



Prediktiv analys

FÖRELÄSNING 6

Dagens fråga

- ♦ Vad är det dyraste du haft sönder?



Dagens agenda

- ♦ Klassificering
- ♦ Olika typer av klassificering metoder,
- ♦ Vad sannolikhet är och hur det används i klassificering
- ♦ Modellen Logistic regression
- ♦ Kategoriska features: Ordinal encoding, One-hot encoding, dummy variabel



Förra föreläsning

- ♦ Regression i Python
- ♦ KNN
- ♦ Model evaluation för regression
 - Cross validation
 - Overfitting
 - Bias-variance tradeoff
 - Regularization
 - Feature selection
- ♦ Lasso regression
- ♦ Diamond prices



Klassifikation



Förutse kategorier



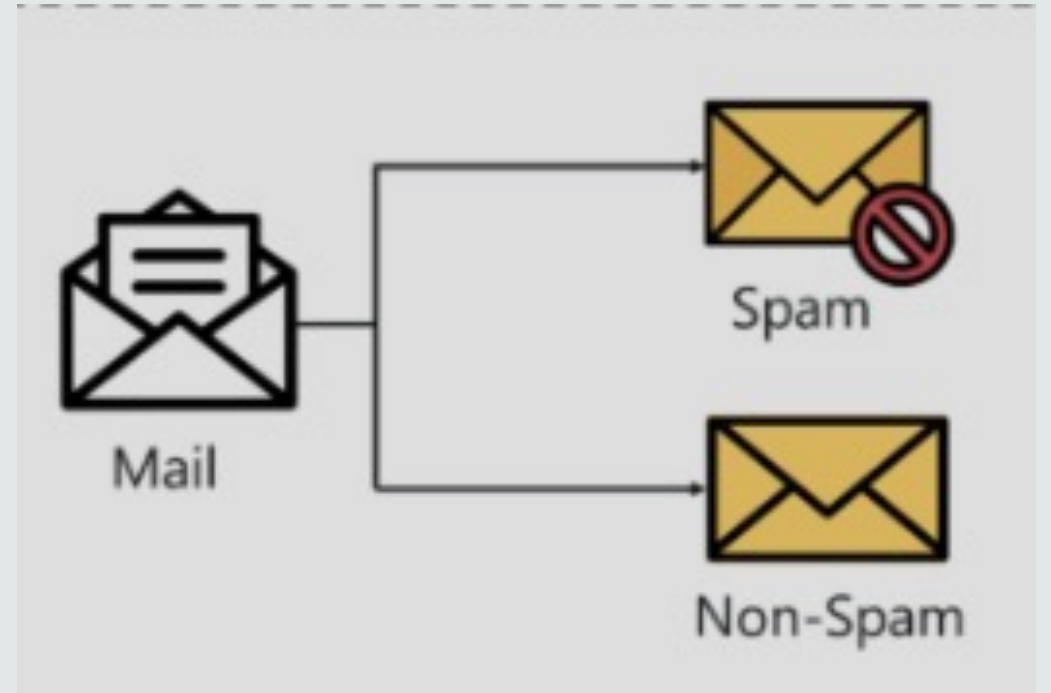
När målet (target) (output, dependent variable) är en kategorisk variabel



