digit recognition

Y = Output

$= \{0, 1, 2, \ldots, 9\}$

= label of the number

X = Input space

= Vectorized discretized Image of the digit

$r \longrightarrow \begin{bmatrix} \vdots \\ \vdots \\ \vdots \end{bmatrix}_{rc} = X = (X_1, X_2, \ldots, X_{rc})$

c

$f : (X_1, X_2, \ldots, X_{rc}) \longmapsto \{0, 1, 2, \ldots, 9\}$

$P\{Y = j \mid X\}$

(I) Shape of the data (n, p). $\frac{n}{p} < 1 \Rightarrow$ Ultra

(II) Content of data (Type) $\frac{n}{p} > 1$ Thin

Consider the i^{th} obs. $X_i = (X_{i1}, X_{i2}, \ldots, X_{ip})^T$ p-dim input

What is X_{ij}?

(i) $X_{ij} \in \{0, 1\}$ Binary input

$X_i \in \{0, 1\}^p = \mathcal{X}$

(ii) $X_{ij} \in [0, 1]$ Unitary Input

Neural Networks

$(X_i, Y_i) \overset{iid}{\sim} P_{XY}(x, y)$

$f : \mathcal{X} \longrightarrow \mathcal{Y}$

$x \longmapsto f(x)$.

$\mathcal{X}$: Input space.

$\mathcal{Y}$: Output space

Def. f is a function ref to as a learning machine

f encodes the pattern of relationship (association) b/w X and Y.

$f \in \mathcal{H}$

eg: $\mathcal{Y}^{\mathcal{X}}$ = All possible mappings from $\mathcal{X}$ to $\mathcal{Y}$

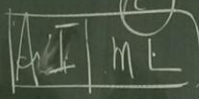
$\mathcal{H} \overset{def}{=}$ function space. Hypothesis space.

Note: (a) $\mathcal{H}$ maybe finite and countable

$\mathcal{H} = \{f_1, f_2, \ldots, f_m\}$

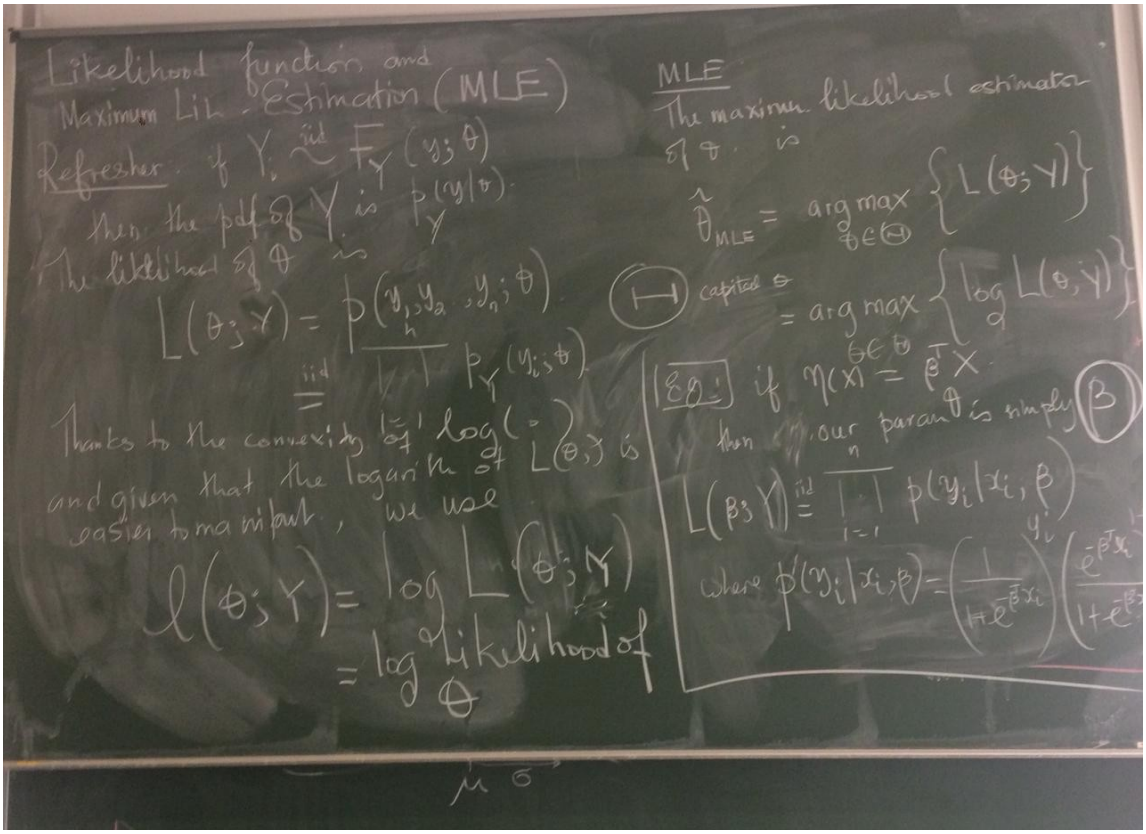
(b) $\mathcal{H} \equiv$ countably infinite

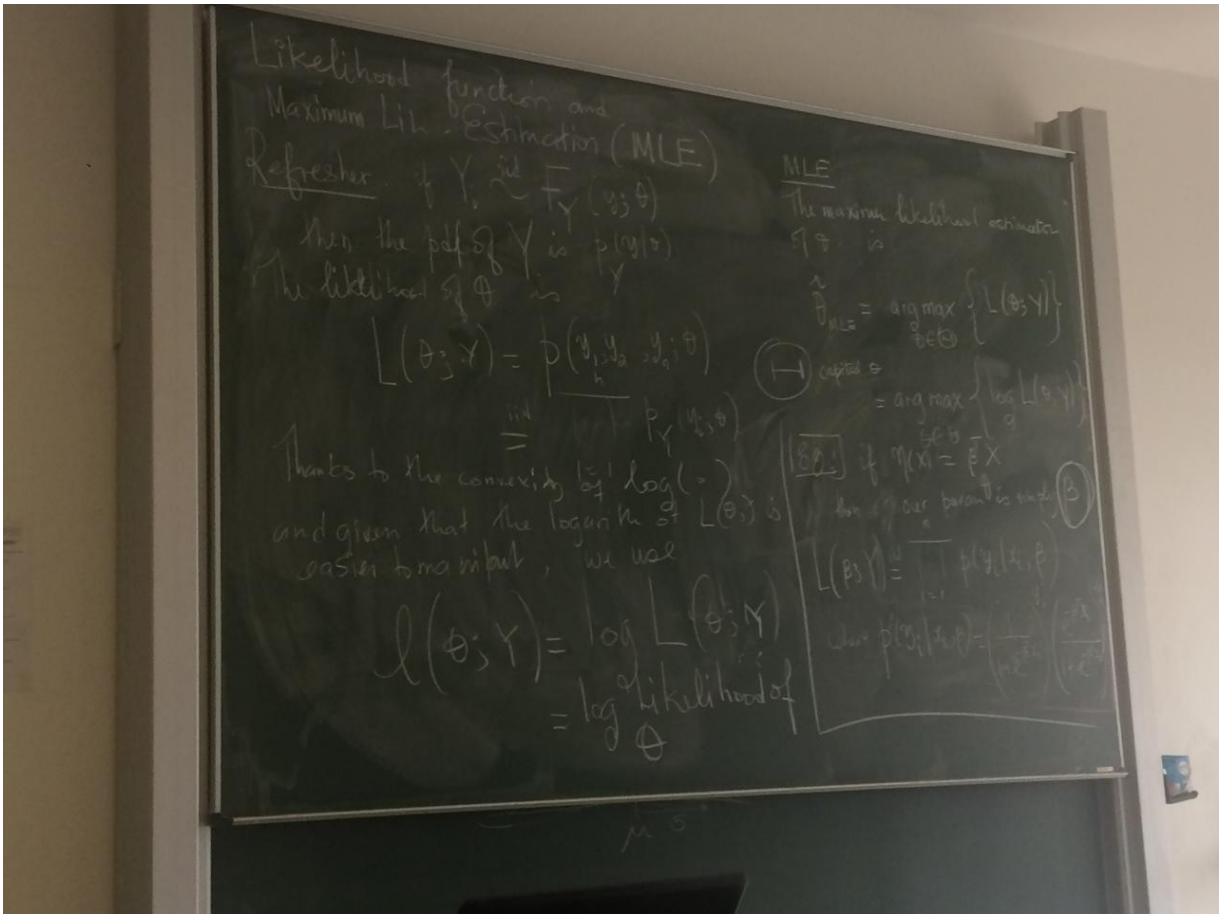
(c) $\mathcal{H}$ = Infinite



Choosing $\mathcal{H}$ is an act of approximation

eg $\mathcal{H}$ = Polynomials of degree m


Likelihood function and
Maximum Lik. Estimation (MLE)
Refresher
MLE
The maximum likelihood estimator
Thanks to the convexity of log(·)
and given that the logarithm of L(θ;Y) is
easier to manipulate, we use
log likelihood of
capital Θ
if
then our parameter is simply
where


Likelihood function and
Maximum Lik. Estimation (MLE)
Refresher
MLE
The maximum likelihood estimator
Thanks to the convexity of log(·)
and given that the logarithm of L(θ;Y) is
easier to manipulate, we use
log likelihood of
capital Θ
if
then our parameter is simply
where

Focus on $p_{Y|x}(y|x)$ posterior density of Y given $X=x$

Central to AI and ML b/c we are typically given $X=x$ and we seek to predict Y given $X=x$

