Evaluating Regression and Classification (Performance evaluation)

Stephen and Peace

Introduction

- We will explore how to measure how well a model works for predicting numbers (regression) and categories (classification).

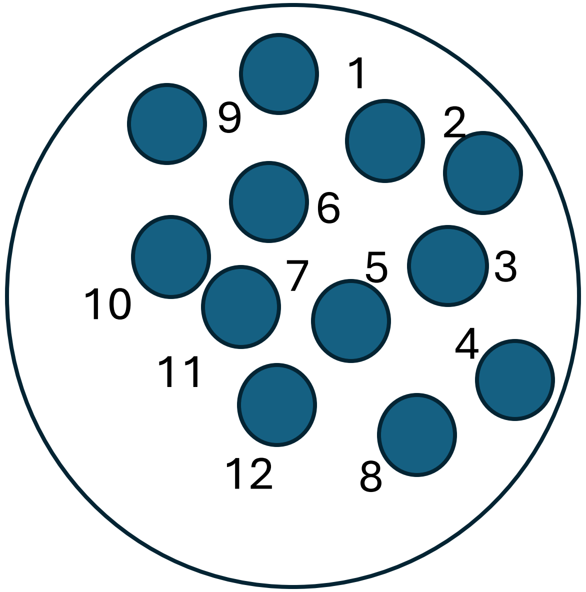
- **Example:** Predicting milk yield (regression) and whether a cow is "High" or "Low" yielding (classification).

Loading the Data

- We first load our dataset from a file called “Animal_data.csv”.

- Think of this like opening a spreadsheet that has animal details and results.

Train the model and predict the same data used for training the model



Train model and predict the same data

Obj	X1	X2	X3	Y
1	4	9	4	20
2	8	4	6	11
3	1	5	7	12
4	3	3	7	11
4	2	4	8	23
5	4	5	4	11
6	5	6	5	22
7	6	4	3	23
8	5	5	4	31
9	6	6	3	22
10	6	7	6	16
11	7	7	6	25
12	8	8	8	33

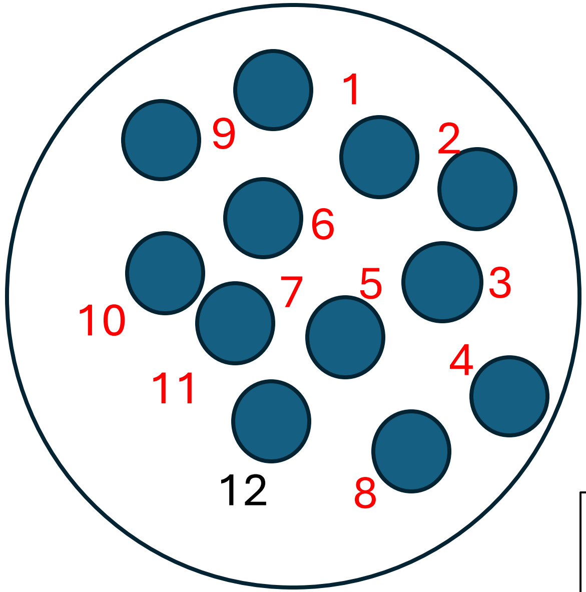
Predict

Y-pred
18
9
13
12
20
8
22
24
30
21
18
34

Leave one out and fold cross-validation

- We split data into K groups.
- Train on K-1 groups, test on the remaining group.
- Repeat K times so each group gets tested.
- Average results = fairer measure of model accuracy.

