Machine Learning Teams

Josh Tobin, Sergey Karayev, Pieter Abbeel

Running ML teams is hard

Running any technical team is hard…

- Hiring great people
- Managing and developing those people
- Managing your team's output and making sure your vectors are aligned
- Making good long-term technical choices & managing technical debt
- Managing expectations from leadership

… And ML adds complexity

- ML talent is expensive and scarce
- ML teams have a diverse set of roles
- Projects have unclear timelines and high uncertainty
- The field is moving fast and ML is the "high-interest credit card of technical debt"
- Leadership often doesn't understand AI

Goal of this module

- Give you some insight into how to think about building and managing ML teams
- Help you get a job in ML

Module overview

- **Roles** — ML-related roles and the skills they require
- **Orgs** — How ML teams are organized and how they fit into the broader organization
- **Managing** — How to manage a ML team
- **Hiring** — How to hire ML engineers. How to get hired. Exam for the course

Module overview

Roles

- **ML-related roles and the skills they require**

Orgs

- How ML teams are organized and how they fit into the broader organization

Managing

- How to manage a ML team

Hiring

- How to hire ML engineers. How to get hired. Exam for the course

Most common ML roles

- ML product manager
- DevOps
- Data engineer
- ML engineer
- ML researcher / ML scientist
- Data scientist

Most common ML roles

- ML product manager
- DevOps
- Data engineer
- ML engineer
- ML researcher / ML scientist
- Data scientist

What's the difference?

Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc

Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc
DevOps engineer	Deploy & monitor production systems	Deployed product	AWS, etc.

Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc
DevOps engineer	Deploy & monitor production systems	Deployed product	AWS, etc.
Data engineer	Build data pipelines, aggregation, storage, monitoring	Distributed system	Hadoop, Kafka, Airflow

Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc
DevOps engineer	Deploy & monitor production systems	Deployed product	AWS, etc.
Data engineer	Build data pipelines, aggregation, storage, monitoring	Distributed system	Hadoop, Kafka, Airflow
ML engineer	Train & deploy prediction models	Prediction system running on real data (often in production)	Tensorflow, Docker

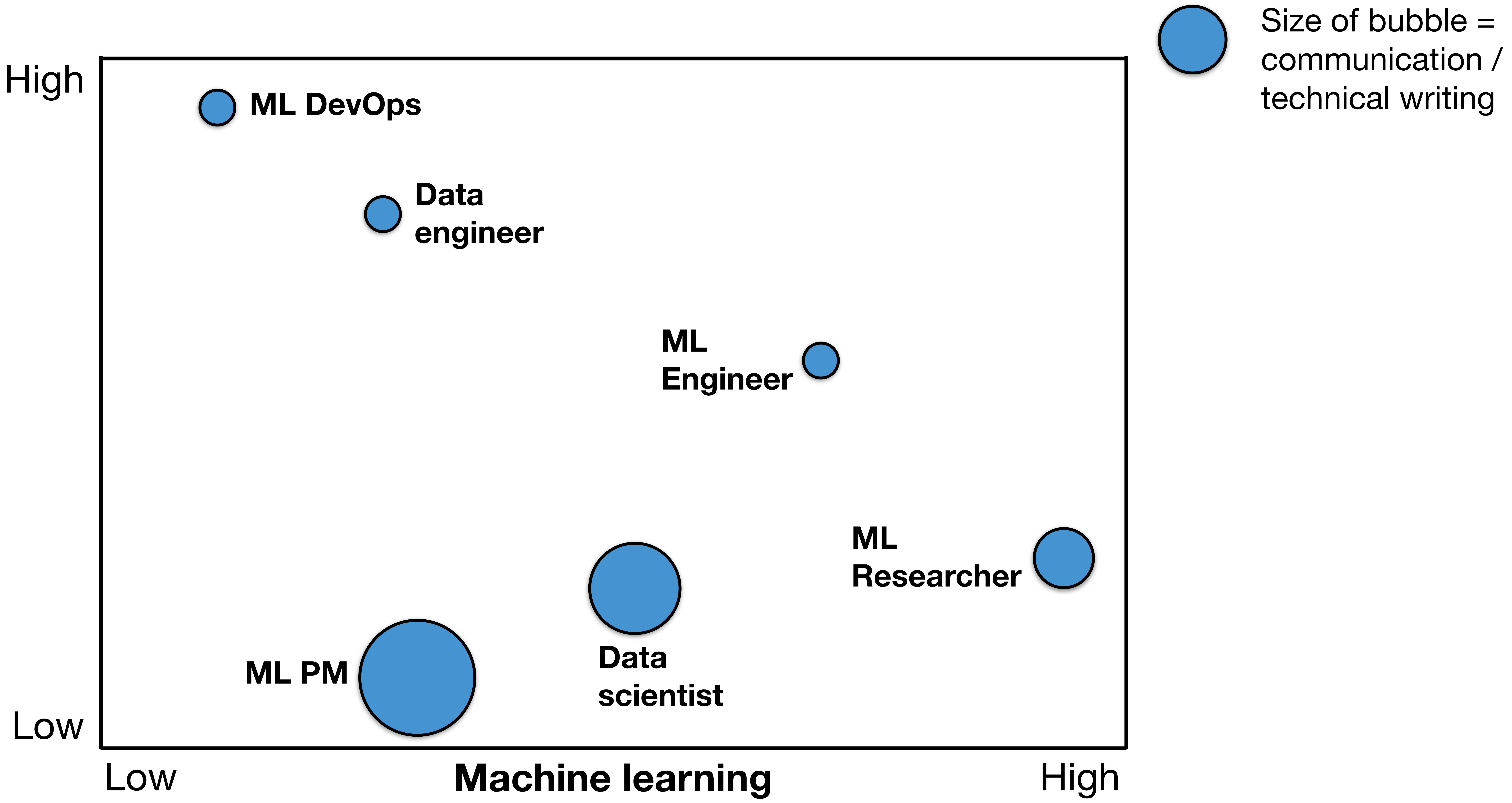
Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc
DevOps engineer	Deploy & monitor production systems	Deployed product	AWS, etc.
Data engineer	Build data pipelines, aggregation, storage, monitoring	Distributed system	Hadoop, Kafka, Airflow
ML engineer	Train & deploy prediction models	Prediction system running on real data (often in production)	Tensorflow, Docker
ML researcher	Train prediction models (often forward looking or not production-critical)	Prediction model & report describing it	Tensorflow, pytorch, Jupyter

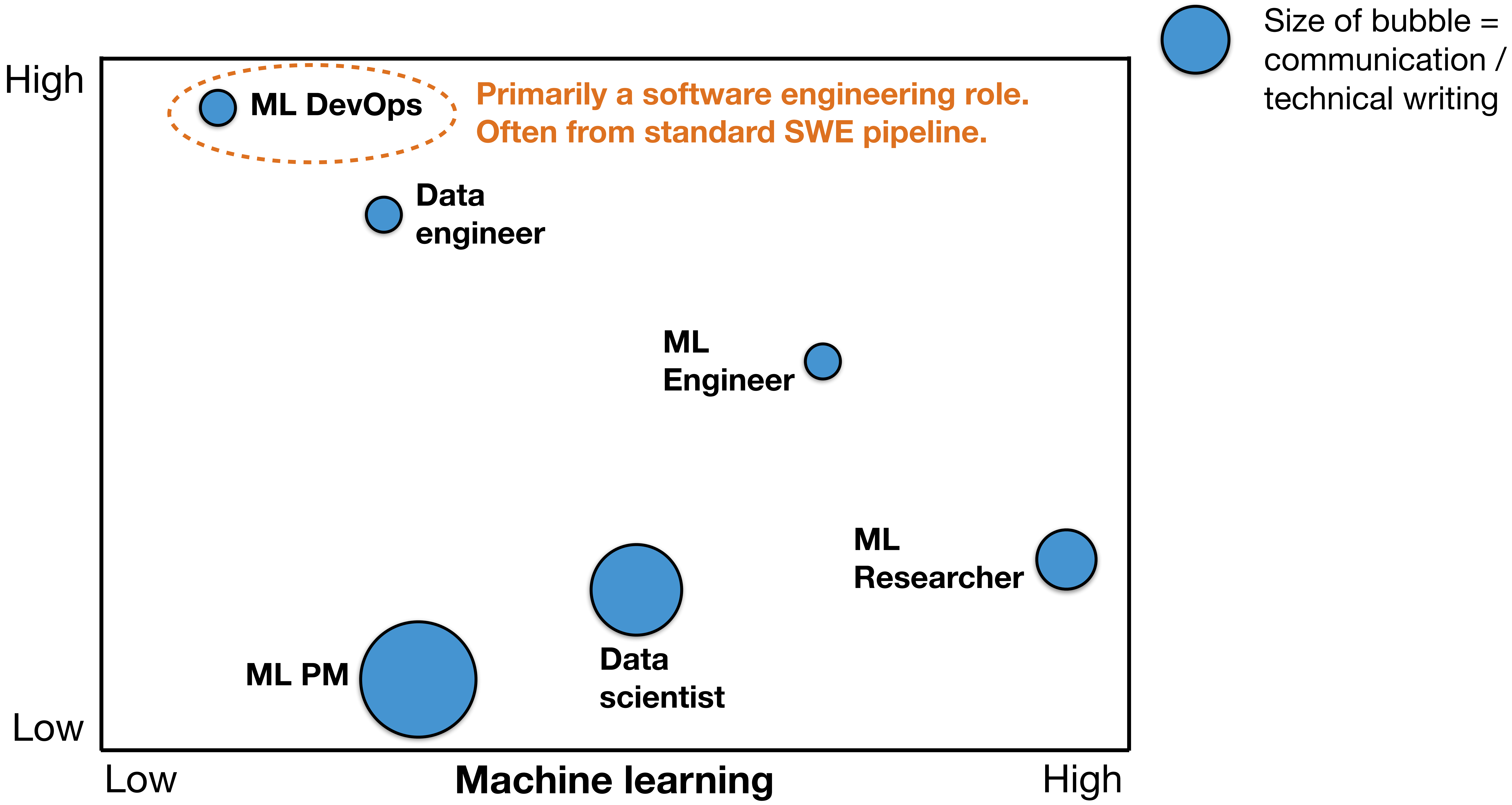
Breakdown of job function by role

Role	Job Function	Work product	Commonly used tools
ML product manager	Work with ML team, business, users, data owners to prioritize & execute projects	Design docs, wireframes, work plans	Jira, etc
DevOps engineer	Deploy & monitor production systems	Deployed product	AWS, etc.
Data engineer	Build data pipelines, aggregation, storage, monitoring	Distributed system	Hadoop, Kafka, Airflow
ML engineer	Train & deploy prediction models	Prediction system running on real data (often in production)	Tensorflow, Docker
ML researcher	Train prediction models (often forward looking or not production-critical)	Prediction model & report describing it	Tensorflow, pytorch, Jupyter
Data scientist	Blanket term used to describe all of the above. In some orgs, means answering business questions using analytics	Prediction model or report	SQL, Excel, Jupyter, Pandas, SKLearn, Tensorflow

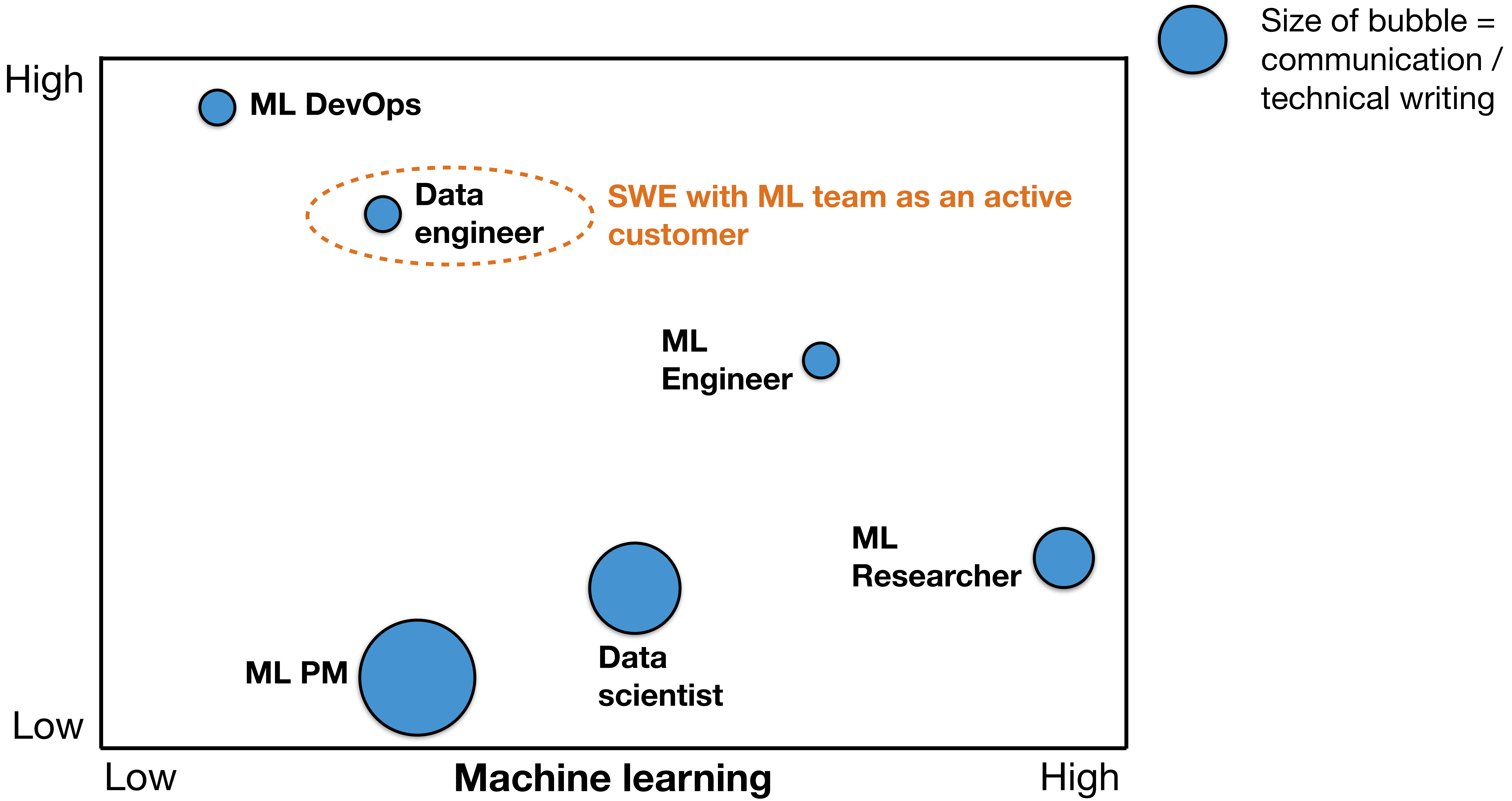
What skills are needed for the roles?



