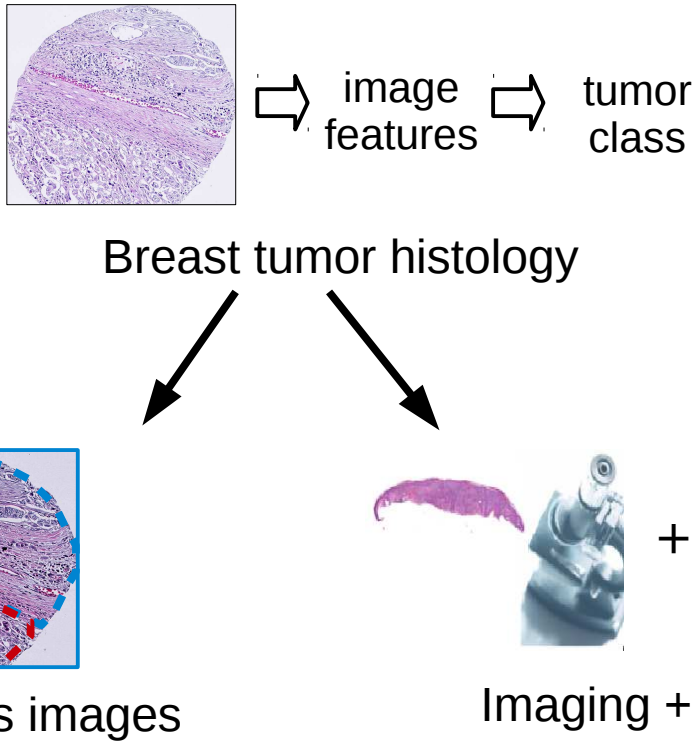
Discriminative Representations for Heterogeneous Images and Multimodal Data

Heather Couture

Dissertation Defense

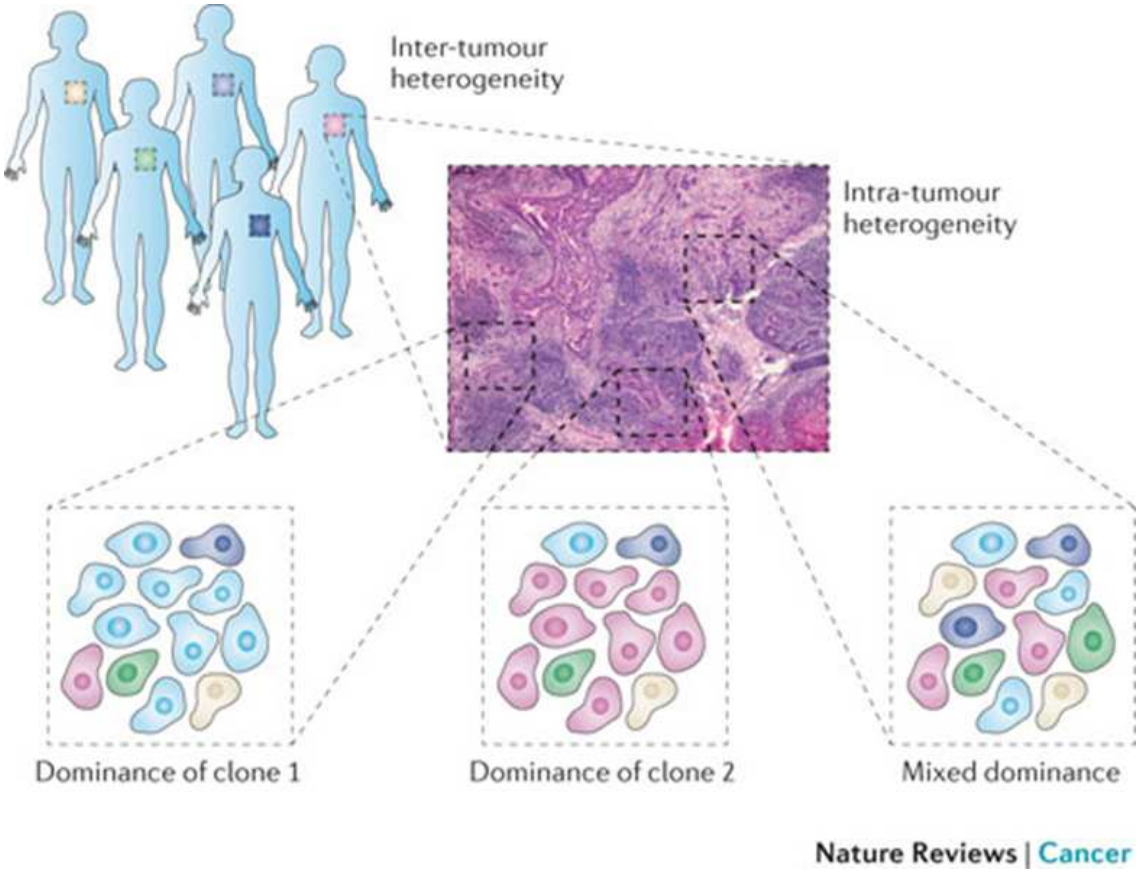
Nov. 19, 2018



Cancer Heterogeneity

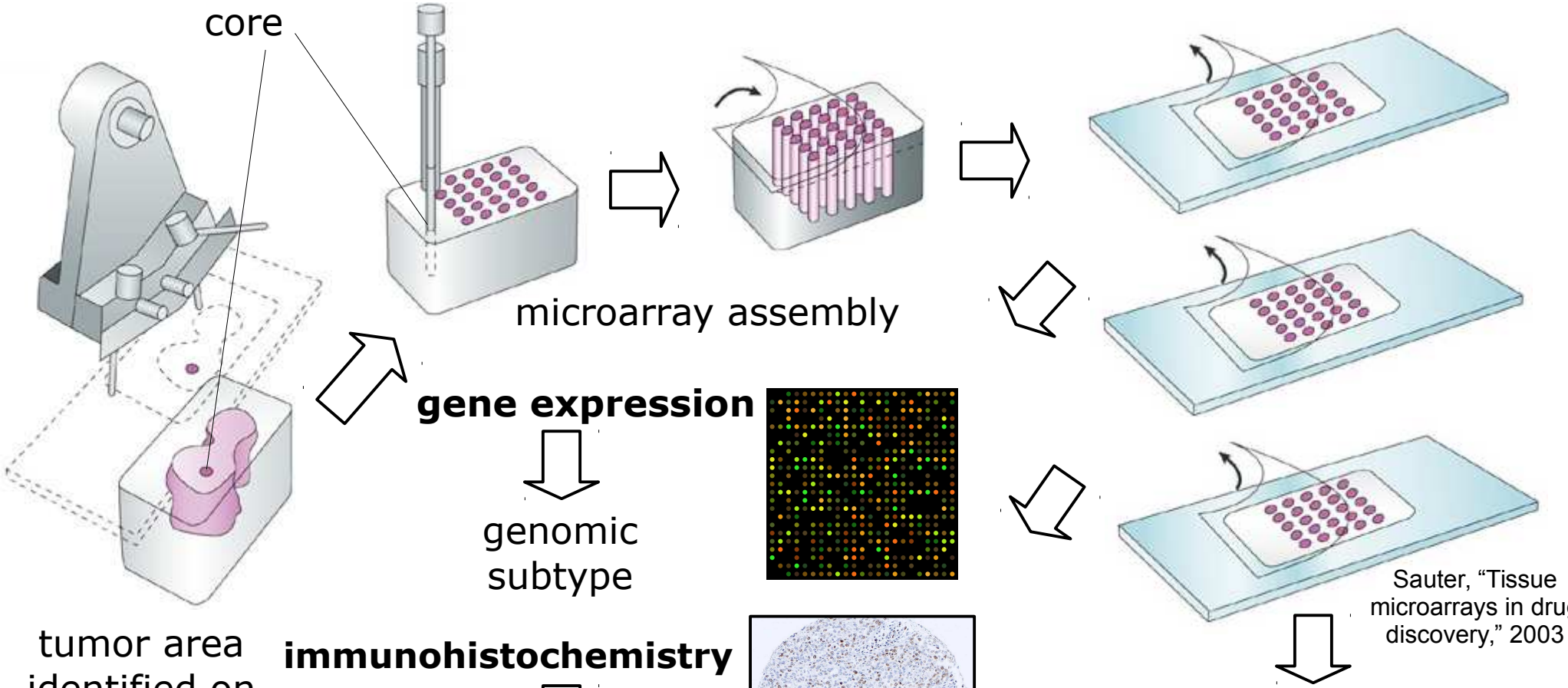
Inter-tumor heterogeneity: subtypes with distinctive
- morphology
- genomics
- clinical presentations
- responses to treatment
- outcomes

Intra-tumor heterogeneity: challenge for diagnosis and treatment



Manusyk, “Intra-tumour heterogeneity: a looking glass for cancer?”, 2012

Tissue Microarray

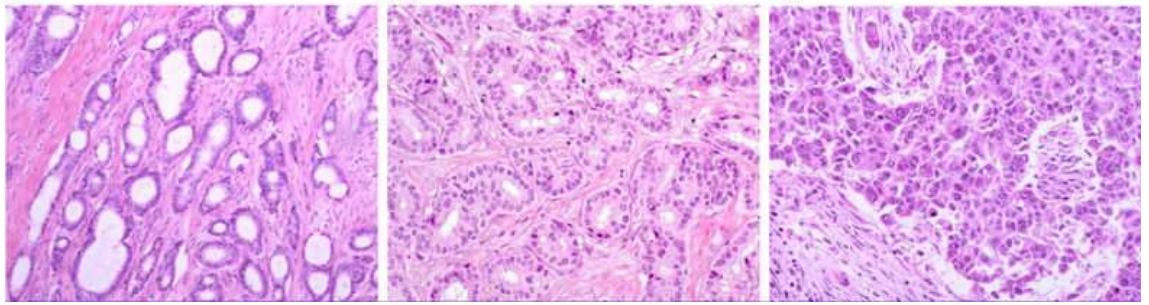


Motivations

Prognosis

Assess tumor aggressiveness:

Grade 1 Grade 2 Grade 3



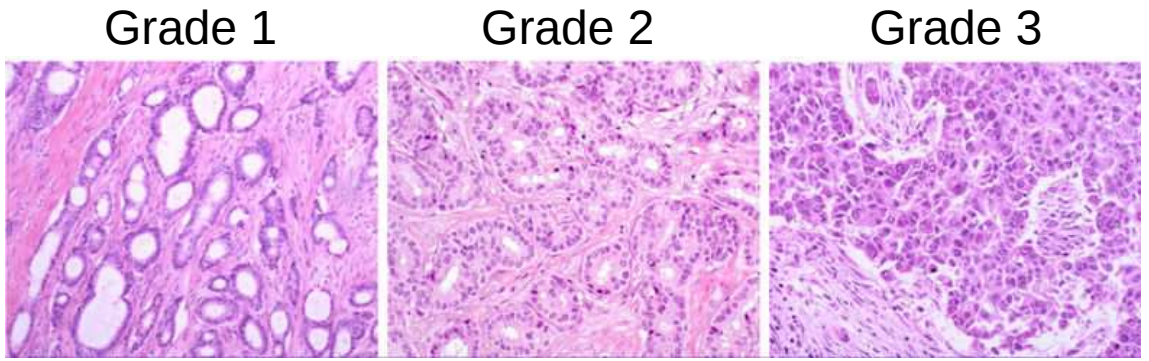
Rakha, 2010

Intermediate grade
highly variable
amongst pathologists

Motivations

Prognosis

Assess tumor aggressiveness:

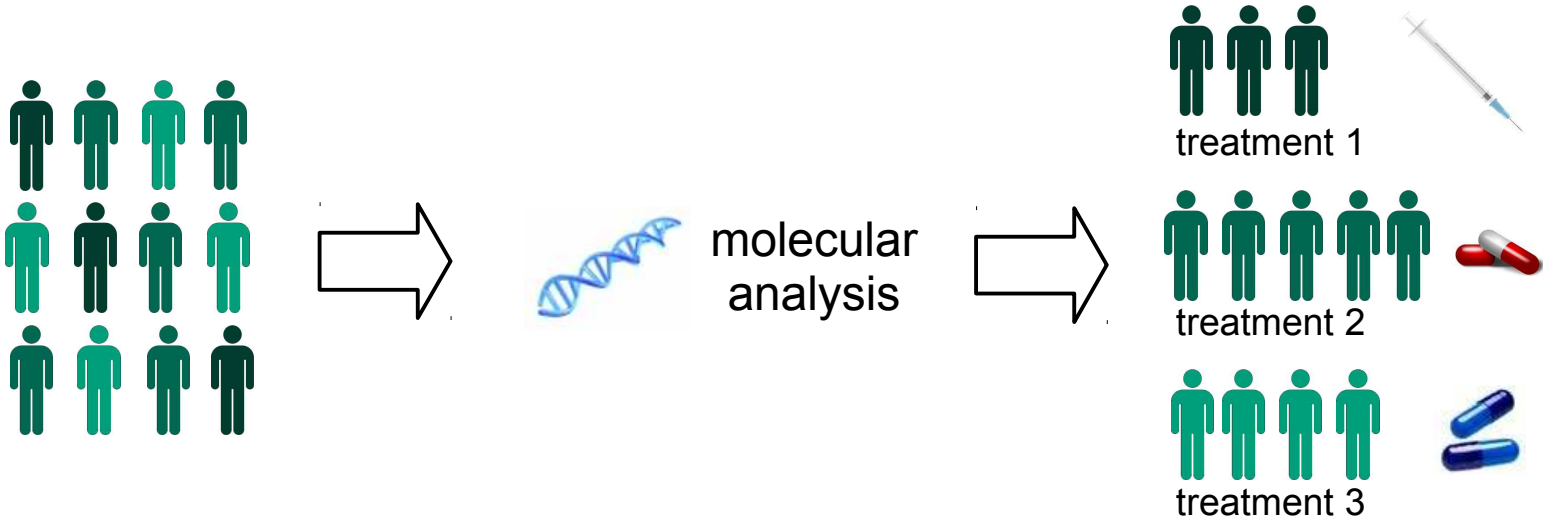


Rakha, 2010

Intermediate grade
highly variable
amongst pathologists

Personalized treatment

Target tumors based on molecular analysis:



Costly

Not routinely
performed

Thesis Statement

Learned representations *for histology images of tissue can capture both* ***intra-*** *and* ***inter-tumor heterogeneity****, enabling* ***discriminative*** *models for tumor properties. Combining these image features with data from other modalities such as* ***genomics*** *in a* ***task-driven*** *model can provide insight into the shared tumor properties and further* ***improve predictions****. These computational techniques using* ***discriminative*** *features can provide a* ***lower cost and more repeatable*** *alternative to molecular methods and insight into tumor* ***heterogeneity****.*

Contributions to Breast Cancer Research

1) Methods to capture **biologically-relevant** features by operating on the H&E stain intensities extracted from histology images
2) A **low cost and repeatable** method for predicting **histological**, **molecular**, and **genomic properties** of tumors from H&E histology
3) A mechanism to find predicted **tumor heterogeneity** from H&E histology

Contributions to Computer Science
will be discussed at the end

Outline

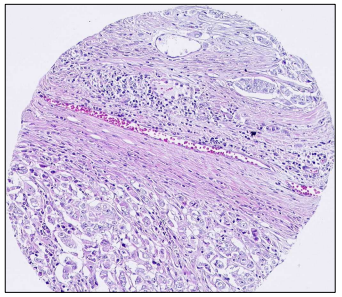
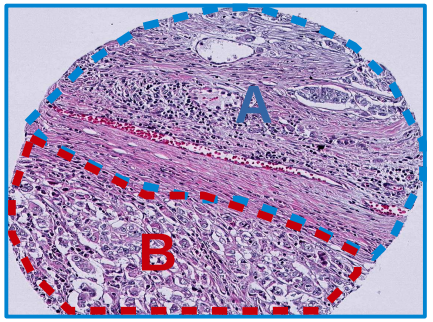


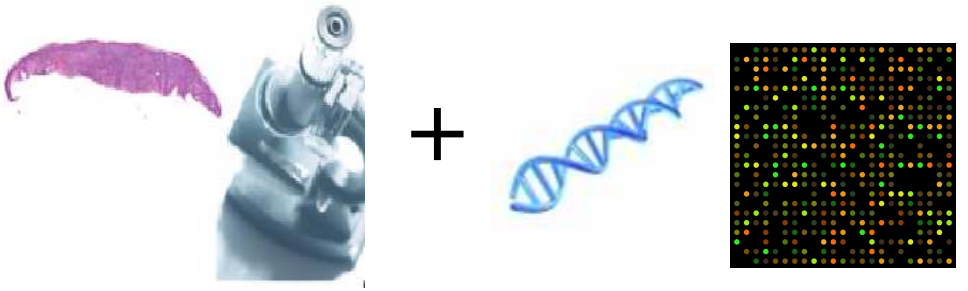
image features

tumor class

1) Representing histology images

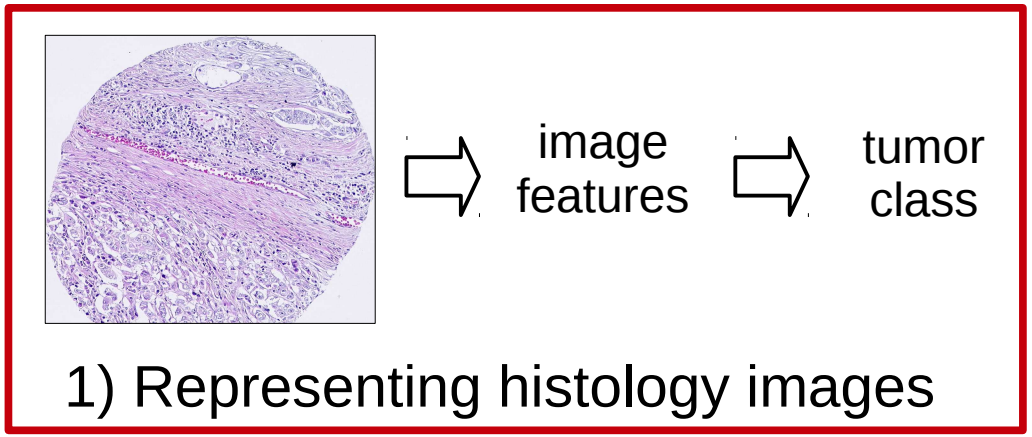


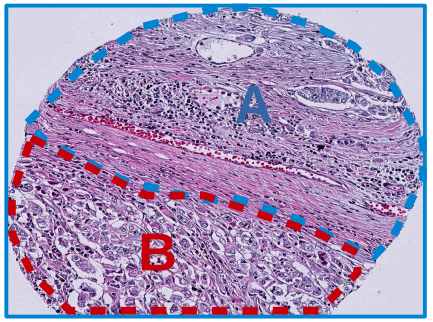
2) Handling heterogeneous images



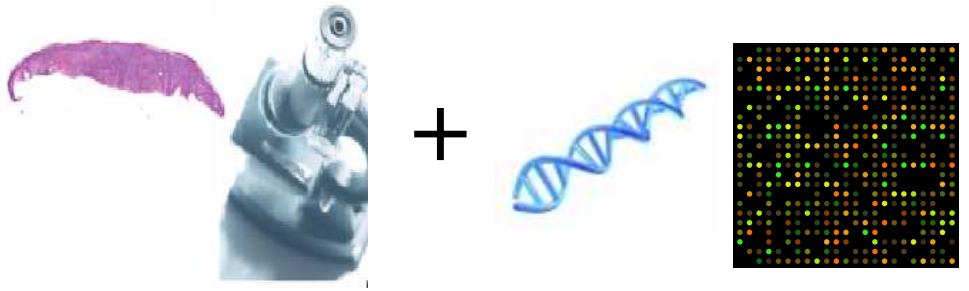
3) Combining imaging & genomics

Outline





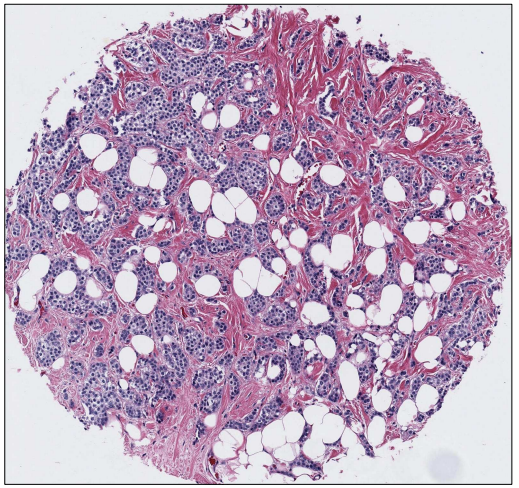
2) Handling heterogeneous images



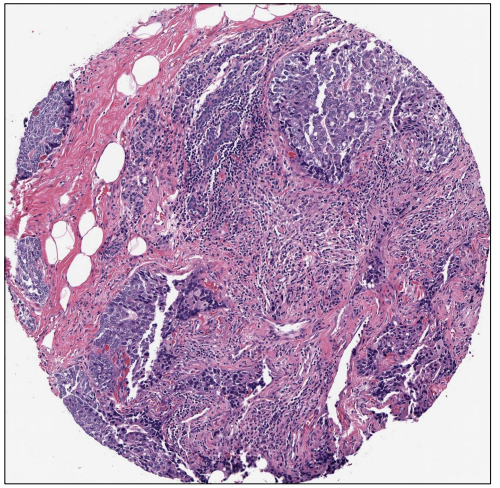
3) Combining imaging & genomics

Problem Definition: Classification

Infer tumor class from histology



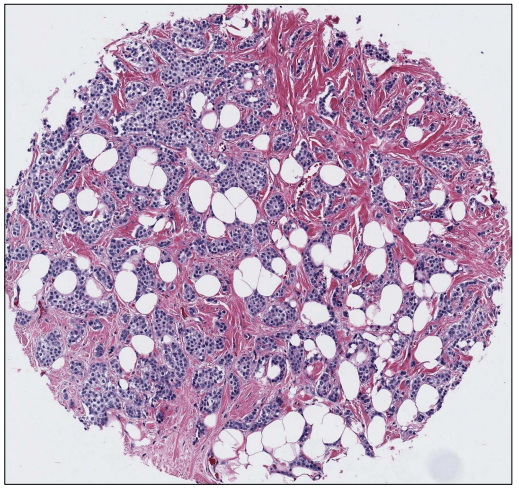
vs



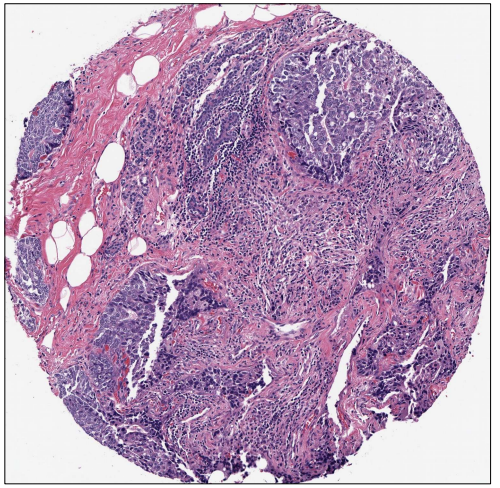
	Class 1	**Class 2**
Grade:	low	high
Histologic subtype:	lobular	ductal
Estrogen receptor status:	positive	negative
Genomic subtype:	non-Basal	Basal
Risk of recurrence (ROR-PT):	low	high

Problem Definition: Classification

Infer tumor class from histology



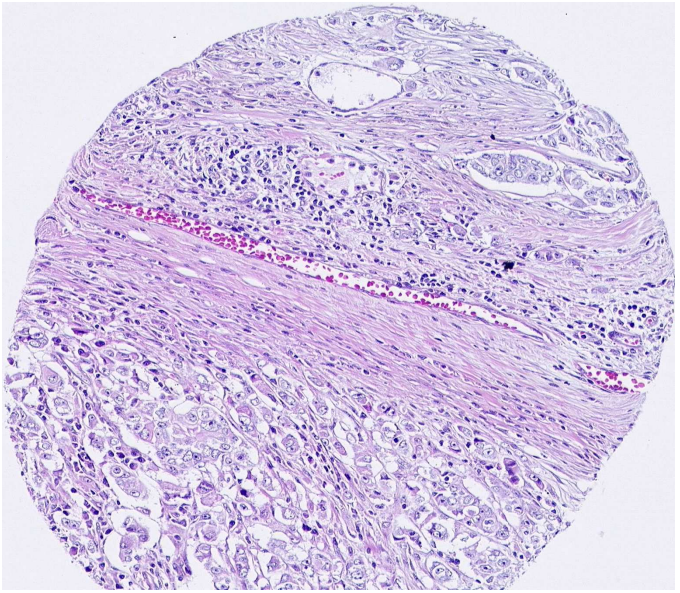
vs



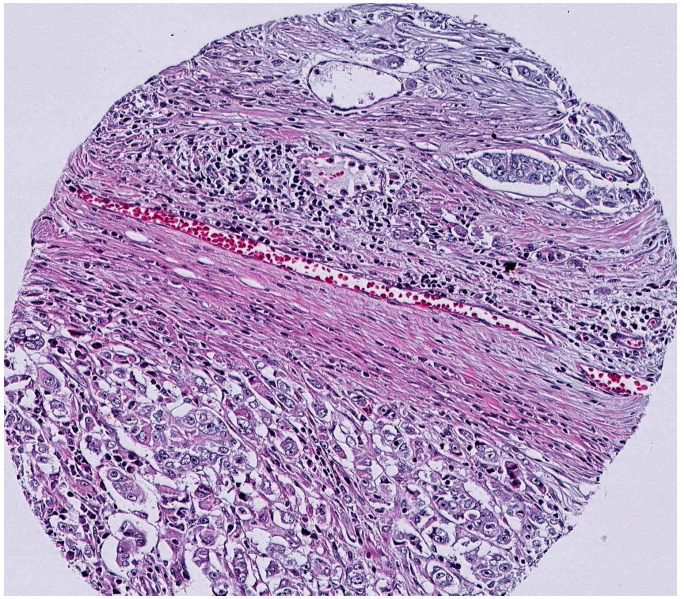
	Class 1	Class 2	Ground truth
Grade:	low	high	pathologist
Histologic subtype:	lobular	ductal	
Estrogen receptor status:	positive	negative	immunohistochemistry
Genomic subtype:	non-Basal	Basal	gene expression
Risk of recurrence (ROR-PT):	low	high	combination

