STAT 432: Basics of Statistical Learning

KNN and Bias-Variance Trade-Off

Shiwei Lan, Ph.D. <shiwei@illinois.edu>

http://shiwei.stat.illinois.edu/stat432.html

February 6, 2019

University of Illinois at Urbana-Champaign

Overview

- The main goal of this lecture is to demonstrate a phenomenon called the bias-variance trade-off.
- We use k-nearest neighbors as the tool for this demonstration
- We also introduce two additional important concepts:
  - Tuning parameters
  - Cross-Validation for selecting tuning parameters

k-Nearest Neighbors

Prediction

- Let's consider a regression model,

  $$Y = f(X) + \epsilon,$$

  where E$(\epsilon) = 0$ and Var$(\epsilon) = \sigma^2$.
- Suppose that from a set of training data, we are able to estimate the regression function as $\widehat{f}$ (called "f-hat").
- We can then predict the value of Y at a target point x_0 by using $\widehat{f}(x_0)$.
- Let's consider a very simple approach for estimating $\widehat{f}$, called k-nearest neighbors

k-Nearest Neighbors

- k-Nearest Neighbor (kNN) is a nonparametric method that predicts the target point $\mathbf{x}$ with averages of nearby observations in the training data
- For regression, the prediction at a given target point x_0 is

  $$\widehat{y} = \frac{1}{k} \sum_{x_i \in N_k(x_0)} y_i,$$

  where $N_k(x_0)$ defines the k samples from the training data (in terms of their feature values) that are closest to x_0.
- How to calculate the distance?
- What k should we use?

More on Distance Measures

- By default, we use Euclidean distance (ℓ_2 norm) for continuous variables

$$d^2(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|_2^2 = \sum_{i=1}^{p} (u_i - v_i)^2$$

  Hence the neighborhood is not invariant to the scaling of the variables.

- Mahalanobis distance is scale-invariant

$$d^2(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u} - \boldsymbol{v})^\mathsf{T} \Sigma^{-1} (\boldsymbol{u} - \boldsymbol{v}),$$

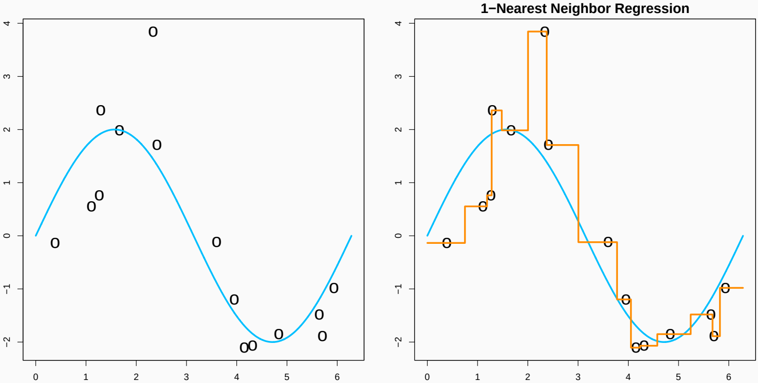
  where Σ is a covariance matrix. In practice, we can use the sample covariance matrix of the training data

k-Nearest Neighbors for Regression

- The the following data is observed, with only 1 feature, uniformly from $[0, 2\pi]$. The true model (blue line) is

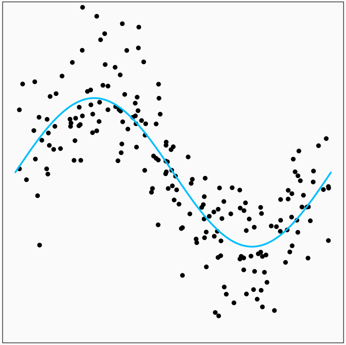
$$Y = 2\sin(X) + \epsilon,$$

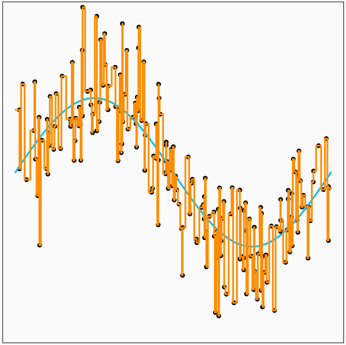
where ϵ is a standard normal error. We fit the data with 1NN.



k-Nearest Neighbors for Regression

Now we simulate 200 observations, and see how the model changes over k.

