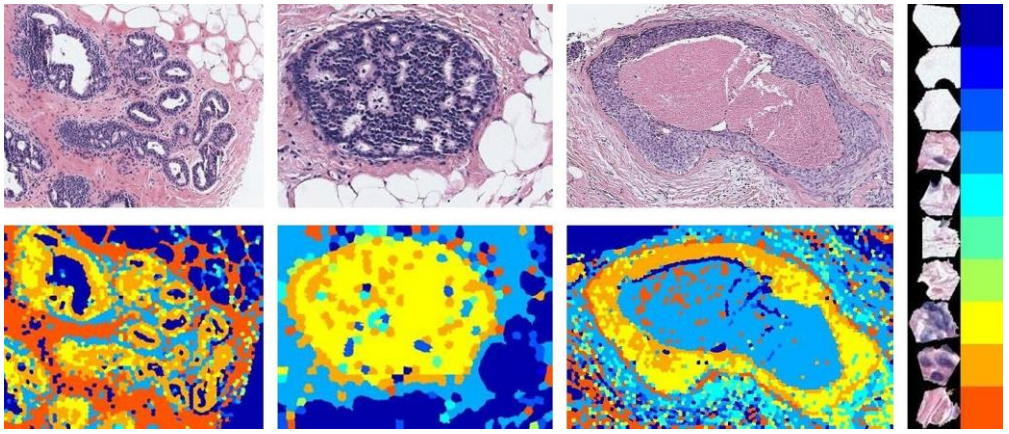
Seattle Children's
HOSPITAL • RESEARCH • FOUNDATION

CRANIOFACIAL

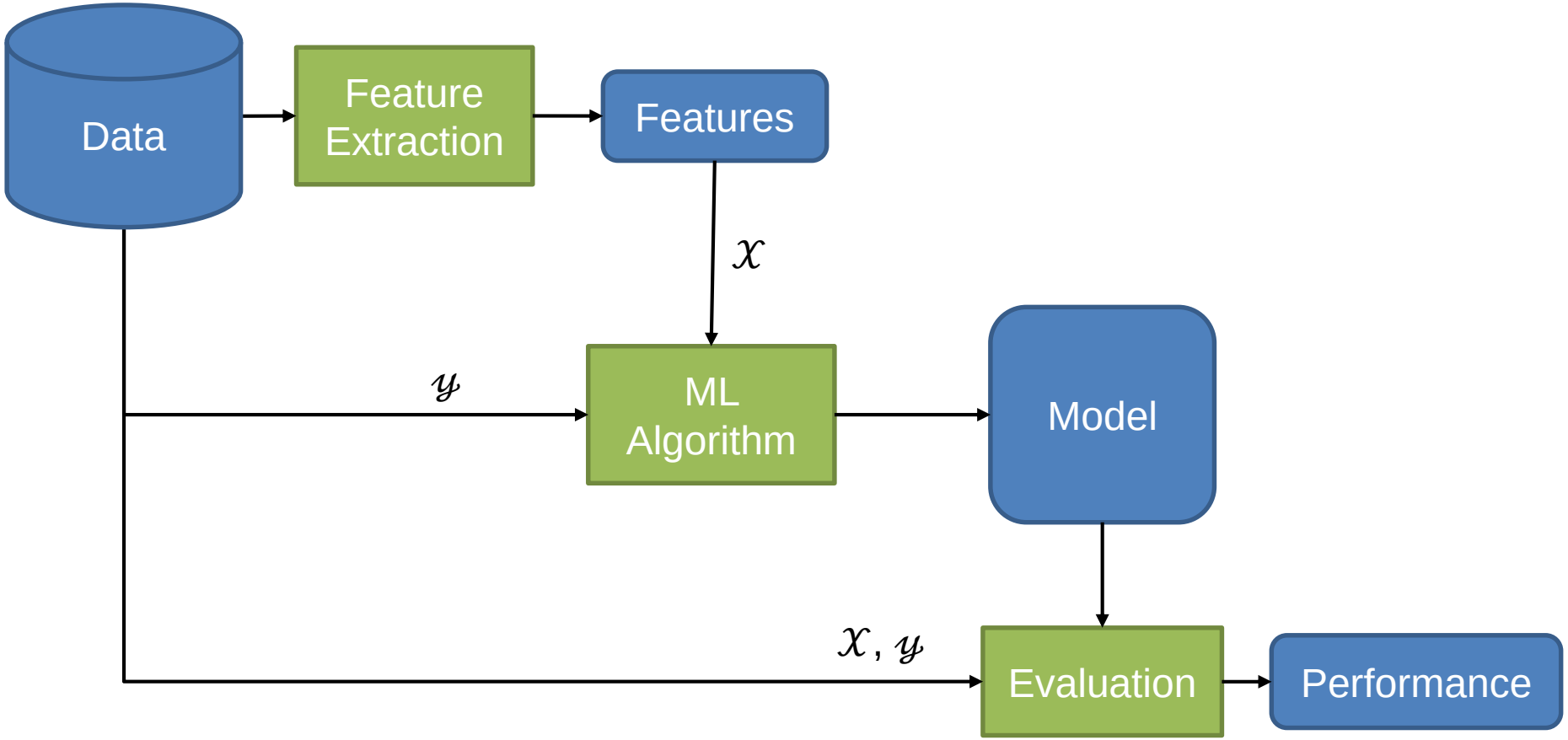
Machine Learning
with Applications in Biomedical Imaging

Ezgi Mercan



Machine Learning

General ML Pipeline

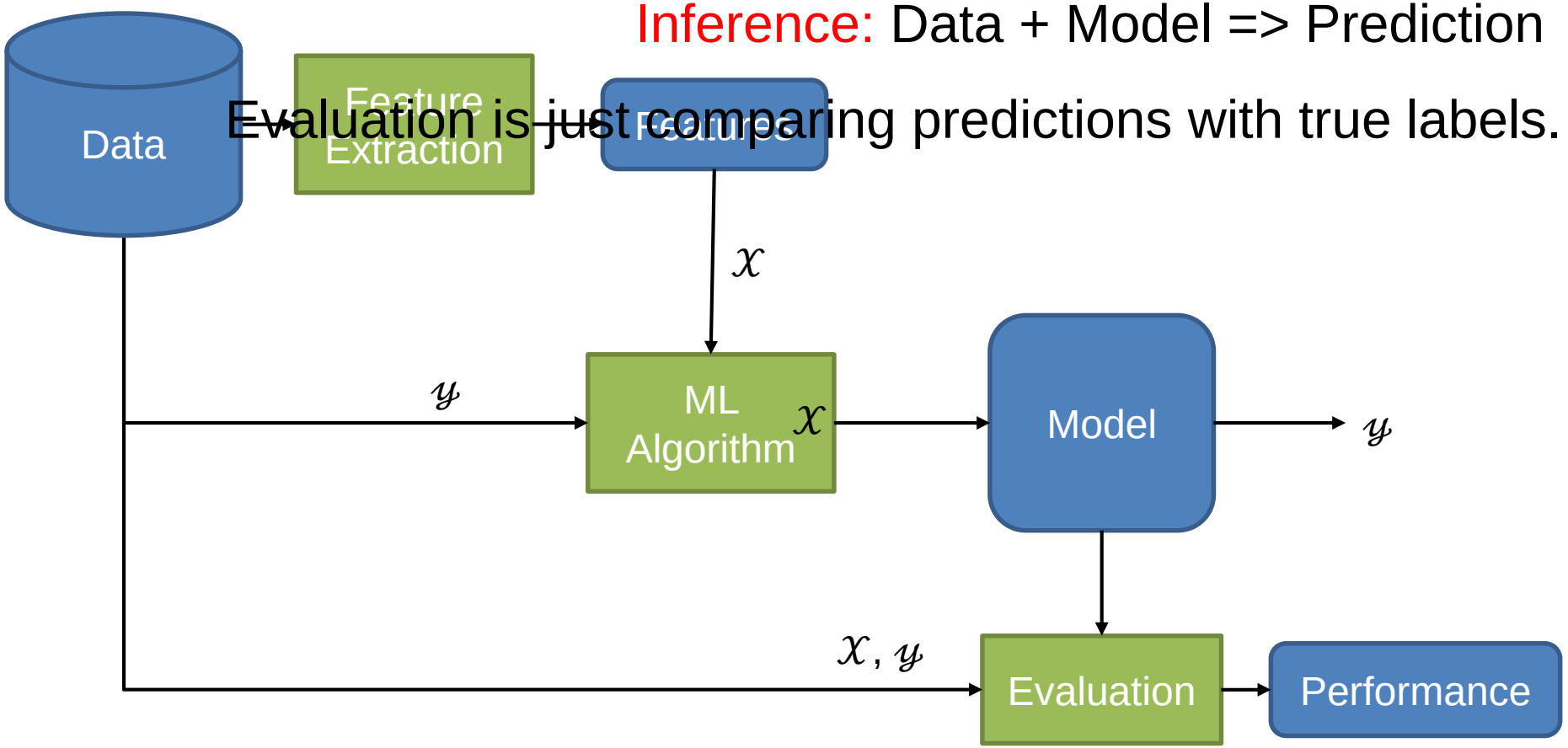


Training and Inference

Training: Data (+Label) => Model

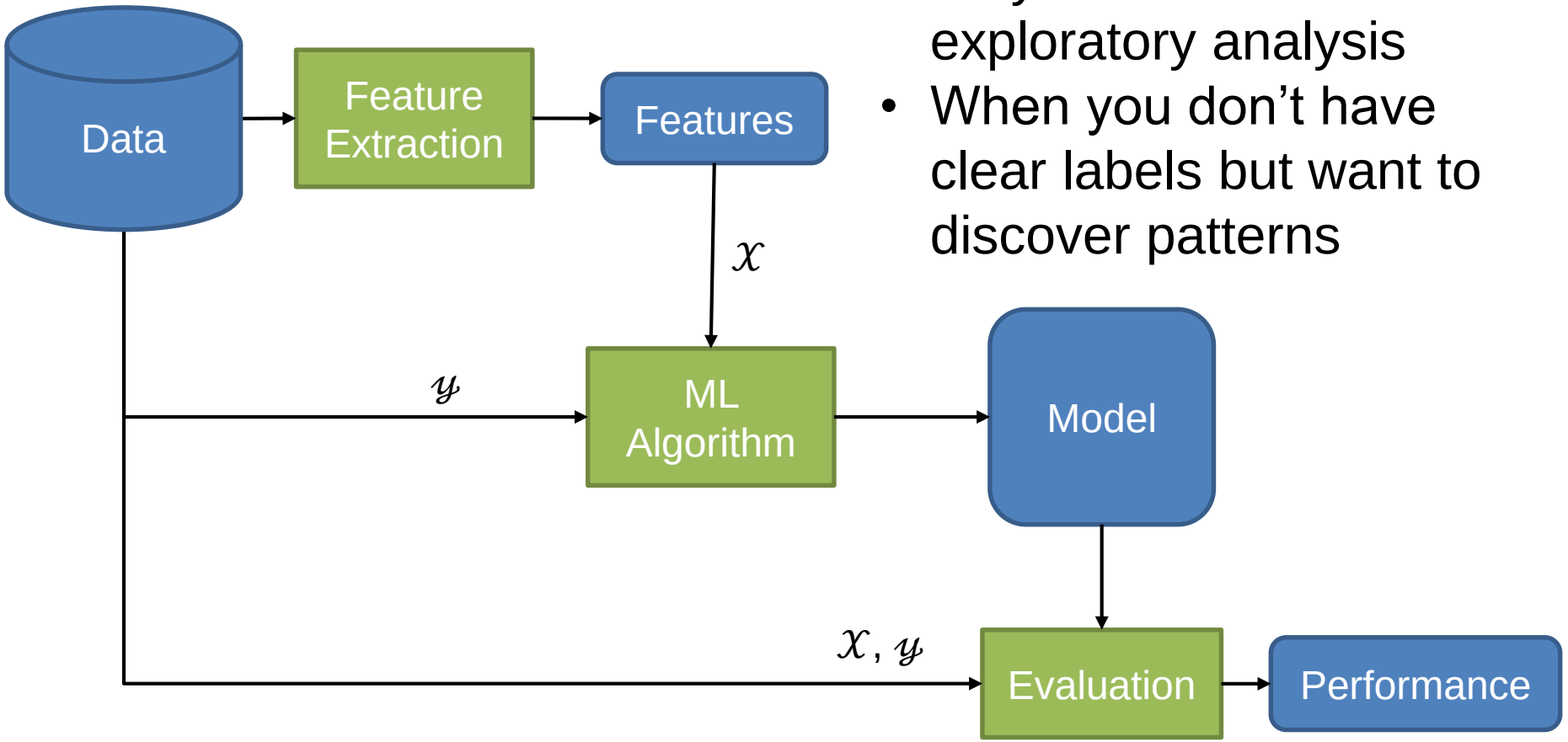
Inference: Data + Model => Prediction

Evaluation is just comparing predictions with true labels.



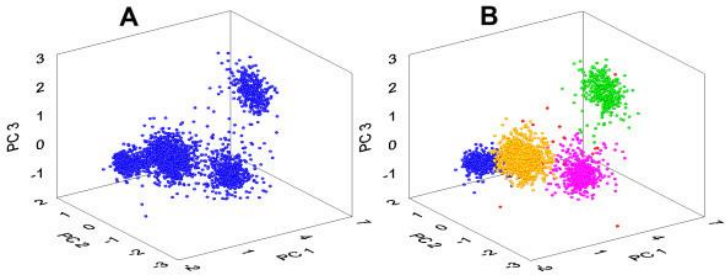
Unsupervised ML

- Very useful for exploratory analysis
- When you don't have clear labels but want to discover patterns



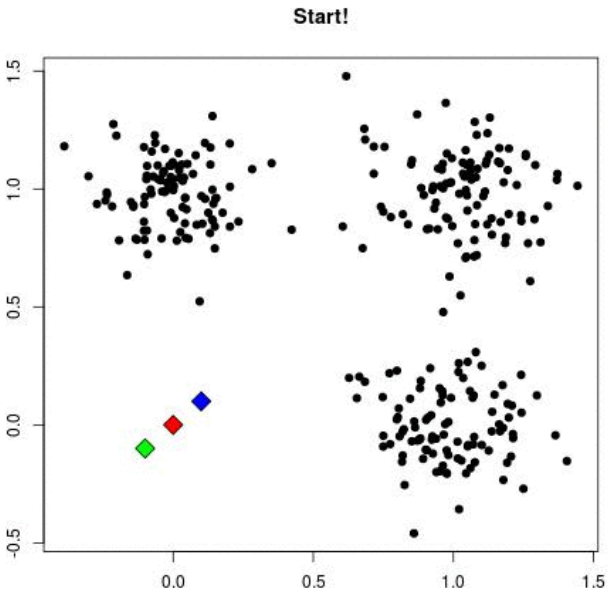
k-means Clustering

- A cluster: Centroid (+size)



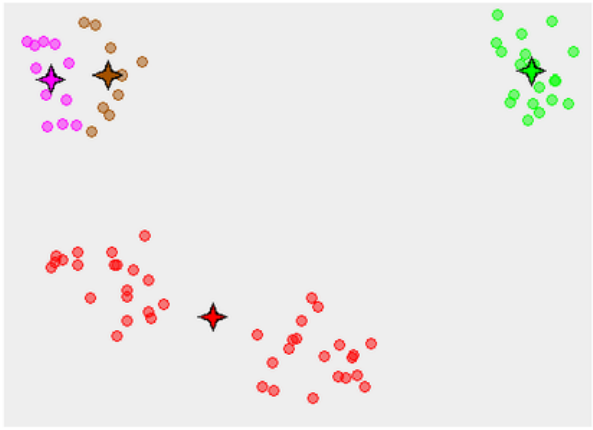
- Basic algorithm
  - Initialize cluster center

