IAML CLASS DISCUSSIONS

September 26, 2019

Bayes rule

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \longrightarrow P(c,x)$$

$$P(x) = \sum_c P(x,c)$$

$$\frac{P(c,x)}{P(x)}$$

Example Question.

① Data consisting records of the form (x_1, x_2, x_3, c) where $x_1 \in \{0,1\}$, class label $c \in \{0,1\}$, x_2 real-valued

- How to use Naive Bayes classification model approach for obtaining a probability of class c given x_1 and x_2, stating assumptions.

Show that max. likelihood estimate of $P(c=1)$ is given by the proportion of times the class att. c is 1 in the data.

model. $P(D)$.

$D = \{(x_{1i}, x_{2i}, x_{3i}, c_i)\}$ $i \in 1, \ldots, N_D$ ↑ IID

Assumption: data is (conditioned on the model) is independent and identically distributed for each data item

→ each data item is assumed to be from the data model, and is independent of each other.

Therefore $P(D) = \prod_{i=1}^{N_D} P(x_{1i}, x_{2i}, x_{3i} | c_i) P(c_i)$

$$P(D) = \prod_{i=1}^{N_D} \left[\prod_{j=1}^{3} P(x_{ji}|c_i) \right] P(c_i)$$

Assumption: atts are conditionally idpdt. given c.

Approach (NB algorithm)

① Estimate max. likelihood parameters for probs in the model.

$$P(C \mid X_1, X_2) = \frac{P(X_1 \mid C)\, P(X_2 \mid C)\, P(C)}{P(X_1, X_2)}$$

where $P(X_1, X_2) = P(X_1 \mid C=1)\, P(X_2 \mid C=1)\, P(C=1) + P(X_1 \mid C=0)\, P(X_2 \mid C=0)\, P(C=0)$

② compute $P(C \mid X_1, X_2)$ for test values X_1, X_2.

③ choose highest probability class for the label.

* NB is very good for missing data because of the independence assumption.

② Training data

C

$(1, 0, \sqrt{13.5}, 1)$, $(0, 1, 0, 1)$, $(0, 1, -2, 0)$, $(1, 0, 2, 0)$,
$(1, 1, -\sqrt{13.5}, 1)$.

2. Using NB, what is $P(C=1 \mid X_1=1, X_2=1)$?

$$P(C=1 \mid X_1=1, X_2=1) = \frac{P(X_1=1 \mid C=1)\, P(X_2=1 \mid C=1)\, P(C=1)}{P(X_1=1, X_2=1)}$$

$$P(X_1=1 \mid C=1)\, P(X_2=1 \mid C=1)\, P(C=1) + P(X_1=1 \mid C=0)\, P(X_2=1 \mid C=0)\, P(C=0)$$

$P(C=1) = 3/5$ $\quad P(C=0) = 2/5$

does not have to add up to 1 ☺

$P(X_1=1 \mid C=1) = 2/3$ $\quad P(X_2=1 \mid C=1) = 2/3$

$P(X_1=1 \mid C=0) = 1/2$ $\quad P(X_2=1 \mid C=0) = 1/2$

$$P(C=1 \mid X_1=1, X_2=1) = \frac{2/3 \times 2/3 \times 3/5}{2/3 \times 2/3 \times 3/5 + 1/2 \times 1/2 \times 2/5} = \frac{8}{11} \approx 0.72$$

③ Discuss problems if training data:

(1, 0, $\sqrt{135}$, 1), (1, 1, 0, 1), (0, 1, -2, 0), (1, 0, 2, 0), (1, 1, $-\sqrt{135}$, 1)

Solution?

this item changed from 0 to 1

$P(X_1 = 1 \mid C = 1) = \frac{3}{3}$ and $P(X_1 = 0 \mid C = 1) = 0$ (count is 0)

Have prior to all numbers → add a 1 to all the counts to normalise.

You prefer probabilities that are central, not extreme.

④ How to include X_3 in the model?

X_3 is real valued, so model using Gaussian dist.

$\mu_{X_3} = 0 \quad \sigma_{X_3}^2 = \frac{27}{3} = 9$

$P(X_3 \mid C=1) = N(x; 0, 9)$

$P(X_3 \mid C=0) = N(x; 0, 4)$

$X_3 = 0.$ ($x - \mu = 0$)

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2} \cdot \frac{(x-\mu)^2}{\sigma^2}\right) = \frac{1}{\sqrt{2\pi\sigma^2}}$$

$$P(X_3 = 0 \mid C=1) = \frac{1}{\sqrt{2\pi \cdot 9}} = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{3} = \frac{2}{5} \cdot \frac{1}{3} = \frac{2}{15}$$

⑤ What happens if we have outliers?

- we have probabilities on the 'tail' of the Gaussians
- we end up multiplying all our values with this really small number that's at the tail of the Gaussian
- can affect it greatly → mixing discrete + real

September 30, 2019

Decision Tree - allows splitting things to quadrants
→ linear DB doesn't allow you to do this

Entropy - amount of uncertainty

$$H = -\sum_{node} P(node) \sum_{label} P(label \mid node) \log P(label \mid node)$$

- additive down the tree
  - adding a new node allows us to compute the reduction in entropy locally for each decision.

sensible modelling - entropy on a node on the training set can only reduce when we split.
→ less uncertainty.

October 3, 2019

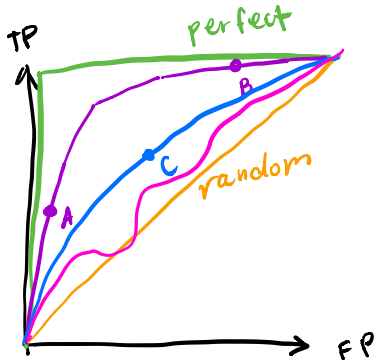
Why is optimising likelihood NOT ALWAYS good idea?
↓ this is the problem

- adding extra parameters to a model ↗ better E_{train}.
  - more parameters = more ways to fit same data
  - you want tighter fit!
- optimisation picks one very good solution
  → may be many good sols.
  → uncertainty in the model
- presence of noise → overfitting, low E_{tr} v high E_{test}

Improving generalisation
- Go Bayesian
- Use very simple models ↗ reduce overfitting
  - but they might not catch some functions we might care about
- Choose models w/ diff. complexities to min. overfitting
  - compare them → cross-validation
- Penalise flexible models, so only use the flexibility if they really have to → regularisation with a parameter
  - simpler model might be better than using complex + regularisation
    → muffling sound vs. turning the volume down

→ can't use max. likelihood as always prefers more complex models.
→ hence cross-validation

ROC curves



a. A/B/C more important in a screening programme for a rare illness over a large population?
↓ positive cases are rare
Low false positive rate.
Find the gradient.

b. Making same bet one one of two options in a prediction market?
A and B

c. Predict whether a safety measure prevents injury or death.
High true positive rate. → B
False positives are okay.
- minor inconvenience.

d. Is the pink line a valid ROC curve?
Yes.
However this is bad.
Norm = have something convex.

