Linear Classification & Logistic Regression

Vladislav Goncharenko
ML researcher

girafe ai

MSU, spring 2024

Recap

Lecture 2:
Linear Regression

- Regression model
- Linear models overview
- Linear regression
- Gauss-Markov theorem
- Regularizations
- Model validation and evaluation

Outline

- Linear classification
  - margin
  - loss functions
- Logistic regression
  - sigmoid derivation
  - Maximum Likelihood Estimation (MLE)
  - logistic loss
  - probability calibration
- Multiclass aggregation strategies
  - One vs Rest
  - One vs One
- Metrics in classification
  - Accuracy, Balanced accuracy
  - Precision, Recall, F-score
  - ROC curve, PR curve, AUC
  - Confusion matrix
- Hardware for ML

Linear Classification

girafe ai

01

Classification problem

$$X \in R^{n \times p}$$

$$Y \in C^n$$

$$|C| < +\infty$$

e.g. $C = \{-1, 1\}$

$$c(X) = \hat{Y} \approx Y$$

Variants of labels

- Binary - choice between two classes, easier for theoretical analysis also widely used in practice
- Multiclass - choose one of many classes
- Multilabel - choose several of many classes

Linear classifier

The most simple linear classifier

$$c(x) = \begin{cases} 1, & \text{if } f(x) \geq 0 \\ -1, & \text{if } f(x) < 0 \end{cases}$$

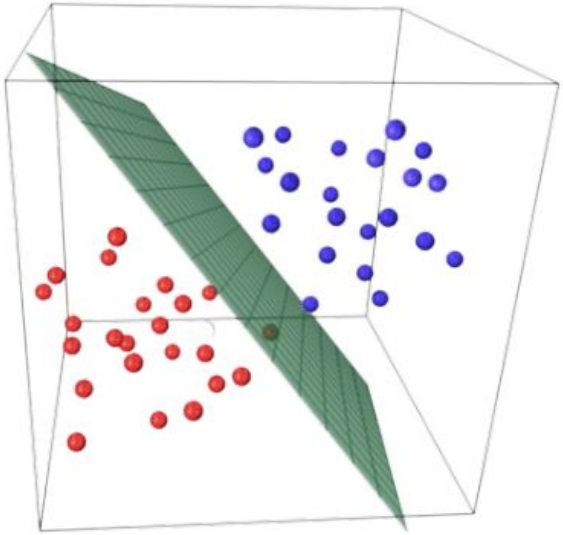
or equivalently

$$c(x) = \text{sign}(f(x)) = \text{sign}(x^T w)$$

Why cutoff value is fixed?

Bias term is implied

Geometrical interpretation



Geometrical interpretation: hyperplane dividing space into two subspaces

$$c(x) = \text{sign}(f(x)) = \text{sign}(x^T w)$$

Margin

Let’s define linear model’s Margin as

$$M_i = y_i \cdot f(x_i) = y_i \cdot x_i^T w$$

main property:

negative margin reveals misclassification

$$M_i > 0 \Leftrightarrow y_i = c(x_i)$$

$$M_i \leq 0 \Leftrightarrow y_i \neq c(x_i)$$

Margin is used in theoretical proofs

Weights choice

Remembering old paradigm

Empirical risk = $\sum_{\text{by object}}$ Loss on object $\longrightarrow$ Min model params

Essential error is misclassification

$$L_{\text{mis}}(y_i^t, y_i^p) = [y_i^t \neq y_i^p] = [M_i \leq 0]$$

Disadvantages

- Not differentiable
- Overlooks confidence

Solution:

estimate it with a smooth function

Iverson bracket $[P] = \begin{cases} 1, & \text{if } P \text{ is true} \\ 0, & \text{otherwise} \end{cases}$

Square loss

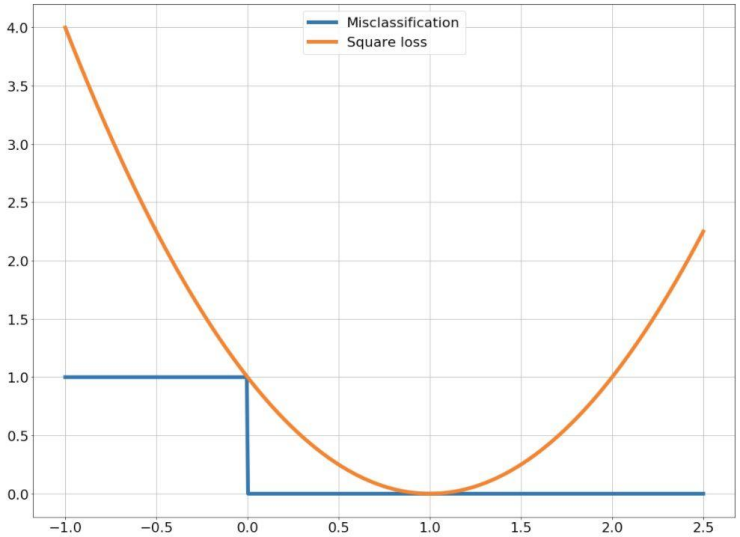
Let's treat classification problem as regression problem:

thus we optimize MSE

$$Y \in \{-1, 1\} \mapsto Y \in R$$

$$L_{\text{MSE}} = (y_i - x_i^T w)^2 = \frac{(y_i^2 - y_i \cdot x_i^T w)^2}{y_i^2} =$$

$$= (1 - y_i \cdot x_i^T w)^2 = (1 - M_i)^2$$

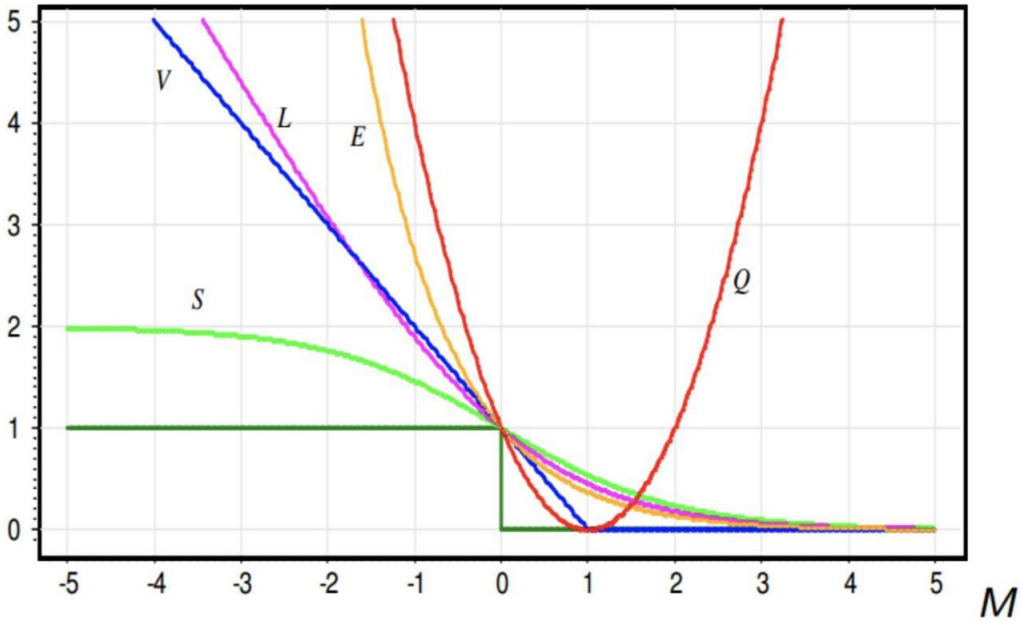


Advantage: already solved

Disadvantage: penalizes for high confidence

No one uses it in practice!

Other losses



$$Q(M) = (1 - M)^2$$
$$V(M) = (1 - M)_+$$
$$S(M) = 2(1 + e^M)^{-1}$$
$$L(M) = \log_2(1 + e^{-M})$$
$$E(M) = e^{-M}$$

Loss functions for classification

Logistic Regression

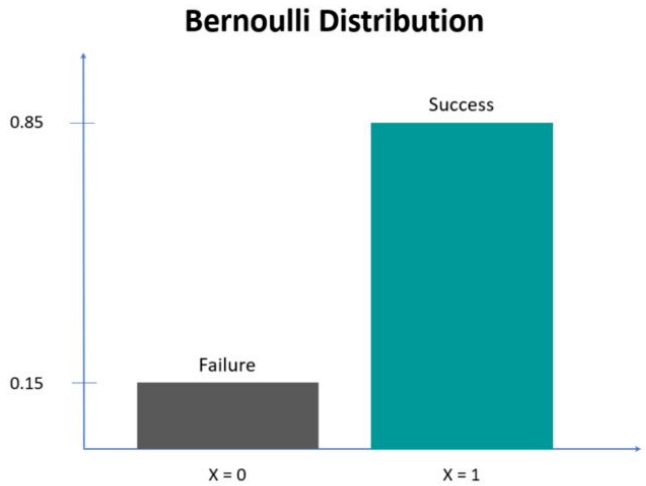
girafe
ai

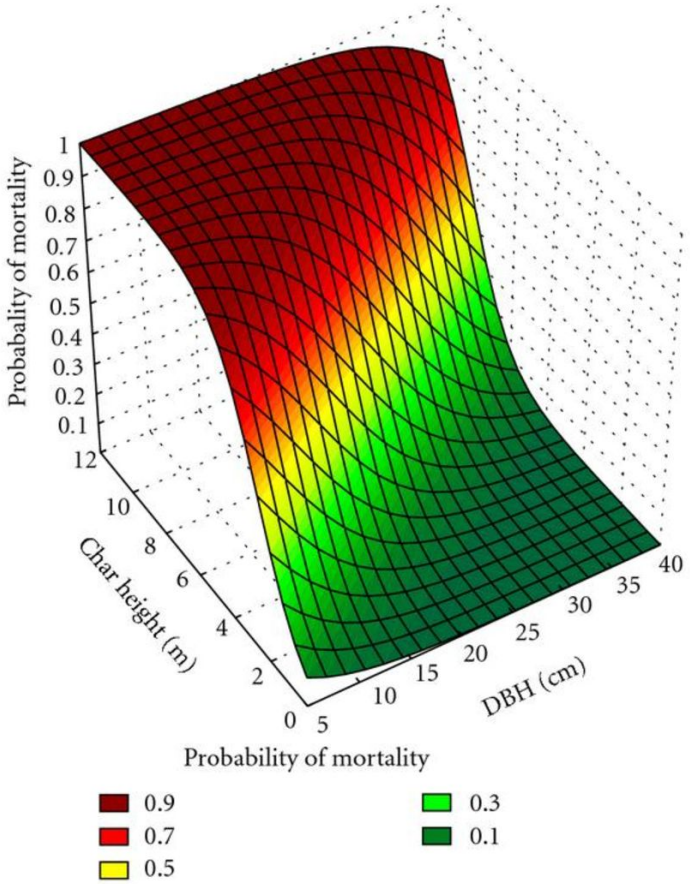
02

Basic assumption

of Logistic regression is that every point of feature space have associated probability of being certain class.

So LogReg generates Bernoulli distribution for each point (since we have two classes).





