# Դասակարգման մետրիկաներ

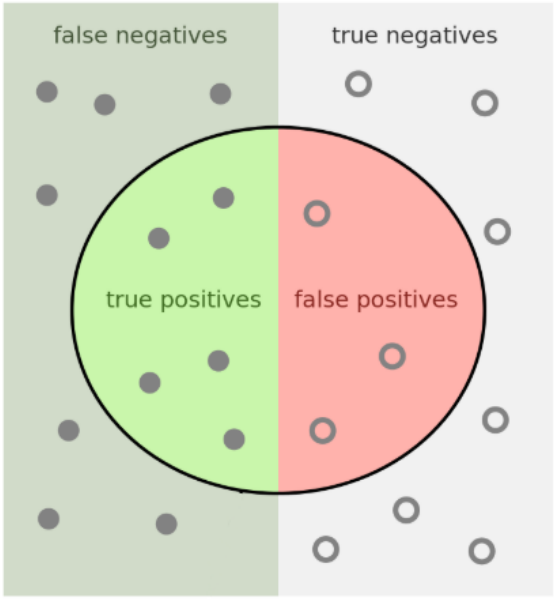
Հայկ Կարապետյան

## 1 Մետրիկաների տեսակներ

Ցանկացած խնդիր լուծելիս, մեզ անհրաժեշտ է հասկանալ, թե մոդելը ինչքան լավ է աշխատում: Կորստի ֆունկցիայի արժեքը մեզ այդքան էլ շատ ինֆորմացիա չի կարող հաղորդել: Օրինակ՝ կորուստը 0.1 է, մեզ բավարար տեղեկություն չի տալիս մոդելի լավ աշխատելու մասին: Այդ պատճառով օգտագործում ենք ուրիշ մետրիկա, որը կոչվում է ճշգրտություն (accuracy): Ճշգրտությունը չափում ենք հետևյալ բանաձևով.

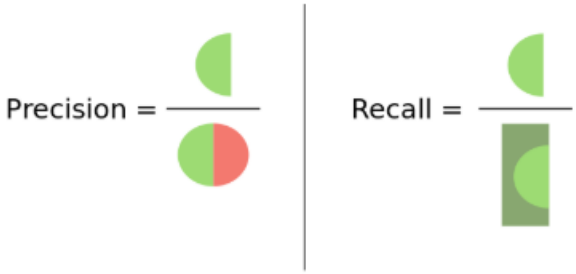
$$\text{Ճշգրտություն} = \frac{\text{Ճիշտ կատարված գուշակությունների քանակ}}{\text{Բոլոր գուշակությունների քանակ}}$$

Այս մետրիկայի արժեքը միշտ չէ, որ հստակ ասում է մոդելի լավ աշխատելը: Օրինակ՝ մոդելի ճշգրտությունը 90% է: Դրանից ելնելով չենք կարող ասել, որ մոդելը լավ է աշխատում: Ամեն ինչ կախված է այն տվյալներից, որոնց համար հաշվել ենք ճշգրտությունը: Օրինակ թեստային տվյալները բաժանված են հետևյալ կերպ. 900 շան նկար, 100 կատվի նկար: Պատկերացնենք մոդելը մեր ուղեղն և մենք աչքերներս փակ, ինչ նկար ցույց են տալիս ասում ենք շուն: Կստացվի, որ մեր ուղեղի ճշգրտությունը 90% է: Այդ պատճառով ճշգրտության տվյալները պետք է հավասարաչափ բաշխված լինեն: Սահմանենք մի քանի տերմիններ: True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN): Դիտարկենք հետևյալ օրինակը. հիվանդանոցում անցկացնում են թեստ և եթե մարդը վարակված է, ապա նրա class-ը positive է (1), եթե առողջ է՝ negative (0): Այն մարդկանց քանակը, որոնց մեր մոդելը ասել է, որ վարակված են և նրանք իսկապես վարակված են եղել, TP-ն է: Կարող ենք կարդալ հակառակ (Positive True), այսինքն մոդելը վերադարձրել է positive և ճիշտ (true) է եղել: Այն մարդկանց քանակը, որոնց մեր մոդելը ասել է վարակված, բայց իրականում նրանք առողջ են եղել, FP-ն է: Մոդելը վերադարձրել է positive և սխալվել է (false): Այն մարդկանց քանակը, որոնց մեր մոդելը ասել է առողջ և իրականում եղել են առողջ՝ TN: Այն մարդկանց քանակը, որոնց մոդելը ասել է առողջ, բայց իրականում եղել են վարակված՝ FN: Ասել է negative և սխալվել է (false): Նկար 1-ում պատկերված են այս 4 տերմինները: Սև օղակի մեջ վերցված են այն մարդիկ ում մոդելը ասել է վարակված (positive):




