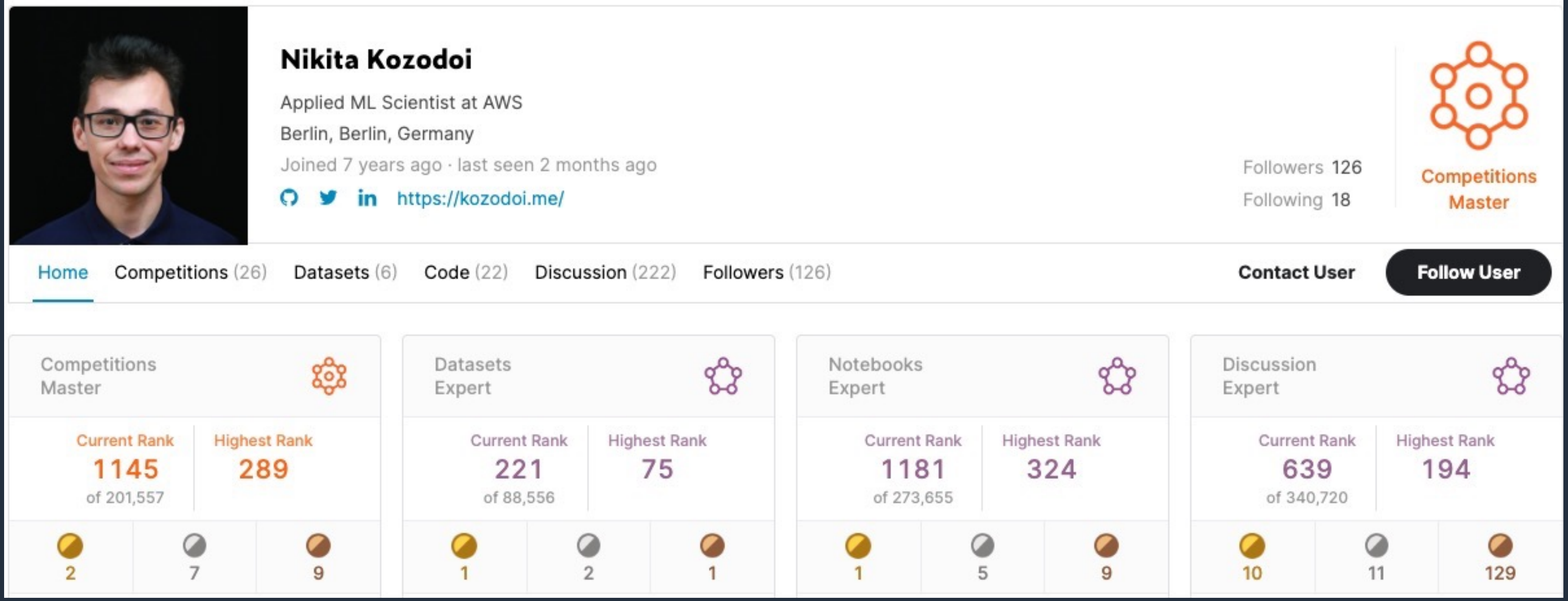
Kaggle Lessons that Work in Industry

Nikita Kozodoi, PhD
Applied Scientist at AWS

30.05.2023

About Me

- Applied Scientist at AWS
- Earned 18 Kaggle competition medals



https://www.kaggle.com/kozodoi

About My Team

Machine Learning Solutions Lab

- Identify, scope and implement highest-value ML use cases to accelerate adoption
- Global team of Data /Applied Scientists, ML Engineers and ML Strategists
- Extensive expertise across a variety of verticals and solution areas

Agenda

- Motivation
- Lesson #1: Compressing your Data
- Lesson #2: Designing Good Validation Strategy
- Lesson #3: Selecting the Right Metric
- Summary

Motivation

Background

- Kaggle is one of the largest online ML communities
  - Offers courses, datasets, and more
  - Mostly known for ML competitions
  - Over 10 million users as of 2022

https://www.kaggle.com

Motivation

- Kaggle competitors fight for every digit in model KPIs

#	Team	Members	Score
1	Kraków, Lublin i Zhabinka		0.81724
2	ikiri_DS	+8	0.81241
3	circlecircle		0.81124
4	alijs & Evgeny		0.81086
5	Large Space Hypothesis		0.81041
6	七上八下	+3	0.81039
7	TenDots	+6	0.81007
8	楼上神仙打架 ¯_(ツ)_/¯	+2	0.80993
9	Vegetable chicken	+7	0.80972
10	Quad Machine		0.80941

