CS 307

Fall 2023

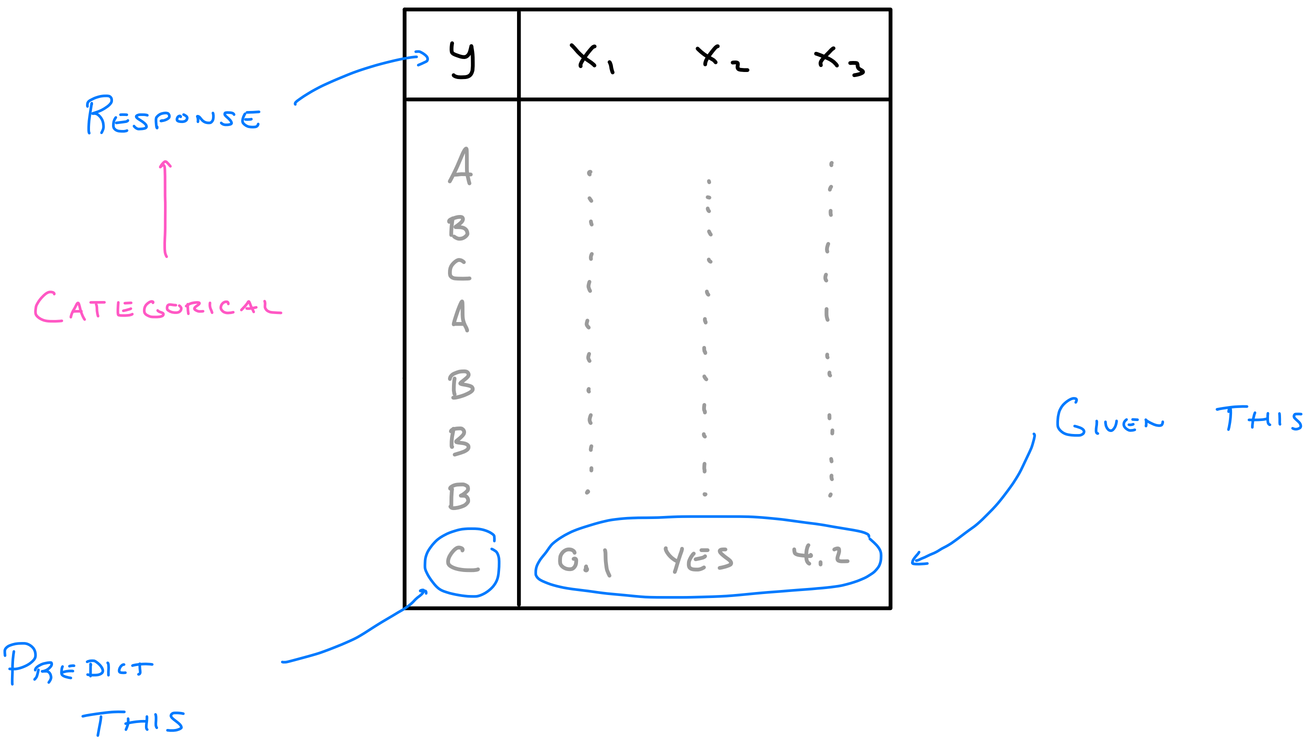
Dalpiaz

Week 08

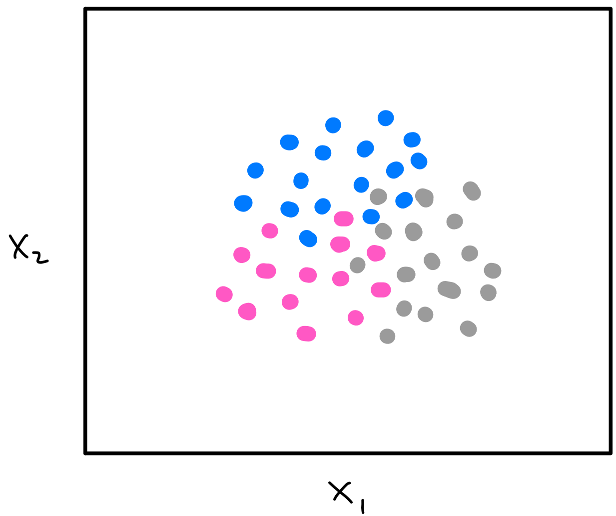
Classification

↳ An introduction

Data View


Response
Categorical
y
x_1 x_2 x_3
A
B
C
A
B
B
B
C
0.1 YES 4.2
Given This
Predict This

Visual Data View



PROBABILITY VIEW

P FEATURES K CATEGORIES

$$(X, Y) \in \mathbb{R}^P \times \{1, 2, \ldots K\}$$

FEATURES RESPONSE

FIND A CLASSIFIER $C(x)$ THAT MINIMIZES

$$P[C(x) \neq Y]$$

PROBABILITY OF MISCLASSIFICATION

WHERE $C : \mathbb{R}^P \longrightarrow \{1, 2, 3, \ldots K\}$

INPUT FEATURES OUTPUT CATEGORY

BAYES CLASSIFIER

← MINIMIZES PROBABILITY OF MISCLASSIFICATION

$$C^B(x) \triangleq \underset{k \in \{1, \ldots, K\}}{\operatorname{ARGMAX}} P[Y = k \mid X = x]$$

GIVEN FEATURE VECTOR x, CLASSIFY OBSERVATION AS THE CATEGORY WITH THE HIGHEST PROBABILITY

DUH?

Example

$$C^B(x=0) = \,?$$

$$\frac{P[x=0 \cap Y=A]}{P[x=0]}$$

JOINT DISTRIBUTION OF (X,Y)

Y \ X	0	1	
A	0.1	0.1	0.2
B	0.2	0.1	0.3
C	0.1	0.4	0.5
	0.4	0.6	

MARGINAL DISTRIBUTION OF X

$$P[Y \mid X=0] = \begin{cases} 0.25 & y=A \\ 0.50 & y=B \\ 0.25 & y=C \end{cases}$$

CONDITIONAL PROBABILITY OF $Y \mid X=0$

$$C^B(x=0) = B$$

$$C^B(x=1) = C$$

Bayes Error

← Average misclassification using Bayes classifier

$$1 - E_X\left[\max_k P[Y=k \mid X=x]\right]$$

"Irreducible error"

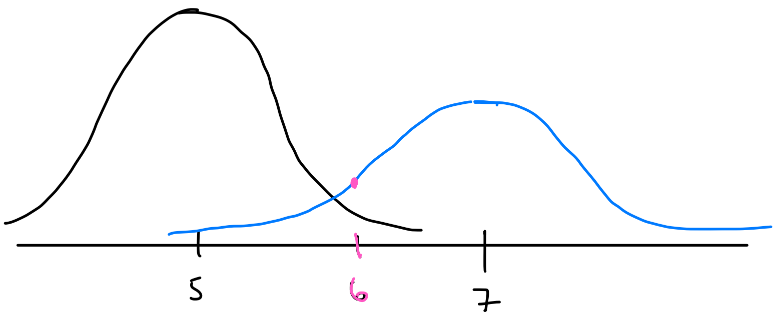
		X = 0	X = 1	
Y	A	0.1	0.1	0.2
	B	0.2	0.1	0.3
	C	0.1	0.4	0.5
		0.4	0.6	

$$= 1 - \left[P[Y=B \mid X=0]\,P[X=0] + P[Y=C \mid X=1]\,P[X=1]\right]$$

$$= 1 - \left[\left(\frac{0.2}{0.4}\right)(0.4) + \left(\frac{0.4}{0.6}\right)(0.6)\right]$$

$$= 1 - [0.2 + 0.4] = \underline{0.4}$$

Example



$$X \mid Y=0 \sim N(\mu=5, \sigma=1) \qquad f_0(x)$$

$$X \mid Y=1 \sim N(\mu=7, \sigma=2) \qquad f_1(x)$$

$$\pi_0 = P[Y=0] = 0.6$$

$$\pi_1 = P[Y=1] = 0.4$$

$$C^B(x=6) = ?$$

CALCULATE

$$P[Y=0 \mid X=6] = \frac{\pi_0 f_0(6)}{\pi_0 f_0(6) + \pi_1 f_1(6)} = \ldots$$

TO SCIPY!

$$P[Y=0 \mid X=6] = \frac{\pi_0 f_0(6)}{\pi_0 f_0(6) + \pi_1 f_1(6)}$$

$$P[Y=1 \mid X=6] = \frac{\pi_1 f_1(6)}{\pi_0 f_0(6) + \pi_1 f_1(6)}$$

SAME DENOMINATOR

ONLY NEED NUMERATOR FOR CLASSIFICATION

In Practice

Don't know $P[Y = k \mid X = x]$!!!

Estimate it !!!

