Final Project Report

Passive Microwave Retrieval of Snow Density and VOD using SMAP satellite Observations at L-band

GitHub: Passive-Microwave-Retrieval-of-Snow-Density-and-VOD-using-SMAP.git

Group Members

Kati, Silas kati0012@umn.edu MS Data Science, Dept. of CSE, CSE, UMN	**Kumawat, Divya** kumaw002@umn.edu Ph.D. Civil Engineering Department, CEGE, UMN	**Ziebol, Destiny** ddziebol@umn.edu MS Data Science, Dept. of CSE, CSE, UMN

Table of Contents

1. Abstract

The recently discovered sensitivity of L-band microwave signature to snow density (SD) and vegetation optical depth (VOD) over snow-covered areas opens up a unique possibility to produce global observation datasets for these cryospheric variables. In this study, we expand the theoretical and experimental results of retrieving SD and VOD from passive L-band satellite observations from the National Aeronautics and Space Administration's (NASA) Soil Moisture Active Passive (SMAP) satellite. Such retrievals could be appealing in the context of improving Snow Water Equivalent (SWE) and vegetation indexes complementing other sensors and climate models. Retrievals using the physics-based model are time-consuming, costly, and need extensive parameterization, which is not plausible on a global scale. We propose a data-driven approach to retrieve SD and VOD on a global scale. The retrieved values for our best-performing machine learning models showed good performance (RMSE 20.981 for SD, 0.045 for VOD) on our test set, and we were able to retrieve VOD values even over the areas where our training set for VOD was missing due to the unavailability of MODIS NDVI over high latitudes. These promising results show the robustness and generalizability of the developed machine-learning models.

2. Introduction

Microwave remote sensing at the L-band frequencies (1–2 GHz) offers a unique capability for space-borne observations of Soil Moisture on a global scale. Earth's atmosphere is transparent in this frequency range, and the penetration depth is high enough in vegetation canopy and dry snow, allowing the signal of surface soil emission to traverse the snow cover and canopy. The European Space Agency (ESA) 's Soil Moisture and Ocean Salinity (SMOS) satellite and the National Aeronautics and Space Administration (NASA) 's Soil Moisture Active Passive (SMAP) satellite were launched for global monitoring of Soil Moisture (SM). The radiometer onboard SMAP provides dual-polarized observations of surface brightness temperatures at 1.4 GHz with an uncertainty of less than 1.3 K (1-sigma) at a spatial resolution of ~40 km and a temporal revisiting time of two to three days. The primary objective of the SMAP mission is to provide surface Soil Moisture and Vegetation Optical Depth estimates.

However, recent theoretical and experimental studies have indicated the feasibility of passive microwave L-band observations for observing dry snow cover characteristics, namely snow density as well. The sensitivity of L-band emission to snow density is based on the dual influence of refraction and impedance matching on observed brightness temperature with changing effective snow permittivity. On the other hand, the permittivity of pure, dry snow depends largely on snow density. In this study, we expand the theoretical and experimental results of retrieving dry snow density to passive L-band satellite observations. Such retrievals could be appealing in improving satellite-based retrievals of Snow Water Equivalent (SWE) using other sensors. Retrievals are applied to observations of the NASA SMAP radiometer on a single angle of observation.

Moreover, temperate-, tropical- and boreal forests cover more than 30% of the Earth's land surface. Their aerial extent and phenology affect the exchange of radiative energy, water, and trace gasses between the ground and the atmosphere. There is a consensus that the altered duration of boreal forests' greening period, at the large scale, impacts ground-atmosphere fluxes to a degree that is relevant for the future climate. Accordingly, remotely sensed forest phenology becomes increasingly important to further constrict modeled climate scenarios. Compared to NDVI and EVI, the so-called Vegetation Optical Depth (VOD) contains complementary information on vegetation phenology, most importantly Vegetation Water Content (VWC) and Above-Ground Biomass (AGB).

However, currently, VOD is not retrieved over snow-covered areas using SMAP. Hence in this study, we expand the applicability of the SMAP satellite observations over snow-covered surfaces by retrieving both snow density and VOD using data-driven machine-learning approaches. Our analysis focuses on snow density retrievals, as these provide the most immediate potential for enhancing present retrievals of Snow Water Equivalent (SWE).

3. Problem Statement

Since it has already been established that SMAP satellite observations are sensitive to Snow Density (SD) and Vegetation Optical Depth (VOD), retrieving these parameters using the recently developed physical models is not desirable because of the lack of parameterization of static variables over the globe. Hence in this study, we propose a data-driven approach using state-of-the-art machine learning models to solve this problem which will circumvent the issue of lack of parameterization of the physical model. In this study, we aim to achieve RMSE accuracy of snow density below 30 kg/m^3 and VOD below 0.1. Using the developed model, we also hypothesize that it is possible to retrieve VOD over the dense snow-covered areas in high northern latitudes where the training information is missing, which shows the generalizability of our developed machine learning models.

4. Datasets

4.1. SMAP Satellite Observations

Here, we use the descending SMAP level-III enhanced brightness temperatures as input to the inversion algorithms at a nominal **spatial resolution of 9 km and temporal resolution of 2-3 days**. These brightness temperatures are obtained by interpolating the along-track overlapping SMAP radiometric observations, at their native resolution, through the Backus–Gilbert interpolation technique. Several ancillary datasets available in the SMAP product are also used in this study, including effective ground temperature Tg, surface roughness parameter h, single-scattering albedo for vegetation ω, and land-cover types based on the International Geosphere-Biosphere Programme (IGBP).

4.2. Snowpack Data

The information about snowpack physical properties, including density ρ_s, and average snowpack temperature T_s are obtained from the ERA 5 reanalysis data over the globe. We consider snow density data obtained from the reanalysis dataset as ground truth for the sake of this project.

4.3. Vegetation Data (VOD)

VOD observations are constructed using the MODIS NDVI dataset (MOD13C1). The MOD13C1 data are cloud-free spatial composites of the gridded 16-day NDVI MOD13A2 product, projected on a $0.05°$ grid. We use the empirical equations to link NDVI and VOD, following the guidelines provided in the SMAP ATBD. In this approach, VOD is related to the total columnar VWC (VWC, [kgm^{-2}]) $\tau = b \times VWC$ with the proportionality value b, which depends on both vegetation type and microwave frequency. VWC is calculated directly from NDVI values using an empirical formula.

5. Data Preprocessing

Two years (2019 - 2020) of SMAP satellite observations were downloaded and preprocessed to extract only the snow-covered pixels. The target variables- snow density (SD) and Vegetation Optical Depth (VOD) are obtained from ERA-5 reanalysis data and MODIS NDVI dataset, respectively. Since these datasets have spatial and temporal resolutions different from that of SMAP satellite datasets, we mapped these two datasets temporally and spatially over the SMAP satellite grid using nearest neighbor and average downscaling approaches. The results obtained after mapping snow density and VOD are shown in Fig.2.

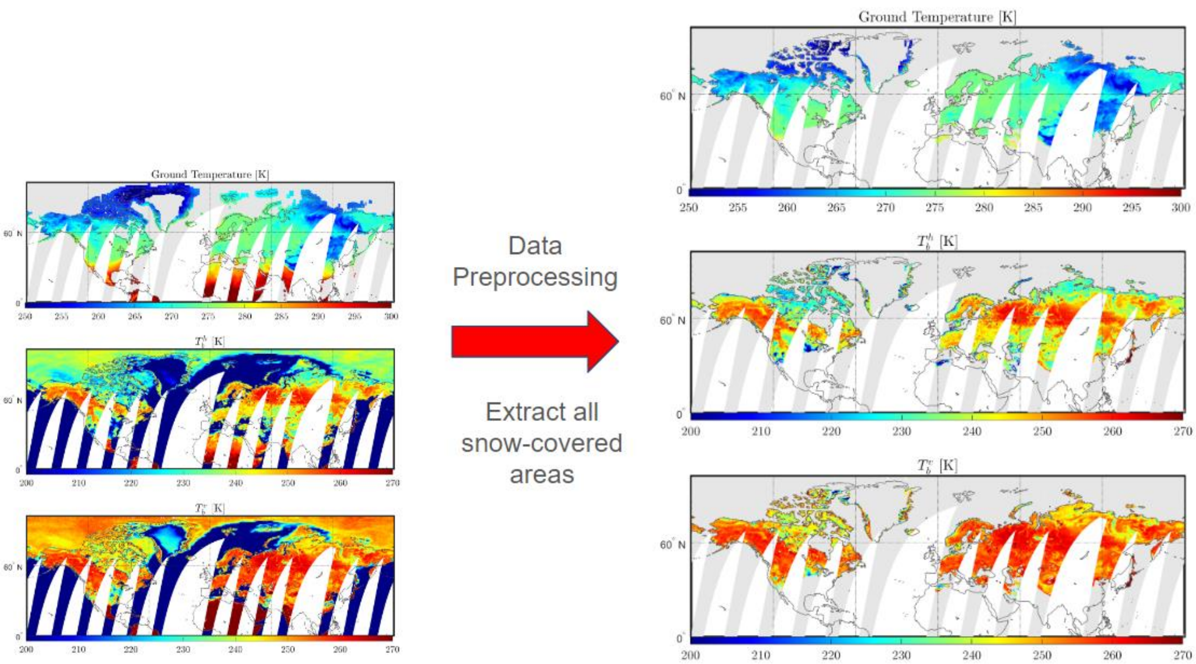


Fig.1: We extract snow-covered areas from all the SMAP satellite observations two years of data from 2019-2020.

