

Living (or not) human
(patient or healthy) or
animal. Specimen/tissue,
biopsy. Physical phantom.

Data Acquisition. Collection.
Observation. Measurement.

**Anatomy
or Function**

Analog → Digital

**Raw Data:
Biosignal or
Bioimage**

Processing

Goal

**Better Health via
Prevention or Therapy**

**Information,
Knowledge, Answers,
Diagnosis, Therapy
Plan**

Enhanced Data. Noise
reduced. Cropping.
“Transformed” data.

**Processed
Data**

Prior Knowledge.
Patient History (Anamnesis)
Dim. Reduction.
Analysis/Interpretation.
Classification/Regression
Understanding.

Data representation.
Feature/parameter extraction.

**Features,
Patterns, or
Parameters**

Computerized Decision Support in Health Care

- After collecting biodata/digitization (Nyquist theorem), we may denoise/enhance it (temporal/Fourier filters), we extract features and reduce dimensionality.
- Now we need to use these feature for answer biomedical/clinical questions, e.g. find/predict diagnosis given features
- Then, we examine how to assess the accuracy of a prediction

Types of Decisions

Known Data → Decisions

- Diagnosis-related...
 - Probability that patient has myocardial infarction given: patient history, ECG'
 - Probability that patient has acute appendicitis, given: signs and symptoms of abdominal pain?
- Therapy-related...
 - What is the “best” therapy for patients at a certain age if an obstruction of more than 90% is seen in the left coronary artery?
 - What amount of insulin should be prescribed for a patient during the next 5 days, given the blood sugar levels and amount of insulin intake during recent weeks?
 - Measures to take if hypertension patient failed drug therapy?

Decision Models

Known Data → Decisions

- Goal:
 - Find an **output answer or class** (healthy or sick, disease A or B) given **observed features** as input
- Requires:
 - Deciding on the useful features
 - Designing and tuning (optimize) the decision model
- Features:
 - Important for computer/human decision learning
 - Without relevant features decision models are not helpful
 - Decision models may help in selecting the important features
 - Signs, symptoms, measurements, test results...
 - Feature vector (e.g 3-D vector)
[temperature weight heart-rate]

Biomedical Knowledge

Leverage medical knowledge for decision *support* models. Two types...

- Scientific or formal knowledge
 - From books, journal articles,...
 - Learning and understanding biological principles and relationships between symptoms and diseases...
 - Related to deduction and cognition
 - Learn from specific examples
- Experiential knowledge
 - Clinician has seen symptoms before and recognized underlying disease
 - Related to induction or recognition
 - Use acquired knowledge to address unseen examples

Decision-Support by Computers

- Requires:
 - Acquire and store data → database
 - Acquire and store knowledge → knowledge-base
 - Structured and formal approaches to problem solving
(different than humans approaches)
 - Ability to deal with partly available patient data
 - Ability to deal with new and unique problems
- How do humans learn, how can computers learn?

Decision-Support by Computers

- Why?
 - Repeatability
 - Different humans can give different decisions for the same problem
 - Even same human can give two different decisions at two different times
 - Reduce or eliminate variability (intra- and inter-expert variability)
 - Scalability
 - More machines and software vs. more expert/trained humans
 - Memory / Storage / Access to data and knowledge
 - Hard for humans to keep up with ever increasing medical knowledge
 - Speed
 - GPU / parallel implementation
 - Automation
 - Automating routine (mundane, tedious) tasks
 - Accuracy ?
 - People sometimes make errors (in simple and complex cases)

ARTICLES

<https://doi.org/10.1038/s42256-019-0052-1>

nature
machine intelligence

Pathologist-level interpretable whole-slide cancer diagnosis with deep learning

Zizhao Zhang¹, Pingjun Chen^②, Mason McGough², Fuyong Xing³, Chunbao Wang⁴, Marilyn Bui⁵, Yuanpu Xie², Manish Sapkota⁶, Lei Cui², Jasreman Dhillon⁵, Nazeel Ahmad⁷, Farah K. Khalil⁵, Shohreh I. Dickinson⁵, Xiaoshuang Shi², Fujun Liu⁶, Hai Su², Jinzheng Cai² and Lin Yang^{2*}

COVER STORY

AI applications in ophthalmology achieve human expert-level performance

Ocular Surgery News U.S. Edition, June 10, 2018

Stanford ML Group

CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning

Pranav Rajpurkar*, Jeremy Irvin*, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, Matthew P. Lungren, Andrew Y. Ng

 nature
International journal of science

Letter | Published: 25 January 2017

Jan. 2017

Dermatologist-level classification of skin cancer with deep neural networks

Andre Esteva ✉, Brett Kuprel ✉, Roberto A. Novoa ✉, Justin Ko, Susan M. Swetter, Helen M. Blau & Sebastian Thrun ✉

Nature 542, 115–118 (02 February 2017) | Download Citation ↴

<https://www.nature.com/articles/nature21056>



“CNN ROC AUC greater than mean ROC area of dermatologists... higher specificity”

Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists

H. A. Haenssle^{a,*}, C. Fink^{b†}, R. Schneiderbauer^b, F. Toberer^b, T. Buhl^b, A. Blum^b, A. Kalloo^b, A. Ben Hadj Hassen^b, L. Thomas^b, A. Enk^c & L. Uhlmann^b

August 2018

<https://academic.oup.com/annonc/article/29/8/1836/5004443>

GENERAL DERMATOLOGY

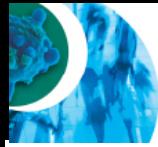
Feb. 2019

BJD
British Journal of Dermatology

Deep-learning-based, computer-aided classifier developed with a small dataset of clinical images **surpasses board-certified dermatologists in skin tumour diagnosis***

