KM 1(1980)1

NAUWKEURIGE BEREKENING VAN GEMIDDELDE EN STANDAARDDEVIATIE MET EEN (ZAK)REKENMACHINE

Floor van Nes*

Samenvatting: Berekening van gemiddelde en standaarddeviatie kan gebeuren volgens de definitie van deze grootheden. In de praktijk worden vaak afgeleide formules gebruikt die het voordeel bieden dat de gegevens slechts één keer behoeven te worden "gelezen". Deze formules zijn analytisch wel gelijk aan de definities, maar numeriek niet. Het blijkt dat uitgerekend de meest gebruikte formule slechte numerieke eigenschappen heeft. Er bestaat een beter one pass algoritme dat werkt volgens het principe van updating.

trefwoorden: statistische programmatuur, gemiddelde, standaarddeviatie, numerieke nauwkeurigheid.

1. Inleiding

Is er nog iets nieuws te vertellen over zo iets eenvoudigs als de berekening van gemiddelde en standaarddeviatie? Niet veel. Maar de ervaring leert dat door programmerende statistici in het algemeen niet de numeriek meest verantwoorde berekeningswijze wordt gebruikt. Misschien komt dit omdat deze in de meeste cursussen en leerboeken niet als zodanig wordt genoemd. Zelfs in veel standaardprogrammatuur voor grote computers en zakrekenmachines wordt geen optimaal algoritme gebruikt. Greenfield en Siday laten dit zien voor een SPSS versie en een TI58**. Toch is een groot deel van de hier gepresenteerde informatie al meer dan tien jaar bekend, zoals blijkt uit de literatuurlijst. Een aantal artikelen verscheen in het door weinig statistici gelezen Communications to the ACM. Het nummer van maart 1968 bevat naast de klassieke brief van E.W. Dijkstra over de schadelijkheid van het goto statement een helaas wat minder

* Centraal Planbureau, Van Stolkweg 14, 's-Gravenhage.

** G. Stemerdink heeft aangekondigd op de Statistische dag 1981 over een uitgebreider onderzoek te zullen rapporteren.

bekende brief van A.J. van Reeken met een nuttige aanvulling op de artikelen van Welford en Neely. In het EIT rapport van Van Reeken (1970) wordt de materie nog eens in extenso beschreven. In 1974 verscheen een artikel van Ling met de resultaten van een uitgebreide simulatiestudie. De voornaamste conclusie was dat er geen algoritme bestaat dat voor alle soorten gegevens efficiënt en nauwkeurig is. Ling probeert een verband te leggen tussen de performance van een algoritme en de variatiecoëfficiënt $(\sigma/|\mu|)$. Het interessantste wat er daarna is gepubliceerd zijn de artikelen van West en van Chan & Lewis in 1979. West geeft een efficiënte en stabiele implementatie van een algoritme met weging van de waarnemingen. In het ongewogen geval is het West-algoritme gelijk aan dat van Van Reeken, terwijl de implementatie niet wezenlijk verschilt. Chan & Lewis introduceren een conditiegetal met betrekking tot de berekening van de variantie en geven de resultaten van een foutenanalyse op basis van de theorie van Wilkinson. Een aantal manieren om gemiddelde en standaarddeviatie te berekenen wordt behandeld in paragraaf 2. In paragraaf 3 wordt de nauwkeurigheid van de genoemde methoden besproken. Tot slot worden de conclusies vermeld in paragraaf 4.

2. Wijze van berekenen

2.1 Definitie

Gegeven m waarnemingen x_i. Dan zijn gemiddelde μ en standaarddeviatie σ als volgt gedefinieerd:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} x_i \qquad (1)$$

$$\sigma^2 = \frac{1}{m+c} \sum_{i=1}^{m} (x_i - \mu)^2 \qquad (2)$$

Voor c in (2) kan worden gekozen tussen:

c=0, dan is σ^2 de steekproefvariantie;

c=-1, dan is σ^2 een zuivere schatter van de populatie-variantie;

c=+1, dan is σ^2 onder alle schatters van de vorm (2) degeen met een minimaal verwacht kwadratisch verlies, bij een normaal verdeelde populatie. Zie Hogg & Craig opg. 8-13 (1970) = 6.14 (1978).

In dit artikel gebruiken we verder (2) met c=-1. Lang niet altijd is in de documentatie van standaardprogramma's en zakrekenmachines vermeld welke versie van (2) wordt toegepast.

