Geospatial Data Science
Content Block II: *Techniques*
Lab 8
<u>Machine learning for geospatial data</u>

Austin J. Brockmeier, Ph.D.

Wednesday, April 5th, 2023

Outline

Lab 8: scikit-learn, regression, neural networks, convolutional neural networks

By The scikit-learn developers - github.com/scikit-learn/scikit-learn/blob/master/doc/logos/scikit-learn-logo.svg, BSD, https://commons.wikimedia.org/w/index.php?curid=71445288

Designing a Machine Learning System

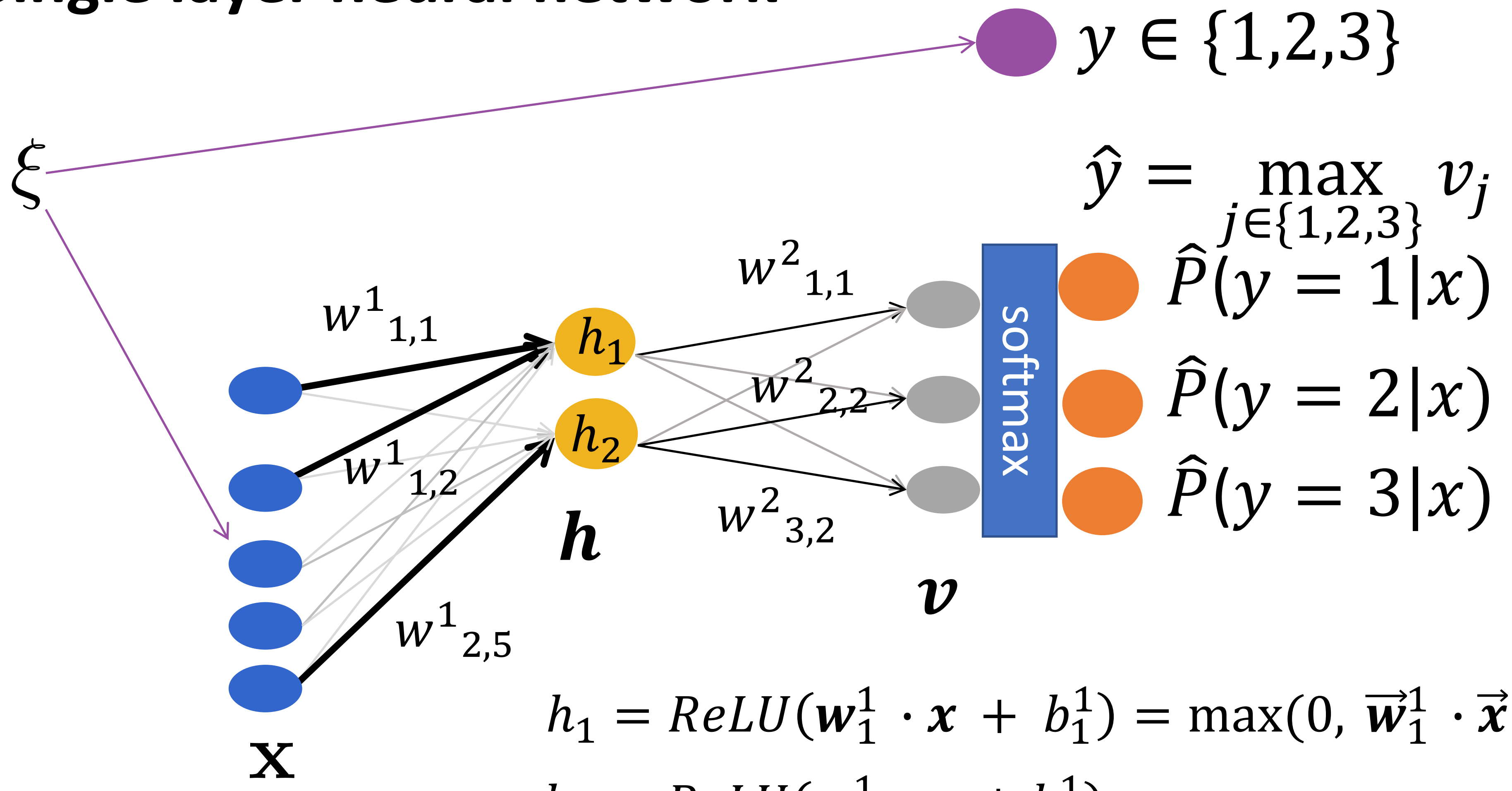
1. **Goal**: What is the task?
2. **Data**:
3. **Model**:
4. **Fitness**:
5. **Training**:
6. **Selection:**

