Determinants of the player's overall score in FIFA22

Introduction

The aim of the project is to run a simple linear regression using the Ordinary Least Squares Method in Python. The following steps have been done:

1. Importing the database from the Kaggle website and selecting the necessary data together with cleaning and processing data to make it appropriate for regression using NumPy and Pandas packages.
2. Descriptive statistics and preliminary data analysis using the packages seaborn and myplotlib together with its extensions.
3. Run regression, analyse results and diagnose the model using the statsmodels package and its extensions.

Part 1. Database and variables

I have used data from the Kaggle website by user BRYANB, regarding the base card stats of footballers in FIFA22. The purpose of the regression is to examine the correlation of individual statistics with the overall score of a footballer's card.

After importing the data, I started by selecting the data I planned to use in the regression. Later, I decided to modify the names of some columns to make further work easier. During the modification, I noticed that some of the data is stored as type (int64), so I changed it to type (object), because I wanted the regression to be a discrete variable.

```
fifa = pd.read_csv('FIFA22.csv')
fifa
```

```
fifa.rename(columns = {'Weak Foot':'WeakFoot',
                       'Skill Moves':'SkillMoves',
                       'Best Position':'BestPosition'},
            inplace = True)
```

```
convert_dict = {'WeakFoot': object,
                'SkillMoves': object}
```

```
fifa=fifa.astype(convert_dict)
```

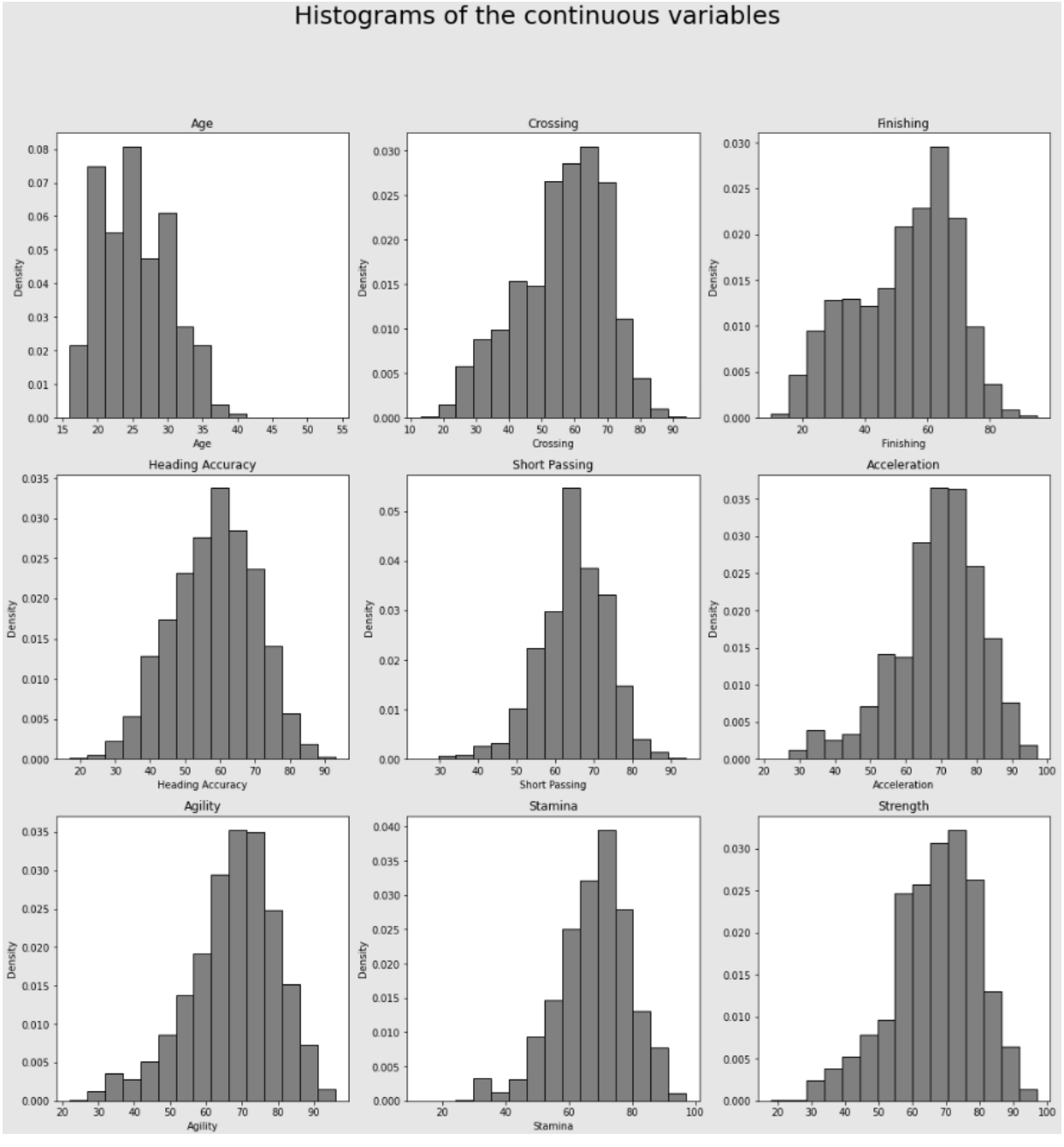
```
fifa1=fifa[['Overall','Age',
       'WeakFoot','SkillMoves','Crossing','Finishing',
      'HeadingAccuracy','Volleys','Dribbling','ShortPassing','LongPassing',
      'Acceleration','Agility','ShotPower','Stamina','Strength','Penalties',
      'BestPosition']]
fifa1
```

