Machine Learning for Healthcare
6.7930 [6.871], HST.956

Lecture 16:

Learning with Noisy Labels, Unsupervised Learning Applications, Weak Supervision

Prof. Manolis Kellis

Slides credit:
David Sontag

Outline for today's class

1. **Learning with noisy labels**
   - Consistent estimation under class-conditional noise (Natarajan et al., NeurIPS '13)
   - Application in health care (Halpern et al., JAMIA '16)
2. Weak supervision

Labels may be noisy

e.g. Predicting who will develop Diabetes.

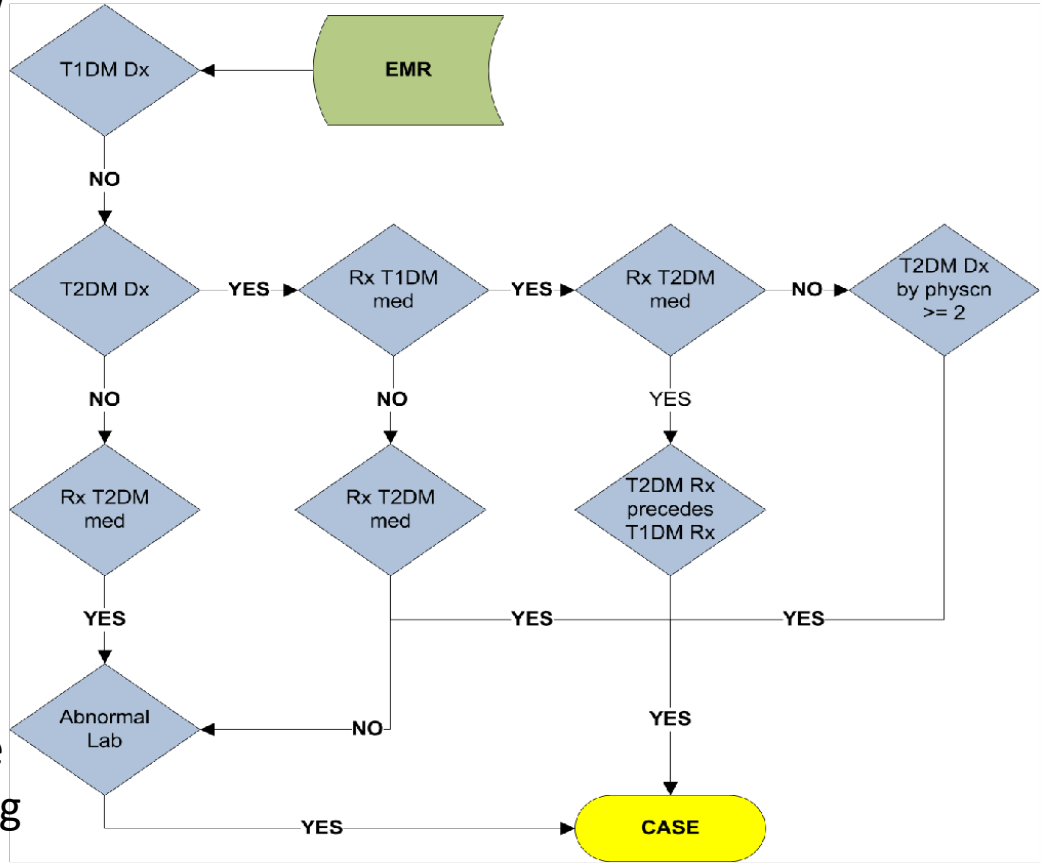
- Who has diabetes at time T + Window W

- Decision tree based on:
  - Diagnosis for T1D or T2D
  - Medications for T1D or T2D
  - Laboratory test *result* for T1D or T2D
  - Lab test *ordering* for T1D or T2d

➔ Some 'hints' are informative, but noisy
➔ Some resulting labels will be wrong
➔ How does it affect learning?

e.g. Billing codes, not diagnosis codes.
e.g. Lab test might actually come negative
Both false pos and false neg affect learning
➔ How to deal with noisy labels?

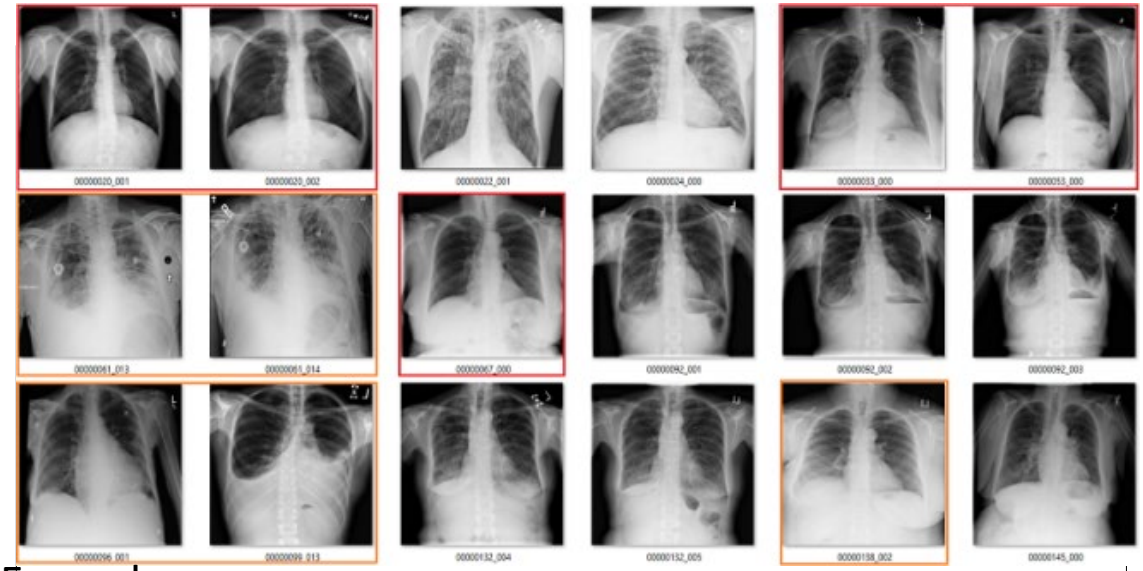
Figure 1: Algorithm for identifying T2DM cases in the EMR.



Source: https://phekb.org/sites/phenotype/files/T2DM-algorithm.pdf

Labels may be noisy

Fibrosis



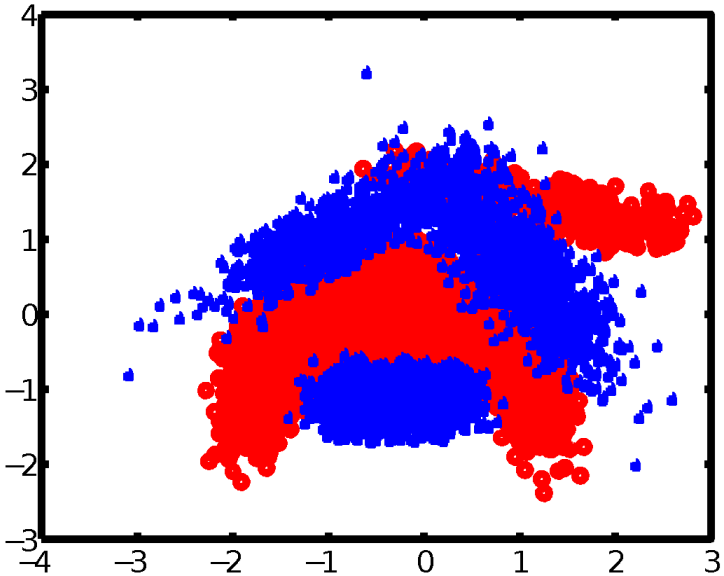
red = mislabeled
orange = maybe mislabeled

Example:

