



NATIONAL RESEARCH
UNIVERSITY

School of Data Analysis and Artificial
Intelligence Department of Computer Science

DATA SCIENCE FOR BUSINESS

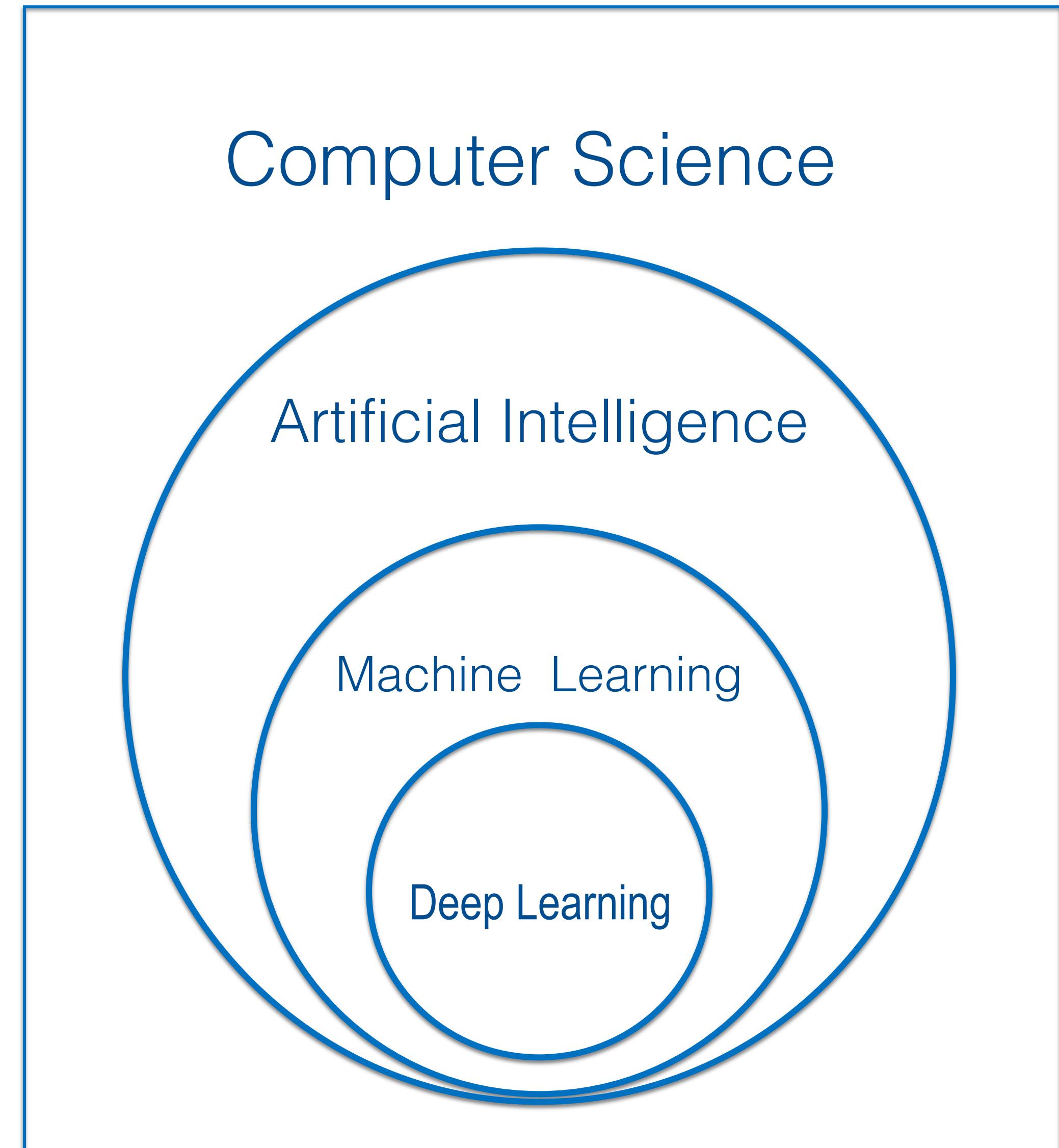
Lecture 2. Introduction to Machine Learning

Moscow, April 15th, 2022.

WHAT IS MACHINE LEARNING?

Machine learning is a subfield of computer science that studies and develops algorithms that can learn from data without being explicitly programmed

Machine learning algorithms can detect patterns in data and use them to predict future data

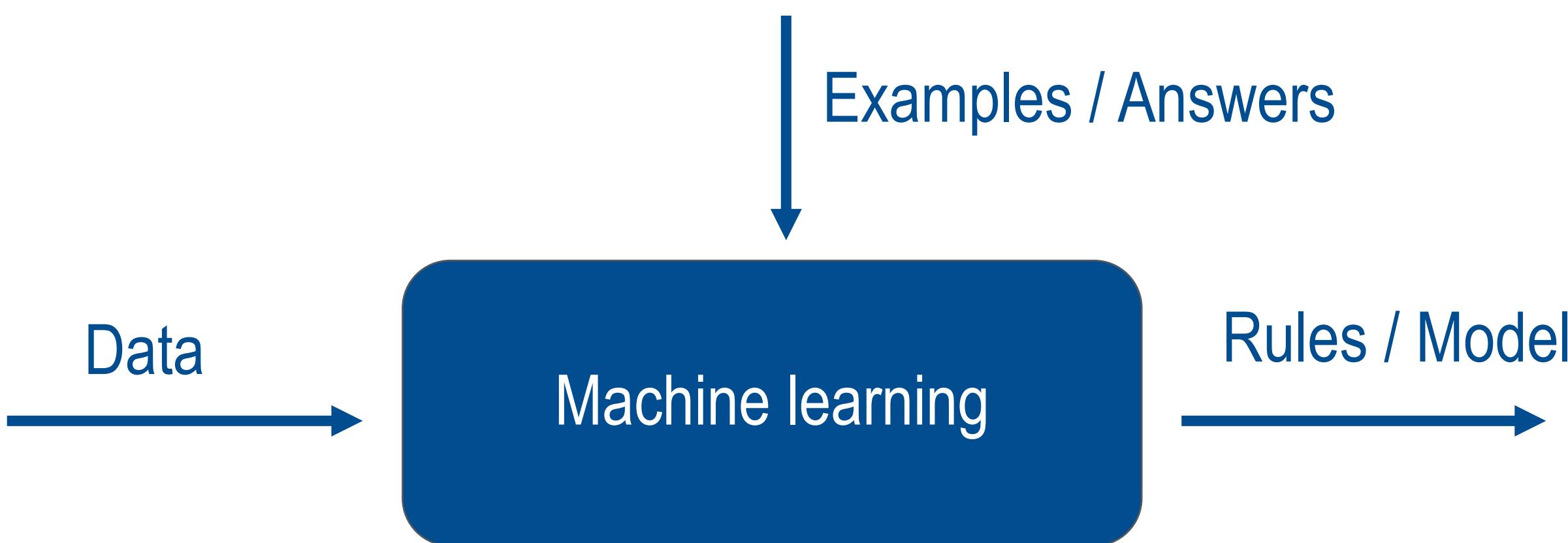


Traditional software: applying given rules to data

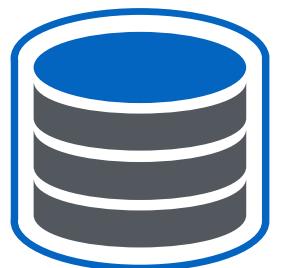


Machine learning –
how is it different?

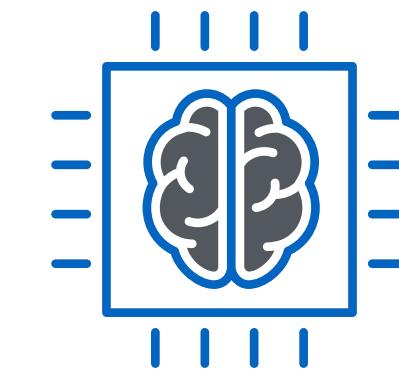
Machine learning: creating rules from data (explicit or implicit)



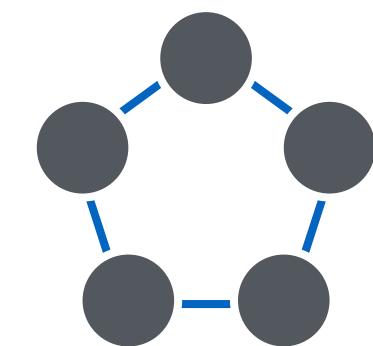
1 Model design, training and testing (model building, feature engineering)



Historical Data



Machine Learning

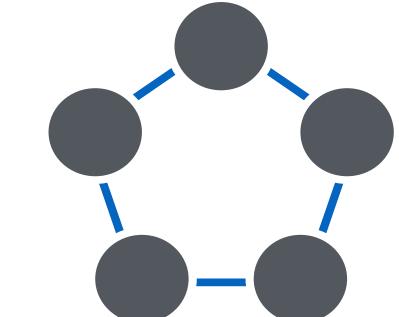


Model

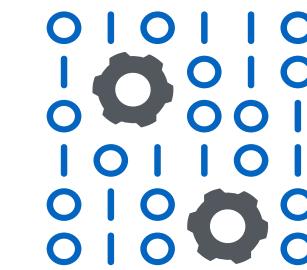
2 Model application (model scoring)



