Exercise 1:

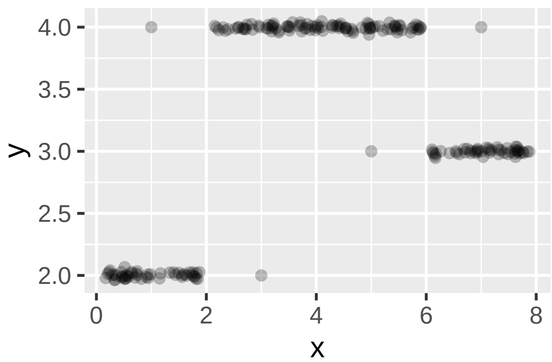
You are given the following table with the target variable `Banana`:

ID	Color	Form	Origin	Banana?
1	yellow	oblong	imported	yes
2	yellow	round	domestic	no
3	yellow	oblong	imported	no
4	brown	oblong	imported	yes
5	brown	round	domestic	no
6	green	round	imported	yes
7	green	oblong	domestic	no
8	red	round	imported	no

a) We want to use a Naive Bayes classifier to predict whether a new fruit is a `Banana` or not. Calculate the posterior probability $\hat{\pi}(\mathbf{x}_*)$ for a new observation $\mathbf{x}_* = (\text{yellow}, \text{round}, \text{imported})$. How would you classify the object?

b) Assume you have an additional feature `Length` that measures the length in cm. Describe in 1-2 sentences how you would handle this numeric feature with Naive Bayes.

Exercise 2:



The above plot shows $\mathcal{D} = \left(\left(\mathbf{x}^{(1)}, y^{(1)}\right), \ldots, \left(\mathbf{x}^{(n)}, y^{(n)}\right)\right)$, a data set with $n = 200$ observations of a continuous target variable y and a continuous, 1-dimensional feature variable $\mathbf{x}$. In the following, we aim at predicting y with a machine learning model that takes $\mathbf{x}$ as input.

a) To prepare the data for classification, we categorize the target variable y in 3 classes and call the transformed target variable z, as follows:

$$z^{(i)} = \begin{cases} 1, & y^{(i)} \in (-\infty, 2.5] \\ 2, & y^{(i)} \in (2.5, 3.5] \\ 3, & y^{(i)} \in (3.5, \infty) \end{cases}$$

Now we can apply quadratic discriminant analysis (QDA):

i) Estimate the class means $\mu_k = \mathbb{E}(\mathbf{x}|z = k)$ for each of the three classes $k \in \{1, 2, 3\}$ visually from the plot. Do not overcomplicate this, a rough estimate is sufficient here.

ii) Make a hand-drawn plot that visualizes the different estimated densities per class.

iii) How would your drawing from ii) change if we used linear discriminant analysis (LDA) instead of QDA? Explain your answer.

iv) Why is QDA preferable over LDA for this data?

b) Given are two new observations $\mathbf{x}_{*1} = -10$ and $\mathbf{x}_{*2} = 7$. State the prediction for QDA and explain how you arrive there.

c) We will now derive the LDA decision boundary for a binary problem in two dimensions. For this, we use the alternative description of the LDA decision rule we (implicitly) used to demonstrate LDA's linearity:

$$\delta_k(\mathbf{x}) = \log \pi_k - \frac{1}{2}\boldsymbol{\mu}_{\boldsymbol{k}}^T \Sigma^{-1} \boldsymbol{\mu}_{\boldsymbol{k}} + \mathbf{x}^T \Sigma^{-1} \boldsymbol{\mu}_{\boldsymbol{k}}, \quad k \in \{1, 2\},$$

with is also maximized to find the loss-optimal prediction. Explicitly state the equation for the boundary between the two classes.
(Hint: first think about the relation of the two discriminant functions on the decision boundary.)

d) Implement you own version of QDA:

```
train_qda <- function(target, data) {...}
predict_qda <- function(target, data) {...}
```

The first function should return the fitted model (use an adequate structure for the model!). The second function should return a factor of classes for `type = 'class'` and a matrix of predicted probabilities for `type = 'prob'` (similar to the standard `mlr3` output options). Your method only needs to work for numeric features. Check your implementation on the `iris` data and compare your results of both types with the `qda()` function from package `MASS`.

e) Turn your QDA classifier into a Naive Bayes classifier by appropriate modification. Can you see why NB is more "naive" than general QDA?