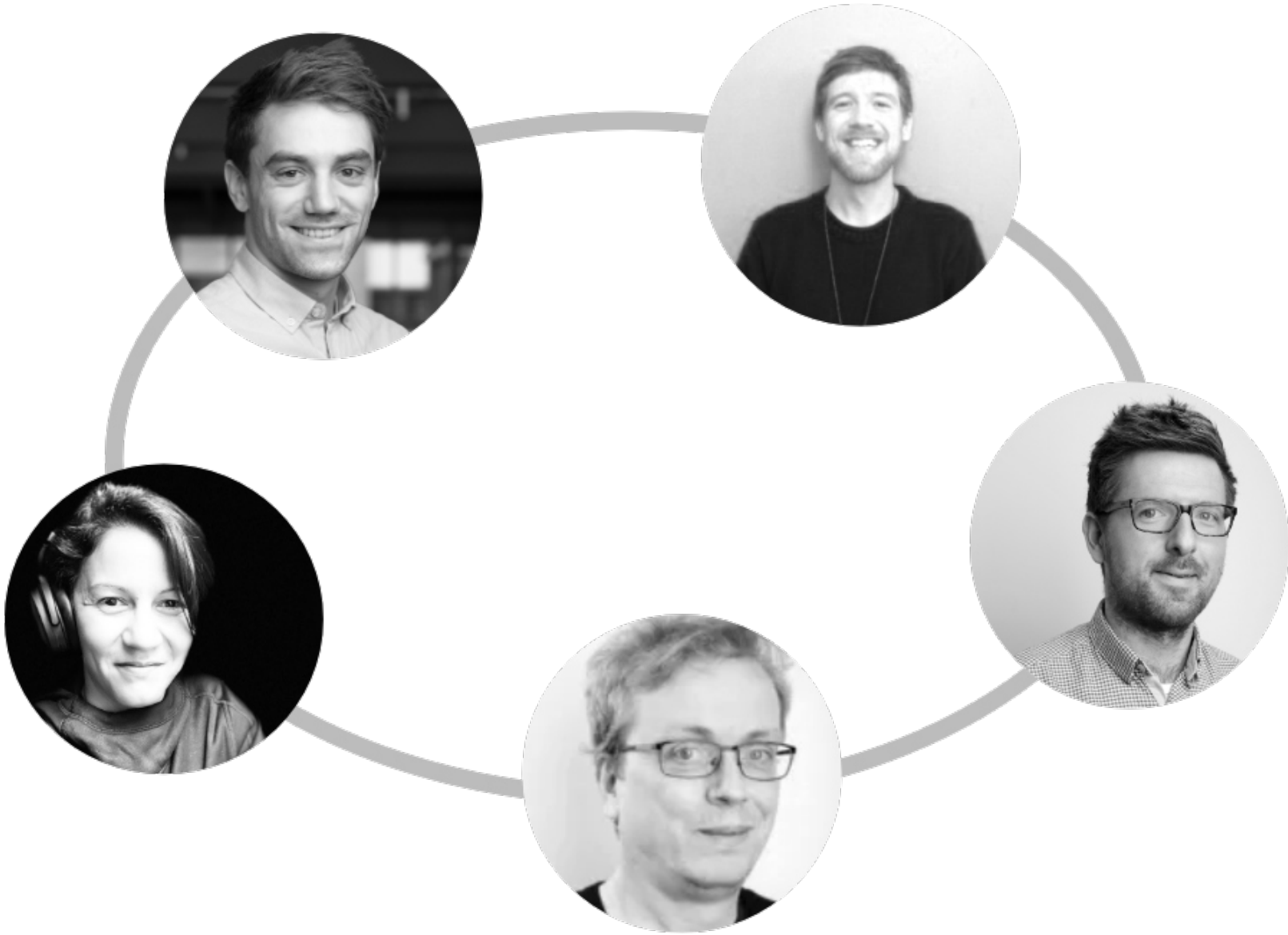
A simulation approach to calculating minimum sample sizes for prediction modelling

The pmsims package for R

Ewan Carr, Gordon Forbes, Diana Shamsutdinova,
Daniel Stahl, and Felix Zimmer

Department of Biostatistics & Health Informatics
King's College London

29th August 2023



30-second version

1. Most prediction models use small samples.
2. Small samples cause overfitting and imprecise estimates.
3. Existing tools can estimate minimum samples for continuous, binary, and survival outcomes.
4. Nothing exists for other models or data types.

We're developing a simulation-based approach that works with any outcome or method.

This talk

1. Background
   - What's the problem we're trying to solve?
   - What solutions currently exist?
2. Our simulation-based approach
   - Workflow and user interface
   - How it compares to other packages
3. Demonstration
4. Development status and next steps

Under construction; feedback welcome.

Most models are developed with inadequate samples

- Small samples the most common cause of bias in 731 models for COVID-19.[3]
- Inadequate samples have been found in:

67% models for COVID-19[3]

56% models using supervised machine learning[4]

73% models in psychiatry[6]

Most models are developed with inadequate samples

- Small samples the most common cause of bias in 731 models for COVID-19.[3]
- Inadequate samples have been found in:

67% models for COVID-19[3]
56% models using supervised machine learning[4]
73% models in psychiatry[6]

Inadequate samples $\rightarrow$ research waste

- Leads to overfitting and inaccurate parameter estimates.
- May generate to inappropriate treatment decisions.
- Data collection can be invasive and inconvenient.

Adequate development samples would improve patient outcomes.

What tools exist?

Most studies ignore sample size.

Or use rules of thumb (e.g., 10 events per variable) that have no rationale in prediction modelling.[2]

RESEARCH ARTICLE

WILEY Statistics in Medicine

Minimum sample size for developing a multivariable prediction model: PART II - binary and time-to-event outcomes

Richard D Riley[1] | Kym IE Snell[1] | Joie Ensor[1] | Danielle L Burke[1] | Frank E Harrell Jr[2] | Karel GM Moons[3] | Gary S Collins[4]

[1]Centre for Prognosis Research, Research Institute for Primary Care and Health Sciences, Keele University, Staffordshire, UK

[2]Department of Biostatistics, Vanderbilt University School of Medicine, Nashville, Tennessee

[3]Julius Centre for Health Sciences and Primary Care, University Medical Centre Utrecht, Utrecht, The Netherlands

[4]Centre for Statistics in Medicine, Nuffield Department of Orthopaedics, Rheumatology and Musculoskeletal Sciences, University of Oxford, Oxford, UK

When designing a study to develop a new prediction model with binary or time-to-event outcomes, researchers should ensure their sample size is adequate in terms of the number of participants (n) and outcome events (E) relative to the number of predictor parameters (p) considered for inclusion. We propose that the minimum values of n and E (and subsequently the minimum number of events per predictor parameter, EPP) should be calculated to meet the following three criteria: (i) small optimism in predictor effect estimates as defined by a global shrinkage factor of ≥ 0.9, (ii) small absolute difference of ≤ 0.05 in the model's apparent and adjusted Nagelkerke's R^2, and (iii) precise estimation of the overall risk in the population. Criteria (i) and (ii) aim to reduce overfitting conditional on a chosen p, and require prespecification of the model's anticipated Cox-Snell R^2, which we show can be obtained from previous studies. The

In 2018, Riley et al. released pmsampsize[5] for R and Stata.

