



Stockholms  
universitet

# Modellpunktsanalys

Maria Käki

Kandidatuppsats 2013:6  
Matematisk statistik  
September 2013

[www.math.su.se](http://www.math.su.se)

Matematisk statistik  
Matematiska institutionen  
Stockholms universitet  
106 91 Stockholm

# Modellpunktsanalys

Maria Käki\*

September 2013

## Sammanfattning

I denna uppsats ska vi behandla ämnet Icke negativ Minstakvadrat Optimering, Nonnegative Least Squares (NNLS). Vi kommer att undersöka om det går att reducera antalet modellpunkter i två dataset. Syftet med att minska antalet modellpunkter är att de beräkningar som genomförs med dataseten går snabbare att genomföra med färre modellpunkter. Viktigt är också att inte förlora signifikant noggrannhet i resultaten då vi minskar antalet modellpunkter. I detta arbete kommer NNLS beskrivas kortfattat, resultat kommer att redovisas från NNLS optimeringen och diskuteras. NNLS fungerar genom att med hjälp av en algoritm hitta de värden på en vektor  $x$  som minimerar minstakvadratfunktionen och med hjälp av denna vektor skapa ett nytt viktat dataset med färre modellpunkter.

---

\*Postadress: Matematisk statistik, Stockholms universitet, 106 91, Sverige.  
E-post: [maria.kakil@gmail.com](mailto:maria.kakil@gmail.com). Handledare: Joanna Tyrcha.

## Abstract

In this thesis we are going to use Nonnegative Least Squares Optimization to investigate if it is possible to reduce the number of model points in two datasets to be able to save time. We will start with a massive dataset from a insurance company, we are going to optimize this dataset and reduce the number of model point without losing significant accuracy. The aim of this thesis is to do this with the Nonnegative Least Squares (NNLS) method. We are going to describe NNLS briefly, results and conclusions from the NNLS optimization are going to be shown and discussed. NNLS works in the following way, given a dataset and a set of observed values we are going to find a nonnegative weight vector by an algorithm. Namely we are going to find the values of the vector  $\mathbf{x}$  which minimizes the least squares function and use these weights to create a new dataset with less model points.

## Förord

Denna uppsats utgör ett självständigt arbete om 15 hp vilket leder till en kandidatexamen i matematisk statistik vid Stockholms Universitet. Arbetet har utförts i samarbete med SEB Trygg Liv och jag vill rikta ett stort tack till Eleonora Andras, aktuarie på Trygg Liv för hjälp och vägledning samt till Harald Nöjd chefsaktuarie på Trygg Liv som har gett mig möjligheten att få genomföra detta arbete. Vidare vill jag rikta ett stort tack till min handledare Joanna Tyrcha professor i Matematisk Statistik för alla värdefulla råd och kommentarer.

# Innehåll

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduktion</b>  | <b>6</b>  |
| <b>2</b> | <b>Metod</b>   | <b>7</b>  |
| 2.1      | Minstakvadratmetoden . . . . .   | 7         |
| 2.2      | Perron Frobenious sats . . . . .   | 9         |
| 2.2.1    | Definition Icke negativ Matris . . . . .   | 9         |
| 2.2.2    | Definition Irreducibel Matris . . . . .  | 9         |
| 2.2.3    | Kriterier som utgör Perron Frobenius sats . . . . .  | 9         |
| 2.2.4    | Karush-Kuhn-Tucker villkoret . . . . .   | 10        |
| 2.3      | Icke negativa Minstakvadratmetoden . . . . .   | 10        |
| 2.3.1    | Allmänt om NNLS . . . . .  | 10        |
| 2.3.2    | Definition NNLS . . . . .  | 11        |
| 2.3.3    | Linjära komplementaritetsproblemet . . . . .   | 11        |
| 2.3.4    | Historia om NNLS . . . . .   | 12        |
| 2.3.5    | NNLS Algoritm . . . . .  | 13        |
| <b>3</b> | <b>Data</b>  | <b>14</b> |
| 3.1      | Privat pensions- och Kapitalförsäkring . . . . .   | 15        |
| 3.2      | Tjänstepension . . . . .   | 16        |
| 3.3      | Riktlinjer . . . . .   | 16        |
| 3.4      | Gamla modellpunktsmetoden . . . . .  | 17        |
| 3.5      | Stress av modellpunkter . . . . .  | 18        |
| <b>4</b> | <b>Resultat</b>  | <b>20</b> |
| 4.1      | Privat pensions- och Kapitalförsäkring . . . . .   | 20        |
| 4.2      | Tjänstepension . . . . .   | 23        |
| 4.3      | Stress av modellpunkter . . . . .  | 26        |
| <b>5</b> | <b>Slutsatser och Diskussion</b>   | <b>27</b> |
| 5.1      | Är det möjligt att minska antalet modellpunkter med Icke negativ minstakvadrat optimering? . . . . . | 27        |
| 5.2      | Hur många modellpunkter erhåller vi med hjälp av NNLS? . . . . .                                     | 27        |
| 5.3      | Hur ser noggrannheten ut hos de nya modellpunkterna? . . . . .                                       | 27        |
| 5.4      | Diskussion . . . . .   | 27        |
| <b>A</b> | <b>NNLS algoritm</b>   | <b>31</b> |
| <b>B</b> | <b>Exempel Icke linjär minstakvadrat</b>   | <b>32</b> |
| <b>C</b> | <b>Bevis av Perron-Frobenius sats</b>  | <b>33</b> |

# 1 Introduktion

Beräkningar på stora dataset kan vara tidskrävande. Vi ska titta närmare på ett fall då syftet är att övergå från en deterministisk till en stokastisk värdering av försäkringar. En följd av detta är att många av de beräkningar som utförs i dataprogrammet *Algo Financial Model*, (AFM) kommer att ta längre tid. AFM är ett modelleringsystem som används av aktuarier för att beräkna risker och värdera information. SEB Trygg Liv använder sig av AFM för att utföra beräkningar, i januari 2014 kommer ett nytt EU-direktiv att träda i kraft. Detta direktiv kallas Solvens II och kommer att medföra att många nya beräkningar på reservsättning kommer att behövas tas fram. Solvens II ersätter gamla EU-direktiv som finns för att säkra att alla försäkringsbolag har tillgångar att ersätta sina försäkringstagare. Då stora dataset används som indata i AFM tar det lång tid att utföra dessa stokastiska beräkningar. Om antalet modellpunkter i indata reduceras minskar datasetets storlek och därmed beräkningstiden i AFM. Det är viktigt att antalet modellpunkter blir så få som möjligt med en god noggrannhet. Indataet heter *policy by policy* (P-by-P), det är skapat av försäkringsbolaget, det innehåller olika förväntade värden för kassaflöden för olika försäkringar. Vi kommer att utgå från P-by-P då vi skapar modellpunkter. Trygg Liv har också utifrån detta dataset skapat modellpunkter, dessa kommer vi att benämna som de gamla modellpunkterna. Vi vill testa om det går att reducera antalet modellpunkter i indata genom att skapa nya modellpunkter från P-by-P. Dessa modellpunkter ska kunna ersätta de gamla. Sättet som vi har valt för att minska antalet modellpunkter i dataset är med hjälp av en dataalgoritm, nämligen Icke negativ minstakvadrat optimering, *Nonnegative Least Squares*, NNLS.

Frågor vi ska besvara i detta arbete är:

- Är det möjligt att minska antalet modellpunkter med Icke negativ minstakvadrat optimering?
- Hur många modellpunkter erhåller vi med hjälp av NNLS?
- Hur ser noggrannheten ut hos de nya modellpunkterna?

## 2 Metod

Vi kommer att utgå från indata och med hjälp av NNLS erhålla vikter som vi kan använda för att minska datasetet. En vikt, är ett ändligt reellt tal större än eller lika med noll som anger hur många gånger vi ska räkna varje försäkring. Vi kommer att erhålla en vikt för varje försäkring. När vi använder NNLS kommer ett antal försäkringar få vikt noll. De försäkringar som får vikt noll kommer inte att ha någon betydelse för datasetet och därmed kan vi skapa ett nytt dataset utan de försäkringar som erhållit vikt noll. På detta sätt erhåller vi ett dataset med färre modellpunkter. NNLS är en algoritm som kan användas i många olika dataprogram, vi väljer att använda oss av programmet R<sup>1</sup>. Vi kommer att beskriva hur datasetet är uppbyggt samt gå igenom hur vi kan minska datasetet med hjälp av NNLS.

### 2.1 Minstakvadratmetoden

Minstakvadratmetoden är en approximativ metod som används för att skatta lösningar till överbestämda system av ekvationer. Ett överbestämt system är ett system som innehåller fler ekvationer än variabler. Dessa system löses ekvationerna så att residualkvadratsumman minimeras. Lösningen beräknas med hjälp av ortogonal matrisfaktorisering. Minstakvadratproblem delas in i två grupper, linjära minstakvadratproblem och icke linjära minstakvadratproblem. Linjära minsta kvadratproblem förekommer oftast stängd form vilket innebär att de kan lösas med ett ändligt antal standardoperationer. Här följer ett exempel på kurvanpassning med hjälp av minsta kvadratoptimering.

#### Exempel minstakvadratoptimering

Låt  $t$  vara en oberoende variabel och låt  $b(t)$  vara en okänd funktion av  $t$  som vi vill skatta. Antag att vi har  $n$  stycken observationer, det vill säga värdena på  $b$  mätta vid olika värden på  $t$ .

$$b_i = b(t_i), \quad i = 1, \dots, n.$$

Idén är att skapa  $b(t)$  genom en linjär kombination av basfunktioner:

$$b(t) \approx A_1\phi_1(t) + \dots + A_m\phi_m(t).$$

Designmatrisen som vi definierar som  $\mathbf{x}$  är en rektangulär matris av ordning  $n$  med  $m$  element. Med designmatris menas en matris som består av förklarande variabler.

$$x_{i,j} = \phi_j(t_i), \quad i = 1, \dots, n \quad j = 1, \dots, m.$$

---

<sup>1</sup><https://www.r-project.org>



Designmatrisen som vi vill hitta har fler rader än kolumner. I matrisnotation kan vi skriva den på formen:

$$\mathbf{b} \approx \mathbf{A}\mathbf{x}. \quad (1)$$

Basfunktionerna  $\phi_j(t)$  kan vara icke linjära funktioner av  $t$  men behöver inte vara det. De okända variablerna,  $A_j$ , är linjära i modellen. Systemet av linjära ekvationer (1) är ett överbestämt system. Det har alltså fler ekvationer än antalet variabler som vi vill skatta. Residualerna är skillnaden mellan det faktiska observerade värdet och det värde som modellen föreslår. Matematiskt skrivs residualen

$$r_i = b_i - \sum_j A_j \phi_j(t), i = 1, \dots, n$$

Vi vill alltså hitta den designmatris som ger den minsta möjliga residualkvadratsumman. Sammanfattat, minstakvadratmetoden minimerar summan av residualerna i kvadrat som matematiskt skrivs

$$\min \|\mathbf{r}^2\| = \min \sum_i^n r_i^2.$$

Icke linjära minstakvadratproblem kan formuleras på följande sätt enligt [7]:

$$\text{minimera } F(x) = \frac{1}{2} \sum_i^n (f_i(x))^2 \quad (2)$$

där  $f_1, \dots, f_n$  är givna funktioner från  $\mathbb{R}^n$  till  $\mathbb{R}$ . Icke linjära minsta kvadratproblem har ingen lösning på stängd form utan måste lösas numeriskt. Detta kan göras genom olika algoritmer. Lösningen till ett icke linjärt minstakvadratproblem kallas för Gauss-Newton lutningen och metoden för att hitta Gauss-Newton lutningen kallas för Gauss-Newton Metoden. I Appendix B finns Gauss-Newton metoden beskriven, där visar vi att minstakvadratproblemet (2) kan kopplas till linjär approximation och användas på ett icke linjärt problem. Lösningen till minstakvadratproblem svarar mot maximum likelihood kriteriet, vilket innebär att det är den mest sannolika lösningen för problemet. Denna lösningen kallas maximumlikelihoodskattaren har en normalfördelning och kan härledas som momentmetodsskattaren. Ett exempel på användningsområde för minstakvadratmetoden är att anpassa en *Generalized Linear Model* GLM modell genom att succesivt upprepa lokal minsta kvadrat approximation av likelihoodfunktionen. Eftersom att det inte alltid är möjligt att välja en modell som passar för alla värden väljs istället den modell som har den minsta residualkvadratsumman, därav kommer namnet minstakvadratmetoden.

## 2.2 Perron Frobenious sats

För att kunna använda oss av NNLS behöver vi göra vissa antaganden angående matrisen som representerar datasetet. Förutom att matrisen måste vara icke negativ måste den även vara irreducibel och anpassad enligt de kriterier som utgör Perron Frobenius Sats. Alla matriser som tas upp här består av reella tal. Följande sats finns i flera formuleringar, vi har valt formuleringen från [10].

### 2.2.1 Definition Icke negativ Matris

Matrisen  $\mathbf{A}$  är icke negativ om  $\mathbf{A} \geq \mathbf{0}$  och alla element  $a_{ij} \geq 0$  för alla  $i$  och  $j$ . Det följer även att transponatet  $\mathbf{A}^T$  är en icke negativ matris. Linjära system av ekvationer  $\mathbf{Ax} = \mathbf{b}$  har icke negativa lösningar när  $\mathbf{b} \geq \mathbf{0}$ .

### 2.2.2 Definition Irreducibel Matris

En  $n \times n$  matris  $\mathbf{A}$  säges vara reducibel om  $n \geq 2$  och det existerar en permutationsmatris  $\mathbf{P}$  som uppfyller ekvationen

$$\mathbf{PAP}^T = \begin{pmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}.$$

Där  $\mathbf{P}$  är en  $n \times n$  matris,  $\mathbf{B}$  och  $\mathbf{D}$  är kvadratiska matriser,  $\mathbf{0}$  är en nollmatris och  $\mathbf{C}$  är en matris av samma storlek som nollmatrisen. Matrisen  $\mathbf{A}$  säges vara irreducibel om den inte är reducibel.

### 2.2.3 Kriterier som utgör Perron Frobenius sats

Låt  $\mathbf{A}$  vara en icke negativ irreducibel  $n \times n$  matris. För att tillämpa NNLS måste  $\mathbf{A}$  uppfylla följande villkor:

1.  $\mathbf{A}$  har ett positivt reellt egenvärde  $\lambda_{max}$  som är större än eller lika med absolutbeloppet av alla andra egenvärden  $\lambda$ .
2.  $\lambda_{max}$  är algebraiskt och geometriskt simpel, det vill säga att  $\lambda_{max}$  har algebraisk multiplicitet ett samt att alla par av egenvektorer som svarar mot  $\lambda_{max}$  är linjärt oberoende.  $\lambda_{max}$  har en egenvektor  $\mathbf{x}$  med  $\mathbf{x} > \mathbf{0}$ .
3. Alla icke negativa egenvektorer är multiplar av  $\mathbf{x}$ .
4. Om  $\mathbf{y}$  är en vektor vars element är större än eller lika med noll och  $\mathbf{y} \neq \mathbf{0}$ , och  $\mu$  är ett tal som uppfyller  $\mathbf{Ay} = \mu\mathbf{y}$  är mindre eller lika med  $\mu\mathbf{y}$  då  $\mathbf{y}$  är större än noll, och  $\mu$  är större eller lika med  $\lambda_{max}$  med  $\mu = \lambda_{max}$  om och endast om  $\mathbf{y}$  är en multipel av  $\mathbf{x}$ .

5. Om det existerar en matris  $\mathbf{S}$  sådan att  $\mathbf{0} \leq \mathbf{S} < \mathbf{A}$ . Då kommer varje egenvärde  $\sigma$  till  $\mathbf{S}$  satisfiera  $|\sigma| < \lambda_{max}$ .
6. Alla antidiagonaler till  $A_i$  har egenvärden vars absolutbelopp är mindre än  $\lambda_{max}$ .  $A_i$  fås av  $\mathbf{A}$  genom att ta bort den  $i$ -te raden och  $i$ -te kolumnen. En antidiagonal är diagonalen från övre högra hörnet till det vänstra nedre hörnet.
7. Om  $\mathbf{A}$  är primitiv, då är absolutbeloppet av alla egenvärden  $\lambda$  till  $\mathbf{A}$  mindre än  $\lambda_{max}$ .

### 2.2.4 Karush-Kuhn-Tucker villkoret

Karush-Kuhn-Tucker (KKT) villkoret måste vara uppfyllt för att en lösning ska vara optimal (vilket i vårt fall innebär den lösning med lägst residu-alkvadratsumma). Detta villkor är nödvändigt för att lösningen ska vara optimal men dock inte tillräckligt för att visa att en lösning är optimal. Följande formulering av Karush-Kuhn-Tucker villkoret finns i [4]. Låt  $f$  vara en objektiv funktion, med en objektiv funktion menas en kostnads- eller förlustfunktion. Låt  $h$  och  $g$  vara begränsningsfunktioner. Vi ska hitta det värde på vektorn  $\mathbf{x}$  som ger det minsta värdet på funktionen. Låt  $\mathbf{x}_{min}$  vara det lokala minimum av funktionen  $f$  som uppfyller:

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \begin{cases} h(\mathbf{x}) = 0 \\ g(\mathbf{x}) \leq 0 \end{cases}$$

Antag att  $\mathbf{x}_{min}$  ligger innanför dessa begränsningarna, då kommer Jacobi-anen av  $\mathbf{x}_{min}$  vara av full rang. Detta innebär att det existerar  $\lambda$  och  $\mu$  sådana att

$$\begin{aligned} \Delta f(\mathbf{x}_{min}) + \lambda^T \Delta h(\mathbf{x}_{min}) + \mu^T \Delta g(\mathbf{x}_{min}) &= 0 \\ \mu^T g(\mathbf{x}_{min}) &= 0 \\ h(\mathbf{x}_{min}) &= 0 \\ \mu &\geq 0 \end{aligned}$$

## 2.3 Icke negativa Minstakvadratmetoden

### 2.3.1 Allmänt om NNLS

Det är svårt att hantera stora datamängder, NNLS är en metod för att reducera datasetets storlek vilket gör det lättare att arbeta med. Följande exempel förklarar grundtanken bakom NNLS.

## Exempel NNLS

Vi antar att det har gjorts flera observationer av linjära funktioner, där varje linjär funktion beror av variabler. Givet ett tillräckligt stort antal av sådana observationer kan vi på ett tillförlitligt sätt skatta de sanna variablerna. Vi låter variablerna skrivas som en vektor  $\mathbf{x} = (x_1, \dots, x_n)^T$ . Våra observationer av olika värden på variablerna representeras av matrisen  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $m$  är antalet rader och  $n$  är antalet kolumner i matrisen  $\mathbf{A}$ . Mängden av observerade värden (summan av alla värden för varje variabel) representeras av matrisen  $\mathbf{b} \in \mathbb{R}^{m \times 1}$ . Målet är att konstruera en vektor  $\mathbf{x} \in \mathbb{R}^{n \times 1}$  som representerar de observerade värdena på bästa sätt. Metoden för att lösa detta är att beräkna  $\mathbf{x}$  för att residualkvadratsumman

$$f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|^2 \quad (3)$$

ska bli så liten som möjligt. NNLS är ett specialfall av icke negativa minstakvadratproblem. I icke linjära minstakvadratproblem där variablerna endast antar icke negativa värden som pixlar, kemiska koncentrationer eller kassaflöden kan dessa bli svåra att lösa. Detta kan formuleras som ett icke linjärt minstakvadratproblem med bivillkor att variablerna bara får anta icke negativa värden. Därför måste residualen begränsas till att bara inkludera icke negativa värden på modellvariabeln  $\mathbf{x}$ .

### 2.3.2 Definition NNLS

Givet en matris  $\mathbf{A} \in \mathbb{R}^{m \times n}$  och en mängd av observerade värden  $\mathbf{b} \in \mathbb{R}^m$ , hitta en icke negativ vektor  $\mathbf{x} \in \mathbb{R}^n$  som minimerar funktionen  $f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|^2$ . Följande definition finns i [4]

$$\min f(\mathbf{x}) = \|\mathbf{Ax} - \mathbf{b}\|^2 \quad (4)$$

med bivillkor att  $\mathbf{x}$  är större än eller lika med noll. Gradienten till  $f(\mathbf{x})$  är

$$\nabla f(\mathbf{x}) = \mathbf{A}^T(\mathbf{Ax} - \mathbf{b})$$

Vilket ger att KKT villkoren vi behöver för NNLS problem är för  $\mathbf{x}$  större än eller lika med noll.

$$\begin{aligned} \nabla f(\mathbf{x}) &\geq 0 \\ \nabla f(\mathbf{x})^T \mathbf{x} &= 0 \end{aligned}$$

### 2.3.3 Linjära komplementaritetsproblemet

Linjära komplementaritetsproblemet, *Linear Complementary Problem* (LCP), är en viktig del i NNLS. För att lösa ekvation (2) upprepas metoden som

löser ekvationen för LCP.

