Regressing to a Weighted Mean
Towards a CMIP Ensemble Numerically Weighted by Performance and Dependence

Stefan Vladusic

18 April 2022

1 Background and Motivation

1.1 Rescuing Democracy

Forecasts of climate variables in the Earth system are often produced by combining the outputs of multiple state-of-the-art Earth System Models (ESMs). The outputs of these so-called multi-model ensembles (MMEs) demonstrate "consistently better performance" than their constitutive models alone [16]. Furthermore, MMEs are an invaluable resource when quantifying model uncertainty of ESMs [16]. As such, it is crucial to ensure that the statistical estimates of MMEs are as unbiased and accurate as possible.

Often MMEs follow what Knutti et al. dub the "democratic approach" [9]. In this approach, all probablistic measures of an MME — including the mean, variance and quantiles of climatological variables — are computed by weighing each model output equally. Thus, it is implicitly assumed that all models in an MME are functionally independent and equally likely to accurately describe relevant climate phenomena [9, 10]. As such, the democractic approach is problematic for at least two reasons. First, model developpers will often share code, parametrization schemes, or ideas with one another. If many constitutive models in an MME share similar features, the democratic approach risks over-representing the forcing response and internal variability of such commonly shared schemes [5, 4]. Second, it is well established that certain models better represent and forecast climate processes than other models. Although it is unclear what it means for one model to be better than another *tout court*, certain models may better represent a specific climatic process than others, by, for instance, better agreeing with historical observations [10].

Due to these limitations of the democratic approach, researchers have investigated numerous alternatives. For instance, Tebaldi et al. have modelled the probability density functions associated with MMEs using a Bayesian framework to generate weighted estimates of regional temperature changes [17]. Researchers have also investigated methods to quantify model interdepdence. The so-called "a priori" approach to quantify model depedence directly compares the development histories or components of models to measure interdependence[1] [2]. Meanwhile, the "a posteriori" approach quantifies interdepdence by computing the distance betwen transformed model outputs by way of some metric [2, 12]. These methods can also generate so-called model "geneologies" or "family trees" using hierarchical clustering, as shown in figure 1

Recently, researchers at ETH Zurich including Reno Knutti, Ruth Lorenz, and Lukas Brenner have introduced a weighting framework for MMEs that accounts both for constitutive model performance and interdependence (see, for instance, [4, 5, 10, 11]). The framework, dubbed climWIP, first begins by defining distances between models and observations using an a posteriori approach described by Sanderson et al. [14, 15]. For a model ensemble and relevant reanalysis products, Sanderson and collaborators take the monthly mean gridded longitude/latitude data for radiative fluxes, surface temperatures, mean temperatures, and humidity data, regrid these outputs onto a 3.75*3.75 latitude longitude grid, and then vectorize these outputs [14, 15]. Hence empirical outcomes are also represented by a vector in some high dimensional space ([14, 15], see also [4]). These vectorized outputs are then embedded in a relatively low dimensional space using principal component analysis. The Euclidean distance between the points representing each model output is computed,

[1] See, for instance, [3, 7].

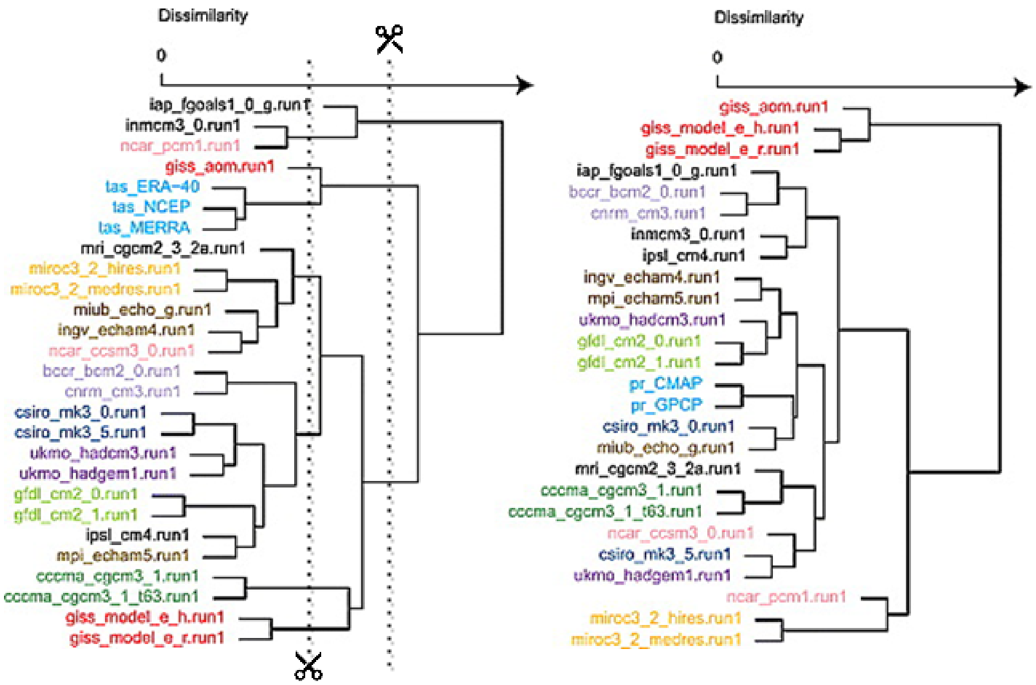

Figure 1: A modified version of figure 1 found in [12]. The figure is a CMIP3 modelling family tree for surface temperatures (left) and precipitation (right). The scissors are included to show how these family trees can be "cut" at particular dissimilarity measures, and thereby create families according to the branches of the resultant subtrees.

and yields a surrogate measure for model interdependence. Since empirical observations/reanalysis products are also represented in this low dimensional model space, model performance is quantified by the distance between model output and empirical observation vectors. A visual representation of the process is shown in figure 2.

ClimWIP then assigns a Gaussian weight w_i for each model m_i in an MME. These weights in turn depend on σ_D and σ_S: two tuneable parameters, determining the relative "closeness" of models to other models and observations [4]. These parameters are found using a rather complicated regression procedure described in §3.1 of [11].

In addressing the two pronged problem of model interdependence and performance, projections given by the climWIP ensemble are notably distinct from outputs of non-weighted MMEs. In particular, climWIP projects reduced temperatures of extreme summer weather events in North America up to 2100 [11] and quicker sea ice decline in the Arctic [10]. Perhaps most significantly, weights calibrated using climWIP for phase 6 of the Coupled Model Intercomparison Project ensemble (CMIP6) shows reductions in the mean and variability of end-of-century warming [4]. These projections are especially important when considering projections of end-of-century warming in CMIP6, since several models in CMIP6 are more sensitive to CO_2 forcing compared to previous CMIP phases (see [18]).

