Adam Tobias 18/07/2025

# Pulmonary Embolism Binary Classification Using Logistic Regression, K Nearest Neighbours, and a Neural Network

Skip to discussion

## *Introduction*

Working as an FY1 on general medicine wards – I found that pulmonary embolism was a very common differential. It is raised in a wide multitude of inpatient scenarios from isolated tachycardia to progressive hypoxia (both of which are an everyday occurrence on the wards) among a host of other presentations. Many CTPAs are performed, of which some low risk scans may be doing more harm than good (Aksu et al., 2025). Given the importance of the diagnosis along with the seemingly vast array of presentations and being discouraged from performing D-Dimers on inpatients, I wanted to try my hand at creating a binary classification model using only basic patient details (age, sex) along with pertinent history/examination findings and observations.

While I found some datasets online that looked ideal (including INSPECT with EHR data with 19,402 patients and MIMIC-IV with over 200,000 patients), I am in the application process of requesting data. In the meantime, I found a much smaller dataset that I thought I could run a trial which was fully open source without application. (Ebrahimi, Soroor Laffafchi and kafan, 2022)

## *Aim*

Develop and compare binary classification algorithms using only bedside clinical data to support early PE risk stratification

## *Study Background*

Demographic: Sina educational hospital, Tehran. N=925 (Laffafchi, Ebrahimi and Kafan, 2024)

Inclusion criteria:

- Inpatients and outpatients with suspected PE between 01/03/2019 – 01/03/2021
    - Note that this was during COVID
- Patients with valid hospital health medical records

- Patients who have been undergo to CTA imaging and their reports are clear and available
- Patients for whom laboratory test results are available.

Exclusion criteria:

- Lack of patients EHR data
- Lack of access to laboratory test results or incomplete information
- Lack of clarity in CTA imaging results and unspecified results
- Patients who were unable to carry out angiographic imaging for some reason.

It is unclear how the data was collected ie manually or scraped from EHR. Additionally, observations seem to be picked from one moment in time but it is unclear how this was chosen.

## *Data Extraction/Preparation*

In my analysis, I did not use blood tests/ past medical history or ICU related data. The columns I extracted were: dyspnoea, sex, fever, chest pain, age, RR, Sat, pulse rate (“PR”), oedema and whether DVT was suspected, as well as if PE was seen on CTPA (“CTA”).

Data head and info

| | Sex | Dyspnea | Fever | Edema | Chest pain | DVT Suspect | Age | PR | RR | Sat | CTA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Non-null count | 925 | 925 | 923 | 925 | 924 | 925 | 925 | 925 | 925 | 924 | 925 |
| Data type | Object | Int64 | Float64 | Int64 | Float64 | Int64 | Int64 | Int64 | Object | Float64 | Int64 |
| | F | 1 | 0.0 | 0 | 0.0 | 0 | 64 | 78 | 18 | 82.0 | 1 |
| | M | 1 | 0.0 | 0 | 0.0 | 0 | 45 | 86 | 20 | 95.0 | 0 |
| | F | 1 | 0.0 | 0 | 0.0 | 1 | 48 | 80 | 18 | 82.0 | 0 |
| | F | 0 | 0.0 | 0 | 0.0 | 0 | 45 | 79 | 16 | 97.0 | 0 |
| | F | 1 | 1.0 | 0 | 0.0 | 0 | 40 | 57 | 10 | 100.0 | 0 |

Given sex, and RR were classed as objects, I converted them to numeric. I then dropped the Na values and the data which did not make logical sense/extreme outliers (eg sats over 100% or less than 60%).

