Федеральное государственное автономное образовательное учреждение

высшего образования

Национальный исследовательский университет

"Высшая школа экономики"

Дисциплина «Основные методы анализа данных»

Отчет о проделанной работе на тему

“Анализ данных футбольного симмулятора FIFA19”

Выполнили студенты группы 194

Горбачев Ринат

Конюх Александр

Преподаватель:

Миркин Борис Григорьевич

Москва, 2021

Оглавление

[Введение 3](#_Toc90162030)

[Линейная регрессия и коэффициент корреляции 5](#_Toc90162031)

[Метод главных компонент для визуализации данных 8](#_Toc90162032)

[Интерпретация кластеров в категориальных признаках и хи-квадрат 9](#_Toc90162033)

[Таблица сопряженности 10](#_Toc90162034)

[Сравнение средних с помощью бутстрэпа 13](#_Toc90162035)

[Заключение 14](#_Toc90162036)

[Приложение 1 15](#_Toc90162037)

[Приложение 3 17](#_Toc90162038)

[Приложение 4 18](#_Toc90162039)

[Приложение 5 19](#_Toc90162040)

# Введение

Нами был выбрал датасет полученный на основе футбольного симмулятора - fifa19. Этот набор данных содержит большое количество различных признаков и отлично подходит для реализации алгоритмов и метода анализа данных рассмотренных на курсе.

Таблица данных была загруженна с соревнования по анализу данных на сайте kaggle.com.

Все вычисления и воспроизводимые результаты были получены с помощью языка python3.

Так как изначальный датасет содержит большое количество объектов и признаков, то вместе с руководителем проекта было решено сократить их количество.

После предобработки данных предствляем основные анализируемые элементы.

Общаяя структура данных:

* **Количество анализируемых объектов: 1000**
* **Количество признаков: 11**

Объекты в нашем случае – футболисты.

Описание основных признаков:

* **Age**(количественный) - возвраст футболиста. Принимает значения [18;40]
* **Overall**(количественный) - общая оценка эффективности футболиста. Принимает значения [77;100]
* **Position2(**номинальный) - позиция футболиста на поле. Принимает одно из следующих значений:

**AT** - нападающий

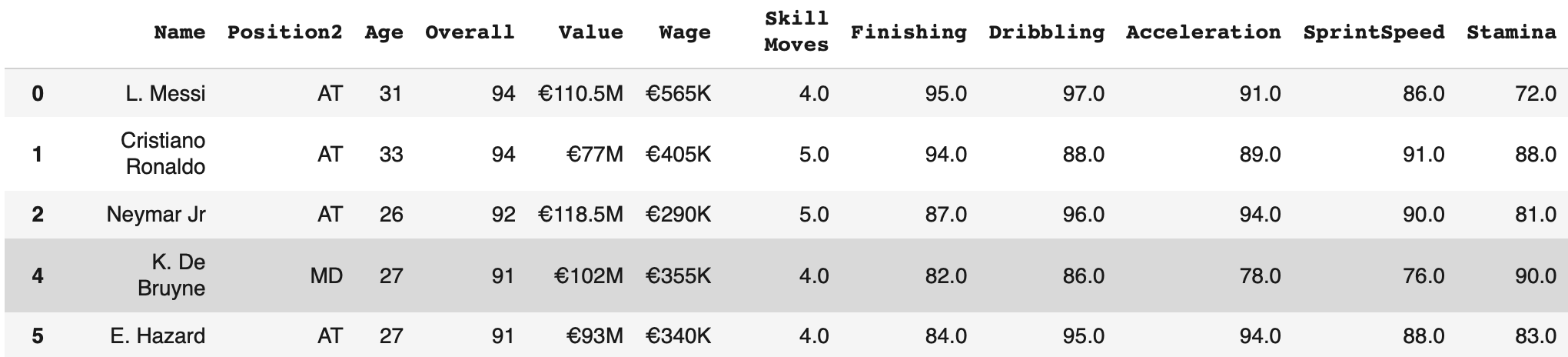
**MD -** полузащитник

**DF -** защитник

* **Value\_f**(количественный) - стоимость футболиста. Принимает значения [0;150M]
* **Wage\_f**(количественный)- зарплата футболиста. Принимает значения [0;565K]
* **Skill Moves**(количественный) - оценка техничности футболиста. Принимает значения [1;5]
* **Acceleration**(количественный) - на сколько быстро футболист набирает максимальную скорость. Принимает значения [40;100]
* **Sprint Speed**(количественный) - максимальная скорость футболиста. Принимает значения [40;100]
* **Stamina**(количественный)– выносливость футболиста.

Принимает значения [40;100]

