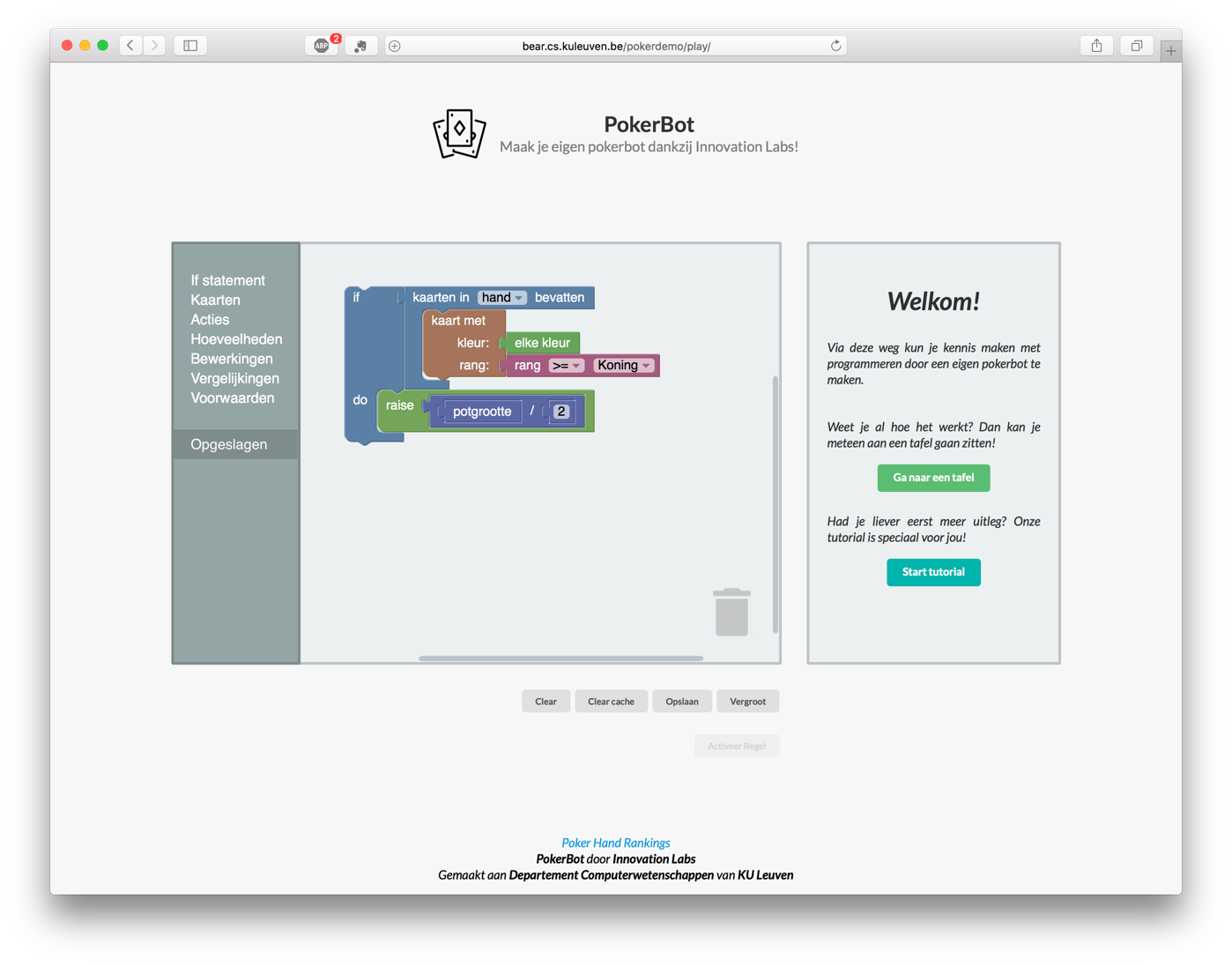


Pokerbot

Bouw je eigen artificieel intelligente poker speler,  
met een leuke mix van programmeren, wiskunde en creativiteit.



**Start de tutorial op**

[*http://dtai.cs.kuleuven.be/pokerbot*](http://dtai.cs.kuleuven.be/pokerbot)

# Deel 1: hoe werkt het

We beginnen met een stapsgewijze introductie tot de programmeer-omgeving we gaan gebruiken om je eigen speler te maken. Dit is een online programmeeromgeving, en je kan die vinden op volgende website:

[*http://dtai.cs.kuleuven.be/pokerbot*](http://dtai.cs.kuleuven.be/pokerbot)

Start met de tutorial om te beginnen (zie rechtse kadertje op de website). De tutorial begint met een videotje dat de spelregels van poker uitlegd, waarna je in 3 hoofdstukken leert hoe zelf een regel-gebaseerde poker speler te maken. In de laatste 2 hoofdstukken van de tutorial neem je het telkens op tegen een eenvoudige andere pokerbot!

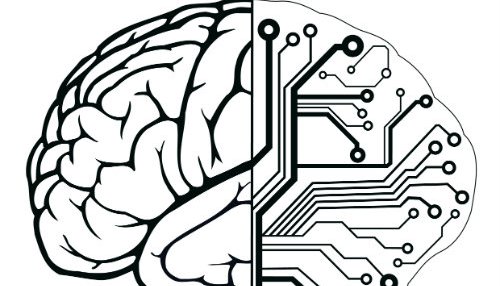
Na het afwerken van de tutorial kan je terug naar het hoofdscherm gaan door op “*verlaat tutorial”* te klikken, onderaan de werkomgeving.

# Deel 2: wat zit er achter

Nu je de tutorial doorlopen hebt en de regels van poker alsook de regelgebaseerde programmeeromgeving wat kent, gaan we dieper in op het waarom en hoe van een artificieel intelligente pokerspeler maken.

Voor dit deel kan je best samenwerken met iemand die aan een andere computer zit. Zo ga je tegen elkaar wat dingen kunnen uitproberen in de online omgeving, alsook de vragen in de bundel bespreken.

## Wat is Artificiële Intelligentie?

Een artificieel intelligent computer programma is een programma dat intelligent gedrag vertoont. Je kan je waarschijnlijk wel inbeelden dat de meningen over wat 'intelligent gedrag' is soms verschillen.

Een intelligente persoon kan vaak heel goed rekenen. Echter, we weten dat computers goed kunnen rekenen, dus dat beschouwen we de dag vandaag niet als artificiele intelligentie.

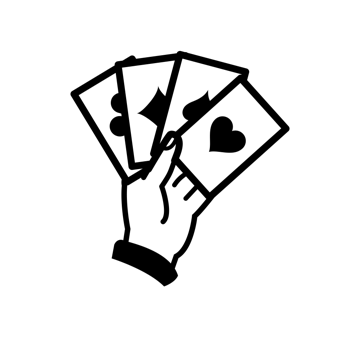
Wel zijn er veel spelen, zoals schaken, othello/reversi en go, waarvan algemeen aanvaard is dat ze intelligentie vereisen, al is het maar omdat mensen er vaak vele jaren over doen om meester te worden in zo een spel. Voor elk van deze spellen is er echter een artificieel intelligent computerprogramma gemaakt dat kan winnen van de beste menselijke speler.

Tegenwoordig komt A.I. ook steeds vaker in het nieuws, vooral als het gaat om zelfrijdende autos, deep learning technieken die audio en beeldmateriaal kunnen interpreteren en 'personal assistant' programma's zoals Siri en Alexa waar je in natuurlijke taal tegen kan spreken en die verstaan wat je wil opzoeken of versturen over het internet.

In deze sessie beginnen we iets eenvoudiger, met een regelgebaseerd systeem. Regelgebaseerde systemen zijn een heel natuurlijke manier om 'intelligent' gedrag te bevatten. Onze regel-gebaseerde bot gaat een bepaalde situatie voorgeschoteld krijgen (*input*) waarna we een bepaalde actie (*output*) terugkrijgen die gebaseerd is op het al dan niet voldoen aan bepaalde regels.