Stain Normalization & Extraction

Original image

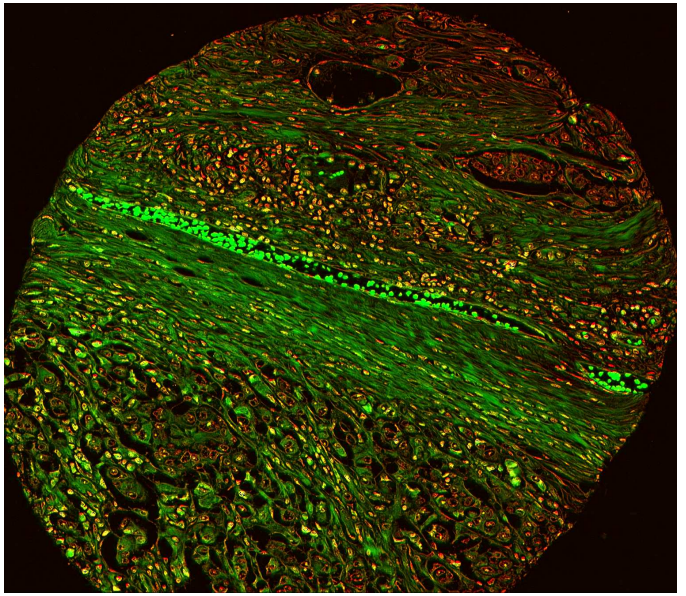


Stain normalized



Hematoxylin: blue
Eosin: pink

Stain channels

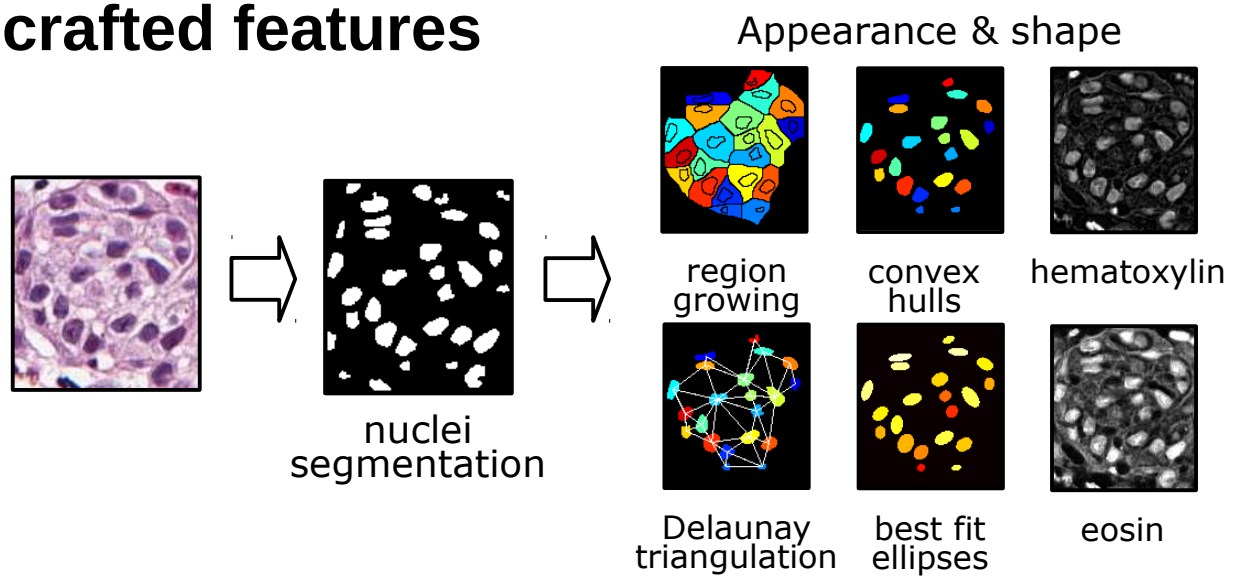


Hematoxylin: red
Eosin: green
Residual: blue

Niethammer, 2010

Background

Hand-crafted features

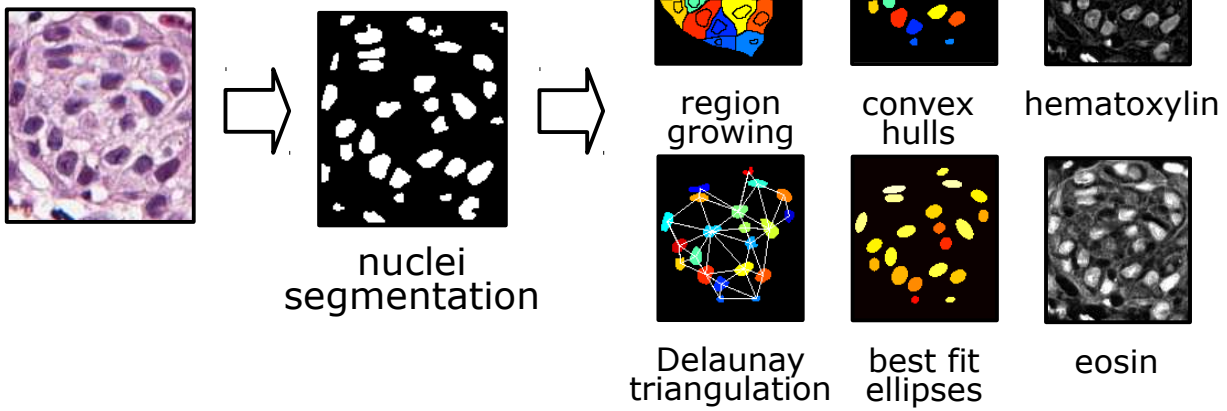


Focused on cell by cell morphology

Difficult to adapt to new data sets

Background

Hand-crafted features

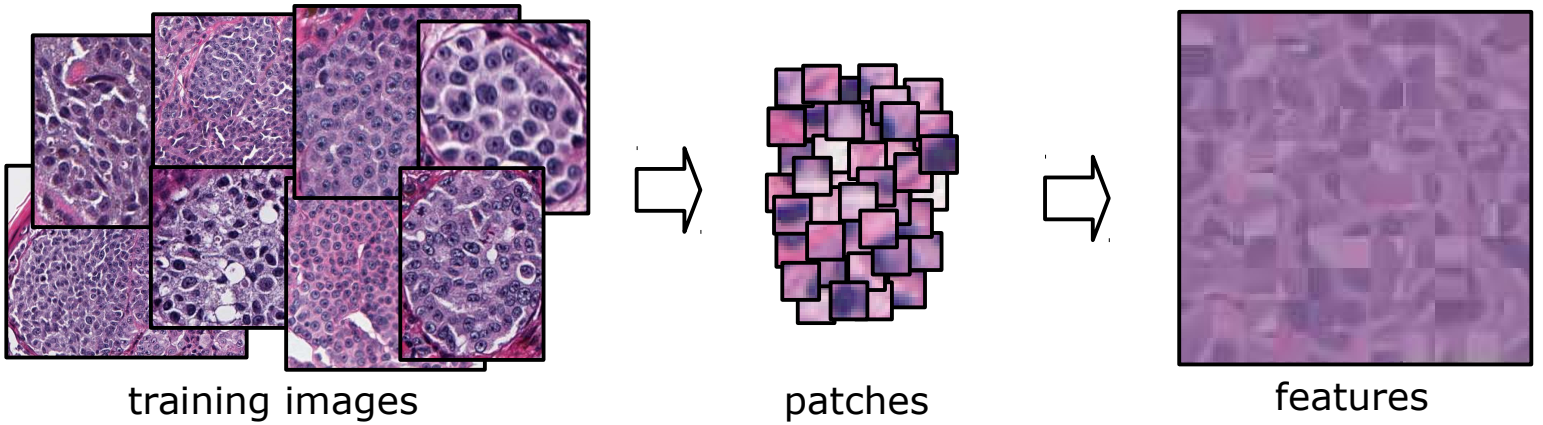


Focused on cell by cell morphology

Difficult to adapt to new data sets

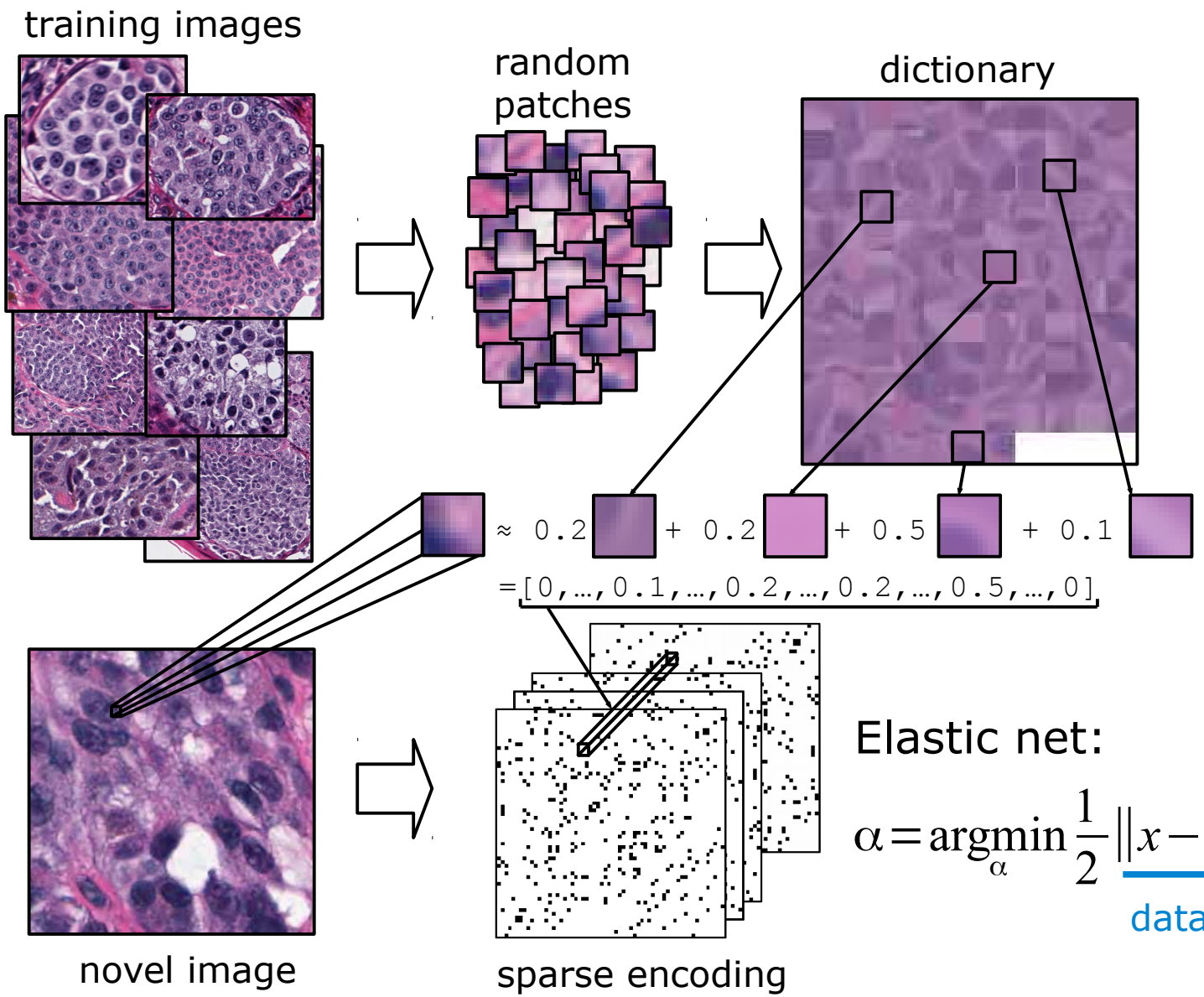
Learned features

Dictionary learning
Deep learning



Adapted to given images

Method 1: Task-driven Dictionary Learning



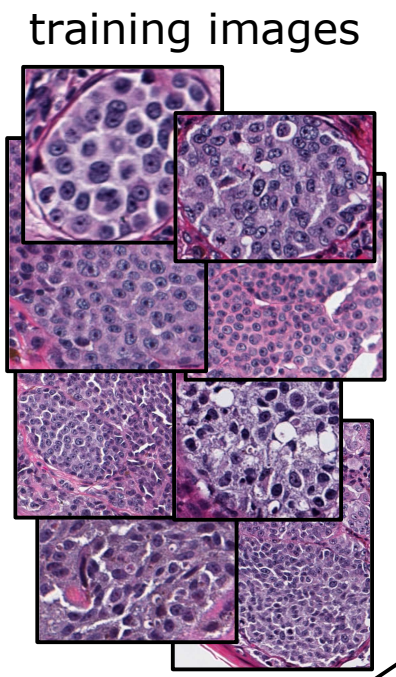
Elastic net:

$$\alpha = \arg\min_{\alpha} \frac{1}{2} \|x - D\alpha\|_2^2 + \lambda_1 \|\alpha\|_1 + \lambda_2 \|\alpha\|_2^2$$

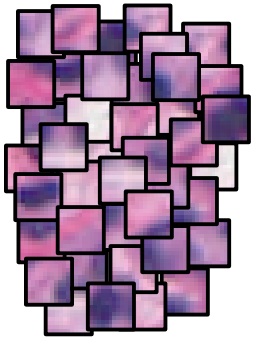
data fitting

sparsity-inducing regularization

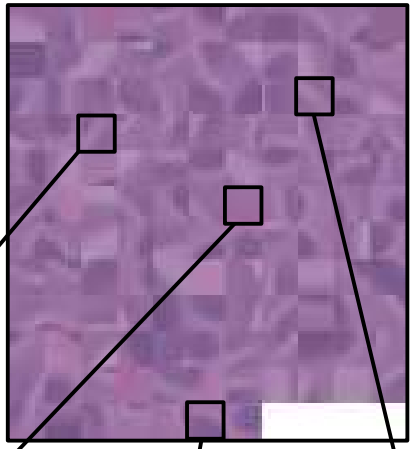
Method 1: Task-driven Dictionary Learning



training images



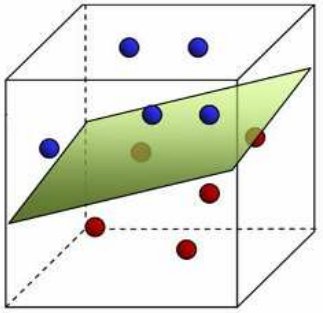
random patches



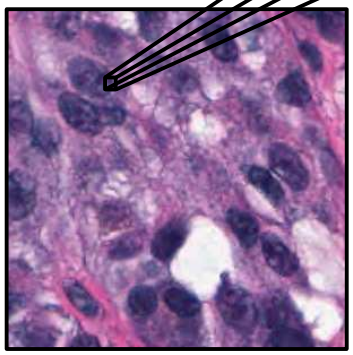
dictionary

update classifier based on current dictionary

update dictionary to improve classification



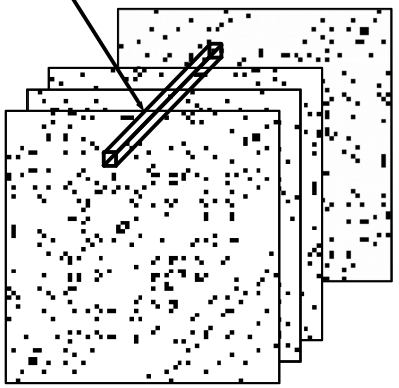
classifier



novel image

$\approx 0.2 + 0.2 + 0.5 + 0.1$

$=[0,\ldots,0.1,\ldots,0.2,\ldots,0.2,\ldots,0.5,\ldots,0]$



sparse encoding

task-driven extension
(Mairal, 2012)

Apply to patches then aggregate patch predictions to image prediction

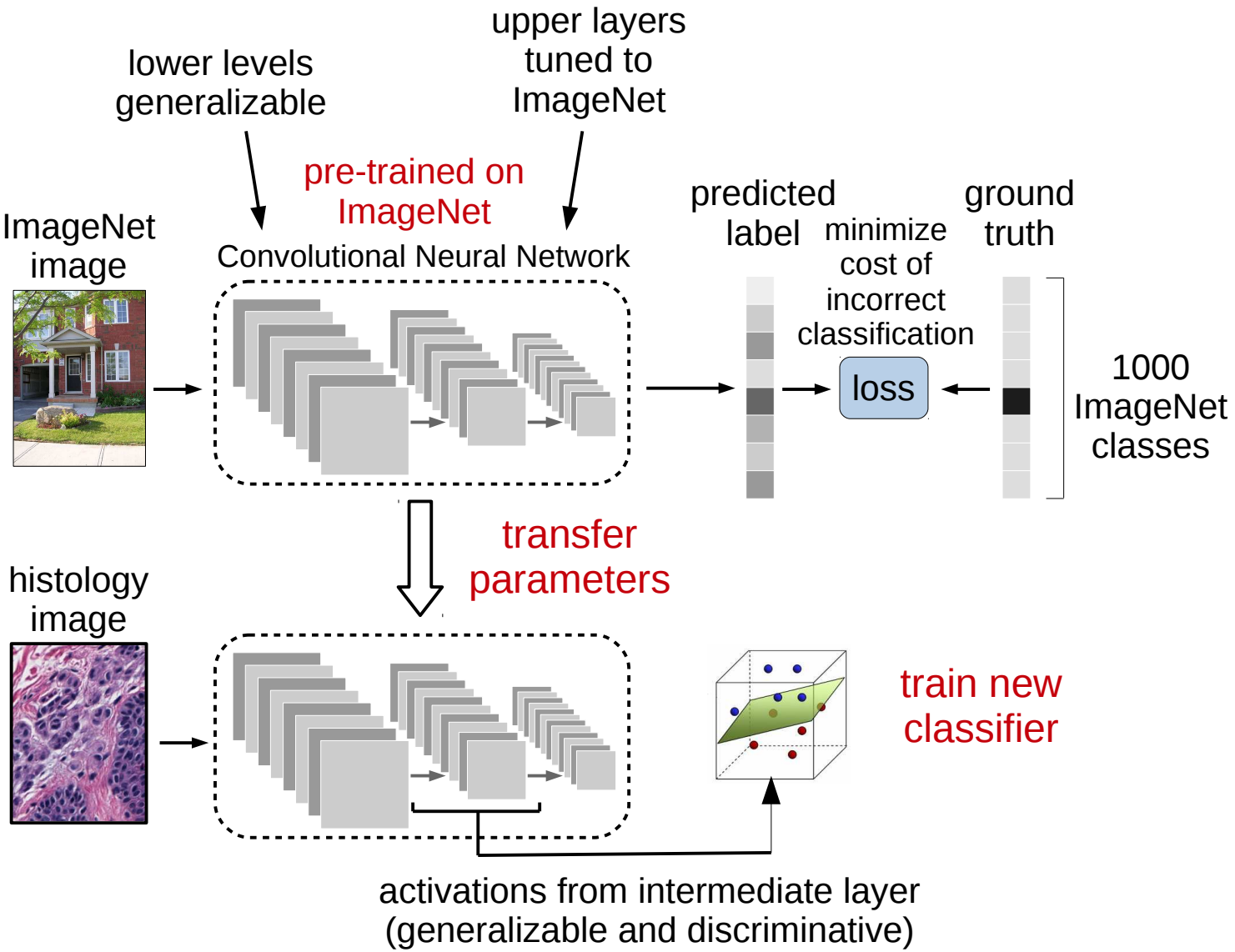
Elastic net:

$$\alpha = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2} \|x - D\alpha\|_2^2 + \lambda_1 \|\alpha\|_1 + \lambda_2 \|\alpha\|_2^2$$

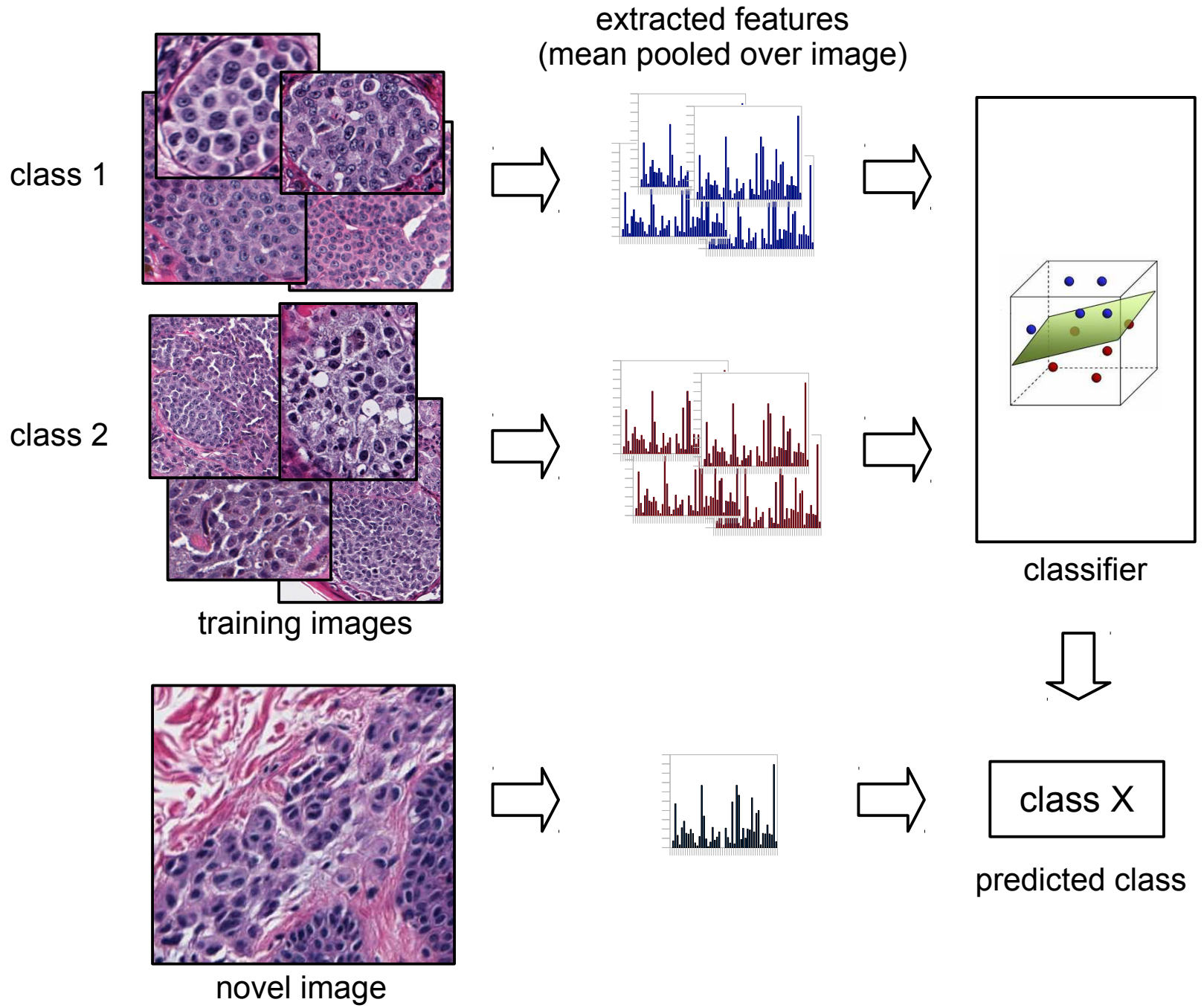
data fitting

sparsity-inducing regularization

Method 2: Transferred Deep Features



Method 1 & 2: Classification



Feature & Classifier Comparison

AUC

Method	Log. reg.	Linear SVM	RBF SVM	DWD
Hand-crafted features	0.789 (0.032)	0.778 (0.027)	0.573 (0.040)	0.728 (0.022)
Dictionary learning (nuclei)	0.812 (0.020)	0.794 (0.017)	0.661 (0.045)	0.755 (0.035)
Dictionary learning (dense)	0.845 (0.020)	**0.855** (0.024)	0.631 (0.035)	0.799 (0.023)
Deep transfer learning	0.832 (0.029)	0.825 (0.032)	0.716 (0.039)	0.811 (0.034)

SPECS data set:
43 Basal, 42 Luminal A
2 cores/patient
5-fold cross-validation

Dictionary learning:
Dictionary size 256
Patch size 17x17
Nuclei-centered patches
vs. dense patches

Deep transfer learning:
AlexNet, conv4

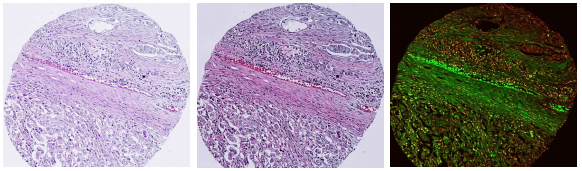
Best results with dictionary learning

Deep transfer learning is promising

Fine-tuning could improve further

Dictionary & Deep Learning Comparison

Original RGB | Normalized RGB | Stain channels



Input image	AUC			Accuracy		
	Basal	**ER**	**Grade**	**Basal**	**ER**	**Grade**
Dictionary						
Original RGB	0.810 (0.008)	0.843 (0.009)	0.905 (0.007)	0.790 (0.012)	0.797 (0.009)	0.828 (0.010)
Normalized RGB	0.817 (0.010)	0.850 (0.009)	0.911 (0.010)	0.780 (0.010)	0.790 (0.012)	0.821 (0.013)
Stain channels	**0.822** (0.014)	**0.860** (0.008)	**0.927** (0.009)	**0.795** (0.012)	**0.805** (0.009)	**0.848** (0.010)
Deep transfer						
Original RGB	0.784 (0.017)	0.819 (0.009)	0.906 (0.014)	0.775 (0.013)	0.767 (0.006)	0.819 (0.013)
Normalized RGB	**0.807** (0.013)	**0.857** (0.011)	0.928 (0.005)	**0.784** (0.014)	0.792 (0.010)	**0.848** (0.006)
Stain channels	0.785 (0.015)	0.851 (0.008)	**0.933** (0.009)	0.778 (0.015)	**0.798** (0.009)	0.842 (0.015)

CBCS data set: 1713 patient samples, 4 cores/patient
Linear SVM, 5-fold cross-validation

Stain normalization improves results

Deep transfer learning works on non-RGB images

Dictionary learning slightly better than deep transfer learning

Task-driven Dictionary Learning

Classification Accuracy

	Unsupervised	**Task-driven**
Patch-level	0.507	**0.520**
Patient-level		
Mean of patch probabilities	0.646	0.642
Sum of log of patch probabilities	0.729	0.664
Linear SVM on histogram of features	0.698	0.713

SPECS data set:
43 Basal, 42 Luminal A
2 cores/patient
5-fold cross-validation

Dictionary size 128
Patch size 9x9

Task-driven extension successful for patch-level accuracy

Less clear for patient-level accuracy

Patient-level labels are weak when applied to small patches

Outline

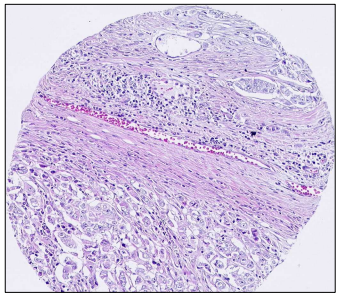
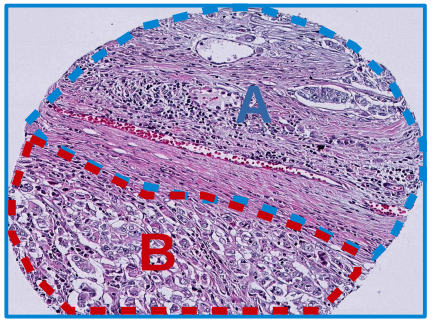


image features

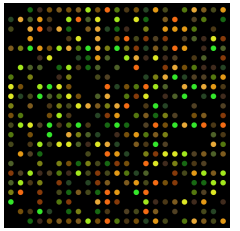
tumor class

1) Representing histology images



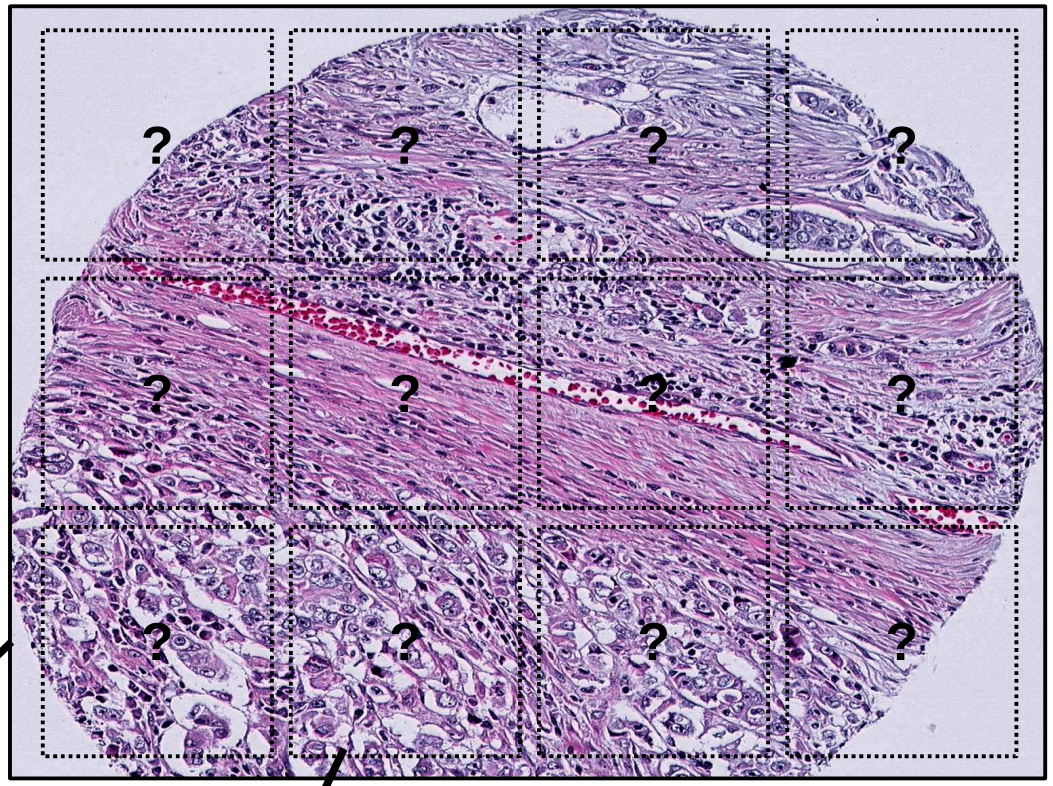
2) Handling heterogeneous images

+



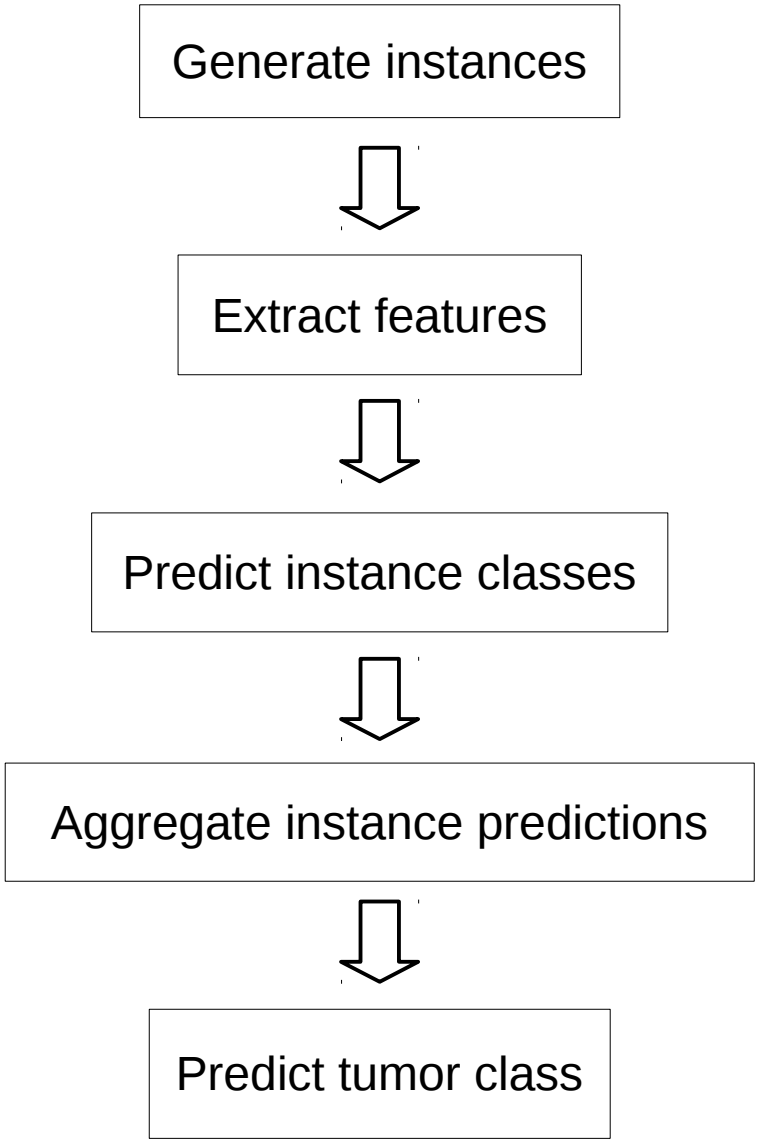
3) Combining imaging & genomics

Multiple Instance Learning



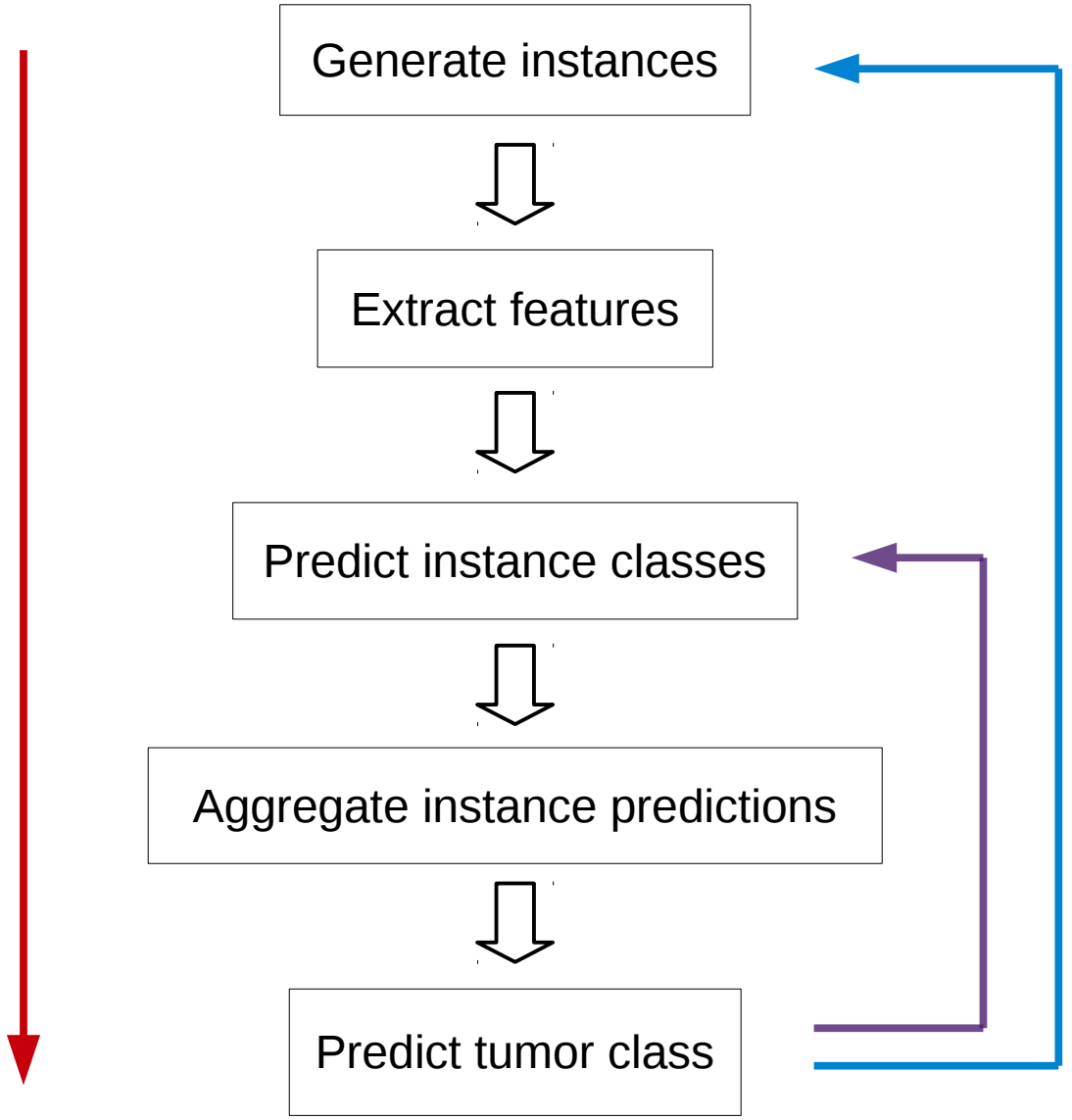
weakly supervised classification

Method Overview



Method Overview

Method 1:
Extract features
+
train classifier



Method 2:
Learn latent instance labels

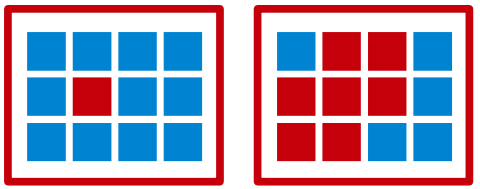
Method 3:
End-to-end fine-tuning of CNN

Multiple Instance Terminology

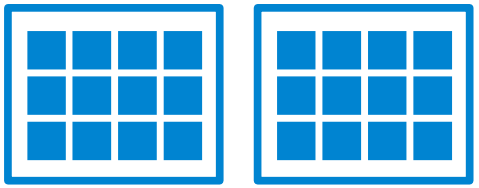
Standard assumption:

- Negative bag: all instances negative
- Positive bag: one or more instances positive

positive bags



negative bags



Good for diagnosis

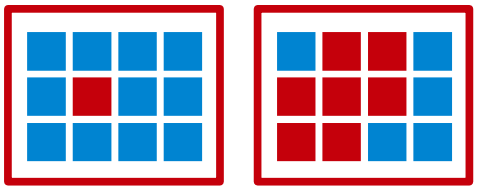
Classes not treated symmetrically

Multiple Instance Terminology

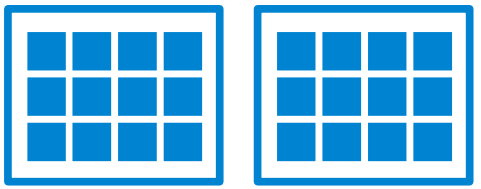
Standard assumption:

- Negative bag: all instances negative
- Positive bag: one or more instances positive

positive bags



negative bags

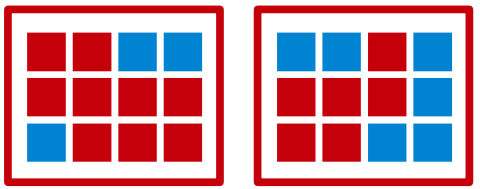


Good for diagnosis

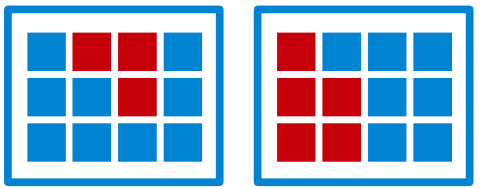
Classes not treated symmetrically

Alternative assumption: majority vote

class 1

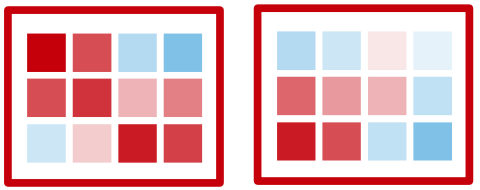


class 2

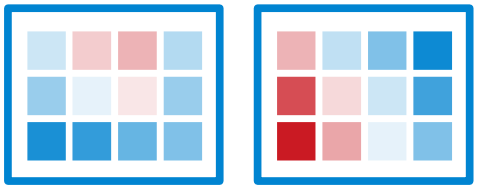


Remove assumption: learn how to aggregate probabilities

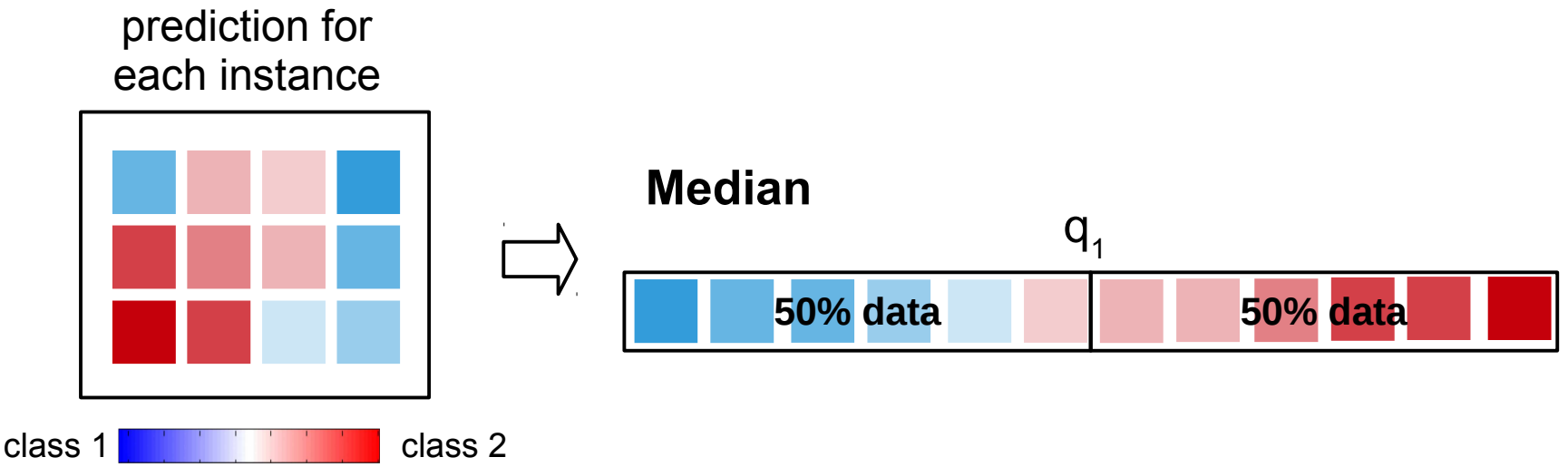
class 1



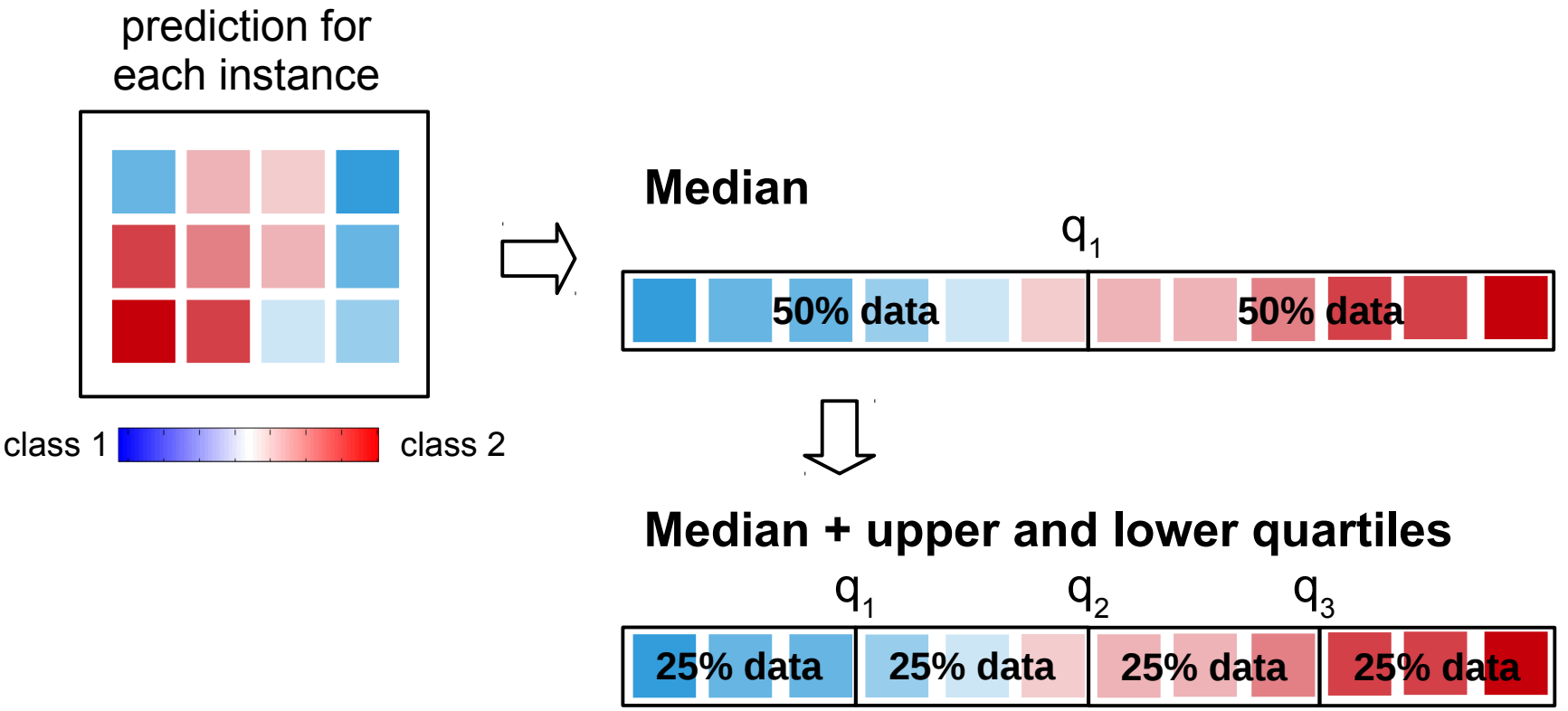
class 2



Method 1: Quantile Aggregation

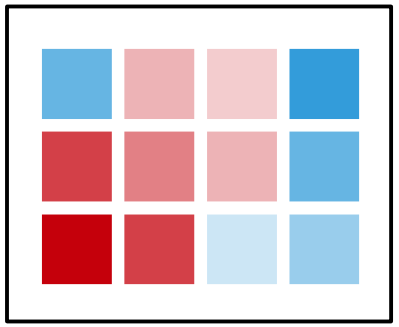


Method 1: Quantile Aggregation



Method 1: Quantile Aggregation

prediction for each instance



class 1 class 2

Median

q_1

50% data **50% data**

Median + upper and lower quartiles

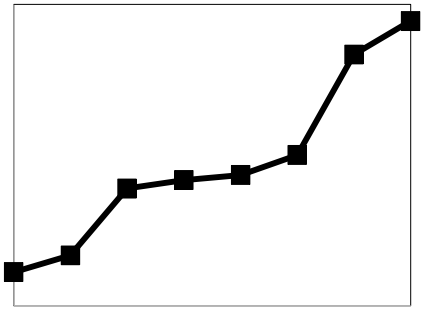
q_1 q_2 q_3

25% data **25% data** **25% data** **25% data**

Quantile function

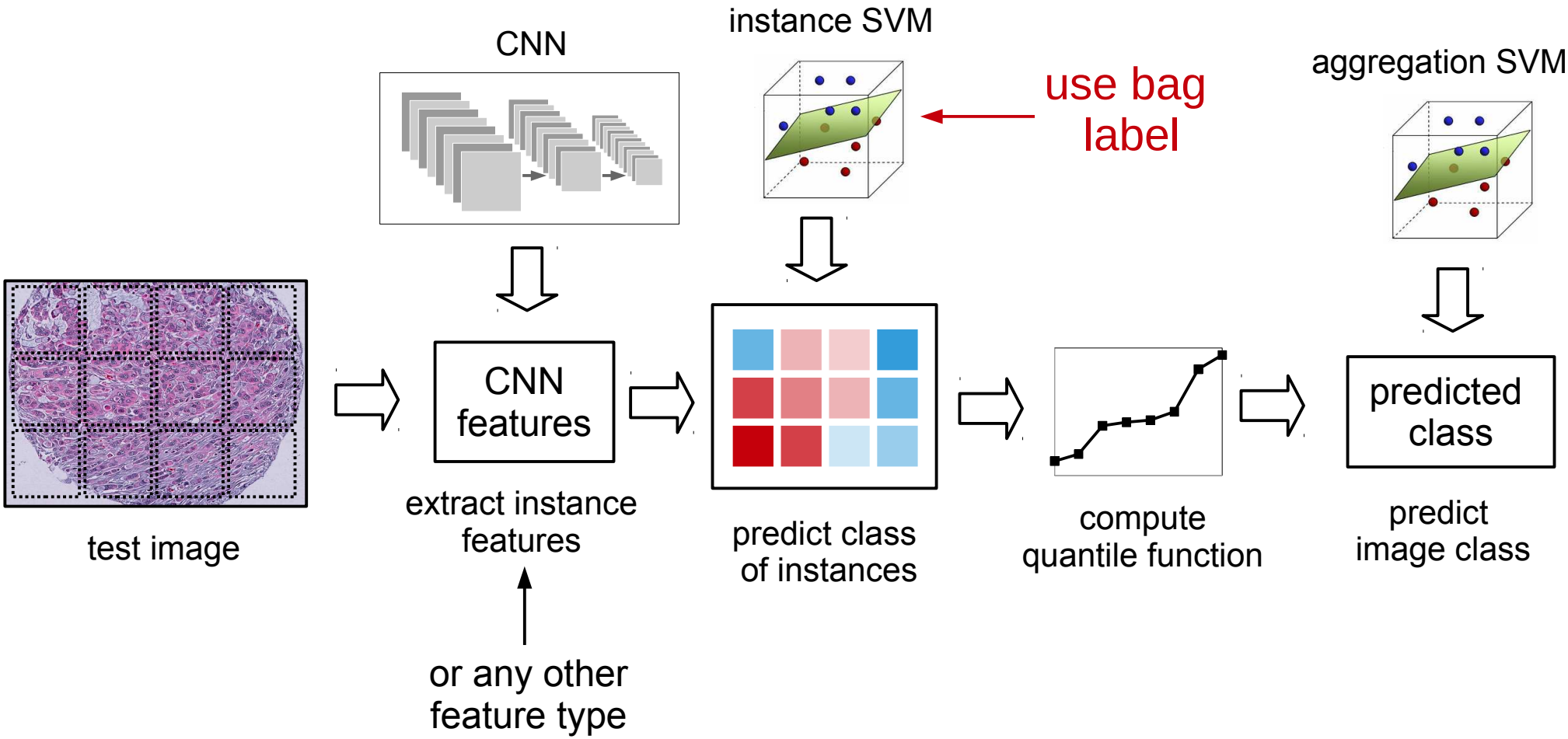
$Q = [\ q_1\ q_2\ \ldots\ q_N\ ]$

for N evenly spaced quantiles

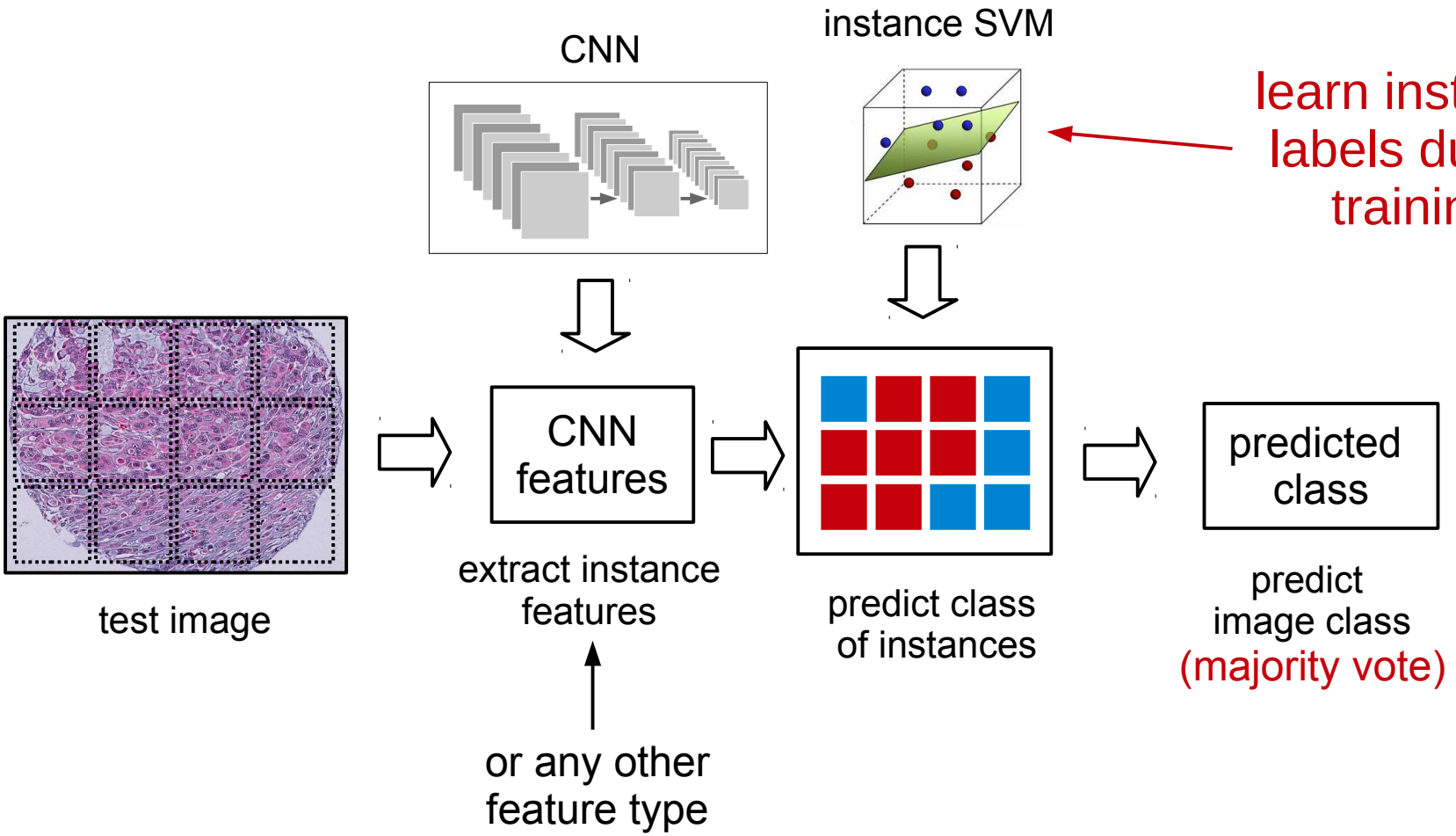


Method 1: Quantile Aggregation

Prediction



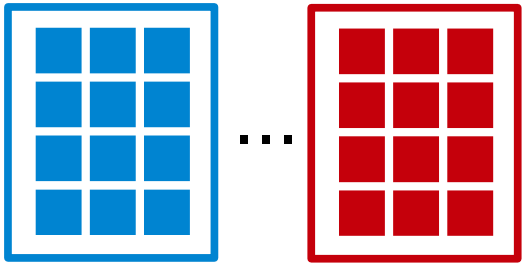
Method 2: Iterative MI with Majority Vote