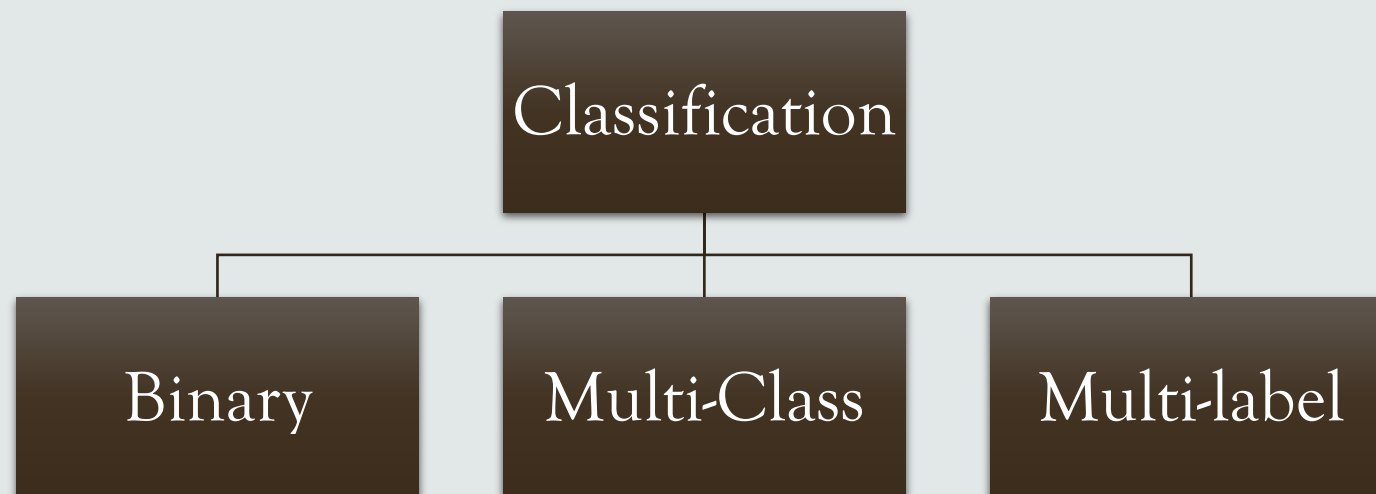
T.ex. hund, katt. Ja/nej osv

Klassificering

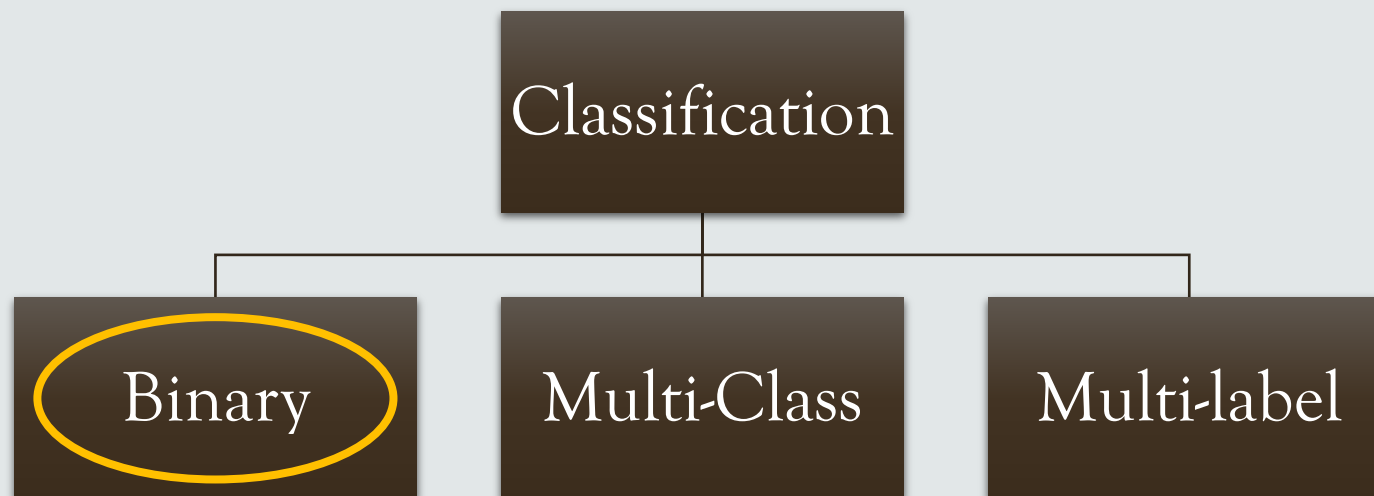
- Klassificering är när datan kategoriseras in i klasser
- Klasserna kallas ofta för *traget*, *label* (*etiketter*), *kategori*
- Output variabeln är *diskret* vilket betyder det finns ett ändligt antal värden



Typer av klassifisering



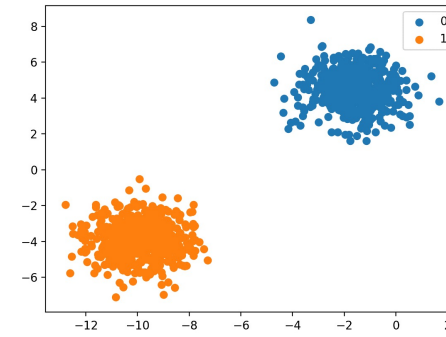
Typer av klassifiering



Vi kommer mestadels jobba med Binary Classification men
förståelsen här används till övriga

Binary Classification

- Binära (binary) tal är uttryckt i 2-talbas och använder endast två siffror, oftast 0 och 1
- Två kategorier:
 - Positiva klassen == 1
 - Negativa klassen == 0
- Exempel
 - Spam vs icke spam
 - Sjuk vs frisk
- Det första vi gör är att välja en av dessa kategorier och benämner den som "positiv"
- Positiv i denna kontexten betyder inte att kategorin är bra eller det man strävar mot. Det är bara en konvention vi använder och det spelar ingen roll vilken kategori vi väljer ur den matematiska eller prediktiva analysens synvinkel.





