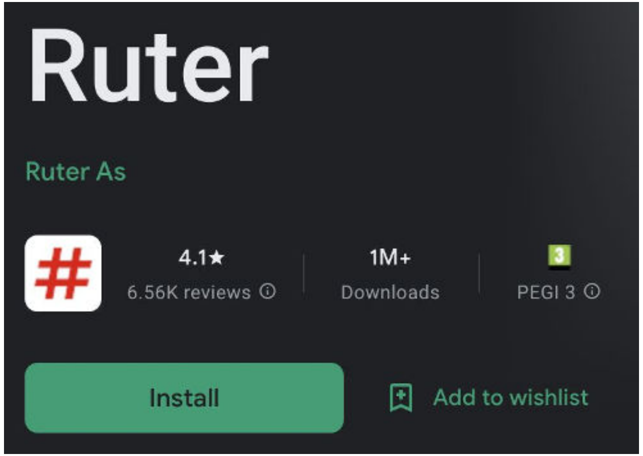
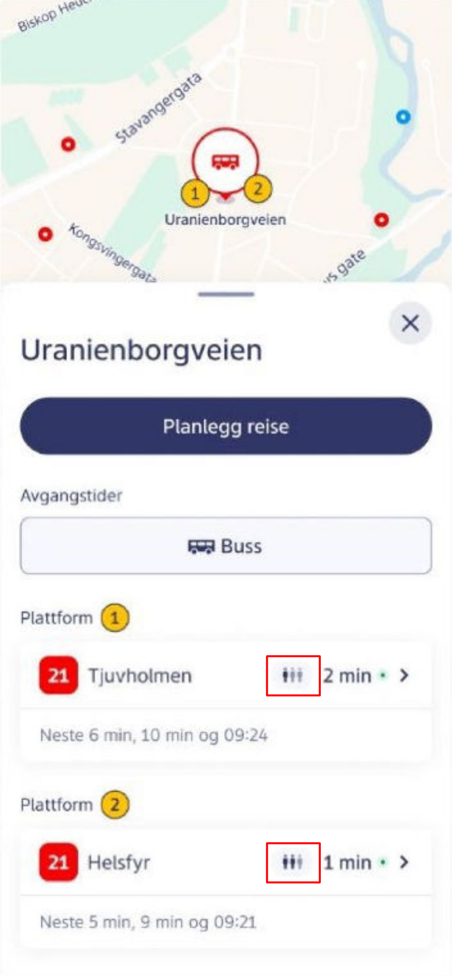
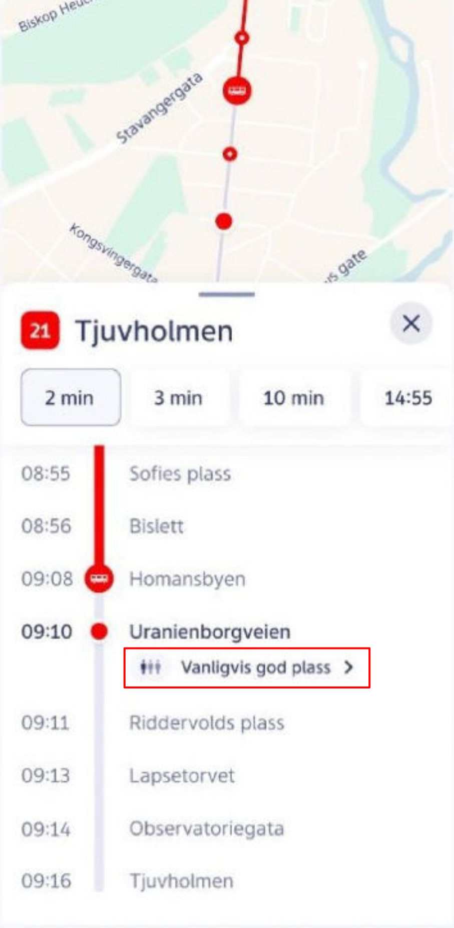
Predicting passenger numbers using machine learning

David Skålid Amundsen


Ruter – Mobility in Oslo/Viken 4+
Plan your trip and buy tickets
Ruter As
#1 in Travel
4.3 • 73.1K Ratings
Free


Ruter
Ruter As
4.1★
6.56K reviews
1M+
Downloads
PEGI 3
Install
Add to wishlist


Stavangergata
Uranienborgveien
Kongsvingergata
Uranienborgveien
Planlegg reise
Avgangstider
Buss
Plattform 1
21 Tjuvholmen 2 min
Neste 6 min, 10 min og 09:24
Plattform 2
21 Helsfyr 1 min
Neste 5 min, 9 min og 09:21


