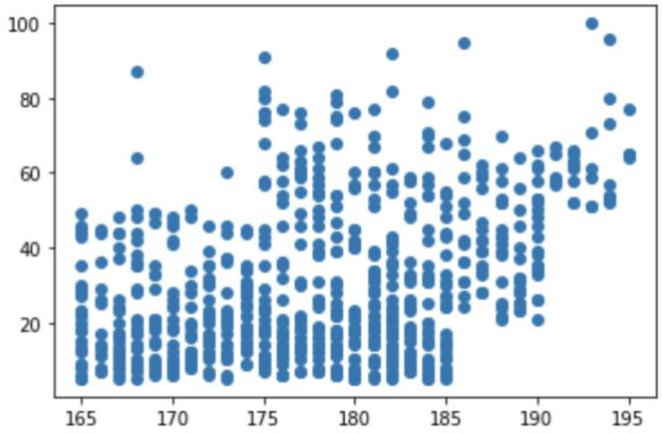
Part A

A1

i) Scatter plot

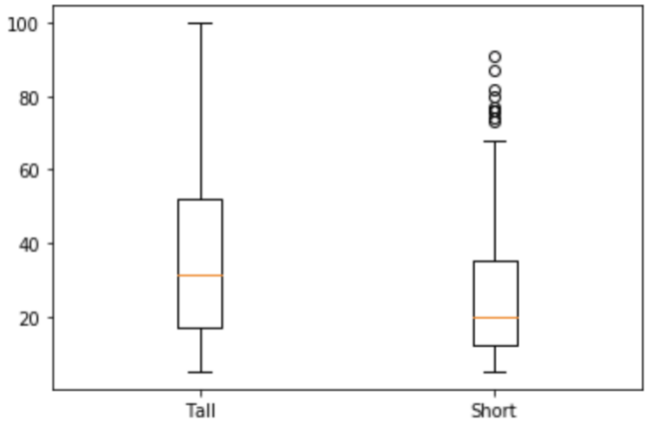
Since the height and heading skills are both continuous variables, first the scatter plot is drawn to visualize the relation between these two variables.



From the above plot, we can see there is a positive correlation between the height and the heading skills.

ii)Boxplot by group

Next, the players are divided into two groups based on the threshold (the median height). Boxplots are drawn to show the distribution of heading skills for each group.



As can be seen, the tall group has relatively higher heading scores than the short group.

Based on the two plots, we can say that there is a positive association between the height and heading skills of players.

A2

i)Correlation test

```
r:0.3644843475392581
p:2.0384816518298775e-23
```

The correlation coefficient is 0.36 with p value less than 0.01. So it shows a significant medium effect as well as a positive correlation between the height and the heading skills.

ii)t test
For the two groups (tall group and short group), first levene test is done to check whether the variances are equal for two groups.

```
# levene test
stat, p = levene(df_tall['Heading'],df_short['Heading'])
stat, p
# p<0.05, we reject H0, which means the variances of two samples are not equal
```

```
(19.31818408928394, 1.278678658692957e-05)
```

P<0.05, we reject H0 and assume the variances are not equal.

Next, the two sample t-test is performed.
Let sample 1 be the heading scores of tall group, and sample 2 be the heading scores of short group.
H0: $\mu_1 - \mu_2 \leq 0$; H1: $\mu_1 - \mu_2 > 0$

```
# t-test
# H0:u1<=u2  H1:u1>u2
# unequal variance
def t_test(x,y,alternative='both-sided'):
            t, double_p = ttest_ind(x,y,equal_var = False)
            if alternative == 'both-sided':
                pval = double_p
            elif alternative == 'greater':
                if np.mean(x) > np.mean(y):
                    pval = double_p/2.
                else:
                    pval = 1.0 - double_p/2.
            elif alternative == 'less':
                if np.mean(x) < np.mean(y):
                    pval = double_p/2.
                else:
                    pval = 1.0 - double_p/2.
            return (t,pval)

t_test(df_tall['Heading'], df_short['Heading'],alternative='greater')
# p<0.001, so we reject H0
```

```
(6.128842934670694, 7.498538109360002e-10)
```

