NBA regular season MVP Prediction

Yutong Ren (Tony)

What is MVP

The NBA Most Valuable Player Award (MVP) is an annual award given since the 1955–56 season to the best performing player of the regular season.

Voting: 101 ballots: 100 media

Add text	Add text	Add text	Add text	Add text
1st position	2nd position	3rd position	4th position	5th position
10 points	7 points	5 points	3 points	1 point
Yutong Ren	Yutong Ren	Yutong Ren	Yutong Ren	Yutong Ren

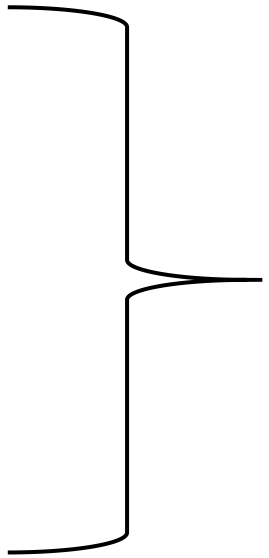
Winner: Highest accumulative points

MVP selection of voters

Personal Stats

Team's overall success

Athlete's health and regularity



Statistic Data

Media narrative

Motivation: Is the MVP really the MVP?

Data Set

Basketball reference:
https://www.basketball-reference.com/awards/mvp.html

Stats of MVP candadites in the previous seasons.
685 rows, 23 columns (1980-2021 MVP candadites stats)

year	rank	player	age	team	award_share	games	mp_per_g	pts_per_g	trb_per_g	ast_per_g	stl_per_g	blk_per_g	fg_pct	fg3_pct	ft_pct	ws	ws_per_48	W	L	W/L	seed	MVP
1980	1	Kareem Abdul-Jabbar	32	LAL	0.665	82	38.3	24.8	10.8	4.5	1.0	3.4	0.604	0	0.765	14.8	0.227	60	22	0.732	2	T

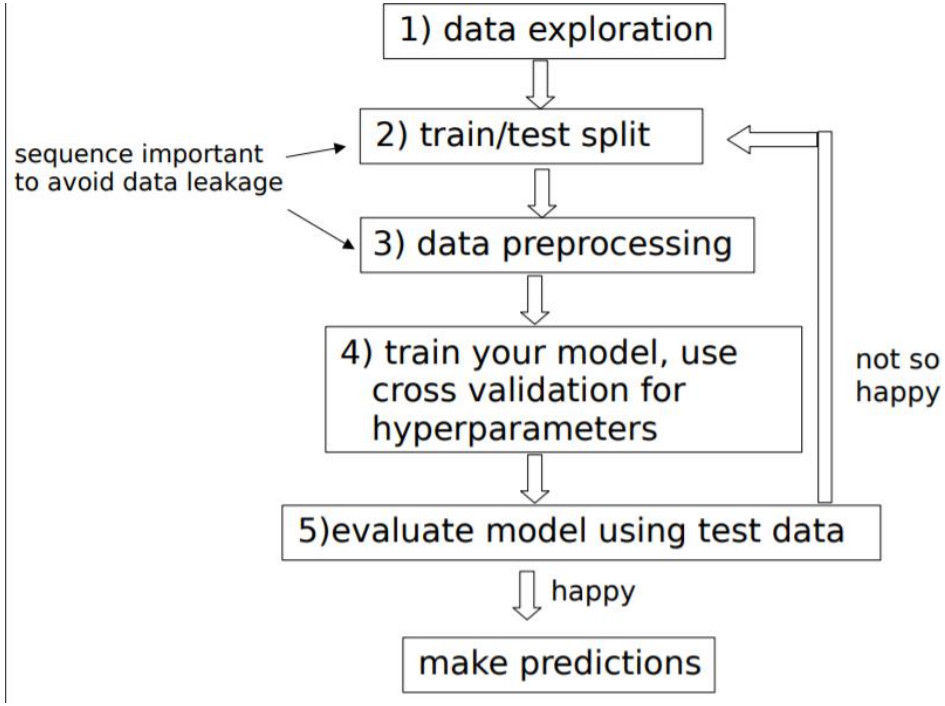
Stats of 2022 MVP candidates
10 rows, 20 columns

year	player	age	team	games	mp_per_g	pts_per_g	trb_per_g	ast_per_g	stl_per_g	blk_per_g	fg_pct	fg3_pct	ft_pct	ws	ws_per_48	W	L	W/L	seed
2022	Nikola Jokic	27	DEN	57	33.1	25.8	13.8	8.0	1.4	0.8	0.571	0.363	0.811	11.7	0.299	38	26	0.594	10

Machine Learning method: Regression and Classification

Classification: Target variable: MVP: {"T", "F"}

Regression: Target variable: Award Share



Reference: Stats302 slides

Classification Method

Data Exploration

Set classification featureas and target variable (MVP: {“T”, “F”})

age	int64
games	int64
mp_per_g	float64
pts_per_g	float64
trb_per_g	float64
ast_per_g	float64
stl_per_g	float64
blk_per_g	float64
fg_pct	float64
fg3_pct	float64
ft_pct	float64
ws	float64
ws_per_48	float64
W	int64
L	int64
W/L	float64
seed	int64

age	0
games	0
mp_per_g	0
pts_per_g	0
trb_per_g	0
ast_per_g	0
stl_per_g	0
blk_per_g	0
fg_pct	0
fg3_pct	25
ft_pct	0
ws	0
ws_per_48	0
W	0
L	0
W/L	0
seed	0

