Predicting patient treatment plan by machine learning models

Jani Juvani

29.12.2023

Data set

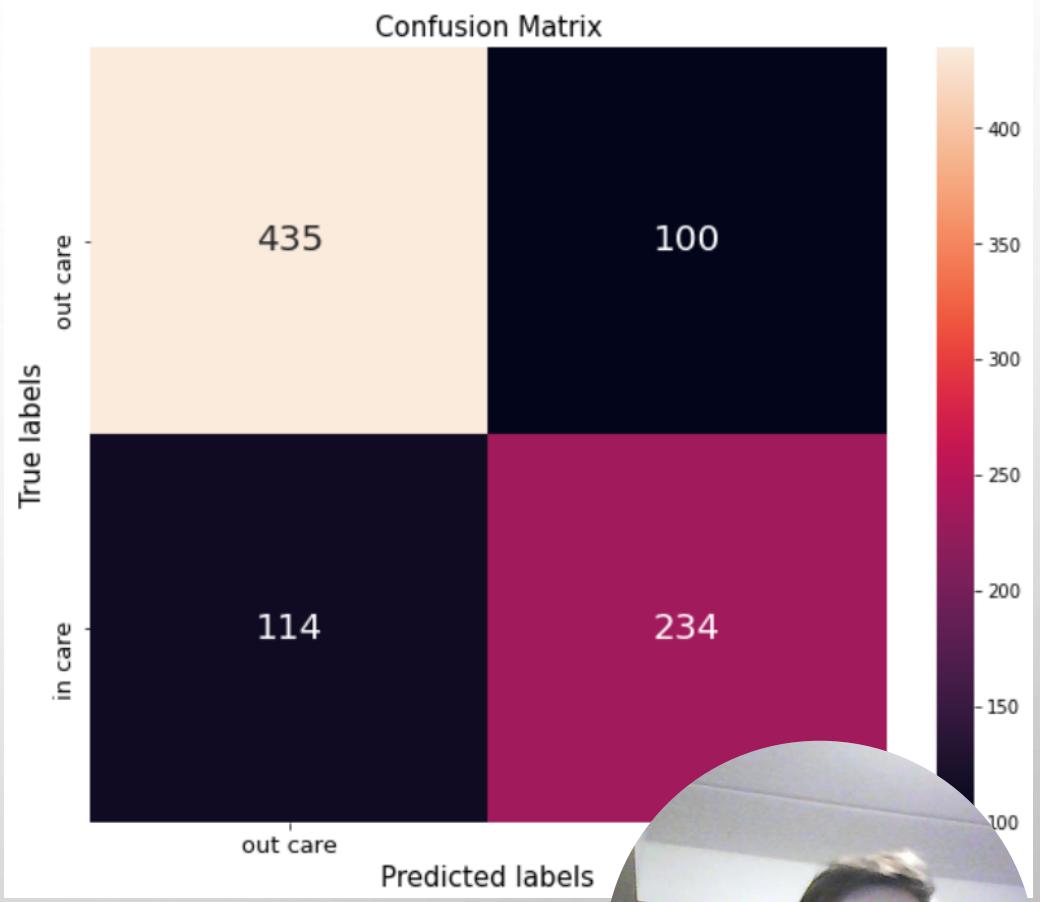
- Consists of 10 features and 1 target variable
  - 8 blood sample measurements
  - Age
  - Sex
  - In care / out care treatment plan (target)
- Collected from Kaggle.com
- In CSV file format

Use case

- To predict patient treatment plan
  - In care
  - Out care
- Helps doctors to make more informed decisions
- Improves the care of patients
  - More ill patients stay in the hospital
- Reduces costs
  - Less unnecessary patients in the hospital
  - Fewer patient callbacks

Solution

- Three different classifier models were examined
  - Logistic regression
  - Support vector machine
  - Deep learning model
- Support vector machine performs best
  - Accuracy 75 %
  - Sensitivity to predict in care 67 %
  - Specificity to predict in care 81 %



