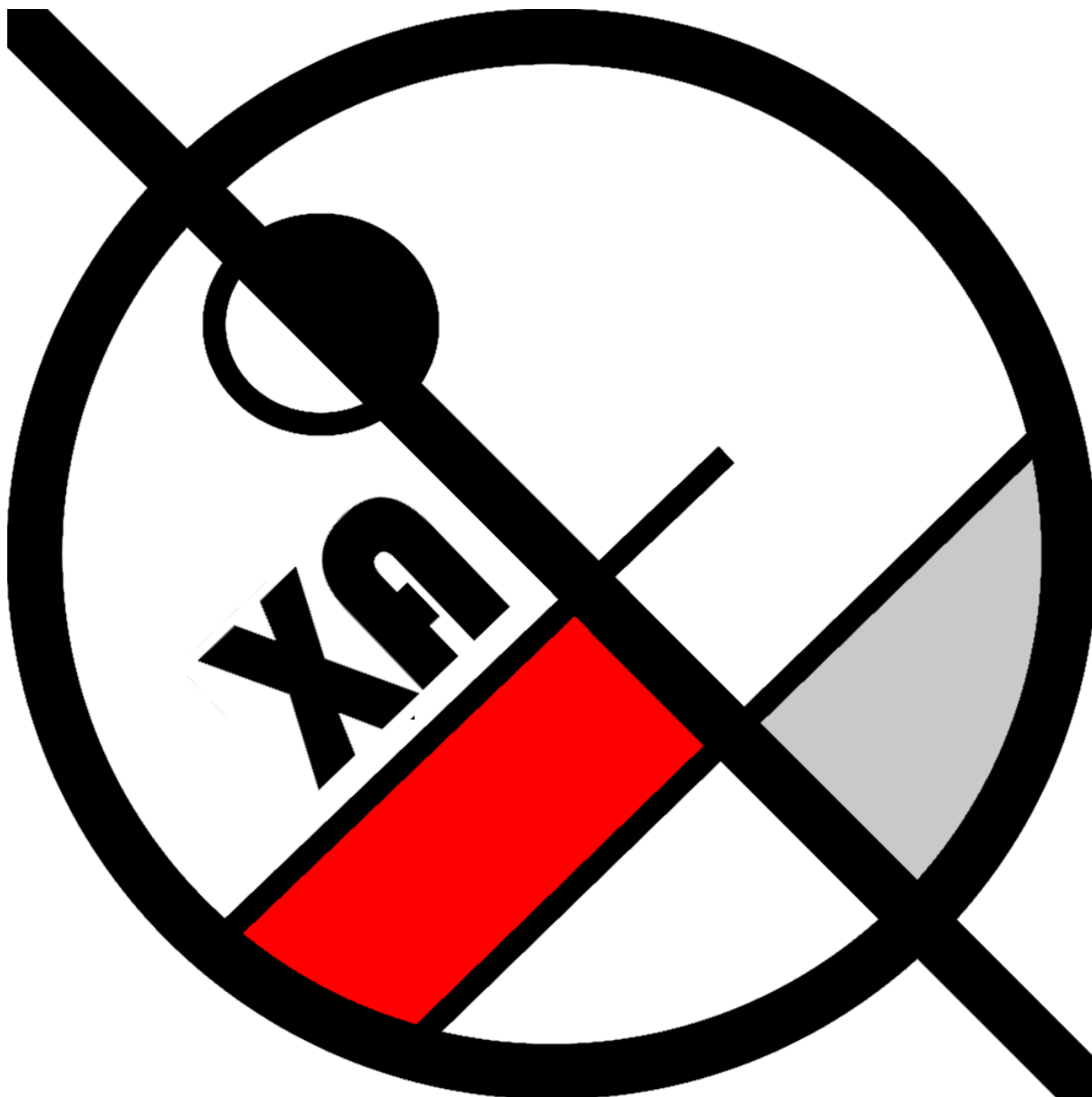


# Overdracht JADS



## Inleiding

Met behulp van een interactief dashboard willen we het mogelijk maken dat gebruikers, artsen/patiënten, kunnen interacteren met een AI-model. We willen verschillende vormen van explainable AI (XAI) mogelijk maken en meten wat het effect van uitleg is op het vertrouwen van de gebruiker op het AI-model. Het beoogde doel is dus een applicatie die gebruikt kan worden in onderzoek naar welke vormen van XAI effectief zijn in het uitleggen van modellen gebaseerd op tabulaire data.

