

BED4: Nøkkelpoenger fra pensum

Oppgavesett D1: Grafisk løsning av LP-problemer

Hvordan formulere lineære beslutningsmodeller (LP-modeller)? Avsnitt 2.6.1 i Ragsdale.

1. *Forstå problemet.*
2. *Identifiser beslutningsvariablene.*
3. *Skriv ned målfunksjonen som en lineær kombinasjon av beslutningsvariablene.*
4. *Skriv ned begrensningene som lineære kombinasjoner av beslutningsvariablene.*
5. *Bestem øvre og nedre grenser for beslutningsvariablene.*

Hvordan plote mulighetsområdet for begrensningene (sidebetingelsene) i en LP-modell med to beslutningsvariabler? Avsnitt 2.10.1-2.10.4 i Ragsdale.

1. *Lag et diagram med en akse for hver av beslutningsvariablene.*
2. *Velg en av begrensningene og plott isokvantlinjen for denne ved hjelp av den fremgangsmåten som er vist på forelesning 1 og i avsnitt 2.10.1 i Ragsdale.*
3. *Marker de punktene som oppfyller begrensningen på følgende måte:*
 - a. *For en begrensning av typen «=» vil de tillatte løsningene være de punktene som ligger på isokvantlinjen.*
 - b. *For en begrensning av typen « $\leq$ » eller « $\geq$ » vil isokvantlinjen alltid dele mulighetsområdet i to deler: de tillatte løsningene og de ikke-tillatte løsningene. Vi er interesserte i de tillatte løsningene, og vi kan enkelt finne ut hvilken side disse ligger på ved å teste ett punkt, for eksempel punktet (0,0). Marker den tillatte siden av isokvanten i diagrammet.*
4. *Gjennomfør punkt 2 og 3 for alle begrensningene.*
5. *Marker hele mulighetsområdet, det vil si de løsningene som oppfyller alle begrensningene samtidig.*

Hvordan vise en nivåkurve (isobidraglinje) for et bestemt nivå på det totale dekningsbidraget? Avsnitt 2.10.5 i Ragsdale.

1. *Anta at bare ett av produktene produseres, det vil si at vi befinner oss på en av aksene i diagrammet. Hvor mye må man produsere for å få det ønskede nivået på det totale dekningsbidraget? Plott dette punktet på aksen. Gjenta for det andre produktet, dvs plott et tilsvarende punkt på den andre aksen i diagrammet.*
2. *Vi har nå to punkter i diagrammet. Ved å trekke en rett linje gjennom dem kan vi vise alle produksjonsplaner som gir det ønskede dekningsbidraget.*

Hvordan finne den optimale løsningen ved hjelp av isobidragslinjer? Avsnitt 2.10.6 i Ragsdale.

1. *Plott en eller flere nivåkurver og legg merke til hvilken retning vi må parallellforskyve nivåkurvene for å få en bedre målfunksjonsverdi (i dette tilfellet, høyere totalt dekningsbidrag).*
2. *Parallellforskyv nivåkurven så langt som mulig i denne retningen inntil den tangerer mulighetsområdet i ett punkt (vi kan i noen tilfeller ha flere tangeringspunkter).*
3. *Finn koordinatene for tangeringspunktet ved å sette opp og løse et ligningssystem basert på isokvantene som krysser hverandre her. Dette er den optimale løsningen.*

Vi kan også finne den optimale løsningen ved å liste opp (enumerere) alle hjørnepunktene i mulighetsområdet, og så velge den hjørneløsningen som har best målfunksjonsverdi. Avsnitt 2.10.7 i Ragsdale.

Dette vil vanligvis være en svært arbeidskrevende måte å løse en LP-modell på. De fleste dataprogrammer for LP-modeller bruker Simplex-algoritmen, som finner optimal løsning ved å søke blant hjørneløsningene på en smart måte.

Spesialtilfeller for LP-modeller (avsnitt 2.11 i Ragsdale):

1. *Flere optimale løsninger (2.11.1). En LP-modell vil ha uendelig mange optimale løsninger dersom nivåkurvene er parallelle med en av sidene i mulighetsområdet.*
2. *Overflødige begrensninger (2.11.2). En begrensning er overflødig dersom den ikke er med på å bestemme mulighetsområdet, det vil si at mulighetsområdet ikke endrer seg om vi fjerner denne begrensningen.*
3. *Ubegrenset målfunksjonsverdi (2.11.3). Det er ingen grense for hvor god målfunksjonsverdien kan bli.*
4. *Ingen tillatt løsning (2.11.4). Noen ganger er det ikke mulig å finne en løsning som oppfylles alle begrensningene, det vil si at mulighetsområdet er tomt.*

Gjennomgangseksemplet i kapittel 2 har maksimering av totalt dekningsbidrag som målsetning, som i eksemplet med Strikk & Sy AS. Metodene i kapittel 2 kan imidlertid også brukes for modeller der vi skal minimere målfunksjonsverdien (for eksempel totalkostnad).

Mulighetsområdet plottes på samme måte for begge typer modeller. Den eneste forskjellen ligger i hvordan den optimale løsningen finnes ved hjelp av nivåkurvene. Når det står «forbedring av målfunksjonsverdien», betyr det enten høyere verdi (i modeller med maksimering) eller lavere verdi (i modeller med minimering). For LP-modeller med positive koeffisienter i målfunksjonen, samt ikke-negative beslutningsvariabler, vil målfunksjonsverdien bli høyere dersom vi parallellforskyver nivåkurven vekk fra origo, mens den vil bli lavere dersom vi parallellforskyver nivåkurven mot origo.

Oppgavesett D2: LP-problemer i regneark

Hvordan implementere en LP-modell i et regneark? Avsnitt 3.3 og 3.4 i Ragsdale.

1. Legg alle dataene til modellen inn i celler i regnearket på en ryddig og forståelig måte.
2. Velg hvilke celler som skal representere beslutningsvariablene (en celle for hver variabel) og marker disse.
3. Legg inn en formel for målfunksjonen i en av cellene i regnearket.
4. Legg inn, for hver begrensning i modellen, en formel som representerer venstresiden til begrensningen i en celle i regnearket.

Hvordan beregne regnearkmodellen ved hjelp av Analytic Solver? Avsnitt 3.5 og 3.6 i Ragsdale. Solver "ser" følgende elementer i regnearkmodellen:

1. Målfunksjonscellen
2. Celler for beslutningsvariablene
3. Celler for venstresidene i begrensningene, samt eventuelle cellereferanser som brukes for å vise høyresidene

Se avsnitt 3.6 i Ragsdale for en forklaring av hvordan man legger inn de tre elementene i Analytic Solver.

Viktige hensyn ved design av regnearkmodeller (avsnitt 3.8 i Ragsdale):

- Kommunikasjon - Det viktigste formålet med en regnearkmodell er å kommunisere informasjon til ledere og andre.
- Pålitelighet - Resultatene fra modellen må være korrekte og konsistente.
- Etterprøvbarehet - Brukeren av modellen må kunne følge beregningene trinn for trinn for å forstå og verifisere resultatene.
- Lett å endre - En godt utformet regnearkmodell må være lett å endre og utvide for å tilfredsstille eventuelle nye brukerbehov.

Praktiske retningslinjer for god design (figur 3.18 i Ragsdale):

- Organiser data først, og bygg deretter modellen rundt data.
- Ikke legg tall direkte inn i formler. Bruk i stedet referanser til regnearkceller der tallene ligger.
- Elementer som er logisk relaterte i modellen, for eksempel høyresider og venstresider i begrensninger, bør plasseres nær hverandre i regnearket.
- Bruk formler som kan kopieres. Det betyr at dere må være bevisste på bruken av absolutte og relative cellereferanser i formlene.
- Summer for kolonner/rader bør plasseres nær de kolonnene/radene som blir summert.
- Brukere som er vant til å lese norsk (eller engelsk) vil lese regnearket fra venstre mot høyre, og fra toppen og nedover.
- Bruk farger, skyggelegging, kantlinjer og skrivebeskyttelse av celler for å skille parametre som brukeren kan endre fra andre deler av modellen.
- Bruk tekstbokser og cellekommentarer for å dokumentere ulike elementer i modellen.

Oppgavesett D4: Sensitivitetsanalyser for LP-problemer

Hvordan generere "Answer Report" i Analytic Solver? "Answer Report" gir en kort oppsummering av løsningen på problemet. Avsnitt 4.4 i Ragsdale.

1. Løs LP-problemet på vanlig måte. Pass på at dere bruker Standard LP/Quadratic Engine til å løse modellen.
2. Klikk på "Reports", "Optimization", "Answer" under "Analytic Solver Platform"-fanen.