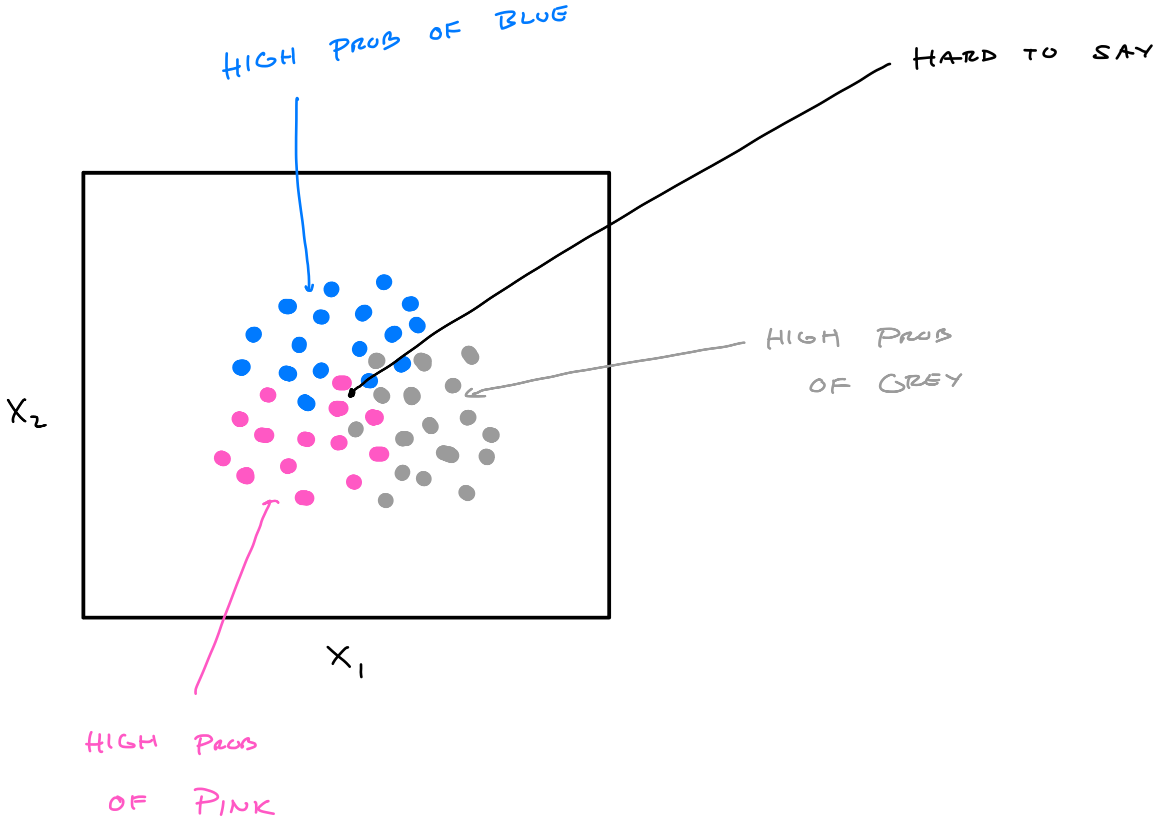
Learned classifier

$$\hat{C}(x) = \underset{k}{\operatorname{argmax}} \underbrace{\hat{P}[Y = k \mid X = x]}_{\text{Estimate of conditional probability}}$$

A "guess" for

$$C^B(x)$$

How?


HIGH PROB OF BLUE
HARD TO SAY!
HIGH PROB OF GREY
X_2
X_1
HIGH PROB OF PINK

ESTIMATING CONDITIONAL PROBABILITIES

KNN w/ sklearn.neighbors.KNeighborsClassifier

Trees w/ sklearn.tree.DecisionTreeClassifier

LINEAR MODELS w/ sklearn.linear_model.LogisticRegression

FAMILIAR INTERFACES

```
clf.fit
clf.predict
clf.predict_proba
```

MAKE CLASSIFICATIONS

ESTIMATE CONDITIONAL PROBABILITIES

METRICS

WOULD LIKE $P\left[C(x) \neq Y\right]$

SETTLE FOR $\frac{1}{n}\sum_{i=1}^{n} I\left(C(x_i) \neq y_i\right)$

MISCLASSIFICATION

$$I\left(C(x_i) \neq y_i\right) = \begin{cases} 1 & C(x_i) \neq y_i \\ 0 & C(x_i) = y_i \end{cases}$$

$$\frac{1}{n}\sum_{i=1}^{n} I\left(C(x_i) = y_i\right)$$ ← ACCURACY

$C(x)$ PLACEHOLDER

$\hat{C}(x)$ LEARNED

$C^B(x)$ BAYES

BINARY CLASSIFICATION

$Y = 0$ or $Y = 1$

"NEGATIVE" "POSITIVE"