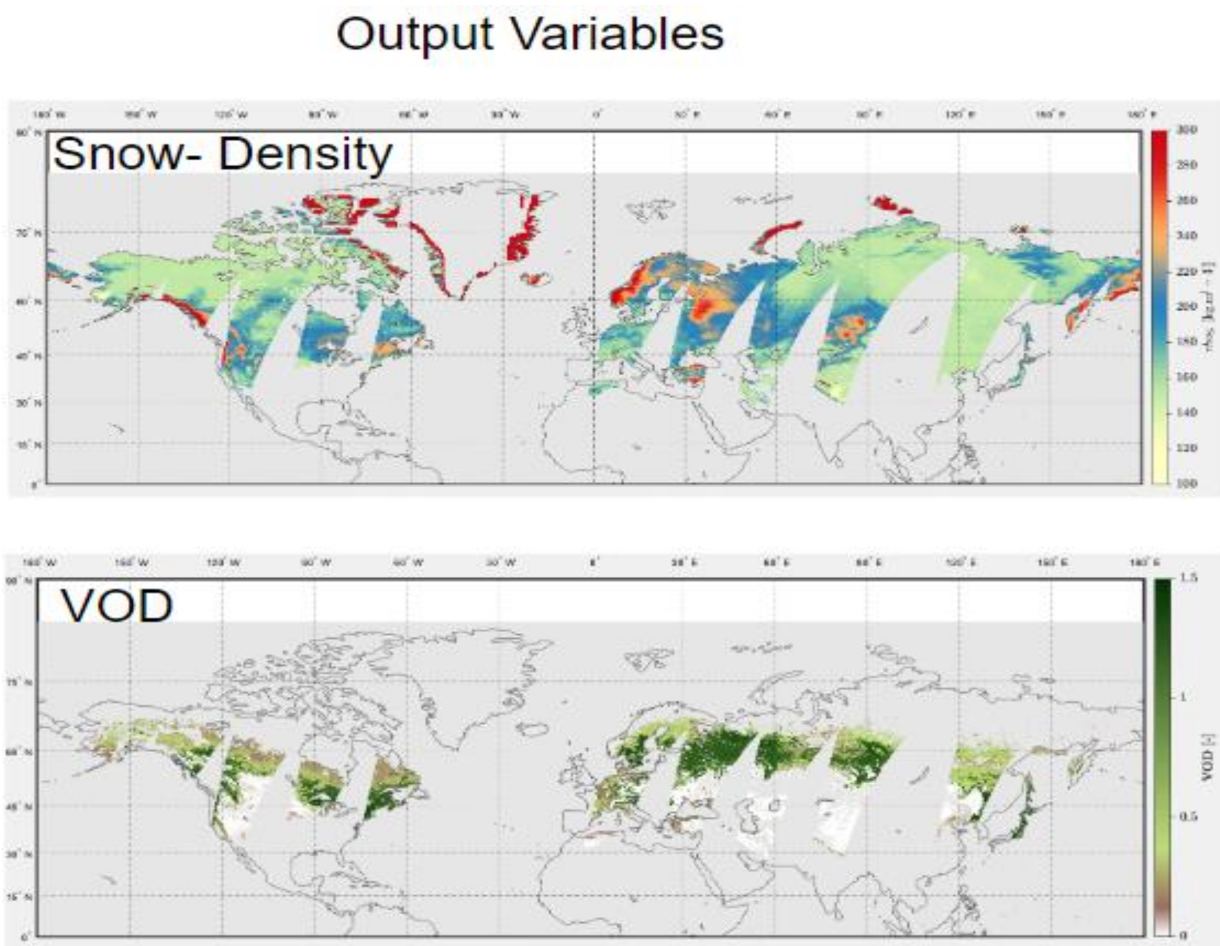


Fig.2: Spatially and temporally mapped snow density and VOD variables over SMAP satellite grid/orbit.

6. Methods

After reviewing the literature regarding similar projects (see references), we decided on five models to develop for VOD and snow density estimation: Ridge Regression, Decision Trees, Random Forest, Deep Neural Network (DNN), and Long Short Term Memory neural network (LSTM). These models rely on an inverse modeling framework - while the conventional physical model uses the inputs of snow density, VOD, ground temperature, and vegetation temperature and outputs brightness temperature, our inverse modeling seeks to exploit the implicit relationship between these variables to invert the forward emission model. We input a combination of static features such as roughness coefficient and single scattering albedo along with dynamic SMAP collected variables of brightness temperature, ground temperature, and vegetation temperature to extract snow density and VOD. This process is illustrated in Fig.3.

The inverse problem framework of the proposed problem statement is as follows:

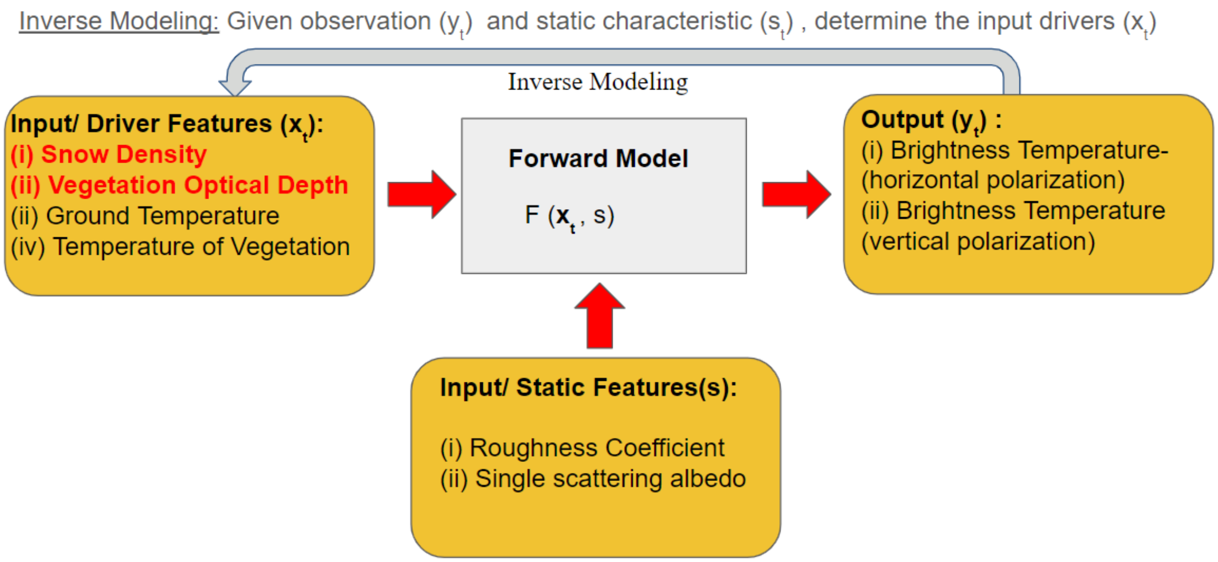


Fig.3: Schematic representation of the proposed inverse problem framework.

To prepare data for modeling, we removed all missing data rows, creating a dense set with no NANs that all of our model types could interpret. We then split our data into discrete train-test sets: all data from 2019 was retained for our train set, and all data from 2020 was set aside for our test set. After cleaning, we had 18,329,587 training records spanning the entire date range of 2019 and 13,371,200 test records spanning from Jan 1st to Nov 20th. The entire cohort of variables for all models was ground temperature, snow temperature, brightness temperature with horizontal polarization, brightness temperature with vertical polarization, roughness coefficient, clay fraction, and single scattering albedo. Optionally some models included location as latitude and longitude and VOD when predicting snow density or snow density when predicting VOD.

For every model, we collected the same performance metrics: Mean Squared error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2. Our models were coded on Jupyter notebooks in Python, with code shared, and ran through on Google Colab with a 2-core Intel(R) Xeon(R) CPU @ 2.20GHz and 13GB of RAM.

6.1. Ridge Regression

Ridge regression is a linear regression model that adds a coefficient to the model to prevent overfitting, especially in the case of multicollinearity. As two of our inputs are alternate bands of each other (vertical and horizontal polarization of brightness temperature), we expect them to be highly related, making ridge regression more suitable than regular linear regression for modeling this data set.

Our model was run using the sklearn RidgeCV function, using 10 fold validation with 3 repeats. We tested both normalized and unchanged data, making separate models for snow density and VOD. Overall performance was low for both model sets.

		Metrics		
		R^2	MAE	RMSE
SD	**RR**	**0.251**	**28.649**	**38.962**
	RR_{MinMax}	0.235	28.863	39.390
VOD	RR	0.279	0.236	0.321
	RR_{MinMax}	**0.282**	**0.234**	**0.321**

Table 1: Error metrics for Ridge Regression on test set (year 2020)

6.2. Decision Tree

The decision tree regression outperformed the ridge regression predictions, as it is a non-linear regressor. We used two training techniques. We initially only tested and trained for the year 2019, which led to positive results. However, the model's performance dropped when tested in 2020 after being trained in 2019. In this case, we discovered that autocorrelation could deceive us into thinking the model is sound. All further models were exclusively trained with 2019 data and tested on the temporally separate 2020 data.

		Metrics		
		R^2	MAE	RMSE
SD	DTR	0.392	22.199	35.100
	DTR_{MinMax}	**0.392**	**22.197**	**35.083**
	DTR_{Robust}	0.391	22.221	35.137
	$DTR_{Standard}$	0.392	22.208	35.117
VOD	DTR	0.980	0.015	0.052
	DTR_{MinMax}	**0.980**	**0.015**	**0.053**
	DTR_{Robust}	0.980	0.015	0.053
	$DTR_{Standard}$	0.980	0.015	0.053
SD with VOD as I/P	**DTR**	**0.390**	**22.236**	**35.153**
	DTR_{MinMax}	0.390	22.243	35.170
	DTR_{Robust}	0.390	22.242	35.168
	$DTR_{Standard}$	0.390	22.236	35.147
VOD with SD as I/P	**DTR**	**0.977**	**0.015**	**0.056**
	DTR_{MinMax}	0.978	0.015	0.056
	DTR_{Robust}	0.978	0.015	0.056
	$DTR_{Standard}$	0.978	0.015	0.056

Table 2: Error metrics for Decision Tree Regression on test set (year 2020)

We employed two models—one for Snow Density and the other for VOD—for each variation in preprocessing. Although tuning and pruning of the decision tree regressors were attempted, the process proved too time-consuming and was abandoned. The results are tabulated in Table 2.

6.3. Random Forest

Because the decision tree regressors were successful, we naturally investigated random forest regression. We used 10 estimators to run the random forest regressors with the same setup as the decision trees, and we compared the outcomes. We found some space for improvement. We attempted to hyperparameter tune the random forests. However, after three days of operation, it failed because of memory resource constraints. We tried to fine-tune the model using 10% of the train data in the following step. However, when we tried tuning the model using 10% of the train data, the model's performance was significantly impacted. We found that hyperparameter tuning on a subset would not help. We believe that optimizing the hyperparameters will enhance the performance of the random forest regressor model.

Additionally, we attempted to scale our data for these models, despite the fact that Random Forests and Decision Trees are unaffected by scaling, as we later discovered. The models trained and tested only on the year 2019 are named RFR_{2019} and the models trained on the year 2019 and tested on year 2020 are named RFR_{2020}. This comparison shows how the drop in performance looks when trained and tested on different years. The results are shown in Table 3.

		Metrics		
		R^2	MAE	RMSE
SD	RFR_{2019}	0.935	5.174	10.814
	RFR_{2020}	**0.531**	**19.778**	**30.846**
VOD	RFR_{2019}	0.996	0.004	0.024
	RFR_{2020}	**0.987**	**0.014**	**0.043**
SD with VOD as I/P	RFR_{2019}	0.935	5.219	10.817
	RFR_{2020}	**0.533**	**19.736**	**30.750**
VOD with SD as I/P	RFR_{2019}	0.996	0.004	0.025
	RFR_{2020}	**0.986**	**0.015**	**0.045**

Table 3: Error metrics for Random Forest Regression on test set (year 2019 and 2020)

Overall, we observe satisfactory prediction plots for the VOD variables in these models. Also, decision trees and random forests handle variance well, which worked to our advantage for VOD.

