reg_year_r_random_points_new_resampled

March 3, 2024

0.1 Importing

[ ]:

```
import xarray as xr
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn import preprocessing

from sklearn.neural_network import MLPRegressor
from sklearn.ensemble import BaggingRegressor

from sklearn.metrics import mean_squared_error as mse

from tqdm.auto import tqdm

import dill
import random

import salishsea_tools.viz_tools as sa_vi
```

0.2 Datasets Preparation

[ ]:

```
def datasets_preparation(dataset):

    drivers = np.stack([np.ravel(dataset['Temperature_(0m-15m)']),
        np.ravel(dataset['Temperature_(15m-100m)']), np.
 ↪ravel(dataset['Salinity_(0m-15m)']),
        np.ravel(dataset['Salinity_(15m-100m)'])])
    indx = np.where(~np.isnan(drivers).any(axis=0))
    drivers = drivers[:,indx[0]]

    diat = np.ravel(dataset['Diatom'])
    diat = diat[indx[0]]

    return(drivers, diat, indx)
```

0.3 Regressor

[ ]:

```
def regressor (inputs, targets):

    inputs = inputs.transpose()

    # Regressor
    scale = preprocessing.StandardScaler()
    inputs = scale.fit_transform(inputs)
    X_train, _, y_train, _ = train_test_split(inputs, targets, train_size=0.35)

    drivers = None
    diat = None

    inputs = None
    targets = None

    model = MLPRegressor(hidden_layer_sizes=200, alpha=0.002)
    regr = BaggingRegressor(model, n_estimators=12, n_jobs=4).fit(X_train,␣
 ↪y_train)

    return (regr)
```

0.4 Regressor 2

[ ]:

```
def regressor2 (inputs, targets, variable_name):

    inputs = inputs.transpose()

    # Regressor
    scale = preprocessing.StandardScaler()
    inputs2 = scale.fit_transform(inputs)

    outputs_test = regr.predict(inputs2)

    m = scatter_plot(targets, outputs_test, variable_name)
    r = np.round(np.corrcoef(targets, outputs_test)[0][1],3)
    rms = mse(targets, outputs_test)

    return (r, rms, m)
```

0.5 Regressor 3

[ ]:

```
def regressor3 (inputs, targets):

    inputs = inputs.transpose()
    # Regressor
    scale = preprocessing.StandardScaler()
    inputs2 = scale.fit_transform(inputs)

    outputs_test = regr.predict(inputs2)

    # compute slope m and intercept b
    m, b = np.polyfit(targets, outputs_test, deg=1)

    r = np.round(np.corrcoef(targets, outputs_test)[0][1],3)
    rms = mse(targets, outputs_test)

    return (r, rms, m)
```

0.6 Regressor 4

[ ]:

```
def regressor4 (inputs, targets, variable_name):

    inputs = inputs.transpose()

    # Regressor
    scale = preprocessing.StandardScaler()
    inputs2 = scale.fit_transform(inputs)

    outputs = regr.predict(inputs2)

    # Post processing
    indx2 = np.full((len(diat_i.y)*len(diat_i.x)),np.nan)
    indx2[indx[0]] = outputs
    model = np.reshape(indx2,(len(diat_i.y),len(diat_i.x)))

    m = scatter_plot(targets, outputs, variable_name + str(dates[i].date()))

    # Preparation of the dataarray
    model = xr.DataArray(model,
        coords = {'y': diat_i.y, 'x': diat_i.x},
        dims = ['y','x'],
        attrs=dict( long_name = variable_name + "Concentration",
        units="mmol m-2"),)

    plotting3(targets, model, diat_i, variable_name)
```

1 Printing

[ ]:

```
def printing (targets, outputs, m):

    print ('The amount of data points is', outputs.size)
    print ('The slope of the best fitting line is ', np.round(m,3))
    print ('The correlation coefficient is:', np.round(np.corrcoef(targets,␣
 ↪outputs)[0][1],3))
    print (' The mean square error is:', np.round(mse(targets,outputs),5))
```

1.1 Scatter Plot

[ ]:

```
def scatter_plot(targets, outputs, variable_name):

    # compute slope m and intercept b
    m, b = np.polyfit(targets, outputs, deg=1)

    printing(targets, outputs, m)

    fig, ax = plt.subplots(2, figsize=(5,10), layout='constrained')

    ax[0].scatter(targets,outputs, alpha = 0.2, s = 10)

    lims = [np.min([ax[0].get_xlim(), ax[0].get_ylim()]),
        np.max([ax[0].get_xlim(), ax[0].get_ylim()])]

    # plot fitted y = m*x + b
    ax[0].axline(xy1=(0, b), slope=m, color='r')

    ax[0].set_xlabel('targets')
    ax[0].set_ylabel('outputs')
    ax[0].set_xlim(lims)
    ax[0].set_ylim(lims)
    ax[0].set_aspect('equal')

    ax[0].plot(lims, lims,linestyle = '--',color = 'k')

    h = ax[1].hist2d(targets,outputs, bins=100, cmap='jet',
        range=[lims,lims], cmin=0.1, norm='log')

    ax[1].plot(lims, lims,linestyle = '--',color = 'k')

    # plot fitted y = m*x + b
    ax[1].axline(xy1=(0, b), slope=m, color='r')

    ax[1].set_xlabel('targets')
    ax[1].set_ylabel('outputs')
    ax[1].set_aspect('equal')

    fig.colorbar(h[3],ax=ax[1], location='bottom')

    fig.suptitle(variable_name)

    plt.show()

    return (m)
```

1.2 Plotting

[ ]:

```
def plotting(variable, name):

    plt.plot(years,variable, marker = '.', linestyle = '')
    plt.legend(['diatom','flagellate'])
    plt.xlabel('Years')
    plt.ylabel(name)
    plt.show()
```

1.3 Plotting 2

[ ]:

```
def plotting2(variable,title):

    fig, ax = plt.subplots()

    scatter= ax.scatter(dates,variable, marker='.', c=pd.DatetimeIndex(dates).
 ↪month)

    ax.legend(handles=scatter.legend_elements()[0],␣
 ↪labels=['February','March','April'])
    fig.suptitle('Daily ' + title + ' (15 Feb - 30 Apr)')

    fig.show()
```

1.4 Plotting 3

[ ]:

```
def plotting3(targets, model, variable, variable_name):

    fig, ax = plt.subplots(2,2, figsize = (10,15))

    cmap = plt.get_cmap('cubehelix')
    cmap.set_bad('gray')

    variable.plot(ax=ax[0,0], cmap=cmap, vmin = targets.min(), vmax =targets.
 ↪max(), cbar_kwargs={'label': variable_name + ' Concentration  [mmol m-2]'})
```

```
    model.plot(ax=ax[0,1], cmap=cmap, vmin = targets.min(), vmax = targets.
 ↪max(), cbar_kwargs={'label': variable_name + ' Concentration  [mmol m-2]'})
    ((variable-model) / variable * 100).plot(ax=ax[1,0], cmap=cmap,␣
 ↪cbar_kwargs={'label': variable_name + ' Concentration  [percentage]'})

    plt.subplots_adjust(left=0.1,
        bottom=0.1,
        right=0.95,
        top=0.95,
        wspace=0.35,
        hspace=0.35)

    sa_vi.set_aspect(ax[0,0])
    sa_vi.set_aspect(ax[0,1])
    sa_vi.set_aspect(ax[1,0])


    ax[0,0].title.set_text(variable_name + ' (targets)')
    ax[0,1].title.set_text(variable_name + ' (outputs)')
    ax[1,0].title.set_text('targets - outputs')
    ax[1,1].axis('off')

    fig.suptitle(str(dates[i].date()))

    plt.show()
```

1.5 Training (Random Points)

[ ]:

```
ds = xr.open_dataset('/data/ibougoudis/MOAD/files/integrated_model_var_old.nc')

ds = ds.isel(time_counter = (np.arange(0, len(ds.Diatom.time_counter),2)),
    y=(np.arange(ds.y[0], ds.y[-1], 5)),
    x=(np.arange(ds.x[0], ds.x[-1], 5)))

dates = pd.DatetimeIndex(ds['time_counter'].values)

drivers, diat, _ = datasets_preparation(ds)

regr = regressor(drivers, diat)
```

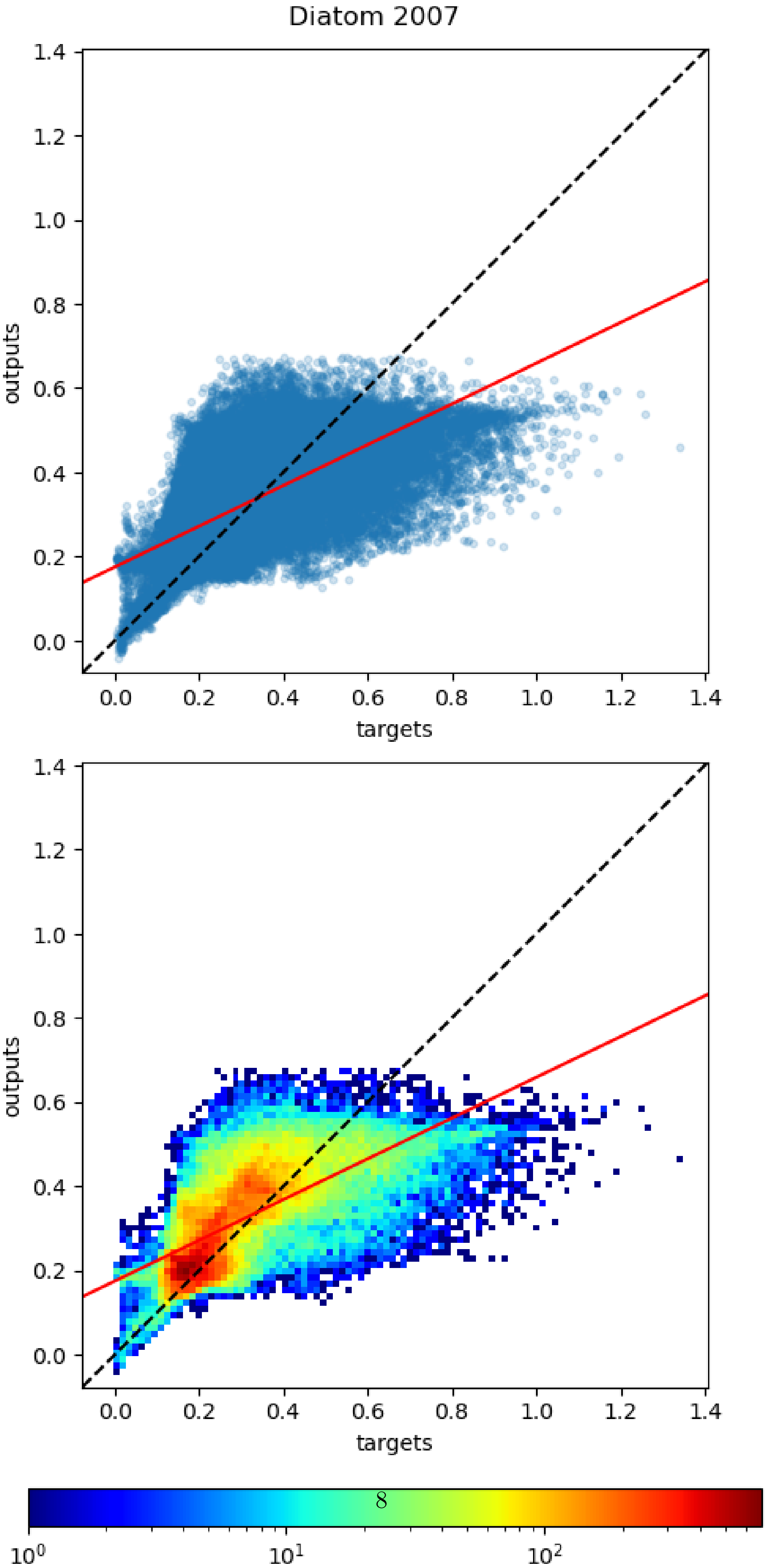
1.6 Other Years (Anually)

[ ]:

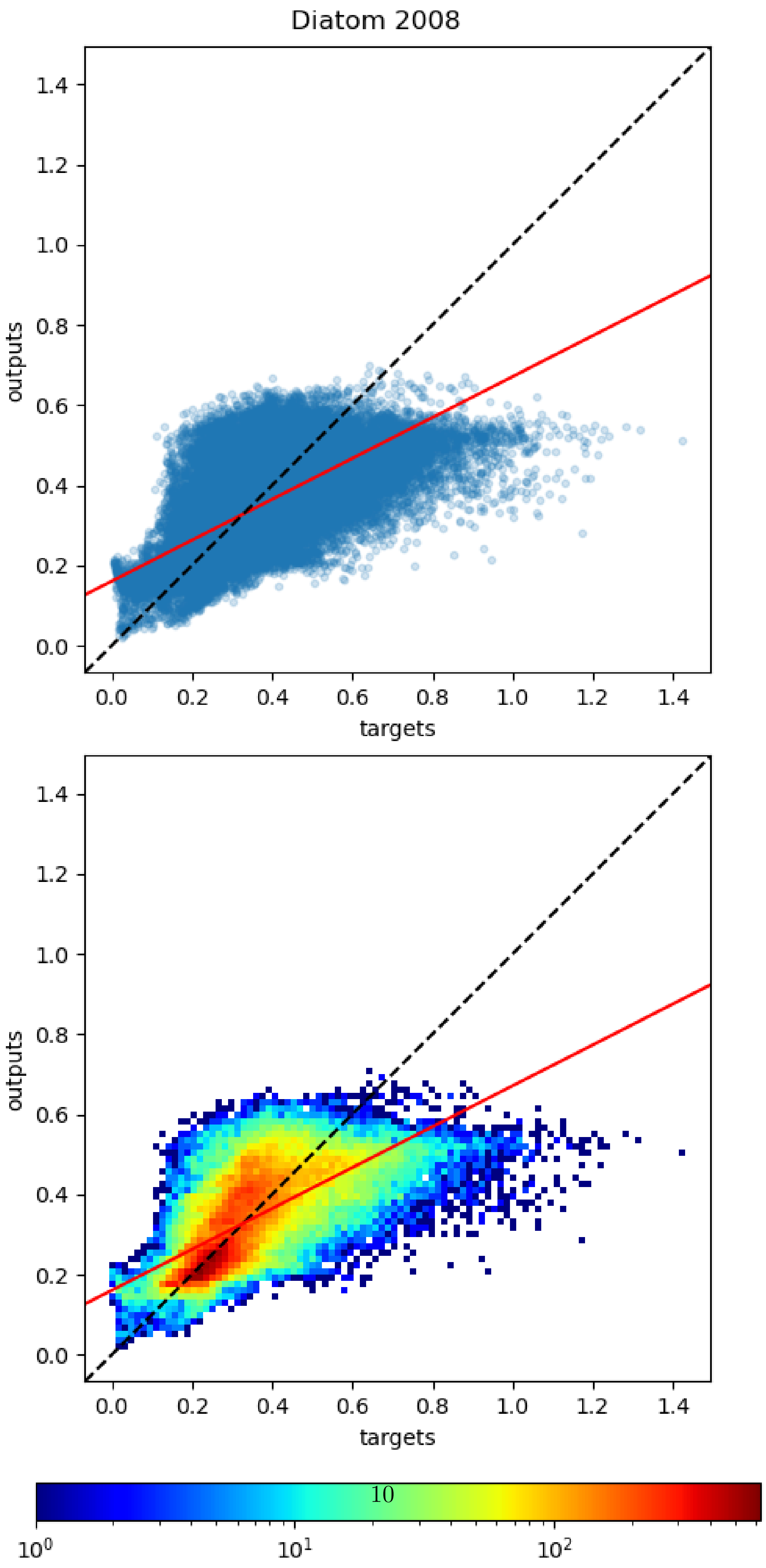
```
years = range (2007,2024)

r_all = []
rms_all = []
slope_all = []

for year in tqdm(range (2007,2024)):

    dataset = ds.sel(time_counter=str(year))

    drivers, diat, _ = datasets_preparation(dataset)

    r, rms, m = regressor2(drivers, diat, 'Diatom ' + str(year))

    r_all.append(r)
    rms_all.append(rms)
    slope_all.append(m)

plotting(np.transpose(r_all), 'Correlation Coefficient')
plotting(np.transpose(rms_all), 'Mean Square Error')
plotting (np.transpose(slope_all), 'Slope of the best fitting line')
```

```
  0%|          | 0/17 [00:00<?, ?it/s]
```

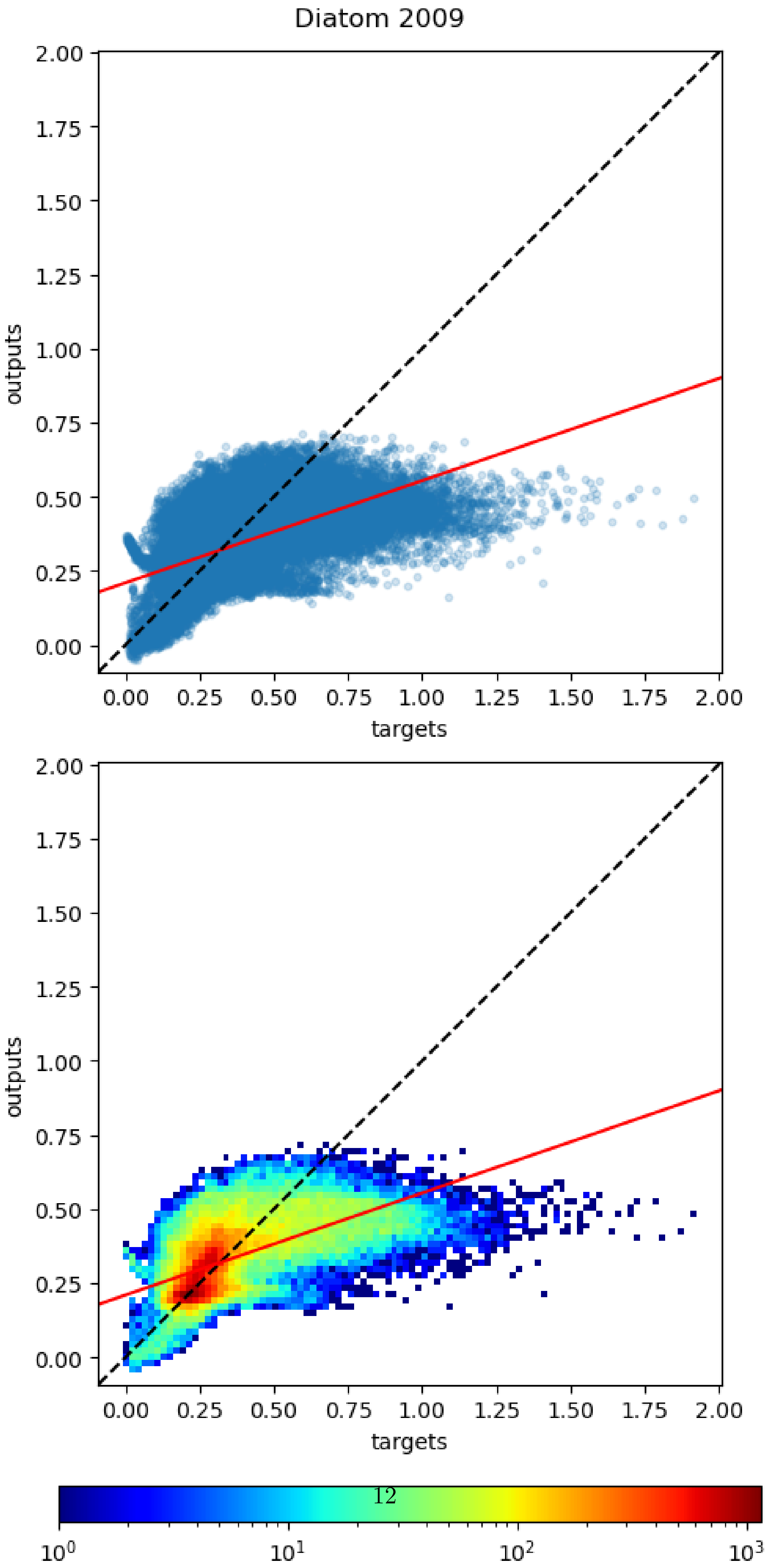
```
The amount of data points is 70794
The slope of the best fitting line is  0.484
The correlation coefficient is: 0.67
 The mean square error is: 0.01434
```


Diatom 2007
outputs
targets
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
outputs
targets
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
10^0
10^1
10^2

