Using Machine Learning in the delivery of ads

Ruth Garcia

NDR Conference – June 07, 2018

Iasi, Romania

What is Skyscanner?

Skyscanner is a leading **global travel search site** offering a comprehensive and **free** flight search service as well as online comparisons for hotels, car hire and **now trains.**

Data Science at Skyscanner

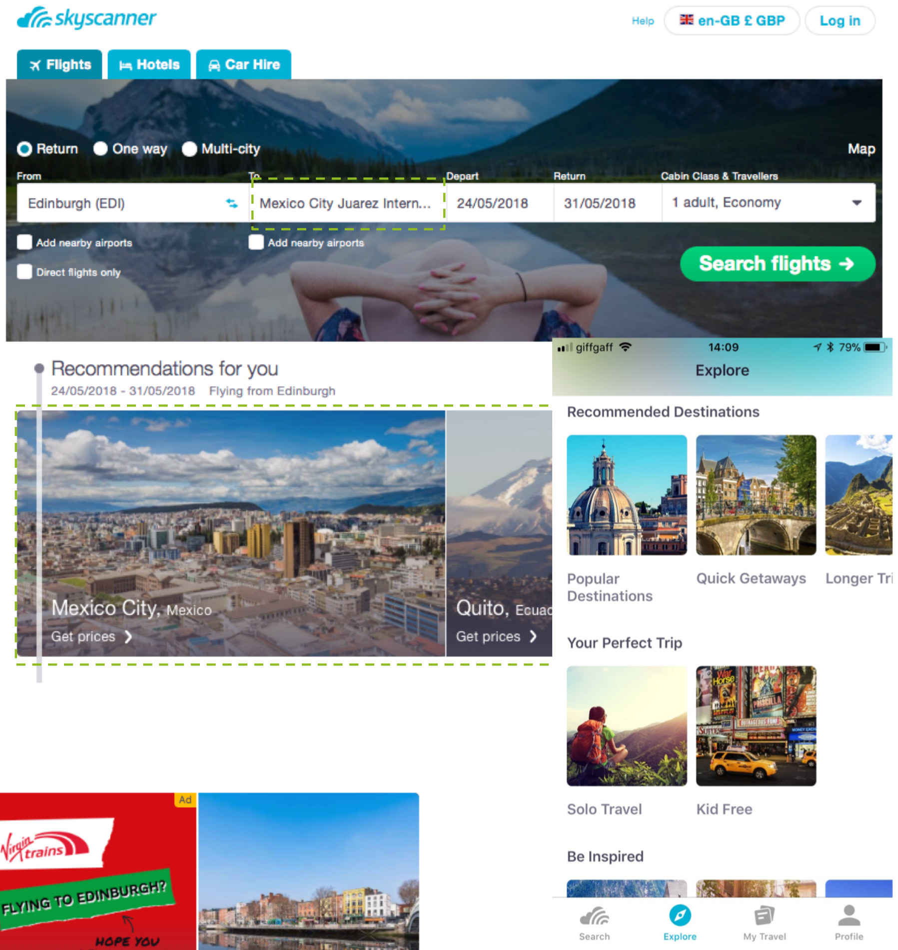
1. **Decision Science:** Ensuring we have the right data; insights and information to make the most impactful, scientific decisions in every aspect of our operations.

2. **Building Data Products:** Leveraging our vast wealth of data to build more contextual, relevant products for Travellers and Travel Suppliers (our Partners)

