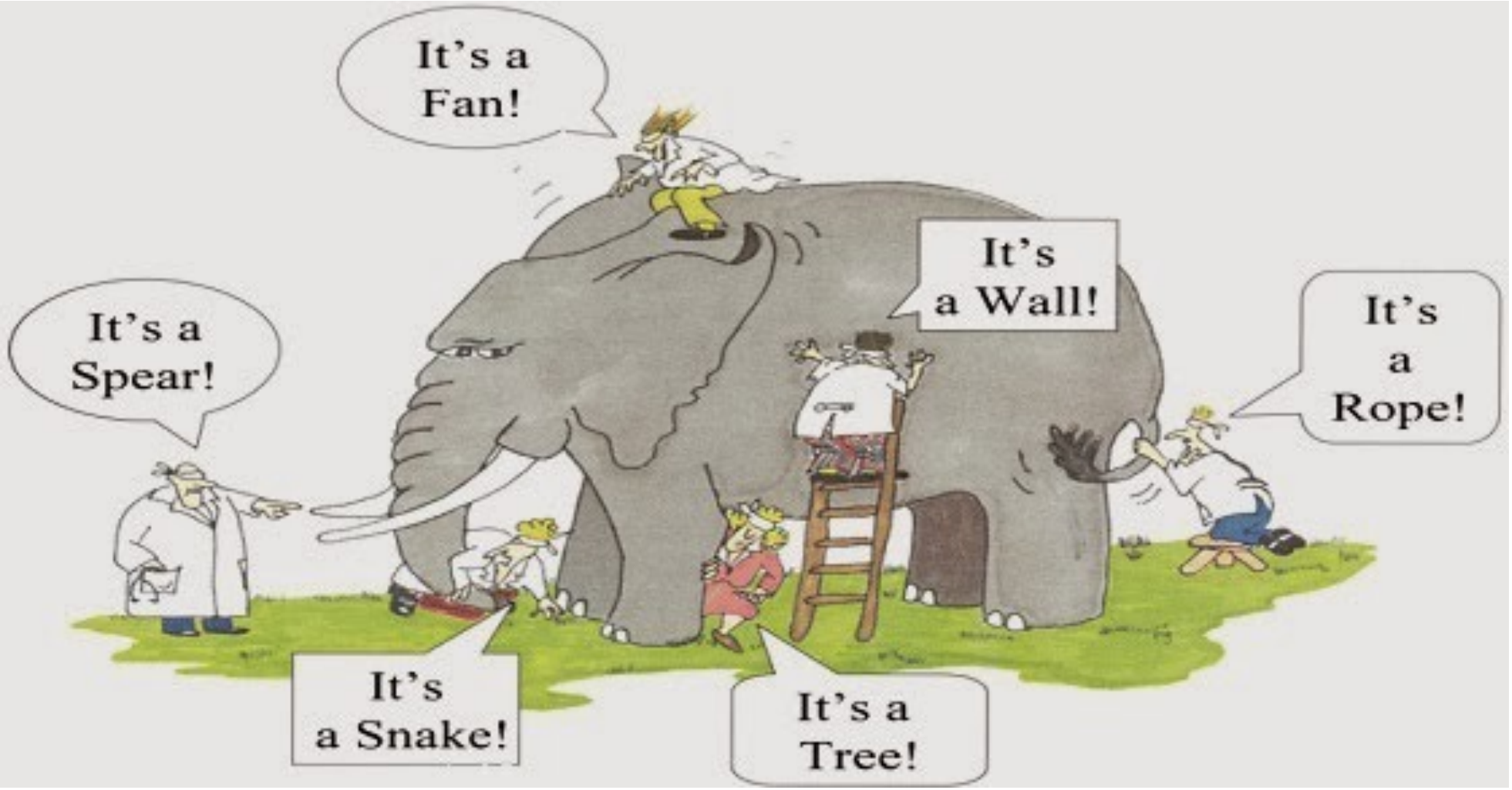

# AI/ML for Environmental Data Analytics

DS 392

Deepak Subramani
Assistant Professor
Dept. of Computational and Data Science
Indian Institute of Science Bengaluru