Model: Popular choices

- **Linear model**
- k-nearest neighbor
- Decision trees
- **Random forest**, gradient boosting
- Neural networks
- Convolutional neural network
- Kernel ridge regression
- Support vector machine
- **Gaussian processes**

Artificial neural networks consist of layers of processing connected together

Single layer neural network



$$h_1 = ReLU(\boldsymbol{w}_1^1 \cdot \boldsymbol{x} + b_1^1) = \max(0, \vec{\boldsymbol{w}}_1^1 \cdot \vec{\boldsymbol{x}} + b_1^1)$$

$$h_2 = ReLU(\boldsymbol{w}_2^1 \cdot \boldsymbol{x} + b_2^1)$$

$$v_j = \boldsymbol{w}_j^2 \cdot \boldsymbol{h} + b_j^2, \text{ for } j = 1, \dots, 3$$

$$\hat{P}(y = j|x) = \frac{e^{v_j}}{\sum_k e^{v_k}}, \text{ for } j = 1, \dots, 3$$

CNN: Where’s Waldo? (Prediction yes or no for each image patch

?

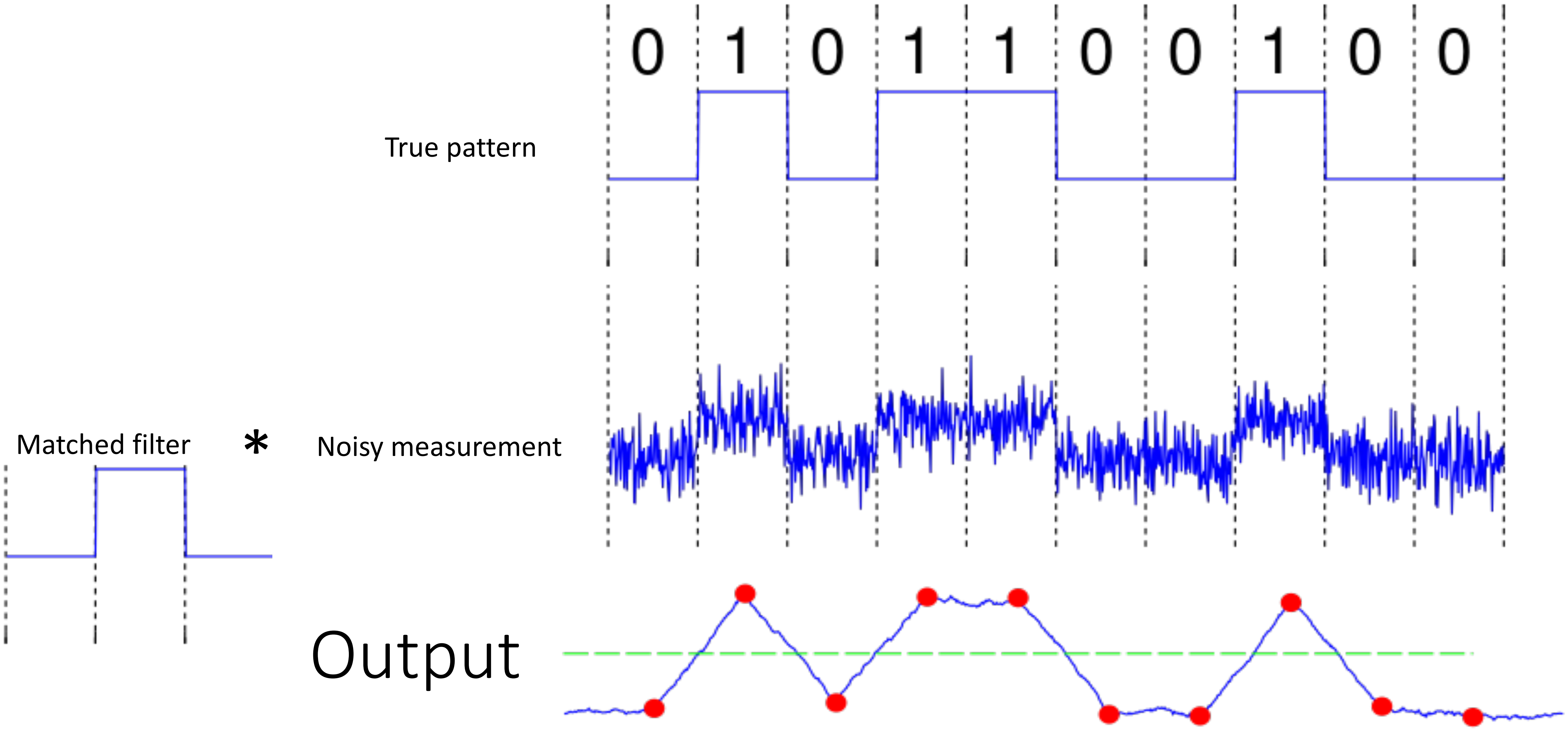
?