Hvordan generere "Sensitivity Report" i Analytic Solver? Når vi løser LP-problemer antar vi at alle koeffisientene i modellen er kjent med sikkerhet. Dette er ofte en urealistisk forutsetning. "Sensitivity Report" gir informasjon om hvor sensitiv optimalløsningen er mht. ulike modellparametere. Avsnitt 4.5 i Ragsdale.

1. Løs LP-problemet på vanlig måte. Pass på at dere bruker Standard LP/Quadratic Engine til å løse modellen.
2. Klikk på "Reports", "Optimization", "Sensitivity" under "Analytic Solver Platform"-fanen.

I tabellen "Decision Variable Cells" viser kolonnene "Allowable Increase" og "Allowable Decrease" hvor mye koeffisientene i målfunksjonen (alt annet like) kan endres før optimalløsningen endres. Hvis "Allowable Increase" / "Allowable Decrease" er null, tyder det på at problemet har flere løsninger.

En måte å finne alternative løsninger på er å legge til en begrensning som sier at målfunksjonen skal være lik optimalverdien og så maksimere eller minimere (her kan dere måtte prøve og feile litt) en av variablene som har "Allowable Increase" / "Allowable Decrease" lik null.

Hva er en skyggepris? Avsnitt 4.5.4 i Ragsdale.

Skyggeprisen til en begrensning angir hvordan målfunksjonsverdien endrer seg dersom verdien av høyresiden (RHS) øker med en enhet. Dette gjelder under forutsetning av at alle andre koeffisienter holdes konstante og at endringen av høyresiden er innenfor "Allowable Increase" og "Allowable Decrease".

Skyggeprisen til en ikke-bindende begrensning er alltid lik null. Dersom verdien av høyresiden (RHS) i en bindende begrensning endres, vil mulighetsområdet og optimalløsningen endres. For å finne ny optimalløsning, må problemet endres og løses på nytt.

I "Sensitivity Report" finnes skyggeprisene i "Constraints" tabellen, i kolonnen "Shadow Price". Her finner dere også kolonner for tilhørende "Allowable Increase" og "Allowable Decrease".

Hva er "reduisert kostnad"? Avsnitt 4.5.9 i Ragsdale.

Den reduserte kostnaden for et produkt er lik differansen mellom marginal profitt per enhet og marginalverdien av ressursene som produktet bruker. Ressursene prises med skyggeprisene.

Vi har følgende sammenhenger mellom optimalverdiene på beslutningsvariablene og optimalverdiene på redusert kostnad:

Type problem	Optimalverdi beslutningsvariable	Optimalverdi redusert kostnad
	På nedre grense	≤ 0
Maksimeringsproblem	Mellom nedre og øvre grense	$= 0$
	På øvre grense	≥ 0
	På nedre grense	≥ 0
Minimeringsproblem	Mellom nedre og øvre grense	$= 0$
	På øvre grense	≤ 0

I "Sensitivity Report", finner vi "Reduced Cost" i en av kolonnene i tabellen "Decision Variable Cells".

Hvordan lage et "Spider Plot"? Avsnitt 4.7 i Ragsdale. Et "Spider Plot" viser hvordan optimalverdien i en resultatcelle endrer seg når vi gjør endringer i ulike inputceller, en om gangen.

1. Bestem hvilken resultatcelle dere vil se effekten på, for eksempel en beslutningsvariabel eller målfunksjonen.
2. Bestem hvilke inputceller / parametere som skal varieres. Kopier utgangsverdiene til nye celler.
3. Bestem variasjonsbredden for inputcellene, for eksempel mellom 90 % og 110 % av utgangspunktet. Bruk funksjonen " $\text{PsiOptParam}(\text{minimumsverdi}; \text{maksimumsverdi}; \text{utgangspunkt})$ " i de opprinnelige inputcellene, der "utgangspunkt" refererer til utgangsverdiene i inputcellene.
4. Bestem hvor mange scenarier dere vil beregne. Med eksempelvis 11 scenarier mellom 90 % og 110 % av utgangsverdiene, løses LP-problemet med 90 %, 92 %, ..., 110 % av utgangsverdien i hver av inputcellene. Med 3 inputceller vil dere da løse 33 LP-problemer for å lage et Spider Plot.
5. Under "Analytic Solver Platform"-fanen, velg "Charts", "Multiple Optimizations", "Parameter Analysis".
6. I dialogboksen som kommer frem, velg resultatcelle ("Result Cells") og inputceller ("Parameters"). Velg så "Vary All Selected Parameters One at a Time" og antall scenarier ("Major Axis Points"). (Se også Ragsdale side 158-160).

Informasjonen i "Sensitivity Report" kan tolkes på lignende måte også i minimeringsproblemer.

Ulike programpakker kan ha ulike konvensjoner når det gjelder fortegn på skyggepriser etc. Det kan derfor være nyttig å tenke gjennom hva en endring i høyresidene gjør med mulighetsområdet. Hvis mulighetsområdet blir større, vil målfunksjonen bli bedre (eller minst like god). I et maksimeringsproblem betyr det at målfunksjonsverdien øker, i et minimeringsproblem at den minker. Omvendt hvis høyresideendringen gir et mindre mulighetsområde.

Oppgavesett D5: Heltallsvariabler og enten/eller-beslutninger

Hvordan formulere heltallsbegrensninger i LP problemer? Avsnitt 6.0 og 6.1 i Ragsdale.

Når noen eller alle beslutningsvariablene i et LP-problem er begrenset til å være heltallige ("integer"), så kalles problemet et lineært heltallsproblem, eller "integer linear programming (ILP) problem". Variabler som ikke trenger å være heltallige kalles kontinuerlige ("continuous"). Hvis en variabel skal være ikke-negativ og heltallig, så kan dette formuleres på følgende måte:

$$X_i \geq 0 \text{ og heltall}$$

eller

$$X_i \geq 0$$

X_i må være heltall

Hva er en LP-relaksasjon ("LP relaxation") og hva er relasjonen mellom optimal målfunksjonsverdi i et heltallsproblem og LP-relaksasjonen? Avsnitt 6.2-6.4 i Ragsdale.

LP-relaksasjonen til et lineært heltallsproblem finnes ved å relaksere eller se bort fra heltallsbegrensningene. Alle tillatte heltallsløsninger er også tillatt i LP-relaksasjonen.

Optimalløsningen til heltallproblemet kan derfor aldri ha en målfunksjonsverdi som er bedre enn optimalløsningen til LP-relaksasjonen.

- I et maksimeringsproblem er den optimale målfunksjonsverdien til LP-relaksasjonen en øvre grense ("upper bound") for målfunksjonsverdien til heltallproblemet. [En løsning som er tillatt i heltallproblemet gir en nedre grense ("lower bound") for optimalløsningen.]
- I et minimeringsproblem er den optimale målfunksjonsverdien til LP-relaksasjonen en nedre grense ("lower bound") for målfunksjonsverdien til heltallproblemet. [En løsning som er tillatt i heltallproblemet gir en øvre grense ("upper bound") for optimalløsningen.]

Hvis LP-relaksasjonen ikke gir en heltallsløsning, kan beslutningsvariablene avrundes oppover eller nedover (trunkering). Avrunding vil typisk gi løsninger som ikke er tillatt eller ikke er optimale.

Hvordan løse lineære heltallsproblemer i Analytic Solver? Avsnitt 6.6 og 6.7 i Ragsdale.

Når dere har definert beslutningsvariable, målfunksjon og øvrige begrensninger i modellen, spesifiser hvilke beslutningsvariabler som skal være heltallige på følgende måte:

1. Marker cellene med beslutningsvariabler som skal være heltallige.
2. Klikk på pluss-tegnet under "Model"-fanen. Velg "Add Constraint".
3. Velg "int" i nedtrekksmenyen i dialogboksen. OK.

(Det er også mulig å bruke "Constraints"-knappen.)

Når Analytic Solver løser et heltallsproblem, hender det at algoritmen stopper før optimalløsningen er funnet. Det kan skje hvis "Integer Tolerance" er satt forskjellig fra 0 (se under "Engine", "Integer", "Integer Tolerance"). Hvis "Integer Tolerance" for eksempel er satt til 0,05 (5%), vil algoritmen stoppe dersom den har funnet en tillatt løsning som er mindre enn 5% fra optimalløsningen (dere vil da kunne se meldingen "Integer solution found within tolerance."). Dere kan sette "Integer Tolerance" til 0 for å være sikre på at det er optimalløsningen som finnes. (Merk! Avsnitt 6.6 i Ragsdale beskriver hvorfor det kan være nyttig med slike stoppe-regler.)

Hva er binære variabler og hvordan kan de brukes i modellering? Avsnitt 6.10 og 6.11 i Ragsdale.