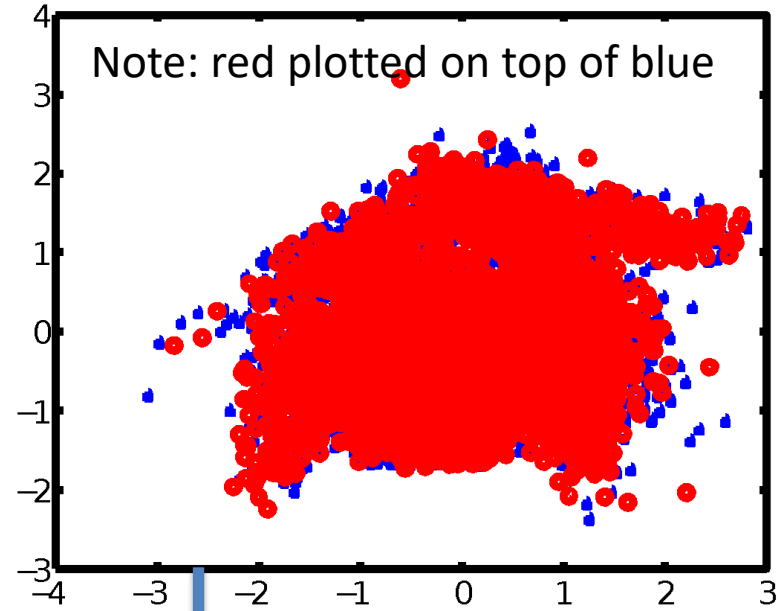
- Dataset released for training machine learning for chest x-ray
- Machine learning algorithms trained on it
- No-one had actually looked at the dataset closely
- Closer inspect revealed many errors and incorrectly-labeled images

➔ Can we ever build perfect ground-truth datasets? Perhaps not.
➔ How does that affect machine learning and the resulting models?
➔ How much noise can our ML algorithms tolerate and still learn correct features?
➔ Can we build machine learning methods that are robust to noisy labels?
➔ Can we 'bootstrap' from ML results to flag problematic labels in our dataset? =

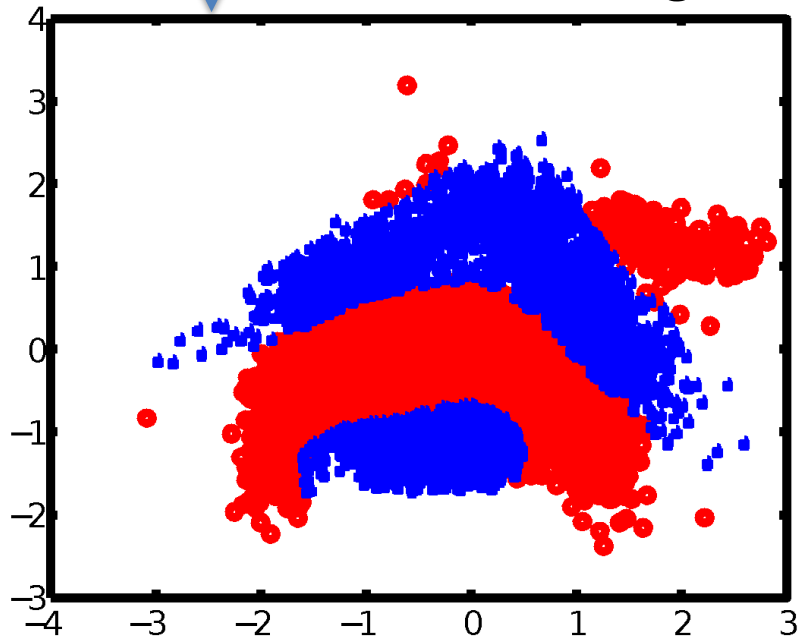
[Wang et al., "Chest X-ray8"]
figure credit: https://lukeoakdenrayner.wordpress.com/2017/12/18/the-chestxray14-dataset-problems/



add 40% label noise



Machine learning



Synthetic dataset

- Two features Red/Blue, +1/-1.

Add noise:

- Flip label for 40% of the dataset
- High noise rate: 0.4

Train classifier.

- Still learns correct function
- Can still accurately predict original 'true' label, even for artificially-flipped-label data points

Reflections:

- Makes you wonder if some original blue points in 'red regions' were incorrectly-labelled as well
- Sufficient generalization should allow you to 'question' some incorrectly labelled data points
- But if you only fix those, does that add bias, what about all those other incorrect ones you don't catch?

[Natarajan et al., NeurIPS '13. Figure 2]

Learning with noisy labels

We will show that if we have

a) ***class-conditional*** label noise and

b) ***lots*** of training data,

learning as usual, substituting noisy labels, works!

This opens the door to using noisy labels for training, and coming up with clever ways of deriving these for free

- We won't know that we're doing really well.
- Our calculated "False positive rate" will be quite high
  (cuz some of our 'false' predictions will actually be true)
- But our model parameters will be quite good
  (good generalization power for future data,
  and in fact more accurate for some of our 'ground-truth' data that's incorrectly labelled)

(Natarajan et al., *Learning with Noisy Labels.* NeurIPS '13)

Natarajan et al: Introduction

- Features X
- True unobserved labels $Y \in \{-1, 1\}$
- Noisy observed labels $\tilde{Y} \in \{-1, 1\}$
- True distribution $P(X, Y, \tilde{Y})$

$\boldsymbol{X}$ (age)	$\boldsymbol{Y}$ (diabetic)	$\tilde{\boldsymbol{Y}}$ (noisy version)
30	-1	
64	1	
75	1	

- Reason about joint distribution of X, Y, Y~
  (even though we do not observe true labels Y,
  still reason about joint distribution to prove guarantees about them)
- Assume process generating all three variables

Natarajan, NeurIPS 2013, “Learning with Noisy Labels”

Natarajan et al: Introduction

- Features X
- True unobserved labels $Y \in \{-1, 1\}$
- Noisy observed labels $\tilde{Y} \in \{-1, 1\}$ (Mnemonic: approx. Y: ~Y ➔ Ỹ)
- True distribution $P(X, Y, \tilde{Y})$

X (age)	Y (diabetic)	$\tilde{Y}$ (noisy version)
30	-1	-1
64	1	1
75	1	-1

- Data sampled from $P(X, \tilde{Y}) = \sum_y P(X, Y = y, \tilde{Y})$

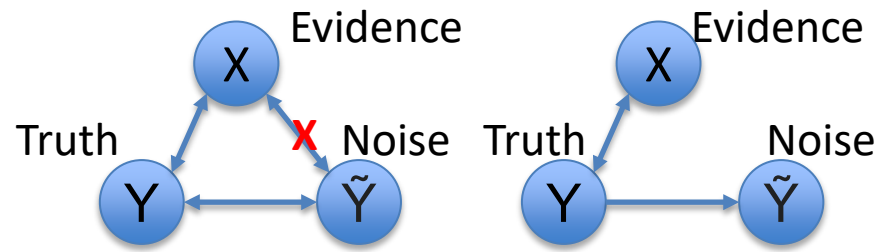
- Marginalize over Y
- Y is binary 1 or -1
- Marginal for each of the two options

Y *exists*, but it is hidden during training

X (age)	$\tilde{Y}$ (noisy version)
30	-1
64	1
75	-1

(Natarajan et al., *Learning with Noisy Labels.* NeurIPS '13)

Assumption: class-conditional label noise



Note: X→Y: causality. Y→X inference

- Assume that $\tilde{Y} \perp\!\!: X|Y$

$$P(X,Y,\tilde{Y}) = P(X,Y)\underline{P(\tilde{Y}|Y)}$$

$\tilde{Y}$ only depends on Y: label noise is independent of input features

- **C**lass-**C**onditional random label **N**oise (CCN): Misclassification noise only depends on Y, not X!
- Since Y is binary, need two parameters to fully define $P(\tilde{Y}|Y)$
- Misclassification rate ρ, separate for true pos (ρ_+), true neg (ρ_-)

flipping from +1 to -1= ρ_+ $\quad \boxed{\rho_+ = P(\tilde{Y} = -1|Y = 1)}$

flipping from -1 to +1: ρ_- $\quad \boxed{\rho_- = P(\tilde{Y} = 1|Y = -1)}$

- Assume (for now) that these rates are known
  - However, the method works when weights are not known
- Only constraint: $\rho_+ + \rho_- < 1$, but either could be large