Vad är sannolikhet (probability)?

- Analysen av slumpmässiga (random) händelser.
- När händelser inte kan förutsägas med total säkerhet. Då kan vi i stället säga hur *sannolikt* de kommer att hända med att använda sannolikhetsteori.
- Sannolikheten för en händelse är ett tal mellan 0 och 1. Ju högre sannolikheten är (närmare 1) ju mer sannolikt är det att händelsen inträffar.
- Klassisk exempel är att kasta en mynt med två sannolika utfall slant eller krona med sannolikheten för båda händelser är 0,5.

Vad är sannolikhetsvektor?

$$p = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}, \quad \sum_{i=1}^n p_i = 1$$

- En sannolikhetsvektor är en vektor med icke-negativa tal som summerar till 1
- Den representerar de möjliga resultaten för en slumpmässig diskret variabel
- Den ger *probability mass function* som är hur man karakteriserar en diskret sannolikhetsfördelning (probability distribution)

$$x_1 = \begin{bmatrix} 0.5 \\ 0.25 \\ 0.25 \end{bmatrix}, \quad x_2 = [0 \quad 1 \quad 0]$$

Probabilistic Classification

- Probabilistic (sannolikhet) klassificering betyder att modellen som används är en *sannolikhetsmodell*.
- Man får en sannolikhetsdistribution för klasserna i stället för att prediktera vilken klass det ska vara. Så output y blir en sannoliketsvektorer
- För binary classification blir det sannolikheten för positivt eller negativt utfall

Enkel logistisk regression (Logistic Regression)



TYPER AV
KLASSIFIKATIONSPPGIFTER



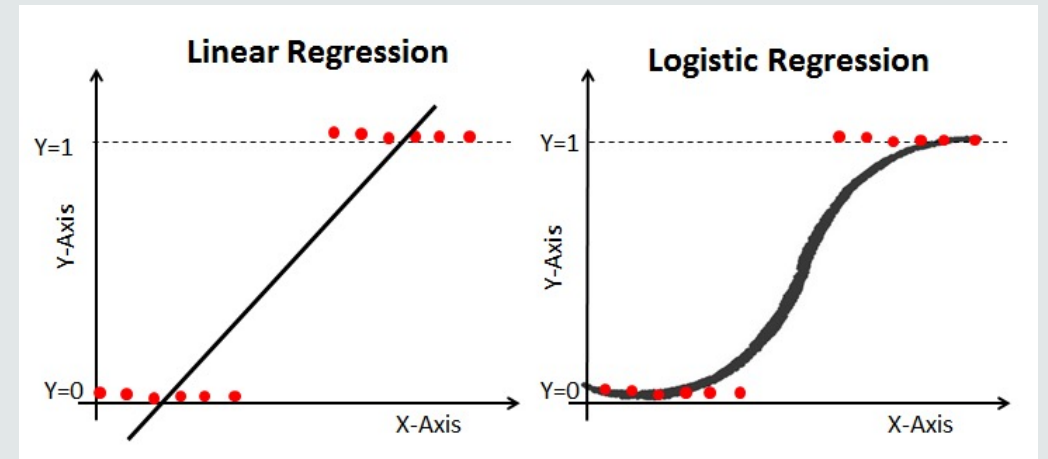
INTENTIONEN BAKOM
“LOGISTIC REGRESSION
MODEL”



“THE LOGISTICREGRESSION
OBJECT” FRÅN SCIKIT-LEARN

Logistic Regression

- Logistisk regression är en modell som används när beroende variabel y är kategorisk
- *Exempel: en mail är spam (1) eller inte (0)*
- Metoden lämpar sig bäst när man är intresserad av att undersöka om det finns ett samband mellan en beroende variabel y som endast kan anta två möjliga värden och oberoende variabler x .
- Den ger sannoliketen för vilken klass target y ska ha



Den generella formeln för Logistic Regression

Hur estimeras dessa sannolikheter?