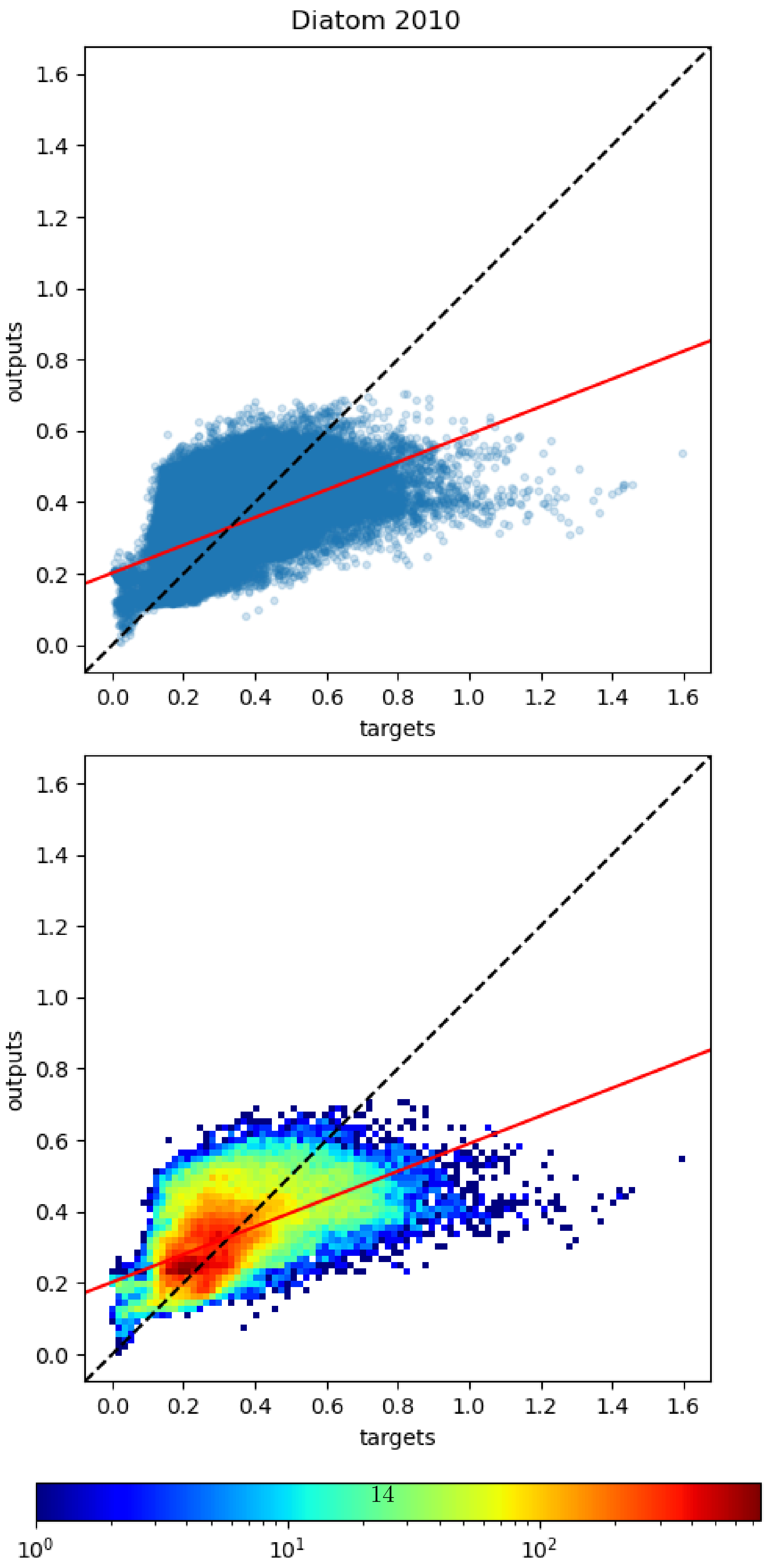
```
The amount of data points is 70794
The slope of the best fitting line is  0.51
The correlation coefficient is: 0.659
 The mean square error is: 0.01228
```


Diatom 2008
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
10
10^0
10^1
10^2

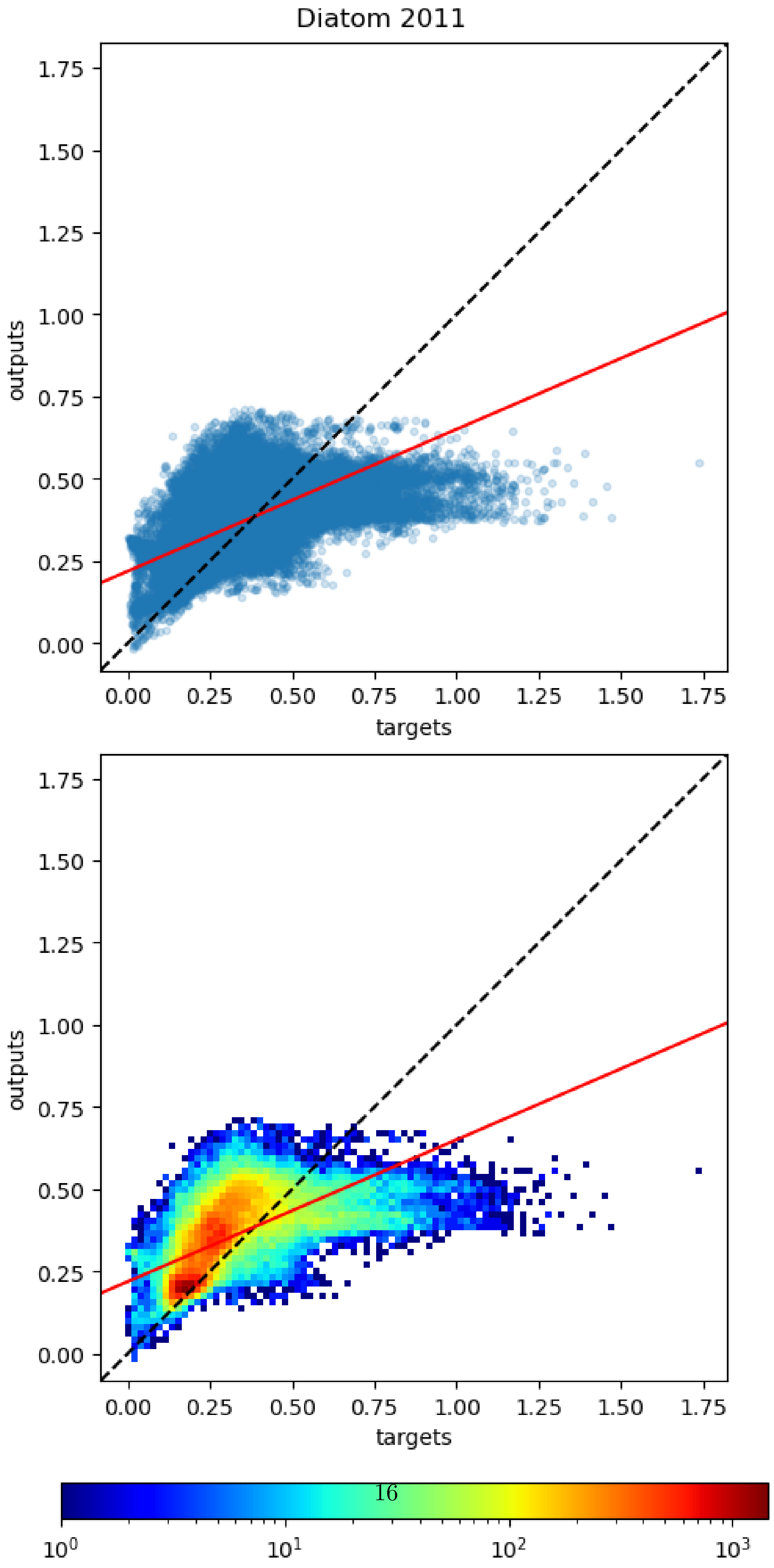
```
The amount of data points is 68931
The slope of the best fitting line is  0.344
The correlation coefficient is: 0.596
 The mean square error is: 0.02519
```


Diatom 2009
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
targets
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
targets
10^0
10^1
10^2
10^3

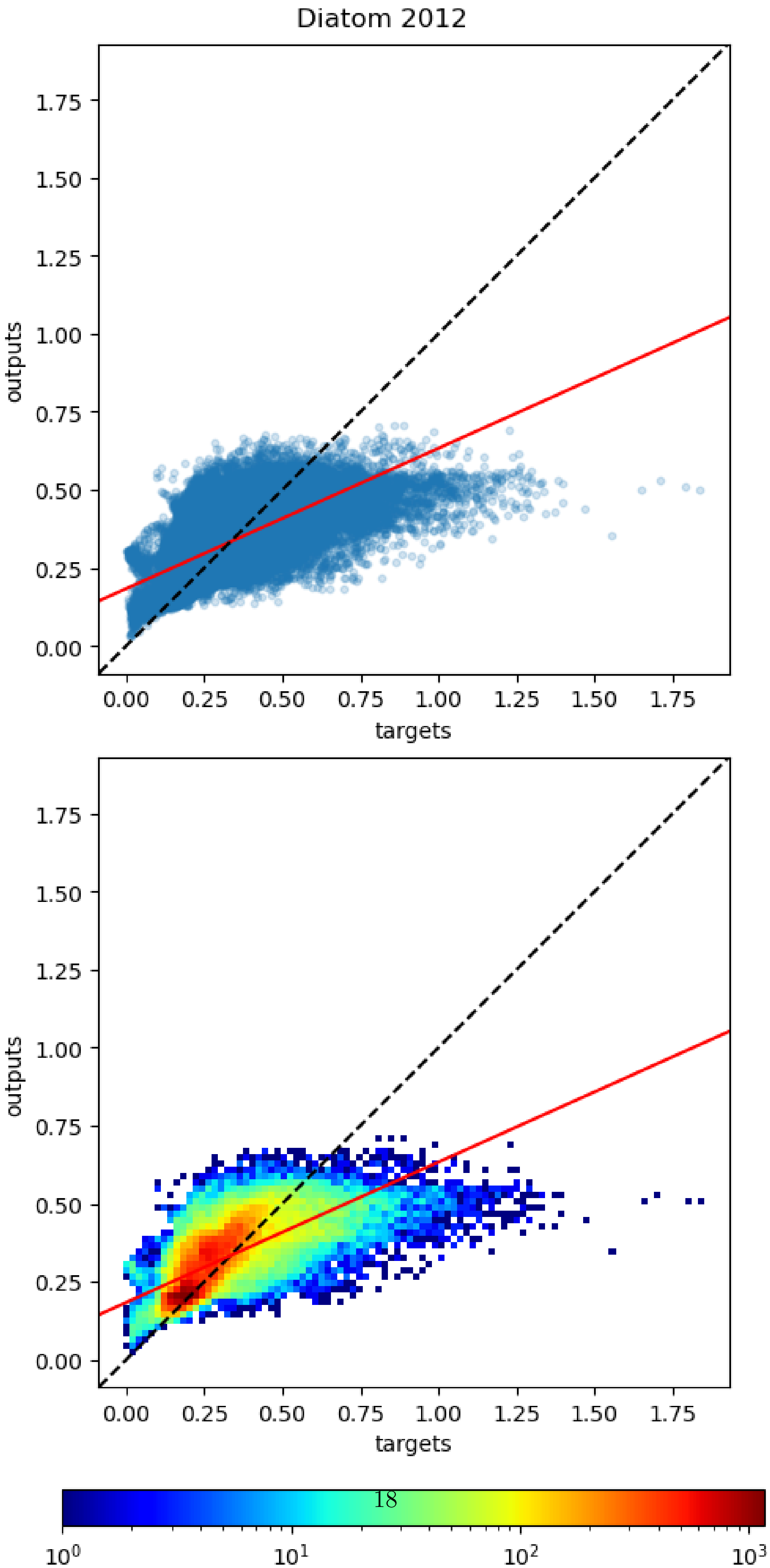
```
The amount of data points is 70794
The slope of the best fitting line is  0.388
The correlation coefficient is: 0.566
 The mean square error is: 0.01468
```


Diatom 2010
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
10^0
10^1
10^2

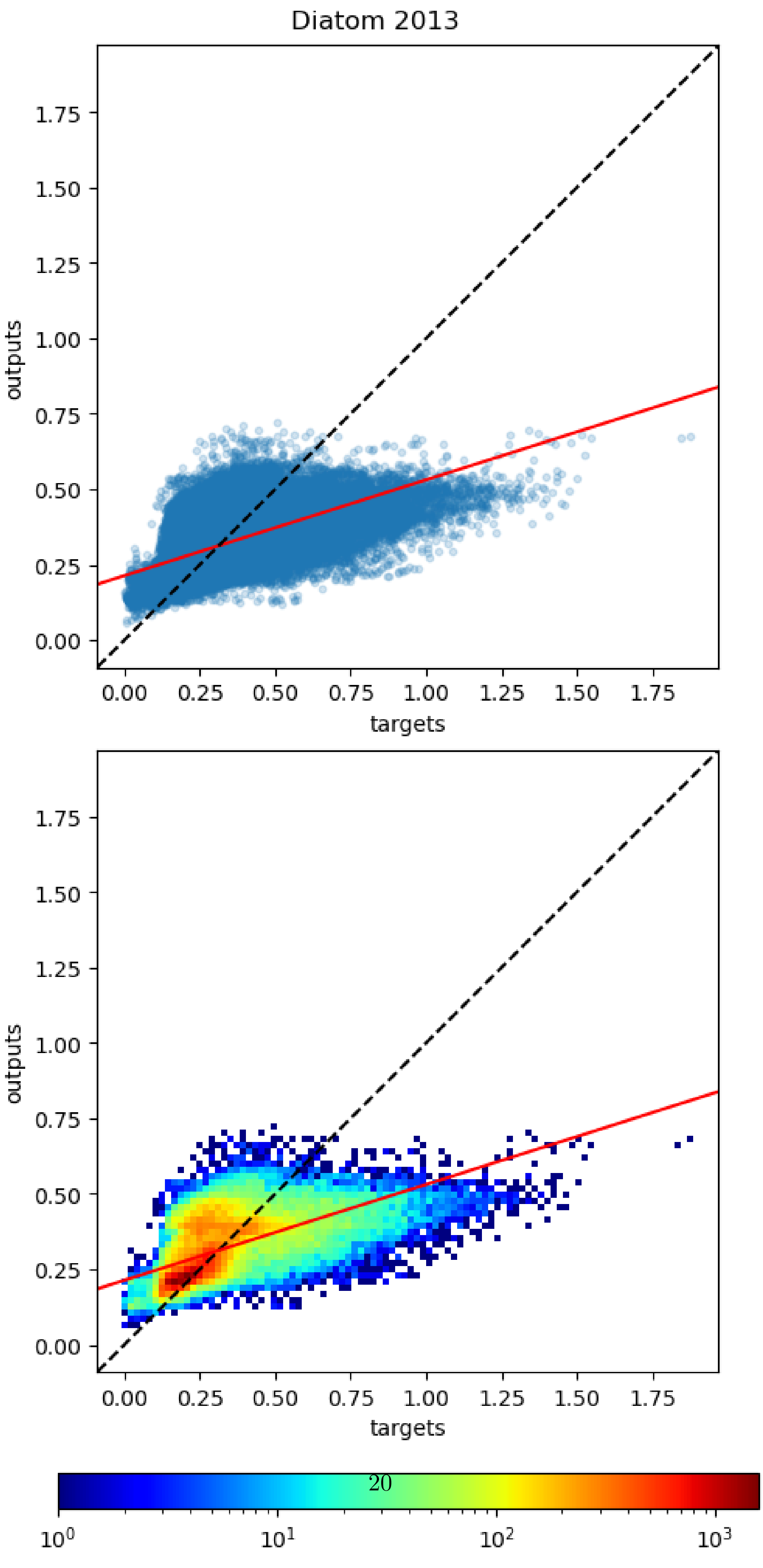
```
The amount of data points is 68931
The slope of the best fitting line is  0.432
The correlation coefficient is: 0.599
 The mean square error is: 0.01815
```


Diatom 2011
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
10^0
10^1
10^2
10^3

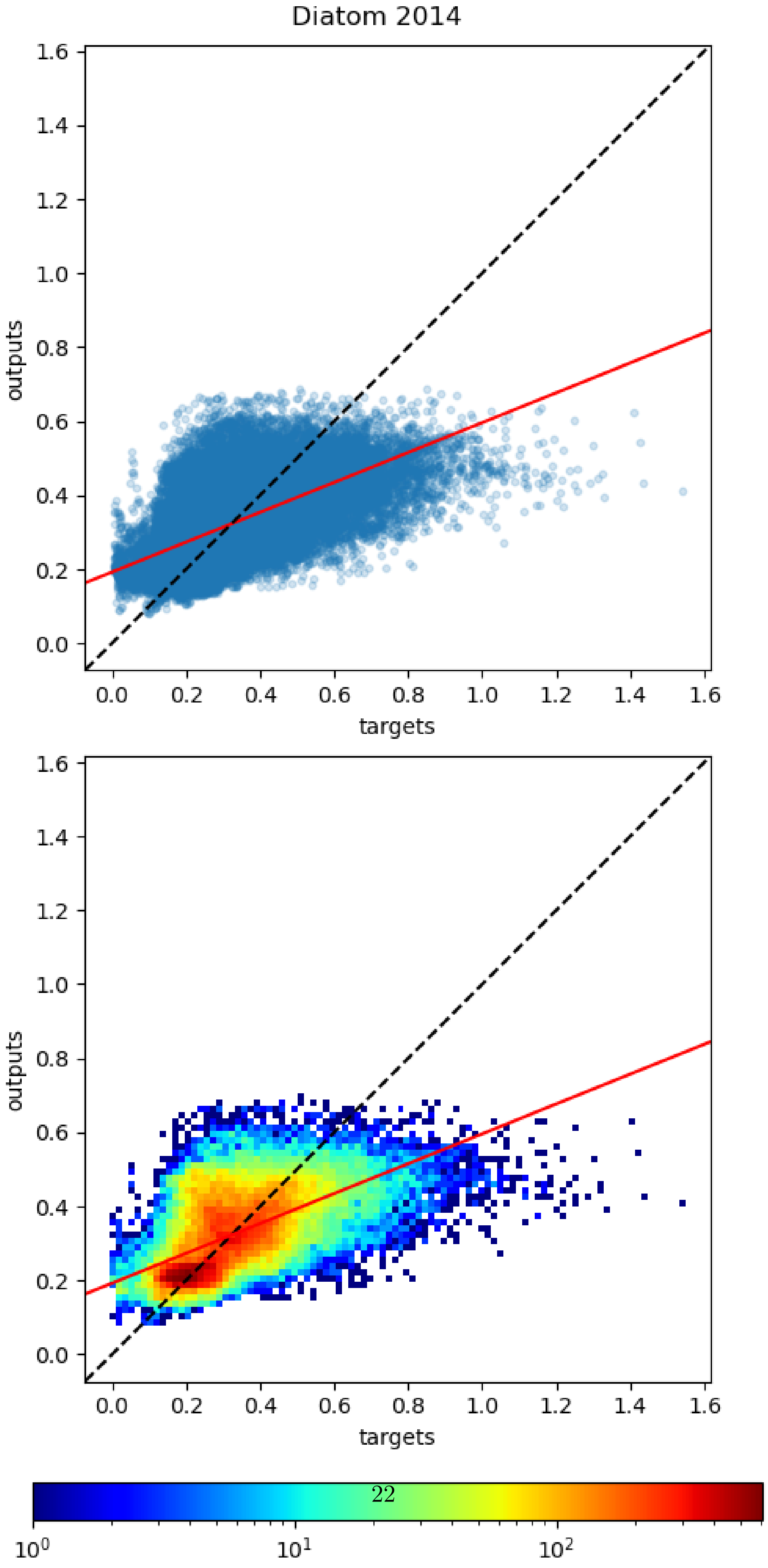
```
The amount of data points is 70794
The slope of the best fitting line is  0.45
The correlation coefficient is: 0.687
 The mean square error is: 0.01352
```


Diatom 2012
outputs
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
10^0
10^1
10^2
10^3

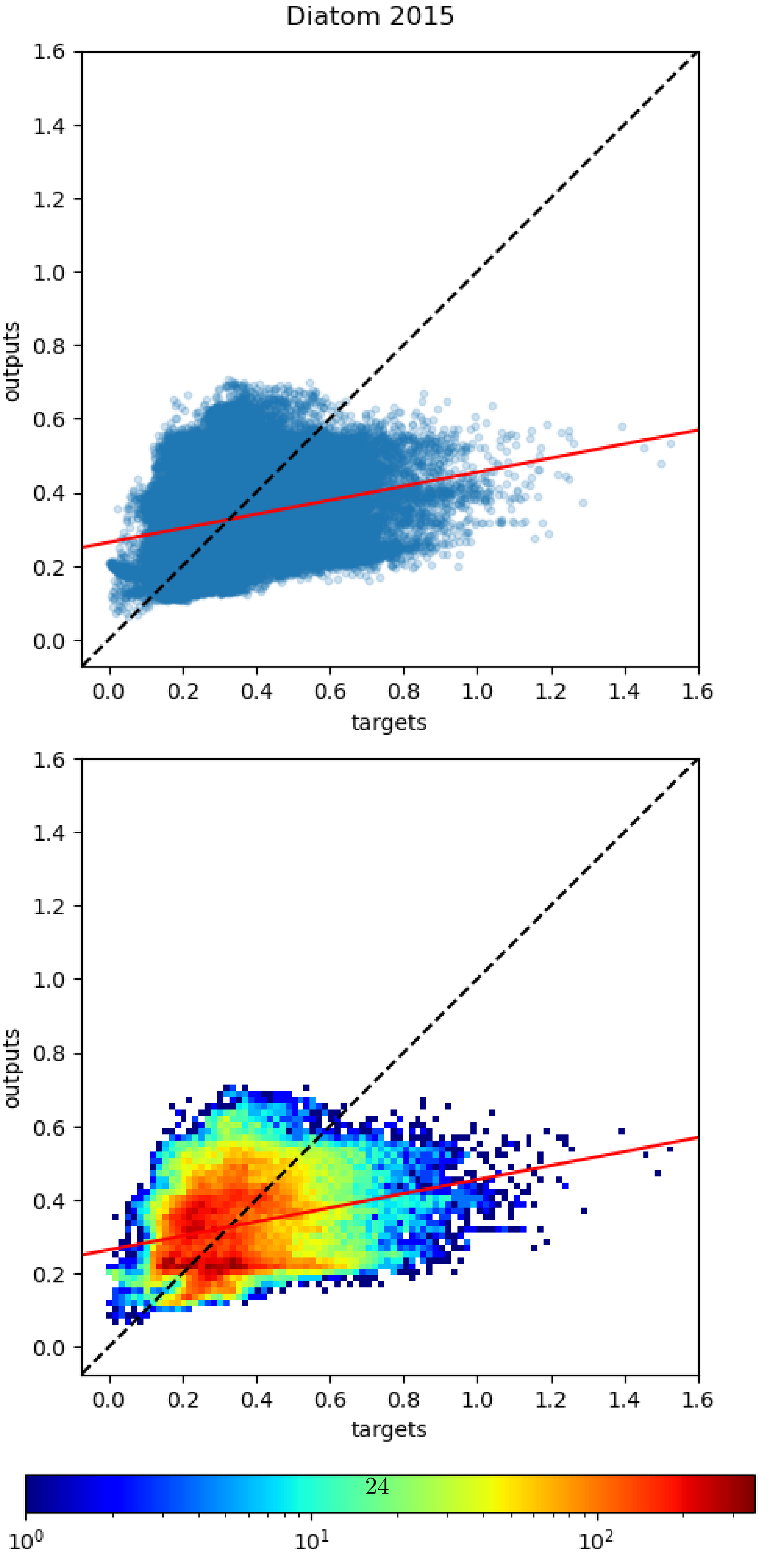
```
The amount of data points is 70794
The slope of the best fitting line is  0.317
The correlation coefficient is: 0.589
 The mean square error is: 0.02121
```