Intuition

I. Let's try to predict probability of an object to have positive class

$$p_+ = P(y = 1|x) \in [0, 1]$$

II. But all we can predict is a real number!

$$y = x^T w \in R$$

III. Time for some tricks

$$\frac{p_+}{1 - p_+} \in [0, +\infty)$$

$$\log \frac{p_+}{1 - p_+} \in R$$

Here is the match

This is called **logit** or **log-odds**

IV. Reverse to closed form

$$\frac{p_+}{1 - p_+} = \exp(x^T w)$$

$$p_+ = \frac{1}{1 + exp(-x^T w)} = \sigma(x^T w)$$

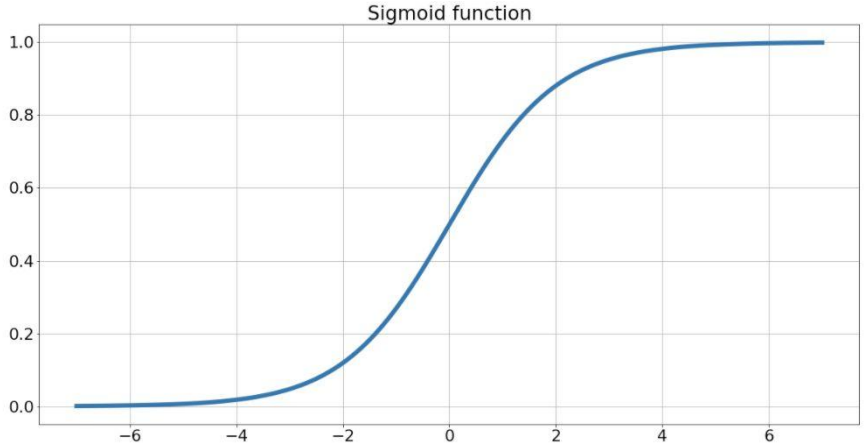
Sigmoid (aka logistic) function

$$\sigma(x) = \frac{1}{1 + exp(-x)}$$

Sigmoid is odd relative to (0, 0.5) point

Symmetric property:

$$1 - \sigma(x) = \sigma(-x)$$



Derivative: $\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$

MLE for Logistic Regression

Just to remind

$$\log L(w|X,Y) = \log P(X,Y|w) = \log \prod_{i=1}^{n} P(x_i, y_i|w)$$

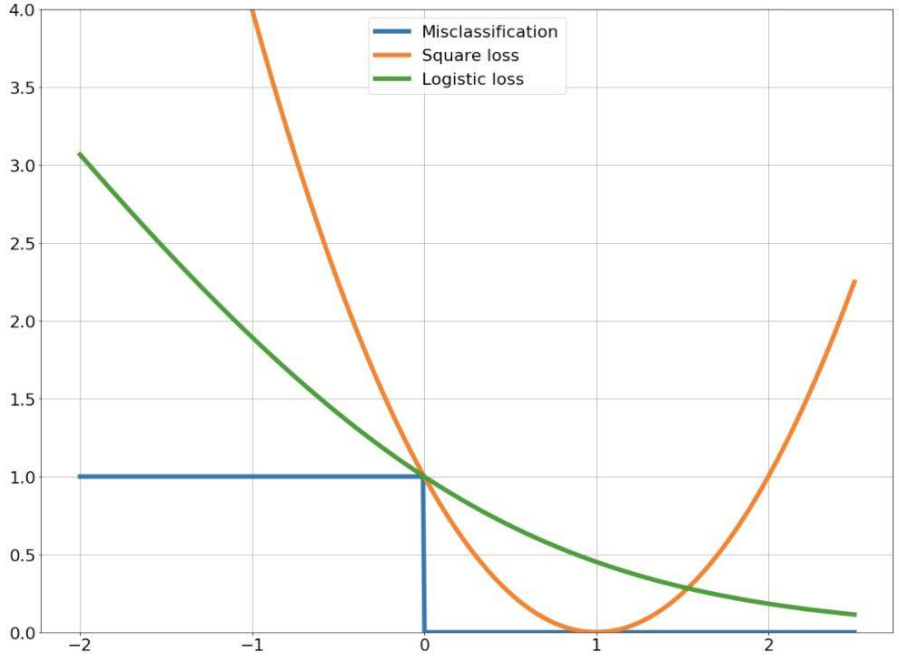
Calculating probabilities for objects (which are modelled as Bernoulli variables)

$$\begin{aligned} \text{if } y_i = 1: &\quad P(x_i, 1|w) = \sigma_w(x_i) = \sigma_w(M_i) \\ \text{if } y_i = -1: &\quad P(x_i, -1|w) = 1 - \sigma_w(x_i) = \sigma_w(-x_i) = \sigma_w(M_i) \end{aligned}$$

$$\log L(w|X,Y) = \sum_{i=1}^{n} \log \sigma_w(M_i) = -\sum_{i=1}^{n} \log(1 + \exp(-M_i)) \to \min_w$$

Logistic loss

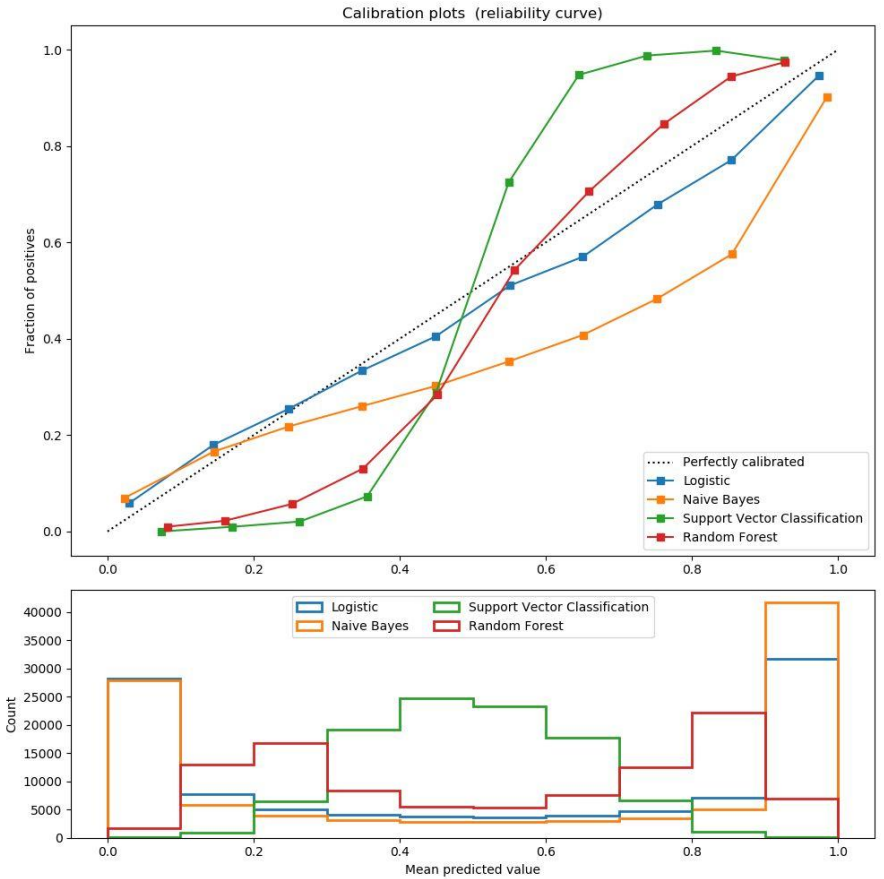
$$L_{Logistic} = \log(1 + \exp(-M_i))$$



Probability calibration

By using Logistic Regression
we generate a Bernoulli distribution
in each point of space, so in practice it
means calibrated classifier

[Calibration discussion]



Generalized linear model

Common distributions with typical uses and canonical link functions

Distribution	Support of distribution	Typical uses	Link name	Link function, $\mathbf{X}\boldsymbol{\beta} = g(\mu)$	Mean function
Normal	real: $(-\infty, +\infty)$	Linear-response data	Identity	$\mathbf{X}\boldsymbol{\beta} = \mu$	$\mu = \mathbf{X}\boldsymbol{\beta}$
Exponential	real: $(0, +\infty)$	Exponential-response data, scale parameters	Negative inverse	$\mathbf{X}\boldsymbol{\beta} = -\mu^{-1}$	$\mu = -(\mathbf{X}\boldsymbol{\beta})^{-1}$
Gamma					
Inverse Gaussian	real: $(0, +\infty)$		Inverse squared	$\mathbf{X}\boldsymbol{\beta} = \mu^{-2}$	$\mu = (\mathbf{X}\boldsymbol{\beta})^{-1/2}$
Poisson	integer: $0, 1, 2, \ldots$	count of occurrences in fixed amount of time/space	Log	$\mathbf{X}\boldsymbol{\beta} = \ln(\mu)$	$\mu = \exp(\mathbf{X}\boldsymbol{\beta})$
Bernoulli	integer: $\{0, 1\}$	outcome of single yes/no occurrence		$\mathbf{X}\boldsymbol{\beta} = \ln\left(\frac{\mu}{1-\mu}\right)$	
Binomial	integer: $0, 1, \ldots, N$	count of # of "yes" occurrences out of N yes/no		$\mathbf{X}\boldsymbol{\beta} = \ln\left(\frac{\mu}{n-\mu}\right)$	

Depending on which distribution we want to predict we have different forms of parameters encoded to formula.

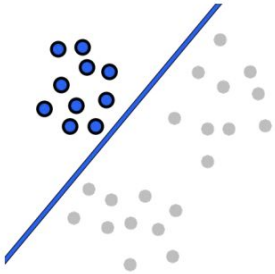
See https://en.wikipedia.org/wiki/Generalized_linear_model#Link_function