Stavangergata
Kongsvingergata
21 Tjuvholmen
2 min
3 min
10 min
14:55
08:55 Sofies plass
08:56 Bislett
09:08 Homansbyen
09:10 Uranienborgveien
Vanligvis god plass
09:11 Riddervolds plass
09:13 Lapsetorvet
09:14 Observatoriegata
09:16 Tjuvholmen

computas

Sensors count boarding and alighting

- Sensors count boarding and alighting passengers

- Can calculate number of people that was on-board

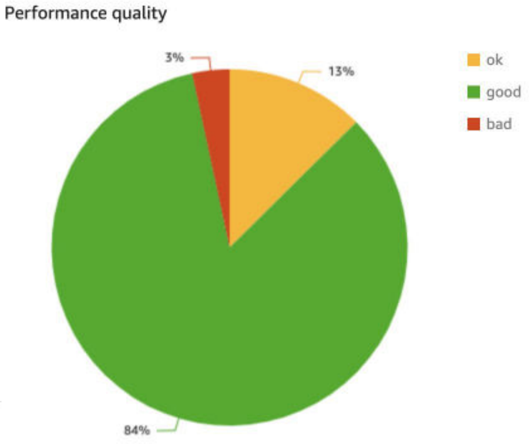
NRK, Parakash

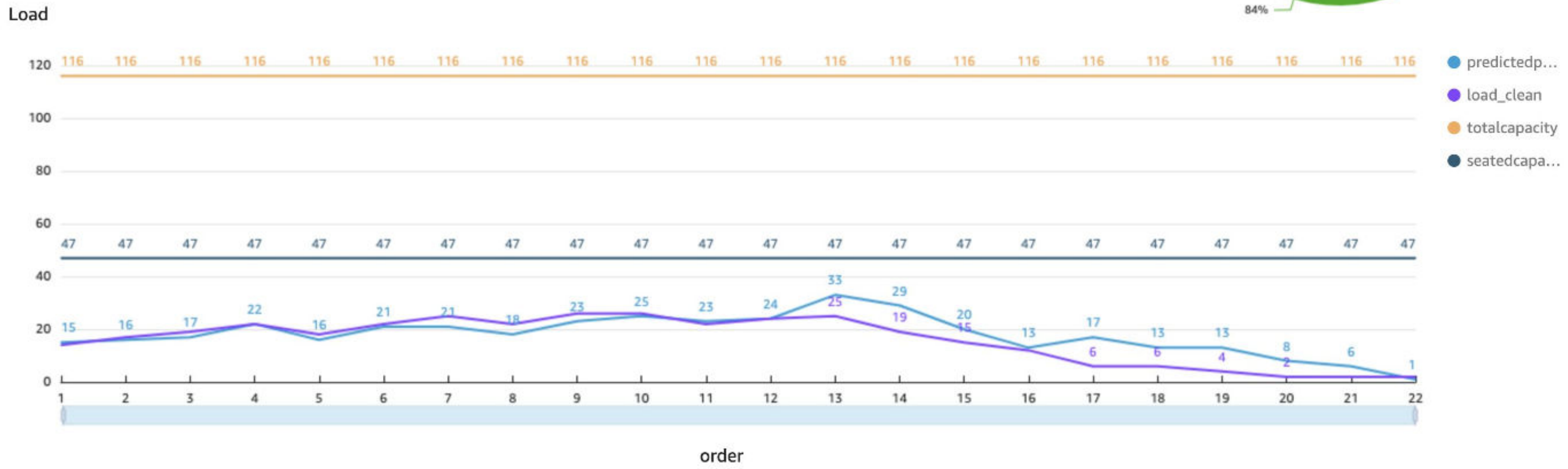
Ruter uses machine learning to predict passenger numbers

- Predict # people onboard
- Use the past three weeks of data
- The model is XGBoost
- Predict for the next three days
- Train and predict every day
- Combined with vehicle capacity
    - Free capacity on vehicle