Diatom 2013
outputs
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
outputs
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
10^0
10^1
10^2
10^3

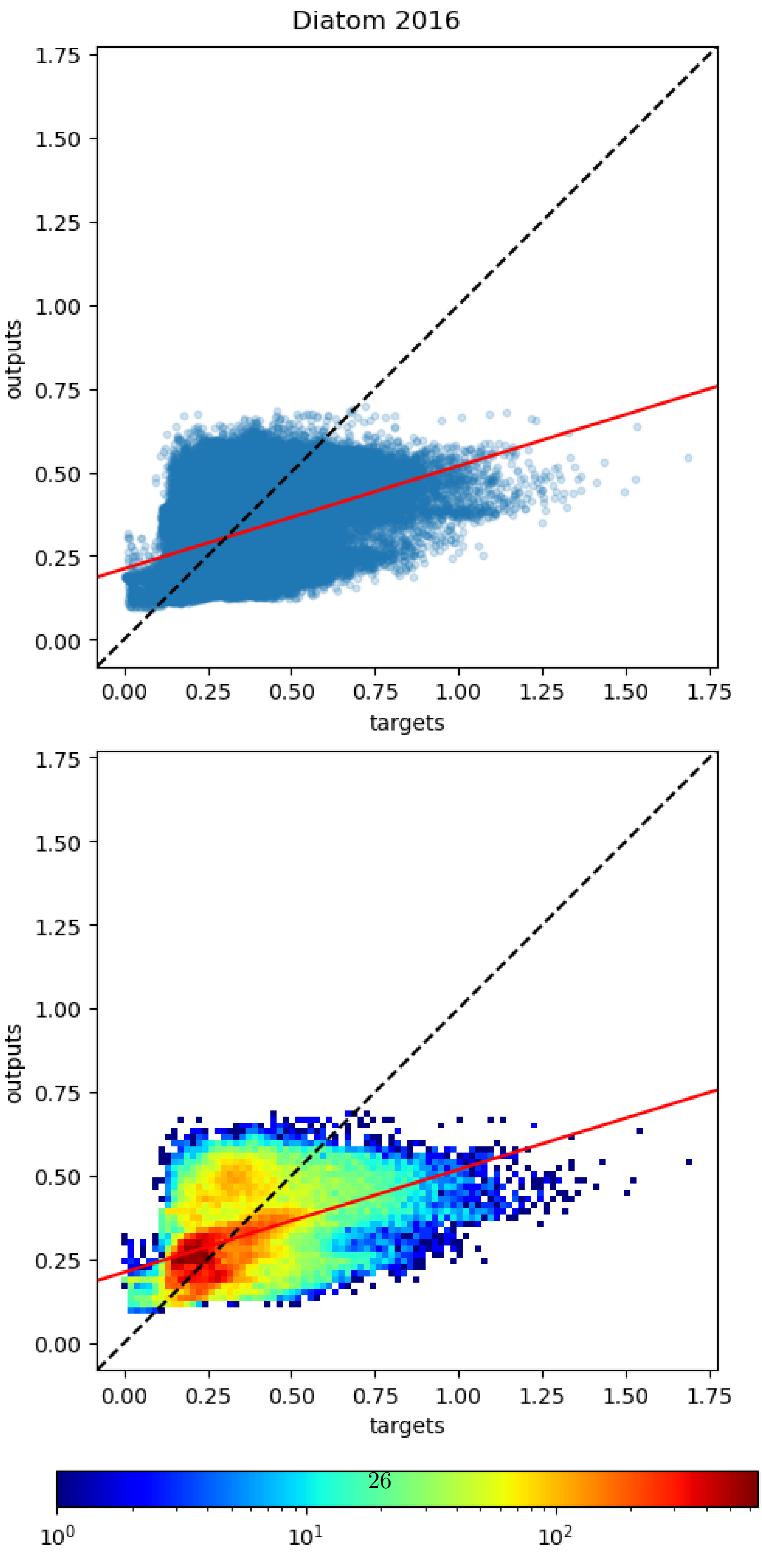
```
The amount of data points is 68931
The slope of the best fitting line is  0.404
The correlation coefficient is: 0.582
 The mean square error is: 0.01414
```


Diatom 2014
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
10^0
10^1
10^2

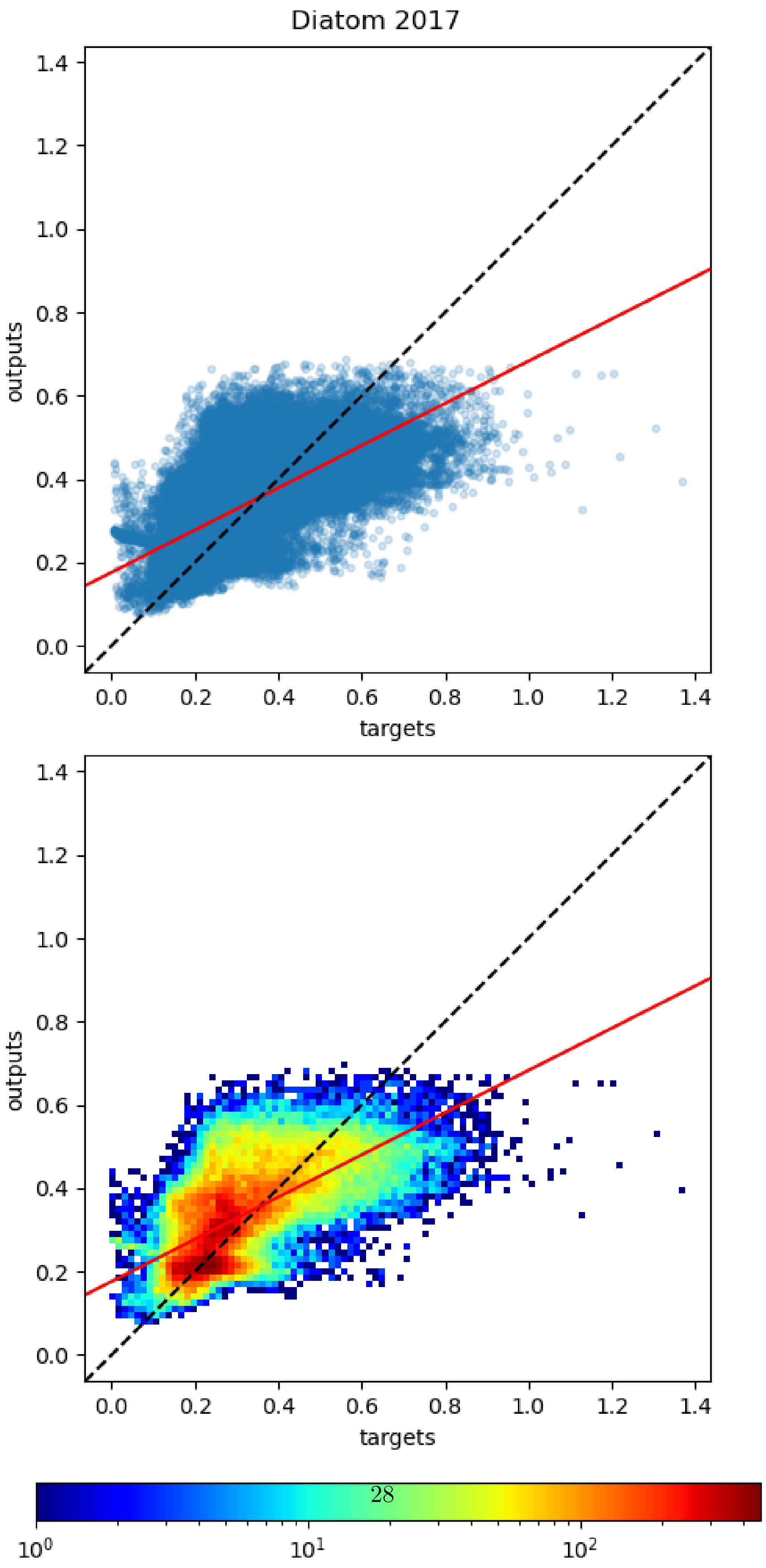
```
The amount of data points is 70794
The slope of the best fitting line is  0.191
The correlation coefficient is: 0.266
 The mean square error is: 0.02648
```


Diatom 2015
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
10^0
10^1
10^2

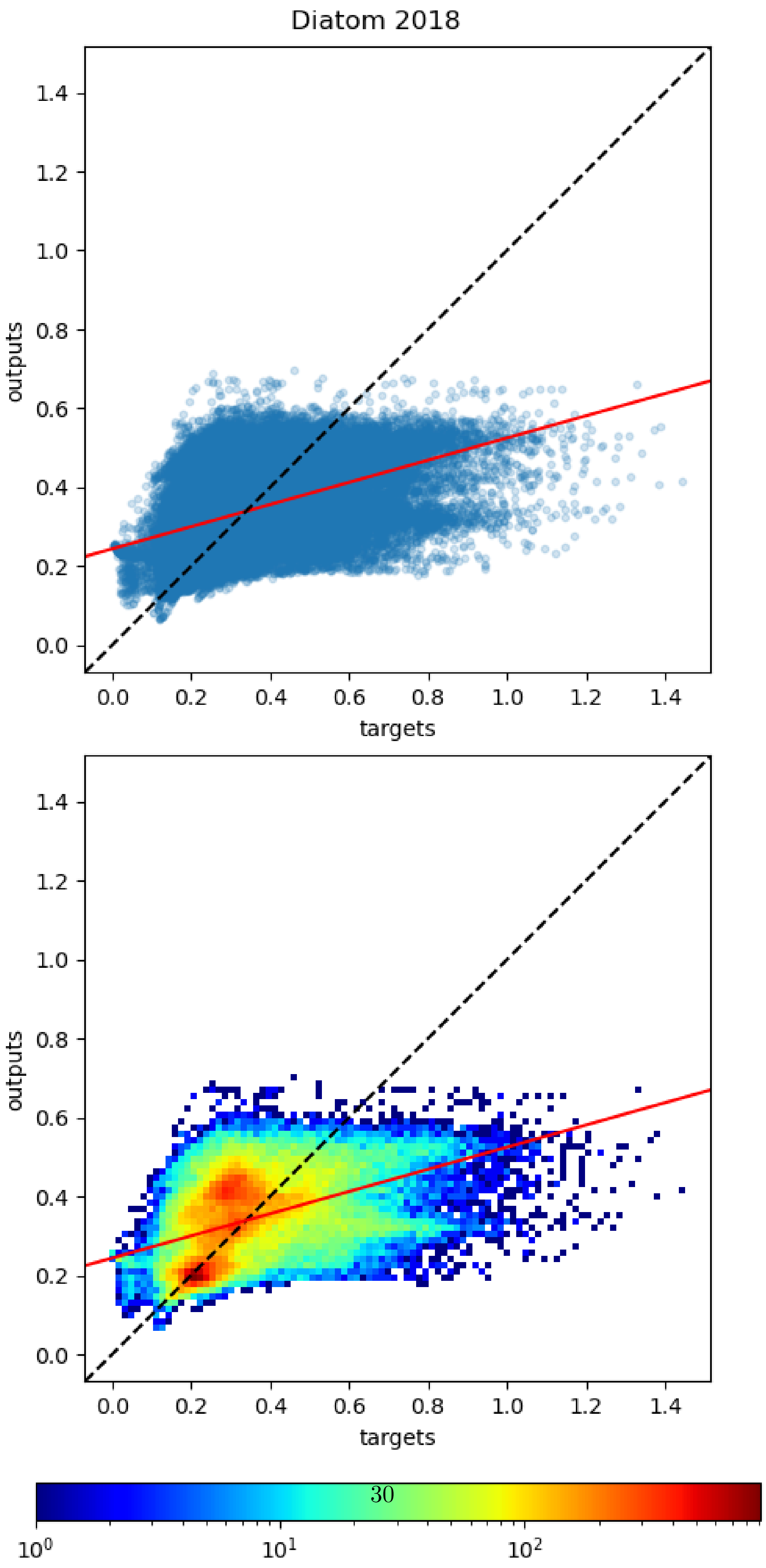
```
The amount of data points is 70794
The slope of the best fitting line is  0.307
The correlation coefficient is: 0.455
 The mean square error is: 0.02429
```


Diatom 2016
outputs
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
10^0
10^1
10^2

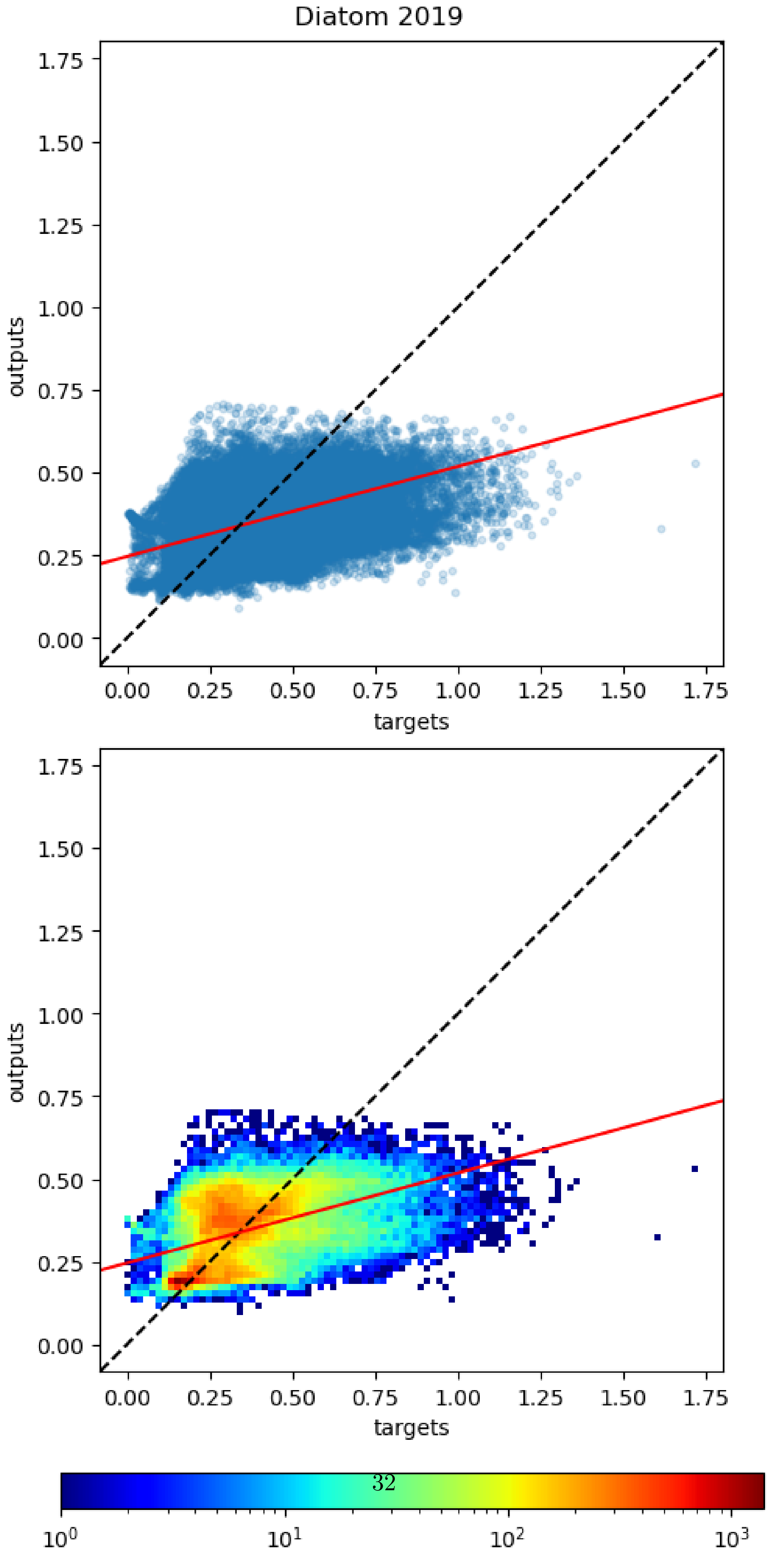
```
The amount of data points is 68931
The slope of the best fitting line is  0.507
The correlation coefficient is: 0.633
 The mean square error is: 0.01183
```


Diatom 2017
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
10^0
10^1
10^2

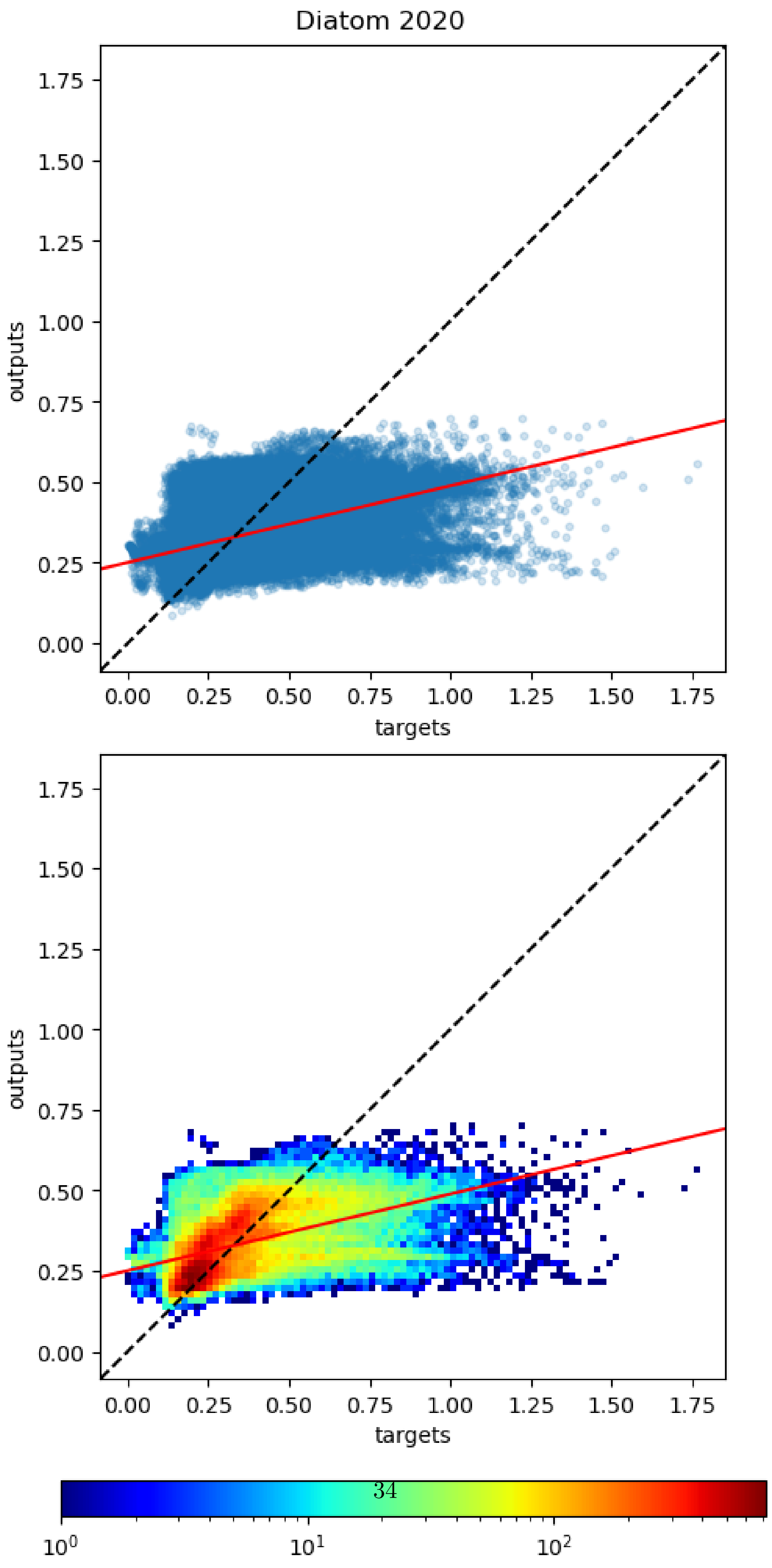
```
The amount of data points is 70794
The slope of the best fitting line is  0.281
The correlation coefficient is: 0.435
 The mean square error is: 0.02128
```