$$p(x) \triangleq P[Y = 1 \mid X = x]$$

$$1 - p(x) = P[Y = 0 \mid X = x]$$

$$C^B(x) = \begin{cases} 1 & p(x) \geq 0.5 \\ 0 & \text{else} \end{cases}$$

METRICS

FP/TP

FN/TN

ETC

Nonparametric Classification

k-Nearest Neighbors and Decision Trees

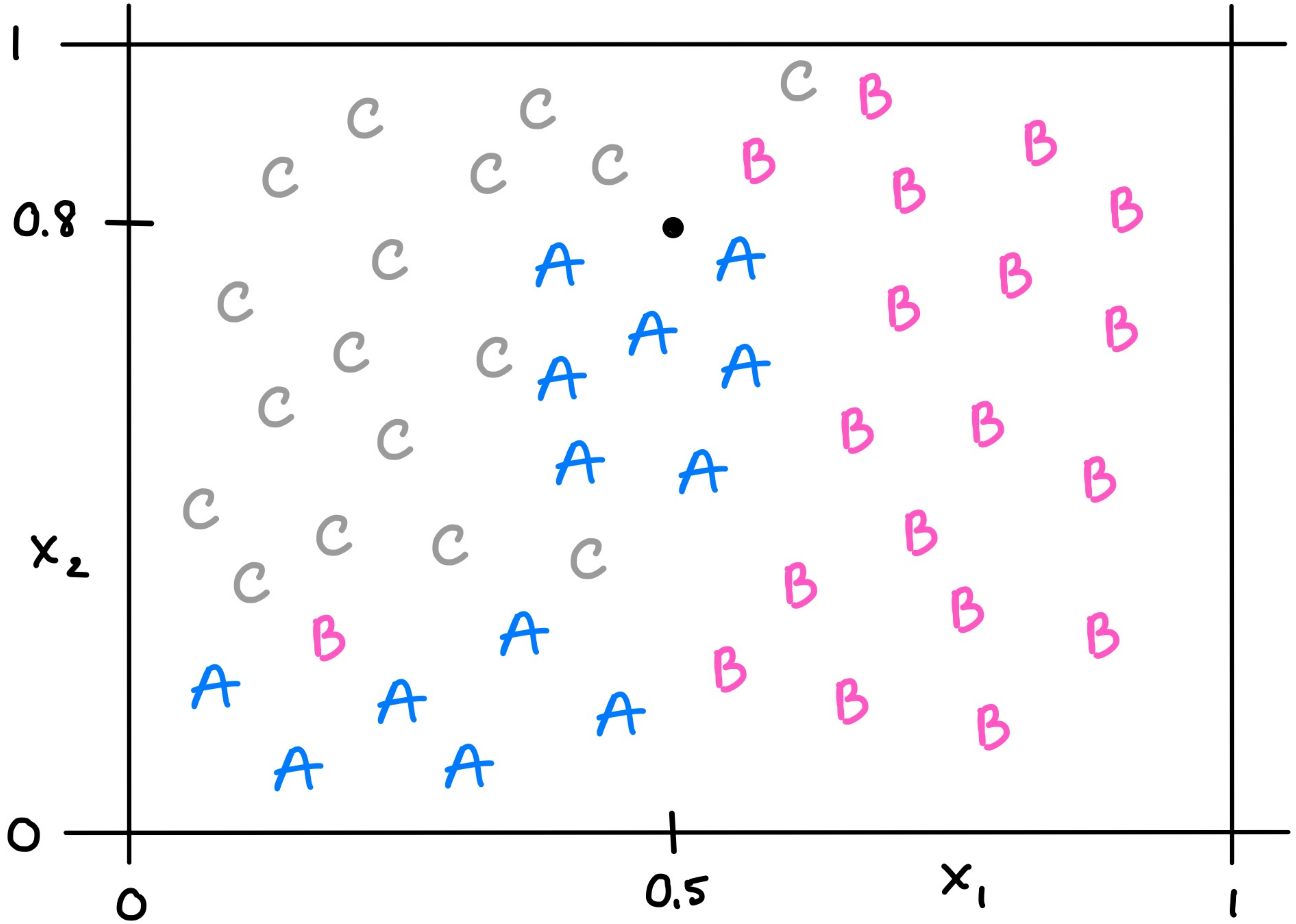
David Dalpiaz /// stat432.org

Classification Setup

Tabular View

y	x_1	x_2
A	⋮	⋮
⋮	⋮	⋮
A	⋮	⋮
B	⋮	⋮
⋮	⋮	⋮
B	⋮	⋮
C	⋮	⋮
⋮	⋮	⋮
C	⋮	⋮
?	0.5	0.8

Graphical View



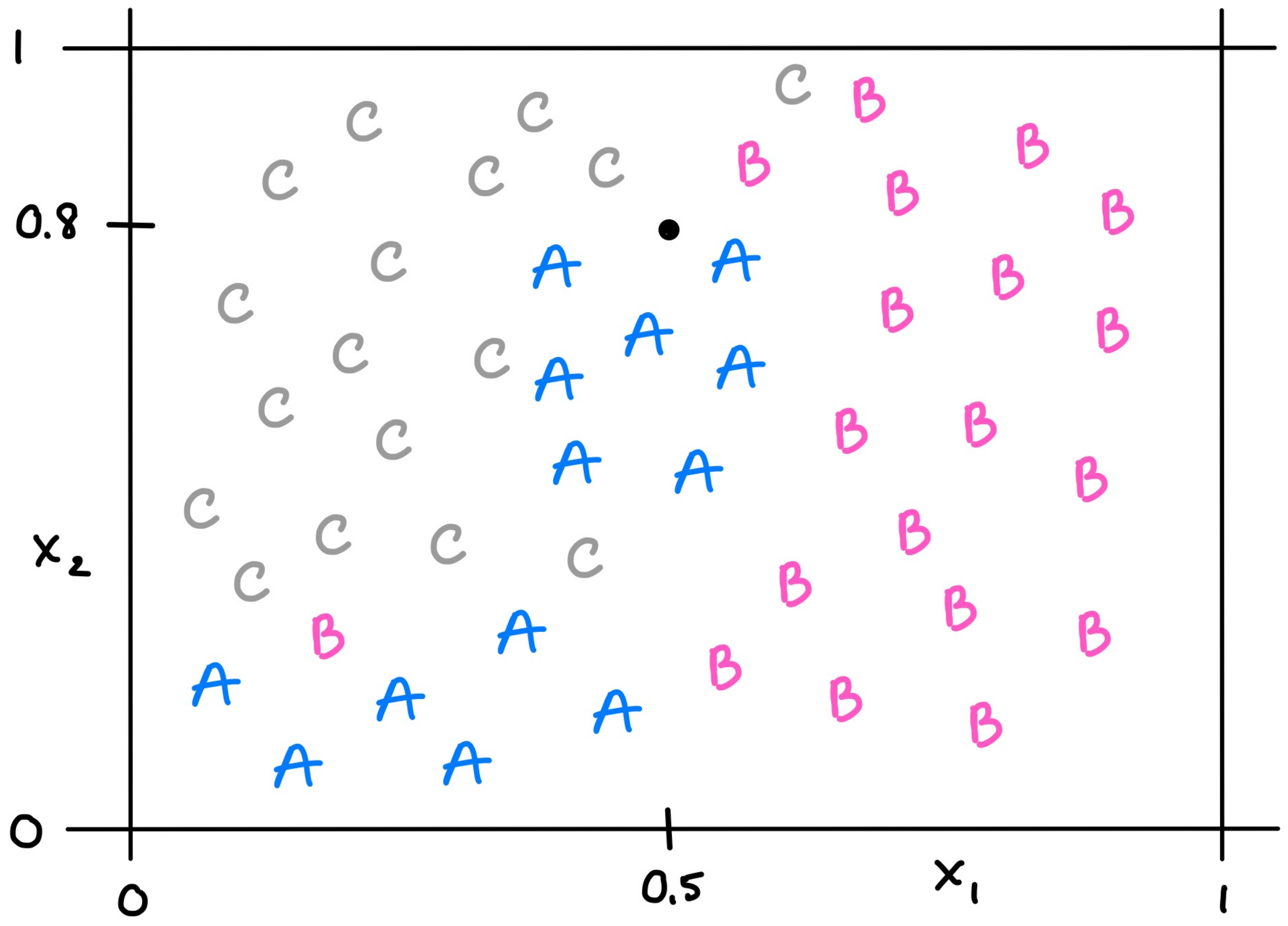
Goal

We want to estimate…

$$p_g(x) = P\left[Y = g \mid X = x\right]$$

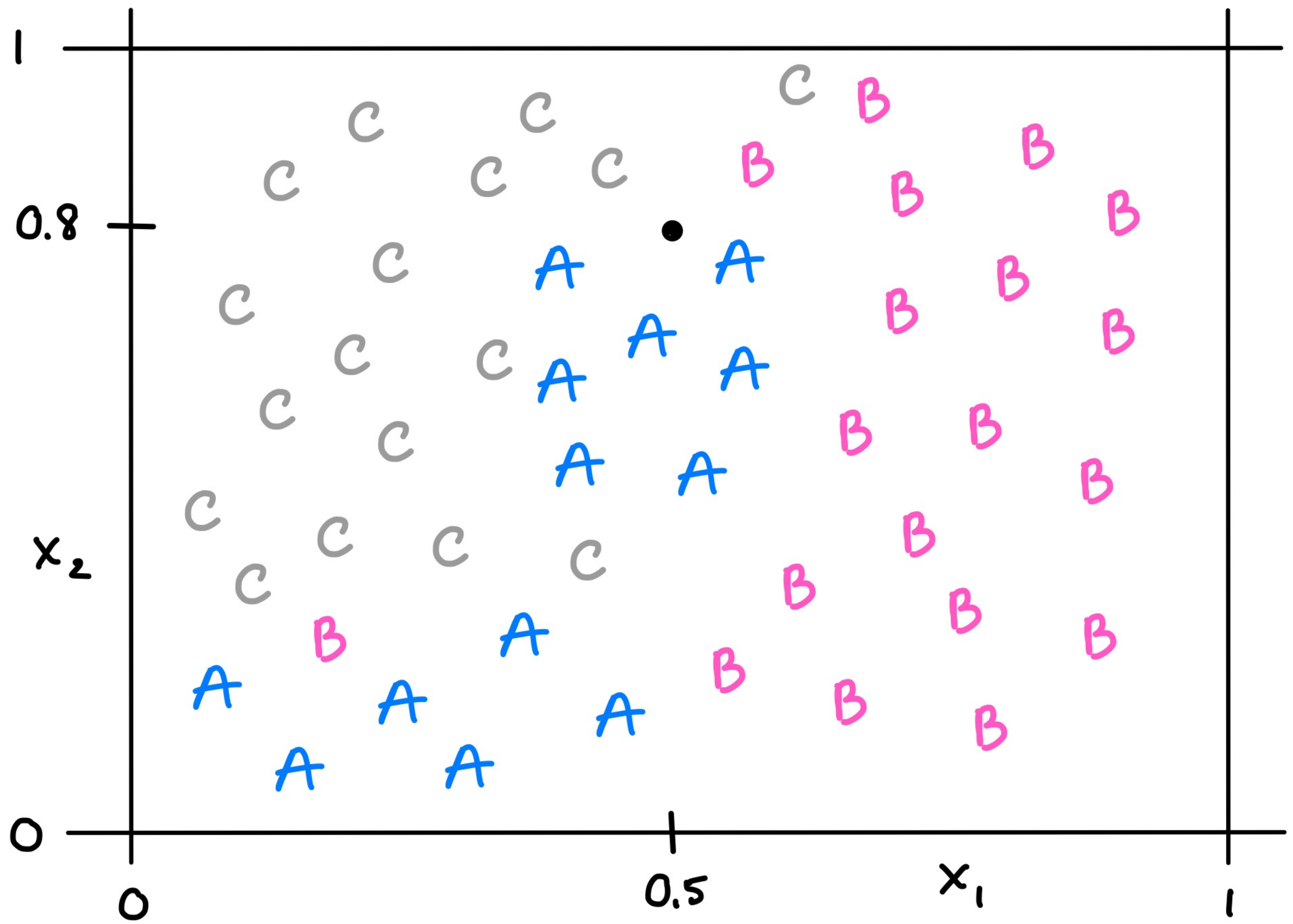
k-Nearest Neighbors (k-NN)

$$\hat{p}_g(x) = \hat{P}\left[Y = g \mid X = x\right] = \frac{1}{k} \sum_{\{i\,:\,x_i \in \mathcal{N}_k(x, \mathscr{D})\}} I\left(y_i = g\right)$$



k-Nearest Neighbors (k-NN)

Let $k = 5$ and $x = (0.5,\ 0.8)$.