Learning with class-conditional noise

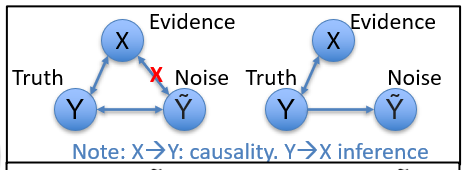
- If we could learn $\eta(X) = P(Y=1|X)$, (<u>prob of true 1 given data X</u>), then we would be able to predict optimally. (but we've never observed Y, so how can we estimate it?)
- Instead, start by estimating <u>prob of noisy-label Y=1 given data X</u>

$$\tilde{\eta}(X) = P(\tilde{Y}=1|X)$$

$$= P(\tilde{Y}=1, Y=1|X) \quad + P(\tilde{Y}=1, Y=-1|X)$$

Separately for True and for False, only two options ➔ Marginalize

$$= P(Y=1|X)P(\tilde{Y}=1|Y=1)$$
$$+ P(Y=-1|X)P(\tilde{Y}=1|Y=-1)$$



$$P(X, Y, \tilde{Y}) = P(X,Y)P(\tilde{Y}|Y)$$

Independence assumption

$$= \eta(X)(1-\rho_+) + (1-\eta(X))\rho_-$$

$$\rho_+ = P(\tilde{Y}=-1|Y=1)$$

$$\rho_- = P(\tilde{Y}=1|Y=-1)$$

$$= \eta(X)(1-\rho_+-\rho_-) + \rho_-$$

$$\rightarrow \eta(X) = \frac{\tilde{\eta}(X)-\rho_-}{1-\rho_+-\rho_-}$$

(Natarajan et al., *Learning with Noisy Labels.* NeurIPS '13)

Learning with class-conditional noise

$$\rightarrow \eta(X) = \frac{\tilde{\eta}(X) - \rho_-}{1 - \rho_+ - \rho_-}$$

$\eta(X)$ is monotonically increasing in $\tilde{\eta}(X)$

- Learn $\tilde{\eta}(X)$ using any ML algorithm which returns calibrated classifiers. Substitute $\tilde{\eta}(X)$ in the above equation to get $\eta(X)$!
- Thus: learn as if the labels were correct
  - ➔ Same ordering of true/false classification regardless.
  - ➔ Simple transformation to get actual true confidence
- Denominator must be >0 (why we needed $\rho_+ + \rho_- < 1$ constraint) [notice, if exactly 1, divide by zero, not good. If >1, could flip labels]
- When might noise be helpful?
  - Privacy: e.g. sexually transmitted disease. Deniability
  - Force generalization, by avoiding overfitting
  - Could *predict* true classes, but only if perfect predictor (typically impossible)

(Natarajan et al., *Learning with Noisy Labels.* NeurIPS '13)

Outline for today's class

1. Learning with noisy labels
   - Consistent estimation under class-conditional noise (Natarajan et al., NeurIPS '13)
   - **Application in health care (Halpern et al., JAMIA '16)**
2. Weak supervision

Application to electronic phenotyping

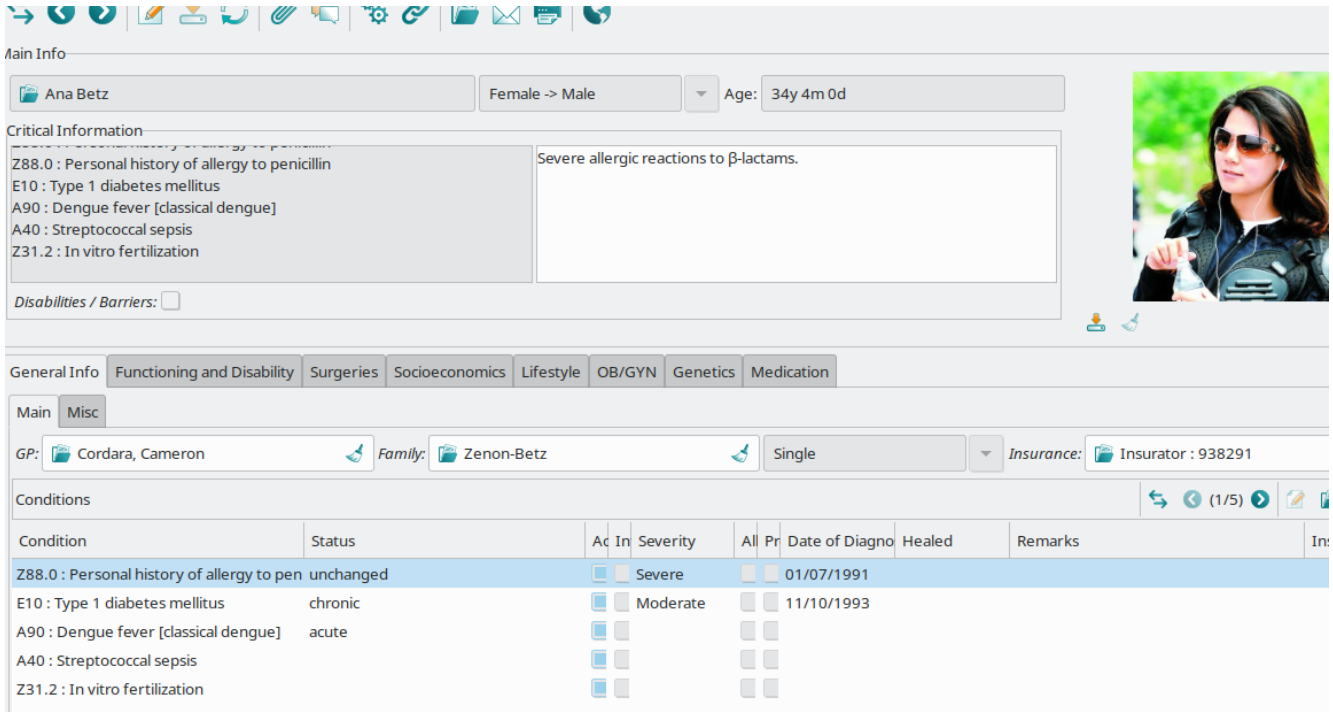


Figure source:
https://commons.wikimedia.org/wiki/File:GNU_Health_patient_main_screen.png

Hundreds of relevant clinical variables

Abdominal pain
Active malignancy
Altered mental status
Cardiac etiology
Renal failure
Infection
Urinary tract infection
Shock
Smoker
Pregnant
Lower back pain
Motor Vehicle accident
Psychosis
Anticoagulated
Type II diabetes
…

[Halpern, Horng, Choi, Sontag, AMIA ’14; Halpern, Horng, Choi, Sontag, JAMIA ‘16]

Simplest approach: rules

- We would like to estimate, for every patient, which phenotypes apply to them (at some point in time)
- Common practice is to derive manual rules:

Expand to include:
nursing facility
nursing care facility
nursing / rehab
nsg facility
nsg faclty (typos)
…

physician response (gold standard)

text contains: *"nursing home"*	Nursing home? T	Nursing home? F	
T	297	129	PPV 0.70
F	1,319	34511	
	Sensitivity 0.18		

297/(297+129)

Slow, expensive, poor sensitivity.

Often we can find noisy labels WITHIN the data!

Phenotype	Example of noisy label ("anchor") ⚓
Diabetic (type I)	gsn:016313 (insulin) in **Medications**
Strep Throat	Positive strep test in **Lab results**
Nursing home	"from nursing home" in **Text**
Pneumonia	"pna" in **Text**
Heart attack	ICD10 I21 in **Billing codes**

How can we use these for machine learning?

Often we can find noisy labels WITHIN the data!