AI = Artificial Intelligence

ML = Machine Learning

Eg: Digit Reg

- Take a picture of a given digit
- Extract the grayscale $r \times c$ matrix
- Vectorize $[\,] \rightarrow [\,]$

$$\begin{cases} \mathbb{P}(Y=0|x) \\ \mathbb{P}(Y=1|x) \\ \vdots \\ \mathbb{P}(Y=9|x) \end{cases}$$

$\hat{f}(x)$ = label of x

Majority rule

$$= \begin{cases} 0 & \text{if } \mathbb{P}(Y=0|x) \text{ is largest} \\ 1 & \text{if } \mathbb{P}(Y=1|x) \text{ is largest} \\ \vdots \\ 9 & \text{if } \mathbb{P}(Y=9|x) \text{ is largest} \end{cases}$$

$$\hat{f}(x) = \underset{j=0,1,\dots,9}{\arg\max}\left(\mathbb{P}(Y=j|x)\right)$$

$$\tilde{x}_{ij} \rightarrow \frac{x_{ij} - x^j_{(1)}}{x^j_{(n)} - x_{(1)}}$$

$$x_{ij} = \tilde{x}_{ij}\left(x^j_{(n)} - x^j_{(1)}\right) + x^{(j)}_{(1)}$$

(III) Distribution

$$p_{XY}(x, y)$$

(IV) Missing Ness

Pattern and incidence of missing values

(V) Outlier and Novelty Detection

Understanding $p_{XY}(x,y)$

- Focus on $p_X(x)$ ref to density estimation in $\mathcal{X}$ key to Cluster analysis
- Focus on $p_{X|Y}(x|y)$ ref estimation of the so called class conditional density. Pillar of generative pattern recog via discriminant Analysis
- Focus of $p_Y(y)$ is usually the easiest in PR thus is simply discrete prob estim

Eg:

Binary Classification

$Y = \{0, 1\}$

$X \subset \mathbb{R}^p$ $\quad x = (x_1, \ldots, x_p)^T = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix}$

$x_j \in \mathbb{R}$

Hypothesized:

Logistic Regr

Assume $P\{Y=1|x\} = \frac{1}{1+e^{-\eta(x)}}$

where $\eta : X \to \mathbb{R}$ ✓

$\eta(x) = \beta^T x$

$\eta(x)$ = nonlinear

$P(Y=0|x) = 1 - P(Y=1|x)$

$= 1 - \pi(x)$

$Y=0$ or $Y=1$

$P(A^c) = 1 - P(A)$

$p(y|x) = \pi(x)^y (1-\pi(x))^{1-y}$

$= \left(\frac{1}{1+e^{-\eta(x)}}\right)^y \left(\frac{e^{-\eta(x)}}{1+e^{-\eta(x)}}\right)^{1-y}$

Goal: Estimate $\eta(x)$

(ii) $X_{ij} \in \{a_1, a_2, \ldots, a_m\}$

(iv) $X_{ij} \in [a, b] \subset \mathbb{R}$

$X_i \in X \subset \mathbb{R}^p$

(v) $X_{ij} \equiv$ Frequency of word j in document i

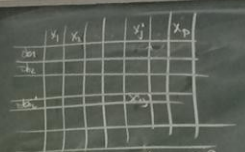
$\in \mathbb{R}^{p \times 1}$

$\equiv$ Text Mining with Bag of Words assumptions.

(vi) $X_{ij} \in \{$strongly disagree, disagree, Neutral, agree, strongly agree$\}$ 5 values

