

HOGESCHOOL ROTTERDAM / CMI

TINLAB

Machine Learning

TINLML03

Aantal studiepunten: 15
Cursusbeheerder: E.R. van der Ven

Cursusbeschrijving

Cursusnaam:	TinLab Machine LEarning																																																						
Cursuscode:	TINLML03																																																						
Kennisdomein:	Praktijklijn																																																						
Aantal studiepunten en studiebelastinguren:	<p>Dit studieonderdeel levert de student 15 studiepunten op, hetgeen overeenkomt met een studielast van 420 uren. De verdeling van deze 420 uren over 11 collegeweken (inclusief herkansingsweek) is als volgt. Let op, uren t.b.v. zelfstandig (onbegeleid) werken zijn schuin gedrukt.</p> <table><tr><td>Kick-off en gastcollege</td><td>1 x 2.0 uur</td><td>2 uur</td></tr><tr><td>Hoorcollege lessen</td><td>13 x 2.0 uur</td><td>26 uur</td></tr><tr><td>Practicum lessen</td><td>14 x 1.5 uur</td><td>21 uur</td></tr><tr><td><i>Wekelijks zelfstandig werken</i></td><td><i>8 x 30 uur</i></td><td><i>240 uur</i></td></tr><tr><td>Boekenclub bijeenkomsten</td><td>6 x 2.0 uur</td><td>12 uur</td></tr><tr><td><i>Vorbereiding boekenclub</i></td><td><i>6 x 2.0 uur</i></td><td><i>12 uur</i></td></tr><tr><td>Ethiek college</td><td>3 x 2.0 uur</td><td>6 uur</td></tr><tr><td>Onderzoek vaardigheden training</td><td>2 x 2.0 uur</td><td>4 uur</td></tr><tr><td><i>Vorbereiding mondeling</i></td><td><i>1 x 1.5 uur</i></td><td><i>1.5 uur</i></td></tr><tr><td>Mondeling</td><td>1 x 0.5 uur</td><td>0.5 uur</td></tr><tr><td><i>Vorbereiding challenge</i></td><td><i>1 x 25 uur</i></td><td><i>25 uur</i></td></tr><tr><td>Challenge</td><td>1 x 5 uur</td><td>5 uur</td></tr><tr><td><i>Werken aan de herkansing</i></td><td><i>1 x 39 uur</i></td><td><i>39 uur</i></td></tr><tr><td>Herkansing</td><td>1 x 1 uur</td><td>1 uur</td></tr><tr><td>SLC</td><td>1 x 20 uur</td><td>20 uur</td></tr><tr><td>Vragen open inloop bijeenkomsten</td><td>2 x 1.5 uur</td><td>3 uur</td></tr><tr><td></td><td>1 x 2.0 uur</td><td>2 uur</td></tr><tr><td>Totaal</td><td></td><td>420 uur</td></tr></table>	Kick-off en gastcollege	1 x 2.0 uur	2 uur	Hoorcollege lessen	13 x 2.0 uur	26 uur	Practicum lessen	14 x 1.5 uur	21 uur	<i>Wekelijks zelfstandig werken</i>	<i>8 x 30 uur</i>	<i>240 uur</i>	Boekenclub bijeenkomsten	6 x 2.0 uur	12 uur	<i>Vorbereiding boekenclub</i>	<i>6 x 2.0 uur</i>	<i>12 uur</i>	Ethiek college	3 x 2.0 uur	6 uur	Onderzoek vaardigheden training	2 x 2.0 uur	4 uur	<i>Vorbereiding mondeling</i>	<i>1 x 1.5 uur</i>	<i>1.5 uur</i>	Mondeling	1 x 0.5 uur	0.5 uur	<i>Vorbereiding challenge</i>	<i>1 x 25 uur</i>	<i>25 uur</i>	Challenge	1 x 5 uur	5 uur	<i>Werken aan de herkansing</i>	<i>1 x 39 uur</i>	<i>39 uur</i>	Herkansing	1 x 1 uur	1 uur	SLC	1 x 20 uur	20 uur	Vragen open inloop bijeenkomsten	2 x 1.5 uur	3 uur		1 x 2.0 uur	2 uur	Totaal		420 uur
Kick-off en gastcollege	1 x 2.0 uur	2 uur																																																					
Hoorcollege lessen	13 x 2.0 uur	26 uur																																																					
Practicum lessen	14 x 1.5 uur	21 uur																																																					
<i>Wekelijks zelfstandig werken</i>	<i>8 x 30 uur</i>	<i>240 uur</i>																																																					
Boekenclub bijeenkomsten	6 x 2.0 uur	12 uur																																																					
<i>Vorbereiding boekenclub</i>	<i>6 x 2.0 uur</i>	<i>12 uur</i>																																																					
Ethiek college	3 x 2.0 uur	6 uur																																																					
Onderzoek vaardigheden training	2 x 2.0 uur	4 uur																																																					
<i>Vorbereiding mondeling</i>	<i>1 x 1.5 uur</i>	<i>1.5 uur</i>																																																					
Mondeling	1 x 0.5 uur	0.5 uur																																																					
<i>Vorbereiding challenge</i>	<i>1 x 25 uur</i>	<i>25 uur</i>																																																					
Challenge	1 x 5 uur	5 uur																																																					
<i>Werken aan de herkansing</i>	<i>1 x 39 uur</i>	<i>39 uur</i>																																																					
Herkansing	1 x 1 uur	1 uur																																																					
SLC	1 x 20 uur	20 uur																																																					
Vragen open inloop bijeenkomsten	2 x 1.5 uur	3 uur																																																					
	1 x 2.0 uur	2 uur																																																					
Totaal		420 uur																																																					
Vereiste voorkennis:	<p>Voor de deelname aan deze TINLAB is het noodzakelijk dat de student volle 40 uur per week beschikbaar is. Dat houdt in, student kan iet tegelijk bezig zijn met project 3/4, project 7/8, stage of minor. Maximaal haalbare studielast is één ander vak in dezelfde periode die de student aan het inhalen is.</p>																																																						
Werkvorm:	<p>Hoorcolleges, practicumlessen, boekenclubs, aanvullende colleges en vooral veel zelfstandig werken.</p>																																																						
Toetsing:	<p>Mondelinge toets en een opleverset van groepsopdrachten en persoonlijke opdrachten. Zie voor details paragraaf Toetsing</p>																																																						
Leermiddelen:	<p>Er is verplichte literatuur. Zie paragraaf Leermiddelen voor een overzicht van te gebruiken literatuur en software.</p>																																																						
Projectdoelen:	<ol style="list-style-type: none">[Analyseren] De student kan op basis van contact met de opdrachtgever zelfstandig het probleem definiëren, onderzoeksvragen opstellen, de scope bepalen en passende kwaliteitseisen opstellen met betrekking tot code, het product zelf en het gebruik van het product.[Analyseren] De student kan op basis van literatuur- en experimenteel onderzoek onderbouwde conclusies trekken.[Manage en Control] De student is in staat om zelfstandig een project te beheren en kan waar nodig het project bijsturen op basis van de juiste onderbouwing.[Adviseren] De student kan de opdrachtgever op passende wijze adviseren over de resultaten en conclusies van het verrichte onderzoek.[Ontwerpen] De student kan op basis van de zelf opgestelde eisen relevante ontwerpen opstellen.[Realiseren] De student kan een werkend, aantoonbaar getest prototype realiseren in overeenstemming met de ontwerpen en de kwaliteitseisen.																																																						
Inhoud:	<p>In dit project maken studenten kennis met verschillende benaderingen binnen de kunstmatige intelligentie. Centrale vragen zijn: <i>Wanneer is een systeem werkelijk intelligent?</i> en <i>Hoe toon je de intelligentie van een systeem aan?</i> Studenten onderzoeken methoden om intelligente systemen te bouwen en leren afwegen</p>																																																						

Opmerkingen:

welke technieken het meest geschikt zijn in een gegeven context. Daarbij komen onder andere Machine Learning, Computational Intelligence, Evolutionary Computing, Neural Networks, Deep Learning en Data Mining aan bod. Als eindopdracht implementeren studenten een AI-controller voor een zelfrijdende auto in de racesimulatie TORCS, waarbij ze gebruik maken van een of meerdere AI-technieken om het gedrag van de auto te trainen en optimaliseren.

