

Experimental Setup, Model Selection, Overfitting, Regularization

Explaining concepts with a polynomial fitting example

Roberto Souza

W2025

Outline

- Learning Goals
- Experimental setup and model selection
- Overfitting and regularization
- Metrics
- Summary

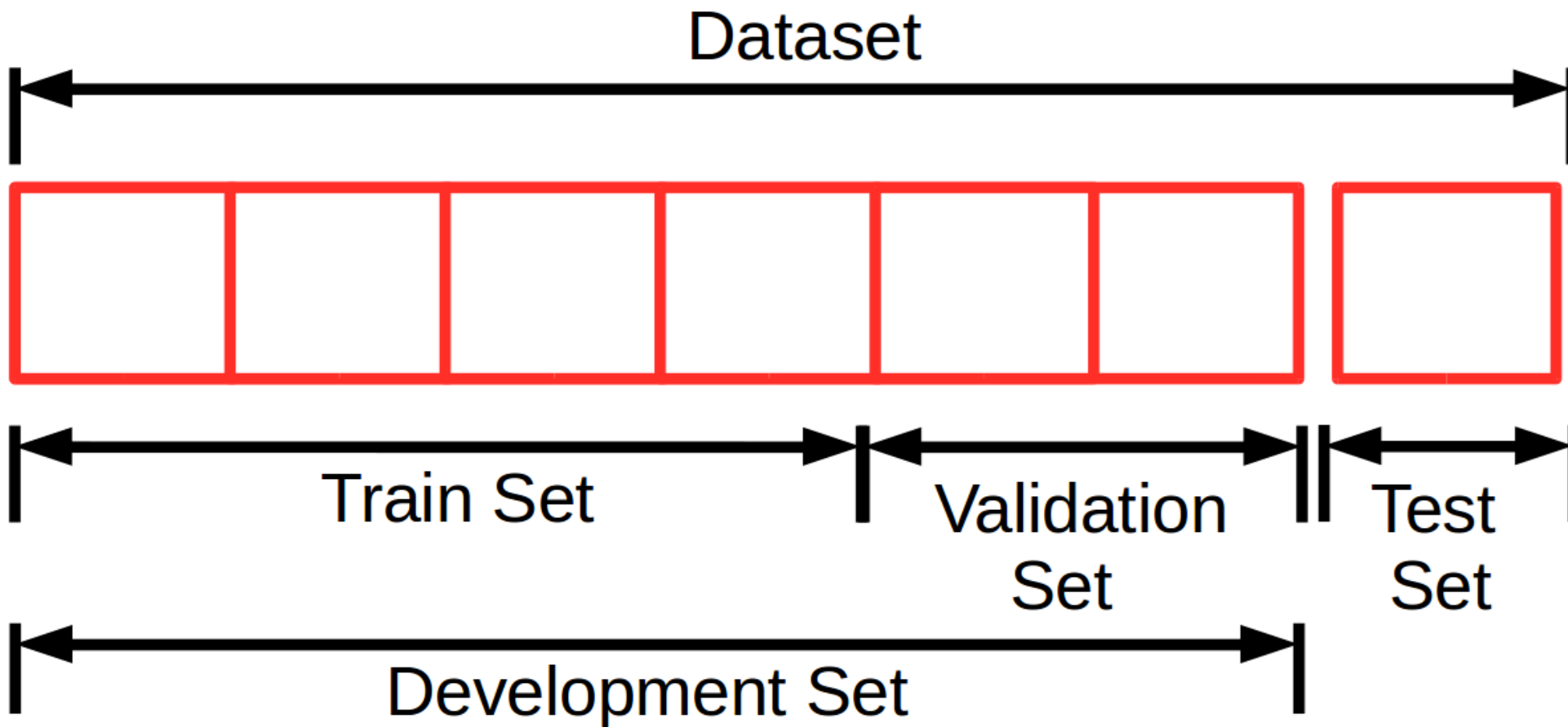
Learning Goals

- Explain how to design your experiment
- Introduce how to select your model
- Introduce the concepts of *over-fitting*, *under-fitting*, and *model generalization*.
- Introduce the concept of *regularization* for reducing model *over-fitting*.

Hands-on Tutorial

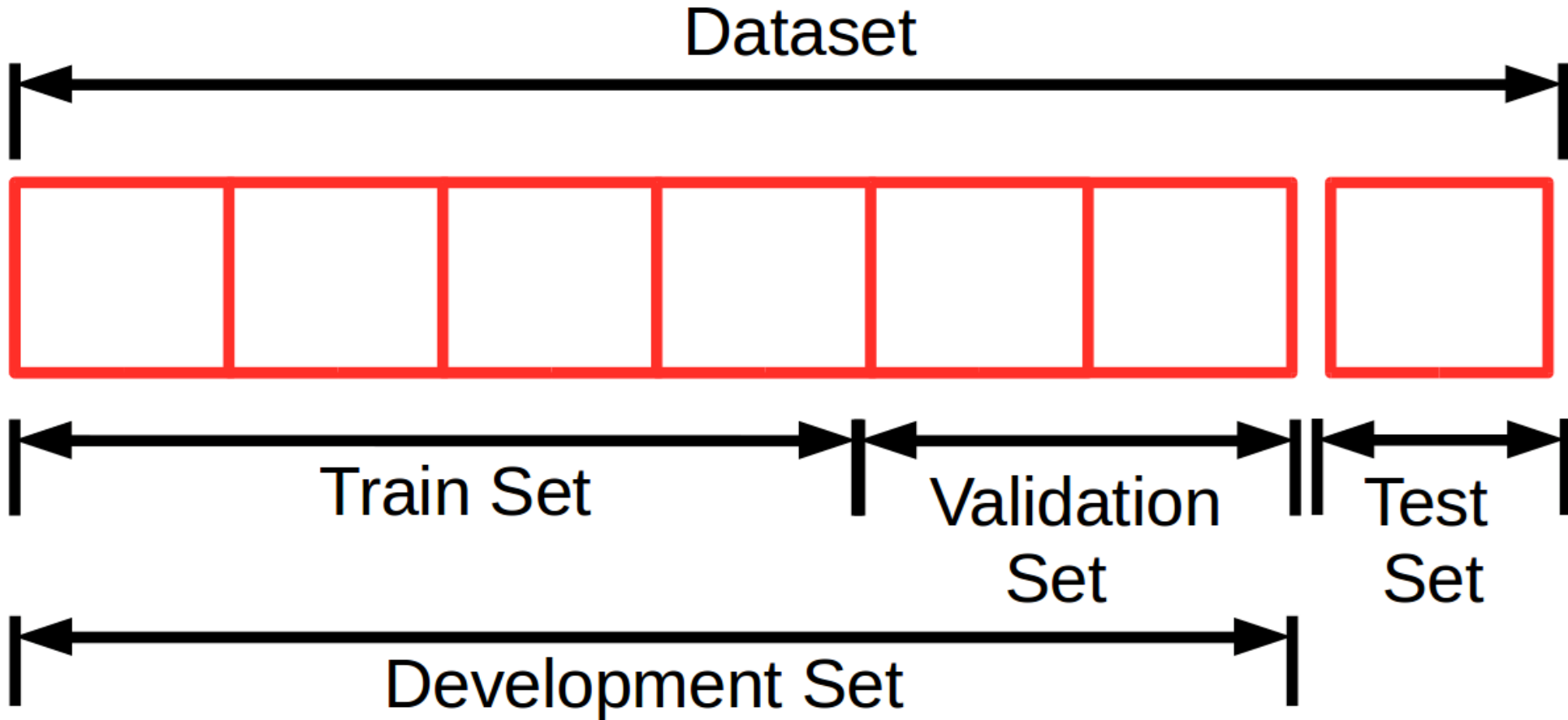
- <https://github.com/rmsouza01/ENEL645-F2024>
- **Tutorial:** [Model selection, overfitting, regularization](#)
- Based on the example presented in chapter 1 of the book: **Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA.**

