Deep Learning for Climate Model Output Statistics

Michael Steininger[1]**, Daniel Abel**[2]**, Katrin Ziegler**[2]**, Anna Krause**[1]**, Heiko Paeth**[2]**, and Andreas Hotho**[1]

[1]Chair of Computer Science X (Data Science), University of Würzburg
`{steininger, anna.krause, hotho}@informatik.uni-wuerzburg.de`
[2]Chair of Physical Geography, University of Würzburg
`{daniel.abel, katrin.ziegler, heiko.paeth}@uni-wuerzburg.de`

Abstract

Climate models are an important tool for the assessment of prospective climate change effects but they suffer from systematic and representation errors, especially for precipitation. Model output statistics (MOS) reduce these errors by fitting the model output to observational data with machine learning. In this work, we explore the feasibility and potential of deep learning with convolutional neural networks (CNNs) for MOS. We propose the CNN architecture ConvMOS specifically designed for reducing errors in climate model outputs and apply it to the climate model REMO. Our results show a considerable reduction of errors and mostly improved performance compared to three commonly used MOS approaches.

1 Introduction

An important source of information for the prospective effects of climate change are numerical climate models such as general circulation models (GCMs) and regional climate models (RCMs). However, these climate models often exhibit systematic errors and deficiencies in representations of climate processes which limit the quality of the resulting projections. This problem is especially pronounced for precipitation. It is therefore common to apply model output statistics (MOS), which are statistical post-processing techniques to reduce these. MOS approaches correct the modeled precipitation to correspond more closely to observational data. This allows us to study future climate conditions and the effects of climate change more accurately especially at a local scale [1].

There are two general approaches to MOS – distribution-wise MOS and event-wise MOS. Distribution-wise MOS corrects the distribution of the simulated variable by mapping means and other distribution characteristics to the observational distribution. Event-wise MOS directly links the simulated and observed time series through statistical models, which generally performs better than distribution-wise MOS [2]. We therefore consider event-wise MOS in this work.

A number of approaches to event-wise MOS have been used in previous work. A very simple approach is local scaling where an individual Linear Regression is fitted per location of interest, which has shown to work reasonably well [2]. Other works propose non-local MOS approaches, where for each location the MOS is aware of the climatic conditions at nearby locations. This can lead to a large number of predictors for the MOS, which is why dimensionality reduction techniques, e. g. principal component analysis (PCA), are often applied [1, 2, 3, 4]. Non-local MOS has been done with a range of machine learning models namely Linear Regression [1, 2], Random Forests (RFs) [3, 4], Support Vector Machines (SVMs) [3, 5, 6], and Multilayer Perceptrons (MLPs) [7].

While these methods have proven to be effective, we believe that there is considerable potential in exploring the use of advanced deep learning techniques. Especially convolutional neural networks (CNNs) [8] have shown proficiency in tasks with geospatial data [9, 10], which indicates potential for

novel non-local MOS with this type of neural network. We believe that their ability to learn spatial patterns is well suited for reducing systematic errors in climate models. It is therefore promising to assess how this can be used for MOS and whether there is potential for performance improvements.

In this work, we examine the feasibility and potential of convolutional deep learning models as MOS. Thus, we propose the CNN architecture ConvMOS specifically designed for climate MOS and apply it to correcting simulated precipitation of the RCM REMO [11, 12, 13]. Our results show that ConvMOS can reduce errors considerably, providing mostly better performance than three commonly used MOS approaches. This suggests that our proposed approach is feasible and promising.

2 Dataset

Model Data   For our study we use daily data of the hydrostatic version of the RCM REMO (version REMO2015) [11, 12, 13] for the period 2000 to 2015. Our study area has been defined over an extended German region with $0.11° × 0.11°$ resolution covering the area from $−1.43°$ to $22.22°$ E and $42.77°$ to $57.06°$ N (GER-11). We use the following 22 MOS predictors from REMO: Daily mean, minimum and maximum temperature $2\,\mathrm{m}$ above surface [K], u- and v-wind $10\,\mathrm{m}$ above surface [$\mathrm{m\,s^{-1}}$], sea level pressure [Pa] and total precipitation (convective + large scale + snowfall) [mm]. Further, the temperature [K], geopotential height [m], and specific humidity [$\mathrm{kg\,kg^{-1}}$] in the pressure levels of 100, 200, 500, 850 and 950 hPa are used. For elevation information the dataset GTOPO $(0.009° × 0.009°)$ [14, 15] is used by REMO, which we also employ as another predictor for ConvMOS. More specifics about the climate model can be found in Appendix A.

