Introduction to machine learning & K-Nearest Neighbors (KNN)

Wenqing Huangfu
UMD Methods Workshop
2023 Fall

- No prior knowledge in addition to basic understanding of R
- Applied machine learning instead of improve models
- No math

- 1. Basic concepts in machine learning
- 2. KNN
- 3. A real example

What is machine learning?

- Machines learn from data
- identify patterns
- and make decisions

Machine learning VS Social Science

Machine learning		Social Science
$\hat{\mathbf{y}}$	~~vs.~~ **and**	$\hat{\beta}$
Predict		Explain

Important to view **prediction** and **explanation** as **compliments**, not substitutes

Source: Mullainathan & Spiess, JEP 2017

Types of machine learning

- **Supervised Learning:**
  - You provide the machine with data that has both the questions and the answers.
  - It learns the relationship between them, so it can give answers to new questions.
    - Examples:

      Regression based; K-Nearest Neighbors (KNN)

      Decision Trees/Random Forest

      Support Vector Machine; Convolutional Neural Networks (CNNs)
- **Unsupervised Learning:**
  - You only give the machine data without specific answers.
  - The machine tries to figure out patterns or groupings on its own.
    - Examples:

      K-means Clustering

      Principal Component Analysis (PCA)

Types of machine learning

- **Active learning:**
  - You provide the machine with a ***small amount*** of data that has both the questions and the answers.
  - The machine can ask for help when unsure.
- **Reinforcement Learning:**
  - The machine interacts with an environment and learns by trial and error.
  - It gets rewards for good actions and penalties for bad ones, guiding it to improve.

Types of machine learning

- **Parametric Models:**
  - Parametric models make assumptions or simplify the data's underlying structure.
  - They have a fixed number of parameters, which are adjusted during the training process.
    - Examples:

      Linear Regression/Logistic Regression

      Neural networks with a fixed architecture
- **Non-parametric Models:**
  - Non-parametric models don't make strong assumptions about the data's underlying structure.
  - They have a flexible number of parameters, which can grow with the training data.
    - Examples:

      K-Nearest Neighbors (KNN)

      Decision Trees (though some might argue they're semi-parametric)

      Support Vector Machine.

Terms of machine learning

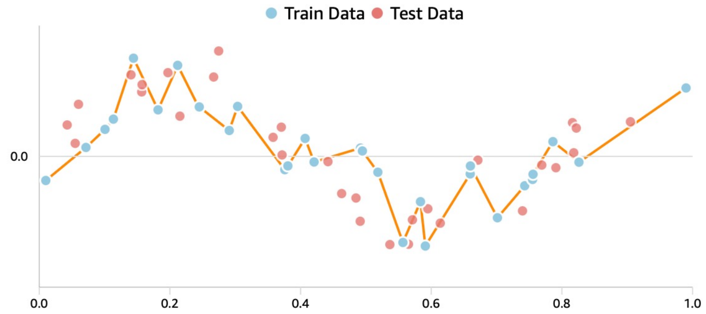
- 1. **Training Data**:
  - The data used to train a machine learning model.
- 2. **Testing Data**:
  - Data used to evaluate how well a trained model will perform on remaining examples.
- 3. **Training error**:
  - compares y to g(x)
  - Tends to be optimistic because g(x) was learned from the same
  - data used to calculate the error
- 4. **Test error**:
  - compares y to g(x) using the test set
  - Better estimates how g(x) generalizes to new data; error is calculated using data not used to learn g(x)

Terms of machine learning

- 5. **Features**:
  - Variables.
- 6. **Label**:
  - Y. The "answer" or "result" for a particular data point in supervised learning.

Terms of machine learning

- 7. **Overfitting (low bias & high variance)**:
  - When a model learns the training data too well, including its noise and outliers, and performs poorly on new, unseen data.



- 8. **Underfitting (high bias & low variance)**:
  - When a model fails to capture the underlying trer performance both on training and testing data.

How to measure/evaluate our models?

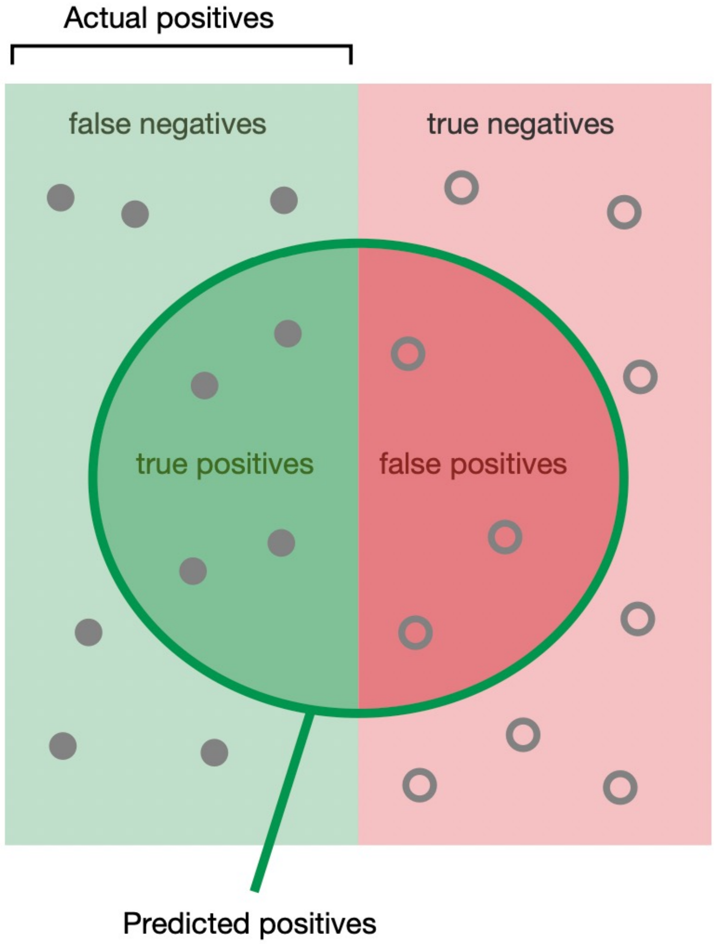
- **1. Mean-squared prediction error:**
  - a measure used to evaluate performance of g(x) for regression problems
    - Training error: $MSE_{Tr} = Ave_{i \in Tr}[y_i - g(x_i)]^2$ (biased when overfitting)
    - Test error: $MSE_{Te} = Ave_{i \in Te}[y_i - g(x_i)]^2$ (mitigates bias by using **out of sample** data)

- **2. Misclassification error rate:**
  - a measure used to evaluate performance of g(x) for classification problems
    - Training error: $Err_{Tr} = Ave_{i \in Tr} I[y_i \neq g(x_i)]$
    - Test error: $Err_{Te} = Ave_{i \in Te} I[y_i \neq g(x_i)]$

Problem: not a good measure when data are imbalanced.

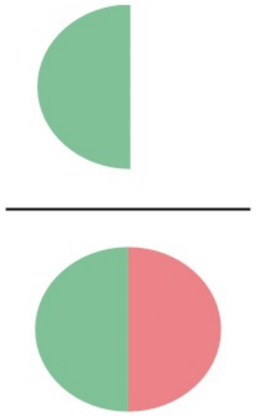
https://mlu-explain.github.io/precision-recall/

How to measure/evaluate our models? More measures…

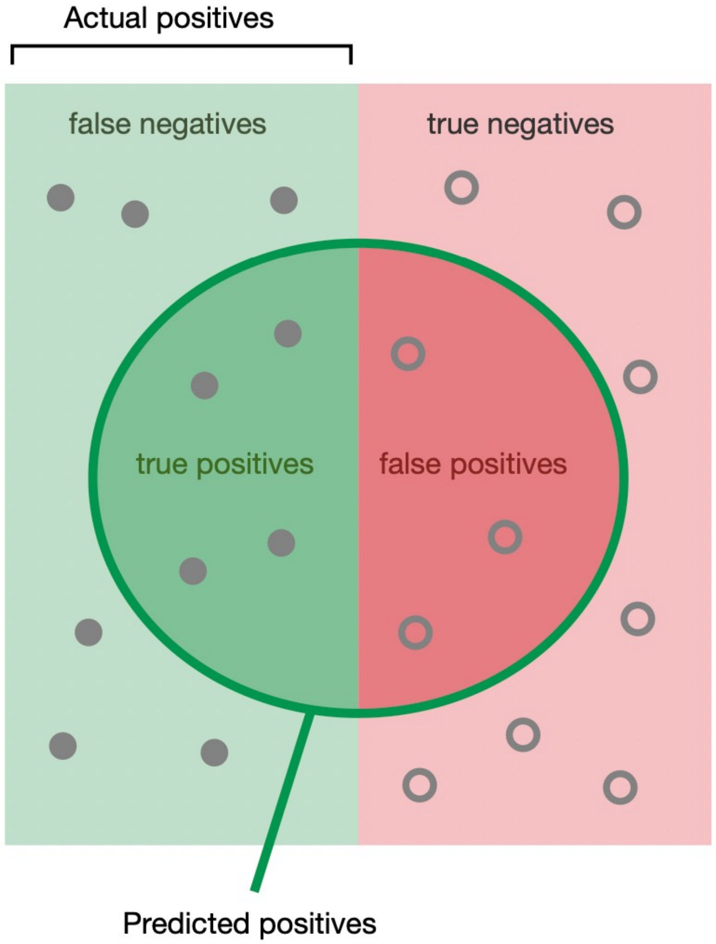


What proportion of predicted positives are in fact positive?

Precision =

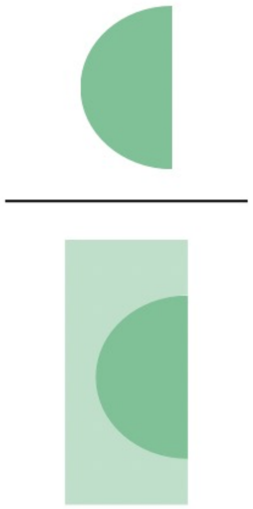


How to measure/evaluate our models? More measures…

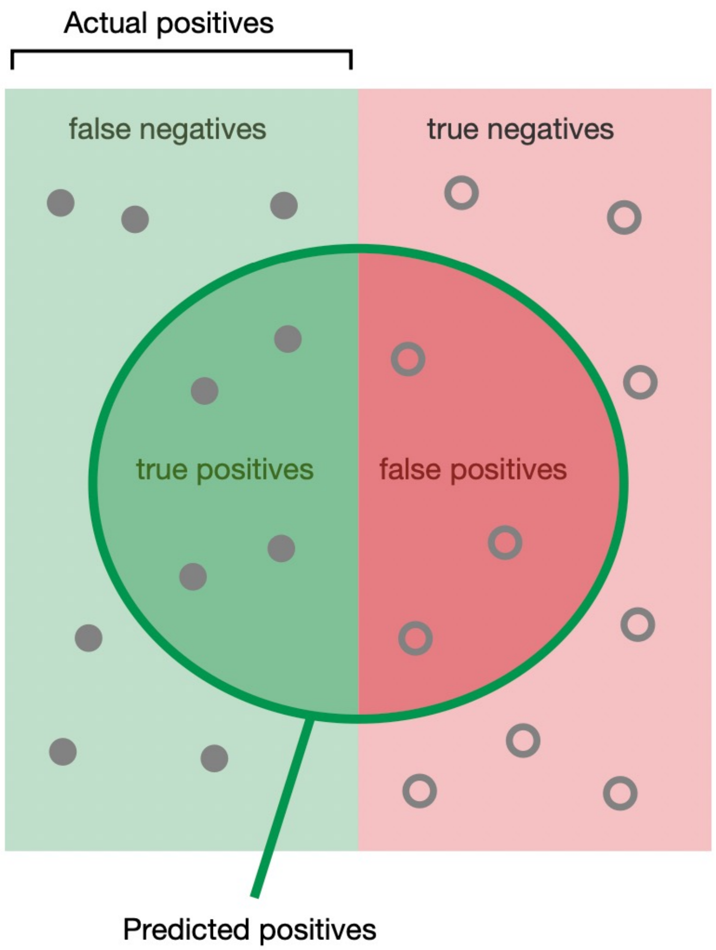


What proportion of the actual positives are predicted to be positive?

