

Developing and deploying AI in the ICU, methodological challenges

Giovanni Cinà

Amsterdam UMC, University of Amsterdam, Pacmed



g.cina@uva.nl, g.cina@amsterdamumc.nl



@CinaGiovanni

My trajectory



THE LONDON SCHOOL
OF ECONOMICS AND
POLITICAL SCIENCE ■

UNIVERSITY OF AMSTERDAM



pacmed



UNIVERSITY OF AMSTERDAM

the inexorable advance of time

AI in healthcare: a long and bumpy road

MIT
Technology
Review

Featured Topics Newsletters Events Podcasts

Sign in Subscribe

ARTIFICIAL INTELLIGENCE

Google's medical AI was super accurate in a lab. Real life was a different story.

HEALTH TECH

Epic's widely used sepsis prediction model falls short among Michigan Medicine patients

The main question we want to answer today

What research is required to make sure than an AI application is going to improve care?



The main question we want to answer today, rephrased

When we implement medical AI

1. What are the methodological challenges we need to resolve?
2. What research can we do to address those issues?





Agenda

1. An example: a tool to aid discharge decisions in the ICU
2. Engage with AI -> Explainable AI
3. Data shift -> Out-of-Distribution detection
4. Treatment effect estimation -> Causal Inference

Agenda

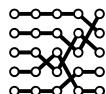
1. **A tool to aid discharge decisions in the ICU**
2. Engage with AI -> Explainable AI
3. Data shift -> Out-of-Distribution detection
4. Treatment effect estimation -> Causal Inference

The intensive care serves as an ideal starting point for a data driven hospital. The discharge decision as a use case.

Intensive Care



Large amount of high quality data



Complex decisions depending on a large variety of factors



Capacity can form a bottleneck in terms of staffing, costs and operations

Pacmed's solution



Improve capacity and prevent readmissions



Build an AI that helps with choosing the optimal moment for discharge



- Reduce readmission rate
- Reduce mortality rate
- Reduce length of stay

Pacmed's approach: strong collaboration with the medical field



Co-development of the ICU tool in partnership with the Amsterdam University medical Center



Research on the product with various academic partnership to ensure methodological rigor



Collaborating openly with regulators to develop best practices regarding responsible deployment of machine learning in healthcare

Pacmed Critical predicts readmission and unforeseen mortality within 7 days for all patients eligible for discharge

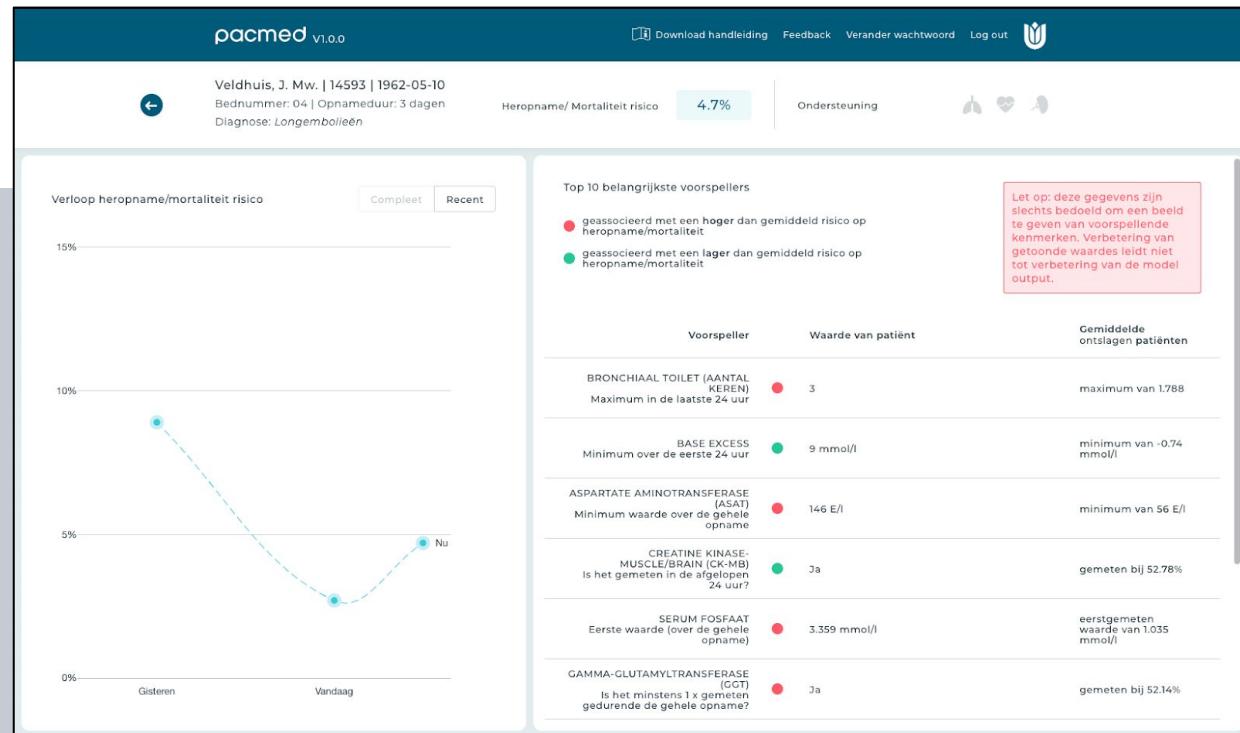
The screenshot shows a web-based application titled "pacmed v1.0.0". The main header includes links for "Download handleiding", "Feedback", "Verander wachtwoord", "Log out", and a logo. Below the header, a button labeled "Toon zonder ondersteuning" is visible. The main content area is titled "Afdelingsmonitor" and displays a table of patient data. The columns are labeled "BEDNR.", "PATIËNTGEGEVENS", "OPNAME DIAGNOSE", "HEROPNAME/MORTALITEIT RISICO", and "ONDERSTEUNING". The table lists 10 patients with their corresponding details and predicted risks:

BEDNR.	PATIËNTGEGEVENS	OPNAME DIAGNOSE	HEROPNAME/MORTALITEIT RISICO	ONDERSTEUNING
01	Janssen, J. Dhr. 14250 1954-11-01	Post-operatief CABG	1.0%	lung heart brain
02	Brandts, M. Mw. 18282 1954-11-11	Coma/verandering bewustzijnsniveau (non-operatief neuro)	1.8%	lung heart brain
03	Estevez, E. Mw. 15045 1940-07-15	Respiratoir - medisch anders	2.5%	lung heart brain
04	Veldhuis, J. Mw. 14593 1962-05-10	Longembolieën	4.7%	lung heart brain
05	Berendse, F. Dhr. 17359 1969-06-12	Cardiovasculair - medisch anders	1.6%	lung heart brain
06	Huygens, S. Dhr. 15982 1968-09-29	Bacteriële pneumonie	6.1%	lung heart brain
07	Tully, T. Dhr. 15066 1939-04-01	Acuut nierfalen	-	lung heart brain
08	Jungens, M. Dhr. 14290 1994-08-15	Bacteriële pneumonie	8.2%	lung heart brain
09	Meester, M. Dhr. 14688 1953-12-16	Congestief hartfalen	-	lung heart brain

A "Meer patienten" button is located at the bottom of the table.

