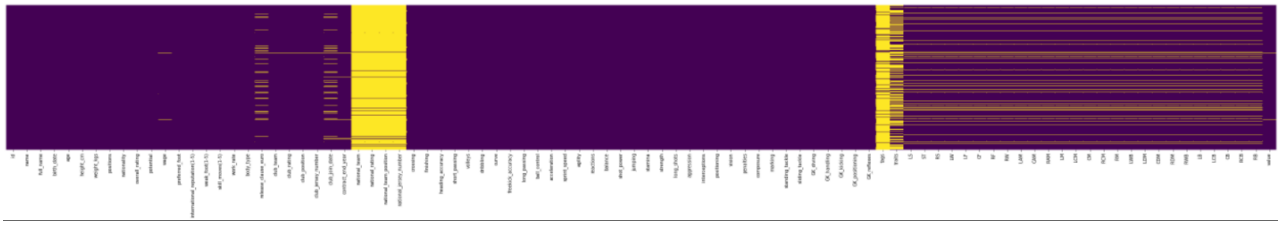
Preprocessing:

1- Handling nulls:

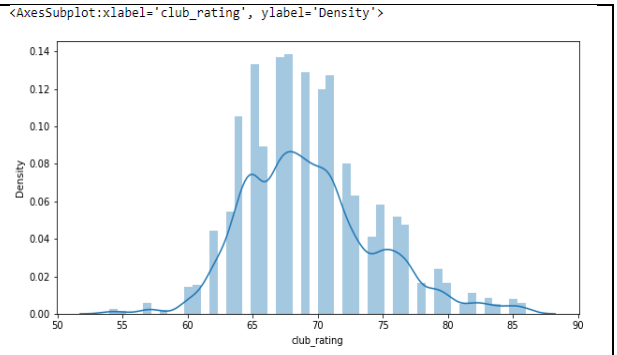
- Representing the nulls in columns using:
  sns.heatmap(player_data.isnull(),yticklabels=False,cbar=False,cmap="viridis",ax=ax)



- Filling nulls with Zeros:
  Columns → club_join_date, contract_end_year
  Reason → we considered, that if we found null in these columns means that player didn't join the club

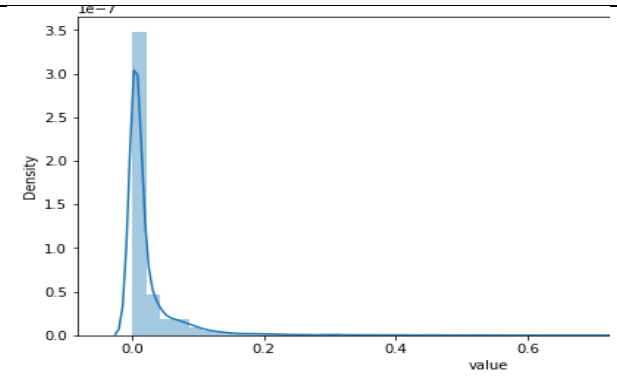
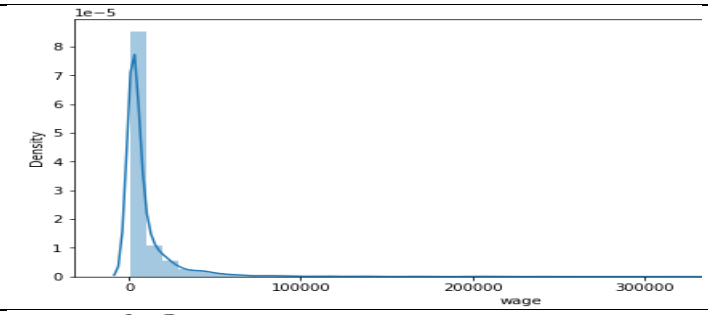
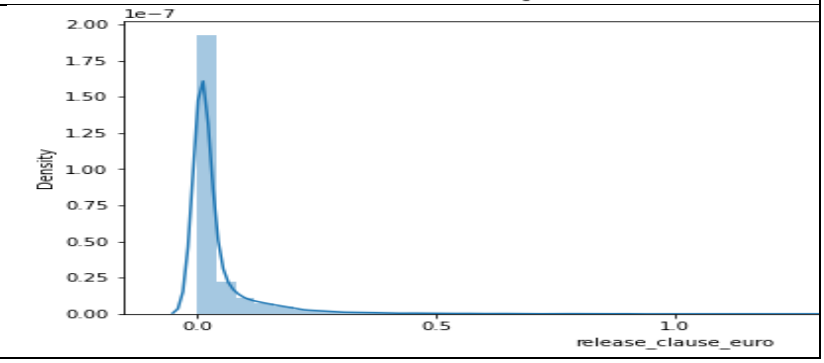
  Columns → national_rating
  Reason → Null values are greater than 90% in column, but non missing values may help us in prediction of value so we replaced it with zeros

