Building Successful Data Science Projects

PyDataBudapest 2022 Keynote

Ian Ozsvald

@IanOzsvald – ianozsvald.com

New projects – pains & gains

Ricardo Pinto • 10:33 PM

I know this is a bit late but I really liked how you talked about the stepping stones that made a successful project in your past newsletters. It would be cool to see the inverse as well, that is what made a project fail.

[On "making $1M for a client" and speeding-up Dask]()

Thoughts - on "making $1M for a client" and speeding-up Dask

Introductions

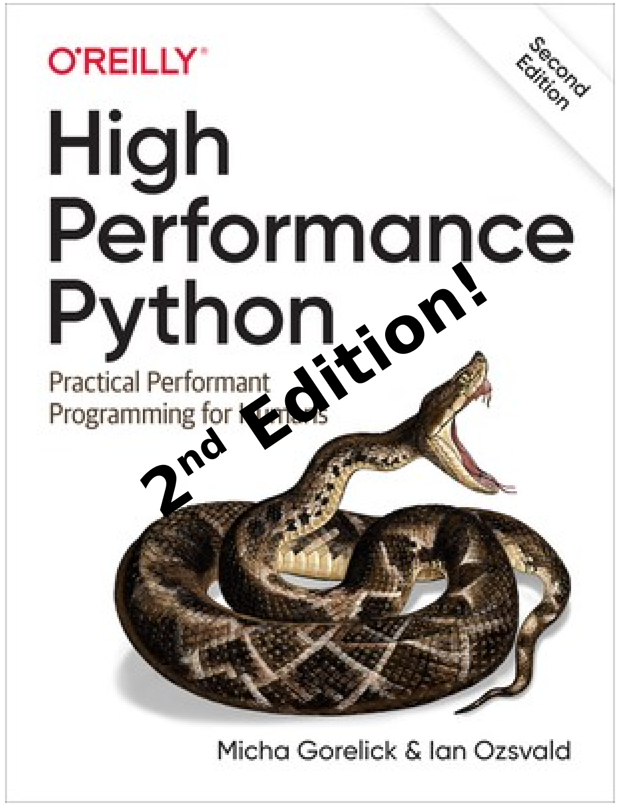
- Interim Chief Data Scientist
- 20+ years experience
- Coaching & public courses
  - I'm sharing from my Successful Data Science Projects course

