MACHINE LEARNING PRINCIPLES

SICSS-NDSU

https://slides.com/zoltanpm/sicss-ndsu-ml

OVERVIEW

Statistical learning

Basic steps: data segmentation

Metrics

Traditional ML vs. deep learning

”

Modeling for explanation or modeling for prediction.

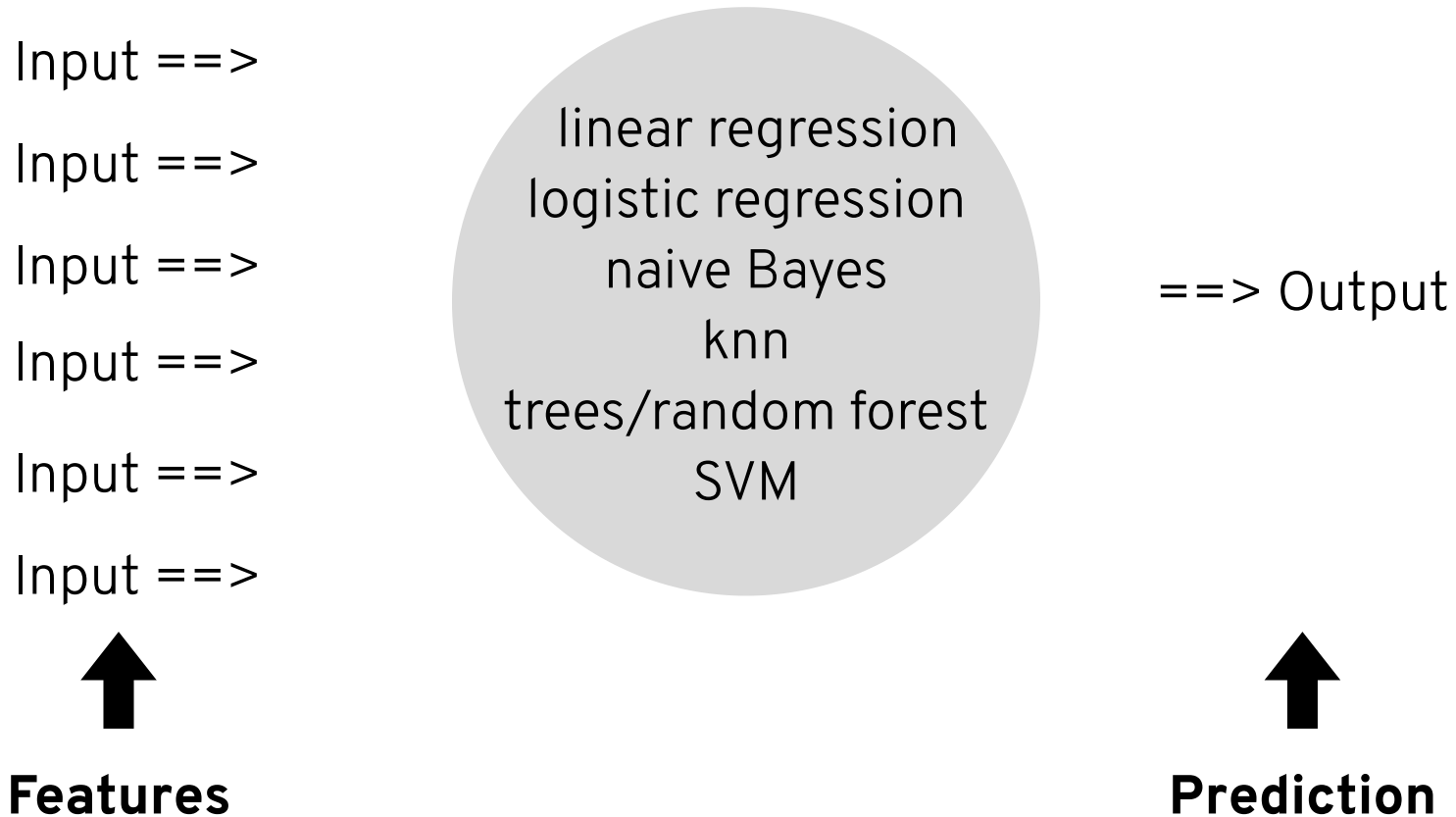
- ModernDive

https://moderndive.com/5-regression.html

SOME KEY POINTS

- ML *code* is hard. But good ML work = team work, and there are critical roles for non-coders
- ML is linear algebra + calculus + probability theory + statistics + SME
- ML is not AI is not Deep Learning
- ML iterates over data many times, making it susceptible to amplifying biases/imbalances in the data
- *Never trust a model where accuracy is only measured on predictions of its own training data*

