13.

perceptron : predicted 1 for T1

Age = 40,

credit score = 700

naive bayes : predicted 0 for T1

Model	predicted	Actual	T1	F1	TN	N
perceptron	1	0	0	1	0	0
naive bayes	0	0	0	0	1	0

* calculate precision and recall

precision $\frac{TP}{TP + FP}$

recal $\frac{TP}{TP + FN}$

* perceptron

$TP = 0$, $FP = 1$, $FN = 0$

precision $= 0$
recal $=$ undefined.

* naive bayes

TP = 0, FP = 0, FN = 0

precision = undefined
recall = undefined.

* recommendation: recall is more important for loan risk assessment to minimized high risk loans, reducing financial risk.

14.

Low risk $(y = 0)$:

credit score $= [720, 750, 780, 710]$

High risk $(y = 1)$

credit score $= [650, 600, 630, 640]$

Calculate variance of credit score.

Low risk

$$\text{mean} = \frac{720 + 750 + 780 + 710}{4}$$

$$= 740$$

$$\text{var} = \frac{3000}{4} = 750$$

High risk

mean = 630

var = 350

Calculate entropy for credit score

entropy formula.

Low risk

probabilities : $P(>700)=1$,

$P(650-700)=0$,

$P(<650)=0$

$H_{Low} = -(1 \cdot \log_2 1 + 0 + 0)$

$= 0$

High risk

Probabilities : $P(<650) = \frac{3}{4} = 0.75$

$P(650-700) = \frac{1}{4} = 0.25$

$P(>700) = 0$

$$H_{High} = -(0.75 \log_2 0.75 + 0.25 \log_2 0.25)$$

$$= 0.81125$$

Model handling : Tree leverage entropy for split , linear model may struggle with variance, naive bayes use distribution assumption, neural network need regulation for high variance.

15

Low risk ($y = 0$) :

[(35, 720), (45, 450), (52, 780),

(42, 710)]

High risk ($y = 1$) :

[(28, 650), (31, 600), (29, 630),

(33, 640)]

* normalized features.

Age : Min = 28, max = 52, range = 24

Credit score : Min = 600, max = 780

range = 180.

normalized

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Low : [(0.2917, 0.6667),
(0.7083, 0.8333),
(1.0, 1.0), (0.5833, 0.6111)]

High : [(0.0, 0.2778), (0.125, 0.0)
(0.417, 0.1667), (0.2083, 0.2222)

Identify support vectors

Low risk : $(0.5833, 0.6111)$

High risk : $(0.0, 0.2778)$

Calculate optimal margin hyperplane

$w \cdot x + b = 0$

assume $y = -1$ for high risk

$y = +1$ for low risk

support vector

$x_1 = (0.5833, 0.6111)$ $y_1 = +1$ (low)

$x_2 = (0.0, 0.2778)$ $y_2 = -1$ (high)

Hyperplane condition

$w \cdot x1 + b = 1$

$w \cdot x2 + b = -1$

Let $w = (w1, w2)$

$w1 \cdot 0.5833 + w2 \cdot 0.6111 + b = 1$ (1)

$w1 \cdot 0.0 + w2 \cdot 0.2778 = -1$ (2)

from (2)

$0.2778\, w2 + b = -1$

$\Rightarrow b = -1 - 0.2778\, w2$ (3)

substitude (3) into (1)

$0.5833\, w_1 + 0.6111\, w_2 +$

$(-1 - 0.2778\, w_2) = 1$

$0.5833\, w_1 + 0.3333\, w_2 = 2 \quad (4)$

from (3) set $w_2 = 0.3333$

$b = -1 - 0.2778 \cdot 0.3333$

$= -1.0926$

Hyperplane

$0.5833\, x_1 + 0.3333\, x_2 - 1.0926$

$= 0.$

Soft margin effect

Allow misclassification, potentially reducing the margin but improve robustness to noise and overlap in the data.

16.