Diatom 2018
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
10^0
10^1
10^2

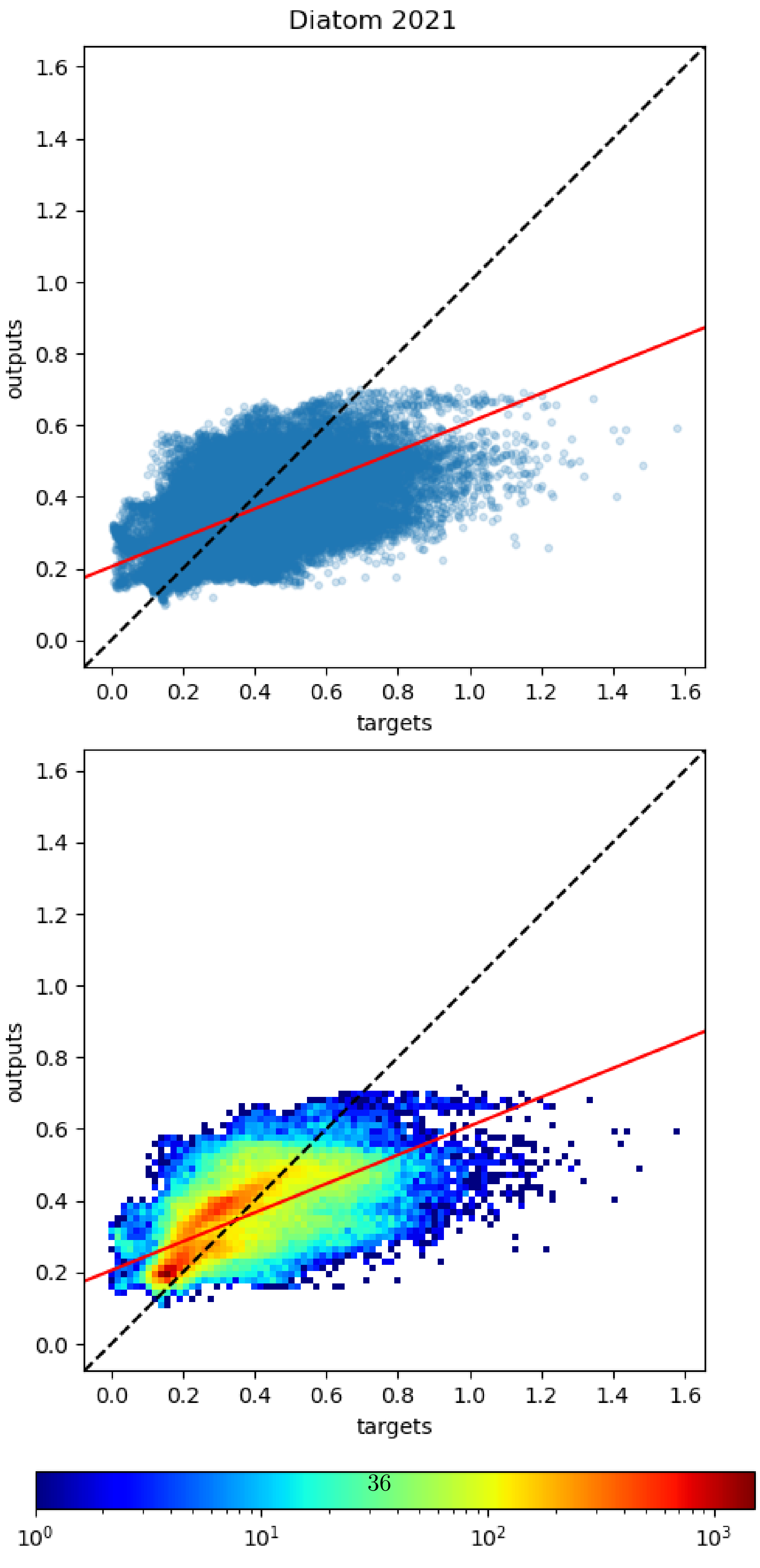
```
The amount of data points is 68931
The slope of the best fitting line is  0.272
The correlation coefficient is: 0.451
 The mean square error is: 0.02484
```


Diatom 2019
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
10^0
10^1
10^2
10^3

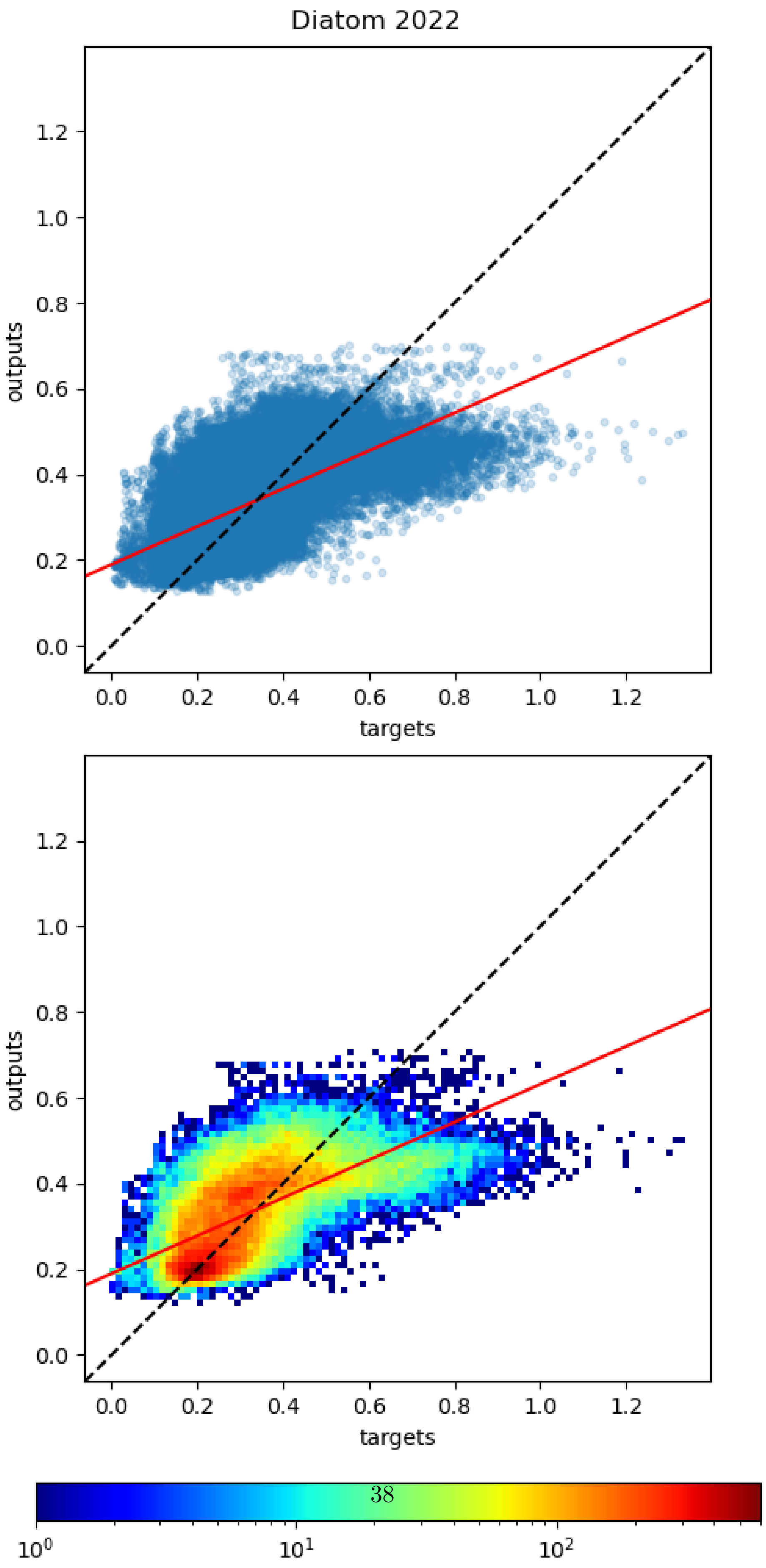
```
The amount of data points is 70794
The slope of the best fitting line is  0.238
The correlation coefficient is: 0.484
 The mean square error is: 0.03245
```


Diatom 2020
outputs
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
outputs
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
10^0
10^1
10^2

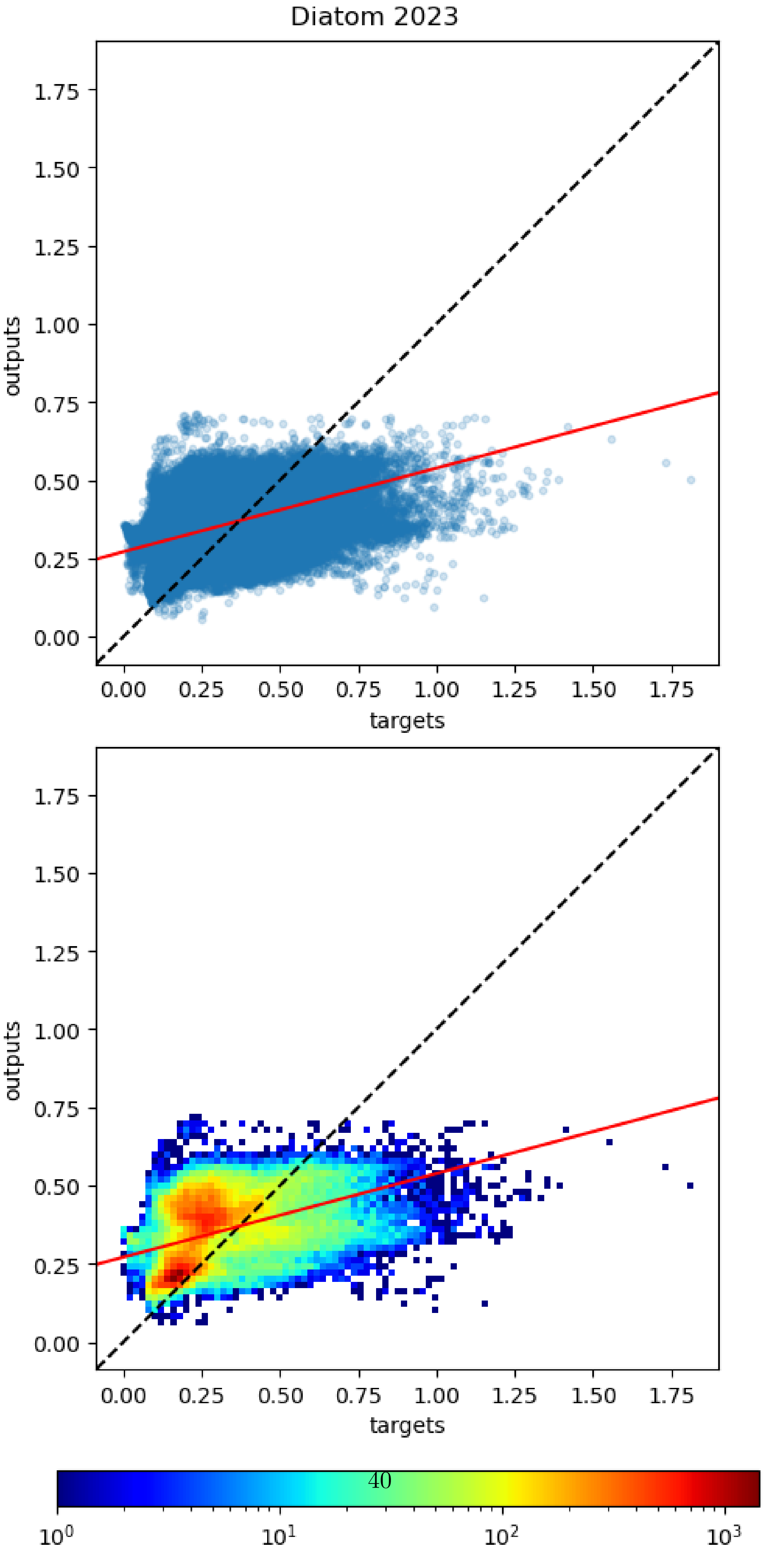
```
The amount of data points is 70794
The slope of the best fitting line is  0.403
The correlation coefficient is: 0.653
 The mean square error is: 0.0173
```

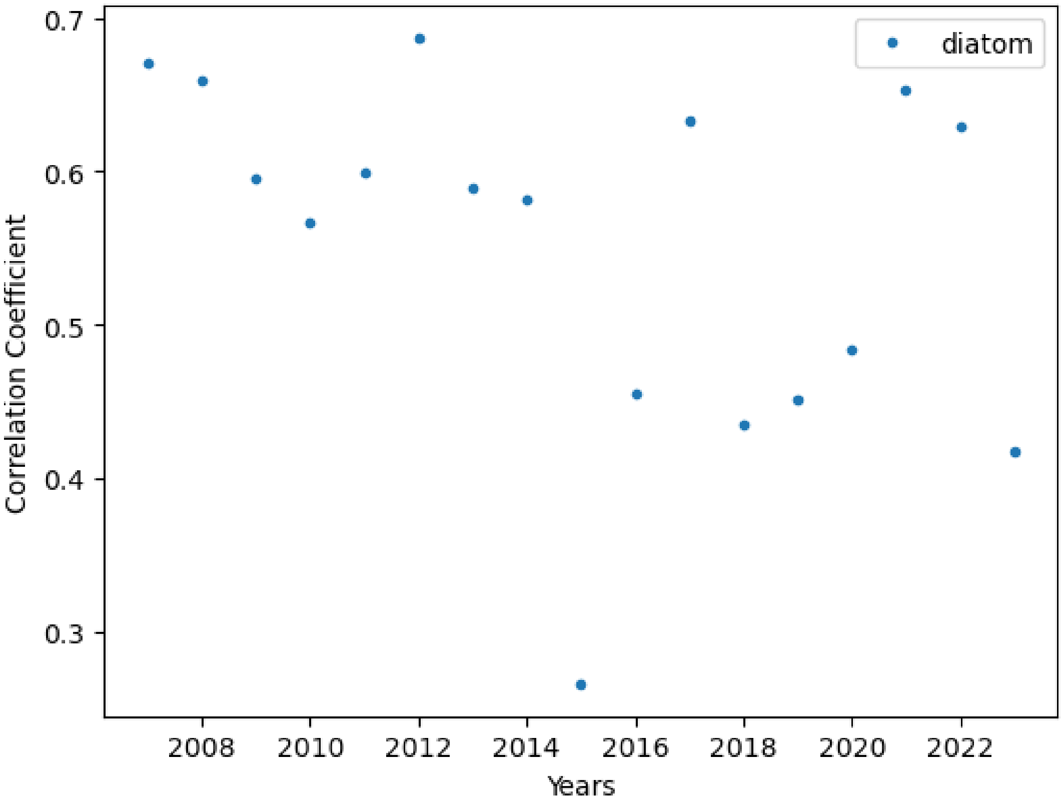

Diatom 2021
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
outputs
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
targets
10^0
10^1
10^2
10^3

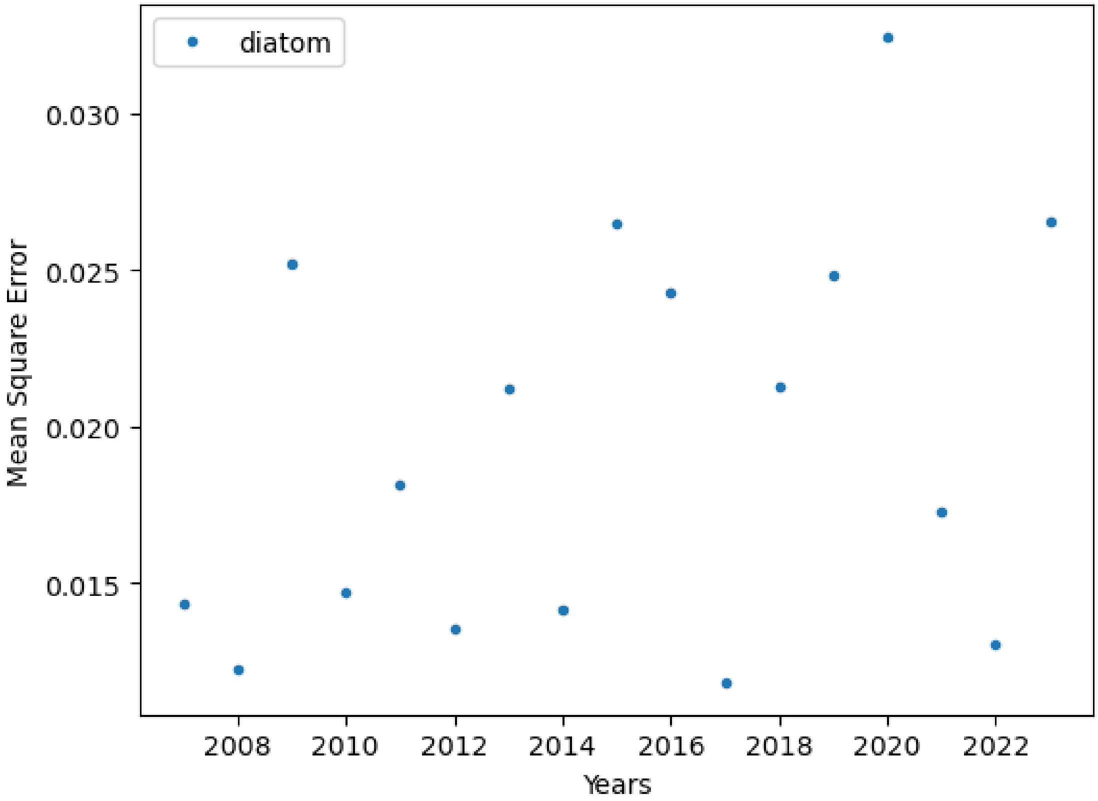
```
The amount of data points is 68931
The slope of the best fitting line is  0.442
The correlation coefficient is: 0.629
 The mean square error is: 0.01305
```

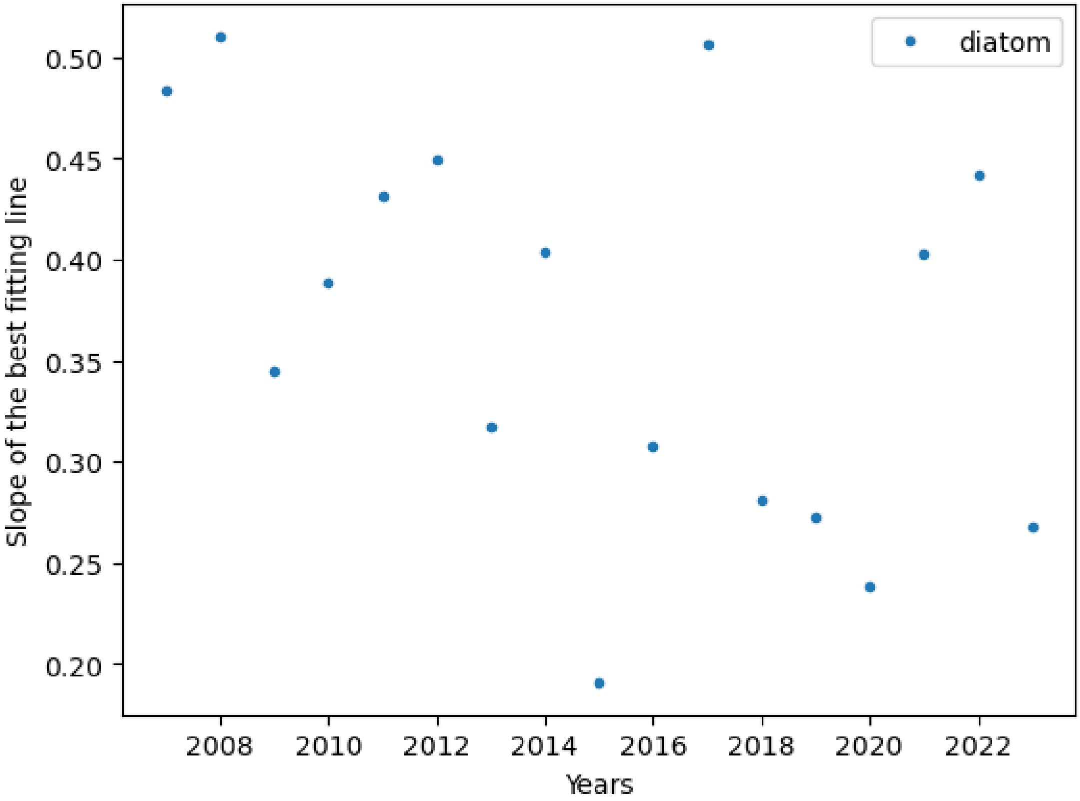

Diatom 2022
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
outputs
targets
0.0
0.2
0.4
0.6
0.8
1.0
1.2
10^0
10^1
10^2

```
The amount of data points is 70794
The slope of the best fitting line is  0.268
The correlation coefficient is: 0.417
 The mean square error is: 0.02655
```


Diatom 2023
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
outputs
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
targets
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
10^0
10^1
10^2
10^3


0.7
0.6
0.5
0.4
0.3
Correlation Coefficient
2008
2010
2012
2014
2016
2018
2020
2022
Years
● diatom


● diatom
0.030
0.025
0.020
0.015
Mean Square Error
2008
2010
2012
2014
2016
2018
2020
2022
Years