För att hitta sannolikheten att target y är lika med 1 (target är i positiva klassen) används formeln:

$$Pr(target = 1|X) = \frac{1}{1 + e^{-Z}}$$

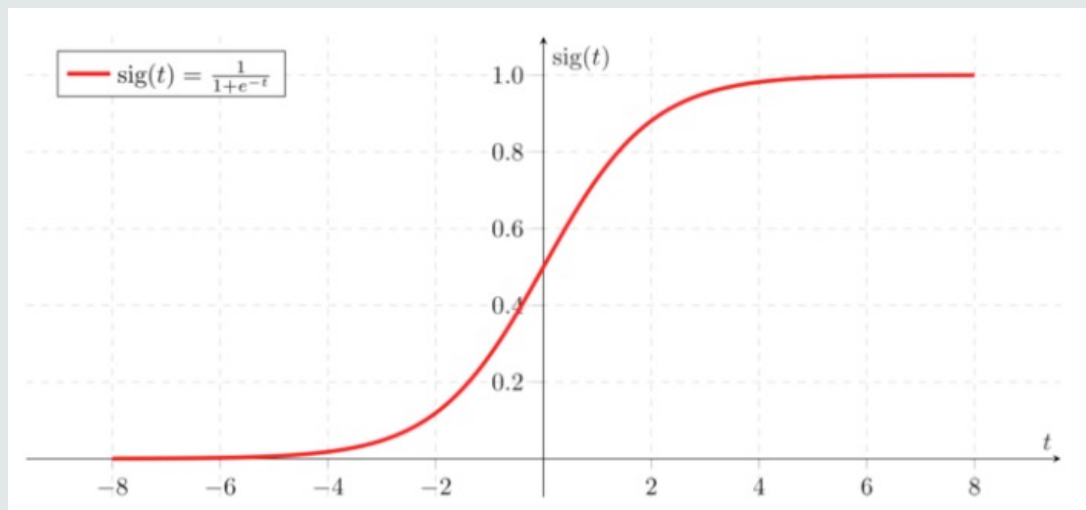
Där:

$$Z = w_0 + w_1x_1 + w_2x_2 + \dots + w_px_p$$

Algoritmen försöker hitta den bästa uppsättningen av vikter (w = weights) så att modellen kan göra de bästa prediktionerna.

(x = feature)

Sigmoid funktionen



- $\frac{1}{1+e^{-z}}$ kallas för sigmoid funktionen
- Kan ta alla tal och mappa de till ett värde mellan 0 och 1
- Om z blir oändligt stor närmar y sig 1 och när z blir oändligt liten närmar y sig 0
- e är exponentialfunktionen

$$Pr(target = 1|X) = \frac{1}{1 + e^{-z}}$$

Decision boundary - threshold

- För att avgör vilken klass datan tillhör kan en *threshold* (tröskel) sättas
- En *threshold* är ett värde där programmet ändras efter denna punkten
- Baserad på thresholden klassificeras den predikterade sannolikheten
- Säger threshold är 0.5 blir alla predikterade sannolikheter större än 0.5 klassificerat till 1, den positiva klassen, och alla sannolikheter mindre än 0.5 klassificerat till 0 som är den negativa klassen

$$\Pr(y = 1) \geq 0.5 \Rightarrow 1 \text{ (positiv)}$$

$$\Pr(y = 1) < 0.5 \Rightarrow 0 \text{ (negativ)}$$

LogisticRegression estimator in scikit-learn

- I scikit_learn använder vi LogisticRegression estimator objektet
- Det utför automatisk regularisering som kan kontrolleras med C parameter. Större värden på C skapar mer regularisering
- Metoden coef_ används för att på weights till input features från den tränade modellen



Tolkning av vikterna (weights)

Positiva vikter är associerade med en högre sannolikhet för att observera den positiv klassen

Negativa vikter är associerade med en lägre sannolikhet för att observera den positiva klassen

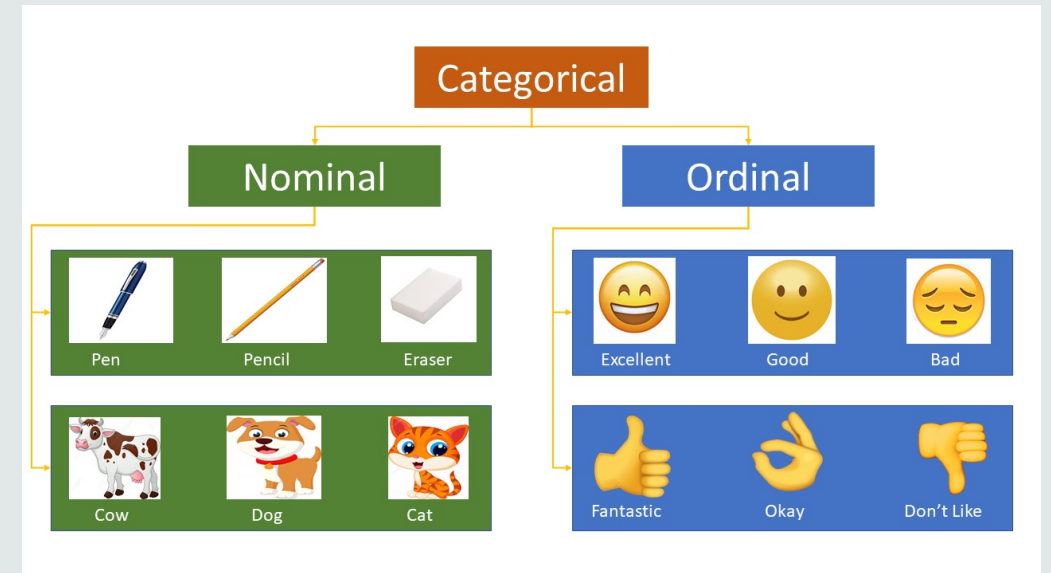
Värdet 0.5 används ofta för att prediktera sannolikheten för en Positiv klass, gränsen kan dock dock modifieras