Training (11) and leave one out cross-validation (1)



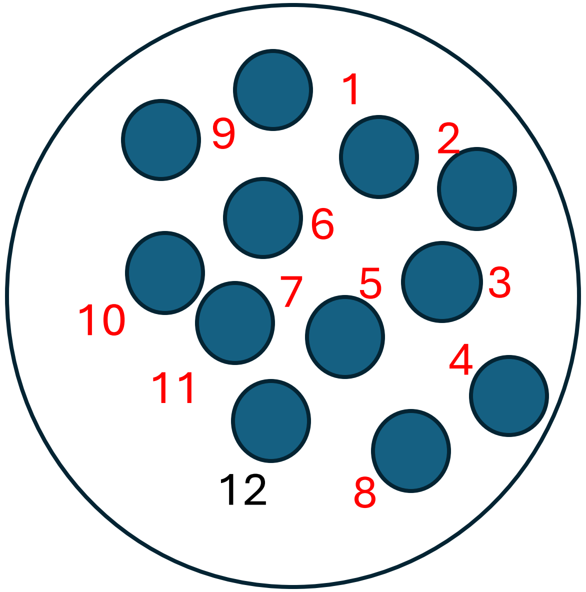
Training data

Test on number 12

Predicted

No	X1	X2	X3	Y	Y-pred
1	4	9	4	20	
2	8	4	6	11	
3	1	5	7	12	
4	3	3	7	11	
4	2	4	8	23	
5	4	5	4	11	
6	5	6	5	22	
7	6	4	3	23	
8	5	5	4	31	
9	6	6	3	22	
10	6	7	6	16	
11	7	7	6	25	
12	8	8	8	33	31

K: Fold cross validation, example K = 4



Training data

Test on 4

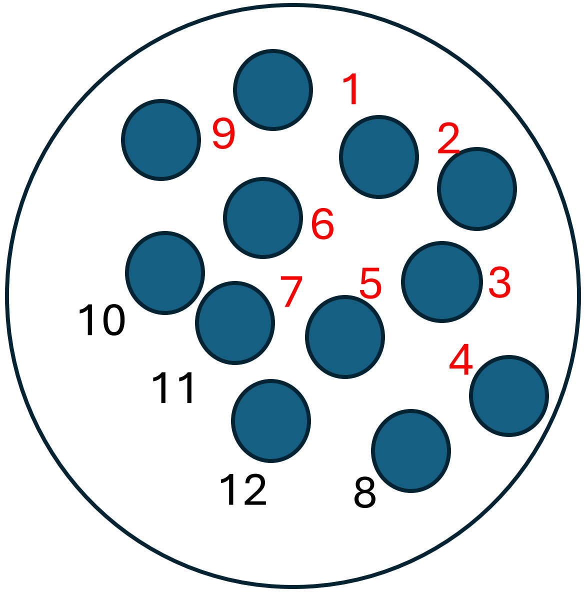
Predicted

No	X1	X2	X3	Y	Y-pred
1	4	9	4	20	
2	8	4	6	11	
3	1	5	7	12	
4	3	3	7	11	
4	2	4	8	23	
5	4	5	4	11	
6	5	6	5	22	
7	6	4	3	23	
8	5	5	4	31	
9	6	6	3	22	20
10	6	7	6	16	18
11	7	7	6	25	24
12	8	8	8	33	31

Train/Test Split (Regression)

- Split data into Training (learn) and Testing (check).
- Train model on 70% of the data, test it on 30%.
- Test results show how model works on unseen data.

Split data into Training (learn) and Testing (check)



Training data

Test data

Predicted

No	X1	X2	X3	Y	Y-pred
1	4	9	4	20	
2	8	4	6	11	
3	1	5	7	12	
4	3	3	7	11	
4	2	4	8	23	
5	4	5	4	11	
6	5	6	5	22	
7	6	4	3	23	
8	5	5	4	31	28
9	6	6	3	22	20
10	6	7	6	16	18
11	7	7	6	25	24
12	8	8	8	33	31

Regression metrics

- **Regression** = predicting numbers (like milk yield in l).
- We check:
  - **RMSE**: How far predictions are from the real value on average.
  - **MAE**: Average error without worrying about positive or negative.
  - **R^2**: How much of the variation in results our model explains.
- Smaller RMSE/MAE is better.
- R^2 closer to 1 is better.

RStudio

- Use the file:Performance_regression.R

Classification

Classification

- **Classification** = predicting categories ("Low" or "High").

- We use logistic regression to estimate the probability of "High" yield.

- Then pick a **threshold** (often 0.5) to decide

Train/Test Split (Classification)

- Same idea as regression but for "Low"/"High".
- Shows accuracy and other metrics for test data.
- Prevents overfitting (memorizing instead of learning).

Calibration Plot

- Checks if predicted probabilities match reality.
- Example: If we say 70% chance of "High", does it really happen 7 out of 10 times?
- Good calibration means predictions are trustworthy.

What is a Confusion Matrix?

- A 2×2 table comparing model predictions vs. actual labels for binary classification.

- Terminology
  - TP (True Positive): predicted High when actual High
  - FP (False Positive): predicted High when actual Low
  - FN (False Negative): predicted Low when actual High
  - TN (True Negative): predicted Low when actual Low

- All metrics below are computed from TP, FP, FN, TN.

Confusion Matrix Layout

	Predicted: High	Predicted: Low
Actual: High	TP	FN
Actual: Low	FP	TN

Accuracy

- **Definition:** Proportion of all predictions that are correct.
- **Formula:** Accuracy = (TP + TN) / (TP + FP + FN + TN)

- Use when
  - ➢Classes are roughly balanced and all errors have similar cost.

- Beware
  - ➢Can be misleading on imbalanced data (e.g., 95% one class).

Precision (Positive Predictive Value)

- **Definition:** Of all predicted High, how many are truly High?
- **Formula:** Precision = TP / (TP + FP)
- Use when
  - False positives are costly (e.g., flagging healthy animals as sick).

Recall (Sensitivity, True Positive Rate)

- **Definition:** Of all actual High, how many did we correctly predict?
- Formula: Recall = TP / (TP + FN)
- Use when
  - ➢Missing positives is costly (e.g., failing to detect truly high-yield animals).

Specificity (True Negative Rate)

- **Definition:** Of all actual Low, how many did we correctly predict?
- **Formula:** Specificity = TN / (TN + FP)
- **Related:** False Positive Rate (FPR) = 1 − Specificity.

Confusion Matrix Heatmap

- Visual version of confusion matrix.
- Darker colors = more cases.
- Easier to explain and spot patterns.

Worked Example (Numbers)

- Example confusion matrix: TP = 40, FP = 10, FN = 20, TN = 30 (Total=100)
- Accuracy = (TP+TN)/Total = (40+30)/100 = 0.700
- Precision = TP/(TP+FP) = 40/(40+10) = 0.800
- Recall (Sensitivity) = TP/(TP+FN) = 40/(40+20) = 0.667
- Specificity = TN/(TN+FP) = 30/(30+10) = 0.750

