If you use data provided here please cite Huber et al. (2019) and Huber et al. (2021).

All lensed type Ia supernovae (LSNe Ia) systems for which microlensed light curves and spectra are available are listed in Table 1. For each system and image number 40000 spectra, respectively light curve sets are available coming from four theoretical type Ia supernova (SN Ia) models and 10000 random microlensing configurations. A light curve set contains the six LSST filters $u, g, r, i, z,$ and y, as well as the two infrared bands J and H. The microlensing calculation is in detail described in (Huber et al. 2019), where the synthetic observables of SNe Ia have been calculated using the radiative transfer code `ARTIS` (Kromer & Sim 2009) for the theoretical models W7 (Nomoto et al. 1984), N100 (Seitenzahl et al. 2013), a sub-Chandrasekhar (sub-Ch) detonation model (Sim et al. 2010), and a merger model (Pakmor et al. 2012). These models have also been used in Suyu et al. (2020) and Huber et al. (2020) for studies on LSNe Ia. In the data respectively code the SN models will be referred to as "me" for the merger model, "n1" for the N100 model, "su" for the sub-Ch model, and "ww" for the W7 model. To calculate the microlensing effect we use magnification maps from `GERLUMPH` (Vernardos et al. 2014; Vernardos & Fluke 2014; Vernardos et al. 2015) where we followed Chan et al. (2021).

1. Usage of light curves

Things to download:

- python script "public_spectra_light_curve.py" from `https://github.com/shsuyu/HOLISMOKES-public/tree/main/HOLISMOKES_VII`
- folder "light_curves.tar" from `https://datashare.mpcdf.mpg.de/s/TgneNzOAAPYAK6h` (we recommend to store the folder in another folder called "data_release_holismokes7")

Steps to do:

- unpack light curves folder using "tar -xf light_curves.tar" in the terminal
- open the code "public_spectra_light_curve.py"
- set the "input_data_path" to the path where you stored the "light_curves" folder
- pick your $N_{\rm sys}$ and $N_{\rm im}$ from Table 1, as well as the supernovae_model (options "me", "n1", "su", and "ww")
- pick a random microlensing event by using for micro_config values from 0 to 9999
- use the functions f_load_microlensed_lightcurve and f_load_macrolensed_lightcurve for microlensed or macrolensed light curves
- specify the filter (parameter filter_) you are interested in, where you can choose from $u, g, r, i, z, y, J,$ and H
- run the code using python2.7

Output and additional information:

The two light curve functions will return the time values and the corresponding magnitude values. The time is the time after explosion in the observer frame. Magnitude values are measured in the AB system. The macrolensed light curves take into account

system number $N_{\rm sys}$	image number $N_{\rm im}$	κ	γ	s	$z_{\rm s}$	$z_{\rm d}$
1	1	0.250895	0.274510	0.6	0.76	0.252
	2	0.825271	0.814777	0.6	0.76	0.252
2	1	0.250895	0.274510	0.6	0.55	0.252
	2	0.825271	0.814777	0.6	0.55	0.252
3	1	0.250895	0.274510	0.6	0.99	0.252
	2	0.825271	0.814777	0.6	0.99	0.252
4	1	0.250895	0.274510	0.6	0.76	0.16
	2	0.825271	0.814777	0.6	0.76	0.252
5	1	0.250895	0.274510	0.6	0.76	0.48
	2	0.825271	0.814777	0.6	0.76	0.252
6	1	0.250895	0.274510	0.3	0.76	0.252
	2	0.825271	0.814777	0.3	0.76	0.252
7	1	0.250895	0.274510	0.59	0.76	0.252
	2	0.825271	0.814777	0.59	0.76	0.252
8	1	0.250895	0.274510	0.9	0.76	0.252
	2	0.825271	0.814777	0.9	0.76	0.252
9	1	0.434950	0.414743	0.6	0.76	0.252
	2	0.431058	0.423635	0.6	0.76	0.252
	3	0.566524	0.536502	0.6	0.76	0.252
	4	1.282808	1.252791	0.6	0.76	0.252