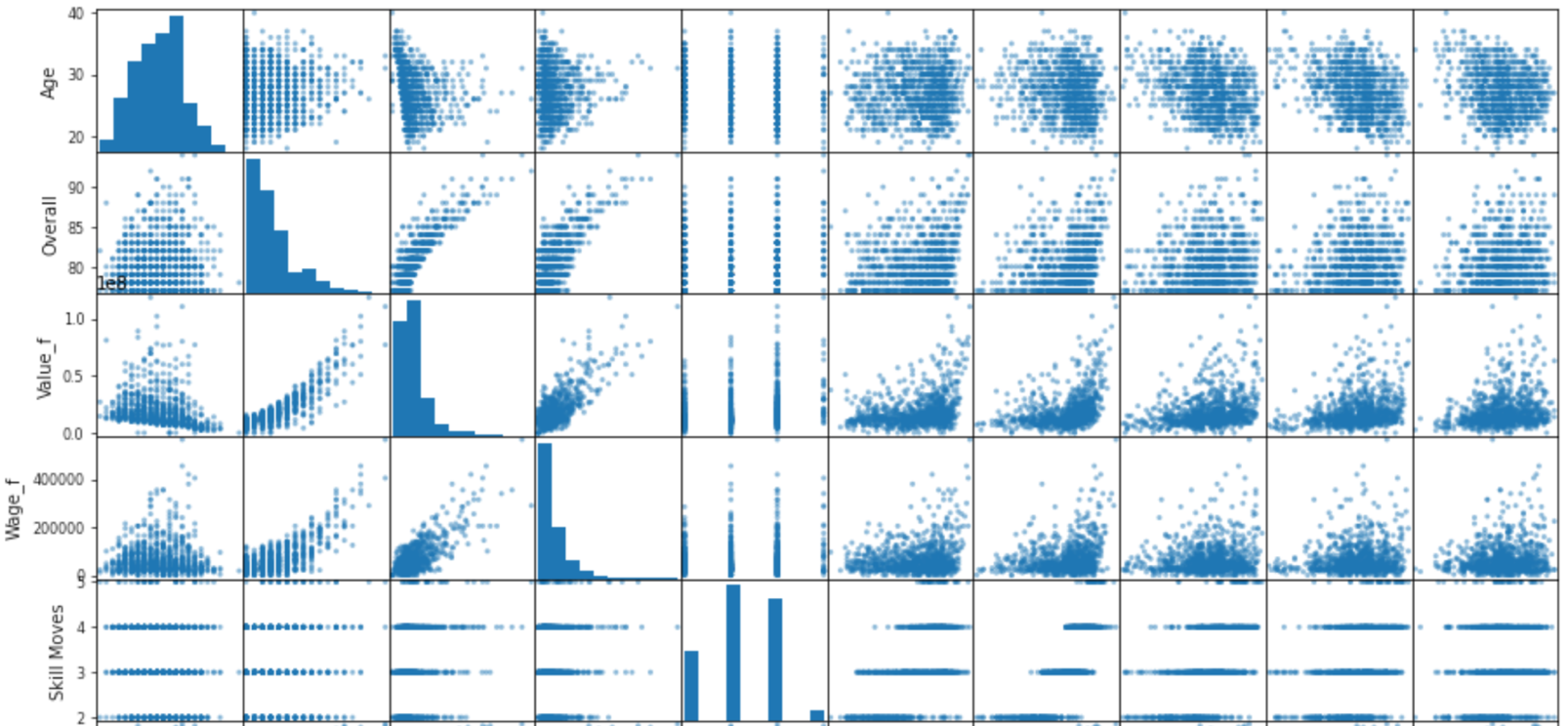
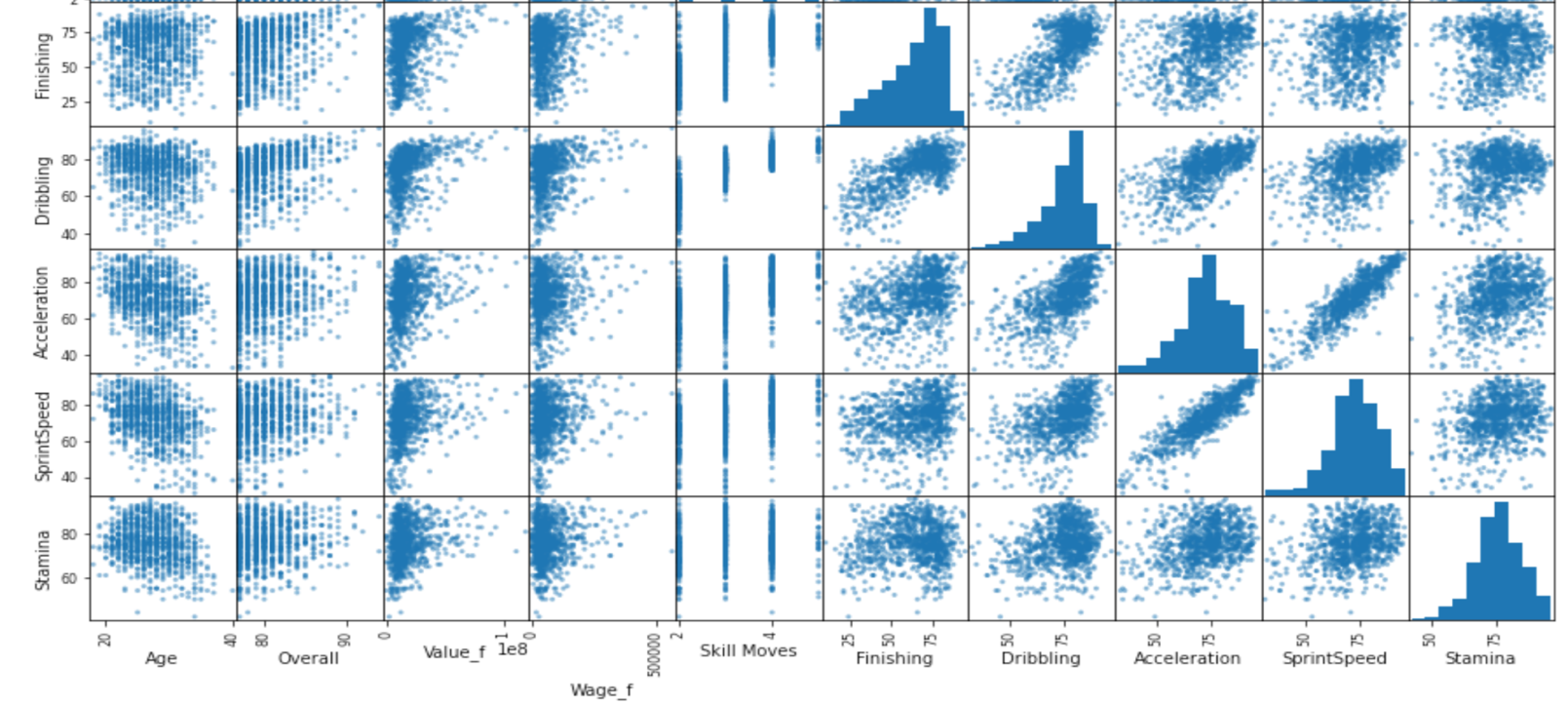
**Адрес данных**: https://www.kaggle.com/karangadiya/fifa19

**Фрагмент таблицы данных**:

Весь код будет добавлен в конце отчета в приложениях.

# Линейная регрессия и коэффициент корреляции

Для начала нужно выбрать два признака, которые обладали бы “линейным” полем рассеивания, то есть как-то зависили друг от друга.

С помощью следующей команды построим графики:

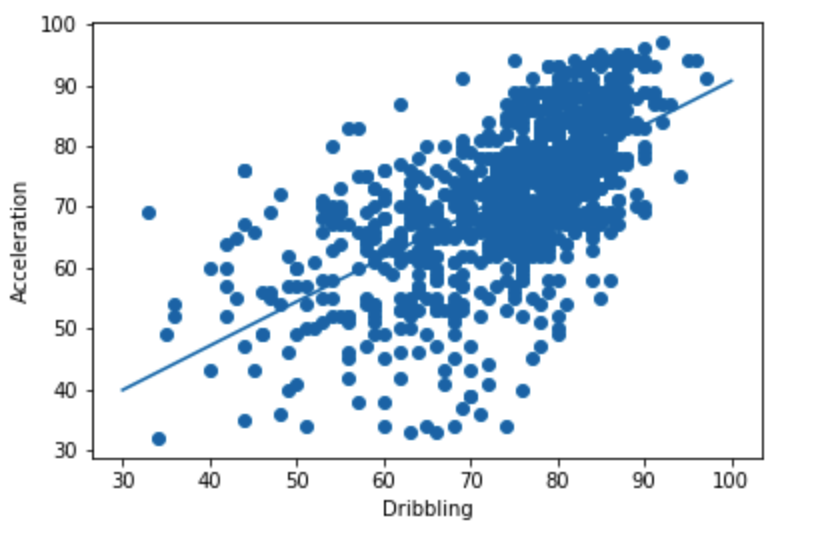
Как видно здесь много линейно зависимых признаков, мы решили выбрать Dribbling и Acceleration. В качестве целевой переменной выберем Dribbling.

Рассчитаем для них коэффициенты регрессии и свободный коэффициент:

Regression coefficient**:** **0.7261247967**

Intercept**:** **18.0782857520**

Вслед за этим обучим линейную модель взятую из библиотеки sklearn и изобразим на графике:



Посчитаем коэффициент корреляции и детерминации:

Correlation coefficient**:** **0.6284067462**

Determination coefficient**:** **0.3948950387**

Из теории мы знаем, что коэффициент детерминации принадлежит отрезку(−∞,1] и равен 1 в случае, когда модель выдает идеальные ответы. В нашем случае он больше 0, но все же не слишком близко к 1, следовательно предсказания нашей модели лучше, чем просто константное предсказание.

Так же посчитаем среднее квадратичное отклонение и коэффициент детерминации использую различные метрики и посмотрим как улучшется качество предсказания.

