




Best Practices For Prediction Using Observational Data

Jenna Reps



OHDSI PatientLevelPrediction R Package

- We have an end-to-end framework that goes from data extraction to model development and validation.
- This has enabled us to do large-scale empirical analysis into best practices for model development

Design and implementation of a standardized framework to generate and evaluate patient-level prediction models using observational healthcare data 

Jenna M Reps , Martijn J Schuemie, Marc A Suchard, Patrick B Ryan, Peter R Rijnbeek

Journal of the American Medical Informatics Association, Volume 25, Issue 8, August 2018,
Pages 969–975, <https://doi.org/10.1093/jamia/ocy032>

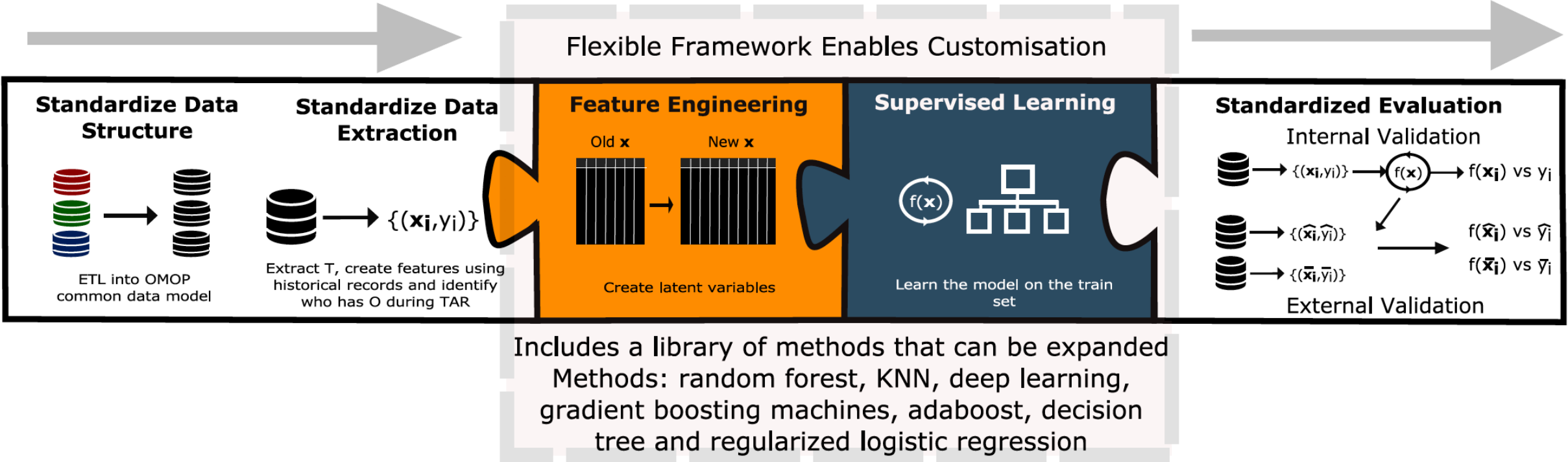
Published: 27 April 2018 **Article history** ▼



OHDSI End-to-end Framework

Patient-Level Prediction Framework

We have developed an end-to-end framework for developing and validating clinically useful prediction models using observational data

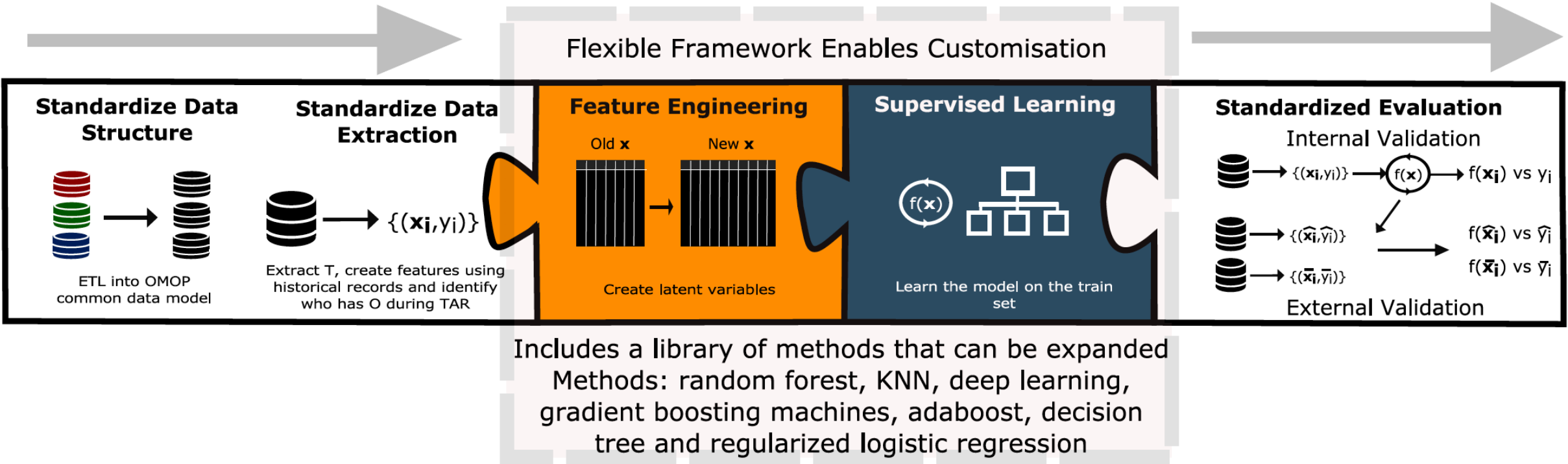




OHDSI End-to-end Framework

Patient-Level Prediction Framework

We have developed an end-to-end framework for developing and validating clinically useful prediction models using observational data



