MLOps — from Prototyping to Production

Gianluca Campanella

6th June 2019

Hello!

My name is **Gianluca** [dʒanˈluːka]

What I do nowadays

I’m a Data Scientist at

Microsoft

in **AzureCAT**

What I do nowadays

I also run my own company

Estimand

that provides

Data Science consulting and training

Today we're talking about…

MLOps

Today we're talking about…

MLOps

($\approx$ a whole bunch of mistakes I made in the last few years)

Things I’ve helped build recently

- High-frequency trading system for sports betting
- Context-aware, personalised search engine
- Content recommender for a mobile app
- Automated forecasting tool for an e-commerce business

Two types of Data Science

Analysis-focused

- Maths and Statistics
- Business Intelligence

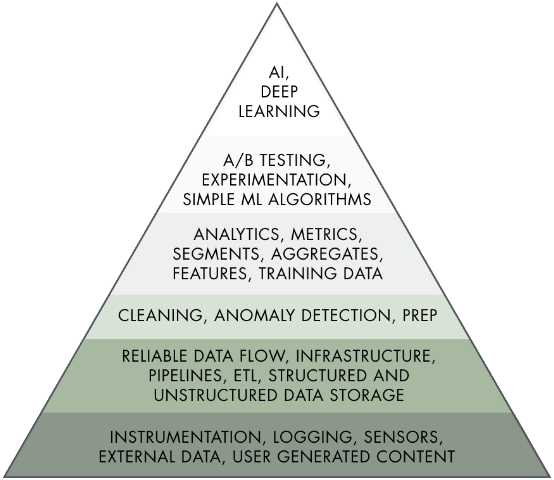
→ Assist human decision-making

Building-focused

- Machine Learning
- Software Engineering

→ Develop and deploy data-driven products

Don’t try to run before you can walk



From M. Rogati

What is MLOps?

What is DevOps?

DevOps

What?

Automation practices between software developers and IT

Why?

To build, test and release software faster and more reliably

DevOps

> *At its essence, DevOps is a culture, a movement, a philosophy.*
>
> — *Atlassian*

MLOps

What?

Automation practices between data scientists, data engineers, software developers and IT

Why?

To build, test, release and monitor software that embeds ML faster and more reliably

Why does this matter?

Why does this matter?

High-risk, high-reward innovation culture

$$\downarrow$$

Iterate quickly $\longleftrightarrow$ Fail fast

Why does this matter?

If it’s not used in production…

It never happened!

Why does this matter?

If *it is* used in production…

It better work!

Why does this matter?

As a Data Scientist, MLOps…

- Is hard but does pay off
- Gives you peace of mind
- Allows you to focus on more interesting tasks

Why does this matter?

As a Software Engineer, MLOps…

- Is something you’re probably already doing
- Increases the dependability of ML systems
- Brings you closer to the Data Science team

Where are we in the DS workflow?

Research question

$\downarrow$

Obtain $\longleftrightarrow$ Explore $\longleftrightarrow$ Model

$\downarrow$

Operationalise

Where are we in the DS workflow?

Research question

$\downarrow$

Obtain $\longleftrightarrow$ Explore $\longleftrightarrow$ Model

$\downarrow$

Operationalise

Where are we in the DS workflow?

Research question

$\downarrow$

Obtain $\longleftrightarrow$ Explore $\longleftrightarrow$ Model

$\downarrow$

Operationalise

Where are we in the DS workflow?

$$\begin{array}{ccccc} & & \text{Data sources} & & \\ & & \downarrow & & \\ \text{ETL} & \longleftrightarrow & \text{Model development} & \longleftrightarrow & \text{Training} \\ & & \downarrow & & \\ & & \text{Deployment} & & \end{array}$$

Where are we in the DS workflow?

Data sources

$\downarrow$

ETL $\longleftrightarrow$ Model development $\longleftrightarrow$ Training

$\downarrow$

Deployment

Data sources

$\downarrow$

ETL $\longleftrightarrow$ Model development $\longleftrightarrow$ Training

$\downarrow$

Deployment

ETL

What?

