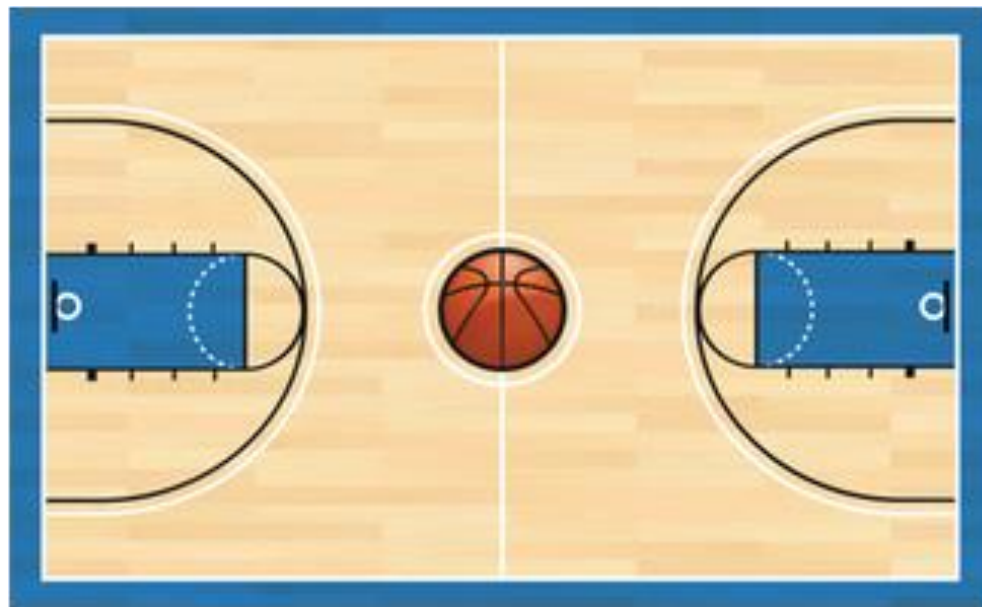


# March Machine Learning Mania 2016



[Ссылка на Kaggle](#)

[Смердов Антон](#)  
Апрель 2016

Completed • \$25,000 • 598 teams

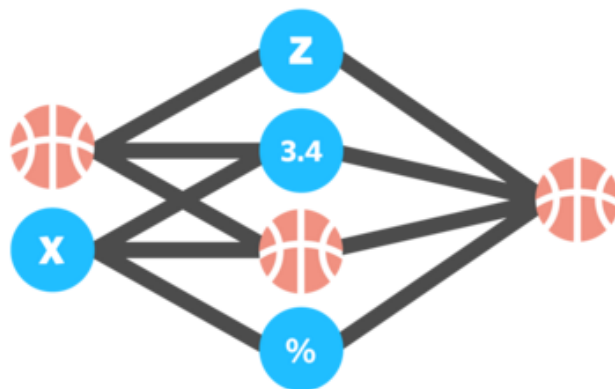
## March Machine Learning Mania 2016

Thu 11 Feb 2016 – Tue 5 Apr 2016 (10 days ago)

### Predict the 2016 NCAA Basketball Tournament

**Update:** although the tournament is over, we're continuing our analysis under the [predictions dataset page](#).

Back for its third year, March Machine Learning Mania challenges data scientists to predict winners and losers of the men's 2016 NCAA basketball tournament. You're provided data covering three decades of historical NCAA games and freely encouraged to use other sources of data to gain a winning edge.



In stage one of this two-stage competition, participants will build and test their models against the previous four tournaments. In the second stage, participants will predict the outcome of the 2016 tournament. You don't need to participate in the first stage to enter the second. The first stage exists to incentivize model building and provide a means to score predictions. The real competition is forecasting the 2016 results.

# Timeline

## Stage 1 - Model Building

- **Mar 12, 2016** - prior to this deadline competitors build and test models on historical data. The leaderboard shows the model performance on historical tournament outcomes.

