

Import relevant packages here.

```
In [1]: import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import math
```

Load the data and verify it is loaded correctly.

- Print it (head, tail, or specific rows, choose a sensible number of rows).
- Compare it to the source file.

```
In [2]: data = pd.read_csv('cf_data.csv', delimiter = ',')
data.head()
```

```
Out[2]:
```

	dv	s	a
0	-0.743240	53.5427	1.242570
1	-0.557230	53.6120	1.777920
2	-0.454769	53.6541	0.544107
3	-0.525396	53.7030	-0.294755
4	-0.601285	53.7592	-0.290961

In the ensuing, you will use `numpy` .

Let's create a grid for the values to plot. But first create **two arrays named `dv` and `s`** using `numpy.linspace` that hold the grid values at the relevant indices in their respective dimension of the grid.

Create a **grid named `a`** with zeros using `numpy.zeros` in to which calculated acceleration values can be stored.

Let the grid span:

- Speed difference `dv` [m/s]
 - From -10 till 10
 - With 41 evenly spaced values
- Headway `s` [m]
 - From 0 till 200
 - With 21 evenly spaced values

```
In [3]: dv = np.linspace(-10, 10, 41)
s = np.linspace(0, 200, 21)
a = np.zeros([len(s), len(dv)])
```

Create from the imported data 3 separate `numpy` arrays for each column `dv`, `s` and `a`.
(We do this for speed reasons later.)

- Make sure to name them differently from the arrays that belong to the grid as above.
- You can access the data of each column in a `DataFrame` using `data.xxx` where `xxx` is the column name (not as a string).
- Use the method `to_numpy()` to convert a column to a `numpy` array.

```
In [12]: testdata = data

DV = testdata.dv.to_numpy()
S = testdata.s.to_numpy()
A = testdata.a.to_numpy()

# print(np.exp(-1))
```

Create an algorithm that calculates all the acceleration values and stores them in the grid. The algorithm is described visually in the last part of the lecture. At each grid point, it calculates a weighted mean of all measurements. The weights are given by an exponential function, based on the 'distance' between the grid point, and the measurement values of `dv` and `s`. To get you started, how many `for`-loops do you need?

For this you will need `math`.

Use an *upsilon* of 1.5m/s and a *sigma* of 30m.

Warning: This calculation may take some time. So:

- Print a line for each iteration of the outer-most `for`-loop that shows you the progress.
- Test you code by running it only on the first 50 measurements of the data.

```
In [13]: epsilon = 1.5
sigma = 30

for x in range(len(dv)):
    for y in range(len(s)):
        sum_weightA = 0
        sum_weight = 0

        for i in range(len(DV)):
            weight = math.exp(-np.abs(DV[i] - dv[x]) / epsilon - np.abs(S[i] - s[y]) / sigma)
            sum_weightA += (weight * A[i])
            sum_weight += weight
        a[y, x] = (sum_weightA / sum_weight)

print(a)
```

```

[[ 5.28790291e-01  5.21972961e-01  5.11960826e-01  4.94430672e-01
  4.74897272e-01  4.56499296e-01  4.38535446e-01  4.23337077e-01
  4.10878222e-01  3.97711606e-01  3.80417405e-01  3.59286888e-01
  3.25705455e-01  2.86877694e-01  2.46327194e-01  2.01001412e-01
  1.48574715e-01  9.01239429e-02  2.70005472e-02 -4.12475842e-02
 -1.15625929e-01 -1.98795616e-01 -2.81304508e-01 -3.57551383e-01
 -4.25937535e-01 -4.85652840e-01 -5.36888322e-01 -5.83529810e-01
 -6.22582423e-01 -6.47975250e-01 -6.67865190e-01 -6.89337871e-01
 -7.04808573e-01 -7.14372459e-01 -7.24796010e-01 -7.33451923e-01
 -7.38391387e-01 -7.42123309e-01 -7.45591592e-01 -7.54176212e-01
 -7.62999274e-01]
[ 5.31528375e-01  5.24724137e-01  5.14736054e-01  4.97226381e-01
  4.77723834e-01  4.59354095e-01  4.41414044e-01  4.26240891e-01
  4.13814337e-01  4.00683995e-01  3.83421707e-01  3.62341903e-01
  3.28812819e-01  2.90055944e-01  2.49390124e-01  2.03926295e-01
  1.51338736e-01  9.27145228e-02  2.95021951e-02 -3.88871108e-02
 -1.13501686e-01 -1.96865417e-01 -2.79812929e-01 -3.56454100e-01
 -4.25186307e-01 -4.85182020e-01 -5.36720439e-01 -5.83743083e-01
 -6.23159670e-01 -6.48692574e-01 -6.68618550e-01 -6.90131842e-01
 -7.05625583e-01 -7.15197056e-01 -7.25631342e-01 -7.34293106e-01
 -7.39231356e-01 -7.42962430e-01 -7.46427880e-01 -7.55025139e-01
 -7.63859333e-01]
[ 5.61006512e-01  5.54163640e-01  5.44165334e-01  5.26371090e-01
