

Transmuting S2S Forecasts into Applications



Ángel G. Muñoz

agmunoz@iri.columbia.edu

Andrew W. Robertson

Simon J. Mason

Rémi Cousin

Outline



1. Introduction
2. A good S2S forecast is a skillful S2S forecast
3. What? Combining seasonal and sub-seasonal forecasts to predict rainfall characteristics?
4. Predictions in flexible format and Forecast-based Financing (FbF)

Outline



1. Introduction

2. A good S2S forecast is a skillful S2S forecast

3. What? Combining seasonal and sub-seasonal forecasts to predict rainfall characteristics?

4. Predictions in flexible format and Forecast-based Financing (FbF)

Introduction

transmute **verb**

trans·mute | \ tran(t)s-'myüt , tranz-\

transmuted; transmuting

Definition of *transmute*

transitive verb

- 1 : to change or alter in form, appearance, or nature and especially to a higher form
- 2 : to subject (something, such as an element) to [transmutation](#)

intransitive verb

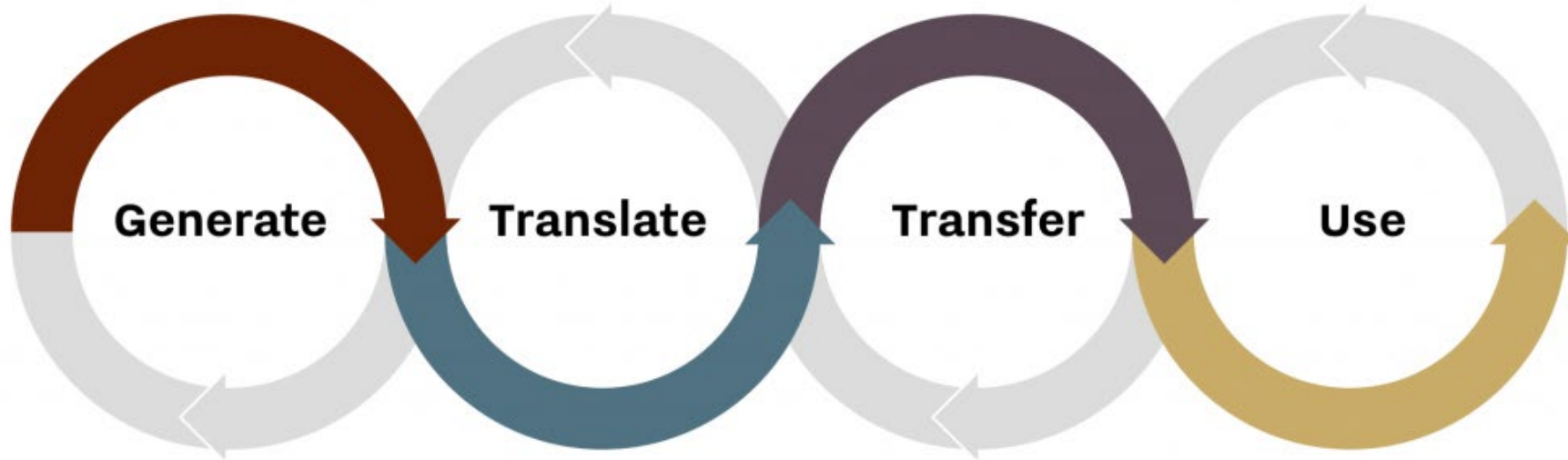
: to undergo transmutation



Introduction

■ Generate climate information and knowledge - learn from the past, monitor the present, forecast the future.

■ Transfer the translated information to the appropriate beneficiaries, in formats and media most useful to their operations



■ Translate the climate knowledge into Information that is relevant to agriculture, public health and other target sectors.

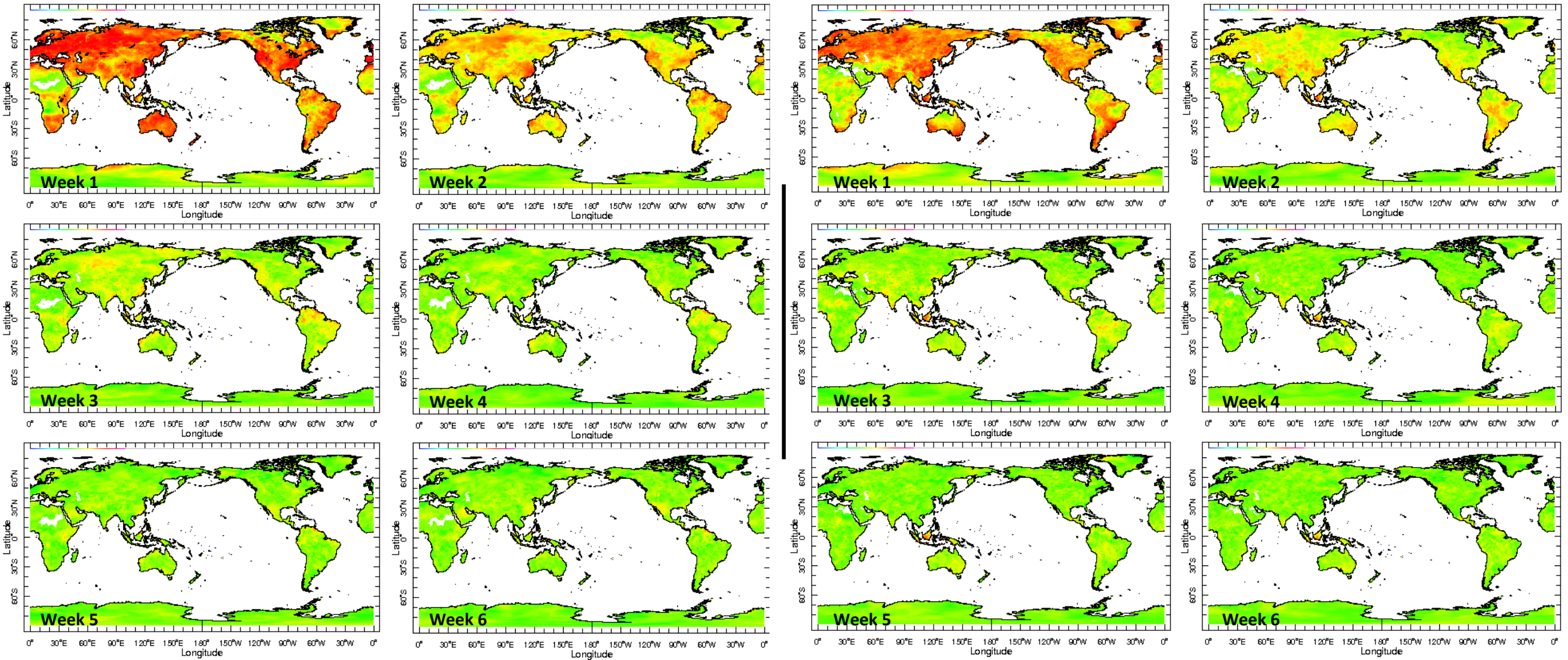
■ Put the translated and transferred climate knowledge to use in operational decision processes, policies and plans. Learn what works and what doesn't.

Outline



1. Introduction
2. A good S2S forecast is a skillful S2S forecast
3. What? Combining seasonal and sub-seasonal forecasts to predict rainfall characteristics?
4. Predictions in flexible format and Forecast-based Financing (FbF)

The Seasonality of Sub-Seasonal Skill



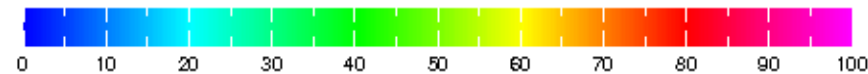
Init: Jan

worse than clim

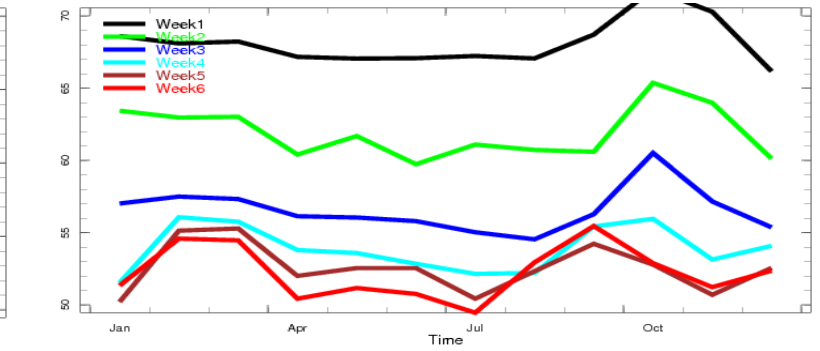
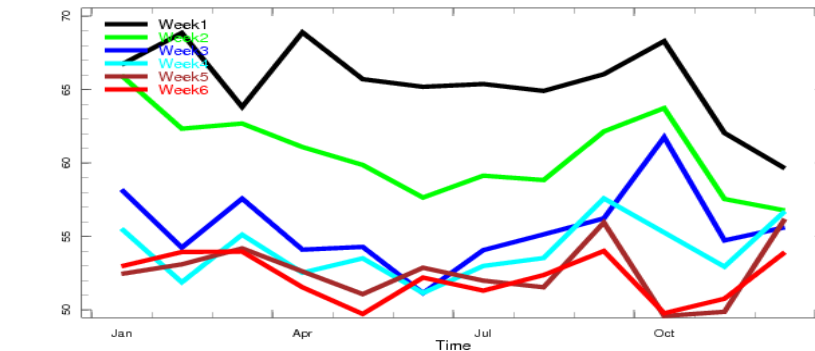
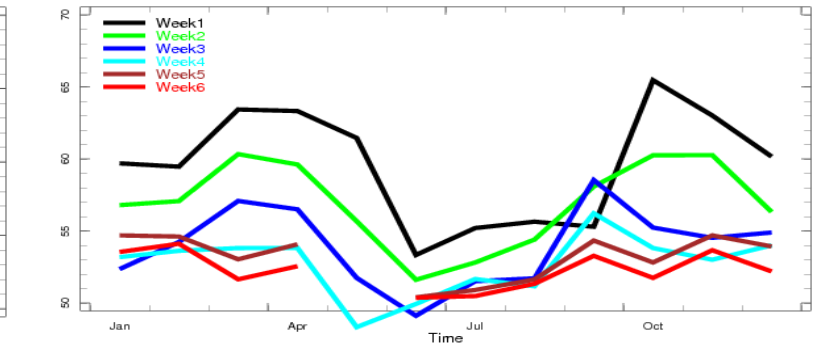
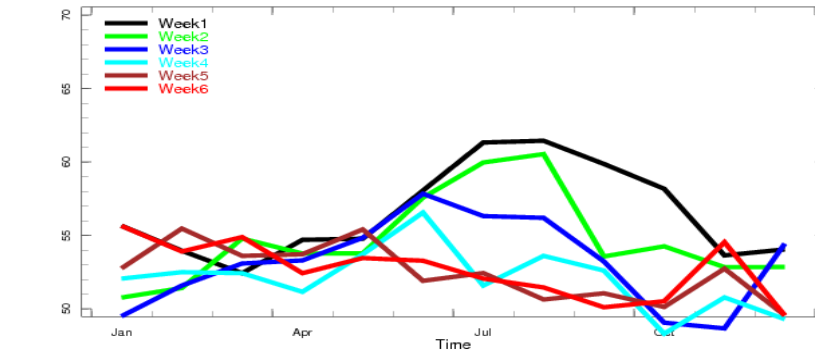
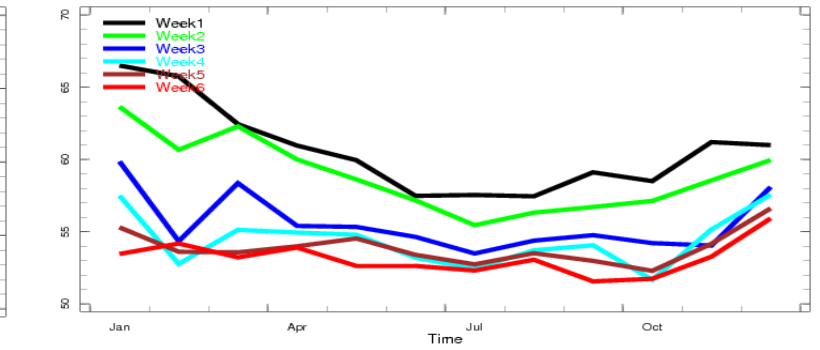
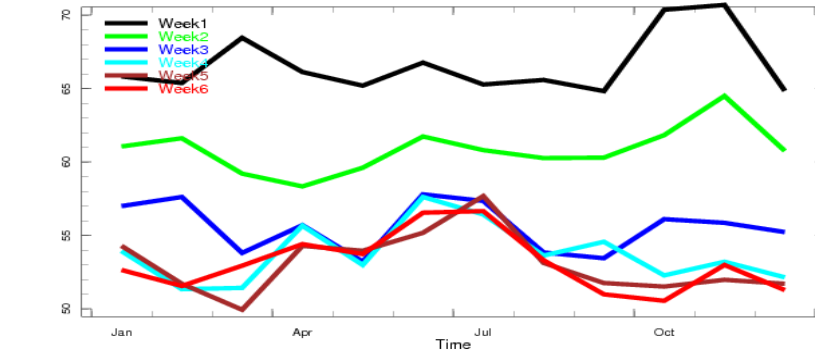
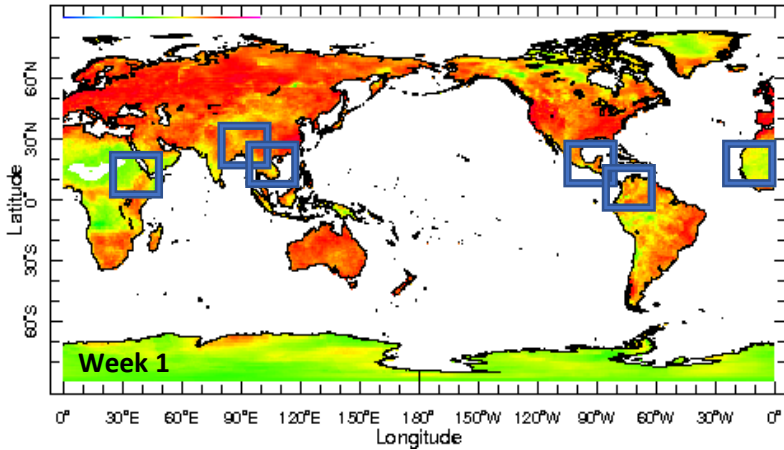
2AFC