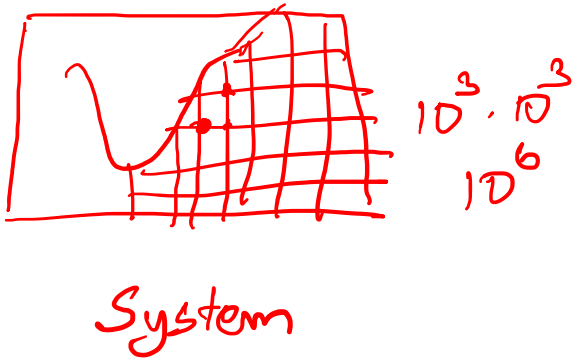
# The Proverbial Elephant




Deepak Subramani, deepakns@iisc.ac.in

# Geosciences: Data and Models

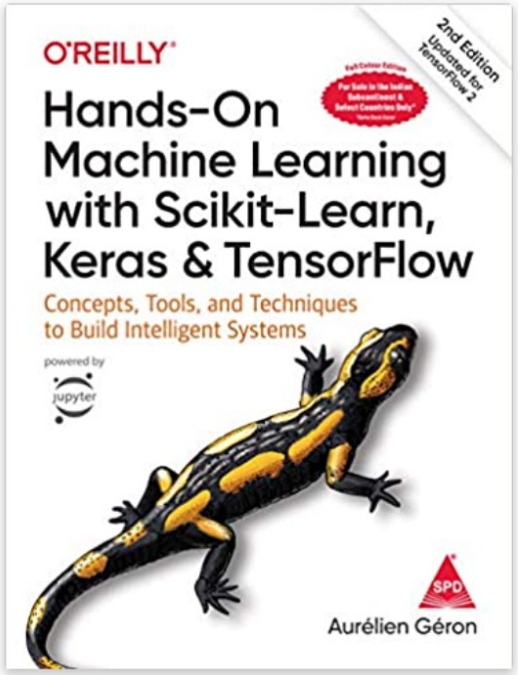
- The science of Earth – Geology, Meteorology, Oceanography, Astronomy
- Characteristics: Large spatial and temporal scales
- Let us take Oceanography
  - 7 equations – The Primitive Equations (momentum, mass, energy, height, state)
  - Bay of Bengal – 2 million sq. km. Avg Depth : 2 km
  - Data on Primitive Variables
    - If 1 km grid and 100 vertical levels $\sim 10^9$ at a time
    - 365 days a year – 4 times a day – Add 3 orders of magnitude
  - Really Big Data is needed to study
- Is data available?
  - Satellites – temporally sparse, medium spatial coverage
  - In-situ – spatially sparse, temporally dense
- Numerical models – Big Compute



# Plan

- Understand the Workflow of Machine Learning
- Go through a practical example
- Understand Regression, Classification Models

# Text Book




See this image

**Follow the Author**

Aurélien Géron + Follow

Hands-On Machine Learning with Scikit-Learn, Keras and Tensor Flow: Concepts, Tools and Techniques to Build Intelligent Systems (Colour Edition) Paperback – 23 October 2019

by Aurelien Geron (Author)

★★★★½ 405 ratings

See all formats and editions

| Kindle Edition | Paperback |
|---|---|
| ₹ 2,161.25 | **₹ 2,275.00** ✓prime |
| Read with Our **Free App** | 2 New from ₹ 2,275.00 |

FREE delivery: **Tomorrow**

Order within 1 hr and 2 mins Details

**Save Extra** with 3 offers

**No Cost EMI:** Avail No Cost EMI on select cards for orders above ₹3000 | Details

**Cashback (3):** Get Flat ₹100 back with Amazon Pay Later. Offer applicable on sign-up. Check eligibility here! | See All

See 1 more

10 Days Replacement Only | Amazon Delivered | No-Contact Delivery

Deepak Subramani, deepakns@iisc.ac.in

# Dynamical Systems Science

P
→ I  O.D.E
→ R
→ D

T.S.
ARMA


$10^9$

State Estimation
$10^{-3}$ → tempora

$$S = \begin{bmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{bmatrix}$$

$10^9$

↑ T spatial

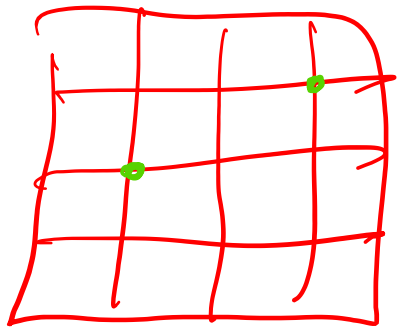
$$\Rightarrow S_{t+1} = f(S_t; \theta)$$

$$\frac{\partial S}{\partial t} = f(S; \theta)$$

A · D · R

I : V · P
B · V · P

I · B · V · P

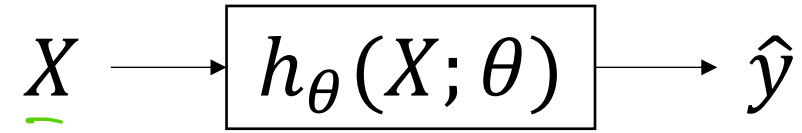


# ML: Mental Model

Data that can be collected


Quantity that must be predicted to make money

Data that can be collected




Machine's Prediction

$h_\theta$ : Machine

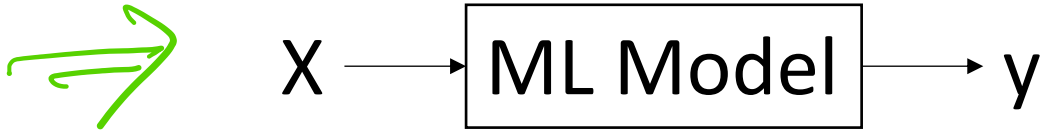
Model Selection
L.R ; NN

$\hat{y} = \theta_0 + \theta_1 x$

① What is $h_\theta$ ?

② What is $\theta$ given some $h_\theta$, $X$, $y$ ?

# Machine Learning




- Two types of general abstract setting,
  - Supervised Learning
    - Input (called features; X) and output (called labels; y) data is available
    - Need a ML model for predicting output for future/other instances of input
  - Unsupervised Learning
    - Input (called features) data – sometimes called raw data is available
    - With this “raw data”, generate some insight y
    - y can be clusters in X, density of X, is an instance of X an anomaly?

# Supervised Learning

- There are two main supervised learning tasks
  1. Regression
     - Task of predicting a continuous outcome variable (y) based on the value of one or multiple predictor variables (X)
     - Example: Market Forecasting, Population Growth Prediction, Advertising Popularity, Quantity of Sales
  2. Classification
     - Task of identifying to which category (y), among a given set, an observation (X) belongs to.
     - Example: Image Classification, Diagnostics, Customer Retention, Customer Acquisition, Whether market will go up or down, Fraud Detection

# Supervised Algorithms

- k-Nearest Neighbors
- Linear Regression
  - Polynomial Regression, Regularization
- Logistic Regression
  - Classification algorithm
- Support Vector Machines (SVMs)
  - Linear SVM, Kernel SVM
- Decision Trees and Random Forests
- Neural networks

Deepak Subramani, deepakns@iisc.ac.in

# Unsupervised Learning



- There are three main unsupervised learning tasks
  1. Clustering
     - Find pattern in data using a notion of “similarity”
     - Close to classification, but here, during training phase we do not know labels
     - Example: Customer Segmentation, Targeted Marketing, Recommender Systems
  2. Density Estimation and Anomaly Detection
     - Find the probability distribution of unlabeled data
     - Gaussian Mixture Models, ksdensity
     - Used for anomaly detection, visualization.
  3. Dimensionality Reduction
     - Reduce the feature space to a manageable more informative quantity

# Unsupervised Algorithms



- Clustering
  - K-Means
  - DBSCAN
  - GMM with EM Algorithm
- Anomaly and Novelty Detection
  - One-Class SVM
  - Isolation Forest
- Visualization and Dimensionality Redution
  - Principal Component Analysis (PCA), kernel PCA
  - Locally Linear Embedding (LLE)

# The Machine Learning Workflow

1. Frame the ML problem by looking at the science need
   a. Identify subproblems
2. Gather the data and do Data Munging/Wrangling
   a. Explore the data
   b. Clean data and prepare for the downstream ML models
3. Explore different models, perform V&V and shortlist promising candidates
4. Fine-tune shortlisted models, draw insights, and combine them together to form the final solution
5. Present your solution
   a. Say a story with the data
6. Deploy: Write a Paper; Publish Code etc