Voor TINLab ML geldt geen verplichte 100% aanwezigheid. Wel zijn de volgende momenten verplicht:

- De kick-off
- Gastcollege(s)
- Coachinggesprekken met docentbegeleiders
- Boekenclubbijeenkomst (gehost door de student)
- Mondelinge toetsmomenten

Hoewel aanwezigheid bij overige bijeenkomsten niet verplicht is, wordt het dringend aanbevolen om actief deel te nemen. Bij afwezigheid dient de student zich vooraf af te melden via Microsoft Teams of e-mail, zowel bij de docent als bij het eigen projectteam.

Cursusbeheerder:

E.R. van der Ven

Datum:

22 april 2025

1 Algemene omschrijving

In deze cursus ga je actief aan de slag met verschillende aspecten van Artificial Intelligence (AI). De term "Kunstmatige Intelligentie" is de laatste jaren enorm populair geworden, maar er bestaat ook veel verwarring over wat AI precies inhoudt. Is Machine Learning hetzelfde als AI? Is een neurale netwerk met drie lagen automatisch Deep Learning? In deze cursus leer je het onderscheid maken tussen de verschillende subvelden binnen AI, zoals Machine Learning, Computational Intelligence, Evolutionary Computing, Neural Networks, Deep Learning en Data Mining.

De colleges begeleiden je stap voor stap door de belangrijkste thema's binnen AI: je begint bij klassieke zoekstrategieën en planning, verdiept je in Machine Learning en neurale netwerken, en maakt kennis met evolutionaire algoritmes. Daarnaast is er aandacht voor actuele maatschappelijke kwesties, ethiek in AI en reflectie op de rol van technologie in onze samenleving.

Tijdens het project bouw je niet alleen kennis op, maar pas je die ook toe in de praktijk. Zo ontwerp je onder andere een intelligente controller voor een gesimuleerde zelfrijdende auto in TORCS. In plaats van zelf te sturen, schrijf jij de code die de auto zelfstandig leert rijden op basis van binnenkomende data en een getraind model. Je mag daarbij zelf kiezen of je werkt in C, C++, Java of Python.

Omdat er in de lessen niet genoeg tijd is om alle AI-kennis klassikaal te behandelen, neem je samen met je medestudenten deel aan een Boekenclub. Hier verdiept je je in actuele AI-artikelen, discussieer je over toepassingen en denk je na over de maatschappelijke impact van AI-technologie. Elke student is één keer verantwoordelijk voor het begeleiden van zo'n sessie. Ook lever je per bijeenkomst minstens één inhoudelijke vraag in over het besproken artikel.

Je krijgt dus niet alleen theoretische kennis, maar leert die ook direct toe te passen in een realistisch project. Tegelijkertijd werk je aan vaardigheden zoals zelfstandig onderzoeken, samenwerken, presenteren, kritisch denken en het maken van ontwerpkeuzes. De cursus is erop gericht om je optimaal voor te bereiden op AI-toepassingen in de praktijk én om je bewust te maken van de bredere impact van deze technologie.

1.1 Inleiding

Of je nu software ontwikkelt, data verwerkt of systemen ontwerpt — AI zal vrijwel zeker een rol gaan spelen in jouw toekomstig werkveld. Kunstmatige intelligentie is inmiddels verweven met allerlei beroepspraktijken: in de gezondheidszorg, logistiek, onderwijs, mobiliteit en cybersecurity worden AI-technieken gebruikt om processen te verbeteren, gedrag te voorspellen en complexe beslissingen te ondersteunen.

Voor jou als ICT-professional betekent dit dat je niet alleen moet begrijpen hoe AI-technieken werken, maar ook wanneer je ze kunt toepassen, welke valkuilen erbij horen en hoe je ethisch verantwoorde keuzes maakt. Dit vraagt om een combinatie van technische kennis, kritisch denken en ontwerpgericht werken.

Denk bijvoorbeeld aan een bedrijf dat een chatbot inzet om klantvragen af te handelen. Jij bent verantwoordelijk voor het trainen van het onderliggende model én voor het borgen van de kwaliteit van de antwoorden. Of je werkt aan een applicatie die op basis van camerabeelden automatisch verdachte situaties detecteert. Hoe zorg je ervoor dat het systeem accuraat is én geen bevooroordeelde beslissingen neemt?

Deze cursus biedt je de basis om zulke vraagstukken te begrijpen en aan te pakken. Niet als eindgebruiker van bestaande AI-oplossingen, maar als ontwerper en bouwer van slimme systemen die écht waarde toevoegen. Daarmee leg je een fundament voor een loopbaan waarin AI geen bijzaak is, maar een kernonderdeel van je vak.

1.2 Relatie met andere onderwijseenheden

Deze cursus bouwt voort op eerdere projecten, het programmeeronderwijs en een gedegen wiskundige basis, die je opdoet in eerste twee jaar aan de opleiding. Daarnaast staat het toepassen van algemene beroepsvaardigheden centraal, bijvoorbeeld via samenwerken, een onderzoekende houding en projectmanagementdocumentatie.

Binnen het TINLab-traject sluit deze cursus naadloos aan op TINLab Advanced Algorithms, maar het is ook mogelijk om vanuit TINLab Embedded Systems in te stromen.

1.3 Leermiddelen

Literatuur

-  AIMA – Russell, S., & Norvig, P. (2020). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
 - Wordt digitaal verstrekt
-  DM – Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann.
 - Te vinden op [O'Reilly platform](https://www.oareilly.com)
-  DL – Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
 - Beschikbaar op <https://www.deeplearningbook.org/>
- Diverse wetenschappelijke artikelen, lijst wordt aangereikt door docenten aan het begin van de cursus, aangevuld door keuzes gemaakt door de student over de onderwerpen van de workshops

Software

- TORCS-installatie, wordt onder begeleiding geïnstalleerd
- IDE en compiler voor zelfgekozen programmeertaal
- GIT
- Docker

2 Programma

2.1 Weekprogramma

Week nummer	Startdag van de week	Lesweek nummer	Maandag		Dinsdag		Woensdag		Donderdag		Vrijdag			
			Ochtend	Middag	Ochtend 09:30 -11:30	Middag 14:00 - 15:30	Ochtend	Middag	Ochtend 09:30 -11:30	Middag 13:00 - 15:00	Ochtend 10:00 -12:00	Middag 14:30 - 16:00		
3.10	21/Apr	1	Tweede Paasdag, HR gesloten		Kick-off met gastcollege		Practicum				Ethiek 1	Boekenclub 1 Groep 1 & Groep 2	Hoorcollege	Practicum
Meivakantie														
4.1	05/May	2	Bevrijdingsdag, HR gesloten		Hoorcollege		Practicum					Boekenclub 2 Groep 3 en 4	Hoorcollege	Practicum
4.2	12/May	3			Hoorcollege		Practicum					Boekenclub 3 Groep 5 & Groep 6	Hoorcollege	Practicum
4.3	19/May	4			Hoorcollege		Practicum				Ethiek 2	Boekenclub 4 Groep 7 & Groep 8	Hoorcollege	Practicum
4.4	26/May	5			Hoorcollege		Practicum		Hemelvaart, HR gesloten					
4.5	02/Jun	project week			Hoorcollege		Practicum					Eindmarkt JSpring conferentie	Hoorcollege	Practicum
4.6	09/Jun	6	2e pinksterdag, HR gesloten		Hoorcollege		Practicum					Boekenclub 5 Groep 9 & Groep 10	Hoorcollege	Practicum
4.7	16/Jun	7			Hoorcollege		Practicum				Ethiek 3	Boekenclub 6 Reserve		Inlooppuur
4.8	23/Jun	8					Inlooppuur						Inlooppuur Joy of Coding conferentie	OPLEVERSET INLEVEREN
4.9	30/Jun	9	Mondelingen op maandag en dinsdag ochtend. Challenge donderdag of vrijdag											
4.10	07/Jul	10	Her mondelingen op vrijdag of donderdag											

2.2 Lestypes gedurende de cursus

Hoorcollege

In deze bijeenkomsten staat de theorie centraal. Studenten luisteren, maken aantekeningen en stellen vragen. De docent biedt structuur en verdieping bij complexe onderwerpen uit de literatuur.

