

IN2110: Språkteknologiske metoder

Logistisk Regresjon

Helene Bøsei Olsen

Språkteknologigruppen (LTG)

20 Februar, 2024



- ▶ Veiledet maskinlæringsmodell for klassifisering
- ▶ Et av de viktigste verktøyene vi har i språkteknologi
 - ▶ Og i mange andre fagfelt
- ▶ Ofte brukt som en baselinemodell
- ▶ Også en viktig byggekloss i mer avanserte maskinlæringsmodeller (bl.a. nevrale nettverk)

Del 1: Hva er logistisk regresjon?

- ▶ Klassifikasjon
- ▶ Modellen:
 - ▶ Lineær regresjon
 - ▶ Sigmoid (binær klassifisering)
 - ▶ Softmax (flere klasser)
- ▶ Fordeler og ulemper

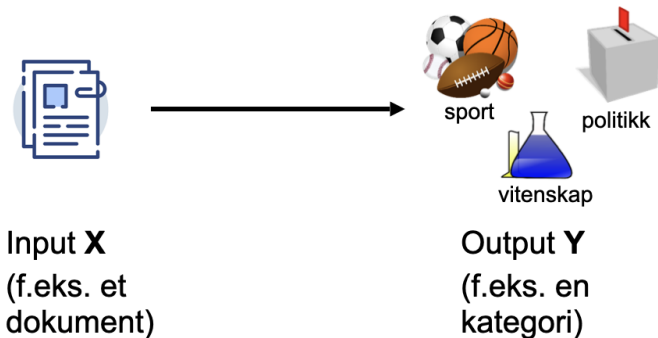
Del 2: Hvordan lærer man en logistisk regresjonsmodell ut fra data?

- ▶ Trening
- ▶ + mer om det neste uke

En klassifikasjonsmodell



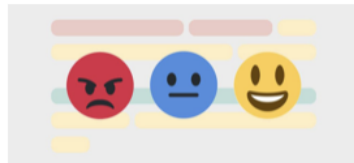
Logistisk regresjon = en **klassifikasjonsmodell**



Eksempler

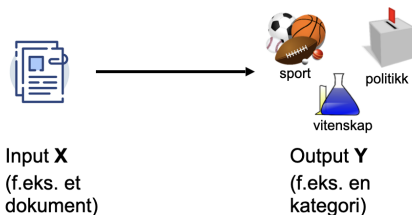
- ▶ Spam-identifisering (Spam/ikke spam)
- ▶ Språkidentifisering (Norsk/Engelsk)
- ▶ Sentimentanalyse (Positivt/negativt)

- **Mats Dotterud-Flaa, Skatte...** Friday >
[TO] Automatisk e-post
Skatteetaten <[https://](https://naturalselectionsynergy.lt.acemlnc.com/Prod/li...)
naturalselectionsynergy.lt.acemlnc.com/Prod/li...
- **Luckyeo** 31/01/2024 >
[TO] Re: About your order
Coupon inside, hurry >
...
- **"Refinansier" <Refinan...** 26/01/2024 >
[TO] Refinansier gjeld selv med inkasso og betal...
<[http://modilas.digital/rd/](http://modilas.digital/rd/c29204heSgQ77579449gaaf285916jsl4976k...)
[c29204heSgQ77579449gaaf285916jsl4976k...](http://modilas.digital/rd/c29204heSgQ77579449gaaf285916jsl4976k...)