The next step was to remove all cards with a goalkeeper position from the database. I did not want to include them in the study as most of the goalkeeper stats are unique and would thus spoil the results.

```
fifa2=fifa1.loc[fifa1['BestPosition']!='GK']
fifa2
```

Part 2. Preliminary analysis and descriptive statistics

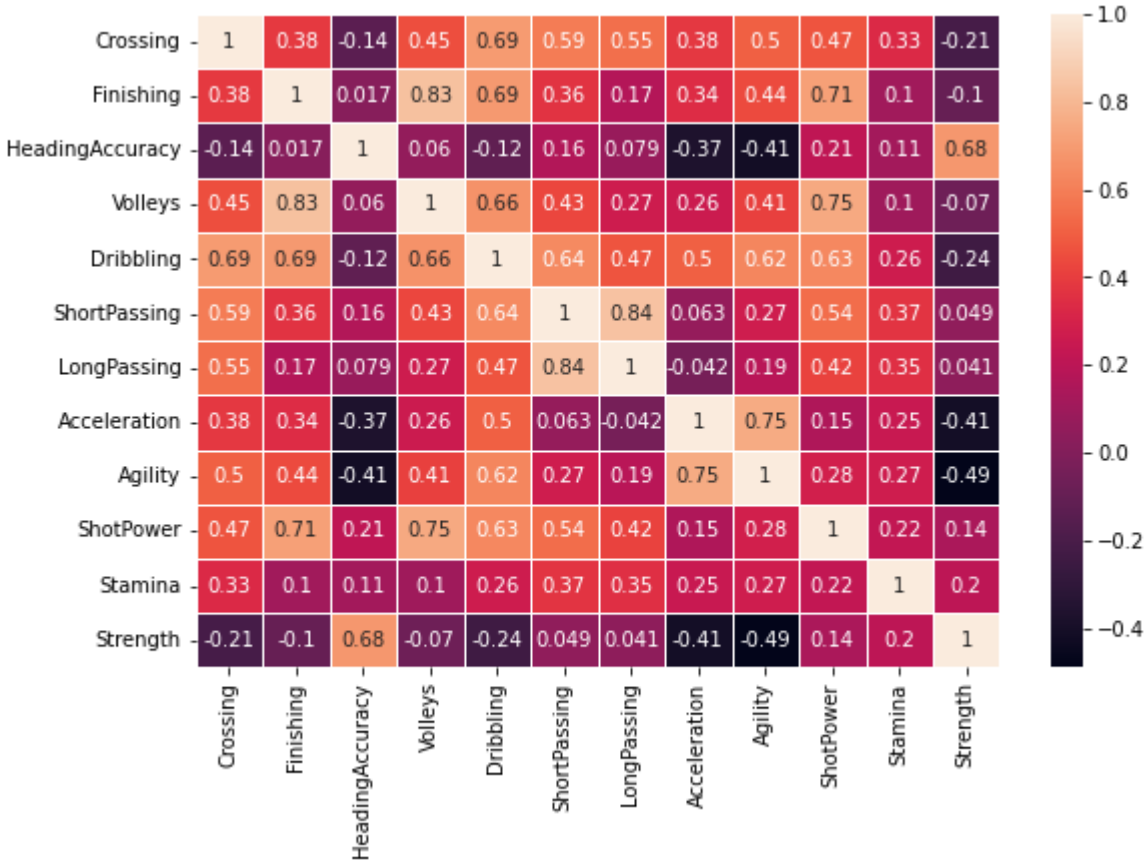
The first thing was to make histograms for each continuous explanatory variable.