Phenotype	Example of noisy label (anchor) ⚓
Heart attack	ICD10 I21 in **Billing codes**

- Suppose we want to know, was a patient admitted to the emergency department for a heart attack?
- Billing codes not available at prediction time (not useful for real-time use), but can be used for labels
- Reasonable to assume that $\rho_- = P\left(\tilde{Y} = 1 \middle| Y = -1\right) \approx 0$ (fraud), but because of noisy nature of billing codes, $\rho_+ = P\left(\tilde{Y} = -1 \middle| Y = 1\right)$ likely non-zero

Called *"positive only" noise* since it implies $P\left(Y = 1 \middle| \tilde{Y} = 1\right) = 1$

Anchor & Learn Algorithm

(special case for **<u>anchors derived from future data</u>**)

Training

1. Treat the anchors (noisy labels) as "true" labels then censor them from the dataset *(as they don't appear in practice for real-time prediction)*
2. Learn a classifier to predict presence/absence of anchor *(whether noisy-label (anchor) $\tilde{Y}$ appears)*
3. Calibration step: divide by $\frac{1}{|P|}\sum_P P(\tilde{Y} = 1|X)$

P = data points with $\tilde{Y} = 1$

Test time

1. First check if anchor is present.
   - If yes: high value, cuz of high positive predictive value of anchor
   - If not: apply the learned classifier + multiply by calibration constant

Often we can find noisy labels WITHIN the data!

Phenotype	Example of noisy label (anchor) ⚓
Nursing home	"from nursing home" in **Triage note**

- We again assume that $\rho_- = P(\tilde{Y} = 1 | Y = -1) \approx 0$, but because many ways to write "from nursing home" in text, we have $\rho_+ = P(\tilde{Y} = -1 | Y = 1)$ likely non-zero
- If we simply learn to predict $\tilde{Y}$ using the notes, we will learn a trivial classifier! It will simply extract mentions of this phrase!
- This is a clear violation of the assumption $\tilde{Y} \perp X | Y$, since $\tilde{Y}$ is derived from X

*In this dataset, we have some natural candidates for anchors that we can take from different parts of the record. For example, we may use **medications** that are specific to a single disease. **Lab tests**, **phrases** in the patient notes, or ICD9 or 10 **billing codes**.*

Anchor & Learn Algorithm

Training

1. Treat the anchors as "true" labels
2. Learn a classifier to predict whether the ***anchor*** appears based on ***all other features*** *(throw away all features used to predict true label)*
3. Calibration step: divide by $\frac{1}{|P|}\sum_P P(\tilde{Y} = 1|X)$

Test time

P = data points with $\tilde{Y} = 1$

1. If the anchor is present: Predict 1
2. Else: Predict using the learned classifier (with calibration)

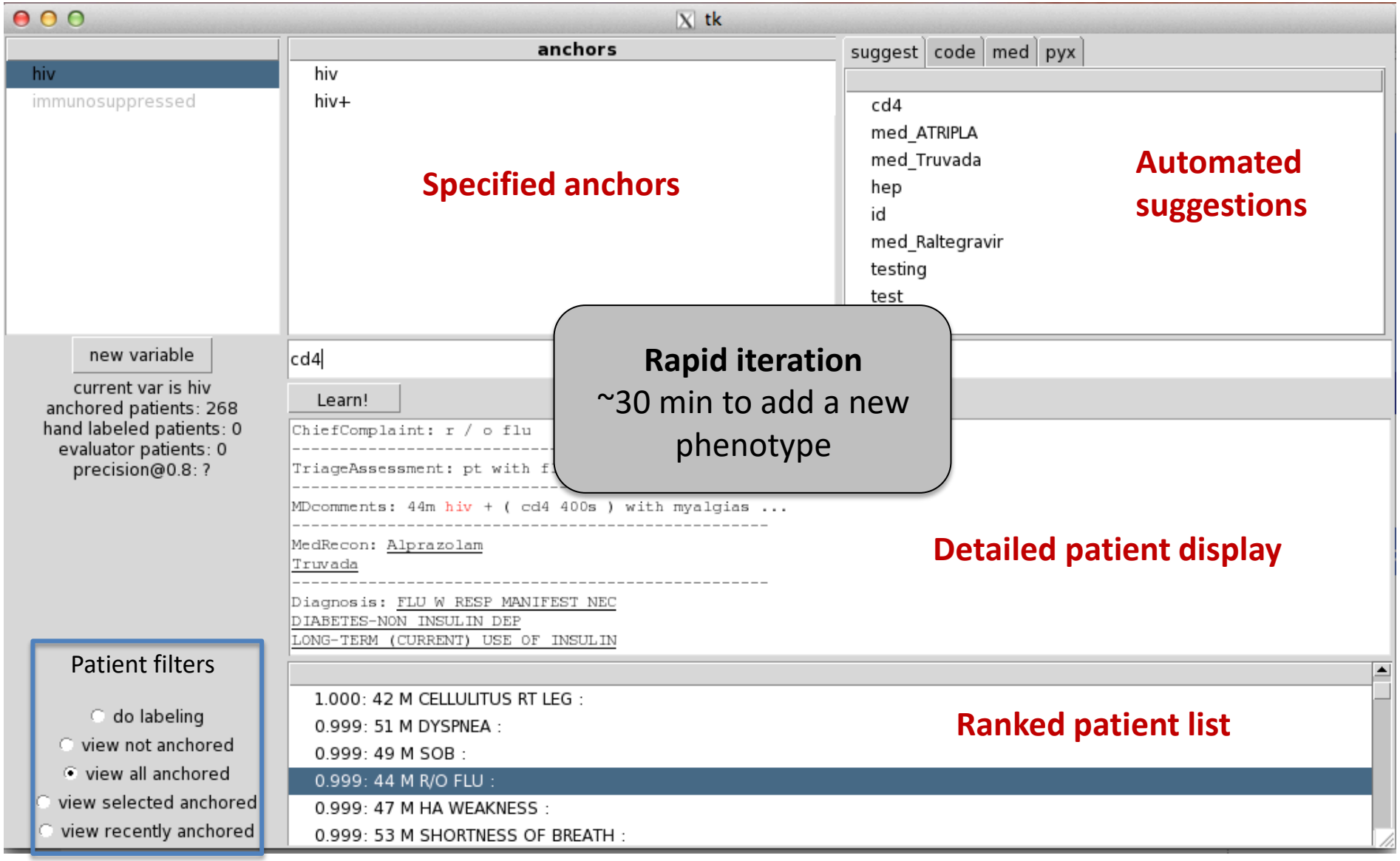
Evaluating phenotypes

- Derived anchors and learned phenotypes using 270,000 patients' emergency department medical records

History	Acute		
Alcoholism	Abdominal pain	Deep vein thrombosis	Laceration
Anticoagulated	Allergic reaction	Employee exposure	Motor vehicle accident
Asthma/COPD	Ankle fracture	Epistaxis	Pancreatitis
Cancer	Back pain	Gastroenteritis	Pneumonia
Congestive heart failure	Bicycle accident	Gastrointestinal bleed	Psych
Diabetes	Cardiac etiology	Geriatric fall	Obstruction
HIV+	Cellulitis	Headache	Septic shock
Immunosuppressed	Chest pain	Hematuria	Severe sepsis
Liver malfunction	Cholecystitis	Intracerebral hemorrhage	Sexual assault
	Cerebrovascular accident	Infection	Suicidal ideation
		Kidney stone	Syncope
			Urinary tract infection

Then used in real-time to predict labels in a hospital setting
Evaluate using ground-truth data

[Halpern, Horng, Choi, Sontag, AMIA '14]
[Halpern, Horng, Choi, Sontag, JAMIA '16]

