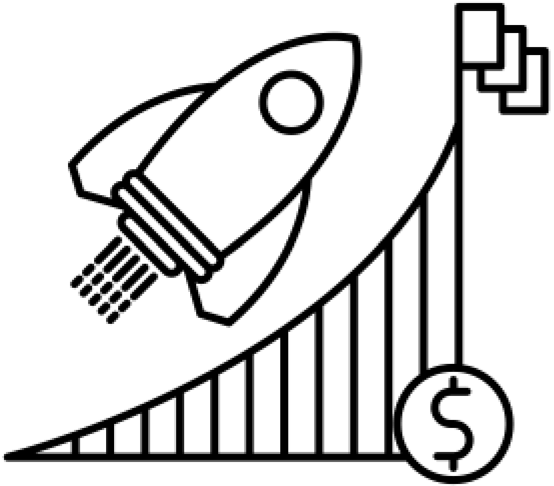


# Career Science

Vincent D.Warmerdam
@fishnets88 - GoDataDriven - koaning.io

# A Talk About Career Things

In this talk I’ll try to give career advice to people who want to do data science in industry.

It will be based on my personal experiences which is from consulting. Some of the statements may not hold so strongly for product companies.

I will also talk about recommenders.

# whois

Senior at GoDataDriven

Worked at a whole bunch of different clients in NL.

GoDataDriven
proudly part of Xebia Group

ebay

KLM

Royal
Flora
Holland

Schiphol
Group

ING

DE PERSGROEP
be part of our passion
PUBLISHING

npo
nederlandse
publieke
omroep

BAKKERSLAND

# whois

Training Partner with Rstudio
Founder of PyData AMS
Blogger at koaning.io

R Studio®

PyData
Amsterdam 2017

koaning.io

# A Talk About Career Things

- Discuss different data roles
- Example of a recommender for these roles
- Example of harder price estimation problem
- Give a long list of career advice
- Give a long list of company advice
- Suggest what the future might look like

# Different Data Roles

# Data Analyst vs. Data Scientist vs. Data Engineer

It is very tempting to call yourself a data scientist just because of the job offers.

So if you can do a bit of SQL and python, can you call yourself a data scientist? Does a data scientist need to do Hadoop?

When are you bluffing? What is the role that a company is looking for?

# **Data Analyst vs.** Data Scientist **vs.** Data Engineer

Analysts scale **knowledge**.

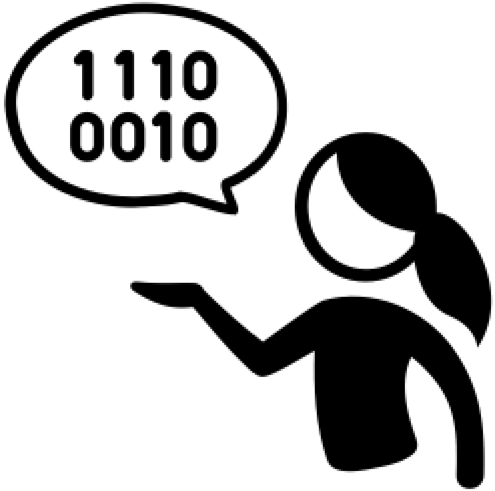
Data Analysts typically work with SQL and charting tools. It can definitely help if they are experienced with statistics and programming, but the main goal remains understanding the data.

# Data Analyst **vs. Data Scientist vs.** Data Engineer

Scientists scale **decisions**.

Scientists want to create algorithms to make better decisions or maybe even automate them. They should not make predictions for things that do not lead to a better decision.

A good example of such an algorithm: predicting time series or a recommendation engine.

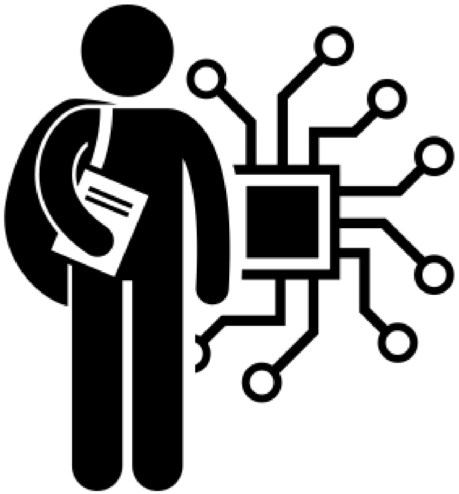


# Data Analyst **vs.** Data Scientist **vs. Data Engineer**

**Data Engineers** scale technology.

To aid analysts and scientists we need engineers who can get infrastructure up and running.