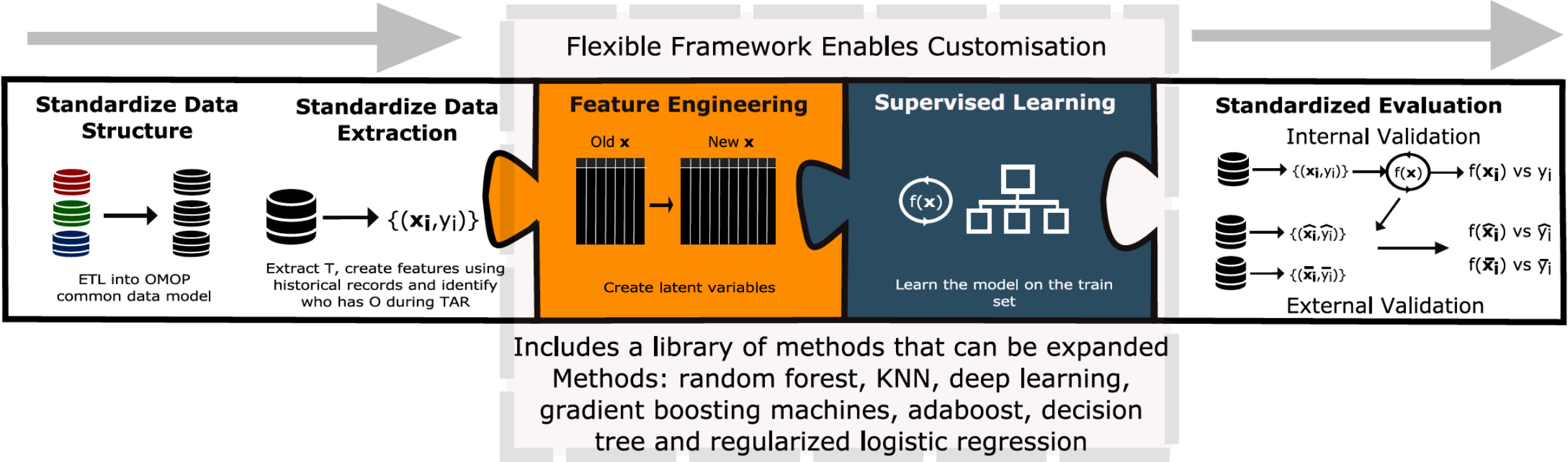
Many parts of the framework involve making subjective choices...



OHDSI End-to-end Framework

Patient-Level Prediction Framework

We have developed an end-to-end framework for developing and validating clinically useful prediction models using observational data



We decided to identify the main choices and have started to investigate them



How We Learn Best Practices

- Using our framework we can measure how a certain choice impacts the model (in terms of calibration and discrimination)
- We have been investigating different choices across a large number of prediction problems
- Most of our empirical investigations involve developing hundreds or thousands of models!

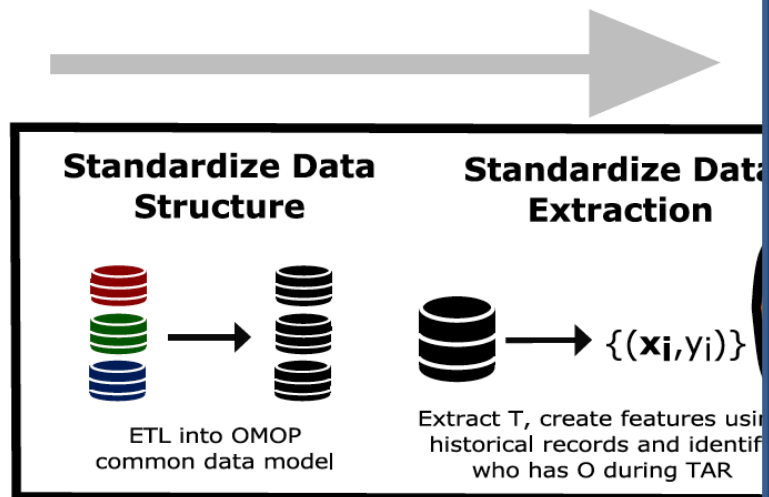




Example Data Extraction

Patient-Level Prediction Framework

We have developed an end-to-end framework for developing and validating clinically useful prediction models using observational data



Q1) What is the best design for extracting labelled data from claims/EHR data?

