Hello from the Northeast!

My name is Jacob Reske, and I’m interning with the newly-formed Machine Learning team in the Studio Technology department. In this presentation, I’ll talk about some of the fundamentals of machine learning (ML), a subset of artificial intelligence that concerns itself from creating algorithms that can “learn” from their experience with prior data— often without even telling them *what* to learn. I’ll go a bit into the development of ML as a branch of computational learning, touch on the differences between ML algorithms— supervised and unsupervised— and discuss some of the more popular ways to teach a machine to learn (regression, classification, clustering, and recommendation systems). Along the way, I’ll give some examples of the (literally) hundreds of applications of ML techniques— from analytics to object/semantic recognition— and provide some case studies. Hopefully, by the end you’ll have a grasp on the value and versatility of ML and how it can be useful in a variety of applications.

A bit about me: I’m a Mathematics/Music double major at Yale, and this next year will be my last as an undergraduate, which is already giving me all kinds of feels. In my spare time, I compose electronic music and go biking, but most certainly not at the same time. My academic focus has been on applying a classical Information Theory framework to problems in machine learning. While doing research for a class on Noise (yep, that’s an actual class), I realized that there is very little research in using machine learning techniques on audio signal processing. Semantic audio analysis is still in the early stages, and the applications of a robust classifier could be big for everyone, from casual Spotify listeners to music historiography nerds (like myself). Basically, this all boils down to: “Wouldn’t it be cool if a computer could understand and make music in the same way that we do?” I’m firmly in the camp of “yes, it would be awesome.” We’ll just worry about the whole “singing computer overlords” thing sometime later.

\_JR