

LTE + Log-normal PDF

```
In [72]: 1 M_H2 = np.sum(m_H2_bar)/M_sun
2 print M_H2/1e9
```

2.8471647994275058

```
In [73]: 1 #using velocity dispersion from sim
2 L_CO_SI_1 = np.sum(l_CO_SI_bar_1*(cell_width_arr**3))
3 print L_CO_SI_1/1e8
```

19.4797439169398

```
In [74]: 1 #using isothermal sound speed
2 L_CO_SI_2 = np.sum(l_CO_SI_bar_2*(cell_width_arr**3))
3 print L_CO_SI_2/1e8
```

0.7345739542543737

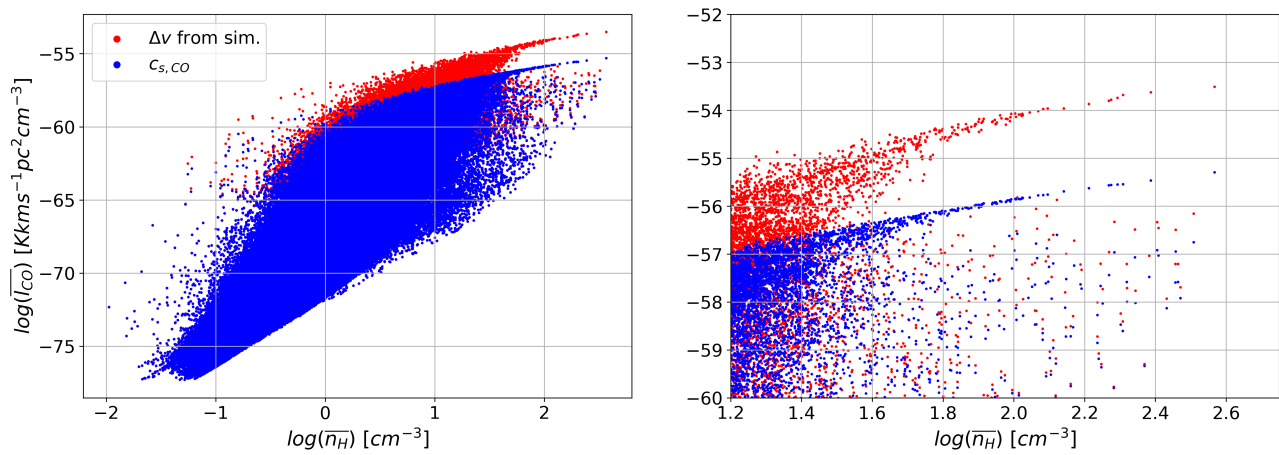
```
In [75]: 1 #using velocity dispersion from sim
2 alpha_CO_1 = L_CO_SI_1/M_H2
3 print alpha_CO_1
```

0.6841804141740124

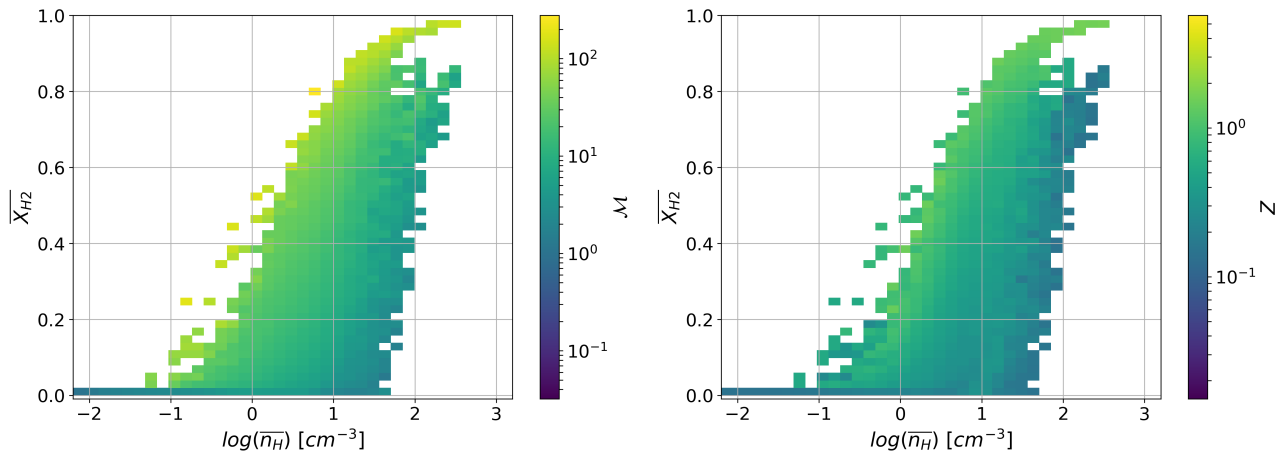
```
In [76]: 1 #using isothermal sound speed
2 alpha_CO_2 = L_CO_SI_2/M_H2
3 print alpha_CO_2
```

