(U19CS012) CS365 Machine Learning (U19CS012)

(BHAGYA RANA) Part A

U19CS012 [BHAGYA VINOD RANA]

U19_CS365-A-BS.pdf

1.> minimizing the Risk ~ maximizing the posterier probability

[classification problem modelled using gaussian]

1.>

① An accepted low risk customer → increases profit
A rejected high risk customer → decreases loss

② Loss for high risk customer erroneously accepted is different from the gain for an erroneously rejected low risk customer.

③ $x_1 \to$ income & $x_2 \to$ saving
det decide outcome

$c = 1$ high risk & $c = 0$ low risk

④ For any new data $X_1 = x_1$ and $X_2 = x_2$ knowing $P(c \mid x_1, x_2)$ we can choose

ⓐ $c = 1$, if $P(c=1 \mid x_1, x_2) > 0.5$ and $c = 0$ otherwise

ⓑ $c = 1$, if $P(c=1 \mid x_1, x_2) > P(c=0 \mid x_1, x_2)$ and $c=0$ otherwise

⑤ Probability of error = $1 - \underbrace{\max\left(P(c=1 \mid x_1, x_2), P(c=0 \mid x_1, x_2) \right)}_{\text{maximizing the posterier probability}}$.

Therefore, we are minimizing the risk by maximizing the ~~pro~~ posterier probability

(U19CS012

(BHAGYA RANA)

(Q2)

Sport Video Summarization

(Feature Based)

Object

① Target objects can be ball, player, umpire, field, audience,

② Object segmentation (computationally expensive)
   (Need to seperate object from environment)

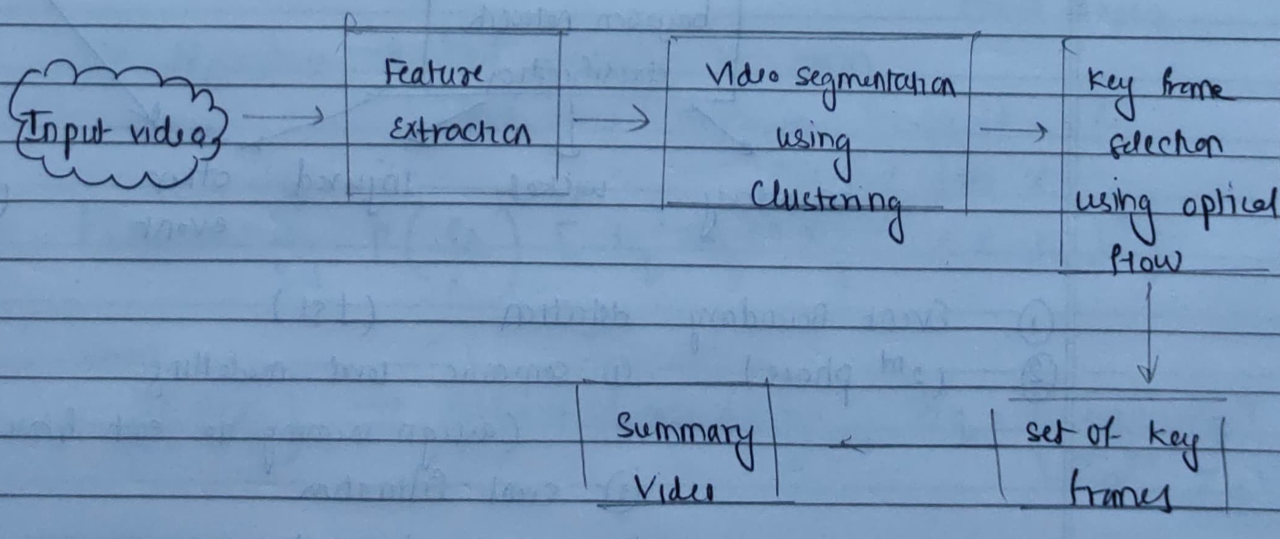
* Aim: Efficient video summary generation using dominant features for cricket.

Ⓐ Color & Change in Motion

(i) partition video into shots using dominant color feature

(ii) select key frames from each shot using change in motion

& Dominant color = Green field in cricket



Ⓑ ~~Fast~~ Event identification & state based event modelling

Sol^n:

⟹ Semantic Based analysis for cricket match video summarization

vision

U19CS012

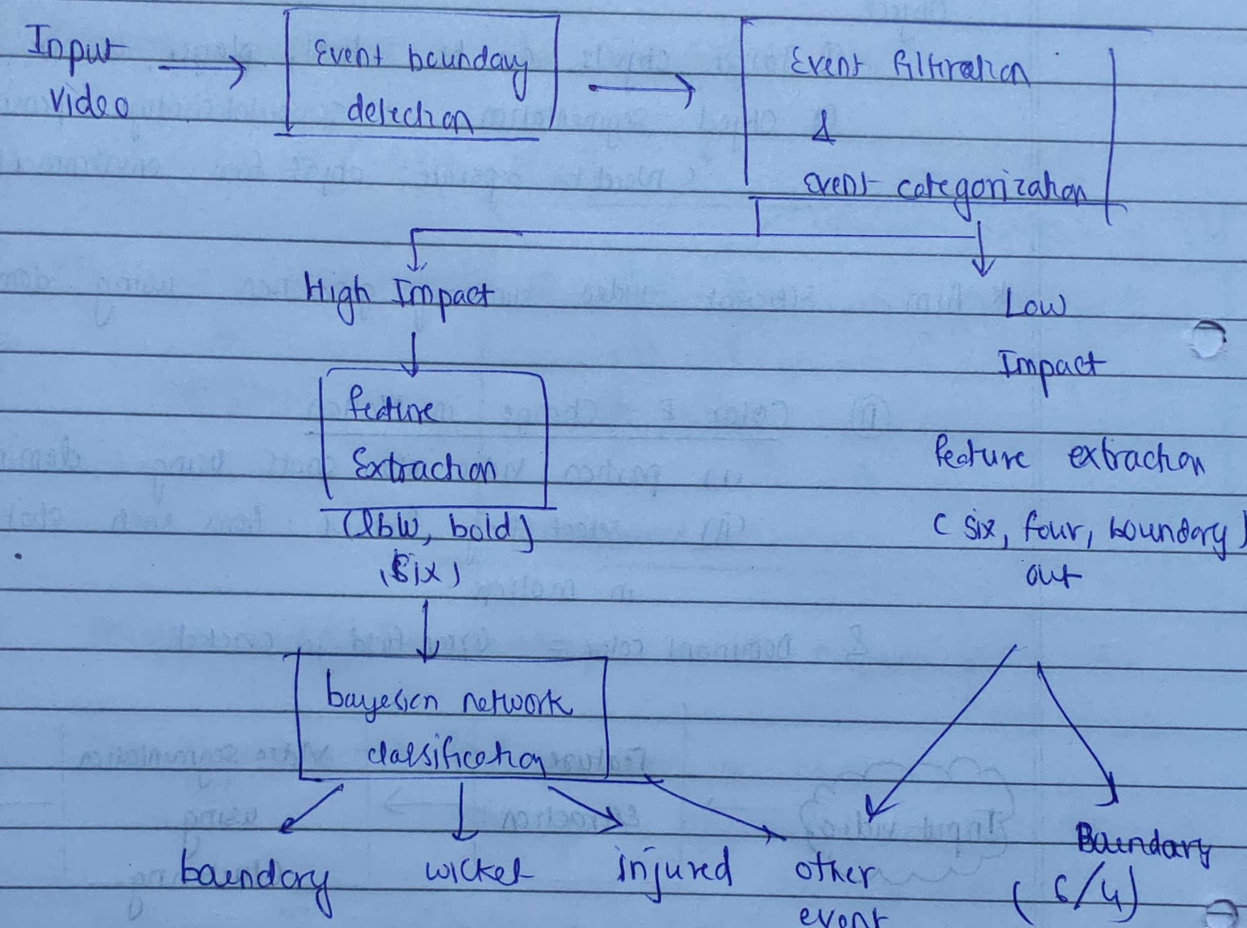
BHAGYA RANA

Hybrid Model

Supervised Training + Unsupervised

(Train it using old season of cricket match)

(Event = set of frames)



① Event Boundary detection (1st)

② (2nd phase) ① semantic level modelling

(assign meaning to each frame)

