

# Dirbtinio intelekto sistemų inžinerija

DI projektų valdymas: projektų planavimas.  
Išteklių paskirstymas ir rizikos valdymas

# Artificial Intelligence System Engineering

AI Project Management: Project planning  
and scheduling. Resource allocation and  
risk management

# Challenges in AI Project Management

—

- Evolving technology and requirements
- Data dependency and quality
- Integration with existing systems
- Ethical considerations and regulatory compliance

# Data dependency and quality

---

- Data is the foundational element that determines the success or failure of a model;
- Data is used for:
  - Training
  - Validation and testing
  - Production and Updates
- Data Accessibility (Before starting to work, try to get a sample data)
- Data Integration

# Project Planning in AI

- Key elements of project planning:
  - Defining objectives
  - Identifying deliverables
  - Creating a project roadmap
- Tools and techniques:
  - Gantt charts
  - Work Breakdown Structure (WBS)

# Scheduling in AI Projects

- Importance of iterative cycles (e.g., Agile sprints)
- Managing dependencies and milestones
- Techniques:
  - Rolling wave planning
  - Kanban boards

# Resource Allocation

---

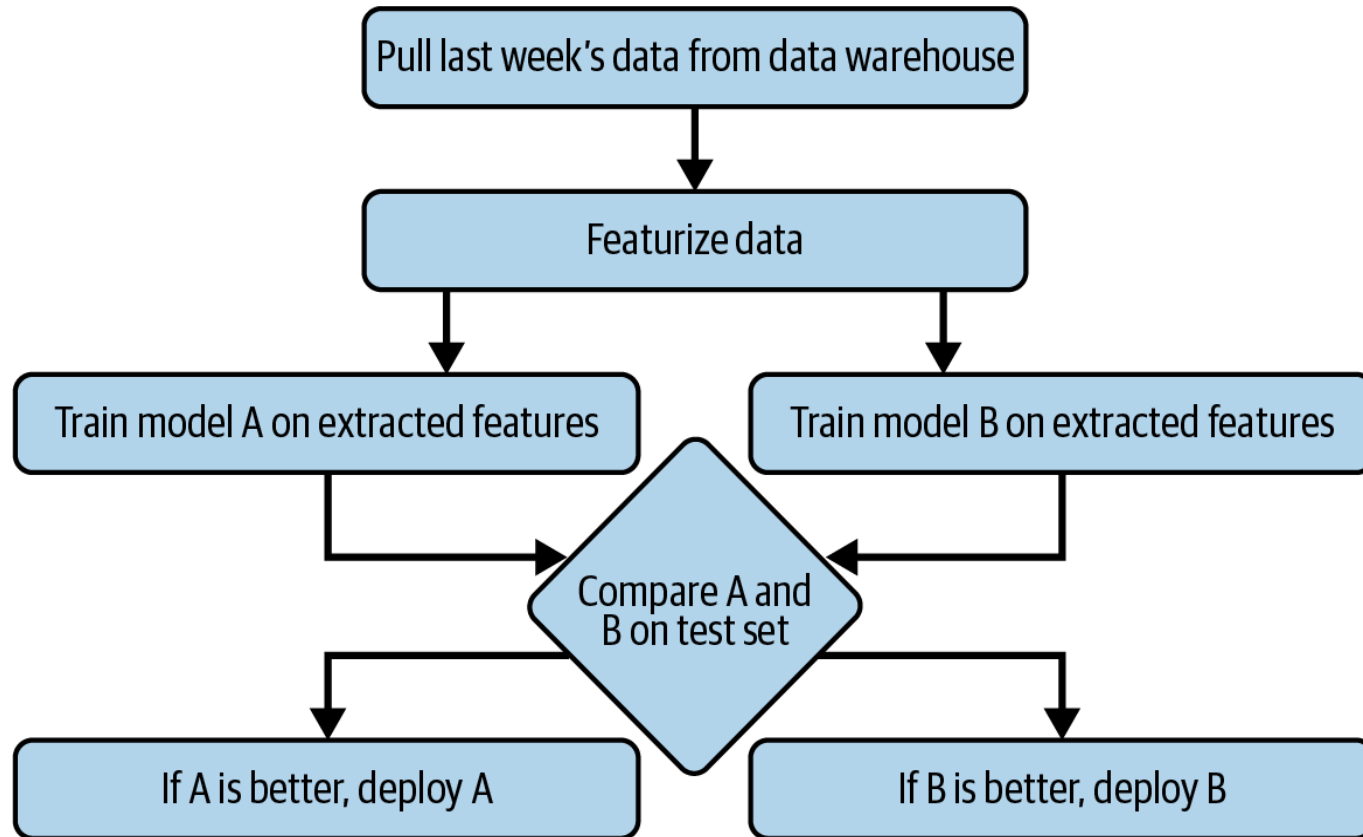
- Types of resources:
  - Human resources (data scientists, ML engineers)
  - Infrastructure resources (cloud computing, GPUs)
  - Data resources (datasets, storage)
- Strategies:
  - Dynamic resource allocation (e.g., scaling cloud resources)
  - Prioritizing tasks based on critical paths
- Tools:
  - Resource management plans
  - Automation with MLOps platforms

# Resource Management for ML workflows

- Developing ML systems is an iterative process
- Two key characteristics of ML workflows that influence their resource management: repetitiveness and dependencies.
- How to solve this?
  - Cron
  - Schedulers
  - and Orchestrators



# Resource Management for ML workflows



- Cron job - run a script at a predetermined time. It does not care if the action ended in success or failure
- DAG - directed acyclic graph

# Resource Management for ML workflows

- Schedulers - cron programs that can handle dependencies (you could try Slurm)
- Allows to control the logic if a job fails/succeeds
- Should optimize for:
  - resource utilization
  - jobs to run
  - resources needed for each job
- Answer questions *when* to run jobs and *what* resources to use

# Resource Management for ML workflows

- 
- Orchestrators answer the question *where* to get those resources
  - Orchestrators deal with:
    - Machines
    - Instances
    - Clusters
    - Service-level grouping
    - Replication
    - Etc.
  - Kubernetes (WS's Elastic Kubernetes Service (EKS), Google Kubernetes Engine (GKE) unless you love setting up K8s clusters)
  - Schedulers usually run on top of orchestrators
    - Slurm, Google's Borg, HashiCorp Nomad, K8s

# Tools

- Workflow management tools for data science:
  - Airflow
  - Argo
  - Prefect
  - Kubeflow
  - Metaflow, etc