Նկար 1: Մուգ օղակի մեջ ըստ մոդելի վարակված (positive) մարդիկ են, իսկ օղակից դուրս առողջ (negative) մարդիկ

Այս խնդրում ակնհայտ է, որ մեզ անհրաժեշտ է քչացնել false negative-ների քանակը, այսինքն եթե մարդը վարակված է հնարավորինս քիչ անգամ ասենք, որ առողջ է: Մոդելի այդպիսի ճշգրտությունը հաշվելու համար առաջանում են երկու մետրիկաներ՝ Precision և Recall (Նկար 2):




Նկար 2: Precision և Recall մետրիկաները պատկերված նկար 1-ում

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}$$

Կարող ենք տեսնել, որ երկուսի արժեքների տիրույթը 0-ից 1 է: Precision-ը մեծագույն արժեքը ընդունում է, երբ FP=0, իսկ Recall-ը՝ FN=0 դեպքում: Մարդուն առողջ կամ վարակված դասակարգելիս ավելի մեծ ուշադրություն ենք դարձնելու Recall-ի արժեքին, ինչքան մեծ լինի, այնքան ավելի քիչ վարակված մարդկանց ենք դասակարգել, որպես առողջ: Այս երկու մետրիկաները կարող ենք միավորել մեկի մեջ: Այն անվանում են F1 արժեք (score):

$$F1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

F1 score-ը հավասար է Precision-ի և Recall-ի միջին հարմոնիկին: Իսկ ինչո՞ւ չենք կարող դիտարկել հանրահաշվական միջինը, հարմոնիկ միջինի փոխարեն: Դիտարկենք մի դեպք, երբ Precision = 0.1 և Recall = 0.95: Այդ դեպքում նրանց հանրահաշվական միջինը կլինի 0.525, մինչդեռ հարմոնիկ միջինը կստացվի մոտավորապես 0.18: Այսինքն եթե երկու մետրիկաներից թեկուզ մեկը փոքր լինի, F1 score-ի արժեքը շատ կփոքրանա:

## 2 Մետրիկա անկախ շեմից

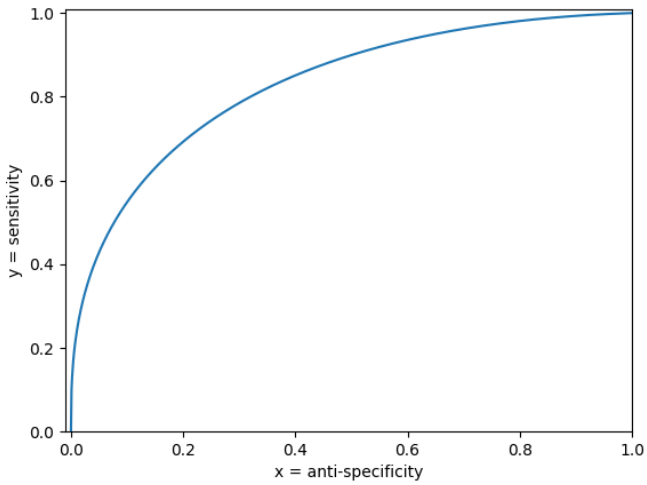
Ունենք sigmoid ակտիվացիոն ֆունկցիայով մոդել: Մոդելը վերադարձնում է 0-ից մեկ միջակայքի թիվ: Մենք որոշում ենք շեմ (threshold), որից մեծ լինելու դեպքում դասակարգում ենք առաջին class-ին, իսկ փոքր լինելու դեպքում դասակարգում ենք, որպես երկրորդ class: Շեմից կախված class-ը կարող ենք գրել հետևյալ pseudocode-ով:

```
y_pred = sigmoid(x)
if y_pred > threshold:
  predicted_class = positive
else:
  predicted_class = negative
```

Այսինքն ստացանք, որ threshold-ից կախված կփոխվեն մեր իմացած մետրիկաների արժեքները (accuracy, precision, recall, F1 score): Այդ պատճառով ներմուծում ենք նոր արժեքներ, որը մեզ կասի, թե որ շեմի դեպքում կունենանք ամենամեծ ճշգրտությունը: Ծանոթանանք երկու տերմինների հետ:

1. TPR (True Positive Rate) = Recall = Sensitivity = $\frac{TP}{TP + FN}$
2. FPR (False Positive Rate) = 1 - Specificity = 1 - $\frac{TN}{TN + FP}$ = $\frac{FP}{TN + FP}$

Դիտարկենք երկու շեմի դեպք։ Երբ threshold = 0 և երբ threshold=1։ Threshold=0, նշանակում է որ մենք միշտ բոլոր տվյալներին ասել ենք positive (վարակված)։ Դա նշանակում է՝ FN = 0, TN = 0 $\rightarrow$ TPR = 1, FPR = 1։ Երբ threshold = 1՝ բոլոր տվյալներին ասել ենք negative։ Դա նշանակում է՝ FP = 0, TP = 0 $\rightarrow$ TPR = 0, FPR = 0։ Այս երկու առանցքներից կազմված գրաֆիկը կոչվում է ROC curve (Նկար 3)։

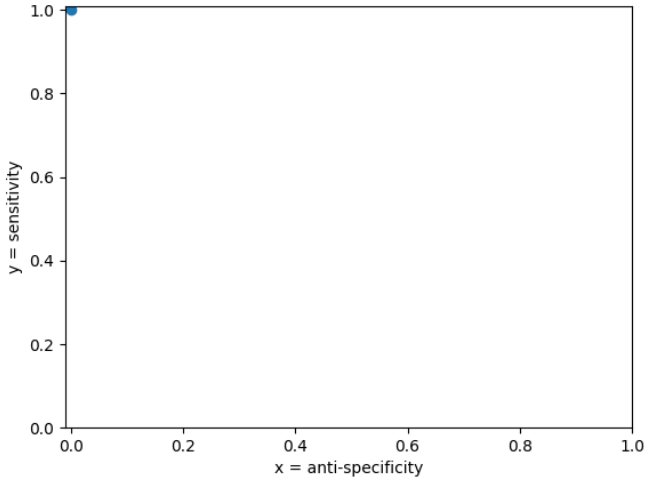



Նկար 3։ TPR և FPR առանցքներից կախված գրաֆիկ

Այս գրաֆիկում (0, 0) կետը համապատասխանում է threshold=1 դեպքին, իսկ (1, 1) կետը՝ threshold=0 դեպքին։ Ձախից աջ շարժվելիս threshold-ը փոքրանում է։ Դրա պատճառն այն է, որ threshold-ը ինչքան փոքրացնենք, այնքան positive-ների (TP, FP) քանակը կշատանա։ Իսկ ե՞րբ այս գրաֆիկը կունենա ամենալավ տեսքը։ Այս գրաֆիկը ստանում ենք ընտրելով տարբեր threshold-ներ և դրանց համար հաշվել TPR-ը և FPR-ը։ Այսինքն կարող ենք ունենալ threshold-ների զանգված, որոնք 0-ից աճեն մինչև 1՝ 0.05 փոփոխությամբ։

$$thresholds = [0,\ 0.05,\ 0.1,\ \ldots,\ 0.95,\ 1]$$

Մեր նպատակն է TPR-ը դարձնել 1 (FP = 0), իսկ FPR-ը թողնել 0 (FP = 0), ցանկացած threshold-ի դեպքում։ Այդ գրաֆիկը կունենա նկար 4-ի տեսքը։