Experiment Design: Train, Validation and Test



WHY?

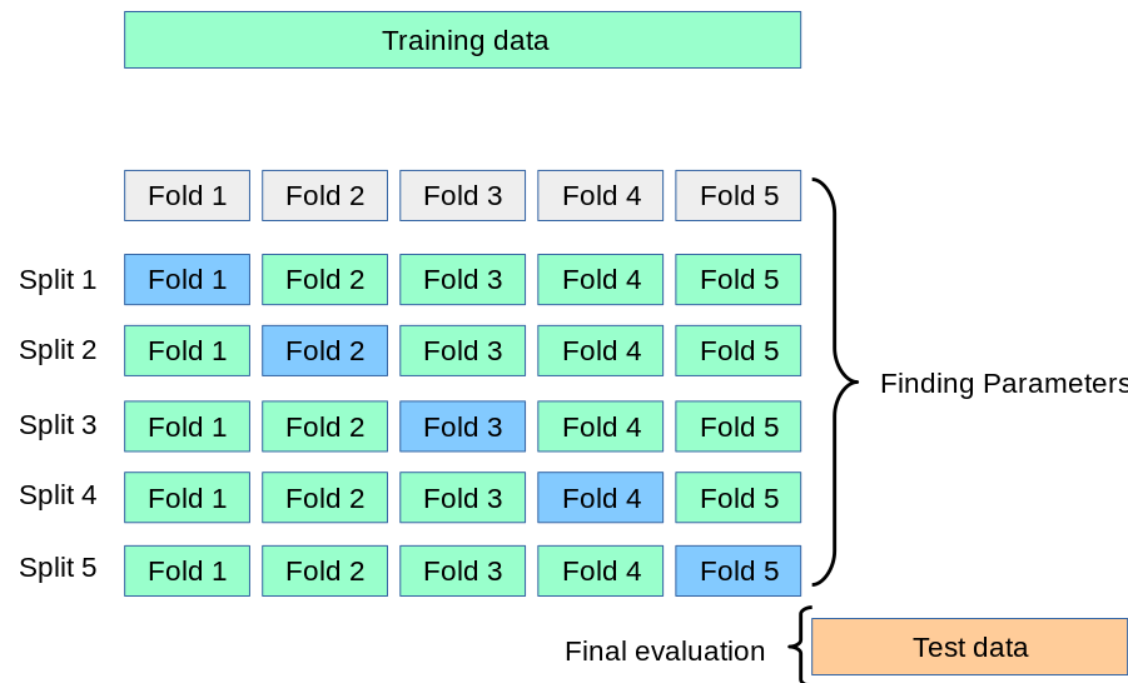
Experiment Design: Train, Validation and Test



- **Train set:** learn parameters of your models
- **Validation set:** model selection
- **Test set:** verify generalizability to unseen data

Experiment Design: k-fold cross validation

- Performs k iterations on the data
- Stratified k-fold: maintain the proportions of each class into folds (unbalance data)