1.2 Weighing the Options

Although climWIP succesfully accounts for model performance and interdependence, we believe there are two limitations that confront the framework. First, climWIP casts all weights as Gaussian weights and thereby requires all weights be cast in a particular parametric form. Second, climWIP is explicitly based on Sanderson et al.'s a posteriori approach to quantifying model interdependence. Since intermodel and observational distances depend only on model outputs rather than shared components or history between models, climWIP cannot at present incorporate a priori interdependence measures[2]. While such measures are

[2] Although figure 5 in [4] shows that climWIP does a good job of representing a priori interdependence measures.

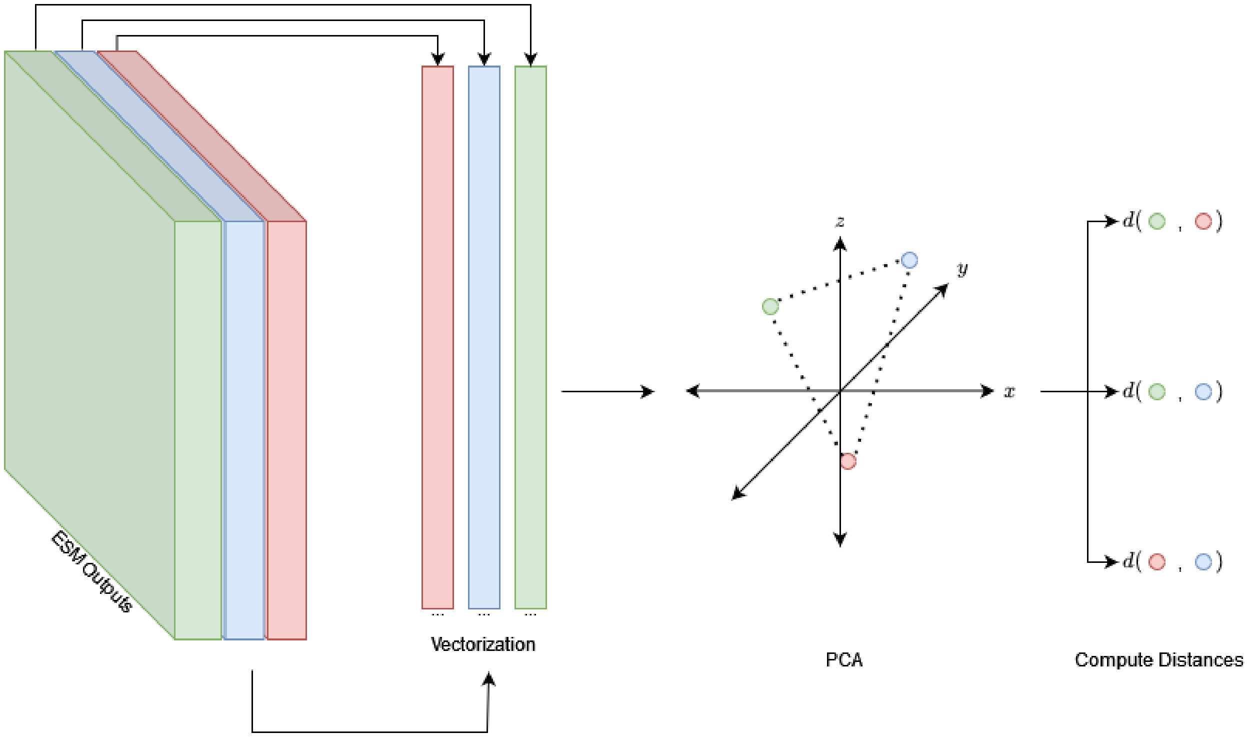

Figure 2: A visualization of the a posteriori approach described in Sanderson et al. [14, 15]. Note that the thickness of the ESM ouputs is only to convey that there are multiple grid outputs in the process. We also note that the loading of the PCA is three dimensional only so that the geometric meaning of the distance between model outputs is clear.

uncommon [2], it is nevertheless desirable to have a weighting scheme which could directly account for them. Succinctly, it is natural to wonder if performance and similarity based weight schemes can be extended to include interdependence measures other than those based on Sanderson et al.'s original 2015 papers [14, 15].

Here, we propose a general framework for weighing MMEs that accounts for both model performance and interdependence. The technique is inspired by regression analysis and machine learning techniques. More formally, our framework solves for weights model weights $\vec{w} = (w_1, ..., w_m)^T \in \mathbb{R}^n$ that minimize a loss function of the form

$$f(\vec{x}; \vec{w}, \vec{\lambda}) = E(\vec{x}; \vec{w}) + P(\vec{w}; \vec{\lambda})$$

where $\vec{\lambda}$ refers to a vector of parameters and hyperparameters alike. Here, $E(\vec{x}; \vec{w})$ corresponds to a function that is minimized when the weighted mean of ensemble outputs best predicts observational data. $P(\vec{w}; \vec{\lambda})$, meanwhile, is ideally a function that is maximized when weights are concentrated in "similar" models. Because the mechanism behind this framework is the combination of a regular cost function and a special penalty function, we dub it the *Penalty Weighted Framework* (PWF).

The key feature of this framework is that it can accommodate any penalty function $P(\vec{w}; \vec{\lambda})$, so long as $P(\vec{w}; \vec{\lambda})$ is differentiable almost everywhere[3]. Then given $f(\vec{x}; \vec{w}, \vec{\lambda}) = E(\vec{x}; \vec{w}) + P(\vec{w}; \vec{\lambda})$, optimal ensemble weights are set to $\vec{w}^* = \text{argmin} f(\vec{x}; \vec{w}, \vec{\lambda})$. And because $f(\vec{x}; \vec{w}, \vec{\lambda})$ is chosen to be differentiable almost everywhere, one can solve for local minima$\vec{w}^*$ iteratively via gradient descent [8]. The upshot is that PWF, if it is successful, can incorporate many different methods of quantifying model similarity and performance, since weights are solved for numerically rather than being explicitly parameterized.

2 Research Questions

We are ultimately interested in whether PWF as introduced in §1.3 can work. As such, our first and most important research question is as follows:

(Q1) Can a penalty weight framework find MME weights which 'reasonably' represent the performance of, and dependence between, models?

In order to test the performance claim, we rely on classic statistical inference and machine learning techniques. Using CMIP6 ensemble and MERRA-5 TAS data, we generate a training and test set, then solve for weights using gradient descent. We can then test the performance of several sets of weights by computing the accuracy of PWF weighted MMEs using a metric like the root mean squared error (RMSE). It is unclear to us how one could quantitatively test for an acccurate representation of "dependence" between models. This is also why 'reasonably' appears in scare-quotes: to emphasize that our means of testing for interdependence are not directly quantitative. Instead, our means of testing for interdependence will consider how weight rankings and magnitudes change when increasing the cost of a penalty functions. By examining these outputs, we can determine heuristically whether the PWF results in weights that are different than those generated when only considering performance (by considering, for instance, residuals given by a non-negative least squares regression).

We then compare penalty weighted MME outputs to democratically weighted MME outputs for SSP-2.6 TAS forecasts between 2015 and 2100. After all, Brunner et al. have shown that weighted MME outputs can significantly alter MME projections [4]. Hence our second research question is as follows:

(Q2) Assuming a 'reasonable' set of weights (per Q1), how do such forecasts differ from their unweighted counterparts?