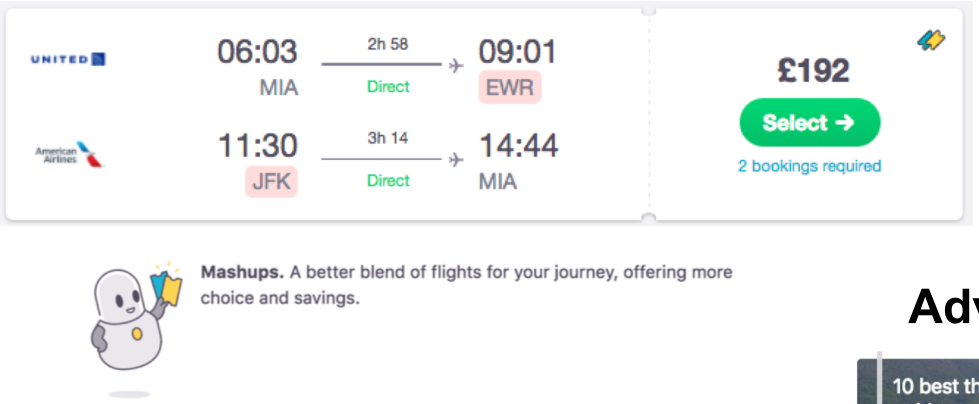
Machine Learning at Skyscanner

- Destination recommendations
- Itinerary mashups
- Advertising

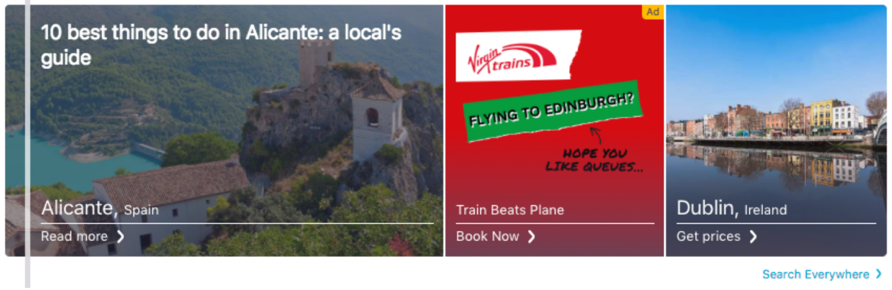
Recommendations



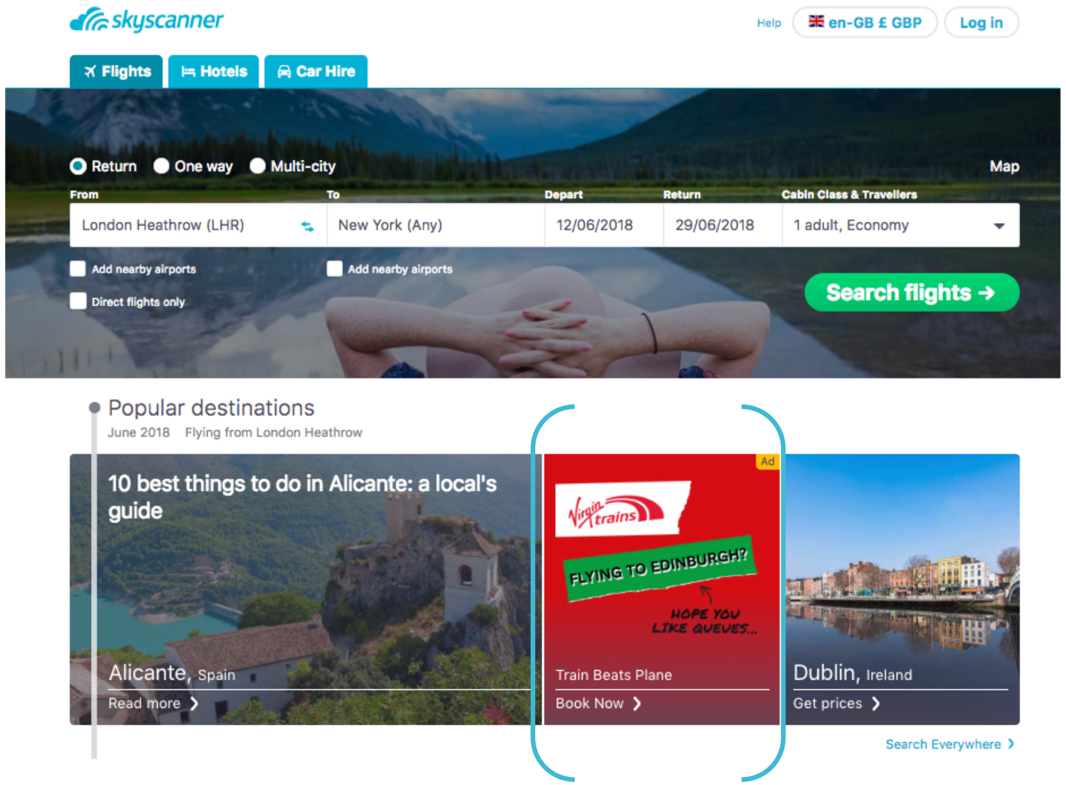
Mashups

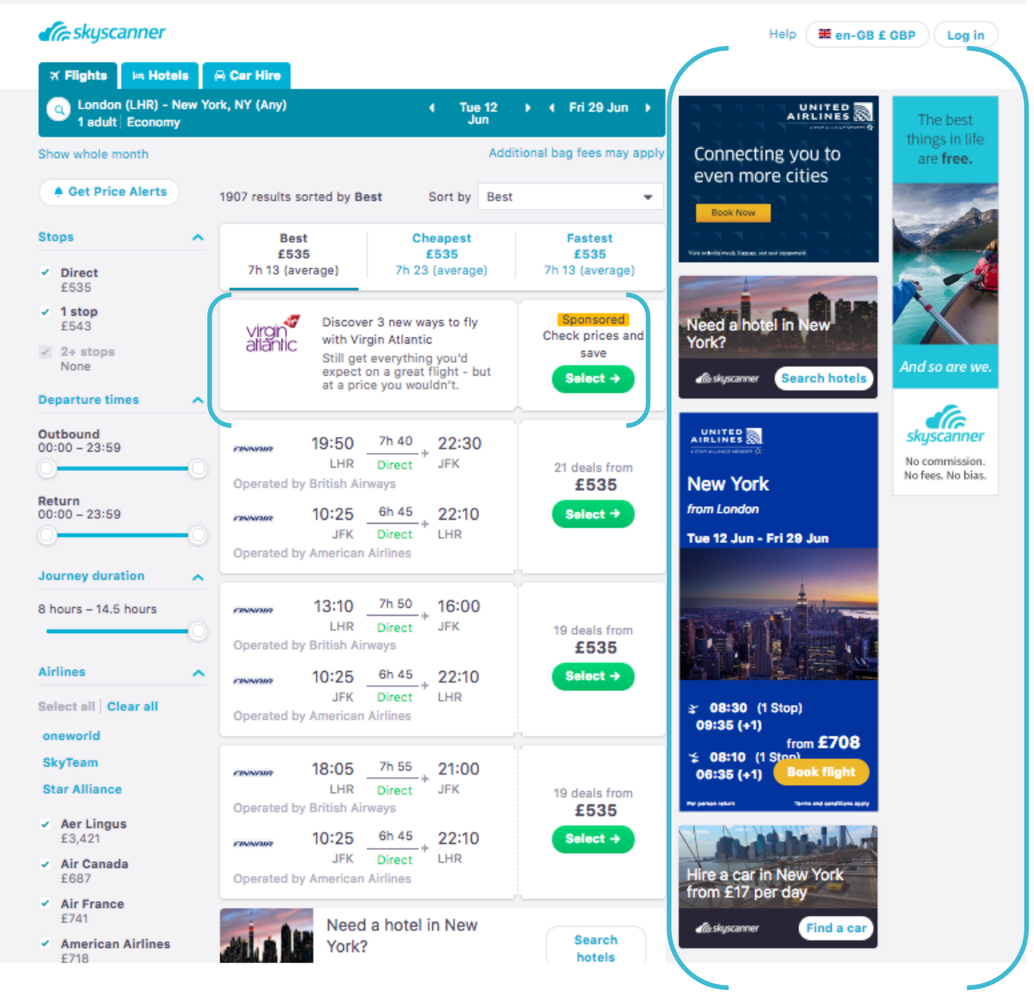


Advertising

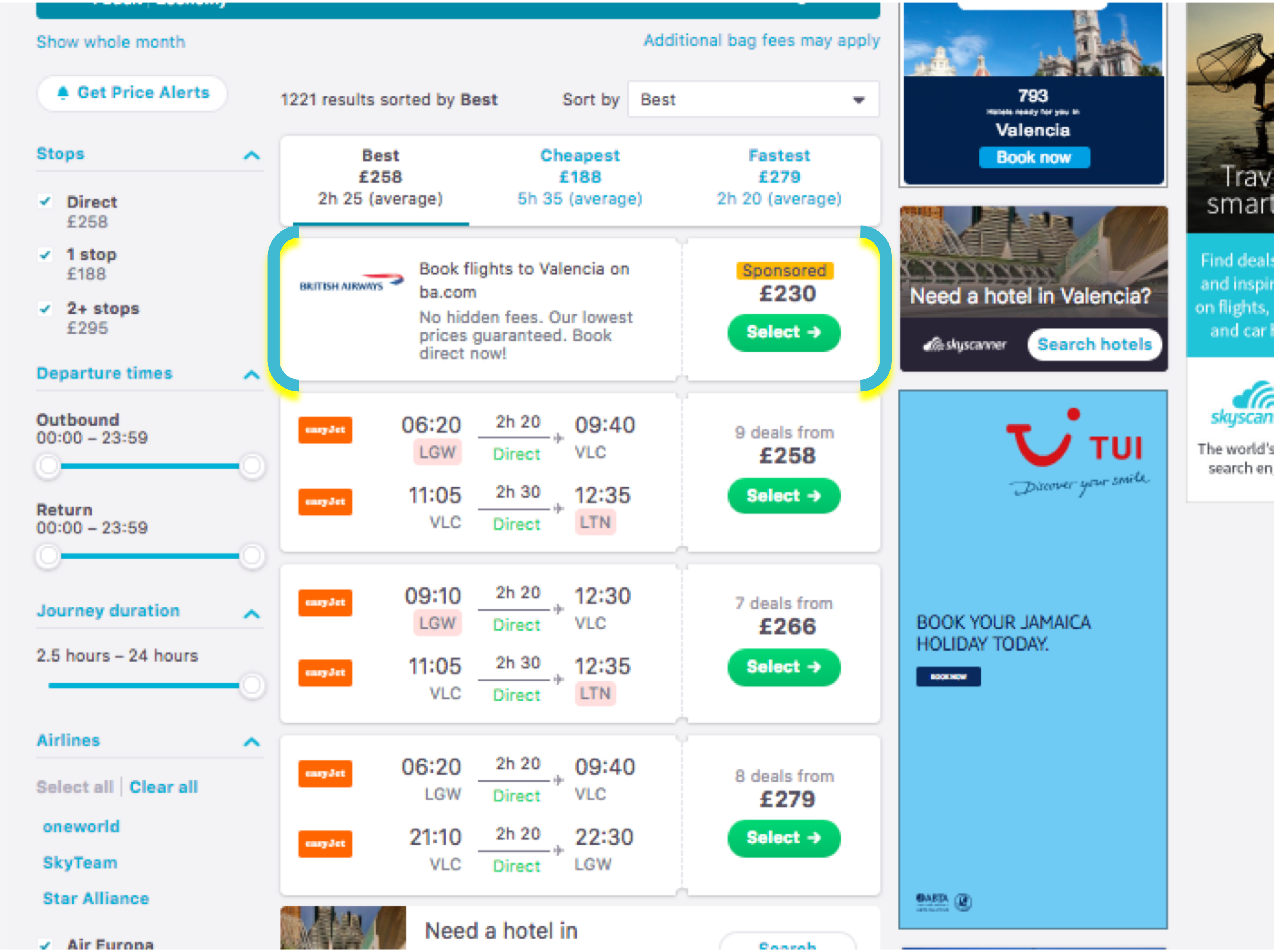


Ads at Skyscanner

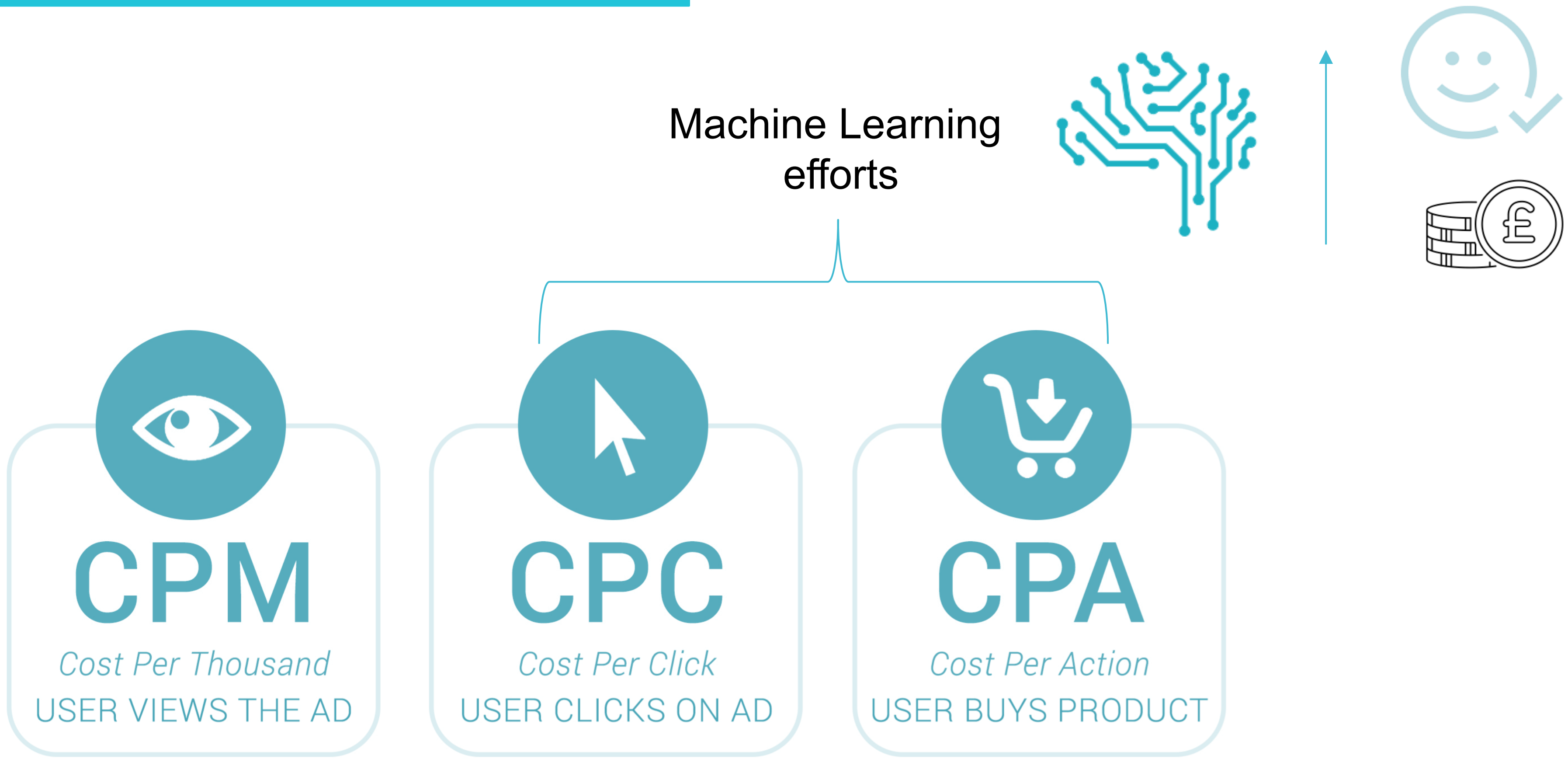




Not always annoying ads



Advertising schemes



Brand awareness

Traffic