6.4. DNN

We next tried a sequential DNN. Our model had 6 hidden layers and 60 hidden units and was implemented with the TensorFlow Keras package. We used RMSE as our loss function, a prolonged learning rate (0.0001), and our batch size was 500. In this, we first attempted to predict SD and VOD simultaneously. Additionally, we tested several scaling techniques and found that standardizing the data produced the best outcomes. The models we trained and tested exclusively on the year 2019 are titled DNN_{2019}, while the models we trained on 2019 and tested on 2020 are named DNN_{2020}, just as our prior models.

			Metrics		
			R^2	MAE	RMSE
Normalized	**SD**	DNN_{2019}	0.593	16.293	25.769
		DNN_{2020}	0.516	20.284	30.191
	VOD	DNN_{2019}	0.274	0.253	0.366
		DNN_{2020}	0.278	0.232	0.328
Standardized	**SD**	DNN_{2019}	0.682	14.430	23.081
		DNN_{2020}	0.540	19.236	28.946
	VOD	DNN_{2019}	0.275	0.242	0.334
		DNN_{2020}	0.280	0.220	0.312
	SD_{only}	**DNN_{2020}**	**0.543**	**18.948**	**28.870**
	VOD_{only}	**DNN_{2020}**	**0.786**	**0.092**	**0.181**

Table 4: Error metrics for Random Forest Regression on test set (year 2019 and 2020)

Table 4 presents the results in tabular form. Given that the SD and VOD variables were on different scales and the variance in SD was significantly more than the variance in VOD, our DNN model was learning more about the SD variable while learning nothing about the VOD variable. As a result, DNN's VOD metrics were worse than our prior models. However, among all of our models so far, we attained a good performance for the SD variable. The performance of our SD variable improved as we tried to create a model for each individual variable, but the performance of the VOD did not improve as much.

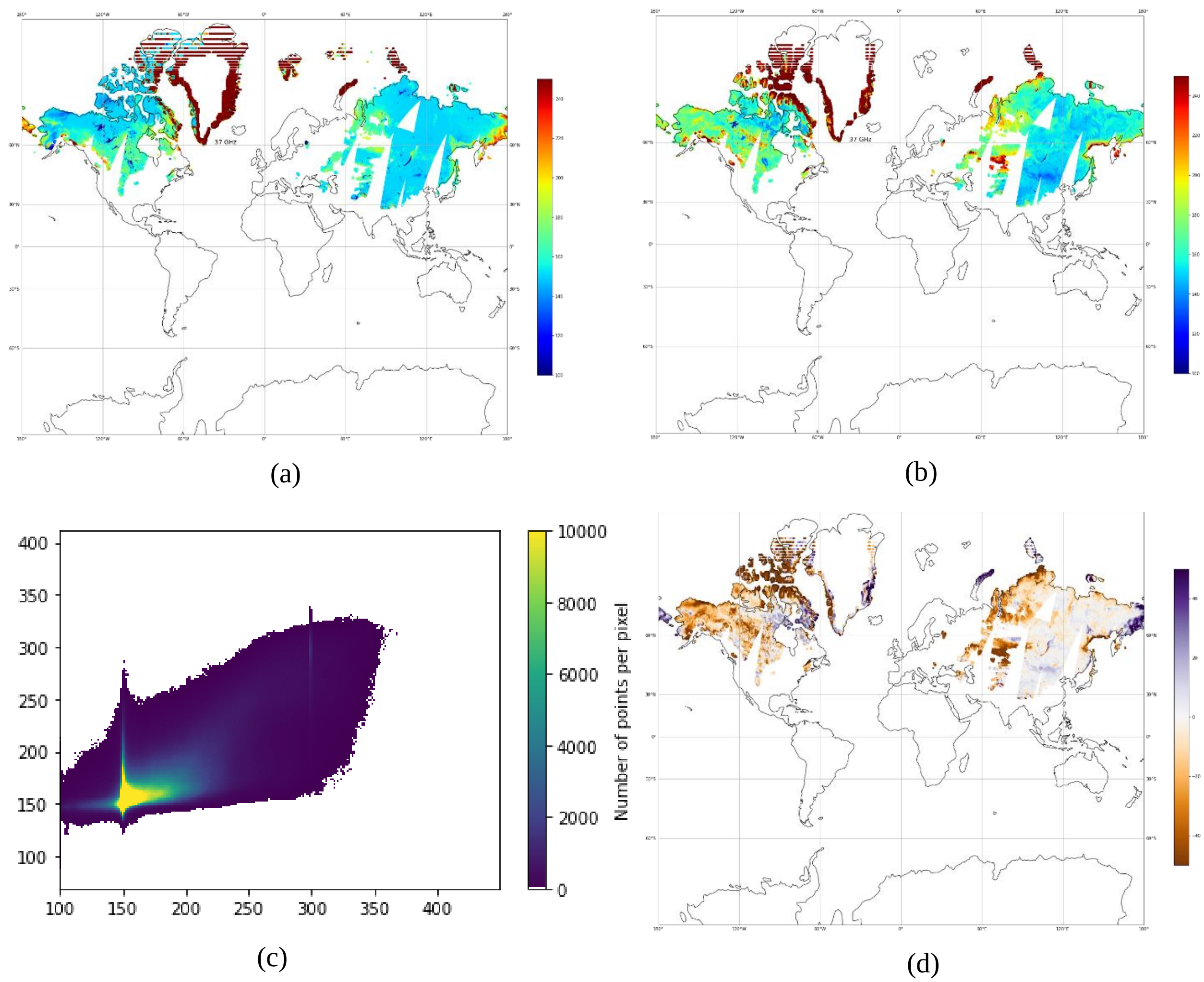


Fig.4: (a) Actual SD plot for Jan 1, 2020, (b) Prediction plot for SD from Standardized SD_{only} DNN_{2020}, (c) Density scatter plot for test set (year 2020), (d) Pixel-wise difference between actual and predicted values

Looking at Fig.4.(c) at the scatter density plots for our DNNs, we can see that while there is a hotspot for the bulk of the predictions with extremely high density, there is a substantial spread with low density. We believe that the model is overlearning on this because our dataset has a higher concentration of data values for some very narrow ranges.

6.5. LSTM

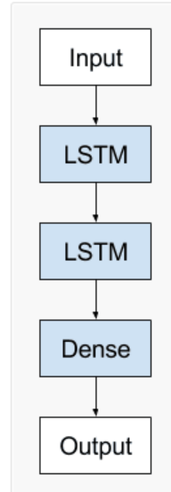


Fig.5: Stacked LSTM Architecture

Long Short Term Memory models are neural networks with the capacity to use patterns from historical data to inform prediction. Our LSTM model was implemented with the TensorFlow Keras package and included two layers, each with 60 nodes, along with two corresponding dropout layers at 20% dropout to prevent overfitting and a single dense layer at the end for prediction. 20% of the test set was reserved for validation of the model during training, mean squared error was used as our loss function, and our models were trained over 20 epochs. We used a training rate of 0.001 and a batch size of 500. Neural networks are sensitive to scaling, so all input values were MinMax scaled before use. This model did not include latitude or longitude as variables and predicted both snow density and VOD simultaneously. We chose these metrics for our architecture for consistency with our DNN (which also employed 60 nodes and 20 epochs with much success) and in the case of layer selection, to capture more complex temporal patterns that may happen over multiple time scales as compared to a single layer LSTM.

LSTMs also requires a choice of how many historical records will be supplied to the model. As our data set has discrete start times but a varying number of records per pixel, picking a window that is too large would decrease the number of pixels that can be included as well as decrease the total number of samples per pixel regardless of length. Picking a window that is too short, however, may not include enough information to capture longer term patterns. An additional consideration is that since all samples must be sequential, longer sample sizes may exclude any snow density or VOD values that are more likely at the beginning of the year. In contrast, shorter samples may be able to capture snow density and VOD values from a wider temporal range. We tested three sample lengths - 30, 60, and 80 records to address this.

Initially, sequential sampling in a rolling window was attempted, which resulted in huge data sets that were very time-consuming to make and even more time-consuming to train on. A compromise was made, and instead of sampling every possible window of 60 or 80 records in our set, a random sample was taken from each pixel that met the minimum record requirement, with one more sample taken for every ten additional records available for that pixel. The dates of the samples were randomized.

		Metrics		
		R^2	MAE	RMSE
SD	**$LSTM_{30}$**	**0.763**	**12.690**	**20.981**
	$LSTM_{60}$	0.560	23.719	34.522
	$LSTM_{80}$	0.637	22.289	34.726
VOD	$LSTM_{30}$	0.651	0.127	0.231
	$LSTM_{60}$	0.309	0.132	0.238
	$LSTM_{80}$	**0.316**	**0.106**	**0.191**

Table 5: Error metrics for LSTM on test set (year 2020)

We saw performance change dramatically as sample size increased, with snow density predictions worsening with larger sample sizes and VOD predictions improving. This may indicate that the temporal patterns between these two variables are on different time scales and perhaps can not be reconciled in a single two-layer LSTM model. Another possible explanation is that snow density performance is improving with the number of samples, as smaller sample sizes allow for more samples to be taken from more pixels, while VOD performance is most related to sample size instead of sample quantity.

7. Results

Our first model, Ridge Regression, performed poorly. As a linear model, this model was constrained in its approach, and we believe, based on the formula for the physical model, that the underlying relationships between variables are not linear. Ridge Regression was quickly discarded in favor of more complex models.

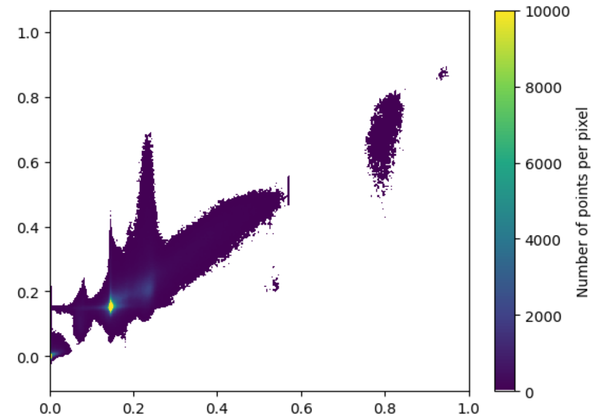
We saw our predictions quickly improve with the implementation of non-linear models. Decision tree regression had a comparatively good performance, especially for VOD, but even the best-performing model failed to clear our < 30 RMSE goal for snow density. The RFR model improved on DTR significantly, nearly reached our goal for snow density RMSE, and was our best-performing model for VOD, easily clearing our goal of <0.1 RMSE with an RMSE of 0.043 on the VOD model without standardization.

