Advanced Data Science
CS109b Wrap-Up

Hanspeter Pfister, Mark Glickman,
Verena Kaynig-Fittkau

Project Dates

- Milestone 1
  - Getting to know your data
  - due Wednesday, April 5, 2017
- Milestone 2
  - Assembling training data
  - due Wednesday, April 12, 2017
- Milestone 3
  - Traditional statistical and machine learning methods
  - due Wednesday, April 19, 2017
- Milestone 4
  - Deep learning
  - **due Wednesday, April 26, 2017** <=
- Milestone 5
  - Final submission, report and screencast
  - **due Wednesday, May 3, 2017**

Video

- The video is a high-level summary of your project for a data science audience
- Do not just scroll through your notebook!
- Think about what you learned about story telling
- Do not try to convey too much information

Some Examples

- Predicting Hubway availability, L. Alexander, G. Goulet-Langlois, J. Wolff https://www.youtube.com/watch?v=2wK8jpNMjXI&feature=youtu.be

- AirBnB success, H. Husain, Y. Mao, L. Awad, X. Li https://www.youtube.com/watch?v=raGjUj5qArc

- The green canvas, A. Hosny, J. Huang , Y. Wang https://vimeo.com/114379373

Report

- 6 pages, no appendix, no code
- Concise summary of your project, the decisions you made, how they turned out
- Paper style (Introduction, Methods, Results, Discussion & Conclusion)
- Written for a data scientist audience:
  - Do not describe what PCA or a SVM is
  - Do describe why you chose PCA or SVM for your task
- We are looking for insight. Show that you understand what you are doing.

Example

Bad: We used a SVM to solve our prediction task. We chose RBF kernel with gamma=0.1 and C=1.0.

Good: We have a high dimensional data set, and therefore choose a SVM, because the maximum margin helps with the curse of dimensionality. RBF is the standard choice, but our problem is so high dimensional that we also tried a linear kernel to reduce the degrees of freedom in the model.

What you learned in CS109a/b

Data Wrangling

- Python
- Pandas
- Spark
- Map Reduce

CS164 – Software Engineering

- David Malan

- Introduction to principles of software engineering and best practices, including code reviews, source control, and unit tests. Topics include Ajax, event handling, HTTP, MVC, object-oriented design, relational databases, and user experience. Projects include web apps with front-end UIs (mobile and desktop) and back-end APIs. Languages include JavaScript and PHP.

CS124 -Data Structures and Algorithms

- Fundamentals
- Graph Algorithms
- Greedy Algorithms
- Dynamic Programming
- Divide and Conquer
- Hashing
- Linear Programming
- Randomized Algorithms
- NP-completeness review
- Novel approaches to NP-complete problems

CS165 – Data Systems

- **Expected learning outcomes**
- Become familiar with the history and evolution of data systems design over the past 4-5 decades.
- Understanding the basic tradeoffs in designing and implementing modern data systems.
- Being able to design a new data system given a data-driven scenario and to built a prototype.
- Being able to understand which data system is a good fit given the needs of an application.
- Advanced C programming and debugging skills.

CS207- Systems Development for Computational Science

- Apply basic software development tool-chains, including source-code control, testing frameworks, and documentation tools, to the process of designing and implementing large software systems;
- Apply design principles to the decomposition of software into re-usable components, and to the production of those components;
- Know how to approach an existing piece of software for maintenance, extension, and modification;
- Design, develop, and deploy a set of software components to produce a scalable, reliable, and reproducible experimental system for scientific investigation;
- Use a variety of approaches to software development team organization, and select techniques that are appropriate in different circumstances.

CS205 – Computing Foundations for Computational Science

- Apply basic computer science concepts such as modularity, abstraction, and encapsulation to scientific problems
- Recognize and recall computer architectures, algorithms, and data structures that are relevant to computational science
- Apply concepts of parallel programming and “parallel thinking” to computational science
- Analyze and visualize large scientific data and implement data-intensive computations on cluster and cloud infrastructures
- Use open-source tools for large- and fine-grain parallel computations, cloud computing, and visualization

CS262 - Introduction to Distributed Computing

- design and implementation of large systems
- running on multiple computers connected by a network.
- investigate the fundamental characteristics of distributed systems
- investigate how to build systems that exploit those fundamental characteristics.