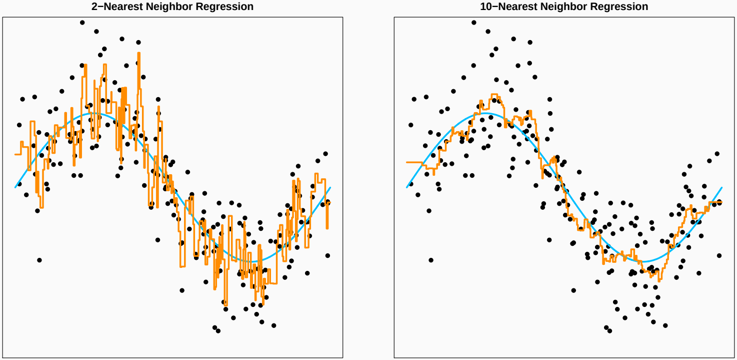




1−Nearest Neighbor Regression

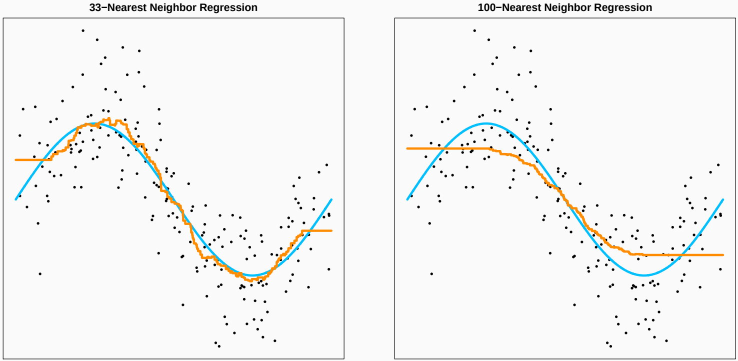
k-Nearest Neighbors for Regression

Now we simulate 200 observations, and see how the model changes over k.



k-Nearest Neighbors for Regression

The model becomes “smoother” as k increases. However, this the model seems to be “off” eventually.



33−Nearest Neighbor Regression

100−Nearest Neighbor Regression

Bias and Variance of a Model

- As k changes, the behaviour of the fitted model also changes.
  - When k is small, the model is “unstable”, but we are using the closest point, which approximate the target well.
  - When k is large, the model is stable, but it can be systematically biased.
- This phenomenon is called the bias-variance trade-off
- But we need to formally define what do we mean by bias and variance (of the estimator $\widehat{f}$)

Bias

- At any target point x_0, the bias of an estimator $\widehat{f}$ is defined as

$$f(x_0) - \mathsf{E}[\widehat{f}(x_0)]$$

- It can be understood this way:
  - Suppose we have many researchers, and each of them collect independently a set of 200 samples.
  - Then, each of them use KNN to give a predilection at the target point x_0, i.e., $\widehat{f}(x_0)$
  - Overall, does the mean of these predictions (averaged across all researchers) differ from the truth?

Variance

- At any target point x, the variance of an estimator $\widehat{f}$ is defined as

$$\mathsf{E}\big[\big(\widehat{f}(x_0) - \mathsf{E}[\widehat{f}(x_0)]\big)^2\big]$$

- It can be understood this way:
    - Suppose we have many researchers, and each of them collect independently a set of 200 samples.
    - Then, each of them use KNN to give a predilection at the target point x, i.e., $\widehat{f}(x)$
    - Overall, what is the variance of all these $\widehat{f}(x)$ values (regardless of whether they are accurate or not)

Bias-Variance Trade-Off

- To demonstrate the trade-off between these two components, lets consider two extreme cases: $k = 1$ and $k = n$
- When $k = 1$, we have low bias but high variance
- When $k = n$, we have low variance but high bias
- Let's analyze these two cases.