Y. Fujisawa , Y. Otomo, Y. Ogata, Y. Nakamura, R. Fujita, Y. Ishitsuka, R. Watanabe, N. Okiyama , K. Ohara⁴ and M. Fujimoto³

<https://onlinelibrary.wiley.com/doi/abs/10.1111/bjd.17470>

 EJC
EUROPEAN JOURNAL OF CANCER

May 2019

Deep learning **outperformed** 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task

Titus J. Brinker ^{a,b,*}, Achim Hekler ^a, Alexander H. Enk ^b, Joachim Klode ^c, Axel Hauschild ^d, Carola Berking ^e, Bastian Schilling ^f, Sebastian Haferkamp ^g, Dirk Schadendorf ^c, Tim Holland-Letz ^h, Jochen S. Utikal ^{i,j,l}, Christof von Kalle ^{a,l}, Collaborators²

<https://www.sciencedirect.com/science/article/pii/S0959804919302217>

Training Stage

1. Collect training data, and extract features:

(feature vector #1, class #1) e.g. ([1 3 2 1], "sick")

(feature vector #2, class #2) ([2 2 3 2], "healthy")

(feature vector #3, class #2) ([3 1 2 1], "sick")

6

(feature vector #N, class #N) ([4 2 -1 5], "healthy")

class#i $\in \{C_1, C_2, C_3, \dots, C_{\text{number of classes}}\}$

2. Design Classifier that...

- a) works well on training data (above)
 - b) generalizes well to new data (below)

Testing Stage: Given new data, calculate feature vector, Predict its class $\in \{C_1, C_2, C_3, \dots, C_{\text{number of classes}}\}$

Prediction

- Classification

	Binary	Multi-Class
Crisp	healthy vs. sick therapy A vs. B	disease A vs. B vs. C vs. D cancer type A vs. B vs C
Fuzzy Probabilistic	Prob(healthy)=? Prob(sick)=? $P_S + P_H = 1$	Prob(disease A)=... Prob(disease B)=... Prob(disease C)=... $P_A + P_B + P_C + \dots = 1$

One Feature and Two Classes

- One feature: Systolic blood pressure
- Two classes: Hypertensive and Non-hypertensive
- Collect training data:

Subject i : feature_i , class_i ∈ { Hypertensive (+ve) , Non-hypertensive (-ve) }

Subject 1: (150 , +ve)

Subject 2: (140, +ve)

Subject 3: (120, -ve)

Subject 4: (118, -ve)

Subject 5: (145, +ve)

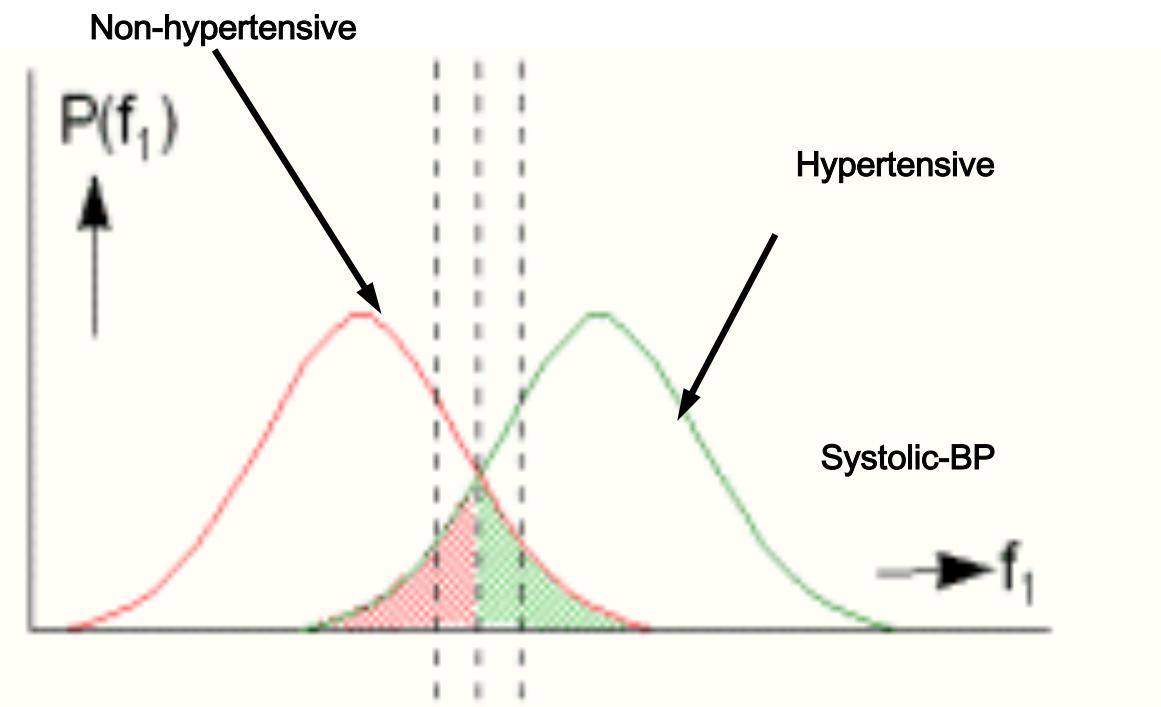
....

Subject 100: (122, -ve)

- Number of Observations (number of subjects): 100
- Number of variables (or features): 1
- Do we need dimensionality reduction (PCA?)? No, we have only one feature.

