Variabelreduksjon

Martin Jullum

Skatteetaten

01.09.21

Oversikt

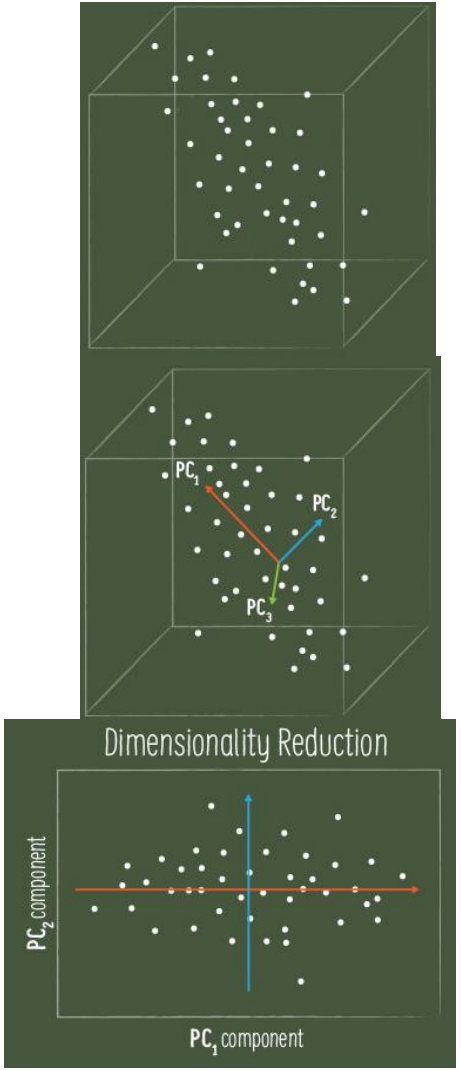
- Kort om variabelreduksjon
- **Ikke-stryrt variabelreduksjon**
- **Variabelreduksjon innen statistikk**
- **Variabelreduksjon innen maskinlæring**
- Utfordringer med med Skatteetatens modeller
- Forslag til metoder for Skatteetaten

Hva er variabelreduksjon

- Regresjonsmodell: $y \approx f(\boldsymbol{x}),\ \boldsymbol{x} = (x_1, \ldots, x_p)$
- Ønsker at p er liten, samtidig som approksimasjon er god
- Hvorfor
  - Mindre p => enklere modell, mindre varians (mer stabil modell), hindrer overtilpasning
  - Raskere trening/prediksjon
  - Enklere å visualisere og tolke* modell
  - Enklere å sikre god datakvalitet (inkl. mindre utfordringer med manglende verdier)
  - Personvern mer problematisk jo mer info man har

Ikke-styrt variabelreduksjon

- Bruk egenskaper ved $\boldsymbol{x}$ til å redusere dimensjonen
- Prinsipalkomponentanalyse (PCA):
  - Transformerer data iterativt til rom der hver nye komponent har størst mulig varians og står ortogonalt på de øvrige
  - Reduksjonsmetode: Inkluder kun de q første prinsipalkomponentene
- Alternativer
  - T-sne: Mer komplisert, ikke-lineær. Laget for visualisering i dim 2 eller 3.
  - ISOMAP: Komplisert, ikke-lineær.
- Utfordringer: Ødelegger tolkning, tar ikke hensyn til respons/modell



Variabelreduksjon innen statistikk

- I denne sammenheng
  - Statistisk modell = Modell $f(\boldsymbol{x};\boldsymbol{\theta})$ tilpasset ved maximum likelihood e.l., dim(θ) = p
- Variabel-reduksjon/-seleksjon = modellvalg for subset av $\boldsymbol{x} = (x_1, \dots, x_p)$. Totalt 2^p ulike modeller
- Bruker kun treningsdata
- Trade off mellom hvor godt modellen passer vs få parametere/variable

Variabelreduksjon innen statistikk 2

- Seleksjon basert på p-verdier
  - Eksempel (lineær) regresjon

$$f(\boldsymbol{x}) = \widehat{\beta_0} + \sum_{j=1}^{5} x_j \widehat{\beta_j}$$

  - P-Verdi for x_j: Sannsynligheten for å estimere en større $\widehat{\beta_j}$ dersom sann $\beta_j = 0$

```
Call:
lm(formula = y ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-2.6858 -0.4844  0.1352  0.6450  1.7453

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.1675     0.1384  -1.210  0.23281
x1            0.5306     0.1754   3.025  0.00414 **
x2           -0.4115     0.1769  -2.326  0.02470 *
x3            0.1289     0.1673   0.771  0.44510
x4           -0.5884     0.1818  -3.237  0.00230 **
x5           -0.2476     0.1432  -1.728  0.09094 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9507 on 44 degrees of freedom
Multiple R-squared:  0.5179,    Adjusted R-squared:  0.4631
F-statistic: 9.453 on 5 and 44 DF,  p-value: 3.568e-06
```

- Men hvordan velge beste modell?
  1. Fjern variable med høyest p-verdi
  2. Re-tilpass modell. Stopp hvis alle p-verdier < α (f.eks 0.05), ellers hopp til 1

