ÜK M259 Machine Learning 26.09.2024 Luis Allamand

Dataset: https://www.kaggle.com/datasets/kukuroo3/body-performance-data
Thema: Vorhersage der Fitnessleistung basierend auf körperlichen und gesundheitlichen Merkmalen

Dataset Beschreibung:

Dies sind Daten, die den Grad der Leistung mit dem Alter und einige Daten zur körperlichen Leistung bestätigen. Das Datenset beinhaltet 12 Spalten in denen sich 2 STRING und 10 DECIMAL Variablen.

EDA:

1. Datenverständnis

Variablen:

Variablen	Typ	Kurzbeschreibung
age	Float64	Das alter wird nach Jahr eingestuft, hier gibt es nur ganze Zahlen
gender	String	Das gender wird in M(Male) und F(Female) eingeteilt
height_cm	Float64	Die Körpergrösse wird in cm gemessen, dezimal werte sind für ein genaues Ergebnis gegeben.
weight_kg	Float64	Das Gewicht wird in KG gemessen, dezimal werte sind für ein genaues Ergebnis gegeben.
body_fat%	Float64	Der Körperfettanteil wird in % gerechnet vom Körpergewicht der Person
diastolic	Float64	Diastolic ist der Druck in den Arterien, wenn das Herz zwischen den Schlägen entspannt ist, in mmHg.
systolic	Float64	Systolic bezieht sich auf den Druck in den Arterien, wenn das Herz sich zusammenzieht und Blut in den Körper pumpt.
gripForce	Float64	Die Griffkraft wird in Kg gemessen.
sit and bend forward_cm	Float64	Bei dieser Variable wird die Distanz in cm gemessen, bei der man sich nach vorne beugen kann im Sitzen
sit-ups counts	Float64	Hier wird die maximale Anzahl der sit-ups gemessen, die die Person kann.
broad jump_cm	Float64	Beim Standweit Sprung wird die maximale Distanz gemessen, die der Athlet die Athletin zurücklegt,
class	String	Bei der class variable wird von A bis D eingeteilt, wie körperlich fit man ist A ist das beste und D das schlechteste

Nach dem Ich die Variablen beschrieben habe, mache ich mir einen ersten Einblick in meinen Datensatz, indem ich data_train.describe() anwende, um erste Ergebnisse zu sehen. Hier das Ergebnis:

	age	height_cm	weight_kg	body fat_%	diastolic	systolic	gripForce	sit and bend forward_cm	sit-ups counts	broad jump_cm
count	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000	13393.000000
mean	36.775106	168.559807	67.447316	23.240165	78.796842	130.234817	36.963877	15.209268	39.771224	190.129627
std	13.625639	8.426583	11.949666	7.256844	10.742033	14.713954	10.624864	8.456677	14.276698	39.868000
min	21.000000	125.000000	26.300000	3.000000	0.000000	0.000000	0.000000	-25.000000	0.000000	0.000000
25%	25.000000	162.400000	58.200000	18.000000	71.000000	120.000000	27.500000	10.900000	30.000000	162.000000
50%	32.000000	169.200000	67.400000	22.800000	79.000000	130.000000	37.900000	16.200000	41.000000	193.000000
75%	48.000000	174.800000	75.300000	28.000000	86.000000	141.000000	45.200000	20.700000	50.000000	221.000000
max	64.000000	193.800000	138.100000	78.400000	156.200000	201.000000	70.500000	213.000000	80.000000	303.000000

(Bild 01)

Hier eine kurze Bildanalyse (Bild01)

count: Anzahl der Datensätze (13393 Personen).
mean: Durchschnittswert (Mittelwert).
std: Standardabweichung, ein Maß für die Streuung der Werte.
min: Minimalwert.
25%: Das 25. Perzentil – 25 % der Werte liegen unter diesem Wert.
50% (Median): Das 50. Perzentil – der mittlere Wert der Verteilung.
75%: Das 75. Perzentil – 25 % der Werte liegen über diesem Wert.
max: Maximalwert.

#	Column	Non-Null Count	Dtype
0	age	13393 non-null	float64
1	gender	13393 non-null	object
2	height_cm	13393 non-null	float64
3	weight_kg	13393 non-null	float64
4	body fat_%	13393 non-null	float64
5	diastolic	13393 non-null	float64
6	systolic	13393 non-null	float64
7	gripForce	13393 non-null	float64
8	sit and bend forward_cm	13393 non-null	float64
9	sit-ups counts	13393 non-null	float64
10	broad jump_cm	13393 non-null	float64
11	class	13393 non-null	object