We maken bijvoorbeeld de volgende regel: *Indien ik twee azen heb, raise ik*. De bot gaat dan een hele reeks handen spelen. Elke hand is een situatie die aan de bot voorgesteld wordt zodat deze kan na gaan of aan de gegeven regel(s) voldaan wordt. Op basis van deze beslissing kan de juiste actie gekozen worden.

## Waarom Poker?

Het automatisch spelen van complexe spellen, zoals schaken en go, is al lange tijd een populair onderwerp in het onderzoeksgebied van de artificiele intelligentie.

Een eigenschap van zulke spellen is dat beide spelers altijd een volledig zicht hebben op de speltoestand (het bord), en dat het spel altijd met dezelfde speltoestand begint. Dit noemen we *spellen met perfecte informatie*. In poker daarentegen wordt het pak kaarten altijd door een deler geschud, dus zonder dat de spelers de volgorde van de kaarten kennen. Daarenboven kan je tijdens het spel enkel de kaarten in je eigen hand zien, niet die in de hand van andere spelers. Dit noemen we *spellen met imperfecte informatie*. Dit maakt het maken van een artificiele intelligente speler veel moeilijker, aangezien die niet *exact* gaat kunnen berekenen hoe het spel gaat verdergaan. Daarvoor is er teveel informatie ongekend en dus teveel mogelijke scenario's om ze allemaal in rekening te brengen.

Tot op de dag van vandaag is poker, meer specifiek de no-limit en pot-limit varianten met meer dan 2 spelers, niet oplosbaar door computerprogrammas. De enigste mogelijkheid is dus om een programma te ontwerpen, een artificieel intelligente bot, die zo goed mogelijke zetten doet. Poker is daarom een zeer interessant spel voor onderzoekers in de artificiele intelligentie, waardoor er actief onderzoek naar gedaan wordt.

In deze sessie gaan we jullie een voorproefje geven van de uitdagingen en mogelijheden van een regel-gebaseerde artificieel intelligente pokerspeler.

## Aan tafel!

Voordat we je in deze bundel enkele strategieën bijleren, moet je eerst eens tegen elkaar spelen: bespreek binnen jouw groep eens welke regels jullie hebben gebruikt om Rho-bot en Sigma-bot te verslaan.

Probeer ze nu ook tegen elkaar uit! Hiervoor moet je eerste een aparte 'tafel' aanmaken. Op de website, duw op de knop 'ga naar een tafel'. Een van jullie moet nu de rechterkant invullen “Tafel Aanmaken”. Hier moet je een naam voor je eigen bot verzinnen, een naam voor jullie tafel die jullie samen verzonnen hebben, alsook het maximum aantal spelers (e.g. 2 in dit geval) en op “Maak tafel & Ga Zitten” duwen. De andere persoon hoeft enkel de linker kant in te vullen “Plaatsnemen”. Hier kies je een naam voor je eigen bot, en vul je de tafelnaam die je samen hebt gekozen in.

Vul nu elk de regel(s) in waarmee je van Sigma-bot bent gewonnen. Welke regel wint van de andere? Zie je ook waarom? Wat zou je nog kunnen aanpassen?

In de volgende stukken leren we je meer over de wiskunde achter poker, wat je zal helpen een betere artificieel intelligente poker bot te maken.

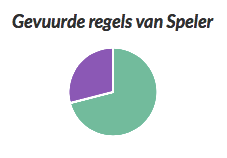
## Strategie

### Inleiding

We zullen in de volgende secties speciale notaties gebruiken om kaarten te beschrijven. Zo duiden **A**, **K**, **Q**, **J** en **T** respectievelijk op Ace (Aas), King (Koning), Queen (Dame), Jack (Boer) en Ten (Tien). Andere cijfers worden niet aangepast. Een symbool achter de waarde van een kaart duidt op zijn kleur. Zo zijn ♠, ♥, ♣ en ♦ respectievelijk schoppen, harten, klaveren en koeken. We gebruiken hiervoor het woord *kleur*, ook al bedoelen we niet enkel rood en zwart maar effectief de vier 'soorten' kleur. Schoppenboer wordt dus voorgesteld als J♠ en een hand met de twee kaarten koeken Aas en klaveren negen als A♦9♣. Indien de kleur niet is aangeduid wordt deze voorgesteld met een •.

