Script	Contents	Functions/files created
PhD/PhD_scripts/aux_func/		
aux_1_filenames	Lists of filenames for the .elev files	N.B.: we left out certain months due to low data coverage (5-6.2002 of ENV and 7-10.2010 of CS2)
aux_2a_combined_coastline_plot	- the **coastline product** used in all subsequent plots is made up of: > Basemap coastline north of 60S (GSHHG: https://www.soest.hawaii.edu/pwessel/gshhg/) > Antarctic Digital Database (ice shelves+coast:https://www.add.scar.org/)	**cite** Paul Holland and Clément Vic for the ADD data **depends on**: PhD_data/land_masks/holland_vic/
aux_2b_coastlines_gridded_land_mask		land_masks/coastline_nested_lists.pkl land_masks/land_mask_gridded_50s.nc
aux_3a_regrid_GEBCO_topog	- regrid bathymetry data to the altimetry grid - https://download.gebco.net/	**coarse_gebco_p5x1_latlon.nc** (/Volumes/SamT5/PhD/data/topog/) - RegularGridInterpolator
aux_3b_bathym_contour_mask	create a gridded mask based on a bathymetry contour by masking the grid points that are inside that contour	
aux_func.py	trend_ci grid_area, area_weighted_avg interp_nan, gaussian_filt (Clement Vic) rotate_vec r_map_ts	
aux_stereoplot.py		
aux_matlab_func.py	reading .mat files into a dict type	
aux_corr_maps	- compute correlation maps and plot them on stereographic proj (for UC and SLA)	
PhD/PhD_scripts/ch2_altimetry/		
A_altimetry/A_along_track/		

A0_text2nc_files_ENV	- convert .elev files to .nc Post-processing: 1. keep valid data (validity: 0=no, 1=yes) 2. keep ocean and lead type only (surface: 0=unknown, 1=Ocean, 2=Lead, 3=Flow) 3. **|SSHA| > 3 m**, where SSHA = Elevation - Mean SSH 4. label tracks with an integer starting from 1 based on the assumption that consecutive tracks are separated by roughly 180 deg (one full revolution of the satellite is roughly a great circle) 5. determine direction (i.e. ascending/descending) 6. (for CS2 only) label to which retracker every point belongs to (1=LRM, 2=SAR, 3=SARIN) Monthly <YearMonth> altimetry record contains: > surface type (1=ocean, 2=lead) > time > lat, lon, > elevation (SSH), > mean SSH [provided by CPOM], > percentage ice concentration (sic, removed outliers (-999)) > sea ice type (sit, keep all types, 1:open water, 2:FirstYearIce, 3:MultiYearIce, 4:Ambiguous) > confidence in sea ice type (csit, keep only good (4) and excellent (5)) > track_num - every track has been assigned a number > track_dir: (1:ascending (S to N), -1:descending) > retracker type	- Selecting certain columns to read from a text file - shapely - shapefile - Label ascending/descending tracks - Assign an index to each point depending inside which contour it is found - Save data to a .netcdf file in altimetry_cpom/1_raw_nc/ (month0207.nc, 201011.nc)
A1_text2nc_files_CS2		
A1_distance_to_land	- for every pointwise measurement, compute the distance to the nearest coastline point - add a new variable (distance_m [in metres]) to the existing .nc file	- geopy - iterate through files in a directory
A1_fig_number_of_spotSSH	- remove data points less than 10 km away from the nearest coastline - plot fraction of discarded data as well as the distribution of data in different modes (time series)	
A1_fig_single_along_track	- uses the Sea Ice data - plot one track with O and L data to show offset (SSHA vs lat, stereoplot showing the location of the track)	
A1_fig_tracks_all	- figures with bone colormap - density of along-tracks	
A2_fig_cs2_modes_mask	plot geographical masks of CS2 retrackers (version 3.8)	
A2_fig_tracks_cs2_modes	figure	
A2_fig_tracks_env_ol	figure	