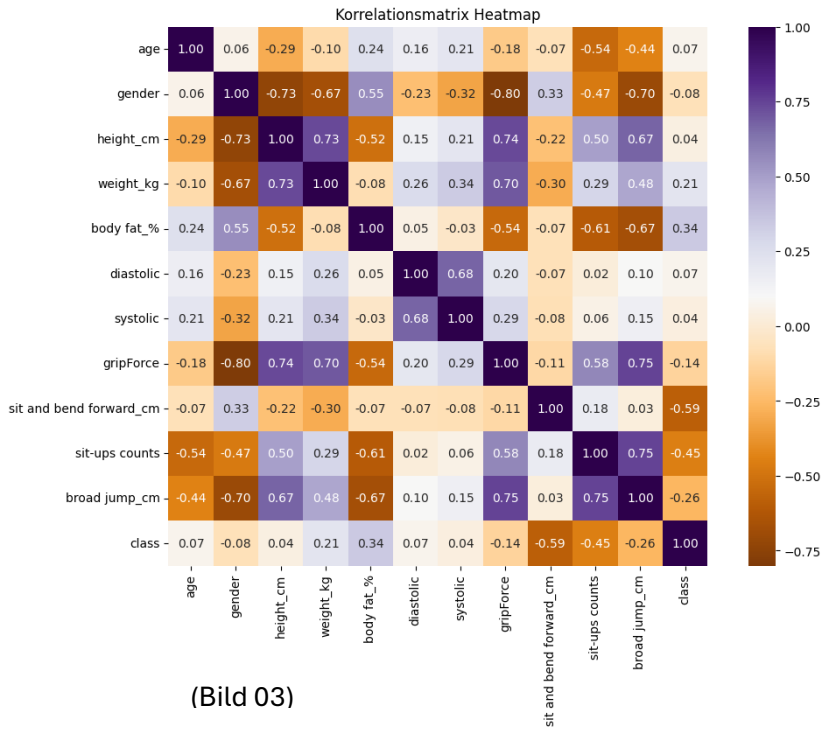
(Bild 02)

Hier eine kurze Bildanalyse (Bild02)

#: Aufzählung, 0 bis 11
Column: Alle Variablen im data_train auflisten.
Nun-Null Count: Hier werden gezählt ob es nicht vorhandene Felder in den Variablen gibt.
Dtype: Bei der Spalte Dtype gibt es den Variablen typ aus.

Duplikate:

Bei den Duplikaten kann Ich nicht viel machen, da Ich viele Daten habe, die mehrere male vorkommen, wie zum Beispiel Alter, Gewicht, Gender, Class und Körpergrösse. Deshalb lösche Ich keine Duplikate, da Ich sonst mehrere tausend Datensätze weniger habe.



(Bild 03)

```python
class_replace = {
    'A':1,
    'B':2,
    'C':3,
    'D':4
}
gender_replace = {
    'M':1,
    'F':2
}
data_train['class'] = data_train['class'].replace(class_replace)
data_train['gender'] = data_train['gender'].replace(gender_replace)
```

(Code ausschnitt 01)

Um solch eine Korrelationsmatrix zu bekommen, muss man zuerst das ‘gender’ und die ‘class’ zu float oder int werten ändern, dies mache Ich mit dem Code 01.

```python
ausgewaehlte_variablen = ['age','gender','height_cm','weight_kg','body fat_%','diastolic','systolic',
                          'gripForce','sit and bend forward_cm','sit-ups counts','broad jump_cm','class']

korrelationsmatrix = data_train[ausgewaehlte_variablen].corr()

plt.figure(figsize=(12, 8))
sns.heatmap(korrelationsmatrix, annot=True, fmt=".2f", cmap='PuOr', square=True, cbar=True)

plt.title('Korrelationsmatrix Heatmap')
plt.show()
```

(Code ausschnitt 02)

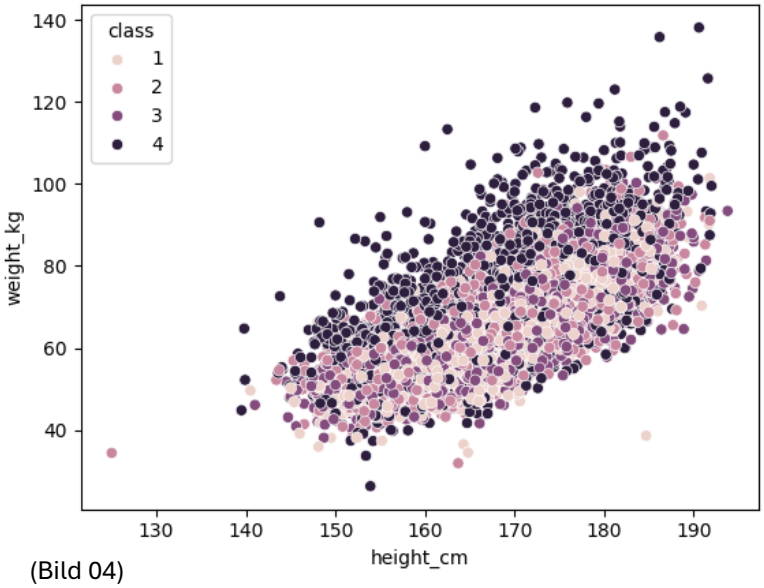
Um die Matrix jetzt noch darzustellen verwende Ich ein heatmap. Ich habe die Farben noch angepasst.
Code 02