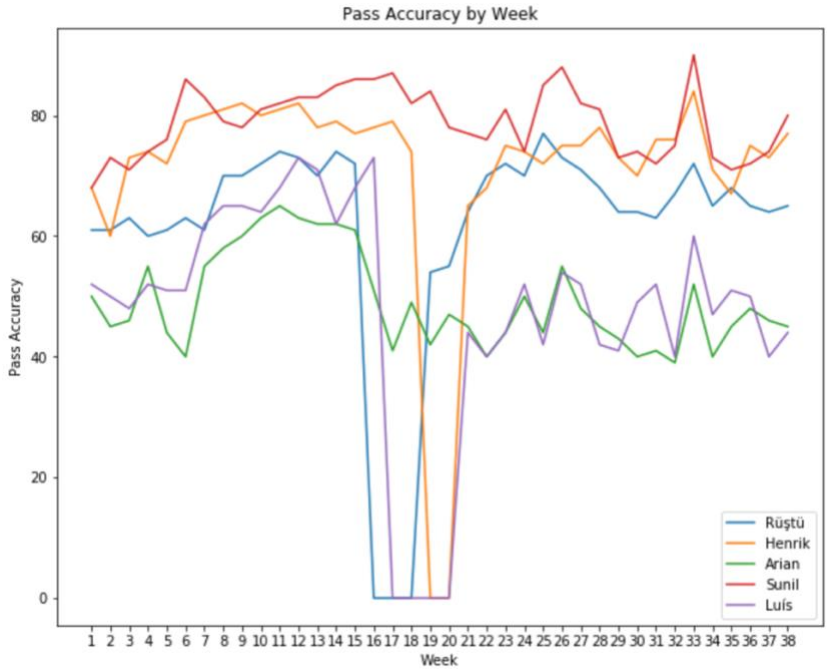
t=6.13, p<0.01. We reject H0, which means taller players have higher heading scores.

Based on the above two test, we can conclude that taller players have better heading skills.

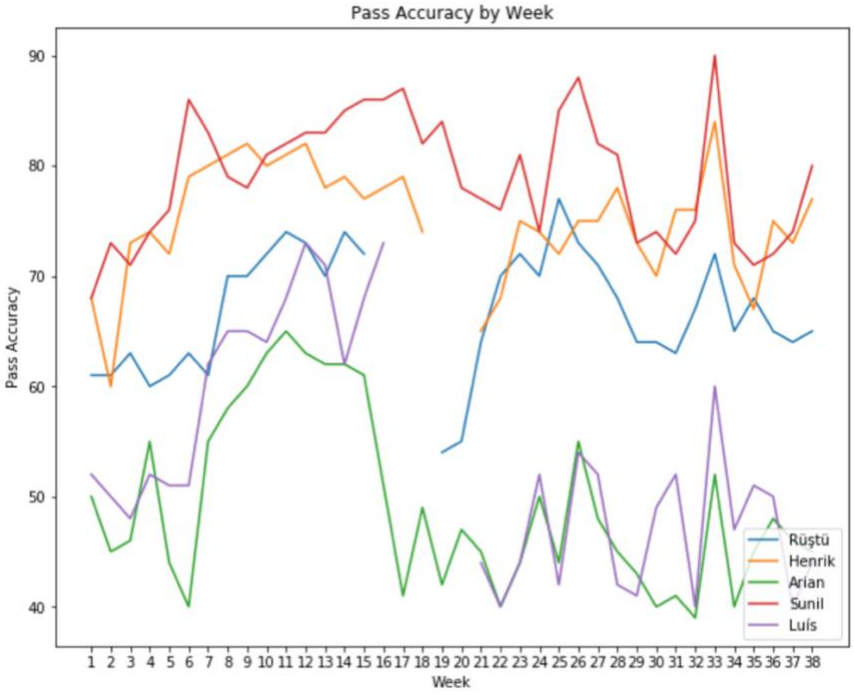
Part B

B1

After formatting the dataset (transpose, etc.), the pass accuracy at each week for the 5 players is plotted.



Since some players have injuries during the season, zero values exist for a certain period of time. To reduce the effect of zero values on time series visualization, zero values are transformed into null values to rescale the plot.