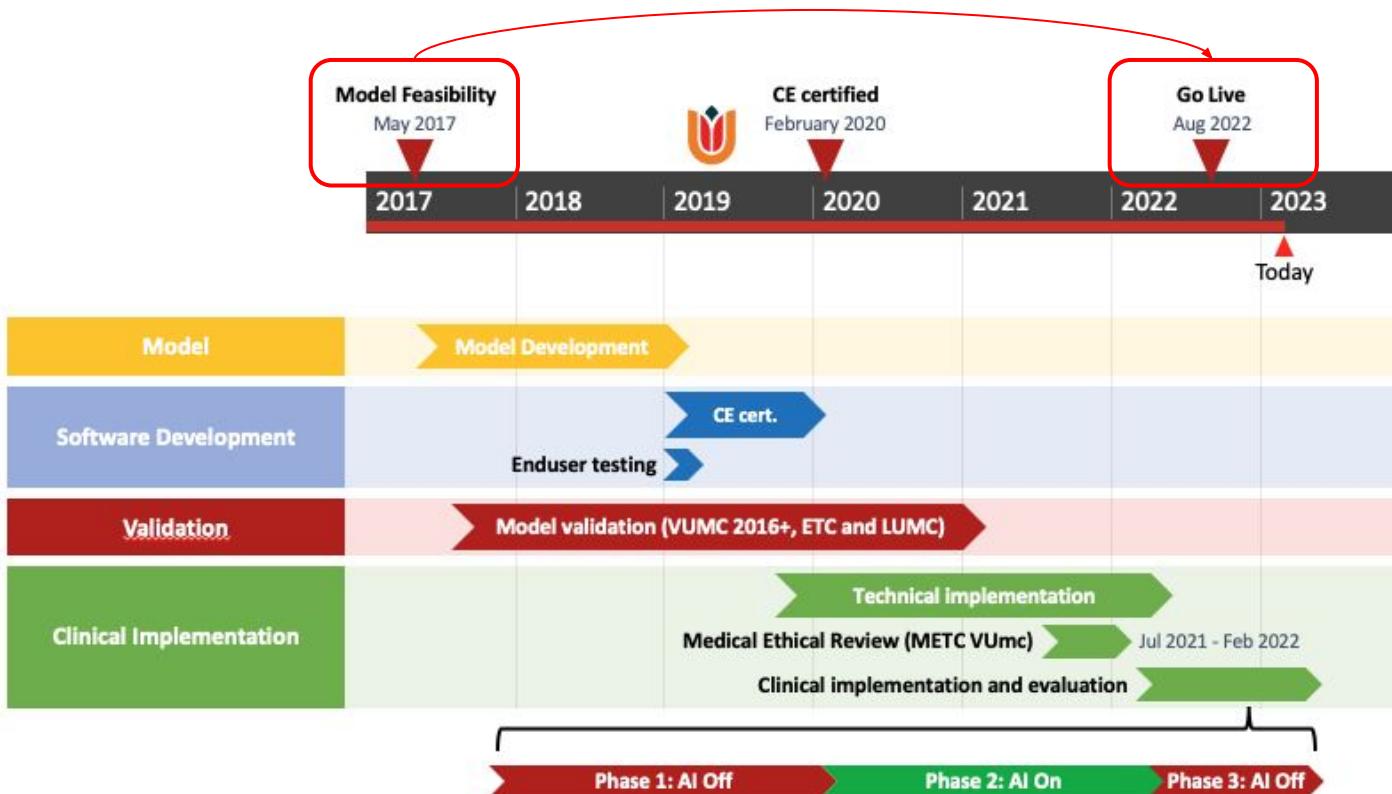
- Overview of all patients on the ICU, with the predicted risk
- Basic information about the patients is displayed
- **Predicted readmission/mortality risk within 7 days based on machine learning model**

It shows the progression of the risk score for every patient, and the features supporting the prediction



- Predicted risk is shown over time
- The most important variables contributing to the individual risk are displayed
- Simple design, tested and validated with >25 intensivists from 3 hospitals
- This interface is going to change soon

After 5 years of work, the tool is now live



Bringing a tool to the bedside is a long journey... and it takes a big team



Agenda

1. A tool to aid discharge decisions in the ICU
2. **Engage with AI -> Explainable AI**
3. Data shift -> Out-of-Distribution detection
4. Treatment effect estimation -> Causal Inference

Explainable AI: (some of) the problems

1. We do not tailor our AI interfaces to the users, or not enough
2. Clinicians must be able to engage with the AI's reasoning, to decide when they agree and when they disagree
3. Explanations should be reliable (no confirmation bias)

Explainable AI I: survey on clinicians' wishes on explainable AI

We develop several techniques to 'explain' what the AI does to the users.

...but have we asked the users what they want?

We put together a survey to gather clinicians' preferences on XAI, it can be found [here](#). For clinicians only!



Welcome to the survey on
Explainable Machine Learning in Healthcare.

This is a survey organised by the Amsterdam Business School, University of Amsterdam, the Netherlands.

The survey is estimated to take around 15-30 minutes. Thank you for taking the time to participate.



0% 100%

A horizontal progress bar at the bottom of the slide, showing a thin grey bar with '0%' at the left end and '100%' at the right end, indicating the survey's completion status.

Explainable AI II: linking to clinicians' known concepts

We develop several techniques to 'explain', but with a Computer Science mindset.

We need explanations linked to concepts clinicians use every day to discuss patients.

Example: give corpus-based explanations based on medical archetypes.



The ABCDE Approach
To
Deteriorating
Or
Critically Ill Patients

Explainable AI III: addressing the problem of confirmation bias

Suppose you get an image and an explanation



dowitcher

red-backed_sandpiper

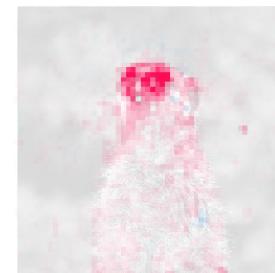


Is this convincing?

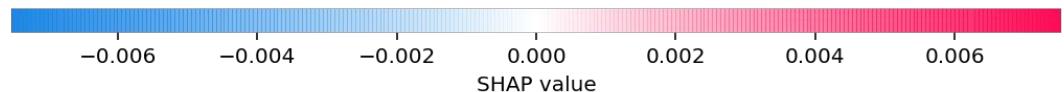
Why?



meerkat



mongoose



What can go wrong: the way in which explanations are used and understood

How do you know that the machine has a concept of 'head' that it is used to classify the meerkat?

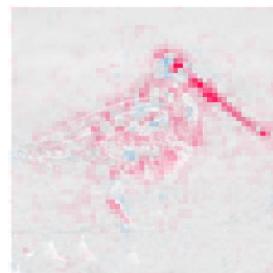
Or 'beak' to classify the dowitcher?



dowitcher



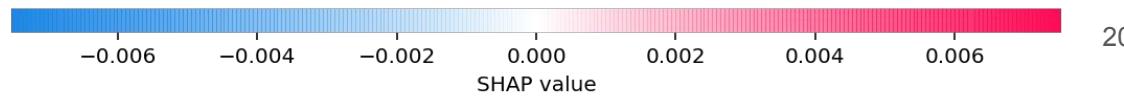
meerkat



red-backed_sandpiper



mongoose



What can go wrong: the way in which explanations are used and understood

The fact that the cloud of pixels highlighted is sensible to us does not mean that it is highlighted for the right reason.

Confirmation bias

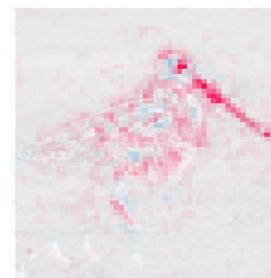
The tendency to believe explanations that confirm our belief/conviction.



dowitcher



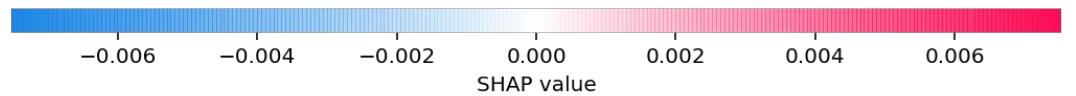
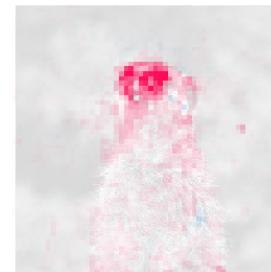
meerkat



red-backed_sandpiper



mongoose



Criticism

Line of argument:

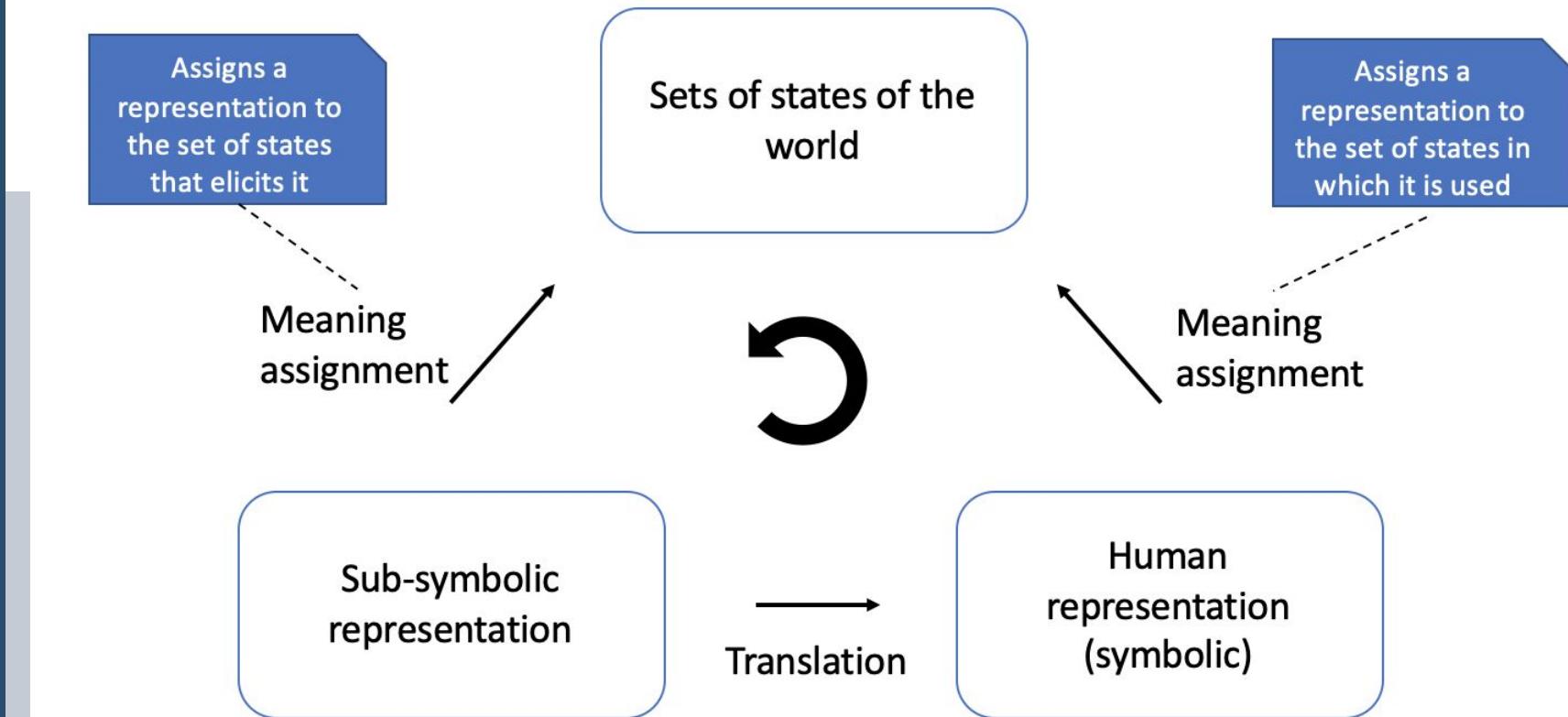
- 1) We have no idea whether the explanation of the machine means what we think it means
- 2) We risk to
 - a) project our belief onto the machine
 - b) accept an explanation that is ungrounded/misleading
- 3) Hence these post-hoc local explanations are not reliable
- 4) We should not use them in medical contexts, and should not be suggested in guidelines etc