One Feature and Two Classes

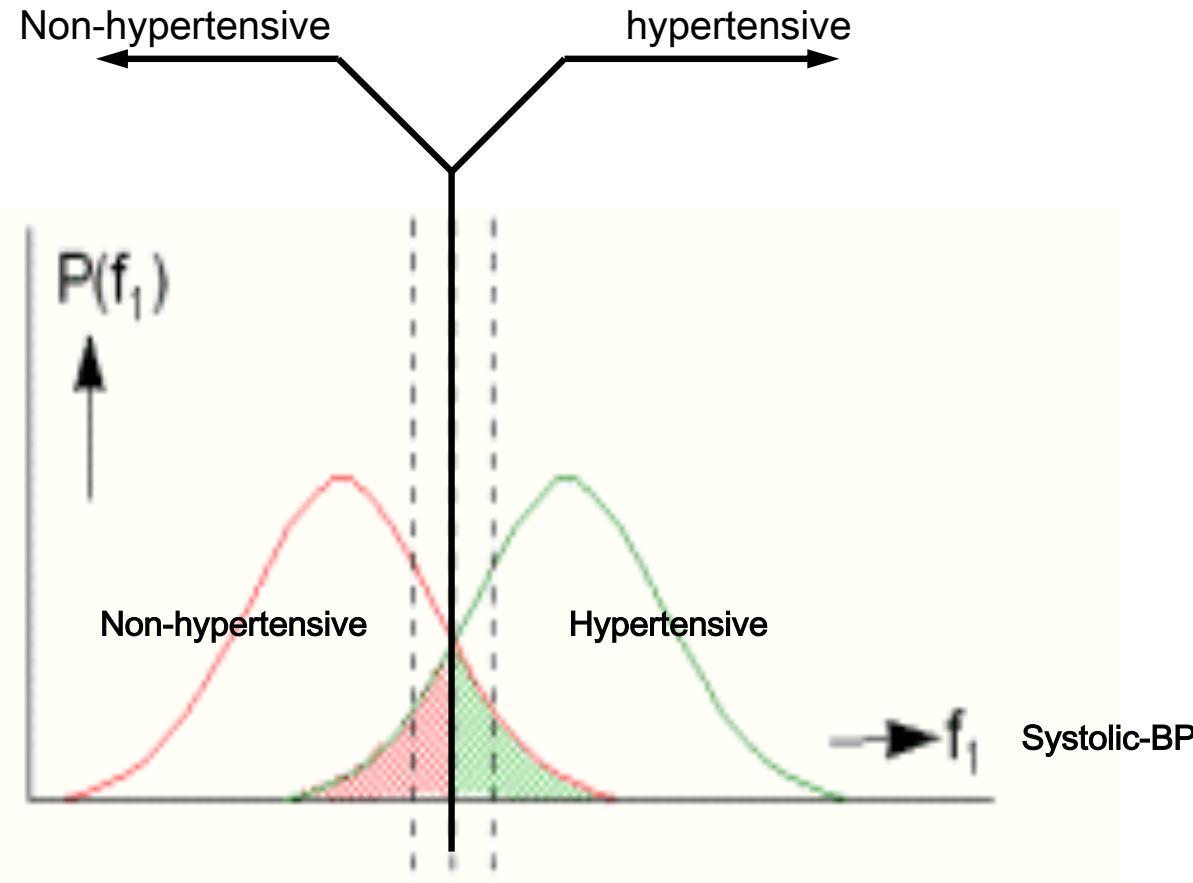
- Plot 1D Histogram of Hypertensive. What's the mean? Std. dev.?
- Plot 1D Histogram of Non-Hypertensive. Mean? Std. dev.?
- Model using probability density function (PDF), e.g. Gaussian with known means and standard deviations.
- Plots of Probability(BP) vs. BP



Single Decision Threshold

Decision Model:

- $BP \geq T \rightarrow \text{hypertensive}$
- $BP < T \rightarrow \text{non-hypertensive}$



[Mis]classification

- T: True (correct classification)
- F: False (incorrect classification)
- P: Positive (having the disease)
- N: Negative (not having the disease, healthy)
- TP: Hypertensive people classified as such
- TN: Healthy people classified as such
- FP: Healthy people classified as hypertensive
- FN: Hypertensive classified as healthy
- Choice of T affects these numbers...
- FP, FN can not be minimized independently
- Expressed as %, fractions, or numbers of persons, or fraction

- **Sensitivity**

Fraction of correctly classified disease
= $TP / \text{"all the P's"} = TP / (TP+FN)$

- **Specificity**

Fraction of correctly classified healthy
= $TN / \text{"all the N's"} = TN / (TN+FP)$

	Decision Model		100%	
	+	-		
Truth	+	TP 60%	FN 40%	100%
	-	FP 30%	TN 70%	

		Patients with bowel cancer (as confirmed on endoscopy)		
		Condition positive	Condition negative	
Fecal occult blood screen test outcome	Test outcome positive	True positive (TP) = 20	False positive (FP) = 180	Positive predictive value $= TP / (TP + FP)$ $= 20 / (20 + 180)$ $= 10\%$
	Test outcome negative	False negative (FN) = 10	True negative (TN) = 1820	Negative predictive value $= TN / (FN + TN)$ $= 1820 / (10 + 1820)$ $\approx 99.5\%$
		Sensitivity $= TP / (TP + FN)$ $= 20 / (20 + 10)$ $\approx 67\%$	Specificity $= TN / (FP + TN)$ $= 1820 / (180 + 1820)$ $= 91\%$	

