In this paper the authors address the question of how to identify useful weather regimes, which are both predictable in current forecasting systems and which do a good job at explaining variability in a target field of interest: in this case extreme wintertime rainfall over Morocco. Much ink has been spilled in the past 20 years on the topic of which weather regimes to use, how to define them, etc. I find the approach the authors have taken here refreshing and exciting. The use of a machine learning approach based on a variational autoencoder architecture to perform this task is, as far as I am aware, a novel innovation in this area, putting aside the authors' recent related paper. The focus on the practical value of the regimes for forecasting is also conceptually clarifying and provides a good motivation for the work.
The authors compare their CMM-VAE regimes to both a classical K-means approach and a surface-weather targeted CCA approach, and show that their method explains the most categorical rainfall variability while remaining similarly well predicted as the classical regime, with only a slight decrease in persistence and separability. They then go on to investigate sources of predictability, linking the predictability of the regime patterns to the polar vortex and Madden Julian oscillation.

I really like this work, and my only comments are editorial in nature; I believe the paper is publishable after minor revisions to increase clarity and to fix a few small typos.

Josh Dorrington

Minor points:
L55: predictable → predictability

You don't actually motivate why you switch from continuous rainfall to categorical precip clusters in this work. Was there a reason beyond simply proof of concept? Maybe a comment around L65 on this?

L141: Just out of curiosity, can you tell us what other regions you tested this on?

The numbering for appendix plots has gone wrong.

You mention a few times that this is a generative model: is there any value in the generative aspect here? I can't immediately think of one, but if you have thoughts perhaps share them in the conclusion?

I don't think its an important enough issue to require any changes here, but its worth bearing in mind for any future work that CCA can behave unreliably for correlated data, and that the closely related PLSR is more stable in this regard (differing only in that it maximixes cross covariance rather than cross correlation) https://arxiv.org/abs/2107.06867. Anecdotally, the scikit learn implementation of PLSR seems to handle full field data fine, so you could perhaps have avoided the district aggregation and ridge regularization process.

Clarification of the ML architecture
My main comment is that the description of the CMM-VAE architecture is quite opaque and not quite self-standing – I had to read your previous RMM-VAE paper in detail, and ultimately the RMM-VAE python code to work out what exactly was going on – and I still think I have some things wrong. I suggest some clarifications on this as follows:

Around L170 it could be useful for the readership to explicitly introduce VAEs as a generalisation of PCA as done in Murphy. E.g.

'Where PCA deterministically maps high-dimensional input data to a low-dimensional space (which due to linearity can be interpreted as a series of patterns), VAEs map input data to a low-dimensional probability distribution, normally parameterised as a multivariate Gaussian, in a space which is not directly interpretable.'

The big green arrow in figure 3 seems to imply that you model $z|t$, but you only model $z|c$ and $c|t$, correct? Could you flip c and mu/sigma in the graphic to clarify that?

In appendix A can you:
a) Reiterate the interpretation of each loss term
b) Explicitly list the various predictive models that make up the CMM-VAE with a bit more exposition about the concrete details? If I have got my head round it, its something like this:

- $q(c^k|x)$= a part of the encoder neural network, predicting probability of each class. This is the only part you actually use once the model is trained.
- $q(z|x)$ = part of the encoder neural network predicts mu and sigma for each class, then (k different?) points in z are sampled from those distributions.
- $q(t|x)$= the third part of the encoder neural network, predicting precipitation class from x
- $p(x|z)$ = the decoder neural network, reproducing x from a point in z.
- $p(z|\, c^k)$ = another mu and sigma used to generate a point in z, but based only on the class assignment. This is a linear regression? I'm also confused why its part of p, not q, as its predicting z.
- $p(c^k|t)$= a logistic regression from precip classes onto the regimes.

A diagram (like a cleaned up equivalent of this one from your last work) would be useful: