PLS-VS: Partial Least Squares med variabelseleksjon

En eksplorativ regresjonsmetode med fokus mot bedre tolkbarhet

PLS-VS: Partial Least Squares with variable selection

An explorativ regression method with focus on better interpretability

Henrik Kjønnerød

Universitetet for miljø- og biovitenskap
Institutt for matematiske realfag og teknologi
Masteroppgave 30 stp. 2012

Sammendrag

I denne oppgaven har vi utviklet en PLS beslektet metode som for multivariate regresjonsproblemer kan brukes til å bygge konkurransedyktige modeller med tanke på prediksjon og tolkbarhet. Metoden er motivert av Powered Partial Least Squares Regression (PPLS) og variabelseleksjonsmetodikk kombinert med bruk av enkel teori om rette linjer. Med dette som grunnlag har vi lykkes med å utvikle en eksplorativ PLS metode kalt PLS-VS (PLS med variabelseleksjon) som i interessante tilfeller kan gi opphav til enkle og tolkbare modeller.

Fra moderne måleteknologier og datainnsamlingsmuligheter eksponeres vi stadig hyppigere for datasett med svært mange og høyt korrelerte forklaringsvariable. Her utfordres vi til å frambringe innsikt og ikke minst oversikt over årsaksforhold som gjerne kokes ned til hvilke forklaringsvariable som er de «viktigste» for å kunne forklare variasjonen i responsen(e) som studeres. Det er derfor et åpenbart «marked» for metoder som PLS-VS som både enkelt og forholdsvis effektivt er i stand til å produsere enkle modeller også i situasjoner der datagrunnlaget i utgangspunktet har høy kompleksitet.

Gjennom arbeidet med PLS-VS har vi sammenlignet vår metode både med noen beslektede PLS-metoder og andre etablerte metoder som anses som relativt godt egnet for analyse av komplekse datasett. Vi har også forsøkt å skaffe innsikt i likheter og forskjeller mellom modellene de ulike studerte metodene produserer på en samling reelle datasett.

PLS-VS viser seg å ha prediksjonsevne fullt på høyde med de kjente metodene vi har sammenlignet den med. Det er derfor grunnlag for å påstå at PLS-VS representerer et ikke uinteressant bidrag til utviklingen av regresjonsmetodikk tilpasset komplekse multivariate datasett. Ikke minst innenfor fagområder som kjemometri, spektroskopi, bioinformatikk og genomikk vil PLS-VS være et relevant verktøy.

Abstract

In this thesis we have developed a PLS-related method that produces competitive models in terms of prediction and interpretability when applied on multivariat regression problems. Yhe method is motivated by Powered Partial Least Squares Regression (PPLS) and variable selection combined with the usage of simple theory of straight lines. On this basis we have succeeded in developing an exploratory PLS-method named PLS-VS (PLS with variable selection) that in interesting cases can give rise to simple end interpretable models.

From modern measurement technologies and data mining abilities we are more frequently exposed to datasets including many and highly correlated explanatory variables. Here we are challenged to produce insight and not least an overview of the causality, that often boils down to which explanatory variables that are the most important to explain the variation in the response variable that are studied. Though, there is clearly a need for methods like PLS-VS that easily and rather efficiently are able to produce sparse models, also in situastions where the data provided has high complexity.

Throughout the study of PLS-VS we have compared our method with familiar PLS-methodology, and other established methods that are considered relatively suitable for analysis of complex dataset. We have also tried to provide insight in similarities and differences between the models the different methods produces on a collection of real datasets.

PLS-VS turns out to possess a predictionability at the same level as the known methods we have compared it with. Therefore there is on a reasonable basis we claim that PLS-VS doesn't represents an uninterested contribution to the development of regression methodology adapted to multivariat datasets. Not least will PLS-VS be a relevant tool within the fields of chemometrics, spectroscopy, bio-informatics and genomics.

Forord

Denne mastergradsoppgaven i anvendt matematikk er det avsluttende arbeidet i mitt 6-årige masterstudie innen retningen "Lektorutdanning i realfag" ved Institutt for matematiske realfag og teknologi (IMT) ved Universitetet for miljø og biovitenskap (UMB)

Arbeidet med oppgaven har vært utfordrende og lærerikt dypdykk inn i statistikkfaget. Det å kunne fordype seg såpass innen ett spesifikt tema har vært en personlig tilfredsstillelse og jeg gleder meg til å følge den videre utviklingen innenfor fagfeltet. Jeg har selvsagt også fått et større innblikk i hvordan forskningsarbeid foregår.

Denne oppgaven hadde ikke vært mulig å få til på egen hånd, og selvom det er et stykke fram fra en masteroppgave til de virkelig store vitenskapsmenns erkjennelser tillater jeg meg å sitere Isaac Newton som en gang skal ha sagt:

> *"Dersom jeg har sett litt lengre enn andre,*
> *så har jeg gjort det ved å stå på kjempers skuldre"*
> *Isaac Newton*

Gjennom studieløpet har jeg vært omgitt av små kjemper i ulike sammenhenger, som har hjulpet meg å «se litt lenger», og jeg vil takke noen av de her:
Jeg vil rette en stor takk til min veileder, Ulf Indahl, som åpnet opp døra for statistikkfaget, og som har motivert og veiledet meg til å skrive en mastergradsoppgave jeg er stolt av. Tusen takk for at døra alltid står på gløtt, slik at jeg kan komme med «ett lite spørsmål eller to». I tillegg vil jeg takke Kristian Liland som har hjulpet meg med å finne både referanselitteratur og passende datasett til benyttelse i denne oppgaven.
Gjennom selve mastergradsarbeidet har jeg vært så heldig å hatt trivelige medstudenter rundt meg som er med på faglige diskusjoner og som gladelig deler av egen erfaring. Jeg vil også takke medkorister i «Sangkoret Noe ganske Annet», som har gitt meg gode minner og opplevelser gjennom hele studietiden.
Tilslutt vil jeg takke venner og familie som alltid stiller opp og backer meg i valgene jeg gjør, og interesserer seg for det jeg driver med. Og sist men ikke minst en stor takk til min kjære samboer, Helene Kittilsen, som har holdt ut med meg når jeg har kommet hjem fra masterjobbinga med varierende humør og alltid er der når jeg trenger det.

Innholdsfortegnelse

Innledning

Regresjon

Hovedpoenget med regresjonjonsanalyse er å analysere sammenhengen mellom flere ulike variable og lage modeller som best mulig beskriver disse sammenhengene kvalitativt. Ofte er vi interessert i å sitte igjen med en modell som beskriver variasjonen i en variabel ved hjelp av variasjonen i en eller flere ulike forklaringsvariable. Regresjon brukes blant annet i kjemi, biologi, økonomi, fysikk, men også av hver og en av oss i dagliglivet. Dette kan for eksempel være tilsynelatende enkle ting som å beregne hvor lang tid en rekke gjøremål tar, eller mengden spiker som trengs for å lage et gjerde. Andre mer sammensatte problemer som kan modelleres kan for eksempel være å beregne massen til en bjørn ut fra lengdemål av ulike kroppsdeler, eller fettinnholdet i en fisk basert på spektroskopiske målinger.

OLS og problemer med denne metoden

Mange regresjonsmetoder tar utgangspunkt i at man har et treningssett som lærer opp en modell, og modellen kan beskrive sammenhengen mellom responsen og forklaringsvariablene eller vi kan predikere responsen til nye observasjoner. En av de enkleste og kanskje mest kjente metodene som gjør dette er minste kvadraters metode. Den produserer modeller som forsøker å best mulig beskriver variasjon i en responsvariabel, ved hjelp av en eller flere forklaringsvariable. Imidlertid støter metoden på vanskeligheter når forklaringsvariablene som skal beskrive responsen er høyt korrelerte eller vi har relativt mange forklaringsvariable sammenlignet med antall observasjoner. Resultatet blir da ofte at OLS produserer ustabile modeller med svært dårlig prediksjon av responsen og dårlig tolkbarhet.

Regresjonsmetoder

For å unngå å få ustabile modeller i de tilfellene OLS får vanskeligheter, har man utviklet alternative regresjonsmetoder med ulike kriterier som forsøker å takle disse problemene. I denne oppgaven har vi valgt å rette fokus mot tre typer av disse. Den første er variabelseleksjonsmetoder som plukker ut et mindre antall «viktige» forklaringsvariable som det gjøres regresjon på. Den andre er krympingsmetoder som minker betydningen av potensielt uviktige forklaringsvariable i modellene som produseres og stabiliserer modellene. Den tredje og siste metode er projeksjonsmetoder som tar utgangspunkt i å lage nye

forklaringsvariable basert på kombinasjoner av de man har, for så å gjøre regresjon på disse. Vi skal videre ta ut et knippe metoder og se på sammenhenger og ulikheter i modellene de produserer i forhold til tolkbarhet og prediksjonsegenskaper.

Validering

For å kunne si noe om prediksjonsevnen til en modell, må den valideres. Ved å kun la resultatene fra en enkelt kjøring på treningssettet danne grunnlaget for hvordan vi skal lage modeller, har vi ingen kontroll på modellens stabilitet og evnen til å predikere på nye datasett. Enkle måter å validere på er for eksempel kryssvalidering og bruk av uavhengige testsett. I kryssvalidering modelleres det gjentatte ganger og et sett med observasjoner blir tatt ut av treningssettet i hver runde. Modellene som lages i hver runde brukes til å predikere på de utelatte observasjonene. Når samtlige observasjoner har blitt brukt til å predikere på gir det samlede resultatet et mål på hvordan modellen kan predikere på nye data, og i tillegg gir det et inntrykk av stabiliteten til modellen. Validering ved hjelp av testsett går ut på å bruke treningssettet til modellering, og vurderer prediksjonsevnen til modellen når den benyttes på testsettet. Fordelen med bruk av testsett er at modellene testes på observasjoner som ikke har vært med å bygge modellen.

Validering kan også brukes som et hjelpemiddel i modellutvelgelsen, for å få ett inntrykk av hva som er optimal modellkompleksitet eller hvordan ulike valg av parametere påvirker stabilitet og prediksjonsevne til modellen.

Problemstilling

For regresjonsproblemer med mange forklaringsvariable er det i mange situasjoner interessant å fjerne ikke informative variabler. I denne oppgaven har målet vært å utvikle en metode motivert av powered partial least squares (PPLS) som effektivt gir forenklede modeller samtidig som den skulle være like god til prediksjon som tradisjonelle multivariate metoder (PLS, PPLS, PCR, ridge regresjon).

Notasjoner

I oppgaven er det forsøkt å holde en konsekvent stil på notasjon som leseren raskt vil kunne kjenne igjen. Vi forsøker å følge engelsk/amerikansk skrivemåte , som skilletegn mellom tall og . som skille for desimaltall. For å beskrive et punkt i to dimensjoner vil dette skrives på

følgende måte: (1.2, 3,4). Følgende kriterier er forsøkt fulgt gjennom hele dokumentet:

- Konstanter angis med små bokstaver: a
- Vektorer angis med små fete bokstaver: $\boldsymbol{a}$
- Matriser angis med store bokstaver: M
- Den transponerte av en vektor eller matrise angis med apostrof: $\boldsymbol{a}'$
- Dimensjonene til en vektor eller matrise angis som konstanter under bokstaven med x mellom: $\underset{n\,x\,p}{M}$
- Gangetegn brukes der det ikke fremgår av sammenhengen at to bokstaver skal multipliseres sammen eller at dette vil fremheves spesielt: $M \cdot \boldsymbol{a}$
- Deling angis med brøkstrek eller skråstrek: $\boldsymbol{a}/\mathrm{b}$, $\frac{\boldsymbol{a}}{\mathrm{b}}$
- Invers av en matrise angis med $^{-1}$ til høyre for bokstaven: M^{-1}
- Kolonne og radnummer angis med kommadelt indeks: $M(2,:)$ (alle elementene i andre rad)
- Datamatriser angis som matrisa X
- Responsvariabelen angis som vektoren $\boldsymbol{y}$

Programvare

Oppgaven er skrevet i Open Office Org. Writer 3.3.0.

Beregninger og dataplott er utført i MathWorks MATLAB R2011a Student Version.

Referansehåndtering er utført med Endnote X5.0.1.

1 Regresjon

1.1 Innledning

Det å observere fenomener og å samle inn informasjon har menneskene mest sannsynlig gjort på en eller annen måte helt siden tidenes morgen. Det å studere og erfare egenskaper ved ting, for så å bruke denne erfaringa til videre utvikling har vært en livsviktig nødvendighet til alle tider. Innsamling av data som er generert til erfaringer har selvsagt blitt brukt til alt fra å finne ut hva som fungerer når man skal finne og bygge en boplass, til det å effektivt kunne jakte ned et dyr.

Denne måten å se på naturen og lære av den ble i nyere tid noe mer strukturert og konkretisert spesielt innenfor medisinfaget. En artig historie som illustrerer dette dreier seg om årelating som i tidligere tider var en anerkjent behandlingsmåte av svært mange ulike sykdommer. Behandlingen gikk ut på å snitte opp blodårene med en kniv og deretter tappe blod ut av kroppen til pasienten. Tidligere trodde man at årsaken til sykdommen befant seg i blodet, og nettopp derfor antok man at å tappe ut blod ville ha en helbredende virkning. I 1849 tok den tyske legen Joseph Dietl (Ill-Vit-Redaksjon 2004; Zajaczkowski 2010) til motmæle mot tradisjonen med årelating og mente at dette ikke var noen god behandlingsmåte. Han mente til og med at årelating faktisk gjorde mer skade enn det gjorde folk friske.

Årsaken til at den tyske legen kunne være så sikker i sin sak var at han utnyttet tallenes tale ved opptellinger og enkle statistiske betraktninger.

Dietl hadde over lengre tid foretatt datainnsamling av pasienter som var blitt behandlet både med og uten årelating. Ved å studere dette tallmateriale nærmere kom han fram til et slående resultat: Andelen pasienter som døde etter behandling med årelating var betydelig større enn andelen pasienter som døde ved alternativ form for behandling. Dermed kunne han med stor sikkerhet slå fast at årelating var en dårligere behandlingsmåte enn alternativ behandling. Han brukte altså tallenes tale til å sammenligne de to behandlingsmetodene. Resultatene hans ble heldigvis etterhvert anerkjent av leger rundt om, men det tok allikevel lang tid før man helt sluttet med årelating som behandlingsform. Mange leger som hadde utført årelating i en årrekke brukte ganske lang tid på å omstille seg, til tross for at de var blitt presentert et statistisk bevis for at annen behandlingsmåte fungerte bedre. På den annen side hadde arbeidet til den tyske legen medført et viktig steg på et annet fagfelt, nemlig anvendt statistikk.

Det Dietl gjorde i 1849 var en enkel tellbar tilnærming til sammenhengen mellom behandlingsmåte og resultat. Ved å studere hvordan resultatet av én behandlingsmåte skilte seg fra resultatene fra en annen behandlingsmåte, ble det ganske klart at nettopp behandlingsmåten tydelig bidro til å forklare forskjellene i overlevelsesrate.

Når man møter andre typer problemer og ønsker å se på hva som er årsaken til noe, kan man komme borti at det er flere ulike faktorer som påvirker resultatet.

Studerer vi hvilke faktorer som er viktige for at langrennsløpere i mosjonistklassen skal gjøre det godt i birkebeinerrennet (Birkebeiner-A/S 2012) kommer man fram til at det er flere faktorer og samspillet mellom ulike faktorer (Trane 2012) som har betydning. Når vi går fra å se på enkeltfaktorers betydning, til å vurdere hvordan flere faktorer og samspillet mellom disse har en betydning for resultatet beveger vi oss over i det som innen statistikk og dataanalyse kalles gjerne multivariat analyse. Her betyr multivariat at det er flere varierende faktorer som inkluderes i analysen. I eksempelet med langrennsløperne kunne vi kanskje utlede at av i alt 20 faktorer var det mellom 5 og 8 sentrale faktorer som hadde viktigst påvirkning. I forbindelse med studier av andre problemstillinger vil det være tilfeller der antallet variabler vi har å gå utifra er mye større.

Som en konsekvens av den teknologiske utviklingen i det moderne samfunnet har vi innen mange fagfelt bedre og mer nøyaktig utstyr og dermed enda bedre forutsetninger for datainnsamling og dataanalyse enn tidligere. For eksempel innen biologien og medisinfaget har genteknologi og forskning på genomet hatt en tilnærmet eksponentiell vekst i nyere tid. Et eksempel på en slik teknologi er DNA mikromatriser og den måler mengden av DNA fragmenter i en celle. På én enkelt mikromatrisechip er det kapasitet til å måle en stor andel av genene til et menneske som er i størrelsesorden ~ 20 000. Hvis vi ønsker å undersøke om uttrykket i genene indikere årsaken til enkelte sykdommer, kan vi undersøke pasienter som har sykdommen, og sammenligne disse med en frisk ved å se etter forskjeller i genetisk uttrykk mellom disse to gruppene. I dette problemer har vi som oftest et relativt lite antall observasjoner (syke og friske pasienter) i forhold til antall variable (gener) vi studerer (~ 20 000).

Med de enormt store datamengdene vi relativt enkelt og effektivt har til rådighet, skulle man jo tro at det kunne sette oss i stand til å forstå og forutse det meste. I praksis behøves dataverktøy og gode nok metoder til effektivt å analysere dataene som samles inn. Metodikken for denne typen dataanalyser omtales gjerne som *multivariat dataanalyse*, og i denne oppgaven vil vi spesielt fokusere på et utvalg multivariate *regresjonsteknikker.*

1.1.1 Multivariate datasett, regresjon og lineære modeller

Når man analyserer multivariate datasett er man gjerne interessert i å se på sammenhengen mellom et sett med forklaringsvariable og hvordan de påvirker en eller flere responsvariable. Dette kalles gjerne for et multivariat regresjonsproblem der man ønsker å tilpasse en modell på en slik måte at den får forklaringsvariablene til best mulig å beskrive de responsvariablene vi har målt. Hva slags type regresjonsmetode man velger å tilpasse er selvsagt opp til den som skal gjøre analysen å bestemme. En av de enkleste typen modeller får vi ved å anta at forklaringsvariablene forholder seg til responsvariabelen på en lineær måte, og med det tilpasse lineære modeller. Det er selvsagt heller ingenting i veien med å tilpasse modeller som for eksempel antar at interaksjon mellom ulike forklaringsvariable eller polynomer av forklaringsvariablene best beskriver responsvariabelen.

Så hvorfor velger vi ofte å anta en lineær relasjon mellom forklaringsvariablene og responsen fremfor mer eksotiske relasjoner? En fordel med lineære modeller er at de antar en enklest mulig struktur mellom variablene og responsen og at de gir oss en oversiktlig fremstilling av forholdene mellom disse. De blir dermed relativt enkelt å tolke hva en modell sier om forholdene mellom variablene vi studerer. Det viser seg at også i mange tilfeller fungerer denne typen modeller vel så bra som andre mer komplekse modeller når vi er interessert i prediksjon og fortolkninger (Hastie et al. 2009). Kombinasjonen av å ha en enkel og tolkbar modell med prediksjonsevne på høyde med andre modeller gjør at lineære modeller ofte er et godt alternativ når man skal analysere data.

Dersom vi har et datasett, X, med $p+1$ forklaringsvariable (inklusive en konstant innledende søyle med 1-ere) målt for n objekter sammen men en enkel tilhørende responsvektor, $\boldsymbol{y}$, målt for de samme n objektene, kan er multivariat lineær regresjonsmodell beskrives matematisk som:

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\epsilon} = \mathbf{1}\beta_0 + \boldsymbol{x}_1\beta_1 + ... + \boldsymbol{x}_p\beta_p + \boldsymbol{\epsilon} \quad \text{(formel 2-1)}$$

der $\boldsymbol{\beta}$ er en (p+1) x 1 vektor med regresjonskoeffisienter for skjæringspunktet, β_0, og stigningstallet for hver variabel og der $\boldsymbol{\epsilon}$ representerer en n x 1 vektor med tilfeldige feilavvik fra den virkelige responsen. Regresjonskoeffisientene i modellen indikerer hvor stor innvirkning hver variabel har på responsen. Når man har estimert passende regresjonskoeffisienter, $\hat{\boldsymbol{\beta}}$, for en modell har man at prediksjonen av responsen til denne modellen er:

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} \quad \text{(formel 2-2)}$$

Vi skal nå se nærmere på den mest kjente metoden, nemlig vanlig minste kvadraters metode.

1.2 OLS

OLS står for Ordinary Least Squares, på norsk minste kvadraters metode (Hastie et al. 2009), som er den mest kjente blant de lineære regresjonsmetodene. Metoden har en svært intuitiv geometrisk løsning, og den kan i mange tilfeller gi oss gode modeller. Metoden tar utgangspunkt i å estimerer de ukjente parameterne, $\boldsymbol{\beta}$, i en lineær regresjonsmodell slik at summen av kvadratavviket mellom responsen og den predikerte responsen minimeres. Dette kan også sees på som å minimere kvadratet av den euklidiske normen til residualene. Rent matematisk går minste kvadraters metode ut på å minimere uttrykket:

$$RSS(\boldsymbol{\beta}) = \sum_{i=1}^{n} (r_i)^2 = \sum_{i=1}^{n} (y_i - X_{(i,:)}\boldsymbol{\beta})^2 = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 \quad \text{(formel 2-3)}$$

med hensyn på $\boldsymbol{\beta}$. RSS er en forkortelse for Residual Sums of Squares, som på norsk oversettes med summen av de kvadrerte residualene.

Regresjonskoeffisientene som minimerer uttrykket har en unik løsning som kan beregnes analytisk ved:

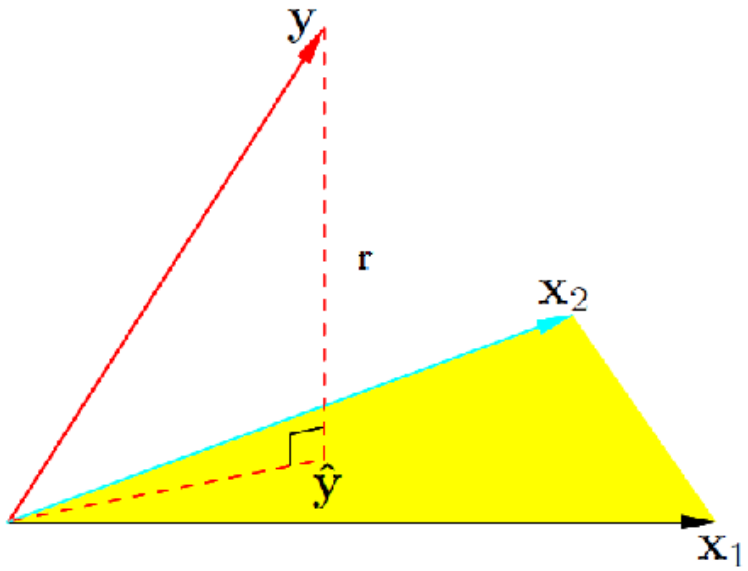
$$\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\boldsymbol{y} \quad \text{(formel 2-4)}$$

under forutsetning at matriseproduktet $X'X$ kan inverteres. (Vi skal straks se på hva som skjer i de tilfellene der $X'X$ ikke er invertibel eller når $X'X$ er nært singulær.) Minste kvadraters tilpasning til responsvektoren $\boldsymbol{y}$ blir da:

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} = X(X'X)^{-1}X'\boldsymbol{y} \quad \text{(formel 2-5)}$$

Ved å stille uttrykket opp på denne måten kan dette lett gjenkjennes som ortogonalprojeksjonen av responsen, $\boldsymbol{y}$, ned på søylene i X matrisa.

For en datamatrise med to forklaringsvariable, kan det enkelt fremstilles hvordan OLS fungerer grafisk, under forutsetning at $X'X$ er invertibel. Denne forutsetningen garanterer lineær uavhengighet mellom forklaringsvariablene.



Figur 1.1: *Figur av hvordan responsvektoren* $\boldsymbol{y}$ *i OLS blir ortogonalprojisert ned i et underrom spent ut av forklaringsvariablene* $\boldsymbol{x}_1$ *og* $\boldsymbol{x}_2$ *. Illustrasjonen er hentet fra* (Hastie et al. 2009).

I figur 1.1 tilsvarer den predikerte responsen, $\hat{\boldsymbol{y}}$, ortogonalprojeksjonen av responsen ned i underrommet spent ut av de to forklaringsvariablene $\boldsymbol{x}_1$ og $\boldsymbol{x}_2$ .Når $\hat{\boldsymbol{y}}$ ligger i rommet spent ut av forklaringsvariablene kan regresjonskoeffisientene til hver forklaringsvariabel enkelt finnes som koeffisientene $(\hat{\beta}_0,\hat{\beta}_1,\hat{\beta}_2)' = \hat{\boldsymbol{\beta}}$ som løser likningen

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} = \mathbf{1}\hat{\beta}_0 + \boldsymbol{x}_1\hat{\beta}_1 + \boldsymbol{x}_2\hat{\beta}_2 \quad .$$

Residualen, $\boldsymbol{r}$, til denne OLS løsningen tilsvarer den delen av responsen $\boldsymbol{y}$ som står ortogonalt på underrommet spent ut av $\boldsymbol{x}_1$ og $\boldsymbol{x}_2$. Som vi ser er av figur 1.1 finnes ingen annen lineærkombinasjon av $\boldsymbol{x}_1$ og $\boldsymbol{x}_2$ som kan gi et mindre residual. En hvilken som helst annen kombinasjon vil gjøre at residualvektoren, $\boldsymbol{r}$, blir lengre, og dette fører dermed til en dårligere tilpasning av de tilgjengelige dataene.

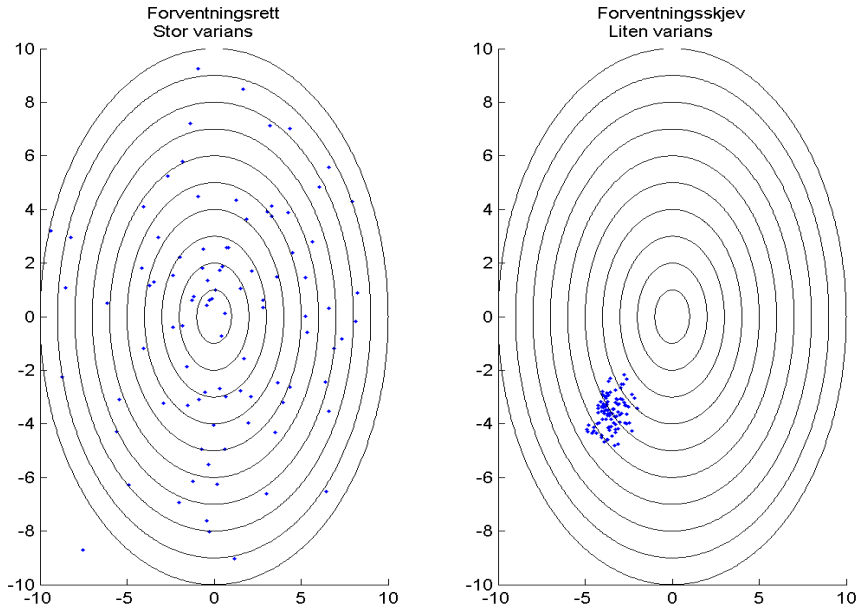
1.2.1 Varians og forventningsskjevhet

OLS er en såkalt forventningsrett metode. Dette betyr at i det lange løp kan det forventes at denne metoden (dersom den er korrekt) gir de sanne regresjonskoeffisientene som representerer sammenhengen mellom responsen og hver forklaringsvariabel. Det kan også vises at OLS blant de forventningsrette metodene er den beste (Hastie et al. 2009). Det kan virke som om at man får «i både pose og sekk» ved å bruke denne metoden. Men dette er dessverre ikke bestandig tilfelle.

Estimatet på regresjonskoeffisientene vi får med OLS kan ofte ha betydelig varians. I

enkelttilfeller kan vi derfor oppleve at metoden gir oss modeller som predikerer dårlig, fordi estimatet av regresjonskoeffisientene er for usikkert. Et interessant alternativ kan derfor være å bruke metoder som er forventningsskjeve, men som kompenserer for dette ved å ha lavere varians. Med forventningsskjevhet menes at metoden i det lange løp ikke gir oss de sanne regresjonskoeffisientene. Men dersom en liten forventningsskjevhet kan bidra til en stor reduksjon i variansen kan allikevel denne typen metoder gi modeller med forbedret prediksjonsevne.

Som illustrasjon på forventningsskjevhet og varians kan man trekke en analogi til blinkskyting på en skytebane. Anta at to personer skyter på en blink, der skytter 1 har stor varians og er forventningsrett i skytinga si, mens skytter 2 har liten varians men er forventningsskjev i skytinga. Det ideelle hadde vært å kombinert liten varians med forventningsrett skyting, men dette har vi dessverre sjelden anledning til. Resultatet fra de to skytterne kan illustreres som under:



Figur 1.2: *Illustrasjon av forventningsskjevhet og varians.*

Hvem av skytterne som skyter best avhenger av hvor forventningsskjev skytter 2 er, og hvor stor reduksjon i varians denne forventningsskjevheten kan gi. Er forventningsskjevheten relativt liten, og reduksjonen i varians stor, vil trolig skytteren med forventningsskjevhet oppnå den beste poengsummen. På samme måte for statistiske metoder har vi at hvis forventningsskjevheten er relativt liten og reduksjonen i varians tilstrekkelig stor, så kan forventningsskjeve modeller gi bedre estimater på regresjonskoeffisientene.

1.2.2 Problemer med OLS – kolinearitet og p > n

Som nevnt tidligere forutsetter vi at $X'X$ er invertibel når vi skal estimere regresjonskoeffisientene i OLS. Når $X'X$ er invertibel garanterer dette at OLS har en unik løsning for estimatene av regresjonskoeffisientene $\boldsymbol{\beta}$. I de tilfellene inversen av $X'X$ er nært singulær vil de estimerte OLS regresjonskoeffisientene bli svært ustabile. Vi skal straks gå litt nærmere inn på dette, men først ser vi på hva som skjer når $X'X$ ikke er invertibel.

Fra lineær algebra er det kjent at en vilkårlig matrise A maksimalt har rangen $r = min(n, p)$, der n er antall rader og p er antall søyler i matrisa. Når ei matrisa har flere søyler enn rader blir rangen til matrisa maksimalt lik n. Overfører vi dette til estimering av regresjonskoeffisientene i OLS ser vi at vi får problemer med å beregne inversen av $X'X$ (formel 2-4). Når antallet observasjoner er mindre enn antallet forklaringsvariable i datasettet vårt får datamatrisa X maksimalt rang lik antall observasjoner, altså n. $X'X$ er i dette tilfellet en kvadratisk p x p matrise med rang maksimalt lik n. Når antallet forklaringsvariable er større enn antall observasjoner, $p > n$, kan vi ikke finne noen vanlig invers til $X'X$ da dette krever at rangen må være p (Lay 2006). Vi kan med andre ord ikke estimere regresjonskoeffisientene til OLS med denne formelen i det hele tatt.

Dette betyr ikke at OLS ikke har en løsning, men at OLS faktisk har uendelig mange løsninger til dette problemet. Dette er fordi man fortsatt kan ortogonalprojisere responsen $\boldsymbol{y}$ ned i underrommet spent ut av vektorene som beskriver forklaringsvariablene, men denne projeksjonen lar seg ikke lenger beskrive som én unik lineærkombinasjon av forklaringsvariablene

$$\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}} = \mathbf{1}\hat{\beta}_0 + \boldsymbol{x}_1\hat{\beta}_1 + ... + \boldsymbol{x}_p\hat{\beta}_p$$

Da datamatrisa X har rang maksimalt lik n, og $p > n$ garanteres at minst en av forklaringsvariablene kan skrives som en lineærkombinasjon av de andre. Derfor eksisterer ikke lenger én unik, men mange lineærkombinasjoner av forklaringsvariablene som gir oss projeksjonen.

I praksis betyr dette at det altså finnes uendelig mange kombinasjoner av regresjonskoeffisientene som gir OLS-løsningen. Det som imidlertid lar seg entydig beregne er vektoren med regresjonskoeffisienter som har lavest norm. Denne kan man bestemme ved hjelp av en såkalt pseudoinvers som lar seg beregne via singulærverdidekomposisjon. Dette omtales nærmere under avsnittet om projeksjonsmetodikk.

Selvom vi kan finne den kombinasjonen av regresjonskoeffisienter som har lavest norm, gir dette ingen garanti for en modell som gir gode prediksjoner ved anvendelse på nye data. Det at det finnes uendelige mange løsninger betyr fra et tolkningsperspektiv at det finnes uendelig mange kombinasjoner av måter å forklare relasjonen mellom responsen og forklaringsvariablene. Det er derfor særdeles hensiktsmessig å søke alternative metoder som er i stand til å produsere modeller som kan gi en sikrere forståelse av relasjonen mellom responsen og forklaringsvariablene.

Derimot når inversen av $X'X$ er nært singulær, vil vi som nevnt kunne få én unik løsning for OLS. De vanskelighetene vi støter på i dette tilfellet er at estimatene på regresjonskoeffisientene vil ha stor varians, og dermed stor usikkerhet – noe som igjen medfører alvorlige tolkbarhetsproblemer. Et tilfelle der inversen av $X'X$ blir nært singulær er når forklaringsvariablene er høyt korrelerte. Et fenomen det er vanlig å observere i tilfeller med høyt korrelerte forklaringsvariable er at man ofte får relativt store regresjonskoeffisienter med motsatt fortegn i OLS-løsningen, helt uavhengig av styrken på deres korrelasjon med responsen. Dette kan dermed gi et svært galt inntrykk av hvilke forklaringsvariable som i virkeligheten har noen betydning for responsen.

Kort oppsummert er de to punktene som gir oss utfordringer når vi benytter OLS direkte:

- Datamatrisa X har flere forklaringsvariable (p) enn antall observasjoner (n), dvs $p > n$
- Forklaringsvariablene er høyt korrelerte

I begge tilfellene (og i kombinasjon av de to tilfellene) blir resultatet gjerne at OLS produserer modeller som gir hverken god prediksjon eller informative og tolkbare modeller. Det fins flere mulige løsninger på problemene nevnt over, og vi skal straks se nærmere på noen av dem.

- Den første typen omtales gjerne som variabelseleksjonsmetoder. Disse tar utgangspunkt i å redusere antallet forklaringsvariable i datasettet uten tap av informasjon som bidrar til prediksjon.
- Den andre typen metoder er krympinsmetoder som regulariserer datamatrisa X, slik at støy filtreres bort og forklaringsvariable med innvirkning på prediksjonen blir identifisert og inkludert i modeller.
- Den tredje er projeksjonsmetoder som lager et datasett med nye forklaringsvariable

laget med lineærkombinasjoner av de opprinnelige forklaringsvariablene (gjerne kalt faktorer eller komponenter).

1.3 Variabelseleksjonsmetoder

For å redusere antallet forklaringsvariable i datasettet kan vi benytte metoder med kriterier som selekterer ut hvilke variable som skal tas med i en avsluttende OLS modell. Vi søker med andre ord en forenklet versjon av datasettet som inneholder færre variable helst uten betydelig tap av verdifull informasjon. Har vi for eksempel flere forklaringsvariable som spenner ut mye av det samme underrommet, vil det være aktuelt å kun bruke et mindre antall forklaringsvariable som fremdeles representerer store deler av det opprinnelige underrommet. På samme måten er det aktuelt å velge bort variable som kun ser ut til å bidra med støy i modellen. Variabler som selekteres ut under modelleringen angis i den endelige modellen ved at regresjonskoeffisientene er satt lik 0. Dette betyr samtidig at slike modeller introduserer noe forventningsskjevhet, men at variansen i regresjonskoeffisientene reduseres – forhåpentligvis såpass mye at disse modellene oppnår bedre prediksjonsegenskaper.

Variabelseleksjonsmetodene er også til hjelp med hensyn på tolkbarhet. De forklaringsvariablene med antatt størst effekt på responsen er de som i den endelige modellen har regresjonskoeffisienter forskjellig fra 0.

1.3.1 Forward stepwise selection

Forward stepwise selection (Hastie et al. 2009), på norsk forover stegvis seleksjon, er blant de enkleste variabelseleksjonsmetodene vi har. Metoden starter med en tom modell og inkluderer én og én variabel i hver runde inntil modellen er god nok eller når alle variablene er inkludert. I hver runde er variabelen med høyest absolutt korrelasjon til residualen kandidat for å bli inkludert i modellen. Dette tilsvarer også variabelen med størst verdi for F-observatoren F_0 i en *partiell F-test*. Variabelen inkluderes i modellen dersom F-observatoren F_0 er høyere enn en predefinert F-verdi. Dersom variabelen inkluderes, tilpasses modellen på nytt og den nye residualen $\boldsymbol{r} = \boldsymbol{y} - \hat{\boldsymbol{y}}$ beregnes. Er derimot F-observatoren F_0 til variabelen lavere enn den predefinerte F-verdien, stopper metoden og modellen som foreligger på daværende tidspunkt blir den endelige modellen.

Tanken bak å inkludere én og én forklaringsvariabel i modellen, er at metoden kun inkluderer de virkelig «viktige» variablene, og gir modeller som beskriver de antatt mest betydningsfulle

relasjonene mellom forklaringsvariablene og responsen. En ulempe med metoden er at den risikerer å ikke inkludere viktige forklaringsvariable fordi de tilsynelatende ikke er betydningsfulle. I tillegg kan det å inkludere én og én variabel hindre metoden til å oppdage mer komplekse strukturer i datasettet. Modellene som produseres kan da bli for enkle, noe som igjen kan føre til dårlig prediksjonsevne på nye observasjoner. På den annen side er fordelen at det er liten sannsynlighet for at metoden produserer overtilpassede modeller.

1.3.2 Forward stagewise selection

Forward stagewise selection (Hastie et al. 2009), på norsk forover trinnvis seleksjon, ligner veldig på forover stegvis seleksjon men er mer kontinuerlig i utvelgelsen av variabler. Metoden tar på samme måte som forover stegvis seleksjon utgangspunkt i en tom modell, men den inkluderer kun en liten bit av hver variabel i hvert steg. Hvor stor bit som inkluderes i hvert steg er en justerbar parameter, og meningen er at den skal inkludere variabler mer kontinuerlig enn forover stegvis seleksjon. Variabelen som er kandidat for å få inkludert en bit av seg til modellen, er variabelen med høyest absolutt korrelasjon til responsen. På samme måte som for forover stegvis seleksjon inkluderes variabelen i modellen dersom F-observatoren F_0 er større enn en predefinert F-verdi. Når variabelen velges, projiseres responsvektoren ned på denne variabelen og man beregner residualen $\boldsymbol{r} = \boldsymbol{y} - \hat{\boldsymbol{y}} \cdot \delta$, der δ angir hvor stor del av projeksjonen som skal trekkes ut av responsvektoren. Utifra dette beregnes hvor stor del av variabelen som blir inkludert i modellen. I neste steg er variabelen med høyest korrelasjon til residualen kandidat for å bli inkludert i modellen. Metoden stopper når variabelen med høyest korrelasjon til residualen har en lavere F-observator F_0 enn den predefinerte F-verdien.

På grunn av de mange trinnene som utføres blir metoden mer beregningskrevende sammenlignet med forover stegvis seleksjon. Fordelen er at metoden inkluderer variable på en mer demokratisk måte. Det har imidlertid vist seg at denne måten å modellere på er svært gunstig når man jobber med datasett med relativt mange forklaringsvariable. Det er også observert at metoden gir liknende resultater med lasso, som senere blir omtalt.

1.3.3 Backward stepwise selection

Backward stepwise selection (Hastie et al. 2009), på norsk bakover stegvis seleksjon, er en

seleksjonsmetode som i motsetning til de to vi allerede har nevnt ekskluderer forklaringsvariable fra modellen. Metoden tar utgangspunkt i en modell der samtlige forklaringsvariable er inkludert, ekskluderer én og én variabel i hver runde. Algoritmen stopper når vi har oppnådd en god nok modell eller når alle variablene er ekskludert fra modellen. I hver runde beregnes F-observatoren F_0 for hver variabel, utifra hvordan modellen ser ut med og uten denne variabelen. Variabelen med lavest F-observator F_0 er kandidat for å bli ekskludert fra modellen, og dersom F-observatoren har lavere verdi enn en predefinert F-verdi ekskluderes variabelen. Er derimot F-observatoren F_0 høyere enn den predefinerte F-verdien, stopper metoden og modellen som foreligger på daværende tidspunkt blir den endelige modellen.

Fordelen med metoden er at forklaringsvariable med åpenbar relasjon til responsen er inkludert i modellen allerede fra starten av, slik at man ikke risikerer at disse uteblir fra modellen, noe som nevnt tidligere kan være ett problem med forward stepwise selection. I tillegg ekskluderer metoden forklaringsvariable som ikke bidrar betydelig til bedre prediksjonen eller er vi antar er betydelig påvirket av støy, og dette fører ofte til at modellene blir mer stabile.

Ulempen med denne metoden er derimot at ofte har den vanskeligheter med å ekskludere høyt korrelerte variable, til tross for at de har liten korrelasjon til responsen. Er to høyt korrelerte variable inkludert i modellen, ser vi ofte at disse får relativt store regresjonskoeffisienter med motsatt fortegn. Ekskludering av en av disse variablene fører til en stor økning i residualen, noe som igjen fører til en relativt høy F-observator. Dermed oppfattes variablene som svært betydningsfulle for god prediksjon, og blir ikke ekskludert fra modellen. Høyt korrelerte variable hjelper dermed hverandre til å ikke bli ekskludert fra modellen, og fortolkningen av modellen gir et galt inntrykk av hvilke variable som er viktige for å kunne beskrive responsen.

1.3.4 Forward/Backward stepwise selection og Subset selection

Forover og bakover stegvis seleksjon er avhengige av at seleksjonen som utføres underveis er optimale ikke bare der og da, men også senere i algoritmen, da variable som først blir inkludert eller ekskludert i modellen forblir i modellen. For ikke å legge så mye ansvar på enkelt-iterasjoner i algoritmen, har det blitt utviklet alternativer som gjør seleksjonen met robust. Eksempler på dette er bakover/forover- og forover/bakover stegvis seleksjon (Montgomery et al. 2001). Disse metodene tar utgangspunkt i henholdsvis fulle og tomme modeller, og kombinerer muligheten til å ekskludere og inkludere variable i hvert steg i

algoritmen. Også her brukes en partiell F-test for å avgjøre om en variabel skal ekskluderes eller inkluderes i hvert trinn. Dermed tillates det å «gjøre om» på valg i tidligere iterasjoner, som i en senere runde viser seg å være ugunstige. Disse metodene kan gi oss modeller som forover- og bakover stegvis seleksjon ikke klarer å finne på grunn av deres mer «enveiskjørte» natur.

Felles for variabelseleksjonsmetodene vi har sett på til nå, er at de følger en «sti» hvor vi gjør ulike veivalg, der variabler blir inkludert eller ekskludert fordi de på det tidspunktet viste seg å være de mest gunstige. Disse stiene begrenser hvilke kombinasjoner av variabler som er aktuelle for modellen, da modellbygningen skjer steg for steg. Gruppevis seleksjon (Hastie et al. 2009) er en metode som er mye friere enn de ovennevnte når den velger hvilke variabler som kan være med i modellen. Metoden finner den beste kombinasjonen av ett forhåndsvalgt antall forklaringsvariable til modellen, basert på et kriterie om lavest mulig verdi av Mallows CP (Montgomery et al. 2001). Dermed unngåes det å ende opp med «lokalt optimale» modeller slik som i forover og bakover stegvis seleksjon. På den annen side er det å velge fritt blant alle forklaringsvariablene en svært optimistisk form for modellbygging, og man må passe på at ikke modellen blir overtilpasset.

1.4 Krympingsmetoder

Med unntak av forover trinnvis seleksjon er metodene innenfor variabelseleksjon diskrete i måten de velger ut variabler på. Dette at en hel variabel blir inkludert eller kastet ut av en modell kan gi modellen stor varians med tilhørende dårlig prediksjon [Hastie T. Et al, 2009]. En annen type metode tar utgangspunkt i å krympe regresjonskoeffisientene for å få en mer kontinuerlig bygging av modellen, og dermed kunne redusere variansen i estimatene av regresjonskoeffisientene. Ved å krympe regresjonskoeffisientene innføres forventningsskjevhet for modellen, og en liten forventningsskjevhet kan ofte innebære en gunstig kompensering for en ellers stor usikkerhet i estimeringen av regresjonskoeffisientene.

Foran beskrev vi OLS som en metode som minimerer kvadratsummen av residualene (formel 2-3)

$$RSS(\boldsymbol{\beta}) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij} \beta_j \right)^2 \quad .$$

Nedenfor skal vi kort presentere metoder som modifiserer OLS-kriteriet for å forbedre den resulterende regresjonsmodellen.

1.4.1 Ridge regresjon

Som kjent får ofte høyt korrelerte forklaringsvariable relativt store regresjonskoeffisienter med ulikt fortegn uavhengig av styrken på korrelasjonen til responsen med OLS. I tolkningen av modellen blir dermed disse forklaringsvariablene ansett som svært betydningsfulle. For å unngå at slike variable får stor innflytelse i en modell innfører ridge regresjon (Hastie et al. 2009) en begrensning på hvor store regresjonskoeffisientene kan være(Hastie et al. 2009). Begrensningen ligger i at størrelsen på kvadratsummen av regresjonskoeffisientene straffes avhengig av en justerbar parameter, λ . For en gitt verdi av straffparameteren estimeres regresjonskoeffisientene slik at uttrykket

$$RSS(\beta)=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2 \quad \text{(formel 2-6)}$$

minimeres. Det første leddet i uttrykket er identisk med OLS kriteriet, og det andre leddet er begrensningen som blir satt med ridge regresjon. Det kan vises at en ekvivalent måte å uttrykke kriteriet i ridge regresjon på er at man skal minimere OLS-kriteriet

$$RSS(\boldsymbol{\beta})=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2 \quad \text{med med hensyn på tilleggsbetingelsen} \quad \sum_{j=1}^{p}\beta_j^2\le t$$

(formel 2-7) ,

der t er den største tillatte verdien for kvadratsummen av regresjonskoeffisientene. Det kan vises at løsningen er analytisk og at regresjonskoeffisientene kan estimeres ved formelen

$$\hat{\boldsymbol{\beta}}^{ridge}=(X'X+\lambda I)^{-1}X'\boldsymbol{y} \quad \text{.(formel 2-8)}$$

Matematisk kan dette forklares med at responsvektoren projiseres ned i underrommet spent ut av prinsipalkomponentene til kovariansmatrisa til datasettet. De største prinsipalkomponentene tilsvarer de retningene eller lineærkombinasjonene av forklaringsvariable som representerer størst variasjon i datasettet, og de minste prinsipalkomponentene de retningene som representerer minst variasjon. Deretter krymper ridge koeffisientene, og koeffisientene til forklaringsvariablene som danner de minste prinsipalkomponenter krympes mest. Vi skal senere gå inn på hva prinsipalkomponenter er.

Det virker i mange sammenhenger å være en rimelig antakelse at responsvariabelen varierer mest i de retningene som forklaringsvariablene også varierer, da man naturlig nok ønsker å studere forklaringsvariable man tror har en sammenheng med responsvariabelen, men dette er ikke alltid tilfellet. Man antar ofte at de retningene i rommet hvor variablene varierer lite i stor

grad er forbundet med støy, og derfor er det positivt at metoden krymper regresjonskoeffisientene som svarer til disse retningene. Vi vil da kunne oppnå mer stabile modeller som fokuserer mest på de antatt viktigste variablene, og oppnå god prediksjon på et uavhengig testsett («nye data»).

Ett resultat av dette blir blant annet at i utgangspunktet store OLS-regresjonskoeffisienter som bidrar lite til å minimere kvadratavviket mellom responsen og den predikerte responsen krympes mest ved bruk av ridge. Spesielt vil høyt korrelerte forklaringsvariable som får relativt store OLS-regresjonskoeffisienter med motsatt fortegn krympes i stor grad. Modellen man finner ved ridge regresjon kan dermed gi en bedre tolkning av hvilke forklaringsvariable som har betydning for responsen.

Ridge regresjon krymper som sagt regresjonskoeffisientene til samtlige forklaringsvariable og dette fører til at enkelte koeffisienter kan bli svært små, men uten at disse blir eksakt lik 0. Dette medfører at metoden ikke kan gjøre variabelseleksjon. Vi skal straks se på krympingsmetoder som også gir ekte variabelseleksjon.

1.4.2 Lasso

Lasso er forkortelsen for least absolute shrinkage and selection operator, på norsk minste absolutte krymping og seleksjons metode (Hastie et al. 2009). Metoden har noen av de samme egenskapene som ridge regresjon og variabelseleksjon. På samme måte som ridge regresjon legger lasso en begrensning på størrelsen på regresjonskoeffisientene, men lasso bruker summen av absoluttverdien til regresjonskoeffisientene som grunnlag for hva som skal straffes ved multiplikasjon med parameteren, λ . For en gitt parameterverdi estimeres regresjonskoeffisientene i lasso slik at uttrykket

$$RSS(\lambda)=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{j=1}^{p}|\beta_j| \quad \text{(formel 2-9)}$$

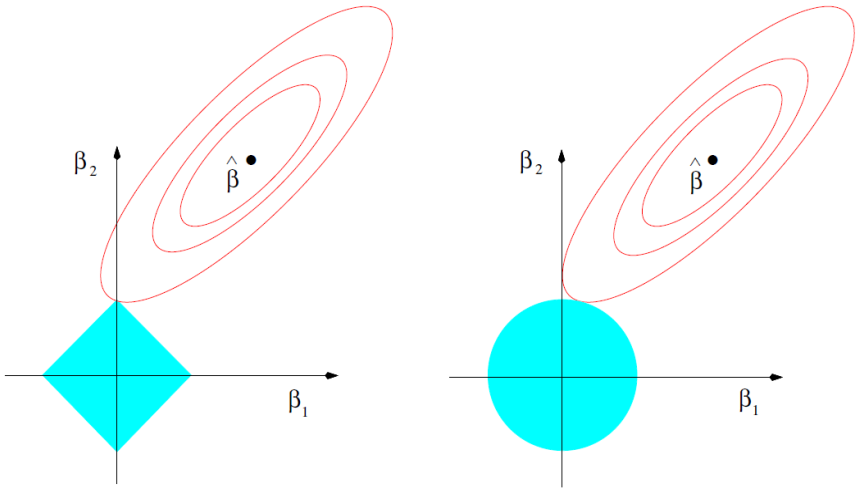
minimeres. Det kan vises at en ekvivalent måte å uttrykke lasso kriteriet på er å minimere OLS-kriteriet

$$RSS(\boldsymbol{\beta})=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2 \quad \text{med hensyn på tilleggsbetingelsen} \quad \sum_{j=1}^{p}|\beta_j|\le t \quad \text{(formel 2-10)},$$

der t er den største tillatte summen av absoluttverdiene til regresjonskoeffisientene. Denne begrensningen krymper ikke bare regresjonskoeffisientene, men den tenderer i tillegg til å

sette regresjonskoeffisienter lik 0, med andre ord utfører metoden også en ekte variabelseleksjon. Fordelen med lasso er at den dermed kan produsere enklere og potensielt mer tolkbare modeller enn OLS.

For å gi en pekepinn på hvordan ridge og lasso modellene beregnes kan det være nyttig å kikke på en illustrasjon (Figur 1.3) av hvordan dette kan foregå på et datasett med kun to forklaringsvariable:



Figur 1.3: *Illustrasjon av hvordan lasso (venstre) og ridge (høyre) estimerer regresjonskoeffisienter. De blå områdene tilsvarer begrensningene av regresjonskoeffisientene gitt ved* $|\beta_1|+|\beta_2|\leq t$ *og* $\beta_1^2+\beta_2^2\leq t$ *, og regresjonskoeffisienter langs en rød kurve har like stort verdi på OLS-kriteriet (RSS). Illustrasjonen er hentet fra* (Hastie et al. 2009).

Av figur 1.3 ser vi at området for tillatte regresjonskoeffisienter er en firkant med skarpe hjørner for lasso, og en sirkel for ridge regresjon. Hver metode finner det punktet av regresjonskoeffisienter, (β_1, β_2) , i det tillatte området som minimerer OLS-kriteriet. For lasso-området har vi at hvis løsningen treffer et hjørne, blir den ene regresjonskoeffisienten i modellen lik 0, og vi får variabelseleksjon. Ridge-området har ikke noen spisse hjørner, og det er dermed så godt som utelukket å få variabelseleksjon med ridge.

Fenomenet viser seg å fungere på tilsvarende måte også for datasett med mer enn to forklaringsvariable, da det tillatte lasso-område blir et romboid med mange hjørner og kanter. Dermed vil lasso-modeller ofte ha at flere regresjonskoeffisienter blir lik 0. En begrensning ved metoden er at den kun har mulighet til å inkludere så mange forklaringsvariable i modellen som det er antall observasjoner i datasettet. Dette kan føre til at metoden gir for enkle modeller til å kunne få god prediksjon. Dette kan også være problematisk i forhold til tolkbarhet, da potensielt betydningsfulle forklaringsvariable ikke kan bli inkludert i modellen.

Lasso tenderer til å oppføre seg som forover stegvis seleksjon ved at den velger kun én høyt

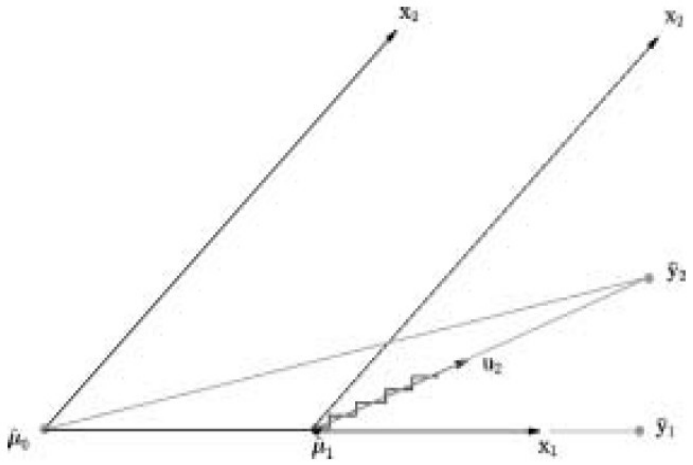
korrelert variabel blant flere, for så å ha vanskeligheter med å velge inn andre korrelerte variabler. Dette kan i enkelte tilfeller være uheldig, da vi kan få tilfeller der en variabel «skygger» for den variabelen som har viktig informasjon om responsvariabelen.

I tilfellene der antall observasjoner, n , er større enn antall variabler, p , og variablene er høyt korrelerte har man ved empiri sett at Ridge regresjon gjør bedre prediksjoner enn lasso. En annen ulempe med det ikke finnes en analytisk løsning for regresjonskoeffisientene slik som det gjør for ridge regresjon. Man hadde lenge trege og beregningstunge algoritmer for å finne løsninger for lasso, noe som gjorde metoden mindre konkurransedyktig med andre raskere metoder. Men en metode har vist seg som svært egnet til effektiv beregning av lasso-regresjonskoeffisientene for enhver straffparameterverdi, og den skal vi se nærmere på i neste avsnitt.

1.4.3 LARS

LARS er forkortelsen for least angel regression, på norsk minste vinklers regresjon (Efron et al. 2004) og er relativt ny blant regresjonsmetodene som utfører variabelseleksjon. Grunnen til at vi nevner metoden under dette delkapittelet er at den har vist seg å kunne estimere regresjonskoeffisientene i lasso svært effektivt.

LARS har likhetstrekk med forover trinnvis seleksjon, ved at den starter med en tom modell og inkluderer deler av forklaringsvariable i modellen. Den variabelen som velges i første runde er den med høyest korrelasjon med responsvariabelen. I motsetning til forover trinnvis seleksjon beregner LARS eksakt hvor stor del av denne variabelen som kan inkluderes før en av de utelatte variablene har like stor korrelasjon med residualen. Dette kan gjøres på grunnlag av teori om kovariansen til variablene og at algoritmen del for del er lineær. I neste runde projiseres en så stor del av residualen ned på disse to variablene, slik at en tredje variabel får like stor korrelasjon med residualen. For et datasett med to forklaringsvariable kan vi illustrere hvordan LARS-algoritmen opererer geometrisk:



Figur 1.4: *Illustrasjon av LARS-algoritmen på et datasett med to forklaringsvariable. Illustrasjonen er hentet fra* (Efron et al. 2004).

I figuren har vi to forklaringsvariable, $\boldsymbol{x}_1$ og $\boldsymbol{x}_2$, som spenner ut et to dimensjonalt rom, og $\bar{\boldsymbol{y}}_2$ er projeksjonen av responsen $\boldsymbol{y}$ ned i dette rommet (som tilsvarer OLS-løsningen). Utgangspunktet er en tom modell, $\hat{\boldsymbol{\mu}}_0=\mathbf{0}$, og residualvektoren er dermed $\bar{\boldsymbol{y}}_2-\mathbf{0}$. $\boldsymbol{x}_1$ har høyere korrelasjon med residualen enn $\boldsymbol{x}_2$, derfor inkluderes den delen av $\boldsymbol{x}_1$ i modellen som bidrar til at $\boldsymbol{x}_1$ har størst korrelasjon til residualen enn $\boldsymbol{x}_2$. Vi får da modellen $\hat{\boldsymbol{\mu}}_1=\hat{\boldsymbol{\mu}}_0+\gamma\boldsymbol{x}_1$, som tilsvarer når korrelasjonen mellom den nye residualen og henholdsvis $\boldsymbol{x}_1$ og $\boldsymbol{x}_2$ er lik. Deretter inkluderes begge variablene til modellen til vi kommer til $\bar{\boldsymbol{y}}_2=\hat{\boldsymbol{\mu}}_0+\gamma_1\boldsymbol{x}_1+\gamma_2(\boldsymbol{x}_1+\boldsymbol{x}_2)$.

I punktet $\hat{\boldsymbol{\mu}}_1$ har begge variablene lik korrelasjon til residualen som også betyr at vinkelen mellom hver av variablene og residualen er lik, og i punktet $\hat{\boldsymbol{\mu}}_0=\mathbf{0}$ hadde vektoren til variabel $\boldsymbol{x}_1$ minst vinkel med residualen. Fra figur 1.4 ser vi at LARS søker å inkludere variable til modellen som hele tiden har lavest vinkel (og dermed størst korrelasjon) til residualen, og det er også herfra metoden har fått navnet. Ofte brukes lengden på «stien» som mål på kompleksitet for modellene fra LARS. Regresjonskoeffisientene til modellen tilsvarer lineærkombinasjonen av forklaringsvariablene som gir modellen med ønsket lengde, i eksempelet på figuren får vi at OLS løsninger får regresjonskoeffisienter gitt ved $\bar{\boldsymbol{y}}_2=\hat{\boldsymbol{\mu}}_0+\gamma_1\boldsymbol{x}_1+\gamma_2(\boldsymbol{x}_1+\boldsymbol{x}_2)=\hat{\boldsymbol{\mu}}_0+(\gamma_1+\gamma_2)\boldsymbol{x}_1+\gamma_2\boldsymbol{x}_2$.

Denne «stien» med regresjonkoeffisienter LARS finner er svært lik regresjonskoeffisientene som minimerer OLS-kriteriet i lasso, med en økende verdi på straffparameteren $\sum_{j=1}^{p}|\beta_j|\leq t$, og med en bestemt modifikasjon i LARS-algoritmen kan man få den til å gi den eksakt

samme løsning som lasso. Fordelen med LARS er at den er svært effektivt beregner regresjonskoeffisientene i lasso sammenlignet med tidligere teknikker, noe som gjør lasso mye mer attraktiv å benytte.

1.4.4 Elastic net

Når man med motivasjon fra egenskapene til variabelseleksjon og ridge regresjon lagde metoden lasso, var det interessant å se hvordan den var som metode sammenlignet med de andre. Man fant da ut at hver av disse metodene var overlegne de andre i noen tilfeller, men det var aldri sånn at én enkelt pekte seg ut som den beste (Zou & Hastie 2005). I tilfellene der antall observasjoner, n , er større enn antall variabler, p , og variablene er høyt korrelerte så man ved empiri at ridge regresjon gir modeller med bedre prediksjonsegenskaper enn lasso. I tillegg har lasso en begrensning på at modellene den produserer maksimalt kan ha like mange regresjonskoeffisienter som det er observasjoner i datasettet. Lasso var allikevel den av metodene som utførte krymping og variabelseleksjon, og var derfor den mest appellerende videreutvikle. De to foreslo da en hybrid med ideer hentet fra lasso og Ridge regresjon som kalles Elastic Net (Zou & Hastie 2005). Metoden tar utgangspunkt i å minimere uttrykket

$$RSS(\boldsymbol{\beta},\lambda,\alpha)=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2+\lambda\sum_{i=1}^{p}\left((1-\alpha)\beta_j^2+\alpha|\beta_j|\right) \quad \text{(formel 2-11)}$$

, der λ er en straffparameter som begrenser størrelsen på regresjonskoeffisientene, og α angir andelen straff som er lasso basert og (1- andelen) som er ridge basert. Elastic net kan dermed sees på som en generalisering av ridge regresjon og lasso. En ekvivalent måte å uttrykke elastic net på er at man skal minimere uttrykket

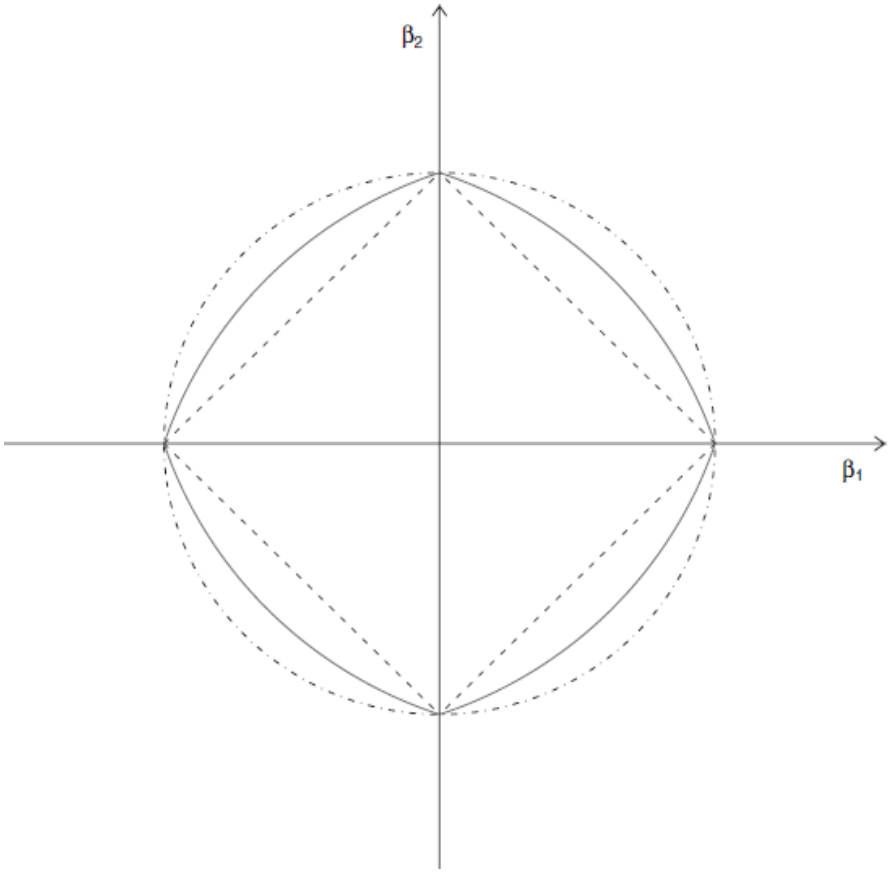
$$RSS(\boldsymbol{\beta})=\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_j\right)^2 \quad \text{med hensyn på} \quad \sum_{j=1}^{p}|\beta_j|\le t_1 \quad \text{og} \quad \sum_{j=1}^{p}\beta_j^2\le t_2$$

(formel 2-12).

Som lasso har ikke metoden en analytisk løsning, men ved hjelp av en modifisert utgave av LARS-algoritmen, LARS-EN, har man også muligheten til effektivt å finne elastic net løsningen.

Da elastic net er en hybrid mellom lasso og ridge regresjon, er det ikke helt uventet at den oppfører seg som en mellomting av hver av metodene. Dette kan vi få et innblikk i ved å studere hvilke områder som avgrenser de tillatte regresjonskoeffisientene. I figur 1.3 så vi at lasso hadde et område formet som et kvadrat som avgrenset mulige regresjonskoeffisienter for

metoden, mens ridge hadde et område formet som en sirkel. Figur 1.5 viser hvordan regresjonskoeffisientene i elastic net begrenses i forhold til lasso og ridge regresjon.



Figur 1.5: *Illustrasjon av område for tillatte regresjonskoeffisienter i to dimensjoner for ridge (ytterst, .-.-.), lasso (innerst, ----) og elastic net (i midten,—). Illustrasjonen er hentet fra* (Zou & Hastie 2005).

Elastic net er en metode som rapporteres som svært konkurransedyktig både i forhold til prediksjon og modellforenkling. Til forskjell fra lasso velger metoden flere variable som har korrelasjon til responsvariabelen uavhengig av om de innbyrdes er høyt korrelerte. Dette gjør at metoden åpner for gruppevis å inkludere variabler som inneholder felles informasjon. Dette kan være med å gi mer stabile modeller som fordeler «ansvar» på flere forklaringsvariable. Metoden utfører også variabelseleksjon på grunn av at lasso restriksjonen er inkludert, og på grunn av ridge bidraget krymper den regresjonskoeffisientene til korrelerte variable. Metoden har som mål å oppdage strukturer i datasettet som gjør at den fanger opp hovedtrekkene av sammenhengen mellom responsvariabelen og variablene. Håpet er at man til slutt sitter igjen med «the big fish», og derav fikk metoden navnet elastic net.

1.5 Projeksjonsmetodikk

Med flere høyt korrelerte forklaringsvariable er det problematisk å bruke OLS til å finne en

fungerende regresjonsmodell. Som vi har vært inne på tidligere er variabelseleksjon et alternativ som kan løse problemet. En alternativ løsning er å benytte et avgrenset antall lineærkombinasjoner av forklaringsvariablene i datasettet og deretter gjøre OLS på disse. Slike lineærkombinasjoner kalles ofte for komponenter eller faktorer. Tanken er å velge komponentene på en sånn måte at vi med et lite antall komponenter kan beskrive datasettet uten store tap av viktig informasjon, og vi får dermed redusert dimenasjonaliteten i problemet. Dette betyr samtidig at metodene introduserer noe forventningsskjevhet, men forhåpentligvis reduseres variansen i regresjonskoeffisientene såpass mye at modellene oppnår bedre prediksjonsegenskaper.

PLS og PCR (se nærmere beskrivelse nedenfor) er to måter for modellering av slike komponenter. Dette er også svært nyttig når vi jobber med datasett der antallet observasjoner, n, er mindre enn antallet variable, p. Som vist tidligere får vi ikke noen entydig løsning for OLS når $p > n$, men vi kan alternativt benytte OLS på lineærkombinasjoner av de originale forklaringsvariablene.

1.5.1 SVD – PCA

SVD er forkortelse for Singular Value Decomposition, på norsk singulærverdidekomposisjon (Lay 2006). Med SVD faktoriseres en matrise A som et produkt av tre ulike matriser:

$A_{nxp} = U_{nxn}\ \Sigma_{nxp}\ V'_{pxp}$. For en A matrisa med rang r, får vi at Σ er konstruert på følgende måte:

$$\Sigma_{nxp} = \begin{bmatrix} D_{rxr} & 0_{rx(p-r)} \\ 0_{(n-r)xr} & 0_{(n-r)x(p-r)} \end{bmatrix}$$

der D inneholder singulærverdiene langs diagonalen i synkende rekkefølge og nuller utenfor diagonalen. U og V' er ortogonale matriser som spenner ut henholdsvis rad- og søylerommet med ortonormale vektorer. Uttrykket kan i tillegg komprimeres ytterligere. Hvis rangen til matrise A er mindre enn antall rader eller antall søyler i matrisa, kan vi skrive A som $A_{nxp} = U_r D V_r'$. Her har U_r de r første søylene i U matrisa og V_r' har de r første radene i V' matrisa.

Fordelen med å kunne gjøre denne faktoriseringen av ei matrise er at vi med et mindre utvalg av vektorer kan beskrive matrisa med ønsket grad av presisjon. Vektorene er allerede rangert etter hvor mye informasjon de inneholder. Dette kan brukes til å redusere støy i et datasett.

Vi kan også bruke SVD til å lage en såkalt pseudoinvers av matrisa, i litteraturen kjent som Moore-Penrose inversen. Da matrisa A kan skrives som $\underset{nxp}{A} = U_r D V_r'$, og D er en diagonal matrise med tall forskjellig fra null langs diagonalen, og U og V' er ortonormale kan vi skrive inversen av matrise A som:

$$A^+ = V_r D^{-1} U_r' \quad \text{(formel 2-13)}$$

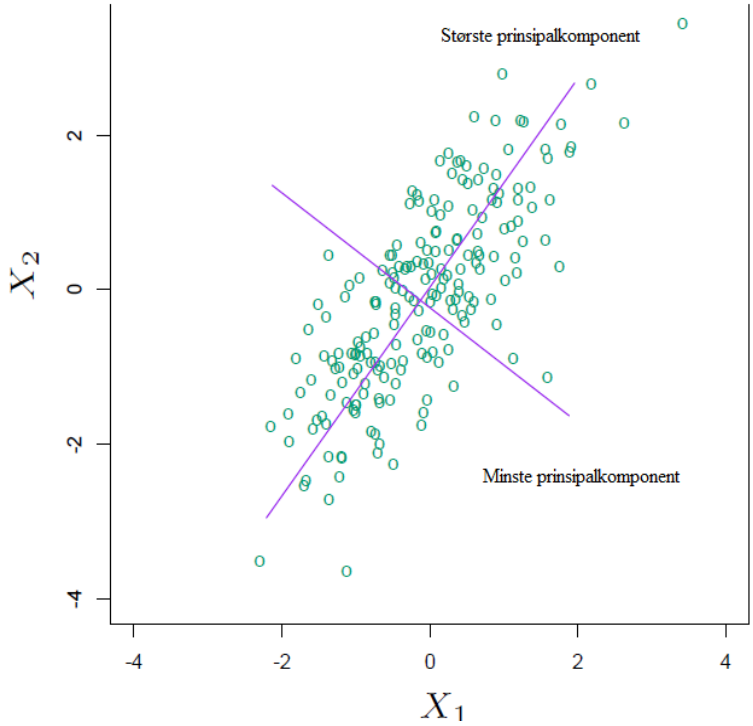
Vi får da at $A^+ A = V_r D^{-1} U_r' U_r D V_r' = I_r$, der I_r er en r ganger r identitetsmatrise til tross for at ikke matrisa A er kvadratisk og ikke vil kunne ha en vanlig invers.

Som nevnt tidligere kan pseudoinversen brukes i OLS for å estimere regresjonskoeffisientene når datamatrisa har flere forklaringsvariable enn antall observasjoner. Estimatet er gitt ved

$\hat{\boldsymbol{\beta}} = X^+ \boldsymbol{y}$ (formel 2-14)

og det kan vises at estimatet gir den kombinasjonen av regresjonskoeffisienter som har lavest norm.

PCA er en forkortelse for Principal Component Analysis, på norsk prinsipalkomponentanalyse (Lay 2006). PCA søker lineærkombinasjoner av forklaringsvariablene som representerer størst mulig del av variasjonen som er tilstede i datasettet. Dette tilsvarer egenvektorene til kovariansmatrisa til datasettet med høyest egenverdi. Man antar at komponentene som representerer størst variasjon i datasettet i stor grad inneholder viktig informasjon, og de komponentene med liten variasjon i stor grad består av støy. Under vises en illustrasjon av prinsipalkomponenter for et datasett med to forklaringsvariable.



Figur 1.6: *Illustrasjon av prinsipalkomponentene til et datasett med to forklaringsvariable. Illustrasjonen er hentet fra* (Hastie et al. 2009).

Beregning av egenvektorene til kovariansmatrisa til datasettet kan være beregningskrevende, spesielt når antallet forklaringsvariable er høyt. En alternativ måte å finne egenvektorene på er med singulærverdidekomposisjon av den sentrerte datamatrisa

$$\underset{n\,x\,p}{X} = U_r D V_r'$$

Singulærverdiene i D matrisa tilsvarer kvadratroten av egenverdiene til egenvektorene til kovariansmatrisa i synkende rekkefølge. Ved å utnytte denne rangeringen kan vi enkelt hente ut de viktigste egenvektorene, slik at vi kan beskrive en stor del av variasjonen i datasettet med et mindre antall komponenter. Dette gjør oss i stand til å eliminere potensiell støy og beskrive forenklinger av datamatrisa X med ønsket presisjon, som avhenger av hvor mange komponenter som benyttes.

1.5.1.1 PCR

PCR er en forkortelse for Principal Component Regression, på norsk prinsipalkomponentregresjon, og er prinsipalkomponentanalyse anvendt i regresjon. Dette er kanskje den mest kjente regresjonsmetoden blant lineærkombinasjonsmetodene. Metoden tar utgangspunkt i komponentene man finner i PCA, og gjør regresjon på disse. Dersom alle

komponentene fra PCA benyttes, spenner disse ut det samme variabelrommet som den opprinnelige X matrisa og PCR gir dermed vanlig OLS regresjon. Ved å utnytte at SVD gir oss komponenter rangert etter viktighet, kan vi med et mindre antall komponenter kan beskrive størst mulig variasjon i datasettet og gjøre PCR på dette.

Ett alternativ til å finne egenvektorer i kovariansmatrisa, er å finne egenvektorene i korrelasjonsmatrisa. Dersom de ulike forklaringsvariablene i et datasett har svært ulik varians, preger ofte de forklaringsvariablene med høyest varians de første komponentene. Ved å finne egenvektorene i korrelasjonsmatrisa kan dette unngås. Ulempen er at man mister informasjon om variasjonen til hver forklaringsvariabel, men gevinsten kan være modeller med bedre prediksjonsegenskaper.

Det å kun bruke variasjonen i datasettet som grunnlag for å finne komponenter gir ingen garanti for at modellene som produseres kan predikere responsen godt. Antakelsen om at komponentene i datasettet med stor variasjon inneholder viktig informasjon for å kunne predikere responsen holder ikke alltid. Ikke sjelden trenger PCR flere komponenter for å gi modeller med gode prediksjonsevner. Vi skal nå se nærmere på en metode som inkluderer responsen når den bestemmer komponentene.

1.5.2 PLS

1.5.2.1 Historikk og tidlig kritikk

PLS danner utgangspunktet for utvikling av den nye regresjonsmetoden som utvikles i denne oppgaven. Det er derfor naturlig å bruke litt ekstra plass på å beskrive denne metodikken. PLS er en forkortelse for Partial least squares (Martens & Næs 1989), på norsk delvis minste kvadraters metode. En alternativ og kanskje vel så dekkende terminologi som blir brukt på forkortelsen av metoden er Projection to Latent Structures, på norsk projeksjon ned på underliggende eller iboende strukturer.

Det som dannet grunnlaget for idéene til metoden startet så tidlig som på midten av 1960-tallet med metoder som NIPALS, Non Iterativ Partial Least Squares (Wold 1966), og deretter PLS path modelling (Wold 1975). På slutten av 1970-tallet og begynnelsen av 1980-tallet gjorde blant annet Herman Wold, Svante Wold og Harald Martens et betydelig arbeid med å utvikle det teoretiske grunnlaget og algoritmen for metoden. Metoden har også klare likhetstrekk til PCR, for på samme måte som PCR bruker variansen til forklaringsvariablene som utgangspunkt for å lage komponenter, bruker PLS kovariansen mellom variablene og

responsen som utgangspunkt.

I starten ble PLS møtt med mye likegyldighet blant tradisjonelle statistikere (Helland 2012). Her var den gjengse oppfatningen at PLS kun var en algoritme, og ikke en tradisjonell statistisk metode. Det ble også stilt spørsmålstegn ved det teoretiske fundamentet til metoden. Denne manglende tilliten til metoden gjorde at enkelte så på PLS som en egen kultur innen statistikkfeltet.

1.5.2.2 Videreutvikling

I dag ser statistikere stort sett på PLS som en av flere mulige regresjonsmetoder. Metoden blir viet mye forskning og det vokser frem nye videreutviklinger av PLS. Da det en stund var noe usikkerhet omkring metodens teoretiske grunnlag, og nettopp dette gjorde at man stod friere til å utforske og utforske og videreutvikle metoden. Ved å kombinere metoden med kjente teoremer eller intuitive ideer har det vokst frem ulike retninger innenfor PLS. Retninger som finnes i dag er blant annet CPLS (Indahl et al. 2008), der PLS-metodikk kombineres med kanonisk korrelasjonsanalyse for å finne relevante underrom, Sparse-PLS (Chun & Keles 2010) , der fokuset er rettet mot at å lage enkle lineærkombinasjoner av forklaringsvariable, St-PLS (Sæbø et al. 2007), der små vektingsvektorene i PLS-algoritmen blir trunkert, og PPLS (Indahl 2005), der en vekter korrelasjonsdelen og standardavviksdelen med potenser. I tillegg benyttes PLS-metodikk kombineret med diskriminantanalyse, for å finne relevante retninger i underrommet spent ut av datamatrisa som kan bidra til god klassifikasjon (Liland 2009).

1.5.2.3 PLS – Algoritme

Følgende kommer beskrivelse av PLS med kovarians mellom den kontinuerlige responsvektoren $\boldsymbol{y}$ og hver variabelvektor i datamatrisa X som utgangspunkt for vektingsvektoren (Martens & Næs 1989). I tillegg til datamatrisa X og responsvektoren $\boldsymbol{y}$ er det vanlig å ha de fire matrisene W, T, P og Q der vektorene som beregnes blir lagret i hver iterasjon. I algoritmen har disse vektorene små bokstaver for de respektive matrisene. I algoritmen vil $\hat{\boldsymbol{q}}_a$ være en skalar og derfor vil Q opptre som en vektor (en matrise med én kolonne). PLS-algoritmen starter vanligvis med at forklaringsvariabelvektorene i datamatrisa og responsen sentreres. Under følger en skisse av algoritmen

Algorime PLS

Utgangspunkt: $X_0 = X - \mathbf{1}\bar{\boldsymbol{x}}$, $\boldsymbol{y}_0 = \boldsymbol{y} - \mathbf{1}\bar{y}$

Den a-te komponenten beregnes etter følgene algoritme:

1. $\hat{\boldsymbol{w}}_a = X'_{a-1}\boldsymbol{y}_{a-1}$
2. $\hat{\boldsymbol{t}}_a = X_{a-1}\hat{\boldsymbol{w}}_a$
3. $\hat{\boldsymbol{p}}_a = X'_{a-1}\hat{\boldsymbol{t}}_a / \hat{\boldsymbol{t}}_a'\hat{\boldsymbol{t}}_a$
4. $\hat{\boldsymbol{q}}_a = \boldsymbol{y}_{a-1}'\hat{\boldsymbol{t}}_a / \hat{\boldsymbol{t}}_a'\hat{\boldsymbol{t}}_a$
5. $X_a = X_{a-1} - \hat{\boldsymbol{t}}_a\hat{\boldsymbol{p}}_a$
6. $\boldsymbol{y}_a = \boldsymbol{y}_{a-1} - \hat{\boldsymbol{t}}_a\hat{\boldsymbol{q}}_a$

Vektingsvektoren, $\hat{\boldsymbol{w}}_a$, består av kovariansen mellom hver forklaringsvariabel i X matrisa og responsen $\boldsymbol{y}$, skalert slik at lengden av vektoren er 1. Score-vektoren, $\hat{\boldsymbol{t}}_a$, blir da lineærkombinasjonen av X matrisa basert på ladningsvektoren. Det er disse score-vektorene som danner grunnlaget for den komprimerte utgaven av X med hjelp av færre komponenter, og det er på disse komponentene vi projiserer responsen ned på. Disse scorevektorene kan lages ortogonale på hverandre, noe som medfører at de er ukorrelerte med hverandre. Som siste ledd i algoritmen projiseres den delen av X matrisa og responsen som ligger i underrommet spent ut av scorevektoren bort. Dette kalles å deflatere X matrisa og responsvektoren og i neste iterasjon vil beregningen av vektorene skje på den deflaterte X matrisa og responsvektoren.

Fordelen med å bruke kovarians som grunnlag for vektingsvektoren i PLS sammenlignet med variansen til hver variabel i PCR, er at responsvektoren er med å påvirker hvordan komponentene lages. Dette fører til at de første PLS komponentene ofte er mer relevante for prediksjon sammenlignet med de første PCR komponentene. Dette fører til at PLS ofte trenger færre komponenter for å produserer gode modeller.

Komponentene som velges gir dermed et gunstig utgangspunkt å gjøre regresjon på. De siste komponentene som produseres i PLS tilsvarer de retningene i variabelrommet som har minst korrelasjon med responsvariabelen. Dette er retninger som er naturlige å anta bidrar lite til å kunne predikere responsen. Nettopp på grunn av dette er PLS attraktiv fordi den gir åpner for å kun inkludere de komponentene som ansees for å være viktigst i modellen. Dermed kan man ved hjelp av et mindre antall komponenter beskrive de viktigste retningene i variabelrommet som bidrar til god prediksjon av responsen.

En ulempe med metoden er at forklaringsvariable med høy varians ofte dominerer i de første komponentene som produseres i algoritmen, og dette bidrar ikke alltid med gode prediksjonsegenskaper til modellen. Kovariansen er som kjent et produkt av korrelasjonen mellom hver forklaringsvariabel og responsen, standardavviket til forklaringsvariabelen og standardavviket til responsen. Når størrelsen på kovariansen avgjør hvor stor vekt en variabel får i vektingsvektoren, kan forklaringsvariable med svært høy varians bli vektet relativt høyt i vektingsvektoren. Dette kan føre til at forklaringsvariable med høy varians preger de første komponentene, noe som ikke alltid bidrar til bedre prediksjonsegenskaper til modellene.

1.5.3 PPLS

PPLS er en forkortelse for Powered PLS, på norsk potensert PLS (Indahl 2005), og er en videreutvikling av PLS-metoden. Metoden tar tak i problemet i PLS med at forklaringsvariable med høy varians blir høyt vektet i vektingsvektoren, og åpner for differensiert vekting mellom disse. PPLS tar utgangspunkt i den samme vektingsvektoren som PLS, men faktoriserer kovariansen til en korrelasjonsdel, en standardavviksdel og en konstant lik 1 med samme fortegn som korrelasjonen:

$$\begin{aligned} \boldsymbol{w}' &= k_1\left[cov(\boldsymbol{y},\boldsymbol{x}_1),\ldots,cov(\boldsymbol{y},\boldsymbol{x}_p)\right] \\ &= k_2\cdot\left[s_1\cdot|corr(\boldsymbol{y},\boldsymbol{x}_1)|\cdot std(\boldsymbol{x}_1),\ldots,s_p\cdot|corr(\boldsymbol{y},\boldsymbol{x}_p)|\cdot std(\boldsymbol{x}_p)\right] \end{aligned} \quad \text{(formel 2-15)}$$

Her er k_1 og k_2 konstanter som garanterer at vektingsvektoren har enhetslengde. Da standardavviket til responsen inngår i samtlige ledd i faktoriseringen, har denne blitt trukket ut og kombiner med konstanten k_2 . For å differensiere vektingen mellom korrelasjon og standardavvik potenseres disse med henholdsvis $(\gamma/(1-\gamma))$ og $((1-\gamma)/\gamma)$ der γ ligger i $U=[0,1]$:

$$\boldsymbol{w}(\gamma)' = k_2\cdot\left[s_1\cdot|corr(\boldsymbol{y},\boldsymbol{x}_1)|^{(\gamma/(1-\gamma))}\cdot std(\boldsymbol{x}_1)^{((1-\gamma)/\gamma)},\ldots,s_p\cdot|corr(\boldsymbol{y},\boldsymbol{x}_p)|^{(\gamma/(1-\gamma))}\cdot std(\boldsymbol{x}_p)^{((1-\gamma)/\gamma)}\right]$$

(formel 2-16).

Det kan gjøres begrensning på U dersom det ikke ønskes for stort fokus på høyt korrelerte forklaringsvariable eller forklaringsvariable med høyt standardavvik i modellen. Settes $U=\{0.5\}$ gir dette vektingsvektorer identiske med de fra PLS. Dersom det ønskes å ha nær variabelseleksjon og spissing av toppene i vektingsvektoren kan man begrense U til kun å ha verdier nær 0 eller nær 1. Dette gjør PPLS til en fleksibel og potensielt kraftfull eksplorativ metode, ved at man kan modellere med ulike begrensninger på parameterverdiene, og derav bygge alternative modeller som kan sammenlignes i forhold til hverandre.

I PPLS er ideen med potenseringen at dette kan bidra til å finne attraktive lineærkombinasjoner av forklaringsvariablene (scorevektorer) med høyere korrelasjon til responsen (eller den deflaterte responsen) enn de vi finner med vanlig PLS. Vi søker i PPLS en potenseringsparameter γ som maksimerer $(corr(\boldsymbol{y}, X\boldsymbol{w}(\gamma)))^2$, der $\boldsymbol{y}$ og X etter den første iterasjon betegner de deflaterte utgavene av responsvektoren og matrisa med forklaringsvariable. Kvadreringen av korrelasjonen gjør at vi kun får positive verdier å forholde oss til i kriteriet som skal maksimeres. Når vi har funnet en vektingsvektor som maksimerer korrelasjonen, brukes denne videre på samme måte som vektingsvektoren i den vanlige PLS-algoritmen. Under følger en skisse av algoritmen:

Algoritme PPLS

Utgangspunkt: $X_0 = X - \mathbf{1}\bar{\boldsymbol{x}}$, $\boldsymbol{y}_0 = \boldsymbol{y} - \mathbf{1}\bar{y}$

Den a-te komponenten beregnes etter følgene algoritme:

1. Sett $\boldsymbol{w}_{(\gamma)}{}' = K_\gamma \cdot [\, s_1 \cdot |corr(\boldsymbol{y}_{a-1}, \boldsymbol{x}_{a-1(1)})|^{\gamma/(1-\gamma)} \cdot std(\boldsymbol{x}_{a-1(1)})^{(1-\gamma)/\gamma}, \ldots$ $, s_p \cdot |corr(\boldsymbol{y}_{a-1}, \boldsymbol{x}_{a-1(1)})|^{\gamma/(1-\gamma)} \cdot std(\boldsymbol{x}_{a-1})^{(1-\gamma)/\gamma}]$
2. Finn $\{\boldsymbol{w}_{(\gamma)} | \gamma \in U\}$ som maksimerer verdien av $(corr(\boldsymbol{y}, X w_{(\gamma)}))^2$
3. Skaler $\boldsymbol{w}_{(\gamma)}$ slik at den har enhets lengde
4. Sett $\boldsymbol{w}_a = \boldsymbol{w}_{(\gamma)}$
5. Beregn den korresponderende scorevektoren $\boldsymbol{t}_a$ og ladningsvektoren $\boldsymbol{p}_a$ og q_a
6. Deflater X_{a-1} til $X_a = X_{a-1} - \boldsymbol{t}_a \boldsymbol{p}_a{}'$ og $\boldsymbol{y}_{a-1}$ til $\boldsymbol{y}_a = \boldsymbol{y}_{a-1} - \boldsymbol{t}_a q_a$.
7. Gjenta algoritmen fra punkt 1 til et ønsket antall komponenter er hentet ut.

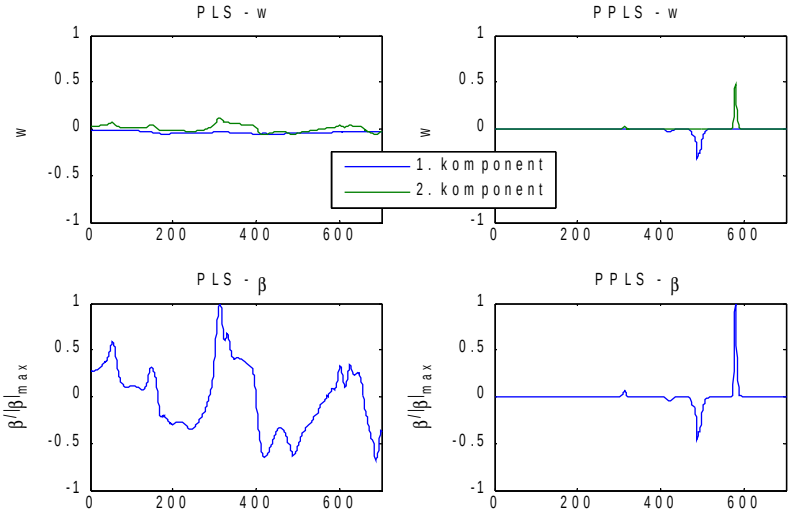
Å finne en verdi for γ som i punkt 2 maksimerer korrelasjonen er et optimeringsproblem som kan løses med en numerisk optimaliseringsalgoritme. Til dette brukes en algoritme basert på Golden section search og parabolsk interpolasjon (Press et al. 1988) som kort beskrives i kapittel 2.1 og 2.2. Disse kan effektivt finne minimumsverdien på en kontinuerlig funksjon over et åpent intervall. Ved å søke etter minimumsverdien av $-(corr(\boldsymbol{y}, X w_{(\gamma)}))^2$ på intervallet (0,1), eller et annet ønsket intervall hvis det er satt begrensninger på U, kan man finne verdien av γ som maksimerer korrelasjonen mellom responsen og scorevektoren. Da Golden section search ikke tar med endepunktene i intervallet, bør vi sammenligne $\boldsymbol{w}(\gamma)$ for den optimale verdien av γ med resultatene som oppnås for den øvre og nedre grensen ($\gamma = max(U)$ og $\gamma = min(U)$). Dette tilsvarer ren variabelseleksjon på forklaringsvariabelen med høyest standardavvik og høyest korrelasjon når $U = [0,1]$.

Noe som favoriserer PPLS fremfor PLS, er at PPLS ofte gir like gode eller bedre prediksjoner

og i tillegg foreslår enklere modeller fokusert mot færre forklaringsvariable. Dette gir oss mer tolkbare modeller, og med multivariate datasett med mange forklaringsvariable er det en fordel å få innsikt i hvilke variable som virker viktige for å kunne forklare variasjon i responsen.

1.6 Regresjonskoeffisienter og vektingsvektoren

De tre gruppene med regresjonsteknikker vi har beskrevet over har ulike tilnærminger til hvordan de kan produsere modeller som egner seg bedre enn i vanlig OLS regresjon. Variabelseleksjonsmetodene fokuserer på å inkludere viktige forklaringsvariable i modellen, og kaste ut de som ikke bidrar til god prediksjon. Ekvivalent kan vi si at variabelseleksjonsmetodene setter regresjonskoeffisienter til forklaringsvariablene som ekskluderes fra modeller til 0. Krympingsmetodene krymper regresjonskoeffisientene, og lasso og elastic net har i tillegg mulighet for å sette regresjonskoeffisienter lik 0. PLS-metodene og PCR derimot opererer med vektingsvektorer, og velger disse for å lage komponenter som det gjøres regresjon på. Det kan virke forvirrende for leseren at det snakkes om vektingsvektorer for PCR og PLS-metodene og regresjonskoeffisienter for variabelseleksjons- og krympings-metodene. Det kan imidlertid vises at regresjonskoeffisientene er avhengige av vektingsvektorene. Dette kan vi få et innblikk i ved å se studere hvordan hver av disse ser ut. Figur 1.7 viser hvordan regresjonskoeffisientene er avhengige av vektingsvektorene, og alle utslag i regresjonskoeffisienter kan spores tilbake til utslag i vektingsvektoren.



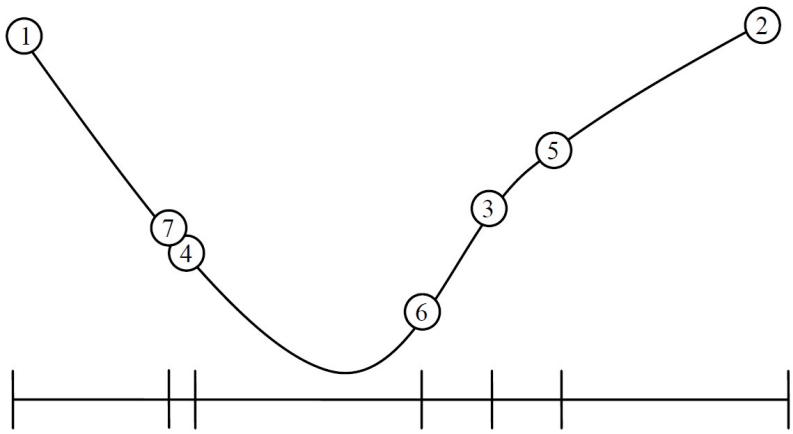
Figur 1.7: *Illustrasjon av forholdet mellom vektingsvektorene og regresjonkoeffisientene. Her avbildes de to første vektingsvektorene for PLS og PPLS, og regresjonskoeffisientene i den tilhørende modellen.*

2 Hjelpemetoder

2.1 Golden section search

Vi velger her å gå gjennom algoritmen som blir brukt i PPLS-algoritmen og som vi vil bruke i metoden vi utvikler. Dette gjøres for å vise at algoritmen bygger på et teoretisk og intuitivt fundament, og ikke kun er en tilfeldig valgt MATLAB-funksjon som har vist seg å fungere tilsynelatende tilfredsstillende for oss på de oppgavene vi har satt den til å utføre. Det er også viktig å få frem at vi ikke er garantert at metoden alltid fungerer optimalt.

Anta at problemet er å finne nullpunktet til en funksjon med én variabel over et område. Forutsatt at funksjonen er kontinuerlig og to punkter med ulikt fortegn på funksjonsverdien i det aktuelle området er kjent, garanterer dette at funksjonen har et nullpunkt. Nullpunktet kan for eksempel finnes ved å finne funksjonsverdien til et punkt midt mellom de to punktene vi har og avhengig av fortegnet til det midterste punktet, søke videre på området mellom det midterste punktet og det av punktene med motsatt fortegn på funksjonsverdien. Å halvere området i hver iterasjon maksimerer krympingen av det aktuelle område i forhold til antall iterasjoner, og etter noen iterasjoner vil man relativt raskt sirkle inn nullpunktet.
En noe mer komplisert oppgave er å finne minimumsverdien til en funksjon. En intuitiv og enkel metode som fungerer tilfredsstillende i mange tilfeller er Golden Section Search. Metoden tar utgangspunkt i at vi har en triplett med punkter (a, b, c) der funksjonsverdienverdien i punkt b er lavere enn i punktene a og c. For hver iterasjon velges et punkt, x, i det største av intervallet (a, b) og (b, c). Anta at (a, b) er det største intervallet og vi velger et punkt, x, her. Dersom verdien er lavere enn verdien i b oppdateres trippelen til (a, x, c), hvis ikke oppdateres trippelen til (x, b, c) . For hver iterasjon vil trippelen komme nærmere funksjonens minimumspunkt. Fra referansen i kapittel 2.5.3 vil man finne at nøyaktigheten på resultatet aldri kan bli bedre enn $\sqrt{\text{maskins numeriske nøyaktighet}}$, men dette holder mer enn godt nok til våre anvendelser.



Figur 2.1: *Grafisk fremstilling av hvordan Golden Section Search metoden fungerer.*

I figur 2.1 ser vi at trippelen (a, b, c) for de fire første iterasjonene med utgangspunkt i (1, 3, 2) er:

1. (1, 4, 2)
2. (1, 4, 5)
3. (1, 6, 5)
4. (7, 6, 5)

Navnet Golden Section Search kommer fra at hvert punkt, x, velges $\frac{3-\sqrt{5}}{2} \approx 0.38197$ = 38.197 % inn i det største av intervallene (a, b) eller (b, c). Dette tallet kan også spores tilbake til det gylne snitt. Det kan vises at dette er den optimale utvelgelsen av punktene x. Området som krympes, blir krympet til 0.61803 av den opprinnelige lengden. Metoden er altså ikke like effektiv som halveringsmetoden som brukes for å finne nullpunkter, man løsningen konvergerer relativt raskt allikevel. En fordel med å velge de nye punktene 38.197 % inn i det største av intervallene gjør at vi ikke risikerer å ende opp med en oscillerende løsning.

I PPLS og PLS-VS (PLS-VS omtales senere) har vi bruk for å finne maksimumsverdien til en funksjon over et område. Dette kan gjøres ved å Golden Section Search på funksjonen med motsatt fortegn. Dersom et minimumspunkt ligger i et av endepunktene klarer ikke metoden å finne disse. For å kompensere for dette vil vi spesifikt sammenligne disse med minimumspunktet for PPLS og PLS-VS for å se om disse ligger lavere.

2.2 Parabolsk interpolasjon

Golden Section Search er en metode laget for å takle de verst tenkelige og mest usamarbeidsvillige funksjonene og som steg for steg «sirkler seg inn» mot et

minimumspunktet. Er funksjonen derimot glatt og tilnærmet parabolsk nær minimumspunktet, kan et alternativ være å benytte parabolsk interpolasjon som gir en mye raskere tilnærming til minimumspunktet. Metoden tar utgangspunkt i trippelen vi bruker i Golden Section Search og lager en parabel gjennom disse. Hvis (1) den tilhørende x-verdien til bunnen av parabelen har lavere funksjonsverdi, i tillegg til (2) at det klarer seg gjennom noen tester som sørger for at vi ikke kommer i en evig løkke, velges denne x-verdien. Ved å kombinere parabolsk interpolasjon der metoden fungerer med iterasjoner med Golden Section Search, vil vi raskt kunne tilnærme oss minimumspunktet til en funksjon.

2.3 Prosessering

I dataanalyse er man interessert i å estimere tolkbare modeller som best beskriver forholdet mellom forklaringsvariablene og responsen. Dersom fenomener i datasettet i stor grad påvirker analysen kan dette gi ett galt inntrykk av forholdet. Det er derfor interessant å undersøke om prosessering av dataene kan føre til at man oppnår bedre resultater. Eksempel på fenomener er at forklaringsvariable har svært ulik varians, at forklaringsvariablene ikke er normalfordelt eller at responsen som predikeres får verdier som ikke er praktisk mulige. Vi skal nå se nærmere på noen former for prosessering av data som kan gi bedre forutsetninger for dataanalysen.

Ved å bruke de opprinnelige måleenhetene til forklaringsvariablene i datasettet, kan dette medføre at variansen til forklaringsvariablene blir svært ulik. Måler man lengden på en planke og oppgir lengden i centimeter kontra desimeter, vil variansen med de nøyaktig samme målingene være 100 ganger større for centimetermålet kontra desimetermålet. En regresjonsmetode som er sårbar for forklaringsvariablene med svært ulik varians er PLS som tenderer til å inkludere forklaringsvariable med svært høy varians i de første komponentene, uavhengig av korrelasjonen til responsen. I tilfeller der forklaringsvariablene ikke bidrar til bedre prediksjon av responsen, må PLS inkludere flere komponenter i modellen. For å unngå at forklaringsvariable med stor varians får stor innflytelse i modellen kan vi standardisere datasettene. Dette gjøres ved å skalere forklaringsvariable i datasettet slik at variansen for hver av disse blir lik.

Flere statistiske metoder forutsetter at dataene som analyseres er tilnærmet normalfordelt. Benyttes metodene på data der dette ikke er tilfelle, kan det gi modeller med dårlig prediksjon. I disse tilfellene bør man vurdere å «normalisere» dataene slik at de blir tilnærmet

normalfordelte, og dette kan gjøres med ulike transformasjonsmetoder. De vanligste transformasjonsmetodene som normaliserer data er kvadratrot- og logaritme- transformasjon, og disse benyttes på forklaringsvariable eller responsen før datasettet analyseres. En ting som er lett å glemme, men viktig å huske på, er å ta hensyn til disse transformasjonene når man tolker modellen som produseres.

Ligger det en naturlig begrensning i hvilke verdier en predikert respons kan ha, kan det ofte gi gode resultater å ta hensyn til dette kontra å bruke «rådata»-utgaven av prediksjonen. Et eksempel på dette er at en prediksjon på antall kollisjoner i et veikryss over en periode ikke kan ha en negativ verdi. Predikerer man at antall kollisjoner i et veikryss i januar måned er -2, gir det mer mening å predikere 0 kollisjoner. Et annet eksempel er at en prediksjon av sannsynlighetene for resultatet i en fotballkamp (H/U/B) bør være mellom 0 og 100% og totalt summere til 100%. I tilfeller der liknende apriori kunnskap til problemet er tilgjengelig, kan det ofte være interessant å studere om dette kan være med å forbedre modellene.

3 Validering

Undersøkelse av prediksjonsevne og tolkbarhet for ulike modeller står sentralt i denne oppgaven. Målet med en regresjonsmodell er gode prediksjoner av responsen basert på forklaringsvariablene. Målet er altså modelltilpasning som avdekker sammenhengen mellom forklaringsvariablene og responsvariabelen i datasettet.

3.1 Modellbygging

I denne oppgava har valideringen to hovedoppgaver. Den ene oppgaven er at den skal hjelpe oss med velge den potensielt beste blant flere modeller. For å være trygge på at modellen er god, behøves en validering. Hvis vi kun bygger modellen utifra det vi observerer når vi tilpasser modellen til treningsdataene får vi ofte et lite realistisk bilde for hvordan modellen vil fungere anvendt på nye data. Modellen er avhengig av treningsdataene, da disse er utgangspunktet for selve treningen av modellen. Ved å igjen bruke modellen på treningsdataene til for eksempel å bestemme hvor mange PLS-komponenter vi bør ta med i modellen, blir modellen dobbelt avhengig av treningsdataene. Ofte kan vi bli fristet til å inkludere for mange komponenter i modellen da dette gode tilpasninger til datasettet, men dette kan igjen føre til at vi velger en for optimistisk modell (overtilpasning) med dårlig prediksjonsevne når modellen blir anvendt på nye data. For å få en indikasjon på hvor mange komponenter man trenger i en modell benyttes kryssvalidering eller uavhengige testdata.

Når vi skal bedømme prediksjonsevnen til en modell trenger vi et mål for denne. Et vanlig mål på prediksjonsfeilen er det forventede kvadratavviket mellom den virkelige og den predikerte responsen, $\theta = E((\boldsymbol{y} - \hat{\boldsymbol{y}})^2)$. Dette estimeres ofte ved såkalt MSE, som er forkortelsen for mean squared error, på norsk det gjennomsnittlige kvadrerte avviket:

$$\hat{\theta}_{(k)} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_{i,(k)})^2 \quad \text{(formel 3-1)}$$

Her er k en parameter som angir kompleksiteten til modellen, *n* er antall observasjoner, y_i er den *i*-te observasjonen og $\hat{y}_{i,(k)}$ er prediksjonen av den *i*-te observasjonen med en modell med kompleksitet k . I denne oppgaven kommer vi til å sammenligne dette målet for de ulike modellene når vi skal velge modeller og vurdere predisjonsevne.

3.2 Kryssvalidering – finne parametere

Kryssvalidering [Hastie T. Et al, 2009] er kanskje den mest brukte metoden som blir brukt for å finne ut hvor godt en modell kan predikere nye testdata. Når vi jobber med datasett med relativt få observasjoner, er det ugunstig å ta ut ett sett med observasjoner og dedikere disse til validering, men med kryssvalidering får vi muligheten til dette. Metoden tar utgangspunkt i at for hver runde tar man ut ett sett med observasjoner fra datasettet og tilpasser modellen til det øvrige datasettet. Deretter anvendes modellen på settet med observasjoner som er tatt ut og vi får simulert hvordan modellen predikerer på nye observasjoner. Dette gjentas til alle observasjonene har blitt tatt ut fra datasettet. Det gjennomsnittlige avviket for prediksjonene blir et mål på hvor godt modellen kan predikere nye data.

Hvor mange og hvordan observasjonene plukkes ut i hver runde er avhengig av typen datasett vi analyserer. *Leave-one-out* er en av de enkleste metodene å plukke ut observasjoner på og den tar utgangspunkt i at én observasjon blir plukket ut av datasettet i hver runde. På datasett med relativt få observasjoner kan denne metoden gi oss ett godt inntrykk av hvordan modellene vil predikere på nye data, men metoden er beregningskrevende, da datasettet tilpasses til en modell like mange ganger som det er antall observasjoner i datasettet. Har datasettet relativt mange observasjoner blir dette nærmest en repetitiv øvelse da hver enkelt observasjon endrer datasettet i liten grad og modellene blir ofte svært like. Da kan det være av større interesse å ta ut større segmenter av datasettet i hver runde og studere hvordan modellene kan predikere på disse.

Når man skal ta ut større segmenter av datasettet i hver runde i kryssvalideringen, er en mye brukt metode å ta ut systematiske blokker. Dette kan for eksempel være å ta ut de k første observasjonene, for så å ta ut de k neste, eller alternativt å ta ut hver 5. observasjon: 1, 6, 11, 16. For store datasett vil disse segmentet representere den variasjonen datasettet inneholder i større grad enn med *leave-one-out*.

En ting man må passe på når man tar ut systematiske blokker er at treningsdatasettet ikke har en strukturer eller er stilt opp på en slik måte at observasjonene i segmentet som tas ut i hver runde blir nært homogene, men derimot er representative for den variasjonen som finnes i datasettet. For eksempel hvis observasjonene er sortert etter størrelsen på responsvariabelen, og vi tar ut segmenter med de k første observasjonene og deretter de k neste, vil segmentene representere den variasjonen som er forbundet med størrelsen på responsen (e.g. Lav, middels og høy verdi på responsvariabelen). Dette kan gi et galt inntrykk av hvordan

modellene kan predikere nye data.

For datasett med en bestemt struktur eller rekkefølge på observasjonene kan det være aktuelt å bruke den siste typen av kryssvalideringsmetoder med segmenter vi skal nevne, nemlig tilfeldig utplukking. Her fordeles observasjonene tilfeldig ut i ett gitt antall segmentet, og i hver runde tas ett segment ut som modellen som blir laget predikerer på. I motsetning til de øvrige kryssvalideringsmetodene kan denne metoden kjøres gjentatte ganger. Da fordeles observasjonene tilfeldig ut i segmentene på nytt for å simulere ett nytt datasett.

Hvilken av de ovennevnte strategiene som benyttes bør vurderes opp imot de datasettene man har. Valget blir da om man skal ta ut enkeltobservasjoner eller segmenter på grunnlag av antall observasjoner i datasettet, og i det siste tilfellet hvordan observasjonene skal plukkes ut til segmentene. Kryssvalidering kan som nevnt brukes til å validere modeller, men den kan også fungere som verktøy til å finne «optimale» parameterverdier for modellene våre.

3.3 Modellutvelgelse

En vanlig strategi for å velge ut antall komponenter som skal inngå i en modell for PLS eller en lignende metode, er å velge den som gir lavest prediksjonsfeil i kryssvalidering. En ulempe med dette er at i enkelte tilfeller leder valget til for komplekse modeller. For at dette skal unngås finnes ett alternativ for hvordan man kan analysere om det finnes en enklere modell som er tilnærmet like god som den vi får med lavest feil i kryssvalideringen. Dette er i tråd med prinsippet om at dersom vi har to forklaringer som er like gode, velge den enkleste (Occam's razor).

En forenklingsstrategi vi skal se nærmere på går ut på å undersøke om det finnes en enklere modell som ikke er signifikant forskjellig fra den som gir minste feil i kryssvalideringen (Indahl 2005). Den minste gjennomsnittlige kvadrerte feilen vi finner med kryssvalidering kalles $\mathrm{MSECV}_{\min}$, og MSECV_i betegner kryssvalideringsfeilen for modellen med i komponenter. Ved å anta at feilen for modellen med lavest kryssvalideringsfeil er en tilfeldig variabel som estimeres forventningsrett og at residualene til de ulike modellene er normalfordelt, er det fornuftig å anta at $n \cdot MSECV_{min} / \sigma^2$ følger en χ^2_n-fordeling (her er n antall frihetsgrader, som tilsvarer antall observasjoner i datasettet som ble brukt i modelleringen). Med dette som utganspunkt kan det lages et $(1-\alpha)$ % konfidensintervall for prediksjonsfeilen $\mathrm{MSECV}_{\min}$ for et gitt signifikansnivå α. Er vi interessert i å velge

en enklere modell dersom den finnes, kan vi undersøke om prediksjonsfeilen til modellene med færre komponenter er innenfor konfidensintervallet for $\mathrm{MSECV}_{\min}$, og eventuelt velge den av de som har prediksjonsfeil innenfor med færrest komponenter. Dette kan i enkelte tilfeller gi et mer realistisk bilde av hvor mange komponenter som behøves i en modell.

3.4 Testsettet – estimere prediksjonsfeil

Den andre oppgaven valideringa har er å hjelpe oss med sammenligning av konkurrerende modeller. Til dette brukes enten kryssvalidering eller et uavhengig testsett. På denne måten fåes et inntrykk av prediksjonsevnen til hver modell på nye data. Ved sammenliging av prediksjonsnøyaktigheten i forhold til modellkompleksitet kan dette gi ett inntrykk av stabiliteten til hver metode. Denne delen av valideringa kan gi et bilde på for hvilke typer data de ulike modellene fungerer godt i forhold til hverandre.

Bruk av testsett er kanskje den enkleste (og mest intuitive) metoden for å finne ut hvor godt en modell kan predikere nye data. Dersom man ikke har et treningssett og et testsett separert, kan man dele datasettet i to og bruke den ene delen til å bygge modellen, og den øvrige delen til å estimere prediksjonsnøyaktigheten på. Når testdataene aldri har vært med i modellbyggingen vil modellen være uavhengig av disse, og estimatet på prediksjonsfeilen vil være forventningsrett. Hvor store del av datasettet som bør brukes til treningsett og hvor stor del som bør dedikeres til testsett finnes det ikke noe nøyaktig svar på. Det vanligste er å dele datasettet prosentvis et sted mellom 50 – 50 og 80 – 20 for henholdsvis trening- og testsettet. Her er det opp til den som gjør analysen hvilken fordeling som skal benyttes. Ønsker man å bruke relativt mange observasjoner for å bygge stabile modeller går dette på bekostning av et mer usikkert estimat av prediksjonsfeilen og visa verca. Man må da finne et kompromiss mellom hvor nøyaktig man ønsker å estimere prediksjonsfeilen og hvor mange observasjoner man ønsker å bruke i modellbyggingen for hvordan fordelingen bør være.

4 Originalbidrag i masteroppgaven: PLS-VS

4.1 Overordnet motivasjon for metodeutviklingen

Med dagens måleteknologi og innsamlingsmuligheter har vi muligheten til å få datasett med svært mange og høyt korrelerte forklaringsvariable. For å kunne få oversikt over hvilke forklaringsvariable som er de «viktigste» for å kunne forklare variasjon i responsen, er det behov for metoder som produserer enkle modeller og utfører variabelseleksjon. Metoder som effektivt er i stand til å velge ut og lage enkle modeller basert på de viktigste forklaringsvariablene er derfor å foretrekke.

Som nevnt innledningsvis i teksten møter modellering basert på OLS problemer når datasettene vi jobber med har flere variabler enn observasjoner og/eller når variablene er høyt korrelerte. Typiske problemer er at OLS gir svært ustabile modeller som ofte også predikerer dårlig. Ulike nevnte metoder for håndtering av disse problemene er:

1. Variabelseleksjonsmetoder som selekterer forklaringsvariable til et mindre datasett som vi utfører OLS på.
2. Krympingsmetoder som krymper regresjonskoeffisienter og enkelte av disse har også implementert variabelseleksjon.
3. Projeksjonsmetoder som bygger opp et begrenset antall komponenter med lineærkombinasjoner av de originale forklaringsvariablene, og deretter anvender OLS på komponentene. Til den siste typen regresjonsmetodikk hører blant annet metodene PCR, PLS og PPLS til.

Psom nevnt kan variable med høy varians dominere de første PLS komponentene. For å unngå dette problemet er det mulig å splitte opp kovariansen i en korrelasjonsdel og en standardavviksdel og differensiere vektingen mellom disse. Dette gjøres i PPLS ved at hver av delene potenseres med en justerbar parameter som kan gi enkle modeller med spissing mot enkeltvariable, eller mer robuste modeller hvor flere variabler er med for å stabilisere modellen. Muligheten til å velge parametere som gir ulike egenskaper til modellene gjør metoden fleksibel og eksplorativ.

Egenskaper fra PLS og variabelseleksjonsmetodikk danner grunnlaget for å motivere metoden vi utvikler og presenterer i denne oppgaven. I den praktiske løsningen utnyttes i tillegg noen enkle idéer fra teori som omhandler rette linjer i 2 dimensjoner.

Resultatet er en PLS-beslektet metode som foretar en oppsplitting av kovariansen til en korrelasjonsdel og en standardavviksdel (som i PPLS) og som differensierer vektingen på hver av disse ved hjelp av en justerbar parameter. Metoden kan begrenses til å lage enkle modeller som fokusere på viktige forklaringsvariable på samme måte som PPLS, og i tillegg selektere vekk mindre viktige forklaringsvariable. Metoden kan også begrenses til å måtte inkludere flere korrelerte forklaringsvariable i en modell for å oppnå mer stabilitet rundt prediksjonsegenskapene og robusthet i modellbyggingen. Ulike avgrensninger som settes på parameterne gjør metoden fleksibel, og ved å utforske slike avgrensninger kan metoden benyttes eksplorativt på lik linje med PPLS.

4.2 Teknisk motivasjon

Da vår metode er motivert fra PLS og PPLS, er det naturlig og gå inn å se nærmere på disse. Vi vil først ta utgangspunkt i vektingsvektoren i PLS og omforme denne, for deretter å utlede hvordan vektingsvektoren for PPLS ser ut. Deretter vil vi vise hvordan vårt forslag til vektingsvektor ser ut og relatere det til det vi kjenner fra de to andre metodene.

Det er velkjent at vektingsvektoren i PLS i hver iterasjon i algoritmen kan skrives som:

$$\boldsymbol{w}' = k \cdot \left[cov(\boldsymbol{y}, \boldsymbol{x}_1), \ldots, cov(\boldsymbol{y}, \boldsymbol{x}_p) \right] \qquad \text{(formel 4-1)}$$

der k er en skalar som garanterer at vektoren har enhets lengde, $\boldsymbol{x}_i$ er den i-te vektoren til forklaringsvariabel i, og $\boldsymbol{x}_i$ og $\boldsymbol{y}$ er de deflaterte versjonene av henholdsvis X matrisa og responsvektoren $\boldsymbol{y}$ (med unntak av i første iterasjon, da disse er de sentrerte utgavene av X matrisa og responsen). Dersom kovariansene faktoriseres som korrelasjonen mellom hver variabel og responsen multiplisert med standardavviket til den gitte variabelen og med standardavviket til responsen i hvert element gjelder følgende omskriving:

$$\boldsymbol{w}' = k \cdot \left[corr(\boldsymbol{y}, \boldsymbol{x}_1) \cdot std(\boldsymbol{x})_1, \ldots, corr(\boldsymbol{y}, \boldsymbol{x}_p) \cdot std(\boldsymbol{x})_p \right] \qquad \text{(formel 4-2)}$$

Legg merke til at standardavviket til responsen er faktorisert i hvert element av vektingsvektoren, så den lar seg trekke ut og inngå i konstanten k . I tillegg skaleres hele vektoren slik at den maksimale absoluttverdien til standardavviket og korrelasjonen blir lik 1. Skaleringstallet som sikrer dette innlemmes i konstanten k . Til slutt faktoriseres

korrelasjonen for hver variabel x_i til produktet av absoluttverdien til korrelasjonen og en konstant, s_i , med absoluttverdi lik 1 og samme fortegn som korrelasjonen:

$$\boldsymbol{w}' = k \cdot \left[s_1 \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)| \cdot std(\boldsymbol{x}_1), \ldots, s_p \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)| \cdot std(\boldsymbol{x}_p) \right] \quad \text{(formel 4-3)}$$

Formel 4-3 er en ekvivalent måte å formulere vektingsvektoren i PLS (formel 4-1) på. Grunnen til at vi formulerer vektingsvektoren på denne tilsynelatende kompliserte måten, er at den danner utgangspunktet for bestemmelse av vektingsvektorene både i PPLS og i den nye metoden vi presenterer.

I PPLS benyttes en parameter, γ som tar verdier i $U = [0,1]$, og som differensierer vektingen mellom korrelasjonen til responsen og standardavviket for hver variabel. Formelen for vektingsvektoren i PPLS er:

$$\boldsymbol{w}_{(\gamma)}' = K_\gamma \cdot \left[s_1 \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)|^{\gamma/(1-\gamma)} \cdot std(\boldsymbol{x}_1)^{(1-\gamma)/\gamma}, \ldots, s_p \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)|^{\gamma/(1-\gamma)} \cdot std(\boldsymbol{x}_p)^{(1-\gamma)/\gamma} \right]$$

(formel 4-4)

der K_γ er en skalar som garanterer at vektoren har enhets lengde. En parameterverdi på γ nær 1 eller nær 0 tilsvarer høy vekting på henholdsvis korrelerte variabler og variabler med høyt standardavvik og spissing mot disse i vektingsvektoren. En parameterverdi på γ lik 0.5 vekter korrelasjonen og standardavviket likt, og vi får derfor en vekting som er identisk med PLS:

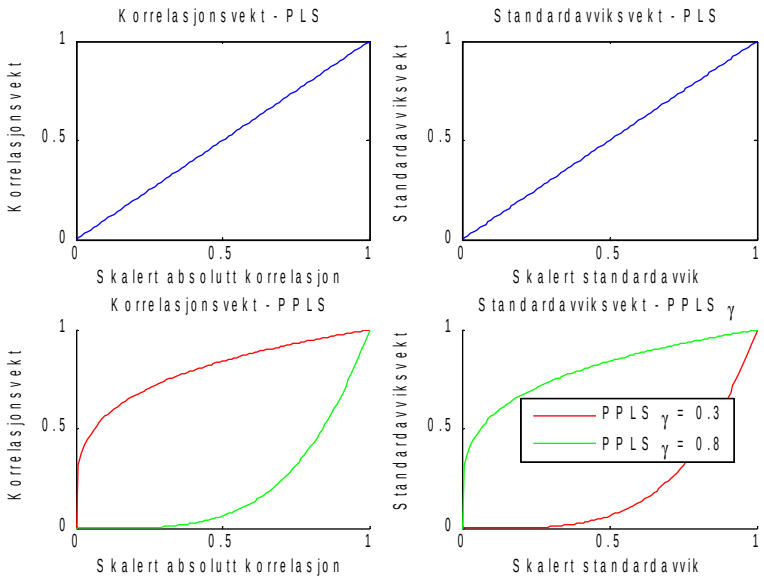
$$\begin{aligned} \boldsymbol{w}_{(0.5)}' &= K_{0.5} \cdot \left[s_1 \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)|^{0.5/0.5} \cdot std(\boldsymbol{x}_1)^{0.5/0.5}, \ldots, s_p \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)|^{0.5/0.5} \cdot std(\boldsymbol{x}_p)^{0.5/0.5} \right] \\ &= K_{0.5} \cdot \left[s_1 \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)| \cdot std(\boldsymbol{x}_1), \ldots, s_p \cdot |corr(\boldsymbol{y}, \boldsymbol{x}_1)| \cdot std(\boldsymbol{x}_p) \right] \\ &= K_{0.5} \cdot \propto \left[cov(\boldsymbol{y}, \boldsymbol{x}_1), \ldots, cov(\boldsymbol{y}, \boldsymbol{x}_p) \right] \end{aligned}$$

I dette uttrykket er hvert element i vektingsvektoren er proporsjonalt med kovariansen, og når vektingsvektoren skaleres slik at den får enhets lengde blir vektingsvektoren for PPLS for $\gamma = 0.5$ identisk med vektingsvektoren i PLS.

For å få et inntrykk av hvordan korrelasjonsdelen og standardavviksdelen vektes i PPLS sammenlignet med PLS kan vi ta utgangspunkt i formel 4-3 for å beskrive vektingsvektoren i PLS, og formel 4-4 for å beskrive vektingsvektoren i PPLS. Ved å plotte verdier på

korrelasjonen og standardavviket til hver variabel langs x-aksen, og den tilhørende transformerte verdien på korrelasjonen og standardavviket på y-aksen for PLS og PPLS kan vi illustrere konsekvensene av å benytte γ lik 0.8 og lik 0.3 som gir vektingsvektorer med fokus rettet mot henholdsvis høyt korrelerte variable og variable med høyt standardavvik. Vektingen i vektingsvektoren, $\hat{\boldsymbol{w}}$, for hver forklaringsvariabel er videre proporsjonal med produktet av den transformerte verdien av korrelasjonen og standardavviket.

For resten av utledningen vil jeg definere terminologi som forenkler beskrivelsen av metodene. Den transformerte verdien for korrelasjonen og standardavviket som tilsvarer verdien på y-aksen for hver av metodene kalles nå for korrelasjonsvekten og standardavviksvekten. Korrelasjonsvekten for en forklaringsvariabel x_i blir i PPLS dermed $|corr(\boldsymbol{y}, \boldsymbol{x}_i)|^{\gamma/1-\gamma}$, der $\boldsymbol{y}$ og $\boldsymbol{x}_i$ er de deflaterte utgavene av responsen og forklaringsvariabelen. I tillegg bruker vi terminologien korrelasjon om den absolutte korrelasjonen, da fortegnet antas å være faktorisert ut, og korrelasjonen og standardavviket antas å være skalert slik at maksimalverdien for hver av disse er lik 1.

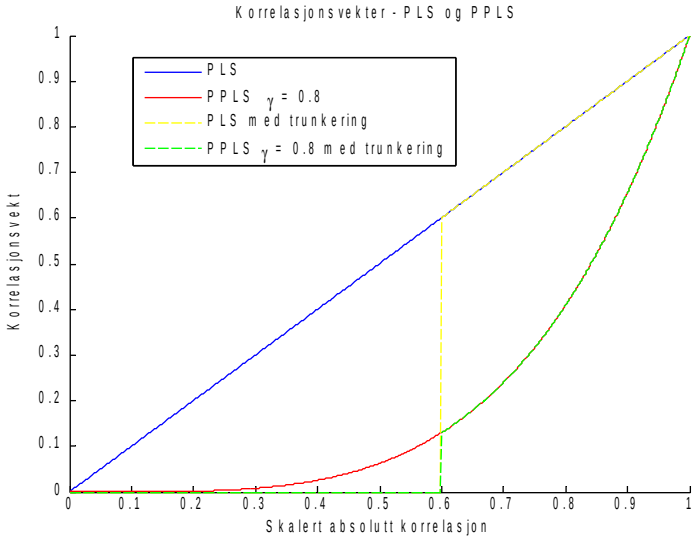


Figur 4.1: *Illustrasjon av beregning av korrelasjonsvekter og standardavviksvekter for PLS og PPLS.*

Som vist i figur 4.1 er korrelasjonsvekten og standardavviksvekten i et en-til-en forhold med korrelasjonen og standardavviket i vanlig PLS. For PPLS derimot (illustrert for valget $\gamma = 0.8$ tilsvarende den grønne kurven) foregår derimot en differensiert vekting mellom høyt og lavt korrelerte variable. Forskjellen i korrelasjonsvekt for en relativt høyt korrelert variabel og en relativt lavt korrelert variabel er mye større for PPLS enn for PLS. Forskjellen i standardavviksvektene derimot er mye mindre for PPLS enn i PLS for forklaringsvariable

med relativt høyt og relativ lavt standardavvik. Når forskjellene i standardavviksvekt blir liten for forklaringsvariablene får vi en standardiseringseffekt av variablene, og vi står igjen med at korrelasjonsvektene får størst innflytelse på vektingsvektorene. Det blir derfor som vi kjenner fra tidligere at med $\gamma > 0.5$ så fokuseres modellene i retning av mer vektleggelse av korrelerte variable. Dette kan bidra til å oppnå enklere og mer tolkbare modeller, og er en viktig fasett vi ønsker å inkludere i den nye metoden.

I de ulike formene for variabelseleksjonsmetodikk er poenget at forklaringsvariablene som i en viss forstand identifiseres som de «viktigste» for å kunne predikere responsen blir inkludert i modellen, mens de som er mindre viktige utelates. De forklaringsvariablene som utelates får i praksis regresjonskoeffisienter lik 0. Ved å trekke en parallell til korrelasjonsvekter og standardavviksvekter i PPLS, kan et alternativ her være å trunkere vektene til forklaringsvariablene som er lavt vektet til verdien 0. Ser vi på korrelasjonsvektene i PPLS med $\gamma = 0.8$, kan vi for eksempel kreve at variable med korrelasjoner lavere enn en gitt verdi $\Delta_{korr} = 0.6$ ansees for å være uviktige nok til å 0-trunkeres, da disse allerede vektes relativt lavt. En tilsvarende trunkering kan også selvsagt utføres på korrelasjonsvektene i PLS fra formel 4-3, men resultatet blir mer brutalt da endringene for disse vektene blir vesentlig større. Da vi ønsker å fasetten fra PPLS med å kunne fokusere modellene mot enkeltvariable, er det denne vi fokuserer på i utgangspunktet. Men vi viser også hvordan resultatet av trunkering på korrelasjonsvektene blir for PLS, for å tydeliggjøre at dette også kan utføres med denne metoden. I figur 4.2 illustreres trunkering av korrelasjonsvekter for PLS og PPLS:

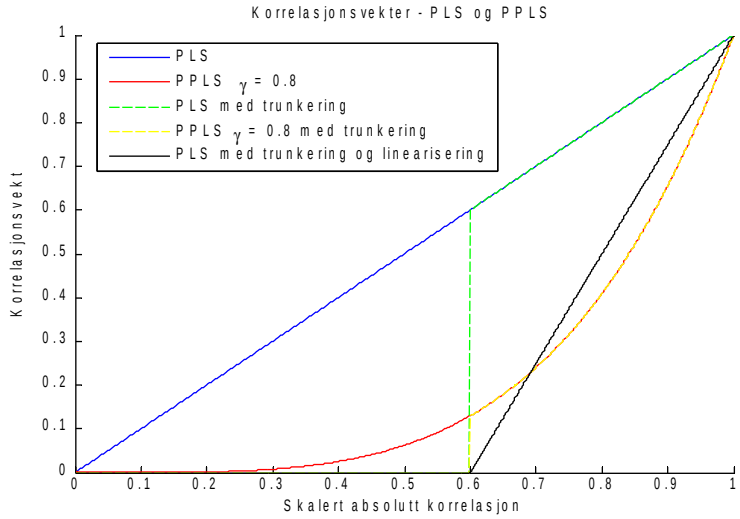


Figur 4.2: *Illustrasjon av trunkering på PLS og PPLS korrelasjonsvekter.*

Når variable med korrelasjon lavere enn $\Delta_{korr}=0.6$ får sine korrelasjonsvekter trunkert til 0, ser vi av figur 4.2 at dette medfører betydelige endringer for korrelasjonsvektene i PLS som gjør et stort hopp når korrelasjonen passerer grenseverdien 0.6. Når variable med korrelasjon lavere enn Δ_{korr} = 0.6 trunkeres til 0 i PPLS, ser vi derimot at dette avviker i mindre grad fra de opprinnelige korrelasjonsvektene. Å sette små korrelasjonsvekter lik 0 kan derfor være en fornuftig forenkling, og en fordel med dette er at vi i tillegg oppnår en ønsket variabelseleksjonseffekt. I og med at vektingen i vektingsvektoren tilsvarer produktet av korrelasjonsvekten og standardavviksvekten, vil alle forklaringsvariable med korrelasjoner lavere enn $\Delta_{korr}=0.6$ vektes bort (med 0-er) i vektingsvektoren. Ved å implementere disse idéene opprettholdes muligheten til å finne modeller som fokuserer mot høyt korrelerte variable (som PPLS) samtidig som variable med lav korrelasjon valges ut av modellen ved at lave korrelasjoner settes lik 0 (variabelseleksjon). Sammen utgjør refleksjonene over grunnlaget for en variabelseleksjonsvariant av PPLS.

4.3 Beskrivelse av den nye metoden

Både for PLS og PPLS gjør kurven som beskriver korrelasjonsvekter et hopp idet korrelasjonen passerer verdien for Δ_{korr}. Et alternativ for å oppnå en mykere overgang kan være å vekte korrelasjoner større enn Δ_{korr} lineært fra 0 til 1. En illustrasjon av dette følger under:



Figur 4.3: *Illustrasjon av ulike måter å beregne korrelasjonsvekter for PLS-metoder.*

Av figur 4.3 ser vi at denne lineariseringa av korrelasjonsvektene kan fungere som en grei lineær approksimasjon av hvordan PPLS med $\gamma=0.8$ beregner korrelasjonsvektene.

Matematisk kan denne lineariseringa beskrives som en rett linje i to dimensjoner der vi kun er interessert i positive verdier på y-aksen. Stigningstallet kan bestemmes med to punkter på linja, og ved å velge $(\Delta_{korr},0)=(0.6,1)$ og $(1,1)$ fra figuren blir stigningstallet

$$a=\frac{y_2-y_1}{x_2-x_1}=\frac{1-0}{1-\Delta_{korr}}=\frac{1}{1-\Delta_{korr}} \quad .$$

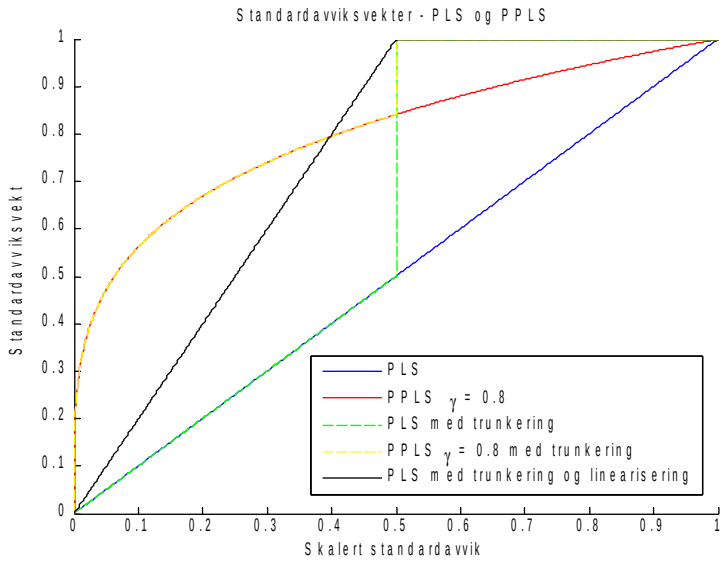
Likninga for den rette linja kan bestemmes med formelen $y=y_1+a(x-x_1)$, og ved innsetting får vi at $y=0+\frac{1}{1-\Delta_{korr}}(x-\Delta_{korr})=\frac{(x-\Delta_{korr})}{1-\Delta_{korr}}$.

Enhver funksjon $f(x)$ kan alltid deles opp i en positiv del $f(x)^{+}=max\{f(x),0\}$ og en negativ del $f(x)^{-}=-min\{f(x),0\}$ slik at $f(x)=f(x)^{+}-f(x)^{-}$ (Rence 2012). Da vi kun er interessert i den delen av linja som gir positive verdier på y-aksen, kan vi fokusere kun på den positive delen av funksjonen. Ved å bytte ut x med korrelasjonen og y med korrelasjonsvekten i likningen for den rette linja, ser vi at korrelasjonsvekten til forklaringsvariabel x_i i den lineariserte utgaven av PPLS kan skrives som:

$$Korrelasjonsvekt_i=\frac{(|korr(\boldsymbol{y},\boldsymbol{x}_i)|-\Delta_{korr})^{+}}{(1-\Delta_{korr})} \qquad \text{(formel 4-5)}$$

for en vilkårlig $\Delta_{korr}\in[0,1)$. I tilfellet der $\Delta_{korr}=0$ gir dette de samme korrelasjonsvektene som for PLS. For $\Delta_{korr}=1$ kan ikke formelen over benyttes, men dette defineres på samme måte som i PPLS. Når $\Delta_{korr}=1$ får samtlige forklaringsvariable korrelasjonsvekt lik 0, med unntak av den (de) høyest korrelerte forklaringsvariabelen som får korrelasjonsvekt lik 1.

Vi skal nå illustrere lineærapproksimering av standardavviksvektene for PPLS med $\gamma=0.8$. I figur 4.1 ligger standardavviksvektene i PPLS nært 1 for variabler med relativt høyt standardavvik. En approksimasjon til dette kan være å trunkere standardavviksvekten til 1 for variable med standardavvik større enn for eksempel $\Delta_{std}=0.5$. På samme måte som for korrelasjonsvektene lineærtransformeres kurvene for å unngå diskontinuitet. Denne ideen er illustrert i figur 4.4.



Figur 4.4: *Illustrasjon av ulike måter å beregne standardavviksvekter for PLS og PPLS.*

Legg merke til at linearisering og trunkering av standardavviksvektene (svart kurve) kan oppfattes som en lineær approksimasjon til hvordan PPLS med $\gamma = 0.8$ beregner standardavviksvektene (rød kurve). Grafisk kan vi beskrive lineariseringa som ei rett linje mellom $(0,0)$ og $(\Delta_{std}, 1)$, skjøtet sammen med den horisontale linja $y(x) = 1$. For $x \in [\Delta_{std}, 1]$. Stigningstallet for linja mellom $(0,0)$ og $(\Delta_{std}, 1)$ blir:

$a = \frac{y_2 - y_1}{x_2 - x_1} = \frac{1-0}{\Delta_{std} - 0} = \frac{1}{\Delta_{std}}$, så likningen for den første rette linja blir $y = \frac{x}{\Delta_{std}}$.

Sammenskjøtningen av de to linjene kan uttrykkes ved:

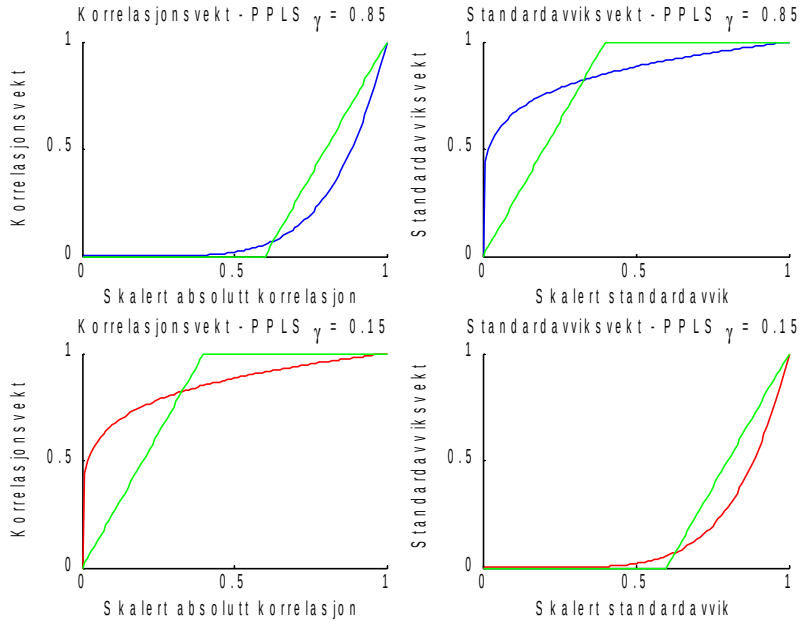
$$y(x) = \frac{x}{\Delta_{std}} - 1 + 1 = -\left(\frac{x}{\Delta_{std}} - 1\right)^{-} + 1 \quad \text{for} \quad x \in (0,1] \ .$$

Ved å bytte ut x med standardavviket og y med standardavviksvekten, ser vi at standardavviksvekten til forklaringsvariabel x_i i den lineariserte utgaven av PPLS kan skrives som:

$$Standardavviksvekt_i = -\left(\frac{std(x_i) - 1}{\Delta_{std}}\right)^{-} + 1 \quad \text{(formel 4-6)}$$

for en valgt $\Delta_{std} \in (0,1]$. I tilfellet der $\Delta_{std} = 1$ gir dette samme standardavviksvekting som for PLS. For $\Delta_{std} = 0$ kan ikke formelen over brukes, og vi definerer isteden at for samtlige forklaringsvariable settes standardavviksvekten da lik 1.

Likningene beskrevet over fungerer for å linearisere PPLS når $\gamma \geq 0.5$, men de fungerer ikke uten videre når $\gamma < 0.5$. Det er imidlertid mulig å benytte seg av en sammenheng mellom korrelasjonsvektingen og standardavviksvekten for håndtering av tilfellene med $\gamma < 0.5$. Det er ikke vanskelig å innse at korrelasjonsvektene i PPLS for en vilkårlig verdi av $\gamma = \gamma_1$ blir beregnet på eksakt samme måten som standardavviksvektene i PPLS når $\gamma = \gamma_2 = 1 - \gamma_1$. Illustrasjonen i figur 4.5 viser hvordan:



Figur 4.5: *Illustrasjon av sammenhengen mellom hvordan korrelasjon- og standardavviksdelen blir vektet i PPLS når vi bruker* γ *og* $1-\gamma$.

Ved å utnytte denne sammenhengen kan vi fortsatt bruke de samme likningene for linearisering PPLS når $\gamma < 0.5$, bare i motsatt rekkefølge. I linearisering av PPLS når $\gamma < 0.5$ brukes formel 4-5 med standarddavviket som input for å finne standardavviksvektene og formel 4-6 med korrelasjonen som input for å finne korrelasjonsvektene.

Vi må derfor forholde oss til to stykkevise lineariseringsvarianter av korrelasjons- og standardavviks-vektene i PPLS:

1) trunkere korrelasjonsvektene til 0 og standardavviksvektene til 1
2) trunkere standardavviksvektene til 0 og korrelasjonsvektene til 1

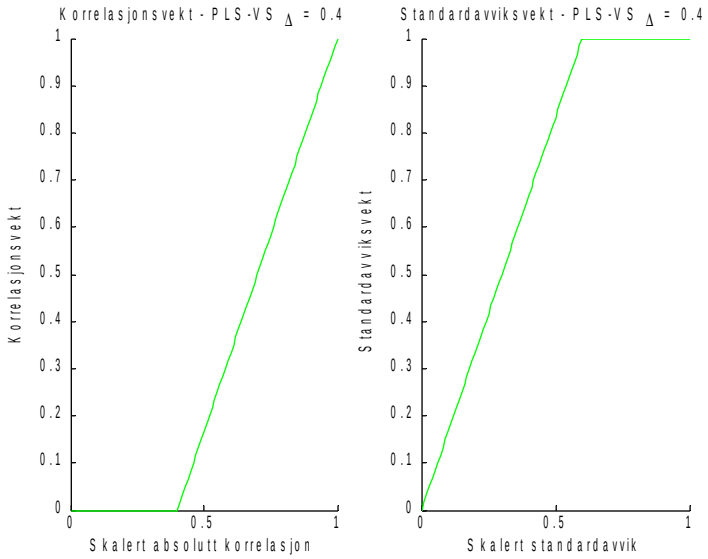
Det å benytte to justerbare parametere åpner for stor fleksibilitet i metoden, men på den annen side betyr det også at metoden blir mer beregningskrevende (i og med at en ekstra parameter skal optimaliseres). For å unngå dette foreslår vi i stedet en reparametrisert restriksjon av

trunkeringsparameterene slik at optimaliseringen begrenses til kun én parameter.

Ved å innføre parameteren $\Delta \in U = [-1,1]$ vil tilfellene med $\Delta \geq 0$ henspeile på situasjonen der korrelasjonsvektene trunkeres til 0, opp til trunkeringsgrensa $x = \Delta$. Korrelasjoner mellom Δ og 1 omregnes til korrelasjonsvekter i et lineært stigende forhold fra 0 til 1. Vi forlanger også som restriksjon at $(1-\Delta)$ definerer en nedre trunkeringsgrense for standardavvikene der standardavviksvektene trunkeres til 1. Standardavvik mellom 0 og $(1-\Delta)$ får standardavviksvekt lineært stigende fra 0 og 1. Spesialtilfellet $\Delta = 0$ resulterer i et sammenfall med vektingsvektoren for vanlig PLS.

I det motsatte tilfellet, når $\Delta < 0$, vil standardavviksvektene trunkeres til 0. $|\Delta|$ angir i dette tilfellet den øvre grensen for størrelsen på standardavviket hvor standardavviksvekten trunkeres til 0. Standardavviksverdier som ligger mellom $|\Delta|$ og 1 gis en standardavviksvekt lineært stigende fra 0 til 1. Ved at $(1-|\Delta|)$ angir nedre trunkeringsgrense for korrelasjoner som får sine korrelasjonsvekt trunkert til 1, og korrelasjoner mellom 0 og $(1-|\Delta|)$ gis korrelasjonsvekter som er lineært stigende fra 0 til 1 er beregningsmåten av samtlige valg av Δ parameteren beskrevet.

Som et eksempel på hvordan korrelasjonsvektene og standardavviksvektene blir, har vi valgt å beskrive tilfellet der trunkeringsparameteren $\Delta = 0.4$. Her får forklaringsvariable med korrelasjon mindre enn 0.4 får korrelasjonsvekt lik 0, og forklaringvariable med korrelasjon mellom 0.4 og 1 får korrelasjonsvekt lineært stigende fra 0 til 1. Forklaringsvariable med standardavvik større enn 0.6 får standardavviksvekt lik 1 og forklaringsvariable med standardavvik mellom 0 og 0.6 får standardavviksvekt lineært stigende fra 0 til 1. Under følger en illustrasjon av dette:



Figur 4.6: *Illustrasjon av korrelasjonsvekter og standardavviksvekter for PLS-VS når* $\Delta = 0.4$.

Vektingsvektoren kan dermed beskrives for en gitt verdi av Δ som:

$$\boldsymbol{w}(\Delta)' = K_\Delta \, [s_1 \cdot f_j(|corr(\boldsymbol{y}, \boldsymbol{x}_1)|, |\Delta|) \cdot f_{3-j}(std(\boldsymbol{x}_1), |\Delta|), \ldots, s_p \cdot f_j(|corr(\boldsymbol{y}, \boldsymbol{x}_p)|, |\Delta|) \cdot f_{3-j}(std(\boldsymbol{x}_p), |\Delta|)] \quad \text{(formel 4-7)}$$

der $-1 < \Delta < 1$, s_i angir fortegnet på den i-te korrelasjonen og K_Δ er en skaleringskonstant som skalerer vektingsvektoren til enhets lengde, og $f(x, |\Delta|)$ er funksjonen som lineærtransformerer korrelasjonen og standardavviket. Her har vi at $j = 1$ hvis $\Delta \geq 0$ og $j = 2$ hvis $\Delta < 0$ og funksjonene som utfører lineærtransformasjonene er:

$$f_1(x, \Delta) = \frac{(x - \Delta)^+}{(1 - \Delta)} \quad \text{og} \quad f_2(x, \Delta) = \left(\frac{x}{1 - \Delta} - 1 \right)^- + 1 \quad .$$

For $\Delta = -1$ og $\Delta = 1$ kan formlene for lineær transformasjon ikke benyttes, da disse gir verdien 0 i nevneren i funksjonen $f_1(x, \Delta)$ og $f_2(x, \Delta)$. Disse tilfellene representerer henholdsvis ren variabelseleksjon av den (eller de) forklaringsvariabelen med høyest standardavvik og høyest korrelasjon til responsen. For $\Delta = -1$ blir vektingsvektoren en vektor med 0-er for alle forklaringsvariablene, og en 1-er for forklaringsvariabelen med høyest standardavvik. For $\Delta = 1$ blir vektingsvektoren en vektor med 0-er for alle forklaringsvariablene, og en 1-er for forklaringsvariabelen med høyest korrelasjon. (Dersom det er m forklaringsvariable med høyest korrelasjon eller standardavvik blir disse vektet med $1/\sqrt{m}$).

Forslaget beskrevet over gir opphav til en alternativ beregningsmåte av vektingsvektorer i en PLS-beslektet innpakning. Det blir dermed enkelt å fokusere modellen mot et mindre antall

forklaringsvariable med hensyn på høy korrelasjon eller høyt standardavvik på lik linje med PPLS avhengig av hvordan Δ varierer. Som optimale parameterverdier for Δ i hver komponent velges parameterverdien som maksimerer den absolutte korrelasjonen mellom den a-te scorevektoren, $\boldsymbol{t}_a = X_{a-1}\boldsymbol{w}_a = X_{a-1}\boldsymbol{w}_a(\Delta)$, og den deflaterte responsen, $\boldsymbol{y}_{a-1}$. Dette kan også sees på som den parameterverdien av Δ i $U = [-1,1]$ (eller et mindre avgrenset parameterområde) som minimerer uttrykket $-(corr(\boldsymbol{y}_{a-1}, X_{a-1}\boldsymbol{w}(\Delta)))^2$. Kvadreringa av uttrykket er gjort for å kun få positive verdier på korrelasjonen, og å unngå å få med knekkpunkter på grafen som vi kunne fått ved bruk av absoluttverdi. For hvert av de to trunkeringstilfellene er dette et minimeringsproblem av en funksjon med én variabel som kan løses via en algoritme basert på Golden section search og parabolsk interpolasjon (den samme algoritmen som brukes for å finne optimale parameterverdier for γ i PPLS). Når vi har funnet en verdi som maksimerer korrelasjonen mellom scorevektoren og responsen i hvert av tilfellene $\Delta < 0$ og $\Delta \geq 0$, velges den av de to med høyest korrelasjon, og vi beregne den korresponderende vektingsvektoren til parameteren. Deretter beregnes scorevektoren og vi deflaterer X matrisa og $\boldsymbol{y}$ responsen på samme måte som i PLS algoritmen i hver iterasjon.

4.4 Oppsummering (PLS-VS)

Metoden PLS-VS er basert på en stykkevis lineærtransformert versjon av måten som korrelasjonsvektene og standardavviksvektene i PPLS blir beregnet på. PLS-VS med trunkeringsparameter $\Delta \in [-1,1]$ kan derfor oppfattes som en lineærtilnærming av PPLS med potenseringsparameter $\gamma \in [0,1]$. PLS-VS er fleksibel på samme måte som PPLS, ved at man med ulike parameterverdier Δ kan fokusere på forklaringsvariable med høy korrelasjon til responsen eller høyt standardavvik i score-vektorene. Det er også mulig å utforske ulike avgrensninger på parameterne slik at metoden lar seg utnytte eksplorativt i retning av for eksempel høy grad av bortvalgte variable (modellering med få selekterte forklaringsvariable).

Nødvendigheten av å betjene to funksjoner for stykkevis lineærtransformering av standardavviks- og korrelasjonsvektene med aktuelle trunkeringstyper gjør at beregningen av vektingsvektorene todelt. Begrensningene som settes på parameterverdiene avgjør hvilke av funksjonene som skal benyttes for å beregne standardavviks- og korrelasjonsvektene. De tre mulighetene for trunkering er:

1. Den aktuelle parametermengden U inneholder både positive og negative verdier for: Både trunkering av korrelasjonsvekter til 0 og standardavviksvekter til 1, og

trunkering av standardavviksvekter til 0 og korrelasjonsvekter til 1.

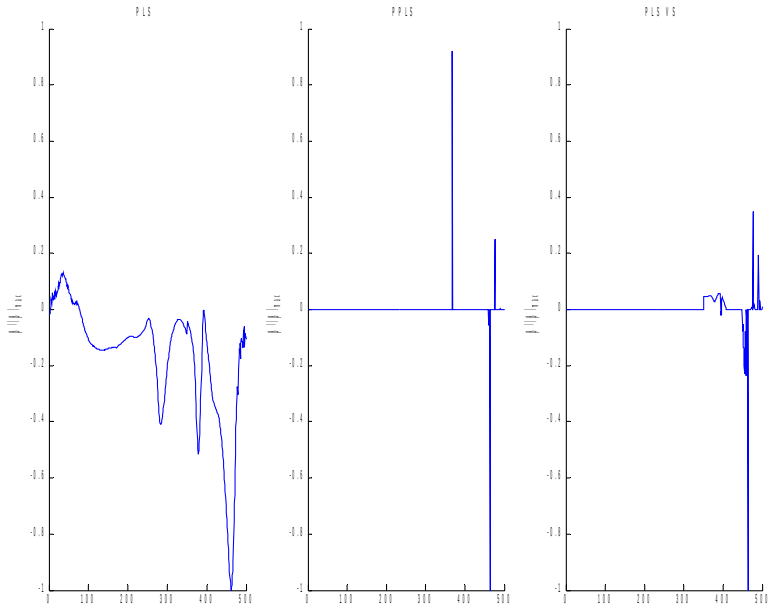
2. Den aktuelle parametermengden U inneholder bare positive verdier for Δ : Kun trunkering av korrelasjonsvekter til 0 og standardavviksvekter til 1.
3. Den aktuelle parametermengden U inneholder bare negative verdier for Δ : Kun trunkering av standardavviksvekter til 0 og korrelasjonsvekter til 1.

4.5 Implementasjonsskisse (MATLAB)

I algoritmen som implementerer metoden er det praktisk å dele beregningene opp i flere funksjoner som vi her velger å omtale som *weights*, *correlations* og *w_calc*. Funksjonen *weights* foretar faktoriseringen av kovariansen mellom responsen og hver forklaringsvariabel til den absolutte korrelasjonen, fortegnet på korrelasjonen og standardavviket, for deretter å skalere korrelasjonen og standardavviket slik at de har maksimalverdi lik 1. Deretter identifiseres hvilket av de tre trunkeringstilfellene vi har, for å bestemme hvilke intervaller av verdier som skal analyseres. Korrelasjonen mellom lineærkombinasjonen $X\boldsymbol{w}(\Delta)$ og responsen $\boldsymbol{y}$ for en gitt Δ parameter beregnes i funksjonen *correlations*, og for det eller de aktuelle intervallene utføres optimeringen av denne funksjonen. Når den optimale verdien for Δ er funnet for de aktuelle intervallene, må $\boldsymbol{w}(\Delta)$ sammenlignes med resultatene som oppnåes for endepunktene av hvert intervall, da optimaliseringen basert på Golden Section Search og parabolsk interpolasjon ikke inkluderer endepunktene. Dersom det aktuelle intervallet er $U=[-0.9,1]$ undersøkes endepunktene ved spesifikt å beregne korrelasjonen mellom $X\boldsymbol{w}(\Delta)$ og $\boldsymbol{y}$ for Δ lik de tre endepunktene $\{-0.9,0,1\}$. Når verdien av Δ som maksimerer korrelasjonen mellom $X\boldsymbol{w}(\Delta)$ og $\boldsymbol{y}$ er funnet, kan den tilhørende vektingsvektoren beregnes med funksjonen *w_calc*. Dersom den tilhørende vektingsvektoren til $\Delta=-1$ eller $\Delta=1$ gir størst korrelasjon mellom $X\boldsymbol{w}(\Delta)$ og $\boldsymbol{y}$, blir vektingsvektoren en vektor med 0-er for alle forklaringsvariablene, og henholdsvis en 1-er for forklaringsvariabelen med høyest standardavvik eller høyest korrelasjon. (Dersom det er m forklaringsvariable med høyest standardavvik eller korrelasjon blir disse vektet med $1/\sqrt{m}$).

Fordelen med PLS-VS er at den deler mange av de attraktive egenskapene til PPLS samtidig som den er istand til å utføre variabelseleksjon. Sammenlignet med ordinær PLS vil PLS-VS (og PPLS) produsere enklere modeller, og sammenlignet med PPLS vil modeller fra PLS-VS gjerne inkludere færre forklaringsvariable. En illustrasjon av disse betraktningene er vist i figur 4.7 der vi sammenligner modeller for PLS, PPLS og PLS-VS som alle er basert på

beregning av tre komponenter fra ett datasett med 500 forklaringsvariable:



Figur 4.7: *Illustrasjons av regresjonskoeffisienter for PLS, PPLS og PLS med variabelseleksjon.*

Det framgår tydelig at PPLS og PLS-VS gir enklere modeller enn PLS som har inkludert samtlige forklaringsvariable i modellen. Ved opptelling viser det seg at PPLS modellen i dette eksempelet har selektert bort 57 forklaringsvariable, mens PLS-VS modellen har selektert bort 417 forklaringsvariable.

5 Presentasjon av data

Når store datamengder og datasett med mange variabler skal analyseres, kan det være en utfordring å få god oversikt over hvordan ulike metoder fungerer ved kun å studere tall i en tabell. Muligheter som kan bedre oversikten betraktelig er å uttrykke resultatene grafisk i form av grafer og diagrammer. Dette er allikevel en utfordring da dataene som analyseres inneholder hundrevis av dimensjoner og det er flere ulike metoder som sammenligne. Dette lar seg ikke uttrykkes billedlig i to eller tre dimensjoner, men ved å finne smarte løsninger kan man allikevel få frem interessante detaljer. Vi forsøker derfor å uttrykke interessante sider av resultatene ved å projisere oss ned i relevante underrom.

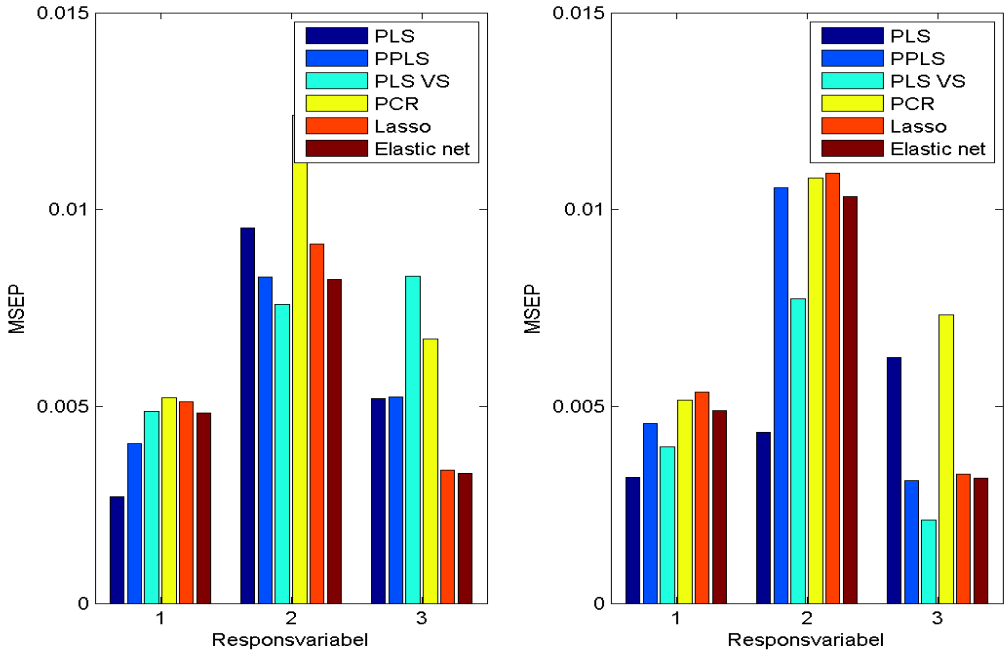
5.1 Søylediagram

For å kunne bedømme hvor gode metodene er i forhold til hverandre, er noe som ofte gjøres å se på hvordan metodene klarer å predikere på uavhengige datasett. Et vanlig mål på prediksjonsfeil er MSE – Mean Square Error. Dette er gjennomsnittet av de kvadrerte avvikene mellom den opprinnelige responsen og responsen vi har predikert for hver observasjon. MSE er et godt mål for prediksjonsfeil og muliggjør sammenligning av prediksjonsegenskaper på metodene seg imellom.

Hver metode produserer modeller av ulik kompleksitet i analysen, og det vil bli en alt for omfattende oppgave å sammenligne alle. Vi kan derfor plukke ut et mindre antall modeller fra hver og sammenligne prediksjonsfeilen (MSE) de får når de predikerer responsen på et uavhengig testsett. Dette vil gi et overordnet inntrykk av hvor gode metodene er sammenlignet med hverandre.

En utfordring i å presentere resultatene på en ryddig måte ligger i at det blir mange tall som skal sammenlignes. I tillegg har to av datasettene som analyseres flere responser. Her vil det være interessant å studere om metoder skiller seg ut for flere responser, da responsen blir predikert fra den samme X-matrisa. En fordel vil derfor være å presentere resultatene av prediksjonene av responser fra samme datasett sammen.

En måte vi fant tilfredsstillende til dette var via flere søylediagram med felles skala og ulike farger for hver metode, da fargene forenkler differensieringen mellom metoden som studeres. Dette gjør at modellen til en metode relativt enkelt kan identifiseres for hver respons, samtidig som de ulike modellene fra en metode også relativt enkelt kan sammenlignes.



Figur 5.1: *Illustrasjon av søylediagram for prediksjonsfeil (MSE) på testsettet for de ulike metodene. Til venstre er resultatet fra modellene med lavest feil i kryssvalideringen, og til høyre er resultatet for den enklere og ikke signifikant forskjellige modellen.*

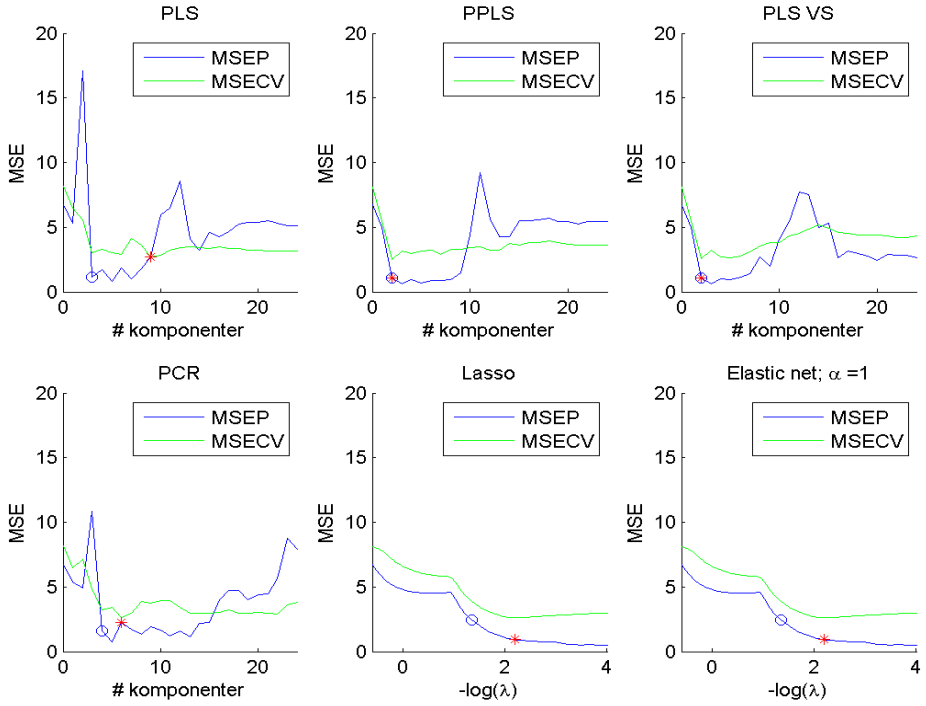
5.2 MSE mot kompleksitet

En annen ting som er interessant å rette fokuset mot er hvordan prediksjonsfeilen endrer seg i forhold til kompleksitet på modellen for hver metode. Dette kan gi et inntrykk av stabiliteten i modellbyggingen for hver metode. God stabilitet kombinert med god prediksjon er gode egenskaper vi ønsker at metoden som anvendes skal ha. Dersom en metode predikerer godt, men er bygget svært ustabilt, er det forbundet mer usikkerhet til hvor godt metoden kan predikere på nye data. I praktiske anvendelser er man ikke kun interessert i hvor god prediksjon en metode gjør, men også hvor sikre vi kan være på prediksjonen. Det er derfor interessant å se på hvordan endringen i prediksjonsfeil endres med kompleksiteten i modellen for de ulike metodene.

Ett annet moment som kan indikere noe om stabiliteten til en metode, er ved å sammenligne modellenes prediksjonsfeil i forhold kompleksiteten av modellen i kryssvalideringen og på det uavhengige testsettet. Kryssvalideringen brukes som et verktøy for å simulere hvordan prediksjonsfeilen endres med kompleksiteten, og dette bør gi en indikasjon på hvordan prediksjonsfeilen vil være på et uavhengig testsettet. Svært store avvik mellom disse kan indikere ustabilitet.

For PCR og PLS-metodene kan vi relatere økende kompleksitet i modellene til et økende antall komponenter. I lasso og elastic net har vi ikke de samme komponentene som gir oss noe mål på kompleksiteten, men har derimot straff-parameteren λ. En minkende verdi av λ tillater å inkludere flere variabler i modellen eller å øke regresjonskoeffisientene til de som allerede er inkludert. Dette kan derfor sees på som en økende kompleksitet. λ-parameteren endres over et relativt stort område av verdier, og endringen av prediksjonsfeil kan da observeres i et veldig komprimert lite område av parameteren. Det kan istedenfor lønne seg å bruke logaritmen av parameteren i plottet, da dette jevner ut endringene.

Ved å bruke prediksjonsfeil som en funksjon av økende grad av kompleksitet vil vi kunne få frem hvor stabil en metode er og ha et grunnlag for å kunne sammenligne stabiliteten mellom metodene. Under følger en illustrasjon av hvordan prediksjonsegenskapene til en metode i forhold til kompleksiteten på modellen kan presenteres.



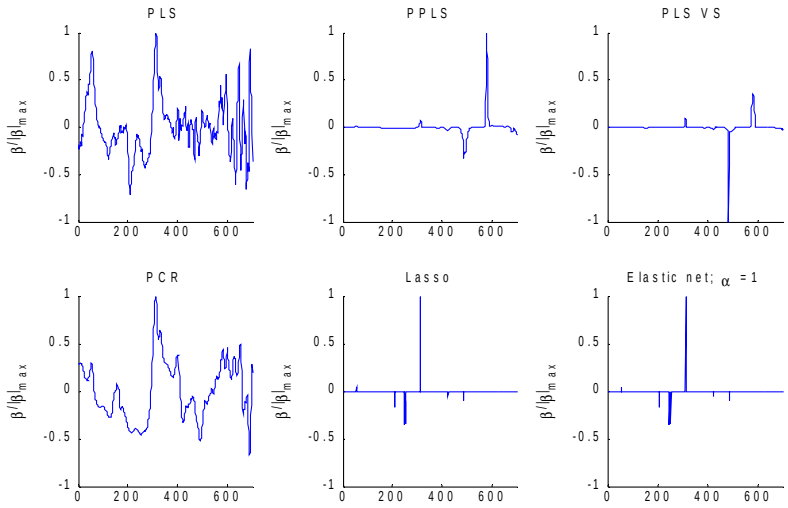
Figur 5.2: *Illustrasjon av prediksjonsfeil som funksjon av økende kompleksitet for de ulike metodene.Den blå kurven representerer prediksjonsfeilen på testsettet, og den grønne representerer feilen i kryssvalideringen. Det røde krysset indikerer modellen med lavest kryssvalideringsfeil, og den blå sirkelen indikerer den enklere men ikke signifikant forskjellige modellen.*

5.3 Regresjonskoeffisienter

Vi har tidligere understreket viktigheten av å kunne produsere enkle og tolkbare modeller,

spesielt i arbeid med multivariate datasett. Det vil derfor også være av interesse å sammenligne hvor enkle modeller de ulike metodene produserer. Såfremt det ikke går på bekostning av dårligere prediksjon favoriseres metoder som produserer enkle modeller, da dette øker tolkbarheten. For å få et inntrykk av enkelheten til modellene, kan vi plotte regresjonskoeffisientene for metodene. Hvis to modeller fokuserer mot de samme forklaringsvariablene indikerer dette at modellene bruker de samme underrommene for å predikere responsen. Dette gjør oss i stand til å kunne avdekke om to ulike metoder som PPLS og lasso som har forskjellige kriterier for modellbyggingen, allikevel ender opp med å predikere responsen i de samme underrommene.

Modellene til de ulike metodene har i mange tilfeller ulik størrelse på regresjonskoeffisientene til forklaringsvariablene. For å være i stand til å avdekke likheter og ulikheter, vil det være hensiktsmessig å projisere regresjonskoeffisientene ned i et relevant underrom der de lettere lar seg sammenligne. Dette kan gjøres ved å omskalere koeffisientene i modellen slik at den maksimale absoluttverdien blir lik 1, og vi kan dermed sammenligne koeffisientene i modellene på en felles skala. Vi får da også et inntrykk over de relative forskjellene innad i hver modell, og hvilke forklaringsvariable som vektlegges betydelig.

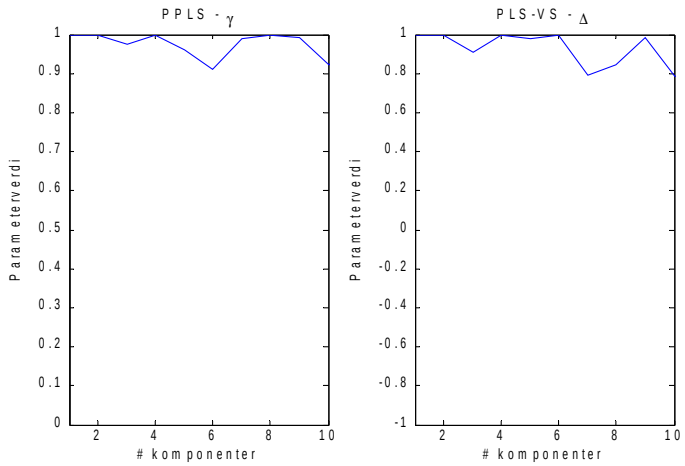


Figur 5.3: *Illustrasjon av de skalerte regresjonskoeffisientene for modellen til de ulike metodene.*

5.4 Potensering og trunkerings-parametere

For å kvalitetssikre at PLS-VS fungerer slik vi ønsker, kan det være interessant å sammenligne trunkeringsparameteren for hver komponent med potenseringsparameteren i PPLS. Da PLS-VS er motivert og videreutviklet fra PPLS, vil det være naturlig at også

parameterne gir komponenter som fokuserer på forklaringsvariable med liknende egenskaper. For eksempel hvis PPLS i en komponent inkluderer svært høyt korrelerte variable, vil vi også forvente at PLS-VS også gjør det, og både γ og Δ være lik eller svært nær 1. Dersom metodene fokuserer mot forklaringsvariable med høyt standardavvik, vil γ være svært nær 0 og Δ være svært nær -1. Ved å plotte parameterverdien for γ på en skala fra 0 til 1, og parameterverdien for Δ på en skala fra -1 til 1, kan vi få et inntrykk om komponentene blir laget av forklaringsvariable med samme egenskaper. Ved stor likhet i parameterverdiene, kan dette tyde på at PLS-VS finner mye av de samme relevante underrom som PPLS og at PLS-VS fungerer slik vi vil. Det er trolig at etter et vist antall komponenter vil de optimaliserte parameterne avvike i større grad, på grunn av at den deflaterte X matrisa og den deflaterte responsen blir nummerisk mer og mer ulike for hver iterasjon. En indikasjon på at vår metode fungerer slik vi ønsker er derfor å fokusere på de første komponentene som lages. En illustrasjon over hvordan sammenhengen av parameterne kan presenteres følger under

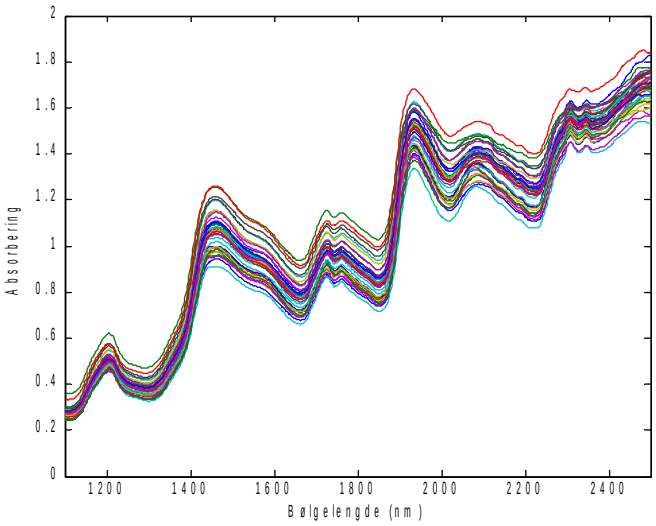


Figur 5.4: *Illustrasjon av potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent.*

6 Testing på reelle datasett

6.1 Datasett

6.1.1 Deigdata



Figur 6.1: *Spektra av deigdatasettet.*

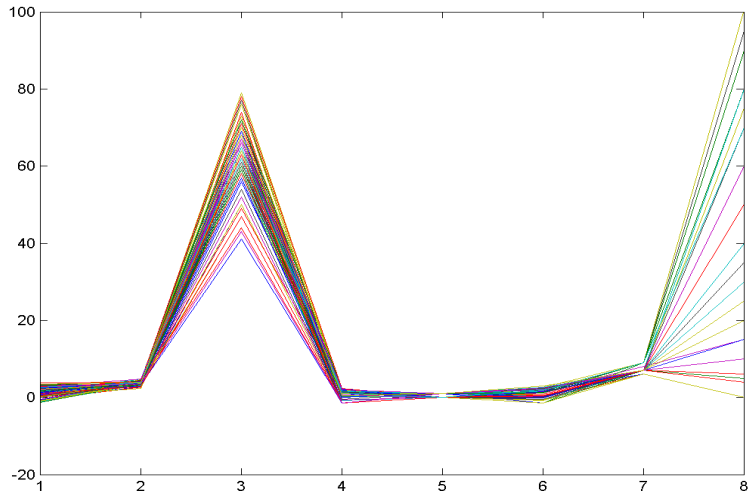
Dette er data fra målinger gjort med nær infrarød (NIR) spektroskopi på kjeksdeig(Indahl 2005). Treningsdatasettet har N = 40 observasjoner av p = 700 bølgelengder (1100 – 2498 nm med på 2 nm mellom hver variabel). I tillegg er det et testdatasett med 32 observasjoner. De to settene ble laget og målt i forskjellige sammenhenger. Datasettet har fire responsvariabler der det har blitt målt prosentandelen av henholdsvis (a) fett, (b) sukker, (c) mel og (d) vann i deigprøver.

Dette er et datasett med flere forklaringsvariable enn antall observasjoner, og her vil som kjent OLS ha vanskeligheter med å finne entydige modeller for prediksjon. Plottet av spekteret viser at kurvene som beskriver hver observasjon er glatte, noe som betyr at nabovariabler er høyt korrelerte for dette datasettet (Figur 6.5). Betydningen av dette er at nabovariabler spenner ut store deler av det samme underrommet, og er for eksempel et resultat av de samme kjemiske egenskapene i en prøve. Det at nabovariabler er høyt korrelerte, har betydning for hvordan modellene til de ulike metodene kan tolkes. Om en modell velger å gi høy regresjonskoeffisient til den første eller den andre av to nabovariabler, betyr dette svært lite da variablene inneholder så mye lik informasjon. To metoder som

inkluderer hver av to nabovariable i sine modeller gir i praksis tilnærmet like modeller.

For deig- og øldatasettet (omtales senere) refereres det kun til bølgelengdene som variabler i stigende rekkefølge, der variabel i er representerer en spesiell bølgelengde gitt av: $\mathrm{i} \rightarrow [(i-1)\cdot 2+\lambda_{\mathrm{low}}]\mathrm{nm}$ der λ_{low} tilsvarer bølgelengde til den første variabelen. For nærmere beskrivelse av datasettet se referanse (Osborne et al. 1984).

6.1.2 Prostatadata

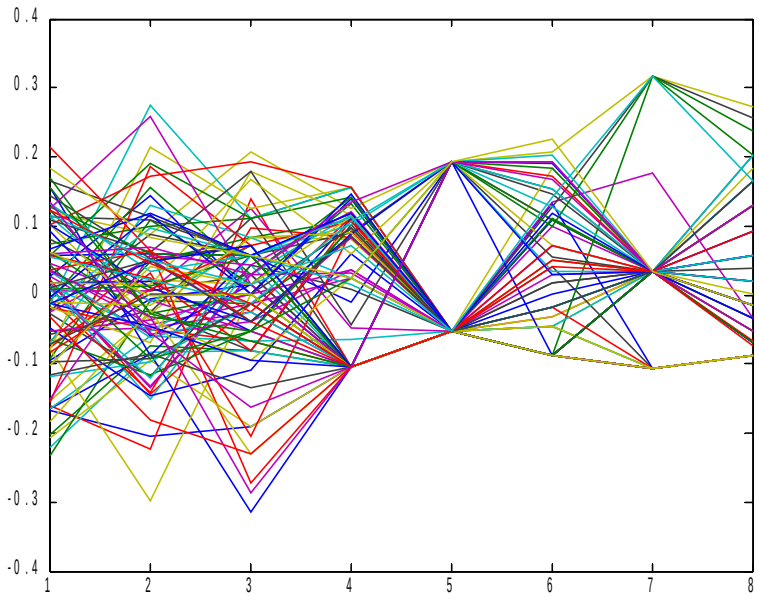


Figur 6.2: *Plott av variablene i prostatadatasettet.*

Dette er data hentet fra et studie av prostatakreft (Stamey et al., 1989). Datasettet er tidligere for å illustrere egenskaper ved elastic net i forhold til lasso (Zou & Hastie 2005). Datasettet har N = 97 observasjoner og p = 8 variabler. I motsetning til de øvrige datasettene har dette datasettet flere observasjoner enn antalle forklaringsvariable. Treningsdatasettet har N = 67 observasjoner og testdatasettet har 30 observasjoner. Forklaringsvariablene representerer 8 kliniske mål av pasienter med prostatakreft. Alle målene er logaritmetransformert, med unntak av forklaringsvariabel nummer 3 som representerer alder, og forklaringsvariabel nummer 8 som representerer en prosentvis score som angir aggressiviteten til kreften. Datasettet har én responsvariabel som representerer logaritmen av ett prostataspesifikt antigen. For nærmere beskrivelse av datasettet se referanse (Zou & Hastie 2005).

