



UPPSALA
UNIVERSITET

Workout

Uppgifter till Beräkningsvetenskap II

2020

WORKOUT



Inledning

Kursen Beräkningsvetenskap II innehåller HT 2018 tre workout-pass. Syftet med dem är att du i gruppen eller på egen hand skall arbeta igenom ett antal övningar med papper och penna. Arbetet med workout-uppgifterna, med möjlighet till hjälp från lärare, ska ge dig tillfälle att bearbeta det stoff som föreläsningarna har behandlat. Närvaro vid workout-passet rekommenderas därför starkt.

Skriftliga redovisningar av lösningar till workout-uppgifterna är obligatoriska och utgör en del av examinationen. Den som är närvarande under workout-passet redovisar direkt till läraren. Den som inte kan närvara får lämna sin redovisning senast dagen efter workout-passet. Den som lämnar in för sent utan godtagbart skäl får ytterligare uppgifter att redovisa.

Uppgifterna löses gruppvis av *miniprojektgrupperna*. Det är viktigt att alla i gruppen har förstått lösningarna. *Har* man förstått, så kommer man att ha goda chanser att klara tentamen.

Som student kan du använda workout-uppgifterna som underlag för en *självdiagnos*. Om du *inte* förstår en lösning som gruppen kommit fram till, bör du läsa på mer om det avsnittet och be lärare eller medstudenter förklara. Lägg inte en workout-uppgift åt sidan förrän du tycker att du verkligen förstår den och kan lösa den (och andra av samma typ) på egen hand.

OBS! Workout-övningarna är tillrättalagda så att de går att genomföra med handräkning. De är enbart tänkta som ett stöd för studenternas lärande och skall *inte* uppfattas som exempel på verkliga tillämpningar. I de tillämpningar som beräkningsmetoderna egentligen är tänkta för krävs datorer för att genomföra beräkningarna inom rimlig tid. Detta kommer miniprojekten att ge exempel på.

Ordinära differentialekvationer, del 1

Uppvärmning – obligatorisk

1. Ställ upp Eulers framåt-differensmetod för differentialekvationen

$$y'(t) - y(t) - t = 0, \quad y(0) = 0$$

och gör två steg med steglängd 0.1.

Medel – obligatorisk

2. Ställ nu upp den klassiska Runge-Kuttas metod för problemet ovan. Genomför *ett* steg med steglängd 0.1.
3. En enkel modell för utvecklingen av en bakteriepopulation ges av

$$y' = Cy \left(1 - \frac{y}{B}\right), \quad y(0) = \alpha, \quad (1)$$

där $y(t)$ betecknar antalet bakterier vid tidpunkt t . Konstanten C beskriver hur snabbt populationen växer. Det finns därtill en övre gräns B för hur många bakterier som kan samexistera på givet utrymme. Inledningsvis antas populationens storlek vara $\alpha \ll B$.

Formulera Heuns metod för problemet ovan. Genomför sedan ett steg med metoden för fallet $C = 10$, $B = 1000$, $\alpha = 1$, där steglängden h väljs till 0.1.

4. Med ordinära differentialekvationer kan man beskriva hur en satellit rör sig kring en planet. Låt $x(t)$ och $y(t)$ beteckna x - respektive y -koordinaten för satellitens position vid tidpunkten t . Planetens position är i origo. Då kan man ställa upp följande modell för hur satellitens position förändras med tiden:

$$x''(t) = -k \frac{x(t)}{\left(\sqrt{x(t)^2 + y(t)^2}\right)^3} \quad (2)$$

$$y''(t) = -k \frac{y(t)}{\left(\sqrt{x(t)^2 + y(t)^2}\right)^3}. \quad (3)$$

Låt oss här för enkelhets skull anta att $k = 1$. Givna begynnelsevärden vid $t = 0$ är satellitens position $(x(0), y(0))$ och dess hastighet i x - respektive y -riktningen $(x'(0), y'(0))$.

- (a) Formulera om modellen ovan till ett system av första ordningens differentialekvationer.
- (b) Beskriv hur det skulle gå till att använda denna modell för att simulera satellitens rörelse kring planeten för $0 \leq t \leq 100$ med hjälp av Matlab-funktionen `ode45`, se nedanstående hjälptext. Du behöver inte skriva ett komplett program, utan det räcker med en programskiss som visar hur den matematiska modellen skulle representeras i programmet och hur `ode45` sedan skulle anropas.

```
ODE45 Solve non-stiff differential equations, medium order method.
[TOUT,YOUT] = ODE45(ODEFUN,TSPAN,YO) with TSPAN = [T0 TFINAL]
integrates the system of differential equations y' = f(t,y)
from time T0 to TFINAL with initial conditions YO. ODEFUN is a
function handle. For a scalar T and a vector Y, ODEFUN(T,Y) must
return a column vector corresponding to f(t,y). Each row in the
solution array YOUT corresponds to a time returned in the column
vector TOUT.
```