Korrelationsmatrix (Bild03)

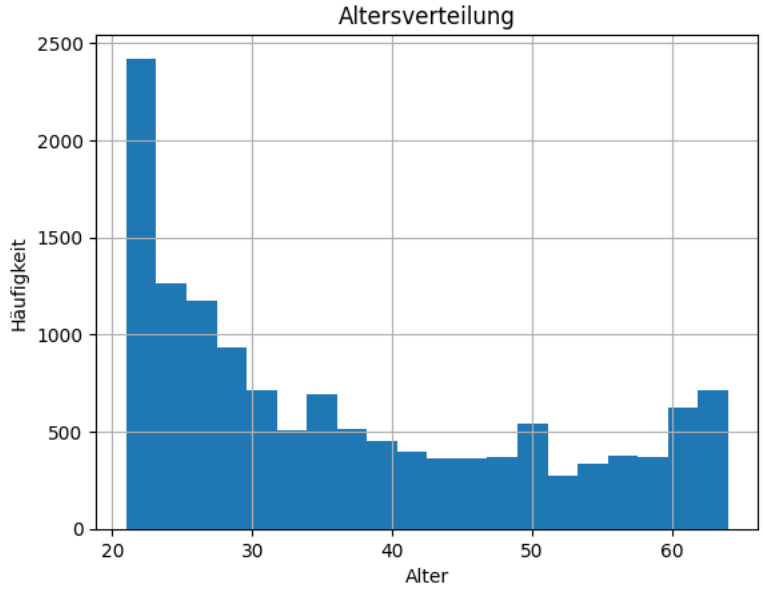
Einige interessante Korrelationen:

- **weight_kg und body_fat_%**: Eine positive Korrelation von **0.48**, was darauf hindeutet, dass mit zunehmendem Gewicht auch der Körperfettanteil steigt.
- **sit-ups_counts und broad_jump_cm**: Hohe positive Korrelation von **0.75** – Personen, die viele Sit-ups schaffen, neigen auch dazu, besser im Weitsprung zu sein.
- **age und sit-ups_counts**: Negative Korrelation von **-0.54** – Ältere Personen machen im Durchschnitt weniger Sit-ups.
- **height_cm und weight_kg**: Hohe positive Korrelation von **0.73** – Größere Personen wiegen tendenziell mehr.
- **gripForce und sit-ups_counts**: Eine starke positive Korrelation von **0.58** – Menschen mit höherer Griffkraft können tendenziell mehr Sit-ups machen.
- **gripForce und gender:** Bei der Griffkraft kann man sehen, dass es eine negative Korrelation von **-0.80** – Männer haben tendenziell eine stärkere Griffkraft.

```
sns.scatterplot(data=data_train, x="height_cm", y="weight_kg", hue="class")
```
(Code ausschnitt 03)



(Bild 04)



(Bild 05)

Scatterplot (Bild 4)

Ich habe mich noch für ein Scatterplot entschieden. Der Code sieht man im Code 03

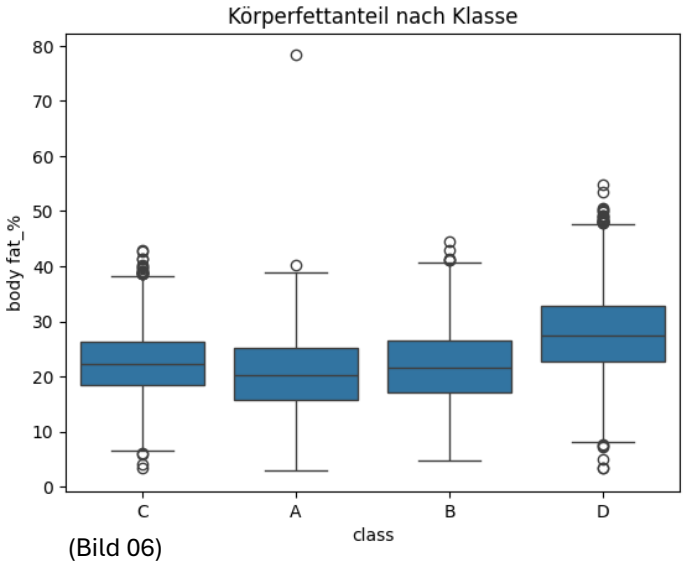
Was auf den ersten Blick unübersichtlich und wie ein Durcheinander aussieht, kann eine Einsicht in die class variablen werfen, um zu verstehen, wie sie funktionieren und welche Eigenschaften sie haben.
1. Die dunklen Punkte sind üblicherweise im oberen Teil. Dies bedeutet, dass Personen mit mehr Gewicht, einen schlechteren 'class' Wert haben.
2. Die hellen Punkte sind eher unten beherbergt.
Somit kann man interpretieren, dass leichter Menschen einen besseren 'class' Wert haben.