Visualization and Storytelling

- EDA
- Effective visualization
- Interactive visualization
- Story telling
- Ethics & Privacy

CS171-Visualization

- **After successful completion of this course, you will be able to…**
- Critically evaluate visualizations and suggest improvements and refinements
- Use JavaScript and other tools to scrape, clean, and process data
- Use standalone visualization applications to quickly explore data
- Apply a structured design process to create effective visualizations
- Conceptualize ideas and interaction techniques using sketching
- Use principles of human perception and cognition in visualization design
- Create web-based interactive visualizations using JavaScript and D3
- Use storytelling principles to design coherent and clear visualizations

CS179 - Design of Usable Interactive Systems

- **Discover**. The first question in Design is "What problem should we solve?". Finding a good problem, particularly one that is important to people other than yourself, is hard because many big problems are so ingrained in people's lives that they no longer notice them. You will learn the techniques for conducting systematic observations and for analyzing the data from those observations. These techniques will help you identify valuable design opportunities.
- **Invent**. Creativity is about finding solutions that are novel, surprising and valuable. Creativity is a skill that can be learned and practiced. You will learn about the cognitive underpinnings of creativity and you will learn several techniques for managing your creative process.
- **Design**. Once you have an idea for a useful product, you will need to design it to be usable. You will learn the basics of human perception, motor performance, and cognition, as well as a number of design principles, that will help you generate designs for usable interactive systems.
- **Prototype**. Once you have a design, you will want to turn it into something that people can use. We will cover a range of prototyping techniques all the way from paper prototypes (which take just a couple of hours to build) to implementing interactive mobile web applications.
- **Evaluate**. Your first design will never be perfect. We will share with you several techniques for evaluating your designs with real people.
- **Communicate**. As a designer, you need to communicate your findings and ideas to other designers, to clients, to funders, etc. Effective communication will be a big part of your professional success. Besides preparing weekly written reports, you will have at least three opportunities during the course to pitch a product concept to external evaluators (designers, entrepreneurs, etc) and to receive feedback.
- **Succeed on a team**. Working on a team is hard: your life depends on several other people, yet you have no authority to tell them what to do. So what can you do as a team member to help make your team happy and successful? You will learn a few techniques that successful teams use to manage communication and conflict.

CS108 - Intelligent Systems: Design and Ethical Challenges

CS108 provides a broad introduction to Artificial Intelligence (AI), situated in the context of AI's current and potential future uses and the design and ethical challenges they raise. It aims to provide students with a basic understanding of how AI technologies work, their strengths and weaknesses, and to enable them to distinguish fact from fiction in discussions of AI systems and their potential societal impacts. It examines ethical dimensions of AI systems' capabilities, considering both anticipated benefits and potential negative impacts, along with the roles of system design and societal policies as ways to address potential negatives.

CS105 – Privacy and Technology (GOV1430)

- Privacy Concepts
- Surveillance
- Tapping and tracing
- Data aggregation, analytics, and privacy
- Data-intensive Science
- Biometrics and DNA

Statistical topics

- Probability review
- Statistical models
- Linear regression
- Logistic regression
- Regularization (ridge and Lasso)
- Principal components
- Discriminant analysis
- Experimental design

STAT 110 & STAT 210

A comprehensive introduction to probability

- sample spaces and events
- conditional probability
- Bayes' Theorem
- Univariate distributions
- Multivariate distributions
- Limit laws
- Markov chains

STAT 111 & STAT 211

Basic concepts of statistical inference from Frequentist and Bayesian perspectives.

- Maximum likelihood methods
- Confidence and Bayesian interval estimation
- Hypothesis testing
- Least squares methods
- Categorical data analysis

STAT 139 & STAT 149

Linear Models

- t-based inference
- Permutation-based alternatives
- Multiple-group comparisons
- Analysis of variance,
- Linear regression
- Model checking and refinement
- Causation versus correlation.