Case 1: $k = n$

- Regardless of what x_0 we are trying to predict, the fitted model is

$$\widehat{f} = \bar{y} = \frac{1}{n}\sum_i y_i,$$

  since every data point will be in the neighborhood of the target point.
- What is the variance of $\bar{y}$?

$$\begin{aligned}\text{Var}(\bar{y}) =& \text{Var}\Big(\frac{y_1 + y_2 + \ldots + y_n}{n}\Big)\\ =& \frac{1}{n^2}\sum_i \text{Var}(y_i)\end{aligned}$$

- Under some conditions, Var(y_i) should be finite, so the variance for this prediction is in the order of $1/n$, which goes to 0 as we collect more and more samples.

Case 1: $k = n$

- What is the bias of $\bar{y}$? That depends on which point we are predicting.
- However, $E[\bar{y}]$ is just the population mean.
- As long as the function $f(x)$ is not a constant, $f(x_0)$ is usually not the population mean.
- So its fair to say that there will be a substantial bias if we use $k = n$.
- Summary: for $k = n$, $\widehat{f}$ has small variance and large bias

Case 2: $k = 1$

- As the number of samples increases, the closest neighbor of x_0 is getting closer and closer.
  - Suppose we have p independent variables, all follows uniform distribution.
  - As long as x_0 is not a boundary point, lets look at a neighboring cube of x_0, with fixed width ϵ on each dimension.
  - The volume of this cube is ϵ^p
  - The probability of no sample point located within this cube is

$$\left(1 - \epsilon^p\right)^n \to 0$$

- Hence, its fair to say that, as the sample size grows, it is almost guaranteed that there exist a point very close to x_0

Case 2: $k = 1$

- Suppose this closest point is x^*
- As long as x_0 is not a point where $f(x)$ jumps,

$$f(x^*) \to f(x_0) \quad \text{as} \quad n \to \infty.$$

- What is our 1NN estimator $\widehat{f}(x_0)$? the y value corresponds to this x^*, which is simply

$$\begin{aligned} & \widehat{f}(x_0) = y^* = f(x^*) + \epsilon^* \\ \implies \qquad & \mathsf{E}[\widehat{f}(x_0)] = f(x^*) \end{aligned}$$

- Hence, 1NN is asymptotically unbiased — small bias

Case 2: $k = 1$

- What about the variance of 1NN?
- Since we only use 1 observation, and its almost always at a close neighbor of x_0, the variance of 1 observation is simply the variance of the error term, i.e.

$$\text{Var}(\widehat{f}(x_0)) \to \text{Var}(\epsilon) = \sigma^2$$

- This is a constant, and is much worse compared with $k = n$.
- Summary: for $k = 1$, $\widehat{f}$ has large variance and small bias

A General Formula

- The bise-variance trade-off can be formally understood by using this breakdown of the prediction error:

$$
\begin{aligned}
&\text{Prediction Error at } x_0\\
&= \mathsf{E}\left[\left(Y - \widehat{f}(x_0)\right)^2\right]\\
&= \mathsf{E}\left[\left(Y - f(x_0) + f(x_0) - \mathsf{E}[\widehat{f}(x_0)] + \mathsf{E}[\widehat{f}(x_0)] - \widehat{f}(x_0)\right)^2\right]\\
&= \underbrace{\mathsf{E}\left[(Y - f(x_0))^2\right]}_{\text{Irreducible Error}} + \underbrace{\left(f(x_0) - \mathsf{E}[\widehat{f}(x_0)]\right)^2}_{\text{Bias}^2} + \underbrace{\mathsf{E}\left[\left(\widehat{f}(x_0) - \mathsf{E}[\widehat{f}(x_0)]\right)^2\right]}_{\text{Variance}}
\end{aligned}
$$

Take-away

- $\mathsf{E}\left[(Y - f(x_0))^2\right]$ is the irreducible error term that cannot be avoided, because we cannot predict ϵ
- $\big(f(x_0) - \mathsf{E}[\widehat{f}(x_0)]\big)^2$ is the squared bias term that evaluates how the average of our estimator deviates from the truth
- $\mathsf{E}\left[\big(\widehat{f}(x_0) - \mathsf{E}[\widehat{f}(x_0)]\big)^2\right]$ is the variance term that reflects the sensitivity of the function estimate $\widehat{f}(x)$ to the training sample
- Important: no estimator can minimize both Bias^2 and Variance. We can only attempt to minimize the sum of the two.

