DE plotting tools

I write a class to simplify DE data visualization. Overview of the current code:

```
class de_data():
    """Double emulsion data plotting tool."""
    def __init__(self, data):
    def __repr__(self):
    def show(self):
    def parameter_space(self, highlight_Chile_data=True):
        """D vs. d, with color coded OD"""
    def generate_msd_repo(self):
        """Generate .jpg images for MSD repo. Takes ~2 min and save images, be careful!"""
    def scatter_0(self, mode="log", highlight_Chile_data=True):
        """Plot tau^* vs. (D-d)/d^2"""
    def look_for_missing_traj(self, traj_folder, fmt="{:02d}.csv"):
        """Check the existence of trajectory data file in given folder, according to the log"""
    def plot_MSD_model_Cristian():
        """plot the MSD model, get a feeling of parameters, save for future use, not finished""
    def plot_0(self, nbins=5, overlap=0, mode="log"):
        """tau vs. (D-d)/d^2, with average"""
    def scatter_1(self, mode="log", highlight_Chile_data=True):
        """R_inf vs. (D-d)/d^2"""
    def plot_1(self, nbins=5, overlap=0, mode="log"):
        """R_inf vs. (D-d)/d^2, with average"""
    def Rinf2_tau(self):
        """Plot $R_\infty^2$ vs. $\tau^*$"""
    def Rinf2_over_tau(self):
        """Plot $R_\infty^2 / \tau^*$ vs. $(D-d)/d^2$"""
    def rescale_Rinf_OD(self):
        """Plot Rinf/OD vs. (D-d)/d^2"""
    def rescale_Rinf_freespace(self):
        """rescale Rinf with (D-d)"""
    def scatter(self, mode="log", highlight_Chile_data=True):
        """I want to implement a more flexible plotting tool to test ideas, but it seems diffic
```

This class interfaces with the main log spreadsheet, and plot the data there using predefined style. In the following, I will demonstrate how to use the tools.

Create a `de_data` object

```
from de_utils import de_data
import pandas as pd

log_dir = r"..\Data\structured_log_DE.ods"
log = pd.read_excel(io=log_dir, sheet_name="main")
data = de_data(log)
```

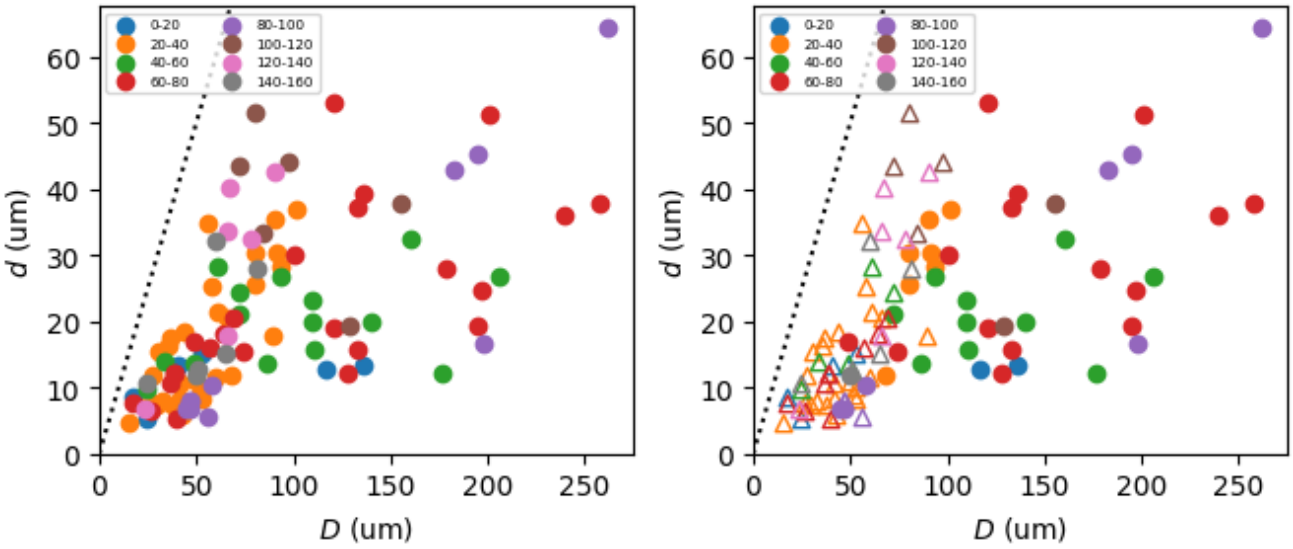
This creates a `de_data` object called `data`. Next, we are going to use `data` to make the various visualizations.

