IAML 2018-2019

1. NB to classify e-mail

   a. Define NB classification method + correct equations to make definitions precise. Why is it naïve? How can the rel. prob dists be estimated from data.

   Use Bayes theorem to get the probability $p(c|\underline{x})$ where c is the class (spam / ham) and $\underline{x}$ is the input vector.

   $$p(c|\underline{x}) = \frac{p(\underline{x}|c)p(c)}{\sum_{i=1}^{K} p(\underline{x}|c_k)p(c_k)}$$

   This is naive because it assumes that the atts in the input vector are independent of each other.

   Count the occurrence of the word within a document class.

   b. You gather 3 "spam" e-mails 3/8

   [1 0 1 0 1], [0 1 0 0 1], [1 1 0 1 0]

   and 5 examples of "ham" emails 5/8

   [0 0 1 1 0], [0 1 0 1 0], [1 0 0 1 0], [0 1 0 0 1], [0 1 0 0 1]

   Classify [1 0 1 0 1]

	1	0	1	0	1
	1	2	3	4	5
spam	2/3	2/3	1/3	1/3	2/3
ham	1/5	3/5	1/5	3/5	2/5

$p(x|\text{spam}) = 2/3 \times 1/3 \times 1/3 \times 2/3 \times 2/3 = 8/243$

$p(x|\text{ham}) = 1/5 \times 2/5 \times 1/5 \times 2/5 \times 2/5 = 8/3125$

$$p(\text{spam}|x) = \frac{8/243 \times 3/8}{8/243 \times 3/8 + 8/3125 \times 5/8} = \frac{625}{706} = 0.88$$

$p(\text{ham}|x) = 1 - 0.88 = 0.12$

so it is SPAM!

c. Use Information Gain to select 5 words.

$$Gain(S,A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v)$$

Decrease on entropy of S after knowing att. A.
$H(S) = -P_{(+)} \log_2 P(+) - P_{(-)} \log_2 P(-)$
Biased over attributes with more values.

d. Delete SPAM

classifier biased towards HAM because fewer training examples

→ affects prior probability.

e. Should we use k-NN

Yes, no training phase in kNN, new data can be classified immediately.

However, it is very computationally expensive with a large volume of data.

f. Customer e-mail relates to topics, but you DK what the topics are. Topics are rep. in presence/absence of words. How to discover topics and group e-mails?

↗ can't use K-means bc you don't know k

Unsupervised learning → look for structure, no labels
→ generative

Use a clustering algorithm!

Agglomerative
→ start with individual clusters
→ group the clusters between distance bet. clusters, i.e. single link

Piazza Answer

1. Represent data as binary vector of indicators for each word as only the presence or absence of words.
2. K-Means clustering w/ Hamming distance measure.
   → group the documents to K clusters because we require an unsupervised method of grouping the data.
3. Use extrinsic evaluation through user quality assessment to assess the choice of K: the point of the choosing the topics is to group them into useful groups for managing emails.

2. a. LOGISTIC REGRESSION

$\underline{w} = (w_0, w_1)$ are the real-valued params of this classifier.

$p(y=1|x) = \sigma(w_0 + w_1 x)$

class label y, $y \in \{0, 1\}$

sigmoid function $\sigma(z) = \frac{1}{1+\exp(-z)}$

Let $z = w_0 + w_1 x$. the likelihood for a single (x, y) instance is given by $L(\underline{w}) = \sigma(z)^y (1-\sigma(z))^{1-y}$.

$\ell(\underline{w})$ = log likelihood for the logistic regression model.

1. Show that $\partial \ell / \partial w_0 = \sum_{i=1}^{n} [y_i - \sigma(z_i)]$

Note that $\sigma'(z) = \sigma(z)(1-\sigma(z))$.
Also calculate $\partial \ell / \partial w_1$.

$$\ell(w) = \log[\sigma(z)^y (1-\sigma(z))^{1-y}]$$
$$= \log(\sigma(z)^y) + \log((1-\sigma(z))^{1-y})$$
$$= y \cdot \log(\sigma(z)) + (1-y)\log(1-\sigma(z))$$

$$\partial \ell / \partial w_0 = \frac{d\ell}{\partial w_0}\left[y \cdot \log(\sigma(w_0 + w_1 x)) + (1-y)\log(1-\sigma(w_0 + w_1 x))\right]$$

TO maximise, you have to maximise the likelihood of the entire dataset.

