Predicting UFC Fight Length

And other variables, by Titus Bridgwood & Ioana Preoteasa kaggle

Predicting UFC Fight Length

And other variables, by Titus Bridgwood & Ioana Preoteasa

UFC

Target audience and deliverables

- Bookies wanting to bet on over-1.5 and under-1.5 fights
- Gym coaches and talent scouts
- MMA enthusiasts

Target audience and deliverables

- Predicting **duration** of fight
- Predicting various metrics of fighter success

How long must we fight?

- Using historical fight data from 1993-2019
- Prior fight data about fighters, e.g. :
  - Prior of wins by KO/TKO, submission
  - Win/loss streaks
  - Heights & Weights of fighters

Our Approach

Clean and Scrub

- Clean our data
- Think about features that we might want to include
- Multicollinearity
- Deciding on UFC21 cutoff

Split & Scale

- Create and training and test split
- **80% - 20%**

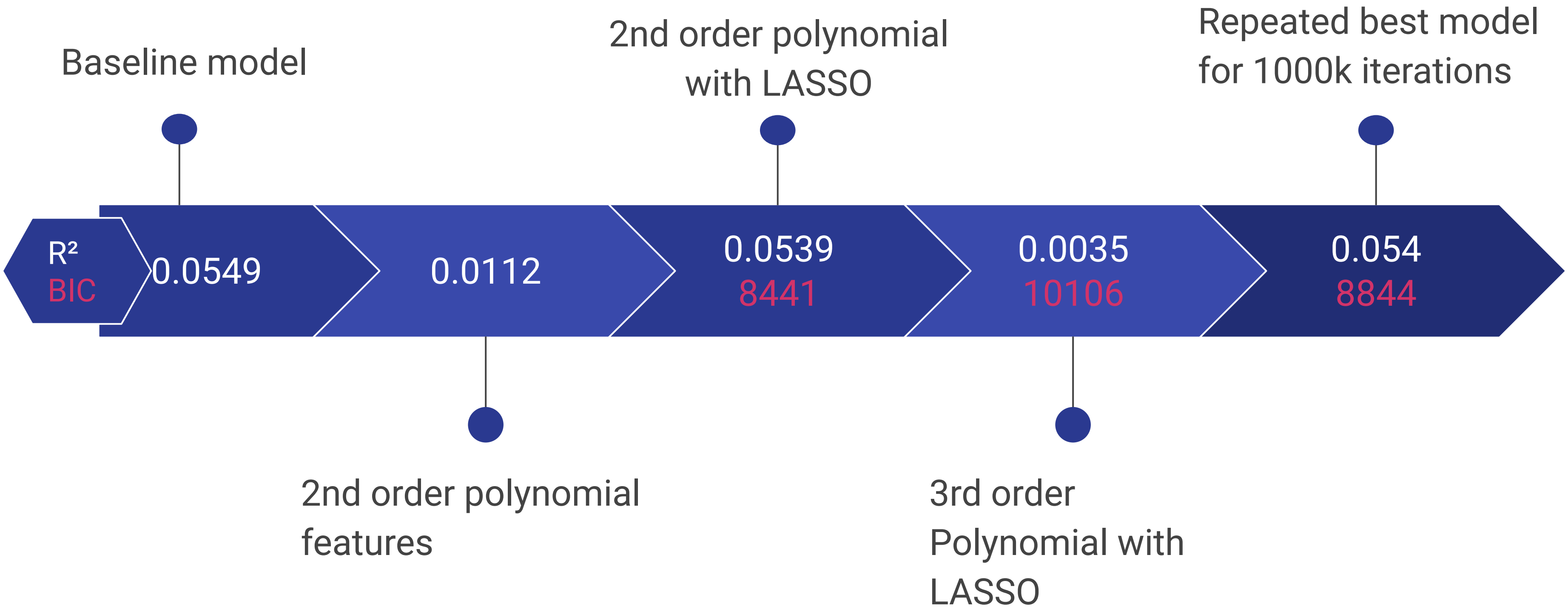
Baseline

- Produce a baseline linear regression from which to improve
- Test baseline model on other variables

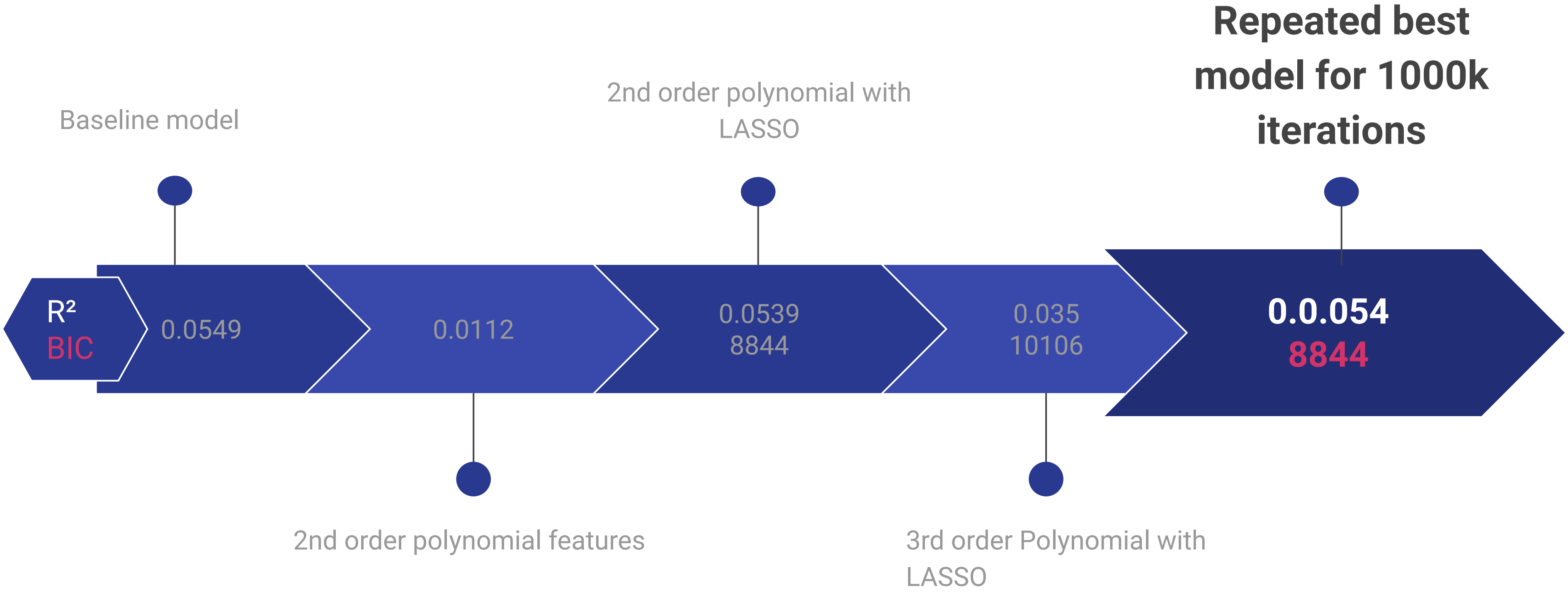
Improve

- We introduce complexity and refine our model features
- Validation and selection