Generalized Linear Models

- Exponential dispersion models
- Binary response models
- Log-linear models
- Multinomial logit models
- Over-dispersed models
- Ordinal response models
- Generalized additive models
- Trees & Random Forests

Other relevant Statistics courses:

- Stat 131 – Time series and prediction
- Stat 140 – Design of experiments
- Stat 160 – Design and analysis of samples
- Stat 183 – Learning from big data
- Stat 186 – Causal inference
- Stat 230 – Multivariate statistical analysis

Machine Learning

- Supervised learning:
  - K-nearest neighbors
  - SVM
  - Classification and Regression Trees
  - Random Forest
  - Neural nets & Deep Learning
- Unsupervised learning:
  - Dimensionality reduction
  - Cluster analysis

CS181 – Machine Learning

- Clustering with K-Means and K-Medoids
- Hierarchical Clustering
- Principal Component Analysis
- Supervised Learning
- Linear Regression
- Model Selection
- Linear Classification
- Classification and CV Review
- Probabilistic Classification
- Neural Networks
- Regression and Classification Trees
- Max-Margin Classification
- Support Vector Machines
- Markov Decision Processes
- Reinforcement Learning
- Partially-Observable Markov Decision Processes
- Expectation Maximization
- Hidden Markov Models

CS281 – Advanced Machine Learning

- Introduction to Inference and Learning
- Simple Discrete Models
- Simple Gaussian Models
- Bayesian Statistics
- Linear Regression
- Linear Classifiers
- Generalized Linear Models
- Directed Graphical Models
- Mixture Models
- Sparse Linear Models
- Exact Inference
- Variational Inference
- Loopy Belief Propagation
- Monte Carlo Basics
- Markov Chain Monte Carlo
- Advanced MCMC
- Latent Dirichlet Allocation
- State Space Models
- Graph Models
- Kernels
- Gaussian Processes
- Dirichlet Processes
- Boltzmann Machines
- Neural Networks

Bayesian Thinking

- Bayes vs Frequentist inference
- Bayes rule
- Prior distributions
- Likelihood
- Posterior distributions
- Naïve Bayes, Dirichlet/Multinomial modeling
- Hierarchical linear models

STAT 120 & STAT 220

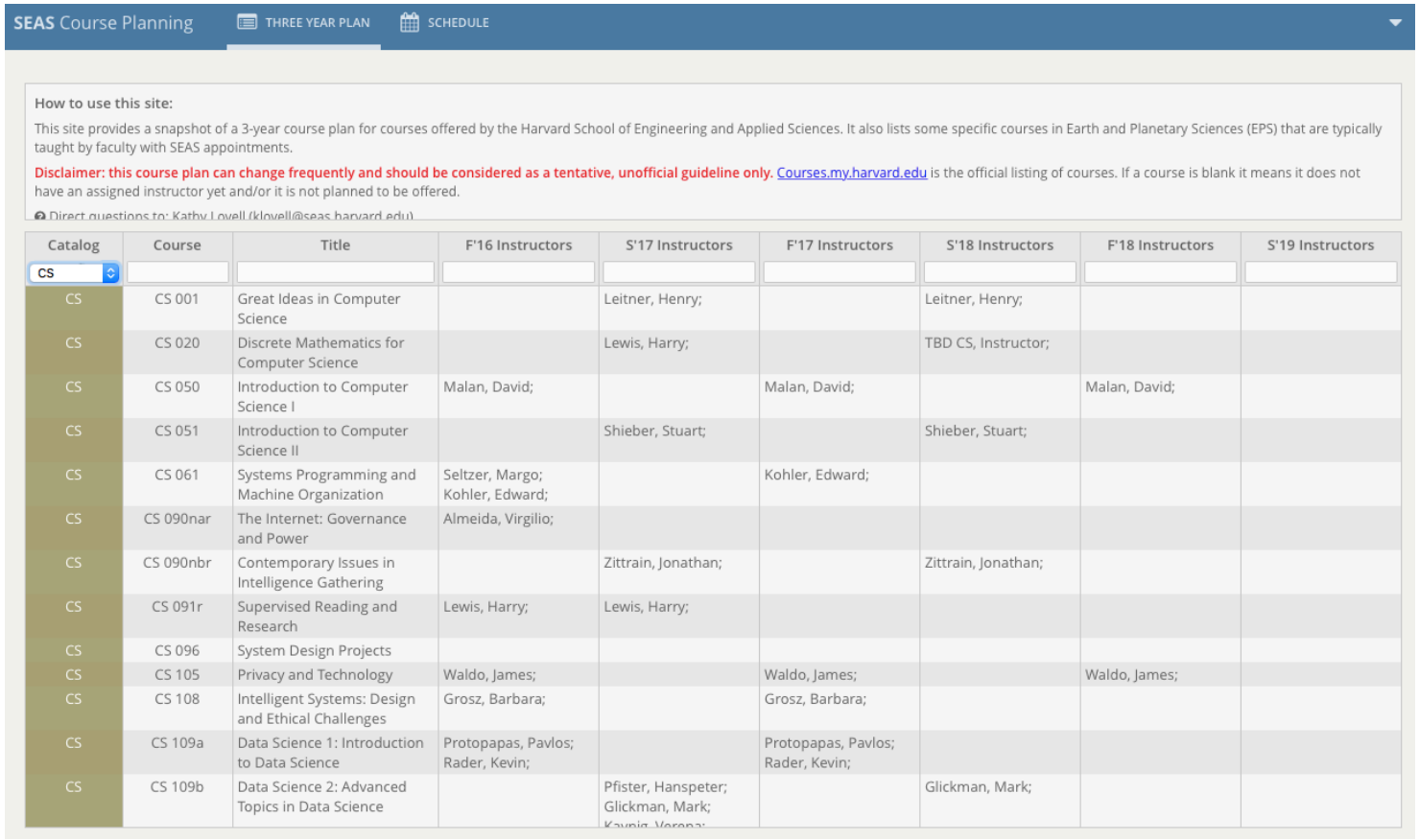
- Basics of the Bayesian inference
- Multi-parameter models
- Inference from large samples
- Hierarchical models
- Non-iterative computational methods
- Missing data
- Iterative sampling methods
- Model checking
- Topics on Bayesian modeling and computation

