1-Introduction to Machine Learning

Reza Rezazadegan

Sharif University of Technology

October 1, 2022

Course info

Machine Learning course at the Department of Mathematics, Sharif University, Fall 2022
Instructor: Reza Rezazadegan
Course webpage: www.rezazadegan.ir/MLcourse
Pre-requisites: familiarity with linear algebra, multivariable calculus, probability theory and basic Python programming
Texts:

- Rezazadegan, Applications of Artificial Intelligence and Big Data in Industry 4.0 Technologies, in Industry 4.0 Vision for Energy and Materials: Enabling Technologies and Case Studies, Wiley, 2022
- Blum, et al, Foundations of Data Science
- Aurelien Geron, Hands-on Machine Learning with Scikit-Learn

TAs: Ali Bagheri, Qazal Farahani
Code: Jupyter notebooks used in this course are available at www.github.com/rezareza007/MLcourse
Evaluation: by student projects or presentation

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do "smart" things, as humans do. Such as recognizing faces, filtering spam, driving cars,...

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do “smart” things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do "smart" things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.
- Three branches of AI: Symbolic AI, Machine Learning, Neural Networks

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do “smart” things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.
- Three branches of AI: Symbolic AI, Machine Learning, Neural Networks
- **Symbolic AI:** intelligence can be reduced to manipulation of symbols, in particular, logic (e.g. expert systems).

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do "smart" things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.
- Three branches of AI: Symbolic AI, Machine Learning, Neural Networks
- **Symbolic AI:** intelligence can be reduced to manipulation of symbols, in particular, logic (e.g. expert systems).
- **Machine Learning (ML)** means enabling computers to learn from data, without having to code for each individual case.

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do "smart" things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.
- Three branches of AI: Symbolic AI, Machine Learning, Neural Networks
- **Symbolic AI:** intelligence can be reduced to manipulation of symbols, in particular, logic (e.g. expert systems).
- **Machine Learning (ML)** means enabling computers to learn from data, without having to code for each individual case.
- For example face or fingerprint recognition, speech recognition, guessing the next work in a text,...

What is Artificial Intelligence (AI)?

- AI is enabling machines, in particular computers, to do “smart” things, as humans do. Such as recognizing faces, filtering spam, driving cars,...
- Intelligence can be defined as the ability to learn from experience and generalize.
- Three branches of AI: Symbolic AI, Machine Learning, Neural Networks
- **Symbolic AI:** intelligence can be reduced to manipulation of symbols, in particular, logic (e.g. expert systems).
- **Machine Learning (ML)** means enabling computers to learn from data, without having to code for each individual case.
- For example face or fingerprint recognition, speech recognition, guessing the next work in a text,...
- **Neural Networks and Deep Learning:** tries to mimic the working of neurons in the brain; hierarchically reduces a given problem into simpler ones.

- ML methods can “learn” (i.e. estimate) complex relations among the quantities in the data.

- ML methods can “learn” (i.e. estimate) complex relations among the quantities in the data.
- Difference between ML and science: Science starts with principles i.e. clear-cut relationships between some quantities such as $F = ma$.

- ML methods can "learn" (i.e. estimate) complex relations among the quantities in the data.
- Difference between ML and science: Science starts with principles i.e. clear-cut relationships between some quantities such as $F = ma$.
- Relations between quantities are not clear-cut in real life: the temperature of a room as a function of heater degree and duration or operation. Or currency exchange rate as a function of time.
- Computers can only understand numbers. Categories, text, images, videos and sound can be turned into numbers. (Won't be covered in this course, except for categories.)