We ❤️ pmsampsize, but. . .

We increasingly need to estimate minimum samples for:

- Other models (e.g., machine learning algorithms, random forests, gradient boosting)
- Other data types (e.g., longitudinal, clustered)

We ❤️ pmsampsize, but. . .

We increasingly need to estimate minimum samples for:

- Other models (e.g., machine learning algorithms, random forests, gradient boosting)
- Other data types (e.g., longitudinal, clustered)

We're developing a simulation-based framework to estimate sample sizes for prediction.

The pmsims package for R

Flexible	Any model or data type
User-friendly	Defaults for common scenarios
Efficient	Estimation via surrogate modelling

Our approach

Setting

1. A **study population** represented by outcome-related individual characteristics (i.e., candidate predictors).
2. A chosen statistical or machine learning **model**.
3. Expected achievable **large-sample performance**, P^*, given population and model.
4. Minimum **acceptable test performance** of the model, P^{OK}.

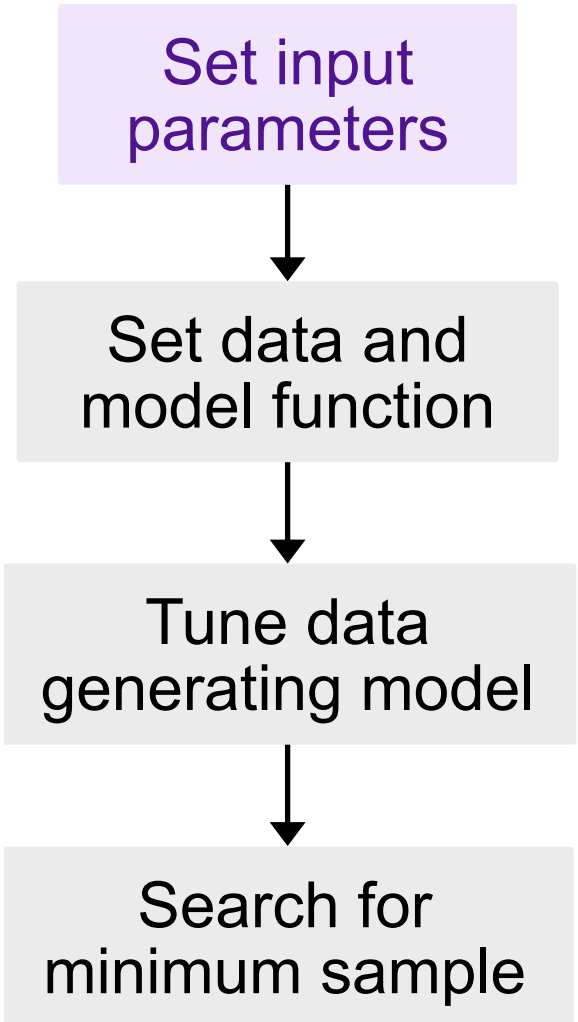
Our approach

Setting

1. A **study population** represented by outcome-related individual characteristics (i.e., candidate predictors).
2. A chosen statistical or machine learning **model**.
3. Expected achievable **large-sample performance**, P^*, given population and model.
4. Minimum **acceptable test performance** of the model, P^{OK}.

Find the minimum sample that ensures test performance of P^{OK} with probability of 80%, given the population, predictors, and P^*.

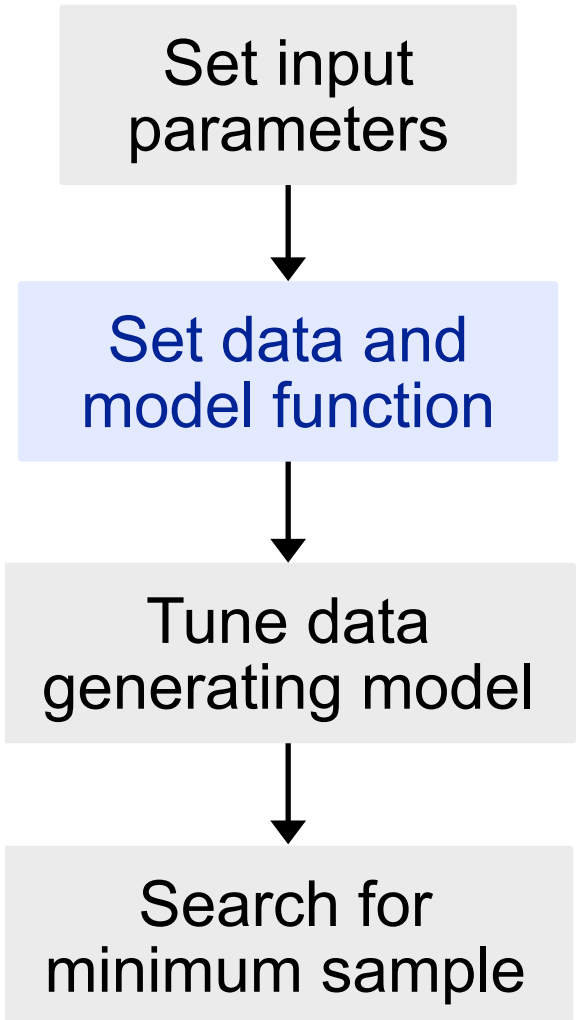
How does it work?



The user specifies:

1. The candidate predictors (number, type)
2. The chosen statistical model
3. The expected large sample performance (P^*)
4. The minimum acceptable performance (P^{OK})

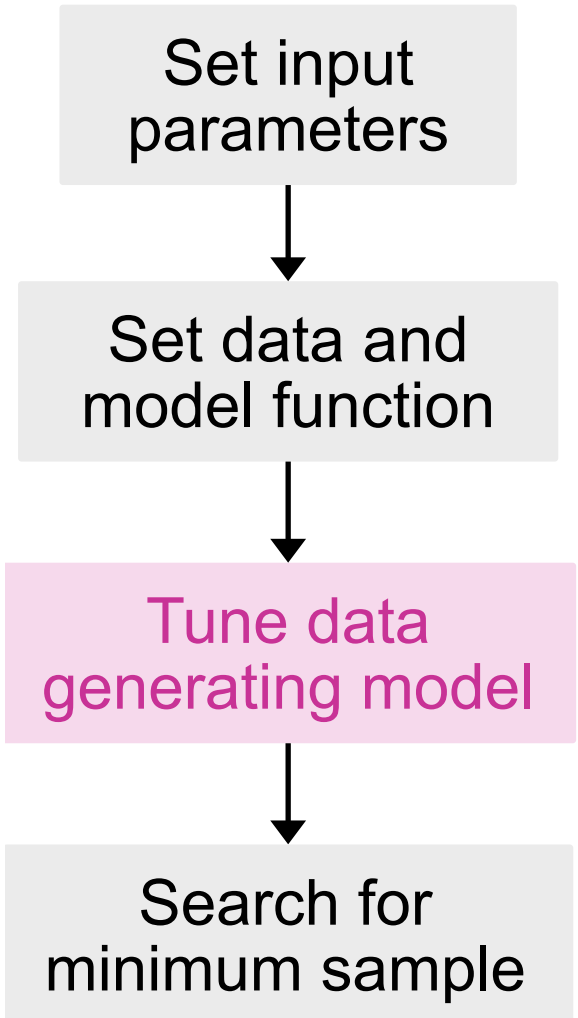
How does it work?



