07-Linear Classification (Formulation)

Regression vs Classification

- A supervised task concerns the prediction of $t \in \mathcal{Y}$ based on input $\boldsymbol{x} \in \mathcal{X}$, i.e., learning a function $h: \mathcal{X} \to \mathcal{Y}$
- Regression: $t \in \mathbb{R}$
- Classification: $t \in \{c_1, \cdots, c_K\}$

  We call $c_1, \cdots, c_K$ classification targets, labels, objectives, outputs, candidates, categories, etc.
- For notational reasons, we use t to represent the output label in classification.

- In regression, there is an order and a distance defined on different values of t. But there is no order information in classification. The distance between $c_1, \cdots, c_K$ is uninformative by the formulation of the task itself.

- Example (spam detection): We would classifier if an email is spam. In this case, $y \in$ {Spam, NotSpam}. Therefore, spam detection formulated as such is a classification task. It is noted that we can let Spam = 1 and NotSpam = 0, so that $t \in \{0,1\}$. If t can choose exactly two values, we call it a binary classification task.

- Example (sentiment analysis): We would like to judge whether a sentence expresses positive sentiment or negative sentiment. If we follow the schema as in Amazon, say, then $t \in \{1, 2, 3, 4, 5\}$, also being a classification task. Since y can choose multiple values, it is known as a multi-class classification.

[Taken from Amazon.ca]

- Example (sentiment analysis): Alternatively, we may formulate the task by regression models, where we predict $t \in \mathbb{R}$, and round it to integers of 1, 2, ..., 5

regression models, where we predict $t \in \mathbb{R}$, and round it to integers of 1, 2, …, 5. Such formulation is a regresssion problem. This shows:

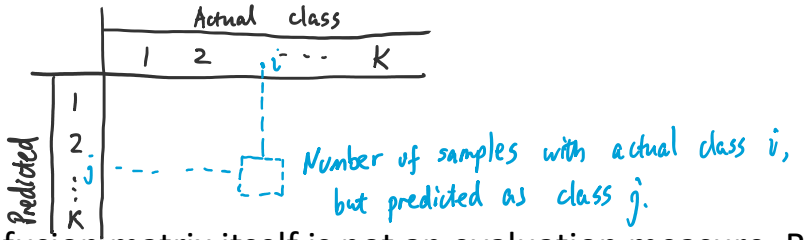
- A task can be either formulated as a regression problem or a classification problem.
- Regression for sentiment analysis is explicitly making use of
  - *Order information*: 2-star is more positive than 1-star, 3-star is more positive than 1-star and 2-star, etc.
  - *Distance information*: The difference between 1-star and 3-star is the same as the distance between 2-star and 4-star, but twice as much as the distance between 1-star and 2-star, etc.
- Classification for sentiment analysis has no order or non-trivial distance information
  - It does not matter if I map, say, 1-star to 2, 2-star to 5, 3-star to 1, 4-star to 4, and 5-star to 2, or any other labels.
  - The distance information between target labels cannot be obtained from the task formulation itself, but can in fact, be learned during training. For example, it is possible to learn that 3-star and 2-star is closer than 2-star and 1-star.
  - Usually, the training objective for classification saturates if we make a correct prediction, which better suits the classification task.
  - The rule of the thumb for modeling a task by regression or classification is to do validation.

- **Multi-label classification**
  - In the default terminology of classification, we usually refer to binary or multi-class classification, i.e., we only associate one label to a data sample, i.e., $t \in \{c_1, \cdots, c_K\}$
  - In certain applications, we may associate more than one label to a data sample, i.e., $\boldsymbol{t} \subseteq \{c_1, \cdots, c_K\}$ where $\boldsymbol{t}$ is the set of labels associated with this data sample. This is sometimes known as multi-label classification.
  - Example (emotion analysis): If we would like to tag a sentence with emotional labels, it is possible that a sentence can be both sad and angry. Assuming sad and angry are in the set of all labels, we have multi-label classification.
  - The most naïve approach to multi-label classification is to perform individual binary classification for $c_1, \cdots, c_K$, i.e., we classifiy $t_k \in \{0,1\}$ indicating if $c_k \in \boldsymbol{t}$ for $k = 1, \ldots, K$.
  - In this case, we can solve multi-label classification by multi-class classification techniques. Advanced research in multi-label classification focuses on modeling the relationship among different labels of t_k.

