

Subject Name (Gujarati)

4341603 -- Summer 2024

Semester 1 Study Material

Detailed Solutions and Explanations

પ્રશ્ન 1(અ) [3 ગુણ]

ઓળય ઉદાહરણનો ઉપયોગ કરીને મશીન લર્નિંગને વ્યાખ્યાયિત કરો

જવાબ

મશીન લર્નિંગ આર્ટિફિશિયલ ઇન્ટેલિજન્સનો એક ભાગ છે જે કમ્પ્યુટર્સને ડેટામાંથી શીખવા અને દરેક કાર્ય માટે સ્પષ્ટ રીતે પ્રોગ્રામ કર્યા વિના નિર્ણયો લેવા માટે સક્ષમ બનાવે છે.

ટેબલ: મશીન લર્નિંગના મુખ્ય ઘટકો

ઘટક	વર્ણન
ડેટા	ટ્રેનિંગ માટે ઉપયોગમાં લેવાતી ઇનપુટ માહિતી
અલ્ગોરિધમ	પેટન શીખતા ગાળિટિક મોડેલ
ટ્રેનિંગ	અલ્ગોરિધમને શીખવવાની પ્રક્રિયા
પ્રિડિક્શન	શીખેલા પેટન આધારિત આઉટપુટ

ઉદાહરણ: ઇમેઇલ સ્પામ ડિટેક્શન સિસ્ટમ હજારો ઇમેઇલોમાંથી "સ્પામ" અથવા "નોટ સ્પામ" તરીકે લેબલ કરેલા ઇમેઇલોમાંથી શીખે છે અને નવા ઇમેઇલોને આપોઆપ વર્ગીકૃત કરે છે.

મેમરી ટ્રીક

"ડેટા ડ્રાઇવ્સ ડિસિજન્સ" - ડેટા અલ્ગોરિધમને બુદ્ધિશાળી નિર્ણયો લેવા માટે પ્રશિક્ષિત કરે છે

પ્રશ્ન 1(બ) [4 ગુણ]

સ્કેમેટિક રેખાકૃતિના ઉપયોગ કરીને મશીન લર્નિંગની પ્રક્રિયા સમજાવો

જવાબ

મશીન લર્નિંગ પ્રક્રિયામાં ડેટા સંગ્રહથી લઈને મોડેલ ડિપ્લોયમેન્ટ સુધીના વ્યવસ્થિત પગલાંઓનો સમાવેશ થાય છે.

```
flowchart LR
    A[ ] --> B[ ]
    B --> C[ ]
    C --> D[ ]
    D --> E[ ]
    E --> F[ ]
    F --> G\{ ? \}
    G --> D
    G --> H[ ]
    H --> I[ ]
```

પ્રક્રિયાના પગલાં:

- ડેટા સંગ્રહ: સંબંધિત ડેટાસેટ એકત્રિત કરવું
- પ્રીપોરેસિંગ: ડેટાને સાફ્ અને તૈયાર કરવું
- ટ્રેનિંગ: ટ્રેનિંગ ડેટાનો ઉપયોગ કરીને અલ્ગોરિધમને શીખવવું
- વેલિડેશન: મોડેલની કામગીરીને ચકાસવી
- ડિપ્લોયમેન્ટ: વાર્સ્ટાફિક પ્રિડિક્શન માટે મોડેલનો ઉપયોગ

મેમરી ટ્રીક

"કમ્પ્યુટર્સ કેન ટુલી થિંક" - કલેક્ટ, કલીન, ટ્રેન, ટેસ્ટ

પ્રશ્ન 1(ક) [7 ગુણ]

યોગ્ય એપ્લિકેશન સાથે વિવિધ પ્રકારના મશીન લર્નિંગ સમજાવો

જવાબ

મશીન લર્નિંગ અભ્યાસિન્ડમસને લર્નિંગ એપ્રોચ અને ઉપલબ્ધ ડેટાના આધારે વર્ગીકૃત કરવામાં આવે છે.

ટેબલ: મશીન લર્નિંગના પ્રકારો

પ્રકાર	લર્નિંગ મેથડ	ડેટા આવશ્યકતા	ઉદાહરણ એપ્લિકેશન
સુપરવાઈઝડ	લેબલડ ડેટાનો ઉપયોગ	ઇનપુટ-આઉટપુટ જોડીઓ	ઇમેઇલ કલાસિફિકેશન
અનસુપરવાઈઝડ	છુપાયેલા પેટર્ન શોધે	માત્ર ઇનપુટ ડેટા	કસ્ટમર સેગમેન્ટેશન
રિઝન્ફોર્મેન્ટ	રિવર્સ્યુલસ દ્વારા શીખે	એન્વાયન્મેન્ટ ફીડબેક	ગેમ પ્લેયિંગ AI

એપ્લિકેશન્સ:

- સુપરવાઈઝડ લર્નિંગ: મેડિકલ ડાયગ્નોસ્ટિસ, ઇમેજ રેકોગ્નિશન, ફ્રોડ ડિટેક્શન
- અનસુપરવાઈઝડ લર્નિંગ: માર્કેટ રિસર્ચ, એનોમેલી ડિટેક્શન, રેકમેન્ડેશન સિસ્ટમ્સ
- રિઝન્ફોર્મેન્ટ લર્નિંગ: ઓટોનોમસ વેહિકલ્સ, રોબોટિક્સ, સ્ટ્રોટેજિક ગેમ્સ

ડાયાગ્નોમાં: લર્નિંગ ટાઇપ્સ

mindmap
root(())

મેમરી ટ્રીક

"સ્ટ્રોક-ટ્રેક યુગ્યુઅલી રીમેમ્બર" - સુપરવાઈઝડ, અનસુપરવાઈઝડ, રિઝન્ફોર્મેન્ટ

પ્રશ્ન 1(ક) OR [7 ગુણ]

મશીન લર્નિંગમાં વિવિધ સમસ્યાઓ શું છે? ત્રણ સમસ્યાઓ કે જે મશીન લર્નિંગનો ઉપયોગ કરીને ઉકેલી શકતી નથી.

જવાબ

ટેબલ: મશીન લર્નિંગની સમસ્યાઓ

સમસ્યા કેટેગરી	વર્ણન	અસર
ડેટા કવોલિટી	અધૂરો, નોઇજી, પક્ષપાતી ડેટા	નબળું મોડેલ પરફોર્મન્સ
ઓવરફિટિંગ	મોડેલ ટ્રેનિંગ ડેટાને યાદ રાખે છે	નબળું જનરલાઇઝેશન
કમ્પ્યુટેશનલ	ઉચ્ચ પ્રોસેસિંગ આવશ્યકતાઓ	રિસોર્સ મર્યાદાઓ
ઇન્ટરપ્રિટેબિલિટી	બ્લેક બોક્સ મોડેલ્સ	પારદર્શિતાનો અભાવ

ML માટે અનુપર્યકૃત સમસ્યાઓ:

- સિંગલ રૂલ-બેસ્ડ ટાસ્ક - મૂળભૂત ગણતરીઓ, સિંગલ if-then લોજિક
- નૈતિક નિર્ણયો - માનવીય મૂલ્યોની આવશ્યકતા ધરાવતા નૈતિક એન્ટ્રેસ
- ફિચર એક્સપ્રેશન - માનવીય લાગણીની આવશ્યકતા ધરાવતી મૂળ કલાત્મક સર્જના

અન્ય સમસ્યાઓ:

- પ્રાઇવ્સી ચિંતાઓ: સંવેદનશીલ ડેટા હેન્ડલિંગ
- બાયસ પ્રોપેશન: અન્યાયકારક અલોરિધમિક નિર્ણયો
- ફીચર સિલેક્શન: સંબંધિત ઇનપુટ વેરિએબલ્સ પરસંદ કરવા

મેમરી ટ્રીક

"ડેટા ફ્રાઇલ્સ કવોલિટી" - ડેટા કવોલિટી સીધી રીતે મોડેલ કવોલિટીને અસર કરે છે

પ્રશ્ન 2(અ) [3 ગુણ]