Now the plot is more readable. Based on this plot, it takes around 6-8 weeks for players to be synchronized with other team members. After 15-17 weeks, the pass accuracy starts to drop again.

B2

Generally, the pass accuracy is much lower when a player returns back from the injury at first. But for some players, after 2-3 weeks the performance is back to normal again, while for some the accuracy drops and maintains at a much lower level afterwards.

Based on the plot, Arian and Luís should be replaced. Arian's pass accuracy is not stable and drops a lot after week 15. Luís's pass accuracy drops a lot after returning back from his injury.

Part C

C1

Apply the k-means clustering to find the most similar players. Here the target is k=3.

The 'PlayerID' is dropped as it is a label and does not have any meaning in the clustering. 'Market Value' is dropped to focus on the performance of players and their natural characteristics.

Before implementing the k-means algorithm, features are normalized using min-max scaler (due to the fact that clustering is distance-based).

A small part of the output with labels attached to each row is shown below:

	0	1	2	3	4	5	6	label
0	0.652174	0.648352	0.618421	0.0	0.0	0.4	0.366667	1
1	1.000000	1.000000	1.000000	0.0	0.0	0.2	0.966667	1
2	0.815217	0.813187	0.776316	0.0	0.0	0.2	0.633333	1
3	0.619565	0.593407	0.526316	0.0	0.0	0.0	0.600000	1
4	0.260870	0.230769	0.092105	0.0	0.0	1.0	0.333333	1
5	0.641304	0.659341	0.618421	0.0	0.0	0.8	0.900000	1
6	0.695652	0.681319	0.644737	0.0	0.0	0.4	0.433333	1
7	0.467391	0.472527	0.421053	1.0	0.0	0.2	0.166667	2
8	0.619565	0.593407	0.486842	1.0	0.0	0.2	0.400000	2
9	0.565217	0.549451	0.526316	0.0	0.0	0.0	0.666667	1

And the number of players for each cluster is:

```
df_young_clustering['label'].value_counts()
```

```
1    154
2     96
0     84
Name: label, dtype: int64
```

Based on the above, the salary for each staff is:

```
The salary for the staff who monitors cluster 0 is $84000
The salary for the staff who monitors cluster 1 is $138600
The salary for the staff who monitors cluster 2 is $96000
```

Part D

D1

i)Predictive part

First, create a prediction model (multiple linear regression is used in this case) to predict the new market value of each player.

The correlation matrix of independent variables are shown below:

	Shooting	Heading	Pressing	Substitute player?	Division	Age	Height
Shooting	1.000000	0.995815	0.969455	-0.201486	-0.195344	0.004268	0.361099
Heading	0.995815	1.000000	0.975290	-0.198541	-0.196091	0.005846	0.364484
Pressing	0.969455	0.975290	1.000000	-0.207341	-0.166752	0.011217	0.355895
Substitute player?	-0.201486	-0.198541	-0.207341	1.000000	-0.102311	-0.052187	-0.132185
Division	-0.195344	-0.196091	-0.166752	-0.102311	1.000000	-0.045574	-0.071093
Age	0.004268	0.005846	0.011217	-0.052187	-0.045574	1.000000	-0.029251
Height	0.361099	0.364484	0.355895	-0.132185	-0.071093	-0.029251	1.000000

Shooting, Heading and Pressing skills are highly correlated, so only one of them will be kept in the regression model.

Next, run a model including all the features.

OLS Regression Results

Dep. Variable:	Market Value	R-squared:	0.487
Model:	OLS	Adj. R-squared:	0.482
Method:	Least Squares	F-statistic:	93.84
Date:	Sat, 17 Apr 2021	Prob (F-statistic):	5.78e-96
Time:	21:55:46	Log-Likelihood:	-10893.
No. Observations:	700	AIC:	2.180e+04
Df Residuals:	692	BIC:	2.184e+04
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.796e+06	1.39e+06	2.016	0.044	7.33e+04	5.52e+06
Shooting	150.4180	2.86e+04	0.005	0.996	-5.61e+04	5.64e+04
Heading	5.368e+04	3.21e+04	1.675	0.094	-9261.644	1.17e+05
Pressing	1225.2815	1.37e+04	0.089	0.929	-2.57e+04	2.81e+04
Substitute player?	5.168e+05	1.17e+05	4.433	0.000	2.88e+05	7.46e+05
Division	-1.457e+06	1.26e+05	-11.554	0.000	-1.71e+06	-1.21e+06
Age	-1.4e+04	1.33e+04	-1.052	0.293	-4.01e+04	1.21e+04
Height	-9487.3237	7593.325	-1.249	0.212	-2.44e+04	5421.395

Based on the p values as well as the multicollinearity issue discussed above, Shooting, Pressing, Age, Height are dropped. Then run another iteration with the remaining features.