Observational Data   For observational data we use the gridded dataset E-OBS [16] version 19.0e which is based on an ensemble of interpolated station data [17]. Since the station density varies in space and time, the interpolation of the station data has some uncertainty [17]. Amongst other variables E-OBS provides daily precipitation sums at a $0.1°$ resolution, which is our predictand. The grids of the model and observational data are interpolated bilinearly to the same $0.11°$ grid [18].

3 Deep Learning for Climate Model Output Statistics

To explore the use of deep learning and CNNs as MOS we propose the architecture ConvMOS.

Idea   The basic idea of ConvMOS stems from two potential sources of error in climate models: First, specific location errors which typically stem from poor grid point representation of topography [1, 2]. Second, systematic errors originating from parameterization, which replaces too complex or too small-scale processes with simplified variants. For precipitation, cloud and rainfall formation is based on parameterization, leading to an overestimation over land [1].

To efficiently reduce both types of errors we combine per-location model parameters, which can learn the characteristics of a specific location, and global model parameters, which can learn spatial precipitation patterns to efficiently help reduce systematic errors in climate models. Thus, we define two module types: Local network and global network.

Local Network   The local network module contains individual model parameters for each location in the study area, allowing it to reduce specific local errors. It is implemented with a linearly activated 1D CNN where the input at each time is first reshaped so that it has the dimensions $(\text{height} * \text{width}, \text{predictors})$ instead of $(\text{predictors}, \text{height}, \text{width})$. In conjunction with setting the kernel size equal to the number of predictors, this allows us to group the convolution for each input channel (i.e. each location) so that each location is convolved with its own set of filters for all predictors. Thus each location has its own model parameters, in which location characteristics can be encoded. This module is not provided with elevation data as it would be static across all times for each location. The output of the local network is a grid with precipitation residuals for each location.

Global Network   The global network learns spatial patterns in precipitation and other predictors. This can be done efficiently with CNNs [19]. The module contains a 2D CNN with 4 layers which learns useful filters for the reduction of systematic errors across the study area using all predictors. Starting from the first layer, the layers have 4, 8, 16, and 1 filters and kernel sizes of 9, 1, 5, and 3
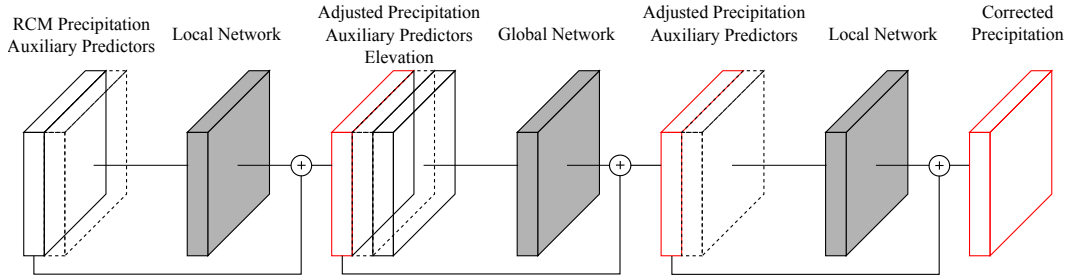
Figure 1: Architecture of ConvMOS.

respectively. Each convolutional layer has its padding parameter set to half its kernel size (rounded down to the nearest whole number) which leads to the output of each layer having the same width and height as its input. All layers use the ReLU [20] activation function, a stride of 1, and a dilation of 1. As with the local network, this module also outputs a grid of precipitation residuals.