# ML Workflow: Tech Stack

1. Frame the ML problem by looking at the science need
2. Gather the data and do Data Munging/Wrangling for each subproblem
   a. **Xarray, Pandas, Numpy, Seaborn, Matplotlib, pyferret**
   b. **sklearn.preprocessing (scaler, OneHotEncoder), sklearn.impute (data cleaning, drop nan etc), custom transformers**
   c. **Spark**
3. Explore different models, perform V&V and shortlist promising candidates
   a. **sklearn.pipeline, sklearn.model_selection, sklearn.xxx (where xxx is a model), XGBoost, TF2, Keras**
4. Fine-tune shortlisted models and combine them together to form the final solution
   a. **sklearn.ensemble.VotingClassifier etc,**
5. Present your solution
   a. **PowerPoint, Seaborn, matplotlib, plotly, dash, javascript (fusion charts, react, d3), Special Packages/Software, overleaf**
6. Deploy
   a. **Google Cloud Platform, AWS SageMaker**

# Human-Elephant Conflict

Science News

**Asian elephants may lose up to 42 percent of habitats in India and Nepal**

Date: February 28, 2019

Source: Forschungsverbund Berlin

Summary: Protecting and expanding suitable habitats for wildlife is key to the ... dangered species, but owing to climate and land use chan... day may not be fitting in 30 or 50 years. An interna... predicted range shifts of Asian elephan... models based on distributio...

Home / Bangladesh / Nation

**Farmer trampled to death by elephant in Chittagong**

Pimple Barua, Chittagong

Published at 09:49 am November 14th, 2021


WILDLIFE & BIODIVERSITY

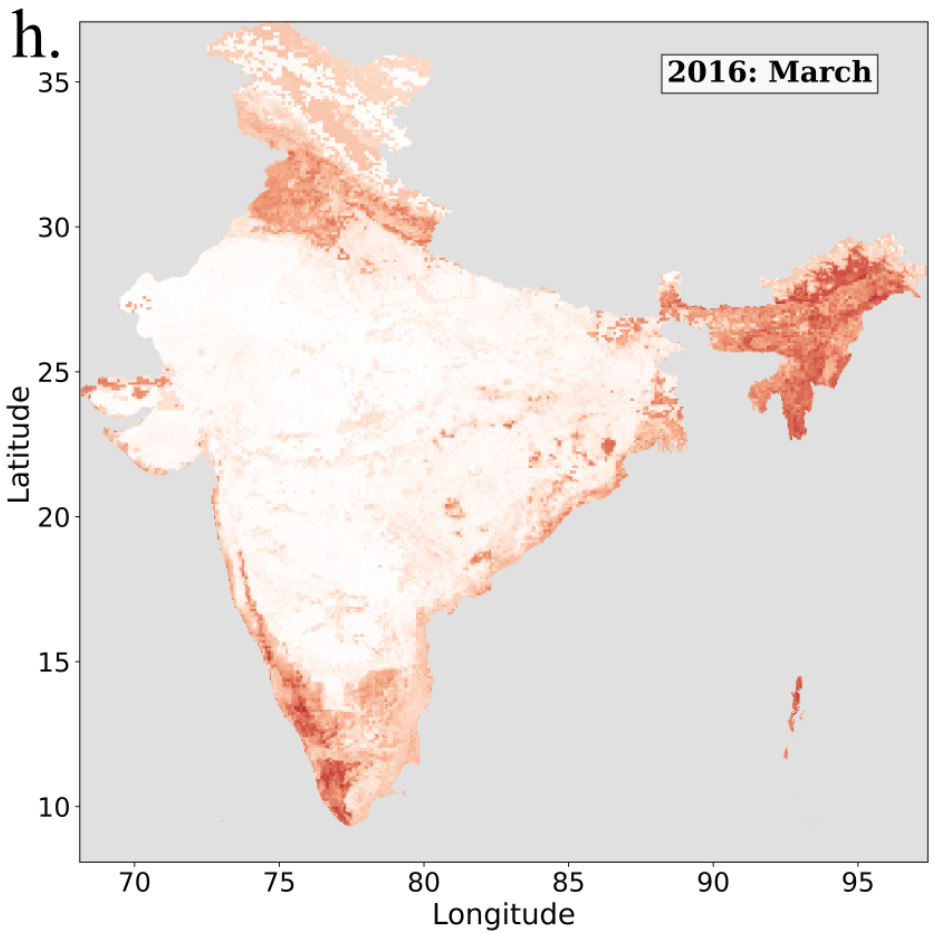
**On same day, 3 elephants die in 2 Odisha districts**