  5.06607117e-01  4.87946347e-01  4.69646848e-01  4.54194324e-01
  4.41636026e-01  4.28362069e-01  4.10727011e-01  3.89415462e-01
  3.56098122e-01  3.17829903e-01  2.76990574e-01  2.31521810e-01
  1.79235481e-01  1.20867267e-01  5.79003306e-02 -1.08622191e-02
 -8.65888130e-02 -1.71077352e-01 -2.55835749e-01 -3.35150975e-01
 -4.07134718e-01 -4.70376122e-01 -5.25184346e-01 -5.75676491e-01
 -6.18563292e-01 -6.46521388e-01 -6.68272576e-01 -6.91557781e-01
 -7.08299669e-01 -7.18787220e-01 -7.30197023e-01 -7.39463177e-01
 -7.44614304e-01 -7.48501176e-01 -7.52092529e-01 -7.61133394e-01
 -7.70398834e-01]
[ 6.19021526e-01  6.11680814e-01  6.01027558e-01  5.81299254e-01
  5.59371526e-01  5.38536032e-01  5.17619570e-01  4.99252591e-01
  4.84020668e-01  4.68386259e-01  4.48184934e-01  4.24334382e-01
  3.89923428e-01  3.51386732e-01  3.10065052e-01  2.64862857e-01
  2.12885431e-01  1.54286047e-01  9.15774361e-02  2.33135028e-02
 -5.11250585e-02 -1.34244544e-01 -2.17821194e-01 -2.96599983e-01
 -3.69486976e-01 -4.35594361e-01 -4.94509612e-01 -5.50105366e-01
 -5.98847380e-01 -6.32298811e-01 -6.57831313e-01 -6.84001679e-01
 -7.03884010e-01 -7.16923408e-01 -7.31035702e-01 -7.42200264e-01
 -7.48322304e-01 -7.52917312e-01 -7.57125635e-01 -7.67818166e-01
 -7.78703311e-01]
[ 6.55960288e-01  6.51354569e-01  6.41578229e-01  6.19852331e-01
  5.96012248e-01  5.74003424e-01  5.50867519e-01  5.29192201e-01
  5.11039302e-01  4.93142283e-01  4.70276267e-01  4.43596977e-01
  4.08983140e-01  3.71690646e-01  3.31289656e-01  2.87067557e-01
  2.35764958e-01  1.77783602e-01  1.15316400e-01  4.79812868e-02
 -2.40120493e-02 -1.04437467e-01 -1.85250155e-01 -2.61294655e-01
 -3.32490147e-01 -3.99524767e-01 -4.61812685e-01 -5.20868195e-01
 -5.73828741e-01 -6.12145012e-01 -6.42690081e-01 -6.74109729e-01
 -6.98659750e-01 -7.14132077e-01 -7.30931766e-01 -7.44822430e-01
 -7.53028389e-01 -7.59032519e-01 -7.64444002e-01 -7.78349902e-01
 -7.92327524e-01]
[ 6.65396701e-01  6.63377584e-01  6.55595132e-01  6.35736258e-01

