

Nyheder i Gym-pakken til Maple 2020

Som følge af ændrede læreplaner, var der en lang række nyheder i Gym-pakken til Maple 2019, men desværre blev disse – som følge af et dødt link - aldrig rigtigt åbenbarede. De kan alle findes i sektionen 'Nyheder i Gym-pakken til Maple 2019' nedenfor.

Der er ingen nye rutiner i Gym-pakken til Maple 2020, men til gengæld er der en række rutiner, der er blevet kraftigt udvidet og nogle, hvor udførelses hastigheden er øget dramatisk. Disse er:

Forbedringer

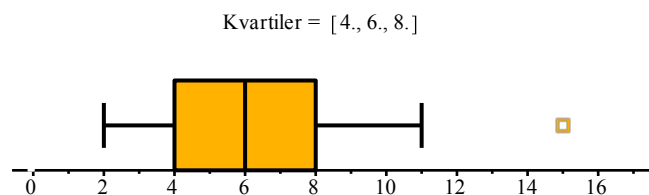
boksplot

Boksplot-kommandoen er blevet reprogrammeret, dels for at implementere outliers, og dels for at simplificere syntaksen, således at de kantede parenteser kun skal benyttes, hvis to boksplot med forskellige options skal sammenlignes.

with(Gym) :

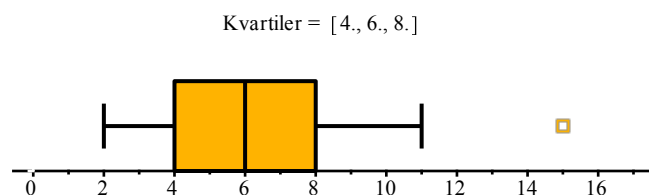
obs := [2, 3, 3, 3, 4, 4, 4, 5, 5, 6, 6, 6, 6, 7, 7, 7, 7, 8, 9, 9, 11, 11, 15] :

boksplot(obs, outliers)

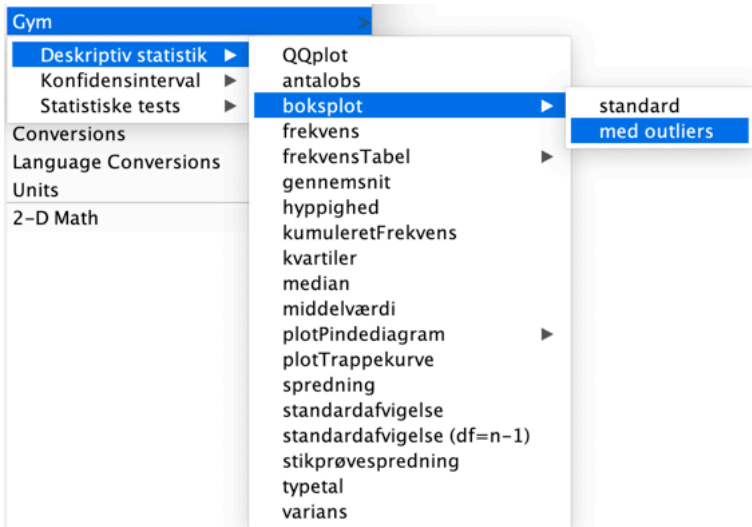


Kommandoen er også implementeret i kontekstmenuen:

obs $\xrightarrow{\text{boksplot}}$

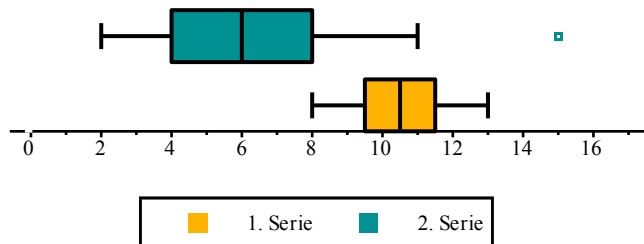


Kommandoen hentes sådan her:



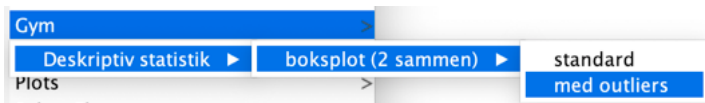
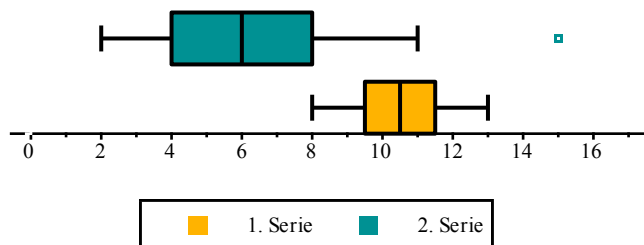
Hvis to observationssæt skal sammenlignes, hvor de begge skal tegnes med eventuelle outliers, er kommandoen denne

```
obs1 := [8, 8, 9, 9, 9, 10, 10, 10, 10, 10, 11, 11, 11, 11, 11, 12, 12, 13, 13, 13] :
obs2 := [2, 3, 3, 3, 4, 4, 4, 5, 5, 6, 6, 6, 6, 7, 7, 7, 7, 8, 9, 9, 11, 11, 15] :
boksplot(obs1, obs2, outliers)
```



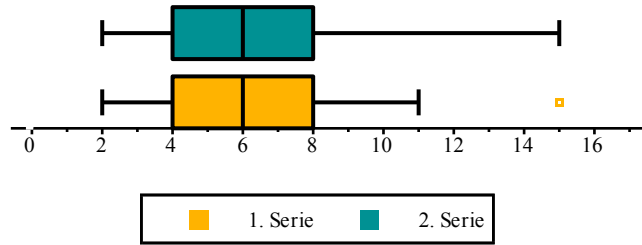
Dette kan også klares fra kontekstmenuen:

```
obs1, obs2  $\xrightarrow{\text{boksplot}}$ 
```



Hvis kun den ene skal tegnes med outliers, kan kontekstmenuen ikke benyttes. Kommandoen er:

`boksplot([obs2, outliers], [obs2])`

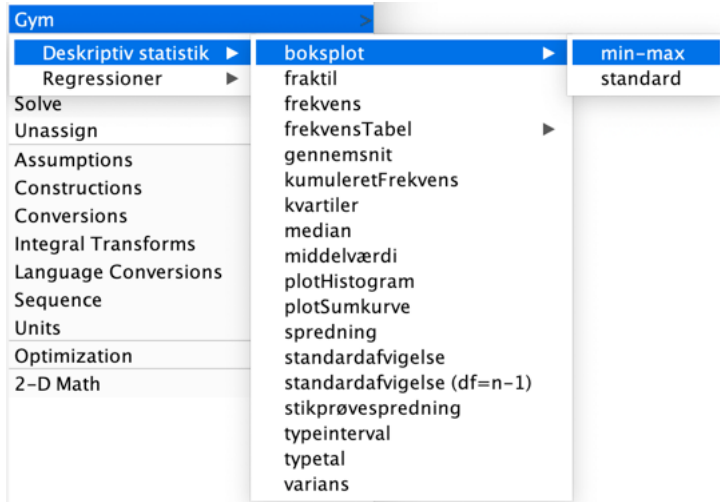
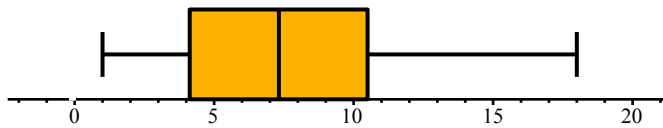


Ved et grupperet observationssæt man den mindste- og største observation tilføjes med `minmax`:

$$grp := \begin{bmatrix} 0..5 & 10 \\ 5..10 & 14 \\ 10..12 & 3 \\ 12..17 & 5 \\ 17..20 & 1 \end{bmatrix} :$$