The false hope of current approaches to explainable artificial intelligence in health care

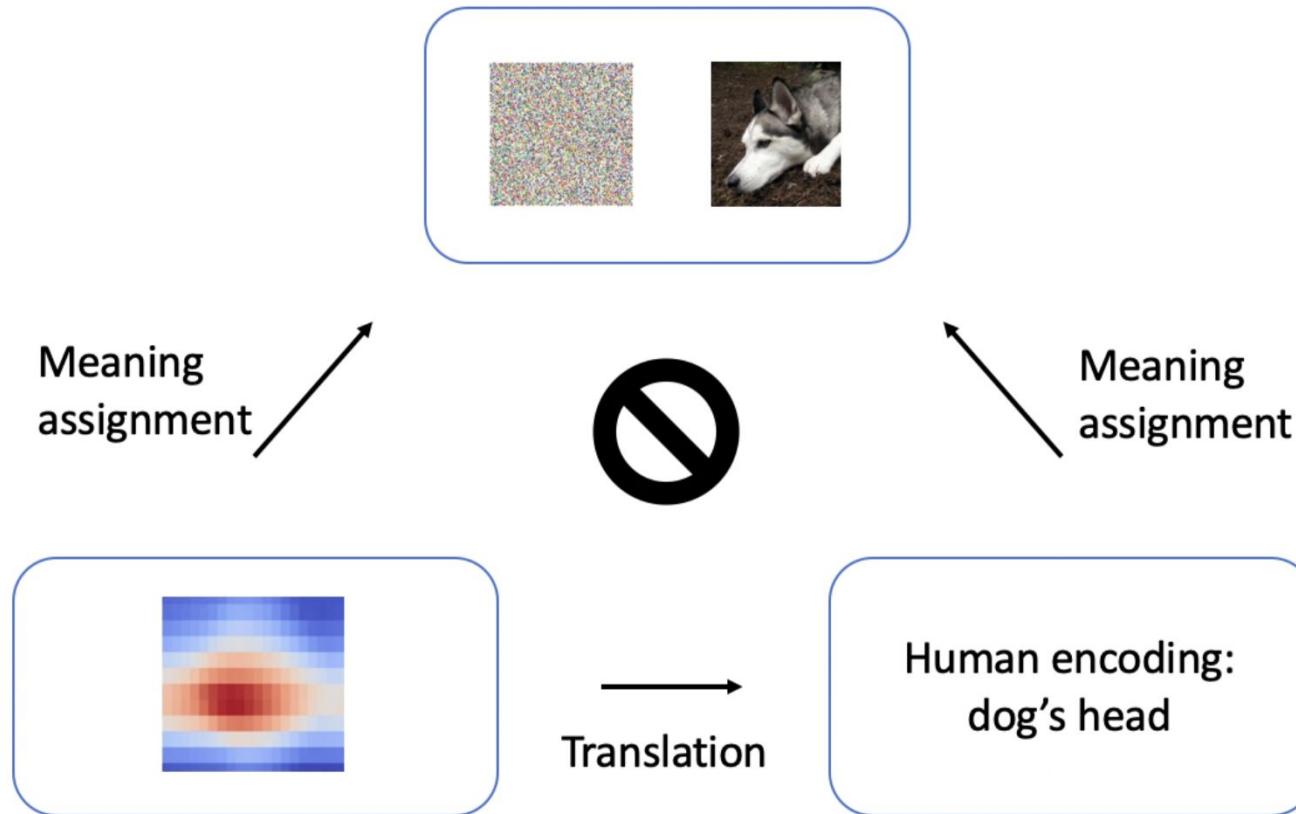
Core issue: semantic match between sub-symbolic and symbolic representations

Humans cannot attribute meaning to a sub-symbolic representation (e.g. a vector or a matrix of numbers) without matching it to a symbolic concept we use or know.

Semantic match is encoded by the commutation of this diagram



The failure of semantic match: example



A reflection on the meaning of features

There are

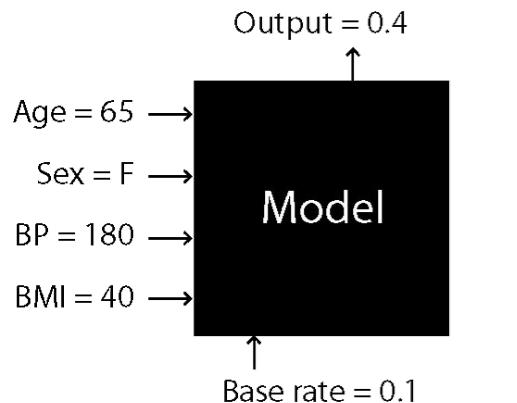
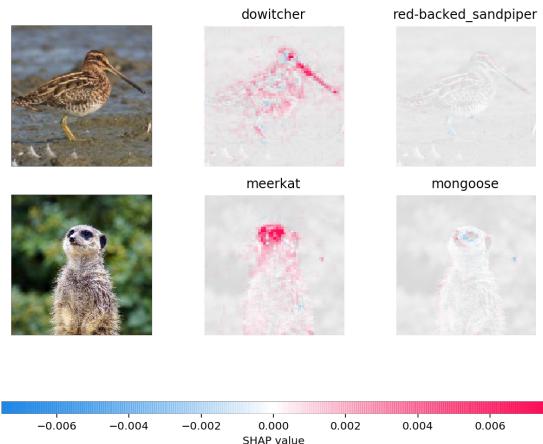
1. low-level features, i.e. the entries of your input vector
2. high-level features, i.e. representations of the problem the machine is using; e.g. entries of the latent space of NN

In general **we do not have access to the meaning of high-level features of a machine (black box).**

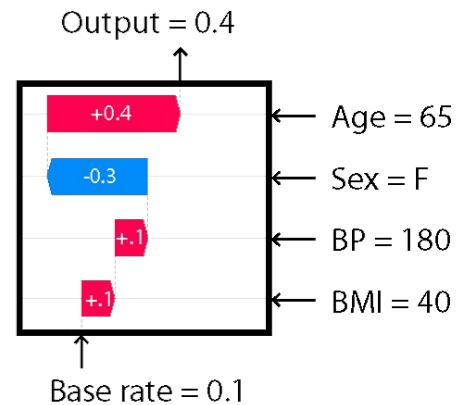
However for low-level features there is a distinction:

3. some data types (e.g. EHR) have low-level features with clear meaning
4. some data types (e.g. images) have low-level features without meaning

Compare these two explanations



Explanation



First conclusion

- 1) In data types where low-level features have meaning, we can use feature attribution at the level of single features because we have semantic match ‘out-of-the-box’

- 2) In all data types, explanations of high level features are unreliable... unless we find a way to access the internal representation of the machine