Recall =



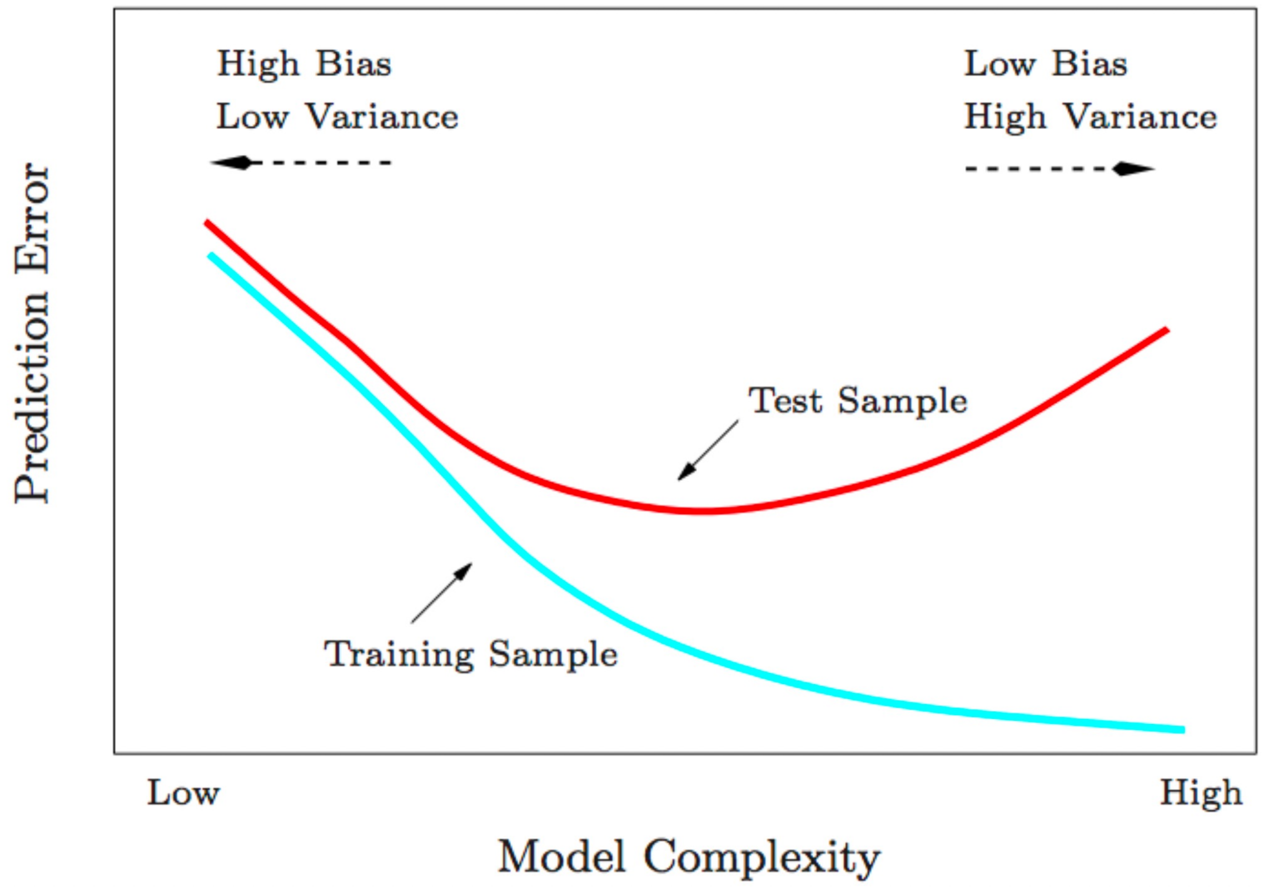
How to measure/evaluate our models? More measures…

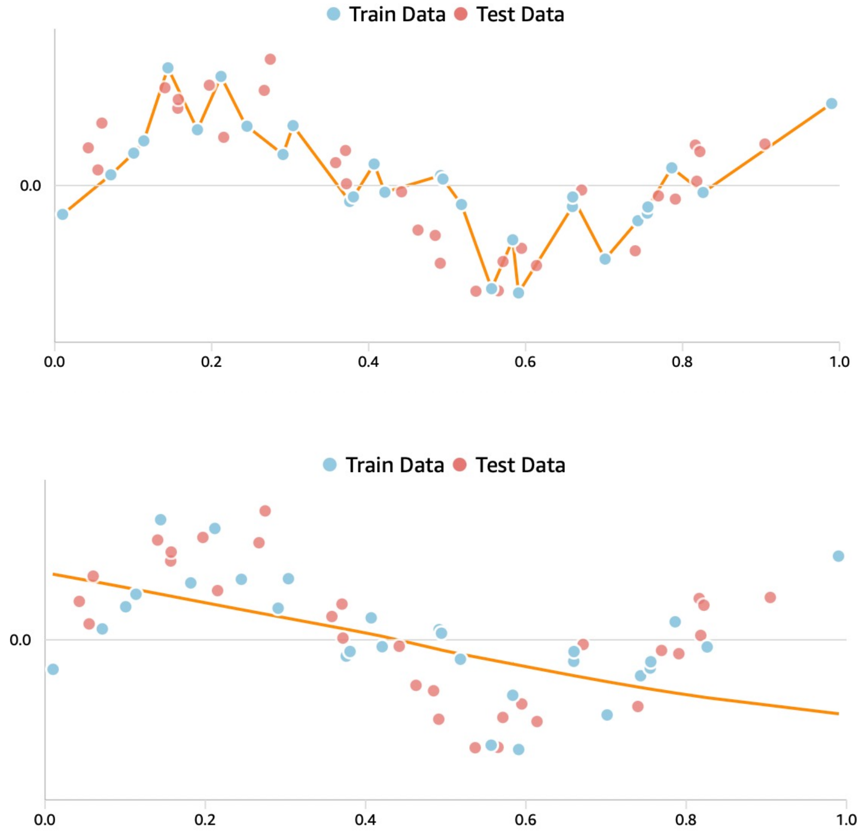


F1 is a performance measure that balances precision and recall.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

