Recap: k-Nearest Neighbors, the Curse of Dimensionality, and Naive Bayes

DS701 Session 11 — Tue Oct 13, 2026 (Monday schedule)

Today's plan

Knowledge-Check Review

KC 1: What does k-NN actually do?

*Describe what a k-nearest-neighbors classifier does at **training** time and at **test** time, and name the one hyperparameter that controls how complex the resulting model is. Is k-NN parametric or nonparametric, and why?*

KC 2: What the curse does to "nearest"

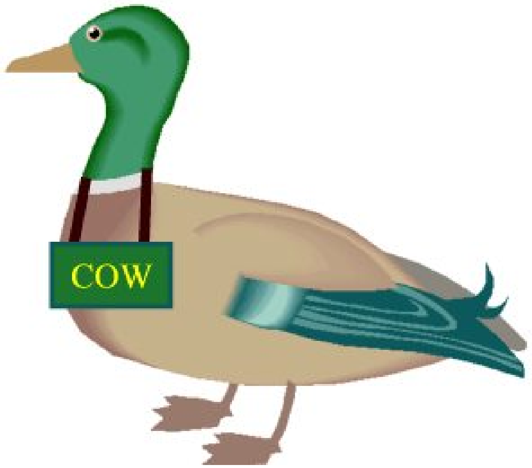
*The lecture showed that in d dimensions the fraction of a unit ball's volume lying in an outer shell of thickness ϵ is $f_d = 1 - (1 - \epsilon)^d$, and that the ratio of the **minimum** to the **average** pairwise distance among random points climbs toward 1 as d grows. Explain in your own words what these two facts do to the word "nearest" in k-nearest neighbors, and why that hurts the classifier.*

KC 3: Naive Bayes meets the curse

State the “naive” assumption in Naive Bayes and say why it is called naive. Then connect it to the curse of dimensionality: a spam filter uses $d = 1{,}000$ binary word features. Roughly how many probabilities per class must Naive Bayes estimate ***with*** *the assumption versus* ***without*** *it, and why might Naive Bayes cope with those 1,000 features better than k-NN would?*

