Klassificering i maskininlärning

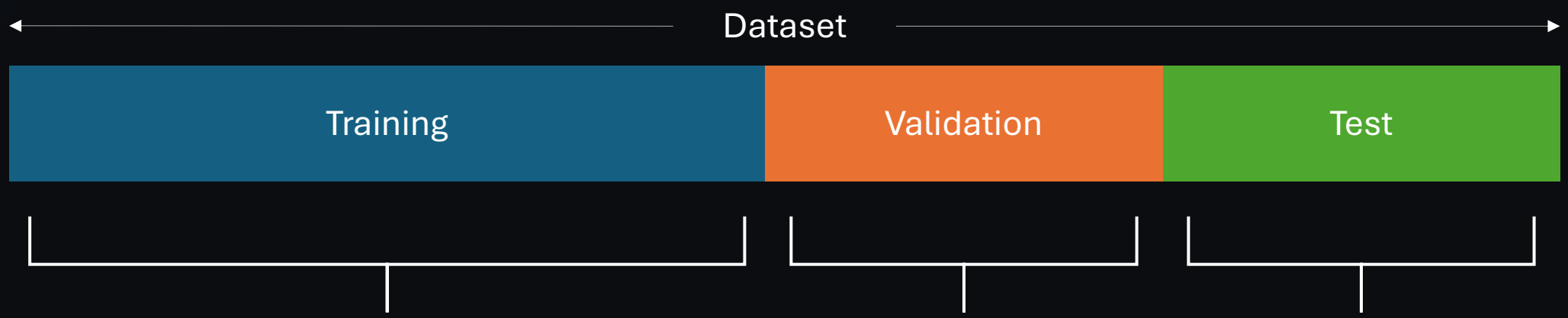
Klassificering vs Regression

- Regression: förutsäger ett tal (kontinuerligt)
  Ex: pris, temperatur, inkomst
- Klassificering: förutsäger en klass/kategori
  Ex: spam/inte spam, sjuk/frisk, 0-9 på en bild
- Ofta: modellen räknar en score/sannolikhet → vi gör ett beslut

Vad kommer in och vad kommer ut?

- Input: features X (t.ex. ålder, köp, klick…)
- Output: label y (klass)
- Modellen lär sig ett mönster: $X \rightarrow y$
- Viktigt: modellen gissar på nya exempel den aldrig sett

Train / Validation / Test



Skapa t.ex. 5 olika modeller

Utvärdera modellerna här och välj den bästa

Testa den valda modellen här och få ett estimate av “generalization error”. Innan du gör det, träna om den valda modellen på train + validation datan.

Viktigt: gör all preprocessing (t.ex. scaling) inuti pipeline efter split → annars risk för data leakage.

Score

- Många modeller ger score/probability
- Ex: p(klass=1) = 0.83
- För att få “0 eller 1” behövs ett beslut: threshold

Threshold

- Beslutregel:
  - om $p \geq 0.5$ → klass 1
  - annars → klass 0
- Men: threshold kan vara 0.2, 0.8, …
- Valet beror på: vilket fel är värst?

Confusion Matrix

- T**P**: Modellen sa positivt och det var positivt
- T**N**: Modellen sa negativ och det var negativt
- F**P**: Modellen sa positiv men det var egentligen negativt – alltså ett falskt alarm
- F**N**: Modellen sa negativ men det var egentligen positivt – alltså en miss

- True/False = hade modellen rätt?
- Positive/Negative = vad sa modellen?

		Predicted	
		Negative (**N**) -	Positive (**P**) +
Actual	Negative -	True Negatives (T**N**)	False Positives (F**P**)
	Positive +	False Negatives (F**N**)	True Positives (T**P**)

Vilket fel är värst?