		Metrics		
		R^2	MAE	RMSE
SD	RR	0.235	28.863	39.390
	DTR_{MinMax}	0.392	22.197	35.083
	$RFR_{w/\ VOD}$	0.533	19.736	30.750
	DNN_{Std}	0.543	18.948	28.870
	$LSTM_{30}$	**0.763**	**12.690**	**20.981**
VOD	RR_{MinMax}	0.282	0.234	0.321
	DTR_{MinMax}	0.980	0.015	0.053
	$RFR_{w/\ SD}$	**0.986**	**0.015**	**0.045**
	DNN_{Std}	0.786	0.092	0.181
	$LSTM_{80}$	0.316	0.106	0.191

Table 6: Best performing model for each model type

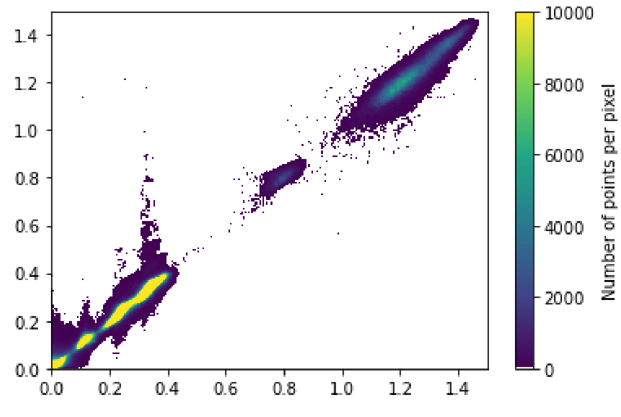
We tried two neural networks, DNN and LSTM. LSTM was our best performer for snow density, although both DNN and LSTM cleared our snow density performance goal. LSTM had our best overall performance for snow density at the lowest sample size tested, with 30 historical samples for each prediction. This suggests that even smaller sample sizes may be viable or even preferable for snow density predictions.

All of our models are able to predict VOD beyond the space where any ground truth exists, $65°N$. VOD does, of course, exist beyond this markaton, but current remote sensing instruments fail to give us data for significant parts of the year due to a lack of light. For preliminary comparisons, we have illustrated the climatology of VOD in Fig.7. As it can be clearly observed from Fig.7, we can see similarities between our retrieved values and VOD climatology over Artic and Boreal forest region. Nonetheless, more robust validation needs to be carried out in our future work. Validating these findings would be difficult and may require boots-on-the-ground data collection, depending on the level of validation rigor required. By extrapolating VOD in this region, we may be able to provide meaningful data for climate models.



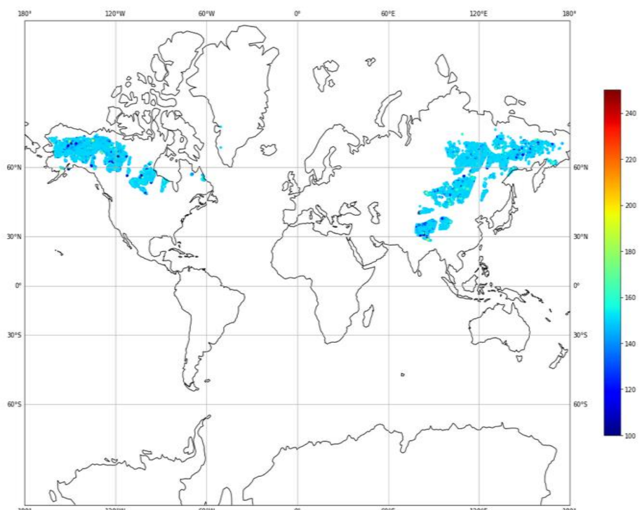
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
10000
8000
6000
4000
2000
0
Number of points per pixel

(a)

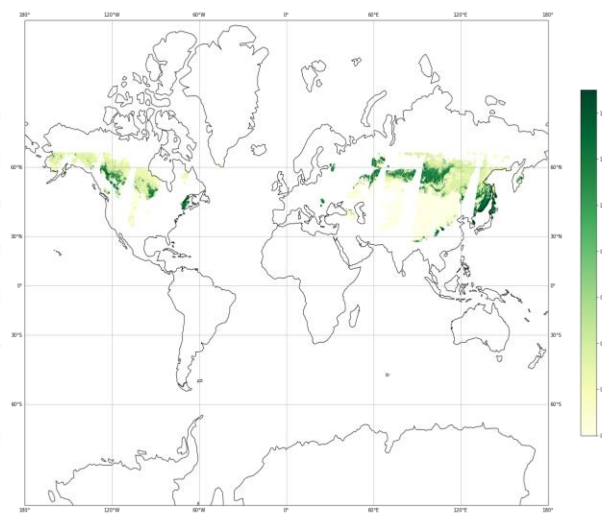


1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
10000
8000
6000
4000
2000
0
Number of points per pixel

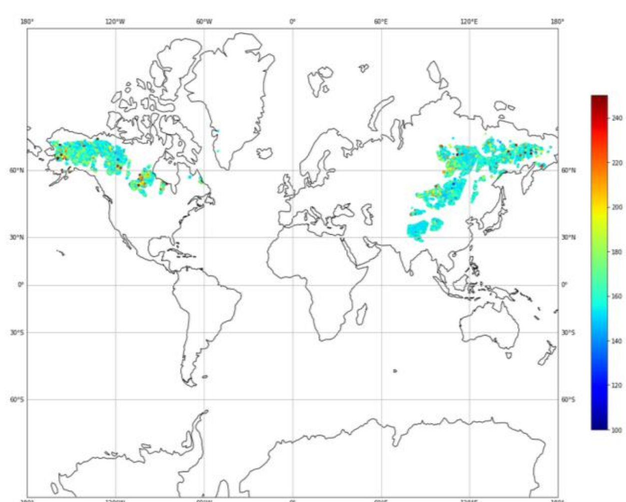
(b)



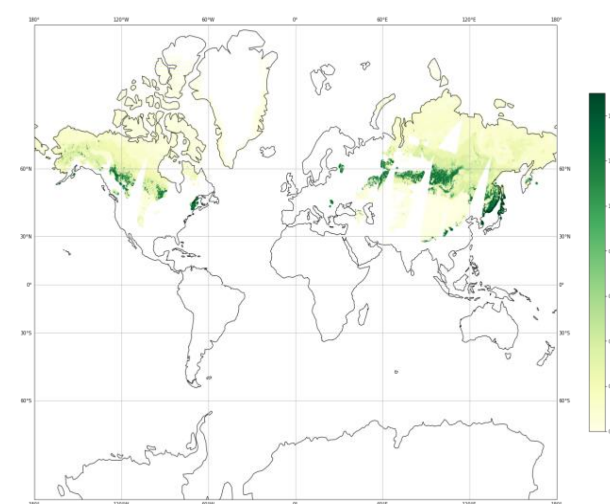
(c)



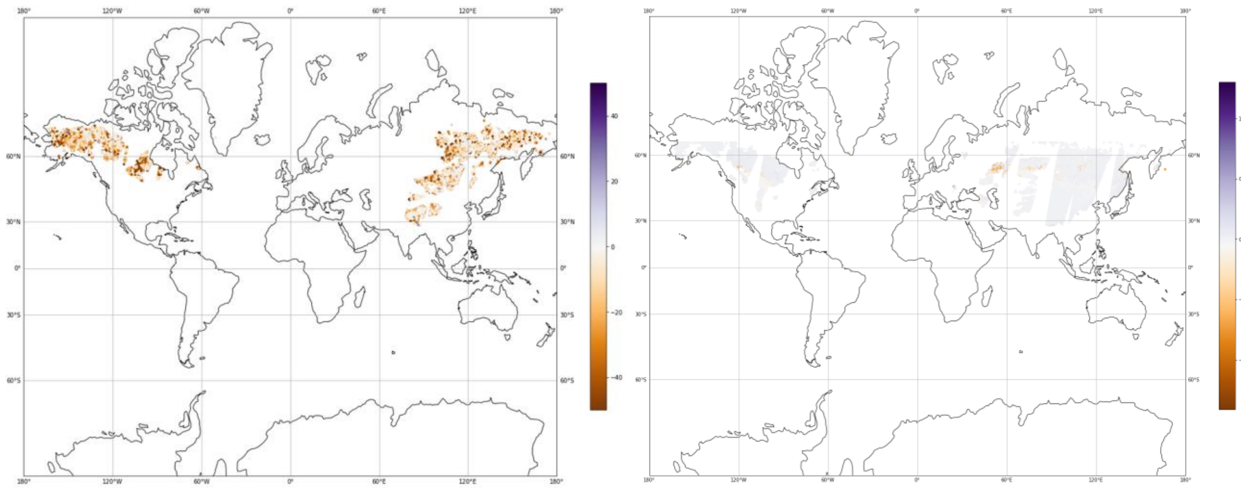
(d)



(e)



(f)



(g) (h)

Fig.6: (a)-(b) Density scatter plot for SD (left is $LSTM_{30}$) and VOD (right is $RFR_{w/\ SD}$) on test set (year 2020), (c)-(d) Ground Truth for SD (left) and VOD (right), (e)-(f) Prediction plots of SD (left is $LSTM_{30}$) and VOD (right is $RFR_{w/\ SD}$), (g)-(h) Pixel-wise difference between actual and predicted values for SD (left is $LSTM_{30}$) and VOD (right is $RFR_{w/\ SD}$)

8. Future Work

There were many parts of this project that we started but were unable to finish due to time constraints, such as knowledge-guided machine learning (KGML), convolutional neural networks (CNNs), exploring feature importance in our models using SHAP or permutation importance, comparing VOD estimates above $65°N$ to leaf area index and tree height data, and hyperparameter tuning on our many models. Any of these would make excellent extensions of this project. Hyperparameter tuning, in particular, should be the first next step for anyone looking to publish this work. A natural extension for comparison and an intuitive next step for this work would be to implement convolutional neural networks (CNNs). LSTMs are designed to work with temporal autocorrelation, and CNNs are adapted to work with spatial autocorrelation. As our data set has both, further work could even implement a newer hybrid CNN LSTM model.

Additionally, more data could be incorporated into VOD and snow density models, such as historical weather data, raw MODIS satellite data or NDVI, and some scant ground truth data. These data can augment the dataset used in this project or be used for competing model design and feature engineering.