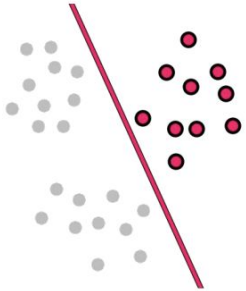
Multiclass aggregation strategies

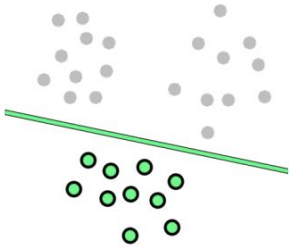
girafe ai

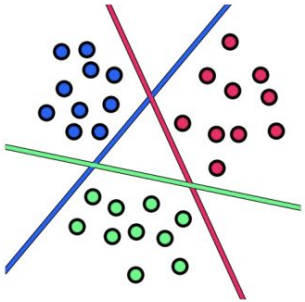
03

One vs Rest training

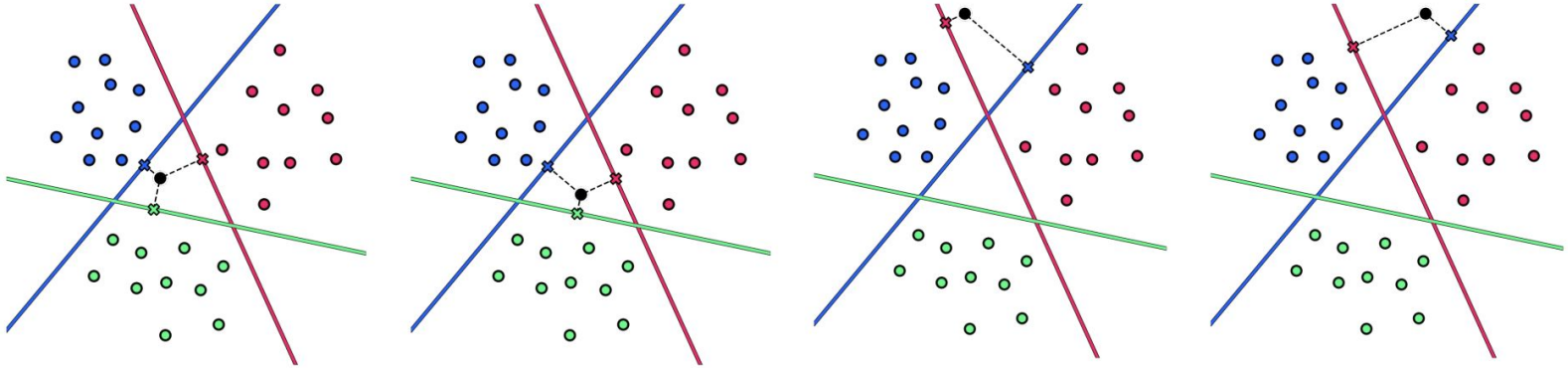






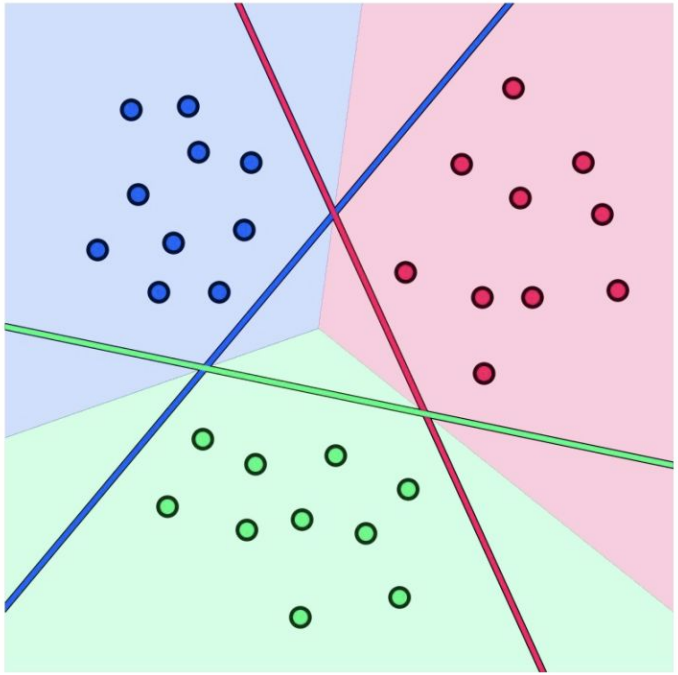


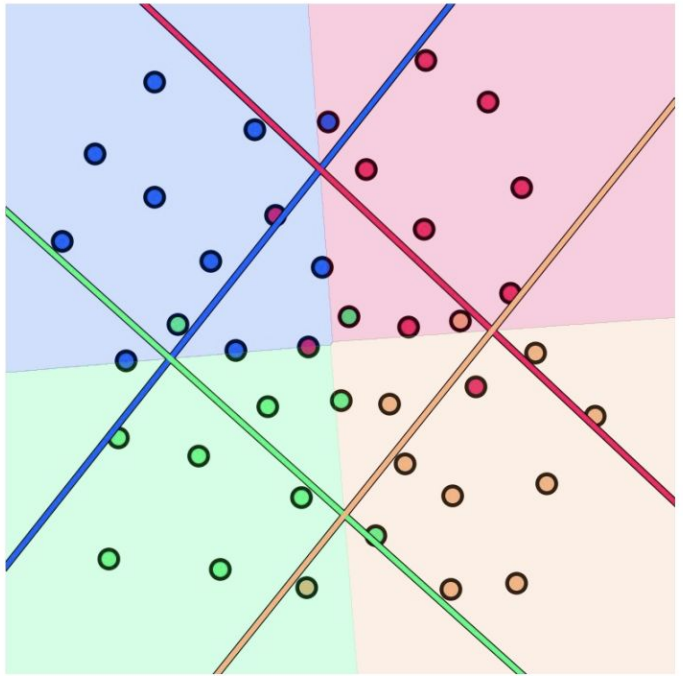
One vs Rest inference algorithm



Take class with max margin

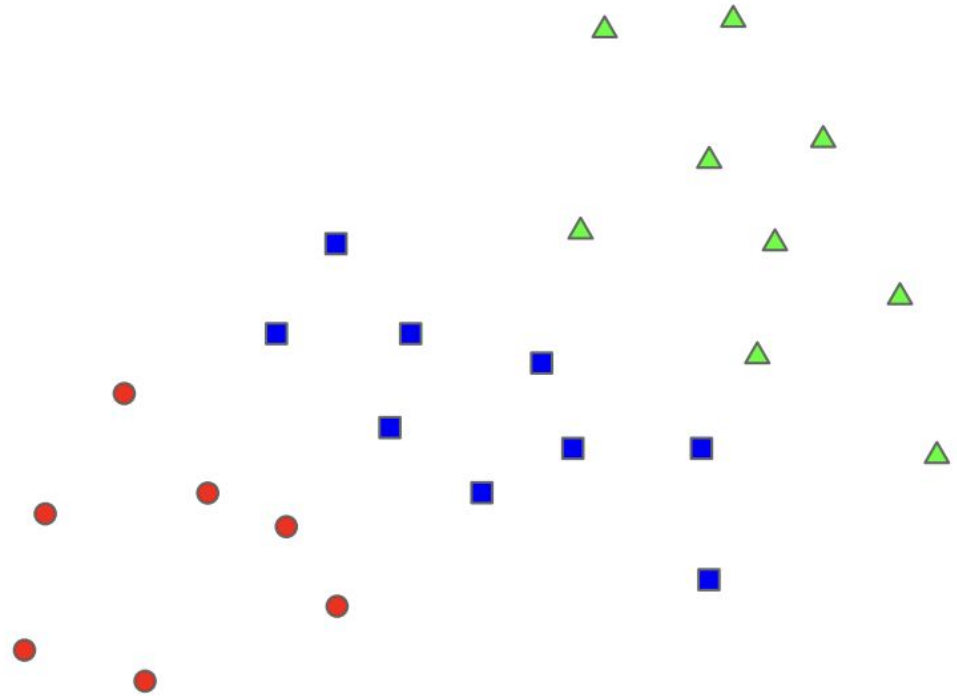
One vs Rest final result





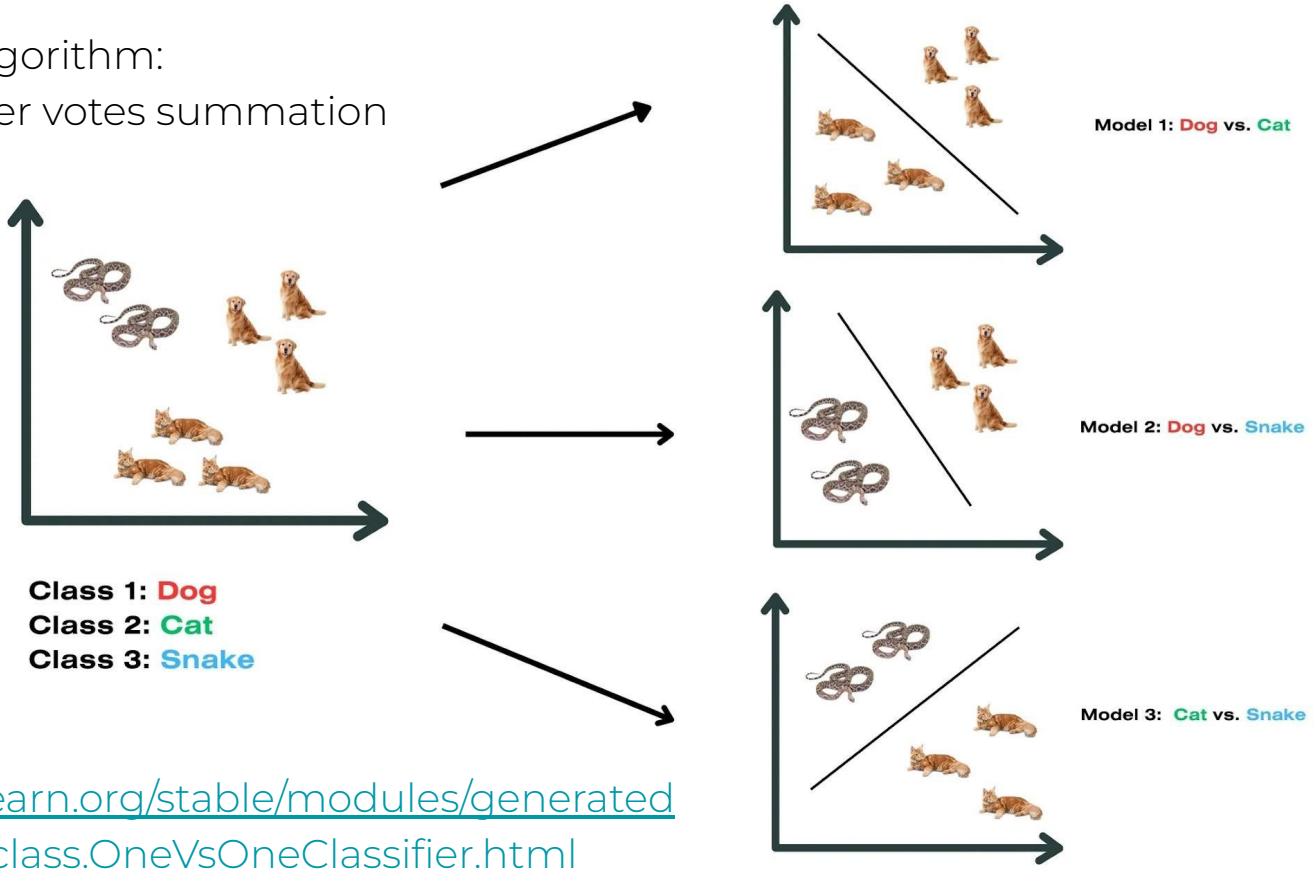
https://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsRestClassifier.html

One vs Rest Failure case?