better than clim

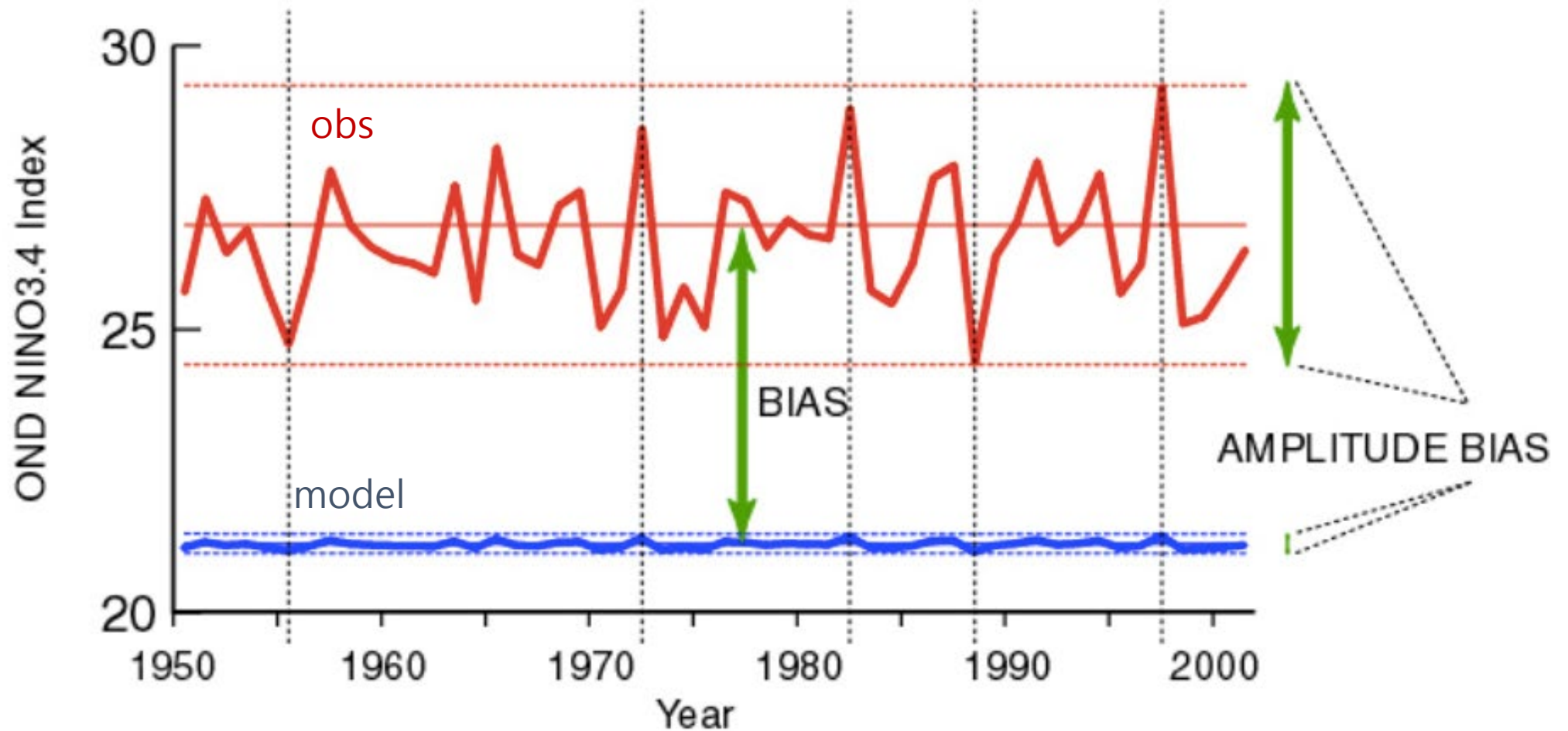
Init: Jul



The Seasonality of Sub-Seasonal Skill

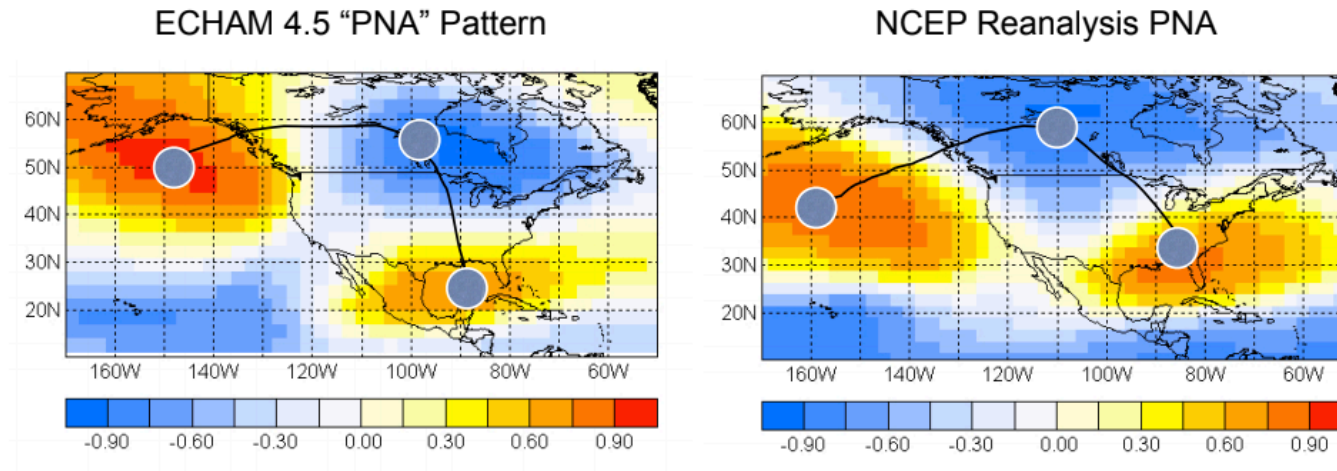


Why do we need to calibrate?



Why do we need to calibrate?

Conditional Bias (errors in patterns of variability)



**Important climate features may be displaced in GCMs
relative to observations: *Systematic spatial biases***

PyCPT

Jupyter PyCPT_v1.2 Last Checkpoint: Last Saturday at 7:53 PM (autosaved)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