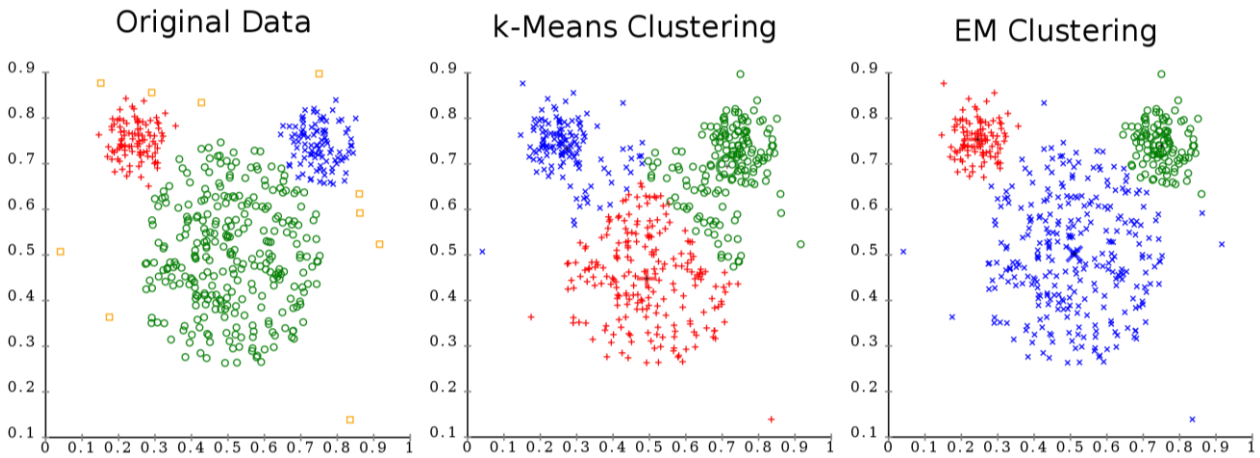
  Until convergence
  - Assign each sample to the cluster
  - Update cluster centers



k-means Clustering

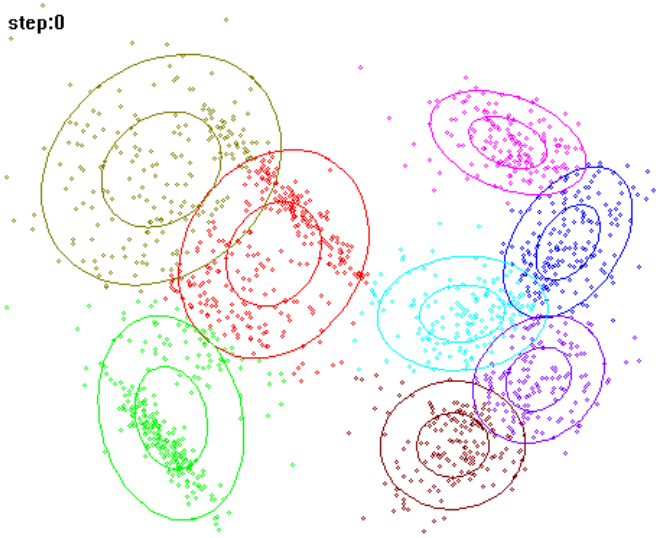
- You need to know k
  - There are tests you can use
- May converge to a local minimum
  - There are modifications that runs several random initiations.
- The clusters are *spherical* and *similar-sized.*





Mixture Models

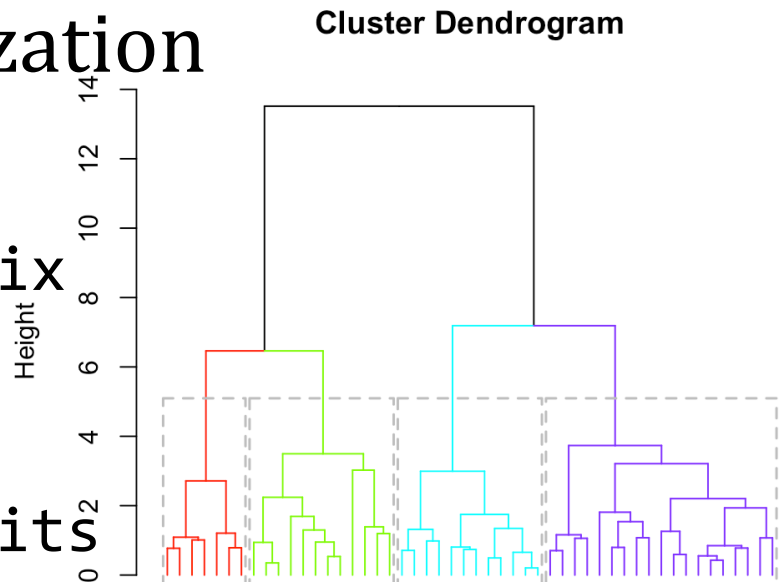
- Clusters are not fixed sized



  - Cluster: (μ, σ)
- *soft assignments*: Each samp
  is a "mixture" of clusters.
  - Pick the highest probability
- Algorithm:
  - `Expectation-Maximization`
- Still susceptible to initialization and local minima.
- You still need to "know" *k*.

Hierarchical Clustering

- Different "distance metrics"
- You don't need to "know" *k*
- Agglomerative (bottom-up) or divisive (top-down)
- Dendrograms for visualization
- Algorithm (bottom-up)
  - Calculate distance matrix
  - Each sample is its own cluster
  - Merge 2 clusters that fits the linkage criteria*

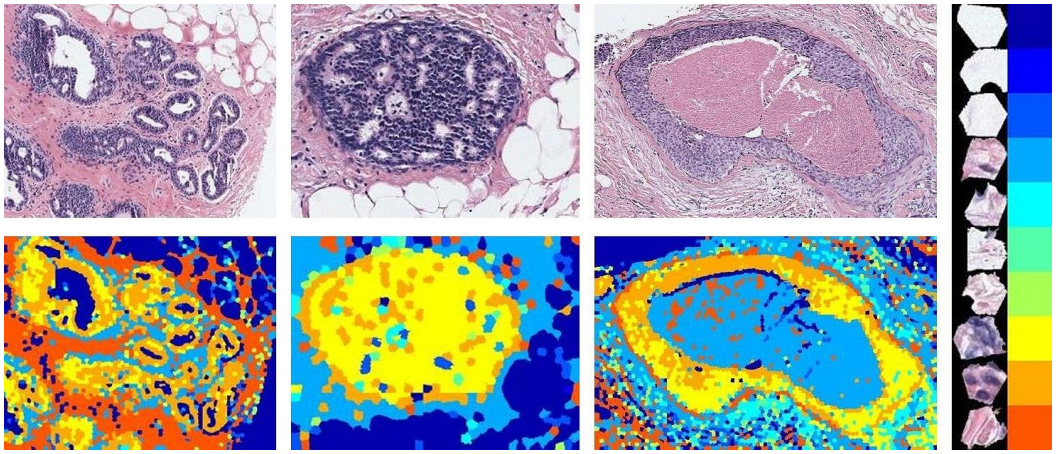


Hierarchical Clustering

- Define distance between any two samples.
- Calculation of pair-wise distances (distance matrix)
  - complexity nightmare, inefficient for large data
- Linkage criteria:
  - Single: combine two clusters that contain closest pair of samples
  - Complete: combine two clusters that contain the closest-farthest pair of samples
- Decide where to "cut" the dendrogram => k

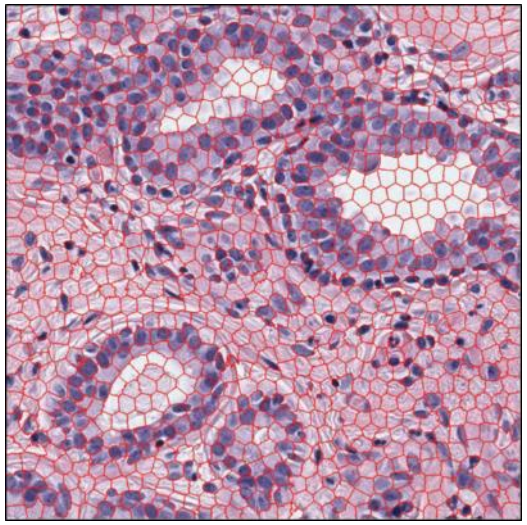
Histopathology Example

- Diagnosis depends on arrangement of different tissue types in the image.
- Problem: discover "different" tissue-types in a data-driven way.

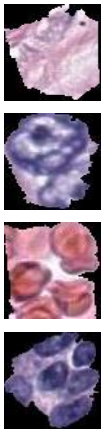


Histopathology Example

Image

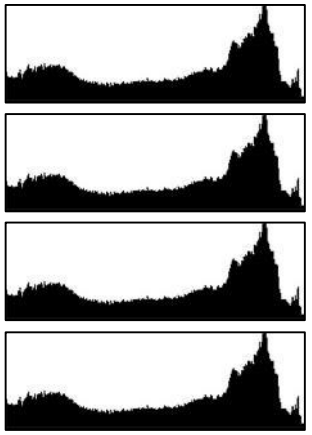


superpixel segmentation →



feature extraction →

color and texture histograms



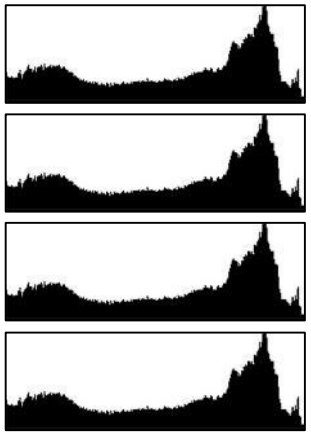
Raw Data

X

- Superpixels reduce dimensionality by 3000 pixels to 1 superpixel.
- Each superpixel cluster can be identified as a biologically meaningful building block of the tissue.

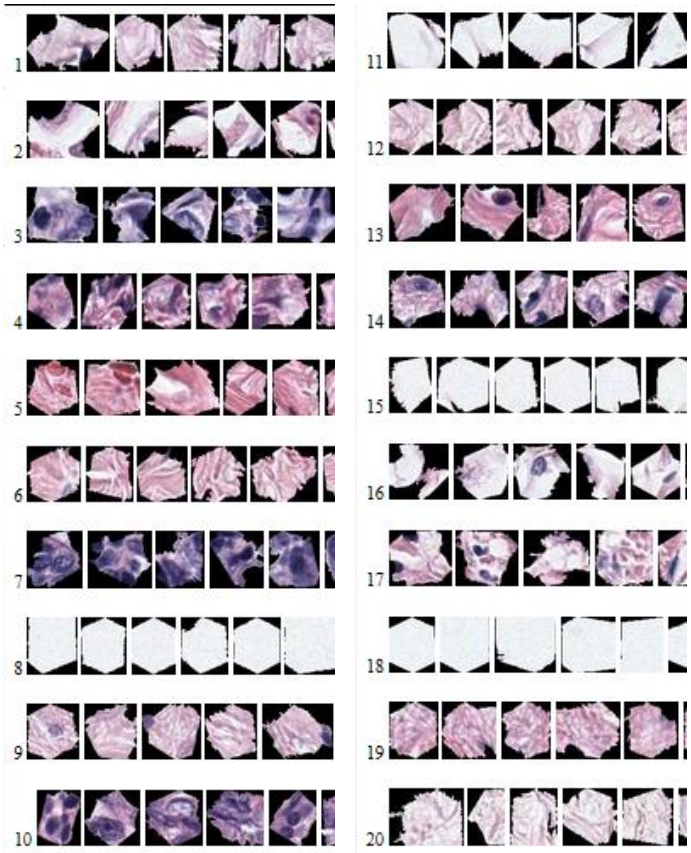
Histopathology Example

color and texture histograms



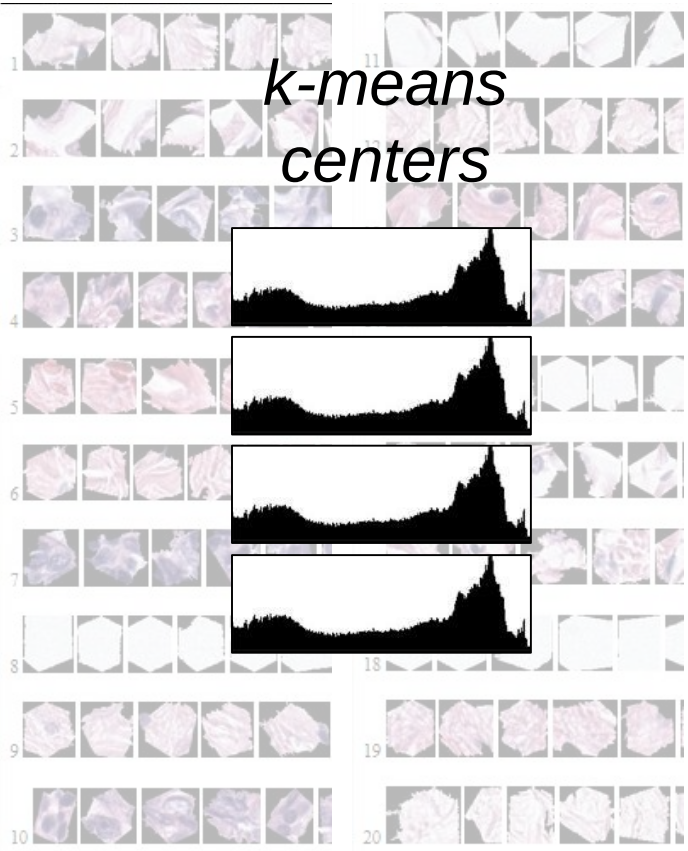
x

k-means clustering (k=200)