One vs One

Inference algorithm:
base classifier votes summation



https://scikit-learn.org/stable/modules/generated/sklearn.multiclass.OneVsOneClassifier.html

Summary

	One vs Rest	One vs One
#classifiers	k	k(k-1)/2
dataset for each	full	subsampled

Metrics in classification

girafe ai

04

Metrics

- Accuracy
  - Balanced accuracy
- Confusion matrix
- Precision, Recall and friends
- F-score
- ROC curve
  - ROC-AUC
- PR curve
  - PR-AUC
- Multiclass generalizations
- Confusion matrix

Accuracy

Ratio of right classifications

$$\text{Accuracy} = \frac{1}{n}\sum_{i=1}^{n}[y_i^t = y_i^p]$$

target: 1 0 1 0 0 0 0 1 0 0

predicted: 0 0 1 0 0 0 0 1 1 0

accuracy = 8/10 = 0.8

$$\text{Balanced accuracy} = \frac{1}{C}\sum_{k=1}^{C}\frac{\sum_i[y_i^t = k \text{ and } y_i^t = y_i^p]}{\sum_i[y_i^t = k]}$$

Confusion matrix

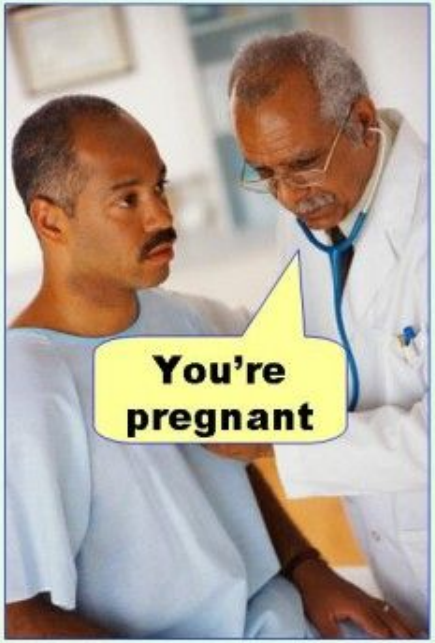
		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error
	Predicted condition negative	**False negative**, Type II error	**True negative**

https://en.wikipedia.org/wiki/Confusion_matrix

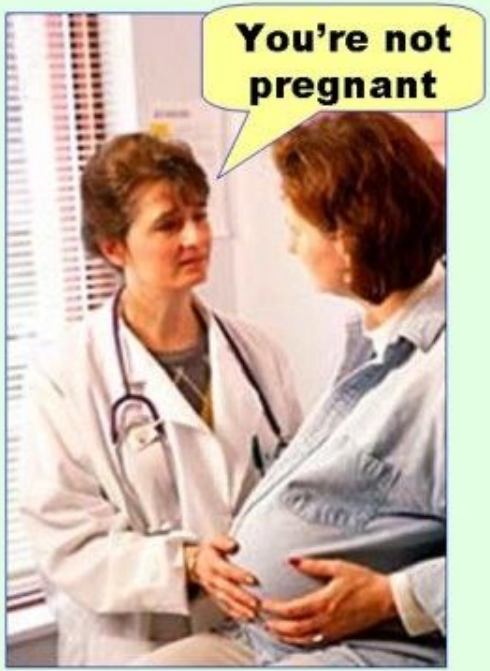
Types of errors

Positive condition is being pregnant as usual in medical tests

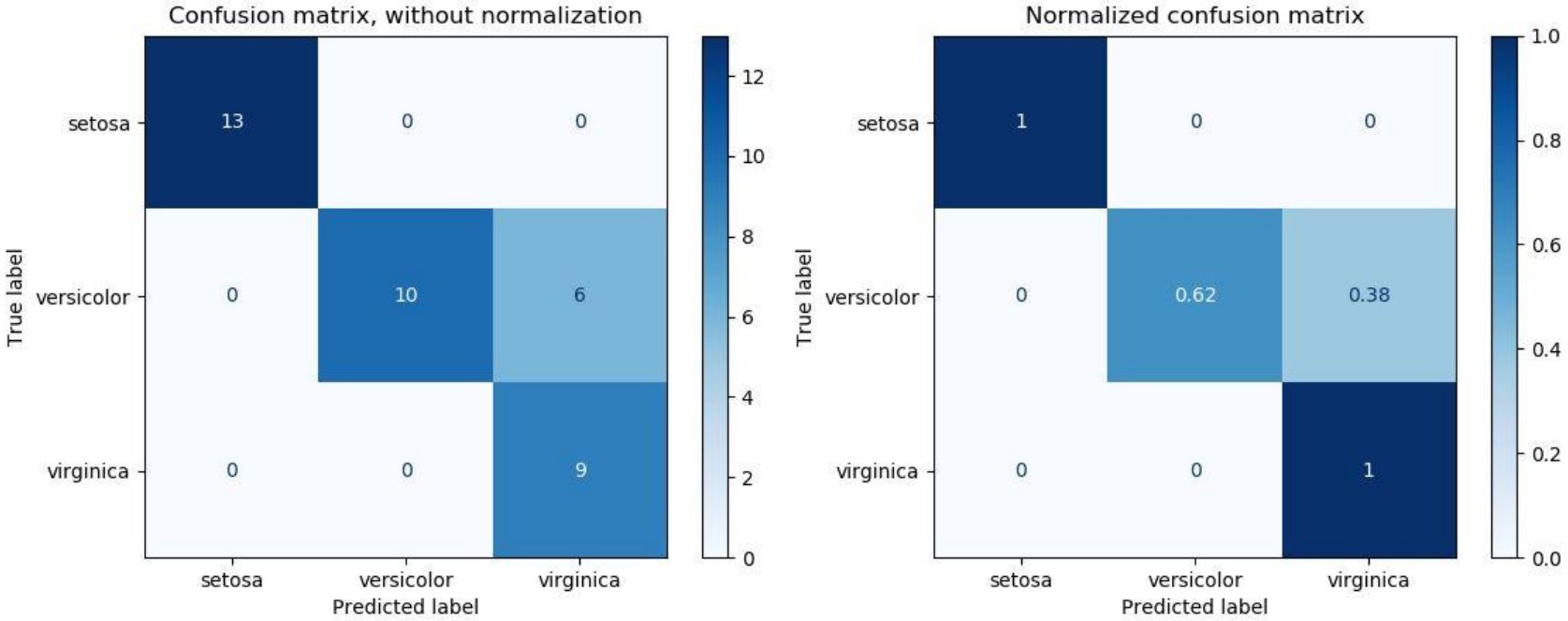
Type I error
(false positive)



Type II error
(false negative)

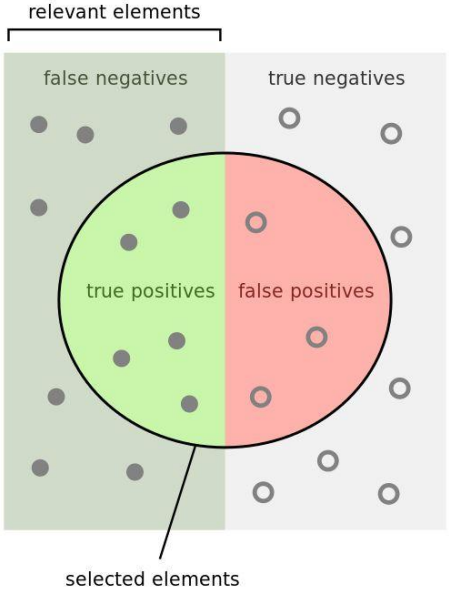


Confusion matrix for multiclass

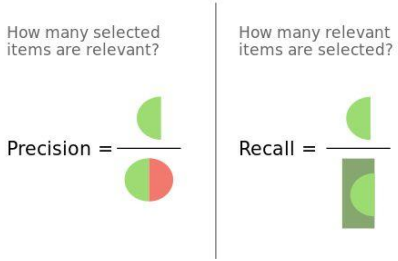


Precision and Recall

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error
	Predicted condition negative	**False negative**, Type II error	**True negative**

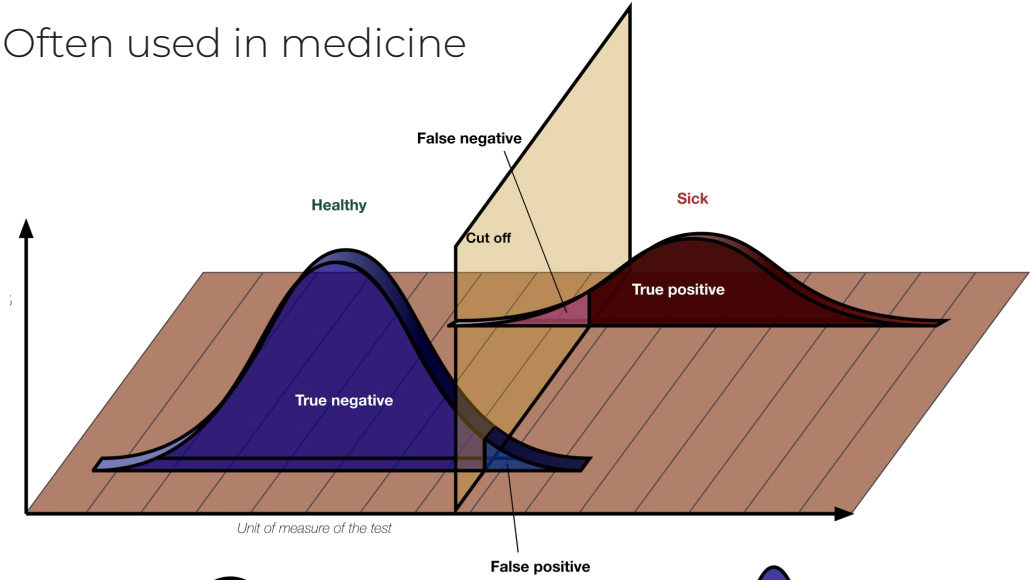


$$\text{Precision} = \frac{TP}{TP + FP} \qquad \text{Recall} = \frac{TP}{TP + FN}$$