Method 2: Iterative MI with Majority Vote

Learn Instance Labels

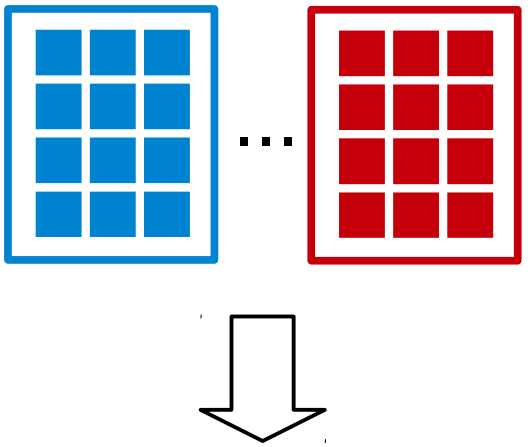
1) Initialize instance labels



Method 2: Iterative MI with Majority Vote

Learn Instance Labels

1) Initialize instance labels

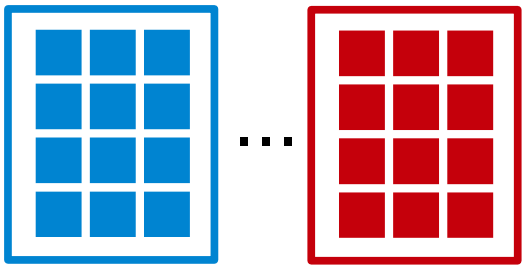


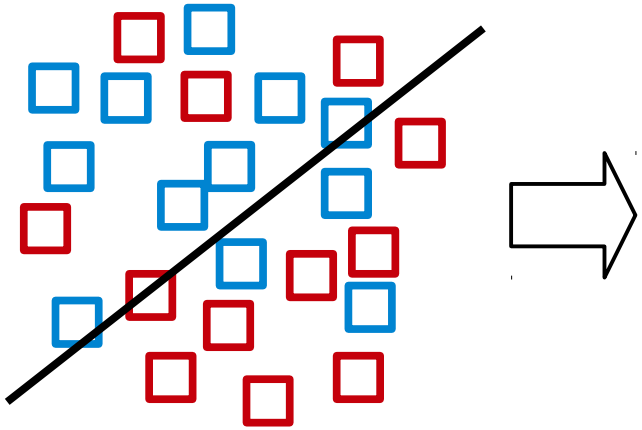
2) Train classifier

Method 2: Iterative MI with Majority Vote

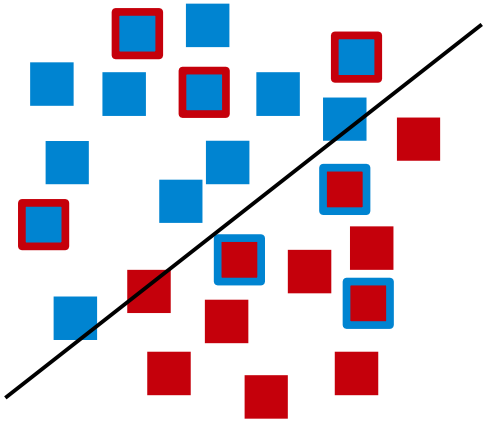
Learn Instance Labels

1) Initialize instance labels





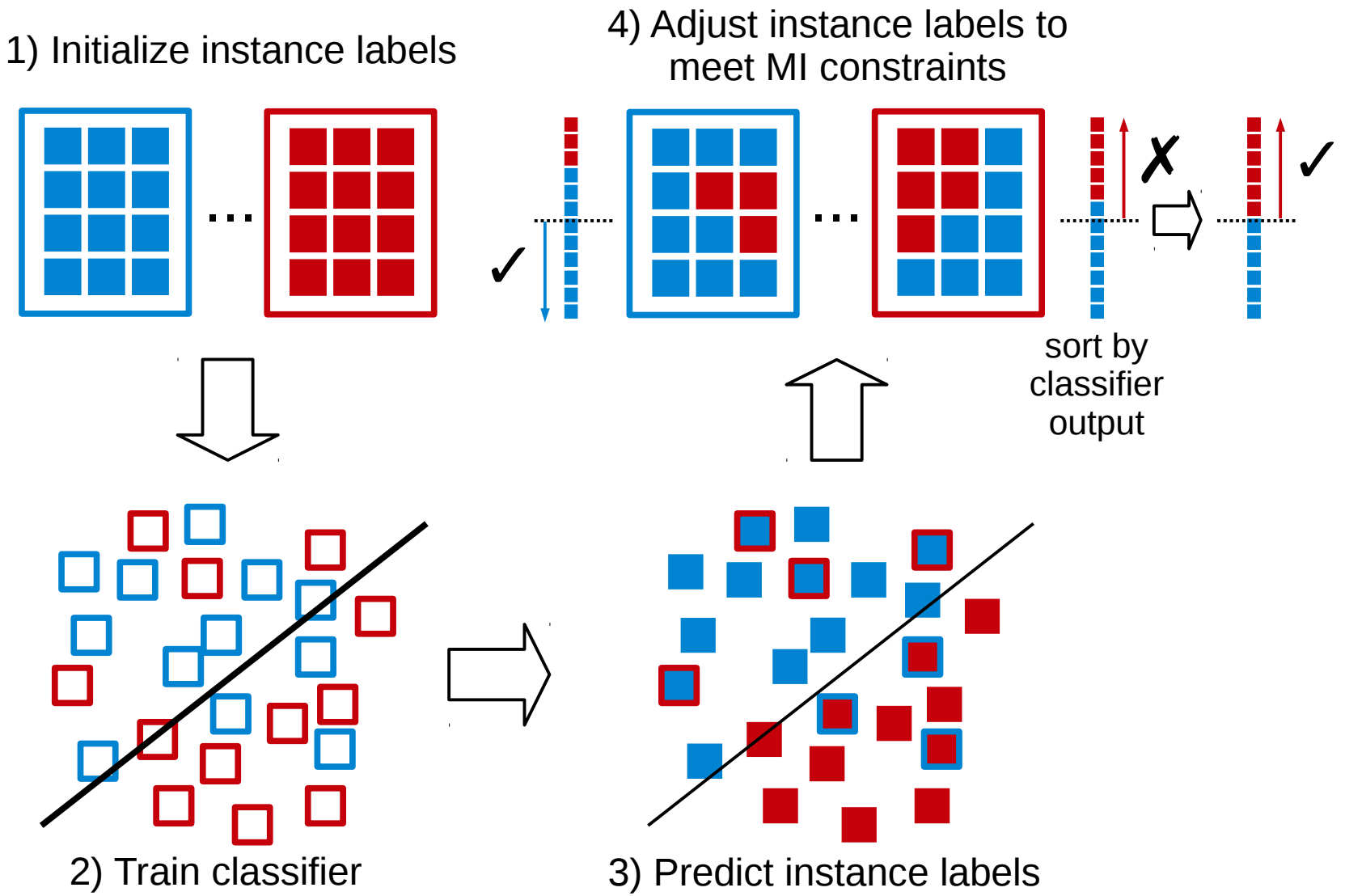
2) Train classifier



3) Predict instance labels

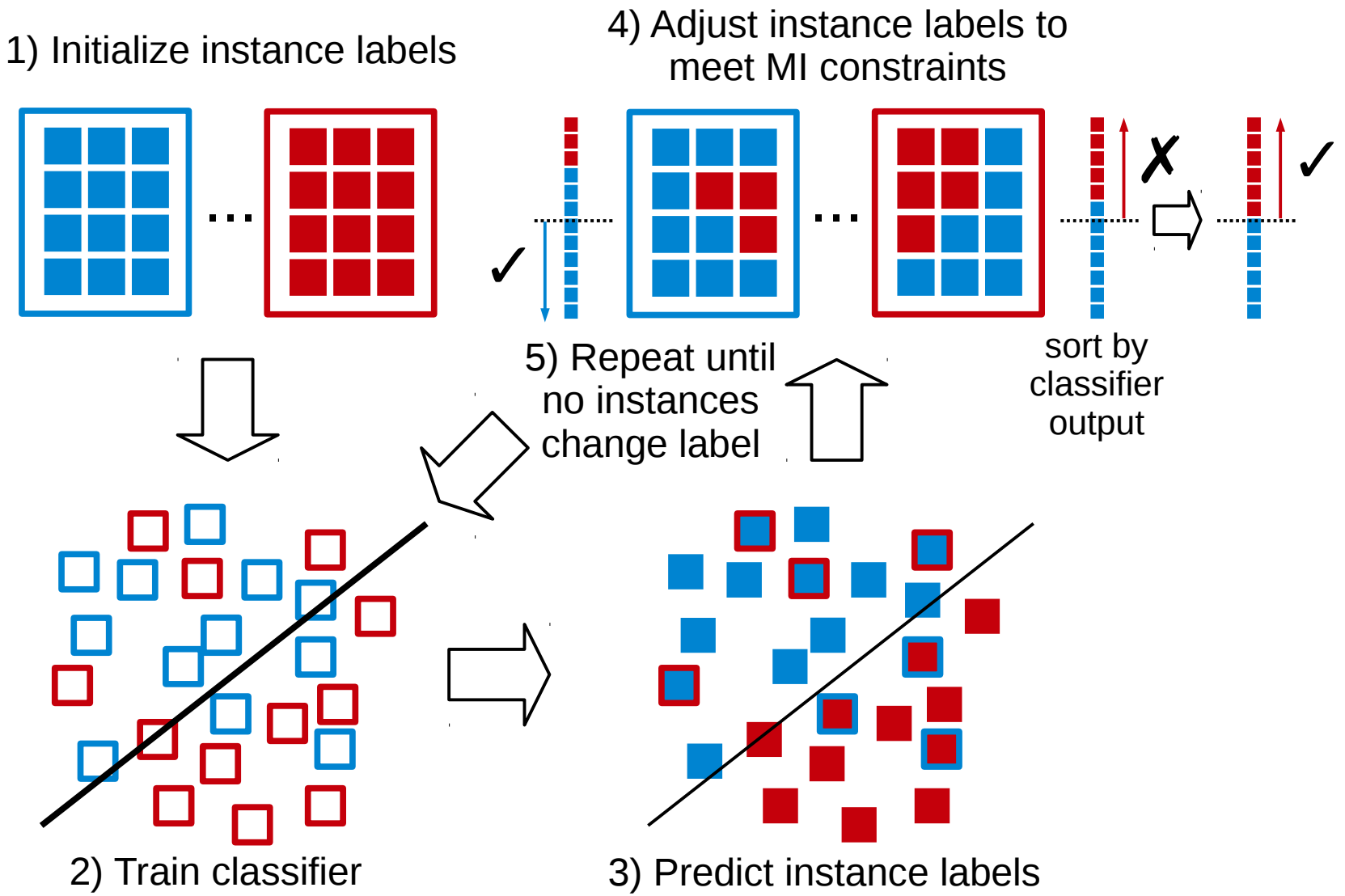
Method 2: Iterative MI with Majority Vote

Learn Instance Labels



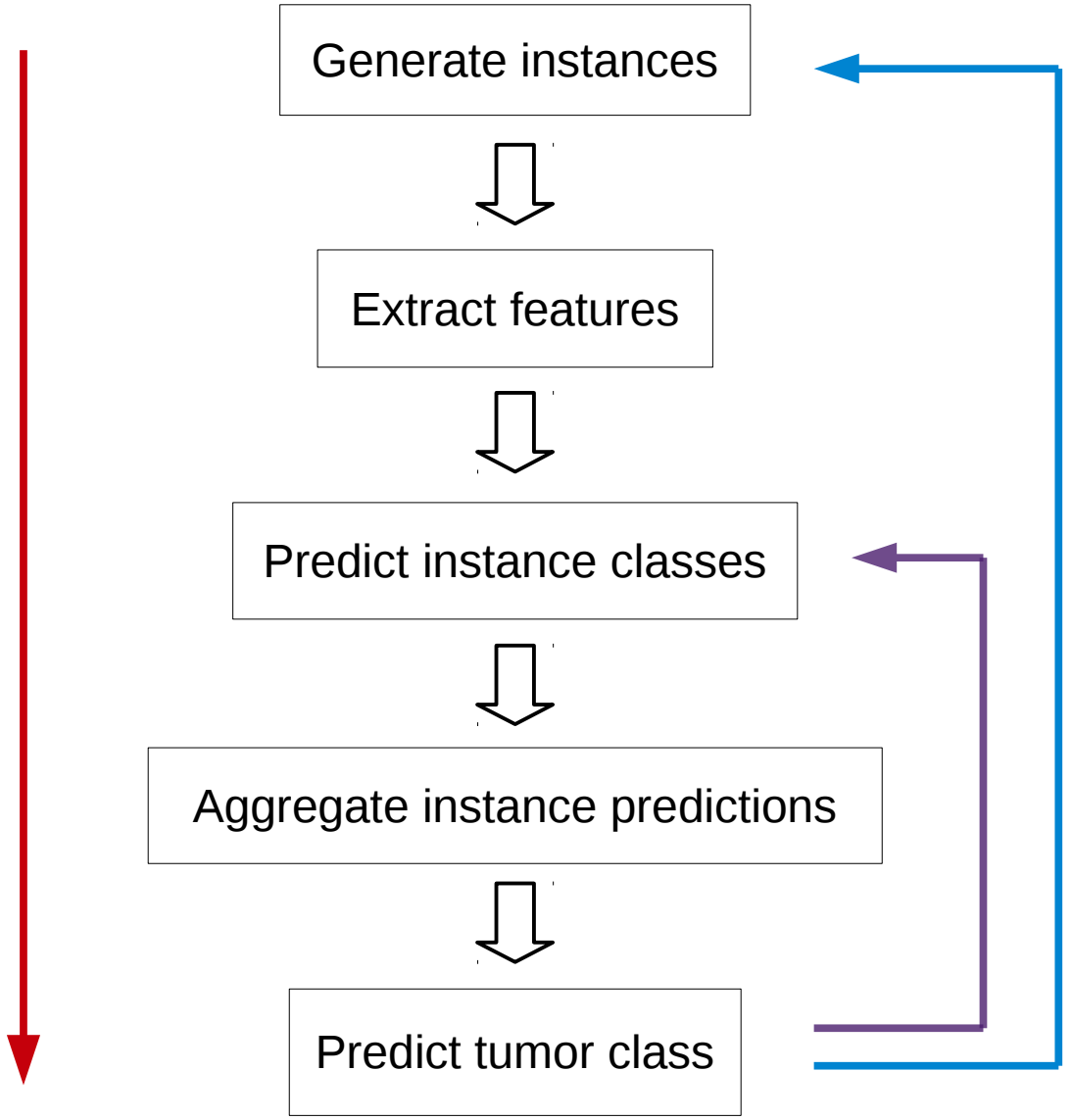
Method 2: Iterative MI with Majority Vote

Learn Instance Labels



Method Overview

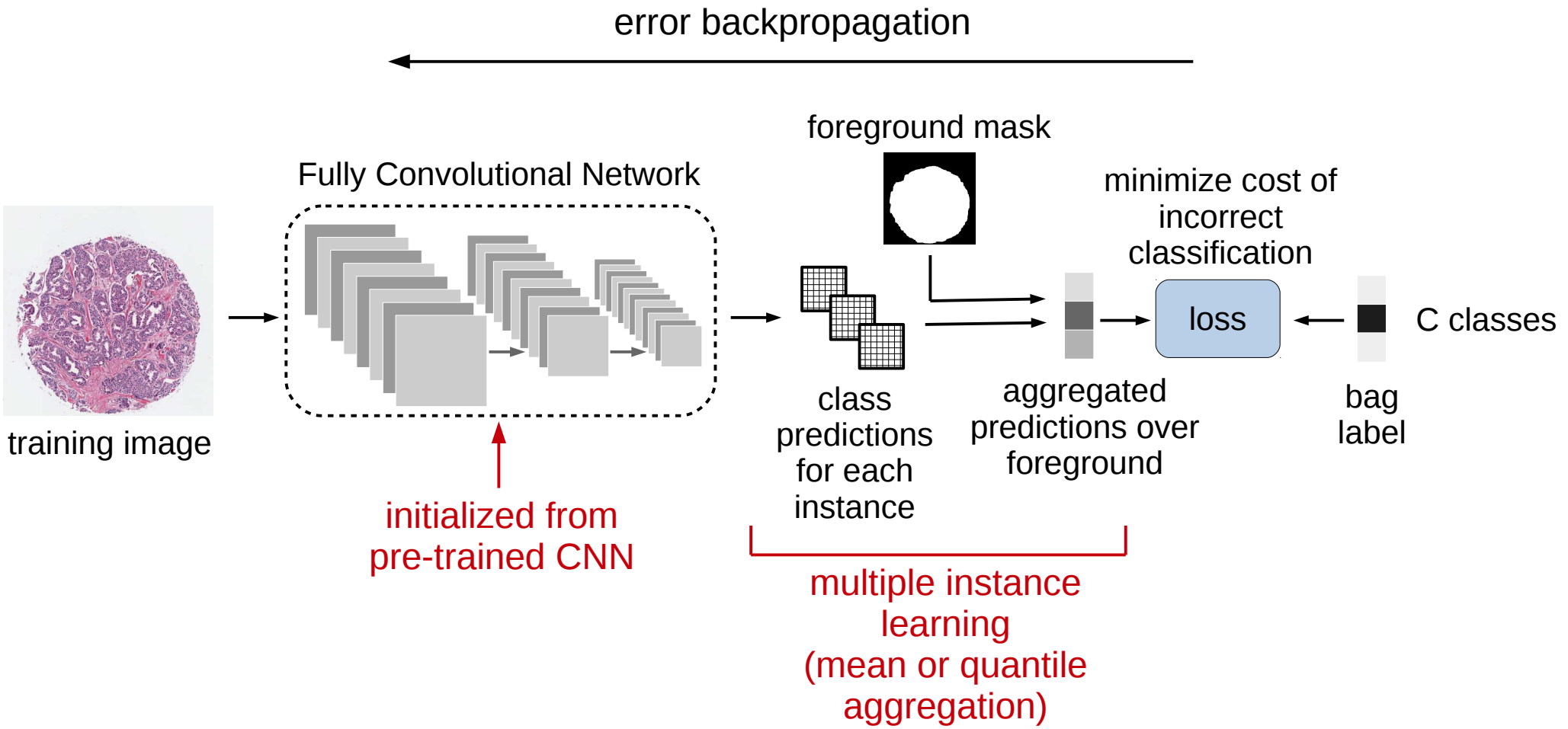
Method 1:
Extract features
+
train classifier



Method 2:
Learn latent instance labels

Method 3:
End-to-end fine-tuning of CNN

Method 3: Fine-tune CNN



Method Comparison

Classification Accuracy

Method	Basal vs. Non-Basal	ER Status	Grade 1 vs. 3
AlexNet			
Baseline: Majority vote	0.776	0.772	0.853
Method 1: Quantile aggregation	0.799	0.815	0.876
Method 2: Iterative MI with majority vote	0.788	0.807	0.870
Method 3: Fine-tune CNN	**0.831**	**0.841**	**0.954**
VGG16			
Baseline: Majority vote	0.807	0.823	0.897
Method 1: Quantile aggregation	0.824	0.853	0.908
Method 2: Iterative MI with majority vote	0.812	0.846	0.905
Method 3: Fine-tune CNN	**0.833**	**0.879**	**0.973**

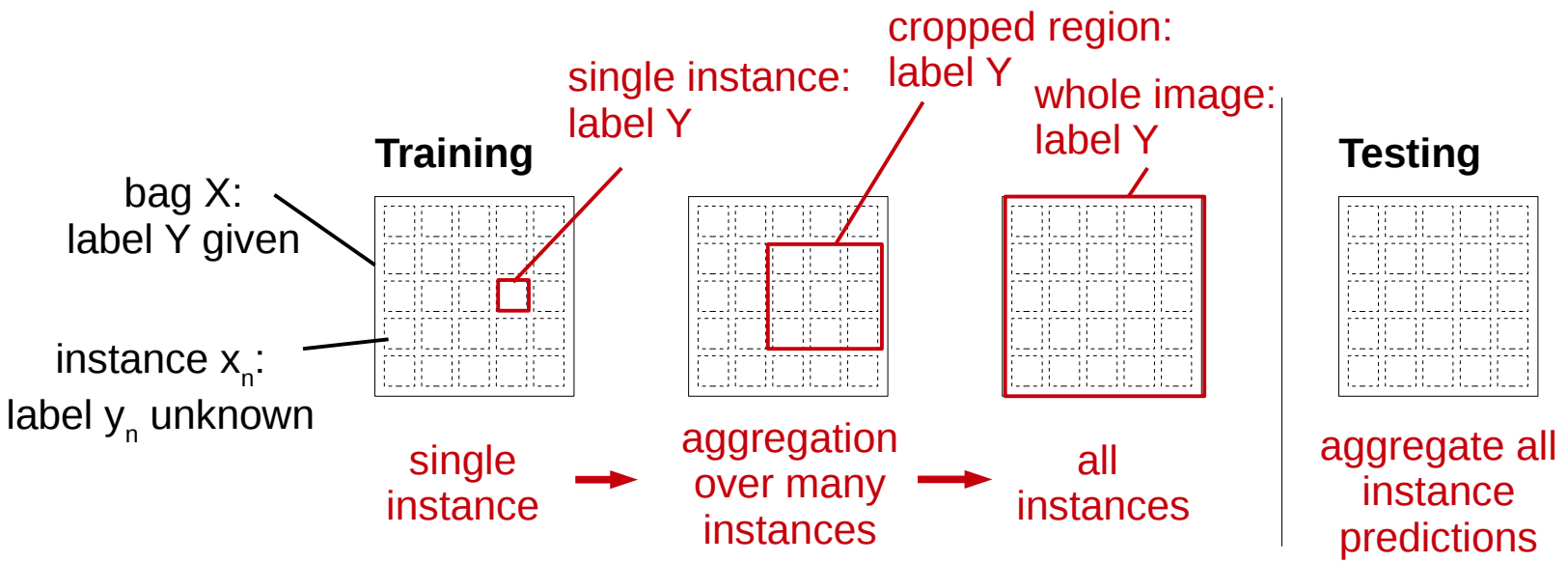
CBCS data set: 1713 patient samples, 4 cores/patient
5x random split: ½ train, ½ test