PyCPTv1.2 -- 30 March 2019

Produce S2S (un)calibrated forecasts and assess associated skill

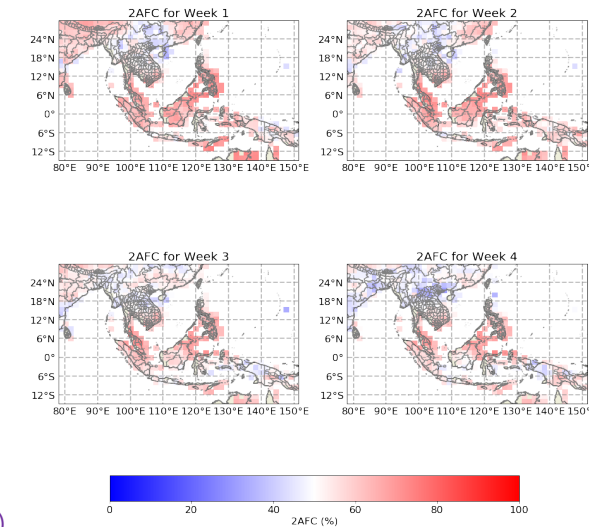
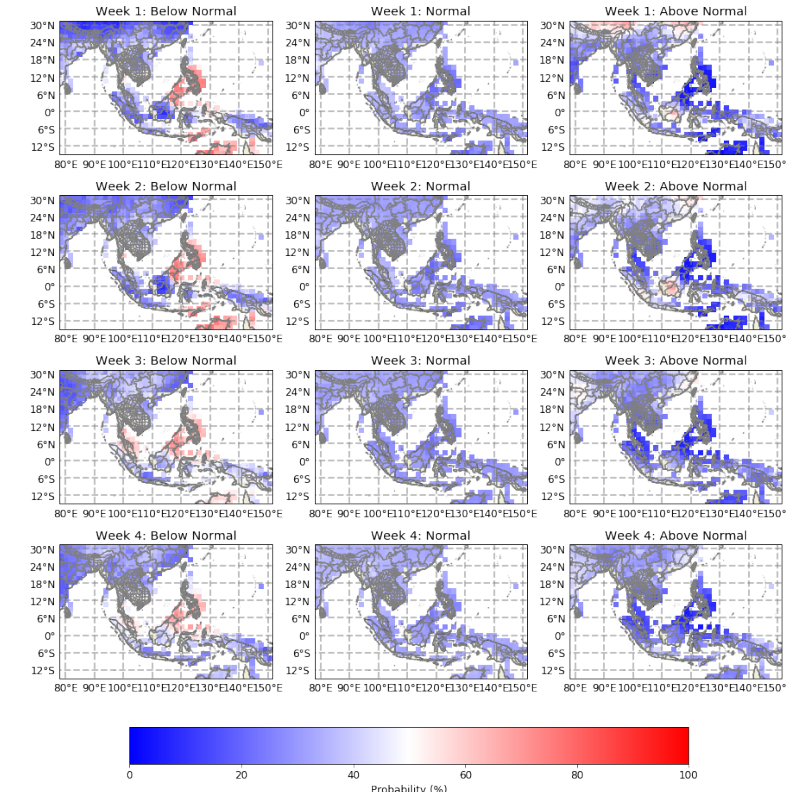
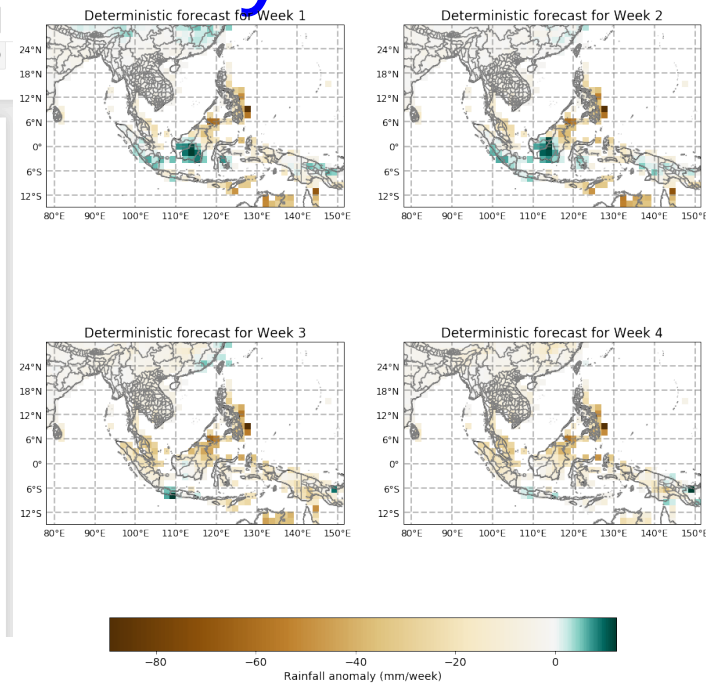
ECMWF/CFSv2/GEFS-TRMM version

Authors:
 Ángel G. Muñoz (agmunoz@iri.columbia.edu) and Andrew W. Robertson (awr@iri.columbia.edu)

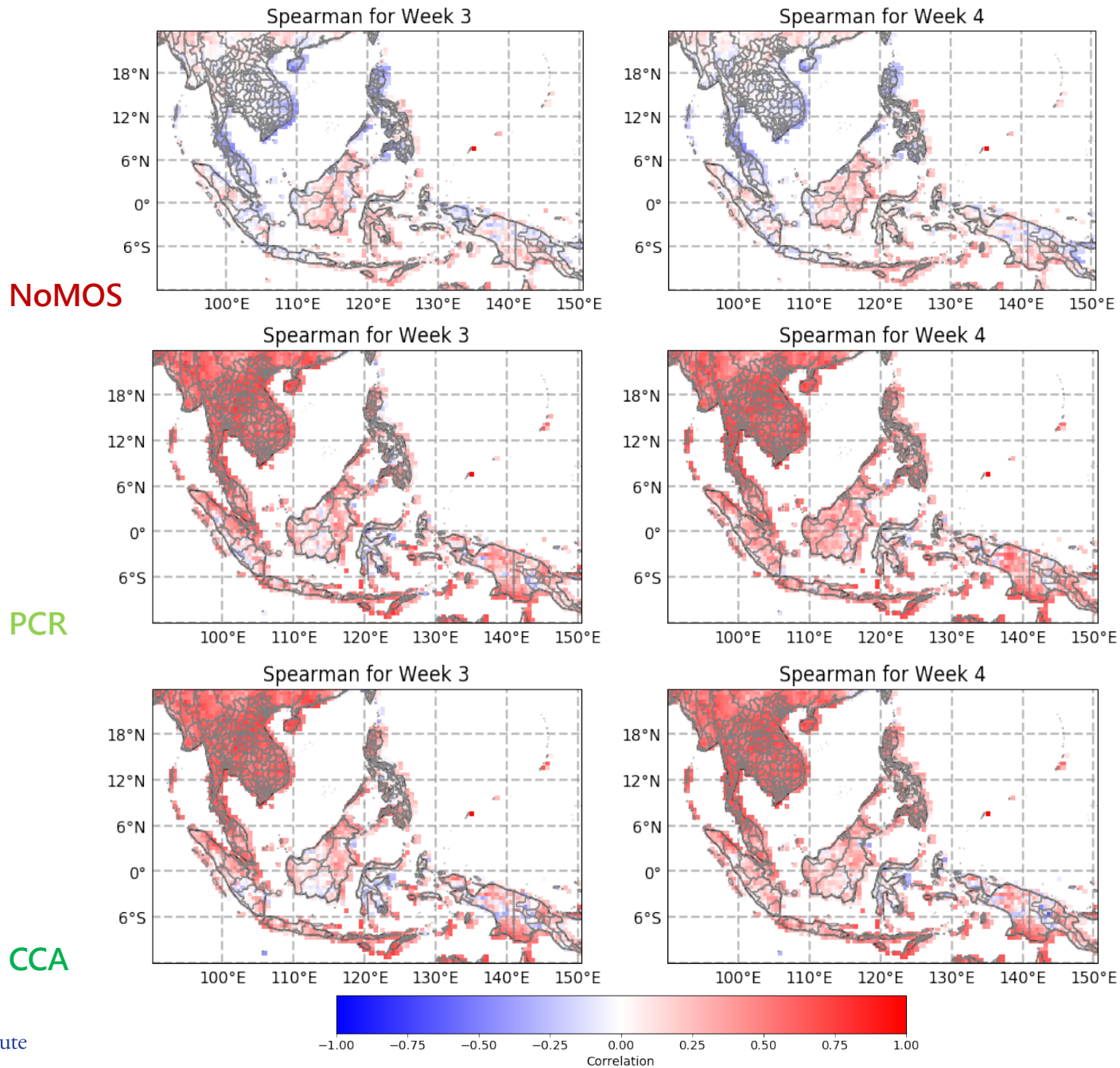
Acknowledgements:
 Rémi Cousin (key Ingrid code), James Doss-Gollin (key Python functions). Part of the effort to develop PyCPT is funded by the Columbia World Project "Adapting Agriculture to Climate Today, for Tomorrow" (ACToday), and NOAA MAPP's projects NA18OAR4310275 (Muñoz) and NA16OAR4310145 (Robertson).

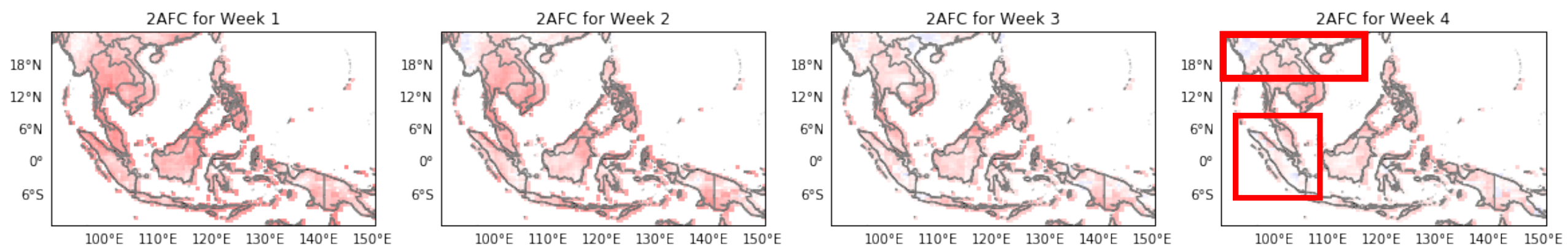
Version log

- 30 Mar 2019, AGM: added PCR option, CHIRPS as obs, flexible format plots, automatically uses retrospective for validation (due to the very high sample size). Solved problems related to masking missing values. ELR still has some problems (values are different from our R or Matlab codes -- working on it, so not included in this version).
- 25 Aug 2018, AGM: added CPC Unified data as obs, plots are now raster maps, fixed field shift due to sequential grads format in CPT, automatic colorbar limits and field name for deterministic forecast
- 24 Aug 2018, AWR: IMD data version (not public).
- 23 Aug 2018, AWR: GEFS SubX model added.
- 15 Aug 2018, AGM: Plotting capabilities added.
- PyCPTv1.1:ECMWF/CFSv2-TRMM, modified by AGM on 1 Aug 2018
- Previous version: 17 Jul 2018, modified by AWR

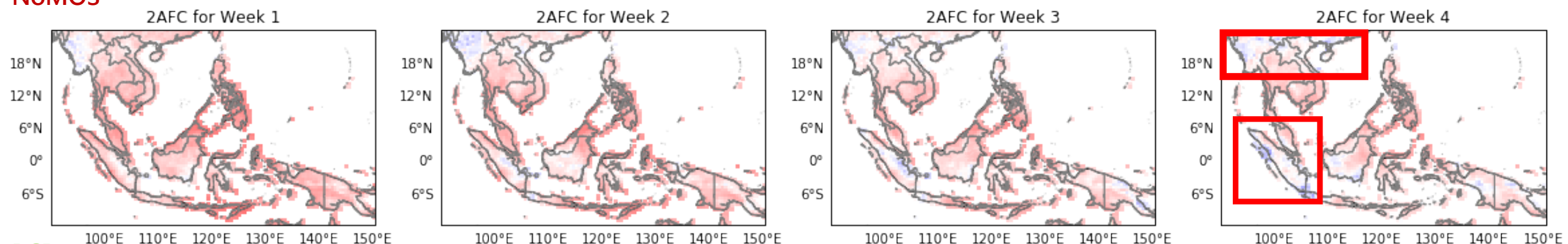


- Python interface for IRI's Climate Predictability Tool (CPT), a widely used research and application Model Output Statistics/Prediction toolbox.
- Publicly available: [GitHub](#).
- Automatically downloads required observations (CHIRPS, TRMM, CPC Unified) and S2S model data from the IRI Data Library (S2S Database and SubX –ECMWF, CFSv2, GEFS, others are being included).
- Computes climatologies, anomalies, a variety of skill metrics (uncalibrated and PCR/CCA-calibrated hindcasts) and probabilistic sub-seasonal forecasts.