- Ex 1: Sjukdomsscreening
  - FN kan vara farligt → vill fånga många sjuka
- Ex 2: Spamfilter
  - FP kan vara irriterande → vill inte stoppa viktiga mail
- Alltså: metric-val beror på mål

Accuracy (andel rätt)

- Accuracy = (TP + TN) / (alla)
- Bra när:
  - Klasserna är ungefär balanserade
  - FP och FN har liknande kostnad
- Risk när:
  - Obalans
  - Olika kostnader

Varför accuracy kan lura

- Att utvärdera klassificeringsmodeller är inte alltid lätt
- Dataset: 1% positiva, 99% negativa
- Modell som alltid gissar “negativ” får 99% accuracy
- Men den hittar 0 av de positiva

Obalans och stratifiering

- Obalans = en klass är mycket vanligare
- Se upp: accuracy kan bli missvisande
- Vi vill ofta behålla klassfördelningen i train/test (stratifiera)

Baseline

- Baseline = enkel referens
- Ex:
  - Gissa alltid majoritetsklass
  - Eller slumpa efter klassfördelning
- Om vår modell inte slår baseline: då har vi problem

Precision

- Precision = $\frac{TP}{TP+FP}$
- Fråga: ”När modellen säger positiv – hur ofta har den rätt?”
- Bra när falska larm (FP) är dyra
  Ex: spamfilter som råkar
  blockera viktiga mejl

		Predicted	
		Negative (**N**) -	Positive (**P**) +
Actual	Negative -	True Negatives (T**N**)	False Positives (F**P**)
	Positive +	False Negatives (F**N**)	True Positives (T**P**)

Recall

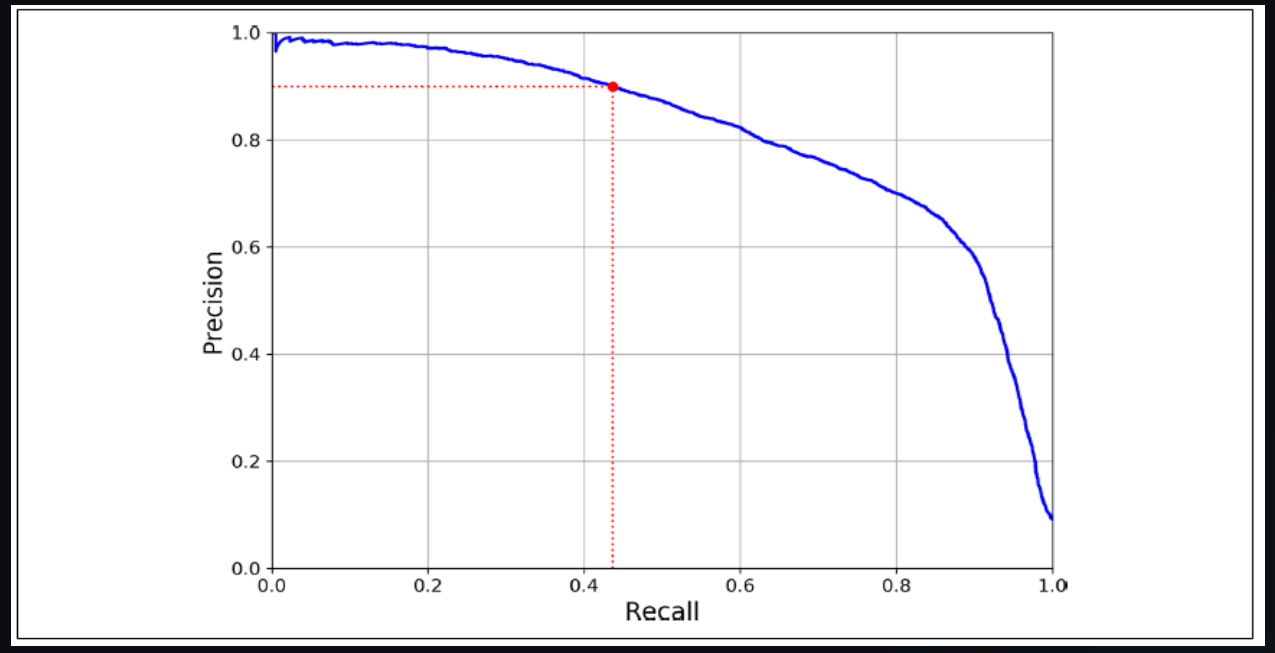
- Recall = $\frac{TP}{TP+FN}$
- Fråga: ”Av alla verkliga positiva – hur många hittar vi?”
- Bra när missar (FN) är dyra