Practicum

Tijdens practica werken studenten op locatie aan opdrachten. Soms is er extra uitleg van de docent, soms wordt er samen naar oplossingen gezocht. Ook is er ruimte voor zelfstandig groepswork, waarbij de docent de voortgang volgt en vragen beantwoordt.

Boekenclub

De boekenclub is een interactieve bijeenkomst waarin twee wetenschappelijke artikelen besproken worden. De bijeenkomst wordt steeds geleid door **twee projectgroepen**. Elke groep kiest één artikel, verzorgt de voorbereiding én leidt het gesprek over dat artikel. Alle studenten lezen beide artikelen en leveren vooraf **minimaal één inhoudelijke vraag per artikel** in.

Per bijeenkomst geldt:

Elke projectgroep is verantwoordelijk voor:

1. **Voorafgaand aan de bijeenkomst:**
 - Het aanleveren van één artikel (in overleg met de docent).
 - Het verzamelen van vragen van medestudenten (bijv. via een Google Form).
 - Het ordenen en clusteren van de ingezonden vragen per thema.
2. **Tijdens de boekenclub:**
 - Het opstellen van een aanwezigheidslijst.
 - Een korte inhoudelijke samenvatting van het artikel geven (bijv. via een presentatie).
 - De discussie op gang brengen op basis van de ingezonden vragen.
 - De discussie begeleiden en verdiepen met vervolgvragen of stellingen.
 - Het noteren van antwoorden en belangrijke inzichten uit de discussie tbv punt 3.
3. **Na afloop van de bijeenkomst:**
 - Het opstellen van een samenvatting van de bijeenkomst. Deze bevat:
 - De kern van het artikel
 - De ingezonden vragen
 - De hoofdpunten uit de discussie
 - De samenvatting wordt toegevoegd aan de groepsrapportage.

Ethiekcolleges

In deze lessen ontdekken studenten dat ethiek in AI meer is dan gevoel of mening. Ze maken kennis met systematische benaderingen zoals gevolgenethiek en waarden-gedreven ontwerp. Aan de hand van praktijkvoorbeelden worden vragen besproken als: *Mag een AI liegen?*, *Wie is verantwoordelijk voor AI-beslissingen?*, en *Wat is eerlijkheid in algoritmes?* In de tweede bijeenkomst werken studenten onder begeleiding van de ethiekdocent aan eigen casussen en morele dilemma's uit hun opdrachten.







Onderzoeksvaardigheidstraining

In deze trainingen, die in overleg met de docent worden gepland, leren studenten hoe zij effectief literatuur zoeken, onderzoeksvragen formuleren, gericht artikelen lezen en een inleiding en samenvatting schrijven voor een eigen paper.

SLC-contactmomenten

Gedurende het blok heeft de SLC-docent individuele of groepsgewijze gesprekken met studenten. Deze momenten zijn gericht op studievoortgang, reflectie en professionele ontwikkeling.

2.3 Het programma voor de hoorcolleges

College	Literatuur/huiswerk	Lesinhoud	Producten
1 – AI vroeger en vandaag	 AIMA Hoofdstuk 1 (zelfstudie)	 Introductie AI, oriëntatie op opdrachten  Gastcollege	
2 – Agents & Neurale Netwerken deel 1	 AIMA Hoofdstuk 2 en Hoofdstuk 18	 Wat is een agent? Hoe leert een perceptron?	 Start Opdracht 1

💡 Start opdracht 1: netwerk zonder libraries		
3 – Zoeken in toestandsruimten	📘 AIMA Hoofdstuk 3	🎯 BFS, DFS, UCS, probleemoplossing 💡 Navigatieproblemen en AI-planning
4 – Heuristische Zoekstrategieën & Optimalisatie	📘 AIMA Hoofdstuk 4	🎯 A*, heuristiekontwerp, planning 💡 Zoekoptimalisatie voor agents
5 – Adversarial search & spelstrategie	📘 AIMA Hoofdstuk 5	🎯 Minimax, alpha-beta pruning 💡 Strategisch denken door agents
6 – Neurale Netwerken deel 2	📘 AIMA Hoofdstuk 19 📘 DL Hoofdstuk 9 📘 DL Hoofdstuk 6 (verdieping toepassing backpropagation)	🎯 Backpropagation, regularisatie, convolutional networks 💡 Aanvullend: introductie CNN en toepassing in beeldherkenning 💡 Reflectie op opdracht 1
7 – Data pipeline & clusteranalyse	📘 AIMA Hoofdstuk 20 📘 DM Hoofdstuk 3 en Hoofdstuk 10	🎯 Voorbewerking data, clustering 💡 Toepasbaar op TORCS-sensordata
8 – Evolutionaire algoritmes & creatieve AI	📘 AIMA Hoofdstuk 4.2 📘 Aanvullend materiaal	🎯 Genetische algoritmes, creatieve toepassingen 💡 Start opdracht 2: melodiegeneratie
9 – Reinforcement learning – basis	📘 AIMA Hoofdstuk 21	🎯 Q – learning, rewards, TORCS-toepassing 💡 Besluitvorming in dynamische omgeving
10 – TORCS – agent design & strategie	📘 AIMA Hoofdstuk 11	🎯 Agentgedrag, state design, observatie 💡 Ontwerp voor opdracht 3
11 – Projectmoment 1		🎯 Begeleiding opdracht 2 en eindopdracht 💡 Peerfeedback en praktische begeleiding
12 – Ethiek, veiligheid & verantwoord AI	📘 AIMA Hoofdstuk 26 en Hoofdstuk 27	🎯 Bias, explainability, risico's 💡 Reflectie op eigen keuzes in opdrachten
13 – Projectmoment 2		🎯 Finetunen TORCS agent 💡 Groep coaching en optimalisatie
14 – Verdieping / herhaling / actuele AI – thema's	📘 Optioneel / aanvullend	🎯 Flexibel: herhaling clustering, LLM, genAI 💡 Verbinden van concepten en actualiteiten

3 Aanwezigheid

Aanwezigheid is bij deze cursus verplicht, zoals aangegeven in de curriculumtabel. Hieronder lees je bij welke lessen je verplicht aanwezig moet zijn en waarom:

College 1 – Kick-off en gastcollege

Je aanwezigheid is verplicht vanwege de officiële aftrap van de cursus en het ontvangen van belangrijke informatie. Daarnaast is er een gastcollege dat essentieel is voor je verdere begrip van de lesstof.

College 2 – Introductie opdracht 1

Tijdens deze les wordt persoonlijke opdracht 1 in detail toegelicht. Om goed van start te gaan en geen cruciale informatie te missen, is aanwezigheid hier noodzakelijk.

College 8 – Introductie opdracht 2

Deze les vormt het startpunt van opdracht 2. Ook hier is aanwezigheid verplicht vanwege de uitgebreide uitleg en het klassikaal opzetten van de opdracht.

College 10 – Introductie eindopdracht

Dit college behandelt de kick-off van de eindopdracht. Omdat je hier alle basisinformatie ontvangt voor het laatste projectonderdeel, is aanwezigheid verplicht.

Colleges 11 en 13 – Groepsbegeleiding / Coaching

Deze bijeenkomsten zijn bedoeld om de voortgang van je opdrachten te bespreken en feedback te krijgen van zowel medestudenten als je docentbegeleiders. Verplichte aanwezigheid is nodig om optimaal gebruik te maken van deze begeleiding.

College 12 – Ethiek

Bij dit college is aanwezigheid verplicht vanwege de bespreking van maatschappelijke en ethische vraagstukken in AI. Dit is een fundamenteel onderdeel van het TINLAB dat je niet mag missen.

Boekenclubbijeenkomsten

Verder zijn er gedurende het project wekelijkse boekenclubbijeenkomsten, bedoeld als activerend en verdiepend leermoment. De boekenclub is verplicht voor de sessie(s) waarvoor je als gespreksleider bent ingedeeld. Alle andere bijeenkomsten worden sterk aanbevolen voor een goede verdieping in de lesstof.