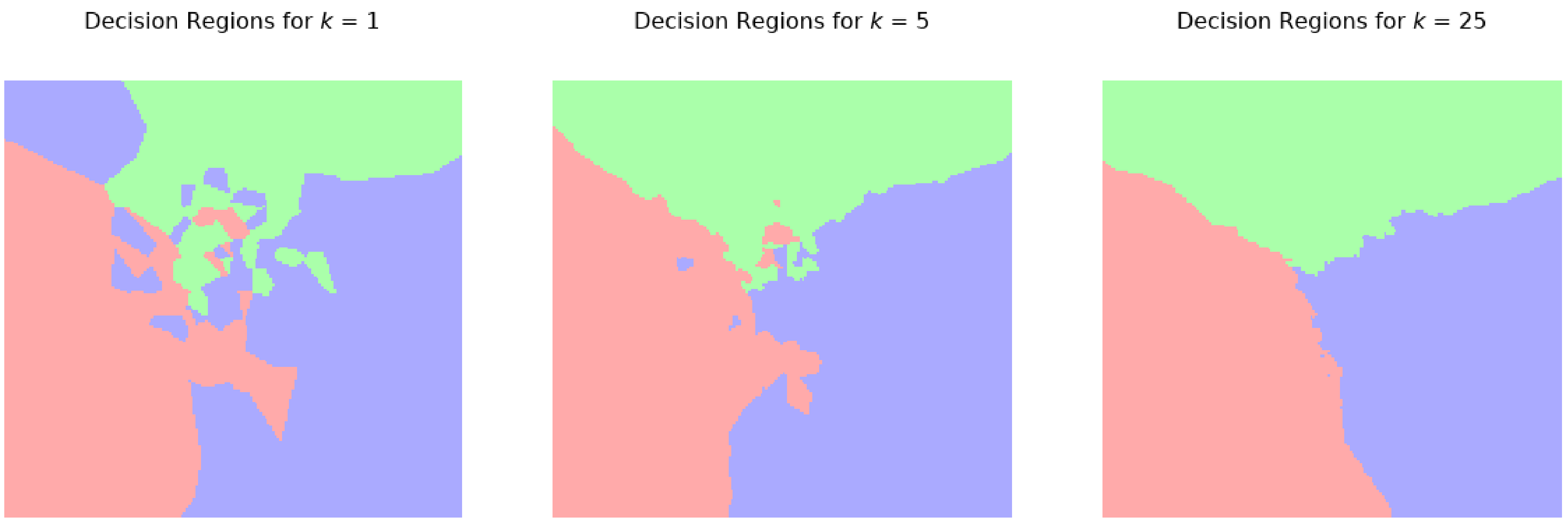
Highlights

The data *is* the model

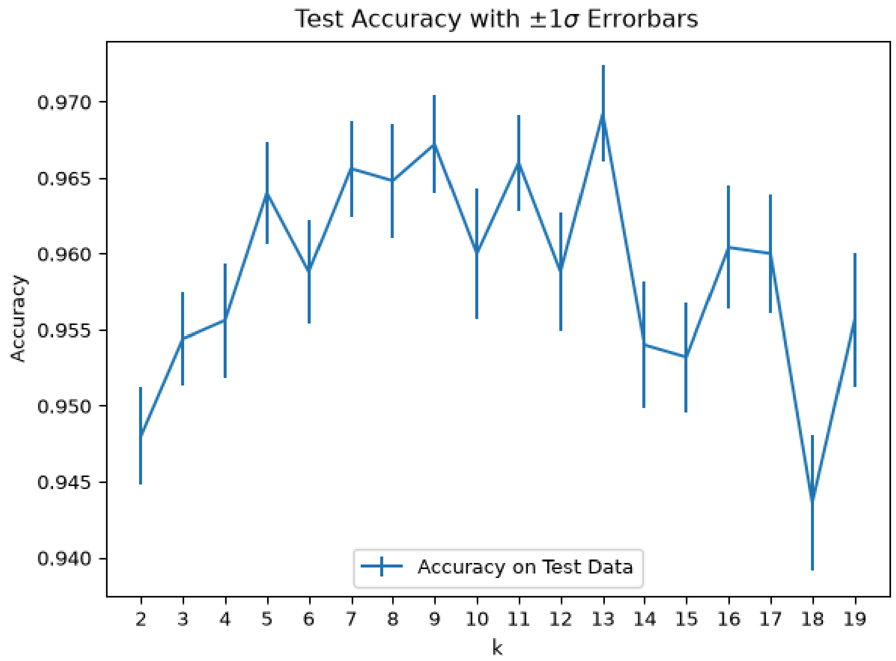


walks like a duck, swims like a duck, quacks like a duck…

k is the bias–variance knob

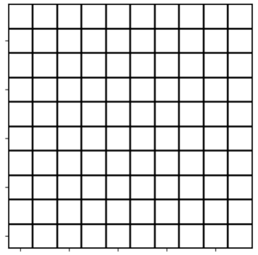


Choosing k — and judging the result

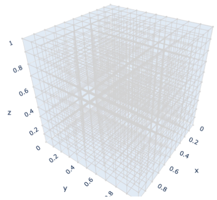


Iris, $k = 2 \ldots 19$, each point the mean of **50 random train/test splits**.