Another question we will address concerns an immediate difficulty presented by the PWF: the delicate balance between the amount of data available versus the ability of datapoints to capture spatiotemporal relations in model outputs. As an example, the inputs to $f(\vec{x}; \vec{w}, \vec{\lambda})$ could be individual grid points in a monthly mean time series of some climatic variable for each MME model. However, any information about this input in relation to its neighboring grid cells is lost. On the other hand, if inputs to $f(\vec{x}; \vec{w}, \vec{\lambda})$ are all relevant climatic variables for grid cells averaged annually, then this will reduce the size of the training

[3] In the case where $w_i = 0$, we simply set $\partial_{w_i} P = 0$. This is normal practice in machine learning libraries like PyTorch and TensFlow for activation functions like ReLU which are not differentiable everywhere [1, 13].

dataset. This presents a difficulty, as gradient descent and machine learning techniques perform best when trained over large datasets [6]. Hence our final question concerns the best way to prepare model outputs in the GR framework:

(Q3) When implementing a weight penalty framework, is it better to average grid points at the risk of creating datasets that are too small? Or is it better to have a large training dataset, at the risk of removing important spatiotemporal information?

We address this concern by implementing a version of PWF on TAS data for monthly and annually averaged model outputs. We will consider the dataset which incurs the lower cost as the dataset which performs better at the weighting task.

3 Methods and Results

3.1 Methods

3.1.1 Data Preprocessing

We begin by preprocessing relevant CMIP6 ensemble method data so that it is compatible with the penalty weighted framework. That is, a set of vector inputs and scalar outputs placed into training and test sets. Our intial dataset consists of monthly TAS outputs from all CMIP6 ensemble models considered in [4] which are available on the KNMI Climate Explorer (https://climexp.knmi.nl). The model outputs range from 1980-2100 and are regridded onto a 90*45 longitude/latitude grid. These outputs are regridded both to ensure that spatiotemporal relations are well-represented in data samples, and so that computational costs remain inexpensive. We note that our dataset does not contain outputs of four models found in Brunner et al.'s ensemble (GFLD-ESM4, MPI-ESM1-2-HR, FGOALS-g3, CNRM-CM6-1), as these outputs are not available with the appropriate grid spacing on Climate Explorer. We also include monthly mean TAS data from the MERRA-2 reanalysis product to represent historical observations. Finally, we create a dataset of annual mean TAS projections by averaging gridded outputs for each year between 1980-2100.

For both the annual and monthly time series datasets, we vectorize temporally indexed outputs corresponding to times between Jan 1980 and Dec 2014. Each 4×4 matrix array corresponding to an index is converted to a 16 dimensional vector via row-major ordering. These 16 dimensional vectors are concatenated using their original indexing, so that model/reanalysis outputs are represented by a single vector.

After vectorization, we concatenate all model outputs, so that each vectorized output corresponds to a row of an $M \times N$ model output matrix. Here $M = 30$ corresponds to the number of vectorized outputs, and $N = 6720, 560$ corresponds to the dimension of vectorized outputs of our datasets, respectively. The matrix components are then standardized, where we take the variance of the entire matrix to be the variance of all its components. For each set of distinct models m_i and m_j, we compute the euclidean distance between their vectorized outputs, and store these distances as a component δ_{ij} of a 29×29 distance matrix δ. Unlike Sanderson et. al [14, 15], however, we do not perform PCA on vectorized outputs.

Finally, we can use the model output matrix to generate training and testing datasets. Assuming the last row of the model output matrix corresponds to MERRA-5 observations, we define the dataset $D := \{(\vec{x}_i, y_i)\}_{i=1}^N$. Here we take $\vec{x}_i$ to represent a feature vector of model outputs for some gridpoint, while y_i is the corresponding MERRA-5 value. We then shuffle elements of D and take the first 70% of datapoints as a training set, the next 10% as a validation set, and the final 20% as a test set.

3.1.2 Solving for Weights and Projections

With the training, validation and test datasets, we can solve for optimal weights using the PWF as described in §1.3 For this particular paper, we assume the loss function E corresponds to a non-negative least squares (NNLS) loss function

$$E(\vec{x}_1 ..., \vec{x}_N, y_1, ..., y_N; \vec{w}) = \frac{1}{2N} \sum_{i=1}^{N} \|y_i - \vec{w}^T x_i\|_2^2$$

where N denotes the size of the relevant dataset. Meanwhile, we consider two penalty functions $P(\cdot)$ dubbed the pointwise and family loss functions, as given in table 1. The pointwise function follows the a posteriori

Name	Penalty Function	Gradient (component k)	Approach
Pointwise	$P(\vec{w};\delta) = \sum_{i=1}^{M} \sum_{j>i} \frac{\lvert w_i w_j \rvert}{\delta_{ij}} + \sum_{i=1}^{M} w_i^2$	$\sum_{i \neq k}^{M} \frac{\lvert w_i \rvert}{\delta_{ki}}$	A posteriori
Family/Odds	$P(\vec{w};\mathcal{F}) = \sum_{F \in \mathcal{F}} \frac{\sum_{i \in F} \lvert w_i \rvert}{\sum_{j \notin F} \lvert w_j \rvert}$	$\sum_{F \in \mathcal{F}} \operatorname{sgn}(w_k) \left[\chi_{\{k \in F\}} - \frac{\chi_{\{k \notin F\}}}{(\sum_{j \notin F} \lvert w_j \rvert^2)} \right]$	A priori

Table 1: The penalty functions used in this paper. Here δ is a model distance matrix, F refers to model families, $\mathcal{F}$ to a set of model families and $\chi_{\{\cdot\}}$ refers to the indicator function.

Family	**Models Families**
MPI/MRI	AWI-CM-1-1-MR, MPI-ESM1-2-HR, MRI-ESM2-0
CESM/NESM	CESM2-WACCM, CESM2, NESM3
ACC & CO.	ACCESS-CM2, ACCESS-ESM1-5, HadGEM3-GC31-LL-f3, KACE-1-0-G, NorESM2-MM
CNRM/EC	CNRM-CM6-1-HR-f2, CNRM-ESM2-1-f2, EC-Earth3-Veg, EC-Earth3
INM	INM-CM4-8, INM-CM5-0
CAN	CanESM5-CanOE-p2, CanESM5-p1
MIROC	MIROC-ES2L-f2, MIROC6
SINGLE	BCC-CSM2-MR, CAMS-CSM1-0, FGOALS-f3-L, FIO-ESM-2-0, GISS-E2-1-G-p3, IPSL-CM6A-LR, MCM-UA-1-0, MPI-ESM1-2-LR

Table 2: Families of models as found in of [4]. Note that SINGLE is not itself family, but that each lister member is treated as a family with a single model.

approach, and therefore depends on the model distance matrix δ. The family loss function is based on the *a priori* approach, and therefore depends on a set of model families. Brunner et. al have provided such a set in figure 5 of [4], which we include in table 2. We note that even though climWIP is an *a posteriori* approach, these families describe models with "shared components or the same origin"[4] and are therefore well suited to our ends.