Average score difference is less than 0.001

Motivation

- Kaggle competitors fight for every digit in model KPIs
- Over time, people learn best practices for different tasks
  - Squeezing the last drop of model performance
  - Improving the training or processing speed
- Some of these lessons can be leveraged in industry

Lesson #1

Compressing your Data

#1. Compressing your Data

https://mlcontests.com/state-of-competitive-machine-learning-2022/

#1. Compressing your Data

Example:

- Tabular dataset does not fit into the RAM
- Scientist has to use a larger instance that costs more

ID	Product type	…	Sales volume
1	"Book"	…	100.00
2	"Game"	…	50.00
3	"Book"	…	80.00

~ 2 Gb

#1. Compressing your Data

Problem:

working with large datasets is slow and requires a lot of compute

Idea:

reduce the data size using lossless compression

Goal:

decrease compute costs and enable faster experiments

#1. Compressing your Data

How:

- **int/float:** choose format depending on the range
- **binary:** convert to bool
- **string:** convert to category

	Memory used	Data range
int8	1 byte	[-128, 127]
int16	2 bytes	[-32768, 32767]
…	…	…
float32	4 bytes	[$-3.4*10^{38}$, $3.4*10^{38}$]
float64	8 bytes	[$-1.7*10^{308}$, $1.7*10^{308}$]

https://towardsdatascience.com/reducing-memory-usage-in-pandas-with-smaller-datatypes-b527635830af

#1. Compressing your Data

ID [int64]	Product type [str]	…	Sales volume [float32]
1	"Book"	…	100.00
2	"Game"	…	50.00
3	"Book"	…	80.00

Lossless data compression

ID [int16]	Product type [category]	…	Sales volume [int32]
1	Book	…	100
2	Game	…	50
3	Book	…	80

```
df = reduce_memory_usage(df)

100%|██████████| 71/71 [01:12<00:00,  1.02s/it]
Memory usage decreased from 1573 Mb to 233 Mb (85.21% reduction)
```

Lesson #2

Designing Good Validation Strategy

#2. Designing Good Validation

Example:

- Company sells products in market A
- Company enters market B with different properties
- Model performs well on historical data from A, but fails on data from B

85% → 69%

Accuracy on validation set — Accuracy in production

#2. Designing Good Validation

Problem:
offline performance often doesn't match performance in production

Idea:
set up validation sample that mimics production as close as possible

Goal:
avoid overfitting to a non-representative validation set

#2. Designing Good Validation

How:

- Use stratified splits for both classification and regression
  - Match expected distributions of multiple features
  - Bin continuous features and stratify based on bin ratios
- Regularly update partitioning to reflect data shifts
- Perform adversarial validation to check split quality

#2. Designing Good Validation

How:

- Use stratified splits for both classification and regression
- Regularly update partitioning to reflect data shifts
    - Consider updating the data split with certain frequency
    - Helps to address data distribution shifts
- Perform adversarial validation to check split quality

#2. Designing Good Validation

How:

- Use stratified splits for both classification and regression
- Regularly update partitioning to reflect data shifts
- Perform adversarial validation to check split quality
    - Combine validation set and new production data into one dataset
    - Train a classifier to distinguish between the data sources

Lesson #3

Optimizing the Right Metric

#3. Optimizing the Right Metric

Example:

- In demand forecasting, over- and underprediction has different costs
  - Overprediction: costs of storing extra items at a warehouse
  - Underprediction: costs of unrealized sales opportunity

#3. Optimizing the Right Metric

Problem:

ML metric optimized by the model doesn’t reflect the business KPI

Idea:

aim at optimizing the KPI on which the solution is evaluated

Goal:

maximize relevant metric on every step of the modeling pipeline

#3. Optimizing the Right Metric

How:

- Modify the ML model to use custom business-inspired loss
  - Deep Learning: create a custom loss class with relevant calculations
  - Tree Models: provide a custom differentiable loss function

$$\boxed{\text{loss} = \begin{cases} \alpha\ |\text{error}| & \text{if} \quad \text{actual} > \text{prediction} \\ \beta\ |\text{error}| & \text{if} \quad \text{actual} \leq \text{prediction} \end{cases}}$$

#3. Optimizing the Right Metric

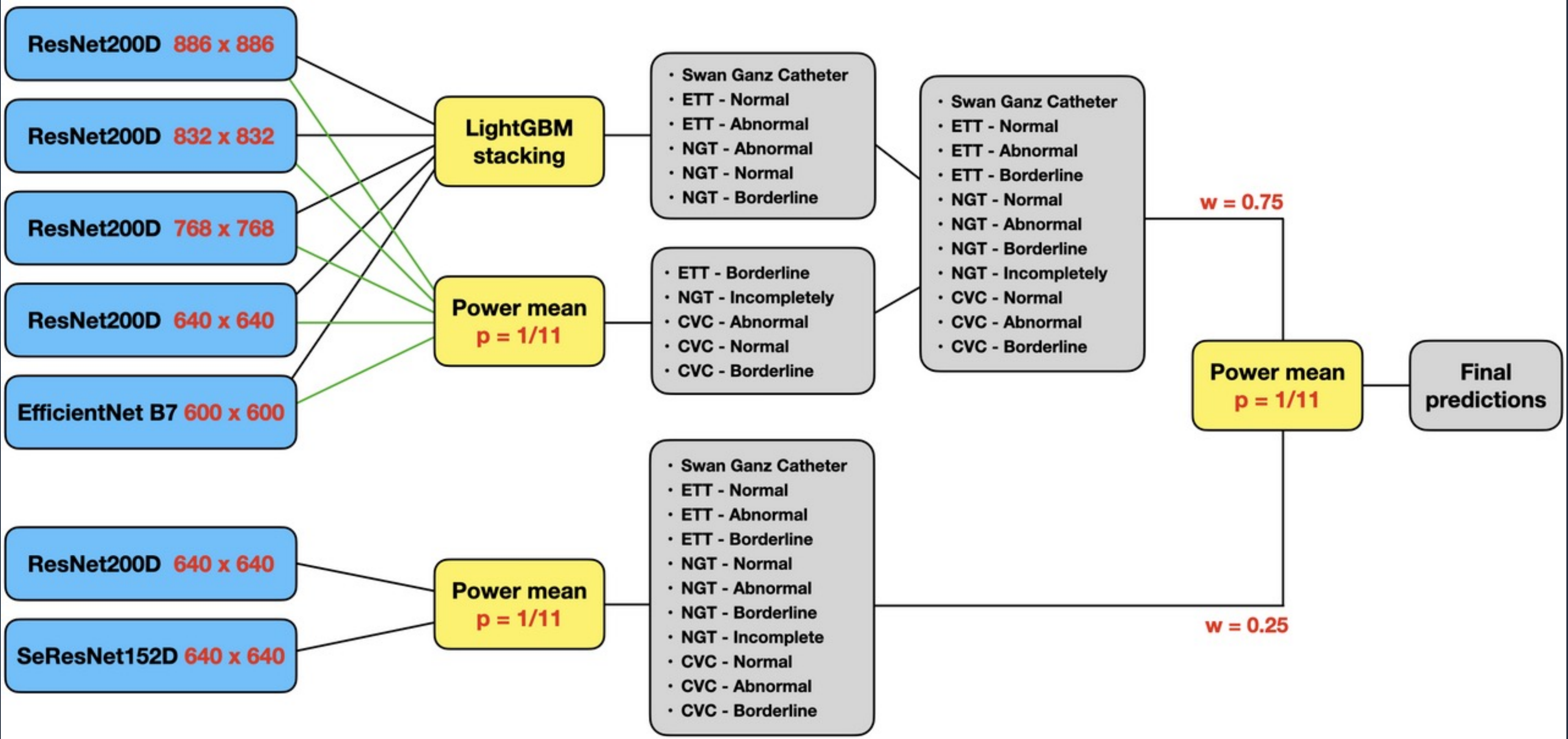
How:

- Modify the ML model to use custom business-inspired loss
- Use business-inspired KPI for tuning and model selection
- Post-process predictions to account for business logic
  - Set negative predictions to zero if relevant
  - Use calibration to get probabilistic predictions
  - Optimize thresholds in classification tasks

