

calculate_speed_per_pedestrian

March 18, 2020

```
In [1]: # For each pedestrian, calculate his / her speed at each time step
# by using pandas' groupby and apply functionality. See:
# - https://towardsdatascience.com/how-to-use-the-split-apply-combine-strategy-in-pandas/
# - https://pandas.pydata.org/pandas-docs/stable/user\_guide/groupby.html

import numpy as np
import pandas as pd

# "pedId" 1 walks from (0,0) to (1,1) and "pedId" walks from (1,1) to(3,3).
# Resulting speeds: "pedId" 1: 1.42 m/s, "pedId" 2: 2.83
# Goal: Insert "speed" column containing these values for each "time" (in first row sp
test_data = pd.DataFrame(
    {
        "time": [1, 1, 2, 2],
        "pedId": [1, 2, 1, 2],
        "x": [0, 1, 1, 3],
        "y": [0, 1, 1, 3]
    }
)

test_data.set_index(["time", "pedId"], inplace=True)

test_data
```

Out[1]:

		x	y
time	pedId		
1	1	0	0
	2	1	1
2	1	1	1
	2	3	3

```
In [10]: def calculate_speed_for_pedestrian(data_frame, delta_time_in_second=1):
    current_positions = data_frame.loc[:, ["x", "y"]]
    old_positions = data_frame.shift().loc[:, ["x", "y"]]

    traveled_distances_in_meter = np.linalg.norm(old_positions - current_positions, a
```

```

speed_in_meter_per_second = traveled_distances_in_meter / delta_time_in_second
speed_in_kilometer_per_hour = speed_in_meter_per_second * 3600.0 / 1000.0

speed_data_frame = pd.DataFrame(speed_in_meter_per_second, columns=["Speed [m/s]"])

old_and_new_data_frame = pd.concat([data_frame.reset_index(), speed_data_frame.reset_index()], axis=1)
old_and_new_data_frame.set_index(["time", "pedId"], inplace=True)

return old_and_new_data_frame

test_data.groupby(["pedId"]).apply(func=calculate_speed_for_pedestrian)

```

Out[10]:

		x	y	index	Speed [m/s]
1	1	0	0	0	NaN
	2	1	1	0	NaN
2	1	1	1	1	1.414214
	2	3	3	1	2.828427