Related Concepts

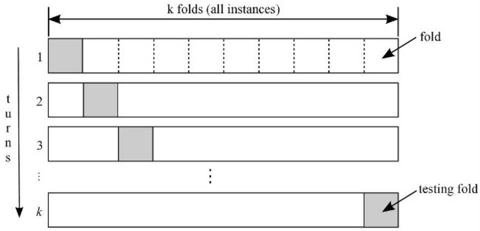
- bias-variance trade-off is also related to concepts such as model complexity, and over- and under-fitting.
- Over-fitting happens when the model performs well on the training sample, but not on the testing sample. Under-fitting is just the opposite.
- Model complexity can be measured in different ways. In statistics, we often use the degrees of freedom (number of parameters in a model)
- The degrees of freedom of a kNN model is roughly n/k
  - intuition: if neighborhoods don't overlap, there would be n/k neighborhoods, with one parameter for each

Balancing the Bias-Variance Trade-off

- Essentially, we need to choose k to minimize the sum of Bias^2 and Variance
- A common approach is called cross-validation (CV).
- The basic idea is to choose k that minimize the prediction error using testing data (since we cannot evaluate Bias^2 and Variance directly)
- A 10-fold CV is carried out as follows

Cross-Validation

- Randomly split the data into 10 equal sized subsamples
- Fit the model using 9 out of 10 subsamples as training data and calculate the testing error using the remaining one.
- Alternate the testing sample, and average the total of 10 experiments



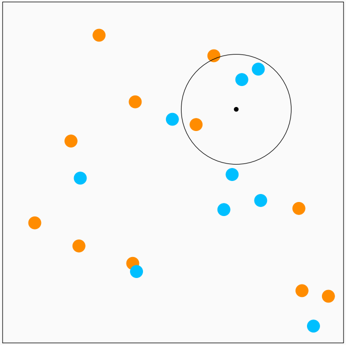
KNN Classification

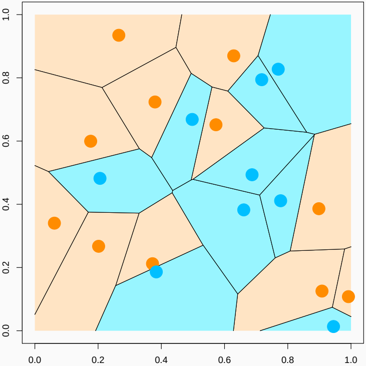
Classification

- There are usually two types of classification goals:
- Hard Classification predicts the label of the outcome
    - Example: $f : \mathbf{R}^p \to \{0, 1\}$
- Soft Classification outputs the probability of observing each possible label
    - Example: $f : \mathbf{R}^p \to [0, 1]$ for the probability of observing “1” in a binary classification
- We will discuss hard classification first.

k-Nearest Neighbors in Classification

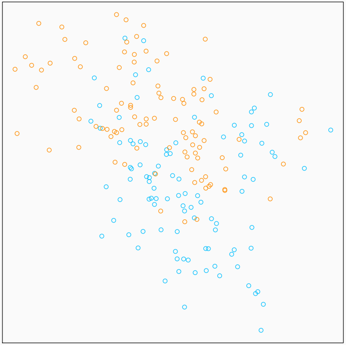
Similar to the regression case, the k-NN classification model does majority vote (the most prevalent class) within the neighborhood of a target point x. 1NN plot is a Voronoi tessellation