Iterative approach



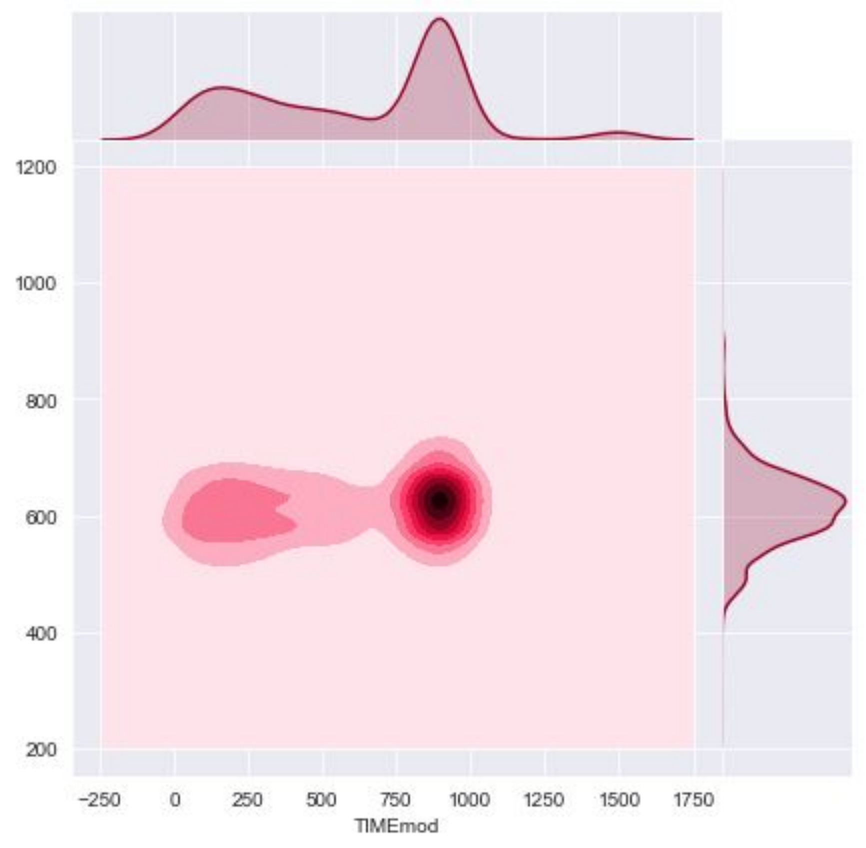
A winning model



Conclusion

- None of the pre-fight variables were good predictors of fight duration. This was attempted both with and without the pre-UFC21 data points
- Even excluding finishing and title bouts did not improve the model