## Stage 2 - 2016 Championship

- **Sunday, Mar 13** - Selection Sunday (68 teams announced)
- **Monday, Mar 14** - Kaggle begins to accept 2016 predictions. Release of up-to-date 2015-2016 season data.
- **Wednesday, Mar 16** - Final deadline to submit 2016 predictions (11:59PM UTC).
- **Mar 17 - Apr 4** - Watch your predictions come true!

All deadlines are at 11:59 PM UTC on the corresponding day unless otherwise noted. The organizers reserve the right to update the contest timeline if they deem it necessary.

# Evaluation

Submissions are scored on the log loss, also called the predictive binomial deviance:

$$\text{LogLoss} = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)],$$

where

- $n$  is the number of games played
- $\hat{y}_i$  is the predicted probability of team 1 beating team 2
- $y_i$  is 1 if team 1 wins, 0 if team 2 wins
- $\log()$  is the natural (base e) logarithm

A smaller log loss is better. Games which are not played are ignored in the scoring. Play-in games are also ignored (only the games among the final 64 teams are scored). The use of the logarithm provides extreme punishments for being both confident and wrong. In the worst possible case, a prediction that something is true when it is actually false will add infinite to your error score. In order to prevent this, predictions are bounded away from the extremes by a small value.

# Специфика

- Предсказываем будущее
- Соревнование проходит уже три года подряд
- Значительное влияние случайности
- Можно использовать любые внешние данные
- Мало информации о турнирных матчах (~2000 строк)
- Преобладание feature engineering'a

# Данные

- Исторические данные с 1985 года: кто, когда, где и с кем сыграл

Season	Daynum	Wteam	Wscore	Lteam	Lscore	Wloc	Numot
1985	20	1228	81	1328	64	N	0
1985	25	1106	77	1354	70	H	0
1985	25	1112	63	1223	56	H	0
1985	25	1165	70	1432	54	H	0
1985	25	1192	86	1447	74	H	0

~145K строк для регулярного сезона  
~2K – для турниров

- Подробные данные с 2003 года, добавляется статистика матчей: броски, подборы, ...

Season	Daynum	Wteam	Wscore	Lteam	Lscore	Wloc	Numot	Wfgm	Wfga	...
2003	10	1104	68	1328	62	N	0	27	58	...
2003	10	1272	70	1393	63	N	0	26	62	...
2003	11	1266	73	1437	61	N	0	24	58	...
2003	11	1296	56	1457	50	N	0	18	38	...
2003	11	1400	77	1208	71	N	0	30	61	...

Lfga3	Lftm	Lfta	Lor	Ldr	Last	Lto	Lstl	Lblk	Lpf
10	16	22	10	22	8	18	9	2	20
24	9	20	20	25	7	12	8	6	16
26	14	23	31	22	9	12	2	5	23
22	8	15	17	20	9	19	4	3	23
16	17	27	21	15	12	10	7	1	14

~71K строк для регулярного сезона  
~850 – для турниров

# Elo rating system

[Wiki](#)

- Каждой команде присваивается начальный рейтинг, например, 1500.
- Для каждой команды считается матожидание выигранных очков (1 – победа, 0 – поражение):

$$E_A = \frac{1}{1 + 10^{(R_B - R_A)/400}} \quad E_B = \frac{1}{1 + 10^{(R_A - R_B)/400}}$$

- Обновляется рейтинг:

$$R'_A = R_A + K(S_A - E_A).$$

К – K-factor, чем меньше К, тем более консервативна система.

- Можно ввести поправки на игру дома, на разность в счёте.

# Другие рейтинговые системы

- [Glicko](#) – улучшенная версия Elo.
- [Chessmetrics](#) – попроще, но более чувствительна к «восходящим звёздам».
- [TrueSkill](#) – рейтинговая система от Microsoft.



# История встреч

- Пусть дана история матчей между двумя командами.