Based on their input, we set:

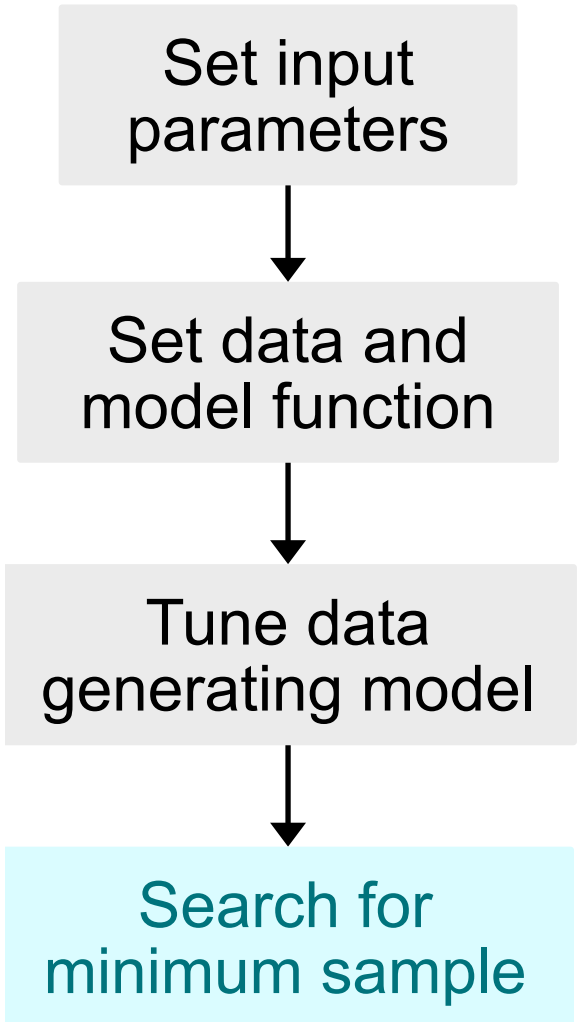
1. A data-generating function
2. A model function
3. A metric function

How does it work?



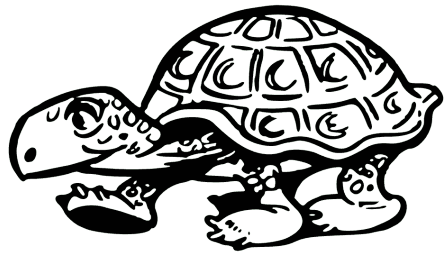
We tune the data generating model, so the large sample performance is P^*.

How does it work?



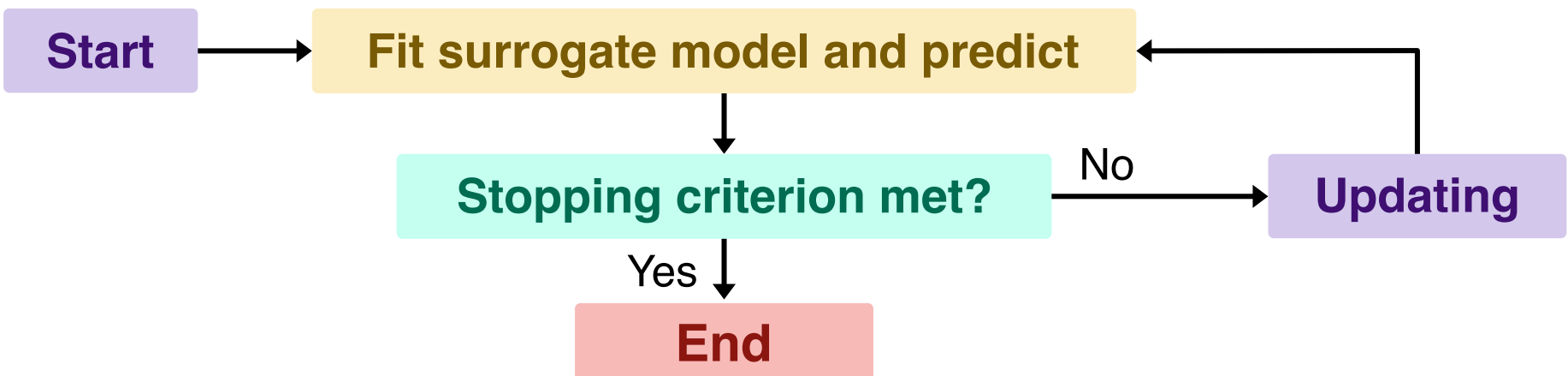
Performing the search

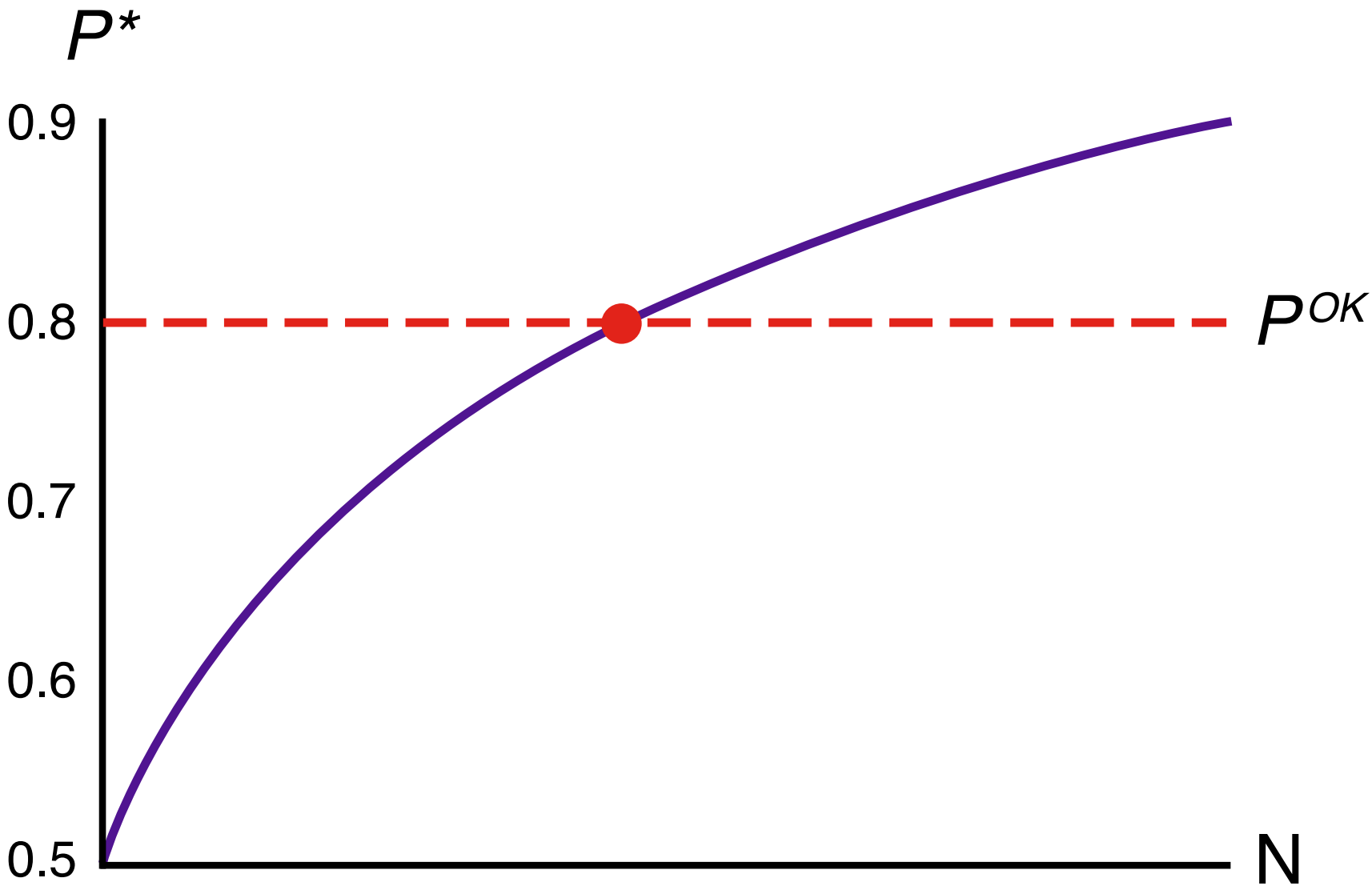
An exhaustive grid search would be too slow.

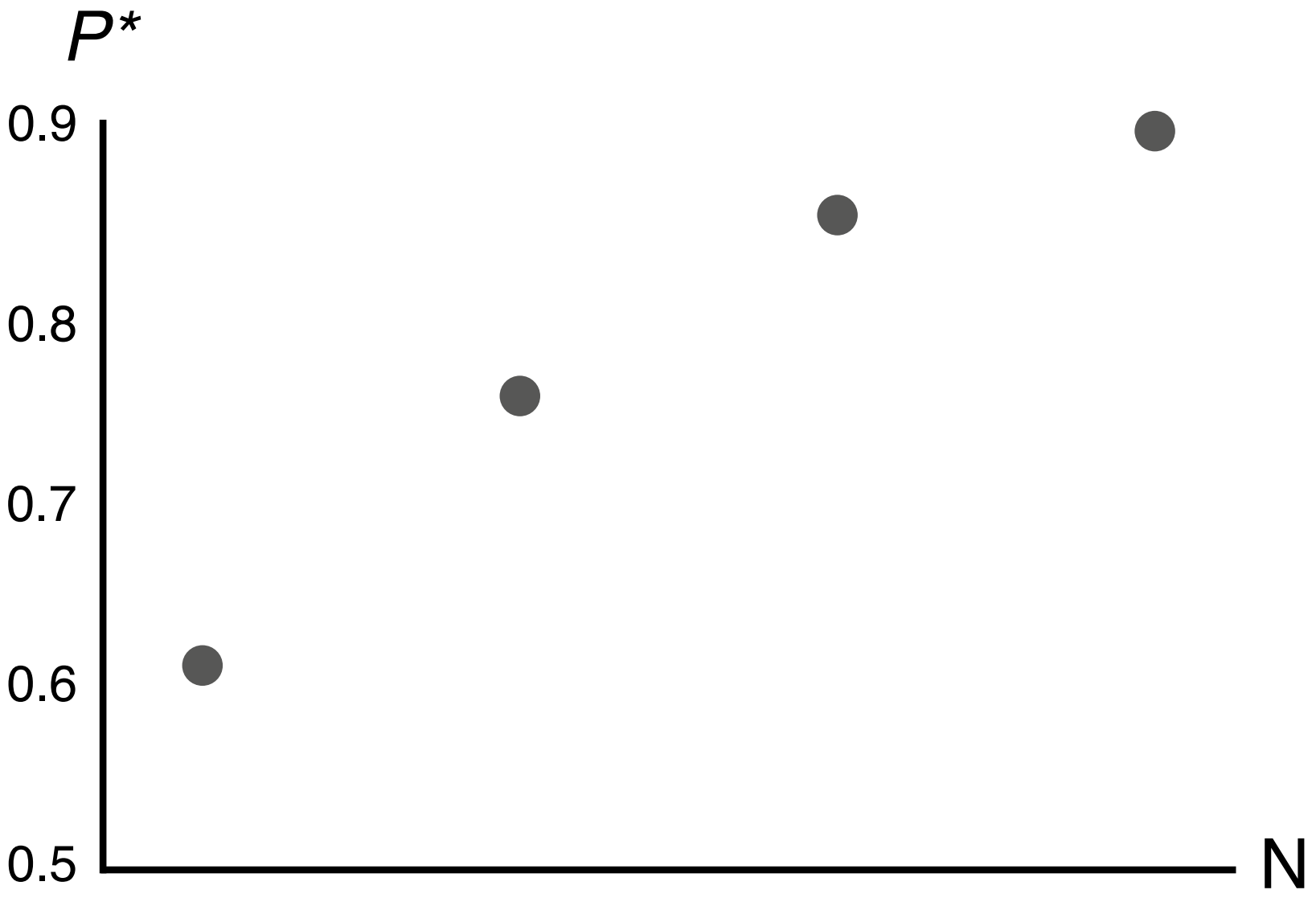


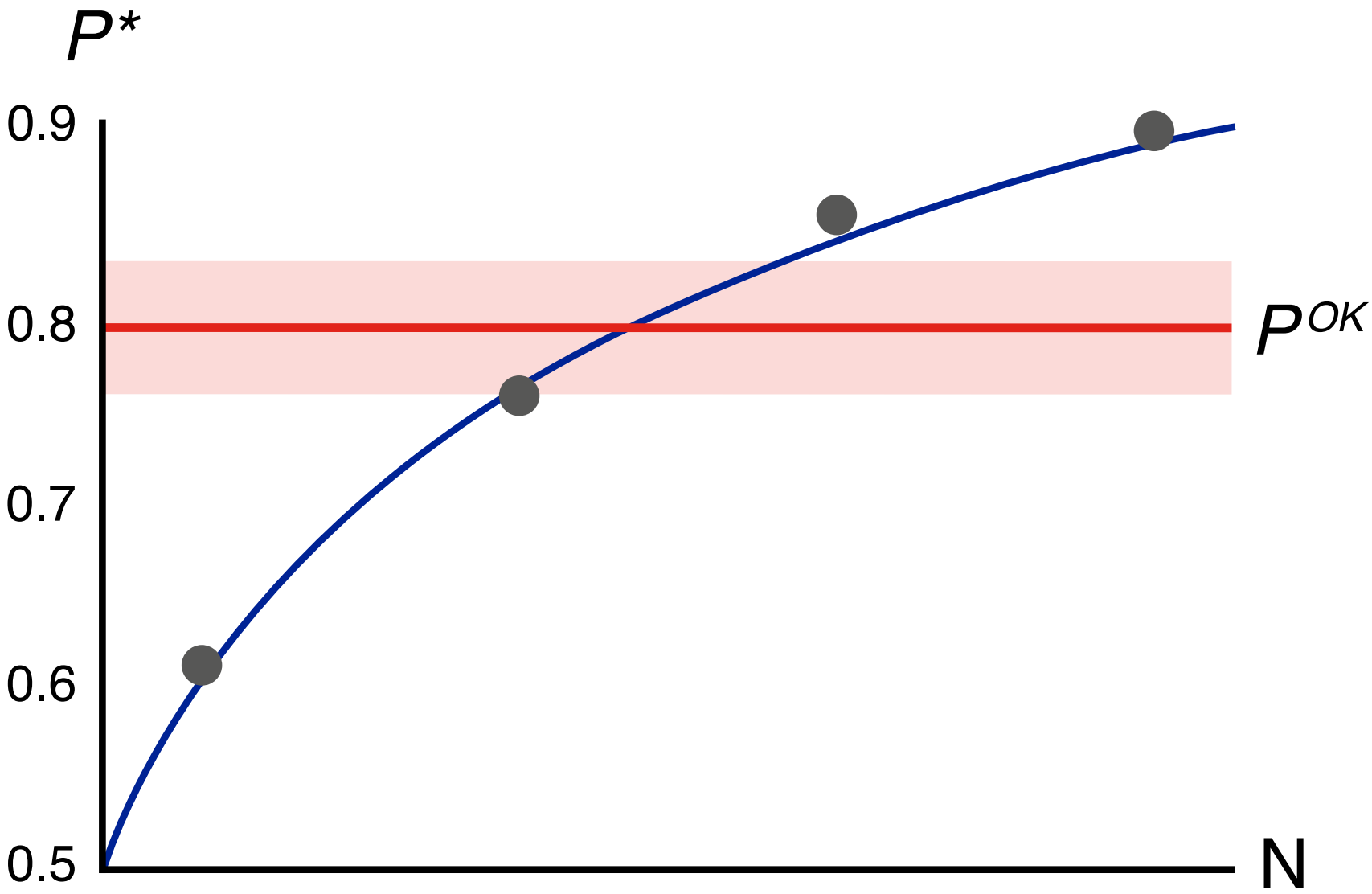
Surrogate modelling with mlpwr

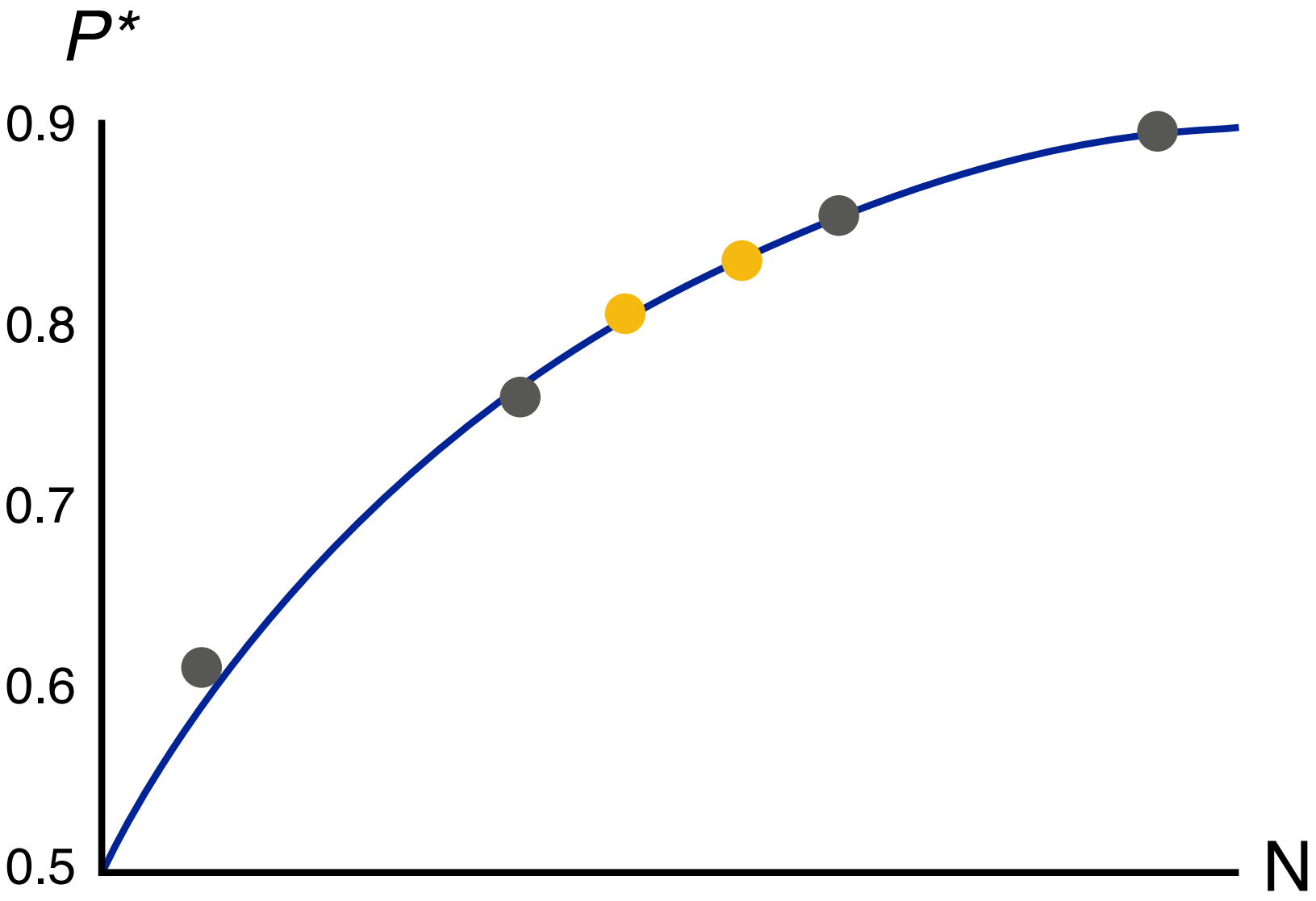
- Approximates the relationship between sample size and P^{OK} using Gaussian process regression.
- Also referred to as 'learning curve fitting'.[1,7]
- Uses the mlpwr R package by Zimmer and Debelak.[8]