Mocht je meer dan één keer zonder geldige reden afwezig zijn bij een verplichte bijeenkomst, dan kun je geen beoordeling krijgen voor dit vak en heb je dit schooljaar geen recht op herkansing.

4 Toetsing en beoordeling

4.1 Opdrachten

4.1.1 Persoonlijke opdracht 1 – Symboolherkenner

In deze opdracht bouw je zelf een neurale netwerk dat op basis van een eenvoudige 3×3 zwart-wit input onderscheid kan maken tussen kruisjes ('X') en rondjes ('O'). Je werkt zonder externe machine learning libraries, om zelf inzicht op te bouwen in de werking van neurale netwerken en het leerproces.

Je ontwikkelt het netwerk in drie fasen:

- Een eenvoudige implementatie zonder verborgen lagen.
- Een uitbreiding met matrixoperaties en een verborgen laag.
- (Optioneel) Een verdiepingsfase met gradient-based learning (steepest descent).

Je test het netwerk op imperfecte symbolen en onderzoekt het effect van ontwerpkeuzes op prestaties. Als eindproduct lever je een werkend programma in én een kort wetenschappelijk artikel waarin je jouw aanpak, keuzes en resultaten toelicht.

De volledige opdrachtbeschrijving is te vinden in [Bijlage B](#).

4.1.2 Persoonlijke opdracht 2 – Melody Maker

In deze opdracht bouw je zelf een genetisch algoritme (GA) dat korte muzikale fragmenten (building blocks) combineert tot smaakvolle composities. Je ontwikkelt het algoritme in twee fasen:

- Fase 1: Basale GA met populatiebeheer, selectie, één-punt of uniform crossover en eenvoudige mutatie-operators.
- Fase 2: Uitbreiding van de fitness-functie met muzikale heuristieken (maatlengte, akkoordprogressies, toonsoort, structuur) en verfijnde recombinate- en mutatiestrategieën.

Je evalueert de robuustheid van je algoritme en luistert naar bijzondere of verrassende composities. Als eindproduct lever je een werkend programma met een genetisch algoritme én een kort technisch artikel waarin je je aanpak, parameters en resultaten toelicht.

De volledige opdrachtbeschrijving is te vinden in [Bijlage C](#).

4.1.3 Ethiekopdracht: We kunnen het maken... maar kunnen we het écht maken?

In deze groepsopdracht analyseren studenten een bestaande of toekomstige AI-toepassing die ethische en maatschappelijke vragen oproept. Centraal staat de vraag of een technologie die technisch mogelijk is, ook wenselijk en verantwoord is om te bouwen. De opdracht resulteert in een gezamenlijk reflectieproduct (bijv. essay, podcast, poster).

De volledige opdrachtbeschrijving is te vinden in [Bijlage D](#).

4.1.4 Eindopdracht – TORCS

Ontwikkel een AI-controller die zelfstandig in TORCS rijdt, waarbij je de door de docent aangeleverde (of zelf gegenereerde) racesensor-data benut om relevante features te selecteren, een neurale netwerk te trainen en je model te integreren in de simulator. De auto schakelt automatisch en schade door botsingen telt niet mee: beoordeeld worden alleen rondetijd en stabiliteit.

De volledige opdrachtbeschrijving is te vinden in [Bijlage E](#).

4.2 Procedure

Ingangseisen toets

Om te mogen deelnemen aan de mondelinge toets, moet het volledige opleverset tijdig en compleet op Brightspace zijn ingeleverd. 'Tijdig' betekent conform het weekprogramma zoals beschreven in [paragraaf 2.1](#). 'Compleet' houdt in dat jouw opleverset bestaat uit twee delen: een persoonlijk gedeelte en een groeps gedeelte.

Persoonlijk gedeelte:

- **Persoonlijke evaluatie op inhoud:** Reflectie op het project en de behaalde leerdoelen.
- **Persoonlijke evaluatie op samenwerking:** Reflectie op de eigen bijdrage en de onderlinge samenwerking in de groep.
- **Verslag over [opdracht 1](#):** Een wetenschappelijke artikel met een beschrijving van de resultaten en de aanpak van opdracht 1.
- **Verslag over [opdracht 2](#):** Een technisch artikel met een beschrijving van de resultaten en de aanpak van opdracht 2.

Groeps gedeelte:

- **Verslag van de gehoste Boekenclub:** Een verslag waarin de uitkomsten en discussiepunten uit de boekenclub bijeenkomst worden vastgelegd.
- **Ethiekopdracht:** Een stuk dat een belangrijk ethisch vraagstuk binnen AI behandelt.
- **Ontwerp- en ontwikkelverslag van de [groepseindopdracht](#):** Een verslag waarin het ontwerp en de ontwikkelfase van de eindopdracht worden toegelicht.

Wanneer alle bovenstaande onderdelen volledig en volgens de gestelde deadlines zijn ingediend, komt het hele groep in aanmerking om deel te nemen aan de mondelinge toetsing. Mocht er een onderdeel ontbreken of te laat worden ingeleverd, dan kan dit gevolgen hebben voor de beoordeling van het vak (zie PBS, [bijlage A](#)).

Toetsduur

Elke modeling duurt 30 minuten. In ongeveer 15–20 minuten presenteert de projectgroep haar project, waarna de resterende tijd wordt ingezet voor vragen.

Beoordeling

De toetsing en de herkansing worden uitgevoerd volgens de Projectbeoordelingssystematiek (PBS) 3.3.1. Zie [bijlage A](#).

De beoordeling van het vak vindt plaats op basis van zowel de ingeleverde opleverset als het mondelinge toetsmoment.

Indien alle onderdelen tijdig en correct zijn ingeleverd en tijdens het mondelinge toetsmoment een gestructureerde presentatie en interactie hebben plaatsgevonden, wordt een cijfer toegekend op een schaal van 6 t/m 10.

Als onderdelen ontbreken of te laat zijn ingeleverd, wordt er een beoordeling NVL (niet voldaan) gegeven. Dit leidt tot een herkansing zonder het voordeel van inhoudelijke feedback op de oorspronkelijke inzending.

Beoordelingscriteria en totstandkomingcijfer

De precieze criteria voor de beoordeling en de manier waarop de eindscore tot stand komt, zijn uitgebreid beschreven in het Projectbeoordelingssysteem (PBS). In het bijzonder in de paragraaf “Bepaling van de hoogte van het cijfer bij een voldoende” in [bijlage A](#). Daar vind je een gedetailleerde uitwerking van de factoren die meespelen in de beoordeling van de opleverset en het mondelinge toetsmoment, zoals de complexiteit van de opdracht, de kwaliteit van de opleverset en de mate van zelfstandigheid van de student(en).

Feedback

Aan het einde van de dag waarop het mondelinge toetsmoment plaatsvindt, krijgen de studenten te horen of zij een voldoende hebben behaald. Groepen die geen voldoende krijgen, ontvangen direct gerichte verbeterpunten en suggesties voor eventuele aanpassingen in een herkansingstraject.

Het exacte cijfer wordt bekendgemaakt tijdens de eindchallenge, waar de resultaten officieel worden gepresenteerd.

4.3 Herkansing

Volgens het Projectbeoordelingssysteem ([PBS](#)) geldt dat, wanneer een student geen voldoende haalt voor deze cursus, in de schriftelijke beoordeling een aantekening wordt opgenomen over de herkansing (reparatieopdracht). De inhoud van deze herkansing is afhankelijk van het resultaat en de feedback tijdens de eerste toetsgelegenheid.

De herkansing vindt plaats binnen het lopende studiejaar, en de exacte deadlines worden vermeld in de reparatieopdracht. Als het project na deze herkansing of na het verstrijken van de deadline voor de herkansing nog steeds niet behaald is, moet de student het project opnieuw in zijn geheel uitvoeren tijdens de volgende reguliere gelegenheid. In dat geval moet men zich opnieuw inschrijven via de gebruikelijke procedure.