Intensiv

5. Implementera Eulers framåt-differensmetod i Matlab. Resultatet skall bli en Matlab-funktion som kan användas för att lösa skalära begynnelsevärdesproblem på formen $y'(t) = f(t, y(t))$. Första raden i funktionen skall vara:

```
function y = euler(f,a,b,y0,n)
```

Inparametern **f** skall vara en Matlab-funktion som beskriver högerledet i ODE-problemet. Värdena **a** och **b** anger begynnelsetid respektive sluttid och **y0** är begynnelsevärdet. Den sista inparametern, **n**, anger antalet tidssteg.

Utparametern **y** är en vektor, där $y(k)$ är den beräknade, approximativa lösningen i tidssteg k , $k = 1, \dots, n$.

Det gör inget om du inte kommer ihåg i detalj hur olika Matlab-kommandon ser ut, så länge programmet i princip är korrekt.

6. Funktionen `euler`, som du skrev ovan, är avsedd för *skalära* begynnelsevärdesproblem. Vilka förändringar skulle krävas i `euler` för att även *system* av första ordningens differentialekvationer skulle kunna hanteras?
7. Du arbetar i ett konsultföretag och har fått i uppdrag att med datorns hjälp simulera förloppet när ett föremål vinschas in till en helikopter. Föremålet hänger i en vajer under helikoptern. När föremålet vinschas in, så kommer linan att pendla fram och tillbaka under helikoptern. När du modellerar förloppet väljer du därför att betrakta linan med vidhängande föremål som en pendel. På grund av invinschningen kommer pendelns längd att variera med tiden. En matematisk modell för en sådan pendel är:

$$\theta''(t) + 2\frac{r'(t)}{r(t)}\theta'(t) + \frac{g}{r(t)}\sin(\theta(t)) = 0$$

där $\theta(t)$ är pendelns vinkelutslag, $r(t)$ är pendelns längd och g är gravitationskonstanten. För beskrivning av själva invinschningen specificeras även en "invinschningsstrategi" i form av en modell för hur $r'(t)$ varierar med tiden, vilket ger ytterligare en differentialekvation

$$r'(t) = f(t, \theta(t), \theta'(t)).$$

(Syftet med att simulera invinschningsförloppet är att man skall kunna undersöka vilken effekt *olika* invinschningsstrategier får.)

För att simulera förloppet numeriskt tänker du dig att använda Heuns metod. Beskriv, så detaljerat du kan, men *utan* att genomföra några räkningar, hur Heuns metod skulle kunna användas för att beräkna invinschningsförloppet (det vill säga för att beräkna hur θ och r varierar med tiden).

Ordinära differentialekvationer, del 2

Uppvärmning – obligatorisk

1. Ställ upp Eulers bakåtdifferensmetod för differentialekvationen

$$y'(t) + 1000 \cos(y(t)) - 100t = 0, \quad y(0) = -0.5$$

och utför *ett* steg med steglängd 0.01. Som startgissning för newtoniterationen används exempelvis det senaste y -värdet. Sätt toleransen till två korrekta decimaler.

Medel – obligatorisk

2. Differentialekvationen ovan är styv.
 - (a) Förklara innebörden i begreppet ”styv differentialekvation”.
 - (b) Förklara varför det skulle ha varit mindre lämpligt att lösa ekvationen ovan med Eulers framåtdifferensmetod.
 - (c) På vilket sätt är det fördelaktigt att använda en metod med adaptivt steglängdsval när man löser en styv differentialekvation numeriskt?
3. När man analyserar numeriska metoders stabilitetsegenskaper kan man använda den sk testekvationen. Man tittar då på fallet då realdelen av λ är negativ (det garanterar att den underliggande ODE:n har en avtagande lösning).
Analysera stabiliteten hos Eulers framåtdifferensmetod. Vilket blir stabilitetsvillkoret?
4. En numerisk metod sägs vara *ovillkorligt stabil* om den är stabil (avtagande) för alla värden på h när realdelen av $\lambda < 0$.
Visa nu att Euler bakåt (implicit Euler) är ovillkorligt stabil. För enkelhets skull kan du i den här uppgiften begränsa analysen till reella värden på λ .
5. Till differentialekvationen

$$\begin{aligned} y'(t) &= f(t, y(t)), & t \geq a \\ y(a) &= c \end{aligned}$$

kan man ställa upp differensmetoden

$$y_{k+1} = y_k + h \cdot f(t_k, y_k)$$

Vad är den gängse tolkningen av de värden som anges som y_k resp. $y(t_k)$?
Hur definieras det *lokala trunckeringsfelet* för metoden?
Använd Taylorutveckling för att avgöra hur det lokala trunckeringsfelet (och det globala trunckeringsfelet) beror av h .

