|  |  |
| --- | --- |
| Biostatica  Door Bart van Trigt, Alistair Vardy en Mark Schrauwen | Matlab Wk2.1 |

Inhoudsopgave

[Versiebeheer 2](#_Toc491774800)

[1 Inleiding 3](#_Toc491774801)

[2 Functies met meerdere outputs 4](#_Toc491774802)

[2.1 Vragen en opdrachten 5](#_Toc491774803)

[2.2 Antwoorden en uitwerkingen 5](#_Toc491774804)

[3 Functie met meerdere inputs 6](#_Toc491774805)

[3.1 Vragen en opdrachten 7](#_Toc491774806)

[3.2 Antwoorden en uitwerkingen 8](#_Toc491774807)

[4 Ingewikkeldere functies maken 9](#_Toc491774808)

[4.1 Vragen en opdrachten 10](#_Toc491774809)

[4.2 Antwoorden en uitwerkingen 11](#_Toc491774810)

[5 Matrices; specifieke terminologie en eigenschappen 12](#_Toc491774811)

[5.1 Vragen en opdrachten 14](#_Toc491774812)

[5.2 Antwoorden en uitwerkingen 15](#_Toc491774813)

[6 Opdrachten 16](#_Toc491774814)

# Versiebeheer

|  |  |  |  |
| --- | --- | --- | --- |
| Versie | Datum | Beschrijving | Door |
| 0.0 | 11-07-2017 | Eerste versie | Alistair Vardy |
| 0.1 | 13-09-2017 | Under construction | Bart van Trigt |
| 0.2 | 14-09-2017 | Meerder outputs en meerdere inputs klaar  Nu verder met ingewikkeldere functies | Bart van Trigt |
| 0.3 | 18-09-2017 |  | Bart van Trigt |
|  |  |  |  |
|  |  |  |  |

# Inleiding

We hebben vorige week heb je kennis gemaakt met MATLAB. Als het goed is weet je nu waarom jij als bewegingstechnoloog MATLAB goed moet gebruiken. Vorige week heb je geleerd wat operatoren zijn, hoe je kan debuggen, hoe je vectoren maakt in MATLAB, standaard functies in MATLAB en je hebt al een keer een functie zelf gemaakt! We gaan in deze reader verder met het maken van een functie. Tot nu toe heb je makkelijke functies gemaakt, je gaf een input en er kwam een output uit. Maar wat nu als we bijvoorbeeld een formule hebben die meerder inputs heeft. Bijvoorbeeld

De volgende leerdoelen zijn hiervoor gemaakt, na het doorwerken van deze reader kan de student:

1. In eigenwoorden uitleggen wat een functie in Matlab is.
2. Gebruik maken van standaard functies in Matlab.
3. Een functie met meerdere outputs gebruiken.
4. Een functie met meerder inputs kunnen gebruiken.
5. Een moeilijkere functie in Matlab maken (met matrix output) (*apart onderdeel matrices en vectoren*)
6. Wk2.1: Matlab terminologie. Denk aan: colon, operator, indices, etc.
7. Wk2.1: Matlab help/doc functionaliteit

**Zie je een fout? Of heb je een aanbeveling dan horen we dat graag! Stuur dan een e-mail naar** [**mjschrau@hhs.nl**](mailto:mjschrau@hhs.nl) **en wij passen het dan z.s.m. aan.**

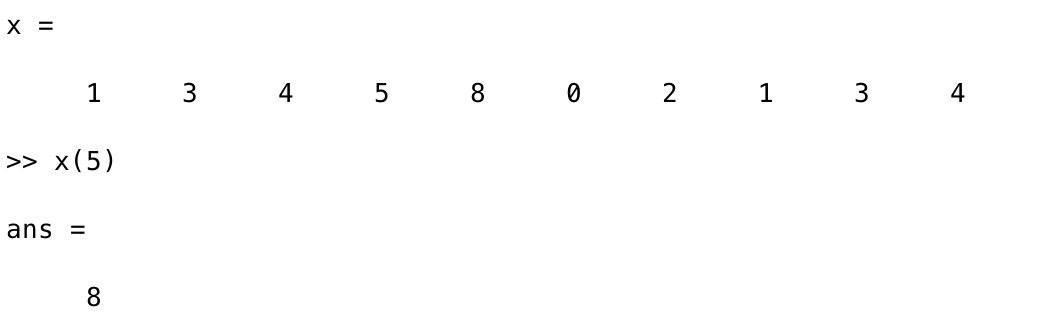
# Herhaling functies vorige week

Vorige week heb je kennis gemaakt met functies. Deze reader gaat hier verder op in. Om deze reader goed te begrijpen is het van belang dat je de reader van vorige week hebt afgerond. We herhalen hier een klein stukje van vorige week, zodat we daarna verder kunnen gaan met het gebruik maken van ingewikkeldere functies.

# Functies met meerdere outputs

We beginnen met een functie die meerder outputs geeft. Stel je wilt bij een 100meter sprint van Usain Bolt weten op welk moment zijn snelheid maximaal is. Nu gaat er vast een lampje branden, vorige week heb je geleerd dat er een functie max(x) is. Waarmee je een maximale waarde kan berekenen. Als output krijg je dan een maximale waarde, maar we willen meer weten, namelijk bij welke afstand is dit? Dit kan je doen door te kijken naar een index nummer.

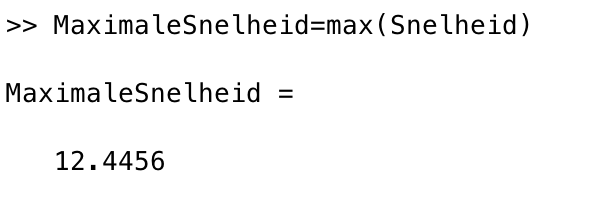
**Wat is een index?**  
Je kunt het vijfde element van een vector x opvragen met het commando x(5). Het getal 5 heet hier de index. Dus wanneer je van het onderstaande plaatje 5 stapjes telt zie je dat je bij 8 uitkomt.



Echter, weten we bij Usain Bolt niet waar zijn maximale snelheid is, dus je kan dan kijken naar op welke moment vinden we de hoogste waarde in de vector en wat is daarbij de index.

Open het bestand Usain\_Bolt.mat of sleep het bestand naar je workspace. Nu verschijnt er een variabele met Snelheid in je Workspace.

Gebruik nu de functie max. Vul in Maximalesnelheid=max(Snelheid)



We weten dus dat zijn maximale snelheid 12.44 m/s is, maar we weten nog steeds niet op welk moment. Dit kunnen we doen door nog een output te krijgen. Dit doe je door voor je functie aanvraag blokhaken te zetten, tussen de blokhaken kan je meerdere output variabele zetten, dat ziet er dan als volgt uit:

[MaximaleSnelheid, index]=max(Snelheid)

Voer dit in in Matlab.  
Op hoeveel meter is Usain Bolt op zijn maximale snelheid? Laat je antwoord zien aan de docent

## Vragen en opdrachten

Beantwoorde de volgende vragen met juist of onjuist:

1. We hebben y =[3 6 7 12 8 15 20 2 4]; Om het maximum te bepalen maken we gebruik van het commando max[y].
2. We hebben een commando max en willen het indexnummer weten dit doen we met het volgende statement: [hoogstewaarde,index]=max(x)
3. Je wilt meerder outputs berekenen hiervoor gebruik je { output1, output2 }=min(x)

## Antwoorden en uitwerkingen

1. Het antwoord is onjuist, het commando bevat ronde haken, geen blokhaken
2. Het antwoord is juist, met dit statement bereken je de hoogste waarde en de index.
3. Het antwoord is onjuist, je gebruikt blokhaken [ ] en geen accolade { }.

**Extra opdracht:**

**Voer het volgende commando in:**

**x = [20 11 11 5 10 23 14 8 8 20];**

**Bereken de index van het maximum en die van het minimum**

**Het kan voorkomen dat het maximum meerdere keren voorkomt.**

**Voer het volgende commando in:**

**x = [20 11 11 5 10 13 14 8 8 20];**

**Onderzoek wat Matlab nu als antwoord geeft.**

# Functie met meerdere inputs

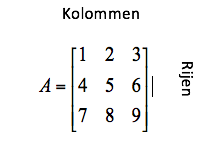
Net hebben we gezien dat een functie meerdere outputs kan hebben, je raad het al, een functie kan ook meerder inputs hebben.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Kolom 1 | Kolom2 | Kolom 3 |
| Rij 1 | 1 | 2 | 3 |
| Rij 2 | 4 | 5 | 6 |
| Rij 3 | 7 | 8 | 9 |

Bekijk de volgende matrix. We versimpelen even wat een matrix is en kunnen het zien als een soort tabel, een matrix bestaat uit rijen en kolommen. In het vervolg van je studie ga je meer leren over matrixen.

Leuk weetje:

De eerste drie letters van MATLAB staat voor matrix en de laatste drie voor laboratory

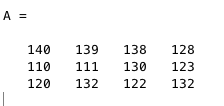


Het berekenen van het gemiddelde van een vector hebben we vorige week al gedaan dat doe je met het commando mean(x), de x is hier je vector. Van een matrix kunnen we de rijgemiddelden en de kolomgemiddelden uitrekenen. Echter, we zullen dit wel moeten aangeven. Een matrix/tabel heeft twee dimensies; de eerste zijn de rijen, de tweede de kolommen. Wanneer je de index opvraagt van bovenstaande matrix krijg je bij het volgende: Element A(3,2) is dan ook gelijk aan 8. Je gaat 3 rijen naar beneden en twee kolommen naar rechts. A(rijen,kolommen)

Stel je hebt een bloeddrukmeting gedaan bij drie proefpersonen, zoals je weet moet je een meting meerder keren uitvoeren wil het betrouwbaar zijn, dus je hebt vier keer hun systolische (boven)bloeddruk gemeten. De proefpersonen zijn de rijen en de metingen de kolommen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Meting 1 | Meting2 | Meting3 | Meting 4 |
| Mark | 140 | 139 | 138 | 128 |
| Bart | 110 | 111 | 130 | 123 |
| Hubert | 120 | 132 | 122 | 132 |
|  |  |  |  |  |

**Van de meetgegevens is de volgende Matrix gemaakt:**



Kopieer de onderstaande code naar je command window en je hebt dezelfde matrix.

A= [ 140 139 138 128; 110 111 130 123; 120 132 122 132]

Nu willen we het gemiddelde van de proefpersoon berekenen over de vier metingen. Dit kan je doen door twee inputs te geven aan de functie mean(input1,input2) dat wordt als volgt mean(A,2). Je geeft hierin nu de matrix mee (A) en met de twee geef je aan dat je de tweede dimensie wilt hebben, dus dat er een kolomvector over moet blijven en deze is de tweede dimensie. Wil je de gemiddelde bloeddruk per proefpersoon weten dan middel je over een rij, dus de vier metingen.

**Vraag: Wie heeft de laagste bloeddruk en hoeveel is die?**  
antwoord: Bart met 118.5

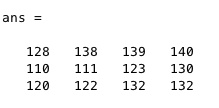
Je kan nu ook middelen over de kolommen, om te kijken of een meting ergens verkeerd ging. Zo blijft er een rijvector over. Matlab telt dan alle rijen bij elkaar op en middelt ze dan over de rijen. Dit doe je dan niet met mean(A,2) maar met mean(A,1). Dus over de eerste dimensie en dit zijn de rijen. Voer dit commando uit. Wat zie je?

Stel we willen de bloedrukmetingen mooi op volgorde zetten per proefpersoon. Hiervoor heeft Matlab ook een ingebouwde functie namelijk sort.

Voer het volgende commando in:



Je geeft nu zelfs drie inputs mee aan de functie. Namelijk je matrix, de dimensie en een string waarmee je zegt oplopend.



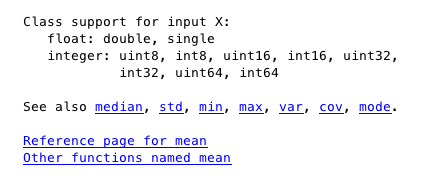
Je ziet nu dat de metingen mooi oplopen per proefpersoon. Probeer het ook eens met ‘decend’ wat zie je nu?

De help functie!

Omdat er zoveel functies zijn kunnen we nooit helemaal onthouden wat voor inputs en outputs er zijn. Daarom maken we gebruik van de help functie.

Voer maar eens in je command window: **help mean**

Er verschijnt nu een hele uitleg wat de input en outputs zijn van de functie. Om het wat overzichtelijker te maken kan je op reference page for mean klikken onderaan.

****

Dit kan je voor elke functie doen, de help functie is je grootste vriend, ook al is het soms lastig te lezen in het engels.

## Vragen en opdrachten

Beantwoorde de volgende vragen met juist of onjuist:

1. Een functie kan meerdere inputs hebben.
2. Wanneer je een index wilt oproepen van een Matrix gebruik je: A(kolommen,rijen)
3. Een functie kan verschillende inputs hebben, op dezelfde plek waarmee je hem aanroept.

Open vragen:

1. Wat is de relatie tussen de help functie en input?
2. Wat gebeurd er wanneer je maar 1 input invoert terwijl een functie wel meerder kan hebben?
3. Met wat kan je een tabel vergelijken?

## Antwoorden en uitwerkingen

1. Juist
2. Onjuist
3. Juist

Open vragen:

1. Met de help functie kan je kijken wat de input is bij een specifieke functie.
2. Dan neemt de functie gewoon de default voor deze inputs.
3. Een Matrix

# Ingewikkeldere functies maken

We hebben vorige week al een functie gemaakt, maar hier gaan we een ingewikkeldere functie maken. We hebben net geleerd dat een functie namelijk meerdere inputs en meerdere outputs kan hebben.

We gaan nu een functie maken waaruit we meteen het gemiddelde, de mediaan, de maximale kniehoek en de minimale kniehoek kunnen bepalen van een fietser.

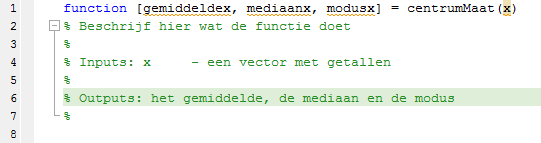
Maak een nieuwe functie met de naam KniehoekUitkomsten.m

De input data in je functie is de kniehoek, je output data is je gemiddeldeKniehoek, MediaanKniehoek, MaxKniehoek en MinKniehoek.

**Zoek de help tekst van de functie mean, median en mode op zodat je de werking van de functies berijpt.**

**Maar een nieuwe functie aan met de naam centrumMaat.m**

De eerste regels zullen er ongeveer zo uitzien



**Maak de functie af.**

**Voeg nu nog een extra output toe waarin je het maximum berekent. Pas de code aan en vergeet daarbij het commentaar niet.**

We kunnen nu aan de functie ook de KniehoekSnelheid mee geven als input, doe dit en bereken dezelfde waarden als hierboven voor de KniehoekSnelheid.

## Vragen en opdrachten

Beantwoorde de volgende vragen met juist of onjuist:

1. Wanneer je een functie maakt kan je zoveel inputs meegeven als je wilt.
2. Je kan een functie alleen aanvragen met je ‘command window’ en niet in een script

Open vragen:

## Antwoorden en uitwerkingen

Open vragen:

1. Juist
2. Onjuist, je kan juist ook een functie aanvragen in je script, dit zorgt ervoor dat je netjes programmeert.

# Specifieke terminologie en eigenschappen

We beschrijven hier een aantal Matlab termen en functies nader die het eenvoudiger maken om een functie te schrijven of data te analyseren

* end – Hiermee geef je het laatste element van een vector aan. Stel



Dan is het resultaat van het commando v(end) gelijk aan 4. Dit is buitengewoon handig als je niet van tevoren weet hoe lang een vector zal zijn.

* Colon (de dubbele punt) – Om van een matrix een hele rij of een hele kolom op te vragen kun je een dubbele punt gebruiken. Stel



Als we de derde kolom op willen vragen dan kunnen we dat doen met het commando A(1:4,3), rijen 1 t/m 4 van de derde kolom. Het volgende commando werkt ook, A(1:end,3), rijen 1 t/m de laatste van de derde kolom. Maar je kun dit nog eenvoudiger doen met het volgende commando: A(:,3). Dit geeft alle elementen uit kolom 3. Zo geeft A(2,:) alle element uit de tweede rij.

* Index – Deze term is uitlegd bij het hoofdstuk: ‘functie met meerdere inputs’
* Transponeren – Vooral bij vector- en matrixbewerkingen is het vaak nodig om de getransponeerde van een matrix te bepalen.
* Operator (*operation)* – een wiskundige bewerking zoals optellen of vermenigvuldigen. Matlab gebruikt de gangbare symbolen (+,-,\*,/,^). Let wel goed op de volgorde van de bewerking. Verwar deze term niet met de Engelse term operator, dit zijn bewerkingen zoals differentiëren en projecteren.
* Dimensie – Een vector of matrix heeft bepaalde dimensies, of afmetingen. In Matlab wordt met de eerste en tweede (en verdere) dimensies van een matrix specifiek de rijen (eerste dimensie) en kolommen (tweede dimensie) bedoeld.

# Handige Matlab functies

Hieronder volgt een beschrijving van een aantal handige Matlab functies.

Maak eerste een matrix(A) en een vector (v) aan in matlab, zoals je hieronder kan zien:

We gaan nu op deze data wat functies loslaten, voer voor elke functie de A en de v in. Erachter staat beschreven wat de functie doet.

* size() – dit commando geeft de afmetingen van een vector of matrix. Voor de matrix geeft het commando size(A) de vector [2 4] terug. De matrix heeft dus 2 rijen en 4 kolommen. Voor de vector geeft size(v) de vector [1 4] terug; een rij en 4 kolommen.
* isempty() – Dit commando geeft aan of een variabele leeg is.
* numel() – Dit commando geeft het aantal elementen van een variabele aan. Voor de matrix A is dat 8 en voor de vector v 4.
* strcmp() en strcmpi() – Met deze commando’s kun je bepalen of twee strings aan elkaar gelijk zijn. Het commando strcmp() houdt wel rekening met hoofdletters, het commando strcmpi() is ongevoelig (*insensitive*) voor hoofdletters
* find() – hiermee kun je de indices achterhalen van elementen met een specifieke inhoud, bijvoorbeeld find(v==2) geeft alle indices van de vector v waar een 2 staat.

## Vragen en opdrachten

1. Maak in Matlab een nieuwe variabele aan S1 = ‘kiwifruit1234’ en een S2 = ‘kiwibird123’. Roep de strcmp() functie aan zodanig dat je de string S1 vergelijkt met S2. Wat krijg je als antwoord terug? Wat betekent het getal dat Matlab teruggeeft?
2. Size() geeft de afmetingen van een vector of matrix. Stel de functie geeft het volgende terug ans = 2 3. Stelling: dit betekend dat de matrix 2 kolommen en 3 rijen heeft. Waar/ niet waar.

## Antwoorden en uitwerkingen

1. De variabele S1 heeft 13 karakter, en S2 heeft er 11. Strcmp() geeft terug of de strings evenveel karakters hebben, wat in dit geval dus niet zo is. Strcmp() geeft een 0 terug, oftewel een 0 is false, is de variabele zijn niet gelijk aan elkaar in karakters
2. Dit is onjuist. In H5 wordt uitgelegd dat de eerste dimensie de rijen zijn en de tweede dimensie de kolommen. Ans = 2 3 betekend dat de matrix 2 rijen heeft en 3 kolommen

# Opdrachten

* Zoek de help tekst van iseven op en bekijk wat het gedrag is voor een vector, bijvoorbeeld

v = [1 2 3 4];

* Maak een string aan met je naam, bijvoorbeeld str = ‘Alistair’; Bepaal de afmetingen en het aantal elementen.
* Stel we hebben een vector met 100 getallen. Probeer te achterhalen hoe
  + je alle oneven elementen kunt ophalen
  + alle elementen met een oneven getal kunt ophalen
* Bekijk de help tekst van het commando find en bepaal op deze manier waar in de vector

v = randi(20,1,20);

de getallen met waarde groter of gelijk aan 10 zitten.

* Wat is het verschil tussen de volgende twee commando’s:
  + find(v==10);
  + v==10

oud:

We kunnen een matrix transponeren. Dat wil zeggen dat de rijen de kolommen worden. Bijvoorbeeld



Wiskundig gezien is dit de notatie. In Matlab gebruiken we een aanhalingsteken ‘om de getransponeerde te krijgen, bijvoorbeeld B2 = B’;.

**Maak de matrix B in Matlab aan en laat zien dat de rijsommen van de getransponeerde gelijk zijn aan de kolomsommen van de originele matrix B.**