OLS Regression Results

Dep. Variable:	Market Value	R-squared:	0.485
Model:	OLS	Adj. R-squared:	0.483
Method:	Least Squares	F-statistic:	218.5
Date:	Sat, 17 Apr 2021	Prob (F-statistic):	7.21e-100
Time:	21:55:46	Log-Likelihood:	-10894.
No. Observations:	700	AIC:	2.180e+04
Df Residuals:	696	BIC:	2.181e+04
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	7.987e+05	1.22e+05	6.530	0.000	5.59e+05	1.04e+06
Heading	5.37e+04	2735.942	19.626	0.000	4.83e+04	5.91e+04
Substitute player?	5.332e+05	1.16e+05	4.607	0.000	3.06e+05	7.6e+05
Division	-1.448e+06	1.25e+05	-11.572	0.000	-1.69e+06	-1.2e+06

Now all variables are significant. The prediction model can be written as:
New Market Value = 798700 + 53700*Heading + 533200*Substitute player – 1448000*Division

Following the above, the new market value of each player is predicted, the underrated players are extracted (current market value<new market value), and underrated ratio is calculated using the given formula.

	PlayerID	Market Value	Shooting	Heading	Pressing	Substitute player?	Division	Age	Height	New Market Value	Underrated Ratio	Market Category
0	145230	5390000.0	87	87	72	0	0	19	168	5470600.0	0.014954	1
1	136184	5068800.0	83	82	63	1	0	20	182	5735300.0	0.131491	1
2	131857	5040000.0	79	80	63	0	0	29	175	5094700.0	0.010853	1
3	143728	4672500.0	72	76	59	0	0	20	180	4879900.0	0.044387	2
4	124547	4290000.3	88	91	76	0	0	27	175	5685400.0	0.325268	2
...	...	...	...	...	...	...	...	...	...	...	...	...
355	182832	522875.0	8	7	1	1	0	24	166	1707800.0	2.266173	3
356	190123	534625.0	9	9	3	0	0	26	170	1282000.0	1.397942	3
357	164523	528750.0	17	19	3	1	0	28	175	2352200.0	3.448605	3
358	174545	511750.0	8	6	5	0	0	20	176	1120900.0	1.190327	3
359	189923	511750.0	4	6	6	0	0	26	176	1120900.0	1.190327	3

360 rows × 12 columns

Sort by 'Market Category' and 'Underrated Ratio', and extract 2, 8, 20 most underrated players from category 1, 2, 3. This dataset is exported into Excel and used directly to build optimization models. (Please see the detailed dataset in 'optimization.xlsx' under worksheet 'clean list'.)

ii)Prescriptive part
Formulate the optimization problem:
Let $X_{ij} = \begin{cases} 1, \ player\ i\ is\ chosen\ for\ the\ skill\ j \\ 0, \ player\ i\ is\ not\ chosen\ for\ the\ skill\ j \end{cases}$
(i=1,2,3,…,30;
j=1,2,3 – j=1:Shooting; j=2:Heading; j=3:Pressing)

Objective function: Max $\frac{1}{3}(\sum X_{i1}Shooting_i + \sum X_{i2}Heading_i + \sum X_{i3}Pressing_i)$

Constraints:
s.t. $\frac{1}{2}(\sum X_{i1}Age_i + \sum X_{i3}Age_i) \leq 22$ (age)
$\sum X_{i1}IsInDivision2_i + \sum X_{i2}IsInDivision2_i + \sum X_{i3}IsInDivision2_i \leq 1$ (division)
$\sum X_{i1}MarketValue_i + \sum X_{i2}MarketValue_i + \sum X_{i3}MarketValue_i \leq 8500000$ (budget)
$\sum X_{i1}IsInCater1_i + \sum X_{i2}IsInCater1_i + \sum X_{i3}IsInCater1_i \leq 1$ (market category)
$\sum X_{i1} = 1, \sum X_{i2} = 1, \sum X_{i3} = 1$ (player)
$X_{i1} + X_{i2} + X_{i3} \leq 1, \forall i$ (assignment)
$X_{ij} \in \{0,1\}$ (binary)

Using Excel Solver, the optimal solution is:
$X_{2,1} = X_{17,2} = X_{12,3} = 1$

Player Number	PlayerID	Selected Skill
2	145230	Shooting

12	133574	Pressing
17	134625	Heading

The optimal objective value (average skill of the selected players) is 79.67.

D2

In this case, synergy among players is considered. The decision variables, constraints are the same as D1. Only the objective function is modified to include the effect of synergy.

Let S_k equal the additional skill scores due to synergy k (k=1,2,3,4,5), then the objective function is:

$$\text{Max } \frac{1}{3}\left(\sum X_{i1}Shooting_i + \sum X_{i2}Heading_i + \sum X_{i3}Pressing_i + S_1 + S_2 + S_3 + S_4 + S_5\right)$$

Based on the description, the formulas for S_k (k=1,2,3,4,5) are as follows:

$$S_1 = 0.5Pressing_1(X_{13}X_{71} + X_{13}X_{72}) + 0.5Pressing_7(X_{73}X_{11} + X_{73}X_{12}) + 0.2Shooting_1(X_{11}X_{72} + X_{11}X_{73}) + 0.2Shooting_7(X_{71}X_{12} + X_{71}X_{13})$$

$$S_2 = 0.1Pressing_2(X_{23}X_{141} + X_{23}X_{142}) + 0.1Pressing_{14}(X_{143}X_{21} + X_{143}X_{22})$$

$$S_3 = 0.3Shooting_3(X_{31}X_{82} + X_{31}X_{83}) + 0.3Shooting_8(X_{81}X_{32} + X_{81}X_{33}) + 0.3Heading_3(X_{32}X_{81} + X_{32}X_{83}) + 0.3Heading_8(X_{82}X_{31} + X_{82}X_{33}) + 0.3Pressing_3(X_{33}X_{81} + X_{33}X_{82}) + 0.3Pressing_8(X_{83}X_{31} + X_{83}X_{32})$$

$$S_4 = 0.15Shooting_1(X_{11}X_{92} + X_{11}X_{93}) + 0.15Shooting_9(X_{91}X_{12} + X_{91}X_{13})$$

$$S_5 = 0.1Shooting_4(X_{41}X_{52}X_{173} + X_{41}X_{172}X_{53}) + 0.1Shooting_5(X_{51}X_{42}X_{173} + X_{51}X_{172}X_{43}) + 0.1Shooting_{17}(X_{171}X_{42}X_{53} + X_{171}X_{52}X_{43}) + 0.1Heading_4(X_{42}X_{51}X_{173} + X_{42}X_{171}X_{53}) + 0.1Heading_5(X_{52}X_{41}X_{173} + X_{52}X_{171}X_{43}) + 0.1Heading_{17}(X_{172}X_{41}X_{53} + X_{172}X_{51}X_{43}) + 0.1Pressing_4(X_{43}X_{51}X_{172} + X_{43}X_{171}X_{52}) + 0.1Pressing_5(X_{53}X_{41}X_{172} + X_{53}X_{171}X_{42}) + 0.1Pressing_{17}(X_{173}X_{41}X_{52} + X_{173}X_{51}X_{42})$$

Using Excel Solver (use 'GRG Nonlinear' rather than 'Simplex LP' this time as the objective function is no longer linear, and using 'Simplex LP' will give us error), the optimal solution is still: $X_{2,1} = X_{17,2} = X_{12,3} = 1$

Player Number	PlayerID	Selected Skill
2	145230	Shooting
12	133574	Pressing
17	134625	Heading

The optimal objective value (average skill of the selected players) is 79.67.

(Possible explanations: Maybe the skill increase due to synergy is not large enough to affect the optimal solution in this case, thus the optimal solution does not change.)