**Feature engineering af forklarende variable**

Anvendt flere populære teorier og koncepter inden for matematik og geometri i dine koder til data manipulation og transformation. Lad mig gå igennem dem:

**Skalering af koordinater**: Du bruger skalering til at konvertere koordinater fra procent til meter ved hjælp af skaleringsfaktorer for længde og bredde af spillebanen. Dette er en grundlæggende operation inden for data transformation, hvor du tilpasser dine data til en bestemt skala for at gøre dem mere sammenlignelige eller anvendelige.

**Beregning af afstand**: Du anvender Pythagoras' sætning til at beregne afstanden fra hver skudlokation til målet. Dette er en velkendt geometrisk formel, der bruges til at beregne afstanden mellem to punkter i et koordinatsystem.

**Beregning af vinkel**: Du bruger trigonometriske principper, herunder cosinusrelationerne, til at beregne vinklen for hvert skud i forhold til målstolpernes position. Dette giver dig indsigt i skudets retning i forhold til målet og er afgørende for at forstå skudmønstre og potentielle målchancer.

**Covering shadows**

Funktionen, du har lavet, hedder create\_covering\_shadows\_polygon, og dens formål er at oprette trekantformede dækningsskygger omkring bolden baseret på placeringen af spillerne og bolden. Her er en detaljeret forklaring på, hvordan funktionen fungerer:

Parametre:

ball\_location\_x, ball\_location\_y: X- og Y-koordinaterne for boldens position.

player\_locations: En dataframe, der indeholder information om spillerpositioner, herunder deres X- og Y-koordinater og hvilket hold de tilhører.

buffer\_distance\_player1, buffer\_distance\_player2, buffer\_distance\_player3: Afstande, der definerer størrelsen af dækningsskyggerne for hver spiller.

triangle\_width\_factors: En vektor, der angiver faktorer for bredden af dækningsskyggerne for hver spiller.

x\_shift\_factors, y\_shift\_factors: Vektorer, der angiver faktorer for forskydning i X- og Y-retning for dækningsskyggerne for hver spiller.

Funktionens arbejdsgang:

Funktionen løber igennem hver spiller i player\_locations og beregner dækningsskyggerne omkring bolden.

For hver spiller beregner den retningen og afstanden fra bolden til spilleren.

Derefter konstrueres trekanten, der repræsenterer dækningsskyggen for den pågældende spiller. Trekantens form og størrelse justeres baseret på de givne faktorer.

Trekantens koordinater tilføjes til en dataframe med holdinformation, og denne dataframe opdateres løbende for hver spiller.

Til sidst returneres en dataframe, der indeholder koordinaterne for alle dækningsskygger omkring bolden.

Resultat:

Funktionen genererer en dataframe med koordinaterne for trekantformede dækningsskygger omkring bolden, hvor hver trekant repræsenterer dækningsskyggen for en spiller.

**Hvad er en Klassificeringsmodel:**

En klassificeringsmodel er en type maskinlæringsmodel, der bruges til at forudsige en diskret målvariabel baseret på inputvariabler. Den diskrete målvariabel kan være kategorisk og repræsenterer normalt forskellige klasser eller grupper.

Formålet med en klassificeringsmodel er at lære mønstre i data og bruge disse mønstre til at forudsige hvilken klasse eller gruppe en given observation tilhører. Modellen trænes ved at bruge et datasæt, hvor både inputvariabler og den tilsvarende målvariabel er kendte. Derefter bruges modellen til at forudsige målvariablen for nye observationer, hvor kun inputvariablerne er kendte.

I din opgave om xG (Expected Goals) har du sandsynligvis oprettet en klassificeringsmodel til at forudsige sandsynligheden for, at et givent skud bliver til et mål. Her er nogle måder, hvordan din xG-model relaterer sig til forklaringen af en klassificeringsmodel:

Inputvariabler: I din xG-model er inputvariablerne sandsynligvis forskellige attributter ved et skud, såsom afstanden til målet, vinklen på skuddet, placeringen på banen osv. Disse inputvariabler bruges til at træne modellen til at forudsige sandsynligheden for, at skuddet resulterer i et mål.

Målvariabel: Din målvariabel i xG-modellen er sandsynligvis binær, hvilket betyder, at den har to klasser: et skud, der resulterer i et mål (positiv klasse) og et skud, der ikke resulterer i et mål (negativ klasse). Modellen forsøger at klassificere hvert skud som enten positivt eller negativt baseret på inputvariablerne.

Træning af modellen: Du har trænet din xG-model ved at bruge et datasæt med skud, hvor resultatet af hvert skud (mål eller ikke-mål) er kendt. Modellen lærer mønstrene i dataene og bruger dem til at foretage forudsigelser på nye skud.

Anvendelse af modellen: Når din xG-model er trænet, kan den bruges til at forudsige sandsynligheden for, at et nyt skud resulterer i et mål baseret på dets egenskaber. Dette kan være nyttigt for holdet til at evaluere skudmuligheder under en kamp og træffe informerede beslutninger om, hvornår og hvor man skal skyde.

Samlet set fungerer din xG-model som en klassificeringsmodel ved at klassificere hvert skud som enten en scoring mulighed eller ej baseret på de observerede træk ved hvert skud. Denne tilgang kan give værdifuld indsigt i, hvordan forskellige faktorer påvirker sandsynligheden for at score et mål i fodbold.