- ▶ Binær klassifikasjon:
 - ▶ 2 klasser
 - ▶ True eller False
 - ▶ hvert datapunkt blir tildelt én klasse
- ▶ Multi-klasse klassifikasjon
 - ▶ Antall klasser > 2
 - ▶ hvert datapunkt blir tildelt én klasse
- ▶ Multi-label klassifikasjon
 - ▶ Antall klasser > 2
 - ▶ ikke gjensidig utelukkende
 - ▶ Hvert datapunkt tildeles flere klasser





- ▶ En logistisk regresjonsmodell gir oss sannsynligheten $P(y | x)$ for en outputklasse y gitt input x
- ▶ y : outputklassen (valgt ut fra en mengde mulige klasser vi har definert på forhånd)
- ▶ x : input (representert som numeriske trekk (features))

- ▶ For logistisk regresjon må vi definere hvilke trekk som skal representere dataen
- ▶ For hvert trekk x_i , har vi en vekt w_i som sier noe om hvor viktig x_i er.
- ▶ Vektene kan være positive eller negative

- ▶ For logistisk regresjon må vi definere hvilke trekk som skal representere dataen
- ▶ For hvert trekk x_i , har vi en vekt w_i som sier noe om hvor viktig x_i er.
- ▶ Vektene kan være positive eller negative

Eksempel Sentiment:

- ▶ x_i = Dokumentet inneholder ordet "Fantastisk": $w_i = +10$
- ▶ x_j = Dokumentet inneholder ordet "Kjedelig": $w_j = -10$
- ▶ x_k = Dokumentet inneholder ordet "Middelmådig": $w_k = -2$

Vi starter med å se på “vanlig” lineær regresjon

- ▶ Vi ønsker å predikere boligpriser i Oslo
- ▶ Vi bruker to trekk:
 1. Størrelse på bolig (kvm)
 2. Beliggenhet (Østkant / Vestkant)

Vi starter med å se på “vanlig” lineær regresjon

- ▶ Vi ønsker å predikere boligpriser i Oslo
- ▶ Vi bruker to trekk:
 1. Størrelse på bolig (kvm)
 2. Beliggenhet (Østkant / Vestkant)
- ▶ Vi kan estimere en regresjonsmodell basert på et dataset av eksempler

kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

$$y = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b$$

Output

Input $\mathbf{x}$ med n
(numeriske) features

skjæringspunktet
(=verdi av y hvis alle
features i $\mathbf{x}$ er lik 0)

$$y = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b$$

Diagram illustrating the linear regression equation $y = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b$.

- Output**: y
- Input $\mathbf{x}$ med n (numeriske) features**: $x_1, x_2, \dots, x_n$ (features)
- skjæringspunktet (=verdi av y hvis alle features i $\mathbf{x}$ er lik 0)**: b (intercept)

- ▶ y : Output
- ▶ $\mathbf{x}$: Input med n antall numeriske trekk (features) $[x_1, x_2, \dots, x_n]$
- ▶ $\mathbf{w}$: vektene i modellen $[w_1, w_2, \dots, w_n]$
 - ▶ Hver vekt er assosiert med et av input-trekkene
- ▶ b : skjæringspunktet (intercept)
- ▶ Vektene $\mathbf{w}$ og skjæringspunktet b estimeres ut fra treningsdata (Neste ukes forelesning)

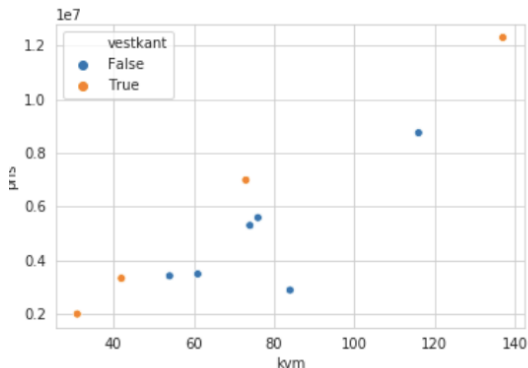
Denne formelen:

$$y = w_1x_1 + w_2x_2 + \dots + w_nx_n + b$$

Kan også skrives slik:

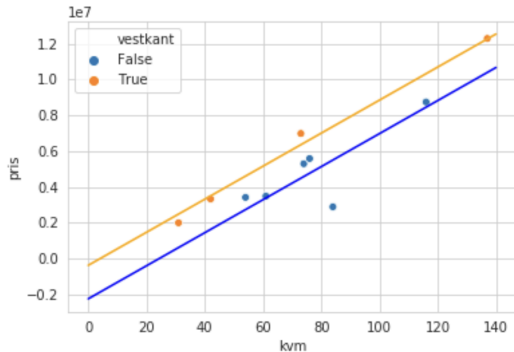
$$\boxed{y = \mathbf{w}^T \mathbf{x} + b} \text{ hvor } \mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \text{ og } \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Tilbake til eksemplet:



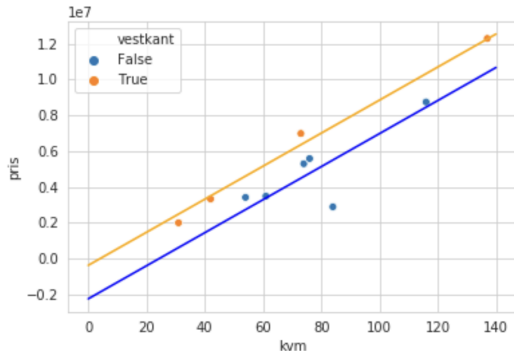
kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

Tilbake til eksemplet:



kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

Tilbake til eksemplet:



kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

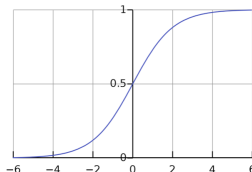
Hva skjer hvis vi har en liten bolig (kvm < 20)?

Kan vi bruke regresjonsformelen til klassifisering?



- ▶ Regresjon gir oss verdier for y mellom $-\infty$ og $+\infty$
- ▶ Målet vårt er å trene en klassifiseringsmodell
- ▶ Vi vil vite hva er $P(y = 1 \mid x)$
 - ▶ *Hva er sannsynligheten for klassen gitt input*
- ▶ Output må være en sannsynlighet (tall mellom 0 og 1)
- ▶ **Løsning:** Sigmoid funksjonen!

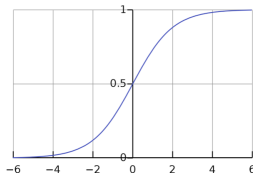
- ▶ Sigmoid = σ = logistic function
- ▶ Tvinger output til å være mellom 0 og 1.
- ▶ Bruker for binær klassifisering (to klasser)
- ▶ **Logistisk regresjon**: Beregne $\mathbf{w}^T \mathbf{x} + b$ og sende det gjennom Sigmoid



Sigmoid

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

- ▶ Sigmoid = σ = logistic function
- ▶ Tvinger output til å være mellom 0 og 1.
- ▶ Bruker for binær klassifisering (to klasser)
- ▶ **Logistisk regresjon**: Beregne $\mathbf{w}^T \mathbf{x} + b$ og sende det gjennom Sigmoid



Sigmoid

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Lineær regresjon = $\mathbf{w}^T \mathbf{x} + b$

Logistisk regresjon = $\sigma(\mathbf{w}^T \mathbf{x} + b)$

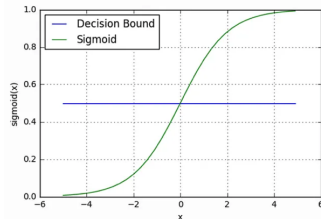
- ▶ Hvis $\mathbf{w}^T \mathbf{x} + b$ er
 - ▶ Veldig negativ (e.g. -100) 0
 - ▶ Veldig positiv (e.g. 100) 1