We implement gradient descent with early stopping to solve for optimal ensemble weights with respect to the family and pointwise penalty functions. We first assume that all models are weighted equally, and then iteratively update these weights. At iteration k, the total penalty weighted loss over the validation set is found, and the weight vector is updated as follows: $\vec{w}^{(t+1)} \leftarrow w^{(t)} - \alpha \nabla E - \alpha \lambda \nabla P$. Here α is the learning rate, and λ is a hyperparameter analogous to the weight decay parameter in L1/L2 regularization[4]. Once the validation loss has not decreased over 5 iterations, gradient descent terminates, and the weights are normalized.

Given a set of normalized optimal weights, we can then compare projections made by a democratically weighted CMIP6 ensemble, and our own PWF ensemble.

3.2 Results

3.2.1 Monthly vs Annual Means

We first consider whether weights trained on annual or monthly mean datasets perform better. To this end we implement the PWF using both datasets and compare the total loss function values. We first compare loss function values by performing a simple NNLS regression. We use RMSE as the relevant loss function, and compare the RMSE values of the training and validation sets for all iterations, as well as the test set error after solving for locally optimal weights.

As seen in figure 3, the annual average dataset performs better in both the NNLS task, and when finding optimal weights with the family penalty function (hyperparameter value $\lambda = 10$). This is further confirmed

[4] See §3.4 of [8] for an overview of L1/L2 regularization.

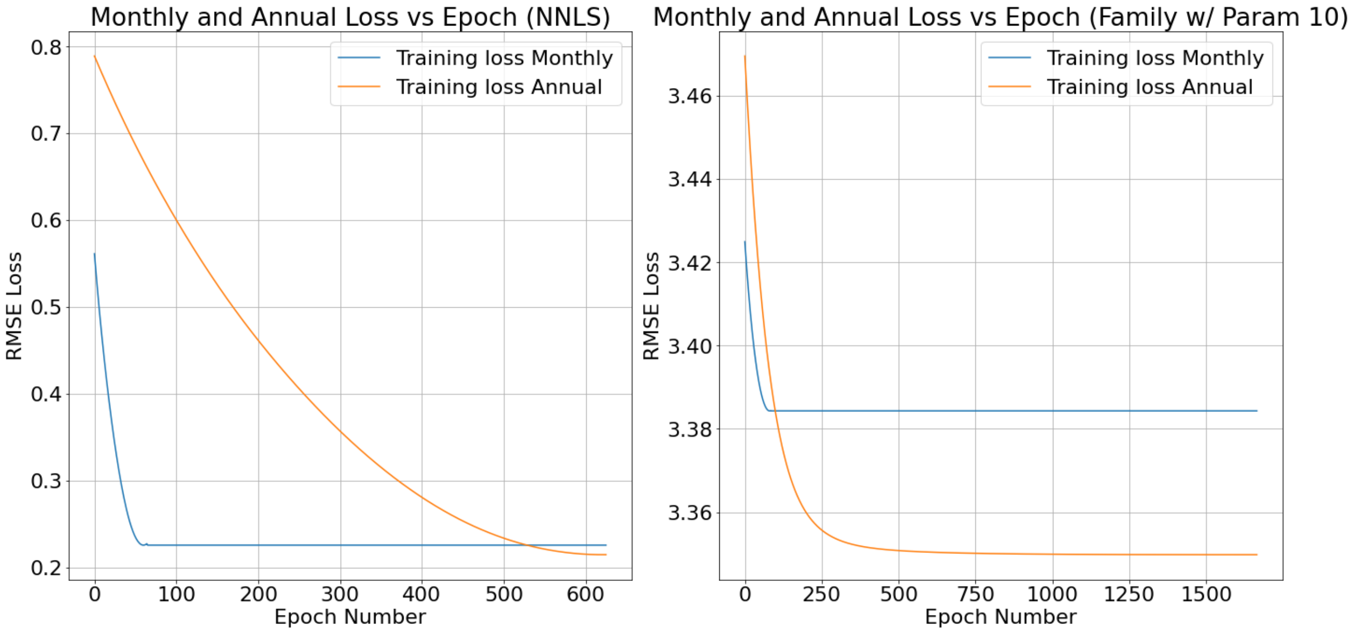

Figure 3: Plots of the training error for the monthly averaged TAS dataset (left) and annually averaged TAS dataset (right). The flatlines for the monthly loss occur because gradient descent stops earlier for the monthly dataset.

Task & Dataset	Test Loss
NNLS Monthly	0.3923
NNLS Annual	0.2664
Family Monthly	3.4000
Family Annual	3.3541

Table 3: Table of loss values of NNLS and family weighting tasks over relevant test sets.

by the test losses, which can be found up to 4 decimal places in in table 3. Therefore, we use the annual average dataset for the rest of our analysis, since the penalty weighting task performs better on this dataset.

3.2.2 Penalty Results

We first consider the weights given to each model if based purely on performance. The resulting weights are given in figure 4. Hence when purely considering the performance aspect of models, we see that the MIROC family is given a proportionally stronger weight than other models, while the INM family models have a comparatively small weight.

In order to determine the influence of penalty functions on optimal weights, we determine optimal weights when the penalty hyperparameter $\lambda = 1, 10, 100$. These values were chosen because the NNLS and penalty losses are usually the same order of magnitude per iteration when $\lambda = 1$. So optimal weights when $\lambda = 10, 100$ should clearly demonstrate the influence of our penalty functions.

From figures, notice when $\lambda = 10, 100$, the pointwise and family penalized weights no longer resemble the NNLS weights. In particular, the range of pointwise penalized weights becomes much larger, ranging from about ~ 0.05 to 0.15 when $\lambda = 100$. However, by far the heaviest weighted models are in the pointwise penalized weights are members of the MIROC family, as was the case for the NNLS weights (see figure 4). Another similarity between the pointwise penalized and NNLS weights are that the INM models remain in a similar position for our weight rankings. These results suggest that the pointwise penalty function may not