Performing the analysis

Architectural choices

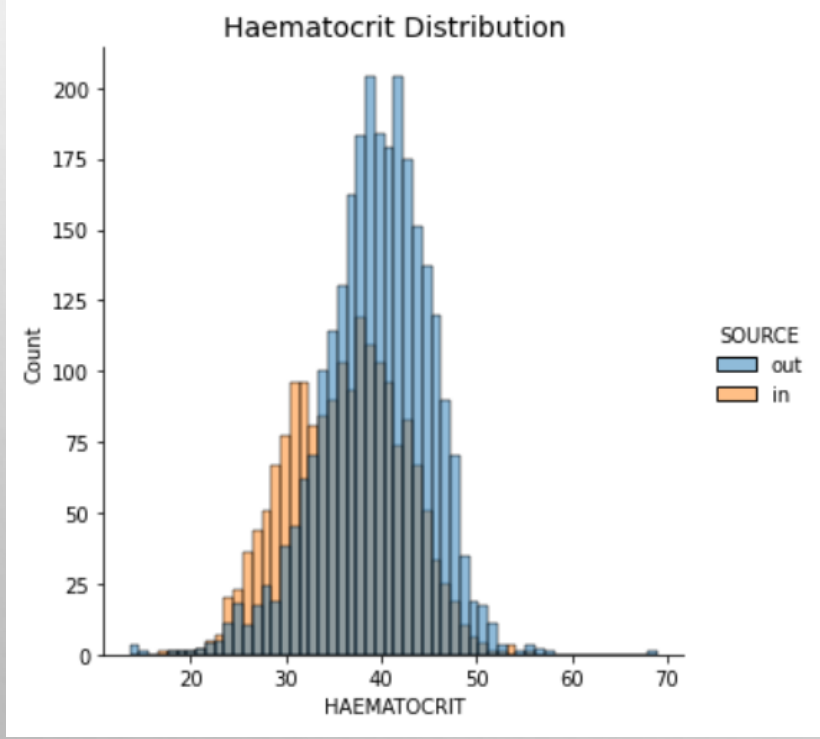
- Data integration
  - Python with Pandas and Scikit-learn libraries
- Data repository
  - Github
- Discovery and exploration
  - Python with Matplotlib and Seaborn libraries
- Actionable insights
  - Python with Scikit-learn and Keras libraries
- Data product
  - Presentation of the results and a Jupyter notebook

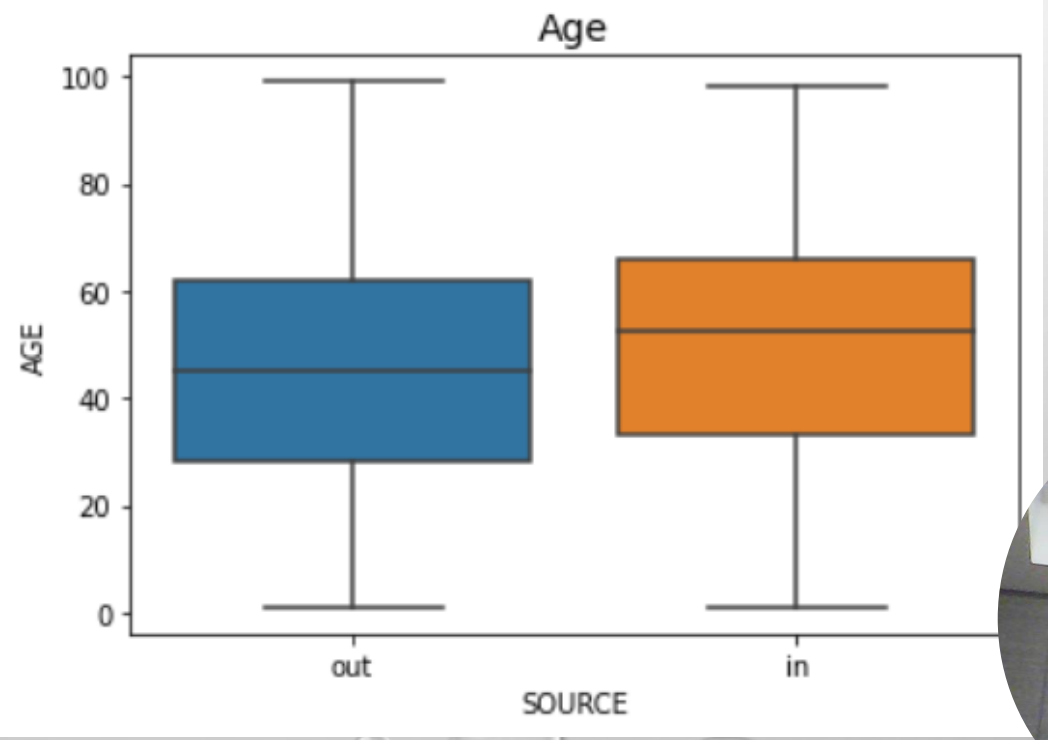
Data pre-processing and feature engineering

- No missing values in data
- Target variable unbalanced, in care 1784/out care 2628
- Encoding character variables
  - Sex M/F → 1/0
  - Target variable in care/out care → 1/0
- Data standardization
  - Features are scaled by removing the mean and scaling to unit variance
- All features selected for prediction models

Data visualization

- Distribution and box plots were used
- Slight differences are observed in distributions with different targets





Models

1. Logistic regression
2. Support vector machine
3. Feed forward neural network
   - 2 dense layers with 20 units and leaky relu activation
   - Output layer with 1 unit and sigmoid activation

