Lecture 6 - 12/09/24

Classification Problem:

- From X we predict one of K classes.
- I.e

  A person with certain symptoms could have:

  - Stroke ⟶ C_1
  - Drug overdose ⟶ C_2
  - Epileptic Seizure ⟶ C_3

  Which of the three does the individual have?

Bayes in Classification

- Look at X to predict C_k.
- As with logistic Regression, useful to consider probs:

  $P(C_k|X)$. → By knowing this, with new datapoint, we can predict C_k

- We can estimate $p(C_k|X)$ from training data using Baye's Theorem:

$$p(C_k|X) = \frac{P(X|C_k)\,p(C_k)}{P(X)}$$

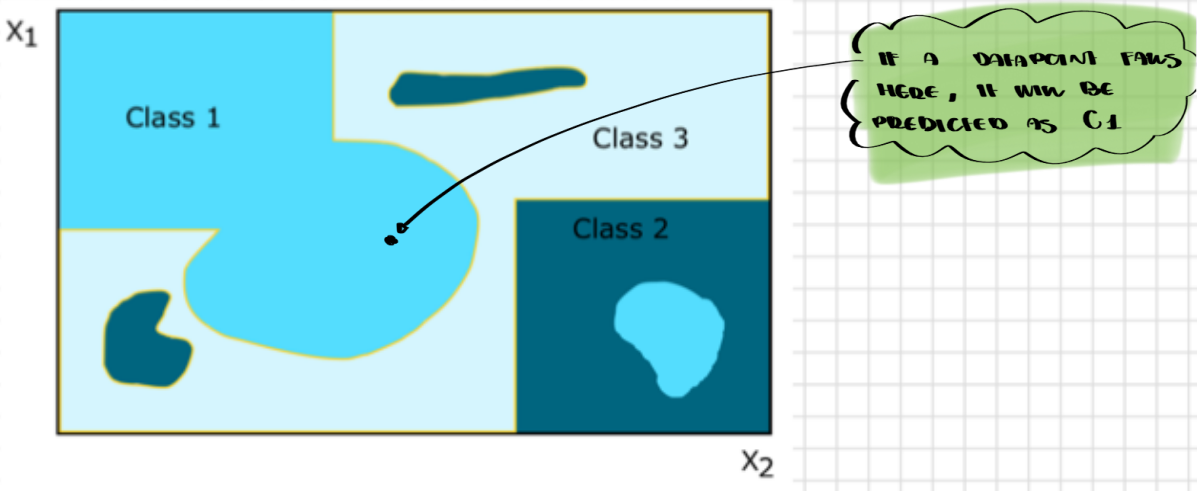
- Let's use $p(C_k|X)$ to classify.

  Aim: as few misclassification as possible.

  We need decision rules!!!

  - When do we classify a point as C_k?
  - Any decision Rule divides feature space into design regions.
  - Decision Regions are separated by decision boundaries!

  } D.R.



Aim to make as few misclassifications as possible. Minimize:

$$p(\text{mistake}) = p(X \in \mathcal{R}_1, \mathcal{C}_2) + p(X \in \mathcal{R}_2, \mathcal{C}_1)$$
$$= \int_{\mathcal{R}_1} p(X, \mathcal{C}_2) dX + \int_{\mathcal{R}_2} p(X, \mathcal{C}_1) dX$$

From the product rule, $p(X, \mathcal{C}_k) = p(X)p(\mathcal{C}_k|X)$, so we get,

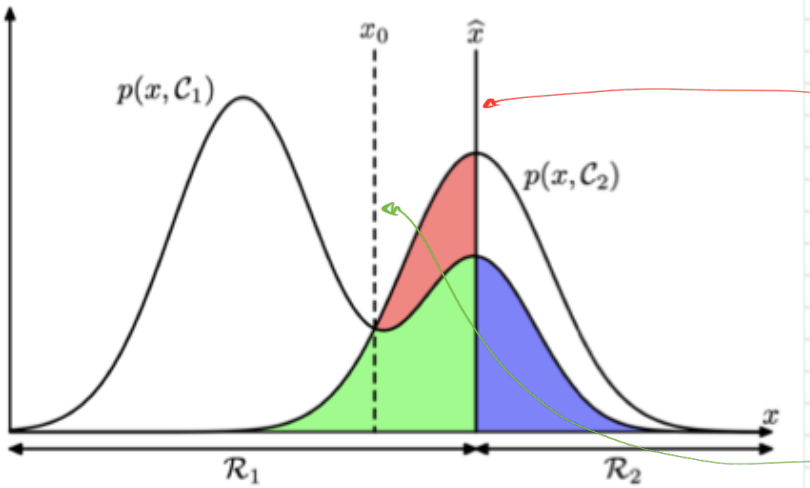
$$p(\text{mistake}) = \int_{\mathcal{R}_1} p(X)p(\mathcal{C}_2|X) dX + \int_{\mathcal{R}_2} p(X)p(\mathcal{C}_1|X) dX$$