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$


Tallet vi får fra $\mathbf{w}^T \mathbf{x} + b$

Logistisk regresjon: Sigmoid

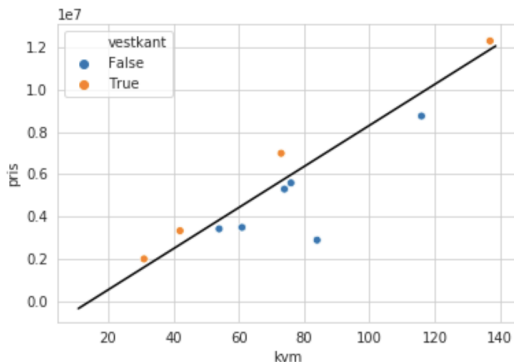
- ▶ Vi har nå et tall mellom 0 og 1
- ▶ Hvordan gjøre en klassifikasjon?
- ▶ **Beslutningsregel:**
 - ▶ 1 hvis sannsynligheten > 0.5
 - ▶ 0 hvis sannsynligheten < 0.5



- ▶ Tilbake til eksemplet
- ▶ Denne gangen med **logistisk regresjon**
- ▶ Nå skal vi predikere om boligen er på østkanten eller vestkanten (binær)
- ▶ Vi bruker kvm og pris som trekk (features)

kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

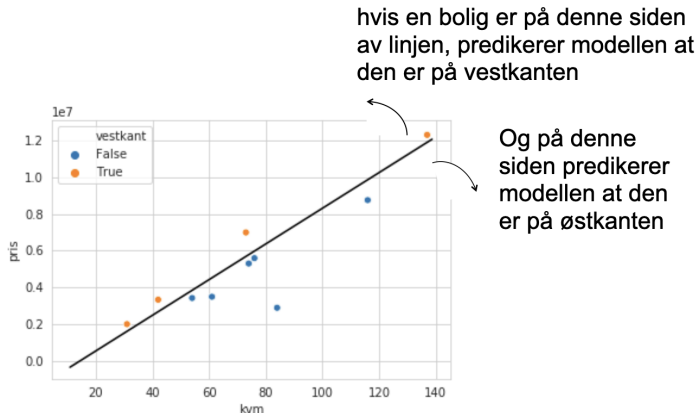
Logistisk regresjon: Boligeksempel



kvm	Vestkant?	Pris
84	Nei	2 890 000
54	Nei	3 420 000
73	Ja	6 990 000
74	Nei	5 300 000
137	Ja	12 300 000
76	Nei	5 590 000
31	Ja	2 000 000
61	Nei	3 490 000
116	Nei	8 750 000
42	Ja	3 330 000

Logistisk regresjon er en lineær modell

Beslutningsregelen er en vektet sum av trekkene (features) fra input



NB: Jo lengre avstanden er mellom punkt og linja, desto høyere er predikerte sannsynligheten.



- ▶ Vi har trent en modell ved hjelp av treningssettet og brukt kvm og pris som trekk/features (x)

- ▶ Vi har trent en modell ved hjelp av treningssettet og brukt kvm og pris som trekk/features (x)

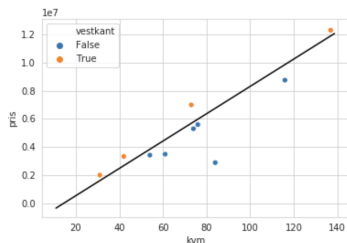
- ▶ Modellen har estimert:

Vektene (w):

- ▶ $w_{\text{kvm}} = -8.15$,
- ▶ $w_{\text{pris}} = 0.09$

Og skjæringspunktet (b):

- ▶ $b = 104$



- ▶ Vi har trent en modell ved hjelp av treningssettet og brukt kvm og pris som trekk/features (x)

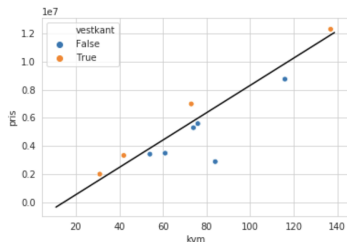
- ▶ Modellen har estimert:

Vektene (w):

- ▶ $w_{\text{kvm}} = -8.15$,
- ▶ $w_{\text{pris}} = 0.09$

Og skjæringspunktet (b):

- ▶ $b = 104$



- ▶ Nå kan vi bruke formelen til å predikere hvor sannsynlig det er for at en ny bolig er på vestkanten.