Train Test Split: 0.6:0.4

Classification

Preprocessing

Filled the nan values by mean in train and test data

Standardize train and test data

```
array([[0.23279216, 0.7252371 , 0.28472271, ..., 0.28651342, 0.00546166,
        0.07162836],
       [0.21541981, 0.69796019, 0.31709796, ..., 0.2240366 , 0.00588527,
        0.02585038],
       [0.19357448, 0.72820876, 0.31156273, ..., 0.42402029, 0.00404663,
        0.15670315],
       ...,
       [0.17038904, 0.73536322, 0.32732631, ..., 0.34974592, 0.00469915,
        0.09864629],
       [0.22190529, 0.72784934, 0.30800454, ..., 0.3106674 , 0.00508607,
        0.0798859 ],
       [0.25540437, 0.66215947, 0.31594466, ..., 0.24594495, 0.00646078,
        0.04729711]])
```

Classification: High precision

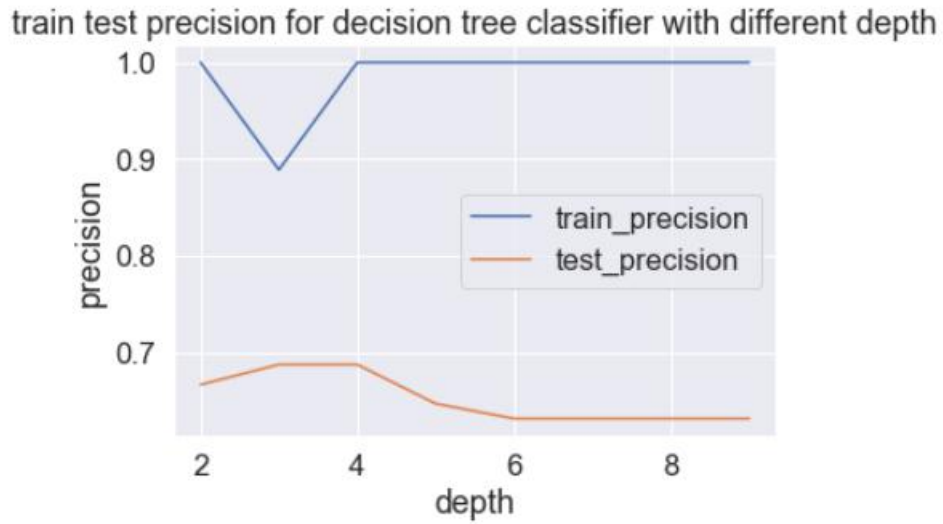
$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total}}$$

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positives} + \text{False Positive}}$$

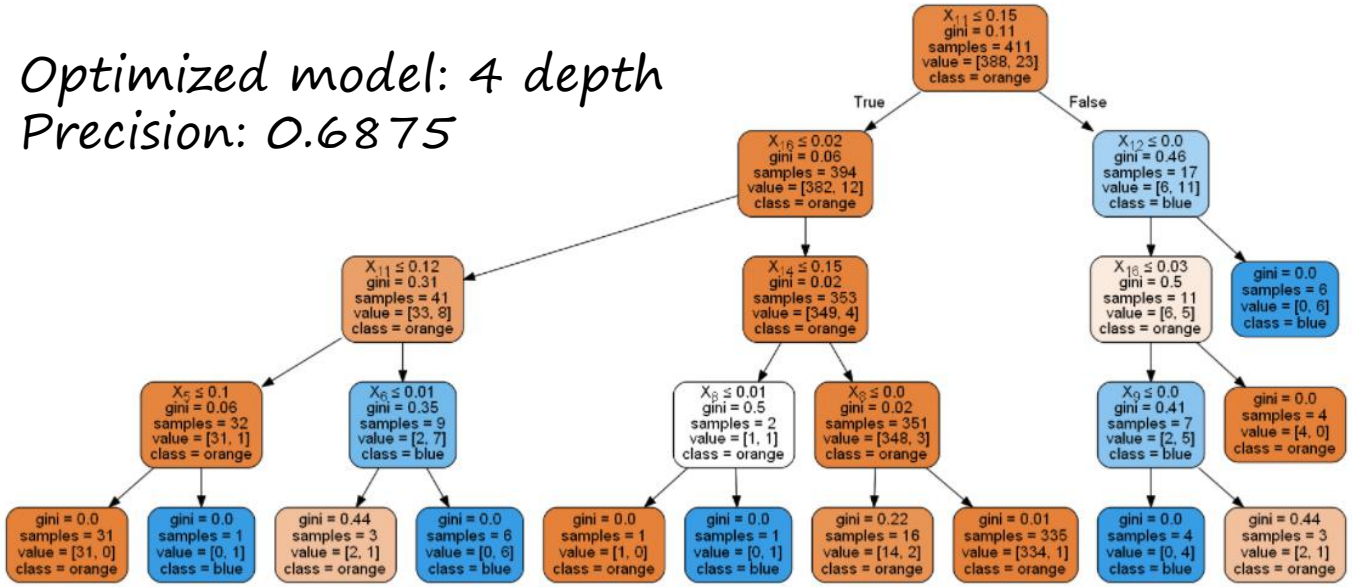
$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positives} + \text{False Negative}}$$