- Class weight 1.5 for target variable 1 (in care) in all models
- Data consists of 4412 samples
  - Training data 80 %
  - Test data 20 %

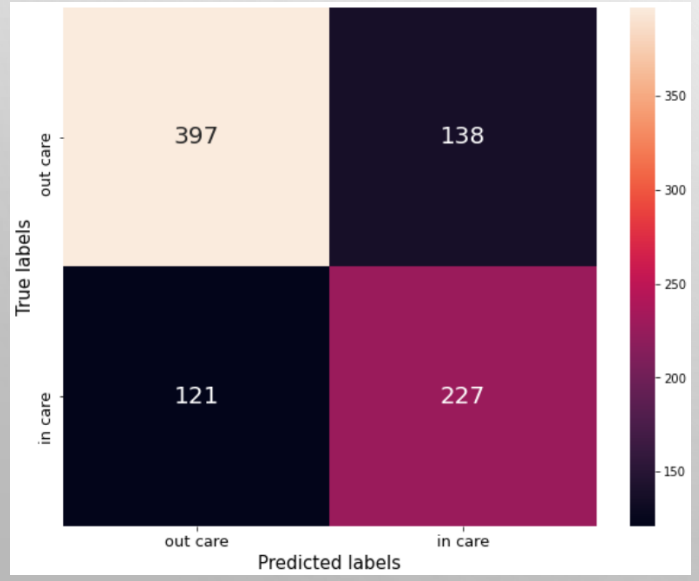
Model performance

Logistic regression

Accuracy 70 %

Sensitivity 65 %

Specificity 74 %

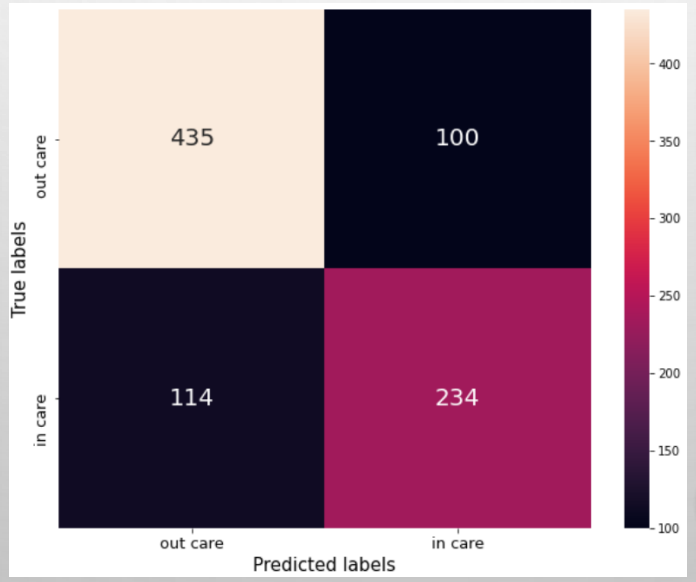


Support vector machine

Accuracy 75 %

Sensitivity 67 %

Specificity 81 %

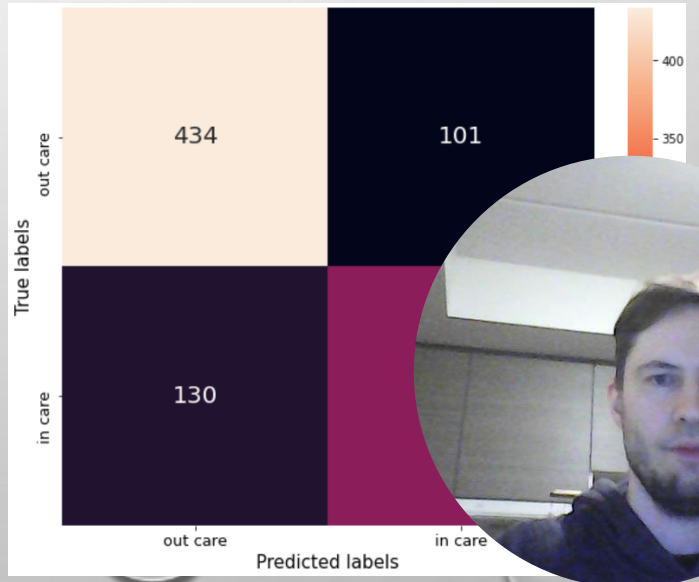


Feed forward NN

Accuracy 73 %

Sensitivity 63 %

Specificity 81 %



Conclusion

- Support vector machine is the best model
  - Acc 75%, Sen 67 %, Spe 81 %
- Non-linear model does not improve model performance
- Feature distributions only slightly different between classes
  - Additional features could improve model performance
- Class weight 1.5 for target variable 1 (in care)
  - Sensitivity can be slightly increased at the expense of accuracy and specificity by raising class weight