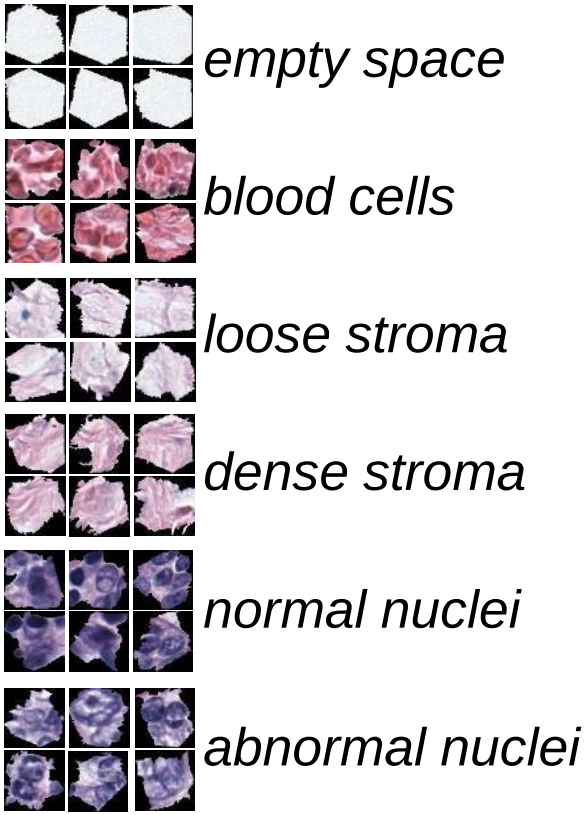
y

Histopathology Example



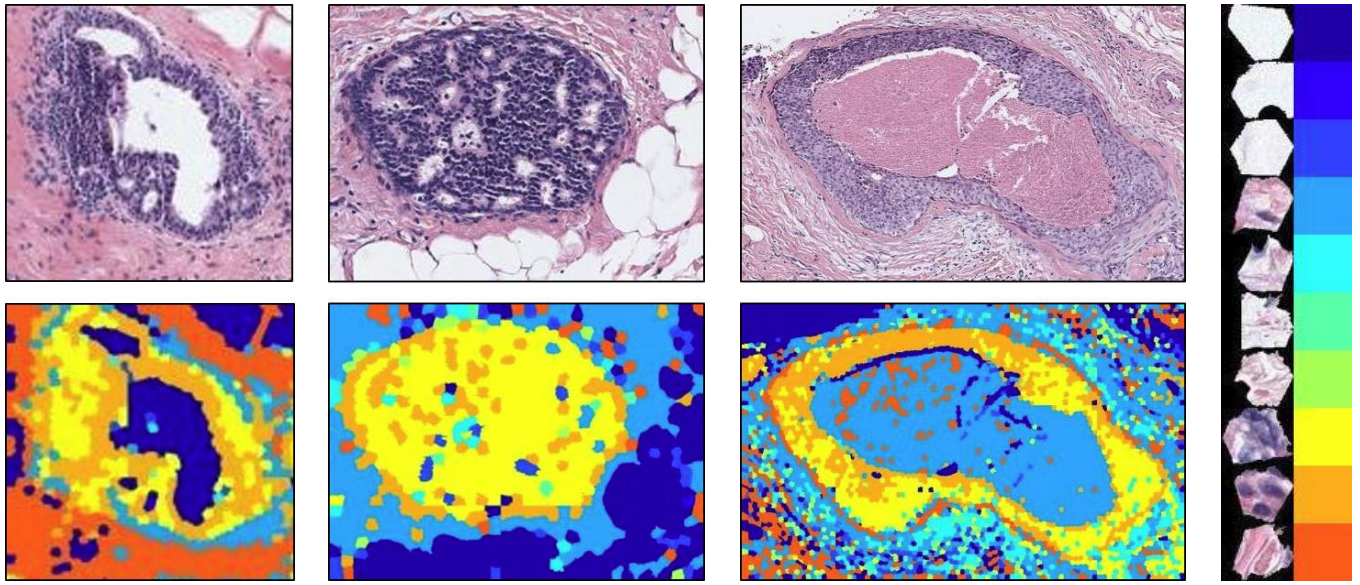
x

Hierarchical clustering



y

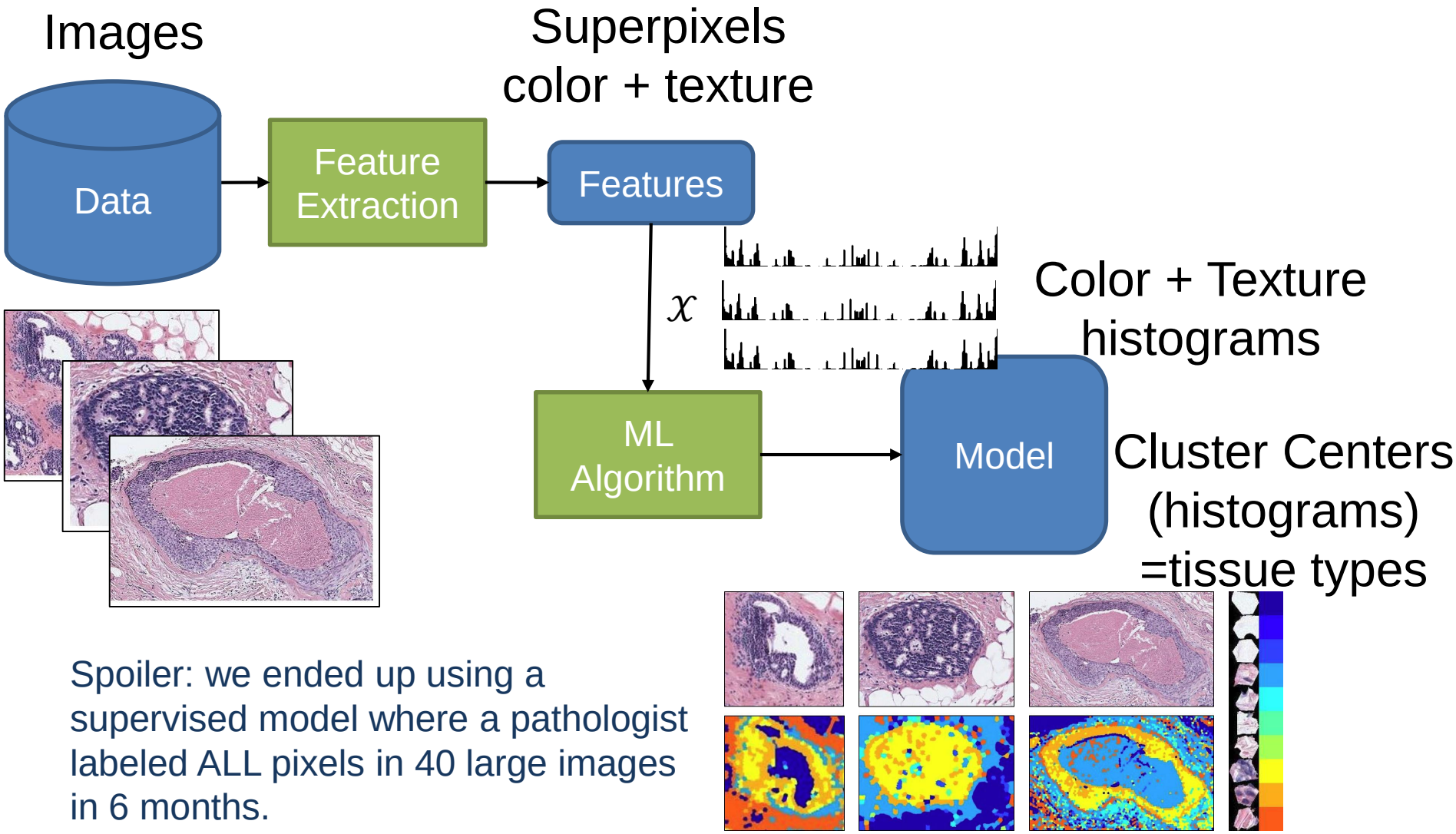
Histopathology Example



benign *atypia* *DCIS*

- Patterns emerge when we label the superpixels in an unsupervised manner.

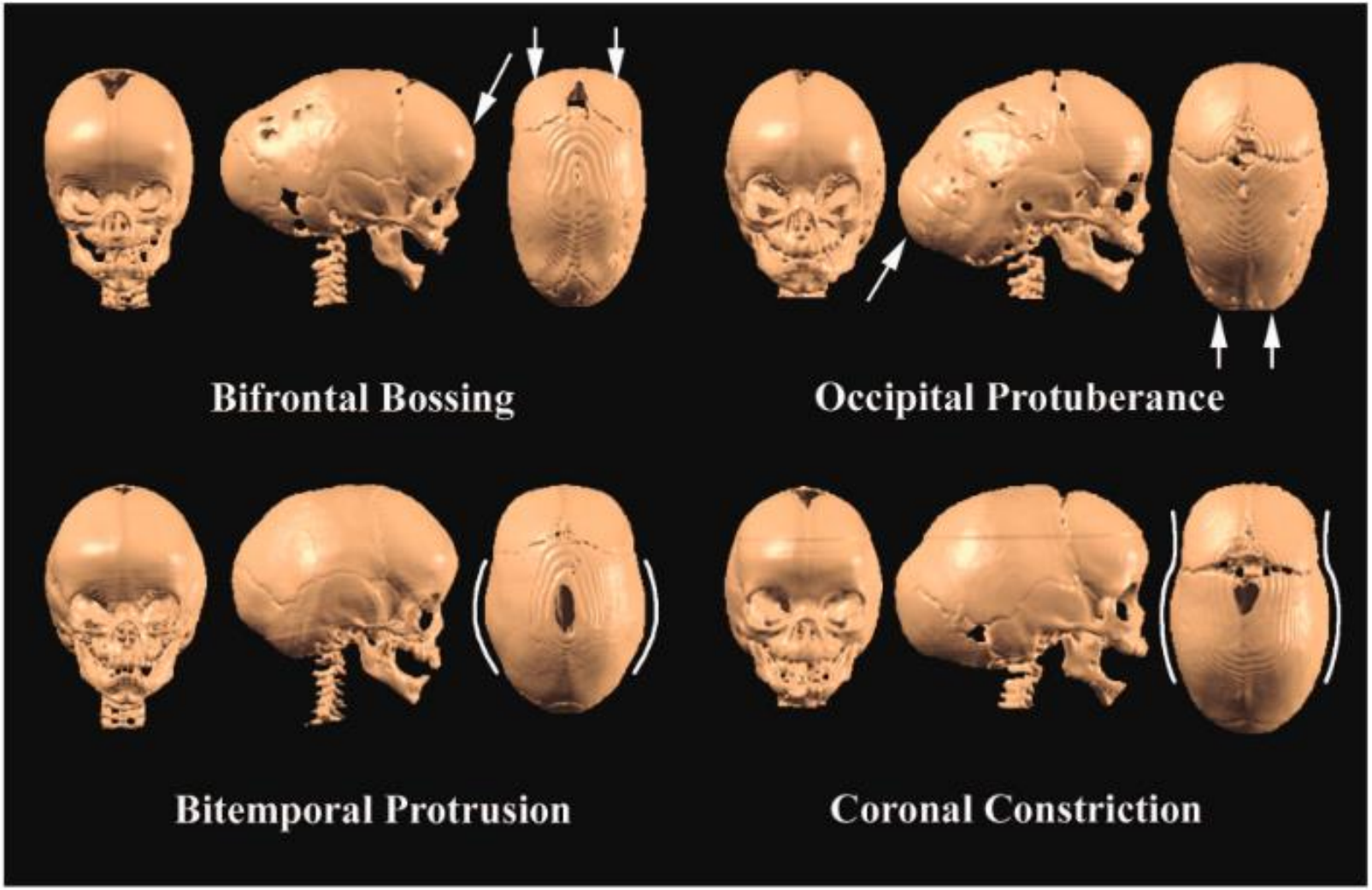
Histopathology Example



Spoiler: we ended up using a supervised model where a pathologist labeled ALL pixels in 40 large images in 6 months.

Craniofacial Example

- Sagittal craniosynostosis subtypes

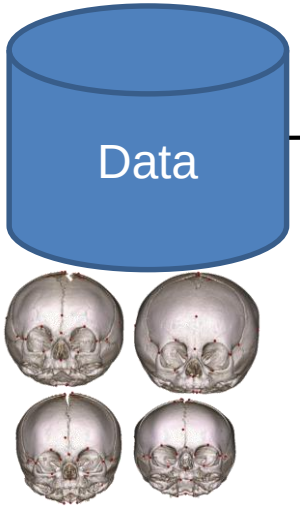


Craniofacial Example

CT Scans

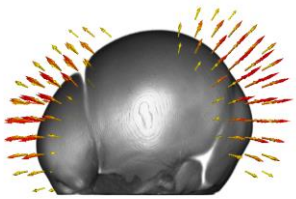
Warp to a template
Warp Fields

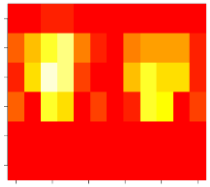
Angle Histograms

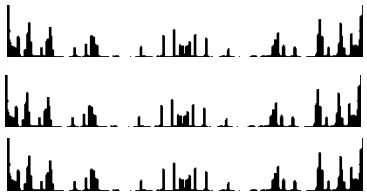


Feature Extraction

Features



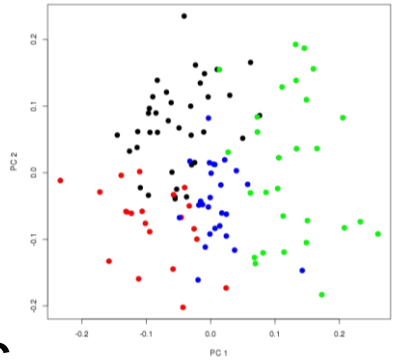




x

ML Algorithm

Model



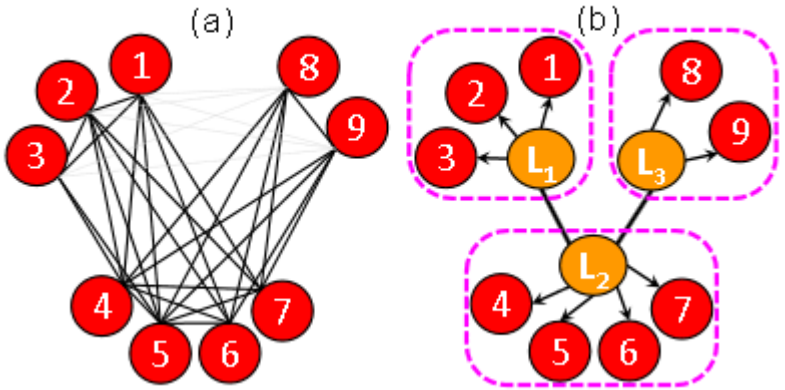
Hierarchical Clustering

Sagittal CS subtypes

We tried to correlate the discovered subtypes to clinician reported phenotypes.

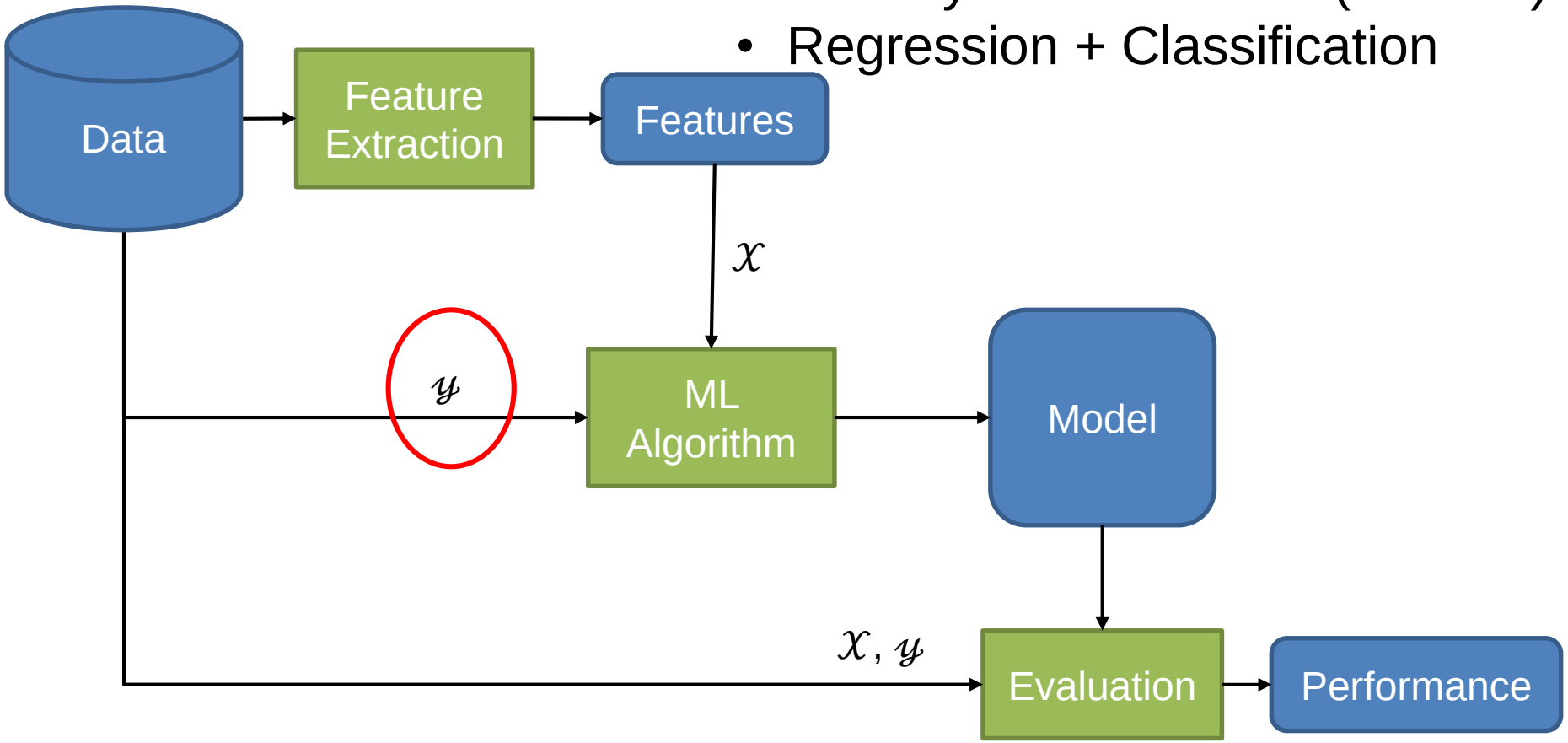
Computational Biology Example

- Clustering is very common in computational biology where protein, gene expression and sequence data is used to discover gene pathways (gene working together).
- Usually requires special algorithms that performs dimensionality reduction.

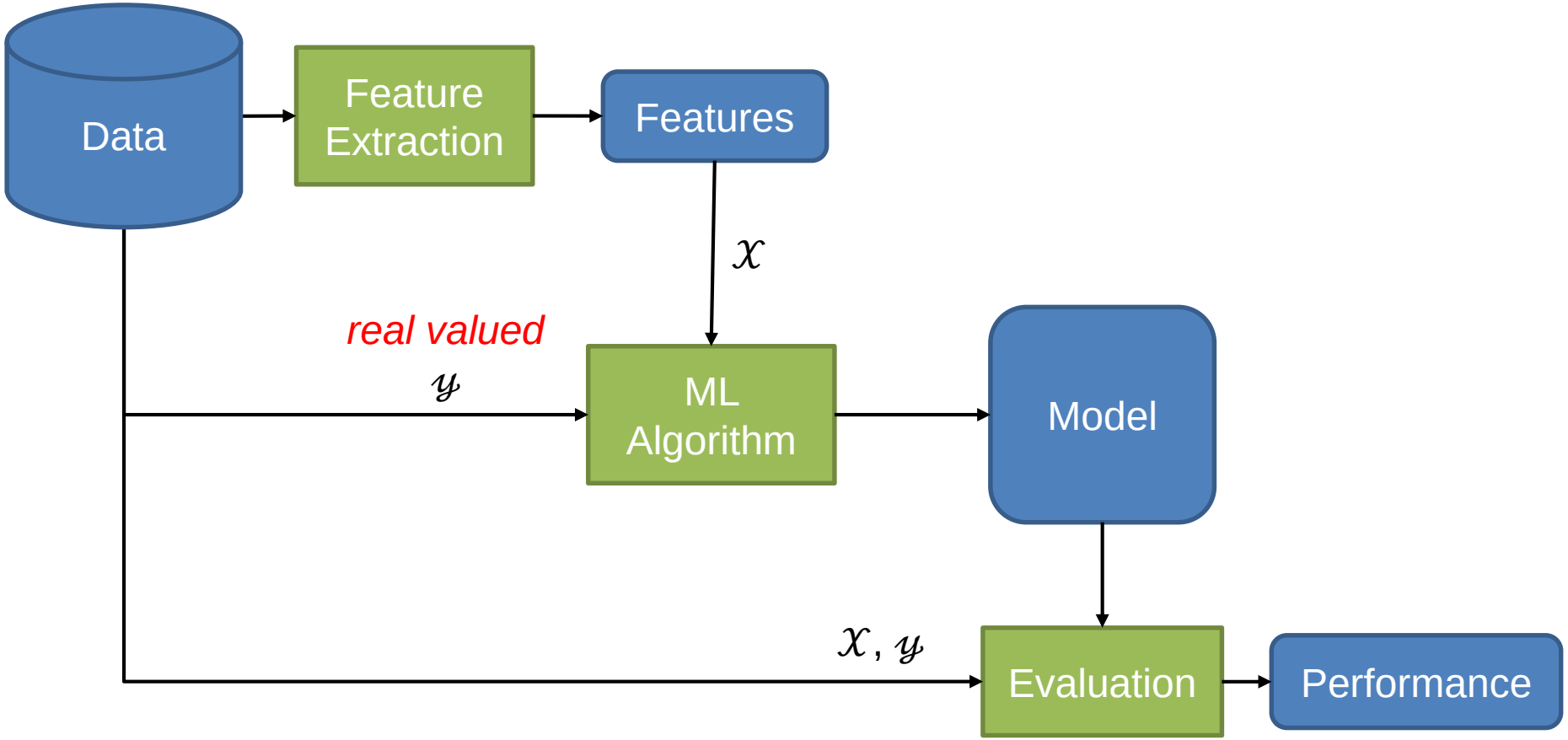


Supervised Machine Learning

- Most common ML.
- When you have labels (classes)
- Regression + Classification



Regression

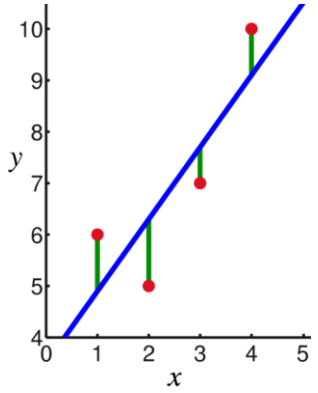

Data
Feature Extraction
Features
x
real valued
y
ML Algorithm
Model
x, y
Evaluation
Performance

Linear Regression

- Bread and butter of basic science.
- Assumes one *dependent* variable and one or more *independent* variables in a linear relationship.

$$y \sim \sum \beta_i x_i + \beta_0$$

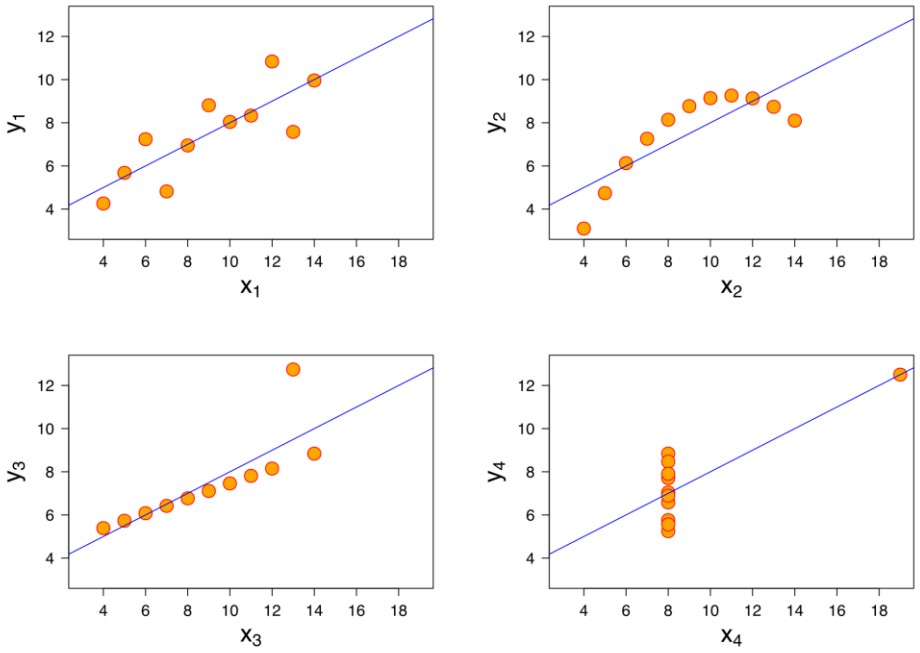
- Estimates the *coefficients* for independent variables – directly interpretable.
- Basic algorithm:
  - Least-squares estimation
  - Maximum-Likelihood Estimation
  - Many other optimizers



Linear Regression

- Linear relationship
- In multi-variable case, correlation structure of predictors.
- *Curse of high-dimensionality*

- Complex extensions:
  - Generalized linear models
  - Regularizations



Lasso

- Least Absolute Shrinkage and Selection Operator
- Variable selection + Regularization

$$y \sim \sum \beta_i x_i + \beta_0 \qquad \sum |\beta_i| \leq t$$

- Forces some coefficients (β_i) to be 0
  - Feature Selection
- Extensions:
  - Elastic Net: Strongly correlated predictors
  - Group Lasso: Select or not select predictors in groups
  - Fused Lasso: Force a smooth transition between coefficients (time-series or spatial relationships)

Cleft Example

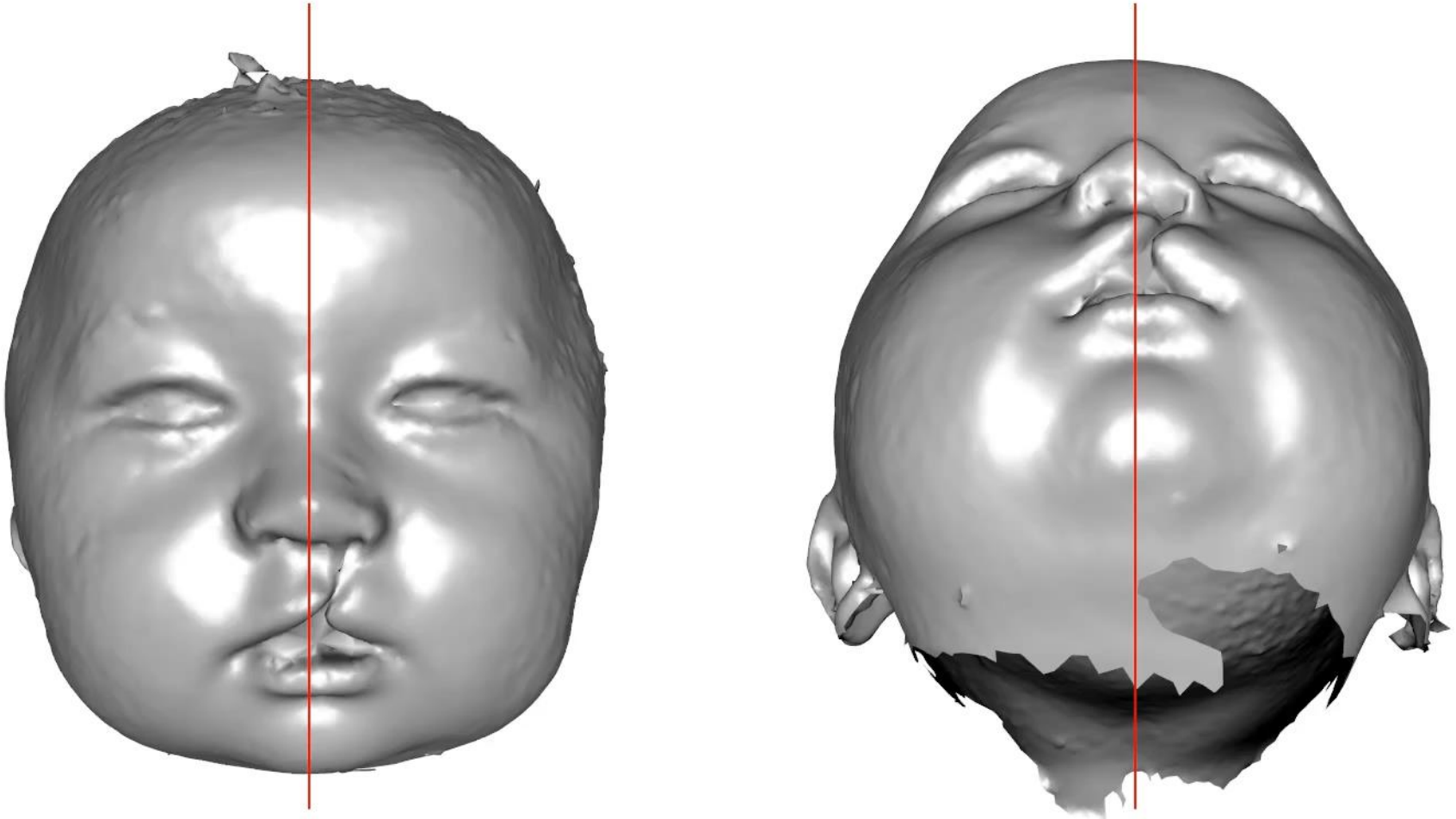
- Modeling the impact of cleft severity on nose deformity in unilateral cleft lip/palate
- Data = 3D surface Meshes
- Anthropometric Landmarks

Cleft Example

$$landmark.positions \sim severity + age + error$$

- We modeled cleft severity as the deviation of subnasale from the midline.
- linear regression + Thin Plate Splines for visualization
- Linear regression allows easy interpretation of the model
  - With 1mm of deviation of subnasale, what happens to cleft side and non-cleft side

Cleft Example



Cleft Example

3D surface meshes

Data

Feature Extraction

Features

Nasal Base Deviation (mm) + age

x

Linear Regression

y

ML Algorithm

Model

Landmark Positions

Nasal Base Deviation (mm) + mean age

x

Model

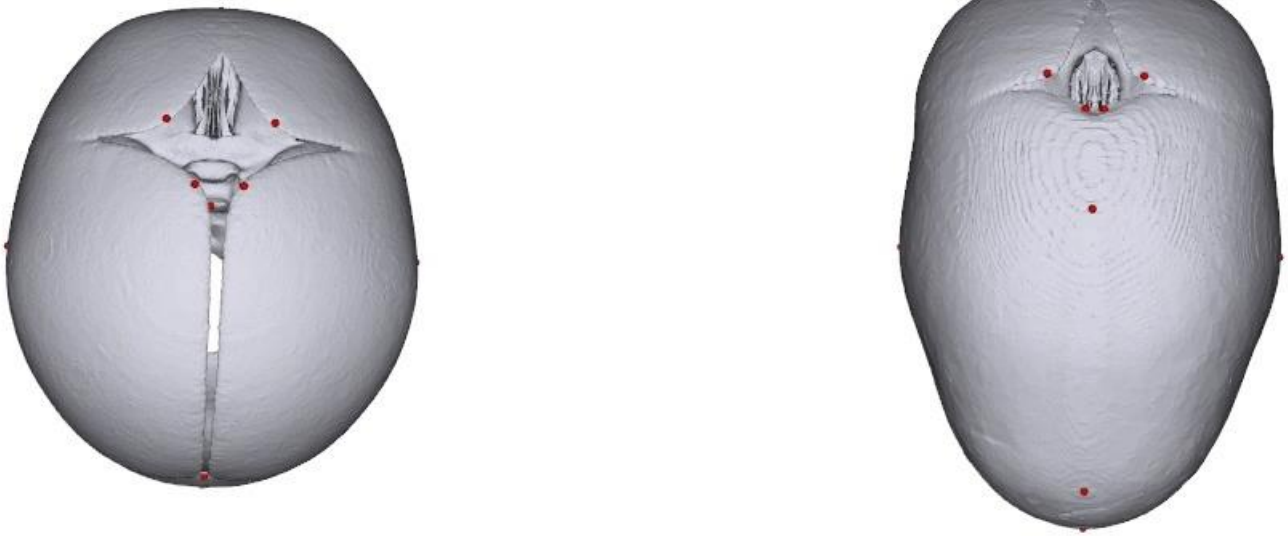
y

Landmark Positions

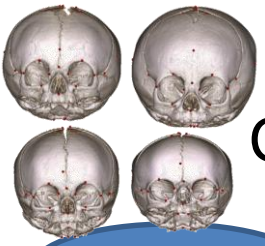
Cranial Growth Example

- Predict landmark locations based on age

$$landmark.position \sim age + error$$



Cranial Growth Example



CT Scans

Data

Feature Extraction

age

Features

x

Linear Regression

y

ML Algorithm

Model

Landmark Positions

age

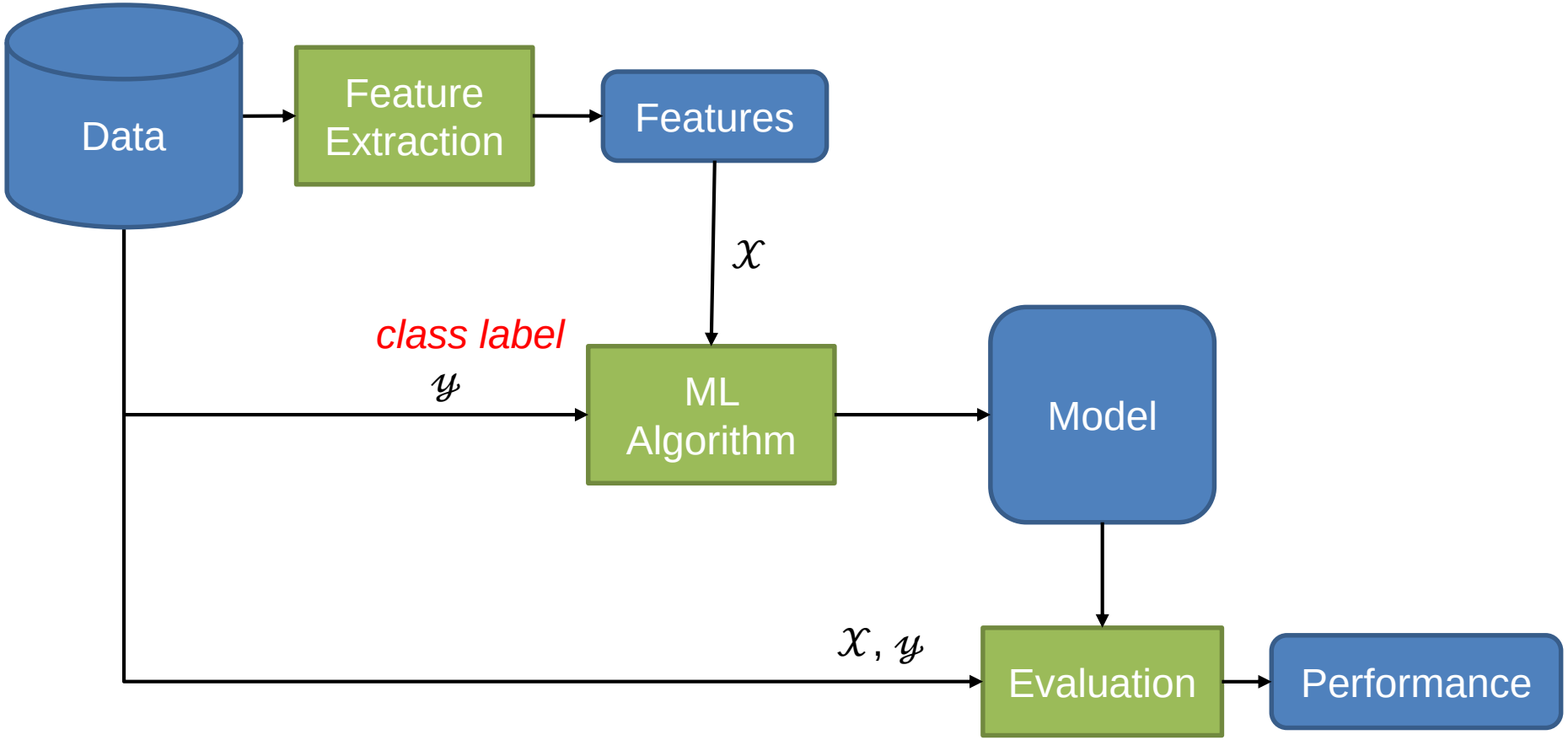
x

Model

y

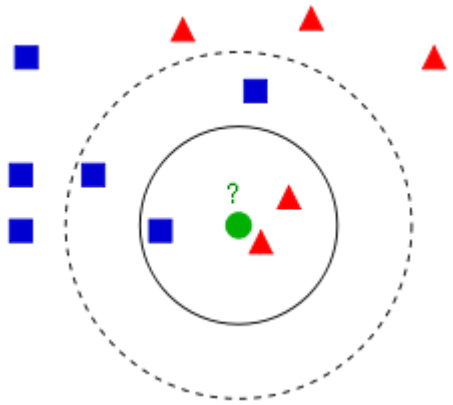
Landmark Positions

Classification



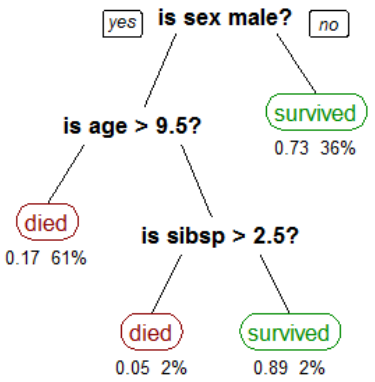
k-Nearest Neighbors

- Non-parametric (you are not learning anything)
- k-NN can be applied to a transformed feature space.