② event filtration

③ event categorization

④ Event Classification → Hidden Markov model

③ Out / ~~Reset~~ Injured

U19CS012

BHAGYA RANA

(Q3)

(GMM based modelling) $P(X/\theta) = \sum_{K=1}^{K} \pi_k N(x|\mu_k, \Sigma_k)$

(Gaussian mixture Models) - Density Based model)

(A) Different parameters to be estimated = D

distribution $N(x|\mu_k, \Sigma_k)$

$\Longrightarrow$ parameters of GMM $\rightarrow$ $\boxed{\theta : \textcircled{1} \{ \pi_k, \quad \textcircled{2}\ \mu_k, \quad \textcircled{3}\ \Sigma_k \quad : k = 1, \ldots k \}}$

(mixture weights π_k)

(k means μ_k)

(covariances)

(B) Design strategy for deciding no. of mixture models for facial images

EM Algorithm $\rightarrow$ find parameter in GMM that gives maximum likelihood of observed data X

$$F(x) = \sum_{q=1}^{n} P\left(\frac{c_q}{x}\right) \left[\mu_q^{Y} \Sigma_q^{YX} \Sigma_q^{XX^{-1}} (x - \mu_q)^{x} \right]$$

where

$P(c_q/x) \longrightarrow$ conditional probability

$$\Sigma_q = \begin{bmatrix} \Sigma_q^{XX} & \Sigma_q^{XY} \\ \Sigma_q^{YX} & \Sigma_q^{YY} \end{bmatrix}$$

$m \rightarrow$ GMM mixtures in facial conversion

$$\mu_q = \begin{bmatrix} \mu_q^{X} \\ \mu_q^{Y} \end{bmatrix}$$

(★) GMM $\rightarrow$ Joint density Model is baseline for mapping source & target faces

(Cepstrum based GMM) ✓

vision

U19CS012

BHAGYA RANA

(Q4)

Incomplete data problem

① ⇒ Here ML estimation is made difficult by the absence of some part of data in more familiar and simple data structure

② Solⁿ = Expectation maximization
1 iteration of EM algorithm
① Expectation Step (E-step)
② Maximization step (M-step)

③ closely related to ad hoc approach to estimation with missing data
→ where the parameters are estimated by filling in initial values for the missing data.
→ the latter are then updated by their predicted values using these initial parameters
→ the parameters are then re-estimated and so on, proceeding iteratively until convergence.

④ Main idea of EM Algorithm = Associate with given incomplete-data, a complete data problem for which ML estimation is computationally more tractable
(reformulating the problem in terms of easily solvable data problem)

Algorithm →

U19CS012

BHAGYA RANA

(EM Algorithm)

1. E-step = manufacturing data for the complete data problem using observed data set of incomplete data problem & current value of parameters, so that

2. the simpler M-step computation can be applied to "completed" data set

3. ⇒ Log likelihood of the complete-data problem that is "manufactured" in E-step

4. as it is based on unobservable data, it is replaced by its conditional expectation given the observed data, E-step is affected using the current fit for the unknown parameter.

$$\left[\partial \log L(\Psi) / \partial \Psi = 0 \right]$$

★ EM Algorithm Approach ⇒ problem of solving the incomplete data likelihood equation indirectly by proceding in terms of complete data log likelihood function $\log L_c(\Psi)$

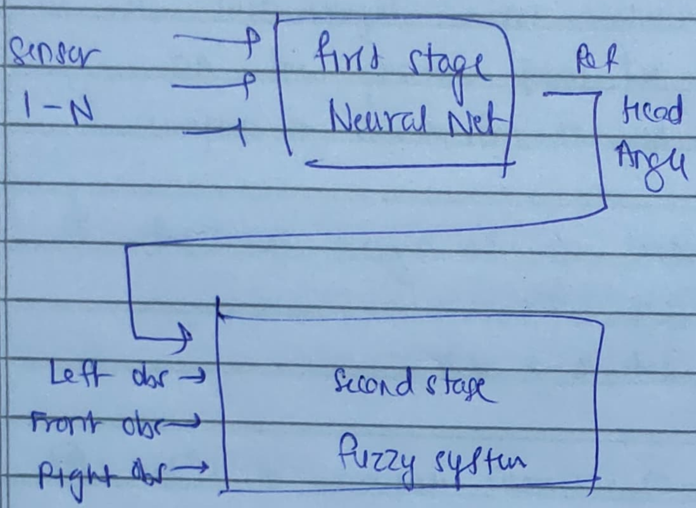
——— x ———

(BHAGYA RANA)
(Uaccsol2)