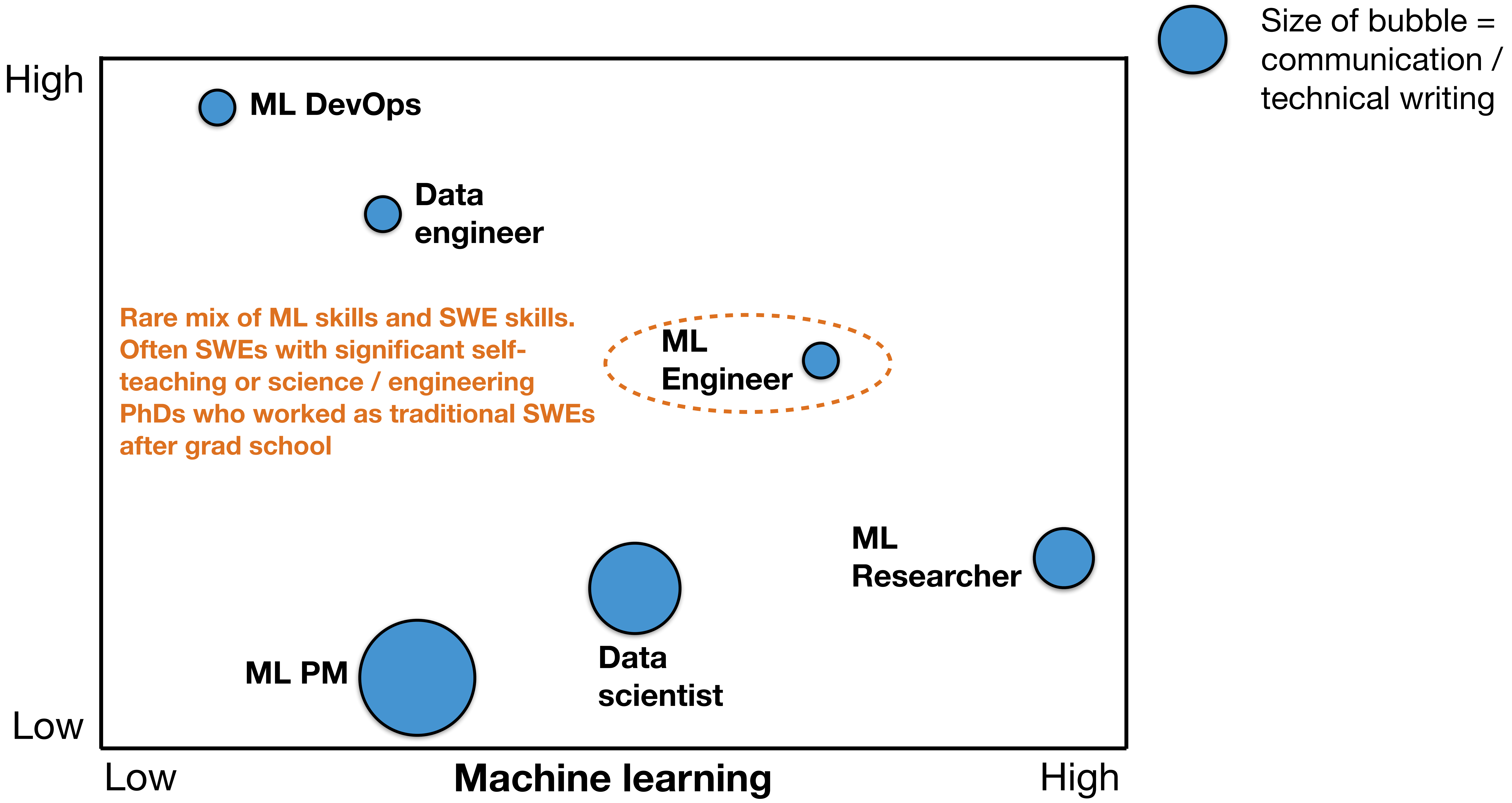
What skills are needed for the roles?



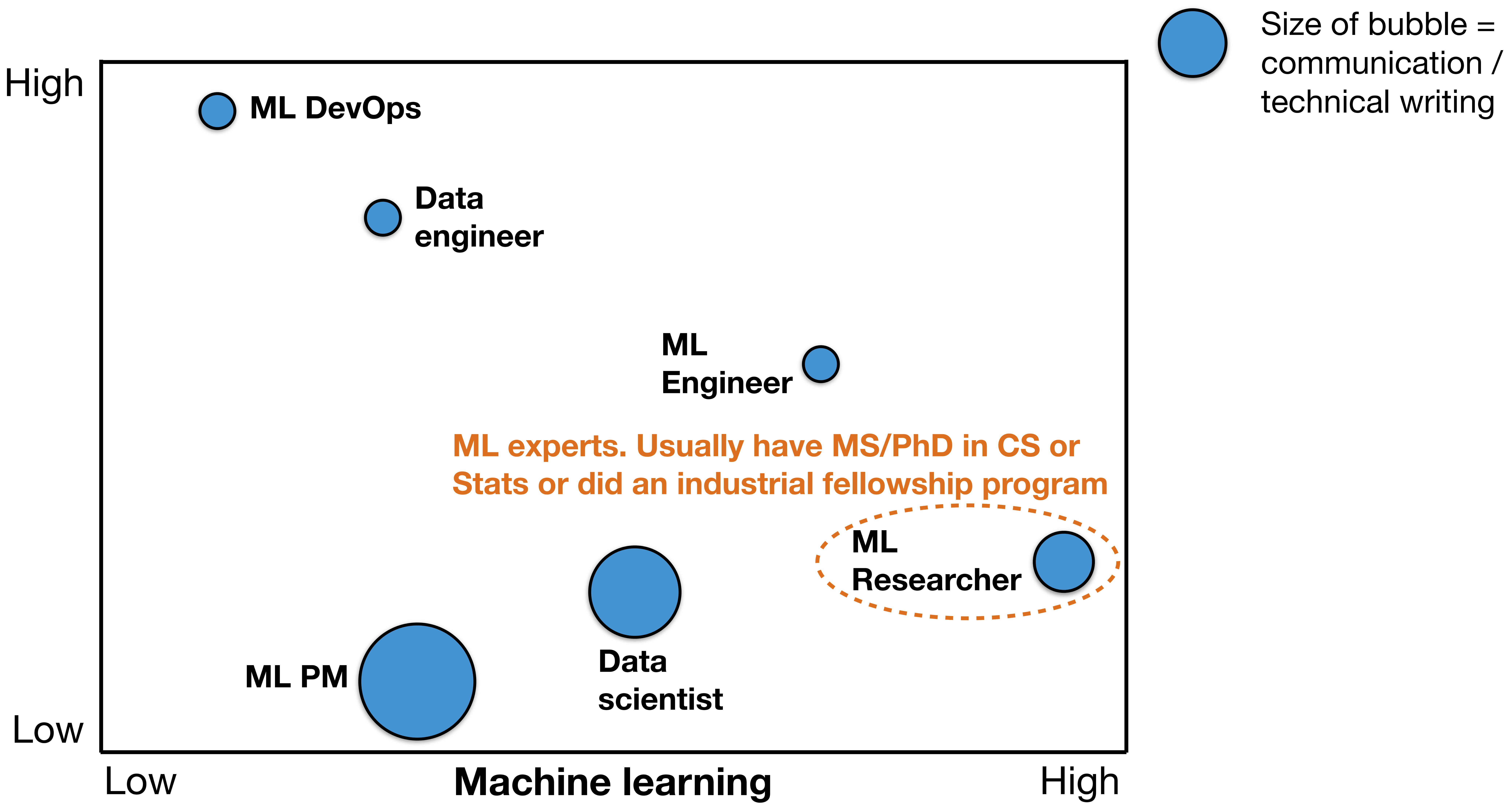
What skills are needed for the roles?



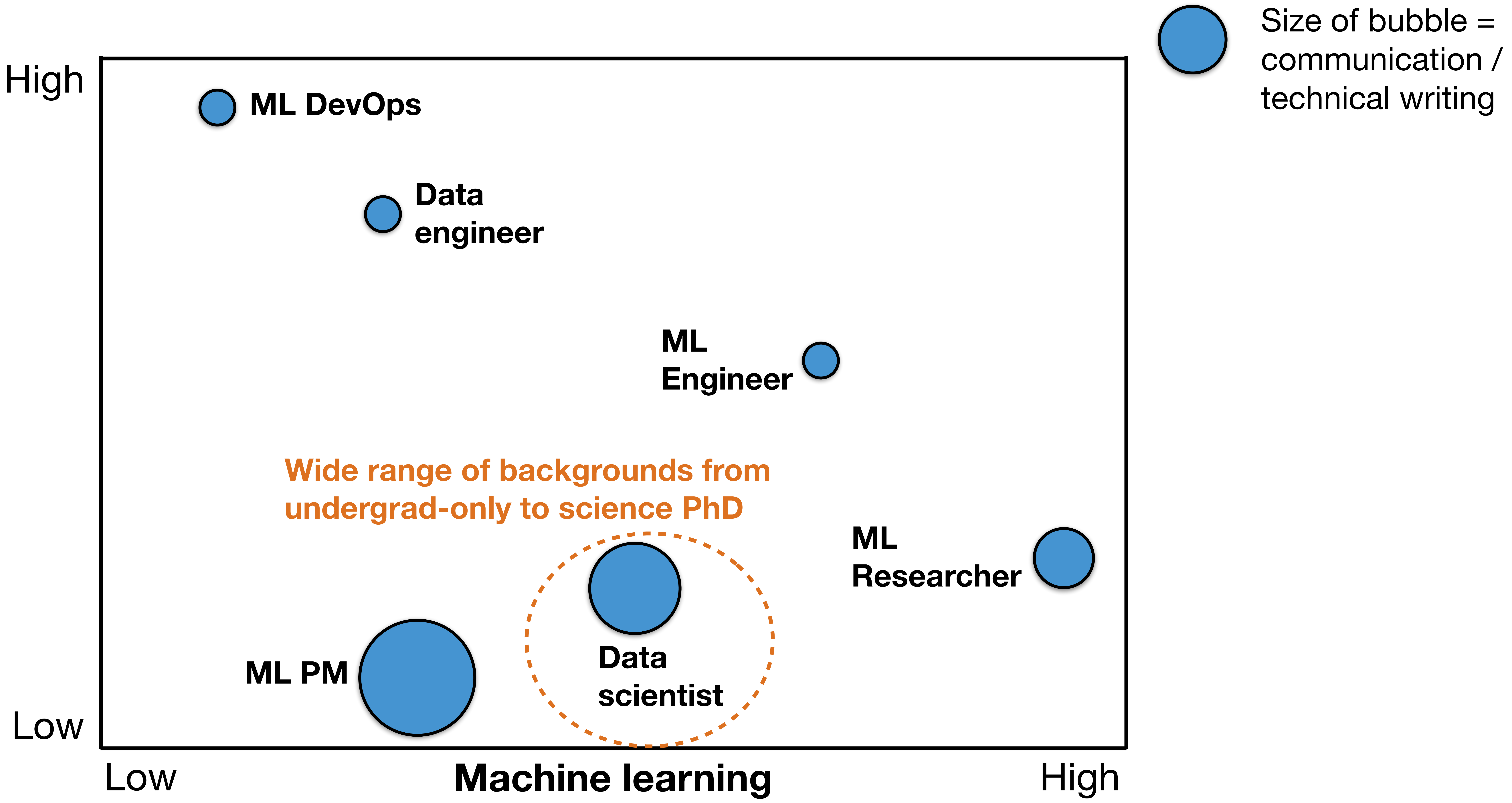
What skills are needed for the roles?