En binær variabel ("binary variable") kan bare ha verdi lik 0 eller 1 og er nyttig for å modellere enten / eller-beslutninger. Hvis en beslutningsvariabel X_i er binær, uttrykkes dette ofte på følgende måte:

$$X_i = 0/1$$

eller

X_i må være binær

I Analytic Solver kan dere spesifisere at en variabel skal være binær på følgende måte:

1. Marker cellene med beslutningsvariabler som skal være binære.
2. Klikk på pluss-tegnet under "Model"-fanen. Velg "Add Constraint".
3. Velg "bin" i nedtrekksmenyen i dialogboksen. OK.

(Det er også mulig å bruke "Constraints"-knappen.)

Hvordan kan binære variabler brukes til å modellere logiske sammenhenger? Avsnitt 6.12 i Ragsdale.

La X_i være en binær variabel som har verdi lik 1 hvis prosjekt i velges og 0 ellers. Noen eksempler på begrensninger som uttrykker logiske sammenhenger:

Prosjekt 1 forutsetter at prosjekt 2 gjennomføres: $X_1 \leq X_2$

Prosjekt 1 forutsetter at prosjekt 2 gjennomføres og vice versa: $X_1 = X_2$

Prosjekt 1 og 2 er gjensidig utelukkende prosjekter: $X_1 + X_2 \leq 1$

Maksimalt k av n prosjekter kan gjennomføres: $\sum_{i=1}^n X_i \leq k$

Akkurat k av n prosjekter kan gjennomføres: $\sum_{i=1}^n X_i = k$

Hvordan bruke heltallsvariable og "big M" til å modellere for eksempel faste kostnader? Avsnitt 6.14 i Ragsdale.

Hvis en variabel, $X \geq 0$, skal knyttes til en binær variabel, Y , slik at

$$Y = \begin{cases} 1 & \text{hvis } X > 0 \\ 0 & \text{hvis } X = 0 \end{cases} ,$$

kan vi bruke "store-M" / "big M"-metoden for å etablere sammenhengen mellom X og Y . Følgende begrensning kobler sammen X og Y :

$$X \leq MY \text{ eller } X - MY \leq 0 ,$$

hvor M er et stort positivt tall, som er en øvre grense for X .

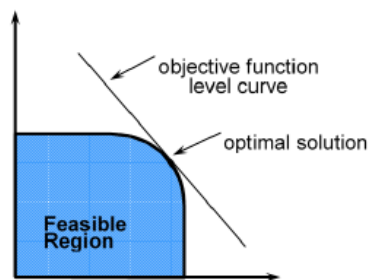
I praksis blir problemer med "big M" enklere å løse dersom M holdes så liten som mulig. M må være stor nok til ikke å begrense X når $Y = 1$. Ofte kan dere finne implisitte øvre grenser for X ut fra begrensninger i modellen eller andre egenskaper ved problemet.

Oppgavesett D7: Ikke-lineære beslutningsmodeller (NLP-problemer)

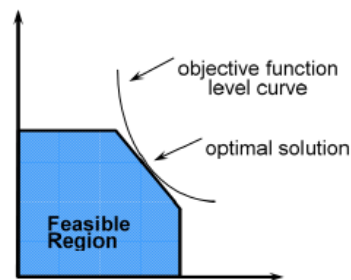
Typer av beslutningsmodeller (avsnitt 8.1 i Ragsdale):

- a) Lineær målfunksjon, lineære begrensninger.
- b) Lineær målfunksjon, ikke-lineære begrensninger.
- c) Ikke-lineær målfunksjon, lineære begrensninger.
- d) Ikke-lineær målfunksjon, ikke-lineære begrensninger.

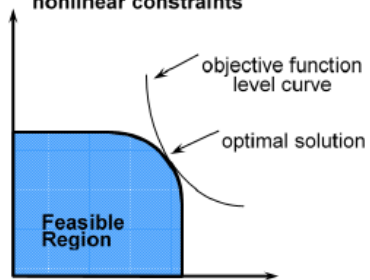
Kategori a) er den vi kjenner fra kapittel 2-4 (LP), mens b)-d) kalles NLP-modeller (Non-Linear Programming). Optimale løsninger for NLP-modeller ligger ikke nødvendigvis i hjørnepunkter, slik som for LP-modeller.



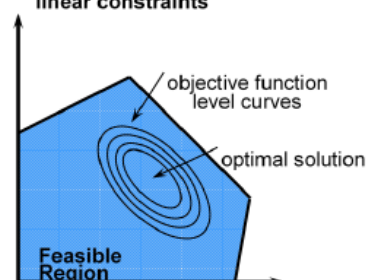
linear objective,
nonlinear constraints



nonlinear objective,
linear constraints



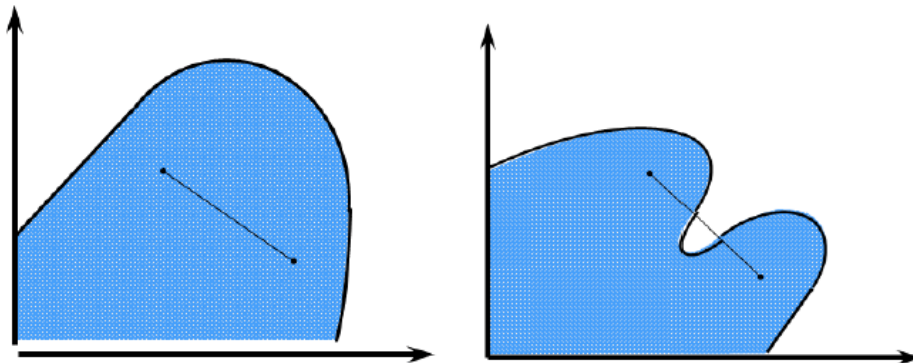
nonlinear objective,
nonlinear constraints



nonlinear objective,
linear constraints

Lokale versus globale løsninger (avsnitt 8.3 i Ragsdale). En løsning er et **lokalt optimal** dersom den er bedre enn alle løsninger som ligger i nærheten. En lokalt optimal løsning trenger ikke nødvendigvis være **globalt optimal**, det vil si at det kan være løsninger andre steder i mulighetsområdet som er bedre.

Når vi skal sjekke om en løsning er globalt optimal hjelper det å vite om mulighetsområdet er konvekt. Et område er **konvekt** dersom enhver rett linje som trekkes mellom to punkter i området i sin helhet ligger inne i området (venstre figur nedenfor). Et område er ikke konvekt dersom vi kan trekke en rett linje mellom to punkter i området slik at ikke hele linjen ligger innenfor området (høyre figur nedenfor).



Dersom en optimeringsmodell med maksimering (minimering) har et konvekt mulighetsområde, og dersom målfunksjonen er konkav (konveks), sier vi at modellen er konveks. For konvekse modeller kan vi vite helt sikkert at en optimal løsning også er globalt optimal.

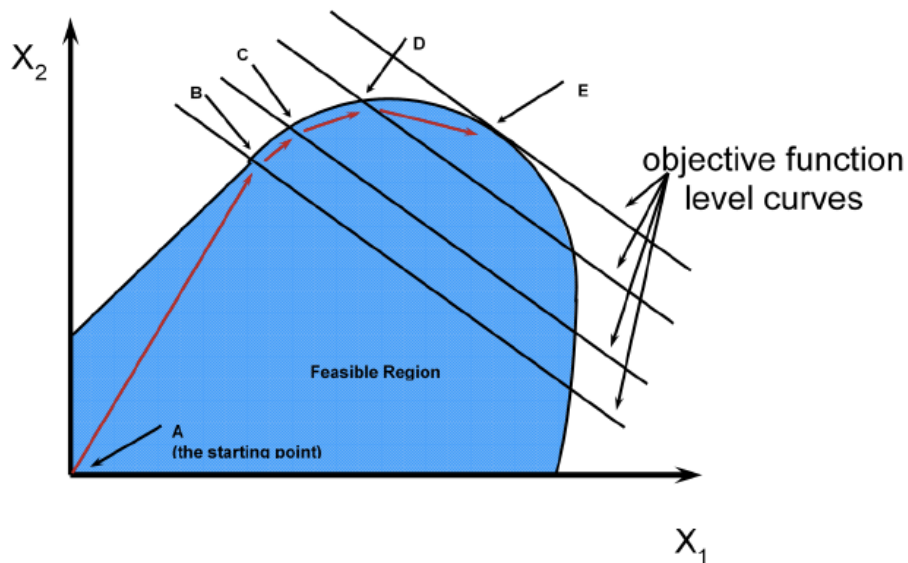
For å avgjøre om målfunksjonen oppfyller kravet kan vi se på nivåkurvene. Dersom nivåkurvene lager et konvekt område på den siden som gir bedre verdi, er kravet oppfylt.