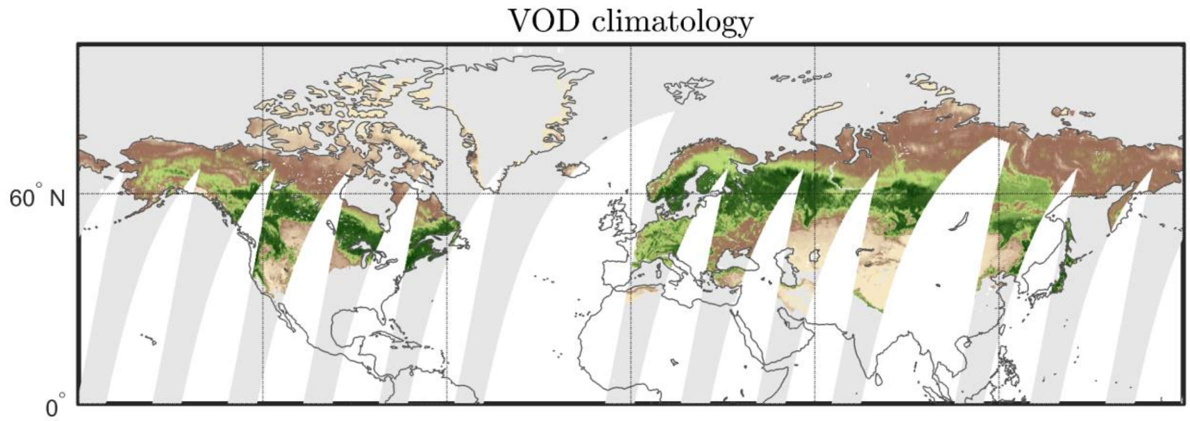


Fig.7: VOD climatology from 15 years of MODIS NDVI data. We use this to compare our VOD retrievals above $65°N$ where we do not have the ground truth data.

9. Conclusion

Our goal from the outset was to create models that could predict VOD and snow density while taking into account the static parameters of a given location as well as take advantage of the more reliable readings from microwave satellites that are not impacted by cloud cover. Minimum RMSE goals were chosen for their relevance in the field as 'fair' amounts of error can be tolerated when being used for further modeling.

We tested five model types, and by focusing on exploring the base pros and cons of each, we were able to try a wider variety of models and find the ones that were most suited to our problem. Two of our models, Random Forest for VOD and DNN for snow density, beat our goal RMSE while making predictions using both static parameters and SMAP satellite data. Overall, our project has been successful, although there are many extensions to this research that could possibly increase prediction accuracy.

10. References

1. R. Naderpour, M. Schwank, C. Mätzler, J. Lemmetyinen, and K. Steffen, "Snow Density and Ground Permittivity Retrieved From L-Band Radiometry: A Retrieval Sensitivity Analysis," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 10, no. 7, pp. 3148-3161, July 2017, DOI: 10.1109/JSTARS.2017.2669336.

2. J. Lemmetyinen et al., "Retrieval of snow parameters from L-band observations - application for SMOS and SMAP," 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2016, pp. 7067-7070, DOI: 10.1109/IGARSS.2016.7730843.

3. Juha Lemmetyinen, Mike Schwank, Kimmo Rautiainen, Anna Kontu, Tiina Parkkinen, Christian Mätzler, Andreas Wiesmann, Urs Wegmüller, Chris Derksen, Peter Toose, Alexandre Roy, Jouni Pulliainen, "Snow density and ground permittivity retrieved from L-band radiometry: Application to experimental data," Remote Sensing of Environment, Volume 180, 2016, Pages 377-391, ISSN 0034-4257, DOI: 10.1016/j.rse.2016.02.002.

4. Abebe Mohammed Ali, Roshanak Darvishzadeh, Andrew Skidmore, Tawanda W. Gara & Marco Heurich (2021) Machine learning methods' performance in radiative transfer model inversion to retrieve plant traits from Sentinel-2 data of a mixed mountain forest, International Journal of Digital Earth, 14:1, 106-120, DOI: 10.1080/17538947.2020.1794064.

5. Brence, J., Tanevski, J., Adams, J. et al. Surrogate models of radiative transfer codes for atmospheric trace gas retrievals from satellite observations. Mach Learn (2022), DOI: 10.1007/s10994-022-06155-2.

6. Venkat Lakshmi, "Remote Sensing of Soil Moisture," International Scholarly Research Notices, vol. 2013, Article ID 424178, 33 pages, 2013. DOI: 10.1155/2013/424178.

7. SMAP Dataset: https://smap.jpl.nasa.gov/.

11. Appendix

➢ Contributions and Learning

❖ Silas

Contributions: Random Forest Regression, Decision Tree Regression, and DNN were the tasks I worked on. The models underwent several iterations and combinations. I tested several feature selection and scaling methods (MinMax, Robust, and Standard scalars) on each of these models. I attempted to perform hyperparameter tuning for the models, but they were abandoned because they took too long. I made the visual projections and plotted them on maps. The table representations and report formatting were my responsibility.

Course takeaways: I am very happy that I enrolled in this course this term. Learning about remote sensing research and projects was a fantastic experience. It was helpful to gain knowledge about the data science methods used in remote sensing to address challenging but crucial issues. The paper presentations were an essential component of the learning process since they provided a thorough understanding of multiple studies and works done by people, each of which solved an intriguing challenge. All of these publications' reviews, which I read and wrote, improved my reading and writing abilities as a data scientist.

Project takeaways: We had all first thought of some very intriguing challenges to solve when it came to our project, but we ultimately opted to go with this one, and I am pleased we did. This concept is somewhat fresh to me in terms of my skill and knowledge of the subject. However, I have learned a lot from Divya about the problem we are trying to solve, and I have also learned how I can use my data science expertise to help with an issue that is brand-new to the field. It was enjoyable to understand and put what I had learned in my previous classes into practice, as well as to see some progress and outcomes. We wish we had some more time to wrap up some of the tasks we had intended to achieve in the last few weeks but ended up taking more time. Things I learned from our project were Decision Trees, and Random Forests are invariant to scaling, how hyperparameter tuning can be done to models, making a custom DNN, how data imbalance and variance affect DNN, using new libraries for plotting and visualizing. If I could start over, there is one thing I would change: I would give tuning all the models a better go with the goal of improving their performance. We experimented with several various models, which was in and of itself an outstanding learning experience.

❖ Destiny Ziebol

Contributions: Ridge Regression and LSTM models start to finish. I finished our data prep after Silas had started it and set up a shared train/test set. I also edited PowerPoint for style and continuity and drafted scripts for our videos. Although not featured in this paper, I did spend a good amount of time trying to implement the SHAP feature importance to help clarify our non-linear models. Unfortunately, I was only successful with our linear models. I also started work on implementing CNNs but had to discard this part of the project due to time constraints. I wrote the Methods, RR, LSTM, results, future work, and conclusion sections of this paper.

Course takeaways: I was surprised by how much I enjoyed reading papers and doing the weekly write-up assignments. I know I spent too much time on them, and perhaps I was too critical, but it's enjoyable to find faults in other people's publications, given that I don't have any myself. The course discussions about papers and the interim project check-in were very informative, and I gained fluency when discussing machine learning and neural networks that I had been specifically seeking from this course. I really enjoyed the way this course took a set of similar problems and looked at them in a bunch of different ways, showing which approaches would be best given each view.

Project takeaways: One thing I specifically wanted out of this course was to implement my own neural network, and I'm happy to say that I've done that. I worked on ridge regression and LSTM models, both of which I had no experience with before this course. I do a lot of linear modeling at work, so I wanted to take the chance to learn another technique. I had coded my own neural networks for other courses, but practically, it's just about meaningless to say you can code your own from the ground up when any sane workplace expects you to use TensorFlow or PyTorch. I am happy to say I now have experience with these libraries and have gained a lot of confidence when implementing them.

It was fascinating to see how each model type responded to scaling or normalization. By implementing so many models, we were able to directly compare performance on the same tasks, which was very useful to me from a learning perspective. This project really hammered home the strengths and weaknesses of decision trees vs. neural networks.

❖ Divya Kumawat

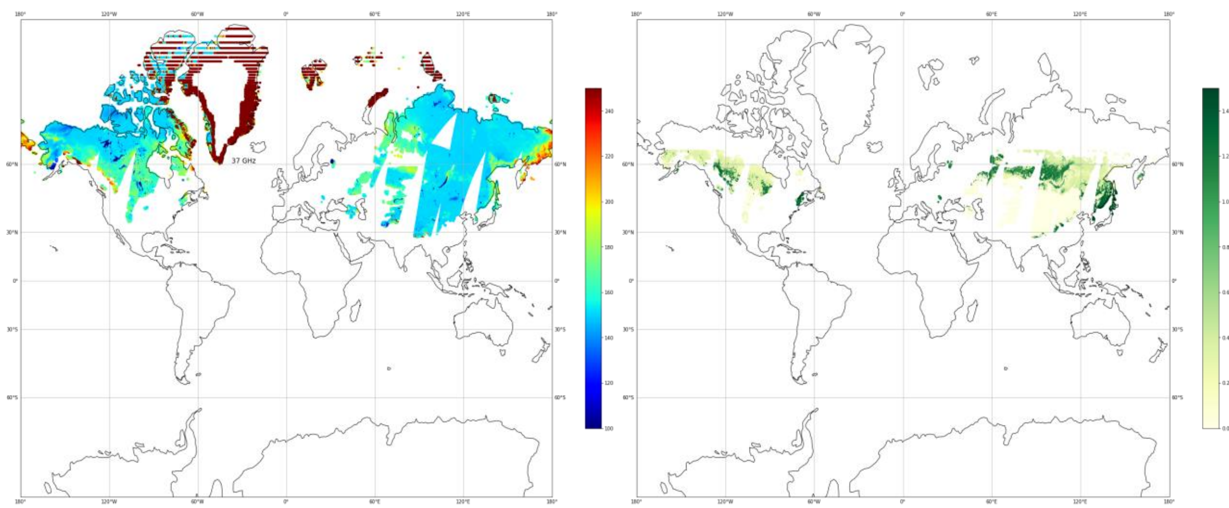
Contributions: I have downloaded all the datasets, including SMAP satellite, MODIS NDVI, and ERA 5 Reanalysis data, and preprocessed it to make it spatially and temporally consistent with the satellite data. I then converted those files into a suitable format to be used for machine learning methods for our project. I worked with Silas on the Dense Neural Network (DNN) for this project. It was a group effort to develop different innovative ways to represent the error structure for further analysis of the proposed models. This topic is related to my Ph.D. dissertation topic; hence I bring the domain knowledge to this group project. I have prepared and framed the problem statement and motivation and have contributed to improving the machine-learning algorithms presented in this paper by adding domain-specific knowledge.

Course Takeaways: The Paper presentation and learning from other's projects was important part of the learning process for me. I reviewed 5 papers, including one which I presented, and I can very confidently say that my knowledge base has been significantly improved by learning about physics explained and guided machine learning algorithms can significantly improve the state-of-the-art methods to solve those specific problems.