From them, we can notice that none of the distributions is normal. This is crucial information in selecting the method for building the correlation matrix. Knowing that the distributions are not normal, the Spearman's method was used.

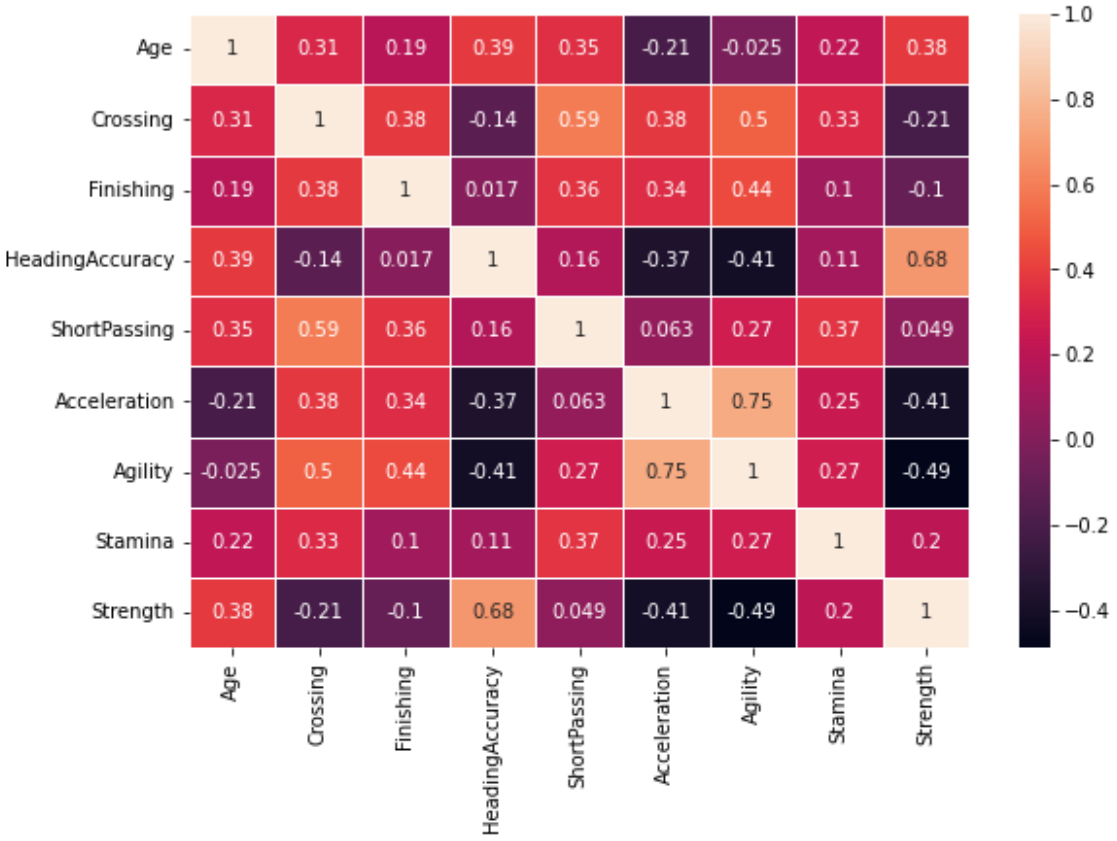
```
corrM1 = fifa2.iloc[:,3:16].corr(method='spearman')
corr
```

```
f, ax = plt.subplots(figsize=(9, 6))
sns.heatmap(corrM1, annot=True, linewidths=0.6)
```