*Tip bij gebruik van het programma:* Het kan doorheen de bundel soms handig zijn om regels op te slaan. Dit kan je doen met behulp van de ‘opslaan’-knop, onderaan je workspace. Dit creert een nieuw item in het linker menu van de workspace, onder 'Voorwaarden', genaamd 'Opgeslagen'. Duw erop om de opgeslagen regels te tonen en in je veld te slepen. Om alle opgeslagen regels te verwijderen, kun je de ‘clean-cache’-knop gebruiken onderaan de workspace.

### Grafieken

Grafieken zijn handig om af te lezen hoe vaak een regel wordt uitgevoerd. Probeer op deze manier eens uit te zoeken hoe vaak een paar Azen voorkomt. Dit kan je doen door een regel te maken die raised indien je een paar asen in je hand hebt. Wacht even tot de grafiek gestabiliseerd is en kijk hoe vaak de actie wordt uitgevoerd. Doe dit ook maar eens met een Aas en een Tien en schrijf het resultaat hieronder:

Kans op **A•, A•**: …………%

Kans op **A•, T•**: …………%

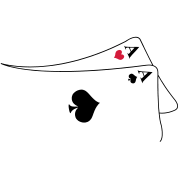
*Tip bij interpreteren van de acties:* het programma zal nooit een geforceerde actie uitvoeren. Dat wil zeggen dat indien de actie 'fold' is, het programma enkel zal folden indien een andere speler effectief een 'raise' actie heeft gedaan. Zoniet kan je een 'call' doen zonder geld te moeten inzetten (ook wel 'check' genaamd). Het programma gaat dus nooit geforceerd folden, maar een check/call doen in dat geval.

*Tip terminologie:* call/fold/check/raise/dealer/blind, wordt het allemaal wat veel? Op de voorlaatste pagina vind je een overzicht van veelgebruikte termen en wat ze betekenen. Op de laatste pagina de ranking van hoe goed een hand is volgens de poker regels.

### Kansen Berekenen

Bepalen hoe vaak een bepaalde regel of hand voorkomt is belangrijk om de juiste actie en plaats van die regel te bepalen. In het geval van 1 regel is dit makkelijk te testen zoals boven beschreven. We willen dit echter kunnen berekenen zonder het te moeten simuleren in een echt spelletje; we willen dus de kans van een hand kunnen bepalen.

Maar wat is nu eigenlijk een kans? Een kans is een hoeveelheid die de verhouding aanduid van hoe vaak iets (een **gebeurtenis**) voorkomt tegenover het aantal mogelijkheden (het **universum**):

Zo is het mogelijk op voorhand te berekenen hoe vaak een regel zal vuren. Dit kan heel wat geld besparen. We beginnen met wat basiskansen te bekijken. Wat zijn de kansen om een van deze kaarten uit een pak van ***52 kaarten*** te trekken? Tip: er bestaat maar één enkele **A**♠!

Kans op **A**♠: …………%

Kans op een **A**: …………%

Wat moeilijker is de kans op een totale hand. Zo kunnen we zelf ook berekenen wat de kans op twee Azen zou zijn. Hiervoor berekenen we de kans op de eerste en de kans op de tweede Aas. Aangezien deze onafhankelijk zijn, kunnen we ze vermenigvuldigen om de totale kans te bekomen. Vergelijk deze met de observaties de je gemaakt hebt op de vorige pagina. Komt deze overeen?

en

Kans op **A•, A•**: …………%

Kans op **A•, K•**: …………%

Als laatste kun je ook makkelijk de kans op moeilijkere situaties berekenen. Probeer eens met wat je nu weet de kans te berekenen op eender welk paar. Voor de gevorderden is er ook nog een extra oefening.

***Tip:*** *Er zijn in totaal 13 ranks in een spel kaarten. De kans is dus 13 keer zo groot als* ***AA*** *gedeeld krijgen!*

Kans op een paar: …………%

Kans op Suited Connectors *(bv***.7♣6♣** *of* Q♥J♥*)* …………%

### Handen Spelen

Het is belangrijk om niet te veel handen te spelen, maar enkel indien je relatief goede kaarten hebt. Je kan dus best een verhouding bijhouden van hoeveel handen je speelt. Je speelt een hand indien je checkt, callt of raised, maar niet wanneer je fold. Deze ratio, die aantoont hoeveel procent van de handen je speelt, heet jouw ***VPIP***.

**Wist je dat…**

Spelers die heel wat handen spelen, vaak bestempeld worden als **fish** aangezien ze makkelijk in de val te lokken zijn.



Je kan dit berekenen door de waarden af te leiden van de grafieken op de website. Een belangrijke tip is dat wanneer jouw ***VPIP*** meer dan bedraagt, je best minder handen begint te spelen. Welk van de onderstaande spelers speelt dus het minst goed?

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Speler 1*** | ***Speler 2*** | ***Speler 3*** |
| ***Checks:*** | 1400 | 3422 | 538 |
| ***Calls:*** | 1293 | 2292 | 192 |
| ***Raises:*** | 440 | 12 | 225 |
| ***Folds:*** | 9583 | 794 | 8038 |

***VPiP*** Speler 1: …………%

***VPiP*** Speler 2: …………%

***VPiP*** Speler 3: …………%

Slechtste Speler: …………

### Expected Value

Om te weten of een regel winstgevend is op lange termijn, is het behulpzaam om de ***expected value*** (of *verwachtingswaarde*) te berekenen. Dit komt neer op het berekenen van de gemiddelde winst per hand. We gaan hier samen door een simpel voorbeeld.

Stel dat we aan een tafel met twee spelers zitten. Per de regels van een twee speler spel moeten we van de tijd de small blind posten (1 credit in ons geval). De andere van de tijd moeten we de big blind posten (2 credits). Wat is dan de totale kost voor ***10 handen***?

: ………… credits

Bereken dan ook de kost per hand als je weet dat de gemiddelde kost per hand kan geschreven worden als (*Met “n” een bepaald aantal handen)*:

: ………… credits

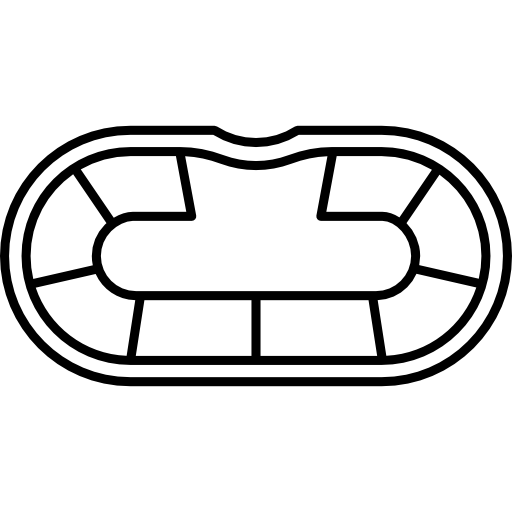
Stel dat we **A•, A•**gedeeld krijgen en besluiten om meteen all-in te gaan voor ***200 credits***. Laten we even aannemen dat twee Azen altijd winnen om het voorbeeld makkelijk te maken. We weten ook dat onze tegenstander altijd callt (dus meegaat door hetzelfde bedrag in te zetten). In dit geval winnen we met onze Azen:

We weten al dat de kans op **A•, A•** gelijk is aan …… . Dit is dus de kans op winnen met die hand. De kans op verliezen is dan …… . De winst per hand, of ***expected value***, zal dan gelijk zijn aan de verwachte winst (kans op winst \* winst) minus het verwachte verlies (kans op verlies \* verlies):

………

Is deze positief? Goed! Dan zal je winst maken op lange termijn. Is dit niet het geval, dan ben je verliezend aan het spelen.

