Summary
This week, we learned about machine learning, the study of algorithms that improve their performance at a task with more experience. We were introduced to the regression model with the case study of housing prices. specifically how to define the linear or polynomial (higher degree) relationship between the input data and the outcome. Error/cost is always present, one way of measuring that loss is mean squared error, and the best way to find the set of parameters for each feature that minimizes the loss is gradient descent. We also familiarized ourselves with Numpy, Pandas, and Scikit-Learn, which are Python libraries we will use in the ML process.

It's used everywhere, not just generative AI. It was used less obvious contexts such as in political campaigns, real estate pricing, and legal advice. Bias is inherent.

Machine Learning: Improving the performance of a model at a particular task after gaining experience on doing that task

Python tools we will be using
- Numpy array: like a Python list but faster and we can conduct more operations on it easily
- Pandas: used for preprocessing data before training the model. We will use it to analyze, clean, and explore large datasets
- Scikit-learn: used to train machine learning models on datasets in order to predict something

Uncertainties
Conceptual questions:
How exactly does training a model (I'm thinking of a polynomial regression model right now) adjust the weights/parameters for its features to create a model that matches the training data very closely? Is this what was referred to as gradient descent? What does the code look like for gradient descent?

How does preprocessing work for textual data?

This isn't really a question, but I've read that over 20% of the internet is already AI-generated. Once I have sufficient knowledge on coding my own models (so a couple weeks down the line), I want to create my own model and train it on its own output data and see how it affects predictions over time.

Ethics/bias questions:
How can we minimize bias in the training data? Who gets to decide what will be or will not be included in the training data?

Types of learning (decided based on tasks)

Reinforcement learning :
- Agents aim to interact with the environment and collect observations of the responses to yield the max reward

Unsupervised learning:
- Unlabeled dataset
  The ML model finds patterns and categorizes the data

Supervised learning:
- Labeled training data

  A function is created that aims to closely resemble (y hat) the actual function (maps x [input/training data] to y [outputs])

Regression model: mathematical equation that estimates the relationship between the **input data** and the **outcome**

Housing Price case study: predict how much a house is worth based on similar house pricing data in the area

Linear regression:

Loss functions: how well the algorithm models the dataset. Smaller values is better

Mean squared error: sum of the squares of the difference between the predicted and observed values. Our goal is to minimize this number. It is one way of measuring loss.

We can't test all possibilities to find the minimum (**argmin**). However, we can use **gradient descent** to first calculate MSE using random weights and see which way the gradient's direction is down.

equation: y_i=w_o+w_1*x_i+e_i
- w_0 is the bias (with nothing)
- w_1 is the weight (how much it increases per increase in input)
- y hat = w_0 hat + w_1 hat * x
  - The hat values are the predicted values. There is always error that we can't account for

Polynomial regression: linear regression, but with more **features** and **parameters** — **Feature extraction** is when data is turned into features

Model: how we assume the world works — Based on a **predictor**, which is learned from the data (to predict y hat and make it as close to y as possible)
- Low **error** = low **cost** = better predictor (unless it is **overfitted**: model is based on the training data too much and doesn't work on the **test data**. Low variance)

Types of bias
- Historical bias: data might be missing for minority groups due to historical reasons that may not reflect the current reality
- Representation bias: when a category in the training data has more information than the other categories
- Measurement bias: error in measuring the data (random/systematic error while sampling)
- Deployment bias: the model is used for something that it was not intended for

ML Pipeline → World data → Training data → Pre-processing the data → Creating the ML model through training → Optimizing the ML model used → Measure the quality of the model