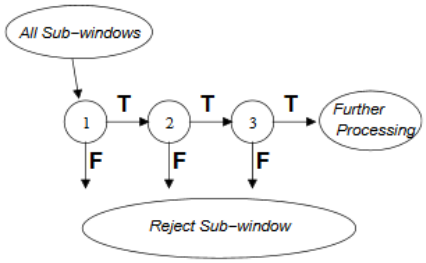


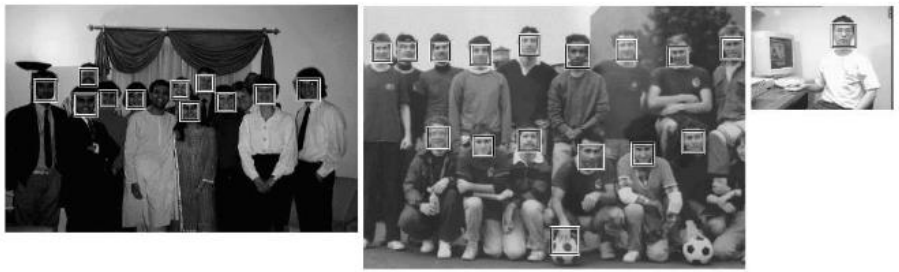
Decision Trees

- Flow chart
- A very popular early ML technique
- Interpretable but non-robust
- Non-generalizable



- Extensions:
  - AdaBoost
  - Random Forest





Logistic Regression

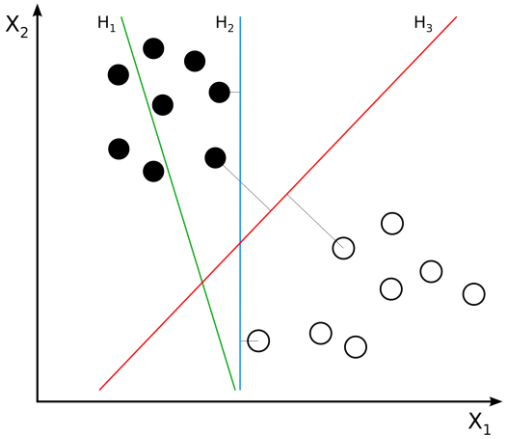
- Binary classification in linear regression way:

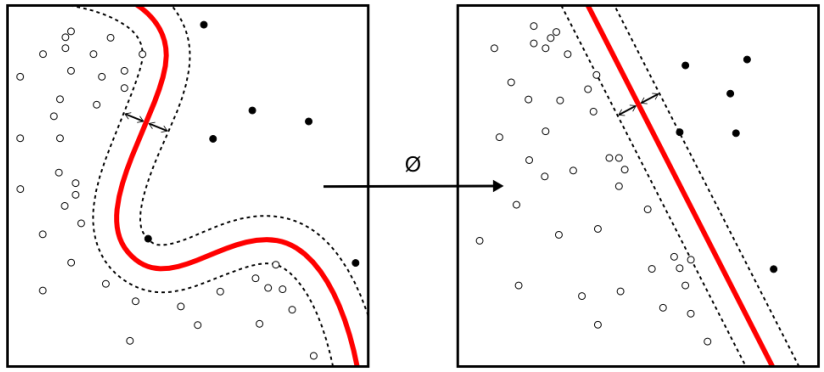
$$\ln\left(\frac{p}{1-p}\right) \sim \sum \beta_i x_i + \beta_0$$

- Natural logarithm of the odds (of being in class 0 or 1) is linearly modeled.
- Assumptions similar to linear regression.
- At inference time, it can produce "probabilities"
- Extensions
  - Multinomial logistic regression: More than 2 class
  - Regularizations (including Lasso)

Support Vector Machines

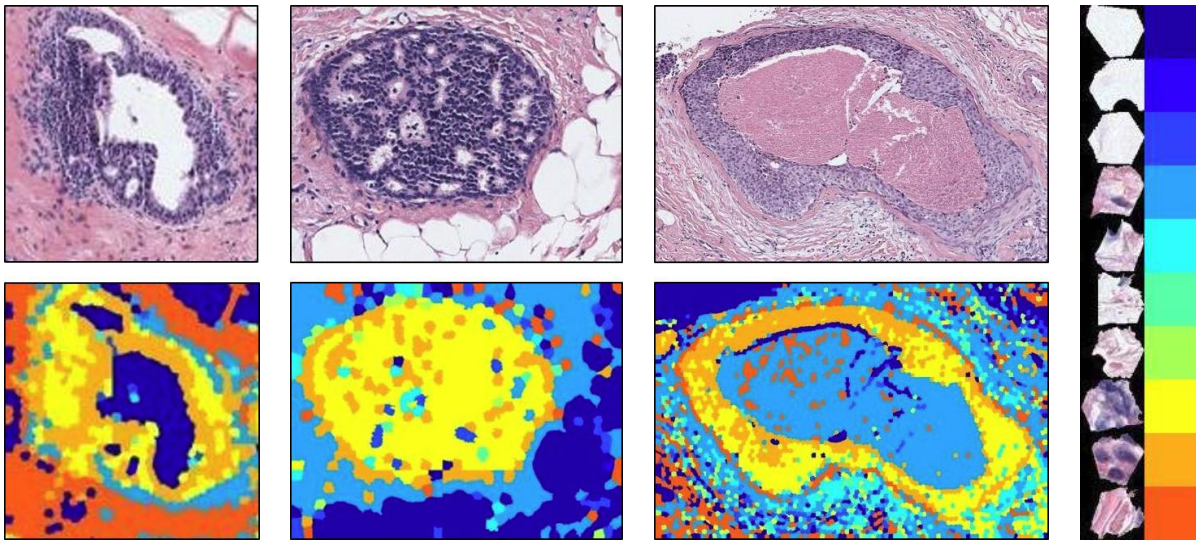
- Non-probabilistic binary classifier
- Maps data into a new space so that they are linearly separable.
  - Kernels: polynomial, Gaussian radial basis function (RBF), hyperbolic tangent



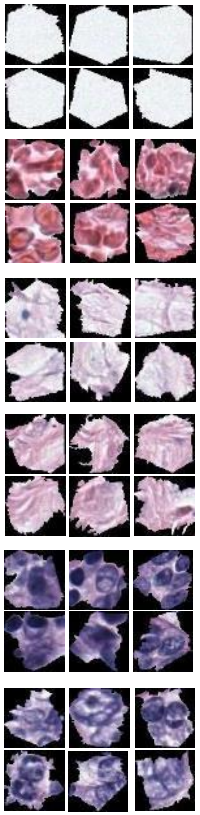


Discriminative vs. Generative

Histopathology Example



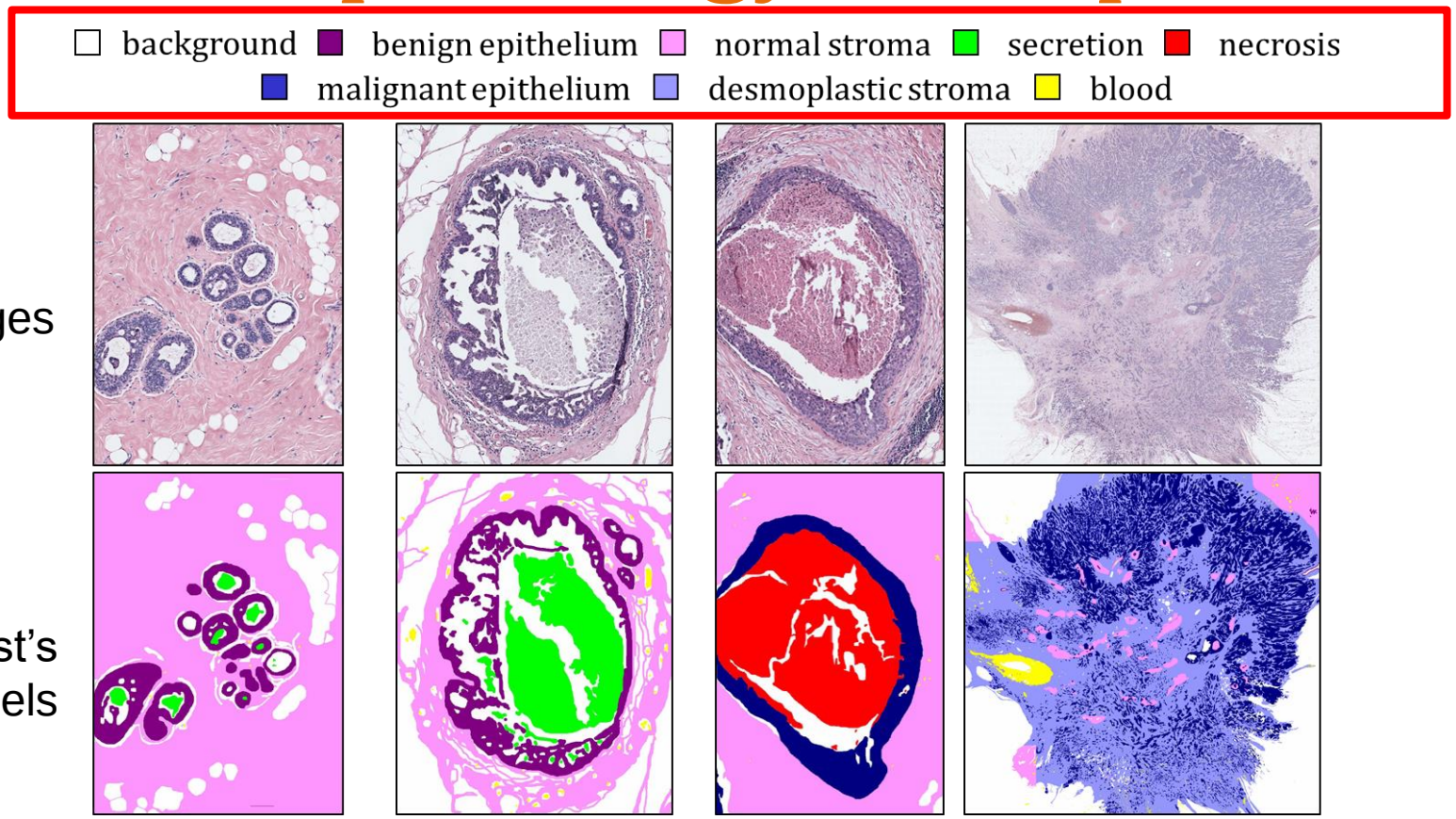
benign *atypia* *DCIS*



empty space
blood cells
loose stroma
dense stroma
normal nuclei
abnormal nuclei

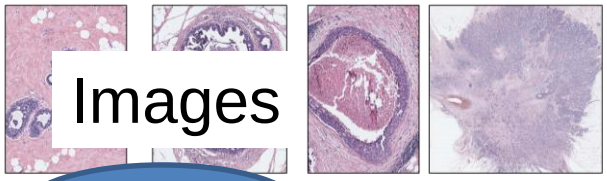
- Based on the clusters we *"discovered"*, we decided to *label* the images so we could train a classifier.

Histopathology Example



- Automatically mark the superpixels in a supervised manner.

Histopathology Example



Images

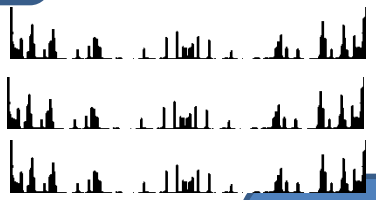
Superpixels
color + texture

Data

Feature Extraction

Features

x



Color + Texture
histograms

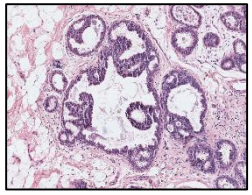
Labels y

ML Algorithm

Model

SVM

New Image

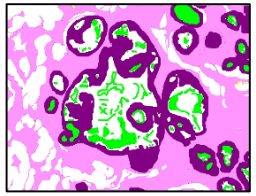


x

Model

y

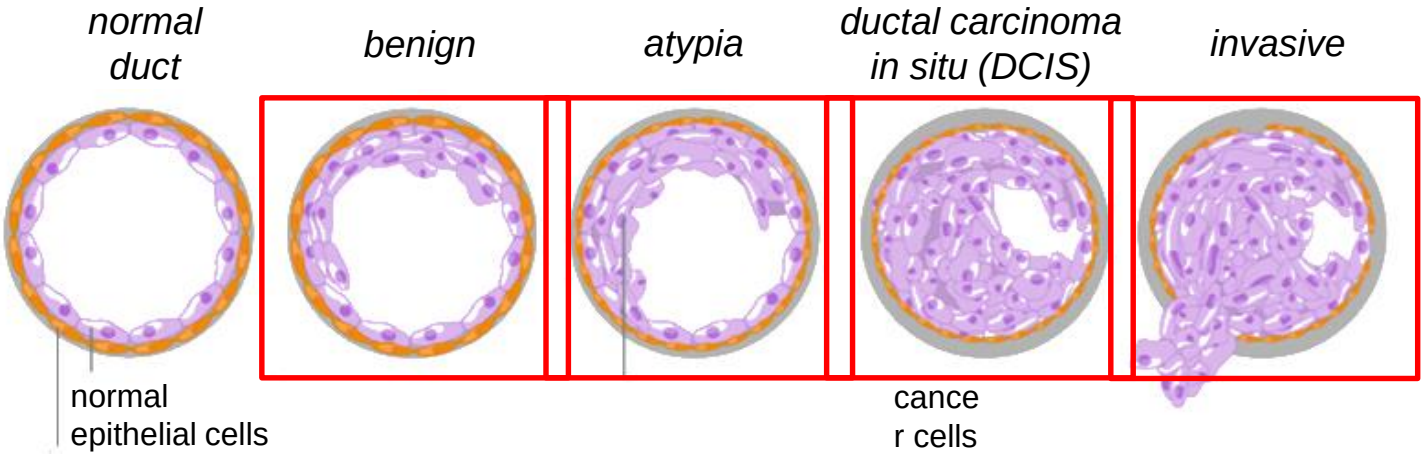
Tissue Labels

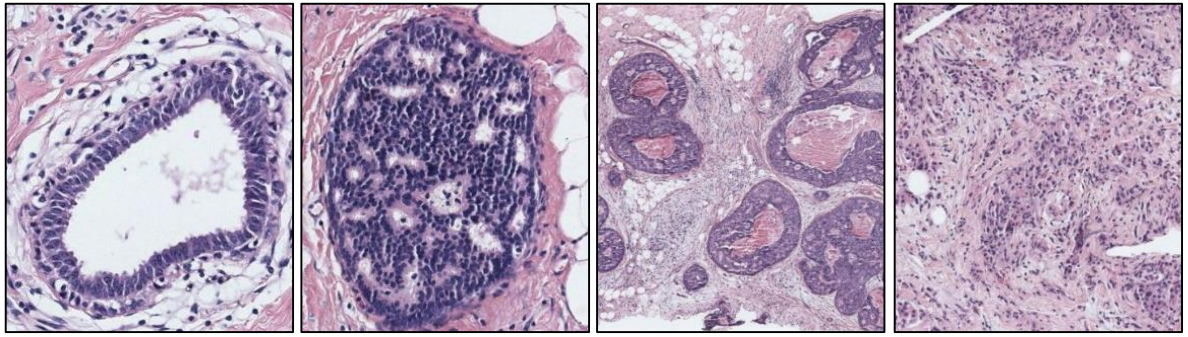


Histopathology Example2

Why label different tissue types in a breast biopsy image?

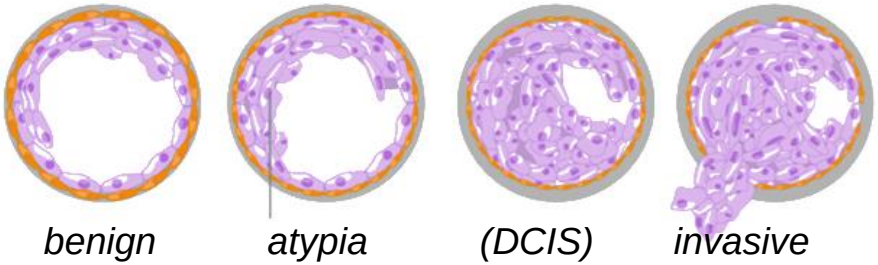
Diagnostic Classification!