- **E**xtract, **T**ransform, **L**oad
- Data Science alchemy
- Heavily informed by domain knowledge and EDA

ETL

Things to keep in mind

- Distributional assumptions
- Transformations
- External data sources

Data sources

$\downarrow$

ETL $\longleftrightarrow$ **Model development** $\longleftrightarrow$ Training

$\downarrow$

Deployment

Model development

How?

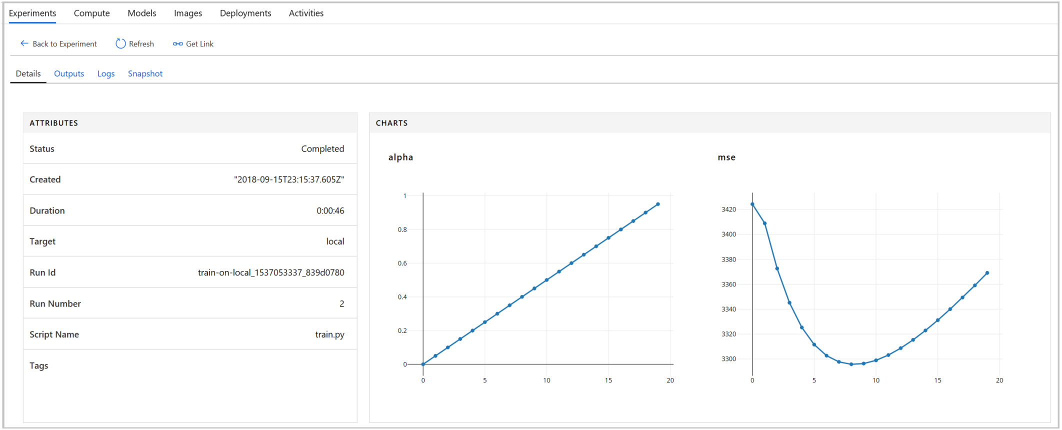
- Hyperparameter tuning
- Automated ML

Model development

Things to keep in mind

- Choice of programming language
- Versioning
- Performance tracking

Model development



Interlude

Online vs offline metrics

- Business metrics $\rightarrow$ online metrics
- Offline metrics $\approx$ online metrics
- Experiment early and often

Interlude

Feedback loops

- Models become self-fulfilling prophecies
- Biased data collection
- Explore-exploit trade-off

Data sources

$\downarrow$

ETL $\longleftrightarrow$ Model development $\longleftrightarrow$ Training

$\downarrow$

Deployment

Training vs scoring

Training

- Historical data $\rightarrow$ model
- Scheduled offline (batch) jobs

Training vs scoring

Training

- Historical data $\rightarrow$ model
- Scheduled offline (batch) jobs

Scoring

- Model + new data $\rightarrow$ predictions
- Online or offline (batch)

Deployment

How?

- Offline (batch) scoring
- Queues
- RPC (e.g. REST endpoint)
- In-process

Deployment

Things to keep in mind

- Throughput and latency requirements
- Impedance mismatch between training and scoring*
- Access control and security
- Other moving parts (e.g. databases)

* I'm looking at you, Apache Spark

Online deployment

How?

- Docker
- Kubernetes
- CI/CD pipeline

Online deployment

Things to keep in mind

- Splitting traffic
- A/B testing
- Incremental roll-out

Data sources

$\downarrow$

ETL $\longleftrightarrow$ Model development $\longleftrightarrow$ Training

$\downarrow$

Deployment

So… we’re done?

Not quite!