Введём для каждой из команд «вес», определяемый давностью её побед:

$$w1 = 0,5 + \sum wins\_1_n * \gamma^n \quad w2 = 0,5 + \sum wins\_2_n * \gamma^n$$

где  $wins\_1_n$  – количество побед первой команды над второй  $n$  лет назад,  $\gamma$  – коэффициент затухания.

Тогда можно сделать предсказание:

$$p1 = \frac{w1}{w1+w2} \quad p2 = \frac{w2}{w1+w2}$$

Например, если команды играли один раз в этом году и больше встреч никогда не было, вероятность повторной победы победителя оценивается в 0.75.

# Как формировать датасет

- Сырые данные нельзя просто так дать алгоритму

Season	Daynum	Wteam	Wscore	Lteam	Lscore	Wloc	Numot
1985	20	1228	81	1328	64	N	0
1985	25	1106	77	1354	70	H	0
1985	25	1112	63	1223	56	H	0
1985	25	1165	70	1432	54	H	0
1985	25	1192	86	1447	74	H	0

# Как формировать датасет

Train:

w_team	l_team	w_team features	l_team features	target
--------	--------	-----------------	-----------------	--------

# Как формировать датасет

Train:

w_team	l_team	w_team features	l_team features	1
l_team	w_team	l_team features	w_team features	0

# Как формировать датасет

Train:

w_team features	l_team features	delta features	1
l_team features	w_team features	- delta features	0

Test:

team_1	team_2	delta features	p1
team_2	team_1	- delta features	p2

$p1+p2$  не всегда равно 1. Например, для xgboost'а.

Тогда можно пересчитать по формулам:

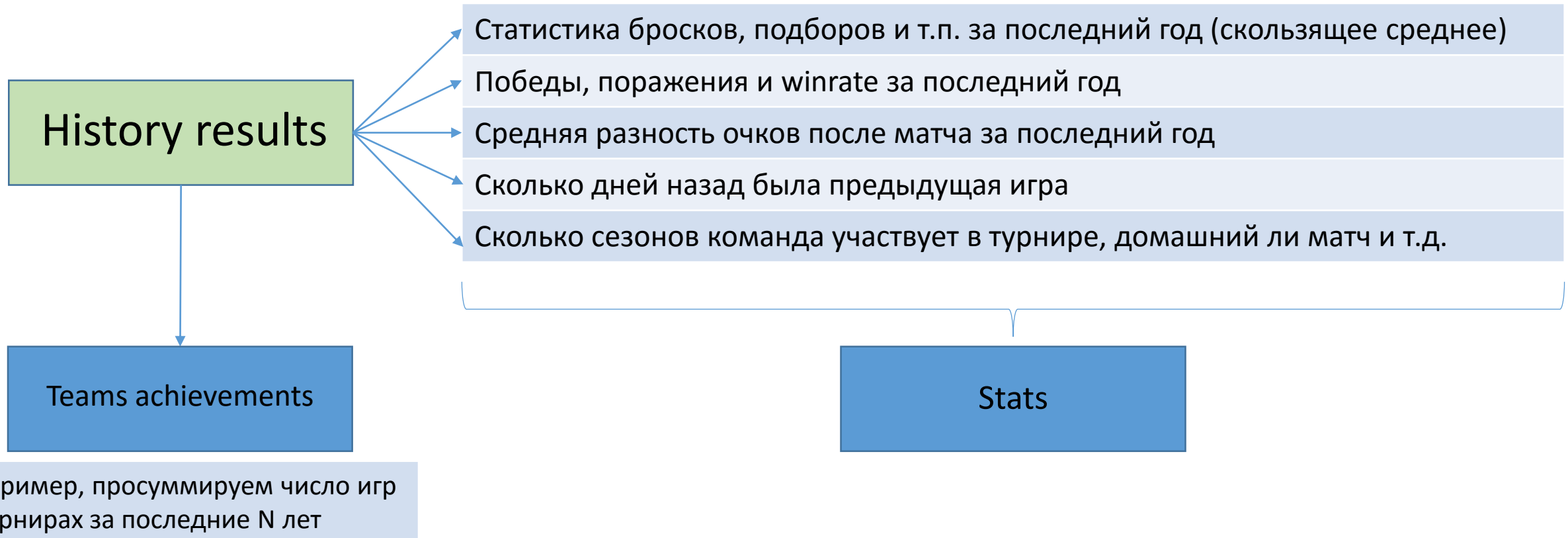
$$p1' = \frac{p1}{p1+p2}$$

$$p2' = \frac{p2}{p1+p2}$$

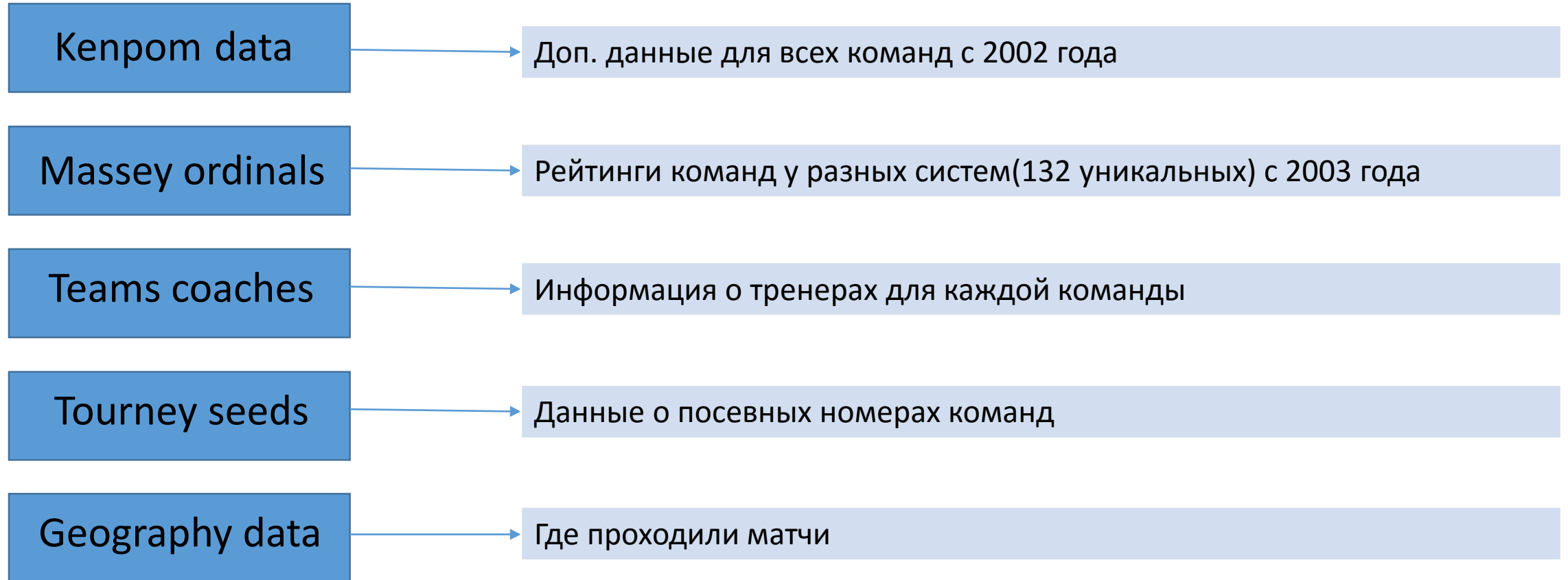
# Идея регрессии

- Пусть 1-я команда победила 2-ю с разницей  $\Delta$ , тогда целевые переменные будут равны  $+\Delta$  и  $-\Delta$  соответственно. Либо можно использовать  $1+0,03*\Delta$  и  $0-0,03*\Delta$ .
- Не теряется информация о том, насколько одна команда оказалась сильнее другой.