- Replace nulls with mean:

Club_rating	

Reason → Data is normally distributed as shown in histogram plot, so mean suitable for it

- Replace nulls with mode:

value	
wage	
release_clause_euro	

Reason → Data is skewed as shown in histogram plot, so mean not suitable and we found the median equal null

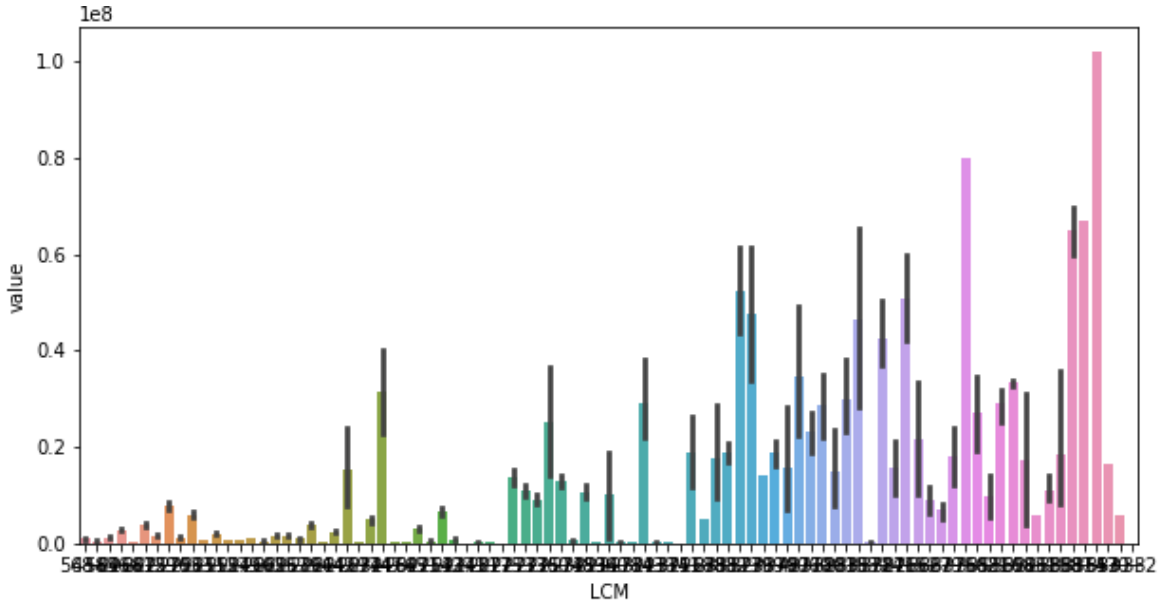
Columns → traits

Reason → replace with mode because it is categorical column

- Drop nulls:
  Columns → national_team, national_team_position, tags, national_jersey_number, club_jersey_number
  Reason → Null values are greater than 95% from each column

- Predict nulls:

Bar plot of (LCM with VALUE)



Columns → From column number 65 to 91 (zero based)
Reason →we found that the last 27 columns are dependent on **value** as shown in figure for example**,** so we predict nulls of each column with respect to **value column.**
Algorithm → predict using KNeighborsClassifier, with X equal "one of the columns, that we want to predict its value" and Y equal "value"

2- Handling categories:

body_type	○ First : we found data not true ,so we replaced it with mean ○ Then: apply Label encoding (as it has ordinal values)
work_rate	○ First: Ranking its categories (Ex:"High/ High": 9, "High/ Medium": 8 , "High/ Low": 7)
positions	○ First: Split string with(",") ○ Then: Generate 4 columns (1st position,2nd position,3rd position, 4th position). ○ Get correlation between 4 columns & y ,found it low so we drop column
preferred_foot	○ Apply One hot encoding ○ Reason → It has only 2 values (Right and Left) and they are not ordinal
traits	○ Try to tokenize it and find 37 unique value, so one hot encoding will not be applicable, and its values are not ordinal, so we drop it
From column number 65 to 91 (Zero based)	○ Apply Target Encoding ○ Reason → it has 89 unique category, so one hot encoding will not be applicable, and its values are not ordinal, so we choose target encoding to rank these categories with respect to target column

3- Feature selection:

- Joining 2 columns into one column:
  Columns → club_join_date, contract_end_year
  Reason → Subtracting these 2 columns (after converting contract_end_year from objects to integers → years only) to get one column called years_player_club which contains the total amount of years that the player spends in this club. Because the dependency between club_join_date & contract_end_year is very high.