Action

Advertising schemes

Brand awareness | Traffic | Action

Advertising Schemes

Machine Learning efforts

CPM
Cost Per Thousand
USER VIEWS THE AD

Brand awareness

CPC
Cost Per Click
USER CLICKS ON AD

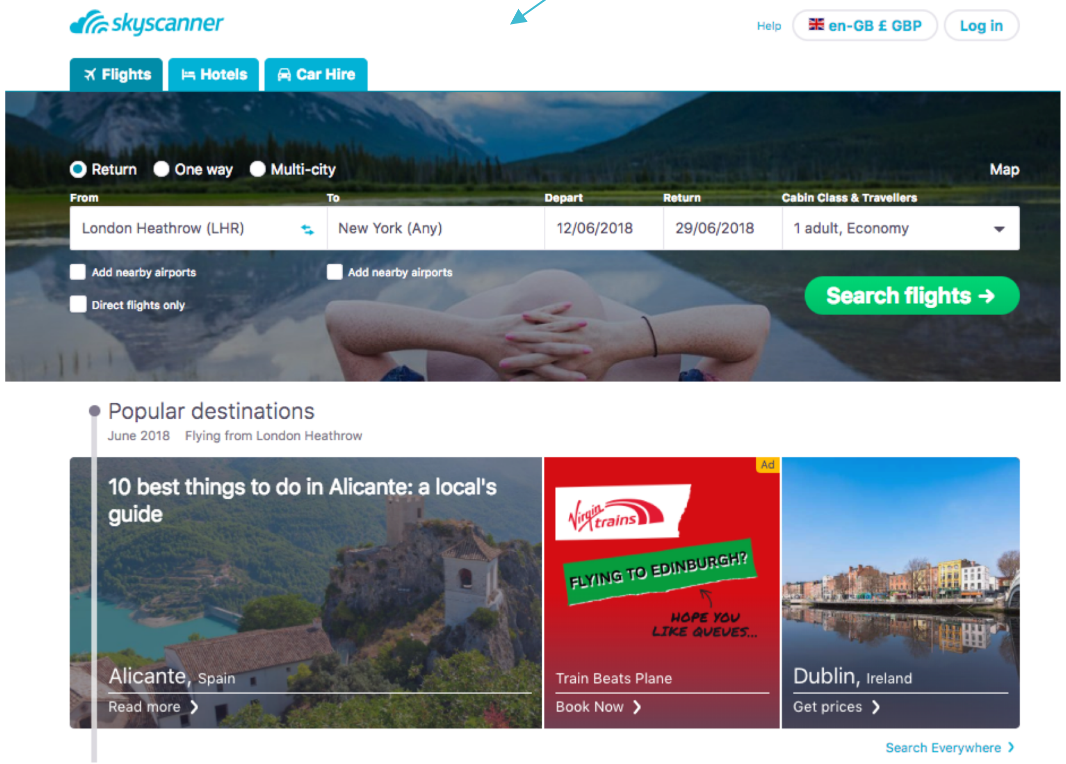
Traffic

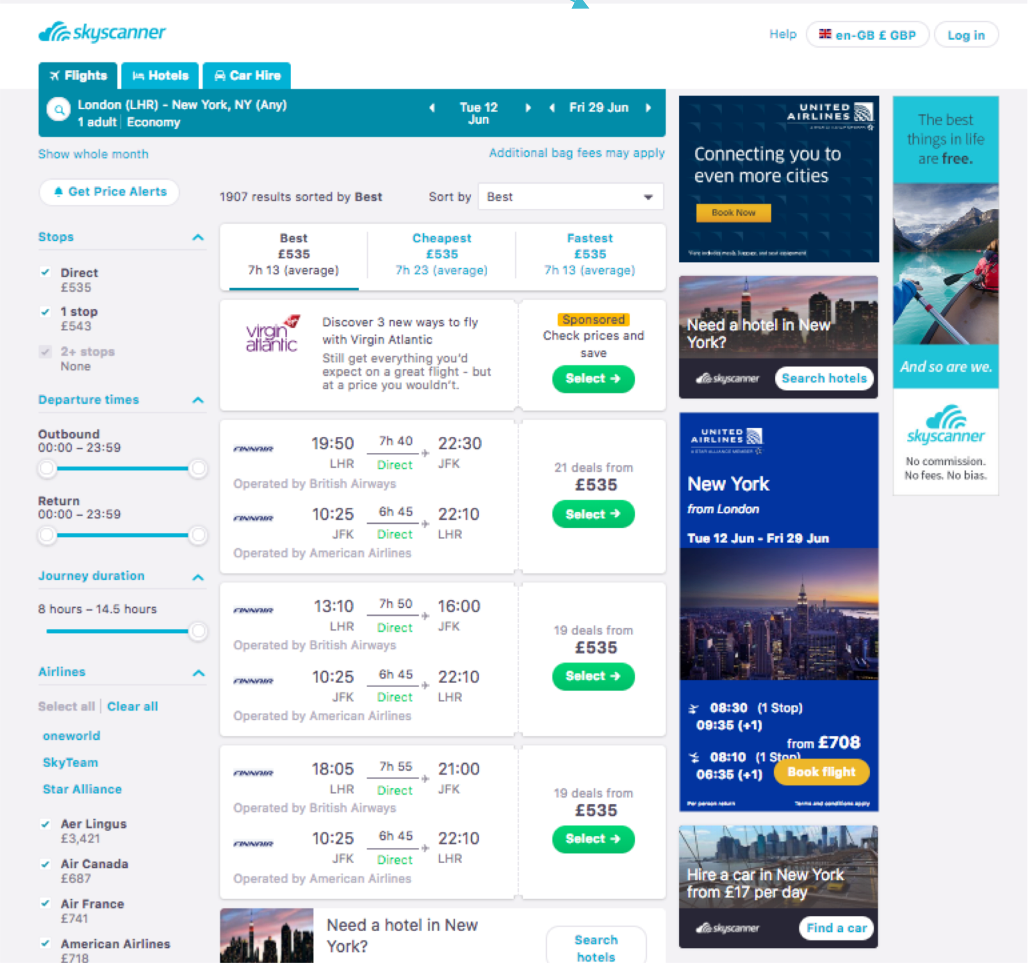
CPA
Cost Per Action
USER BUYS PRODUCT

Action

Ads at Skyscanner

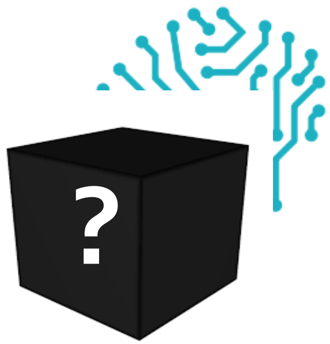
DoubleClick for Publishers





Ads at Skyscanner

DoubleClick
for Publishers