In dit document wordt beknopt beschreven wat er gemaakt is in het semester 2024, hoe men er mee om gaat, en hoe de volgende groep het beste verder kan werken.

## Inhoudsopgave

Inleiding .....	2
SD-componenten:.....	4
Similar Patiënts.....	4
Change Values.....	5
Input Features .....	6
Counterfactuals.....	7
Global en Local Explanation Grafieken .....	9
Zoekfunctionaliteit .....	11
Patient en Behandelingen component .....	12
Overige taken.....	13
Database.....	14
Github Stories gebruik .....	15
AI .....	17
Explanations.....	17
Model.....	17
Counterfactuals.....	17
Vervolgstappen.....	17
Bronnen .....	19
Literatuur .....	19
Contact info.....	20
Appendix .....	21
A.....	21

## SD-componenten:

Deze applicatie heeft meerdere componenten die zijn opgesplitst. In het vervolg van dit hoofdstuk wordt er per hoofdstuk behandeld wat elk component doet, wat wij eraan hebben aangepast, wat Danielle er mogelijk nog aan wil laten aanpassen/toevoegen en hoe je de backend ervan gebruikt. Hoe je de applicatie opstart kan je in de instructions.md in de setup folder lezen.

### Similar Patiënts

Doel van de component:

Het doel van het *similar patiënts* component is om de patiënten die het meeste op de huidige patiënt lijken te filteren en te laten zien. Dit wordt gedaan door middel van een backend functionaliteit die het AI-model gebruikt om de patiënten te vergelijken. Je moet daarvoor aangeven op welke features er moet worden vergeleken en hoeveel gelijke patiënten er moeten worden terug gegeven.

Wat wij hebben gedaan:

- Ervoor gezorgd dat er *similar patiënts* uit de dataset worden opgehaald i.p.v. hardcoded.
- Een begin aan een filterfunctie gemaakt om patiënten op geselecteerde features te vergelijken.
- Een zoekveld toegevoegd om de hoeveelheid gewenste *similar patiënts* te bepalen.
- De progress-bar aangepast zodat het een weergave van het aantal *similar patiënts* met SAD weergeeft.

Wat Danielle nog wil veranderen aan dit component:

- Nog inzichtelijker maken waar de *similar patiënt* op gelijk zijn, en de verschillen met een kleur highlighten en In stappen kunnen filteren/sorteren.
- Front-end mooier maken.
- Filter functionaliteit moet nog afgemaakt worden.

Backend Endpoint:

- Methode: POST
- Url: [http://localhost:5000/get\\_similarity](http://localhost:5000/get_similarity)
- Request body: {"amount": 2, "ID": 7552, "filter": filter}
- Response body: {"Confidence": [66.9800,32.34350], "Patients": [{patient}, {patient.}, "Predictions": [0.6632,0.32252]}
- Betekenis variabelen:

- **Amount:** hoeveelheid gewenste similar patiënten (komt voort uit het input veld).
- **ID:** het patiënt ID van de huidige ingeladen patiënt.
- **Filter:** lijst met filterwaarden
- **Confidence:** de confidence scores van de similar patiënten op index.
- **Patiënten:** de similar patiënten op index.
- **Predictions:** de predictions van de similar patiënten op index.

## Change Values

Doel van de component:

Het doel van het *change values* component is om te kijken of er bij een aanpassing van een waarde van een of meerdere features er een andere predictie en zekerheidsscore door het AI-model wordt gegenereerd. Wij hebben ervaren dat het aanpassen van de feature 'vent' er vaak toe leidt dat er een andere predictie wordt gegenereerd.

Wat wij hebben gedaan:

- Ervoor gezorgd dat het component de data van de nieuwe dataset gebruikt.
- Het AI-model met de applicatie gekoppeld.

Wat Danielle nog wil veranderen aan dit component:

- 'Ethnicity' en 'ICUnits' als dropdown maken.
- 1 en 0 waarden veranderen naar checkboxes.
- Front-end mooier maken.
- Geen onzin/invalid data in de input velden kunnen stoppen.

Backend Endpoint:

- Methode: POST
- Url: <http://localhost:5000/predict>
- Request body: {patient}
- Response body: {"prediction": 0.4444518983364105, "confidence": 55.55}
- Betekenis variabelen:
  - **Patient:** de input waarden van de aanpassingen die op de patiënt zijn gedaan (de invulvelden).
  - **Prediction:** de nieuwe predictie op de aangepaste data (boven de 0.5 heeft iemand SAD onder de 0.5 heeft iemand geen SAD).
  - **Confidence:** de confidence scores van de nieuwe predictie.