Overview parameters

To see the distribution of the parameters D, d and OD, simply use the command `data.parameter_space()`. In the current data set, I combine the data obtained both in Paris and in Chile. To discern the data based on sources, simply pass an additional keyword `highlight_Chile_data=True`:

```
data.parameter_space() # left
data.parameter_space(highlight_Chile_data=True) # right
```

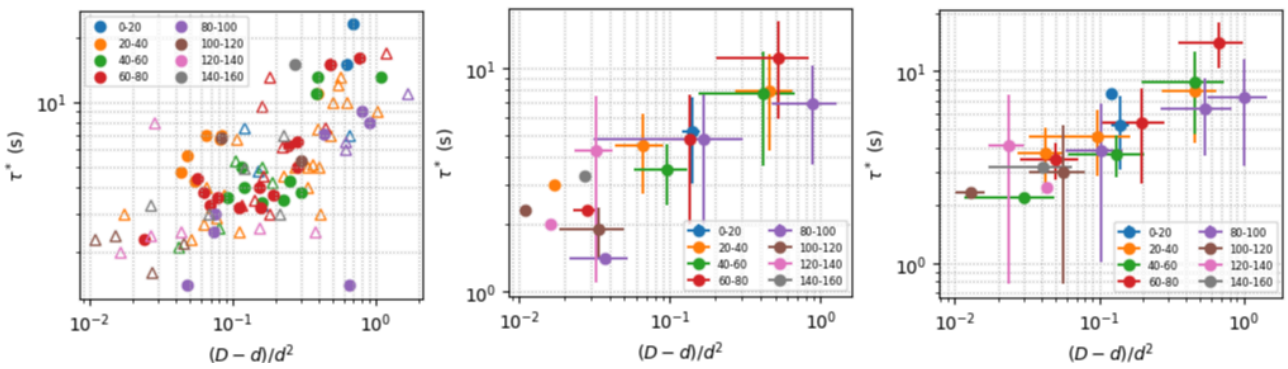
This will result in the scattered plot with two different symbols on the right.



Binning the scattered points

In the previous section, I demonstrated the first useful tool: `highlight_Chile_data`. Here, I will show another one: binning the scattered points. This is implemented in the `plot_0` and `plot_1` functions. By specifying the number of bins `nbins` and overlap between bins `overlap`, a plot with binned data is generated. As shown below, the scattered plot (left) can be binned to give the plot in the middle by setting `nbins=5, overlap=0`. If we adjust the parameters, for example, `nbins=6, overlap=0.5`, we will get the plot on the right.