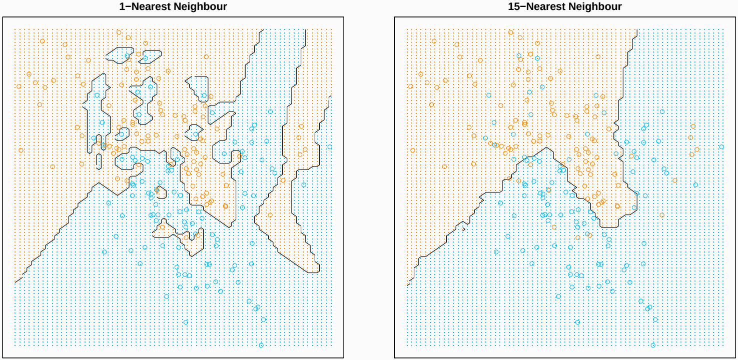
k-Nearest Neighbors in Classification

Let’s look at a classification example from the HTF text book.
(BLUE = 0, ORANGE = 1)



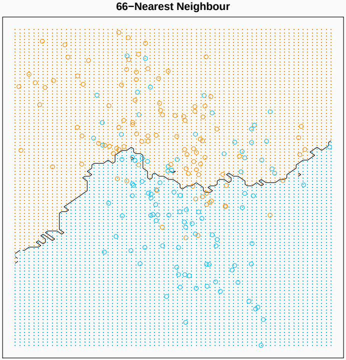
k-Nearest Neighbors in Classification

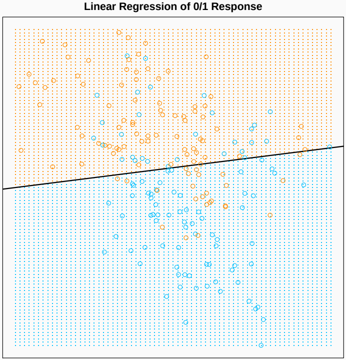
We fit k-NN classification model to the example. Of course, we would not expect 1NN to perform well...



k-Nearest Neighbors in Classification

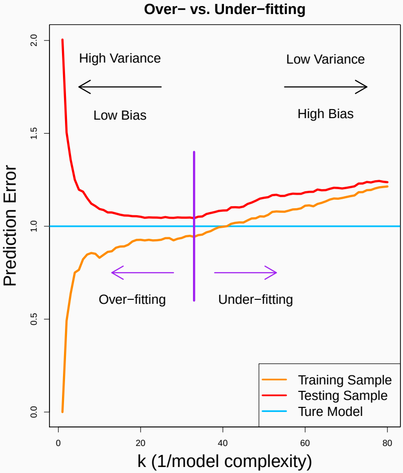
As we further increase k, the model tends to be less complex. Compare 66NN with a linear model that uses only 3 parameters.





Model Complexity, over- and under-fitting

- Model complexity $\uparrow$ (small k) $\longrightarrow$ Bias^2 $\downarrow$ and Variance $\uparrow$
- Model complexity $\downarrow$ (large k) $\longrightarrow$ Bias^2 $\uparrow$ and Variance $\downarrow$



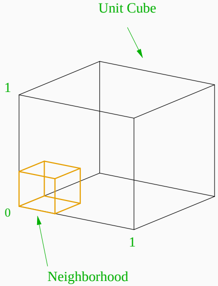
New Challenges

New Challenge

- High-dimension low sample size ($p \gg n$)
  - The resolution of the handwritten digit example is $16 \times 16 = 256$
  - Some common imaging data in medical are 1024×1024 while only a few hundred samples are available
  - Strategy games (Go, StarCraft, DOTA, LOL, etc.) may have a huge number of variables
- Curse of Dimensionality
  - For fixed n, as p increases, the data become sparse
  - As p increases, the number of possible models explodes (computation burden, variable selection necessary)

Curse of Dimensionality

- The curse of dimensionality is well illustrated by a subcubical neighborhood for uniform data in a unit cube.
- Suppose the sample points are evenly spread out on $[0,1]^p$, and we want to capture 10% of the data by constructing a hypercube neighborhood of x. What is the edge length l of this cube? Since the volume of the cube is $l^p = 10\%$, we need $l = 0.1^{1/p}$,
- Read more in `ISL` Ch 2.2.2. Check ESL Video.



- When $p = 1$, l = 0.1
- When $p = 2$, l = 0.32
- When $p = 10$, l = 0.79