MI techniques for training always beneficial

Fine-tuning CNN gives largest improvement

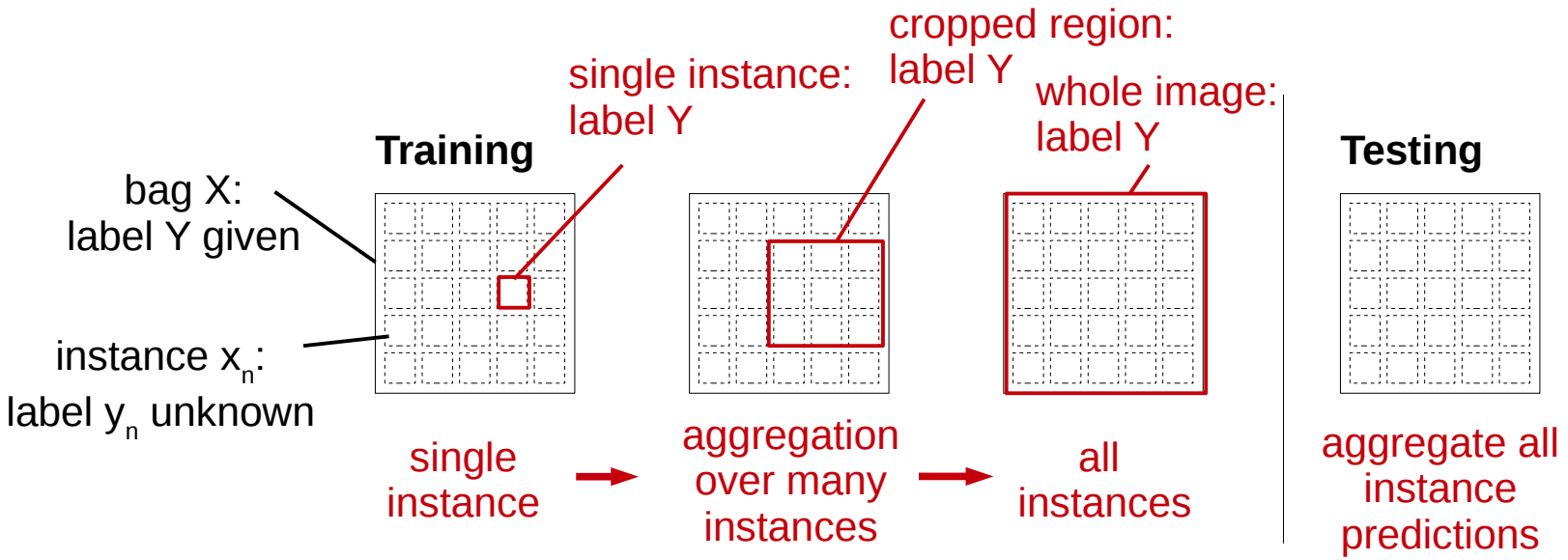
Method 3: Fine-tune CNN

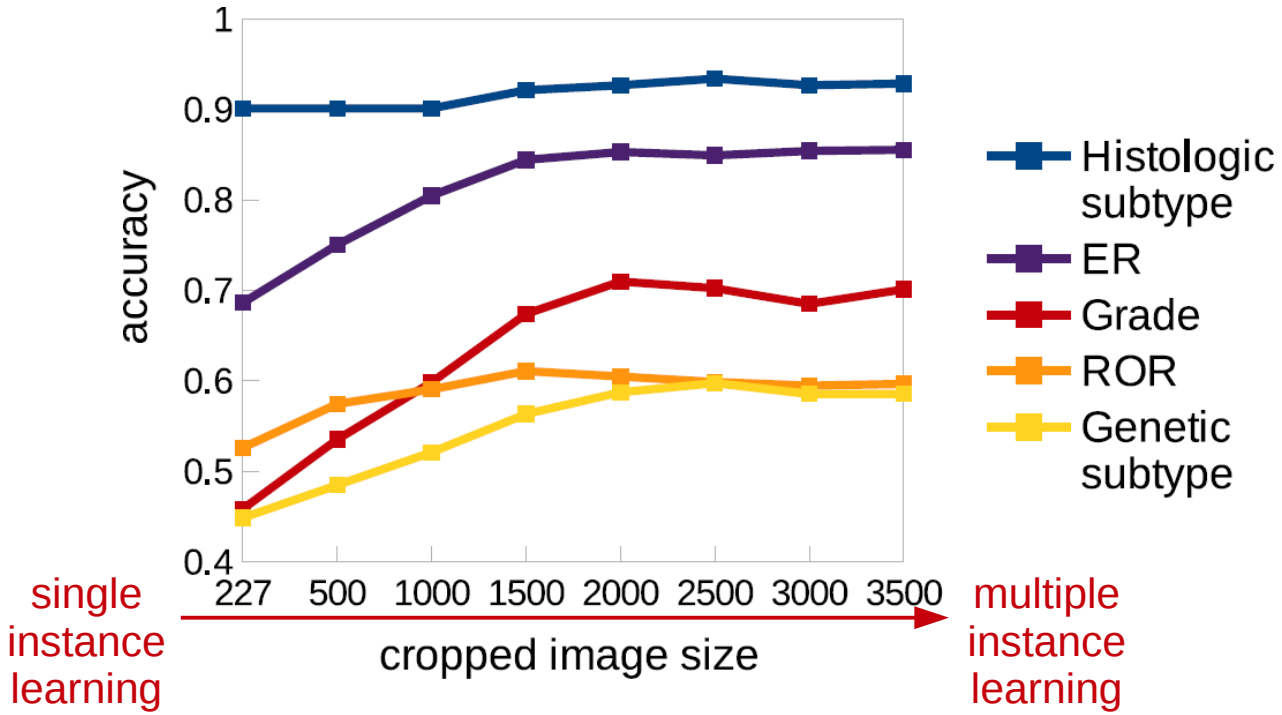
MI Learning



Method 3: Fine-tune CNN

MI Learning

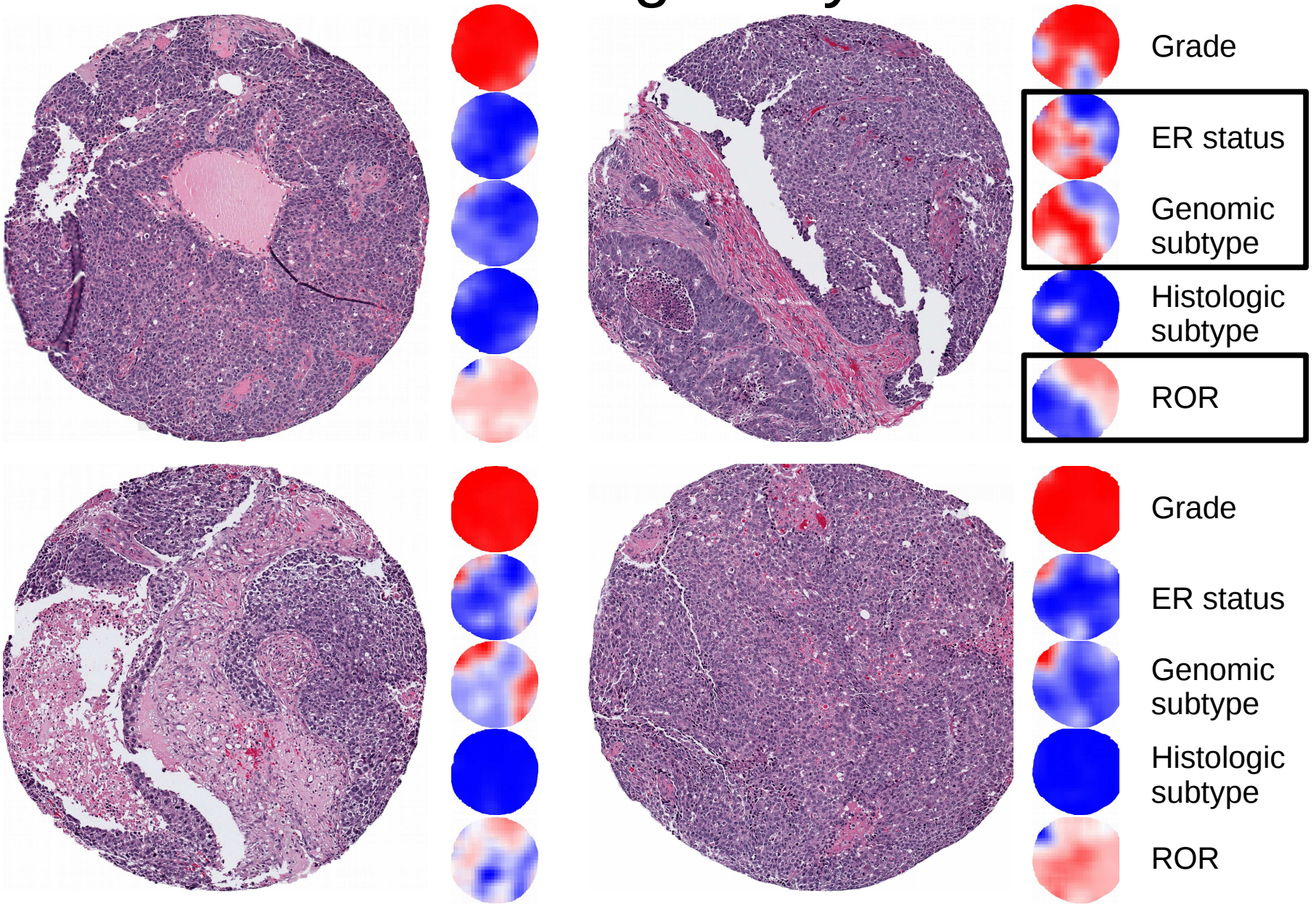




MI techniques during training essential on heterogeneous images

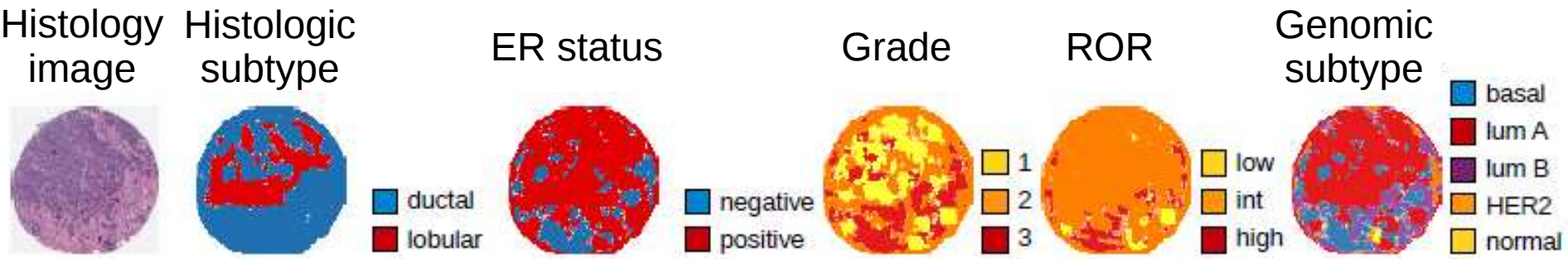
Method 1: Quantile Aggregation

Heterogeneity



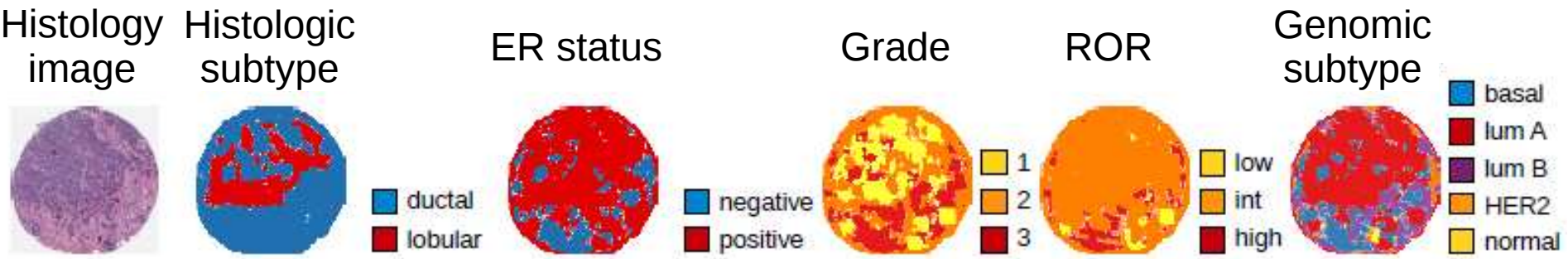
Method 3: Fine-tune CNN

Heterogeneity

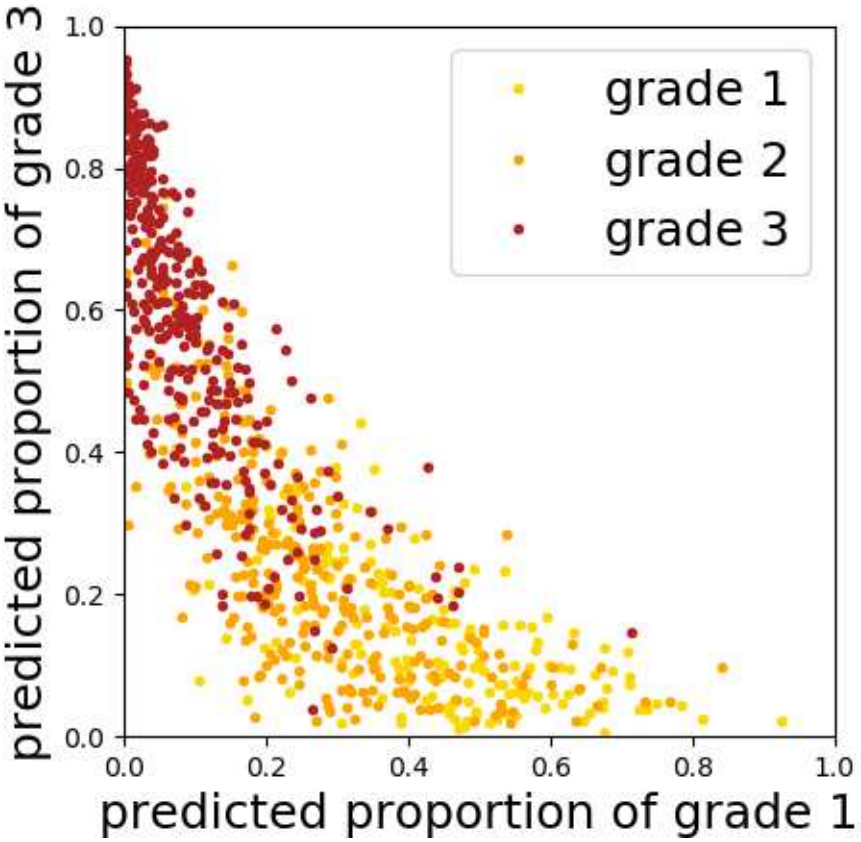


Method 3: Fine-tune CNN

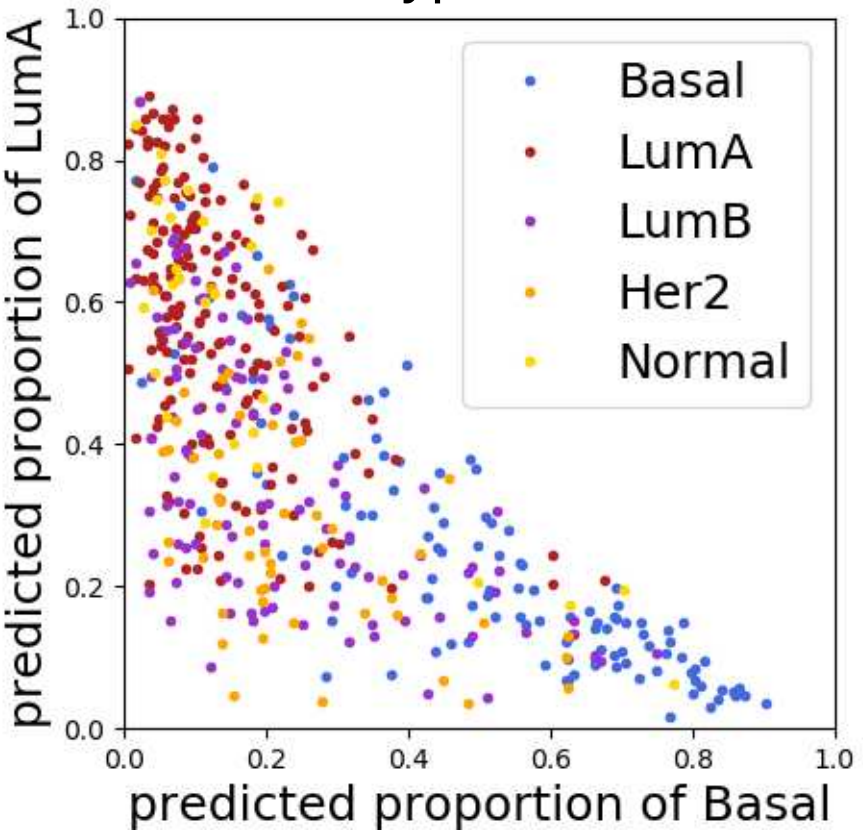
Heterogeneity



Grade 1 vs. 3



Genomic Subtype Basal vs. LumA



Future Work

- Validation of heterogeneity
- Outcome prediction
- Model visualization and interpretation
- Other cancer and disease types

Outline

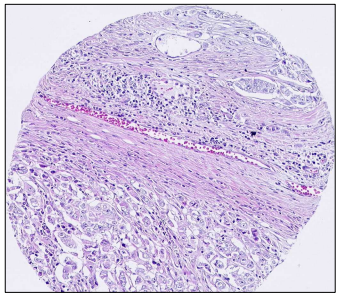
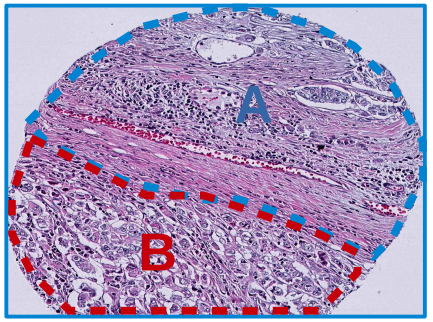


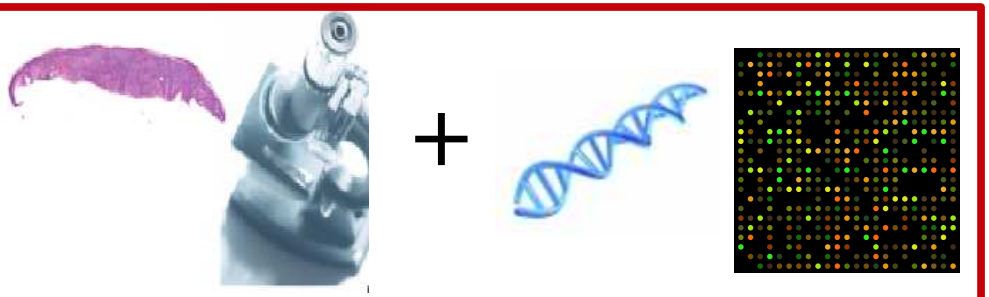
image features

tumor class

1) Representing histology images

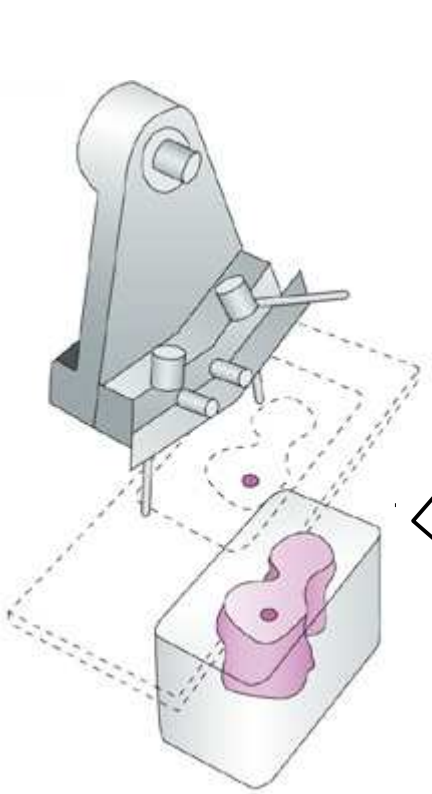


2) Handling heterogeneous images

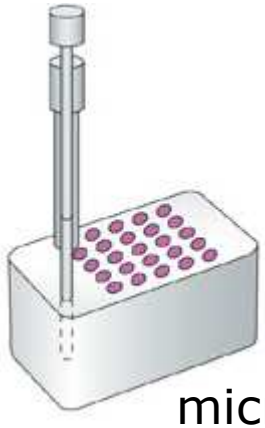


3) Combining imaging & genomics

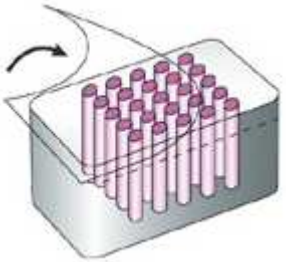
Tissue Microarray

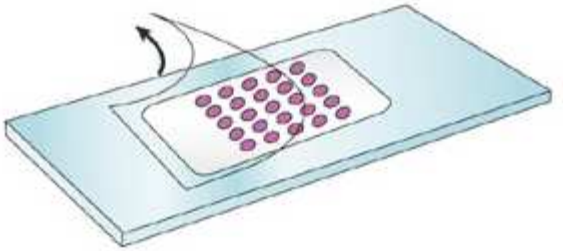


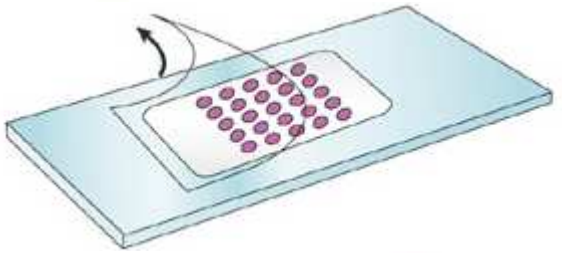
tumor area identified on slide

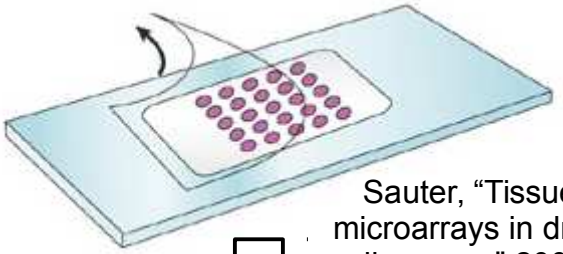


microarray assembly





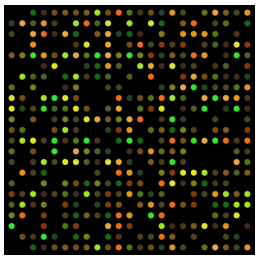




Sauter, "Tissue microarrays in drug discovery," 2003

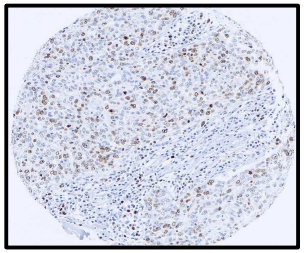
gene expression

genomic subtype



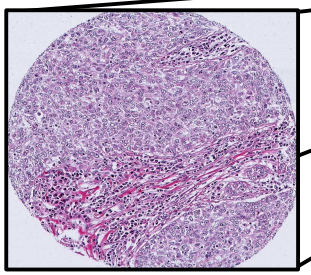
immunohistochemistry

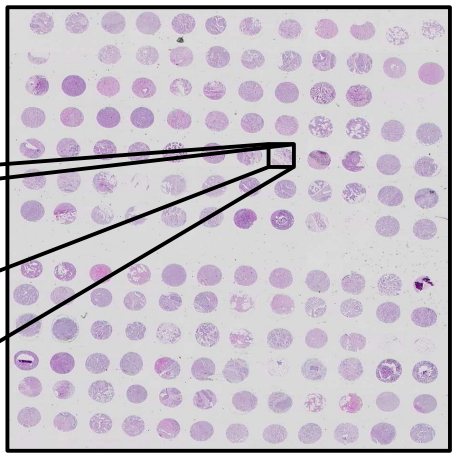
receptor status



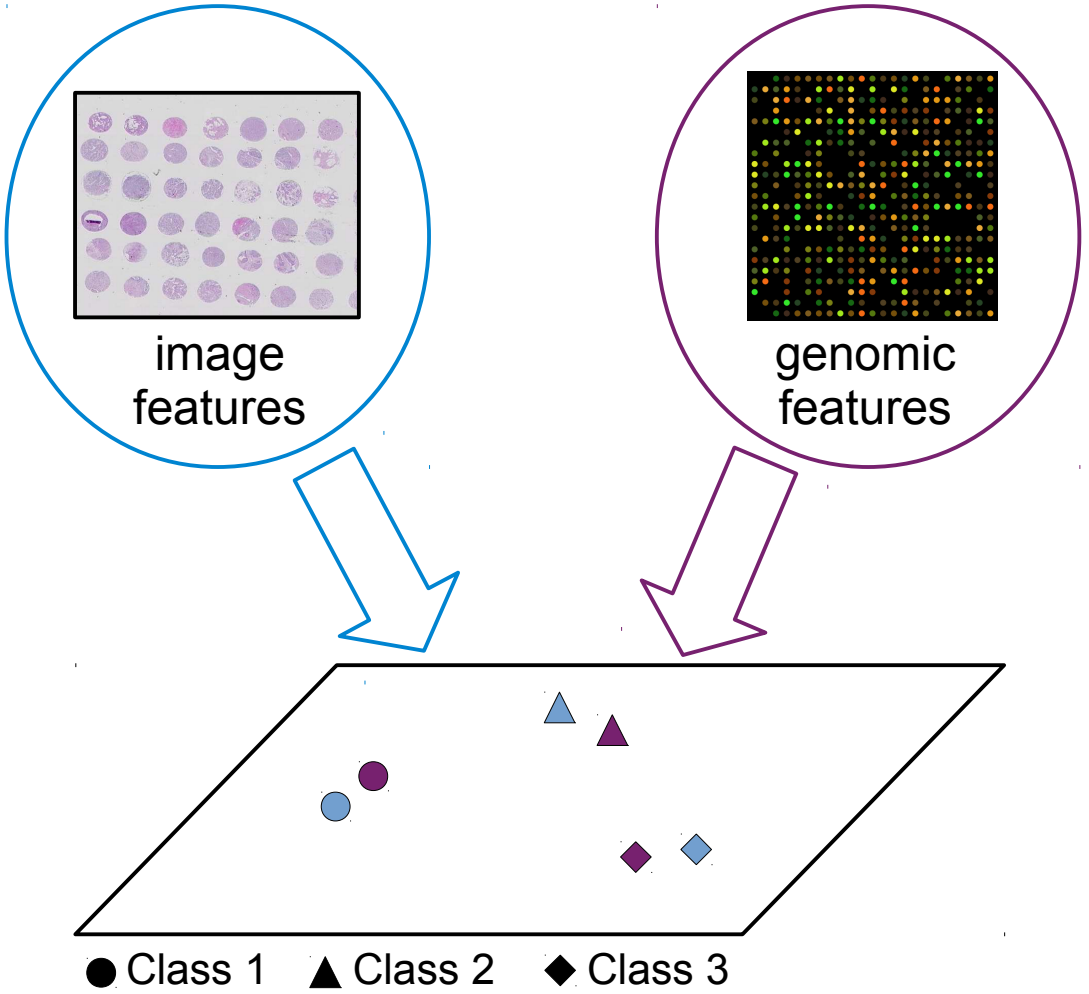
H&E histology

histologic subtype, grade





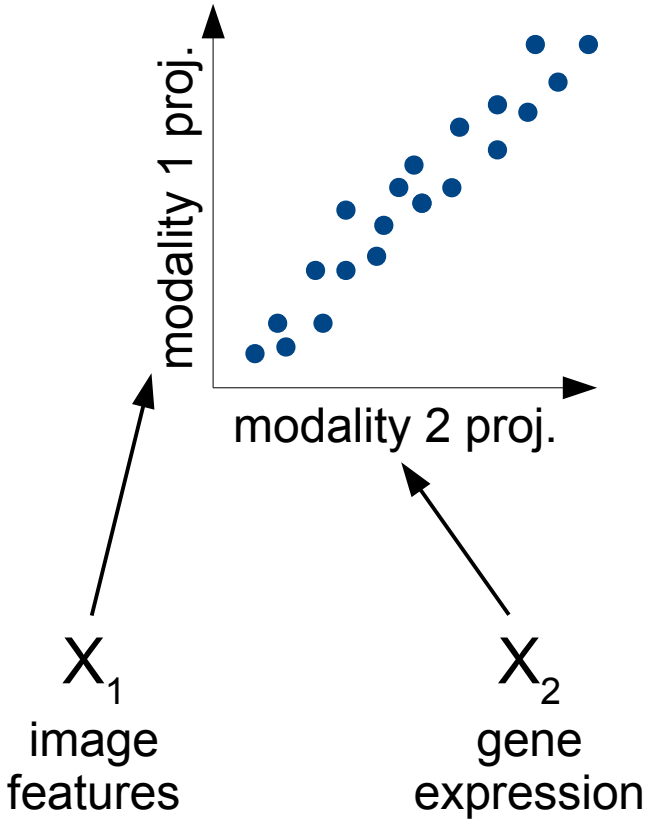
Discriminative Common Space



Extract shared components of data

Improve discriminability

Standard Solution: Canonical Correlation Analysis



Find space in which modalities are maximally correlated

Challenges:

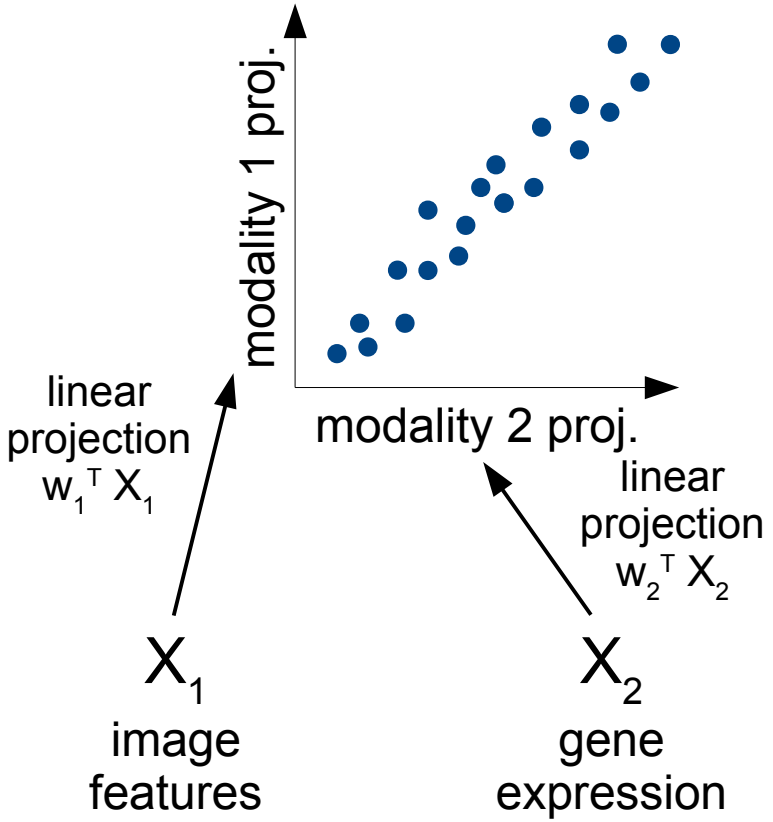
Features may not be discriminative

Not robust to small training set size or high dimensional, low sample size (HDLSS) data

My solution:

Add task-driven component to a deep variant of CCA

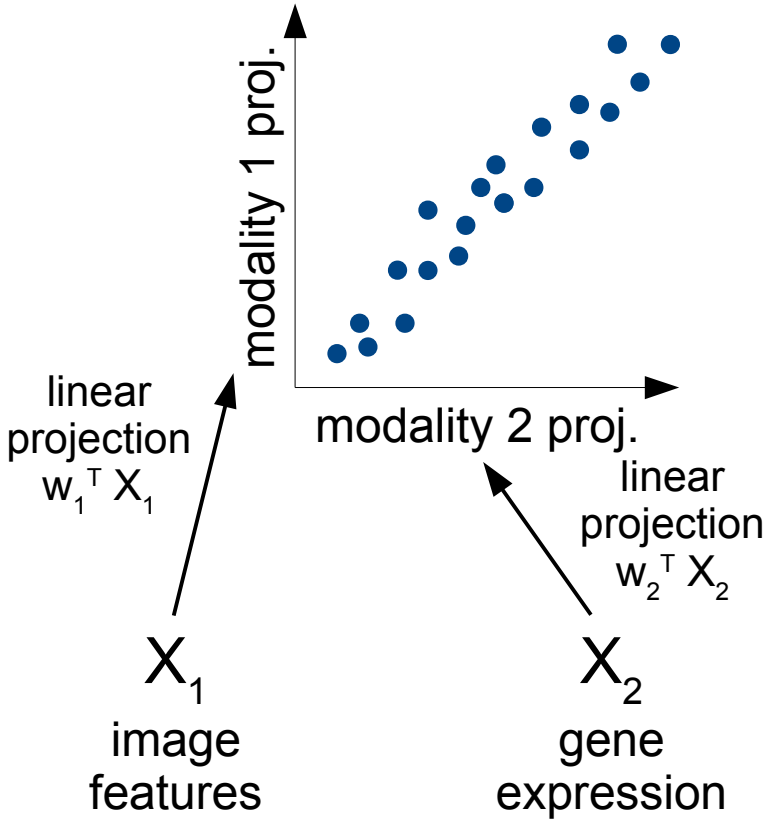
Background: Canonical Correlation Analysis