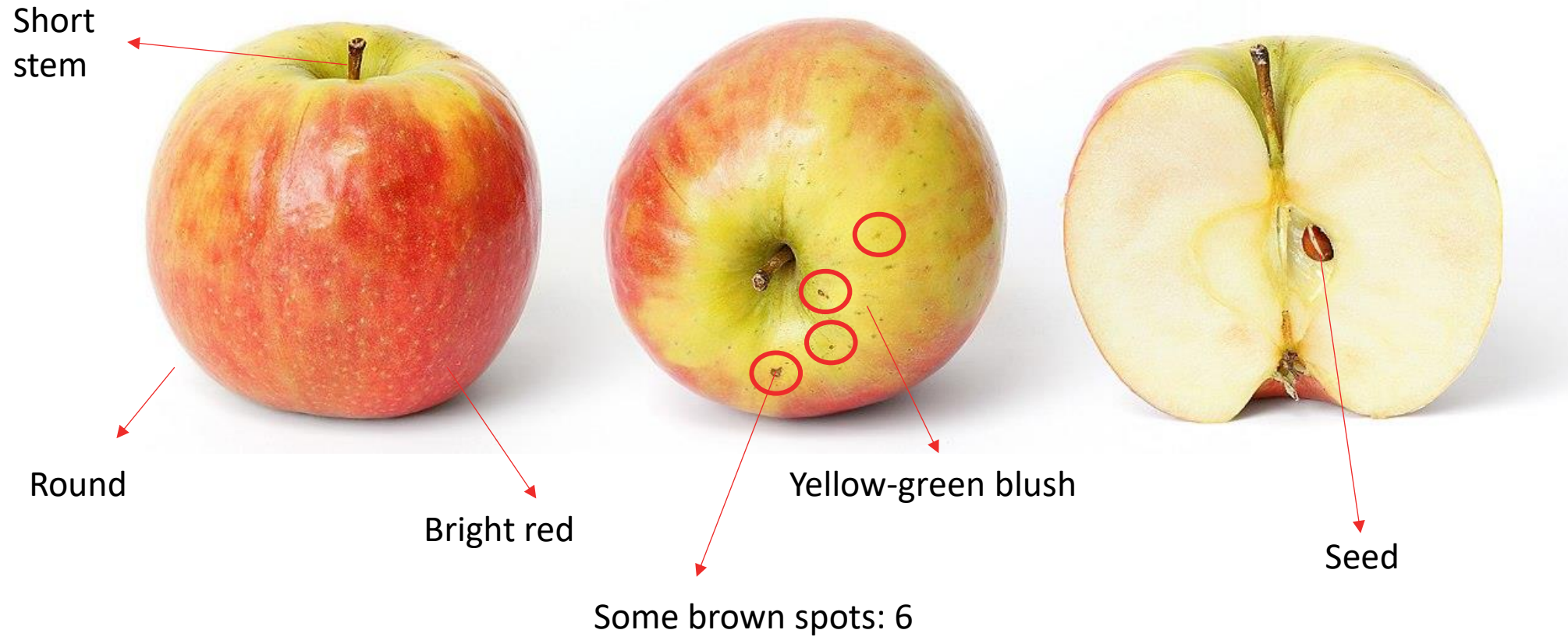


Source: https://scikit-learn.org/stable/modules/cross_validation.html

Under- and Over-fitting



Under- and Over-fitting



Complex model

Under- and Over-fitting

- What is an apple?

- 1 - Short stem
- 2 – Round
- 3 – Bright and red
- 4 – Yellow-green blush
- 5 – Seed
- 6 – Some brown spots

Under- and Over-fitting

- What is an apple?

- 1 - Short stem
- 2 – Round
- 3 – Bright and red
- 4 – Yellow-green blush
- 5 – Seed
- 6 – Some brown spots



Under- and Over-fitting

- What is an apple?

- ~~1 – Short stem~~
- ~~2 – Round~~
- 3 – Bright and red or green or yellow
- 4 – Yellow-green blush
- ~~5 – Seed~~
- ~~6 – Some brown spots~~

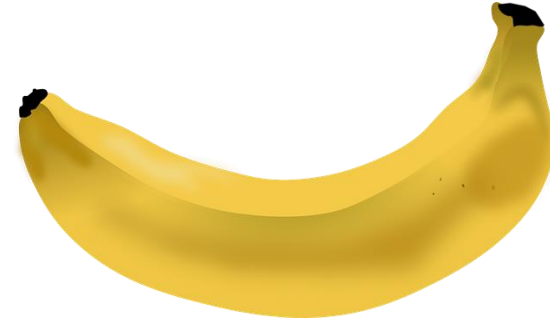


Simple model

Under- and Over-fitting

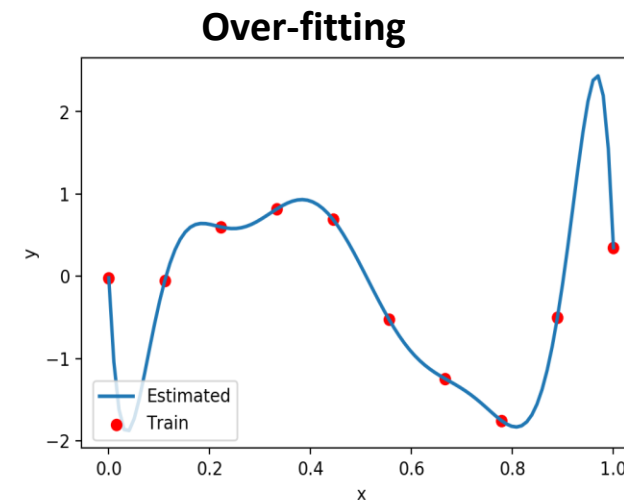
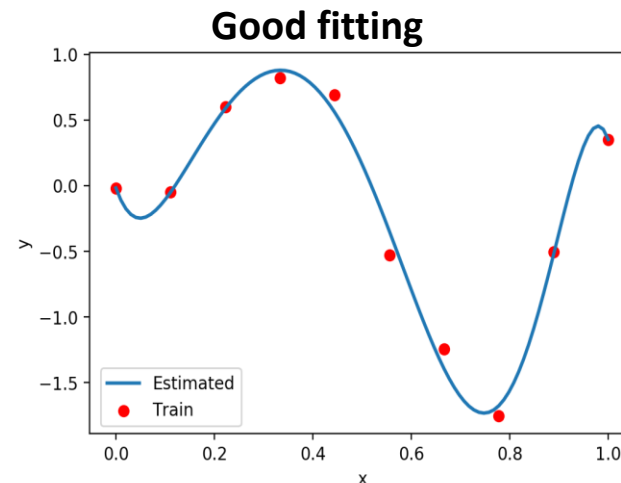
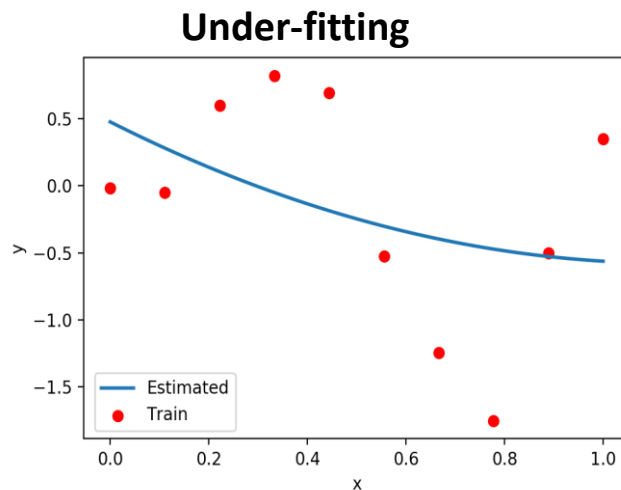
- What is an apple?