New Data

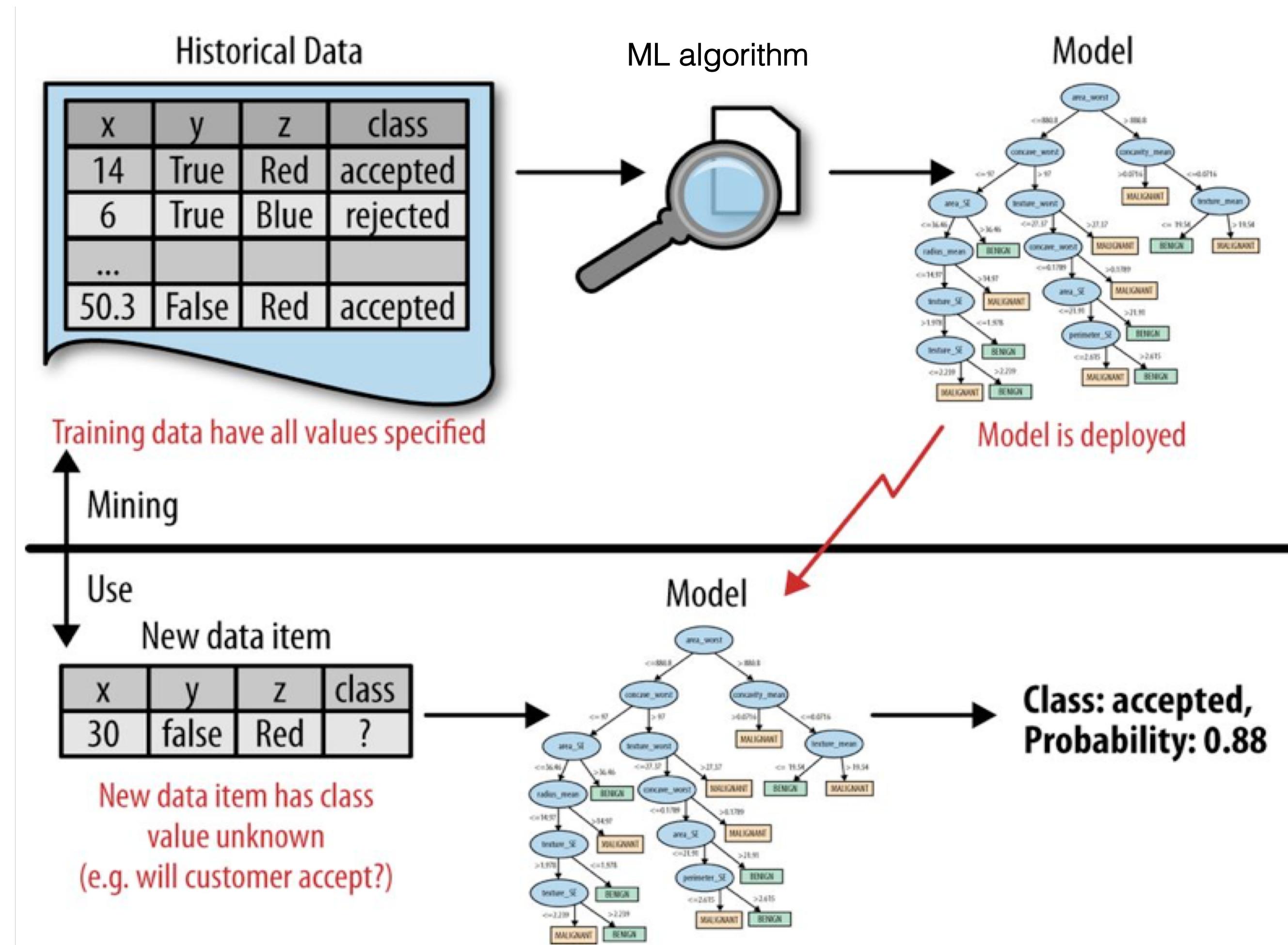


Model

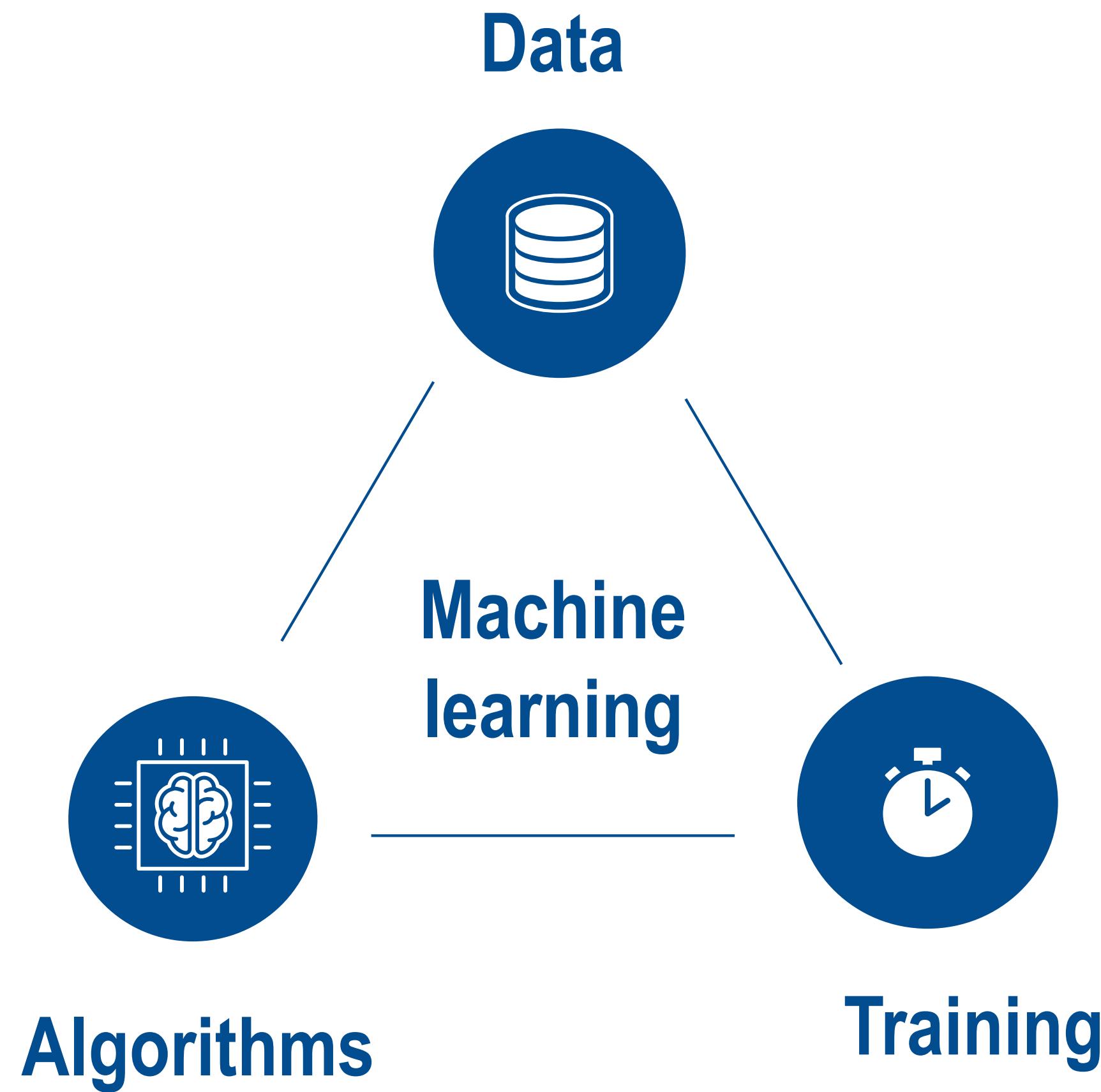


Predictions

MODEL TRAINING AND APPLICATION



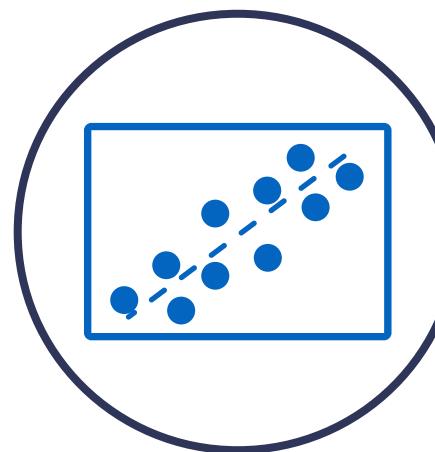
TRIAD OF ALGORITHMS, DATA AND TRAINING



"Learning" is the process of estimating an unknown dependency or structure of a system (building a model) from a limited number of observation (data points) and ability to generalize it onto previously unseen data

Finding patterns in data and generalizing them

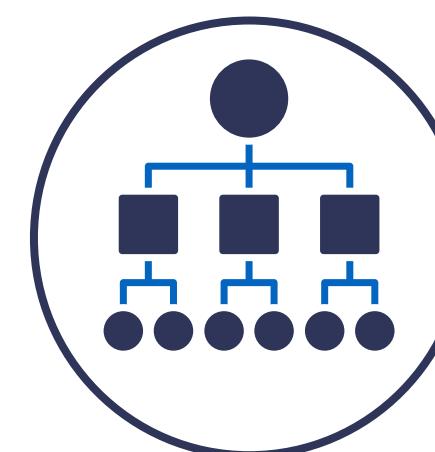
THREE TYPES OF MACHINE LEARNING



Supervised Learning

The goal is to learn mapping from given inputs X to outputs Y, given a labeled set of input-output (X-Y) pairs.

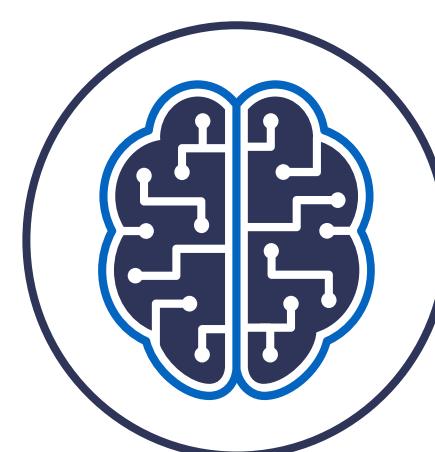
X – input data / independent variable
Y – response/ dependent variable



Unsupervised Learning

The goal is to learn patterns and structure in data given only inputs X. (no output Y information given at all)

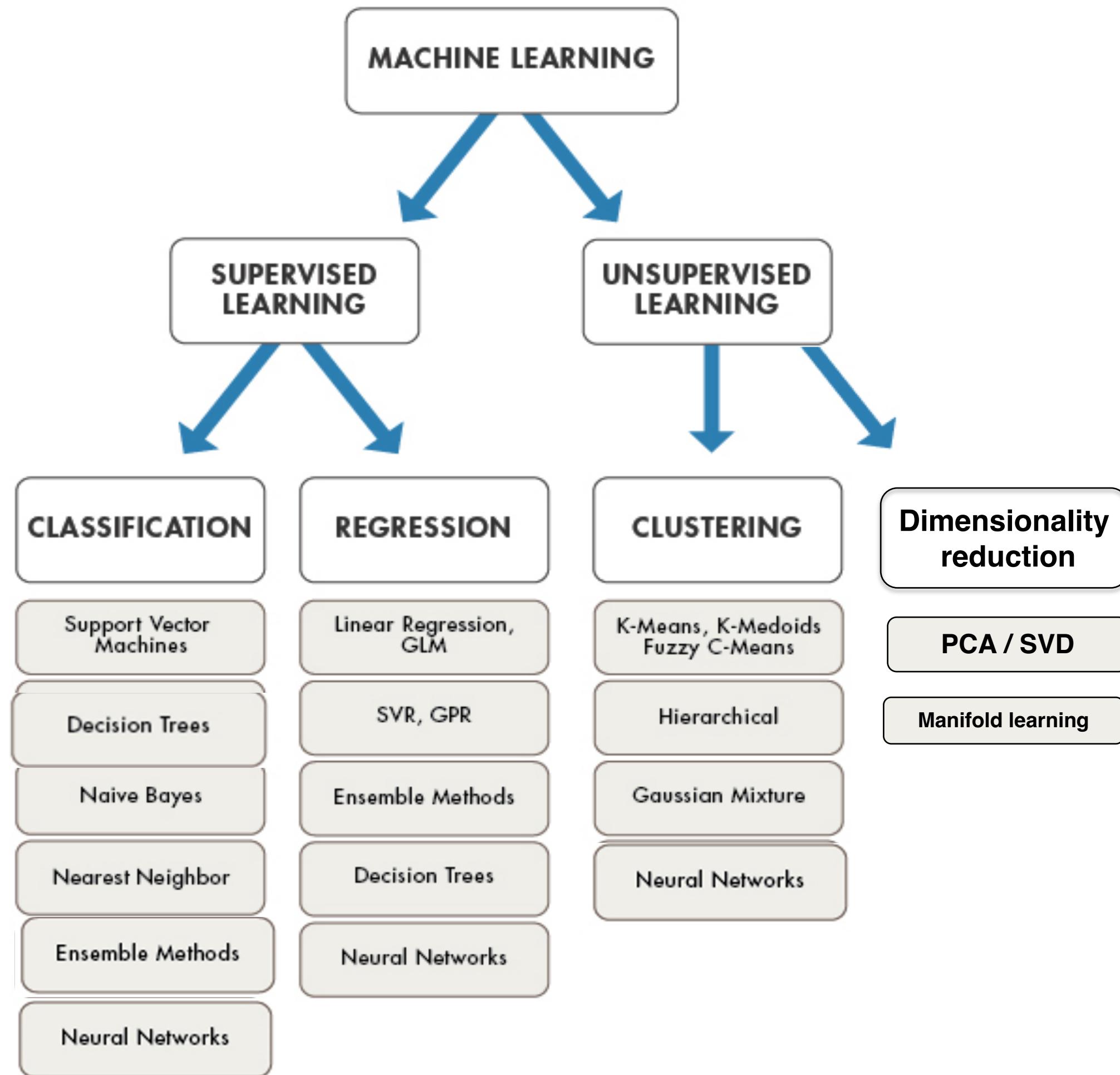
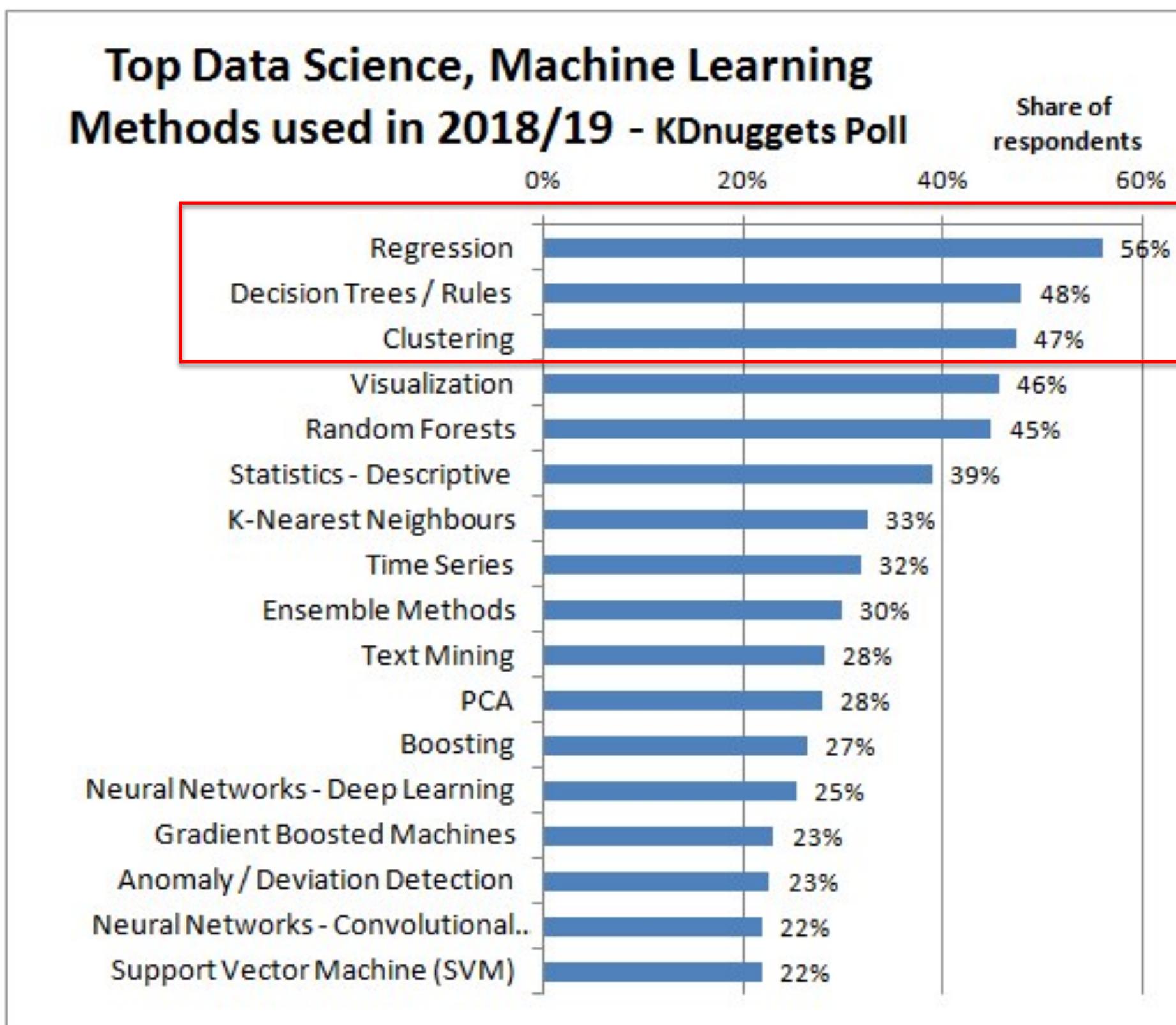
X – input data / independent variable