$\{0, 1\}$ 2 values

$\Rightarrow$ Ordinal Input



$x_{ij} \Rightarrow \frac{x_{ij} - \min}{\text{Range}(x)}$

ultra Thin

$\frac{n}{f}$

saturated desp

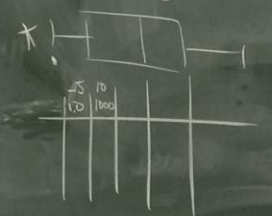
Cubitize

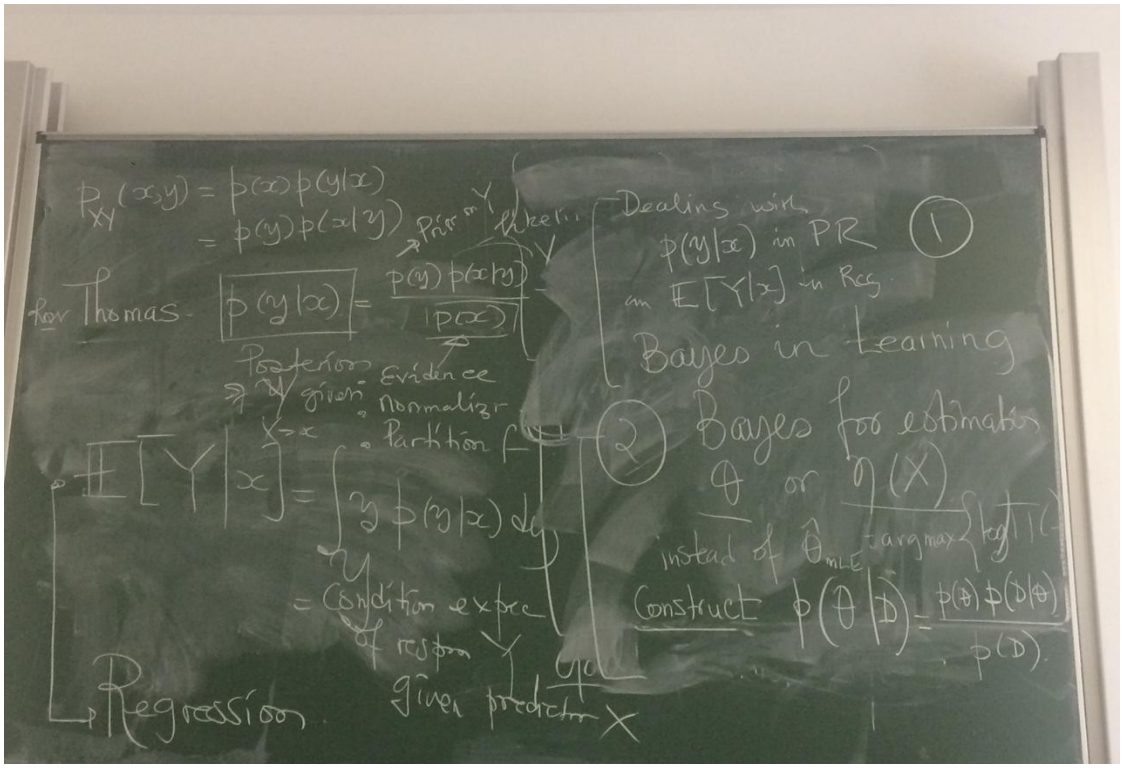
$X_i \in [a, b]^p$

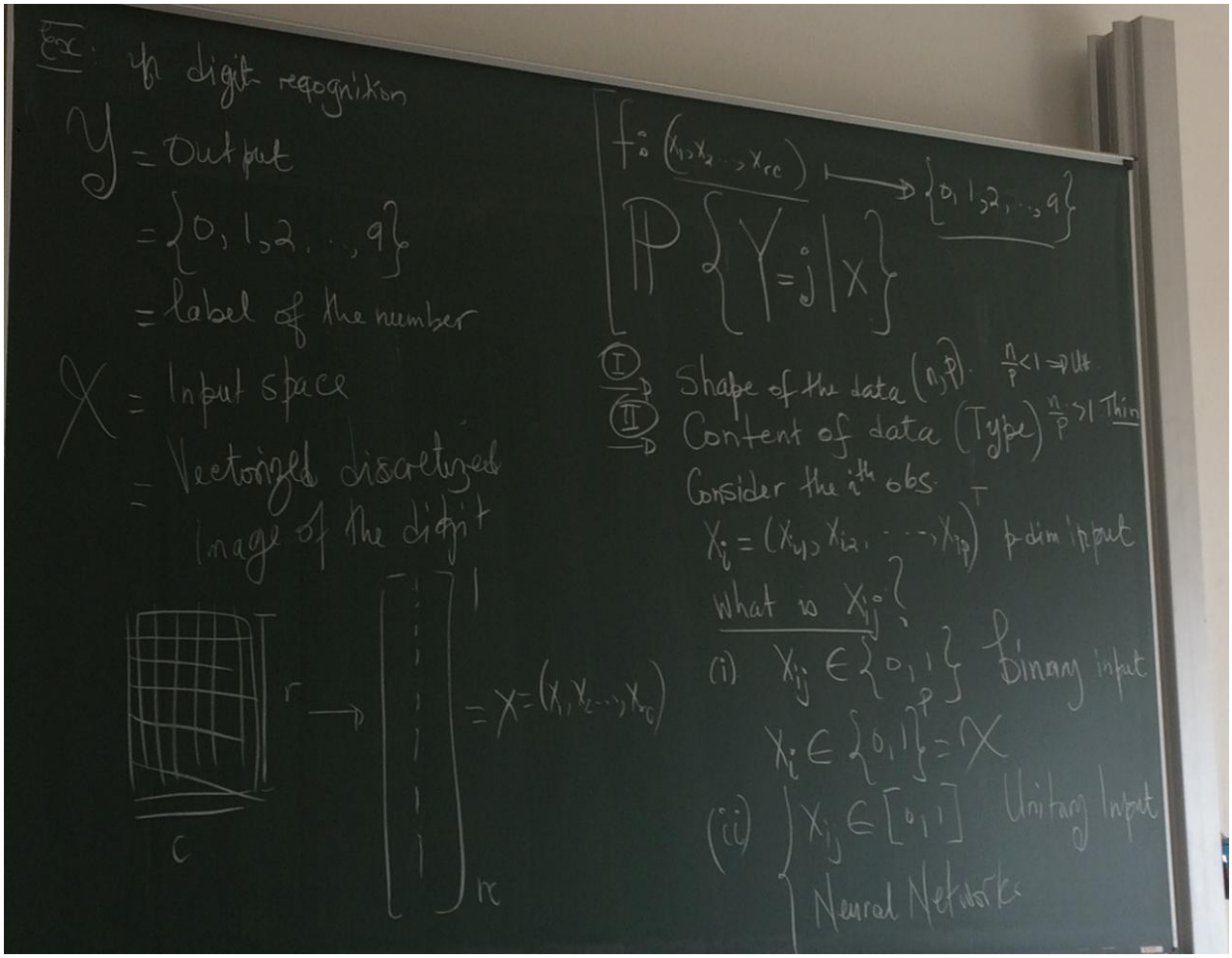
$X_i \in [0, 1]^p$

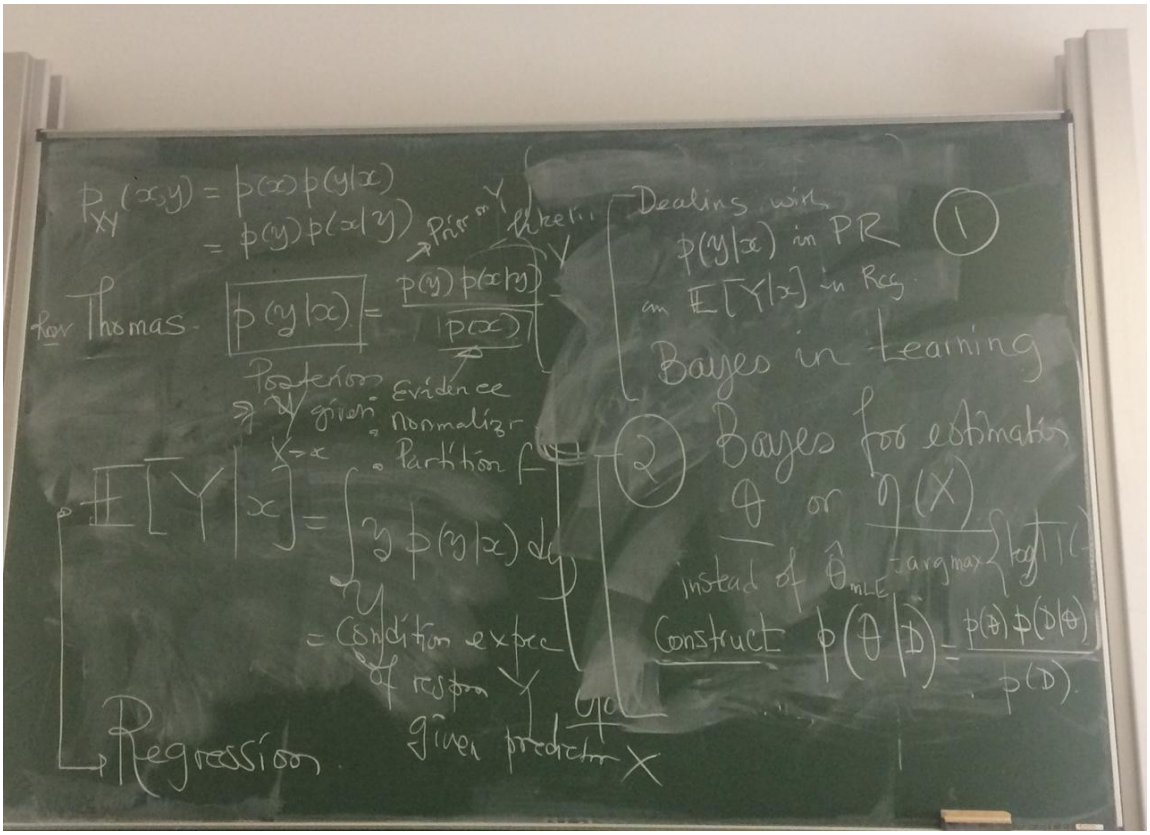
$x_{ij} \to \frac{x_{ij} - \min_j x_{ij}}{\max_j x_{ij} - \min_j x_{ij}}$