Histogramm (Bild 5)

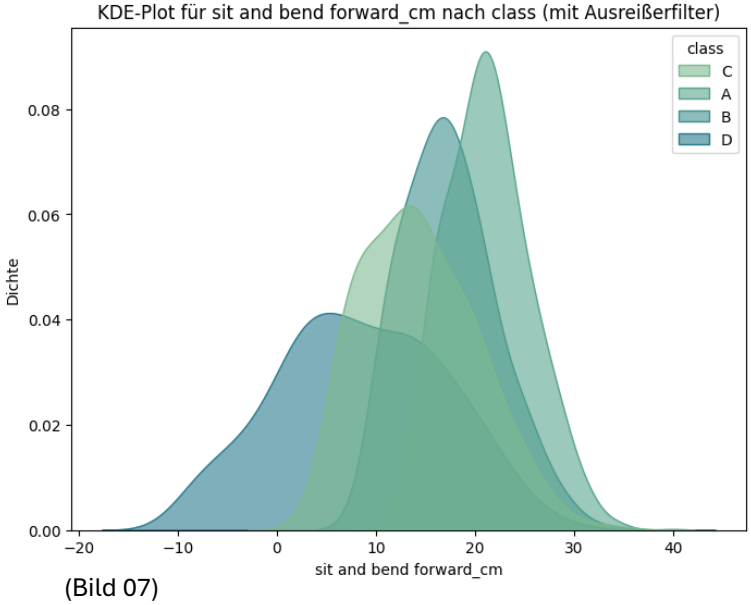
Um mehr über die Verteilung der Daten zu wissen, erstellte Ich ein Histogramm über die Alters Verteilung, und ihre Häufigkeit.

Bei der Verteilung habe Ich gemerkt, dass sehr viele Daten im Alter von 20-25 gesammelt wurden.

Boxplots:



(Bild 06)



(Bild 07)

<u>Boxplot 01</u> (Bild 6)

Bei einer richtiger EDA, darf ein Boxplot nicht fehlen. Hier können wir sehen, was der Median ist, wo das Max. und Min. sind, und bei welchen Daten, das System denkt, es gäbe Ausreisser Daten.

Gruppe A: In der Kerze, sehen wir, dass es einen Mindestwert gib von unter 5, und einen Max. Wert, von 40, mit einem Ausreisser.

Gruppe B: In dieser Kerze, sehen wir einen minimen Unterschied bei den Min. Max. Werten und bei dem Ausreisser.

Gruppe C: Hier haben wir deutlich mehr Ausreiser, das liegt daran, dass wir eine grössere Verteilung der Daten haben.

Gruppe D: Bei der D Kerze, haben wir ebenfalls mehr Ausreisser aus dem gleichen Grund wie vorhin.

<u>KDE-Plot sabf = (sit and bend forward)</u>

Ich wollte noch mehr Grafiken zu der ‘class’ Variable, da diese wichtig für unsere spätere Modelle wichtig sein wird.

Die Dichte ist hier neu dazugekommen, und bedeutet wie dicht ein wert ist dies geht von 0.0-0.1. Wenn mehrere gleiche Werte von **sabf** vorkommen, dann ist der Graph höher und somit dichter.

Bei dieser grafik, haben wir mehrere Merkmale, die wir analysieren können.

- Zuerst muss geklärt werden, warum wir Werte unter 0 haben. Diese haben wir, weil die Bandbreite zu fest angepasst wurde, oder die Maschine eine Glatte Kurve erzeugen möchte.
- Warum haben wir eine grössere Dichte bei der ‘class’ A. Im Bild 05 konnten wir ja sehen, dass es mehr Daten über junge Personen gibt als alte Personen. Da wir auch wissen, dass junge Menschen eher bei ‘class’ A oder B eingestuft sind, haben wir eine hohe Dichte bei der ‘class’ A

2. Datenquellen

Die Daten stammen von der Webseite Kaggle die mir von der Instruktorin empfohlen wurde. Die Webseite beinhaltet sehr viele Datasets, die von User hochgeladen wurden.
In meinem Fall handelt es Sich um das Dataset Body Performance Data von dem User KUKUROO3, der das Dataset vor 2 Jahren aktualisiert hat.
In der Beschreibung wurde folgender Link geschrieben (siehe unten) der als Quelle angegeben wurde. Dieser führt auf eine Koreanische Webseite 문화 빅데이터 플랫폼 übersetzt heisst das Kulturelle Big Data-Plattform. Leider kann Ich kein Koreanisch und kann somit die Quelle nicht genau analysieren.