$$\hat{P}\left[Y = A \mid X = x\right] = 3/5$$

$$\hat{P}\left[Y = B \mid X = x\right] = 1/5$$

$$\hat{P}\left[Y = C \mid X = x\right] = 1/5$$

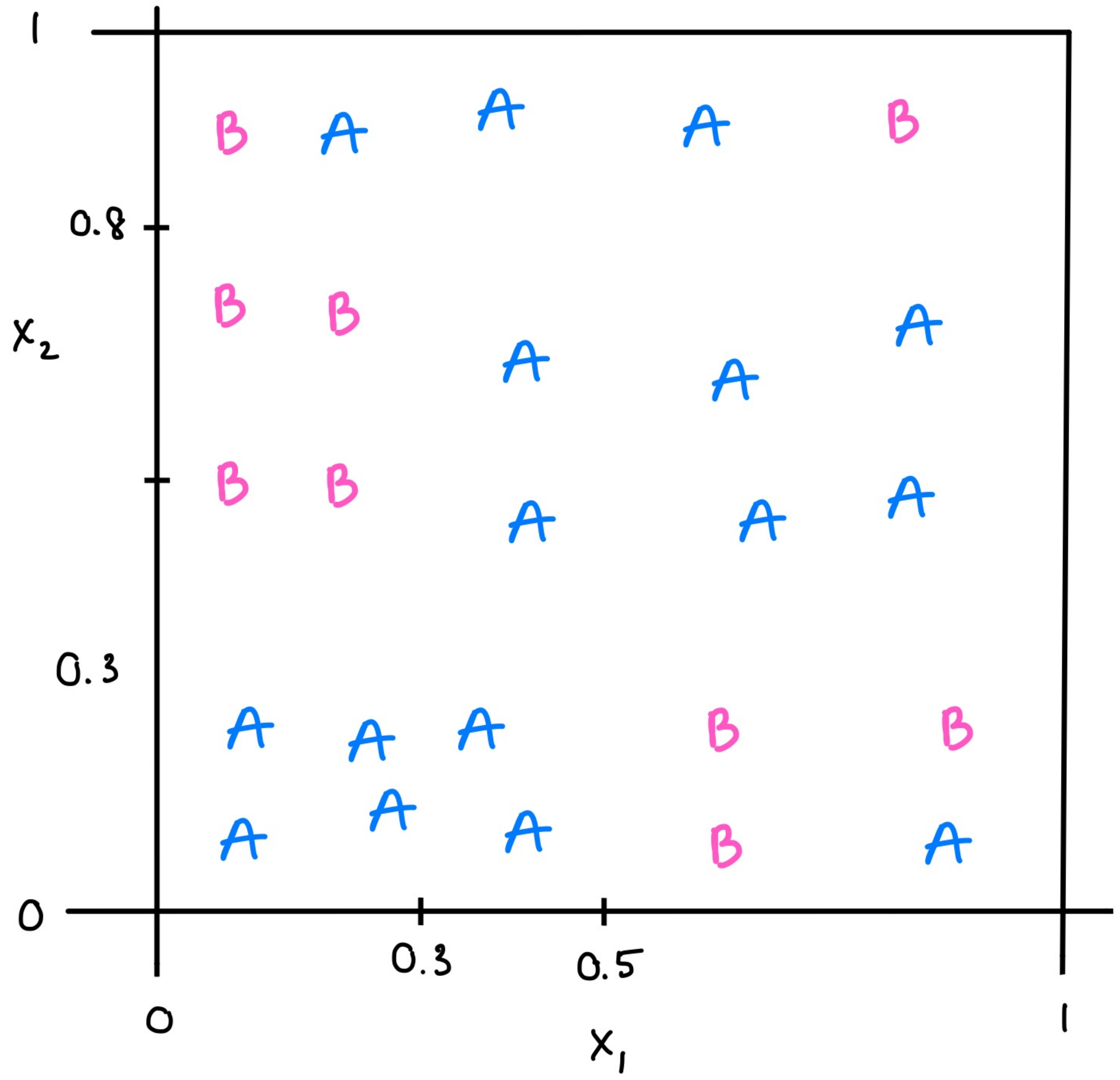
k-Nearest Neighbors (k-NN)

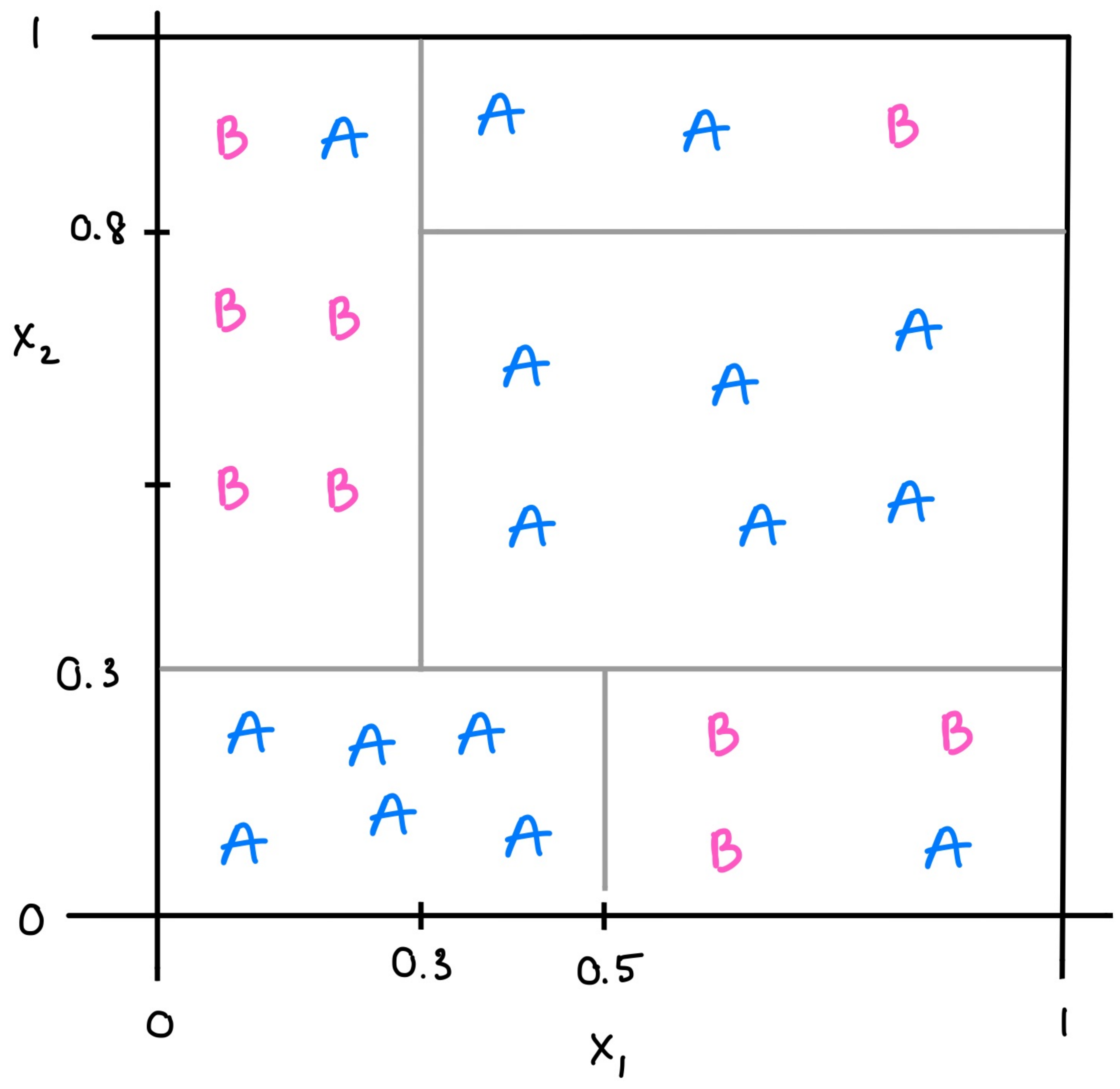
Future Practical Considerations

- Beware the **curse of dimensionality**!
- If there are two categories, consider an odd value of k to avoid **ties**.
    - Check documentation to see how specific implementations break unavoidable ties. Sometimes this is done at *random*!
- Can use any **distance** metric to determine nearest neighbors, but often Euclidean.
- **Scaling** of feature variables can have a big impact.
- k will need to be **tuned**.
- Recall: k-NN is *fast* at training time (memorize data), ***slow*** at prediction time.
- Recommended **R** package and function: **`caret::knn3`**

Decision Trees

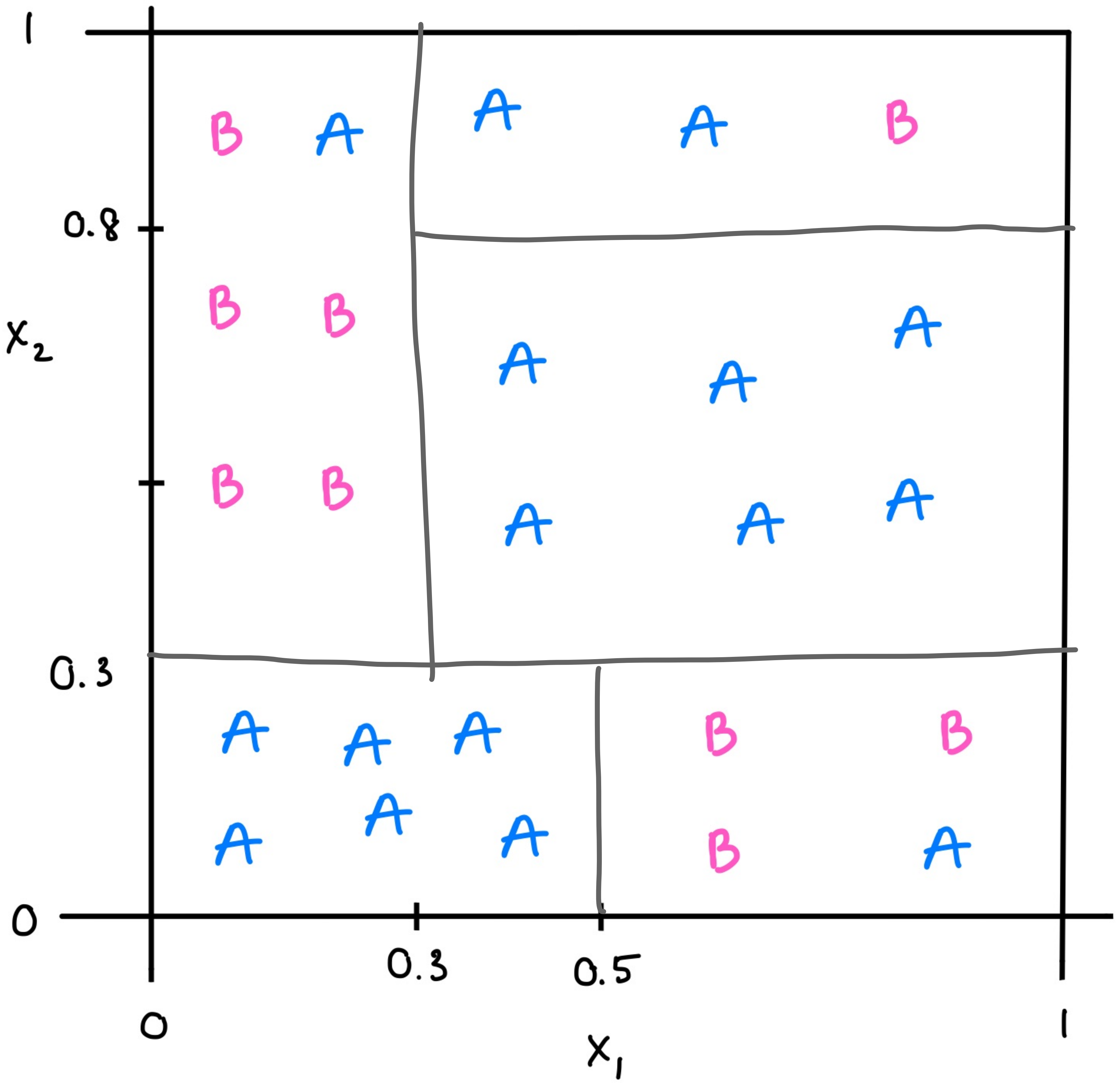
Neighborhoods via Recursive Partitioning