- Dropping columns:
  Columns → GK_handling, GK_kicking, GK_positioning, GK_reflexes, agility, ball_control, curve, dribbling, freekick_accuracy, long_passing, long_shots, marking, penalties, positioning, reactions, release_clause_euro, short_passing, shot_power, sliding_tackle, sprint_speed, standing_tackle, volleys
  CAM, CB, CDM, CF, CM, RAM, RB, RCB,RCM, RDM,RF,RM,RS,RW, RWB , ST

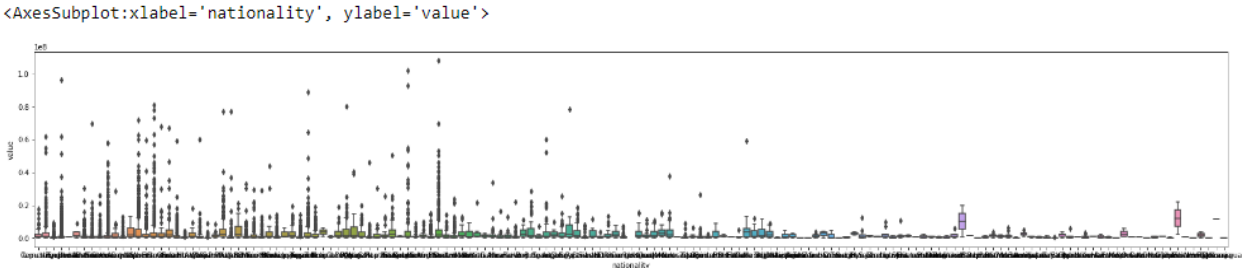
  Reason → High correlation with other features

  Columns → birth_date, club_position, club_team
  Reason → Has high correlation with other features. Using ANOVA algorithm

  Columns → Id, name, height_cm, full_name
  Reason → Low correlation with target column

Columns → nationality

Reason →it has a lot of outliers and low correlation with value as shown in the figure below.



Box plot of (nationality with value)

Regression techniques:

Regression technique	**Polynomial regression**	**Ridge regression**
differences	Linear regression with multi variable. In it we try to increase degree of model to fit data well	Linear regression with multi variable. This method performs L2 regularization, trying to avoid overfitting
Validation method	○ Used cross validation to impute degree of model, found that **degree 2** is the best because it has least cross validation score	○ Used GridSearchCv to tune alpha parameter(best alpha=1). ○ Used cross validation to impute degree of model, found that **degree 2** is the best because it has least cross validation score.
Cross validation score	1214106.4763454667	1089362.915462453
Test (RMSE)& accuracy	RMSE: 1325026.8357935972 Accuracy:0.95952518882685	RMSE:1215024.7146390676 Accuracy:0.9659665710951
Training (RMSE)&accuracy	RMSE: 514390.74800835684 Accuracy:0.9910649111659714	RMSE:721964.04594457 Accuracy:0.982398741646426
Time of training	7.771404981613159s	2.0188004970550537s

Train & test size:

Split data 80% training 20% test
X_train= (11490, 38)
X_test= (2873, 38)

Conclusion:

Applying preprocessing on Player value dataset. Through filling null values in the columns using mean, mode, or prediction nulls using other columns.

Also using one hot encoding, label encoding, and target encoding algorithms to handle categorical data. Then display correlation among all features and dropping features that have low correlation with value column or very high correlation with any feature.

After preprocessing we have applied **cross validation** and from it we knew that the best degree can be used in **polynomial regression** is **2nd degree** ,its cross validation = 1214106.4763454667 and test accuracy=0.95952518882685**.**

Then we tried also **ridge regression** , its cross validation =1089362.915462453 and test accuracy=0.9659665710951**.**