Sensitivity and specificity

Often used in medicine

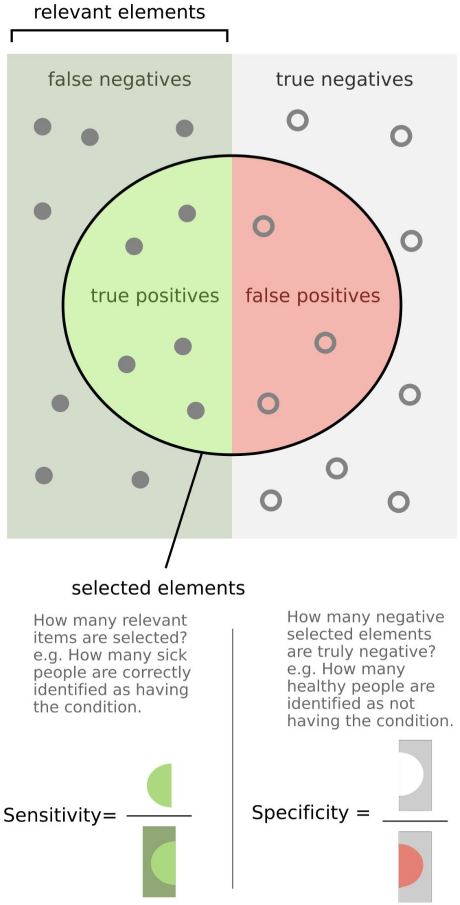


PPV =

NPV =

Sensitivity =

Specificity =



Plenty of other ratios

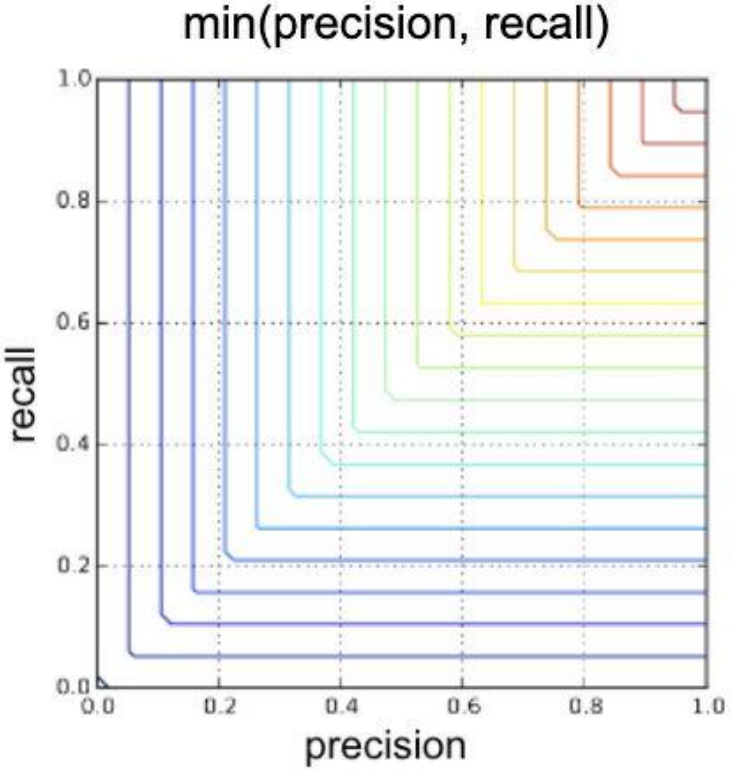
		Predicted condition			
	Total population $= P + N$	**Predicted positive (PP)**	**Predicted negative (PN)**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)** [a]	**True positive** (TP), hit[b]	**False negative** (FN), miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate type II error [c] $= \frac{FN}{P} = 1 - TPR$
	Negative (N)[d]	**False positive** (FP), false alarm, overestimation	**True negative** (TN), correct rejection[e]	False positive rate (FPR), probability of false alarm, fall-out type I error [f] $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2\,PPV \times TPR}{PPV + TPR} = \frac{2\,TP}{2\,TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

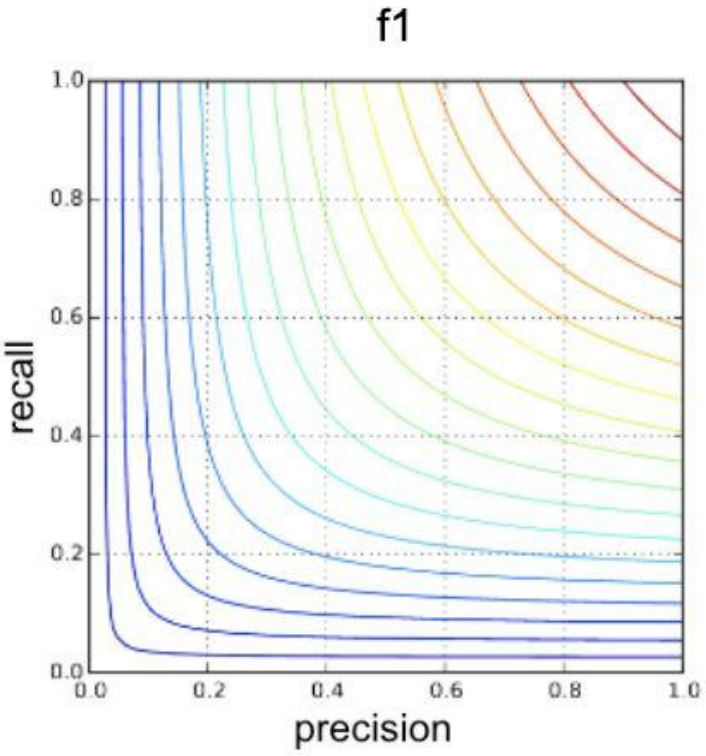
Sources: [12][13][14][15][16][17][18][19] view · talk · edit

Different disciplines invent their ratios to fit their problem setup

F-score motivation

We need the one metric to rule them all!





F-score

Harmonic mean of precision and recall

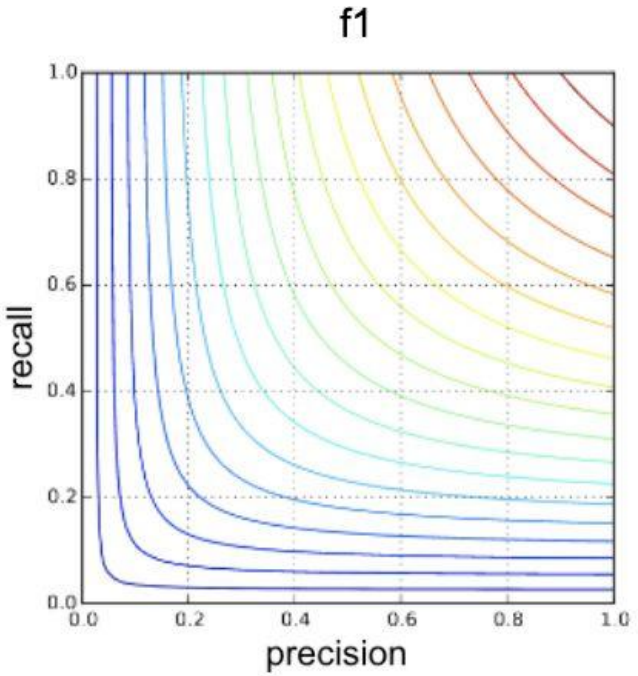
Tends to be closer to the smaller one

$$F_1 = \frac{2}{\text{precision}^{-1} + \text{recall}^{-1}} = 2\,\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Generalization to different ratio between

Precision and Recall

$$F_\beta = (1 + \beta^2)\,\frac{\text{precision} \cdot \text{recall}}{\beta^2\,\text{precision} + \text{recall}}$$



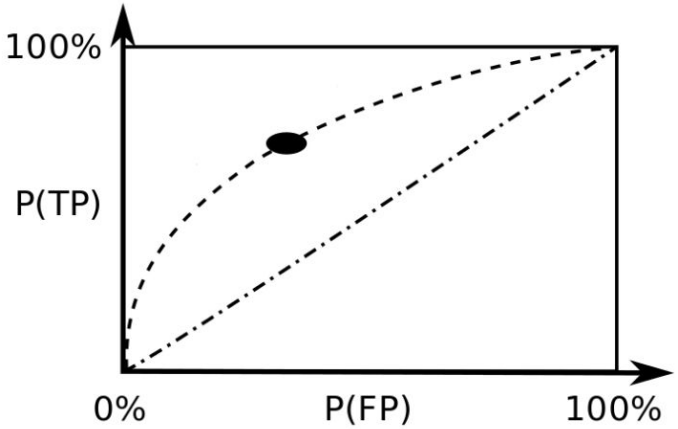
Examples of imbalanced errors

- Medicine
  - COVID test
- Security
  - Flood prediction (and other disasters)
- Banking and finance
  - Loans, mortgages

Equalization coefficient is estimated by risks of FP vs FN

Receiver Operating Characteristic (ROC)

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	**True positive**	**False positive**, Type I error
	Predicted condition negative	**False negative**, Type II error	**True negative**



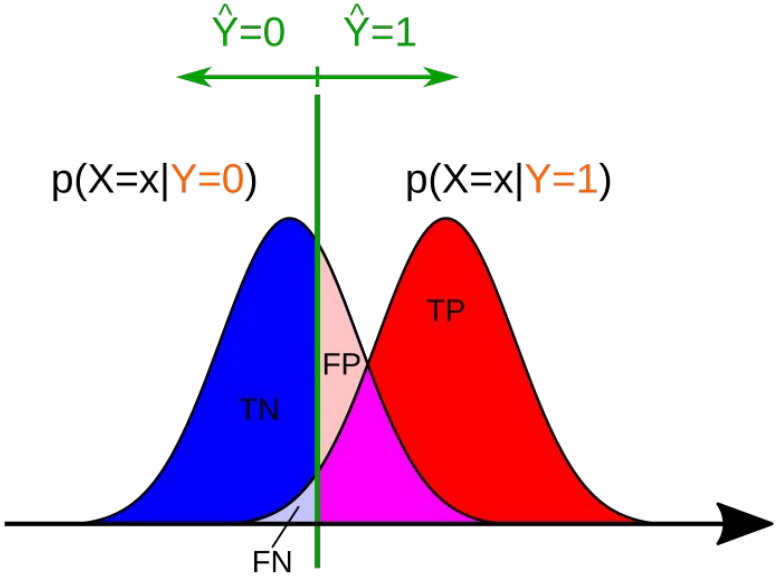
$$\mathrm{FPR} = \frac{FP}{FP + TN}$$

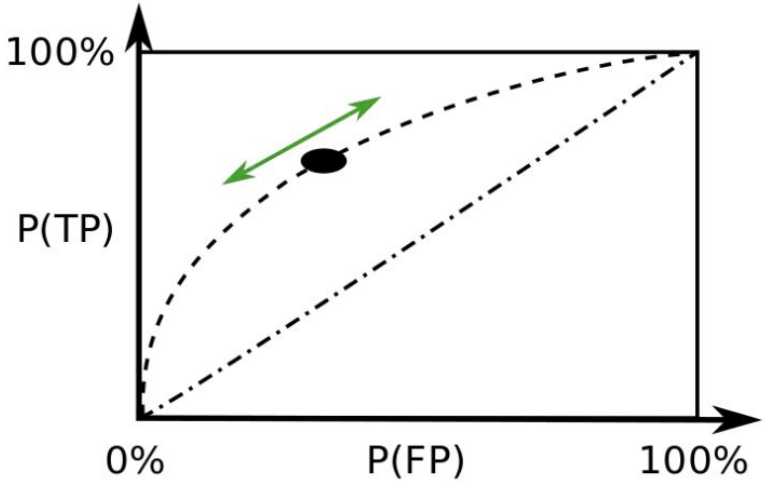
$$\mathrm{TPR} = \frac{TP}{TP + FN} (= \mathrm{Recall})$$

ROC calculation

Classifier have to predict probabilities

Objects get sorted by positive probability





Line is plotted as threshold moves

ROC properties

Baseline is random predictions.

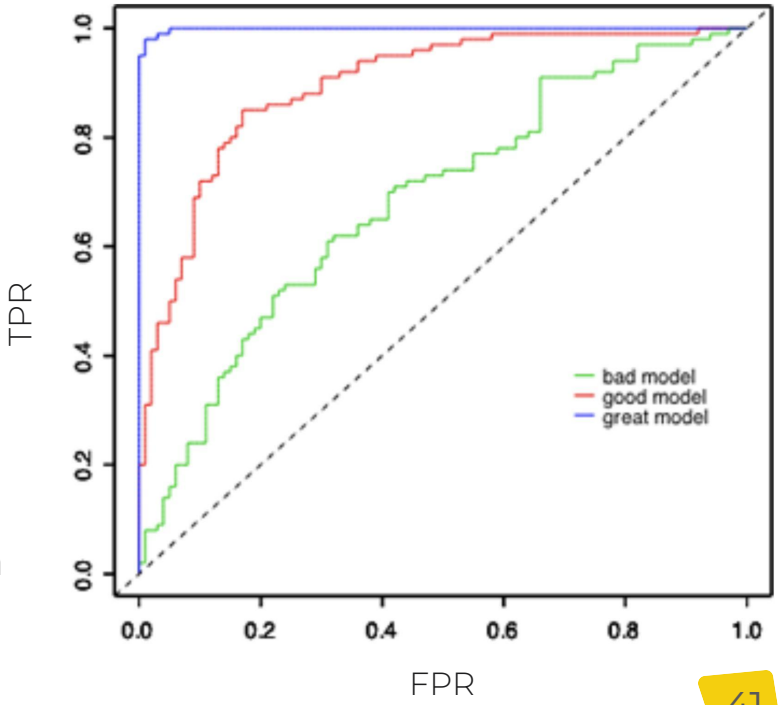
Always above diagonal (for reasonable classifier).