`grp` $\xrightarrow{\text{boksplot}}$

Kvartiler = [4.125, 7.321, 10.50]



plotResidualer

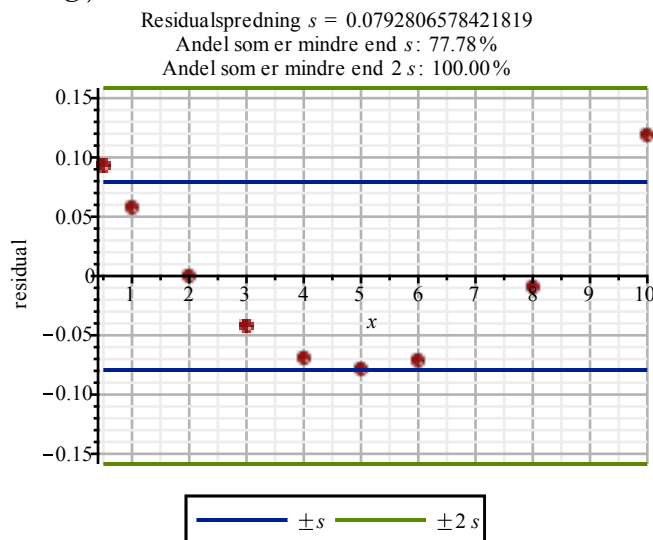
Kommandoen `plotResidualer` er blevet udstyret med 4 vandrette linjer $y = \pm s$ og $y = \pm 2s$, hvor s er residualspreddingen. Desuden udregnes, hvor stor en procentdel af residualerne der falder i intervallerne $[-s, s]$ og $[-2s, 2s]$.

with (Gym) :

$X := \langle 0.5, 1, 2, 3, 4, 5, 6, 8, 10 \rangle :$

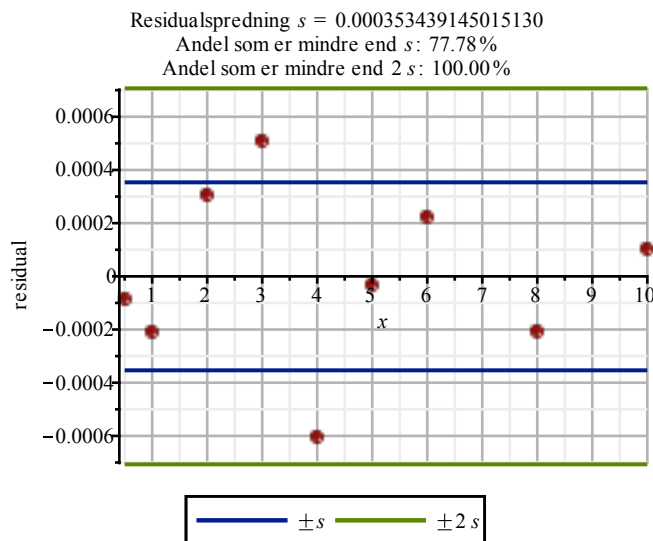
$Y := \langle 0.002, 0.008, 0.033, 0.074, 0.130, 0.204, 0.294, 0.522, 0.816 \rangle :$

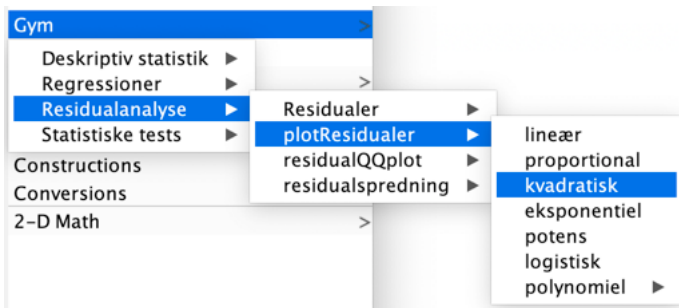
$plotResidualer(X, Y, LinReg)$



Kommandoen vil virke på regressionstyperne: *LinReg*, *PropReg*, *KvadReg*, *ExpReg*, *PowReg*, *Polyreg* og *LogistReg*. Fx en kvadratisk regression:

$X, Y \xrightarrow{plotResidualer}$





Kommandoen tillader alle Maples plot options som ekstra parametre.

