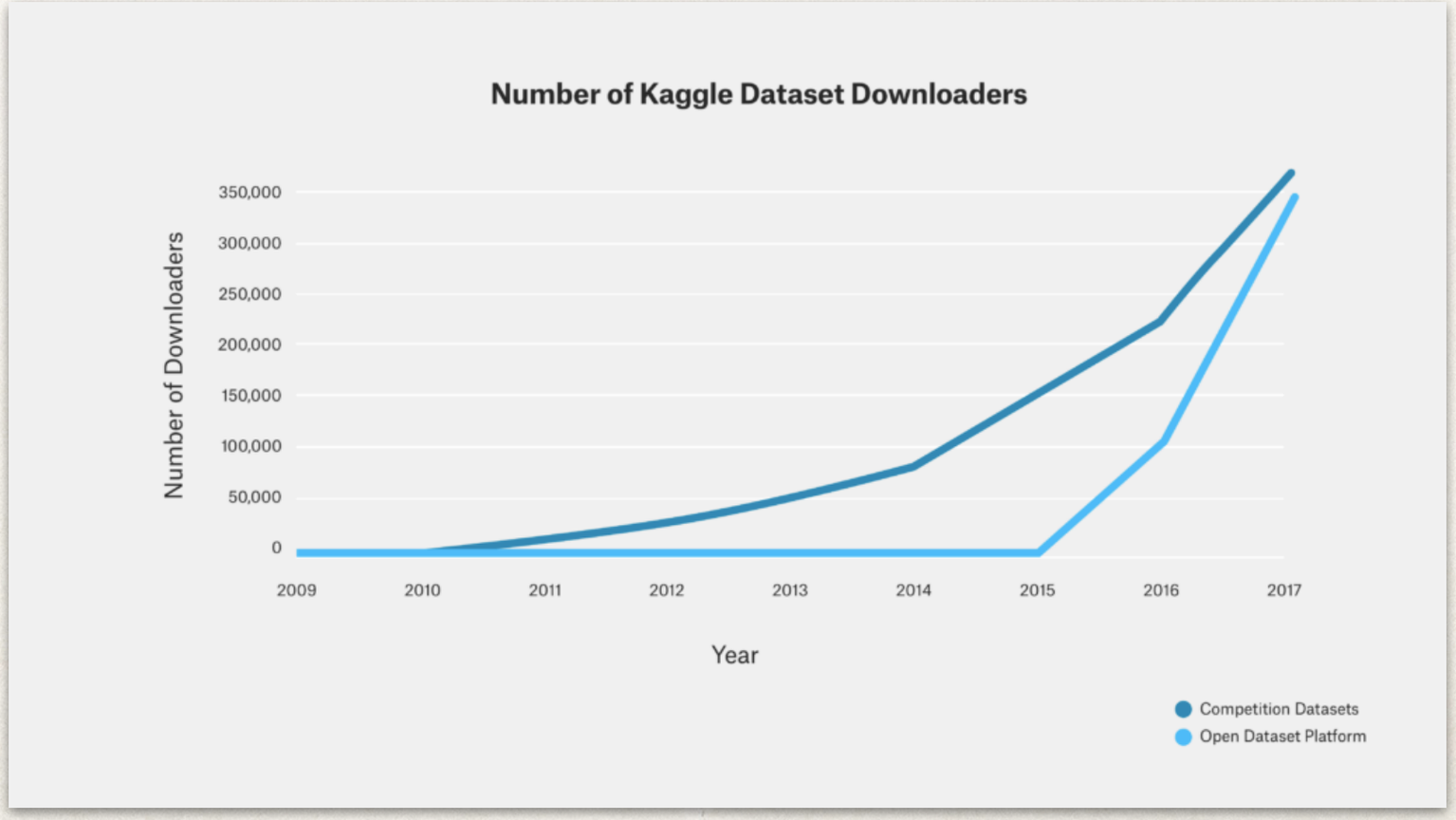
How to compete on Kaggle

Valentin Kuznetsov, Cornell University

SOSC 2018

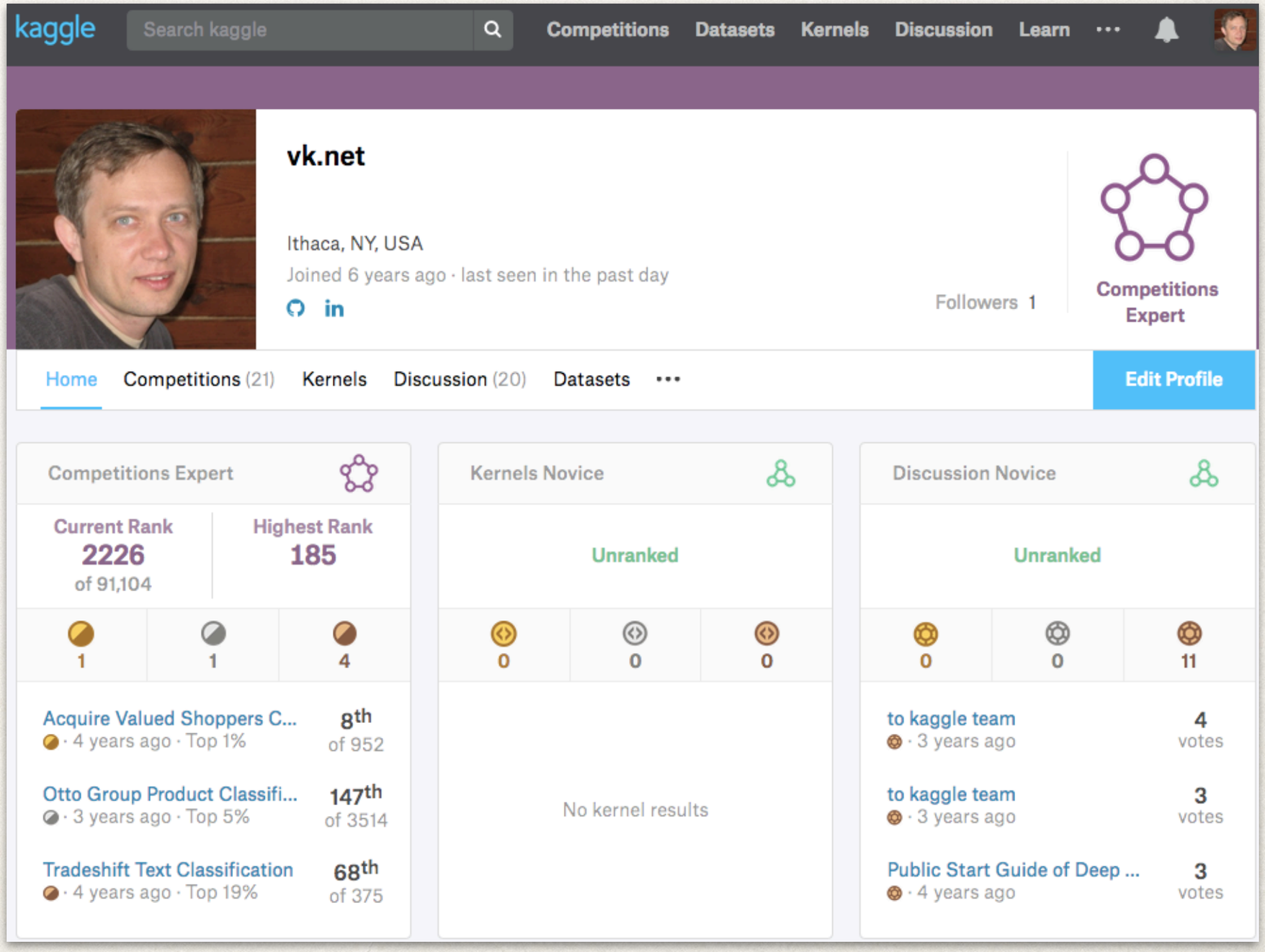
kaggle.com

- It is open platform for Data Scientist to compete over published datasets
- In 2017: 120K DataScientists compete in 44 competitions, a total prize sum was $4.75M+, 600K new users joined, 1.3M total users



My kaggle profile

- My goal is to learn ML/DL/AI and not prizes
  - my main source of DataScience
- I competed alone in my free time apart from regular job, teachings, student projects, family, etc.
  - turns out it is much tougher to compete alone since amount of info, data, training, ideas significantly increases
  - I was mostly active around 2015



Kaggle competitions

Acquire Valued Shoppers Challenge
Predict which shoppers will become repeat buyers
Featured · 4 years ago
8/952
Top 1%

Otto Group Product Classification Challenge
Classify products into the correct category
Featured · 3 years ago · internet, tabular data
147/3514
Top 5%

Tradeshift Text Classification
Classify text blocks in documents
Featured · 4 years ago
68/375
Top 19%