## Input Features

Doel van de component:

Het doel van het input features component is om de features die het AI-model en dus de applicatie gebruikt uit te leggen en aan te geven welke waarde elke feature van een patiënt heeft.

Wat wij hebben gedaan:

- Ervoor gezorgd dat het component de data van de nieuwe dataset gebruikt.

Wat Danielle nog wil veranderen aan dit component:

- ICUUnits en Ethnicity groeperen
- Alle 1 en 0 waarden veranderen naar 'true' en 'false' (in de backend moeten ze nog steeds 0 en 1 zijn).
- Front-end mooier maken.
- De dropdown in het menu in de sidebar moet een andere naam hebben.

## Counterfactuals

Doel van de component:

Het doel van het *counterfactuals* component is om op basis van verschillende methodes een zelf bepaalde hoeveelheid *counterfactuals* te generen. Een *counterfactual* is een scenario van de patiënt die een andere predictie genereert maar toch zoveel mogelijk op de huidige patiënt lijkt. Hiervoor zijn momenteel dus 3 methodes: *genetic*, *feature select* en *KDtree*, waarvan *genetic* de default methode is. Het kan daarnaast ook zo zijn dat er voor sommige methodes soms geen *counterfactuals* mogelijk zijn. In dat geval wordt er een error terug gegeven.

Als je de *genetic* methode kiest wordt er een fictieve patiënt gegenereerd om het kantelpunt van de voorspelling te vinden; hij geeft dus een fictieve patiënt terug die zoveel mogelijk op de huidige patiënt lijkt, maar toch een andere voorspelling heeft. Deze methode duurt echter wel erg lang (8 sec).

Als je de *random* methode kiest wordt er net als bij de *genetic* een fictieve patiënt gegenereerd, alleen dan worden er alleen waarden aangepast aan de zelfgekozen features. Stel je voor dat je alleen de leeftijd kiest voor de *random* methode, dan worden er alleen patiënten gegenereerd die een andere predictie hebben en alleen een wijziging bij de 'age' feature hebben. Momenteel kan je nu nog niet zelf bepalen welke features worden aangepast, dit is nu nog hardcoded in het AI-model. De features die nu worden aangepast zijn: 'age', 'weight', 'temperature' en 'gcs'.

Als je de *KDtree* methode kiest wordt er in de dataset naar een bestaand patiënt gezocht die zoveel mogelijk op de huidige patiënt lijkt, maar wel een andere voorspelling heeft.

Wat wij hebben gedaan:

- De AI-methodes gemaakt en geïmplementeerd.
- Een zoekveld toegevoegd om de hoeveelheid *counterfactuals* te bepalen.
- Ervoor gezorgd dat er een dynamisch aantal *counterfactuals* te zien zijn.
- Een drop down toegevoegd om de methode te bepalen.

Wat Danielle nog wil veranderen aan dit component:

- Highlighten wat de verschillende waarden zijn.
- Een korte uitleg in het front-end toevoegen wat elke counterfactual methode doet en inhoudt. (Hoe ze aan hun data komen)
- Ethnicity en Units groeperen en alle 1 en 0 waarden aanpassen naar true en false.
- Front-end mooier maken.

#### Backend Endpoint:

- Methode: POST
- Url: <http://localhost:5000/counterfactual>
- Request body: {"amount": 2, "data": patient, "method": "GENETIC"}
- Response body: [{counterfactualPatient}, {counterfactualPatient}]
- Betekenis variabelen:
  - **Amount:** hoeveelheid gewenste counterfactuals (komt voort uit het input veld).
  - **Method:** de gewenste methode om de counterfactuals te bepalen.
  - **CounterfactualPatient:** de counterfactual patiënt die is opgehaald.



## Global en Local Explanation Grafieken

Doel van de component:

Het doel van de *explanation Grafieken* componenten is om het AI-model begrijpelijker te maken voor de gebruikers, specifiek de technische artsen. Je kan verschillende grafieken kiezen en e

lke grafiek heeft weer andere input data nodig. Voor de uitleg van elke specifieke *explanation Grafiek* verwijs ik naar de AI-documentatie. De *Local Grafieken* hebben een ingeladen patiënt nodig. De *Global Grafieken* hebben dat niet nodig.

Er is een specifieke explanation die de AI'ers wel hebben gemaakt maar die niet in de applicatie is gekomen. Dat is de Interactive Force Plot, dit komt omdat het AI-model daarbij een interactief html scherm terugstuurt. Dit kan helaas, voor zover wij weten, niet buiten de Jupyter Notebook worden weergegeven.