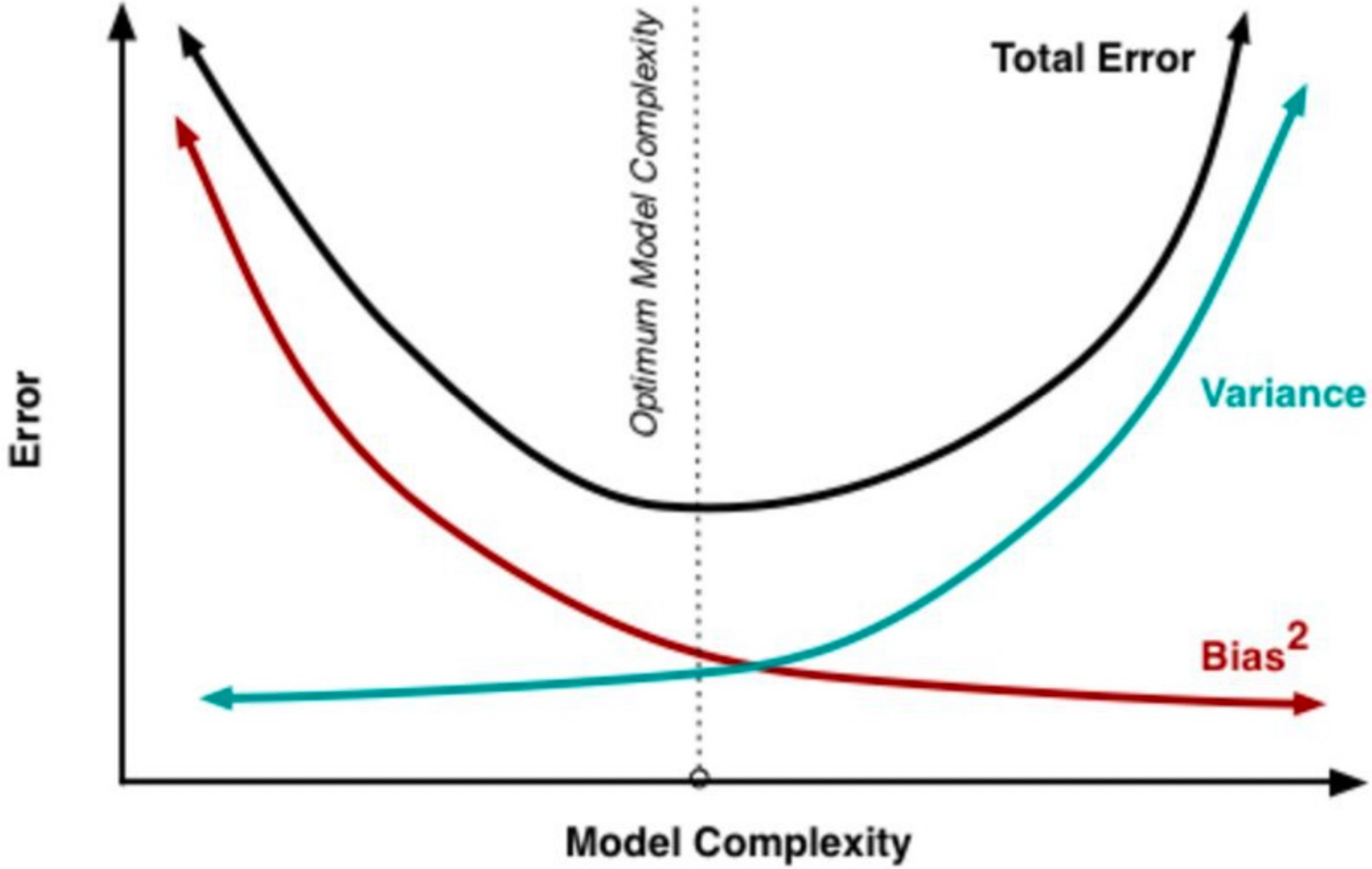
Decomposing Error into Bias and Variance





Variance: Variance captures how much the model's predictions vary for different training sets.
Bias: Bias measures how much on average the predictions of a machine learning model are different from the correct values.