Link: https://www.bigdata-culture.kr/bigdata/user/data_market/detail.do?id=ace0aea7-5eee-48b9-b616-637365d665c1

3. Datenbereinigung

Ich hatte 2 Variablen, die mit Buchstaben gefüllt sind. Das ist einmal die Variable gender, dort habe Ich entweder 'M' oder 'F', und die Variable class, wo Ich 'A', 'B', 'C' oder 'D' habe. Diese habe Ich dur Zahlen ersetzt, also A=1 B=2... Wie man im Code 04 lesen kann.

```
class_replace = {
    'A':1,
    'B':2,
    'C':3,
    'D':4
}
gender_replace = {
    'M':1,
    'F':2
}
data_train['class'] = data_train['class'].replace(class_replace)
data_train['gender'] = data_train['gender'].replace(gender_replace)
```

(Code ausschnitt 04)

Ich habe zwar Duplikate, jedoch sollte Ich diese nicht löschen. Wenn Ich meine Duplikate löschen würde, hätte Ich ca. 8000 Datensätze weniger. Dies hätte wiederum ein Ausmass auf das was Ich schon analysiert habe

4. Wichtigste Erkenntnisse EDA

Erkenntnis01:

Mein erstes Erkenntnis ist, dass das Dataset viele Korrelationen hat. Daher werde Ich auf gute Resultate bei den Modellen kommen. Somit fällt es mir einfacher 3 funktionierende Modelle zu errichten.

Erkenntnis02:

Das zweite Erkenntnis ist, dass Ich eine klare Trennung bei der 'class' Variable habe, und es somit einfacher fällt für die Maschine diese zu unterteilen.

Hypothesen:

Hypothese 1:

Männer haben eine deutlich **stärkere Griffkraft** als **Frauen**.

Untersuchung:

- Bei der Untersuchung der Korrelationsmatrix, erkennen wir, dass die Griffkraft ein hoher Einfluss auf das Geschlecht hat.
- Das Modell 02 bestätigt den ersten Punkt, da die Gewichtung auf der Grafik (Bild 9) den höchsten Wert mit 0.45 erreicht.

Fazit: Das Geschlecht hat eine Auswirkung auf die Griffkraft.

Hypothese 2:

Personen mit **höherem Körperfettanteil** ‘body fat_%’ haben eine **geringere** körperliche **Leistungsfähigkeit**, **gemessen** an der **Griffkraft** ‘gripForce’, der **Anzahl** der **Sit-Ups** ‘sit-ups counts’ und der **Weitsprungweite** ‘broad jump_cm’.

Untersuchung:

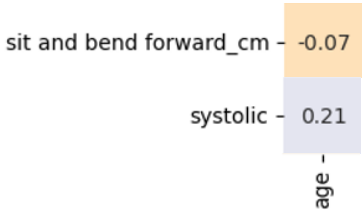
- Beim Betrachten der Korrelationsmatrix sehen wir, dass der Wert 0.34 beträgt. Der Wert ist nicht besonders gross, kann trotzdem etwas aussagen.
- Auf der Grafik vom Modell01 Versuch01(Bild 8) kann man erkennen, dass die Wichtigkeit an zweiter Position ist, und das trotz der geringen Korrelation.
- Die weiteren 3 Eigenschaften haben ebenfalls einen grossen Einfluss auf die körperliche Leistungsfähigkeit. Die Wichtigkeit von diesen Variablen kann man ebenfalls im Bild 8 sehen.

Fazit: Die Hypothese stimmt, aber in meinem Fall ist die sit and bend forward Variable geeigneter für meine Behauptung.

Hypothese 3:

Ältere Personen zeigen einen **höheren** systolischen **Blutdruck** ‘systolic’ und eine **geringere Flexibilität**, gemessen durch den Wert für ‘sit and bend forward_cm’, im Vergleich zu jüngeren Personen.

Untersuchung:



- Analysieren wir zuerst die Korrelation zwischen den 3 variablen. Die zwei Variablen sit and bend forward und systolic korrelieren nicht hervorragend mit der age Variable.
- Bei dem Modell 03 konnte Ich mithilfe einer Linearen-Regression, wie im Bild 10, die Beziehung bestimmen, und habe einen R^2 Wert bekommen von 0.04. Dies ist ein sehr schlechter Wert.
- Obwohl Ich mehrere ältere Menschen kenne, die einen Bluthochdruck haben, komme Ich mit meinen Graphen auf und Zahlen nicht auf einen geeigneten Wert.
- Ich habe auch recherchiert (Quelle 02), und bin auf mehrere Webseiten gekommen, die besagen, dass im höheren Alter einen grösseren Blutdruck Wert zustande kommt.
- Mein Ergebnis mit einem niedrigen R^2-Wert bedeutet nicht unbedingt, dass die Hypothese falsch ist. Es bedeutet eher, dass das Alter allein nicht ausreichend ist, um den systolischen Blutdruck zuverlässig vorherzusagen. Es könnte sein, dass andere Variablen (wie Gewicht, Fitnesslevel, Ernährung usw.) stärker ins Gewicht fallen.

