Applied Deep Learning, Exercise 1: Generating a global climate classification by clustering features extracted from Vegetation Optical depth

Leander Moesinger, ID:11937136

October 29, 2020

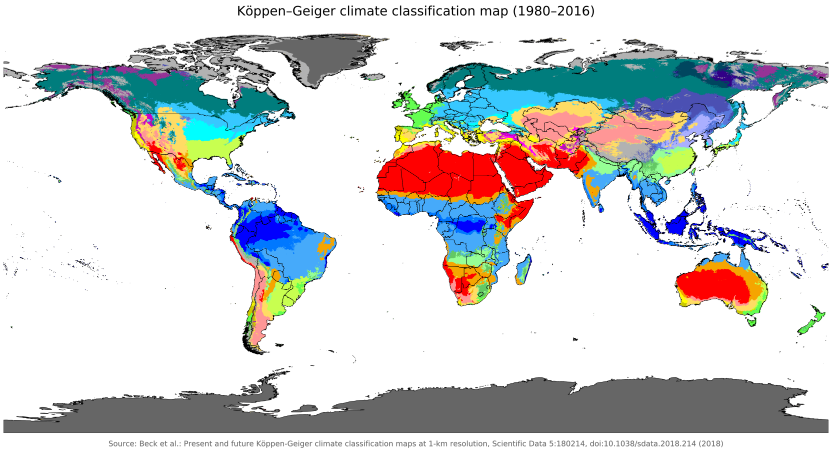
1 Topic Decision

Current global climate classifications often rely on some hand-crafted rules based on topic knowledge. E.g the Koeppen-Geiger classification (fig. 1) segments the earth into different climates based on its creators experience as a botanist and uses some logical statements using precipitation and temperature. From a practical standpoint, there probably was a lot manual tweaking involved in finding the optimal parameters to generate a map that matched the expected output. Also, the number of classes is static, therefore an user can not segment the globe into finer climate classes if desired.
We will try to automatize the whole process: Given some vegetation data as input, can we extract meaningful features from it and cluster the globe? The application will be a small program where an user can enter a number and receives a global map segmented into that many climate classes.

From a machine learning perspective, this is a "Bring your own method" topic, as it is planned to use an already existing dataset and apply some already existing methods (with some tweaks) on it. From a Geo-scientist perspective this is a "Beat the stars" topic as, as far as I am aware, no one has done this before in this field with similar data, and i am absolutely certain that no one has done this with the data i am using. It definitely could be published in a journal if extended a bit further.

2 Data

The Earth radiates microwave radiation. Part of this radiation is attenuated by water in vegetation. This degree of attenuation is described by Vegetation Optical Depth (VOD). Fig 2 shows the global mean VOD from 2002-2017. If VOD is high, such as in the tropics or Boreal forests, it means that there is a lot of vegetation or that the vegetation is very wet. If VOD is low, such as in


Köppen–Geiger climate classification map (1980–2016)

Source: Beck et al.: Present and future Köppen-Geiger climate classification maps at 1-km resolution, Scientific Data 5:180214, doi:10.1038/sdata.2018.214 (2018)

Figure 1: Up to date Koeppen-geiger classification (Beck et al., 2018).

the deserts, it is because the vegetation is dry or because there is no vegetation in the first place. VOD is also dynamic over time (fig 3), following the vegetation growth. VOD is unitless an theoretically ranges from 0 (no attenuation) to infinity (all radiation is attenuated), but practically ranges from 0 to about 1.

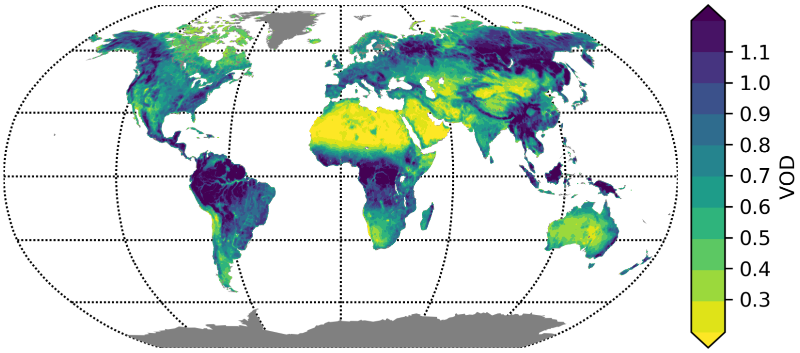


Figure 2: Mean VOD between 2002 and 2017 (Moesinger et al., 2020).

VOD contains a lot of information about the vegetation, and therefore it should be possible to use it to generate a climate classification from it. But instead of using some hand drawn features, we will use machine learning to do this automatically. This is not a trivial problem. For example, the mean VOD is very similar in the Tropics and the Boreal forests (fig 2), but very different types of vegetation are present on both locations. The temporal VOD pattern for both locations is quite different however, as tropical VOD has almost no seasonal cycle, while boreal VOD has a very strong seasonality. Therefore we

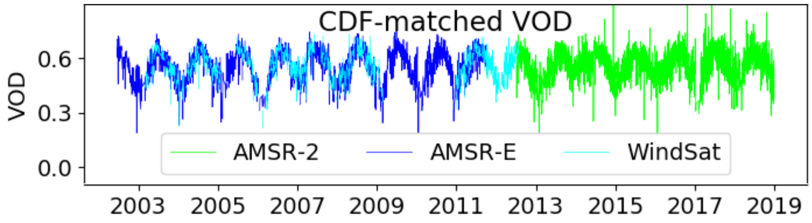


Figure 3: VOD time series measured by various spaceborne sensors for a location in Austria. It is high in summer, low in winter (Moesinger et al., 2020).

need a machine learning algorithm that is able to learn patterns in time series, such as neural networks. Special challenges will likely be that the input time series contain gaps, the signal is rather noisy (e.g. see fig. 3) and that the data volume is quite large (about 250GB uncompressed).

We are going to use VOD distributed via the VOD Climate Arcive (VODCA), an open-access[1] VOD database with global extent, that stretches over the past 30 years. It has a quarter degree spatial resolution and a daily temporal resolution (Moesinger et al., 2020).

3 Methods

We will use an autoencoder to extract features and cluster them with a shallow learner, very similar to Richard et al. (2020) or some parts of the architecture in Tavakoli et al. (2020).

An autoencoder (fig. 4) is an encoder/decoder setup that takes a VOD time series as input, encodes it into some low dimensional latent representation, and then tries to reconstruct the original time series again. In detail, the autoencoder will use convolutions to detect patterns and a symmetric encoder/decoder setup, as in Richard et al. (2020). An alternative would be to use recurrent neural networks, as in Ienco and Interdonato (2020).
The latent representation should contain all the information of the time series. It can be used as a features for some clustering algorithm to generate a climate classification map. As of now the idea is to use some simple shallow learner clustering algorithm, such as k-means or hierarchical clustering.

[1] https://zenodo.org/record/2575599

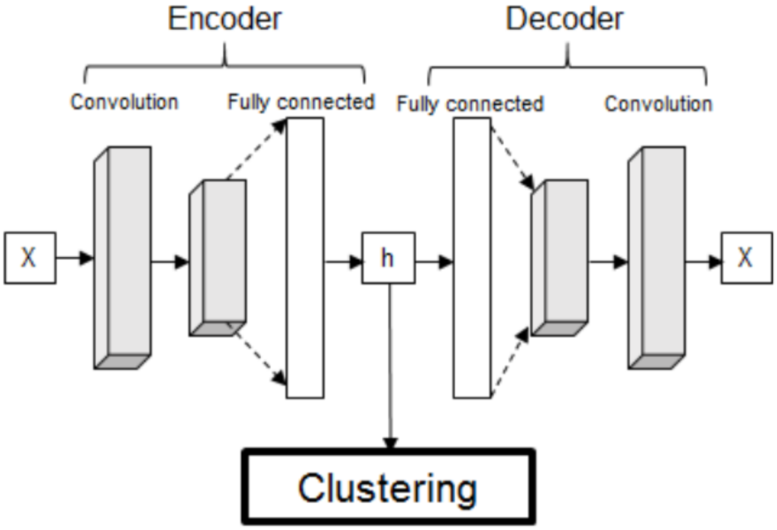


Figure 4: Proposed autoencoder setup (Richard et al., 2020).

4 Individual tasks and time estimates

- **Data gathering and preprocessing:** I have all the data and code to read it, zero hours for that part. But the preprocessing might take some time, as the current storage format is not built for reading small subsets of the data quickly, which will be nescessary for training. Maybe **1-10h**, hard to say at this point.

- **designing and building an appropriate network:** Getting up a minimal working setup working will take probably **20-30h**. I have a bit of experience with building networks with tensorflow/tensorflow-probability, but i want to use pytorch/pyro because i heard good things about it from multiple people. It will take some time to read into those packages.

- **training and fine-tuning that network:** Getting the network to actually produce something useful will take a lot of time, as it will also likely lead to some adjustments to the design. Maybe **20-30h**.

- **building an application to present the results** I have no experience at all with coding something interactive, so i guess coding the map segmentor will take me also some **10-30h**.

- **writing the final report:** I am not sure what the technical requirements on the report are. Depending on that, something like **10-30h**.

- **preparing the presentation of your work:** I am used to presentations, but preparing graphics takes time. About **10h**.

References

Beck, H. E., Zimmermann, N. E., McVicar, T. R., Vergopolan, N., Berg, A., and Wood, E. F. (2018). Present and future köppen-geiger climate classification maps at 1-km resolution. *Scientific Data*, 5.

Ienco, D. and Interdonato, R. (2020). Deep Multivariate Time Series Embedding Clustering via Attentive-Gated Autoencoder. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 12084 LNAI, pages 318–329. Springer.

Moesinger, L., Dorigo, W., de Jeu, R., van der Schalie, R., Scanlon, T., Teubner, I., and Forkel, M. (2020). The global long-term microwave vegetation optical depth climate archive (vodca). *Earth System Science Data*, 12(1):177–196.

Richard, G., Grossin, B., Germaine, G., Hébrail, G., and de Moliner, A. (2020). Autoencoder-based time series clustering with energy applications.

Tavakoli, N., Siami-Namini, S., Khanghah, M. A., Soltani, F. M., and Namin, A. S. (2020). Clustering time series data through autoencoder-based deep learning models.