Architecture   The architecture is depicted in Figure 1. It expects a 3D input with dimensions $(predictors, height, width)$ for each time step. The data is sequentially passed through three modules (depicted in gray) where each module adjusts the precipitation input with the goal of reducing the error. The architecture employs so called "shortcut connections" for each module where each module's output is added to its precipitation input, which eases training for neural networks [21]. We first apply a local network to correct any specific local errors. Thereafter, the global network aims to reduce any remaining systematic errors across the study area. Finally, another local network makes sure that the systematic corrections of the global network are not introducing new local errors. The training procedure is described in Appendix B.

4   Experiment

To evaluate ConvMOS we apply it to the data described in Section 2. We also apply three other commonly used MOS approaches, a local Linear Regression, a non-local Principal Component Regression approach and a non-local RF method, for comparison.

Local Linear Regression (Lin)   For each cell in the study area a separate Linear Regression is fitted where the predictor is the simulated precipitation and the predictand is the observed precipitation. This approach is local in that each Linear Regression is unaware of conditions in nearby cells [2].

Non-local Principal Component Regression (NL PCR)   Instead of only using the large-scale conditions at a specific location for a Linear Regression, we provide all available predictors at each nearby location which is at most ± 5 cells away in either direction on the grid. To reduce the dimensionality of the predictors, we apply a supervised PCA [22], which is explained in Appendix D [2].

Non-local Random Forest (NL RF)   For the non-local Random Forest MOS approach we provide all available predictors of each location ± 5 cells away, as with the non-local PC regression approach. Following [3] and [4] we also apply a supervised PCA (see Appendix D). Each location is fitted with its own RF. Hyperparameters are optimized at each location individually (see Appendix C).

Experimental Setup   We split the 16 years of daily data into a training (2000–2009), a validation (2010), and a test set (2011–2015). All predictors are standardized based on the training set so that they have a mean of zero and a standard deviation of one. We tried different hyperparameters for our architecture and selected the ones presented in this work based on the validation set performance. All results reported in the following are based on the unseen test set after fitting the MOS on the training set. For evaluation we use a number of common MOS metrics, namely RMSE, Pearson Correlation, Skill Score [23], R^2, and bias to assess different aspects of MOS performance. ConvMOS is trained 10 times since its fitting method is, in contrast to the linear comparison methods, non-deterministic. This results in slight performance differences for each fitted instance. Despite its inherent randomness, the RF method is only calculated once since this computation already took over four days for our study area with 15 CPU cores in parallel.

3

Table 1: Experimental results. Mean metrics on the test set for all study area locations available in observational data. All means and standard deviations are rounded to two decimal places. Correlation mean is calculated with Fisher's z-transformation [24].

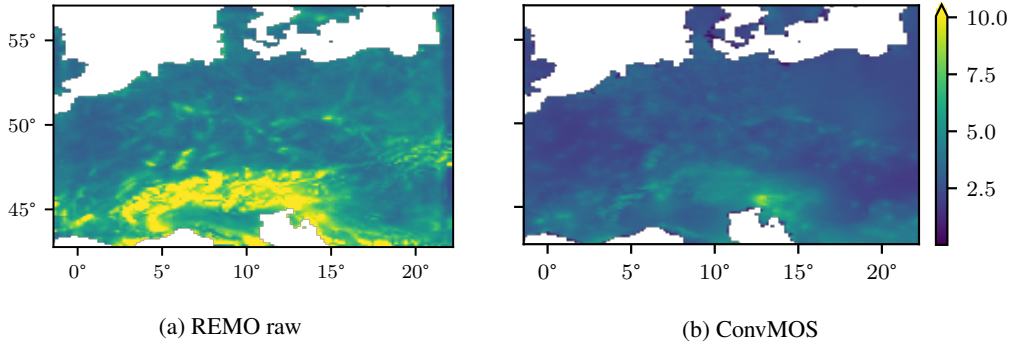| Metric MOS	RMSE	Corr.	Skill	R^2	Bias
None	5.32	0.49	**0.93**	−28.24	0.31
Lin	3.77	0.49	**0.93**	0.23	0.03
NL PCR	3.37	0.62	0.92	0.36	**0.02**
NL RF	3.39	0.61	0.81	0.36	0.03
ConvMOS	**2.99 ± 0.01**	**0.72 ± 0.00**	0.92 ± 0.00	**0.49 ± 0.01**	−0.10 ± 0.06

(a) REMO raw

(b) ConvMOS

Figure 2: RMSE of precipitation in mm for the test set across the study area. Note that there are some cells in REMO raw with considerably higher RMSE than 10 mm but we limited the colorbar's extent for better visibility of the general performance.

Results   Table 1 shows the mean metrics on the test set for all study area locations available in observational data (i.e. land points). All MOS approaches improve all metrics considerably when compared to applying no MOS, except for the skill score. This means that the precipitation distribution of REMO is already rather close to that of the observations with a skill score of 0.93 and can barely be improved by the MOS methods. ConvMOS is showing the best performance of all tested MOS approaches for the metrics RMSE, correlation, and R^2. This indicates that our approach is able to estimate precipitation more accurately than all considered comparison methods. The skill score is very close but still reduced slightly by 0.01 compared to the best value. ConvMOS shows less bias than REMO but it seems to have a tendency to underestimate precipitation. The other approaches to tend to overestimate, but to a lesser extent. ConvMOS is also showing rather stable performance as can be seen on the standard deviations in Table 1 despite its non-deterministic fitting procedure. We also ran this experiment with precipitation as the only climate predictor as some prior work has done [2, 3, 4, 6] but found all methods to perform worse without additional predictors.

Figure 2 visualizes RMSEs for all locations with observational data across the study area for the raw REMO output and ConvMOS. We can see that our approach reduces error across most locations. Especially the precipitation in the Alps and other mountainous regions is improved considerably.

5   Conclusion

In this work we explored the feasibility and possibilities of deep learning MOS. To this end, we proposed the CNN-based ConvMOS architecture specifically designed to reduce errors in climate model outputs, which we applied to the RCM REMO. All in all, the initial results for ConvMOS seem promising. Our MOS approach is able to improve the daily precipitation data considerably. Improvements in MOS allow for more accurate climate data especially at high spatial resolutions. While our approach mostly provides better performance than the other standard approaches considered here, additional comparisons have to be made in the future with other MOS techniques and data from different climate models.

Acknowledgments and Disclosure of Funding

References

[1] Heiko Paeth. Postprocessing of simulated precipitation for impact research in west africa. part i: model output statistics for monthly data. *Climate Dynamics*, 36(7-8):1321–1336, 2011.

[2] Jonathan M Eden and Martin Widmann. Downscaling of gcm-simulated precipitation using model output statistics. *Journal of Climate*, 27(1):312–324, 2014.

[3] Zulfaqar Sa'adi, Shamsuddin Shahid, Eun-Sung Chung, and Tarmizi bin Ismail. Projection of spatial and temporal changes of rainfall in sarawak of borneo island using statistical downscaling of cmip5 models. *Atmospheric research*, 197:446–460, 2017.

[4] Muhammad Noor, Tarmizi bin Ismail, Shahid Ullah, Zafar Iqbal, Nadeem Nawaz, and Kamal Ahmed. A non-local model output statistics approach for the downscaling of cmip5 gcms for the projection of rainfall in peninsular malaysia. *Journal of Water and Climate Change*, 2019.

[5] Sahar Hadi Pour, Shamsuddin Shahid, Eun-Sung Chung, and Xiao-Jun Wang. Model output statistics downscaling using support vector machine for the projection of spatial and temporal changes in rainfall of bangladesh. *Atmospheric Research*, 213:149–162, 2018.

[6] Kamal Ahmed, Shamsuddin Shahid, Nadeem Nawaz, and Najeebullah Khan. Modeling climate change impacts on precipitation in arid regions of pakistan: a non-local model output statistics downscaling approach. *Theoretical and Applied Climatology*, 137(1-2):1347–1364, 2019.

[7] Sanaz Moghim and Rafael L Bras. Bias correction of climate modeled temperature and precipitation using artificial neural networks. *Journal of Hydrometeorology*, 18(7):1867–1884, 2017.

[8] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[9] Xingjian Shi, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Deep learning for precipitation nowcasting: A benchmark and a new model. In *Advances in neural information processing systems*, pages 5617–5627, 2017.

[10] Michael Steininger, Konstantin Kobs, Albin Zehe, Florian Lautenschlager, Martin Becker, and Andreas Hotho. Maplur: Exploring a new paradigm for estimating air pollution using deep learning on map images. *ACM Transactions on Spatial Algorithms and Systems (TSAS)*, 6(3):1–24, 2020.

[11] Detlev Majewski. The europa-modell of the deutscher wetterdienst. *ECMWF Proc." Numerical Methods in atmospheric models." Reading*, 2:147–191, 1991.

[12] Daniela Jacob. A note to the simulation of the annual and inter-annual variability of the water budget over the baltic sea drainage basin. *Meteorology and Atmospheric Physics*, 77(1-4):61–73, 2001.

[13] D Jacob, BJJM Van den Hurk, Ulf Andrae, G Elgered, C Fortelius, LP Graham, SD Jackson, U Karstens, Chr Köpken, R Lindau, et al. A comprehensive model inter-comparison study investigating the water budget during the baltex-pidcap period. *Meteorology and Atmospheric Physics*, 77(1-4):19–43, 2001.

[14] EDC DAAC. Gtopo 30 database. eros data center distributed active archive center. *US Geological Survey, EROS Data Center, Sioux Falls, South Dakota*, 1996.

[15] Dean B Gesch, Kristine L Verdin, and Susan K Greenlee. New land surface digital elevation model covers the earth. *Eos*, 80(6):69–70, 1999.

[16] M R Haylock, N Hofstra, A M G Klein Tank, E J Klok, P D Jones, and M New. A European daily high-resolution gridded data set of surface temperature and precipitation for 1950-2006. *Journal of Geophysical Research Atmospheres*, 113(20):D20119, 2008.

[17] Richard C Cornes, Gerard van der Schrier, Else JM van den Besselaar, and Philip D Jones. An ensemble version of the e-obs temperature and precipitation data sets. *Journal of Geophysical Research: Atmospheres*, 123(17):9391–9409, 2018.

[18] Uwe Schulzweida. CDO User Guide, 2019.

[19] Thomas Vandal, Evan Kodra, Sangram Ganguly, Andrew Michaelis, Ramakrishna Nemani, and Auroop R Ganguly. Deepsd: Generating high resolution climate change projections through single image super-resolution. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 1663–1672, 2017.

[20] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Eric Bair, Trevor Hastie, Debashis Paul, and Robert Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137, 2006.

[23] SE Perkins, AJ Pitman, NJ Holbrook, and J McAneney. Evaluation of the ar4 climate models' simulated daily maximum temperature, minimum temperature, and precipitation over australia using probability density functions. *Journal of climate*, 20(17):4356–4376, 2007.

[24] N Clayton Silver and William P Dunlap. Averaging correlation coefficients: Should fisher's z transformation be used? *Journal of Applied Psychology*, 72(1):146, 1987.

[25] Erich Roeckner, Klaus Arpe, Lennart Bengtsson, M. Christoph, Martin Claussen, Lydia Dümenil, Monika Esch, Marco Giorgetta, Ulrich Schlese, and Uwe Schulzweida. The atmospheric general circulation model ECHAM-4: Model description and simulation of present-day climate. Technical report, Max-Planck-Institute of Meteorology, Hamburg, 1996.

[26] Stefan Hagemann. An Improved Land Surface Parameter Dataset for Global and Regional Climate Models. Technical Report 336, Max Planck Institute for Meteorology, Hamburg, 2002.

[27] Tido Semmler. *Der Wasser- und Energiehaushalt der arktischen Atmosphäre*. PhD thesis, Hamburg, 2002.

[28] Sven Kotlarski. *A Subgrid Glacier Parameterisation for Use in Regional Climate Modelling*. PhD thesis, Hamburg, 2007.

[29] Joni-Pekka Pietikäinen. The regional aerosol-climate model REMO-HAM. *Geoscientific Model Development*, 5:1323–1339, 2012.

[30] D Dee, S Uppala, A Simmons, P Berrisford, P Poli, S Kobayashi, U Andrae, M A Balmaseda, G Balsamo, P Bauer, P Bechtold, A Beljaars, L van den Berg, J Bidlot, N Bormann, C Delsol, R Dragani, M Fuentes, A J Geer, L Haimberger, S B Healy, H Hersbach, E V Holm, L Isaksen, P Kallberg, M Köhler, M Matricardi, A P McNally, B M Monge-Sanz, J.-J. Morcrette, B.-K. Park, C Peubey, P de Rosnay, C Tavolato, J.-N. Thépaut, and F Vitart. The ERA-Interim reanalysis: configuration and performance of the data assimilation system. *Quarterly Journal of the Royal Meteorological Society*, 137:553–597, 2011.

[31] Paul Berrisford, D Dee, K Fielding, M Fuentes, P Kallberg, S Kobayashi, and S Uppala. The ERA-Interim Archive. Version 2.0. Technical report, ECMWF, Reading, 2011.

[32] Claas Teichmann. *Climate and Air Pollution Modelling in South America with Focus on Megacities*. Phd thesis, Hamburg, 2010.

[33] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[34] Rich Caruana, Steve Lawrence, and C Lee Giles. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems*, pages 402–408, 2001.

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Table 2: Search space for the RF hyperparameter random search.

Hyperparameter	Search space
n_estimators	$10 - 2000$
max_features	$0.01 - 1.0$
max_depth	$10 - 110$
min_samples_split	$2 - 10$
min_samples_leaf	$1 - 10$
bootstrap	True or False

A   Climate Model Data

For our study we use daily data of the hydrostatic version of the RCM REMO (version REMO2015) [11, 12, 13] for the period 2000 to 2015. REMO is based on the Europa Modell [11] with the model of the GCM ECHAM4 [25] with some improvements implemented since then (e.g. [26, 27, 28, 29]). The reanalysis ERA-Interim ($0.75° \times 0.75°$) [30, 31] is used as forcing data, providing the lateral boundary conditions. The atmosphere's vertical resolution is represented by 27 hybrid levels with increasing distance to the top of the atmosphere. In lower levels they follow the topography [32]. As mentioned in the main paper the dataset GTOPO ($0.009° \times 0.009°$) [14, 15] is used by REMO for elevation information. Both model and observational data for the MOS methods is provided at $0.11°$ resolution. The data is arranged on a 2D grid with 121×121 cells or locations.

B   ConvMOS Training Details

The architecture is fitted with the Adam optimizer [33], the mean squared error (MSE) as the loss function and a learning rate of 0.001. Only errors at locations where observational data is available were incorporated for the MSE. Training is conducted for at most $100\,000$ epochs. Early stopping is used to stop training when the validation MSE is not improving for more than 40 epochs in a row, preventing considerable overfitting [34].

C   Random Forest Hyperparameter Optimization

Each location in our study area has its own RF instance for MOS which uses the RandomForestRegressor from scikit-learn [35]. Since RF performance depends considerably on its hyperparameters we look for optimal values with a random search. For each cell we train 20 RF instances on the training set with hyperparameter values sampled randomly from the search space shown in Table 2. Each instance is evaluated on the validation set. The RF instance with the best R^2 is then applied on the test set.

D   Supervised Principal Component Analysis

Like other previous MOS approaches [1, 3, 4] we preprocess our predictors for the standard MOS methods to reduce dimensionality and remove potentially unhelpful information. Like [3] and [4] we use supervised PCA [22]. First, we select the best predictors based on a univariate regression. How many of the predictors are retained is set according to a grid search with our validation data. In this search we try all values between only choosing the single best predictor and using the 30 best predictors. Then, PCA reduces the dimensionality of these predictors, keeping the first components that explain $95\,\%$ of the variance [3].