

EMS740U/P–Machine Learning and Artificial Intelligence for Engineering

Case Study: Artificial Intelligence in Air Traffic Management (ATM)

Data-Centric Engineering in Airport Airside Operations

16/09/2025

Student Pack – 2025/2026

Contents

1.	Problem description.....	2
2.	Description of tasks.....	2
2.1	Data collection/selection and processing	2
2.2	Dimensionality reduction using PCA	3
2.3	NN & LR & ANFIS.....	3
2.4	Comparison.....	4
2.5	Conclusion.....	4
3.	Assessment.....	4
4.	Time table	6
Appendix A: Codes for airport data processing		6
Appendix B: Descriptions of airport data features		6
Appendix C: Steps to success (For information only).....		7

1. Problem description

In order to optimise airport airside operations, accurate taxi time prediction has played an indispensable role. It is not only important to create more robust schedules and identify choke points between gate and runway for practitioners, but also helps the government analysts to estimate the optimal airport capacity and evaluate the regulation impacts.

This case study utilises taxiing data from Manchester International Airport (MAN) ranking among the 2nd busiest airports in the UK.

In order to ensure taxi time prediction accuracy, one should comprehensively consider relevant features that may affect taxi time. In this case study, the data comes with up to 25 features, aiming to provide a sufficient set of features for the taxi time prediction. These relevant features are divided into three categories, including (a) aircraft and airport operational factors, (b) airport congestion level and (c) aircraft average speed.

You will need to complete 4 tasks in this case study:

- (i) Collecting>Selecting and pre-processing data using the programs downloaded from QM+.
- (ii) Applying *feature engineering* technologies, in particular Principal Component Analysis (PCA), for feature extraction using the dataset that has been collected/selected by you.
- (iii) Applying *supervised learning*, including the Neural Network (NN), Linear Regression (LR) and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) to predict the taxi time.
- (iv) Discussing the *pros and cons* of the different machine learning tools from the aspects of prediction accuracy, generalisation capability and model transparency.

You may also find **Appendix C: Steps to Success** helpful to complete the above tasks.

2. Description of tasks

2.1 Data collection/selection and processing

The students will work together in groups. Each group should collect data from <https://opensky-network.org/> for MAN or select data over a period of time from the provided dataset. In order to achieve appropriate data sets for machine learning studies, the following rules need to be followed:

- Each data set should be **a minimum of** 10 times the number of useful features data points (you need to decide the time to start recording/retrieving your data and for how long), but **do not exceed** 2500 data points (otherwise it will consume lots of your time).
- Data sets collected/selected by groups should be different (consider different time periods).

The collected/selected data sets should be pre-processed, so that they are suitable for machine learning based modelling. Data processing is conducted using the programs downloaded from QM+. Details of the data processing programs are shown in **Appendix A**.

The final pre-processed dataset file named features.csv. This file will be used as input for the next task (PCA analysis).

2.2 Dimensionality reduction using PCA

The available features in the collected/selected data after pre-processing are explained in **Appendix B**.

We will not work with all the features extracted in the previous task. Instead we will attempt to select the best features for the pre-processed dataset. There are different ways to do this. For the purpose of this task, the feature selection will be conducted using PCA, which is a linear dimensionality reduction technique that extracts useful information from a high-dimensional space by projecting it into a lower-dimensional sub-space.

Split the pre-processed dataset using a typical partition of 70% for training, 20% for validation and 10% for testing. It is important that the PCA analysis for feature selection is only performed on the training set (the validation/test sets should remain unseen until the final test).

You will need to do some research on how to use PCA as a tool to **select** features (e.g., a paper included in W3.3 slide provides a potential way). You need to decide how many features you will use and explain the reason of the choice.

At the end of this task, save the reduced (fewer-feature) datasets as `train_reduced_gX.csv`, `validation_reduced_gX.csv` and `test_reduced_gX.csv`, ready for the regression tasks..

2.3 NN & LR & ANFIS

NN and LR are classical machine learning models and the foundation of many other advanced machine learning approaches. ANFIS represent a hybrid intelligent system. In this case study, you need to apply these three models to solve the taxi time prediction problem.

For NN, refer to the following rules:

- Applying *Back Propagation (BP)* NN for prediction.
- Choose the number of the hidden layer nodes and explain the reason.

For LR, refer to the following rules:

- Applying the *polynomial basis function* for LR modelling.
- Decide the maximum order of the polynomial function and explain the reason.

For ANFIS, refer to the following rules:

- Applying the *Clustering* for constructing the initial Fuzzy Rule-based System.
- Decide the learning algorithms to further train Fuzzy Rule-based System and explain the reason.

