

## অধ্যায় ১৩ : পারফরম্যান্স (Performance)

আমরা এ পর্যন্ত যতগুলো মেশিন লার্নিং অ্যালগরিদম দেখলাম, সবগুলোই তিনটি ভাগের মধ্যে কোনো একটিতে পড়ে –

- রিগ্রেশন : লিনিয়ার রিগ্রেশন
- ক্লাসিফিকেশন : লজিস্টিক রিগ্রেশন, সাপোর্ট ভেস্ট মেশিন, কে-নিয়ারেস্ট নেইবরস, নাইভ বেইজ ক্লাসিফায়ার, ডিসিশন ট্রি, পারসেপ্ট্রন এবং নিউরাল নেটওয়ার্ক
- ক্লাস্টারিং : কে-মিনস ক্লাস্টারিং

এ ছাড়াও এর বাইরে, ডেটার ডাইমেনশন কমিয়ে আনার জন্য আমরা দেখিয়েছি প্রিসিপাল কম্পোনেন্ট অ্যানালাইসিস।

আমাদের বইতে আমরা এতক্ষণ পর্যন্ত শুধু দেখেছি অ্যালগরিদমগুলো কীভাবে ডেটাসেটের ওপরে ব্যবহার করতে হয়। কিন্তু, অ্যালগরিদমগুলো আদৌ ভালো কাজ করছে কি না, অথবা করলেও কতটুকু ভালো কাজ করছে তা-ও তো বোঝা দরকার। কিংবা ধরি, যদি ডেটাসেটের ওপরে একাধিক ক্লাসিফিকেশন অ্যালগরিদম আমরা প্রয়োগ করি, তাহলে কোন অ্যালগরিদম সবচেয়ে ভালো কাজ করবে সেটিও আমাদের বুঝতে হবে, তাই না?

এজন্য বিভিন্ন অ্যালগরিদমের পারফরম্যান্স নির্ণয় করার জন্য কিছু পারফরম্যান্স মেট্রিক (Performance Metric) আছে, এদের সম্পর্কে আমাদের জানা প্রয়োজন। নিচে এদেরকে একটি টেবিল আকারে দেওয়া হলো :

অ্যালগরিদম টাইপ	Performance Metrics
রিগ্রেশন	R - Squared Value
ক্লাসিফিকেশন	Confusion Matrix, Accuracy, Precision, Recall, Specificity, F1 Measure, ROC Curve
ক্লাস্টারিং	Elbow Method

টেবিল 13.1

এখানে উল্লেখ্য, Elbow Method ঠিক কোনো পারফরম্যান্স মেট্রিক নয়, বরং কে-মিনস ক্লাস্টারিং-এর K অর্থাৎ ক্লাস্টার সংখ্যার অপটিমাল মান বের করার পদ্ধতি।

এটি কে-মিনস ক্লাস্টারিং-এর অধ্যায়ের সঙ্গে জুড়ে না দিয়ে আলাদা করে দেওয়ার পেছনে কারণ হচ্ছে, যাতে একটু আলাদা করে চোখে পড়ে। পদ্ধতিটি একটু ভিন্ন এবং অত্যন্ত গুরুত্বপূর্ণ। তাই এটি এই অধ্যায়ে আলোচনা করা হয়েছে।

তো শুরু করা যাক!

### পরিচেদ ১৩.১ : আর-স্কয়ারড ভ্যালু (R-Squared Value)

আমাদের প্রথম কে-মিনস ক্লাস্টারিং হলো আর-স্কয়ারড ভ্যালু (R-Squared Value)। এটি মূলত ব্যবহার করা হয় কোনো রিপ্রেশন মডেল কত ভালোভাবে কাজ করছে, সেটি নির্ণয় করার জন্য। এই R-Squared-এর মান যত বড়ো হবে, বুঝতে হবে যে আমাদের রিপ্রেশন মডেল তত ভালো কাজ করেছে, কিংবা ডেটাতে তত ভালো ফিট করেছে।

এটি বের করার সূত্র হচ্ছে –

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

এখানে  $\hat{y}_i$ ,  $\bar{y}$  এবং  $y_i$  হচ্ছে যথাক্রমে আমাদের মডেলের আন্দাজ করা প্রতিটি মান, সমষ্টি প্রকৃত আউটপুট মানের গড়মান এবং প্রতিটি প্রকৃত আউটপুটের মান।

যেমন ধরি, আমাদের রিপ্রেশন মডেল একবার চালানোর পরে আমরা পাই :

অরিজিনাল আউটপুট, $y_i$	$y_i - \bar{y}$	$(y_i - \bar{y})^2$	মডেলের দেওয়া আউটপুট, $\hat{y}$	$\hat{y} - \bar{y}$	$(\hat{y} - \bar{y})^2$
2	-2	4	2.8	-1.2	1.44
4	0	0	3.4	-0.6	0.36
5	1	1	4	0	0
4	0	0	4.6	0.6	0.36
5	1	1	5.2	1.2	1.44
$\bar{y} = 4$		$\sum_{i=1}^n (y_i - \bar{y})^2 = 6$	$\sum_{i=1}^n (\hat{y}_i - \bar{y})^2 = 3.6$		

টেবিল 13.1.1

### অধ্যায় ১৩ : পারফরম্যান্স (Performance)

সুতরাং, এই ক্ষেত্রে,  $R^2 = \frac{3.6}{6} = 0.6 = 60\%$  যেটি কিনা কিছুটা মধ্যমগোছের পারফরম্যান্স। এই মান ১-এর যত কাছাকাছি যাবে, আমরা ধরে নেব, আমাদের মডেল তত নিখুঁতভাবে মান আন্দজ করতে পারছে অর্থাৎ মডেলের অ্যাকুরেসি (Accuracy) তত ভালো হচ্ছে।

### পরিচ্ছেদ ১৩.২ : কনফিউশন ম্যাট্রিক্স (Confusion Matrix)

প্রথমেই বলে রাখি, কনফিউশন ম্যাট্রিক্স (Confusion Matrix) নিজে কোনো পারফরম্যান্স মেট্রিক নয়, কিন্তু ক্লাসিফিকেশনের জন্য বাকি যতগুলো পারফরম্যান্স মেট্রিক আছে, সবগুলোই তৈরি হয় এই কনফিউশন ম্যাট্রিক্স থেকে, তাই এটি বোবা জরুরি।

ধরুন, একজন রোগীর সম্পর্কে বিভিন্ন তথ্য আমরা কোনো মেশিন লার্নিং অ্যালগরিদমকে ইনপুট দিয়েছি এবং সেই অ্যালগরিদমের কাজ হচ্ছে আমাদের দেওয়া তথ্যের ওপরে ভিত্তি করে এটি প্রেডিষ্ট বা নির্ণয় করা যে, ওই রোগীর ক্যানসার আছে কি না। এখন চিন্তা করে দেখুন, এখানে চার ধরনের পরিস্থিতির উভব হতে পারে :

True Positive	False Positive
অ্যালগরিদম বলেছে রোগীর ক্যানসার আছে এবং সত্যিই তাই	অ্যালগরিদম বলেছে রোগীর ক্যানসার আছে কিন্তু আসলে নেই
False Negative	True Negative
অ্যালগরিদম বলেছে রোগীর ক্যানসার নেই, কিন্তু আসলে আছে	অ্যালগরিদম বলেছে রোগীর ক্যানসার নেই এবং সত্যিই তাই

টেবিল 13.2.1

এখন, আরেকটু গুছিয়ে লিখলে ওপরের চার্টটি দাঁড়ায় :

		Actual Data	
		(Positive)	(Negative)
Predicted Data	(Positive)	True Positive	False Positive
	(Negative)	False Negative	True Negative

টেবিল 13.2.2

এটি কনফিউশন ম্যাট্রিক্স। এখন, ভালো করে লক্ষ করবেন, কনফিউশন ম্যাট্রিক্সের ভেতরে আমি নতুন চারটি টার্ম ব্যবহার করেছি। সেগুলোর একটু বর্ণনা দিয়ে নিই :