AM207 - Monte Carlo Methods for Inference and Data Analysis

- Introduction to basic Monte Carlo methods
- Bayes formalism and sampling
- Stochastic Optimization
- Dynamic systems
- Advanced sampling methods
- Graphical Models

SEAS 3-year Schedule

https://info.seas.harvard.edu/courses/#/threeYearPlan



SEAS Course Planning | THREE YEAR PLAN | SCHEDULE

How to use this site:

This site provides a snapshot of a 3-year course plan for courses offered by the Harvard School of Engineering and Applied Sciences. It also lists some specific courses in Earth and Planetary Sciences (EPS) that are typically taught by faculty with SEAS appointments.

Disclaimer: this course plan can change frequently and should be considered as a tentative, unofficial guideline only. Courses.my.harvard.edu is the official listing of courses. If a course is blank it means it does not have an assigned instructor yet and/or it is not planned to be offered.

Direct questions to: Kathy Lovell (klovell@seas.harvard.edu)

Catalog	Course	Title	F'16 Instructors	S'17 Instructors	F'17 Instructors	S'18 Instructors	F'18 Instructors	S'19 Instructors
CS								
CS	CS 001	Great Ideas in Computer Science		Leitner, Henry;		Leitner, Henry;		
CS	CS 020	Discrete Mathematics for Computer Science		Lewis, Harry;		TBD CS, Instructor;		
CS	CS 050	Introduction to Computer Science I	Malan, David;		Malan, David;		Malan, David;	
CS	CS 051	Introduction to Computer Science II		Shieber, Stuart;		Shieber, Stuart;		
CS	CS 061	Systems Programming and Machine Organization	Seltzer, Margo; Kohler, Edward;		Kohler, Edward;			
CS	CS 090nar	The Internet: Governance and Power	Almeida, Virgilio;					
CS	CS 090nbr	Contemporary Issues in Intelligence Gathering		Zittrain, Jonathan;		Zittrain, Jonathan;		
CS	CS 091r	Supervised Reading and Research	Lewis, Harry;	Lewis, Harry;				
CS	CS 096	System Design Projects						
CS	CS 105	Privacy and Technology	Waldo, James;		Waldo, James;		Waldo, James;	
CS	CS 108	Intelligent Systems: Design and Ethical Challenges	Grosz, Barbara;		Grosz, Barbara;			
CS	CS 109a	Data Science 1: Introduction to Data Science	Protopapas, Pavlos; Rader, Kevin;		Protopapas, Pavlos; Rader, Kevin;			
CS	CS 109b	Data Science 2: Advanced Topics in Data Science		Pfister, Hanspeter; Glickman, Mark; Kaynig, Verena;		Glickman, Mark;		

Other Data Science Curricula

Online Courses

- Deep Learning
  - Stanford CS231n – CNNs for Visual Recognition
    - » http://cs231n.stanford.edu/
  - Stanford CS224d – Deep Learning for NLP
    - » http://cs224n.stanford.edu/
- Coursera / Geoff Hinton / Neural Networks
    - » Starts May 15
    - » https://www.coursera.org/learn/neural-networks

Institute for Applied Computational Science

iacs.seas.harvard.edu

- **Academic program**: courses, degree programs, summer undergraduate research program, industry student research collaboration
- **Biweekly seminar series**
- **International Programs**:
  - Harvard/Politecnico di Milano Capstone Project
  - Harvard/University of Chile Computational Project
- **Compute Fest**: workshops, student challenge, symposium

Computational Science and Engineering SM

Master's of Science: 1 year degree

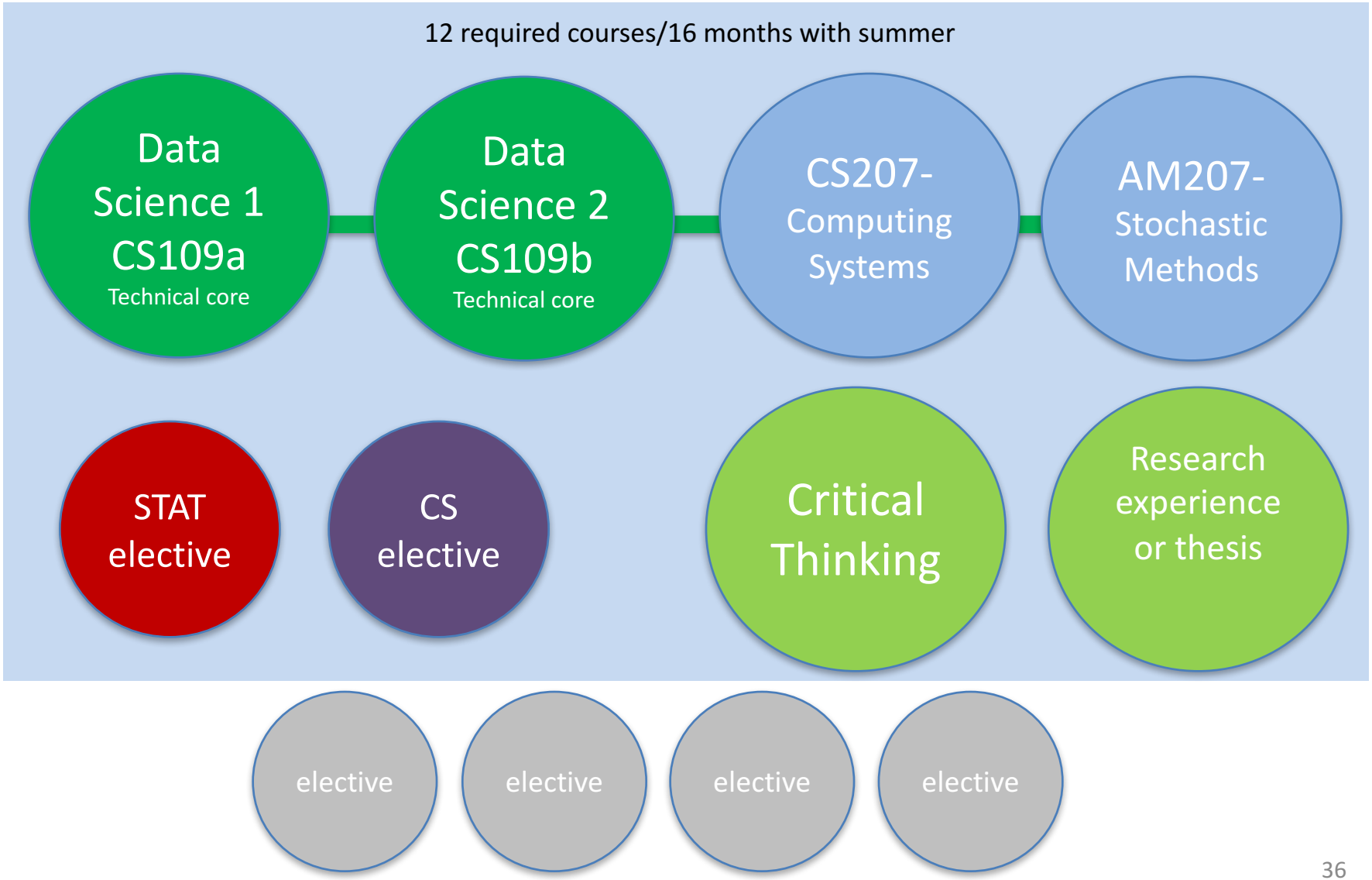
Master's of Engineering: 2 years with thesis