Fazit: Nach meinen Modell stimmt meine Hypothese nicht, trotzdem ist die Behauptung nicht ausgeschlossen.

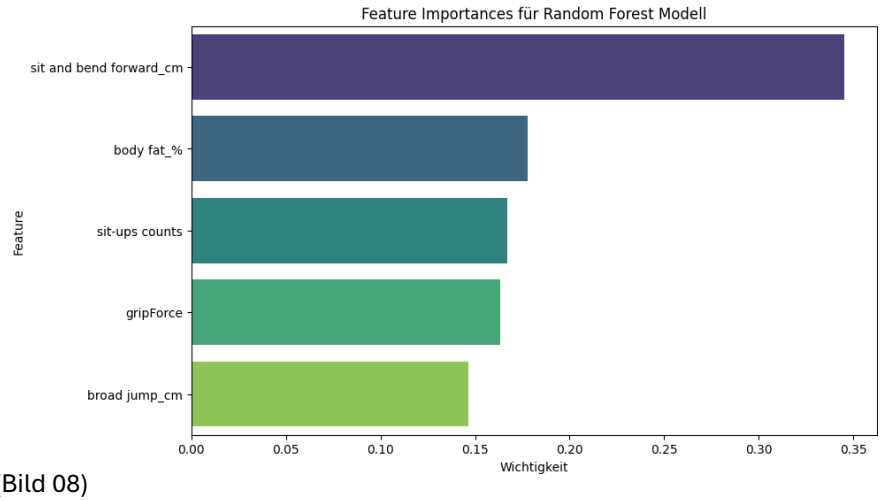
Modelle:

Modell01: Ein Modell das die Fitnessklasse eines Kunden (A, B, C oder D) basierend auf verschiedenen Körperlichen und Leistungsmerkmalen vorhersagen kann.

Versuch 1.

Unter Modelle im Colab ist der erste Versuch gecodet.
Der war sehr schlecht, mit einer accuracy von 62.75%
Dieser Wert muss definitiv verbessert werden. Um ein neues Modell zu erstellen muss Ich wissen, an was es gescheitert ist.

Ich habe in meinem 1. Versuch den Random Forest Classifier benutzt. Das könnte das Problem gewesen sein. Wenn ein Dataset nicht genug analysiert wird und nicht ein optimaler Algorithmus genommen wird, dann kann das Probleme mit der Genauigkeit des Modells geben.



(Bild 08)

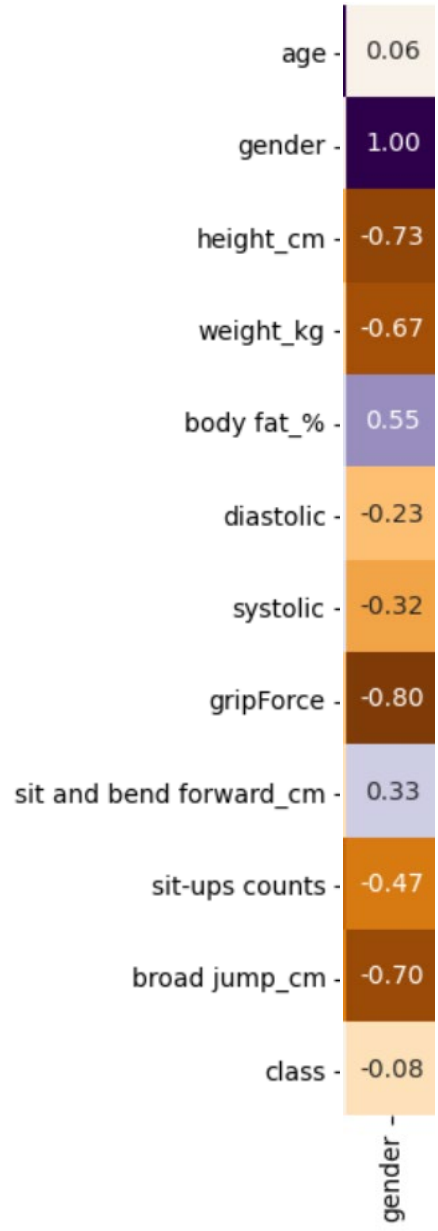
Versuch 2.

Den Code findet man wie im Versuch im Colab.
Ich versuchte jetzt mehrere Modelle, die mir Google und KI vorgeschlagen haben. *Quelle01*. Der VotingClassifier basiert auf dem Ensemble Lernverfahren das mehrere Klassifikatoren kombiniert, um de Genauigkeit zu verbessern. Bei dem Klassifikator kann man entscheiden zwischen Soft Voting und Hard Voting. Die Soft Voting wird die Wahrscheinlichkeit vorhergesagt. Beim Hard Voting wird die Mehrheitsabstimmung angewendet.