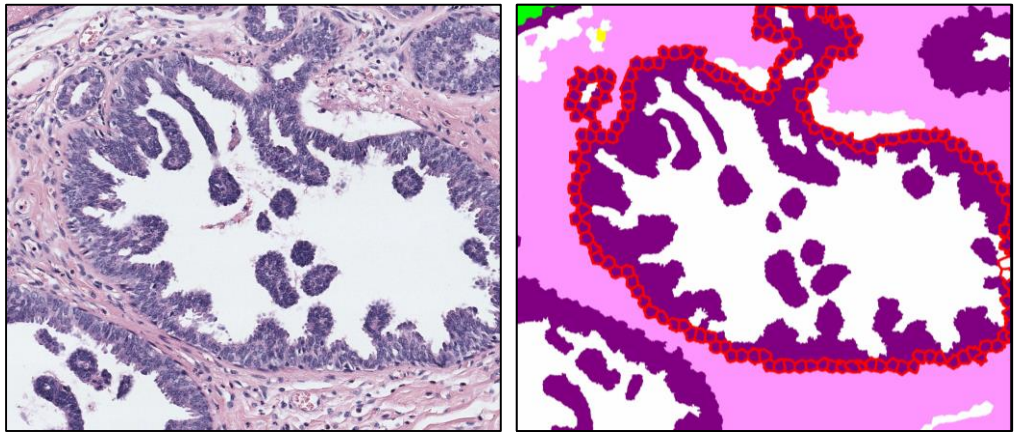




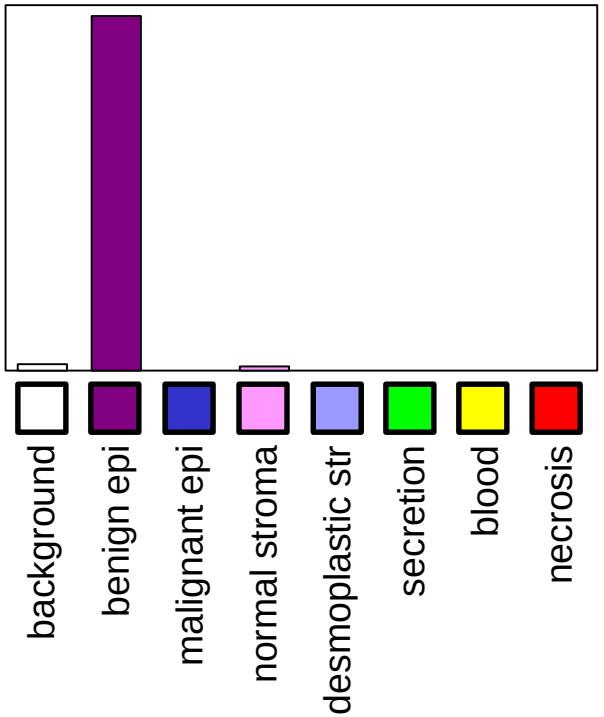
Histopathology Example2

Structure Feature: Summarizes structural changes using tissue labels.



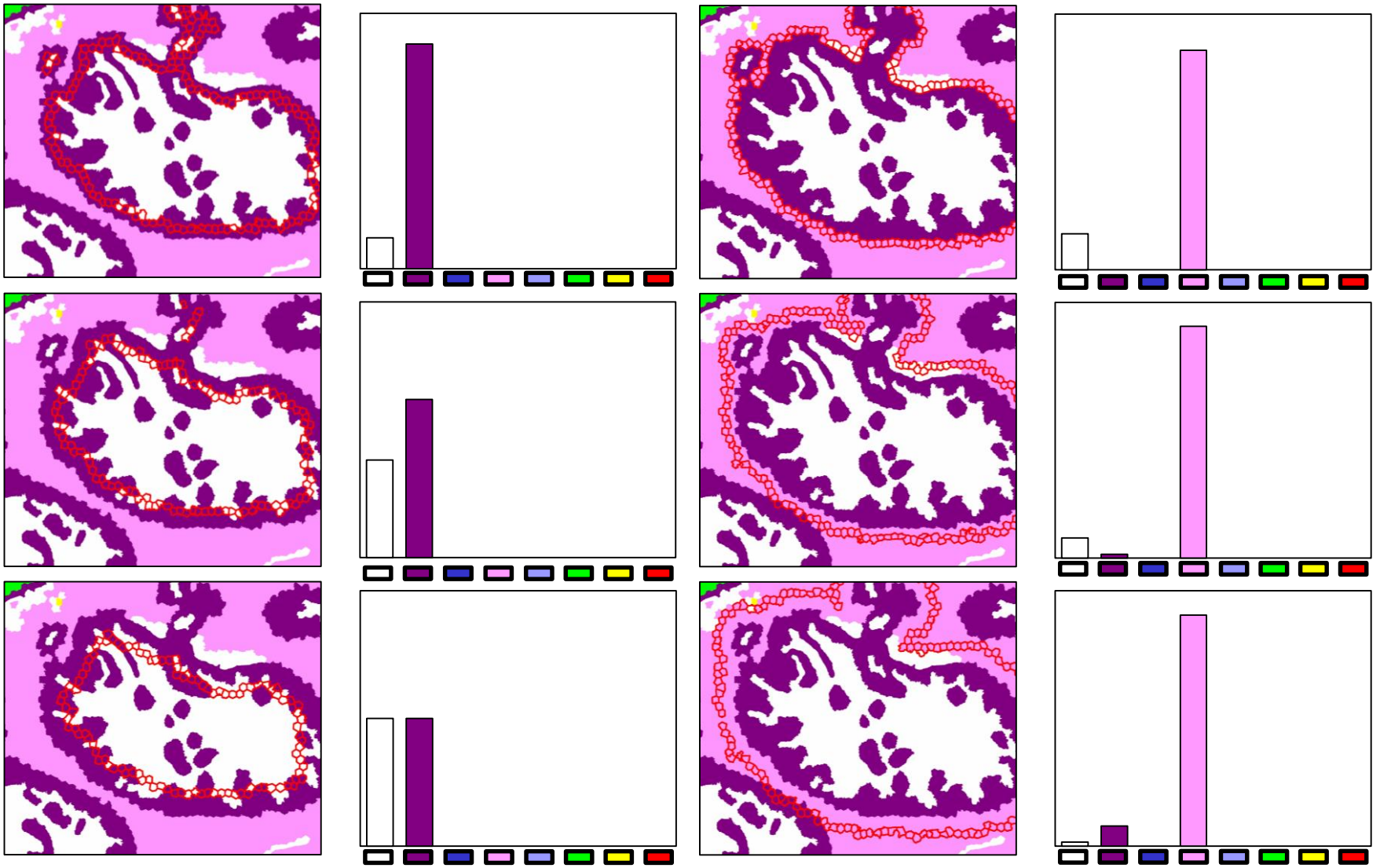


duct layer

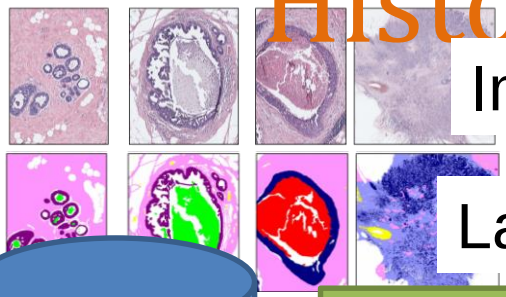


Inner Layers

Outer Layers



Histopathology Example2



Images

Labels

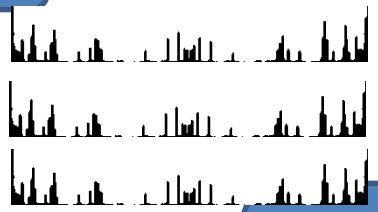
that we automatically found using SVM

Data

Feature Extraction

Features

x



Structure Feature

Diagnosis

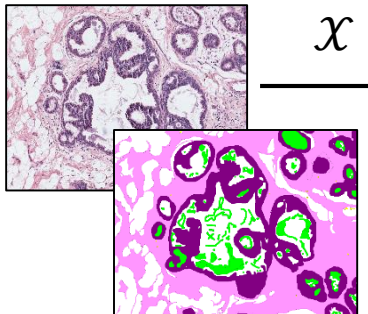
y

ML Algorithm

Model

SVM

New Image (+Label)



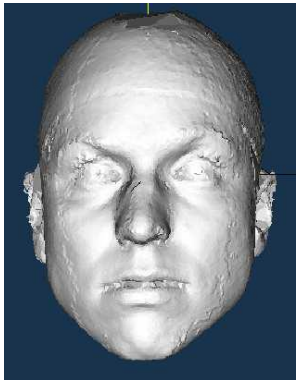
x

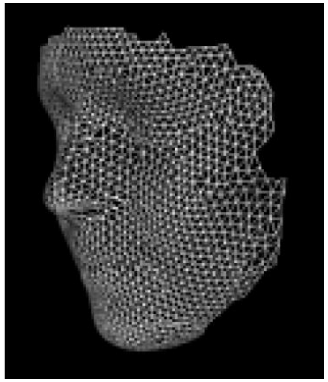
Model

y

Diagnosis

FaceBase Example

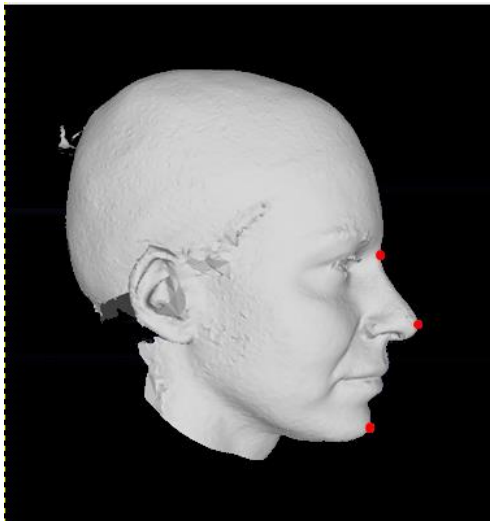


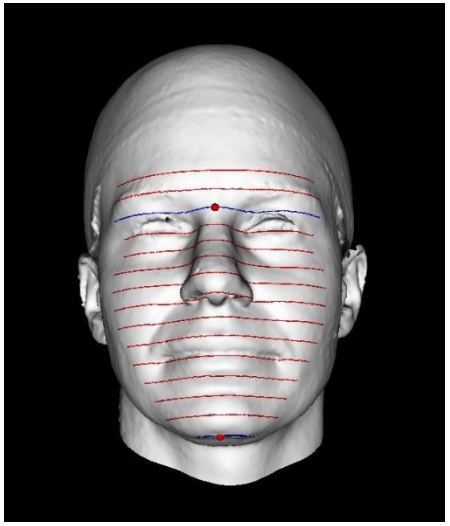


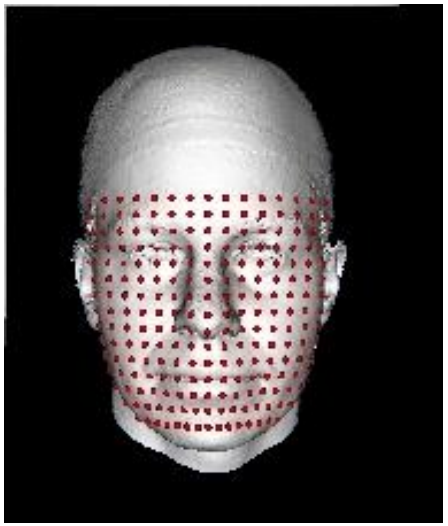
- ~2000 human face surface meshes with age and sex information.

FaceBase Example

Feature Extraction







x_1, y_1, z_1
x_2, y_2, z_2
x_3, y_3, z_3
x_4, y_4, z_4
...

3D Surface Mesh

3D coordinates

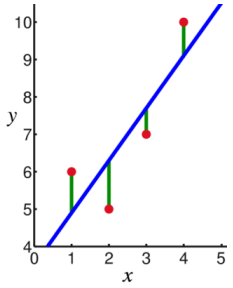
Raw Data

X

FaceBase Example

x_1, y_1, z_1
x_2, y_2, z_2
x_3, y_3, z_3
x_4, y_4, z_4
...

3D coordinates



$$\ln\left(\frac{p}{1-p}\right) \sim \sum \beta_i x_i + \beta_0 \qquad \sum |\beta_i| \leq t$$

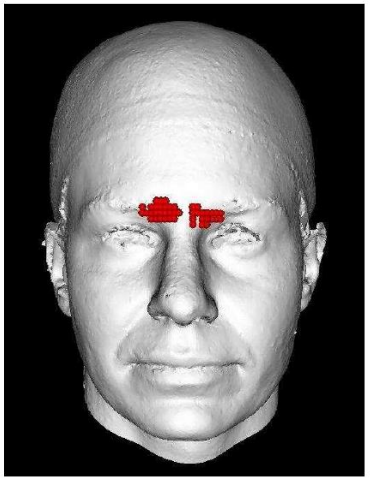
Logistic Regression + group lasso

Sex:
Male or Female
(0,1)

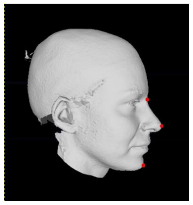
$\mathcal{X}$

$\mathcal{Y}$

- Logistic regression models can be interpreted.
- We used “group lasso” so all 3 coordinates of one point would be selected together.
- Image shows most discriminative features (points) for sex classification.

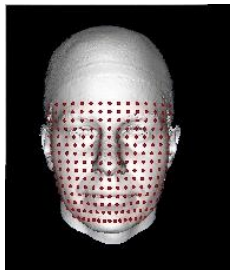


Classification



3D surface mesh
+ binary sex label

Pseudo-landmarking



x_1, y_1, z_1
x_2, y_2, z_2
x_3, y_3, z_3
x_4, y_4, z_4
...

Data

Feature Extraction

Features

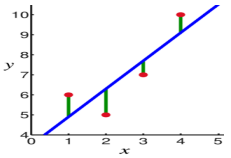
x Pseudo-landmark coordinates

sex
y

ML Algorithm

Model

Logistic Regression
+ Lasso

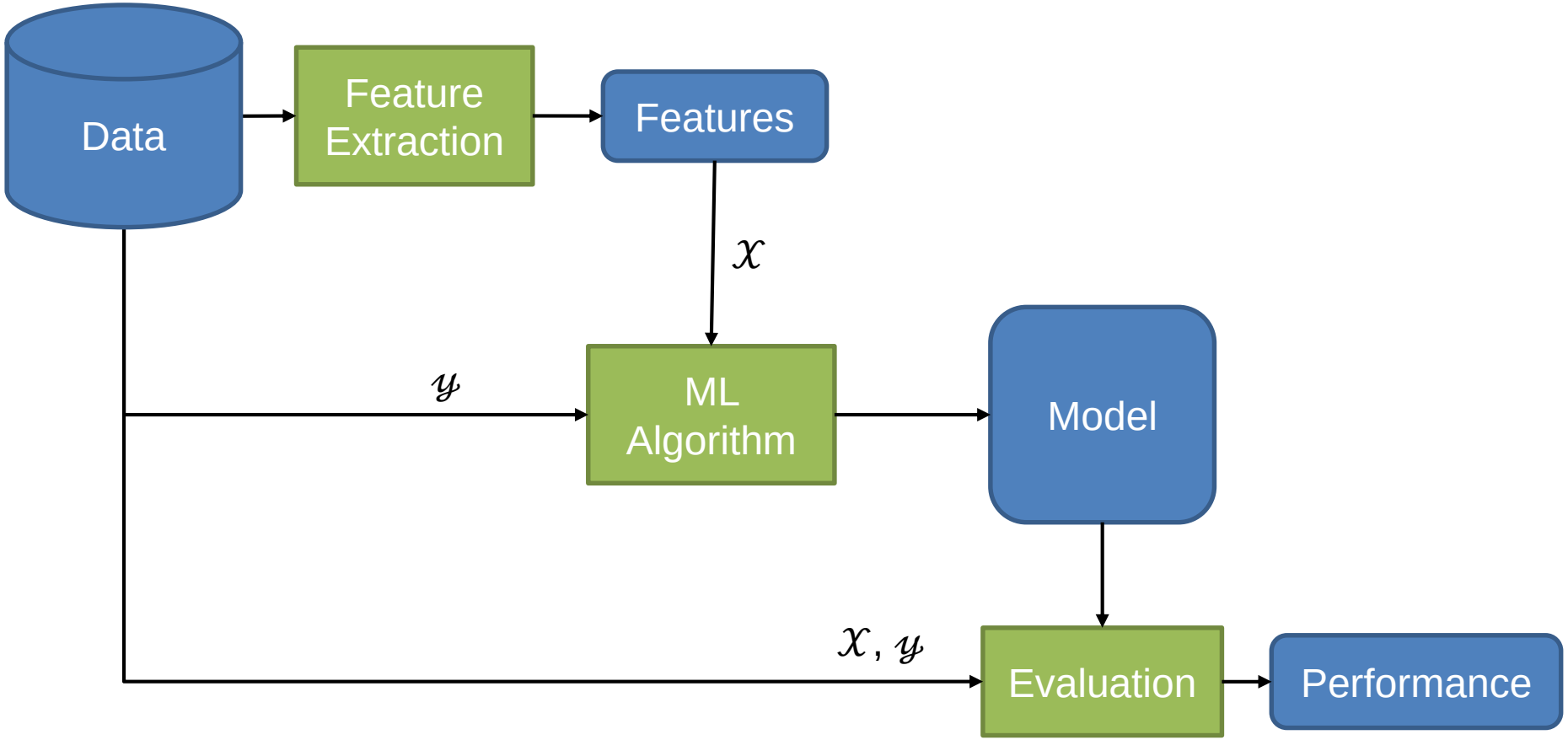


x, y

Evaluation

Performance

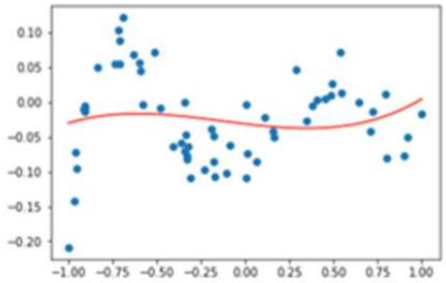
Supervised ML



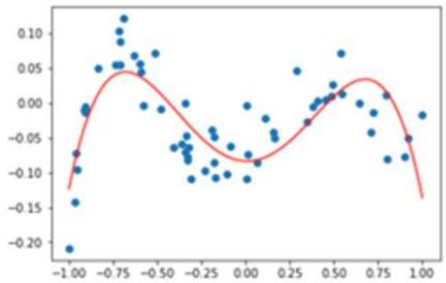
Evaluation

- How do you know your model is good?
- Most ML algorithms minimize "error"
  - The model *fits* the data.
- There is no end to *fitting.*