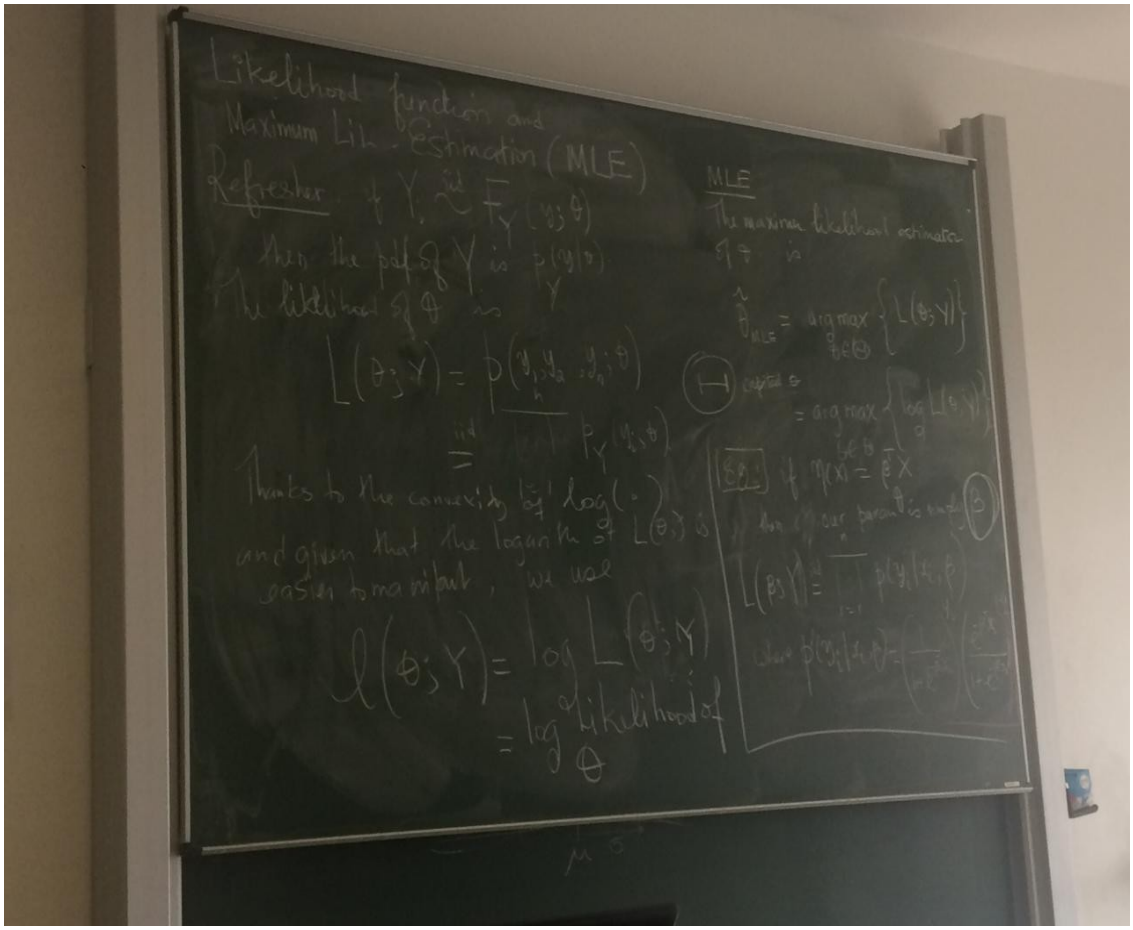
Outlier: $x^* > Q_3 + 3 IQR$

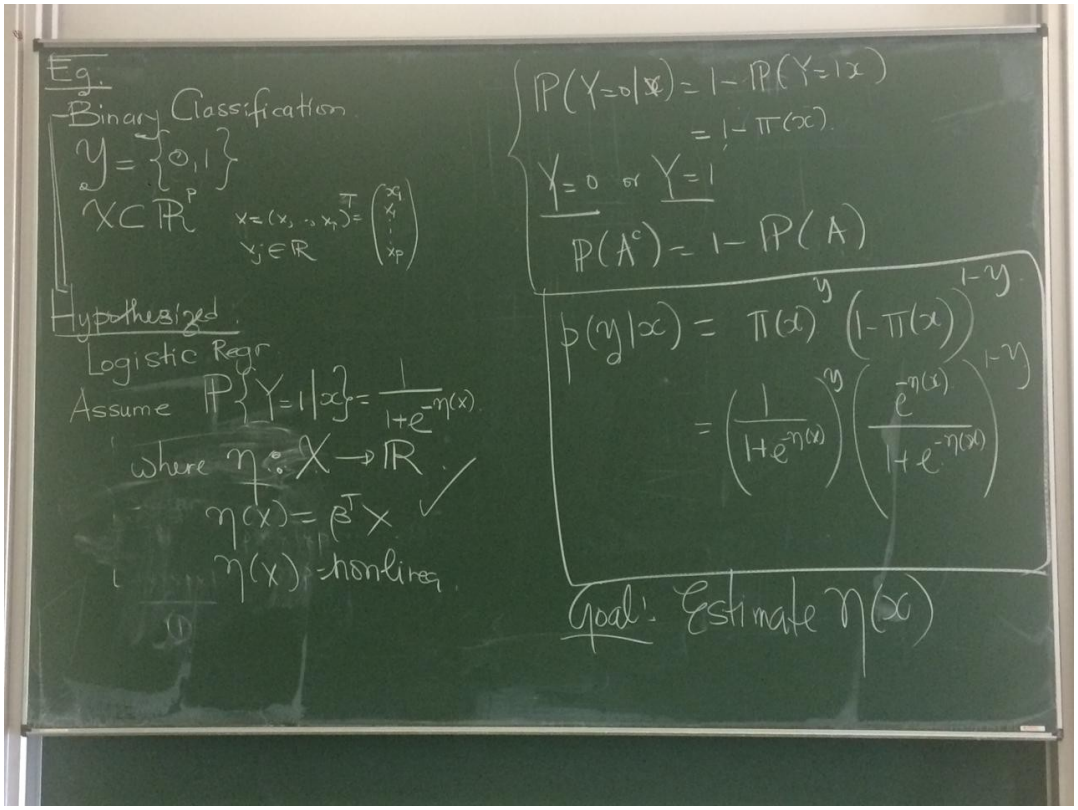


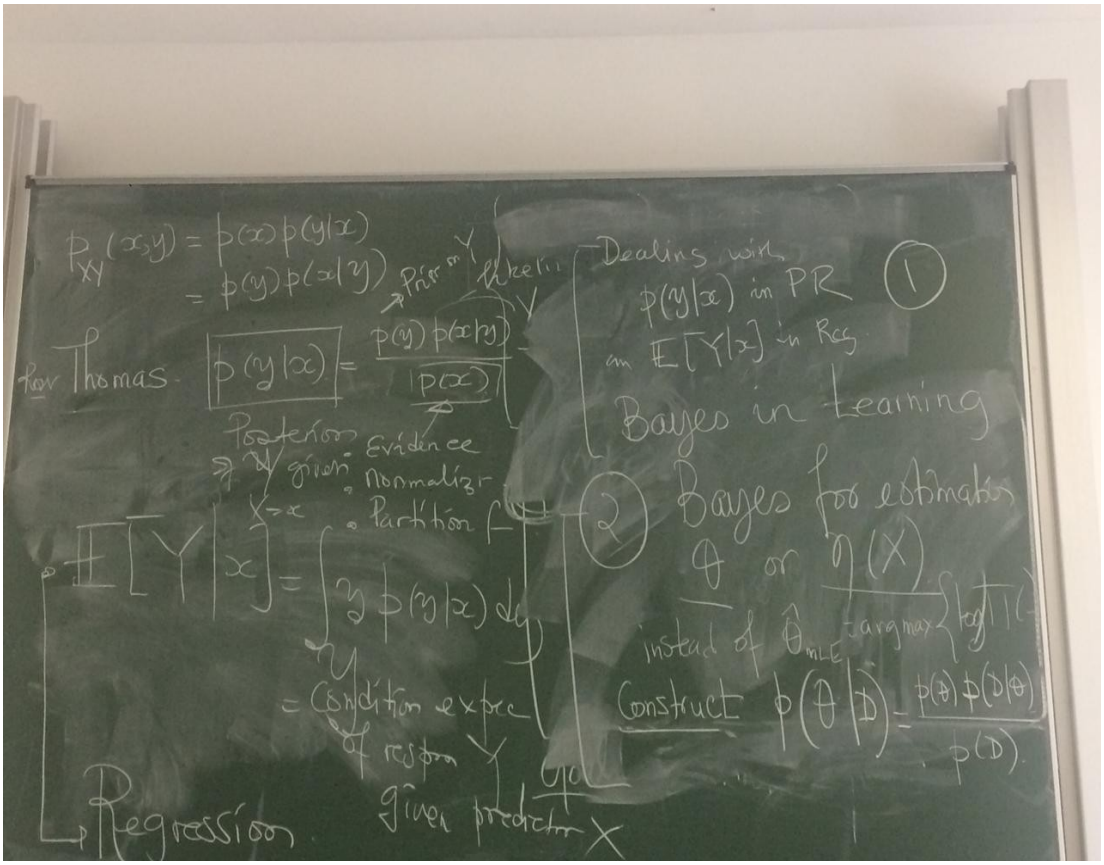

$P_{XY}(x,y) = p(x)p(y|x)$
$= p(y)p(x|y)$
Prior on Y
Likeli..
for Thomas. $p(y|x) = \frac{p(y)p(x|y)}{p(x)}$
Posterior
Evidence
Normalizer
Partition f
$\mathbb{E}[Y|x] = \int_y y\, p(y|x)\, dy$
= Condition expec of respon Y given predictor X
Regression.
Dealing with $p(y|x)$ in PR
or $\mathbb{E}[Y|x]$ in Reg.
Bayes in learning
(2) Bayes for estimation
θ or $\eta(X)$
instead of $\hat{\theta}_{MLE} = \arg\max\{\log l(\theta)$
Construct $p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}$

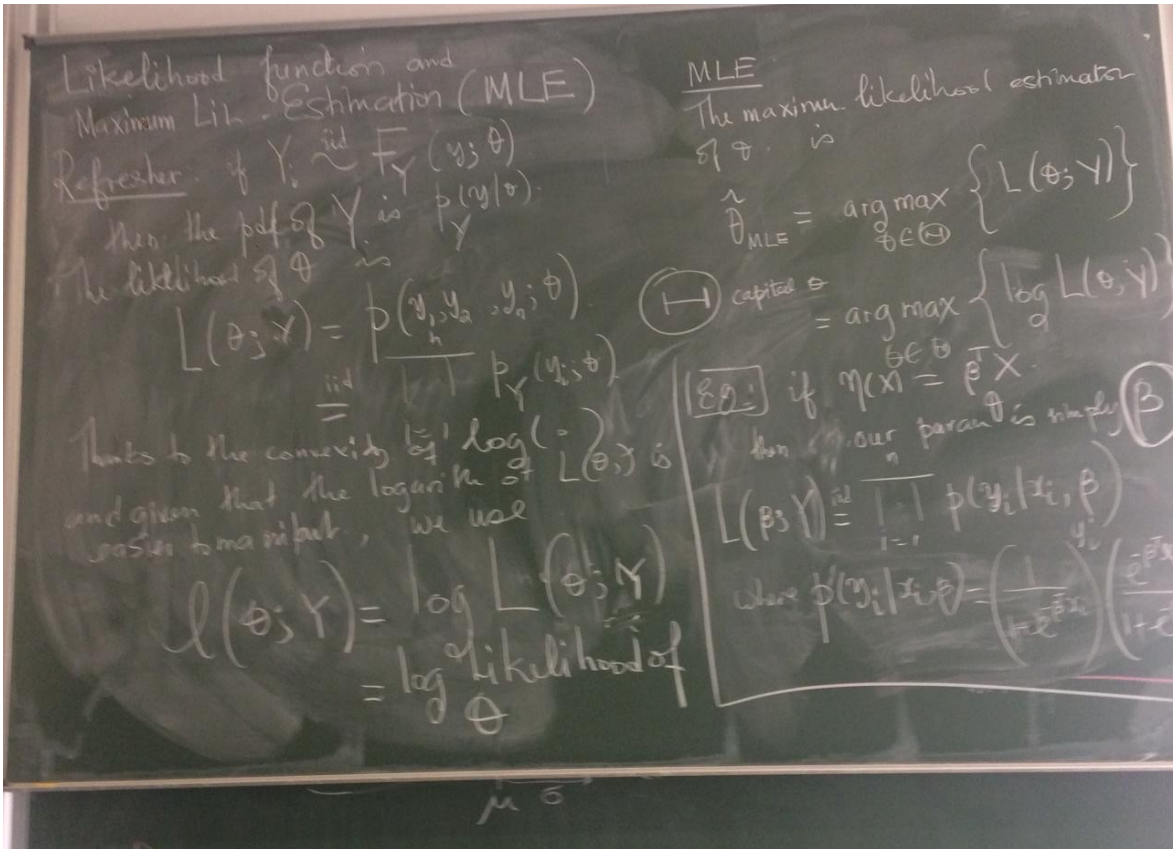

Ex: digit recognition
Y = Output
$= \{0, 1, 2, \ldots, 9\}$
= label of the number
X = Input space
= Vectorized discretized
= Image of the digit
r
c
$= X = (X_1, X_2, \ldots, X_{rc})$
rc
$f: (X_1, X_2, \ldots, X_{rc}) \longrightarrow \{0, 1, 2, \ldots, 9\}$
$P\{Y = j | X\}$
(I) Shape of the data (n, p). $\frac{n}{p} < 1 \Rightarrow$ Ult
(II) Content of data (Type) $\frac{n}{p} > 1$ Thin
Consider the i^{th} obs.
$X_i = (X_{i1}, X_{i2}, \ldots, X_{ip})^T$ p-dim input
What is X_{ij}?
(i) $X_{ij} \in \{0, 1\}$ Binary input
$X_i \in \{0, 1\}^p = \mathcal{X}$
(ii) $X_{ij} \in [0, 1]$ Unitary Input
Neural Networks