1. ট্রু পজিটিভ (True Positive) : যখন অ্যালগরিদম কোনো কিছুকে সত্য বলে প্রেডিষ্ট করে এবং সেটি আসলেও সত্য হয়, তখন তাকে বলা হয় ট্রু পজিটিভ। যেমন, আমাদের এই ক্ষেত্রে, মেশিন যদি বলে রোগীর ক্যানসার আছে এবং যদি আসলেও রোগীর ক্যানসার থাকে, তখন তাকে বলা হবে ট্রু পজিটিভ।
2. ফলস পজিটিভ (False Positive) : যখন অ্যালগরিদম কোনো কিছুকে সত্য বলে প্রেডিষ্ট করে, কিন্তু সেটি আসলে সত্য নয়, তখন তাকে বলা হয় ফলস পজিটিভ। যেমন, আমাদের এই ক্ষেত্রে, মেশিন যদি বলে রোগীর ক্যানসার আছে, কিন্তু যদি আসলে রোগীর ক্যানসার না থাকে, তখন তাকে বলা হবে ফলস পজিটিভ।
3. ফলস নেগেটিভ (False Negative) : যখন অ্যালগরিদম কোনো কিছুকে মিথ্যা বলে প্রেডিষ্ট করে, কিন্তু সেটি আসলে সত্য হয়, তখন তাকে বলা হয় ফলস নেগেটিভ। যেমন, আমাদের এই ক্ষেত্রে, মেশিন যদি বলে রোগীর ক্যানসার নেই, কিন্তু যদি আসলে রোগীর ক্যানসার থাকে, তখন তাকে বলা হবে ফলস নেগেটিভ।
4. ট্রু নেগেটিভ (True Negative) : যখন অ্যালগরিদম কোনো কিছুকে মিথ্যা বলে প্রেডিষ্ট করে এবং সেটি আসলেই মিথ্যা হয়, তখন তাকে বলা হয় ট্রু নেগেটিভ। যেমন, আমাদের এই ক্ষেত্রে, মেশিন যদি বলে রোগীর ক্যানসার নেই এবং যদি আসলেও রোগীর ক্যানসার না থাকে, তখন তাকে বলা হবে ট্রু নেগেটিভ।

এখান থেকে আমরা পরিষ্কার বুঝতে পারছি যে ফলস পজিটিভ এবং ফলস নেগেটিভ আমাদের মোটেও কাম্য নয়। এখন পরিস্থিতি অর্থাৎ প্রবলেমের ধরনের ওপরে ভিত্তি করে আমাদের ঠিক করতে হবে যে আমরা ফলস পজিটিভের হার কমাব, নাকি ফলস নেগেটিভের, নাকি দুটোই কমানোর পদক্ষেপ নেব।

যেমন, যদি ক্যানসারের রোগীর উদাহরণ থেকে বলি, ফলস পজিটিভ মানে কোনো রোগীর ক্যানসার নেই, তাকে ভুল করে বলা হয়েছে তার ক্যানসার আছে। এর ফলে কী হবে, তার ক্যানসার না থাকা সত্ত্বেও হয়তো তার ক্যানসারের চিকিৎসা করা হবে। এটি অবশ্যই খারাপ এবং ক্ষতিকর।

কিন্তু চিন্তা করে দেখুন, যদি আমরা ফলস নেগেটিভ পাই, তাহলে কী হবে? কোনো রোগীর ক্যানসার থাকা সত্ত্বেও মেশিন রিপোর্ট দেবে তার ক্যানসার নেই। সেই রোগী হয়তো সেই ভুল রিপোর্ট নিয়ে খুশিমনে বাসায় চলে যাবে এবং কিছুদিনের মধ্যেই ক্যানসারে ভুগে মারা যাবে।

সুতরাং, দুটোর মধ্যে তুলনা করে আমরা দেখতে পাই যে, এই ক্ষেত্রে, দু-একটা ফলস পজিটিভ চলে এলে যতটুকু ক্ষতি হবে, তার চেয়ে বরং একটিও যদি ফলস নেগেটিভ থাকে, তাহলে তাতে

অনেক বেশি ক্ষতি হতে পারে, এমনকি মানুষের গ্রাণ্ড চলে যেতে পারে, তাই ফলস নেগেটিভ কোনোভাবেই এই সিস্টেমে গ্রহণযোগ্য নয়।

আবার, যদি আমরা অ্যালগরিদমকে আমাদের ইমেইল স্প্যাম (Spam) কি না সেটি যদি চিনতে শো�, তাহলে সে ক্ষেত্রে, ফলস পজিটিভ থাকলে অ্যালগরিদম আমাদের গুরুত্বপূর্ণ ইমেইলগুলো ভুলক্রমে স্প্যাম ভেবে মুছে দেবে, যার ফলে আমাদের বেশ বড়োসড়ো ক্ষতি হয়ে যেতে পারে। কিন্তু, যদি ফলস নেগেটিভ আসে, তাহলে কয়েকটা আজেবাজে মেইল আমাদের ইনবরে চলে আসা ছাড়া আর তেমন কোনো বড়ো ক্ষতি কিন্তু হবে না। সুতরাং, এ ক্ষেত্রে আমরা ফলস নেগেটিভে কিছুটা ছাড় দিতে পারলেও ফলস পজিটিভ একেবারেই ঘটতে দেওয়া যাবে না।

আবার যদি আমরা কোনো ফেসিয়াল রিকগনিশন (Facial Recognition)-ভিত্তিক বায়োমেট্রিক সিকিউরিটি সিস্টেম (Biometric Security System) তৈরি করি, যেখানে আপনার চেহারার সঙ্গে ডেটাবেজে ম্যাচ খুঁজে তবেই আপনাকে কোনো সিস্টেমে ঢুকতে দেবে, সে ক্ষেত্রে আমরা কিন্তু ফলস পজিটিভ (অন্য কোনো মানুষকে ভুলক্রমে আপনি মনে করে সিস্টেমে ঢুকতে দিয়েছে) কিংবা ফলস নেগেটিভ (আপনার সিস্টেমে ঢোকার অনুমতি থাকা সত্ত্বেও আপনাকে সিস্টেমে ঢুকতে দিচ্ছে না) কোনোটাই মেনে নিতে পারব না। দুটি আমাদের সিস্টেমের জন্য সমানভাবে খারাপ।

তাই, সিস্টেমের চাহিদা অনুযায়ী আমাদের ঠিক করতে হবে, কোনটি মিনিমাইজ করব।

### পরিচ্ছেদ ১৩.৩ : অ্যাকুরেসি (Accuracy)

কোনো মেশিন লার্নিং অ্যালগরিদমের অ্যাকুরেসি (Accuracy) হচ্ছে মূলত সে কতগুলো সঠিক প্রেডিকশন দিতে পারল, তার একটি হিসাব। অর্থাৎ, আমাদের মূল কাজ এখানে হচ্ছে ট্রু পজিটিভ এবং ট্রু নেগেটিভ নিয়ে।

আমরা কোনো মেশিন লার্নিং অ্যালগরিদমের অ্যাকুরেসি Accuracy মাপি এভাবে –

$$\checkmark \text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Positive} + \text{False Negative} + \text{True Negative}}$$

অর্থাৎ, আমাদের ট্রু পজিটিভ ও ট্রু নেগেটিভের সংখ্যা যত বেশি হবে আমাদের মডেলের অ্যাকুরেসি তত বাড়বে।

কখন অ্যাকুরেসি ব্যবহার করব – এই প্রশ্নের উত্তরে বলা যায়, যদি আমাদের ডেটাসেটের টার্গেট ভ্যারিয়েবলগুলো মোটামুটি ব্যালান্সড থাকে, সেই ক্ষেত্রে পারফরম্যান্স মেট্রিক হিসেবে অ্যাকুরেসি

ব্যবহার করতে পারি। অর্থাৎ, ধরা যাক, আমাদের কাছে থাকা 100টি জনের মধ্যে থেকে আমরা বের করতে চাইছি কোনটি আপেল আর কোনটি কমলার ছবি। এখন যদি, এই ডেটাসেটে মোটামুটি 50% থাকে আপেলের ছবি আর বাকি 50% থাকে কমলার ছবি – এরকম ক্ষেত্রে অ্যাকুরেসি ব্যবহার করা যায়।