- In this course, we mostly focus on multi-class classification.

Evaluation measures (measure of success)

- Accuracy $\frac{\#\text{ correctly predicted}}{\#\text{ total samples}}$
- Weighted risk
  - In medical test, for example, false negative is worse than false positive. Suppose a false negative is 10 times as disastrous as a false positive, then the weighted risk could be 10 * #false negative + 1 * #false positive
  - In a multi-class scenario, we may have a confusion matrix:



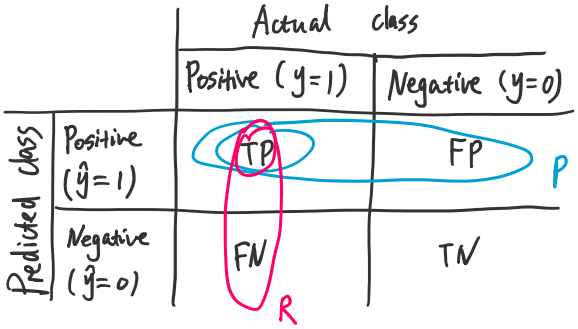
The confusion matrix itself is not an evaluation measure. But we may assign weight to every cell (although this could be difficult). On the other hand, the confusion matrix is also important for diagnostic purposes.

- Precision, Recall, F-score
  - Consider a binary classification task, and the class distribution is skewed to the category 0 (containing more negative samples $t = 0$ than positive $t = 1$).

Precision (P) = $\frac{|\{\hat{t}=1\} \cap \{t=1\}|}{|\{\hat{t}=1\}|}$

Recall (R) = $\frac{|\{\hat{t}=1\} \cap \{t=1\}|}{|\{y=1\}|}$

F_β-score = $\frac{(1+\beta^2) \cdot P \cdot R}{\beta^2 \cdot P + R}$



Usually, we use F_1-score, i.e., $F_1 = \frac{2P \cdot R}{P + R}$

- In a skewed-class prediction, we cannot easily cheat the F-measure.
  - Written assignment:
    - Suppose we have 10% positive samples and 90% negative samples,
    - Compute P, R, and F_1 for majority guess and uniform random guess.
      - Majority guess

$$\hat{t} = \arg\max_t \sum_{m=1}^{M} \mathbb{1}\{t^{(i)} = t\}$$

- (Uniform) random guess

$$\hat{t} \sim 0.5\langle 0\rangle + 0.5\langle 1\rangle$$

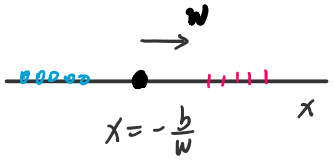
- We mentioned that the positive category (it doesn't matter you call it $t = 0$ or $t = 1$) should be the minority class. What if we compute P, R, and F_1 by saying 90% samples are positive and 10% negative? Compute the numbers and discuss the consequences.

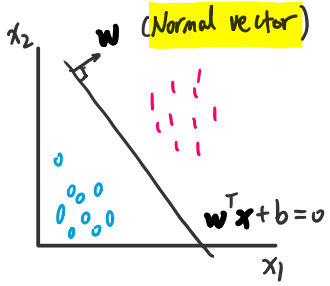
Linear classification hypothesis class

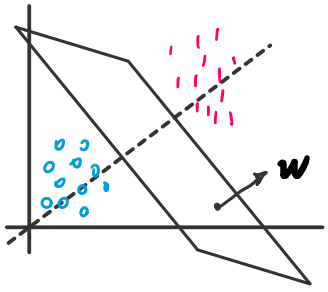
- Let input $\boldsymbol{x} = (x_1, x_2, \cdots x_d)^T \in \mathbb{R}^d$ and output $t \in \{0, 1\}$
- A linear classification hypothesis class can be represented as

$$\mathcal{H} = \{h_{\boldsymbol{w},b} : h_{\boldsymbol{w},b}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{w}^T\boldsymbol{x} + b \geq 0 \\ 0 & \text{otherwise} \end{cases}\}$$