(Q5) Robot Navigation

Q5.>

Two stage Neuro-Fuzzy system



① Input from sensors
↓
neural network

② output by experiment

③ Neural network is trained accordingly

(MSE) mean square error

④ make adjustment in Neural Network

D19CS012
(BHAGYA RANA)

(Q6)

(Explain Hyper plane classification using maximal-margin)
(Illustrated & derive)

① if $w \rightarrow$ weight vector realising a functional margin of 1 on the positive point x^+ and the negative point x^-, compute it Geometric margin

② functional margin of 1 implies

$$\langle w \cdot x^+ \rangle + b = +1$$
$$\langle w \cdot x^- \rangle + b = -1$$

③ while to compute Geometric margin, Normalize w, geometric margin $\gamma \longrightarrow$ functional margin of resulting classifier

$$\gamma = \frac{1}{2}\left(\left\langle \frac{w}{\|w\|_2} \cdot x^+ \right\rangle - \left\langle \frac{w}{\|w\|_2} \cdot x^- \right\rangle \right)$$

$$= \frac{1}{2\|w\|_2} \left(\langle w \cdot x^+ \rangle - \langle w \cdot x^- \rangle \right) = \frac{1}{\|w\|_2}$$

④ Resulting margin $= \frac{1}{\|w\|_2}$

⑤ Given linearly seperable training sample

$$S = ((x_1, y_1); (x_2, y_2) \ldots (x_l, y_l))$$

⑥ Hyperplane $(w \cdot b)$ that solves optimization problem

$$\text{minimize}_{w, b} \quad \langle w, w \rangle$$

subject to $y_i (\langle w \cdot x_i \rangle + b) \geq 1, \quad i = 1 \ldots l,$

$\rightarrow$ realises the maximum margin hyperplane with geometric margin $= \left[\gamma = \frac{1}{\|w\|_2} \right]$

(transform it to dual problem

BHAGYA PANA
U19CS012

(Q7)

(COVID Disease)

system parameters

→ True positive
True negative
False positive
False negative

Forecasting the time based on past epidemic using machine learning

(Density function for time of covid)

~~dist~~ (distribution)

$$N = S(t) + I(t) + R(t)$$

$$\frac{d\,S(t)}{dt} = -\beta\, I(t)\, S(t)$$

$$\frac{d\,(R(t))}{dt} = \gamma\, I(t)$$

$$\frac{d\,I(t)}{dt} = \beta\, I(t)\, S(t) - \gamma\, I(t)$$

(curve fitting) → $(\beta \,\&\, \gamma)$ (minimizing the least squares)

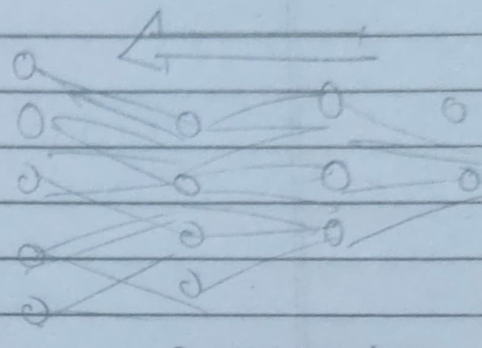
$$\boxed{I(t) = \frac{N}{(1 - K e^{-\beta t})}}$$

U19CS012

(BHAGYA RANA)

(Q8)

Back propogation Neural Network



Backpropogation

① Backpropogation = practice of fine tuning the weights of a neural net base on the error rate (i.e. loss) obtained in previous epoch (i.e. iteration)

Proper tuning of weights ensures ① lower error rates

② making model reliable by increasing its generalization.

③ We need activation function → determing activation value at every node

& hypothesis function

4 loss function

& calculate delta's in back propogation

④ We do delta calculation, step at every unit back-propogating the loss into the neural net & finding out what loss every node / unit is responsible for.

⑤ Update the weights

— x —