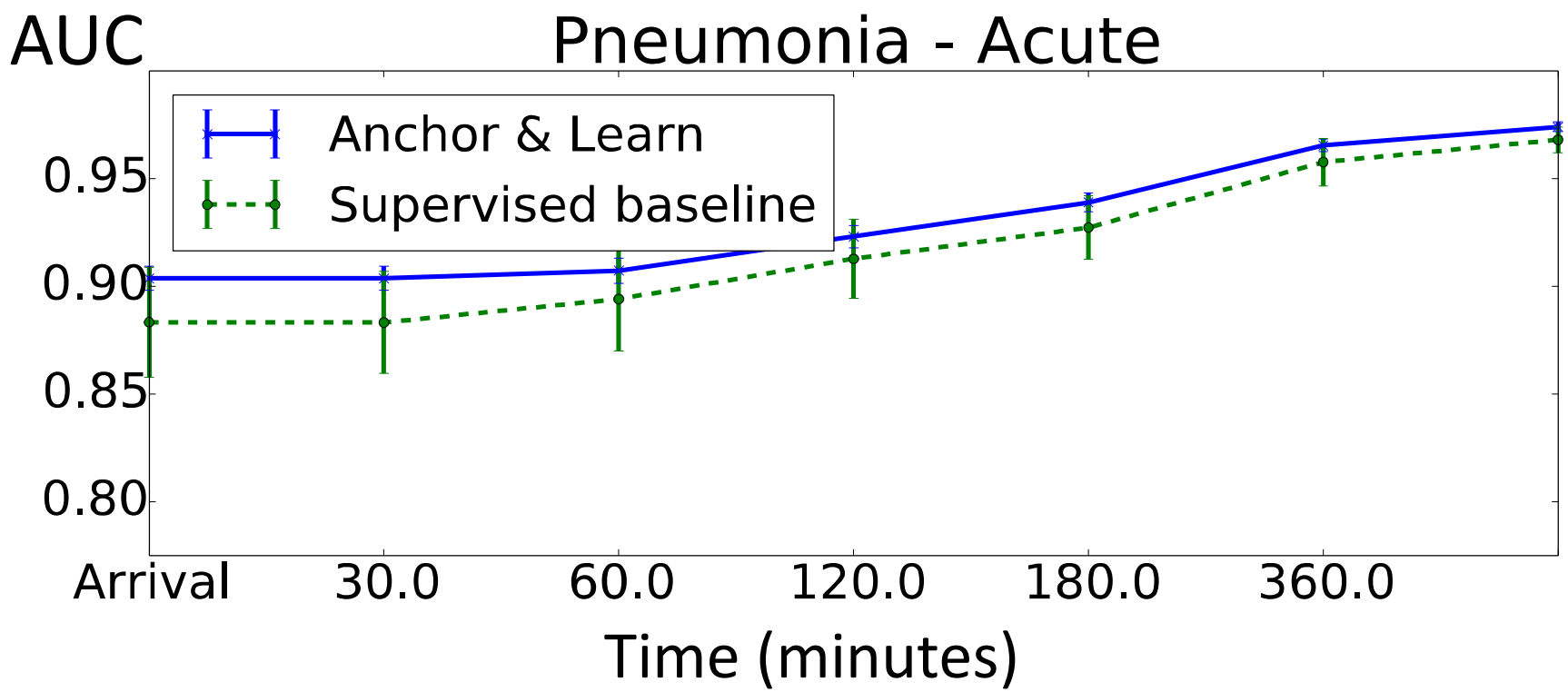

tk
hiv
immunosuppressed
anchors
hiv
hiv+
Specified anchors
suggest
code
med
pyx
cd4
med_ATRIPLA
med_Truvada
hep
id
med_Raltegravir
testing
test
Automated suggestions
new variable
current var is hiv
anchored patients: 268
hand labeled patients: 0
evaluator patients: 0
precision@0.8: ?
cd4
Learn!
Rapid iteration
~30 min to add a new phenotype
ChiefComplaint: r / o flu
TriageAssessment: pt with f
MDcomments: 44m hiv + (cd4 400s) with myalgias ...
MedRecon: Alprazolam
Truvada
Diagnosis: FLU W RESP MANIFEST NEC
DIABETES-NON INSULIN DEP
LONG-TERM (CURRENT) USE OF INSULIN
Detailed patient display
Patient filters
do labeling
view not anchored
view all anchored
view selected anchored
view recently anchored
1.000: 42 M CELLULITUS RT LEG :
0.999: 51 M DYSPNEA :
0.999: 49 M SOB :
0.999: 44 M R/O FLU :
0.999: 47 M HA WEAKNESS :
0.999: 53 M SHORTNESS OF BREATH :
Ranked patient list

Evaluating phenotypes

- Derived anchors and learned phenotypes using 270,000 patients' emergency department medical records
- To obtain ground truth, added a small number of questions to patient discharge procedure, rotated randomly

[Halpern, Horng, Choi, Sontag, AMIA '14]
[Halpern, Horng, Choi, Sontag, JAMIA '16]

Evaluating phenotypes



Comparison to supervised learning using labels for 5000 patients
Gets better over time, as predictions become more confident
Performance similar to supervised baseline (possibly even better?)

Evaluating phenotypes – example model (cardiac etiology)

Anchors

ICD9 codes
410.* acute MI
411.* other acute …
413.* angina pectoris
785.51 card. shock

Pyxis
coron. vasodilators
loop diuretic

Highly weighted terms

Ages
age=80-90
age=70-80
age=90+

Medications
lasix
furosemide

Sex=M

Pyxis
aspirin
clopidogrel
Heparin Sodium
Metoprolol Tartrate
Morphine Sulfate
Integrilin
Labetalol

nstemi
stemi
ntg
lasix
nitro

cp
chest pain
edema
cmed
chf exacerbation
sob
pedal edema

Unstructured text

[Halpern, Horng, Choi, Sontag, AMIA '14]
[Halpern, Horng, Choi, Sontag, JAMIA '16]

Evaluating phenotypes – example model (cardiac etiology)

Anchors

ICD9 codes
410.* acute MI
411.* other acute …
413.* angina pectoris
785.51 card. shock

Pyxis
coron. vasodilators

cardiac medicine
BIDMC shortform

Highly weighted terms

Ages
age=80-90
age=70-80
age=90+

Medications
lasix
furosemide

Sex=M

Pyxis
aspirin
clopidogrel
Heparin Sodium
Metoprolol
Tartrate
Morphine Sulfate
Integrilin
Labetalol

nstemi
stemi
ntg
lasix
nitro

cp
chest pain
edema
cmed
chf exacerbation
sob
pedal edema

Unstructured text

[Halpern, Horng, Choi, Sontag, AMIA '14]
[Halpern, Horng, Choi, Sontag, JAMIA '16]

Outline for today's class

1. Learning with noisy labels
   - Consistent estimation under class-conditional noise (Natarajan et al., NeurIPS '13)
   - Application in health care (Halpern et al., JAMIA '16)
2. **Weak supervision**