5 Plagiaat

Onder plagiaat wordt verstaan: het overnemen van teksten, afbeeldingen of code zonder correcte bronvermelding. Je mag dus wel werk van anderen gebruiken in je eigen werk, als je maar duidelijk maakt dat het niet je eigen werk is en een correcte bronvermelding geeft. Plagiaat is een vorm van fraude. Onder

fraude wordt eveneens verstaan als de student een andere student in de gelegenheid stelt onderdelen van zijn of haar werk over te nemen.

Het plegen van fraude en plagiaat is niet toegestaan. Ingeleverd werk kan gescand worden op plagiaat. Bij vermoeden van fraude en/of plagiaat wordt de Examencommissie geïnformeerd. Voor meer informatie zie de Hogeschoolgids bijlage 4, hoofdstuk 9.

6 Changelog

Versie	Wijzigingen
2018 / 2019	Eerste versie
2019 / 2020	PBS 3.1.1 Wijziging van ontvankelijkheid van de TINLab lessen.
2019 / 2020	Feedback van CC verwerkt
2020 / 2021	Aangepaste opdrachten, weekplanning, opleverset. PBS 3.3.1 Flankerend onderwijs bijgewerkt Aanwezigheidsplicht aangepast
2022 / 2023	Aangepaste weekplanning, hoorcollege onderwerpen.
2023 / 2024	Aangepaste weekplanning, toegevoegde gastcollege, verschoven hoorcollege onderwerpen.
2024 / 2025	PBS 3.7, nieuwe aantal ect, i.v.m. semester 6, aangepaste weekindeling voor collegejaar 24-25, opdrachten opgenomen in de modulewijzer

Bijlage A: Projectbeoordelingssystematiek 3.7

Inleiding

In dit document wordt de voor alle cursussen binnen de praktijklijn (projecten, stage, TINLab en afstuderen) geldende toetsvormen en -producten beschreven, alsmede de wijze van beoordeling en de algemeen geldende regels voor herkansingen.

Toetsvorm

De officiële toetsvorm voor alle cursussen binnen de praktijklijn is een combinatie van 'Opdracht' (O) en 'Mondelinge Toets' (M). Hiermee wordt bedoeld dat de toetsing altijd bestaat uit het inleveren van toetsproducten en uit een gepland mondeling toetsmoment.

Becijfering

Bij de becijfering van de cursus kan er gekozen worden uit de volgende opties:

- een geheel cijfer van 6 t/m 10,
- een ND (niet deelgenomen)
- een NVL (niet voldaan)

Toetsproducten

Iedere cursus kent twee toetsproducten opgeleverd door de student:

1. Een **opleverset** met direct bewijsmateriaal van het uitgevoerde werk. Hiermee wordt het beroepsproduct aangetoond. In de cursushandleiding van de betreffende cursus wordt gespecificeerd waar het beroepsproduct uit moet bestaan. Dit zijn onder andere documenten, rapporten, verslagen, programmacode, (technische) tekeningen, filmpjes enz.
2. Een **verantwoording**: een vastlegging van hoe de cursus is uitgevoerd, zoals projectplanning, evaluaties, voortgangsverslagen, reflectie op de persoonlijke ontwikkeling. Hiermee wordt het proces en de persoonlijke groei van de student beschreven.

Beide toetsproducten worden ingeleverd via Brightspace (jaar 1, 2 en TINLabs) of Praktijklink (stage en afstuderen). Alle documenten moeten geüpload worden; alleen een link naar een externe locatie is niet voldoende. In de cursushandleiding wordt beschreven in welke vorm de toetsproducten moeten worden ingeleverd.

Mondeling toetsmoment

In de weken rondom het inlevermoment van de toetsproducten wordt er ook een mondeling toetsmoment gepland met de student of projectgroep en beoordelaar(s). Tijdens dit mondeling toetsmoment kan er aandacht besteed worden aan de samenwerking binnen de groep, kan er een demonstratie van het beroepsproduct gegeven worden, kunnen onduidelijkheden uit de opleverset bevraagd worden, verdediging van het afstudeerwerk, enz. Het doel van de mondelinge toetsing wordt nader beschreven in de cursushandleiding.

Beoordeling

De beoordeling vindt plaats door tenminste één beoordelaar, en op eindniveau van de opleiding door tenminste twee beoordelaars. Tevens vindt er kalibratie plaats tussen de beoordelaars. De procedure van de beoordeling verschilt per cursus en staat toegelicht in de cursushandleiding. Voorafgaand aan de beoordeling kunnen de toetsproducten gecontroleerd worden op *volledigheid*. In de cursushandleiding staat in dat geval beschreven uit welke onderdelen de toetsproducten minimaal moeten bestaan. Ook moeten de toetsproducten ingeleverd worden in het *juiste bestandsformaat*. Zie voor het lijstje met toegestane bestandsformaten onderaan deze tekst. Bij onvolledigheid of verkeerde bestandsformaten is het ingeleverde werk *niet ontvankelijk* en wordt het niet inhoudelijk beoordeeld. In dit geval wordt de beoordeling NVL (niet voldaan) gegeven en moet het werk opnieuw ingeleverd worden voor de herkansing, zonder het voordeel van inhoudelijke feedback bij de eerste gelegenheid. Tevens is

de student in dit geval uitgesloten van deelname aan het mondeling toetsmoment als dat later plaatsvindt dan het inlevermoment van de toetsproducten.

Bij cursussen waarin in groepen gewerkt wordt krijgen de groepsleden in principe een gemeenschappelijke beoordeling van de docent(en), met dien verstande dat de docent(en) één of meerdere groepsleden een andere beoordeling mogen geven indien daarvoor valide redenen bestaan. In alle gevallen geldt dat deze worden beargumenteerd aan de hand van de projectdoelen, dan wel de aanvullende aspecten. **De beoordeling en argumentatie zijn altijd schriftelijk en altijd toegankelijk voor de betrokken student(en).**

Een **voldoende** beoordeling wordt behaald dan en slechts dan als

- alle projectdoelen, zoals aangegeven in de cursushandleiding, behaald zijn, én
- de toetsproducten volledig zijn.

Bepaling van de hoogte van het cijfer bij een voldoende

Als aan de eisen voor een voldoende voldaan is, wordt een geheel cijfer toegekend van 6 t/m 10, op basis van één of meer van de volgende aspecten. Deze aspecten kunnen per cursus nader gespecificeerd zijn in de cursushandleiding. Per aspect kunnen één of meer punten toegekend worden.

- **Complexiteit van de omgeving / context** van het project of de opdracht. Hiermee wordt bedoeld dat de omgeving/context het standaardniveau overstijgt. Denk aan de plek waar de opdracht of het project uitgevoerd wordt, zoals een multidisciplinair project bij een bedrijf of zorginstelling of onderzoeksinstituut. Denk ook aan diversiteit van stakeholders.
- **Complexiteit van de gebruikte technieken.** Hiermee wordt bedoeld of de student of groep technieken gebruikt heeft die het standaardcurriculum overstijgen of zeer innovatief zijn. Denk ook aan complexere functionaliteiten ten opzichte van de beschikbare tijd.
- **Kwaliteit van de opleverset.** Hiermee wordt kwaliteit bedoeld die de standaard overstijgt; denk aan de bruikbaarheid van het opgeleverde werk en hoe het aan professionele eisen voldoet die gangbaar zijn in het werkveld.
- **Zelfwerkzaamheid van de student(en).** Hier wordt gekeken naar de zelfstandigheid, zelfredzaamheid en het zelfsturend vermogen van de student en/of de projectgroep.
- **Visie op het werkveld.** Hierbij wordt gekeken of de student een onderbouwde visie op ontwikkelingen en/of ethische aspecten binnen het werkveld ontwikkeld heeft.

Bij het waarderen van deze aspecten wordt altijd een schriftelijke argumentatie gegeven per aspect, waarbij het toegekende aantal extra punten toegelicht wordt.

Beoordeling van een onvoldoende

Als niet aan de eisen voor een voldoende voldaan is (ontbrekende en/of onvoldoende toetsproducten), zal de beoordeling NVL (niet voldaan) gegeven worden. Een ND (niet deelgenomen) wordt toegekend indien er geen toetsproducten zijn ingeleverd.