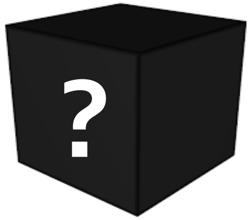
=

Make sure,
ads are delivered

No need to worry about how ads are delivered

Delivery of Ads

DoubleClick
for Publishers

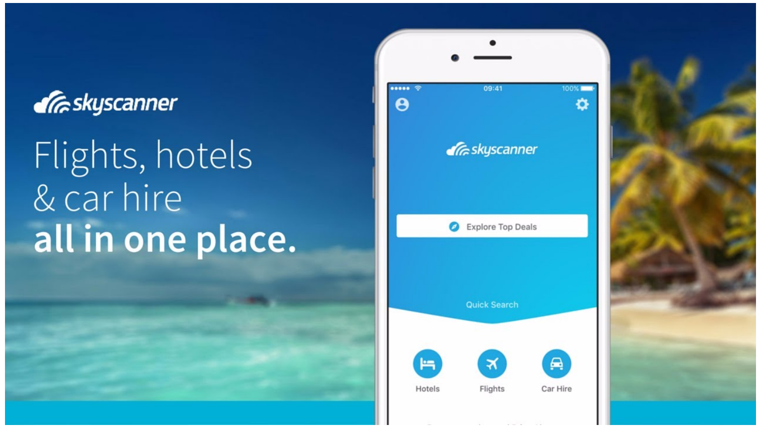


Black box algorithms

Technical difficulties

Data ownership

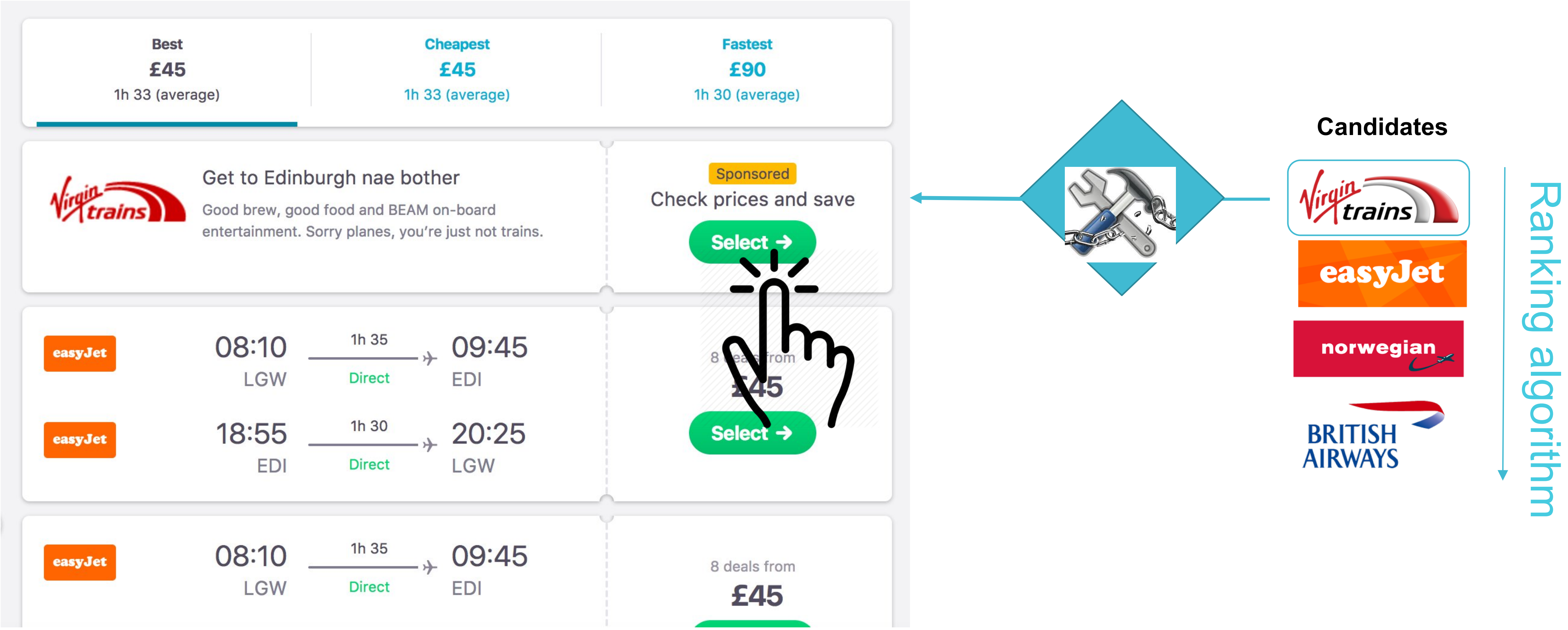
Solution



Skyscanner Ads Manager

Click prediction algorithm

Click prediction algorithm



Cloud services and tools at Skyscanner

Languages

python

node JS