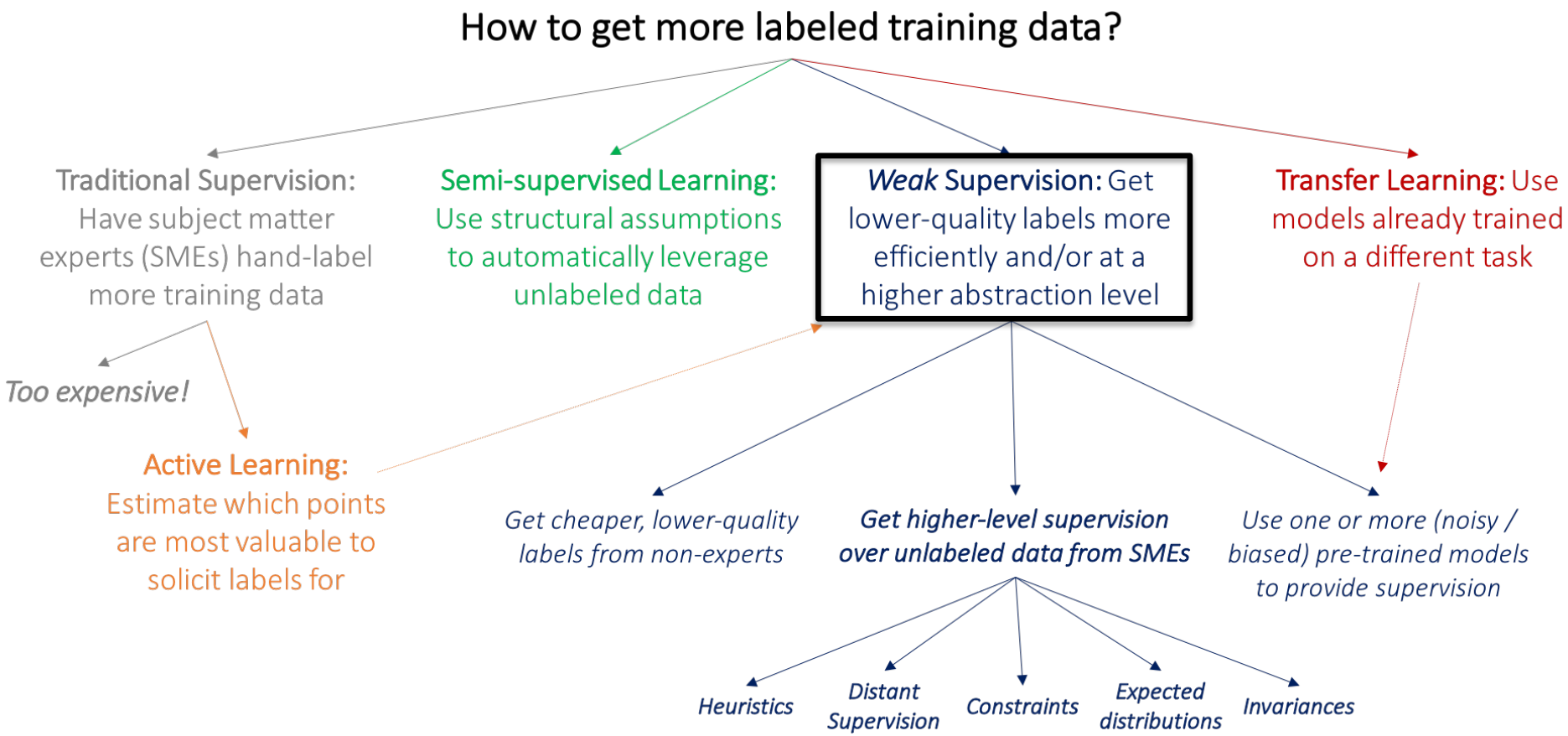
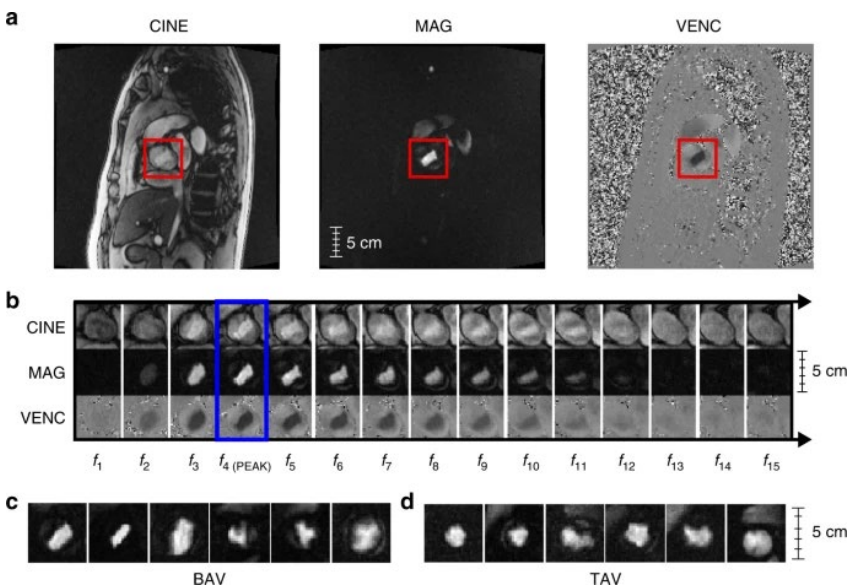


Figure from: https://www.snorkel.org/blog/weak-supervision

Weak supervision

- Define one or more labeling functions l(x) that outputs a label (or no label) for each example
- E.g., for sentiment analysis
  “good” -> +1
  “bad” -> -1
- Reconcile conflicting labels; ignore data points that are unlabeled
- Learn a model on the labeled data points

Classifying Aortic Valve Malformations



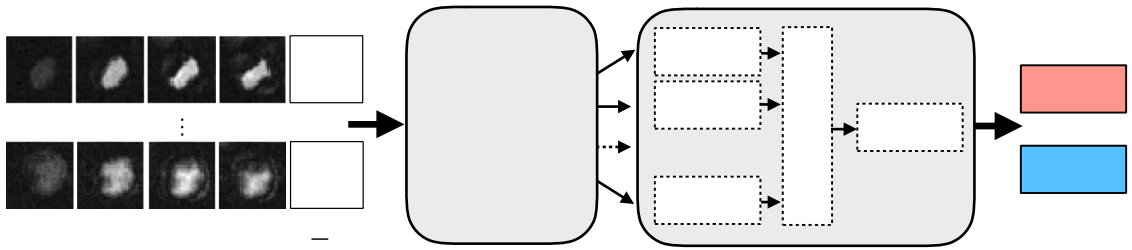
- Data: MRI sequences for 14,328 subjects from the UK Biobank
- True gold standard labels for *aortic valve malformations* (BAV) derived for 412 subjects
- Goal: Train a model which can classify BAV (positive or negative) when given a new MRI sequence

[Fries et al. *Weakly supervised classification of aortic valve malformations using unlabeled cardiac MRI sequences.* Nature Communications 2019]

Classifying Aortic Valve Malformations

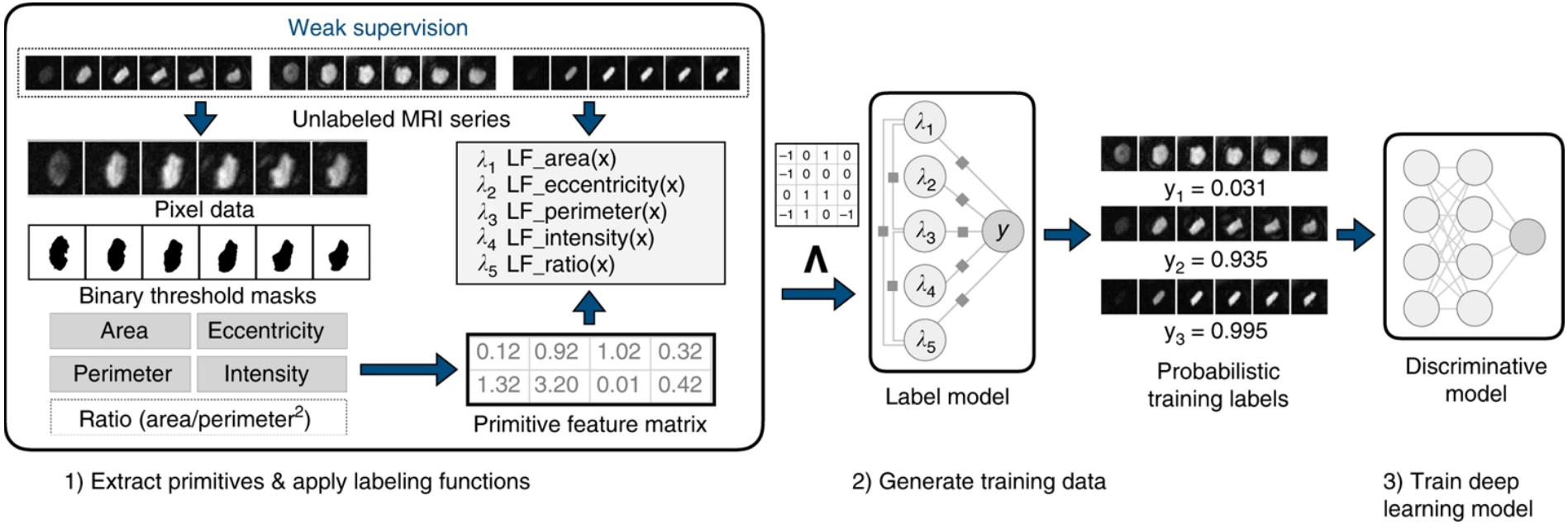
Methodology:

1. Train a factor graph-based model to predict noisy labels for all unlabeled examples
2. Train a hybrid convolutional NN / LSTM using the derived noisy labels



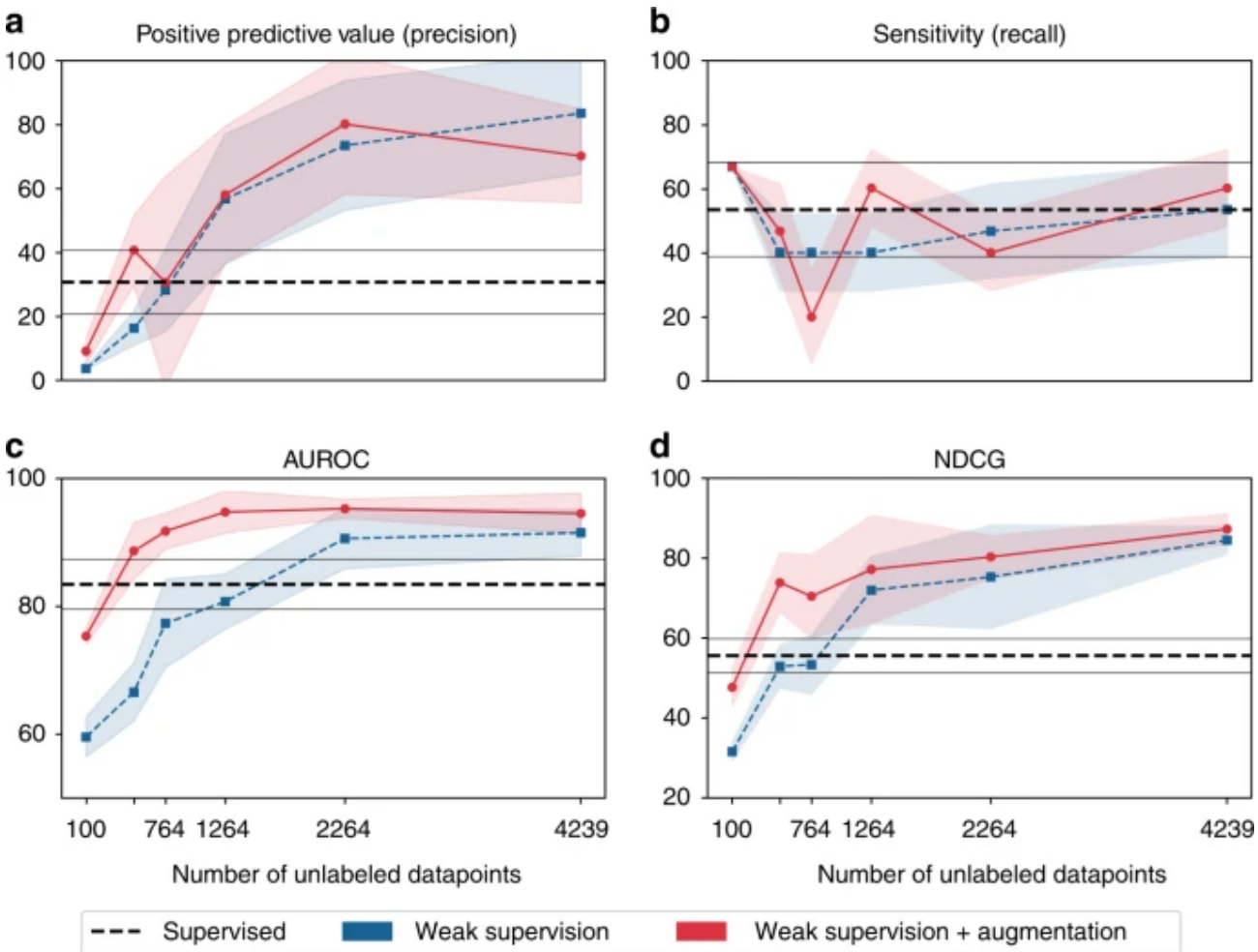
[Fries et al. *Weakly supervised classification of aortic valve malformations using unlabeled cardiac MRI sequences.* Nature Communications 2019]

Classifying Aortic Valve Malformations



[Fries et al. *Weakly supervised classification of aortic valve malformations using unlabeled cardiac MRI sequences.* Nature Communications 2019]

Classifying Aortic Valve Malformations



[Fries et al. *Weakly supervised classification of aortic valve malformations using unlabeled cardiac MRI sequences.* Nature Communications 2019]

Weak supervision for text classification

- Example labeling functions:

```
# Setting LF output values
ABSTAIN_VAL = 0
HEMORRHAGE_VAL = 1
NO_HEMORRHAGE_VAL = -1

def LF_positive_hematoma(report):
    """
    Checking for words indicating hematoma
    """
    r1 = re.compile('(No|without|resolution|scalp|subgaleal)\\s([\S]*\\s){0,10}(hematoma)',re.IGNORECASE)
    r = re.compile('hematoma',re.IGNORECASE)
    for s in report.report.sentences:
        if r.search(s.text) and (not r1.search(s.text)):
            return HEMORRHAGE_VAL
    return ABSTAIN_VAL

def LF_hemorrhage_hi_cover(report):
    """
    Checking for both hemorrhage and hematoma
    """
    if LF_positive_hemorrhage(report) == 0 and LF_positive_hematoma(report) == 0:
        return NO_HEMORRHAGE_VAL
    return HEMORRHAGE_VAL
```

Reference: https://github.com/HazyResearch/cross-modal-ws-demo/blob/master/lfs/lfs_hct.py