- ML methods can "learn" (i.e. estimate) complex relations among the quantities in the data.
- Difference between ML and science: Science starts with principles i.e. clear-cut relationships between some quantities such as $F = ma$.
- Relations between quantities are not clear-cut in real life: the temperature of a room as a function of heater degree and duration or operation. Or currency exchange rate as a function of time.
- Computers can only understand numbers. Categories, text, images, videos and sound can be turned into numbers. (Won't be covered in this course, except for categories.)
- Two types of numerical data:
  - Data with units e.g. blood pressure, temperature of a furnace
  - Raw e.g. the pixel values of an image

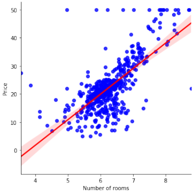
- ML methods can “learn” (i.e. estimate) complex relations among the quantities in the data.
- Difference between ML and science: Science starts with principles i.e. clear-cut relationships between some quantities such as $F = ma$.
- Relations between quantities are not clear-cut in real life: the temperature of a room as a function of heater degree and duration or operation. Or currency exchange rate as a function of time.
- Computers can only understand numbers. Categories, text, images, videos and sound can be turned into numbers. (Won’t be covered in this course, except for categories.)
- Two types of numerical data:
  - Data with units e.g. blood pressure, temperature of a furnace
  - Raw e.g. the pixel values of an image
- Raw data is more suitable for neural networks and deep learning.

AI as function approximation

- AI is, simply put, approximating functions and relations!

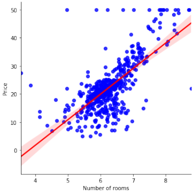
AI as function approximation

- AI is, simply put, approximating functions and relations! If we know the values of f at points $x_1, x_2, \ldots, x_n$ then how can we infer $f(x)$ for general x?



AI as function approximation

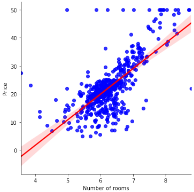
- AI is, simply put, approximating functions and relations! If we know the values of f at points $x_1, x_2, \ldots, x_n$ then how can we infer $f(x)$ for general x?



- Examples of f:
  - Gives us the probability of a loan applicant defaulting, based on his/her demographic and financial data.
  - Takes an image as an input and tells us what objects or faces are in it
  - Tells us the stock prices as function of (future) time

AI as function approximation

- AI is, simply put, approximating functions and relations! If we know the values of f at points $x_1, x_2, \ldots, x_n$ then how can we infer $f(x)$ for general x?



- Examples of f:
  - Gives us the probability of a loan applicant defaulting, based on his/her demographic and financial data.
  - Takes an image as an input and tells us what objects or faces are in it
  - Tells us the stock prices as function of (future) time
- Function spaces are infinite dimensional! To approximate functions (or relationships) we need to make an assumption on the function i.e. assuming it belongs to a parametric family of functions.

AI as function approximation, continued

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**

AI as function approximation, continued

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**
- Assuming f is a linear combination of a set of basis functions: **Support Vector Machines!**

$$f(x) = a_0 + a_1 K(x, x_1) + a_2 K(x, x_2) + \cdots + a_n K(x, x_n) \tag{1}$$

AI as function approximation, continued

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**
- Assuming f is a linear combination of a set of basis functions: **Support Vector Machines!**

$$f(x) = a_0 + a_1 K(x, x_1) + a_2 K(x, x_2) + \cdots + a_n K(x, x_n) \tag{1}$$

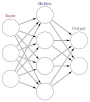
- Assuming f is a composition of functions of the form $\phi(z_1, z_2, \ldots, z_k) = h(\sum_i a_i z_i)$, where h is a nonlinear function: **Neural Networks and Deep Learning!**

AI as function approximation, continued

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**
- Assuming f is a linear combination of a set of basis functions: **Support Vector Machines!**

$$f(x) = a_0 + a_1 K(x, x_1) + a_2 K(x, x_2) + \cdots + a_n K(x, x_n) \quad (1)$$

- Assuming f is a composition of functions of the form $\phi(z_1, z_2, \ldots, z_k) = h(\sum_i a_i z_i)$, where h is a nonlinear function: **Neural Networks and Deep Learning!**

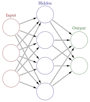


AI as function approximation, continued

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**
- Assuming f is a linear combination of a set of basis functions: **Support Vector Machines!**

$$f(x) = a_0 + a_1 K(x, x_1) + a_2 K(x, x_2) + \cdots + a_n K(x, x_n) \tag{1}$$

- Assuming f is a composition of functions of the form $\phi(z_1, z_2, \ldots, z_k) = h(\sum_i a_i z_i)$, where h is a nonlinear function: **Neural Networks and Deep Learning!**