Intensiv

6. Vilket blir stabilitetsvillkoret för Heuns metod tillämpad på ekvationen $y'(t) = \lambda y(t)$?

7. Ett begynnelsevärdesproblem löses med en numerisk metod som vi här betecknar med M . Tre olika steglängder används: h , $h/2$ och $h/8$. I tabellen nedan visas begynnelsevärdet $y(0)$ i första raden. Kolumnen längst till vänster visar resultaten av de första tio stegen med metoden M för steglängd h . De två kolumnerna till höger visar resultaten vid samma tidpunkter men för steglängd $h/2$ respektive $h/8$.

h	$h/2$	$h/8$
10.000000	10.000000	10.000000
11.056879	11.056880	11.056880
12.235428	12.235430	12.235430
13.548443	13.548446	13.548446
15.010066	15.010071	15.010071
16.635927	16.635933	16.635933
18.443297	18.443306	18.443306
20.451267	20.451279	20.451279
22.680935	22.680949	22.680950
25.155615	25.155633	25.155634
27.901077	27.901098	27.901100

Du ska nu använda dessa resultat för att ta reda på vilken noggrannhetsordning metoden M har.

Ledning: För att bestämma noggrannhetsordningen kan du ta reda på hur felet förändras då steglängden minskar från h till $h/2$. För att uppskatta felet för dessa två steglängder kan du i uttrycket för felet ersätta den exakta lösningen med den lösning som erhöles för $h/8$.

8. Smittspridning i en population kan modelleras med följande differentialekvation:

$$P' = (k + 0.1 \sin(t)) P (C - P),$$

där $P(t)$ representerar antalet infekterade individer vid tidpunkten t och konstanten C är det totala antalet individer i populationen. Parametern k representerar risken för att bli smittad och sinusfunktionen modellerar exempelvis årstidsbunden variation i mottaglighet för smitta.

- (a) Låt $g(t, P)$ beteckna högerledet i differentialekvationen ovan. En hel familj av numeriska metoder för lösning av ekvationen kan då uttryckas genom:

$$P_{k+1} = P_k + h (c_1 g(t_k, P_k) + c_2 g(t_{k+1}, P_{k+1}))$$

Varje specifikt val av värden på c_1 och c_2 ger en specifik metod i denna familj. För vilka värden på c_1 och c_2 får man **(i)** en explicit metod, respektive **(ii)** en implicit metod.

- (b) Vilka värden på c_1 respektive c_2 ger en metod med noggrannhetsordning större än eller lika med 1? Vilka värden på c_1 och c_2 ger högst noggrannhetsordning?
9. Antag att du skall lösa ett ODE-problem med funktionen `euler` och att du väljer ett n -värde som gör att steglängden blir h_0 . Om det visar sig att detta val av steglängd inte ger tillräckligt *noggrann* lösning, så finns två alternativ:
- (a) Minska steglängden till ch_0 , där $0 < c < 1$, så att Euler framåt ger tillräcklig noggrannhet.
- (b) Behåll steglängden h_0 , men byt till en numerisk metod med högre noggrannhetsordning än Euler framåt. Vi antar att metoden med högre noggrannhetsordning ger tillräcklig noggrannhet för $h = h_0$ och att båda metoderna är stabila för $h = h_0$.

Det bästa alternativet är det som ger kortast exekveringstid för att beräkna lösningen med önskad noggrannhet.

Din uppgift nu är att utreda valet mellan dessa två alternativ när metoden med högre noggrannhetsordning är Heuns metod. Hur litet måste c vara för att det skall löna sig att övergå till Heuns metod med steglängd h_0 i stället för att använda Eulers metod med steglängd ch_0 ? (OBS! Det går inte att komma fram till ett exakt värde på c , utan det räcker med en mera grovkornig analys.)

Monte Carlo-metoder

Uppvärmning – obligatorisk

1. Förklara kort skillnaden mellan en deterministisk och stokastisk *modell*, samt skillnaden mellan en stokastisk och deterministisk *metod*.
2. Förklara i ord hur och varför en Monte Carlo-metod för att räkna ut värdet på en integral fungerar. Diskutera i vilka situationer en Monte Carlo-metod kan vara att föredra framför en mer ”traditionell” deterministisk metod.