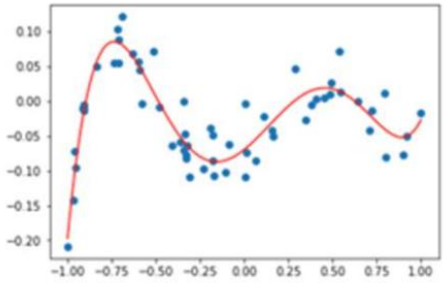
overfit



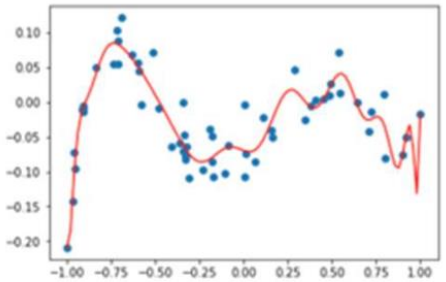
$p = 3$



$p = 4$



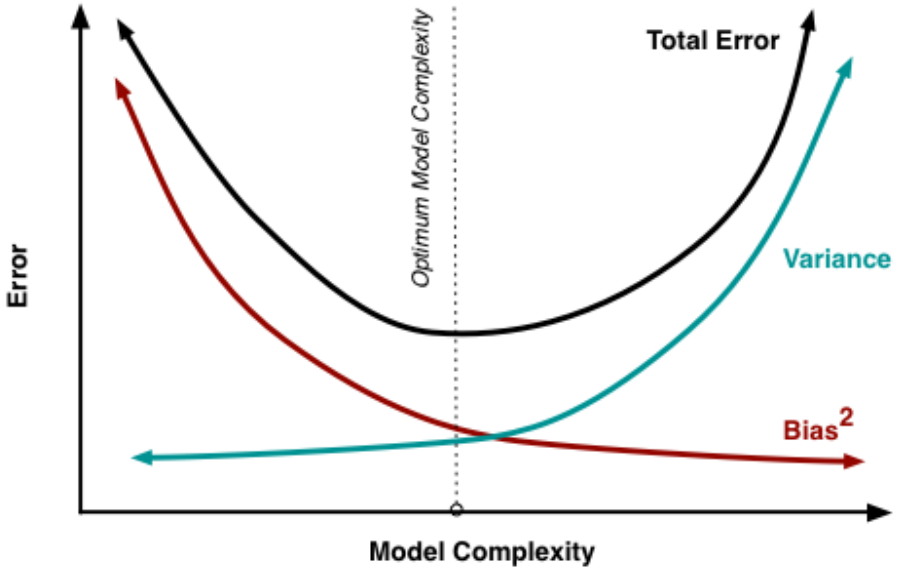
$p = 5$



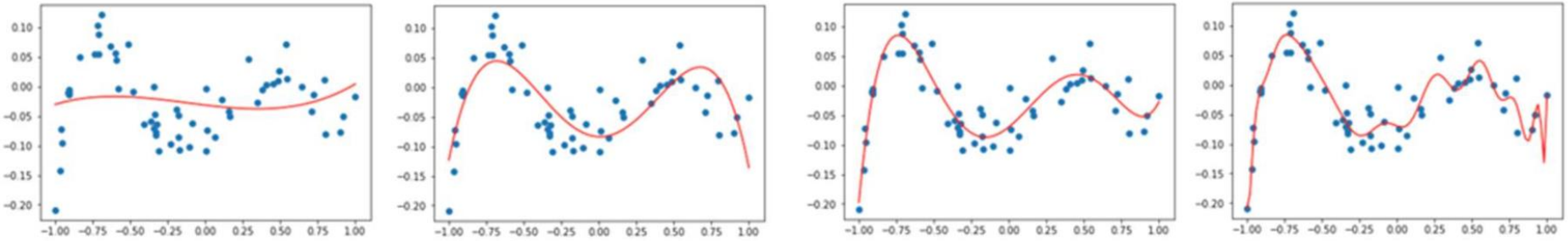
$p = 20$

Bias-Variance Trade-Off

- Bias: underfitting
- Variance: overfitting

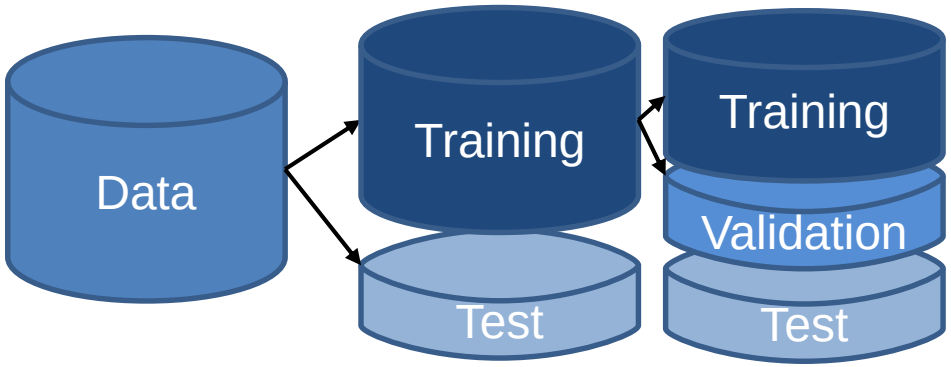


- Strategies
  - Regularization
  - Feature selection
  - Dimensionality reduction
  - Larger datasets ☺



Training/Validation/Test

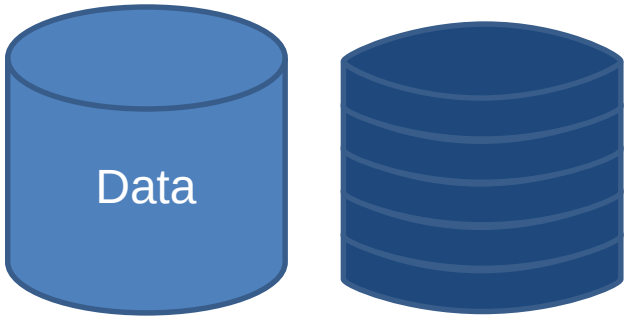
- Split the data and forget about test set.
- Fit the model to the training data, report the error (or any evaluation metric) on test data.
- If your model has parameters you need to adjust (number of clusters, polynomial degree etc…), you are due for another division.



- Train on training set, test on validation set, pick the parameter that gives the least error on validation set.
- Then report the error on test set.

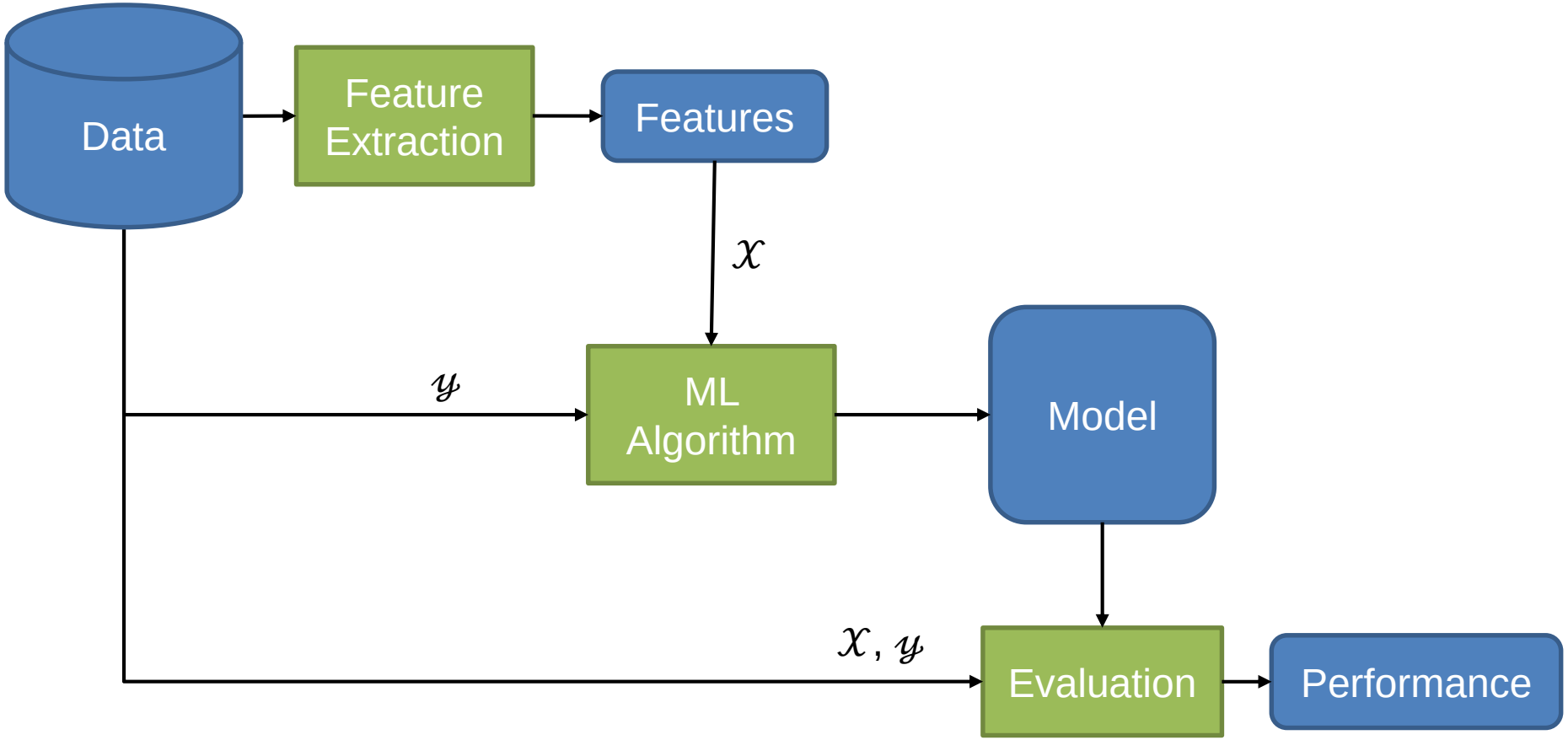
Cross-Validation

- Splitting the data has a randomness to it. What if you got lucky and all *easy* samples ended up in the test set?
  - Divide the data *k* equal subsets: *folds*
  - Use *k-1* folds as training set, leftover fold as test set. Repeat *k*-times.
  - Use the same approach for validation set.



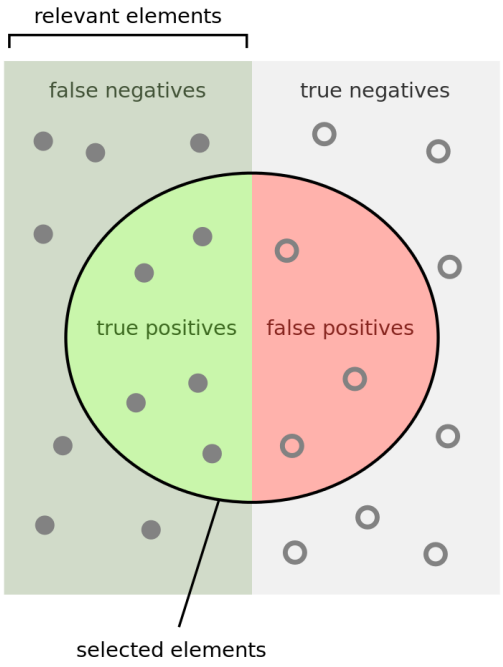
Ideally, use leave-one-out-cross-validation (LOOCV) where your test set is 1 sample!

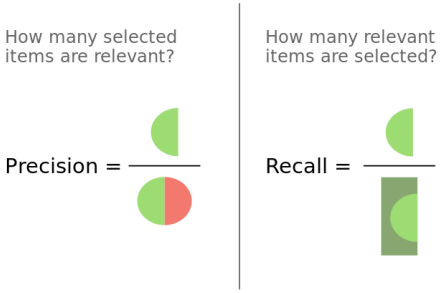
Supervised ML



Evaluation Metrics

- Binary Classification
  - Precision-Recall
  - Sensitivity-Specificity
  - Accuracy
  - F-measure
    - $F = 2 \times \frac{precision \times recall}{precision + recall}$
  - ROC curves
    - Area Under the Curve (AUC)
- Multi-class Classification
  - Confusion matrices





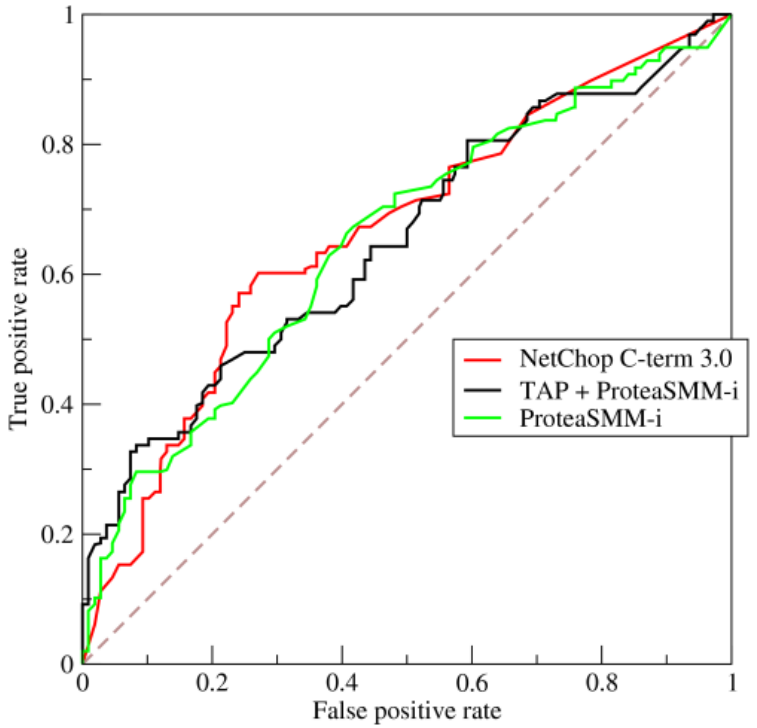
Confusion Matrices

		True Label	
		True	False
Predicted Label	True	True Positive	False Positive
	False	False Negative	True Negative

- You want the diagonal to be high numbers.
- Can be extended to multi-class.