```

6.11661065e-01	5.89217220e-01	5.64973598e-01	5.41149811e-01
5.21294742e-01	5.01733085e-01	4.76997661e-01	4.47960334e-01
4.12862866e-01	3.76930454e-01	3.37614684e-01	2.94468942e-01
2.44679147e-01	1.89003120e-01	1.28678643e-01	6.35654668e-02
-5.13664941e-03	-8.16847311e-02	-1.58929523e-01	-2.31555135e-01
-3.00293229e-01	-3.66815712e-01	-4.29624890e-01	-4.90310850e-01
-5.46411895e-01	-5.88067196e-01	-6.22481911e-01	-6.57920963e-01
-6.87673698e-01	-7.08033703e-01	-7.29163866e-01	-7.47850866e-01
-7.59570936e-01	-7.67193621e-01	-7.73265314e-01	-7.90066190e-01
-8.08157999e-01]			
[6.43229605e-01	6.43996170e-01	6.40748699e-01	6.28556458e-01
6.12343424e-01	5.96276165e-01	5.75667741e-01	5.52974836e-01
5.33850313e-01	5.14215791e-01	4.87937277e-01	4.55233449e-01
4.18174258e-01	3.82292910e-01	3.42847752e-01	3.00085683e-01
2.51346926e-01	1.96668873e-01	1.37507801e-01	7.37773516e-02
7.78104554e-03	-6.59697763e-02	-1.40191807e-01	-2.09646533e-01
-2.75616866e-01	-3.40089938e-01	-4.00380406e-01	-4.59467264e-01
-5.15214641e-01	-5.57074256e-01	-5.92376762e-01	-6.28726049e-01
-6.61115485e-01	-6.84673980e-01	-7.09938086e-01	-7.34773269e-01
-7.51681385e-01	-7.62319871e-01	-7.68993270e-01	-7.89787042e-01
-8.14096561e-01]			
[6.10552631e-01	6.14932771e-01	6.16949839e-01	6.12810614e-01
6.05658891e-01	5.97555851e-01	5.83397715e-01	5.64148086e-01
5.46067409e-01	5.25481488e-01	4.97544128e-01	4.63281495e-01
4.25657929e-01	3.90399761e-01	3.50651157e-01	3.07905546e-01
2.60267564e-01	2.06631783e-01	1.48373630e-01	8.58557793e-02
2.20807942e-02	-4.97248568e-02	-1.21706551e-01	-1.88957989e-01
-2.52340412e-01	-3.14258316e-01	-3.70652031e-01	-4.26252363e-01
-4.78561004e-01	-5.16150265e-01	-5.48359948e-01	-5.83336575e-01
-6.15941412e-01	-6.41540248e-01	-6.69221482e-01	-6.96752524e-01
-7.14896095e-01	-7.27337334e-01	-7.36817183e-01	-7.66720915e-01
-8.02012538e-01]			
[5.47471154e-01	5.56325997e-01	5.64307859e-01	5.68965431e-01
5.71724758e-01	5.72163728e-01	5.64557784e-01	5.49181023e-01
5.33833959e-01	5.15327247e-01	4.88271844e-01	4.55443802e-01
4.19663236e-01	3.87178294e-01	3.48737723e-01	3.07356834e-01
2.61739682e-01	2.09856995e-01	1.52709741e-01	9.19230492e-02
3.06037888e-02	-3.83952145e-02	-1.07241670e-01	-1.71518145e-01
-2.31405115e-01	-2.89926108e-01	-3.42356553e-01	-3.94578397e-01
-4.43910602e-01	-4.77935773e-01	-5.08396662e-01	-5.43612549e-01
-5.76054016e-01	-6.02490410e-01	-6.29061971e-01	-6.53267597e-01
-6.64051877e-01	-6.71754193e-01	-6.79719051e-01	-7.09400974e-01
-7.47428800e-01]			
[4.81963497e-01	4.96857227e-01	5.12603502e-01	5.27038449e-01
5.36993457e-01	5.43417987e-01	5.41127744e-01	5.27281735e-01
5.12931150e-01	4.95595815e-01	4.70963281e-01	4.41125442e-01
4.06806754e-01	3.76498646e-01	3.40237469e-01	3.00608549e-01
2.56957510e-01	2.07248864e-01	1.52272420e-01	9.41827764e-02
3.60673170e-02	-2.95533010e-02	-9.52182514e-02	-1.56338076e-01
-2.12543504e-01	-2.67279421e-01	-3.15293937e-01	-3.63519791e-01
-4.08964778e-01	-4.38194970e-01	-4.64679961e-01	-4.96653334e-01
-5.26350336e-01	-5.52100992e-01	-5.78354752e-01	-6.03916279e-01
-6.12768101e-01	-6.15989584e-01	-6.18582297e-01	-6.37095166e-01
-6.61623304e-01]			
[4.05246338e-01	4.27677900e-01	4.52643086e-01	4.77858210e-01
4.94701034e-01	5.07384017e-01	5.12281245e-01	5.01730313e-01