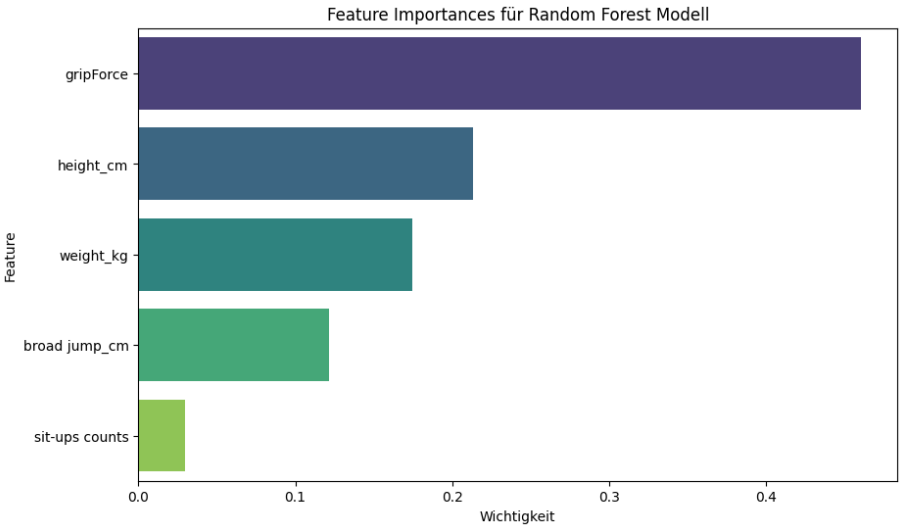
Mit dem Modell konnte Ich eine accuracy von 77.19% erzielen. Also konnte Ich die Maschine um mehr als 13% verbessern. Dies ist immer noch kein unglaublicher Wert, weil wir im Unterricht Modelle gecodet haben, die nochmals 10%-15% besser waren.
Also musste Ich weiter studieren, warum das Ergebnis nicht einwandfrei ist, und habe das Target nochmals analysiert. Bei der Korrelationsmatrix wurden die richtigen Werte genommen, also nicht die Werte, die einen schlechten Wert haben, aber mir ist aufgefallen, dass bei der 'class' variable 4 Ergebnisse gesucht werden*(A, B, C oder D),* Dies bedeutet, dass wenn ein Affe immer zufällig entscheiden würde, er «Theoretisch» eine accuracy von 25% hätte *100% / 4 = 25%*. Also ist der Wert nicht so schlecht. Mit mehr Datensätze könnte man den Wert ebenfalls verbessern. Da Ich aber keine weitere Datensätze habe, kann Ich an dem nichts ändern.

Modell02: Ein Modell, dass das Geschlecht einer Person vorhersagt, Körpergrösse, Gewicht, Griffkraft, Standweitsprung und sit-ups.

Versuch 1.



Bei meinem ersten Versuch hat es direkt perfekt funktioniert. Das Modell konnte eine accuracy von 95.33% erzielen.
Diesmal habe Ich den Random Forest Classifier angewendet, den Ich im Modell01 Versuch01 benutzt habe, wo er ein schlechtes Resultat geleitet hat. Beim diesem Versuch konnte er ein grossartiges Ergebnis erzielen, weil die Daten besser korreliert haben, die richtigen Features ausgewählt wurden und der geeignete Klassifikator benutzt wurde. Auch hier kann Ich kein Mean Square Error abfragen, da Ich 2 mögliche Zahlen habe 1 oder 2 also A oder B. Deshalb Macht es nicht Sinn diesen hier anzuwenden.
Allerdings habe Ich eine Grafik erstellt, die anzeigt wie viel Wichtigkeit die Variable hat. Siehe Bild 8



(Bild 09)

Modell03: Mein drittes Modell erstellt eine Lineare Regression, bei der die Beziehung zwischen dem Blutdruck und dem Alter erstellt wird.

Versuch01.