Wat wij hebben gedaan:

- Koppeling met het AI-model gemaakt.
- Dropdown en input velden voor de verschillend grafieken gemaakt.

Wat Danielle nog wil veranderen aan dit component:

- Misschien een oplossing bedenken om toch wel de interactieve force plot in de applicatie te krijgen.
- Front-end mooier maken.
- De overige explanations die klaarstaan in de backend in de frontend toevoegen (er is een explanation die html teruggeeft).

Backend Endpoint:

- Methode: POST
- Url: [http://localhost:5000//generate\\_shap](http://localhost:5000//generate_shap)
- Request body: {"feature\_name": "age", "index": 7552, "interaction\_index": "temperature", "list\_indices": [0,2,525], "max\_display": 5, "range": [0, 50], "type\_visualization": "BAR\_PLOT"}
- Response body: {"Image": "base64"}  
Of: {"html": html....}
- Betekenis variabelen:

- **Feature\_name:** de naam van een feature waar naar gekeken moet worden.
- **Index:** het patiënt ID/index van de huidige ingeladen patiënt in de dataset.
- **Interaction\_index:** de naam van een feature waar mee vergeleken moet worden.
- **List\_indices:** de indexes van de patiënten waar naar gekeken moet worden.
- **Max\_display:** de hoeveelheid features waar de importance voor wordt laten zien.
- **Range:** de patiënt indexen waar tussen wordt gekeken.
- **Type\_visualization:** het type visualisatie/grafiek.

## Zoekfunctionaliteit

Doel van de component:

Het doel van het *Zoek Functionaliteit* component is om patiënten uit de dataset op te zoeken. Als je wilt zoeken heb je meerdere opties om op te filteren. Daarnaast kan je ook op nog meer features zoeken. Dit doe je door de geavanceerde opties te openen. Dit zorgt ervoor dat de patiënten die die input data bevatten worden teruggegeven, daarnaast kan je ook op patiënt ID zoeken (wat eigenlijk gewoon de index van de patiënt in de dataset is).

Wat wij hebben gedaan:

- Ervoor gezorgd dat het component de data van de nieuwe dataset gebruikt.
- De filters gemaakt.

Wat Danielle nog wil veranderen aan dit component:

- Ethnicity en Units een dropdown met checkboxen aanmaken, dan ben je niet gebonden aan 1 specifiek ethnicity of Unit bij het zoeken.
- Een range gebruiken bij alle numerieke waarden (zoals bij de leeftijd wordt gedaan). Dan ben je niet gebonden aan 1 leeftijd bij het zoeken.
- Niet bij specifieke waarden de drempels zoeken (thresholds voor de slider).
- Niet een hele rij van paginanummers weergeven, eerder 1, 2, 3 ... 20, 21, 22 .... 55, 56, 57 voor als je op pagina 21 zit.
- Je moet naast de more info knop ook op de hele balk van de patiënt kunnen klikken om naar het overview scherm te gaan.
- De patiënt wordt ingeladen bij het klikken op more info, misschien de naam veranderen naar 'patiënt inladen'.
- Je moet direct na het zoeken een patiënt in kunnen laden zonder nog op andere knoppen te hoeven klikken.
- Front-end mooier maken.
- Of de patient wel of geen SAD heeft.

#### Backend Endpoint:

- Methode: POST
- Url: <http://localhost:5000/searchvalues>
- Request body: {lijst met zoekwaarden}
- Response body: {[patiënts]}
- Betekenis variabelen:
  - Lijst met zoekwaarden: Dit is een lijst waar kenmerken in kunnen staan. De lijst bestaat uit een column key. Bijvoorbeeld ID is de column en de key is dan 100.
  - Patiënts: Dit is een lijst van alle patiënten gefilterd op de zoekwaarden.

#### Patient en Behandelingen component

##### Doel van de component:

Het doel van dit component is om een dokter de mogelijkheid te geven om zelf een behandeling en patient te kunnen aanmaken en invullen. Dit was echter een idee van ons zelf waar Danielle erg enthousiast over was. Helaas zijn we hier door tijdgebrek niet aan toegekomen. Mocht Danielle dit nog steeds interessant vinden staan er nog stories in het project die gaan over het implementeren van de patient en behandelingen CRUD frontend en backend.