The development has raised concerns among forest officials and animal conservationists

By Ashis Senapati
Published: Monday 25 October 2021

# Data Science Problem: Predict Habitat Suitability

## Machine Learning for Asian Elephant Habitat Suitability Estimation in India



# Data Munging

1. Data Cleaning → In this step the primary focus is on
    1. Handling missing data
    2. Handling noisy data
    3. Detection and removal of outliers
2. Data Integration → Gather from various data sources and combine them before cleaning
3. Data Transformation → Convert the raw data into a specified format for feeding to downstream ML models.
    1. Normalization
    2. Aggregation
    3. Standardization
4. Data Reduction → Following data transformation and scaling, the redundancy w ithin the data is removed and is organized efficiently.

# Data Munging

- Use Xarray and netcdf files

# ML Workflow Step 2: Data Munging

- Needs practice
- There are some guidelines – which are abstract ideas
- In all labs, spend time to do the data munging

# Predictors and Targets

Input to the ML

| Category | Variable | Source | Unit | Spatial resolution |
|---|---|---|---|---|
| Climatic | Monthly precipitation | https://www.worldclim.org | mm | 2.5 minute |
| | Monthly minimum temperature | https://www.worldclim.org | °C | 2.5 minute |
| | Monthly maximum temperature | https://www.worldclim.org | °C | 2.5 minute |
| Topographic | Elevation above sea level | https://www.worldclim.org | m | 30 arc-seconds |
| | Distance to rivers and water-bodies | Derived using QGIS with data downloaded from https://www.openstreetmap.org | m | |
| | Distance to roads | Derived using QGIS with data downloaded from https://www.openstreetmap.org | m | |
| | Land Use Land Cover (LULC) | https://bhuvan.nrsc.gov.in | categorical | 30m |
| Vegetation related | Net Primary Productivity (NPP) | https://neo.sci.gsfc.nasa.gov | $gC/m^2/day$ | 0.1 degrees |
| | Leaf Area Index (LAI) | https://neo.sci.gsfc.nasa.gov | $m^2/m^2$ | 0.1 degrees |
| | Normalized Difference Vegetation Index (NDVI) | https://neo.sci.gsfc.nasa.gov | Dimensionless | 0.1 degrees |

Target of the ML

GBIF Global Biodiversity Information Facility

Asian Elephant Presence Data

# Step 3: Explore different ML Models

1. Linear Regression with Regularization
   - For regression task – predicting a continuous variable
2. Logistic Regression
   - For classification tasks alone
3. Support Vector Machines – Linear and Kernel
4. Decision Tree
5. Random Forests
6. K-Nearest Neighbours
   - Instance-Based Method
7. Naïve Bayes
   - Simplest Bayesian Network Model
8. Neural Networks

# Regression Models

- Predict the value of a continuous variable in a supervised setting
- Use other variables as predictors

$$\hat{y} = h_\theta(X; \theta)$$

- Variable with a hat, $\hat{y}$ is usually the prediction from a model
- $h$ is called the hypothesis, or the ML Model parametrized by $\theta$
- $\theta$ can be calculated, estimated or ***learned*** from data by solving the optimization problem

$$\min \sum_{j=1}^{m} \left(\hat{y}^{(j)} - y^{(j)}\right)^2$$

# Training, Validation, Testing

- In all settings, developing a ML model involves two stages
  1. Training-Validation
     - Training: Find the parameters of the ML model
       - Parameters are trained from data directly by an optimization algorithm
     - Validation is to see if parameters found during training are generalizing well to data not seen
     - Validation is used to tune the hyperparameters of the ML model
       - These are set by users and not directly computed by the optimization algorithm from data
  2. Testing
     - Testing is similar to validation, but the performance at test stage is not generally used to improve the model
     - Test data is completely unseen during model development

# Train-Test Split – 2 Strategies

- We must separate the dataset to training and testing sets
  - Training set is used for building the model and testing set it used to evaluate model performance
  - The train set can be further split to training set and validation set
- Test set should not be used for model building purpose

**Agnostic Random Split**

```
from sklearn.model_selection import train_test_split
train_set, test_set = train_test_split(dataset, test_size=0.2,random_state=42)
```