Exam Questions.

① Describe a method to regularise NB to prevent overfitting.
Any similar method for DT?

Gaussian NB → add threshold to the variances of the Gaussian
→ not allowing Gs to 'blow up'
→ add a soft thing
→ penalise variances so things are not too large or too small
→ stop things to fit too tight
→ tight variance → good for training
→ terrible for unseen data
→ adding counts (everything on one class, none on other)
e.g. $P(x=1|c=1) = 5/5 = 5/7$ $P(x_1=1|c=0) = 0/5 = 1/7$
→ move everything to the middle just. of extremes.

DT → adding additional counts to nodes
→ pulls things closer to 1/2

Regularation → more data, the more data overwhelms regularisation
→ negligible in infinite data

② cross-val. to choose min. no of dp per leaf in DT
2, 3, .. k folds.
Build a pruned DT.
→ stop when the data items k in a leaf node.

Across folds set k.
Pick the ones with the best average performance.

③ contingency tables

TP = 400 FN = 0
FP = 60 TN = 0

"I've got a big bucket. Everything goes to the positive bucket."

huge overlaps in negatives

TP = 200 FN = 200
FP = 20 TN = 40

low acc. but gives information

more useful
useful in identifying a region of negatives.

October 7, 2019

Linear Regression

Example Question

* consider a linear regression from vectors $\underline{x} \in \mathbb{R}^d$ to scalars y

$$y = \underline{w}^T \underline{\phi}(\underline{x}) + \epsilon \rightarrow \text{Gaussian noise w/ zero } \mu \text{ and } \sigma^2.$$

$\underline{w}$: column vector of parameters

$\underline{\phi}(\underline{x}) = (\phi_1(\underline{x}), \phi_2(\underline{x}) \ldots)^T$
where each ϕ_i is a feature mapping points in $\mathbb{R}^d$ to scalars.

Given training dataset $D = \{(\underline{x}_1, y_1), (\underline{x}_2, y_2) \ldots (\underline{x}_N, y_N)\}$, the loss L

$$L(\underline{w}) = \frac{1}{2} \sum_i (y_i - \underline{w}^T \underline{\phi}(\underline{x}_i))^2.$$

① Show that minimising $L(\underline{w})$ ≡ maximising likelihood, irrespective of the fixed choice for the value σ?

$\epsilon \sim N(0, \sigma^2)$ (zero mean)

$y \sim N(\underline{w}^T \underline{\phi}(\underline{x}), \sigma^2)$

$$p(y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2} \frac{(y_i - \underline{w}^T \underline{\phi}(\underline{x}))^2}{2\sigma^2}\right]$$

What if I have all the datapoints? $x_1 y_1 \dots x_N y_N$

$$P(D_y | D_x) = \prod_i \left[\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - w^T \phi(x_i))^2}{2\sigma^2} \right) \right]$$

$$-\log P(D_y | D_x) = \sum_i \frac{(y_i - w^T \phi(x_i))^2}{2\sigma^2} + \log \sqrt{2\pi\sigma^2}$$

minimise loss = maximise likelihood ← no w here (constant)

$$-\log P(D_y | D_x) = \sum_i (y_i - w^T \phi(x_i))^2 \cdot \frac{1}{2\sigma^2} + \log \sqrt{2\pi\sigma^2}$$

* finding minima is the same under scalar multiplication and addition.

* we don't need to know the noise to find optima!

(2) By computing derivatives show that the parameter vector that minimises the training is given by

$$\underline{w} = (\Phi^T \Phi)^{-1} \Phi^T y$$

where $y = (y_1, y_2, \dots, y_N)^T$ and Φ is the design matrix $\Phi^T = (\phi(\underline{x}_1), \dots, \phi(\underline{x}_N))$.

$$L(w) = \frac{1}{2} \sum_i (y_i - \sum_j w_j \phi_{ij})^2$$

* Φ → row corresponds to different items (i)
column corresponds to the features (j)

Calculate derivatives

$$\partial L / \partial w_k = \frac{1}{2} 2 \sum_i (y_i - \sum_j w_j \phi_{ij}) \cdot \frac{d(y_i - \sum_j w_j \phi_{ij})}{d w_k}$$

$$+ \frac{d(y_k - w_k \phi_{ik})}{d w_k} = \phi_{ik}$$

$$= -\sum_i (y_i - \sum_j w_j \phi_{ij}) (\phi_{ik})$$

↑ to multiply w/ y_i

$$= -\Phi^T y + \Phi^T \Phi \underline{w}$$

set derivatives to 0

$$\Phi^T y = \Phi^T \Phi \underline{w}$$

↳ square matrix (assumption)

$$w = (\Phi^T \Phi)^{-1} \Phi^T y$$

Hence proved!

③ Assumptions about the rel. between M and N using

$$\underline{w} = (\Phi^T \Phi)^{-1} \Phi^T y$$

$\Phi^T \Phi$ would be a square.

N would be large.
- more data the better!

④ Is LR sensitive to outliers?

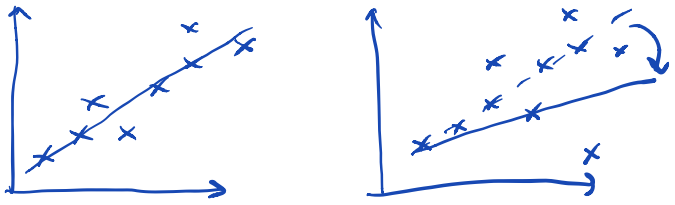
value that is far from the line → dramatic effect on the reg'n. optimisation of $E(\underline{w})$ v sensitive to values on the tail of G. line deviates dramatically.

Why?

$$E(\underline{w}) = \frac{1}{2} \sum_i (y_i - w^T \phi(x_i))^2$$

squared of the distance.

Result: adjust whole line to compensate.



⑤ Model accident rate on a road as a fn of traffic density, outside air temp, date, time.

Discuss what features ϕ you might choose and why.