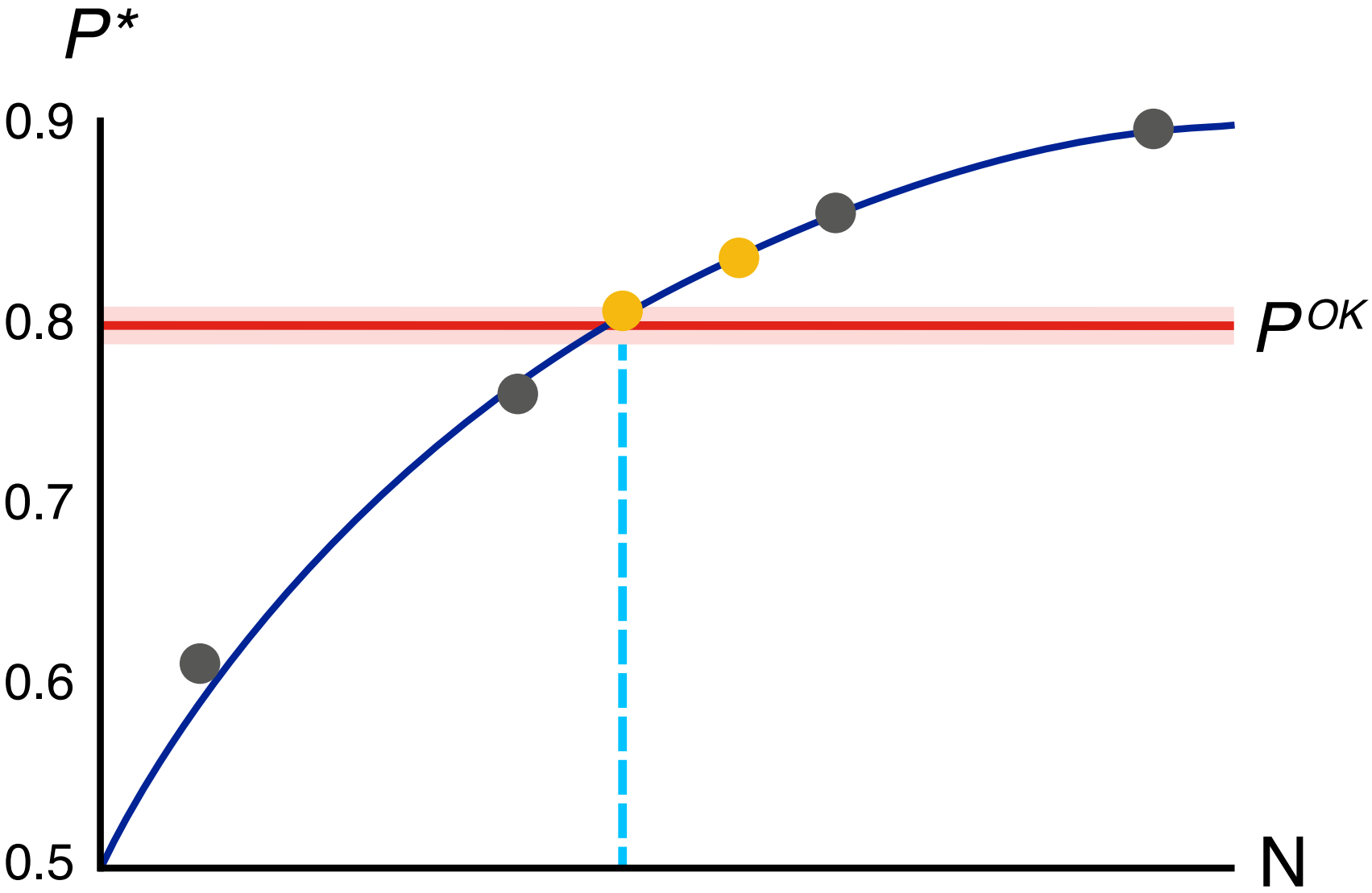



P*
0.9
0.8
0.7
0.6
0.5
P^OK
N

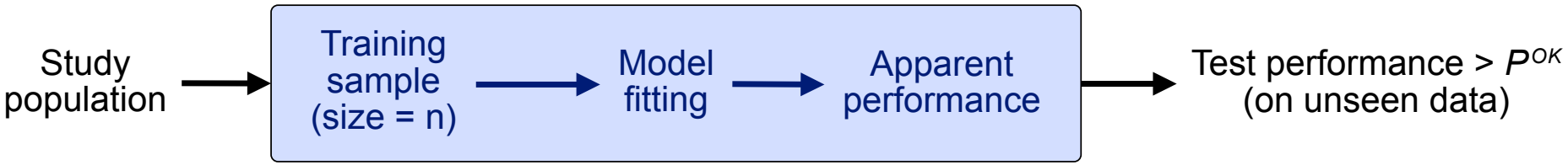

P*
0.9
0.8
0.7
0.6
0.5
N


P*
0.9
0.8
0.7
0.6
0.5
P OK
N


P*
0.9
0.8
0.7
0.6
0.5
N

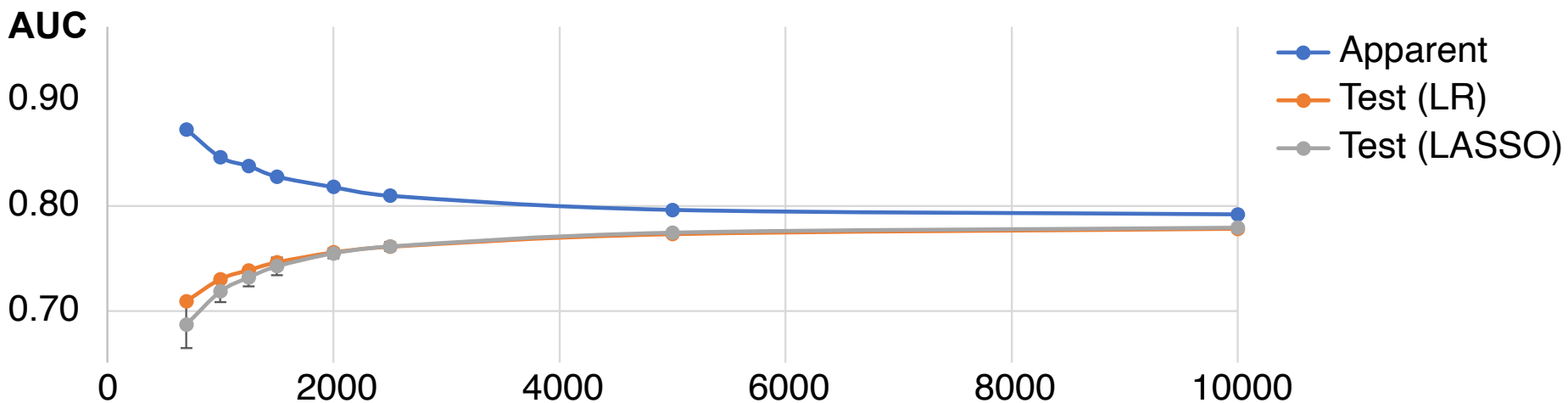

P*
0.9
0.8
0.7
0.6
0.5
P OK
N

What is the performance of a prediction model?



Apparent vs. test performance (or "actual" performance)

- Train/test performances are random variables of the drawn sample.
- Test performance is expected to be worse than apparent; but difference reduced with higher n.
- A prediction model is as good as its test performance.



How do we assess performance?

We identify the minimum sample that meets three criteria:

1. Overall fit

Within 0.1 of the achievable large sample fit (e.g., R^2, Brier).

2. Discrimination

Within 0.1 of the achievable large sample discrimination (e.g., C-statistic, AUC).

3. Calibration slope

A calibration slope of 0.9 to 1.1.

The choice of metrics and thresholds are user-configurable.

Two approaches to estimating minimum samples

	pmsims	pmsampsize
Approach	Simulate absolute test performance	Analytical closenesss of train-test; prevent overfitting
Target	Sample ensuring test performance P^{OK} with 80% probability.	Sample ensuring apparent and test performances are sufficiently close.
How	Tune data generator, use mlpwr to search for minimum sample meeting criteria.	Targets small train-test difference in R^2; or uniform shrinkage above given threshold (e.g., 0.9).
Calibration	Calibration slope criterion is similar to uniform shrinkage criterion. Slope is defined as minimizing the error between y^{test} and $\alpha + slope \times \hat{y}^{test}$.	Uniform shrinkage: GLM models where estimates depend on a linear predictor, $x^T\hat{\beta}$, with $\hat{\beta} - OLS$ estimates from the training sample. $s \cdot x^T\hat{\beta}$ may $\downarrow$overfitting and $\uparrow$performnce on unseen cases.