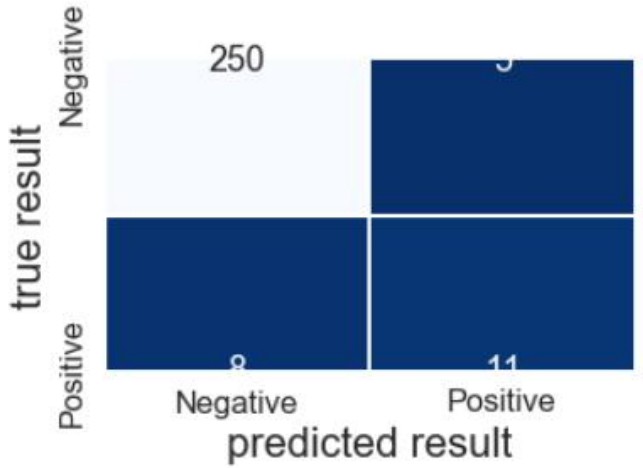
Reference: https://blog.gitguardian.com/secrets-detection-accuracy-precision-recall-explained/

Decision Tree for classification



Optimized model: 4 depth
Precision: 0.6875



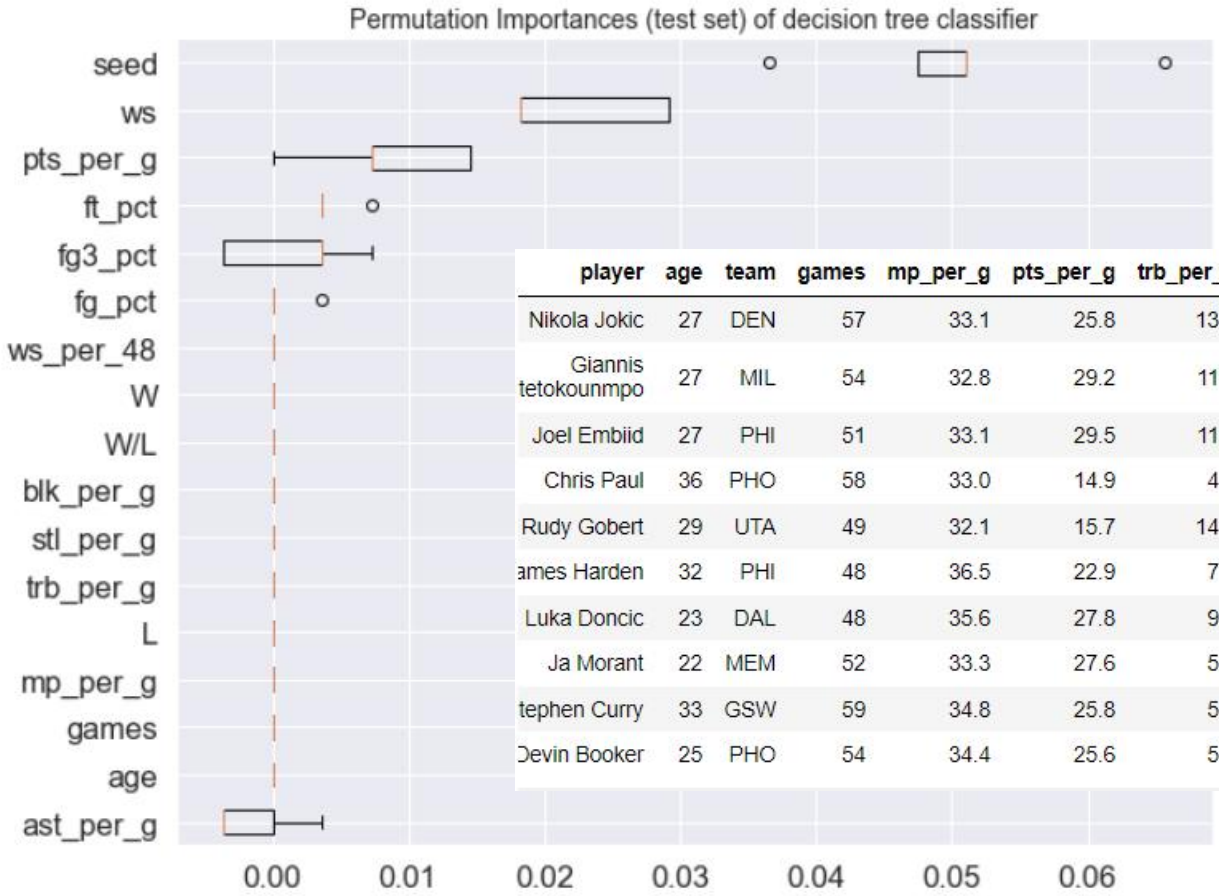


Confusion matrix of test set

	candidates	predicted_MVP
0	Nikola Jokic	F
1	Giannis Antetokounmpo	F
2	Joel Embiid	F
3	Chris Paul	T
4	Rudy Gobert	F
5	James Harden	F
6	Luka Doncic	F
7	Ja Morant	F
8	Stephen Curry	F
9	Devin Booker	F