## *Feature Descriptions*

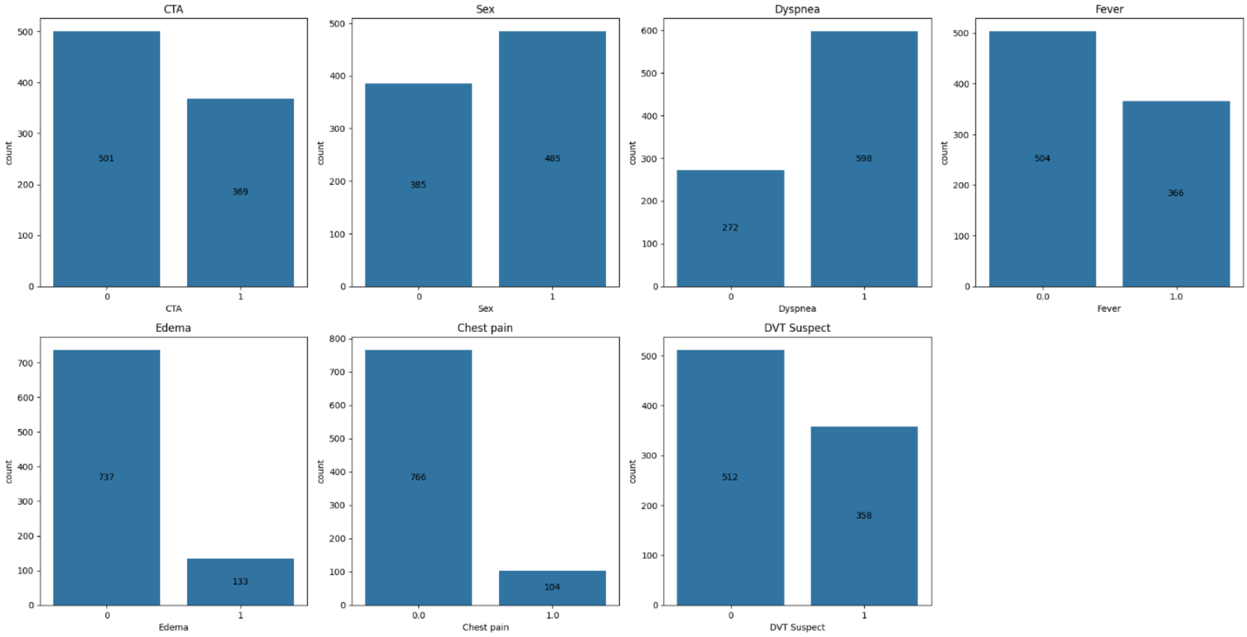


Figure 1.0: Bar Plot of Count of Binary Features

Note that the majority (no PE) on CTPA percentage is 57.6%, with a positive PE on 42.4% in this dataset. 56% of the dataset are males. 42% of the dataset have a fever, which intuitively seems high (may be due to large number of COVID patients).

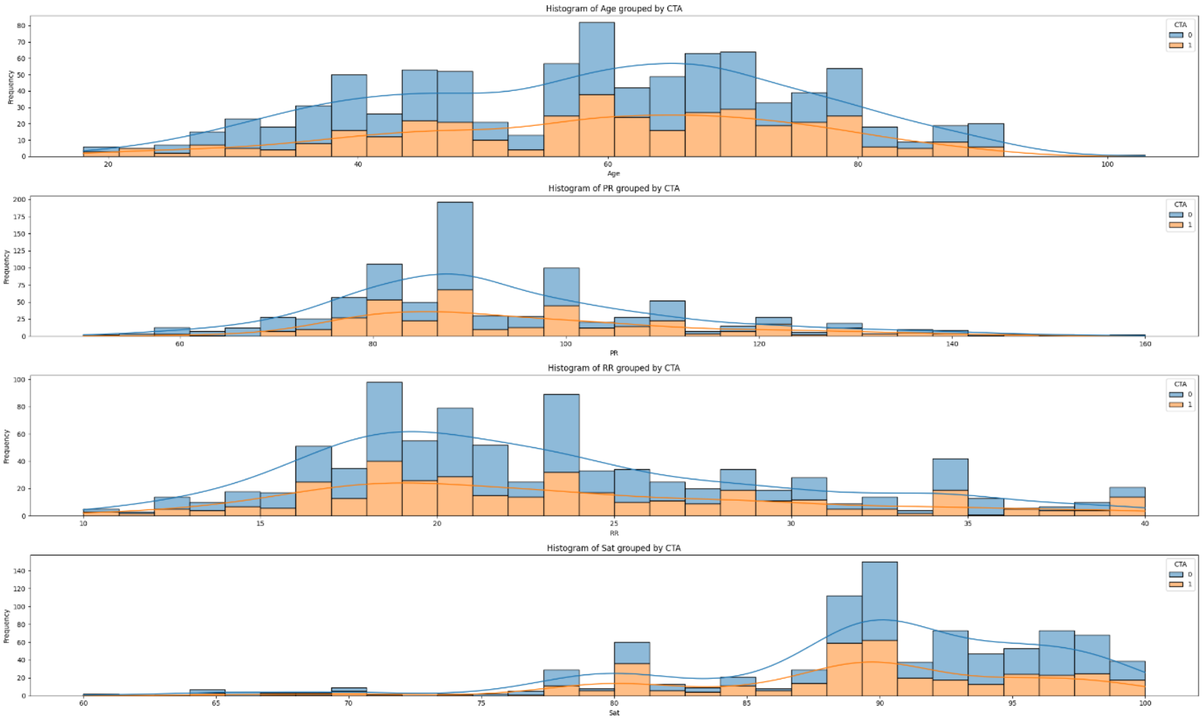


Figure 2: Histograms of Continuous Features Split by CTA results

Age appears to have a bimodal distribution, whereas RR and Sat appear slightly left skewed and Sat with a right skew.