Legg merke til at LP-modeller er konvekse, siden mulighetsområdet er konvekt (avgrenset av rette linjer) og nivåkurvene er rette linjer.

Det er også mulig å teste for konveksitet i Analytic Solver ved å velge «Optimize» -> «Analyze original/transformed model», eller ved å klikke på knappen for «Analyze without solving» i fanen «Model». Det er ikke alltid at Analytic Solver klarer å avsløre at en modell er konveks.

GRG-algoritmen (Generalized Reduced Gradient) for løsning av NLP-modeller (avsnitt 8.2 i Ragsdale):

1. Velg et **startpunkt** som ligger i mulighetsområdet. I *Analytic Solver* vil startpunktet være den løsningen som ligger i regnearket når beregningen startes.
2. Er det mulig å bevege seg i en lovlig retning (innenfor mulighetsområdet) og samtidig forbedre målfunksjonsverdien (dvs gå til en nivåkurve med bedre verdi)?
 - a. Dersom svaret er nei konkluderer algoritmen med at den har funnet en optimal løsning. Avslutt.
 - b. Dersom svaret er ja beregnes den **maksimale steglengden**, dvs hvor langt man kan gå i den valgte retningen uten å forlate mulighetsområdet. Gå til det nye punktet og repeter trinn 2.



Valg av startpunkt for NLP-modeller (avsnitt 8.3 i Ragsdale):

- *For NLP-modeller kan vi få en lokalt optimal løsning som ikke nødvendigvis er globalt optimal.*
- *Hvilken lokalt optimal løsning vi får avhenger av hvilket startpunkt vi velger.*

Uansett er det lurt å unngå null som startpunkt. Det kan gjøre at Analytic Solver ikke kommer videre fordi den ikke klarer å beregne hvilken retning det er gunstig å gå.

Når Solver stanser får vi en av disse tre meldingene:

1. *“Solver found a solution. All constraints and optimality conditions are satisfied.”*
I de fleste tilfeller betyr dette at Solver har funnet et lokalt optimum. Dersom modellen er konveks, er dette som oftest også et globalt optimum. Merk at noen ganger er løsningen hverken globalt eller lokalt optimal, for eksempel dersom startpunktet var lik null.
2. *“Solver has converged to the current solution. All constraints are satisfied.”*
Målfunksjonsverdien endret seg veldig lite i de siste iterasjonene, så Solver har valgt å avslutte beregningene. Vi har en tillatt løsning, men vi har ingen garanti for at den er optimal (lokalt eller globalt).
3. *“Solver cannot improve the current solution. All constraints are satisfied.”*
Modellen er degenerert og Solver itererer mellom de samme løsningene hele tiden. Dette kan ofte løses ved å eliminere unødvendige begrensninger i modellen.

Porteføljevalgproblemer (avsnitt 8.9 i Ragsdale).

Forventet avkastning for porteføljen = $\sum_{i=1}^n \mu_i p_i$

Varians for porteføljen = $\sum_{i=1}^n \sigma_i^2 p_i^2 + 2 \sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} p_i p_j$

der

p_i = andel av porteføljen som består av investering i

μ_i = forventet avkastning for investering i

σ_i^2 = varians av avkastningen til investering i

$\sigma_{ij} = \sigma_{ji}$ = kovarians mellom investering i og j

Variansen for porteføljen kan også beregnes ved hjelp Excel-formelen

$SUMPRODUCT(MMULT(p;C);p)$

der

$p = [p_1 \ p_2 \ \dots \ p_n]$ er det området i regnearket der porteføljevektene ligger og

$C = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1n} \\ \sigma_{21} & \sigma_2^2 & \dots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_n^2 \end{bmatrix}$ er området der variansene og kovariansene ligger.

Se eksemplet i avsnitt 8.9.4 i Ragsdale¹.

Sensitivitetsrapport for NLP-problemer (avsnitt 8.10 i Ragsdale).

Sensitivitetsrapporten for NLP-problemer kan hentes fram på samme måte som tilsvarende rapport for LP-problemer (se kapittel 4), nemlig ved å velge «Reports» -> «Optimization» -> «Sensitivity» etter at Analytic Solver er ferdig med beregningene.

Rapporten for NLP-problemer minner om den tilsvarende rapporten for LP-problemer. Lagrange-multiplikatorene i NLP-rapporten tilsvarer skyggeprisene i LP-rapporten. Det angis imidlertid ikke noe gyldighetsområde for Lagrange-multiplikatorene, og de kan i utgangspunktet bare brukes til å beregne effekten på målfunksjonsverdien av å gjøre små endringer i høyresidene til begrensningene.

Tilsvarende viser sensitivitetsrapporten for et NLP-problem det som kalles «Reduced Gradient» for hver beslutningsvariabel. Redusert gradient har en tilsvarende tolkning som redusert kostnad. For variabler som ligger på nedre (øvre) grenseverdi viser denne hvor mye målfunksjonsverdien vil bli redusert (økt) dersom variabelen økes med en enhet.

NB! Gurobi Solver rapporterer «allowable increase» og «allowable decrease» også for noen ikke-lineære modeller, som modellen i oppgave 6. Merk at slike gyldighetsområder aldri er meningsfulle for ikke-lineære modeller. Både målfunksjonen og begrensningene må være lineære for at de skal gi mening!

Evolusjonære/genetiske algoritmer (avsnitt 8.12 i Ragsdale):

- *Ulike løsninger til et optimeringsproblem kan representeres som vektorer av tall (som kromosomer).*
- *Hvert kromosom har en verdi (fitness value) som sier noe om hvor egnet det er til å overleve.*
- *Genetiske algoritmer starter med en tilfeldig valgt populasjon av løsningsvektorer (kromosomer), og nye løsningsvektorer dannes ved hjelp av*
 - *Krysskombinasjoner, dvs ved å kombinere verdier fra flere av de eksisterende løsningsvektorene.*
 - *Mutasjoner, dvs tilfeldige endringer i en eller flere verdier i en eksisterende løsningsvektor.*
- *De løsningsvektorene (kromosomene) som har høyest fitness-verdi overlever til neste generasjon, og prosessen gjentas.*

Fordeler ved genetiske algoritmer:

- *De kan være bedre til å håndtere ikke-konvekse problemer, der tradisjonelle NLP-algoritmer lett kan ende opp i lokalt optimale løsninger.*
- *De tolerer ikke-kontinuerlige funksjoner som ikke kan deriveres, og der tradisjonelle NLP-algoritmer derfor ikke klarer å finne ut hvilken retning de skal gå i. Eksempler på ikke-kontinuerlige funksjoner i Excel er IF(), LOOKUP(), COUNTIF(), SUMIF(), AVERAGEIF(), MAX() og MIN().*

Merk at vi ikke har noen garanti for at den løsningen vi får ut av en genetisk algoritme er optimal, selv ikke når beslutningsproblemet er konvekt. I mange tilfeller kan det lønne seg å omformulere problemet slik at vi kan bruke tradisjonelle løsningsalgoritmer.

Oppgavesett D8: Simulering

Hvorfor bruke simulering til å analysere usikkerhet i beslutningssituasjoner? Avsnitt 12.2 i Ragsdale.

Alternativet til å bruke simulering er ofte å anta forventede verdier for usikre variabler, og å beregne målfunksjonsverdien ved å plugge inn forventningsverdiene. Hva er galt med denne fremgangsmåten?

1. Dersom sammenhengen mellom de usikre variablene og målfunksjonsverdien er ikke-lineær kan dette gi for høy/lav forventningsverdi for målfunksjonen.
2. Ved å bruke forventningsverdier får vi ikke noe inntrykk av hvor stor spredningen (risikoen) i målfunksjonsverdien er.

Noen måter å analysere usikkerhet på (avsnitt 12.3 i Ragsdale):

1. Best-case / worst-case analyser.
2. Følsomhetsanalyser (what-if).
3. Simulering.

Hvordan gjennomføre en simulering? Avsnitt 12.5-12.8 i Ragsdale.

For å simulere usikkerheten i en modell må vi gjennomføre følgende trinn:

- 1) Identifiser de usikre variablene (regnearkcellene) i modellen.
- 2) Velg passende slumptallgeneratorer (RNG = Random Number Generators) for de usikre variablene. En slumptallgenerator er en matematisk funksjon som genererer tilfeldige tall fra en gitt sannsynlighetsfordeling. I Analytic Solver velges en slumptallgenerator ved å legge inn en psi-funksjon i cellen som representerer den usikre variabelen. Se listen nedenfor.
- 3) Simuler n trekninger av verdier for de usikre variablene, og bruk modellen til å beregne n observasjoner for resultatvariablene i modellen. For at de n verdiene av en resultatvariabel skal bli lagret i Analytic Solver må vi legge til «+ PsiOutput()» i cellen som representerer resultatvariabelen.
- 4) Analyser observasjonene av resultatvariablene i utvalget. Diagrammer og statistiske beregninger for en resultatvariabel kan hentes fram i Analytic Solver ved å dobbeltklikke på cellen som inneholder resultatvariabelen.