Visual explanation

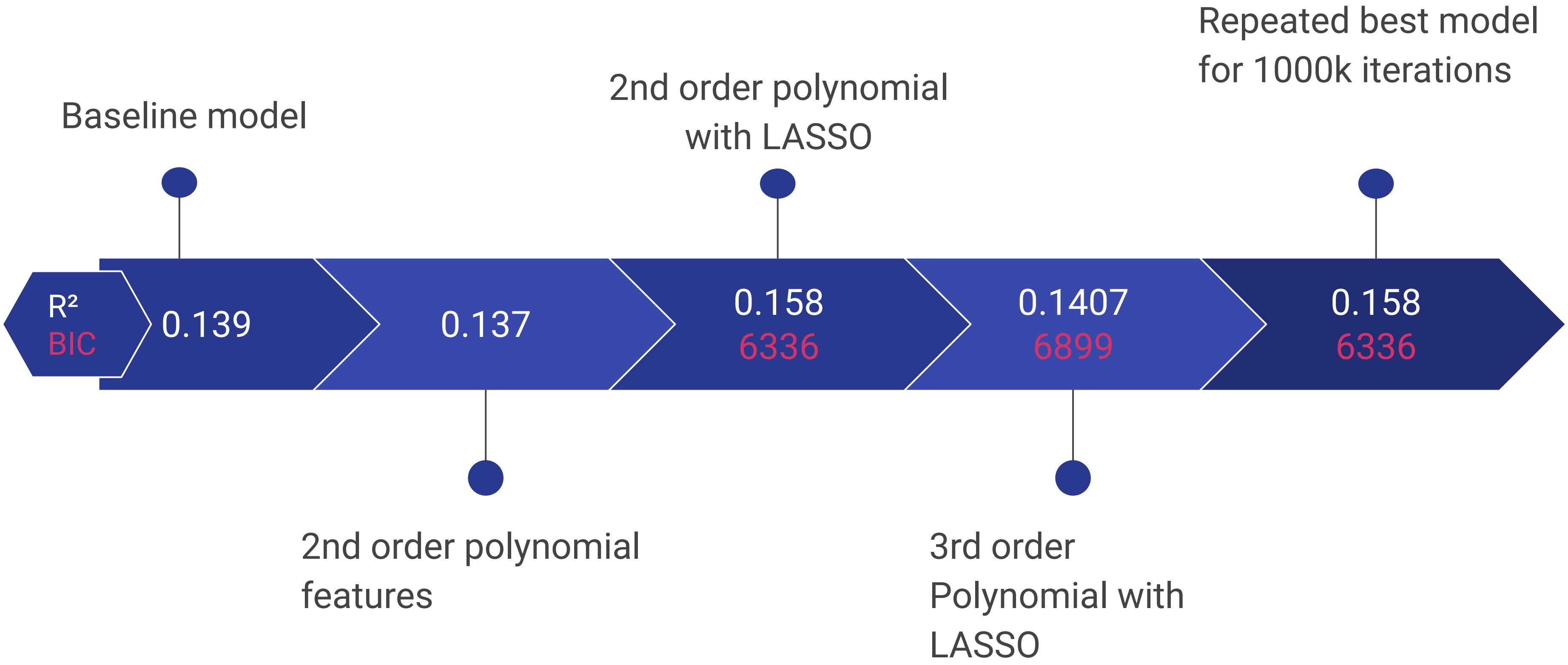


- Non-normal distribution of y-test values
- Some overlap around mean

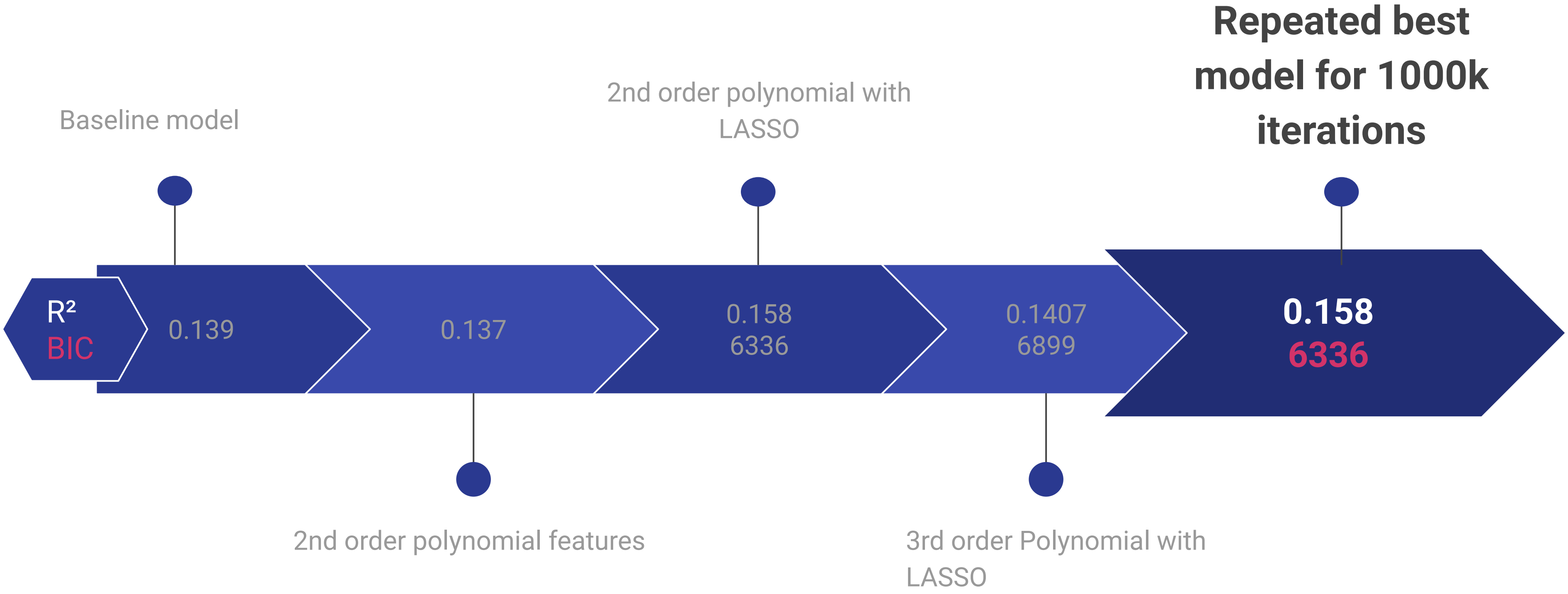
How well will we fight?

- Utilize same data available prior to a fight, we try to predict:
    - % of significant strikes successfully landed by fighter
    - Useful metric for MMA gurus, experts and for gyms looking for talent to hire for their stable