Predictions are good enough for production

Root mean squared error (RMSE): 5 passengers





Bringing this model into production

First attempt: Provide csv-files to other teams

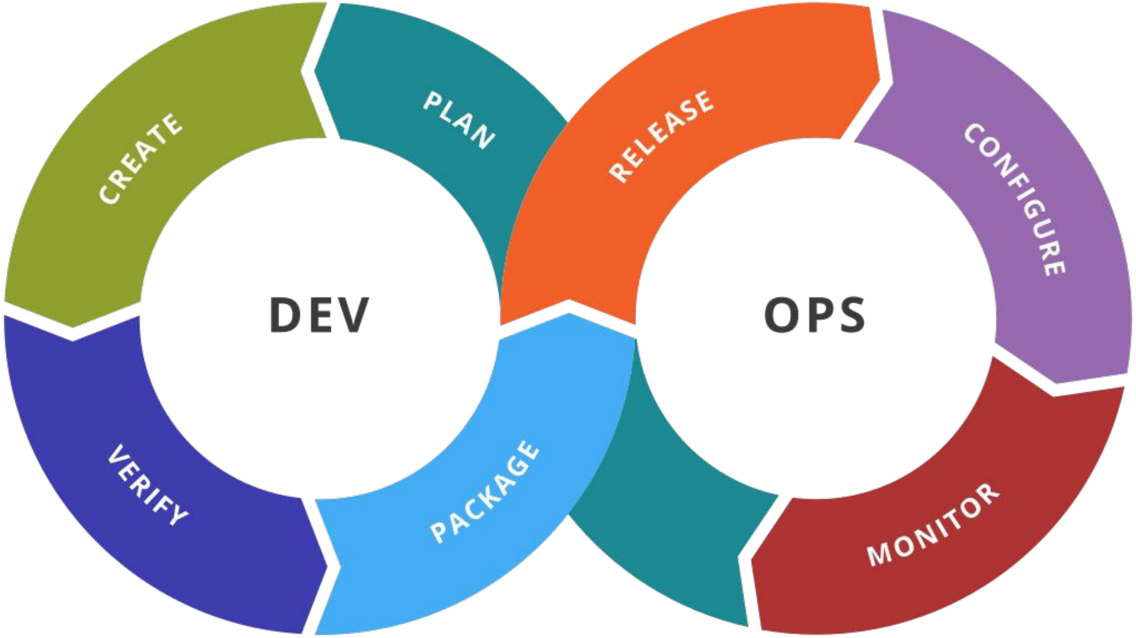
- Daily model training on local machine
- Upload artifacts to git repository

model_2020-08-06.csv	model prediction for 2020-08-06	11 Aug 2020
model_2020-08-07.csv	model prediction for 2020-08-07	11 Aug 2020
model_2020-08-08.csv	model prediction for 2020-08-08	11 Aug 2020
model_2020-08-09.csv	model prediction for 2020-08-09	11 Aug 2020
model_2020-08-10.csv	model prediction for 2020-08-10	11 Aug 2020
model_2020-08-11.csv	model prediction for 2020-08-11	11 Aug 2020
model_2020-08-12.csv	model prediction for 2020-08-12	18 Aug 2020
model_2020-08-13.csv	model prediction for 2020-08-13	18 Aug 2020
model_2020-08-14.csv	model prediction for 2020-08-14	18 Aug 2020
model_2020-08-15.csv	model prediction for 2020-08-15	18 Aug 2020

First attempt was not successful. Why?

- Difficult for the machine learning team to
  - reproduce models
  - efficiently monitor models
  - keeping model up-to-date
- Difficult for other teams to
  - get the predictions
  - understand how to use the predictions
  - trust the predictions
- Need more robust methodology

DevOps enables delivery of applications at high velocity



Wikimedia

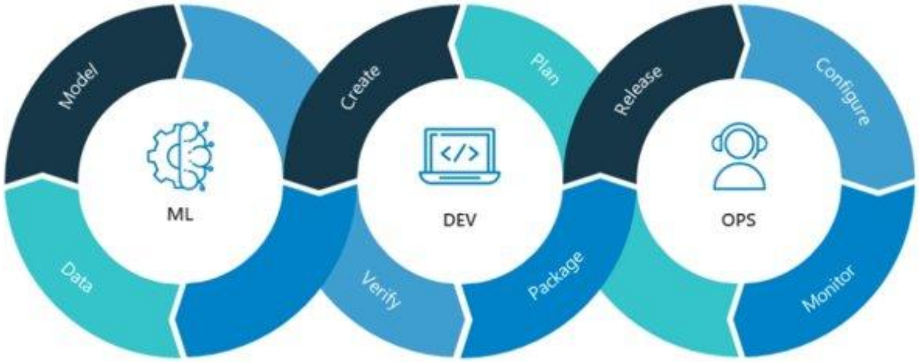
MLOps

An ML engineering culture and practice that aims at unifying ML system development (Dev) and ML system operation (Ops)