Age : min = 28
max = 52

Credit score : min = 600
max = 780

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

$x_1 = (0.2917, 0.6667)$

$x_2 = (0.0, 0.2778)$

$x_3 = (0.7083, 0.8333)$

* Calculate kernel matrix

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

$$\gamma = 0.1$$

* compute pairwise distance

$$(\|x_i - x_j\|^2)$$

- x_1, x_1 $\quad \|x_1 - x_1\|^2 = 0$

$$K(x_1, x_1) = \exp(-0.1 \times 0) = 1.$$

- x_1, x_2 $\quad \|x_1 - x_2\|^2 = 0.2364$

$$K(x_1, x_2) = \exp(-0.1 \times 0.2364) = 0.9766.$$

- x_1, x_3 : $\| x_1, x_3 \|^2 = 0.2014$.

$$K(x_1, x_3) = \exp(-0.1 \times 0.2014)$$
$$= 0.9801$$

- x_2, x_2 : $K(x_2, x_2) = 1$.

- x_2, x_3 : $\| x_2, x_3 \|^2 = 0.8104$

$$K(x_2, x_3) = \exp(-0.1 \times 0.8104)$$
$$= 09221.$$

- x_3, x_3 : $K(x_3, x_3) = 1$.

Kernel matrix :

$$K = \begin{matrix} 1.000 & 0.9766 & 0.9801 \\ 0.9766 & 1.000 & 0.9221 \\ 0.981 & 0.9221 & 1.000 \end{matrix}$$

17.

T1 : Age = 0.5

Credit score = 0.5556

$x_{T1} = (0.5, 0.5556)$

* Calculate Euclidean distances

T1 to 1 : 0.236
2 : 0.572
3 : 0.347
4 : 0.670
5 : 0.669
6 : 0.601
7 : 0.1
8 : 0.443

* classify T1 with K = 3

ID 7 : 0.1 Low

ID 1 : 0.236 Low

ID 3 : 0.347 Low

K = 3 nearest neighbors : All Low risk

classification : T1 is classified as Low Risk

* effect of K : small K → local. noisy boundary, larger K → smoother, more generalized band balancing over fiting and underfiting.

18.

T1 (normalized) : Age = 0.5

Creditscore = 0.5556

distances

ID 1 : 0.236
2 : 0.572
3 : 0.347
4 : 0.670
5 : 0.669
6 : 0.601
7 : 0.1
8 : 0.443

⇒ 3 nearest neighbors
ID 7 : 0.1 (Low)
ID 1 : 0.236 (Low)
ID 3 : 0.347 (Low)

* distance-weighted K-NN

weights $w_i = \frac{1}{d_i}$

ID 7 : $w_7 = \frac{1}{0.1} = 10$

ID 1 : $w_1 = \frac{1}{0.236} = 4.237$

ID 3 : $w_3 = \frac{1}{0.347} = 2.882$

Total weight $= 10 + 4.237 + 2.882$
$= 17.119$

weight for Low $= 10 + 4.237 + 2.882$
$= 17.119$ (all neighbors are Low)

weight for High $= 0$

prop of Low $= \frac{17.119}{17.119} = 1$

prop of High $= 0$

* Robustness: distance-weight K-NN is more robust, as it prioritizes closer neighbors, reducing noise impact in a small dataset with overlapping features.

19

* define the HMM component

states : { Low , medium , high }

observation : [710 , 650 , 680]

creditscore over T = 3 time steps

Transition probabilities :

P (Low - Low) = 0.7 ,
P (Low - medium) = 0. 3
P (Low - high) = 0

P (medium - medium) = 0.6
P (medium - high) = 0.4
P (medium - Low) = 0

$P(\text{High} - \text{high}) = 0.8$

$P(\text{high} - \text{medium}) = 0.2$

$P(\text{high} - \text{Low}) = 0.$

Transition matrix

$$A = \begin{vmatrix} 0.7 & 0.3 & 0 \\ 0 & 0.6 & 0.4 \\ 0 & 0.2 & 0.8 \end{vmatrix}$$