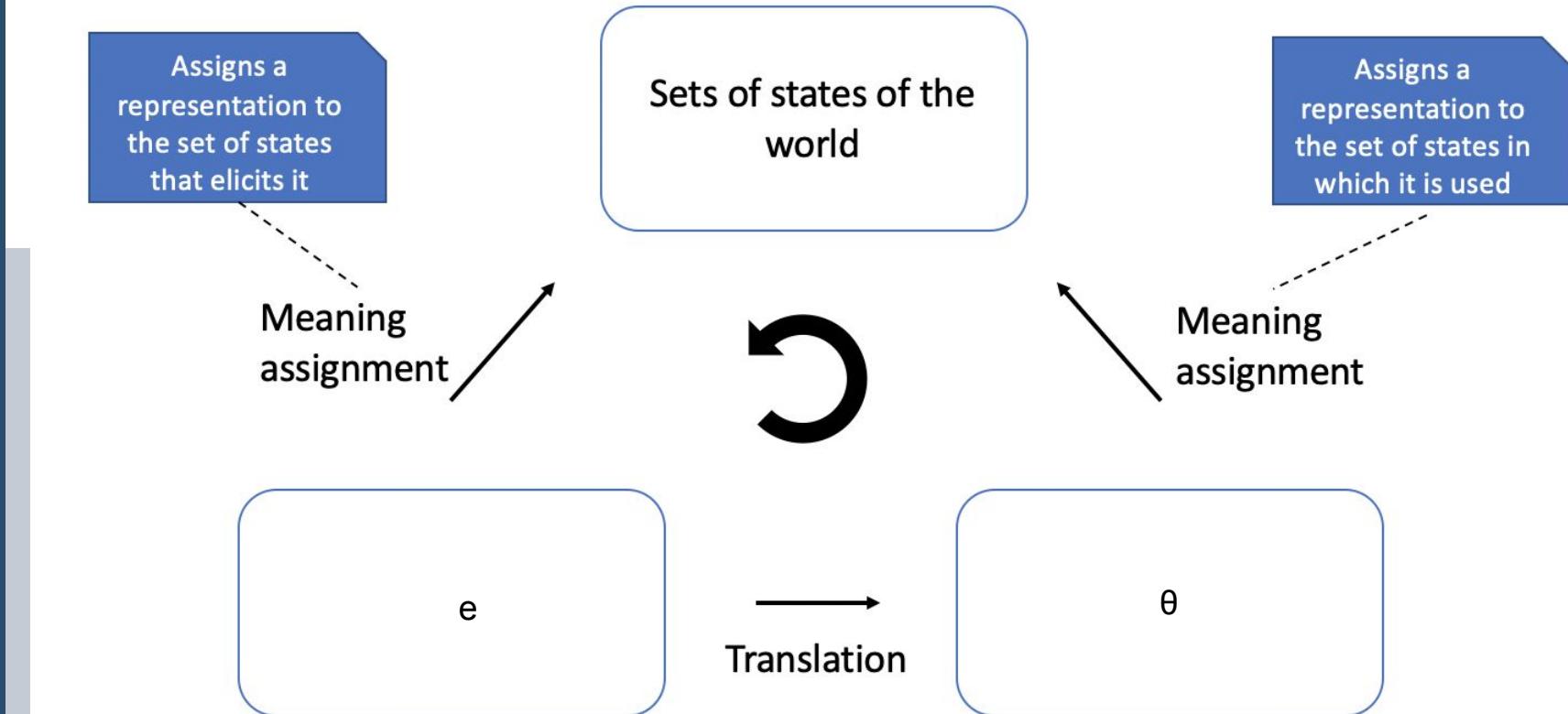
So... let's find a way to access the internal representation of the machine

Assume a ML model f has been trained on labeled data, and we are considering a **sample** (x, y) . Denote a local **feature attribution method** with M and say that $M(f, x) = e$ is the **explanation** for why model f gives prediction $f(x)$ on input x .

We formulate an **hypothesis θ** of what is highlighted by the explanation. We are interested in testing whether we have **semantic match between θ and e** .



Recap of semantic match diagram

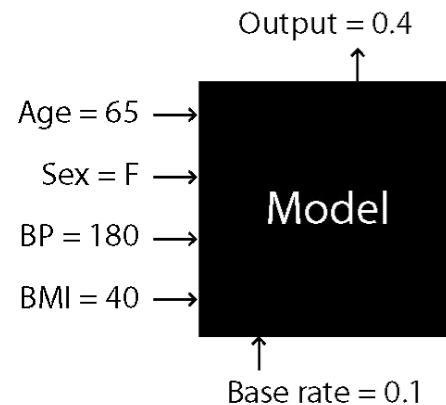


Agenda

1. A tool to aid discharge decisions in the ICU
2. Engage with AI -> Explainable AI
- 3. Data shift -> Out-of-Distribution detection**
4. Treatment effect estimation -> Causal Inference

The problem of Out Of Distribution (OOD) data

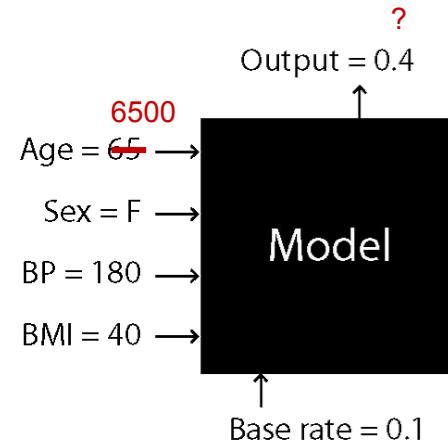
Suppose you have an AI software implemented in hospitals. At first the model receives data similar to training data.



The problem of Out Of Distribution (OOD) data

Suppose you have an AI software implemented in hospitals. At first the model receives data similar to training data.

Then for some reason the data arriving to the model changes remarkably. Now the software's output is not reliable.



The problem of Out Of Distribution (OOD) data

Suppose you have an AI software implemented in hospitals. At first the model receives data similar to training data.

Then for some reason the data arriving to the model changes remarkably. Now the software's output is not reliable.

Often the user doesn't realize!



The causes of data shift (aka covariate shift) in a medical context

1. The demographics of the population change
2. The treatment protocols change
3. There are bugs in the code
4. Systematic human errors in data input
5. Third party manipulation
6. ...



But...why is this a new problem?

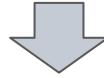
We have always had this problem, and we solved it with outlier detection and statistical tests to detect distribution shift (e.g. SPM).

What is new is **high-dimensional data**. Many of those techniques do not scale to high dimensionality (e.g. K-S test).



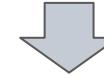
Detecting OOD periodically vs in real time

Monitoring data shift
periodically



Errors accumulate
before change is
detected

Monitoring in real time



Less certainty but
errors can be prevented

We want a reliable way to flag OOD patients in real time. What does the literature say?

Can we use a model's uncertainty to flag OOD samples?

- No conclusive information in literature
- Lack of tests on medical data
- Very little tests on structured data (like EHRs)

Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift

Yaniv Ovadia*
Google Research
yovadia@google.com

Zachary Nado
Google Research
znado@google.com

Joshua V. Dillon
Google Research
jvdillon@google.com

Emily Fertig*†
Google Research
emilyaf@google.com

D Sculley
Google Research
dsculley@google.com

Balaji Lakshminarayanan‡
DeepMind
balajiln@google.com

Jie Ren†
Google Research
jjren@google.com

Sebastian Nowozin
Google Research
nowozin@google.com

Jasper Snoek‡
Google Research
jsnoek@google.com

We benchmarked several methods to see if they work in practice

We tested on public datasets

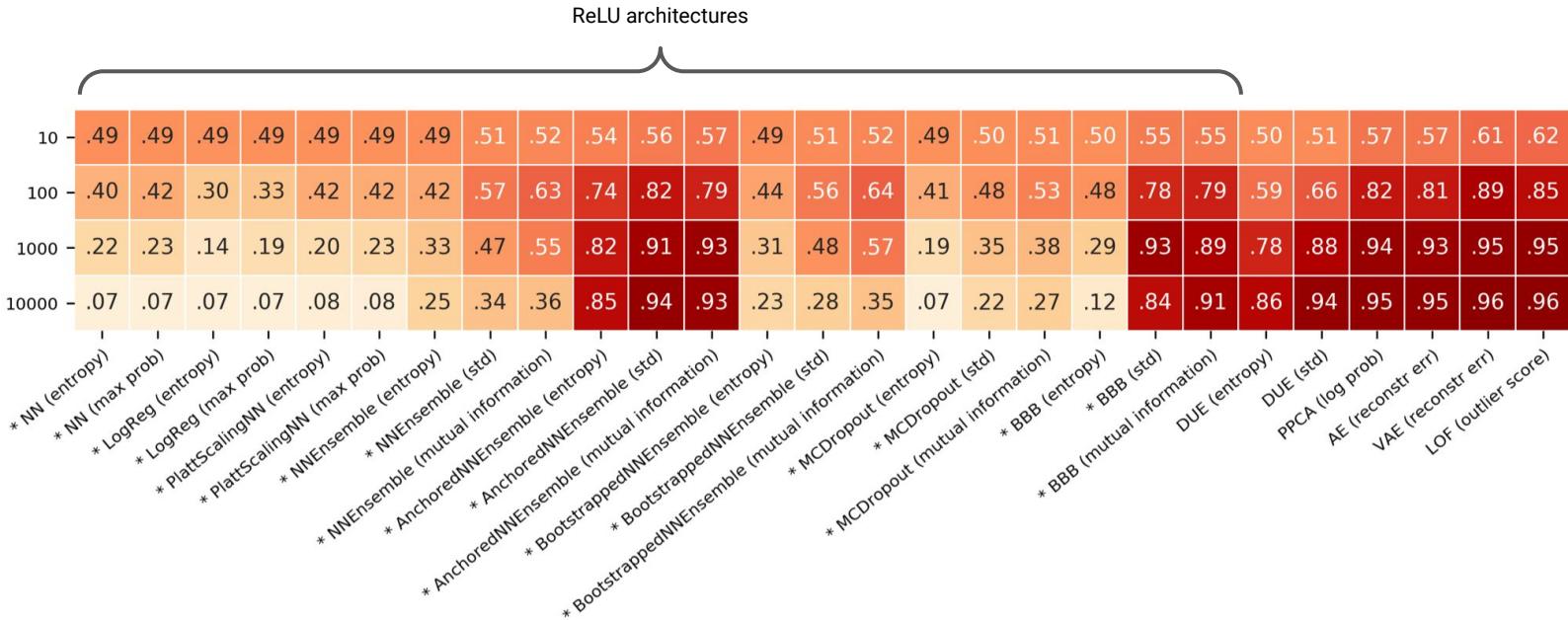
- MIMIC III
- eICU

We tested ~21 (27) combinations of models and uncertainty metrics

Experiments simulating different failure modes:

- **Perturbation:** Simulate data corruption by scaling a single feature
- **OOD groups:** Remove certain patients from training set to simulate shift in demographics / new conditions
- **Domain adaptation:** Use MIMIC-III data set as a new group of patients for a model trained on eICU, and vice versa

Perturbation experiment: scaling a single feature

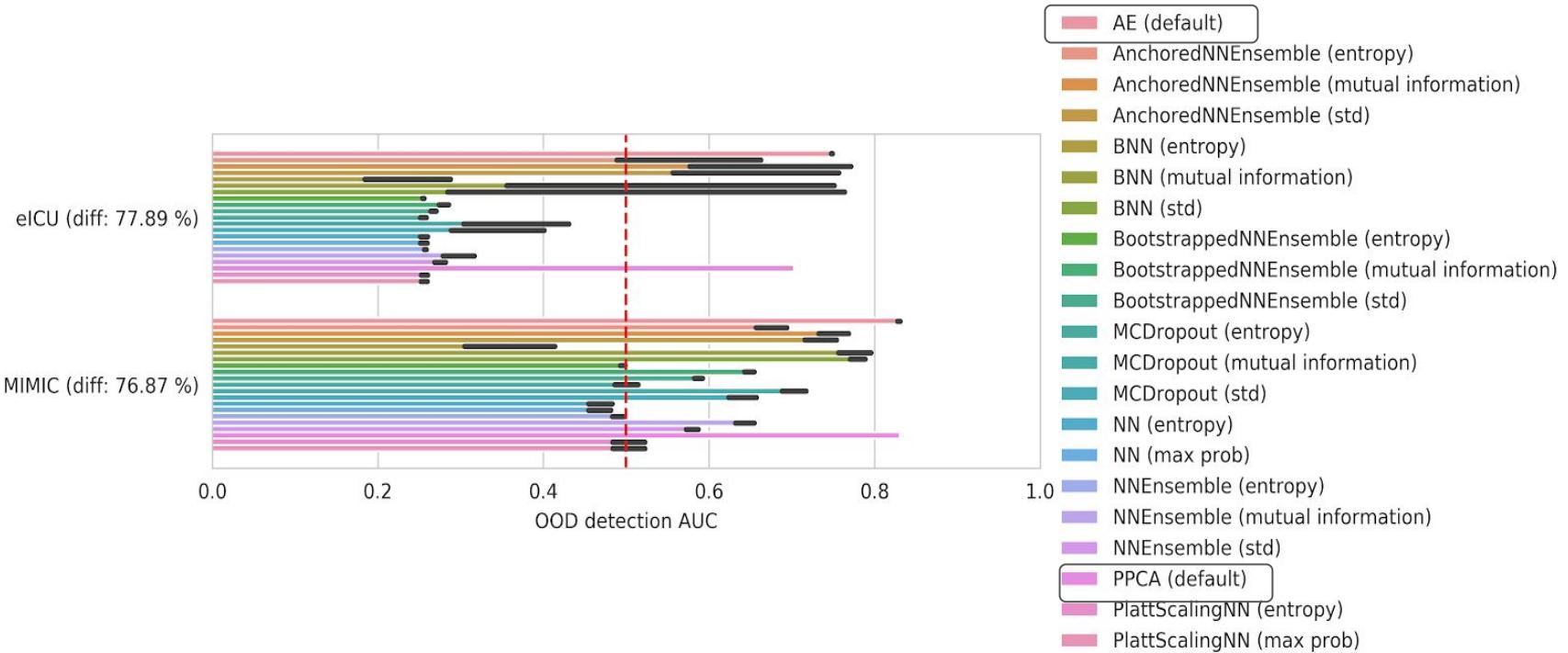


AUC-ROC of OOD detection for ReLU architectures mostly goes down if we scale a feature with larger and larger values

OOD Groups experiment

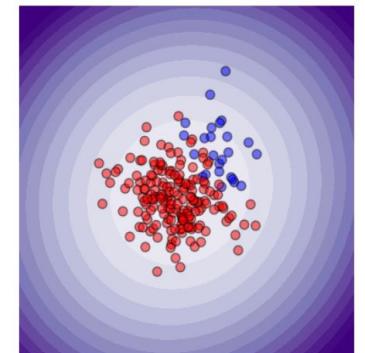
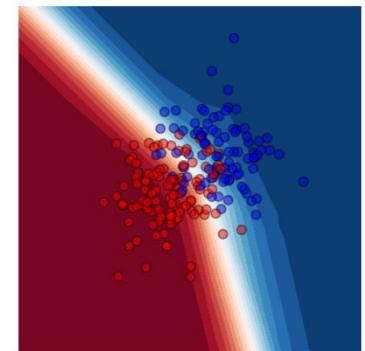
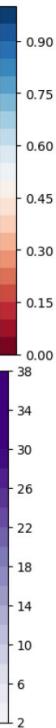
	OOD detection AUC MIMIC																							
	AE (default)	AnchoredNNEnsemble (entropy)	AnchoredNNEnsemble (mutual information)	AnchoredNNEnsemble (std)	BNN (entropy)	BNN (mutual information)	BNN (std)	BootstrappedNNEnsemble (entropy)	BootstrappedNNEnsemble (mutual information)	BootstrappedNNEnsemble (std)	MCDropout (entropy)	MCDropout (mutual information)	MCDropout (std)	NN (entropy)	NN (max prob)	NNEnsemble (entropy)	NNEnsemble (mutual information)	NNEnsemble (std)	PPCA (default)	PlattScalingNN (entropy)	PlattScalingNN (max prob)			
Acute and unspecified renal failure (size: 20.22 %, diff: 57.14 %)	.54	.62	.61	.61	.61	.38	.44	.62	.59	.61	.62	.54	.60	.62	.62	.62	.59	.61	.54	.62	.62			
Elective admissions (size: 13.43 %, diff: 64.46 %)	.54	.34	.36	.35	.49	.57	.56	.32	.39	.35	.32	.46	.36	.32	.32	.38	.35	.54	.32	.32				
Emergency/ Urgent admissions (size: 86.57 %, diff: 59.35 %)	.67	.54	.56	.56	.52	.59	.56	.52	.48	.50	.52	.53	.54	.51	.51	.42	.46	.62	.52	.52				
Epilepsy; convulsions (size: 4.56 %, diff: 40.48 %)	.58	.58	.57	.58	.50	.50	.46	.59	.59	.59	.59	.52	.58	.58	.58	.59	.60	.60	.57	.58	.58			
Ethnicity: Black/African American (size: 9.54 %, diff: 50.51 %)	.50	.49	.49	.49	.30	.56	.60	.49	.50	.49	.49	.48	.48	.49	.49	.50	.49	.50	.49	.49				
Ethnicity: White (size: 71.16 %, diff: 32.99 %)	.50	.51	.51	.51	.50	.53	.49	.52	.51	.52	.52	.51	.52	.52	.52	.51	.51	.51	.50	.52	.52			
Female (size: 44.99 %, diff: 31.46 %)	.50	.54	.53	.54	.59	.50	.49	.54	.53	.53	.54	.51	.53	.54	.54	.54	.53	.53	.50	.54	.54			
Hypertension with complications and secondary hypertension (size: 10.76 %, diff: 51.36 %)	.50	.53	.53	.53	.47	.47	.42	.53	.51	.52	.53	.51	.52	.53	.53	.51	.52	.49	.53	.53				
Male (size: 55.01 %, diff: 29.93 %)	.52	.48	.49	.49	.48	.51	.50	.48	.49	.48	.48	.51	.49	.48	.48	.49	.48	.51	.48	.48				
Newborn (size: 14.58 %, diff: 69.90 %)	.95	.75	.83	.82	.27	.36	.34	.81	.98	.97	.81	.88	.92	.80	.80	.79	.94	.92	.87	.87	.87			
Thyroid disorders (size: 8.18 %, diff: 39.12 %)	.50	.54	.53	.53	.39	.39	.44	.54	.52	.53	.53	.50	.52	.54	.54	.53	.53	.50	.54	.54				