$$\ell(\underline{w}) = \sum_{i=1}^{n} y_i \log \sigma(w_0 + w_1 x) + (1-y_i)\log(1-\sigma(w_0 + w_1 x))$$

$\sigma' = \sigma(w_0 + w_1 x) \cdot (1 - \sigma(w_0 + w_1 x)) \cdot 1$ ↗ 1

$f'(x) = \frac{\sigma(w_0 + w_1 x)\cdot(1-\sigma(w_0 + w_1 x))}{\sigma(w_0 + w_1 x)}$ ✓ $1 - \sigma(1)$

$f'(x) = 0 - \frac{\sigma(w_0 + w_1 x)(1-\sigma(w_0 + w_1 x))}{\sigma(w_0 + w_1 x)}$

$= 0 - 1 + \sigma(w_0 + w_1 x) - \sigma(w_0 + w_1 x)$

$= y_i[(1-\sigma(w_0 + w_1 x)] + (1-y_i)[\sigma(w_0 + w_1 x)]$

$= y_i - y_i\sigma[w_0 + w_1 x] + (\sigma(w_0 + w_1 x) - y_i\sigma(z))$

$= y_i - y_i \cdot \sigma(z) + \sigma z - y_i \cdot \sigma(z)$

$= y_i [1 - \sigma(z_i)] + (1 - y_i)[\sigma(z_i)]$

$= y_i - y_i \sigma(z_i) + \sigma(z_i) - y_i \sigma(z_i)$

$\sigma(w_0 + w_1 x) = \sigma(w_0 + w_1 x) \cdot (1 - \sigma(w_0 + w_1 x) \cdot 1$

$= y_i [(1 - \sigma(w_0 + w_1 x)] + (1 - y_i)[-1 + \sigma(w_0 + w_1 x))$

$y_i - y_i \sigma(z_i) \quad - (1 - y_i)(1 - \sigma(w_0 + w_1 x))$

$- [1 - \sigma(w_0 + w_1 x)$

$- y_i$

GENERAL with w_j *correct

$$\partial l / \partial(w_j) = \sum_{i=1}^{n} y_i \frac{1}{\sigma(\underline{w}^T \underline{x})} \cdot \sigma(\underline{w}^T \underline{x})(1 - \sigma(\underline{w}^T \underline{x}) \cdot x_{ij}$$

$$+ (1 - y_i) \cdot \frac{1}{1 - \sigma(\underline{w}^T \underline{x})} \cdot (-\sigma(\underline{w}^T \underline{x})(1 - \sigma(\underline{w}^T \underline{x})) x_{ij}$$

$$= \sum_{i=1}^{n} y_i (1 - \sigma(\underline{w}^T \underline{x})) x_{ij} + (1 - y_i)(-\sigma(\underline{w}^T \underline{x})) x_{ij}$$

$$- \sum_{i=1}^{n} x_{ij} [y_i - y_i \sigma(\underline{w}^T \underline{x}) - \sigma(\underline{w}^T \underline{x} + y_i \sigma(w^T x)]$$

$$= \sum_{i=1}^{n} x_{ij} [y_i - \sigma(\underline{w}^T \underline{x})]$$

$$\partial l / \partial(w_0) = \sum_{i=1}^{n} [y_i - \sigma(z_i)] \quad * \text{ for } w_0, x \text{ always } 1$$

$$\partial l / \partial(w_1) = \sum_{i=1}^{n} x_i [y_i - \sigma(z_i)]$$

ii. Data

Instance	x	y
1	0.0	1
2	-0.5	0

$\underline{w} = (-1, 1)$ $\quad p = (y=1 \mid x)$

One step gradient ascent. $\eta = 0.2$.

Instance	x	y	$\sigma(\underline{w}^T \underline{x}_i)$ ← $P(y=1 \mid x_i)$	$y_i - \sigma(\underline{w}^T \underline{x}_i)$
1	0.0	1	0.269	0.731
2	-0.5	0	0.182	-0.182

$x_0 = 1$ (bias)

$1 \rightarrow [-1 \ 1] \begin{bmatrix} 1 \\ 0 \end{bmatrix} = -1 \rightarrow \sigma(-1) = \frac{1}{1+\exp(1)} = 0.269$

$2 \rightarrow [-1 \ 1] \begin{bmatrix} 1 \\ -0.5 \end{bmatrix} = -1 - 0.5 \rightarrow \sigma(-1.5) = \frac{1}{(1+\exp(1.5)} = 0.182$

$\partial y / \partial w_0 = (1 \times 0.731) + (1 \times (-0.182)) = 0.549$

$\partial y / \partial w_1 = (0 \times 0.731) + (-0.5 \times -0.182) = 0.091$

$$w' = w + \eta g = \begin{bmatrix} -1 \\ 1 \end{bmatrix} + 0.2 \begin{bmatrix} 0.549 \\ 0.091 \end{bmatrix}$$

$$= \begin{bmatrix} -0.8902 \\ 1.0182 \end{bmatrix}$$

b. GENERALIZATION and OVERFITTING

i. Gen. err vs. Training error

Gen-err: error on future (unseen) dataset

Training error: mispredictions on training data.

$$E_{trn} = \frac{1}{n} \sum_{i=1}^{n} \text{error}(f_D(x_i), y_i)$$

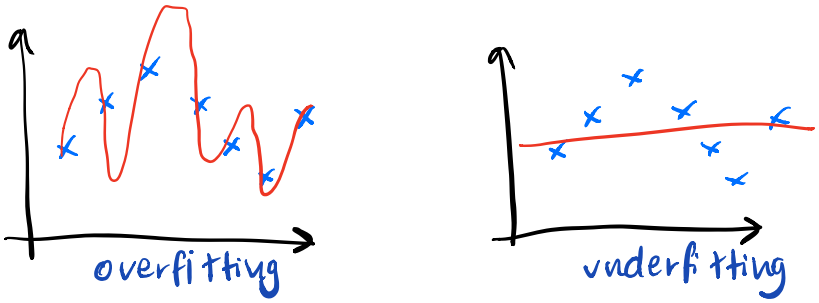
ii. Equation for generalisation error

$$E_{gen} = \int \underbrace{\text{error}(f_D(x), y)}_{\text{error as before}} \underbrace{p(y,x)}_{\text{how often we see such } x \text{ and } y} dx$$

over all possible x and y

iii. Graph for overfitting and underfitting

makes more mistakes on unseen data but fewer on trn
makes more mistakes on both unseen and trn.



iv. Generalisation: purpose, method for lin. reg, op. approach.

- adds complexity parameter to a learning algorithm.
- example method
  - ridge regression ⇒ squared error w/ quadratic regularisation

$$E(w) = |y - \phi w|^2 + \lambda |w|^2$$

$$\hat{w} = (\phi^T \phi + \lambda I)^{-1} \phi^T y$$

- optimisation approach: calculus

3. a. SUPPORT VECTOR MACHINES (SVMs)
2-class classification

i. Maximum Margin Hyperplane

Margin → distance between the decision boundary and the closest training point.

Maximum Margin Hyperplane
→ the decision boundary that has the maximum distance towards the closest training point.

i.e. the hyperplam $w^T x + w_0$ such that

$$\min_i |w^T x_i + w_0| = 1 .$$

ii. Solution for weight vector for the maximum margin hyperplane, explain support vector, and how a class prediction is made for a new test input vector.

margin → $\min_i \cdot \frac{1}{\|w\|} |w^T x_i + w_0|$

support vectors are the vectors closest to the maximum margin hyperplane. they have direct bearing on the optimum location of the MMH.

predict 1 if sign $(w^T x + w_0)$ positive,
0 if sign $(w^T x + w_0)$ negative.

iii. Non-linear SVMs map from the input space x to a feature space $\underline{x}$ to a feature space $\phi(x)$.
The 'kernel trick' can be used to compute $\phi(x) \cdot \phi(y)$ efficiently.

verify that

$$\phi(\underline{x}) = \begin{bmatrix} 1 \\ 2x_1 \\ 2x_2 \\ 2x_1^2 \\ 2\sqrt{2}x_1 x_2 \\ 2x_2^2 \end{bmatrix}$$

corresponds to the kernel $K(x,y) = (1 + 2(x \cdot y))^2$
for $x, y \in \mathbb{R}^2$.