ML IS STATISTICAL LEARNING



ML IS


DATA
(INPUT)
Program
Traditional
Programming
Output
DATA
(INPUT)
Output
Machine
Learning
Program

DATA PRE-PROCESS

TRAINING

Human-coded data using discipline-specific standards for IRR/ICC

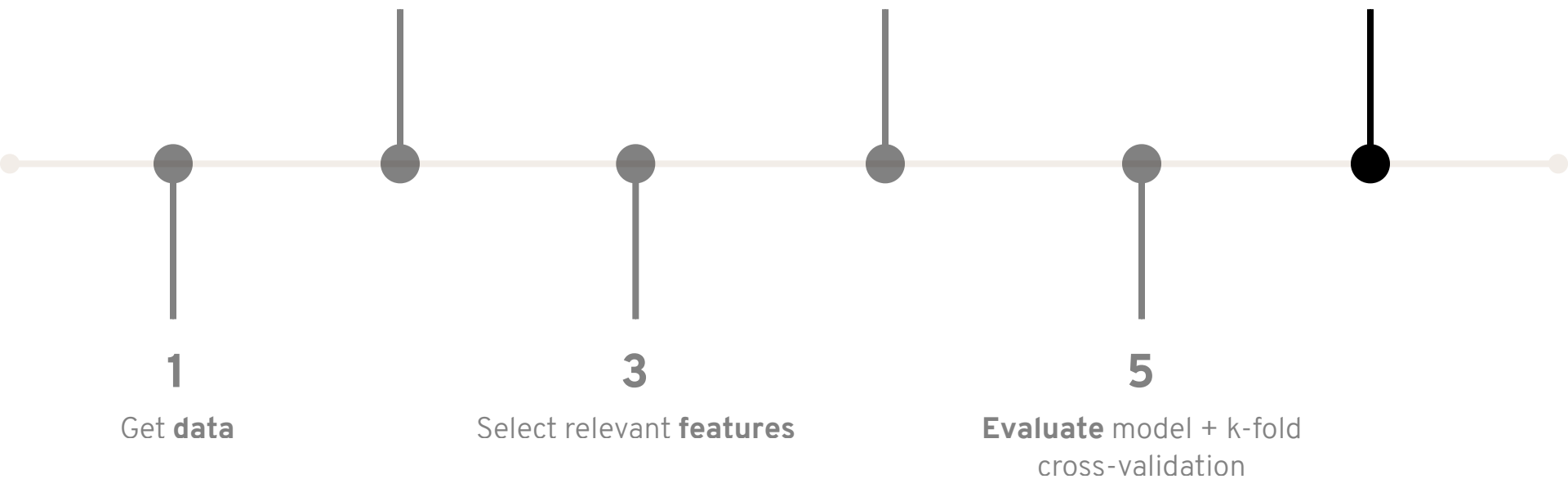
VALIDATING

Slice of human-coded data used for internal validations during training runs

TESTING

Data that's never been exposed to the model; special considerations for timeseries

STEPS


2
Preprocess
4
Train model
6
Predict/test predictions
1
Get **data**
3
Select relevant **features**
5
Evaluate model + k-fold cross-validation

METRICS

- Confusion Matrix
- Recall/Sensitivity ~ Precision/Specificity
- Accuracy ... eh (*Accuracy Paradox*)
- ROC/AUC
- F1