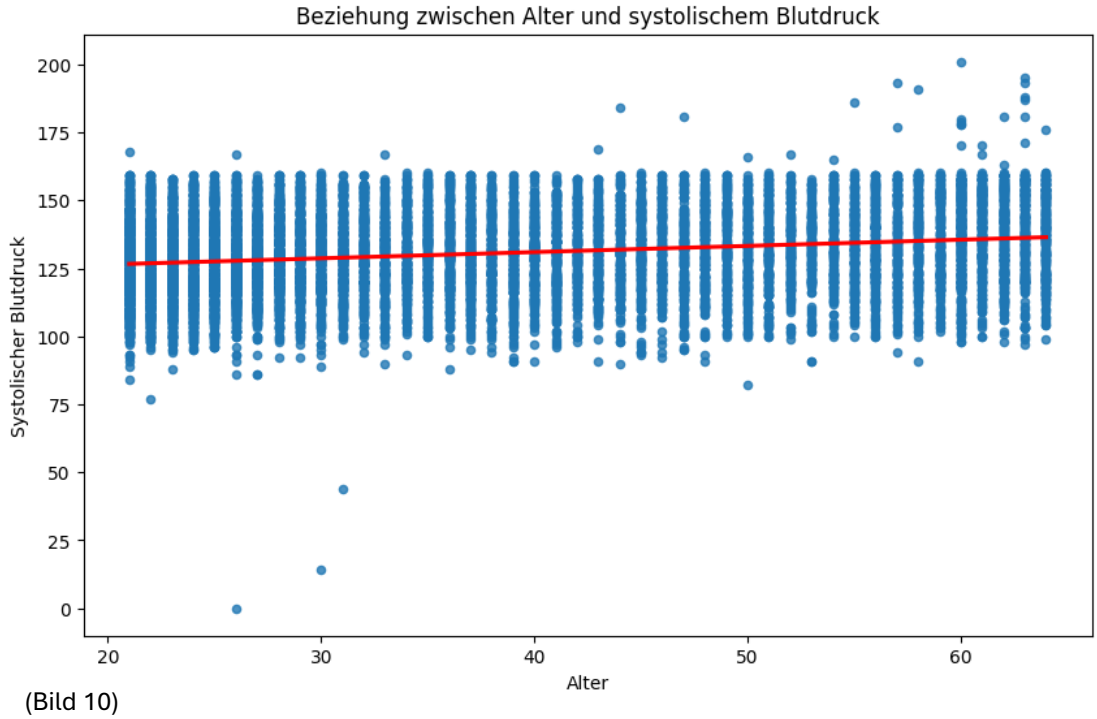
Lineares Regressionsmodell: Hier wird ein Modell erstellt, um vorherzusagen, wie der systolische Blutdruck auf Basis des Alters variiert. Der R^2-Score zeigt die Güte des Modells (wie gut es die Variabilität der Daten erklärt).

Scatterplot mit Regressionslinie: Ein Streudiagramm zeigt, wie sich der Blutdruck mit dem Alter ändert, und die Regressionslinie gibt eine Schätzung der linearen Beziehung.

Interpretation:

Ein positiver R^2-Wert (z.B. nahe 1) und eine deutlich steigende Regressionslinie im Plot würden zeigen, dass das Alter einen Einfluss auf den systolischen Blutdruck hat.

$R^2 = 0.04$



(Bild 10)

Zusammenfassung:

Resultate:

Ich bin sehr zufrieden mit meinem End Resultat, Ich konnte die EDA ausführlich beschreiben mit mehreren Grafiken und Texten.
Ich hatte keine fehlenden, doppelten oder falschen Daten.

Obwohl nicht alle Hypothesen richtig waren, konnte Ich sie einwandfrei mit Grafiken und den erstellten Modellen beantworten. Ich hatte immer das gleiche vorgehen bei der Untersuchung. Ich schaute zuerst wie sie korrelieren, danach analysierte Ich mit dem programmierten Modell wie die Variablen aufeinander korrelieren.

Die Modelle funktionierten ebenfalls nicht alle beim ersten Versuch, aber Ich habe mein Bestes gegeben diese zu Verbessern und das gelingte mir. Ich holte mir mehrere Male Hilfe auf Google, von Chat GPT oder von Kolleg*innen, dies finde Ich aber gerecht, da man lernen sollte die Mittel wie Chat GPT nutzen, und nicht runter zumachen.

Einsatzfähigkeit Modelle:

Das erste Modell funktioniert einwandfrei, hat aber eine tiefe accuracy.

Das zweite Modell funktioniert ebenfalls einwandfrei und hat eine stolze accuracy von 95.33%.

Das dritte und letzte Modell funktioniert ebenfalls, hat aber einen sehr tiefen R^2 -Wert, den Ich mit Sicherheit verbessern hätte können, Ich aber den Fokus auf den Rest gelegt habe, da das dritte Modell nicht obligatorisch war.

Mögliche Verbesserungen:

Ich hätte definitiv mehr Zeit in die Recherche und Analyse der Hypothesen stecken können, aber da fehlte mir die Zeit dazu.

Die Gestaltung des Codes und der Dokumentation hätte Ich auch noch verbessern können, da mir das eigentlich sehr am Herzen liegt und Ich realisiert habe, dass die Dokumentation in der Zukunft, zum Beispiel für einen Kunden sehr Wichtig ist, weil er den Code nicht versteht, aber ein Ergebnis sehen will.

Quellen:

Quelle 01: https://scikit-learn.org/dev/modules/generated/sklearn.ensemble.VotingClassifier.html

Quelle 02: https://aktiia.com/ch/normale-blutdruckwerte-je-nach-alter?srsltid=AfmBOopOR4BlcBKzfTgxPwAy1x68nHhVLO29jj71l786r1c8woJpVBM3