Medel – obligatorisk

3. I kursen Beräkningsvetenskap I ingick workout-uppgifter där du beräknade integralen

$$I = \int_0^1 x^2 e^x dx$$

numeriskt med trapetsformeln respektive Simpsons formel. Steglängd var $h = 1/4$, vilket innebär att 5 stycken x -värden användes i beräkningen av integralen. Syftet var att förstå hur dessa algoritmer fungerar. Därför använde vi ovanstående testproblem som har känd lösning ($I = e - 2$) så att vi också kunde beräkna hur stort felet blev i de approximationer vi fick med de nämnda metoderna. Det visade sig att trapetsformeln i detta fall beräknade I med absolut fel ca -0.04 medan motsvarande fel för Simpsons formel blev ca -0.0006 .

Beräkna nu en approximation av I med den Monte Carlo-metod som beskrivs i kompendiet. Använd 5 stycken x -värden. Jämför den beräknade approximationen med det exakta svaret. Hur stort blev det absoluta felet? Jämför med felet för trapetsformeln och Simpsons formel.

Om du får tid kan du upprepa Monte Carlo-beräkningen. Du kommer ju då att få en annan approximation än första gången, eftersom beräkningspunkterna slumpas fram.

4. Som modell för hur en akties pris varierar i tiden används ofta en s.k. stokastisk differentialekvation (SDE) (den del som beskriver ”slumpen” i aktiekursen involverar samma ”Brownian motion” som ni såg i laborationen). Antag nu att ni får en Matlabfunktion

```
function x = f(x0,T)
```

som simulerar 10 stycken aktier enligt den stokastiska SDE modellen från något startvärde x_0 till tiden T . Returvärdet x är priset vid tiden T på de olika aktierna, och alltså en array med 10 element.

- (a) Skriv ett matlabskript som med Monte Carlo beräknar det förväntade totala värdet på portföljen vid tiden T .
 - (b) I Hellanders kompendium beskrivs hur man kan beräkna ett konfidsintervall för det resultat som Monte Carlo-metoden gav. Gör nu ett tillägg till ditt program så att det beräknar och skriver ut konfidsintervallet. Använd Matlabs inbyggda funktion `std` för beräkning av standardavvikelse. Anropet `std(x)` ger standardavvikelsen för talföljden i vektorn x
5. I miniprojektet kommer vi att behöva generera slumpetal från *exponentialfördelningen*

$$\begin{aligned} f(\tau; a_0) &= a_0 e^{-a_0 \tau}, \tau \geq 0 \\ &= 0, \tau < 0 \end{aligned}$$

med hjälp av algoritmen ”inverse transform sampling”. Målet är att härleda en formel för hur exponentialfördelade slumpstal τ kan beräknas givet likformigt fördelade slumpstal u .

Slå upp algoritmen i kompendiet. Genomför sedan stegen i algoritmen enligt nedan, så att du får fram den sökta formeln för τ .

- (a) Beräkna den kumulativa fördelningsfunktionen (CDF),

$$F(t) = P(\tau \leq t)$$

för exponentialfördelningen.

- (b) För ett givet tal $u \in [0, 1]$ hitta τ så att $F(\tau) = u$. Med andra ord: hitta inversen till F , det vill säga $\tau = F^{-1}(u)$.

Intensiv

6. Den Monte Carlo-metod för integrering som du använde i uppgift 2 ovan byggde på likformigt fördelade slumpstal. Föreslå nu ett sätt att approximera integralen

$$I = \int_{-a}^a x^2 e^{-\frac{x^2}{2}} dx$$

med Monte Carlo genom att använda slumpstal från en annan fördelning. Skriv ner formel/algorithm. Du kan anta att du har tillgång till slumpstal från olika fördelningar.

7. En vanlig strategi i beräkningsproblem som involverar en diskretisering, t ex numerisk integration och lösning av ordinära differentialekvationer är att adaptivt placera ut beräkningspunkterna där de behövs bäst. Analogin till detta för Monte Carlo-integrering kallas *importance sampling* där man försöker styra hur man samplar sina slumpstal så fler hamnar i områden där integranden har ett stort värde. I stället för att beräkna $I_1 = \int_{\Omega} g(x) dx$ genom att sampla x från en uniform fördelning, väljer vi en funktion $p(x)$ och beräknar $I_2 = \int_{\Omega} \frac{g(x)}{p(x)} p(x) dx$

- (a) Hur använder du Monte Carlo för att uppskatta värdet av I_2 ?
- (b) Som vi har sagt är I_1 medelvärdet av $g(X)$ om X är en likformigt fördelad slumpvariabel. Variansen i Monte Carlo-uppskattningen av integralen I_1 är proportionell mot $\sigma^2 = \int_{\Omega} (g(x) - E[g(X)])^2 dx$. Försök resonera kring hur $p(x)$ ska väljas för att variansen i en uppskattning av I_2 ska vara mindre än den för I_1 .
- (c) Vilka praktiska begränsningar har vi i valet av $p(x)$?