?

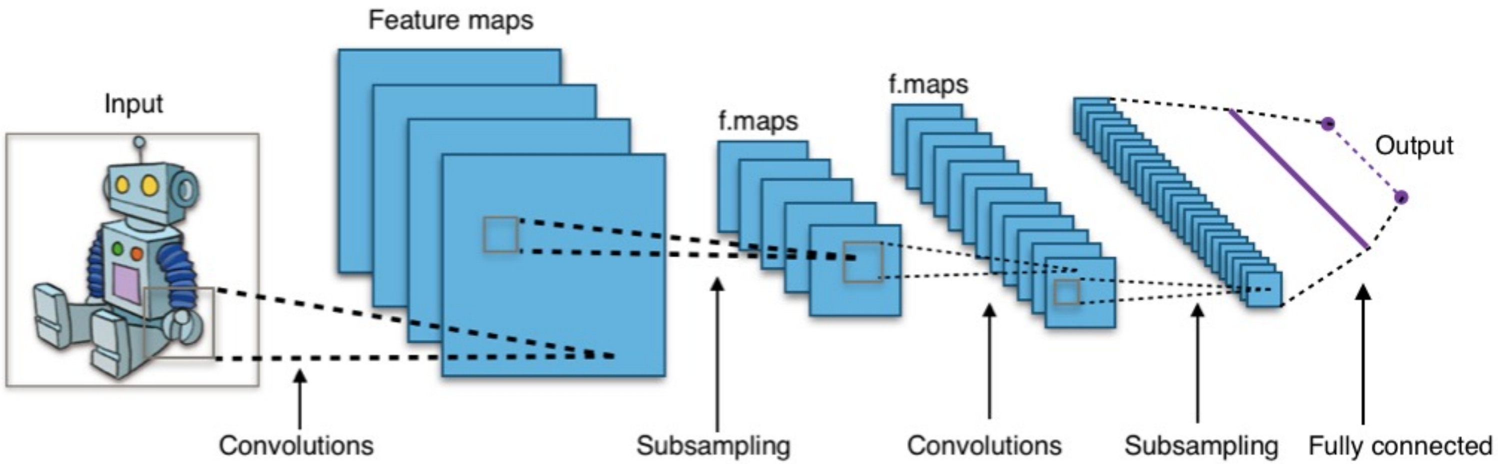
?

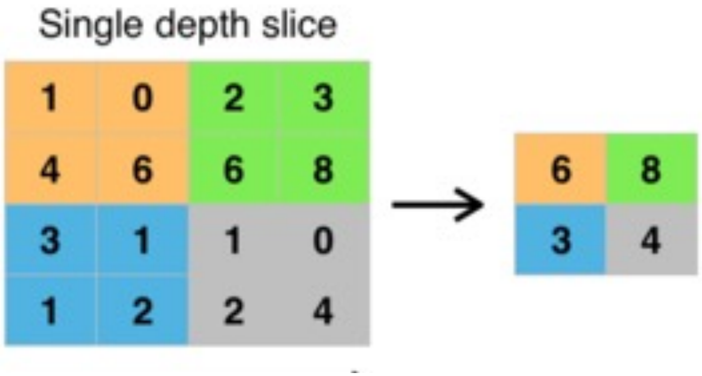
https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Signal_Processing_and_Modeling/Signals_and_Systems_%28Baraniuk_et_al.%29/13%3A_Capstone_Signal_Processing_Topics/13.04%3A_Matched_Filter_Detector