Slide by Sina Nek Akhtar, Google

MLOps extends DevOps principles to ML

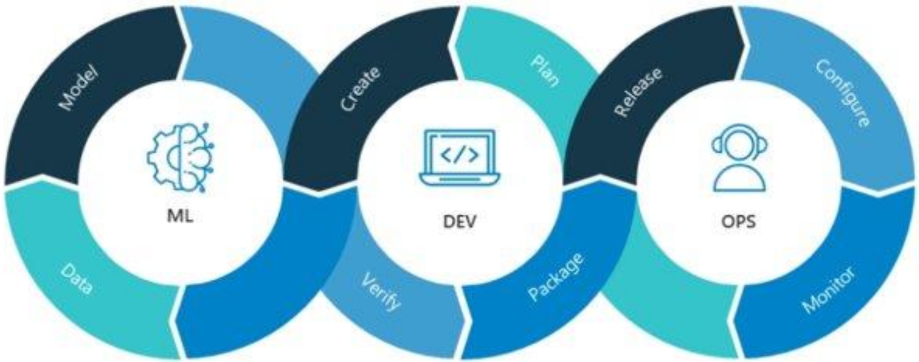
- Start with a quick simple model
- Tracked and reproducible experiments
- Automate as much as possible
  - Automated testing
  - Automated packaging
  - Automated training
- Monitor model and data



Neal Analytics

MLOps has many benefits

- Adapt to changes in the real world:
  - Actively monitor production model quality
  - Frequently retrain
  - Reusable components
- Shorter development cycles
  - => shorter time to market
- Increased
  - reliability
  - performance
  - scalability
  - return on investment for ML projects



Current production setup

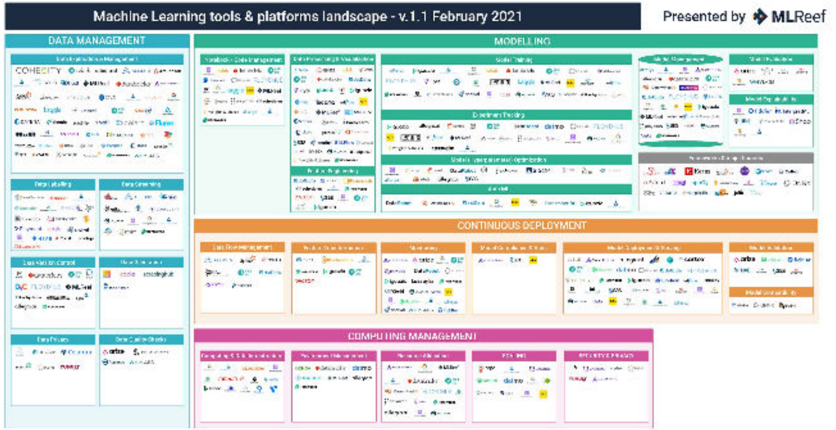
Choosing ML(Ops) infrastructure framework

- Cloud Provider (Ruter's choice):
  - Managed, easy to get started
  - Vendor lock-in, expandability/features missing?
- Frameworks:
  - Flexible
  - Often manage infrastructure yourself, too many tools?
- Make everything from scratch
  - If nothing else fits your need
  - Manage infrastructure, reusability, scalability, maintainability