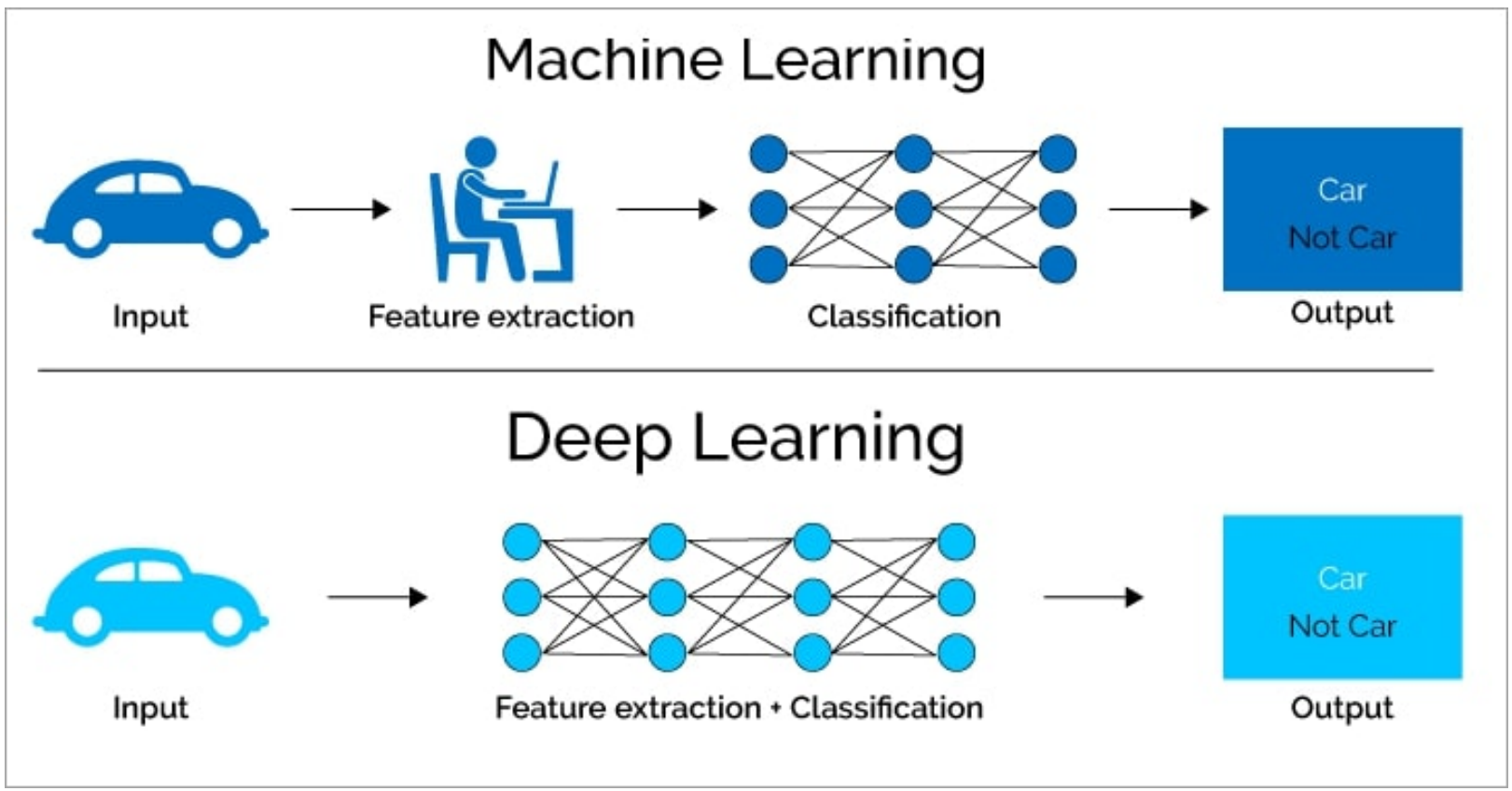
all images from

https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

TRADITIONAL ML VS. DEEP LEARNING: THE SICSS-RUTGERS EXAMPLE

```r
model1 <- glm(outcome ~ lengthoftweet +
                nnegative +
                nprofanity +
                capitals,
              data = training_hate,
              family=binomial(link = "logit"))
```

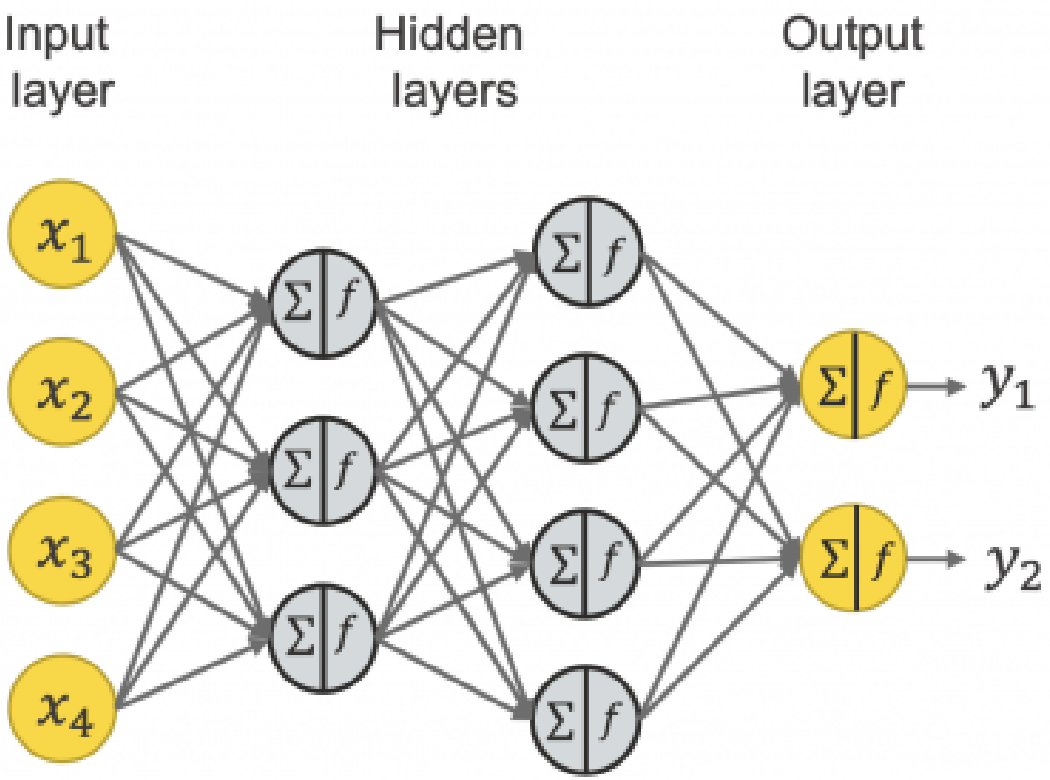
TOWARD DEEP LEARNING



from

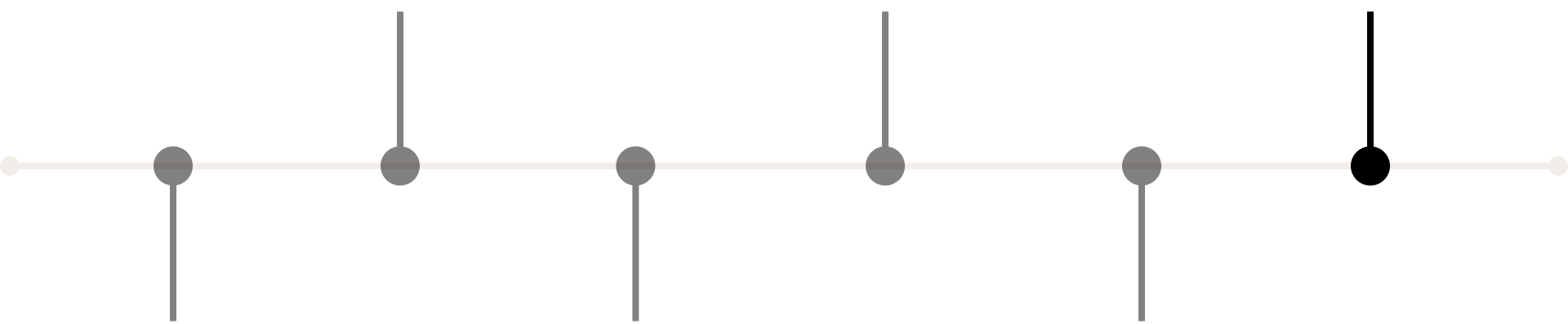
https://levity.ai/blog/difference-machine-learning-deep-learning

TOWARD DEEP LEARNING



from
https://levity.ai/blog/difference-machine-learning-deep-learning,
https://theconversation.com/deep-learning-and-neural-networks-77259
https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks

STEPS



1

Get **data**

2

Preprocess

3

Pick a model (or start from scratch)

huggingface.co

kaggle

etc.

4

Fine-Tune/Train model

5

Evaluate model + tune hyperparameters

6

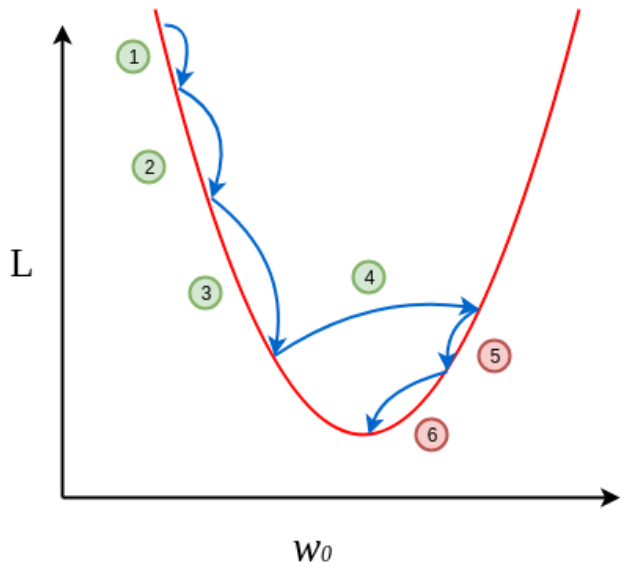
Predict/test predictions

NOT BINARY

Full Sentence Inputs	Feature Engineering	Features for Expertise Classification?
computing-intensive	more efficient	
more variable	more intelligible	
easier	requires expertise	
removes judgment	requires judgment	

HOW IT LEARNS

- Backpropagation
- Learning rate and gradient descent



from https://towardsdatascience.com/gradient-descent-explained-9b953fc0d2c