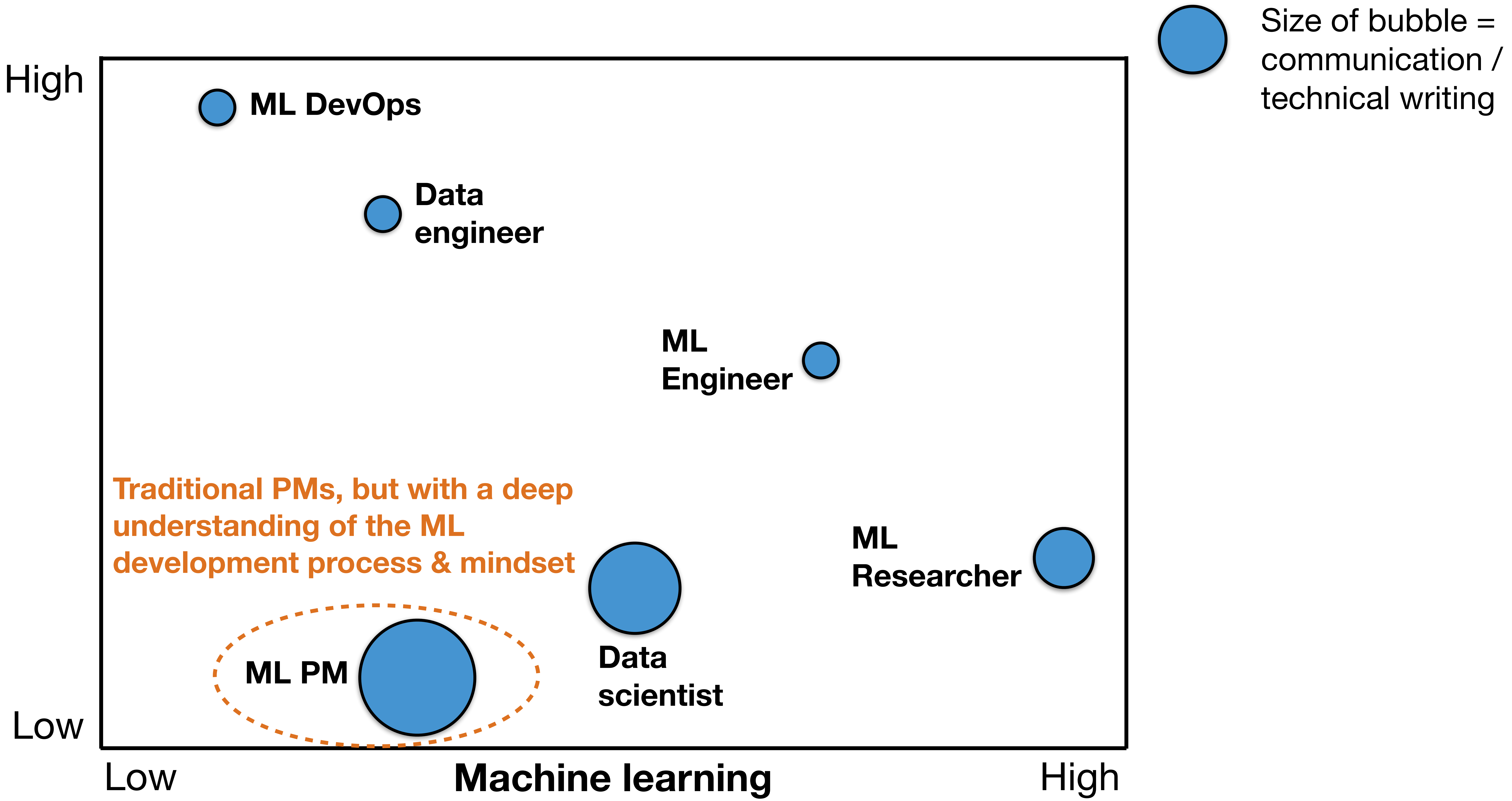
What skills are needed for the roles?



What skills are needed for the roles?



What skills are needed for the roles?



Questions?

Module overview

Roles

- ML-related roles and the skills they require

Orgs

- How ML teams are organized and how they fit into the broader organization

Managing

- How to manage a ML team

Hiring

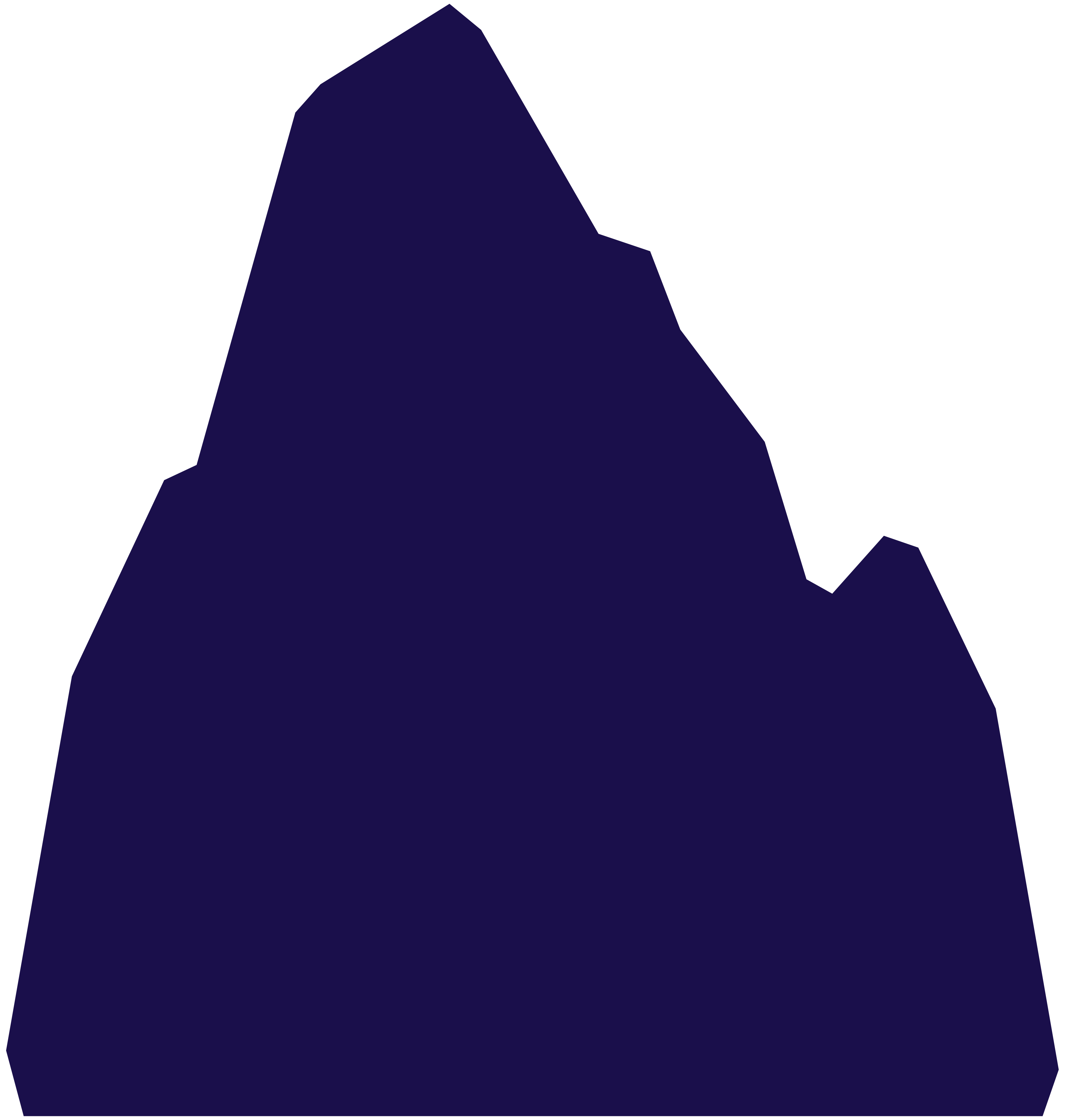
- How to hire ML engineers. How to get hired. Exam for the course

ML org structures - lessons learned

- No consensus yet on the right way to structure a ML team
- This lecture: taxonomy of best practices for different organizational maturity levels

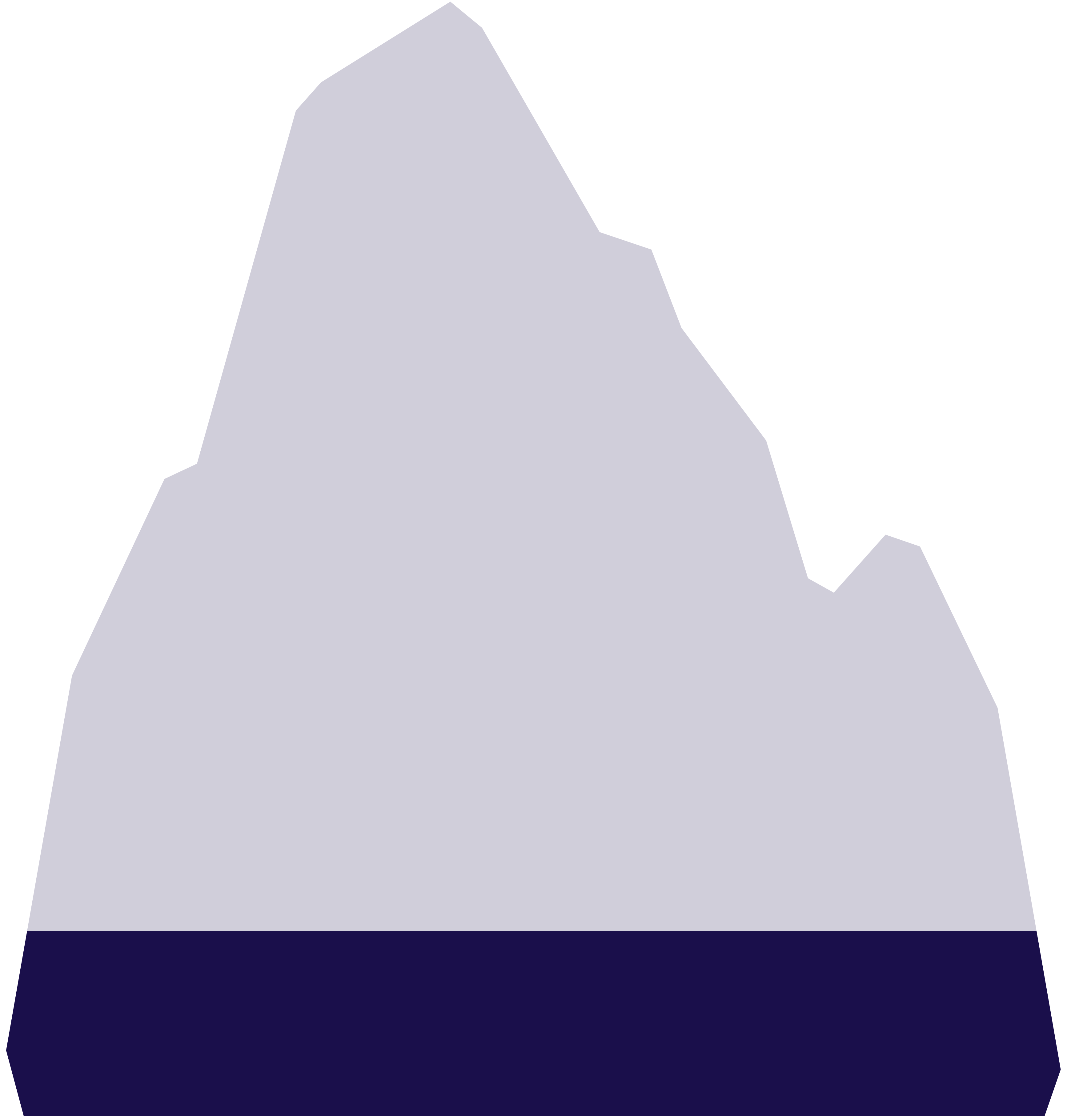
ML organization archetypes

The ML Organization Mountain



ML organization archetypes

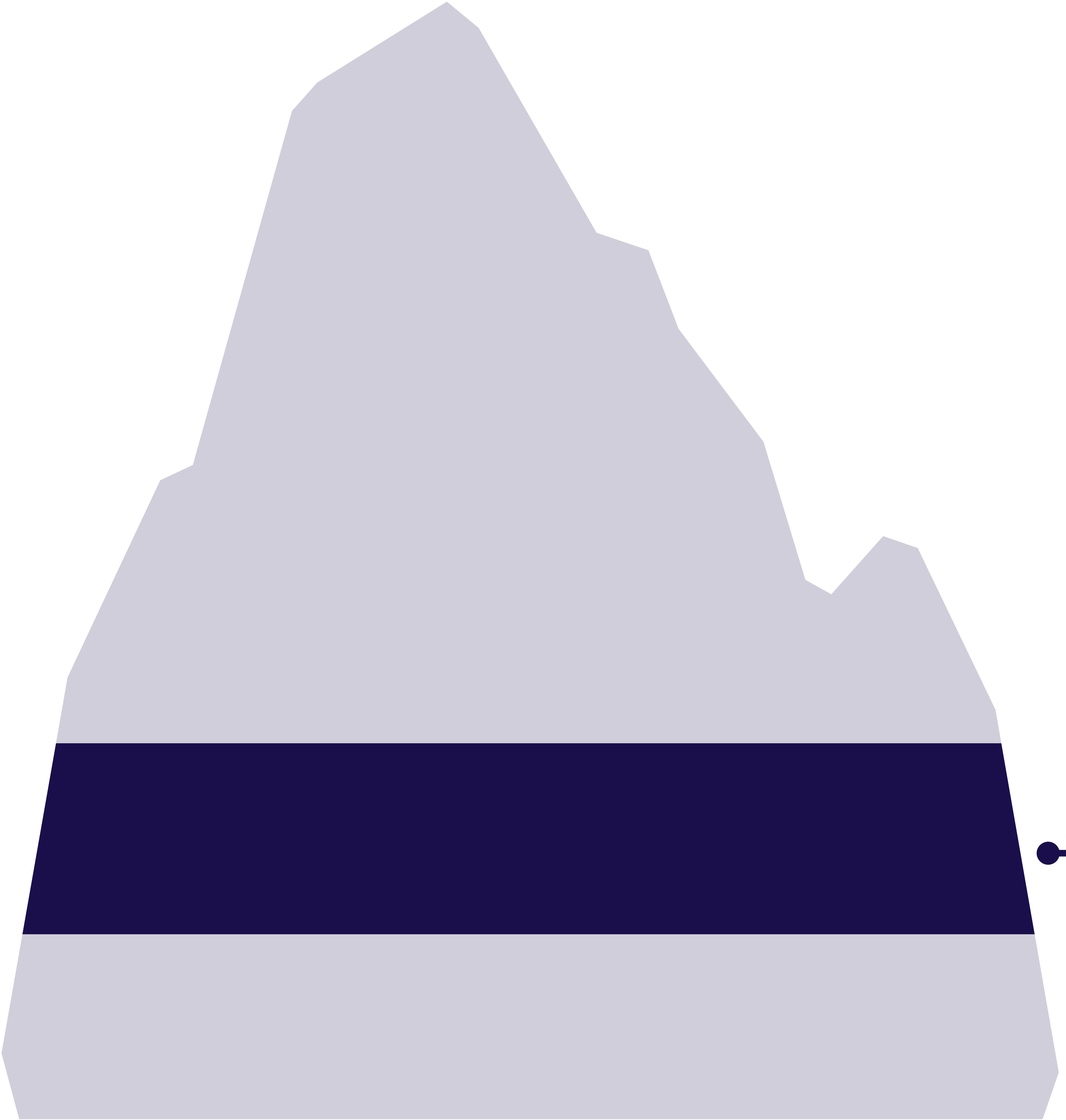
The ML Organization Mountain



Nascent / Ad-Hoc ML	
What it looks like	• No one is doing ML, or ML is done on an ad-hoc basis • Little ML expertise in-house
Example organizations	• Most small-medium businesses • Less technology-forward large companies (education, logistics, etc)
Advantages	• Often low-hanging fruit for ML
Dis-advantages	• Little support for ML projects, difficult to hire and retain good talent