  Ex: sjukdom, bedrägeri, säkerhet

		Predicted	
		Negative (**N**) -	Positive (**P**) +
Actual	Negative -	True Negatives (T**N**)	False Positives (F**P**)
	Positive +	False Negatives (F**N**)	True Positives (T**P**)

Precision / Recall trade-off

- Lägre threshold = fler positiva prediktioner
  - Recall ↑ (färre missar)
  - Precision ↓ (fler falska alarm)
- Högre threshold = färre positiva prediktioner
  - Precision ↑
  - Recall ↓
- Ingen magisk lösning: välj utifrån målet



Ibland kan båda förbättras samtidigt när modellen blir bättre – men principen att det finns en trade-off gäller nästan alltid i praktiken.

F1 Score (balansmått)

$$F_1\ Score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

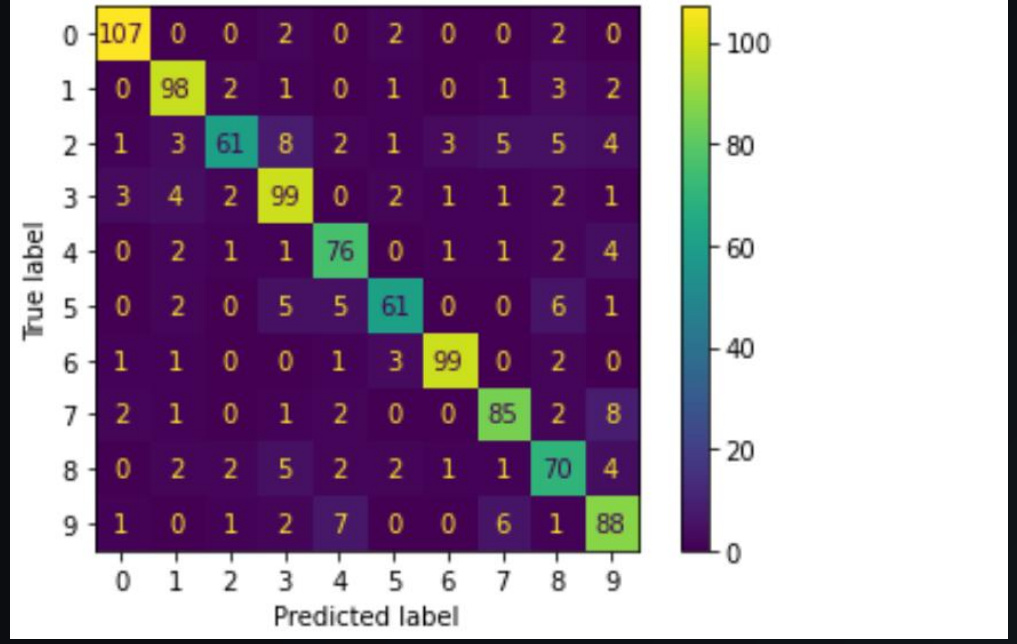
- F1 kombinerar precision och recall
- Hög F1 kräver att båda är rimligt höga
- Användbart när:
  - klasser är obalanserade och du vill ha balans
  - Man vill ha en sammanfattande siffra
- Men: man ska fortfarande titta på precision & recall var för sig

Precision	Recall	F_1 Score
1	1	1
1	0,01	0,019802
0,01	1	0,019802
0,9	0,4	0,553846
0,6	0,7	0,646154
0,9	0,1	0,18
0,6	0,4	0,48