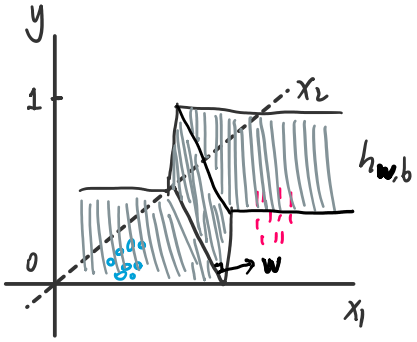
The hyperplane $\boldsymbol{w}^T\boldsymbol{x} + b = 0$ is known as the decision boundary/surface







A hypothesis $h_{\boldsymbol{w},b}$ is a thresholded mapping from the input space $\mathbb{R}^d$ to the output space {0,1}, with the threshold being $\boldsymbol{w}^T\boldsymbol{x} + b = 0$.

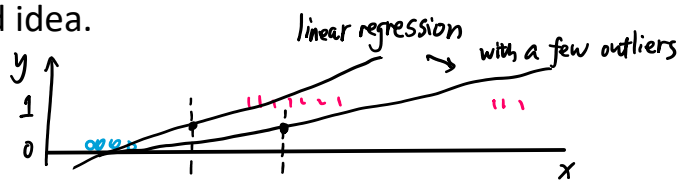


This cartoon shows the hypothesis function $h_{\boldsymbol{w},b}$ for linear classification is a step function, different from linear regression. In future study, however, we rarely use this but just draw the decision boundary in the $\boldsymbol{x}$ space.

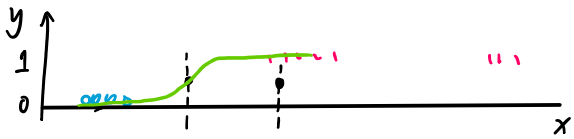
A few heuristics for classification

- (Bad idea) Linear regression for classification
  - Least square regression being the best linear unbiased estimate (BLUE) shows theoretical guarantee with the assumption of Gaussian noise.
  - For classification, y is not Gaussian distributed. Thus, there's no gurantee so far for MSE in a classification task.

- On the contrary, the below examples shows that MSE for classification is a bad idea.



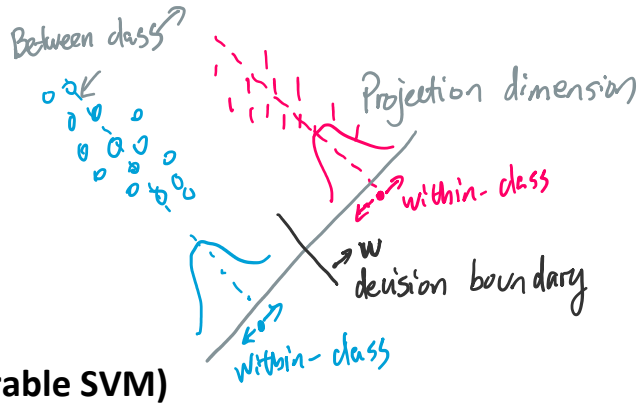
  - MSE penalizes for well predicted samples (those $\hat{y} \gg 1$)
  - Prone to outliers
  - Fix: squash the linear line as some bounded curve.



  - New problem: MSE is designed to learn closeness rather than correctness. Suppose $y = 1$, MSE is too weak for a prediction of 0.1, compared with 0.9
    => Need more principled way of training a classifier

- **Fisher's linear discriminant**

$$\text{maximize } \frac{\text{Between class variance}}{\text{Within class variance}}$$



- **Max-margin classification (linearly separable SVM)**
  maximize the margin