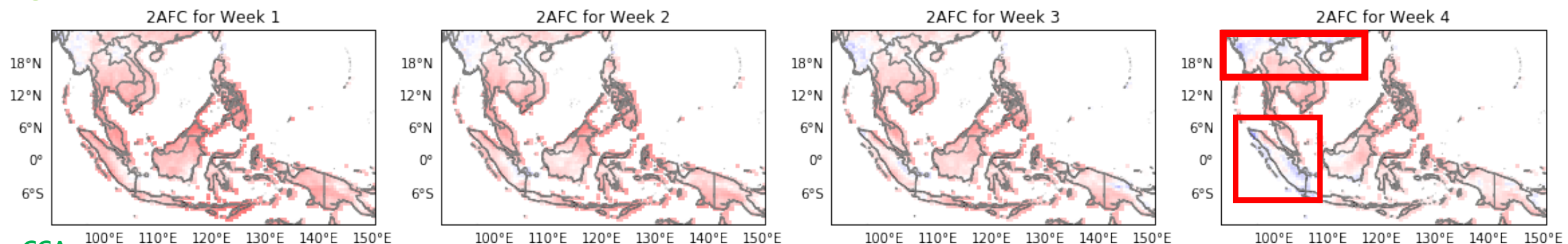




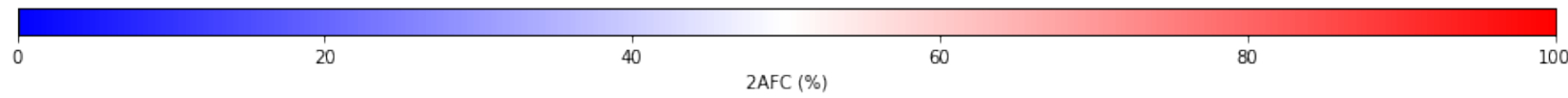
NoMOS



PCR

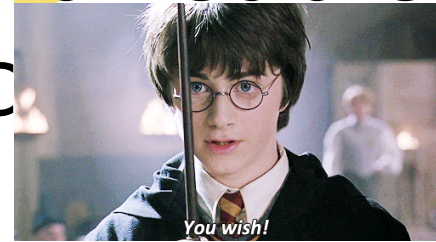


CCA



Outline

1. Introduction
2. A good S2S forecast is a skillful S2S forecast
3. What? Combining seasonal and sub-seasonal forecasts to predict rainfall characteristics
4. Predictions in flexible format and Forecast-based Financing (FbF)



Search or jump to... Pull requests Issues Marketplace Explore

agmunozs / Weather-typing Unwatch 1 Star 1 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided. Edit

Manage topics

11 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

agmunozs Update README.md Latest commit f7d6e20 31 seconds ago

PyWR-CFSv2.ipynb	Minor update	11 minutes ago
PyWR_ECMWF.ipynb	Minor update	4 minutes ago
README.md	Update README.md	20 seconds ago
S2S_WTs_wEOFproj_PL...	Latest version --reanalysis needs to be included	6 months ago
WT4S2S.py	Add files via upload	9 months ago
WT4S2Sv3.ipynb	Add files via upload	9 months ago
WT4S2Sv3_members.ip...	Add files via upload	9 months ago
WTs.py	Add files via upload	9 months ago
WTs_ICTP_Obs.m	Add files via upload	9 months ago
WTs_ICTP_SPEEDY.m	Add files via upload	9 months ago
WTs_ICTP_SPEEDY_loca...	Add files via upload	9 months ago
WTs_ICTP_SPEEDY_loca...	Add files via upload	9 months ago
WeatherTypes_ECMWF.i...	Add files via upload	4 months ago

README.md

Weather-typing

Several weather-typing codes, in Matlab and Python.

PyWR

Python script to compute weather types/regimes using K-means. Model weather types are projected into the observed ones in the EOF space.

Authors:
James Doss-Gollin (james.doss-gollin@columbia.edu) and Ángel G. Muñoz (agmunoz@iri.columbia.edu)

PyWR

Download Data

Download ERAI and re-forecast data. 5-day running average and naïve climatology/anomalies already computed using the IRI Data Library (see links for details)

```
In [19]: mkdir -p data
         tree data

/bin/sh: tree: command not found
```

To download data from the IRI data library, you need an authentication key. This is stored in a file called .IRIDLAUTH, but is not part of the GitHub repository -- you need to contact the IRI Data Library to request access. Once you have done so, you can put your own authentication key in a file called .IRIDLAUTH and use this code. This is a moderately annoying step, and we apologize, but it is required by the S2S Database Terms and Conditions and is necessary for us to share all our code while maintaining some security.

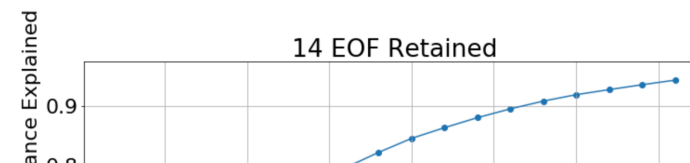
NB if you're using git, be sure to add .IRIDLAUTH to your gitignore file :)

```
In [20]: with open('.IRIDLAUTH') as file:
         authkey = file.read()
```

Dimension Reduction

We need to choose a percentage of variance explained that we will require:

```
In [24]: n_eof = get_number_eof(X=reanalysis['ratio'].values, var_to_explain=0.9, plot=True)
```



Reanalysis Weather Typing

Now we perform the clustering. We will manually specify the number of clusters we want to create and the number of simulations we want to run.

```
In [9]: ncluster = 4 # use 4 WTs
        n_sim = 25 # typically several hundred -- this is for quick preliminary computation only
```

Now we can use this to run the classifiability index on our centroids

```
In [10]: centroids, wtypes = loop_kmeans(X=reanalysis_pc, n_cluster=ncluster, n_sim=n_sim)
         class_idx, best_part = get_classifiability_index(centroids)
         print('The classifiability index is {}'.format(class_idx))
```

The classifiability index is 0.9991181553526277

Now that we have identified a suitable partition, we can use it to keep only the corresponding centroid and set of weather type labels. To take advantage of the scikit-learn syntax, we then use these centroids to define a KMeans object.

```
In [11]: best_fit = KMeans(n_clusters=ncluster, init=centroids[best_part, :, :], n_init=1, max_iter=1).fit(
         reanalysis_pc)
```

Model Weather Types

Now, for each ensemble member, calculate the weather types by projecting the model geopotential height anomaly fields onto the reanalysis regimes and assigning each day of the forecast to the closest centroid in terms of Euclidean distance. The model anomaly fields are first calculated by first taking 5-day running averages and then subtracting the hindcast climatology on a daily basis. The anomalies are then projected into the reanalysis EOFs, and the distances then calculated in EOF space.

```
In [12]: # start with reanalysis
         reanalysis_composite = reanalysis.copy()
         model_clust = best_fit.fit_predict(reanalysis_pc) # get centroids
         weather_types = xr.DataArray(
             model_clust,
             coords = {'time': reanalysis_composite['time']},
             dims='time'
         )
```

No description, website, or topics provided. Edit

11 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

agmunozs Update README.md	Latest commit f7d6e20 31 seconds ago
PyWR-CFSv2.ipynb	Minor update 11 minutes ago
PyWR_ECMWF.ipynb	Minor update 4 minutes ago
README.md	Update README.md 20 seconds ago
S2S_WTs_wEOFproj_PL...	Latest version --reanalysis needs to be included 6 months ago
WT4S2S.py	Add files via upload 9 months ago
WT4S2Sv3.ipynb	Add files via upload 9 months ago
WT4S2Sv3_members.ip...	Add files via upload 9 months ago
WTs.py	Add files via upload 9 months ago
WTs_ICTP_Obs.m	Add files via upload 9 months ago
WTs_ICTP_SPEEDY.m	Add files via upload 9 months ago
WTs_ICTP_SPEEDY_loca...	Add files via upload 9 months ago
WTs_ICTP_SPEEDY_loca...	Add files via upload 9 months ago
WeatherTypes_ECMWF.i...	Add files via upload 4 months ago

PyWR

Weather-typing

Several weather-typing codes, in Matlab and Python.

PyWR

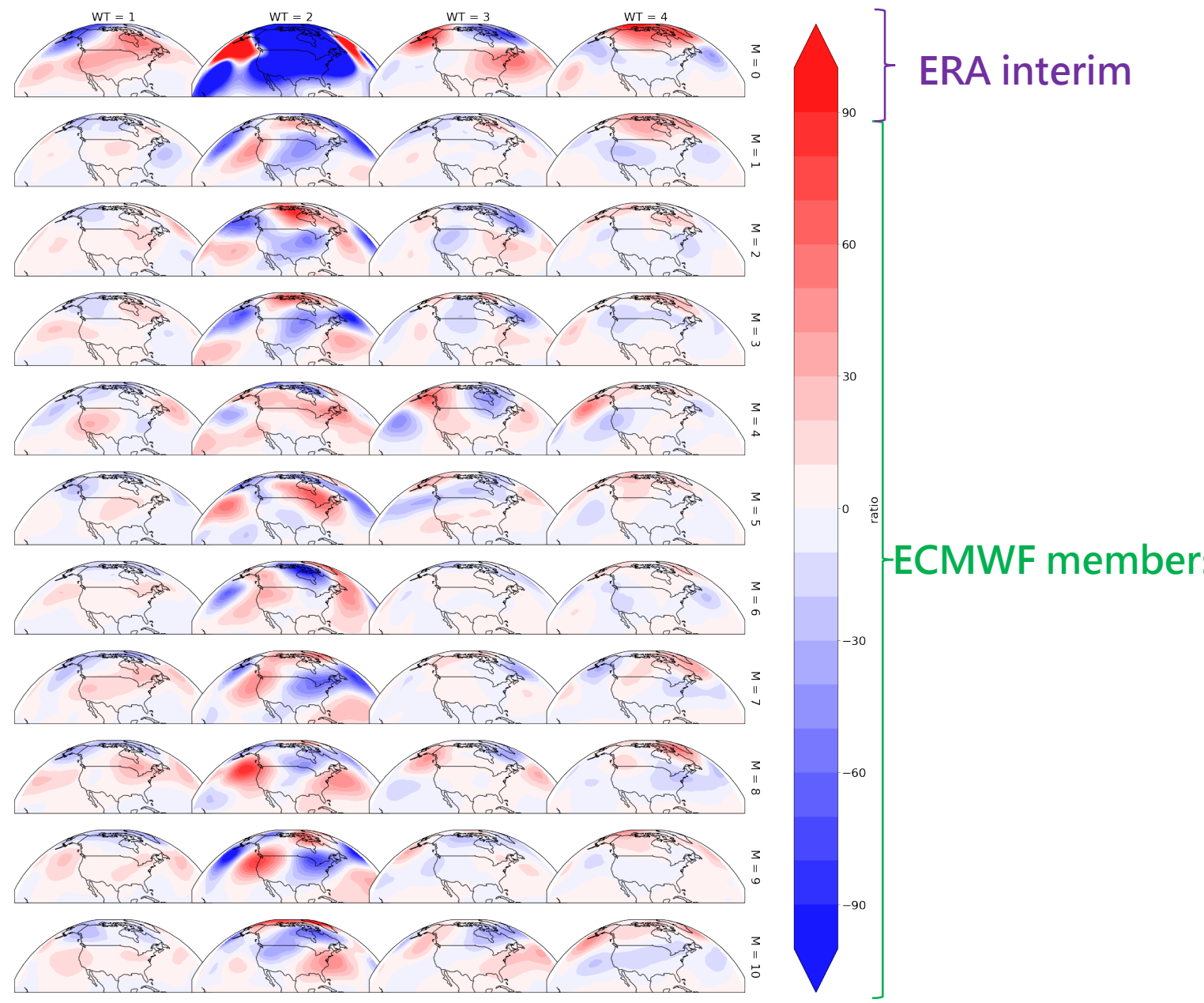
Python script to compute weather types/regimes using K-means. Model weather types are projected into the observed ones in the EOF space.