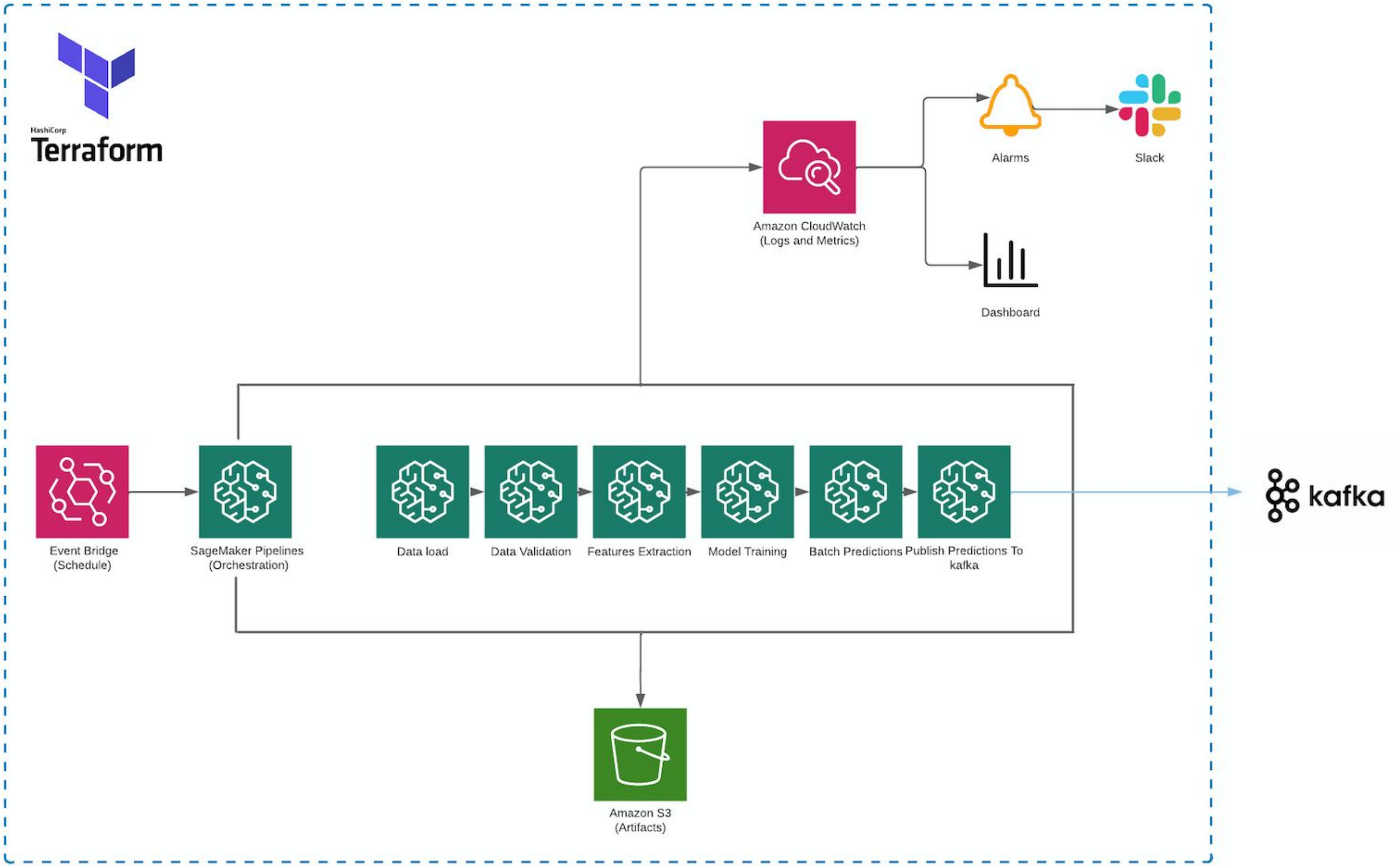
Vertex AI

Azure Machine Learning

Amazon SageMaker



Ruter uses Amazon SageMaker



Each pipeline execution is reproducible

daily-20220525-125206

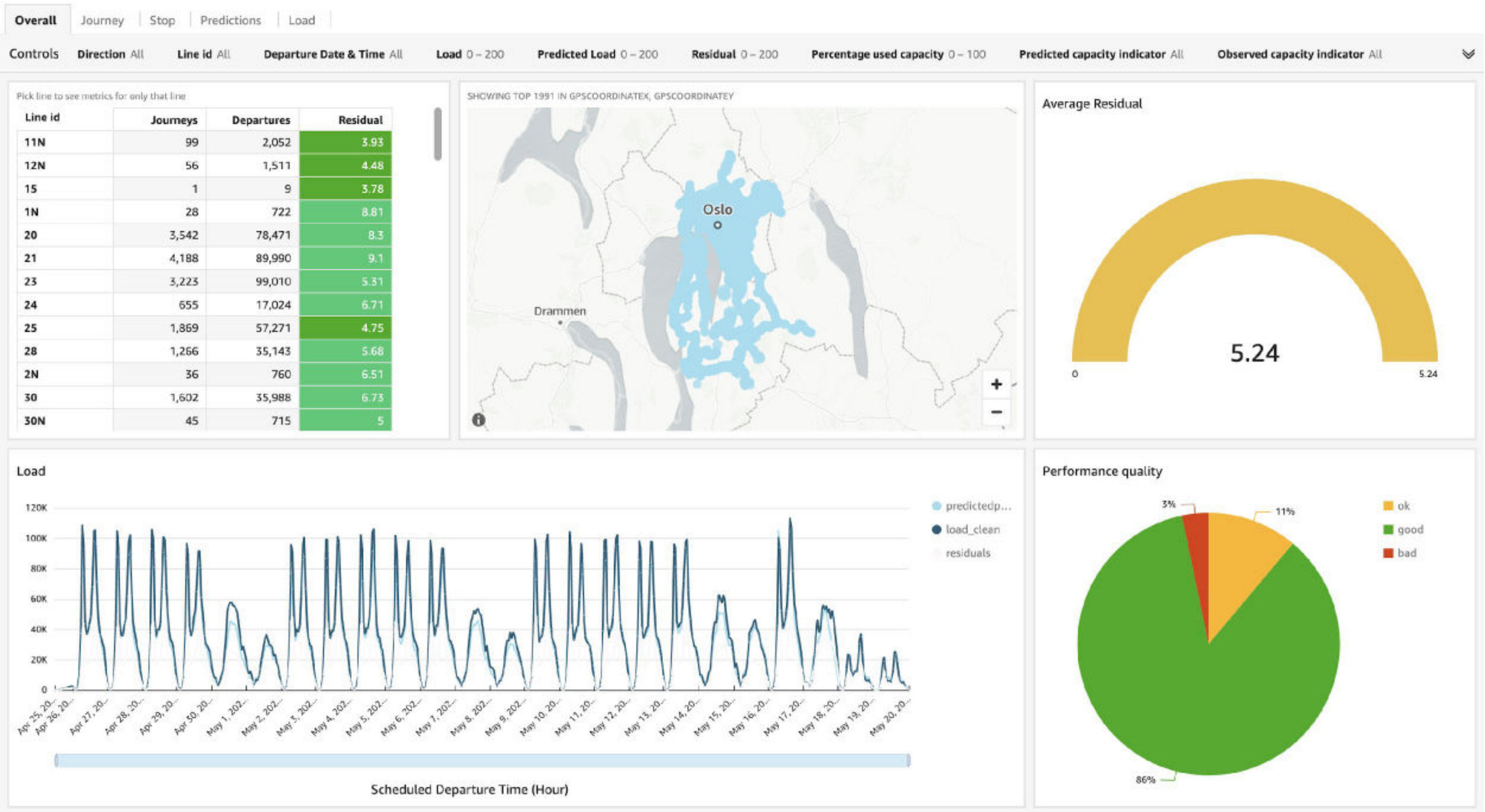
Pipeline triggered automatically

Status

Graph | Parameters | Settings