P(X) CONTRIBUTES TO BOTH INTEGRALS SO MISTAKE IS LIMITED THE MOST BY THE ASSIGNMENT ON THE CLASS WITH THE HIGHEST $p(C_k|X)$ AT X.

: PROB OF X (THAT IS IN REGION 1) AND BE CLASSIFIED AS C_2

: PROB OF X (THAT IS IN REGION 2) AND BE CLASSIFIED AS C_1

- WE OBVIOUSLY WANNA MINIMIZE THE WRONG PREDICTIONS!!!



- LOOK AT THE AREAS UNDER THE FUNCTIONS!

TO THE LEFT CLASSIFY AS C_1, TO THE RIGHT C_2

: HERE WE CORRECTLY CLASSIFY C_2 BUT MISSCLASSIFY C_1

: CORRECTLY CLASSIFY C_1 BUT MISSCLASSIFY C_2

: MISSCLASIFY EVERYTHING

THAT'S WHY WE MOVE THE BOUNDARY HERE

MINIMIZE INTEGRALS!!!!

- WHAT WE'VE JUST DEVELOPED IS THE BAYES CLASSIFIER.
- WITH BAYES CLASSIFIER WE CLASSIFY TO K WITH HIGHEST POSTERIOR PROBS:

$$d(x) = \arg\max P(y = y | x)$$

- NOT ALL MISTAKES ARE EQUAL THOUGH
  - THERE ARE SOME THAT ARE WORSE THAN OTHER!
  - I.E. FOR CANCER, A FALSE POSITIVE CAUSES STRESS TO PATIENT
    A FALSE NEGATIVE MAY CAUSE THE DEATH OF A PATIENT!
  - WE REALLY NEED TO LIMIT FALSE NEGATIVES!

LOSS & LOSS MATRIX!

- WE WANT TO PUNISH THE ALGORITHM W CLASSIFICATION IT MAKES
- WE ASSIGN A LOSS $L_{kj} \geq 0$ DEPENDS ON K, TRUE CLASS, AND J, ASSIGNED CLASS.
  - L_{kj} IS AN ENTRY IN A LOSS MATRIX
- THE BIGGER THE LOSS FOR A GIVEN CLASSIFICATION, THE BIGGER THE PUNISHMENT!

	cancer	normal
cancer	0	1000
normal	1	0

NO PUNISHMENT

BIG PUNISHMENT!

- THE CONCEPT OF LOSS LET US EXPRESS OUR GOAL:
  - WE WANNA MINIMIZE THE EXPECTED LOSS:

$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(X, C_k) dX$$

Again, we can rewrite this to get

$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(X) p(C_k|X) dX.$$

So we minimize loss by assigning a new X to the class j that minimizes

$$\sum_k L_{kj} p(C_k|X).$$

- MINIMIZING POSTERIOR EXPECTED LOSS IS ENOUGH
  - IF $d(X)$ MINIMISES POSTERIOR EXPECTED LOSS $\forall x$ THEN $d(X)$ ALSO MINIMISES THE EXPECTED LOSS
  - WE OFTEN USE 0-1 LOSS:

$$d(j,k) = \begin{cases} 1, & j \neq k \\ 0, & j = k \end{cases}$$

In regression, we can also use loss, e.g. *Squared error loss*

$$L(y, d(x)) = (y - d(x))^2.$$

Absolute error loss

$$L(y, d(x)) = |y - d(x)|.$$

Remember: the loss function is usually non-negative.

- THE EXPECTED LOSS IS A THEORETICAL QUANTITY:

ALSO NAMED → TEST ERROR, GENERALISATION ERR, RISK, PREDICTOR ERROR.