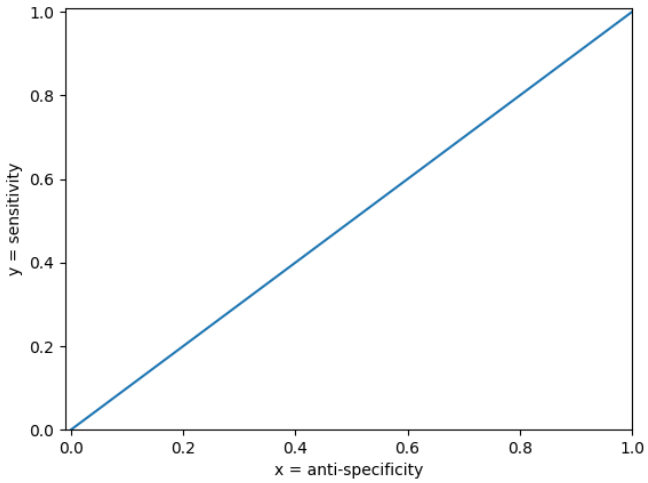

Նկար 4։ Անկախ threshold-ից՝ TPR = 1

Նկար 3-ի ROC curve-ի դեպքում մեզ անհրաժեշտ է ընտրել այն threshold-ը, որի դեպքում մենք ամենամոտն ենք (0, 1) կետին (TPR=1, FPR=0)։ Քանի որ մեր գրաֆիկի կետերը վերջավոր են (thresholds-ը զանգված է), կարող ենք ամեն threshold-ի արժեքի համար հաշվել TPR-ը և FPR-ը և հեռավորություն հաշվել (0, 1) կետից։ Որ threshold-ի դեպքում այդ հեռավորությունը եղավ ամենափոքրը՝ դա էլ կվերցնենք։ ROC curve-ի միջոցով ընտրում ենք threshold-ը, որի դեպքում ունենք ամենալավ արդյունքը։ Հիմա ներմուծենք մի մետրիկա, որը անկախ threshold-ից կվերադարձնի մի թիվ և դրա միջոցով կհասականանք մոդելի ճշգրտությունը։ Այն կոչվում է AUC (Area Under the Curve)։ ROC curve-ը ստանալուց հետո, հաշվում ենք գրաֆիկով և x-երի առանցքով սահմանափակված պատկերի մակերեսը։ Դիտարկենք AUC-ի երկու դեպք։

1. AUC $\approx 0.5$ (Նկար 5)։ Վերցնենք հավասարաչափ տվյալներ (10 positive, 10 negative)։ Նկար 1-ից և այս փաստից կարող ենք ասել, որ $TP + FN = TN + FP$։ Եթե AUC = 0.5, նշանակում է.

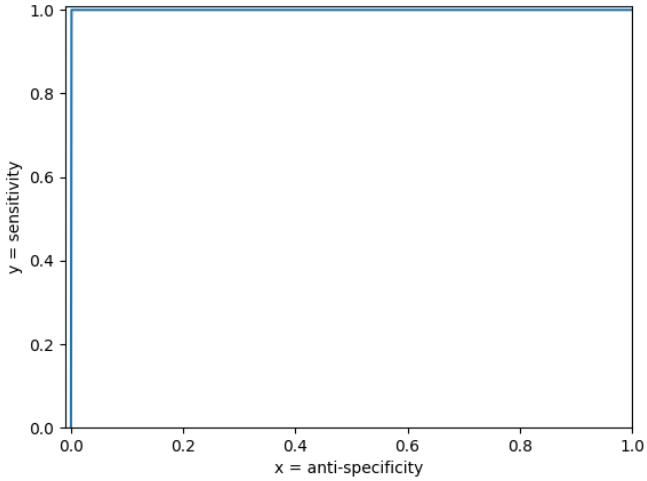
$$TPR = FPR$$
$$\frac{TP}{TP + FN} = \frac{FP}{TN + FP}$$
Քանի որ $TP + FN = TN + FP$
$$TP = FP$$