- More cars → more accidents, even tho acc. rate constant
- speeds go down with high density
- temperature → special value at 0 C (frozen)
- date/time → rush hour, weekends vs. weekdays

October 10, 2019

Logistic Regression

Generative vs. conditional

Naive Bayes: $P(c)\,P(\underline{x}|c) \rightarrow P(c|\underline{x})$

Log. regr: $P(\underline{x})\,P(c|\underline{x}) \rightarrow P(c|\underline{x})$ → smooth, infinitely differentiable

→ implicitly has understanding of $P(\underline{x})$.

→ benefits: map $\underline{x} \rightarrow c$ is high dim to low dim.

→ easier to build a flexible map.

* Either is fine when modeling a stationary system.
  If things might change, consider casual structure. → Robust!
  - Need to know why things change

Logistic Regression

- Lin Reg + Log. Trick
  ↓
  push linear outputs through a squashing function to get probs.

- Binary case $P(c=1|\underline{x}) = \sigma(f(\underline{x}))$

- $\sigma(x) = \dfrac{\exp(x)}{1+\exp(x)} = \dfrac{1}{1+\exp(-x)}$

- Consequence $P(c=0|\underline{x}) = 1-\sigma(f(\underline{x}))$
  → no closed form solution
  → instead convex optimization: optimal solution exists

Loss

$$P(c|\underline{x}) = [\sigma(\underline{w}^T\underline{x}+b)]^c \times [1-\sigma(\underline{w}^T\underline{x}+b)]^{1-c}$$

$$L(\underline{w}, b) = -\log P(c|\underline{x}) = -c\log\sigma(\underline{w}^T\underline{x}+b) - (1-c)\log(1-\sigma(\underline{w}^T\underline{x}+b))$$

↳ individual single data item loss

* Important info $\partial\sigma/\partial x\ \sigma(x) = \sigma(x)(1-\sigma(x))$

Derivatives

$$L(\underline{w}, b) = \sum_i -c_i\log\sigma(\underline{w}^T\underline{x}_i+b) - (1-c_i)\log(1-\sigma(\underline{w}^T\underline{x}_i+b))$$

↑ negative log probability of the whole data

* maximise log probability, minimise loss (⊖ log prob)
  → go down hill, not uphill.

$$\partial L(\underline{w},b)/\partial w_j = \sum_i -c_i \left[\frac{\sigma(\underline{w}^T\underline{x}_i+b)(1-\sigma(\underline{w}^T\underline{x}_i+b))}{\sigma(\underline{w}^T\underline{x}_i+b)} x_{ij} \right] - (1-c_i)\left[\frac{-\sigma(\underline{w}^T\underline{x}_i+b)(1-\sigma(\underline{w}^T\underline{x}_i+b))}{(1-\sigma(\underline{w}^T\underline{x}_i+b))} x_{ij} \right]$$

$$= -\sum_i c_i(1-\sigma(\underline{w}^T\underline{x}_i+b))x_{ij} + (1-c_i)\sigma(\underline{w}^T\underline{x}_i+b)x_{ij}.$$

$$= \sum_i (c_i - \sigma(\underline{w}^T\underline{x}_i+b))x_{ij}$$

Multivariate distributions

$$P(y=k|\underline{x}) = \text{softmax}(f) = \frac{\exp(f_k(\underline{x}))}{\sum_{k'}\exp(f_{k'}(\underline{x}))}$$ (↑ output for each class; → to normalize things) [0, 1]

- Log. Reg. can be used for more than 1 class.
  - ↳ Logistic sigmoid $\sigma(z) = \frac{1}{1+\exp(-z)}$

Binomial and Multinomial
- replace bernoulli and multivariate w/ binomial and multinomial

Features
- can replace $\underline{x}$ w/ features $\phi(\underline{x})$
- compute features ahead of time
- there is curse of dimensionality

Radial Features
- ↳ distance from some reference points
- methods – distance from key locations
  - use each data as radial feature – non param.
    - grows when data grows

⊕ very flexible
⊖ scaling effects

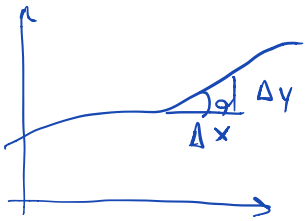
- things can only happen locally
  - rent/buy w.r.t income.
  - kind of nearest neighbor
  - RBF confused with really different data
    - extremes should just be similar to normal data.

October 14, 2019

Optimisation

Generic optimisation

$y = f(x)$



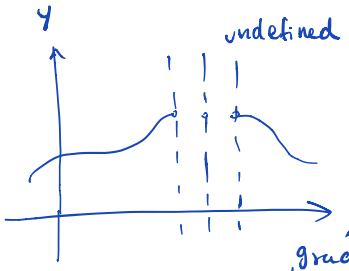
$\alpha = \tan\left(\frac{\Delta y}{\Delta x}\right)$

if $\Delta y \oplus$ and $\Delta x \oplus$, increasing fn.
$\Delta y \ominus$ and $\Delta x \ominus$, decreasing fn.

$\frac{\Delta y}{\Delta x} = \tan(x)$

$\lim_{\Delta x \to 0} \frac{\Delta y}{\Delta y} = \frac{\partial f}{\partial x} = f'(x)$

tells us how fast we are incr/dec.



No slope at undefined regions
→ can trip up optimization

Generalise → gradient $\nabla f = \left\langle \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots \frac{\partial f}{\partial x_N} \right\rangle^T$

$f \equiv (E(\underline{w}))$ → error of the weights

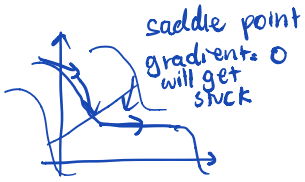
step towards the decrease of that error.

$w^{t+1} = w^t - \eta \nabla E(\underline{w})$ *greedy

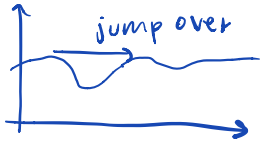
↳ learning rate

$E(\underline{w})_{MSE} = \frac{1}{n} \sum_i (y_i - f(x_i))^2$ ⟶ convex!

