Open science in the stroke projects

Anna Laws, Kerry Pearn, Michael Allen

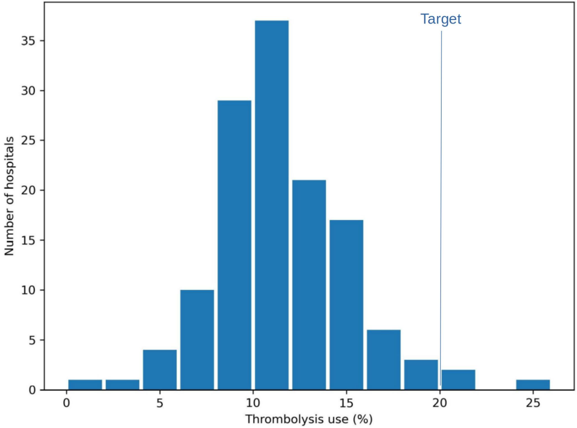
November 2022

Outline

1. Stroke Projects
2. SAMueL
3. Open Science Tooling in Stroke Work
4. The PenCHORD Way?

Stroke Projects

The problem....



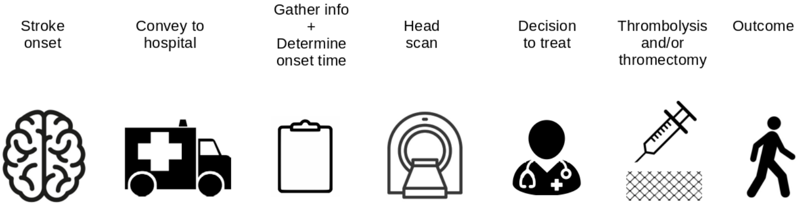
The central problem we are investigating is that there is a NHS target to give clot-busting drugs (*thrombolysis*) to 20% of patients, but actually only 11% patients are receiving them (and this varies from 5% to 25% between hospitals).

Stroke projects

- SAMueL: Stroke Audit Machine Learning
  - Emergency stroke clinical pathway simulation
  - Machine learning to learn and compare clinical decision-making between hospitals
  - Detailed (disability-level) clinical outcome model
  - Health economics model
- OPTIMIST: OPTimising IMplementation of Ischaemic Stroke Thrombectomy
  - Modelling and optimising the pre-hospital emergency stroke pathway.
- Mobile Stroke Units (hopefully!)
- Geographic modelling (update)
- Whole stroke system modelling?

SAMueL

Breaking down the emergency stroke pathway into key steps

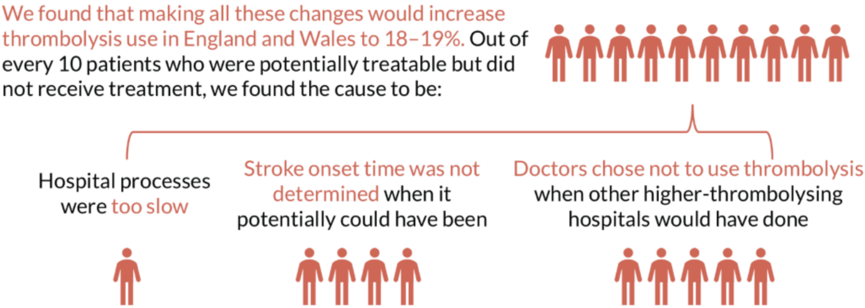


We can model key changes to pathway:

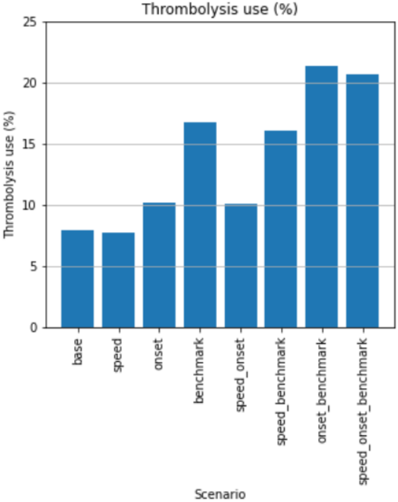
- What if the pathway were faster?
- What if hospital determined the stroke onset time in more patients?
- What if clinical decision-making was like that of *benchmark* hospitals? (Predict what treatment a patient would receive at other hospitals).

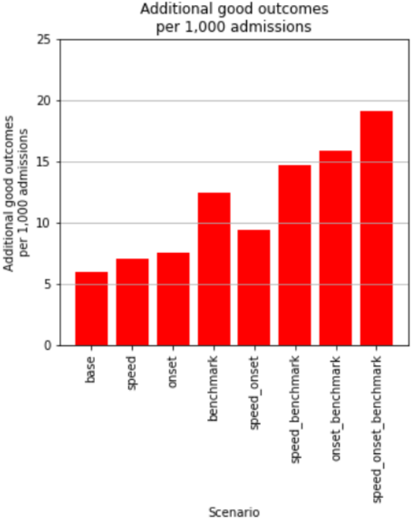
We model these changes with a hospital's own patient population, to allow for inter-hospital variation in patient population characteristics.