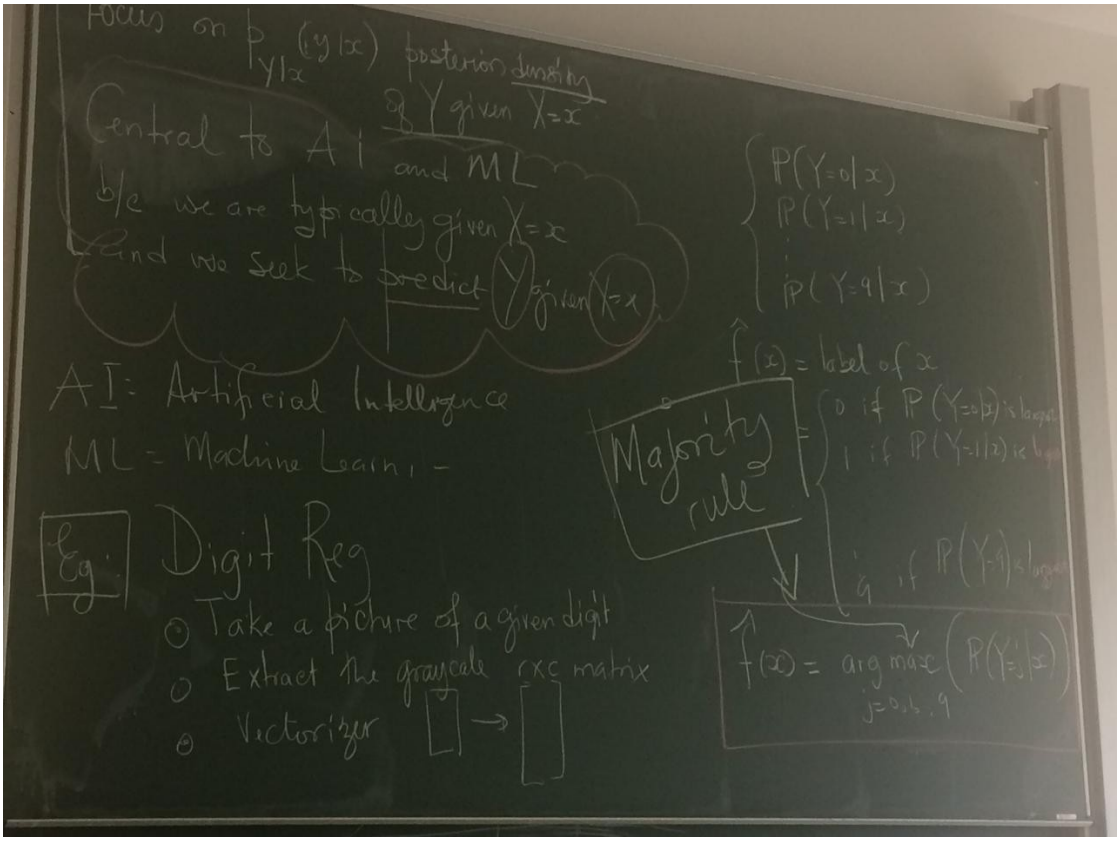

Dealing with
p(y|x) in PR
and E[Y|x] in Reg.
Bayes in Learning
Bayes for estimation
θ or η(X)
instead of θ̂MLE
Construct
Regression
Prior on Y
Likeli.
Posterior
Evidence
normalizer
Partition f
= Condition expec
of respon Y
given predictor X

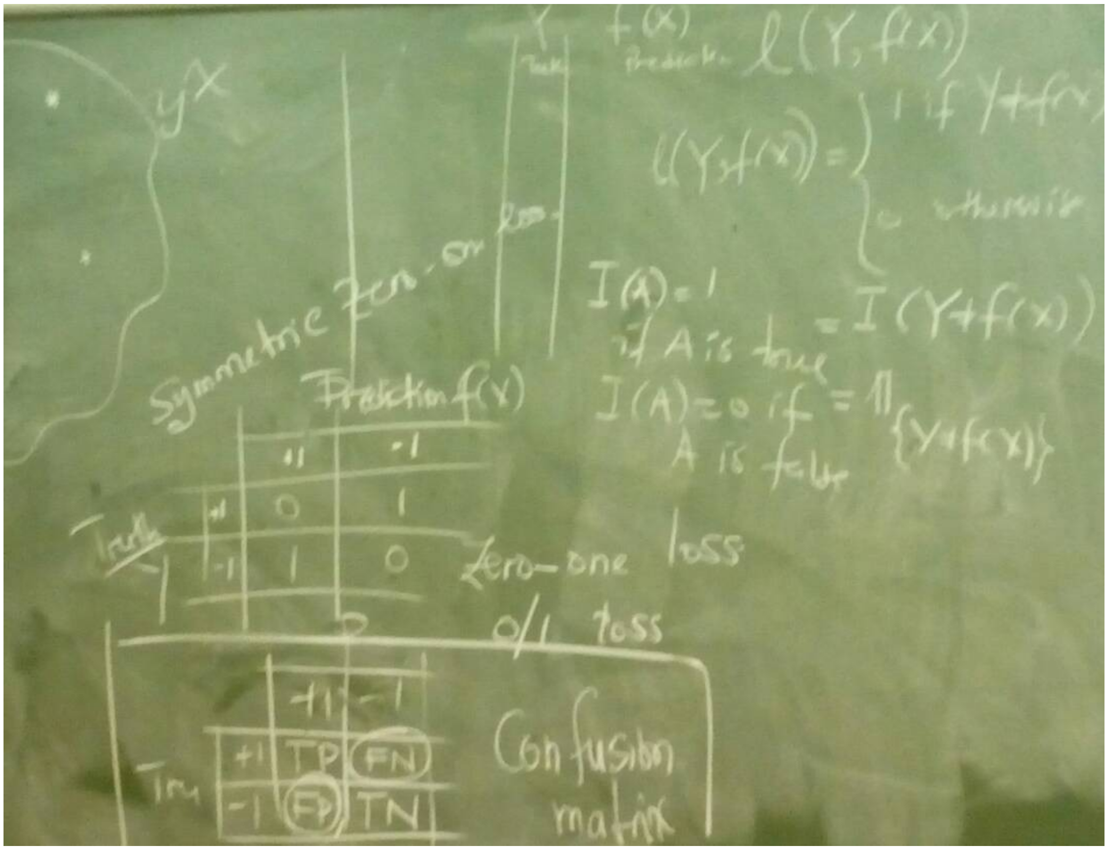

Likelihood function and
Maximum Likelihood estimation (MLE)
Refresher
MLE
The maximum likelihood estimator
Thanks to the convexity of log(·)
and given that the logarithm of L(θ;Y) is
easier to manipulate, we use
l(θ;Y) = log L(θ;Y)
= log likelihood of
θ

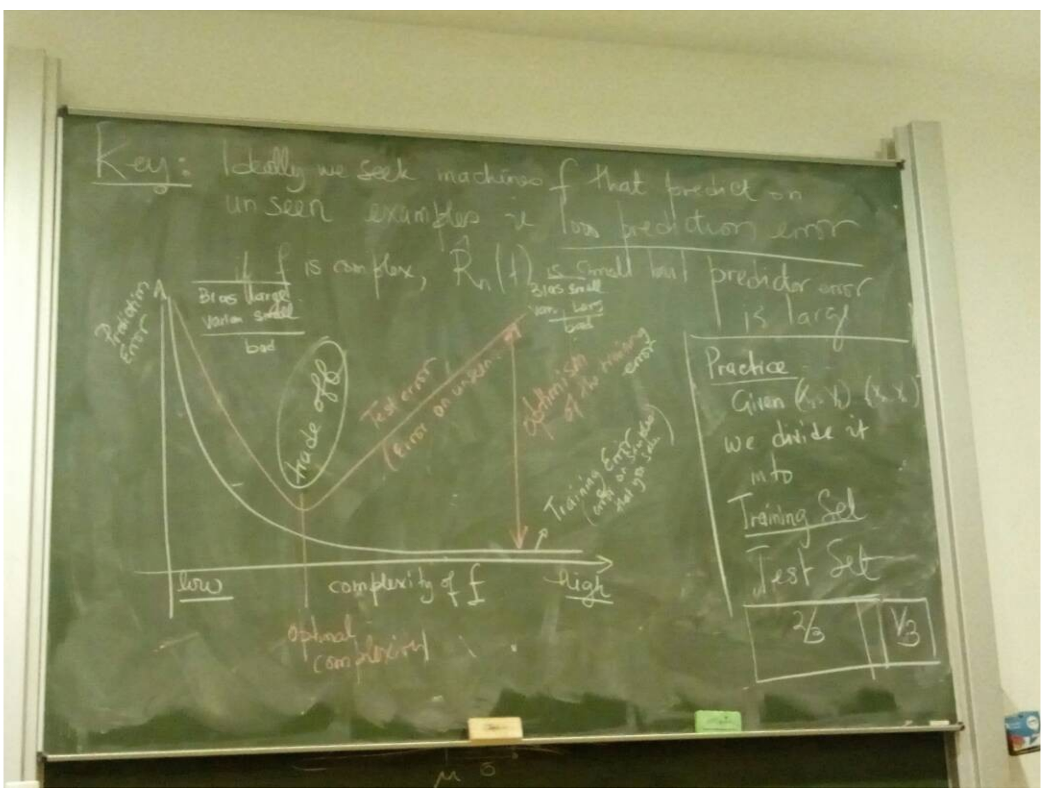

Eg:
Binary Classification
$y = \{0, 1\}$
$X \subset \mathbb{R}^p$
$x = (x_1, \dots, x_p)^T = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix}$
$x_j \in \mathbb{R}$
Hypothesized:
Logistic Regr
Assume $P\{Y=1|x\} = \frac{1}{1+e^{-\eta(x)}}$
where $\eta : X \to \mathbb{R}$
$\eta(x) = \beta^T x$
$\eta(x)$ = nonlinear
$P(Y=0|x) = 1 - P(Y=1|x)$
$= 1 - \pi(x)$
$Y=0$ or $Y=1$
$P(A^c) = 1 - P(A)$
$p(y|x) = \pi(x)^y (1-\pi(x))^{1-y}$
$= \left(\frac{1}{1+e^{-\eta(x)}}\right)^y \left(\frac{e^{-\eta(x)}}{1+e^{-\eta(x)}}\right)^{1-y}$
Goal: Estimate $\eta(x)$