Q2) How much do the phenotypes impact the models?

Q3) How to address loss to follow-up when using a cohort design?

Q4) Should you use all data recorded prior to index for engineering features or restrict to the last year or last six months or some other time period (addressing left-censoring)?

Q5) What is the gain in using temporal features or measurements?

...



Example Learning

Q1) What is the impact of data size on model development (learning curves)

Q2) Does the test/train/validation design make an impact when developing models in big data?

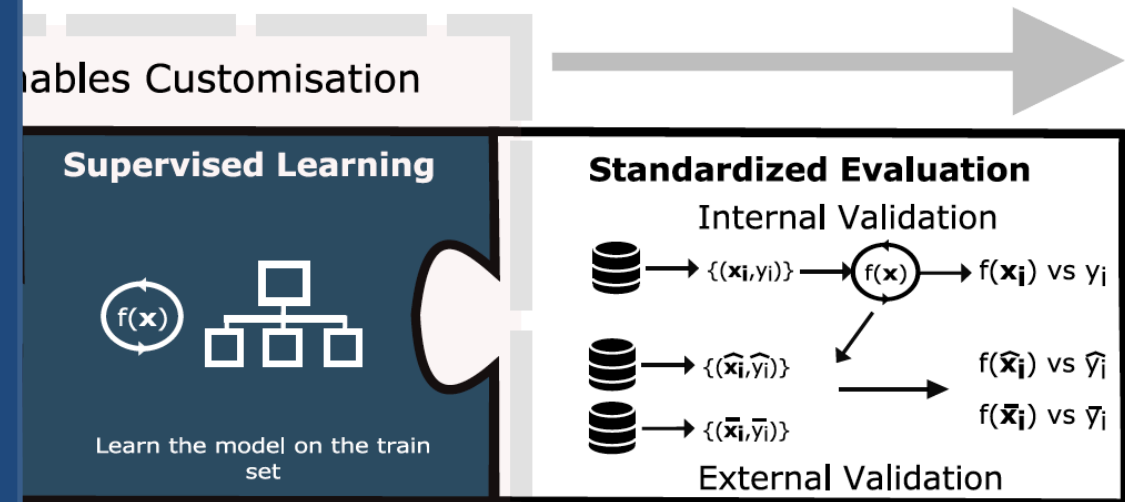
Q3) Can we find hyper-parameter combinations per classifier that consistently lead to good performing models when using claims/EHR data?

Q4) Can deep learning (or other advanced methods) give us significantly better performance?

Q4) How to improve models transportability?

- Can we use transfer learning to learn models using large datasets and fine tune on smaller datasets?
- Can ensembles combining models developed using different databases do better in new data than the single database models?

and validating clinically useful prediction models using



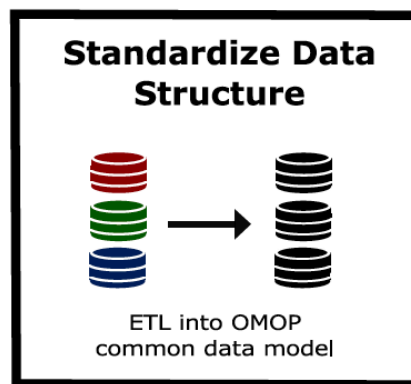
Methods that can be expanded
t, KNN, deep learning,
nes, adaboost, decision
logistic regression



Example Interpretation

Patient-Level Prediction Framework

We have developed an end-to-end framework for developing and validating clinically useful prediction models using observational data



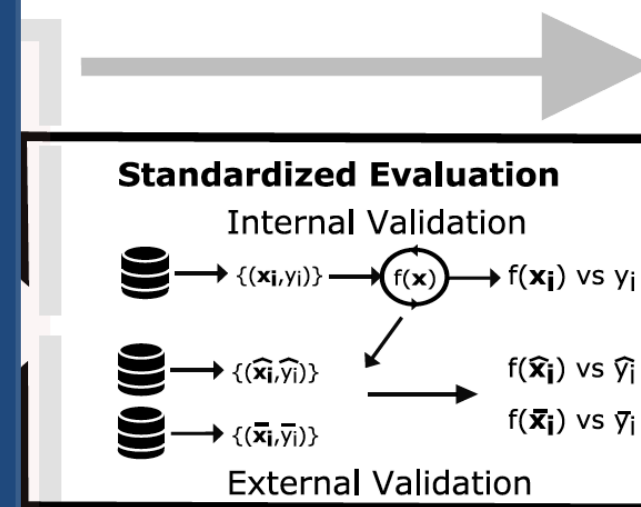
Q1) How should we present models? (e.g., new visualizations)

Q2) How to interpret external validation performance (can we figure out why the performance drops or stays consistent)?

Q3) Is there a feature importance metric we should be using as best practice?

Q4) Is there a way to automatically simplify models?

...





Questions

- Email: jreps@its.jnj.com
- PLP package:
<https://github.com/OHDSI/PatientLevelPrediction/tree/master>