Herkansing

Indien een student geen voldoende haalt voor een cursus dan wordt bij de schriftelijke beoordeling een aantekening opgenomen over de herkansing (reparatieopdracht). Hetgeen herkanst moet worden is afhankelijk van het resultaat (en de feedback) van de eerste gelegenheid.

Binnen het lopende studiejaar moet er herkanst kunnen worden. De deadline voor de herkansingen worden per cursus specifiek benoemd in de cursushandleiding en/of de reparatieopdracht. Als het project na de herkansing of de deadline van de herkansing nog niet is behaald, dient het gehele project opnieuw gedaan te worden bij een volgende gelegenheid. Men dient zich dan voor aanvang van het project in te schrijven via Osiris en beschikbaarheid van het project in Praktijklink aan te vragen bij de coördinator.

Bestandtypes

De volgende bestandsformaten zijn toegestaan in de opleverset:

- **Opleverset:** .ZIP (geen .RAR of .7zip)
- **Verslagen/Ontwerpen:** .PDF of .DOCX (geen .ODT)
- **Code:** in de juiste extensie van de programmeertaal (.TXT/.DOCX zijn niet toegestaan)
- **Afbeeldingen:** .PDF, .PNG of .JPEG
- **Video's:** .MP4 (geen overdreven hoge resolutie; max. 100 MB voor alle video's samen)
- **Planningen:** .XLSX, .PDF, indien je een link naar een Trello meestuurt zorg dan dat je deze in een .TXT-bestand plaatst en dat de Trello of andere online omgeving toegankelijk is voor mensen van buitenaf.

Bijlage B: Persoonlijke opdracht 1 – Symbool-herkenner

Doel

Zelf een eenvoudig neurale netwerk bouwen zonder gebruik van externe machine learning libraries, om inzicht te krijgen in de basisprincipes van neurale netwerken en het leerproces.

Leerdoelen

Praktisch inzicht krijgen in wat nu werkelijk essentieel is voor een eenvoudig neurale netwerk en wat "bijkomende" optimalisaties zijn (hoe belangrijk ook).

Vertrouwen krijgen in het feit dat neurale netwerken niet alleen uit de hoge hoed van Google worden getoverd, maar dat je die met gebruik van veel gezond verstand en weinig wiskunde gewoon zelf kunt maken. Werken met activatiefuncties, gewichten, fouten, backpropagation en kostenfuncties.

Opdrachtomschrijving

Ontwerp en bouw een neurale netwerk dat kruisjes ('X') van rondjes ('O') kan onderscheiden op basis van een binaire invoer van 3x3 pixels. De input is zwart-wit: 1 = zwart (actief), 0 = wit (achtergrond). Een verborgen laag is in deze eerste versie nog niet nodig. Je gebruikt geen externe machine learning libraries; standaard programmeerfunctionaliteit is voldoende. De opdracht mag in een programmeertaal naar keuze worden uitgevoerd, mits je je houdt aan de beperkingen en zelf de netwerkstructuur implementeert.

Fase 1 – Perceptron zonder verborgen lagen

1. Implementeer een netwerk met een inputlaag en een outputlaag, zonder verborgen lagen.
2. Begin met een directe, intuïtieve implementatie met behulp van zelfontworpen klassen Node en Link.
3. Gebruik de softmax-functie in de outputlaag
4. Maak gebruik van de backpropagation-algoritme startend met willekeurige variaties van de weight factoren. Varieer één weight factor per leercyclus over de hele trainingset en behoud alleen de wijziging die gemiddeld over deze cyclus het beste scoort.
5. Gebruik Mean Squared Error (MSE) als cost function.
6. Test ook met 'onvolmaakte' voorbeelden (bijv. kruisjes met ontbrekend lijntje of rondjes met een gat). Probeer te achterhalen hoe robuust je netwerk is voor imperfecte of afwijkende input.

Fase 2 – Matrixgebaseerde aanpak (verdieping)

1. Bouw een versie met vectoren en matrices dat verder in principe hetzelfde doet als versie uit fase 1.
2. Voeg een hidden layer toe en experimenteer met het aantal nodes in de hidden layer.
3. Observeer het effect ervan op de classificatienauwkeurigheid(of foutpercentatge), neem dit op in je verslag.
4. Gebruik sigmoid activatie functie in je hidden layers.
5. Gebruik in eerste instantie zelf gemaakte matrix vector operaties.
6. Stap in versie 2.2 over op library's als DeepJavaLibrary (https://d2l.djl.ai/chapter_preliminaries/linear-algebra.html), Vectorz (<https://github.com/mikera/vectorz>) of numpy voor matrix-bewerkingen.

Fase 3 – (Optioneel) Steepest descent

1. Stap ten slotte over op het gebruik van steepest descent (de gradienten-methode).
2. Gebruik hiervoor de sigmoid-functie en de bijbehorende analytische afgeleide, namelijk:
 $s(x) * (1 - s(x))$.
3. Observeer het verloop van de cost function als functie van het aantal iteraties waarbij de volledige trainingset wordt doorlopen.
4. Formeel begrip van de wiskunde achter de afleiding is niet vereist; je mag hiervoor verwijzen naar literatuur op het web.
5. Deze opdracht is facultatief en dient als verdieping.

Randvoorwaarden

- Gebruik geen externe AI/ML-libraries (zoals TensorFlow, PyTorch, Keras, Scikit-learn, DL4J, etc.).
- Gebruik wel standaard wiskundige en programmeerfunctionaliteit.
- Later mag je wiskundige libraries gebruiken.
- Je moet je code (met name fase 1 & 2) kunnen uitleggen in eigen woorden.

Inputdata

De invoer bestaat uit 3x3 binaire rasterpatronen die een 'X' of een 'O' voorstellen. Je kunt een trainingset en testset zelf ontwerpen of overnemen uit het opgegeven voorbeeldbestand – inputdata.txt.

Inputdimensie: 9

Outputdimensie: 2

Outputcodering: {'O': (1, 0), 'X': (0, 1)}

De waarden zijn binair: 1 = zwart, 0 = wit.

Test ook met onvolmaakte voorbeelden, zoals kruisjes met een ontbrekende lijn of rondjes met een gat, om te evalueren hoe robuust je netwerk is bij afwijkende invoer.

Eindproduct

- Je demonstreert en levert een werkend programma in.
- Je schrijft een wetenschappelijk artikel (richtlijn: 2 tot 4 pagina's), met de structuur zoals voorgesteld in [bijlage F: Structuur van een wetenschappelijk artikel](#).

Bijlage C: Persoonlijke opdracht 2 – Melody Maker

Korte uitleg over genetische algoritmen (GAs)

Genetische algoritmen zijn zoek- en optimalisatietechnieken geïnspireerd door natuurlijke selectie. Je werkt met een populatie van mogelijke oplossingen (chromosomen), beoordeelt ze met een fitness-functie, selecteert de beste individuen, combineert hun 'genen' via crossover, en introduceert variatie met mutatie. Door meerdere generaties evolueert de populatie naar steeds betere oplossingen.

Doel

Ontwikkelen van een eigen genetisch algoritme voor muzikale compositie, waarbij je inzicht krijgt in de werking en samenhang van de kernelementen van GAs: building blocks, mutatie, recombinitie, selectiedruk en diversiteit.

Leerdoelen

Praktische ervaring met de bouwstenen van een genetisch algoritme:

- Definieren en manipuleren van building blocks
- Ontwerpen van mutatie- en recombinitie-operators
- Toepassen van selectiedruk via een fitness-functie
- Bewaken van diversiteit in de populatie
- Integreren van heuristieken om muzikale kwaliteit te sturen

Opdrachtomschrijving

Maak in de door jou gekozen programmeertaal een GA dat korte muzikale fragmenten ("building blocks") combineert tot een smakvolle compositie. Jouw rol als omgeving: bepaal zelf welke composities een hoge fitness krijgen door ze een score te geven.