		Condition (as determined by "Gold standard")		Prevalence $= \frac{\sum \text{Condition positive}}{\sum \text{Total population}}$	
Total population		Condition positive	Condition negative		
Test outcome	Test outcome positive	True positive	False positive (Type I error)	Positive predictive value (PPV) Precision $= \frac{\sum \text{True positive}}{\sum \text{Test outcome positive}}$	False discovery rate (FDR) $= \frac{\sum \text{False positive}}{\sum \text{Test outcome positive}}$
	Test outcome negative	False negative (Type II error)	True negative	False omission rate (FOR) $= \frac{\sum \text{False negative}}{\sum \text{Test outcome negative}}$	Negative predictive value (NPV) $= \frac{\sum \text{True negative}}{\sum \text{Test outcome negative}}$
	Accuracy (ACC) = $\frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{Total population}}$	True positive rate (TPR), Sensitivity, Recall $= \frac{\sum \text{True positive}}{\sum \text{Condition positive}}$	False positive rate (FPR), Fall-out $= \frac{\sum \text{False positive}}{\sum \text{Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR-}}$
	False negative rate (FNR), Miss rate = $\frac{\sum \text{False negative}}{\sum \text{Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\sum \text{True negative}}{\sum \text{Condition negative}}$		Negative likelihood ratio (LR-) $= \frac{\text{FNR}}{\text{TNR}}$	

[Mis-] classification

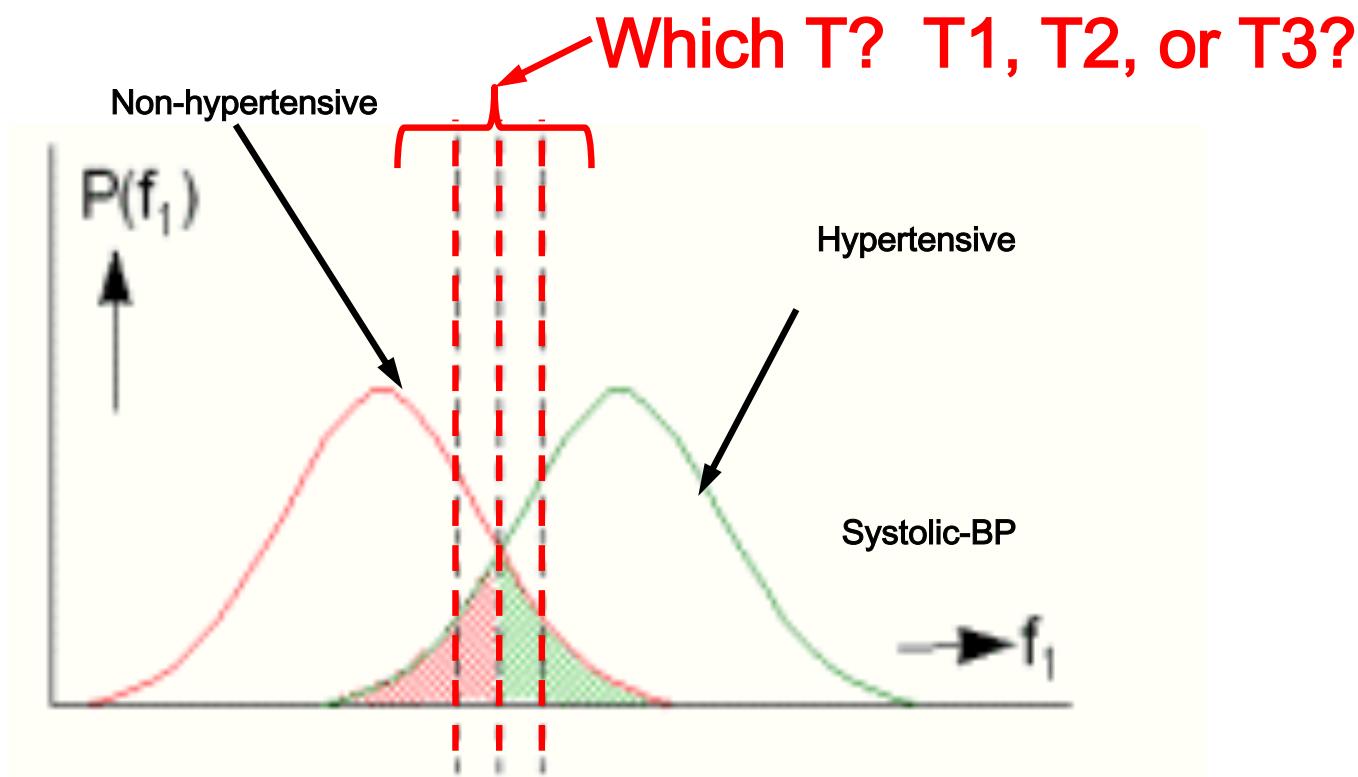
- | | | | |
|----|--|----|---|
| a. | Total number of people | a. | $300+200+1000+4000=5500$ |
| b. | Total number of diseased | b. | 500 |
| c. | Total number of healthy | c. | 5000 |
| d. | Total number of healthy classified as diseased (FP) | d. | 1000 |
| e. | Total number correctly classified healthy (TN) | e. | 4000 |
| f. | Total number of diseased classified as healthy (FN) | f. | 200 |
| g. | Total number of correctly classified diseased (TP) | g. | 300 |
| h. | Fraction of healthy people (Prior Probability) | h. | $(1000+4000)/5500=5000/5500=\sim0.91$ |
| i. | Fraction of diseased people (Prior Probability) | i. | $(300+200)/5500=500/5500=\sim0.091$ |
| j. | Fraction of correctly classified diseased (Sensitivity) | j. | $300/500=0.6 = \text{TP} / (\text{TP}+\text{FN})$ |
| k. | Fraction of correctly classified healthy (Specificity) | k. | $4000/5000=0.8 = \text{TN} / (\text{TN}+\text{FP})$ |

		Model	
		+	-
truth	+	300	200
	-	1000	4000

Number of persons

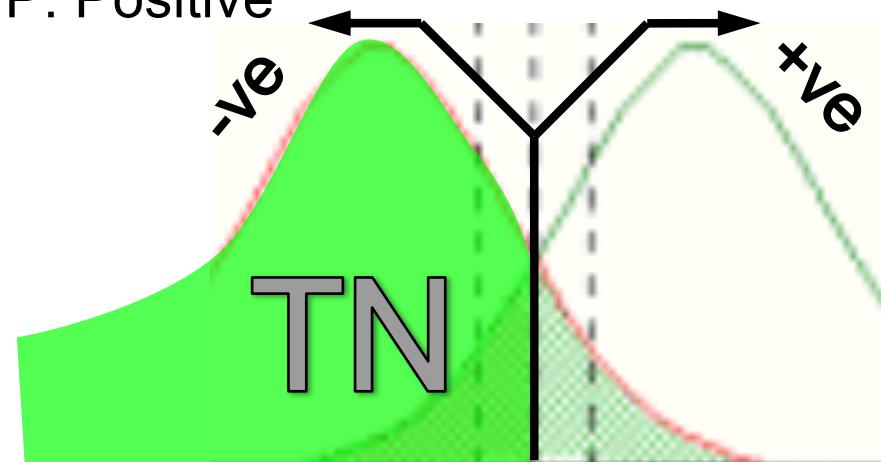
Single Decision Threshold

- What value of BP can act as threshold (T)
- Typically, perfect T does not exist....
- Overlap \rightarrow Classification (decision) error
- Some BP values appear in both classes

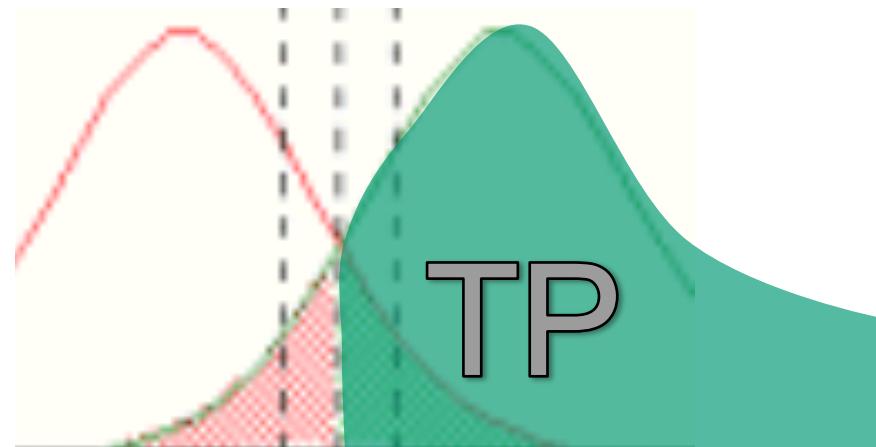


T: True
F: False
N: Negative
P: Positive

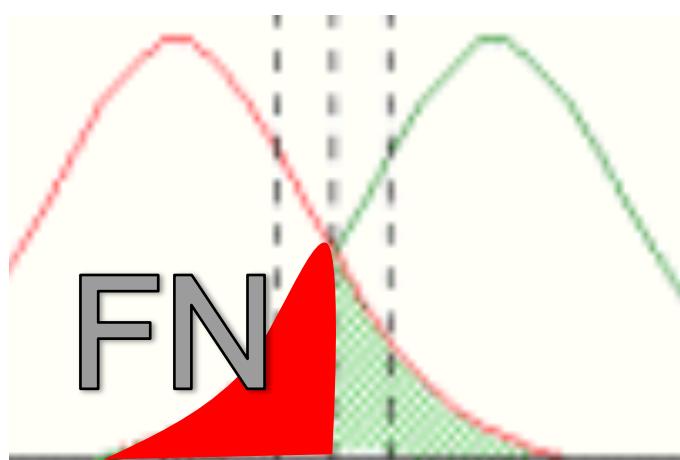
[Mis]classification



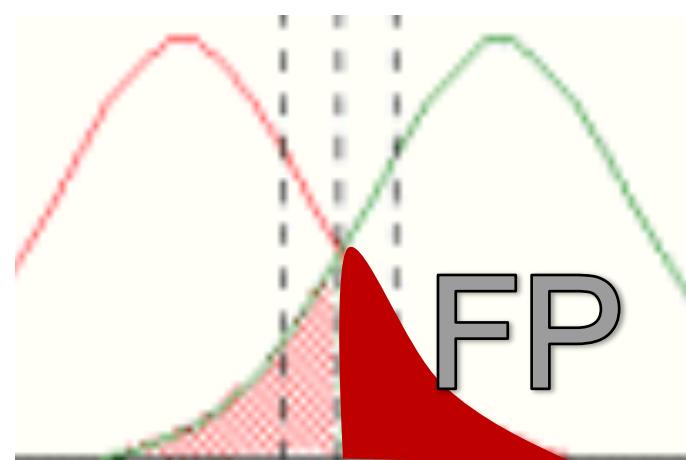
Area = prob. of correctly classified as non-hypertensive



