02-620 Week 3

Machine Learning for Scientists

Aidan Jan

January 26, 2026

Classification

The goal of classification is to find a functional mapping $f : X \to Y$, where Y is discrete-valued.

- SNPs for X and disease/healthy status for Y
- Gene expression for X and disease/healthy status for Y
- Pathology images for X and tumor/healthy for Y
- Electronic medical records for X and diagnosis for Y
- Genome sequence features for X and transcription factor binding site or not for Y

Training vs. Testing

- In training, the goal is to improve the model using input data and output pairs.
- In testing, the goal is to classify unseen new input data and provide the output.

Different Types of Classifiers

- K-nearest neighbor
  - Non-parametric method: no model, no parameters, no learning (lazy)
- Naive Bayes
  - Parametric method, generative model: model $P(Y, X|\theta)$ to obtain $P(Y|X, \theta)$
- Logistic Regression
  - Parametric method, discriminative model: model $P(Y|X, \theta)$

K-nearest neighbors (KNN) classifier

- Given N training data points $(x_1, y_1), \cdots, (x_N, y_N)$, kNN performs no explicit learning (i.e., no learnable parameters)
- **Inference:** A new data point x_i, is classified by majority vote among its k-nearest neighbors, defined as the k training points with the smallest Euclidean ($l2$) distances $\|x_{i'} - x_i\|_2^2$

How to select k

- Small k: classification is sensitive to noise
- Large k: too much smoothing. (If $k = N$, sample size, all test inputs will receive the same classification.)
- Select k that is not too small and not too large

Computation Time

- **Learning:** No training or parameter learning - cheap!
- **Inference:** When a new data point $x_{i'}$ arrives, kNN must compute the distance between $x_{i'}$ and all N training samples, incurring an $O(ND)$ computational cost - expensive!

Naive Bayes Classifier

Example: Predicting Cancer from genotype

Individual	Locus 1 X_1	Locus 2 X_2	Locus 3 X_3	Healthy/Cancer Y
1	0	0	1	1
2	1	0	2	1
3	0	2	0	1
4	2	0	0	0
5	2	1	2	0
6	1	2	1	0

Here, the input X represents the allele. 0 = AA (minor allele homozygous), 1 = AT (heterozygous), 2 = TT (major allele homozygous). Y represents healthy (0) or cancer (1). We want to

- learn a classifier, $f : (X_1, X_2, X_3) \rightarrow Y$
- learn a probabilistic model for $P(Y|X)$, where Y is discrete

$P(Y|X)$ is given as

Combination	X_1	X_2	X_3	$P(Y = 1\|X_1, X_2, X_3)$	$P(Y = 0\|X_1, X_2, X_3)$
1	0	0	0	0.01	0.99
2	0	0	1	0.50	0.50
3	0	0	2	0.30	0.70
4	0	1	0	0.25	0.75
5	0	1	1	0.70	0.30
6	0	1	2	0.05	0.95
7	...	...	...	...	...

- How many probability parameters must be specified?
- How can this distribution be learned from data?
- Note that $P(Y = 0|X_1, X_2, X_3) = 1 - P(Y = 1|X_1, X_2, X_3)$

How many parameters are needed?

- Suppose $X = [X_1, \ldots, X_D]$ for D SNPs
  - X_j's: random variables taking values from $\{0, 1, 2\}$
  - Y: binary random variables
- To estimate $P(Y|X_1, X_2, \ldots, X_D)$
- If we have 30 SNPs in X: $P(Y|X_1, X_2, \ldots, X_{30})$