  - WE CAN ESTIMATE EXPECTED LOSS BY EMPIRICAL RISK:

$$\frac{1}{n} \sum_{i=1}^{n} L(Y_i, d(X_i))$$

WHERE n IS THE # OF OBSERVATIONS

TEST & TRAINING ERROR

- TRAINING ERROR IS THE EMPIRICAL RISK COMPUTED FROM THE TRAINING SET
  - GENERALLY A BAD ESTIMATOR
- THE TEST ERROR DENOTES BOTH:
  - THE TRUE EXPECTED LOSS
  - THE ESTIMATE = EMPIRICAL RISK FROM TEST DATA
- TEST ERROR COMPUTED BY CROSS-VALIDATION IS ALSO ESTIMATE OF EXPECTED LOSS.

BAYES CLASSIFIER

IN THE LANGUAGE OF LOSS:

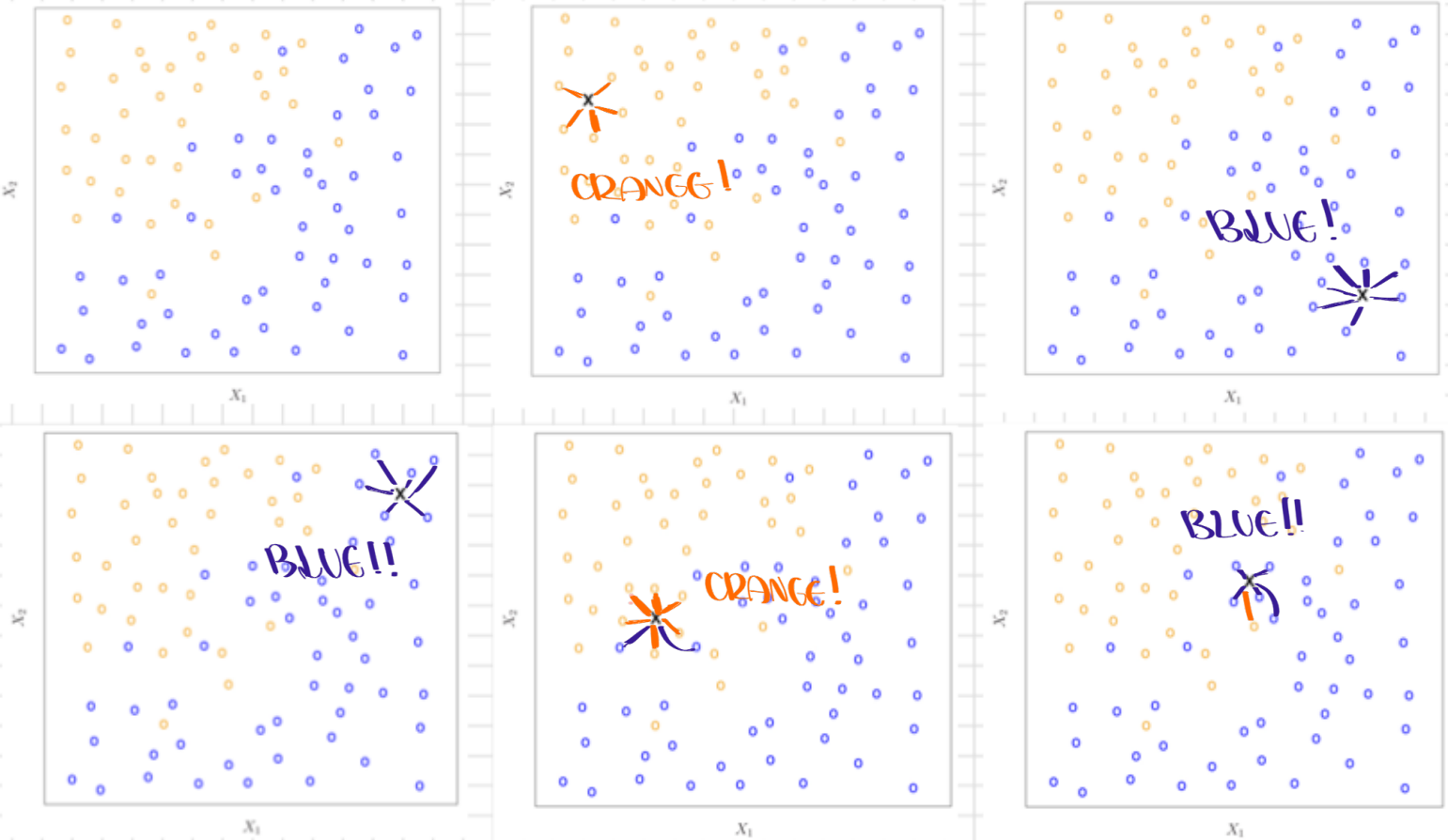
- THE BAYES CLASSIFIER MINIMISES THE EXPECTED LOSS UNDER THE SPECIFIC CHOICE 0-1 LOSS.