Calibration plot for test set

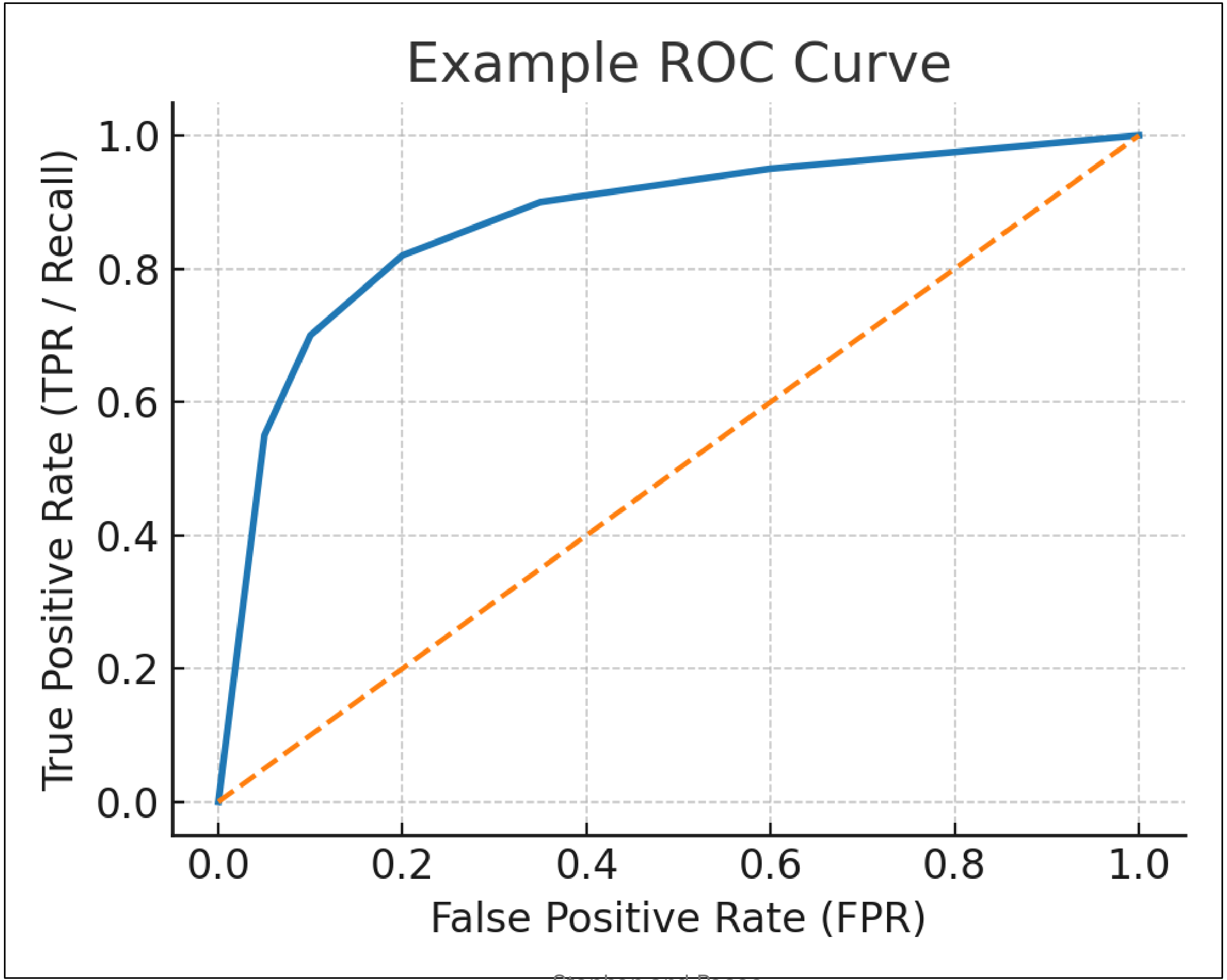
- Like in confusion matrix only using the test data.
- Visual version of confusion matrix.
- Darker colors = more cases.
- Easier to explain and spot patterns

Receiver Operating Characteristic (ROC)

- ROC (Receiver Operating Characteristic) plots model performance across all thresholds.

- x = FPR = FP/(FP+TN),

- y = TPR (Recall) = TP/(TP+FN).


Example ROC Curve
True Positive Rate (TPR / Recall)
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate (FPR)

ROC

- **How it is built:** vary the classification threshold on predicted scores/probabilities.
- **AUC (Area Under Curve):** 0.5 ≈ random, 1.0 = perfect; higher is better, threshold-independent.
- **Interpretation:** curves closer to the top-left are better; diagonal is random guessing.
- **Use cases:** compare classifiers and choose operating points; for rare positives, also check PR curves.

Receiver Operating Characteristics Curve (ROC)

- Shows trade-off between True Positives and False Positives.

- AUC (Area Under Curve) closer to 1 = better model.

- ROC helps choose the right threshold.

Precision-Recall Curve

- Focuses on "High" predictions.

- Precision = out of all predicted "High", how many are correct?

- Recall = out of all real "High", how many did we find?

Threshold Sweep

- Shows how Accuracy, Precision, Recall, Specificity, and F1 change as the threshold moves from 0 to 1.

- Helps pick the best threshold for the situation.