Weak supervision for text classification

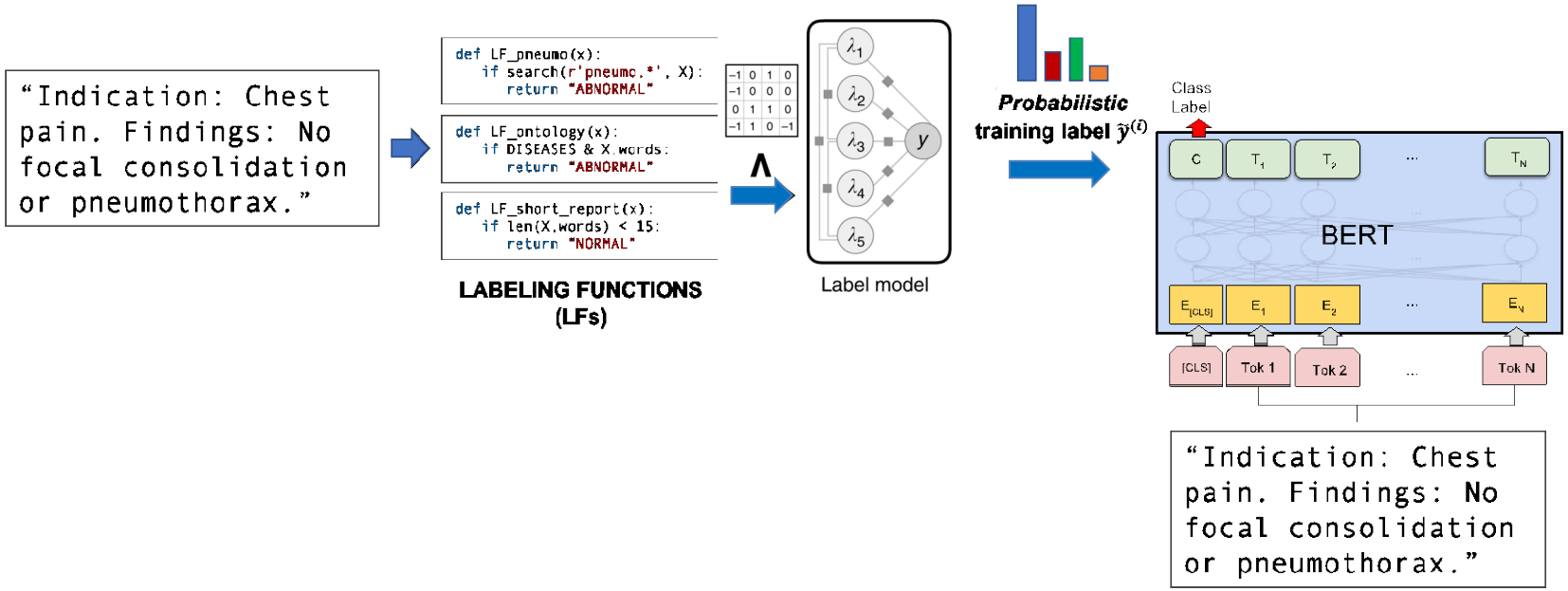
- Example labeling functions:

```
# Setting LF output values
ABSTAIN_VAL = 0
HEMORRHAGE_VAL = 1
NO_HEMORRHAGE_VAL = -1

def LF_normal_V01(report):
    """
    Checking for specific normal phrase
    """
    r = re.compile('Normal CT of the Head',re.IGNORECASE)
    for s in report.report.sentences:
        if r.search(s.text):
            return NO_HEMORRHAGE_VAL
    return ABSTAIN_VAL

def LF_normal_V02(report):
    """
    Checking for specific normal phrase
    """
    r = re.compile('No acute intracranial abnormality',re.IGNORECASE)
    for s in report.report.sentences:
        if r.search(s.text):
            return NO_HEMORRHAGE_VAL
    return ABSTAIN_VAL
```

Reference: https://github.com/HazyResearch/cross-modal-ws-demo/blob/master/lfs/lfs_hct.py

Weak supervision for text classification

- Use BERT as “end model”



- Why does this not simply learn to reproduce the labeling functions?

Weak supervision for text classification

Table 1: Statistics of all the tasks, domains and datasets included in WRENCH.

Task (#)	Domain (#)	Dataset (#)	#Label	#LF	Train #Data	Dev #Data	Test #Data
Sentiment Class.	Movie	IMDb [61, 79]	2	5	20,000	2,500	2,500
	Review	Yelp [107, 79]	2	8	30,400	3,800	3,800
Spam Class.	Review	Youtube [1]	2	10	1,586	120	250
	Text Message	SMS [2, 3]	2	73	4,571	500	500
Topic Class.	News	AGNews [107, 79]	4	9	96,000	12,000	12,000
Question Class.	Web Query	TREC [49, 3]	6	68	4,965	500	500
Relation Class.	News	Spouse [11, 77]	2	9	22,254	2,811	2,701
	Biomedical	CDR [13, 77]	2	33	8,430	920	4,673
	Web Text	SemEval [31, 109]	9	164	1,749	200	692
	Chemical	ChemProt [41, 102]	10	26	12,861	1,607	1,607

[Zhang et al. WRENCH: A Comprehensive Benchmark for Weak Supervision. NeurIPS Track on Datasets and Benchmarks, 2021]

Weak supervision for text classification

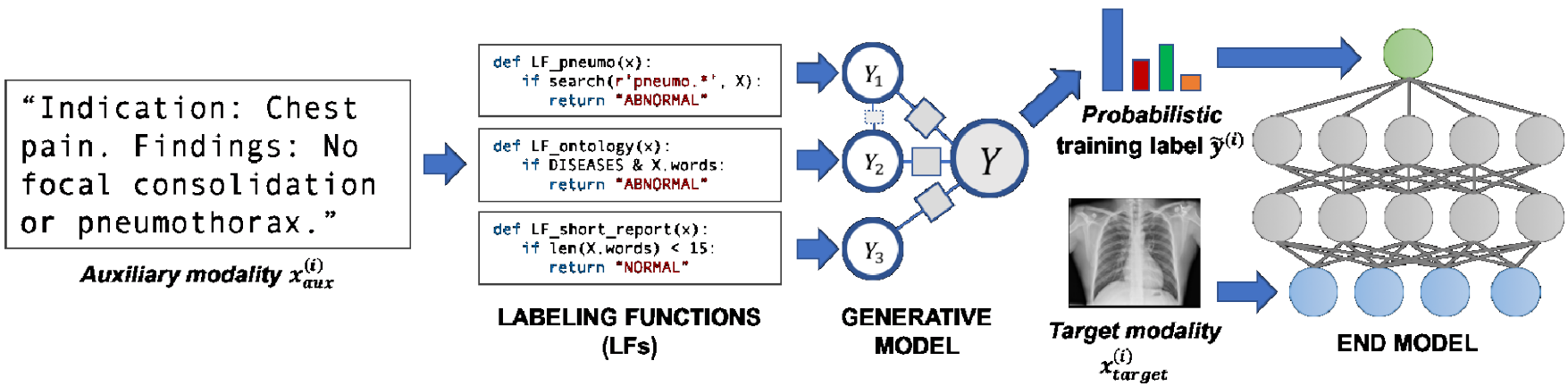
EM: end model (R=RoBERTa, RC=COSINE-RoBERTa, BC=COSINE-BERT)
LM: label model (MV="majority vote", WMC="weighted majority vote")

		Best Gold		Top 1			Top 2			Top 3		
Dataset	**Metric**	**EM**	**Value**	**EM**	**LM**	**Value**	**EM**	**LM**	**Value**	**EM**	**LM**	**Value**
IMDb	Acc.	R	93.25	RC	MeTaL	88.86	RC	FS	88.48	RC	MV	88.48
Yelp	Acc.	R	97.13	RC	FS	95.45	RC	FS	95.33	RC	DS	95.01
Youtube	Acc.	B	97.52	BC	MV	98.00	RC	MV	97.60	RC	MV	97.60
SMS	F1	B	96.96	RC	WMV	98.02	RC	MeTaL	97.71	RC	WMV	97.27
AGNews	Acc.	R	91.39	RC	DS	88.20	RC	MV	88.15	RC	WMV	88.11
TREC	Acc.	R	96.68	RC	DP	82.36	RC	MeTaL	79.84	BC	DP	78.72
Spouse	F1	–	–	BC	FS	56.52	–	MeTaL	46.62	RC	MV	46.28
CDR	F1	R	65.86	–	MeTaL	69.61	–	DP	63.51	RC	DP	61.40
SemEval	Acc.	B	95.43	BC	DP	88.77	BC	MV	86.80	RC	DP	86.73
ChemProt	Acc.	B	89.76	BC	DP	61.56	RC	MV	59.43	RC	MV	59.32

[Zhang et al. WRENCH: A Comprehensive Benchmark for Weak Supervision. NeurIPS Track on Datasets and Benchmarks, 2021]

Weak supervision with multiple views

- Alternatively, one could just use the noisy labels from the label model to directly train the downstream model:



- Co-training (Blum & Mitchell, '98) can be used to improve performance further

[Dunnmon et al., Cross-Modal Data Programming Enables Rapid Medical Machine Learning. arXiv:1903.1101, 2019.]

Conclusion

- Can be difficult to get labeled data for machine learning in health care
- Often possible to quickly derive *noisy* labels (i.e., anchors or labeling functions)
- With conditionally independent noise, ML as usual can be used (with recalibration)
  - $x \perp \tilde{Y} | Y$ (noise rate constant for all examples)
  - Can sometimes *censor* the features to make this assumption more realistic (the anchor & learn method)
  - Alternatively, use pretrained representations