- ~~1 – Short stem~~
- ~~2 – Round~~
- 3 – Bright and red or green or yellow
- 4 – Yellow-green blush
- ~~5 – Seed~~
- ~~6 – Some brown spots~~

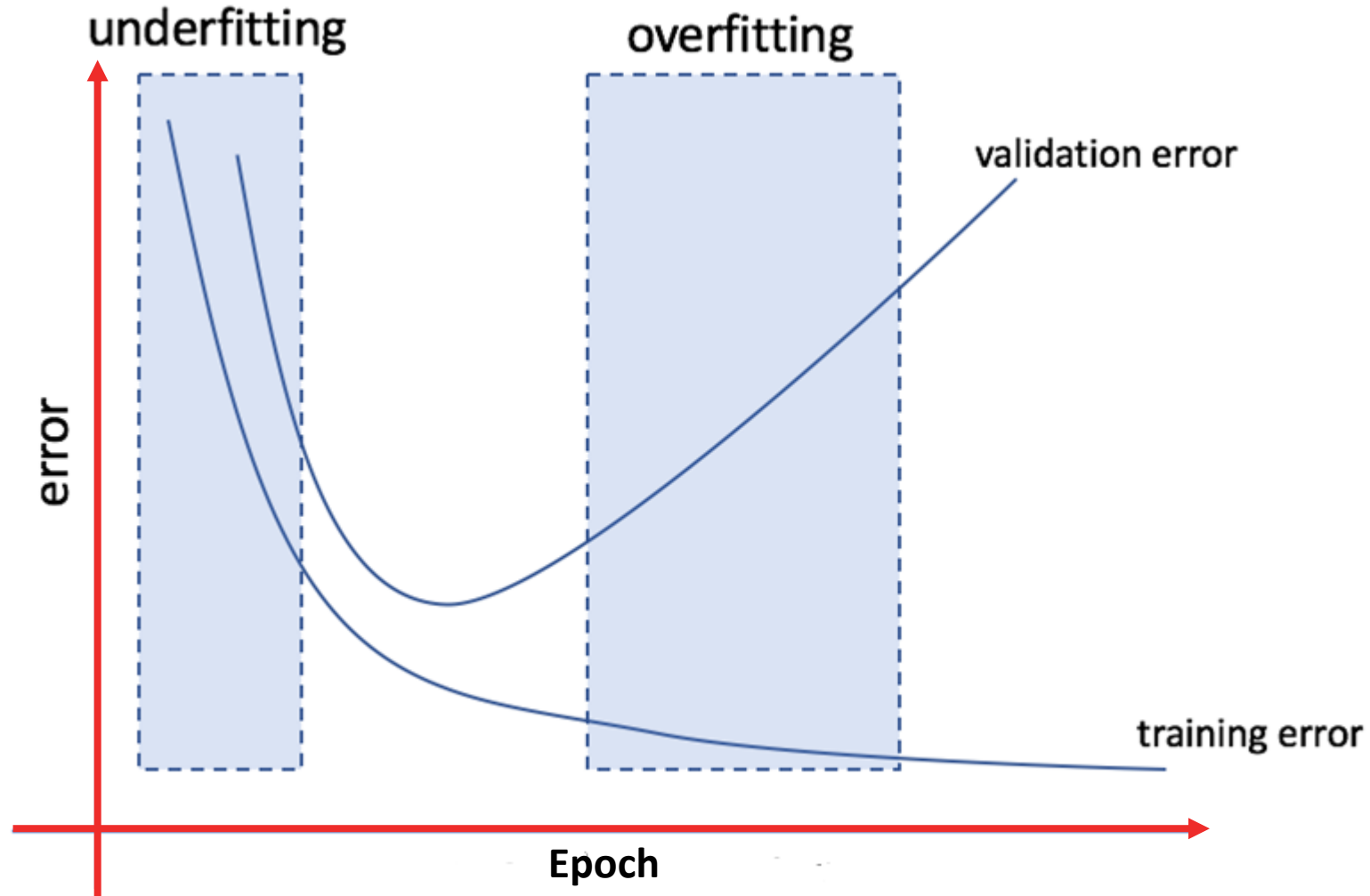


Under- and Over-fitting

- Under-fitting: too inflexible; captures no pattern
 - fitting a linear model to non-linear data
- Over-fitting: too flexible; fits to noise in the data
 - model is excessively complex ($\text{\#features} \gg \text{\#samples}$ or \#parameters too high)
 - decision boundary does not generalize \rightarrow poor results for new samples



Under- and Over-fitting

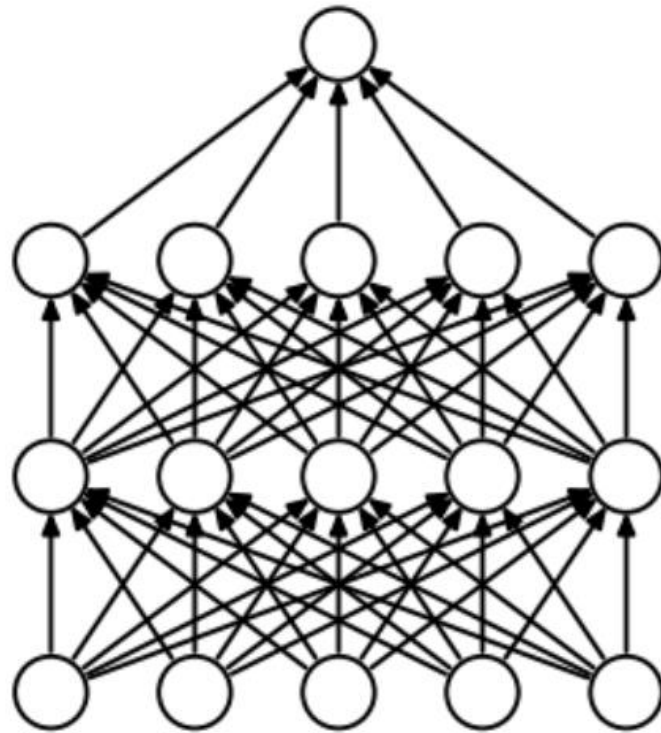


Techniques to Avoid Over-fitting

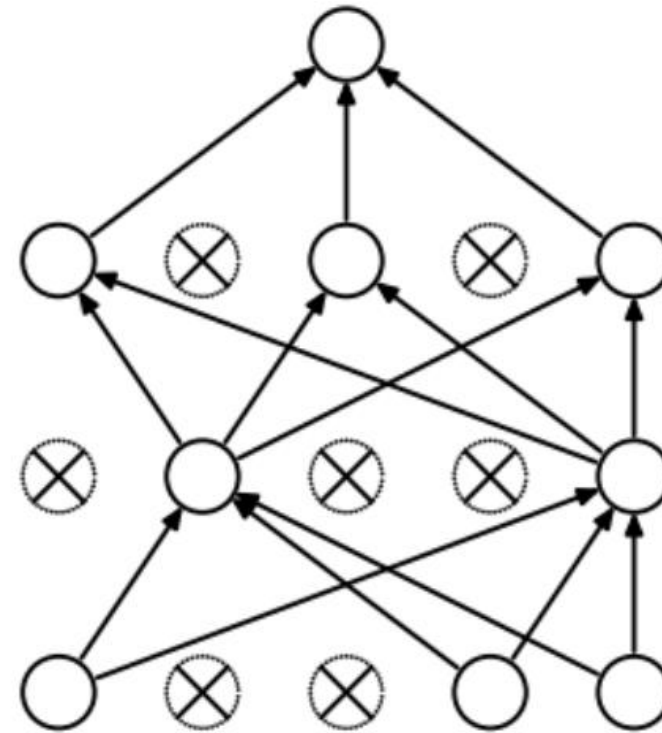
- More data
- Reduce model complexity (*i.e.*, number of trainable parameters)
- Regularization
 - Dropout
 - L1 & L2 regularization
- Data augmentation