ML organization archetypes

The ML Organization Mountain



ML R&D

What it looks like
- ML efforts are centered in the R&D arm of the organization
- Often hire researchers / PhDs & write papers

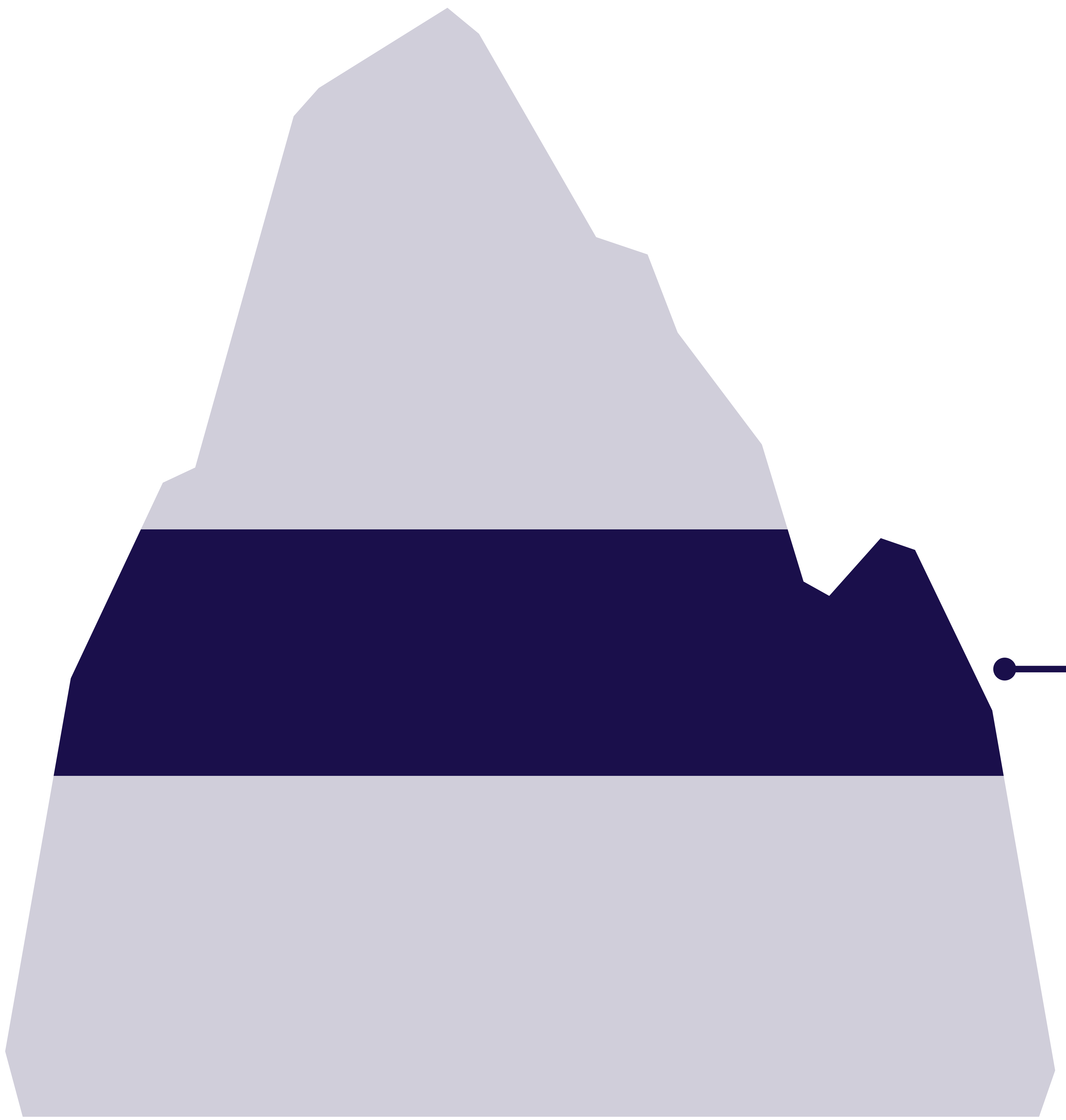
Example organizations
- Larger Oil & gas, manufacturing, telecom companies

Advantages
- Often can hire experienced researchers
- Can work on long-term business priorities & big wins

Dis-advantages
- Difficult to get data
- Rarely translates into actual business value, so usually the amount of investment remains small

ML organization archetypes

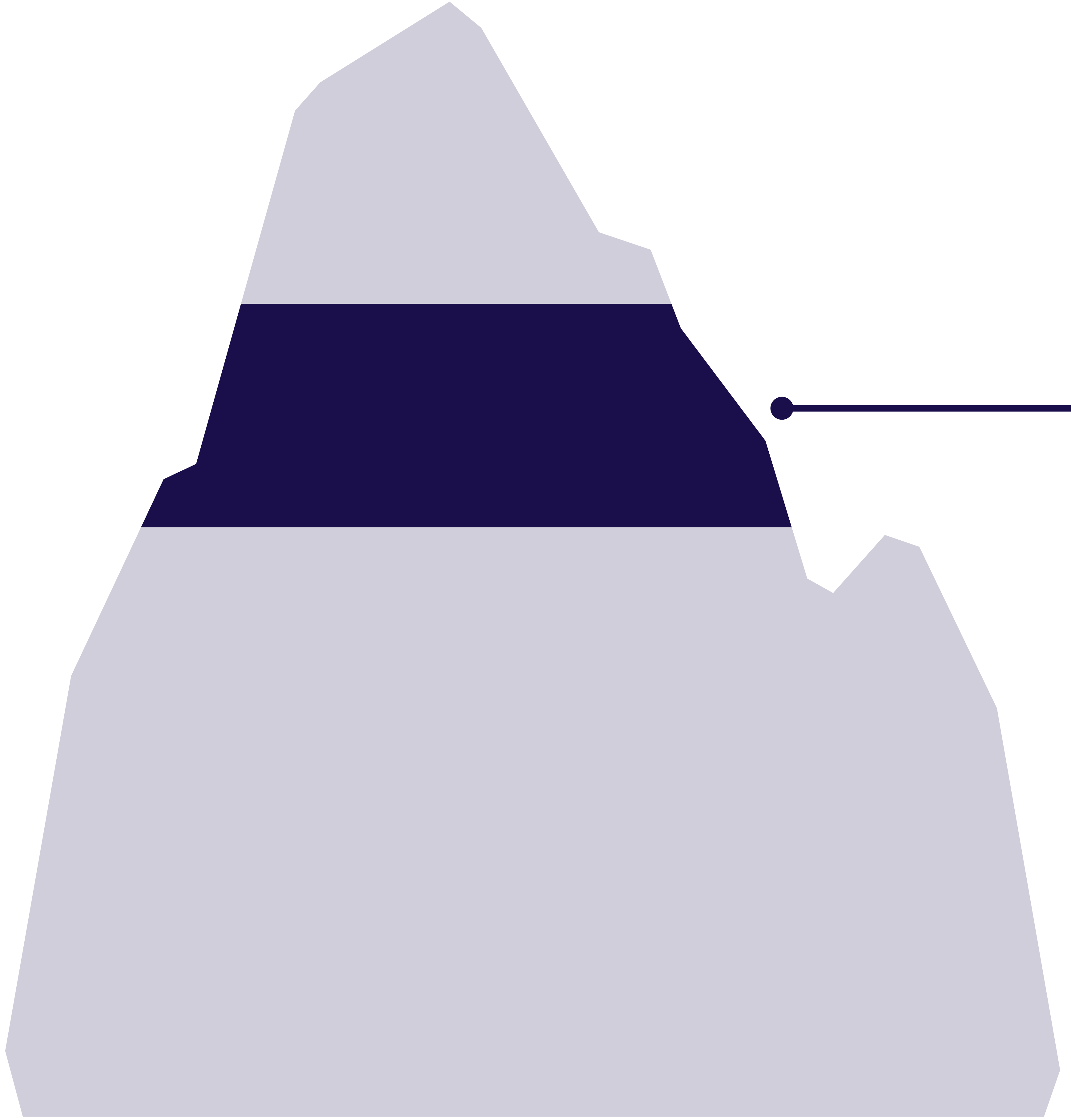
The ML Organization Mountain



ML embedded into business / product teams	
What it looks like	• Certain product teams or business units have ML expertise along-side their software or analytics talent • ML reports up to the team's engineering lead or tech lead
Example organizations	• Software / technology companies • Financial services companies
Advantages	• ML improvements are likely to lead to business value • Tight feedback cycle between idea and product improvement
Dis-advantages	• Hard to hire and develop top talent • Access to resources (data / compute) can lag • ML project cycles conflict with engineering mgmt • Long-term projects can be hard to justify

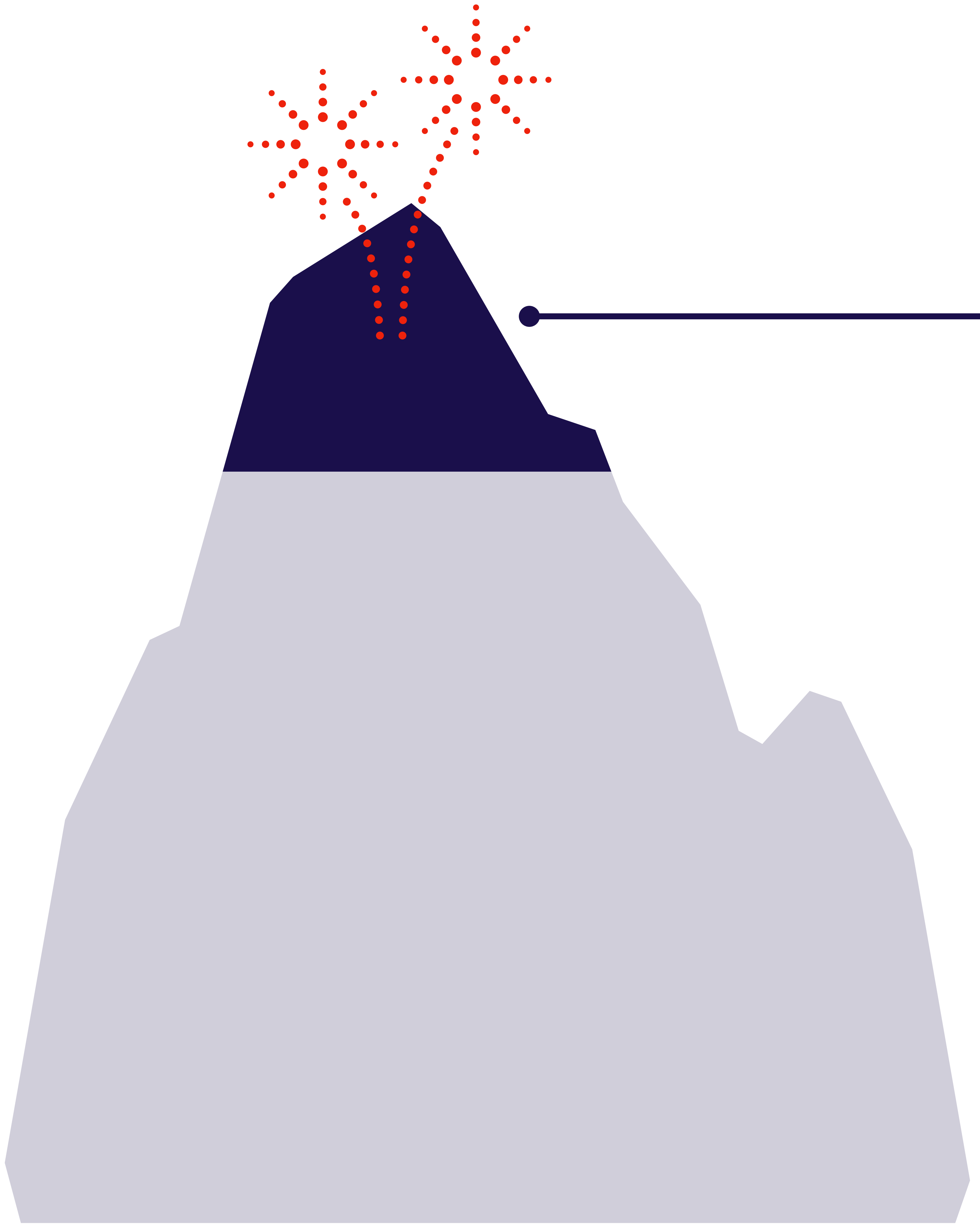
ML organization archetypes

The ML Organization Mountain



Independent ML Function	
What it looks like	• ML division reporting to senior leadership (often CEO) • ML PMs work with MLRs, MLEs, and customers to build ML into products • Teams sometimes publish long-term research
Example organizations	• Large financial services companies
Advantages	• Talent density allows to hire & train top practitioners • Senior leaders can marshal data / compute resources • Can invest in tooling, practices, and culture around ML development
Dis-advantages	• Model handoffs to lines of business can be challenging - users need to buy-in and be educated on model use • Feedback cycles can be slow

ML organization archetypes



ML-First Organizations	
What it looks like	• CEO buy-in • ML division working on challenging, long-term projects • ML expertise in every line of business focusing on quick wins and working with central ML division
Example organizations	• Large tech companies • ML-focused startups
Advantages	• Best data access: data thinking permeates the org • Recruiting: ML team works on hardest problems • Easiest deployment: product teams understand ML
Dis-advantages	• Hard to implement • Challenging & expensive to recruit enough talent • Culturally difficult to embed ML thinking everywhere

ML team structures - design choices

	Key questions
Software engineering vs research	• To what extent is the ML team responsible for building or integrating with software? • How important are SWE skills on the team?
Data ownership	• How much control does the ML team have over data collection, warehousing, labeling, and pipelining?
Model ownership	• Is the ML team responsible for deploying models into production? • Who maintains deployed models?