Area = prob. correctly classified as hypertensive

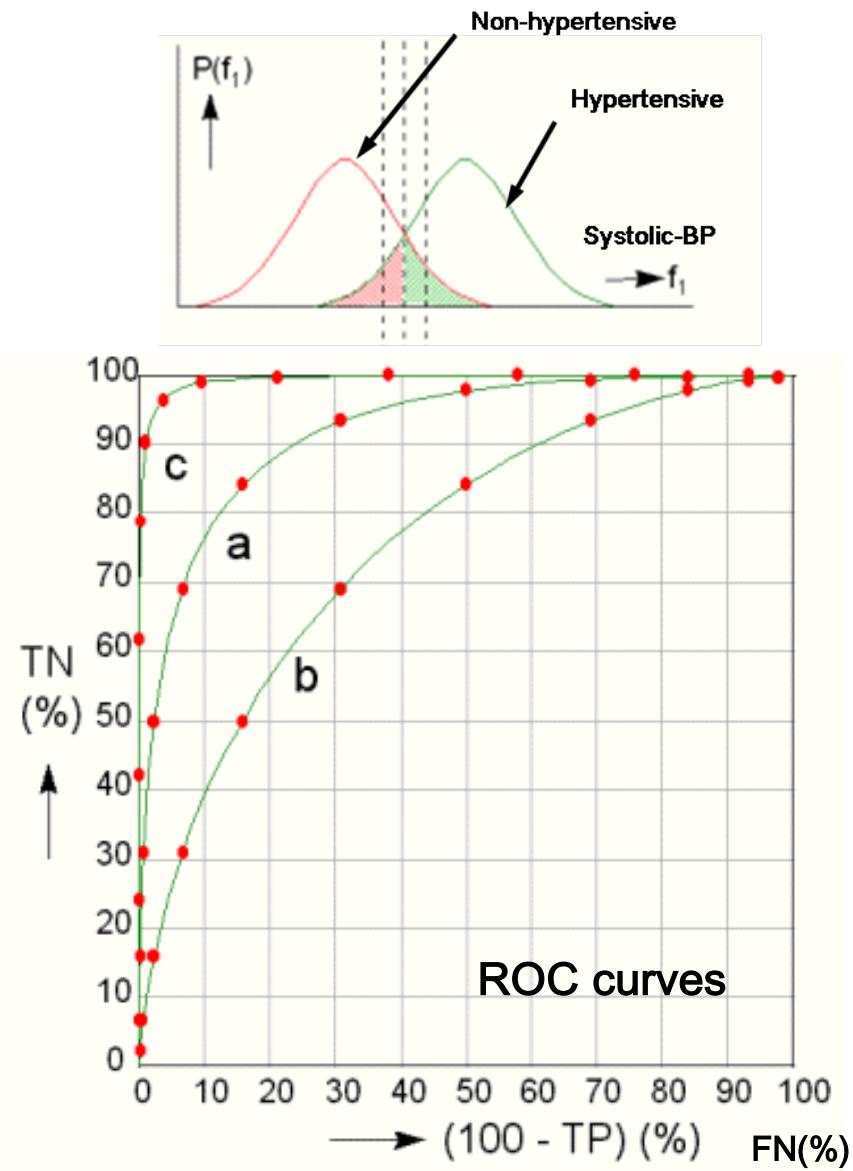
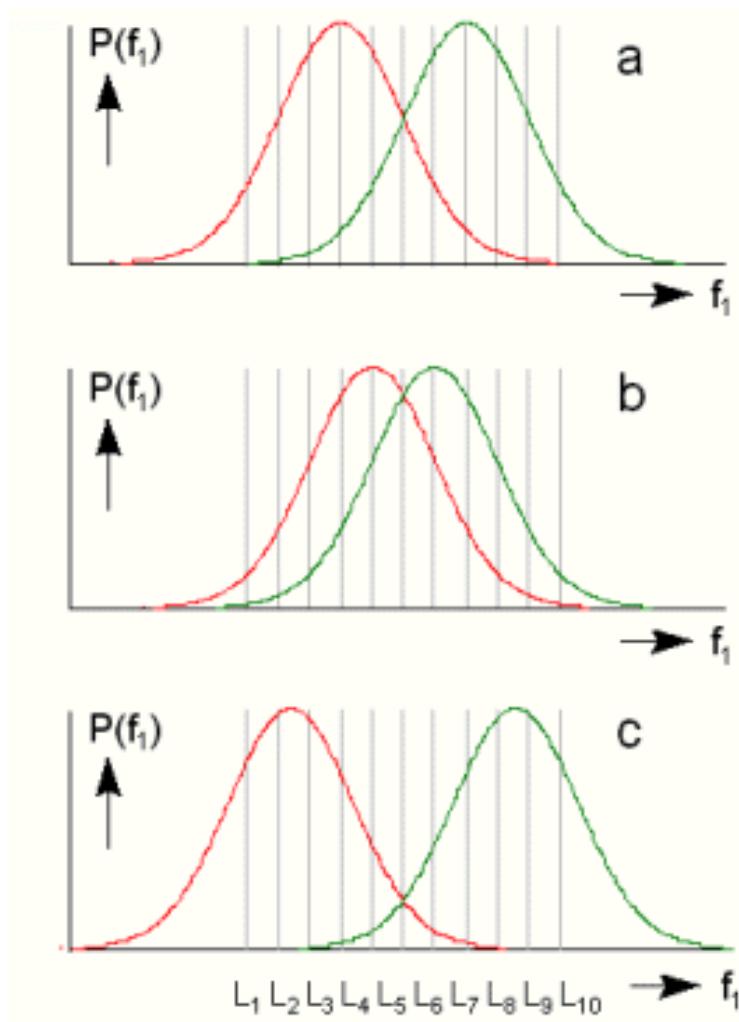