If below - change sign of predictions.

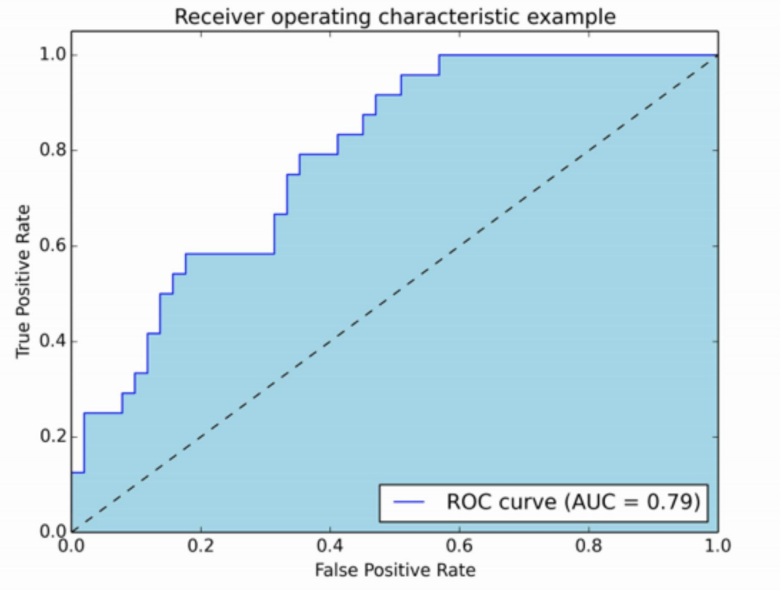
Strictly higher curve means better classifier.

Number of steps (thresholds) proportional to dataset size.

In practice thresholds are fixed before calculation



ROC Area Under Curve (ROC-AUC)

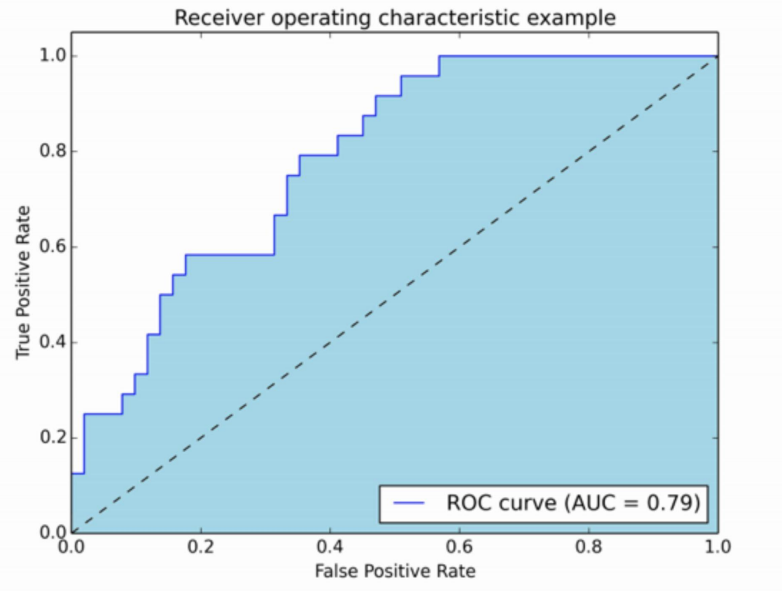


Effectively lays in (0.5, 1)

Bigger ROC-AUC doesn't imply higher curve everywhere

More explanations with pictures

ROC-AUC properties



Equal to fraction of correctly sorted paris
Because we compute it over predictions sorted by score.

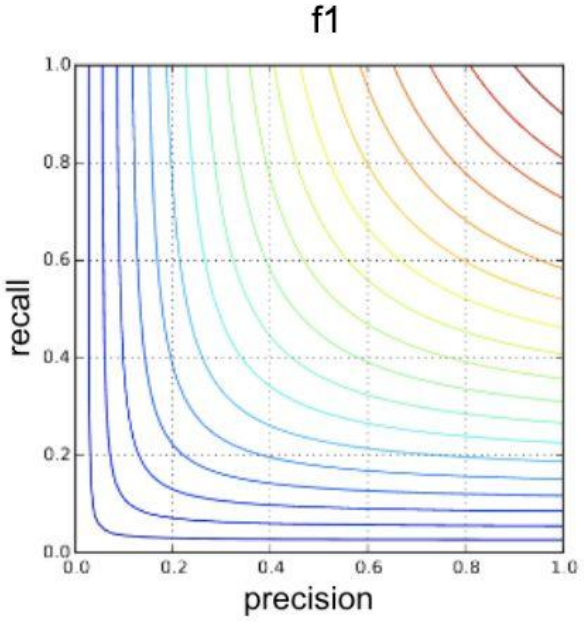
Scale-invariant
It measures how well predictions are ranked, rather than their absolute values.
If we multiply all predictions by constant metric will not change.

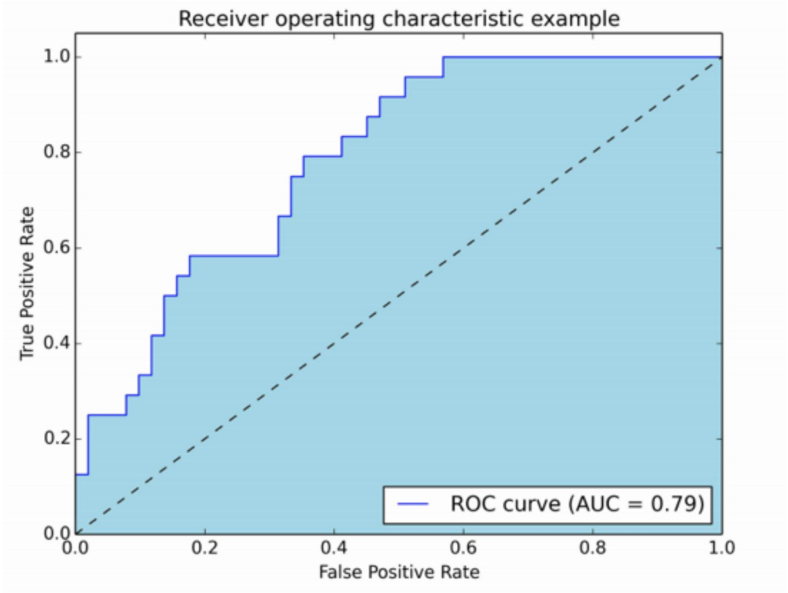
Classification-threshold-invariant
It measures the quality of the model's predictions irrespective of what classification threshold is chosen.

Source

F-score vs ROC-AUC

Which one to tune?





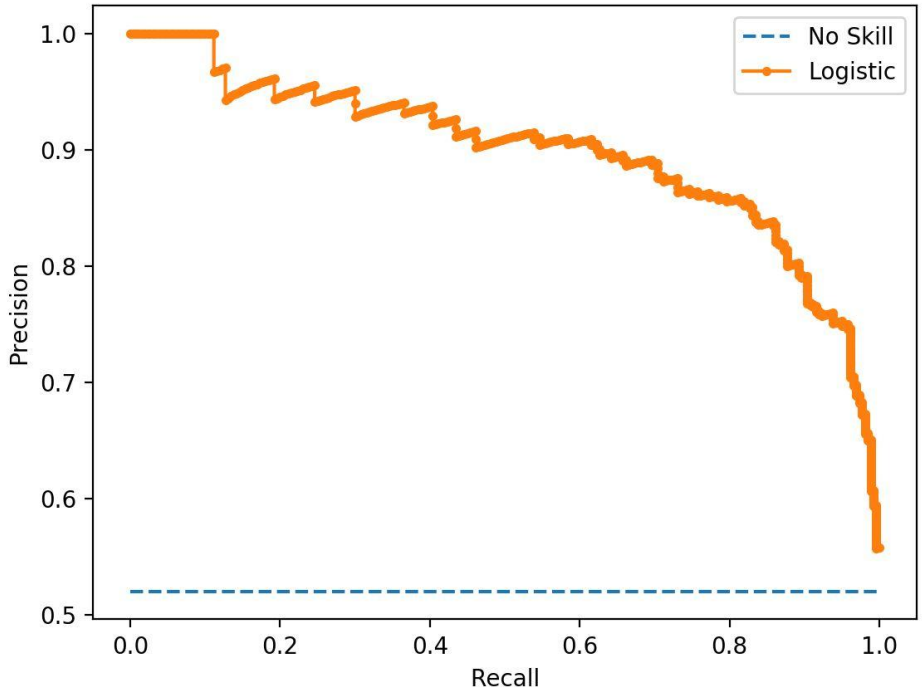
Precision-Recall Curve

AUC is in (0, 1)

Source of AP metric

(important for next semester)

Nice article



Multiclass metrics

As with linear models we need some magic to measure multiclass problems

Basically it's mean of one or another kind

Detailed info here and here

average	Precision	Recall	F_beta
"micro"	$P(y, \hat{y})$	$R(y, \hat{y})$	$F_\beta(y, \hat{y})$
"samples"	$\frac{1}{\lvert S\rvert}\sum_{s\in S} P(y_s, \hat{y}_s)$	$\frac{1}{\lvert S\rvert}\sum_{s\in S} R(y_s, \hat{y}_s)$	$\frac{1}{\lvert S\rvert}\sum_{s\in S} F_\beta(y_s, \hat{y}_s)$
"macro"	$\frac{1}{\lvert L\rvert}\sum_{l\in L} P(y_l, \hat{y}_l)$	$\frac{1}{\lvert L\rvert}\sum_{l\in L} R(y_l, \hat{y}_l)$	$\frac{1}{\lvert L\rvert}\sum_{l\in L} F_\beta(y_l, \hat{y}_l)$
"weighted"	$\frac{1}{\sum_{l\in L}\lvert\hat{y}_l\rvert}\sum_{l\in L}\lvert\hat{y}_l\rvert P(y_l, \hat{y}_l)$	$\frac{1}{\sum_{l\in L}\lvert\hat{y}_l\rvert}\sum_{l\in L}\lvert\hat{y}_l\rvert R(y_l, \hat{y}_l)$	$\frac{1}{\sum_{l\in L}\lvert\hat{y}_l\rvert}\sum_{l\in L}\lvert\hat{y}_l\rvert F_\beta(y_l, \hat{y}_l)$

Multiclass metrics

- **macro** simply calculates the mean of the binary metrics, giving equal weight to each class
- **weighted** accounts for class imbalance by computing the average of binary metrics in which each class's score is weighted by its presence in the true data sample.
- **micro** gives each sample-class pair an equal contribution to the overall metric (except as a result of sample-weight).
- **samples** applies only to multilabel problems.
- Selecting average=None will return an array with the score for each class.