2.2 Handberekening

Een van ouds bekende variant op de bovengenoemde formule is:

$$\mu = \alpha + \frac{1}{m} \sum_{i=1}^{m} (x_i - \alpha) \qquad (3)$$

$$\sigma^2 = \frac{1}{m-1} \left[\sum_{i=1}^{m} (x_i - \alpha)^2 - m\,(\mu - \alpha)^2 \right] \qquad (4)$$

In de praktijk van de handberekening werd deze variant veel gebruikt. Voor α werd dan een getal gekozen in de buurt van een op het oog geschat gemiddelde. Bij voorkeur koos men α zo, dat de getallen $x_i - \alpha$ gemakkelijk te kwadrateren zijn. Het spreekt vanzelf dat α intuïtief altijd zo wordt gekozen dat $|\mu - \alpha|$ hooguit van dezelfde orde is als σ, meestal zelfs kleiner.

2.3 Klassiek "eentraps" algoritme

Met de komst van de eerste rekenmachines werd formule (2) voorgoed op stal gezet, want dit is een tweetraps (two pass) algoritme. Dat wil zeggen dat eerst μ moet worden berekend (pass 1), waarna de gegevens opnieuw dienen te worden ingetoetst of ingelezen om σ te kunnen berekenen (pass 2).

Formule (4) is een eentraps (one pass) algoritme omdat de keuze van α tevoren wordt gemaakt. Mits er tenminste twee accumulatieregisters beschikbaar zijn kan nu met één keer invoeren van elke x_i tegelijk de som en de kwadraatsom worden bijgewerkt. De beste keuze voor α leek nu $\alpha=0$. Want waarom die extra aftrekking als het rekenwerk toch door de machine wordt gedaan? De formules:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} x_i \qquad (5)$$

$$\sigma^2 = \frac{1}{m-1}\left[\sum_{i=1}^{m} x_i^2 - m\mu^2\right] \qquad (6)$$

zijn de laatste decennia onder statistici gepropageerd als de enige praktisch bruikbare. Hoewel (6) inderdaad vaak bruikbaar is, is het zelden het beste en in bepaalde gevallen een uitgesproken slecht algoritme.
Als σ klein is ten opzichte van μ moet in formule 6 het verschil van twee ongeveer even grote getallen worden berekend. Het is bekend dat dit bij een eindige rekennauwkeurigheid een onnauwkeurig resultaat oplevert.

2.4 Bijwerkende algoritmes

Bijwerkende (updating) algoritmes zijn eentraps algoritmes waarbij voor iedere nieuwe waarneming het gemiddelde en de standaarddeviatie wordt bijgewerkt door berekening uit de lopende waarde en de nieuwe waarneming. Dit gebeurt als volgt:
Zij het gemiddelde na k waarnemingen

$$M_k = \frac{1}{k} \sum_{i=1}^{k} x_i$$

Dan is het gemiddelde na k+1 waarnemingen

$$M_{k+1} = \frac{1}{k+1} \sum_{i=1}^{k+1} x_i = \frac{1}{k+1} (k.M_k + x_{k+1})$$

$$= M_k + (x_{k+1} - M_k)/(k+1) \qquad (7)$$

Zij de standaarddeviatie na k waarnemingen

$$S_k^2 = \frac{1}{k-1} T_k \quad ; \quad T_k = \sum_{i=1}^{k} (x_i - M_k)^2$$

Dan geldt na k+1 waarnemingen

$$S_{k+1}^2 = \frac{1}{k} T_{k+1} \quad ; \quad T_{k+1} = \sum_{i=1}^{k+1} (x_i - M_{k+1})^2$$

Invullen van M_{k+1} uit (7) en afsplitsen van de $(k+1)^e$ term levert:

$$T_{k+1} = \sum_{i=1}^{k} \left(x_i - M_k - \frac{x_{k+1} - M_k}{k+1}\right)^2 + \left(x_{k+1} - M_k - \frac{x_{k+1} - M_k}{k+1}\right)^2$$

$$= T_k + k.(x_{k+1} - M_k)^2/(k+1) \qquad (8)$$

Het berekenen van de lopende kwadraatsom T_k kan op diverse manieren gebeuren. Chan & Lewis laten zien dat (8) de beste is. Het is de basis voor de algoritmes van Welford, Van Reeken en West.

2.5 Programmabeschrijving

Van de hierboven genoemde algoritmes wordt nu een beschrijving gegeven in een pseudo-programmeertaal (steenkoolalgol). Deze is niet bedoeld om een bestaande computer te programmeren, maar om te worden begrepen door personen die bekend

zijn met talen als ALGOL, PASCAL, FORTRAN of BASIC. De syntax spreekt hopelijk voor zich. De conventie DO...OD is ontleend aan ALGOL68 en dient om een samengesteld statement te markeren (vergelijk begin...end).

