The course builds on DS110 and during the first part of the semester introduces a number of classification and regression algorithms on top of the popular python packages numpy, pandas, matplotlib and scipy. It then moves to introducing a high performance language (Rust) and how to use it to implement a number of fundamental CS data structures and algorithms (lists, queues, trees, graphs etc), Students are expected to propose and complete an independent project on a large graph dataset using Rust.

Date	Topics Covered
Week 1	Course overview, supervised and unsupervised learning, decision trees.
Week 2	Classification, Regression, Pandas, Interpolation Homework: Markdown and decision trees
Week 3	Clustering, k-means, linear programming, linear regression Homework: Numpy and K-clustering
Week 4	Loss functions, overfitting, underfitting, hyperparameter tuning Homework: Pandas and Linear Programming
Week 5	Programming languages, documentation, source control, basics of Rust. Homework: Rust, overfitting and underfitting
Week 6	Rust: project manager, functions, flow control, arrays, tuples, enums, memory management Homework: Data set research
Week 7	Rust: ownership, borrowing, methods, copying, references, generics and traits. Homework: Basic Math in Rust
Week 8	Rust: Collections, Vectors, Hash Maps, Hash Sets, Graphs Homework: Enums, Structs and Traits in Rust
Week 9	Rust: Graph algorithms, modules and external files. Homework: Generics and methods in Rust
Week 10	Rust: Parsing, stacks and queues, DFS, BFS, Priority queue, Binary heaps. Homework: Simple decision tree in Rust
Week 11	Rust: Sorting, shortest paths, strings, &str, closures and iterators

	Homework: Graph pagerank in Rust
Week 12	Rust: Binary search trees, dynamic programming, greedy algorithms.
Week 13	Rust: Multithreading and parallel programming.

Homework dates are when the homeworks are due (they will be handed out the week before).