Loss and metric comparison

	Loss	Metric
Direction	↓ Down	↑ Up
Differentiability	Yes	No
Limits	R	[0, 1]
Used	Train	Val, Test
Interpretability	No	Yes
Count	1	Many

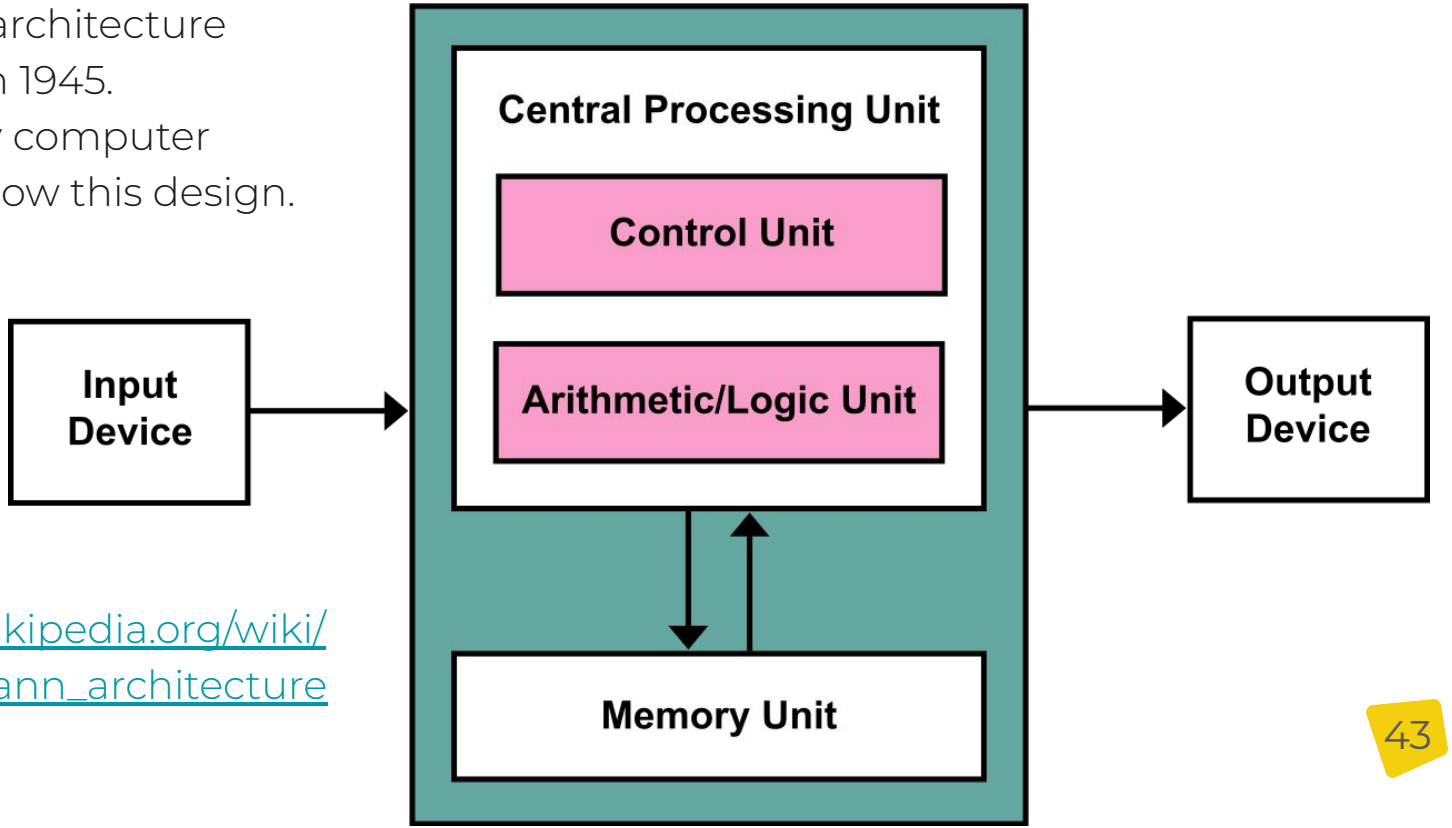
Hardware for ML

girafe ai

05

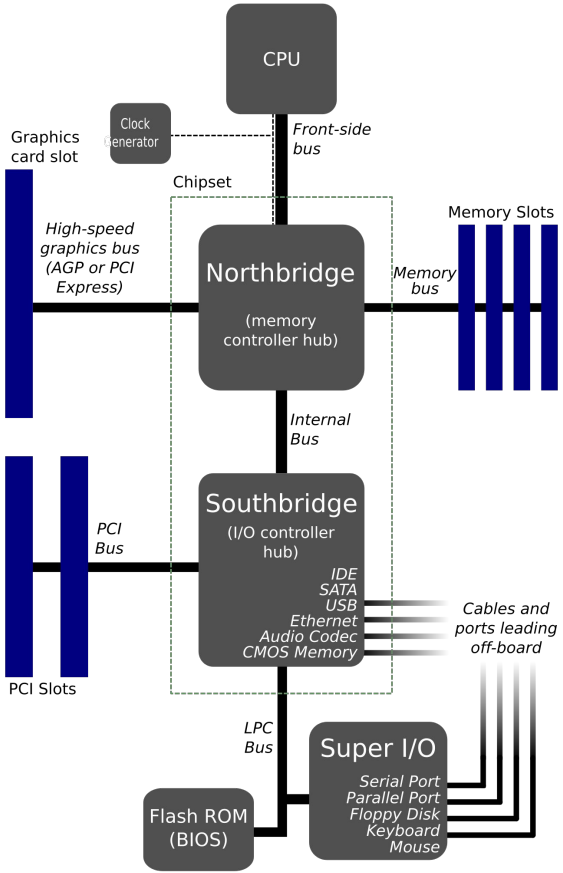
von Neumann architecture

Computer architecture proposed in 1945.
Up to today computer systems follow this design.

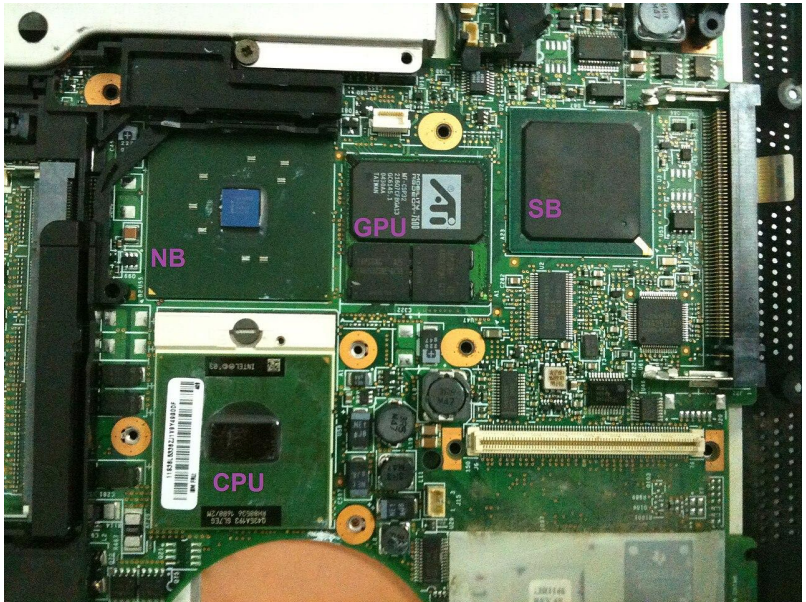


https://en.wikipedia.org/wiki/Von_Neumann_architecture

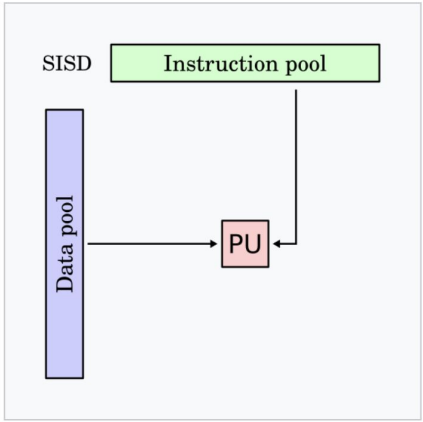
von Neumann architecture

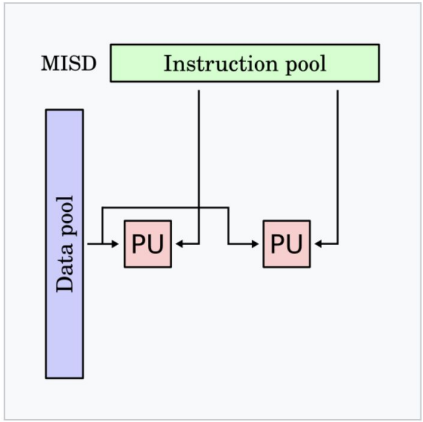


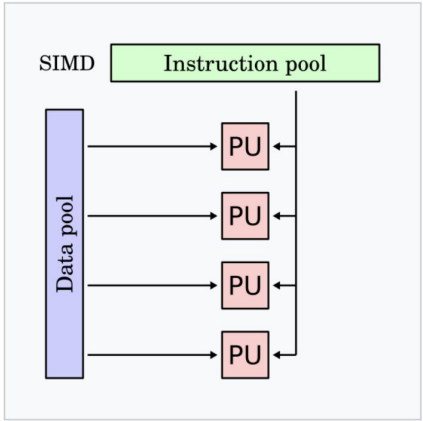
e.g. for many years (up to 2010s) this concept was presented in may motherboards by two chips:
Northbridge (CPU, GPU and memory)
Southbridge (all IO: PCI, SATA, Ethernet, USB)