4.89717933e-01	4.73614919e-01	4.51614995e-01	4.24977122e-01
3.92714771e-01	3.64232278e-01	3.30380035e-01	2.93174956e-01
2.52315023e-01	2.05919253e-01	1.53816145e-01	9.93062914e-02
4.43098761e-02	-1.82383769e-02	-8.15109289e-02	-1.40497790e-01
-1.93810847e-01	-2.44795551e-01	-2.87912417e-01	-3.31432583e-01
-3.72501509e-01	-3.96967643e-01	-4.20139289e-01	-4.50366688e-01
-4.80117961e-01	-5.07215760e-01	-5.35507747e-01	-5.66999755e-01
-5.77680201e-01	-5.78029451e-01	-5.74028959e-01	-5.83611813e-01
-6.02072025e-01]			
[3.25398619e-01	3.57066542e-01	3.93338794e-01	4.30951300e-01
4.57642083e-01	4.77239615e-01	4.86643199e-01	4.79831288e-01
4.70766044e-01	4.56713789e-01	4.36703740e-01	4.10555614e-01
3.78879894e-01	3.51445897e-01	3.18751596e-01	2.83785564e-01
2.46446393e-01	2.04500603e-01	1.55497671e-01	1.04333017e-01
5.27251495e-02	-6.43878729e-03	-6.73368670e-02	-1.24834402e-01
-1.76189645e-01	-2.24703230e-01	-2.64234896e-01	-3.05077625e-01
-3.44411360e-01	-3.67250230e-01	-3.88067762e-01	-4.15955519e-01
-4.44865434e-01	-4.71937681e-01	-5.00619796e-01	-5.36608564e-01
-5.52190139e-01	-5.54976013e-01	-5.52556250e-01	-5.64111230e-01
-5.83157103e-01]			
[2.55238525e-01	3.00236916e-01	3.46952398e-01	3.90159589e-01
4.22387397e-01	4.47585461e-01	4.60508486e-01	4.57403612e-01
4.51921663e-01	4.40947723e-01	4.23682576e-01	3.97741219e-01
3.67270642e-01	3.41582601e-01	3.08495324e-01	2.74514632e-01
2.39658663e-01	2.00445462e-01	1.53112150e-01	1.03825397e-01
5.39918293e-02	-2.98508273e-03	-6.17998176e-02	-1.17842578e-01
-1.67024063e-01	-2.13266080e-01	-2.49630301e-01	-2.88756605e-01
-3.27865253e-01	-3.50905396e-01	-3.71033632e-01	-3.98434536e-01
-4.28753238e-01	-4.57164999e-01	-4.85742815e-01	-5.24670092e-01
-5.42447204e-01	-5.47734124e-01	-5.47958847e-01	-5.61363396e-01
-5.79701236e-01]			
[1.80344447e-01	2.36151577e-01	2.94512008e-01	3.42209526e-01
3.78888193e-01	4.09422642e-01	4.25217999e-01	4.25868036e-01
4.24832373e-01	4.17886722e-01	4.04172958e-01	3.80444942e-01
3.52284991e-01	3.28592838e-01	2.95908049e-01	2.63096374e-01
2.30472560e-01	1.93669916e-01	1.48327584e-01	1.01364312e-01
5.37755663e-02	-3.55421434e-04	-5.65866423e-02	-1.11194000e-01
-1.58879102e-01	-2.04333188e-01	-2.39554030e-01	-2.78058056e-01
-3.16945364e-01	-3.40114812e-01	-3.58538953e-01	-3.85118967e-01
-4.17143728e-01	-4.45577070e-01	-4.73798393e-01	-5.13261113e-01
-5.33408749e-01	-5.42760022e-01	-5.46997592e-01	-5.63687725e-01
-5.80864351e-01]			
[1.25921300e-01	1.85160487e-01	2.44396189e-01	2.92149148e-01
3.33071863e-01	3.65996324e-01	3.80268685e-01	3.81905312e-01
3.84596681e-01	3.81804447e-01	3.72151064e-01	3.52388250e-01
3.27700005e-01	3.07976833e-01	2.79407258e-01	2.49264008e-01
2.18795927e-01	1.84240180e-01	1.40528751e-01	9.48796134e-02
4.82983026e-02	-4.64681262e-03	-5.95468653e-02	-1.13358806e-01
-1.60644324e-01	-2.05544031e-01	-2.40674536e-01	-2.79329526e-01
-3.18741637e-01	-3.42633954e-01	-3.59473318e-01	-3.85668241e-01
-4.19054672e-01	-4.47768445e-01	-4.75480616e-01	-5.15346971e-01
-5.38520327e-01	-5.53848637e-01	-5.63575112e-01	-5.84731654e-01
-5.98225652e-01]			
[4.12265476e-02	1.05304663e-01	1.67260942e-01	2.17672692e-01
2.64224070e-01	3.00626169e-01	3.15024933e-01	3.16570472e-01
3.22760059e-01	3.23851296e-01	3.18688862e-01	3.04234132e-01

2.84601900e-01	2.71054236e-01	2.48849176e-01	2.22848042e-01
1.96157262e-01	1.65530210e-01	1.24544795e-01	8.06587080e-02
3.52647266e-02	-1.61216789e-02	-6.91170862e-02	-1.21877255e-01
-1.68525901e-01	-2.12875746e-01	-2.48232970e-01	-2.87515077e-01
-3.27634435e-01	-3.52670856e-01	-3.69419578e-01	-3.94444903e-01
-4.26264938e-01	-4.53929784e-01	-4.81539556e-01	-5.23158208e-01
-5.51979309e-01	-5.77674607e-01	-5.96200694e-01	-6.23712832e-01
-6.29207861e-01]			
[-6.12275409e-02	5.62352870e-03	7.06731526e-02	1.26282402e-01
1.82400393e-01	2.25654853e-01	2.39892901e-01	2.37980294e-01
2.45025619e-01	2.44220038e-01	2.39769797e-01	2.29606653e-01
2.15498722e-01	2.10002418e-01	1.96427606e-01	1.76005322e-01
1.55243606e-01	1.31156128e-01	9.50220855e-02	5.43825394e-02
1.12624040e-02	-3.72003512e-02	-8.66960718e-02	-1.37534443e-01
-1.82998932e-01	-2.26287905e-01	-2.61906280e-01	-3.02059065e-01
-3.43033359e-01	-3.69819539e-01	-3.86595401e-01	-4.09637837e-01
-4.38694305e-01	-4.64477325e-01	-4.91876699e-01	-5.36509414e-01
-5.74745072e-01	-6.16993514e-01	-6.48122981e-01	-6.82184432e-01
-6.71915356e-01]			
[-1.70910239e-01	-1.06619058e-01	-4.11453261e-02	1.96101766e-02
8.67671593e-02	1.37518805e-01	1.46754730e-01	1.35663238e-01
1.40640478e-01	1.41882757e-01	1.44244532e-01	1.44548316e-01
1.42515161e-01	1.52719939e-01	1.54250232e-01	1.45163217e-01
1.33629588e-01	1.16123653e-01	8.39403020e-02	4.51286354e-02
1.93416931e-03	-4.51241608e-02	-9.23750974e-02	-1.41878504e-01
-1.87158085e-01	-2.30852695e-01	-2.67931705e-01	-3.09860592e-01
-3.51671223e-01	-3.80140489e-01	-3.97767281e-01	-4.20429199e-01
-4.48846420e-01	-4.75186868e-01	-5.05659187e-01	-5.58098338e-01
-6.11939911e-01	-6.78203969e-01	-7.24024636e-01	-7.60402098e-01
-7.22984524e-01]			
[-2.55253038e-01	-1.98458643e-01	-1.31973440e-01	-6.53209197e-02
2.04170807e-02	9.05197106e-02	1.05302297e-01	9.25522925e-02
9.40801581e-02	9.46340013e-02	9.80059329e-02	1.04296627e-01
1.09349009e-01	1.25894712e-01	1.33646523e-01	1.29171470e-01
1.20890459e-01	1.05375858e-01	7.45713054e-02	3.65403389e-02
-7.44870992e-03	-5.34146403e-02	-9.80421342e-02	-1.45925647e-01
-1.90717479e-01	-2.34880443e-01	-2.73568451e-01	-3.16980349e-01
-3.57512293e-01	-3.85382693e-01	-4.02763144e-01	-4.25255616e-01
-4.53718354e-01	-4.79855749e-01	-5.09927448e-01	-5.61410333e-01
-6.06956265e-01	-6.51952274e-01	-6.79295195e-01	-6.97467202e-01
-6.49940779e-01]			
[-3.00273997e-01	-2.46400534e-01	-1.75539140e-01	-1.03415975e-01
-1.40316237e-02	5.99180386e-02	7.85314901e-02	7.00883594e-02
6.89803671e-02	6.40602671e-02	6.56195784e-02	7.16170279e-02
7.77922295e-02	9.74152565e-02	1.11509874e-01	1.11871785e-01
1.08243459e-01	9.67236287e-02	6.99241354e-02	3.62823216e-02
-4.23324367e-03	-4.91716500e-02	-9.36945694e-02	-1.41958449e-01
-1.88194770e-01	-2.35043318e-01	-2.78068482e-01	-3.25231572e-01
-3.64202984e-01	-3.91513284e-01	-4.08771296e-01	-4.31129239e-01
-4.59686382e-01	-4.85217099e-01	-5.14229070e-01	-5.64276614e-01
-5.97560498e-01	-6.14336758e-01	-6.23080269e-01	-6.32940399e-01
-5.87557930e-01]			
[-2.35765885e-01	-1.82604025e-01	-1.31165031e-01	-7.90071955e-02
-9.77132845e-03	5.13080195e-02	6.55588433e-02	5.84794772e-02
5.13025459e-02	3.82144618e-02	3.36368704e-02	3.78868441e-02
4.44898826e-02	6.87174380e-02	8.96043837e-02	9.37566033e-02

```

9.36910526e-02  8.60204249e-02  6.39464898e-02  3.64689908e-02
1.02050595e-03 -4.23830235e-02 -8.62851111e-02 -1.34410793e-01
-1.81934244e-01 -2.31603680e-01 -2.79216075e-01 -3.30326775e-01
-3.68124057e-01 -3.95814904e-01 -4.14058243e-01 -4.37393829e-01
-4.67489195e-01 -4.93634351e-01 -5.23258604e-01 -5.74889279e-01
-5.95710430e-01 -5.82963641e-01 -5.81207817e-01 -5.98363866e-01
-5.68579803e-01]]