Serving recommendations to millions of people requires more than a script that runs on a laptop. Same thing goes for handling AB test data.



# Data Analyst vs. Data Scientist vs. Data Engineer

In my experience, you want all these roles filled in a team. These people need to be able to talk to each other if you want to create great data products.

Let’s discuss an example product:

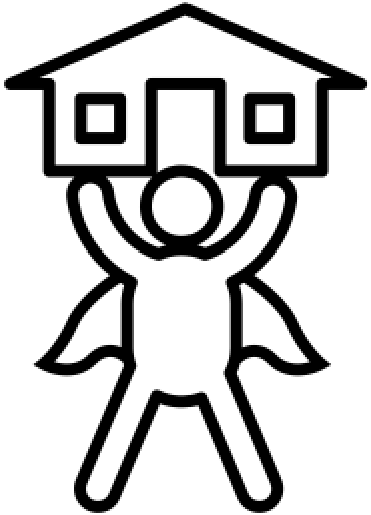
**video recommender service**

# Data Analyst vs. Data Scientist vs. Data Engineer

There is a bit of arrogance around this data science business though. It’s weird because the lines are blurry.

In general I’d much rather have a great analyst on my team than a poor scientist. Analysts are super useful to an organisation but they typically have a different way of adding value.

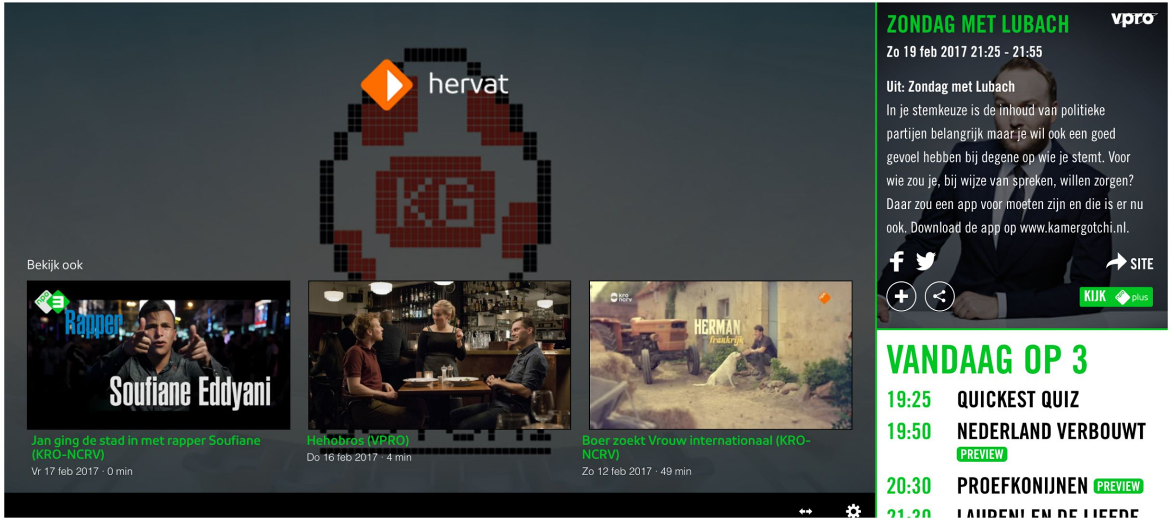
Let’s discuss an example of a typical scientist's task: a recommender!



# Recommender Example

# An example of an algorithm

I was once tasked with making a recommender system over at the Dutch BBC.



Jan ging de stad in met rapper Soufiane (KRO-NCRV)
Vr 17 feb 2017 · 0 min

Hehobros (VPRO)
Do 16 feb 2017 · 4 min

Boer zoekt Vrouw internationaal (KRO-NCRV)
Zo 12 feb 2017 · 49 min

ZONDAG MET LUBACH
Zo 19 feb 2017 21:25 - 21:55

Uit: Zondag met Lubach
In je stemkeuze is de inhoud van politieke partijen belangrijk maar je wil ook een goed gevoel hebben bij degene op wie je stemt. Voor wie zou je, bij wijze van spreken, willen zorgen? Daar zou een app voor moeten zijn en die is er nu ook. Download de app op www.kamergotchi.nl.

VANDAAG OP 3
19:25 QUICKEST QUIZ
19:50 NEDERLAND VERBOUWT PREVIEW
20:30 PROEFKONIJNEN PREVIEW
21:30 LAUREN EN DE LIEFDE

There are three slots that need to be filled. We have logs of user-ids and itemviews. How would you approach it?

# Here was my proposal

$$R_{A\to B} = \frac{p(\text{people saw B given they saw A})}{p(\text{person saw B})}$$

$$R_{A\to B} = \frac{p(A \to B)}{p(B)}$$

# An example of an algorithm

The next part was made with data frames. This can be done with **pandas**, **dplyr**, **spark** or **sql**.

The recommender was done in a few weeks because and it was awesome:

- It was easy to debug.
- It was fast to implement.
- It was easy to explain.
- We could actually write tests.
- We could extend it to user-item.
- It outperformed other algorithms.

$$R_{A \to B} = \frac{p(A \to B)}{p(B)}$$

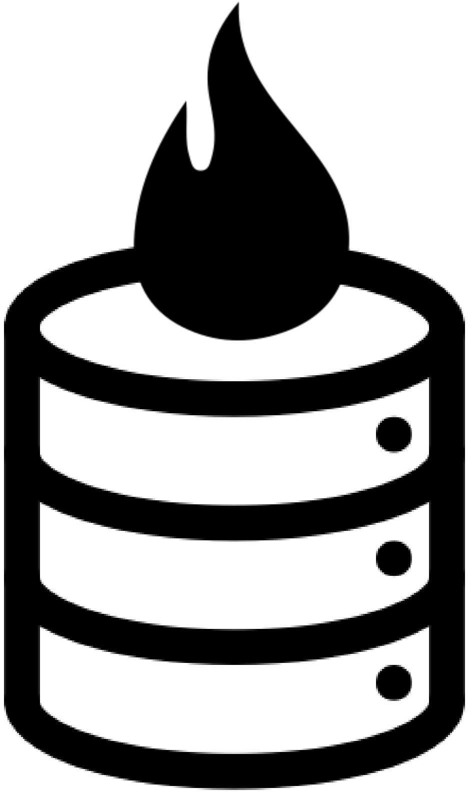
# This was awesome

People were suddenly interested in seeing much more diverse content.

This resulted in our algorithm breaking the caching layer.

mwuhahahahaha

It worked so well that the client needed to purchase additional hardware!



# Another Bonus: Item-Item vs. User-Item

$$R_{A \to B} = \frac{p(\text{people saw B given they saw A}}{p(\text{person saw B})}$$

$$R_{A \to B} = \frac{p(A \to B)}{p(B)}$$

$$R_{A,B,C \to D} = R_{A \to D} \times R_{B \to D} \times R_{C \to D}$$

# Still some problems.

The algorithm is actually still very flawed.

- Cold start
- How fast can we update? Can we do streaming?
- Legal rules and stuff (die hard vs. sesame street)
- We recommend series, how to pick episodes
- There is no decay in there.

The cool thing about not having a black box algorithm, you can fix things like decay with maths:

$$R_{j\to i} = \frac{p(s_i|s_j)}{p(s_i|s_j^{\mathsf{c}})} = \frac{\frac{\sum u_i \cap u_j}{\sum u_j}}{\frac{\sum u_i \cup u_j^{\mathsf{c}}}{\sum u_j^{\mathsf{c}}}} \to \frac{\frac{\sum f(u_i \cap u_j)}{\sum f(u_j)}}{\frac{\sum f(u_i \cup u_j^{\mathsf{c}})}{\sum f(u_j^{\mathsf{c}})}}$$

# Different Roles have Different Worries

Data Analyst => Keep an eye on metrics. Analyse A/B tests.
Understand user behavior+generate A/B tests.