LinearRegression

MSE **2.146343967263835**

R**^2** Determenation **0.7163350062879218**

RandomForestRegressor

MSE **2.047768689744677**

R**^2** Determenation **0.7344369543475122**

GradientBoostingRegressor

MSE **1.6900830767872876**

R**^2** Determenation **0.7780014177566766**

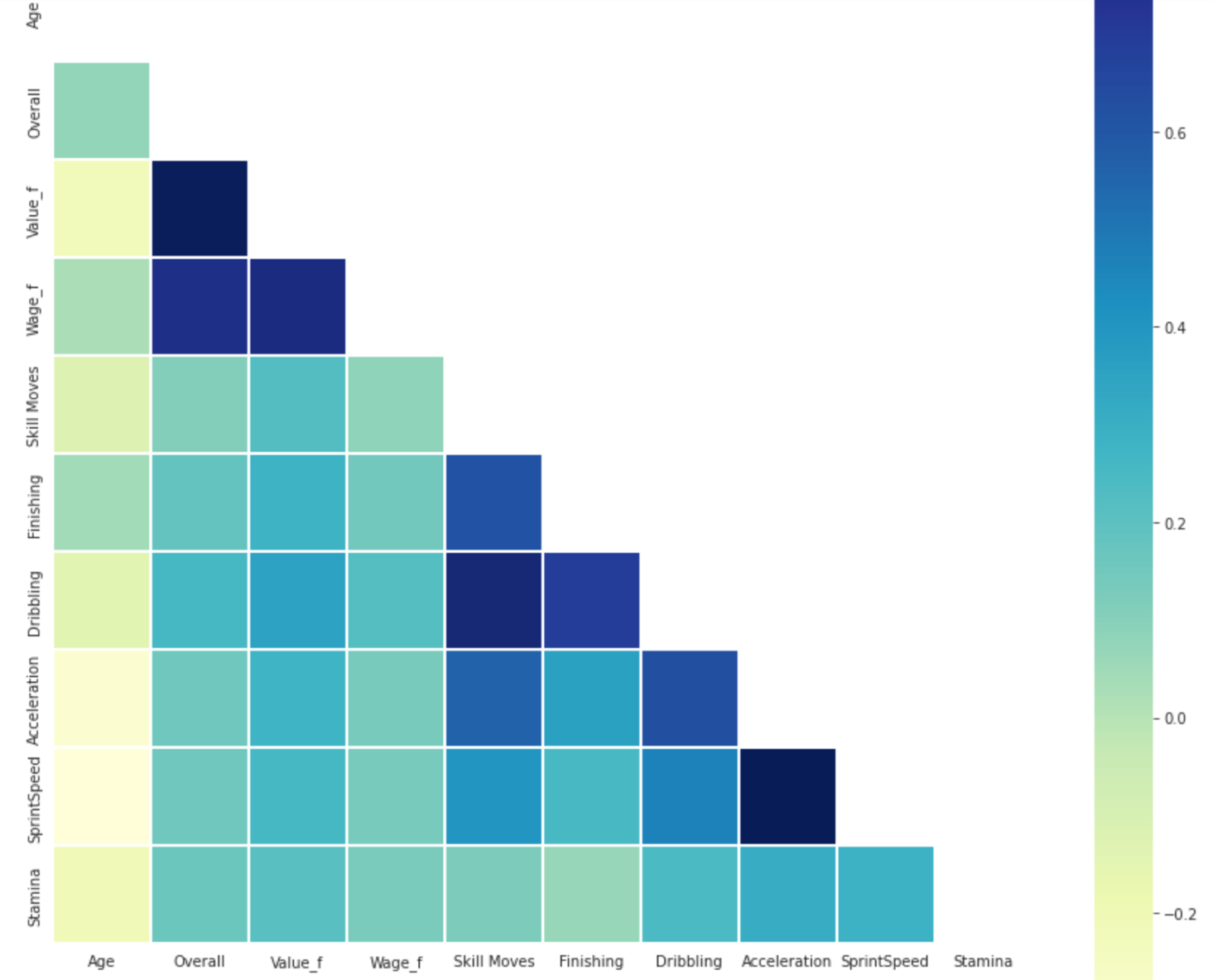
Как видим качество модели увеличилось в 2 раза по сравнению с обычной прямой и ошибка получается совсем небольшой, около 2%. Это так же косвенно подтвержается близкими значениями средних и медиан.

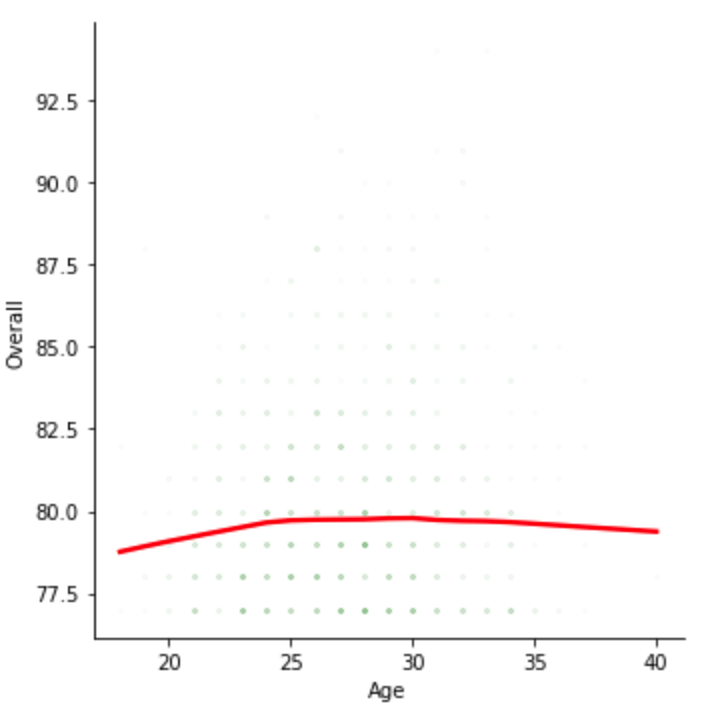
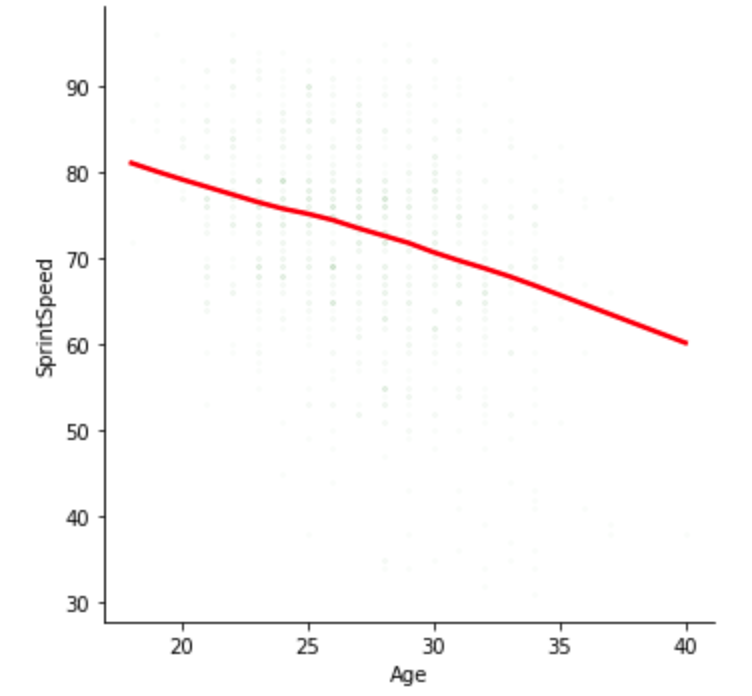
Dribbling Acceleration

**74.52647352647352** **75.91808191808192**

**74.0** **76.0**

Так же для наглядности наших вычислений построим график корреляций между всеми признаками:



Логическое применение алгоритма может быть выявлено в наблюдении различных закономерностей, например:

К сожалению, в среднем скорость угасает с увеличением возраста, а вот характеристика средней оценки уже не так сильно зависит от возраста.

