# **Exploratory big data analysis with Python Pandas Saurabh P**

**This is my documentation for the exploratory data analysis that I will be conducting on the Kaggle   
dataset** [**The big dataset of ultra-marathon running (kaggle.com)**](https://www.kaggle.com/datasets/aiaiaidavid/the-big-dataset-of-ultra-marathon-running/discussion/420633)**. This dataset is a collection of   
information on the ultra-marathons that took place from 1798 to 2022. I will be conducting an initial  
exploratory analysis on the dataset to gain some insights.**   
  
  
  
  
  
**Importing libraries**

At first, I imported a couple of libraries namely, pandas and seaborn. Pandas is a library that is used for working with datasets. It has functions for analyzing, cleaning, exploring, and manipulating data. While seaborn is used for creating statistical graphs in Python. It builds on top of matplotlib and integrates closely with pandas data structure.

**Uploading the data set to Jupyter environment**

I downloaded the dataset from Kaggle.com and extracted the zip file to my desktop environment. Then, it was uploaded to the Jupyter environment for easy access.

**Checking the dataset and its datatypes**

The dataset has over 7 million records and 13 columns. Many of the formats of the columns are in the European format which will be dealt with later.

**Cleaning the data**

My first requirement was to filter out the USA races with 50km or 50mi. Used the following query to get this output - df[df['Event distance/length']=='50km']. This is what a part of the output looks like-

A table with text and numbers

Description automatically generated

Then I checked for the columns with miles as the distance criteria.   
The output looks something like this-  
A screenshot of a computer

Description automatically generated

Next, I combined the data for 50km values and 50mi values.   
The ‘isin’ command was used to merge these data values. The output looks something like this-   
A screenshot of a computer

Description automatically generated  
Next, I wanted to filter the data only for the year of 2020. The output -  
A screenshot of a computer

Description automatically generated

This brought us down to 63000+ rows from a whopping 7million.

Next, I combined all the filters which were- USA RACES with 50km or 50mi in the year 2020. This is what the output looks like-   
A screenshot of a computer

Description automatically generated  
  
This brought us down to a manageable 26000 rows.

Next, I want to remove the tag ‘USA’ from the event name as the whole of the dataset is not USA specific. This will improve the overall look of the column and won’t look messy. This is what it looks like-  
  
A screenshot of a computer

Description automatically generated

Next, I cleaned up the athlete age to something very accurate. I subtracted the athlete birth year from the year of race to get their age.   
  
I also removed the ‘h’ from athlete performance to make the column look cleaner.  
  
  
Next I dropped some columns that wouldn’t be that uselful.   
  
My final dataset looks a lot cleaner now with all the columns that are important. Here is a snipped of the dataset after cleaning-   
  
A white table with black text

Description automatically generated  
As you can see, it is a lot more readable now.

**Cleaning up the null values**  
  
Looking at the overview of the data it tells us that athlete\_age has over 200 null values.   
A screenshot of a computer

Description automatically generated  
  
I dropped these null values and it left me with a total of 25857 values.   
  
There were no duplicate values when I check for them.  
  
**Fixing the datatypes**  
  
A screenshot of a computer

Description automatically generated  
  
Changed values:   
athlete age to integer  
athlete average speed to float  
  
  
  
**Renaming and Reordering Columns**  
  
I wanted to rename the columns as it makes data viewing easier. This also makes manual coding easier as I will be eliminating all the uppercase values and turning them into lowercase. I will also rename the event lists to something more specific. I will also reorder the columns to a sequence that I like as I am the one who is working on the dataset. I would suggest new users to also do this as it will improve your analyzing speed.  
  
A screenshot of a computer

Description automatically generated  
  
This is what renaming and reordering columns will give you.

**Extracting specific race data**   
  
So my friend Ryan, he participated in a race named “everglade”. I will query a line to get some details on his run. He told me that his performance time was 9:19. So I looked for the time stamp and got to know that his id was ‘222509’.  
  
A screenshot of a computer

Description automatically generated  
  
These are the two races that my friend Ryan participated in.   
  
  
  
  
**Charts and Graphs**

-I want to visually see the difference between the number of 50k and 50mi race  
A screen shot of a graph

Description automatically generated

-Next, I wanted to see the gender ratio in these races  
A graph of a running race

Description automatically generated with medium confidence

-Highest average speed among the participants  
A graph of a number of blue and black bars

Description automatically generated

-Comparison of average speeds in 50k and 50mi between males and females  
A diagram of different colored shapes

Description automatically generated with medium confidence

-Distribution between age and average speed  
A blue dot diagram with numbers

Description automatically generated with medium confidence

-Layers the above plot with athlete\_gender  
A diagram of a number of dots

Description automatically generated with medium confidence

# My research questions Q1) Difference in speed for the 50k, 50mi between males and females A white background with black text Description automatically generated This groupby query shows us that females have a higher mean time in the 50km races while men have the higher mean time in the 50mi races. This could mean that females have higher endurance for longer runs. Q2) What age groups are the best in the 50mi race (20+ races min) A table with numbers and a few digits Description automatically generated People aged 29 have the best times in a 50mi race while younger participants are shown to perform poorer.

# Q3) What age groups are the worst in 50m race A table of numbers and a few digits Description automatically generated with medium confidence Obvious as it is, elderly participants perform the worst in a 50mi race. These elderly people should motivate you guys to get active as their age is not stopping them from participating in ultra races. Q4) Seasonal information- Are the participants slower in summer or in winter? Using the lambda functionality I categorized the dates into seasons, namely, ‘Winter’, ‘Summer’, ‘Fall’, ‘Spring’. A screenshot of a computer Description automatically generated A final groupby query shows us that Summers being hot are the toughest to do a ultra run in while winters are the easiest to compete in while spring and fall being the next easy seasons.

# 

Q1) Difference in speed for the 50k, 50mi between males and females  
  
A white background with black text

Description automatically generated  
  
  
This groupby query shows us the mean times of males and females. Females have a higher mean time in the 50km races while males have the higher mean time in the 50mi races.