Dropout

- Learn redundant paths -> gain robustness



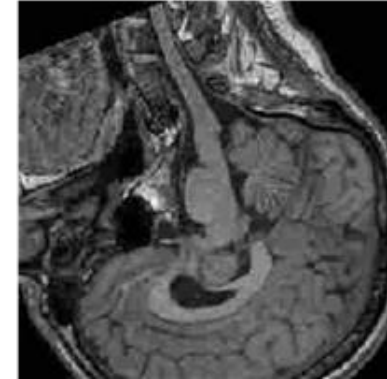
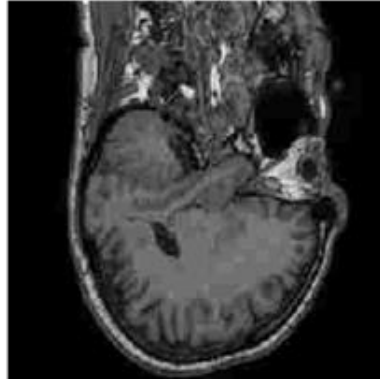
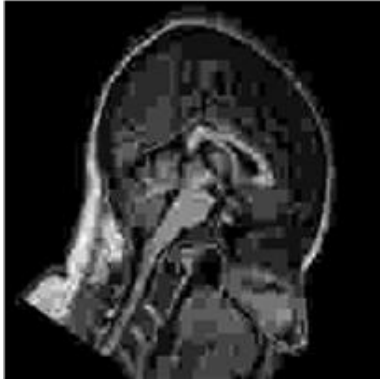
(a) Standard Neural Net



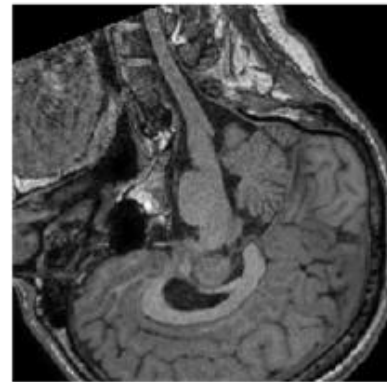
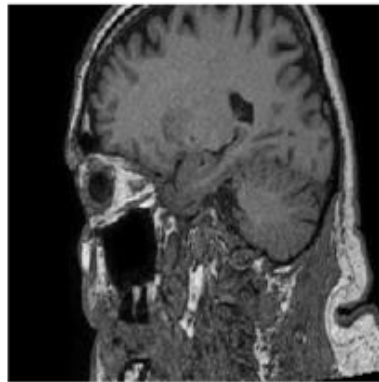
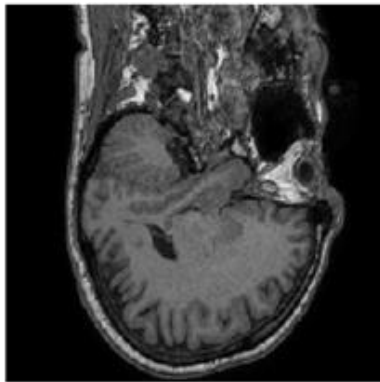
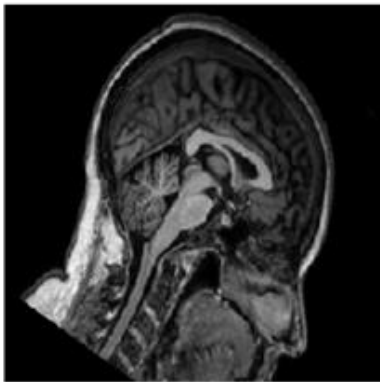
(b) After applying dropout.

Data Augmentation

- Supervised Data = Images + labels



JPEG
compressed



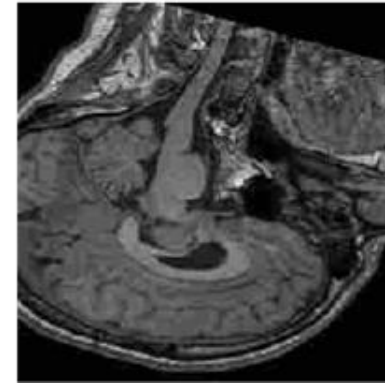
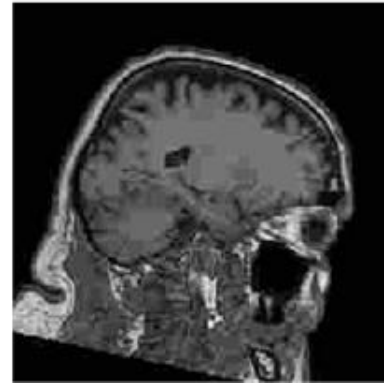
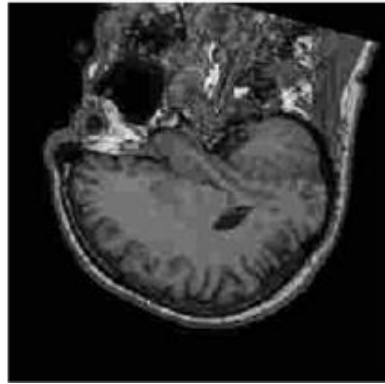
Reference

1st epoch

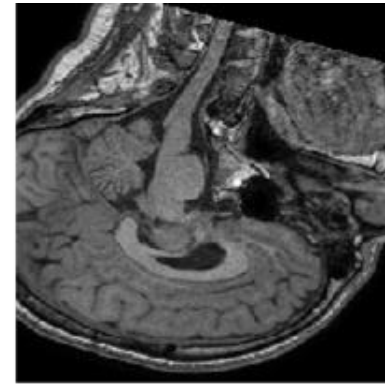
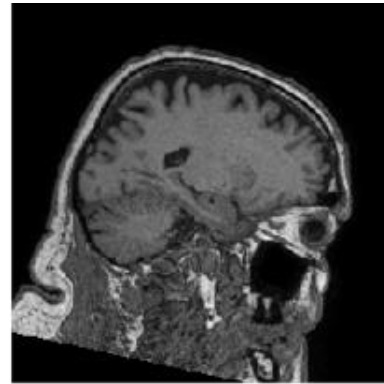
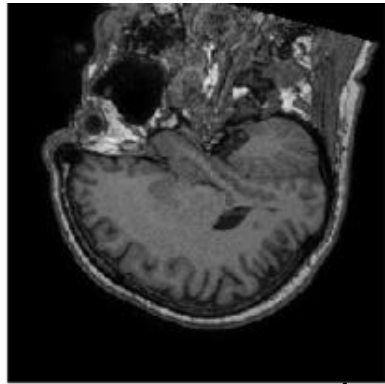
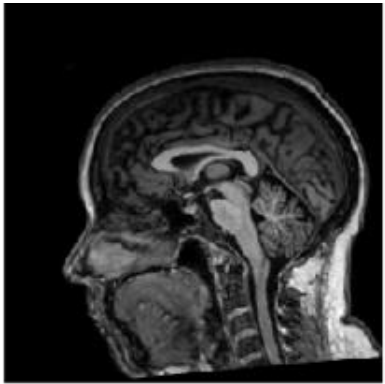
Data augmentation illustration (regression)