Data Scientist => Can we make the algorithm better?
Can we make the algorithm faster?
Can we make the algorithm streaming?
Fix the cold start part.

Data Engineer => How can we scale all the tech?
How do we serve this?
How can we assign A/B tests + collect?

# Company Advice

# Production

$$U_{\text{early}}(t) = \max(0, 150(t - 0.5))$$

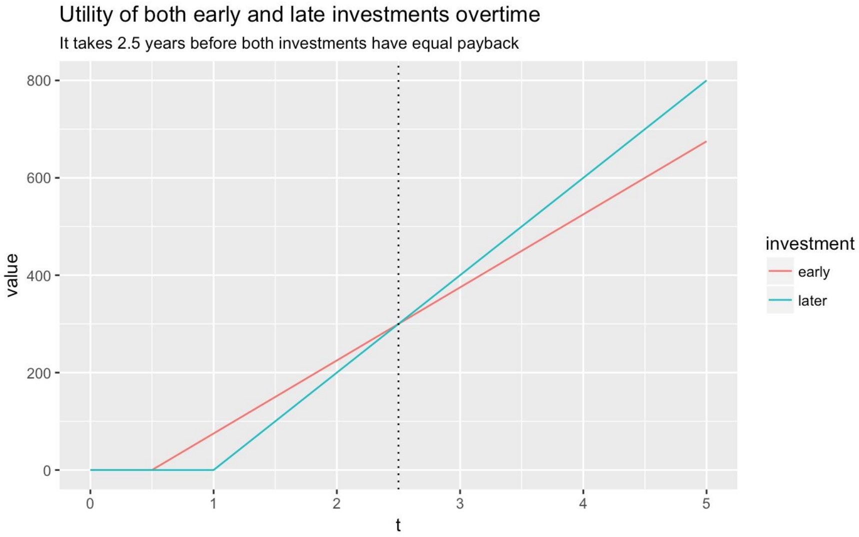
$$U_{\text{later}}(t) = \max(0, 200(t - 1.0))$$

Value from a data scientist usually means that some algorithm needs to run in production.

Pick between:

- 150 daily extra revenue in 6 months
- 200 daily extra revenue in a year

It may be better to have many things in production than it is to bet on one big data science product.

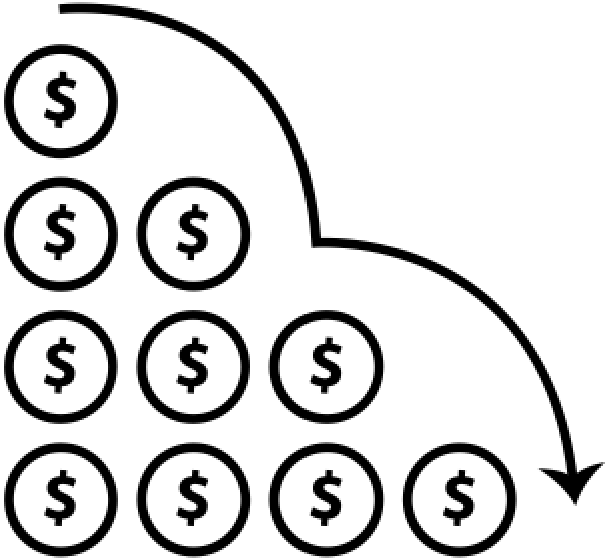


# Marginal Returns

Suppose we've made some recommenders over time;

**v0** of a recommender: +4.1%
**v1** of a recommender: +5.1%
……………………………………....
**v4** of a recommender: +5.7%

At some point it may be better to not spend months on version 5. Maybe we can improve search instead?



# Experiments should be able to Fail

Just like algorithms can get stuck in a local optima, organisations can get stuck on a certain strategy.

If you want to increase revenue, would you ever remove a banner to see what the effect is?

I’ve seen situations where removing the banners actually makes more revenue.