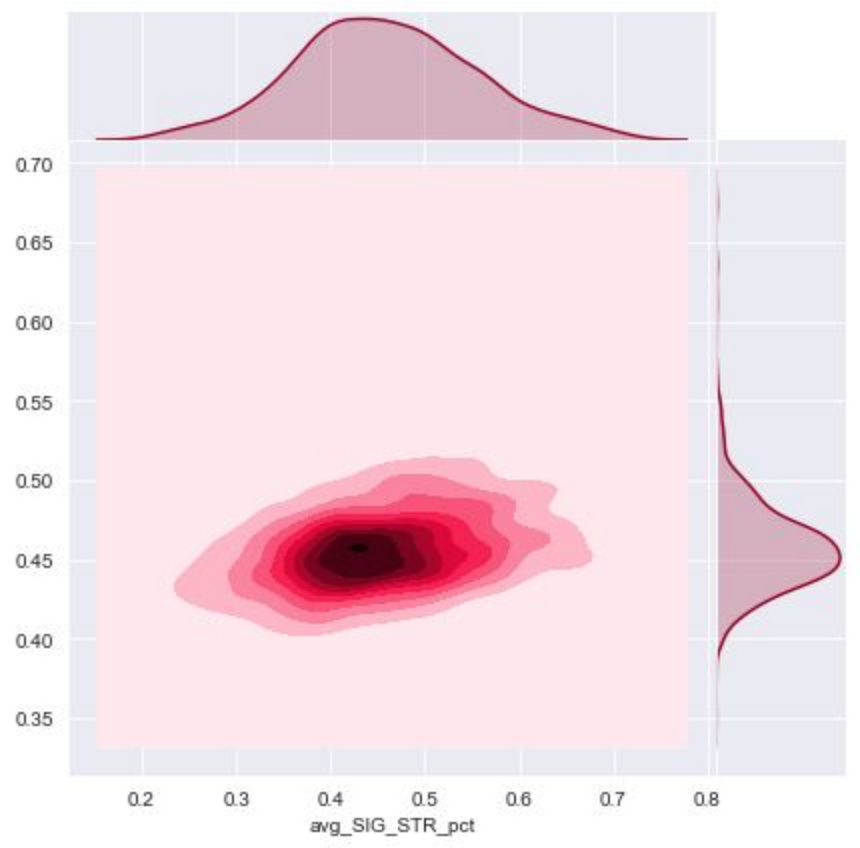
Iterative approach

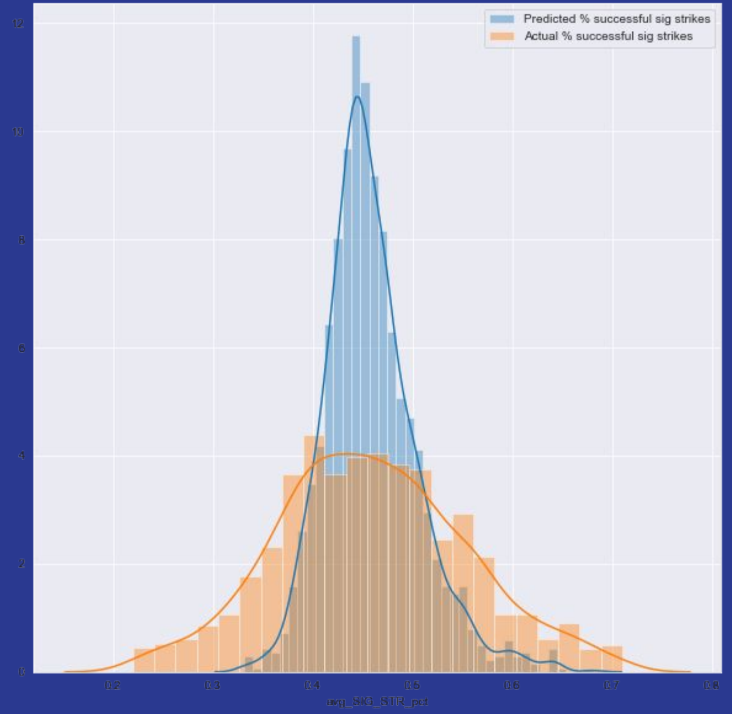


A winning model