સામાન્ય મશીન લર્નિંગ સમસ્યામાં વિવિધ પ્રકારના ડેટાનો સારાંશ આપો

જવાબ

ટેબલ: મશીન લર્નિંગમાં ડેટા પ્રકારો

ડેટા પ્રકાર	વર્ણન	ઉદાહરણ
ન્યુમેરિકલ	માત્રાત્મક મૂલ્યો	ઉંમર: 25, ઊંચાઈ: 170cm
કેટેગોરિકલ	અસ્પષ્ટ કેટેગરીઓ	રંગ: લાલ, વાદળી, લીલો
ઓડિનલ	કમબદ્ધ કેટેગરીઓ	રેટિંગ: નબળું, સારં, ઉત્તમ
બાઇનરી	બે શક્ય મૂલ્યો	લિંગ: પુરુષ/સ્ત્રી

લક્ષણો:

- સ્ટ્રક્ચર્ડ: ટેબલોમાં વ્યવસ્થિત (ડેટાબેસેસ, સ્પ્રેડશીટ્સ)
- અનસ્ટ્રક્ચર્ડ: ઇમેજ, ટેક્સ્ટ, ઓડિયો ફાઇલો
- ટાઇમ-સીરીઝ: સમય પર ડેટા પોઇન્ટ્સ

મેમરી ટ્રીક

"નંબર્સ કાઉન્ટ બેટર દેન વર્ડ્સ" - ન્યુમેરિકલ, કેટેગોરિકલ, બાઇનરી, ટેક્સ્ટ

પ્રશ્ન 2(બ) [4 ગુણ]

બંને એટ્રિબ્યુટ માટે વેરિયન્સ ગણતરી કરો. નક્કી કરો કે કઈ એટ્રિબ્યુટ મીનની આસપાસ સ્પેડ આઉટ છે

જવાબ

આપેલ ડેટા:

- એટ્રિબ્યુટ 1: 32, 37, 47, 50, 59
- એટ્રિબ્યુટ 2: 48, 40, 41, 47, 49

ગણતરીઓ:

એટ્રિબ્યુટ 1:

- મીન = $(32+37+47+50+59)/5 = 225/5 = 45$
- વેરિયન્સ = $[(32-45)^2 + (37-45)^2 + (47-45)^2 + (50-45)^2 + (59-45)^2]/5$
- વેરિયન્સ = $[169 + 64 + 4 + 25 + 196]/5 = 458/5 = 91.6$

એટ્રિબ્યુટ 2:

- મીન = $(48+40+41+47+49)/5 = 225/5 = 45$
- વેરિયન્સ = $[(48-45)^2 + (40-45)^2 + (41-45)^2 + (47-45)^2 + (49-45)^2]/5$
- વેરિયન્સ = $[9 + 25 + 16 + 4 + 16]/5 = 70/5 = 14$

પરિણામ: એટ્રિબ્યુટ 1 (વેરિયન્સ = 91.6) એટ્રિબ્યુટ 2 (વેરિયન્સ = 14) કરતાં વધુ સ્પેડ આઉટ છે.

મેમરી ટ્રીક

"હાયર વેરિયન્સ શોડ સ્પ્રેડ" - વધુ વેરિયન્સ વધુ વિખેરાઈને દર્શાવે છે

પ્રશ્ન 2(ક) [7 ગુણ]

ડેટા ગુણવત્તા સમસ્યા તરફ દોરી જતા ફેક્ટર્સની યાદી બનાવો. આઉટલાયર્સ અને મિસિંગ વેલ્ચુ કેવી રીતે હેન્ડલ કરવું

જવાબ

ટેબલ: ડેટા ગુણવત્તા સમસ્યાઓ

ફેક્ટર	કારણ	સોલ્યુશન
આપૂર્ણતા	મિસિંગ ડેટા કલેક્શન	ઇમ્પ્યુટેશન ટેકનિક્સ
અસંગતતા	વિવિધ ડેટા ફોર્મેટ્સ	સ્ટેન્ડર્ડાઇઝેશન
અચોક્કસત્તા	હુમન/સેન્સર એરર્સ	વેલિડેશન રૂલ્સ
નોઇજ	રેન્ડમ વેરિએશન્સ	ફિલ્ટરિંગ મેથડ્સ

આઉટલાયર્સ હેન્ડલ કરવું:

- ડિટેક્શન: સ્ટેટિસ્ટિકલ મેથડ્સ (Z-score, IQR)
- સ્ટીમેન્ટ: એક્સ્ટ્રીમ વેલ્ચુઝને રીમૂવ, ટ્રાન્સફોર્મ, અથવા કેપ કરવી
- વિઝ્યુઅલાઇઝેશન: બોક્સ પ્લોટ્સ, સ્કેટર પ્લોટ્સ

મિસિંગ વેલ્ચુઝ હેન્ડલ કરવું:

- ડિલીશન: આપૂર્ણ રેકૉર્ડ્સ રીમૂવ કરવા
- ઇમ્પ્યુટેશન: મીન, મોડિયન, અથવા મોડ સાથે ભરવું
- પ્રિડિક્શન: મિસિંગ વેલ્ચુઝની આગાહી કરવા માટે ML નો ઉપયોગ

કોડ ઉદાહરણ:

```
\#
df.fillna(df.mean())  #
df.dropna()           #
```

મેમરી ટ્રીક

"કલીન ડેટા મેક્સ મોડેલ્સ" - સાફ્ ડેટા બેહતર મોડેલ્સ બનાવે છે

પ્રશ્ન 2(અ) OR [3 ગુણ]

વિવિધ મશીન લર્નિંગ પ્રવૃત્તિઓ આપો

જવાબ

ટેબલ: મશીન લર્નિંગ પ્રવૃત્તિઓ

પ્રવૃત્તિ	હેતુ	ઉદાહરણ
ડેટા કલેક્શન	સંબંધિત માહિતી એકમિત કરવી	સર્વ, સેન્સર્સ, ડેટાબેસેસ
ડેટા પ્રીપોરેસિંગ	ડેટાને સાફ્ અને તૈયાર કરવું	નોઇજ રીમૂવ કરવું, મિસિંગ વેલ્ચુઝ હેન્ડલ કરવું
ફીચર એન્જિનિયરિંગ	અર્થપૂર્ણ વેરિએબ્લ્સ બનાવવા	રોડિયાંથી ફીચર્સ એક્સ્ટ્રાક્ટ કરવા
મોડેલ ટ્રેનિંગ	અલગોરિધમને પેટર્ન શીખવવા	ટ્રેનિંગ ડેટાસેટનો ઉપયોગ
મોડેલ ઇવેલ્ચુઅશન	પરફોર્મન્સ આકારણી	ટેસ્ટ એક્ષ્યુરસી, પ્રિસ્જન, રિકોલ
મોડેલ ડિપ્લોયમેન્ટ	મોડેલને પ્રોડક્શનમાં મૂકવું	વેબ સર્વિસેસ, મોબાઇલ એપ્સ

મુખ્ય પ્રવૃત્તિઓ:

- એક્સ્પ્લોરેટરી ડેટા એનાલિસિસ: ડેટા પેટર્ન સમજવા
- હાયપરપેરામીટર ટ્યુનિંગ: મોડેલ સેટિંગ્સ ઓપ્ટિમાઇઝ કરવા
- કોડ-વેલિડેશન: મજબૂત પરફોર્મન્સ આકારણી

મેમરી ટ્રીક

"ડેટા મોડેલ્સ પર્ફોર્મ એક્સેલન્ટલી" - ડેટા તૈયારી, મોડેલ બિલ્ડિંગ, પર્ફોર્મન્સ ઇવેલ્યુઅશન, એક્ઝિક્યુશન

પ્રશ્ન 2(બ) OR [4 ગુણ]

નીચેની સંખ્યાઓના મીન અને મીડિયન ની ગણતરી કરો: 12,15,18,20,22,24,28,30

જવાબ

આપેલ સંખ્યાઓ: 12, 15, 18, 20, 22, 24, 28, 30

મીન ગણતરી: મીન = $(12+15+18+20+22+24+28+30)/8 = 169/8 = 21.125$

મીડિયન ગણતરી:

- સંખ્યાઓ પહેલેથી સોર્ટ કરેલી છે: 12, 15, 18, 20, 22, 24, 28, 30
- સમ કાઉન્ટ (8 સંખ્યાઓ)
- મીડિયન = $(4\text{મી સંખ્યા} + 5\text{મી સંખ્યા})/2 = (20 + 22)/2 = 21$

ટેબલ: સ્ટેટિસ્ટિકલ સમરી

માપદંડ	મૂલ્ય	વર્ણન
મીન	21.125	સરેરાશ મૂલ્ય
મીડિયન	21	મધ્યમ મૂલ્ય
કાઉન્ટ	8	કુલ સંખ્યાઓ