Decomposing Error into Bias and Variance



Decomposing Error into Bias and Variance

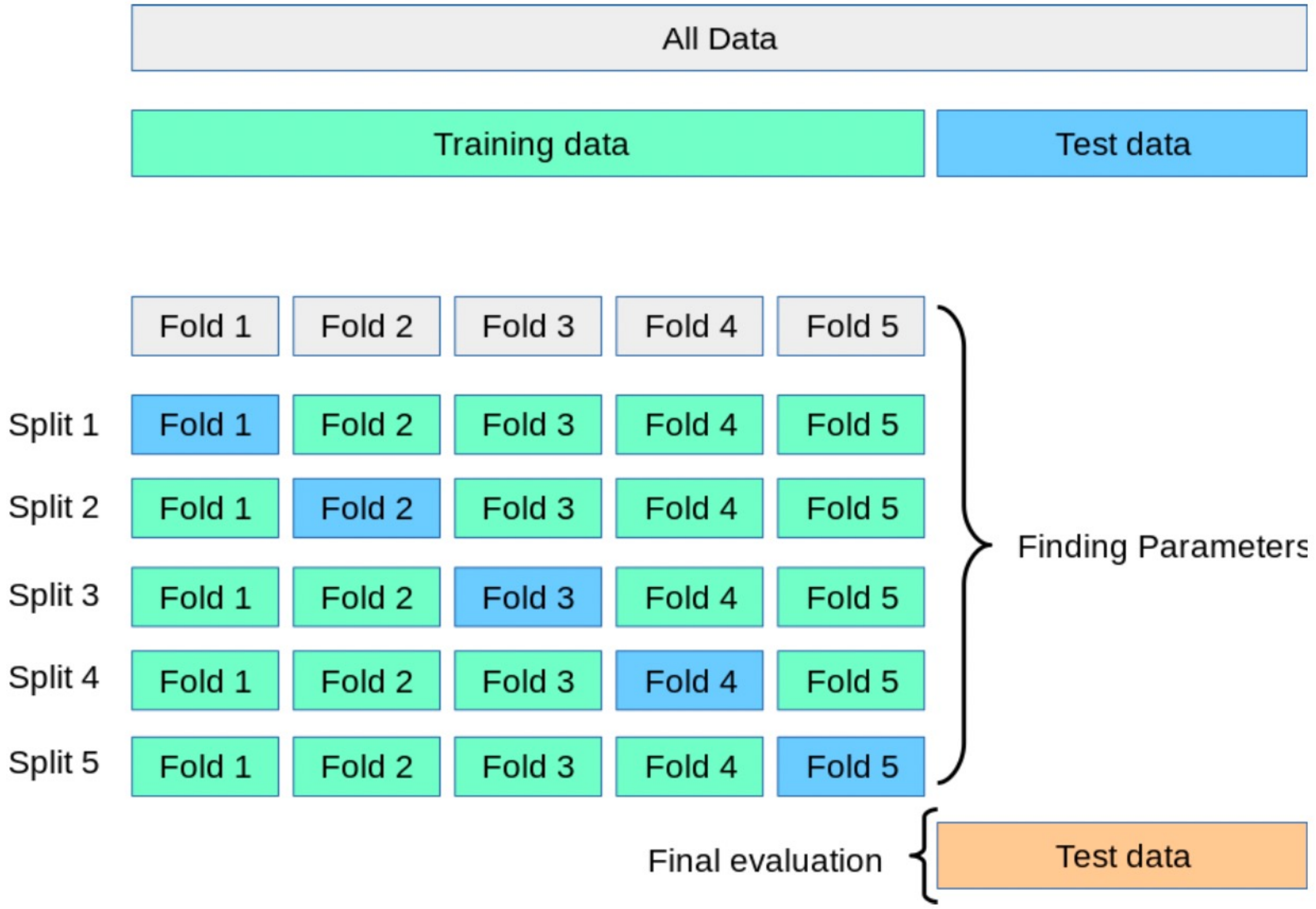
Models are biased when:

- Parametric: The form of the model does not incorporate all the necessary variables (omitted variable bias)
- Parametric: The functional form is too simple (e.g. a linear approximation)
- Non-parametric: The model provides too much smoothing.

Models are variable when:

- Parametric: The form of the model incorporates too many variables.
- Parametric: The functional form is too complex.
- Non-parametric: The model does not provide enough smoothing.

How to measure/evaluate our models? More measures…



Divide data into K roughly equal-sized parts ($K = 5$ here)

K-fold cross validation error

- K can be anything; popular values are 5, 10, n.
- The cross-validated error rate tends to be closer to the true error rate than to the apparent error rate.
- The computational cost can become a concern.

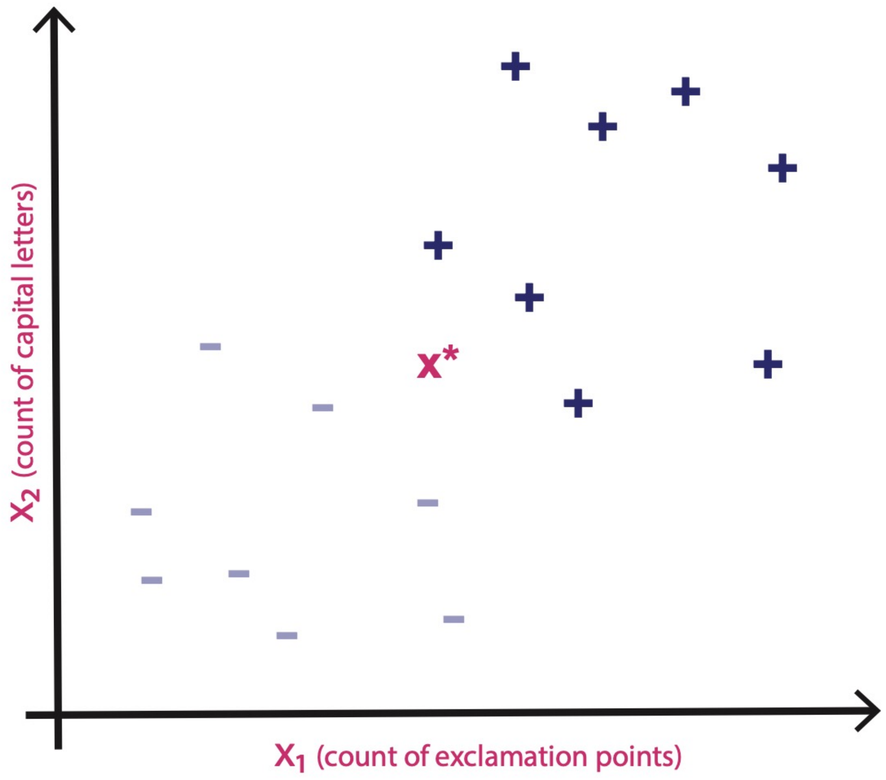
Note:
Since each training set is only $(K - 1)/K$ as big as the original training set, the estimates of prediction error will typically be biased upward.
This bias is minimized when K = n (LOOCV)(Leave One Out Cross-Validation), but this estimate has high variance, as noted earlier.