Convolution or matched filtering

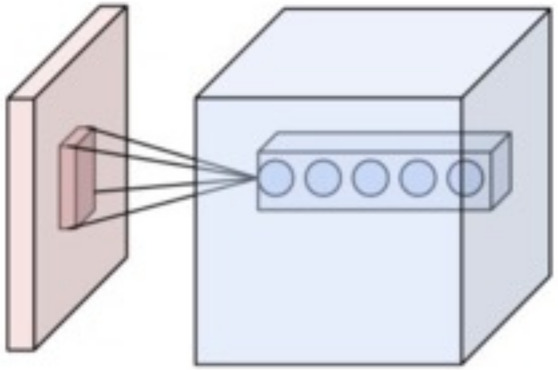


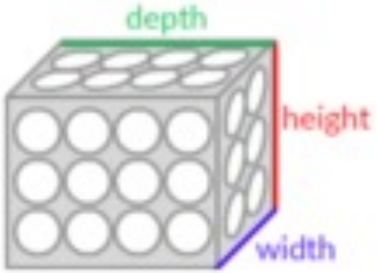
Depth is the number of channels/attributes/layers





Subsampling via
max pooling with a 2x2 filter and stride = 2

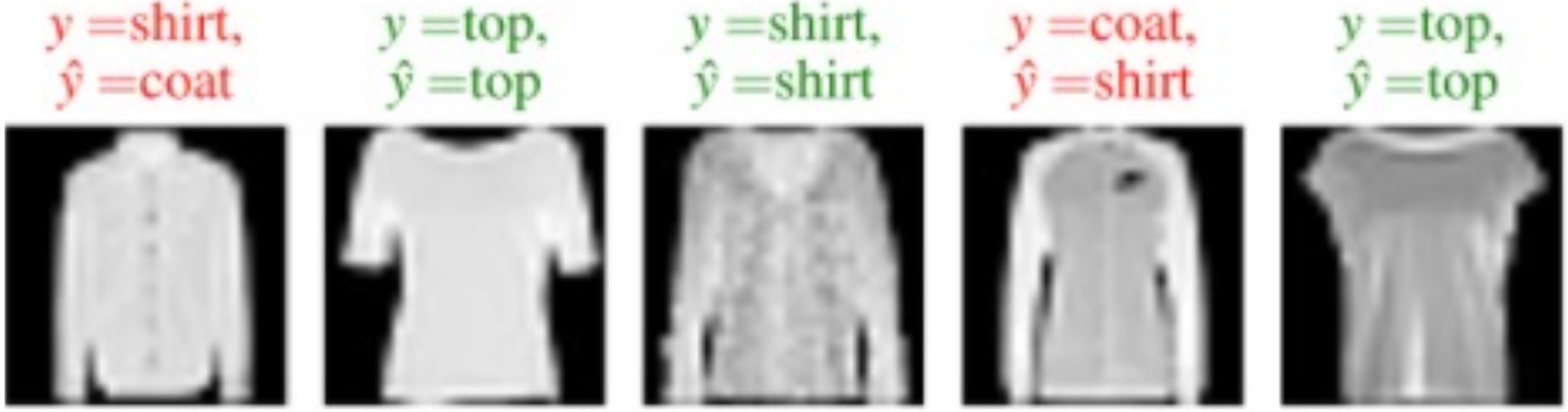




https://poloclub.github.io/cnn-explainer

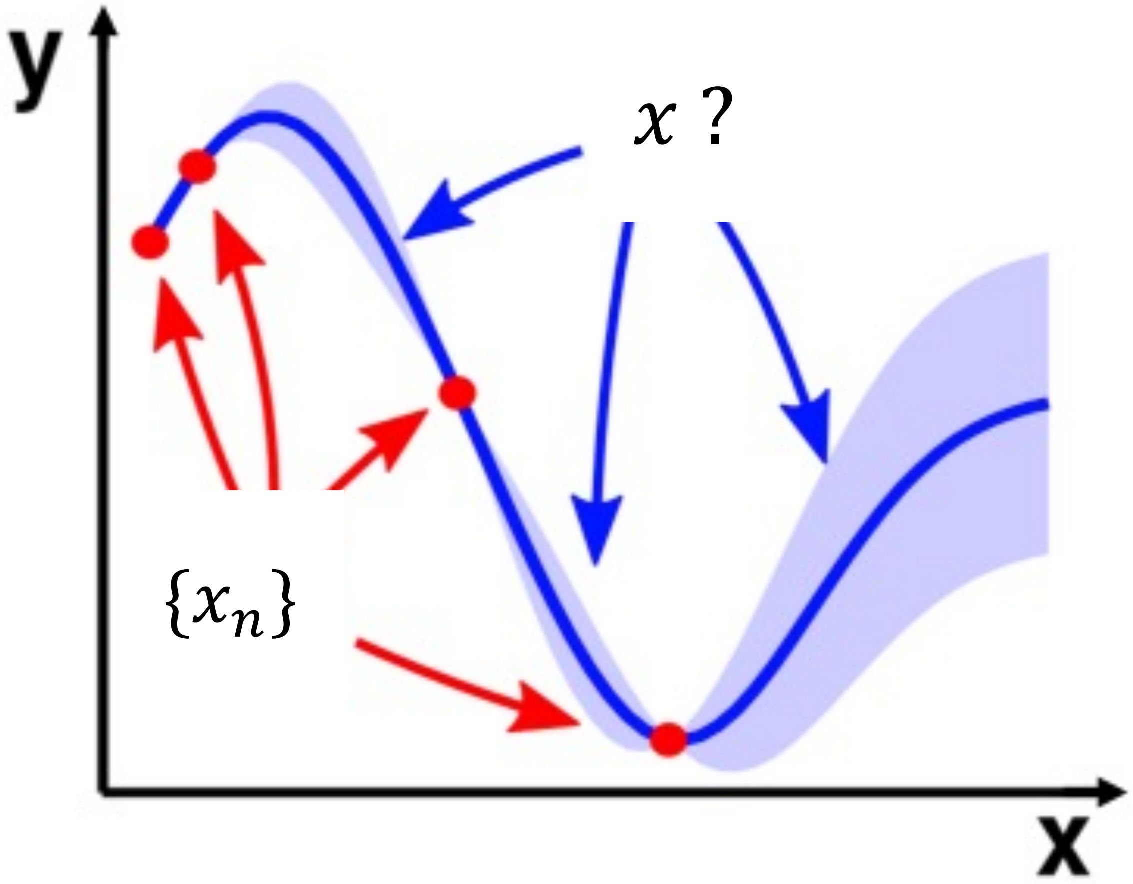
Example of CNN classifier on Fashion MNIST

- Error rate is 273/2646



Non-linear models

- k-nearest neighbor
- Decision trees, random forests, gradient boosting
- Neural networks
- Kernel ridge regression
- Gaussian process

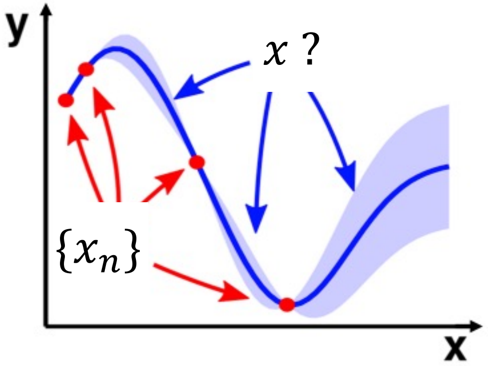


Phase 1. Fit relationship
Phase 2: Find x that gives a specific y with high confidence (near seen data) and fits constraints!