Academic Program:

Core courses: Numerical Methods (AM205), Stochastic Methods (AM207), Computing Foundations (CS205), and Systems Development (CS207)

Key electives: Data Science (1 and 2), Capstone research project course, computer science and statistics courses

Data Science SM (start 2018)



Data Science SM course schedule

Fall 1	Spring 1	Fall 2
Data Science 1	Data Science 2	Capstone
CS 207	AM 207	
Critical Thinking		
sample electives:		
STAT 139	CS 181	CS 281
STAT 131	STAT 149	AC 290
CS 171	CS 205	
	AC 298r	

Statistics Electives

(minimum 1, maximum 5)

- Stat131, Time series and prediction
- Stat139, Statistical sleuthing through linear models
- Stat140, Design of experiments
- Stat149, Generalized linear models
- Stat160, Design and analysis of sample surveys
- Stat171, Introduction to stochastic processes
- Stat186, Causal inference
- Stat210, Probability I
- Stat212, Probability II
- Stat211, Statistical inference I
- Stat213, Statistical inference II
- Stat220, Bayesian data analysis
- Stat221, Statistical computing and learning
- Stat225, Spatial statistics
- Stat230, Multivariate statistical analysis
- Stat232r, Topics in missing data
- Stat240, Matched sampling and study design
- Stat244, Linear and generalized linear models

Computer Science Electives

(minimum 1, maximum 5)

- CS51: Introduction to computer science II
- CS105, Privacy and technology
- CS124: Data structures and algorithms
- CS125, Algorithms and complexity
- CS134, Networks
- CS165, Data systems
- CS171, Visualization
- CS181, Machine learning
- CS187, Computational linguistics
- CS205, Computational foundations of computational science
- CS281, Computability and complexity
- CS222, Algorithms at the ends of the wire
- CS224, Advanced algorithms
- CS265, Big data systems
- CS281 Advanced machine learning

Applied Mathematics Electives

- AM205 Advanced Scientific Computing: Numerical Methods
- AM120, Applicable linear algebra
- AM121, Introduction to optimization: models and methods
- AM107, Graph theory and combinatorics
- AM221, Advanced optimization