Problem 1



Problem 2



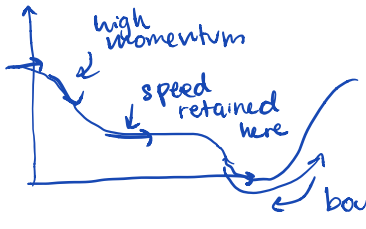
$w^{t+1} = w^t - \eta \nabla E(w^t)$

↓ step size too high
'skip' over global minimum

Momentum

$$z^{t+1} = \beta \cdot z^t + \nabla E(w_t)$$

$$w^{t+1} = w^t + \alpha z^{t+1}$$



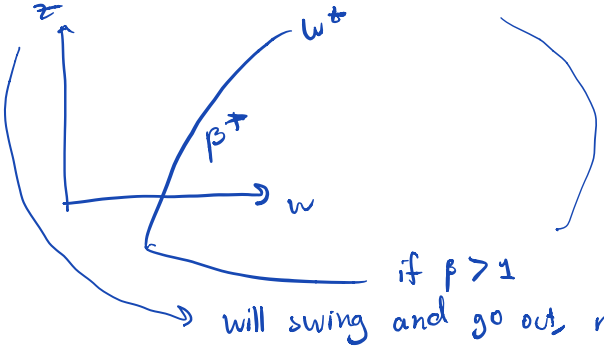
* accumulates gradient over time.

$$z^{t+1} = \underbrace{\beta z^t}_{\text{damping}} + \nabla E(w_t)$$

velocity

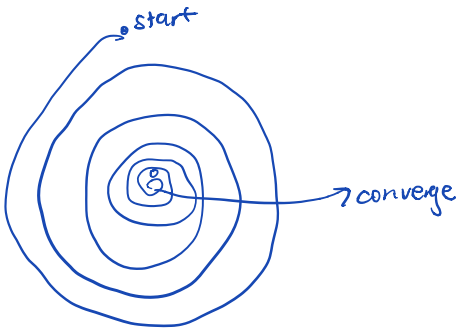
$$w^{t+1} = w^t + \alpha z^{t+1}$$

time step



if $\beta > 1$

will swing and go out, rotate and then converge

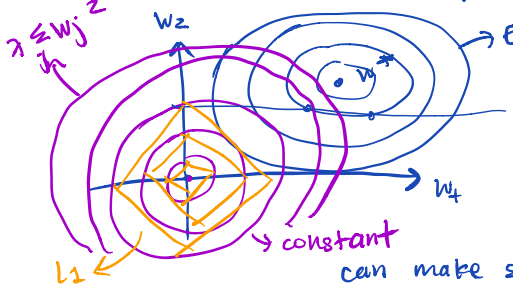


Regularisation

$$E(w) = \frac{1}{n} \sum_i (y_i - f(x_i; w))^2$$

$$E'(w) = E(w) + \lambda \sum_j w_j^2$$

* E' big when w_j big



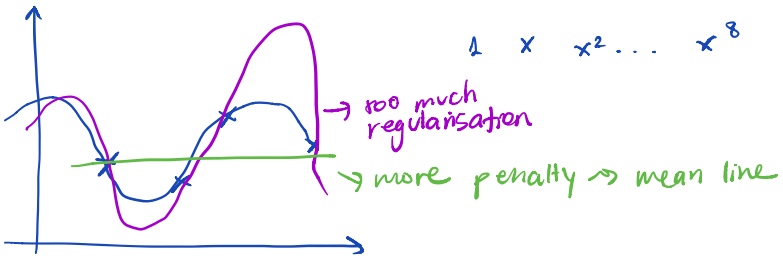
$$L_1 = E'(w) = E(w) + \lambda \sum_j |w_j|$$

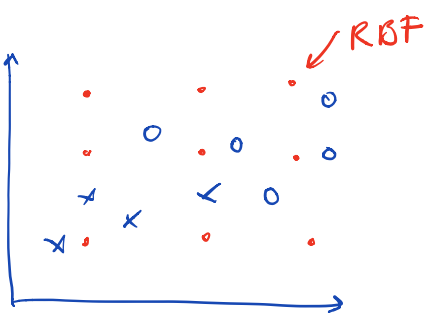
can make second loss as little as possible

* Along w_1, the value of the error does not change much, hence L_2 can be large (smaller w_1)

* Along w_2, $E(w)$ varies more, hence less L_2

Regularisation - add info in order to solve an ill-posed problem or to prevent overfitting.
↓ not enough data to solve.
- less freedom in parameters

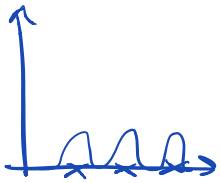




$e^{-(x-c)^2/h^2}$

* Reg. penalises high weights on RBF, make DB smooth → close to a straight line.

If we dont regularise on RBF → you get 'islands'



October 21, 2019

SVM

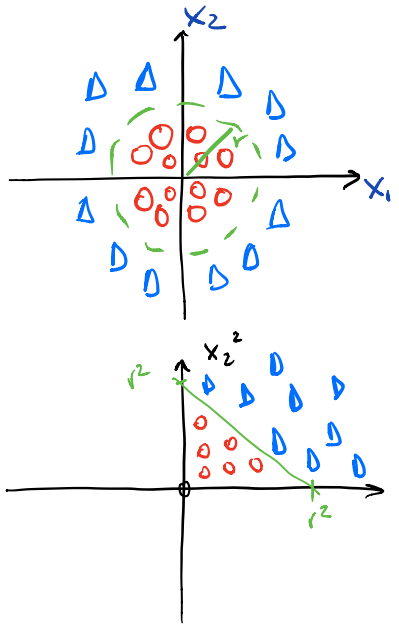
Q1. maximising $1/\|w\|$ $\propto$ minimising $\|w\|^2$.
Why is this minimising the squared of $\|w\|$ instead of just $\|w\|$ itself?

This is solving for αs in $\underline{w} = \sum_i \alpha_i y_i \underline{x}_i$
solution is quadratic programming problem.
But we know that it is $\propto$ minimising $\|w\|^2$.

Q2. Projection of x onto w
$w^T x$.
↳ assume w is unit vector
w/o assumption → $\frac{1}{\|w\|} w^T x$

Q3. How can we deal w/ infinite dimensions because of kernel trick?
* point of kernel trick
- all we have to compute is $x_i \cdot x$
↳ new test point
- $\phi(x_i) \cdot \phi(x_j) = K(x_i, x_j)$
- only ever need these dot products
- we can operate in kernel space.

Kernel trick



Are these linearly separable? No.

$K(x_i, x_j) = \exp(|x_i - x_j|^2 / \sigma^2)$

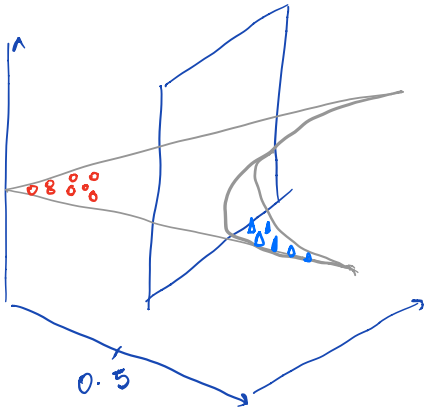
To separate, draw a circle.
$x_1^2 + x_2^2 = r^2$.