- The associated error, Bayes error rate, is a theoretical lower bound.
- Remember: with a Bayes classifier, we classify to the class k with the highest posterior probability.

$$d(x) = \arg\max_k P(y = y \mid x)$$

K-nearest neighbour

- What if we see x's that we've never observed? How do we classify?



- By looking at the picture, our first move is to look at it's neighbours & then say that x will be part of that class
- KNN (k-nearest neighbour) approximates the posterior class probs

Classify point x_0

1. Find k points in the training data closest to x_0 → set $\mathcal{N}_0$
2. Estimate posterior prob for class j as fraction of $\mathcal{N}_0$ from j:

$$P(Y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in \mathcal{N}_0} I(y_i = j).$$

3. Choose class with highest posterior probs.

- Choosing a right k matters, but obviously there is not a method to decide this!

↓

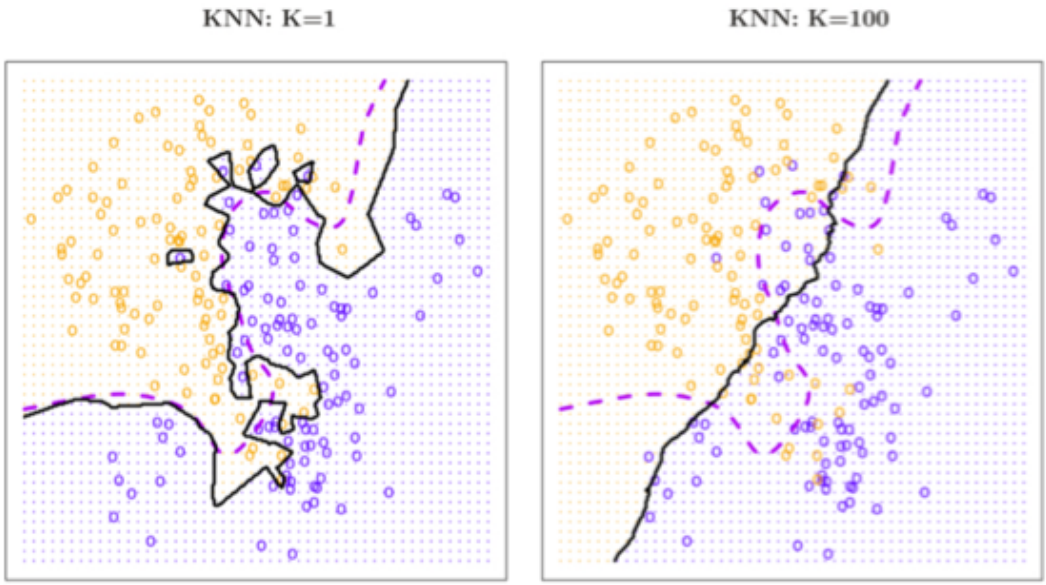


FIGURE 2.16. *A comparison of the KNN decision boundaries (solid black curves) obtained using $K = 1$ and $K = 100$ on the data from Figure 2.13. With $K = 1$, the decision boundary is overly flexible, while with $K = 100$ it is not sufficiently flexible. The Bayes decision boundary is shown as a purple dashed line.*

Summery for KNN:

- Resulting decision Rule is simply:
  - KNN assigns a class according to a majority vote among K closest points.
- Gives extremely flexible boundaries
  - Recall image above.
- Good classifier with error rate close to Bayes err rate.
- Often does not work when in high dimension feature space.
  - many points, maybe big spread!
- K can be chosen by cross-validation