K-Nearest Neighbors (KNN)

- A good start to learn machine learning as it is super easy to understand.

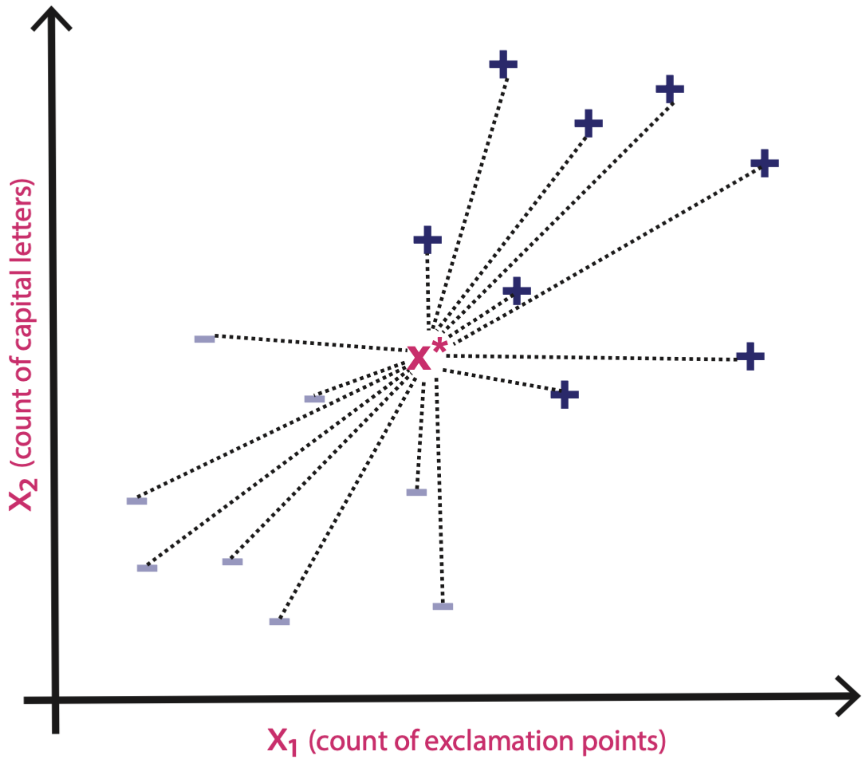
- KNN is a simple example of a **non-parametric & supervised model** where the model structure is determined from the dataset with no assumptions about the underlying data distribution.

K-Nearest Neighbors (KNN)



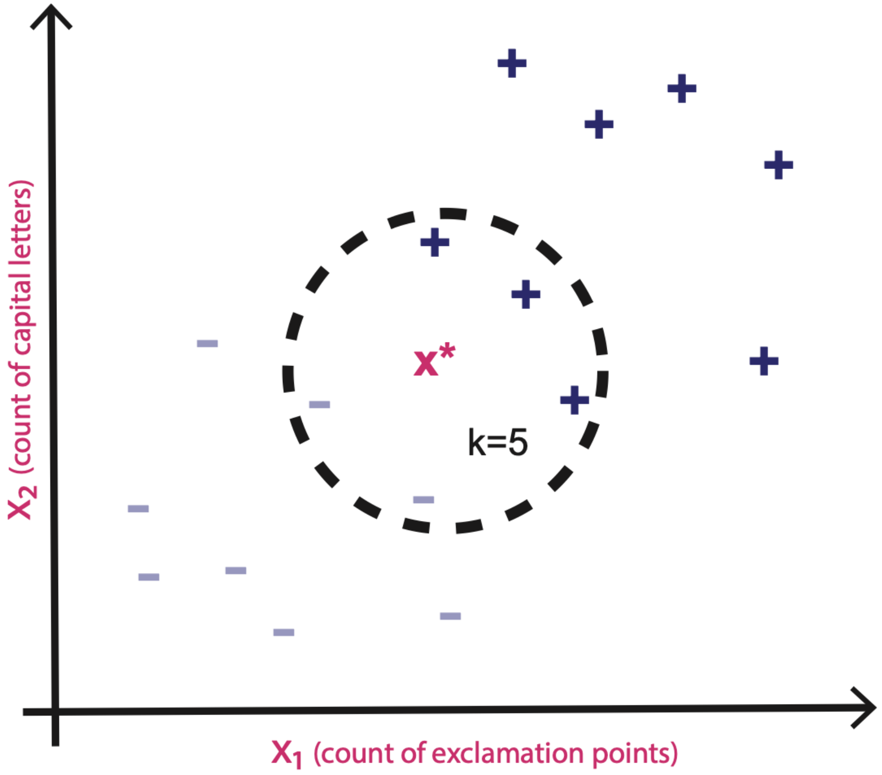
1. Calculate the distance between each x_i and reference point x^*
2. Find the k closest neighbors to x^*
3. Return the majority class of $y_i \in N_k(\mathbf{x}^*)$

K-Nearest Neighbors (KNN)



1. **Calculate the distance between each** x_i **and reference point** x^*
2. Find the k closest neighbors to x^*
3. Return the majority class of $y_i \in N_k(\mathbf{x}^*)$