Data Augmentation

- Supervised Data = Images + labels



JPEG
compressed



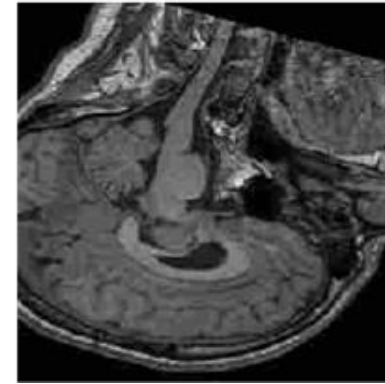
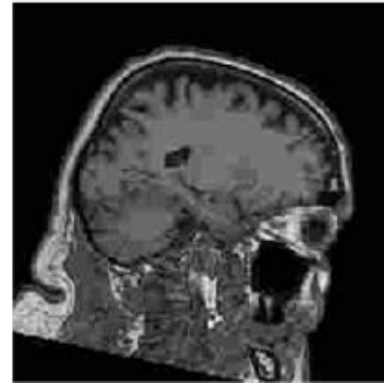
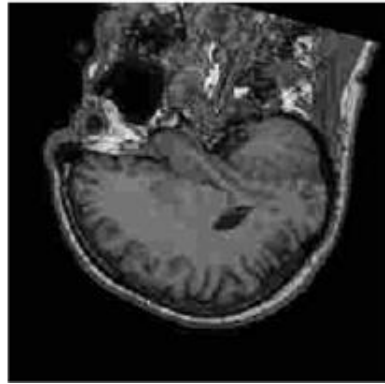
Reference

2nd epoch

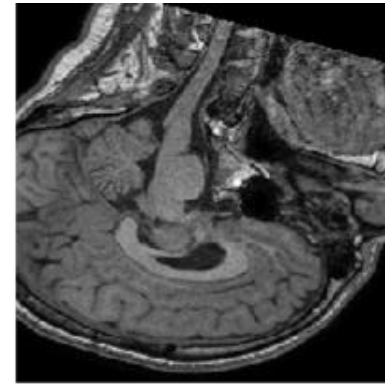
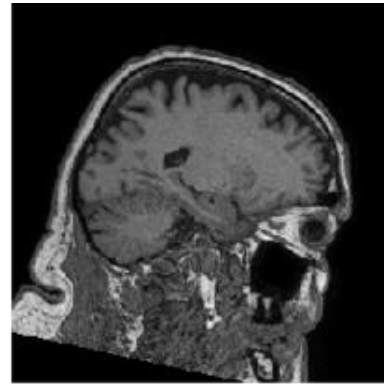
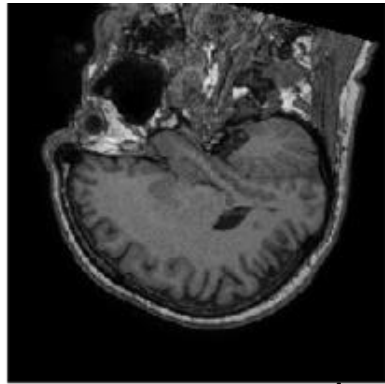
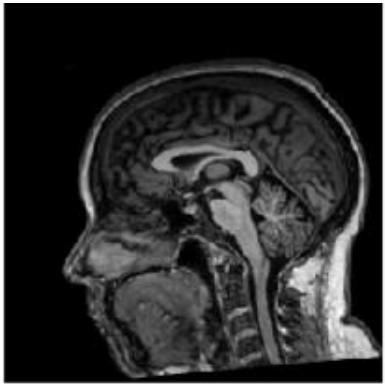
Data augmentation illustration (regression)