ML team structures - design choices

	ML R&D
Software engineering vs research	• Research prioritized over SWE skills • Researcher-SWE collaboration lacking
Data ownership	• ML team has no control over data • ML team typically will not have data engineering component
Model ownership	• Models are rarely deployed into production

ML team structures - design choices

	ML R&D	Embedded ML
Software engineering vs research	• Research prioritized over SWE skills • Researcher-SWE collaboration lacking	• SWE skills prioritized over research skills • Often, all researchers need strong SWE as everyone expected to deploy
Data ownership	• ML team has no control over data • ML team typically will not have data engineering component	• ML team generally does not own data production / mgmt • Work with data engineers to build pipelines
Model ownership	• Models are rarely deployed into production	• ML engineers own the models that they deploy into production

ML team structures - design choices

	ML R&D	Embedded ML	ML Function
Software engineering vs research	• Research prioritized over SWE skills • Researcher-SWE collaboration lacking	• SWE skills prioritized over research skills • Often, all researchers need strong SWE as everyone expected to deploy	• Each team has a strong mix of SWE and research skills • SWE and researchers work closely together within team
Data ownership	• ML team has no control over data • ML team typically will not have data engineering component	• ML team generally does not own data production / mgmt • Work with data engineers to build pipelines	• ML team has a voice in data governance discussions • ML team has strong internal data engineering function
Model ownership	• Models are rarely deployed into production	• ML engineers own the models that they deploy into production	• ML team hands off models to user, but is responsible for maintaining them

ML team structures - design choices

	ML R&D	Embedded ML	ML Function	ML First
Software engineering vs research	• Research prioritized over SWE skills • Researcher-SWE collaboration lacking	• SWE skills prioritized over research skills • Often, all researchers need strong SWE as everyone expected to deploy	• Each team has a strong mix of SWE and research skills • SWE and researchers work closely together within team	• Different teams are more or less research oriented • Research teams collaborate closely with SWE teams
Data ownership	• ML team has no control over data • ML team typically will not have data engineering component	• ML team generally does not own data production / mgmt • Work with data engineers to build pipelines	• ML team has a voice in data governance discussions • ML team has strong internal data engineering function	• ML team often owns company-wide data infrastructure
Model ownership	• Models are rarely deployed into production	• ML engineers own the models that they deploy into production	• ML team hands off models to user, but is responsible for maintaining them	• ML team hands off models to user, who operates and maintains them

ML team structures - design choices

	ML R&D	Embedded ML	ML Function	ML First
Software engineering vs research	• Research prioritized over SWE skills • Researcher-SWE collaboration lacking	• SWE skills prioritized over research skills • Often, all researchers need strong SWE as everyone expected to deploy	• Each team has a strong mix of SWE and research skills • SWE and researchers work closely together within team	• Different teams are more or less research oriented • Research teams collaborate closely with SWE teams
Data ownership	• ML team has no control over data • ML team typically will not have data engineering component	• ML team generally does not own data production / mgmt • Work with data engineers to build pipelines	• ML team has a voice in data governance discussions • ML team has strong internal data engineering function	• ML team often owns company-wide data infrastructure
Model ownership	• Models are rarely deployed into production	• ML engineers own the models that they deploy into production	• ML team hands off models to user, but is responsible for maintaining them	• ML team hands off models to user, who operates and maintains them

Questions?

Module overview

Roles

- ML-related roles and the skills they require

Orgs

- How ML teams are organized and how they fit into the broader organization

Managing

- **How to manage a ML team**

Hiring

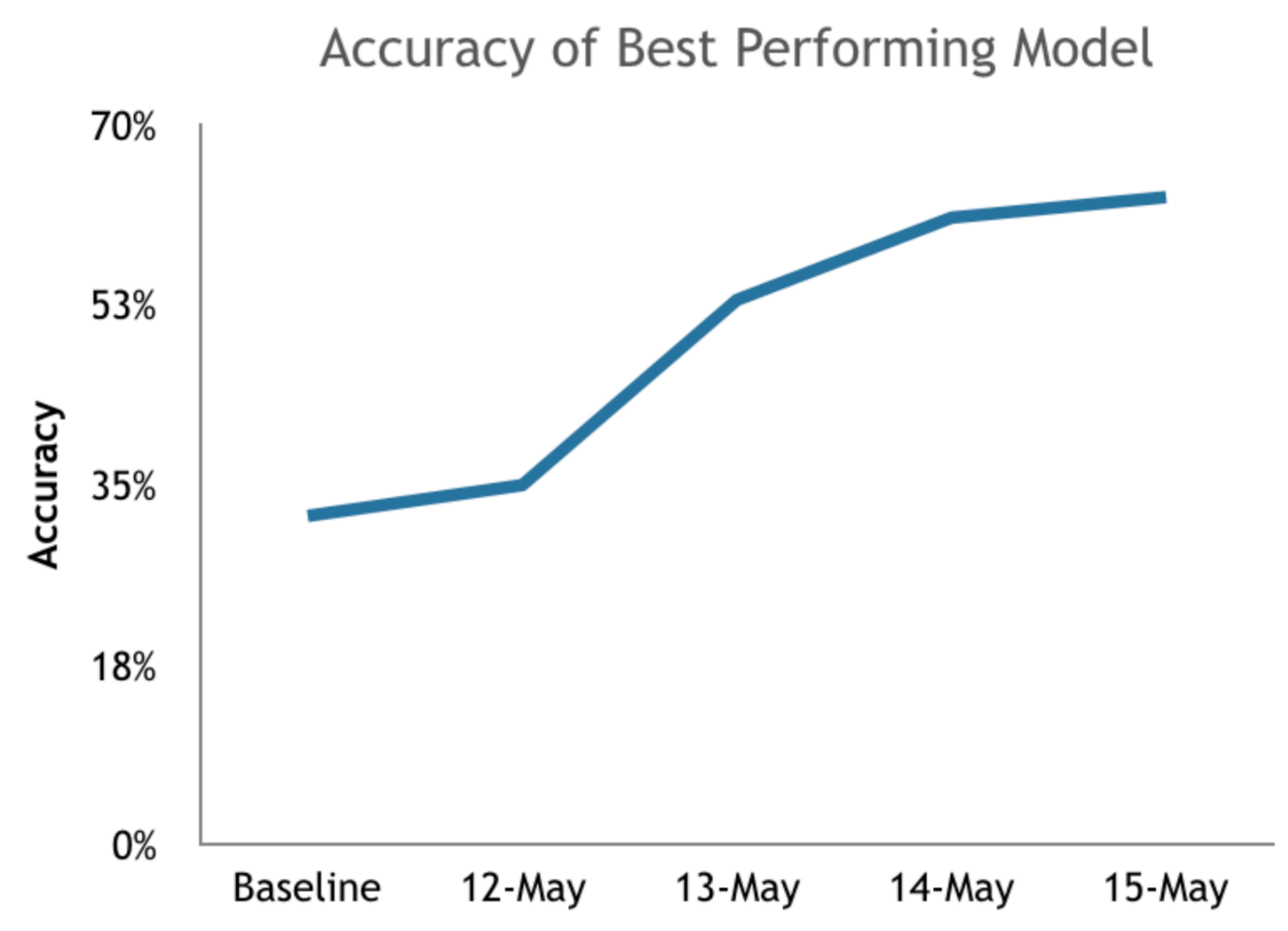
- How to hire ML engineers. How to get hired. Exam for the course

Managing ML teams is challenging

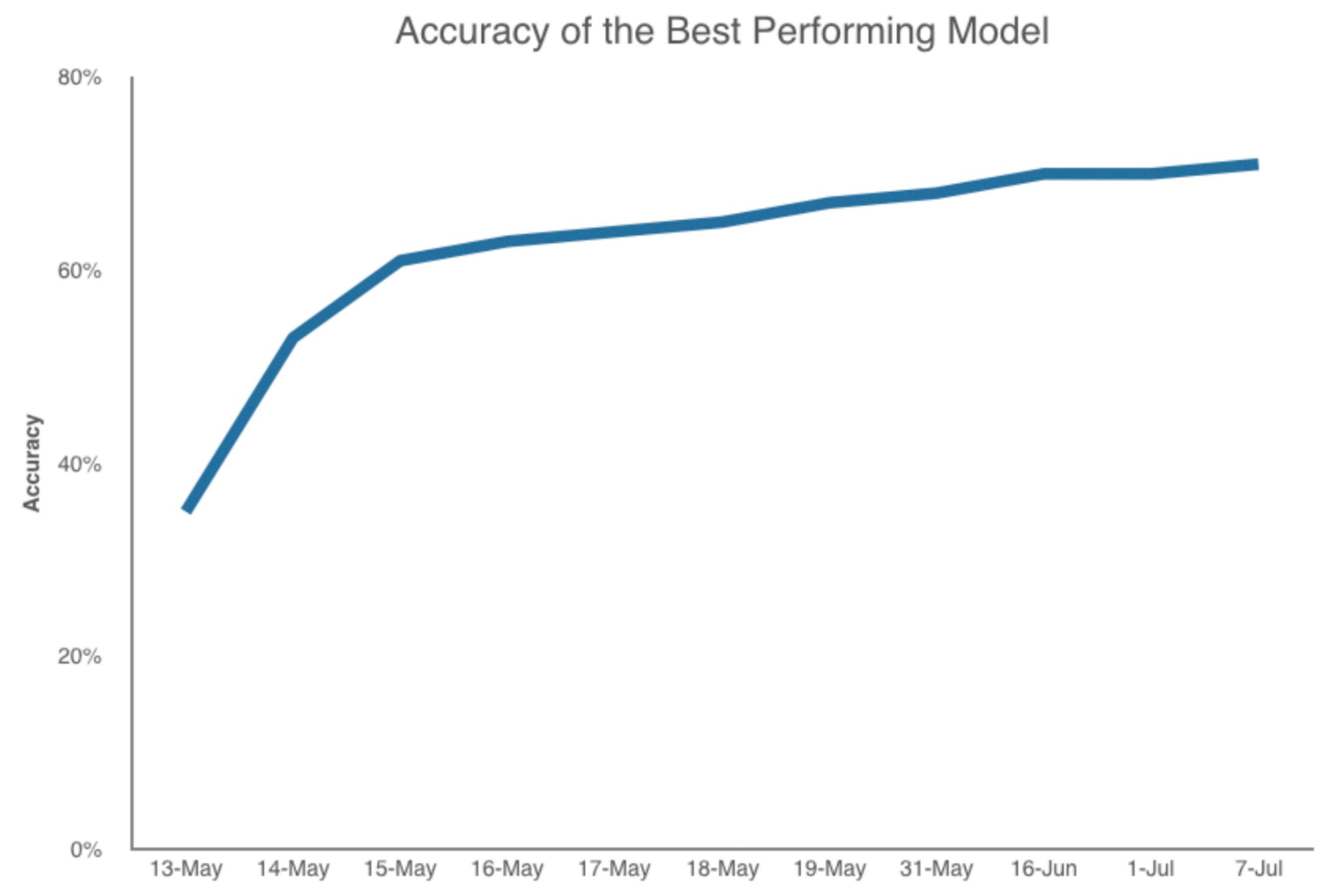
- It’s hard to tell in advance how hard or easy something is