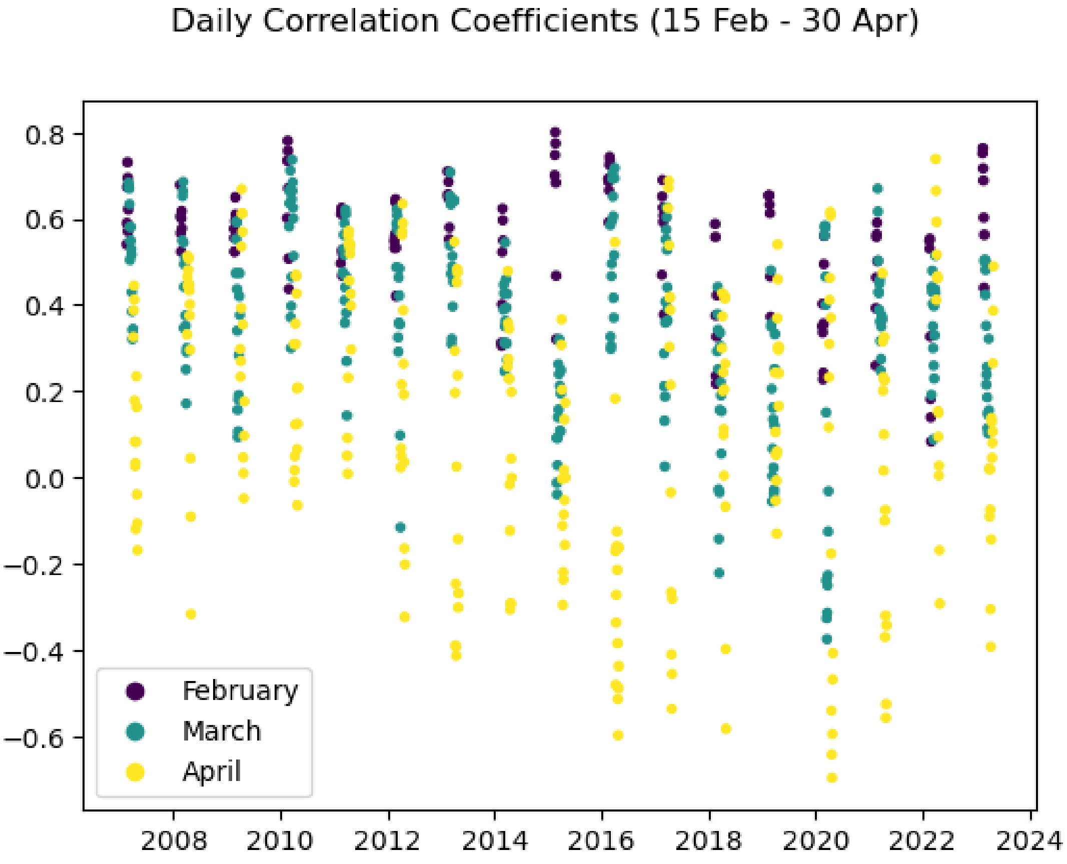


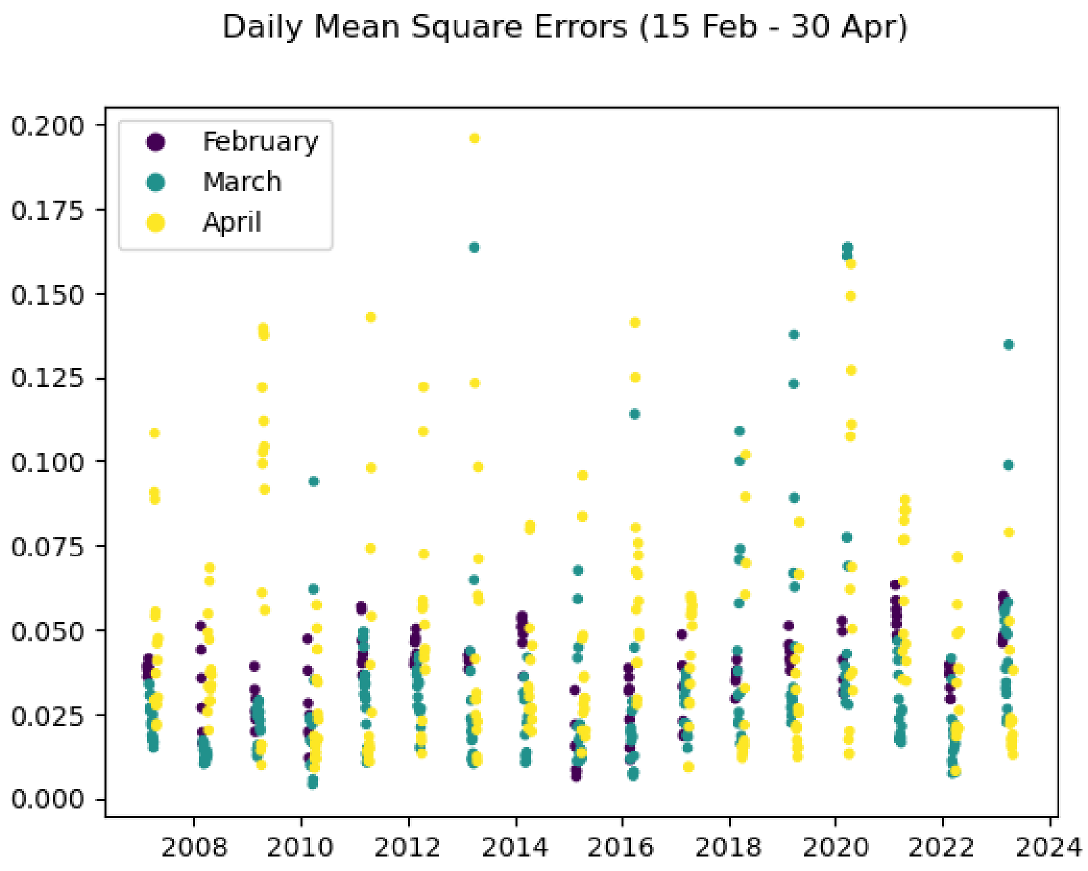
1.7 Other Years (Daily)

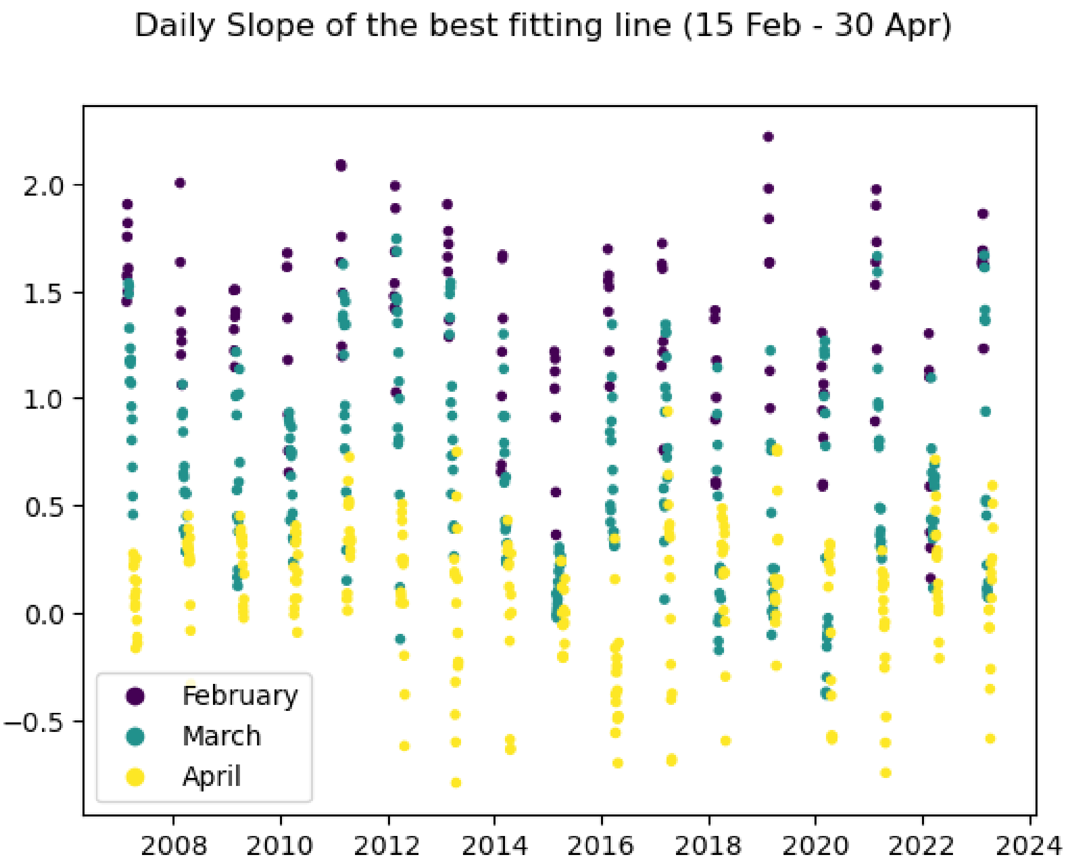
[ ]:

```python
r_all2 = np.array([])
rms_all2 = np.array([])
slope_all2 = np.array([])

for i in tqdm(range (0, len(ds.time_counter))):

    dataset = ds.isel(time_counter=i)

    drivers, diat, _ = datasets_preparation(dataset)

    r, rms, m = regressor3(drivers, diat)

    r_all2 = np.append(r_all2,r)
    rms_all2 = np.append(rms_all2,rms)
    slope_all2 = np.append(slope_all2,m)

plotting2(r_all2, 'Correlation Coefficients')
plotting2(rms_all2, 'Mean Square Errors')
plotting2(slope_all2, 'Slope of the best fitting line')
```

```
  0%|          | 0/640 [00:00<?, ?it/s]
```



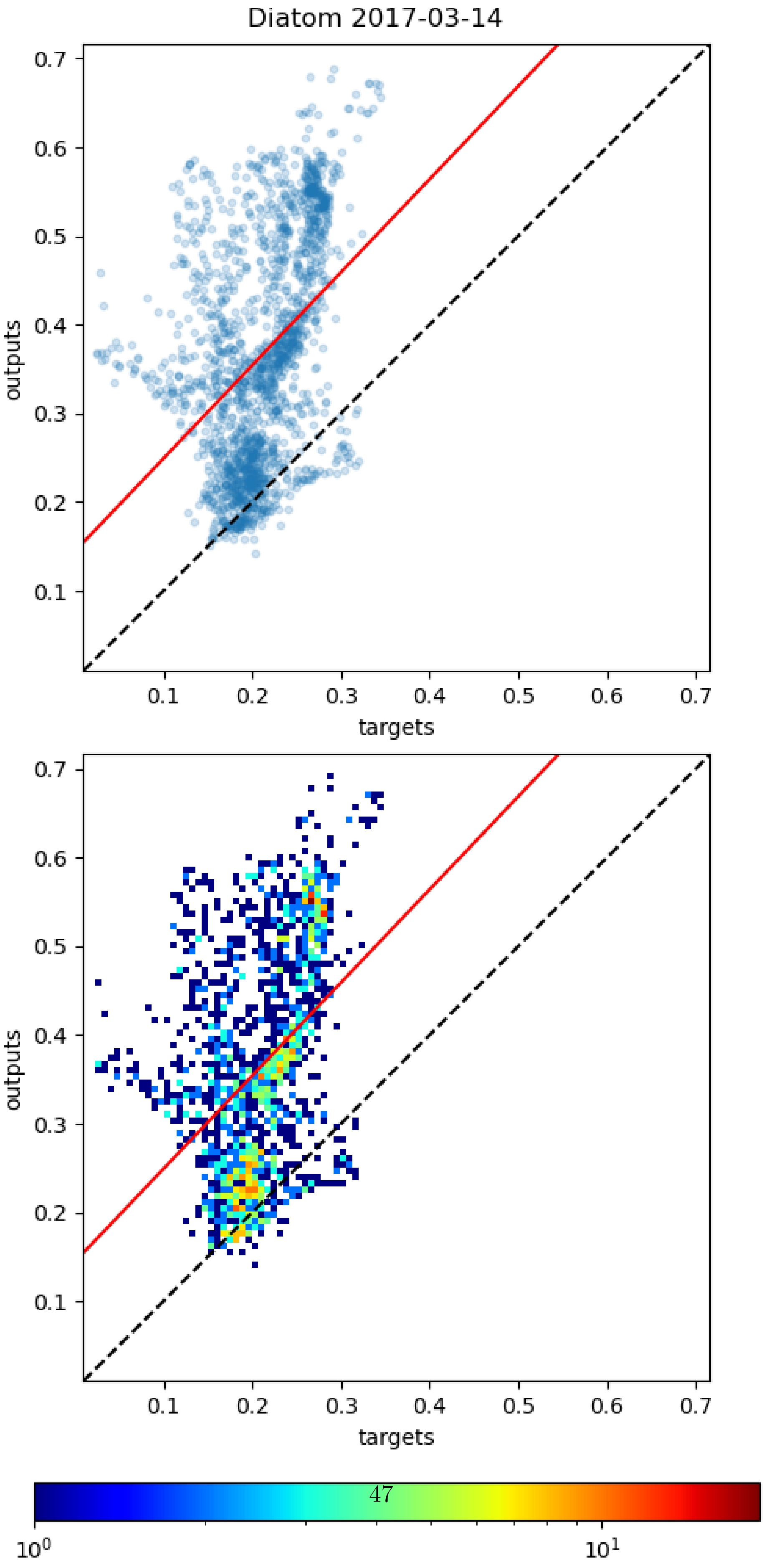

Daily Mean Square Errors (15 Feb - 30 Apr)
February
March
April
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
2008
2010
2012
2014
2016
2018
2020
2022
2024

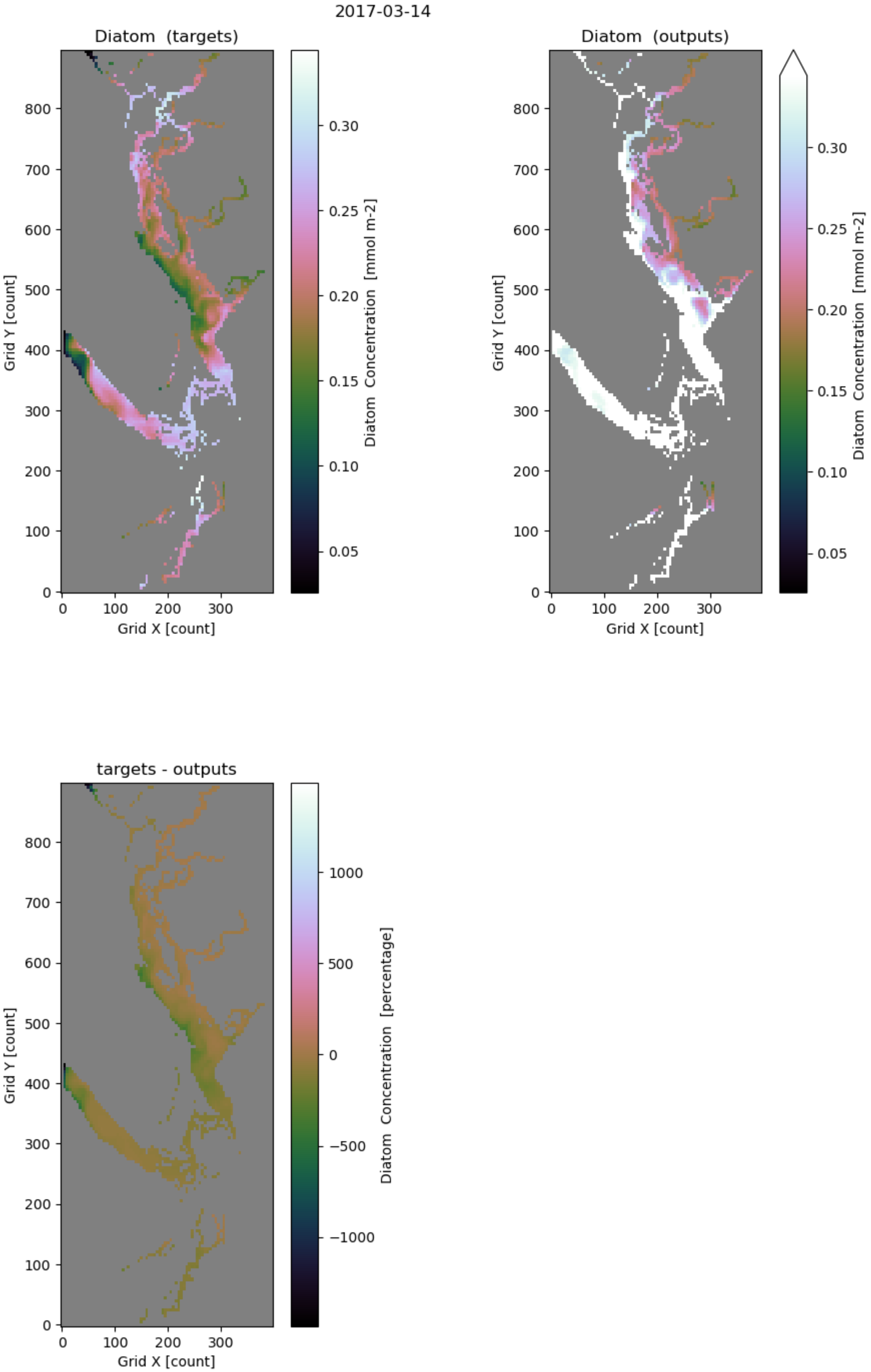


2 Daily Maps

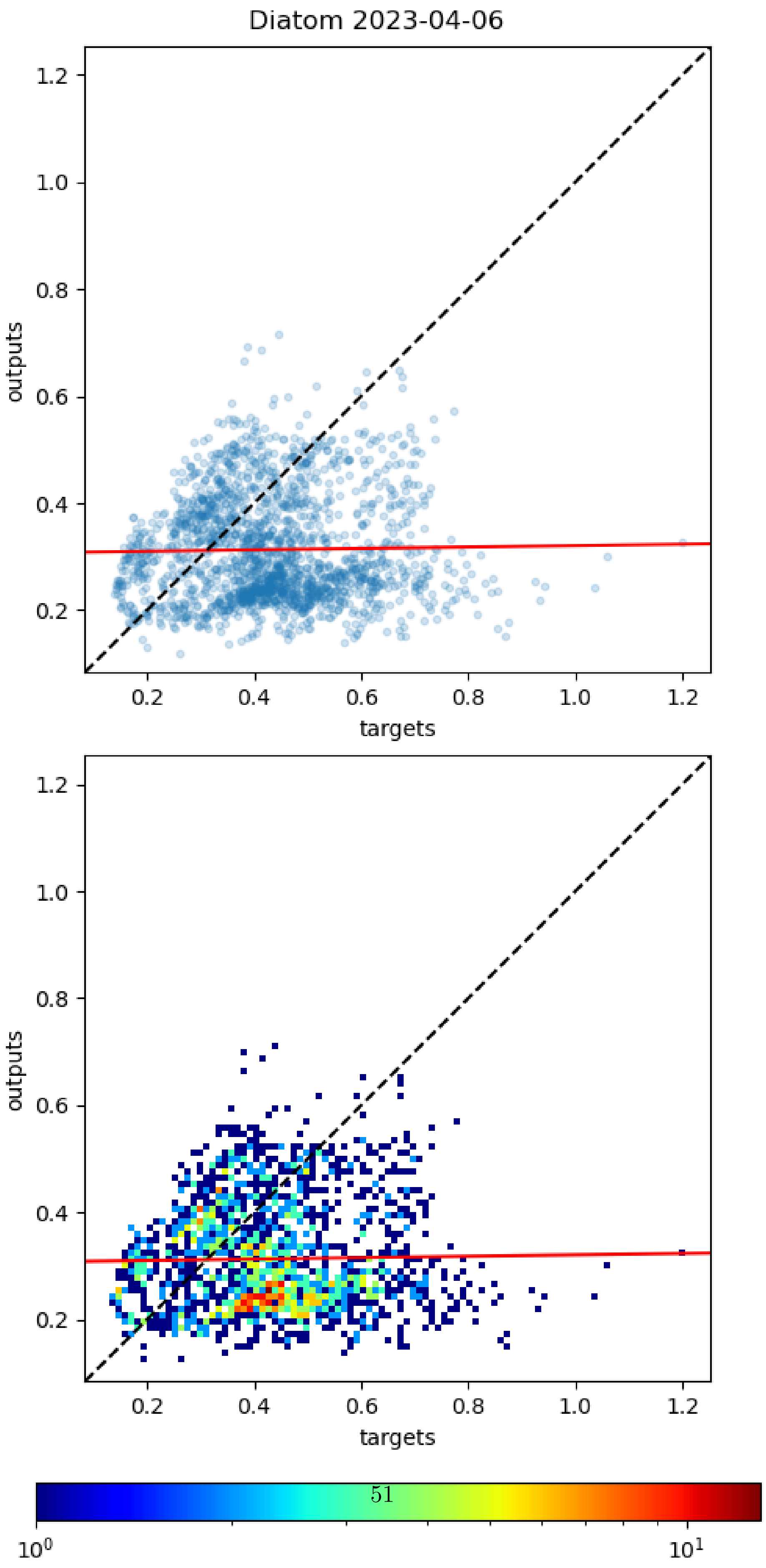
```
[ ]: maps = random.sample(range(0,len(ds.time_counter)),10)

     for i in tqdm(maps):

         dataset = ds.isel(time_counter=i)
         drivers, diat, indx = datasets_preparation(dataset)

         diat_i = dataset['Diatom']

         regressor4(drivers, diat, 'Diatom ')
```

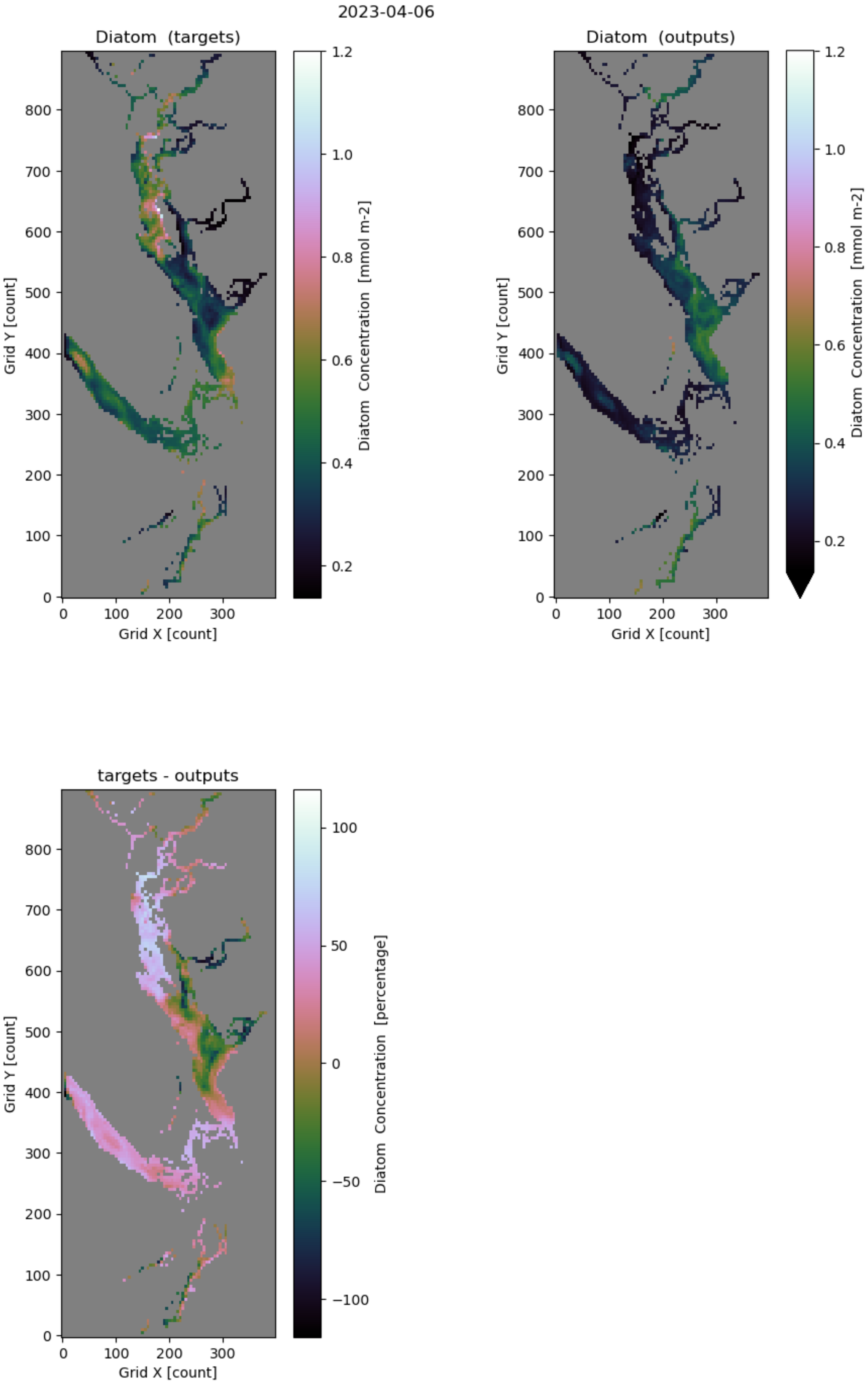
```
  0%|          | 0/10 [00:00<?, ?it/s]

The amount of data points is 1863
The slope of the best fitting line is  1.049
The correlation coefficient is: 0.408
 The mean square error is: 0.03732
```