```

The following code will plot the data for you. Does it make sense when considering:

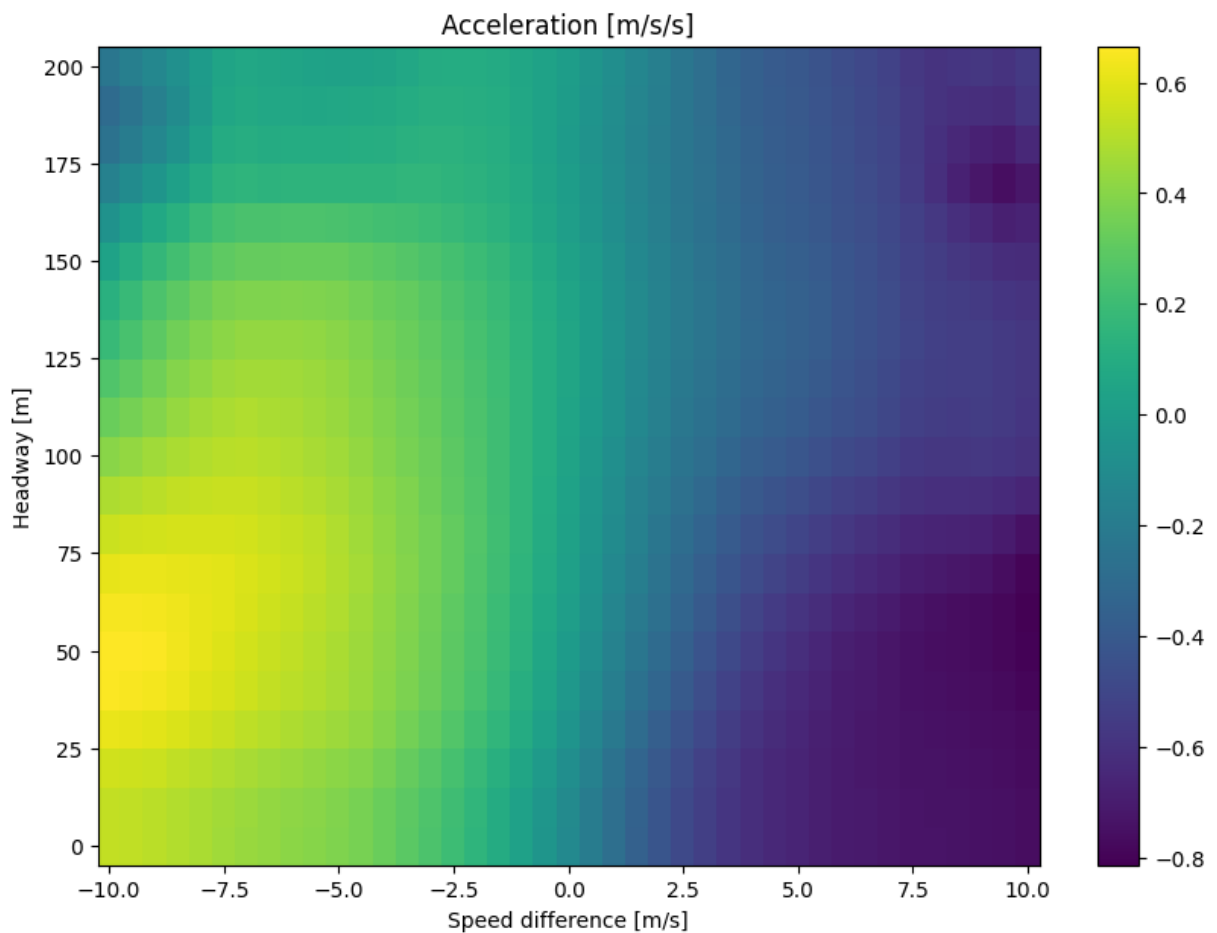
- Negative (slower than leader) and positive (faster than leader) speed differences?
- Small and large headways?

```

In [14]: X, Y = np.meshgrid(dv, s)
print(X.shape)
axs = plt.axes()
p = axs.pcolor(X, Y, a, shading='nearest')
axs.set_title('Acceleration [m/s/s]')
axs.set_xlabel('Speed difference [m/s]')
axs.set_ylabel('Headway [m]')
axs.figure.colorbar(p);
axs.figure.set_size_inches(10, 7)

```

(21, 41)



In []: