Inleiding programmeren

1^{e} jaar wis-, natuur- en sterrenkunde

Universiteit van Amsterdam

september 2014

hacker opgaves

Week 2: bisectie

Week 3: bingo

Week 4: fitten

Week 5: fractals

Week 6: botsende deeltjes

Vrij: solar system

1 Bisectie methode: oplossen vergelijkingen

Probeer de oplossing van een vergelijking te vinden door iteratief steeds dichter bij de oplossing te komen.

Probleem: bepaal oplossing x_0 voor $f(x_0) = y$.
Gegeven $f(x)$ met één oplossing voor $f(x) = y$ op $a \leq x \leq b$.

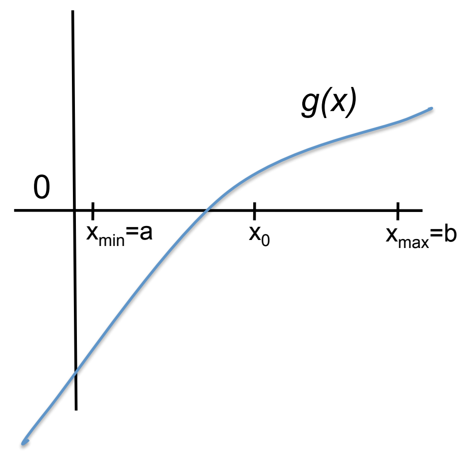
Oplossing: bepaal nulpunt x_0 van $g(x) = f(x) - y$

Als je de functie herschrijft, $g(x) = f(x) - y$, dan vertaalt het probleem zich in het vinden van het nulpunt van g(x) op het gebied $a \leq x \leq b$. Er geldt dus nu dat $f(a)f(b) < 0$, want als $f(a) < 0$ dan $f(b) > 0$ en andersom.

Definieer aan het begin $x_{\min} = a$ and $x_{\max} = b$. We gaan nu iteratief het nulpunt vinden door in elke stap $x_{\min}$ of $x_{\max}$ opnieuw af te schatten.

Het idee bij elke stap in de iteratie:

1) Definieer $x_0 = \frac{1}{2}(x_{\min} + x_{\max})$
2) Bepaal $g(x_0)$
3) Als $g(x_0) > 0$, dan $x_{\max} = x_0$ ($x_{\min}$=$x_{\min}$)
   Als $g(x_0) < 0$, dan $x_{\min} = x_0$ ($x_{\max}$=$x_{\max}$)
4) return to 1)



opgavenset bisectie

opgave [1]: bisectie methode

a) Bereken x=$\sqrt{5}$ met behulp van de bisectiemethode. Stop als $|x^2 - 5| < 0.0001$.

b) Numerieke precisie. Maak een grafiek van x^2-5 als functie van het aantal stappen dat je maakt in de bisectiemethode. Wat is uiteindelijk de precisie die je kan halen ? Wat definiëert dat ?

2 Bingo!

Op vakantie spelen vier kinderen een potje bingo. In het spel zijn 90 balletjes met getallen van 1 t/m 90. Elke speler krijgt een bingokaart met 15 getallen en er wordt steeds 1 balletje uit de pot gehaald. Wie het eerst al zijn getallen 'vol' heeft heeft gewonnen.

Tijdens een van de spelletjes waren er 7 balletjes over toen de eerste persoon bingo had (er waren dus 83 balletjes getrokken). Dat lijkt bijzonder. De vraag is nu: hoe bijzonder is dit ?

Concreet: gebruik dezelfde 4 bingo-kaarten, maar simuleer steeds een nieuw potje bingo. Welke fractie van bingo spelletjes is pas afgelopen nadat er 83 of meer balletjes getrokken zijn ?

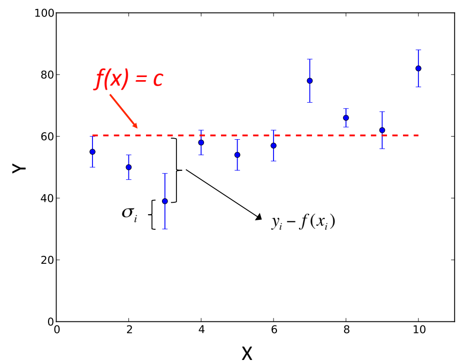
Note: je mag de volgende syntax gebruiken om element i uit een lijst te halen: `del L_getallen[i]`.

```
[1] L_getallen = [990,34,12,56]
[2] del L_getallen[2]
[3] print L_getallen
>>  [990,34, 56]
```

3 Fitten van data en foutenbepaling (theorie)

Om de onderliggende fenomenen van (natuurkundige) verschijnselen te achterhalen wordt data verzameld om afhankelijkheden te onderzoeken. Dat kan de massa van het Higgs boson zijn, de vervaltijd van uranium, maar ook het aantal kinderen in een gezin als functie van de gemiddelde lengte van de ouders. Je kan dan zoeken naar een (causaal) verband: lineair, exponentieel, etc. en daarbij ook de bijbehorende parameters bepalen met hun onzekerheid. Als je een goede beschrijving hebt gevonden kan je daarmee vervolgens ook voorspellingen doen.

Om de 'beste' waarde te vinden hebben we een maat nodig om de *'goedheid'* van de fit te kwantifiseren. We doen dat hier met de χ^2-maat: de som van de gemiddelde afwijking van de meetpunten tot het model gewogen met hun fout: 'hoeveel standaardafwijkingen ligt dit punt weg van mijn functie'.



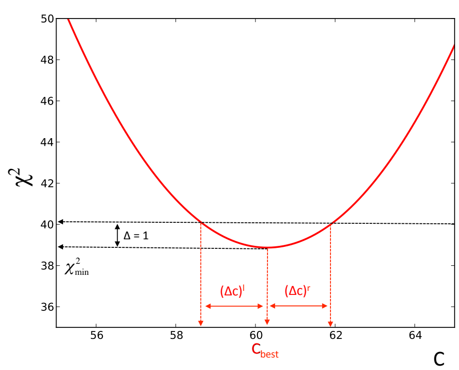
$$\chi^2 = \sum_{i \text{ (datapunten)}} \left(\frac{y_i - f(x_i|\vec{\alpha})}{\sigma_i} \right)^2,$$

met $\vec{\alpha}$ de vector met functie-parameters.

Voorbeeld: Een model met maar 1 parameter $f(x|\vec{\alpha}) = c$ wordt gebruikt om de data in (bovenste) figuur te beschrijven. Probeer dan verschillende waardes voor c en bereken voor elke waarde van c de bijbehorende χ^2. Die figuur is beneden te zien.

$\rightarrow$ **de beste waarde van c:** c_{best}
De waarde van c waarbij de χ^2 minimaal is.

$\rightarrow$ **de onzekerheid op** $c_{\text{best}}(\Delta_c)$
De fout in de positieve richting $(\Delta_c)^r$ en negatieve richting $(\Delta c)^l$ zijn die waardes van c waarbij de χ^2 1 hoger is dan χ^2_{min}. In de meeste gevallen geldt $(\Delta c)^l = (\Delta c)^r = \Delta c$



Het eindresultaat van je meting is dan: $c = c_{\text{best}}\,{}^{+(\Delta_c)^r}_{-(\Delta_c)^l}$

opgave [1]: fitten van een model aan de data

De onderstaande data-set geeft voor een specifieke voetballer het percentage goede passes (y) weer als functie van het aantal gespeelde wedstrijden in net Nederlands elftal (x). De onzekerheid op het aantal goede passes (het is een schatting gebaseerd op de korte samenvatting op televisie) is weergegeven als σ_y.

x	1	2	3	4	5	6	7	8	9	10
y	55	50	39	58	54	57	78	66	62	82
σ_y	5	4	9	4	5	5	7	3	6	6

a) maak een plot van deze data met fouten
Computing tip: gebruik de functie `plt.errorbar(x,y, yerr=yerror)`

b) bereken de beste waarde van c als $f(x)$ =c en de bijbehorende onzekerheid Δc.

c) Wat gebeurt er met Δc als de fout in elk meetpunt 2x kleiner wordt ?

Het lijkt erop of er een 'trend' zichtbaar is, dus we breiden ons model uit met een lineaire term: $f(x) = \mathrm{b}x + \mathrm{c}$.

d) bereken de beste waardes voor b en c als $f(x) = \mathrm{b}x + \mathrm{c}$.