► Oppgave:

Hvis boligen er 97 kvm og er solgt for 7650 tkr, hvor sannsynlig er det at boligen er på vestkanten i følge modellen?

Vektene (w)

$$w_{\text{kvm}} = -8.15,$$

$$w_{\text{pris}} = 0.09$$

Skjæringspunktet (b)

$$b = 104$$

Hovedformel logistisk regresjon:

$$y = \sigma(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)$$

$$\text{hvor } \sigma(z) = \frac{1}{1 + e^{-z}}$$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$
- ▶ Formelen:

$$y = \sigma(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)$$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$
- ▶ Formelen:

$$y = \sigma(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)$$

- ▶ Vi setter inn tallene:

$$y = \sigma((-8.15 \times 97) + (7650 \times 0.09) + 104)$$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$
- ▶ Formelen:

$$y = \sigma(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)$$

- ▶ Vi setter inn tallene:

$$y = \sigma((-8.15 \times 97) + (7650 \times 0.09) + 104)$$

- ▶ Neste steg er Sigmoid:

$$y = \sigma(1.95) = \frac{1}{1 + e^{-1.95}} = 0.8754$$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$
- ▶ Formelen:

$$y = \sigma(w_1x_1 + w_2x_2 + \cdots + w_nx_n + b)$$

- ▶ Vi setter inn tallene:

$$y = \sigma((-8.15 \times 97) + (7650 \times 0.09) + 104)$$

- ▶ Neste steg er Sigmoid:

$$y = \sigma(1.95) = \frac{1}{1 + e^{-1.95}} = 0.8754$$

- ▶ Så $P(y = \text{vestkant} \mid x) = 0.8754$

- ▶ Vi har $x_1 = 97$ (kvm) og $x_2 = 7650$ (pris)
- ▶ Og $w_1 = -8.15$, $w_2 = 0.09$, og $b = 104$
- ▶ Formelen:

$$y = \sigma(w_1x_1 + w_2x_2 + \dots + w_nx_n + b)$$
- ▶ Vi setter inn tallene:

$$y = \sigma((-8.15 \times 97) + (7650 \times 0.09) + 104)$$
- ▶ Neste steg er Sigmoid:

$$y = \sigma(1.95) = \frac{1}{1 + e^{-1.95}} = 0.8754$$
- ▶ Så $P(y = \text{vestkant} \mid x) = 0.8754$

I Python:

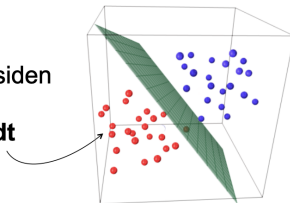
```
In [1]: import math
```

```
In [2]: 1/(1+math.exp(8.15*97 - 0.09*7650 -104))
```

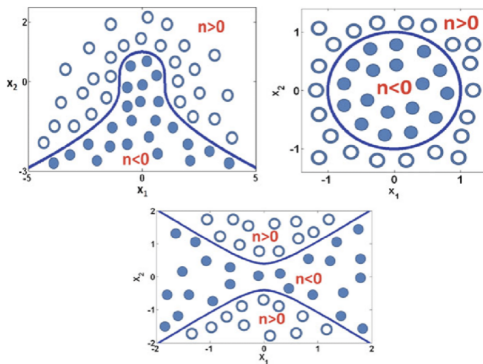
```
Out[2]: 0.875446641812576
```

- ▶ Antall trekk/features er ofte langt høyere enn 2
 - ▶ Mange språktechnologiske modeller kan ha tusenvis av features
 - ▶ For eksempel “bag-of-words” features som indikerer om et ord forekommer eller ikke
- ▶ Når antall trekk/features > 2 er beslutningsregelen en (hyper)plan

