Vedlegg A

I. BESKRIVELSE AV LOGSITISK REGRESJON

A. Binær logsistisk regresjon

For å modellere sannsynlighet med binær logistsik regresjon, brukes sigmoidfunksjonen som tar verdier mellom 0 og 1:

$$\sigma(z) = \frac{1}{1 + \exp(-z)} = \frac{\exp z}{1 + \exp z}.$$

Her er z representert som $z = \beta_0 + \beta_1 x$, der β_0 og β_1 er parametere som må tilpasses i modellen. Da blir

$$p(x) = \frac{\exp(\beta_0 + \beta_1 x)}{1 + \exp)(\beta_0 + \beta_1 x)}$$

som kan generaliseres til

$$p(\boldsymbol{x}) = \frac{\exp(\boldsymbol{w}\boldsymbol{x} + \boldsymbol{b})}{1 + \exp(\boldsymbol{w}\boldsymbol{x} + \boldsymbol{b})},$$

i tilfellet hvor inndataene våre består av mer enn én parameter. I et datasett bestående av sammenhørende verdier $\boldsymbol{x_i}$ og y_i, vil da sannsynligheten for gitt utfall være

$$P = \prod_{i=1}^{n} \left[p(y_i = 1|\boldsymbol{x_i}, \boldsymbol{w}, \boldsymbol{b})\right]^{y_i} \left[1 - p(y_i = 1|\boldsymbol{x_i}, \boldsymbol{w}, \boldsymbol{b}))\right]^{1-y_i}.$$

Vi ønsker å finne verdiene $\boldsymbol{w}$ og $\boldsymbol{b}$ som gir størst mulig sannsynlighet, dvs at modellen vår passer best mulig med det som er observert. For å gjøre det enklere, så tar vi den naturlige logaritmen av denne funksjonen

$$\log(P) = \sum_{i=1}^{n} (y_i \log p(y_i = 1|\boldsymbol{x_i}, \boldsymbol{w}, \boldsymbol{b}) + (1 - y_i) \log\left[1 - p(y_i = 1|\boldsymbol{x_i}, \boldsymbol{w}, \boldsymbol{b})\right]).$$

Hvis vi så setter inn $p(\boldsymbol{x})$ i denne får vi et uttrykk som kalles for *cross entropy* som benyttes som kost-funksjon i logistisk regresjon:

$$\mathcal{C}(\boldsymbol{w}, \boldsymbol{b}) = -\sum_{i=1}^{n} \left(y_i(\boldsymbol{w}\boldsymbol{x_i} + \boldsymbol{b}) - \log\left(1 + \exp\left(\boldsymbol{w}\boldsymbol{x_i} + \boldsymbol{b}\right)\right)\right).$$

Optimering, eller trening, av modellen består så i å finne det beste settet med parametere $\boldsymbol{w}$ og $\boldsymbol{b}$. Deretter kan modellen benyttes til å klassifisere ukjente tilfeller gjennom å beregne sannsynligheten $p(\boldsymbol{x})$ og en på forhånd bestemt terskelverdi for å angi hvilken klasse dne hører hjemme i.

B. Klassifisering med mer enn to klasser

Dersom klassifiseringsproblemet har mer enn to klasser, må modellen over tilpasses for å ta høyde for mer enn to utfall. Sannsynligheten for et gitt utfall blir nå

$$P = \prod_{i=1}^{n} \prod_{k=1}^{K} p_k(\boldsymbol{x_i}, \boldsymbol{w}, \boldsymbol{b})^{y_i},$$

hvor det er totalt K klasser. Sannsynlighetsmodellen må også tilpasses at det er flere klasser og dette gjøres gjennom å erstatte sigmodifunksjonen med *softmax*:

$$p_k(\boldsymbol{x_i}) = \frac{\exp\left(\boldsymbol{w_k}\boldsymbol{x} + \boldsymbol{b_k}\right)}{\sum_{l=1}^{K-1} \exp\left(\boldsymbol{w_k}\boldsymbol{x} + \boldsymbol{b_k}\right)}.$$

Totalt sett får vi da til slutt en kostfunksjon gitt ved:

$$\mathcal{C}(\boldsymbol{w}, \boldsymbol{b}) = -\sum_{i=1}^{n} \sum_{k=1}^{K} \log \frac{\exp\left(\boldsymbol{w_k}\boldsymbol{x} + \boldsymbol{b_k}\right)}{\sum_{l=1}^{K-1} \exp\left(\boldsymbol{w_k}\boldsymbol{x} + \boldsymbol{b_k}\right)}.$$

Denne kostfunksjoen kan så benyttes for å finne optimale verdier av modellparameterne. Men for å gjøre prediskjoner i flerklasseklassifisering, klassifisering med mer enn to utfall, må det beregnes en sannsynlighet for hver av klassene. Deretter kan man f.eks. benytte såkalt *one hot encoding* som vil si at man plasserer tilfellet i den klassen som har høyest sannsynlighet. På denne måten vil man i alle tilfeller tildele en klasse. Modellen vi da predikere kun én klasse selv om det var flere klasser som hadde høy sannsynlighet og også selv om ingen av klassene hadde høy sannsynlighet.