### Invloed van de Tafelgrootte

Laten we nu eens kijken hoe de tafelgrootte onze EV-waarde kan beïnvloeden. Stel dat we met ***10 spelers*** aan tafel zitten in plaats van ***2***. We moeten ineens maar van de handen een small blind betalen en van de handen de big blind. Dit verandert onze kost (met nog steeds een small blind gelijk is aan 1 en een big blind aan 2) als volgt:

= …… credits

= …… credits

De winst blijft echter hetzelfde. Je kan nu opnieuw ***de expected value*** berekenen.

………

Merk op dat het aantal spelers de uitkomst van deze regel drastisch heeft veranderd! Terwijl deze eerst een negatieve verwachtingswaarde had, is deze nu positief. De uitkomst van bepaalde regels kan dus sterk afhangen van de tafelgrootte.

### EV: Flush-Draw

We nemen even het volgende voorbeeld:

**Speler A:** A♠K♠

**Speler B:** A♥A♣

**Tafel:** 2♠9♠3♣T♥

**Situatie:** De potgrootte is ***200 chips*** en Speler B bet ***65 chips***.

Wat is de ***EV*** van Speler A?

Beantwoord onderstaande vragen om uiteindelijk tot de oplossing te bekomen. Laat het duidelijk zijn dat Speler A aan flush moet halen om te winnen. In alle andere gevallen zal hij verliezen.

* Hoeveel kaarten op de River bezorgen Speler A winst (dit zijn ***outs***)?
* Nu je het aantal outs weet, alsook dat er ***44 kaarten*** over zijn, wat is de kans op winst voor Speler A?
* Hoeveel chips moet speler A riskeren om mee te gaan in de pot?
* Hoeveel kan Speler A winnen wanneer hij de hand wint? Let op! Dit is meer dan ***200 chips***.
* Wat is de ***expected value*** van Speler A wanneer hij zou callen?

We zien dat deze speler heel wat in rekening moet brengen om te weten of deze hand voordelig is voor hem of niet. Merk op dat het mogelijk is om al deze berekeningen te modeleren met de web interface. Het is dus mogelijk om zeer ver te gaan in het gebruiken van ***EV-berekeningen***. Er zijn heel wat berekeningen die je kan uitvoeren met potgrootte, hoeveelheid om te callen, stackgrootte, enz.

Indien je dit voorbeeld te makkelijk vond, kun je eens kijken welke invloed de volgende aanpassingen hebben op de keuze van de speler uit voorgaand vraagstuk.

**Variant 1:** Stel dat Speler B niet **A**♥**A♣** heeft, maar **A**♥**A**♠.

**Variant 2:** Stel dat Speler B niet ***65 chips*** bet, maar ***70 chips***.

### Extra: EV: Kleine Paren

Een ander voorbeeld wat te maken heeft met ***expected value*** is de volgende redering. Het verschil met vorig voorbeeld is dat hier een waarde variabel is. Stel we willen een regel maken wat betreft de kleine paren (paren lager dan een 5). We volgen de volgende simpele redenering.

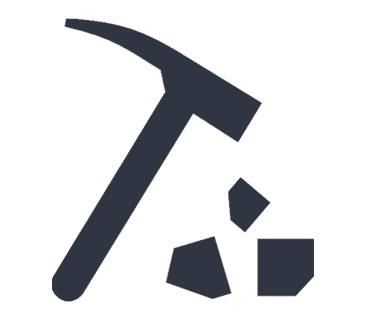
* Indien we een three of a kind[[1]](#footnote-2) maken (of beter), dan spelen we verder.
* Is dit niet het geval, dan folden we.

Het berekenen van de ***expected value*** is nu iets geavanceerder.