**Split to avoid sampling Bias**

```
from sklearn.model_selection import StratifiedShuffleSplit
split = StratifiedShuffleSplit(n_splits=1, test_size=0.2, random_state=42)
for train_index, test_index in split.split(dataset, dataset[shuffle_category]):
        strat_train_set = dataset.loc[train_index]
        strat_test_set = dataset.loc[test_index]
```

# Success Metrics, Validation of ML Models

- Success Metrics are defined based on the task we perform
- Regression
  - Mean Squared Error and Root Mean Squared Error
  - Mean Absolute Error
  - Mean Relative Absolute Error
  - $R^2$ for linear regression problems [Very common in statistics, and early days]
  - Advanced use: Maximum error, MSE only for some part of the data, etc

# Cross-Validation

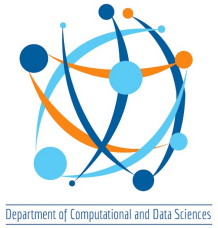
- Cross-Validation involves using the train set to construct multiple train-validate splits and perform the model building/training activity
- Test set should not be used for model building purpose

- K Fold Cross Validation
  - from sklearn.model_selection import cross_val_score
  - scores = cross_val_score(tree_reg, housing_prepared, housing_labels, scoring="neg_mean_squared_error", cv=10)
  - tree_rmse_scores = np.sqrt(-scores)

- GridSearchCV for Hyper Parameter Tuning

# Classification Models

- Other than Regression, where we predict continuous values, Classification, where we predict classes is an important supervised learning task
- Classification models are data-driven models that enable us to distinguish which class a data point belongs to.
- Binary Classifier: A data-driven model that identifies if a particular combination of attributes (i.e., data) belongs to one class (1) or not (0)
- Evaluating a classifier is the tricky part which we will dive deep into first.




# Success Metrics, Validation of ML Models


- Classification
  - Accuracy
  - Recall
  - Precision
  - F1 Score (Jaccard Index)
  - Receiver Operator Characteristics
  - Cross Entropy

# Classification: Performance Measures

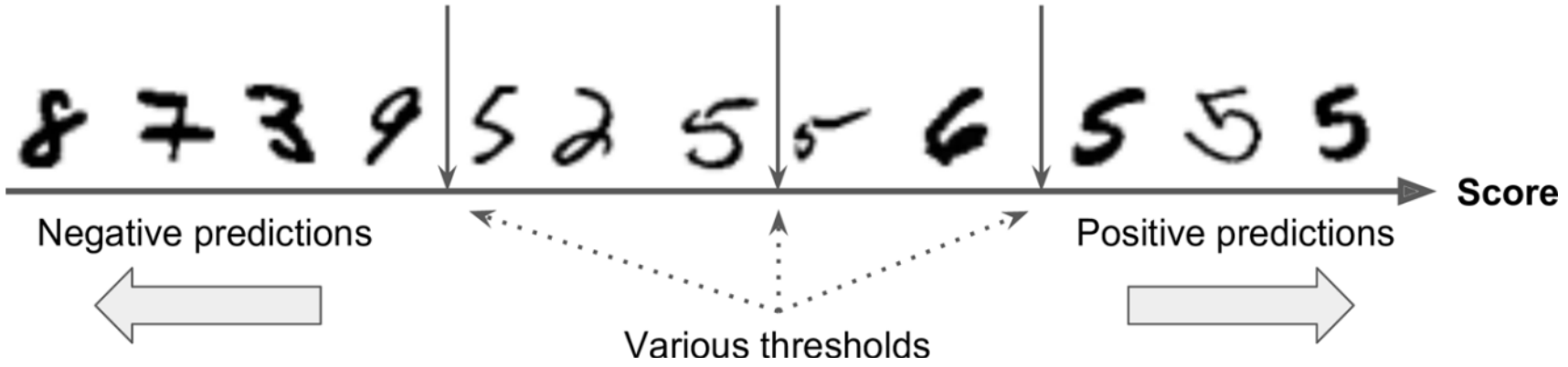
- Can we use plain accuracy?
- Let us design a classifier (i.e., a medical diagnostic test) for Covid-19.
  - The test is such that it simply return –ve for everyone
  - What will be the accuracy? – Greater than 90% as only 1% of the population currently has ever had the disease, and only about 5% of those testing are +ve
  - So is our classifier good? Surely >90% accuracy is good? No? Why?
- Accuracy is generally not enough for use as a performance measure for classifiers, especially for skewed datasets where some classes are more frequent than others (as in Covid-19)