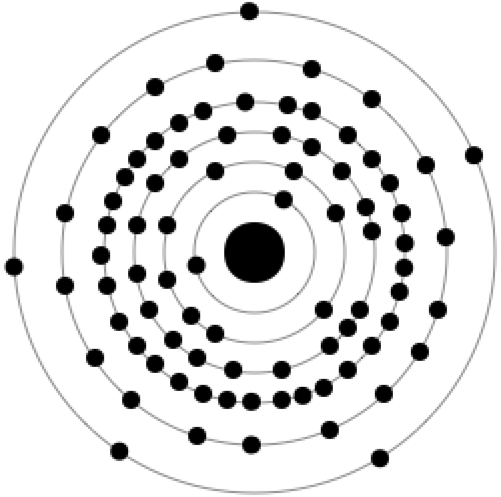
Again; velocity matters.
Always be experimenting!

# Build tools and pipelines

We have this internal deep learning tool we use called **denseryo.**

```
> densoryo grab catdog
> densoryo make-train-test
> densoryo train --name dropout50 --k 5
> densoryo train --name dropout30 --k 5
> tensorboard --logdir=logdir
> densoryo serve --name dropout50
```

This automates the boring stuff and makes you much more productive. We can also run this easily in docker containers.



# Privacy! Don’t be evil! Legal! Legal! Legal!

We can do some scary stuff with facebook data but probably also your company’s data.

Detecting gender is super easy if you have your users with email addresses.

vincentwarmerdam@gmail.com
jessicawarmerdam@gmail.com

Please don’t be evil. Legal will be against you soon too. GDPR.

# Career Advice

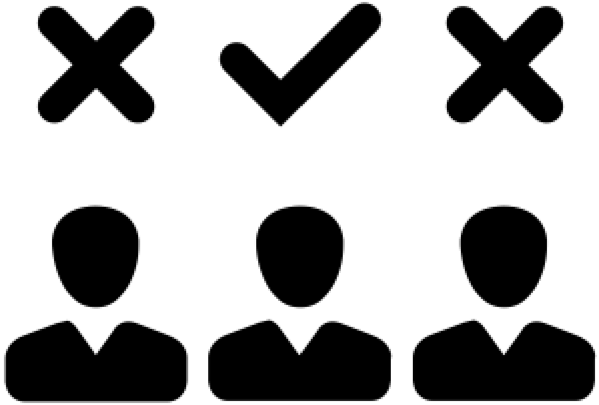
# Projects

Who would you hire?

Phd who wrote complicated papers.

Somebody who took the udacity deep learning course AND the coursera course.

Somebody who built a website that allows you to upload a photo and automatically replaces all faces with a random cat face. You can see that this app works.



# Hiring Data Scientists (1)

There is a difference between book-smart and street-smart. The latter is more important.

Stable code and smart code is more important than a fancy algorithm after all. A data scientist that has never worked with git is a bad sign.

# Hiring Data Scientists (2)

We standardise the process and score candidates via:

1. Maths/Stats
2. Programming
3. Algorithms
4. Communication
5. Creativity

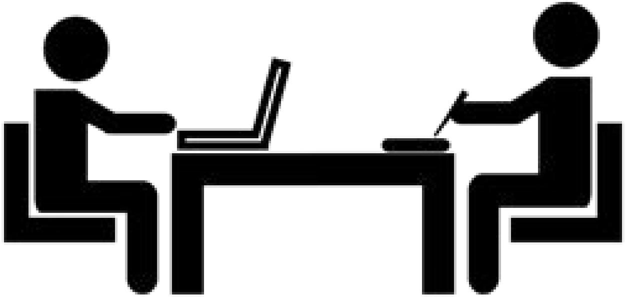
Most people focus on 1 and 3 but often fail on point 5.

# Hiring Data Scientists (3)

The final step of our assessment is a code assignment that takes 16 hours. We pick a relevant task and may be normal for your business to do something similar.

We then have two consultants sit with you and discuss the work. It is a lot of effort but we care about who we hire.

Best advice; we also do our best to keep the people we hire.