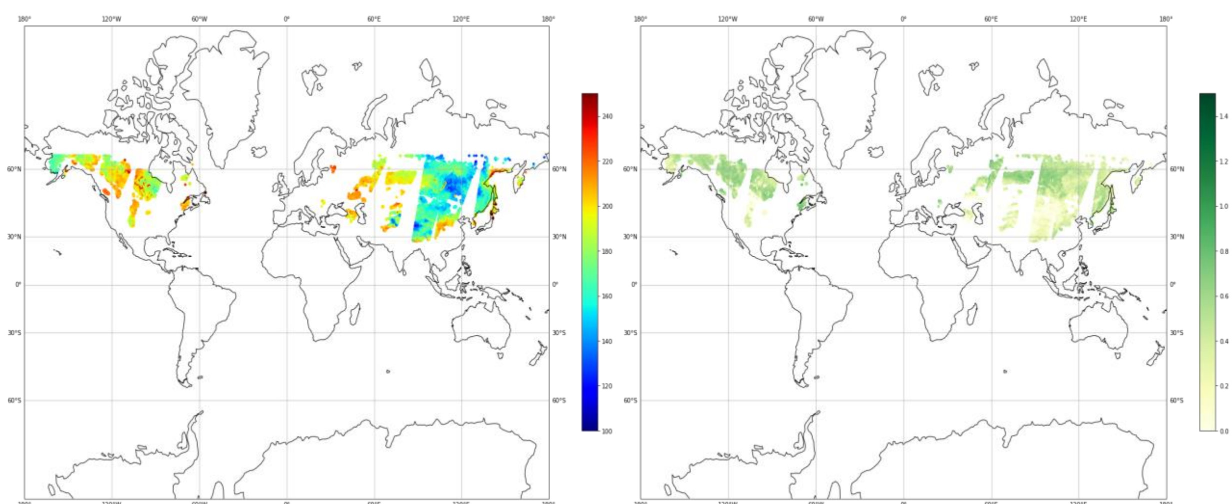
Project Takeaways: Since we tried a range of methods, from simpler ones like Ridge Regression and Decision Trees to more complex ones like DNN and LSTM, the weakness and strengths of all kinds of models come as a very natural concluding result from the performance of the models. I learned how variability in the sample set used for training DNN hugely impacts the performance in less occurring sample range. While at the same time, ridge regression is less affected by the sample distribution when it comes to regression kind of problems. It was huge learning to start a completely new problem from scratch and go through the whole process of downloading, preprocessing, testing baseline models, and delivering satisfactory results for the problem statement we started with in our introductory presentation. I would also like to mention that our group meetings were particularly helpful as everyone has a very diverse set of expertise, and we learned a lot from each other throughout the duration of this project. I intend to keep working on this project and try to improve it further by implementing our future works section.

➢ Figures and Plots

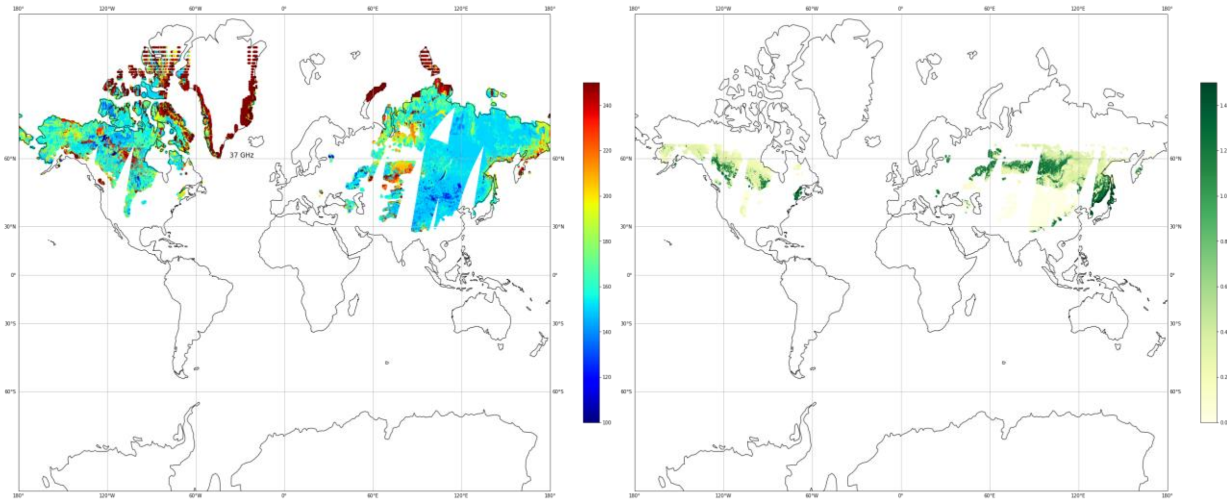
Ground truth of SD (left) and VOD (right)



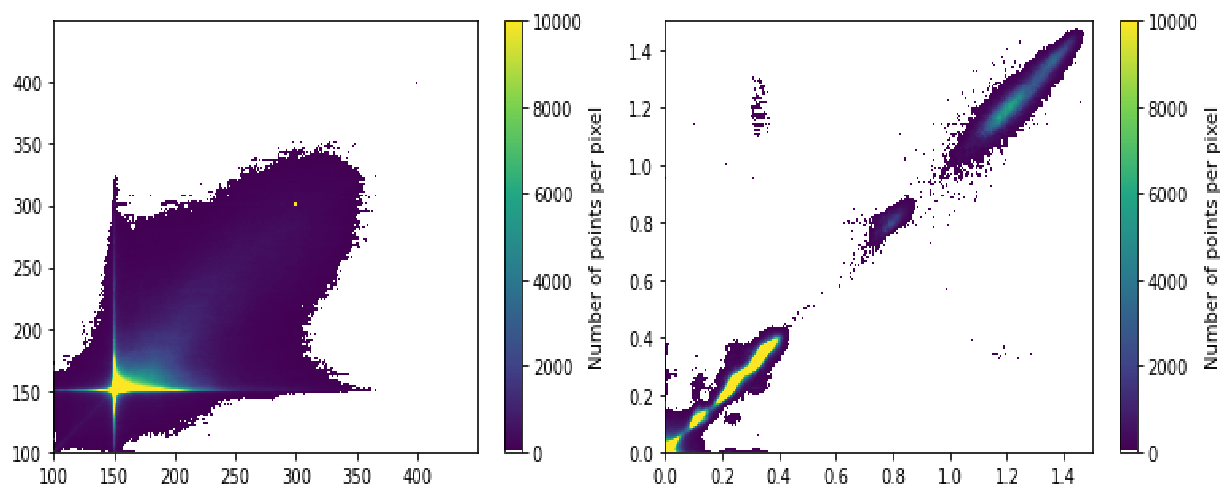
Ridge Regression predictions of SD (left) and VOD (right)



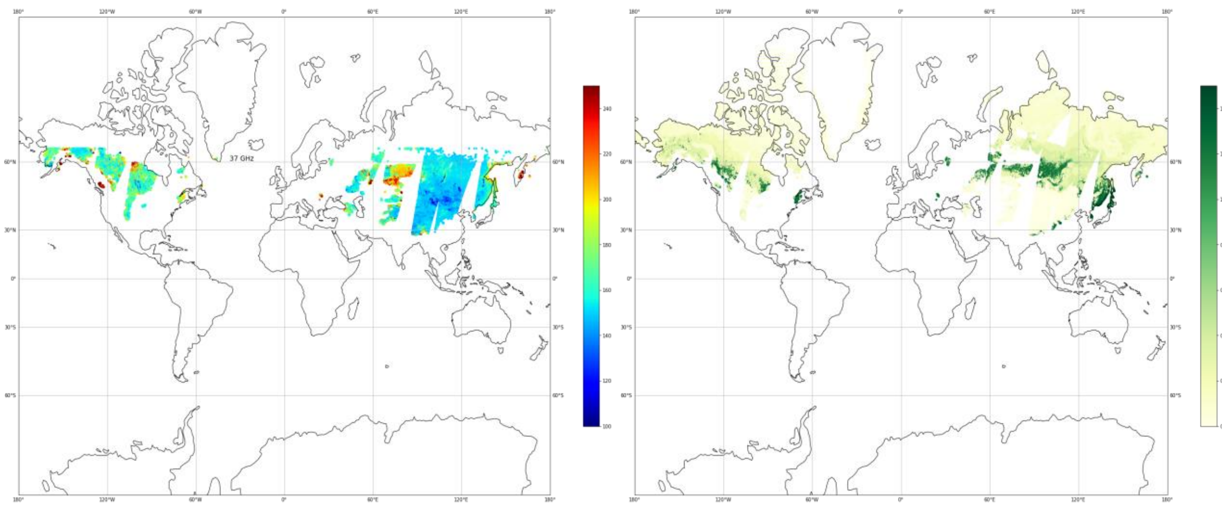
Decision Tree predictions of SD [MinMax Scaled, w/o VOD] (left) and VOD (right)



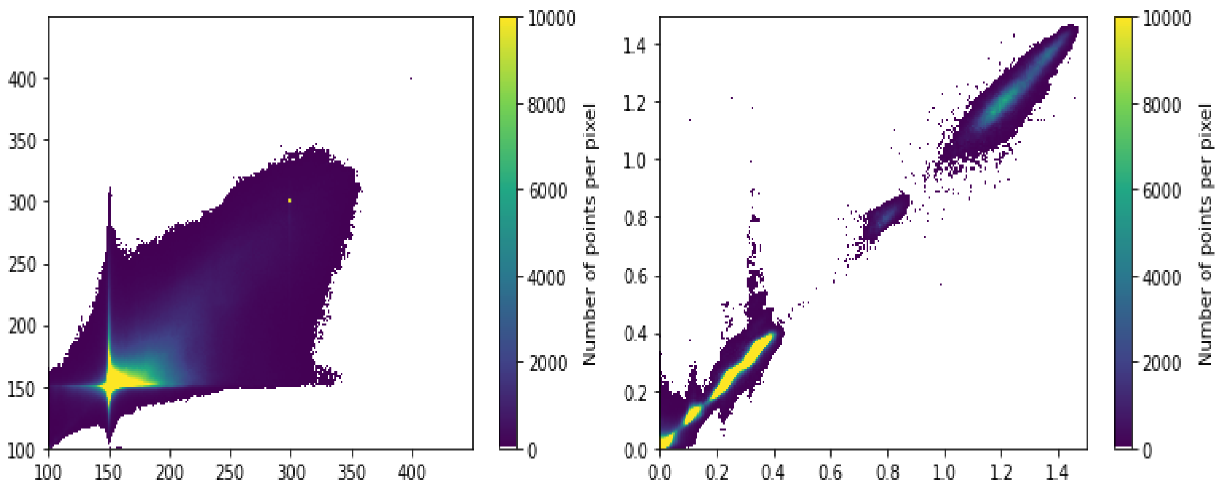
Decision Tree scater density plots for test set (year 2020) of SD [MinMax Scaled, w/o VOD] (left) and VOD [MinMax Scaled, w/o SD] (right)



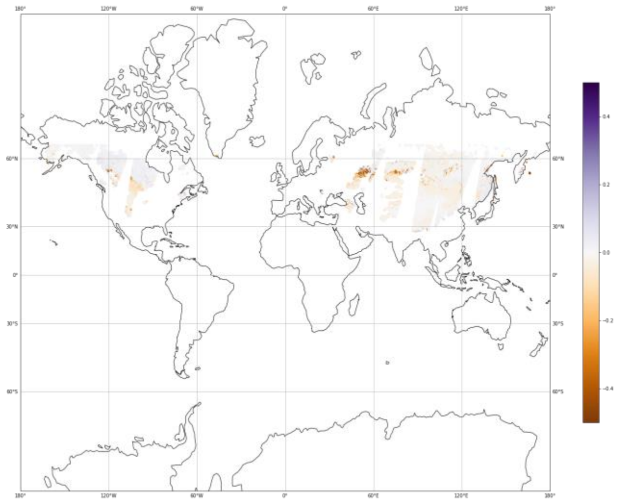
Random Forest predictions of SD [w/ VOD] (left) and VOD [w/ SD)] (right)



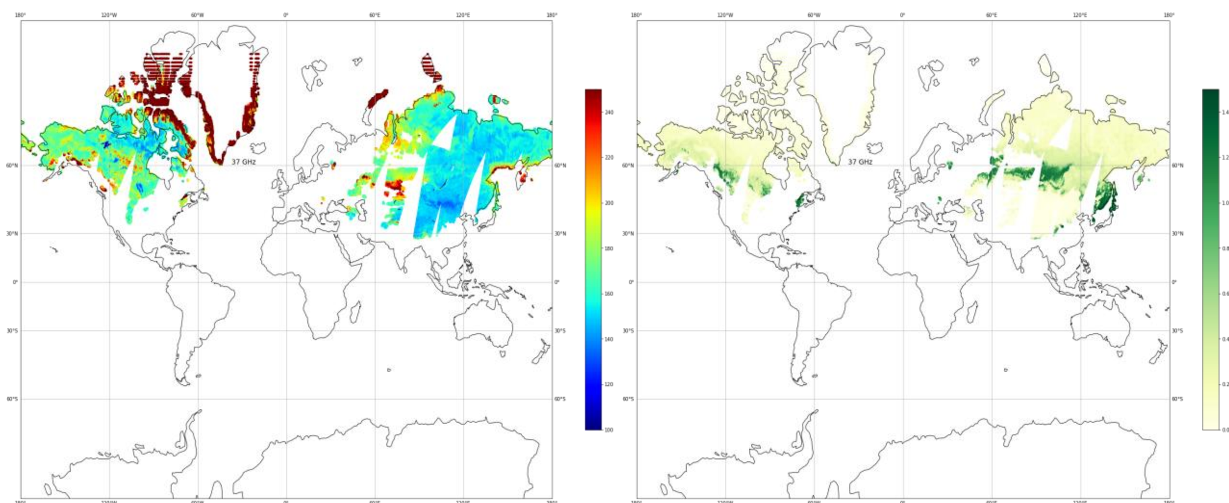
Random Forest scatter density plots for test set (year 2020) of SD [w/ VOD] (left) and VOD [w/ SD)] (right)



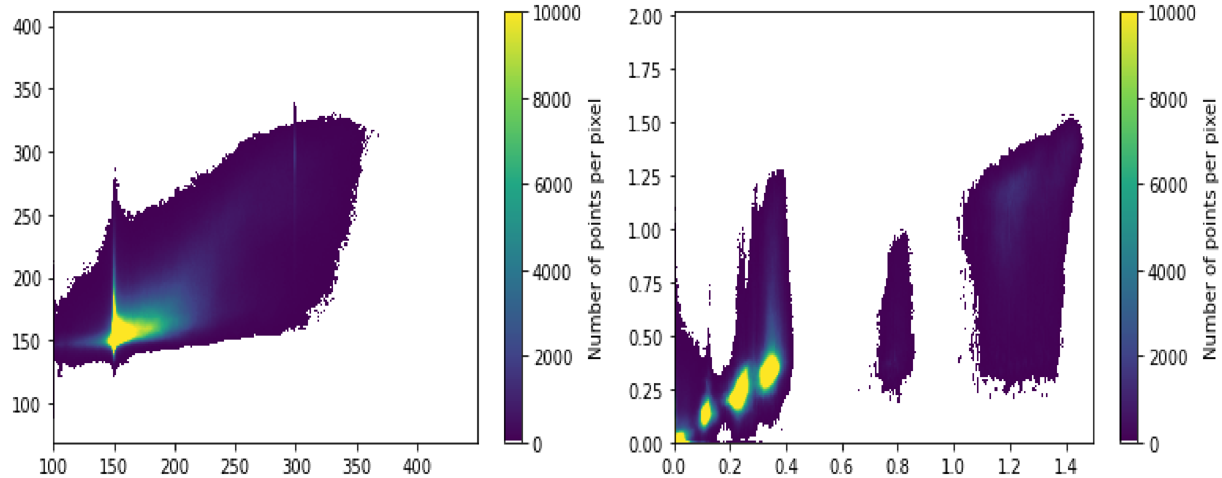
Random Forest pixel-wise difference between actual and predicted values



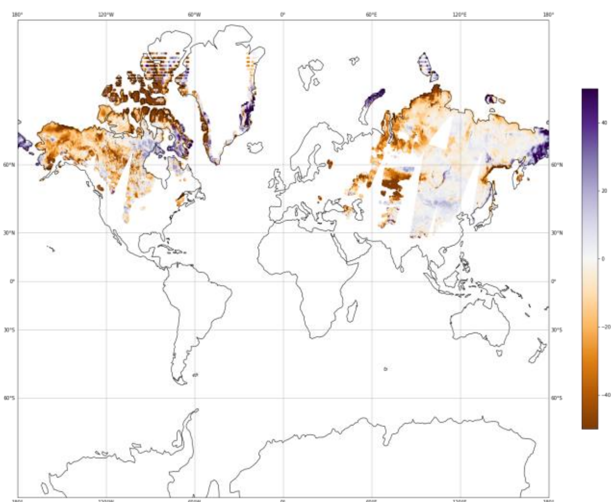
DNN$_{2020}$ (Standardized) predictions of SD (left) and VOD (right)



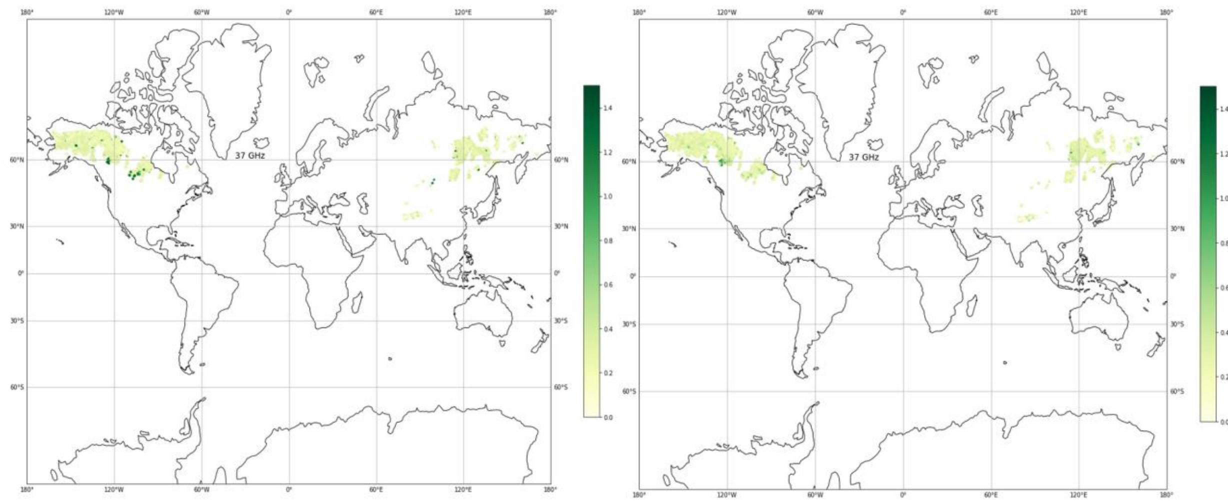
DNN$_{2020}$ (Standardized) scatter density plots for test set (year 2020) of SD (left) and VOD (right)



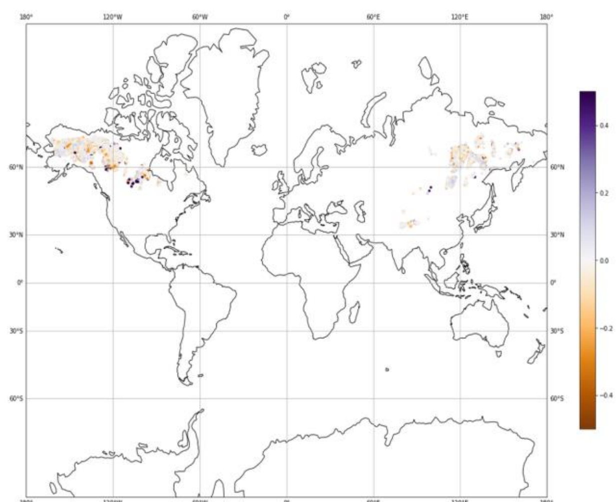
DNN$_{2020}$ (Standardized) pixel-wise difference between actual and predicted values



LSTM$_{80}$ ground truth VOD data from Nov 1st, 2020 and predictions of VOD (right)