## Overige taken

- User logging: voor vervolgonderzoek moet er bijgehouden worden wat een gebruiker exact gedaan heeft op het dashboard . (Story #695)
- UI taal in de hele applicatie aanpassen. (Story #310)
- Verschillende gebruikers typen aan de applicatie toevoegen. (Story #324)
- Gebruikers en behandelingen zelf kunnen aanmaken. (Story's: #325, #326, #327 en #328)
- De similar patiënts waarden die gelijk aan de ingeladen patiënt zijn highlighten. (Story #336)
- Het design van het pop-up scherm bij de change values verbeteren. (Story #286)
- Een onderzoek naar de LIME explanations, dit wordt in het AI-hoofdstuk verder behandeld.
- Aanpassingen aan de input features component: per feature laten zien hoe belangrijk het voor de predictie is, duidelijk maken wanneer er bij een predictie geen data is. (Story #278)
- Aanpassingen aan de counterfactuals component: Kleuren toevoegen op basis van het percentage van de zekerheidsscore. (Story #303)
- Bedenken wat je met het Global explanations scherm in het dashboard wilt.
- Aanpassingen aan de change values component: Het aanpassen van waarden moet lower case vriendelijke zijn, Velden vastzetten op een datatype, bestaande error meldingen in de console weghalen, per waarde aangeven wat de kantelpunten in de predictie zijn, een andere naam voor de change values bedenken, onderzoek doen naar een sensitiviteits-analyse, drop down van de units en ethnicity maken, checkboxes voor de true-or-false waarden maken, duidelijk maken wanneer er een nieuwe predictie is gegenereerd, de SAD progress bar implementeren in de change values. (Story #291)
- Error logging verbeteren. (Story #216)
- De andere explanations (visualisaties) aan de frontend toevoegen, momenteel werken alleen de SHAP-visualisaties in de frontend, maar dit moet nog worden opgesplitst in een LOCAL en een GLOBAL scherm (alles wat de index/patient ID gebruikt is een LOCAL explanation). Aan die schermen moeten dus ook de overige explanations worden toegevoegd. De backend van deze explanations werken alwel alleen het is nog niet aan de frontend toegevoegd.
- Filter functionaliteit bij de similar patients afmaken. Momenteel is er nog een error bij een nieuwe methode (maar\_dan\_beter). Hier is het belangrijk om of die error te fixen, of zelf een nieuwe methode maken en verbinden met de rest. Wanneer er op geen waarden worden gefilterd wordt er een None gestuurd naar de back-end. (Story #633)
- Log-in functionaliteit

## Database

Wij hebben er in samenspraak met Danielle voor gekozen om geen database meer te gebruiken (oorspronkelijk werd een PostgreSQL database gebruikt). Dit hebben wij gekozen omdat er een hele nieuwe dataset door het AI-model wordt gebruikt.

Daarnaast hebben een aantal explanations altijd de gehele dataset nodig. Dit heeft ervoor gezorgd dat wij dus hebben besloten om geen database toe te voegen. In productie zou je daarentegen wel een database willen hebben.

## Github Stories gebruik

Wij hebben verschillende 'views' gemaakt om het gebruik van het Github Project te vergemakkelijken en makkelijk en inzichtelijk te kunnen inzien welke taken in welke sprint zijn ingepland. Dit zijn de volgende views:

- **Current iteration view:** laat alle taken in het project zien die aan de huidige iteratie zijn gekoppeld in een overzichtelijk scrum bord. Het doel hiervan is om dit tijdens de daily standup te gebruiken. Taken zijn in dit geval de 'issues/taken' die in een story zitten (filtert op de tag: 'task'), het zijn dus de taken die je daadwerkelijk uitvoert.

Scenario's in de sprint:

- Als je tijdens de sprint met een nieuwe taak of story komt en die echt direct gedaan moet worden kan je die in de 'New Issues' status zetten. Dan kan dit de volgende standup makkelijk worden behandeld en besproken en vergeet je het niet in de burndown chart te stoppen.
  - Als je tijdens de sprint erachter komt (in overleg met Danielle) dat een story/taak of dubbel is met een andere story/taak, kan je die de 'Unnecessary' status geven. Dan kan dit de volgende standup makkelijk worden behandeld en besproken en vergeet je het niet uit de burndown chart te halen. En staat het nog steeds ergens gelogd.
  - Als je een taak op een dag hebt afgerond kan je die naar 'Review' slepen. Dan kan dit de volgende standup makkelijk worden behandeld en besproken en vergeet je niet de burndown chart te updaten.
  - Als je een taak voorlopig niet meer gaat doen of niet meer gaat afmaken kan je die naar 'On hold' slepen.
- **Next iteration view:** Is hetzelfde als de current iteration view alleen dan voor de volgende iteratie.
  - **story-board view:** laat alle stories die in de huidige iteratie zitten zien. Hij gebruikt dus een filter om alle story typen te laten zien. Het doel hiervan is hetzelfde als de current iteration view alleen dan voor stories.