SAMueL-1 Summary: What did we find?

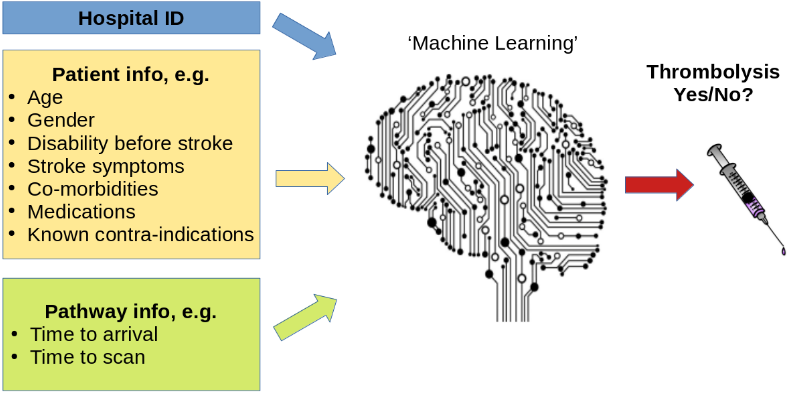


Applying our models at hospital level





Machine learning overview



Machine learning (and nearly all *artificial intelligence*) is based on the simple principle of recognising similarity to what has been seen before.

We accessed 240,000 emergency stroke admissions in England and Wales over three years.

Model accuracy, and simplification

Our machine learning models use XGBoost classification, and are based on all patients who arrive within 4 hours of known stroke onset.

The full model has 61 patient features:

- Overall accuracy = 85.2%
- Best combined sensitivity and specificity = 84.3%
- ROC AUC = 0.921

A simplified model with 8 features

- Overall accuracy = 84.8%
- Best combined sensitivity and specificity = 83.8%
- ROC AUC = 0.916

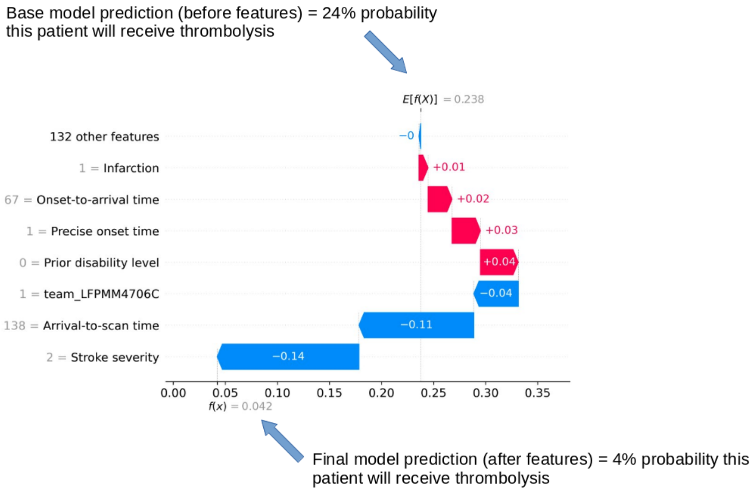
The 8 features of the simplified model are:

1. Arrival-to-scan time
2. Stroke type (infarction/haemorrhage)
3. Stroke severity (NIHSS)
4. Precise or estimated stroke onset time
5. Prior disability level (mRS)
6. Stroke team
7. Use of AF anticoagulants
8. Onset-to-arrival time

There are only very weak correlations between the selected features with no R-squared being greater than 0.05.

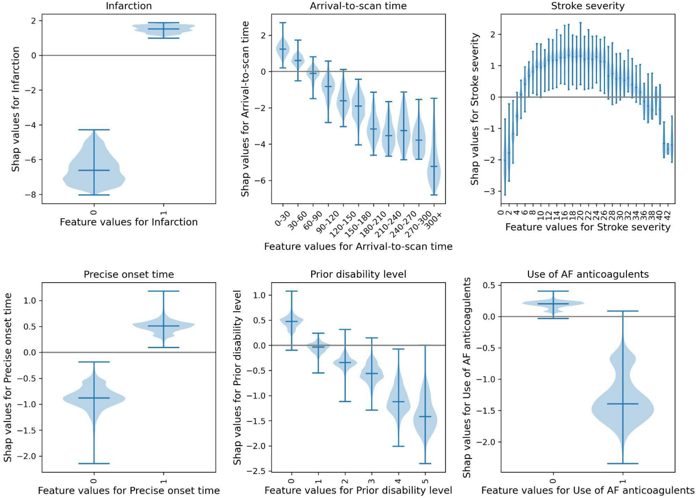
Explaining model predictions with SHAP values

SHAP values show the influence of features (even for *'black box'* models).