Table 1: Lensed supernova Ia systems for which microlensed spectra and light curves are available.

the magnification factor $\mu = \frac{1}{(1-\kappa)^2-\gamma^2}$, where all microlensing cases contain magnification from macrolensing and microlensing. All light curves are calculated for the source redshift listed in Table 1 and are therefore in the observer frame. If you are interested in different source redshifts please look at Section 3. Despite microlensing there are no other sources of uncertainties taken into account. If you like to calculate observational noise for the light curves you can check Section 2.2 in Huber et al. (2021).

2. Usage of spectra

Things to download:

- python scripts "public_spectra_light_curve.py" and "SNLens_object.py" from `https://github.com/shsuyu/HOLISMOKES-public/tree/main/HOLISMOKES_VII` (the pythons scripts need to be stored in the same folder)
- folders "LSNeIa_class.tar" from `https://datashare.mpcdf.mpg.de/s/jNq6IXM9wuFIVKP` and "spectra.tar" from `https://datashare.mpcdf.mpg.de/s/d2fy8m1QwC6v3OO` (this file is with 116 GB really large, to start you can also just download $N_{\mathrm{sys}} = 1$ from `https://datashare.mpcdf.mpg.de/s/8SCDeASdpbT9qkD`, if you do so rename "spectra_single" to "spectra" for the python script). Further, we recommend to store the folders "LSNeIa_class" and "spectra" in another folder called "data_release_holismokes7".

Steps to do:

- unpack tar files using "tar -xf spectra.tar" and "tar -xf LSNeIa_class.tar" in the terminal to get the folders
- open the code "public_spectra_light_curve.py"
- set the "input_data_path" to the path where you stored the "spectra" and "LSNeIa_class folders
- pick your N_{sys} and N_{im} from Table 1, as well as the supernovae_model (options "me", "n1", "su", and "ww")
- pick a random microlensing event by using for micro_config values from 0 to 9999
- use the functions f_load_microlensed_flux and f_load_macrolensed_flux to get the microlensed and macrolensed flux
- specifiy the time after explosion by picking a "time_bin" from Table 2
- run the code using python2.7

Output and additional information:

The two functions for the spectra will return the wavelength, the flux, and the time value in days at which the spectrum was taken. All outputs are in the observer frame (see Section 3 for different source redshifts). The wavelength is measured in Å and the flux in erg/(Åcm^2s). The macrolensed spectrum takes into account the magnification factor $\mu = \frac{1}{(1-\kappa)^2-\gamma^2}$, where all microlensing cases contain magnification from macrolensing and microlensing. Despite microlensing there are no other sources of uncertainties taken into account.

3. Usage of light curves and spectra for a variaty of different source redshifts

Things to download:

- "public_redshifted_spectra_light_curve.py", "SNLens_object.py" and the folder "filter_information.tar" from `https://github.com/shsuyu/HOLISMOKES-public/tree/main/HOLISMOKES_VII` (the pythons scripts and the "filter_information" folder need to be stored in the same folder)
- skip this point if you downloaded stuff for Section 2 otherwise download folders "LSNeIa_class.tar" from `https://datashare.mpcdf.mpg.de/s/jNq6IXM9wuFIVKP` and "spectra.tar" from `https://datashare.mpcdf.mpg.de/s/d2fy8m1QwC6v3OO` (this file is with 116 GB really large, to start you can also just download $N_{\mathrm{sys}} = 1$ from `https://datashare.mpcdf.mpg.de/s/8SCDeASdpbT9qkD`, if you do so rename "spectra_single" to "spectra" for the python script). Further, we recommend to store the folders "LSNeIa_class" and "spectra" in another folder called "data_release_holismokes7".