Diatom 2017-03-14
0.7
0.6
0.5
0.4
0.3
0.2
0.1
outputs
0.1
0.2
0.3
0.4
0.5
0.6
0.7
targets
0.7
0.6
0.5
0.4
0.3
0.2
0.1
outputs
0.1
0.2
0.3
0.4
0.5
0.6
0.7
targets
10^0
10^1

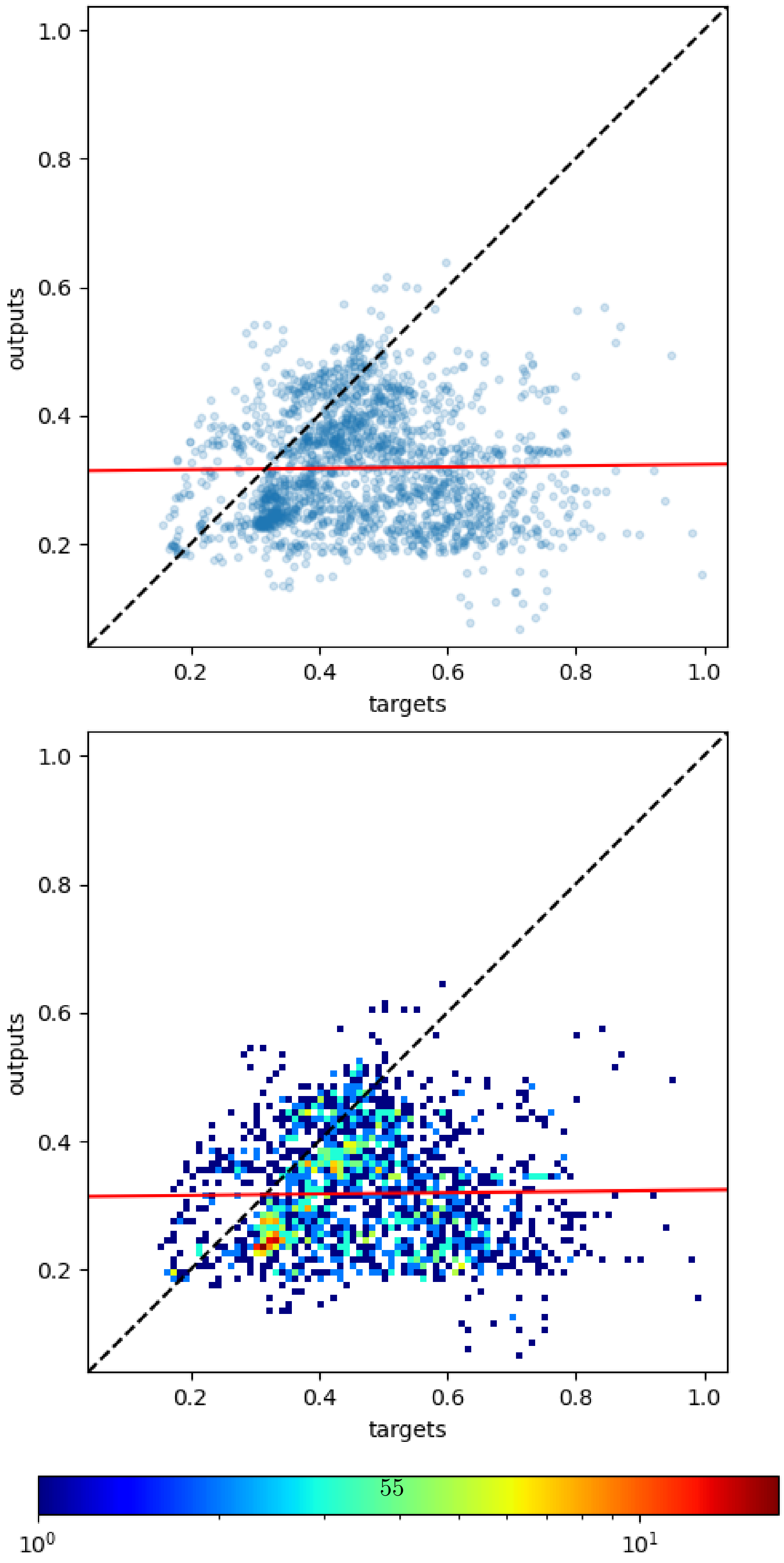

2017-03-14
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.05
0.10
0.15
0.20
0.25
0.30
Diatom Concentration [percentage]
−1000
−500
0
500
1000

```
The amount of data points is 1863
The slope of the best fitting line is  0.013
The correlation coefficient is: 0.019
 The mean square error is: 0.04392
```

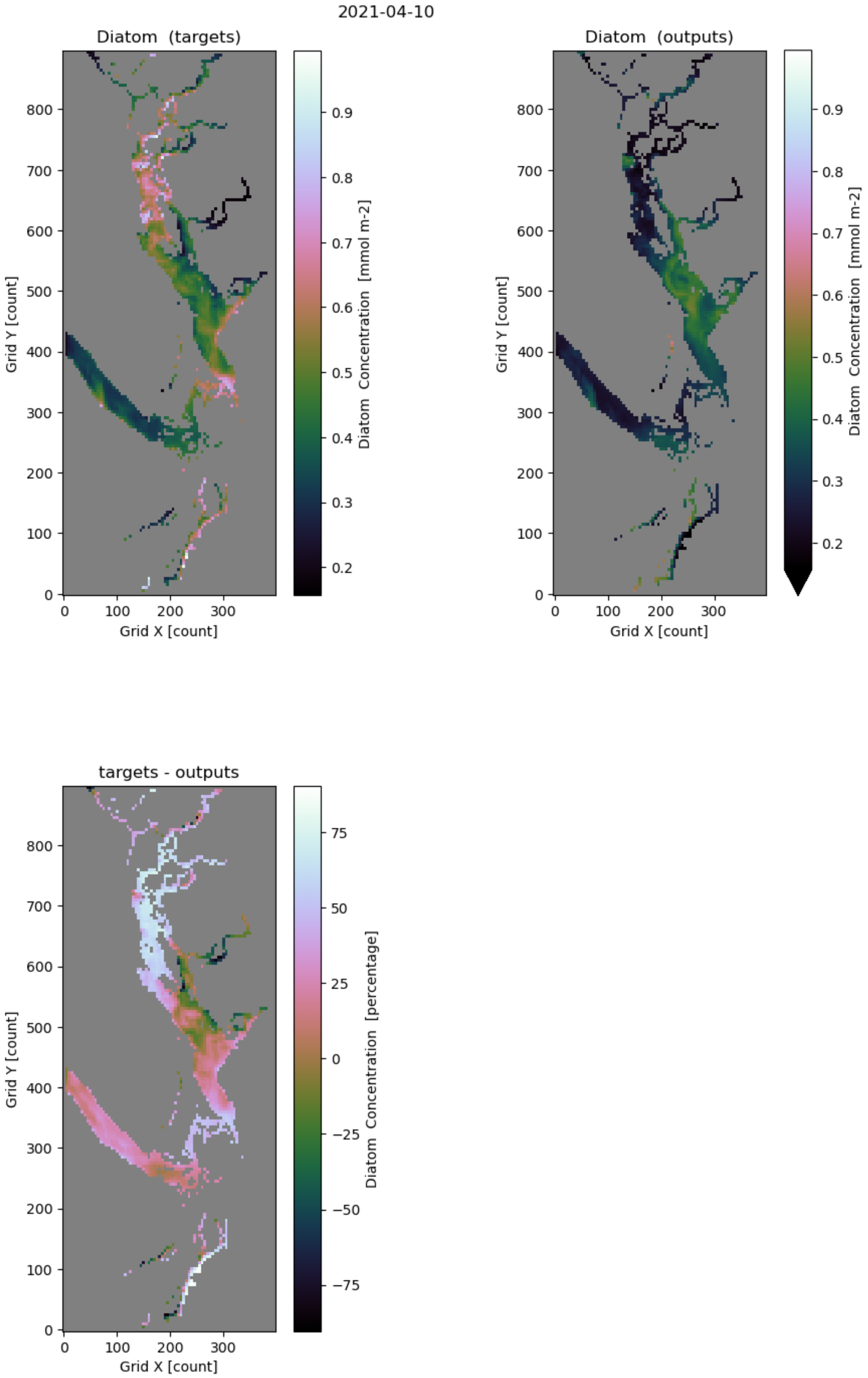

Diatom 2023-04-06
1.2
1.0
0.8
0.6
0.4
0.2
outputs
0.2
0.4
0.6
0.8
1.0
1.2
targets
1.2
1.0
0.8
0.6
0.4
0.2
outputs
0.2
0.4
0.6
0.8
1.0
1.2
targets
10^0
10^1


2023-04-06
Diatom (targets)
Diatom (outputs)
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.2
0.4
0.6
0.8
1.0
1.2
targets - outputs
Diatom Concentration [percentage]
−100
−50
0
50
100

```
The amount of data points is 1863
The slope of the best fitting line is  0.01
The correlation coefficient is: 0.016
 The mean square error is: 0.04869
```



Diatom 2021-04-10



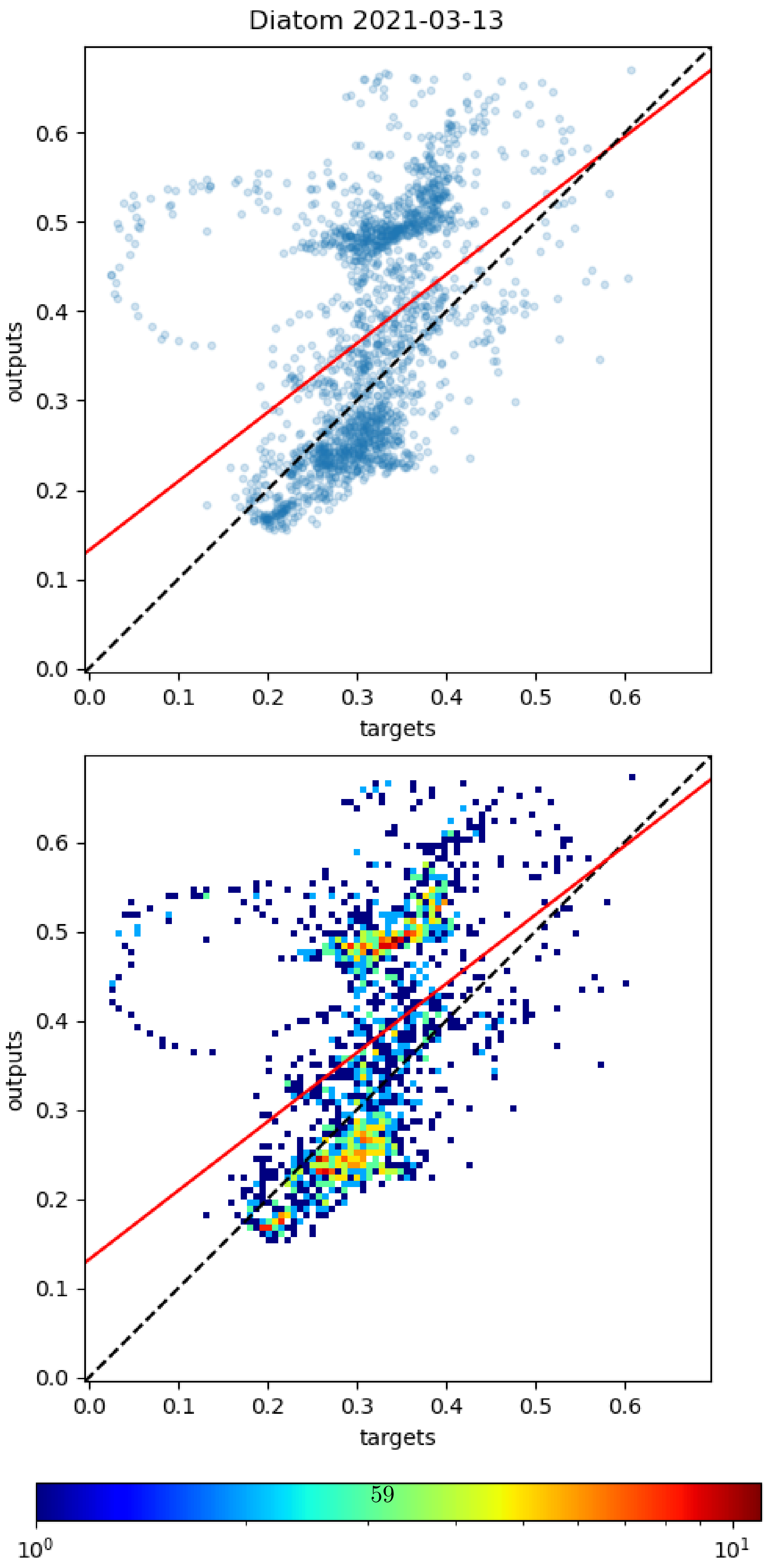
2021-04-10

Diatom (targets)

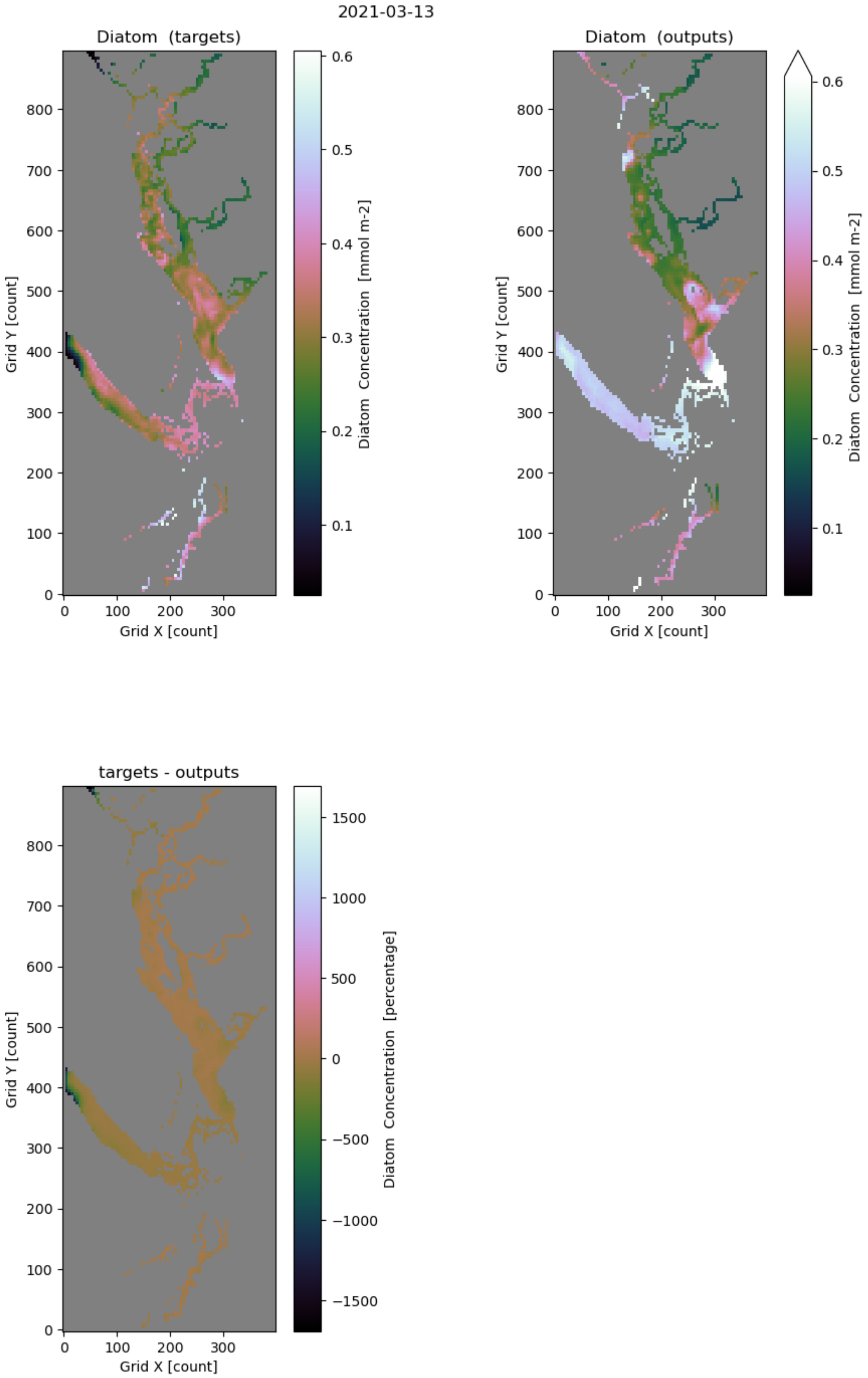
Diatom (outputs)

targets - outputs

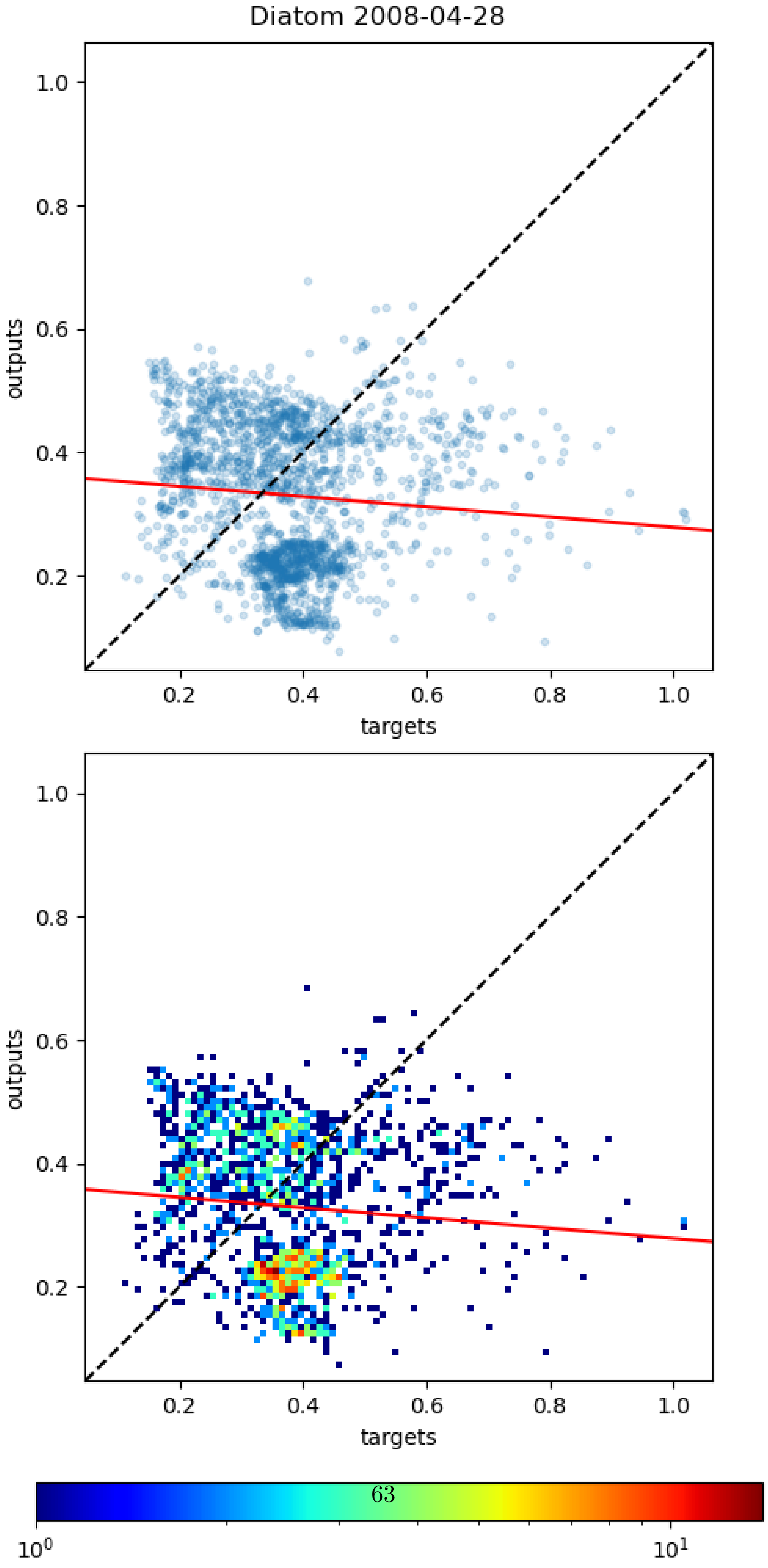
```
The amount of data points is 1863
The slope of the best fitting line is  0.773
The correlation coefficient is: 0.45
 The mean square error is: 0.0174
```