2022 predicted MVP

Feature Importance of decision tree



player	age	team	games	mp_per_g	pts_per_g	trb_per_g	ast_per_g	stl_per_g	blk_per_g	fg_pct	fg3_pct	ft_pct	ws	ws_per_48	W	L	W/L	seed
Nikola Jokic	27	DEN	57	33.1	25.8	13.8	8.0	1.4	0.8	0.571	0.363	0.811	11.7	0.299	38	26	0.594	10
Giannis tetokounmpo	27	MIL	54	32.8	29.2	11.6	6.0	1.0	1.4	0.544	0.296	0.722	10.6	0.288	40	25	0.615	8
Joel Embiid	27	PHI	51	33.1	29.5	11.2	4.4	1.1	1.4	0.490	0.351	0.822	9.2	0.262	39	24	0.619	11
Chris Paul	36	PHO	58	33.0	14.9	4.5	10.7	1.9	0.3	0.487	0.330	0.843	8.5	0.212	51	13	0.797	1
Rudy Gobert	29	UTA	49	32.1	15.7	14.8	1.2	0.7	2.2	0.711	0.000	0.682	8.8	0.270	40	23	0.635	3
ames Harden	32	PHI	48	36.5	22.9	7.9	10.3	1.3	0.6	0.425	0.344	0.872	1.4	0.461	39	24	0.619	11
Luka Doncic	23	DAL	48	35.6	27.8	9.2	8.8	1.2	0.5	0.450	0.332	0.739	5.1	0.143	39	25	0.609	9
Ja Morant	22	MEM	52	33.3	27.6	5.8	6.6	1.2	0.4	0.494	0.343	0.761	6.4	0.177	44	22	0.667	6
tephen Curry	33	GSW	59	34.8	25.8	5.2	6.4	1.3	0.4	0.432	0.379	0.918	7.7	0.180	43	21	0.672	2
Devin Booker	25	PHO	54	34.4	25.6	5.2	4.6	1.1	0.4	0.448	0.368	0.864	5.2	0.134	51	13	0.797	1

ANN for classification

Traing data: MVP: T: 23

MVP: F: 456

Imbalanced data in training set.

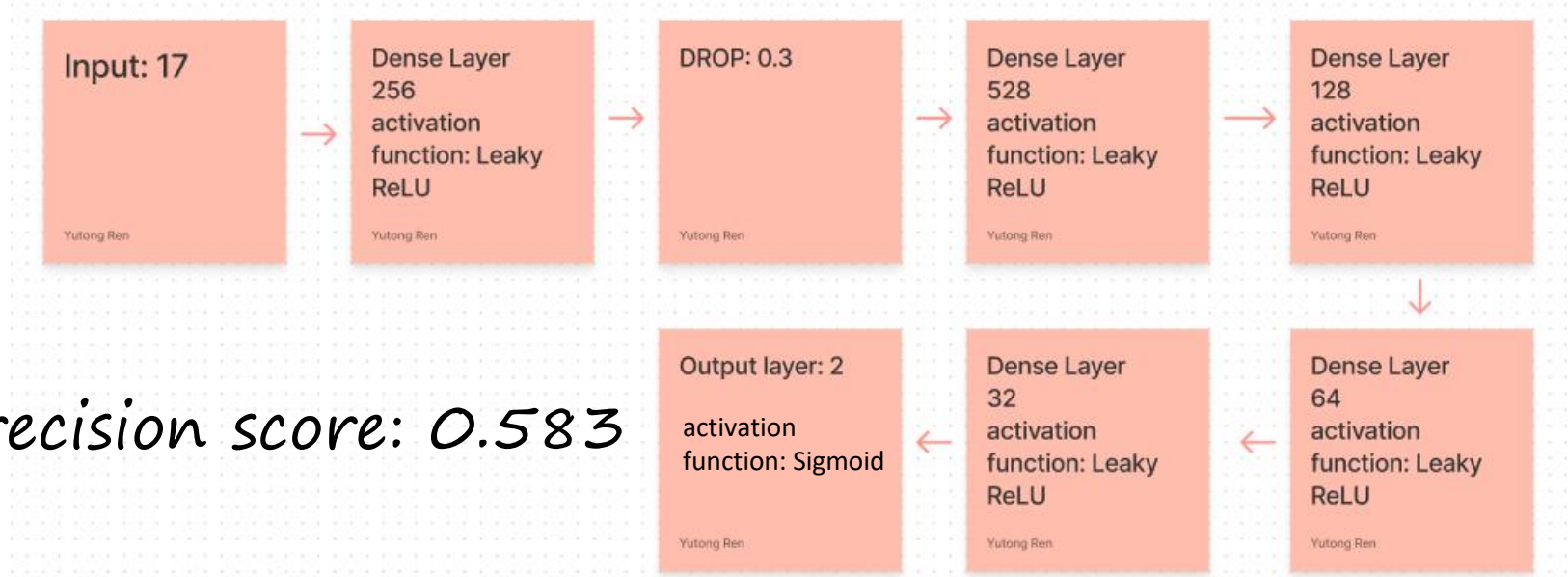
Implement SMOTE

Train data with smote: MVP: T: 456

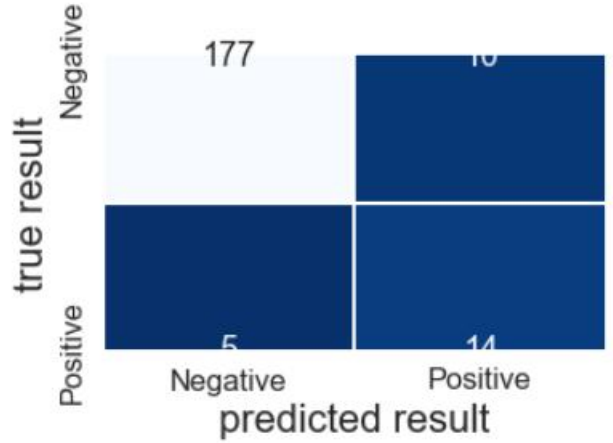
MVP: F: 456

Use one hot encoder to transform categorical data MVP {T,F} to {[1,0],[0,1]}

ANN Structure



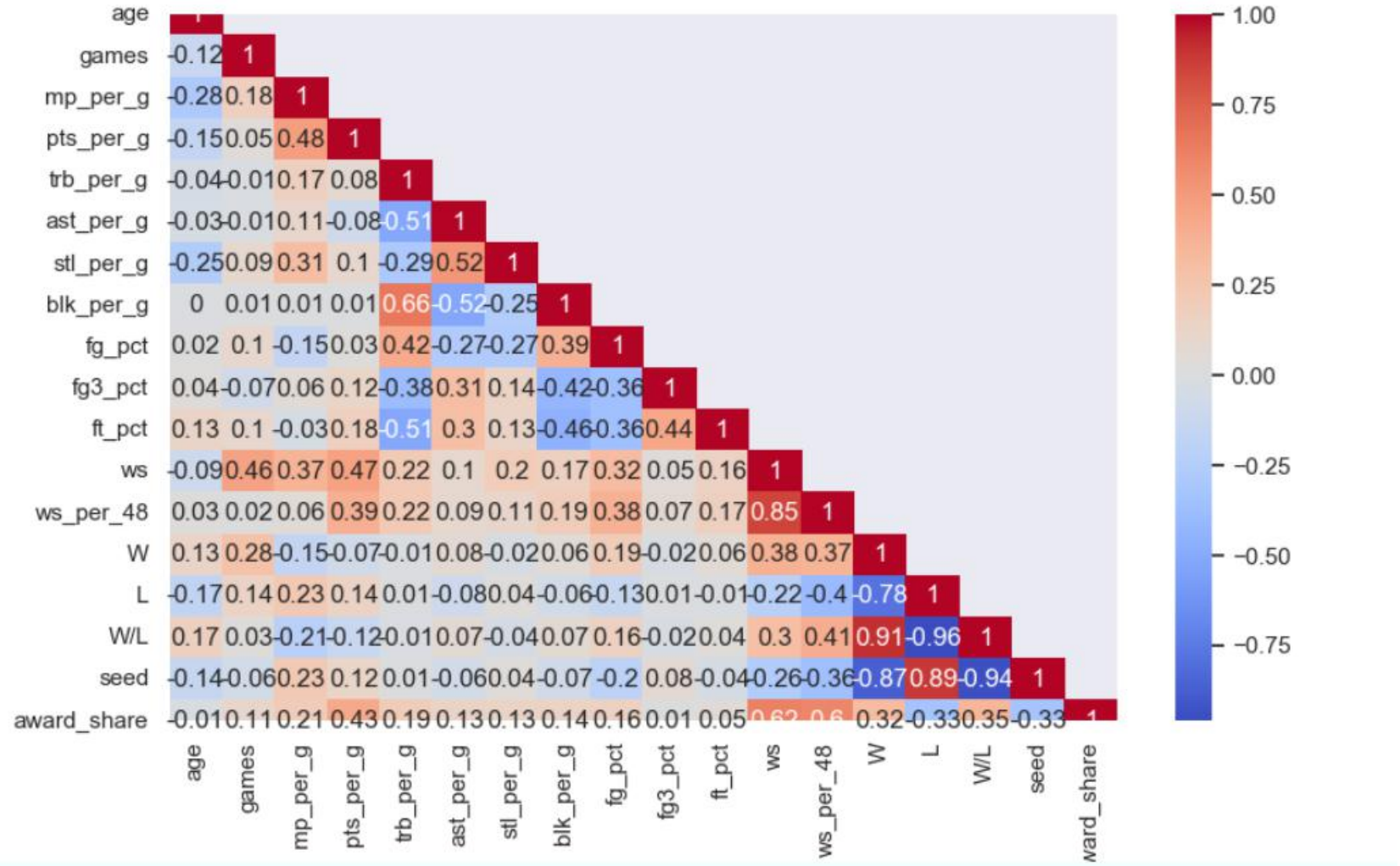