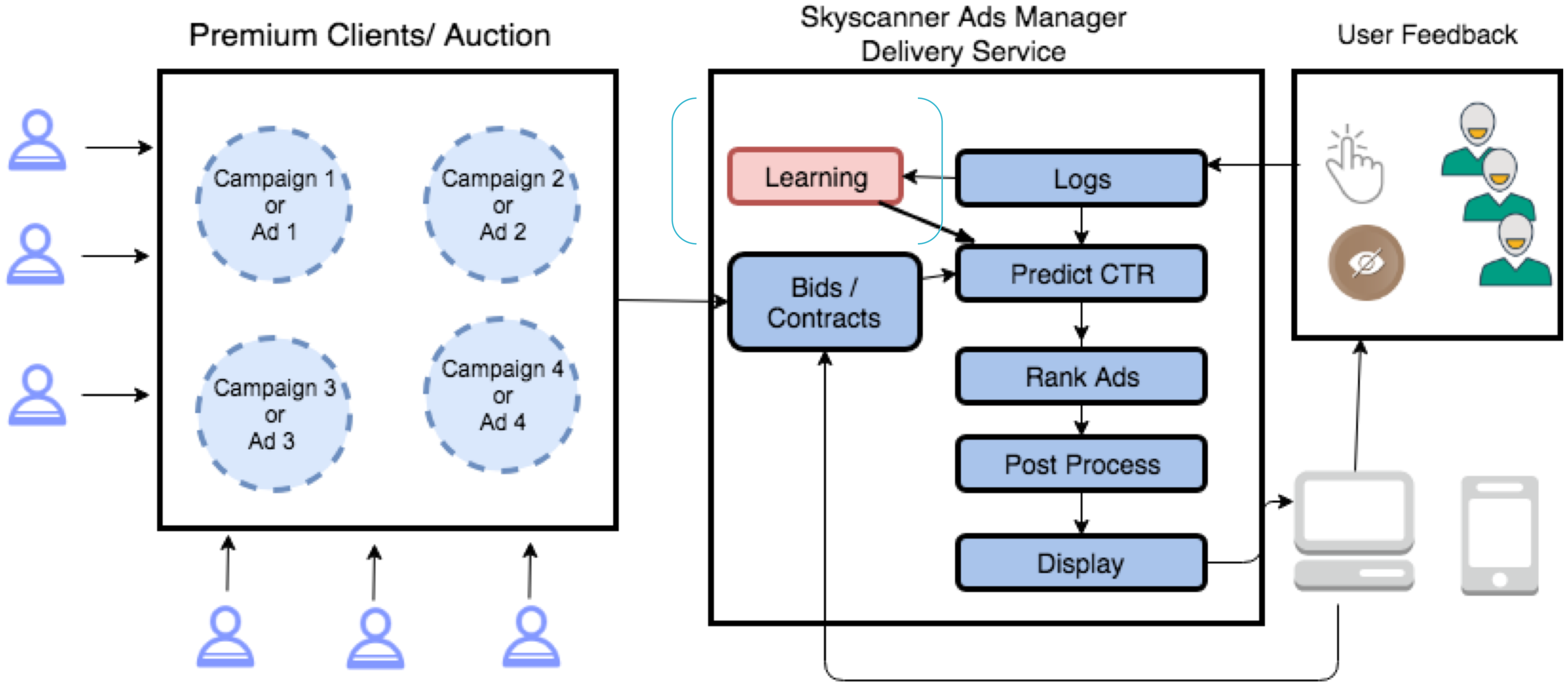
AWS services

S3

Amazon Athena

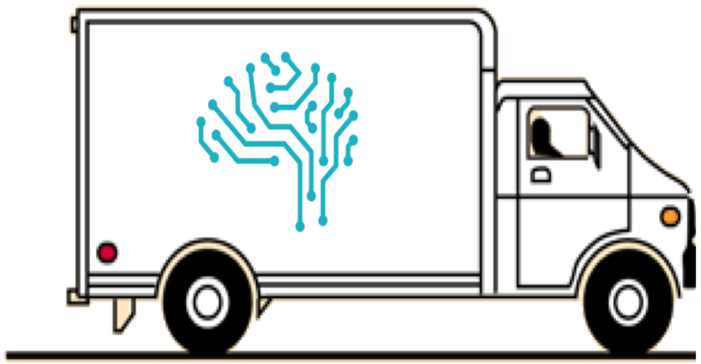
Batch

Overview of an Advertising System

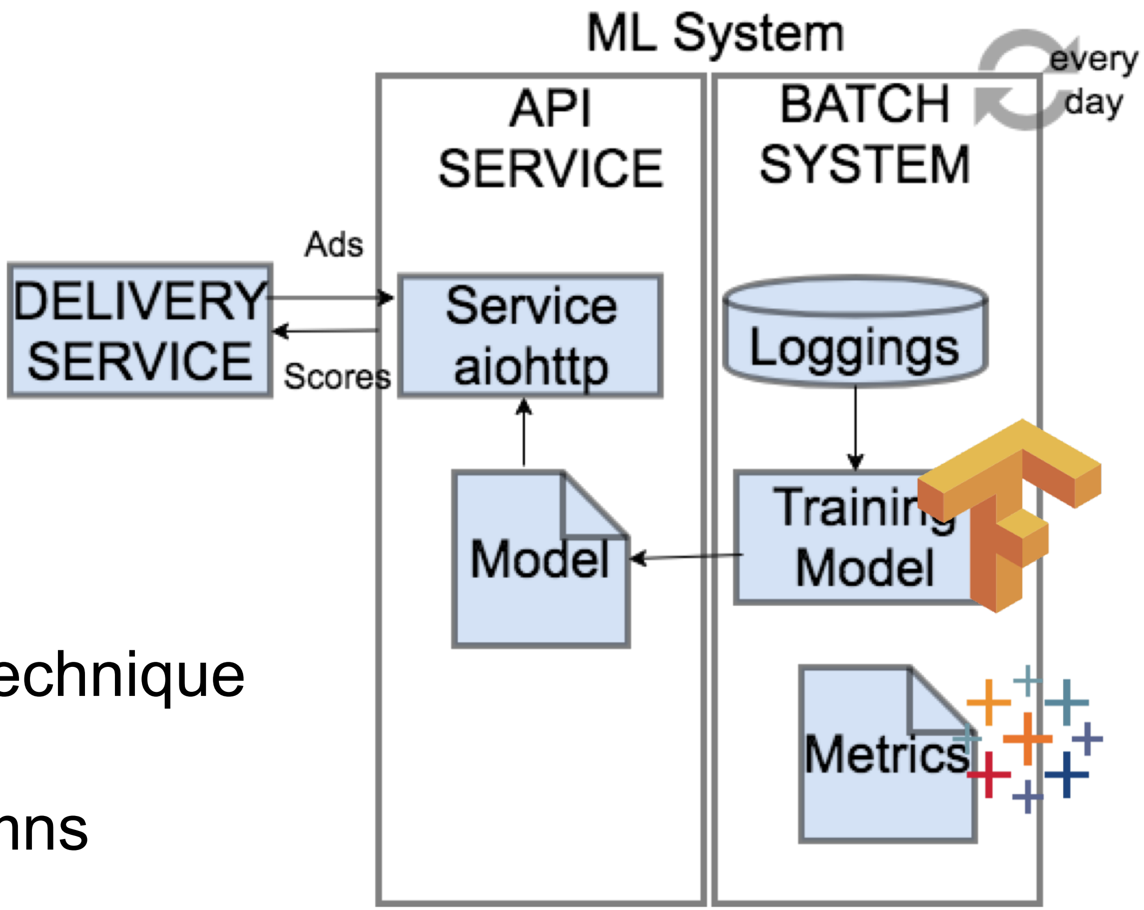


Expectation vs. reality

Expectation



Flexible but with the risk of not being fast enough.



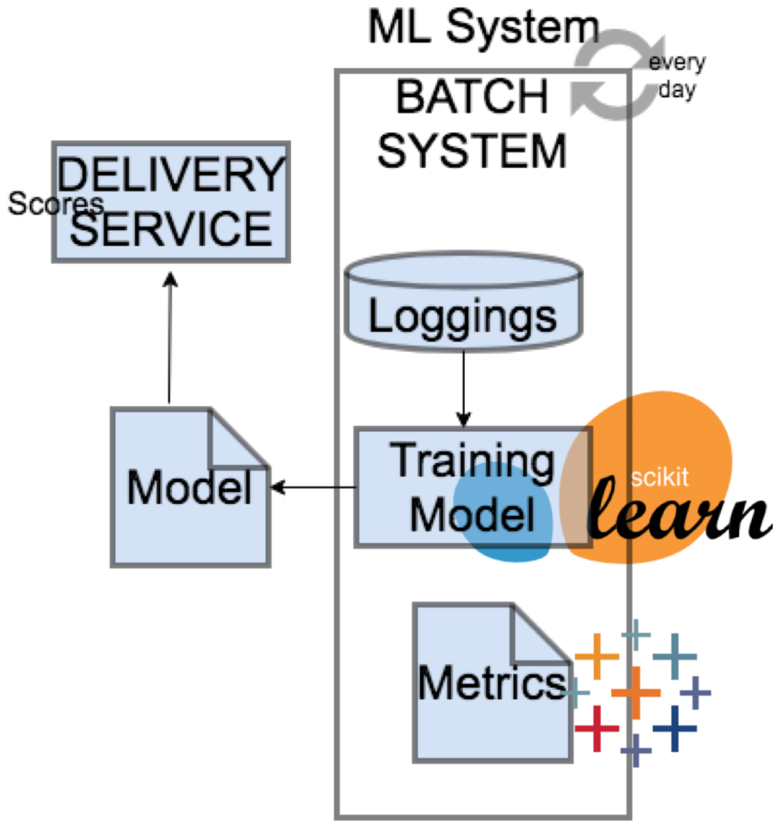
Features:
- User history,
- User features,
- Route features ,
- Ad features with, colors, text

Tensorflow
- Optimization technique
- Embeddings
- Crossed columns