New feature space, (x_1^2, x_2^2).
$\underline{x} \rightarrow \begin{pmatrix} x_1^2 \\ x_2^2 \end{pmatrix}$

Example of a kernel for 2D input space

$$\phi(x_i) = \begin{pmatrix} x_{i,1}^2 \\ \sqrt{2}\, x_{i,1} x_{i,2} \\ x_{i,2}^2 \end{pmatrix} \quad \text{then} \quad K(\underline{x}_i, \underline{x}_j) = (\underline{x}_i^T \underline{x}_j)^2$$

How would this transformation look like?



* makes the data linearly separable.

SVM vs. Logistic Regression

SVM optimization

$$\min. \quad \|\underline{w}\|^2 + C \sum_i \xi_i$$

subject to $y_i f(\underline{x}_i) \geq 1 - \xi_i, \; i = 1, \ldots, n.$

where $\xi_i \geq 0$ for all i and $f(\underline{x}_i) = \underline{w}^T \underline{x}_i + w_0$

The optimization problem can be re-written as

$$\min_{\underline{w} \in \mathbb{R}^d} \|\underline{w}\|^2 + C \sum_i \underbrace{\max(0, 1 - y_i f(\underline{x}_i))}_{g_{hinge}}$$

$y_i f_i > 1 \Rightarrow 1 - y_i f_i < 0$

* $g_{hinge} = \max(0, 1 - y_i f_i) \rightarrow g_{hinge} = 0$

$y_i f_i = 1 \geq 1 - y_i f_i = 0 \rightarrow 1 - 1 = 0 \rightarrow g_{hinge} = 0$

$y_i f_i < 1 \rightarrow 1 - y_i f_i > 0 \rightarrow g_{hinge} = 1 - y_i f_i.$

Logistic regr. w/ ridge penalty

$$p(y = +1 \mid \underline{x}) = \frac{1}{1 + e^{-f(x)}} \qquad p(y = -1 \mid \underline{x}) = \frac{1}{1 + e^{f(x)}}$$

Combining these we have

$$p(y_i|x_i) = \frac{1}{1+e^{-y f(x_i)}}$$

Loss $\rightarrow -\log(p(y_i|x_i)) = \log(1+e^{-y_i f(x_i)})$

$$\min_{w \in \mathbb{R}^d} \|w\|^2 + C \sum_i \log(1+e^{-y_i f(x_i)})$$

$\|w\|^2$: squared penalty; $C = 1/\lambda$

Define $g_\sigma(z) = \log(1+e^{-z})$

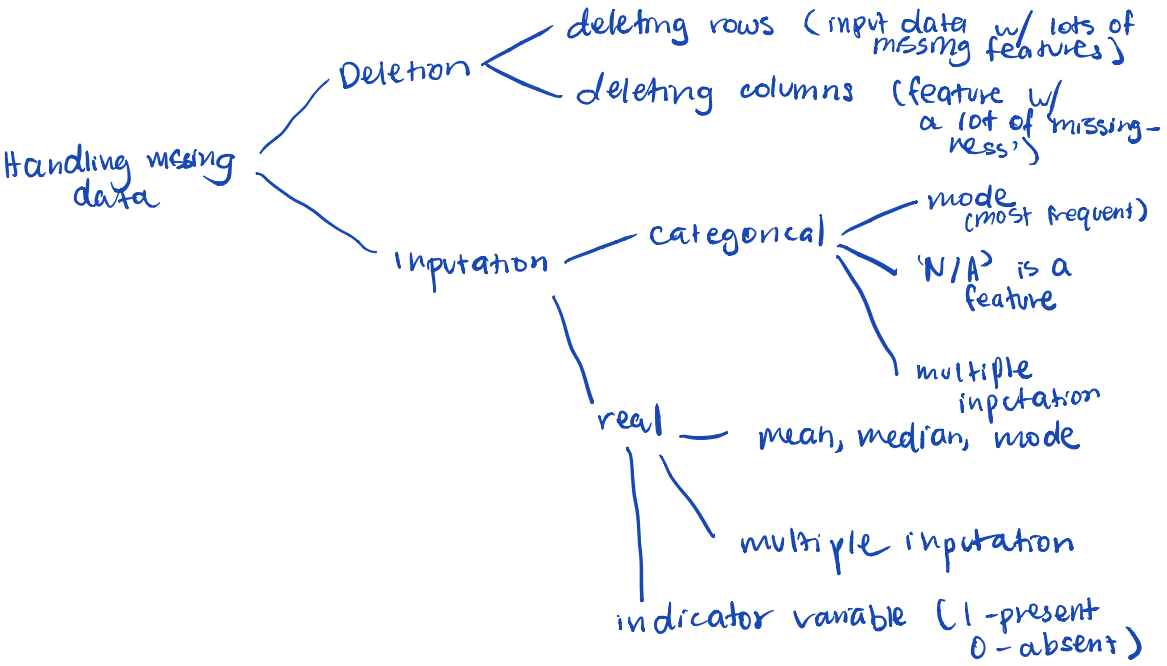
SVM has a similar op. prob, but with $g_{hinge}(z)$

* Both are convex
* use g_σ is not sparse, but log regr. gives probability output

October 24, 2019

Merits of distance measure
- validation (hyperparameter = distance)

Missing values
- understand WHY the data can be missing



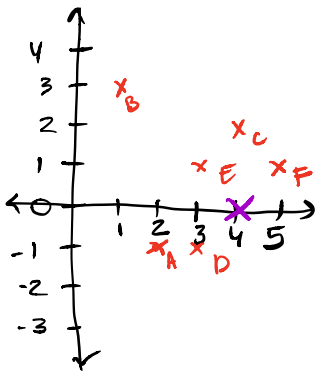
K-Nearest Neighbors
- Voronoi tesselation
  - partitions the space that surrounds each dp.
- Odd k → tie-breaker
- Increase k → smooth out decision surface.

Example Question

① Consider the ff. dataset.

A (2, -1, -2) B (1, 3, -3) C (4, 2, 0)
D: (3, -1, -1) E (3, 1, -1) F (5, 1, 1)

Use KNN to predict x_3 from $x_1 = 4$, $x_2 = 0$. Assume k=3



k=3 → NN = D, E, F

$\hat{x}_3 = \frac{-1 + -1 + 1}{3} = -\frac{1}{3}$

Introduce new point G (10, 1, -1000)
How would the outlier affect the preds of lin. reg (fit a plane) and kNN reg.
Be specific about which inputs would be affected.

- With LR, the entire plane would be pushed because of G's value
  - plane tries to go as close to ALL
- With kNN, affects points that are close to G.
  - if G is a neighbor → x_3 value will be near G's x_3
  - local effect

Suppose H (5, 1, 0) and k=3.
- you have 4 NNs!
  - add all, divide by two
  - noise: diff y values, same x value.

Can you do with integers? Yes. use mode instead of average.

Kernels in KNN

$$p(y=1|x) = \frac{\sum_{i=1}^{n} 1(y_i=1) K(x_i, x)}{\sum_{i=1}^{n} K(x_i, x)}$$

Parzen windows

$\sum_{i=1}^{n} w_i = 1$

$\rightarrow 1 \geq w_i \geq 0$ weight on the datapoint

* points which are nearby has more weight

SVM

$$\hat{y}(x) = \text{sign}\left[\sum_i y_i \alpha_i K(x, x_i) + w_0\right]$$ dB

↳ uses α to define support vectors

Parzen

$$p(y=-1|x) = \frac{\sum_{i=1}^{n} 1(y_i=-1) K(x, x_i)}{\sum_{i=1}^{n} K(x, x_i)}$$

decision boundary

$$p(y=1|x) = p(y=-1|x) = 0.5$$

$$\sum_{i=-1}^{n} 1(y_i=1) K(x, x_i) = \sum_{i=1}^{n} 2(y_i=-1) K(x, x_i)$$

$$= \boxed{\sum_{i=1}^{n} y_i K(x, x_i) = 0.}$$ dB

October 28, 2019

Intrinsic Evaluation

- look at pairs
  - should they be in the same group?

Rand Index = $\frac{TP + TN}{TP + TN + FP + FN}$ (Accuracy)

- Overlapping clusters → use AMI
  → very rare case
- use table → how good is this assignment?
- AMI - information one variable has on the other?

	Y_1	Y_2	Y_3
X_1			X
X_2	X		
X_3		X	

→ perfect information

Example

$\{-4, -3, -2, 0, 3, 5, 8\}$.

Run k-means. $K = 2$, $\mu_1 = -6$ $\mu_2 = 5$

① $c_1 = \{-4, -3, -2\}$ $\mu_1' = \frac{-4-3-2}{3} = -3$

$c_2 = \{0, 3, 5, 8\}$ $\mu_2' = \frac{0+3+5+8}{4} = 4$

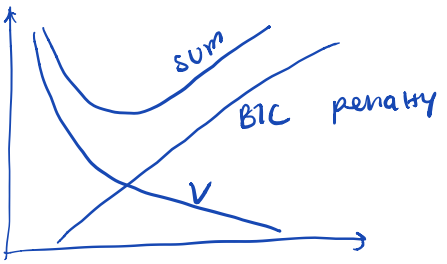
② $c_1 = \{-4, -3, -2, 0\} = \mu_1' = \frac{-4-3-2+0}{4} = -2.25$

$c_2 = \{0, 3.5, 8\}$ $\mu_2' = \frac{3+5+8}{3} = 5.33$

③ updates do not change!

Choosing the value of K
- cross-validation with hyperparameters
- optimise V for $k = 2, 3 \ldots$

$$V = \sum_j \sum_{x_i \to c_j} D(c_j, x_i)^2$$



* BIC – Bayesian Information Criterion

* penalise no. of cluster centres

$BIC = \max\left(L - \frac{p}{2} \lg n\right)$

$V = -cL$

$\max \longrightarrow -\frac{V}{c} - \frac{f}{2} \log n$

$\min \longrightarrow \frac{V}{c} + \frac{p}{2} \log n$ → no. of parameter kD

October 31, 2019

Q: Prior Probability for EM
We use P(b|x) to label the data instance. By using Bayes rules in the second line, we need to know prior prob. How can you get prior prob. if we DK which class the instances belong to?
Randomly estimate / plot?

$$b_i = P(b|x_i) = \frac{P(x_i|b)\,P(b)}{P(x_i|b)\,P(b) + P(x_i|a)\,P(a)}$$

what is the initial P(b)?

We need to have an initial setting (guess)
$P(a) = 0.5 \quad P(b) = 0.5$

then they get updated
$P(b) = (b_1 + b_2 + \ldots + b_n)/n$
$P(a) = 1 - P(b)$ * total law of probs → sum up to 1.

* $P(a|x_i) + P(b|x_i) = 1$.

We can choose to re-estimate after each iteration.
- we might want to fix variance (σ)

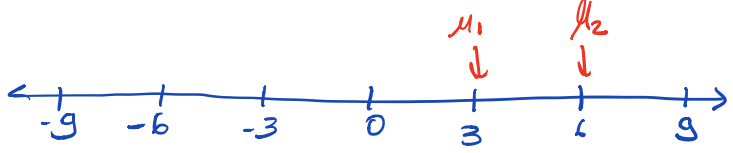
A Gaussian MM with K components

$$p(x) = \sum_{k=1}^{K} p_k \times p(x|k)$$

$$p(x|k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left\{-\frac{(x-\mu_k)^2}{2\sigma_k^2}\right\}$$

Consider dataset {-9, -6, -3, +3, +6, +9}.
Initial means = $\mu_1 = 3$, $\mu_2 = 6$ $\sigma_1 = \sigma_2 = 1$. Assume priors = 1/2

① Compute post. probs of the two Gaussians for the obs. $x = 6$. Report the results to 2 DP.



$$p(x=6 \mid k=1) = \frac{1}{\sqrt{2\pi \cdot 1}} \exp\left(-\frac{(6-3)^2}{2 \cdot 1}\right) = \frac{1}{\sqrt{2\pi}} e^{-9/2}$$

$$= 0.0044$$

$$p(x=6 \mid k=2) = \frac{1}{\sqrt{2\pi \cdot 1}} \exp\left(-\frac{(6-6)^2}{2 \cdot 1}\right) = \frac{1}{\sqrt{2\pi}} = 0.3990$$

$$p(k=1 \mid x=6) = \frac{p(k=1)\,p(x=6 \mid k=1)}{p(k=1)\,p(x=6 \mid k=1) + p(k=2)\,p(x=6 \mid k=2)}$$

$$= \frac{1/2 \times 0.0044}{1/2 \times 0.0044 + 1/2 \times 0.3990} = 0.01$$

$$p(k=2 \mid x=6) = 1 - 0.01 = 0.99$$

* Think how fast the Gaussian decays.