The curse, part 1: space explodes

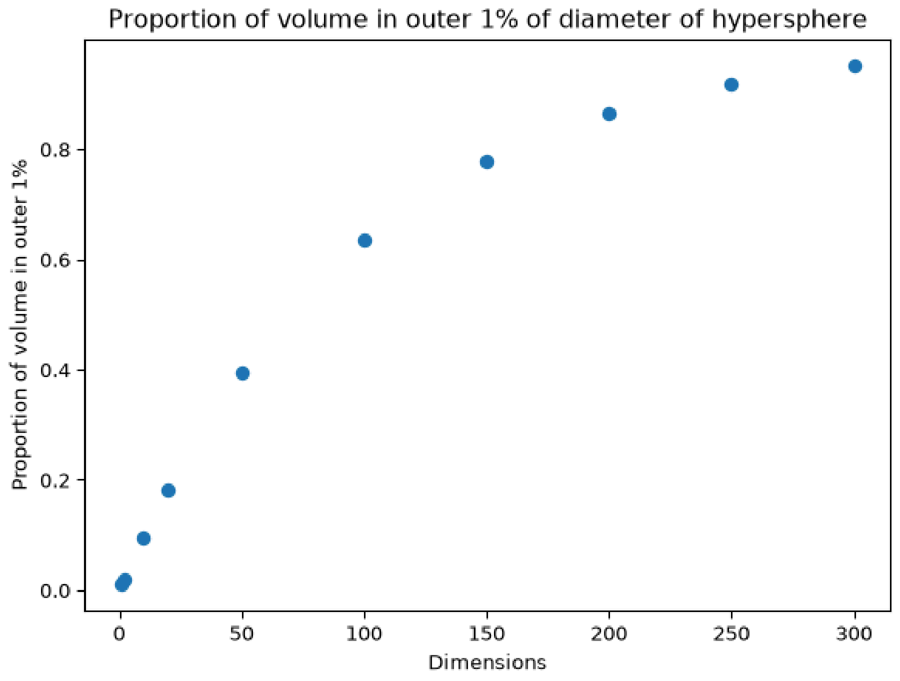


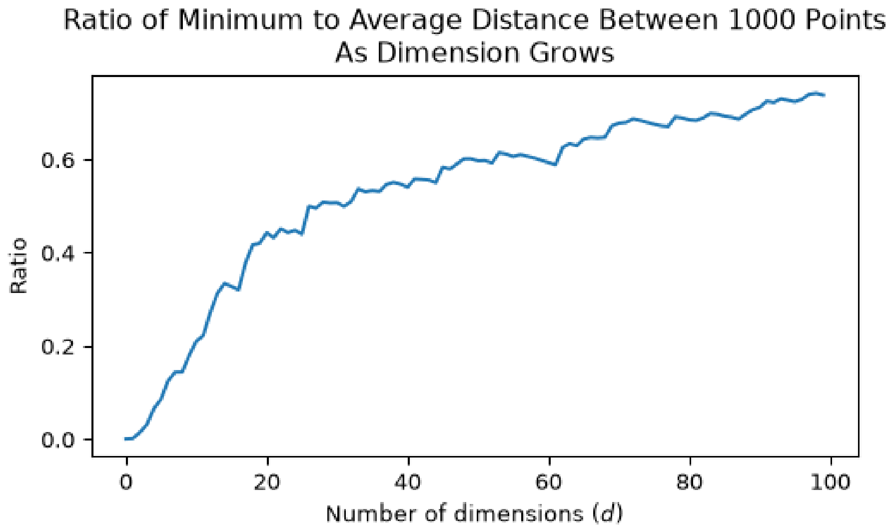
10^2 bins



10^3 bins

The curse, part 2: “nearest” stops meaning anything





1,000 uniform points: **min** pairwise distance / **average** pairwise distance $\to 1$.

Fraction of a unit ball in its outer ϵ-shell: $f_d = 1 - (1 - \epsilon)^d \to 1$.

Naive Bayes: classify with Bayes' rule

$$P(C_i \mid \mathbf{x}) = \frac{P(\mathbf{x} \mid C_i)\, P(C_i)}{P(\mathbf{x})} \qquad \Longrightarrow \qquad \hat{C} = \arg\max_{C_i}\; P(\mathbf{x} \mid C_i)\, P(C_i)$$

The naive assumption — and why it still works

$$P(x_1, \dots, x_d \mid C_i) = \prod_{j=1}^{d} P(x_j \mid C_i)$$

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

In-Class Activity

Activity: a classifier bake-off, then measure the curse

Goal: on your section's slice of a real diagnostic dataset, sweep k under two metrics and choose it by cross-validation, then pit k-NN against Gaussian Naive Bayes and a decision tree — accuracy, confusion matrix, precision, recall — and hand-compute Naive Bayes on a toy table. Then **measure the curse**: nearest/farthest distance ratios and shell volumes as d grows, watch k-NN collapse as noise dimensions are added, and rescue it — and find out whether Naive Bayes suffers the same way.

- Work in groups of 2–3. Open **your section's** notebook.
- Parts marked **(autograded)** are submitted to Gradescope; the rest is participation.
- Staff will circulate — be ready to explain any part of your work.
- Dependencies: `numpy`, `pandas`, `matplotlib`, `scipy`, `scikit-learn`.

Section A1 Open in Colab Open on GitHub

Going deeper

Full lecture notes: k-Nearest Neighbors, Curse of Dimensionality, and Naive Bayes

- Parametric vs. Nonparametric Models
- k-Nearest Neighbors — the 1-, 2-, 3-NN pictures; Varying k — decision regions for $k = 1, 5, 25$
- Challenges for k-NN — cost, scaling, and the curse
- Points are far apart in high D · Fraction of points in outer shell · Curse of Dimensionality Example · Implications of the Curse
- Precision and Recall and the Confusion Matrix on the two-circles data
- Train-Test Splitting and Hyperparameters: k-NN — many splits, error bars, choosing k on iris
- Naive Bayes — Bayes Rule, The “Naive” Assumption, Worked Example
- Where k came from: Generalization — bias–variance, hold-out and K-fold cross-validation