Fordeling	Slumptallgenerator i ASP
Binomisk	PsiBinomial(n, p)
Kjirkvadrat	PsiChisquare(λ)
Poisson	PsiPoisson(λ)
Uniform	PsiUniform(min, max)
Eksponensial	PsiExponential(λ)
Triangulær	PsiTriang(min, mest sannsynlig, max)
Normal	PsiNormal(μ, σ)
Trunkert normal	PsiNormal(μ, σ , PsiTruncate(min, max))

Hvordan analysere resultatene fra en simulering? Avsnitt 12.9 i Ragsdale.

Fordelingen til en resultatvariabel kan utforskes ved å dobbeltklikke på cellen som inneholder resultatvariabelen. I tillegg finnes det en rekke funksjoner som beregner statistiske nøkkeltall om de n verdiene som er beregnet for en resultatvariabel (resultatcelle) X :

<i>Funksjon</i>	<i>Verdi</i>
$PsiMax(X)$	Maksimum for resultatcelle X
$PsiMin(X)$	Minimum for resultatcelle X
$PsiMean(X)$	Gjennomsnitt for resultatcelle X
$PsiStdDev(X)$	Standardavvik for resultatcelle X
$PsiPercentile(X, y)$	Gir verdien x slik at $P(X < x) = y$
$PsiTarget(X, y)$	Gir sannsynligheten $P(X < y)$
$PsiExpLoss(X)$	Forventet tap $E(X X < 0) * P(X < 0)$
$PsiExpGain(X)$	Forventet gevinst $E(X X > 0) * P(X > 0)$
$PsiCVar(X, y)$	Gjennomsnitt for andelen y av de laveste verdiene for X

Hvor usikre er estimatene vi får fra en simulering? Avsnitt 12.10 i Ragsdale.

Når vi simulerer trekker vi et tilfeldig utvalg fra en (uendelig) populasjon. Basert på utvalget vil vi estimere noen størrelser, for eksempel gjennomsnittet. Dersom vi gjentar trekningen, vil estimatet da bli det samme? Svaret er nei, men jo større utvalg vi tar, jo nærmere kommer estimatet den sanne verdien som gjelder for populasjonen.

Konfidensintervall for gjennomsnittet i populasjonen er

$$\left[\bar{y} - 1,96 \cdot \frac{s}{\sqrt{n}}, \bar{y} + 1,96 \cdot \frac{s}{\sqrt{n}} \right]$$

der

$\bar{y}$ = gjennomsnitt for utvalget

s = standardavvik for utvalet

n = antall trekninger i utvalget ($n \geq 30$)

Merk at konfidensintervall for gjennomsnitt kan beregnes vha formelen $PsiMeanCI()$ i Analytic Solver. $PsiMeanCI(X; 0,95)$ gir halvparten av bredden i et 95% konfidensintervall for resultatvariabelen X .

Konfidensintervall for en andel av en populasjon er

$$\left[\bar{p} - 1,96 \cdot \sqrt{\frac{\bar{p}(1 - \bar{p})}{n}}, \bar{p} + 1,96 \cdot \sqrt{\frac{\bar{p}(1 - \bar{p})}{n}} \right]$$

der

$\bar{p}$ = andel-estimat basert på et utvalg

n = antall trekkninger i utvalget ($n \geq 30$)

Hvordan kjøre flere simuleringer for å analysere effekten av ulike beslutninger? Avsnitt 12.14 i Ragsdale.

For å se effekten av ulike verdier for en parameter i simuleringsmodellen kan man kjøre flere simuleringer. For å gjøre dette i Analytic Solver må man bruke funksjonen `PsiSimParam()` i de cellene som inneholder parametrene man vil variere. I eksemplet i oppgave 5 kan man for eksempel variere innkjøpskvantumet i intervallet 100-1100 ved å legge inn følgende formel i cellen for innkjøpskvantumet:

`PsiSimParam({100\200\300\400\500\600\700\800\900\1000\1100})`

Eventuelt kan formelen henviser til et område i regnearket der de aktuelle verdiene for innkjøpskvantumet ligger.

For å kjøre simuleringer for alle disse verdiene gjøres følgende:

1. Velg «Charts» -> «Multiple Simulations» -> «Parameter Analysis». Dersom dere ønsker en tabell velges «Reports» -> «Simulation» -> «Parameter Analysis».
2. Velg en resultatvariabel øverst i dialogboksen, for eksempel profittcellen i modellen fra oppgave 5.
3. Velg hvilke statistiske nøkkeltall dere vil plote for den valgte resultatvariabelen. Her kan dere for eksempel vise gjennomsnitt (mean), standardavvik, og en rekke andre nøkkeltall.
4. Angi en eller flere parametre som dere vil variere. Dette vil være parametre som er definert ved hjelp av `PsiSimParam()`.
5. Velg hvordan parametrene skal variere (one parameter at a time / all parameters simultaneously / two selected parameters independently), og sett antall punkter på x-aksen.
6. Til slutt klikker dere OK, og Analytic Solver vil utføre simuleringene som trengs for å lage diagrammet.

Hvordan integrere simulering og optimering i samme modell? Avsnitt 12.15 i Ragsdale.

Simulering og optimering kan kombineres i Analytic Solver. Man kan for eksempel bruke en av funksjonene som er vist i tabellen på side 3 i målfunksjonen eller begrensninger i et optimeringsproblem. Dersom dette gjøres vil Analytic Solver velge en «Evolutionary Solver» som løsningsmetode, og dere har ingen garanti for at Solver finner en optimal løsning. Se stoffet om evaluasjonære/genetiske algoritmer i kapittel 8.

Oppgavesett D10: Beslutningsproblemer under usikkerhet (1)

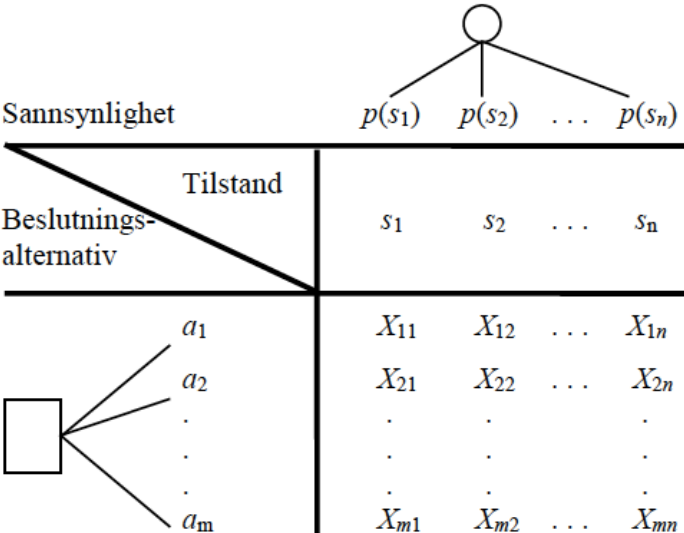
Hvilke grunnleggende komponenter finnes i beslutningsproblemer under usikkerhet? [Vi antar at vi ser på beslutningssituasjoner der det skal foretas et valg mellom gjensidig utelukkende alternativer og at usikkerheten er knyttet til bestemte begivenheter som ikke påvirkes av beslutningstaker og som har avgjørende betydning for konsekvensen av de fleste (eller alle) beslutningsalternativene.]