1. Building blocks
 - Vooraf gedefinieerde clichés, d.w.z. bepaalde clichématige muziekfragmenten die je vaak in populaire songs tegenkomt (zie Songfestival als treffend voorbeeld)
2. Mutatie & Recombinatie
 - Ontwerp mutatie operatoren op individuele maten of toonladders.
 - Implementeer recombinitie (één punt of uniform crossover) van goed scorende fragmenten.
3. Selectie
 - Gebruik tournamentsselectie of roulette wielselectie om ouders te kiezen.
 - Behoud fragmenten die in meerdere goed scorende composities voorkomen.
4. Diversiteit
 - Zorg dat je basisverzameling van de buildingblocks breed genoeg blijft.
5. Heuristiek
 - Voorzie de fitness-functie van muzikale heuristieken (dat wil zeggen voorkennis over wat zoal voor velen goed klinkt), zoals: 8 of 16 maten van 4 tellen, afwisseling van coupletten en refreinen, II–V–I-progressies en toonsoorten beperkt tot de witte toetsen met een afsluiting op

Fase 1 – Basale GA

1. Implementeer een eenvoudige populatie van verzamelingen building blocks.
2. Definieer een fitness-functie waarin je handmatig scores toekent.
3. Voer selectie, crossover en mutatie uit.

4. Documenteer in je verslag de werking en afstemming van de parameters (populatiegrootte, mutatiekans, crossover-percentage)

Fase 2 – Heuristieken & Geavanceerde operators

1. Breid je fitness-functie uit met geautomatiseerde heuristieken (lengte, akkoordprogressies, maatvoering).
2. Experimenteer met verschillende recombinate- en mutatiestrategieën (bijv. pitch-mutatie vs. ritme-mutatie).
3. Analyseer de invloed op zowel de kwaliteit van de composities als de populatiediversiteit.

Randvoorwaarden

- Gebruik geen kant en klare AI/ML – frameworks, standaard programmeerfunctionaliteit volstaat.
- Audio-bibliotheken mogen voor afspelen of schrijven van WAV/MP3-bestanden.
- Je moet je code uitleggen in eigen woorden.

Inputdata

In de meegeleverde map **music.zip** vind je alles wat je nodig hebt om je gegenereerde notenreeksen om te zetten in audio en een voorbeeld te beluisteren:

- **bach.py**
Een voorbeeldscript waarin een tuple **bach** met twee tracks van (noot, duur) -paren staat. Het toont hoe je via onderstaande code tijdelijke WAV-bestanden maakt en mixt tot één **song.wav**.

```
from muser import Muser
muser = Muser()
muser.generate(bach)
```

- **muser.py**
Bevat de **Muser** – klasse met de methode **generate(song)**, die per track **track_XXX.wav** – bestanden genereert en deze mixt tot **song.wav**.
- **song.wav**
Een kant-en-klaar voorbeeld van de output van **bach.py**. Luister hiernaar om een referentie te hebben voor de klank, timing en mix van je eigen GA – composities.

Als je de Python wave file generator uit **music.zip** wilt gebruiken vanuit een andere taal, is 't het eenvoudigst om de broncode van muziekstukken zoals **bach.py** in z'n geheel vanuit die andere taal te genereren. JSON over sockets mag ook.

Nodig is ook de Python library **tomita**, te installeren met:

```
python -m pip install tomita
```

Op sommige systemen in plaats van **python**: **python3**, **python3.9**, **python39** of **py39**.

Eindproduct

- Je demonstreert en levert een werkend programma dat jouw GA uitvoert en meerdere composities genereert in.
- Je schrijft een technisch artikel (richtlijn: 2 tot 4 pagina's), met de structuur zoals voorgesteld in [bijlage G: Structuur van een technisch artikel](#).

Bijlage D: Groepsopdracht – ethische dilemma

Doel van de opdracht

In deze opdracht verkennen jullie als projectgroep een bestaande of denkbare AI-toepassing die morele, maatschappelijke of technologische grenzen opzoekt. Denk bijvoorbeeld aan AI in rechtspraak, zorgrobots, creatieve AI, gezichtsherkenning of besluitvorming over mensenlevens. Jullie analyseren de toepassing aan de hand van ethische kaders en stellen samen de vraag:

We kunnen dit misschien maken... maar kunnen we dit echt maken – verantwoord, wenselijk, als mens en maatschappij?

Centrale vragen

- Wat gebeurt er als deze technologie werkelijkheid wordt?
- Welke waarden worden geraakt of onder druk gezet?
- Wie draagt verantwoordelijkheid?
- Hoe verhouden maakbaarheid en moraliteit zich tot elkaar?

Leerdoelen

- Moreel leren redeneren over technologie in ontwikkeling
- Werken met ethische kaders zoals gevolgenethiek, plichtenethiek, waardenethiek en morele verbeelding
- Samen reflecteren op de verhouding tussen kunnen, mogen en willen
- Leren samenwerken aan een kritisch en publiekgericht product

Opdrachtproduct

Elke projectgroep levert één gezamenlijk product aan. Dit mag een essay zijn, maar ook een andere vorm aannemen, zoals een podcast, video, interactieve poster, fictieve use-case of position paper.

Richtlijn:

- Tekst: ca. 1000 woorden
- Alternatieve vorm: 5–7 minuten lengte

Verplicht in het product:

- Beschrijving van de gekozen AI-toepassing of toekomstscenario
- Analyse van een concreet ethisch dilemma of spanningsveld
- Toepassing van minimaal twee ethische denkkaders
- Formulering van een gezamenlijke en onderbouwde positie als projectgroep

Bijlage E: Groepsopdracht – Autonome TORCS driver

Doel

Ontwikkelen van een eigen genetisch algoritme voor muzikale compositie, waarbij je inzicht krijgt in de werking en samenhang van de kernelementen van GAs: building blocks, mutatie, recombinitie, selectiedruk en diversiteit. Ontwikkel een autonome AI-controller voor de racesimulator TORCS die, zonder zelf te schakelen en waarbij eventuele botsingen (schade) niet in de beoordeling meetellen, snel en consistent rondes kan rijden.

Leerdoelen

- Inzicht krijgen in end-to-end ontwikkeling van een AI-agent in een complexe simulatiewereld.
- Zelfstandig relevante sensordata selecteren via data-analyse.
- Ontwerpen, trainen en fine-tunen van een neurale netwerk voor real-time besturing.
- Integratie van een getraind model binnen een externe simulator via een programmeerinterface.
- Opzetten en uitvoeren van systematische tests en resultaatanalyse.

Opdrachtomschrijving

Je krijgt van de docent een set racesensor-data (positie, snelheid, afstand tot bocht, etc.). Je mag deze aanvullen met zelf gesimuleerde data. Het schakelen van de auto wordt volledig door TORCS afgehandeld: je agent stuurt alleen stuurberekening, gaskracht en remsignaal aan. Schade door botsingen negeer je bij de evaluatie van prestaties—alleen snelheid en stabiliteit bepalen je score.

Fase 1 – Data-analyse & Featureselectie

1. Verken en analyseer de aangeleverde sensorlogbestanden met beschrijvende statistieken en visualisaties.
2. Identificeer welke features (bijv. hoek naar de volgende bocht, relatieve snelheid, track-offset) de grootste correlatie hebben met ronden-tijd en stabiliteit.
3. Normaliseer je gekozen features (scaling, outlier-affhandeling).
4. Optioneel: genereer of simuleer aanvullende data om under- of overrepresentatie in de trainingsset te voorkomen.

Fase 2 – Modelimplementatie & Training

1. Stel een dataset samen van inputvectors (sensordata) en bijbehorende stuur-, gas- en remcommando's uit voorbeeld-runs.
2. Bouw en train een neurale netwerk (of hybride model) dat op basis van de gekozen features je agent bestuurt:
3. Experimenteer met aantal lagen, neurons en activatiefuncties.
4. Pas loss-functies en optimizerinstellingen aan (bijv. MSE, Adam).
5. Evalueer met cross-validation en bewaak over-/underfitting via learning-curves.