### Exempel LCP

Givet en matris  $\mathbf{A} \in \mathbb{R}^{m \times n}$  och mängden av observerade värden  $\mathbf{b} \in \mathbb{R}^m$  hitta en vektor  $\mathbf{x} \in \mathbb{R}^n$  som minimerar funktionen

$$\lambda = \nabla f(\mathbf{x}) = \mathbf{A}^T \mathbf{A} \mathbf{x} - \mathbf{A}^T \mathbf{b} \geq 0$$

$$\mathbf{x} \geq 0$$

$$\lambda^T \mathbf{x} = 0$$

LCP kan ses som mängden av de KKT villkor som används för kvadratisk programmering. Uppgiften är alltså att hitta ett icke negativt  $\mathbf{x}$  som löser ekvationen  $(\mathbf{A}\mathbf{x} - \mathbf{b})^T \mathbf{A}\mathbf{x} = 0$ . Att hantera icke negativa begränsningar när man arbetar med stora mängder av icke linjära ekvationer är svårt. En liknande men mer lätthanterlig formulering av NNLS är att använda sig av residualvektorvariabeln. Den definieras som  $\mathbf{p} := \mathbf{b} - \mathbf{A}\mathbf{x}$ . Vi kan med denna definition av  $\mathbf{p}$  använda oss av

$$\min_{\mathbf{x}, \mathbf{p}} \frac{1}{2} \mathbf{p}^T \mathbf{p}$$

för att erhålla ekvationen

$$\mathbf{A}\mathbf{x} + \mathbf{p} = \mathbf{b}, \mathbf{x} \geq 0$$

Fördelen med denna ekvation är att det är en enklare och separerbar funktion som också innehåller linjära icke negativa begränsningar.

### 2.3.4 Historia om NNLS

Sedan 1990-talet har NNLS beräkningar använts för att skatta icke negativa matriser och tensorfaktoriseringar. En tensor är en multidimensionell vektor, ordet tensor är en generalisering av vektorer, matriser och skalärer. Albert Einstein var en av de första som använde tensorer, han använde dem för att beskriva fysikens lagar. Efter detta har olika tensoranalystekniker blivit kända. Tensoranalyser har flera användningsområden men lämpar sig speciellt bra för stora dataset. NNLS används på stora icke negativa dataset, för att få dataset med samma egenskaper men av lägre dimension. Genom att använda sig av olika bivillkor och begränsningar med låg rang på ett dataset kan denna metod reducera antalet dimensioner i datasetet. Det finns flera olika metoder för att reducera dimensioner i matriser och dataset. Några exempel är:

- Principiell komponentanalys, *Principal Component Analysis* (PCA)

- Oberoende komponentanalys *Independent Component Analysis* (ICA)
- Approximativ icke negativ matrisfaktorisering *Approximative Nonnegative Matrix Factorization* (NMF)
- Icke negativ tensorfaktorisering *Nonnegative Tensor Factorization* (NTF)

NNLS tillhör NTF. De dataset som behandlas med NTF består av matriser som är representerade av tensorer. Några exempel på användningsområden för denna typ av beräkningar är i numerisk analys, Markov Kedje analys, ekonomiska undersökningar och statistik.

### 2.3.5 NNLS Algoritmen

NNLS algoritmen beräknar lösningen till en vektor,  $\mathbf{x}$ . Denna vektor löser minsta kvadratproblemet, där  $\|\mathbf{Ax} - \mathbf{b}\|$  minimeras med bivillkor att  $\mathbf{Gx} \geq \mathbf{h}$ .  $\mathbf{A}$  är en  $m \times n$  matris och  $\mathbf{b}$  är en vektor med  $m$  element. Det som skiljer NNLS från andra minstakvadratoptimeringar är att  $\mathbf{A}$  i andra minstakvadratoptimeringar brukar vara en godtycklig  $m \times n$  matris, för NNLS är  $\mathbf{G}$  är en identitetsmatris och  $\mathbf{h}$  är en nollvektor. Detta problem har alltid minst en lösning men det finns bara en unik lösning om rangen för  $\mathbf{A}$  är mindre än  $n$ . Detta gör också att lösningen kommer att innehålla minst  $n - m$  element som är lika med noll. Algoritmen finns i Appendix A. Det går att bevisa att denna algoritm konvergerar till ett ändligt antal upprepningar och därmed kan en lösning erhållas utan att behöva avsluta algoritmen innan ekvationen är fullständigt löst. Beviset för detta finns i Lawson och Hansson [3].

### 3 Data

Det dataset vi utgår från då vi ska skapa modellpunkter heter P-by-P. Vi har tillgång till två olika dataset som är konstruerade utifrån P-by-P, det första av dem består av försäkringar för Privat pensions- och Kapitalförsäkring (PK) och det andra består av försäkringar för Tjänstepension (T). Dessa två dataset innehåller samma variabler men kommer ändå att behandlas separat. De behandlas separat eftersom att allt eventuellt överskott som blir i PK portföljen ska återinvesteras i den och samma sak gäller för T. Vi vet också att försäkringarna i T är mer homogena än försäkringarna i PK. Dataseten PK och T kan ses som matriser där varje kolumn i matrisen motsvarar en försäkring och varje rad motsvarar en variabel. Varje försäkring i  $\mathbf{A}$  kan ses som en tensor. Datasetet T innehåller 57949 försäkringar, och datasetet PK innehåller 21881 försäkringar. Varje försäkring motsvarar en modellpunkt och då beräkningar genomförs med dataseten gäller det att ju färre modellpunkter ett dataset har desto kortare blir vår beräkningstid. För varje försäkring har vi värden på förväntade kassaflöden för våra variabler från år 0 som är beräknat per den 30:e juni 2012 och framåt i tiden. Antalet variabler är samma för de båda dataseten. Totalt har vi 20 olika variabler, 15 av de variablerna är kassaflöden, 1 variabel representerar antal försäkringar, 2 variabler är nuvärdesberäknade kassaflöden och 2 variabler svarar mot reservsättning. Åren vi har tillgång till för variablerna är varje år fram till och med år 51 därefter finns var 5:e till 10:e år representerade. För alla variabler har vi alltså tillgång till år 1-51, 55, 60, 65, 70 och 80. Det finns dock vissa variabler som även innehåller år 0, dessa variabler är *number of policies*, *PV Cashflow FDB*, *PV Cashflows Stoch*, *Reserve 2:nd order* samt *Reserve tot*. Varje år i varje variabel motsvarar en rad i matrisen  $\mathbf{A}$ . För att utföra NNLS behöver vi ett aggregerat dataset,  $\mathbf{b}$ , detta dataset består av summan av alla försäkringarna för varje år. Det vill säga det aggregerade datasetet  $\mathbf{b}$  är ekvivalent med radsummorna i  $\mathbf{A}$ . Första elementet är summan av alla försäkringars förväntade kassaflöden för variabel  $V_1$  år 1. I tabellen nedan finns de variabler som är representerade i våra dataset, samt hur många olika år vi har värden för på respektive variabel.

|    | Variabel                     | Antal år |
|----|------------------------------|----------|
| 1  | CF Asset management fee FDB  | 56       |
| 2  | CF Asset management fee Gtee | 56       |
| 3  | CF Benefit Death FDB         | 56       |
| 4  | CF Benefit Death Gtee        | 56       |
| 5  | CF Benefit Mat FDB           | 56       |
| 6  | CF Benefit Mat Gtee          | 56       |
| 7  | CF Benefit Sur FDB           | 56       |
| 8  | CF Benefit Sur Gtee          | 56       |
| 9  | CF Charge tax reduction FDB  | 56       |
| 10 | CF Charge tax reduction Gtee | 56       |
| 11 | CF Comm total                | 56       |
| 12 | CF Expense total             | 56       |
| 13 | CF FDB                       | 56       |
| 14 | CF Premium total             | 56       |
| 15 | CF Total                     | 56       |
| 16 | Number of policies           | 57       |
| 17 | PV Cashflows FDB             | 57       |
| 18 | PV Cashflows Stoch           | 57       |
| 19 | Reserve 2:nd                 | 57       |
| 20 | Reserve Tot                  | 57       |

Alla dessa variabler ger oss totalt  $m=1125$  rader i matris  $\mathbf{A}$ . Vi låter  $V_j$  stå för variabel  $j$ ,  $j=1,\dots,m$  och  $F_i$  för försäkring  $i$ ,  $i=1,\dots,n$ , detta ger att för element  $a_{ij} \in \mathbf{A}$  är  $a_{ij} = V_j F_i$ . Vi kan alltså skriva,

$$\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} \sum_i^n a_{1,i} \\ \vdots \\ \sum_i^n a_{m,i} \end{pmatrix}$$

Det finns tre stycken variabler som är extra viktiga för beräkning av reservsättningen. Dessa variabler är *CF Premium Total*, *Reserve 2:nd* och *Reserve Tot*, vi benämner dessa variabler som nyckelvariabler.

### 3.1 Privat pensions- och Kapitalförsäkring

PK innehåller alla variablerna ovan och värdena är hämtade från 21881 försäkringar från privat pensions- och kapitalförsäkring. Alla värdena finns i P-by-P. Vi kommer att behandla data som en stor matris  $\mathbf{A}$  med  $m=1125$  rader, och  $n=21881$  kolumner. Utifrån detta dataset kan vi skapa ett aggregerat dataset för varje variabel vilket betecknas med  $\mathbf{b}$ .  $\mathbf{b}$  är en  $m \times 1$  matris



som innehåller alla aggregerade värden för varje variabel. När vi skattar vektorn  $\boldsymbol{x}$  med NNLS erhålls en vektor av dimension  $n \times 1$  för detta dataset.

### 3.2 Tjänstepension

T innehåller alla variablerna ovan och värden som är hämtade från 57949 försäkringar från Tjänstepension. Denna data behandlas som en matris  $\boldsymbol{A}$  som innehåller  $m=1125$  rader och  $n=57494$  kolumner. Eftersom dataseten är lika och vi kommer att använda oss av samma metod kommer vi nedan bara att använda notationen  $\boldsymbol{A}$ . Vi kommer även att ha tillgång till variabler för hela den grupperade portföljen för respektive dataset vilket betecknas med matrisen  $\boldsymbol{b}$ .