কিন্তু যদি, আমাদের ডেটাসেটের টার্গেট ভ্যারিয়েবলগুলো ভালোভাবে ব্যালান্সড না থাকে, তখন অ্যাকুরেসি ব্যবহার করা ঠিক হবে না। ধরা যাক, আমাদের কাছে 100 জন সন্তান্য ক্যানসার রোগীর ডেটা আছে, এর মধ্যে 95 জনের (95%) ক্যানসার নেই, আর বাকি 5 জনের (5%) সত্যি সত্যিই ক্যানসার ধরা পড়েছে।

এখন, ধরা যাক, আমরা যেই অ্যালগরিদম তৈরি করেছি সেটি খুবই অকার্যকর, একেবারেই ক্যানসার রোগী ঠিকমতো চিনতে পারে না। সে ক্ষেত্রে সে এই 100 জন রোগীর ক্ষেত্রে আউটপুট দেবে যে 100 জনের কেউই ক্যানসারে আক্রান্ত নয়।

এখন আমরা যদি এ ক্ষেত্রে শুধু অ্যাকুরেসি মাপি, তাহলে আমরা হয়তো বলে বসব যে আমাদের মডেল খুব ভালো, একেবারে 95% অ্যাকুরেসি অর্জন করেছে, কিন্তু আসলেই কি তাই? আমাদের 5 জন ক্যানসার রোগী ছিল, মডেল একজনকেও চিনতে পারেনি। তার মানে, ক্যানসার নির্ণয়ে সম্পূর্ণ ব্যর্থ হয়েছে, সুতরাং এ ক্ষেত্রে শুধু অ্যাকুরেসি দিয়ে মডেলের ভালো/মন্দ বিচার করলে হবে না।

মোটামুটি এই-ই ছিল অ্যাকুরেসির ধারণা। আশা করি সবাই বুঝতে পেরেছেন।

### পরিচ্ছেদ ১৩.৪ : প্রিসিশন (Precision) ও রিকল (Recall)

আমাদের পরিচ্ছেদ ১৩.২-এ ব্যবহৃত কনফিউসন ম্যাট্রিক্সটি ছিল এরকম :

		Actual Data	
		(Positive)	(Negative)
Predicted Data	(Positive)	True Positive	False Positive
	(Negative)	False Negative	True Negative

টেবিল 13.4.2

প্রিসিশন (Precision)-এর ধারণাটি আমরা এখান থেকেই নেব। প্রিসিশন মানে হচ্ছে, আমাদের ক্যানসারের উদাহরণের সঙ্গে তাল মিলিয়ে যদি বলতে চাই – অ্যালগরিদম যতজন রোগীকে

### অধ্যায় ১৩: পারফরম্যান্স (Performance)

'ক্যানসার আছে' বলে ঘোষণা দিয়েছে এবং তাদের স্বার মধ্যে, যাদের আসলেই ক্যানসার আছে, এদের একটি অনুপাত।

অর্থাৎ,

$$\checkmark \text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

উদাহরণস্বরূপ, ধরা যাক, আমাদের 100 জন সন্তান্য রোগীর মধ্যে মাত্র 5 জনের আসলেই ক্যানসার আছে, বাকি 95 জনের নেই। এখন ধরা যাক, আমাদের অ্যালগরিদম খুবই অকার্যকর এবং সে সবাইকে ক্যানসার রোগী হিসেবে সন্দেহ করছে। সুতরাং, অ্যালগরিদম রিপোর্ট দেবে যে, 100 জনের সবাইই ক্যানসার রোগী। অর্থাৎ,

$$\text{True Positive} + \text{False Positive} = 100$$

কিন্তু, বাস্তবে রোগী হচ্ছে 5 জন, অর্থাৎ,

$$\text{True Positive} = 5$$

সুতরাং

$$\text{Precesion} = \frac{5}{100} = 0.05 = 5\%$$

এখন আসি রিকল (Recall)-এর ক্ষেত্রে কী হবে সেটাতে। রিকল হচ্ছে মূলত অ্যালগরিদম সত্যিকারের ক্যানসার রোগীদের মধ্যে কতজনকে ঠিক ঠিক প্রেডিষ্ট করতে পেরেছে এবং সত্যিই কতজনের ক্যানসার আছে (অ্যালগরিদম সেটাকে ঠিকমতো প্রেডিষ্ট করে থাকুক, কিংবা না-ই থাকুক) তার একটি অনুপাত।

অর্থাৎ,

$$\checkmark \text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

এখনে ফলস নেগেটিভও যোগ করার কারণ হচ্ছে, ফলস নেগেটিভদের ক্ষেত্রে সত্যি সত্যি ক্যানসার থাকা সত্ত্বেও অ্যালগরিদম তাদের ক্ষেত্রে বলবে যে ক্যানসার নেই।

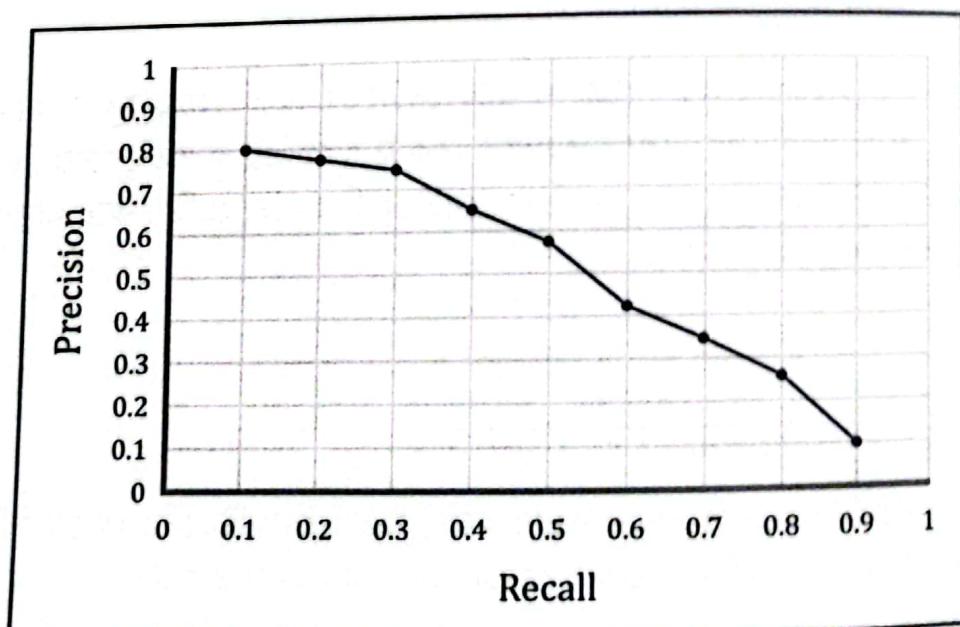
যদি, 100 জনের ভেতরে 10 জনের সত্যিই ক্যানসার থাকে এবং তাদের মধ্যে 5 জনকে অ্যালগরিদম ঠিক ঠিক বের করে ফেলতে পারে, তাহলে –

$$\text{Recall} = \frac{5}{10} = \frac{1}{2} = 50\%$$

এতক্ষণের আলোচনা থেকে এটি কিন্তু পরিষ্কার বোঝা যাচ্ছে যে, প্রিসিশন কাজ করলে ফলস পজিটিভ নিয়ে, অর্থাৎ বোঝাবে যে আমরা কতজন সত্যিকারের ক্যানসার রোগীকে নির্ণয় করতে পেরেছি। অন্যদিকে রিকল কাজ করবে ফলস নেগেটিভ নিয়ে, অর্থাৎ আমরা কতজন সত্যিকারের ক্যানসার রোগীকে ঠিকমতো বের করতে পারিনি, সেটি বোঝাবে।

তাই, আমরা যদি চাই আমাদের সিস্টেমের ফলস নেগেটিভ করাতে, তখন আমাদের চেষ্টা করতে হবে যে আমাদের রিকলের মান যেন 100%-এর কাছাকাছি নিয়ে যাওয়া যায়। যত কাছাকাছি নিতে পারব, আমাদের ফলস নেগেটিভ তত কমে আসতে থাকবে।

এখানে একটু খেয়াল রাখতে হবে, রিকল বাড়ানোর জন্য, অর্থাৎ, আমাদের যাতে কোনো ক্যানসার রোগী বাদ না পড়ে, সেটি নিশ্চিত করার জন্য আমরা কী করব? যত বেশি সন্তুষ্ম মানুষকে ক্যানসার রোগী হিসেবে প্রেডিষ্ট করানোর চেষ্টা করব, তাই না? যত বেশি মানুষ আমরা ক্যানসার রোগী হিসেবে নেব, ততই আমাদের আসল ক্যানসার রোগীদের বাদ পড়ার সন্তাননা কমে আসবে, ঠিক? কিন্তু সেই সঙ্গে দেখুন, ক্যানসারে আক্রান্ত নয়, এমন রোগীও ক্যানসার রোগী হিসেবে চিহ্নিত হওয়ার সন্তাননা এবং সংখ্যা বাড়তে থাকবে, অর্থাৎ ফলস পজিটিভ বাড়তে থাকবে। তাই রিকল বাড়ানোর সঙ্গে সঙ্গে এটিও লক্ষ রাখতে হবে, যাতে প্রিসিশন খুব বেশি কমে না যায়, অর্থাৎ অনেক বেশি ফলস পজিটিভ বেড়ে না যায়।



গ্রাফ 13.4.1 : প্রিসিশন-রিকল গ্রাফ

একইভাবে, যদি আমরা চাই ফলস পজিটিভ করাতে, আমাদের তখন চেষ্টা করতে হবে প্রিসিশনের মান 100%-এর যত কাছাকাছি নিয়ে যাওয়া যায়। এখন ফলস পজিটিভ করানোর জন্য আমরা কী করব? আমাদের মোট ক্যানসার রোগী হিসেবে চিহ্নিত রোগীর সংখ্যা কমানোর চেষ্টা করব, যাতে আসল ক্যানসার রোগী বাদে অন্য কেউ না থাকে, তাই না? কিন্তু অনেক সময় এই ক্যানসার

### অধ্যায় ১৩ : পারফরম্যান্স (Performance)

রোগী হিসেবে চিহ্নিত রোগীর সংখ্যা কমাতে গিয়ে আমরা আসল ক্যানসার রোগীকেও বাদ দিয়ে দিতে পারি, যেটি কিনা ফলস নেগেটিভ বাড়িয়ে দেবে। তাই, প্রিসিশন বাড়ানোর সঙ্গে সঙ্গে এটিও নিষ্ঠিত করতে হবে, যাতে রিকল খুব বেশি কমে না যায়। গ্রাফ 13.4.1-এ ব্যাপারটি আরেকটু সহজ করে বোঝানোর চেষ্টা করা হলো :

গ্রাফ 13.4.1 থেকে ভালোমতো লক্ষ করলে দেখবেন, রিকলের মান যখন খুবই কম, প্রিসিশনের মান তখন অনেক বেশি। আন্তে আন্তে রিকলের মান বাড়ানোর সঙ্গে সঙ্গে আমাদের প্রিসিশনের মান কমেছে। এটিকে বলে প্রিসিশন-রিকল ট্রেডঅফ (Precision-Recall Tradeoff)। তাই, আমাদের সিস্টেমের চাহিদা অনুসারে আমাদের ঠিক করে নিতে হবে যে আমরা প্রিসিশনকে ম্যাস্ক্রিমাইজ করব নাকি রিকলকে।

এই ছিল প্রিসিশন এবং রিকলসংক্রান্ত আলোচনা।

### পরিচ্ছন্দ ১৩.৫ : এফ-ওয়ান মেজার (F1 Measure)

আমরা এতক্ষণ দেখলাম, কী করে প্রিসিশন এবং রিকল নির্ণয় করতে হয় এবং কখন কোনটি কীভাবে বাড়াতে কিংবা কমাতে হয়। কথা হচ্ছে, প্রিসিশন ও রিকল দুটি সম্পূর্ণ ভিন্ন ভিন্ন পারফরম্যান্স মেট্রিক। এখন একটি অ্যালগরিদমকে যাচাই করার জন্য যদি এই দুটি ভিন্ন ভিন্ন মানের পরিবর্তে দুটি মিলিয়ে একটি মান ব্যবহার করা যায়, সেটি আরো ভালো হয় না?

ঠিক এই ধারণার থেকেই এফ-ওয়ান মেজার (F1 Measure)-এর জন্ম। এফ-ওয়ান মেজারকে সহজ করে বলা যায় যে, এটি হচ্ছে প্রিসিশন ও রিকল এর মানের 'এক বিশেষ ধরনের' গড়। এখন কথা হচ্ছে, বিশেষ ধরনের গড় আবার কী? গড় তো গড়ই। দুটি মান যোগ করব, দুই দিয়ে ভাগ করব, এই তো হলো গড়! এর বাইরেও আবার কিছু আছে নাকি?

তার উত্তর দেওয়ার আগে চলুন একটু দেখে নিই প্রিসিশন ও রিকলের সাধারণ গড় মান নিলে কোন পরিস্থিতির উদ্বেক হয়।

আমরা যদি সাধারণ গাণিতিক গড় নিই, তবে –

$$\checkmark F1\ Measure = \frac{Precision + Recall}{2}$$

এটি কিছু কিছু ক্ষেত্রে ভালো কাজ করলেও কিছু কিছু ক্ষেত্রে খুবই ক্রটিপ্রবণ ফলাফল দেবে। যেমন, ধরা যাক, আমাদের 100 জন রোগীর মধ্যে 3 জনের ক্যানসার আছে, বাকি 97 জনের নেই। আরো ধরা যাক, আমাদের মডেল এতই অকার্যকর যে সে 100 জনের সবাইকে ক্যানসার রোগী হিসেবে প্রেডিষ্ট করেছে। তাহলে –

Actual Data			
Predicted Data	(Cancer)	(No Cancer)	
	(Cancer)	3	97
	(No Cancer)	0	0

টেবিল 13.5.1

এখন, তাহলে এই টেবিল থেকে আমরা যদি প্রিসিশন ও রিকল বের করে ফেলি তাহলে দাঁড়ায়  
এরকম -

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{3}{100} = 3\%$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} = \frac{3}{3} = 100\%$$

সূতরাং এখান থেকে আমাদের এফ-ওয়ান মেজার হয়,

$$F1 \text{ Measure} = \frac{3 + 100}{2} = 51.5\%$$

এখন দেখুন, আমাদের মডেলটি কিন্তু খুবই অকার্যকর ছিল, সেসব রোগীকেই ক্যানসার রেপ্টি  
হিসেবে প্রেডিষ্ট করেছে। কিন্তু তা সত্ত্বেও আমাদের এফ-ওয়ান মেজার আসছে 51.5% রেটি কিন্তু  
বেশ মাঝামাঝি পর্যায়ের একটি মান, তাই না? অথচ এত বাজে মডেলের ক্ষেত্রে অনেক কম ধৰণের  
কথা। আর তাই, আমরা শুধু গাণিতিক গড় ব্যবহার করে এফ-ওয়ান মেজারের মান হিসাব করলে  
সব সময় সঠিক তথ্য পাব না। সূতরাং, আমাদের সেই 'বিশেষ গড়'-এ ফিরে যেতে হবে।

আমাদের বিশেষ এই গড়মানটির একটি গালভরা নাম আছে। একে বলা হয় হারমনিক গড় বা  
হারমনিক মিন (Harmonic Mean)।

হারমনিক গড়ের বৈশিষ্ট্য হলো যদি, আমরা যদি X ও Y সাধারণ গড় (arithmetic mean) হিসাব  
করি, তাহলে দেখুন এটি X ও Y-এর ঠিক মাঝ বরাবর থাকবে। কিন্তু, হারমনিক গড়ের ক্ষেত্রে  
সেটি একটু আলাদা হবে। যদি X ও Y-এর মান মোটামুটি সমান হয়, তবে সে ক্ষেত্রে হারমনিক  
গড়ের মান দুজনের সাধারণ গড়, অর্থাৎ মাঝ বরাবর থাকবে। কিন্তু যদি, এমন হয় যে X ও Y-এর  
মধ্যে যে-কোনো একটি বড়ো এবং অপরটি ছোটো, তখন হারমনিক গড়ের মান X ও Y-এর মধ্যে  
যেটি ছোটো সেটির কাছাকাছি থাকবে। অর্থাৎ, যদি X ছোটো হয়, তাহলে X-এর কাছাকাছি থাকবে,  
কিন্বা Y ছোটো হলে Y-এর কাছাকাছি থাকবে।