Learning Outcomes

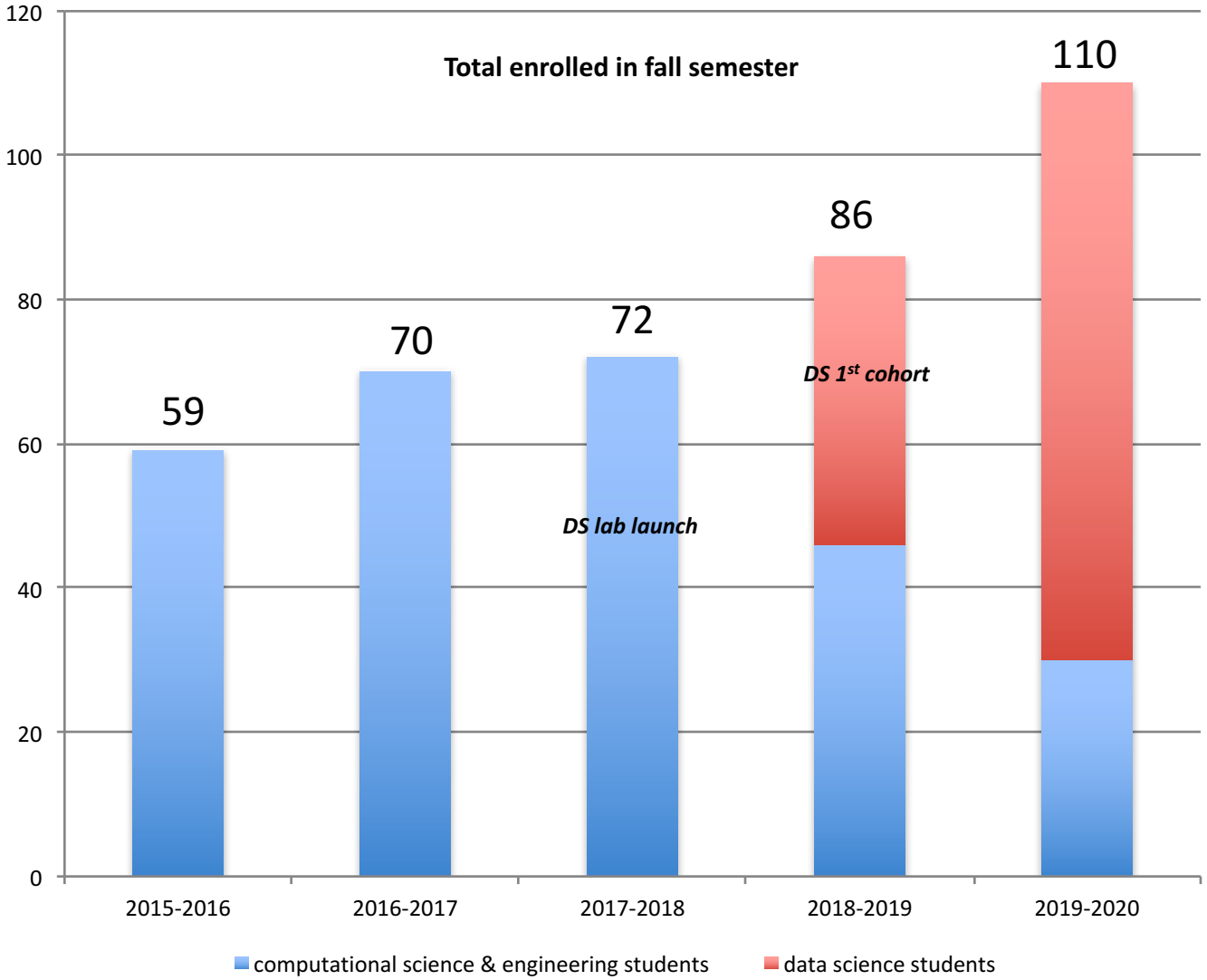
A graduate of the Harvard SM in Data Science program should be able to:

- Build statistical models and understand their power and limitations
- Design an experiment
- Use machine learning and optimization to make decisions
- Acquire, clean and manage data
- Visualize data for exploration, analysis, and communication
- Collaborate within teams
- Deliver reproducible data analysis
- Manage and analyze massive data sets
- Assemble computational pipelines to support data science from widely available tools
- Conduct data science activities aware of and according to policy, privacy, security and ethical considerations
- Apply problem-solving strategies to open-ended questions