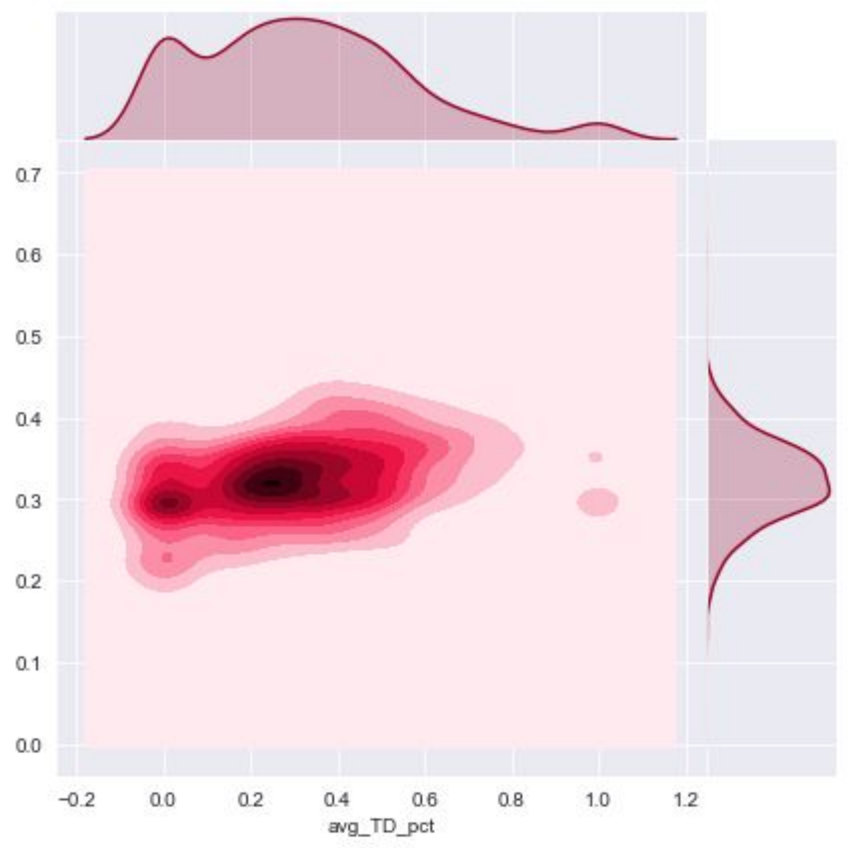


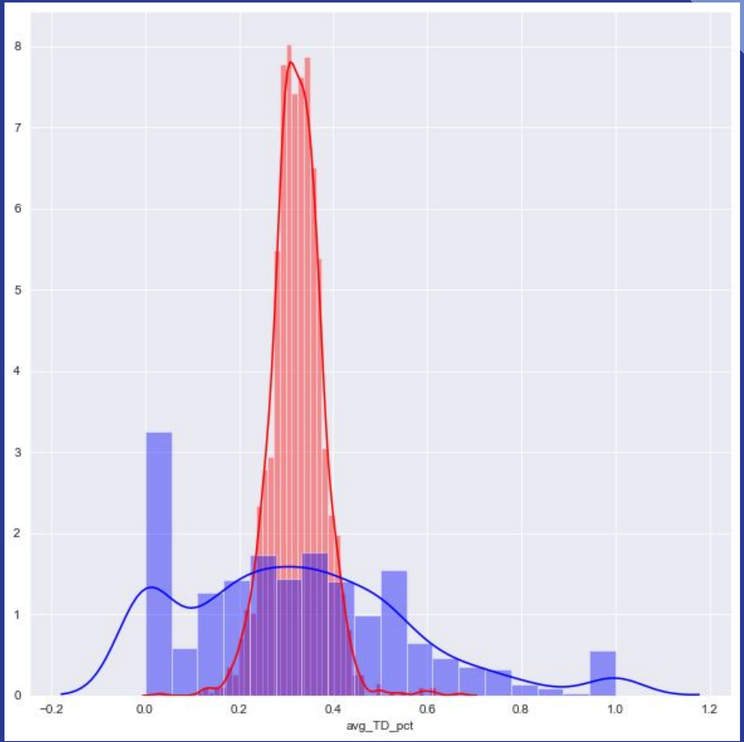
Visual explanation





Same but for Takedown % (R2 0.095)