Programma 1, de definitie (formule 1 en 2)

```
SX = 0
FOR  I=1,M
     DO  SX=SX+X(I)
     OD
XM =  SX/M
T  = 0
FOR  I=1,M
     DO  D=X(I)-XM
         T=T+D*D
     OD
S2 = T/(M-1)
```

Programma 2, het klassieke eentrapsalgoritme (formule 5 en 6)

```
SX = 0
X2 = 0
FOR  I=1,M
     DO  SX=SX+X(I)
         X2=X2+X(I)* X(I)
     OD
XM = SX/M
S2 = (X2 - XM*SX)/(M-1)
```

Programma 3a, het bijwerkende algoritme volgens West (formule 7 en 8)

```
XM = X(1)
T  = 0
FOR  I=2,M
     DO  D=X(I)-XM
         R=D/I
        XM=XM+R
         T=T+(I-1)*D*R
     OD
S2 = T/(M-1)
```

Programma 3b, het bijwerkende algoritme volgens Van Reeken (formule 7 en 8)

```
XM = 0
T  = 0
FOR  I=1,M
     DO  D=X(I)-XM
        XM=XM+D/I
        XX=D*D
         T=T+XX-XX/I
     OD
S2=  T/(M-1)
```

2.6 Efficiëntie van de berekening

In tabel 1 is van de genoemde programma's het aantal delingen, vermenigvuldigingen en additieve bewerkingen opgenomen.

Tabel 1 Aantal bewerkingen per berekening van gemiddelde en variantie uit m waarnemingen

programma	/	*	+ -
1 definitie	2	m	3m+1
2 klassiek	2	m+1	2m+2
3a updating(West)	m	2m-2	4m-3
3b updating(Van Reeken)	2m+1	m	4m+1

De laatste twee algoritmes zijn duurder dan de eerste. Het klassieke eentrapsalgoritme vereist de minste bewerkingen, terwijl toepassing van de definitie nauwelijks duurder is indien alle gegevens zich in het centraal geheugen bevinden.

3. Nauwkeurigheid van de berekening

Bij zakrekenmachines en moderne computers speelt de rekentijd van dit soort eenvoudige berekeningen nauwelijks meer een rol van betekenis. Veel belangrijker is de betrouwbaarheid van het antwoord. Behalve de precisie van de computer is ook de keuze van het algoritme hierop van invloed, terwijl de aard van de gegevens eveneens een rol speelt. Ling vond in zijn simulatiestudie een verband tussen de nauwkeurigheid van het antwoord en de variatiecoëfficiënt van de gegevens. Chan en Lewis voeren een foutenberekening uit voor verschillende algoritmen. Zij karakteriseren de gegevens daarbij met een conditiegetal κ, dat de conditie van de gegevens voor berekening van de standaarddeviatie moet beschrijven. De κ is een eigenschap van de gegevens zelf, die los staat van de machineprecisie en het gekozen algoritme. Bij exacte berekening zou een relatieve onzekerheid γ in de gegevens een relatieve onzekerheid $\kappa\gamma$ in de standaarddeviatie veroorzaken. De door Chan en Lewis gedefi-

nieerde κ kan worden herschreven tot*:

$$\kappa^2 = 1 + m\,\mu^2/(m-1)\sigma^2 \backsimeq 1 + \mu^2/\sigma^2 \qquad (9)$$

Gegevens met een kleine variatiecoëfficiënt krijgen een groot conditiegetal, hetgeen overeenkomt met de bevindingen van Ling. In tabel 2 staan voor de drie eerdergenoemde algoritmen de door Chan en Lewis berekende bovengrens voor de relatieve fout in de standaarddeviatie van m waarnemingen met conditiegetal κ, berekend op een computer met relatieve machineprecisie η (η is het kleinste getal waarvoor geldt $1+\eta>1$ op een machine).

Tabel 2 Bovengrens van de relatieve fout in de standaarddeviatie volgens Chan & Lewis

programma	bovengrens van de relatieve fout
1 definitie	$2\kappa\eta + (\frac{m}{2} + 1)\eta$
2 klassiek	$(\frac{3}{2} m + 1)\kappa^2\eta + \eta$
3 updating (West)	$(\frac{\sqrt{2}}{3} m + 7\sqrt{m + 1})\kappa\eta + (\frac{m}{2} + 2)\eta$

Hoewel het hier om bovengrenzen gaat kunnen we wel concluderen dat in het algemeen:

- de onnauwkeurigheid evenredig is met de machineprecisie η. Overgang op dubbele precisie geeft dus een nauwkeuriger resultaat;
- de relatieve fout ten gevolge van rekenonnauwkeurigheid groter wordt naarmate het aantal waarnemingen toeneemt;

*Als $\kappa\gamma$ de bovengrens is voor de fout in σ is (9) juist, hoewel de door Chan & Lewis gegeven afleiding mijns inziens niet geheel correct is. Ik neem aan dat de veronderstelde denkfout in de afleiding geen invloed heeft op de resultaten in tabel 2 aangezien de formule voor κ wel goed is.