# Confusion Matrix

| | True Label A | True Label ~A |
|---|---|---|
| Pred. Label A | True Positive | False Positive |
| Pred. Label ~A | False Negative | True Negative |

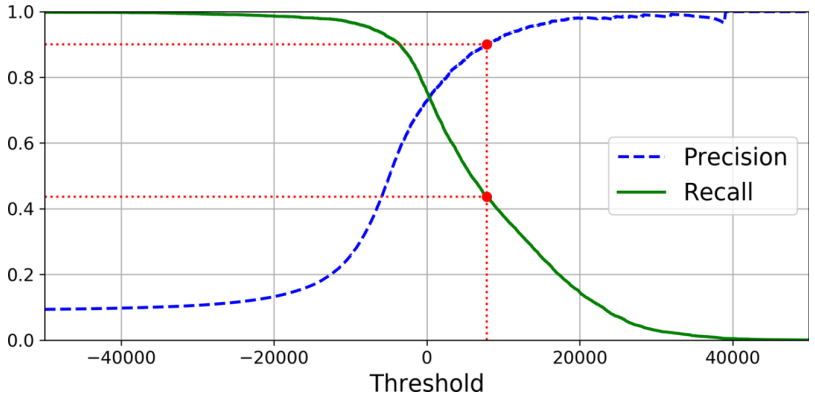
- Precision = $\frac{TP}{TP+FP}$;
  - Among all prediction of Label A, how many are actually Label A
  - Trivial 100% Precision – Make only one Prediction of Label A, and ensure that it is correct. Then TP=1, FP=0, and Precision=1
- Recall = $\frac{TP}{TP+FN}$;
  - Of all true Label A, how many does our classifier predict as Label A
  - Combined with Precision, we now have a good sense of the goodness of our classifier
- Typically, we want high precision and high recall
- F1 Score = $\frac{2}{\frac{1}{precision}+\frac{1}{recall}} = \frac{TP}{TP+\frac{FN+FP}{2}}$
- F1 Score is the harmonic mean of precision and recall. Hence F1 will be high only if precision and recall are high

# Precision Recall Tradeoff

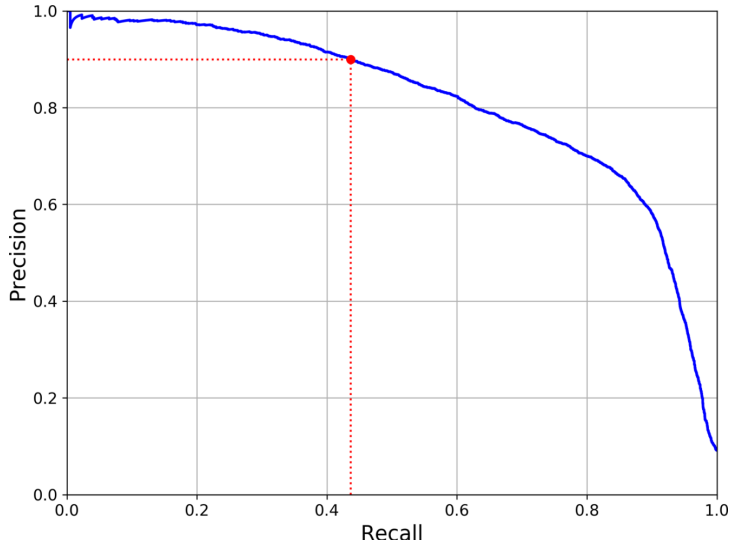


# Precision-Recall Tradeoff

| | True Label A | True Label ~A |
|---|---|---|
| Pred. Label A | True Positive | False Positive |
| Pred. Label ~A | False Negative | True Negative |




- If you label almost all as A (low threshold), then recall is high, but precision is poor
- If you label almost all as ~A (high threshold), then precision is potentially high (or Not defined), but recall is nearly zero
- If TP=0, both are zero
- For different thresholds, we usually plot both precision and recall vs threshold.
- We can also plot Precision vs Recall
- Choose the right threshold from the above curve and settle for a Precision-Recall tradeoff