Authors:

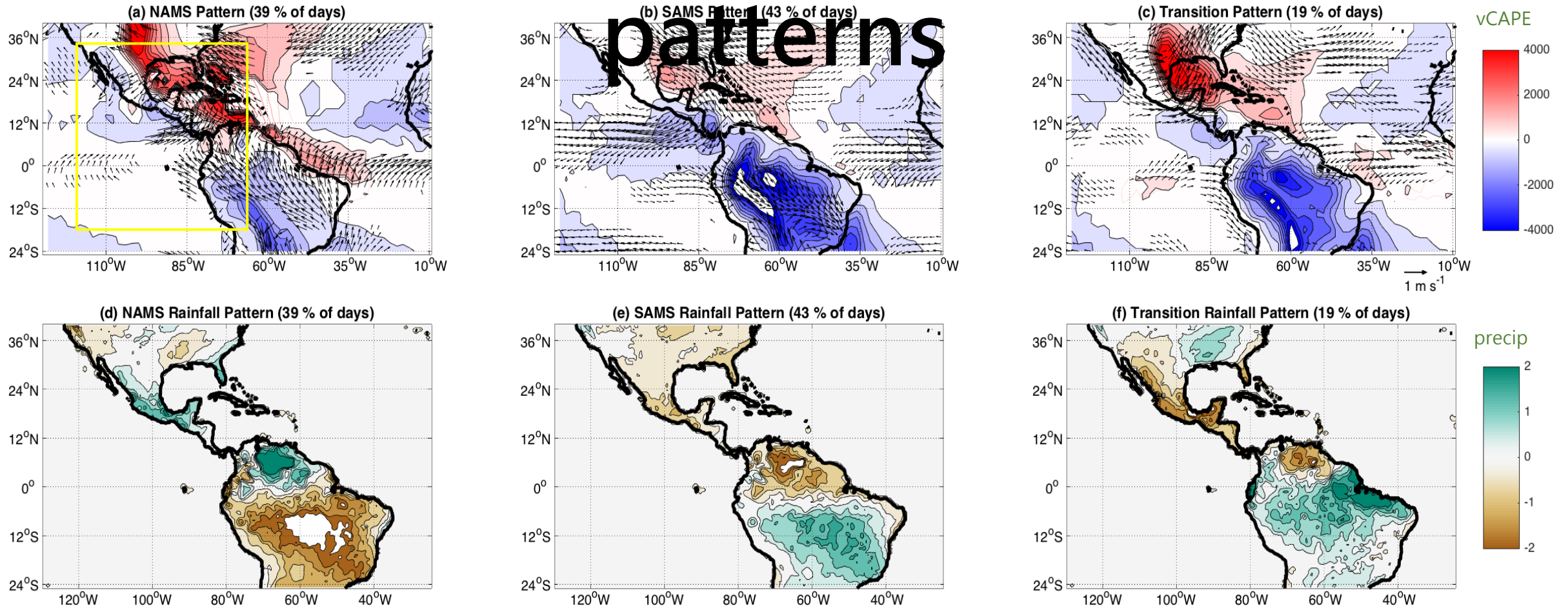
James Doss-Gollin (james.doss-gollin@columbia.edu) and Ángel G. Muñoz (agmunoz@iri.columbia.edu)



<https://github.com/agmunozs/Weather-typing>

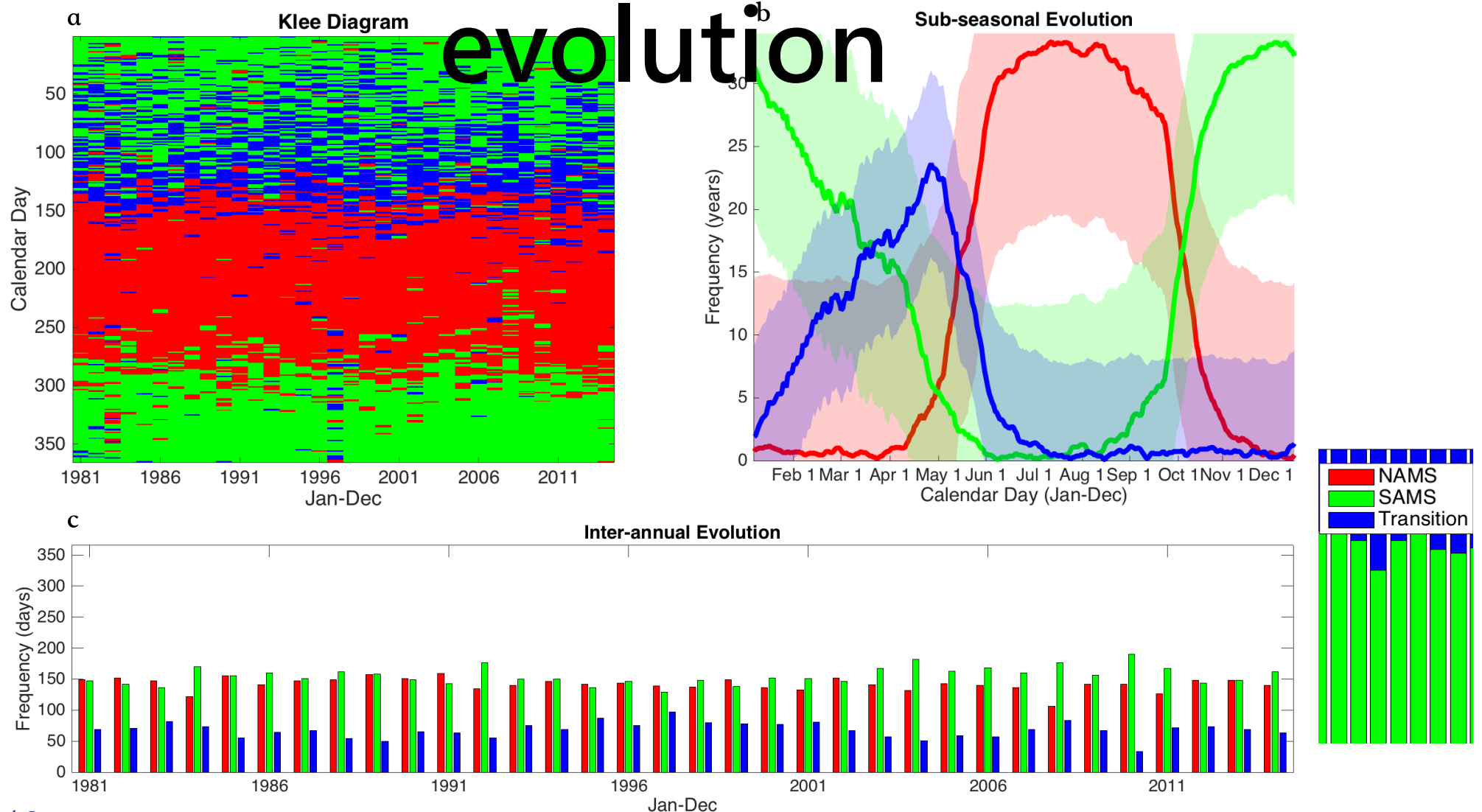


Circulation Regimes: spatial

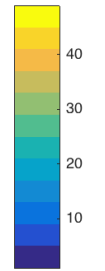
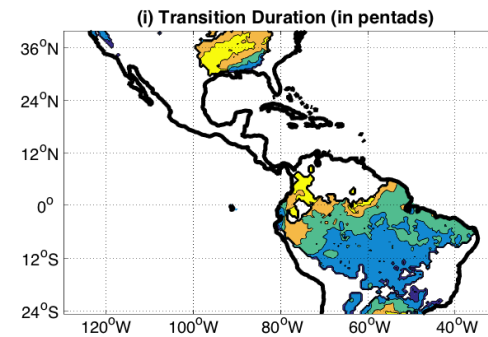
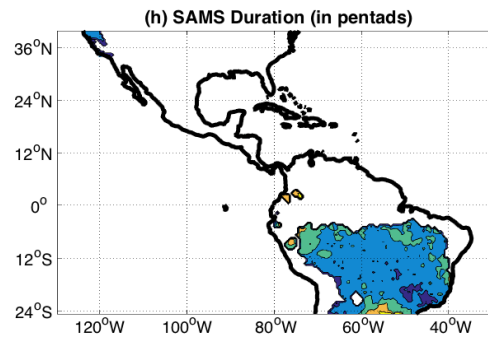
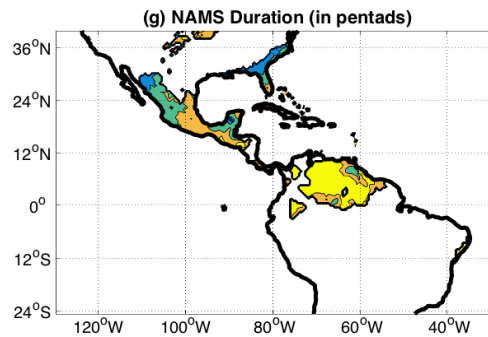
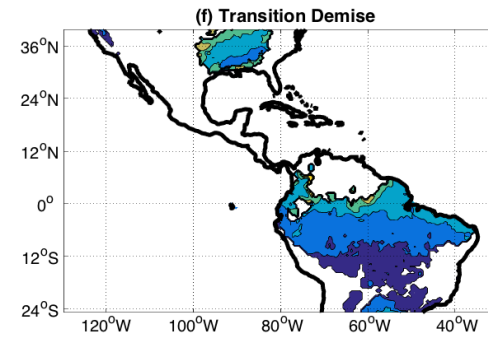
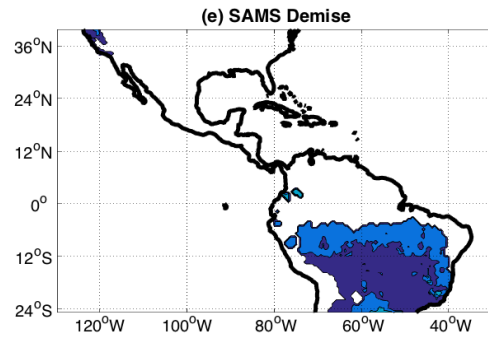
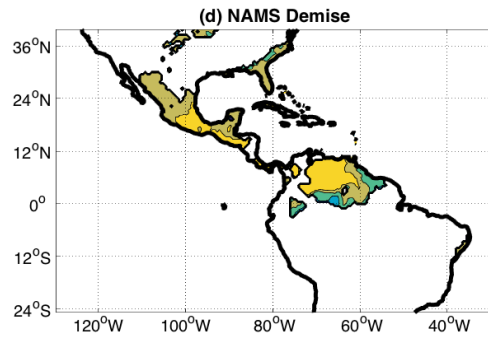
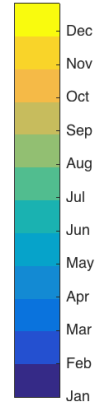
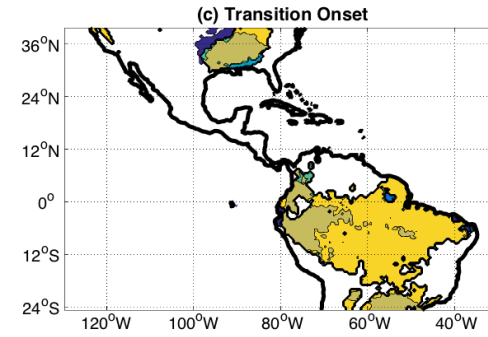
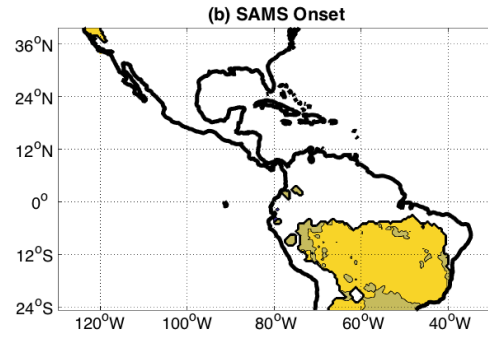
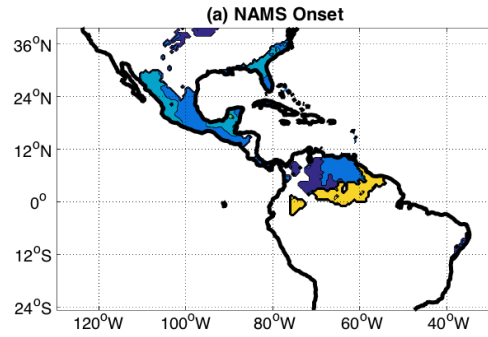