Punkter på denne siden
av planen vil være
klassifisert som **rødt**



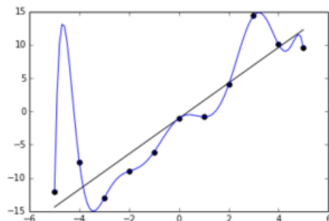
- Logistisk regresjon og andre lineære modeller er ikke alltid et godt valg



1. Kan trenes på **beskjedne datamengder**

- ▶ Fordi lineære modeller har (relativt) få parametre som må estimeres: vektene w og skjæringspunktet b
- ▶ Til sammenlikning kan store nevrale nettverkt inneholde millioner av parametre og krever veldig store mengder treningsdata

1. Kan trenes på **beskjedne datamengder**
2. Gode evner til **generalisering**
 - ▶ = Mindre risiko for overtrening
 - ▶ Fordi modellen er enkel, vil den generalisere ganske bra fra treningsdata til nye, ukjente datapunkter



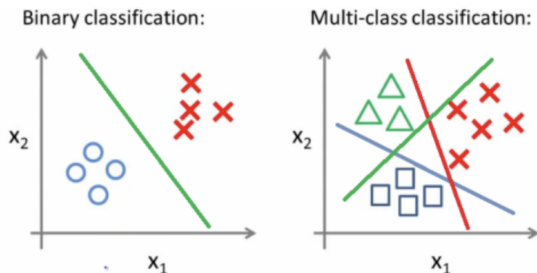
1. Kan trenes på **beskjedne datamengder**
2. Gode evner til **generalisering**
3. **Rask** (trening og prediksjon) og **skalerbar**
 - ▶ Kan skalere til store dataset (med millioner av datapunkter eller millioner av features)
 - ▶ kan kjøres på egen maskin

1. Kan trenes på **beskjedne datamengder**
2. Gode evner til **generalisering**
3. **Rask** (trening og prediksjon) og **skalerbar**
4. **Forklarbar**
 - ▶ Vi kan forstå hvilke deler av input som påvirker modellen
(Mer om det neste uke)



- ▶ Logistisk regresjon med sigmoid σ er begrenset til 2 klasser (Vestkant/ikke-vestkant)
- ▶ For K antall klasser (hvor $K > 2$) bruker vi **multinomial logistisk regresjon** ("multi-klasse")
- ▶ Og vi erstatter sigmoid med **softmax**

- ▶ I multinomial logistisk regresjon har vi én beslutningsregel per klasse:



- ▶ Det betyr at vi også har:
- ▶ Separate vektorer (w) for hver klasse
- ▶ Separat skjæringspunkt (b) for hver klasse

- ▶ Vi bruker fortsatt den samme formelen for den lineære funksjonen, men vi må utvide den til flere klasser.
- ▶ Vi må erstatte Sigmoid σ med **Softmax**:

$$P(y = i \mid \mathbf{x}) = \text{softmax}(\mathbf{w}_i^T \mathbf{x} + b_i)$$

$$\text{hvor } \text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

- :
- ▶ Sannsynlighet for å tilhøre en klasse i er igjen basert på en vektet sum $\mathbf{x}$
 - ▶ og deretter normalisert med Softmax slik at vi får en sannsynlighetsfordeling



$\mathbf{w}_i^T \mathbf{x} + b_i = [z_1, z_2, \dots, z_k]$ hvor k er antall klasser



$\mathbf{w}_i^T \mathbf{x} + b_i = [z_1, z_2, \dots, z_k]$ hvor k er antall klasser

Input til softmax = $[z_1, z_2, \dots, z_k] = [0.6, 1.1, -1.5, 1.2, 3.2, -1.1]$

$\mathbf{w}_i^T \mathbf{x} + b_i = [z_1, z_2, \dots, z_k]$ hvor k er antall klasser

Input til softmax = $[z_1, z_2, \dots, z_k] = [0.6, 1.1, -1.5, 1.2, 3.2, -1.1]$

$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$ er det samme som:

$$\text{softmax}(z) = \left[\frac{e^{z_1}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_2}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_k}}{\sum_{j=1}^K e^{z_j}} \right]$$