এই হারমনিক গড়ের সূত্র হচ্ছে,

### অধ্যায় ১৩ : পারফরম্যান্স (Performance)

$$\text{Harmonic Mean} = \frac{2XY}{X+Y}$$

অর্থাৎ যদি আমরা প্রিসিশন ও রিকল-এর হারমনিক গড় নিই তাহলে –

$$\checkmark \text{Harmonic Mean} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

আমরা এখন, আমাদের সেই ক্যানসারের উদাহরণে প্রিসিশন ও রিকল-এর হারমনিক গড় নিলে পাই –

$$\text{Harmonic Mean} = \frac{2 * 3 * 100}{3 + 100} = \frac{600}{103} = 5.825\%$$

এখন কি মনে হচ্ছে না, যে এই স্কোরটি আমাদের ওই মডেলের জন্য উপযুক্ত হয়েছে? ঠিক তাই। যেসব মডেলের পারফরম্যান্স খারাপ, প্রতিটি মডেলকে এভাবে স্কোর করিয়ে দিয়ে উপযুক্ত শান্তি দেয় এই হারমনিক গড়।

তাই, এফ-ওয়ান মেজারের সময় আমরা অবশ্যই সাধারণ গড়ের পরিবর্তে হারমনিক গড় ব্যবহার করব।

### পরিচ্ছেদ ১৩.৬ : স্পেসিফিসিটি (Specificity)

আবারও, আমাদের পূর্বে ব্যবহৃত টেবিল থেকে –

		Actual Data	
Predicted Data		(Positive)	(Negative)
	(Positive)	True Positive	False Positive
	(Negative)	False Negative	True Negative

টেবিল 13.6.1

স্পেসিফিসিটি (Specificity) হচ্ছে যতজনের ক্যানসার ছিল না বলে মেশিন প্রেডিষ্ট করেছে এবং তাদের মধ্যে আসলেই যারা ক্যানসার রোগী নয়, তাদের একটি অনুপাত। এটি দেখা যাচ্ছে রিকলের ঠিক বিপরীত।

অর্থাৎ,

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}}$$

এখানে ফলস পজিটিভও যোগ করার কারণ হচ্ছে, যারা ফলস পজিটিভ তাদের কিন্তু আসলে ক্যানসার নেই, কিন্তু ভুল ডায়াগনোসিস করে মেশিন বলেছে তাদের ক্যানসার আছে।

ধরা যাক, আগের মতোই, 100 জনের ভেতরে 5 জনের ক্যানসার আছে, বাকি 95 জনের নেই। এখন মেশিন বলল ওই 5 জনসহ মোট 10 জনের ক্যানসার আছে, অর্থাৎ 90 জনের ক্যানসার নেই। অতএব,

$$True\ Negative = 5; \ False\ Positive = 5$$

তাহলে,

$$Specificity = \frac{5}{10} = 50\%$$

এই হচ্ছে স্পেসিফিসিটি। আমাদের লক্ষ্য থাকবে স্পেসিফিসিটির মান যতদূর সন্তুষ্ট বাড়ানো।

### পরিচ্ছেদ ১৩.৭ : আরওসি কার্ভ (ROC Curve)

এতক্ষণ আমরা যা যা পড়েছি, মোটামুটি সবকিছুর জ্ঞান আমাদের প্রয়োজন হবে আরওসি কার্ভ (ROC Curve) বোঝার জন্য। আরওসি কার্ভের পুরো নাম – রিসিভার অপারেটিং ক্যারেকটারিস্টিক কার্ভ (Receiver Operating Characteristic Curve)। নাম দেখে হয়তো মনে হতে পারে যে বেশ জবরিজং একটি পারফরম্যান্স মেট্রিক এটি এবং আসলেই তা-ই।

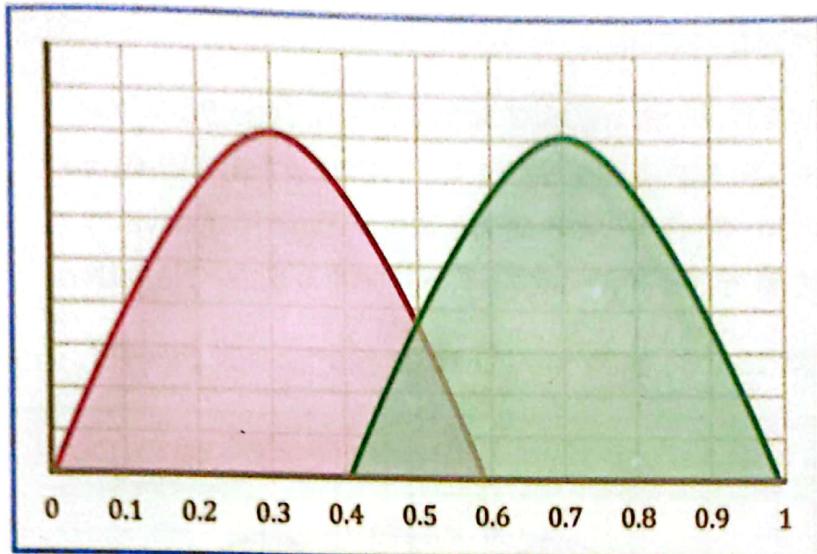
আরওসি কার্ভ দিয়ে আমরা মূলত যেটি বুঝি, সেটি হচ্ছে একটি মডেল কত ভালোভাবে দৃটি ভিন্ন ভিন্ন ক্লাসের মধ্যে পার্থক্য করতে পারে। একটি খুব ভালো মডেল খুব ভালোভাবে দৃটি ক্লাসের মধ্যে পার্থক্য করতে পারবে, কিন্তু একটি খারাপ মডেল কখনোই তা পারবে না।

আরওসি কার্ভের ধারণায় যাওয়ার আগে, আমরা আমাদের TP, TN, FP ও FN-এর ধারণাটি একটু ভিন্নভাবে দেখি এবার।

ধরা যাক, মেশিন আবারও আগের মতোই ক্যানসারের রোগী প্রেডিষ্ট করছে। এবার আমরা তার অনুমানকে একটি সন্তান্যতার মান দিয়ে চিহ্নিত করব। গ্রাফ 13.7.1 দেখি। এখানে, লাল রঙের কার্ভ থেকে দেখতে পাচ্ছি, যাদের সন্তান্যতার মান 0 থেকে 0.6-এর মধ্যে আছে তাদের সত্তি সত্যিই ক্যানসার নেই। আর সবুজ রঙের কার্ভ থেকে দেখতে পাচ্ছি, যাদের সন্তান্যতার মান 0.4 থেকে 1-এর মধ্যে আছে তাদের সত্যিই সত্যিই ক্যানসার আছে। সন্তান্যতার মানটি কোনোভাবে হিসাবনিকাশ করে বের করা হয়েছে বলে ধরে নেওয়া যাক।