### 3.3 Riktlinjer

Modellpunkter är konstruerade försäkringsavtal som representerar flera försäkringsavtal med liknande egenskaper. Modellpunkter bör i teorin uppträda som en aggregering av de försäkringar som de representerar. Dock har en perfekt matchning, det vill säga att modellpunkter uppträder exakt som originaldata, inte varit möjlig i praktiken. Därför finns riktlinjer som har satts upp av försäkringsbolaget, riktlinjerna rör hur stor differens som accepteras mellan det resultat som erhålls från modellpunkter jämfört med resultat som erhålls från P-by-P. För våra nyckelvariabler accepteras en differens på 1% och för övriga variabler accepteras en differens på 5%. Vi kommer att använda dessa riktlinjer för att testa om det dataset som erhållts med hjälp av NNLS optimeringen är tillräckligt bra för att testa att köra i AFM. Med tillräckligt bra menas att de uppfyller nedanstående kriterier. Våra nyckelvariabler testades alltså enligt följande kriterium

$$0.99 \leq \frac{\sum_{i=1}^n x_i F_i}{\sum_{i=1}^n F_i} \leq 1.01 \quad (5)$$

medan övriga variabler testades enligt nedanstående kriterium

$$0.95 \leq \frac{\sum_{i=1}^n x_i F_i}{\sum_{i=1}^n F_i} \leq 1.05, \quad (6)$$

där  $F_i$  är policy  $i = (1, \dots, n)$ . Vektorn  $\boldsymbol{x}$  är den som erhållits då vi genomfört en NNLS optimering med datasetet. Alla kassaflöden är nuvärdesberäknade genom att multiplicera med en diskonteringsfaktor

$$d_k = \frac{1}{(1+r)^k} \quad (7)$$

$d = (d_1, \dots, d_{80})$ , där  $k$  är antalet år  $(1, \dots, 80)$  och  $r$  är räntan. Räntan i AFM programmet tas från en räntekurva, vilket innebär att varje år  $i$  har en egen ränta  $r_i$ . Räntekurvan beräknas dels med en jämn kurva och dels med en stokastisk kurva. För att förenkla vårt arbete lite kommer vi att räkna med en konstant ränta,  $r=2.5\%$ . Vi har valt den räntanivån för att den verkliga räntan förväntas variera mellan 2% och 3 %.

Vårt nuvärde blir alltså

$$\widehat{PV}(p) = \sum_{i=1}^n \hat{p}_i \times d \quad (8)$$

där  $d$  är diskonteringsfaktorn. Kvoterna (6) och (5) kallar vi för våra teoretiska noggrannhetskvoter. De teoretiska noggrannhetskvoterna används för att titta på noggrannheten på alla variabler för att undersöka om det är lämpligt att köra det erhållna datasetet i AFM. För att det ska vara lämpligt bör alla variabler ligga innanför riktlinjerna, dock kan en högre avvikelse tillåtas på variabler med låga belopp. Detta på grund av att det är totalsumman vi i första hand intresserar oss av och den mäts bäst med våra tre nyckelvariabler. Om datasetet sedan testas i AFM erhålls en verklig noggrannhetskvot, den verkliga noggrannhetskvoten är ofta nära den teoretiska noggrannhetskvoten. Det är den verkliga noggrannhetskvoten som vi använder oss av när vi bedömer om våra vikter,  $\boldsymbol{x}$ , är användbara.

### 3.4 Gamla modellpunktsmetoden

Den tidigare metoden för att ta fram modellpunkter, gamla modellpunktsmetoden, genomförs i två delsteg. I det första steget skapas modellpunkterna utifrån P-by-P genom att dela in försäkringarna i olika grupper och i det andra delsteget reduceras antalet modellpunkter. Data som finns representerad i P-by-P kan delas in i tre kategorier: numeriska värden, karaktäristiska värden och datum. Med hjälp av data kan försäkringarna delas upp i olika undergrupper. För numeriska värden tas medelvärden, karaktäristiska värden delas in i grupper, till exempel kön kan ha grupperna man och kvinna. Datum får stå kvar som datum, på detta sätt erhålls flera kategorier, utifrån det beräknas antal försäkringar i varje kategori. Detta förfarande kan beskrivas med följande förenklade exempel.

#### Exempel gamla modellpunktsmetoden

Vi har ett försäkringsbestånd av 100 försäkringstagare och vi är intresserade av att dela in efter två kategorier, kön och prisklass. Kön kan anta

värdena kvinna eller man och prisklass kan anta värdena A, B eller C. Varje ruta innehåller antalet försäkringar som faller in under respektive kategori. Till exempel har vi 20 män som tillhör prisklass A, 10 kvinnor som tillhör prisklass A, 25 män som tillhör prisklass B, 15 kvinnor som tillhör prisklass B, 15 män som tillhör prisklass C och 15 kvinnor som tillhör prisklass C. Då erhåller vi följande rutnät:

|   | Man | Kvinna |
|---|-----|--------|
| A | 20  | 10     |
| B | 25  | 15     |
| C | 15  | 15     |

Detta exempel innehåller sex stycken modellpunkter med vikter 20, 10, 25, 15, 15, 15. I det andra steget reduceras antalet modellpunkter, detta genomförs med en slags trial and error process då varje kategori analyseras var för sig för att sedan bedömas om de tros vara nödvändiga. Till exempel kan vi reducera antalet variabler i prisklass från tre stycken (A, B, C) till två stycken (A, B) om prisklass B och C innehåller samma egenskaper för den totala reservsättningen. Då erhåller vi istället följande rutnät:

|   | Man | Kvinna |
|---|-----|--------|
| A | 20  | 10     |
| B | 40  | 30     |

Då har vi lyckats reducera antalet modellpunkter från sex stycken till fyra stycken och de nya modellpunkterna får vikterna 20, 10, 40 och 30. I fallet med P-by-P datasetet erhålls betydligt fler kategorier och efter full bearbetning nåddes ett resultat med drygt 2000 modellpunkter. Nackdelen med denna metod är att så fort en variabel reducerats måste modellen köras i AFM för att kunna utvärdera resultatet, detta blir alltså en typ av trial and error process. En till egenskap hos denna modellpunktsmetod är att det krävs stor kunskap om de underliggande variablerna för att kunna göra en så effektiv gruppindelning som möjligt. Oavsett blir det en tidskrävande process som innefattar en svår avvägning. Desto mer specificerad kategori indelning desto fler modellpunkter erhålls, fler modellpunkter ger dock en högre noggrannhet och även därför krävs en stor bakgrundskunskap om försäkringsbeståndet för att kunna genomföra gruppering av data på ett korrekt sätt. Detta gör att denna modellpunktsmetod således blir svår, ineffektiv och tidskrävande.

### 3.5 Stress av modellpunkter

Ett sätt att testa de erhållna modellpunkterna ytterligare är att stressa dem. När man stressar modellpunkter betyder det att man använder modellpunkterna som indata i ett program i AFM. Detta program kör olika ekonomiska

scenarion, det vill säga att variablerna beter sig på olika sätt som förutspås utefter olika möjliga ekonomiska scenarion som kan inträffa. En stress testar fem olika scenarion och när vi kör P-by-P datat och våra modellpunkter i stressen vill vi att noggrannheten för våra modellpunkter ska ligga innanför de riktlinjer vi har. Vi får alltså ett resultat för varje stress och vi kommer att redovisa siffrorna för en utvald stress för PK samt för en utvald stress för T.

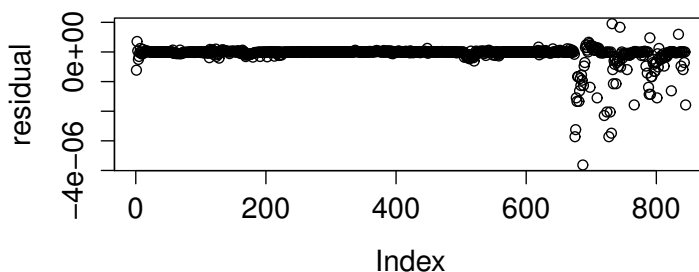
## 4 Resultat

Vi undersöker om de modellpunkter vi erhåller med hjälp av NNLS kan ersätta de gamla modellpunkterna. Vi tittar främst på noggrannheten för våra nyckelvariabler, det vill säga hur mycket resultatet skiljer sig i procent då vi kör våra modellpunkter i AFM samt då vi kör P-by-P i AFM. När vi kör modellpunkterna för att testa i AFM erhåller vi en output för 17 variabler. Därför redovisas resultatet för dessa 17 variabler då vi jämför noggrannhetskvoterna. Detta resultat kommer att jämföras med hur mycket de gamla modellpunkterna skiljer sig från P-by-P i procent då man kör dem i AFM. En jämförelse görs för att se hur procenttalen på de nya modellpunkterna ser ut relativt procenttalen för de gamla. Vi vill uppnå den optimala noggrannheten på 100% och desto närmare 100 % vi kommer desto bättre. Vi kommer också att undersöka om antalet modellpunkter vi erhåller med NNLS är färre än antalet modellpunkter som erhålls med hjälp av den gamla metoden. Med hjälp av den gamla metoden erhålls ett dataset med drygt 2000 modellpunkter för både PK och T. Noggrannheten för variablerna redovisas i andra kolumnen i tabellerna nedan. Vi har kört flera omgångar då vi tagit med olika variabler i NNLS optimeringen för att se hur resultatet skiljer sig. Vi har först kört hela dataseten och därefter plockat bort en variabel i taget för att se om resultatet ger färre modellpunkter och fortfarande är innanför de riktlinjer vi har satt upp. De som uppfyllt kraven på noggrannheten har sparats som nya dataset och sedan har vi eliminerat ytterligare en variabel. För de som fortfarande erhållit färre modellpunkter och hamnat innanför riktlinjerna fortsätter vi eliminera variabler. Vi kommer endast att redovisa de två dataset som gav bäst resultat för PK respektive T, totalt fyra stycken. Målet är att få så bra noggrannhetskvoter som möjligt för våra nyckelvariabler. För att modellpunkterna ska vara användbara krävs också att noggrannhetskvoterna ligger innanför riktlinjerna, det är även önskvärt att det ska vara betydligt färre än de gamla modellpunkterna som använts tidigare. Om vi erhåller två dataset med lika bra noggrannhetskvoter är det dataset som innehåller färre modellpunkter att föredra.