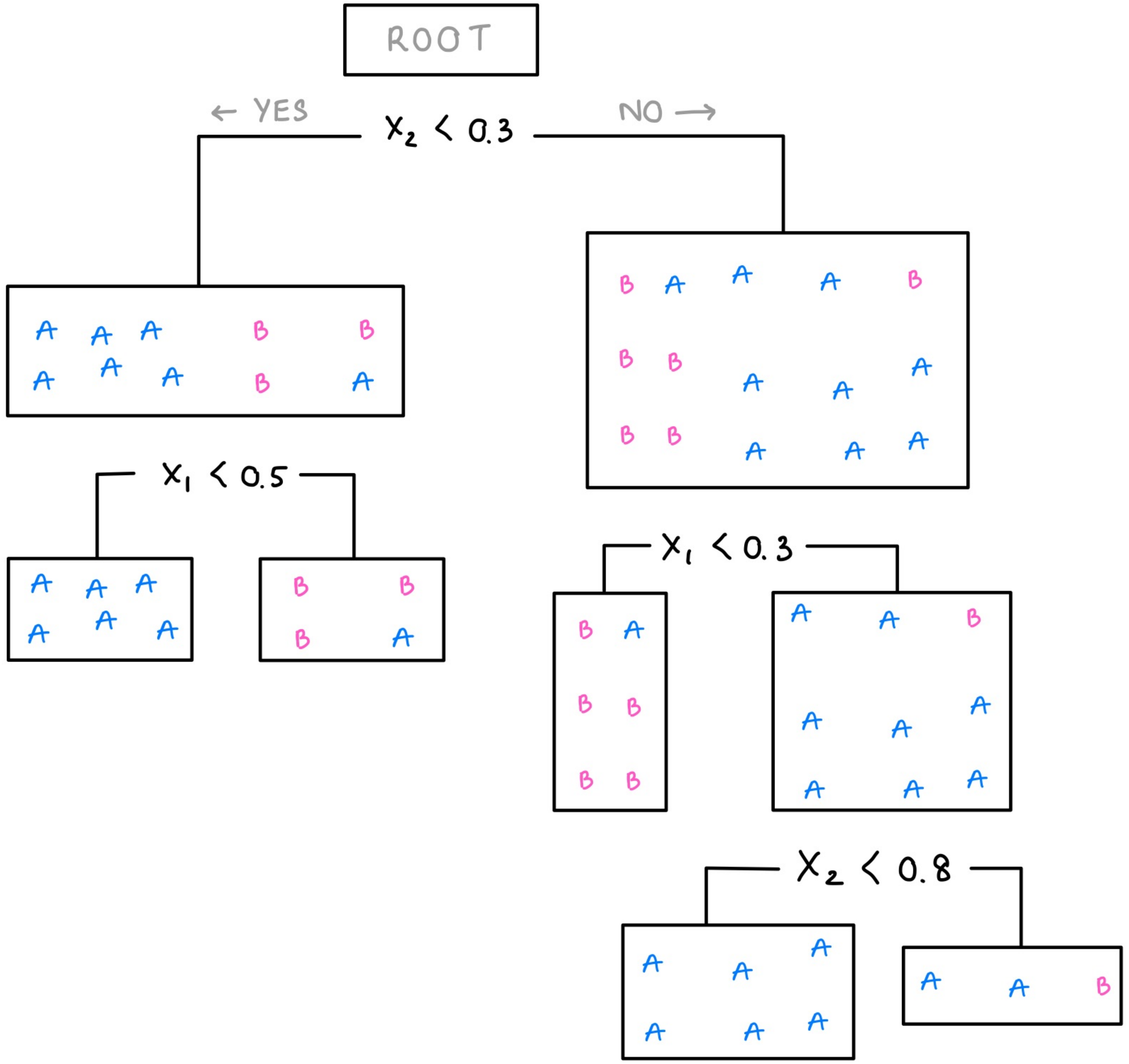




Decision Trees

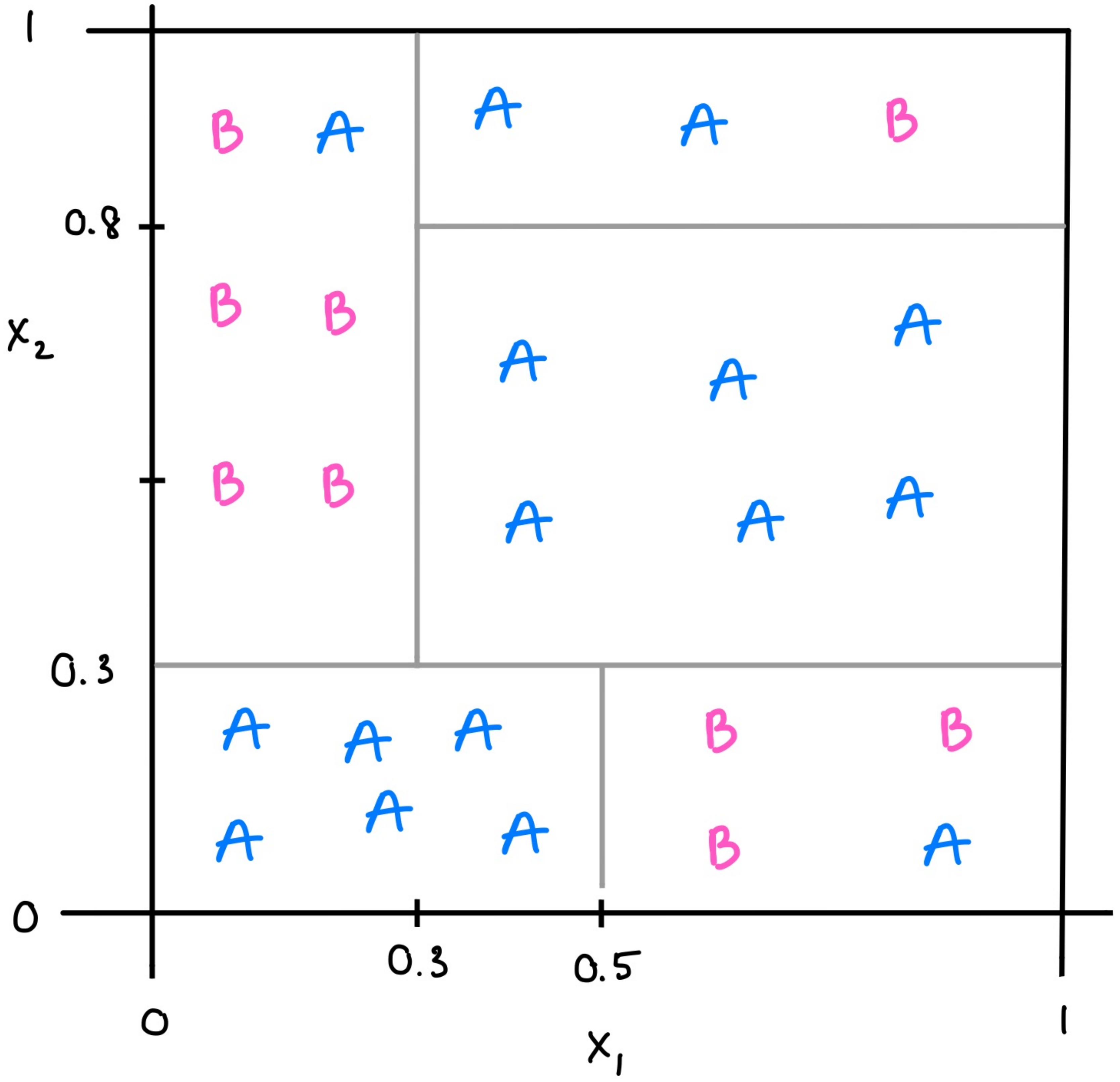
Neighborhoods via Recursive Partitioning