$K(x_i, x_j) = x_1^T x_2 = \phi(x)^T \phi(y)$

$$\begin{bmatrix} 1 & \sqrt{2}x_1 & \sqrt{2}x_2 & x_1^2 & \sqrt{2}x_1x_2 & x_2^2 \end{bmatrix} \begin{bmatrix} 1 \\ \sqrt{2}y_1 \\ \sqrt{2}y_2 \\ y_1^2 \\ \sqrt{2}y_1y_2 \\ y_2^2 \end{bmatrix}$$

$$\phi(x)^T\phi(y) = 1 + 4x_1y_1 + 4x_2y_2 + 4x_1^2y_1^2 + 8x_1x_2y_1y_2 + 4x_2^2y_2^2$$

$$= 1 + 4(x_1y_1 + x_2y_2) + 4(x_1y_1 + 2x_1x_2y_1y_2 + x_2^2y_2^2)$$

$$= 1 + 4(x \cdot y) + 4(x \cdot y)^2$$

$$= [1 + 2(x \cdot y)]^2$$ hence proved.

iv. Compare with SVM with linear kernel, i.e. $K(x,y) = x \cdot y$

logistic regression
- dense
- all points affect

only SVs affect
if N is much more sparse
$\sum_{i=1} a_i' x_i y_i$
↳ most are zero
$a \neq 0 \rightarrow$ SV, on margin

?

b. EVALUATION and ROC curves

$$TPR = \frac{TP}{TP + FN} = \frac{\#\text{ positives correctly classified}}{\#\text{ total positives}}$$

$$FPR = \frac{FP}{FP + TN} = \frac{\#\text{ negatives incorrectly classified}}{\#\text{ total negatives}}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{\#\text{ correctly classified}}{\#\text{ total}}$$

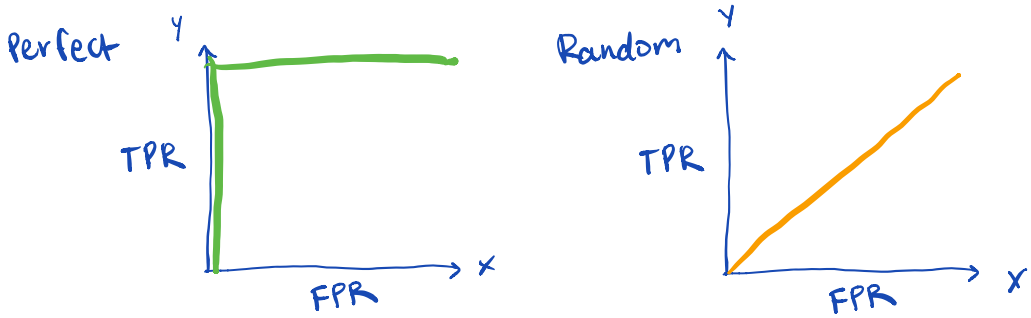
i. Simple strategy that will classify novelists w/ at least 99% accuracy. TPR? FPR?

Classify to the most probable class (class with most instances → highest prior).

(?) TPR:

FPR:

ii. ROC curve



Why does the perfect curve represent the best possible classifier?

$TPR = \frac{X}{0}$ $FPR = \frac{0}{X}$

Really high true positive rate and very low false positive rate.

iii. positives = {0.9, 0.4, 0.7, 0.8}
negatives = {0.1, 0.7, 0.2, 0.3, 0.2, 0.5, 0.3, 0.6}
TPR FPR thresholds: 0.00, 0.25, 0.45, 0.65, 1.00
sketch ROC for positive class!

$T = 0.00 \rightarrow$ TP = 4, FN = 0 — $TPR = \frac{4}{4} = 1.0$
FP = 8, TN = 0 — $FPR = \frac{8}{8} = 1.0$ (1.0, 1.0)

$T = 0.25 \rightarrow$ TP = 4, FN = 0 — $TPR = \frac{4}{4} = 1.0$ (0.6, 1.0)
FP = 5, TN = 3 — $FPR = \frac{5}{8} = 0.6$

$T = 0.45 \rightarrow$ TP = 3, FN = 1 — $TPR = \frac{3}{4} = 0.75$ (0.375, 0.75)
FP = 3, TN = 5 — $FPR = \frac{3}{8} = 0.375$

$T = 0.65 \rightarrow$ TP = 3, FN = 1 — $TPR = 3/4 = 0.75$ (0.125, 0.75)
FP = 1, TN = 7 — $FPR = 1/8 = 0.125$

$T = 1.0 \rightarrow$ TP = 0, FN = 4 — $TPR = 0/4 = 0$ (0.0, 0.0)
FP = 0, TN = 8 — $FPR = 0/8 = 0$

ROC curve