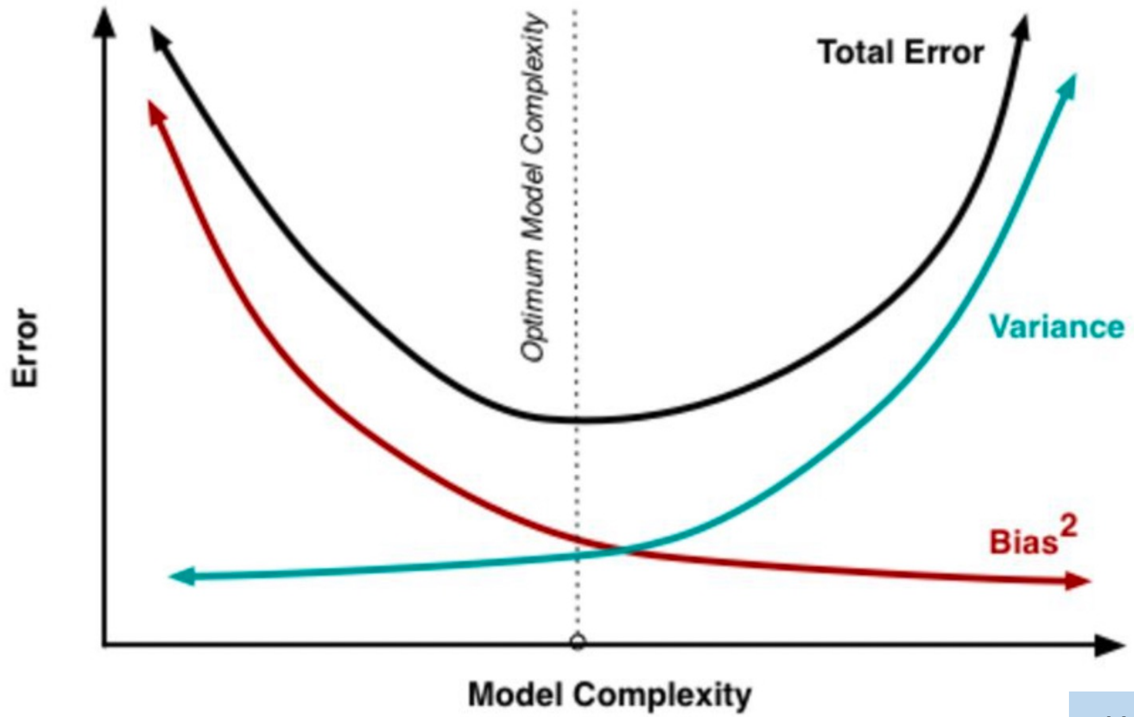
K-Nearest Neighbors (KNN)



1. Calculate the distance between each x_i and reference point x^*
2. **Find the k closest neighbors to x^***
3. Return the majority class of $y_i \in N_k(\mathbf{x}^*)$

How to choose K?

- 1. $\sqrt{n}$: square root of n (number of data points in the training dataset)
- 2. Try different K and evaluate models



-K

Real Example Now!

KNN - Pros & Cons

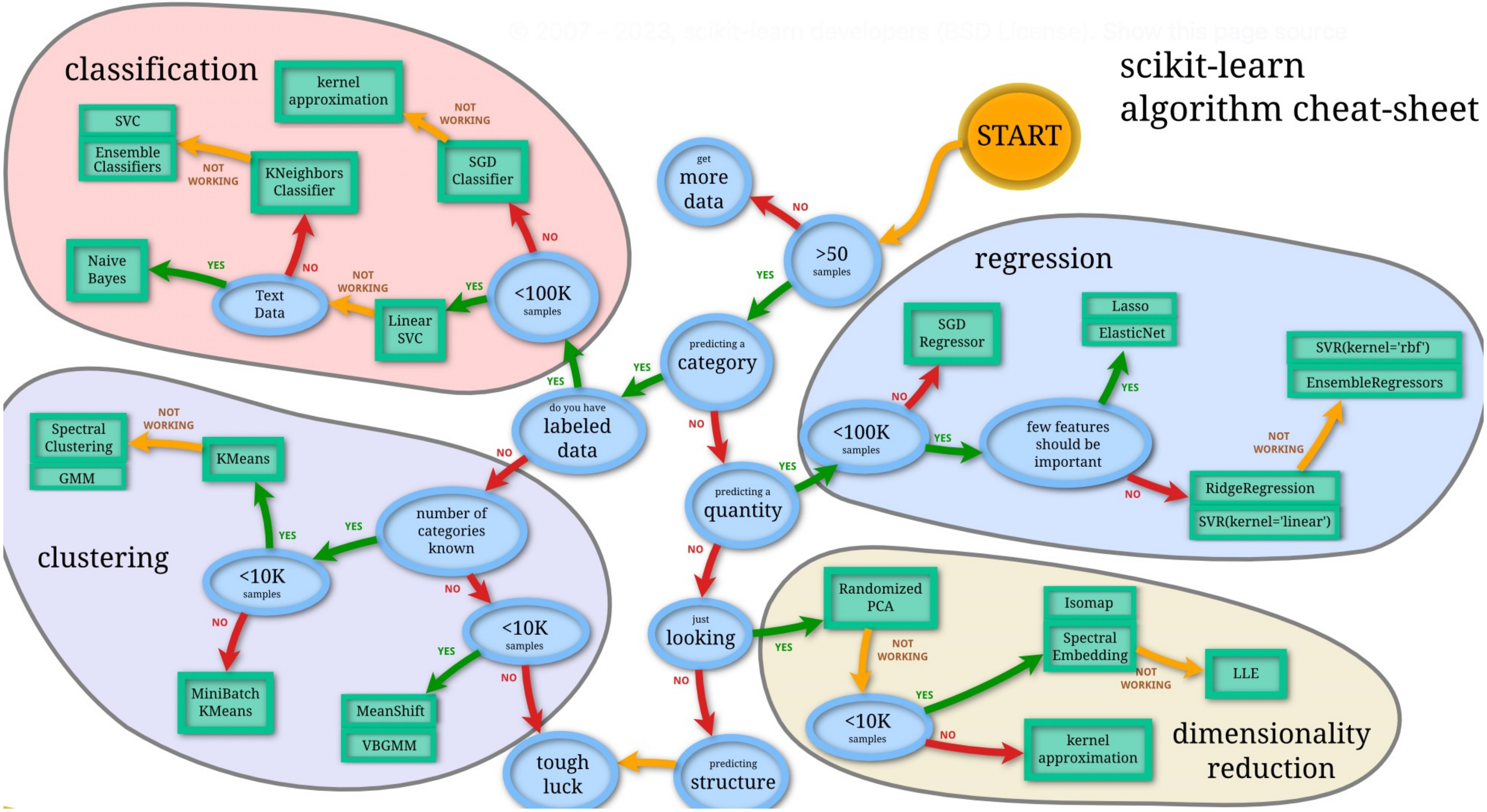
- Pros:
  - Works well for non-linear data.
  - Model adapts easily to changes in the dataset.