0.025800190926850392

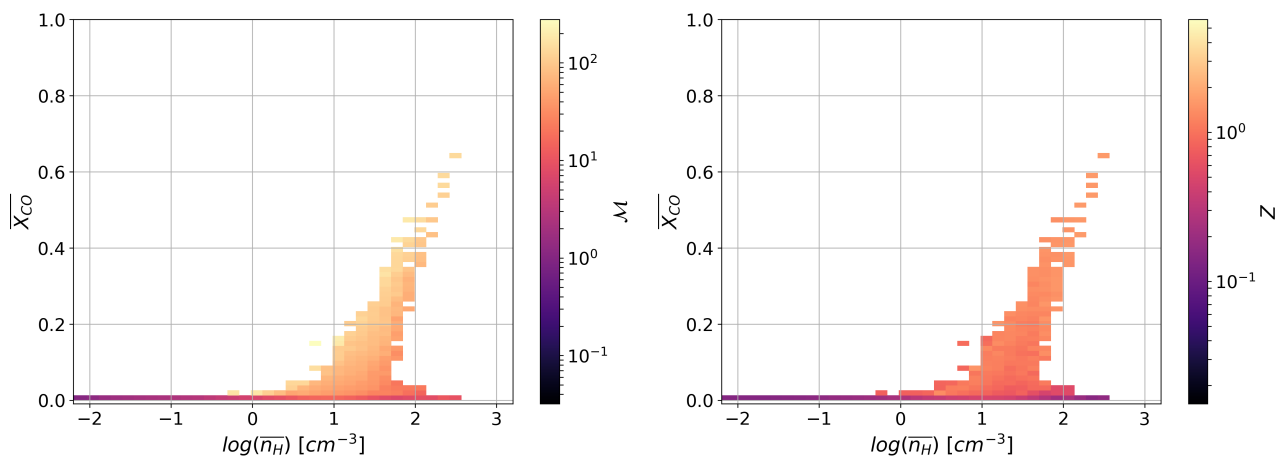
LTE + Log-normal PDF



LTE + Log-normal PDF



LTE + Log-normal PDF



LTE + Power-law PDF

```
In [25]: 1 M_H2 = np.sum(m_H2_bar)/M_sun
2 print M_H2/1e9
```

2.8184595847244887

```
In [26]: 1 #using velocity dispersion from sim
2 L_CO_SI_1 = np.sum(l_CO_SI_bar_1*(cell_width_arr**3))
3 print L_CO_SI_1/1e8
```

10.585428671137132

```
In [27]: 1 #using isothermal sound speed
2 L_CO_SI_2 = np.sum(l_CO_SI_bar_2*(cell_width_arr**3))
3 print L_CO_SI_2/1e8
```