Bonus Lesson

Lesson That Should Not be Learned

Avoid Heavy Ensembling



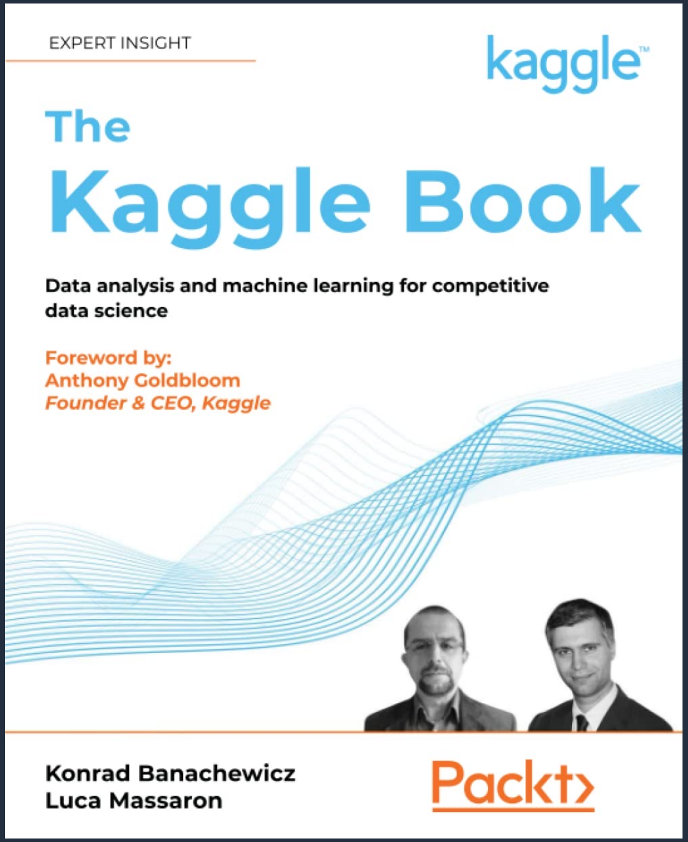
https://www.kaggle.com/c/ranzcr-clip-catheter-line-classification/discussion/226664

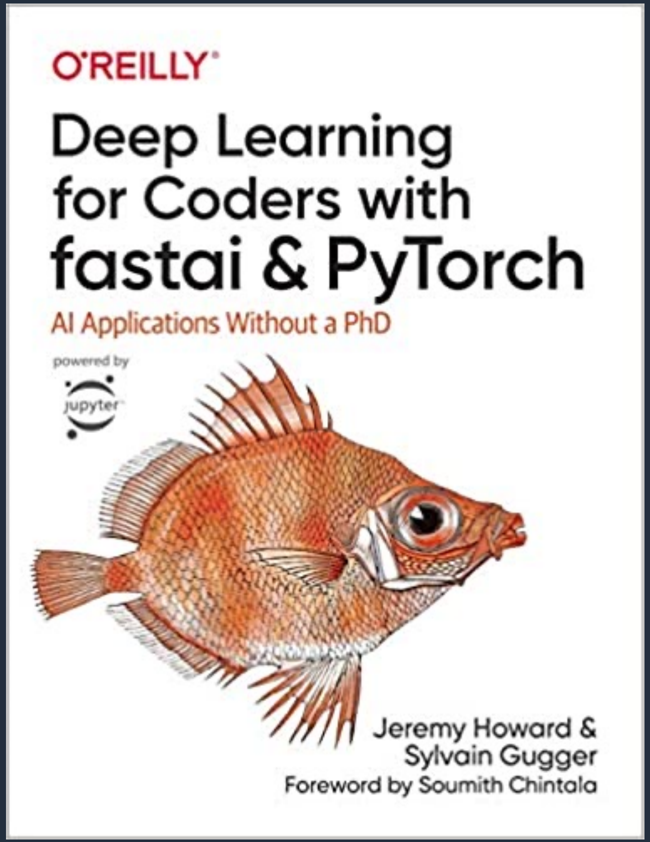
Take Aways

- Some lessons from competitive ML are useful in industry
  - Compressing your data
  - Designing good validation
  - Optimizing the right metric

- When starting ML projects, check recent Kaggle competitions
  - Look through discussions & notebooks for similar problems
  - Winners usually publish their code, including different tricks

Further References





Thank you!

Nikita Kozodoi, PhD
Applied Scientist at AWS