binomialTest

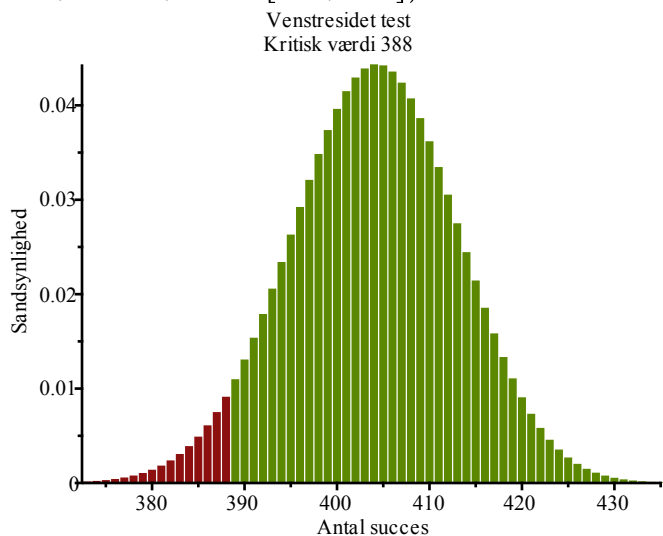
Der er ingen ændringer i selve kommandoen, som er fra en tid, hvor antalsparameteren næppe oversteg 50. I vore dage har vi set eksempler på antalsparametre på 5-600. Kommandoen tegnede alle pinde i pindediagrammet – også dem, der var meget små.

I den nye implementation tegnes et symmetrisk område om middelværdien, der bliver beregnet således, at der vises et antal røde pinde for at markere det kritiske område. Desuden kan alle Maples plot options benyttes.

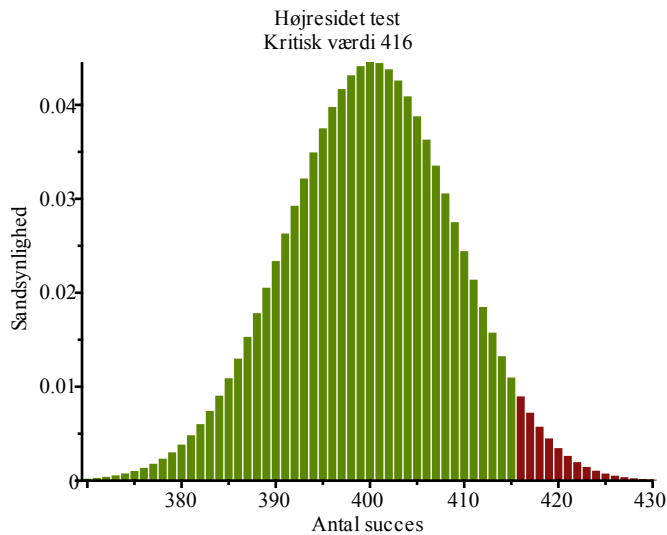
Plottet tegnes i den nye implementation på et øjeblik:

with(Gym) :

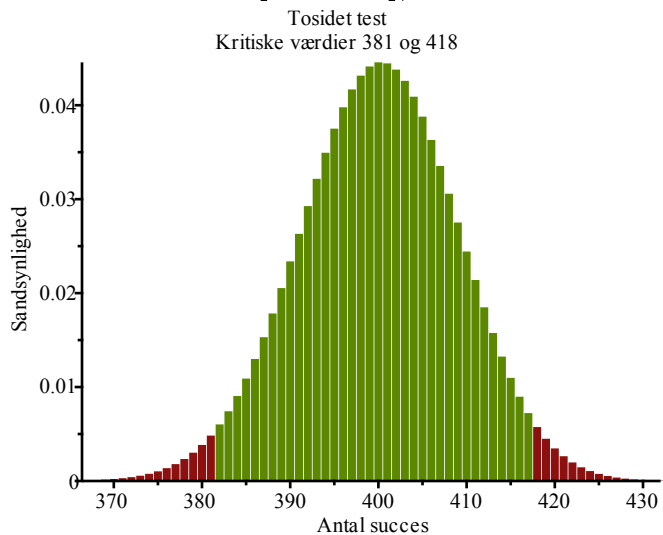
binomialTest(505, .8, 0.05, venstre, size = [500, 400])



binomialTest(500, .8, 0.05, højre, size = [500, 400])



