টপিক

Numerical ডেটাকে Category ডেটায় রূপান্তর করা
Feature Engineering-এর ভাষায় একে Discretization বা Binning বলা হয়
এর বিশেষ কেস হলো Binarization, যেখানে আউটপুট শুধু 0 বা 1

1) **Discretization** বা **Binning** কী

Continuous numerical value কে কিছু নির্দিষ্ট interval বা bin-এ ভাগ করা
ফলে সংখ্যাটা সরাসরি ব্যবহার না করে সেটাকে একটি category হিসেবে দেখা হয়

উদাহরণ
Age: 4, 17, 23, 45, 62, 80
Binning করলে হতে পারে
0–10 Child
10–20 Teen
20–40 Adult
40+ Senior

এখন Age আর raw number না, বরং একটি label বা category

2) কেন **Binning** করা হয়

Outlier হ্যান্ডেল করতে

Outlier মানে অস্বাভাবিক বড় বা ছোট মান
যেমন Titanic ডেটায় Fare = 512
Raw numeric রাখলে মডেল এই extreme মানকে বেশি গুরুত্ব দিতে পারে
Binning করলে 512 একটি High Fare টাইপ bin-এ পড়ে যায়
ফলে negative influence কমে

ডেটার **distribution** বা **spread** উন্নত করতে

ডেটা অনেক সময় skewed থাকে
Quantile binning করলে প্রতিটি bin-এ প্রায় সমান সংখ্যক ডেটা পড়ে
ফলে model stable signal পেতে পারে

কিছু মডেলে শেখা সহজ করতে

কিছু ক্ষেত্রে raw numeric সম্পর্ক non-linear হয়
Bins বানালে feature একধরনের step-like behavior তৈরি করে
এতে group-based pattern ধরতে সুবিধা হতে পারে

তবে trade-off আছে
Binning করলে precision কিছুটা কমে, কারণ exact value হারিয়ে যায়

3) Binning-এর ৩টি জনপ্রিয় কৌশল

3.1 Equal Width বা Uniform Binning

পুরো range কে সমান প্রস্থে ভাগ করা

Formula
Bin width = (max - min) / number of bins

উদাহরণ
Age min=0, max=80, bins=4
width = (80-0)/4 = 20
Bins: 0–20, 20–40, 40–60, 60–80

কখন ভালো
ডেটা যদি মোটামুটি evenly spread হয়

কখন সমস্যা
ডেটা যদি skewed হয়, অনেক ডেটা এক bin-এ গাদাগাদি হয়ে যায়

3.2 Equal Frequency বা Quantile Binning

প্রতিটি bin-এ প্রায় সমান সংখ্যক data point রাখার চেষ্টা

উদাহরণ
100 data, bins=4
প্রতি bin-এ প্রায় 25টা করে data

কখন ভালো
Outlier ও skewed distribution-এর ক্ষেত্রে ভালো কাজ করে
Distribution balanced হয়

কখন সীমাবদ্ধতা
Bins-এর width সমান থাকে না
কোথাও ছোট range, কোথাও বড় range হতে পারে

3.3 KMeans Binning

KMeans clustering ব্যবহার করে natural group অনুযায়ী bins তৈরি

কখন ভালো
ডেটা যদি naturally cluster তৈরি করে বা multi-modal হয়

কখন সীমাবদ্ধতা
ফল dataset এবং initialization-এর উপর sensitive হতে পারে
Interpretability অনেক সময় কমে

4) Custom Binning

Domain knowledge ব্যবহার করে নিজের মতো bins বানানো

উদাহরণ
Age < 18 Child
18–60 Adult

60 Retired

সুবিধা
Interpretability বেশি
Business rule অনুযায়ী align করা যায়

ঝুঁকি
ভুল rule হলে bias বা ভুল decision boundary তৈরি হতে পারে

5) Binarization

Discretization-এর বিশেষ রূপ
আউটপুট শুধু 2টা class: 0 এবং 1

Mechanism
একটা threshold সেট করা হয়
x ≤ threshold হলে 0
x > threshold হলে 1

উদাহরণ
Income > 6 lakh হলে Tax payer = 1
নাহলে 0

Use case
Image processing-এ grayscale থেকে black-white conversion
Simple and fast, তবে threshold ভুল হলে signal নষ্ট হতে পারে

6) Scikit-learn দিয়ে ধারণাগতভাবে কী করা হয়

KBinsDiscretizer

Key parameters
n_bins: কয়টা bin
strategy: uniform, quantile, kmeans
encode: ordinal বা onehot

encode কেন গুরুত্বপূর্ণ
ordinal হলে bin label 0,1,2 হিসেবে আসে
onehot হলে প্রতিটি bin আলাদা feature column হয়
অনেক মডেলে onehot উপকারী হতে পারে

Binarizer

threshold সেট করলেই 0/1 তৈরি

7) Titanic ডেটায় accuracy কেন বাড়তে পারে

Age এবং Fare অনেক সময় non-linearভাবে target-এর সাথে সম্পর্কিত
যেমন মাঝারি বনাম খুব বেশি Fare এই পার্থক্যটা বেশি গুরুত্বপূর্ণ হতে পারে
Age-এও child, adult, senior গ্রুপভিত্তিক pattern কাজ করতে পারে
Binning করলে এই group signal স্পষ্ট হয়
তাই কখনো performance উন্নতি দেখা যায়

8) কখন ব্যবহার করবে, কখন এড়াবে

ব্যবহার করা ভালো যখন

ডেটা skewed বা outlier-heavy
Interpretability দরকার
Non-linear relation সন্দেহ হচ্ছে
ডেটা ছোট এবং noisy

এড়ানো ভালো যখন

Exact numeric precision দরকার, যেমন regression price prediction
Feature খুব smooth এবং ইতিমধ্যেই predictive
Bins ভুল হলে performance drop করতে পারে