Supervised Learning: Learning from a randomly sampled batch of labeled data

Dataset (tuple of tuples)

Learner (function)

Predictor = Model = Hypothesis (function)

$(x_{1,1}, \ldots, x_{1,d})^T$

$D = ((\vec{x}_1, Y_1), (\vec{x}_2, Y_2), \ldots, (\vec{x}_n, Y_n))$

$D \in (\mathcal{X} \times \mathcal{Y})^n$

$(\vec{X}_i, Y_i) \sim \mathbb{P}_{\vec{X},Y}$

independent for all $i \in \{1, \ldots, n\}$

D n feature-label pairs

$\mathcal{X}$ set of features

$\mathcal{Y}$ set of labels/targets

Ex: $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \mathbb{R}$

$\hat{f} : \mathcal{X} \to \mathcal{Y}$

f a function from features to labels

Ex $\hat{f}(x) = 2x + 1$, $\mathcal{X} = \mathbb{R}$

$\mathcal{A} : (\mathcal{X} \times \mathcal{Y})^n \to \{f \mid f : \mathcal{X} \to \mathcal{Y}\}$

$\mathcal{A}$ a function from datasets to predictors

Ex: $\mathcal{A}(D) = \hat{f}$ where $\hat{f} : \mathcal{X} \to \mathcal{Y}$

$$\hat{f}(x) = \begin{cases} y_i & \text{if } x = x_i \text{ for some } i \in \{1, \ldots, n\} \\ 0 & \text{otherwise} \end{cases}$$

Ex (of features and labels/targets):

$\vec{X}_i \in \mathbb{R}^3$ # of rooms, # of floors, age of a house

$Y_i \in \mathbb{R}$ price

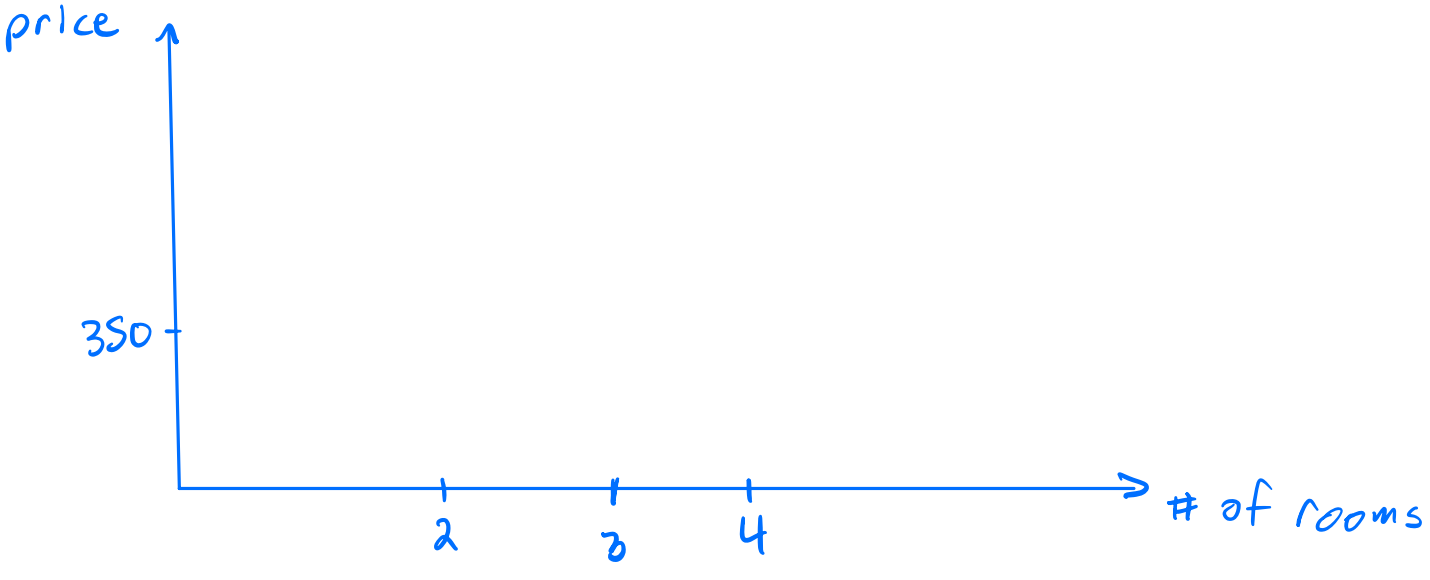
$\vec{X}_i \in \mathbb{R}^2$ amount of chemical 1, amount of chemical 2 in a wine

$Y_i \in \{0, 1\}$ type of wine

$\vec{X}_i \in \mathbb{R}^{400}$ pixel value of a $20 \times 20 = 400$ pixel image

$Y_i \in \{\text{cat}, \text{dog}, \text{bird}\}$ type of animal

What is a feature and what is a label is a design choice. Usually a feature is info that is easy to gather. And the label is hard, which is why you want to predict it


price
350
2
3
4
# of rooms

Regression: $Y \in \mathcal{Y}$ represent something with a notion of order

(Usually $\mathcal{Y}$ is $\mathbb{R}$ or some interval)