Bij veel banken en verzekeraars werken wis- en natuurkundigen in de zogenaamde risk-analysis departments. Laten we proberen de eindstand van de AEX te voorspellen in het jaar 2000 door gebruik te maken van de eindstand van de AEX in de jaren 1991-1999. Er zit geen 'fout' op de standen. Zet in de χ^2-formule de fout op de meetwaarde op 1,00.

jaar	1991	1992	1993	1994	1995	1996	1997	1998	1999
AEX	125.72	129.71	187.99	188.08	220.24	294.16	414.61	538.36	671.41

e) Maak een grafiek van de eindstand van de AEX als functie van het jaar sinds 1991.

f) Fit de grafiek met een polynoom van graad 2: bereken dus de beste waardes voor a, b en c als $f(x) = \mathrm{a}x^2 + \mathrm{b}x + \mathrm{c}$.
Tip 1: gebruik x = 1, 2, 3 i.p.v. 1991, 1992, 1993 etc.
Tip 2: probeer de waardes van a, b en c te schatten voor je gaat fitten.

g) Wat is je voorspelling voor het jaar 2000 ? Zet beide waardes in de plot en vergelijk deze met de echte waarde in het jaar 2000. En ? Hoe kan dat ?

"resultaten behaald in het verleden bieden geen garantie voor de toekomst"

4 Fractal: Mandelbrot set

We gaan proberen de Mandelbrot set te tekenen op het scherm. Dit vereist kennis van complexe getallen.

Voor elk punt in het complexe vlak ($z_0 = x_0 + iy_0$) definiëren we een polynoom die aangeeft naar welke punt (complex getal) het punt zich verplaatst. Een belangrijke set polynomen wordt gegeven door:

$$f(z) = z^2 + c$$

Elk punt z_0 dat je definiëert geeft een nieuw punt $z_1 = f(z_0) = z_0^2 + c$. Het is zo steeds mogelijk om het volgende punt te berekenen, immers $z_2 = f(z_1) = z_1^2 + c$ en meer algemeen $z_n = f^{(n)}(z_0)$. Er zijn nu 2 mogelijkheden:

De reeks convergeert	$\rightarrow z_0$ is *wel* onderdeel van de set
De reeks divergeert	$\rightarrow z_0$ is *geen* onderdeel van de set

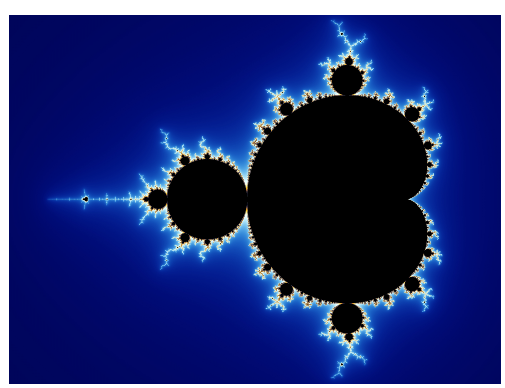
Elke keuze van c geeft een uniek patroon. De Mandelbrotset is speciaal omdat in dat geval geldt dat c het startpunt zelf is, dus $c = z_0$. Als je een punt een kleur geeft als het in de set zit krijg je de karakteristieke plaatjes zoals beneden.

opgavenset Fractal

opgave [1]: Mandelbrot set

Bepaal voor elk punt z in het complexe vlak of het convergeert of divergeert onder de polynoom $f(z) = z^2 + z_0$. Als het convergeert teken het punt blauw, als het divergeert teken je het wit.

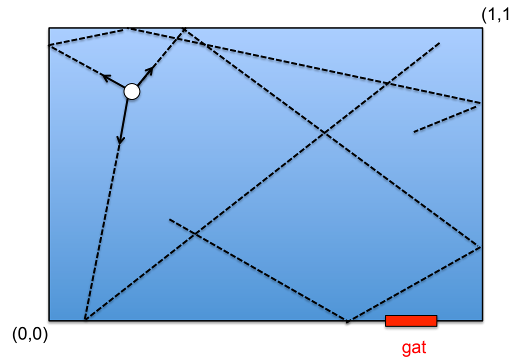
Note: in deze opgave krijgen we niet echt een 'glossy' plot zoals hier rechts. In dat soort figuren wordt de kleur bepaald door de snelheid waarmee een punt divergeert (of convergeert).



Voor specifieke waarden van c in de functie $f(z) = z^2 + c$ zijn er zogenaamde Julia sets. Probeer er daar een paar van na te maken.