National Data Science Bowl
Predict ocean health, one plankton at a time
Featured · 3 years ago · oceanography, image data, multiclass classification
90/1049
Top 9%

Homesite Quote Conversion
Which customers will purchase a quoted insurance plan?
Featured · 2 years ago · tabular data, binary classification
102/1764
Top 6%

- Homesite competition

1,764	1,939	36,387
Teams	Competitors	Entries

Homesite dataset

- Using an anonymized database of information on customer and sales activity, including property and coverage information, Homesite is challenging you to predict which customers will purchase a given quote. Accurately predicting conversion would help Homesite better understand the impact of proposed pricing changes and maintain an ideal portfolio of customer segments.
- This dataset represents the activity of a large number of customers who are interested in buying policies from Homesite. Each QuoteNumber corresponds to a potential customer and the QuoteConversion_Flag indicates whether the customer purchased a policy.
- The provided features are anonymized and provide a rich representation of the prospective customer and policy. They include specific coverage information, sales information, personal information, property information, and geographic information. Your task is to predict QuoteConversion_Flag for each QuoteNumber in the test set.
- Train sample: 299 columns (28 categorical variables), 260K rows (200MB); test sample 174K rows (131MB)

Homesite leaderboard

1	—	KazAnova | Faron | clobber	0.97024
2	—	Frenchies	0.97018
3	▲1	New Model Army | CAD & QuY	0.97001
4	▼1	Gilberto | Leustagos | Stanislav	0.96988
5	—	The Northern Hemisphere	0.96983
6	▲1	victor, clustifier & adam	0.96968
7	▼1	monkeys rising	0.96961
8	—	A Few with NO Clue	0.96960
9	▲2	Daniel FG	0.96959
10	▼1	VinaKago	0.96956
100	▼11	BMX	0.96793
101	▲23	Overfitters	0.96792
102	▲101	vk.net	0.96792
		All Zeros Benchmark	0.50000

1st place

-0.49%

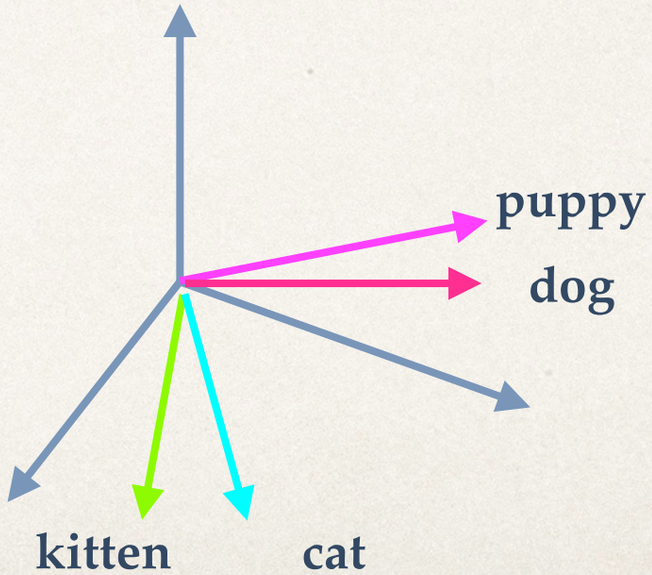
Recipe

- Node/environment setup
  - introduction to Anaconda
- Data exploration
  - introduction to R
- System limitations
  - issues with python, R and others
- Data preprocessing
  - intro to Python tools, common format, data scaling, normalization, working with NAs, etc.
- Training and modeling
- Reaching the limit
  - embeddings, stacking, etc.

Word embedding

- A way to capture multi-dimensional relationships between categories
  - e.g. Sun and Sat may have similar effect while other days may be treated independently
  - you define a dimension of word vector up-front
  - it projects categorical variables into another phase space, e.g. days may be sunny or rainy, season or off season; all of these features are hidden from original data representation
- Use NN or other ML algorithms to train the model to find best representation of embedded variables

puppy	[0.9, 1.0, 0.0]
dog	[1.0, 0.2, 0.0]
kitten	[0.0, 1.0, 0.9]
cat	[0.0, 0.2, 1.0]



Embeddings recipe

- Identify categorical variables and order them
- Define embedded matrix and cardinality of categorical variable
- Perform one-hot-encoding
- Train Neural Network model
- Extract NN weights (embeddings matrix)
- Plug embeddings matrix into regular ML model instead of categorical variable
- Train ML model with embeddings matrices