# Признаки

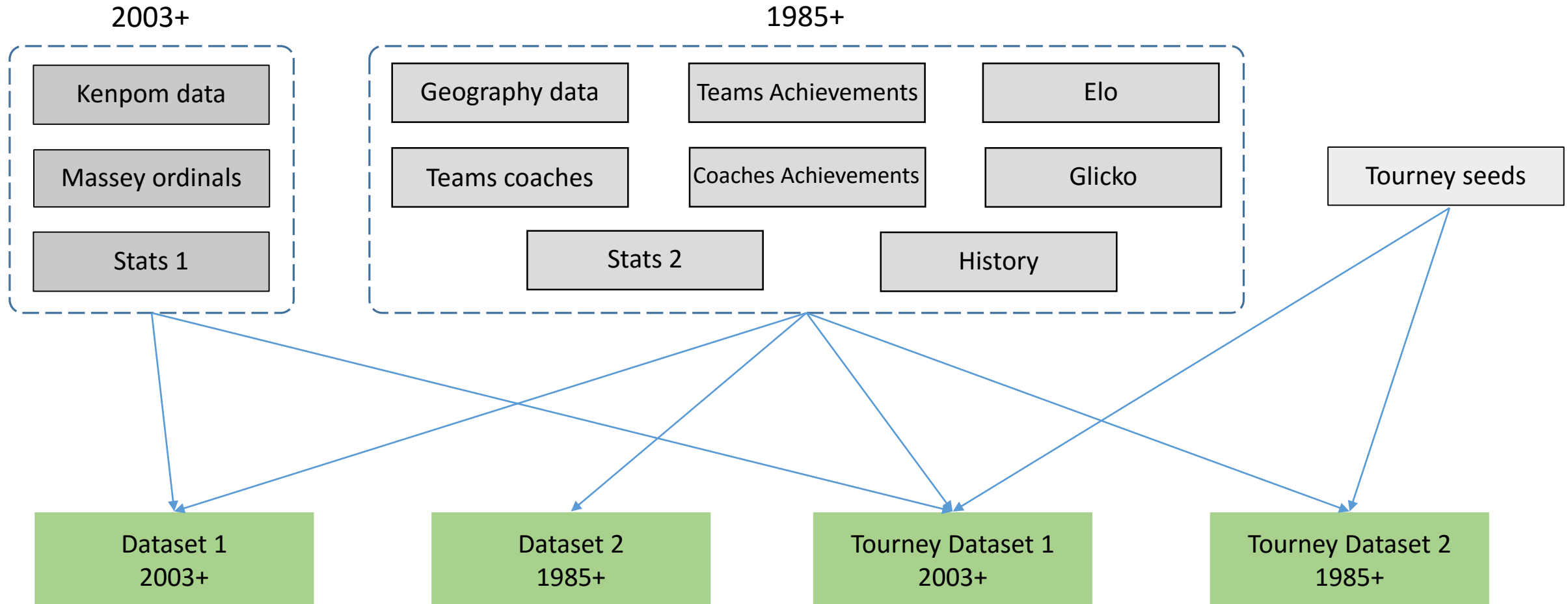


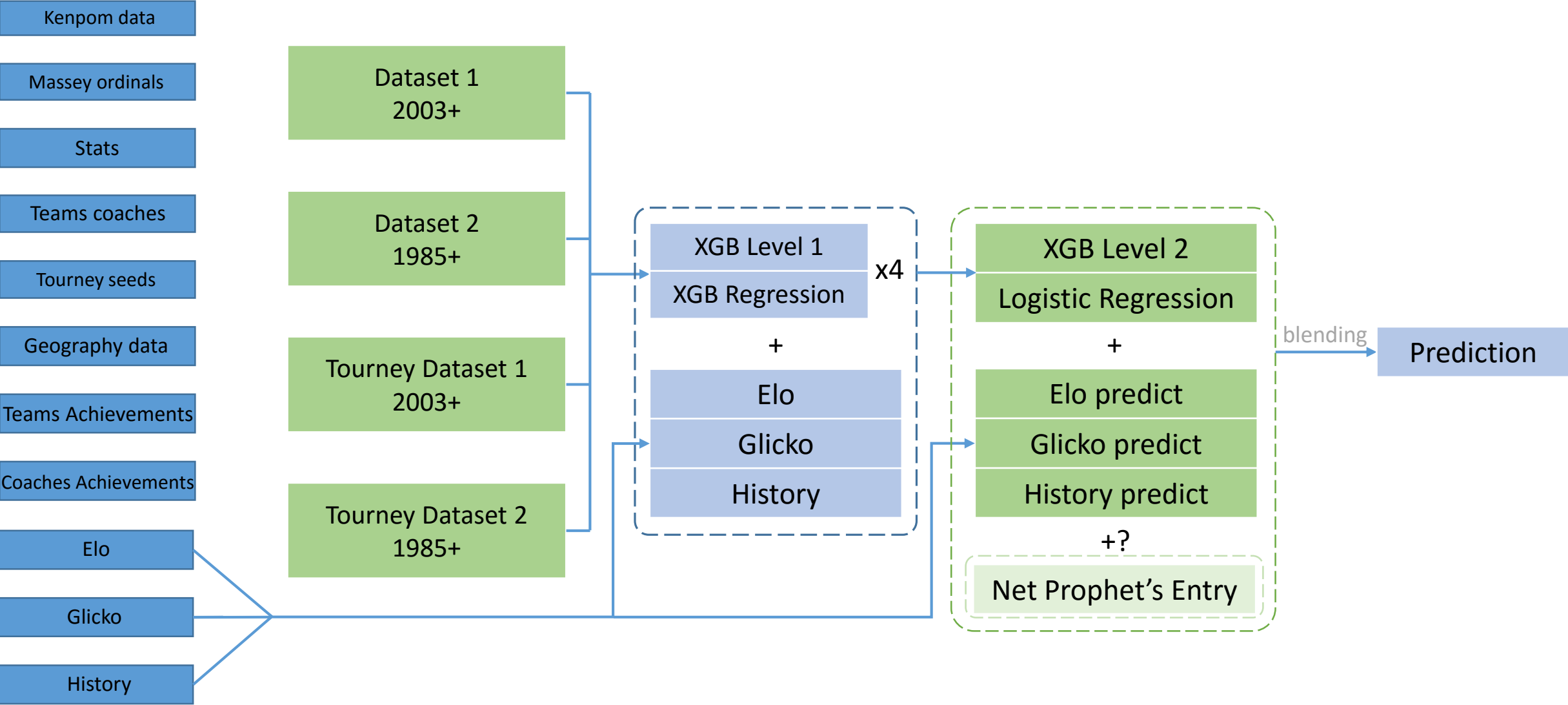
# Дополнительные данные





# Идея разных датасетов





# Наиболее важные признаки

- Географические данные
- Elo
- $\Delta(\Delta\text{score})$
- Предсказание по истории игр
- Рейтинги некоторых систем

# Анализ предсказаний участников

## 2016 March ML Mania Predictions

Forecasting the 2016 NCAA Basketball Tournament

by **William Cukierski** · Version 1 · last updated 1 month ago

Description Scripts Forum Download Data (27.75 MB) New Notebook **New Script**



### Scripts

<a href="#">Your Predictions vs the Field</a>	22 votes
<small>last run 1 month ago</small>	
<a href="#">Official First Round Predict...</a>	16 votes
<small>last run 1 month ago</small>	
<a href="#">Your Second Round vs the...</a>	7 votes
<small>last run 1 month ago</small>	