Ex: house prices, stock prices, energy consumption, weather prediction

We use:

$$\ell(f(\vec{X}), Y) = |f(\vec{X}) - Y| \quad \text{absolute loss}$$

or

$$\ell(f(\vec{X}), Y) = \left(f(\vec{X}) - Y\right)^2 \quad \text{squared loss}$$

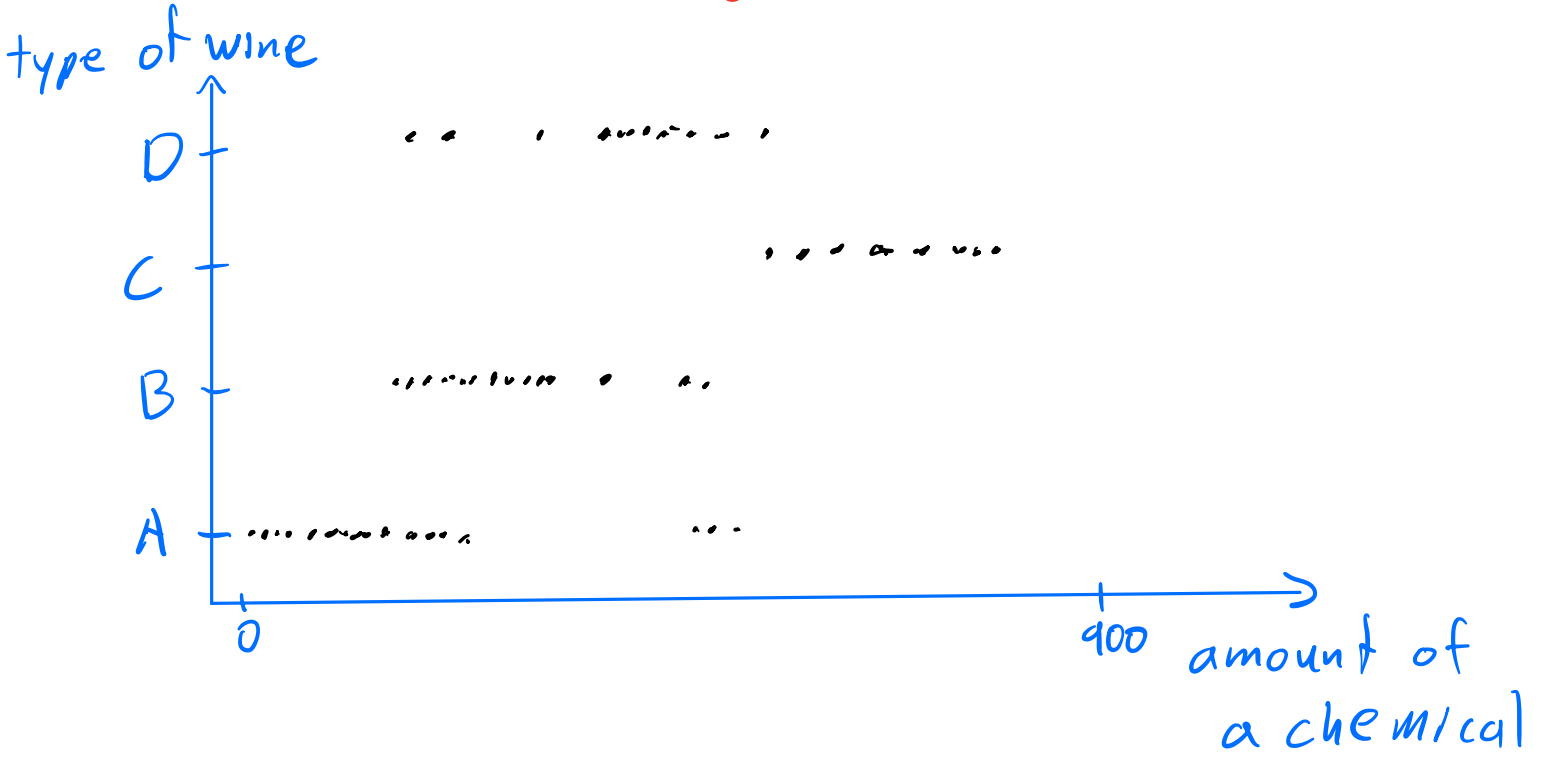
Classification: if $Y \in \mathcal{Y}$ represents something without order

(Usually $\mathcal{Y}$ is finite)

Ex (of $\mathcal{Y}$): type wines, type of image, type of email, type of disease

Ex: $f(\vec{x})$ is a predictor that takes as input the amount of a chemical in a wine and outputs the type of wine

Suppose you got multiple wines, what would a good f be?



for ℓ we use:

$$\ell(f(\vec{x}), Y) = \begin{cases} 0 & \text{if } f(\vec{x}) = Y \\ 1 & \text{otherwise} \end{cases}$$

0-1 loss

Ex: $L(f)$ if we use 0-1 loss $\quad y = \{A, B, C, D\}$

$$L(f) = \mathbb{E}[\ell(f(\vec{x}), Y)] = \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} \ell(f(\vec{x}), y)\, p(x, y)\, dx$$

$$= \int_{\mathcal{X}} \left(\sum_{y \in \mathcal{Y}} \ell[f(\vec{x}), y)\, p(y|x) \right) p(x)\, dx$$

Ex: Let $\mathcal{F}$ be all linear functions

ERM picks the line that best fits the data