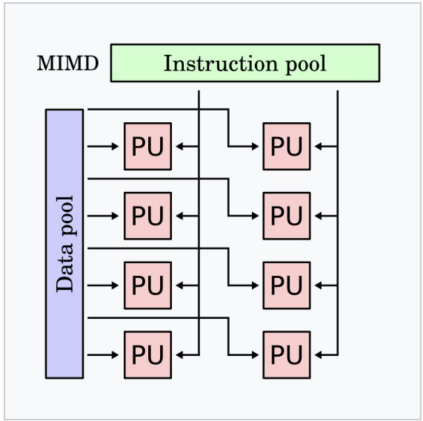


Flynn’s taxonomy









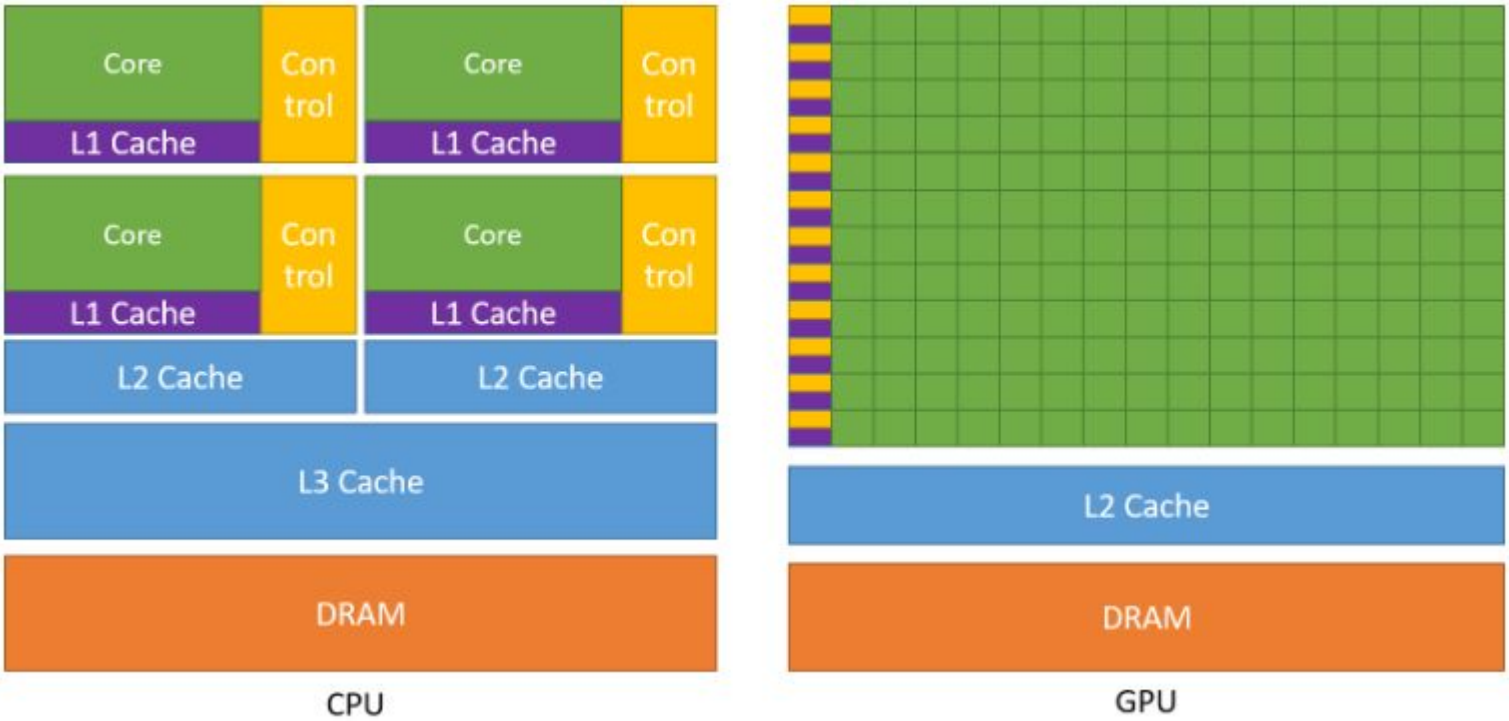
classification of multiprocessor computer architectures, proposed by Michael Flynn in 1966.

It has been used as a tool in the design of modern processors and their functionalities.

Further development of this taxonomy leads to pipelined and vector processors.

https://en.wikipedia.org/wiki/Flynn's_taxonomy

CPU vs GPU architecture



Hardware comparison

CPU

number of transistors

GPU

number of transistors

Matrix processing in Python

Numpy is concentrated on vectorized processing on CPU.
While PyTorch do it on GPU using CUDA.

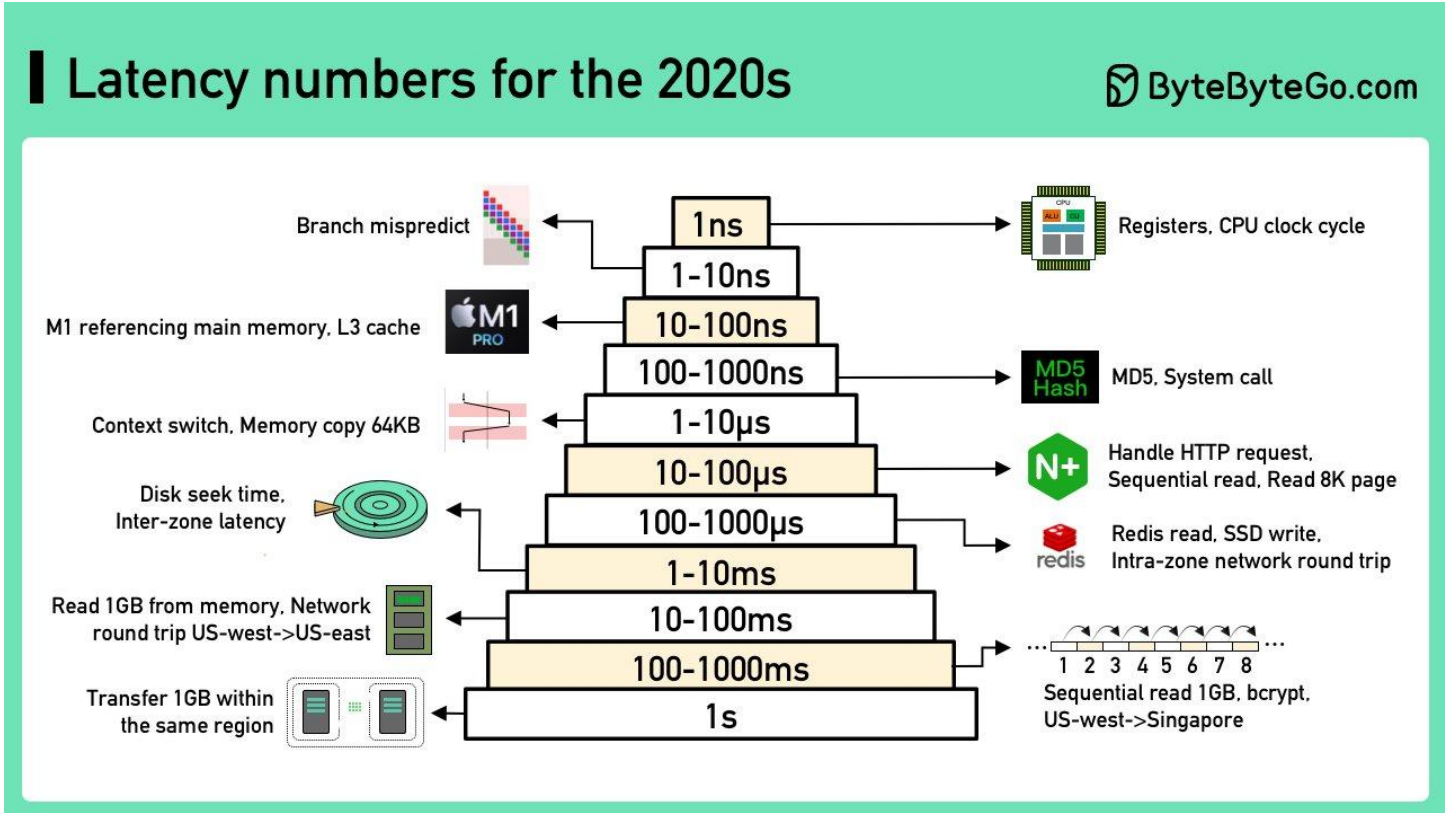
CUDA which is proprietary parallel computing platform and API that allows to use GPUs for accelerated general-purpose processing.

CUDA API is an extension of the C programming language that adds the ability to specify thread-level parallelism in C and also to specify GPU device specific operations (like moving data between the CPU and the GPU and GPU's virtual instruction set).

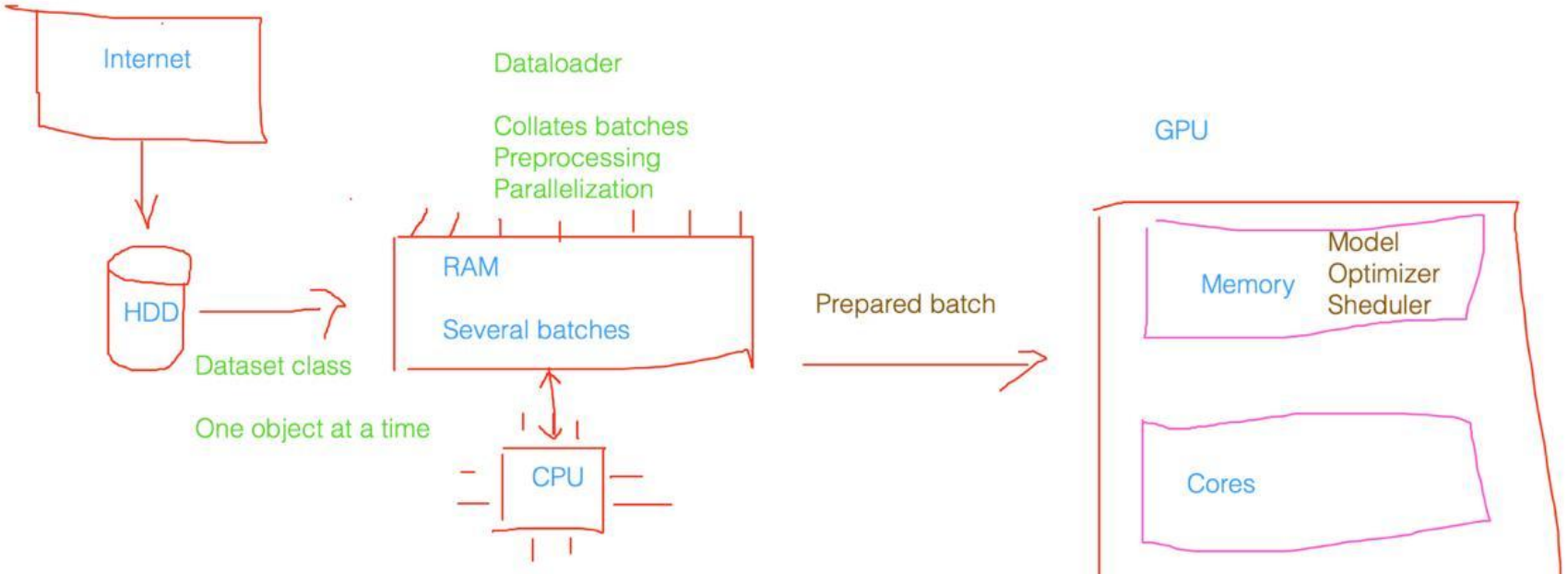
Data Locality principle

Ulrich Drepper, What every programmer should know about memory (2007)

Stackoverflow discussion



Data flow in Pytorch



1. Data streaming from network
2. Multi GPU (DDP)
3. GPU only compute (DALLE)

Revise

- Linear classification
  - margin
  - loss functions
- Logistic regression
  - sigmoid derivation
  - Maximum Likelihood Estimation
  - Logistic loss
  - probability calibration
- Multiclass aggregation strategies
  - One vs Rest
  - One vs One
- Metrics in classification
  - Accuracy, Balanced accuracy
  - Precision, Recall, F-score
  - ROC curve, PR curve, AUC
  - Confusion matrix

Next time

- Support Vector Machines
- Principal Component Analysis
- Linear Discriminant Analysis

Thanks for attention!

Questions?