* Wat is de kans dat we three of a kind maken (of beter)? Tip: Bereken eerst de kans dat dit niet zo is.
* Stel nu dat het bedrag dat we moeten callen **X** is. We weten ook de kans dat we de flop missen. Wat zal dan het gemiddelde verlies per hand zijn?
* Wat is nu de minimumwaarde  dat we na de flop moeten winnen wanneer we een three of a kind (of beter) maken zodat ?

**Wist je dat…**

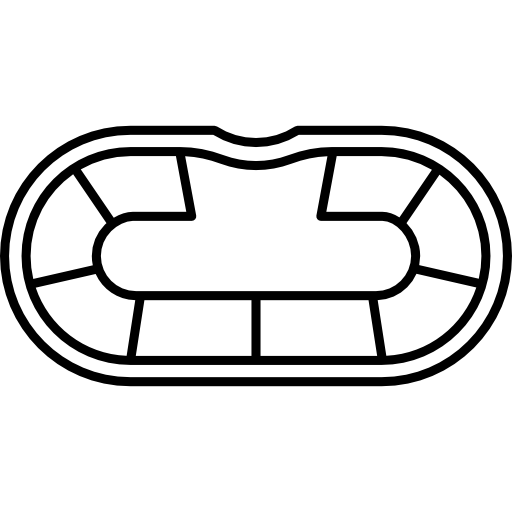
*Goedkoop* op zoek gaan naar *Three of a kind* met lage paren ook wel bekend staat als ***setmining*** en wordt vaak gebruikt door professionele pokerspelers.



Stel nu dat we, indien onze hand verbeterd op de flop, we de hele stack van de tegenstander kunnen winnen. In dit geval is het perfect mogelijk om deze complexere formule in te geven in de web interface. Je kan hier zo ver in gaan als je zelf wil.

Probeer zelf eens formule te gebruiken in een regel die ervoor zorgen dat deze een positieve verwachtingswaarde heeft. Kun jij er zelf vinden die we nog niet hebben aangehaald in deze bundel?

### Extra: Gebruik van Positie

Een deel van de interface die we nog niet gebruikt hebben, is de beschikbare hoeveelheid *“aantal actieve spelers”*. Dit kan gebruikt worden om te weten te komen hoeveel andere spelers mee in de hand zitten. Een regel die hier rekening mee houdt volgt de volgende redenering.

Des te meer spelers in de hand, des te kleiner is de kans dat een bepaalde hand de sterkst is. Stel dat we een hand hebben die  van de tijd de beste is, de andere de slechtste. Indien deze hand het opneemt tegen twee spelers, dalen zijn kansen om te winnen. Er geldt nu de volgende regel, met de kans op de beste hand en de kans op de beste hand tegen speler .

Het is dus mogelijk om een regel enkel te laten vuren indien het aantal aantal actieve spelers lager is dan een bepaald aantal. Probeer zelf deze componenten maar eens uit in de web interface.

# Deel 3: tegen elkaar!

De informatie uit het vorige deel kan je gebruiken om een betere poker bot te maken. Zo weet je dat je je VPIP in de gaten moet houden als je niet teveel geld wil verliezen; dat zowel de kans om een bepaalde hand te hebben uitmaakt, alsook de kans dat je met die hand wint, en dat je dit ook kan gebruiken om 'expected value' of verwachte winst te berekenen.

Enkele tips op basis hiervan: zeer onwaarschijnlijke handen gaan nagenoeg geen invloed hebben op je winst, hoe groot de pot ook mag zijn; focus best eerst op redelijk eenvoudige handen.

Vergeet ook niet dat een goede pokerbot meerdere regels heeft, en dat de volgorde van die regels veel kan uitmaken! Hou naast de VPIP ook in de gaten hoe vaak elke regel wordt uitgevoerd.

Zo, nu zijn jullie klaar om het allemaal tegen elkaar op te nemen!

De leerkracht zal nu 1 of 2 groepstafels aanmaken (afhankelijk van de grootte van de klas), waarin je het tegen elkaar kan opnemen.