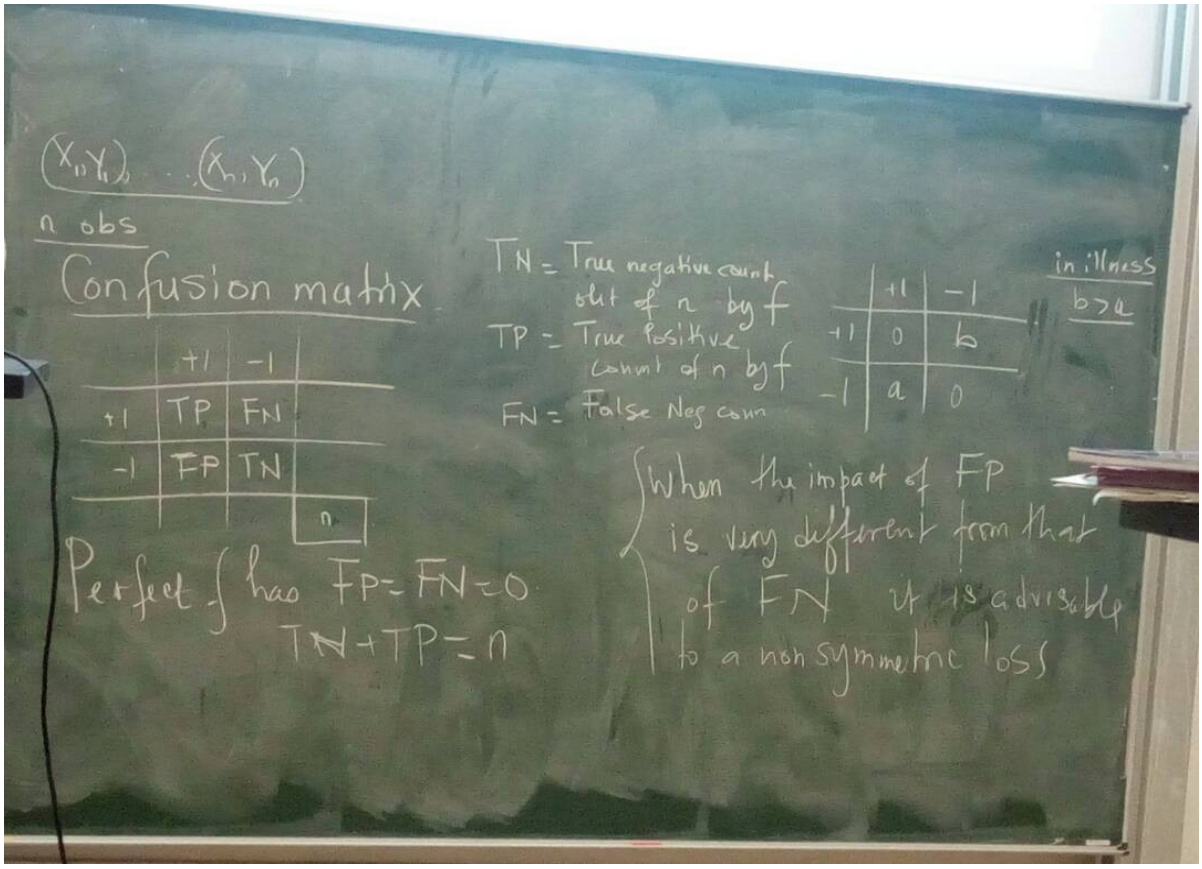

$p_{XY}(x,y) = p(x)p(y|x)$
$= p(y)p(x|y)$
Prior on Y
Likeli...
Rev Thomas $p(y|x) = \frac{p(y)p(x|y)}{p(x)}$
Posterior
Evidence
normaliz-
Partition
$E[Y|x] = \int y\, p(y|x)\, dy$
= Condition expec
of respon
given predictor X
Regression
Dealing with
$p(y|x)$ in PR
or $E[Y|x]$ in Reg
Bayes in Learning
2 Bayes for estimation
θ or $\eta(X)$
instead of $\hat{\theta}_{MLE}$ $\arg\max\{\log$
Construct $p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}$

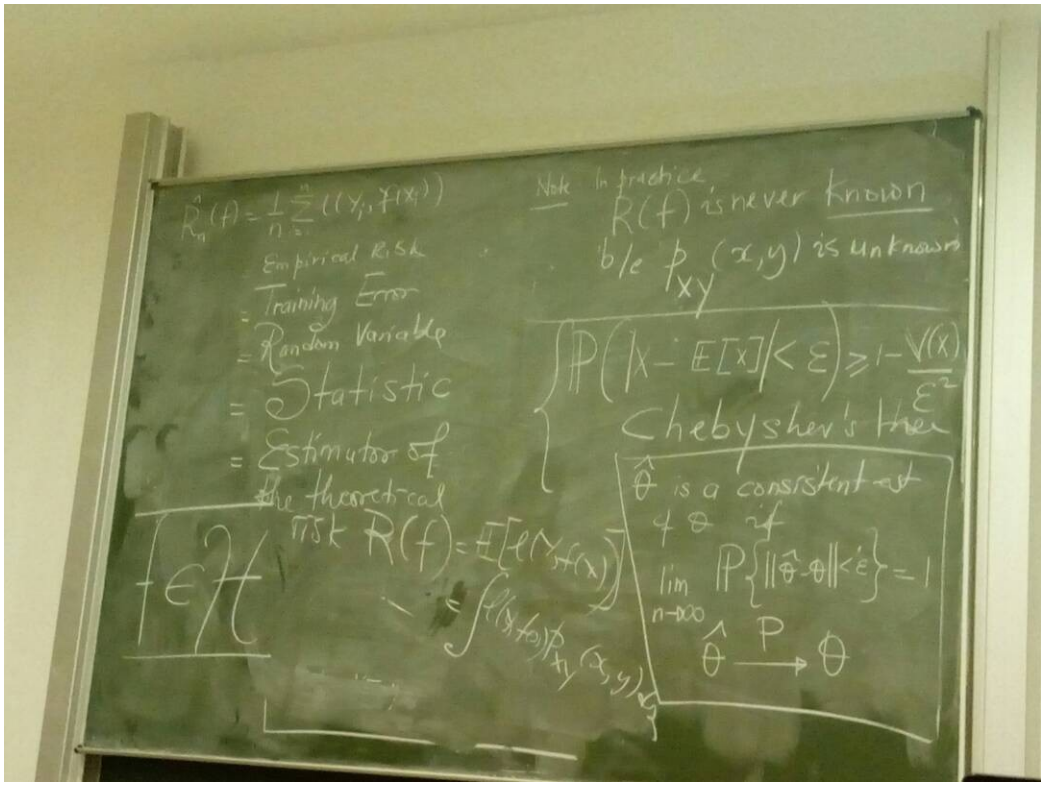

Likelihood function and Maximum Lik. Estimation (MLE)
Refresher: if $Y_i \overset{iid}{\sim} F_Y(y;\theta)$
Then the pdf of Y is $p_Y(y|\theta)$
The likelihood of θ is
$L(\theta;Y) = p(y_1, y_2, \dots, y_n; \theta)$
$\overset{iid}{=} \prod_{i=1}^{n} p_Y(y_i;\theta)$
Thanks to the convexity of $\log(\cdot)$ and given that the log of $L(\theta;Y)$ is easier to manipulate, we use
$\ell(\theta;Y) = \log L(\theta;Y)$
$= \log$ likelihood of θ
MLE
The maximum likelihood estimator of θ is
$\hat{\theta}_{MLE} = \arg\max_{\theta \in \Theta} \{L(\theta;Y)\}$
Θ capital θ
$= \arg\max_{\theta \in \Theta} \{\log L(\theta;Y)\}$
E.G.: if $\eta(x) = \beta^T x$
then our param θ is simply β
$L(\beta;Y) \overset{iid}{=} \prod_{i=1}^{n} p(y_i|x_i,\beta)$
where $p(y_i|x_i,\beta) = \left(\frac{1}{1+e^{\beta^T x_i}}\right)^{y_i} \left(\frac{e^{\beta^T x}}{1+e^{\beta^T x}}\right)$