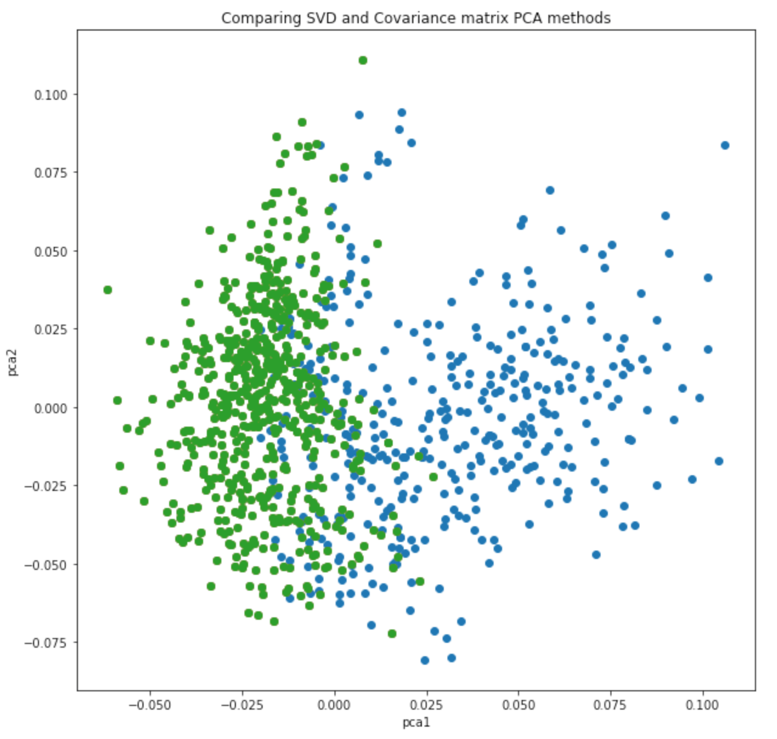
# Метод главных компонент для визуализации данных

Для реализации этого алгоритма было выбрано 3 признака: Stamina, Dribbling и Finishing.

Суть метода главных компонент заключается в том, чтобы на основе каких-либо признаков, изначально представленных в датасете, получить новые, являющиеся линейной комбинацией старых. На лекциях было пару способов реализации, а именно через сингулярное разложение и через собственные векторы матрицы ковариации.

Для этого необходимо нормировать данные и это можно сделать по средствам **range normalization** и **z-scoring.**

После того как были получены новые признакми pca1 и pca2 построим их график распределения, причем оранжевым цветом отмечены те, которые по каждому из pca2 больше среднего, равного нулю после z-scoring.

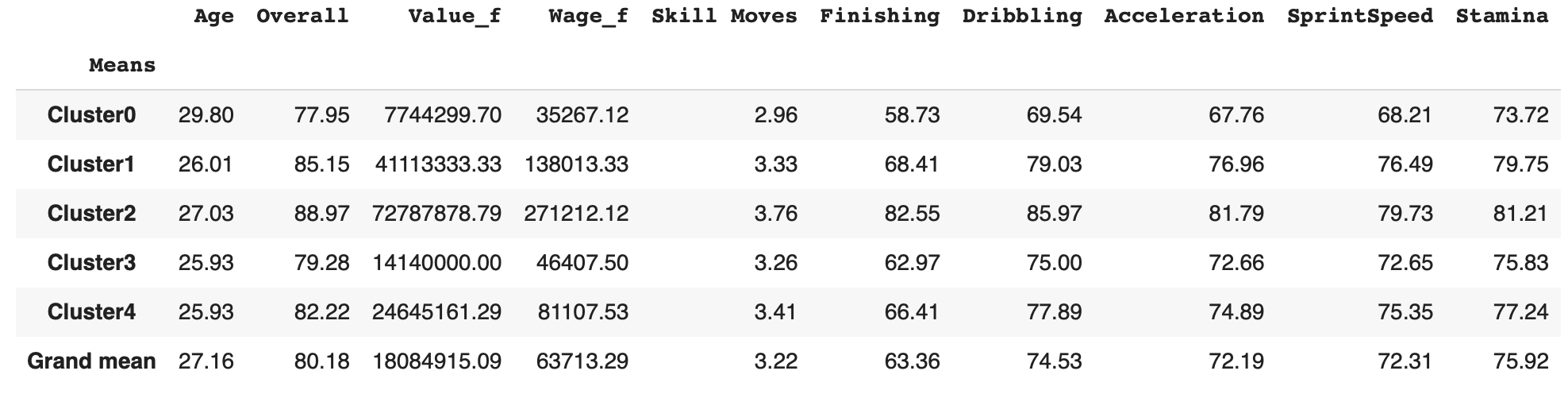
Отметим, что наши полученные кластеры немного пересекаются в центре. Это ожидаемый результат, потому что наши признаки довольно похожи хоть и имеют разное распределение, но принимают одни и те же значения из интервала [40,100]. Так же известно, что помимо кластеризации, метод главных компонент позволяет уменьшить размерности данных, что и видим.

После этого к Z-scored данным был применен другой способ PCA, заключающийся в нахождении собственных векторов матрицы ковариации. Были получены те же самые новые признаки. Таким образом, мы проверили, что оба способа нахождения главных компонент дают одинаковый результат.

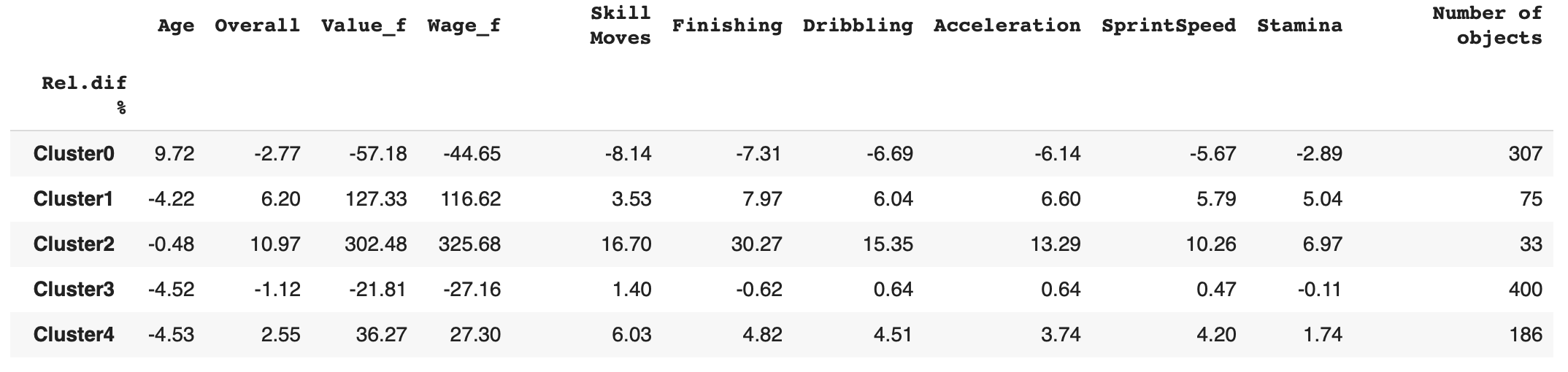
График, подтверждающий совпадение главных компонент, полученных различными способами:

# Интерпретация кластеров в категориальных признаках и хи-квадрат

Средние значения признаков по пяти кластерам:

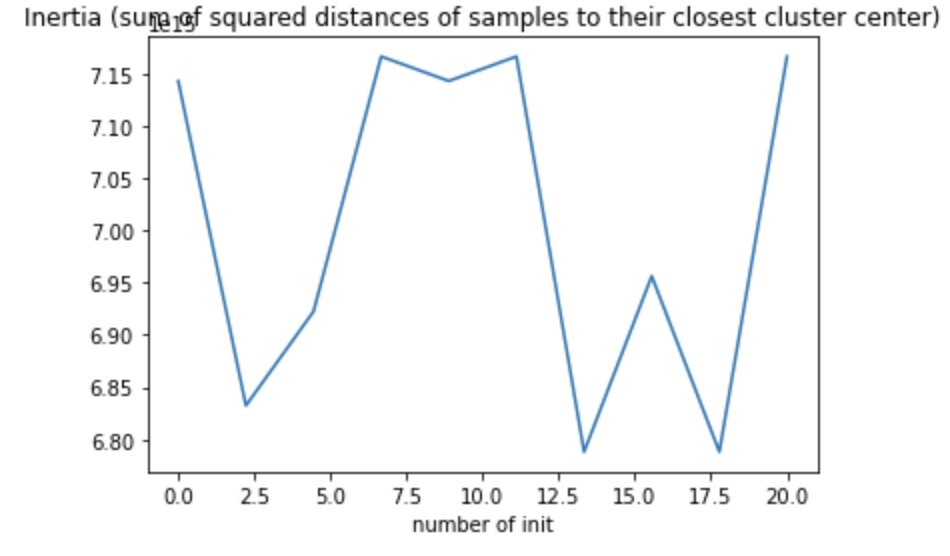


Также посчитаем отклонения внутри-кластерных средних значений признаков от общего среднего значения признака.



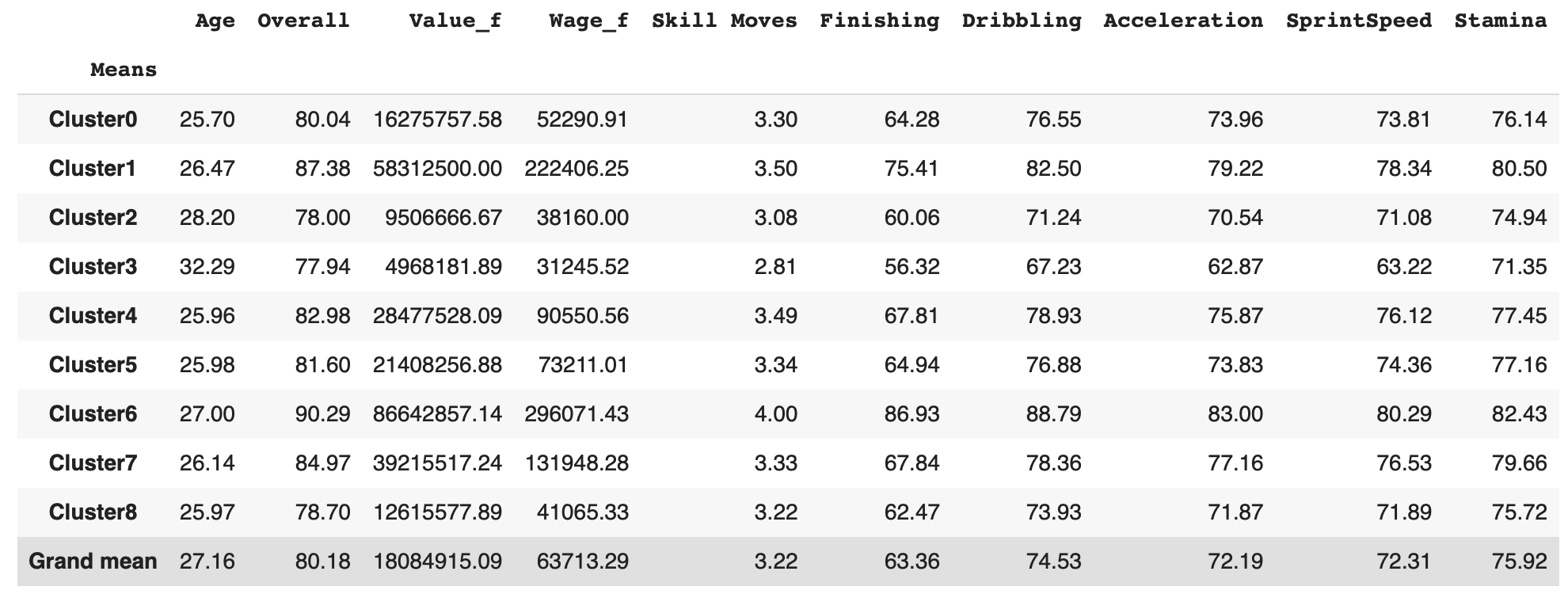
Можем заметить что в группе что попали в кластер 0 самые старые игроки, а в 4 кластере самые младшие.

График распределения минимума критерия метода на каждой из инициализаций:

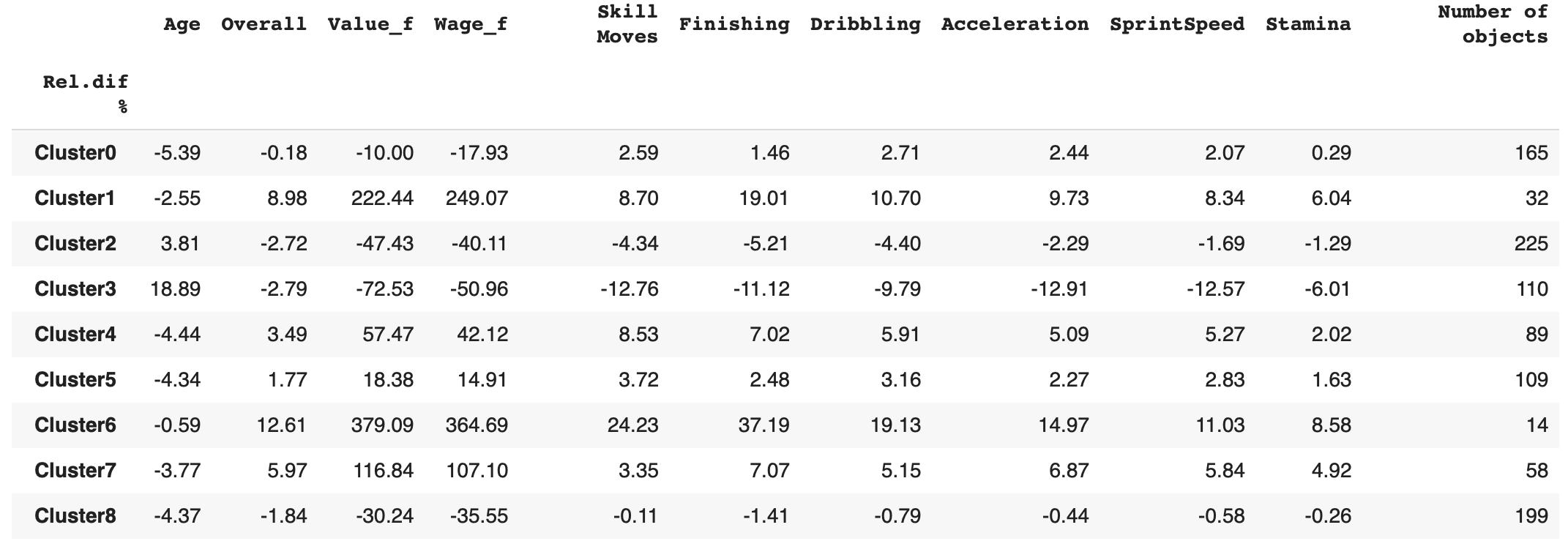


И то же самое делаем для 9 кластеров:

Средние значения выбранных признаков по девяти кластерам:



Отклонения внутри-кластерных средний значений:



Можно заметить что у кластеров 7 и 0 Skill Moves очень близки по значению, при этом отклонения от средних значений у них тоже практически не отличаются. Таких примеров можно привести достаточно, чтобы сделать вывод что такое большое количество кластеров не обязательно для этих данных. По-этому использование 5и кластеров будет более эффективно.