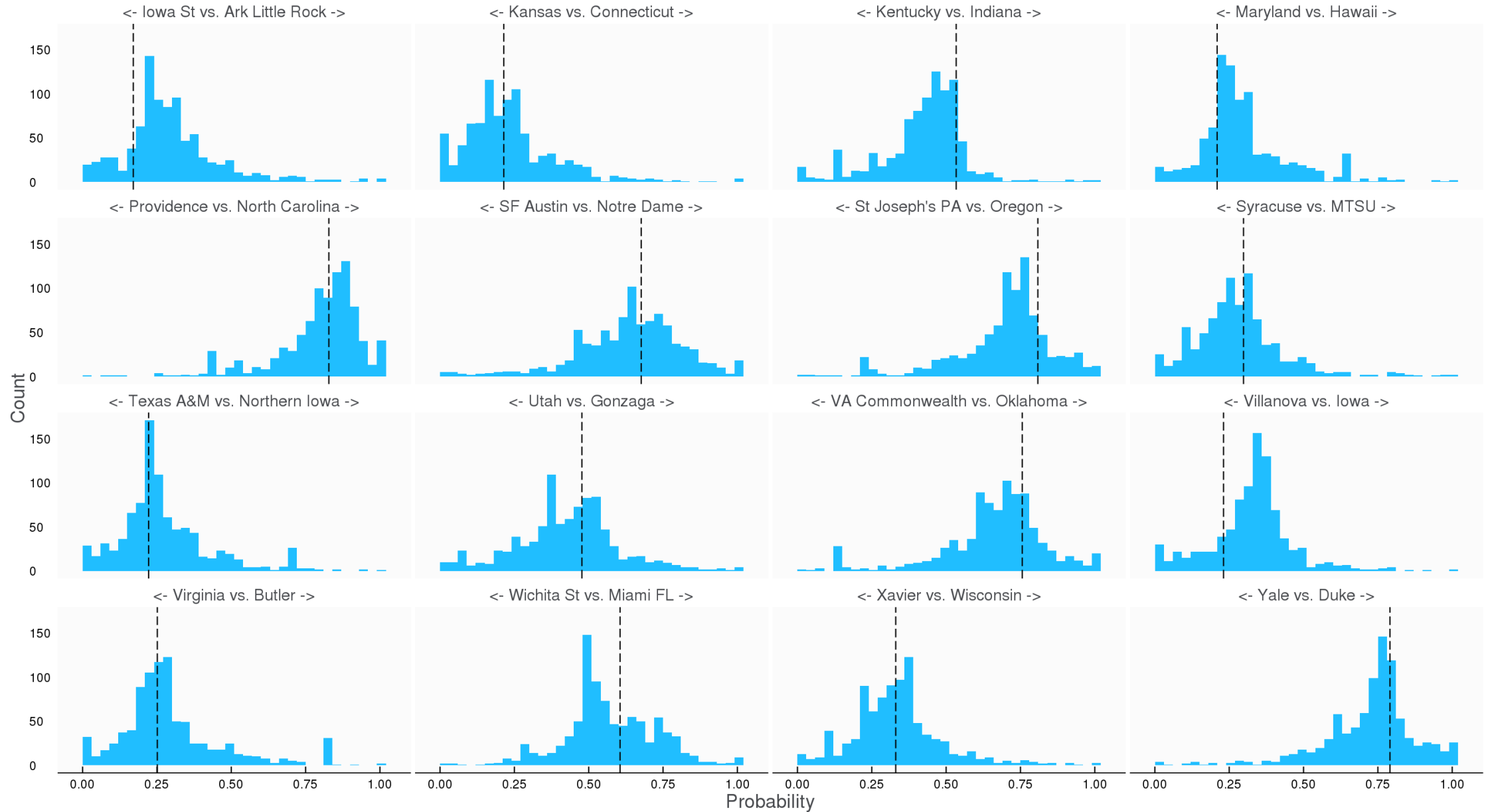
### Discussion

<a href="#">Official Championship Pre...</a>	0 replies
<small>3 weeks ago</small>	
<a href="#">Sweet Sixteen</a>	4 replies
<small>1 month ago</small>	
<a href="#">Your Predictions vs the Field</a>	7 replies
<small>1 month ago</small>	