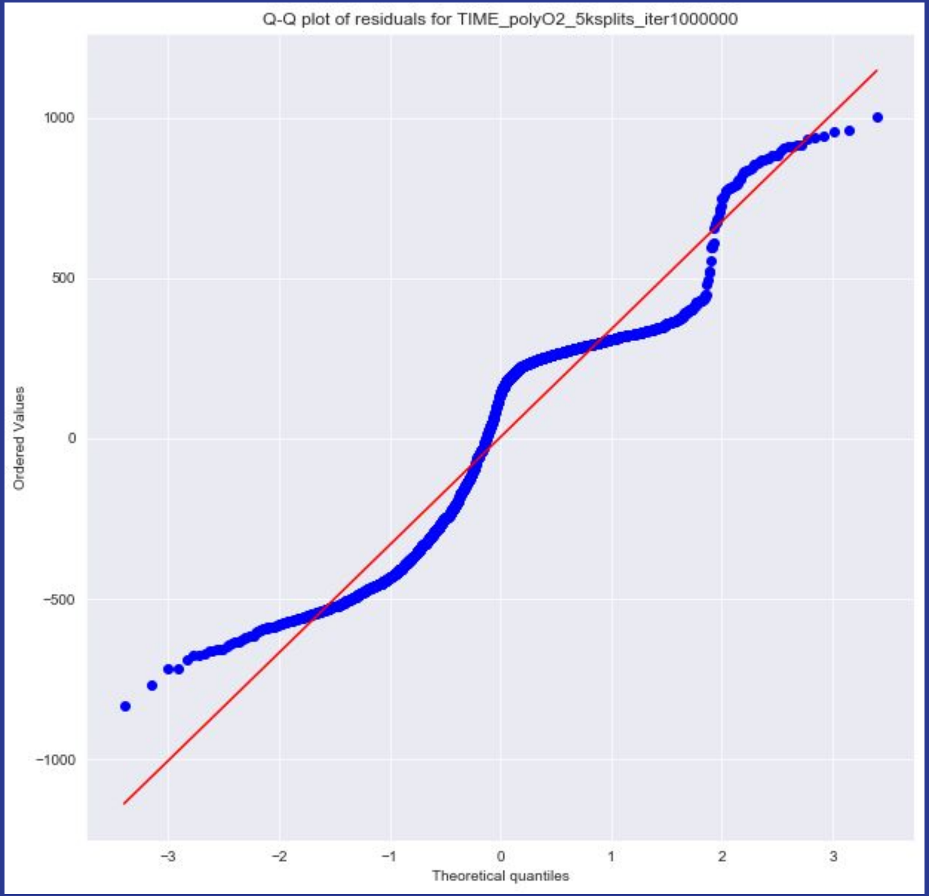




Final conclusion

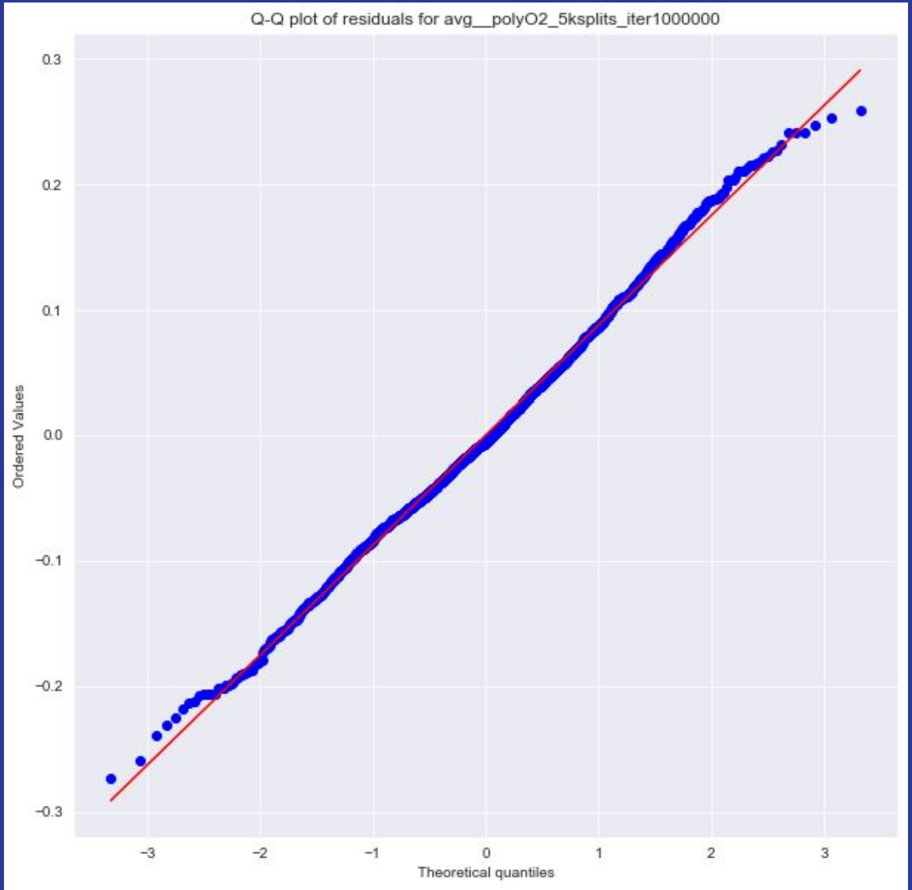
1. Limitations of linear regression with LASSO
2. Future iterations of project using clustering and classification methods
3. Ultimately other exogenous variables must be introduced: e.g. training regime, fighter diet, gym.

Any questions?

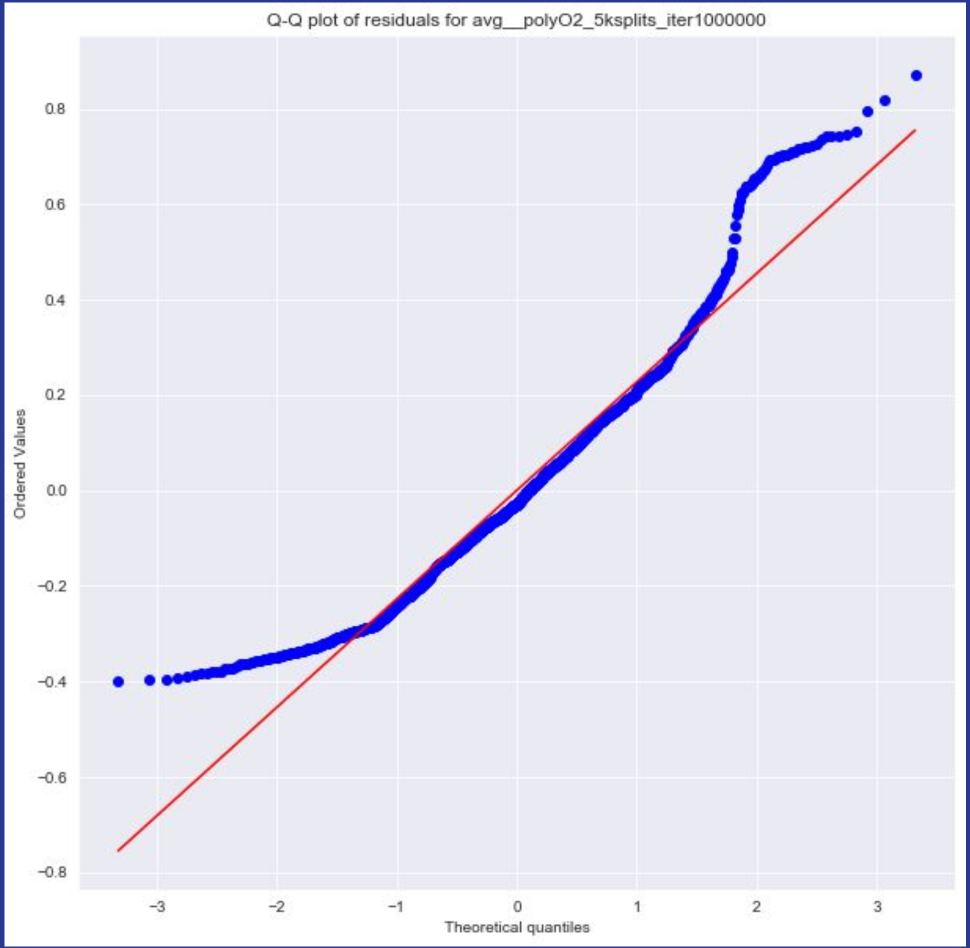


Q-Q plot of residuals for TIME_polyO2_5ksplits_iter1000000
Ordered Values
1000
500
0
−500
−1000
−3
−2
−1
0
1
2
3
Theoretical quantiles

	R2	bic	Optimal_alpha	Mean_cvs
TIME_polyO1_5ksplits	0.054977	NaN	NaN	0.067798
TIME_polyO2_5ksplits	0.011290	NaN	NaN	-0.451234
TIME_polyO2_5ksplits	0.026338	NaN	NaN	0.023284
TIME_polyO2_5ksplits	0.053927	8844.141471	16.996733	0.055992
TIME_polyO3_5ksplits	0.035014	10106.220633	50.853286	-0.112056
TIME_polyO2_5ksplits_iter1000000	0.053927	8844.141471	16.996733	0.055992


Q-Q plot of residuals for avg__polyO2_5ksplits_iter1000000
Ordered Values
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−3
−2
−1
0
1
2
3
Theoretical quantiles

	R2	bic	Optimal_alpha	Mean_cvs
avg__polyO1_2ksplits	0.139112	NaN	NaN	0.109887
avg__polyO1_5ksplits	0.137528	5760.263966	0.000223	0.104738
avg__polyO2_5ksplits	0.158222	6336.460891	0.002696	0.113615
avg__polyO3_5ksplits	0.140723	6899.923302	0.006092	0.040550
avg__polyO2_5ksplits_iter1000000	0.158222	6336.460891	0.002696	0.113615


Q-Q plot of residuals for avg__polyO2_5ksplits_iter1000000
Ordered Values
Theoretical quantiles
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−3
−2
−1
0
1
2
3

	R2	bic	Optimal_alpha	Mean_cvs
avg__polyO1_2ksplits	0.080793	NaN	NaN	0.083822
avg__polyO1_5ksplits	0.081109	5887.302642	0.000138	0.078200
avg__polyO2_5ksplits	0.088838	6539.150056	0.007406	0.095000
avg__polyO3_5ksplits	0.020973	7522.576689	0.049496	0.022712
avg__polyO2_5ksplits_iter1000000	0.088838	6539.150056	0.007406	0.095000