Data Augmentation

- Supervised Data = Images + labels



JPEG
compressed



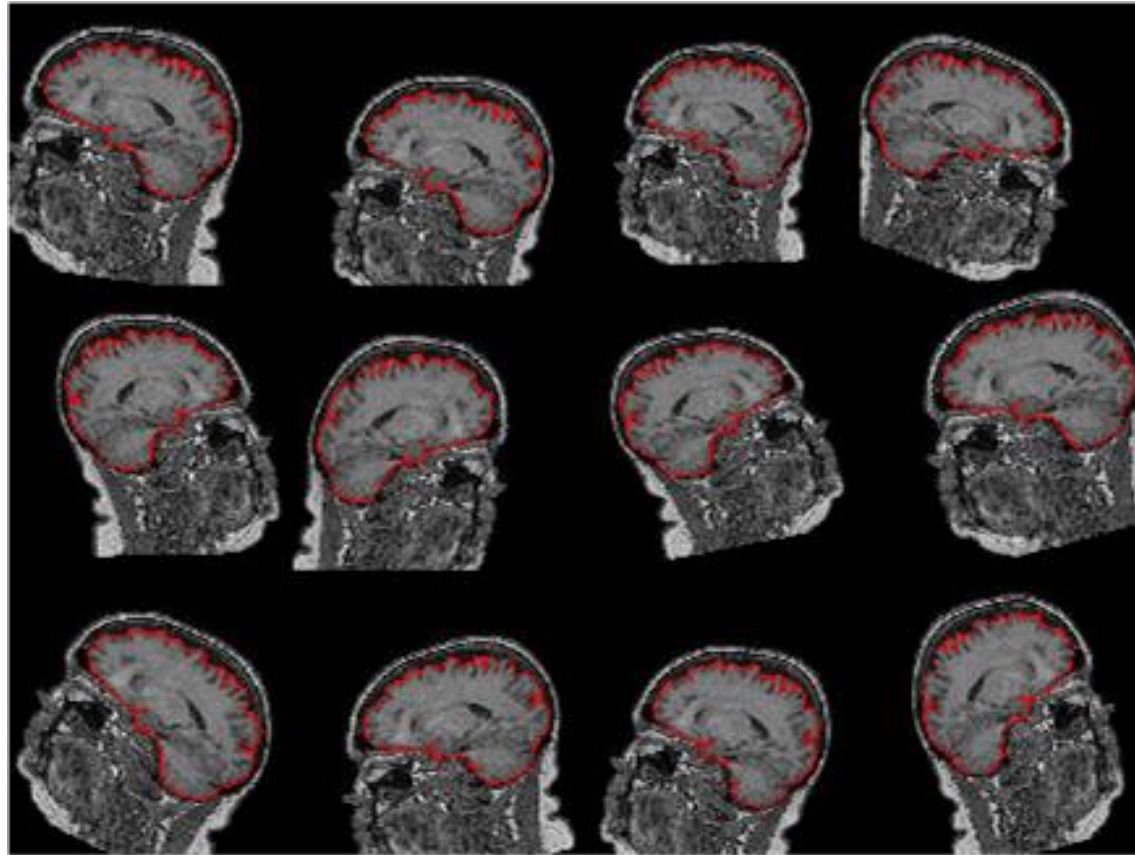
Reference

3rd epoch

Data augmentation illustration (regression)

Data Augmentation

- Supervised Data = Images + labels



Data augmentation illustration (segmentation)

L1 & L2 Regularization

- L1 regularization (Lasso)

The idea of the regularization is to penalize your model by decreasing its complexity.

L1 regularization can be seen as a feature selection because by zeroing some of the weights it can tell us what features are not important

L1 Regularization

$$\text{Cost} = \sum_{i=0}^N (y_i - \sum_{j=0}^M x_{ij} W_j)^2 + \lambda \sum_{j=0}^M |W_j|$$

Quickly to zero

- ~~1 – Short stem~~
- ~~2 – Round~~
- 3 – Bright and red or green or yellow
- 4 – Yellow-green blush
- ~~5 – Seed~~
- ~~6 – Some brown spots~~

L1 & L2 Regularization

- L2 regularization (Weight Decay)

L2 regularization is commonly known as weight decay because it shrinks the weight according to the regularize factor

L2 Regularization

$$\text{Cost} = \sum_{i=0}^N (y_i - \sum_{j=0}^M x_{ij} W_j)^2 + \lambda \sum_{j=0}^M W_j^2$$

Become smaller
(Not necessarily zero)

- 1 - Short stem x 0.1
- 2 – Round x 0.9
- 3 – Bright and red or green or yellow x 0.9
- 4 – Yellow-green blush x 0.8
- 5 – Seed x 0.3
- 6 – Some brown spots x 0.01

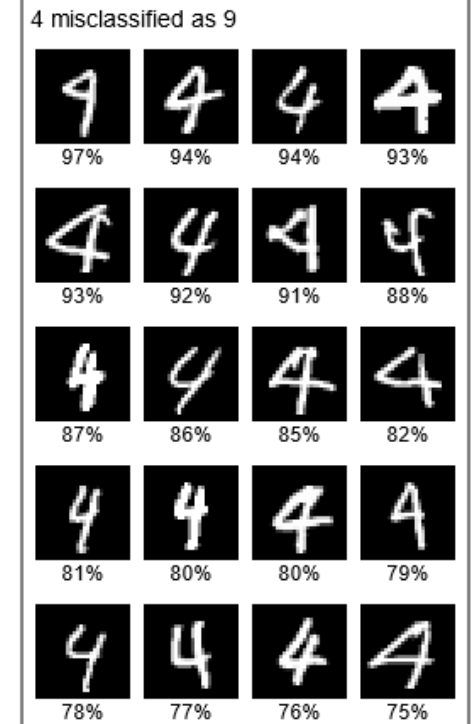
Metrics - Classification

- Confusion matrix

		Prediction outcome		
		positive	negative	
Actual value	positive	TP	FN	$TP + FN$
	negative	FP	TN	$FP + TN$
		$TP + FP$	$FN + TN$	

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

	predicted 0	predicted 1	predicted 2	predicted 3	predicted 4	predicted 5	predicted 6	predicted 7	predicted 8	predicted 9
actual 0	954	0	0	7	1	10	6	3	7	3
actual 1	0	1031	4	3	1	4	1	2	16	2
actual 2	12	21	852	18	11	8	14	20	29	5
actual 3	2	5	9	899	1	71	0	12	23	7
actual 4	2	8	2	2	861	7	7	1	4	89
actual 5	7	5	9	24	3	833	12	8	12	2
actual 6	11	6	2	0	6	31	902	0	8	1
actual 7	3	10	5	3	7	7	1	1041	0	14
actual 8	2	28	4	29	2	31	1	9	882	21
actual 9	7	3	1	7	10	11	1	44	4	873



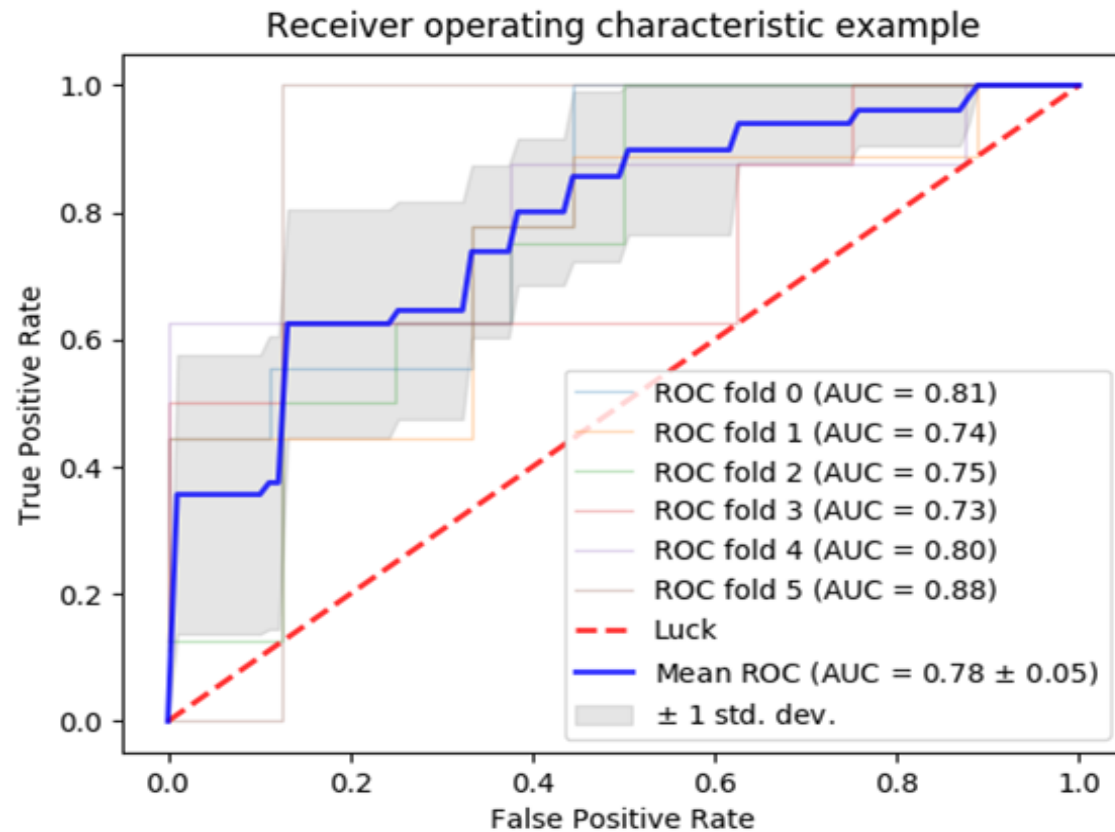
Source: Adapted from https://ml4a.github.io/demos/confusion_mnist/