Maximize correlation

$$X_1 \in \Re^{d_1 x n}, \quad X_2 \in \Re^{d_2 x n}$$

Background: Canonical Correlation Analysis



Maximize sum correlation from orthogonal projections

$$X_1 \in \Re^{d_1 x n}, \quad X_2 \in \Re^{d_2 x n}$$

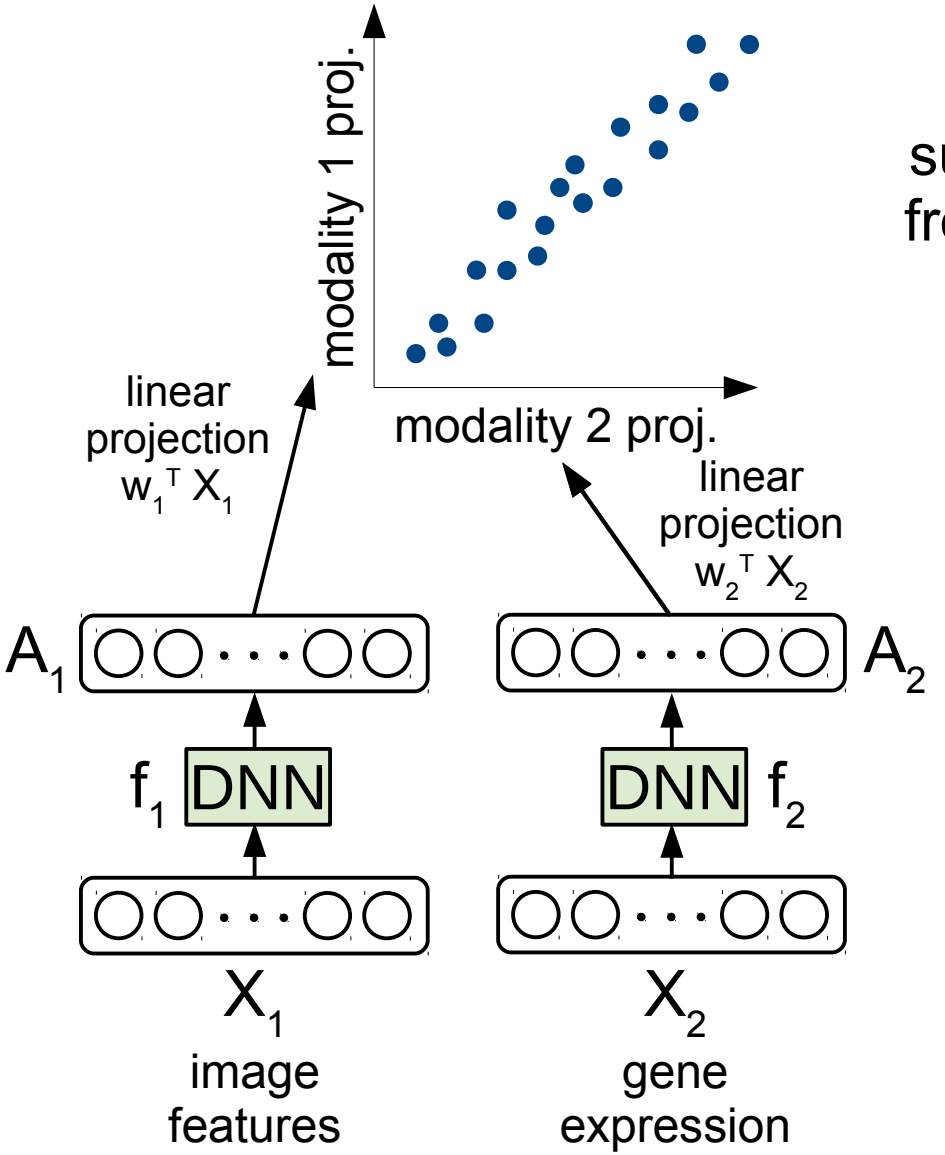
Goal: $\underset{W_1, W_2}{\text{argmax}} \sum_k \text{corr}\left(\left(w_1^{(k)}\right)^T X_1, \left(w_2^{(k)}\right)^T X_2 \right)$ (sum correlation)

such that $\left(w_1^{(i)}\right)^T X_1 X_1^T w_1^{(j)} = \left(w_2^{(i)}\right)^T X_2 X_2^T w_2^{(j)} = 0 \quad \text{for} \quad i \neq j$ (orthogonal projections)

Solved with SVD

Background: Deep Canonical Correlation Analysis

(Andrew, 2013)



Maximize sum correlation from orthogonal projections

$$A_1 \in \Re^{d_o x n}, \quad A_2 \in \Re^{d_o x n}$$

features replaced by activations

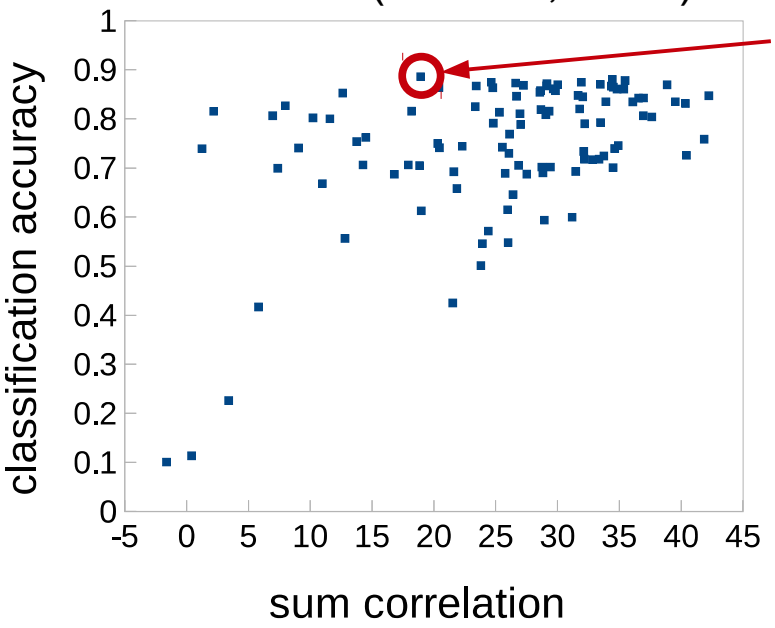
Deep CCA: Correlation vs. Accuracy

MNIST: 28 x 28 images of digits
Split into left and right: 28 x 14 each

Cross-modal classification:

1) Train CCA with both modalities
2) Compute projections for both modalities
3) Train classifier on projection from 1st modality
4) Test with projection from 2nd modality

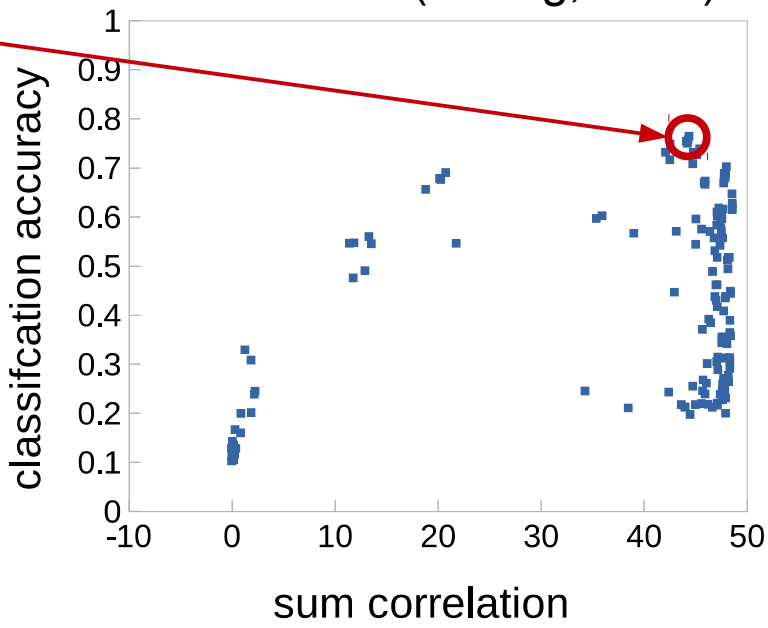
DCCA (Andrew, 2013)



best classification accuracy

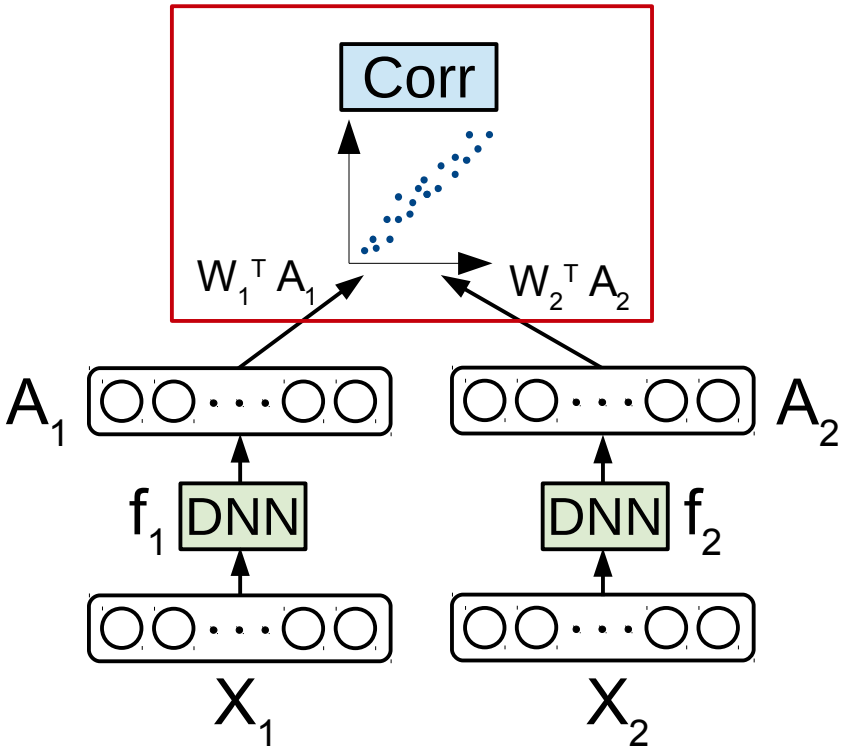
correlation ≠ classification accuracy

SoftCCA (Chang, 2018)



variety of hyperparameters
50 components

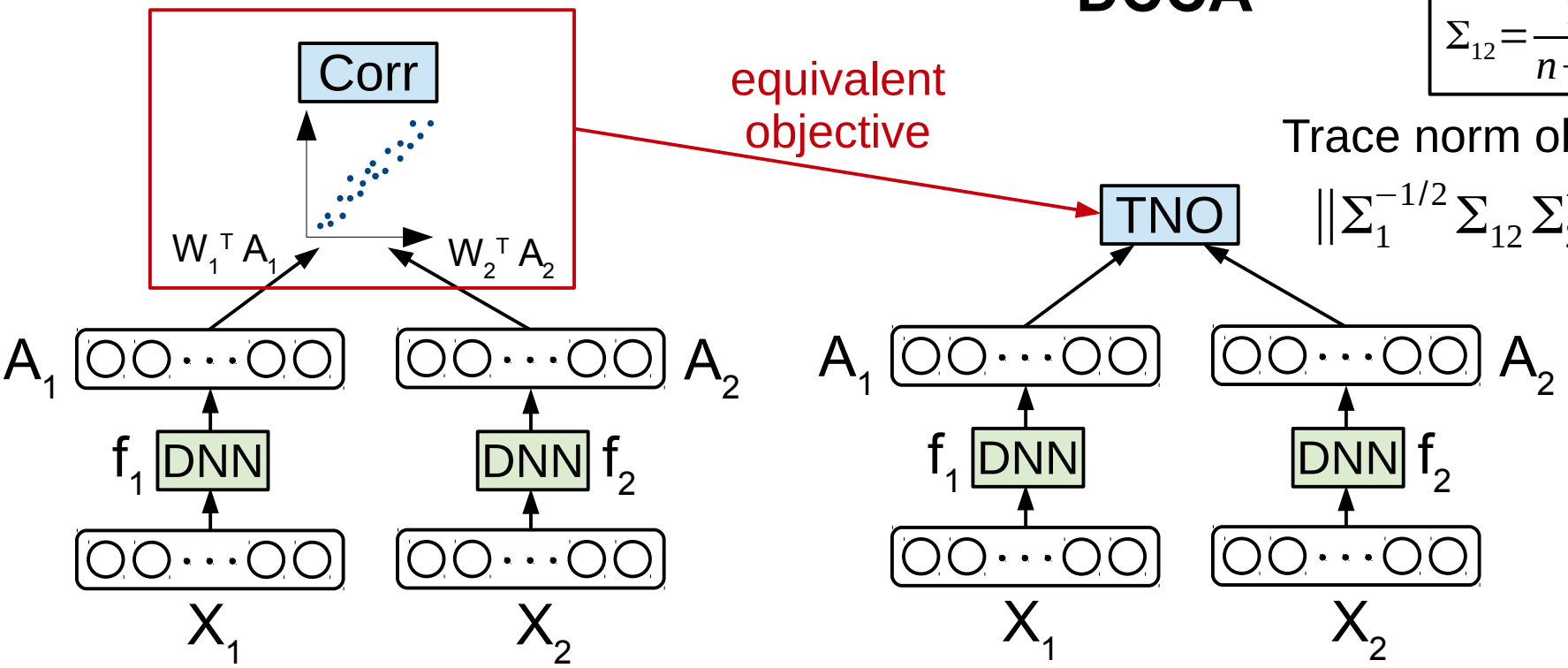
Deep CCA Challenges



Deep CCA Challenges

$$\Sigma_1 = \frac{1}{n-1} A_1 A_1^T$$
$$\Sigma_2 = \frac{1}{n-1} A_2 A_2^T$$
$$\Sigma_{12} = \frac{1}{n-1} A_1 A_2^T$$

DCCA



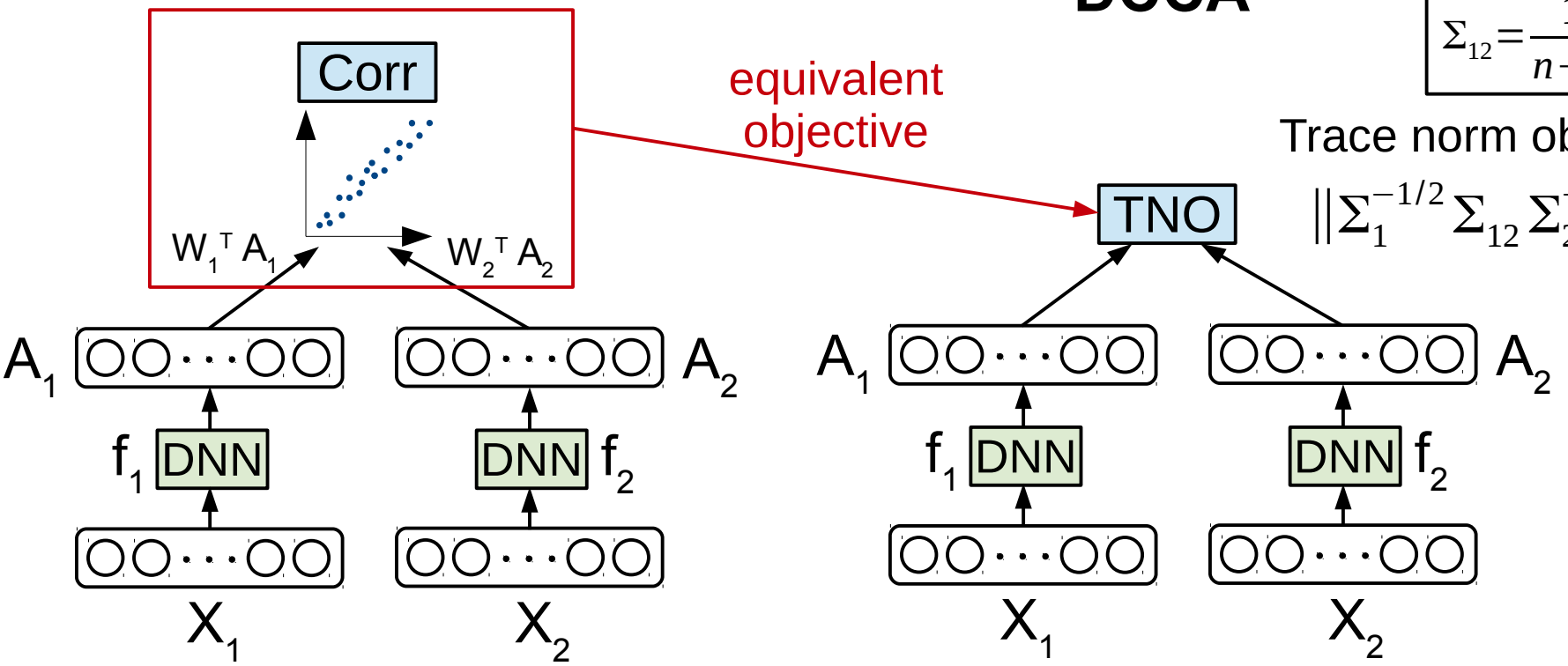
Trace norm objective:

$$\left\| \Sigma_1^{-1/2} \Sigma_{12} \Sigma_2^{-1/2} \right\|_{tr}$$

Deep CCA Challenges

$$\Sigma_1 = \frac{1}{n-1} A_1 A_1^T$$
$$\Sigma_2 = \frac{1}{n-1} A_2 A_2^T$$
$$\Sigma_{12} = \frac{1}{n-1} A_1 A_2^T$$

DCCA



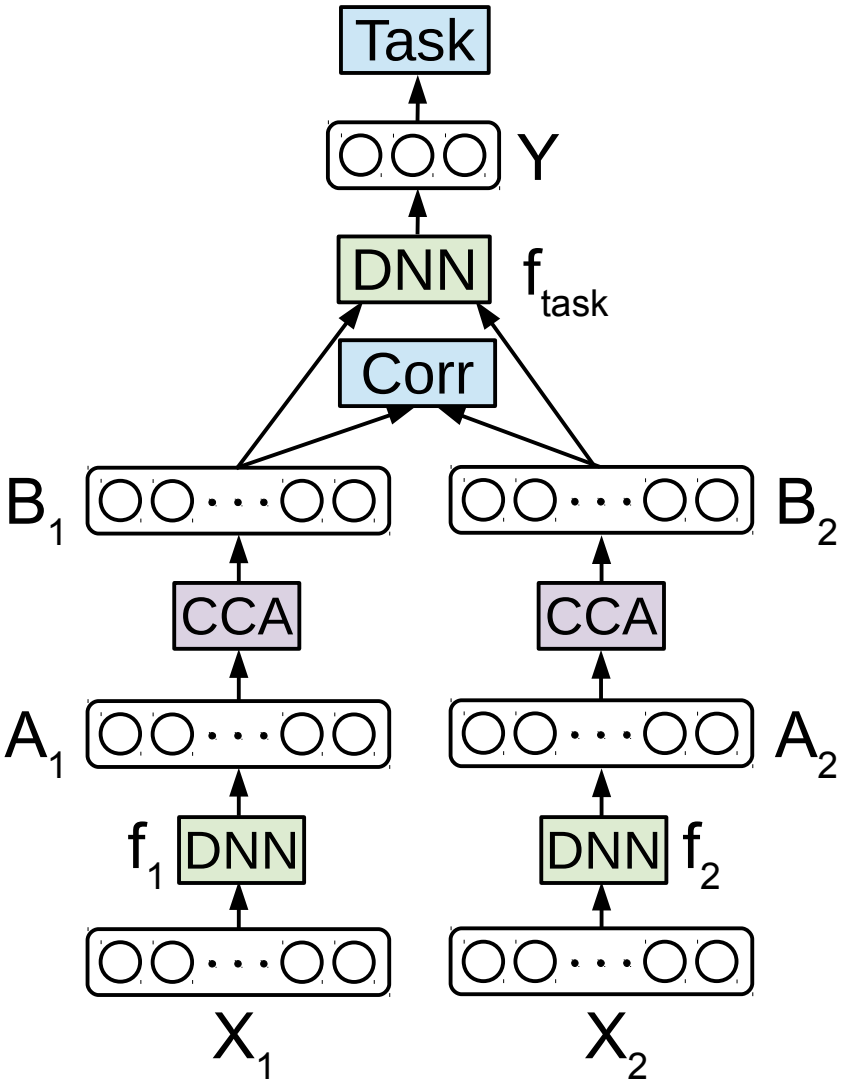
Trace norm objective:

$$\left\| \Sigma_1^{-1/2} \Sigma_{12} \Sigma_2^{-1/2} \right\|_{tr}$$

Compute linear CCA projections from A_1 and A_2 after DNN optimization

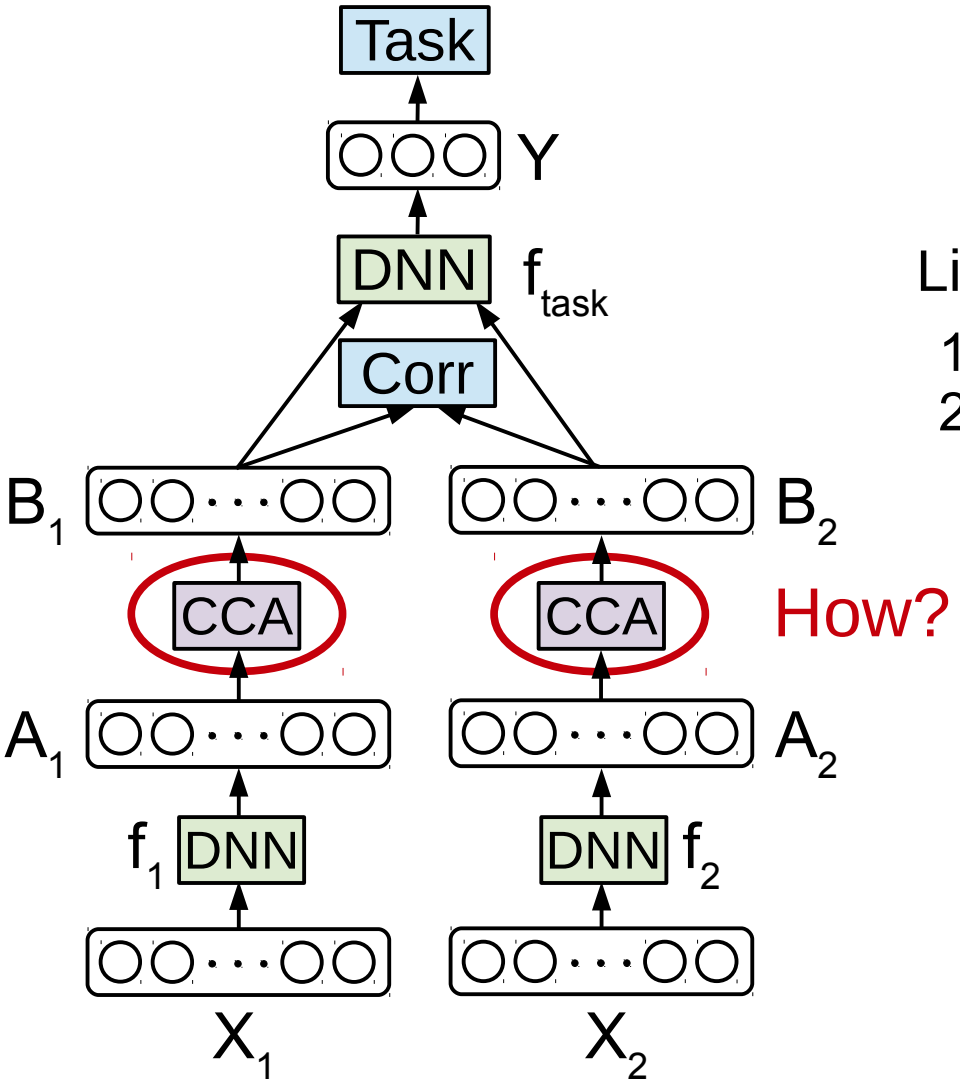
Task-driven Deep CCA

Goal: add task-driven objective (e.g., for classification)



Task-driven Deep CCA

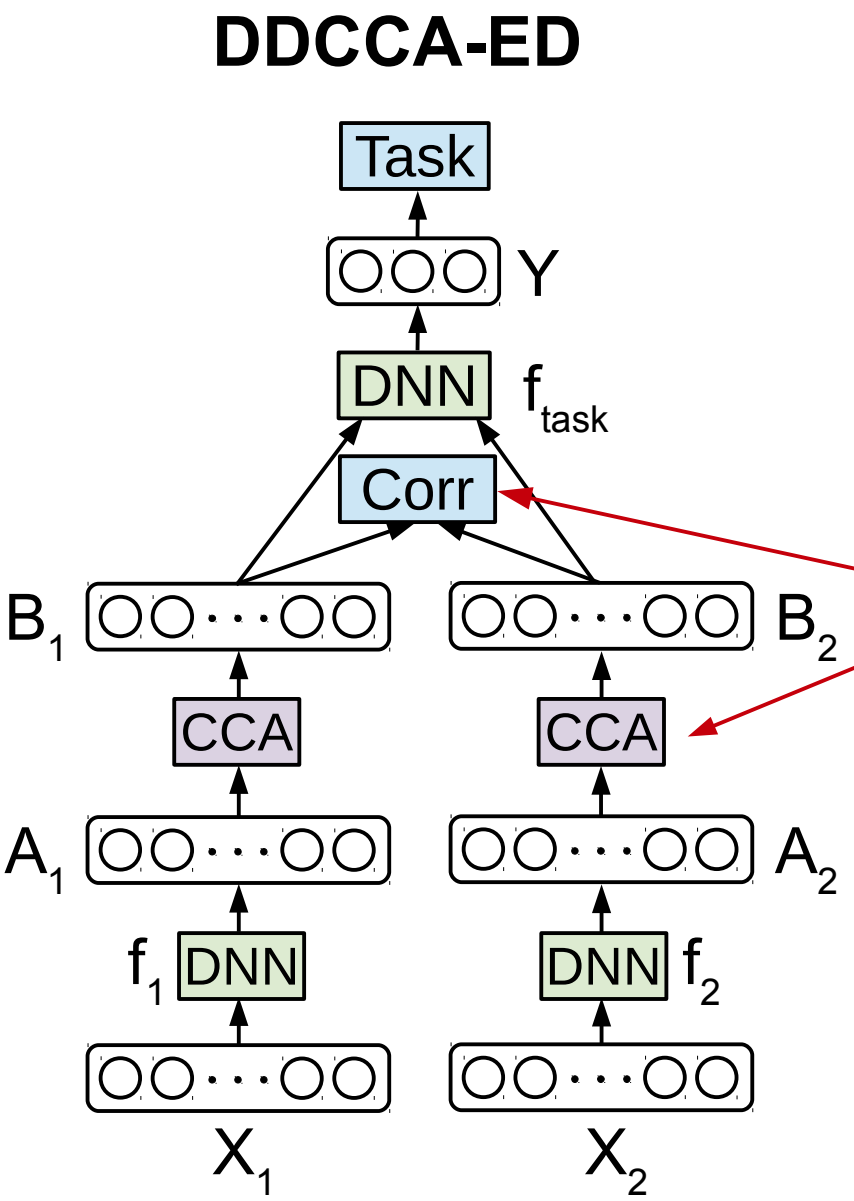
Goal: add task-driven objective (e.g., for classification)



Linear CCA:

1) Maximize sum correlation
2) Such that projections are orthogonal

Solution 1: Eigendecomposition

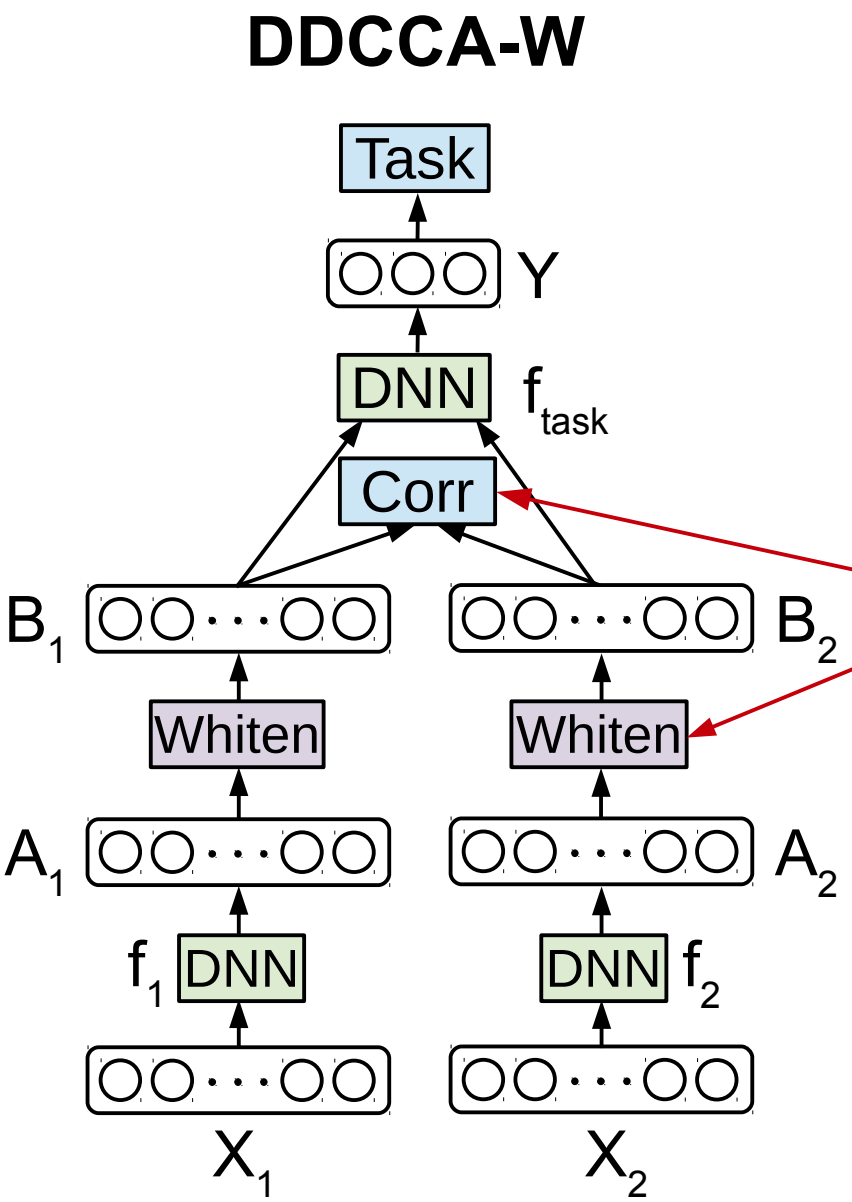


1) Maximize sum correlation
2) Such that projections are orthogonal

Compute CCA projections with eigendecomposition instead of SVD

aka CCA Layer (Dorfer, 2018)

Solution 2: Whitening

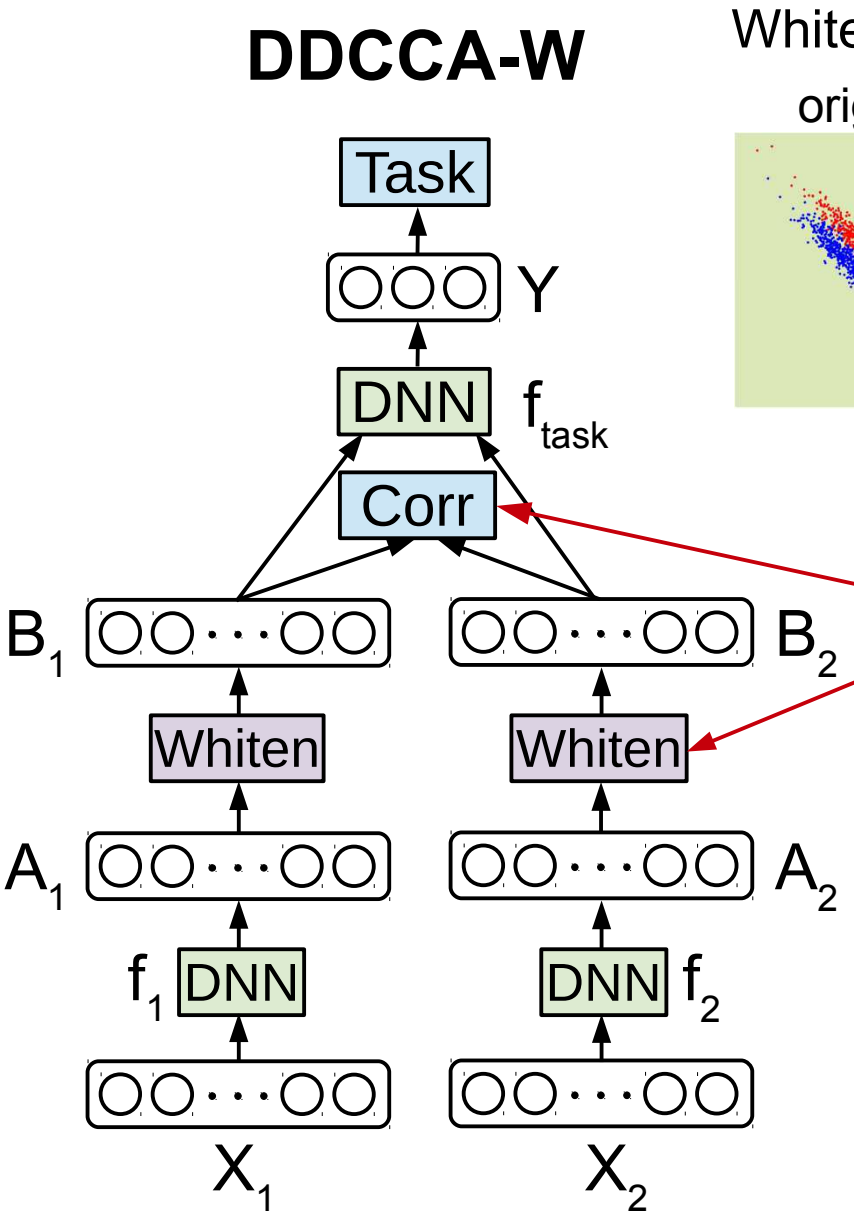


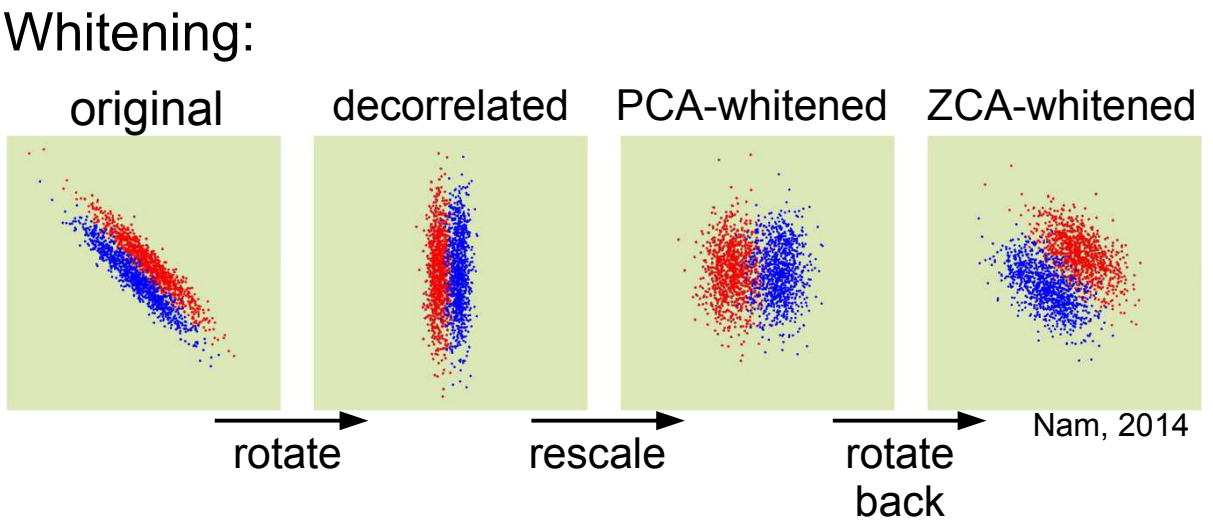
1) Maximize sum correlation
2) Such that projections are orthogonal

No explicit CCA projection:
learn ZCA whitening transform to orthogonalize activations

aka Decorrelated Batch Normalization (Huang, 2018)

Solution 2: Whitening





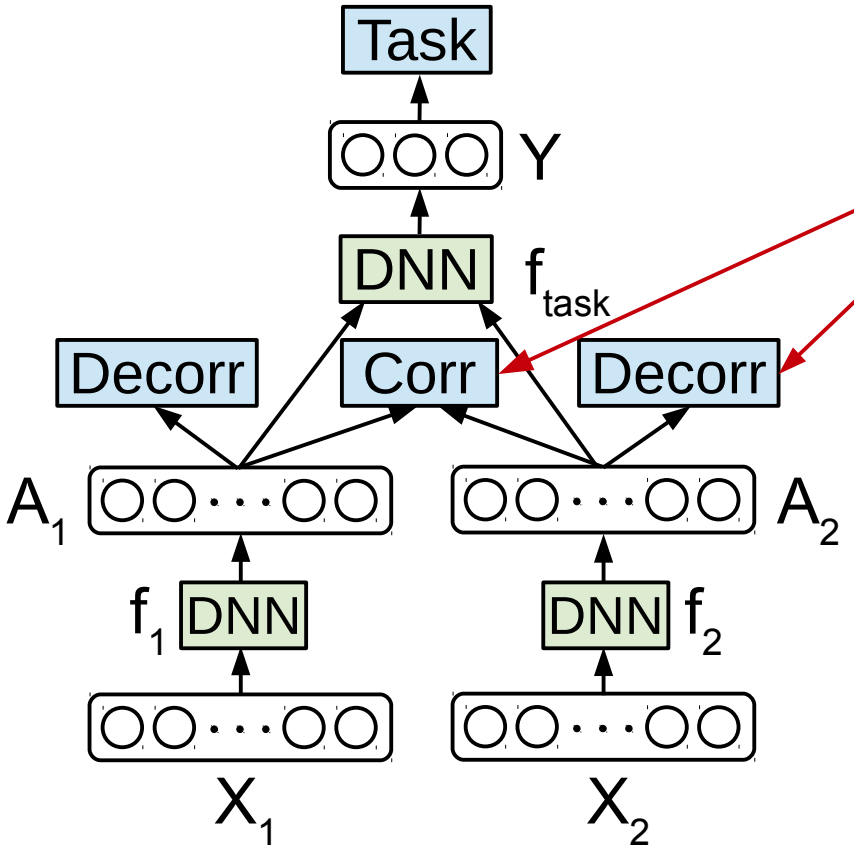
1) Maximize sum correlation
2) Such that projections are orthogonal

No explicit CCA projection:
learn ZCA whitening transform to orthogonalize activations

aka Decorrelated Batch Normalization (Huang, 2018)

Solution 3: Soft Decorrelation

DDCCA-SD



1) Maximize sum correlation
2) Such that projections are orthogonal

Encourage orthogonality using regularization

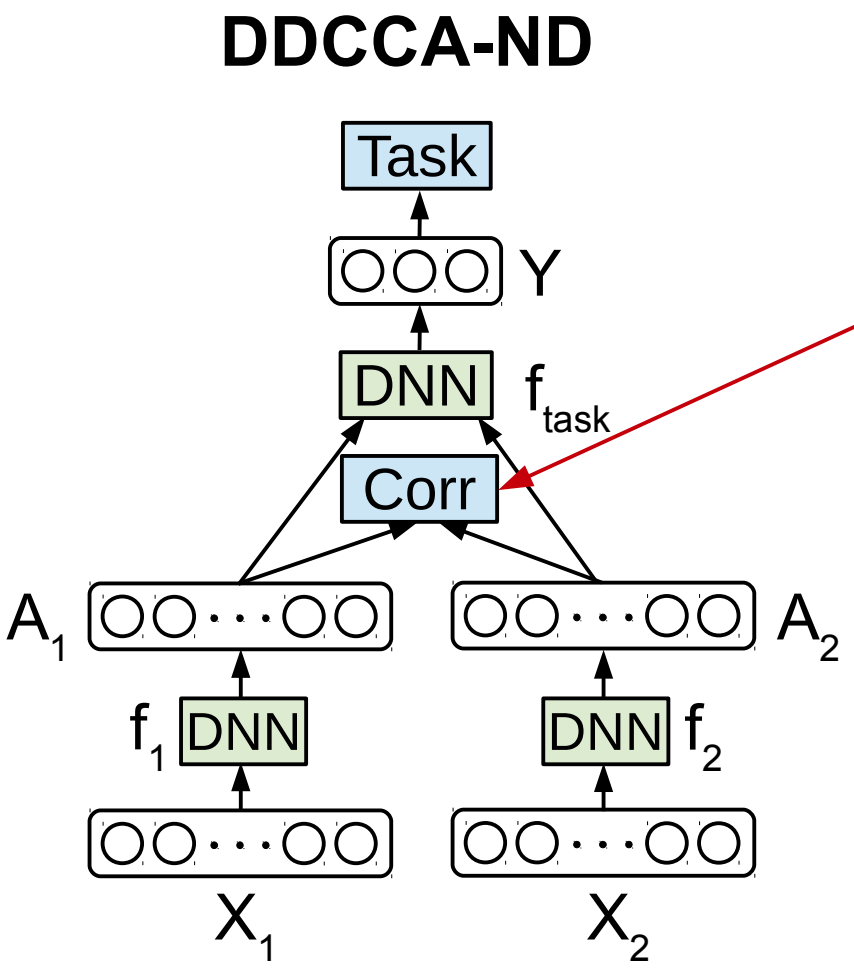
aka DeCov (Cogswell, 2016) or
Soft Decorrelation Loss (Chang, 2018)

$$\mathscr{L}_{Decorr}(\Sigma) = \sum_{i=1}^{d_o} \sum_{j \neq i}^{d_o} |\Sigma_{i,j}|$$

$$\Sigma_1 = \frac{1}{n-1} A_1 A_1^T \qquad \Sigma_2 = \frac{1}{n-1} A_2 A_2^T$$

Penalize off-diagonal elements of covariance matrix

Solution 4: No Explicit Decorrelation

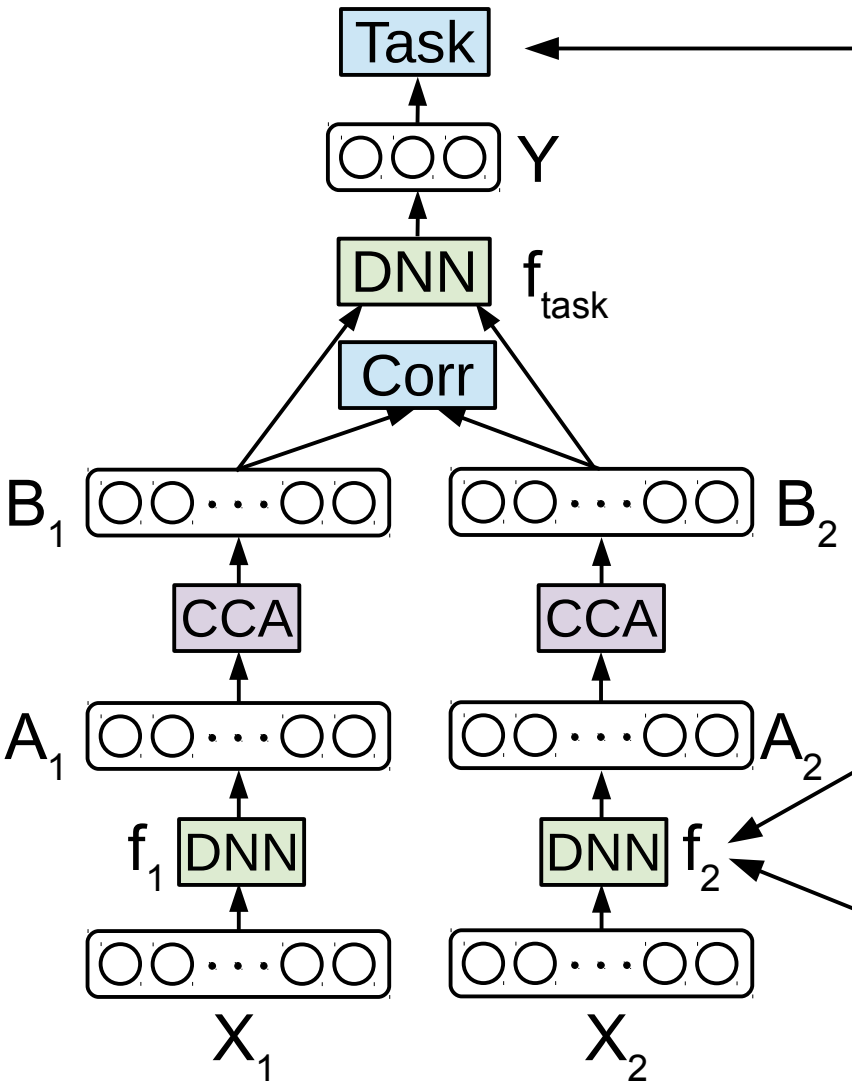


1) Maximize sum correlation

~~2) Such that projections are orthogonal~~

Rely on task objective to decorrelate as needed

Benefits of All DDCCA Models



Improve discriminability

Benefits large and small training sets

Learn non-linear transformation

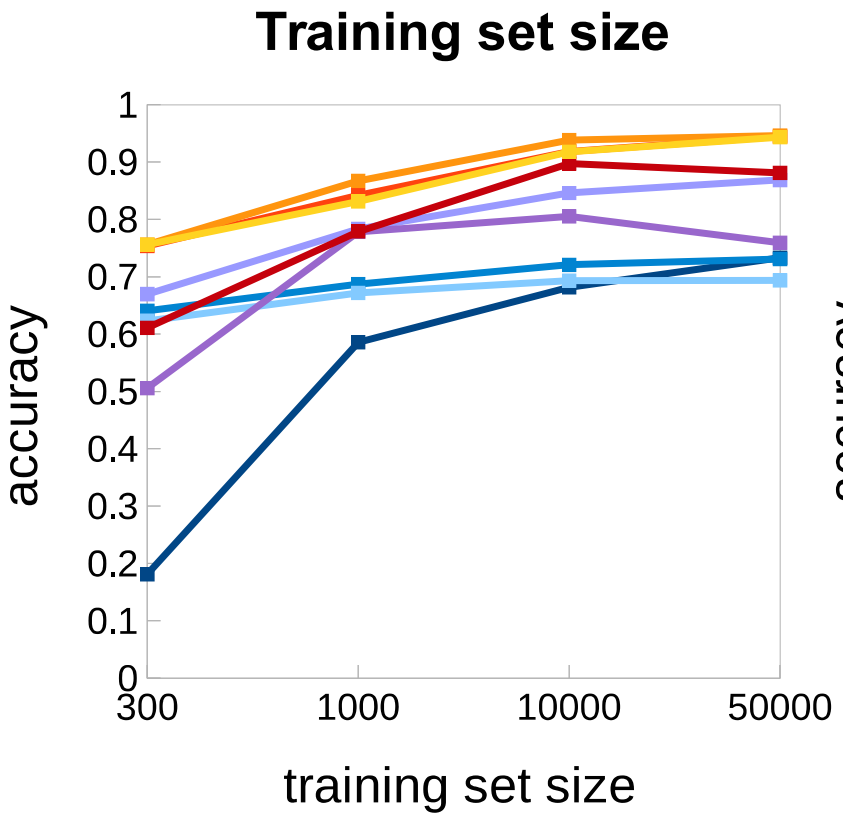
More powerful representation, especially benefiting large training sets

Downsize input representation, reducing overfitting difficulty of CCA

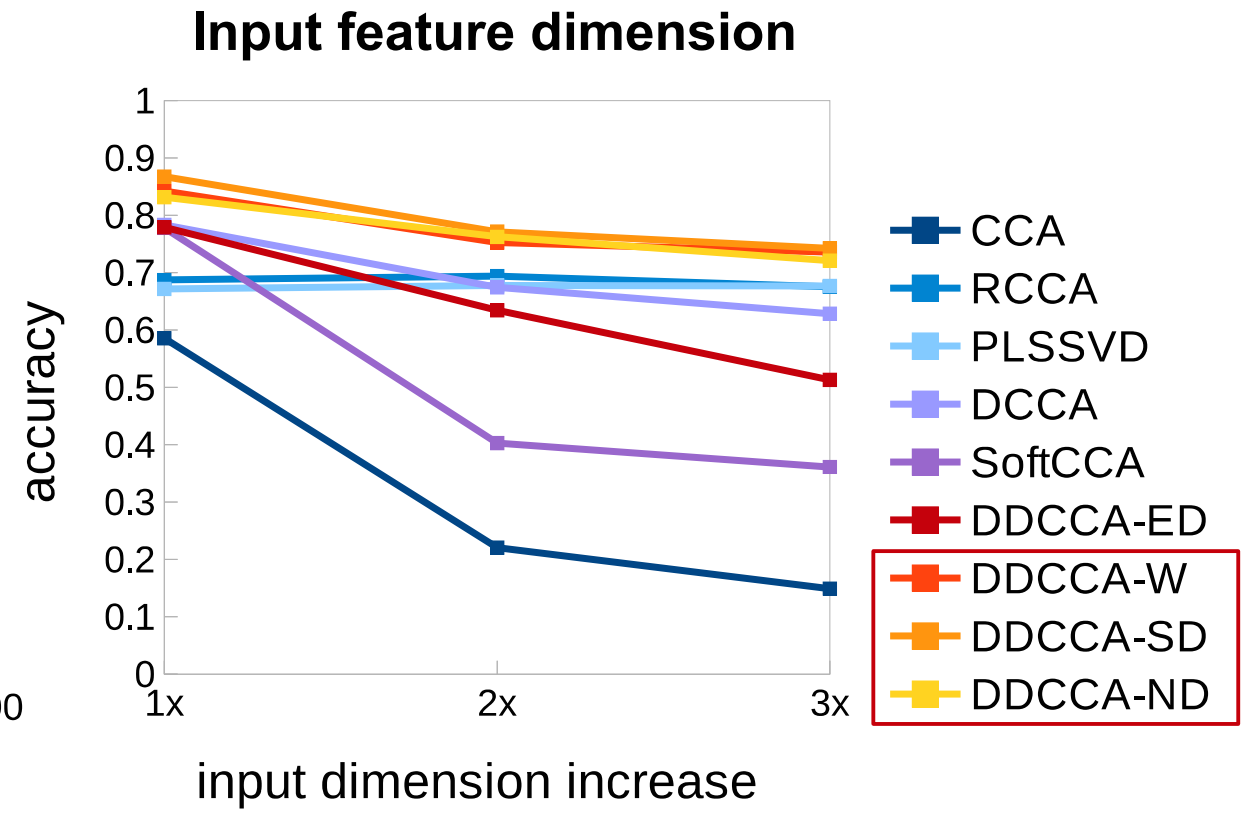
Improved robustness for small training set and HDLSS data

Experiments: MNIST Split

Cross-modal classification



# of components: 50

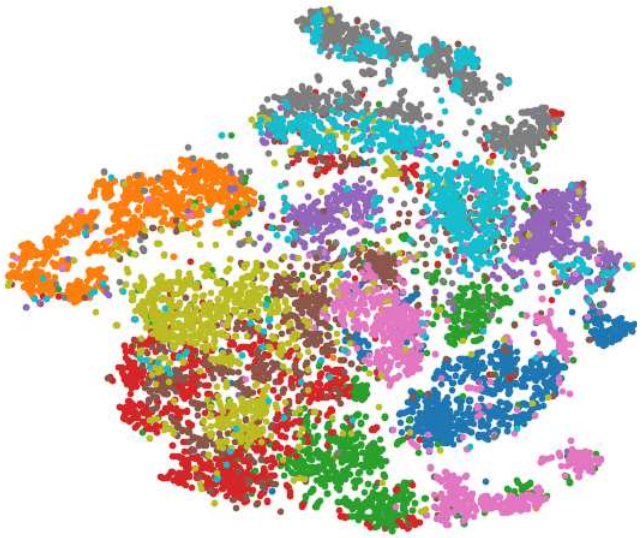


training set size: 1000
# of components: 50

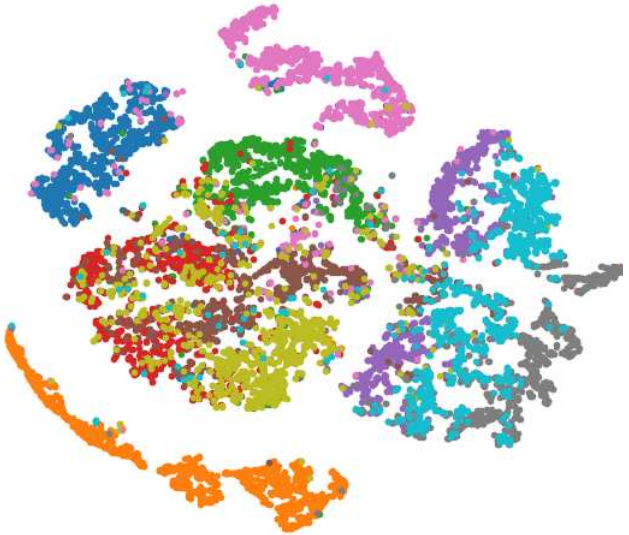
Robust to small training set size and HDLSS data

MNIST Split: Visualization with t-SNE

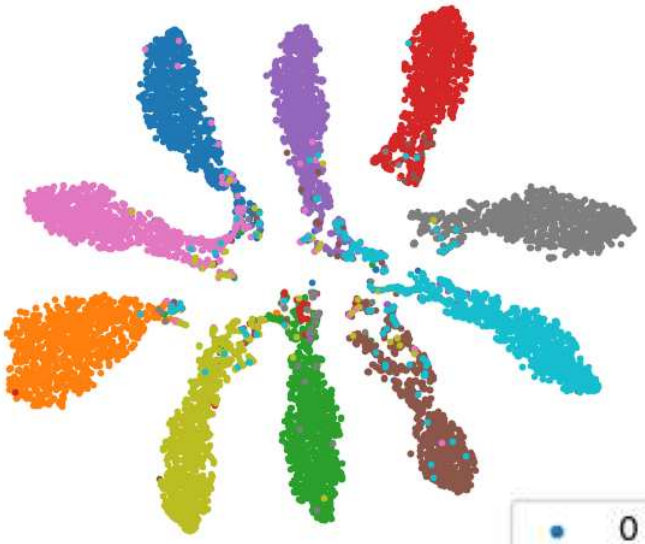
CCA

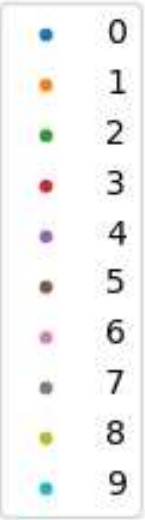


DCCA



DDCCA-W





Digits 0 to 9
training set size: 10,000
# of components: 50

Digits are better clustered with task-driven method

Carolina Breast Cancer Study: Cross-modal

Cross-modal classification accuracy

1003 patients

Image features:
VGG16 – output of 4th set of conv layers + mean pool → 512 D

Genomic features:
PAM50 – expression for 50 genes

Train/validation/test:
Random split ½, ¼, ¼

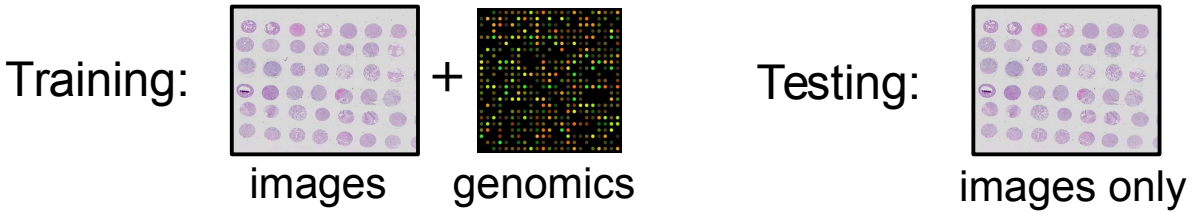
Train PAM50, Test Image

Method	Basal	ER	Grade
CCA	0.732 (0.010)	0.637 (0.008)	0.741 (0.005)
RCCA	0.815 (0.008)	0.811 (0.003)	0.877 (0.010)
PLS-SVD	0.650 (0.016)	0.656 (0.003)	0.797 (0.010)
DCCA	0.787 (0.010)	0.785 (0.011)	0.867 (0.012)
SoftCCA	0.780 (0.010)	0.769 (0.014)	0.848 (0.015)
DDCCA-ED	0.802 (0.015)	0.803 (0.029)	0.852 (0.011)
DDCCA-W	**0.820** (0.008)	**0.828** (0.006)	**0.917** (0.019)
DDCCA-SD	0.796 (0.004)	0.811 (0.004)	0.874 (0.019)
DDCCA-ND	0.766 (0.013)	0.805 (0.007)	0.878 (0.011)