Domain adaptation experiment



Insights from the experiments

- Uncertainty estimation techniques fail to identify novel examples, even in “obvious” cases
- ReLU architectures do the opposite of what we want
- Density estimation techniques perform better at this, but also not great



Trust Issues: Uncertainty Estimation Does
Not Enable Reliable OOD Detection On
Medical Tabular Data

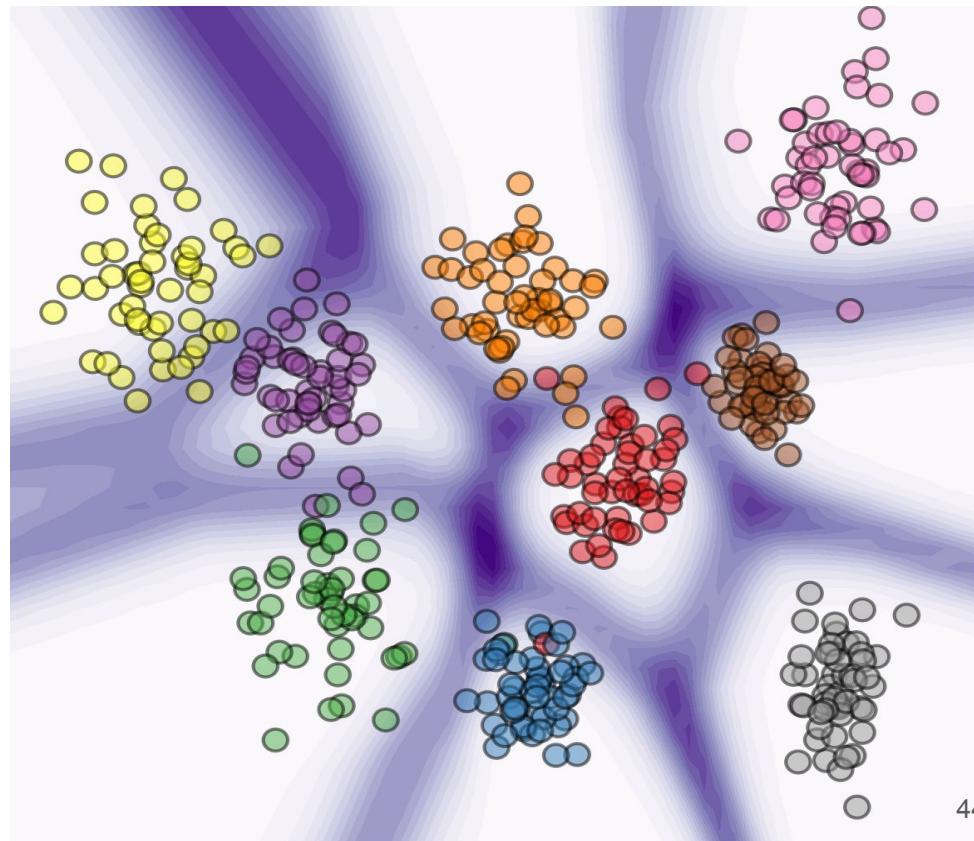
Dennis Ulmer, Lotta Meijerink, Giovanni Cinà Proceedings of the Machine Learning for Health
NeurIPS Workshop, PMLR 136:341-354, 2020.

The root of the problem: overconfidence

Neural Networks that use uncertainty to detect OOD points seem to suffer from severe overconfidence.

If we take one feature and scale it up by A LOT then the models are still very certain.

Consequence: some models are more certain at classifying OOD points than in-domain data!

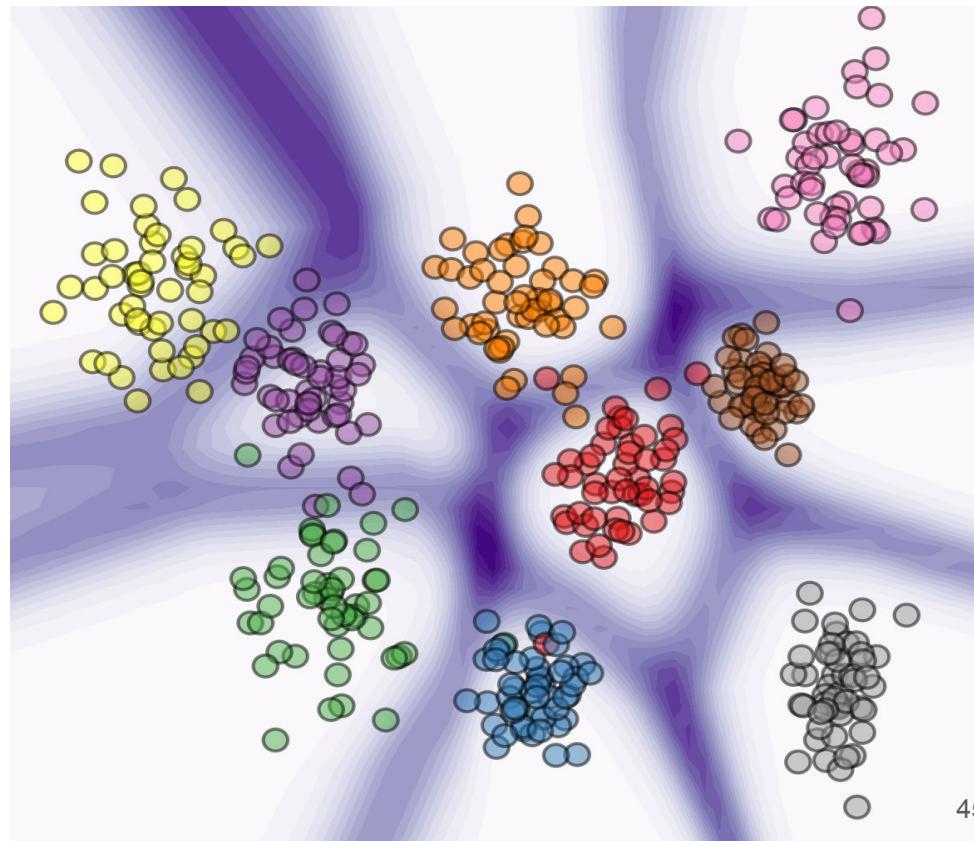


The perturbation experiment again

If we take one feature and scale it up by A LOT then the models are still very certain.

Now suppose the feature we are scaling up is

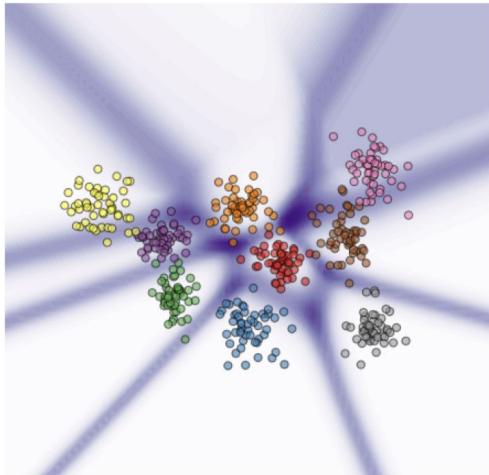
1. Predicting recidivism for convicts
 - a. Amount of previous felonies
2. Predicting risk of mortgage default
 - a. amount of debt
3. Almost any medical problem



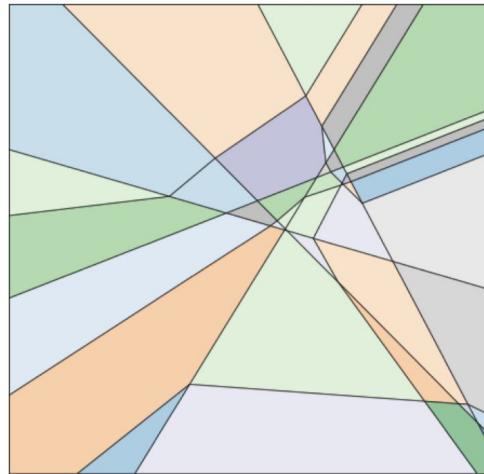
This raises some questions

- Can this behaviour observed on synthetic data be proven to be a systematic bias?
- Does this phenomenon apply to several uncertainty metrics?
- Which network architectures are affected by this?

