**Data Science with Python3**

De opdracht Python3 for Data Science betreft het schrijven van een Jupyter notebook voor inlezen, verwerken en weergeven van informatie uit een aangeleverd bestand. Het bestand dat gebruikt wordt bevat data van een stappenteller, die per minuut registreert hoeveel stappen er genomen zijn en wat dat betekent in termen van metabool equivalenten (METs).

De opdracht is opgedeeld in een aantal stukken. Afgezien van deel 1 zijn die in principe onafhankelijk van elkaar uit te werken en elke deelopdracht is opgesteld voor ongeveer twee uur werk, inclusief het raadplegen van materiaal verstrekt tijdens de cursus, het zoeken in de offline documentatie van Python3/Jupyter en/of het zoeken op internet, met name met Google en Stackoverflow. De verschillende deelopdrachten worden niet ineens verstrekt. De eerste twee deelopdrachten staan in dit document.

***Opdracht deel 1.***

Het eerste deel van de opdracht betreft het inlezen van de data. Bij een eerste naieve poging, die wellicht al ondernomen is, valt op dat de data niet direct ingelezen kan worden als pandas dataframe. Helaas is dit te vaak het geval bij echte data, zoals ook in dit bestand. Het probleem hier is dat decimale getallen geschreven zijn met een komma, volgens de Nederlandse regels, in plaats van met een punt, volgens Angelsaksisch gebruik, en dat deze waarden in een ‘comma-separated-value’ (csv) bestand zijn geschreven. De extra komma’s leiden tot een conflict in het bestandsformaat en dit moet eerst opgelost worden. Hiervoor kan gebruik gemaakt worden van de volgende code, die in Jupyter gebruikt kan worden om het bestand te lezen en een gecorrigeerd bestand te schrijven:

# Some of the fields in the csv have been written with commas

# That has to be corrected. The bad lines have a comma too many

badfile = open("export.csv")

fixedfile = open("fixed.csv", "w")

for line in badfile:

# Remove the trailing comma.

# It causes misreading of the column names.

line = line.rstrip()

line = line.rstrip(',')

if line.count(",") > 11:

# Fix the line. Read in and split in fields.

fields = line.split(',')

# The 6th/7th column contain the integer and fraction of a float

# Pop the fraction part out of the list and combine parts

fields[5] = ".".join([fields[5],fields.pop(6)])

print(",".join(fields), file=fixedfile)

else:

# Write the line to the outputfile as is

print(line, file=fixedfile)

# Done reading and writing - closing files

badfile.close()

fixedfile.close()

***Deelopdracht 1:*** *Zet een Jupyter Python3 notebook op dat gebruikt kan worden voor alle opdrachten met numpy, pandas en matplotlib. Voeg de hierboven gegeven code in en zorg dat het originele bestand ingelezen kan worden en gecorrigeerd wegeschreven kan. Lees dan het gecorrigeerde bestand in als Pandas dataframe. Zet de waardes van de kolom “M\_DATE” om naar* **datetime** *(pandas datum type) en verwijder dubbel geregistreerde tijden (drop duplicate times).*

***Deelopdracht 2:*** *Bereken het aantal stappen per dag, per week en per maand en laat deze zien als afzonderlijke lijn grafieken. Bereken daarnaast het gemiddelde per dag van de week en print de resultaten als staafdiagram. Beschrijf in het notebook wat de data en de grafieken laten zien.*

***Deelopdracht 3:*** *Om te kijken of er verschil is tussen de dagen van de week is het niet genoeg alleen naar de gemiddelden te kijken. Deze kunnen immers beinvloed worden door uitbijters. Geef de data per dag van de week weer als boxplots en als violinplots om inzicht te krijgen in de verdeling en beschrijf in het notebook wat de data en de grafieken laten zien, met name of er echt een verschil is tussen verschillende dagen.*

***Deelopdracht 4:*** *Bepaal met behulp van de MET waarde en de formule op* [*https://nl.wikipedia.org/wiki/MET-waarde*](https://nl.wikipedia.org/wiki/MET-waarde) *de cumulatieve hoeveelheid calorien die verbrand zijn in februari, in mei en in augustus. Toon de drie grafieken samen en kies onderscheidende kleuren en/of lijnen. Vergelijk de grafieken en beschrijf het verloop in het notebook.*

***Deelopdracht 5:*** *Bekijk met behulp van een scatterplot of er een relatie is tussen resultaten van opeenvolgende dagen. Let er daarbij op dat voor het aftrekken van dataframes of series geldt dat de indices bepalend zijn. Voor het verschil tussen dag* **a** *en dag* **a+1** *moeten de indices van de laatste serie dus worden aangepast.*

***Deelopdracht 6:*** *Er zou kunnen gelden dat als op een dag meer gelopen is, dat de volgende dag dan rustiger aan wordt gedaan. In dat geval zal het verschil, zoals bepaald bij* ***5****, alternerend positief en negatief zijn. Het product van die verschillen op opeenvolgende dagen is dan ook negatief. Bepaal dit product voor alle dagen en toon de verdeling van het resultaat als histogram. Kies een aantal bins dat een mooie verdeling toont en beschrijf in het notebook wat de grafiek laat zien en welke conclusie getrokken kan worden.*