### 4.1 Privat pensions- och Kapitalförsäkring

Detta dataset optimeras med hjälp av NNLS. Vid användning av alla variabler erhålls ett resultat som ger oss en dålig noggrannhetskvot för merparten av variablerna. Vi testar att ta bort en variabel i taget, här visar det sig att de dataset som erhöll en högre noggrannhetskvot var de där vi tagit bort variabler 1, 3, 5, 7 eller 9 (de med namn FDB). Därefter gick vi vidare och tog bort två variabler i taget, de bästa resultaten gavs av de dataset där vi tagit bort två av variablerna från FDB. Denna metod fortsatte och vi eliminerade succesivt variabler. Det slutliga och bästa resultatet för PK datasetet är det där vi använder oss av alla variabler utom FDB variablerna.

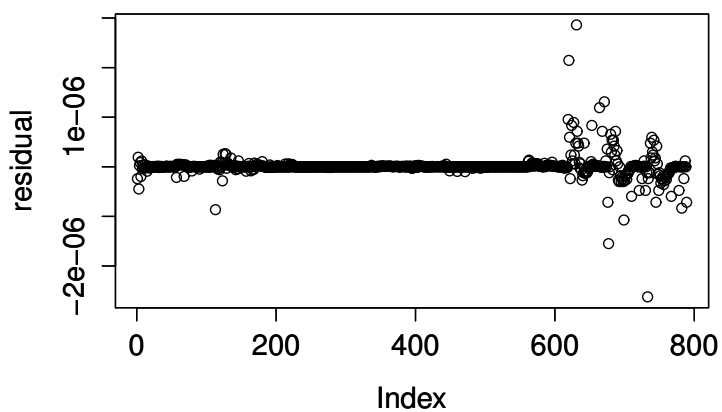
Vi testar att ta bort motsvarande variabler för Gtee och ta bort variabel 2, 4, 6, 8, eller 10. Successiv eliminering som ovan för att slutligen ta bort alla. Resultatet då vi tagit bort Gtee variablerna blev dock sämre än resultatet då vi tagit bort FDB variablerna. Vi behöll dock FDB total som är totalsumman av de övriga variablerna som slutar på FDB, om man eliminerar denna variabel också blir noggrannhetskvoterna längre från 100%. När vi tittar på noggrannheten i resultatet så redovisas noggrannheten även för de bortvalda variablerna. Resultatet som erhålls från denna optimering kallar vi för resultat a), och dataset PK a). PK a) är alltså datasetet PK som innehåller alla variabler utom 1, 3, 5, 7 och 9. När försäkringsbeståndet PK a) optimeras med hjälp av NNLS erhålls en vektor  $\mathbf{x}$  av dimension  $n \times 1$  med  $n = 21881$ . Denna vektor innehåller 21040 element som är lika med noll, detta ger därför 841 stycken modellpunkter. Residualkvadratsumman är  $1.976 \times 10^{-20}$ . Tittar vi på residualplottarna kan vi se att residualen är större för variablerna med höga index, PV cashflow FDB och PV Cashflow Stoch, Reserve 2:nd order och Reserve total, dessa variabler har index 621 till 845. Att residualerna blev större för dessa värden påverkar inte noggrannheten i dessa variabler lika mycket procentuellt då de består av större belopp. Noggrannheten för de utvalda variablerna och dessa modellpunkter redovisas i tabell 4.1 .



Figur 1: Residualplott för modellpunktsuppsättning a) för dataset PK. På y-axeln så har vi c som representerar residualens storlek och på x-axeln så har vi variabel index.

b) Det dataset som ger näst bäst noggrannhetskvoter är då vi utesluter variabel 1, 3, 5, 7, 9 och 15. Variablerna 1, 3, 5, 7 och 9 uteslöts precis som i datasetet, utöver dessa så utesluter vi även variabel 15, den heter CF total. En intressant sak att nämna är att variabel 15 representerar ett totalbelopp för alla Gtee kassafflöden, det vill säga variabel 2, 4, 6, 8 och 10. Vi kallar detta dataset för PK b). Då PK b) optimeras med hjälp av NNLS så ger det en vektor  $\mathbf{x}$  med 21881 element, denna vektor innehåller 21096 element som är lika med noll. Detta innebär att vi erhåller ett dataset med 785 modellpunkter. Antalet modellpunkter är alltså färre för detta dataset

jämfört med PK a). Dock så är residualkvadratsumman högre,  $3.072 \times 10^{-20}$ . Om vi tittar på residualplotten ser vi att de variabler med högre index, har ett högre totalbelopp på residualerna. Dessa variabler innehåller högre belopp och därför kan de ändå procentuellt sett vara noggrannare än vissa av variablerna med små residualer.



Figur 2: Residualplott för modelpunktsuppsättning b) för dataset PK. På y-axeln så har vi c som representerar residualens storlek och på x-axeln så har vi variabel index.

**Tabell 4.1** Resultat för PK

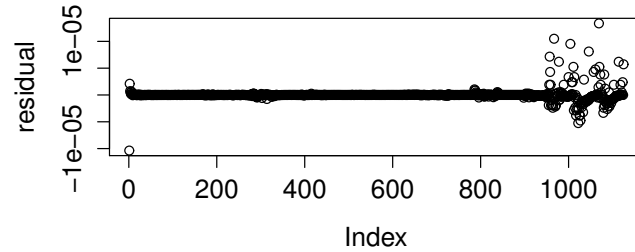
Följande tabell visar hur respektive variabel skiljer sig från referensdata. Siffrorna är angivna i procent %

| Variabel                           | Gamla MPT | Nya MPT a) | Nya MPT b) |
|------------------------------------|-----------|------------|------------|
| <i>CF Asset Management Fee FDB</i> | 130,8     | 109        | 100        |
| <i>CF Asset Magement Fee Gtee</i>  | 94,9      | 98         | 110        |
| <i>CF Benefit Death FDB</i>        | 80,6      | 86         | 100        |
| <i>CF Benefit Death Gtee</i>       | 87,3      | 100        | 100        |
| <i>CF Benefit Math FDB</i>         | 1690,5    | 170        | 146        |
| <i>CF Benefit Math Gtee</i>        | 102,4     | 100        | 100        |
| <i>CF Benefit Surr FDB</i>         | 811,7     | 81         | 107        |
| <i>CF Benefit Surr Gtee</i>        | 91,9      | 100        | 100        |
| <i>CF Charge Tax Red FDB</i>       | 479,1     | 99         | 108        |
| <i>CF Charge Tax Red Gtee</i>      | 96,0      | 98         | 100        |
| <i>CF Premium Total</i>            | 105,0     | 100        | 100        |
| <i>CF Terminal Bonus Total</i>     | 198,5     | 84         | 99         |
| <i>CF Expense and Comm</i>         | 100,6     | 100        | 100        |
| <i>Number of Policys</i>           | 102,3     | 100        | 100        |
| <i>PV CF Stoch</i>                 | 100,3     | 101        | 101        |
| <i>Reserve 2:nd order</i>          | 100       | 100        | 100        |
| <i>Reserve Total</i>               | 96,8      | 100        | 100        |

## 4.2 Tjänstepension

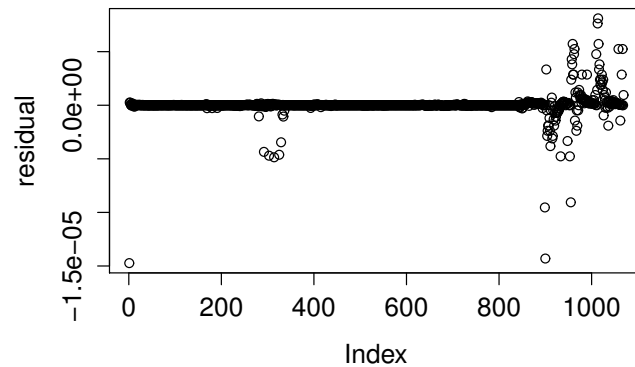
a) När försäkringsbeståndet T optimeras med hjälp av NNLS, så använder vi oss av samma förfarande som för datasetet PK. Det visar sig att det dataset som erhåller bäst noggrannhetskvoter är T med alla variabler inkluderade. Vi kallar detta dataset T a). När vi optimerar T a) med hjälp av NNLS så erhåller vi en vektor  $\boldsymbol{x}$  är av dimension  $n \times 1$  med  $n = 57949$ . I vektorn  $\boldsymbol{x}$  är 56907 element lika med noll, detta innebär att vi erhåller 1042 stycken modellpunkter i datasetet. Precis som för de dataset som erhållits av PK så plottar vi även här residualerna för T a). Även för detta dataset så ser vi att de variabler med ett högre index har större absolutbelopp på residualerna. Det följer samma mönster som PK att de variabler med högre belopp varierar mer. Residualkvadratsumman för denna optimering är  $7.903 \times 10^{-19}$ . Resultatet då vi testat modellpunkterna i AFM redovisas i 4.2.





Figur 3: Residualplott för modellpunktsuppsättning a) för dataset T. På y-axeln så har vi  $c$  som representerar residualens storlek och på x-axeln så har vi variabel index.

b) Det dataset som ger näst bäst noggrannhet på modellpunkterna för T är det dataset då vi endast utesluter variabel 15, CF total. Detta dataset kallar vi för T b). När vi optimerar T b) med NNLS så erhålls en vektor  $\boldsymbol{x}$  där endast 986 av de 57949 elementen är skilda från noll. Detta innebär att vi får 56 stycken modellpunkter färre än T a). Residualkvadratsumman för denna optimering är  $1.102 \times 10^{-16}$ , alltså högre än för T a). Om vi tittar på residualplotten nedan så ser vi återigen att det är de variabler med högre belopp som har en högre variation. Resultaten på noggrannhetskvoterna relativt referensdata redovisas i tabell 4.2.



Figur 4: Residualplott för modellpunktsuppsättning b) för dataset PK. På y-axeln så har vi  $c$  som representerar residualens storlek och på x-axeln så har vi variabel index.

**Tabell 4.2** Resultat för *T*

Följande tabell visar hur respektive variabel skiljer sig från referensdata.

Siffrorna är angivna i procent %

| <i>Variabel</i>                     | <i>Gamla MPT</i> | <i>Nya MPT a)</i> | <i>Nya MPT b)</i> |
|-------------------------------------|------------------|-------------------|-------------------|
| <i>CF Asset Management Fee FDB</i>  | 103,5            | 101               | 102               |
| <i>CF Asset Management Fee Gtee</i> | 100,8            | 100               | 100               |
| <i>CF Benefit Death FDB</i>         | 108,6            | 88                | 143               |
| <i>CF Benefit Death Gtee</i>        | 107,2            | 87                | 87                |
| <i>CF Benefit Math FDB</i>          | 116,6            | 102               | 99                |
| <i>CF Benefit Math Gtee</i>         | 99,6             | 101               | 100               |
| <i>CF Benefit Surr FDB</i>          | 104,4            | 101               | 100               |
| <i>CF Benefit Surr Gtee</i>         | 98,3             | 100               | 100               |
| <i>CF Charge Tax Red FDB</i>        | 116,6            | 101               | 102               |
| <i>CF Charge Tax Red Gtee</i>       | 100,7            | 100               | 100               |
| <i>CF Premium Total</i>             | 106,9            | 100               | 100               |
| <i>CF Terminal Bonus Total</i>      | 342              | 105               | 135               |
| <i>CF Expense and Comm</i>          | 99,8             | 100               | 100               |
| <i>Number of Policys</i>            | 100,5            | 100               | 100               |
| <i>PV CF Stoch</i>                  | 99,9             | 100               | 101               |
| <i>Reserve 2:nd order</i>           | 100              | 100               | 100               |
| <i>Reserve Total</i>                | 100,1            | 100               | 100               |

### 4.3 Stress av modellpunkter

I stressen vill vi att noggrannheten för våra modellpunkter ska ligga innanför de riktlinjer vi har. Vi stressar flera gånger och får fem resultat för varje stress. Resultaten blir liknande varandra och i tabellen nedan redovisas siffrorna för en utvald stress för respektive dataset.

**Tabell 4.3** *Stress av modellpunkter*

*Följande tabell visar hur respektive variabel skiljer sig från referensdata, då vi stress testat datasetet med modellpunkter. Siffrorna är angivna i procent %*

| <i>Variabel</i>                    | <i>Stess PK</i> | <i>Stress T</i> |
|------------------------------------|-----------------|-----------------|
| <i>CF Asset Management Fee FDB</i> | 97,3            | 101,7           |
| <i>CF Assetagement Fee Gtee</i>    | 103,2           | 100,3           |
| <i>CF Benefit Death FDB</i>        | 90,9            | 112             |
| <i>CF Benefit Death Gtee</i>       | 100             | 99,5            |
| <i>CF Benefit Math FDB</i>         | 128,1           | 132,7           |
| <i>CF Benefit Math Gtee</i>        | 100             | 99,5            |
| <i>CF Benefit Surr FDB</i>         | 91,2            | 115,4           |
| <i>CF Benefit Surr Gtee</i>        | 98,5            | 99,9            |
| <i>CF Charge Tax Red FDB</i>       | 87,3            | 120,9           |
| <i>CF Charge Tax Red Gtee</i>      | 103,1           | 100,4           |
| <i>CF Premium Total</i>            | 99,9            | 99,9            |
| <i>CF Terminal Bonus Total</i>     | 87,3            | 131,1           |
| <i>CF Expense and Comm</i>         | 100,3           | 99,8            |
| <i>Number of Policys</i>           | 100             | 99,5            |
| <i>PV CF Stoch</i>                 | 99,8            | 100,1           |
| <i>Reserve 2:nd order</i>          | 100             | 99,3            |
| <i>Reserve Total</i>               | 100             | 100,3           |

## 5 Slutsatser och Diskussion

I detta arbete utgår vi från två dataset med modellpunkter för försäkringar och använder oss av NNLS programmet i R för att svara på följande frågor, Är det möjligt att minska antalet modellpunkter med Icke negativ minsta kvadratoptimering?

Hur många modellpunkter erhåller vi med hjälp av NNLS?

Hur ser noggrannheten ut hos de nya modellpunkterna?

### 5.1 Är det möjligt att minska antalet modellpunkter med Icke negativ minstakvadrat optimering?

Jag vill påstå att det är möjligt att reducera antalet modellpunkter med NNLS, det resultat som erhöles med hjälp av NNLS gav bra resultat utefter de kriterier vi har satt upp. Modellpunkterna visade ett tillfredställande resultat då de kördes i AFM.

### 5.2 Hur många modellpunkter erhåller vi med hjälp av NNLS?

Vi erhöles färre än hälften av de modellpunkter som erhöles med hjälp av den gamla modellpunktsmetoden för både TK och P.

### 5.3 Hur ser noggrannheten ut hos de nya modellpunkterna?

Noggrannheten var bra på alla nyckelvariabler. En del variabler uppfyllde inte riktlinjerna, men resultatet godtogs ändå tack vare att vi erhöles en förbättring mot de gamla modellpunkterna. Våra riktlinjer var uppfyllda för alla nyckelvariabler vilket var ett krav. På de variabler med sämre resultat kan vi titta på totalbelopp för att få en hint om det stämmer. Då vi summerade alla variabler med FDB i namnet vi erhöles från modellpunkterna och jämförde med resultatet från summan av alla variabler med FDB kassaflöden vi erhöles från P-by-P stämde det till 100%. Detsamma gällde för Gtee. Vi erhöles även större residualer på variablerna med höga index, i samtliga fall motsvarade det variablerna, PV Cashflow FDB, PV Cashflow Stoch, Reserve 2:nd order samt Reserve total. Detta kan förklaras med att det är mycket större belopp i dessa variabler än i de övriga vilket innebär att en större residual här blir mindre procentuellt sett då vi jämför variabel för variabel.

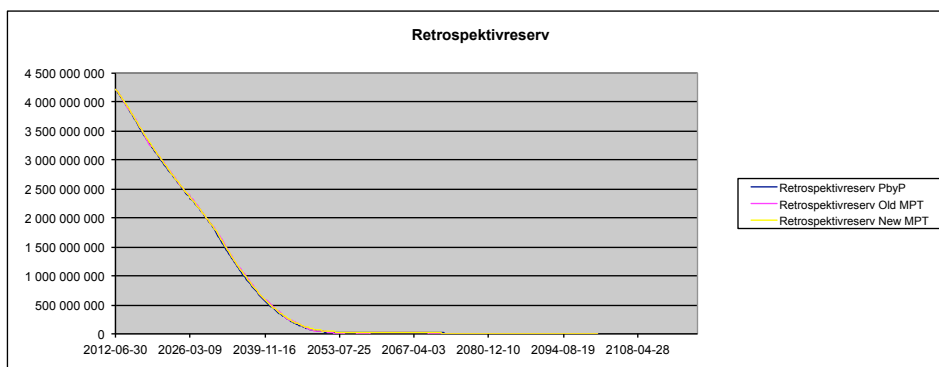
### 5.4 Diskussion

Jag vill påstå att de nya modellpunkterna är bättre än de gamla då vi fått bättre noggrannhet och färre modellpunkter. När NNLS optimering genomföres på dataset bör stor vikt läggas vid att använda ett så bra indata-

taset som möjligt. Jag har fått information från försäkringsbolaget att det försäkringsbeståndet PK innehåller vissa skillnader mellan försäkringarna, samt att försäkringsbeståndet i T är mer homogent. Med homogent menas att försäkringarna i beståndet har lika egenskaper. Ett homogent dataset underlättar NNLS optimeringen, för att det blir färre grupperingar, det vill säga fler vikter satta till noll. Fler vikter satta till noll ger färre modellpunkter. Eftersom summan av alla vikter är lika med summan av alla försäkringar innebär detta att vikterna blir större. I vår optimering kan vi se att det ger ett bättre resultat för T än för PK. Med bättre resultat menas ett noggrannare resultat samt färre modellpunkter i förhållande till antalet variabler vi utgår från. Antalet variabler vi tar hänsyn till då vi genomför optimeringen påverkar noggrannheten i resultatet, desto fler variabler vi tar hänsyn till desto noggrannare resultat. De variabler som innehåller störst belopp väger tyngre i optimeringen. Detta gynnar de variabler som har stora belopp och i vårt fall lämpar det sig bra eftersom det är den totala reserven som är det viktigaste. En följd av detta är att de variablerna med högre belopp är de som är mest intressanta för oss. De variabler som innehåller ett lägre belopp tillåts därför att variera mer procentuellt sätt. Vi testade även att minska antalet år som vi använt i optimeringen, det vill säga bara behålla en del av årtalen för varje variabel. Det gav en liten eller ingen minskning av modellpunkter då vi tog bort få årtal. Då flera årtal togs bort erhöles mycket sämre noggrannhetskvoter. Vi valde därför att inte redovisa dessa resultat. Noggrannheten i datasetet har ett negativt samband med antalet modellpunkter. Då antalet variabler som tas hänsyn till i optimeringen minskar då minskar antalet modellpunkter men i proportion till det ökar residualkvadratsumman. De dataset som kan vara att föredra i detta arbete är alltså PK a) och T a). De gav några fler modellpunkter än PK b) och T b) men vi erhåller istället lägre residualkvadratsummor och en bättre noggrannhet.

Vi använder oss av R till arbetet eftersom det är ett kostnadsfritt programmeringsspråk. På detta sätt har vi undvikit kostnader för programlicenser för NNLS optimeringen. R har fungerat över förväntan med tanke på de extremt stora dataset som vi har använt oss av.

## Retrospektivreserv



Figur 5: Totala reserven, här har vi plottat hela reserven då denna har beräknats i AFM med hela datasetet, de nya modellpunkterna och de gamla modellpunkterna.

## Referenser

- [1] ANDERSSON P., TYRCHA J.(2012) Notes in Econometrics , Compendium  
Stockholm University , Department of Mathematics.
- [2] BAPAT R. B., RAGHAVAN T. E. S. (1997) Nonnegative Matrices and  
Applications Cambridge University Press, UK.
- [3] C.L. LAWSON AND R.J. HANSON (1995) Solving Least-Squares Problems, SIAM.
- [4] DONGHUI CHEN AND ROBERT J. PLEMMONS Nonnegativity  
Constraints in Numerical Analysis, [http://www.wfu.edu/~plemmons/  
papers/nonneg.pdf](http://www.wfu.edu/~plemmons/papers/nonneg.pdf)
- [5] HÖGLUND T. (2008) Mathematical Asset Management, John Wiley &  
Sons
- [6] JOURNAL OF STATISTICAL SOFTWARE, (APRIL 2009), VOLUME 30  
An R Function for Sampling Linear Inverse Problems, [http://www.  
jstatsoft.org/v30/c01/paper](http://www.jstatsoft.org/v30/c01/paper)
- [7] MOLER C. (2004) Numerical Computing with Matlab, SIAM.
- [8] MOODY T. CHU Solving Least squares [http://www4.ncsu.edu/  
~mtchu/Teaching/Lectures/MA529/chapter4.pdf](http://www4.ncsu.edu/~mtchu/Teaching/Lectures/MA529/chapter4.pdf)
- [9] SOETAERT K., VAN DEN MEERSCHE K., VAN OELVEN N.R-project  
User's Guide for the `limSolve` Package,  
[http://cran.r-project.org/web/packages/limSolve/limSolve.  
pdf](http://cran.r-project.org/web/packages/limSolve/limSolve.pdf)
- [10] STERNBERG S. (2010) Dynamical Systems, Dover books.

## A NNLS algoritm

| Steg | Beskrivning   |
|------|---|
| 1    | Mängden $P$ är en nollmängd, $Z:=1,2,\dots,n$ , och $x:=0$  |
| 2    | Beräkna $n$ -vektorn $w$ som definieras som $E^t(f - Ex)$   |
| 3    | Om mängden $Z$ är tom eller om $w_j \leq 0$ för alla $j$ som existerar i $Z$ , gå till steg 12.   |
| 4    | Hitta ett index $t$ som existerar i $Z$ så att $w_t = \max(w_j : j \in Z)$ .  |
| 5    | Flytta indexet $t$ från mängden $Z$ till mängden $P$ .  |
| 6    | Låt $E_p$ vara $m \times n$ matrisen som är definierad av kolumn $j$ i $E_p := \begin{cases} \text{kolumn } j \text{ i } E & \text{om } j \in P \\ 0 & \text{om } j \in Z \end{cases}$ Beräkna $n$ -vektorn $z$ som en lösning till minsta kvadrat problemet $E_p Z = f$ .<br>Notera att de enda komponenterna $Z_j, j \in P$ , är lösta för detta problem.<br>Definiera $Z_j := 0$ för $j \in Z$ . |
| 7    | Om $Z_j > 0$ för alla $j \in P$ , sätt $x := z$ och gå till steg 2  |
| 8    | hitta ett index $q \in P$ sådant att $\frac{x_q}{x_q - z_q} = \frac{x_j}{x_j - z_j} \leq 0, j \in P$  |
| 9    | sätt $\alpha := \frac{x_q}{x_q - z_q}$  |
| 10   | sätt $x := x + \alpha(Z - x)$   |
| 11   | Flytta från mängden $P$ till mängden $Z$ alla index $j \in P$ för vilka $X_j = 0$ . Gå till steg 6  |
| 12   | Beräkningen är klar   |



## B Exempel Icke linjär minstakvadrat

Nedan följer ett exempel på hur ett icke linjärt minsta kvadratproblem kan lösas med hjälp av Gauss-Newtonmetoden. Alla definitioner från detta exempel finns i [7].

$$\text{Minimera } F(X) = \frac{1}{2} \sum_i^n f_i(x)^2 = \frac{1}{2} \|f(x)\|_2^2$$

där  $x \in \mathbb{R}^n$ ,  $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$  och  $f(x) = (f_1(x), \dots, f_m(x))^T$  är en jämn funktion av  $x$ . Ett nödvändigt villkor för att något  $x$ ,  $\bar{x}$  ska vara en kritisk punkt är att  $g(x)$ , som vi definierar som  $\nabla f(\bar{x})$  är lika med noll, att  $\nabla f(x)$  är lika med  $J(x)^T f(x)$  där vi definierar Jacobianen som

$$J(x) := \frac{\partial f}{\partial x} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

Notera att  $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ . Vi kan lösa ut när  $g(x)$  är lika med noll med hjälp av Newton Raphson metoden, vi låter  $x^{(k)}$  vara den nuvarande skattaren av  $\bar{x}$ , och beräknar ett Newton steg  $p_k$  genom att lösa ekvationen

$$g'(x^{(k)})p_k = -g(x^{(k)})$$

och sedan uppdatera  $x^k$  till

$$x^{(k+1)} = x^{(k)} + \tau_k p_k$$

där variabeln  $\tau_k$  väljs på ett sådant sätt att  $F(x)^{(k+1)}$  är strikt monotont ökande. Matrisen  $g'(x)$  är Hessian matrisen av  $F$  och beräknas

$$g'(x) = J_f(x)^T J_f(x) + \sum_i^n H_i f_i(x) \quad (9)$$

där  $H_i(x) := \frac{\partial^2 f_i}{\partial x_s \partial x_t}$  är Hessian matrisen av  $f_i(x)$ .

Vanligtvis om  $\|f_i\|$  går mot noll när  $x^{k+1}$  närmar sig lösningen kommer den andra matrisen i ekvation 9 också gå mot noll. Därför skattas newtonriktningen med lösningen till ekvationen

$$J(x^{(k)})^T J(x^{(k)}) P_k = -J(x^{(k)})^T f(x^{(k)}). \quad (10)$$

Lösningen till 10 är lösningen till minsta kvadratproblemet Minimera  $\frac{1}{2} \|J(X)^{(K)} P + F(x^k)\|_2^2$  och den lösningen är unik om  $J(x^{(k)})$  är av full rang.

Vi noterar också att Taylorutvecklingen

$$f(\bar{x}) \approx f(x) + J(\bar{x})(\bar{x} - x) + f(x)\|_2^2 \quad (11)$$

kommer att minimera  $\frac{1}{2} \|f(z)\|_2^2$  om  $x$  ligger nära  $\bar{x}$  vara approximativt lika med  $\frac{1}{2} \|J(x)(z - x) + f(x)\|_2^2$ .

## C Bevis av Perron-Frobenius sats

### Definitioner

Följande definitioner är hämtade från [10]

Låt  $\mathbf{P} := (\mathbf{I} + \mathbf{A})^k$ , och låt  $k$  vara så stor att  $\mathbf{P}$  är en positiv matris för  $v \leq w, v \neq w \rightarrow \mathbf{P}v < \mathbf{P}w$ .

Låt  $Q$  vara den positiva kvadranten, som definieras  $Q := \{x \in \mathbb{R}^n : x \geq 0, x \neq 0\}$ .

Låt  $C$  vara skärningen mellan enhetsfären och den positiva kvadranten, som definieras  $C := \{x \geq 0 : \|x\| = 1\}$  för något  $z \in Q$ .

Låt  $L(z) := \max\{s : \mathbf{S}z \leq \mathbf{A}z\} = \min \frac{(\mathbf{A}z)_i}{z_i}$

Om vi definierar  $L(rz) = L(z)$  för något  $r > 0$  där  $r$  är en konstant, då innebär detta att  $L(z)$  bara beror på matrisen  $\mathbf{A}$  genom  $z$ . Om  $z \leq y$  och  $z \neq y$  då följer det att  $\mathbf{P}z < \mathbf{P}y$ . Om  $sz \leq \mathbf{A}z$  där  $s\mathbf{P}z \leq \mathbf{P}\mathbf{A}z = \mathbf{A}\mathbf{P}z$  alltså  $L(\mathbf{P}z) \geq L(z)$ .

Om  $L(z)z \neq \mathbf{A}z$  då  $L(z)\mathbf{P}z < \mathbf{A}\mathbf{P}z$ .

Då är  $L(\mathbf{P}z) > L(z)$  om inte  $z$  är en egenvektor till  $\mathbf{A}$  med egenvärde  $L(z)$ .

Om vi utgår från de definitioner vi gjort ovan kan vi använda följande idé för beviset.

**Idé,** Vi tittar på en positiv vektor som maximerar  $L$ , samt visar att den egenvektorn uppfyller alla egenskaper i satsen.

Vi delar upp beviset i flera steg för att det ska vara lättare att följa.

### Steg 1, Hitta en positiv egenvektor

$P(C)$  är en kompakt mängd där alla element har strikt positiva komponenter. Funktionen  $L$  kontinuerlig i  $P(C)$  och därför når  $L$  sitt maximum,  $L_{max}$ , på mängden  $P(C)$ . Eftersom  $L(z) \leq L(\mathbf{P}z)$  är detta det största värdet av  $L$  på alla  $Q$ . Eftersom  $L(\mathbf{P}z) > L(z)$  om inte  $z$  är en egenvektor av  $\mathbf{A}$ , så kan vi dra slutsatsen att  $L_{max}$  nås vid någon egenvektor. Vi kallar den egenvektorn för  $x$  av  $\mathbf{A}$ ,  $x > 0$  med  $L_{max}$  som egenvärde. Eftersom  $\mathbf{A}x > 0$  och  $\mathbf{A}x = L_{max}x$  har vi att  $L_{max} > 0$ .

### Steg 2, Visa att det är det största egenvärdet

Visa att  $L_{max}$  är det största egenvärdet. Låt  $y$  vara en egenvektor med egenvärde  $\lambda$ , och låt  $|y|$  vara vektorn vars komponenter är  $|y_j|$  är absolutvärdena av komponenterna av  $y$ . Vi har  $|y| \in Q$ . Av  $\mathbf{A}y = \lambda y$  fås att  $\lambda y_i = \sum_j A_{ij}y_j$ . Eftersom att alla  $A_{ij} \geq 0$  fås att  $|\lambda||y_i| = \sum_j A_{ij}|y_j|$ . Detta

kan förkortas till  $|\lambda||y| = \mathbf{A}|y|$ . Enligt definitionen av  $L$ , säger detta att  $|\lambda| \leq L(|y|) \leq L_{max}$ . Vi kan alltså använda notationen  $\lambda_{max} := L_{max}$  eftersom att vi har visat att  $|\lambda| \leq L_{max}$ . Med detta så är punkt 1) i satsen visad. (Värt att notera är att vi inte kan skriva  $\lambda_{max}=0$ , eftersom att då skulle alla egenvärden till  $\mathbf{A}$  vara lika med noll, och därför vara oanvändbara.

$\mathbf{A}$  är irreducibel vilket innebär att  $\lambda_{max} \geq 0$ ).

**Steg 3** visa att  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}, \mathbf{S} \neq \mathbf{A} \rightarrow \lambda_{max}(\mathbf{S}) \leq \lambda_{max}(\mathbf{A})$ .