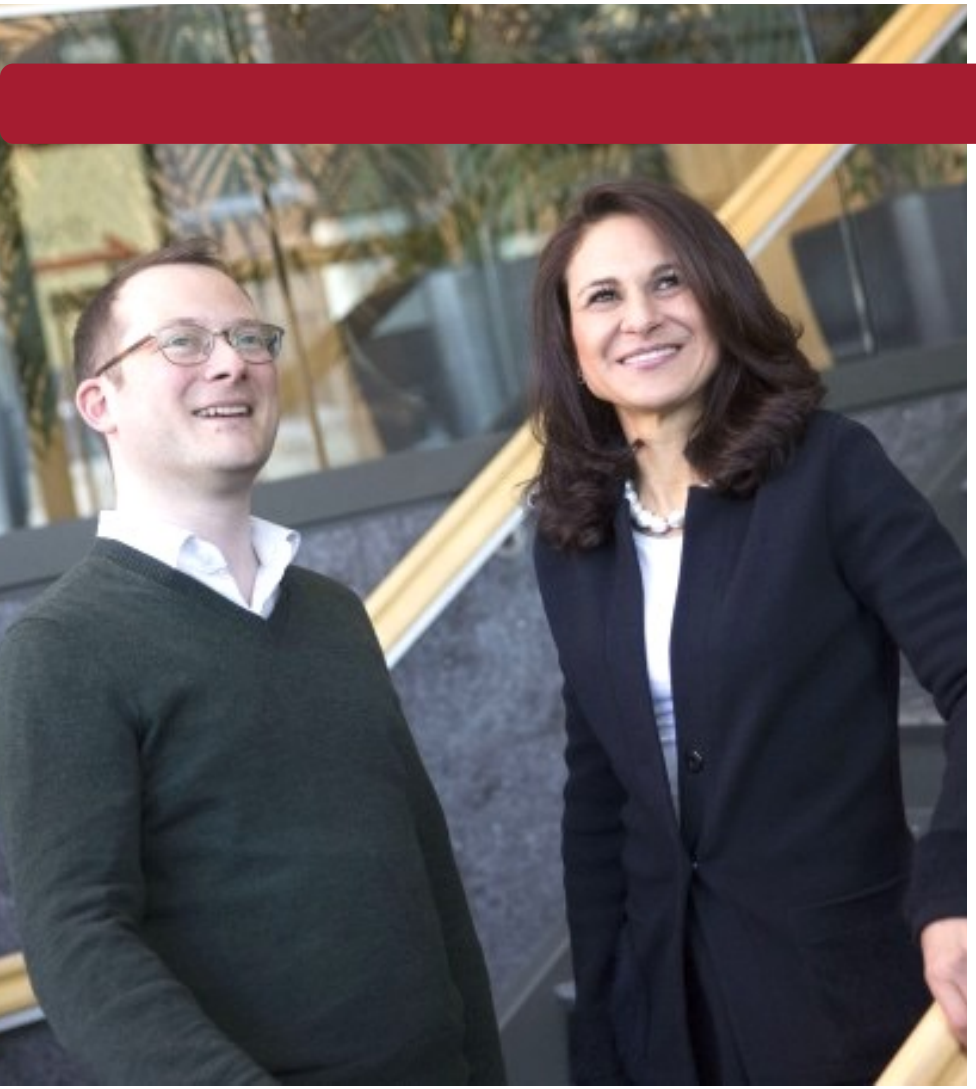
Data Science Learning Outcomes and Courses

Outcome	DS I CS109a	DS II CS109b	AM 207	CS 207	Research Experience	Critical Thinking
Build Stat models	X	X	X			
Design Experiment	X	X				
Machine Learning & Optimization	X	X	X			
Acquire, clean & manage data	X	X	X		X	
Visualize Data for Understanding	X	X			X	
Collaborate within teams	X	X	X	X	X	
Reproducible Data Analysis	X	X	X	X		
Computational Infrastructure				X		
Handle large data	X	X				
Ethics						X
Problem Solving					X	

Projected Levels of Master's Students (under IACS administration)




HARVARD
DATA
SCIENCE
INITIATIVE



CO-DIRECTORS

FRANCESCA DOMINICI

Professor of Biostatistics
Harvard T.H. Chan School of Public Health

DAVID PARKES

Harvard College Professor
George F. Colony Professor of Computer Science
Area Dean for Computer Science
Paulson School of Engineering & Applied Sciences

DATA SCIENCE @ HARVARD

Harvard not only has a **strong capacity to lead in this field**, but also to create fundamental change in the way we draw insights from data for the world at large.

Harvard's future success within and across the **expansive range of academic and professional disciplines** is directly linked to what we can achieve in data science globally.

- Recommendations of the June 2016 committee:

1. Establish a Data Science Institute – a physical home for data science
2. Develop data science educational programs
3. Establish a career path for non-faculty data scientists
4. Invest in shared research platforms across the University

2 NEW DATA SCIENCE RESEARCH LABS + IQSS

2 MILLION IN STARTUP FUNDING FROM THE PROVOST

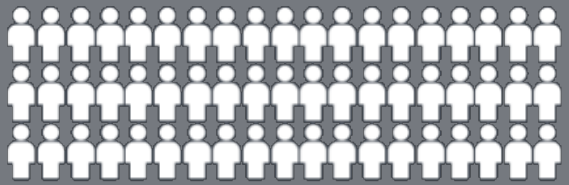
26+ GRANT APPLICATIONS SUBMITTED

8 DATA SCIENCE POSTDOC FELLOWS

3 MASTERS PROGRAMS

1 GIFT

75+ FACULTY ENGAGED



8 HARVARD SCHOOLS

WWW.DATASCIENCE.HARVARD.EDU