# Таблица сопряженности

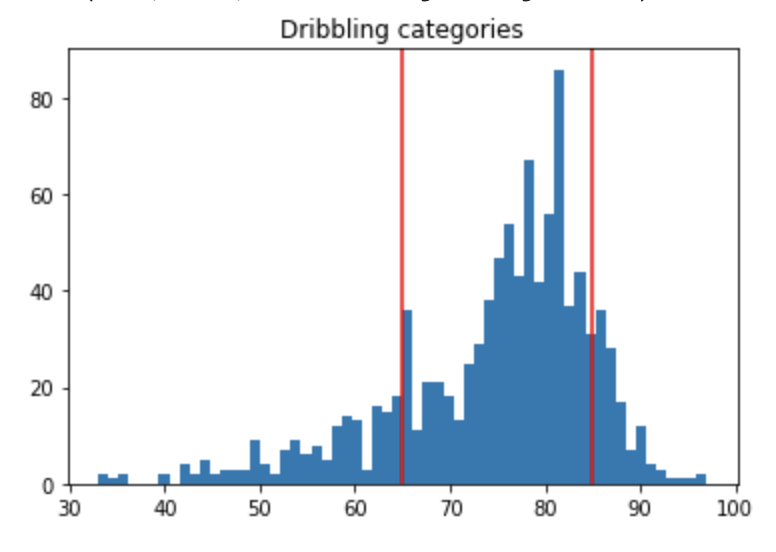
Мы выбрали такие номинальные признаки:

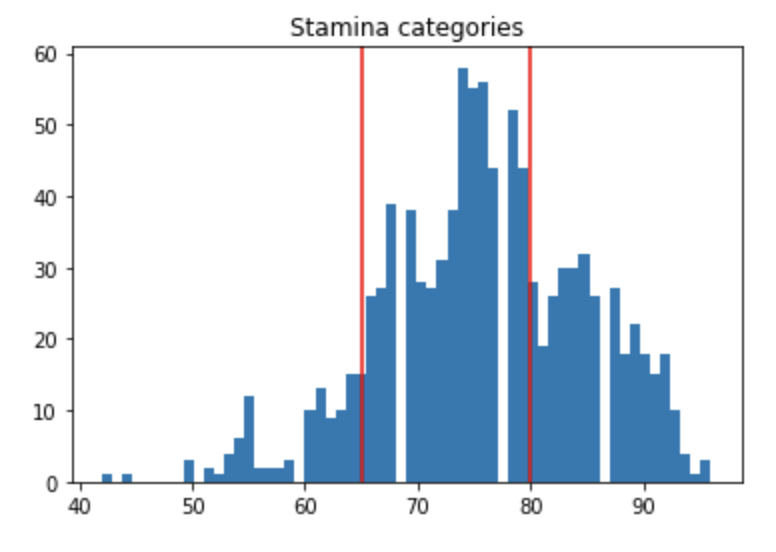
Position - выбрали из нашего датасета.

Stamina - номинальный признак, который сформируем из количественного признака в исходном датасете.

Dribbling - второй номинальный признак.

Посмотрим на распределения Stamina и Dribbling:

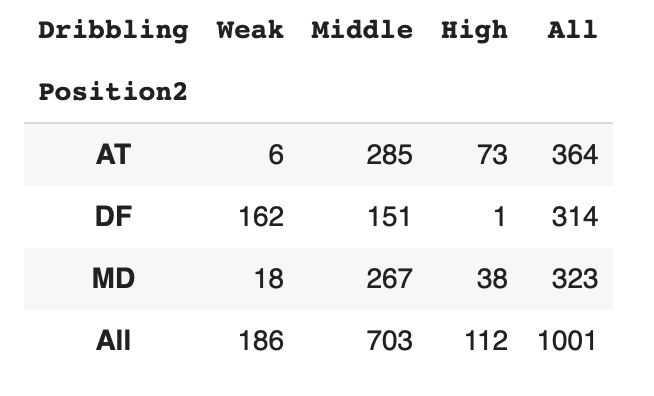


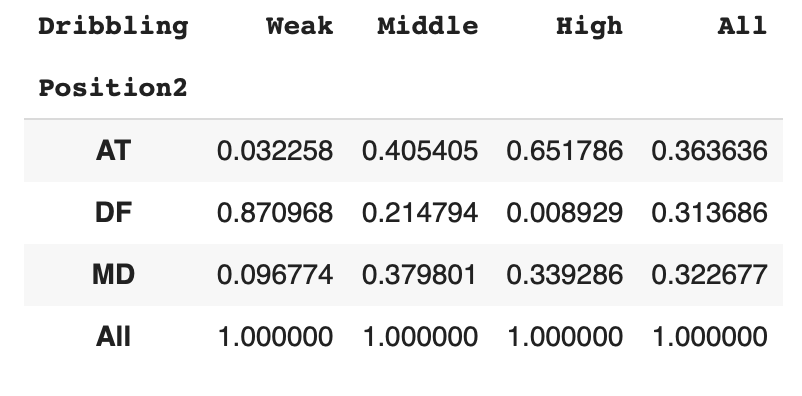


Красными полосами мы поделили эти признаки, каждый на три категории -Weak, Middle, High.

Дрибблинг будем считать Weak, если его значение меньше 67, Middle, если, больше 67 но меньше 86, и все что больше 86 будем считать High.

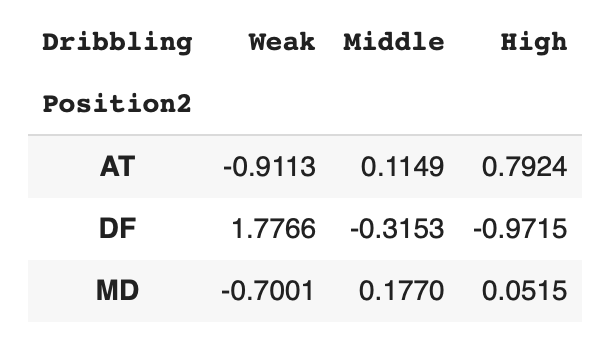
Стамина у нас будет Weak, если его значение меньше 66, Middle, если больше 66 но меньше 81, и High если больше 81.

Далее мы построили таблицу сопряженности и таблицу условной вероятности Position & Dribbling 



По этим таблицам можно сделать вывод что атакующих игроков с плохим дрибблингом очень мало, а игроки с хорошим дрибблингом в осномном играют в нападении.

Далее мы построили таблицу индексов Кетле:



Видно, что сохраняются те связи что и с таблицей сопряженности и с условной вероятностью.

Средний индекс Кетле = 0.36060

Далее перейдем к признаку Stamina.