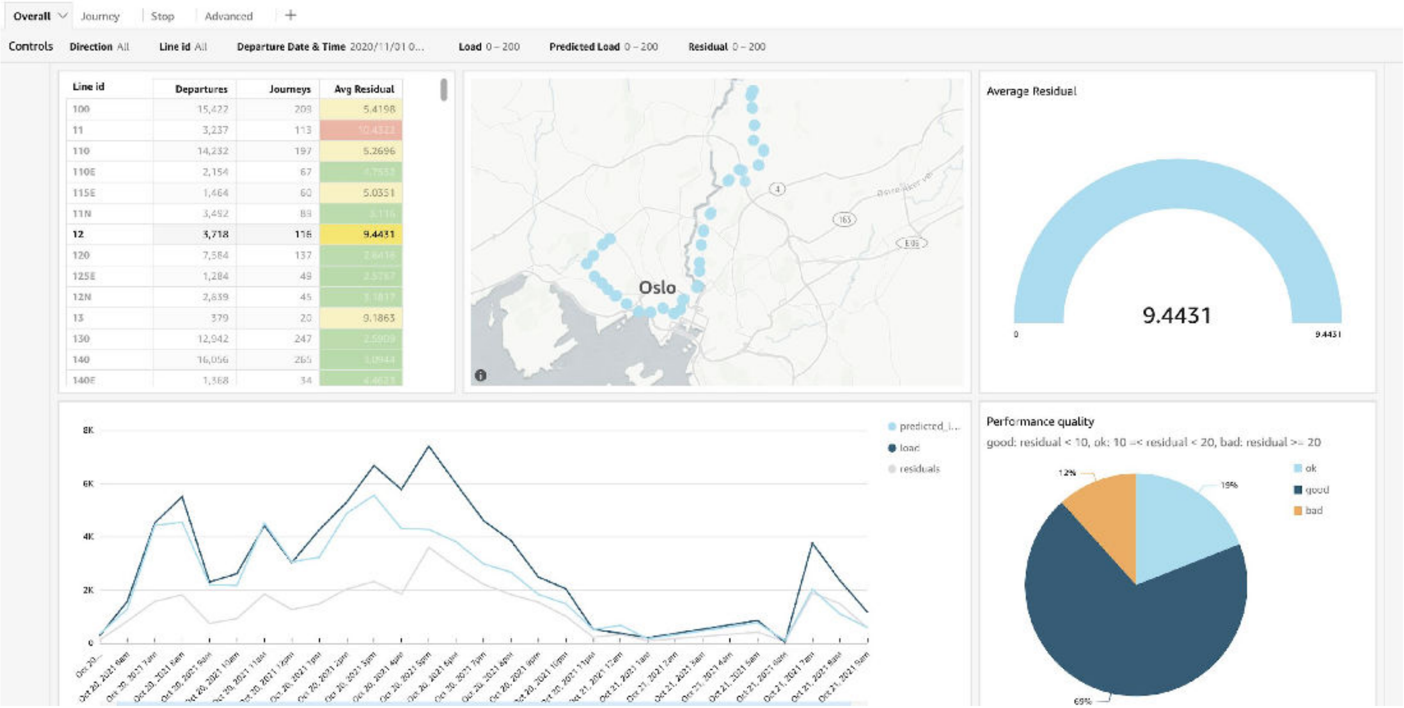
Parameters	Type	Value
TargetDate	Boolean	2022-05-25
DbSyncDatetime	String	2022-05-25T07:00:24.320Z
TriggerMdlMonitorLambda	String	True
PipelineLambdaArn	String	FOR_INTERNAL_USE_ONLY
BatchRunName	String	FOR_INTERNAL_USE_ONLY
BucketPrefix	String	daily
ExecutePublishPredictions	Boolean	true