$\mathbf{w}_i^T \mathbf{x} + b_i = [z_1, z_2, \dots, z_k]$ hvor k er antall klasser

Input til softmax = $[z_1, z_2, \dots, z_k] = [0.6, 1.1, -1.5, 1.2, 3.2, -1.1]$

$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$ er det samme som:

$$\text{softmax}(z) = \left[\frac{e^{z_1}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_2}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_k}}{\sum_{j=1}^K e^{z_j}} \right]$$

Nevneren $\sum_{j=1}^K e^{z_j}$ brukes for å normalisere alle verdiene

$\mathbf{w}_i^T \mathbf{x} + b_i = [z_1, z_2, \dots, z_k]$ hvor k er antall klasser

Input til softmax = $[z_1, z_2, \dots, z_k] = [0.6, 1.1, -1.5, 1.2, 3.2, -1.1]$

$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$ er det samme som:

$$\text{softmax}(z) = \left[\frac{e^{z_1}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_2}}{\sum_{j=1}^K e^{z_j}}, \frac{e^{z_k}}{\sum_{j=1}^K e^{z_j}} \right]$$

Nevneren $\sum_{j=1}^K e^{z_j}$ brukes for å normalisere alle verdiene

Resultatet blir en sannsynlighetsdistribusjon:

Input: $[0.6, 1.1, -1.5, 1.2, 3.2, -1.1]$

Output: $[0.055, 0.090, 0.006, 0.099, 0.74, 0.010]$

Binær

- ▶ 2 Klasser:
- ▶ Lineær funksjon:
$$z = \mathbf{w}^T \mathbf{x} + b$$

Hvor z er ett tall
- ▶ z gis som input til Sigmoid σ
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$
- ▶ Output er en sannsynlighet (ett tall) mellom $[0, 1]$

Multinomial

- ▶ K antall klasser ($K > 2$)
- ▶ Lineær funksjon:
$$z_k = \mathbf{w}_i^T \mathbf{x} + b_i$$
- ▶ Hvor z består av like mange tall som antall klasser (k)
- ▶ z_k gis som input til softmax
$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$
- ▶ Output er en sannsynlighets-distribusjon



- ▶ La oss si at vi lager en modell for tekstklassifisering basert på (multinomial) logistisk regresjon, med 10 klasser.
- ▶ Modellen tar i bruk 120 features, og blir trent på et datasett med 300 000 tekster
- ▶ **Spørsmål:** Hvor mange parametre inneholder modellen?



- ▶ Parametre: “Interne innstillingene” modellen lærer fra treningsdata.
- ▶ I logistisk regresjon lærer modellen to typer parametre: vektene ($\mathbf{w}$) og skjæringspunktet (b)
- ▶ Når vi har multinomial logistisk regresjon, lærer modellen separate vektorer for hver klasse i tillegg til ett skjæringspunkt per klasse.
- ▶ Da har vi:
 - ▶ 10 skjæringspunkt (b) +
 - ▶ 120 trekk (vektene ($\mathbf{w}$) $\times$ 10 klasser)
 - ▶ = 1210 parametre

Logistisk regresjon

- ▶ En veiledet klassifiseringsmodell
- ▶ Bruker formelen fra lineær regresjon

$$w_1x_1 + w_2x_2 + \dots + w_nx_n + b$$

- ▶ Binær klassifisering:

Sigmoid: $\sigma = \frac{1}{1+e^{-z}}$

- ▶ omgjør output til $[0,1]$

- ▶ Multiklasse klassifisering (multinomial):

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}}$$

- ▶ output er en sannsynlighetsdistribusjon

- ▶ Vektene og skjæringspunktet estimeres under trening.

- ▶ Utgjør parameterne til modellen
- ▶ Multinomial: Estimerer ett skjæringspunkt og et separat sett med vekter per klasse