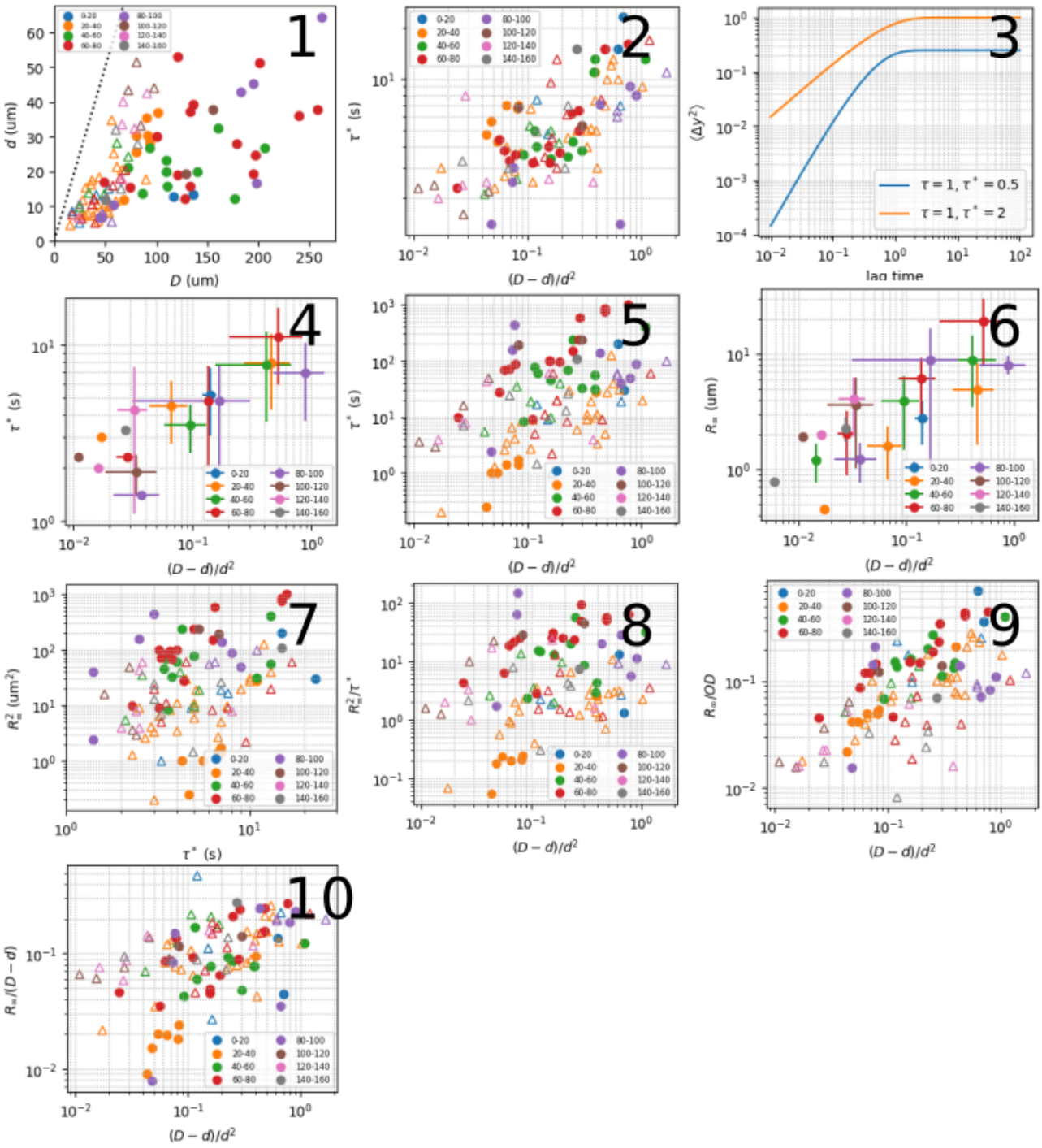
```
data.scatter_0() # left
data.plot_0(nbins=5, overlap=0) # middle
data.plot_0(nbins=6, overlap=0.5) # right
```



Examples

Here, the outcome of each function (almost) is shown.

```
data.parameter_space(highlight_Chile_data=True) # 1
data.scatter_0(mode="log", highlight_Chile_data=True) # 2
data.plot_MSD_model_Cristian() # 3
data.plot_0(nbins=5, overlap=0, mode="log") # 4
data.scatter_1(mode="log", highlight_Chile_data=True) # 5
data.plot_1(nbins=5, overlap=0, mode="log") # 6
data.Rinf2_tau() # 7
data.Rinf2_over_tau() # 8
data.rescale_Rinf_OD() # 9
data.rescale_Rinf_freespace() # 10
```


1
d (um)
D (um)
2
τ* (s)
(D − d)/d²
3
⟨Δy²⟩
lag time
τ = 1, τ* = 0.5
τ = 1, τ* = 2
4
τ* (s)
(D − d)/d²
5
τ* (s)
(D − d)/d²
6
R∞ (um)
(D − d)/d²
7
R∞² (um²)
τ* (s)
8
R∞²/τ*
(D − d)/d²
9
R∞/OD
(D − d)/d²
10
R∞/(D − d)
(D − d)/d²
0-20
20-40
40-60
60-80
80-100
100-120
120-140
140-160