   Ստացանք, որ TP-ների քանակը հավասար է FP-ներին, որը նշանակում է տվյալների մի մասին ճիշտ ենք ասել, մյուս մասին սխալ։ Մոդելի ճշգրտությունը 50%-է, որը նույն պատահական դասակարգչի (random classifier) դեպքն է։




Նկար 5։ Պատահական դասակարգիչ՝ AUC=0.5

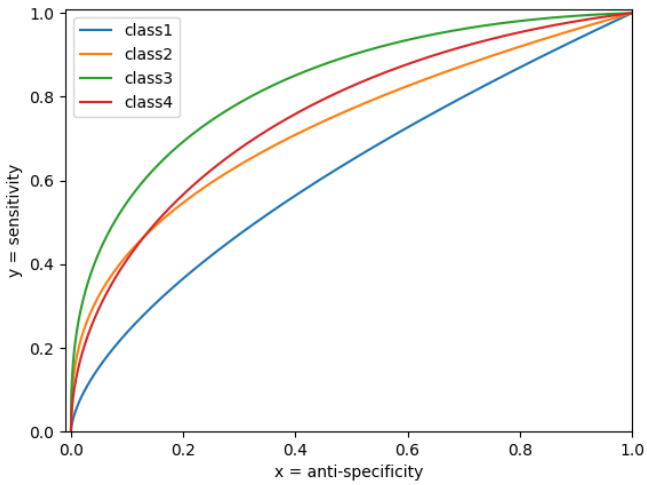
2. AUC $\approx 1$։ Սա նշանակում է, որ գոյություն ունի threshold, որի դեպքում TPR = 1 և FPR = 0։ Այս դեպքում կամ կստանանք նկար 4-ի դեպքը կամ կստանանք ուղղանկյուն (Նկար 6)։




Նկար 6: threshold-ը 1-ից փոքր դեպքերում TPR=1

Նկար 6-ը նշանակում է, որ երբ մոդելը մարդուն ասել է առողջ՝ վերադարձրած հավանականությունը եղել է 1, իսկ երբ ասել է վարակված՝ վերադարձրած հավանականությունը եղել է 0: Այն շատ վստահ է եղել իր որոշումների մեջ: Այս դեպքում նույնպես վերցնելու ենք (0, 1) կետին ամենամոտ threshold-ը:

Իսկ ինչպե՞ս կարող ենք մի քանի class-երի դեպքում ընտրել threshold: Մի քանի class-ի դեպքում կիրառելու ենք one vs all մեթոդը: Մեր առաջին class-ը դիտարկելու ենք positive, իսկ մնացած class-երը negative: Ամեն class-ի համար կառուցելու ենք ROC curve (Նկար 7) և վերջում ընտրելու ենք այն threshold-ը, որի դեպքում բոլոր գրաֆիկների վրա ընտրված կետը մոտ կլինի (0, 1) կետին: Threshold-ի ընտրելու ընթացքը տեղի է ունենալու հետևյալ կերպ: Մեր thresholds զանգվածի ամեն անդամի համար հաշվելու ենք բոլոր class-երի TPR-ի և FPR-ի արժեքը և հեռավորությունը (0, 1) կետից: Հաշված արժեքները միջինացնելու ենք և այդ պահի threshold-ի համար կունենանք հեռավորություն: Հետո նույն գործողությունը կատարելու ենք զանգվածի մյուս անդամների համար և վերջում ընտրելու ենք փոքրագույն հեռավորություն ունեցող threshold-ը:




Նկար 7: ROC curve մի քանի class-ի դեպքում

Սա տարբերակներից մեկն է:

Մյուս տարբերակն է ամեն class-ի համար ընտրել առանձին threshold։ Օրինակ պետք է կատարենք (1. շուն, 2. կատու, 3. փիղ, 4. ոչ մի բան) դասակարգում։ Շան համար threshold=0.7, կատվի համար՝ 0.6, փղի համար՝ 0.65։ Երբ մոդելը վերադարձնի, որ p հավանականությամբ նկարում շուն է պատկերված, ստուգելու ենք $p > 0.7$ պայմանը։ True արդյունքի դեպքում կասենք նկարում շուն է պատկերված, հակառակ դեպքում կասենք ոչ մի բան պատկերված չէ նկարում։