② Estimate the posterior probability of the 1st Gaussian for every point in ds.

$p(k=1 \mid x=-9) = 1.00$

$p(k=1 \mid x=-6) = 1.00$ ↖ on the left

$p(k=1 \mid x=-3) = 1.00$

$p(k=1 \mid x=3) = 0.99$

$p(k=1 \mid x=6) = 0.01$ ↘ on the right

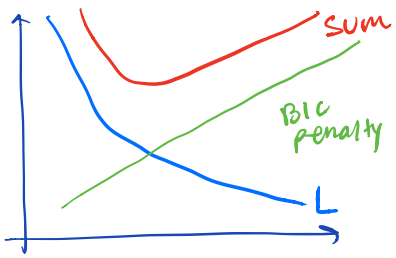
$p(k=1 \mid x=9) = 0.00$

* Gaussian decay → squared of the distance

How to pick k → Occam's razor

BIC: minimize $-L + \frac{p}{2} \log n$.

As k increase, log likelihood increase.

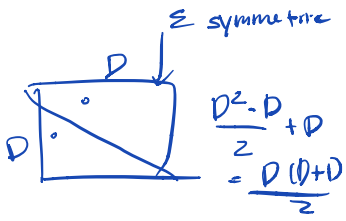


means μ_k ⟶ KD parameters

covariance matrices Σ_k ⟶ $K\frac{(D+1)}{2}$

mixing proportions p_k ⟶ $K-1$ for each param

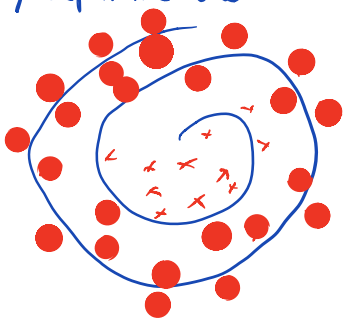
K Gaussian in D dimensions



Real-world applications
- real-valued vector data
  - handwritten digits, normalized 0,1.

November 4, 2019

Q1. Dimensionality reduction can convert non-linearly separable problem into a linearly separable one?



* if we unfold the spiral, x one side • one side

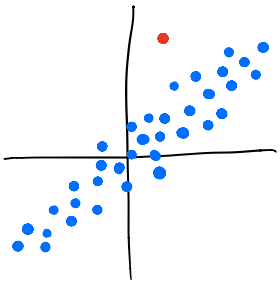
* non-linear dim. red.

Q2: $\Sigma e = \lambda e$ ← 'length'

gives you back same vector, but scaled

if $|e|^2 = 1$, then λe has length λ.

Q3. Sensitivity to outliers is not a problem for PCA



covariance matrix will be affected by outliers

* SOME effect on the resulting principal component.

Q4: Lagrange Multiplier formulation from slides
Eigenvector = Direction of max variance

$$V = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{d} x_{ij} e_j \right)^2 - \lambda \left(\sum_{j=1}^{d} e_j^2 - 1 \right)$$

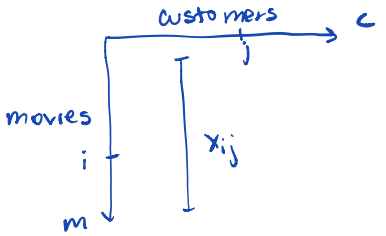
In the Lagrange multipler part of the eqn ↙ is the length of the vector.

Why is there no square root?

$|e_j|$ is unit vector, so $\sqrt{1^2} = 1$
* easier to differentiate → 'convenience'

Example Question

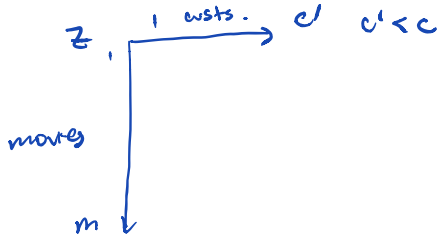
① NovieFlix - recommend movies to customers



company wants to keep the privacy of the customers.
They run PCA on X to get Z

How would you reduce the dimensionality?

* Reduce dimensionality of the customers.
Z → same rows, fewer colums

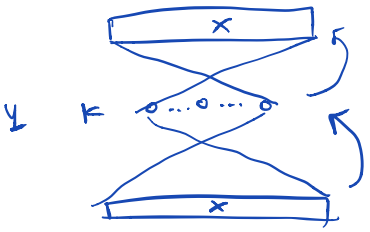


* Take those diff. vectors. do PCA on them, pick eigenvectors.

* PCA - not taking out colums
- each colunth is some combination of the customers

* Movies data is sparse
- PCA on missing data → it is handled
- good on data that has a lot of 'missingness'
- low-rank decomposition

Auto-encoder



$K < D$

decode to original

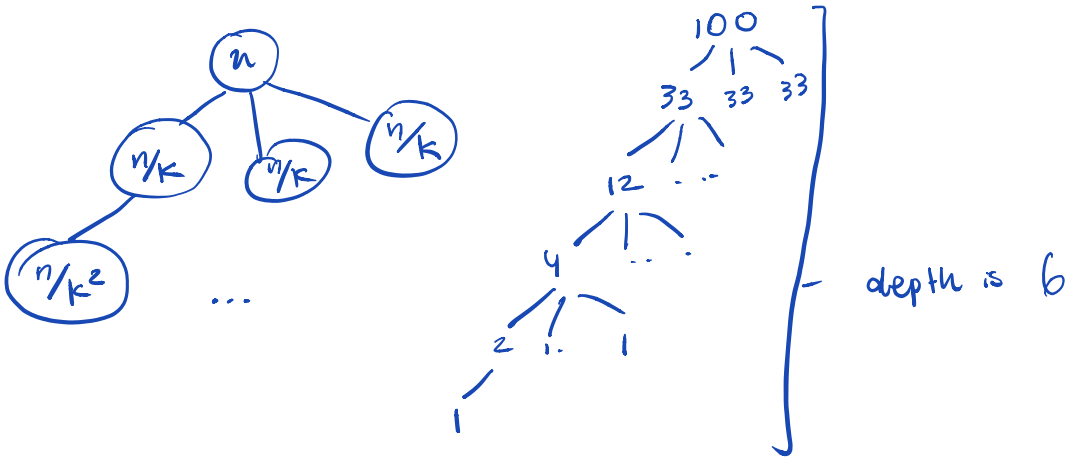
encode from hi-D

$v_j = x \cdot e_j$

+use principal components or anything thats similar

November 11, 2019

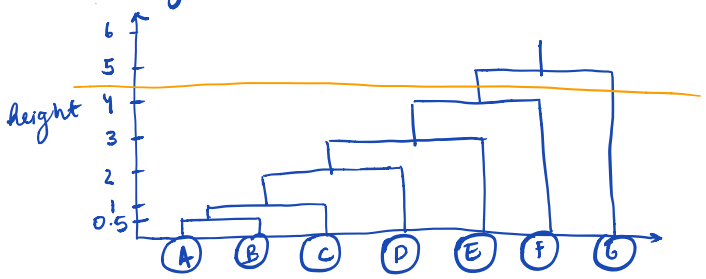
Q1. How many levels are there in a tree created by top-down divisive clustering with a ds of 100 entries, setting K=3.
→ clusters may not be of equal size