`binomialTest(500, .8, 0.05, tosidet, size = [500, 400])`



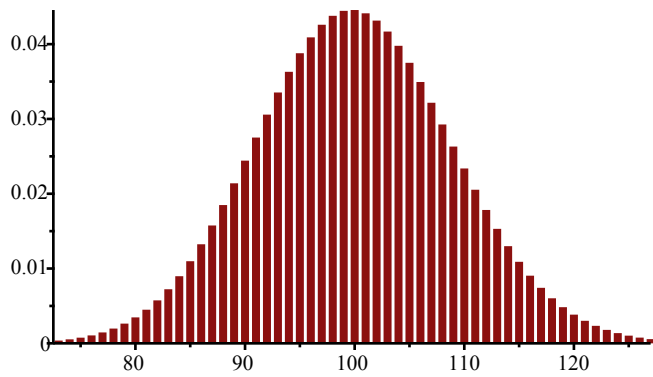
pinediagramBIN

Kommandoen `pinediagramBIN` led af samme skavank som `binomialTest`: Ved en stor antalsparameter tog det en rum tid at generere plottet – og det plot, der kom, var ikke ventetiden værd, idet alle pindene mere eller mindre smeltede sammen til en stor plamage.

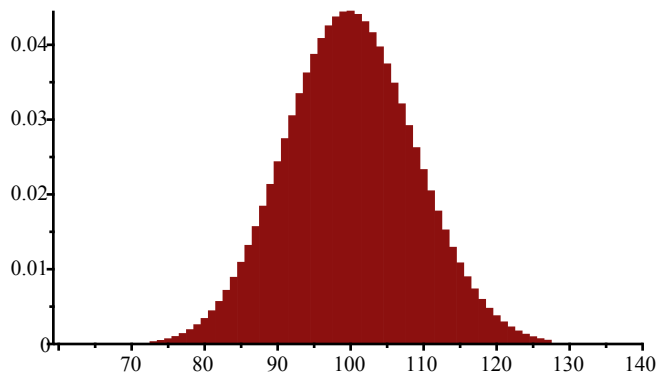
I den nyeste implementation tegnes som standard kun i intervallet $[\mu - 3 \cdot \sigma, \mu + 3 \cdot \sigma]$. Det bevirker, at plottet tegnes på et øjeblik. Ønskes et større område tegnet, så kan dette klares med `view`-parameteren.

`with(Gym) :`

`pinediagramBIN(500, 0.2, size = [500, 300])`



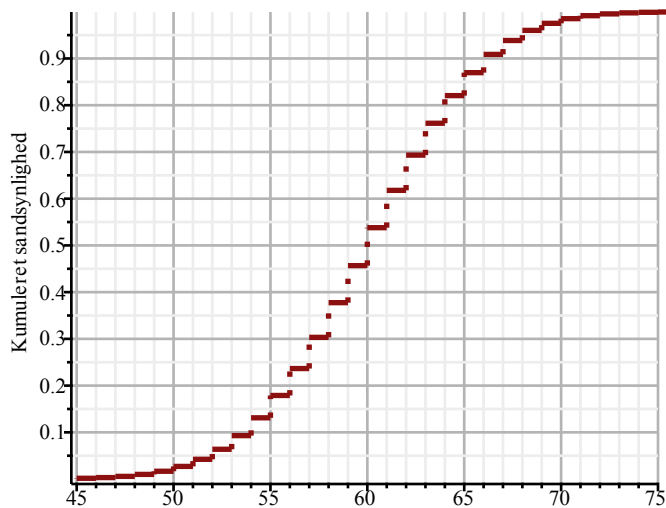
`pindediagramBIN(500, 0.2, view = [60..140, "default"], size = [500, 300])`



trappekurveBIN

Kommandoen *trappekurveBIN* har undergået de samme forandringer som *pindediagramBIN*, så man nu kan få en fornuftig graf tegnet for større værdier af antalsparameteren.

`trappekurveBIN(100, 0.6, labels = ["", "Kumuleret sandsynlighed"], labeldirections = ["default", "vertical"], size = [500, 400])`



plotSumkurve

Af en eller anden grund var kommandoen *plotSumkurve* blevet meget langsom. Det er der ændret i den nye implementation. Der er ingen ændringer i selve kommandoen – ud over, at alle Maples plot options kan sendes til kommandoen.

PolyReg

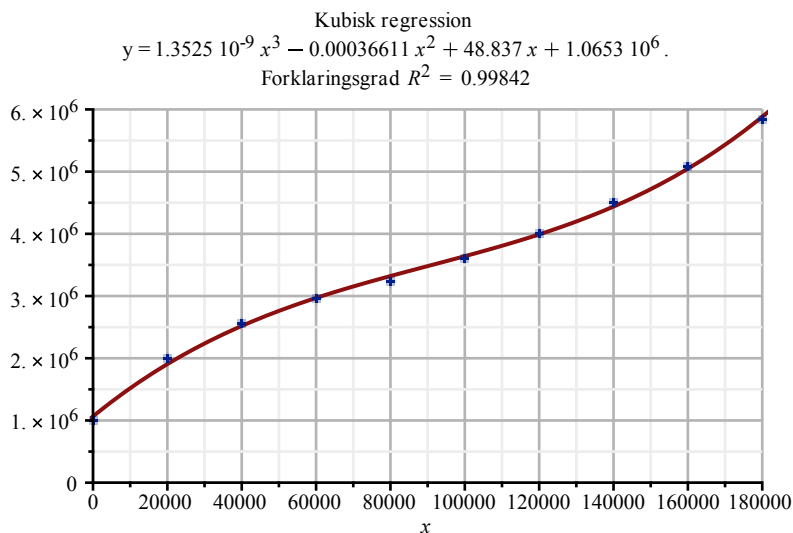
Det har vist sig, at i visse situationer giver PolyReg et resultat, der ligger et pænt stykke fra, hvad man finder i andre værktøjer. Problemet kan opstå, hvis der ikke er data nok til at lave en tilstrækkelig godt estimat for alle parametre i modellen. Hvis det er tilfældet, benytter Maples indbyggede kommando *PolynomialFit* sig af en metode, der hedder Single Value Decomposition. Her er et eksempel på, hvad der kan ske:

with(Gym) :

$$M := \begin{bmatrix} 0. & 1.0000000 \cdot 10^6 \\ 20000.0 & 2.0000000 \cdot 10^6 \\ 40000.0 & 2.5600000 \cdot 10^6 \\ 60000.0 & 2.9600000 \cdot 10^6 \\ 80000.0 & 3.2400000 \cdot 10^6 \\ 100000.0 & 3.6000000 \cdot 10^6 \\ 120000.0 & 4.0000000 \cdot 10^6 \\ 140000.0 & 4.5000000 \cdot 10^6 \\ 160000.0 & 5.0800000 \cdot 10^6 \\ 180000.0 & 5.8400000 \cdot 10^6 \end{bmatrix} :$$