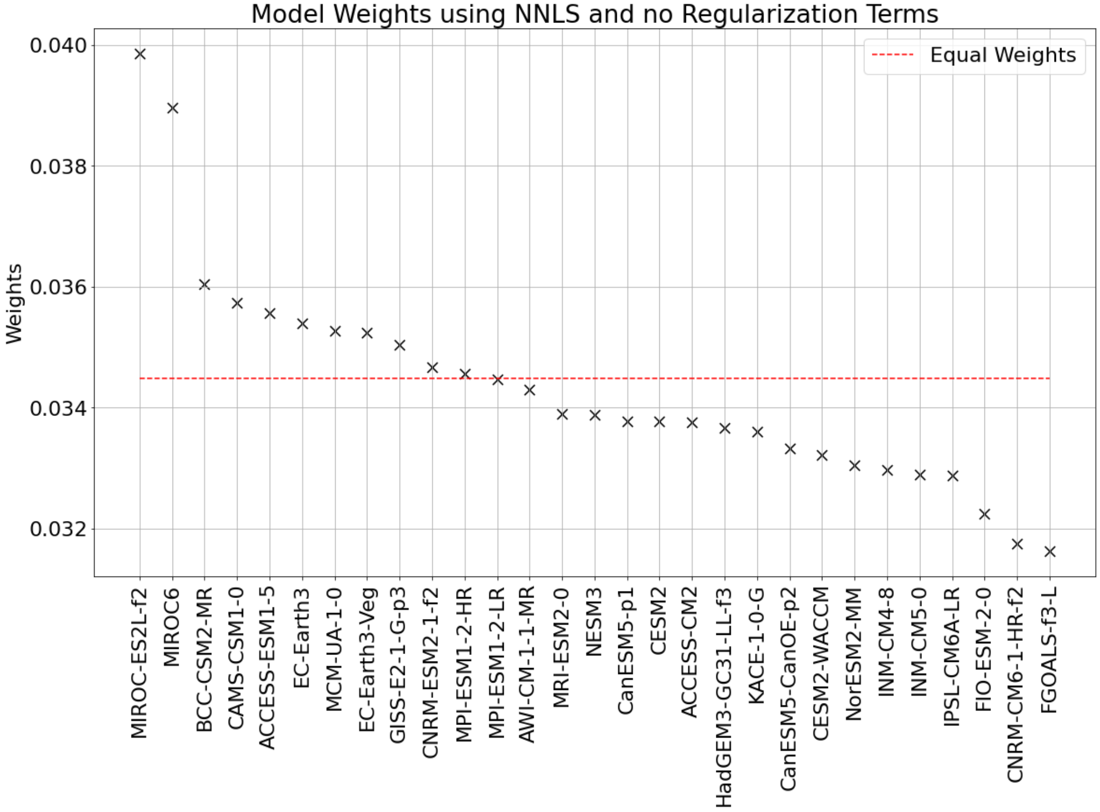

Figure 4: Weights of each CMIP6 model component with no penalty function. The weights correspond to the coefficients found using non-negative least squares. These weights are found via gradient descent learning rate $\alpha = 10^{-4}$.

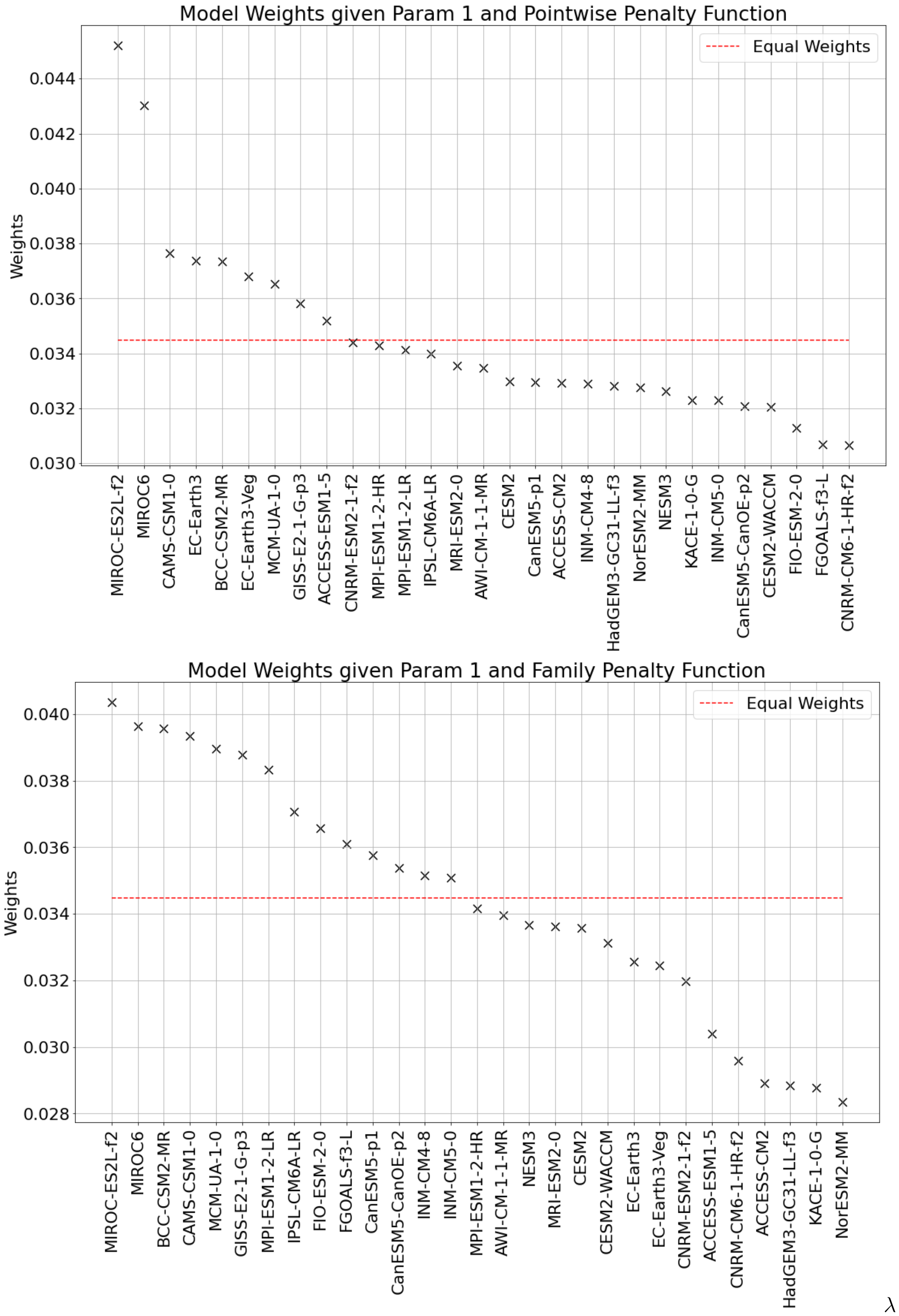

Figure 5: Weights of each CMIP6 model component when including the pointwise (top) and family (bottom) penalty functions with hyperparameter value $\lambda = 1$.