Precision score: 0.583



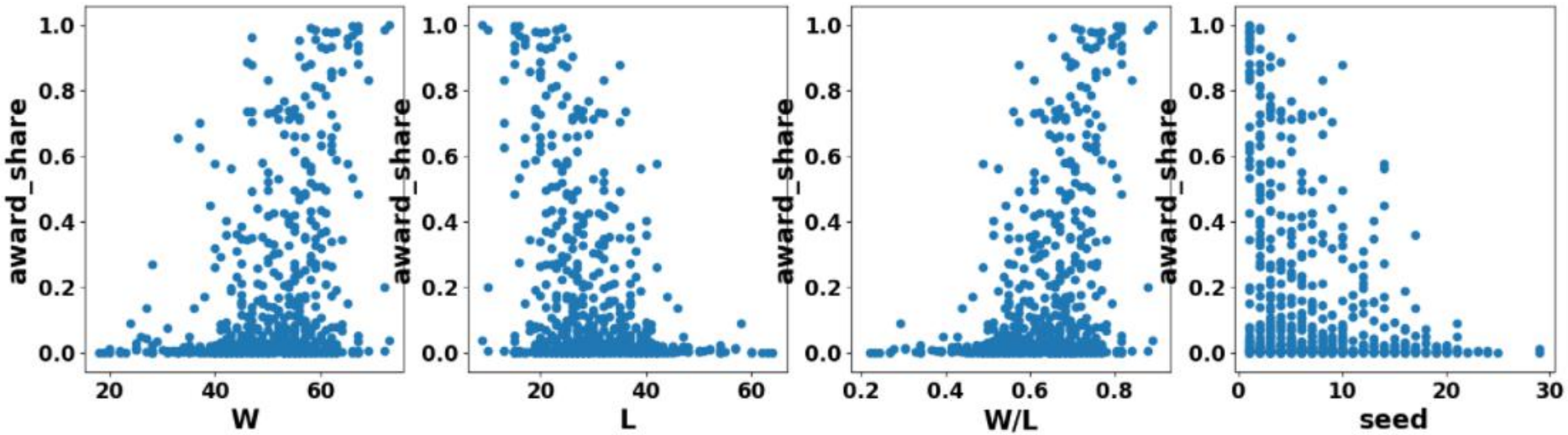
	player	predicted MVP
0	Nikola Jokic	T
1	Giannis Antetokounmpo	F
2	Joel Embiid	F
3	Chris Paul	F
4	Rudy Gobert	F
5	James Harden	F
6	Luka Doncic	F
7	Ja Morant	F
8	Stephen Curry	F
9	Devin Booker	F