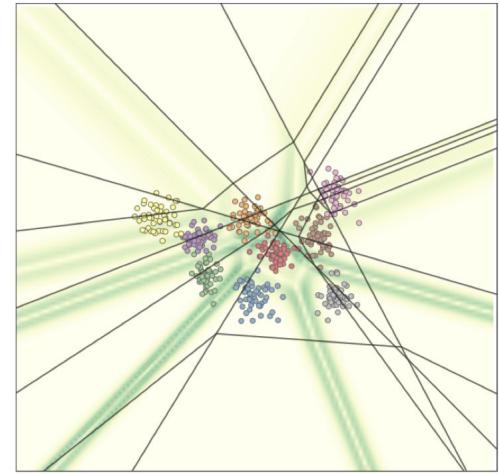
Broken mirrors: ReLU networks are piecewise affine functions



(a) Predictive entropy $\tilde{H}[p_{\theta}(y|\mathbf{x})]$ of ReLU classifier.



(b) Polytopal linear regions induced by same classifier [Arora et al., 2018].



(c) Magnitude of gradient of predictive entropy $\|\nabla_{\mathbf{x}} \tilde{H}[p_{\theta}(y|\mathbf{x})]\|_2$.

Intuition: generalization behavior is due to linearity on the polytopes

Our theoretical result

Theorem 1 (Convergence of uncertainty in the limit)

Given a set of ReLU networks, suppose that their Jacobian matrices with respect to the input do not contain any zero entries. Then, whenever uncertainty is measured via either of the following metrics

1. Max. softmax probability (Hendrycks & Gimpel, 2017)
2. Class variance (Smith & Gal, 2018)
3. Predictive entropy (Gal & Ghahramani, 2016)
4. Mutual information (Smith & Gal, 2018)

the network(s) will converge to fixed uncertainty scores when scaling a feature of an input in the limit.

technical conditions on the network and the polytopes

No matter how you measure uncertainty, by scaling a feature the uncertainty stabilizes

- Holds for: Single networks, Ensembles, MC Dropout, Bayes-by-backprop etc. (forms of Bayesian model averaging)
- We showcase this behaviour for several models in experiments on synthetic data

[Submitted on 9 Dec 2020 ([v1](#)), last revised 26 Feb 2021 (this version, v3)]

Implementing OOD detection for a specific medical use case: development and deployment

Machine Learning without **OOD detection**:

Development

1. Gather data
2. Train a predictive model
3. Evaluate performance of the predictive model with ground-truth labels

Deployment:

4. Get new input
5. Predict on new inputs

Implementing OOD detection for a specific medical use case: development and deployment

Machine Learning with **OOD detection**:

Development

1. Gather data
2. **Train an OOD detector on this data**
3. **Evaluate performance of the OOD detector**
4. Train a predictive model
5. Evaluate performance of the predictive model with ground-truth labels

Deployment:

6. Get new input
7. **Check new inputs with OOD detectors**
8. Predict on new inputs

Implementing OOD detection for a specific medical use case: ...how exactly?

1. OOD samples typically come *after* development... how do we train and select an OOD detector?
2. How do we medically validate an OOD detector?
3. How do we ensure that an OOD detector can catch all possible OOD samples?
4. Once we flag an OOD sample, what happens?

Our contribution: guidelines for implementing OOD detection in medical AI use cases

- We describe variables influencing performance of OOD detectors
- We show how to create OOD tests from available data
- How to validate OOD detection with interpretability tools
- Show a practical example on real-life EHR data
- Github repository to apply to any tabular datasets

Out-of-Distribution Detection for Medical Applications:
Guidelines for Practical Evaluation

Karina Zadorozhny
Pacmed BV - Amsterdam, The Netherlands

KARINA.ZADOROZHNY@GMAIL.COM

Patrick Thoral
Paul Elbers

Department of Intensive Care Medicine, Laboratory for Critical Care Computational Intelligence (LCCCI), Amsterdam Medical Data Science (AMDS), Amsterdam UMC, Vrije Universiteit, Amsterdam, The Netherlands

P.THORAL@AMSTERDAMUMC.NL
P.ELBERS@AMSTERDAMUMC.NL

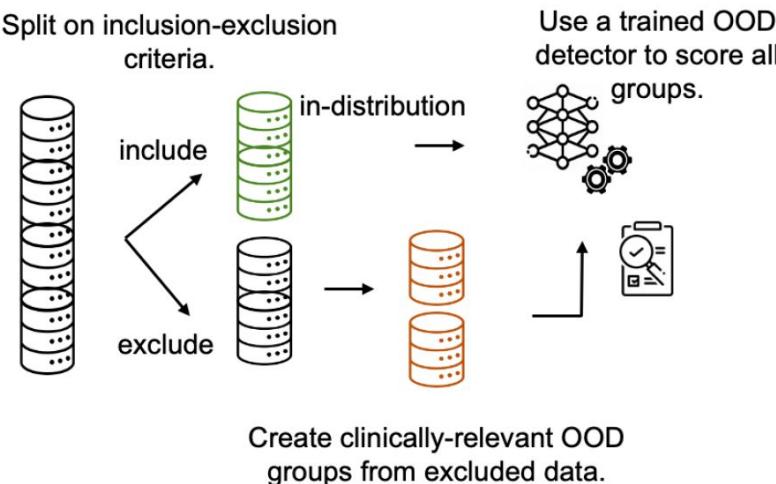
Giovanni Cinà
Pacmed BV - Amsterdam, The Netherlands

GIOVANNI.CINA@PACMED.NL

How to design OOD detection tests for medical data? An example

- Medical data often require definition of inclusion-exclusion criteria
→ Use these groups as OOD

A. Using excluded data



Practical example on real-world EHR data

Dataset:

AmsterdamUMC ICU dataset

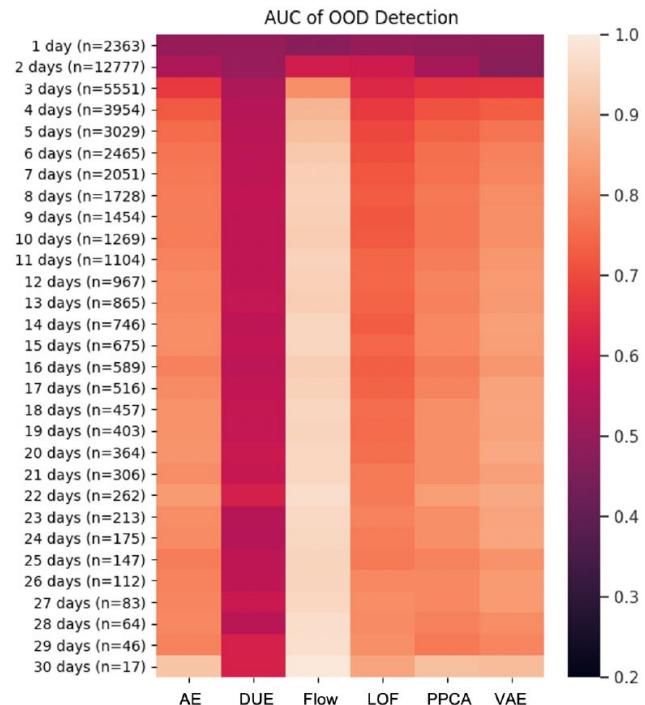
- tabular data
- mixed type data (continuous and categorical)
- downstream task: prediction of hospital readmission at discharge time
- unbalanced: only 5% adverse outcomes

Density estimators:

- Autoencoder (AE)
- Variational Autoencoder (VAE)
- Local Outlier Factor (LOF)
- Deterministic Uncertainty Estimation (DUE)
- Probabilistic PCA (PPCA)
- Normalizing Flow