Variabelreduksjon innen statistikk 3

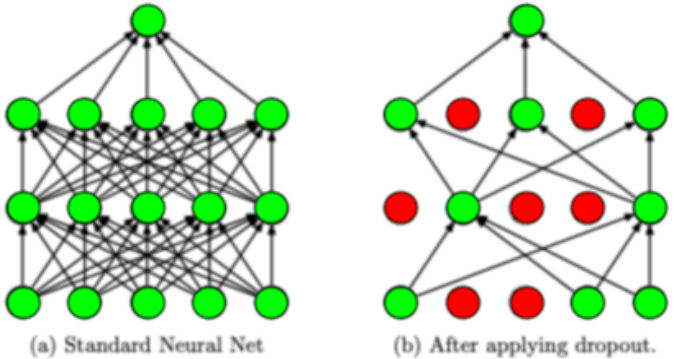
- Informasjonskriterier
  - Beregn en score for hvor "god" hver av de 2^p modellene er og straff for antall parameter. Velg modellen med best score.
  - Definisjoner av "score":
    - AIC = $2\log\left(L\left(\widehat{\boldsymbol{\theta}}\right)\right) - 2p$
    - BIC = $2\log\left(L\left(\widehat{\boldsymbol{\theta}}\right)\right) - \log(n)p$
    - FIC, DIC, GIC,…
- Hvis veldig mange modeller: Forward/backward-stagewise selection:
  - Backword: Start med modell med alle p variabler, og beregn AIC/BIC for alle modeller med $p-1$ variabler. Velg beste og fortsette til ingen forbedring
  - Forward: Start med ingen variabler og legg til en og en tilsvarende
  - Grådige (greedy) algoritmer -> Finner sjelden optimalt subset, men kan være nær.

Variabelreduksjon innen maskinlæring

- Mest vanlig med automatisk/implisitt variabelreduksjon som reduserer utnyttelsen av variablene
  - ML-metoder gir ofte best prediksjoner med ***tilgang*** på alle variabler
- Regularisering
  - Lasso: Lineær regresjon $f(\boldsymbol{x}) = \widehat{\beta_0} + \sum_{j=1}^{5} x_j \widehat{\beta_j}$ som straffer store verdier av $\widehat{\beta_i}$ med L_1-norm som setter noen til 0 (variabelreduksjon)
  - Ridge: Som Lasso, men L_2-norm, som kun krymper alle $\widehat{\beta_j}$, gir redusert antall *effektive* parameter/frihetsgrader
  - Brukes også ofte i nevrale nett og tre-boosting metoder (xgboost, lightgbm, catboost)

Variabelreduksjon innen maskinlæring 2

- Kolonne-sampling
  - For random forest og tre-boosting:
    - Hvert tre trenes med et tilfeldig trukket utvalg av variabler fra hele datasettet
- Drop-out
  - For nevrale nett (og tre-boosting):
    - Ved hvert nye steg i treningen (ny epoch eller tre), fjernes enkelte noder/trær når man trener/oppdaterer parametere for å ikke legge for mye vekt på enkelte deler observasjoner/ variabler

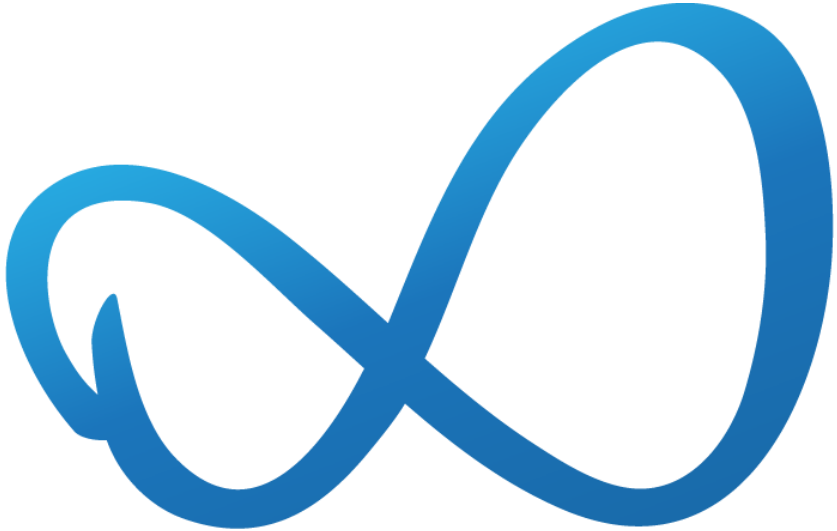


(a) Standard Neural Net (b) After applying dropout.

- Finnes mange flere teknikk for å hindre overtilpasning

Variabelreduksjon innen maskinlæring 3

- Finnes også teknikker for ***reell*** variabelreduksjon
  - Dvs reduksjon av dim($\boldsymbol{x}$) $= p$ for $\boldsymbol{x}$ som inngår i $f(\boldsymbol{x})$.
- Boruta
  - Repeter følgende K ganger:
    - Legg til rad-randomiserte kopier av hver variabel til originalt datasett
    - Beregn feature importance
  - Inkluder alle variabler som hadde høyere score enn alle kopi-variablene i f.eks. 95% av kjøringene.
- Noen tilpasser også en statistisk modell på forhånd og gjøre variabelreduksjon med den (Lasso, forward/backword)



BigInsight