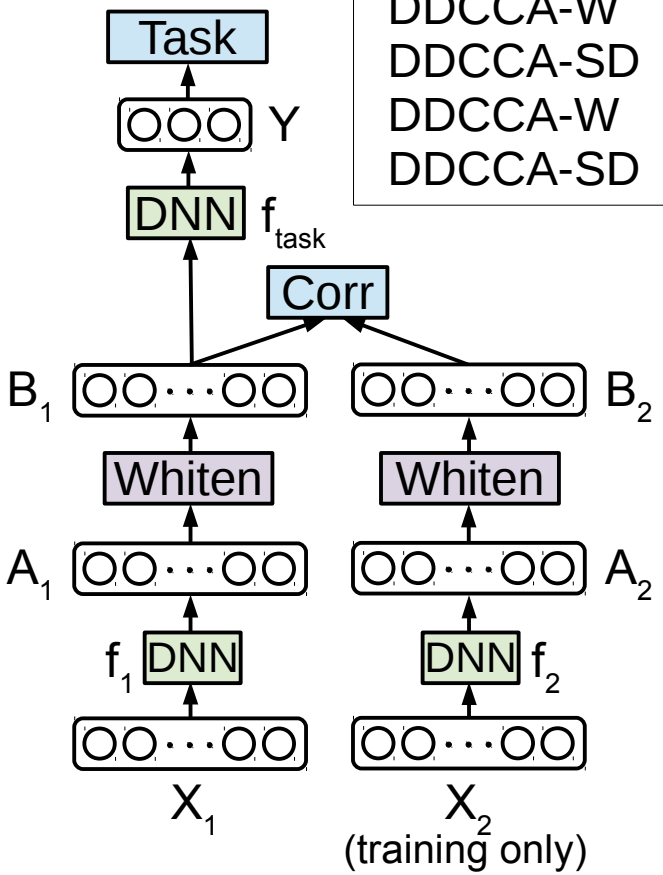
Train Image, Test PAM50

Method	Basal	ER	Grade
CCA	0.943 (0.005)	0.869 (0.006)	0.828 (0.011)
RCCA	0.978 (0.003)	0.905 (0.008)	0.836 (0.009)
PLS-SVD	0.912 (0.018)	0.865 (0.005)	0.794 (0.012)
DCCA	0.976 (0.005)	0.874 (0.010)	0.854 (0.015)
SoftCCA	0.978 (0.004)	0.897 (0.010)	0.843 (0.007)
DDCCA-ED	0.971 (0.009)	0.889 (0.010)	0.833 (0.010)
DDCCA-W	**0.983** (0.006)	**0.908** (0.009)	0.845 (0.005)
DDCCA-SD	0.978 (0.005)	**0.908** (0.009)	**0.848** (0.004)
DDCCA-ND	0.975 (0.004)	0.896 (0.005)	0.817 (0.005)

Carolina Breast Cancer Study: Regularization



Method	Training data	Basal	ER	Grade
Linear SVM	Image only	0.785 (0.004)	0.838 (0.003)	0.897 (0.006)
DNN	Image only	0.796 (0.007)	**0.852** (0.008)	0.907 (0.009)
DDCCA-W	Image+PAM50	0.827 (0.006)	0.839 (0.007)	**0.911** (0.012)
DDCCA-SD	Image+PAM50	0.820 (0.010)	0.826 (0.009)	0.859 (0.021)
DDCCA-W	Image+GE163	**0.840** (0.010)	0.838 (0.009)	0.910 (0.017)
DDCCA-SD	Image+GE163	0.812 (0.011)	0.815 (0.020)	0.891 (0.020)



Genomic features:

PAM50 – expression for 50 genes

GE163 – larger set of 163 genes

Large improvement for some tasks

Future Work

- Shared and individual representations
- Fine-tune CNN as one modality
- Data augmentation

Contributions to Computer Science

1) **Discriminative** representations for histology images using **dictionary learning** or **deep transfer learning**
2) **Multiple instance learning** methods for handling large, **heterogeneous** images with an **SVM** on any type of feature set or with a **CNN** for end-to-end training
3) A set of **multimodal** methods to find a shared space that is also **discriminative**
4) Techniques for **deep learning** on problems traditionally viewed as **“small data”**

Acknowledgements

Advisor: Marc Niethammer

Other committee members: Steve Marron, Chuck Perou, Steve Pizer, Alex Berg

Collaborators: Melissa Troester, Lindsay Williams, Joseph Geradts, David Eberhard, Nancy Thomas, Susan Wei, Jayson Miedema

Funding: Royster Society, Lineberger Comprehensive Cancer Center, CISMM (NIH)

Questions?

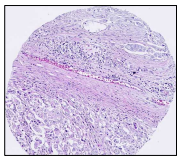
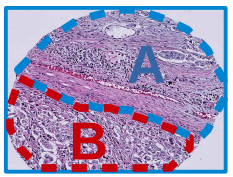


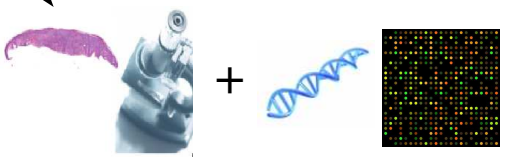
image features

tumor class

1) Representing histology images



2) Handling heterogeneous images



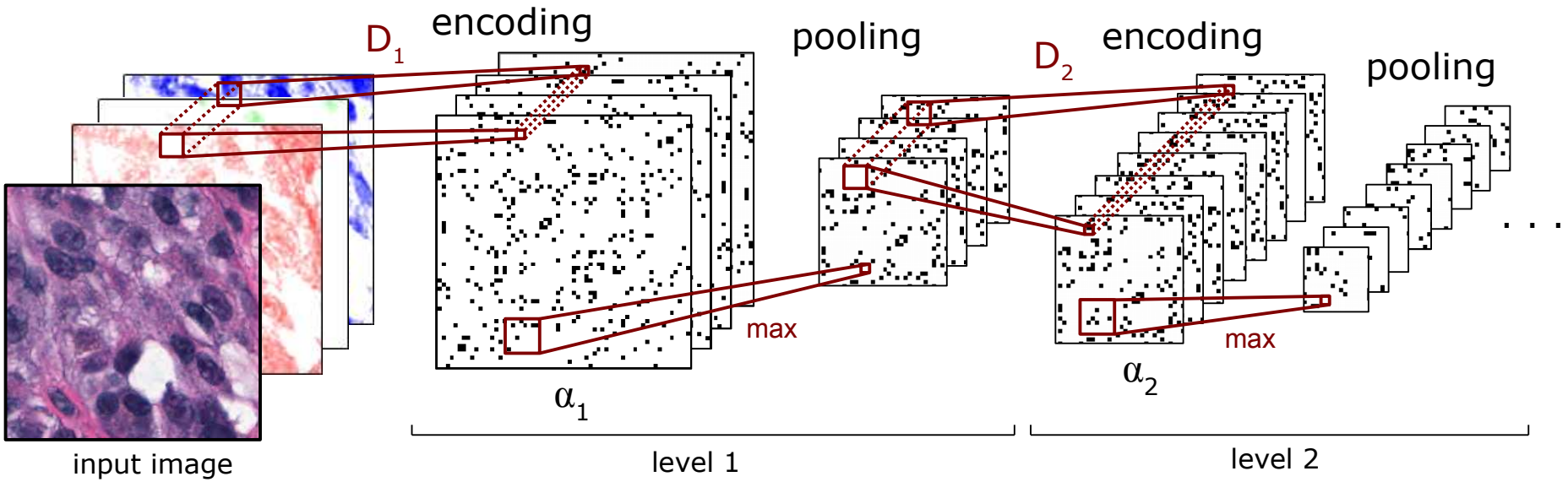
3) Combining imaging & genomics

Extra slides

Publications

- **ISBI 2015**: Hierarchical task-driven feature learning for tumor histology
- **MICCAI 2018**: Multiple instance learning for heterogeneous images: training a CNN for histopathology
- **npj Breast Cancer 2018**: Image analysis with deep learning to predict breast cancer grade, ER status, histologic subtype, and intrinsic subtype

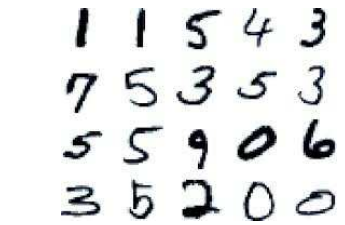
Method 1: Task-driven Dictionary Learning



Task-Driven Dictionary Learning

Simplification: classify individual patches

Train classifier by minimizing expected loss
Optimize over classifier and dictionary (Mairal, 2012)



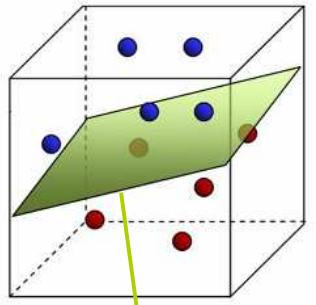
applied to small images e.g., MNIST

$$f(y, x, \alpha) = E[loss(y, w, \alpha(x, D))] + \frac{\nu}{2}\|w\|_2^2$$

loss — labels — classifier — sparse encodings — regularization

e.g., logistic

$$\log\left(1 + e^{-y w^T \alpha(x, D)}\right)$$



w
separating hyperplane (classifier)

Task-Driven Dictionary Learning

$$f(y, x, \alpha) = E[loss(y, w, \alpha(x, D))] + \frac{\upsilon}{2}\|w\|_2^2$$

labels
classifier
sparse encodings
patches
dictionary

Solution: stochastic gradient descent

Initialize dictionary D with unsupervised learning

Initialize classifier w with logistic regression on set of patch encodings

Repeat until convergence:

Select set of labels y and patches x from training set

Compute sparse encodings α

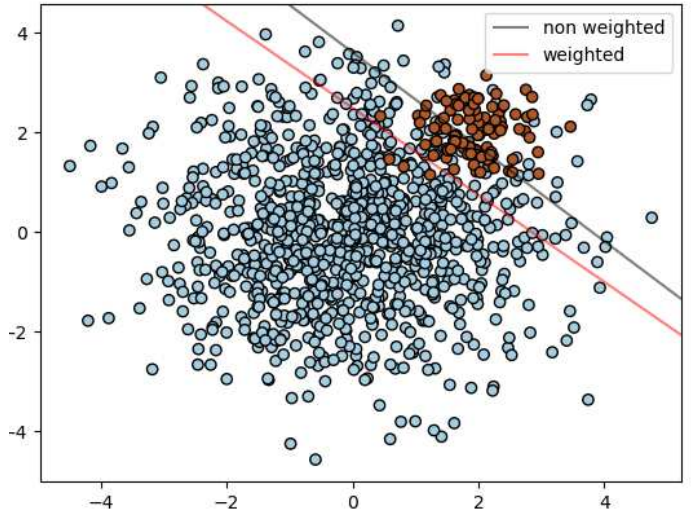
Update classifier w using $\nabla_w f(y,x,\alpha)$

Update dictionary D using $\nabla_D f(y,x,\alpha)$

Method 1: Pre-trained CNN

Sample Weighting

Sample weighting for unbalanced data



http://scikit-learn.org/stable/auto_examples/svm/plot_separating_hyperplane_unbalanced.html

$$w_c = \frac{1}{\#\text{ samples of class } c}$$

Sample weighting by grade

$$w_{c,g} = \frac{1}{\#\text{ samples of class } c, \text{ grade } g}$$

Method 1: Quantile Aggregation

Classification Accuracy

	Sensitivity (%)	Specificity (%)	Accuracy (%)	Kappa
Grade low-int vs high			82	0.64
ER status	84	72	84	0.64
Basal vs non-Basal	78	73	77	0.47
ROR-PT low-med vs high risk	79	74	76	0.47
Histologic subtype ductal vs lobular	71	96	94	0.66

CBCS data set: 859 patient samples, 4 cores/patient
Random division into 2/3 training, 1/3 test

Method 1: Quantile Aggregation

Inter-rater Agreement

Grade:

	Accuracy	Kappa
Image analysis	82%	0.64
Agreement between two pathologists	89%	0.78
Reported by other groups		0.6-0.7, 0.5

ER Status Kappa:

Image analysis	0.64
Centralized pathology and SEER classifications	0.70
Different IHC antibodies	0.6-0.8
Medical records and staining of tissues	0.62

PAM50 Subtype Accuracy:

Image analysis	77%
IHC vs. RNA-based for Basal	90%
IHC vs. RNA-based for Luminal A	77%

Method 3: Fine-tune CNN

Classification Accuracy (Multi-class)

	Mean	**Quantile**
Histologic subtype ductal vs lobular	0.931 (0.004)	0.952 (0.003)
ER status	0.833 (0.008)	0.841 (0.006)
Grade	0.680 (0.003)	0.676 (0.006)
ROR-PT	0.595 (0.003)	0.582 (0.008)
Intrinsic subtype	0.548 (0.006)	0.544 (0.003)

Background: Canonical Correlation Analysis

Given: input data $X_1 \in \Re^{d_1 x n}$, $X_2 \in \Re^{d_2 x n}$ (mean centered)
Compute covariance matrices:

$$\Sigma_1 = \frac{1}{n-1} X_1 X_1^T \qquad \Sigma_2 = \frac{1}{n-1} X_2 X_2^T \qquad \Sigma_{12} = \frac{1}{n-1} X_1 X_2^T$$

Goal: find projections w_1 and w_2 such that the data are maximally correlated

$$\underset{w_1, w_2}{\text{argmax}}\, \text{corr}\left(w_1^T X_1, w_2^T X_2\right) = \underset{w_1, w_2}{\text{argmax}} \frac{w_1^T \Sigma_{12} w_2}{\sqrt{w_1^T \Sigma_1 w_1 w_2^T \Sigma_2 w_2}}$$

Alternatively, constrain projections to have unit variance:

maximize: $w_1^T \Sigma_{12} w_2$ covariance

subject to: $w_1^T \Sigma_1 w_1 = w_2^T \Sigma_2 w_2 = 1$ unit variance

Background: Canonical Correlation Analysis

Find **multiple pairs** $(w_1^{(i)}, w_2^{(i)})$ such that

$$(w_1^{(i)})^T \Sigma_1 w_1^{(j)} = (w_2^{(i)})^T \Sigma_2 w_2^{(j)} = 0 \quad \text{for i} \neq \text{j}$$

orthogonal projections

Let $W_1 = [w_1^{(1)}, \ldots, w_1^{(k)}]$ and $W_2 = [w_2^{(1)}, \ldots, w_2^{(k)}]$

maximize: $\operatorname{tr}(W_1^T \Sigma_{12} W_2)$

subject to: $W_1^T \Sigma_1 W_1 = W_2^T \Sigma_2 W_2 = I$

Solution:

Let $T = \Sigma_1^{-1/2} \Sigma_{12} \Sigma_2^{-1/2}$

SVD $T = U_1 \operatorname{diag}(\sigma) U_2^T$

Compute W_1 and W_2 from top k singular values of T:

$$W_1 = \Sigma_1^{-1/2} U_1^{(1:k)} \qquad W_2 = \Sigma_2^{-1/2} U_2^{(1:k)}$$

Background: Deep CCA

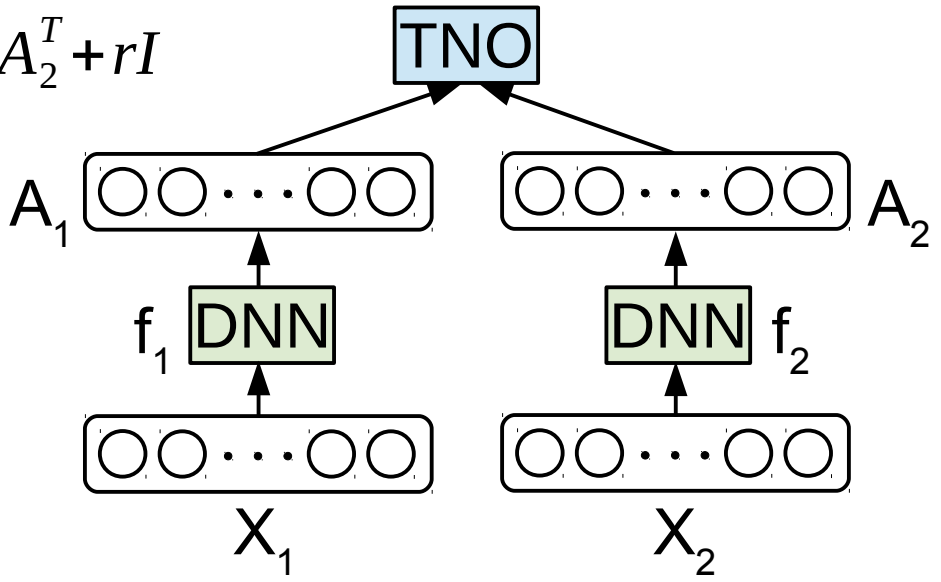
Andrew et al., ICML, 2013

Given: $A_1 = f_1(X_{1,}\Theta_1),\ A_2 = f_2(X_{2,}\Theta_2),\ A_1 \in \Re^{d_o x n},\ A_2 \in \Re^{d_o x n}$

Compute covariance matrices:

$$\Sigma_1 = \frac{1}{n-1} A_1 A_1^T + rI \quad \Sigma_2 = \frac{1}{n-1} A_2 A_2^T + rI$$

$$\Sigma_{12} = \frac{1}{n-1} A_1 A_2^T$$



Same objective:

maximize: $tr(W_1^T \Sigma_{12} W_2)$

subject to: $W_1^T \Sigma_1 W_1 = W_2^T \Sigma_2 W_2 = I$

Trick: when k=d_o (# of components = # of features in A)

$$\sum_{i=1}^{k} \mathrm{corr}\left((w_1^{(i)})^T A_{1,} (w_2^{(i)})^T A_2 \right) = \sum_{i=1}^{k} \sigma_i \quad \text{equivalent to} \quad \|T\|_{\mathrm{tr}} = \mathrm{tr}\left(T^T T\right)^{1/2}$$

Solution: maximize $||T||_{tr}$ (trace norm objective)

$$T = \Sigma_1^{-1/2} \Sigma_{12} \Sigma_2^{-1/2}$$
$$T = U_1 \mathrm{diag}(\sigma) U_2^T$$

Compute linear CCA after DNN optimization complete

Task-driven Deep CCA

Problem: maximizing sum correlation does not always result in improved cross-modal classification accuracy

Solution: add task-driven objective (e.g., for classification)

$$\mathcal{L}_{\text{task}} + \lambda\, \mathcal{L}_{\text{CCA}}$$

CCA objective:

covariance

$$\underset{W_1, W_2, \Theta_1, \Theta_2}{\operatorname{argmax}} \operatorname{tr}\left(W_1^T \Sigma_{12} W_2\right) \quad \text{equivalent to} \quad \underset{W_1, W_2, \Theta_1, \Theta_2}{\operatorname{argmin}} \left\| W_1^T A_1 - W_2^T A_2 \right\|_F^2$$

ℓ_2 distance

separable across batches

when $\left(w_1^{(i)}\right)^T \Sigma_1 w_1^{(i)} = \left(w_2^{(i)}\right)^T \Sigma_2 w_2^{(i)} = 1$ for all i

unit variance

Challenge: need to compute CCA projection in network

Batch Normalization

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

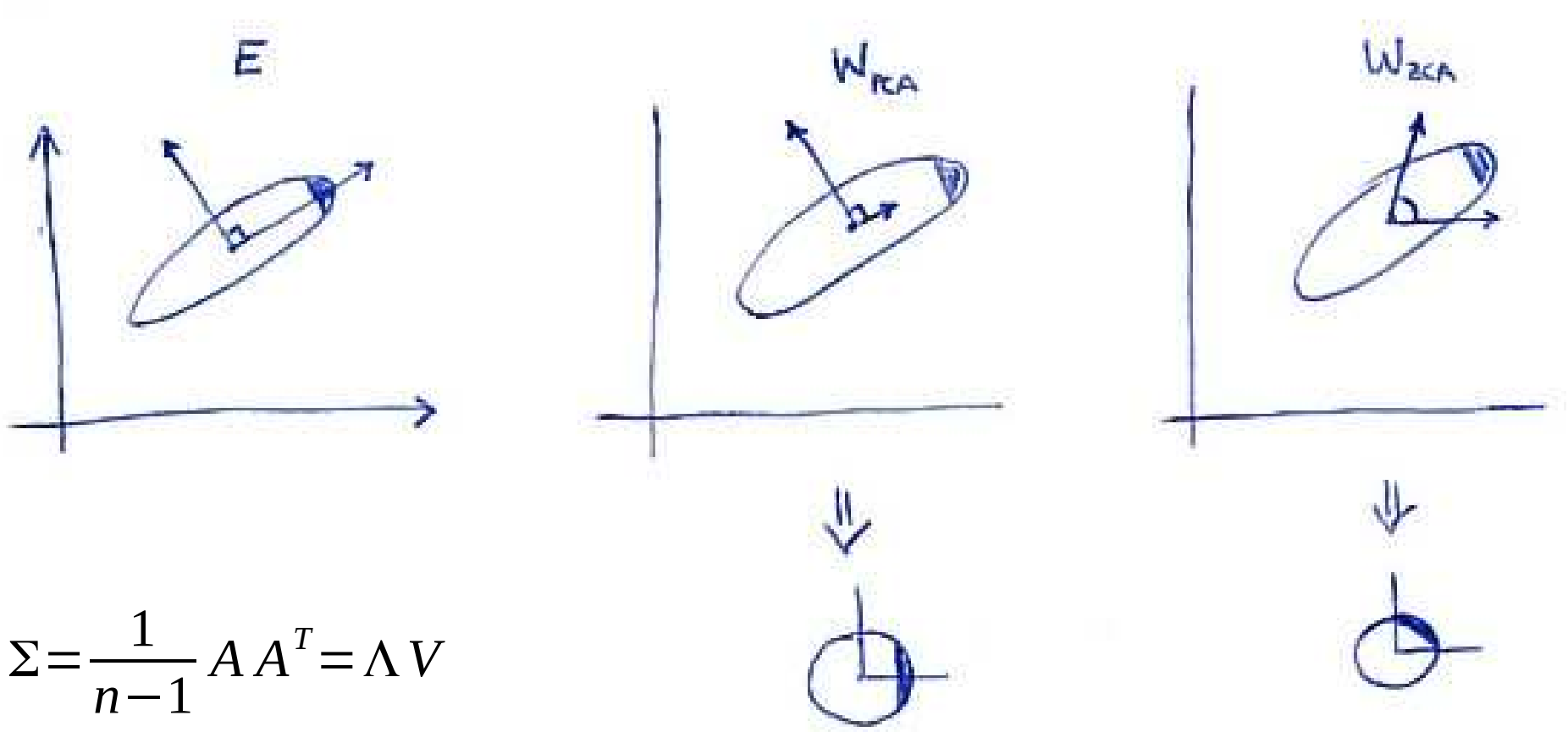
$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

ZCA Whitening

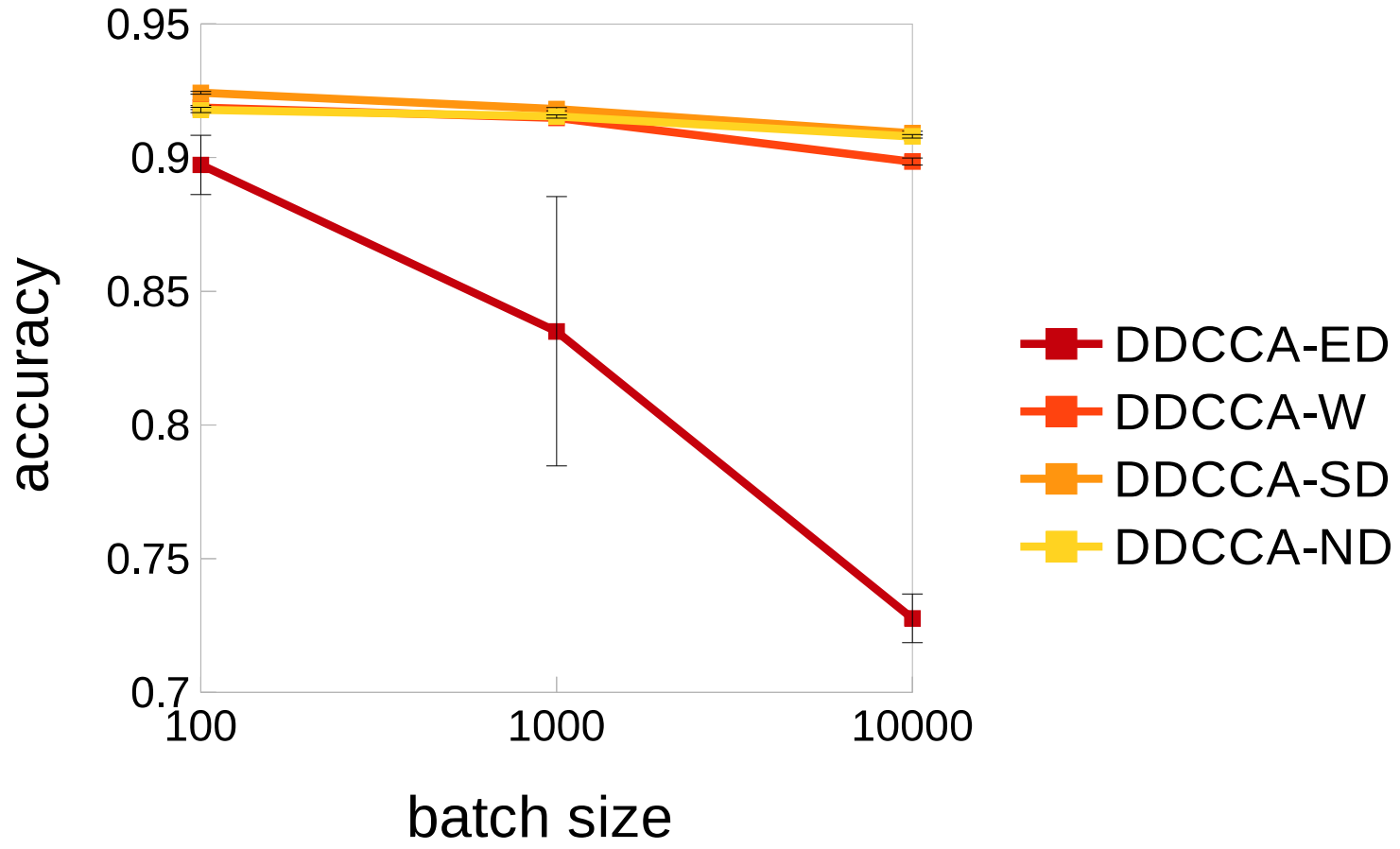


$$\Sigma = \frac{1}{n-1} A A^T = \Lambda V$$

$$B = V \Lambda^{-1/2} V^T A$$

https://stats.stackexchange.com/questions/117427/what-is-the-difference-between-zca-whitening-and-pca-whitening

Batch Size


0.95
0.9
0.85
0.8
0.75
0.7
accuracy
100
1000
10000
batch size
DDCCA-ED
DDCCA-W
DDCCA-SD
DDCCA-ND

Visualization with t-SNE

Original image features

Regularization with DDCCA-W