Decision Trees

Estimating Conditional Probabilities



$$\hat{p}_g(x) = \hat{P}\left[Y = g \mid X = x\right] = \frac{\sum_i I\left(y_i = g\right) I\left(x_i \in \mathcal{N}(x)\right)}{\sum_i I\left(x_i \in \mathcal{N}(x)\right)}$$

$$\hat{p}_A(x_1 = 0.9,\ x_2 = 0.1) = 1/4$$

$$\hat{p}_B(x_1 = 0.9,\ x_2 = 0.1) = 3/4$$

Decision Trees

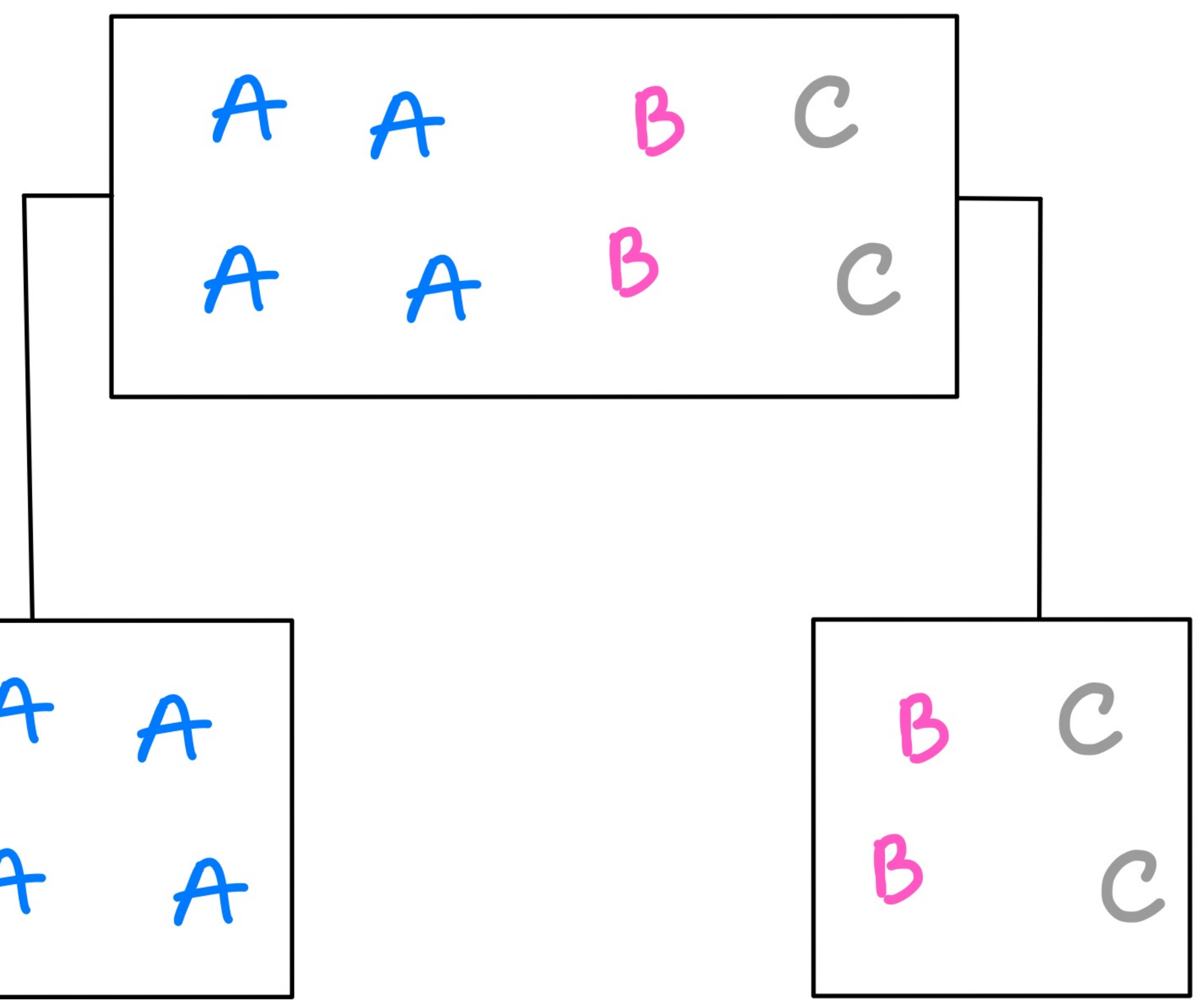
Node Probabilities

$$\hat{p}_g\left(\mathcal{N}\right) = \frac{\sum_i I\left(y_i = g\right) I\left(x_i \in \mathcal{N}\right)}{\sum_i I\left(x_i \in \mathcal{N}\right)}$$

$\hat{p}_A = 4/8$

$\hat{p}_B = 2/8$

$\hat{p}_C = 2/8$



$\hat{p}_A = 4/4$

$\hat{p}_B = 0/4$

$\hat{p}_C = 0/4$

$\hat{p}_A = 0/4$

$\hat{p}_B = 2/4$

$\hat{p}_C = 2/4$

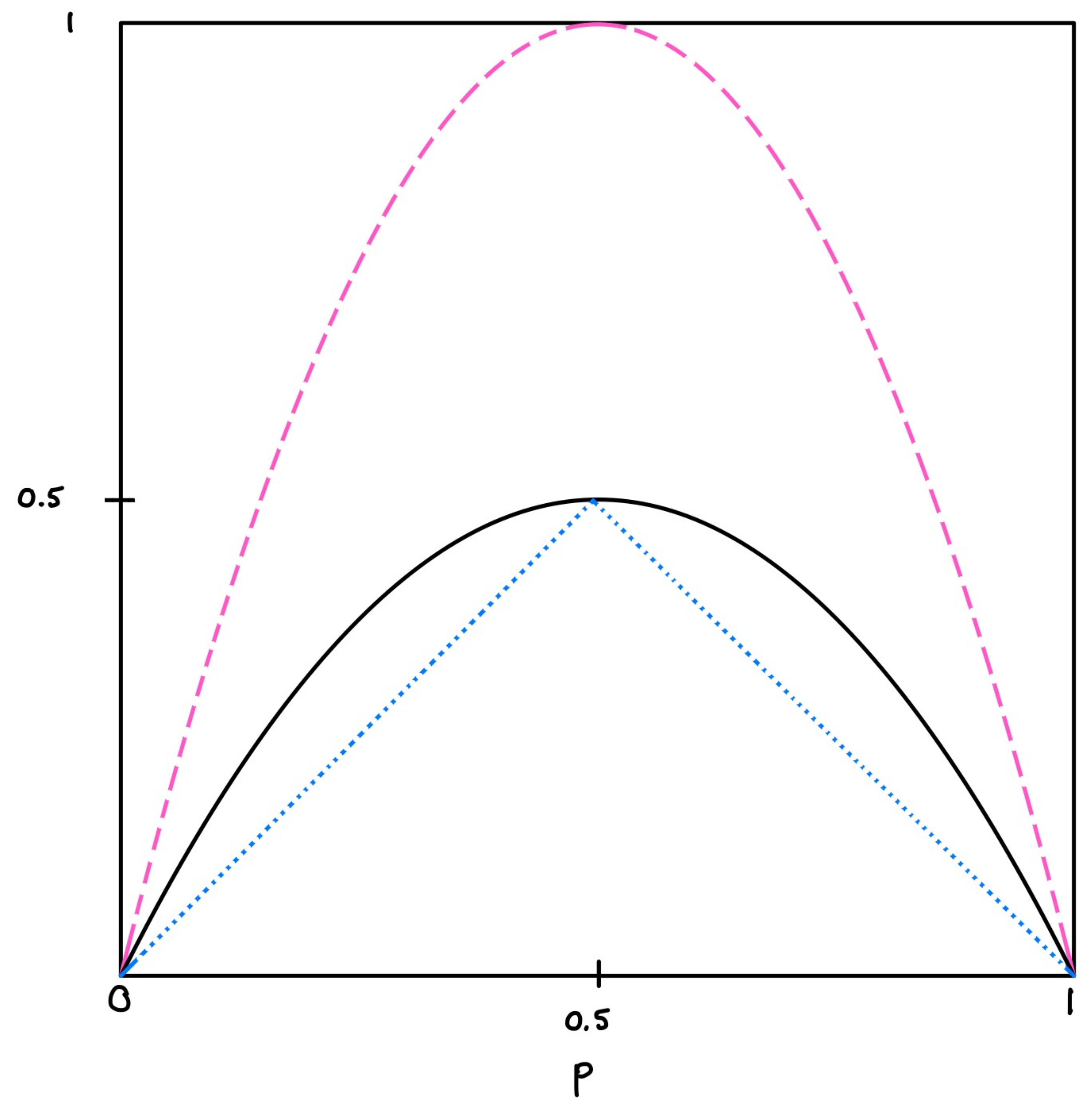
Decision Trees

Variance Measures for Nodes

$$\text{Gini}(\mathcal{N}) = \sum_{g=1}^{G} \hat{p}_g \left(1 - \hat{p}_g\right) = 1 - \sum_{g=1}^{G} \hat{p}_g^2$$

$$\text{Entropy}(\mathcal{N}) = -\sum_{g=1}^{G} \hat{p}_g \log\left(\hat{p}_g\right)$$