General instructions for all models:

- Ensure you use the same training/validation/test split for all models to allow fair comparison. Start from the .csv files created in the PCA task.
- Before training, standardise the features to zero mean and unit variance to improve numerical stability and reduce rounding errors.
- Evaluate model performance using metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2 .
- Visualise results using scatter plots of predicted vs actual values and/or residual plots.
- Statistical tests of the obtained regression models are needed.

2.4 Comparison

Compare NN, LR and ANFIS from the aspects of prediction accuracy, generalisation capability and model transparency.

The prediction accuracy is quantified by the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Relative Error (MRE), etc.. Statistical tests and/or interval estimation are needed as the means to report the model performance/skill.

2.5 Conclusion

Read the following three reports that are provided on QM+. You will gain an understanding of challenges associated with the safe use of AI, the importance of data to AI, methods for determining different sub-datasets for training and verification purposes, and appreciate the urgent need to develop new technologies, processes, tools, and guidance for assuring the safety of systems based on these technologies. Draw conclusions of your work and results in this coursework, using these understandings of AI in life critical engineering.

- Read the report “The FLY AI Report” produced by EUROCONTROL. This report provides an overview of the many ways that AI is already applied in the aviation sector and ATM and assesses its potential to transform the sector.
- Read the report “AFE 87 – Machine Learning” produced by Aerospace Vehicle Systems Institute. This report provides an overview of different Machine Learning paradigms and how these emerging technologies present new challenges to existing certification processes in the aviation sector.
- Read the article “Aircraft taxi time prediction: Comparisons and insights” to get a better idea of how different Machine Learning algorithms have been used and the pros and cons of different methods.

3. Assessment

The **first** deliverable of this case study is a 5-minute pre-recorded group presentation (**15%** of the whole module). The **second** deliverable of this case study is a 10-page group report (**25%** of the whole module). The main parts and the suggested weightage of each part to be included in the presentation/report are indicated in the table below. Both deliverables will be marked against these main parts.

Main parts	Guidance	Weightage (100%)
Introduction	<ul style="list-style-type: none">• Details of individual contributions towards the case study and report.• Description of the problem and how the predictive model can be used to improve ATM at present and in future operations concepts.	5%
Data Preparation	<ul style="list-style-type: none">• Description of your data set and justification of your choice of training, validation and testing sub-datasets.• Description of the steps taken during the PCA analysis and interpreting results, including<ol style="list-style-type: none">1. Pre-processing data for the PCA,2. Calculating the principal components,3. Choosing the appropriate number of principal components,4. Interpreting the principal components and feature importance,5. Plotting selected principal components,6. Assessing the information loss.	30%

NN & LR&ANFIS	<ul style="list-style-type: none"> Description of steps taken to construct the NN, LR and ANFIS models and interpreting results, including <ol style="list-style-type: none"> Choice of model structures, Which sub-dataset(s) is(are) used for training, testing and validation, Quantify and visualise the performance of the predictive model on all sub-datasets, Conducting the statistical tests of the regression models, Assessing transparency and generalisation capability of the model. 	35%
Comparison	<ul style="list-style-type: none"> Discussing the pros and cons of the NN, LR and ANFIS approaches in taxi time prediction from the aspects of prediction accuracy, generalisation capability and model transparency. Quantifying and visualising the comparison results in the report. 	20%
Conclusions	<ul style="list-style-type: none"> Drawing conclusions of the study, as well as further crucial steps for machine learning approaches to be applied in real operational environment at airports. 	10%

The pre-recorded presentation should be no more than 5 minutes consisting of 5~10 slides. The presentation should be clear and concise, including all the main parts mentioned above. Submission via QM+ for this deliverable include:

- Pre-recorded presentation (5 minutes).

The group report should have the generic/standard structure and describe the problem, the way in which it was tackled, the analysis carried out and the results obtained. The report should be clear and legible and conform to the following requirements:

- Font size 11 is the minimum font that is acceptable and page margins should be at least 2 cm in all directions.
- Maximum 10 pages of A4 inclusive of the title, authors, details of individual contributions, main body of the report and Lists of references.
- You need to include a list of references based on a standard method (such as Harvard or IEEE style: you can find information about Harvard/IEEE style in the internet)). All the references need to be cited within the text.
- Report exceeding the page limit, or not adhering to the specified format, will be penalised.
- Submission should be made in a single PDF file through the QM+ submission point.
- Machine Learning codes and dataset need to be ready to run for checking and assessment, packed and uploaded to the QM+.
- You need to submit the following via QM+ ("X" in the file names refers to your group's letter, e.g. A, B, C....). **The total size of all submitted files, including the video, must not exceed 50 MB.**
 - Pre-processed collected/selected dataset: the pre-processed full dataset named as features.csv that will be used as the input to Feature Reduction.
 - Feature Reduction: PCA_gX.py (Example: PCA_gA.py for Group A submission). This code will read the file 'features.csv' (see **Appendix A**) and perform a PCA analysis of the feature values in the training set, to select the most important features for modelling and predictions. It will then write three CSV files containing selected features and target values for training, validation and testing.
 - train_reduced_gX.csv: Dataset containing the training set and only the features selected by the PCA algorithm.