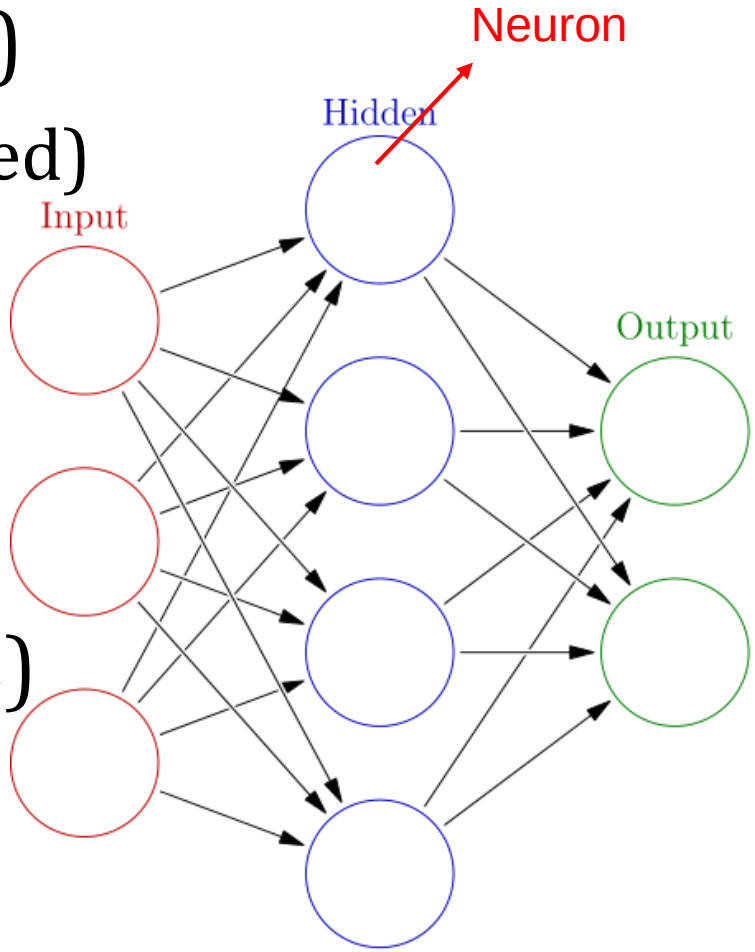
ROC Curve and AUC

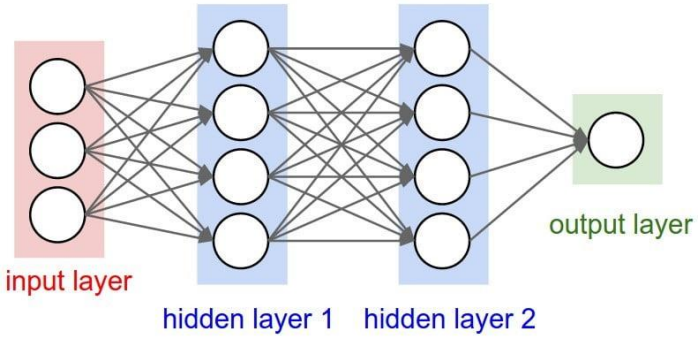
- Discrimination threshold (probability cut-off) is varied and *true positive rate* vs *false positive rate* is plotted.
- Ordering samples
  - Higher the area under the curve the better.
  - Diagonal is the random chance.



(Deep) Neural Nets

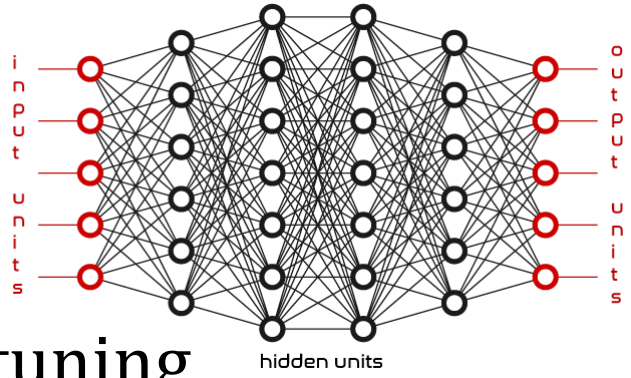
- Artificial Neural Nets (40s)
  - Neurons fire (or are activated) based on the input.
  - A combination of inputs/ outputs
  - Learn weights / thresholds.
- Backpropagation (70s-80s)



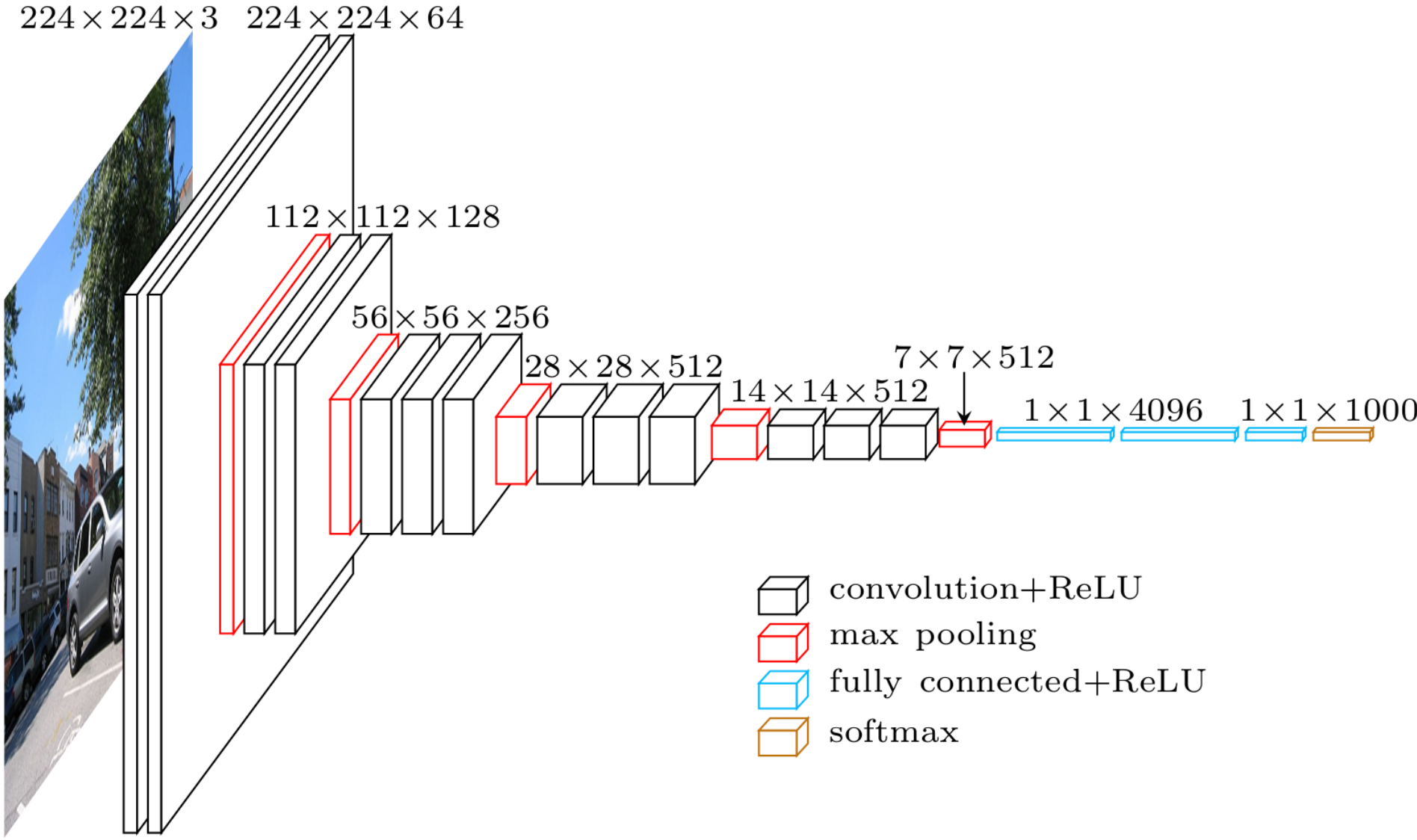


Deep Neural Nets

- The plot thickens…
  - GPUs: Perfect machine for repeated matrix calculations (that's what graphic is all about)
    - Nvidia "big bang" (2000s)
  - Transfer learning
  - Pre-trained networks / fine tuning

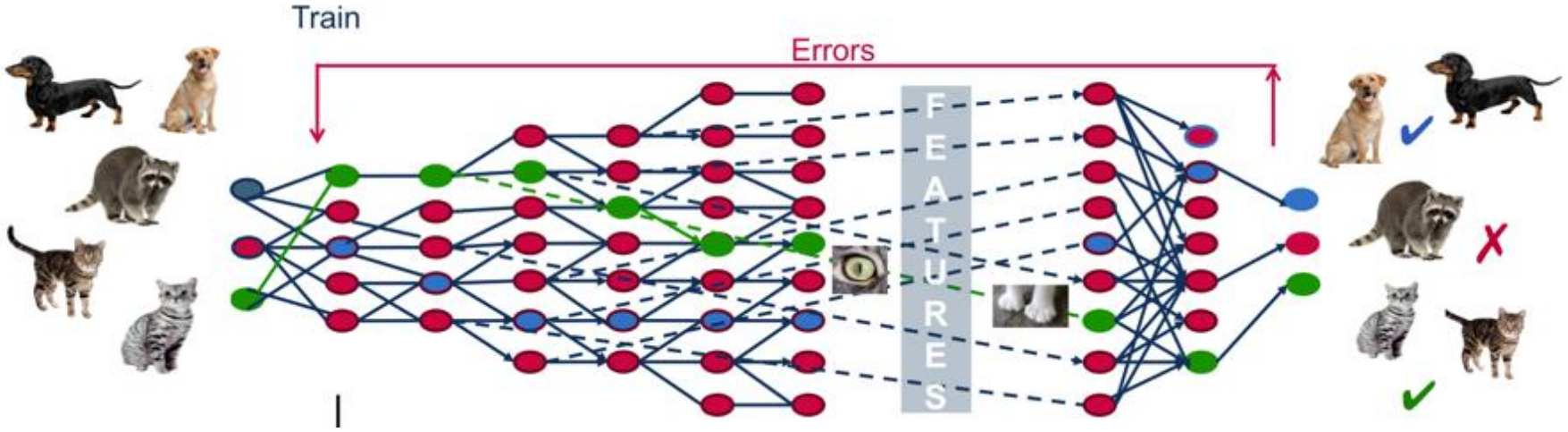


Deeper Neural Nets

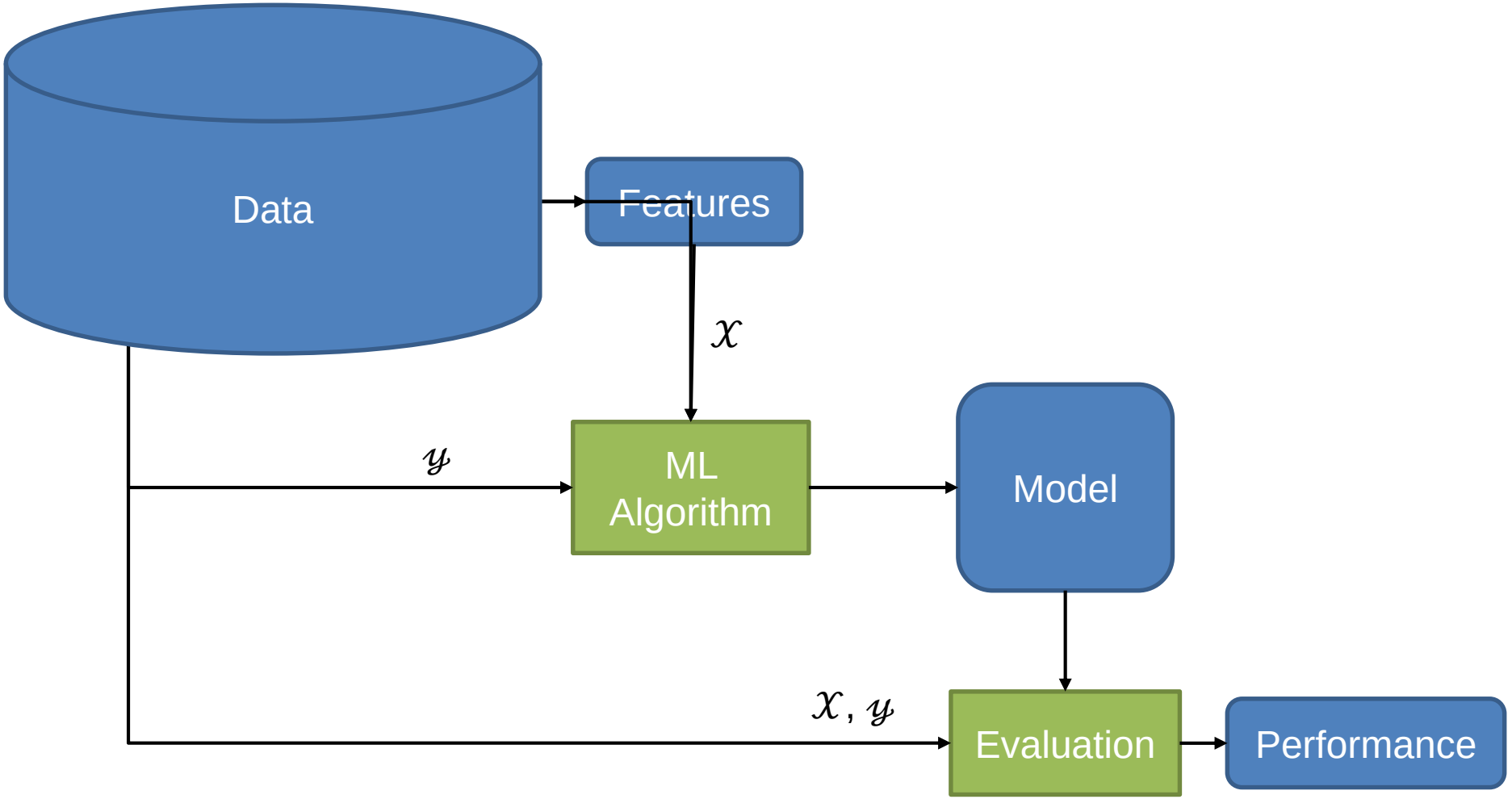


Convolutional Neural Nets

- The input of the network is the image
- The first layers are "convolutional layers" which are filters/kernels applied to image
- The filters themselves are learnt.
- No more *"feature extraction"*

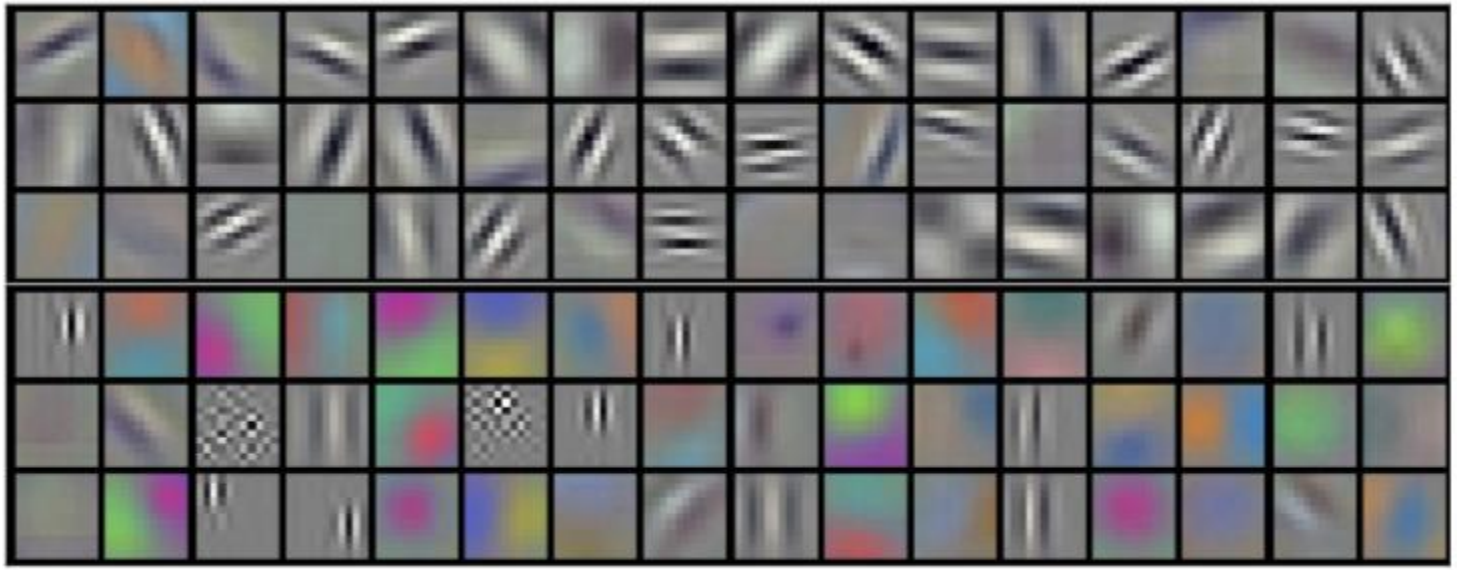


Convolutional Neural Nets



Convolutional Neural Nets

- What do they learn?



Learnings of First convolution layer on image of size 224X224X3

Convolutional Neural Nets

- Anyone can download a network trained for object detection
  - The convolutional layers are trained to be traditional filters/kernels (edge – corner detection etc.)
- Strip the last layers where the output calculation is done
- Add new layers based on their task
- Fine-tune
  - Train at a slower rate so weights and other parameters change just slowly to adjust to the new task

With Great Power, Comes Great Responsibility

- Remember that your model is as good as your data and your assumptions
  - Garbage in, garbage out
- Use training / validation / test sets and cross-validation.
- Understand what your model does
  - All models are wrong… but some are useful

With Great Power, Comes Great Responsibility

- Occam's razor
  - When presented with competing hypotheses that make the same predictions, one should select the solution with the fewest assumptions
- More data is not necessarily good data.
  - Variability, *"difficult"* examples
  - Start with simple data exploration