Managing ML teams is challenging

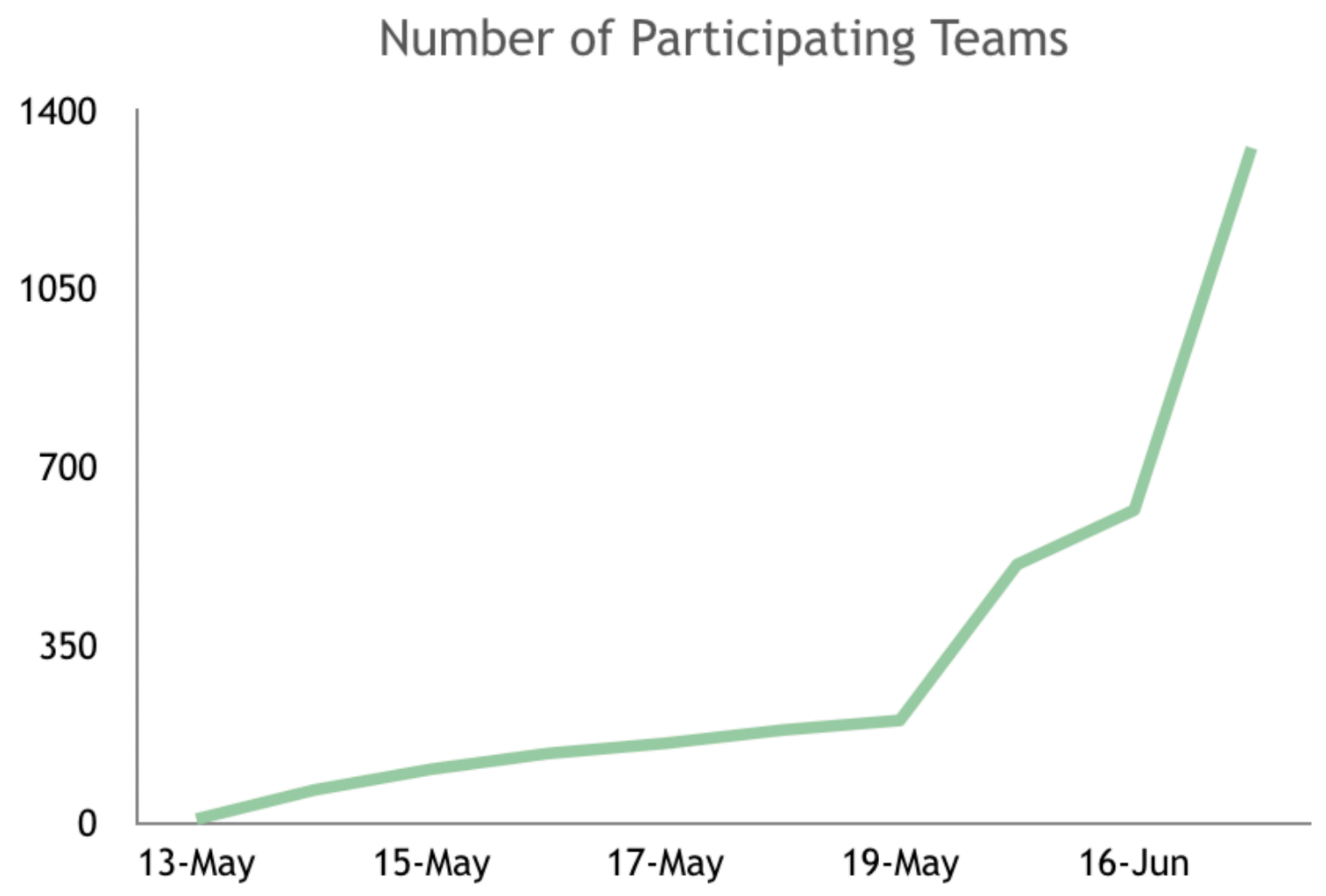
Accuracy improvement in first week



Accuracy improvement in three months



Effort



It's hard to tell in advance how easy or hard something is

https://medium.com/@l2k/why-are-machine-learning-projects-so-hard-to-manage-8e9b9cf49641

Managing ML teams is challenging

- It's hard to tell in advance how hard or easy something is
- ML progress is nonlinear
    - Very common for projects to stall for weeks or longer
    - In early stages, difficult to plan project because unclear what will work
    - As a result, estimating project timelines is extremely difficult
    - I.e., production ML is still somewhere between "research" and "engineering"

Managing ML teams is challenging

- It’s hard to tell in advance how hard or easy something is
- ML progress is nonlinear
- There are cultural gaps between research and engineering
    - Different values, backgrounds, goals, norms
    - In toxic cultures, the two sides often don’t value one another

Managing ML teams is challenging

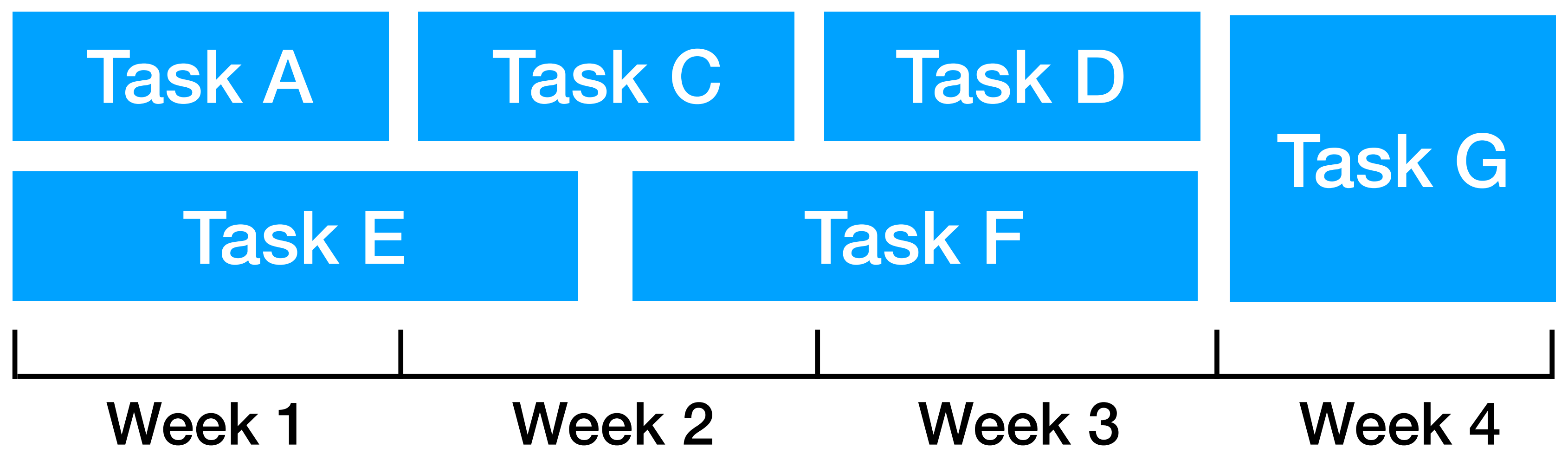
- It’s hard to tell in advance how hard or easy something is
- ML progress is nonlinear
- There are cultural gaps between research and engineering
- Leaders often don’t understand it

How to manage ML teams better

- Do ML Project planning probabilistically

How to manage ML teams better

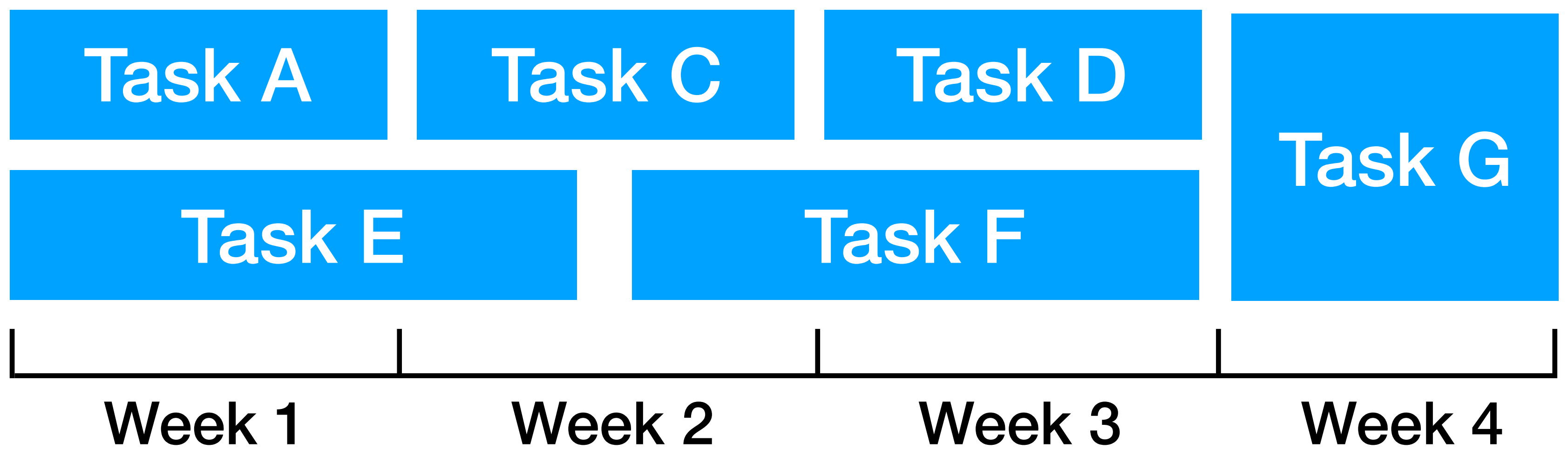
- Do ML project planning probabilistically
  - From:



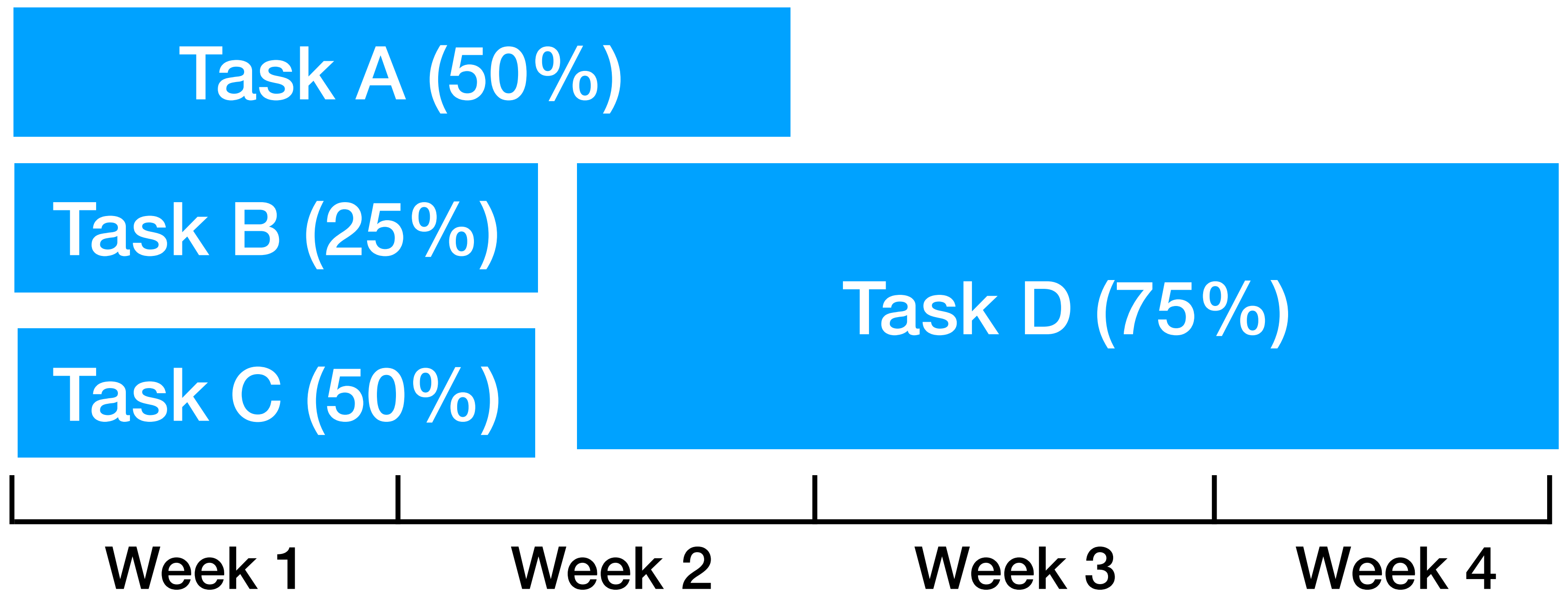
  - To:

How to manage ML teams better

- Do ML project planning probabilistically
  - From:

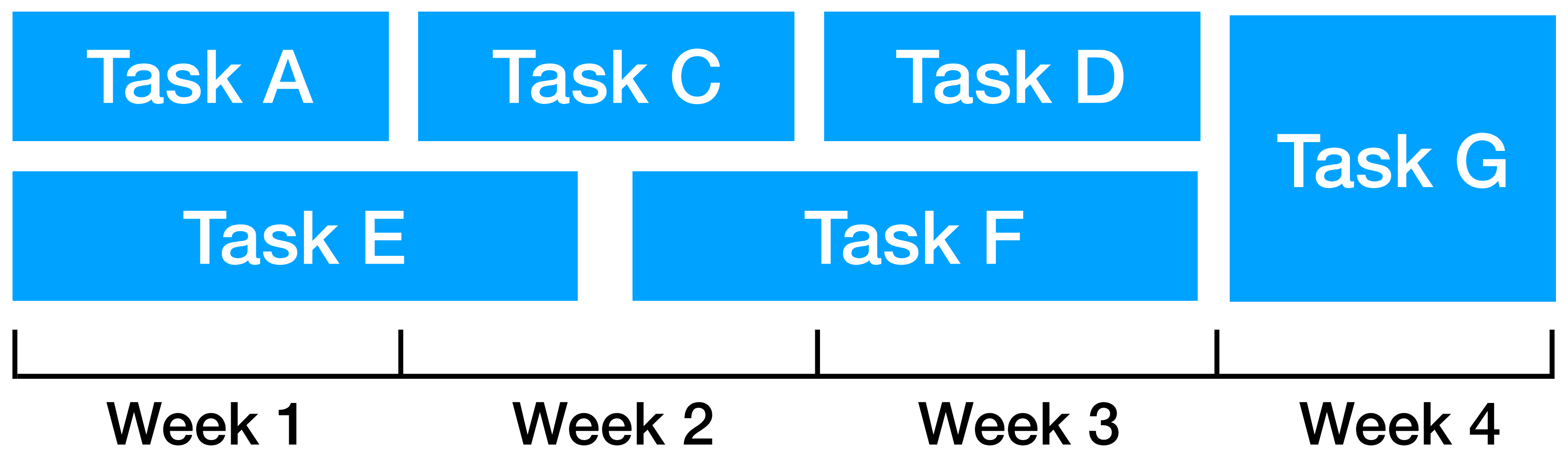


  - To:

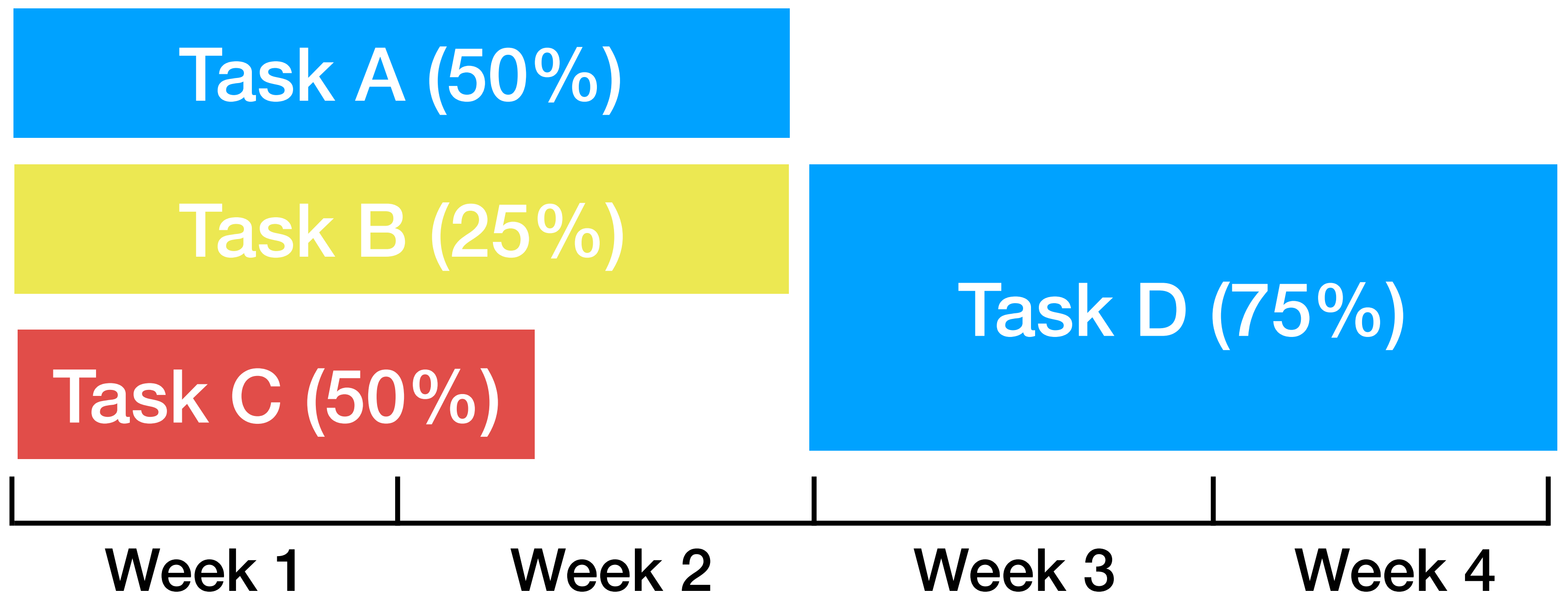


How to manage ML teams better

- Do ML project planning probabilistically
    - From:

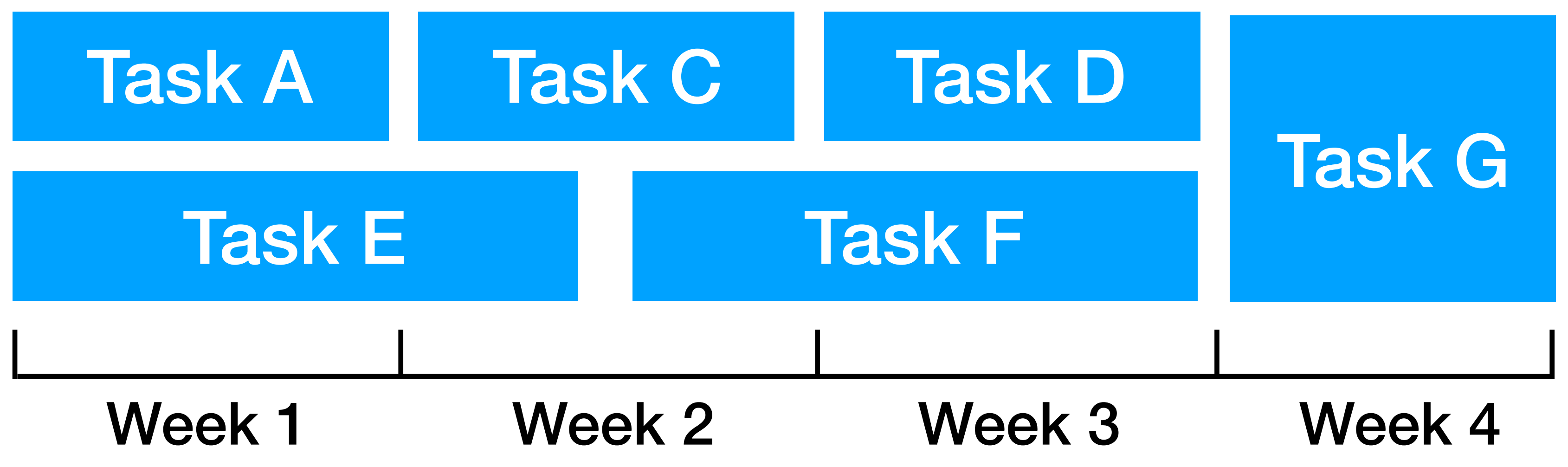


- To:

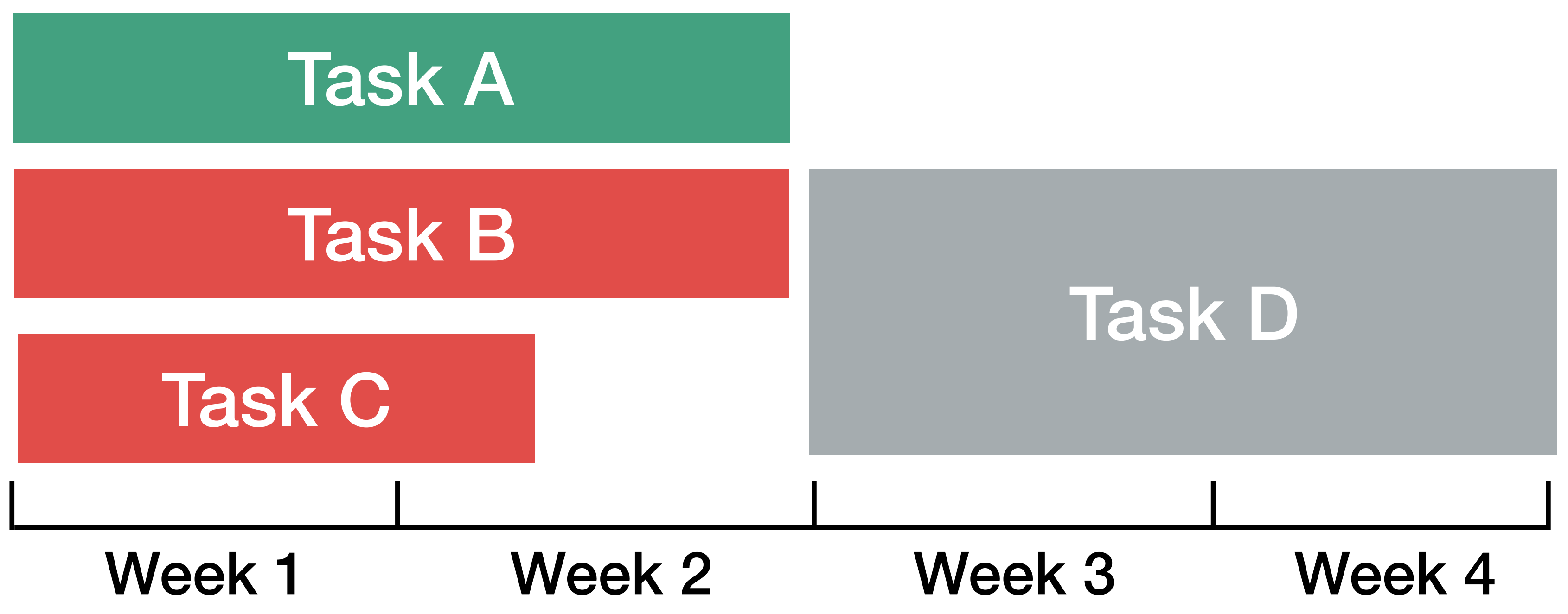


How to manage ML teams better

- Do ML project planning probabilistically
  - From:

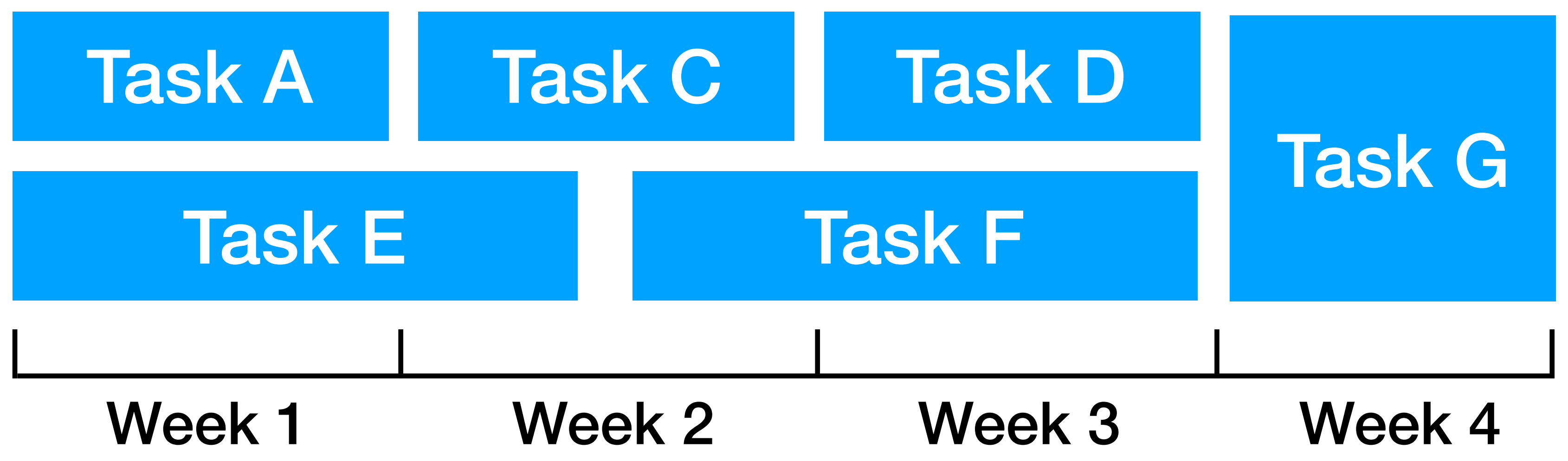


  - To:

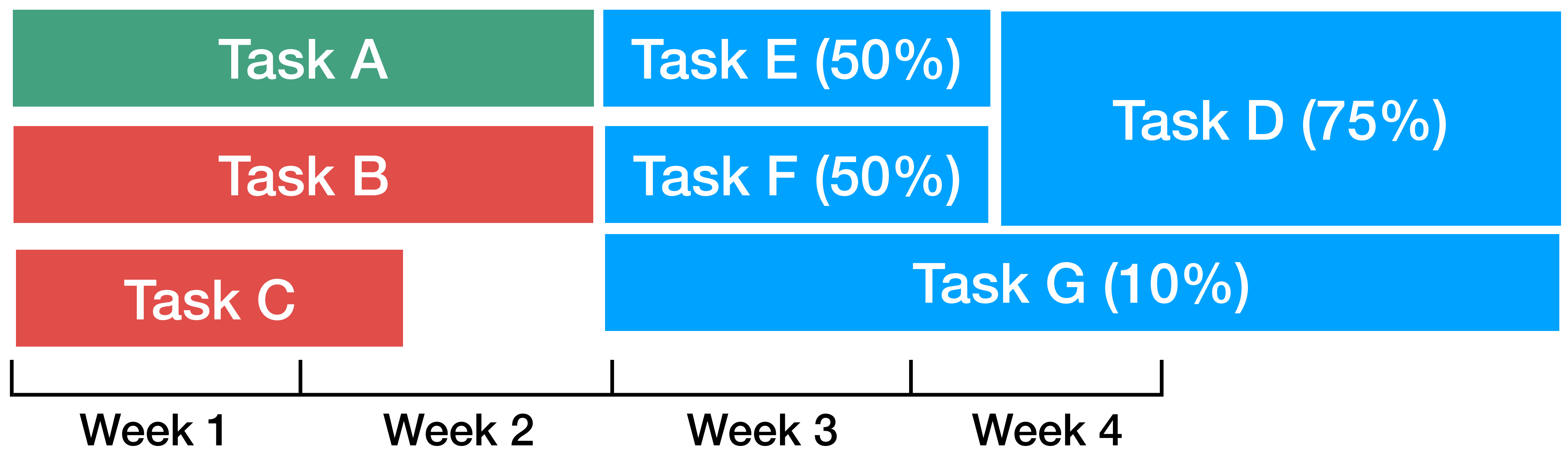


How to manage ML teams better

- Do ML project planning probabilistically
  - From:



  - To:



How to manage ML teams better

- Do ML Project planning probabilistically
- Attempt a portfolio of approaches
- Measure progress based on inputs, not results
- Have researchers and engineers work together
- Get end-to-end pipelines together quickly to demonstrate quick wins
- Educate leadership on ML timeline uncertainty

Resources for educating execs

- https://a16z.com/2016/06/10/ai-deep-learning-machines/
- Pieter’s upcoming AI Strategy class: https://emeritus-executive.berkeley.edu/artificial-intelligence/

Questions?