Story typen tags:

- Enable Story
  - Research Story
  - User Story
  - Learning Story
  - Actiepunt (een story met daarin alle actiepunten voor die sprint)
- **Next iteration story-board view:** is hetzelfde als het story-board view alleen dan voor de volgende iteratie.
  - **Stories per label per iteratie view:** is een view waarin je alle stories in het project kan zien, dit kan je bekijken per iteratie en vervolgens per prioriteit. Deze view is bedoeld om het voorbereiden voor de planning overzichtelijk te maken. Je

kan dan namelijk makkelijk inzien welke stories er nog gedaan moeten worden en welke prioriteit ze hebben.

- **Taken per label per iteratie view:** is hetzelfde als het stories per label per iteratie view, alleen dan voor de taken, dit is bedoeld om makkelijk in te kunnen zien welke taken je in iteraties hebt ingepland en welke taken je nog niet hebt ingeschat.



## AI

In de repository, in de map documentatie/23-24, zijn drie notebooks te vinden waar het overgrote deel van het AI-werk staat.

- **overzicht-schermen.ipynb**: een overzicht van alle gemaakte explanations, voorzien met uitleg, commentaar m.b.t. implementatie en een uitgebreide glossary om niet-AI specialisten.
- **SAD\_train.ipynb**: hierin valt terug te vinden hoe het model getraind is.
- **visualisaties\_html\_png.ipynb**: dit notebook bevat instructies voor hoe explanations/visualisaties worden omgezet in een format waar het front-end iets mee kan, is ook voorzien van een aantal comments betreffende implementatie.

## Explanations

In de bijlage is een html export van 'overzicht-schermen.ipynb' te vinden. Deze is voorzien met uitgebreide documentatie voor al onze explanations. Dit bestand valt te openen met je favoriete browser.

## Model

Het model (xgb.pkl) is een reconstructie van het model in Yang et al. (2023), voor gedetailleerde informatie kan je het best die paper lezen. De data (MIMIC-IV.dta) is afkomstig van de volgende git repository: <https://github.com/bbycat927/SAD>, het is een subset van de MIMIC-IV dataset (Johnson et al., 2023). Voor een overzicht van de gebruikte features zie appendix A.

## Counterfactuals

Voor counterfactuals hebben we zelf twee algoritmen geschreven:

- Een KDTree van SKLearn om bestaande counterfactuals te zoeken in de dataset.
- Een genetisch algoritme om fictieve patiënten te genereren.

## Vervolgstappen

- Het genetisch algoritme is enigszins traag, er kan gekeken worden naar optimalisatie.
- Er zijn geen explanations op basis van LIME. Wij hebben dit links laten liggen gezien het enigszins vergelijkbaar is met SHAP, maar het kan ook zinnige dingen opleveren. (Story #344)
- Danielle heeft voorgesteld dat er met Large Language Models (LLMs) de explanations uitgelegd en geïnterpreteerd kunnen worden
- De 'outlier score' (zie 'overzicht schermen' notebook) is vrij basaal. Er kan gekeken worden naar betere manieren om outliers te bepalen.

- Het AI-Lectoraat van de HU werkt ook aan XAI; er is met hen al samengewerkt dit jaar, maar de explanations uit dat project zijn verder niet door ons verwerkt. Er zou contact opgenomen kunnen worden met Henry Maathuis om te kijken naar mogelijkheden om dit in het dashboard te verwerken.

## Bronnen

[1] Johnson, A., Bulgarelli, L., Pollard, T., Horng, S., Celi, L. A., & Mark, R. (2023). **MIMIC-IV (version 2.2)**. *PhysioNet*. <https://doi.org/10.13026/6mm1-ek67>.