Таблица сопряженности с Position:

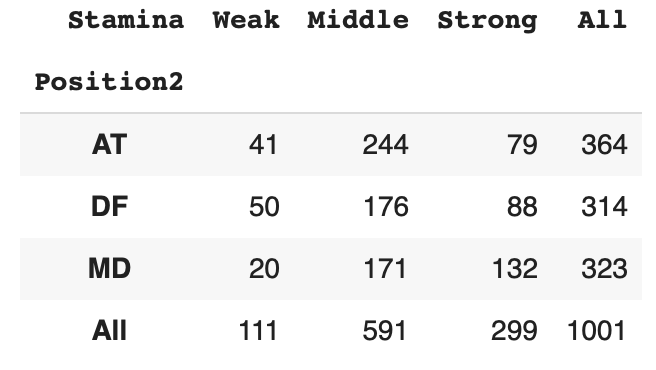
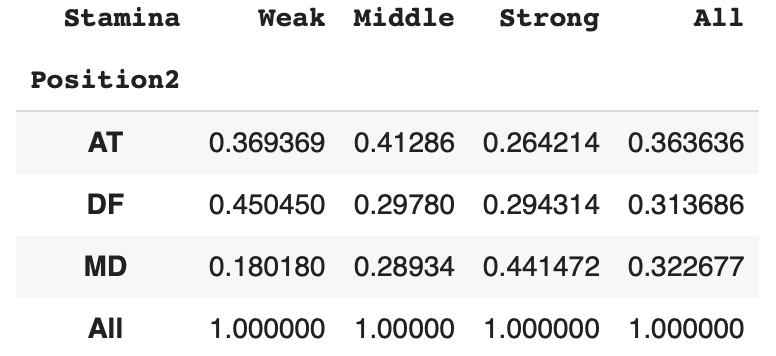
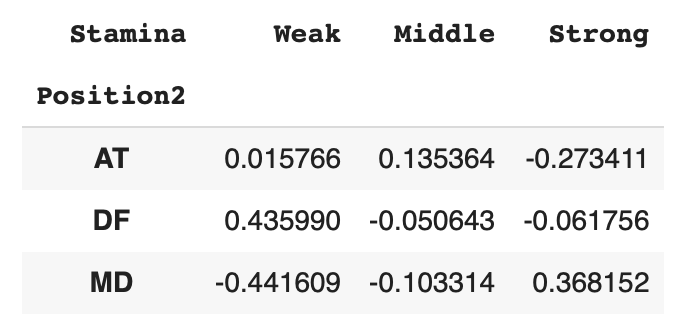


Таблица условных вероятностей:



Тут можно сделать те же выводы, как и с предыдущим признаком, так как выносливых игроков ставят чаще всего на позицию атакующих.

Матрица Кетле:



Средний индекс Кетле = 0.04158

Из индексов Кетле можно сделать вывод что связь имеют признаки Position & Dribbling, нежели Position & Stamina. Наверное все таки в выборе позиции игрока больше играет роль Дрибблинг чем Выносливость, ибо возможно игроки с малой выносливостью могут иметь высокое значение дрибблинга, и при этом показывать хорошую скорость на малой дистанции, в то время как полузащитникам и защитникам дрибблинг не так важен как выносливость.

Для двух пар признаков были посчитаны хи-квадраты:

Для пары с дрибблингом = 0.36060

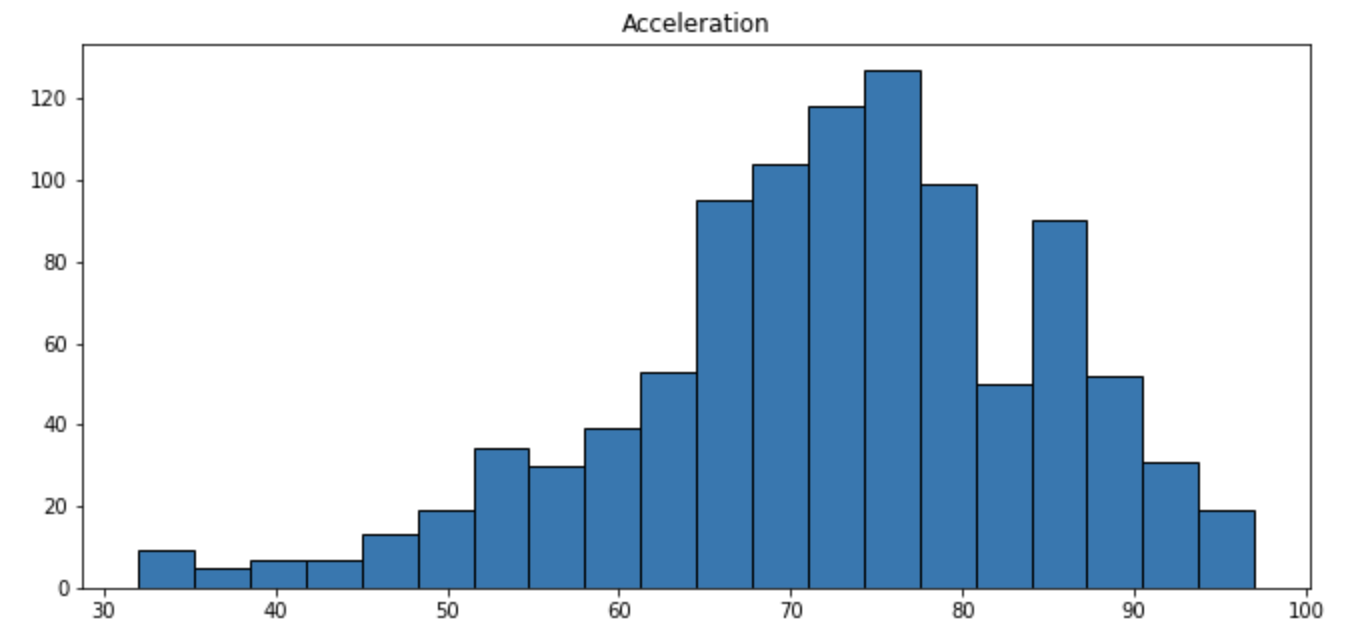
Для пары с выносливостью = 0.0415

Как и ожидалось, они совпадают с индексами Кетле.

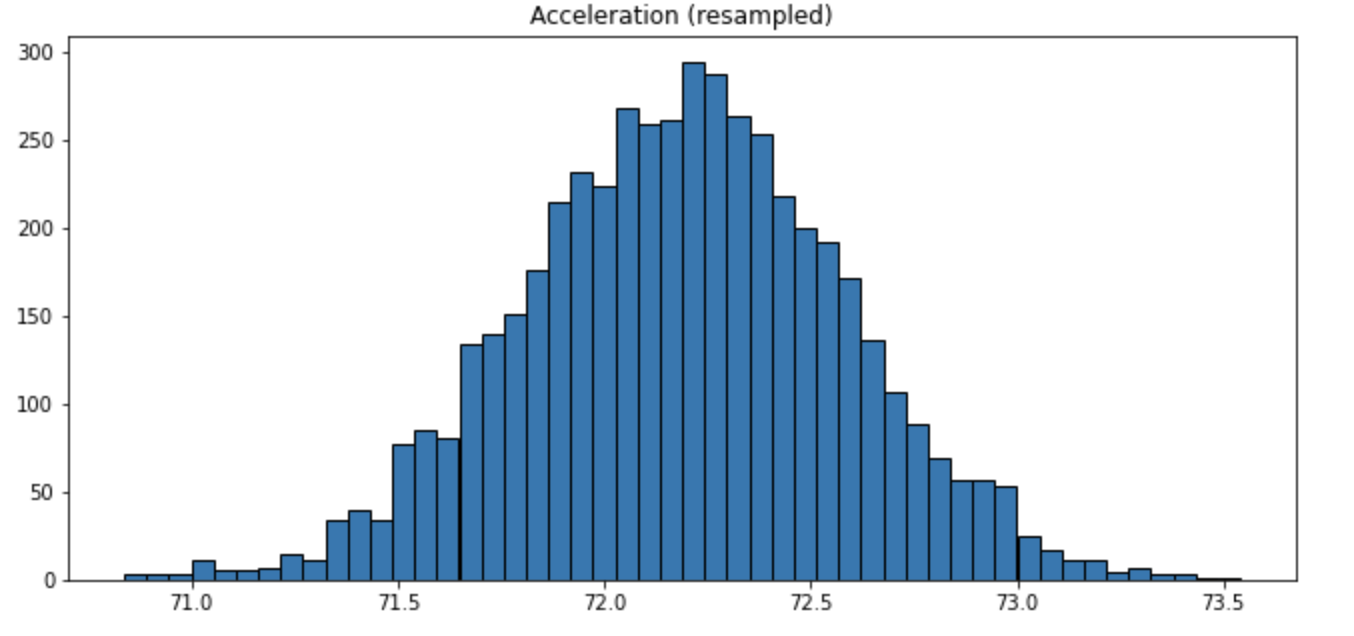
# Сравнение средних с помощью бутстрэпа

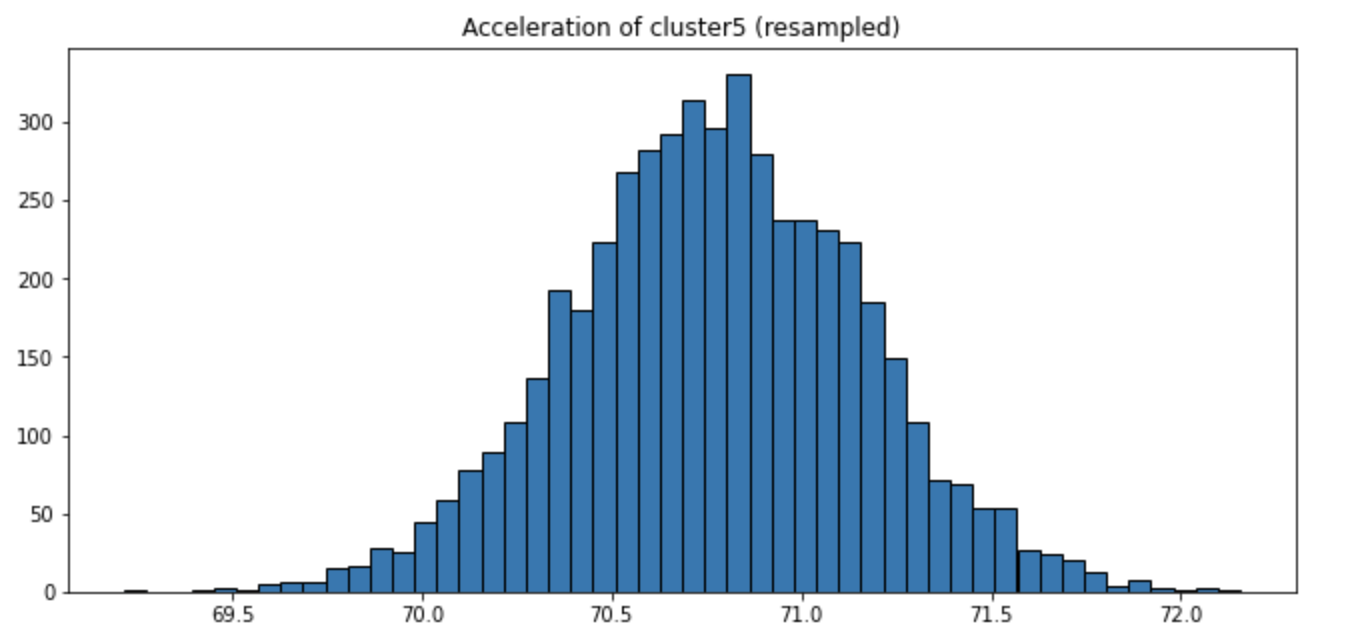
Тут мы использовали 5000 итераций для бутстрэпа. В качестве признака использовали признак Acceleration.

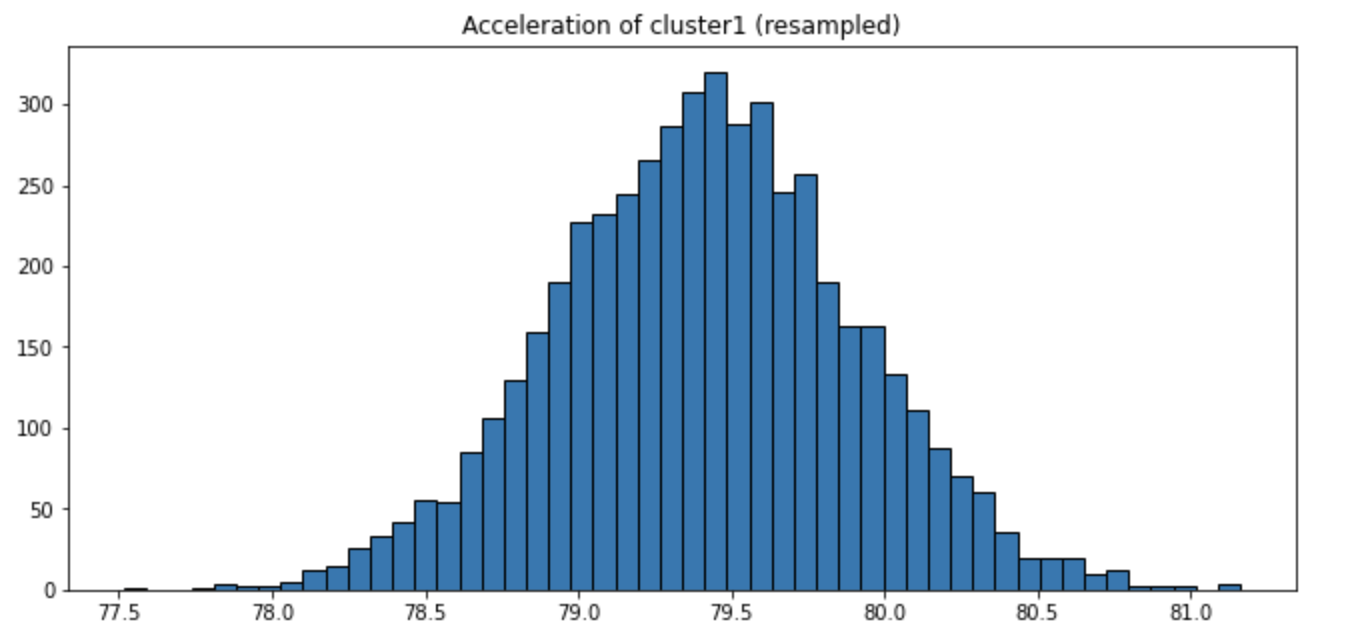
Для начала посмотрим изначальное распределение этого признака:



Далее, при помощи бутстрэпа, мы получили 5000 выборок объектов по этому признаку. Посмотрим распределение средних значений этого признака на всей выборке, а также на 1 и 5 кластере:







Далее посчитаем 95% доверительный интервал по всей выборке для этого признака:

С опорой = 71.42, 72.95

Без опоры = 71.42, 72.96

95% доверительныйинтервал для разности средних значений признака на кластерах 1 и 5:

С опорой = 7.49, 9.79

Без опоры = 7.49, 9.78

95% доверительный интервал для разности средних значений признака на кластере 5 и всей выборке.

С опорой = 6.25, 8.21

Без опоры = 6.28, 8.22

По данным результатам можно сказать что значения полученные при помощи бутстрэпа с опорой и без опоры почти не отличаются.

# Заключение

Возможно, помимо простого применения алгоритмов наши результаты могут быть полезны менеджерам футбольных клубов для лучшего подбора футболистов и достижения спортивных результатов. Были использованы такие алгоритмы и методы анализа данных, как K-means, бутстрэп, таблица сопряженности, таблица условных вероятностей, метод главных компонент, линейная регрессия и другие. Каждый метод позволил проанализировать данные нашего датасета и найти какие-то закономерности.

Скорее всего мы подобрали не лучший вариант датасета для этого задания так как много похожих признаков таких как Dribbling, Acceleration, Stamina, Sprint Speed.