મેમરી ટ્રીક

"મિડલ મેક્સ મીડિયન" - મધ્યમ મૂલ્ય મીડિયન આપે છે

પ્રશ્ન 2(ક) OR [7 ગુણ]

ડેટા પ્રીપ્રોસેસિંગના સંદર્ભમાં ડાયમેન્શનાલિટી રિડક્ષન અને ફીચર સબસેટ સિલેક્શન પર ટૂંકી નોંધ લખો

જવાબ

ડાયમેન્શનાલિટી રિડક્ષન અપ્રસ્તુત ફીચર્સને દૂર કરે છે અને કોમ્પ્યુટેશનલ જટિલતા ઘટાડ છે જ્યારે મહત્વપૂર્ણ માહિતી જાળવી રાખે છે.
ટેબલ: ડાયમેન્શનાલિટી રિડક્ષન ટેકનિક્સ

ટેકનિક	મેથડ	વપરાશ
PCA	પ્રિન્સિપલ કર્મ્પોન્નટ એનાલિસિસ	લીનિયર રિડક્ષન
LDA	લીનિયર ડિસ્ટિન્શનન્ટ એનાલિસિસ	કલાસિફિકેશન ટાસ્ક્સ
t-SNE	નોન-લીનિયર એમ્બેડિંગ	વિભ્યુઅલાઇઝેશન
ફીચર સિલેક્શન	મહત્વપૂર્ણ ફીચર્સ પરંદ કરવા	ઓવરફિટિંગ ઘટાડવું

ફીચર સબસેટ સિલેક્શન મેથડ્સ:

- ફિલ્ટર મેથડ્સ: સ્ટેટિસ્ટિકલ ટેસ્ટ્સ, કોરિલેશન એનાલિસિસ
- રેપર મેથડ્સ: ફોરવર્ડ/બેકવર્ડ સિલેક્શન
- એમ્બેડ મેથડ્સ: LASSO, રિજ રિગ્રેશન

ફાયદાઓ:

- કોમ્પ્યુટેશનલ કાર્યક્રમતા: ઝડપી ટ્રેનિંગ અને પ્રિડિક્ષન
- સ્ટોરેજ રિડક્ષન: ઓછી મેમરી આવશ્યકતાએ
- નોઇજ રિડક્ષન: અપ્રસ્તુત ફીચર્સ દૂર કરવા
- વિભ્યુઅલાઇઝેશન: 2D/3D પ્લોટિંગ સક્ષમ કરવું

કોડ ઉદાહરણ:

```
from sklearn.decomposition import PCA
pca = PCA(n_components=2)
reduced_data = pca.fit_transform(data)
```

મેમરી ટ્રીક

“રિજ્યુસ ફીચર્સ, ઇમ્પ્રૂવ પર્ફોર્મન્સ” - ઓછા ફીચર્સ ધારી વાર બેહતર મોડેલ્સ તરફ દોરી જાય છે

પ્રશ્ન 3(અ) [3 ગુણ]

શું બાયસ ML મોડેલના પરફોર્મન્સને અસર કરે છે? ટૂંકમાં સમજાવો

જવાબ

હા, બાયસ પ્રિડિક્શન-સમાં સિસ્ટેમેટિક એરર્સ બનાવીને ML મોડેલના પરફોર્મન્સને નોંધપાત્ર રીતે અસર કરે છે.

ટેબલ: બાયસના પ્રકારો

બાયસ પ્રકાર	વર્ણન	અસર
સિલેક્શન બાયસ	બિન-પ્રતિનિધિત્વકારી ડેટા	નબળું જનરલાઇઝેશન
કન્ફર્મેશન બાયસ	અપેક્ષિત પરિણામોની તરફેણ	ત્રાંસા નિષ્કર્ષો
અભ્યારિધમિક બાયસ	મોડેલ ધારણાઓ	અન્યાયકારક પ્રિડિક્શન-સ

પરફોર્મન્સ પર અસરો:

- અંડરફિટિંગ: ઉચ્ચ બાયસ અનિ સરળ મોડેલ્સ તરફ દોરી જાય છે
- નબળી ચોકસાઈ: સિસ્ટેમેટિક એરર્સ એકંદર પરફોર્મન્સ ઘટાડ છે
- અન્યાયકારક નિષ્યથો: પક્ષપાતી મોડેલ્સ જૂથો સામે ભેદભાવ કરે છે

ઘટાડવાની વ્યૂહદયનાયો:

- વિવિધ ટ્રેનિંગ ડેટા
- કોસ-વેલિડેશન ટેકનિક્સ
- બાયસ ડિટેક્શન અભ્યારિધમસ

મેમરી ટ્રીક

“બાયસ બ્રેક્સ બેટર પર્ફોર્મન્સ” - બાયસ મોડેલની અસરકારકતા ઘટાડ છે

પ્રશ્ન 3(બ) [4 ગુણ]

કોસ-વેલિડેશન અને બૂટસ્ટ્રેપ સેમ્પલિંગની સરખામણી કરો

જવાબ

ટેબલ: કોસ-વેલિડેશન vs બૂટસ્ટ્રેપ સેમ્પલિંગ

પાસું	કોસ-વેલિડેશન	બૂટસ્ટ્રેપ સેમ્પલિંગ
મેથડ	ડેટાને ફોલ્ડ્સમાં વિભાજિત કરવું	રિપ્લેસમેન્ટ સાથે સેમ્પલ કરવું
ડેટા ઉપયોગ	બધો ડેટા વાપરે છે	માલ્ટિપલ સેમ્પલ્સ બનાવે છે
હેતુ	મોડેલ ઇવેલ્યુઅશન	અનિશ્ચિતતાનો અંદાજ
ઓવરલેપ	સેટ્સ વચ્ચે કોઈ ઓવરલેપ નથી	ડુલિકેટ સેમ્પલસની મંજૂરી

કોસ-વેલિડેશન:

- ડેટાને k સમાન ભાગોમાં વહેંચે છે
- k-1 ભાગોમાં ટ્રેન કરે છે, 1 ભાગમાં ટેસ્ટ કરે છે
- મજબૂત ઇવેલ્યુઅશન માટે k વખત પુનરાવર્તન કરે છે

બૂટસ્ટ્રેપ સેમ્પલિંગ:

- રિપ્લેસમેન્ટ સાથે રેન્ડમ સેમ્પલ બનાવે છે
- સમાન સાઇગના માલ્ટિપલ ડેટાસેટ્સ જનરેટ કરે છે
- કોન્ફિડન્સ ઇન્ટરવલ્સનો અંદાજ કાઢે છે

એપ્લિકેશન્સ:

- કોસ-વેલિડેશન: મોડેલ સિલેક્શન, હાયપરપેરામીટર ટ્યુનિંગ
- બૂટસ્ટ્રેપ: સ્ટેટિસ્ટિકલ ઇન્ફરન્સ, કોન્ફિડન્સ એસ્ટિમેશન

મેમરી ટ્રીક

"કોસ ચેકસ, બૂટસ્ટ્રેપ બિલ્ડુસ" - કોસ-વેલિડેશન પરફોર્મન્સ ચેક કરે છે, બૂટસ્ટ્રેપ કોન્ફિડન્સ બિલ્ડ કરે છે

પ્રશ્ન 3(ક) [7 ગુણ]

કન્ફ્યુઝન મેટ્રિક્સ ગણતરી અને મેટ્રિક્સ

જવાબ

આપેલ માહિતી:

- True Positive (TP): 83 (પ્રિડિક્ટેડ ખરીદશે, વાસ્તવમાં ખરીદ્યું)
- False Positive (FP): 7 (પ્રિડિક્ટેડ ખરીદશે, નથી ખરીદ્યું)
- False Negative (FN): 5 (પ્રિડિક્ટેડ નહીં ખરીદે, વાસ્તવમાં ખરીદ્યું)
- True Negative (TN): 5 (પ્રિડિક્ટેડ નહીં ખરીદે, નથી ખરીદ્યું)

કન્ફ્યુઝન મેટ્રિક્સ:

	પ્રિડિક્ટેડ ખરીદશે	પ્રિડિક્ટેડ નહીં ખરીદે
વાસ્તવમાં ખરીદે	83 (TP)	5 (FN)
વાસ્તવમાં નહીં ખરીદે	7 (FP)	5 (TN)