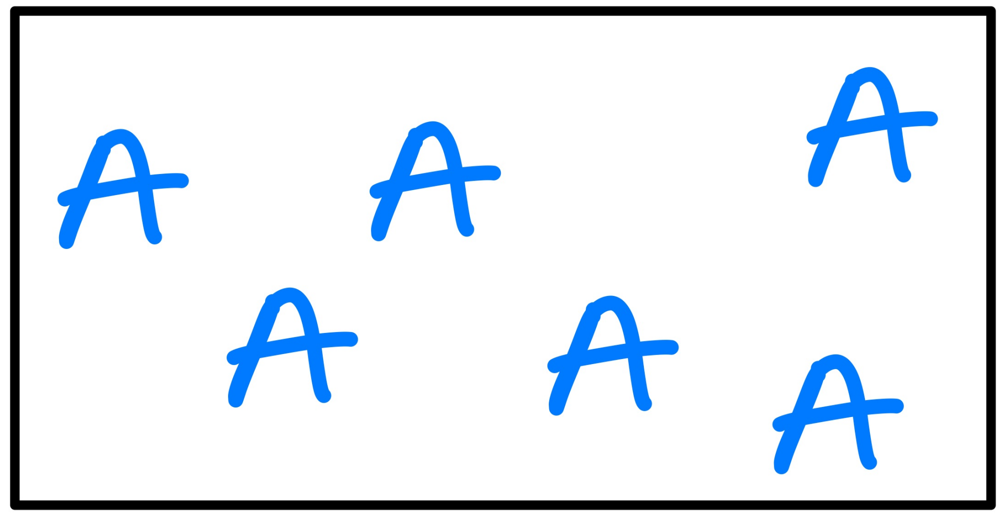
$$\text{Error}(\mathcal{N}) = 1 - \max_{g}\left(\hat{p}_g\right)$$

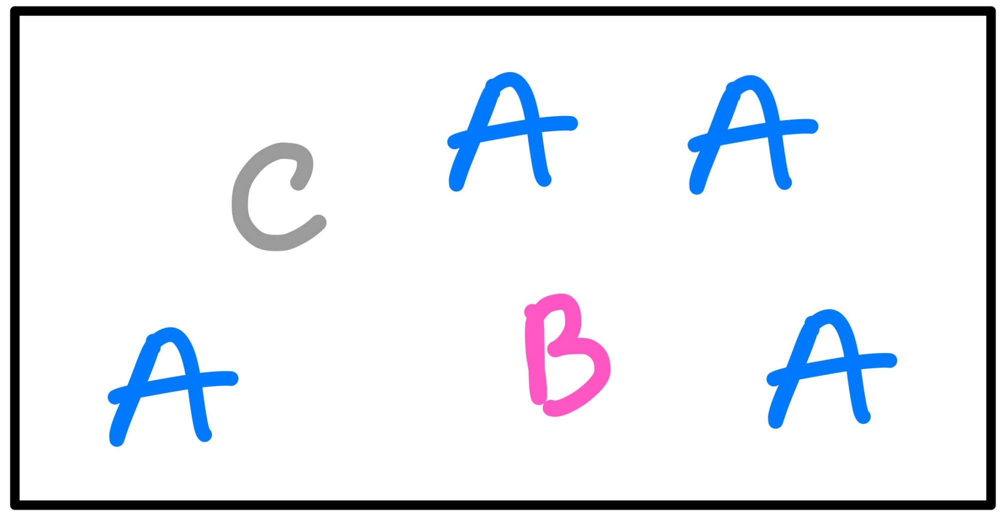


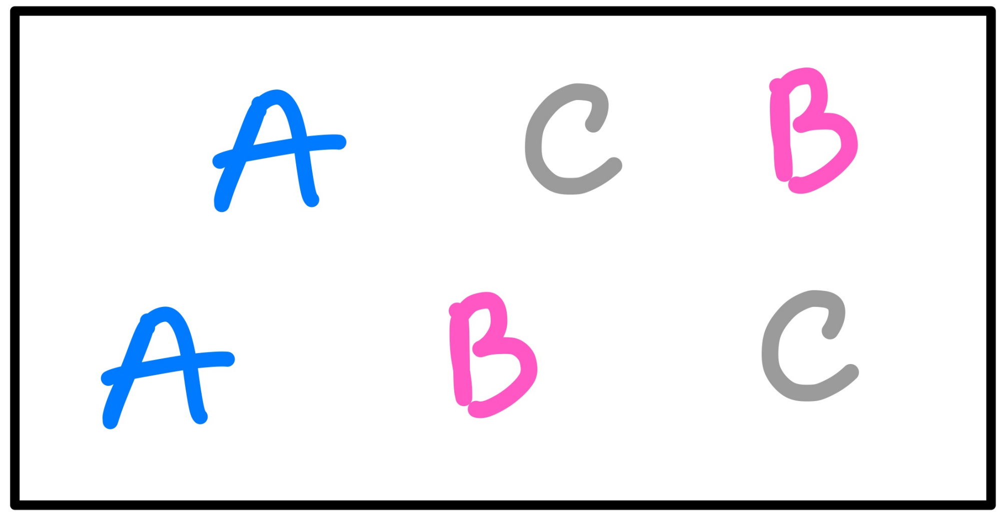
Decision Trees

Calculating Gini

$$\text{Gini}(\mathcal{N}) = 1 - \sum_{g=1}^{G} \hat{p}_g^2$$







$\hat{p}_A = 6/6$

$\hat{p}_B = 0/6$

$\hat{p}_C = 0/6$

$\text{Gini}(\mathcal{N}) = 0$

$\hat{p}_A = 4/6$

$\hat{p}_B = 1/6$

$\hat{p}_C = 1/6$

$\text{Gini}(\mathcal{N}) = 0.5$

$\hat{p}_A = 2/6$

$\hat{p}_B = 2/6$

$\hat{p}_C = 2/6$

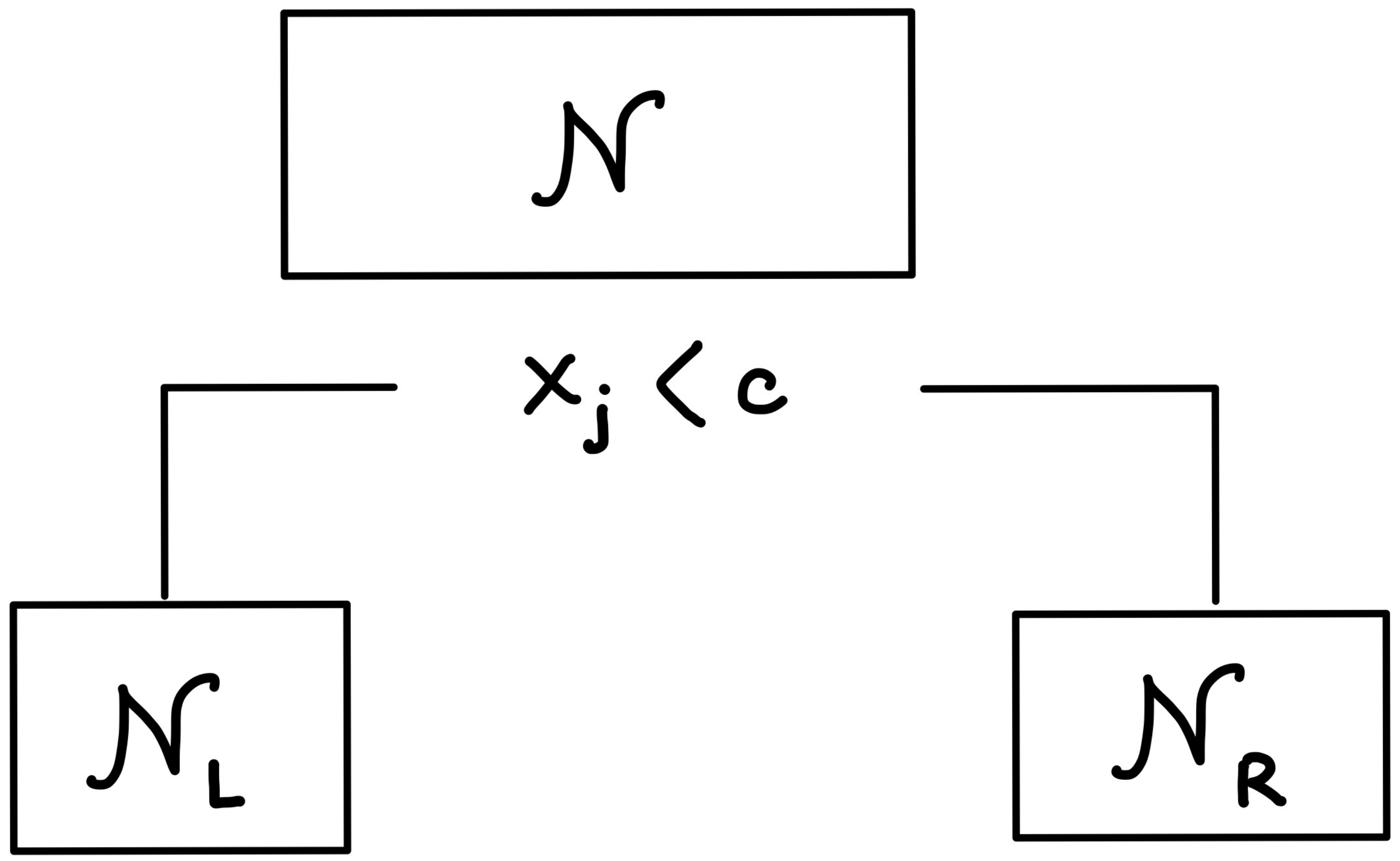
$\text{Gini}(\mathcal{N}) = 0.66\overline{6}$

Decision Trees

How To Split

Consider all splits of the node $\mathcal{N}$ of the form:

- Create node $\mathcal{N}_L$ where $x_j < c$.
- Create node $\mathcal{N}_R$ where $x_j \geq c$.