Model performance is monitored

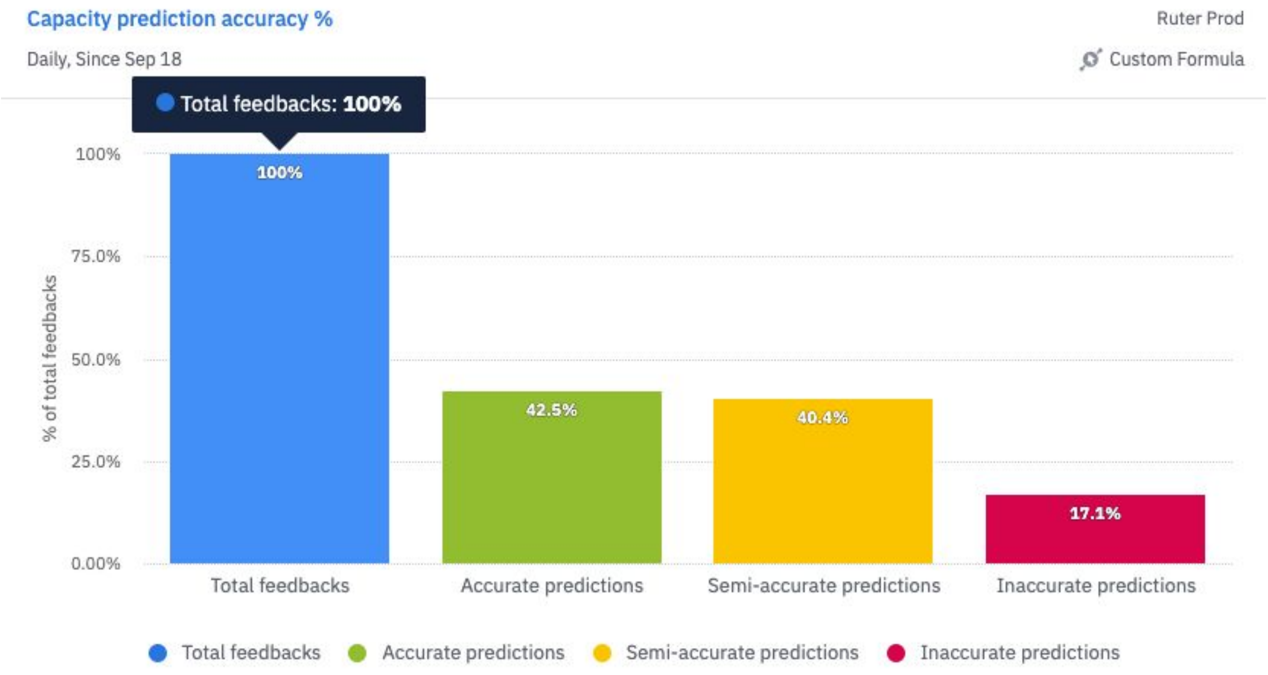


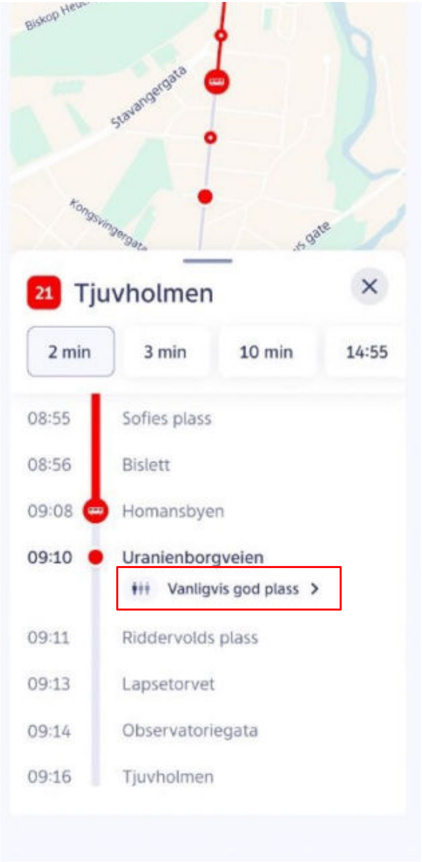
Amazon QuickSight

Model monitoring enables deep-dives



User feedback is also collected





Final thoughts

- Developing ML systems is different from traditional IT systems
  - Data + code -> models
  - MLOps: unify model development and operations
- MLOps frameworks help you productionise ML applications
  - Setting up an MLOps framework for production takes time
  - Choosing MLOps framework involves trade offs
- MLOps is worth it!
- Want to know more? Visit Computas’ stand here at AI+.