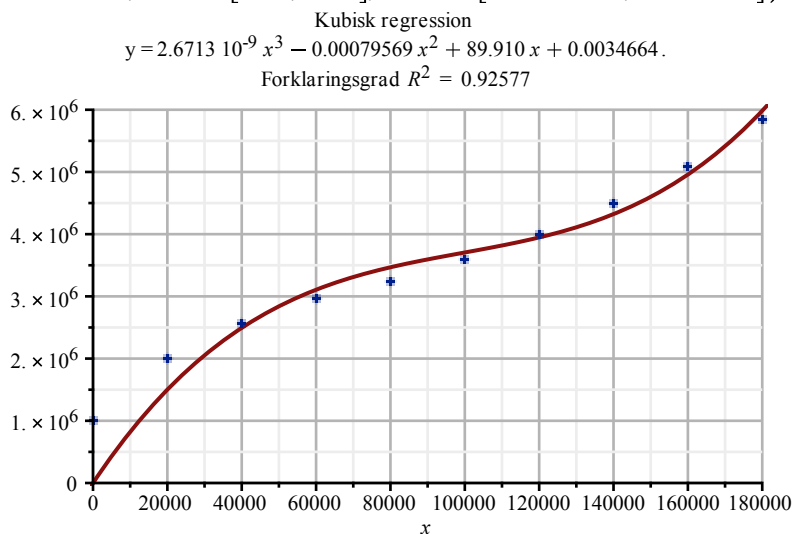
Uden SVD:

PolyReg(M, 3, size = [600, 400], view = [0..180000, 0..6·10⁶])



Med SVD

$PolyReg(M, 3, svd = 10^{-12}, size = [600, 400], view = [0..180000, 0..6 \cdot 10^6])$



Ud fra de to grafer, er det nemt at se, at det bedste resultat opnås uden SVD. I alle andre situationer, der er testet, har der ikke været nogen forskel på, om SVD er slået til eller fra, så derfor er der nu sket den ændring i *PolyReg*, at SVD som standard er slået fra.

Ønskes det imidlertid stået til, skal man sætte *svd* til 10^{-12} . Se kommandoen ovenfor.

Kryptologi

Efter et ønske fra mange sider, er Kryptologi-pakken igen en del af Gym-pakken, men dog en selvstændig del, der fungerer helt uafhængig af Gym-pakken.

Pakken indlæses således:

with(Kryptologi)

[*IC, ICMR, alfaSubs, alfabetglidning, auvray, beaufort, bigrammer, bogstavStatistik, cæsar, (1.1.8.1) friedmanTest, kasiskiTest, mono_break, orloff, periodeTest, trigrammer, vigenere, wheatstone, willard*]

Kryptologi-pakken består af en række rutiner til klassisk kryptologi. Pakken vil senere blive udvidet, så

den også kommer til at omfatte talteori, primtalstest og moderne kryptering.

- (Roteret alfabet (Cæsar systemer) og vilkårligt permuterede alfabeter)
- Polyalfabetiske kryptosystemer (Vigénère, Beaufort, Willard, Auvray)
- Apparater, der ikke længere kan skaffes, men fremstilles som papmodeller (Orloff, Wheatstone)

I e-bogen *Vejledning til Kryptologi-pakken* vil nogle af pakkens rutiner blive behandlet. Beskrivelserne, der ledsages af små instruktive eksempler, vil ikke omfatte alle detaljer. For en mere detaljeret beskrivelse henvises til on-line hjælpen til Kryptologi-pakken.

Rettelser

Excel import af kolonner

Der har været problemer i bl.a. *punktPlot*, hvis data bliver importeret kolonnevist (ad to omgange) fra Excel. Datatypen på de importerede kolonner vil ikke umiddelbart matche de legale datatyper i *punktPlot*, men kunne dog nemt løses ved at konvertere det importerede til fx. vektorer.

Denne konvertering sker nu automatisk.

Manglende afskærmning af lokale variable i LinReg

Hvis x og/eller R var definerede i Mapledokumentet, fik man en fejlmeddelelse i forbindelse med x og regressionen blev ikke udført.

Hvis R var defineret, kom værdien af dette R til at stå opløftet i anden potens som værende lig med foklaringsgraden.

Begge dele er rettet i denne version af Gym-pakken.

Forklaringsgraden i PropReg

Her havde der indsneget sig en fejl i programmeringen, så forklaringsgraden fejlede allerede på 3 decimal. Dette er bragt i orden i den nye version.

residualspredning

Kommandoen *residualspredning* regnede korrekt indtil 10. decimal, hvorefter den fejlede – sikkert som følge af noget afrunding.

Dette er også repareret.

reelSolve

Her var kun mulig at én ligning som input.

Nu kan lister og mængder af ligninger løses.