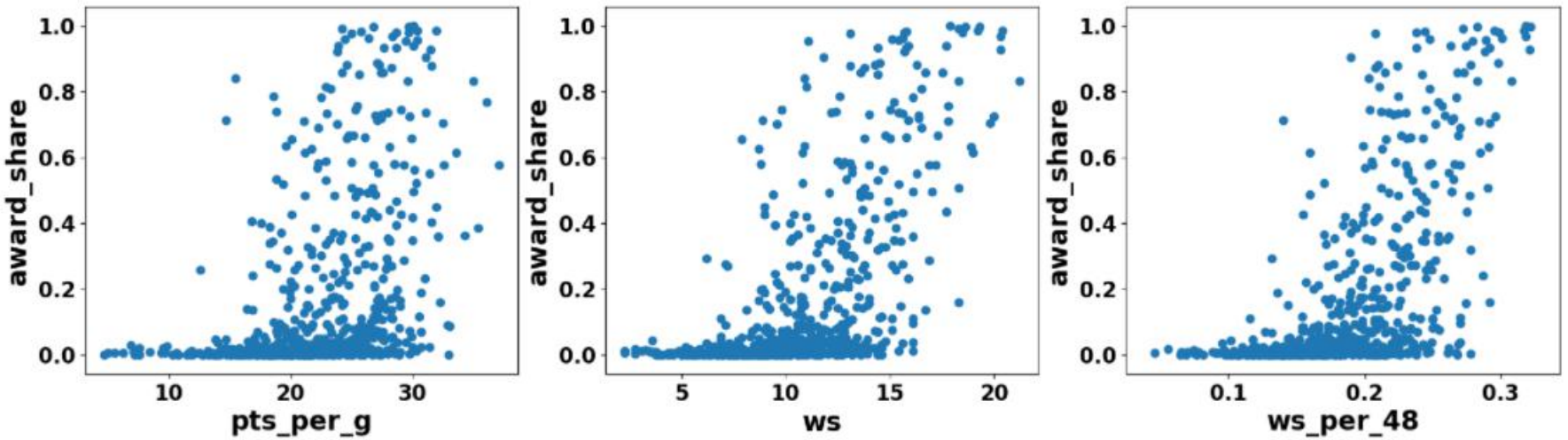
Regression Method

Data Exploration



Plot the features correlation > |0.3| with award share





Abandon 'seed', keep the other six features

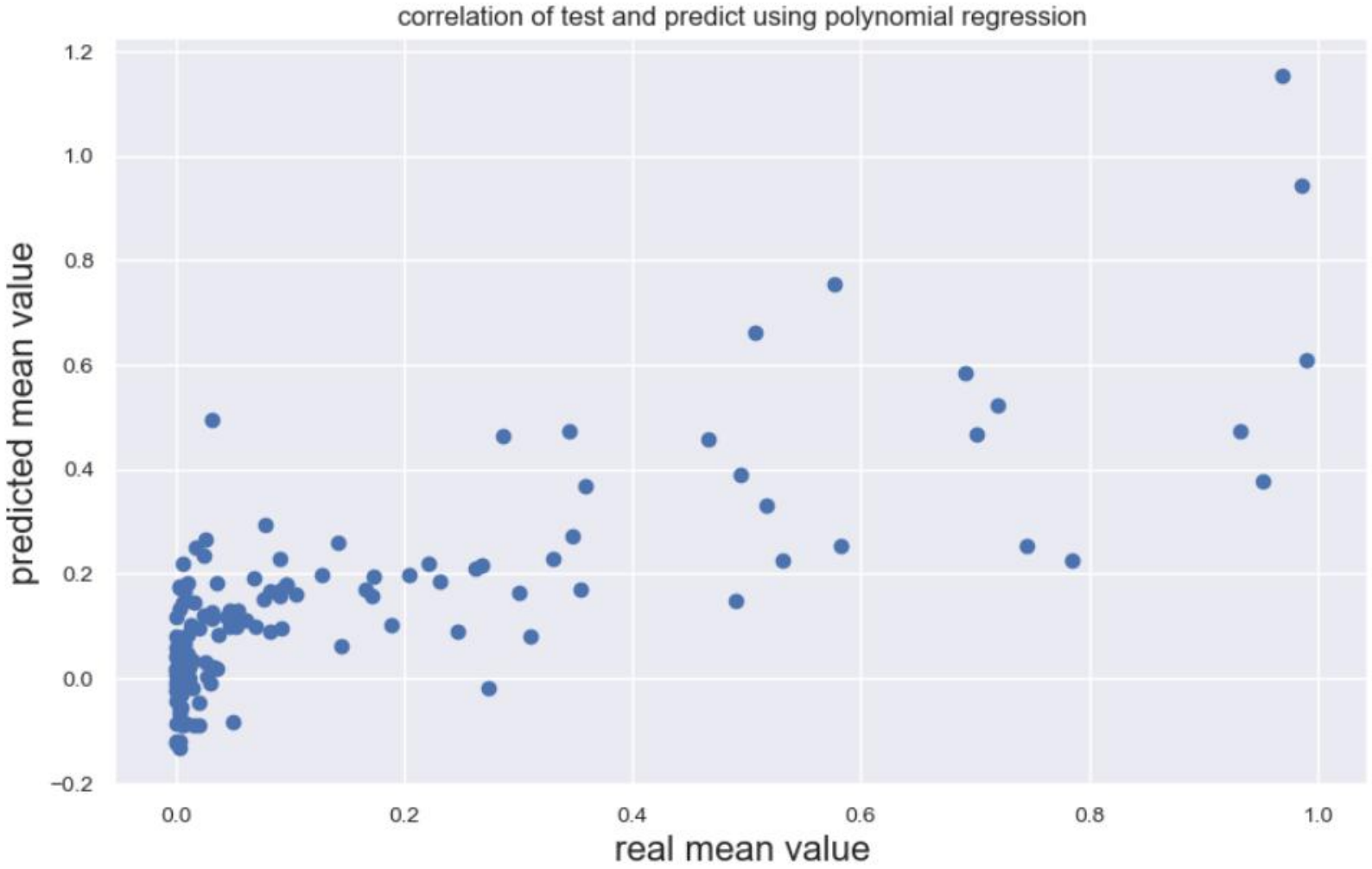
Regression Method

Train Test Split: 0.75:0.25

Preprocessing: Standardization

```
array([[0.2284751 , 0.20205281, 0.0033261 , 0.01042903, 0.85483881,
        0.41964814],
       [0.34993303, 0.15327645, 0.00245821, 0.01058475, 0.86760255,
        0.31812094],
       [0.25597299, 0.22377513, 0.00373495, 0.0090315 , 0.74055079,
        0.57956149],
       ...,
       [0.41153546, 0.15471258, 0.00239804, 0.00736432, 0.60337906,
        0.66526409],
       [0.42236218, 0.13564917, 0.00337581, 0.00827768, 0.67824584,
        0.58575777],
       [0.37824251, 0.1906104 , 0.00329101, 0.00981346, 0.80413761,
        0.41696024]])
```

Polynomial regression



[a,b]
degree 2 polynomial
[1, a, b, a^2, ab, b^2].

6 features ——> 28 features

Precision: 0.611

	player	award_share
0	Nikola Jokic	0.378906
1	Giannis Antetokounmpo	0.376953
2	Joel Embiid	0.298828
3	Chris Paul	0.0195312
4	Rudy Gobert	0.113281
5	James Harden	0.0839844
6	Luka Doncic	0.0410156
7	Ja Morant	0.0800781
8	Stephen Curry	0.191406
9	Devin Booker	-0.130859

XGBoost for regression

R2 score for test set: 0.628

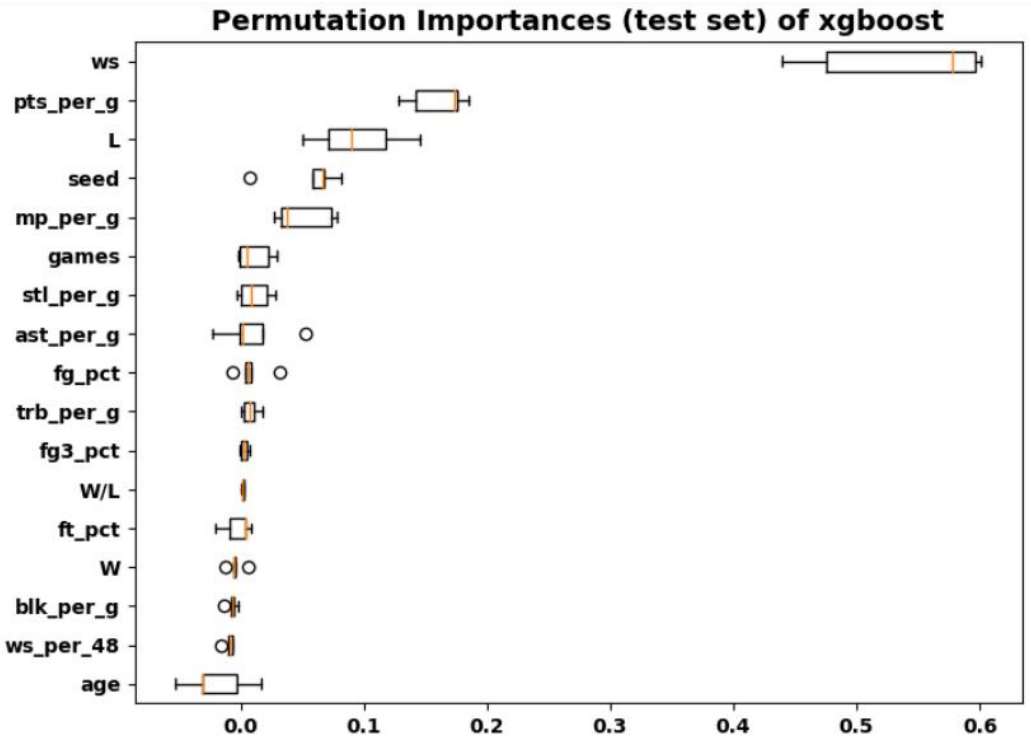
Input features

```
age          int64
games        int64
mp_per_g   float64
```

player	age	team	games	mp_per_g	pts_per_g	trb_per_g	ast_per_g	stl_per_g	blk_per_g	fg_pct	fg3_pct	ft_pct	ws	ws_per_48	W	L	W/L	seed
Nikola Jokic	27	DEN	57	33.1	25.8	13.8	8.0	1.4	0.8	0.571	0.363	0.811	11.7	0.299	38	26	0.594	10
Giannis etokounmpo	27	MIL	54	32.8	29.2	11.6	6.0	1.0	1.4	0.544	0.296	0.722	10.6	0.288	40	25	0.615	8
Joel Embiid	27	PHI	51	33.1	29.5	11.2	4.4	1.1	1.4	0.490	0.351	0.822	9.2	0.262	39	24	0.619	11
Chris Paul	36	PHO	58	33.0	14.9	4.5	10.7	1.9	0.3	0.487	0.330	0.843	8.5	0.212	51	13	0.797	1
Rudy Gobert	29	UTA	49	32.1	15.7	14.8	1.2	0.7	2.2	0.711	0.000	0.682	8.8	0.270	40	23	0.635	3
mes Harden	32	PHI	48	36.5	22.9	7.9	10.3	1.3	0.6	0.425	0.344	0.872	1.4	0.461	39	24	0.619	11
Luka Doncic	23	DAL	48	35.6	27.8	9.2	8.8	1.2	0.5	0.450	0.332	0.739	5.1	0.143	39	25	0.609	9
Ja Morant	22	MEM	52	33.3	27.6	5.8	6.6	1.2	0.4	0.494	0.343	0.761	6.4	0.177	44	22	0.667	6
ephen Curry	33	GSW	59	34.8	25.8	5.2	6.4	1.3	0.4	0.432	0.379	0.918	7.7	0.180	43	21	0.672	2
evin Booker	25	PHO	54	34.4	25.6	5.2	4.6	1.1	0.4	0.448	0.368	0.864	5.2	0.134	51	13	0.797	1

```
ws_per_48    float64
W              int64
L              int64
W/L          float64
seed           int64
```

	player	predicted MVP
0	Nikola Jokic	0.534516
1	Giannis Antetokounmpo	0.296173
8	Stephen Curry	0.128352
9	Devin Booker	0.121876



Final Prediction

Decision Tree Classifier: Chirs Paul
ANN: Nicola Jokic
Polynomial Regression: Nicola Jokic
XGBoost Regression: Nicola Jokic

Final prediction: Nicola Jokic

Future Improvements

1. Implement PCA after standardization to reduce the dimension of the input features and visualize data

2. Select most important features from the permutation importance figure of xgboost and apply them back to create another xgboost model for prediction.

Thanks for watching!