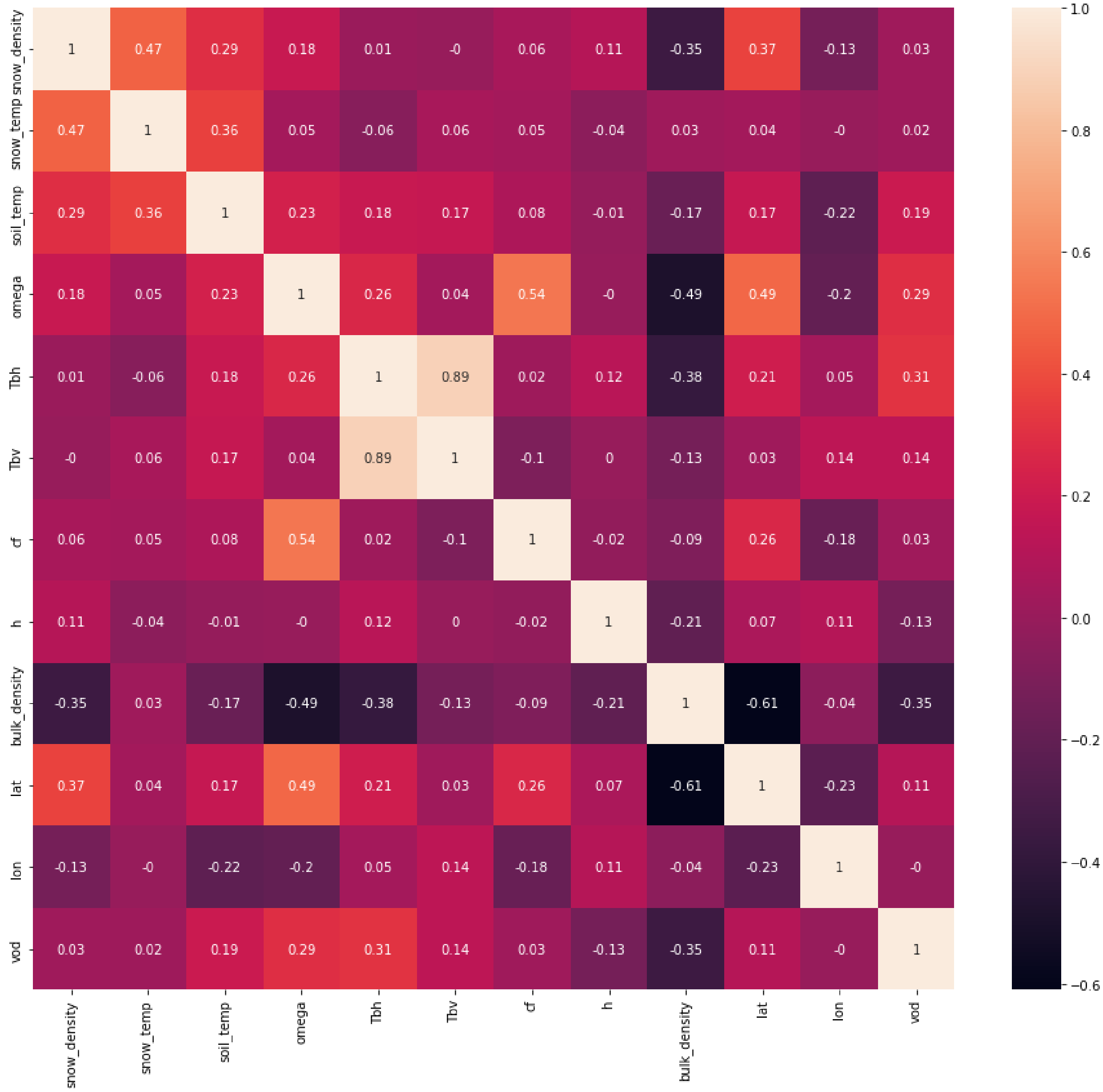


LSTM$_{80}$ pixel-wise difference between actual and predicted values

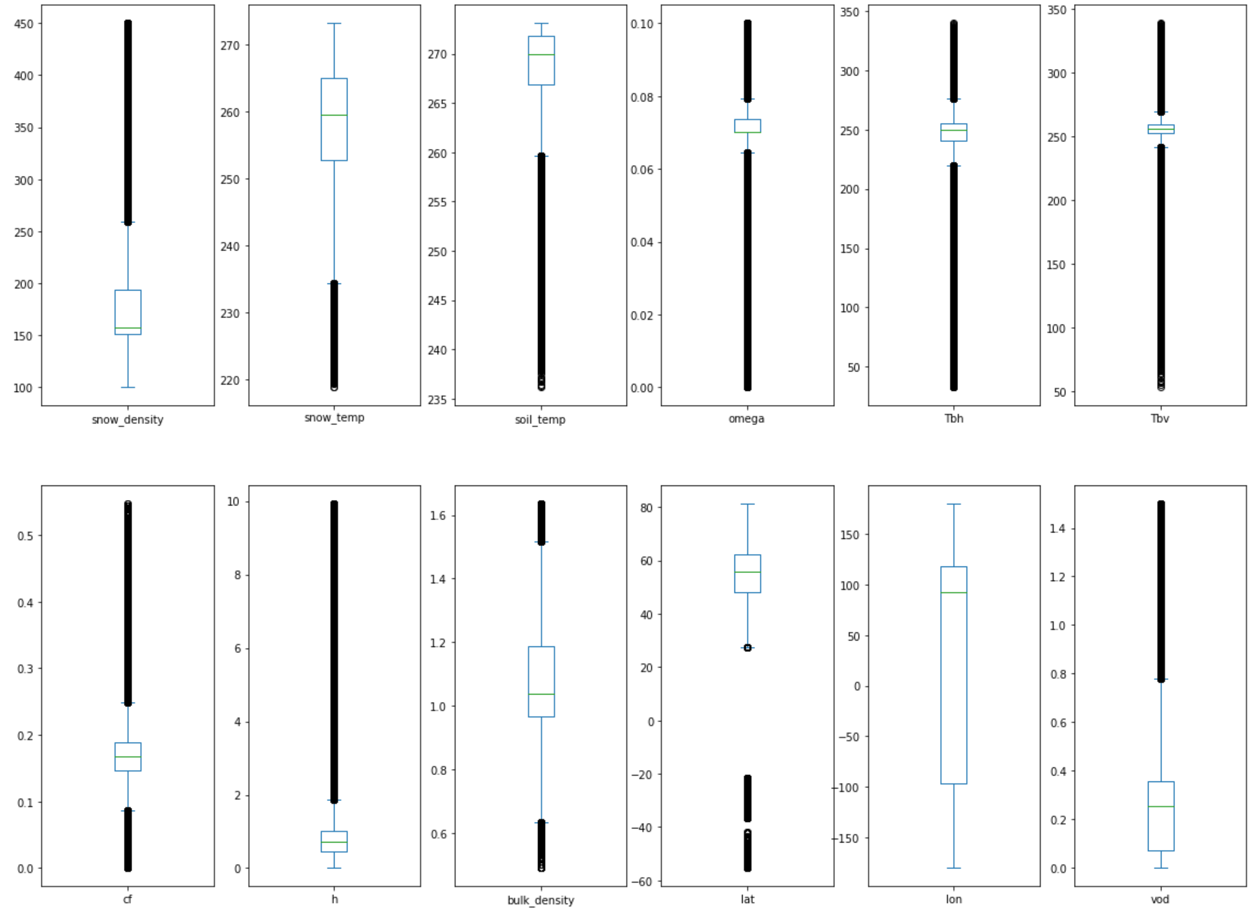


Dataset plots

Correlation heatmap of all variable

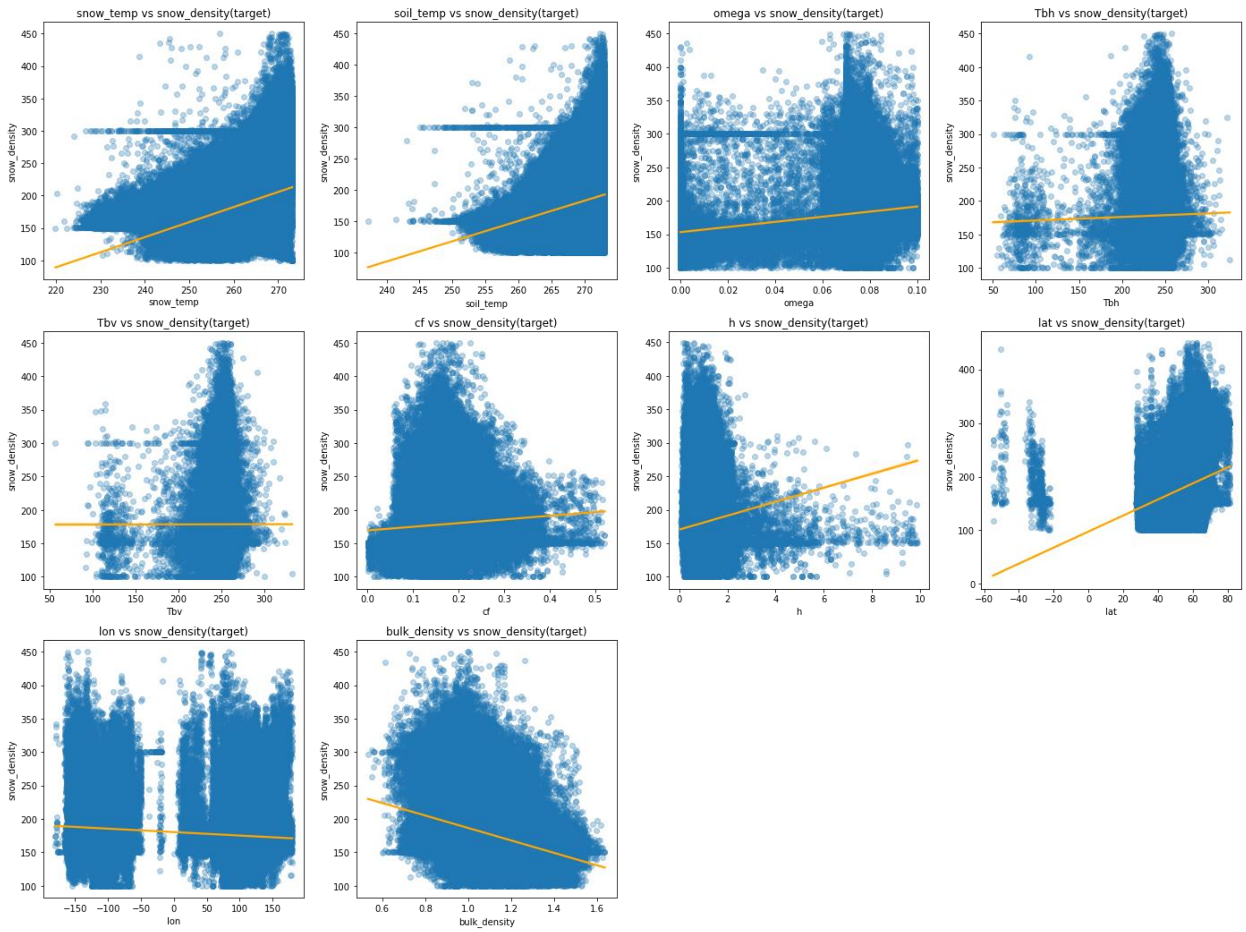


Boxplot of all variables and their scales



Regression plots

SD target variable



VOD target variable