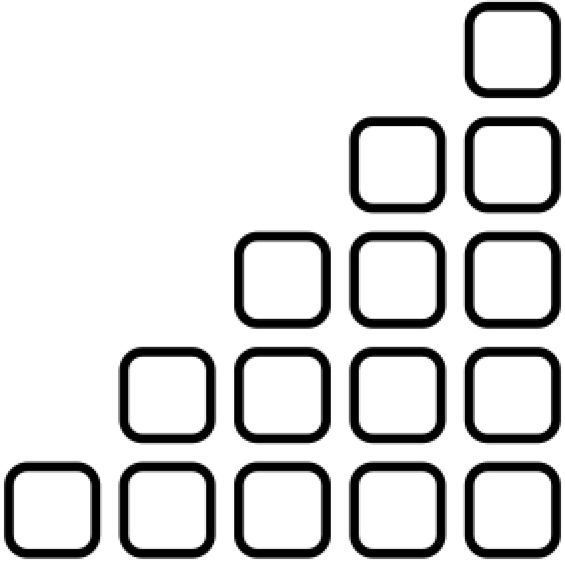


# If you're starting out

Pick a problem and build a resume:

- Get better at a video game
- Get better at a card game
- What is the best house to buy?
- When is the best time to buy a ticket around the world?

Properly coding these problems have been great exercises for me when I started out.

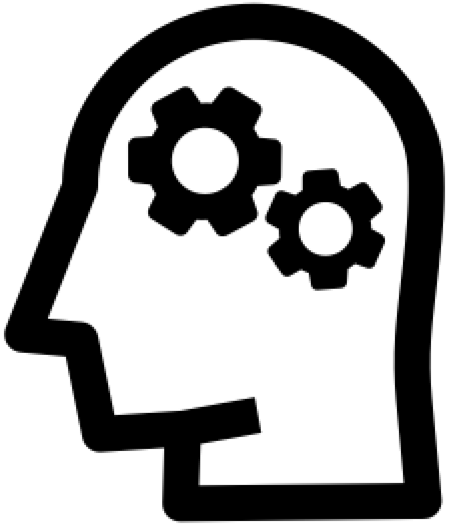


# Knowledge vs. Understanding

“What I cannot create, I do not understand.”
 - Richard Feynmann

Suppose you want to learn deep learning, then you need to do more than read a blog post. Force yourself to get the code working and force yourself to rebuild it from scratch.

Knowing that something works and understanding how something works are two different things.

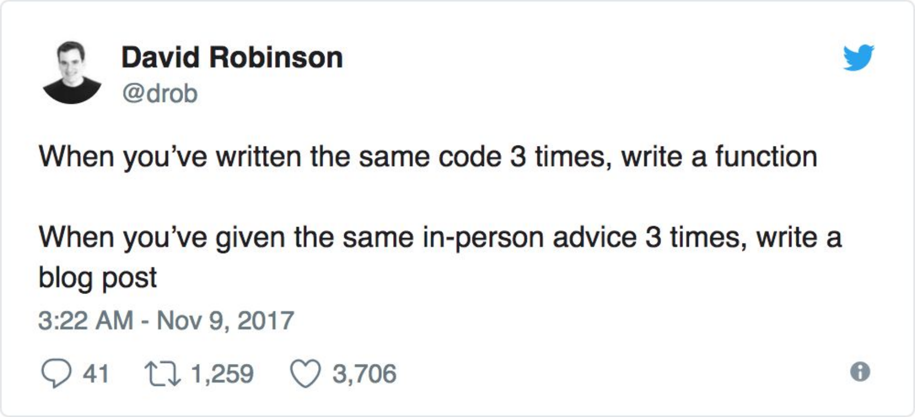


# Blog!

And if you're going to write a tutorial, please turn it into a blog!

Even if you write 2 posts a year, it will make for a great portfolio. If you don't want to set up your own blog consider posting to:

- Medium
- Hackernoon



David Robinson
@drob

When you've written the same code 3 times, write a function

When you've given the same in-person advice 3 times, write a blog post

3:22 AM - Nov 9, 2017

# Future of Tech Communities: Slack?

I met **@tanyacash21** at a conference and she created an international slack for friendly tech conversations.

The idea is super cool because at any point in time you can ask a relevant question to random scientist or developer.

She even organises online mini conferences, it is a cool experiment.

Conf -> Meetup -> Forum?

# Beware the Burn

I have a lot of activities.