[2] Yang, Z., Hu, J., Hua, T., Zhang, J., Zhang, Z., & Yang, M. (2023). **Development of a machine learning-based prediction model for sepsis-associated delirium in the intensive care unit**. *Scientific Reports*, 13(1). <https://doi.org/10.1038/s41598-023-38650-4>

## Literatuur

[1] Van De Sande, D., Van Genderen, M. E., Verhoef, C., Van Bommel, J., Gommers, D., Van Unen, E., Huiskens, J., & Grünhagen, D. J. (2021). **Predicting need for hospital-specific interventional care after surgery using electronic health record data**. *Surgery*, 170(3), 790–796. <https://doi.org/10.1016/j.surg.2021.05.005>

[2] Van De Sande, D., Van Genderen, M. E., Verhoef, C., Huiskens, J., Gommers, D., Van Unen, E., Schasfoort, R. A., Schepers, J., Van Bommel, J., & Grünhagen, D. J. (2022). **Optimizing discharge after major surgery using an artificial intelligence-based decision support tool (DESIRE): An external validation study**. *Surgery*, 172(2), 663–669. <https://doi.org/10.1016/j.surg.2022.03.031>

[3] Pang, K., Li, L., Ouyang, W., Liu, X., & Tang, Y. (2022). **Establishment of ICU Mortality Risk Prediction Models with Machine Learning Algorithm Using MIMIC-IV Database**. *Diagnostics*, 12(5), 1068. <https://doi.org/10.3390/diagnostics12051068>

[4] Sun, Y., He, Z., Ren, J., & Wu, Y. (2023). **Prediction model of in-hospital mortality in intensive care unit patients with cardiac arrest: a retrospective analysis of MIMIC - IV database based on machine learning**. *BMC Anesthesiology*, 23(1). <https://doi.org/10.1186/s12871-023-02138-5>

[5] Ramaravind K. Mothilal, Amit Sharma, and Chenhao Tan (2020). **Explaining machine learning classifiers through diverse counterfactual explanations**. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. <https://arxiv.org/abs/1905.07697>

[6] M. Sahakyan, Z. Aung and T. Rahwan, **Explainable Artificial Intelligence for Tabular Data: A Survey**, in *IEEE Access*, vol. 9, pp. 135392-135422, 2021, doi: 10.1109/ACCESS.2021.3116481. <https://ieeexplore.ieee.org/document/9551946>

[7] Lundberg, S. M., & Lee, S.-I. (2017). **A Unified Approach to Interpreting predictions**. *Washi Washington: Curran Associates, Inc.* [https://proceedings.neurips.cc/paper\\_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf)

[9] Lundberg, S. M., & Lee, S.-I. (2017). **A Unified Approach to Interpreting Model predictions**. Washington: Curran Associates, Inc.  
[https://proceedings.neurips.cc/paper\\_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf)

## Contact info

Als er nog uitleg of hulp nodig is over de code of overige info, mail dan graag deze email:  
brouwer\_AI@outlook.com