Et beslutningsproblem under usikkerhet har følgende grunnleggende komponenter:

- 1. **Beslutningstaker:** Den som har myndighet til å ta den endelige beslutningen.*
- 2. **Beslutningsalternativer:** En samlet mengde av mulige valg som beslutningstaker står overfor. Alternativene betegnes ofte $a_1, a_2, \dots, a_m$. Beslutningstaker har full kontroll over alternativene og står fritt til å velge ett av dem.*
- 3. **Tilstander ("states of nature"):** Ytre forhold av betydning i beslutningssituasjonen og som beslutningstaker ikke har noen muligheter til å påvirke. Tilstandene betegnes ofte $s_1, s_2, \dots, s_n$. Tilstandene er altså ikke-kontrollerbare begivenheter for beslutningstakeren. En av dem inntreffer etter at et alternativ er valgt, og de er således direkte knyttet til usikkerheten i beslutningssituasjonen. Beslutningstaker forutsettes å ha full oversikt over hvilke relevante tilstander som kan inntreffe.*
- 4. **Sannsynligheter:** Anslag på hvor stor sjanse det er for at de respektive tilstander skal inntreffe. I stor grad vil sannsynlighetsvurderingene være basert på de subjektive oppfatninger som beslutningstakeren har, ut fra den informasjon som vedkommende sitter inne med. Sannsynligheten for at en bestemt tilstand s_j skal inntreffe betegnes ofte $p(s_j)$. Ofte forsøker vi å komme frem til en mest mulig objektiv sannsynlighetsfordeling ved å benytte historiske observasjoner. Det forutsettes da implisitt at de faktorene som påvirker utfallene er stabile over tid. Andre ganger kan vi lage sannsynlighetsfordelinger basert på mer subjektive oppfatninger.*
- 5. **Konsekvenser:** De resultater som oppnås som følge av å velge ett bestemt alternativ og at én bestemt tilstand inntreffer. Konsekvenser forutsettes gitt for hvert mulig par av alternativ og tilstand. Konsekvensverdier betegnes ofte X_{ij} , hvor fotskrift i står for at alternativ a_i er valgt og fotskrift j for at tilstand s_j inntreffer.*
- 6. **Preferanser:** Grunnlaget for hvordan beslutningstaker bedømmer de forskjellige, mulige konsekvenser. Det forutsettes at beslutningstaker kan spesifisere hvilken konsekvens som foretrekkes i valget mellom to (eventuelt at en er indifferent).*
- 7. **Beslutningskriterier eller beslutningsregler ("decision rules"):** Et beslutningskriterium tilordner en numerisk verdi til hvert av alternativene ut fra de aktuelle konsekvensene og, ofte også, deres sannsynlighet for å inntreffe. Kriteriet er basert på beslutningstakers preferanser og viser hvilket alternativ som er best, eventuelt at to eller flere alternativer er likeverdige.*

Hva er en beslutningstabell ("payoff matrix")? Avsnitt 14.4 i Ragsdale.

En beslutningstabell gir en oppsummering av konsekvensene for hver kombinasjon av beslutningsalternativ og tilstand. Over matrisen kan vi notere tilstandssannsynlighetene (hvis disse er oppgitt). [Figurene på toppen av og til venstre i tabellen forklares når beslutningstrær presenteres nedenfor.]



Sannsynlighet		$p(s_1)$	$p(s_2)$	$\dots$	$p(s_n)$
Beslutnings- alternativ	Tilstand	s_1	s_2	$\dots$	s_n
		s_1	s_2	$\dots$	s_n
	a_1	X_{11}	X_{12}	$\dots$	X_{1n}
	a_2	X_{21}	X_{22}	$\dots$	X_{2n}
	$\cdot$	$\cdot$	$\cdot$		$\cdot$
	$\cdot$	$\cdot$	$\cdot$		$\cdot$
	a_m	X_{m1}	X_{m2}	$\dots$	X_{mn}

Beslutningstabeller er kun anvendelige i grunnleggende beslutningstilfeller når valg av alternativ skjer på ett, og bare ett, tidspunkt og én tilstand inntreffer etterpå.

Hvilke beslutningskriterier finnes og hvordan kan de brukes til å fatte beslutninger? Avsnitt 14.5 og 14.6 i Ragsdale.

Det er utviklet en rekke ulike beslutningskriterier, som gjerne deles inn i to hovedgrupper, avhengig av om tilstandssannsynligheter inngår i beregningene eller ikke. Ingen beslutningskriterier fungerer best i alle sammenhenger, alle har noen svakheter. De gir likevel økt innsikt i problemet og derigjennom bedre beslutninger.

I det følgende forutsetter vi at konsekvensverdiene er av typen «jo større, desto bedre». (Er konsekvensene av den motsatte typen, f.eks. kostnader, slik at lavest mulig verdi alt annet like er best, må det settes minustegn foran disse for å oppnå konsistens og korrekt beslutning.)

Vi ser først på noen beslutningskriterier uten bruk av tilstandssannsynligheter: maximax, maximin og minimax beklagelse.

A. Maximax-kriteriet (Optimismekriteriet)

Med maximax-kriteriet finner vi det alternativet som har det mest fordelaktige utfallet. Først bestemmes beste konsekvensverdi for hvert alternativ og deretter velger vi det alternativet hvor denne verdien er størst.

B. Maximin-kriteriet (Pessimismekriteriet)

Med maximin-kriteriet finner vi det alternativet som har det minst ufordelaktige utfallet. Først bestemmes verste konsekvensverdi for hvert alternativ, og deretter velger vi det alternativet hvor denne er størst.

C. Minimax beklagelse ("Minimax regret")

Her er fokus på det vi har gått glipp av, når vi i etterkant ser hvilken tilstand som har inntruffet (etterpåklokskap). Dette kalles for beklagelse og er en alternativkostnad. For en gitt tilstand er beklagelsen til et bestemt alternativ lik forskjellen mellom beste konsekvensverdi i den gitte tilstanden og konsekvensverdien til alternativet vi ser på. Med dette utgangspunktet lages det en beklagelsesmatrise, som er helt lik beslutningstabellen, bortsett fra at beklagelsesverdier, og ikke konsekvensverdier, inngår. Beklagelsesverdier kan per definisjon ikke bli negative. Minimax beklagelse innebærer at en beregner størst mulig beklagelse for hvert beslutningsalternativ og velger det alternativet hvor denne er minst (siden en foretrekker minst mulig beklagelse).

Hvordan kan vi ta hensyn til tilstandssannsynligheter i beslutningskriteriene? Avsnitt 14.7 og 14.15 i Ragsdale.

Vi vil se på tre beslutningskriterier som bruker tilstandssannsynligheter: maksimum forventet verdi, minimum forventet beklagelse og maksimum forventet nytte.

A. Maksimum forventet verdi ("expected monetary value", EMV)

For hvert beslutningsalternativ beregner vi forventet verdi, dvs. en veiet sum av alternativets konsekvenser med tilstandssannsynlighetene som vektor. Deretter velges det alternativet som får den høyeste verdien.

B. Minimum forventet beklagelse ("expected regret" eller "expected opportunity lost", EOL)

For hvert beslutningsalternativ beregner vi forventet beklagelse og velger det som får den minste verdien.

C. Maksimum forventet nytte ("expected utility")

En nyttefunksjon omgjør konsekvensverdier til nytteverdier. Ved å bruke en nyttefunksjon kan vi ta hensyn til risikoholdning når vi tar beslutninger. Hvis for eksempel nyttefunksjonen er stigende (førstederivert > 0 , slik at jo høyere konsekvensverdi, desto høyere nytte), og dessuten konkav (andrederivert < 0), er beslutningstaker risikoavers. Når nyttefunksjonen er kjent, finner vi nytteverdien for hver konsekvensverdi og deretter forventet nytte ved å bruke tilstandssannsynlighetene. Vi velger det alternativet som har høyest forventet nytte.

En sikkerhetsekvivalent, CE, er det pengebeløpet som beslutningstaker synes er like godt å motta sikkert som å velge et usikkert alternativ. Nytteverdien til sikkerhetsekvivalenten er altså lik forventet nytte av det usikre alternativet.

Hvor sensitive er beslutningene i forhold til de estimerte sannsynlighetene? Avsnitt 14.7.3 i Ragsdale.

Også for beslutningsproblemer under usikkerhet er det vanlig å foreta følsomhetsanalyser for å vurdere hvor robust det valgte alternativet er for endringer i forutsetningene. Typisk er vi opptatt av endringer i tilstandssannsynlighetene. Når følsomhetsanalysen utføres i beslutningssituasjoner med to tilstander kan vi vise innenfor hvilke intervaller ulike sannsynligheter gir samme beslutning. En slik følsomhetsanalyse kan gjøres for alle de tre beslutningskriteriene som bruker tilstandssannsynligheter: maksimum forventet verdi, minimum forventet beklagelse og maksimum forventet nytte.