Reinforcement Learning

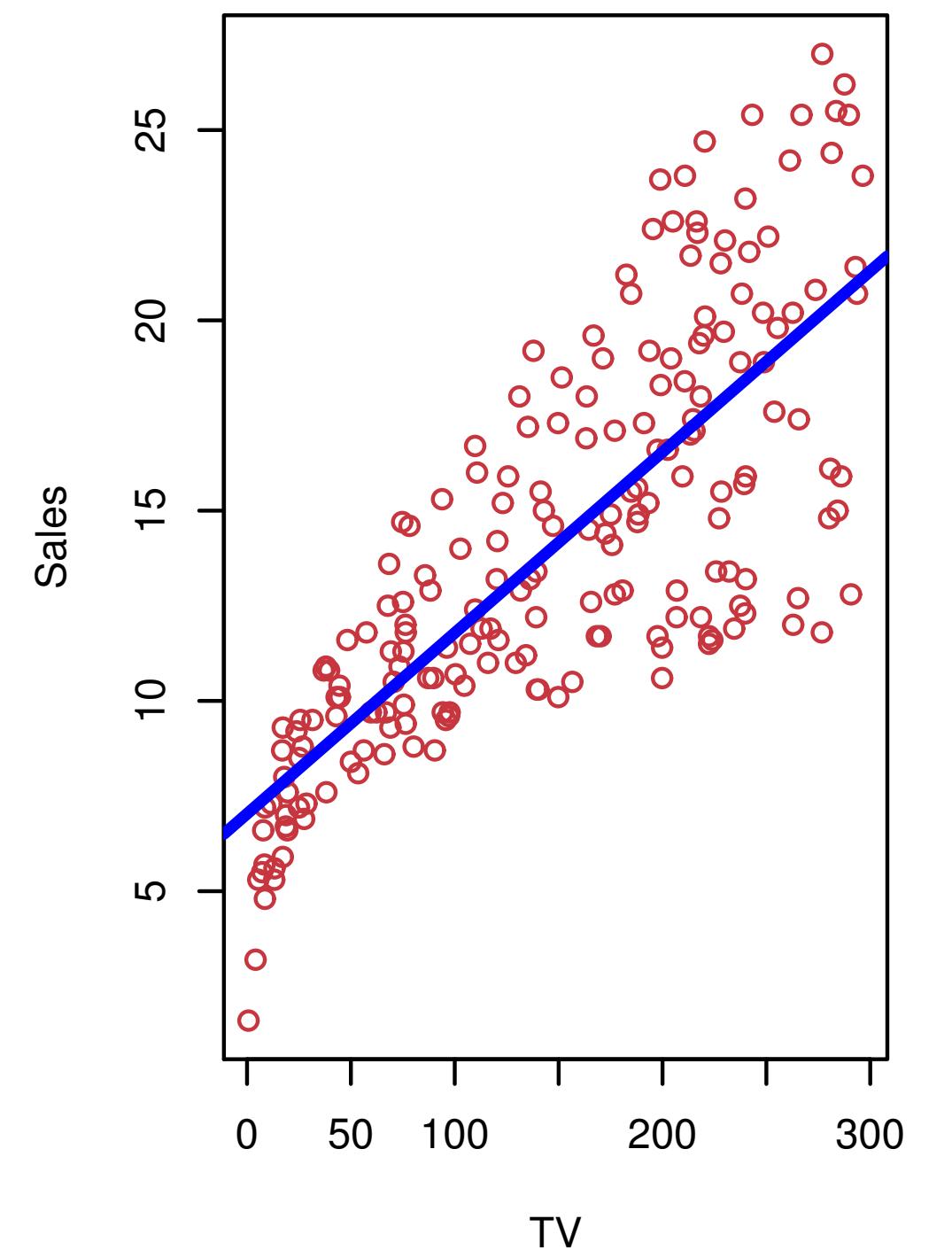
The goal is to optimise actions in a way that maximises cumulative reward. no explicitly labeled data is given, but “rewards” and “punishment” signals are provided