We use vCAPE in the yellow box and k -means to identify the circulation patterns (a)-(c), with vectors showing wind anomalies at 850mb; (d)-(f) are rainfall anomaly composites. Data: 20thC reanalysis, at daily resolution (1981-2015).

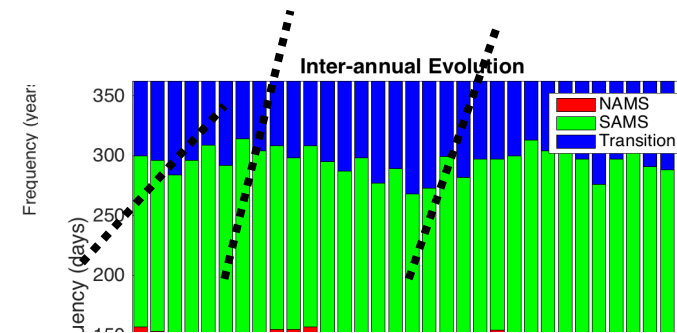
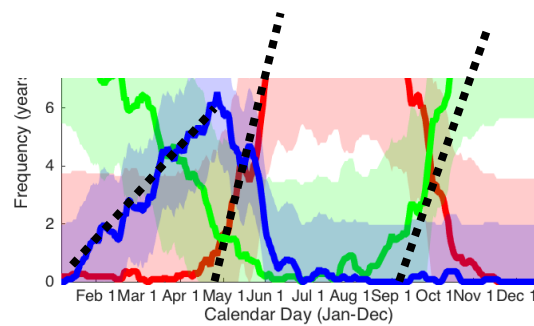
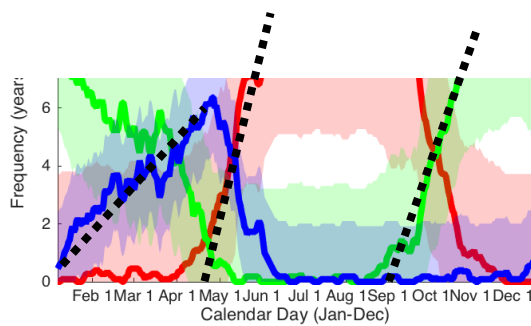
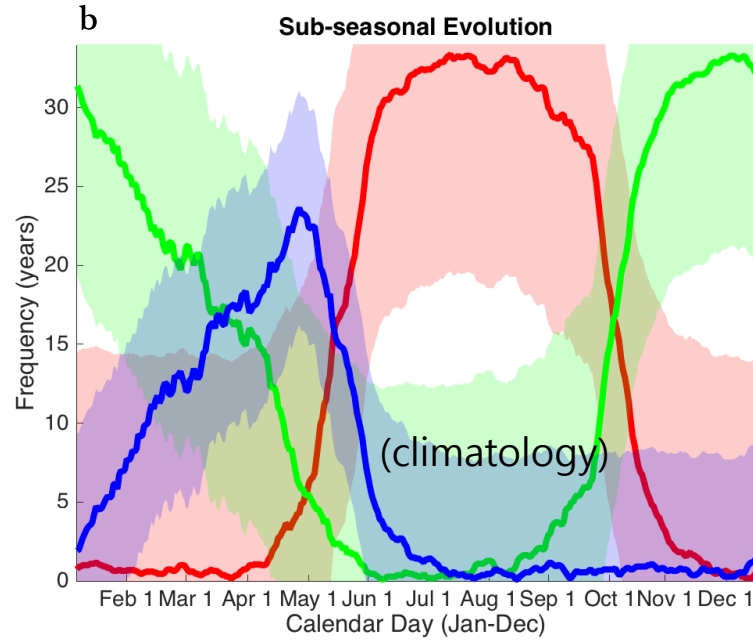
Circulation Regimes: temporal evolution



Circulation Regimes and Rainfall

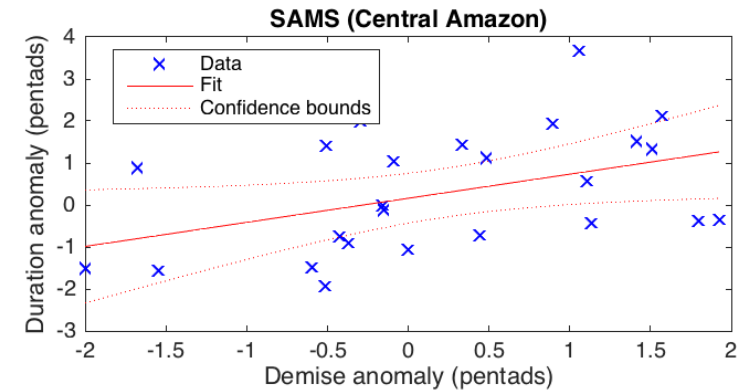
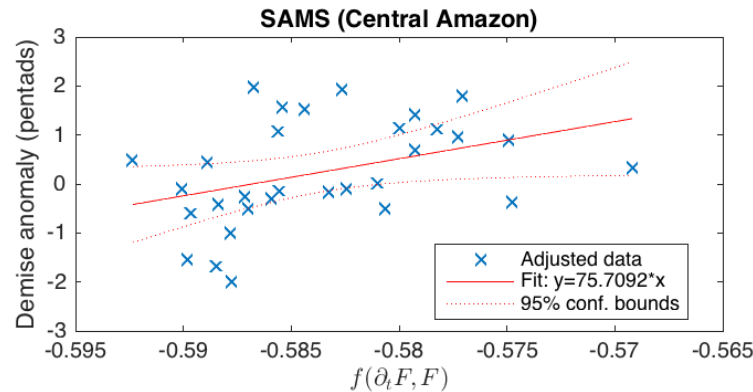
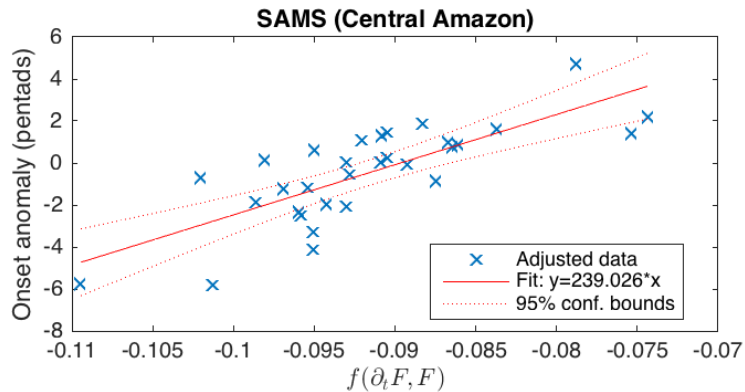
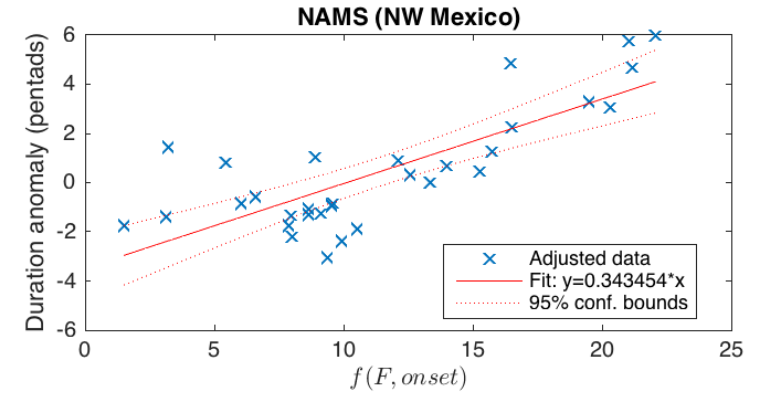
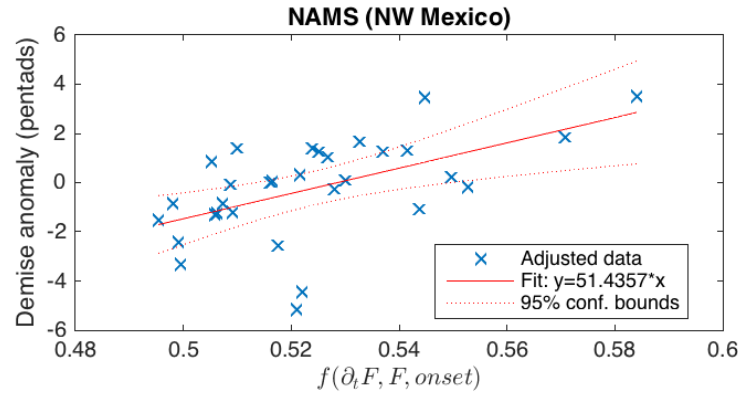
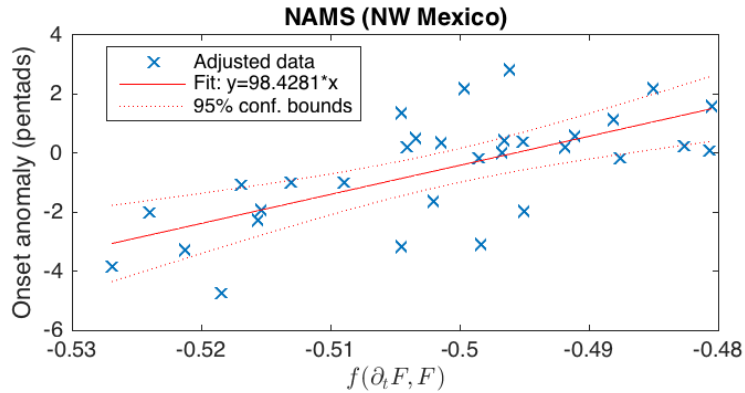