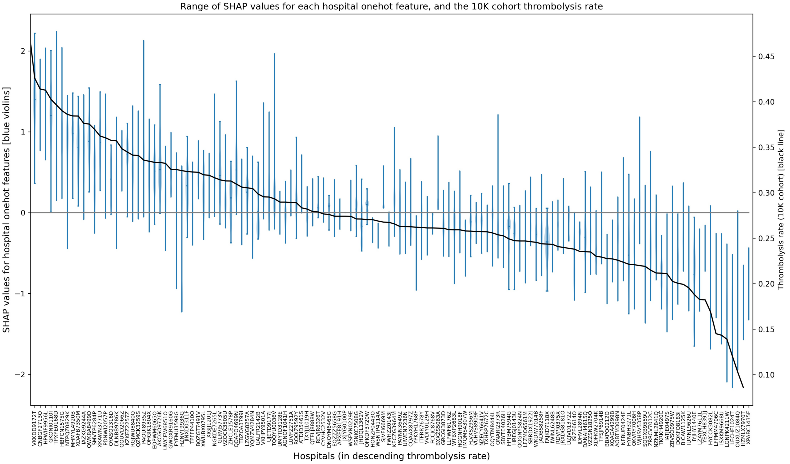
What drives use of thrombolysis across all hospitals?

Note: SHAP values here are *log odds*. Each step-change in value of ±1 changes the chances of receiving thrombolysis about 3-fold. (Plots are in order of feature importance.)



Hospital SHAP values

Hospital SHAP values show a hospital's willingness to use thrombolysis. But different patients will have different hospital SHAP values, reflecting the decision-making at each hopsital.



Investigating how hospitals differ in thrombolysis decision-making: artificial patients

Base patient:

- Onset to arrival = 80 mins
- Arrival to scan = 20 mins
- Infarction = Yes
- NIHSS = 15
- Prior disability level = 0
- *Precise onset time = Yes*
- Use of AF anticoagulents = No

Proportion of hospitals predicted to give thrombolysis:

- Base patient: 99%
- NIHSS = 4: 73%
- Pre-stroke mRS=3: 86%
- Estimated stroke onset time: 64%

Open Science Tooling in Stroke Work

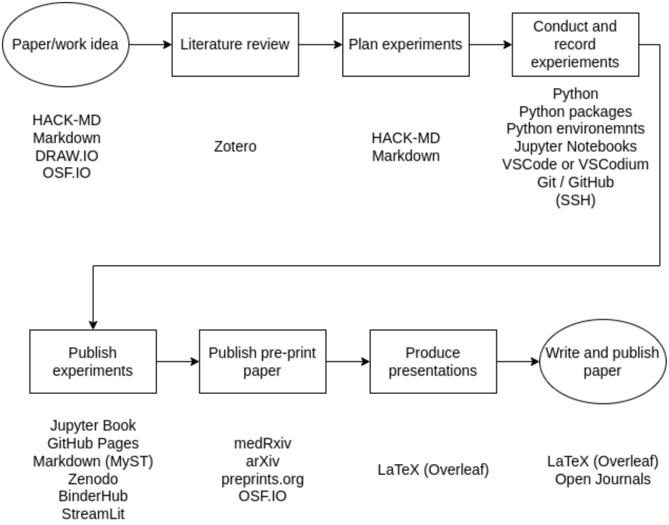
Caveats

- You may know all of this!
- Learning these tools takes time
- We are still learning
- We still shout at things at times
- Other tools exist
- These are probably *not* tools for people who's job keeps them in Microsoft Office most of the day

But.....

- It works; we've used Linux/FOSS for 5+ years
- We just use free/open tools
- The benefits, and quality of what may be produced, far outweigh the challenges

Our stack of tools



Examples of tools

- Jupyter Book & Notebooks: https://samuel-book.github.io/stroke_outcome/intro.html
- GitHub: https://github.com/samuel-book/stroke_outcome
- Zenodo: https://zenodo.org/account/settings/github/
- VS Code
- Overleaf: https://www.overleaf.com/project/630f1aaad6a9ffb27b51490a
- HackMD: https://hackmd.io/@N4jCROVmS9SqGmgj3U66XA/HJPwxmtSs
- Zotero: https://www.zotero.org/groups/4707796/mja_stroke/items/H5JR5N99/library
- Open Print Servers: https://www.medrxiv.org/content/10.1101/2020.07.18.20156653v2
- OSF (Open Science Framework): https://osf.io/dashboard
- Streamlit: https://samuel2-stroke-outcome.streamlit.app/Interactive_demo
- BinderHub: https://github.com/MichaelAllen1966/2004_covid_dialysis

General lessons

- Open Science takes time - quality is not necessarily cheap
  - We tend to go through, and refine, notebooks several times to check for errors, make them clear, clean, and understandable
- Working with others helps
  - We pass notebooks around at times - both for coding and summarising
  - If a mistake gets through, no single person is to blame
  - Psychological support
- Git can be hard! Allow time to get used to it
  - Create a branch for each notebook you work on
  - Commit and push often
  - Visual Studio Code has nice Git and GitHub integration (sorry Tom)
- If you are new to LaTeX, don't start with presentations (or posters)

The PenCHORD Way?

The PenCHORD Way?

- Do we want a set of tools we more actively support and train people in?
- *Supportive* not *coercive*
- If so, what tools?
- What documentation/training/support material?
  - Jupyter Book?
  - *Mattermost* (FOSS Slack alternative for shared online 'live' support
  - Other?
- What gaps do we have?
- How do we involve everyone in developing the PenCHORD Way?