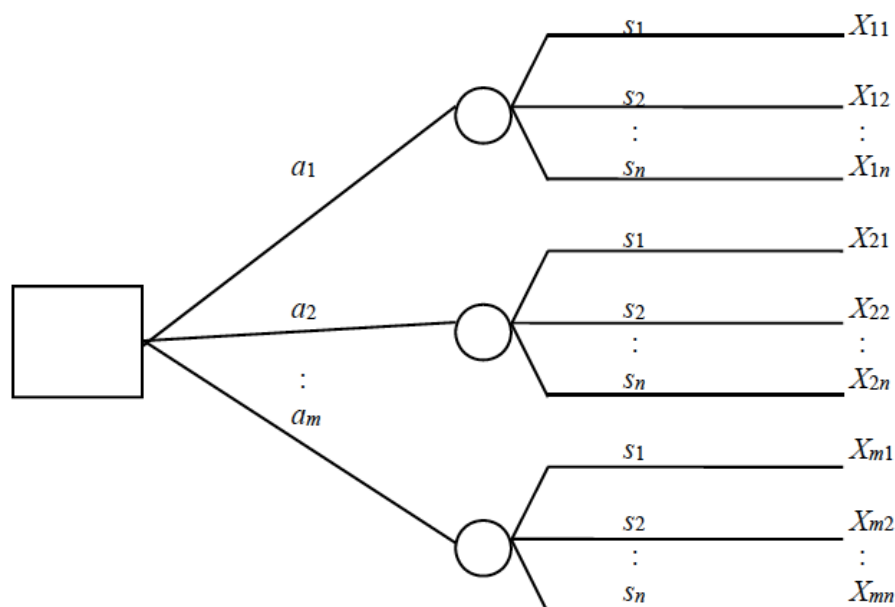
Hvordan lager vi et beslutningstre ("decision tree")? Avsnitt 14.9 i Ragsdale.

Et beslutningstre består av ulike typer punkter, og ut fra disse løper det grener som forbinder punktene. Et **beslutningspunkt** ("decision node") angis med et kvadrat og viser at det skal foretas et valg av alternativ. Et **sjansepunkt** ("event node") angis med en sirkel og viser at en bestemt tilstand vil inntreffe etter punktet.

Det betyr at på grenene ut fra et **beslutningspunkt** finner vi alle **alternativene**, mens vi på grenene ut fra et **sjansepunkt** finner alle **tilstandene**.

Enden av hver gren i beslutningstreet kalles **løv** eller **løvnoder** ("leaves" eller "terminal nodes"). Konsekvensene avmerkes på enden av hver gren i beslutningstreet, dvs. i løvnodene. Kontantstrømmer og sannsynligheter knyttet til ulike alternativer og tilstander kan også avmerkes på grenene.

Beslutningstre med ett beslutningspunkt og én usikker hendelse:



Beslutningstrær kan brukes til å modellere mer komplekse situasjoner, for eksempel problemer som involverer flere beslutninger og usikre hendelser. Et beslutningstre får også frem det sekvensielle aspektet, dvs. utviklingen over tid.

Hva er tilbakerullingsprinsippet ("rolling back") og hvordan brukes dette i beslutningsanalyse? Avsnitt 14.9.1 i Ragsdale.

Tilbakerullingsprinsippet er en beskrivelse av hvordan vi går frem i selve beslutningstreet for å finne løsningen. Løsningen viser hvilke grener det er optimalt å velge på veien frem til enden (toppen) av treet (løvene). Spesielt når beslutningstreet er stort, gir dette en god oversikt over den beste strategien for beslutningstakeren.

Løsningen av et beslutningsproblem skjer i tilbakerullingsprinsippet på følgende måte:

- *Vi begynner i løvnodene til høyre i treet og regner oss nedover mot roten (dvs. noden helt til venstre i treet).*
- *For hvert sjansepunkt beregner vi forventet verdi basert på sannsynligheter og verdier på grenene til høyre for punktet. Denne forventningsverdien settes over sjansepunktet og representerer den forventede verdien for beslutningen hvis gren ender i sjansepunktet.*
- *Deretter finner vi forventet verdi for et beslutningspunkt, som høyeste verdi over de sjansepunktene som det løper grener fra beslutningspunktet til. Ikke-optimale beslutninger får sine beslutningsgrener beskåret (anført med $\parallel$) for å vise at det ikke er lurt å klatre i disse grenene opp mot toppen.*

Oppgavesett D11: Beslutningsproblemer under usikkerhet (2)

Hva er verdien av informasjon? Avsnitt 14.8 og 14.13 i Ragsdale.

De beslutningsproblemene under usikkerhet som vi ser på, er slik at vi kjenner alle mulige tilstander på beslutningstidspunktet, men vi vet ikke hvilken av tilstandene som vil inntreffe. Siden denne usikkerheten påvirker oppnådd konsekvensverdi, forstår vi umiddelbart at det kan være verdifullt i forkant å få nærmere greie på hva som kommer til å skje.

Informasjon i denne sammenhengen betyr rett og slett at beslutningstakeren vil forandre sin oppfatning om sannsynligheten for de ulike tilstandene.

Hva er forventet verdi med perfekt informasjon og hvordan kan vi analysere verdien av perfekt informasjon i et beslutningstre?

Med perfekt informasjon vet vi med sikkerhet hvilken tilstand som inntreffer og vi kan velge det beslutningsalternativet som gir beste konsekvensverdi for hver tilstand. I det følgende antar vi at beslutningstaker ønsker å maksimere forventet verdi (EMV).

I denne situasjonen kan vi tenke oss at det skal tas to beslutninger i sekvens; 1) om vi skal hente inn (perfekt) informasjon eller ikke og 2) den opprinnelige beslutningen.

I beslutningstreet starter vi da med et beslutningspunkt med to grener: om det skal hentes inn informasjon (Info+) eller ikke (Info-). Hvis det ikke hentes inn informasjon, er beslutningssituasjonen den samme som før. På enden av "Info-"-grenen vil vi derfor ha det opprinnelige beslutningstreet, der vi ikke har ny informasjon. I enden av "Info+"-grenen vil vi først få et sjansepunkt, fordi selv om informasjonen er perfekt og vi håper at en gunstig tilstand skal inntreffe, så kan ikke informasjonskilden påvirke hvilken tilstand som inntreffer. Sannsynligheten for at de ulike tilstandene inntreffer er som før. Når informasjonen er innhentet, er imidlertid all usikkerhet avslørt og i neste beslutningspunkt, over de opprinnelige beslutningsalternativene, kan vi velge den beslutningen som gir best konsekvensverdi.

Vi finner løsningen i beslutningstreet ved å bruke tilbakerullingsprinsippet.

Merk at ofte foretas det ikke noen beskjæring av den ene grenen ut fra første beslutningspunkt (Info+ / Info-). Det kan vi ikke gjøre før vi vet hva informasjonen koster. [Dersom vi visste hva informasjonen koster, kunne vi trukket fra dette beløpet i alle konsekvensverdier som ligger på grener etter at informasjonsinnhenting er valgt. Da ville tilbakerulling gitt oss alle optimale valg direkte.]

Verdien over sjansepunktet i enden av "Info+"-grenen betegnes forventet verdi med perfekt informasjon eller forventet verdi under sikkerhet ("Expected Value With Perfect Information" eller "Expected Value Under Certainty", EVUC).

Hva er verdien av perfekt informasjon? Avsnitt 14.8 i Ragsdale.

Med perfekt informasjon vil beslutningstakeren aldri ta en beslutning som i ettertid vurderes som uheldig. Dette er en klar forbedring av beslutningssituasjonen, som må gi seg utslag i en høyere (eller i alle fall minst like høy) forventet verdi. Denne forbedringen i forventet verdi betegnes forventet verdi av perfekt informasjon ("Expected Value of Perfect Information", EVPI). EVPI utgjør altså forskjellen mellom EVUC og forventet verdi av det beste alternativet uten bruk av informasjon, dvs. $EVPI = EVUC - EMV$.

Hva er sammenhengen mellom forventet verdi av perfekt informasjon (EVPI) og minimum forventet beklagelse (EOL)? Avsnitt 14.8 i Ragsdale.

Med perfekt informasjon kan vi alltid velge det alternativet som viser seg å bli det beste, fordi vi får vite hvilken tilstand som helt sikkert kommer til å inntreffe. Grunnlaget for verdien av perfekt informasjon er de beløpssmessige besparelsene som følger av dette.

For kriteriet minimum forventet beklagelse, gjorde vi lignende vurderinger, men med motsatt innfallsvinkel: hva vi går glipp av.

Det er derfor ikke overraskende at forventet verdi av perfekt informasjon tilsvarer nettopp minimum forventet beklagelse. Ved å innhente perfekt informasjon slipper vi å beklage oss over at vi kunne ha gjort det bedre, vi foretar alltid de riktige valgene, også vurdert i etterkant. Økningen i forventningsverdi som følger av dette, tilsvarer forventningsverdien av de mulige tapene (kalkulert ved forventet beklagelse) som vi da unngår.

Hva er verdien av imperfekt informasjon, eller "Sample information"? Avsnitt 14.13 i Ragsdale.

Forventet verdi av perfekt informasjon er en øvre grense for hva informasjon kan være verd for beslutningstaker. Bruk av imperfekt informasjon betyr at beslutningstakeren får en klarere oppfatning om hvilken tilstand som vil inntreffe, men den er ikke 100% sikker. Da vil rimeligvis også verdien av informasjonen være mindre.