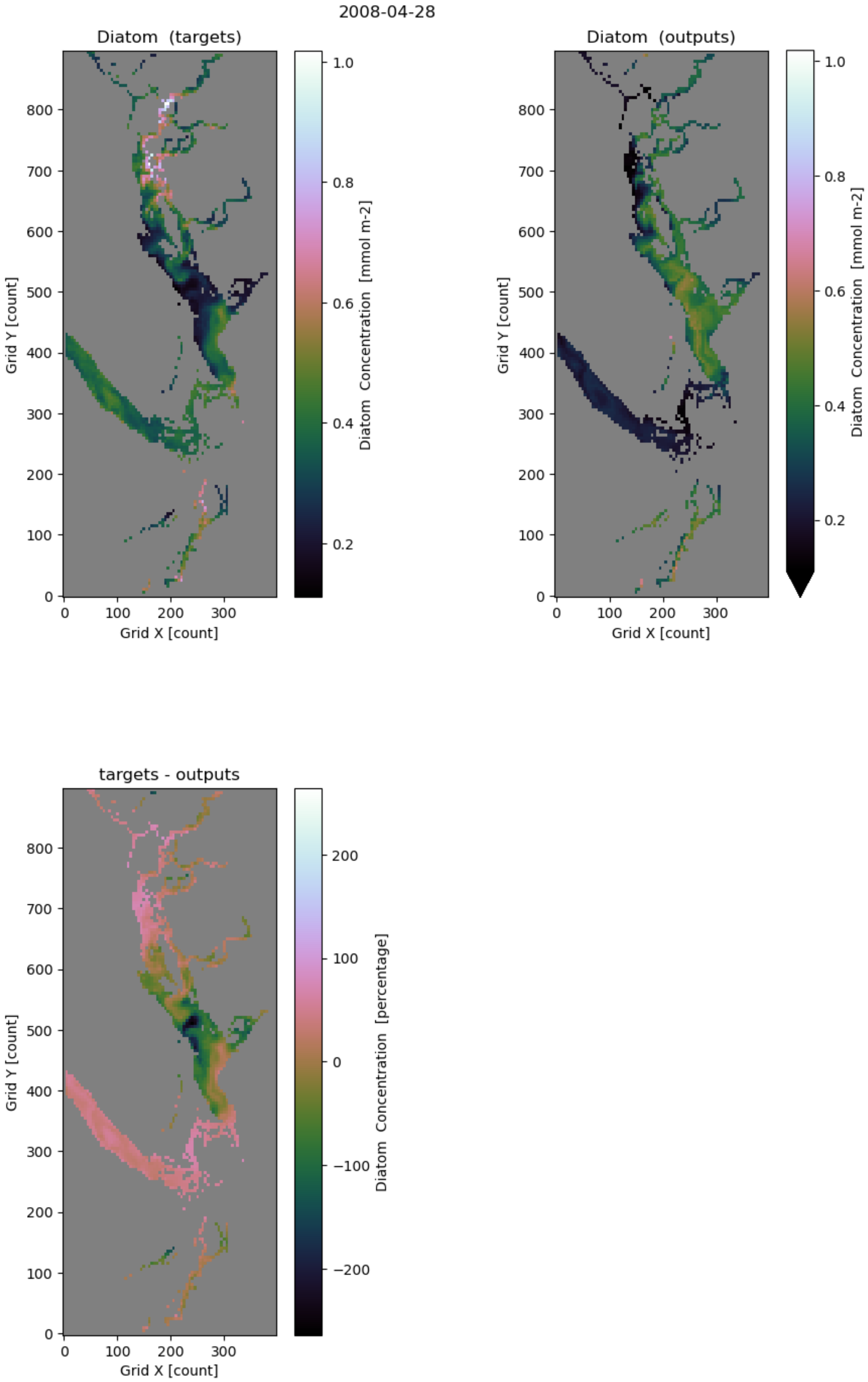


Diatom 2021-03-13

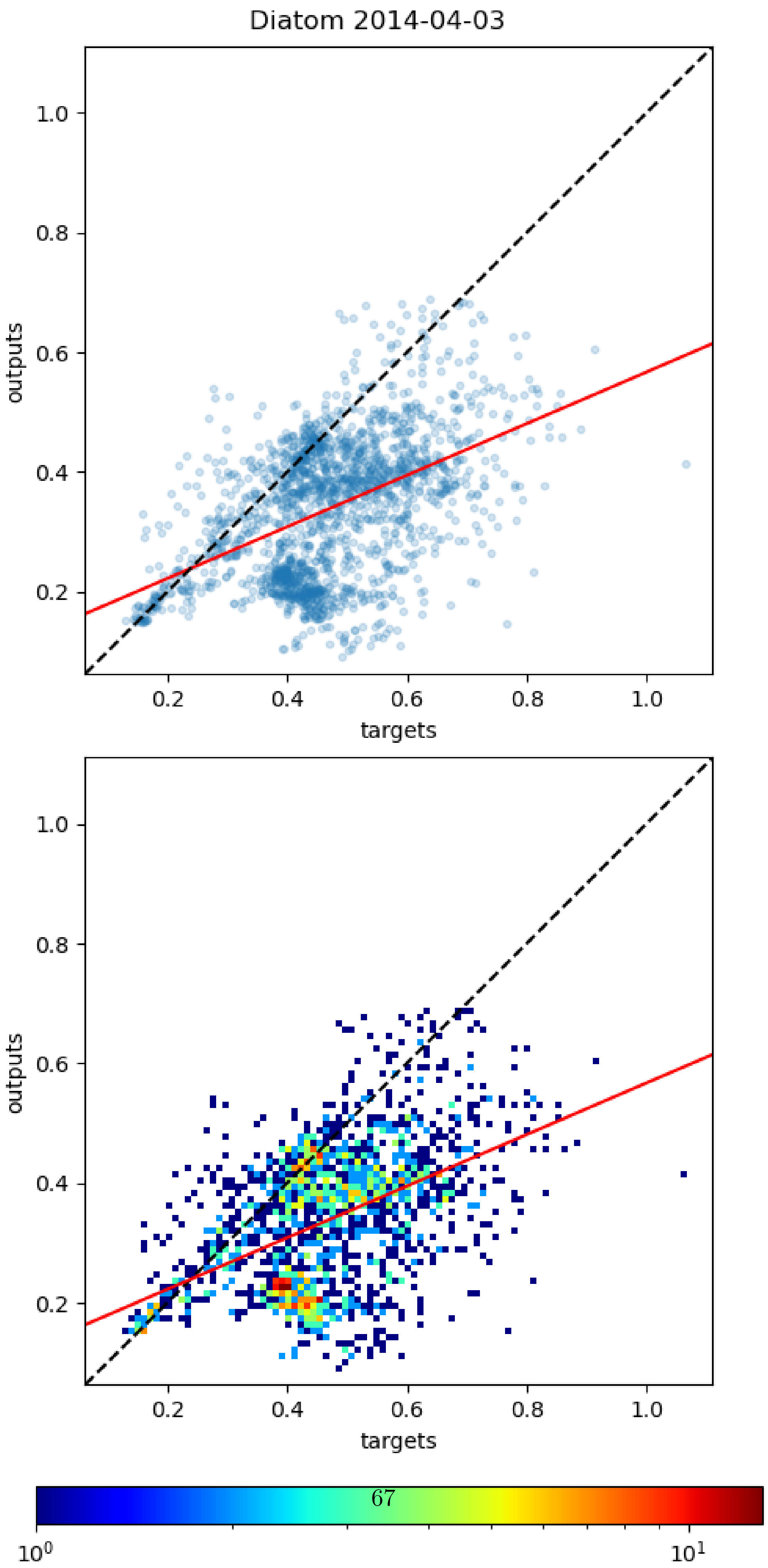

2021-03-13
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.1
0.2
0.3
0.4
0.5
0.6
Diatom Concentration [percentage]
−1500
−1000
−500
0
500
1000
1500

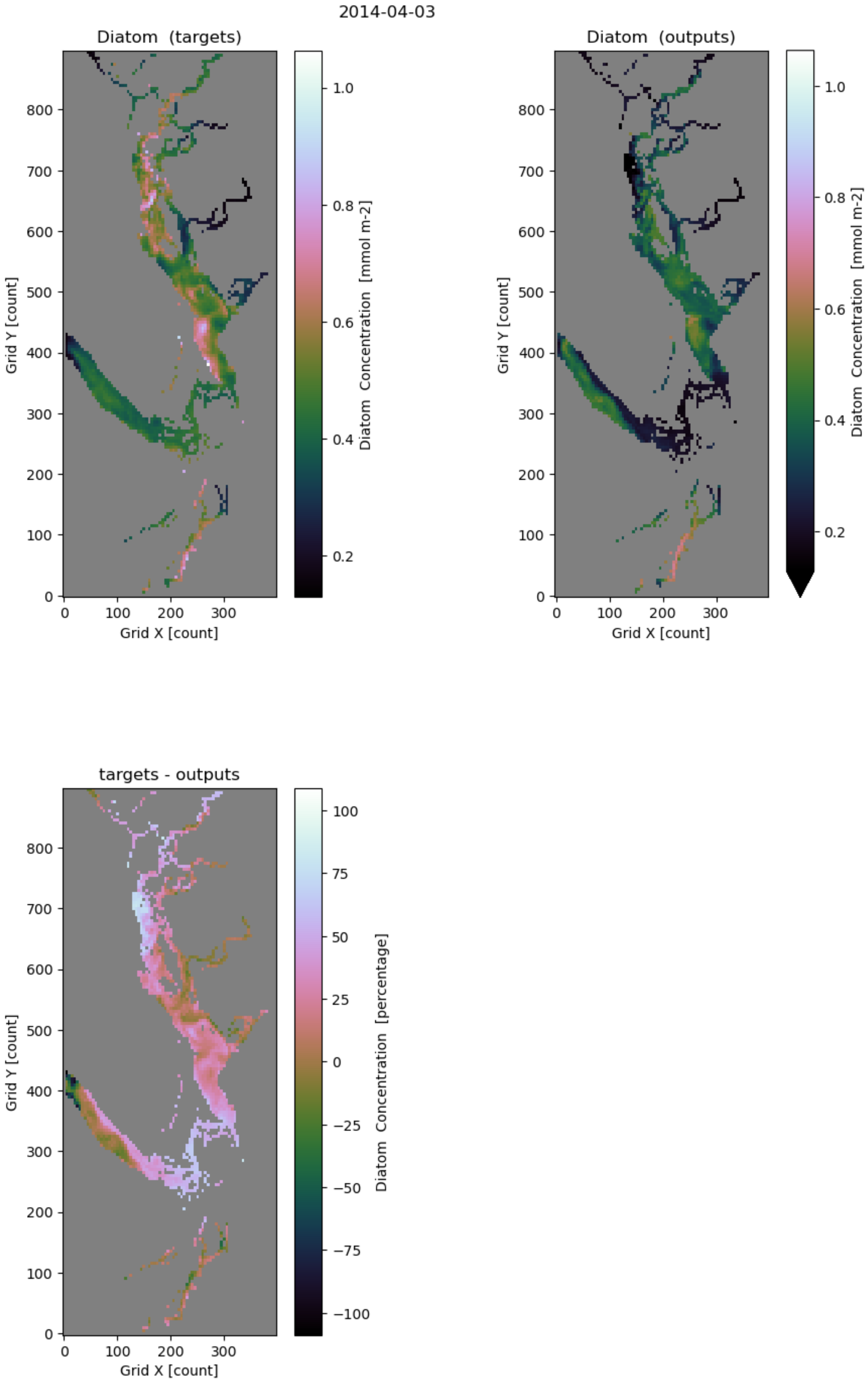
```
The amount of data points is 1863
The slope of the best fitting line is  -0.083
The correlation coefficient is: -0.09
 The mean square error is: 0.03353
```


Diatom 2008-04-28
outputs
targets
0.2
0.4
0.6
0.8
1.0
outputs
targets
0.2
0.4
0.6
0.8
1.0
10^0
10^1

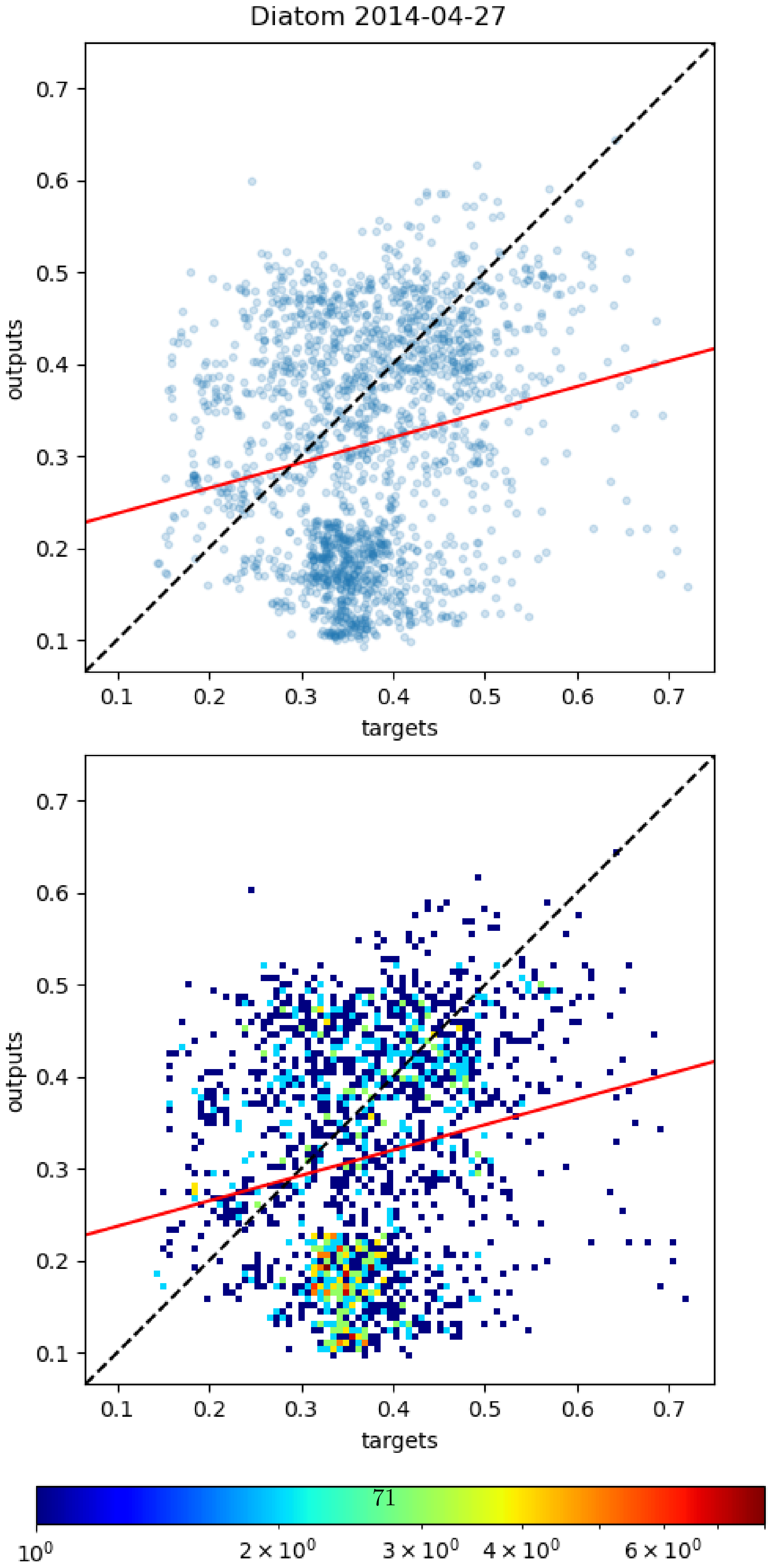

2008-04-28
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.2
0.4
0.6
0.8
1.0
Diatom Concentration [percentage]
−200
−100
0
100
200

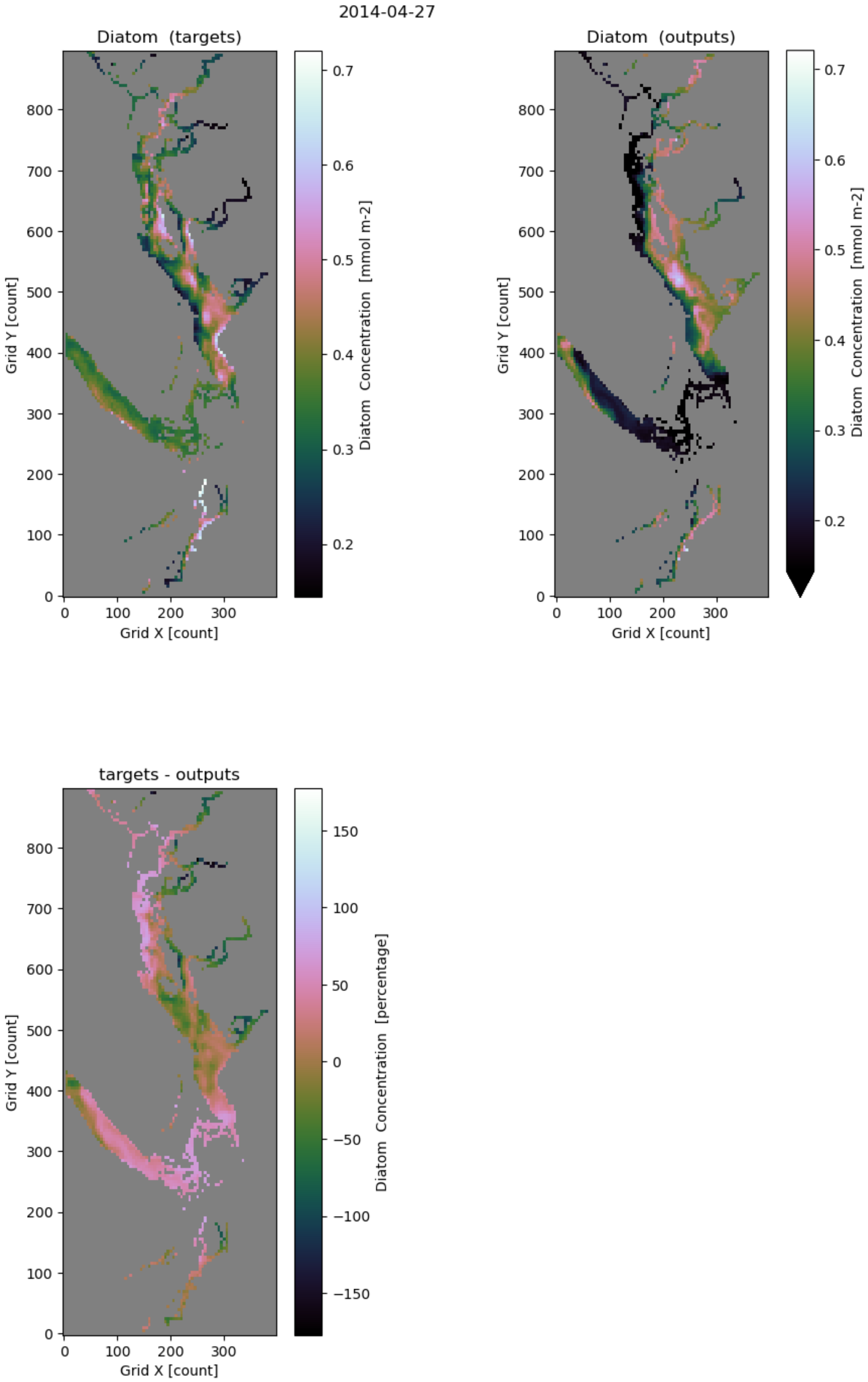
```
The amount of data points is 1863
The slope of the best fitting line is  0.431
The correlation coefficient is: 0.479
 The mean square error is: 0.0332
```


Diatom 2014-04-03
1.0
0.8
0.6
0.4
0.2
outputs
0.2
0.4
0.6
0.8
1.0
targets
1.0
0.8
0.6
0.4
0.2
outputs
0.2
0.4
0.6
0.8
1.0
targets
10^0
10^1