Determine the best split using:

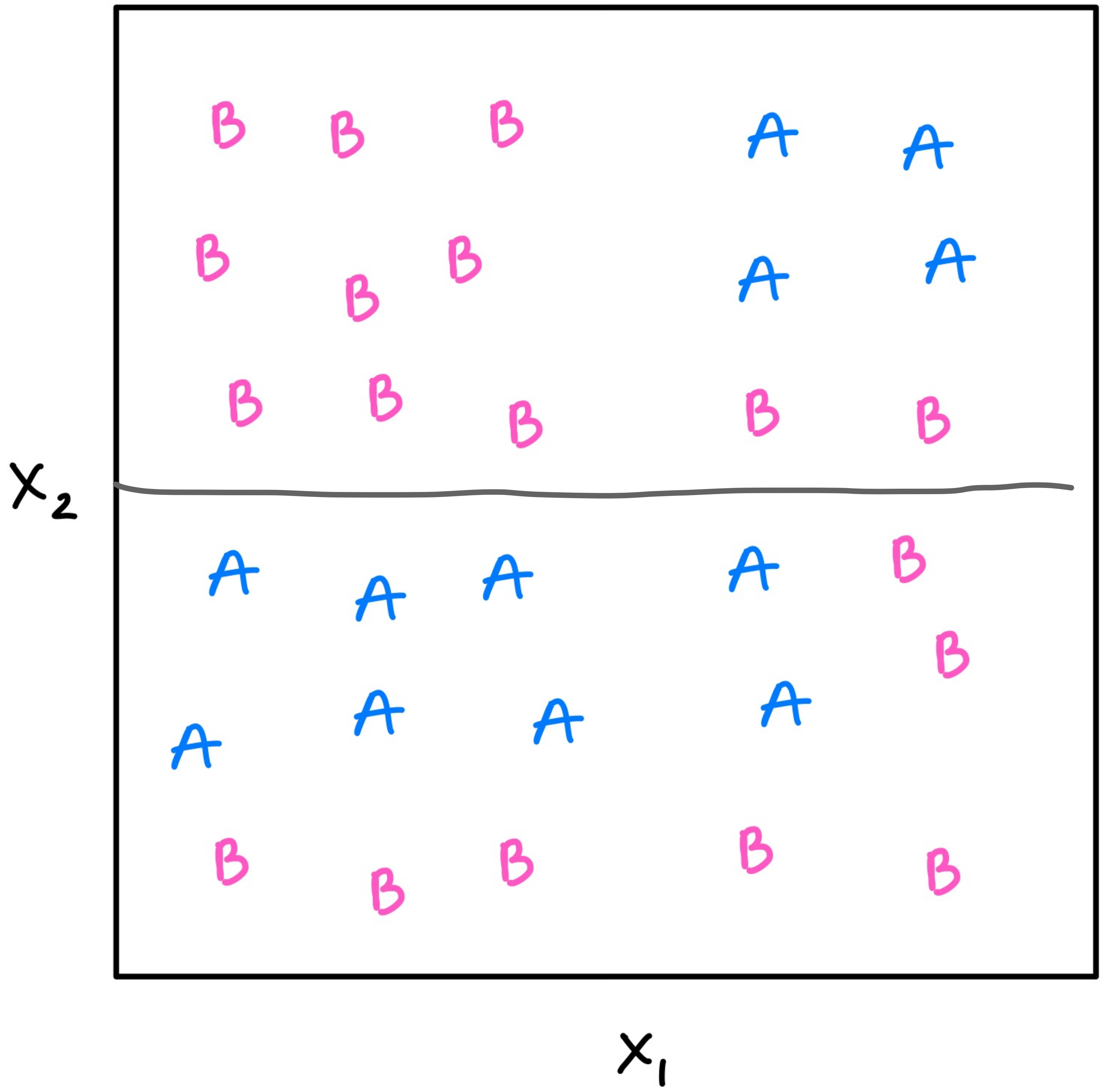
WEIGHTS

$$\min_{j,c}\left[\frac{|\mathcal{N}_L|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_L\right)+\frac{|\mathcal{N}_R|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_R\right)\right]$$

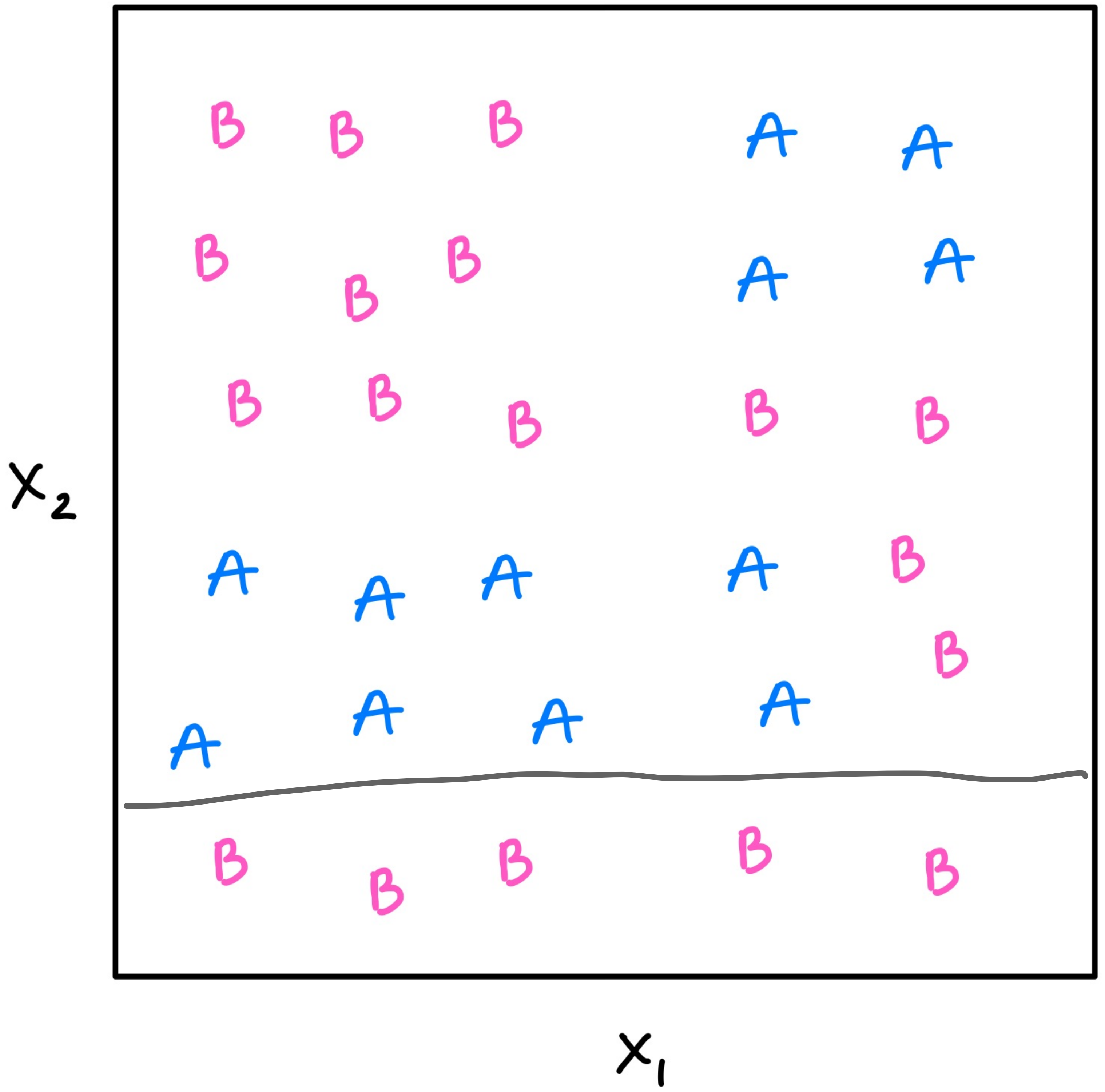
VARIANCES

Decision Trees

Which Split?



$$\frac{|\mathcal{N}_L|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_L\right) + \frac{|\mathcal{N}_R|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_R\right) = \quad 0.44\overline{4}$$



$$\frac{|\mathcal{N}_L|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_L\right) + \frac{|\mathcal{N}_R|}{|\mathcal{N}|}\text{Gini}\left(\mathcal{N}_R\right) = \quad 0.416$$

Decision Trees

Future Practical Considerations

- Many possible **tuning parameters** depending on specific implementation. These could include:
  - Minimum observations in node to split.
  - Minimum improvement to accept split.
  - Maximum tree depth.
- For splitting **numeric features**, only need to consider the midpoint between each of the order statistics of a feature.
- Beware: **categorical features**!
- Much ***faster*** than k-NN at prediction time.
  - This will be useful later when we grow entire forests instead of single trees.
  - We’ll also speed up training by adding **randomness**, which brings other benefits as well.
- Does feature **scaling** have an effect?
- Recommended **R** packages and functions: **`rpart::rpart, rpart.plot::rpart.plot`**