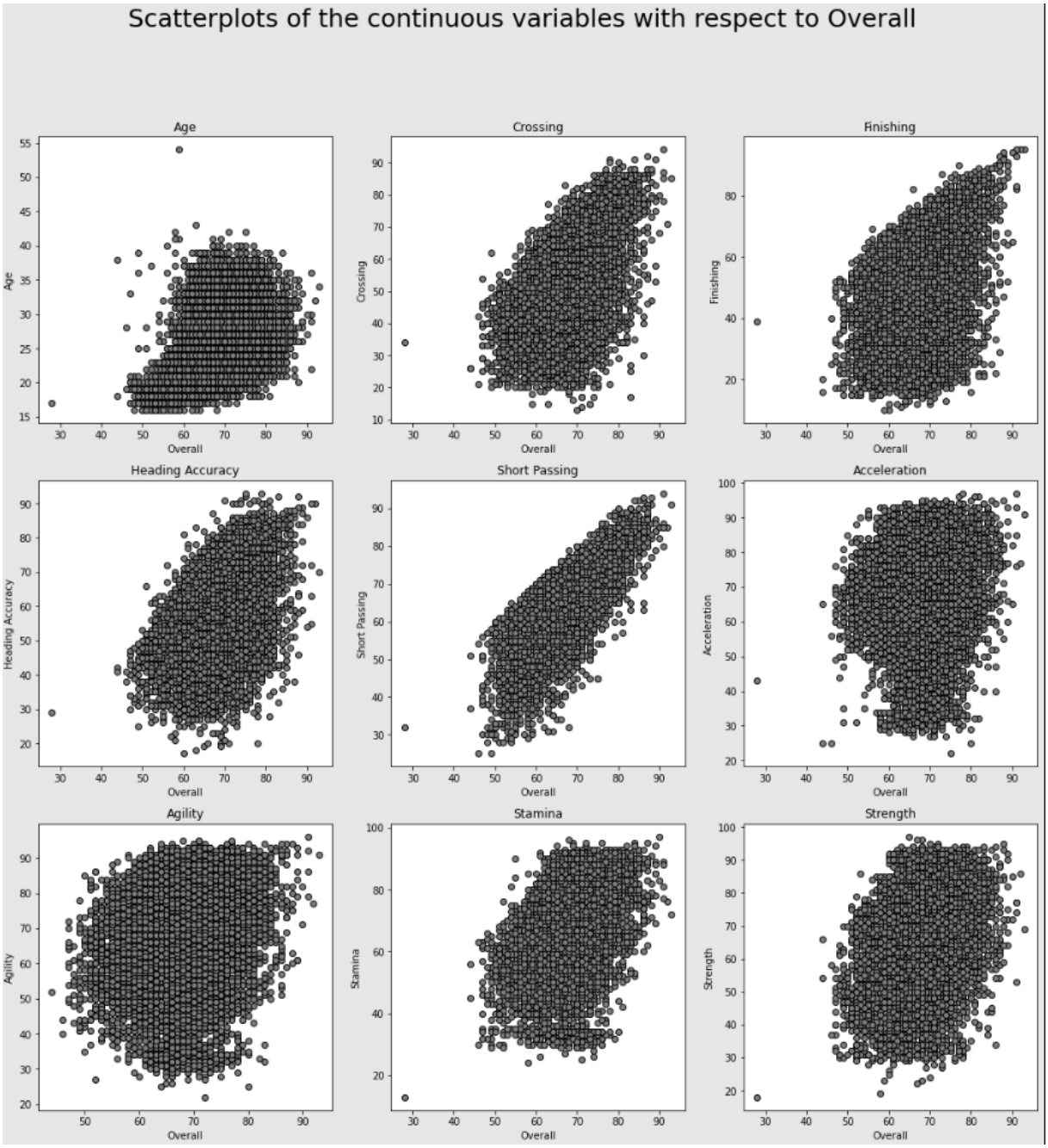
Initially, 17 explanatory variables were selected. I noticed a collinearity problem in the correlation matrix. I decided to remove some variables to reduce this problem.

```
fifa3=fifa[['Overall','WeakFoot','SkillMoves',
            'Age','Crossing','Finishing',
            'HeadingAccuracy','ShortPassing',
            'Acceleration','Agility','Stamina',
            'Strength','BestPosition']]
```



After removing the variables, the collinearity problem was significantly reduced. Some variables such as Strength and Heading Accuracy remained strongly correlated with each other. This is due to the very strong correlation between the two characteristics in general. In the end, I decided to keep both variables.

The next step was to perform scatterplots for each pair of the independent variable overall and all continuous explanatory variables. This is an important move as we can pre-estimate the correlation between the variables and it may give us suggestions that the correlation between the variables is not linear but, for example, quadratic.



The accompanying graphs mostly show a positive, linear relationship between the variables. Only in the case of the variable Age, a non-linear relationship may be questionable, but in this case, possible changes in the functional form will be diagnosed by a reset test.