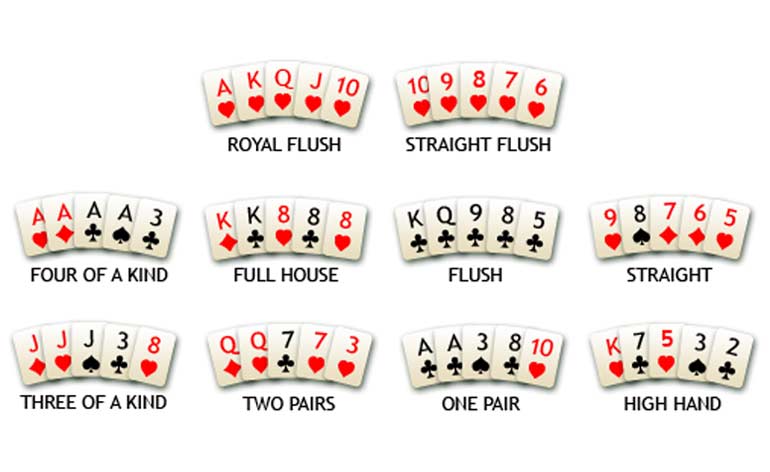
Daarnaast staat het je ook vrij om zelf een tafel te maken, en daarop anderen uit te nodigen; vb om wat rustiger dingen uit te kunnen proberen (of juist minder rustig).

# Appendix A. Terminologie

|  |  |
| --- | --- |
| Big Blind | Voordat de kaarten worden gedeeld, moet de tweede speler links van de deler een verplichte inzet plaatsen, de zogenaamde ‘big blind’. Indien er maar twee spelers zijn, is de speler links van de deler de big blind. |
| Call | Je gaat mee met de bet of raise van de tegenstander. |
| Check | Als een speler aan de beurt is, en er voor zijn beurt nog niet is gebet of geraised en hij dat zo wil laten, dan checked hij. De beurt gaat dan naar de volgende speler. |
| Dealer | Om de beurt is iedereen deler. Deze beurt wordt met de klok mee doorgegeven. Dit zorgt ervoor dat de volgorde van de spelers telkens veranderd, alsook dat iedereen eens de blinds moet betalen. |
| Expected Value | De verwachte uitkomst van een hand op lange termijn. |
| Flop | De eerste drie gemeenschappelijke kaarten die worden gedeeld, waarna een nieuwe inzetronde volgt. |
| Fold | Je kaarten weggooien. Je hebt te slechte kaarten, waardoor je niet mee speelt en je inzet accepteert als verlies. |
| Postflop | De situatie nadat de flop is gedeeld. Zowel de flop, turn als River zijn postflop. |
| Potgrootte | De grootte van de pot. |
| Preflop | De situatie alvorens de flop is gedeeld. Voorbeeld: “Ik ging preflop all-in” houdt in dat je all-in ging alvorens de flop was gedeeld. |
| Raise | Het verhogen van een bet of raise van de tegenstander. |
| River | De vijfde en laatste kaart die open wordt gelegd op de tafel. Hierna volgt weer een inzetronde. |
| Small Blind | Voordat de kaarten worden gedeeld, moet de eerste speler die direct links van de deler zit een verplichte inzet plaatsen, de zogenaamde ‘small blind’. Indien er maar twee spelers zijn, is de deler de small blind. |
| Stackgrootte | Het aantal chips in jouw bezit. Wanneer je al deze chips bet of raised, dan ben je all-in. |
| Turn | De vierde gemeenschappelijke kaart, gevolgd door een inzetronde. |
| VPIP | Afkomstig van Voluntarily Put $ In Pot. Dit getal duidt aan hoeveel procent van de handen een speler speelt. Hieruit kan worden afgeleid welke mogelijke handen hij kan hebben. |

# Appendix B. Handrankings

Van links naar rechts en boven naar onder, royal flush is het beste wat je kan halen, een hoge hand het minst goede.



Zoals je wel kan verwachten is de kans op een goede hand veel kleiner dan de kans op een minder goede hand...

1. Een *Three of a kind* waarvan *2* van de *3* kaarten in je hand zitten, noemt men ook een *set*. [↑](#footnote-ref-2)