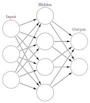


- Assuming the value of f at a new point x is the average of $f(x_i)$ where x_i are the nearest neighbors of x: **k-Nearest Neighbors!**

- Assuming the function f is linear or polynomial: **Linear or polynomial regression!**
- Assuming f is a linear combination of a set of basis functions: **Support Vector Machines!**

$$f(x) = a_0 + a_1 K(x, x_1) + a_2 K(x, x_2) + \cdots + a_n K(x, x_n) \qquad (1)$$

- Assuming f is a composition of functions of the form $\phi(z_1, z_2, \ldots, z_k) = h(\sum_i a_i z_i)$, where h is a nonlinear function: **Neural Networks and Deep Learning!**



- Assuming the value of f at a new point x is the average of $f(x_i)$ where x_i are the nearest neighbors of x: **k-Nearest Neighbors!**
- No Free Lunch theorem: without any assumptions on the data, no method is better than any other.

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

- The target variable can be discrete (**classification**) or continuous (**regression**).

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

- The target variable can be discrete (**classification**) or continuous (**regression**).
- Most classification methods have a regression counterpart and vice versa.

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

- The target variable can be discrete (**classification**) or continuous (**regression**).
- Most classification methods have a regression counterpart and vice versa.
- **Unsupervised Learning:** Learning from unlabeled data e.g. clustering or dimensional reduction.

Different categories of Machine Learning

- **Supervised Learning:** Learning from labeled data.
- Learning the relation between *independent variables* (**Features**) and the *target variable* (**Label**).

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
5	5.4	3.9	1.7	0.4	setosa
6	4.6	3.4	1.4	0.3	setosa
7	5.0	3.4	1.5	0.2	setosa
8	4.4	2.9	1.4	0.2	setosa
9	4.9	3.1	1.5	0.1	setosa

- The target variable can be discrete (**classification**) or continuous (**regression**).
- Most classification methods have a regression counterpart and vice versa.
- **Unsupervised Learning:** Learning from unlabeled data e.g. clustering or dimensional reduction.
- **Reinforcement Learning:** Optimizing the behavior of an agent in an environment. Used e.g. in automated playing of games e.g. chess, robotics,...

Instance-based, model-based and rule-based machine learning

- **Instance-based:** prediction (inference) is done based on "similar" data instances. It needs to keep (a subset) of training data for prediction (inference).

Instance-based, model-based and rule-based machine learning

- **Instance-based:** prediction (inference) is done based on “similar” data instances. It needs to keep (a subset) of training data for prediction (inference).
- Examples: KNN, Support Vector Machine

Instance-based, model-based and rule-based machine learning

- **Instance-based:** prediction (inference) is done based on "similar" data instances. It needs to keep (a subset) of training data for prediction (inference).
- Examples: KNN, Support Vector Machine
- **Model-based:** learns a model (hypothesis) for the whole dataset.

Instance-based, model-based and rule-based machine learning

- **Instance-based:** prediction (inference) is done based on “similar” data instances. It needs to keep (a subset) of training data for prediction (inference).
- Examples: KNN, Support Vector Machine
- **Model-based:** learns a model (hypothesis) for the whole dataset.
- Examples: Linear regression, neural networks

Instance-based, model-based and rule-based machine learning

- **Instance-based:** prediction (inference) is done based on "similar" data instances. It needs to keep (a subset) of training data for prediction (inference).
- Examples: KNN, Support Vector Machine
- **Model-based:** learns a model (hypothesis) for the whole dataset.
- Examples: Linear regression, neural networks
- **Rule-based:** Learns a bunch of rules, each for a subset of the data.
- Examples: Learning classifier systems, association rule mining

AI Explainability

- ML methods need to be **explainable** i.e. we need to understand how an AI method arrives at a conclusion.

AI Explainability

- ML methods need to be **explainable** i.e. we need to understand how an AI method arrives at a conclusion.
- Black-box AI can be a fool!