Bonus slides

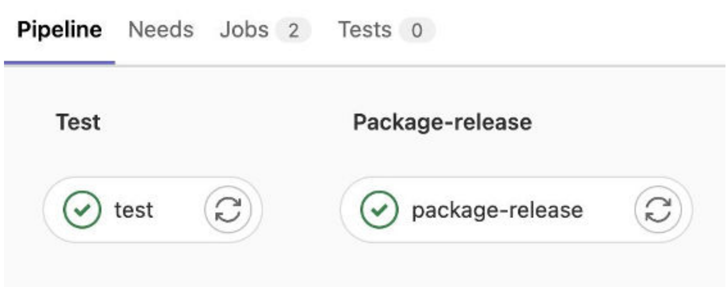
We use (Gitlab) CI for both containers and infrastructure

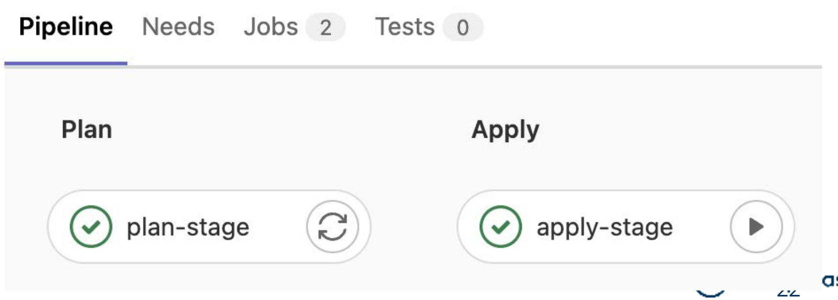
Pipeline steps

- Python
- Containerised
- Versioned
- Automatic testing, building

Infrastructure

- Terraform
- Three interacting environments
- Dev-infrastructure on demand





Where should ML knowledge be placed?

One ML team:

- Tasks:
  - Develop use-cases
  - Data analysis
  - Model development
  - Monitoring
  - Infrastructure
- Consequences:
  - Tightly knit ML team
  - Isolated ML team
  - Friction with product teams
  - Broader tech stack in team

One ML team lends out resources:

- One ML team
- ML resources “lent out” to product teams
- Reduced friction with other teams
- Follow product until the end user
- Does it work?

ML infra team + product teams:

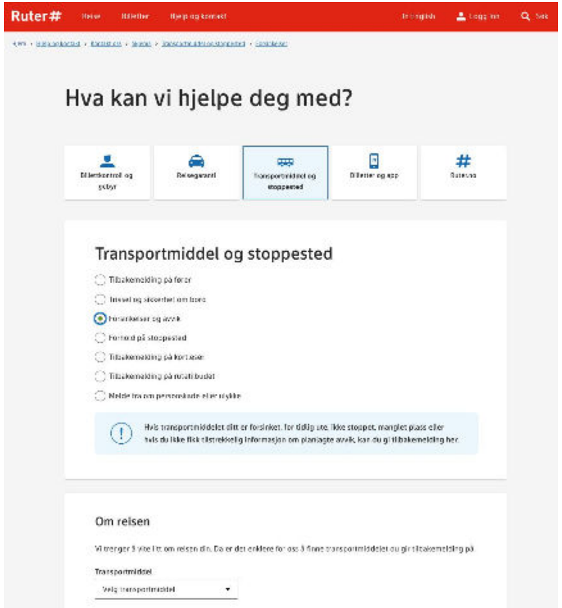
- Central ML infrastructure team
- Data Scientists in product teams
- More data science work for data scientists

ML team per ML domain:

- Cross-enterprise ML applications
- Organise ML teams based on area, e.g.
  - anomaly detection
  - recommendation
  - image processing
- Separate ML infrastructure team?

Production ML at Ruter

- Classification of customer inquiries
  - More efficient customer service
- Predicting key stops
  - Better information on displays in buses
- Predicting crowdedness



Many improvements are possible

- Decouple training and predictions
  - Does the model need to be retrained every day?
- Add automatic integration tests
  - Unit tests are in place
  - Integration tests are manual
- Provide a better experience for model development and feature engineering
  - Focus has been on stable model in production
  - Take advantage of built-in experiment tracking tools?
- Use live data to improve estimates