In designing the course, I had the following intent:

- ▶ CSCI 381 has minimal overlap with MATH 465 and CSCI 384—the courses are complementary but not redundant or interdependent.
- ▶ CSCI 381 requires no prior experience in math besides Calc I and linear algebra—any additional math material (probability, partial derivatives) is self contained.
- ▶ CSCI 381 emphasizes the *algorithms* of machine learning—the *mathematical background* and the *applications* are supporting topics.
- ▶ CSCI 381's goal is understanding rather than comprehensiveness—it's better to understand the topics covered than to cover more topics.

This intent was pursued by the following practices:

- ▶ Frequent, short(-ish), closed-ended programming assignments to implement techniques (almost) from scratch.
- ▶ A single, semester-long, open-ended applied project.
- ▶ Frequent, lightweight quizzes to summarize the main points/concepts/terms.
- ▶ A typical four-day pattern for topics consisting of concepts, applications (in lab), mathematical details, algorithmic details.
- ▶ Periodic readings in ethical/social/legal issues throughout the semester, with one week of intense discussion at the end of the semester.

Things I think I got mostly right:

- ▶ General intent, goals, and scope of the course.
- ▶ Selection of topics (KNN, LinReg, LogReg, GMM, SVM, PCA, MLP, CNN).
- ▶ Pacing, rhythm, and workload.
- ▶ Balance among math, algorithms, applications, and ethics.

Things that I know need to be fixed:

- ▶ Textbook.
- ▶ Instructions and guidance for semester project.
- ▶ Notational consistency.
- ▶ Code style consistency and other incidental sources of frustration in the programming assignments.