Area = prob. incorrectly classified as non-hypertensive

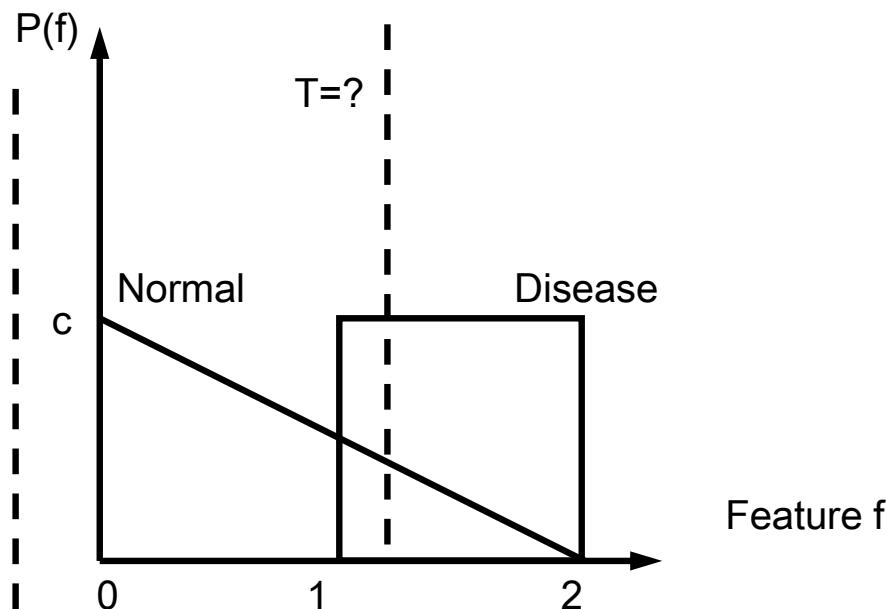


Area = prob. incorrectly classified as hypertensive

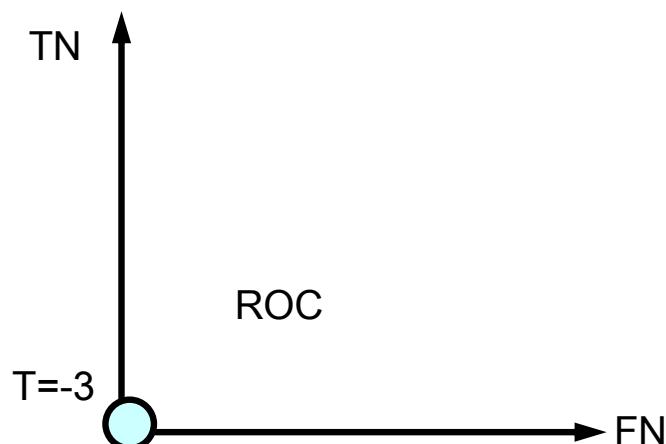
Receiver Operating Characteristics (ROC)



Receiver Operating Characteristics (ROC)

