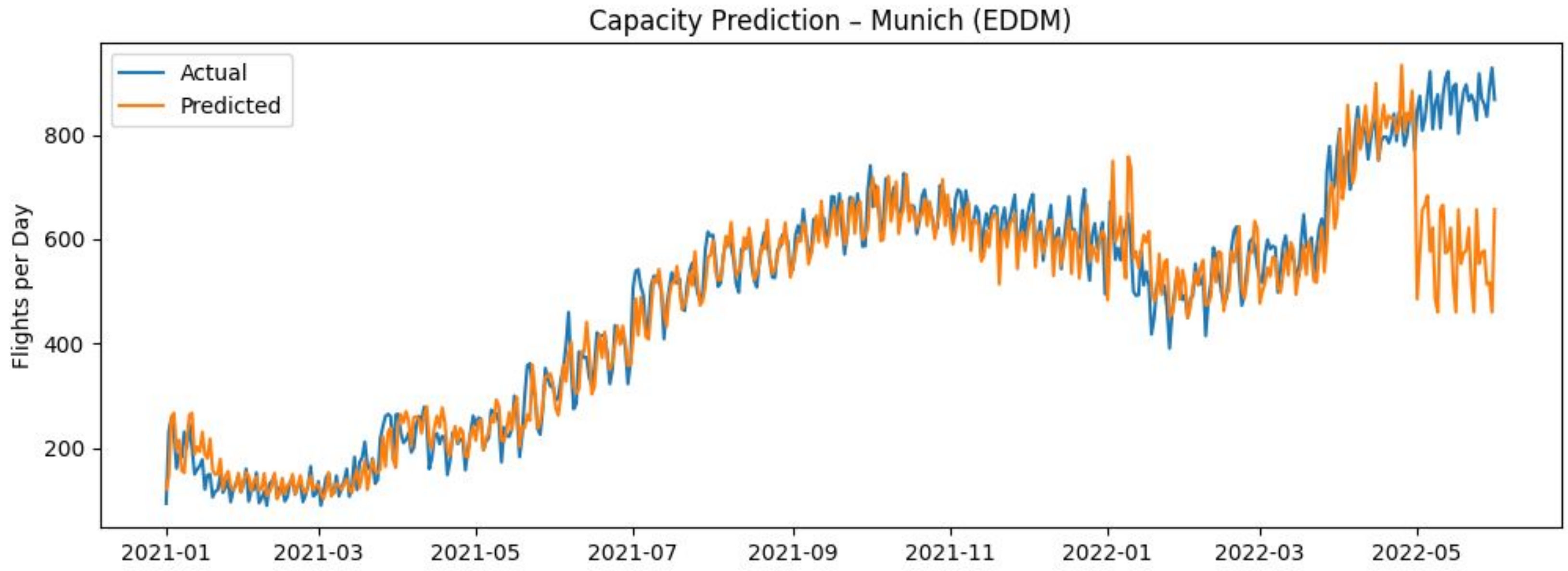
Result

Average Flight trend for the complete dataset of EU flights



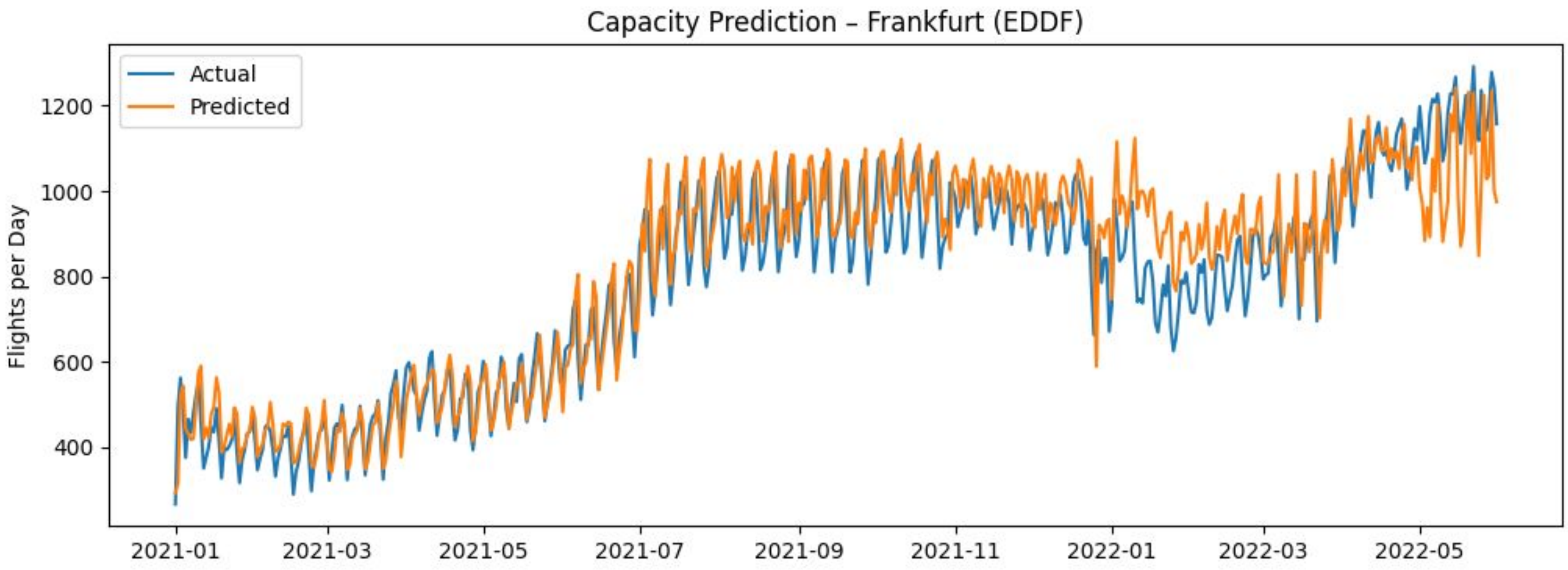
Result

Capacity predicting for Munich



Result

Capacity predicting for Frankfurt



Discussion

Challenges

- Same-day IFR caused unrealistic accuracy
- EU airports are too diverse without lag features
- Train/test split didn't align after filtering

Improvements/ future Work

- Add weather (visibility, wind, precipitation)
- Include holiday/event data
- full time-series forecasting
- Add airport clustering for improved regional insights

Conclusion

- Cleaned and explored EU flight dataset (2016–2022)
- Compared regression models fairly (leakage-free)
- XGBoost selected as best learning model
- Built a realistic forecasting model using lag features
- Accurate predictions for major German airports
- Demonstrated clear ML workflow:
  EDA → Cleaning → Modeling → Comparison → Forecasting

Thanks a lot!