- Kernel regression
  - Advanced by Prof. Grace Wahba at UW-Madison

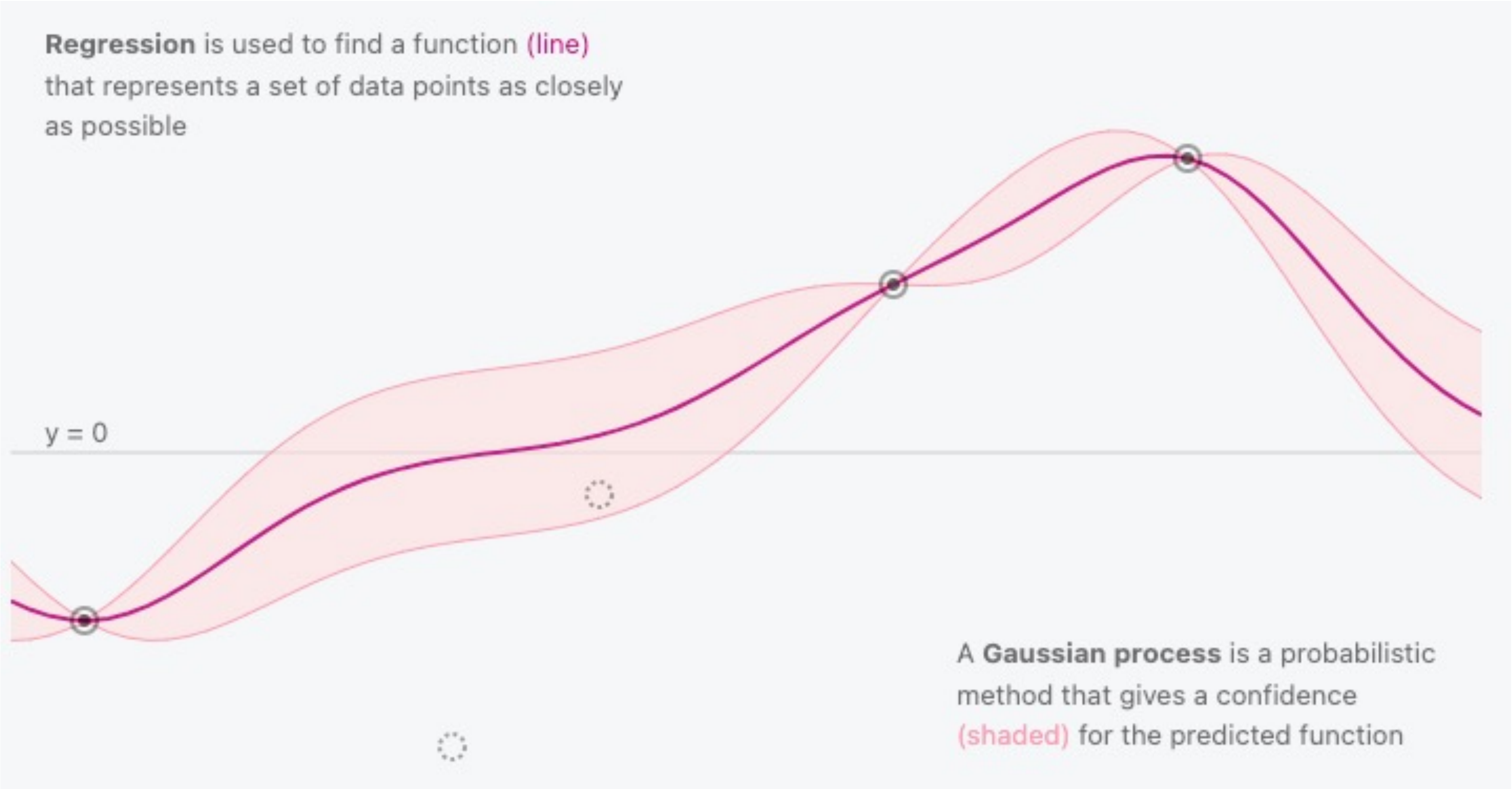
$$E[Y|x,\{(x_i,y_i)\}_{i=1}^{n}] = \bar{f}(x) = [\kappa(x,x_1),\dots,\kappa(x,x_n)]\mathbf{K}^{-1}\vec{\mathbf{y}} = \boldsymbol{K}\vec{\boldsymbol{\alpha}}$$

```
krr.fit(X,y).predict(x)
```



By The scikit-learn developers - github.com/scikit-learn/scikit-learn/blob/master/doc/logos/scikit-learn-logo.svg, BSD, https://commons.wikimedia.org/w/index.php?curid=71445288

Gaussian process (Kriging)



The predicted value at x is normally distributed with mean $f(x)$, and variance σ_x^2

$\mathcal{N}(f(x), \sigma_x^2)$

$$\sigma_x^2 = \mathrm{cov}\big(f(x), f(x)\big) = \kappa(x, x) - \, [\kappa(x, x_1), \ldots, \kappa(x, x_N)]\mathbf{K}^{-1}[\kappa(x, x_1), \ldots, \kappa(x, x_N)]$$

https://distill.pub/2019/visual-exploration-gaussian-processes/