- Hashing

Expectation vs. reality

Not flexible but fast and easier to implement.

Reality



Features:

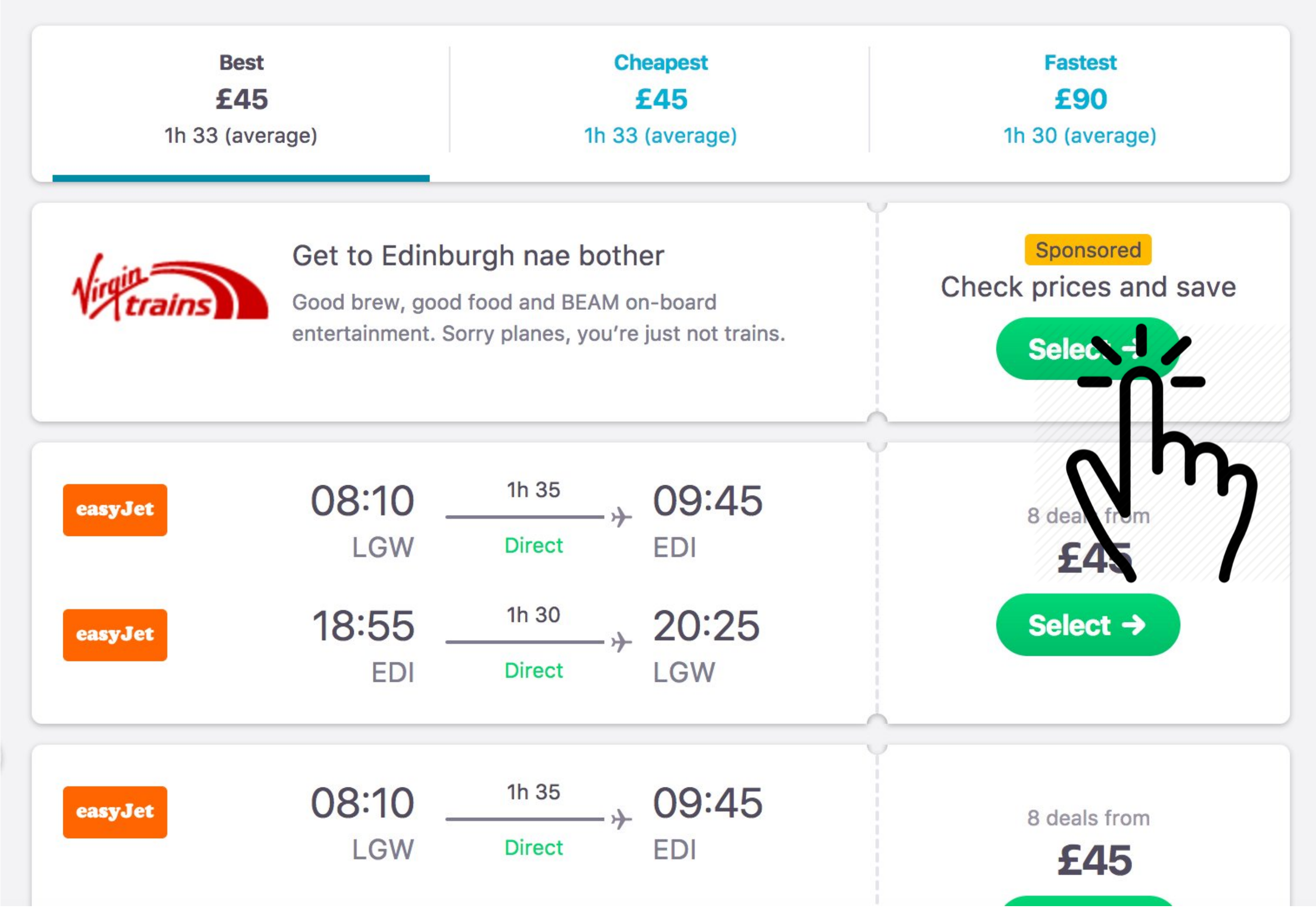
- ~~User history,~~
- User features,
- Route features ,
- Ad features with, ~~colors, text~~

~~Tensorflow~~

- ~~Optimization technique~~
- ~~Embeddings~~
- ~~Crossed columns~~
- ~~Hashing~~

Challenges in building the model

Goal: model, that estimates the likelihood of whether an impression will result in a click or not.