0.6103197747485133

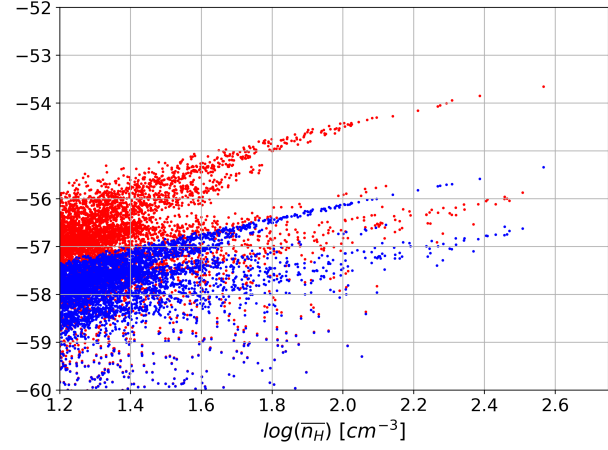
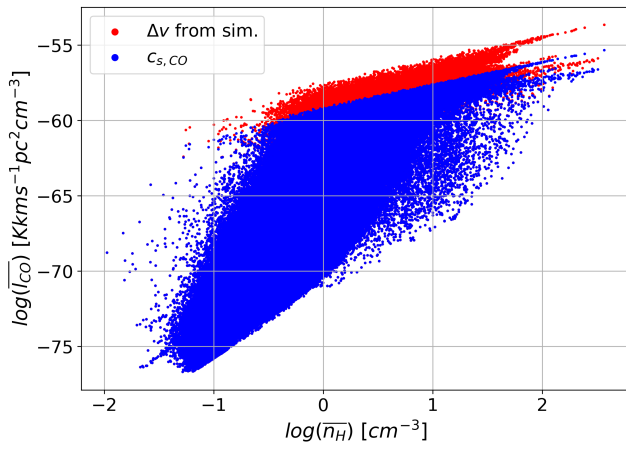
```
In [28]: 1 #using velocity dispersion from sim
2 alpha_CO_1 = L_CO_SI_1/M_H2
3 print alpha_CO_1
```

0.3755749675641307

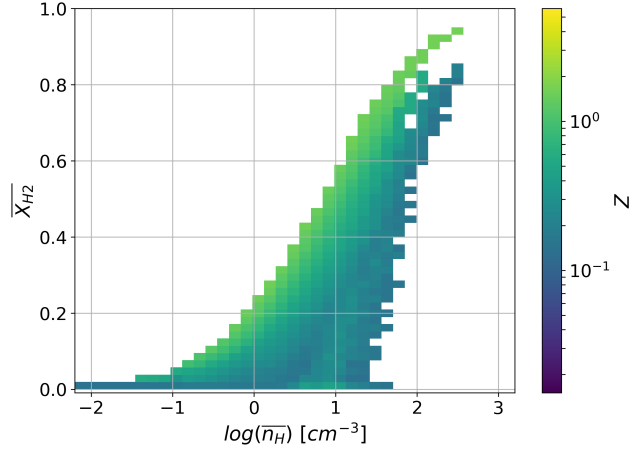
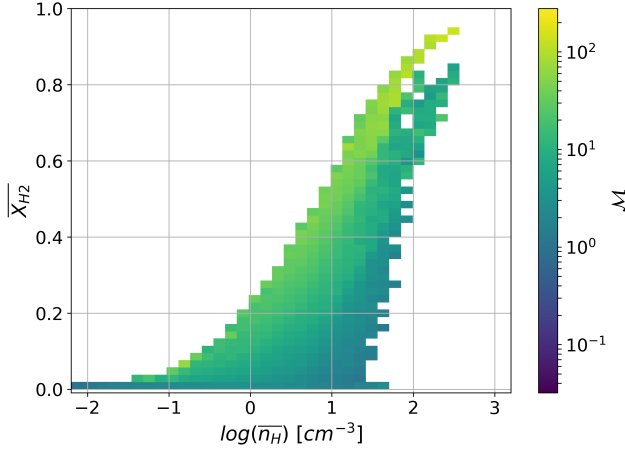
```
In [29]: 1 #using isothermal sound speed
2 alpha_CO_2 = L_CO_SI_2/M_H2
3 print alpha_CO_2
```

0.02165437383088016

LTE + Power-law PDF



LTE + Power-law PDF



LTE + Power-law PDF