Anta att  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}$ . Om  $\mathbf{z} \in Q$  är en vektor som uppfyller att  $\mathbf{sz} \leq \mathbf{S}\mathbf{z}$  då  $\mathbf{S}\mathbf{z} \leq \mathbf{A}\mathbf{z}$  betyder det att  $\mathbf{sz} \leq \mathbf{A}\mathbf{z}$ . Detta i sin tur ger att  $L_S(\mathbf{z}) \leq L_T(\mathbf{z})$  för alla  $\mathbf{z}$  och därför blir  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}, \rightarrow \lambda_{max}(\mathbf{S}) \leq \lambda_{max}(\mathbf{A})$ . Som följd av alla dessa olikheter ges att  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}, \mathbf{S} \neq \mathbf{A} \rightarrow \lambda_{max}(\mathbf{S}) \leq \lambda_{max}(\mathbf{A})$ .

**Steg 4, Visa att  $\lambda_{max}(\mathbf{A}^T) = \lambda_{max}(\mathbf{A})$ .**

Om vi använder de resultaten vi har för  $\mathbf{A}$  på  $\mathbf{A}^T$  så kan vi se att  $\mathbf{A}^T\mathbf{f}$  också har ett positivt största egenvärde, vi kan kalla detta egenvärde för  $\theta$ . (Därefter så ska vi visa att  $\theta = \lambda_{max}$ ).

Det finns en radvektor  $\mathbf{w} > 0$  sådan att  $\mathbf{w}^T\mathbf{A} = \theta\mathbf{w}^T$ . Då gäller att  $\mathbf{w}^T\mathbf{A}\mathbf{x} = \theta\mathbf{w}^T\mathbf{x} = \lambda_{max}\mathbf{w}^T\mathbf{x}$ , vilket innebär att  $\theta = \lambda_{max}$  eftersom  $\mathbf{w}^T\mathbf{x} > 0$ .

**Steg 5. Visa de första två påståendena i del 4) av satsen.**

Antag att  $\mathbf{y} \in Q$  och  $\mathbf{A}\mathbf{y} \leq \mu\mathbf{y}$ . Då  $\lambda_{max}\mathbf{w}^T\mathbf{y} = \mathbf{w}^T\mathbf{y} \leq \mu\mathbf{w}^T\mathbf{y}$  Det leder till att  $\lambda_{max} \leq \mu$ , här använder vi en gång att alla komponenterna av  $\mathbf{w}$  är positiva och någon komponent av  $\mathbf{y}$  är positiv  $\mathbf{w}^T\mathbf{y} > 0$ . Speciellt om  $\mathbf{A}\mathbf{y} = \mu\mathbf{y}$  då är  $\mu = \lambda_{max}$ . Vidare om  $\mathbf{y} \in Q$  och  $\mathbf{A}\mathbf{y} \leq \mu\mathbf{y}$  då  $\mu \leq 0$  och  $0 < \mathbf{P}\mathbf{y} = (1 + T)^{n-1}\mathbf{y} \leq (1 + \mu)^{n-1}\mathbf{y}$  vilket ger att  $\mathbf{y} > 0$  Vi har nu bevisat de första två påståendena.

Om  $\mu = \lambda_{max}$  då är  $\mathbf{w}^T\mathbf{A}\mathbf{y} - \lambda_{max}\mathbf{y} = 0$  men  $\mathbf{A}\mathbf{y} - \lambda_{max}\mathbf{y} \leq 0$  och därför är  $\mathbf{x}^T(\mathbf{w}^T(\mathbf{A}\mathbf{y} - \lambda_{max}\mathbf{y})) = 0$  vilket ger att  $\mathbf{A}\mathbf{y} = \lambda_{max}\mathbf{y}$ . Det sista påståendet av del 4 (att  $\mathbf{y}$  är en skalärmultipel av  $\mathbf{x}$ ) följer från Steg 2 eftersom att vi har visat att  $\mathbf{y}$  är en egenvektor med egenvärde  $\lambda_{max}$ .

**Steg 6 Visa att om  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}, \mathbf{S} \neq \mathbf{A}$**  Då kommer varje egenvärde  $\sigma$  till  $\mathbf{S}$  satisfiera  $|\sigma| < \lambda_{max}$ . Antag att  $\mathbf{0} \leq \mathbf{S} \leq \mathbf{A}$  och  $\mathbf{S}\mathbf{z} = \sigma\mathbf{z}, \mathbf{z} \neq 0$ . Då  $\mathbf{A}|\mathbf{z}| \geq \mathbf{S}|\mathbf{z}| \geq |\sigma||\mathbf{z}|$  kommer  $|\sigma| \leq L_{max}(\mathbf{A}) = \lambda_{max}$ , Men om  $|\sigma| = \lambda_{max}(\mathbf{A})$  då  $L_A(|\mathbf{z}|) = L_{max}(\mathbf{A})$  kommer  $|\mathbf{z}| > 0$  och  $|\mathbf{z}|$  är också en egenvektor av  $\mathbf{A}$  med samma egenvärde. Men då är  $(\mathbf{A} - \mathbf{S})|\mathbf{z}| = 0$  och detta är omöjligt om inte  $\mathbf{S} = \mathbf{A}$  eftersom  $|\mathbf{z}| > 0$ . Genom att byta ut den  $i$ :te raden och kolumnen av  $\mathbf{A}$  mot nollor erhåller vi  $\mathbf{S} \geq \mathbf{0}$  där  $\mathbf{S} < \mathbf{A}$  eftersom att  $\mathbf{A}$  är irreducibel utgör det ett hinder för att alla elementen i en rad är det. Detta bevisar påståendet att egenvärdena av  $T_i$  alla är mindre eller lika med absolutbeloppet av  $\lambda_{max}$ .

**Lemma 1, från linjär algebra.**

Låt  $\mathbf{A}$  vara en (kvadratisk) matris och låt  $\lambda$  vara en diagonal matris av samma storlek med elementen  $\lambda_1, \dots, \lambda_n$  längs diagonalen. Utvidga  $\det(\lambda - \mathbf{A})$  längs den  $i$ -te raden visar att  $\frac{\partial}{\partial \lambda_i} \det(\lambda - \mathbf{A}) = \det(\lambda_i - A_i)$  där index  $i$  betyder matrisen som erhålls genom att eliminera den  $i$ :te raden och den  $i$ :te kolumnen från matrisen.

Genom att sätta  $\lambda_i = \lambda$  och applicera kedjeregeln från matematisk analys får vi  $\frac{\delta}{\delta\lambda} \det(\lambda \mathbf{I} - \mathbf{A}) = \sum_i \det(\lambda \mathbf{I} - T_i)$ . Därefter ska vi visa att  $\lambda_{max}$  har algebraisk och geometrisk multiplicitet 1. Varje matris  $\lambda_{max} \mathbf{I} - T_i$  har strikt positiv determinant vilket ges av de tidigare bevisen. Detta visar att derivatan av det karakteristiska polynomet av  $\mathbf{A}$  inte är noll vid  $\lambda_{max}$ , och därför är den algebraiska multipliciteten och följaktligen den geometriska multipliciteten av  $\lambda_{max}$  lika med ett. Detta bevisar 2) och därför är alla utom det sista påståendet i satsen visat. Det sista påståendet säger att om  $\mathbf{A}$  är primitiv då kommer alla egenvärden av  $\mathbf{T}$  satisfiera  $|\lambda| < \lambda_{max}$ .

### Steg 7, visa punkt 7 i satsen

Egenvärdena till  $\mathbf{A}^k$  är de samma som den  $k$ :te potensen av egenvärdena till  $\mathbf{A}$ . Vi vill visa att det inte finns några andra egenvärden av en primitiv matris vars absolutbelopp är lika med  $\lambda_{max}$  påståendet ovan är dock inte tillräckligt för att bevisa detta för en positiv matris men att dela den positiva matrisen med  $\lambda_{max}$  reduceras problemet till att bevisa lemma 2 nedan.

**Lemma 2** Låt  $T > 0$  vara en positiv matris med  $\lambda_{max} = 1$ . Då kommer alla andra egenvärden till  $\mathbf{T}$  uppfylla villkoret  $|\lambda| < 1$ .

### Bevis av lemma 2.

Antag att  $\mathbf{z}$  är en egenvektor till  $\mathbf{T}$  med egenvärde  $\lambda$  där  $|\lambda| = 1$ . Där  $|\mathbf{z}| = |\lambda \mathbf{z}| = |\mathbf{T} \mathbf{z}| \leq |\mathbf{T}| |\mathbf{z}| = \mathbf{T} |\mathbf{z}| \rightarrow |\mathbf{z}| \leq \mathbf{T} |\mathbf{z}|$ . Låt  $\mathbf{y} := \mathbf{T} |\mathbf{z}| - |\mathbf{z}|$  och  $\mathbf{y} \geq 0$ . Om man även antar att  $\mathbf{y} \neq 0$ . Då  $\mathbf{T} \mathbf{y} > 0$  och  $\mathbf{T} |\mathbf{z}| > 0$  existerar det ett  $\epsilon > 0$  sådant att  $\mathbf{T} \mathbf{y} > \epsilon \mathbf{T} |\mathbf{z}|$  och därför är  $\mathbf{T} (\mathbf{T} |\mathbf{z}| - |\mathbf{z}|) > \epsilon \mathbf{T} |\mathbf{z}|$  eller  $B (\mathbf{T} |\mathbf{z}|) > \mathbf{T} |\mathbf{z}|$ , där  $B := \left( \frac{1}{1+\epsilon} \right) \mathbf{A}$ . Detta ger att  $B^k \mathbf{T} |\mathbf{z}| > \mathbf{T} |\mathbf{z}|$  för alla  $k$ . Men alla egenvärden av  $B$  är  $< 1$  i absolutbelopp, vilket ger  $B^k \rightarrow 0$ . Eftersom alla element av  $\mathbf{T} |\mathbf{z}|$  är  $\leq 0$  i motsats till det faktum att  $\mathbf{T} |\mathbf{z}| > 0$ . Alltså  $|\mathbf{z}|$  är en egenvektor av  $\mathbf{T}$  med egenvärde 1. Men  $|\mathbf{T} \mathbf{z}| = |\mathbf{z}|$  vilket ger att  $|\mathbf{T} \mathbf{z}| = \mathbf{T} |\mathbf{z}|$  vilket bara kan inträffa om alla element av  $\mathbf{z}$  är av samma tecken. Därför måste  $\mathbf{z}$  vara en multipel av våran egenvektor  $\mathbf{x}$  och eftersom det inte finns några andra egenvektorer med alla element av samma tecken annat än multiplar av  $\mathbf{x}$  så är  $\lambda=1$ .