Antaganden Logistic regression

- Har **inte** samma antagen som linear regression som linjäritet, normalitet och homogen varians
- **Binär y** - beroende variabel måste vara binär (anta två olika värden)
- **Oberoende observationer** - input features x måste vara oberoende av varandra, ingen korrelation
- **Större dataset** – borde vara minst 10 fall av minst förekommande klass för varje oberoende variabel

Kategoriska features

- De flesta machine learning algoritmerna kan inte hantera **kategoriska input features** och måste konverteras till numeriska värden (scikit-learn kräver numerisk data)
- *Exempel kategorisk feature är veckodagar: måndag, tisdag, ..., söndag*
- Finnes undantag så som **classification tree** som hanterar kategoriska features
- Två typer kategoriska features:
- **Nominal** - variabler som inte är relaterade till varandra i någon ordning som *färg* (röd, grön, blå)
- **Ordinal** - variabler med en viss ordning emellan dem som *betyg* (U, G, VG).



Kategoriska features

- Ett numerisk värde kan också delas in i en ordinal variabel, kallad **diskretisering**. Ex numerisk värde mellan 1 och 9 delas in i tre 1-3, 4-6, 7-9.
- Tre metoder för att konvertera kategorisk data:
 1. Ordinal Encoding
 2. One-Hot Encoding
 3. Dummy Variable Encoding



Ordinal Encoding

- När det finns en naturlig ordning mellan variablerna
- Var aktsam med att använda ordinal ifall ett sådant förhållanden inte finns!
Det kan ge fel prediktion. Exemplet bredvid är ett sådant.


Education	Encoding
Graduate School	1
University	2
High School	3
Others	4

One-Hot Encoding

- För nominala variabler där det inte är någon numerisk relation mellan variablerna.
- Variabeln ersätts med en unik binär variabel.
- En binär variabel tar värdet 1 eller 0 där värdet indikerar om något finns eller inte

Color	Integer Encoding	One-Hot Encoding
Red	0	[1, 0, 0]
Green	1	[0, 1, 0]
Blue	2	[0, 0, 1]

Dummy variabel



Costumer	Education
Id1	3
Id2	1
Id3	4
Id4	1
Id5	3
Id6	2
Id7	2

Costumer	grad_school	university	high_school
Id1	0	0	1
Id2	1	0	0
Id3	0	0	0
Id4	1	0	0
Id5	0	0	1
Id6	0	1	0
Id7	0	1	0

- Problem med One-Hot Encoding är att det skapar redundans
- Om vi till exempel vet att $[1, 0, 0]$ representerar "blå" och $[0, 1, 0]$ representerar "grön" behöver vi inte en annan binär variabel för att representera "röd", istället kan vi använda 0-värden för både "blå" och "grön" ensam, t.ex. $[0, 0]$.
- När du ändrar en kategorivariabel till dummy variabler kommer du att ha en färre dummyvariabel än du hade kategorier. Det beror på att den sista kategorin redan indikeras genom att ha ett 0 på alla andra dummyvariabler.
- Dummy variabel ändrar K-kategorier till K-1 binära variabler.
- Inkluderad sista kategorin läggs det bara till överflödig information, vilket resulterar i multicollinearity som kan före till overfitting

Predicting Credit Card Default

- Default (positiv klass) är att man inte betalar krediträkningen
- Varje rad är en kund
- Hands on genomgång Python



Vad har vi gjort idag?

- ♦ Klassificering
- ♦ Olika typer av klassificering metoder, vad sannolikhet är och hur det används i klassificering
- ♦ Modellen Logistic regression
- ♦ Kategoriska features: Ordinal encoding, One-hot encoding, dummy variabel



Nästa lektion

- ♦ Klassificering
- ♦ Evaluering av klassifikationsmodeller:
 - Error metrics
 - Confusion Matrix
 - Threshold value
- ♦ Classification trees
- ♦ Rule inference