Deepak Subramani, deepakns@iisc.ac.in

Plots from Geron Chapter 3

# Other Definitions

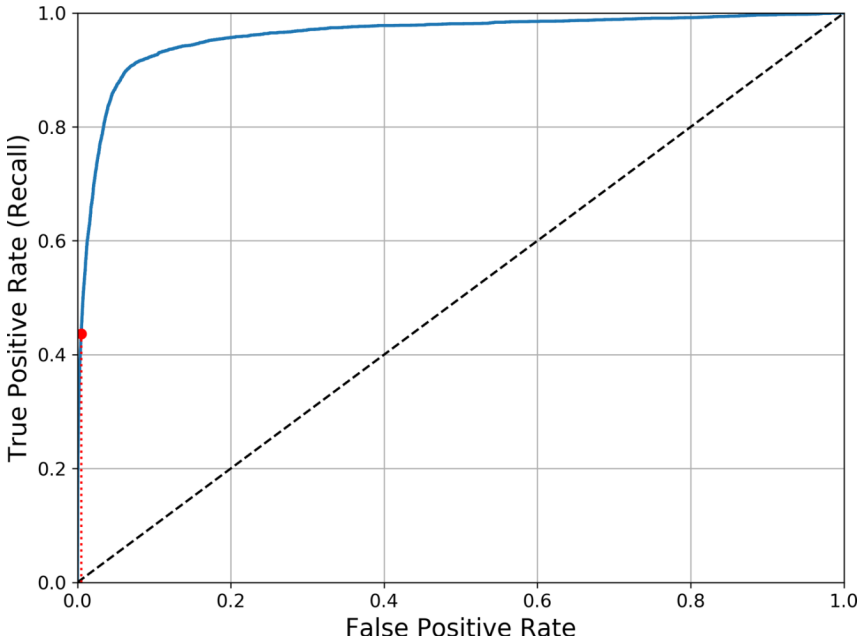
| | | True condition | | | | |
|---|---|---|---|---|---|---|
| | Total population | Condition positive | Condition negative | Prevalence = $\frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$ | Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$ | |
| Predicted condition | Predicted condition positive | **True positive** | **False positive**, Type I error | Positive predictive value (PPV), Precision = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Predicted condition positive}}$ | False discovery rate (FDR) = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Predicted condition positive}}$ | |
| | Predicted condition negative | **False negative**, Type II error | **True negative** | False omission rate (FOR) = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Predicted condition negative}}$ | Negative predictive value (NPV) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Predicted condition negative}}$ | |
| | | True positive rate (TPR), Recall, Sensitivity, probability of detection, Power = $\frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$ | False positive rate (FPR), Fall-out, probability of false alarm = $\frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$ | Positive likelihood ratio (LR+) = $\frac{\text{TPR}}{\text{FPR}}$ | Diagnostic odds ratio (DOR) = $\frac{\text{LR+}}{\text{LR−}}$ | $F_1$ score = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$ |
| | | False negative rate (FNR), Miss rate = $\frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$ | Specificity (SPC), Selectivity, True negative rate (TNR) = $\frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$ | Negative likelihood ratio (LR−) = $\frac{\text{FNR}}{\text{TNR}}$ | | |

Confusion Matrix; Wikipedia

Deepak Subramani, deepakns@iisc.ac.in

# Receiver Operating Characteristic (ROC) Curve

- True Positive Rate (TPR) – Recall or Sensitivity
- True Negative Rate (TNR) – Specificity – Ratio of negative instances correctly classified as negative
- False Positive Rate = 1 – True Negative Rate
- ROC plot is sensitivity vs 1-specificity, i.e., TPR vs FPR
- Compute TPR and FPR for different thresholds and plot it
- The ROC must be as much away from the 45-degree line as possible
- Calculate the Area Under the Curve (AUC) for quantifying goodness of the classifier
- AUC=1 is a perfect classifier, and AUC=0.5 for a random classifier



# PR vs ROC – Which to Use?

- As a rule of thumb, prefer the PR curve whenever
  - the positive class is rare or
  - when you care more about the false positives than the false negatives.
- Otherwise, use the ROC curve.

# Main Challenges of ML

- Insufficient Training Data
- Non-representative/Biased/Skewed Training Data
- Poor Quality/Error Prone Data (Missing entries etc)
- Overfit or Underfit