Agglomerative clustering

Dataset = {-3.5, -3, -2, 0, 3, 7, 12}
A B C D E F G

a. single link



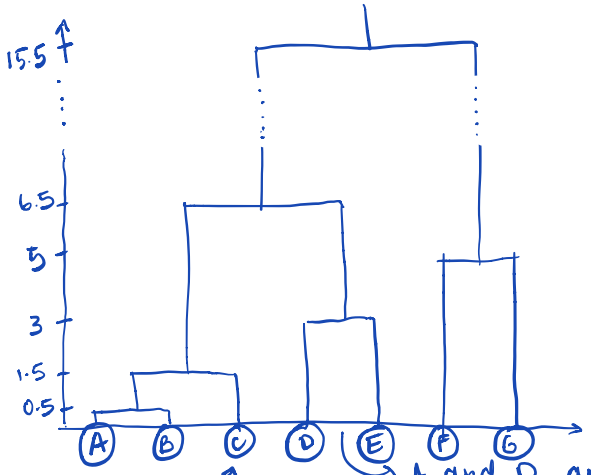
cut here if we want to have 2 clusters:

$C_1 = \{A, B, C, D, E, F\}$

$C_2 = \{G\}$

single link makes string out clusters

b. Complete link



* makes more spherical clusters

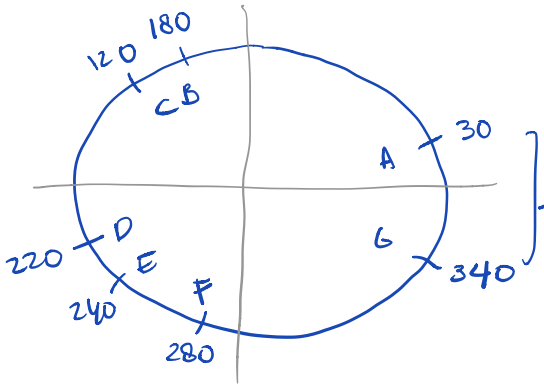
A and D are 3.5 apart! so D,E closer

a and c are at a dist 15 away

* nearest outside of the cluster to the farthest inside cluster

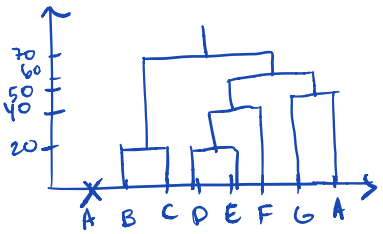
Example Question

Dataset in degrees = {30, 100, 120, 220, 240, 280, 340}



arithmetic modulo 360

$|340 - (360 + 30)| = 50$



single-link

November 18, 2019

Rosenblatt's rule for training perceptrons

$$\hat{y} = \text{sign}(\underline{w}^T \underline{x}) \begin{cases} 1 \\ -1 \end{cases}$$

Perceptron Learning Rule

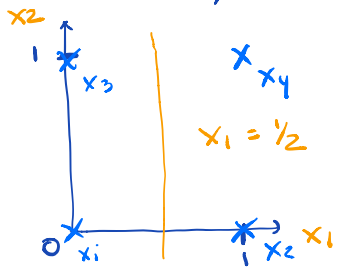
repeat
  for i in $1, 2, \ldots, n$
    $\hat{y}_i \leftarrow \text{sign}(\underline{w}^T \underline{x}_i)$ $\begin{cases} 1 \text{ if } \geq 0 \\ -1 \text{ if } < 0 \end{cases}$
    if $\hat{y}_i \neq y_i$
      $\underline{w} \leftarrow \underline{w} + y_i \underline{x}_i$ ← 'update', dependent on true label
until all training examples correctly classified
  * if no separating hyperplane, will run forever.

Example Question

Data:

	x_0	x_1	x_2	y
x_1^T	1	0	0	-1
x_2^T	1	1	0	1
x_3^T	1	0	1	-1
x_4^T	1	1	1	1

- Initialise weight vector $\underline{w}^T = (0, 0, 0)$
- Run PLR on dataset to convergence. What is final $\underline{w}$?
- Consider the dp $(\underline{x}_1, \underline{x}_2, \underline{x}_3, \underline{x}_4)$ in order and repeat if not ALL correctly classified.



P.L.R.
$\hat{y}_i = \text{sign}[w^T x_i]$
if $\hat{y}_i \neq y_i$
$\underline{w} \leftarrow \underline{w} + y_i \underline{x}_i$

First update
$\text{sign}(z) = \Theta(z)$

$\Theta(w \cdot x_1) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \Theta(0) = 1 \neq y_1 = -1$

$w \leftarrow \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = (-1, 0, 0)$

$\Theta(w \cdot x_2) = \Theta(-1) \neq y_2 = 1$

$w \leftarrow \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = (0, 1, 0)$

$\theta(w \cdot x_3) = \theta(0) = 1 \neq y_3 = -1$

$w \leftarrow \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}$

$\theta(w \cdot x_4) = \theta(-1) = -1 \neq y_4 = 1$

$w \leftarrow \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = (0, 2, 0)$

Now run through all of it again (second update)

$\theta(\underline{w} \cdot x_1) = \theta(0) = 1 \neq y_1 = -1$

$w \leftarrow (0, 2, 0) - (1, 0, 0) = (-1, 2, 0)$

$\theta(\underline{w} \cdot x_2) = \theta(1) = 1 = y_2$ correct

$\theta(\underline{w} \cdot x_3) = \theta(-1) = -1 = y_3$ correct

$\theta(\underline{w} \cdot x_4) = \theta(1) = 1 = y_4$ correct

Finally check x_1

$\theta(w \cdot x_1) = \theta(-1) = -1 = y_1 = -1$ correct.

correct $\underline{w} = (-1, 2, 0)$

PB → $\underline{w} \cdot x = 0$

$\begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = -1 + 2x_1 = 0$ ← BIAS!

$x_1 = 1/2$