We still need to automate ETL and training

Retraining

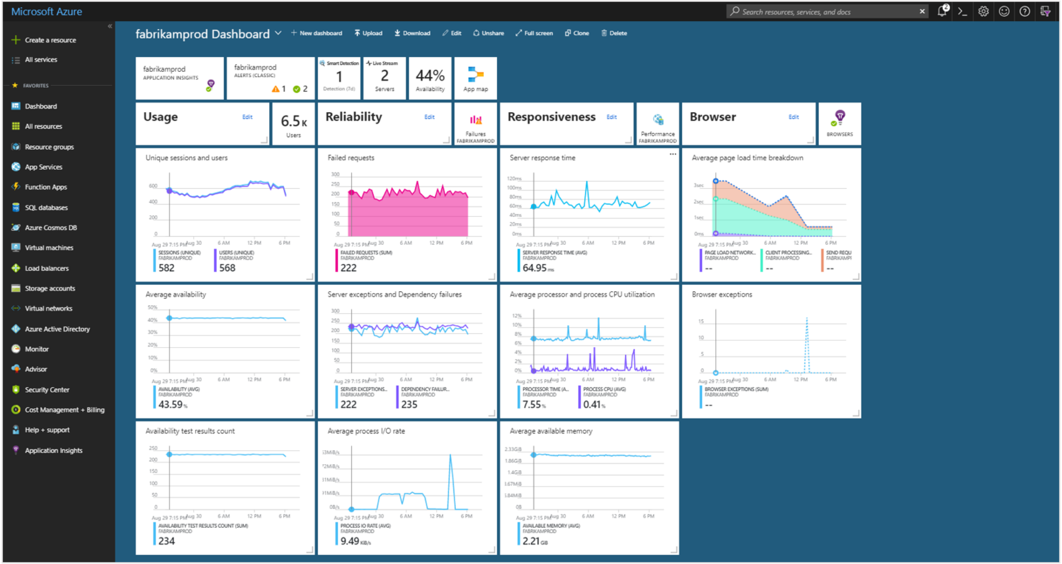
Things to keep in mind

- Data drift
- Performance tracking
- Golden set $\rightarrow$ gated deployment

Monitoring

- Logging
  - Data drift
  - Statistical performance
  - Serving performance
- Anomaly detection and alerting
- Fallback mechanisms

Monitoring



Microsoft Azure
Search resources, services, and docs
Create a resource
All services
FAVORITES
Dashboard
All resources
Resource groups
App Services
Function Apps
SQL databases
Azure Cosmos DB
Virtual machines
Load balancers
Storage accounts
Virtual networks
Azure Active Directory
Monitor
Advisor
Security Center
Cost Management + Billing
Help + support
Application Insights
fabrikamprod Dashboard
New dashboard
Upload
Download
Edit
Unshare
Full screen
Clone
Delete
fabrikamprod
APPLICATION INSIGHTS
fabrikamprod
ALERTS (CLASSIC)
1
2
Smart Detection
1
Detection (7d)
Live Stream
2
Servers
44%
Availability
App map
Usage
Edit
6.5K
Users
Reliability
Edit
Failures
FABRIKAMPROD
Responsiveness
Edit
Performance
FABRIKAMPROD
Browser
Edit
BROWSERS
Unique sessions and users
SESSIONS (UNIQUE)
582
USERS (UNIQUE)
568
Failed requests
FAILED REQUESTS (SUM)
222
Server response time
SERVER RESPONSE TIME (AVG)
64.95 ms
Average page load time breakdown
PAGE LOAD NETWORK...
CLIENT PROCESSING...
SEND REQU
Average availability
AVAILABILITY (AVG)
43.59%
Server exceptions and Dependency failures
SERVER EXCEPTIONS...
222
DEPENDENCY FAILUR...
235
Average processor and process CPU utilization
PROCESSOR TIME (A...
7.55%
PROCESS CPU (AVG)
0.41%
Browser exceptions
BROWSER EXCEPTIONS (SUM)
Availability test results count
AVAILABILITY TEST RESULTS COUNT (SUM)
234
Average process I/O rate
PROCESS IO RATE (AVG)
9.49 KB/s
Average available memory
AVAILABLE MEMORY (AVG)
2.21 GB

Now we're really done!

But then…

Do it again!

- Versioning
- Roll-back mechanisms
- Experimentation

Recap

Data sources

$\downarrow$

ETL $\longleftrightarrow$ Model development $\longleftrightarrow$ (Re)training

$\updownarrow$

Experimentation $\longleftrightarrow$ Deployment $\longleftrightarrow$ Monitoring

Recap

As a Data Scientist…

- Familiarise yourself with the tools
- Try moving some of your workloads away from your laptop
- Understand where Engineering is coming from

Recap

As a Software Engineer…

- Ramp up on containers and orchestration
- Check out the different cloud offerings
- Help your fellow Data Scientists

Thank you!

If you want to keep in touch…

@ gianluca@campanella.org   gcampanella   gcampanella