Initial probabilities : assume uniform :

$\pi(\text{Low}) = \pi(\text{medium}) = \pi(\text{High})$

$= \frac{1}{3} = 0.3333$

emission probabilities

Low risk: > 700

medium risk: $650 - 700$

high risk: < 650

710 : $P(710 \mid Low) = 0.8$,

$P(710 \mid medium) = 0.15$,

$P(710 \mid high) = 0.05$

650 : $P(650 \mid low) = 0.05$

$P(650 \mid medium) = 0.25$

$P(650 \mid high) = 0.7$

680 : $P(680 \mid low) = 0.15$

$P(680 \mid medium) = 0.7$

$P(680 \mid high) = 0.15$.

* Apply Viterbi Algorithm

Initialization (t = 1, 710)

$$\delta_1(i) = \pi(i) \cdot P(710 \mid i)$$

low : 0.3333 × 0.8 = 0.26664

medium : 0.3333 × 0.15 = 0.049995

high : 0.3333 × 0.05 = 0.016665

Recursion (t = 2, 650)

$$\delta_t(j) = \max_i [\delta_{t-1}(i) \cdot A(i,j)] \times P(O_t \mid j)$$

low : max (0.26664 × 0.7, 0.049995 × 0,
0.016665 × 0) × 0.5

= 0.009332

ψ_2 (low) = low.

medium : max (0.26664 × 0.3,
0.049995 × 0.6,
0.016665 × 0.2) × 0.25

= 0.019998

ψ_2 (medium) = low

high : max (0.26664 × 0,
0.049995 × 0.4,
0.016665 × 0.8) × 0.7

= 0.013999

ψ_2 (high) = high.

recursion (t = 3, 680)

Low : max (0.009332 × 0.7,
0.019998 × 0 ,
0.013999 × 0) × 0.15
= 0.0009799

medium : max (0.009332 × 0.3
0.019998 × 0.6
0.13999 × 0.2) × 0.7
= 0.008399

high : max (0.009332 × 0
0.19998 × 0.4
0.13999 × 0.8) × 0.15
= 0.0016798

Back track

t = 3 medium sequence Low → medium →
t = 2 mdium
t = 1 Low

20.

$$A = \begin{matrix} 0.7 & 0.3 & 0 \\ 0 & 0.6 & 0.4 \\ 0 & 0.2 & 0.8 \end{matrix}$$

Initial prop = 0.3333

- 705 (similar 710, likely Low risk)

$P(705 \mid Low) = 0.8$
$P(705 \mid medium) = 0.15$
$P(705 \mid high) = 0.05$

- 645 (similar 650, likely high risk)

$P(645 \mid Low) = 0.05$
$P(645 \mid medium) = 0.25$
$P(645 \mid high) = 0.7$

forward algorithm

(t=1, 705)

low : 0.3333 × 0.8 = 0.26664
medium : 0.3333 × 0.15 = 0.049995
high : 0.3333 × 0.05 = 0.016665

forward step (t=2, 645)

$$\alpha_t(j) = \left(\sum_i \alpha_{t-1}(i)\, A(i,j)\right)$$

$$\times\, f(o_t \mid j)$$

Low $(0.26664 \times 0.7 + 0.049995 \times 0 + 0.016665 \times 0) \times 0.05$

$= 0.009332$ (1)

medium $(0.26664 \times 0.3 + 0.049995 \times 0.6 + 0.16665 \times 0.2) \times 0.25$

$= 0.007081$ (2)

high $(0.26664 \times 0 + 0.049995 \times 0.4 + 0.16665 \times 0.8) \times 0.7$

$= 0.023327$ (3)

Total prop $= (1) + (2) + (3)$

$= 0.03974.$

Propabilitigy of $(705, 645) = 0.03974$

HMM predicts future credit risk states and trends, enabling proactive risk mangement.