- validation_reduced_gX.csv: Dataset containing the validation set and only the features selected by the PCA algorithm.
- test_reduced_gX.csv: Dataset containing the training set and only the features selected by the PCA algorithm.
- Machine Learning: XX_gX.py (Example: NN_gA.py for Neural Network of Group A submission) for NN and LR, and ANFIS_gX.m for ANFIS. These codes will read the datasets of the reduced features and apply the respective machine learning methods (NN, LR, ANFIS) to train, test and validate the elicited models for taxi time prediction.
- report_gX.pdf: group report in a single PDF file.
- presentation_gX: 5-minute pre-recorded group presentation (any video format).

4. Time table

Date	Missions	Time	Place
Week 4 (15 & 16 Oct), Wednesday & Thursday	Course works release	17:00-18:00	PBL Session
Week 13 (17Dec), Wednesday	- Submission of Pre-recorded Presentation, Report, dataset, and codes	Before 24:00	QM+

Appendix A: Codes for airport data processing

Codes that are provided by us on QM+;

- Map of the airport: man_map.txt
- Data Collection: Opensky.exe – This executable allows you to collect raw data from <https://opensky-network.org/>. After the successful execution, it outputs JSON files containing flight information.
- Data Cleaning: clean.exe – This executable takes JSON files as an input and combines these files in one readable CSV file. The output file name is ‘cleaned.csv’.
- Preprocessing (map matching): snap.exe – This executable takes ‘cleaned.csv’ and airport map as inputs. It matches (snaps) the aircraft movements into the airport map. It outputs snapped movement information in a file named ‘snapped.csv’. This operation allows us to calculate aircrafts’ position details while they are taxiing.
- Feature Extraction: feature_extract.exe – This executable takes ‘snapped.csv’ and airport map as inputs. It uses these inputs to extract features of aircrafts while they are taxiing. Then, these features are printed out into the file named ‘features.csv’. You should be able to observe all features stated in **Appendix B**.

Appendix B: Descriptions of airport data features

Features	Type	Description
1. id	String	Unique ID of the flight
2. type	String	‘arrival’ or ‘departure’
3. taxi_time	Numerical	Taxi time in minutes
4. gate (block) hour	Numerical	Hour at the gate
5. runway hour	Numerical	Hour when reaching the runway

6. rwy_day	Numerical	Day when reaching the runway
7. distance	Numerical	Taxi distance
8. angle	Numerical	Sum of angles along the route
9. angle error	Numerical	Indication if there was a pushback
10. shortest path	Numerical	Shortest path distance between start_node/end_node
11. distance_gate	Numerical	Distance around the gate
12. distance_long	Numerical	Sum of long straight segments>500m
13. distance_else	Numerical	Distance excluding gate and long
14. rwy	String	Name of the runway we are using (for multi runway airports)
15. rwy_num	Numerical	Number of the runway we are using
16. operation_mode	Numerical	Runway operation mode (for multi runway airports, e.g. rwy 1 for departures, rwy 2 for arrivals)
17. other_moving_ac	Numerical	Number of aircraft moving at the same time
18. QDepDep	Numerical	Number of other aircraft that reach runway and depart while current aircraft is on the way to runway
19. QDepArr	Numerical	Number of other aircraft that arrive at stand while current aircraft is on the way to runway
20. QArrDep	Numerical	Number of other aircraft that reach runway and depart while current aircraft is on the way to stand
21. QArrArr	Numerical	Number of other aircraft that arrive at stand while current aircraft is on the way to stand
22. NDepDep	Numerical	Number of other aircraft on the way to runway when current aircraft pushes back
23. NDepArr	Numerical	Number of other aircraft on the way to stand when current aircraft pushes back
24. NArrDep	Numerical	Number of other aircraft on the way to runway when current aircraft lands and starts taxiing
25. NArrArr	Numerical	Number of other aircraft on the way to stand when current aircraft lands and starts taxiing

Appendix C: Steps to success (For information only)

1. Clarify terms and concepts
2. Define the problem (divide into secondary problems)
3. Analyse the problem by brainstorming (give possible explanations)
4. Make a clear strategy on how to solve each secondary problem by clarifying data sets, theories, machine learning tools and software that may needed.
5. Always check the codes to avoid severe bugs.
6. Collect information before next session.
7. Report the outcome of research.