Metrics - Classification

- Receiver operating characteristic (ROC) curve

$$\text{Sensitivity} = TP / P$$

$$\text{Specificity} = TN / N$$

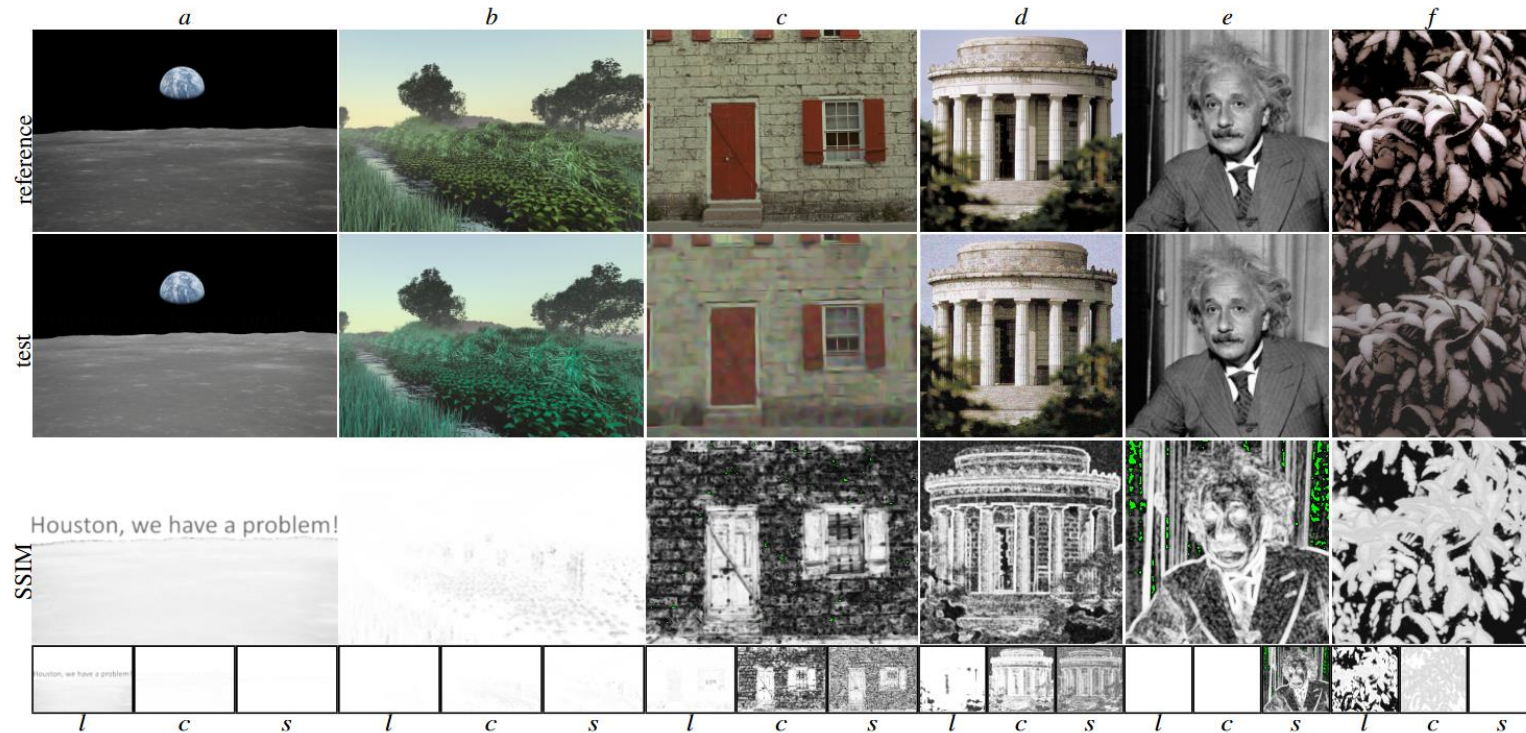


Metrics - Regression

- Structural Similarity (SSIM)

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Luminance, Contrast, and Structure



Source: [Understanding SSIM](#)

Metrics - Regression

- Normalized Root Mean Squared Error (NRMSE)

$$\text{RMSD}(\hat{\theta}) = \sqrt{\text{MSE}(\hat{\theta})} = \sqrt{E((\hat{\theta} - \theta)^2)}.$$

$$\text{NRMSE} = \frac{\text{RMSD}}{y_{\max} - y_{\min}}$$

E = target – prediction

E = 0.5 – 1.0

E = -0.5

$E^2 = 0.25$

E = 0.5

Metrics - Regression

- Peak Signal to Noise Ratio (PSNR)

$$\begin{aligned}MSE &= \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2 \\PSNR &= 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \\&= 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \\&= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE)\end{aligned}$$

E = target – prediction

$E = 0.5 - 1.0$

$E = -0.5$

$MAX = 255$

$PSNR = 20 \times \log_{10}(255) - 10 \times \log_{10}(0.25)$

$PSNR = 48.1$

Summary

- For large datasets, a single train/val/test split is often sufficient
- The validation set is used for model selection
- Overfitting makes your model less generalizable to new datasets
- Model overfitting can be mitigated by employing techniques, such as regularization

Thank you!