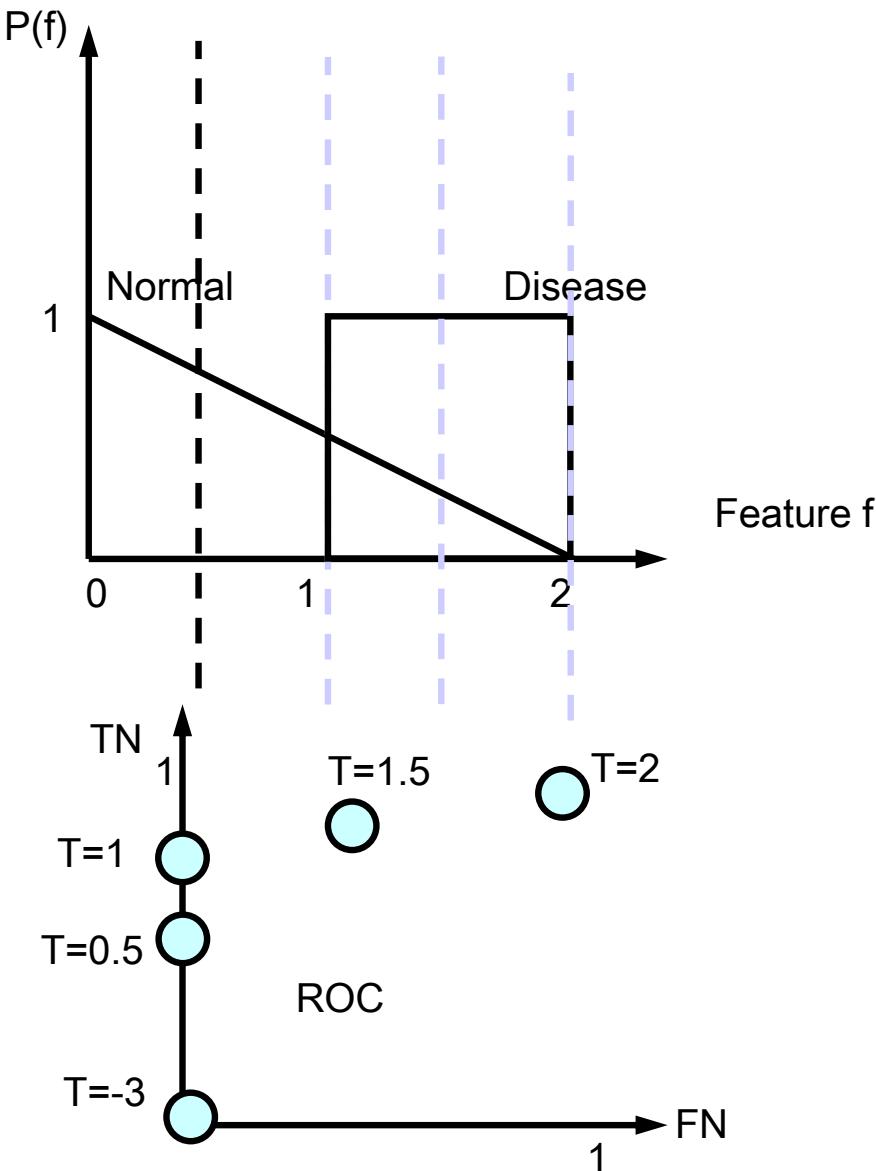


- What is the value c ?
 $c=1$
- Decision parameter? Rule?
 $T, f > T \rightarrow \text{Disease}, f < T \rightarrow \text{Normal}$
- $T = -3$, what is the probability of:
 - a. correctly classifying normal?
 - b. correctly classifying disease?
 - c. classifying normal as disease?
 - d. classifying disease as normal?



- $P(a)=0, P(b)=1, P(c)=1, P(d)=0$
- Which from a,b,c,d refers to FP,TP,FN,TN?
 $c \rightarrow \text{FP}, b \rightarrow \text{TP}, d \rightarrow \text{FN}, a \rightarrow \text{TN}$

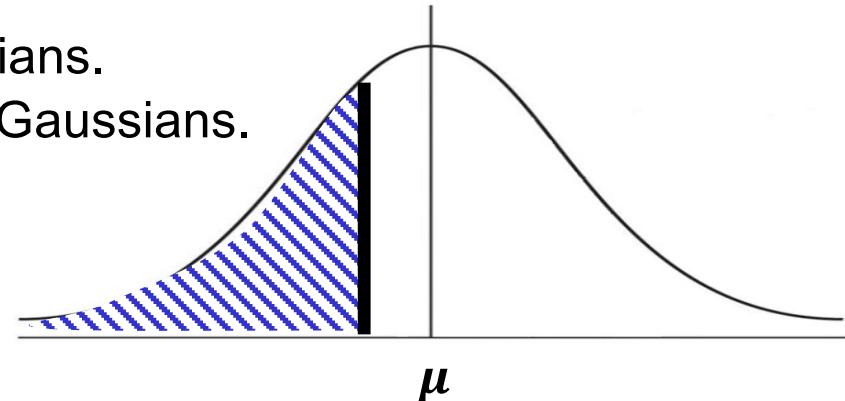
Receiver Operating Characteristics (ROC)



- $T=0.5$, probability of TN, TP, FP, FN?
 $P(TN)=(3/4+1)/2*1/2=0.4375$
 $P(TP)=1$
 $P(FP)=1/2*1.5*3/4=0.5625$
 $P(FN)=0$
- $T=1$:
 $P(TN)=(1/2+1)/2*1=0.75$
 $P(TP)=1$
 $P(FP)=1/2*1*1/2 =0.25$
 $P(FN)=0$
- $T=1.5$
 $P(TN)=(1/4+1)/2*1.5=0.9375$
 $P(TP)=1/2*1=0.5$
 $P(FP)=1/2*1/2*1/4 =0.0625$
 $P(FN)=1/2*1=0.5$
- $T=2$
 $P(TN)=1$
 $P(TP)=0$
 $P(FP)=0$
 $P(FN)=1$

Common to represent distributions with Gaussians.

To find ROC, we need to perform integrals on Gaussians.



%normcdf(x) integrated Gaussian from -inf to x

```
normcdf(3)      %when we don't provide mu, std.dev, they are by default 0 and 1.  
normcdf(3,2,1) %here we provide mean 2 and std.dev. 1  
normcdf(inf)   %returns 1, since we are integrating from -inf to 0, i.e. left-half of Gaussian  
normcdf(0)      %returns 0.5, since we are integrating left-half of Gaussian  
normcdf(inf,10,4) %mean, std.dev. we still get 1  
normcdf(0,0,4)   %if the mean is zero, and we integrate from -inf to 0, we still get 0.5  
normcdf(0,1,0.5) %if the mean is >0, and we integrate from -inf to 0, we get <0.5  
normcdf(0,-1,0.5) %if the mean is <0, and we integrate from -inf to 0, we get >0.5  
normcdf(3.22,3.22,0.5) %if we integrate from -inf to mean, we always get 0.5  
normcdf(3,4,0.5)   %if we integrate from -inf to x<mean, we get <0.5  
normcdf(5,4,0.5)   %if we integrate from -inf to x>mean, we get >0.5  
normcdf(5,4,0.5)-0.5 % if we want to integrate from mean to x (x>mean) , we integrate from -inf to x  
and subtract from 0.5  
0.5-normcdf(4,5,0.5) % if we want to integrate from x to mean (mean>x) , we integrate from -inf to x  
and subtract it from 0.5  
normcdf(3,2,1)-normcdf(2.5,2,1) % if we want to integrate between a and b (b>a)
```

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/(2\sigma^2)}$$

$$t^2 = \frac{x^2}{2\sigma^2} \quad \text{and} \quad dt = \frac{1}{\sqrt{2\sigma}} dx$$

$$\int_{-x}^x G(x) dx = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt = \operatorname{erf} x = \operatorname{erf} \left(\frac{x}{\sqrt{2\sigma}} \right)$$

$$\operatorname{erfc} x = 1 - \operatorname{erf} x = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$$

`erf(x)` returns the **Error Function** evaluated for each element of `x`.

`erfc(x)` returns the **Complementary Error Function**

`erfinv(x)` returns the **Inverse Error Function**

`erfcinv(x)` returns the value of the **Inverse Complementary Error Function**

http://www.mhtlab.uwaterloo.ca/courses/me755/web_chap2.pdf