### Top Contributors

	DrewWham	1st
	William Cukierski	2nd
	bepd50	3rd

# kaggle.com - March Machine Learning Mania 2016 (Second Round)



#	Δrank	Team Name <small>↑ model uploaded * in the money</small>	Score <small>👤</small>	Entries	Last Submission UTC (Best - Last Submission)
1	—	Mallorqui ‡ *	0.481310	2	Tue, 15 Mar 2016 13:36:41 (-0h)
2	—	JeremyJames *	0.500614	2	Wed, 16 Mar 2016 22:42:19
3	—	BAYZ ‡ *	0.508344	2	Wed, 16 Mar 2016 04:57:44
4	—	oneShiningMGF ‡ *	0.513189	2	Wed, 16 Mar 2016 22:44:39
5	—	Magic ‡ *	0.519706	2	Wed, 16 Mar 2016 02:25:19
6	—	Method 2 the Madness ‡	0.524432	1	Tue, 15 Mar 2016 18:15:28
7	—	LuckyGuesser ‡	0.525254	2	Tue, 15 Mar 2016 04:18:37 (-0.4h)
8	—	Glicko ‡	0.525337	2	Tue, 15 Mar 2016 01:08:31
9	—	Nicholas Canova	0.526241	2	Wed, 16 Mar 2016 20:36:50 (-0.1h)
10	—	Matt Morris	0.526970	2	Wed, 16 Mar 2016 20:29:08 (-0h)
11	—	vince0 ‡	0.528506	1	Tue, 15 Mar 2016 17:04:00
12	—	hairykrishna ‡	0.529270	2	Tue, 15 Mar 2016 17:01:55 (-1h)
13	—	thmavrid ‡	0.531271	2	Wed, 16 Mar 2016 20:31:36 (-0.5h)
14	—	Kevin Lee	0.531435	2	Wed, 16 Mar 2016 07:25:02 (-1.3h)
15	—	AdamGilfix	0.531449	2	Wed, 16 Mar 2016 06:07:00 (-0.3h)
16	—	Willie Liao ‡	0.531542	2	Wed, 16 Mar 2016 23:37:13
17	—	ex01 ‡	0.532063	2	Wed, 16 Mar 2016 20:32:15
18	—	JustDukelt 🏆	0.532262	2	Wed, 16 Mar 2016 21:34:20 (-0h)
19	—	Economom ‡	0.533121	2	Tue, 15 Mar 2016 02:56:35 (-0.1h)
20	—	Kieran ‡	0.533807	2	Wed, 16 Mar 2016 06:50:16
21	—	mlandry ‡	0.533870	2	Wed, 16 Mar 2016 09:29:59
22	—	mfontcada ‡	0.534064	2	Wed, 16 Mar 2016 19:50:03 (-2.6h)
23	—	Hack-a-bracket 🏆 ‡	0.534083	2	Wed, 16 Mar 2016 05:24:23 (-0.3h)
24	—	JustForFun 🏆 ‡	0.534261	2	Wed, 16 Mar 2016 17:07:53
25	—	DataArtist ‡	0.534413	2	Wed, 16 Mar 2016 00:00:17 (-0h)
26	—	Anton Smerdov ‡	0.535092	2	Wed, 16 Mar 2016 23:49:39 (-0.2h)
27	—	This bracket's gonna be YUUUUGE! ‡	0.535532	2	Wed, 16 Mar 2016 00:25:54
28	—	KadenHsu ‡	0.536344	1	Tue, 15 Mar 2016 18:13:29
29	—	Marin Kovacic ‡	0.536914	2	Wed, 16 Mar 2016 19:31:32

# Идеи на будущее

1. Добавить новую информацию (данные о ставках, игроках...)
2. Использовать алгоритмы: NeuralNets, KNN...
3. Можно оптимизировать не logloss, а матожидание выигрыша в деньгах или место на leaderboard
  - Проанализировать предсказания других участников
4. Придумать метод симуляции турнира
  - Поможет получить больше данных
  - Будет полезен для анализа предсказаний других участников