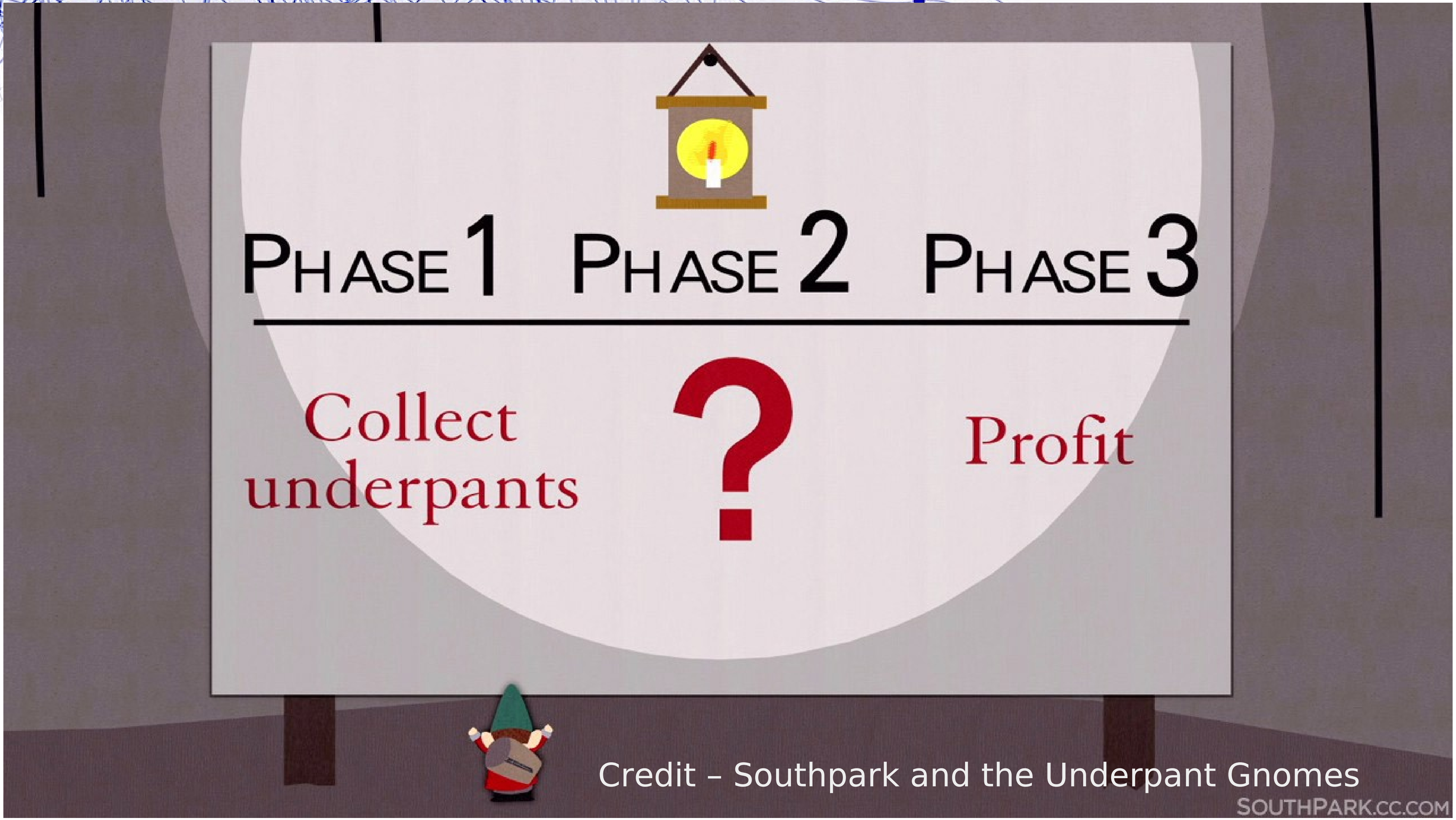
Part of **PyData – 180 groups**

PyData London Meetup

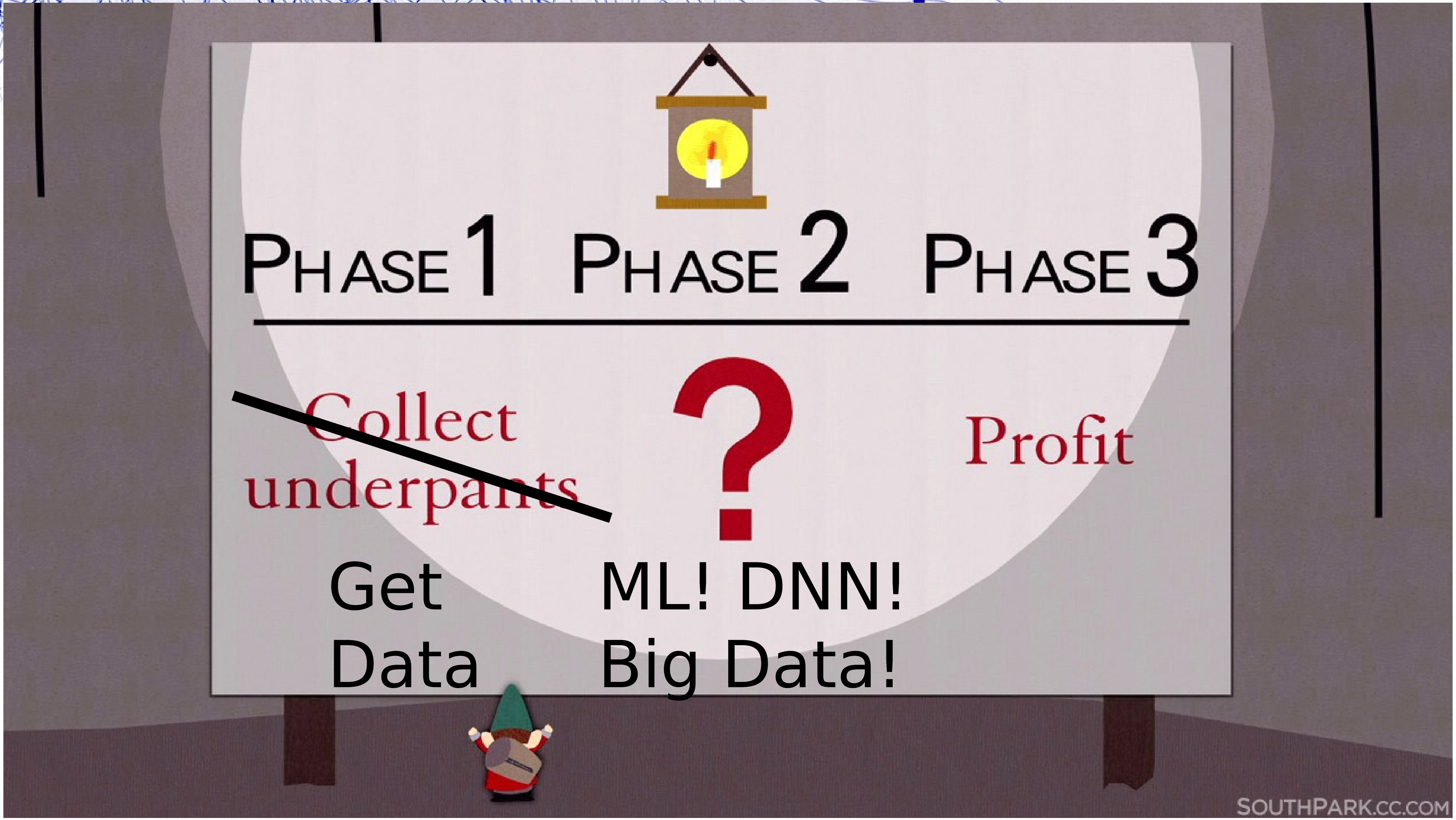
London, United Kingdom

11,107 members · Public group





Credit – Southpark and the Underpant Gnomes


PHASE 1
PHASE 2
PHASE 3
~~Collect underpants~~
?
Profit
Get Data
ML! DNN! Big Data!
SOUTHPARK.CC.COM

Story – Automated price comparison

Samsung AU8000 43 Inch Smart TV (2021) - Crystal 4K AirSlim Smart TV with HDR10+, Built in Alexa, Dynamic Crystal Colour, Adaptive Sound, Motion Xcelerator, Samsung Q-Symphony Audio -...

Samsung UE43AU8000 (2021) HDR 4K Ultra HD Smart TV, 43 inch with TVPlus, Black £319.00 ← Best price not on Amazon...

- Find “cheapest TV” on other sites (famous at the time)
- We agreed the specification verbally
- Sklearn, BoW model, gold validation set – all sensible
- What could go wrong?

Story – Automated price comparison

- The specification changed *despite having agreement*
- They held back the “hard data” so I could have an easy start
- This is not what we discussed

Solution – write a specification

- What problem needs solving? What examples do you have? What is it worth to the business?
- How would an expert solve this? Do they solve it?
- Get the bosses to agree to your specification

Specification:

Table of Contents

Story – insurance and Big DS Projects

- Boss in new department wanted $$$ Big Success
- “Success” was sold to business departments, then the Data Science team were involved *after agreement*
- We got to find out if there was even data in a database
- Sometimes it was just on paper

Solution – talk to the client first

- Your client knows more than you do
- What do they *need*?
- What's *feasible with the data*?
- What's it *$worth*?