Circulation Regimes: predictability



Candidate predictors: $f(\partial_t F, F, onset)$ $f_t(\partial_t F, F)$

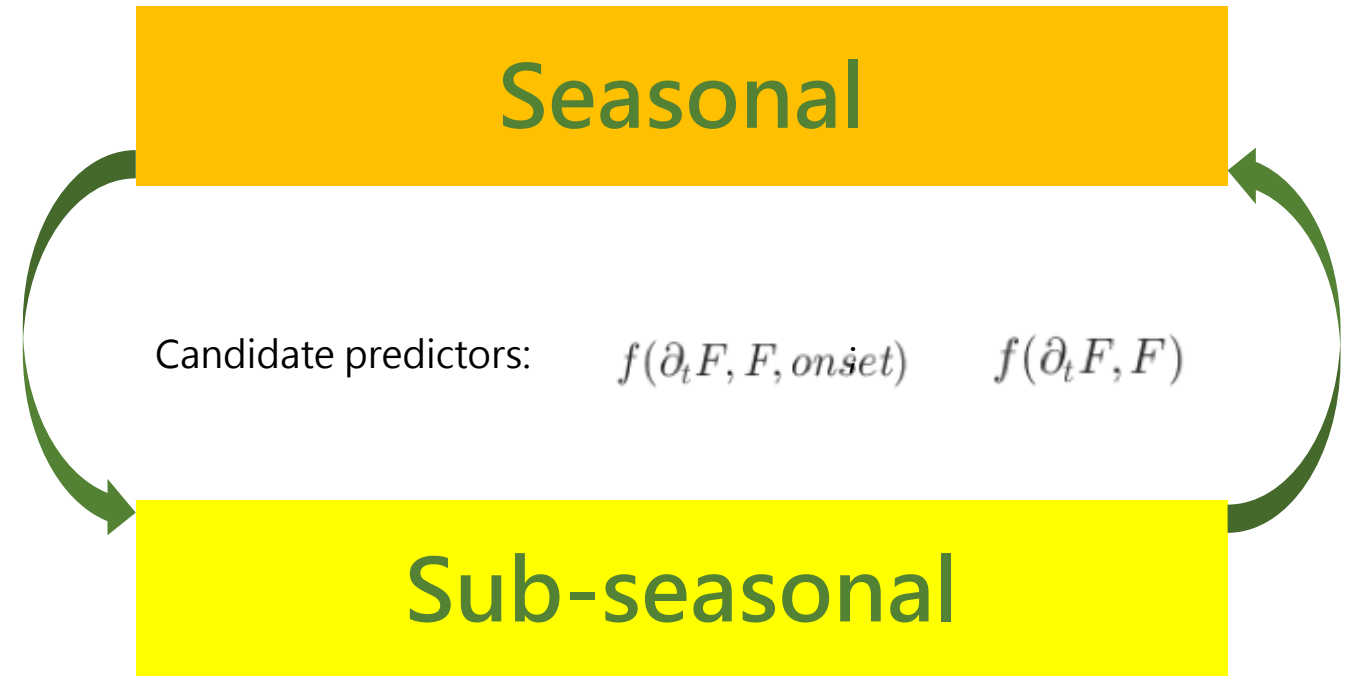
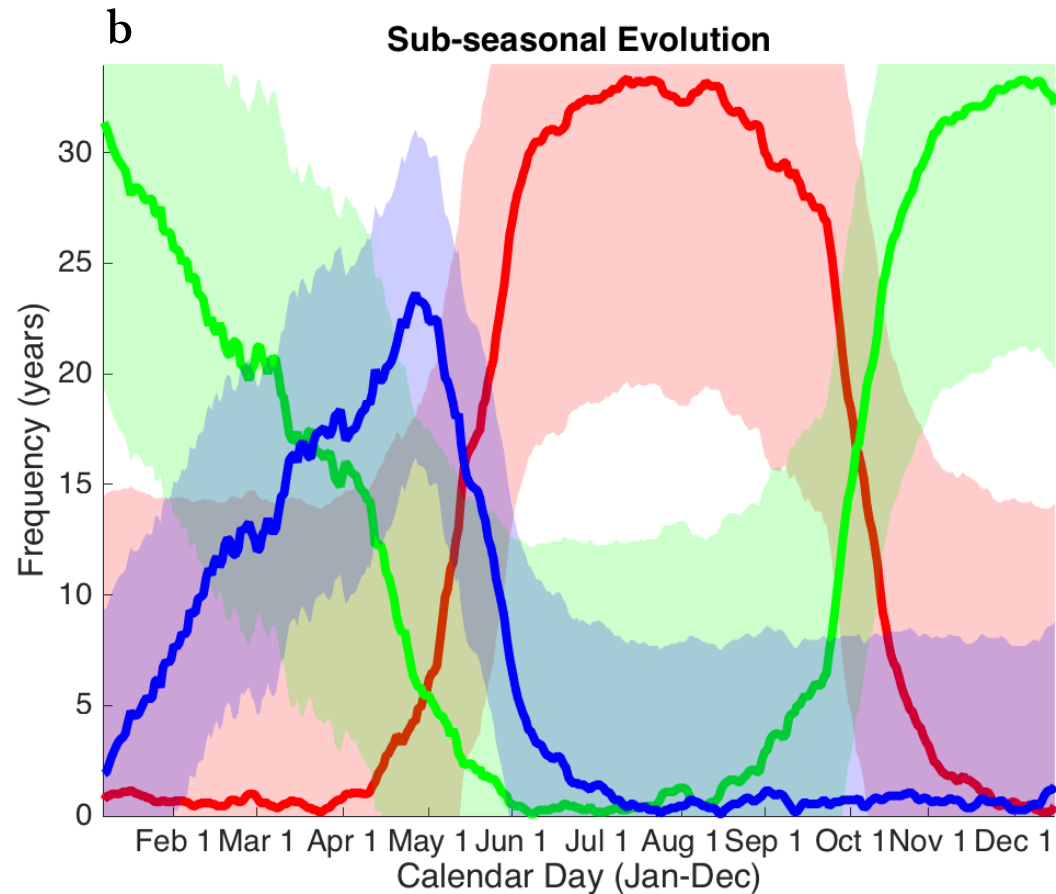
Predictability of Rainfall Characteristics



Candidate predictors: $f(\partial_t F, F, onset)$ $f(\partial_t F, F)$

Combining Seasonal and Subseasonal Predictions

Multi-layered S2S
forecast system

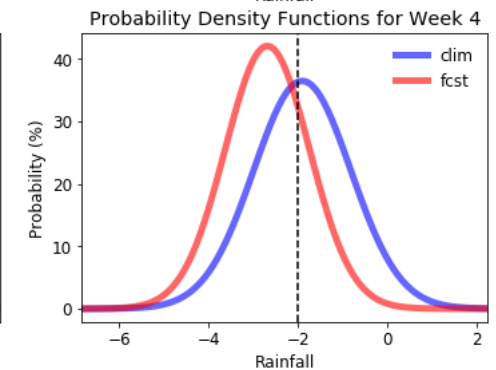
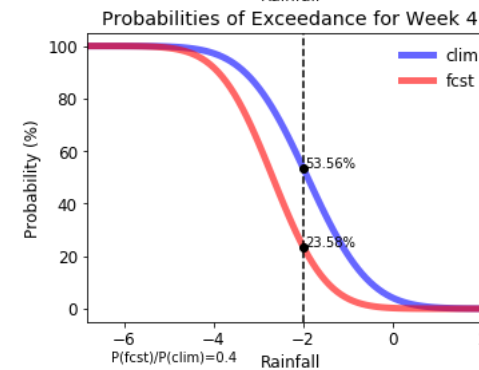
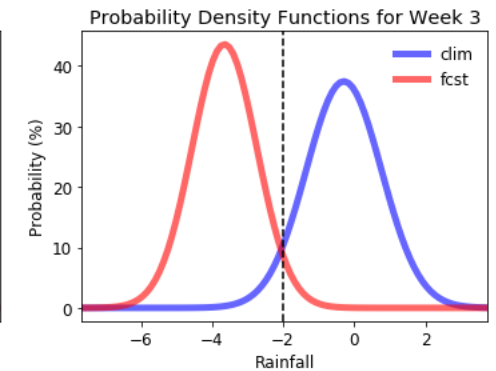
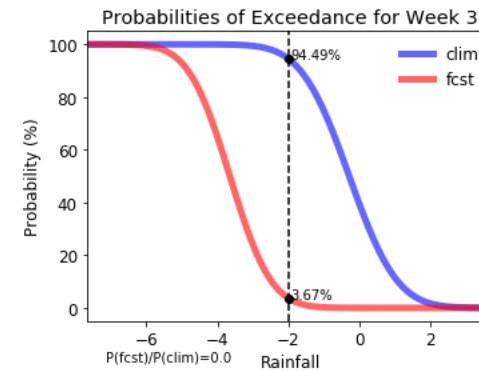
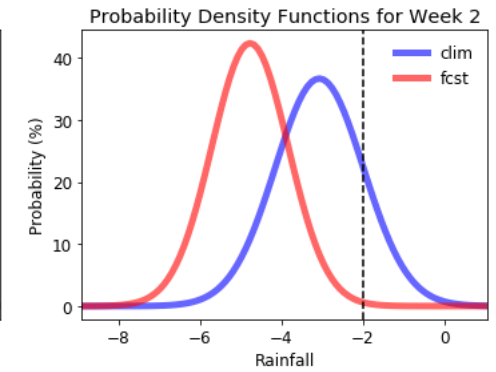
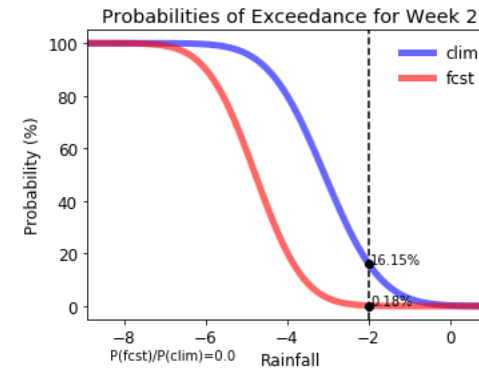
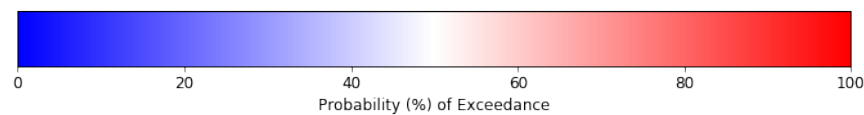
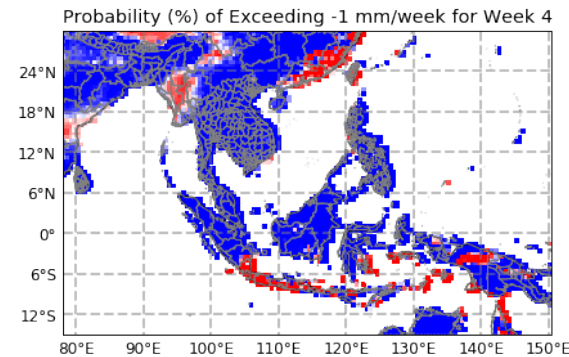
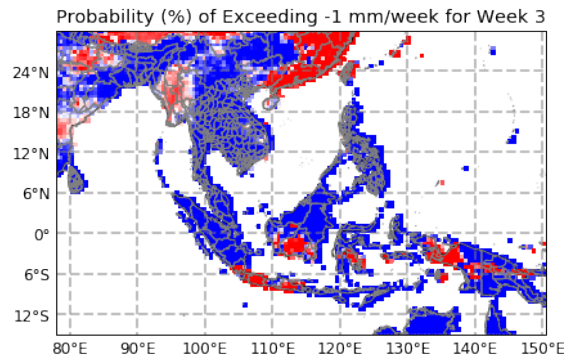
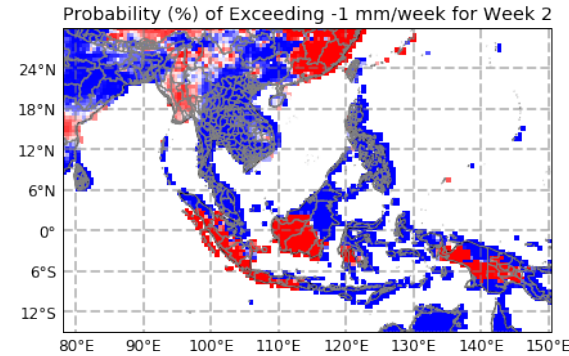
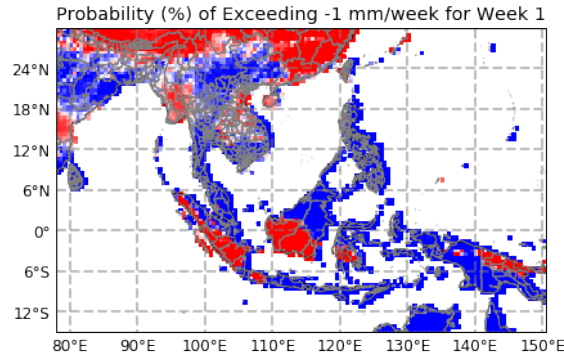