5 Botsende deeltjes in een doos (hacker week 6)

Er zijn verschillende mogelijkheden om de simulatie van week 6 uit te breiden met meer realisme. Het visualiseren van een simulatie is erg leuk en interessant omdat het snel programmeerfouten blootlegt en fenomenen laat zien die soms niet gelijk opvallen als je naar de vergelijkingen zelf kijkt. We zullen de deeltjes ook een 'echte' afmeting geven en de deeltjes te laten botsen.



a) Animatie puntdeeltjes. Gebruik het voorbeeld `animation_template_circles.py` om om de deeltjes door de doos te zien bewegen als functie van de tijd.

b) Animatie deeltjes met afmeting. Geef de deeltjes een afmeting en laat ze netjes van de wand ketsen als de rand van het deeltje de wand raakt

c) Laat de deeltjes realistisch botsen, dus met de echte afmeting.
Tip voor afleiding: *2-dimensional elastic collisions without trigonometry*

Zodra twee deeltjes botsen veranderen zowel de snelheid als de bewegingsrichting van de deeltjes. Interessant is dat de snelheidsverdeling uiteindelijk de Boltzmannverdeling zal volgen, onafhankelijk van de begincondities. We willen dat graag bewijzen door de verdeling te bekijken na een tijd lang botsen.

d) Begin met dezelfde condities als in vraag 1a) en maak een grafiek van de snelheidsverdeling na 100 en na 1000 botsingen.

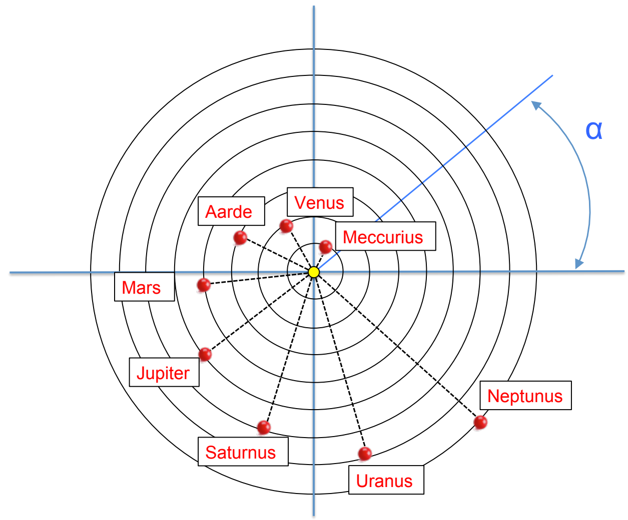
e) Probeer (om fluctuaties te voorkomen) een 'gemiddelde' snelheidsverdeling te maken. Dus bijvoorbeeld het gemiddelde profiel tussen botsing tijdstap 95 en 105.

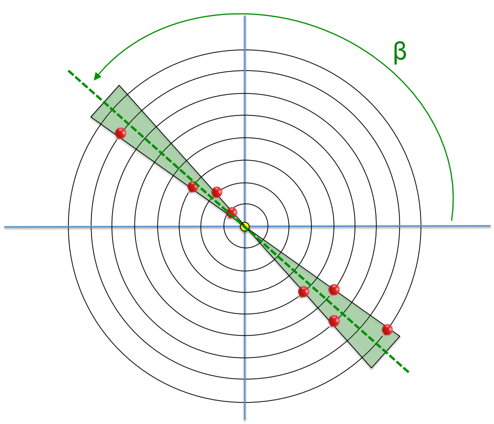
f) Andere begincondities. Begin nu eens met een beginsituatie waarin alle deeltjes dezelfde snelheid hebben en vergelijk de uiteindelijke snelheidsverdeling met die uit de vorige vraag. Wat valt je op ? Zijn ze inderdaad hetzelfde ?

6 Solar system

Ons zonnestelsel bestaat uit 8 planeten die elk met een eigen omwentelingssnelheid om de zon draaien. We gaan in deze opgave de positie (de hoek α) van elk van de planeten berekenen als functie van de tijd.

In deze opgave beginnen de planeten allemaal op $\alpha = 0$ en we gaan bepalen hoe vaak het voorkomt dat de planeten ongeveer op één lijn staan.





Neem stapjes in de tijd en bepaal voor elke stap in de tijd de as (hoek β) waarvoor de planeten het meest gealinieerd zijn. Doe dit door de som van de afwijkingen (Δ_i) in hoek tussen β en die van de individuele planeten te minimaliseren: $\Delta_{\text{tot}} = \sum_i \Delta_i$.

opgave [1]:
Beschouw Mercurius en Venus en plot Δ_{tot} als functie van de tijd gedurende 5 jaar. Na hoeveel tijd staan de 2 planeten weer op één lijn ? Plot ook β versus de tijd.

opgave [2]:
Hetzelfde als bij vraag 1, maar dan met 3 planeten. Beschouw ook de Aarde.

opgave [3]:
Hetzelfde als bij vraag 1, maar dan met 8 planeten gedurende 100 jaar. Wat is de minimale waarde van Δ_{tot} in die periode ? Teken de configuratie op het scherm.