Data Maturity Model

Building up an AI Center of Excellence in an Energy Utility

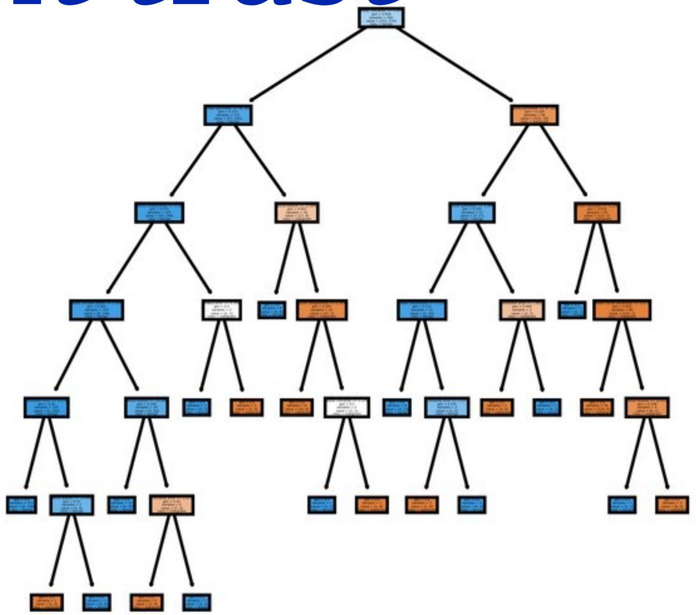
Rachel Berryman
Deputy Head of AI Center of Excellence
50Hertz Transmission

Reference: https://www.svds.com/thought-leadership/data-maturity-assessment/

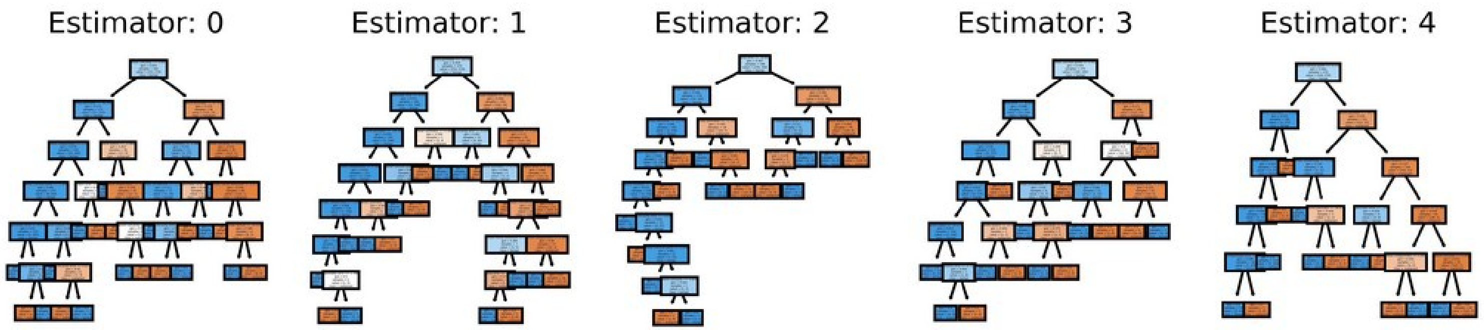
Story – VC and Dirty Data

- “We want to investigate our data, please do magic in 6 weeks and impress us”
- Agreed a derisking project, identified many issues, proposed next project – a sane start
- Later we built models to prioritise interesting companies

Story – insurance & low client trust



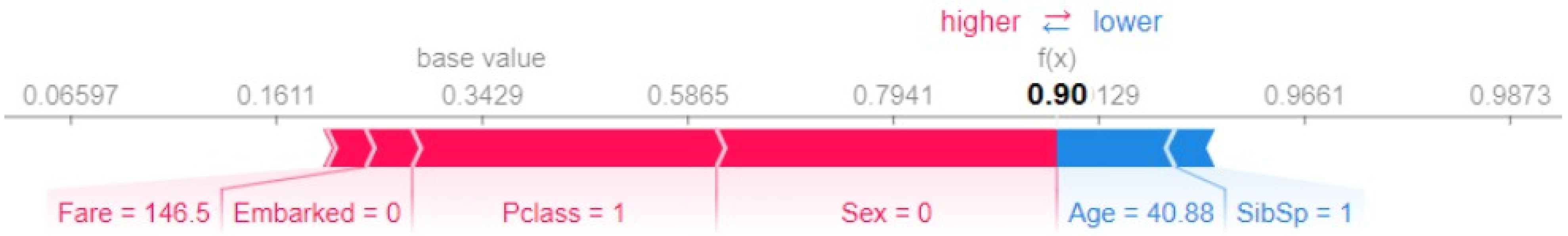
- ML Project nearly finished...
- Client didn't trust "ML"
- Colleague drew many diagrams...



https://stackoverflow.com/questions/40155128/plot-trees-for-a-random-forest-in-python-with-scikit-learn

Story – SHAP to explain predictions

- We found data errors → iteration → build confidence
- The client ultimately agreed “this is useful, I want it” by diagnosing cases *they knew personally*



https://towardsdatascience.com/using-model-interpretation-with-shap-to-understand-what-happened-in-the-titanic-1dd42ef41888

Story – Always have a baseline

- Insurance – the “mean model” beats Random Fr. – huge embarrassment
- VC – My Logistic Regression beat the human rules (encoded as derived sklearn estimators)

```python
class TemplateClassifier(BaseEstimator, ClassifierMixin):

    def __init__(self, demo_param='demo'):
        self.demo_param = demo_param

    def fit(self, X, y):

        # Check that X and y have correct shape
        X, y = check_X_y(X, y)
        # Store the classes seen during fit
        self.classes_ = unique_labels(y)

        self.X_ = X
        self.y_ = y
        # Return the classifier
        return self

    def predict(self, X):
```

https://scikit-learn.org/stable/developers/develop.html

Story – VC and “nobody to check the results”

Best Algorithm for Tabular/Business Data: Sorry, it's not deep learning

Pafka Szilárd, PhD
Chief Scientist
Epoch (USA)

- 10/10,000 chance of success
- The junior associates have their own methods
- They won’t risk time on “crazy Ian’s ML”
- Client suggested “more advanced methods” but Log.Reg. and GBMs very good (accepting limited signal!)

Solution – get clients involved early

- Deliver early and often to client
- Give them enough so *they look cool*
- Use simplest models (e.g. linear), make lots of pictures, diagnose problems, figure out the *value to them*

Story – automated contract recruitment and “new superpowers”

- Need “the face fits” and “relevant skills”
- Similarity tool for company and skills from PDF text
- Client annotated data & scored results from week 1
- “You’ve **given us a superpower**, we phone the top 10 results, sign a contract, then we’re done for the day”

Story – insurance and “no ML, please write SQL”

- Successful Random Forest model for insurance total-loss prediction
- **“We can’t deploy Python, please write SQL”**
- Colleague had to hand-write SQL rules from RF model – did it ever actually work? Was it right?

Solution – plan for deployment early

- **Operationalization is often hard** (especially for v1)
- In your specification think about the client, their needs and how to deploy so they can use the tools
- Sit with the client – *how do they work right now?*
- A corporate might take 6 months to provision a machine

Story – recruitment & deployment

- Initial deployment – CSV for similarity results, then Jupyter Notebooks, then microservices + Flask with black-box tests (now I'd use FastAPI + Streamlit or Viola)
- Boss sat next to me and we typed examples together
- Tests caught MongoDB corruption and MySQL "3 byte unicode"