Figure: Image classifier is still confident about its predictions when 95% of the picture is removed! Source: B. Carter *et al.*, "What made you do this? Understanding black-box decisions with sufficient input subsets"

AI Explainability

- ML methods need to be **explainable** i.e. we need to understand how an AI method arrives at a conclusion.
- Black-box AI can be a fool!

Figure: Image classifier is still confident about its predictions when 95% of the picture is removed! Source: B. Carter *et al.*, "What made you do this? Understanding black-box decisions with sufficient input subsets"

- The more parameters a model has, the less explainable it becomes!

- The more parameters a model has, the less explainable it becomes!
- John von Neuman: “With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.”

- The more parameters a model has, the less explainable it becomes!
- John von Neuman: “With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.”

- Always prefer the simplest methods that gets the job done! (Occam’s Razor)

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.
- A good model is one that generalizes well to unseen data i.e. is accurate on data not in the training data, as well.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.
- A good model is one that generalizes well to unseen data i.e. is accurate on data not in the training data, as well.
- Thus, data is divided into *training set* and *test set*.

Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.
- A good model is one that generalizes well to unseen data i.e. is accurate on data not in the training data, as well.
- Thus, data is divided into *training set* and *test set*.
- Typically 10% to 30% of data is reserved for test.

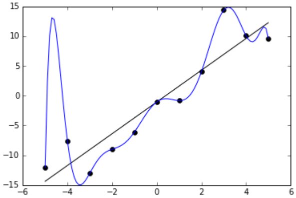
Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.
- A good model is one that generalizes well to unseen data i.e. is accurate on data not in the training data, as well.
- Thus, data is divided into *training set* and *test set*.
- Typically 10% to 30% of data is reserved for test.
- A model that has high accuracy on the training set but low accuracy on the test set suffers from *overfitting*.

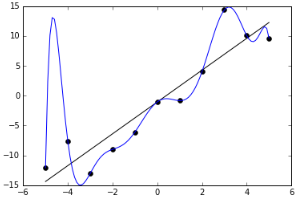
Model accuracy and generalization

- Remember: we have a dataset: $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$ and we want a model $f(x)$ that predicts the label y for $x \notin D$. The components of the x_i are called *features* and the y_i are called the *labels* of the data.
- *Accuracy* in supervised learning measures how good a classification or regression model f fits the training data.
- Accuracy for regression: $R^2 = \sum_i (y_i - f(x_i))^2$.
- Accuracy for classification: the fraction of correct predictions (true positives): $TP/(TP + FP)$.
- A good model is one that generalizes well to unseen data i.e. is accurate on data not in the training data, as well.
- Thus, data is divided into *training set* and *test set*.
- Typically 10% to 30% of data is reserved for test.
- A model that has high accuracy on the training set but low accuracy on the test set suffers from *overfitting*.

- **Overfitting:** when the model fits the training data well but does not generalize as well.

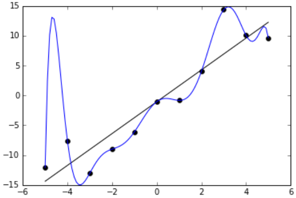


- **Overfitting:** when the model fits the training data well but does not generalize as well.



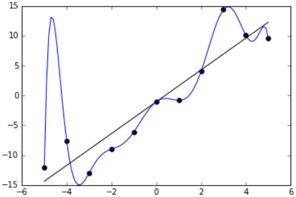
- Overfitting happens when we train a complex model on a small dataset.

- **Overfitting:** when the model fits the training data well but does not generalize as well.



- Overfitting happens when we train a complex model on a small dataset.
- To avoid overfitting we should use simpler models or more data.

- **Overfitting:** when the model fits the training data well but does not generalize as well.



- Overfitting happens when we train a complex model on a small dataset.
- To avoid overfitting we should use simpler models or more data.
- We can also train more than one model together (Multi-task Learning).

Model bias and variance

- A model's *bias* is part of its generalization error which is due to wrong assumptions.
- A model's *variance* is its sensitivity to small variations in data.
- Bias-variance tradeoff: simpler models tend to have more bias while more complex models tend to have more variance.