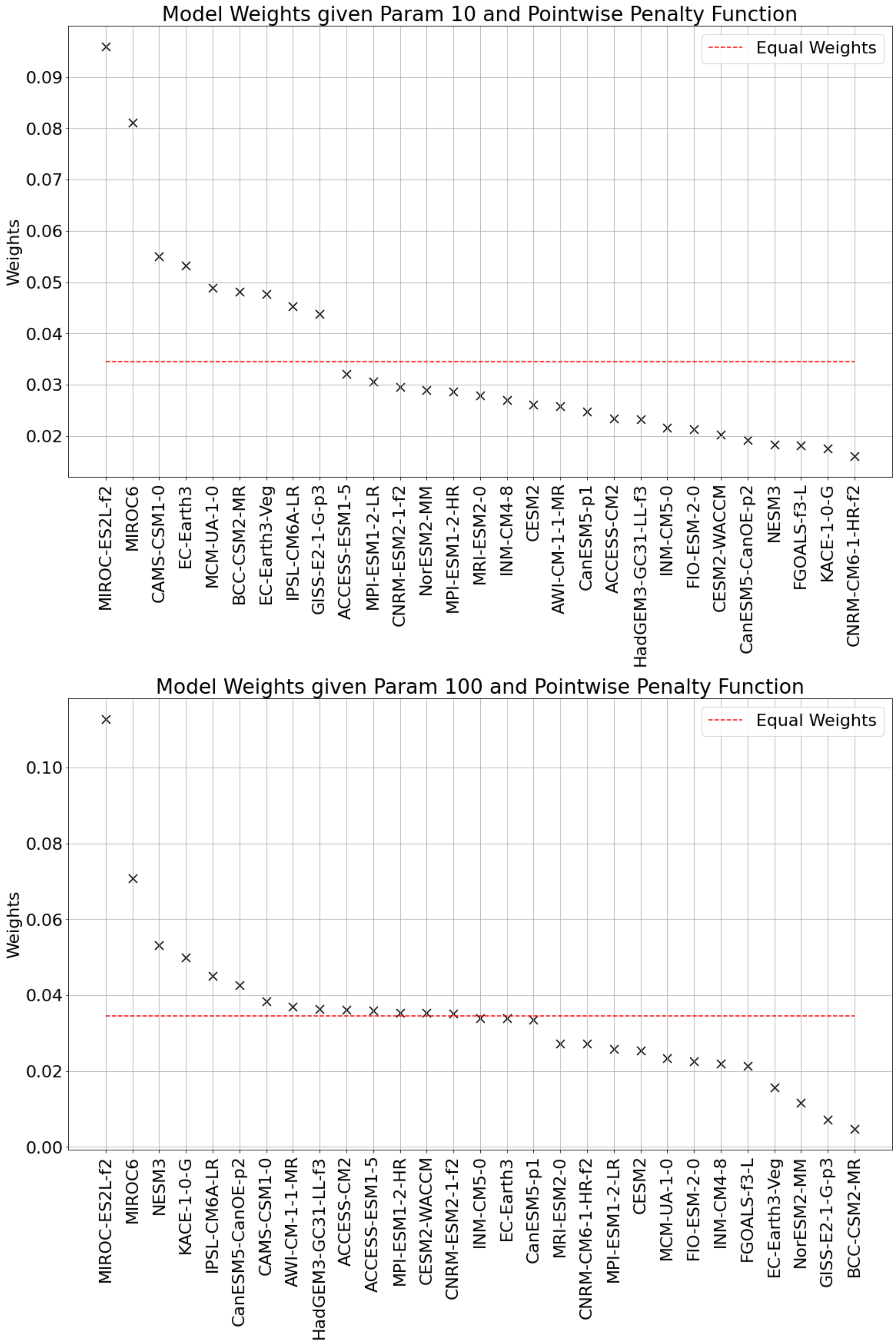

Figure 6: Weights of each CMIP6 model component when including the pointwise penalty function and hyperparameter values $\lambda = 10$ (top) and $\lambda = 100$ (bottom).

\lambda+

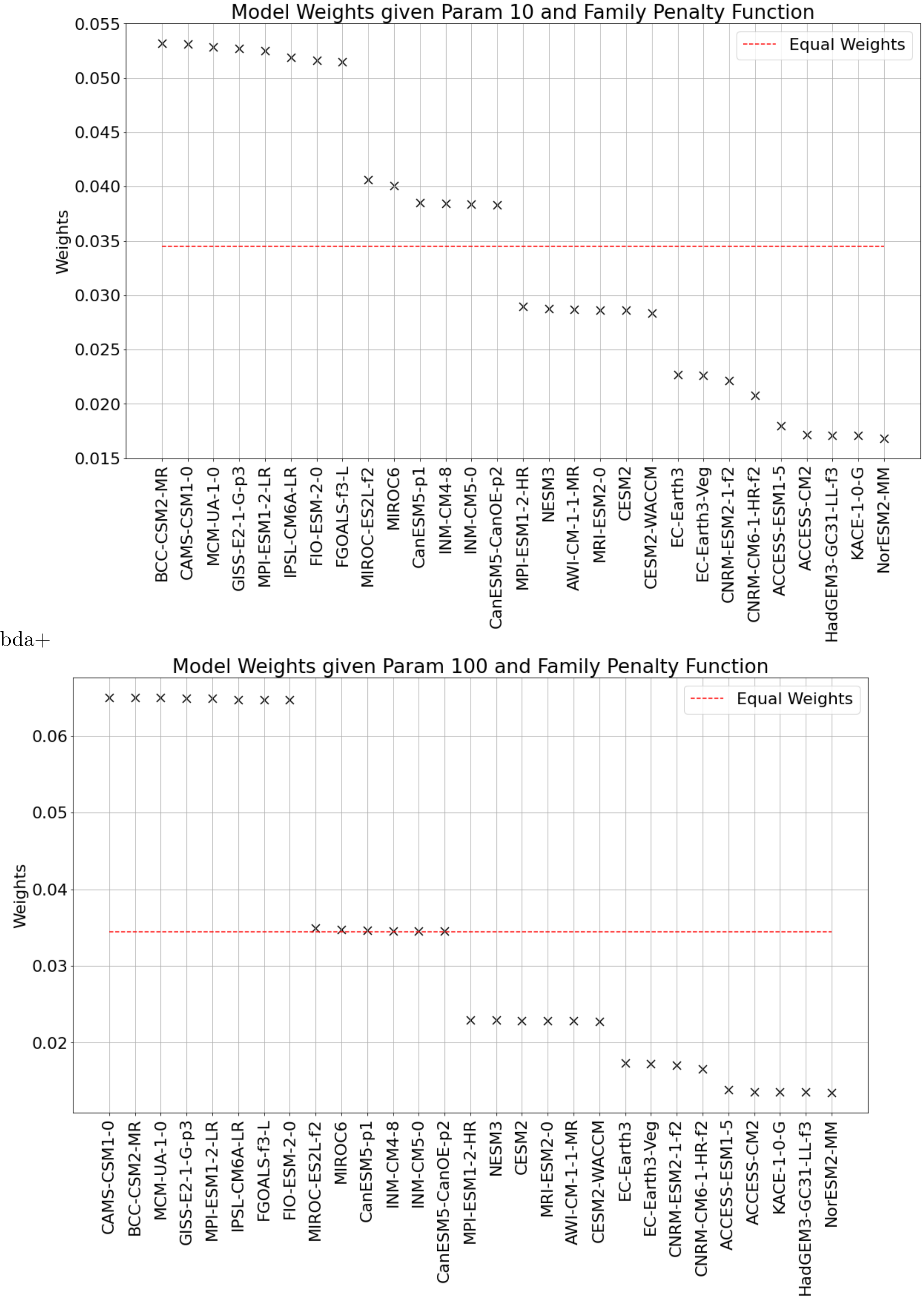

Figure 7: Weights of each CMIP6 model component when including the family penalty function and hyperparameter values $\lambda = 10$ (top) and $\lambda = 100$ (bottom).

actually account for model interdependence well. Indeed, we shall return to this point in §4. Meanwhile, the family weighted functions quickly group models together into five distinct clusters. Comparing the family penalized model weights for $\lambda = 100$ to table 2 reveals that the first cluster of models corresponds directly to models that are in families with single members (i.e. in SINGLE). The next cluster consists of models that are in families with two members (i.e. in CAN, INM or MIROC). This trend continues, and the optimal family weight for a given model almost entirely depends on the size of its family. So it is clear that as λ increases from 0 to 100, model weights transition from being entirely dependent on performance, to being almost entirely dependent on their familial interdependence.