10.9.2 The utf8mb3 Character Set (3-Byte UTF-8 Unicode Encoding)

Historically, MySQL has used `utf8` as an alias for `utf8mb3`

Story – Making $1M for my client

- Finding insurance fraud and overbilling – really hard!
- Prior fraud project 6 months old & no results
- We **derisked projects early** – 2+ months of discussion
- Found positive examples, **assigned $value**, prioritised
- Agreed a **delivery schedule**

Story – Making $1M for my client

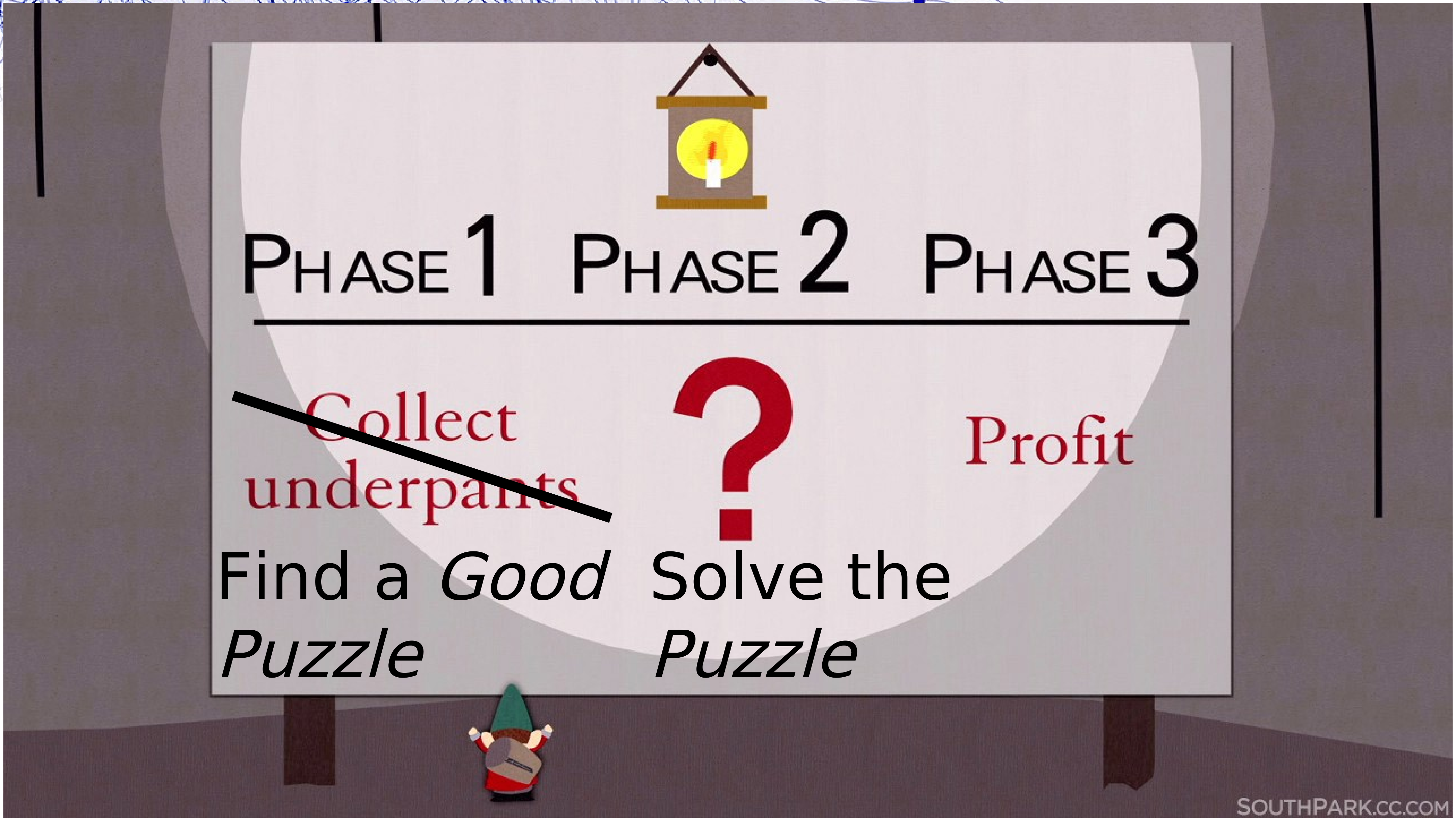
- Mix of better SQL ($0.4M), counting ($0.8M), percentiles ($0.4M), lots of discussion, lots of SQL (**problem rich!**)
- Isolation Forest + GBM good but rules better for client
- Boss' boss writing their own BI as they're so inspired
- New team begging us to start with them