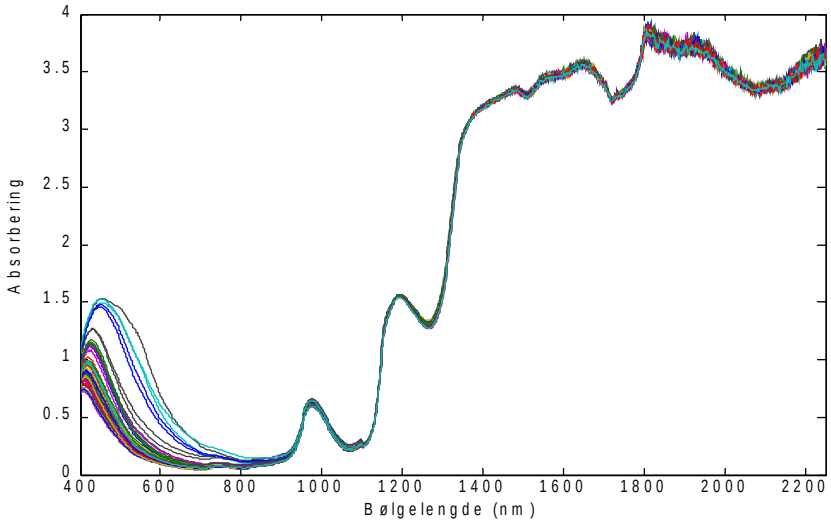
Da forklaringsvariabel nummer 3 og 8 ikke er logaritmetransformert, ser vi av figur 6.2 at disse har vesentlig større variasjon enn de andre variablene. Det skiller omtrent en størrelsesorden mellom variansen til variabel 8 og variabel 3, og ytterligere mellom en og to

størrelsesordener til de øvrige variablene. I artikkelen som omhandlet elastic net og lasso hvor datasettet ble benyttet, ble forklaringsvariablene standardisert i forkant av analysen. I figur 6.3 vises de standardiserte regresjonskoeffisientene.



Figur 6.3: *Plott av de sentrerte og standardiserte variablene i prostatadatasettet.*

6.1.3 Øldata



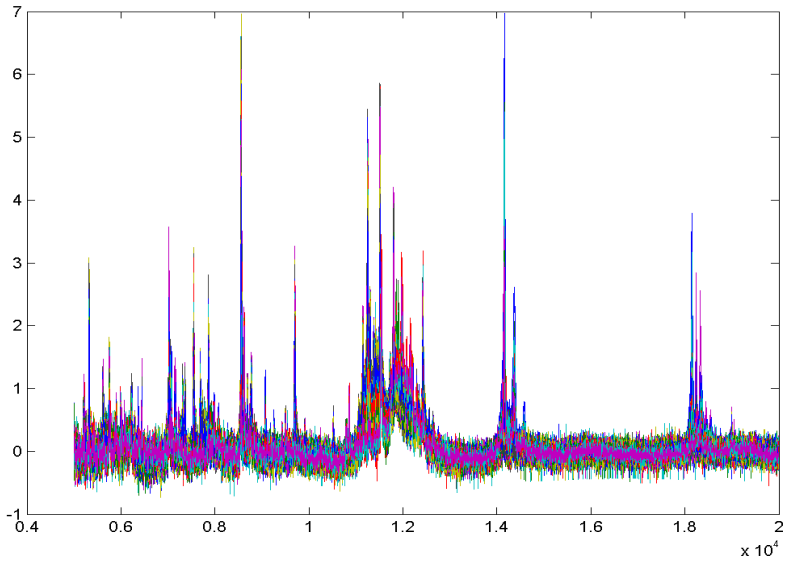
Figur 6.4: *Spektra av øldatasettet.*

Dette er data hentet fra målinger gjort med nær infrarød (NIR) spektroskopi på øl (Indahl 2005). Datasettet har N = 60 observasjoner av p = 926 bølgelengder (400 til 2250 nm med 2 nm mellom hver variabel). Det finnes ikke ett separat testsett i datasettet, så dette lages ved å

sortere observasjonene i stigende rekkefølge, og plukke ut hver tredje observasjon (i dette tilfellet 2, 5, 8, ..., 59) til testsettet. De øvrige observasjonene brukes som treningssett. En oppdeling som denne sikrer at treningssettet og testsettet representerer tilnærmet den samme variasjonen som det totale datasettet inneholdt. Datasettet har én responsvariabel som representerer den opprinnelige konsentrasjonen av ekstrakt i øl.

Plottet av spekteret viser at kurvene som beskriver hver observasjon er glatte og svært jevne for variablene som tilsvarer bølgelengder mellom 900 og 1700 nm, og i hver ende av skalaen på bølgelengdene varierer observasjonene mer. Der kurvene ligger jevnt er variablene høyt korrelerte. For variable som representerer bølgelengder mindre enn 900 nm er det selve observasjonene som varierer mye i forhold til hverandre, mens for variable som representerer bølgelengder større enn 1700 nm skyldes variasjonen støy fra instrumentet som blir brukt. Dette er typiske eksempler på fenomener som oppstår med NIR målinger og er med på å gjøre prediksjonen vanskeligere. På samme måte som for deigdatasettet gjelder det at dersom to modeller har inkludert nabovariable, gir dette i praksis tilnærmet like modeller, og prediksjonene vil også være svært like. For nærmere beskrivelse av datasettet se referanse (Osborne et al. 1984).

6.1.4 MALDI-TOF-data



Figur 6.5: *Spektra av MALDI-TOF-datasettet.*

Dette er data hentet fra målinger på blandinger av ku-, geit- og saue-melk (Liland et al. 2009). Metoden som benyttet heter Matrix-assisted laser desorption/ionisation time-of-flight

(MALDI-TOF) og er en massespektrometriteknikk som brukes for å identifisere proteiner, peptider og andre ioniserbare stoffer. Når denne teknikken anvendes er det vanlig å analysere flere prøver av hver blanding (dette kalles replikat), da målingene kan bli ubrukelige om man ikke får plassert prøven rett i instrumentet eller at molekylene ikke blir ionisert.

Forsøket er et såkalt designet eksperiment der blandinger av 3 melketyper i 45 ulike blandingsforhold blir målt. Hvert blandingsforhold har fire replikater, og hele forsøket ble utført 2 ganger. Resultatene fra det første forsøket blir brukt som treningssettet og har N = 180 observasjoner (45 blandinger á 4 replikater). For å redusere støyen forbundet med teknikken beregnes gjennomsnittet av de fire replikatene i det andre forsøket, og disse 45 «gjennomsnittsobservasjonene» danner grunnlaget for testsettet. Spekteret som er brukt hadde opprinnelig 21451 forklaringsvariable, men ble redusert til kun p = 6179 forklaringsvariable da man med apriorikunnskap kjenner til at den øverste delen av spekteret ikke inneholder signifikant informasjon. Før analysen ble gjennomført ble datasettet normalisert og grunnlinjekorrigert.

Datasettet har 3 responsvariabler, der hver av disse indikerer andelen av de ulike melketypene. Blandingene av melketypene ble gjort for hånd, og har i etterkant blitt målt eksakt. I analysen benyttes de eksakte målingene av melkeblandingene.

6.2 Bruk av metoder

6.2.1 Begrensinger

I denne oppgaven sammenlignes prediksjonsegenskapene til modeller fra ulike metoder på fire forskjellige datasett, i tillegg til å studere hvordan vår metode PLS-VS fungerer sammenlignet med andre. Det er derfor naturlig å sammenligne PPLS og PLS, som danner grunnlaget og motivasjonen for vår metode. Vi velger også å ta med PCR, da denne metoden i likhet med de andre gjør regresjonen på lineærkombinasjoner av de opprinnelige forklaringsvariablene. Til slutt er det også interessant å studere hvordan PLS-VS fungerer sammenlignet med andre metoder som anses som relativt godt egnet for analyse av komplekse data, og derfor inkluderes lasso og elastic net i analysen.

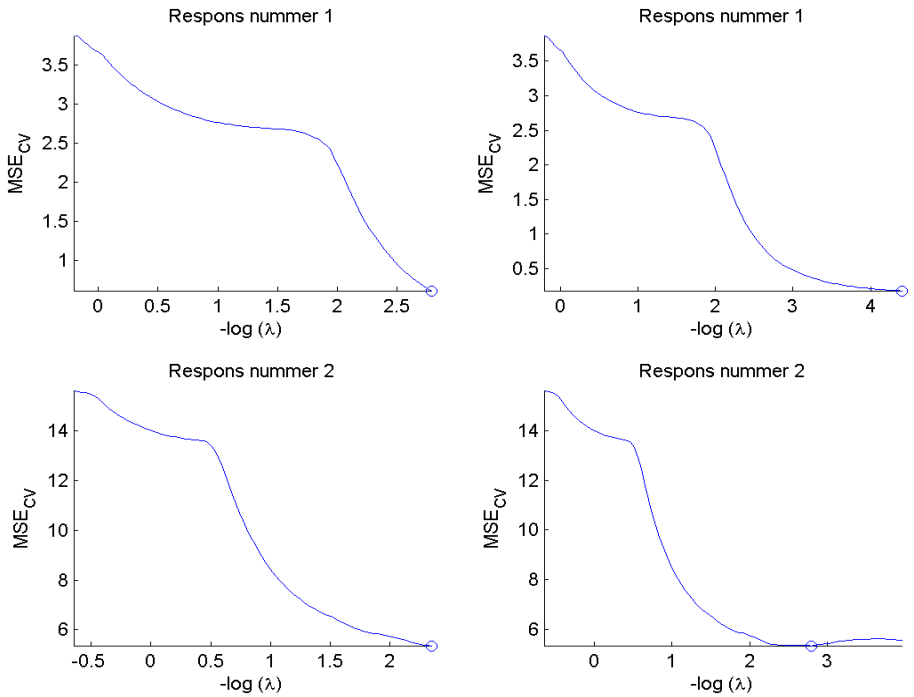
For flere av metodene har vi justerbare innstillinger og i tillegg gir hver metode oss modeller av ulik kompleksitet. Det blir totalt svært mange mulige kombinasjoner å kunne sammenligne metodenes prediksjonsegenskaper på i den påfølgende analysen. For å begrense mulighetene

har vi valgt ut forhåndsinnstillinger som skal brukes for metodene og en strategi for hvilke modeller fra metodene som skal studeres i analysen.

For PPLS og PLS-VS har vi anledning til å begrense området på verdiene på potenseringsparameteren og trunkeringsparameteren som er tillatte for modellene. Vi har her valgt å tillate modellene å kunne optimere på alle parameterverdier i gyldighetsområdet til metodene. PPLS har derfor mulighet for å velge verdier for gamma i mengden $\gamma \in [0,1]$, og PLS-VS har mulighet til å velge verdier for delta i mengden $\Delta \in [-1, 1]$.

For metodene elastic net og lasso har vi benyttet et publisert MATLAB-script for å estimere modellene (*Glmnet for Matlab* 2011). Algoritmen benytter Pathwise Coordinate Optimization (Friedman et al. 2007). Etter anbefaling fra forfatterne av koden har vi for elastic net med kryssvalidering funnet en optimal parameter for α fra utvalget $\alpha = [0.1, 0.2, \ldots, 0.9, 1.0]$, der α angir andelen av restriksjonen som skal være lasso-relatert, og (1-andelen) som skal være ridge-relatert. Denne måten å optimalisere α parameteren i elastic net er dermed svært beregningsbesparende i forhold til å evaluere alle mulige verdier for α.

For elastic net og lasso er det i scriptet forhåndsinnstilte minimumsverdier for λ parameteren, avhengig av forholdet mellom antall observasjoner og antall variabler. Når det er flere variabler enn observasjoner er minste tillatte verdi for λ 0.05, og i det motsatte tilfelle er minste tillatte verdi for λ 0.01. Denne forhåndsinnstillingen har vi også benyttet oss av, med unntak av for deigdatasettet. Her ser det ut til at begrensningen på 0.05 hindrer oss i å finne λ parameteren som gir minimums kryssvalideringsfeil (Figur 6.6). Ved å senke den nedre grensen til 0.01 tillates mer komplekse modeller, noe som gjør at vi i kryssvalideringen er bedre istand til å gjøre gode prediksjoner. For den første responsen ser vi at kurven for kryssvalideringsfeilen som funksjon av kompleksiteten flater ut med den nye grensen, og for den andre responsen får vi et tydelig minimum for prediksjonsfeilenfeilen. Vi bruker dermed 0.01 som den laveste tillatte grensen på λ parameteren når vi analyserer deigdatasettet, men for de andre datasettene benyttes de forhåndsinnstilte grensene på 0.05 når $p > n$ og 0.01 når $n > p$.



Figur 6.6: *Prediksjonsfeil for lasso i kryssvalideringen som funksjon av modellkompleksitet. Til venstre er den nedre begrensningen for straffparamteren* λ *= 0.05 (forhåndsinnstilling). Til høyre er den nedre begrensningen for straffparameteren* λ *= 0.01. Grafene tilhører henholdsvis respons nummer 1 og 2 for deigdatasettet.*

For PCR og PLS-metodene har vi muligheten til å avgjøre maksimalt hvor mange komponenter som skal inkluderes i modellen. For prostatadatasettet med kun 8 forklaringsvariable velger vi å ha muligheten for å inkludere maksimalt 8 komponenter i modellene, da beregningen av komponentene ikke er særlig krevende. I analysen gjort tidligere på deig- og øl-datasettet ble modeller med opptil 25 komponenter evaluert. For å gjøre resultatene mest mulig sammenlignbare vil vi også tillatte opptil 25 komponenter i modellene. I analysene gjort på MALDI-TOF-datasettet ble det i tidligere analyse ble modeller med opptil 10 komponenter evaluert. For å også her gjøre resultatene mest mulig sammenlignbare vil vi også tillatte opptil 10 komponenter i modellene. For MALDI-TOF-datasettet er det spesielt gunstig å ikke beregne for mange komponenter, da det høye antall forklaringsvariablene fører til at arbeidet er svært beregningskrevende og vil ta lang tid. I tillegg vil vi også ta med 0-modellen til hver metode i analyse, som tilsvarer en tom modell som predikerer responsen lik gjennomsnittet av alle responsverdiene.

I sammenligningen plukkes det ut to modeller for hver metode som danner grunnlaget for analysen. Den ene modellen som plukkes ut er modellen som har lavest feil i kryssvalideringen. Med utgangspunkt i denne sjekkes det om det eksisterer en enklere modell som ikke har signifikant større prediksjonsfeil. Signifikansnivået som benyttes er $\alpha = 0.05$

Den andre modellen som plukkes ut for hver metode er den enklere og ikke signifikant forskjellige modellen. I tilfellene der det ikke eksisterer en enklere modell, brukes modellen med lavest kryssvalideringsfeil også her.

For å spesifisere hva som menes med enklere modeller, nevnes kort hva som legges til grunn for dette. For PLS-metodene og PCR regnes modellen for å være mer kompleks jo flere komponenter som inkluderes. Lasso og elastic net derimot baserer modellenes kompleksitet med utgangspunkt i straff-parameteren λ. Her vil en minkende verdier på straff-parameteren øke kompleksiteten på modellen. Enklere modeller som studeres er dermed modeller med færre komponenter enn den med lavest feil i kryssvalideringen for hver metode, eller modeller med høyere verdi på straff-parameteren λ.

6.2.2 Kryssvalidering

I tidligere studier gjort på datasettene vi benytter har det blitt brukt ulike former for kryssvalidering. For å gjøre våre resultatene mest mulig sammenlignbare med disse har vi valgt å bruke samme type kryssvalidering. På deig- og øldatasettet brukes dermed 5-fold kryssvalidering, på prostatadatasettet brukes 10-fold kryssvalidering (Zou & Hastie 2005). For disse datasettene brukes systematisk utplukking av observasjoner, slik at hver k-te observasjon blir plukket ut i hver runde i kryssvalideringen. På MALDI-TOF-datasettet brukes 45-fold kryssvalidering, og her plukkes replikatene for hvert blandingsforhold ut i hver runde.

6.2.3 Prosessering

I tidligere publikasjoner har det blitt gjort ulike former for prosessering av datasettene både før og etter analysen. For å få mest mulig sammenlignbare resultater til tidligere forskning har vi forsøkt å følge denne prosesseringen i størst mulig grad:

- Deig- og øldatasettet ble brukt uten noen form for prosessering verken før eller etter analysen (Indahl 2005).
- Prostatadatasettet ble preprosessert ved å sentrere og standardisere X matrisa før analysen, og ingen form for prosessering ble gjort etter analysen (Zou & Hastie 2005).
- MALDI-TOF-datasettet ble pre-preprosessert ved at responsene ble skalert slik at summen av disse i hver observasjon ble 1. I tillegg ble den tredje responsen kvadratrottransformert før analysen. Etter analysen blei responsene post-prosessert før

de ble sammenlignet med de opprinnelige responsene. I post-prosesseringen ble først den tredje responsen re-transformert med kvadrering, og av apriori kjennskap til forsøket ble hver respons trunkert til å ha verdi mellom 0 og 1. Deretter ble responsene skalert slik at summen av disse for hver observasjon ble lik 1. Trunkeringen gjøres da vi vet at i en blanding er maksimal andel 1 og minimal andel 0. Skaleringene gjøres da vi vet at en prøve kun består av de tre melketypene og derfor må andelene summeres til 1. Den siste skaleringen bidrar til å stabilisere prediksjonene, da da transformeringen og re-transformeringen ikke garanterer at prediksjonene av responsene for hver prøve fortsatt summerer til 1.

I vår analyse blir de nevnte prosesseringene fullt for deig-, øl- og prostatadatasettet. For MALDI-TOF-datasettet har vi derimot valgt å utelate enkelte elementer i prosesseringen. I post-prosesseringen ønsker vi ikke å endre prediksjonene av responsene ved å skalere de i det siste steget. Modellene elastic net produserer for hver respons kan ha ulik verdi for α parameteren. Vi ønsker derfor ikke å gjøre prediksjonene avhengige av hverandre med skalering dersom α parameterne i modellene for hver respons ikke er lik. Som konsekvens av å utelate skaleringen, velger vi også å utelate transformeringen av den tredje responsen i analysen. Da hensikten med skaleringen er å stabilisere prediksjonene på grunn av transformeringen, er det derfor naturlig at også dette utelates.
For å oppsummere blir den prosesseringen vi gjør for MALDI-TOF-datasettet skalering av responsene slik at de summerer til 1 i pre-preprosesseringen, samt trunkering av responsene slik at de har verdier mellom 0 og 1 i post-prosesseringen. Dette gjøres ikke kun for elastic net, men for samtlige metoder.

6.2.4 Hva skal sammenlignes

I analysen vil si sammenligne prediksjonsegenskapene til de ulike metodene for modellene med lavest kryssvalideringsfeil og de enklere metodene som ikke er signifikant forskjellig fra disse. I tillegg vil det være interessant å sammenligne stabiliteten til modellene, hvor enkle modeller som produseres, om metoden selekterer variabler og hvor beregningskrevende de ulike metodene er. For å få et inntrykk av hvordan vår videreutviklede PLS metode fungerer er det også interessant å sammenligne med hvordan den oppfører seg sammenlignet med PPLS. For hvert datasett vil vi starte med å se på prediksjonsfeil for de valgte modellene til hver metode, for så å gå mer inn på detaljer rundt modellene. For datasettene med flere responser vil vi gå gjennom en og en respons i kronologisk rekkefølge.

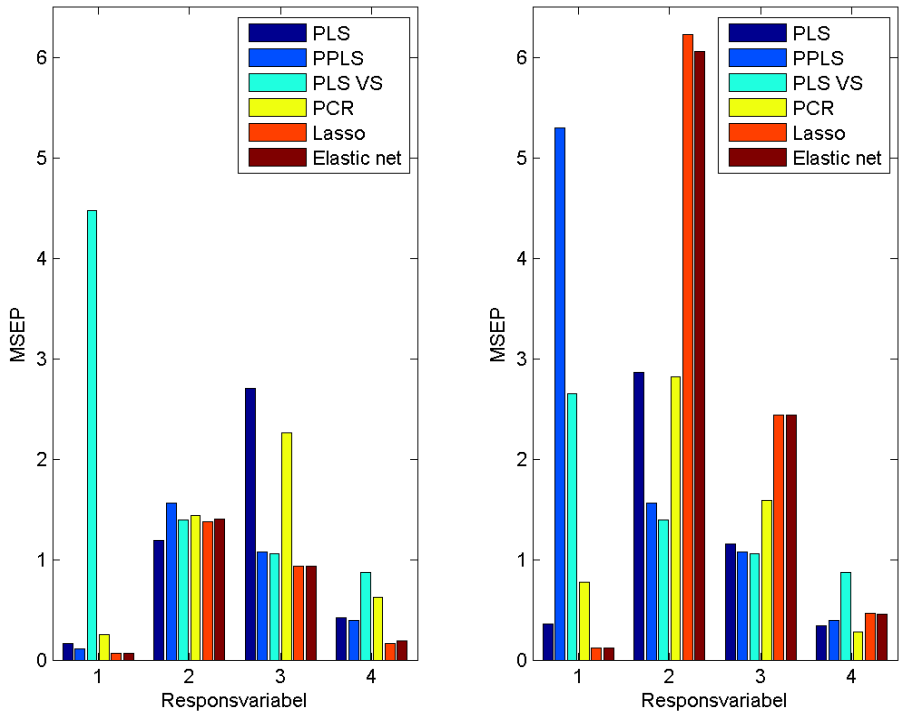
6.3 Hvilke modeller sammenlignes

For å velge ut hvilke modeller som skal sammenlignes fra hver metode har vi benyttet oss av resultatene vi har fått i kryssvalideringen. Vi har plukket ut to modeller som skal sammenlignes for hver metoden: modellen med lavest kryssvalideringsfeil og en modell som er mindre kompleks men ikke signifikant forskjellig fra denne. Dette er gjort da disse modellene ofte er de som er interessante å bruke i praktiske anvendelser. Resultatene av disse modellenes prediksjonsfeil på et uavhengig testsett er vårt grunnlag for å si noe om hvor god en metode er sammenlignet med en annen. Dette vil gi et overordnet inntrykk av hvor gode metodene er sammenlignet med hverandre.

6.4 Deigdatasettet

I analysen av disse dataene har vi fått noe avvikende resultater fra det som er funnet tidligere (Indahl 2005). Noe av avvikene kommer at siden gang analysen blei gjennomført har koden for PPLS-metoden gjennomgått kvalitetsikring og blitt forbedret. Det vil derfor også trolig forekomme avvik i øldatasettet, da også dette datasettet blei brukt i den ovennevnte analysen. Resultatene vil like fullt være gyldig for å gjøre sammenligning av metodene, men vi nevner dette slik at leseren er informert om avvikende resultater.

Av figur 6.7 ser vi at lasso og elastic net scorer jevnt over best med lavest feil for tre av fire responsvariabler for modellene med lavest kryssvalideringsfeil, og jevngodt resultat med de andre metodene der de ikke er best. Sammenligner vi derimot de enklere men ikke signifikant forskjellige modellene er det ikke lenger noen metode som skiller seg ut som best. Her produserer hver metode modeller med svært gode og svært dårlige prediksjonsevner for de ulike responsvariablene. Elastic net og lasso er de to beste metodene for den første responsen, PPLS og PLS-VS er de to beste for den andre responsen, PLS, PPLS og PLS-VS er de jevnt beste for den tredje responsen og PCR er best for den siste.

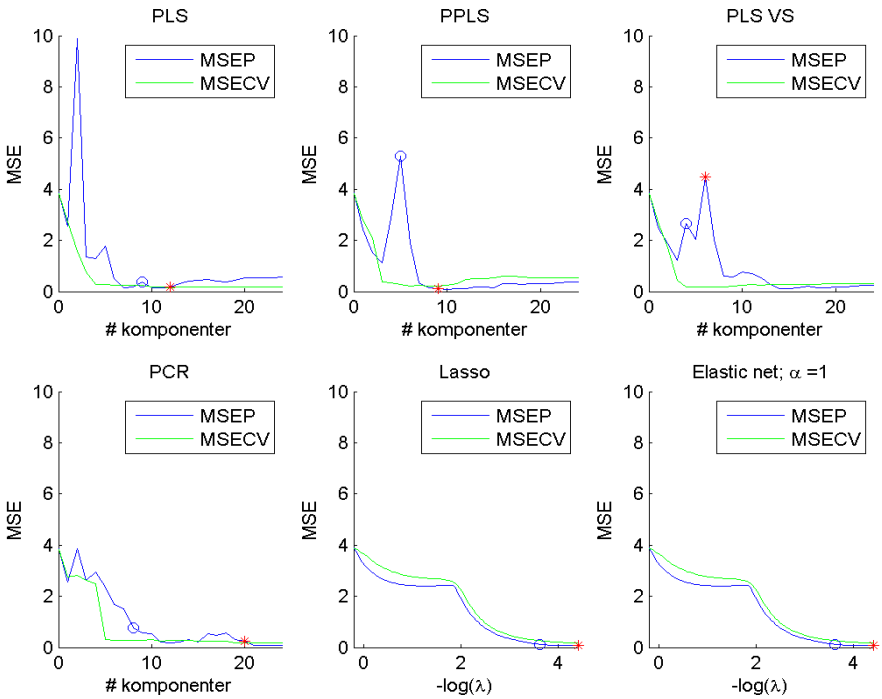


Figur 6.7: *Prediksjonsfeil (MSE) på testsettet for de ulike metodene for hver respons. Til venstre er resultatet fra modellene med lavest feil i kryssvalideringen, og til høyre er resultatet for den enklere og ikke signifikant forskjellige modellen.*

En ting vi legger godt merke til for dette datasettet er at modellene med lavest kryssvalideringsfeil og de enklere modellene har svært ulik prediksjonsfeil for den samme responsen for flere av metodene. Spesielt får PPLS mye høyere prediksjonsfeil for den enklere modellen for den første responsen, og lasso og elastic net får mye høyere prediksjonsfeil for den andre og tredje responsen. Når vi går nærmere inn og analyserer hver respons, skal vi også studere hva dette kan komme av.

6.4.1 Fettinnhold – første respons

For den første responsen skiller PLS-VS seg ut med dårligere prediksjon sammenlignet med de andre metodene for modellene med lavest kryssvalideringsfeil, og PPLS og PLS-VS skiller seg ut for den enklere modellen. Denne oppførselen kan lettest forklares ved å studere prediksjonsfeilen mot kompleksiteten av modellen (Figur 6.8). Her får lasso og elastic net (modellene er identiske da den optimale verdien for $\alpha = 1$) glatte kurver for prediksjonsfeilen på kryssvalideringssettet og på testdatasettet, mens PLS, PPLS og PLS-VS får mer hakkete kurver for testsettet. PCR har også den samme tendensen, men ikke i like stor grad. Ustabiliteten i modellene gjør det vanskelig å velge en optimal modellkompleksitet i kryssvalideringen, da dette har lite samsvar med prediksjonsfeilen på testsettet.



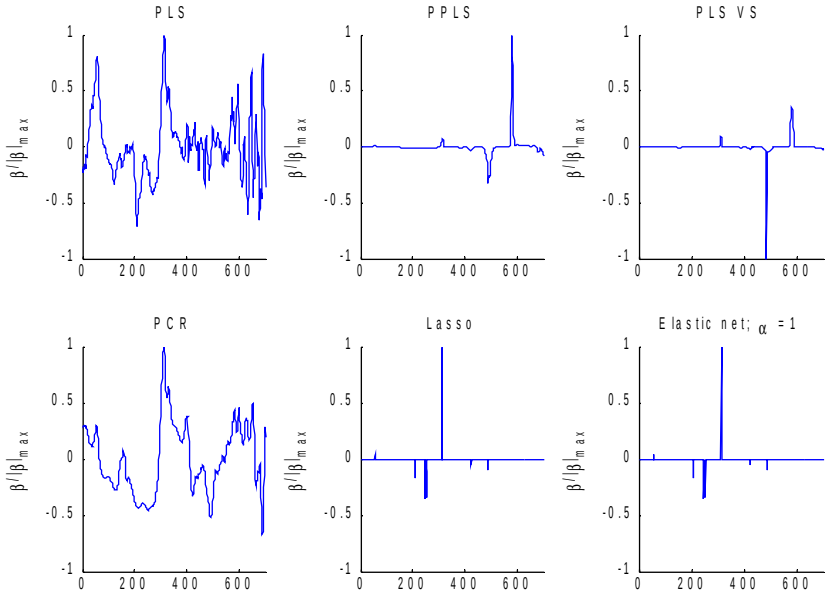
Figur 6.8: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den første responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

For denne responsen slår det svært dårlig ut for PPLS og PLS-VS. PLS-VS velger en modell med seks komponenter i kryssvalideringen i tillegg til en enklere modell med fire komponenter. På testsettet viser det seg at modellene predikerer svært dårlig, og samtlige PLS-VS modeller med et annet antall komponenter har lavere prediksjonsfeil (med unntak av 0-modellen som plasserer seg et sted mellom disse). Også PPLS møter på dette problemet for den enklere modellen med fem komponenter. I kryssvalideringen ser dette ut til å være en god

modell, men når vi tester modellen på det uavhengige testsettet viser det seg at denne modellen predikerer dårligst for metoden.

Dersom modeller med flere komponenter hadde blitt valgt, hadde disse predikert omtrent like godt som lasso og elastic net. En antydning til den samme ustabiliteten ser vi også hos PLS for modellen med to komponenter, men denne blir ikke valgt i modellutvelgelsen. Det ser ut til at dette fenomenet i datasettet kun påvirker PLS-metodene, da vi ikke ser noe lignende for de andre metodene.

En interessant observasjon i figur 6.8, er hvordan prediksjonsfeilen nærmest stabiliserer seg på et nivå for de enkleste modellene, for så synke når modellene øker i kompleksitet. Skillet kommer ganske brått og det kan nesten antydes som et trappetrinn på grafen. Dette kan tyde på at de enkle modellene er for enkle til å avdekke de underliggende strukturene i datasettet, mens etter en viss kompleksistet på modellene er de i stand til å beskrive responsen bedre og prediksjonsfeilen synker brått. Det synes tydeligst for lasso og PCR, og dette kan muligens ha sammenheng med den brå økningen vi så i prediksjonsfeil for PLS-metodene.

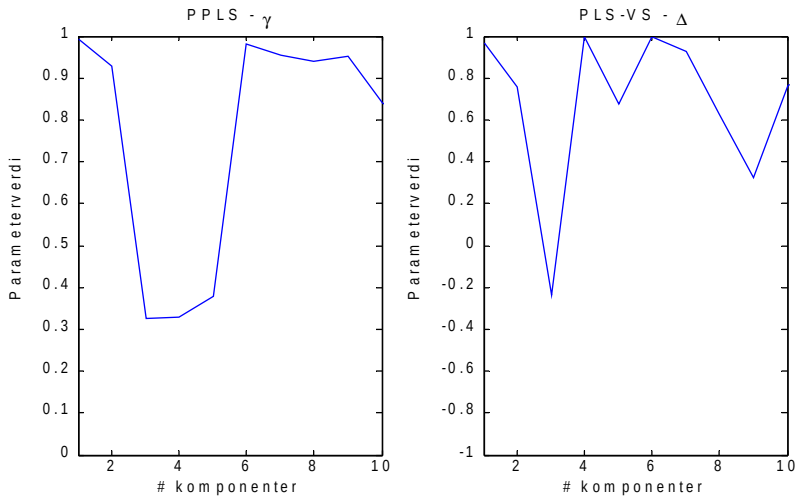


Figur 6.9: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den første responsen.*

PPLS, PLS-VS, lasso og elastic net produserer mye enklere modeller sammenlignet med PCR og PLS for den første responsen (Figur 6.9). En ting vi kan legge merke til er at regresjonskoeffisientene for PLS-VS og PPLS ganske ulike fra elastic net og lasso. Metodene velger forskjellige forklaringsvariable som skal beskrive responsen best mulig. PPLS og PLS-VS inkluderer variabler omkring nummer 500 og 600 i modellen, mens lasso og elastic net

finner variabler omkring nummer 300. Dette betyr trolig at for prediksjonen finner de ulike metodene ulike underrom som de hver seg mener er gode for prediksjon av responsen.

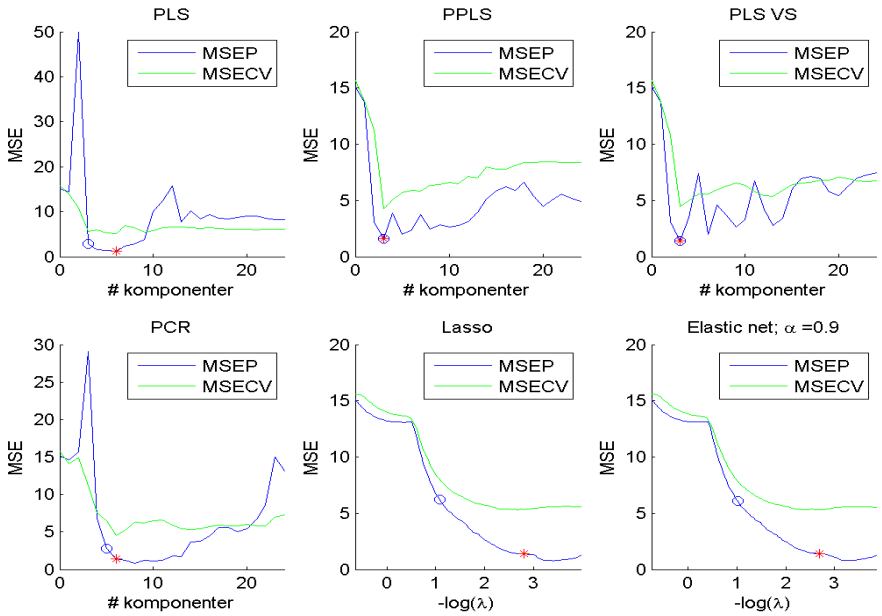
Av figur 6.11 over de optimaliserte parameterne for hver komponent, ser vi at for de tre første komponentene bruker PPLS og PLS-VS lineærkombinasjoner av forklaringsvariable med svært like egenskaper. De to første komponentene blir laget av svært høyt korrelerte forklaringsvariable, og den tredje komponenten preges av forklaringsvariable med høy kovarians. Felles for PPLS og PLS-VS er at de enklere modellene (med henholdsvis fem og fire komponenter) som velges i modellutvelgelsen nettopp har inkludert en komponent med svært høyt fokus på korrelerte variable (Figur 6.10). Begge metodene oppnår svært dårlig prediksjon av responsen for de enklere modellene. Her kan det være interessant å studere om begrensning av potenserings- og trunkerings-parameteren kan bidra til stabilisering av modellene. Vi kommer tilbake til dette etter analysen av de andre responsene i deigdatasettet.



Figur 6.10: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den første responsen.*

6.4.2 Sukkerinnhold – andre respons

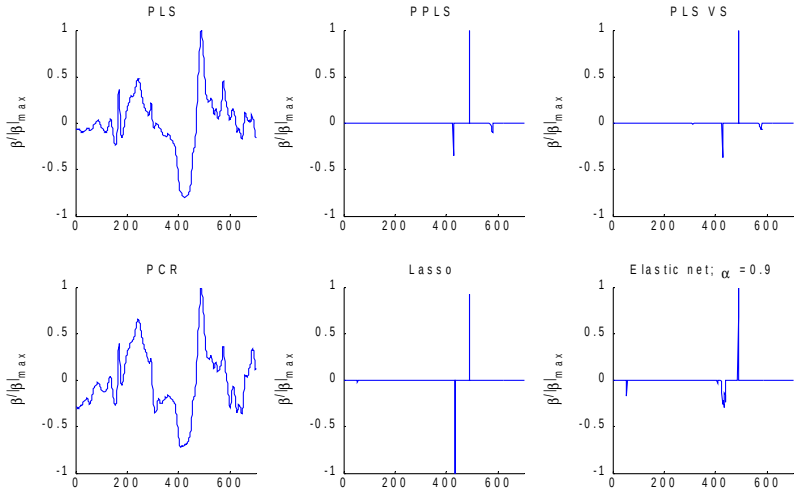
For lasso og elastic net får modellene en markant økning i prediksjonsfeil på testsettet når vi sammenligner modellene med lavest kryssvalideringsfeil med de enklere modellene for den andre responsen. Denne økningen ser ut til at kommer av at modellene med lavest kryssvalideringsfeil gir gode modeller og at de enklere modellene rett og slett er for enkle til å kunne gi god prediksjon av responsen (Figur 6.11). Metodene bygger modellene med svært jevnt synkende prediksjonsfeil, noe som fører til at man får god kontroll på prediksjonsfeilen i forhold til modellkompleksiteten. Dette bidrar trolig til at vi sjelden ender opp med overtilpassede modeller for metodene. De enklere modellene klarer da trolig ikke å avdekket like mye av de underliggende strukturene i datasettet og er derfor ikke i like god stand til å predikere godt.



Figur 6.11: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den andre responsen. OBS: Skalaene på y-aksen er ikke lik for alle metodene. Dette er gjort for å få frem de ustabile metodene til PLS og PCR. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

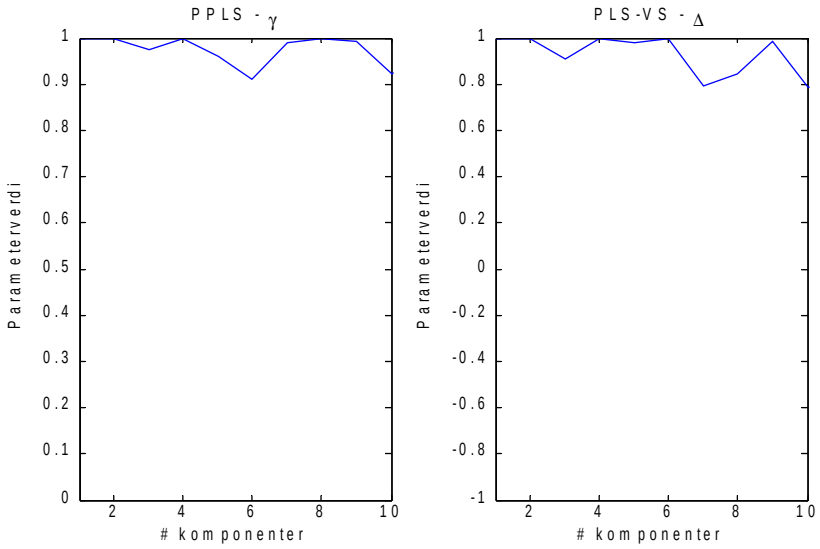
Modellene med lavest kryssvalideringsfeil for elastic net og lasso er derimot mer på høyde med PLS-metodene og PCR i forhold til prediksjonsfeil. Av figuren ser vi at modellen med lavest prediksjonsfeil også er den modellen for PPLS og PLS-VS som gir oss modeller med den laveste prediksjonen på testsettet. Modeller med flere komponenter har stor variasjon i prediksjonsfeil, spesielt for PLS-VS, men vi kan også se de samme tendensene i PPLS, noe som kan tyde på at metodene er mer ustabile med flere komponenter.

PPLS, PLS-VS, lasso og elastic net gir for den andre responsen svært enkle modeller, og regresjonskoeffisientene er svært like hverandre, mens PLS og PCR gir kompliserte modeller. Allikevel gir PCR og PLS modellene størst regresjonskoeffisienter til forklaringsvariablene de andre metodene fokuserer sine modeller på. Det tyder på at metodene finner de samme underrommene for prediksjon av responsen, men i tillegg mange andre underrom. PPLS, PLS-VS, lasso og elastic net er mer «presise» og trenger et mindre underrom enn de andre metodene for å oppnå god prediksjon.



Figur 6.12: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den andre responsen.*

Av figur 6.13 ser vi at PPLS og PLS-VS for de 5 første komponentene velger variabler til komponentene med omtrent identiske egenskapene for den andre responsen. I de to første komponentene utfører metodene ren variabelseleksjon på variablene med størst korrelasjon med responsvariabelen, og i den tredje komponenten brukes svært høyt korrelerte variable. Modellene som velges i modellutvelgelsen har kun tre komponenter, men vi kan allikevel se at begge metodene fokuserer komponentene mot høyt korrelerte variable opp til omkring den 10. komponenten.



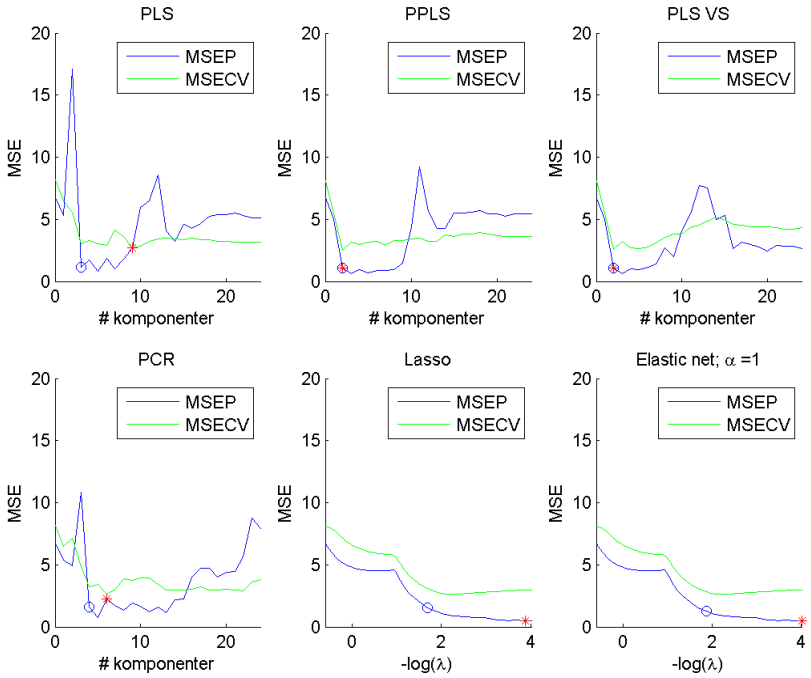
Figur 6.13: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den andre responsen.*

Modellene som velges med lavest feil i kryssvalidering og som den enklere modellen har begge 3 komponenter for PPLS og PLS-VS og vi kan lett identifisere bidragene fra hver komponent i regresjonskoeffisientene til modellene (Figur 6.12). Her representerer de to toppene bidraget fra de to første komponentene i modellen, og den lavere og bredere toppen bidraget fra den tredje komponenten.

6.4.3 Melinnhold – tredje respons

For den tredje responsen endrer prediksjonsfeilen for modellene til PPLS og PLS-VS ganske likt for hver komponent som blir med i modellen (Figur 6.14). Prediksjonsfeilen for modellene med økende kompleksitet følger hverandre ganske tett opptil omkring 10 komponenter. Dette gir en god indikasjon på at PLS-VS estimerer modeller på svært lik måte som PPLS for denne responsen, og at de trolig finner mye av de samme underrommene for prediksjon av responsen. Når vi kommer til figuren av de optimaliserte parameterne skal vi også se at dette er tilfellet.