# Appendix

## A

		Missing	Overall	NON-SAD	SAD
n			14620	9230	5390
age, mean (SD)		0	66.9 (15.9)	66.7 (15.8)	67.3 (16.1)
weight, mean (SD)		160	83.1 (23.6)	83.1 (23.0)	83.2 (24.6)
gender, n (%)	FEMALE	0	8518 (58.3)	5416 (58.7)	3102 (57.6)
	MALE		6102 (41.7)	3814 (41.3)	2288 (42.4)
race, n (%)	AISAN	0	426 (2.9)	311 (3.4)	115 (2.1)
	BLACK		1266 (8.7)	766 (8.3)	500 (9.3)
	HISPANIC		557 (3.8)	360 (3.9)	197 (3.7)
	OTHER		642 (4.4)	417 (4.5)	225 (4.2)
	WHITE		9723 (66.5)	6372 (69.0)	3351 (62.2)
	unknown		2006 (13.7)	1004 (10.9)	1002 (18.6)
first_careunit, n (%)	CCU	0	1366 (9.3)	881 (9.5)	485 (9.0)
	CVICU		3461 (23.7)	2772 (30.0)	689 (12.8)
	MICU		3078 (21.1)	1601 (17.3)	1477 (27.4)
	MICU/SICU		2706 (18.5)	1780 (19.3)	926 (17.2)
	NICU		534 (3.7)	234 (2.5)	300 (5.6)
	SICU		1887 (12.9)	1112 (12.0)	775 (14.4)
	TSICU		1588 (10.9)	850 (9.2)	738 (13.7)
temperature, mean (SD)		48	36.7 (0.8)	36.7 (0.8)	36.8 (0.9)
heart_rate, mean (SD)		1	89.7 (20.3)	88.2 (19.6)	92.3 (21.1)
resp_rate, mean (SD)		24	19.6 (6.0)	19.1 (5.9)	20.6 (6.1)
spo2, mean (SD)		5	97.1 (4.0)	97.3 (3.7)	96.7 (4.3)
sbp, mean (SD)		6	120.3 (23.9)	119.6 (23.1)	121.4 (25.1)
dbp, mean (SD)		15	66.5 (17.7)	65.7 (16.8)	67.9 (19.0)
mbp, mean (SD)		14	81.5 (17.8)	81.0 (17.0)	82.5 (19.0)
wbc, mean (SD)		115	13.1 (8.1)	12.8 (7.8)	13.7 (8.4)
hemoglobin, mean (SD)		96	10.3 (2.2)	10.3 (2.1)	10.5 (2.3)
platelet, mean (SD)		102	191.5 (106.0)	190.7 (105.3)	192.9 (107.1)
bun, mean (SD)		60	28.2 (22.9)	26.1 (21.0)	31.7 (25.6)
cr, mean (SD)		56	1.5 (1.5)	1.4 (1.5)	1.6 (1.6)
glu, mean (SD)		66	150.2 (74.4)	144.9 (66.5)	159.3 (85.5)
Na, mean (SD)		50	137.4 (5.5)	136.9 (5.0)	138.2 (6.2)
Cl, mean (SD)		51	103.8 (6.7)	103.8 (6.3)	103.9 (7.3)
K, mean (SD)		59	4.3 (0.8)	4.3 (0.8)	4.3 (0.9)
Mg, mean (SD)		608	2.0 (0.5)	2.0 (0.5)	2.0 (0.5)
Ca, mean (SD)		1399	8.2 (0.9)	8.2 (0.8)	8.2 (0.9)
P, mean (SD)		1341	3.8 (1.5)	3.7 (1.3)	4.0 (1.7)
inr, mean (SD)		1609	1.5 (0.8)	1.5 (0.7)	1.6 (0.8)
pt, mean (SD)		1578	17.0 (9.8)	16.7 (8.8)	17.5 (11.3)
ptt, mean (SD)		1658	37.8 (22.4)	37.2 (21.4)	39.0 (24.0)
bicarbonate, mean (SD)		57	22.2 (4.6)	22.5 (4.3)	21.7 (5.1)
aniongap, mean (SD)		64	15.0 (4.6)	14.4 (4.2)	16.0 (4.9)
gcs, mean (SD)		2	14.2 (2.4)	14.3 (2.4)	14.1 (2.3)
vent, n (%)	FALSE	0	8023 (54.9)	5974 (64.7)	2049 (38.0)
	TRUE		6597 (45.1)	3256 (35.3)	3341 (62.0)
crrt, n (%)	FALSE	0	14364 (98.2)	9152 (99.2)	5212 (96.7)
	TRUE		256 (1.8)	78 (0.8)	178 (3.3)
vaso, n (%)	FALSE	0	7482 (51.2)	4992 (54.1)	2490 (46.2)
	TRUE		7138 (48.8)	4238 (45.9)	2900 (53.8)
seda, n (%)	FALSE	0	7962 (54.5)	4968 (53.8)	2994 (55.5)
	TRUE		6658 (45.5)	4262 (46.2)	2396 (44.5)
sofa_score, mean (SD)		0	3.6 (1.9)	3.4 (1.7)	3.9 (2.2)
ami, n (%)	FALSE	0	12976 (88.8)	8293 (89.8)	4683 (86.9)
	TRUE		1644 (11.2)	937 (10.2)	707 (13.1)
ckd, n (%)	FALSE	0	11680 (79.9)	7417 (80.4)	4263 (79.1)
	TRUE		2940 (20.1)	1813 (19.6)	1127 (20.9)
copd, n (%)	FALSE	0	14088 (96.4)	8944 (96.9)	5144 (95.4)
	TRUE		532 (3.6)	286 (3.1)	246 (4.6)
hyperte, n (%)	FALSE	0	8322 (56.9)	5158 (55.9)	3164 (58.7)
	TRUE		6298 (43.1)	4072 (44.1)	2226 (41.3)
dm, n (%)	FALSE	0	11962 (81.8)	7477 (81.0)	4485 (83.2)
	TRUE		2658 (18.2)	1753 (19.0)	905 (16.8)
sad, n (%)	NON-SAD	0	9230 (63.1)	9230 (100.0)	
	SAD		5390 (36.9)		5390 (100.0)
aki, n (%)	FALSE	0	6462 (44.2)	4541 (49.2)	1921 (35.6)
	TRUE		8158 (55.8)	4689 (50.8)	3469 (64.4)
stroke, n (%)	FALSE	0	13479 (92.2)	8777 (95.1)	4702 (87.2)
	TRUE		1141 (7.8)	453 (4.9)	688 (12.8)
	TRUE		1852 (12.7)	703 (7.6)	1149 (21.3)