- bij gegevens met een kleine variatiecoëfficiënt het klassieke eentrapsalgoritme minder betrouwbaar is (fout evenredig met conditie in het kwadraat).

Deze conclusies worden ook bevestigd in simulatie-experimenten (zie Ling, Chan & Lewis, Greenfield & Siday). We geven in tabel 3 één voorbeeld uit Chan en Lewis. Er worden 20 steekproeven, elk met 100 waarnemingen, getrokken uit $N(1,\sigma)$. De standaarddeviaties worden met de drie algoritmes berekend op een DEC 10 ($\eta \approx 10^{-8}$). Het aantal correcte significante cijfers wordt gemiddeld over de 20 steekproeven.

Tabel 3 Gemiddeld aantal correcte significante cijfers van S^2

programma	σ=	.1	.01	.001	.0001
1 definitie		7.7	7.8	7.8	7.3
2 klassiek		5.5	3.4	1.4	0.0
3 updating (West)		7.6	6.7	5.7	4.8

Dit effect kan ook worden gedemonstreerd aan de hand van een eenvoudig getallenvoorbeeld. Bereken de standaarddeviatie van de drie getallen 7.01, 7.02, 7.03. Bij gebruik van het klassieke algoritme wordt $\Sigma x_i^2 = 147.8414$ en $m\mu^2 = 147.8412$. Wanneer wordt gerekend met 6 significante cijfers is het resultaat echter $\Sigma x_i^2 = 147.841$ en $m\mu^2 = 147.841$. Men vindt dan dus $\sigma=0$. De andere twee algoritmes leveren bij berekening met 6 cijfers wel correct $\sigma=0.01$. Het mag verbazing wekken dat bij zo'n eenvoudige opgave een berekening met 6 significante cijfers een antwoord met 0 significante cijfers oplevert.

4. Conclusies

De keuze van een algoritme voor de berekening van gemiddelde en standaarddeviatie wordt mede bepaald door de aard van de gegevens. Als de variatiecoëfficiënt niet te klein

is geven alle besproken methoden acceptabele resultaten. Voor een gebruiker verdient het dus aanbeveling ervoor te zorgen dat hij geen gegevens met een kleine variatiecoëfficiënt aan een (zak)rekenmachine aanbiedt.
De ontwerpers van een statistisch pakket of een zakrekenmachine weten echter niet van te voren welke gegevens een gebruiker zal aanbieden. Zij dienen daarom een algoritme te kiezen dat bij alle soorten gegevens een zo betrouwbaar mogelijk antwoord geeft. Wanneer de gegevens in een centraal geheugen ter beschikking staan is het tweetrapsalgoritme volgens de definitie een efficiënte en betrouwbare keuze. Indien de gegevens met de hand moeten worden ingevoerd (zoals bij een zakrekenmachine) of van een extern geheugen worden gelezen (zoals bij veel standaardpakketten) is een updating algoritme nodig. De versie van West zoals beschreven in programma 3a is dan aan te bevelen.

Literatuur

1. Chan, T.F.C. and Lewis, J.G. Computing Standard Deviations: Accuracy. Comm. ACM 22 (1979) p.526-531.

2. Greenfield, T and Siday, S. Statistical Computing for Business and Industry. The Statistician 29 (1980) p.33-55.

3. Hogg, R.V. and Craig, A.T. Introduction to Mathematical Statistics. Collier McMillan (1970).

4. Ling, R.F. Comparison of several algorithms for computing means and variances. J. Amer. Statistical Assoc. 69 (1974), p.859-866.

5. Neely, P.M. Comparison of several algorithms for computation of means, standard deviations and correlation coëfficiënts. Comm. ACM 9 (1966) p.496-499.

6. Van Reeken, A.J. Dealing with Neely's Algorithms. Comm. ACM 11 (1968) p.149.

7. Van Reeken, A.J. The effect of truncation in statistical computation. E.I.T. Research Memorandum No 10. Tilburg, 1970.

8. Welford, B.P. Note on a method for calculating corrected sums of squares and products. Technometrics 4 (1962) p.419-420.

9. West, D.H.D. Updating Mean and Variance estimates: An improved method. Comm. ACM 22 (1979) p.532-535.