What are the distinctive features of these approaches?

	pmsims	pmsampsize
Flexibility	Any model/data	Closed form only for some models
Complex designs	Specified by user	Not possible
Speed	Slower*	Fast
Of 100 training samples of size n*	Test performance above P^{OK} in 80%	Mean test performance = P^{OK}
Large test performance variability	Adjusted for (using 0.2 quantile)	Not adjusted for

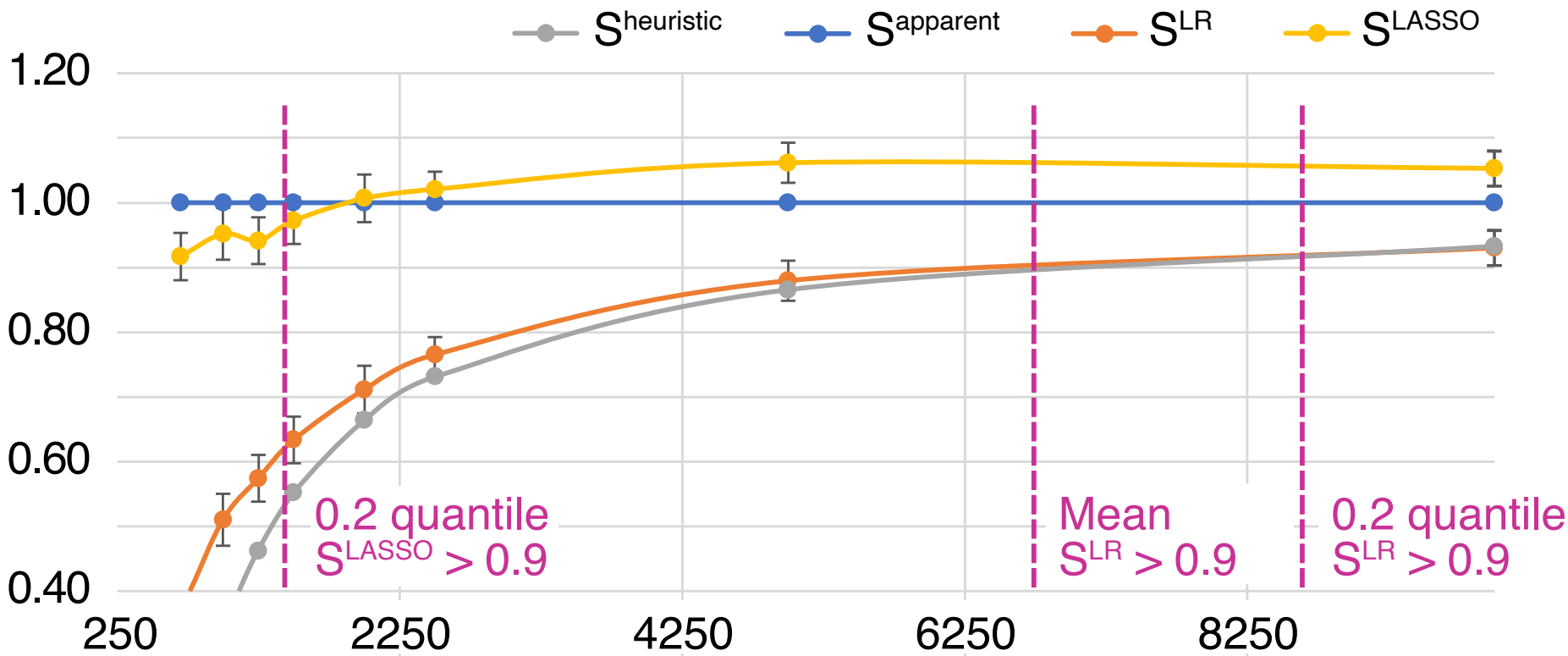
Compared to pmsampsize, our approach may suggest:

Smaller N for machine learning models:

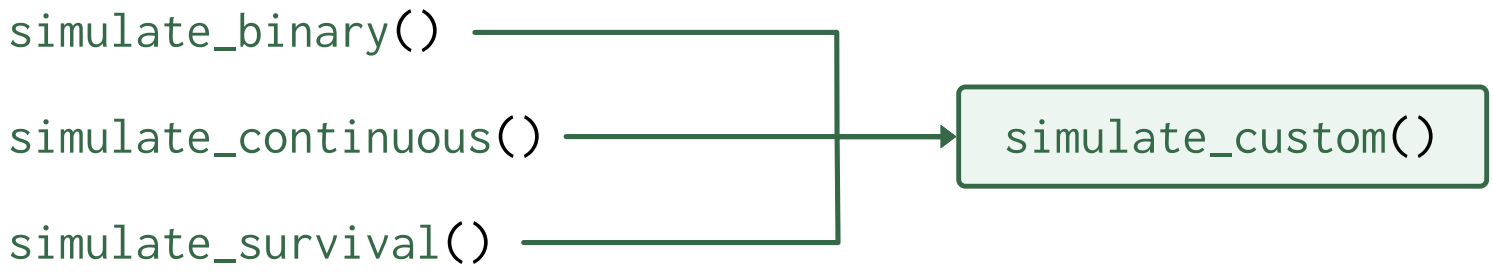
- Tend to overfit but may still achieve sufficient test performance

Larger N for noisy data and models with high variance:

- 0.2 quantile test performance < mean performance.



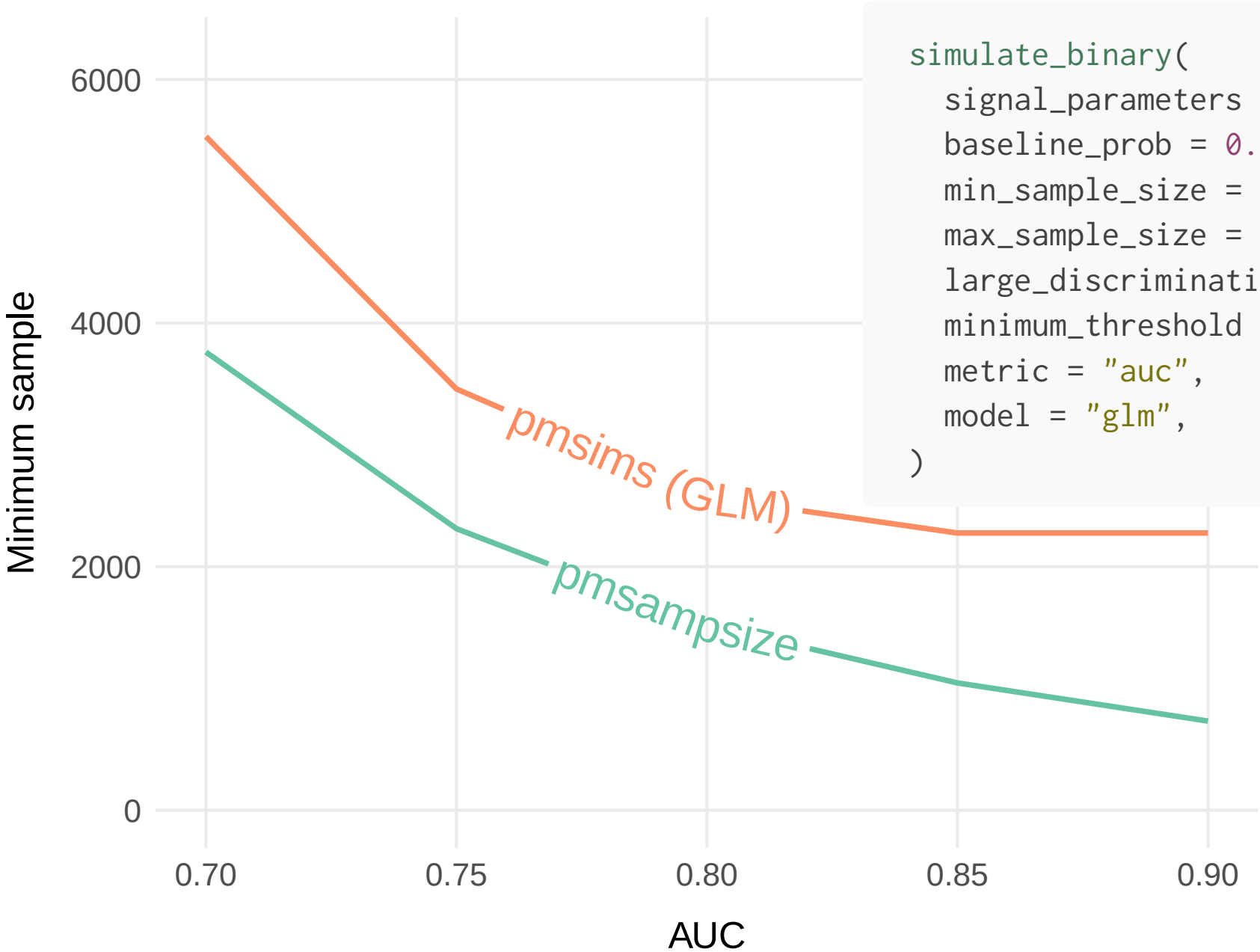
The user interface




```
simulate_continuous <-
  function(
    signal_parameters = 30,
    noise_parameters = 0,
    min_sample_size = 300,
    max_sample_size = 10000,
    large_discrimination = 0.7,
    minimum_threshold = 0.1,
    model = "lm",
    metric = "r2",
    ...
  )
```

```
simulate_binary <-
  function(
    signal_parameters = 30,
    noise_parameters = 0,
    baseline_prob = 0.1,
    min_sample_size = 300,
    max_sample_size = 10000,
    large_discrimination = 0.8,
    minimum_threshold = 0.1,
    metric = "auc",
    model = "glm",
    ...
  )
```

Example: Binary outcome, logistic regression




```
simulate_binary(
  signal_parameters = 20,
  baseline_prob = 0.1,
  min_sample_size = 500,
  max_sample_size = 15000,
  large_discrimination = ...,
  minimum_threshold = 0.1,
  metric = "auc",
  model = "glm",
)
```

Example: Custom model function

What if a model hasn't been implemented? e.g., XGBoost

```
model_function <- function(d) {
  dmat <- xgboost::xgb.DMatrix(
    as.matrix(d[, -1]),
    label = d[, 1]
  )
  param <- list(
    objective = "binary:logistic",
    booster = "gblinear",
    alpha = 0.0001,
    lambda = 1
  )
  xgboost::xgb.train(
    param,
    dmat,
    nrounds = 2
  )
}
```

```
metric_function <- function(data,
                            fit,
                            model) {
  dmat <- xgboost::xgb.DMatrix(
    as.matrix(data[, -1]),
    label = data[, 1]
  )
  y_hat <- predict(fit, dmat)
  pROC::auc(data[, 1], y_hat)[1]
}
```

```
simulate_custom(
  data_function = data_function,
  model_function = model_function,
  metric_function = metric_function,
  ...
)
```

Development status

✓ Framework

R package

✓ Data generators

Linear, binary, survival

✓ Model generators

Linear, logistic, Cox, LASSO

What's next?

1. Machine learning

Defaults for common algorithms (e.g., random forest).

2. Longitudinal and clustered data

Data generators and models (e.g., landmarking, joint).

3. More sophisticated data generators

Synthesise common data types (e.g., genetic); user control.

4. Performance

Parallelisation, caching of common tuning parameters.

3. More sophisticated data generators

Synthesise common data types (e.g., genetic); user control.

4. Performance

Parallelisation, caching of common tuning parameters.

Follow fediscience.org/@ewan for updates

Enter email at tinyurl.com/is-pmsims-ready-yet
to receive one email when its ready

Come and talk to us

Thank you for listening.

github.com/ewancarr/pmsims-iscb

ewan.carr@kcl.ac.uk

diana.shamsutdinova@kcl.ac.uk

fediscience.org/@ewan

References I

[1] Rosa L. Figueroa et al. “Predicting Sample Size Required for Classification Performance”. In: *BMC Medical Informatics and Decision Making* 12.1 (Feb. 2012), p. 8. ISSN: 1472-6947. DOI: 10.1186/1472-6947-12-8. (Visited on 08/17/2023).

[2] Maarten van Smeden et al. “No Rationale for 1 Variable per 10 Events Criterion for Binary Logistic Regression Analysis”. In: *BMC Medical Research Methodology* 16.1 (Nov. 2016), p. 163. ISSN: 1471-2288. DOI: 10.1186/s12874-016-0267-3. (Visited on 07/31/2023).

[3] Laure Wynants et al. “Prediction Models for Diagnosis and Prognosis of Covid-19: Systematic Review and Critical Appraisal”. In: *BMJ* 369 (Apr. 2020), p. m1328. ISSN: 1756-1833. DOI: 10.1136/bmj.m1328. (Visited on 08/16/2023).

[4] Constanza L. Andaur Navarro et al. “Risk of Bias in Studies on Prediction Models Developed Using Supervised Machine Learning Techniques: Systematic Review”. In: *BMJ* 375 (Oct. 2021), n2281. ISSN: 1756-1833. DOI: 10.1136/bmj.n2281. (Visited on 07/31/2023).

References II

[5] Richard D. Riley et al. “Penalization and Shrinkage Methods Produced Unreliable Clinical Prediction Models Especially When Sample Size Was Small”. In: *Journal of Clinical Epidemiology* 132 (Apr. 2021), pp. 88–96. ISSN: 1878-5921. DOI: 10.1016/j.jclinepi.2020.12.005.

[6] Alan J. Meehan et al. “Clinical Prediction Models in Psychiatry: A Systematic Review of Two Decades of Progress and Challenges”. In: *Molecular Psychiatry* 27.6 (June 2022), pp. 2700–2708. ISSN: 1476-5578. DOI: 10.1038/s41380-022-01528-4. (Visited on 07/31/2023).

[7] Alimu Dayimu et al. *Sample Size Determination via Learning-Type Curves*. Mar. 2023. arXiv: 2303.09575 [stat]. (Visited on 08/16/2023).

[8] Felix Zimmer and Rudolf Debelak. “Simulation-Based Design Optimization for Statistical Power: Utilizing Machine Learning”. In: *Psychological Methods* (in press). ISSN: 1082-989X.