Module overview

Roles	• ML-related roles and the skills they require
Orgs	• How ML teams are organized and how they fit into the broader organization
Managing	• How to manage a ML team
Hiring	• **How to hire ML engineers. How to get hired. Exam for the course**

Hiring for ML - outline

- The AI Talent Gap
- Sourcing
- Interviewing
- Finding a job
- Course exam

Hiring for ML - outline

- **The AI Talent Gap**
- Sourcing
- Interviewing
- Finding a job
- Course exam

The AI Talent Gap

How many people know how to build AI systems?
5,000 (actively publishing research [Element AI])
10,000 (estimated num people with the right skillset [Element AI])
22,000 (PhD-educated AI researchers [Bloomberg])
90,000 (upper bound on number of people [Element AI])
200,000 - 300,000 (Number of AI researcher / practitioners [Tencent])

3.6M (Number of software developers in the US)
18.2M (Number of software developers in the world)

Sources: The AI Talent Shortage (Nikolai Yakovenko) https://medium.com/@Moscow25/the-ai-talent-shortage-704d8cf0c4cc
Just How Shallow is the Artificial Intelligence Talent Pool (Jeremy Kahn)
https://www.bloomberg.com/news/articles/2018-02-07/just-how-shallow-is-the-artificial-intelligence-talent-pool

The AI talent gap

Fierce competition for AI talent

"Everyone agrees that the competition to hire people who know how to build artificial intelligence systems is intense. It's turned once-staid academic conferences into frenzied meet markets for corporate recruiters and driven the salaries of the top researchers to seven-figures."

(Bloomberg)

Sources: The AI Talent Shortage (Nikolai Yakovenko) https://medium.com/@Moscow25/the-ai-talent-shortage-704d8cf0c4cc
Just How Shallow is the Artificial Intelligence Talent Pool (Jeremy Kahn)
https://www.bloomberg.com/news/articles/2018-02-07/just-how-shallow-is-the-artificial-intelligence-talent-pool

The AI talent gap

Fierce competition for AI talent

“Hiring is crazy right now. ML is a young field that got popular very quickly. There’s a ton of demand and not a lot of supply.”

(Computer Vision Engineer at Series C startup)

The AI talent gap

Fierce competition for AI talent

“Hiring for ML is really challenging and takes way more time and effort than we expected. We have someone working on it full-time and we’re still only able to get a few people per quarter”

(Startup Founder)

Hiring for ML - outline

- The AI Talent Gap
- **Sourcing**
- Interviewing
- Finding a job
- Course exam

Most common ML roles

- ML product manager
- DevOps
- Data engineer
- ML engineer
- ML researcher / ML scientist
- Data scientist

Most common ML roles

- ML product manager
- DevOps
- Data engineer
- ML engineer
- ML researcher / ML scientist
- Data scientist

Slightly different mindset required. Helpful to look for demonstrated interest in AI - courses, conferences, re-implementations, etc

Most common ML roles

- ML product manager
- DevOps
- Data engineer
- ML engineer
- ML researcher / ML scientist
- Data scientist

Our focus

How to hire MLEs - the wrong way

- Job Description (Unicorn Machine Learning Engineer)
  - Duties
    - Keep up with the state of the art
    - Implement models from scratch
    - Deep understanding of mathematics & ability to come up with new models
    - Build tooling & infrastructure for the ML team
    - Build data pipelines for the ML team
    - Deploy & monitor models into production
  - Requirements
    - PhD
    - At least 4 years tensorflow experience
    - At least 4 years as a software engineer
    - Publications in top ML conference
    - Experience building large-scale distributed systems

How to hire MLEs - the right way

- Hire for software engineering skills, interest in ML, and desire to learn. Train to do ML.
- Go more junior. Most undergrad computer science students graduate with ML experience.
- Be more specific about what you need. Not every ML engineer needs to do DevOps.

How to hire MLRs

- Look for quality of publications, not quantity (e.g., originality of ideas, quality of execution)
- Look for researchers with an eye for working on important problems (many researchers focus on trendy problems without considering why they matter)
- Look for researchers with experience outside of academia
- Consider hiring talented people from adjacent fields (physics, statistics, math)
- Consider hiring people without PhDs (e.g., talented undergraduate / masters students, graduates of Google/Facebook/OpenAI fellowship programs, dedicated self-studiers)

How to find MLE/MLR candidates

- Standard sources: LinkedIn, recruiters, on-campus recruiting, etc
- Monitor arXiv and top conferences and flag first authors of papers you like
- Look for good reimplementations of papers you like
- Attend ML research conferences (NeurIPS, ICLR, ICML)

How to attract MLR / MLE candidates

What do machine learning practitioners want?	How to make your company stand out?
• Work with cutting edge tools & techniques	• Work on research-oriented projects. Publicize them. Invest in tooling for your team & empower employees to try new tools.
• Build skills / knowledge in an exciting field	• Build team culture around learning (reading groups, learning days, professional development budget, conference budget)
• Work with excellent people	• Hire high-profile people. Help your best people build their profile through publishing blogs & papers.
• Work on interesting datasets	• Sell the uniqueness of your dataset in recruiting materials.
• Do work that matters	• Sell the mission of your company and potential impact of machine learning on that mission. Work on projects that have a tangible impact today.

Hiring for ML - outline

- The AI Talent Gap
- Sourcing
- **Interviewing**
- Finding a job
- Course exam

What to test in an ML interview?

- Hire for strengths
- Meet a minimum bar for everything else

What to test in an ML interview?

- Validate your hypotheses of candidate's strengths
  - Researchers: make sure they can think creatively about new ML problems, probe how thoughtful they were about previous projects
  - Engineers: make sure they are great generalist SWEs
- Make sure candidates meet a minimum bar on weaker areas
  - Researchers: test SWE knowledge and ability to write good code
  - SWEs: test ML knowledge

What happens in a ML interview?

- Much less well-defined than software engineering interviews
- Common types of assessments:
  - Background & culture fit
  - Whiteboard coding (similar to SWE interviews)
  - Pair coding (similar to SWE interviews)
  - Pair debugging (often ML-specific code)
  - Math puzzles (e.g., involving linear algebra)
  - Take-home ML project
  - Applied ML (e.g., explain how you'd solve this problem with ML)
  - Previous ML projects (e.g., probing on what you tried, why things did / didn't work)
  - ML theory (e.g., bias-variance tradeoff, overfitting, underfitting, understanding of specific algorithms)

Hiring for ML - outline

- The AI Talent Gap
- Sourcing
- Interviewing
- **Finding a job**
- Course exam

Where to look for a ML job?

- Standard sources: LinkedIn, recruiters, on-campus recruiting, etc
- ML research conferences (NeurIPS, ICLR, ICML)
- Apply directly (remember, there's a talent gap!)
- This course
    - Those who pass the exam will get access to our recruiting database

How to stand out for ML roles?

More impressive

- Build software engineering skills (e.g., work at a well-known software company)
- Exhibit interest in ML (e.g., conference attendance, online courses taken)
- Show you have broad knowledge of ML (e.g., write blog posts synthesizing a research area)
- Demonstrate ability to get ML projects done (e.g., create side projects, re-implement papers)
- Prove you can think creatively in ML (e.g., win Kaggle competitions, publish papers)

How to prepare for the interview?

- Prepare for a general SWE interview (e.g., “Cracking the Coding Interview”)
- Prepare to talk in detail about your past ML projects (remember details, prepare to talk about tradeoffs and decisions you made)
- Review how basic ML algorithms work (linear / logistic regression, nearest neighbor, decision trees, k-means, MLPs, ConvNets, recurrent nets, etc)
- Review ML theory
- Think about the problems the company you’re interviewing with may face and what ML techniques may apply to them

Hiring for ML - outline

- The AI Talent Gap
- Sourcing
- Interviewing
- Finding a job
- **Course exam**

Overview of the exam

- Designed to help you prepare for ML engineering interviews
- Take on your own time

Outline

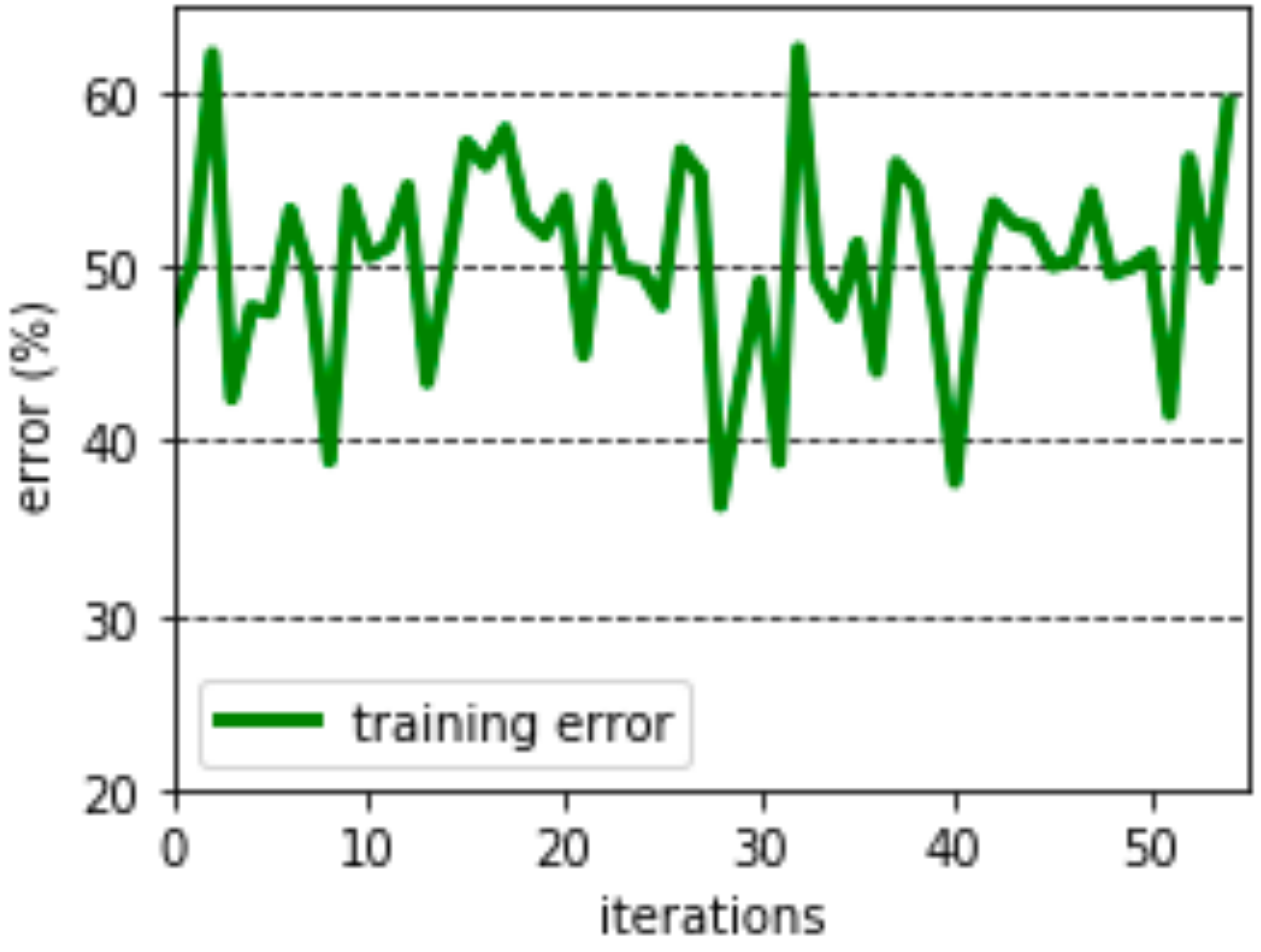
- **Problem setup** (e.g., for a particular problem, what data would you look for, what model would you choose, and what metric would you optimize?)
- **Algorithm knowledge** (e.g., how does an LSTM work?)
- **Understanding of common conventions** (e.g., what problem with RNNs does an LSTM solve?)
- **ML theory** (e.g., understanding bias and variance)
- **ML debugging** (e.g., what's likely to cause the bug here?)

Example question 1

- Why does the Residual Block in the ResNet architecture help with the vanishing gradient problem?

Example question 2

- Suppose you see the following learning curve when training on a single batch. Which of the following could be the cause? (Select all that apply)
  - Shuffled labels
  - Learning rate too low
  - Learning rate too high
  - Numerical instability
  - Too big a model



Example question 3

- **For each of the following prediction tasks, select the loss function that is best suited for it.**
- Predict sale price of a house listed for sale.
- Predict whether an image contains pornography or not.
- Predict the category of an email: personal, promotional, reminder, or spam.
- Predict whether a voice sample belongs to the owner of the phone.

(a) Mean Squared Error
(b) Categorical cross-entropy
(c) Binary cross-entropy
(d) CTC loss
(e) GAN loss

Questions?

Conclusion

Roles
- **Lots of different skills involved in production ML, so there's an opportunity for many to contribute**

Orgs
- **ML teams are becoming more standalone, hence more interdisciplinary**

Managing
- **Managing ML teams is hard. There's no silver bullet, but shifting toward probabilistic planning can help**

Hiring
- **Talent is scarce, so be specific about what is must-have. It can be hard to break in as an outsider - use projects to build awareness.**

Thank you!