Story – Making $1M for my client

- **New problem!**
- No bandwidth in Fraud team for new results – we swamped them (in a good way)!
- Getting an organisation to move up the Data Maturity Model is hard and just takes time

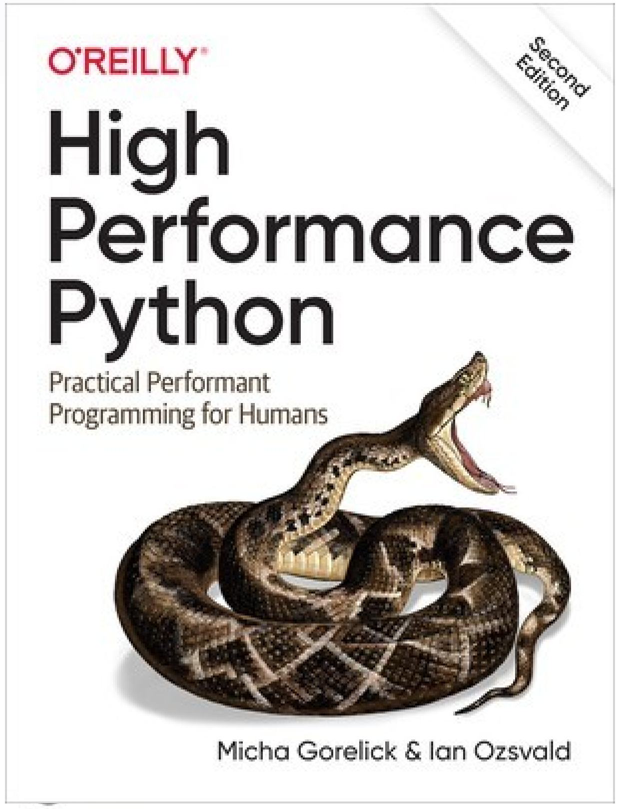
A colleague's view

Some things helped in the past:
1) **Set expectations of what good looks like** e.g for a classifier get 5 experts to label same data and show they agree in 80% of cases
2) **Show context** - map of different types of project on a grid of expected accuracy/outcome/value and where ours would fit
3) **Is it a solved problem?** Got internal data? Why not use API?
4) **Direct benefit estimate** e.g. if we detect further 20 cases of X and prevent y, what's it worth to the business?
5) **Human in the loop** - share result with human expert for final decision
- *Elena Nemtseva (private communication, with permission, thanks!)*


PHASE 1 PHASE 2 PHASE 3
~~Collect underpants~~
?
Profit
Find a *Good* *Puzzle*
Solve the *Puzzle*
SOUTHPARK.CC.COM

Summary

- Solve the whole puzzle & deliver value
- **NotANumber.email**
  A Pythonic Data Science Newsletter
- See blog for my classes + **many past talks**
- I'd **love a postcard** if you learned something new!



Thanking @heatherscarlettrose for a post-public-talk Thank You card, these are always much appreciated!

You’re in charge

- This is *your career – you’re in charge*
- Identify possible problems
- Make sensible choices
- (accept some failures!)
- Enjoy yourself

A checklist for you

- You should write your own **specification**
- Identify risks, **talk to the experts**, get good examples
- Quickly **deliver results** & iterate
- **Deploy often**, deploy early (be embarrassed and learn)