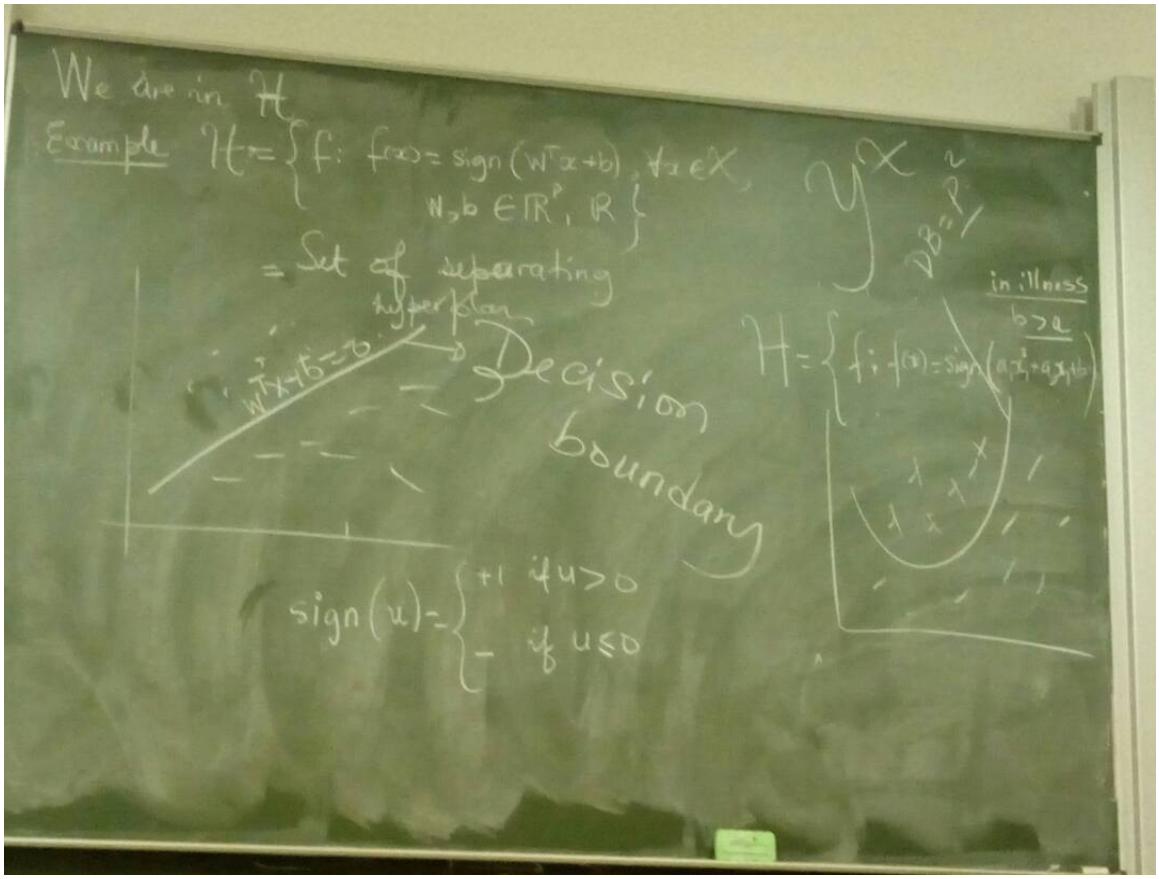

Focus on $p_{Y|x}(y|x)$ posterior density of Y given $X=x$
Central to AI and ML
b/c we are typically given $X=x$
and we seek to predict Y given $X=x$
AI: Artificial Intelligence
ML: Machine Learning
Eg: Digit Reg
Take a picture of a given digit
Extract the grayscale rxc matrix
Vectorize [ ] → [ ]
$\begin{cases} P(Y=0|x) \\ P(Y=1|x) \\ \vdots \\ P(Y=9|x) \end{cases}$
$\hat{f}(x)$ = label of x
Majority rule
$= \begin{cases} 0 \text{ if } P(Y=0|x) \text{ is largest} \\ 1 \text{ if } P(Y=1|x) \text{ is largest} \\ \vdots \\ 9 \text{ if } P(Y=9|x) \text{ is largest} \end{cases}$
$\hat{f}(x) = \arg\max_{j=0,1,\dots,9} \left(P(Y=j|x)\right)$

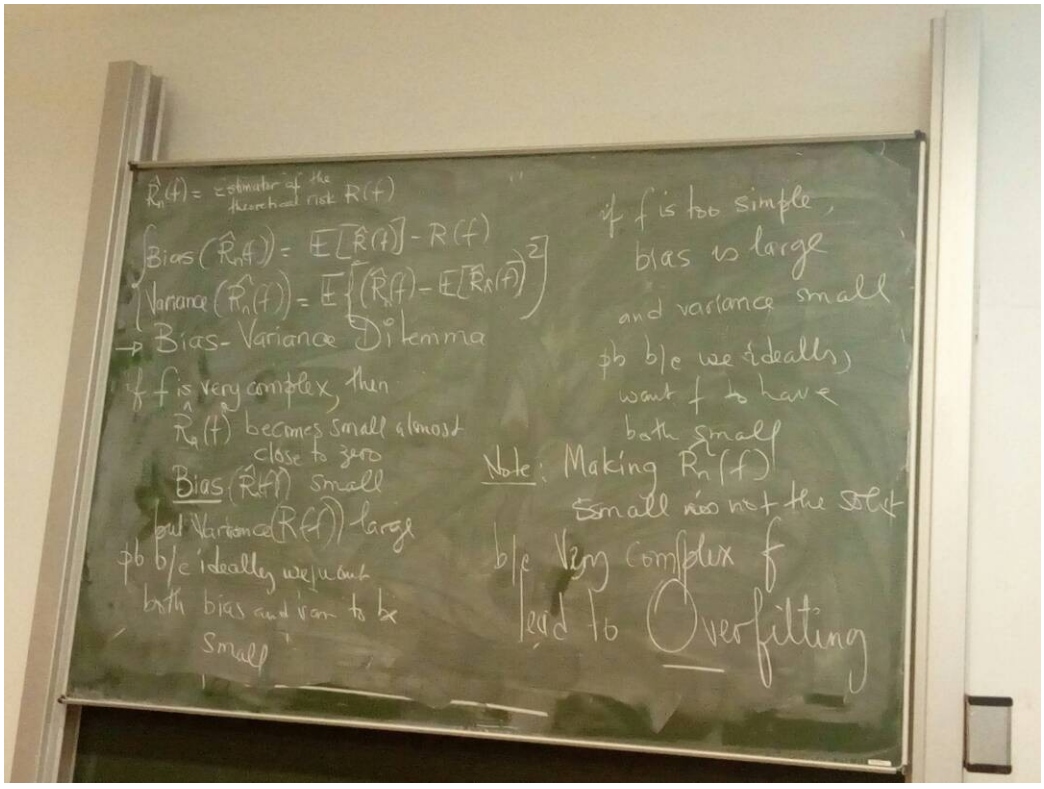

Symmetric zero-one loss
Prediction f(x)
Truth
I(A) = 1 if A is true
I(A) = 0 if A is false
otherwise
Zero-one loss
0/1 loss
+1 -1
TP FN
FP TN
Confusion matrix

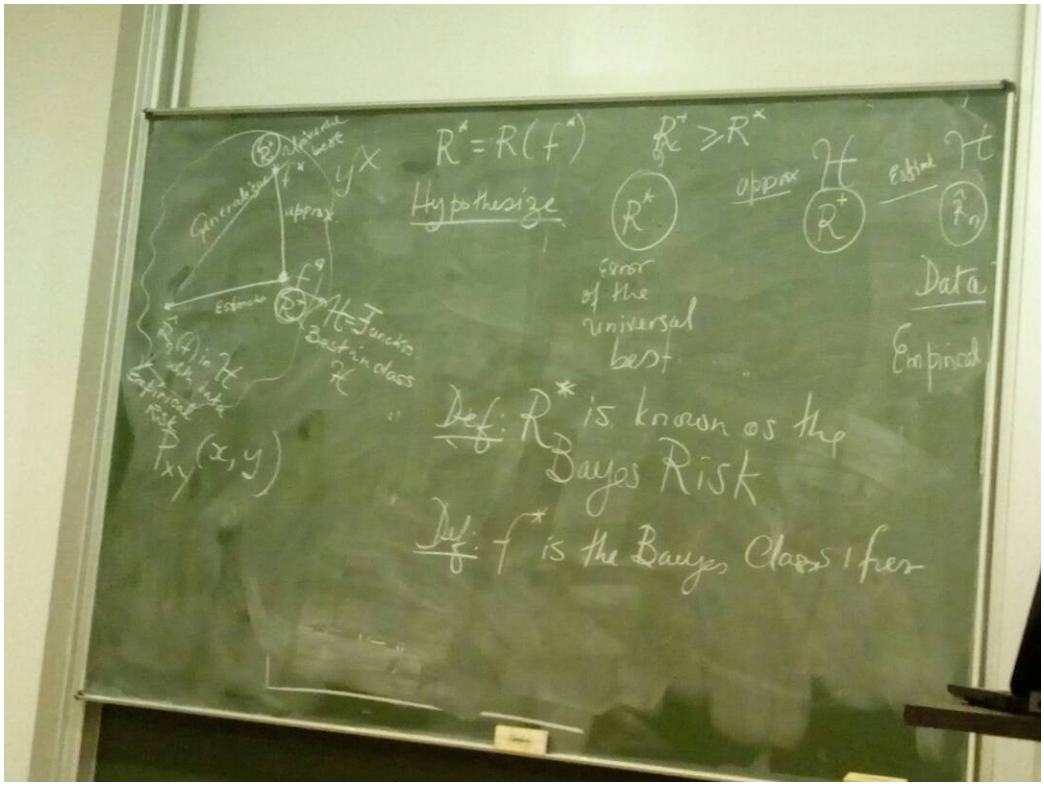

Key: Ideally we seek machines f that predict on unseen examples i.e. low prediction error
is complex
but predictor error is large
Prediction Error
Bias large
bad
trade off
Test error
Training Error
low
complexity of f
high
optimal complexity
Practice
we divide it into
Training Set
Test Set
2/3
1/3

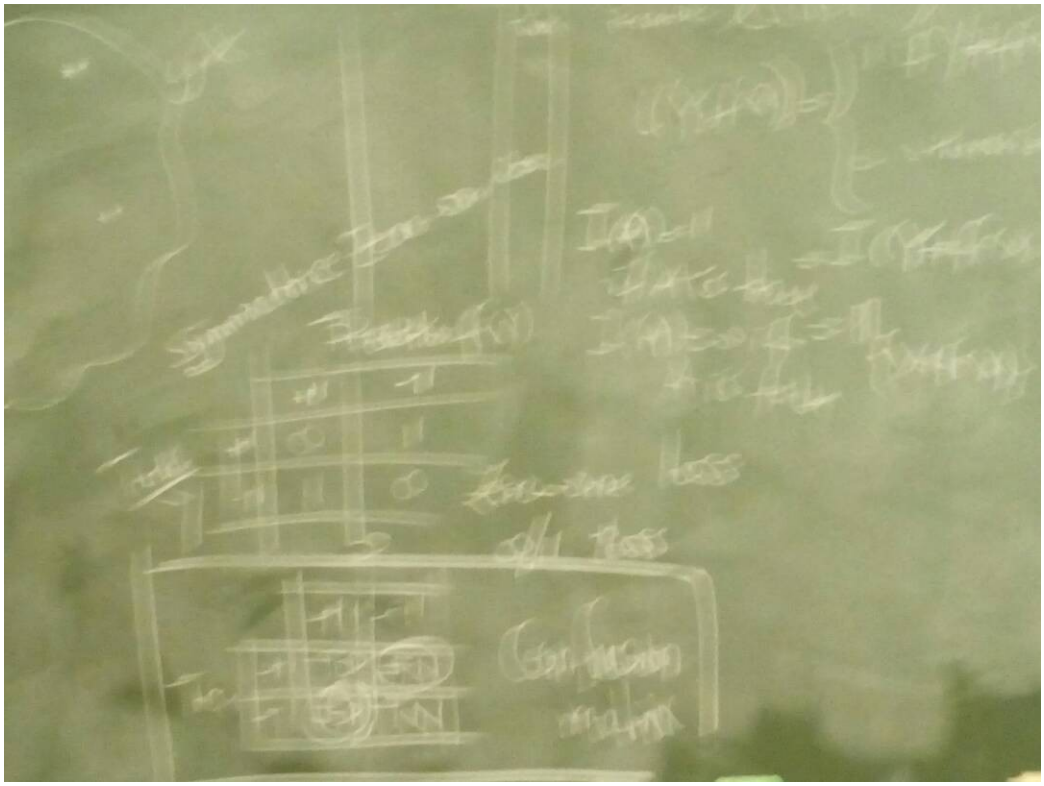

$(X_1, Y_1) \dots (X_n, Y_n)$
n obs
Confusion matrix
TN = True negative count out of n by f
TP = True Positive count of n by f
FN = False Neg coun
+1 -1
+1 TP FN
-1 FP TN
n
Perfect f has FP = FN = 0
TN + TP = n
+1 -1
+1 0 b
-1 a 0
in illness
b > a
When the impact of FP is very different from that of FN it is advisable to a non symmetric loss

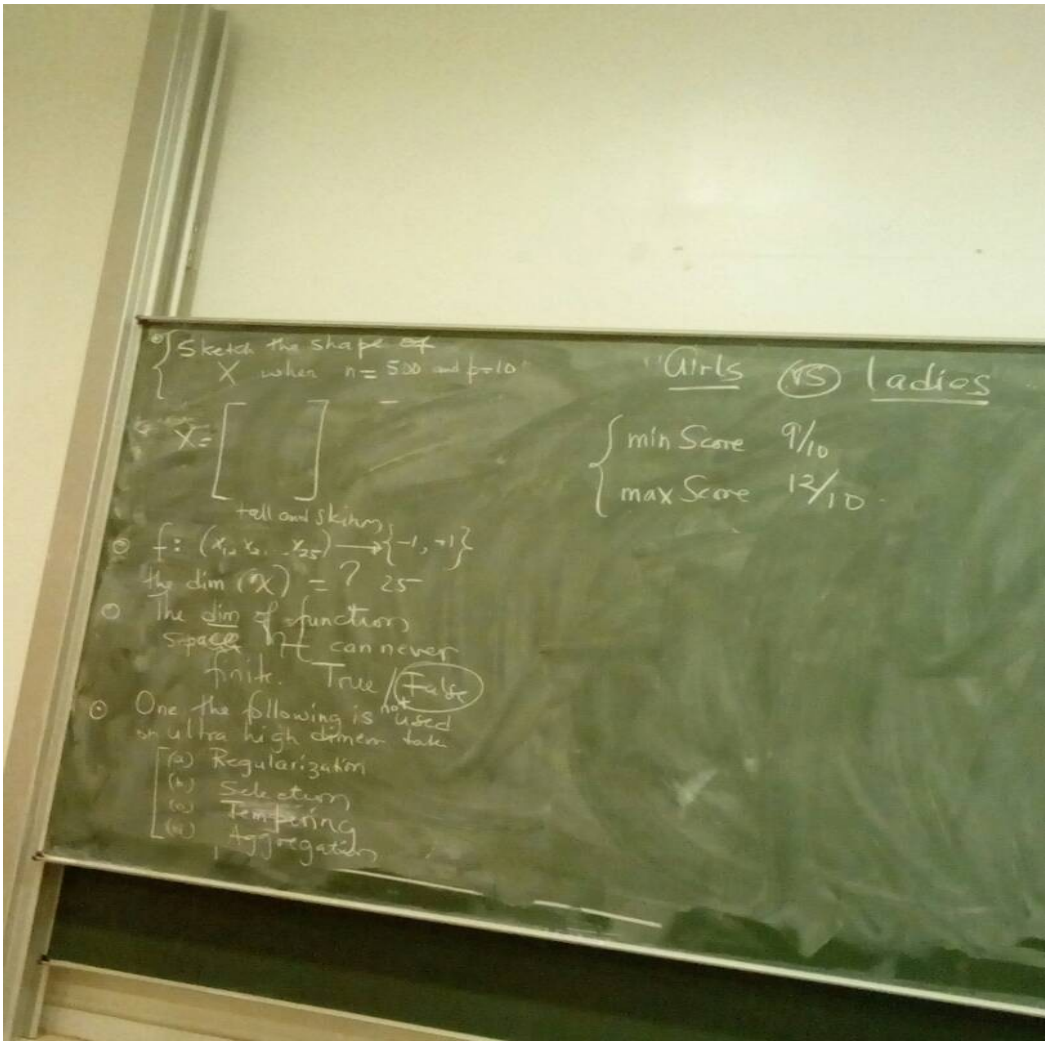

$\hat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$
Empirical Risk
= Training Error
= Random Variable
= Statistic
= Estimator of the theoretical
$f \in \mathcal{H}$
risk $R(f) = E[L(Y, f(X))]$
$= \int L(y, f(x)) P_{XY}(x, y)$
Note In practice $R(f)$ is never known
ble $P_{XY}(x, y)$ is unknown
$P(|X - E[X]| < \varepsilon) \geq 1 - \frac{V(X)}{\varepsilon^2}$
Chebyshev's the
$\hat{\theta}$ is a consistent est
of θ if
$\lim_{n \to \infty} P\{\|\hat{\theta} - \theta\| < \varepsilon\} = 1$
$\hat{\theta} \xrightarrow{P} \theta$


We are in $\mathcal{H}$
Example $\mathcal{H} = \{f : f(x) = \text{sign}(w^T x + b), \forall x \in \mathcal{X}, w, b \in \mathbb{R}^2, \mathbb{R}\}$
= Set of separating hyperplan
$w^T x + b = 0$
Decision boundary
$\text{sign}(u) = \begin{cases} +1 & \text{if } u > 0 \\ - & \text{if } u \leq 0 \end{cases}$
$\mathcal{X}$
$\mathcal{Y}$
DB = ?
in illness
$b > a$
$H = \{f : f(x) = \text{sign}(ax_1^2 + ax_2 + b)$


$\hat{R}_n(f)$ = estimator of the theoretical risk $R(f)$
$\text{Bias}(\hat{R}_n(f)) = E[\hat{R}(f)] - R(f)$
$\text{Variance}(\hat{R}_n(f)) = E\left[\left(\hat{R}_n(f) - E[\hat{R}_n(f)]\right)^2\right]$
→ Bias-Variance Dilemma
if f is very complex, then
$\hat{R}_n(f)$ becomes small almost close to zero
Bias $(\hat{R}(f))$ small
but Variance $(\hat{R}(f))$ large
pb b/c ideally we would want both bias and vari to be small
if f is too simple, bias is large and variance small
pb b/c we ideally want f to have both small
Note: Making $\hat{R}_n(f)$ small is not the best b/c Very complex f lead to Overfitting


$R^* = R(f^*)$
$R_{\mathcal{H}}^* \geq R^*$
Hypothesize
Upper
Estim
Data
Empirical
Error of the universal best
Def: R^* is known as the Bayes Risk
Def: f^* is the Bayes Classifier
Generalization
Approx
Estimation
Empirical Risk
$R_{xy}(x,y)$




Sketch the shape of X when n = 500 and p = 10
X =
tall and skinny
f: (x1, x2, ... x25) → {-1, +1}
dim(X) = ? 25
The dim of function space H can never finite. True False
One the following is not used in ultra high dimen. task
(a) Regularization
(b) Selection
(c) Tempering
(d) Aggregation
Girls VS ladies
min Score 9/10
max Score 12/10