Steps to do:

- unpack tar files using "tar -xf spectra.tar", "tar -xf LSNeIa_class.tar", and "tar -xf filter_information.tar" in the terminal to get the folders
- open the code "public_redshifted_spectra_light_curve.py"
- set the "input_data_path" to the path where you stored the "spectra" and "LSNeIa_class folders"
- set the "output_data_path" to the folder where to store the new data with the modified redshift
- pick your N_{sys} and N_{im} from Table 1
- set the source redshift "source_redshift_output" for the new data
- pick the number of random microlensing positions your new output should contain by choosing a value for "amount_of_random_microlensing_positions", where the maximum value is 10000 (very long runtime)
- use the function "f_get_microlensed_lightcurve" to create light curves and "f_get_microlensed_spectra" to create spectra
- run the code using python2.7

Output and additional information:

time bin	rest-frame time after explosion [d]
6	3.4
7	3.7
8	4.0
9	4.4
10	4.7
11	5.1
12	5.6
13	6.0
14	6.6
15	7.1
16	7.7
17	8.4
18	9.1
19	9.9
20	10.7
21	11.6
22	12.6
23	13.7
24	14.9
25	16.2
26	17.5
27	19.0
28	20.7
29	22.4
30	24.3
31	26.4
32	28.7
33	31.1
34	33.8
35	36.6
36	39.8
37	43.2
38	46.8
39	50.8
40	55.2
41	59.9
42	65.0
43	70.5

Table 2: Time bins and the corresponding rest-frame time after explosion. To calculate the observed time after explosion, the rest-frame time after explosion needs to be multiplied with (1+source redshift).

If you use "source_redshift_output" = "source_redshift_microlensing_calculation" you get the exact microlensing calculation. As soon as you use a different "source_redshift_output" value the microlensing calculation will just be an approximation but everything else will be exact for the choosen "source_redshift_output". The microlensing is just approximated because it was pre-calculated using the "source_redshift_microlensing_calculation" value, which sets the physical scale of the microlensing magnification map and therefore the apparent size of the SN Ia in such a map changes, which leads to a different likelyhood for a microlensing event to occur. However in terms of the achromatic phase of SN Ia color curves there was no strong dependency on the scale of the microlensing map for typical source and lens redshifts (Huber et al. 2020). The impact on different science cases can be estimated by comparing results from $N_{\mathrm{sys}} = 1$ with "source_redshift_output" = 0.55 or "source_redshift_output" = 0.99 to the results from $N_{\mathrm{sys}} = 2$ and $N_{\mathrm{sys}} = 3$.

The data output from "public_redshifted_spectra_light_curve.py" can be used in the same way as described in Section 1 and 2 (in "public_spectra_light_curve.py" choose a "source_redshift" and change "input_data_path" to the "output_data_path" from "public_redshifted_spectra_light_curve.py").

References

Chan, J. H. H., Rojas, K., Millon, M., et al. 2021, A&A, 647, A115
Huber, S., Suyu, S., Noebauer, U., et al. 2020
Huber, S., Suyu, S. H., Ghoshdastidar, D., et al. 2021
Huber, S. et al. 2019, A&A, 631, A161
Kromer, M. & Sim. 2009, MNRAS, 398, 1809
Nomoto, K., Thielemann, F.-K., & Yokoi, K. 1984, ApJ, 286, 644
Pakmor, R., Kromer, M., Taubenberger, S., et al. 2012, ApJ, 747, L10
Seitenzahl, I. R., Ciaraldi-Schoolmann, F., Röpke, F. K., et al. 2013, MNRAS, 429, 1156
Sim, S. A., Röpke, F. K., Hillebrandt, W., et al. 2010, ApJ, 714, L52
Suyu, S. et al. 2020, arXiv e-prints (arXiv:2002.08378)
Vernardos, G. & Fluke, C. J. 2014, Astron. Comput., 6, 1

Vernardos, G., Fluke, C. J., Bate, N. F., & Croton, D. 2014, Astrophys. J. Suppl., 211, 16
Vernardos, G., Fluke, C. J., Bate, N. F., Croton, D., & Vohl, D. 2015, ApJ Suppl., 217, 23