### অধ্যায় ১৩: পারফরমেন্স (Performance)

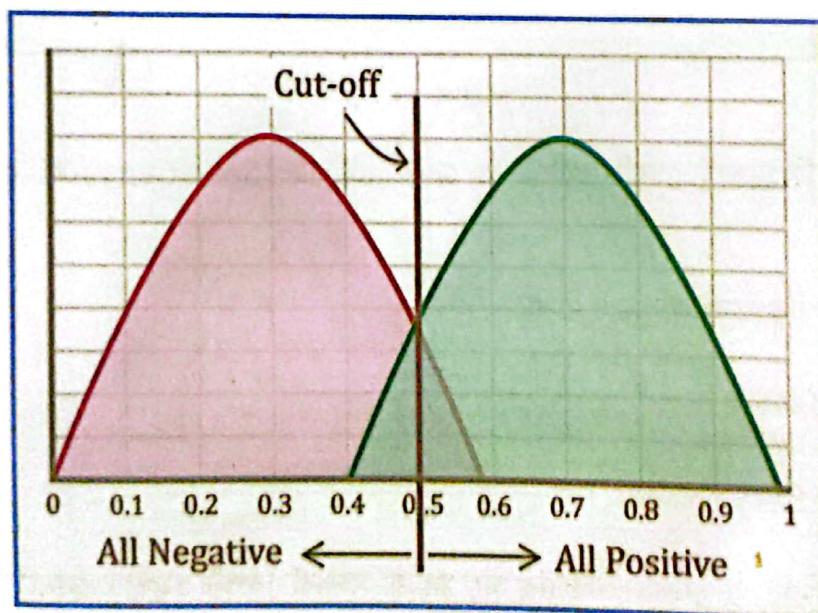
এখন কথা হচ্ছে, তালো করে লক্ষ করবেন, এখানে কার্ড দুটির কিছু অংশ ওভারল্যাপ (Overlap) করছে। সম্ভাব্যতা যদি  $0.4$  থেকে  $0.6$ -এর মধ্যে কোনো মান হয়, তখন কিন্তু পজিটিভ কিংবা নেগেটিভ দুটির যে-কোনোটিই আউটপুট হতে পারে। কিন্তু, তা তো হতে দেওয়া যায় না! কেউ তো একই সঙ্গে পজিটিভ ও নেগেটিভ দুটিই হতে পারে না।



গ্রাফ 13.7.1

তাই, আমরা এখন যেটি করব, সেটি হচ্ছে একটি থ্রেসহোল্ড মান ঠিক করব এবং এটি মেশিনকে শিখিয়ে দেব। থ্রেসহোল্ড মানের নিচের সব সম্ভাব্যতার জন্য মেশিন বলবে ক্যানসার নেই; আর থ্রেসহোল্ড মানের ওপরের সব মানের জন্য মেশিন বলবে ক্যানসার আছে।

এখন, আমরা যদি থ্রেসহোল্ড মান ধরি  $0.5$ , তাহলে –



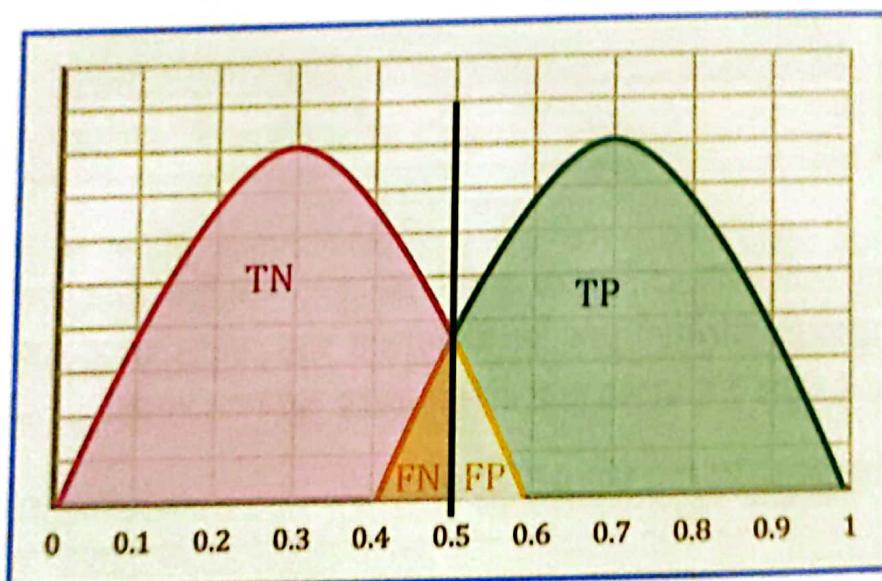
গ্রাফ 13.7.2

এখন ভালোমতো লক্ষ করুন, 0.5-এর মুই পাশেই এমন কিছু মান আছে, যেটোর সেধানে থাকার কথা নয়। 0.5-এর ডান পাশে শুধুই সবুজ গ্রাফ থাকার কথা, লাল গ্রাফের কোনো অংশ থাকার কথা নয়। একইভাবে, 0.5-এর বাঁ পাশে শুধুই লাল গ্রাফ থাকার কথা, সবুজ গ্রাফের কোনো অংশ থাকার কথা নয়।

এখন তাহলে বুঝাতেই পারছেন,

- 0.5-এর ডান পাশের সব সবুজ গ্রাফের মান = True Positive
- 0.5-এর ডান পাশের লাল গ্রাফের অংশবিশেষ = False Positive
- 0.5-এর বাঁ পাশের সব লাল গ্রাফের মান = True Negative
- 0.5-এর বাঁ পাশের সবুজ গ্রাফের অংশবিশেষ = False Negative

নিচের গ্রাফ (গ্রাফ 13.7.3) থেকে ধারণাটি আরো পরিষ্কার হবে বোধকরি :



গ্রাফ 13.7.3

এখন আমরা ইতিমধ্যেই স্পেসিফিসিটি ও সেনসিটিভিটি (Sensitivity) বা রিকল বিষয় দুটি জানি।

আপনাদের সুবিধার জন্য আবারও এখানে দিচ্ছি –

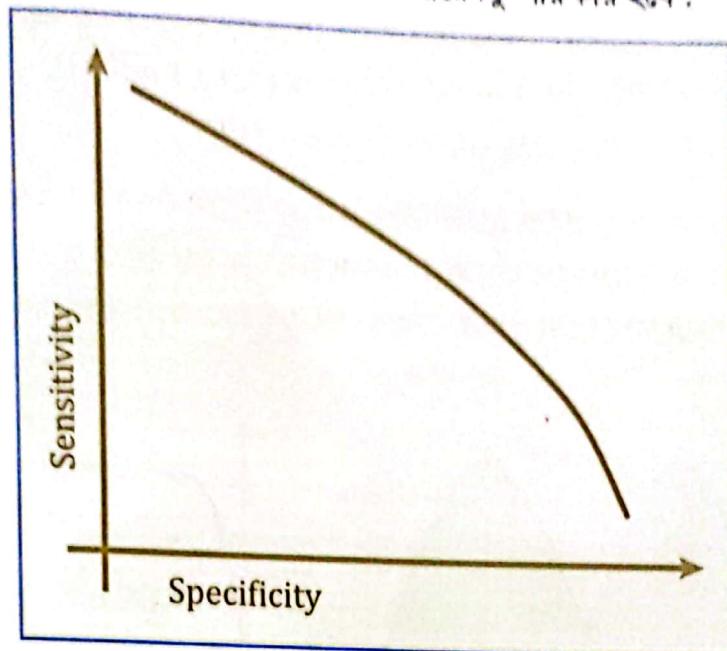
$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Positive}}$$

$$\text{Sensitivity (Recall)} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

এখন স্পেসিফিসিটি ও সেনসিটিভিটির মধ্যেকার সম্পর্ক বিপরীতমূখী। সেনসিটিভিটি বাড়লে স্পেসিফিসিটি কমে, আবার সেনসিটিভিটি কমলে স্পেসিফিসিটি বাঢ়ে।

অধ্যায় ১৩ : পারফরম্যান্স (Performance)

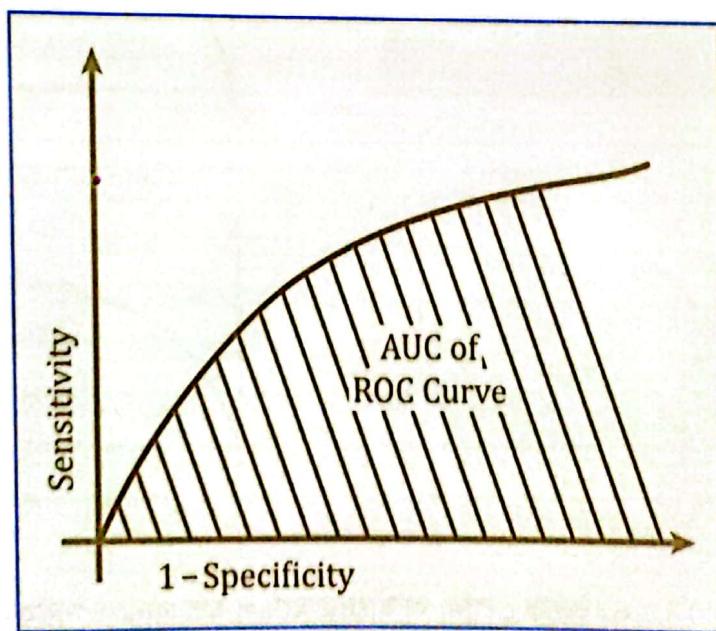
নিচের গ্রাফ (গ্রাফ 13.7.4) থেকে হয়তো বিষয়টি আবেকটু পরিষ্কার হবে :



গ্রাফ 13.7.4

এখন, ROC Curve-এর জন্য আমরা শুধু Specificity-এর মানের বদলে  $(1 - Specificity)$ -এর মান Sensitivity-এর বিপরীতে প্লট করব।

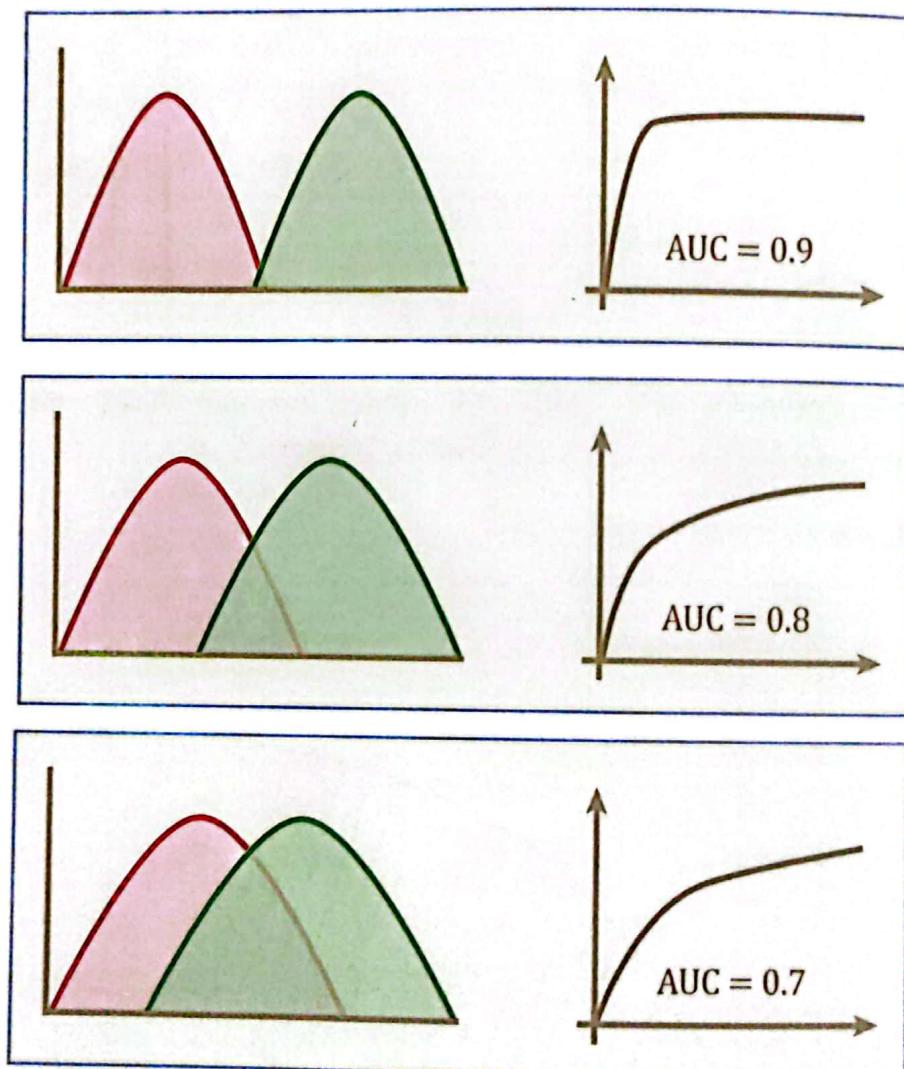
তখন গ্রাফটি দাঁড়াবে এরকম (গ্রাফ 13.7.5) :



গ্রাফ 13.7.5

এই আরওসি কার্ড গ্রাফে তাহলে দেখা যাচ্ছে সেনসিটিভিটি যত বাড়বে, ( $1 - Specificity$ )-এর মানও তত বাড়বে। আরেকটি টার্ম আমি এখানে ব্যবহার করেছি, দেখবেন টার্মটি হচ্ছে AUC। এখানে, AUC হচ্ছে এরিয়া আন্ডার দ্য আরওসি কার্ড (Area Under the ROC Curve) বা সংক্ষেপে এরিয়া আন্ডার কার্ড (Area Under Curve - AUC)।

আমাদের, এই আরওসি কার্ড গ্রাফে AUC অর্থাৎ কার্ডের নিচের ক্ষেত্র যত বড়ো হবে, আমাদের মডেল তত ভালো কাজ করছে বলে ধরা হবে। আর ক্ষেত্র যদি কম হয়, তাহলে ধরে নিতে হবে যে আমাদের মডেল দুটি ক্লাসের মধ্যে পার্থক্য করার ক্ষেত্রে খুব একটা ভালো পারফরম করছে না।

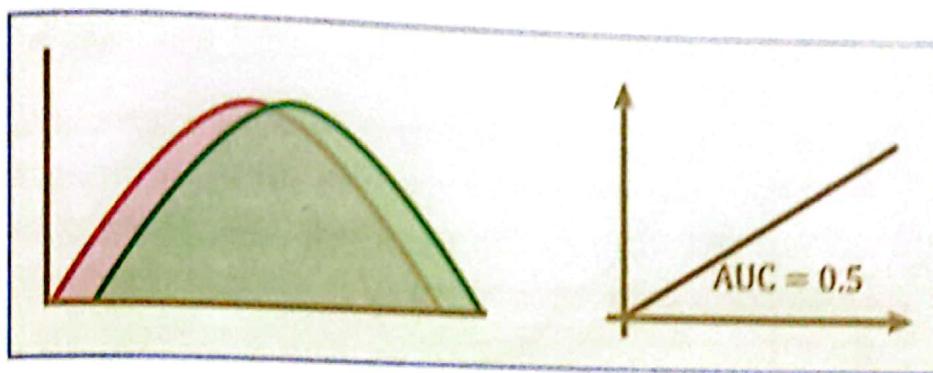


গ্রাফ 13.7.6

ওপরের গ্রাফ (গ্রাফ 13.7.6) থেকে দেখুন, আমাদের মডেল যত খারাপ পারফরম করছে, অর্থাৎ ওভারল্যাপ যত বাড়ছে, আমাদের আরওসি কার্ডের AUC ততই কমছে।

### অধ্যায় ১৩: পারফরম্যান্স (Performance)

আরো দেখুন, মডেল যদি আয় পুরোপুরিই খারাপ পারফরম করে, অর্থাৎ আয় পুরোপুরিই ওভারল্যাপ করে, যদি দুটি ফ্লাসের ভেতরে একেবাবেই আলাদা না করতে পারে, তখন একই হয় এরকম (গ্রাফ 13.7.7) :



গ্রাফ 13.7.7

এখন সবশেষে আসি, কেন আমরা ( $1 - Specificity$ ) ব্যবহার করছি, তার ব্যাখ্যায়।

আমরা জানি,

$$Specificity = \frac{TN}{TN + FP}$$

$$1 - Specificity = 1 - \frac{TN}{TN + FP} = \frac{TN + FP - TN}{TN + FP} = \frac{FP}{TN + FP}$$