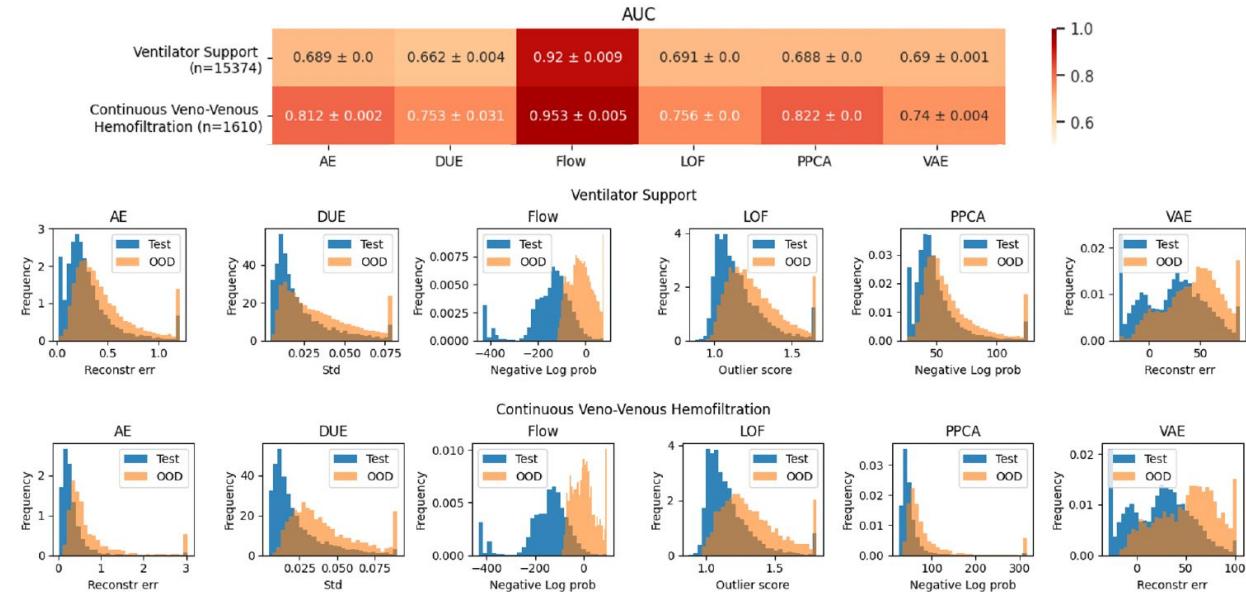
Comparing detectors on real-world EHR data

- Detecting patients that are far from discharge



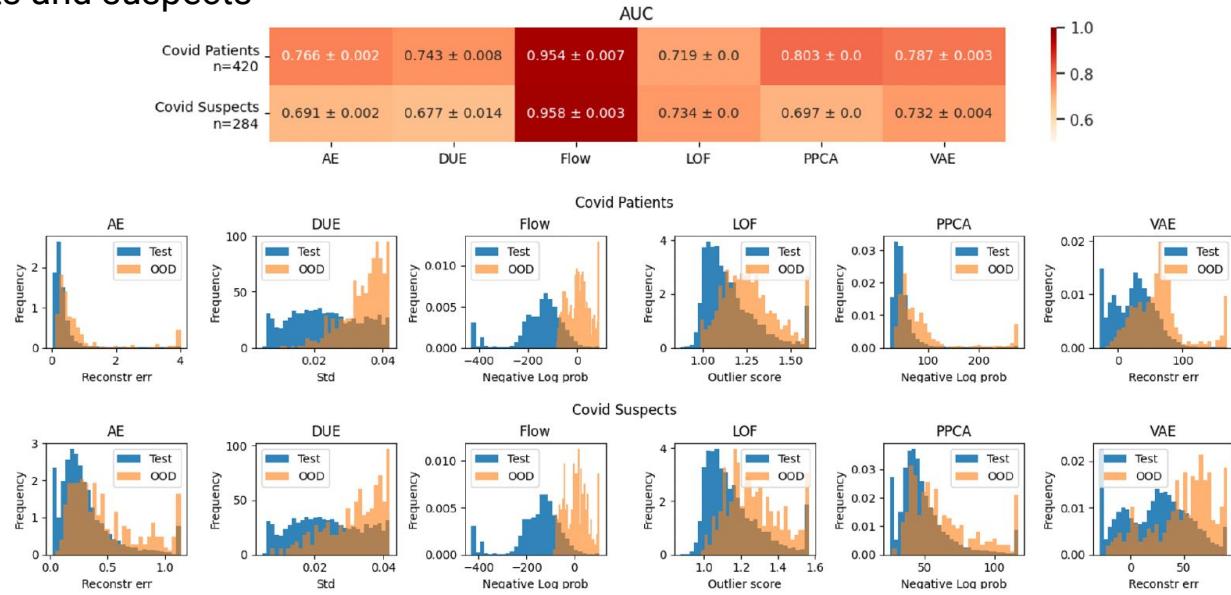
Comparing detectors on real-world EHR data

- Detecting patients that are far from discharge
- Detecting patients on ventilation and CVVH



Comparing detectors on real-world EHR data

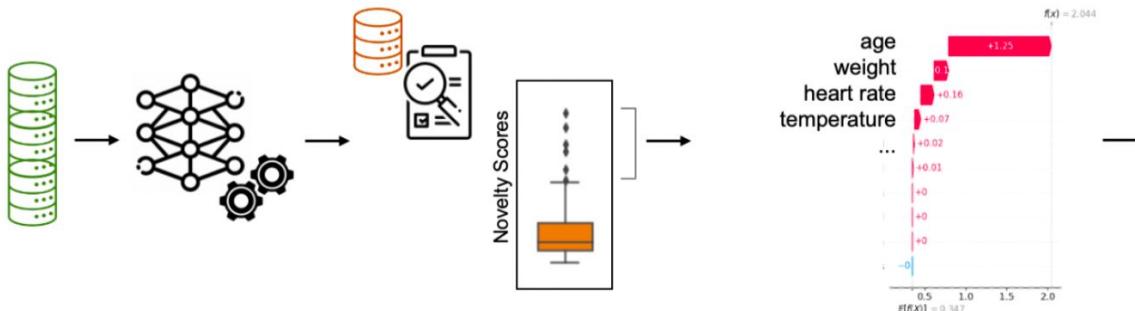
- Detecting patients that are far from discharge
- Detecting patients on ventilation and CVVH
- Detecting COVID-19 patients and suspects



Checking validity of OOD detectors with interpretability tools

- Assess interpretability on a dataset-level
- Inspect important features individually with clinicians (qualitative)

B. Qualitative inspection of outliers with medical experts



1. Train OOD detectors on in-distribution train data.

2. Use the OOD detector to score test data. Inspect the highest scoring samples.

3. For each sample, use SHAP to rank features.

4. Compare feature values of the sample with the rest of the data. Assess importance with clinicians.

Feature	Sample Value	Data Mean	Data Var
age	38	67	14
weight	63	76	34
heart rate	124	80	20

Where we are when it comes to reliable OOD detection, and what comes next

Preliminary conclusions:

- It is better to have a decent (although not great) OOD detector than none at all
- We have some working solutions
- What is the best model is rather case-dependent

Next steps:

- What is missing is a principled solution (**models that know what they don't know**) -> Theoretical work
- More robust round of tests to ensure that **models work well in real-world scenarios** -> benchmarking and community challenge

Agenda

1. A tool to aid discharge decisions in the ICU
2. Engage with AI -> Explainable AI
3. Data shift -> Out-of-Distribution detection
4. **Treatment effect estimation -> Causal Inference**

Causal Inference in the ICU: (some of) the problems

1. Assessing treatment effect for treatment of dynamic length
2. Estimating the adverse side effects of medications

Causal Inference in the ICU I: estimating effects for treatments of dynamic length

In the ICU patients receive some treatments 'as long as needed', meaning:

- Treatment starts when some conditions are met
- Duration is not fixed
- The necessity of treatment is periodically re-evaluated

Question: how much is enough, and how much is too much?

Admission to ICU

Mechanical ventilation

Proning

...

Causal Inference in the ICU I: estimating effects for treatments of dynamic length

At every decision point, we would want to estimate the effect of continuing or stopping treatment.

Example: right now Pacmed Critical shows only the risk when the discharge option is taken, but not if it is *not* taken.

We have a 3-year project funded on this topic, starting this spring.

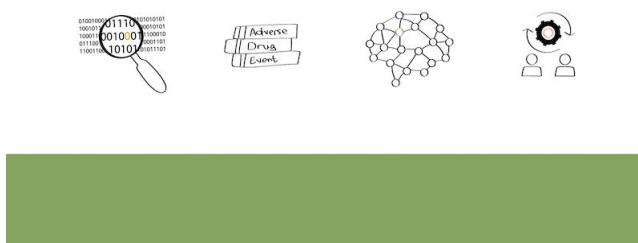


Medegefinancierd door
de Europese Unie

Causal Inference in the ICU II: estimating side effect of medications

Medications undergo RCTs to test effects on clinical outcomes, but often adverse drug events are not thoroughly researched.

Example: nephrotoxicity of antibiotics in the ICU.



In summary: what we are working on

1. A tool to aid discharge decisions in the ICU
2. Explainable AI
3. Out-of-Distribution detection
4. Causal inference

Who is (or will be) working on it

1. A tool to aid discharge decisions in the ICU
2. Explainable AI
3. Out-of-Distribution detection
4. Causal inference



UNIVERSITY
OF AMSTERDAM

pacmed



If any of the topics above is of interest, we are happy to collaborate!

Q&A