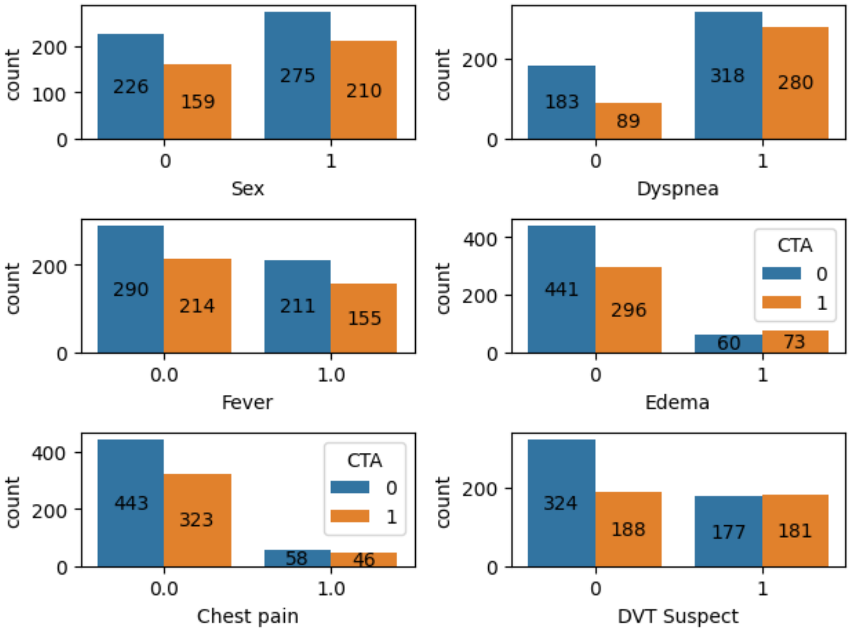
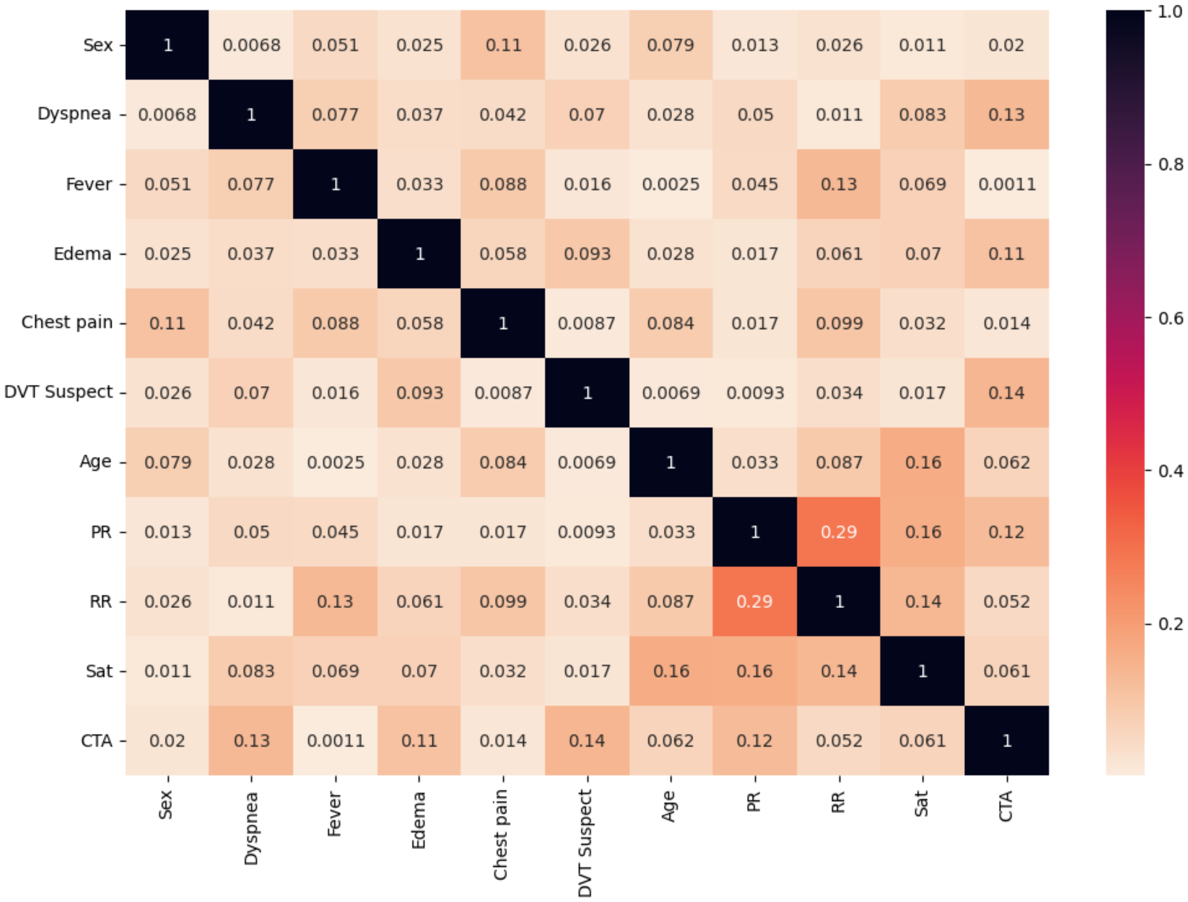


Figure 3: Bar Plot of Count of Binary Features Grouped by CTA

Visually, it appears that dyspnoea, oedema, chest pain and whether DVT suspected could be useful for predicting CTPA result.



| | Sex | Dyspnea | Fever | Edema | Chest pain | DVT Suspect | Age | PR | RR | Sat | CTA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Sex | 1 | 0.0068 | 0.051 | 0.025 | 0.11 | 0.026 | 0.079 | 0.013 | 0.026 | 0.011 | 0.02 |
| Dyspnea | 0.0068 | 1 | 0.077 | 0.037 | 0.042 | 0.07 | 0.028 | 0.05 | 0.011 | 0.083 | 0.13 |
| Fever | 0.051 | 0.077 | 1 | 0.033 | 0.088 | 0.016 | 0.0025 | 0.045 | 0.13 | 0.069 | 0.0011 |
| Edema | 0.025 | 0.037 | 0.033 | 1 | 0.058 | 0.093 | 0.028 | 0.017 | 0.061 | 0.07 | 0.11 |
| Chest pain | 0.11 | 0.042 | 0.088 | 0.058 | 1 | 0.0087 | 0.084 | 0.017 | 0.099 | 0.032 | 0.014 |
| DVT Suspect | 0.026 | 0.07 | 0.016 | 0.093 | 0.0087 | 1 | 0.0069 | 0.0093 | 0.034 | 0.017 | 0.14 |
| Age | 0.079 | 0.028 | 0.0025 | 0.028 | 0.084 | 0.0069 | 1 | 0.033 | 0.087 | 0.16 | 0.062 |
| PR | 0.013 | 0.05 | 0.045 | 0.017 | 0.017 | 0.0093 | 0.033 | 1 | 0.29 | 0.16 | 0.12 |
| RR | 0.026 | 0.011 | 0.13 | 0.061 | 0.099 | 0.034 | 0.087 | 0.29 | 1 | 0.14 | 0.052 |
| Sat | 0.011 | 0.083 | 0.069 | 0.07 | 0.032 | 0.017 | 0.16 | 0.16 | 0.14 | 1 | 0.061 |
| CTA | 0.02 | 0.13 | 0.0011 | 0.11 | 0.014 | 0.14 | 0.062 | 0.12 | 0.052 | 0.061 | 1 |

Figure 4: Correlation Heatmap of Features

The most notable correlations between all features include PR and RR, PR and Sat, Age and Sat.

| Feature | Correlation | P-value |
|---|---|---|
| Age | 0.062 | 0.065476 |
| PR | 0.124 | 0.000234 |
| RR | 0.052 | 0.128856 |
| Sat | -0.061 | 0.071188 |

Table 2: Point-Biserial Correlation of Continuous Features Against CTA

| Feature | F Value | P-value |
|---|---|---|
| Sex | 0.35 | value = 0.553746 |
| Dyspnea | 15.46 | value = 0.000091 |
| Fever | 0.0011 | value = 0.974044 |
| Edema | 10.09 | value = 0.001540 |
| Chest pain | 0.16 | value = 0.689879 |
| DVT Suspect | 16.80 | value = 0.000045 |

ANOVA (F-test) for Binary Features Against CTA

Notable features:

- PR: Correlation = 0.124, p-value = 0.000234
- Dyspnea: F = 15.46, p-value = 0.000091
- Edema: F = 10.09, p-value = 0.001540
- DVT Suspect: F = 16.80, p-value = 0.000045

## *Feature Combination/Manipulation*

Given that both the dataset and the number of features I extracted was small, I combined them in multiple ways, including multiplying the continuous variables together and creating filters (eg tachycardia HR>100 and has dyspnoea). I also created a modified NEWS score with available observations.

Given that the saturations appeared right skewed, I created a new feature with the log of Sat.

Given that RR and PR appeared left skewed, I performed log on their flip.

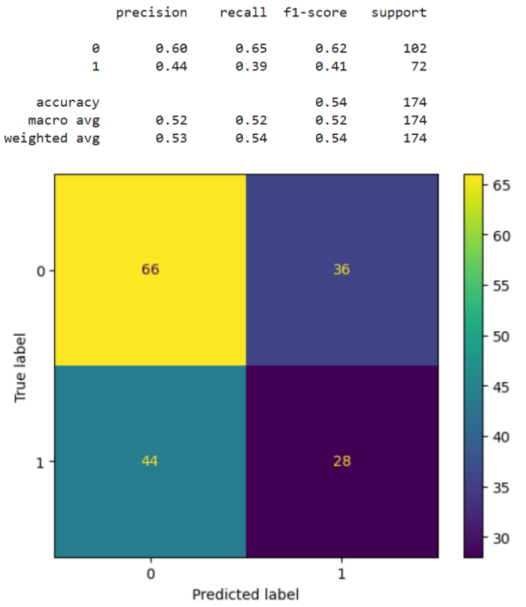
## *Model Creation*

In order to create a binary classification model – I used three suitable algorithms (logistic regression, KNN and a neural network).

### Logistic regression

I firstly used a logistic regression algorithm with the base features I extracted ("Sex", "Dyspnea", "Fever", "Edema", "Chest pain", "DVT Suspect", "Age", "PR", "RR", "Sat")

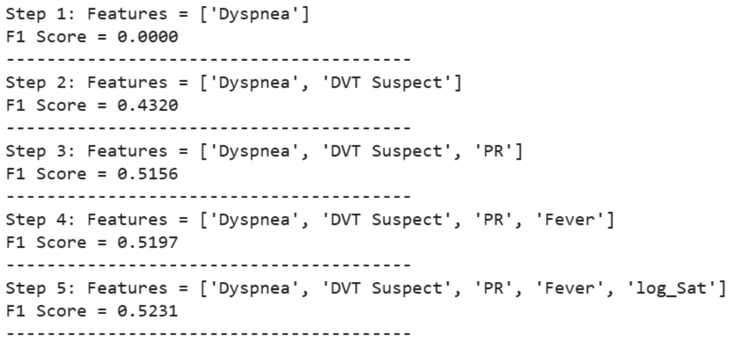
This model had an accuracy of 54% with an f1 of 0.41. It is therefore slightly better than randomly guessing but would have achieved a higher overall accuracy if guessed "no PE" everytime (figure 5)



```
              precision    recall  f1-score   support

           0       0.60      0.65      0.62       102
           1       0.44      0.39      0.41        72

    accuracy                           0.54       174
   macro avg       0.52      0.52      0.52       174
weighted avg       0.53      0.54      0.54       174
```

Figure 5: Confusion Matrix and Metrics of Logistic Regression Model Using Base Features

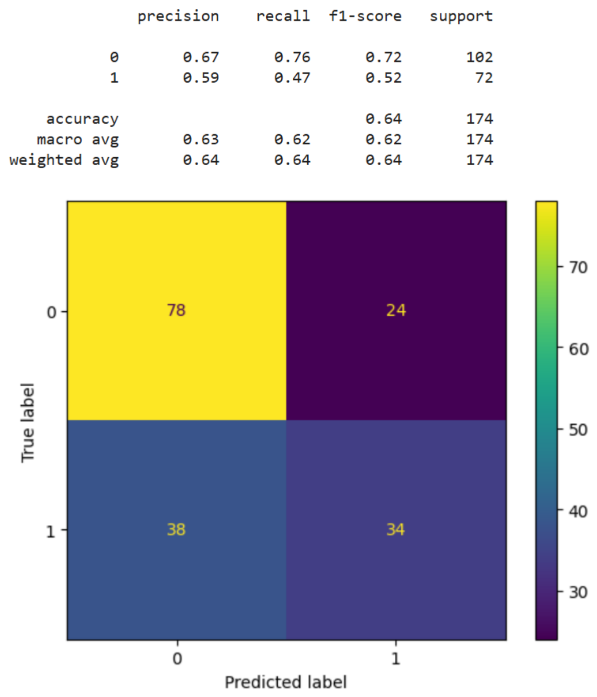
I then performed a forward stepwise approach to feature selection and included the modified variables.



```
Step 1: Features = ['Dyspnea']
F1 Score = 0.0000
----------------------------------------
Step 2: Features = ['Dyspnea', 'DVT Suspect']
F1 Score = 0.4320
----------------------------------------
Step 3: Features = ['Dyspnea', 'DVT Suspect', 'PR']
F1 Score = 0.5156
----------------------------------------
Step 4: Features = ['Dyspnea', 'DVT Suspect', 'PR', 'Fever']
F1 Score = 0.5197
----------------------------------------
Step 5: Features = ['Dyspnea', 'DVT Suspect', 'PR', 'Fever', 'log_Sat']
F1 Score = 0.5231
----------------------------------------
```

Figure 6: Forward Stepwise Feature Selection

I then created another logistic regression model with these features.

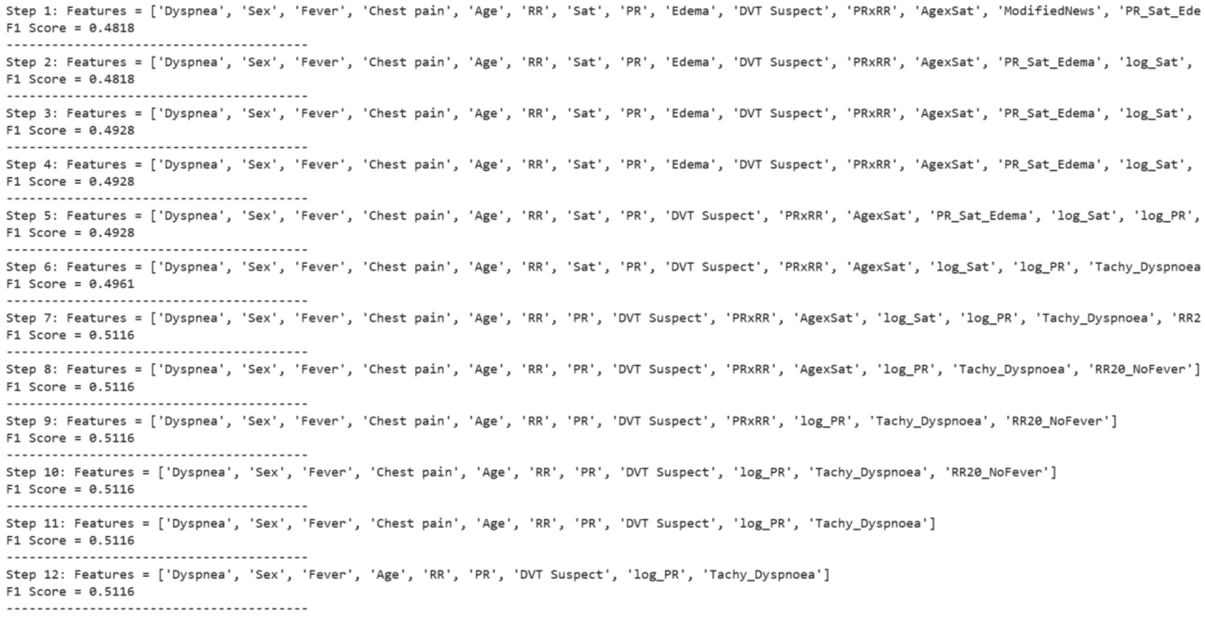


```
              precision    recall  f1-score   support

           0       0.67      0.76      0.72       102
           1       0.59      0.47      0.52        72

    accuracy                           0.64       174
   macro avg       0.63      0.62      0.62       174
weighted avg       0.64      0.64      0.64       174
```

Figure 7: Confusion Matrix and Metrics of Logistic Regression Model Using Forward Stepwise Selection

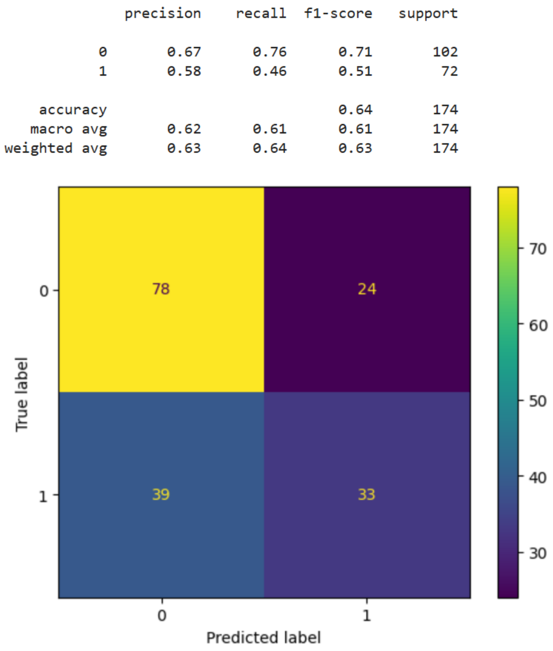
After optimising which attributes to use in the model, it performed much better than previously. Now it performed with an accuracy=0.64 (from 0.54) with an f1=0.52 (from 0.41).

I attempted the same with backwards stepwise selection. If after removing the least useful feature, the model had the same F1 score, I removed it.



```
Step 1: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'Edema', 'DVT Suspect', 'PRxRR', 'AgexSat', 'ModifiedNews', 'PR_Sat_Ede
F1 Score = 0.4818
----------------------------------------
Step 2: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'Edema', 'DVT Suspect', 'PRxRR', 'AgexSat', 'PR_Sat_Edema', 'log_Sat',
F1 Score = 0.4818
----------------------------------------
Step 3: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'Edema', 'DVT Suspect', 'PRxRR', 'AgexSat', 'PR_Sat_Edema', 'log_Sat',
F1 Score = 0.4928
----------------------------------------
Step 4: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'Edema', 'DVT Suspect', 'PRxRR', 'AgexSat', 'PR_Sat_Edema', 'log_Sat',
F1 Score = 0.4928
----------------------------------------
Step 5: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'DVT Suspect', 'PRxRR', 'AgexSat', 'PR_Sat_Edema', 'log_Sat', 'log_PR',
F1 Score = 0.4928
----------------------------------------
Step 6: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'Sat', 'PR', 'DVT Suspect', 'PRxRR', 'AgexSat', 'log_Sat', 'log_PR', 'Tachy_Dyspnoea
F1 Score = 0.4961
----------------------------------------
Step 7: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'PR', 'DVT Suspect', 'PRxRR', 'AgexSat', 'log_Sat', 'log_PR', 'Tachy_Dyspnoea', 'RR2
F1 Score = 0.5116
----------------------------------------
Step 8: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'PR', 'DVT Suspect', 'PRxRR', 'AgexSat', 'log_PR', 'Tachy_Dyspnoea', 'RR20_NoFever']
F1 Score = 0.5116
----------------------------------------
Step 9: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'PR', 'DVT Suspect', 'PRxRR', 'log_PR', 'Tachy_Dyspnoea', 'RR20_NoFever']
F1 Score = 0.5116
----------------------------------------
Step 10: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'PR', 'DVT Suspect', 'log_PR', 'Tachy_Dyspnoea', 'RR20_NoFever']
F1 Score = 0.5116
----------------------------------------
Step 11: Features = ['Dyspnea', 'Sex', 'Fever', 'Chest pain', 'Age', 'RR', 'PR', 'DVT Suspect', 'log_PR', 'Tachy_Dyspnoea']
F1 Score = 0.5116
----------------------------------------
Step 12: Features = ['Dyspnea', 'Sex', 'Fever', 'Age', 'RR', 'PR', 'DVT Suspect', 'log_PR', 'Tachy_Dyspnoea']
F1 Score = 0.5116
----------------------------------------
```

Figure 8: Backwards Stepwise Feature Selection



```
              precision    recall  f1-score   support

           0       0.67      0.76      0.71       102
           1       0.58      0.46      0.51        72

    accuracy                           0.64       174
   macro avg       0.62      0.61      0.61       174
weighted avg       0.63      0.64      0.63       174
```

Figure 9: Confusion Matrix and Metrics of Logistic Regression Model Using Backwards Stepwise Selection

This performed similarly to the forwards selection – accuracy=0.64 (0.64), f1=0.51 (0.52).

# KNN

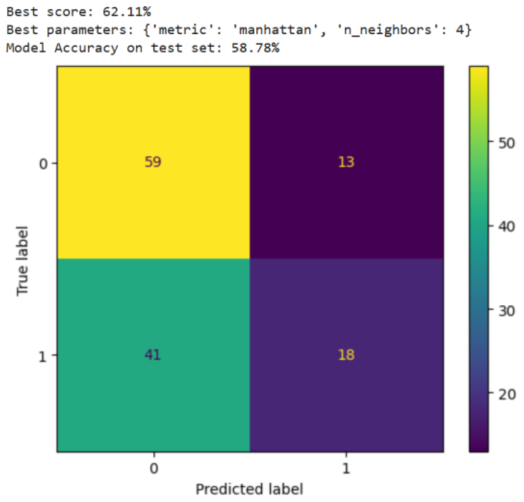
After evaluating logistic regression, I applied a k-Nearest Neighbours (KNN) algorithm to see how it would perform on this classification task. KNN does not rely on assumptions about the distribution of the data. This may be useful in this case given the wide variety of clinical presentations of PEs.

## Initial Feature-Level Exploration

I began by testing the predictive strength of each feature individually using a k = 3. Each model was trained using only 1 feature at a time to see how well it alone could classify the CTPA outcome. Only 3 features had individual predictive accuracies >50%, suggesting that they did not hold strong discriminatory power individually.

## KNN Model Using All Features

I tried training a model with all the features available, with min-max scaling. As part of this, I performed a grid search for cross validation to optimise K and hyperparameters using (n-neighbours 1 to 19 and Minkowski or Manhattan distance metrics). However, this model only achieved an accuracy of 58.78%:

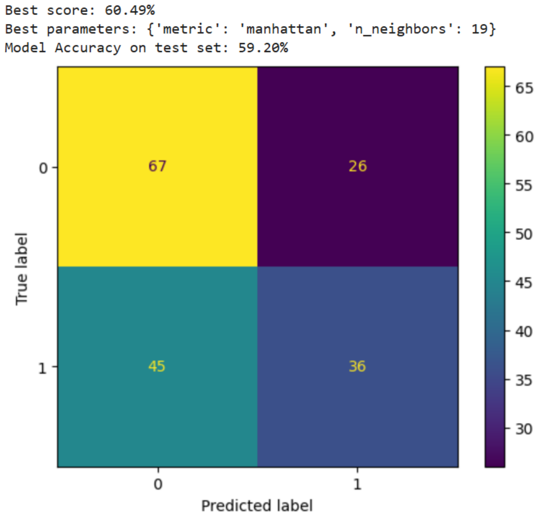


```
Best score: 62.11%
Best parameters: {'metric': 'manhattan', 'n_neighbors': 4}
Model Accuracy on test set: 58.78%
```

Figure 9: Confusion Matrix and Metrics of KNN Model Using All Features

## KNN Model Using Forward Selected Features

Given the underwhelming results from the full-feature model, I retrained the KNN model using only the top features identified from forward stepwise selection. Again, I used a grid search to optimise hyperparameters as before, however this model performed only slightly better at 59.20%:



```
Best score: 60.49%
Best parameters: {'metric': 'manhattan', 'n_neighbors': 19}
Model Accuracy on test set: 59.20%
```

Figure 10: Confusion Matrix and Metrics of KNN Model Using Previous Feature Selection

# Neural Network

To test the performance of a more complex model, I implemented a basic feedforward neural network using TensorFlow/Keras. This model allowed to test the non linear interaction of features and could potentially detect complex relationships missed by linear models like logistic regression.

## Model Architecture

The model consisted of:

- An input layer matching the number of selected features.
- Two layers with ReLU activation: 64 and 32 neurons respectively to help with non linearity/abstract features.
- Dropout layers to reduce overfitting.
- A final output layer with a sigmoid activation for binary classification.

The model was compiled using the Adam optimizer with a learning rate of 0.001.

Result:

- Test Accuracy: 0.5862
- Test AUC: 0.4816

# *Discussion*

## Model Performance Summary

- Logistic Regression with forward stepwise feature selection performed best (accuracy ~64%, F1 ~0.52).
- KNN and Neural Network did not outperform logistic regression - likely due to small sample size and feature limitations.
- Marginal improvement over baseline (guessing all "no PE" cases), suggesting some predictive signal in clinical features.

## Important Predictors

- Dyspnoea, oedema, DVT suspicion, and HR showed strong individual associations with PE. Along with these, fever and log(saturation) were included in the strongest logistic regression model
- Results support known clinical patterns in PE presentations (e.g., SOB, tachycardic, DVT signs, right sided heart strain).

# Limitations

Bias/confounding

- Data collected during COVID, possibly skewing symptom prevalence. This may confound the data as COVID patients presenting to hospital will likely have dyspnoea with potentially dropped O2 sats and high RR/HR. Additionally there may be some population bias, as the patients presenting at this point in time may not reflect the general population (increased prevalence of immunosuppression etc)
- Selection bias – only patients who were imaged were included. This may exclude patients with suspected PE who were not imaged, potentially biasing towards more severe or obvious cases.
- Information/measurement bias - data accuracy depends on the quality of EHR entries and imaging reports. There may be issues in measuring the observations or misclassification of CT reports. Note ~50 rows deleted due to NA fields or strange data entered with questionable data accuracy.

Other dataset limitations/considerations:

- Small sample size (N=925)
- Observations were taken at one ?arbitrary moment in time and does not reflect an entire admission or progressively worsening clinical state.
- Note that the data does not include oxygen requirement. It is unclear whether the saturations included were on or off oxygen.

Model limitations:

- All data from one hospital in Tehran raises questions of external validity. Findings may not generalise to other populations (e.g., UK hospitals).
- Potential overfitting in neural network due to limited data.
- Low feature richness, none of which had very strong predictive value. Blood tests and imaging data purposefully not included (eg. D dimer, CXR, ECG). PMH not included
- No time course/longitudinal data.
- Dataset had a moderate balance (42.4% PE+ve) - higher than typical inpatient settings. Class weighting or other techniques could have been applied to handle this.
- Limited hyperparameter tuning
- The NN was quite simple and may not have been optimal for this dataset

# Future Considerations

- Use larger datasets like INSPECT or MIMIC-IV when access is approved
- The data included whether patients were COVID positive, and this could have been used to potentially mitigate any confounding effect.
- Instead of losing data to NA/strange values – could impute the mean instead of dropping the row.
- Incorporate other bedside data with full and detailed observations (eg including O2 requirement and temp instead of fever).
- Incorporate longitudinal data, such as observations over time.

## *Summary*

PE is a common and often challenging diagnosis due to its varied clinical presentations. My aim was to develop and compare simple binary classification models using basic bedside data - such as age, sex, symptoms, and vital signs - on a publicly available dataset. This would help me learn more about the methods and challenges before performing this on a larger dataset. Among logistic regression, K nearest neighbours, and a neural network, logistic regression performed best, achieving around 64% accuracy and demonstrating some predictive value beyond chance. Key predictors included dyspnoea, oedema, suspected DVT, heart rate, and oxygen saturation. The study was limited by power of the dataset, single centre data collection, and the timing during COVID, which may affect generalisability. My future work should explore larger datasets with more comprehensive clinical variables to improve model performance.

## *Sources*

Aksu, E.A., Uzun, O., İlkyaz Işıksungur, Büşra Adıgüzel Gündoğdu, Furkan Cem Kökten, Burak Özbek and Muzaffer Elmali (2025). Overuse of Computed Tomography Pulmonary Angiography in the Diagnosis of Pulmonary Thromboembolism ‘Real-Life Data’. *International Journal of General Medicine*, [online] Volume 18, pp.1103–1109. doi:https://doi.org/10.2147/ijgm.s499926.

Ebrahimi, A., Soroor Laffafchi and kafan, samira (2022). An Applicable Dataset of Electronic Health Records with a Focus on CTA Results in Pulmonary Embolism Disease. *figshare*. [online] doi:https://doi.org/10.6084/u002Fm9.figshare.21308463.v3.

- Dataset - PE_EHR_CTAresults.xlsx on https://figshare.com/articles/dataset/PE-EHR-CTA_Pulmonary_Embolism_Electronic_Health_Record_dataset_applicable_in_Machine_Learning_and_Neural_Network/21308463

Laffafchi, S., Ebrahimi, A. and Kafan, S. (2024). Efficient management of pulmonary embolism diagnosis using a two-step interconnected machine learning model based on electronic health records data. *Health Information Science and Systems*, 12(1). doi:https://doi.org/10.1007/s13755-024-00276-9.