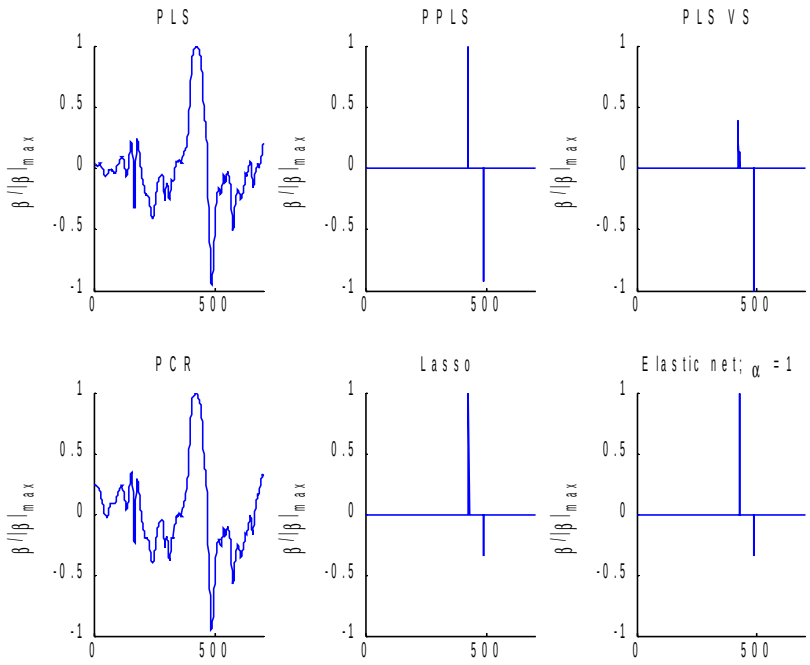
PLS og PCR skiller seg for denne responsen ut med dårligst prediksjon for modellene med lavest kryssvalideringsfeil, mens lasso og elastic net (modellene er identiske da den optimale verdien for $\alpha = 1$) gir oss de dårligste modellene for den enklere modellen. På samme måte som for den andre responsen blir de enklere modellene for enkle til å gjøre god prediksjon for elastic net og lasso, mens modellene med lavest kryssvalideringsfeil er like gode som de vi får med de andre metodene.



Figur 6.14: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den tredje responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

Likhetene mellom PPLS og PLS-VS kan også kjennes igjen i at regresjonskoeffisientene til modellene (Figur 6.15) er svært like, og for denne responsen er modellene svært like de som lasso og elastic net produserer. PLS og PCR produserer mye mer kompliserte modeller, men har tydelige topper på regresjonskoeffisientene omkring de samme variablene de andre metodene fokuserer modellene sine på.

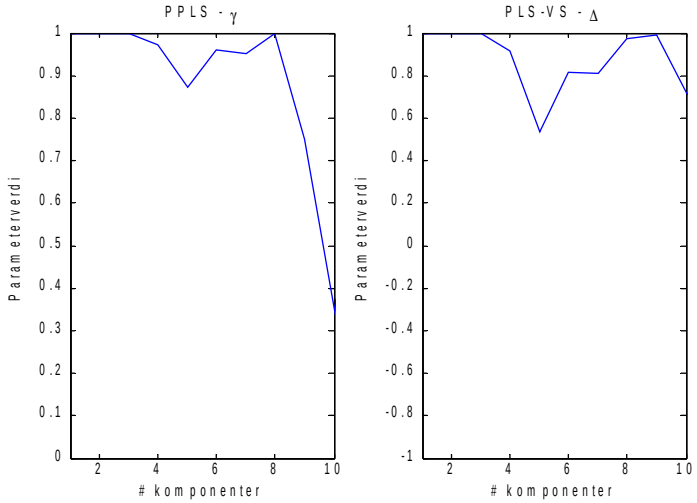
PPLS, PLS-VS, lasso og elastic net produserer svært enkle modeller med henholdsvis to, fem, tre og tre variabler i modellene for denne responsen. Variablene som velges er de samme eller de ligger de svært nær hverandre (metodene med valgte forklaringsvariable i modellen i parentes: PPLS (424 og 489), PLS-VS (423, 424, 425, 427 og 489), lasso og elastic net (427, 488 og 489)). Fra arbeid med NIR data vet vi fra tidligere at nærliggende variabler ofte har høy korrelasjon, og dermed finner modeller som inkluderer nabovariable i modellen omtrent de samme underrommene for å predikere responsen.



Figur 6.15: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den tredje responsen.*

At PPLS og PLS-VS finner liknende underrom får vi også bekreftet da potenserings- og trunkerings-parameteren velger forklaringsvariable med svært like egenskaper for de første komponentene (Figur 6.16). De tre første komponentene utføres ren og tilnærmet ren variabelseleksjon mot høyt korrelerte variable, og også i den fjerde komponenten er det svært høyt korrelerte forklaringsvariable som er med på å danne komponentene. For modeller med

flere komponenter avviker egenskapene til forklaringsvariablene som danner komponentene noe mer, men man kan allikevel antyde at til og med den åttende komponenten er svært lik for PPLS og PLS-VS!

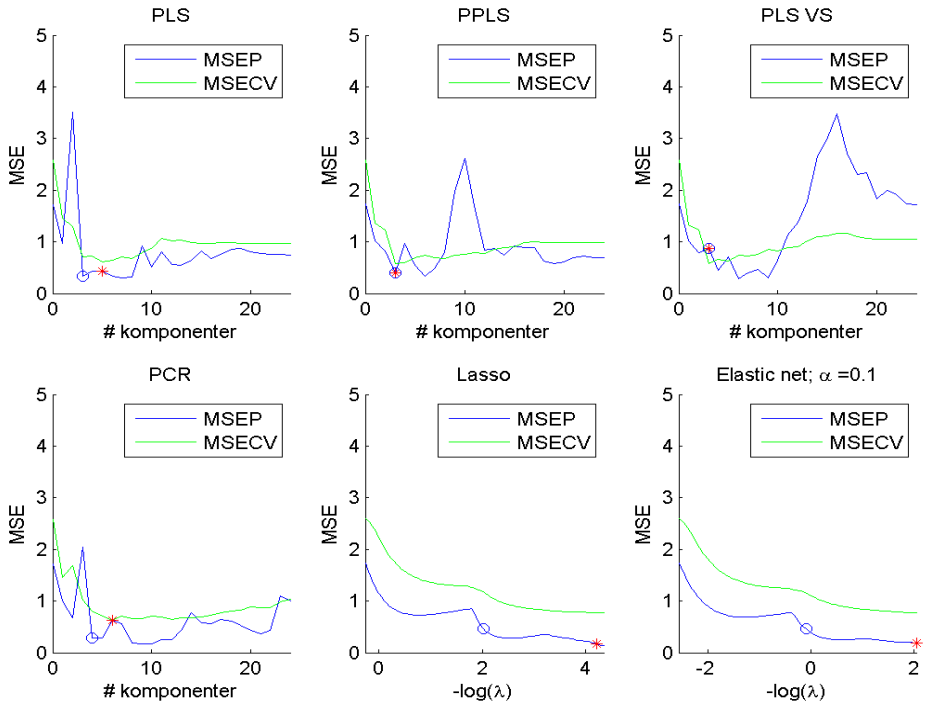


Figur 6.16: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den tredje responsen.*

6.4.4 Vanninnhold – fjerde respons

For den fjerde responsen har PLS-VS dårligst prediksjonsfeil for både modellen med lavest kryssvalideringsfeil og den enklere modellen mens lasso har lavest prediksjonsfeil for modellene med lavest kryssvalideringsfeil og PCR har lavest for de enklere modellene.

Prediksjonsfeilen i kryssvalideringen ser ut for flere av metodene til ikke å gi et helt riktig inntrykk for hvordan prediksjonsfeilen på det uavhengige testsettet blir i forhold til modellkompleksitet. Lasso og elastic net har to tydelige minimum på prediksjonsfeilen på testsettet, men vi oppdager kun det ene i kryssvalideringen. Modellen for lasso og elastic net med lavest kryssvalideringsfeil kommer svært nær det ene minimumet, og dette er også modellene med lavest prediksjonsfeil for denne responsen for alle metodene.



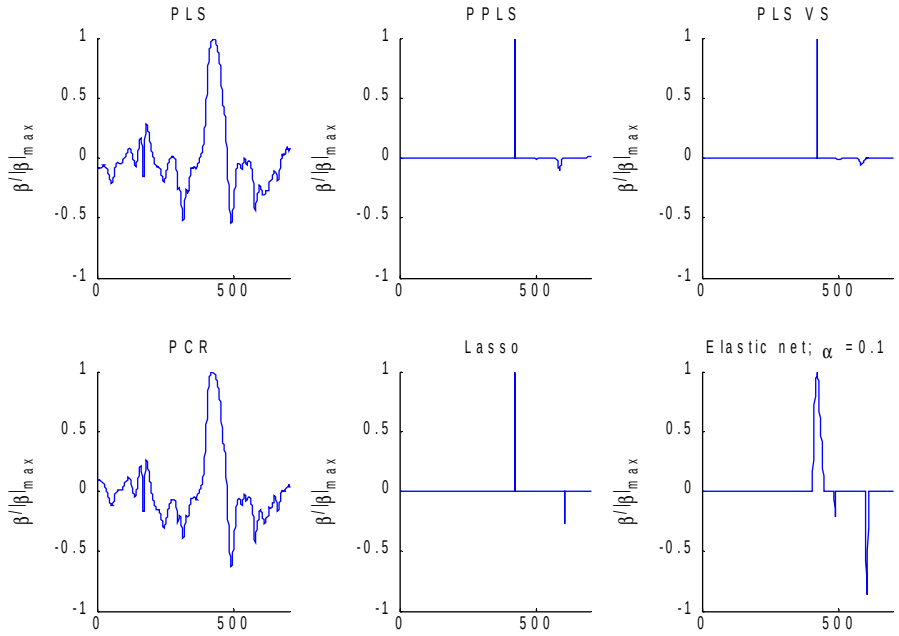
Figur 6.17: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den fjerde responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

For PLS-metodene og PCR ser vi at prediksjonsfeilen på testsettet avviker i større grad fra prediksjonsfeilen i kryssvalideringen for modeller med ulik kompleksitet. Dette tyder på at metodene produserer ustabile modeller for denne responsen. PLS-VS og PPLS får også en svært drastisk økning i prediksjonsfeil for modeller som har inkludert mer enn åtte komponenter i modellen, men disse blir heldigvis ikke valgt i modellutvelgelsen. Dette kan

tyde på at man ikke bør inkludere for mange komponenter i modellen for PPLS og PLS-VS, da det kan være fare for overtilpasning av modellen.

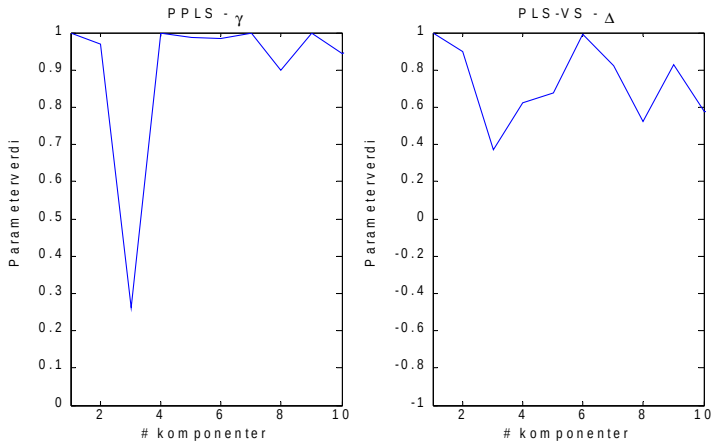
Fra teorien har vi at PPLS har mulighet til å spisse modellene mot enkelte variabler, og gi lav vekt til variabler med antatt liten betydning for modellen. Lasso har også denne spissingen mot enkeltvariabler som fasett i metoden men den velger å trunkere vekten av uviktige variabler til 0. PLS-VS har et lignende kriterie til PPLS med spissing av variabler, men i tillegg trunkeres koeffisientene til uviktige variabler til 0. Elastic net åpner for å produsere mer stabile modeller ved å inkludere korrelerte variable i modellen (for NIR datasett kjenner vi til at nærliggende variabler ofte er høyt korrelerte) for å gi bedre prediksjonen.

Modellene vi får for den fjerde responsen står nesten frem som et skoleeksempel over hvordan metodene produserer modeller (Figur 6.18). Her ser vi at PPLS, PLS-VS og lasso spisser modellene mot enkeltvariabler, mens elastic net lager en modell der nærliggende (og korrelerte) forklaringsvariable er inkludert for å øke stabiliteten. Modellen til PPLS har inkludert samtlige forklaringsvaribler i modellen, men det er tre tydelige områder der regresjonskoeffisientene er relativt store. Modellen PLS-VS produserer har trunkert 20.9 % av regresjonskoeffisientene til 0, lasso har trunkert 99.7 % og elastic net har trunkert 91.7 % Vi ser også at modellene som produseres fra disse metodene er svært mye enklere enn de vi får fra PLS og PCR, der alle variablene er inkludert i modellen og mange har middels til høye regresjonskoeffisenter. Dette gjør at tolkningen til modellene PLS og PCR produserer er mer komplisert, mens de andre metodenes modeller forteller oss at det er to eller tre viktige områder med viktige variabler for å kunne gi god prediksjon.



Figur 6.18: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den fjerde responsen.*

Av figur 6.19 ser vi at de to første komponentene blir laget av lineærkombinasjoner som fokuserer på forklaringsvariable med relativt like egenskaper. Her er det høyt korrelerte forklaringsvariable som preger komponentene, men for de neste komponentene avviker parameter i større grad. Dette kan trolig også forklare prediksjonsfeilen for PPLS og PLS-VS ikke endres like likt med økende kompleksitet for denne responsen, slik vi så i figur 6.17.



Figur 6.19: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den fjerde responsen.*

Datasett	Metode	Antall komponenter/ λ / α, λ	Prediksjonsfeil (MSEP)	Korrelasjon	Antall koeffisienter lik 0
Deig (fett)	PLS	9 (12)	0.3609 (0.1664)	0.9744 (0.9915)	0 (0)
Deig (fett)	PPLS	5 (9)	5.3005 (0.1149)	0.7640 (0.9908)	0 (0)
Deig (fett)	PLS-VS	4 (6)	2.6534 (4.4800)	0.8267 (0.8285)	110, 15.7 % (109, 15.6%)
Deig (fett)	PCR	8 (20)	0.7729 (0.2539)	0.9241 (0.9764)	0 (0)
Deig (fett)	Lasso	0.0271 (0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7 %)
Deig (fett)	Elastic net	1, 0.0271 (1, 0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7 %)
Deig (sukker)	PLS1	3 (6)	2.8682 (1.1932)	0.9151 (0.9764)	0 (0)
Deig (sukker)	PPLS1	3 (-)	1.5637 (-)	0.9608 (-)	185, 26.4% (-)
Deig (sukker)	PLS1/m VS	3 (-)	1.3921 (-)	0.9619 (-)	676, 96.6% (-)
Deig (sukker)	PCR	5 (6)	2.8221 (1.4430)	0.9353 (0.9710)	0 (0)
Deig (sukker)	Lasso	0.3420 (0.0612)	6.2286 (1.3791)	0.8324 (0.9553)	697, 99.6 % (687, 98.1 %)
Deig (sukker)	Elastic net	0.9, 0.3627 (0.9, 0.0680)	6.0623 (1.4022)	0.8400 (0.9549)	674, 96.3 % (666, 95.1 %)
Deig (mel)	PLS1	3 (9)	1.1578 (2.7042)	0.9251 (0.8832)	0 (0)
Deig (mel)	PPLS1	2 (-)	1.0779 (-)	0.9288 (-)	698, 99.7% (-)
Deig (mel)	PLS1/m VS	2 (-)	1.0560 (-)	0.9296 (-)	695, 99.3% (-)
Deig (mel)	PCR	4 (6)	1.5893 (2.2612)	0.8827 (0.9159)	0 (0)
Deig (mel)	Lasso	0.2578 (0.1116)	2.4374 (0.9363)	0.8381 (0.9333)	697, 99.5 % (695, 99.3%)
Deig (mel)	Elastic net	1, 0.2578 (1, 0.1116)	2.4374 (0.9363)	0.8381 (0.9333)	697, 99.5 % (695, 99.3 %)
Deig (vann)	PLS1	3 (5)	0.3365 (0.4242)	0.9539 (0.9271)	0 (0)
Deig (vann)	PPLS1	3 (-)	0.3914 (-)	0.9188 (-)	0 (-)
Deig (vann)	PLS1/m VS	3 (-)	0.8710 (-)	0.9214 (-)	146, 20.9 % (-)
Deig (vann)	PCR	4 (6)	0.2811 (0.6217)	0.9521 (0.9096)	0 (0)
Deig (vann)	Lasso	0.1329 (0.0149)	0.4637 (0.1682)	0.8820 (0.9610)	698, 99.7 % (686, 98 %)
Deig (vann)	Elastic net	0.1, 1.1030 (0.1, 0.1298)	0.4590 (0.1932)	0.8838 (0.9542)	642, 91.7 % (539, 77%)

Tabell 6.1: *Resultater for den enklere modellen fra de ulike metodene på det uavhengige testsettet på deigdatasettet. Resultatene for modellen med lavest feil i kryssvalideringen er i parentes. I de tilfellene der det ikke finnes noen enklere og ikke signifikant forskjellig modell fra den med lavest feil i kryssvalideringen markeres dette med (-).*

6.4.5 Oppsummering deigdatasettet:

For deigdatasettet har ingen av metodene skilt seg spesielt ut med bedre prediksjon av responsen. Her har samtlige metoder produsert modeller med svært god prediksjon for noen av responsene, men middels prediksjon for de andre.

I forhold til tolkbarhet har PLS og PCR produsert modeller der samtlige forklaringsvariable er inkludert i modellen, og flere av disse har relativt middels til høy vekting. Derimot produserer PPLS, PLS-VS, lasso og elastic net enkle modeller der det er rettet fokus mot enkelte områder for variablene. De tre siste metodene utfører variabelseleksjon for alle responsene, mens PPLS gjør det for to av responsene.

For dette datasettet har det vist seg at elastic net og lasso uten unntak har dårligere prediksjon av responsen for den enklere modellen sammenlignet med modellen med lavest kryssvalideringsfeil. For to av responsene er forskjellen liten, men for de to andre går metodene fra å ha middels til dårligst prediksjon sammenlignet med de andre metodene. Dette tyder på at vi sjelden får modeller med bedre prediksjon av responsen dersom vi velger enklere modeller enn den med lavest kryssvalideringsfeil. For disse metodene gir kryssvalideringen et godt bilde av prediksjonsfeilen til en modell med en gitt modellkompleksitet på nye data.

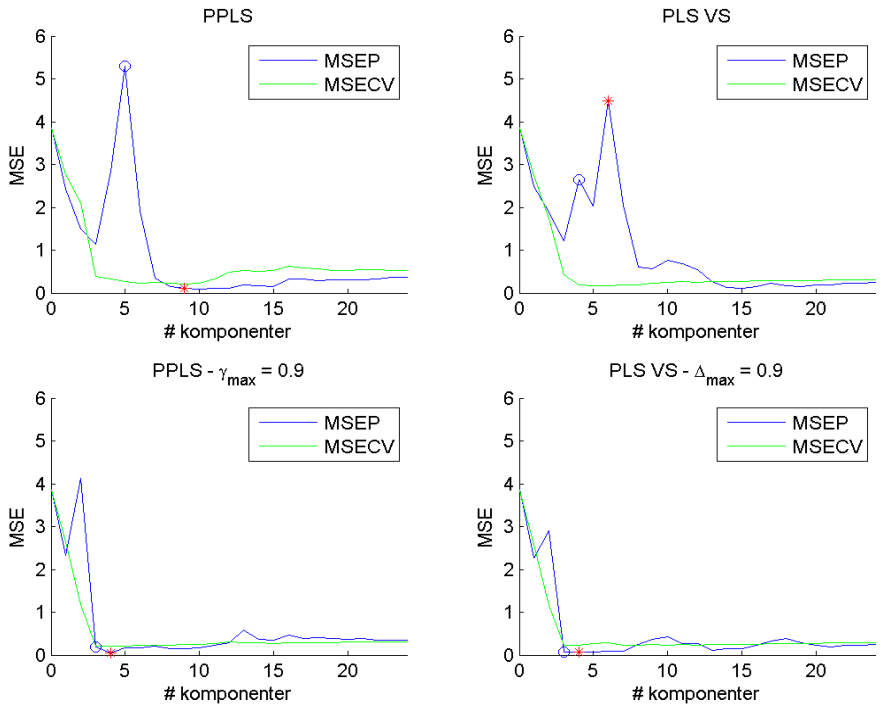
Regresjonskoeffisientene til modellene som PLS-VS og PPLS produserer er svært like for de fire responsene. Dette tyder på at vi har klart å utvikle PLS-VS slik at den fungerer på noenlunde samme måte som PPLS. I tillegg selekterer metoden variable på samme måte som lasso og elastic net gjør. For en av responsene utfører metoden svært sterk seleksjon der 99.7 % av regresjonskoeffisientene ble satt lik 0. For de tre andre responsene var ikke seleksjonen like sterk, da ble mellom 15.7% og 26.4% av koeffisientene satt lik 0.

6.4.6 Eksplorativ analyse av den første responsvariabelen

I modelleringen foran så vi at for den første responsen produserte PPLS og PLS-VS modeller som predikterte vesentlig dårligere enn de andre metodene. Fra grafene for prediksjonsfeil mot kompleksitet kom det tydelig fram at de enklere modellene som ble valgt påfallende dårlig prediksjon i forhold til de øvrige metodene.

Felles for begge metodene var at den siste komponenten som ble inkludert i modellene og som bidro til den uventede økningen i prediksjonsfeilen skyldtes en tilnærmet variabelseleksjon på korrelerte variabler.

Det er derfor av interesse å se om en øvre begrensning på potenserings- og trunkerings-parameteren kan bidra til å oppnå mer. Vi undersøkte dette ved å innføre en øvre begrensning på potenseringsparameteren i PPLS og trunkeringsparameteren i PLS-VS. I begge tilfellene ble verdien 0.9 benyttet som største tillatte verdi.



Figur 6.20: *Prediksjonsfeil som funksjon av kompleksitet for PPLS og PLS-VS med og uten begrensning på paramaeterverdiene for den første responsen.*

Ved å innføre begrensninger på paramterne på PPLS og PLS-VS lot det seg gjøre å stabilisere metodene og få jevnere kurver for prediksjonsfeil målt mot modellkompleksistet. Som vist i figur 6.20 oppnås det gode modeller met et lavete antall komponenter. Prediksjonsfeilen for

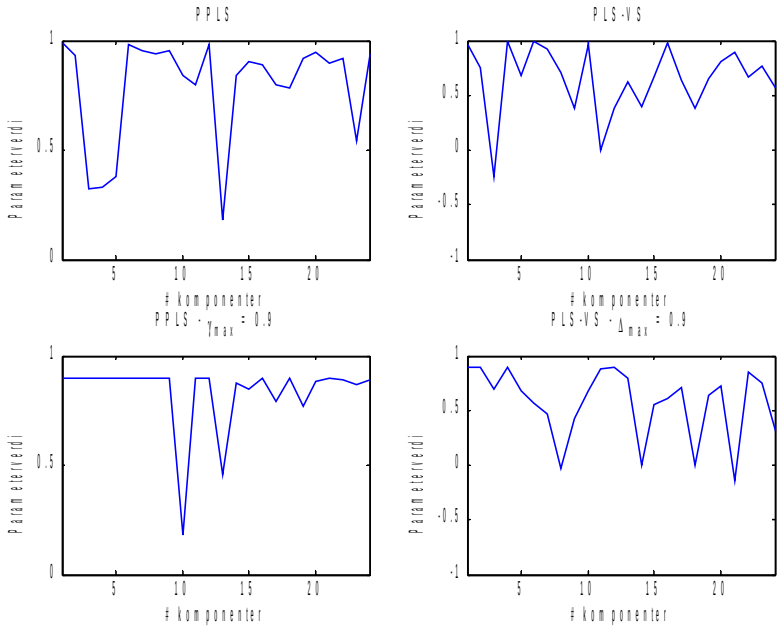
PPLS og PLS-VS med begrensning er her mye likere den vi får fra modellen til lasso og elastic net (Tabell 6.2). Når vi ser på den enkle modellene er prediksjonsfeilen for PPLS 0.1842, PLS-VS 0.0610 og for lasso og elastic net er den 0.1236. Ved å begrense parameterne for PPLS og PLS-VS utlignes altså den store forskjellen i prediksjonsfeil ved sammenligning med bruk av disse metodene uten begrensning på parameterverdiene.

Datasett	Metode	Antall komponenter/ λ / α, λ	Prediksjonsfeil (MSEP)	Korrelasjon	Antall koeffisienter lik 0
Deig (fett)	PLS1	9 (12)	0.3609 (0.1664)	0.9744 (0.9915)	0, 0 % (0)
Deig (fett)	PPLS	5 (9)	5.3005 (0.1149)	0.7640 (0.9908)	0, 0 % (0)
Deig (fett)	PLS1 VS	4 (6)	2.6534 (4.4800)	0.8267 (0.8285)	110, 15.7 % (109, 15.6 %)
Deig (fett)	PPLS (begrenset)	3 (4)	0.1842 (0.0519)	0.9943 (0.9937)	0, 0 % (0)
Deig (fett)	PLS1 VS (begrenset)	3 (4)	0.0610 (0.0662)	0.9956 (0.9947)	365, 52.4 % (347, 49.6 %)
Deig (fett)	PCR	8 (20)	0.7729 (0.2539)	0.9241 (0.9764)	0, 0 % (0)
Deig (fett)	Lasso	0.0271 (0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7%)
Deig (fett)	Elastic net	1, 0.0271 (1, 0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7%)

Tabell 6.2: *Utsnitt fra tabell og inkludering av resultater fra de begrensede modellene av PPLS og PLS-VS for den første responsen.*

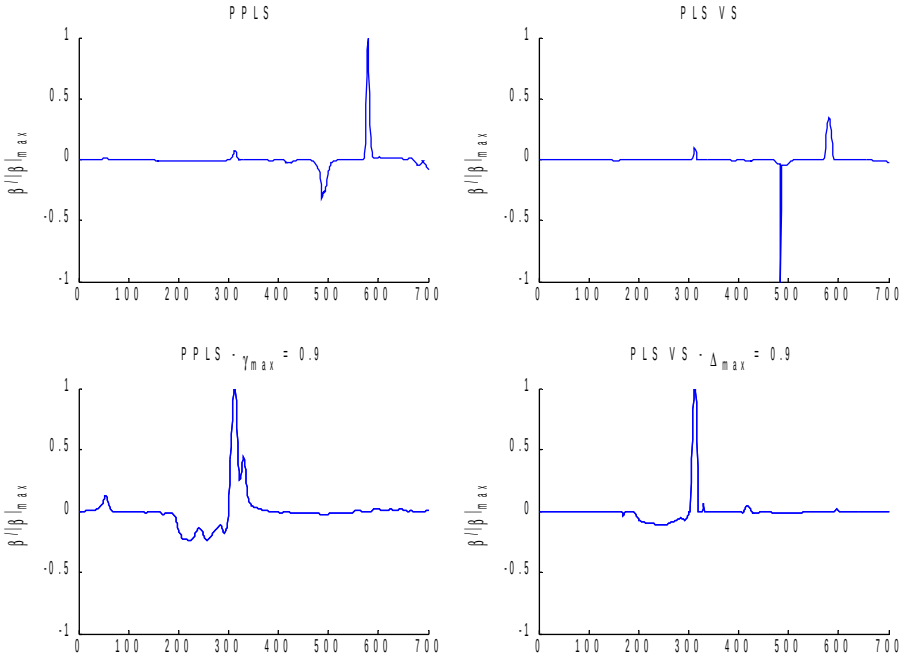
Begrensningen av parameterne bidrar til at det ikke lenger tillates spissing mot enkeltvariable i modellkomponentene som dermed tvinges til å inkludere flere variable som igjen bidrar til stabilisering av modellene.

Med parameterbegrensning er det fortsatt de høyest korrelerte variablene som blir valgt inn i modellene for PPLS og PLS-VS, men dette skjer ikke lenger med tilnærmet ren variabelseleksjon (Figur 6.21).



Figur 6.21: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS med og uten begrensning for hver komponent for den første responsen.*

Ved å sammenligne modellene generert med og uten parameterbegrensninger (Figur 6.22) ser vi at ikke de samme forklaringsvariablene fokuserer. I modellene fra PPLS og PLS-VS uten begrensning ser vi at variabler rundt nummer 300 kun har en liten topp, mens for modellene med begrensning later disse til å være de mest betydningsfulle variablene for modellen (Figur 6.22). Variablene som har størst vekt i modellene uten begrensning er ubetydelige når vi sammenligner mot modellene med begrensning. Modellene for PPLS og PLS-VS med begrensning samsvarer godt med modellene til lasso og elastic net (Figur 6.8) med fokus på variabler omkring nummer 300.



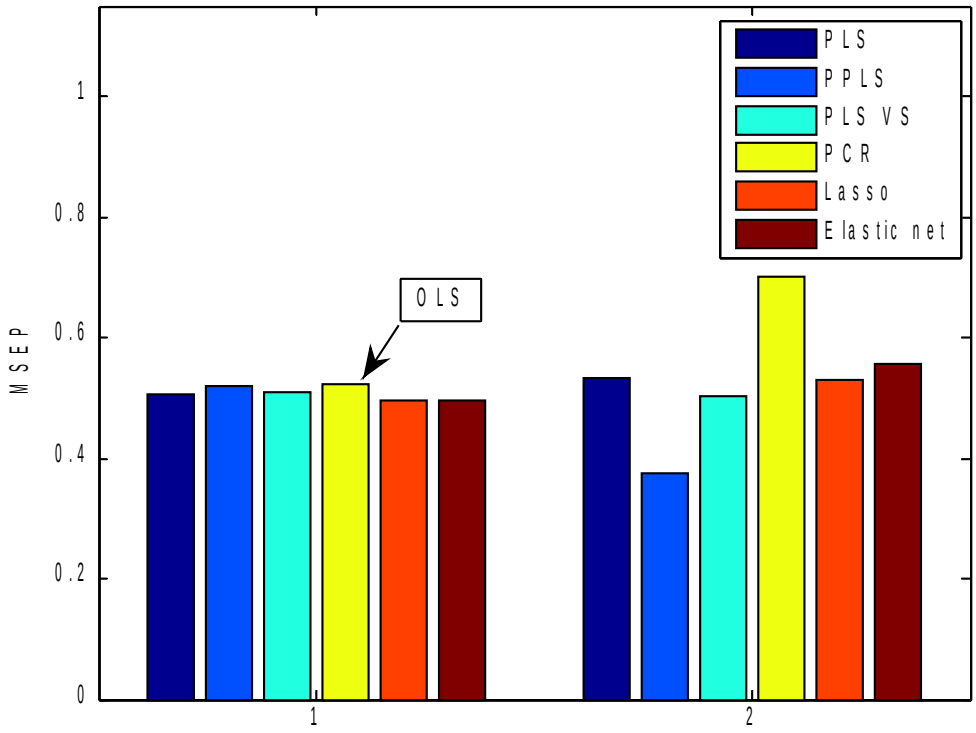
Figur 6.22: *Skalerte regresjonskoeffisienter den enklere modellen til PPLS og PLS-VS med og uten begrensning på parameterverdiene på den første responsen.*

Resultatet over illustrerer at muligheten med justering av området for tillatte parameterverdier kan være av interesse i praktiske anvendelser. I kjemometri og spektroskopi (, som dette datasettet er hentet fra,) der man ved hjelp av spektre er interessert i å kunne prediktere innholdet i en prøve er det sjelden dette kan gjenspeiles med kun at lys med en bestemt bølgelengde blir absorbert, men trolig lys fra et gitt område eller kombinasjoner av områder.

Ved å legge begrensninger på mulighetene for de høyest korrelerte forklaringsvariablene til å dominere i modellene, fordeler man også «ansvaret» på flere forklaringsvariabler. Årsaken til at en forklaringsvariabel skiller seg ut med høy korrelasjon med responsen kan være at den faktisk er svært relevant for prediksjonen, men det kan også komme av at tilfeldig variasjon eller støy i måleinstrumentet favoriserer denne variabelen. Ved å vektlegge flere korrelerte variabler i modellen kan man derfor oppnå stabile modeller som gir oss bedre prediksjoner.

6.5 Prostatadatasettet

I analysen av disse dataene klarte vi ikke å gjenskape tidligere resultater. I artikkelen der elastic net ble foreslått (Zou & Hastie 2005) utpekte metoden seg som den desidert beste sammenlignet med blant annet lasso og OLS. Med de fremgangsmåtene vi har brukt har vi ikke klart å gjenskape et like godt resultat. Med bruk av andre optimaliseringskriterier for å finne parameterne i elastic net kunne vi trolig fått like gode resultater som tidligere rapportert. Dette har vi imidlertid ikke vært i stand til, så den videre analysen vil basere seg på våre resultater. På den måten vil sammenligningsgrunnlaget være mest mulig rettferdig for de ulike metodene.

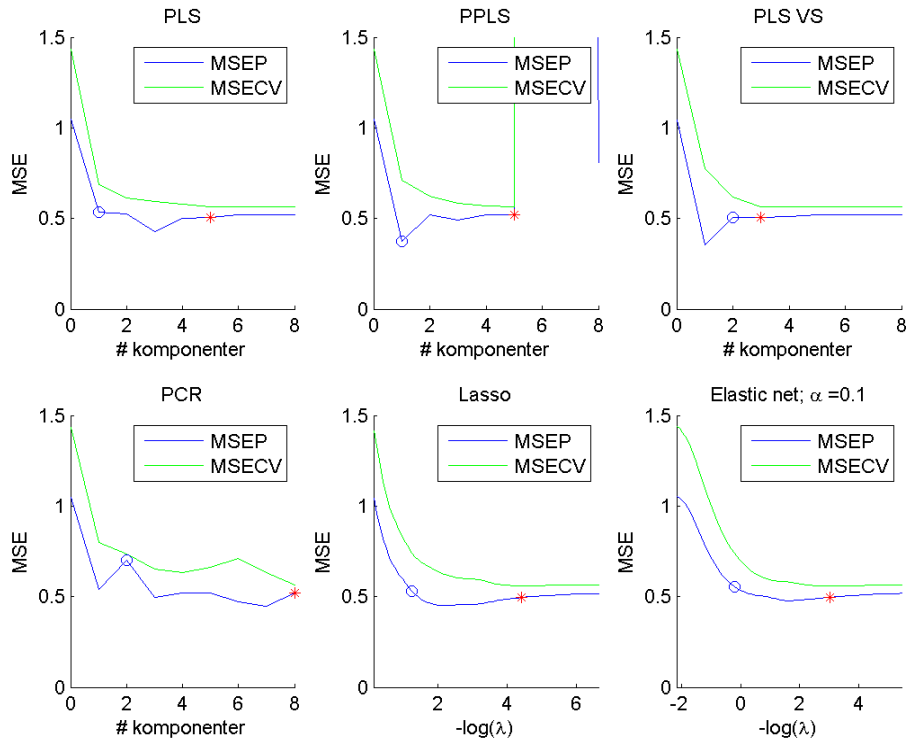


Figur 6.23: *Prediksjonsfeil (MSE) på testsettet for de ulike metodene. Til venstre (1) er resultatet fra modellene med lavest feil i kryssvalideringen, og til høyre (2) er resultatet for den enklere og ikke signifikant forskjellige modellen.*

Modellene som velges på grunnlag av lavest kryssvalideringsfeil gir omtrent like god prediksjonsnøyaktighet for de ulike metodene (Figur 6.23). Lasso har lavest prediksjonsfeil, tett fulgt av elastic net, men svært lite skiller de ulike metodene. Modellen med lavest kryssvalideringsfeil for PCR har inkludert alle komponentene i modellen (like mange komponenter som det er forklaringsvariabler), så prediksjonsfeilen tilsvarer prediksjonsfeilen

vi får ved med vanlig OLS regresjon. Derimot er det større sprik i prediksjon for de enklere modellene. Her gir PPLS lavest og PCR høyest prediksjonsfeil, mens de resterende metodene ligger noe spredt midt i mellom disse.

Prediksjonsfeilen i PPLS har en kraftig og uventet økning for modeller med mer enn fem komponenter for både testsettet og i kryssvalideringen (Figur 6.24). Trolig kommer dette av at de siste komponentene som blir beregnet i PPLS gjør beregningene på en nesten singulær X matrise. Beregningene er så krevende at de går utover datamaskinens numeriske nøyaktighet, som gjør at modellene vi beregner blir svært ustabile og unøyaktige. Prediksjonsfeilen for modellen til PPLS laget med seks komponenter på testsettet er eksempelvis i størrelsesorden 10^{25} (!). Heldigvis blir ikke modellene med seks, sju og åtte komponenter valgt i kryssvalideringen, slik at modellene vi vurderer ikke blir påvirket av dette. Hadde derimot denne typen ustabiliteter befunnet seg i en av de enklere modellene i forhold til de som ble valgt hadde det vært rom for bekymring. Dette tyder på at for PPLS må man generelt passe på å ikke inkludere for mange komponenter i en modell når man benytter PPLS i regresjon, og trolig gjelder dette da også for PLS-VS.

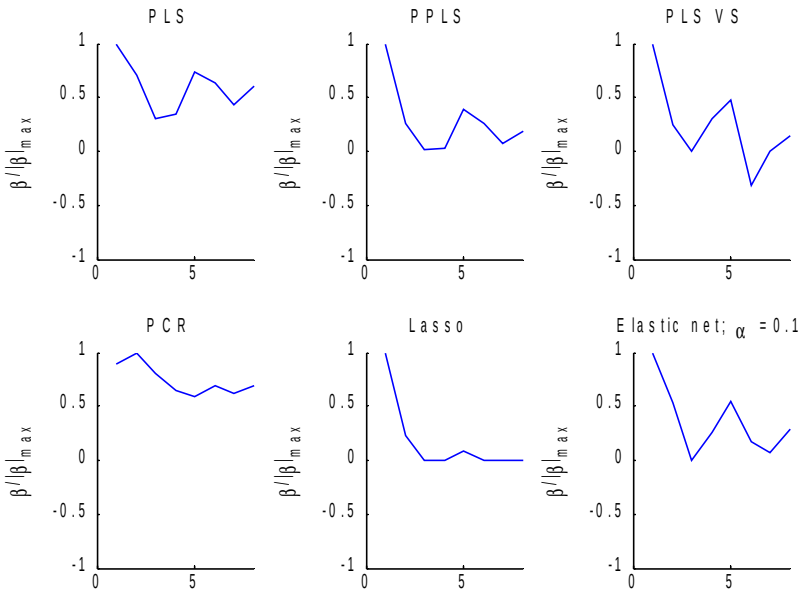


Figur 6.24: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

Med unntak av ustabilitetene til PPLS når modellene har mange komponenter har de ulike metodene nokså jevne og glatte kurver for prediksjonsfeil i kryssvalideringen. Prediksjonsfeilen blir mer ujevn i testsettet testsettet for alle metodene med unntak for lasso og elastic net. Disse metodene gir svært stabile modeller og har et tydelig minimum på prediksjonsfeil på testsettet for modeller med λ parameter mellom den valgt med lavest kryssvalideringsfeil, og den enklere modellen. Dette har vi imidlertid ingen anledning til å avdekke ved kun å se på kryssvalideringsfeil. Det ser imidlertid ikke ut til at et annet valgt av lambda-parameter kunne gitt oss like god prediksjon som det er vist i tidligere studier med våre optimaliseringskriterier, da den laveste feilen vi kunne oppnådd for elastic net der α er lik 0.1 ligger på 0.48.

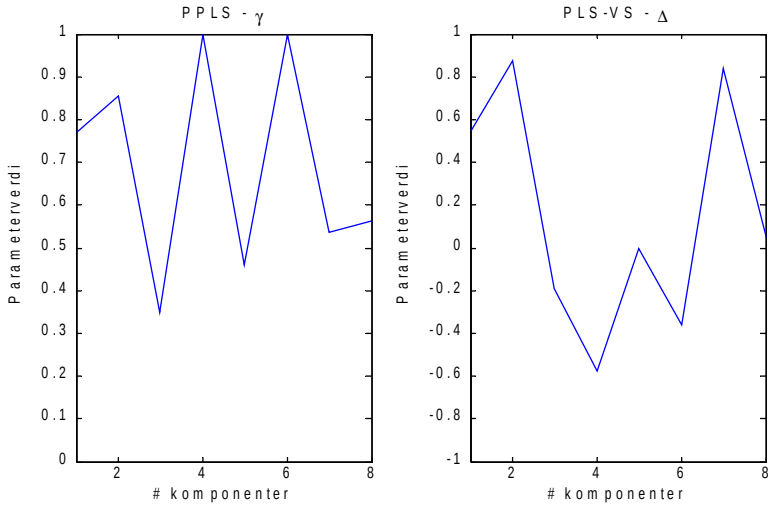
PLS-VS har et tydelig minimum på prediksjonsfeil på testsettet for modellen med en komponent, men i kryssvalideringen får vi ikke inntrykk av dette. Modellen med en komponent for PLS-VS viser at også denne metoden også har potensiale til like god prediksjon som for den enklere modellen i PPLS.

Ved studering av regresjonskoeffisientene til de ulike metodene kan det observeres at modellene har flere likhetstrekk med hverandre (Figur 6.25). Alle metodene med unntak av PCR vektlegger de to første forklaringsvariablene relativt høyt, samtidig som den femte forklaringsvariabelen blir vektet relativt høyere enn den fjerde og den sjette slik at kurven får en topp der. Av metodene som utfører variabelseleksjon er det lasso som gir den enkleste modellen, da alle forklaringsvariablene med unntak av en, to og fem er selektert bort. PLS-VS har den nest enkleste modellen da variabel tre og sju er selektert bort, og elastic net har selektert bort variabel tre.



Figur 6.25: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene.*

Parameterne for potenseringen i PPLS og trunkeringen for PLS-VS ser vi at følger hverandre ganske likt fram til og med den tredje komponenten (Figur 6.26). Her inkluderes henholdsvis forklaringsvariable som er middels høyt korrelert til responsen, høyt korrelert til responsen og til sist forklaringsvariable med høy kovarians med responsen. Med dette kan vi si at metodene finner omtrent de samme underrommene for prediksjon i de tre første komponentene. Dette gjenspeiles også i hvordan kurvene for prediksjonsfeilen på testsettet for de tre første komponentene for PPLS og PLS-VS er svært like og at regresjonskoeffisientene for de enklere modellene likner hverandre. I den fjerde komponenten inkluderes forklaringsvariable i med svært høy korrelasjon til responsen i PPLS, mens variable med middels høyt standardavvik blir inkludert i komponenten til PLS-VS. Avviket kan være et resultat av numeriske unøyaktigheter, for vi vet at allerede for den sjette komponenten er X matrisa nært singulær slik at modellen blir for ustabil til å kunne gi god prediksjon.