MACHINE LEARNING METHODS



SUPERVISED LEARNING: REGRESSION

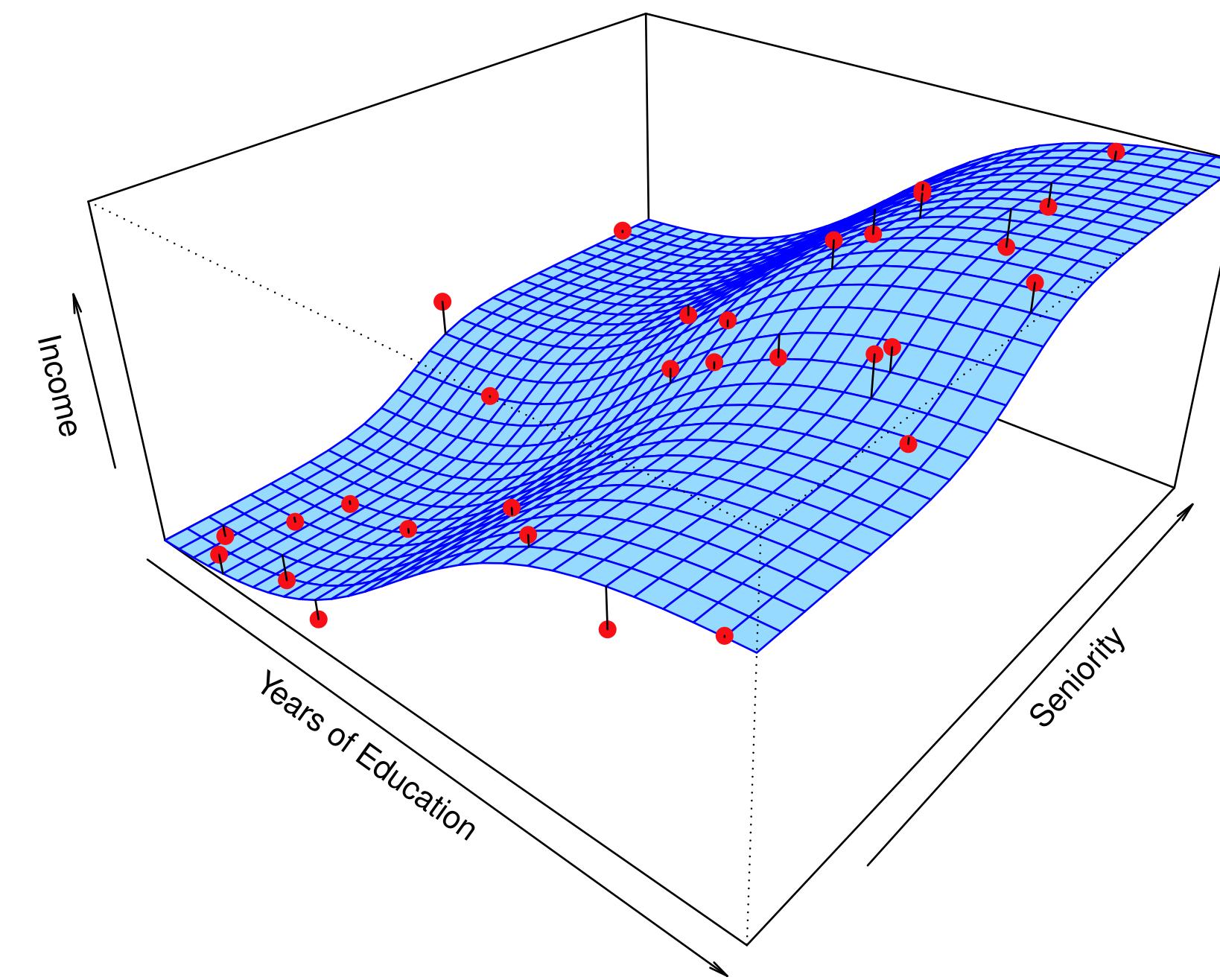
Response variable Y – real valued



univariate

Sales

Sales



multivariate

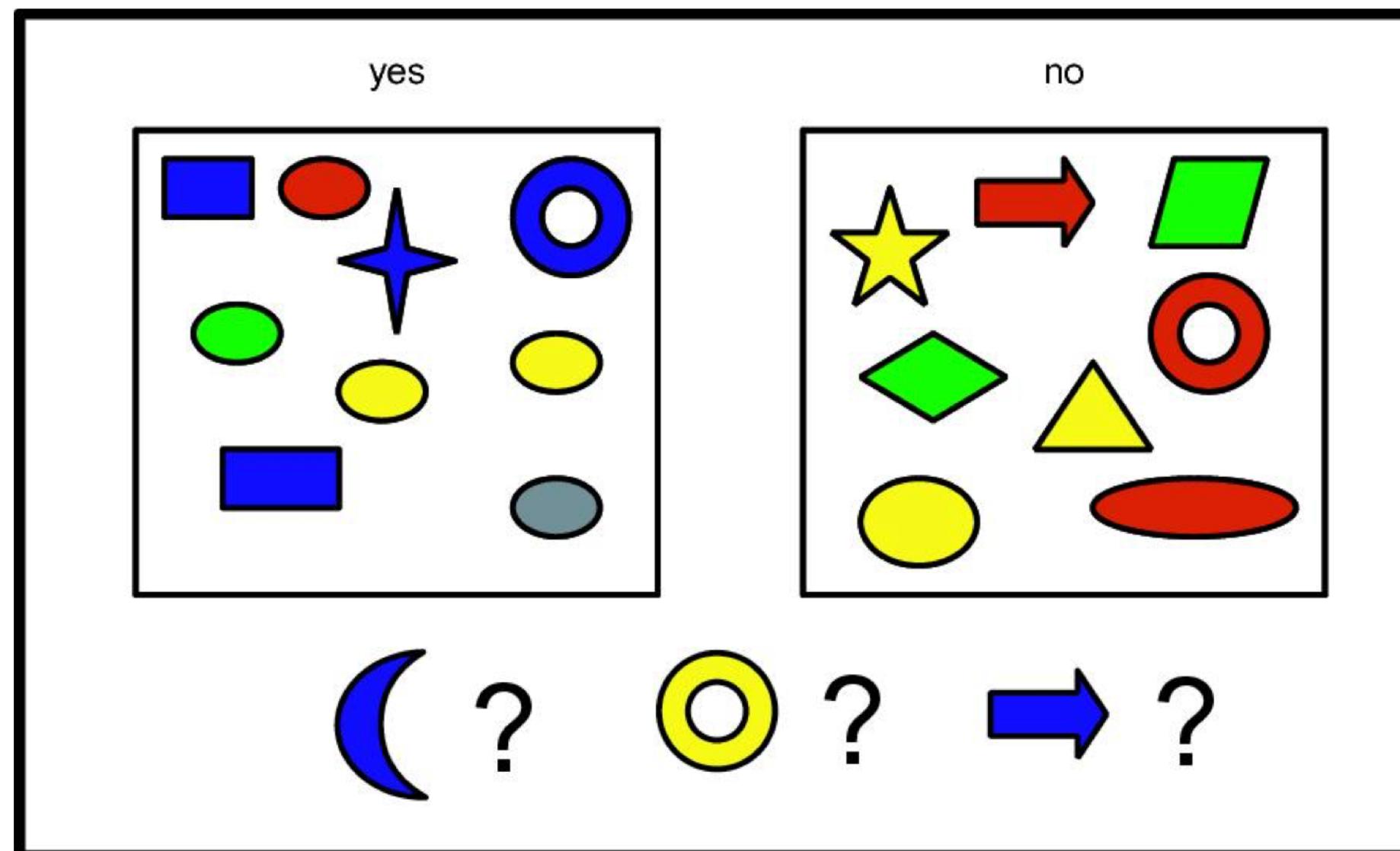
Income

Years of Education

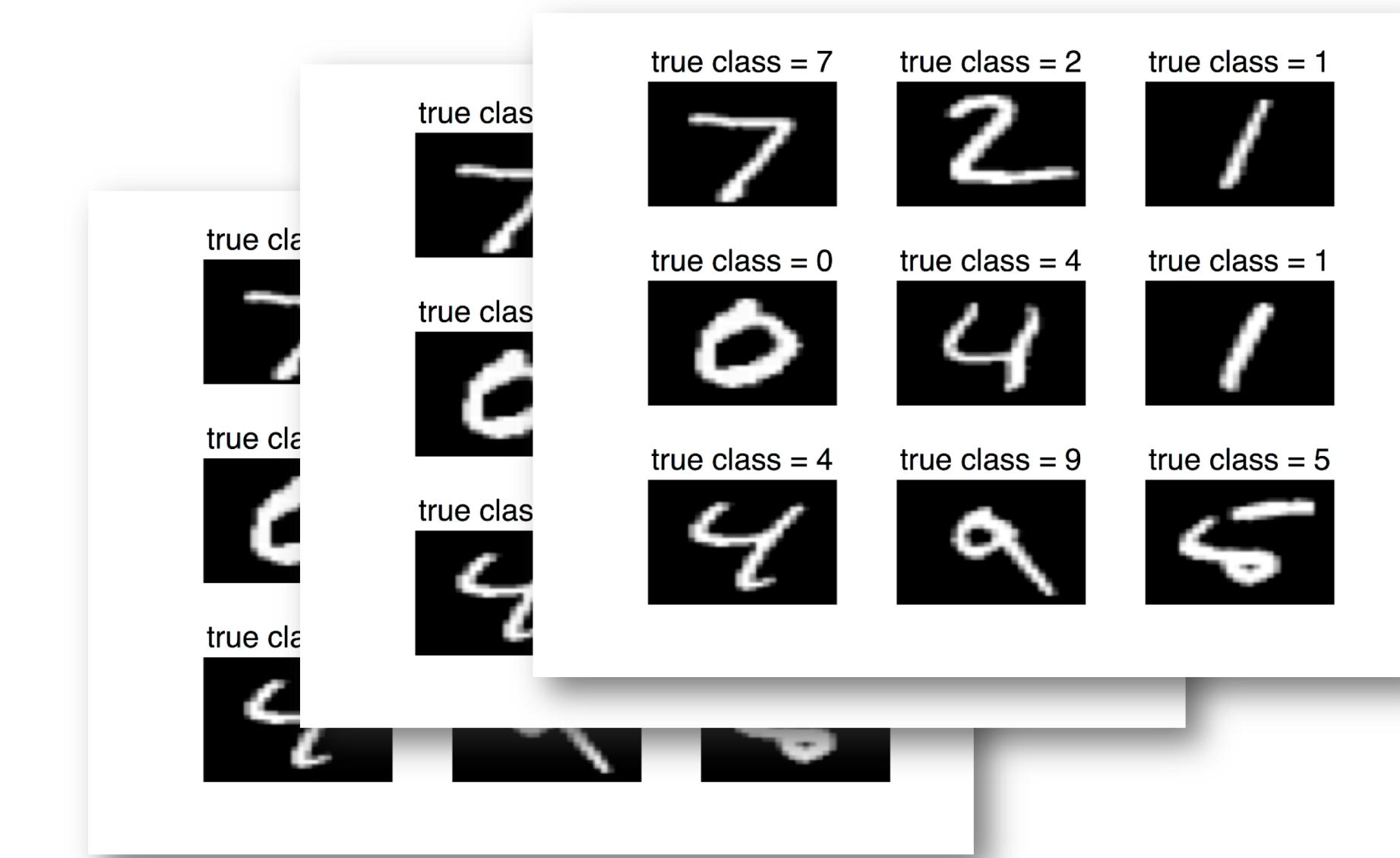
Seniority

SUPERVISED LEARNING: CLASSIFICATION

Response variable Y – categorical



binary

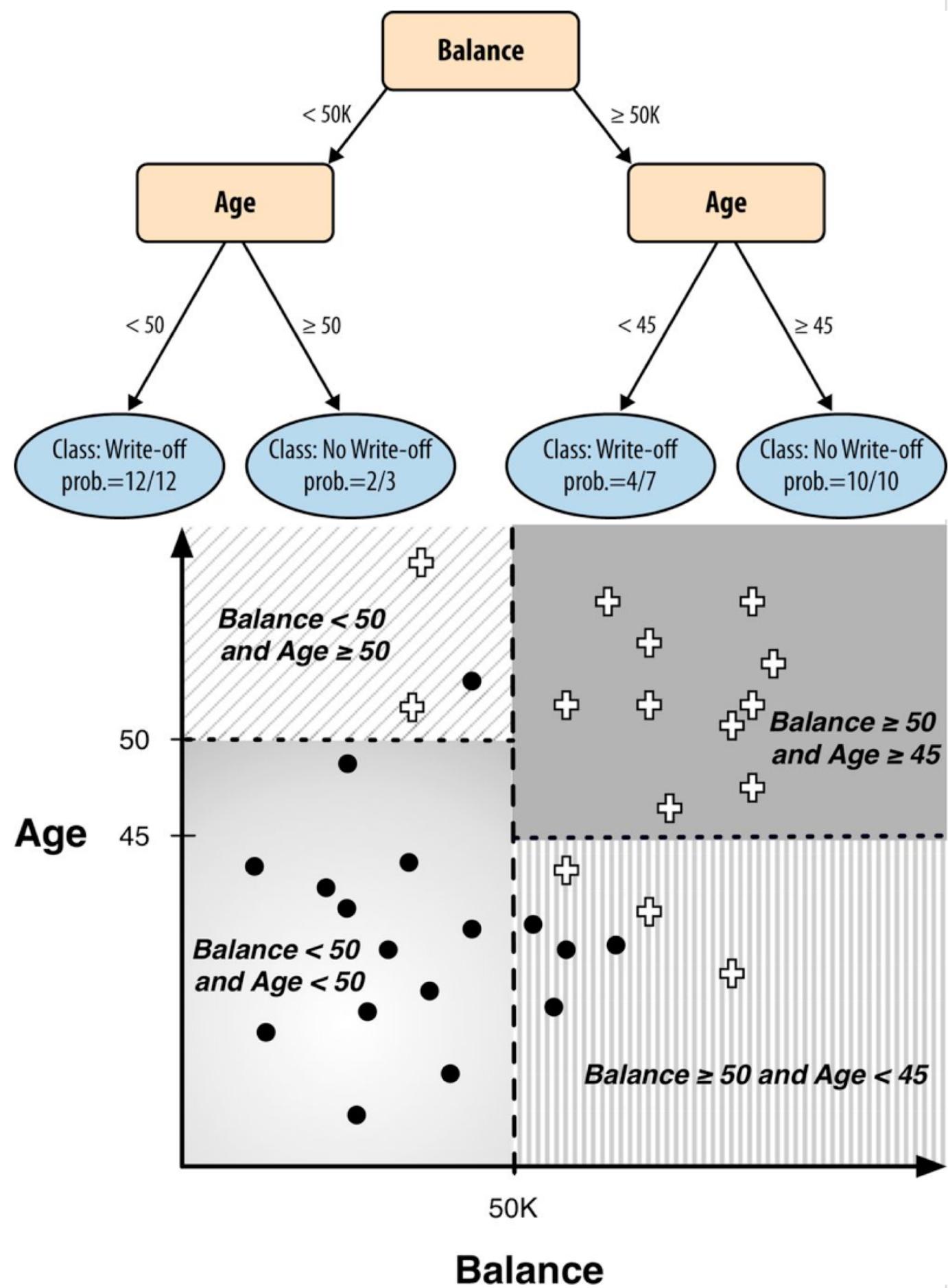


4 1 5 7 1 3 3 6 4 8 1 9 7 6 3 6 9 3 0 6
4 7 7 8 1 3 7 2 4 6 4 3 2 8 6 1 4 3 0 9
1 1 7 6 5 8 6 0 0 3 9 5 4 1 5 7 2 3 2 1
3 5 2 5 7 3 2 9 7 1 6 9 4 6 8 3 2 4 1 9

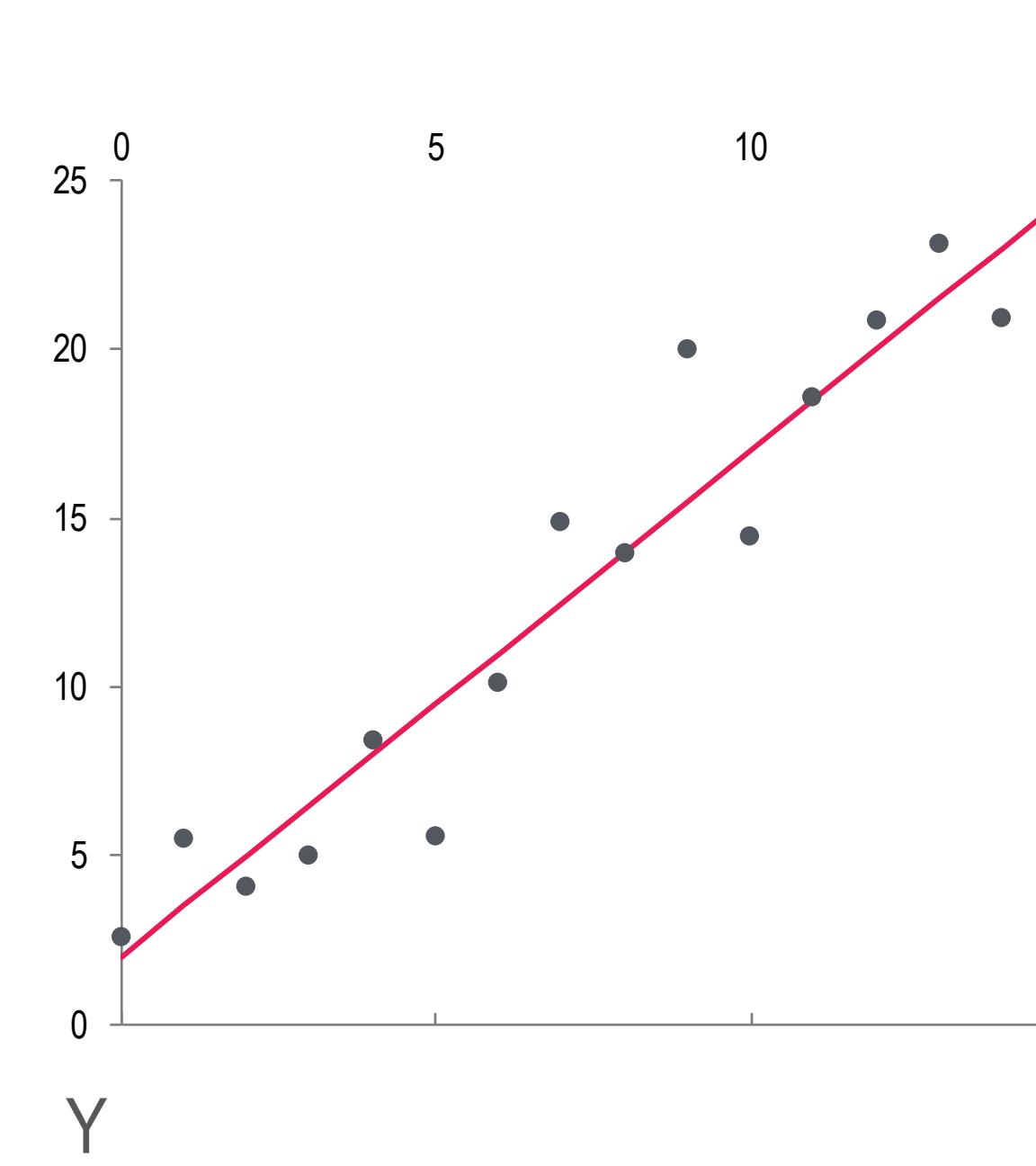
multiclass

SUPERVISED LEARNING: CLASSIFICATION

Response variable Y – categorical

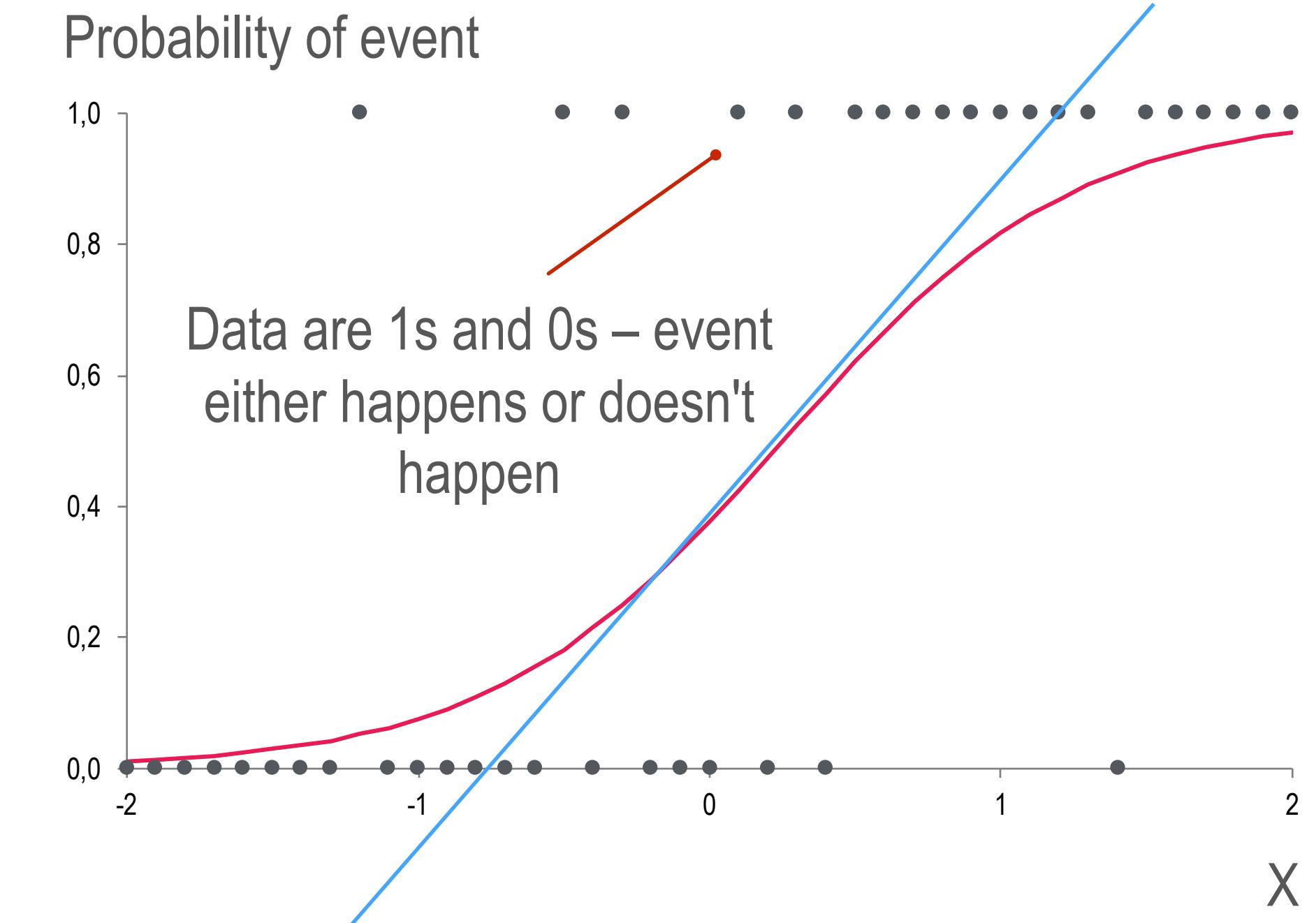


REGRESSION AND CLASSIFICATION



Regression

Predict a numeric variable



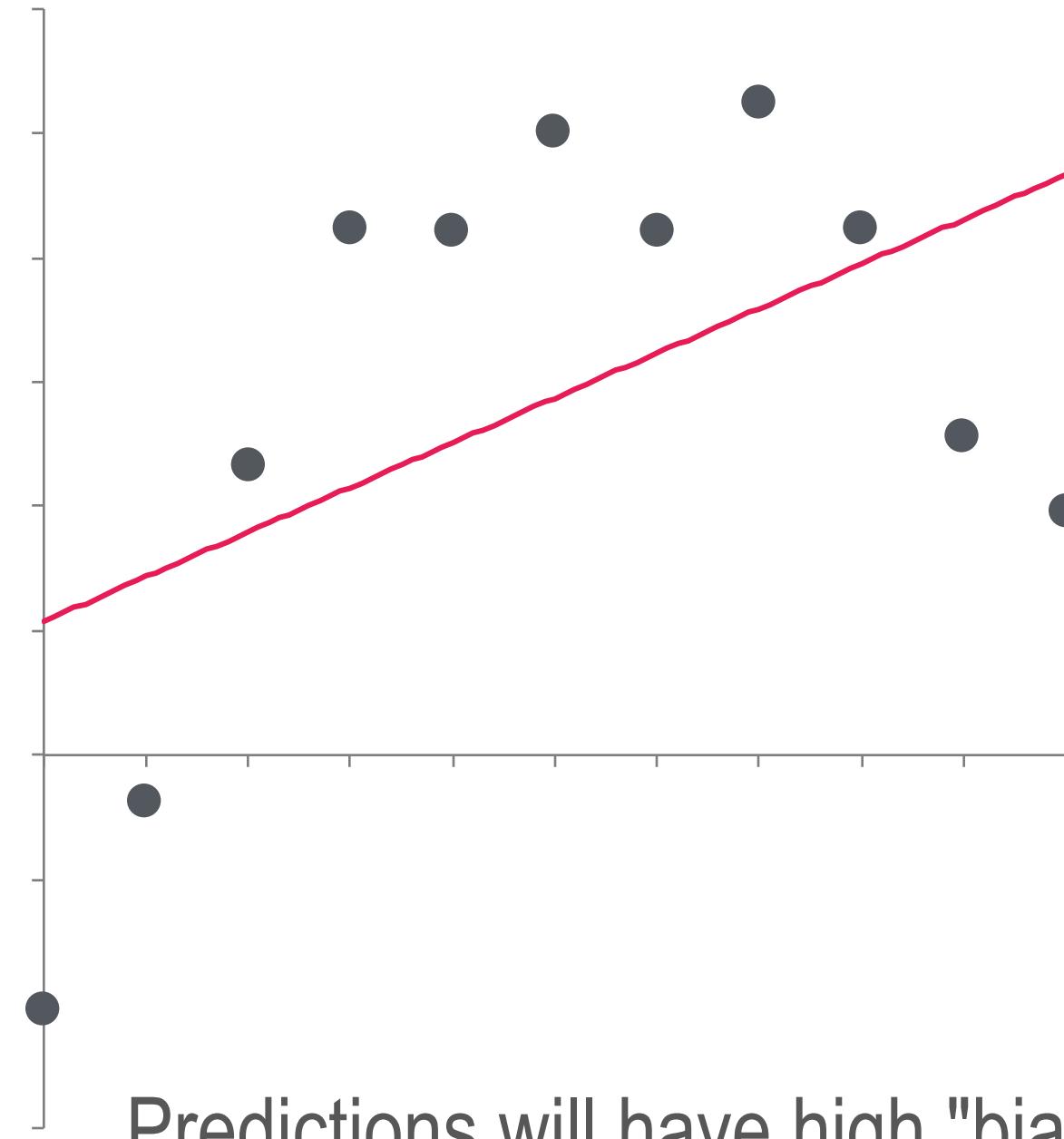
Classification

Predict a binary (or categorical) outcome

MODEL OVERTTING

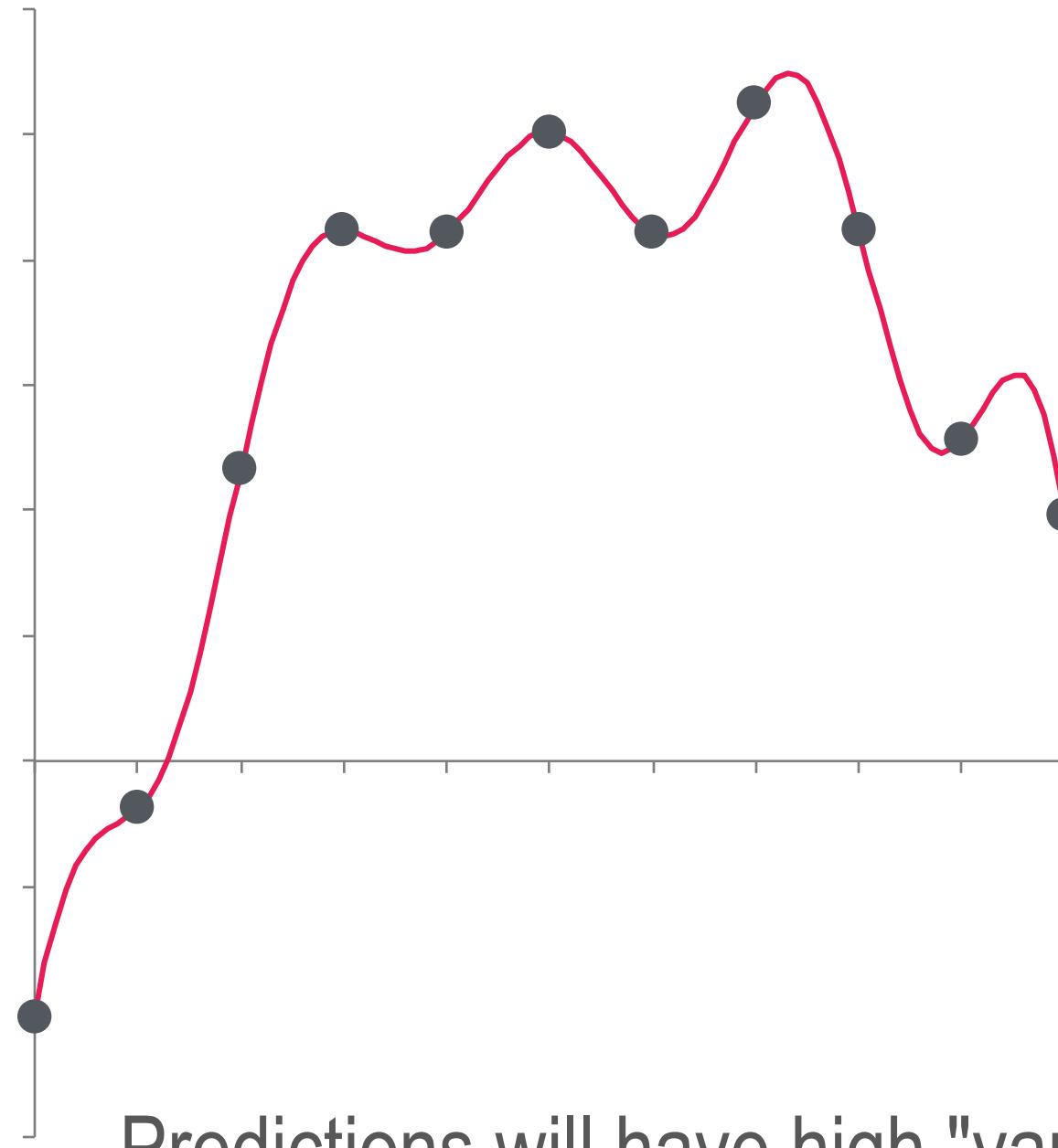
Regression

Too simple



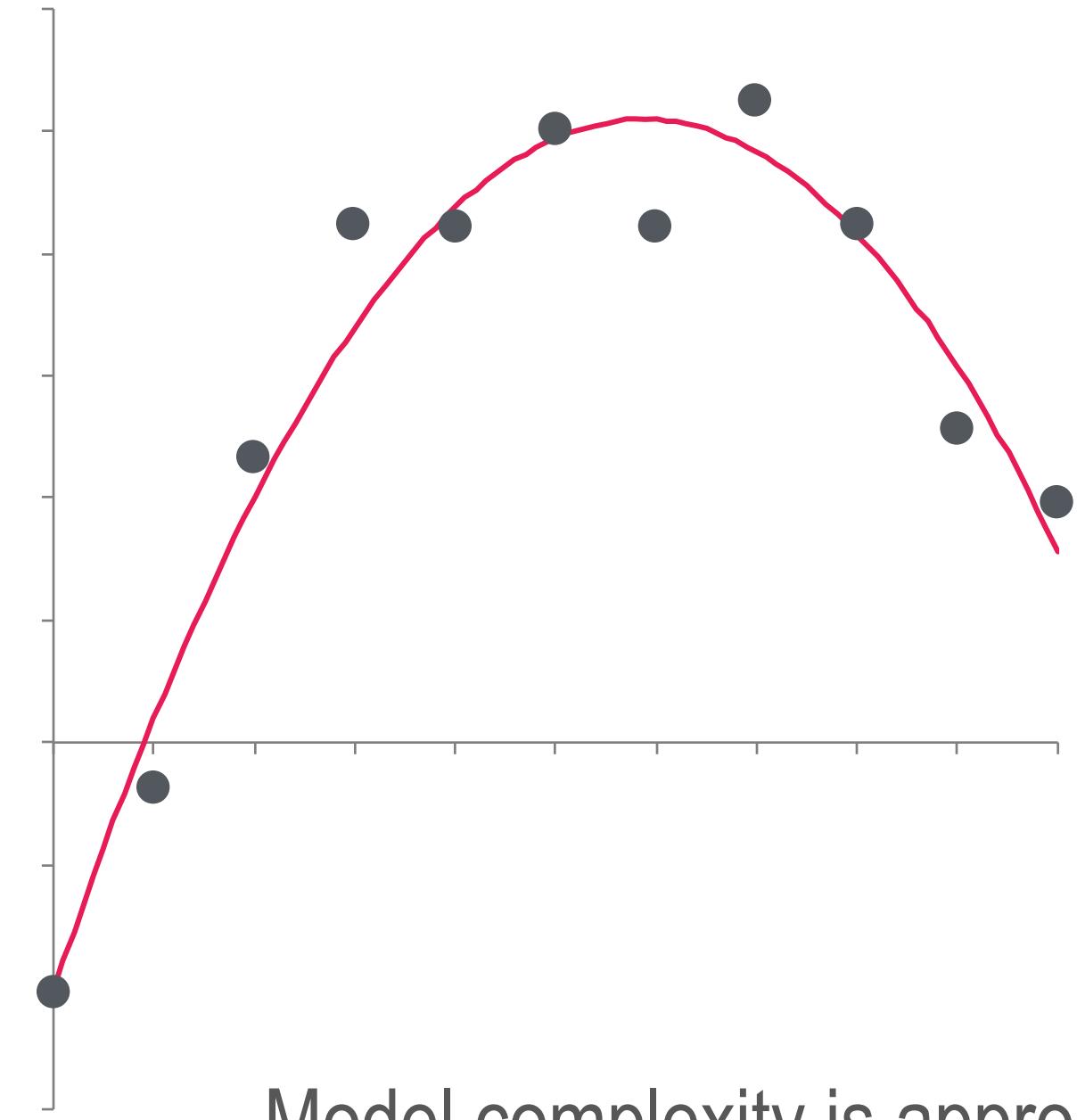
Predictions will have high "bias" –
from inadequate assumptions

Too complex



Predictions will have high "variance"
– driven by noise in the training data

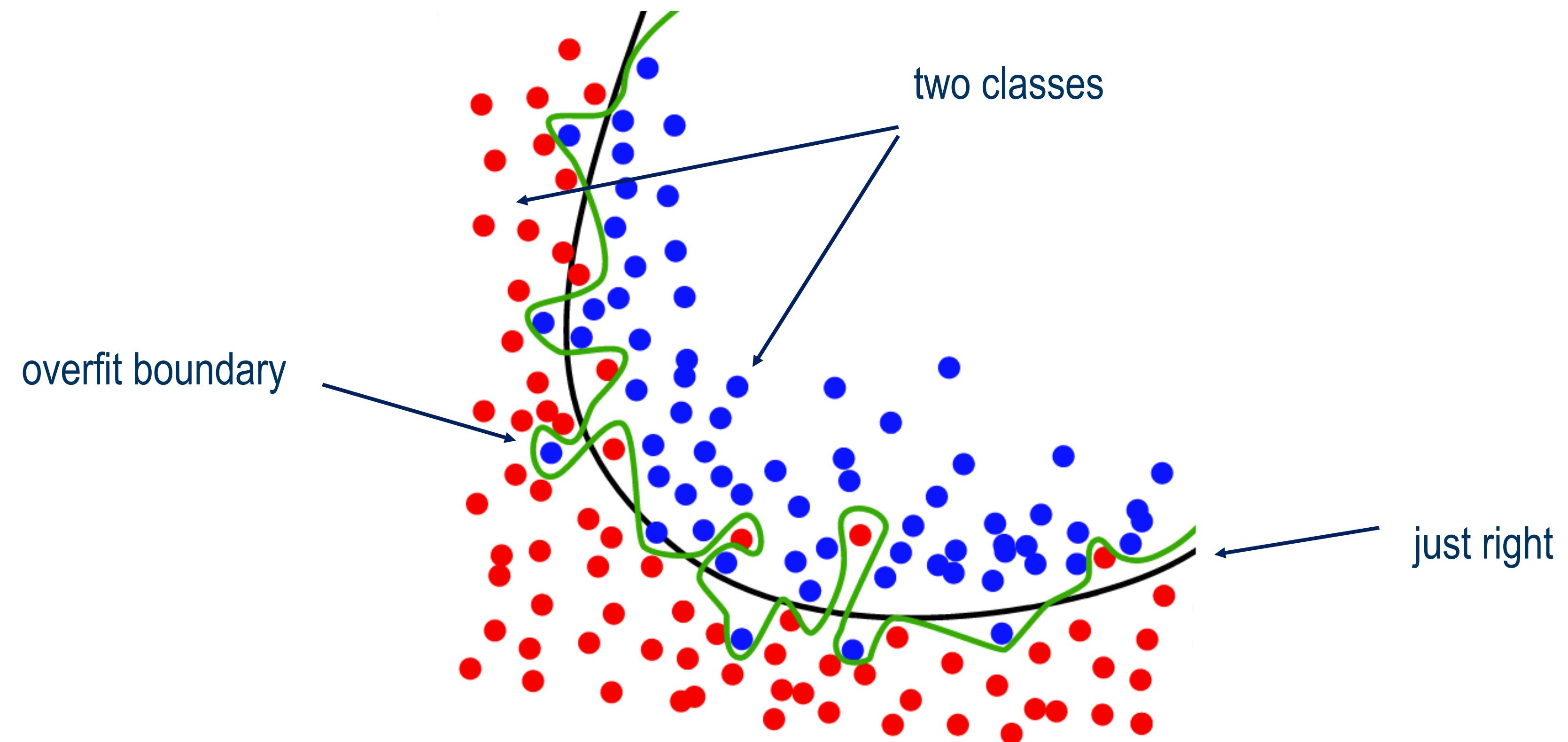
Just right



Model complexity is appropriate
given the noise

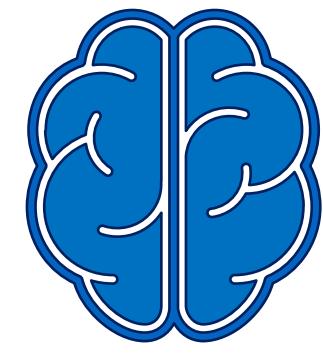
MODEL OVERFITTING

Classification



PREDICTION ACCURACY VS EXPLAINABILITY

Model explainability



White box models

- Interpretable by design
- Easy to explain
- Quick to run
- Limited tuning needed

Prediction accuracy



Black box models

- Lots of work to get insights Better predictive performance
- Potential for overfitting
- Often lot of tuning required

Model properties

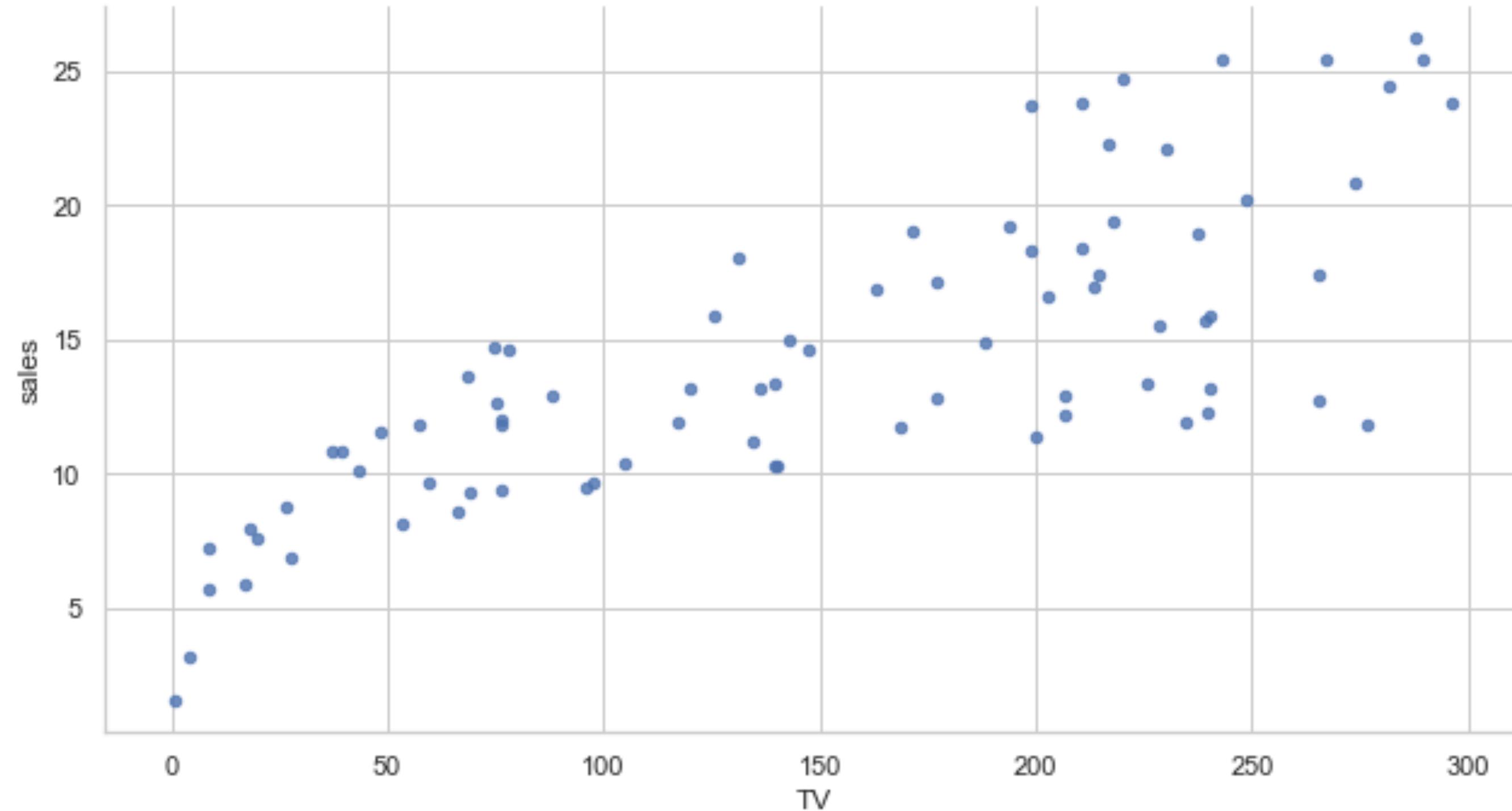
Algorithm examples

- Linear / logistic regression
- Decision trees

- Random forests
- Gradient boosting
- Neural networks
- Deep learning

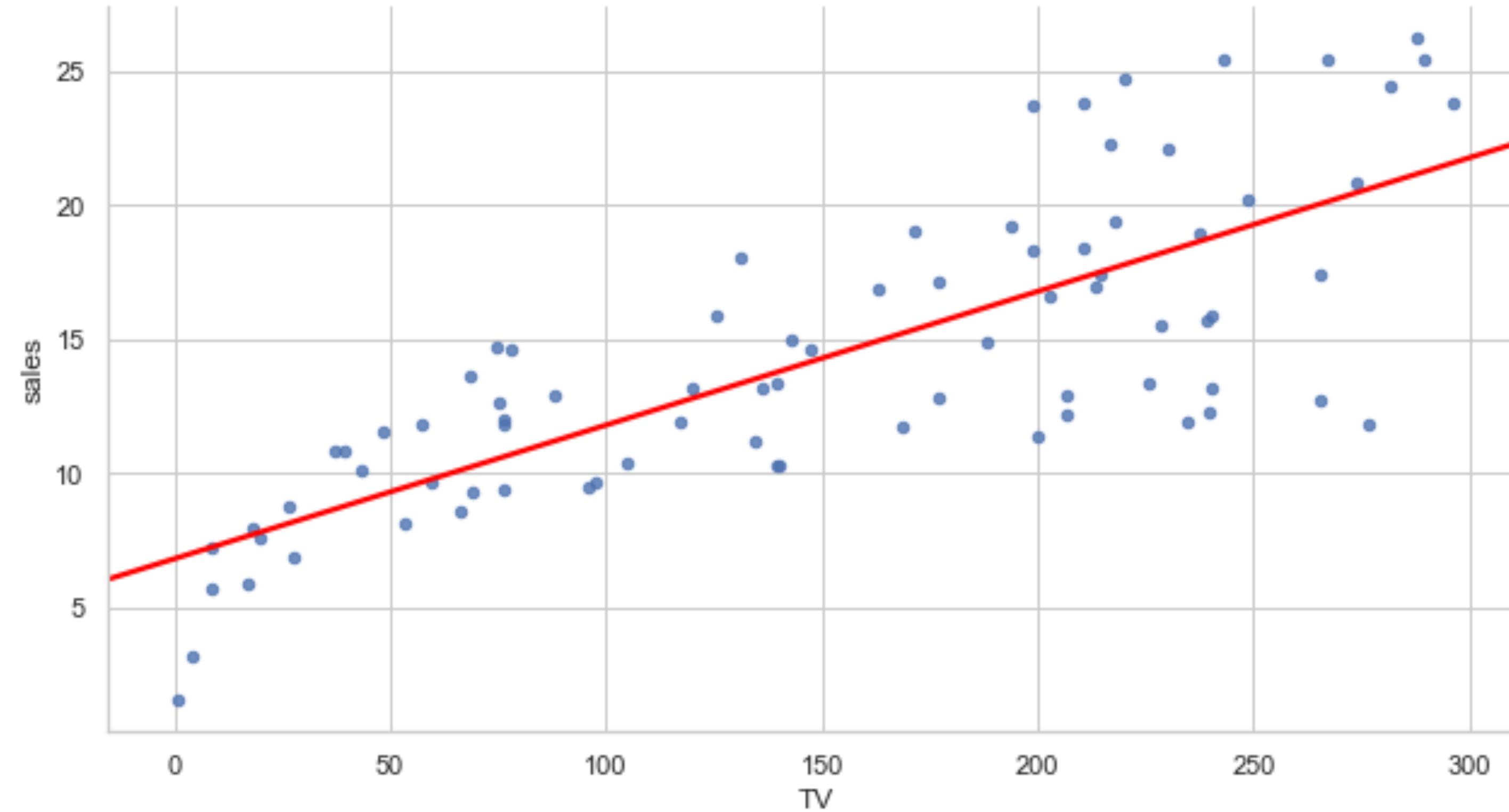
REGRESSION

Modeling



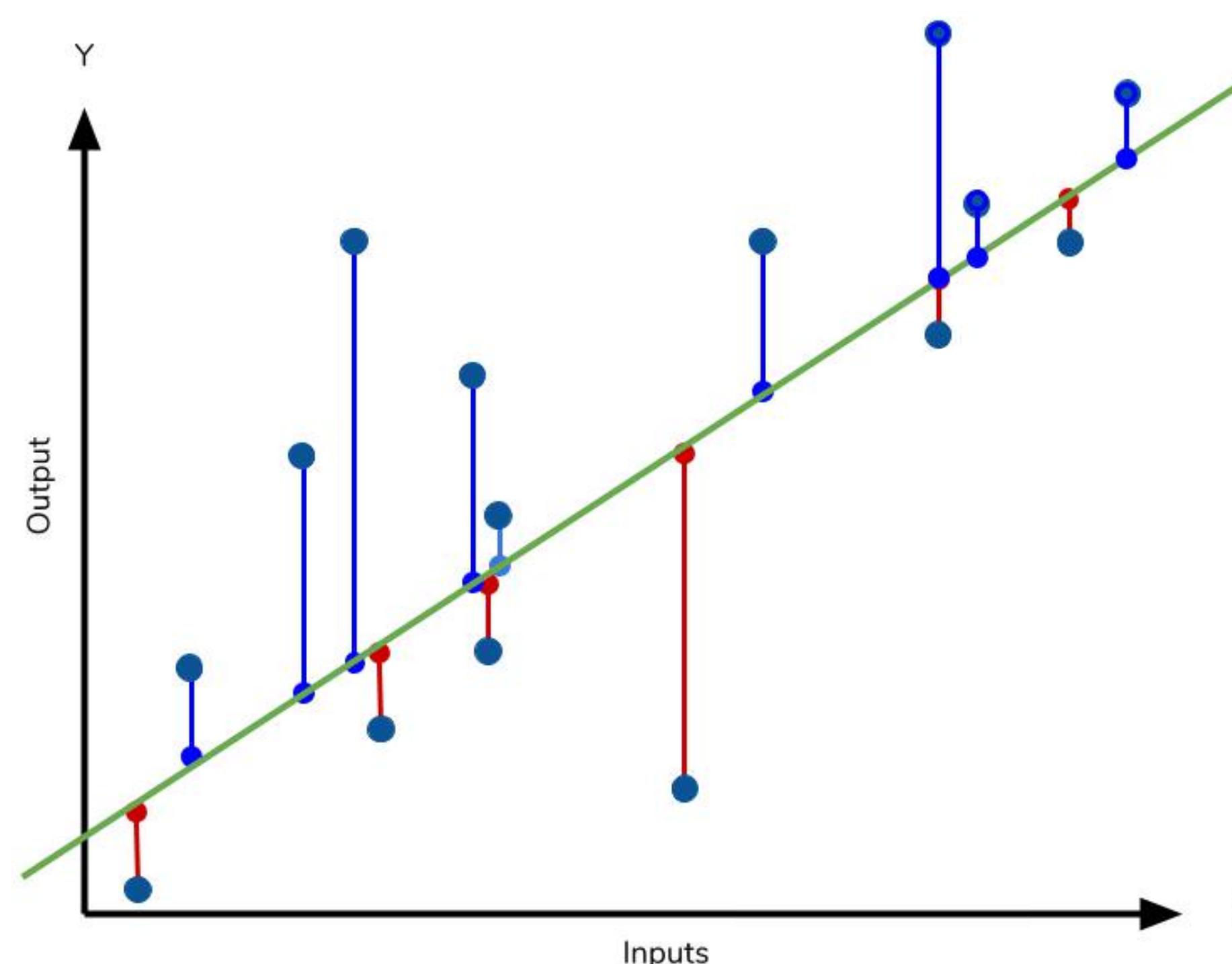
REGRESSION

Quality metrics



REGRESSION EVALUATION

Error metrics/loss function



Standard quality metrics

Mean absolute error:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}|$$

Mean squared error:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2$$

Root mean squared error:

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2}$$

R-squared:

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

residuals
variance

Baseline model (model= mean value), $R^2 = 0$

Perfect model (residual =0), $R^2 = 1$

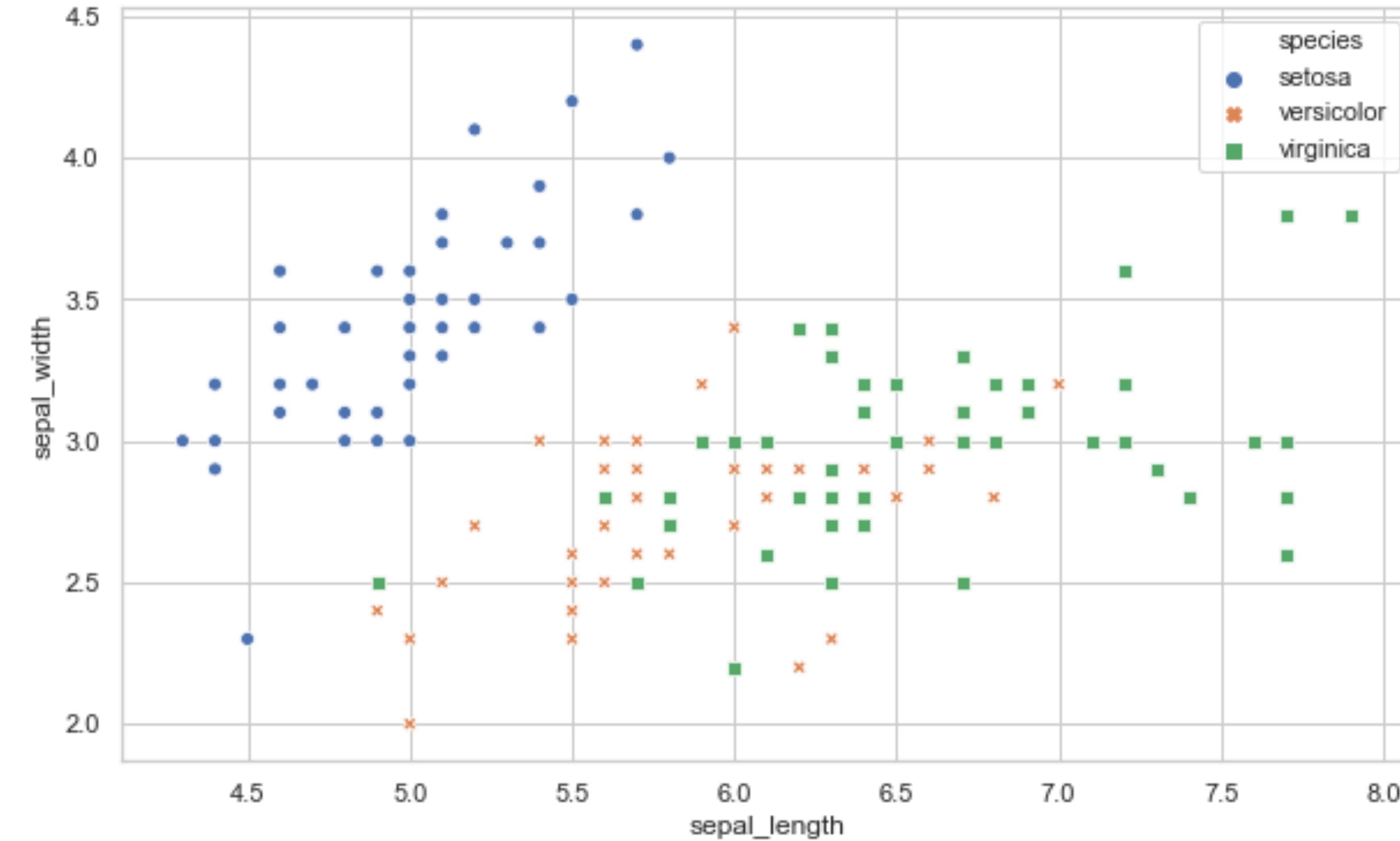
Where,

\hat{y} – predicted value of y

\bar{y} – mean value of y

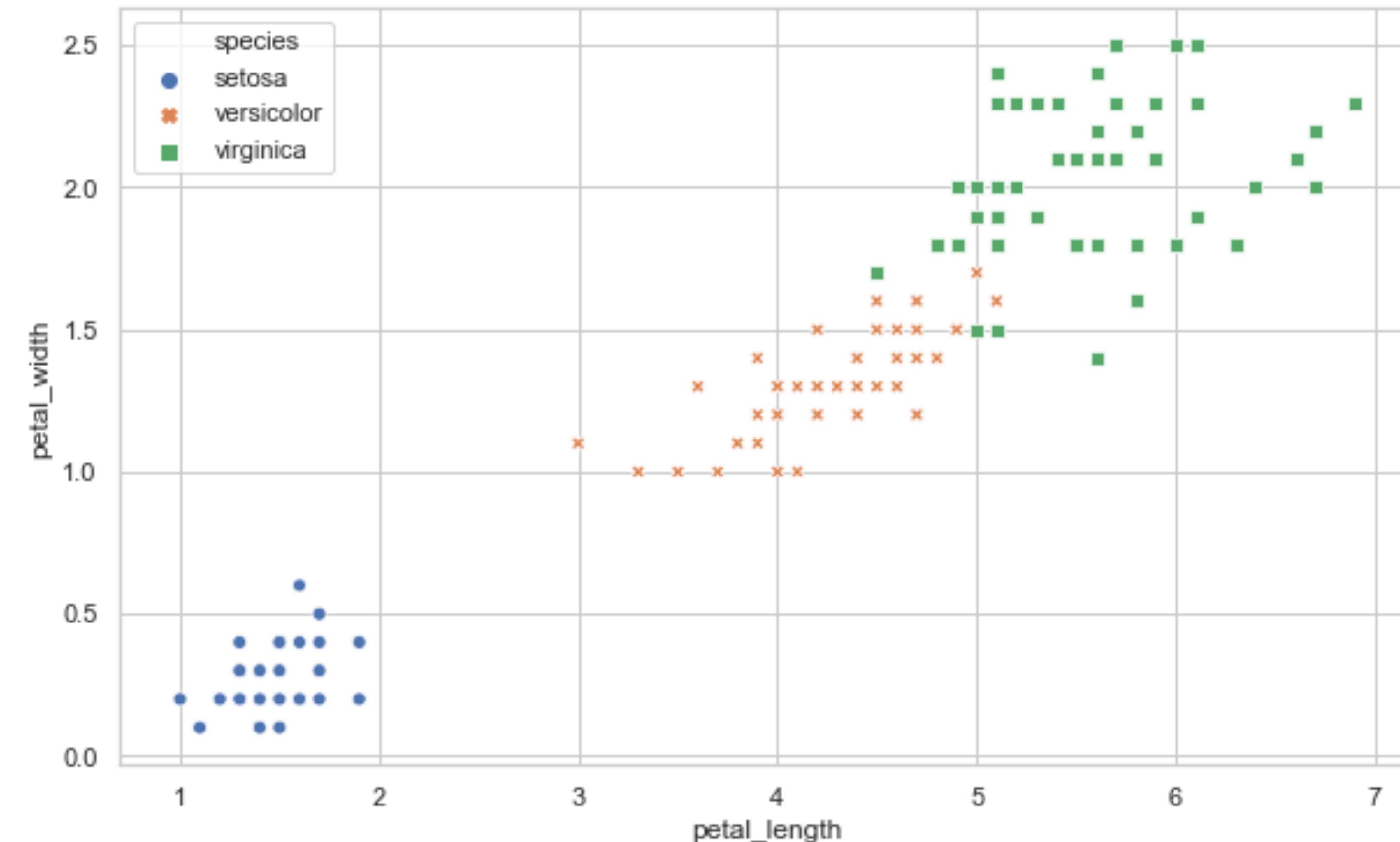
CLASSIFICATION

Classification



CLASSIFICATION

Classification



CLASSIFICATION EVALUATION

Quality metrics

		Actual	
		Yes (or 1)	No (or 0)
Predicted	Yes (or 1)	True positives TP	False Positives FP
	No (or 0)	False Negatives FN	True negatives TN

True positive = Predict event and event happens

True negative = Predict event does not happen, nothing happens

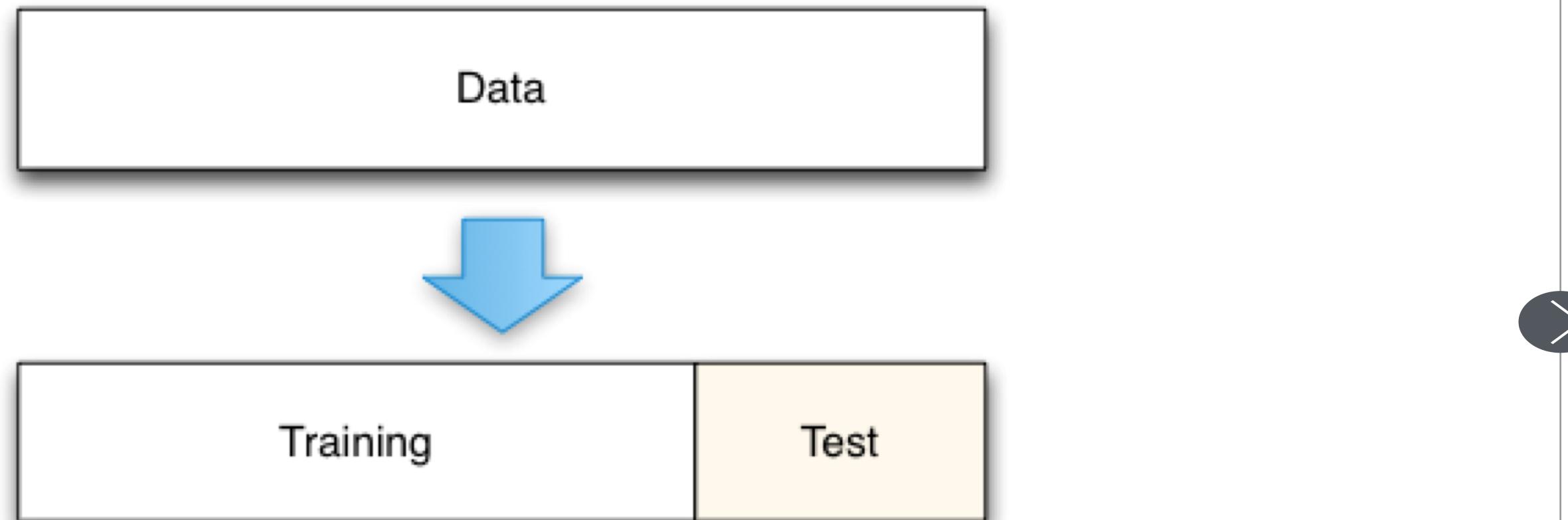
False positive = Predict event and event does not happen (false alarm)

False negative = Fail to predict event that does happen (missed alarm)

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

TRAINING AND TESTING

Train-test split



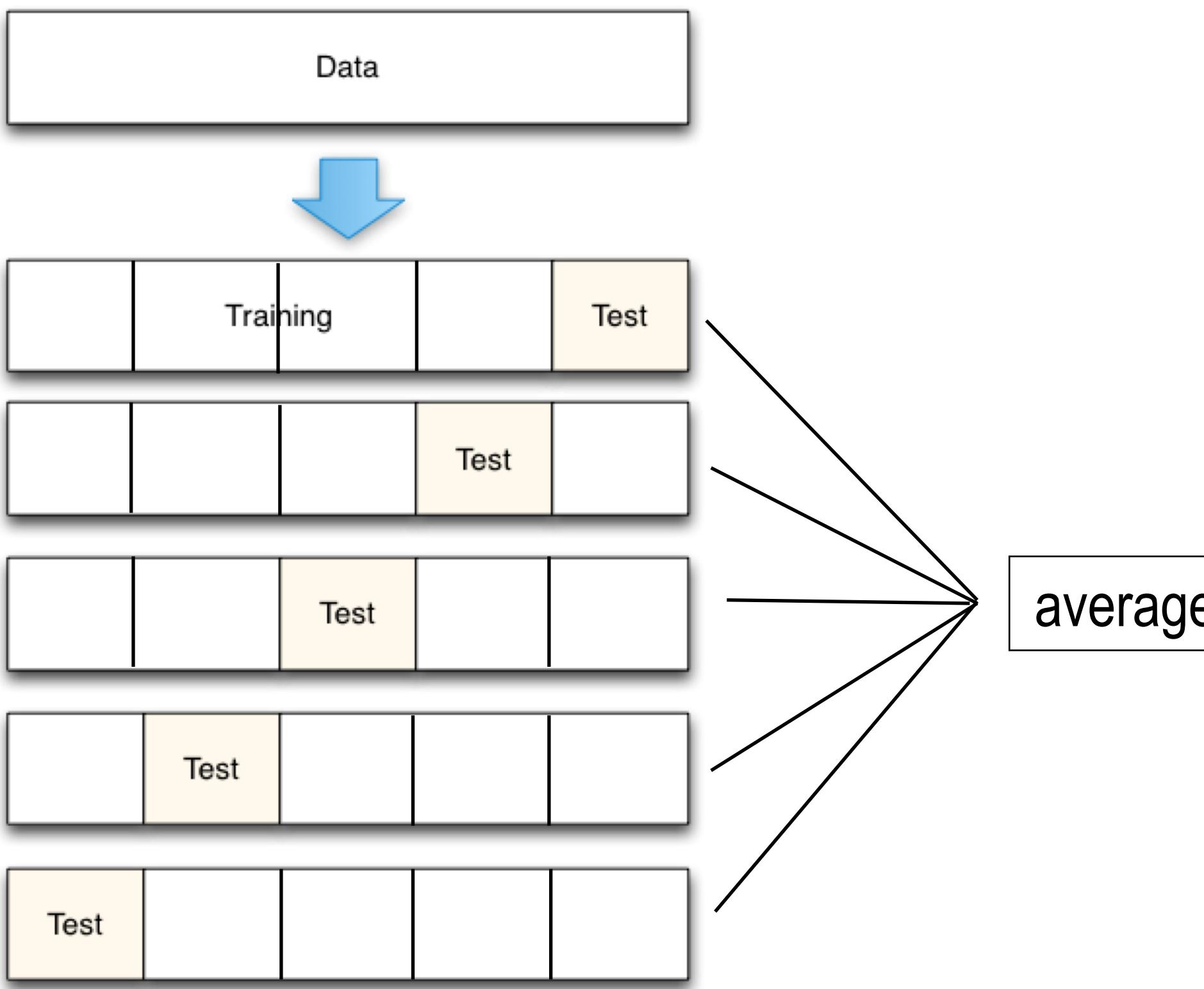
- 70%-90% of the data
- Used to build the model
- 10%-30% of the data
- Used to check the performance of the model on unseen data

Train & Test split

- Measure algorithm performance on both train and test sets!
- Performance will be worse on the test set
- Algorithms hyperparameter tuning can be used to improve test set performance
- Avoid overfitting!
- Actual performance of the algorithm in production will not be better than on test set!

TRAINING AND TESTING

Cross-validation

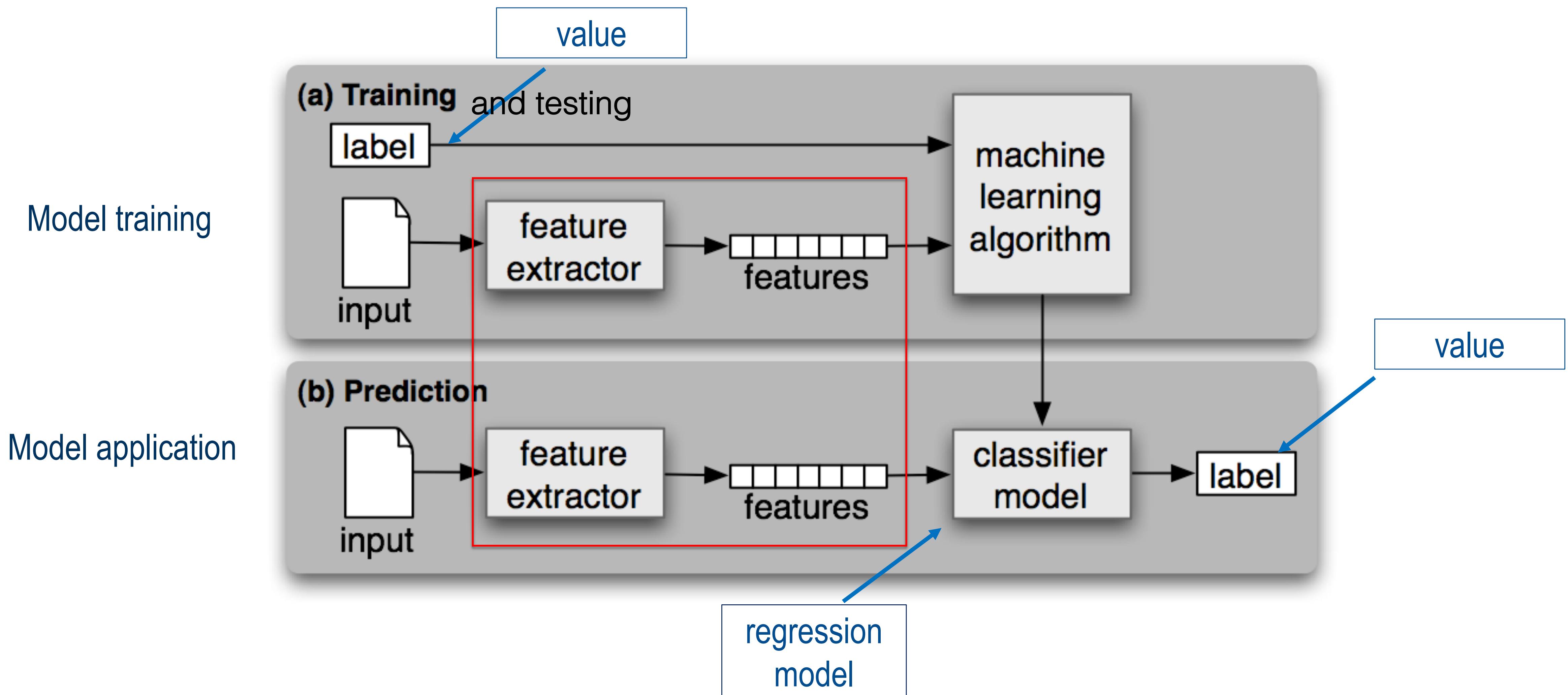


5-fold cross-validation

Cross-validation

- Makes best use of the data
- Data split in to N "folds" at random
- N models built. On each model, N-1 folds are used for training and one is used for testing
- Evaluation criteria averaged across folds
- Allows use of eg 90% training data / 10% test data splits for 10-fold cross validation
- More data for training increases predictive power
- Reduces the chance of getting lucky/unlucky just due to the way a single train/test split is done
- More time/computer resources consuming

TYPICAL SUPERVISED LEARNING PIPELINE

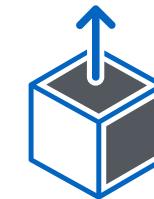


A SUPERVISED MACHINE LEARNING WORKFLOW

Prepare data



Define problem and potential solution



Get the data



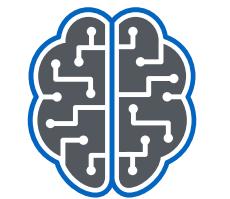
Understand the data



Clean the data



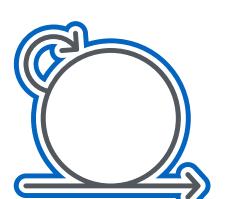
Feature engineering



Build and test model



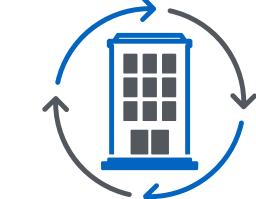
Understand the model



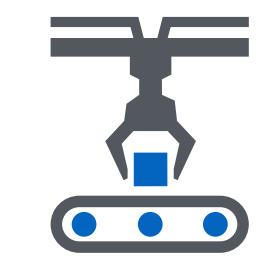
Iterate



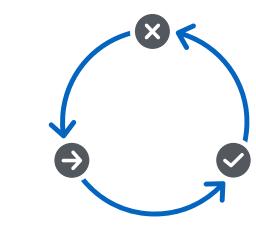
What does it mean for the business?



What are we going to change?



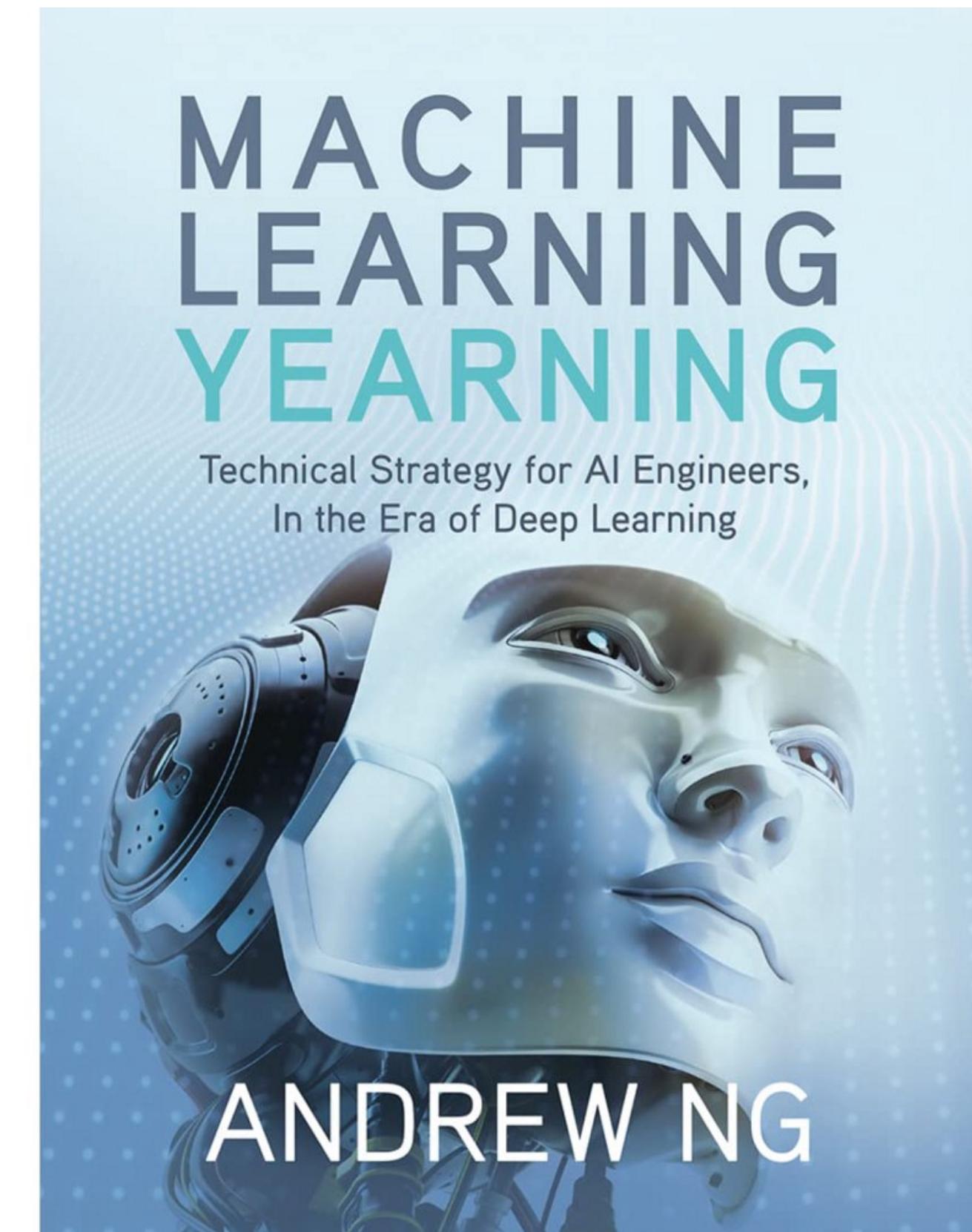
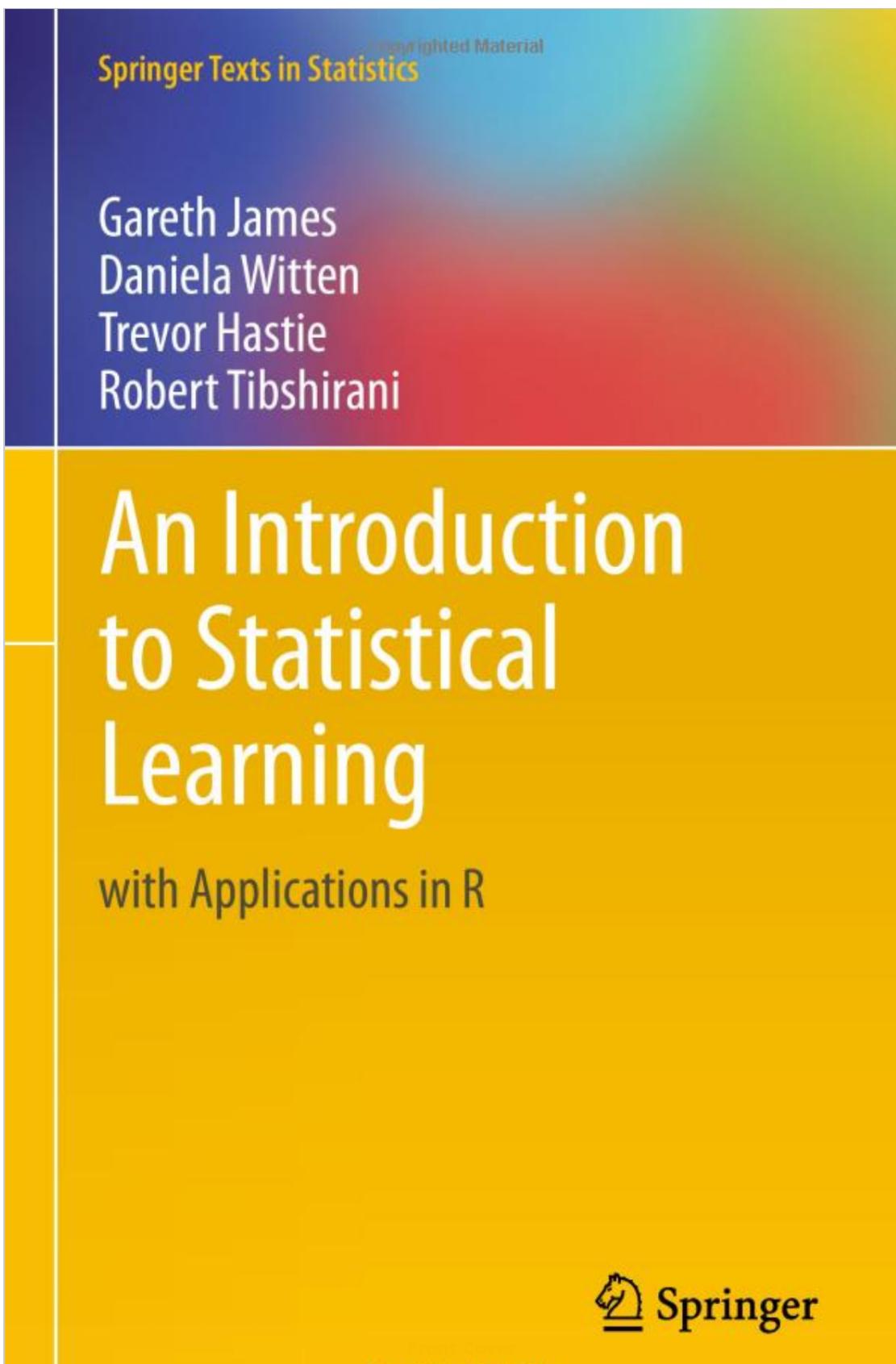
Productionise



Ongoing monitoring and improvements



A FEW MORE BOOKS





NATIONAL RESEARCH
UNIVERSITY