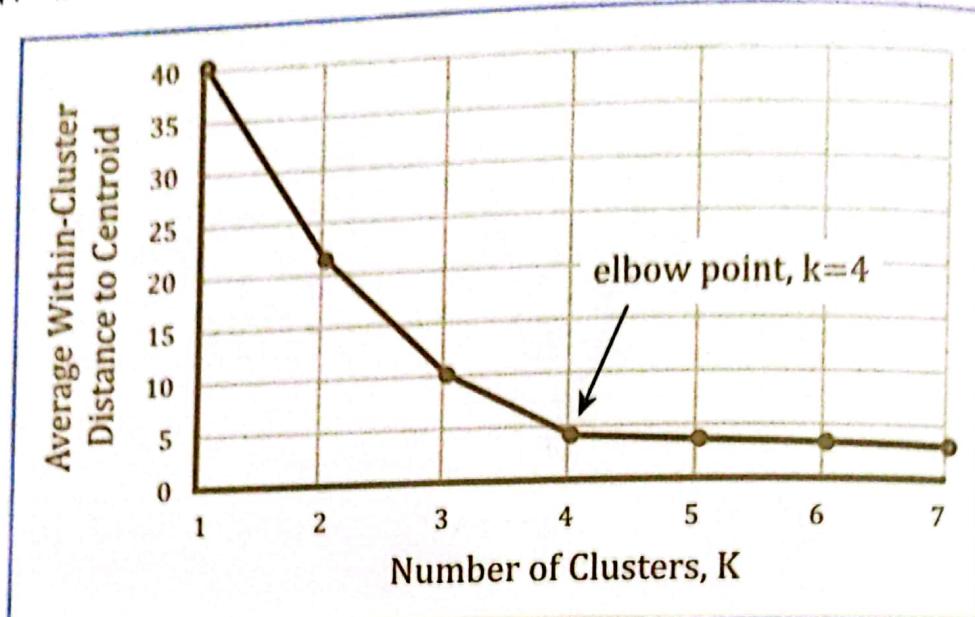
এখান থেকে বোঝা যাচ্ছে, স্পেসিফিটি আমাদের দিত ট্রু নেগেটিভের হার, কিন্তু ( $1 - Specificity$ ) আমাদের দেবে ফলস পজিটিভের হার। সুতরাং, সেনসিটিভিটিকে ট্রু পজিটিভের হার এবং ( $1 - Specificity$ )-কে ফলস পজিটিভের হার বলা চলে।

এখন তাহলে, আমরা শুধু পজিটিভ নিয়ে কাজ করছি, কোনো ধরনের নেগেটিভ নিয়ে কাজ করছি না। গ্রাফ 13.7.3 থেকে, আমরা যদি আমাদের প্রেসহোল্ডের মান বাড়াই, আমরা তাহলে ট্রু পজিটিভ ও ফলস পজিটিভ দুটিই কমিয়ে নিয়ে আসব। আবার যদি প্রেসহোল্ডের মান কমাই, তাহলে আমরা ট্রু পজিটিভ ও ফলস পজিটিভ দুটিই বাড়িয়ে ফেলব। অর্থাৎ, আমরা বলতে পারি যে, আমরা এখন শুধু প্রেসহোল্ডের ডান পাশের দুটি মান (ট্রু পজিটিভ ও ফলস পজিটিভ) নিয়েই কাজ করছি, প্রেসহোল্ডের উভয় পাশের মান নিয়ে নয়। এতে আমাদের হিসাবের বেশ ভালো রকমের সুবিধা হয় দেখেই ( $1 - Specificity$ ) ব্যবহার করা হয়ে থাকে।

### পরিচ্ছেদ 13.8 : এলবো মেথড (Elbow Method)

এলবো মেথড (Elbow Method) ব্যবহার করা হয় কে-মিনস ফ্লাস্টারিংয়ের ক্ষেত্রে।

আগেই বলা হয়েছে, এটি আসলে ঠিক কোনো পারফরম্যাল মেট্রিক নয়। এটি হলো মূলত কেনিস ক্লাস্টারিংয়ের ফ্রেঞ্চে  $K$ -এর অপটিমাল মান বের করার একটি পদ্ধতি মাত্র।



গ্রাফ 13.8.1

ওপরের গ্রাফটি (গ্রাফ 13.8.1) হচ্ছে এলবো মেথডের গ্রাফ। এটি যেভাবে তৈরি করা হয়েছে তা হলো, প্রথমে  $K$ -এর বিভিন্ন মান নেওয়া হয়েছে। ধরি, যদি  $K = 3$  হয়, তবে গোটা ডেটাসেট তিনটি ক্লাস্টারে ভাগ হয়ে যাবে, তাই না? এর পরে, আমাদের ডেটাসেটের প্রতিটি পয়েন্টকে এই তিনটি ক্লাস্টারের কোনো একটিতে অ্যাসাইন করা হয়েছে এবং প্রতিবার একটি করে ডেটা পয়েন্টকে কোনো ক্লাস্টারে অ্যাসাইন করার পরে সেই ক্লাস্টারের সেন্ট্রয়েড আপডেট করা হয়েছে (বিস্তারিত অধ্যায় ৭-এ রয়েছে)।

এরপরে, প্রতিটি ডেটাপয়েন্ট  $p$  যে ক্লাস্টারে অ্যাসাইন করা হয়েছে, সেই ক্লাস্টারের সেন্ট্রয়েডের সঙ্গে ওই ডেটা পয়েন্টের দূরত্ব নির্ণয় করা হয়েছে। অর্থাৎ, সহজ কথায় কোনো  $p(x, y)$  যদি ক্লাস্টার 1-এ থাকে, তাহলে ক্লাস্টার 1-এর সেন্ট্রয়েড  $c_1$ -এর সঙ্গে  $p$ -এর দূরত্ব বের করা হয়েছে। আবার, যদি অন্য একটি ডেটা পয়েন্ট  $q(x, y)$  ক্লাস্টার 2-তে থাকে, তাহলে  $q$ -এর সঙ্গে ক্লাস্টার 2-এর সেন্ট্রয়েড  $c_2$ -এর দূরত্ব বের করা হয়েছে। পরবর্তী সময়ে প্রতিটি ডেটা পয়েন্টের জন্য পাওয়া দূরত্বের মানকে বর্গ করে, বর্গগুলো যোগ করে আমরা যে মান পাই সেটি হচ্ছে  $K = 3$ -এর জন্য সাম অব স্কয়ারড এরর (Sum of Squared Error) বা এসএসই (SSE) মান।

একইভাবে আমরা,  $K$ -এর বাকি সব মানের জন্য SSE মান বের করব। আমরা  $K$ -এর মান 1 থেকে 10 পর্যন্ত নিতে পারি। গ্রাফ 13.8.1-এ  $K = 1$  থেকে 7 পর্যন্ত দেখানো হয়েছে। একটু লক্ষ করলে দেখব, গ্রাফটি দেখতে অনেকটা মানুষের হাতের কনুই বা এলবো (Elbow)-এর মতো। গ্রাফে

### অধ্যায় ১৩ : পারফরম্যান্স (Performance)

এলবো পয়েন্ট (elbow point) হিসেবে যেটিকে চিহ্নিত করা আছে, সেই পয়েন্টটিই হচ্ছে আমাদের হাতের কনুইয়ের ভাঁজ।

এই এলবো পয়েন্টটি বের করতে পারলেই আমাদের কাজ শেষ।  $K$ -এর যে মানের জন্য আমরা এই এলবো পয়েন্টটি পাচ্ছি, সেটি-ই হবে আমাদের  $K$ -এর অপটিমাল মান।

এখন, এই এলবো পয়েন্টের মান বের করার জন্য কিন্তু কোনো গাণিতিক সূত্র নেই। এটি চোখে দেখে অনেকটা আন্দাজ করে বের করতে হয়। লক্ষ করলে দেখবেন, গ্রাফটি  $K = 1$  থেকে যখন শুরু হয় তখন SSE অনেক বেশি ছিল।  $K$ -এর মান বাড়ার সঙ্গে সঙ্গে হঠাতে করে SSE-এর মান অনেকখানি নেমে আসে, একটি নির্দিষ্ট বিন্দু বা পয়েন্টের পরে এই পরিবর্তনের হার অনেকখানি কমে আসে। এই পয়েন্টটিকেই আমরা এলবো পয়েন্ট বলি।