I’ve had moments where I’ve actually had **too many**. Be careful about this, very careful.

- Do a sport.
- Eat healthy.
- Talk to friends.
- Have a few hobbies.
- Optimise for joy.
- Turn off that screen.
- Own a cat. They help with that last one.

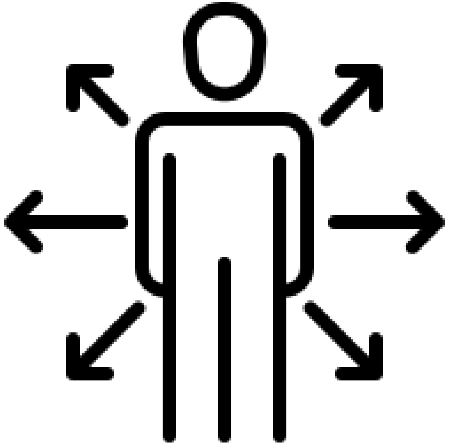
# The Future

# The Future

We are still in a bit of hype but things are calming down a bit.

In part, I expect that parts of the data science role will become more of a **commodity**, much like a mobile developer or a data analyst is nowadays.

Knowing just about algorithms probably won’t be enough in the next few years.



# The Future

Commodity part of the system will be:

```
model.fit()
model.predict()
model.score()
```

This is hard part:

```
.clean_data()
.get_data(realtime=True)
.frame_problem()
.debug(clarity=True)
.weld_to_production_systems()
.monitor_results()
```

# The Future

Having said that, some of the simple things aren’t even solved!

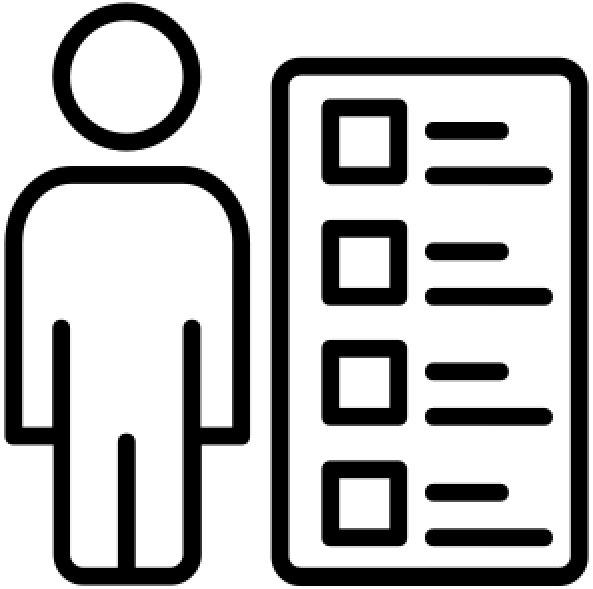
A/B testing is an unsolved problem!
Monitoring DS Algorithms is an unsolved problem!
Explaining models is an unsolved problem!
Quantifying uncertainty is an unsolved problem!
Missing data at inference is an (unsolved) problem!

**My Main Problem**
Maintaining sanity with different data sources, schedulers, bugs, logs and technical ML debt.

# The Future

I would focus a bit on programming/ engineering skills for career planning. Be able train machine learning in the cloud on very large datasets, understand how databases could fail, learn about CI. That stuff will be more and more useful.

A lot of the rest is creativity, domain knowledge and common sense. This is done by hacking at very different types of problems and perhaps some senior supervision.



# The Future

Streaming is going to be a thing.

Can we update a recommender after every mouse click? There are definately use-cases for it but this does not fit the train/test paradigm anymore. There's also algorithms for it but to get them to work it helps if you are a bit creative with the maths.

**Observation:**
If you've solved machine learning in streaming you've also solved it in batch.

# The Future for Companies

New tech appears all the time and you want to keep the rare employees happy.

- Use open source software and spend the budget on education
- Allow two fridays a month for crazy experiments or new tech or OS work
- Automate the boring stuff, the hard part will be sanity behind all the complexity of it all
- Facilitate data employees, don’t command them or they will leave

# Best advice I ever got.

Never let your school get in the way of your education.

Optimise for joy.

Wear sunscreen (google it).