Challenges: Which model to use?

Model Possibilities (easy to read in node.js):

- **Logistic regression**
- Random Forest : gets lost
- Neural networks: too slow hard to put it in json

Gridsearch for hyperparameteres

Solvers:

- Logistic regression: Liblinear, sag

Train all data at once

- **SGDClassifier**

Train data in batches

Saves memory

Challenges: Categorical values

Categorical values:

- One hot encoding
- **One hot encoding grouping less frequent features**
- **Hashing trick**

- One hot encoding

Creatives
C1
C2
C3

C1	C2	C3
1	0	0
0	1	0
0	0	1

- One hot encoding grouping
- Less frequent features

Creatives
C1
C2
C3
C4

C1	C2	C3	C4
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Challenges: Categorical values

- **Hashing Trick: Same hashing function in training, testing and production**

id	features
123	creative1, advertiser2,mobile, etc.
321	creative2, advertiser4,mobile, etc.

id	Feat_1	Feat_2	Feat_3	….	Feat_k
123	1	0	1	….	0
321	1	0	0	….	1

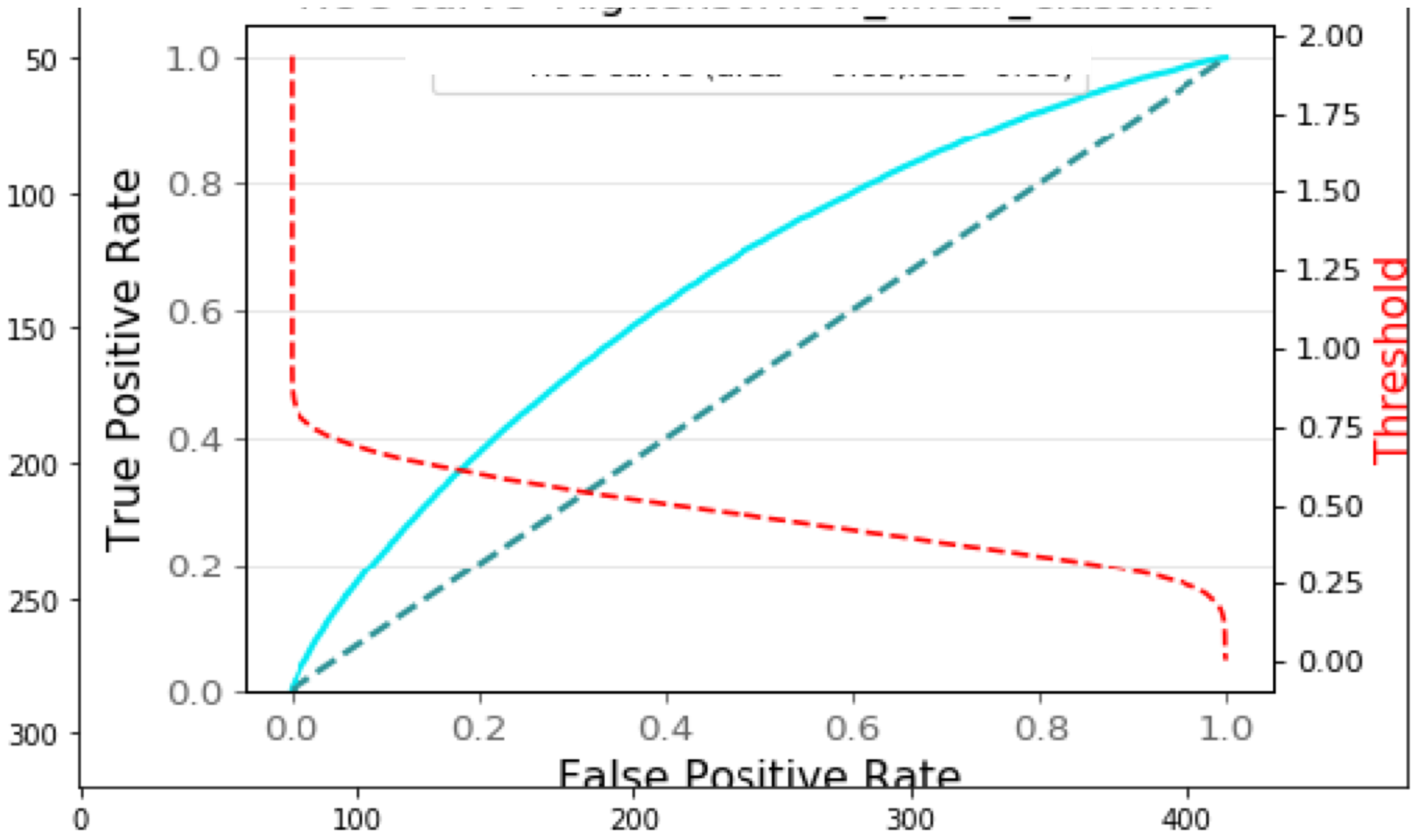
We gain

feature space reduction
(k is determined upfront)

new categories
n-grams are assigned to existing bins

collisions
usually don't bother us

Machine Learning Performance



AUC: if caring about ranking

Log-Loss: if caring about the value of CTR

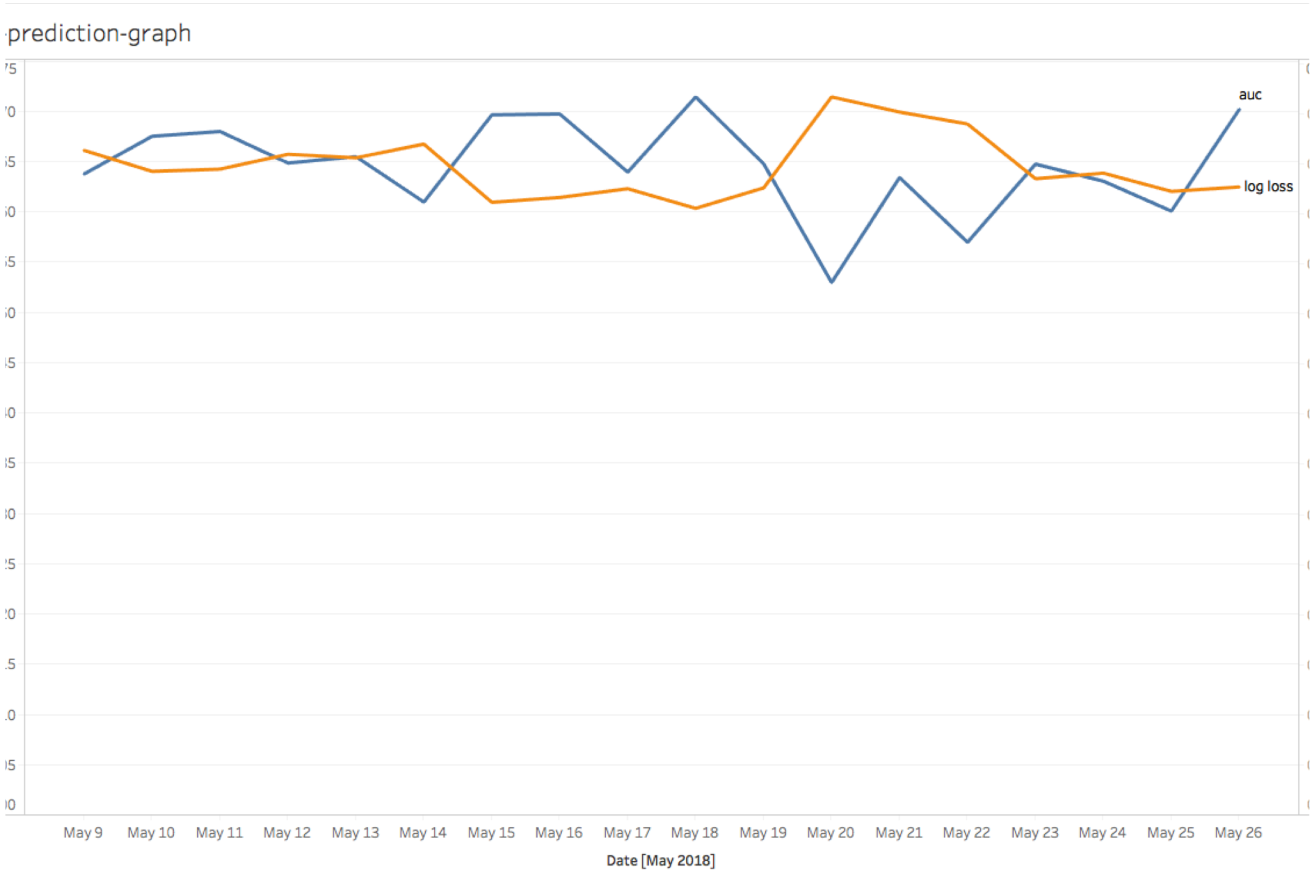
Other metrics:

Precision at 1: based on target groups.

Mean Reciprocal Rank

Challenges: How to monitor performance?

Updating model based on different Sampling methods.



click-prediction

to	algorithm	sampling	days
2018-05-10	logistic_regression_balanced	random	7
2018-05-11	logistic_regression	undersampling	6
2018-05-12	logistic_regression_balanced	not-sampled	5
2018-05-13	logistic_regression	undersampling	7
2018-05-14	logistic_regression_balanced	not-sampled	7
2018-05-15	logistic_regression	undersampling	3
2018-05-16	logistic_regression_balanced	not-sampled	6
2018-05-17	logistic_regression_balanced	not-sampled	7
2018-05-18	logistic_regression_balanced	not-sampled	7
2018-05-19	logistic_regression_balanced	not-sampled	7
2018-05-20	logistic_regression_balanced	not-sampled	7
2018-05-21	logistic_regression_balanced	undersampling	7
2018-05-22	logistic_regression_balanced	undersampling	5
2018-05-23	logistic_regression_balanced	not-sampled	3
2018-05-24	logistic_regression_balanced	not-sampled	7
2018-05-25	logistic_regression	random	7
2018-05-26	logistic_regression	random	7
2018-05-27	logistic_regression_balanced	random	5

The road ahead: Balancing exploitation and exploration

EXPLOITATION

Playing the machine that (currently) pays out the most.

Choose ad based on ONLY CTR

EXPLORATION

Playing the other machines to see if any pay out more.

Choose ad based on OTHER criteria

The road even further ahead: quality

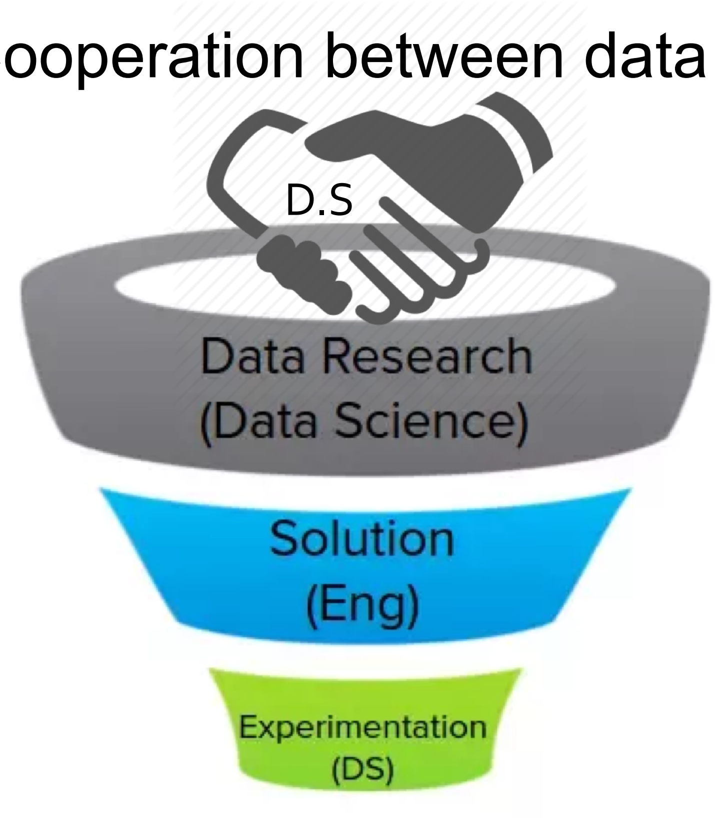
Post-click, colors, images, text, landing page

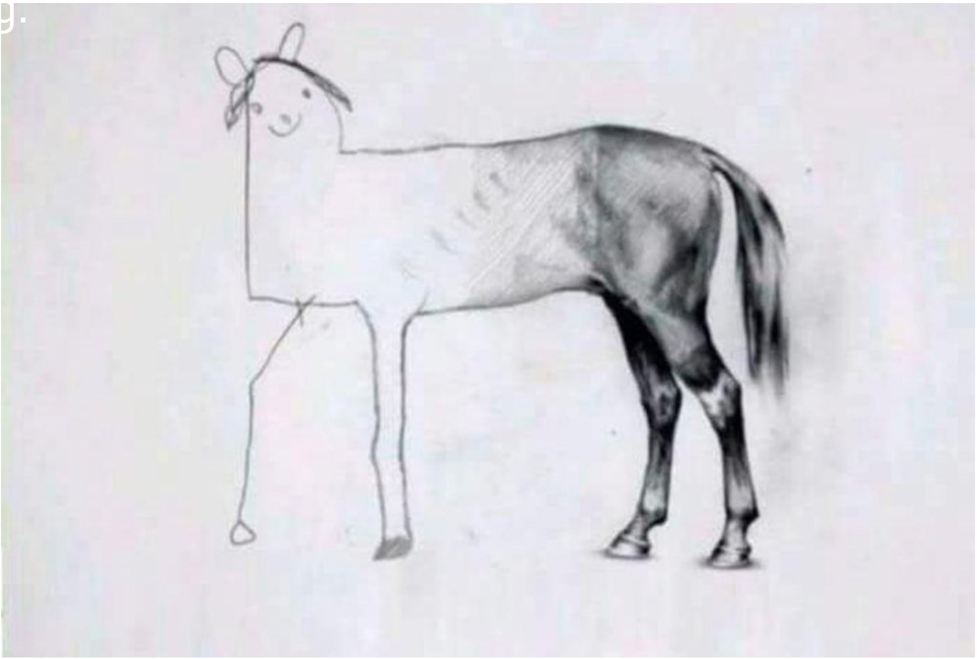
Conclusions

1. Machine Learning can bring many benefits to the product the challenge is to prove it

2. Bringing ML into production could hard at the beginning

3. Cooperation between data scientists and engineers is crucial





Thank you

@ruthygarcia

Questions?