ગણતરીઓ:

- અ) એરર રેટ: એરર રેટ = $(FP + FN) / કુલ = (7 + 5) / 100 = 0.12 = 12\%$
 બ) પ્રિસિઝન: પ્રિસિઝન = $TP / (TP + FP) = 83 / (83 + 7) = 83/90 = 0.922 = 92.2\%$
 ક) રિકોલ: રિકોલ = $TP / (TP + FN) = 83 / (83 + 5) = 83/88 = 0.943 = 94.3\%$
 ડ) F-મેઝર: $F\text{-મેઝર} = 2 \times (\times) / (+)F = 2 \times (0.922 \times 0.943) / (0.922 + 0.943) = 0.932 = 93.2\%$
 ટેબલ: પરફોર્મન્સ મેટ્રિક્સ

મેટ્રિક	મૂલ્ય	અર્થધારણ
એરર રેટ	12%	મોડેલ 12% ખોટો આગાહીઓ કરે છે
પ્રિસિઝન	92.2%	પ્રિડિક્ટેડ ખરીદદારોમાંથી 92.2% ખરેખર ખરીદે છે
રિકોલ	94.3%	મોડેલ 94.3% વાસ્તવિક ખરીદદારોને ઓળખે છે
F-મેઝર	93.2%	સંતુલિત પરફોર્મન્સ માપદંડ

મેમરી ટ્રીક

"પ્રોફેસિનલ રિકોલ ફાઇન્ડ્સ એવરીવન" - પ્રિસિઝન ચોક્સાઈ માપે છે, રિકોલ બધા પોઝિટિવ શોધે છે

પ્રશ્ન 3(અ) OR [3 ગુણ]

સંક્ષિપ્તમાં વ્યાખ્યાયિત કરો: અ) ટાર્ગેટ ફંક્શન બ) કોસ્ટ ફંક્શન ક) લોસ ફંક્શન

જવાબ

ટેબલ: ફંક્શન વ્યાખ્યાયો

ફંક્શન	વ્યાખ્યા	હેતુ
ટાર્ગેટ ફંક્શન	ઇનપુટથી આઉટપુટ સુધીની આદર્શ મેપિંગ	આપણે શું શીખવા માગીએ છીએ
કોસ્ટ ફંક્શન	અંકુદર મોડેલ એરરને માપે છે	કુલ પરફોર્મન્સનું મૂલ્યાંકન
લોસ ફંક્શન	એક પ્રિડિક્શન માટે એરર માપે છે	વ્યક્તિગત પ્રિડિક્શન એરર

વિગતવાર સમજૂતી:

- ટાર્ગેટ ફંક્શન: $f(x) = y$, સાચો સંબંધ જેણો આપણે અંદાજ કાઢવા માગીએ છીએ
- કોસ્ટ ફંક્શન: તમામ લોસ ફંક્શનની સરેરાશ, $J = (1/n) \sum_{i=1}^n loss(y_i, \hat{y}_i)$
- લોસ ફંક્શન: એક સેમ્પલ માટે એરર, દા.ત., $(y_i - \hat{y}_i)^2$

સંબંધ: કોસ્ટ ફંક્શન સામાન્ય રીતે તમામ ટ્રેનિંગ ઉદાહરણોમાં લોસ ફંક્શનની સરેરાશ હોય છે.

મેમરી ટ્રીક

"ટાર્ગેટ કોસ્ટસ લેસ" - ટાર્ગેટ ફંક્શન આદર્શ છે, કોસ્ટ ફંક્શન એકદર એરર માપે છે, લોસ ફંક્શન વ્યક્તિગત એરર માપે છે

પ્રશ્ન 3(બ) OR [4 ગુણ]

બેલેન્ડ ફિટ, અંડરફિટ અને ઓવરફિટ સમજાવો

જવાબ

ટેબલ: મોડેલ ફિટિંગ પ્રકારો

ફિટ પ્રકાર	ટ્રેનિંગ એરર	વેલિડેશન એરર	લક્ષણો
અંડરફિટ	ઊચો	ઊચો	ખૂબ સાંદુર્ય મોડેલ
બેલેન્ડ ફિટ	નીચો	નીચો	આદર્શ જટિલતા
ઓવરફિટ	ખૂબ નીચો	ઊચો	ખૂબ જટિલ મોડેલ

વિઝ્યુઅલાઇઝન:

Mermaid Diagram (Code)

```
{Shaded}
{Highlighting} []
graph LR
    A[ ] --- B[ ]
    B --- C[ ]
    A --- D[ ]
    C --- E[ ]
    B --- F[ ]
{Highlighting}
{Shaded}
```

લક્ષણો:

- અંડરફિટ: મોડેલ ખૂબ સાંદુર્ય, પેટર્ન કેપ્ચર કરી શકતું નથી
- બેલેન્ડ ફિટ: યોગ્ય જટિલતા, સારં જનરલાઇઝન
- ઓવરફિટ: મોડેલ ખૂબ જટિલ, ટ્રેનિંગ ડેટાને યાદ રાખે છે

સોલ્યુશન્સ:

- અંડરફિટ: મોડેલ જટિલતા વધારવી, ફીચર્સ ઉમેરવા
- ઓવરફિટ: રેગ્યુલરાઇઝન, કોસ-વેલિડેશન, વધુ ડેટા

મેમરી ટ્રીક

"બેલેન્ડ બિંગ્સ બેસ્ટ રિજલ્ટ્સ" - સંતુલિત મોડેલ્સ નવા ડેટા પર શ્રેષ્ઠ પરફોર્મ કરે છે

પ્રશ્ન 4(અ) [3 ગુણ]

કલાસિફિકેશન લાર્નિંગ સ્ટેપ્સ આપો

જવાબ

ટેબલ: કલાસિફિકેશન લાર્નિંગ સ્ટેપ્સ

સ્ટેપ	વર્ણન	હેતુ
ડેટા કલેક્શન	લેબલ ઉદાહરણો એકત્રિત કરવા	ટ્રેનિંગ મટેરિયલ પ્રદાન કરવું
પ્રીપોરેસિંગ	ડેટાને સાફ અને તૈયાર કરવું	ડેટા ગુણવત્તા સુધારવી
ફીચર સિલેક્શન	સંબંધિત એટ્રિબ્યુટ્સ પસંદ કરવા	જટિલતા ઘટાડવી
મોડેલ ટ્રેનિંગ	ટ્રેનિંગ ડેટામાંથી શીખવું	કલાસિફિકેશર બનાવવું
ઇવેલ્યુઅશન	મોડેલ પરફોર્મન્સ ટેસ્ટ કરવું	ચોક્સાઈ આકારવી
ડિપ્લોયમેન્ટ	નવી આગાહીઓ માટે ઉપયોગ	પ્રેક્ટિકલ એપ્લિકેશન

વિગતવાર પ્રક્રિયા:

- ડેટાસેટ તૈયાર કરવું ઇનપુટ ફીચર્સ અને કલાસ લેબલ્સ સાથે
- ડેટા સ્પાલટ કરવું ટ્રેનિંગ અને ટેસ્ટિંગ સેટ્સમાં
- કલાસિફિકેશર ટ્રેન કરવું ટ્રેનિંગ ડેટાનો ઉપયોગ કરીને
- મોડેલ વેલિડેટ કરવું ટેસ્ટ ડેટાનો ઉપયોગ કરીને
- પેરામીટર્સ ફાઇન-ટ્યુન કરવા આદર્શ પરફોર્મન્સ માટે

મેમરી ટ્રીક

"ડેટા પ્રેપેરેશન ફેસિલિટેટ્સ મોડેલ એક્સેલ-ન્સ" - ડેટા પ્રેપ, ફીચર સિલેક્શન, મોડેલ ટ્રેનિંગ, ઇવેલ્યુઅશન

પ્રશ્ન 4(બ) [4 ગુણ]

લીનિયર રિલેશનશિપ ગણતરી

જવાબ

આપેલ ડેટા:

કલાકો (X)	પરીક્ષા સ્કોર (Y)
2	85
3	80
4	75
5	70
6	60

લીનિયર રિલેશન ગણતરી:

સ્ટેપ 1: મીન્સ કેલ્ક્યુલેટ કરવા

- $\bar{X} = (2+3+4+5+6)/5 = 4$
- $\bar{Y} = (85+80+75+70+60)/5 = 74$

સ્ટેપ 2: સ્લોપ (b) કેલ્ક્યુલેટ કરવું

- $-ન્યુમેરેટર = \sum(X-\bar{X})(Y-\bar{Y}) = (2-4)(85-74) + (3-4)(80-74) + (4-4)(75-74) + (5-4)(70-74) + (6-4)(60-74)$
- $= (-2)(11) + (-1)(6) + (0)(1) + (1)(-4) + (2)(-14) = -22 - 6 + 0 - 4 - 28 = -60$
- $\cdot ડિનોમિનેટર = \sum(X-\bar{X})^2 = (-2)^2 + (-1)^2 + (0)^2 + (1)^2 + (2)^2 = 4 + 1 + 0 + 1 + 4 = 10$
- $\cdot b = -60/10 = -6$

સ્ટેપ 3: ઇન્ટરસોન (a) કેલ્ક્યુલેટ કરવું

- $a = \bar{Y} - b = 74 - (-6) \times 4 = 74 + 24 = 98$

લીનિયર ઇકવેશન: $Y = 98 - 6X$

અર્થધટન: સ્માર્ટફોન ઉપયોગના દરેક વધારાના કલાક માટે, પરીક્ષા સ્કોર 6 પોઇન્ટ ઘટે છે.

મેમરી ટ્રીક

"મોર ફીન, લેસ સ્કોર" - ફોનના ઉપયોગ અને ગ્રેડ્સ વચ્ચે નેગેટિવ કોરિલેશન

પ્રશ્ન 4(ક) [7 ગુણ]

વર્ગીકરણના પગલાંને વિગતવાર સમજાવો

જવાબ

કલાસિફિકેશન એ સુપરવાઇઝ લર્નિંગ પ્રક્રિયા છે જે ઇનપુટ ડેટાને પૂર્વનિર્ધારિત કેટેગરીઓ અથવા કલાસોમાં સૌંપે છે.

વિગતવાર કલાસિફિકેશન સ્ટેપ્સ:

1. સમસ્યા વ્યાખ્યા

- કલાસો અને ઉદ્દેશ્યો વ્યાખ્યાયિત કરવા
- ઇનપુટ ફીચર્સ અને ટાર્ગેટ વેરિએબલ ઓળખવા
- સફળતાના માપદંડો નક્કી કરવા

2. ડેટા કલેક્શન અને તૈયારી

flowchart LR

```
A[ ] --{-{-}} B[ ]
B --{-{-}} C[ ]
C --{-{-}} D[ ]
D --{-{-}} E[ ]
E --{-{-}} F[ ]
```

3. ફીચર એન્જિનિયરિંગ

- ફીચર સિલેક્શન: સંબંધિત એટ્રિબ્યુટ્સ પસંદ કરવા
- ફીચર એક્સ્પ્રેક્શન: નવા અર્થપૂર્ણ ફીચર્સ બનાવવા
- નોર્મલાઇઝનાન: ફીચરને સમાન રેન્જમાં સ્કેલ કરવા

4. મોડેલ સિલેક્શન અને ટ્રેનિંગ

ટેબલ: સામાન્ય કલાસિફિકેશન અલ્ગોરિધમ્સ

અલ્ગોરિધમ	શ્રેષ્ઠ માટે	કાયદાઓ
ડિસિઝન ટ્રી	ઇન્ટરપ્રિટેબલ રૂલ્સ	સમજવામાં સરળ
SVM	હાઇ-ડાયમેન્શનલ ડેટા	સારાં જનરલાઇઝેશન
ન્યૂરલ નેટવર્ક્સ	જટિલ પેટન્સ	ઉચ્ચ ચોક્સાઈં
નાઈવ બેઝ	ટેક્સ્ટ કલાસિફિકેશન	જડપી ટ્રેનિંગ

5. મોડેલ ઇવેલ્યુઅશન

- કન્ફ્યુઝન મેટ્રિક્સ: વિગતવાર પરફોર્મન્સ એનાલિસિસ
- કોસ-વેલિડેશન: મજબૂત પરફોર્મન્સ અંદાજ
- મેટ્રિક્સ: એક્સ્પ્રેક્શન, પ્રિસિજન, રિકોલ, F1-સ્કોર

6. હાયપરપેરામીટર ટ્યુનિંગ

- આદર્શ પેરામીટર્સ માટે ગ્રિડ સર્ચ
- પેરામીટર સિલેક્શન માટે વેલિડેશન સેટ

7. અંતિમ ઇવેલ્યુઅશન અને ડિપ્લોયમેન્ટ

- અદ્દશ્ય ડેટા પર ટેસ્ટ કરવું
- પ્રોડક્શન ઉપયોગ માટે મોડેલ ડિપ્લોય કરવું
- સમય જતાં પરફોર્મન્સ મોનિટર કરવું

મેમરી ટ્રીક

"પ્રોપર ડેટા મોડેલિંગ ઇવેલ્યુઅટ્સ પર્ફોર્મન્સ થોરોલી" - પ્રોબ્લેમ ડેફિનિશન, ડેટા પ્રોપ, મોડેલિંગ, ઇવેલ્યુઅશન, પર્ફોર્મન્સ ટેસ્ટિંગ, ટ્યુનિંગ।

પ્રશ્ન 4(અ) OR [3 ગુણ]

શું K વેલ્યુની પસંદગી KNN અલ્ગોરિધમના પરફોર્મન્સને પ્રભાવિત કરે છે? ટૂંકમાં સમજાવો

જવાબ

હા, K વેલ્યુ ડિસિઝન બાઉન્ડરી અને મોડેલ જટિલતાને અસર કરીને KNN અલ્ગોરિધમના પરફોર્મન્સને નોંધપાત્ર રીતે પ્રભાવિત કરે છે.

ટેબલ: K વેલ્યુની અસર

K વેલ્યુ	અસર	પરફોર્મન્સ
નાનું K (k=1)	નોઇજ પ્રત્યે સંવેદનશીલ	હાઇ વેરિયન્સ, લો બાયસ
મધ્યમ K	સંતુલિત નિર્ણયો	આદર્શ પરફોર્મન્સ
મોટું K	સ્મૂથ બાઉન્ડરીઝ	લો વેરિયન્સ, હાઇ બાયસ

અસર એનાલિસિસ:

- $k=1$: ટ્રેનિંગ ડેટા પર ઓવરફિટ થઈ શકે, આઉટલાયર્સ પ્રત્યે સંવેદનશીલ
- આદર્શ k : સામાન્ય રીતે વિષમ સંખ્યા, બાયસ-વેરિયન્સ ટ્રેડઓફને સંતુલિત કરે
- મોટું k : અંડરફિટ થઈ શકે, સ્થાનિક પેટન્સ ગુમાવે

સિલેક્શન વ્યૂહરચના:

- આદર્શ k શોધવા માટે કોસ-વેલિડેશનનો ઉપયોગ
- શરૂઆતના બિંદુ તરીકે $k = \text{ટ્રાય કરો}$
- કોમ્પ્યુટેશનલ કોરસ VD ચોક્સાઈનો વિચાર કરો

મેમરી ટ્રીક

“સ્મોલ K વેરીઝન, લાર્જ K સ્મૂથ્સ” - નાનું k વેરિયન્સ બનાવે, મોટું k સ્મૂથ બાઉન્ડરીઝ બનાવે

પ્રશ્ન 4(બ) OR [4 ગુણ]

SVM મોડેલમાં સપોર્ટ વેક્ટર્સને વ્યાખ્યાયિત કરો

જવાબ

સપોર્ટ વેક્ટર્સ એ મહત્વપૂર્ણ ડેટા પોઇન્ટ્સ છે જે સપોર્ટ વેક્ટર મશીન અલ્ગોરિધમમાં ડિસિઝન બાઉન્ડરી (હાયપરપ્લેન)ની સૌથી નજીક આવેલા હોય છે.

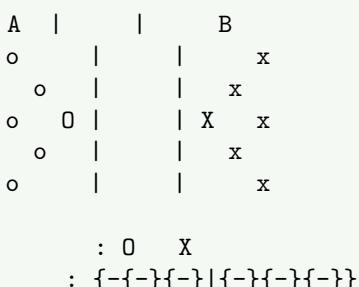
ટેબલ: સપોર્ટ વેક્ટર લક્ષણો

પાસું	વર્ણન	મહત્વ
સ્થાન	હાયપરપ્લેનની સૌથી નજીકના પોઇન્ટ્સ	ડિસિઝન બાઉન્ડરી વ્યાખ્યાયિત કરે
અંતર	બાઉન્ડરીથી સમાન અંતર	મેન્ડિસમ્મ માર્જિન
ભૂમિકા	હાયપરપ્લેનને સપોર્ટ કરે	આદર્શ વિભાજન નક્કી કરે
સંવેદનશીલતા	તેમને રીમૂવ કરવાથી મોડેલ બદલાય	મોડેલ સ્ટ્રક્ચર માટે મહત્વપૂર્ણ

મુખ્ય ગુણધર્મો:

- માર્જિન ડિફીનીશન: સપોર્ટ વેક્ટર્સ કલાસો વચ્ચે મેન્ડિસમ્મ માર્જિન નક્કી કરે છે
- મોડેલ ડિપેન્ડન્સી: માત્ર સપોર્ટ વેક્ટર્સ જ અંતિમ મોડેલને અસર કરે છે
- બાઉન્ડરી ફોર્મેશન: આદર્શ વિભાજક હાયપરપ્લેન બનાવે છે

ડાયાગ્રામ:



ગાણિતિક મહત્વ: સપોર્ટ વેક્ટર્સ $y_i(w \cdot x_i + b) = 1$ કન્સ્ટ્રેઇન્ટને સંતુષ્ટ કરે છે, જ્યાં તેઓ માર્જિન બાઉન્ડરી પર બરાબર સ્થિત હોય છે.

મેમરી ટ્રીક

“સપોર્ટ વેક્ટર્સ સપોર્ટ ડિસિઝન્સ” - આ વેક્ટર્સ ડિસિઝન બાઉન્ડરીને સપોર્ટ કરે છે

પ્રશ્ન 4(ક) OR [7 ગુણ]

લોજિસ્ટિક રિગ્રેશનને વિગતવાર સમજાવો

જવાબ

લોજિસ્ટિક રિગ્રેશન એ બાઇનરી કલાસિફિકેશન માટે વપરાતી સ્ટેટિસ્ટિકલ મેથડ છે જે લોજિસ્ટિક ફંક્શનનો ઉપયોગ કરીને કલાસ મેભરશિપની સંભાવનાને મોડેલ કરે છે.

ગાળિતિક આધાર:

સિગ્મોઇડ ફંક્શન: $\sigma(z) = 1 / (1 + e^{-z})$ જ્યાં $z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n$

ટેબલ: લીનિયર VS લોજિસ્ટિક રિગ્રેશન

પાસું	લીનિયર રિગ્રેશન	લોજિસ્ટિક રિગ્રેશન
આઉટપુટ	સતત મૂલ્યો	સંભાવનાઓ (0-1)
ફંક્શન	લીનિયર	સિગ્મોઇડ (S-કર્વ)
હેતુ	આગાહી	કલાસિફિકેશન
ઓરર ફંક્શન	મીન સ્કવર્ડ એરર	લોગ-લાઇકલીહુડ

મુખ્ય ઘટકો:

1. લોજિસ્ટિક ફંક્શન ગુણાધમો:

- S-આકારનો કર્વ: 0 અને 1 વચ્ચે સ્મૂથ ટ્રાન્ઝિશન
- એસિમ્પોટસ: 0 અને 1ની નજીક પહોંચે પણ કયારેય પહોંચાતું નથી
- મોનોટોનિક: હમેશા વધતું ફંક્શન

2. મોડેલ ટ્રેનિંગ:

- મેક્સિમમ લાઇકલીહુડ ઓસ્ટેમેશન: જોયેલા ડેટાની સંભાવના વધારતા પેરામીટર્સ શોંધવા
- ગ્રેડિન્ટ ડિસેન્ટ: પુનરાવર્તક ઓટિમાઇઝેશન અલ્ગોરિધમ
- કોસ્ટ ફંક્શન: લોગ-લોસ અથવા ક્રોસ-એન્ટ્રોપી

3. નિર્ણય લેવું:

- થ્રોલ્ડ: બાઇનરી કલાસિફિકેશન માટે સામાન્ય રીતે 0.5
- પ્રોબેબિલિટી આઉટપુટ: $P(y=1 | x)$ કલાસ સંભાવના આપે છે
- ડિસ્ઝિન રૂલ: $P(y=1 | x) > 0.5$ હોય તો પોઝિટિવ તરીકે કલાસિફાય કરવું

ફાયદાઓ:

- પ્રોબેબિલિટિક આઉટપુટ: આગાહીઓમાં વિશ્વાસ પ્રદાન કરે છે
- કોઈ ધારણાઓ નથી: ઇન્ડિપેન્ડન્ટ વેરિએબલ્સના વિતરણ વિશે
- ઓછું ઓવરફિટિંગ: જટિલ મોડેલ્સની તુલનામાં
- કડપી ટ્રેનિંગ: કાર્યક્ષમ કોમ્પ્યુટેશન

એપ્લિકેશન્સ:

- મેડિકલ ડાયગ્નોસ્ટિસ
- માર્કાટિંગ રિસ્પોન્સ આગાહી
- કેડિટ એપ્પ્લોવલ નિર્ણયો
- ઇમેઇલ સ્પામ ડિટેક્શન

કોડ ઉદાહરણ:

```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model.fit(X_train, y_train)
predictions = model.predict(X_test)
probabilities = model.predict_proba(X_test)
```

મેમરી ટ્રીક

"સિગ્મોઇડ સ્કવેશોસ ઇન્ફિનિટ ઇનપુટ" - સિગ્મોઇડ ફંક્શન કોઈપણ વાસ્તવિક સંખ્યાને સંભાવનામાં કન્વર્ટ કરે છે

પ્રશ્ન 5(અ) [3 ગુણ]

Matplotlib python library પર ટૂંકી નોંધ લખો

જવાબ

Matplotlib એ ડેટા સાયન્સ અને મશીન લર્નિંગમાં સ્ટેટિક, એનિમેટેડ અને ઇન્ટરેક્ટિવ વિજ્યુઅલાઇડેશન બનાવવા માટેની વ્યાપક Python લાઇબ્રેરી છે.

ટેબલ: Matplotlib મુખ્ય ફીચર્સ

ફીચર	હેતુ	ઉદાહરણ
Pyplot	MATLAB-જેવું પ્લોટિંગ ઇન્ટરફેસ	લાઇન પ્લોટ્સ, સ્કેટર પ્લોટ્સ
Object-oriented	એડવાન્સડ કસ્ટમાઇઝેશન	ફિગર અને એક્સેસ ઓફજેક્ટ્સ
મલિટપલ ફોર્મેટ્સ	વિવિધ ફોર્મેટમાં સેવ કરવું	PNG, PDF, SVG, EPS
સબપ્લોટ્સ	એક ફિગરમાં મલિટપલ પ્લોટ્સ	ગ્રિડ એરેન્જમેન્ટ્સ

સામાન્ય પ્લોટ પ્રકારો:

- લાઇન પ્લોટ: સમય પર વલણો
- સ્કેટર પ્લોટ: વેરિએબલ્સ વરચે સંબંધ
- હિસ્ટોગ્રામ: ડેટા વિતરણ
- બાર ચાર્ટ: કેટેગોરિકલ કમ્પેરિઝન્સ
- બોક્સ પ્લોટ: સ્ટેટિસ્ટિકલ સમરીજ

મૂળભૂત ઉપયોગ:

```
import matplotlib.pyplot as plt
plt.plot(x, y)
plt.xlabel({X })
plt.ylabel({Y })
plt.title({ })
plt.show()
```

એપ્લિકેશન્સ: ડેટા એક્સપ્લોરેશન, મોડેલ પરફોર્મન્સ વિઝ્યુઅલાઇઝેશન, પ્રોજ૆ક્ટેશન ગ્રાફિક્સ

મેમરી ટ્રીક

"Matplotlib મેક્સ પ્રિટી પ્લોટ્સ" - ડેટા વિઝ્યુઅલાઇઝેશન માટે આવશ્યક ટૂલ

પ્રશ્ન 5(બ) [4 ગુણ]

દ્વિ-પરિમાણીય ડેટા માટે K-means કલસ્ટરિંગ

જવાબ

આપેલ પોઇન્ટ્સ: {(2,3),(3,3),(4,3),(5,3),(6,3),(7,3),(8,3),(25,20),(26,20),(27,20),(28,20),(29,20),(30,20)}

K-means અલોરિધમ સ્ટેપ્સ:

સ્ટેપ 1: સેન્ટ્રોઇડ્સ ઇનિશિયલાઇઝ કરવા

- કલસ્ટર 1: (4, 3) - ડાબા જૂથમાંથી પસંદ કરેલું
- કલસ્ટર 2: (27, 20) - જમણા જૂથમાંથી પસંદ કરેલું

સ્ટેપ 2: નજીકના સેન્ટ્રોઇડ્સ પોઇન્ટ્સ સૌંપવા

ટેબલ: પોઇન્ટ એસાઇનમેન્ટ્સ

પોઇન્ટ	C1નું અંતર	C2નું અંતર	સોંપેલ કલસ્ટર
(2,3)	2.0	25.8	કલસ્ટર 1
(3,3)	1.0	24.8	કલસ્ટર 1
(4,3)	0.0	23.8	કલસ્ટર 1
(5,3)	1.0	22.8	કલસ્ટર 1
(6,3)	2.0	21.8	કલસ્ટર 1
(7,3)	3.0	20.8	કલસ્ટર 1
(8,3)	4.0	19.8	કલસ્ટર 1
(25,20)	23.8	2.0	કલસ્ટર 2
(26,20)	24.8	1.0	કલસ્ટર 2
(27,20)	25.8	0.0	કલસ્ટર 2
(28,20)	26.8	1.0	કલસ્ટર 2
(29,20)	27.8	2.0	કલસ્ટર 2
(30,20)	28.8	3.0	કલસ્ટર 2

સ્ટેપ 3: સેન્ટ્રોઇડસ અપડેટ કરવા

- નવું C1 = $((2+3+4+5+6+7+8)/7, (3+3+3+3+3+3)/7) = (5, 3)$
- નવું C2 = $((25+26+27+28+29+30)/6, (20+20+20+20+20+20)/6) = (27.5, 20)$

અંતિમ કલસ્ટર્સ:

- કલસ્ટર 1: {(2,3),(3,3),(4,3),(5,3),(6,3),(7,3),(8,3)}
- કલસ્ટર 2: {(25,20),(26,20),(27,20),(28,20),(29,20),(30,20)}

મેમરી ટ્રીક

"સેન્ટ્રોઇડસ એટ્રોક્ટ નિયરેસ્ટ નેબર્સ" - પોઇન્ટ્સ નજીકના સેન્ટ્રોઇડમાં જોડાય છે

પ્રશ્ન 5(ક) [7 ગુણ]

Scikit-learn ના ફંક્શન્સ અને તેનો ઉપયોગ આપો: a. ડેટા પ્રીપ્રોસેસિંગ b. મોડેલ સિલેક્શન c. મોડેલ ઇવેલ્યુઅશન અને મેટ્રિક્સ

જવાબ

Scikit-learn ડેટા પ્રીપ્રોસેસિંગથી લઈને મોડેલ ઇવેલ્યુઅશન સુધીના મશીન લર્નિંગ વર્કફ્લો માટે વ્યાપક સાધનો પ્રદાન કરે છે.

a) ડેટા પ્રીપ્રોસેસિંગ ફંક્શન્સ:

ટેબલ: પ્રીપ્રોસેસિંગ ફંક્શન્સ

ફંક્શન	હેતુ	ઉદાહરણ ઉપયોગ
StandardScaler()	ફીચર્સને નોર્મલાઇઝ કરવા	મીન દૂર કરવું, યુનિટ વેરિયન્સ
MinMaxScaler()	[0,1] રેન્જમાં સ્કેલ કરવું	ફીચર સ્કેલિંગ
LabelEncoder()	કેટેગોરિકલ લેબલ્સ એન્કોડ કરવા	ટેક્સ્ટને નંબરમાં કન્વર્ટ કરવું
OneHotEncoder()	ડમી વેરિએબલ્સ બનાવવા	કેટેગોરિકલ ફીચર્સ હેન્ડલ કરવા
train_test_split()	ડેટાસેટ સ્પિલટ કરવું	ટ્રેનિંગ/ટેસ્ટિંગ વિભાજન

કોડ ઉદાહરણ:

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X\_scaled = scaler.fit\_transform(X)
```

b) મોડેલ સિલેક્શન ફંક્શન્સ:

ટેબલ: મોડેલ સિલેક્શન ટૂલ્સ

ફંક્શન	હેતુ	અપ્લિકેશન
GridSearchCV()	હાયપરપેરામીટર ટ્યુનિંગ	આદર્શ પેરામીટર્સ શોધવા
RandomizedSearchCV()	રેન્ડમ પેરામીટર સર્ચ	જડપી પેરામીટર ઓપ્ટિમાઇઝેશન
cross_val_score()	કોસ-વેલિડેશન	મોડેલ પરફોર્મન્સ ઇવેલ્યુએશન
StratifiedKFold()	સ્ટ્રેટિકફોલ્ડ સેમ્પલિંગ	સંતુલિત કોસ-વેલિડેશન
Pipeline()	પ્રીપોરેસિંગ અને મોડેલિંગ ભેગું કરવું	સ્ટ્રીમલાઇન વર્કફ્લો

કોડ ઉદાહરણ:

```
from sklearn.model\_selection import GridSearchCV
param\_grid = \{{'C': [0.1, 1, 10]}\}
grid\_search = GridSearchCV(SVM(), param\_grid, cv=5)
grid\_search.fit(X\_train, y\_train)
```

c) મોડેલ ઇવેલ્યુએશન અને મેટ્રિક્સ ફંક્શન્સ:

ટેબલ: ઇવેલ્યુએશન મેટ્રિક્સ

ફંક્શન	હેતુ	વપરાશ કેસ
accuracy_score()	એકેંદર ચોક્સાઈં	સામાન્ય કલાસિફિકેશન
precision_score()	પોઝિટિવ પ્રિડિક્શન ચોક્સાઈં	ફોલ્સ પોઝિટિવ્સ ઘટાડવા
recall_score()	ટૂ પોઝિટિવ રેટ	ફોલ્સ નેગેટિવ્સ ઘટાડવા
f1_score()	પ્રિસિઝન/રિકોલનું હાર્મોનિક મીન	સંતુલિત મેટ્રિક
confusion_matrix()	વિગતવાર એરર એનાલિસિસ	ભૂલો સમજવી
classification_report()	વ્યાપક મેટ્રિક્સ	સંપૂર્ણ મૂલ્યાંકન
roc_auc_score()	ROC કર્વ હંડળનો વિસ્તાર	બાઇનરી કલાસિફિકેશન

કોડ ઉદાહરણ:

```
from sklearn.metrics import classification\_report
print(classification\_report(y\_true, y\_pred))
```

વર્કફ્લો ઇન્ટિગ્રેશન:

- પ્રીપોરેસિંગ: ડેટાને સાફ્ અને તૈયાર કરવું
- મોડેલ સિલેક્શન: અલ્ગોરિધમ્સ પરાંદ કરવા અને ટ્યુન કરવા
- ઇવેલ્યુએશન: પરફોર્મન્સનું વ્યાપક આકારણી

મેમરી ટ્રીક

“પ્રીપોરેસસ, સિલેક્ટ, ઇવેલ્યુએટ” - Scikit-learn માં સંપૂર્ણ ML વર્કફ્લો

પ્રશ્ન 5(અ) OR [3 ગુણ]

NumPy ના મુખ્ય ફીચર્સની યાદી બનાવો

જવાબ

NumPy (Numerical Python) Python માં વૈજ્ઞાનિક કોમ્પ્યુટિંગ માટેનું મૂળભૂત પેકેજ છે, જે શક્તિશાળી એરે ઓપરેશન્સ અને ગાણિતિક ફંક્શન્સ પ્રદાન કરે છે.

ટેબલ: NumPy ના મુખ્ય ફીચર્સ

ફીચર	વર્ણન	કાયદો
N-dimensional Arrays	કાર્યક્રમ એરે ઓફજેક્ટ્સ	જડપી ગાણિતિક ઓપરેશન્સ
Broadcasting	વિવિધ સાઇઝના એરે પર ઓપરેશન્સ	લવચીક કોમ્પ્યુટેશન્સ
Linear Algebra	મેટ્રિક્સ ઓપરેશન્સ, ડીક્રોઝિશન્સ	વૈજ્ઞાનિક કોમ્પ્યુટિંગ
Random Numbers	રેન્ડમ સેમ્પલિંગ અને ડિસ્ટ્રિબ્યુશન્સ	સ્ટેટિસ્ટિકલ સિમ્યુલેશન્સ
Integration	C/C++/Fortran સાથે કામ કરે છે	ઉત્ત્ચ પરફોર્મન્સ

મુખ્ય ક્ષમતાઓ:

- ગાણિતિક ફંક્શન્સ: નિકોલાભિતિ, લોગેરિધમિક, એક્સપોનેન્શિયલ
- એરે મેન્યુલેશન: રિશેપિંગ, સ્પલાયરિંગ, જોઇનિંગ એરેજ
- ઇન્ડેક્સિંગ: એડવાન્સ્ડ સ્લાઇસિંગ અને બૂલિયન ઇન્ડેક્સિંગ
- મેમરી કાર્યક્રમતા: ઓપ્ટિમાઇઝ ડેટા સ્ટોરેજ

એપ્લિકેશન્સ: ડેટા એનાલિસિસ, મશીન લર્નિંગ, ઇમેજ પ્રોસેસિંગ, વૈજ્ઞાનિક સંશોધન

મેમરી ટ્રીક

"નંબર્સ નીડ NumPy's પાવર" - ન્યુમેરિકલ કોમ્પ્યુટેશન્સ માટે આવશ્યક

પ્રશ્ન 5(બ) OR [4 ગુણ]

એક-પરિમાણીય ડેટા માટે K-means કલસ્ટરિંગ

જવાબ

આપેલ ડેટાસેટ: {1,2,4,5,7,8,10,11,12,14,15,17}

3 કલસ્ટર્સ માટે K-means અભ્યોરિધમ:

સ્ટેપ 1: સેન્ટ્રોઇડ્સ ઇનિશિયલાઇઝ કરવા

- C1 = 3 (પ્રારંભિક મૂલ્યોની આસપાસ)
- C2 = 9 (મધ્યમ મૂલ્યોની આસપાસ)
- C3 = 15 (પછીના મૂલ્યોની આસપાસ)

સ્ટેપ 2: નજીકના સેન્ટ્રોઇડ્સ પોઇન્ટ્સ સૌંપવા

ટેબલ: પોઇન્ટ એસાઇનમેન્ટ્સ (ઇટરેશન 1)

પોઇન્ટ	C1નું અંતર	C2નું અંતર	C3નું અંતર	સૌંપેલ કલસ્ટર
1	2	8	14	કલસ્ટર 1
2	1	7	13	કલસ્ટર 1
4	1	5	11	કલસ્ટર 1
5	2	4	10	કલસ્ટર 1
7	4	2	8	કલસ્ટર 2
8	5	1	7	કલસ્ટર 2
10	7	1	5	કલસ્ટર 2
11	8	2	4	કલસ્ટર 2
12	9	3	3	કલસ્ટર 2
14	11	5	1	કલસ્ટર 3
15	12	6	0	કલસ્ટર 3
17	14	8	2	કલસ્ટર 3

સ્ટેપ 3: સેન્ટ્રોઇડ્સ અપડેટ કરવા

- નવું C1 = $(1+2+4+5)/4 = 3$
- નવું C2 = $(7+8+10+11+12)/5 = 9.6$
- નવું C3 = $(14+15+17)/3 = 15.33$

અંતિમ કલસ્ટર્સ:

- કલસ્ટર 1: {1, 2, 4, 5}
- કલસ્ટર 2: {7, 8, 10, 11, 12}
- કલસ્ટર 3: {14, 15, 17}

પ્રશ્ન 5(ક) OR [7 ગુણ]

Pandas library ના ફંક્શન્સ અને તેનો ઉપયોગ આપો: a. ડેટા પ્રીપ્રોસેસિંગ b. ડેટા ઇન્સ્પેક્શન c. ડેટા કલીનિંગ અને ટ્રાન્સફોર્મેશન

જવાબ

Pandas ડેટા મેનિપ્યુલેશન અને એનાલિસિસ માટેની શક્તિશાળી Python વાઈભેરી છે, જે ઉચ્ચ-સ્તરના ડેટા સ્ટ્રક્ચર્સ અને ઓપરેશન્સ પ્રદાન કરે છે.

a) ડેટા પ્રીપ્રોસેસિંગ ફંક્શન્સ:

ટેબલ: પ્રીપ્રોસેસિંગ ફંક્શન્સ

ફંક્શન	હેતુ	ઉદાહરણ
read_csv()	CSV ફાઈલો લોડ કરવા	pd.read_csv('data.csv')
head()	પ્રથમ n રોઝ જોવા	df.head(10)
tail()	છેલ્લા n રોઝ જોવા	df.tail(5)
sample()	રેન્ડમ સેમ્પલિંગ	df.sample(100)
set_index()	કોલમને ઇન્ડેક્સ તરીકે સેટ કરવું	df.set_index('id')

b) ડેટા ઇન્સ્પેક્શન ફંક્શન્સ:

ટેબલ: ઇન્સ્પેક્શન ફંક્શન્સ

ફંક્શન	હેતુ	પ્રદાન કરેલી માહિતી
info()	ડેટાસેટ ઓવરવ્યુ	ડેટા ટાઇપ્સ, મેમરી વપરાશ
describe()	સ્ટેટિસ્ટિકલ સમરી	મીન, std, min, max
shape	ડેટાસેટ ડાયમેન્શન્સ	(રોઝ, કોલમ્સ)
dtypes	ડેટા ટાઇપ્સ	કોલમ ડેટા ટાઇપ્સ
isnull()	મિસિંગ વેલ્ચુઝ	નલ્સ માટે બૂલિયન માર્ક
value_counts()	યુનિક વેલ્ચુઝ કાઉન્ટ કરવા	ફીકવન્સી ડિસ્ટ્રિબ્યુશન
corr()	કોરિલેશન મેટ્રિક્સ	ફીચર રિલેશનશિપ્સ

કોડ ઉદાહરણ:

```
\#
print(df.info())
print(df.describe())
print(df.isnull().sum())
```

c) ડેટા કલીનિંગ અને ટ્રાન્સ્ફરેશન ફંક્શન્સ:

ટેબલ: કલીનિંગ ફંક્શન્સ

ફંક્શન	હેતુ	વપરાશ
dropna()	મિસિંગ વેલ્ચુઝ રીમૂવ કરવા	df.dropna()
fillna()	મિસિંગ વેલ્ચુઝ ભરવા	df.fillna(0)
drop_duplicates()	ડુલિકેટ રોજ રીમૂવ કરવા	df.drop_duplicates()
replace()	વેલ્ચુઝ રિપ્લેસ કરવા	df.replace('old', 'new')
astype()	ડેટા ટાઇપ્સ બદલવા	df['col'].astype('int')
apply()	ડેટા પર ફંક્શન એપ્લાય કરવું	df.apply(lambda x: x*2)
groupby()	ડેટા ગ્રૂપ કરવું	df.groupby('category')
merge()	ડેટાસેટ્સ જોઈન કરવા	pd.merge(df1, df2)
pivot()	ડેટા રિશેપ કરવું	df.pivot(columns='col')

એડવાન્સ્ડ ઓપરેશન્સ:

- સ્ટ્રિંગ ઓપરેશન્સ: str.contains(), str.replace()
- તારીખ ઓપરેશન્સ: to_datetime(), dt.year
- કેટેગોરિકલ ડેટા: pd.Categorical()

વર્કફ્લો ઉદાહરણ:

```
\#
df = pd.read\_\_csv({data.csv})
df = df.dropna()
df[\{category\}] = df[\{category\}].astype(\{category\})
df\_\_grouped = df.groupby(\{type\}).mean()
```

ફાયદાઓ:

- સહજ સિન્કેક્સ: શીખવા અને વાપરવામાં સરળ
- પરકોર્મન્સ: મોટા ડેટાસેટ્સ માટે ઓપ્ટિમાઇઝડ
- ઇન્ફ્રાશેર્ટ: NumPy, Matplotlib સાથે સારી રીતે કામ કરે છે
- લવચીકતા: વિવિધ ડેટા ફોર્મેટ્સ હેન્ડલ કરે છે

મેમરી ટ્રીક

"Pandas પ્રોસેસેસ ડેટા પર્ફેક્ટલી" - વ્યાપક ડેટા મેનિપ્યુલેશન ટૂલ