A3_bin_numpts	- compute number of points in every grid cell	
A_altimetry/B_offsets/		
B1_bin_ssha_OL_files	- bin SSHA O/L separately and save in a file - discard pts less than 10km away from the nearest coastline; land mask also applied - compare using the mean/median as the bin statistic - bin size: 1 deg lon x 0.5 deg lat	altim_cpom/2_grid_offset/ b01_bin_ssha_OL_env_mean.nc b01_bin_ssha_OL_env_median.nc b01_bin_ssha_OL_cs2_mean.nc b01_bin_ssha_OL_cs2_median.nc
B2_OL_analysis.py	- compute and plot OL climatology	
B2_OL_plots.py	> plot OL offset climatology and save time series and spread of values in every month in a file	**b02_OL_offset_env_30mean.nc** **b02_OL_offset_env_30median.nc** **b02_OL_offset_cs2_30mean.nc** **b02_OL_offset_cs2_30median.nc**
B3_bin_ssha_SAR_cs2	- correct along-track leads with the OL mclim - grid SAR and SARIn (mean/median as bin stats) - save in a file	b03_bin_ssha_SAR_cs2_mean.nc b03_bin_ssha_SAR_cs2_median.nc
B3_SAR_plots_file	- plot SAR-SARIn offset climatology and spread - save in file	**b03_SAR_offset_cs2_30mean.nc** **b03_SAR_offset_cs2_30median.nc**
B4_bin_ssha_LRM_cs2	- correct along-track SARIn with mclim offset - grid LRM/SAR(In) separately (mena/median stats) - save file	b04_bin_ssha_LRM_cs2_mean.nc b04_bin_ssha_LRM_cs2_median.nc
B4_LRM_plots_file	- plot LRM-SAR(In) mclim offset and monthly spread - save in file	**b04_LRM_offset_cs2_30mean.nc** **b04_LRM_offset_cs2_30median.nc**
B5_fig_stereoplot_offset	spatial distribution of the O-L offset example	
A_altimetry/C_merging_satellites/		
C0_crop_lonlat	- use along-track data to extract lon/lat in separate text files to be used for interpolating the geoid at those locations	
C1_correct_bin_dot_env	- correct along track leads - discard point measuremts that are < 10 km from nearest coastline - subtract geoid (EGM2008, GOCO05); keep only |DOT| < 3 m - bin data (mean/median as bin stat) - interpolate (nearest neighbour - griddata in python) and filter (Gaussian filter sigma=150 km) - save in file	**dot_env_30bmean_goco05.nc** **dot_env_30bmean_egm08.nc** **dot_env_30bmedian_goco05.nc** **dot_env_30bmedian_egm08.nc**

C2_correct_bin_dot_cs2	- correct along track leads and retrackers - discard point measuremts that are < 10 km from nearest coastline - subtract geoid (EGM2008, GOCO05); keep only |DOT| < 3 m - bin data (mean/median as bin stat) - interpolate (nearest neighbour - griddata in python) and filter (Gaussian filter sigma=150 km) - save in file	**dot_cs2_30bmean_goco05.nc** **dot_cs2_30bmean_egm08.nc** **dot_cs2_30bmedian_goco05.nc** **dot_cs2_30bmedian_egm08.nc**
C3_intersat_offset	sla_env = dot_env - mdt_env sla_cs2 = dot_cs2 - mdt_cs2 > compute monthly maps of sla_env-sla_cs2 > compute the median/mean in each bin over the overlap period > compute area-weighted average of the mean/median map of residuals - estimate a **constant** value > estimate error?	
C3b_RMS	compute RMS of residuals after correcting CS2 DOT with the intersatellite offset	
C4_combined_dot	- merge satellites; compute geostrophic velocity	
A_altimetry/D_validation/		
D1a_cmems_processing	- processing of copernicul/aviso SLA data to prepare it for comaprison in the open ocean with the altimetry product	
D1b_validation_aviso	- spatial map of temporal correlation with aviso SLA in the open ocean	
D1x_aviso_monthly_avg		
d2_extract_altim_anom	- wrapper to apply linear trend and subtract time mean from monthly data for a given period	
D2_validation_bpr_DPS	- temporal correlation at every grid cell with the time series from the Bottom Pressure Recorder Drake Passage South	
D2_validation_bpr_DPSdeep	- temporal correlation at every grid cell with the time series from the Bottom Pressure Recorder Drake Passage Deep	
D2_validation_bpr_myrtlec	- temporal correlation at every grid cell with the time series from the Bottom Pressure Recorder MyrtleC	
D2_validation_SAM	correlation with SAM index (Southern Mode pattern)	
A_altimetry/E_analysis/		
E0_SLA_example_plot.py	stereographic projection of SLA; needs some tweaking	
E1_DOT_climatology_plot.py	- stereographic plot of monthly climatological DOT/SLA	

E2_linear_trend_altim.py		
E2_linear_trend_figure.py		
E3_eof	- compute EOF and geostrophic velocity	
E3_rec_altimetry_to_file		
E4_wavelet_analysis		
B_sea_ice/		
B1a_SIC_readfile	- read the raw files for inspection	
B1b_SIC_regrid_to_altim	- regrid SIC to wind and then to altimetry (gradually coarsen the dataset makes it less jumpy - I have seem figures somewhere)	
B2a_SIdrift_regrid_to_altim.py		
B2b_SIdrift_combine	- combine files for easier access later, I think for computing ocean surface stress	
Bx_SIC_weddell_crop	- cropped the dataset for the weddell region	
Bx_weddell_plot.py		
C_climate_indices/		
C1_combine_climate_indices_files.py		