Figur 6.26: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent.*

Datasett	Metode	Antall komponenter/ λ / α, λ	Prediksjonsfeil (MSEP)	Korrelasjon	Antall koeffisienter lik 0
Prostata	PLS1	1 (5)	0.5334 (0.5055)	0.7108 (0.7208)	0 (0)
Prostata	PPLS1	1 (5)	0.3735 (0.5208)	0.8068 (0.7111)	0 (0)
Prostata	PLS1/m VS	2 (3)	0.5033 (0.5079)	0.7234 (0.7188)	2, 25 % (0)
Prostata	PCR	2 (8)	0.7010 (0.5213)	0.6103 (0.7108)	0 (0)
Prostata	Lasso	0.2878 (0.0122)	0.5297 (0.4952)	0.7664 (0.7282)	5, 62.5 % (1, 12.5 %)
Prostata	Elastic net	0.1, 1.2458 (0.1 0.0480)	0.5563 (0.4968)	0.7623 (0.7273)	1, 12.5% (1, 12.5%)

Tabell 6.3: *Resultater for den enklere modellen fra de ulike metodene på det uavhengige testsettet på prostatadatasettet. Resultatene for modellen med lavest feil i kryssvalideringen er i parentes. I de tilfellene der det ikke finnes noen enklere og ikke signifikant forskjellig modell fra den med lavest feil i kryssvalideringen markeres dette med (-).*

6.5.1 Oppsummering prostatadatasettet

For prostatadatasettet klarte vi ikke med våre optimaliseringskriterier å gjenskape den svært gode prediksjonen elastic net hadde sammenlignet med lasso og OLS (PCR modellen med lavest kryssvalideringsfeil hadde inkludert samtlige komponenter i modellen, som tilsvarer OLS modellen). Våre resultater ble at elastic net og lasso predikerte omtrent like godt, og PPLS predikerte best for den enklere modellen med en komponent. PLS-VS hadde også svært

god prediksjon for komponenten med en modell, men denne ble ikke valgt i kryssvalideringen. Allikevel ser vi at det er et potensiale for metoden til å gjøre like gode prediksjoner som PPLS.

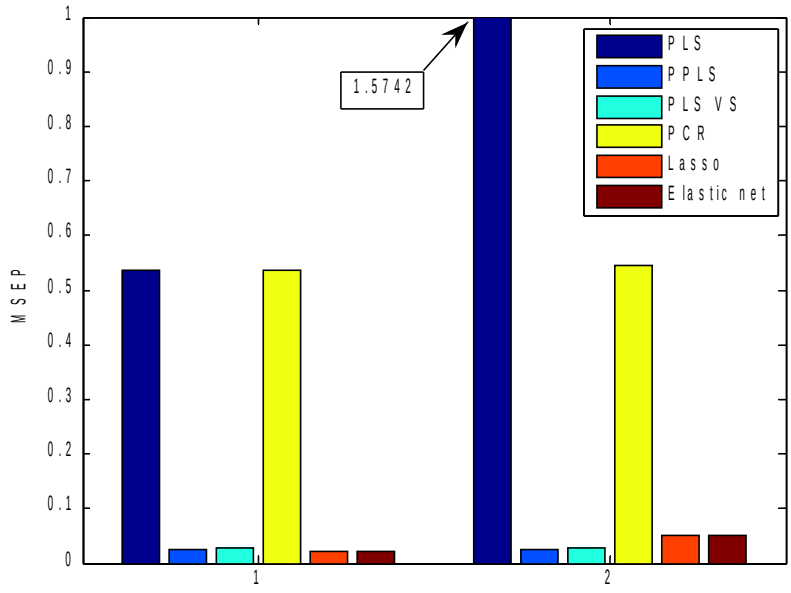
PPLS og PLS-VS gir liknende modeller der det har blitt inkludert opptil tre komponenter. For modeller med flere komponenter avviker metodene i større grad, men dette kan også komme av numerisk unøyaktighet. Av figur 6.24 vet vi at PPLS lider av numerisk unøyaktighet etter 5 komponenter, da metoden produserer ustabile modeller. Men med utgangspunkt i at PPLS og PLS-VS inkluderer like forklaringsvariable i de tre første komponentene og at PLS-VS også utfører automatisk variabelseleksjon på regresjonskoeffisientene, kan det tyde på at metoden fungerer slik vi ønsker.

I forkant av analysen ble datasettet preprosessert med at hver forklaringsvariabel ble standardisert, slik at hver forklaringsvariabel var sentrert og hadde varians lik 1. Dette medførte at variabel 8 og variabel 3 fikk redusert sin varians betraktelig, og med det unngikk vi at disse variablene kom til å være fremtredende i de første komponentene til PLS og PCR. I vår analyse valgte vi henholdsvis modeller med en og to komponenter for PLS og PCR sine enklere modeller, og det er like mange komponenter PPLS og PLS-VS har i sine modeller (Tabell 6.3). Det er fra tidligere kjent at variabler med høy varians ofte kan bli høyt vektet i de første komponentene vi får fra PLS og PCR. En testkjøring (resultatene er ikke tatt med) viser at på det ikke-standardiserte datasettet har PLS, PPLS, PLS-VS og PCR henholdsvis tre, en, en og fem komponenter i den enklere og ikke signifikant forskjellige modellen. Dette tyder på at standardiseringen av datasettet kan bidra til at PLS og PCR sine første komponenter inneholder relevant informasjon for prediksjon.

6.6 Øldatasettet

I analysen av disse dataene fikk vi svært god prediksjon for de fleste metodene, mens enkelte metoder skilte seg ut med svært dårlig prediksjon. For dette datasettet ser det ut til at våre resultater samsvarer godt med det man tidligere har fått (Indahl 2005), og vi har mye bedre samsvar her enn for deigdatasettet. Prediksjonsfeilen som er rapportert er roten av MSE og det kan vises ved omregning at feilen for PLS modellen er 1.8879 og for PPLS modellen 0.0299 for de enklere og ikke signifikant forskjellige modellene. Sammenligner vi dette med våre resultater ser vi at også vi får frem at PPLS har mye lavere prediksjonsfeil enn PLS (tabell 6.4).

For PPLS, PLS-VS, lasso og elastic net (modellen for lasso og elastic net er identiske da elastic net får laveste kryssvalideringsfeil for modellen der $\alpha = 1$ som for oss tilsvarer lasso-metoden) får vi omtrent like god prediksjon, mens PLS og PCR ikke klarer å produsere gode nok modeller til å klare like god prediksjon (Figur 6.27). For modellene valgt med lavest kryssvalideringsfeil gir lasso og elastic net best prediksjon, men for de enklere modellene er det PPLS som predikerer best med PLS-VS rett bak.

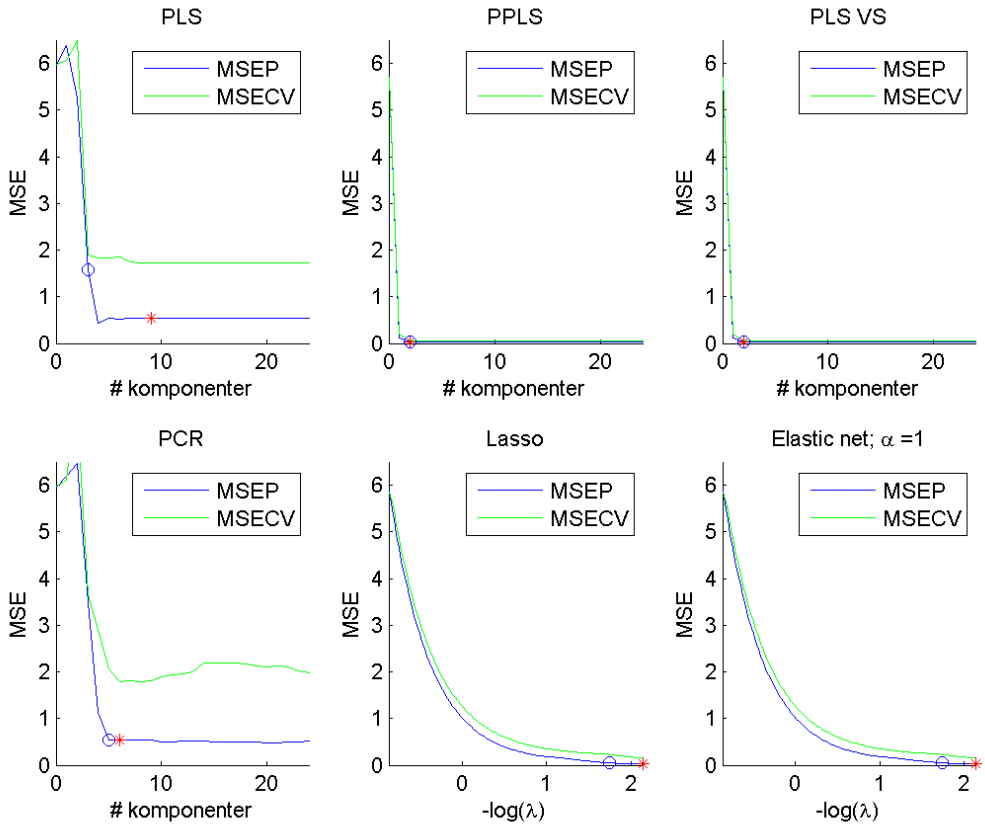


Figur 6.27: *Prediksjonsfeil (MSE) på testsettet for de ulike metodene. Til venstre (1) er resultatet fra modellene med lavest feil i kryssvalideringen, og til høyre (2) er resultatet for den enklere og ikke signifikant forskjellige modellen.*

Av figur 6.28 ser vi at PLS og PCR ikke har mulighet til å oppnå like lav prediksjonsfeil som de andre modellene for PLS og PCR har lite å hente på å inkludere flere komponenter i

modellene da prediksjonsfeilen på er ganske lik for modeller med flere komponenter (Figur 6.28), men PLS kunne fått noe bedre prediksjon dersom modellen med 4 komponenter hadde blitt valgt. Uansett er resultatene for disse metodene langt dårligere sammenlignet med PPLS, PLS-VS, lasso (og elastic net). Dette tyder på at PLS og PCR ikke klarer å oppdage de underliggende strukturene i dette datasettet på samme måte som de andre metodene. PPLS, PLS-VS og lasso klarer å avdekke mer av de underliggende strukturene i datasettet, og får til bedre prediksjon.

Prediksjonsfeilen i forhold til kompleksitet ser ut til å være forholdsvis lik i kryssvalideringen som på testsettet for PPLS, PLS-VS, lasso og elastic net. Dette gjør at kurvene for prediksjonsfeilen for metodene blir ganske jevne. Det ser ikke ut til at prediksjonsfeilen gjør noen uventede endringer, noe som kan tyde på at modellene som lages er ganske stabile.



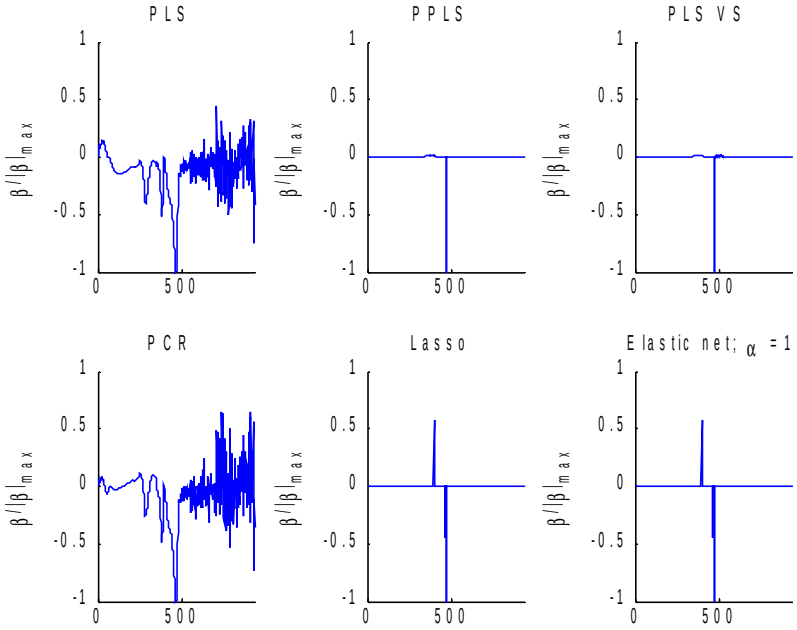
Figur 6.28: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

PPLS og PLS-VS trenger kun to komponenter for å finne modellen med lavest kryssvalideringsfeil mens PLS og PCR trenger henholdsvis ni og 6 komponenter. Det viser seg imidlertid at enklere modeller som ikke er signifikant forskjellig finnes for både PLS og

PCR med tre og fem komponenter, mens det ikke finnes enklere modeller for PPLS og PLS-VS. Dette bekrefter det vi vet om at PPLS og PLS-VS er i stand til å finne gode komponenter for prediksjon tidlig i algoritmen, mens PLS og PCR trenger flere komponenter for å oppnå like god prediksjon.

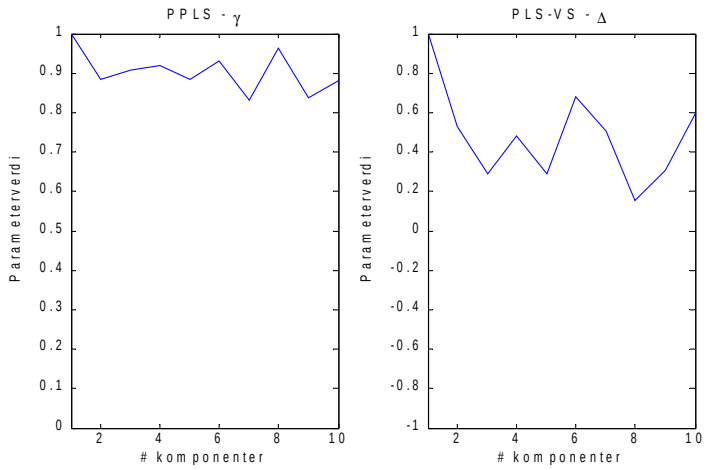
For dette datasettet kan vi igjen se at PPLS, PLS-VS, lasso og elastic net klarer å avdekke de underliggende strukturene i datasettet, da metodene produserer modeller der den samme forklaringsvariabelen har høyest regresjonskoeffisient (Figur 6.29). På lik linje med lasso og elastic net selekterer PLS-VS ut variabler (Tabell 6.4) og gir en modell med få forklaringsvariabler. Lasso og elastic net har de enkleste modellene med henholdsvis 8 og 3 forklaringsvariabler (over 99 % av forklaringsvariablene er selektert ut av modellen) i modellen med lavest kryssvalideringsfeil og modellen som er enklere og ikke signifikant forskjellig. Modellen fra PLS-VS (modellen med lavest kryssvalideringsfeil er også den enkleste modellen som blir valgt) inneholder 150 forklaringsvariabler, man allikevel selekteres mange bort (83.8 %). Derimot selekterer PPLS 6 forklaringsvariabler ut av modellen. Regresjonskoeffisientene blir allikevel svært lik de andre metodenes på grunn av at mange av regresjonskoeffisientene i PPLS er svært små. Her kommer det tydelig fram at PLS-VS gjør jobben med å selektere bort variabler automatisk, og oppfører seg som en hybrid mellom PLS-metodene og krympingsmetodene.

Modellene som blir produsert av PLS og PCR gir vekt til alle variabler, og dette er igjen et tegn på at metodene ikke i like stor grad som de andre metodene klarer å finne de underliggende strukturer i datasettet. Regresjonskoeffisientene til PLS og PCR likner svært mye på hverandre, og er mye mer komplekse sammen sammenlignet med modellene til de ander metodene. Det går allikevel an å merke seg at PLS og PCR gir høyest regresjonskoeffisient til den samme variablene som de andre metodene også velger.



Figur 6.29: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene.*

Studerer vi parameterne for PPLS og PLS-VS ser vi at for de to første komponentene lages disse av lineærkombinasjoner av forklaringsvariable med svært like egenskaper. Den første komponenten består i hovedsak av variabelseleksjon av høyt korrelerte forklaringsvariable for begge metodene, og den i andre komponenten er det fokusering på variabler med middels til høy korrelasjon med responsvariabelen. Dette lar seg lett identifiseres i regresjonskoeffisientene til modellene fra PPLS og PLS-VS. Her har vi tydelig én variabel som skiller seg ut som ble selektert i første komponent, og i den andre komponenten har variabler litt før og rett etter blitt selektert. For modeller med flere komponenter avviker metodene i større grad da PPLS bygger komponentene med hensyn på høyt korrelerte variabler i modellene, mens PLS-VS også inkluderer middels korrelerte variabler i komponentene.



Figur 6.30: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent.*

Datasett	Metode	Antall komponenter/ λ / α, λ	MSEP	Korrelasjon	Antall koeffisienter lik 0
Øl	PLS1	3 (9)	1.5742 (0.5377)	0.8586 (0.9611)	0 (0)
Øl	PPLS1	2 (-)	0.0240 (-)	0.9981 (-)	6, 0.6 % (-)
Øl	PLS1/m VS	2 (-)	0.0274 (-)	0.9978 (-)	776, 83.8 % (-)
Øl	PCR	5 (6)	0.5464 (0.5362)	0.9566 (0.9579)	0 (0)
Øl	Lasso	0.1758 (0.1186)	0.0498 (0.0234)	0.9974 (0.9990)	923, 99.7 % (918, 99.1 %)
Øl	Elastic net	1, 0.1758 (1, 0.1186)	0.0498 (0.0234)	0.9974 (0.9990)	923, 99.7 % (918, 99.1 %)

Tabell 6.4: *Resultater for den enklere modellen fra de ulike metodene på det uavhengige testsettet på øldatasettet. Resultatene for modellen med lavest feil i kryssvalideringen er i parentes. I de tilfellene der det ikke finnes noen enklere og ikke signifikant forskjellig modell fra den med lavest feil i kryssvalideringen markeres dette med (-).*

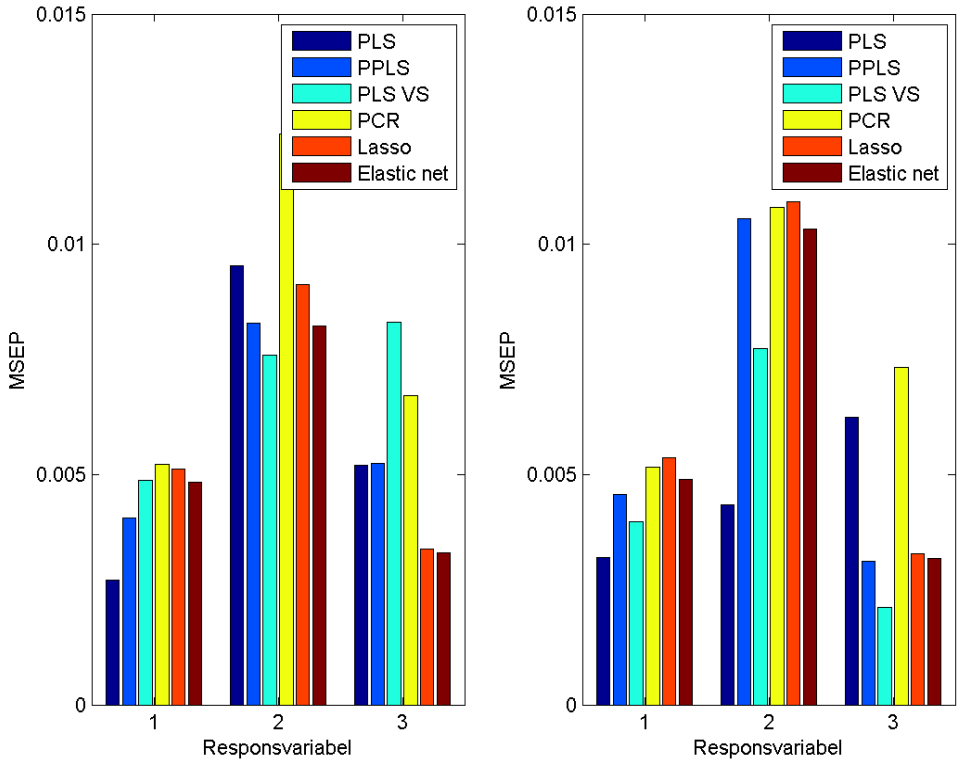
6.6.1 Oppsummering øldatasettet:

Resultatene på øldatasettet tyder på at metodene som har fasetten at de kan spisse modellene mot enkeltvariable gjør bedre prediksjon enn PLS og PCR. PPLS, PLS-VS, lasso og elastic net gir alle størst regresjonskoeffisient til den samme variabelen i de enklere modellene, og projiserer seg da ned i det samme underrommet for å gjøre god prediksjon for dette datasettet. PLS-VS viser at den på dette datasettet er i stand til å selektere variabler på lik måte som lasso og elastic net, da over 80% av forklaringsvariablene får regresjonskoeffisient lik 0 for den enklere modellen. Vi ser også at parameterne for PPLS og PLS-VS følger hverandre mest for de første komponentene, og deretter blir forskjellene større og større.

6.7 MALDI-TOF-datasettet

I analysen av disse dataene har vi valgt å utelate post prosesseringen som er gjort i tidligere studier. Vi er primært interessert i å sammenligne de ulike metodene og se på hvordan de fungerer i forhold til hverandre. Våre resultater blir derfor noe annerledes enn det som tidligere er rapportert (Liland et al. 2009). I praktisk bruk av metodene vil derimot prosesseringen være høyst aktuell for å forsøke å gjøre modellene enda bedre til å predikere. Allikevel ser det ut til at hovedtrekkene i resultatene samsvarer med det man tidligere har fått.

Prediksjonsfeilen som er rapportert er en skalert utgave av roten av MSE og det kan vises ved omregning at denne feilen for for hver respons for PLS modellene er henholdsvis 0.0034, 0.0042 og 0.0029 og PPLS modellen 0.0045, 0.0069 og 0.0023. (PPLS metoden som er brukt bruker også en form for støyredusering av modellene før responsene blir predikert). Sammenligner vi dette med våre resultater ser vi at vi også får frem at PLS er bedre enn PPLS for de to første responsene, mens PPLS er best for den siste (Figur 6.31 og tabell 6.5) for de enklere modellene.



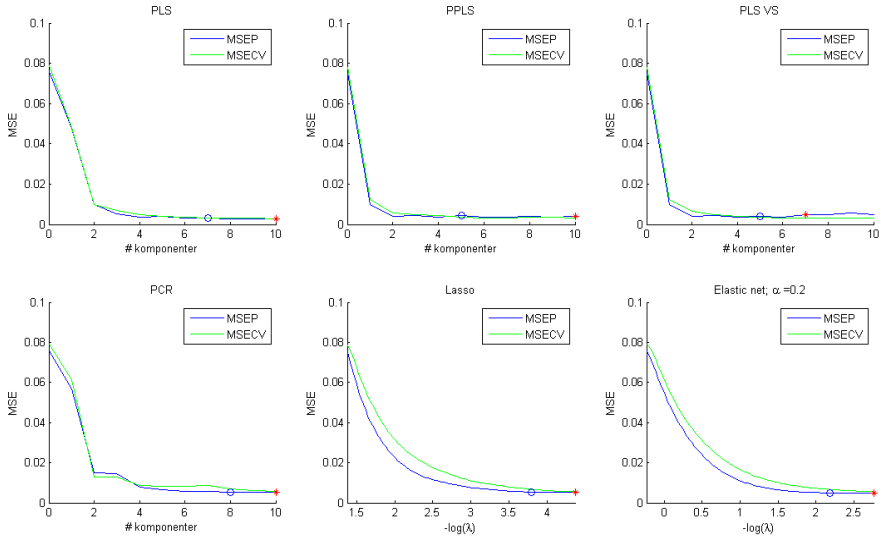
Figur 6.31: *Prediksjonsfeil (MSE) på testsettet for de ulike metodene for hver respons. Til venstre er resultatet fra modellene med lavest feil i kryssvalideringen, og til høyre er resultatet for den enklere og ikke signifikant forskjellige modellen.*

For modellene med lavest kryssvalideringsfeil er det ingen modell som skiller seg ut som bedre enn de andre for de tre responsene. Her scorer PLS best for den første responsen, PLS-VS best for den andre responsen, og elastic net best for den tredje. For de enklere modellene er det PLS som scorer best på den første og nå også på den andre responsen, og for den tredje scorer PLS-VS best.

Sammenligner vi modellene med lavest kryssvalideringsfeil og de enklere modellene får PLS-VS bedre prediksjon for de enklere modellene for to av tre responser, mens PPLS scorer dårligere for to av tre. Elastic net og lasso predikerer svært likt for begge modellene for den første og den tredje responsen, men får dårligere prediksjon med den enklere modellere for den andre responsen.

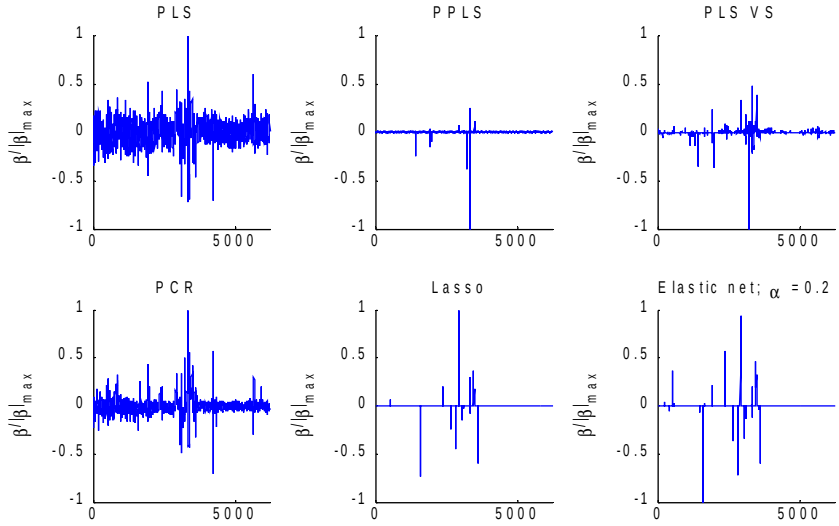
6.7.1 Andel kumelk – første respons

For den første responsen får alle metodene lav prediksjonsfeil og det er lite som skiller de ulike metodene, men PLS viser seg å være best. Metodene gir alle stabile modeller og kryssvalideringen gir oss et godt inntrykk av prediksjonsfeilen i forhold til kompleksitet på modellene, da prediksjonsfeilen på et uavhengig testsett følger denne svært godt for metodene (Figur 6.32). For denne responsen har ikke lasso og elastic net noen særlig effekt av å evaluere de enklere modellene. Prediksjonsfeilen er omtrent lik for modellen med lavest kryssvalideringsfeil og den enklere modellen for begge metodene.



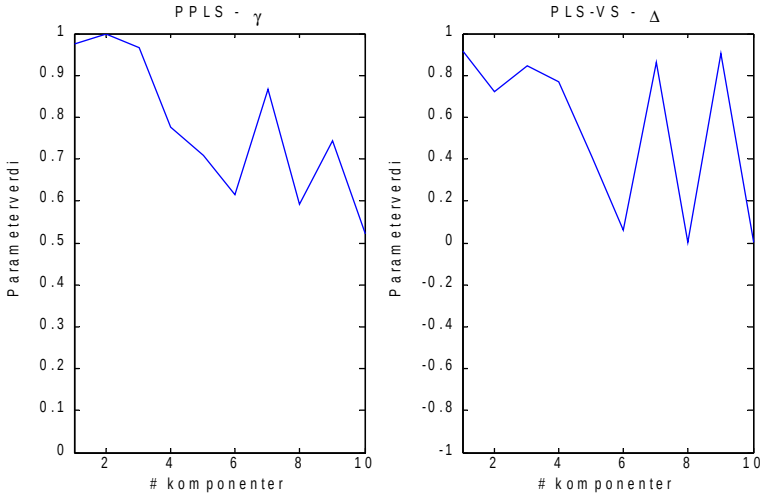
Figur 6.32: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den første responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

Modellene til metodene har alle et tyngdepunkt med store regresjonskoeffisienter for variablene omkring variabel nummer 3000 (Figur 6.33). PLS og PCR gir modeller der de fleste variablene har middels vekt, mens modellene til PPLS, PLS-VS, lasso og elastic net plukker seg ut enkelte variabler som vektes høyere og vi får dermed enklere modeller. Selvom PPLS har tydelige topper selekterer den ikke bort variabler, men samtlige variabler er med i modellen og mange av variablene har relativt små koeffisienter. PLS-VS, lasso og lasso gir oss modeller med få variabler i, der henholdsvis 91 %, 99.5 % og 98.8 % av variablene har regresjonskoeffisienter lik 0 (Tabell 6.5).



Figur 6.33: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den første responsen.*

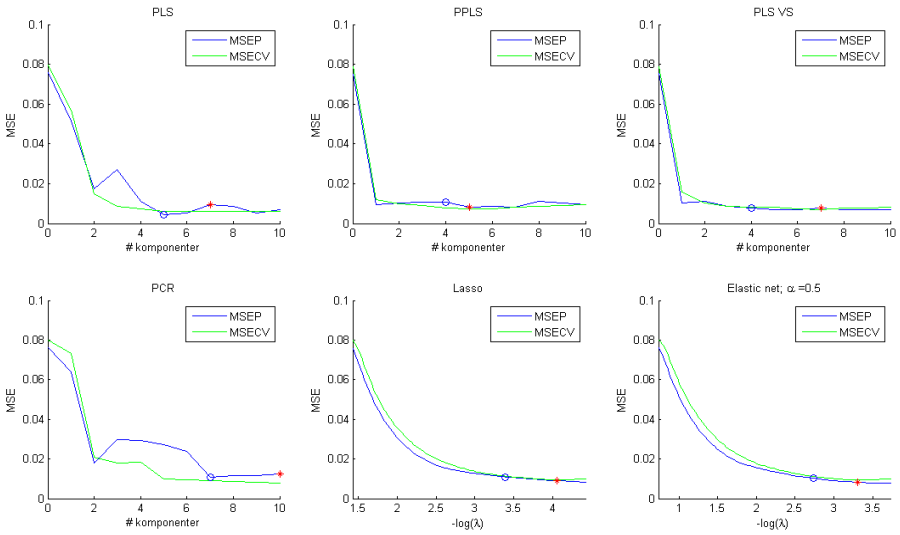
En interessant observasjon er at parameterne velger relativt like variabler til hver komponent for denne responsen. Potenseringsparameteren i PPLS og trukeringsparameteren i PLS-VS er for de tre og fire første komponentene relativt høye, så de første komponentene består av variabler med relativt høy korrelasjon til responsvariabelen. Deretter velges vekselvis variabler med høy korrelasjon og høy kovarians for de samme komponentene. Det kan antas at modellene vi får fra PPLS og PLS-VS da vil være svært like med et gitt antall komponenter. Dette stemmer også godt overens med hvordan prediksjonsfeilen endrer seg i forhold til kompleksiteten for de to metodene.



Figur 6.34: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den første responsen.*

6.7.2 Andelen geitemelk – andre respons

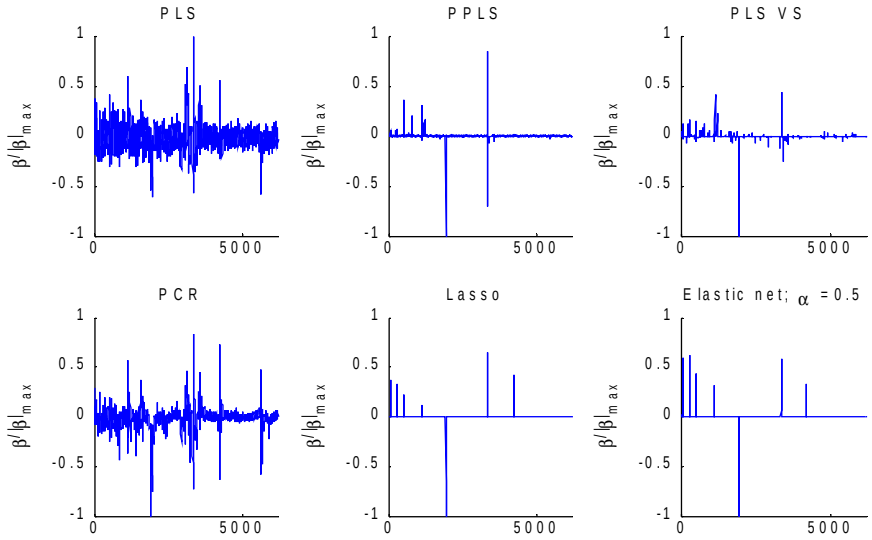
For den andre responsen skiller PLS seg ut med svært god prediksjon for den enklere modellen med fem komponenter. I kryssvalideringen er det lite som tilsier at nettopp denne modellen skal skille seg spesielt ut, men dette er svært gunstig for prediksjonen (Figur 6.35). For modeller med mindre enn fem komponenter er det litt overraskende økning i prediksjonsfeil, noe som tyder på at prediksjon med disse modellene er mer negativt. Noe av det samme ser vi for PCR, der modellen med to komponenter har god prediksjon, mens for modellene med tre til seks komponenter er påvirket av informasjon som gir dårligere prediksjonsevne. Modellen som er valgt (sju komponenter) har lavere prediksjonsfeil. Modellene til PPLS, PLS-VS, lasso og elastic net har alle jevnere kurver for prediksjonsfeil og vi kan være tryggere på å få gode modeller ved bruk av disse. Lasso og elastic net får dårligere prediksjon når vi evaluerer de enklere modellene sammenlignet med modellene med lavest kryssvalideringsfeil.



Figur 6.35: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den andre responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

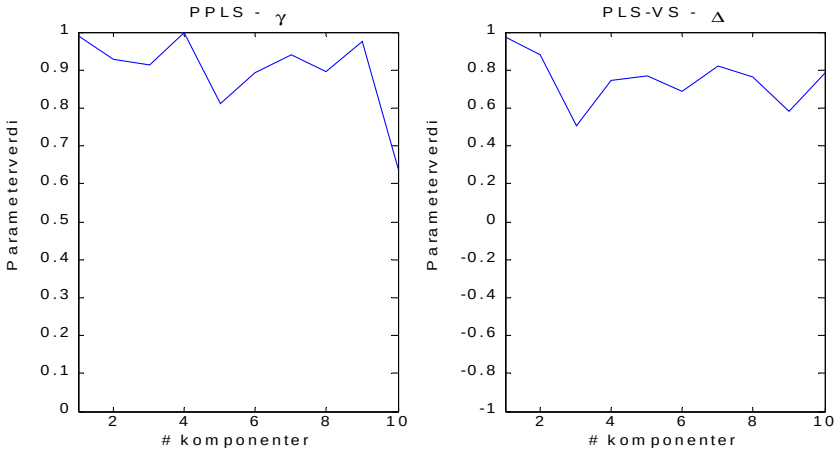
Som for den første responsen har PLS og PCR svært kompliserte modeller, der mange av variablene er vektet middels høyt og enkelte variabler er vektet svært høyt. På den andre responsen produseres enklere modeller der færre variabler er høyt og middels vektet (Figur 6.36). PPLS har her selektert bort 2.8% av variablene, PLS-VS har selektert bort 93.9%, lasso har selektert bort 99.8% og elastic net har selektert bort 99.5% (Tabell 6.5). For denne responsen ser vi at disse modellene er likere enn for den forrige responsen. Metodene velger

tydelig svært nærliggende variabler til modellen, og variabelen som vektlegges mest for PPLS og PLS-VS (1931) er nabovariabelen til den som vektlegges i lasso og elatic net (1932). Tydelige topper i samme område på regresjonskoeffisientene tyder på at metodene også her finner mange av de samme underrommene for prediksjon.



Figur 6.36: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den andreresponsen.*

De to første komponentene til PPLS og PLS-VS har svært like optimaliseringsparametere (Figur 6.37). Her inkluderes høyt korrelerte variabler i modellen for begge metodene. Etter to komponenter er det større forskjell på parameterne, men begge fortsetter å inkludere høyt til middels høyt korrelerte variabler i de neste komponentene.

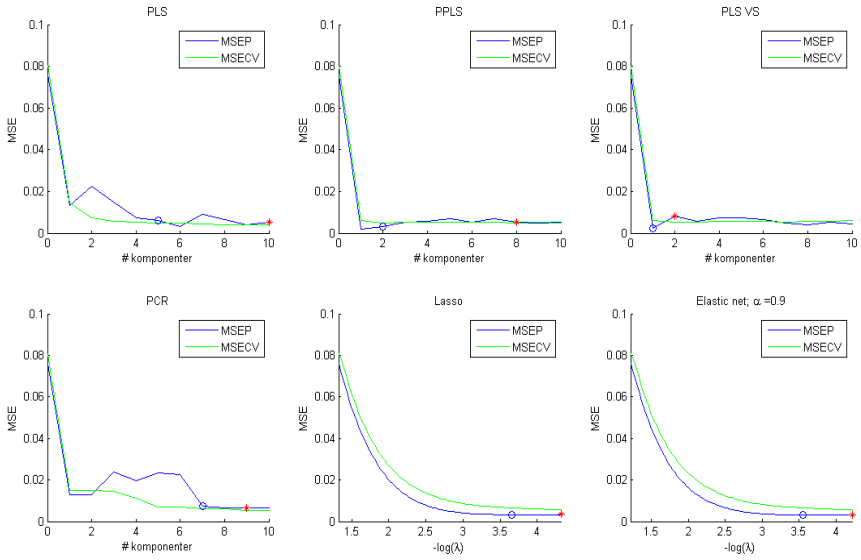


Figur 6.37: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den andre responsen.*

6.7.3 Andelen sauemelk – tredje respons

For den tredje responsen scorer PLS-VS dårligst på prediksjonen for modellen med lavest kryssvalideringsfeil, men best for den enklere modellen. Også PPLS får en svært god forbedring i prediksjon for den enklere modellen. PCR, lasso og elastic net har omtrent uendret prediksjonsfeil for de to modellene mens PLS får dårligere prediksjon når vi ser på den enklere modellen.

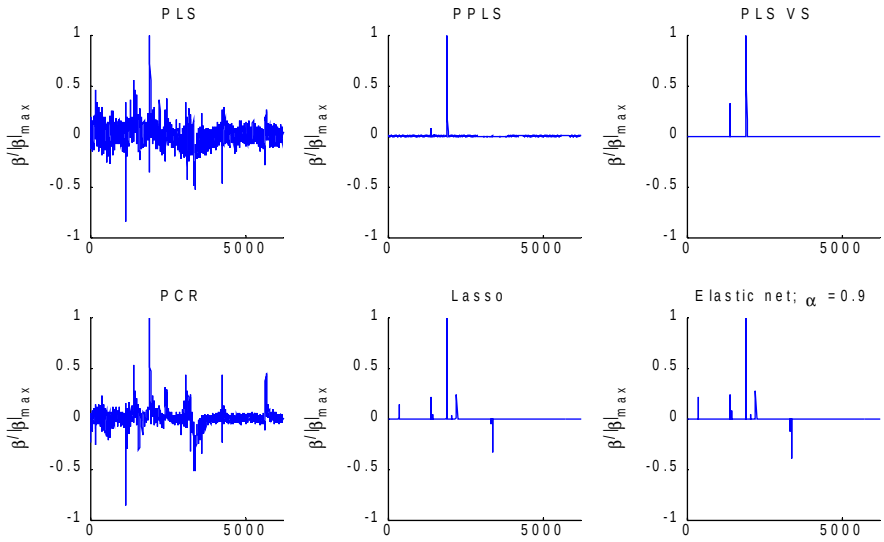
kryssvalideringen gir ikke inntrykk av det, men PLS-VS får dårligere prediksjon for modellen med lavest kryssvalideringsfeil (to komponenter) og bedre prediksjon for den enklere modellen (en komponent!) (Figur 6.38). I kryssvalideringen ser det ut som disse modellene gir omtrent lik prediksjonsfeil, men det er ikke tilfellet for det uavhengige testsettet. På testsettet ser vi at PPLS har svært god prediksjon for modellen med én komponent, men denne modellen blir ikke valgt. PLS og PCR gir oss noe ustabile modeller med færre komponenter enn de vi velger oss, men modellene stabiliserer seg etter et vist antall komponenter. Lasso og elastic net har igjen svært jevne og glatte kurver og det er svært liten endring i prediksjonsfeil for modellen med lavest kryssvalideringsfeil og den enklere modellen.



Figur 6.38: *Prediksjonsfeil som funksjon av kompleksitet for de ulike metodene for den tredje responsen. Den grønne kurven representerer prediksjonsfeilen i kryssvalideringen, den blå kurven representerer prediksjonsfeilen på det uavhengige testsettet, den røde stjernen indikerer modellen med lavest feil i kryssvalideringen og den blå sirkelen indikerer den enklere modellen.*

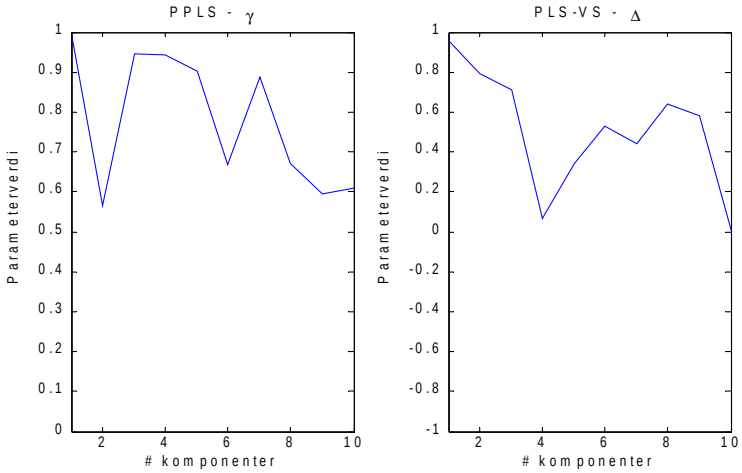
For den tredje responsen får vi svært enkle modeller for PPLS, PLS-VS, lasso og elastic net med få høyt vektede variabler, mens PLS og PCR gir modeller der mange av variablene er middels vektet og et par variabler er relativt høyt vektet (Figur 6.39). For alle modellene med

unntak av den fra PCR, gies den samme variabelen (nummer 1920) størst regresjonskoeffisient. Den variabelen med nest høyest koeffisient i modellen til PPLS og PLS-VS (1410) er nabovariabel med den nest største i modellen til lasso og elastic net (1409). Metodene finner her tydelig mye av det samme underrommet for prediksjon. PPLS metoden inkluderer alle variablene i modellen, men de fleste har relativt lav vekt. PLS-VS derimot selekterer bor 99.7% av variablene, og lasso og elastic net selekterer bort 99.6%.



Figur 6.39: *Skalerte regresjonskoeffisienter for den enklere modellen til de ulike metodene på den tredje responsen.*

For den første komponenten i PPLS og PLS-VS er det høyt korrelerte variabler som inkluderes (Figur 6.40). Dette viser seg på testsettet å være de beste modellene for PPLS og PLS-VS, men denne velges i kryssvalideringen kun med PLS-VS. Deretter avviker parameterne i større grad. Den andre komponenten i PPLS er tilnærmet som den blir valgt i vanlig PLS, der variabler vektes på grunnlag av høyest kovarians. PLS-VS inkluderer her en komponent med høyt korrelerte variabler. Prediksjonsfeilen for disse modellene spriker en del her, da PPLS får en modell med lavere prediksjonsfeil for denne modellen. For denne responsen ser vi ikke like tydelig sammenheng mellom parameterene som vi har gjort for de andre responsene i datasettet



Figur 6.40: *Potenseringsparameteren i PPLS og trunkeringsparameteren for PLS-VS for hver komponent for den tredje responsen.*

Datasett	Metode	Antall komponenter/ λ / α, λ	Prediksjonsfeil (MSEP)	Korrelasjon	Antall koeffisienter lik 0
Maldi-tof (ku)	PLS1	7 (10)	0.0032 (0.0027)	0.9815 (0.9834)	0 (0)
Maldi-tof (ku)	PPLS1	5 (10)	0.0046 (0.0041)	0.9760 (0.9783)	0 (0)
Maldi-tof (ku)	PLS1/m VS	5 (7)	0.0040 (0.0049)	0.9791 (0.9752)	5620, 91.0 % (1236, 20. 0 %)
Maldi-tof (ku)	PCR	8 (10)	0.0052 (0.0052)	0.9726 (0.9741)	0 (0)
Maldi-tof (ku)	Lasso	0.0225 (0.0126)	0.0054 (0.0051)	0.9675 (0.9673)	6149, 99.5 % (6132, 99.2 %)
Maldi-tof (ku)	Elastic net	0.2, 0.1123 (0.2, 0.0632)	0.0049(0.0048)	0.9710 (0.9696)	6102, 98.8 % (6080, 98.4 %)
Maldi-tof (geit)	PLS1	5 (7)	0.0043 (0.0095)	0.9761 (0.9648)	0 (0)
Maldi-tof (geit)	PPLS1	4 (5)	0.0106 (0.0083)	0.9482 (0.9585)	171, 2.8 % (0)
Maldi-tof (geit)	PLS1/m VS	4 (7)	0.0077 (0.0076)	0.9513 (0.9528)	5803, 93.9 % (5692, 92.1 %)
Maldi-tof (geit)	PCR	7 (10)	0.0108 (0.0124)	0.9601 (0.9567)	0 (0)
Maldi-tof (geit)	Lasso	0.0336 (0.0173)	0.0109 (0.0091)	0.9404 (0.9489)	6165, 99.8 % (6143, 99.4 %)
Maldi-tof (geit)	Elastic net	0.5, 0.0652 (0.5, 0.0367)	0.0103 (0.0082)	0.9459 (0.9551)	6151, 99.5 % (6133, 99.3 %)
Maldi-tof (sau)	PLS1	5 (10)	0.0063 (0.0052)	0.9798 (0.9842)	0 (0)
Maldi-tof (sau)	PPLS1	2 (8)	0.0031 (0.0052)	0.9903 (0.9776)	0 (0)
Maldi-tof (sau)	PLS1/m VS	1 (2)	0.0021 (0.0083)	0.9878 (0.9862)	6161, 99.7 % (6128, 99.2 %)
Maldi-tof (sau)	PCR	7 (9)	0.0073 (0.0067)	0.9753 (0.9743)	0 (0)
Maldi-tof (sau)	Lasso	0.0259 (0.0133)	0.0033 (0.0034)	0.9850 (0.9850)	6157, 99.6 % (6137, 99.3 %)
Maldi-tof (sau)	Elastic net	0.9, 0.0288 (0.9, 0.0148)	0.0032 (0.0033)	0.9853 (0.9852)	6155, 99.6 % (6135, 99.3 %)

Tabell 6.5: *Resultater for den enklere modellen fra de ulike metodene på det uavhengige testsettet på MALDI-TOF-datasettet. Resultatene for modellen med lavest feil i kryssvalideringen er i parentes. I de tilfellene der det ikke finnes noen enklere og ikke signifikant forskjellig modell fra den med lavest feil i kryssvalideringen markeres dette med (-).*

6.7.4 Oppsummering MALDI-TOF:

På MALDI-TOF-datasettet skiller PLS seg ut med best prediksjon sammenlignet med de andre metodene for den første og den andre responsen, mens metoder som har mulighet til å spisse modellen mot enkeltvariabler viser seg å predikere bedre på den tredje responsen når vi evaluerer de enklere modellene. Blant variabelseleksjonsmetodene er PLS-VS den beste for denne responsen med den enklere modellen der kun 18 variabler er inkludert i modellen(!).

At PLS predikerer best på de to første responsene kan muligens komme av at prediksjon av andelen kumelk og geitemelk ikke så lett kan forklares med enkeltvariabler, men i kombinasjon av mange variabler. Andelen av sauemelk derimot, predikeres bedre for metodene som fokuserer modellene mot enkeltvariabler. Dette var også den samme trenden som har blitt oppdaget i tidligere studier (Liland et al. 2009). I forsøket som var gjort på melkeblandingene var det også avdekket en forskjell mellom disse melkeblandingene, da sauemelk inneholdt mer proteiner per gram enn det ku og geitemelk inneholdt og derfor hadde spektrene høyere intensitet. Dette kan muligens ha noe å si for hvilke metoder som egner seg best til prediksjon av andeler av melketypene. Det er imidlertid langt unna oppgavens fagområde, så å finne ut av dette får overlates til mer kyndig personell.

Gjentatte ganger ser vi at lasso og elastic net lager svært stabile modeller, og kompleksitetsparameteren gir svært jevn endring i prediksjonsfeilen. Modellene med lavest kryssvalideringsfeil er for alle responsene like god eller bedre for metodene sammenlignet med den enklere modellen. Feilen vi får i kryssvalideringen stemmer også godt med den vi får når vi ser på testsettet da trenden er lik i forhold til kompleksiteten av modellen. Dette gir oss en trygghet når vi skal velge modeller, da relasjonen mellom feil i kryssvalideringen stemmer godt med det vi observerer på nye data. Denne stabiliteten og tryggheten må man imidlertid bøte for, i og med at selve modelltilpassningen tar svært lang tid på dette datasettet. Da elastic net i kryssvalideringen må finne gode parametere for α og λ, sammen med det faktum at det er et relativt stort antall variabler i datasettet, viste det seg at beregningen tok svært lang tid sammenligenet med de andre metodene. Lasso har høyere prediksjonsfeil enn elastic net for alle responsene på modellene med lavest kryssvalideringsfeil og de enklere modellene men bruker mindre enn en tiendel av tiden. Lasso og PLS-VS bruker omtrent like lang tid, men for samtlige responser har PLS-VS bedre prediksjon for den enklere modellen sammenlignet med den modellen fra elastic net. For datasett med enda flere forklaringsvariable enn det vi har i dette datasettet vil trolig differansen i tidsbruk bli enda

større.

PLS-VS og PPLS har for dette datasettet svært mange likheter i forhold til prediksjonsfeil for hver komponent. Metodene følger den samme trenden med økende kompleksitet, noe som tyder på at PLS-VS fanger opp mye av det samme som PPLS gjør. Modellene vi får ut for metodene viser også at det er mange av de samme variablene som vektlegges. Parameterne som velges er også svært like, og metodene følger hverandre tett i forhold til når variabler med høy korrelasjon, høy kovarians eller høyt standardavvik dominerer i komponentene som beregnes. Resultatene viser at metoden også i betydelig grad selekterer variabler. For dette datasettet bortselekteres over 90% av variablene i modellene til hver respons. Når vi sammenligner de enklere modellene fra PLS-VS og PPLS med modellene med lavest kryssvalideringsfeil fra lasso og elastic net, viser det seg at PLS-VS har lavest prediksjonsfeil for alle de tre responsene for dette datasettet.

7 Oppsummering – diskusjon

7.1 Teori

I masteroppgaven min har vi gått igjennom kjente metoders ulike tilnærminger til lineær regresjon på multivariate data. Med OLS som utgangspunkt, har vi i denne oppgaven redegjort for kjent problematikk med datasett der forklaringsvariablene er høyt korrelerte og dette i kombinasjon med situasjoner der man har flere forklaringsvariable enn observasjoner i datasettet. Vi har også problematisert at med moderne teknologiske fremskritt åpnes stadig større muligheter for å generere kompliserte multivariate datasett i tilknytning til analyser av avanserte problemstillinger. Fordi tradisjonelle tradisjonelle statistiske regresjonsmetoder ikke fungerer tilfredstillende i slike situasjoner, er det nødvendig å utvikle nye analysemetoder som er istand til å produsere enkle og tolkbare modeller som er egnet til å forklare variasjon i responsene som studeres.

Vårt eget metodebidrag, PLS-VS er sammenlignet med flere andre regresjonsmetoder basert på blant annet variabelseleksjon, projeksjoner ned på underrom, krympingsmetoder og kombinasjoner av disse. Vi har vært innom hvilke kriterier variabelseleksjons- krympings- og projeksjons-metoder benytter, samt litt av teorien som ligger bak for hvordan metodene fungerer til modellering. Dette har vi gjort for å danne forståelse av hvordan de forskjellige metodene faktisk fungerer, og hvilke fellestrekk og ulikheter vi kan forvente i bruken av dem.

Hjelpemetoder som blir brukt internt i implementasjonen av PLS-VS og i PPLS er kortfattet beskrevet. Dette har vi gjort for å vise at disse metodene har et teoretisk fundament og er intuitive, men ikke er ufeilbarlige i optimeringen. Likedan er prosesseringsteknikker (brukt til normalisering av datasett og innlemming av apriorikunnskap) benyttet i forkant og etterkant av diverse analyser beskrevet.

Validering både ved hjelp av kryssvalidering og bruk av uavhengige testsett utgjør selve grunnlaget for valg av optimal modellkompleksitet og sammenligning av prediksjonsegenskaper til ulike metoder. Som en sentral del av dette har vi sett benyttet en metode for modellforenkling og -utvelgelse som benytter at feilen i modellen med minimal MSE kan oppfattes som en χ^2 -fordelt størrelse. I denne metoden faller valget på den «enkleste» modellen som ikke predikerer signifikant forskjellig fra den «beste» modellen vi empirisk observerer. Hensikten er både å oppnå modeller som har enklere tolkning og som

minsker risikoen for overtilpasning.

Hovedmålet i oppgaven har vært utviklingen av en eksplorativt anvendbar PLS metode kalt PLS-VS, som både kan gi enkle og fleksible modeller med god prediksjonsevne. Motivasjonen bak metoden kommer fra vanlig PLS og videreutviklingen PPLS, og i sluttproduktet PLS-VS er disse kombinert med en variant av variabelseleksjon som bruker enkel teori om stykkevis rette linjer i to dimensjoner. Vi har tatt utgangspunkt i hvordan vektingsvektoren i PLS og PPLS blir laget, og funnet approksimasjoner til dette som inneholder elementer fra variabelseleksjon. En bonus ved metoden vår (tilsvarende som ved PPLS) er at den tillater en form for eksplorativ utforskning i modelleringen ved at brukeren kan definere avgrensninger av ulike parameterverdien. Dette fører til begrensninger på vektingsvektoren slik at den for eksempel kun genererer komponenter bestående av lineærkombinasjoner av et begrenset utvalg av de tilgjengelige forklaringsvariablene. For å finne optimale parameterverdier har vi benyttet oss av en metode som finner vektingsvektoren som maksimerer korrelasjonen mellom scorevektoren og den deflaterte responsen.

Sist men ikke minst har vi jobbet fokusert med ulike (og mer eller mindre hensiktsmessige) grafiske måter å presentere data og resultater fra de multivariate dataanalysene på. Grafikken vi endte opp med å bruke illustrerer prediksjonsfeil som funksjon av modellkompleksitet, og tolkbarhet for modellene som produseres gjennom plott av regresjonskoeffisientene. Vi har også benyttet fargevarierte søylediagrammer for en forenklet sammenligning av alle undersøkte metoder for ulike datasett og responser. Ved å sammenligne plott av de optimaliserte parameterene fra PPLS og PLS-VS har vi kunnet kontrollere om metoden vår fungerte som forventet.

7.2 Hva skjedde i praksis

Vår nye metode PLS-VS har blitt sammenlignet med vanlig PLS, PPLS, PCR, *lasso* og *elastic net*. For hver metode har vi fokusert på to modeller, modellen med lavest kryssvalideringsfeil og en enklere modell som ikke gir signifikant dårligere prediksjonsfeil fra modellen med den laveste kryssvalideringsfeilen. For disse to modellene har vi igjen sammenlignet prediksjonsfeil ved anvendelse på et uavhengig sett med nye testdata. Dette har dannet grunnlaget for å vurdere om noen av metoder skiller seg ut med betydelig høyere eller lavere prediksjonsfeil sammenlignet med de andre metodene.

Vi har også sett på hvordan modeller er bygget opp med de ulike metodene, og hvor stabilt

disse blir bygget. I tillegg har tolkbarheten til de enkleste modellene som blir produsert for hver metode blitt vurdert og sammenlignet, med fokus på om metodene produserer enkle modeller og om de utfører variabelseleksjon. Til sist har vi som en kvalitetssjekk sett på om PLS-VS lager lineærkombinasjoner av forklaringsvariable med omtrent samme egenskaper som PPLS. Dette vil tyde på om vi har klart oppnå fleksibiliteten som vi ønsker av metoden vår.

I analysedelen av oppgaven erfarte vi at på deigdatasettet så viste PPLS og PLS-VS tendenser til å produsere ustabile modeller. Ved å utnytte de eksplorative mulighetene i disse metodene (begrensninger på potenseringsparameteren og trunkeringsparameteren) utvidet vi analysen vår ved å studere de stabiliserende konsekvensene av dette. Mye tyder på at ustabile modeller ofte kan være et resultat av at komponentene ble dominert av forklaringsvariable som var blant de høyest korrelerte med responsvariabelen.Ved å innføre en øvre begrensning på parameterverdiene, oppnår man å tvinge de genererte komponentene til å vektlegge flere av forklaringsvariablene med det resultat at komponentene ble mer robuste. Dette gav en positiv bekreftelse på nytteverdien av å bruke PLS-VS eksplorativt.

Fra resultatene av prediksjonsfeil for de to modellene vi evaluerte, var det ingen metode som skilte seg spesielt ut som et bedre alternativ enn de andre. PCR var kanskje den metoden som skilte seg mest ut som metoden med høyest prediksjonsfeil. I og med at det ikke er påfallende forskjeller i prediksjonsfeil for de øvrige metodene, vil det i praksis være andre egenskaper som blir avgjørende for hvilke(n) metode(r) som er gunstigst å bruke. Egenskaper som enkelhet, tolkbarhet og/eller beregningsintensivitet er i denne sammenhengen interessante.

Ikke uventet erfarte vi at PLS og PCR ofte resulterte ikompliserte modeller, med samtlige forklaringsvariable inkludert i modellene og med middels til relativt høye regresjonskoeffisienter. PPLS, PLS-VS, lasso og elastic net produserer på den annen side temmelig enkle modeller hvor det fokuseres inn mot et mindre antall viktige forklaringsvariable. Disse modellene blir dermed enklere å tolke. Hvis man gjennom en forstudie finner at enkelte forklaringsvariable alene er nok til å kunne beskrive responsen på en god måte, kan man bruke dette som grunnlag til å avgjøre om man skal utelate å måle «uviktige» forklaringsvariable i et påfølgende forsøk. I tillegg har vi sett at PLS-VS på lik linje med lasso og elastic net utfører variabelseleksjon i modellene. PPLS gjør dette bare i mindre og sjeldnere grad. Dette understøtter at implementasjonen av PLS-VS har fungert tilfredsstillende. Vi har også erfart at PPLS, PLS-VS, lasso og elastic net gjerne finner en god del av de samme forklaringsvariablene. Dette er interessant og hyggelig bekreftelse, spesielt

siden disse metodene har benyttet til dels svært ulike kriterier for å bygge modeller.

For lasso og elastic net har vi gjennom analysene gjort oss noen erfaringer om hvordan de to modellene som evalueres predikerer på det uavhengige testsettet. I samtlige tilfeller har vi har sett at den enklere modellen har omtrent lik eller høyere prediksjonsfeil sammenlignet med modellen med lavest kryssvalideringsfeil. Sammen med en undersøkelse av grafikken over prediksjonsfeil som funksjon av modellkompleksitet, har vi sett at modellen med lavest kryssvalideringsfeil ofte gir en brukbar indikasjon på optimal modellkompleksitet for prediksjon på det uavhengige testsettet. Disse metodene bygger sine modeller svært stabilt, og det er så godt som ingen antydning på at modellene med lavest feil i kryssvalideringen er overtilpasset. Modeller med lavest kryssvalideringsfeil som indikasjon på optimal modellkompleksitet kan derfor anbefales som tilstrekkelige. For PLS, PPLS og PLS-VS var det i mange tilfeller derimotgunstig å se på den enklere modellen. Disse metodene ser ut til lettere å produsere overtilpassede modeller, og det vil derfor være av større interesse å velge den enkleste modellen blandt de to kandidatene. For PCR fant vi ingen klar tendens til hvilken av de to modellene som hadde best prediksjon av responsen.

En fordel med PPLS, PLS-VS og lasso er at de er betydelig mindre beregningskrevende enn elastic net. Når vi i kryssvalideringen optimaliserer for to parametere i elastic net, blir dette nødvendigvis en mer beregningstung operasjon sammenlignet med de andre metodene. Dette var spesielt påfallende for MALDI-TOF-datasettet. I og med at den lange tiden det tok å optimalisere parameterne ikke kan kompensere med bedre prediksjon av responsen, favoriserete dette PPLS, PLS-VS og lasso.

En viktig kvalitetssjekk av PLS-VS i forhold til PPLS, var blant annet å se om metodene gav komponenter der forklaringsvariable med tilnærmet like egenskaper ble brukt som utgangspunkt. For alle datasettene vi studerte er erfaringen at metodene i stor grad fungerte nokså likt for de første komponentene, og i noen tilfeller ble komponentene laget av forklaringsvariable med like egenskaper omtrent opp til den tiende komponenten. Det at parameterne /komponentene fokuserte på forklaringsvariable med like egenskaper for de første komponentene tyder på at metodene finner omtrent de samme underrommene som basis for prediksjon. I enkelte tilfeller, spesielt for den andre og tredje responsen for deigdatasettet og den første responsen for MALDI-TOF-datasettet, så vi at komponentene også fokuserte på forklaringsvariable med like egenskaper også etter de første komponentene. Dette er en sterk indikasjon på at vi har klart å få PLS-VS til å fungere på en måte som ligger svært nær PPLS.

7.3 Målet og erfaringer

I innledningen av oppgaven nevnte vi at målet var å utvikle en metode motivert av PPLS som hindrer ikke informative forklaringsvariable i å forstyrre modellene våre. Å lykke med dette vil opplagt forenkle modelltolkning og forhåpentligvis gi like gode prediksjoner som mer tradisjonell PLS-metodikk. PLS-VS har vist seg å ha prediksjonsevne fullt på høyde med de metodene vi har sammenlignet den med. Metoden har også vist at den i kan gi enkle modeller på lik linje med lasso og elastic net. Det er derfor grunnlag for å påstå at PLS-VS representerer et ikke uinteressant bidrag til utviklingen av regresjonsmetodikk tilpasset komplekse multivariate datasett.

7.4 Videre arbeid og muligheter

Innen multivariat regresjon er det mye å ta tak i for videre studier. Videre utvikling av kriterier for sammenligning av ulike regresjonsmetoder bør etableres, slik at man kan få enda bedre innsikt i hvordan forskjellige kandidatmetoder fungerer på ulike typer datasett. Det er også viktig å forstå potenielle bruksområder for PLS-metodikk. I og med at PLS-VS i stor grad lykkes med å fjerne ikke informative forklaringsvariable uten at prediksjonsevnen forringes vesentlig sammenlignet med PPLS, vil det opplagt interessant å undersøke hvordan vår metode kan modifiseres og tilpasses klassifikasjonsproblemer.

Appendix

A Forkortelser

LARS – Least Angle

LASSO – Least Absolute Shrinkage Selection Operator

MSE – Mean Square Error

NIPALS – Non Iterative Partial Least Squares

OLS – Minste kvadraters metode

PCA – Prinsipalkomponentanalyse

PCR – Prinsipalkomponentregresjon

PLS – Partial Least Squares

PLS-VS – Partial Least Squares with Variable Selection

PPLS – Powered Partial Least Squares

SVD – Singulærverdidekomposisjon

B Matlab-kode

B.1 Vanlig PLS - rutine

```
function [W, T, P, Q, mX, mY, beta, delta] = pls_cutoff(Xdata,Ydata,comp,
lower,upper)
% Xdata, Yvector - Ydata, comp - number of components to be extracted,
lower-
% use 1 to -1, upper - use 1 to minus 1. (1 and 1 is default). in case of
% positive values, lower and upper is the highest possible value of
% truncation to 0 for the standardeviation and correlation respectively. If
% negative values of lower or upper, then this is the lower limit of
% truncation to zero of the correlation and standarddeviation respectively.
% Algorithm returns regression coeffisients, W, P, T and Q according to
% notation conventions, means of the X-variables, mean og the Y-vector and
% the truncationparameter
[n,p] = size(Xdata)   ;
% Centering X and Y:
mX = mean(Xdata);
mY = mean(Ydata);
X = Xdata - ones(n,1)*mX;
Y = Ydata - mY;
if nargin == 3
    lower = 1;
    upper = 1;
```

```
end

% Declaration of variables:
W = zeros (p,comp);             % Loadingweights
T = zeros (n,comp);             % T-scores
P = zeros (p,comp);             % P-loadings
Q = zeros (comp,1);             % Q-loadings
beta = zeros(p,comp);           % Regressioncoeffisients for models
delta = zeros (comp,1);         % For truncation-parameters from calculations
of w-s in weights
orto = 0;                       % <> 0 indicatesorthogonalization of
loadingweights.

for i = 1:comp                  % Notation corresponding to frame 3.4 of
Martens & Næs
    [w,delta(i)] = weights(X, Y, W(:,1:comp), lower, upper, orto);
    % weights returns loadingdweights (w) and the best corresponding
    % truncationsparameter (delta), according to the chosen strategy
    W(:,i) = w/norm(w);
    T(:,i) = X*W(:,i);
    P(:,i) = X'*T(:,i)*(T(:,i)'*T(:,i))^-1;
    Q(i,1) = Y'*T(:,i)*(T(:,i)'*T(:,i))^-1;
    X = X - T(:,i)*P(:,i)';
    Y = Y - T(:,i)*Q(i,1)';
    beta(:,i) = W(:,1:i)*(P(:,1:i)'*W(:,1:i))^-1*Q(1:i,1);
end
```

B.2 weights

```
function [w, delta] = weights(X, Y, W, lower, upper, orto );
% [w, delta] = weights(X, Y, W, lower, upper, orto) - The function
% pre-processing the X and Y data and according to the limits it's decided
% whether:
% 1) both truncation of correlation and standarddeviation to 0 should be
% evaluated (truncation of standarddeviation from lower to 0, and
% truncation of correlation from upper to 0)
% 2) truncation of correlation to 0 should be evaluated (from upper to
-lower)
% 3) truncation of standarddeviation to 0 should be evaluated (from lower
to - upper)
% for the choosen truncation option, we choose the truncation parameter in
% the given interval and the corresponding loadingweight w that maximizes
% the absolute correlation between Y and Xw. The optimization is calculated
% by calling the MATLAB system-function 'fminbnd' to minimize the function
% 'correlations(delta, X, Y, signx, corr, stdx)'.
%
% 'fminbnd' implements a one-variable optimization by golden section search
% and parabolic interpolation method (see Press, W. H., Teukolsky, S. A.,
% Vetterling, W. T. & Flannery, B. P. (1988). Numerical Recipes in C; The
% Art of Scientific Computing. 2 edt.)
%
% When an optimized parameter is found, we calculate the corresponding
% loadingweight w.
%
% The solution [w, delta] consisting of a loading weight (w) and the
% corresponding parameter delta is returned. Optionally orthogonalization 2
% of w with respect to earlier loadnig weights in W executed.

[corr, stdx] = corrmat(X,Y);  % Correlation between X and Y and
standarddeviations og X
```

```
signx = sign(corr);             % Sign of correlasions

corr = corr/max(abs(corr));     % Scaling absolute values of correlation and
standard deviations to
stdx = stdx/max(abs(stdx));     % assure max(cor) = max(abs(corr)) = 1

if upper + lower < 0
    error('The lower and upper limit has to define a nonempty interval of
values. Nonempty intervals require that upper limit + lower limit >= 0')
end

if upper >= 0
    if lower >= 0
            % 1) both truncation of correlation and standarddeviation to 0
should be
            % evaluated (truncation of standarddeviation from 0 to lower,
and
            % truncation of correlation from 0 to upper)

        c = zeros (5,1);
        [delta_over,c(1)] = fminbnd(@(delta) correlations(delta, X, Y,
signx, corr, stdx), 0, upper); % 1 - calculating the truncation of
correlations interval
        [delta_low,c(2)]  = fminbnd(@(delta) correlations(delta, X, Y,
signx, stdx, corr), 0, lower); % 2 - calculating the truncation of standard
deviations interval.
        c(3) = correlations(0, X,Y,signx,corr, stdx);          % 3 - PLS-
solution

%           Checking endpoints
        if upper==1 % 4 - Variable selection from correlation
            ind1=(abs(corr) == max(abs(corr)));
            w1 = zeros(size(X,2),1); w1(ind1) = 1;
            c(4) = -(corrmat(X*w1,Y)^2);
        else
            c(4) = correlations(upper, X,Y,signx,corr, stdx);
        end
         if lower==1 % 5 - Variable selection from standard deviation
            ind2=(stdx == max(stdx));
            w2 = zeros(size(X,2),1); w2(ind2) = 1;
            c(5) = -(corrmat(X*w2,Y)^2);
        else
            c(5) = correlations(lower, X,Y,signx,corr, stdx);
        end

        [cm, cmin] = max (-c); % Determine which is the most sucessfull.

       % Calculating loadingweight
        if cmin == 1;      % 1 - truncating correlations to 0
            delta = delta_over;
            w = w_calc(delta_over, signx, corr, stdx);
        elseif cmin== 2;  % 2 - truncating standard deviations to 0
            delta = - delta_low;
            w = w_calc(delta_low, signx, stdx, corr);
        elseif cmin ==3; % 3 - PLS-solution
            delta = 0;
            w = w_calc(0, signx, corr, stdx);
        elseif cmin ==4;
            delta = upper;
            if upper==1   % 4 - Variable selection from correlation
                ind1=(abs(corr) == max(abs(corr)));
```

```
                w = zeros(size(X,2),1); w(ind1) = 1;
            else          % 4 - upper limit
                w = w_calc(upper, signx, corr, stdx);
            end
        else
            delta = -lower;
            if lower==1  % 5 - Variable selection from standard deviation
                ind2=(stdx == max(stdx));
                w = zeros(size(X,2),1); w(ind2) = 1;
            else          % 5 - lower limit
                w = w_calc(lower, signx, stdx, corr);
            end
        end
    else
       % 2) truncation of correlation to 0 should be evaluated (from -lower
to upper)

        c = zeros(3,1);
        [deltamiddle,c(1)] = fminbnd(@(delta) correlations(delta, X, Y,
signx, corr, stdx), (abs(lower)) , upper); % 1 - calculating the truncation
of correlations interval

        % Checking endpoints:
        if upper==1  % 2 - Variable selection from correlation
            ind1=(abs(corr) == max(abs(corr)));
            w1 = zeros(size(X,2),1); w1(ind1) = 1;
            c(2) = -(corrmat(X*w1,Y)^2);
        else         % 2 - upper limit
            c(2) = correlations(upper, X,Y,signx,corr, stdx);
        end
        c(3) = correlations(abs(lower),X,Y, signx, corr, stdx); % 3 - lower
limit

        [cm, cmin] = max (-c); % Determine which is the most sucessfull.

        % Calculating loadingweight
        if cmin == 1;       % 1 - truncating correlations to 0
            delta = deltamiddle;
            w = w_calc(deltamiddle, signx, corr, stdx);
        elseif cmin== 2; % 2 - Variable selection from correlation
            delta = upper;
            if upper==1
                ind1=(abs(corr) == max(abs(corr)));
                w = zeros(size(X,2),1); w(ind1) = 1;
            else         % 2 - upper limit
                w = w_calc(upper, signx, corr, stdx);
            end
        else             % 3 - lower limit
            delta = abs(lower);
            w = w_calc(abs(lower), signx, corr, stdx);
        end
    end
else
    % 3) truncation of standarddeviation to 0 should be evaluated (from
-upper to lower)

    c = zeros(3,1);
    [deltamiddle,c(1)] = fminbnd(@(delta) correlations(delta, X, Y, signx,
stdx, corr), abs(upper), lower); % calculating the lower gamma interval.

    % Checking endpoints:
```

```
        if lower==1  % 2 - Variable selection from standard deviation
            ind2 =(stdx == max(stdx));
            w2 = zeros(size(X,2),1); w2(ind2) = 1;
            c(2) = -(corrmat(X*w2,Y)^2);
        else         % 2 - "lower" limit
            c(2) = correlations(lower, X,Y,signx,stdx, corr);
        end
        c(3) = correlations(abs(upper),X,Y,signx,stdx, corr); % 3 - "upper"
limit

        [cm, cmin] = max (-c); % Determine which is the most sucessfull.

         % Calculating loadingweight
        if cmin == 1;       % 1 - truncating standard deviation to 0
            delta = -deltamiddle;
            w = w_calc(deltamiddle,signx,stdx,corr);
        elseif cmin== 2; % 2 - Variable selection from standard deviation
            delta = -lower;
            if lower==1
                ind2=(abs(stdx) == max(abs(stdx)));
                w = zeros(size(X,2),1); w(ind2) = 1;
            else         % 2 - "lower" limit
                w = w_calc(lower, signx, stdx,corr);
            end
        else             % 3 - "upper" limit
            delta = upper;
            w = w_calc(abs(upper), signx, stdx,corr);
        end

end
if orto ~= 0             % Optional: To remove the W-directions accounted
for by earlier
    w = w-W*W'*w;        % loading weights if indicatied by "orto"
end;
```

B.3 correlations

```
function c = correlations(delta, X, Y, sign, v1, v2)
% Maximizing function - For a given parameter, the correlation between the
% response Y and X*w is calculated, where w is the corresponding
% loadingweight to the parameter. This is to be maximized, which equals to
% minimize the correlation with the oposite signe values of v1 smaller than
% delta is set to 0, and values of v2 larger than (1-delta) is set to 1.
% The remaining values of v1 and v2 are linearly transformed from 0 to 1.

n = length(v1);
ind = (abs(v1)<delta);     % Identifying values of v1 smaller than delta
a = 1/(1-delta);           % calculating slope
vekt_v1 = (abs(v1/(1-delta))-ones(n,1)*delta/(1-delta)); % linear
transformation of v1
vekt_v1(ind) = 0;          % Setting small values of v1 to 0.

deltas = 1 - delta;
inds = (abs(v2)>deltas);   % Identifying values of v2 larger than (1- delta)
vekt_v2 = abs(v2*a);       % linear transformation of v2
vekt_v2(inds) = 1;         % Setting large values of v2 to 1

w = sign.*vekt_v1.*vekt_v2; % corresponding loadingweight

c = -(corrmat(X*w,Y)^2);
```

B.4 w_calc

```
function w = w_calc(delta, sign, v1, v2)
% For a given truncationparameter, the corresponding loadingweight are
% calculated

v1 = abs(v1);
v2 = abs(v2);

n = length(v1);
ind =(v1<delta);                % Identifying values of v1 smaller than
delta
a = 1/(1-delta);                % calculating slope
vekt_v1 = v1/(1-delta)-ones(n,1)*delta/(1-delta);  % linear transformation
of v1
vekt_v1(ind) = 0;               % linear transformation of v1

deltas = 1 - delta;
inds = (v2>deltas);             % Identifying values of v2 larger than (1-
delta)
vekt_v2 = v2*a;                 % linear transformation of v2
vekt_v2(inds) = 1;              % Truncating large values of v2 to 1

w = sign.*vekt_v1.*vekt_v2;     % corresponding loadingweight
```

B.5 corrmat

```
function [Corr, stdX] = corrmat(X,Y)
% Computing the correlation between the columns of X and Y

[n,p] = size(X)  ;
Xs = X - ones(n,1)*mean(X);
Ys = Y - mean(Y);
stdX = std(Xs,1)' ;
stdY = std(Ys,1);

ind = (stdX==0);
stdX(ind) = 1;      % Remove insignificant std.

Corr = (Xs'*Ys).*(stdX.^-1)*(n*stdY)^-1;
stdX(ind) = 0;
Corr(ind,:) = 0;
```

C Fullstedig tabell

Datasett	Metode	Antall komponenter/ λ / α, λ	Prediksjonsfeil (MSEP)	Korrelasjon	Antall koeffisienter lik 0
Deig (fett)	PLS1	9 (12)	0.3609 (0.1664)	0.9744 (0.9915)	0 (0)
Deig (fett)	PPLS1	5 (9)	5.3005 (0.1149)	0.7640 (0.9908)	0 (0)
Deig (fett)	PLS1/m VS	4 (6)	2.6534 (4.4800)	0.8267 (0.8285)	110, 15.7 % (109, 15.6%)
Deig (fett)	PCR	8 (20)	0.7729 (0.2539)	0.9241 (0.9764)	0 (0)
Deig (fett)	Lasso	0.0271 (0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7 %)
Deig (fett)	Elastic net	1, 0.0271 (1, 0.0123)	0.1236 (0.0671)	0.9867 (0.9919)	691, 98.7 % (691, 98.7 %)
Deig (sukker)	PLS1	3 (6)	2.8682 (1.1932)	0.9151 (0.9764)	0 (0)
Deig (sukker)	PPLS1	3 (-)	1.5637 (-)	0.9608 (-)	185, 26.4% (-)
Deig (sukker)	PLS1/m VS	3 (-)	1.3921 (-)	0.9619 (-)	676, 96.6% (-)
Deig (sukker)	PCR	5 (6)	2.8221 (1.4430)	0.9353 (0.9710)	0 (0)
Deig (sukker)	Lasso	0.3420 (0.0612)	6.2286 (1.3791)	0.8324 (0.9553)	697, 99.6 % (687, 98.1 %)
Deig (sukker)	Elastic net	0.9, 0.3627 (0.9, 0.0680)	6.0623 (1.4022)	0.8400 (0.9549)	674, 96.3 % (666, 95.1 %)
Deig (mel)	PLS1	3 (9)	1.1578 (2.7042)	0.9251 (0.8832)	0 (0)
Deig (mel)	PPLS1	2 (-)	1.0779 (-)	0.9288 (-)	698, 99.7% (-)
Deig (mel)	PLS1/m VS	2 (-)	1.0560 (-)	0.9296 (-)	695, 99.3% (-)
Deig (mel)	PCR	4 (6)	1.5893 (2.2612)	0.8827 (0.9159)	0 (0)
Deig (mel)	Lasso	0.2578 (0.1116)	2.4374 (0.9363)	0.8381 (0.9333)	697, 99.5 % (695, 99.3%)
Deig (mel)	Elastic net	1, 0.2578 (1, 0.1116)	2.4374 (0.9363)	0.8381 (0.9333)	697, 99.5 % (695, 99.3 %)
Deig (vann)	PLS1	3 (5)	0.3365 (0.4242)	0.9539 (0.9271)	0 (0)
Deig (vann)	PPLS1	3 (-)	0.3914 (-)	0.9188 (-)	0 (-)
Deig (vann)	PLS1/m VS	3 (-)	0.8710 (-)	0.9214 (-)	146, 20.9 % (-)
Deig (vann)	PCR	4 (6)	0.2811 (0.6217)	0.9521 (0.9096)	0 (0)
Deig (vann)	Lasso	0.1329 (0.0149)	0.4637 (0.1682)	0.8820 (0.9610)	698, 99.7 % (686, 98 %)
Deig (vann)	Elastic net	0.1, 1.1030 (0.1, 0.1298)	0.4590 (0.1932)	0.8838 (0.9542)	642, 91.7 % (539, 77%)
Prostata	PLS1	1 (5)	0.5334 (0.5055)	0.7108 (0.7208)	0 (0)
Prostata	PPLS1	1 (5)	0.3735 (0.5208)	0.8068 (0.7111)	0 (0)
Prostata	PLS1/m VS	2 (3)	0.5033 (0.5079)	0.7234 (0.7188)	2, 25 % (0)
Prostata	PCR	2 (8)	0.7010 (0.5213)	0.6103 (0.7108)	0 (0)
Prostata	Lasso	0.2878 (0.0122)	0.5297 (0.4952)	0.7664 (0.7282)	5, 62.5 % (1, 12.5 %)

Prostata	Elastic net	0.1, 1.2458 (0.1 0.0480)	0.5563 (0.4968)	0.7623 (0.7273)	1, 12.5% (1, 12.5%)
Øl	PLS1	3 (9)	1.5742 (0.5377)	0.8586 (0.9611)	0 (0)
Øl	PPLS1	2 (-)	0.0240 (-)	0.9981 (-)	6, 0.6 % (-)
Øl	PLS1/m VS	2 (-)	0.0274 (-)	0.9978 (-)	776, 83.8 % (-)
Øl	PCR	5 (6)	0.5464 (0.5362)	0.9566 (0.9579)	0 (0)
Øl	Lasso	0.1758 (0.1186)	0.0498 (0.0234)	0.9974 (0.9990)	923, 99.7 % (918, 99.1 %)
Øl	Elastic net	1, 0.1758 (1, 0.1186)	0.0498 (0.0234)	0.9974 (0.9990)	923, 99.7 % (918, 99.1 %)
Maldi-tof (ku)	PLS1	7 (10)	0.0032 (0.0027)	0.9815 (0.9834)	0 (0)
Maldi-tof (ku)	PPLS1	5 (10)	0.0046 (0.0041)	0.9760 (0.9783)	0 (0)
Maldi-tof (ku)	PLS1/m VS	5 (7)	0.0040 (0.0049)	0.9791 (0.9752)	5620, 91.0 % (1236, 20. 0 %)
Maldi-tof (ku)	PCR	8 (10)	0.0052 (0.0052)	0.9726 (0.9741)	0 (0)
Maldi-tof (ku)	Lasso	0.0225 (0.0126)	0.0054 (0.0051)	0.9675 (0.9673)	6149, 99.5 % (6132, 99.2 %)
Maldi-tof (ku)	Elastic net	0.2, 0.1123 (0.2, 0.0632)	0.0049(0.0048)	0.9710 (0.9696)	6102, 98.8 % (6080, 98.4 %)
Maldi-tof (geit)	PLS1	5 (7)	0.0043 (0.0095)	0.9761 (0.9648)	0 (0)
Maldi-tof (geit)	PPLS1	4 (5)	0.0106 (0.0083)	0.9482 (0.9585)	171, 2.8 % (0)
Maldi-tof (geit)	PLS1/m VS	4 (7)	0.0077 (0.0076)	0.9513 (0.9528)	5803, 93.9 % (5692, 92.1 %)
Maldi-tof (geit)	PCR	7 (10)	0.0108 (0.0124)	0.9601 (0.9567)	0 (0)
Maldi-tof (geit)	Lasso	0.0336 (0.0173)	0.0109 (0.0091)	0.9404 (0.9489)	6165, 99.8 % (6143, 99.4 %)
Maldi-tof (geit)	Elastic net	0.5, 0.0652 (0.5, 0.0367)	0.0103 (0.0082)	0.9459 (0.9551)	6151, 99.5 % (6133, 99.3 %)
Maldi-tof (sau)	PLS1	5 (10)	0.0063 (0.0052)	0.9798 (0.9842)	0 (0)
Maldi-tof (sau)	PPLS1	2 (8)	0.0031 (0.0052)	0.9903 (0.9776)	0 (0)
Maldi-tof (sau)	PLS1/m VS	1 (2)	0.0021 (0.0083)	0.9878 (0.9862)	6161, 99.7 % (6128, 99.2 %)
Maldi-tof (sau)	PCR	7 (9)	0.0073 (0.0067)	0.9753 (0.9743)	0 (0)
Maldi-tof (sau)	Lasso	0.0259 (0.0133)	0.0033 (0.0034)	0.9850 (0.9850)	6157, 99.6 % (6137, 99.3 %)
Maldi-tof (sau)	Elastic net	0.9, 0.0288 (0.9, 0.0148)	0.0032 (0.0033)	0.9853 (0.9852)	6155, 99.6 % (6135, 99.3 %)

Tabell 8.1: *Resultater for den enklere modellen fra de ulike metodene på det uavhengige testsettet for alle datasettene. Resultatene for modellen med lavest feil i kryssvalideringen er i parentes. I de tilfellene der det ikke finnes noen enklere og ikke signifikant forskjellig modell fra den med lavest feil i kryssvalideringen markeres dette med (-).*

D Referanser

Birkebeiner-A/S. (2012). *Om selve rennet*. http://www.birkebeiner.no/Birkebeinerrennet/Om-selve-rennet/ (lest 04.05).

Efron, B., Hastie, T., Johnstone, I. & Tibshirani, R. (2004). Least Angle Regression. *The Annals of Statistics*, 32 (2): 407 - 499.

Friedman, J., Hastie, T., Hofling, H. & Tibshirani, R. (2007). Pathwise Coordinate Optimization. *The Annals of Applied Statistics*, 1 (2): 302 - 332.

Glmnet for Matlab. (2011). Lasso (L1) and elastic-net regularized generalized linear models Tilgjengelig fra: http://www-stat.stanford.edu/~tibs/glmnet-matlab/ (lest 06.05).

Hastie, T., Tibshirani, R. & Friedman, J. (2009). *The Elements of Statistical Learning*.

Helland, I. (2012). *PLS-kritikk* (Mail korrespondanse 12.01).

Ill-Vit-Redaksjon. (2004). Har årelating virket? *Illustrert Vitenskap*, 9: 12.

Indahl, U. G. (2005). A twist to partial least squares regression. *Journal of Chemometrics*, 19.

Lay, D. C. (2006). *Linear Algebra and Its Applications*.

Liland, K. H. (2009). *PLS in regression and classification: A historical review.* Prøveforelesning i forbindelse med forsvar av PhD, Ås.

Liland, K. H., Mevik, B. H., Rukke, E. O., Almøy, T. & Isaksson, T. (2009). Quantitative whole spectrum analysis with MALDI-TOF MS, Part II: Determining the

concentration of milk in mixtures. *Chemometrics and Intelligent Laboratory Systems*, 99: 39–48.

Martens, H. & Næs, T. (1989). *Multivariate Calibration*.

Montgomery, D. C., Peck, E. A. & Vining, G. G. (2001). *Introduction To Linear Regression Analysis*.

Osborne, B. G., Fearn, T., Miller, A. R. & Douglas, S. (1984). Application of near infrared spectroscopy to the compositional analysis of biscuits and biscuit doughs. *J. Sci. Food Agric.*, 35: 99–105

Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1988). *Numerical Recipes in C; The Art of Scientific Computing*. 2 utg.

Rence, J. (2012). *Positive Part*. http://mathworld.wolfram.com/PositivePart.html MathWorld Wolfram (lest 04.05).

Trane, C. (2012). Prestasjoner i lange løp - hvordan noen bakgrunnsfaktorer, trening, og langrennserfaring påvirket sluttiden til deltakerne i Birkebeinerrennet i 2011. http://www.hil.no/for_medarbeidere/aktuelt/thrane_om_paavirkninger_til_sluttiden_i_birkebeinerrennet: Høgskolen i Lillehammer.

Wold, H. (1966). Multivariate Analysis. *Academic Press, New York*: 391 - 420.

Wold, H. (1975). Path Models with Latent Variables. The NIPALS Approach. *Quantitative Sociology*: 307-357.

Zajaczkowski, T. (2010). Joseph Dietl (1804 - 1878). Innovator of medicine and his credit for urology. *Central European Journal of Urology*, 63 (2): 62 - 67.

Zou, H. & Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, 67 (2): 301 - 320.