Vi kan også analysere verdien av imperfekt informasjon i et beslutningstre. Forklaringen av beslutningstreet er helt parallell med tilfellet perfekt informasjon. Først må vi ta stilling til om informasjon skal innhentes eller ikke. Hvis nei, blir situasjonen som uten informasjon. Hvis ja, inntreffer først en sjansesituasjon med informasjonskildens utsagn før vi velger beslutningsalternativ. Forskjellen i forhold til tilfellet med perfekt informasjon er at informasjonskildens utsagn ikke er 100% sikkert. Selv om utsagnet er at tilstanden vil bli god, så kan den likevel bli dårlig. Imperfekt informasjon betyr jo at informasjonen ikke alltid er riktig. Etter at vi har tatt beslutning om alternativ, vil det derfor komme et sjansepunkt. Sannsynlighetene i dette sjansepunktet er imidlertid endret avhengig av utsagnet til informasjonskilden.

Hvordan kan vi beregne betingede sannsynligheter? Avsnitt 14.14 i Ragsdale.

Et uttrykk for en betinget sannsynlighet er:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

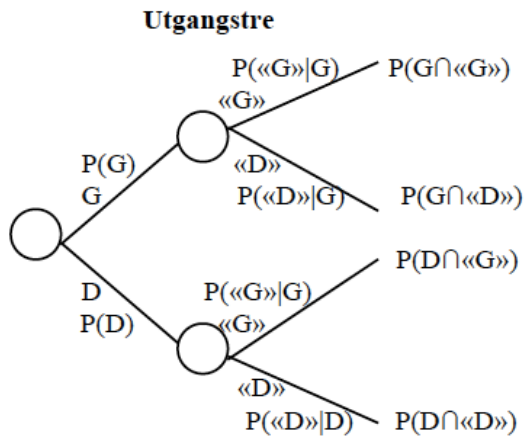
Som impliserer at

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$$

Bayes' lov er gitt ved:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|\bar{A})P(\bar{A})}$$

Vi kan også regne på betingede sannsynligheter ved å bruke sannsynlighetstrær:

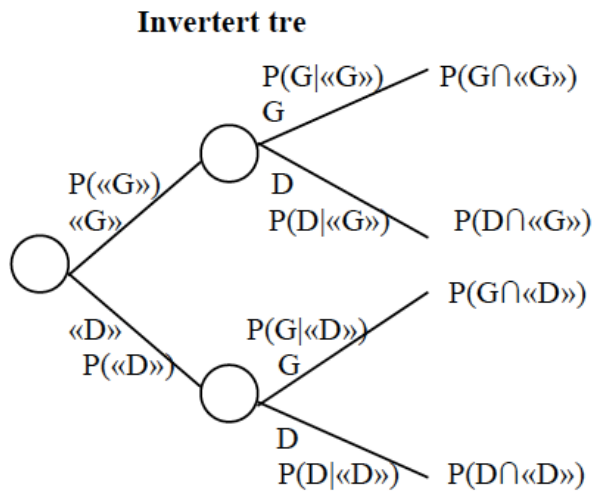


Ofte er informasjonen som oppgis om informasjonskilden de simultane sannsynlighetene i løvnodene i sannsynlighetstreet over, eller de betingede sannsynlighetene etter det andre sjansepunktet.

Når vi har de simultane sannsynlighetene, som enten er oppgitt eller som kan finnes ved å multiplisere gjennom treet, kan vi lage en tabell som gjør det enkelt å finne totalsannsynlighetene:

	Faktisk		Sum
	G	D	
Utsagn			
«G»	$P(G \cap \text{«G»})$	$P(D \cap \text{«G»})$	$P(\text{«G»})$
«D»	$P(G \cap \text{«D»})$	$P(D \cap \text{«D»})$	$P(\text{«D»})$
Sum	$P(G)$	$P(D)$	

Da kan vi regne oss frem til det inverterte sannsynlighetstreet som har nettopp de betingede sannsynlighetene som vi trenger i beslutningstreet der vi analyserer verdien av imperfekt informasjon, nemlig sannsynlighetene for at tilstand G eller D inntreffer gitt den (imperfekte) informasjonen vi har mottatt.



Oppgavesett D12: Prognoser og tidsserier

Hvordan lage prognoser for stasjonære tidsserier? Avsnitt 11.4-11.6 i Ragsdale.

En tidsserie består av observasjoner av en kvantitativ variabel over tid.

En stasjonær tidsserie har ikke noen signifikant oppadgående eller nedadgående trend over tid.

Ekstrapolering tar utgangspunkt i tidligere observasjoner av en tidsserievariabel for å forutsi fremtidige verdier av variabelen:

$$\hat{Y}_{t+1} = f(Y_t, Y_{t-1}, Y_{t-2}, \dots)$$

Et glidende gjennomsnitt (moving average) tar gjennomsnittet av de k foregående periodene:

$$\hat{Y}_{t+1} = \frac{Y_t + Y_{t-1} + \dots + Y_{t-k+1}}{k}$$

Merk at det finnes ingen generell metode for å bestemme k .

Hvordan kan vi måle nøyaktigheten i en prognosemodell? Avsnitt 11.2 i Ragsdale.

Noen vanlige mål er: mean absolute deviation (MAD), mean absolute percent error (MAPE), the mean square error (MSE) og root mean square error (RMSE). Vi vil fokusere på the mean square error (MSE) eller gjennomsnittlig kvadrert avvik:

$$\text{MSE} = \sum_{i=1}^n \frac{(Y_i - \hat{Y}_i)^2}{n}$$

[Merk at vi bør være varsomme når vi sammenligner MSE-verdier for to forskjellige prognose-teknikker. Den laveste MSE-verdien kan skyldes at metoden passer veldig godt til eldre verdier, men dårlig til de siste verdiene. Noen ganger kan det derfor være klokt å beregne MSE kun ved hjelp av de nyeste verdiene.]

Hva er vektet glidende gjennomsnitt (weighted moving average)? Avsnitt 11.5 i Ragsdale.

Vektet glidende gjennomsnitt er gitt ved formelen:

$$\hat{Y}_{t+1} = w_1 Y_t + w_2 Y_{t-1} + \dots + w_k Y_{t-k+1}$$

$$\text{hvor } 0 \leq w_i \leq 1 \text{ og } \sum w_i = 1$$

Hva er eksponentiell glatting (exponential smoothing)? Avsnitt 11.6 i Ragsdale.

Eksponentiell glatting er gitt ved formelen:

$$\hat{Y}_{t+1} = \hat{Y}_t + \alpha(Y_t - \hat{Y}_t)$$

hvor $0 \leq \alpha \leq 1$

Prognosen kan initialiseres ved å sette første prognoseverdi lik første observasjon.

Hvordan lage prognoser for tidsserier med additive sesongeffekter? Avsnitt 11.8 i Ragsdale.

Sesongvariasjoner er et regelmessig gjentakende mønster i tidsserier. Sesongeffektene kan være additive eller multiplikative. Additive sesongeffekter tenderer til å være av samme størrelsesorden hver gang de inntreffer. Multiplikative sesongeffekter tenderer til å øke over tid.

Modell for additiv sesongeffekt:

$$\hat{Y}_{t+n} = E_t + S_{t+n-p}$$

hvor

$$E_t = \alpha(Y_t - S_{t-p}) + (1 - \alpha)E_{t-1}$$

$$S_t = \beta(Y_t - E_t) + (1 - \beta)S_{t-p}$$

$$0 \leq \alpha \leq 1$$

$$0 \leq \beta \leq 1$$

p er antall ulike sesonger, E_t er forventet nivå i tidsperiode t og S_t er sesongfaktoren i tidsperiode t.

Modellene kan initialiseres på ulike måter, men for de første p periodene er det vanlig å sette E lik gjennomsnittet av observasjonene, og sesongfaktoren S lik differansen mellom observert verdi og gjennomsnittet gitt ved E (se også side 581 i Ragsdale).

Hvordan lage prognoser for tidsserier med multiplikative sesongeffekter? Avsnitt 11.9 i Ragsdale.

Modell for multiplikativ sesongeffekt:

$$\hat{Y}_{t+n} = E_t \times S_{t+n-p}$$

hvor

$$E_t = \alpha(Y_t/S_{t-p}) + (1 - \alpha)E_{t-1}$$

$$S_t = \beta(Y_t/E_t) + (1 - \beta)S_{t-p}$$

$$0 \leq \alpha \leq 1$$

$$0 \leq \beta \leq 1$$

p er antall ulike sesonger, E_t er forventet nivå i tidsperiode t og S_t er sesongfaktoren i tidsperiode t.

En måte å initialisere modellen for de første p periodene er å sette E lik gjennomsnittet av de første p observasjonene, og sesongfaktoren S lik forholdet mellom observert verdi og gjennomsnittet gitt ved E (se også side 584 i Ragsdale).