Multi-class: flera klasser

- Ex: siffror 0 – 9
- Confusion matrix blir större (10x10)
- Metrics kan beräknas som:
  - macro (alla klasser lika viktiga)
  - micro (räkna totalt)
  - weighted (viktat efter klass-storlek)

```
1 display_confusion_matrix(y_test, y_pred_linear)
```



Receiver Operating Characteristic (ROC)

$$TPR\ (Recall) = \frac{TP}{P} = \frac{TP}{TP + FN}$$

Fråga: Vad besvarar TPR?

Svar: Andelen av den positiva klassen som vi predikterar korrekt.

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN}$$

Fråga: Vad besvarar FPR?

Svar: Andelen av den negativa klassen som vi felaktigt predikterar som positiv.

ROC-kurvan

- ROC: True Positive Rate vs False Positive Rate
- Varje punkt = en threshold
- AUC = area under curve
- 1.0 = perfekt
- 0.5 = slump

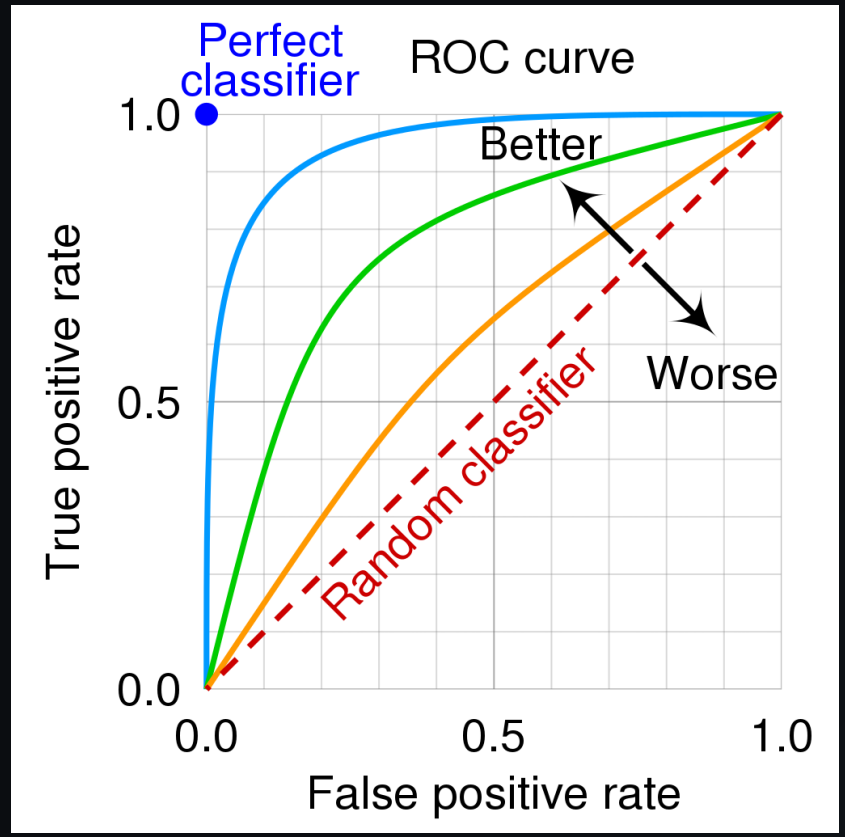


Bild tagen från: https://en.wikipedia.org/wiki/Receiver_operating_characteristic

När ROC/AUC är bra

- ROC/AUC bra för:
  - Jämföra modeller över många thresholds
  - Få en stabil helhetsbild
- Kan bli mindre intuitivt vid stark obalans
- Då kan PR-kurva ibland vara mer informativ

Vi kommer primärt använda confusion matrix + precision/recall/F1 i kursen. ROC/AUC är ett extra verktyg.

Klassificering i maskininlärning

Joakim Lindh