Outline

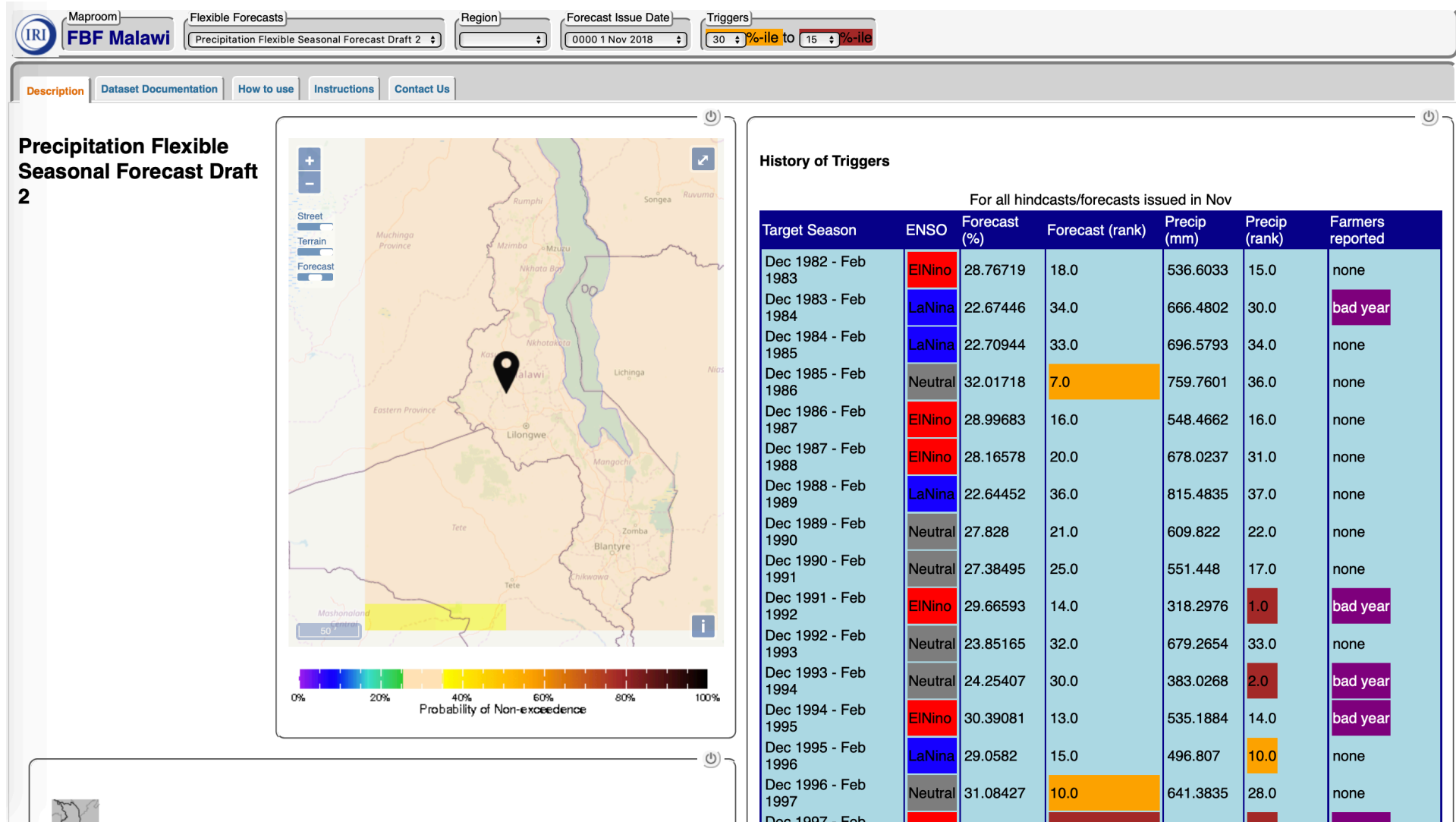


1. Introduction
2. A good S2S forecast is a skillful S2S forecast
3. What? Combining seasonal and sub-seasonal forecasts to predict rainfall characteristics?
4. **Predictions in flexible format and Forecast-based Financing (FbF)**

Predictions in Flexible Format

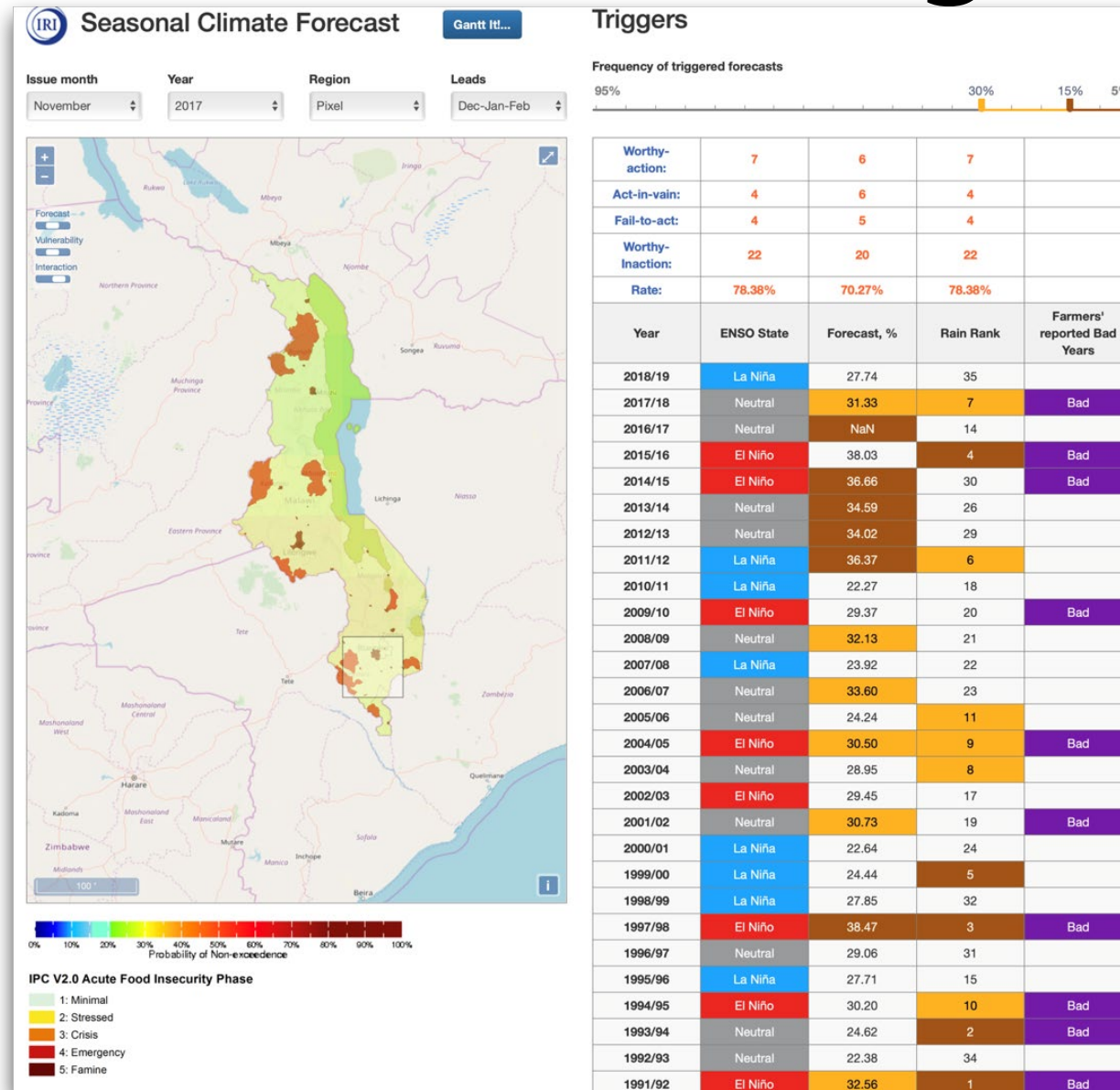


Forecast-based Financing Maproom



Forecast-based Financing Maproom

Python
Interface for
the Data
Library
Maproom!



Summary



1. A good S2S forecast is a skillful S2S forecast
2. Some benefits in combining seasonal and sub-seasonal forecasts to predict rainfall characteristics
3. Advantages in predictions in flexible format

Transmuting S2S Forecasts into Applications

The background of the slide is a dark blue map of the Americas. Overlaid on the map are several irregular, semi-transparent shapes in shades of yellow, green, and red, representing forecast data or model outputs. The shapes are most prominent in the central and southern parts of the continent.

Ángel G. Muñoz

agmunoz@iri.columbia.edu

Andrew W. Robertson

Simon J. Mason

Rémi Cousin