Part 3. Regression results and diagnostics

After an initial analysis of the data, I was ready to run a regression.

```
reg1 = smf.ols('Overall ~ Age + Crossing + Finishing + HeadingAccuracy + ShortPassing + Acceleration + Agility + Stamina +
Strength + WeakFoot + SkillMoves', data=fifa4).fit()
```

OLS Regression Results

Dep. Variable:	Overall	R-squared:	0.789
Model:	OLS	Adj. R-squared:	0.789
Method:	Least Squares	F-statistic:	3533.
Date:	Mon, 06 Jun 2022	Prob (F-statistic):	0.00
Time:	22:59:14	Log-Likelihood:	-37657.
No. Observations:	15129	AIC:	7.535e+04
Df Residuals:	15112	BIC:	7.548e+04
Df Model:	16		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	13.2893	0.546	24.326	0.000	12.219	14.360
WeakFoot[T.2.0]	0.2712	0.575	0.471	0.637	-0.857	1.399
WeakFoot[T.3.0]	-0.0571	0.573	-0.100	0.921	-1.181	1.067
WeakFoot[T.4.0]	0.1646	0.575	0.286	0.775	-0.963	1.293
WeakFoot[T.5.0]	0.4229	0.597	0.708	0.479	-0.748	1.593
SkillMoves[T.2.0]	0.9334	0.150	6.241	0.000	0.640	1.227
SkillMoves[T.3.0]	1.4765	0.155	9.536	0.000	1.173	1.780
SkillMoves[T.4.0]	4.0572	0.172	23.633	0.000	3.721	4.394
SkillMoves[T.5.0]	6.8223	0.328	20.780	0.000	6.179	7.466
Age	0.1614	0.006	25.591	0.000	0.149	0.174
Crossing	0.0388	0.003	14.488	0.000	0.034	0.044
Finishing	-0.0049	0.002	-2.464	0.014	-0.009	-0.001
HeadingAccuracy	0.1744	0.003	60.161	0.000	0.169	0.180
ShortPassing	0.3820	0.004	97.914	0.000	0.374	0.390
Acceleration	0.0887	0.003	26.679	0.000	0.082	0.095
Agility	-0.0108	0.004	-2.961	0.003	-0.018	-0.004
Stamina	0.0470	0.002	18.847	0.000	0.042	0.052
Strength	0.0518	0.003	18.014	0.000	0.046	0.057

Omnibus:	270.574	Durbin-Watson:	1.961
Prob(Omnibus):	0.000	Jarque-Bera (JB):	322.189
Skew:	0.270	Prob(JB):	1.09e-70
Kurtosis:	3.468	Cond. No.	8.77e+16

Starting with diagnostics, the F-statistic equal to 3353 gives no grounds to reject the null hypothesis that the variables are chosen incorrectly.

The Durbin-Watson test examines the presence of an autocorrelation problem. The statistic of 1.961 is within the range of [1.5, 2.5] so we can conclude that there is no autocorrelation problem in the model.

The Jarque-Bera test checks whether the data is normally distributed. The statistic equal to 322.189 tells us that there are no grounds to reject the null hypothesis that the data is normally distributed, thus we conclude that we do not have a normally distributed data. However, in the case of huge sample, we should not be scared of that problem, as the Central Limit Theorem holds.

```
resettest = smd.linear_reset(res=reg1, power=2, test_type="fitted", use_f=True)
print(resettest)
```

```
<F test: F=243.14075476170152, p=2.155852300570797e-54, df_denom=1.51e+04, df_num=1>
```

Also, the Ramsey RESET test was performed to check the if the functional form of the model is correct. The F statistic equal to 243.14 and p-value < 0.001 informs us that there are grounds to reject the null hypothesis that our functional form is linear and correct.

Moving to the analysis of the regression results. From the regression table, we can see, that only the variable WeakFoot is statistically insignificant for alpha = 5%. All ther variables are statistically significant for alpha = 5%, where there positive correlation can be observed between Overall and SkillMoves (all the levels), Age, Crssing, HeadAccuracy, ShortPassing, Acceleration, Stamina, Strenght. The negative relationship can be observed between the Overall and Finishing and Agility.

Sources

- https://www.kaggle.com/datasets/bryanb/fifa-player-stats-database?select=FIFA22_official_data.csv
- Kurs: 2400-ZEWW796 / Data processing and analysis in Python language / konw./ mgr Damian Zięba / Piątek 13:15 - 14:45 co tydzien / A102 (uw.edu.pl)