Although the pointwise and family penalized weights lead are significantly different, both sets reduce ensemble TAS projections. Indeed, this reduction in warming is obvious even when $\lambda = 1$ for family penalized weights, as shown in figure 8. This result would provide additional support to [4] et al.'s conclusion that weighing models by performance and interdependence reduces ensemble mean TAS projections. However, as we will discuss in §4, we believe these results are limited in the case of family penalized weights, and ultimately uninformative when considering pointwise penalized weights.

4 Discussion

Returning to our research questions, we believe that the answers to (2) and (3) are straightforward. First, annually averaged data performs better on training and testing sets when implementing the penalty weighted framework. Second, if our sets of penalized weights genuinely capture the performance and interdependence of models in our CMIP6 ensemble, then these sets decrease the values of mean surface air temperature projections in the SSP1-2.6 pathway. Indeed, figure 8 shows that SSP-2.6 TAS projections are reduced when determining weights by either the pointwise penalty or the family penalty functions. Notice, however, that this conclusion is less robust than the conclusion of [4], where Brunner et la. argue that climWIP reduces global warming CMIP6 ensemble projections *tout court*. Indeed, Brunner et al.'s results are based on considering the means and 66% ranges over a spread of CMIP6 projections, including the SSP-2.6 and SSP5-8.5 pathways [4]. Our results, however, do not indicate anything about projection spread or variability, and are consequently quite limited.

In our view, answering our first research question is more complicated. If the question concerns whether the specific sets of weights found in one of figures 5-7 adequately capture model performance and interdependence, then we do not think this is the case. First, our training set consists entirely of TAS data. However, as noted by Knutti, different models are better at forecasting different variables [10]. Consequently, our dataset requires more climatic data to accurately assess model performance. Second, we believe the pointwise penalty function does not actually capture model dependence or performance well. Indeed we note that the pointwise function is directly proportional to the magnitudes of each weight. The pointwise function, and therefore the total cost of a set of weights, decreases whenever the magnitudes of all weights are decreased. This ensures that all weights decrease in size when increasing the hyperparameter λ, as seen in figure 8 (recall that the plotted weights in figures 5-7 are normalized). Furthermore, the distance term of the pointwise function $\sum_{i=1}^{M} \sum_{j>i} \frac{|w_i w_j|}{\delta_{ij}}$ is uniquely zero when all but one weight are set to zero. Consequently, we suspect the pointwise function encourages all weights except those performing well at the NNLS task to go to zero, which is precisely the opposite of our intention. These reasons suggest that our particular sets of weights do not adequately capture model performance and interdependence.

However, we believe our results demonstrate that PWF *could* generate 'reasonable' sets of weights in principle, even if the particular sets of weights given above are not sufficient. Indeed, we believe the family penalty function adequately captures some form of model interdependence given its limiting behaviour (see figure 7). Because the weights given when $\lambda = 0$ and $\lambda = 100$ are notably distinct, and because the weights when $\lambda = 100$ clearly correspond to weighing models by family size alone, our results suggest that for smaller values of $\lambda \simeq 1$, that the family penalty function is sensitive to both performance and interdependence. Perhaps this function would provide a useful sets of weights if trained on more robust data, and if there were a more systematic means for testing our rather slippery notion of 'reasonability'.

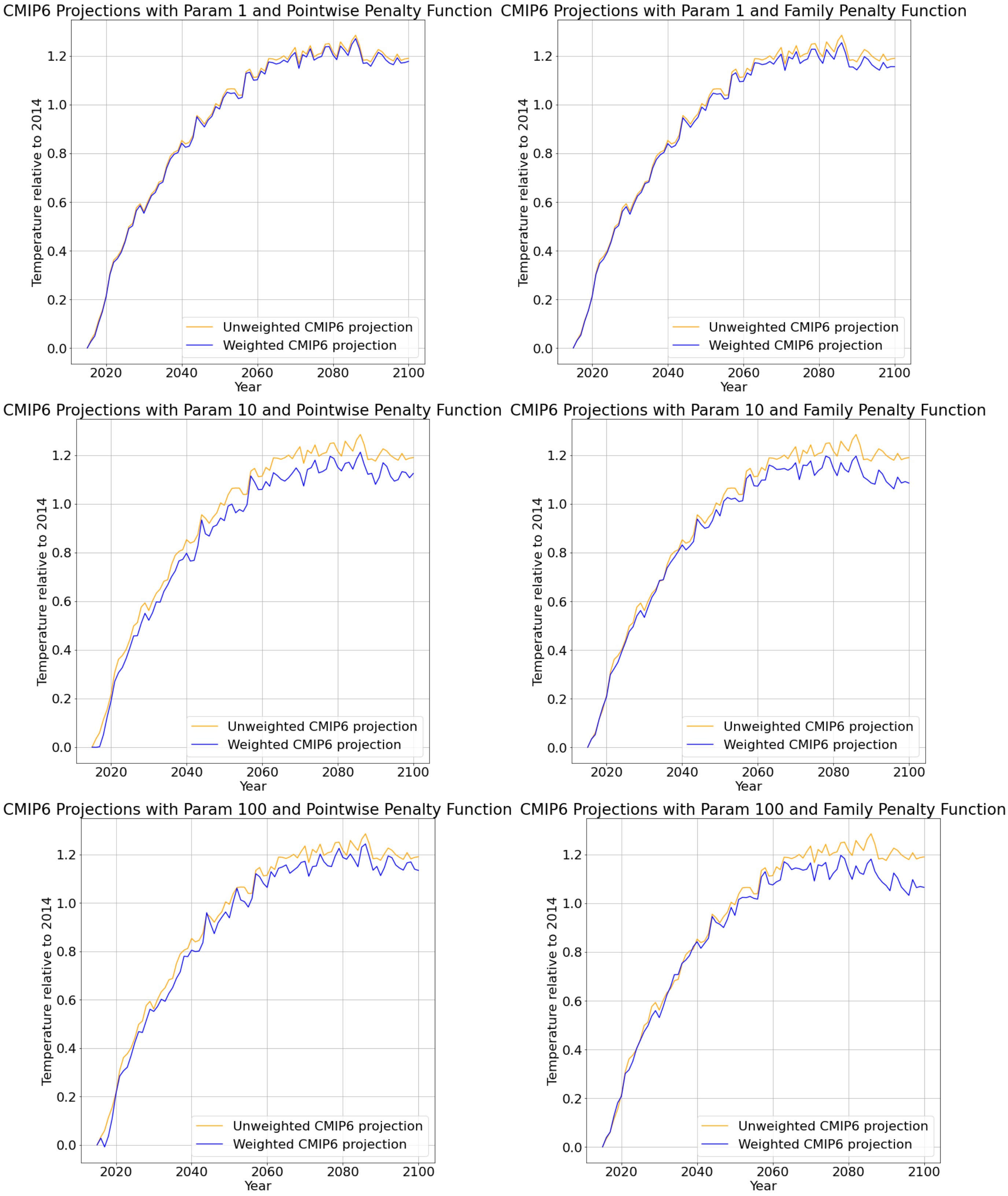

Figure 8: Ensemble mean TAS projections given by the pointwise and family penalty functions (left and right, respectively). The rows correspond to hyperparameter values of 1 (top), 10 (middle) and 100 (bottom) respectively.

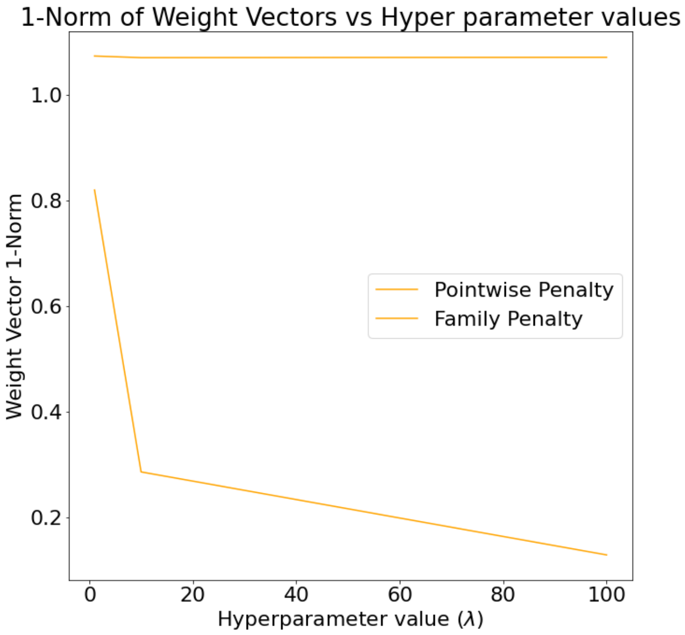

Figure 9: 1-Norm of unnormalized weights given by the pointwise and family penalty functions as a function of λ. Notice that the family penalty function is norm preserving, while the pointwise weights decrease in norm vs λ.

5 Conclusions

We have introduced the penalty weighted framework based on statistical learning techniques, as well as the previous work of Saunders et al., Lorenz et al., and Brunner et al. (see [4, 14, 15, 10]). This framework numerically solves for a set of optimal weights given the outputs of a multimodel ensemble, by minimizing a cost function considering both performance and model interdependence. We tested this framework using two candidate functions that penalize sets of weights not accounting for interdependence over monthly and annually averaged TAS data for low-resolution CMIP6 model outputs. Based on the resulting cost values and sets of weights, we found that both candidate functions performed better on annually averaged data vs monthly averaged data, and that all sets of weights reduced the average ensemble TAS projections for the SSP-2.6 pathway. Ultimately, our results are limited by the fact that we only considered mean projection outputs for a single pathway and a single climate variable, and because the pointwise penalty function does not capture model interdependence. However, the weights garnered by the family function suggests that the PWF is promising, even given the limited results of this paper.

Code Availability

All code is included at the following link: https://drive.google.com/file/d/1JoLAvvNU17SxFKw9YDI7AFsiRMyd5-E7 (fair warning: at present the code isn't very legible!).

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI16́, pages 265– 283, USA, 2016. USENIX Association.

[2] G. Abramowitz, N. Herger, E. Gutmann, D. Hammerling, R. Knutti, M. Leduc, R. Lorenz, R. Pincus, and G. A. Schmidt. Esd reviews: Model dependence in multi-model climate ensembles: weighting, sub-selection and out-of-sample testing. *Earth System Dynamics*, 10(1):91–105, 2019.

[3] Julien Boé. Interdependency in multimodel climate projections: Component replication and result similarity. *Geophysical Research Letters*, 45(6):2771–2779, 2018.

[4] L. Brunner, A. G. Pendergrass, F. Lehner, A. L. Merrifield, R. Lorenz, and R. Knutti. Reduced global warming from cmip6 projections when weighting models by performance and independence. *Earth System Dynamics*, 11(4):995–1012, 2020.

[5] Lukas Brunner, Ruth Lorenz, Marius Zumwald, and Reto Knutti. Quantifying uncertainty in european climate projections using combined performance-independence weighting. *Environmental Research Letters*, 14(12):124010, nov 2019.

[6] Google Developers. The size and quality of a data set | data preparation and feature engineering for machine learning, 7 2019.

[7] Paul N. Edwards. History of climate modeling. *WIREs Climate Change*, 2(1):128–139, 2011.

[8] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[9] Reto Knutti. The end of model democracy? *Climatic Change*, 102(3-4):395–404, 2010.

[10] Reto Knutti, Jan Sedláček, Benjamin M. Sanderson, Ruth Lorenz, Erich M. Fischer, and Veronika Eyring. A climate model projection weighting scheme accounting for performance and interdependence. *Geophysical Research Letters*, 44(4):1909–1918, 2017.

[11] Ruth Lorenz, Nadja Herger, Jan Sedláček, Veronika Eyring, Erich M. Fischer, and Reto Knutti. Prospects and caveats of weighting climate models for summer maximum temperature projections over north america. *Journal of Geophysical Research: Atmospheres*, 123(9):4509–4526, 2018.

[12] David Masson and Reno Knutti. Climate model genealogy. *Geophysical Research Letters*, 38(8), 2011.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[14] Benjamin M. Sanderson, Reto Knutti, and Peter Caldwell. Addressing interdependency in a multimodel ensemble by interpolation of model properties. *Journal of Climate*, 28(13):5150 – 5170, 2015.

[15] Benjamin M. Sanderson, Reto Knutti, and Peter Caldwell. A representative democracy to reduce interdependency in a multimodel ensemble. *Journal of Climate*, 28(13):5171 – 5194, 2015.

[16] Claudia Tebaldi and Reto Knutti. The use of the multi-model ensemble in probabilistic climate projections. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1857):2053 – 2075, 2007.

[17] Claudia Tebaldi, Richard Smith, Doug Nychka, and Linda Mearns. Quantifying uncertainty in projections of regional climate change: A bayesian approach to the analysis of multimodel ensembles. *Journal of Climate*, 18(10):1524 – 1540, 2005.

[18] Mark D. Zelinka, Timothy A. Myers, Daniel T. McCoy, Stephen Po-Chedley, Peter M. Caldwell, Paulo Ceppi, Stephen A. Klein, and Karl E. Taylor. Causes of higher climate sensitivity in cmip6 models. *Geophysical Research Letters*, 47(1):e2019GL085782, 2020. e2019GL085782 10.1029/2019GL085782.