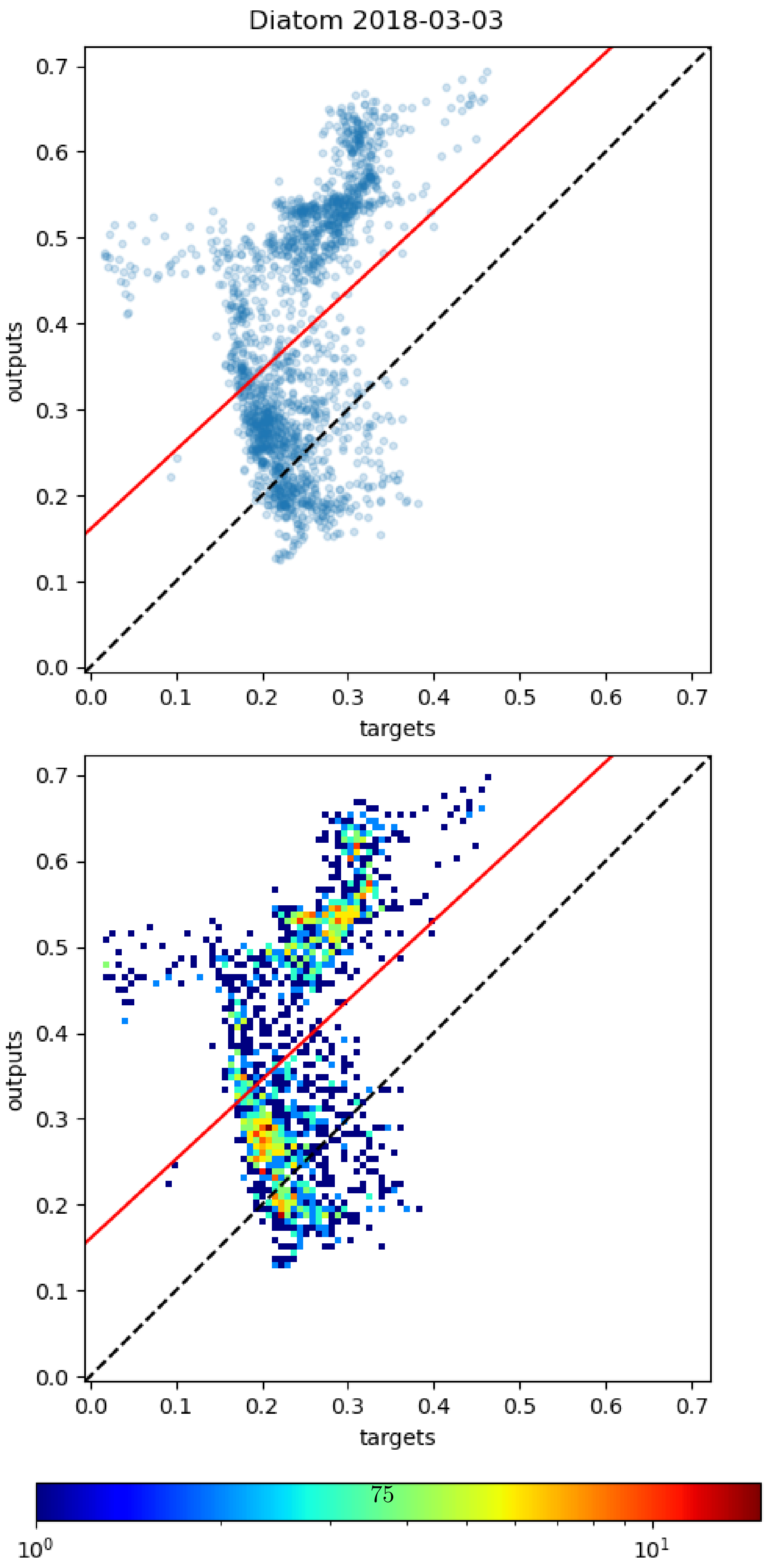

2014-04-03
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.2
0.4
0.6
0.8
1.0
Diatom Concentration [percentage]
−100
−75
−50
−25
0
25
50
75
100

```
The amount of data points is 1863
The slope of the best fitting line is  0.276
The correlation coefficient is: 0.199
 The mean square error is: 0.02332
```

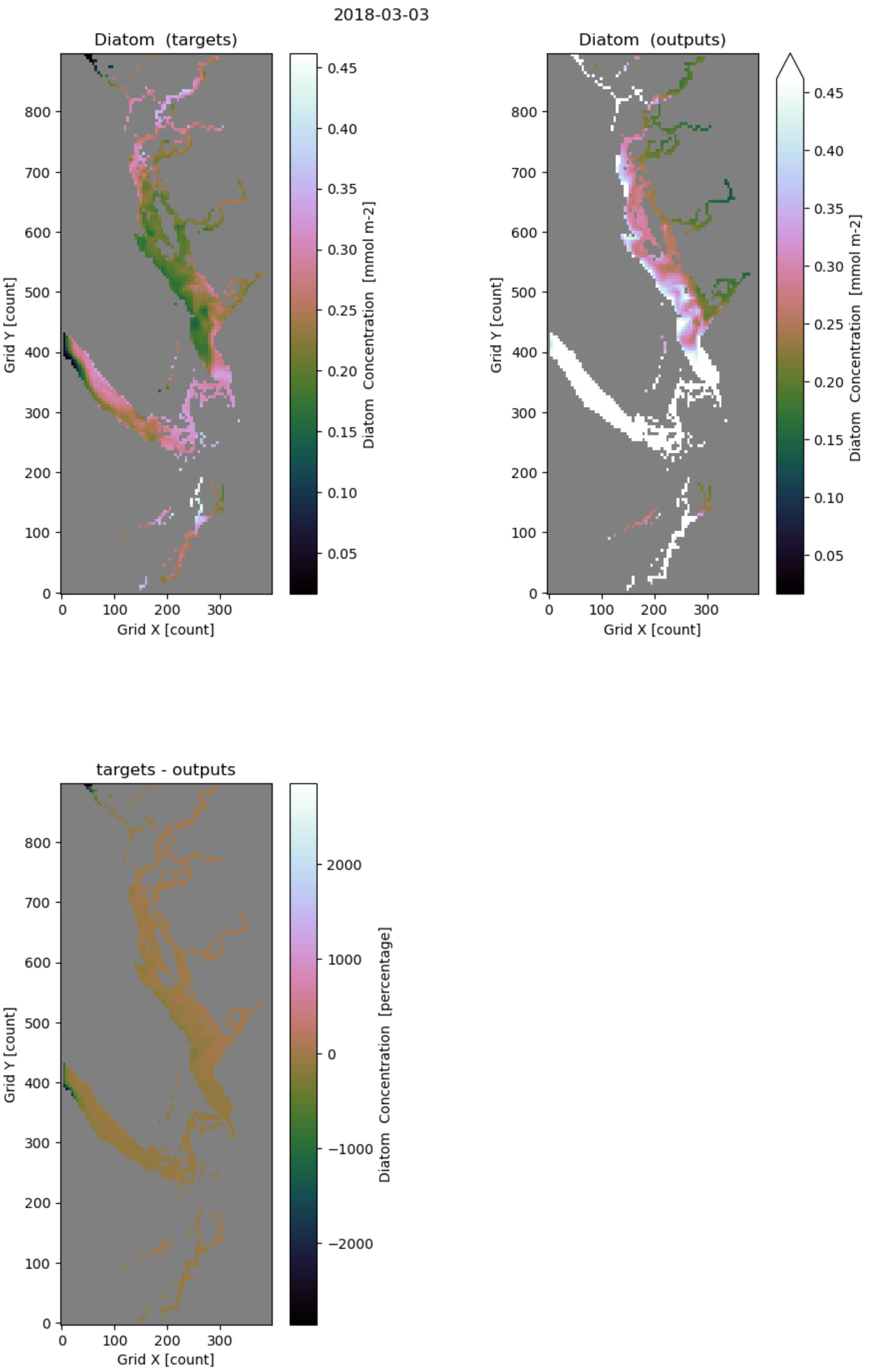

Diatom 2014-04-27
outputs
0.1
0.2
0.3
0.4
0.5
0.6
0.7
targets
0.1
0.2
0.3
0.4
0.5
0.6
0.7
outputs
0.1
0.2
0.3
0.4
0.5
0.6
0.7
targets
0.1
0.2
0.3
0.4
0.5
0.6
0.7
10^0
2×10^0
3×10^0
4×10^0
6×10^0


2014-04-27
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.2
0.3
0.4
0.5
0.6
0.7
Diatom Concentration [percentage]
−150
−100
−50
0
50
100
150

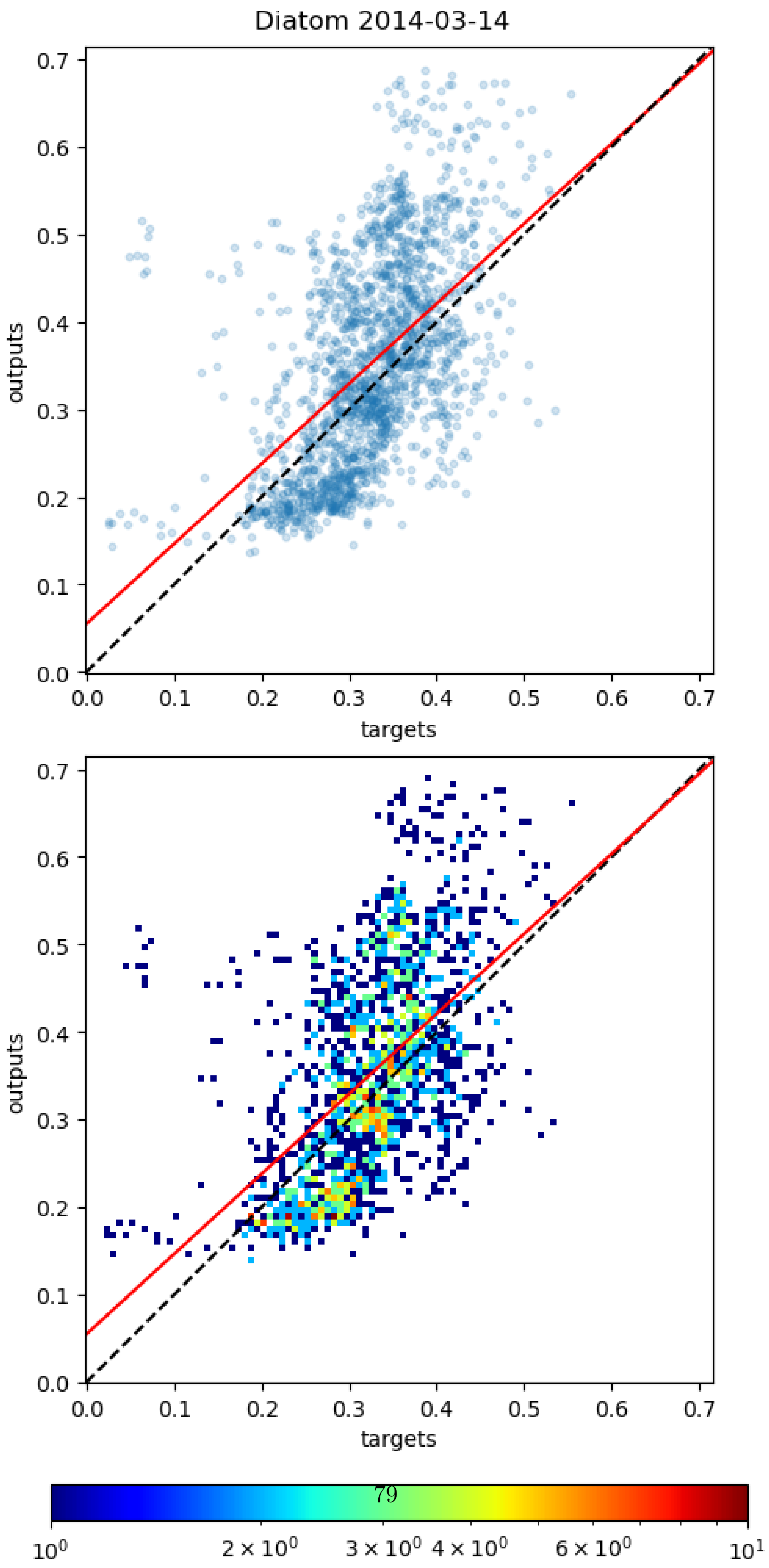
```
The amount of data points is 1863
The slope of the best fitting line is  0.925
The correlation coefficient is: 0.379
 The mean square error is: 0.03772
```

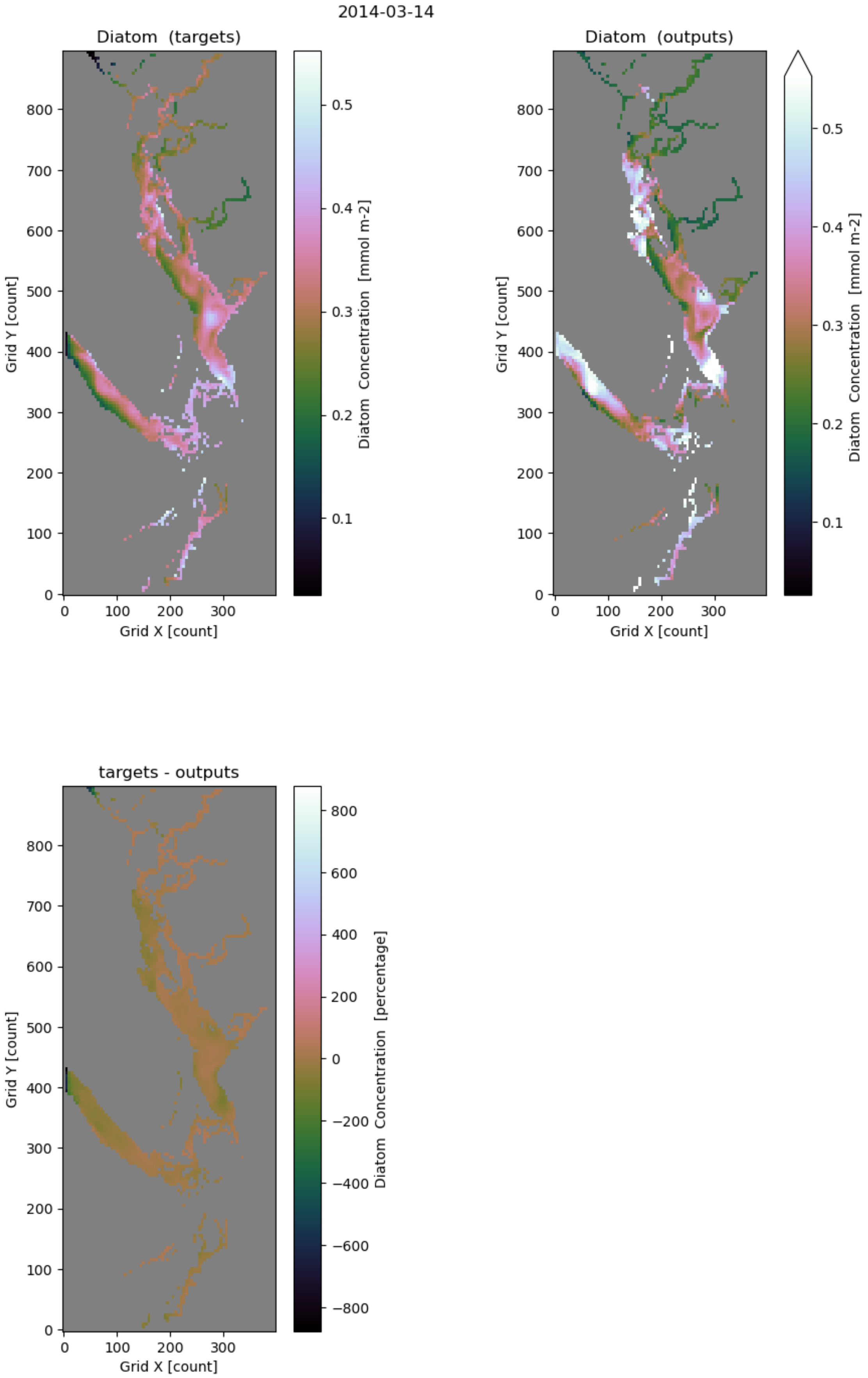


Diatom 2018-03-03

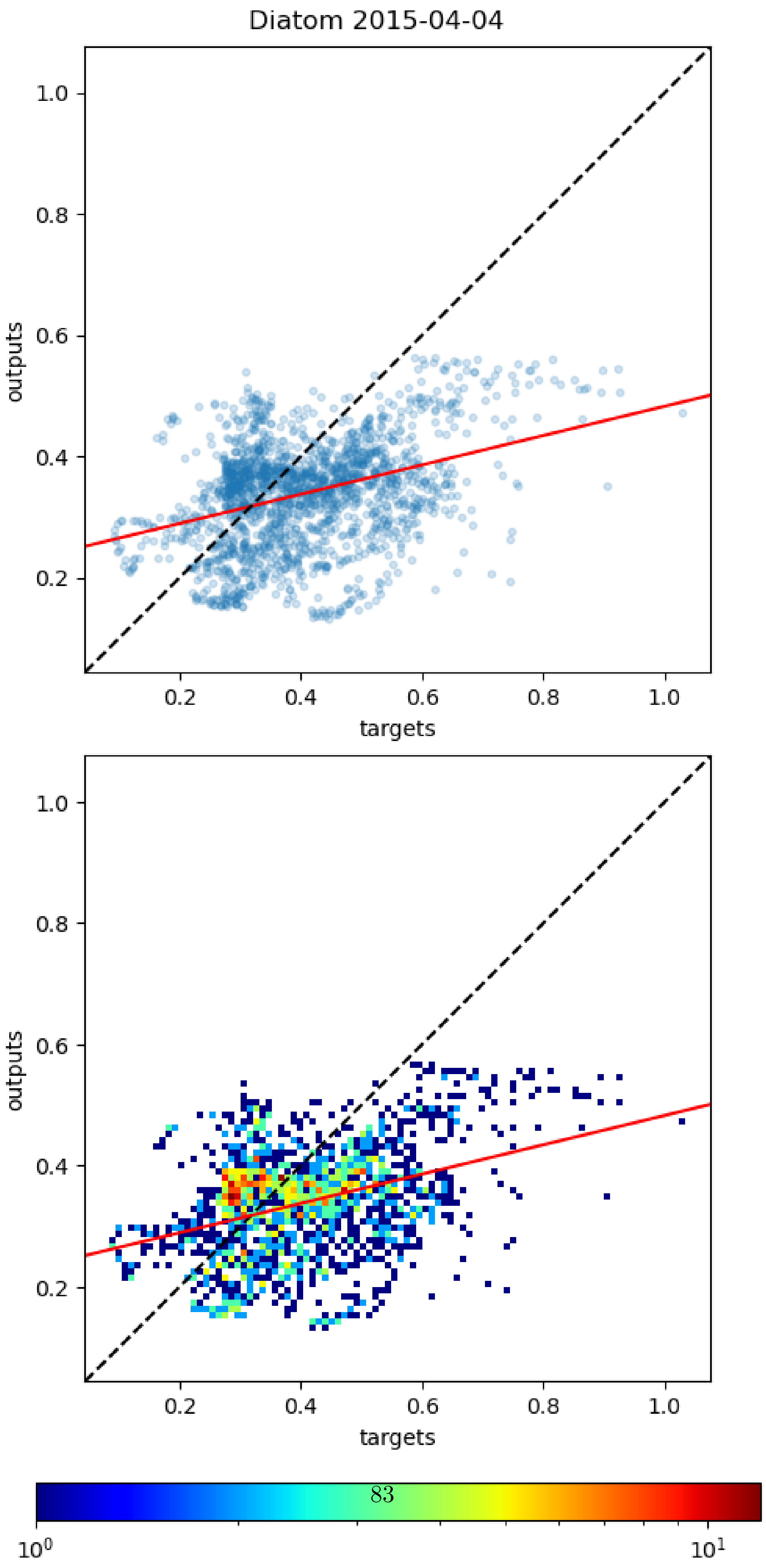

2018-03-03
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
Diatom Concentration [mmol m-2]
−2000
−1000
0
1000
2000
Diatom Concentration [percentage]

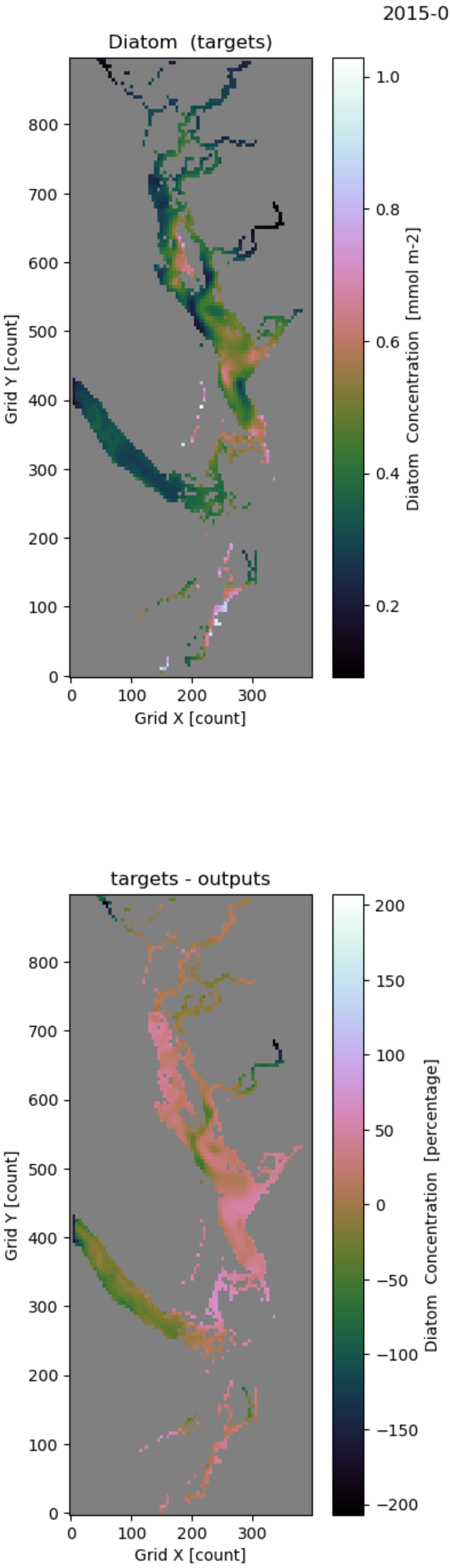
```
The amount of data points is 1863
The slope of the best fitting line is  0.915
The correlation coefficient is: 0.545
 The mean square error is: 0.01061
```


Diatom 2014-03-14
outputs
targets
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
10^0
2×10^0
3×10^0
4×10^0
6×10^0
10^1


2014-03-14
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
0
100
200
300
400
500
600
700
800
Grid X [count]
0
100
200
300
Diatom Concentration [mmol m-2]
0.1
0.2
0.3
0.4
0.5
Diatom Concentration [percentage]
−800
−600
−400
−200
0
200
400
600
800

```
The amount of data points is 1863
The slope of the best fitting line is  0.242
The correlation coefficient is: 0.367
 The mean square error is: 0.02019
```


Diatom 2015-04-04
outputs
1.0
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
1.0
targets
outputs
1.0
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
1.0
targets
10^0
10^1


2015-04-04
Diatom (targets)
Diatom (outputs)
targets - outputs
Grid Y [count]
Grid X [count]
0
100
200
300
400
500
600
700
800
Diatom Concentration [mmol m-2]
0.2
0.4
0.6
0.8
1.0
Diatom Concentration [percentage]
−200
−150
−100
−50
0
50
100
150
200

[ ]:

```
```