- Cons:
  - Computationally expensive, especially when the dataset grows, because it has to compute the distance to every single data point in the dataset for every test point.
  - Sensitive to irrelevant features and the scale of the data. Often, features need to be normalized. Can be heavily swayed by outliers if an inappropriate k value is chosen.
  - KNN does not work well in high dimensions unless data lie on or close to a low-dimensional subspace, so can be solved by dimension reduction.

KNN – Application in Political Science

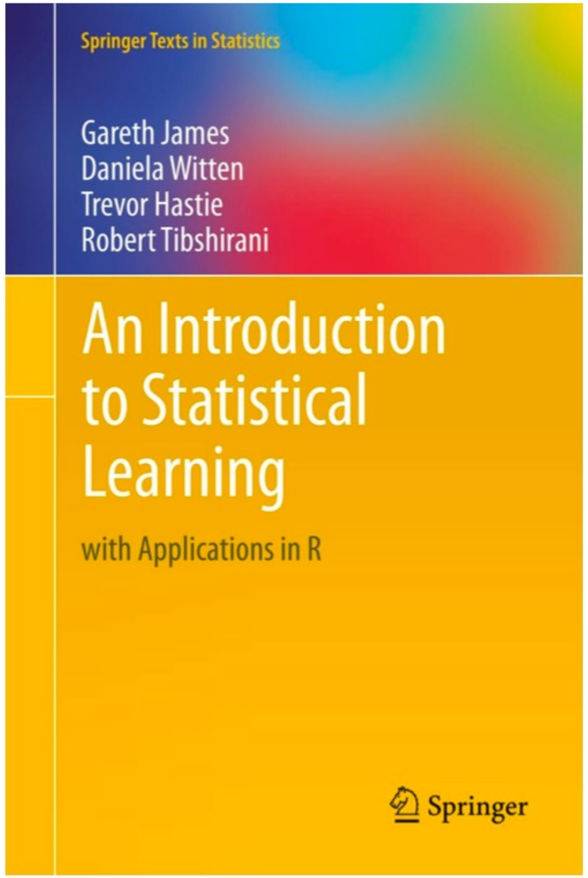
- Carroll, R. J., & Kenkel, B. (2019). Prediction, proxies, and power. American Journal of Political Science, 63(3), 577-593.
  - They model dispute outcomes as a function of the participants’ military capabilities (26 features).
  - They propose Dispute Outcome Expectations (DOE) score using machine learning.
  - KNN is one of their model.

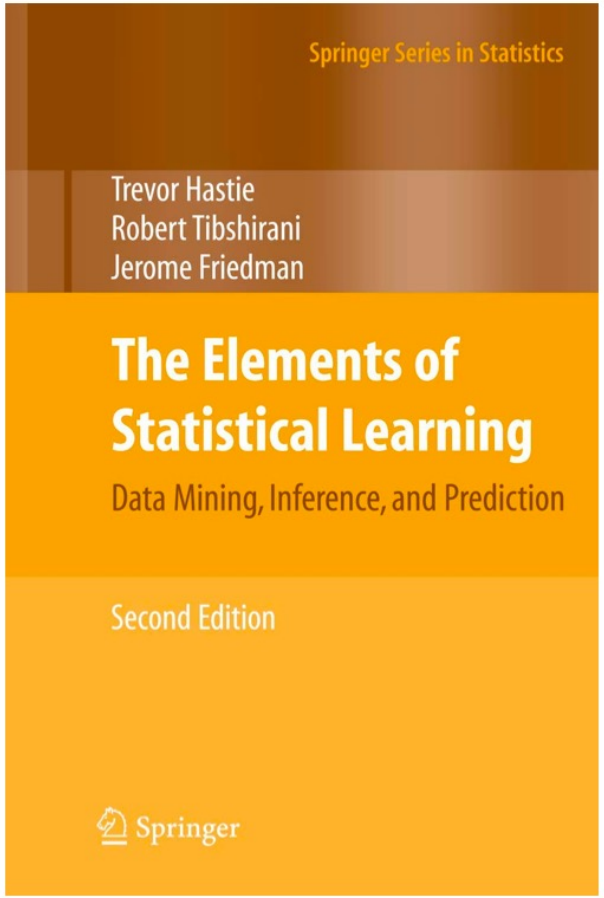
- Sometimes can also use machine learning to predict counterfactuals and compare that with observables to study causality.



https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

Further learning resources





Further learning resources

- Public courses: Applied Machine Learning for Social Science
  - Blake Miller, Department of Methodology, London School of Economics.
  - Friedrich Geiecke, Department of Methodology, London School of Economics.
  - Slides, replication codes: https://github.com/lse-my474/lectures

Week	Topic
1	What is Machine Learning?
2	Generalization, Inference, Prediction, and Causality
3	Linear Discriminant Analysis, Logistic Regression
4	Gradient Descent, Bootstrap, Cross-Validation, Hyperparameters
5	Regularization, Decision Trees
6	*Reading Week*
7	Support Vector Machines, Active Learning
8	Bias, Fairness, Accountability, and Transparency in ML
9	Ensembles, Bagging, Boosting
10	Dimension Reduction and Clustering
11	Neural Networks

- Thank you!