Fase 3 – Integratie, Testen & Evaluatie in TORCS

1. Integreer je getrainde model als autonome driver: gebruik de TORCS-API of sockets om je netwerk real-time aan te roepen.
2. Definieer minimaal drie testscenario's (bijv. rechte stukken, snelle bochtencombinaties, variërende startposities) inclusief acceptatiecriteria:

- Rondetijd (gemiddeld en minimaal)
 - Stabiliteit (standaarddeviatie van de track-offset)
 - Botsingsvrije rit (botsingen neger je, maar registreer ze voor diagnostiek)
3. Voer per scenario ten minste vijf runs uit en verzamel de statistieken.
 4. Analyseer de testresultaten: identificeer zwakke punten (bijv. traag bochtgedrag) en pas je model of preprocessing aan.
 5. Herhaal test-tweak-test totdat je agent voldoet aan de gestelde rondetijd- en stabiliteits-criteria

Randvoorwaarden

- Geen schakellogica: TORCS regelt versnellingsbak en koppeling; je agent stuurt enkel gas, rem en stuurhoek.
- Schade negeren: Bij evaluatie telt alleen tijd en trajectafwijking; botsingen hebben geen negatieve score.

Eindproduct

- Executable TORCS-controller: een script of programma dat jouw AI-agent laadt en zelfstandig laat rijden.
- Volledig eindverslag volgens de structuur uit [Bijlage H](#).

Bijlage F: Structuur van het wetenschappelijk artikel

1. Titel en auteursnaam

2. Inleiding

Doel van de opdracht

Relevantie van symboolherkenning binnen AI

3. Aanpak per fase

Fase 1: Structuur van het netwerk, trainingswijze, gedrag bij imperfecte input

Fase 2: Verschillen t.o.v. fase 1, invloed van de hidden layer, gebruik van matrices

Fase 3 (optioneel): Gebruik van steepest descent, analyse van convergentie

4. Resultaten

Foutpercentages, voorbeelden van correcte/foute classificatie

Grafiek of tabel als ondersteuning (indien mogelijk)

5. Reflectie

Wat werkte goed, wat was lastig?

Wat heb je geleerd over netwerken en leren?

Wat zou je bij een volgende versie anders doen?

6. Bronnen (optioneel)

Verwijzingen naar externe informatie of literatuur die je gebruikt hebt (bijv. voor sigmoid-afleiding)

Bijlage G: Structuur van het technisch artikel

1. Titel en auteursnaam

2. Abstract

Korte samenvatting (max 150 woorden) met:

- *Doel van de opdracht*
- *Kort methodisch ontwerp (GA, heuristieken)*
- *Belangrijkste bevindingen of resultaten*

3. Inleiding & Doelstelling

Context en achterliggende vraagstelling
Specifieke doelstellingen van Melody Maker
Scope van dit project

4. Architectuur & Ontwerp

Overzicht high-level componenten (b.v. GA-Engine, Fitness-module, Audio-backend)
Dataflow-diagram of beschrijving van de hoofddatapaden
Motivatie voor belangrijke ontwerpkeuzes

5. Implementatie

Beschrijving programmeertaal(s) en libraries
Uitleg van kernklassen/modules (b.v. Population, Individual, Operators)
Voorbeeldcode of pseudocode van cruciale routines (selectie, crossover, mutatie)

6. Test & Validatie

Teststrategie
Beschrijving testscenario's en gebruikte testdata
Samenvatting testresultaten (b.v. tabelletje of korte bulletpoints)

7. Gebruik & Deployment

Installatie- en setup-instructies
Uitleg command-line flags of configuratiebestanden
Voorbeeldworkflow (hoe genereer je snel een song.wav)

8. Conclusies & Aanbevelingen

Beknopt overzicht van wat is bereikt
Praktische leerpunten en eventuele beperkingen
Suggesties voor verdere verbetering

9. Bijlagen (optioneel)

Architectuurdiagrammen
Volledige code-snippets of configuratiebestanden
Extra testdata of output-voorbeelden

10. Referenties

Lijst van geraadpleegde bronnen (boeken, artikelen, documentatie)
Vermeld ook links naar gebruikte code-voorbeelden (b.v. muser.py, bach.py, sockets_demo.zip)

Bijlage H: Structuur van het eindverslag

1 Introductie en Context

Projectdoel:

Beschrijf het doel van het project en de motivatie voor de ontwikkeling van een autonome TORCS-agent.

Context en Relevantie:

Leg uit wat TORCS inhoudt, waarom een AI-driver hierin interessant is en hoe dit project past binnen de bredere toepassing van AI in autonome systemen.

2 Projectplanning en Methodologie

Projectplanning:

Geef een overzicht van de vooraf gemaakte planning en benoem de belangrijke mijlpalen.

Methodologie:

Beschrijf de werkmethode (bijv. Agile/Scrum) die je hebt toegepast om het project stap voor stap uit te voeren. Licht de fasen toe, zoals requirements, ontwerp, implementatie en testen, en benoem hoe data wordt verzameld en geanalyseerd.

(Hier beantwoord je de vraag “hoe” je te werk bent gegaan.)

3 Literatuurstudie en Theoretisch Kader

Theoretisch Kader

Bespreek de achtergrond en rationale van de gekozen aanpak. Leg onder meer de werking van neurale netwerken en andere relevante AI-technieken uit, en verwijst naar de belangrijkste literatuur en eerdere onderzoeken. Hier geef je antwoord op “wat” de theorie is en “waarom” jouw aanpak wetenschappelijk onderbouwd is.

Literatuurstudie

Geef een overzicht van de geraadpleegde bronnen en benoem hoe deze de ontwerpkeuzes en methodologie van je project ondersteunen. Gebruik de inzichten die je tijdens de boekenclubbijeenkomsten hebt opgedaan.

4 Requirements en Specificaties

Functionele Requirements

Beschrijf de taken en functies die de TORCS-agent moet uitvoeren.

Niet-functionele Requirements

Vermeld prestatie-eisen, randvoorwaarden en kwaliteitscriteria.

Acceptatiecriteria

Specificeer de criteria die gebruikt worden om te bepalen of de agent succesvol werkt.

5 Ontwerp en Architectuur

Systeemontwerp

Geef een overzicht van de architectuur met diagrammen en flowcharts waarin de opbouw van de AI-driver en de integratie binnen TORCS worden toegelicht.

Neuraal Netwerk Verantwoording

Licht de technische details toe van het getrainde neuraal netwerk, bijvoorbeeld:

- Aantal lagen en de keuze daarvoor
- Het aantal inputneuronen en welke data dit representeert
- Gebruikte activatiefuncties en de onderbouwing van deze keuze
- Eventuele andere hyperparameters (zoals leersnelheid, biases, optimizer, etc.)

6 Input Data Analyse

Data Acquisitie en Preprocessing

Beschrijf de herkomst van de inputdata en hoe deze is verzameld en voorbereikt.

Data Exploratie en Analyse

Presenteer visualisaties en statistieken van de exploratieve data-analyse, en ga in op de kwaliteit en bruikbaarheid van de data. Sluit af met een bespreking van de belangrijkste inzichten die je tijdens deze analyse hebt opgedaan.

7 Implementatie

Ontwikkelingsproces

Beschrijf de ontwikkeling en integratie van de AI-driver in TORCS.

Versiebeheer

Verwijs naar de online GIT-repository, inclusief details over activity-log, issue-management en versiebeheer.

8 Testspecs en Testrapporten

Testspecificaties

Omschrijf de testscenario's en testgevallen, inclusief de acceptatiecriteria.

Testresultaten

Documenteer de uitgevoerde tests met grafieken, tabellen en een evaluatie van de prestaties van de AI-driver.

9 Resultaten en Analyse

Resultaten

Vat de belangrijkste uitkomsten van het project samen.

Analyse

Voer een gedetailleerde bespreking uit van de resultaten, met nadruk op performance metrics en een vergelijking met de gestelde requirements.

10 Conclusie en Aanbevelingen

Samenvatting

Vat de hoofdbevindingen van het project samen.

Conclusies

Trek conclusies over de effectiviteit van de AI-driver en de algehele prestaties van de TORCS-agent.

Aanbevelingen

Geef suggesties voor verdere optimalisatie, vervolgonderzoek of implementatieverbeteringen.

11 Bijlagen en Referenties

Technische Bijlagen

Relevante code fragmenten, configuratiebestanden en extra visualisaties als nodig.

Documentatie en Extra Rapporten

Voeg aanvullende documenten toe, zoals planningsoverzichten, testdata en andere ondersteunende rapporten.

Literatuurlijst

Geef een overzicht van alle geraadpleegde bronnen en referenties.